

ELEVENTH INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION

Held under the patronage of the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT)

MAY 7– 12, 2018

PHOENIX SEAGAIA CONFERENCE CENTRE
Miyazaki**, JAPAN**

CONFERENCE PROCEEDINGS

Editors: Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, Stelios Piperidis, Takenobu Tokunaga

Assistant Editors: Sara Goggi, Hélène Mazo



The LREC 2018 Proceedings are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License

LREC 2018, ELEVENTH INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION

Title: LREC 2018 Conference Proceedings

Distributed by:
ELRA – European Language Resources Association
9, rue des Cordelières
75013 Paris
France

Tel.: +33 1 43 13 33 33
Fax: +33 1 43 13 33 30

www.elra.info and www.elda.org
Email: info@elda.org and lrec@elda.org

ISBN 979-10-95546-00-9
EAN 9791095546009

Introduction to LREC 2018 by Nicoletta Calzolari
Chair of the 11th edition of LREC
ELRA Honorary President

Welcome to the 11th edition of LREC in Miyazaki, first LREC in Asia!

LREC 20th Anniversary

It is the LREC 20th Anniversary and LREC has become one of the most successful conferences of the field. Data are pervasive in Natural Language Processing and Language Technology: we call our data Language Resources (LR). But when LREC was started by ELRA, in 1998 in Granada, from an idea of Antonio Zampolli and Joseph Mariani, it was really a new adventure and a challenge. There were well established big conferences but he thought that the new emerging field of Language Resources deserved its own dedicated forum. In the keynote talk I gave at LREC1998 I could say: "the infrastructural role of Language Resources as the necessary common platform on which new technologies and applications can be based is nowadays widely recognised." This could not have been said only few years before. I had the pleasure and the honour of being involved in LREC from the beginning, first as member of the Program Committee and since 2004 as Conference Chair.

LREC is probably the most influential ELRA achievement, and a service with the major impact on our community. Also through LREC, ELRA contributes to shape our field, making the Language Resource field a scientific field in its own right.

Why LREC in Asia this time? AFNLP (the Asian Federation of NLP) asked us if we could hold an LREC in Asia as the best instrument to promote Language Resources in Asia. We were glad to accept this challenge and here we are.

Some LREC2018 figures

As expected given the change in continent, we did not break any record this time, but the figures are not far from the previous. We received 1102 submissions for the main conference, 34 workshop proposals and 8 tutorial proposals.

A very large part of our community was involved in the reviewing effort, to be able to assign few papers per reviewer: 1263 colleagues accepted to act as reviewers (more than in 2016) out of 1796 invited (268 declined and 265 unfortunately not answering). Few reviewers did not complete the task (only 26 reviews missing, not so bad), but knowing that this always happens we recruited some pinch-reviewers able to act at the last moment: a good move to keep for the future.

The Program Committee has also been enlarged with 3 colleagues from Japan and one from USA. We had as usual a very hard job, examining about 3300 reviews, to understand – beyond the scores and in particular when they greatly differed – the relevance, the novelty, but also the appropriateness for an oral or poster presentation. I am sure we made mistakes, every reviewing effort is not immune from subjectivity, but as usual we discussed in a face to face meeting not only general policies, criteria and how to be consistent, but also borderline cases

to arrive at agreed decisions. Overall we all believe we received in average good submissions. We have in the main program 718 papers: 188 Orals and 530 Posters.

We also have 29 Workshops and 5 Tutorials.

I am proud that around 1100 participants have already registered at the end of April, similar to last time. They come from 63 countries. The Japanese are the largest group and in general there is a larger participation from Asian countries, in particular China, as we obviously hoped.

These figures have a clear significance. The field of Language Resources and Evaluation is very alive and constantly flourishing.

LREC acceptance rate: a motivated choice for an inclusive conference

The LREC acceptance rate, 65% this year, is different from other major conferences but for us it is a motivated decision. This is one of the reasons why LREC succeeds to provide a comprehensive picture of the field and to show how it is evolving. For us it is important not only to hear about new methodologies but also to understand how various methods or resources are able to spread, for which purposes, usages, applications, and for which languages. Multilingualism – and equal treatment of all languages – is an essential feature of LREC, as it is the attempt of putting the text, speech and multimodal communities together as well as academics and industrials. LREC wants to be an "inclusive" conference.

Quality is not undermined by our acceptance rate: in 2017 Google Scholar Metrics h5-index, LREC ranks 4th in Computational Linguistics top conferences (5th considering ArXiv which is the first).

LREC2018 Novelties

Industry Track

Because of the interest in joining forces between academy and industry, this time we decided to experiment with a new Industry Track. We spoke about this at last LREC with Linne Ha from Google and we asked her if she wanted to organise it for LREC2018.

Special Speech Session

A special session on "Speech resources collection in real-world situations" was proposed to us by Kikuo Maekawa and Yuichi Ishimoto (National Institute for Japanese Language and Linguistics): we gladly accepted also to strengthen the participation of the speech community at LREC.

Oriental-COCOSDA Conference

Also O-COCOSDA is organised together with LREC. We spoke with Satoshi Nakamura, its chair, at last LREC and he kindly offered to organise it jointly with LREC. We are very pleased of this also because it is another opportunity to reach the Asian speech community.

ELRA Individual Members Assembly

ELRA has recently introduced "individual membership" in addition to institutional membership. This was decided to give a voice inside ELRA to the large LREC community and offer them its services. The first assembly of ELRA individual members is held on the first day of the conference.

The LREC Club

From the answers received, it seems that the LREC Club of those who attended all editions, the really faithful ones, is composed of 23 members. I want to thank them for their loyalty!

LREC2018 Trends

I quickly sketch here, as I always do, my perception – subjective and impressionistic – of LREC2018 trends and how certain topics fluctuate from an LREC to the other. The comparison with previous years shows the topics with steady progress, or even great leaps forward, the stable ones and those more affected by the fashion of the moment.

Trends in LREC2018 topics

Among the areas that continue to be trendy and are even increasing I can mention:

- Less-Resourced Languages
- Social Media analysis, appearing in 2012 and since then constantly growing
- Semantics in general and in particular Sentiment, Emotion and Subjectivity
- Information extraction, Knowledge discovery, Text mining are booming
- Lexicons (in its various forms)
- Discourse, Dialogue, Conversational systems and Interactivity
- Multimodality, also for Less-Resourced languages
- Tools, Systems, Applications for various purposes: Question Answering, Summarisation, etc.
- Evaluation methodologies
- Computer Aided Language Learning

Stable "usual" topics, some very well-represented, others in the medium/low range, are:

- Infrastructural issues, policies, strategies and Large projects: topics that receive special attention at LREC, differently from other major conferences
- Corpus creation, annotation, use, …
- Speech related topics, a little increasing but not as much as we would like
- Sign language (also a very successful workshop)
- Crowdsourcing
- Anaphora and Coreference
- Temporal and Spatial annotation

New trends for this LREC:

- Digital Humanities (new for LREC in 2016, now increased)
- Bibliometrics, Scientometrics, Infometrics
- Language Modelling

Decreasing topics with respect to the past, even if some still numerous:

- Grammar and syntax and also Treebanks that had a big increase in 2016
- Multilinguality and Machine Translation, very high in 2016
- Ontologies
- Standards and metadata are much less represented
- Linked data, a new topic in 2014, seems no longer so fashionable
- Web services and workflows also no longer so popular

The recognition given by the LR community to infrastructural issues, strategies and policies may be also due to the fact that we must often work in large groups, for many languages, we must build on each other work, connect various resources and tools, make available what already exists and use standardised formats. Infrastructures (on many dimensions) are really needed for our field to progress: to pay proper attention to these issues is another distinguishing feature of LREC.

LREC-related initiatives

Proceedings in Thomson Citation Index

Since 2010 the LREC Proceedings are included in CPCI (Thomson Reuters Conference Proceedings Citation Index): an important achievement, providing a better recognition to all LREC authors and useful in particular for young colleagues.

LRE Journal and LREC

After each LREC we ask to the authors of papers suggested by the 3 reviewers as appropriate for LRE if they want to submit an extended version to the *LRE journal*, coedited by Nancy Ide and myself. I am glad to report that also the journal has a large and increasing number of submissions, testifying the great interest for the field of LRs and Evaluation.

Citation of Language Resources

Also this year we encouraged citations of LRs in a special References section (introduced in 2016), providing recommendations on how to cite. I hope this becomes normal practice, to keep track of the relevance of LRs but also to provide due recognition to those working on LRs.

LRE Map and Share your Language Resources

As usual we encouraged descriptions of LRs in the LRE Map, an innovative instrument introduced at LREC2010 with the aim of monitoring the wealth of data and technologies developed and used in our field. And we ask, since 2014, to share the LRs with all the community.

In this LREC about 1000 LRs have been described in the Map. Just few hints at some data in the 2018 Map: WordNets, Wikipedia, Prague TreeBank are the most cited LRs; Corpora are by far the most frequent type (half of the LRs); and about 85% of the LRs are in some way available (not bad).

Replicability of research results

I believe that research is strongly affected also by infrastructural (meta-research) activities as those mentioned above. With these initiatives I hope we are able to promote in our field what is in use in more mature disciplines, i.e. ensure proper documentation and reproducibility of research results as a normal practice. ELRA and LREC are thus influential in strengthening the Language Resources and Evaluation scientific ecosystem and fostering sustainability.

Acknowledgments

As usual, it is my pleasure to express here my deepest gratitude to all those who made this LREC2018 possible and hopefully successful.

I first thank the Programme Committee members, not only for their dedication in the huge task of selecting the papers, the workshops and tutorials, but also for the constant involvement in the various aspects around LREC. I wish to thank each of them, the new ones:

Chris Cieri, Koiti Hasida, Hitoshi Isahara, Take Tokunaga, and obviously the “old” ones: Khalid Choukri, Thierry Declerck, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, Stelios Piperidis. A particular thanks goes to Jan Odijk, who has been so helpful in the preparation of the program.

I thank ELRA and the ELRA Board: LREC is a major service from ELRA to all the community!

A very special thanks goes to Sara Goggi and Hélène Mazo, chairs of the Editorial Committee, for the dedication and competence in managing the so many tasks they have in their hands, and the capability to tackle the many big and small problems of such a large conference (not an easy task). They are the two pillars of LREC, without whose commitment for many months LREC would not happen. So much of LREC organisation is on their shoulders, and this is visible to all participants.

I am especially grateful to Hitoshi Isahara and Kyoko Kanzaki, the Local Committee, for their great efforts in dealing with so many local matters and for their patience and true commitment, looking at so many details. We owe them a lot in organising a successful LREC.

My appreciation goes also to the distinguished members of the Advisory Board, chaired by Makoto Nagao, for their support and precious advices.

I am very grateful to the Local Liaison Committee, representing a number of Asian associations and organisations that have supported LREC, in particular with dissemination tasks.

I express my great gratitude to the Sponsorship Committee, and to all the Sponsors that have helped with financial support, believing in the importance of our conference.

I thank the Japanese Ministry of Education, Culture, Sports, Science and Technology for its precious support.

I am particularly grateful to the local authorities, the Governor of Miyazaki prefecture and the Mayor of Miyazaki city, very supportive of LREC since my first visit in 2016. We met them, and other local authorities, several times. We thank them also for their financial support.

In my many visits to Miyazaki I have been impressed by the great sense of hospitality of locals, and among them I wish to thank at least Manmatsu Hayashi and Rie Saita for their great help and kindness.

Also on behalf of the Program Committee, I praise our impressively large Scientific Committee. They did a wonderful job.

I thank the workshop and tutorial organisers, who complement LREC of so many interesting events.

I thank the organisers of the Industry Track, of the Special Speech session and of O-COCOSDA.

A big thanks goes to all the LREC authors, who provide the “substance” to LREC, and give us such a broad picture of the field.

This time I really want to thank also Softconf (Rich Gerber and Paolo Gai) and their constant efforts to make START a better tool for us. I greatly appreciated the new feature for plagiarism detection: I must say it was useful to detect some paper too similar to others …

I thank the European Commission for the interest in our conference, and hope that funding agencies will be impressed by the quality and quantity of initiatives in our sector that LREC

displays, and by the fact that the field attracts the best groups of R&D from all continents. The success of LREC for us means the success of the field of Language Resources and Evaluation.

I finally thank the two institutions that always dedicate a great effort to LREC: ELDA in Paris and ILC-CNR in Pisa. Without their commitment LREC would not be possible. The last, but not least, thanks are thus, in addition to Hélène Mazo and Sara Goggi, to all the others who – with different roles – have helped and will help during the conference: Roberto Bartolini, Damien Bihel, Irene De Felice, Riccardo Del Gratta, Pawel Kamocki, Valérie Mapelli, Monica Monachini, Vincenzo Parrinelli, Vladimir Popescu, Valeria Quochi, Caroline Rannaud, Alexandre Sicard.

And lastly, my final words are for all the LREC2018 participants, the true protagonist of LREC. Now LREC is in your hands. I hope that you discover new paths, that you perceive the vitality and strength of the field, that you have fruitful conversations (conferences are useful also for this) and most of all that you profit of so many contacts to organise new exciting work and projects in the field of Language Resources and Evaluation … which you will show at the next LREC.

This Japanese LREC in Miyazaki has a sort of Mediterranean flavour, typical of LREC. I am sure you will appreciate the Japanese great hospitality and kindness. And I hope that Miyazaki will enjoy the invasion of LRECers!

With all the Programme Committee, I welcome you at LREC2018 and wish you a very fruitful Conference.

Enjoy LREC2018 in Miyazaki!

Message to LREC 2018 participants by Henk van den Heuvel
ELRA President

Dear Colleagues and Friends,

Twenty years of LREC! It is my honour and pleasure to welcome you to this 11th edition of our successful Conference. Welcome to Miyazaki!

We are also very grateful with our guests representing the European Commission. Your presence here is deeply appreciated. Especially we welcome, Gael Kent, Director Data at the European Commission- DG CONNECT in Luxemburg. We are looking forward to your speech.

We are very honoured to have literally in our midst Prof. Makoto Nagao, Professor Emeritus of Kyoto University. We greatly admire your contributions to such various fields as Machine Translation, Natural Language Processing, Pattern Recognition, Image Processing and Library Science.

After 20 years LREC we have broken with the tradition to convene around the Mediterranean area, and look for another venue to meet and network. Honestly, we see this as an exceptional move motivated by our deep desire to intensify the ties with our Asian colleagues as we know them from the Asian Federation of Natural Language Processing (AFNLP), the Board of which is also closely involved in the organisation of this LREC through the Local Liaison Committee. We are very pleased to see so many of our Asian colleagues here in Miyazaki.

As President of ELRA it is my duty and pleasure to point out a couple of developments that are taking place in our Association. Already in 2012 one of my predecessors, Stelios Piperidis, referred to the dazzling speed of changes in which our community is finding itself. In his opening speech at LREC 2012 he also mentioned the upcoming of data-driven techniques and numerical and learning methods. In our days we see how algorithms and techniques developed in the area of Artificial Intelligence have come to play a paramount role in the area of Language and Speech technology. This technology puts special demands on the amount and preprocessing of Language Resources for training and testing purposes. Large amounts of data are collected from the web and continuously processed and used for application refinement. Now, in this rapidly changing field, ELRA has to find its way as one of the traditional sustainable key-players in language resources management and intermediary between stakeholders. It is evident that LRs remain essential also in our time, it is also evident that well-targeted annotated resources remain essential for supervised training approaches. Therefore, there remains an important role for ELRA as a sustainable LR broker offering relevant and high quality resources both to academia and commercial parties.

However, the changes that we see around us force us to continuously reflect on our *raison d'être* for our members in consideration of what their demands are for LRs and in terms of the services we offer around them. As a result of that, ELRA's Board has introduced important changes in its membership policy.

First of all, to stimulate continuity for our institutional members we have introduced a discount on membership fees upon membership continuation, starting with a discount of 15%

for the second year up to 30% for the third year and following. Second, we have equalized the fees for EU and non-EU members to the EU-members fee. Last but not least, as of January 2018 ELRA has introduced individual membership. An event such as LREC shows how vivid and productive the community around LRs is, and advocates for establishing a permanent link within this community, not only a biennial meeting point. For this reason ELRA has decided to open up its memberships for individuals, too, and to offer this membership with special services and benefits, of which the reduced registration fee is the one now most salient.

Employees of institutional ELRA members are also individual ELRA members if and when they want to use ELRA member services (including discount on LREC registration fees). They will not have to pay the individual membership fees as well since their organization covers for that.

In addition, one position in the ELRA Board will be reserved for a representative from the individual members, and this member is elected by the individual members only. This Board member has the same rights as the other ELRA Board members on all issues related to Board matters.

There will be a General Meeting for individual members at each LREC where they can convene with their representative and the Board to discuss ELRA matters concerning individual members. This meeting will be organized for the first time in this LREC 2018, namely this very afternoon at 18:00. The content of the meeting is an interesting mixture of relevant issues from the ELRA board, an inventory of wishes from individual members, and a self- introduction of Board applicants.

You are all invited to attend this first ELRA membership meeting, where we will tell more about the new membership policy, the special services for members and the election of the new Board member. We have sent out an invitation and an agenda for this.

Another observation that requires our persistent attention is that there are many players offering LRs both at the national and international level, and this landscape is becoming quite diffuse. This implies that we need to identify and re-identify times and again what our, ELRA's, position is compared to other LR brokers. It is ELRA's firm belief that this can best be done through cooperation. In this way we have set up a successful cooperation with for instance, LDC, by identifying the differences in membership policies, LR production and distribution strategies, and using each other's strengths in cooperation. In the same spirit ELRA has now set up a Collaboration Agreement with CLARIN ERIC. In this Collaboration Agreement we have clearly identified where our mutual and complementary strengths are and how we can bring these together to the benefit of both organisations. The objective of such agreements is not that one organization becomes part of the other but that both remain independent whilst joining forces. Indeed, here we see an important role for our association in facilitating synergies.

Another example of such a synergy has been established in our Special Interest Group for Under-resourced Languages, SIGUL. Created in April 2017, SIGUL is a joint Special Interest Group of the European Language Resources Association (ELRA) and of the International Speech Communication Association (ISCA). Through its establishment of the Special Interest Group on Under-resourced Languages, ELRA reasserts its active involvement in contributing to enhance the support for the languages with little or no technological support.

I would like to take the opportunity to thank all those who have worked so hard to make this conference a fantastic event: the LREC Programme Committee, chaired by Nicoletta Calzolari, the Scientific Committee, the Conference Editorial Committee headed by our LREC cornerstones Sara Goggi and Hélène Mazo, the International Advisory Committee chaired by Prof. Makoto Nagao, the group in Pisa, Khalid Choukri and the ELDA staff in Paris, the Local Committee headed by Prof. Hitoshi Isahara and Dr Kyoko Kanzaki. Each one of them in his/her own role has been taking care of the incredible amount of issues that emerge when undertaking the organisation of such a complex and demanding conference as LREC. Our particular thanks go to our sponsors and supporters.

We thank workshop and tutorial organizers, project consortia participating in the HLT Village; you have all exceeded yourselves once more to make this LREC such a great event.

Dear LREC Participants, in the end this is your conference. With your active participation in the oral sessions, your lively discussions with the presenters at the poster sessions, your visits to the HLT Village and Exhibition Boots and participation in the Industry Track I am confident you will make LREC 2018 yet another success.

Introductory message of Khalid Choukri,
ELRA Secretary General
ELDA Chief Executive Officer

ELRA and ELDA are very pleased to welcome you in Miyazaki to this 11th LREC to celebrate the 20th anniversary of LREC with all of you this week.

On behalf of the ELRA/ELDA team I would like to share with you some news on the activities we conducted since the last LREC in Portorož (Slovenia).

The Declaration of Granada

But first let me to share some feelings about this special LREC with you, as we are celebrating the 20th anniversary of this major forum established in 1998 in Granada (Spain), organized for its 11th edition, here in Japan.

Soon after the establishment of ELRA in 1995, its Board realised that, at that time, the language resources and the evaluation of language technologies were given very little attention at the main events. Today, we are glad that such message is spread widely and is endorsed by the major conferences in which special sessions are expressly devoted to Language Resources and Evaluation!!

Remember the first LREC, remember Granada, not only the Alhambra! With over 400 participants instead of the expected 100 attendees, we realized the importance of such forum for the community. This was confirmed over the years by a steady attendance of 1200 participants to the last editions of LREC.

I would like to take this opportunity to go back to the spirit of Granada, paying a tribute to those who were behind it, Professors Antonio Zampolli and Angel Martin Municio. I would like to bring up one of the major outcomes of that first event: “the declaration of Granada”. Its recommendations are still relevant and topical, more urgent than ever to implement.

The declaration of Granada[1] comprised 10 articles. I am highlighting and commenting here some of the crucial ones that we can continue to endorse today:

- "**At this moment, language resources are one indispensable key to unlock the potential of the global information Society**"

We are still facing this issue 20 years later and if we agree that the Information Society has made tremendous progress with the emergence of social networks which have strengthened links within and between communities, social or commercial activities cross borders are still hindered by language barriers. In 2015, surveys mentioned that 24 languages are used in LinkedIn user interfaces, 48 on Twitter, 91 on Google Translate (as pairs for its translation of content and now about 103), over 150 on Facebook, just over 300 in Wikipedia. These

[1] Granada Declaration: http://www.elra.info/media/filer_public/2013/09/06/v3n3.pdf)

numbers may seem impressive, but remember that this is **out of 7097 living languages or 3,909 with writing systems.** And most of these languages are used in interfaces with automatic processing of content used in Search and/or MT only. Language Resources are essential assets. Back in Granada, we stated that "They constitute an essential infrastructure". Such infrastructure is missing for a huge number of languages. The LRE Map service provided by ELRA, inventorying the LRs reported in major conferences, continue to expose the existing gaps.

- **"All sectors of society, and all languages, have an interest in seeing these resources developed, for a variety of purposes, economic, social, industrial and cultural."**

ELRA continues to promote the concept of Basic Language Resource Kit, a Kit that would help process every language for (at least) the basic NLP functions. We stressed the importance of this approach to policy makers, emphasised the need to support small communities, and mentioned the lack of interest from private sector for non-lucrative/non-strategic languages. We also insisted that such "*core language resources should remain in the public domain*" to ensure a wide use by both research and development stakeholders. Reviewing the current situation at major data centers and repositories, we can barely count more than 100 different languages, often with scarce resources (many speech resources for the major languages, very few treebanks, very few aligned corpora, mostly aligned with English, etc.)

- **"For each language, there is a need for strategy to co-ordinate existing resources and create new ones."**

ELRA, along with LDC, their partner in the USA, did their best to offer distribution/sharing channels for Language Resources produced within publicly funded projects and some offered by private bodies. However the identified resources represent less than 15% of what exists. Coordination of the distribution but also documentation and production, have proved to be challenging. We still feel it is crucial to coordinate building roadmaps for every language and enhance the involvement of local public and private bodies. It is also essential to continue international cooperation to disseminate the know-how acquired for a given language. We are glad that a conference like LREC contributes to sharing such expertise and value the implication of governmental (regional and national) and international bodies.

We introduced the International Standard Language Resource Number (now part of the activities of the International Standardisation Organisation, ISO TC37/SC4) to assign a unique identifier with each identified Language Resource to improve the way we reference it (this is also part of the LREC submission process that distinguishes Bibliographical data from LR data). The idea is not only to provide an ID, unique and persistent, wherever the LR is stored, even for those LRs on local servers outside the Internet. This is an uphill struggle but we are convinced that it is an important step in our work to improve the identification of existing resources, the assessment of LR impact factor as well as the citation mechanism.

- **"When resources have been created, there is a continuing requirement for support and maintenance."**

This is a key part of our mission and we tried to convince data producers and funders to account for the necessary maintenance of and support for Language Resources. We introduced the validation process and the "bug" reporting mechanism, as part of ELRA procedures, to encourage sharing experiences on the use of LRs and their enhancement over time. We still face funding scenarios that provide subsidies for data production and not for

other issues like IPR clearance, documentation, sharing, maintaining, etc. In Granada, we anticipated that resources would undergo some repurposing with the new uses that emerge and we insisted on the need to envisage a wide range of applications on the basis of the same resources. The community seems to be sensitive to this, but some legislators are debating the adoption of more legal constraints. We need to join forces to convince funders and decision makers about the importance of more openness and long term policies. The introduction of the Data Management Plan (DMP) by ELRA, and soon the DMP Wizard, will help each data manager to adopt up-to-date standards and best practices for data management.

- **"Understanding of the role, usefulness and optimum means of preparation for language resources is a research theme in itself."**

Over the last decades, and especially within the last 3-4 years, we have seen an impressive breakthrough in the HLT field. The new data-intensive machine learning and the computing capabilities, are proving the crucial usefulness of LRs. Making LRs widely available is the core mission of a few organisations. ELRA is very happy to be among these organizations and is making the necessary investments to acquire more expertise to cost-effectively produce and share LRs. The setup of an internal legal team is helping to shed light on a large number of legal issues that impede the use/re-use of LRs. Working on standards is also an important aspect to help facilitate the interoperability and sharing of data. One of our mottos was that "Common evaluation requires common standards". We still feel that common tasks in the "challenges" and evaluation campaigns are essential instruments to assess progress, share knowledge, and improve cooperation. It is a pity that many "Evaluation campaigns" are happening with very little coordination which makes them hard to find for new comers.

> ***Granada was 20 years ago and we see that some visionary recommendations are still needed today. A multilateral, concrete, and lasting cooperation remains on top of our action.***

ELRA activities since 2016

Now allow me to get back to ELRA activities carried out over the last couple of years.

We continue our actions on data sharing, through the identification, negotiation, and distribution agreements with right holders when necessary. We continue to produce resources for projects as well as for partners. Our policy remains consistent: whenever the data is offered to the community, after the shortest possible embargo period, the costs for partners are set to production costs. This position remains fundamental to our policy. We continue to invest in research and development of tools to improve and automate our production procedures. Most of our tools are shared as open source packages.

We continue also to work on our quality control methodologies so as to supply validated resources with validation procedures that guarantee the adequacy of the produced datasets with respect to the initial specifications and the state of the art.

To ensure an efficient distribution of Language Resources, ELRA has migrated its catalogue of resources to a new platform, based on e-Commerce features, redesigned with a new interface and an improved navigation. This foreshadows further developments that will incorporate e-licensing, e-payment and e-delivery of resources.

ELRA continues to support the set-up of LR repositories for data deposit by third parties. Based on its involvement in the jointly-developed META-SHARE platform, we continue the

promotion of such efforts to ensure that the major data holders adhere to some common practices. A new repository was set up as part of an EU service contract to store data for MT provided by the public sector. Such initiative is now spreading across Europe, and a coordination action is establishing local repositories (known as Local Relay Stations). If we succeed to set up such stations for each country in order to collect all language datasets produced by translations services and secure these for MT training and tuning, one can anticipate good progress for these languages and domains. The repositories can accommodate any Language Resource modality.

If the establishment of such a local repository is of interest to your organization and your network, let us discuss how to work on it together.

As part of this process, we continue to work on all issues related to sustainability and preservation of data for the generations to come.

An updated ELRA Data Management Plan is made available and reviews all necessary aspects for an optimal management of resources with an easy-to-use checklist. We are working to automate the customisation of such DMP for each project. Our members will benefit from this automatic DMP Wizard, accompanied with the support of our experts, free of charge. We hope that such approach will improve sustainability and preservation of Language Resources but also make them easy to identify.

ELRA continues to be involved in the new trends in HLTs. It continues to support the new trends in MT. Many of our projects (some of which are funded under a European Program known as Connecting Europe Facility (CEF) focus on data production, including via requests for donations from translation services, but also crawling of adequate data to which we have access and re-use rights. Many resources come from organizations that belong to the Public Sector. A directive (called Public Sector Information directive, PSI) entered into application in the European Union, similar rules exist in many other countries, stating that publicly produced data should be made publicly available. This makes some of the resources needed by our community (e.g. textual corpora) available for new domains and new genres. Some geographical areas offer a multilingual environment (EU, India? South Africa, etc.), and hence more resources should be available for MT development.

Unfortunately there are still important legal restrictions on the re-use of data, even for research purposes. We continue to vilify the current legal framework, in particular in Europe, e.g. the European Union is working on a new directive on copyright in the Digital Single Market. The initial proposal for this act contained a mandatory exception for text and data mining carried out by research institutions. However, the current debates within the European decision makers seem to suggest that the exception will fall short of meeting the objective of the exception. The beneficiaries of the new exception may be limited to public research institutions, and – more importantly – 'lawful access' will be a prerequisite for data mining, which will probably result in wider implementation of digital protection measures by right holders. It is unlikely to get the exception for research that we claim since years now as a fair use doctrine for research purposes (that remains the privilege of a few countries).

The current legal framework has a strong impact on the capacity of the community to produce IPR cleared and sharable data. ELRA heavily invested in legal training and has been, for many years now, one of the few organizations that works both with in-house legal experts and a network of external practitioners/lawyers.

Another critical novelty in Europe is the new legal framework governing the processing of personal data. It goes beyond the users expectations, for more ethical behaviour on the management of their data. This may hinder the new developments of resources and technologies (e.g. Crowdsourcing activities). The new regulation (General Data Protection Regulation (GDPR)) will impose more restrictions on managing several aspects of data e.g. data protection by design and by default, privacy impact assessment, pseudonymisation and anonymization, before the data can be shared (this will of course impact also production, repackaging, repurposing of data).

To share information on these matters, a dedicated workshop on legal and ethical issues continues to be organized within LREC and will be held this week as well.

Of course, ELRA does not focus on EU issues and EU languages only (we distribute resources for more than 70 languages). In 2017, ELRA entered into an important agreement with the International Speech Communication Association (ISCA[2]) to join forces in the promotion of activities related to the Less-Resourced Languages (LRL). ELRA and ISCA agreed to merge their groups and set up a join Special Interest Group for Under-resourced Languages (SIGUL[3]). Co-chaired by a representative of ELRA and a representative of ISCA, SIGUL will continue to organize events for the LRL and encourage cooperation actions to support these languages.

As you may know, United Nation General Assembly proclaimed 2019 as the International Year of indigenous Languages. UNESCO is leading the corresponding events. ELRA proposed to organize an important international event related to HLT and Indigenous languages. We hope to draw attention to the importance of HLT and LRs for the preservation and development of local cultures and put under spotlights the role our community could play for these languages.

We continue to develop the LRE Map application. LRE Map was established to reference all LRs described by authors when submitting papers to conferences and journals. Started with LREC, it is used by other events but not as widely as we hope. In addition to identifying over 7000 instances of LRs, it helps identify existing gaps for languages lacking such modalities and ensure a minimal cooperation when planning new productions. If you are involved in the organisation of a conference, let us see how we can work together.

ELRA is also taking part in several standardisation activities. It is naturally involved in ISO/TC37/SC 4 on Language Resource management but also on ISO/IEC JTC1/SC35 about user interfaces and accessibility. ELRA brings its knowledge of the HLT field to ensure that all ICT services and products are accessible to all, in particular to users with specific needs. Some of the HLT applications are offering valuable services when converting speech into text, text into speech, sub-titling/captioning audio-visual streams, providing audio descriptions, translations (e.g. subtitles), easy-reading features (both in mono- and multilingual contexts). Such services are valuable to everyone and not only hearing or visually impaired users. Translation from text or speech to Sign languages is a big challenge that many partners are working on and ELRA will support them.

As a conclusion to my message, I would like to reiterate my statement uttered at almost all LRECs since 1998. Please remember that we can help you share your data for all types of use. We can work out a contractual framework that suits your expectations, including adopting very permissive licences and a free-of-charge policy. We can guarantee the availability as well

[2] https://www.isca-speech.org/iscaweb/index.php/about-isca
[3] http://www.elra.info/en/sig/sigul/

as the sustainability of your resources. During the conference, an ELRA booth is available where we will be happy to interact with you on such topics.

About 10 years ago, we identified about 20 resources, some were on the web, others well known to the community. We keep monitoring their availability. Believe it or not, about 30% disappeared and these are not necessarily the ones that were obsolete and useless. Some right holders also disappeared and the "orphan" resources with them.

Acknowledgments

Finally, I would like to express my deep thanks to our partners and supporters, who throughout the years make LREC so successful. I would like to thank our Sponsors: Google, Amazon, Arcadia, EML (European Media Laboratory GmbH), GSK Language Resources, Riken-AIP, Yahoo Research Japan, and the publisher Hituzi Syobo our media sponsor: MultiLingual Computing, Inc.

I also would like to thank the HLT Village participants, we hope that such gathering offers the projects an opportunity to foster their dissemination and hopefully to discuss exploitation plans with the participants.

I would like to thank the Local Advisory Committee. Its composition of the most distinguished personalities of Japan denotes the importance of language and language technologies for the country.

I would like to thank the LREC Local Committee, chaired by Prof. Hitoshi Isahara and the LREC Local Organizing Committee, for providing support to the organization of this LREC Edition in Japan.

Finally I would like to warmly thank the joint team of the two institutions that devoted so much effort over months and often behind curtains to make this one week memorable: ILC-CNR in Pisa and my own team, ELDA, in Paris. These are the two LREC coordinators and pillars: Sara Goggi and Hélène Mazo, and the team: Roberto Bartolini, Damien Bihel, Irene De Felice, Valérie Mapelli, Monica Monachini, Vincenzo Parrinelli, Vladimir Popescu, Caroline Rannaud, and Alexandre Sicard.

Now LREC 2018 is yours: we hope that each of you will achieve valuable results and accomplishments. We, ELRA and ILC-CNR staff, are at your disposal to help you get the best out of it.

Once again, welcome to Miyazaki and Japan, welcome to LREC 2018

Message to LREC 2018 participants by of Hitoshi Isahara and Kyoko Kanzaki, Chairs of the Local Committee

Welcome to Japan! Welcome to Miyazaki!!

On behalf of the whole local team, we would like to extend our warmest welcome to all of you participating in this 11th edition of LREC in Miyazaki, Japan!

You may know personally some Japanese researchers in the fields which are involved in LREC, and also may know that Japan is a country with four distinctive seasons, blessed with beautiful nature, world heritage sites, rich culture and respected traditions. However, we are sure that most of you are not familiar with Miyazaki.

Miyazaki has beautiful coastline facing the Pacific Ocean, and has many shrines involved with the myth of the birth of Japan, such as Miyazaki Shrine sacred to Emperor Jinmu, supposedly the first Emperor of Japan. Most stories in the mythology associated with the creation of Japan and the origin of the imperial line took place in Miyazaki Prefecture on the island of Kyushu.

You can visit scenic places and historical places in Miyazaki in this occasion, or you will be able to visit here again with your friends and family.

During your stay here, we would like you to experience Omatsuri, festival in Japan. You can enjoy Shrine Maiden Dance, Kagura performance (a sacred music and dancing performance dedicated to the Shinto gods) and local cuisine during reception. Lunch will be served in stall style.

On this occasion, we would like to thank all the supporters in Japan from the bottom of **our** hearts.

We first appreciate the patronage of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) towards LREC2018.

We thank Miyazaki Prefecture and Miyazaki city for their continuous support from the beginning of LREC's venue selection process. We offer the Governor, Mr. Shunji Kono, and the Mayor, Mr. Tadashi Tojiki our heartfelt thanks for their great kindness and efforts. Thanks to their generous understandings, preparation of LREC went smoothly.

We also thank Miyazaki Convention and Visitors Bureau for its financial and human support. We are grateful to Mr. Mitsunori Mera, Mr. Toshiaki Tomitaka and Mr. Manmatsu Hayashi.

We are sure that all participants satisfy high quality service of the conference venue, Phoenix Seagaia Resort. We appreciate very hard work by Mr. Hirofumi Matsunaga, Mr. Manmatsu Hayashi, Ms. Rie Saita and Mr. Satoru Kamibayashi to meet the requirements which LREC indicated.

One of the highlights of LREC2018 is the Reception at Miyazaki Shrine which is sacred to the first Emperor of Japan. We are grateful to its Chief Priest, Mr. Hidekiyo Sugita, for accepting us.

Participants can enjoy local liquor at the Reception and Gala Dinner. Let's thank to Miyazaki Sake Brewers Association.

We would like to thank Ms. Rika Kubota for organizing interpreter volunteers during LREC.

We would like to thank the Japan National Tourism Organization (JNTO) which supported us during LREC's venue selection process, including site visit to choose the best place for LREC in Japan.

Lastly, we would like to thank Mr. Manmatsu Hayashi for his great effort to make LREC success. The word "impossible" couldn't be found in his dictionary.

We are ready to welcoming you with *omotenashi*, our traditional spirit of hospitality.

Enjoy LREC2018 in Miyazaki!

LREC 2018 Committees

- **Conference Programme Committee**

Nicoletta Calzolari	**ILC/CNR, Pisa, Italy (Conference chair)**
Khalid Choukri	ELRA, Paris, France
Christopher Cieri	LDC, Philadelphia, USA
Thierry Declerck	DFKI GmbH, Saarbrücken, Germany
Koiti Hasida	The University of Tokyo, Tokyo, Japan
Hitoshi Isahara	Toyohashi University of Technology, Toyohashi, Japan
Bente Maegaard	CST, University of Copenhagen, Denmark
Joseph Mariani	LIMSI-CNRS, Orsay, France
Jan Odijk	UIL-OTS, Utrecht, The Netherlands
Asuncion Moreno	Universitat Politècnica de Catalunya, Barcelona, Spain
Stelios Piperidis	ILSP, Athens, Greece
Takenobu Tokunaga	Tokyo Institute of Technology, Tokyo, Japan

- **Advisory Board**

Shyam S. Agrawal	KIIT, Gurgaon (India)
Hiroya Fujisaki	University of Tokyo (Japan)
Eva Hajičová	UFAL, Charles University, Prague (Czech Republic)
Yuming Li	Beijing Language and Culture University (PRC)
Mark Liberman	Linguistic Data Consortium, Philadelphia (USA)
Makoto Nagao (Chair)	**Professor Emeritus University of Kyoto (Japan)**
Jun'ichi Tsujii	Artificial Intelligence Research Center, Tokyo (Japan)

- **Local Liaison Committee**

Key-Sun Choi	KAIST (Korea)
Chu-Ren Huang	The Hong Kong Polytechnic University (Hong Kong SAR - PRC)
Toru Ishida	Department of Social Informatics, Kyoto University (Japan)
Haizhou Li	National University of Singapore (Singapore)
Satoshi Nakamura	Nara Institute of Science and Technology (Japan)
Byong-Rae Ryu	Chungnam National University (Korea)
Virach Sornlertlamvanich	Sirindhorn International Institute of Technology, Thammasat University (Thailand)
Le Sun	Chinese Academy of Sciences, Beijing (PRC)
Kam-Fai Wong	The Chinese University of Hong Kong (Hong Kong SAR - PRC)
Chengqing Zong	Chinese Academy of Sciences, Beijing (PRC)

Scientific Committee

The Programme Committee is very grateful to Scientific Committee members who reviewed the submissions and contributed to designing the conference programme. The list of the members of Scientific Committee is published on the LREC 2018 web site.

- **Local Committee**

Hitoshi Isahara (Chair)	Toyohashi University of Technology, Toyohashi, Japan
Kyoko Kanzaki	Toyohashi University of Technology, Toyohashi, Japan

- **Conference Editorial Committee**

Sara Goggi	ILC/CNR, Pisa, Italy
Hélène Mazo	ELDA/ELRA, Paris, France

- **Organising Committee**

Roberto Bartolini	ILC/CNR, Pisa, Italy
Damien Bihel	ELDA/ELRA, Paris, France
Irene De Felice	University of Pisa, Italy
Riccardo Del Gratta	ILC/CNR, Pisa, Italy
Sara Goggi	**ILC/CNR, Pisa, Italy (Co-chair)**
Valérie Mapelli	ELDA/ELRA, Paris, France
Hélène Mazo	**ELDA/ELRA, Paris, France (Co-chair)**
Monica Monachini	ILC/CNR, Pisa, Italy
Vincenzo Parrinelli	ILC/CNR, Pisa, Italy
Vladimir Popescu	ELDA/ELRA, Paris, France
Valeria Quochi	ILC/CNR, Pisa, Italy
Caroline Rannaud	ELDA/ELRA, Paris, France
Alexandre Sicard	ELDA/ELRA, Paris, France

- **Sponsorship Committee**

Nicoletta Calzolari	ILC/CNR, Pisa, Italy
Khalid Choukri	ELRA, Paris, France
Tatjana Gornostaja	Tilde, Riga, Latvia
Hitoshi Isahara	Toyohashi University of Technology, Toyohashi, Japan
Kyoko Kanzaki	Toyohashi University of Technology, Toyohashi, Japan
Jimmy Kunzmann	EML GmbH, Heidelberg, Germany
Joseph Mariani	LIMSI-CNRS & IMMI, Orsay, France
Satoshi Sekine	New York University, New York City, USA

Acknowledgements

The European Language Resources Association, ELRA, and the LREC Committees acknowledge with gratitude the support and sponsoring of the following institutions.

Sponsors and supporters

- Google *(Platinum)*
- AIP, Center for Advanced Intelligent Project *(Bronze*)
- Amazon AWS *(Bronze*)
- Arcadia Computing Innovation *(Bronze*)
- EML European Media Laboratory *(Bronze*)
- Yahoo Research Japan *(Bronze)*
- GSK Language Resources Association *(Supporter)*
- Hituzi Syobo *(Publisher*)
- Multilingual (*Media Sponsor*)

Supporting Institutions

- Evaluations and Language resources Distribution Agency (ELDA), Paris (France)
- Istituto di Linguistica Computazionale (ILC) of the Italian National Research Council (CNR), Pisa (Italy)
- Toyohashi University of Technology, Japan
- Miyazaki Convention and Visitors Bureau

Augmenting Librispeech with French Translations: A Multimodal Corpus for Direct Speech Translation Evaluation

Ali Can Kocabiyikoglu*, Laurent Besacier*, Olivier Kraif†
*LIG, UGA, G-INP, CNRS, INRIA, Grenoble, France
†LIDILEM, UGA, Grenoble, France
Univ. Grenoble Alpes, F-38400, Saint-Martin d'Heres
alicankocabiyikoglu@gmail.com, laurent.besacier@univ-grenoble-alpes.fr, olivier.kraif@univ-grenoble-alpes.fr

Abstract

Recent works in spoken language translation (SLT) have attempted to build end-to-end speech-to-text translation without using source language transcription during learning or decoding. However, while large quantities of parallel texts (such as Europarl, OpenSubtitles) are available for training machine translation systems, there are no large (>100h) and open source parallel corpora that include speech in a source language aligned to text in a target language. This paper tries to fill this gap by augmenting an existing (monolingual) corpus: LibriSpeech. This corpus, used for automatic speech recognition, is derived from read audiobooks from the LibriVox project, and has been carefully segmented and aligned. After gathering French e-books corresponding to the English audio-books from LibriSpeech, we align speech segments at the sentence level with their respective translations and obtain 236h of usable parallel data. This paper presents the details of the processing as well as a manual evaluation conducted on a small subset of the corpus. This evaluation shows that the automatic alignments scores are reasonably correlated with the human judgments of the bilingual alignment quality. We believe that this corpus (which is made available online) is useful for replicable experiments in direct speech translation or more general spoken language translation experiments.

Keywords: direct speech translation, bilingual alignment, librispeech corpus

1. Introduction

Attention-based encoder-decoder approaches have been very successful in Machine Translation (Bahdanau et al., 2014), and have shown promising results in End-to-End Speech Translation (Bérard et al., 2016; Weiss et al., 2017) (translation from raw speech, without any intermediate transcription). End-to-End speech translation is also attractive for language documentation, which often uses corpora made of audio recordings aligned with their translation in another language (no transcript in the source language) (Blachon et al., 2016; Adda et al., 2016; Anastasopoulos and Chiang, 2017). However, while large quantities of parallel texts (such as Europarl, OpenSubtitles) are available for training (text) machine translation systems, there are no large (>100h) and open source parallel corpora that include speech in a source language aligned to text in a target language. For End-to-End speech translation, only a few parallel corpora are publicly available. For example, *Fisher* and *Callhome* Spanish-English corpora provide 38 hours of speech transcriptions of telephonic conversations aligned with their translations (Post et al., 2013). However, these corpora are only medium size and contain low-bandwidth recordings. Microsoft Speech Language Translation (MSLT) corpus also provides speech aligned to translated text. Speech is recorded through $Skype$ for English, German and French (Federmann and Lewis, 2016). But this corpus is again rather small (less than 8h per language).

Paper contributions. Our objective is to provide a large corpus for direct speech translation evaluation which is an order of magnitude bigger than existing corpora described in the introduction. For this, we propose to enrich an existing (monolingual) corpus based on read audiobooks called *LibriSpeech*. The approach is straightforward: we align e-books in a foreign language (French) with the English utterances of *LibriSpeech*. This results in 236h of English speech automatically aligned to French translations at the utterance level[1].

Outline.

This paper is organized as following: after presenting our starting point (*Librispeech*) in section 2., we describe how we aligned foreign translations to the speech corpus in section 3.. Section 4. describes our evaluation of a subset of the corpus (quality of the automatically obtained alignments). Finally, section 5. concludes this work and gives some perspectives.

2. Our Starting Point: Librispeech Corpus

Our starting point is *LibriSpeech* corpus used for Automatic Speech Recognition (ASR). It is a large scale corpus which contains approximatively 1000 hours of speech aligned with their transcriptions (Panayotov et al., 2015). The read audio book recordings derive from a project based on collaborative effort: LibriVox. The speech recordings are based on public domain books available on *Gutenberg Project*[2] and are distributed with *LibriSpeech* as well as the original recordings.

We start from this corpus[3] because it has been widely used

[1] Our dataset is available at `https://persyval-platform.univ-grenoble-alpes.fr/DS91/detaildataset`

[2] `https://www.gutenberg.org/`

[3] Another dataset could have been used: TED Talks - see `https://www.ted.com` - but we considered it was be better to start with a read speech corpus for evaluating End-2-End speech translation.

in ASR and because we believe it is possible to find the text translations for a large subset of the read audiobooks.

subset	hours	per-spk minutes	female spkrs	male spkrs	total spkrs
dev-clean	5.4	8	20	20	40
test-clean	5.4	8	20	20	40
dev-other	5.3	10	16	17	33
test-other	5.1	10	17	16	33
train-clean-100	100.6	25	125	126	251
train-clean-360	363.6	25	439	482	921
train-other-500	496.7	30	564	602	1166

Table 1: Details on *LibriSpeech* corpus

Table 1. gives details on *Librispeech* as well as data split. Recordings are segmented and put into different subsets of the corpus according to their quality (better quality speech segments are put in the clean part). Note that in order to obtain a balanced corpus with a large number of speakers, each speaker only read a small portion of a book (8-10 minutes for dev and test, 25-30 minutes for train). Moreover, in training data, speech segments are obtained by splitting long signals according to ($> 0.3s$) silences in order to obtain segments that are maximum 35s long.

3. Aligning Foreign Translations to Librispeech

3.1. Overview

The main steps of our process are the following:

- Collect e-books in foreign language corresponding to English books read in *Librispeech* (section 3.2.),
- Extract chapters from these foreign books, corresponding to read chapters in *Librispeech* (section 3.3.),
- Perform bilingual text alignement from comparable chapters (section 3.4.),
- Realign speech signal with text translations obtained (section 3.5.).

These different steps are described in the next subsections.

3.2. Collecting Foreign Novels

LibriSpeech corpus is composed of 5831 chapters (from 1568 books) aligned with their transcriptions. We used the given metadata to search e-books in foreign language (French) corresponding to English books read in *Librispeech*. Firstly, we used *DBPedia* (Auer et al., 2007) in order to (automatically) obtain title translations. Secondly, we used a public domain index of French e-books[4] to find Web links matching titles we found. Then, we finished this process for the entire *LibriSpeech* corpus by manually searching for French novels in different public domain resources. Overall, we collected 1818 chapters (from 315 books) in French to be aligned with *Librispeech*. Some of the public domain resources that we used are: Gutenberg Project[5], Wikisource[6], Gallica[7], Google Books[8], BEQ[9], UQAC[10].

Audiobooks available in LibriSpeech are of different literary genres: most of them are novels, however there are also poems, fables, treaties, plays, religious texts, *etc*. Belonging to the public domain, most of the texts are old and not available publicly in foreign language. Therefore, the novels that were collected in foreign language are mostly novels from world's classics. As few of them are ancient texts, some translations are in old French.

3.3. Chapters Extraction

LibriSpeech transcriptions are provided for each chapter. As the readers only read a short period of time[11], transcriptions may correspond to incomplete chapters. For the same reason, books are not read entirely. Therefore, in order to obtain an alignment at the sentence level, a first step was to decompose English and French language books into chapters. This step was achieved by a semi-automatic process. After converting books to text format (both English and French), regular expressions were used to identify chapter transitions. Then, each French chapter was extracted and aligned to its counterpart in English. After manual verification of all chapters, we obtained 1423 usable chapters (from 247 books).

3.4. Bilingual Text Alignement

The 1423 parallel chapters establish the comparable corpus from which we extracted bilingual sentences. This was done using an off-the-shelf bilingual sentence aligner called *hunAlign* (Varga et al., 2007). *HunAlign* takes as input a comparable (not sentence-aligned) corpus and outputs a sequence of bilingual sentence pairs. It combines (Gale-Church) sentence-length information as well as dictionary-based alignment methods.

Initial dictionary available for alignment was the default French-English (40k entries) lexicon created for $LFAligner$[12] (wrapper for *hunAlign* created by Andras Farkas). We enriched this dictionary by adding entries from other open source bilingual dictionaries. Different dictionaries (woaifayu, apertium, freedict, quick) from a language learning resource were gathered in various formats and adapted to *hunAlign* dictionary format[13]. We finally obtained and used a dictionary of 128,000 unique entries.

In order to improve the quality of sentence level alignments, data had to be pre-processed. For English and

[4] https://www.noslivres.net/

[5] http://www.gutenberg.org/

[6] http://www.wikisource.org/

[7] http://gallica.bnf.fr/

[8] http://books.google.com

[9] http://beq.ebooksgratuits.com

[10] http://www.uqac.ca/

[11] One goal of *Librispeech* was to have as many speakers as possible

[12] https://sourceforge.net/projects/aligner/

[13] https://polyglotte.tuxfamily.org

French, our extracted chapters were cleaned with regular expressions. Then, we used Python NLTK (Bird, 2006) sentence split to detect sentence boundaries in the corpora. Furthermore, the bitexts were stemmed (removing suffixes to reduce data sparsity). Finally, parallel sentences found were brought back to their initial form with reverse stemming. This last step was done using Google's $diff - patch - match$ library (Fraser, 2012).

English Sentence	French Sentence
Oh, I beg your pardon!	≪Oh! je vous demande bien pardon!
A lane was forthwith opened through the crowd of spectators.	Un chemin fut alors ouvert parmi la foule des spectateurs.
No, "said Catherine," he is not here; I cannot see him anywhere.	- Non, dit Catherine, il n'est pas ici. Jamais je ne parviens à le rencontrer.

Table 2: Examples of parallel sentences obtained from comparable corpora made up of aligned book chapters

Table 2. shows examples of 3 bilingual sentences obtained from 3 different chapters.

3.5. Realigning Speech Signal with Text Translations

In order to associate parallel sentences to speech signal transcriptions, realignment of speech segments of *LibriSpeech* was necessary. This realignment is a two step process: first, we forced aligned *Librispeech* English transcripts to match English sentences obtained in the previous stage ; secondly, we resegmented the speech signal according to new sentence splits.
For the first step, we used $mweralign$, a tool for realigning texts in a same language but with a different sentence tokenization (Matusov et al., 2005). We applied $mweralign$ to realign our speech transcriptions in English to the English sentences of our bilingual corpus obtained in section 3.4.. The outcome of this first step is a new sentence segmentation for our English transcriptions that are now correctly aligned to our French translations.
The second step was to resegment the speech signals to match them to the new sentence segmentation. We did that by:

- creating a big wav file by concatenating speech segments for each chapter,
- re-aligning the large speech wav signal to the transcripts using $gentle$[14] toolkit, an off-the-shelf English forced-aligner based on $Kaldi$ ASR toolkit (Povey et al., 2011),
- re-segmenting speech according to the desired sentence split.

Table 3. presents and overview of final data (speech with aligned translations) obtained after this final step. For each sentence pair, we also added En-Fr machine translation output of our English transcripts (*Google Translate*). So we have 2 French translations in the end (a correct one from automatic alignement ; a noisy one from MT).

Chapters	Books	Duration (h)	Total Segments
1408	247	~236h	131395

Table 3: Statistics of the final multimodal and bilingual corpus obtained (English speech aligned to French text)

4. Human Evaluation of a Corpus Subset

4.1. Protocol

Now that we have obtained a multimodal alignment between (English) speech signals and (French) translations, we want to evaluate its quality. At this point, the only score available is the confidence score given by $hunalign$ indicating confidence for aligned sentences. One goal of this human evaluation, that can only be made on a corpus subset, is to see if $hunalign$ score has a good correlation with human judgements.
50 sentences from 4 different chapters have been chosen for evaluation. These chapters were chosen according to their average alignment scores (from $hunalign$). We chose two chapters that were near the mean of overall alignment scores (hypothesized medium quality alignments), one chapter which was above the mean score (hypothesized good quality alignment) and a final chapter below mean score (hypothesized bad quality alignment). These sentences were evaluated by three annotators. We established a scale from 1 to 3 to judge matching quality between English speech and English transcriptions. This 3-step scale is precise enough because few errors were found in speech alignments. We established a scale from 1 to 5 to judge quality between bilingual text alignments. Overall, 200 sentences were evaluated (on both scales) by 3 annotators.
We give, as example below, sentences for each mark (1-5) for human evaluation of bilingual alignments. Two different dimensions are evaluated at the same time: the accuracy of alignment (an alignment can be wrong, partial or correct) and the fact that translational equivalence is compositional and may be isolated from the current context.

- **1.** Wrong alignment

  English: COMMIT TO ME I SHALL LET PASS NO ADVANTAGE

  French: Je sais, par exemple, que maintenant il souffre de la faim dans un vaste désert, où l'on ne saurait trouver de nourriture.

- **2.** Partial alignment with slightly compositional translational equivalence

  English: THAN IN SET TERMS AND IN COURTLY LANGUAGE

[14] https://github.com/lowerquality/gentle

Chapter	Average confidence score ($hunalign$)	Average speech alignment score (max 3)	Average textual alignment score (max 5)
Ivanhoe Chapter XXIII	1.34	2.82	4.64
Alice's Adventures in Wonderland Chapter V	1.14	2.98	4.28
A Tale of Two Cities Book III, Chapter III	0.96	2.86	3.86
Adventures of Huckleberry Finn Chapter VIII	0.66	2.9	2.58
Average	**1.02**	**2.89**	**3.84**

Table 4: Results of human evaluation by 3 annotators.
Kappa's Cohen (weighted) for inter annotator agreement for textual alignment is 0.76

French: Mais il paraît que tu préfères être courtisée avec l'arc et la hache, plutôt qu'avec des phrases polies et avec la langue de la courtoisie.

- **3.** Partial alignment with compositional translation and additional or missing information

---

English: SO AT LAST BEGAN THE EVENING PAPER AT LA FORCE

French: C'est ainsi qu'enfin débuta le journal du soir à la Force, le jour où la pauvre Lucie avait vu danser la carmagnole.

- **4.** Correct alignment with compositional translation and few additional or missing information

---

English: THE NIGHT WAS DARK AND A COLD WIND BLEW

French: La nuit était sombre; le vent âpre et froid chassait devant lui avec rage les nuages rapides.

- **5.** Correct alignment and fully compositional translation

---

English: WHAT IS A CAUCUS RACE

French: Qu'est-ce qu'une course cocasse?

4.2. Results

Table 4. reports our human evaluations for the 4 chapters. The first thing that we can notice is that the alignment quality is higher for chapters with higher confidence scores. The first evaluation (speech alignement ; scale 1-3) shows an average score of 2.89/3 which confirms that our re-segmentation of speech signals worked correctly. The second evaluation (bilingual alignment ; scale 1-5) shows an average score of 3.84/5. Some sentences were found uncorrectly aligned but overall, the alignment quality can be considered as correct. The main reason why the average alignment score varies between chapters is reflected by the translations compositionnality. Also, the dictionary that we used for bilingual alignments is inadequate for old texts and results in lower overall confidence scores.
We also computed automatic correspondence scores obtained with a cross-language textual similarity detection between transcriptions and their translations (Ferrero et al., 2016). Our idea was to add another automatic score in addition to $hunalign$ score. We computed the correlation between human evaluation scores and $hunalign$ scores and obtained a correlation of 0.41. The same correlation was obtained between human evaluation scores and those obtained automatically with method of (Ferrero et al., 2016). This shows that automatic alignment scores are reasonably correlated with human judgments and could be used to extract a subset of the best alignments by ranking them according to $hunalign$ score for instance.

5. Conclusion

We have presented a large corpus (236h) which is an augmentation of *Librispeech* in order to provide a bilingual speech-text corpus for direct (end-2-end) speech translation experiments. The methodology described here could be used in order to add other languages than French (German, Spanish, etc.) to our augmented *Librispeech*. The current corpus contains several ancient texts, so it would also be interesting to extend it to other kinds of corpora: different speaking styles (not only read speech), more contemporary texts, etc.
For direct speech translation experiments, preliminary experiments have been done recently and will be presented at next ICASSP 2018 conference (Bérard et al., 2018). Our online repository[15] provides a data split for speech translation experiments and results show that it is possible to train compact and efficient end-to-end speech translation models in this setup, but the dataset is challenging (BLEU score around 15 for direct speech translation task - more details in (Bérard et al., 2018)).

[15] see https://persyval-platform.univ-grenoble-alpes.fr/DS91/detaildataset

6. Bibliographical References

Adda, G., Stücker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon, F., and Zerbian, S. (2016). Breaking the unwritten language barrier: The Bulb project. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia.

Anastasopoulos, A. and Chiang, D. (2017). A case study on using speech-to-translation alignments for language documentation. *arXiv preprint arXiv:1702.04372*.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The semantic web*, pages 722–735.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bérard, A., Pietquin, O., Servan, C., and Besacier, L. (2016). Listen and translate: A proof of concept for end-to-end speech-to-text translation. In *NIPS workshop on End-to-end Learning for Speech and Audio Processing*.

Bérard, A., Besacier, L., Kocabiyikoglu, A. C., and Pietquin, O. (2018). End-to-end automatic speech translation of audiobooks. In *Accepted to Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Blachon, D., Gauthier, E., Besacier, L., Kouarata, G.-N., Adda-Decker, M., and Rialland, A. (2016). Parallel speech collection for under-resourced language studies using the LIG-Aikuma mobile device app. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia, May.

Federmann, C. and Lewis, W. D. (2016). Microsoft speech language translation (mslt) corpus: The iwslt 2016 release for english, french and german.

Ferrero, J., Agnes, F., Besacier, L., and Schwab, D. (2016). A multilingual, multi-style and multi-granularity dataset for cross-language textual similarity detection.

Fraser, N. (2012). google-diff-match-patch-diff, match and patch libraries for plain text.

Matusov, E., Leusch, G., Bender, O., and Ney, H. (2005). Evaluating machine translation output with automatic sentence segmentation. In *International Workshop on Spoken Language Translation (IWSLT) 2005*.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.

Post, M., Kumar, G., Lopez, A., Karakos, D., Callison-Burch, C., and Khudanpur, S. (2013). Improved speech-to-text translation with the fisher and callhome spanish-english speech translation corpus.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, number EPFL-CONF-192584. IEEE Signal Processing Society.

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., and Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam Studies in the Theory and History of Linguistic Science Series 4*, 292:247.

Weiss, R. J., Chorowski, J., Jaitly, N., Wu, Y., and Chen, Z. (2017). Sequence-to-sequence models can directly transcribe foreign speech. *arXiv preprint arXiv:1703.08581*.

Evaluating Domain Adaptation for Machine Translation Across Scenarios

Thierry Etchegoyhen[1], Anna Fernández Torné[2], Andoni Azpeitia[1], Eva Martínez Garcia[1] and Anna Matamala[2]

[1]Vicomtech, Donostia / San Sebastián, Spain [2]Universitat Autònoma de Barcelona, Barcelona, Spain

[1]{tetchegoyhen, aazpeitia, emartinez}@vicomtech.org [2]{Ana.Fernandez.Torne, Anna.Matamala}@uab.cat

Abstract

We present an evaluation of the benefits of domain adaptation for machine translation, on three separate domains and language pairs, with varying degrees of domain specificity and amounts of available training data. Domain-adapted statistical and neural machine translation systems are compared to each other and to generic online systems, thus providing an evaluation of the main options in terms of machine translation. Alongside automated translation metrics, we present experimental results involving professional translators, in terms of quality assessment, subjective evaluations of the task and post-editing productivity measurements. The results we present quantify the clear advantages of domain adaptation for machine translation, with marked impacts for domains with higher specificity. Additionally, the results of the experiments show domain-adapted neural machine translation systems to be the optimal choice overall.

Keywords: Machine Translation, Domain Adaptation, Quality Evaluation, Productivity Evaluation, Subjective Evaluation

1. Introduction

Statistical machine translation (SMT) (Brown et al., 1990) has been the dominant approach to automated translation for the last two decades, with neural machine translation (NMT) (Bahdanau et al., 2015) quickly becoming the new main paradigm in academic research and the industry, on the basis of the improvements it provides across the board (Toral and Sánchez-Cartagena, 2017). The data-driven nature of both approaches conditions the quality of their output to the availability of large volumes of adequate training resources for a given domain. However, domain-specific resources are usually scarce, thus making proper domain adaptation as much a challenge as it is a goal in developing accurate machine translation (MT) systems.

Domain adaptation has been extensively explored within SMT, with numerous studies focusing on the selection of supplementary data (Axelrod et al., 2011; Gascó et al., 2012; Eetemadi et al., 2015), translation model combination (Foster and Kuhn, 2007; Sennrich, 2012) or the integration of external information (Bisazza et al., 2011), to cite only a few. In NMT, domain adaptation is a more recent endeavour, with fine tuning currently the main method to gear generic translation networks towards specific domains (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016; Crego et al., 2016).

Progress in machine translation technology has also given rise to large generic machine translation systems, many of which are freely available online. The increasing translation quality they provide, in part due to growing user feedback and training data covering multiple domains, has made them popular alternatives even in cases where domain-adapted systems might be better suited, although this specific aspect has not been fully evaluated yet. So far, large online translation systems have been essentially compared to academic systems for news-related translation (see, e.g., (Toral et al., 2011; Bojar et al., 2016)), rather than to systems tuned for the kind of specific domains that are more typical in the translation industry.

Machine translation quality and usefulness can be evaluated under various modalities. Quality can be measured via automated metrics such as BLEU (Papineni et al., 2002) or TER (Snover et al., 2006), among others. Additionally, or alternatively, direct assessments of translation quality can be made by professional translators or native speakers, and usefulness can be assessed via measurements of productivity gains and losses when post-editing machine-translated text. Over the years, human evaluations along these lines have shown the usefulness of machine translation in various scenarios (Plitt and Masselot, 2010; Pinnis et al., 2013; Etchegoyhen et al., 2014; Koehn and Germann, 2014). With the rise of neural machine translation, recent studies have also centred on comparing statistical and neural machine translation in different scenarios (Zoph et al., 2016; Castilho et al., 2017b).

In this paper, we focus on evaluating the benefits of domain adaptation in three distinct scenarios involving different domains and language pairs, with varying degrees of domain specificity and available in-domain resources. Domain-adapted statistical and neural machine translation systems are compared to each other and to generic online systems, thus providing an evaluation of the main options in terms of automated translations. Alongside automated translation metrics, we present experimental results involving professional translators in terms of quality assessment, subjective evaluations of the task and post-editing productivity measurements.

In the remainder of this paper, we describe the corpora and machine translation systems that were prepared, the design of the human quality and productivity evaluations, and the results in terms of automated metrics, human subjective assessments on a wide range of aspects, and objective analyses of post-editing results on the tasks carried by professional translators.

2. Domain Adaptation Scenarios

In order to evaluate different real-life scenarios for machine-translated content, we selected the three domains described below for our experiments:

DOMAIN	LANGS	TRAIN	DEV	TEST
MTOOL	ES-DE	25,256	1,984	3×50
ELEV	ES-FR	106,521	1,996	3×50
INTORG	EN-ES	23,138	1,998	3×50

Table 1: In-domain corpora statistics (number of parallel segments)

LANG	OOD	CORPUS								
		OPSUBS	UN	EUROP	JRC	NEWSCOM	CCRAWL	TED	WIKI	TOTAL
ES-DE	Generic	550,000	99,575	533,900	543,594	201,091	0	0	0	*1,784,385*
ES-FR	Generic	500,000	500,000	500,000	500,000	191,080	0	0	0	*2,191,079*
	WCrawl	0	0	0	0	0	0	0	13,177	*13,177*
EN-ES	Generic	499,000	499,000	551,000	293,586	206,137	499,000	156,895	0	*2,468,292*
	Pool	0	8,079,790	1,604,400	697,557	207,137	0	157,895	0	*10,410,392*

Table 2: Out-of-domain corpora statistics (number of parallel segments)

- MTOOL: Industrial documentation of machine tool components and processes.
- ELEV: Installation and maintenance documentation of elevators.
- INTORG: Reports and press releases of international non-profit organisms.

All three domains are representative of the various domains typically handled by translation services providers, each one being characterized by its own specialised vocabulary and constructions, which range from highly specific, as is the case for the MTOOL domain, to more general, as with the INTORG domain.

In addition to choosing markedly distinct domains, we selected different language pairs for each evaluation scenario: Spanish-German for MTOOL, Spanish-French for ELEV, and English-Spanish for INTORG.

The INTORG scenario is meant to evaluate domain adaptation in the least favourable case, i.e. where freely available training resources are abundant: the topics and language found in the texts of international organisms are rather close to those available in the United Nations and Europarl corpora (Eisele and Chen, 2010; Koehn, 2005), for instance; English-Spanish is also the language pair with the most abundant available parallel corpora, see, e.g., the resources in the OPUS repository (Tiedemann, 2012).

The other two scenarios and language pairs were chosen as representative of cases with strong demand in terms of internationalisation and relatively limited training resources, which represents a rather typical state of affairs in the translation services industry.

Finally, the three selected domains vary in terms of volumes of available corpora, both in-domain and related out-of-domain. These disparities in terms of amounts of training data are rather typical in the development of domain-adapted machine translation systems, with scarce resources for highly specific domains a particularly common scenario. Additionally, out-of-domain data that could complement scarce in-domain data might be limited in volume for highly specific domains. The task of domain adaptation is thus dependent on both the available in-domain data and the amounts of exploitable out-of-domain data for a given domain. The corpora collected for the domains at hand are described in more detail in the next section.

3. Corpora

In-domain data were provided in the form of translation memories for all three domains, with an additional collection of Spanish technical manuals previously translated into French for the ELEV domain. Since the documents provided in the latter case were unpaired, document alignment was performed using an in-house file name matcher, exploiting strong file naming consistency, and sentences were then aligned with Hunalign (Varga et al., 2005). As shown in Table 1, training data were particularly scarce for the MTOOL and INTORG domains, with at most 25 thousand unique parallel segments.

From the original data, we extracted around 2000 segments as development sets per domain, to serve as either tuning sets for SMT systems or validation sets for NMT systems. As test sets, for each domain we extracted 3 sets of 50 sentence pairs which had to be representative of the domain in terms of average sentence length and vocabulary, and be coherent in sequence, i.e., sampling was not performed randomly on a per sentence basis. These conditions were meant to allow for human quality and productivity evaluations that centred on realistic translation scenarios, as described in Section 5..

To complement the scarce in-domain datasets, we compiled the out-of-domain data described in Table 2.[1] Distinct freely available corpora were selected depending on the language pair and domain. The *Generic* datasets were prepared mainly to serve as basis for the NMT models, to be further fine-tuned with in-domain data for domain adaptation. For each corpus selected to compose the generic multi-domain corpus, parallel sentence pairs were first sorted by increasing perplexity scores according to language models trained on the entire monolingual sides of each parallel corpus, where the score was taken to be the arithmetic mean of source and target perplexities. Subsets of the ranked corpora were then selected to compose the

[1] Unless described otherwise, all corpora were downloaded from the OPUS website (*op. cit.*).

final corpus, with an upper selection bound taken to be either the median average perplexity score or the top n pairs if selecting up to median perplexity would result in over representing the corpus.

For ES-FR, we also included a small corpus, *WCrawl*, created from Wikipedia with an in-house crawler targeting domain terminology and the STACC$_w$ comparable sentence aligner (Azpeitia et al., 2017). Finally, for EN-ES, we prepared a data pool based on the concatenation of all corpora relevant to the domain of international news and regulations.

4. Models

As mentioned in Section 1., we aimed to evaluate the benefits of domain adaptation in various scenarios. With the aforementioned paradigm shift towards neural machine translation, it became necessary to further evaluate the potential differences between domain-adapted SMT and NMT machine translation systems. Thus, for each domain adaptation scenario described in Section 2. we trained two such domain-adapted systems. Statistical MT systems were standard phrase-based models built with the Moses toolkit (Koehn et al., 2007), with phrases of maximum length 5 and n-gram language models of order 5 built with KenLM (Heafield, 2011). Neural MT systems follow the attention-based encoder-decoder approach (Bahdanau et al., 2015) and were built with the OpenNMT toolkit (Klein et al., 2017). Generic translations were obtained from Google Translate in June 2017, where, to the best of our knowledge, ES-EN translations were generated by their NMT system and by their phrase-based SMT engines for the other two language pairs.

Several techniques are available to perform domain adaptation in SMT and we selected the method that gave the best results on the evaluation sets for each scenario.

For the INTORG domain, the optimal approach involved ranking the out-of-domain *Pool* dataset using the relative frequency ratio approach (RFR) of (Etchegoyhen et al., 2017) and selecting the best 1,000,000 sentence pairs as supplementary data. A phrase table was then created from the selected data and combined with the in-domain phrase table with the fill-up method of (Bisazza et al., 2011). In the ELEV domain, a similar approach was used, applying RFR ranking on the *Generic* dataset merged with the crawled data and selecting the best 98,845 sentence pairs, corresponding approximately to the size of the in-domain. Since manually revised domain-specific terms were available for this domain, we also included 162 phrasal term translations as favoured translation options using the XML-markup functionality in Moses, a domain adaptation technique readily available for SMT modelling. Finally, for the MTOOL domain we combined the phrases from the entire *Generic* dataset, via fill-up as well.

For all of our NMT models, domain adaptation was performed via fine-tuning (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016; Crego et al., 2016), i.e. by further training the generic networks on the in-domain data.

5. Human Evaluation

A field quasi-experiment, for which no random assignment of participants to treatment groups was applied, was conducted with 15 professional translators for the INTORG and ELEV domains, and 22 for the MTOOL domain. Participants performed the assigned tasks in a real-world environment, thus favouring external validity.

Following an approved ethical procedure, the experiment consisted of a remunerated assignment and a volunteer, optional part. The evaluation aimed to compare the three machine translation systems in the three domains previously described, and was performed taking three different aspects into account: quality, post-editing (PE) productivity, and attitude. We describe each aspect in turn below.

5.1. Quality Assessment

Quality was first assessed at the segment level. The quality of the raw MT segments was assessed by scoring their fluency and adequacy on a scale from 1 to 4 using the TAUS DQF on-line tool.[2] Fluency conveyed to what extent the translated segment flowed naturally with no grammatical or spelling mistakes and was considered genuine language by native speakers (Koehn and Monz, 2006). In turn, adequacy assessed the amount of information of the source segment that was actually present in the target one (Koponen, 2010). Comparing the three different translated versions of each source segment was also considered a valuable quality indicator, so that a ranking task was also conducted.

Quality was also measured at the document level, by means of post-questionnaires where participants were asked to give their subjective global perception of the texts in terms of the aforementioned fluency and adequacy, as well as PE necessity, PE easiness and PE effort, as defined below:

- FLUENCY: the overall level of fluency of the machine-translated text.
- ADEQUACY: the overall level of adequacy of the machine-translated text.
- NECESSITY: the need for post-editing, i.e. whether the machine-translated text required many modifications overall.
- EASINESS: the easiness of the post-editing task, i.e. whether the necessary edits were technically simple overall.
- EFFORT: the mental effort required by the post-editing task, i.e. whether the necessary edits were cognitively difficult overall.

5.2. Post-editing Productivity Measurement

While conducting the post-editing task, the TAUS DQF tool automatically calculated the time to edit, i.e. "the average number of words processed by the post-editor in a given timespan" (SPEED from now onwards) and the post-editing effort (hereafter, WORK), namely "the average percentage of word changes applied by the post-editor on the MT output provided" (TAUS, n.d.). SPEED and WORK were considered two relevant indicators of post-editing productivity.

[2] https://dqf.taus.net/

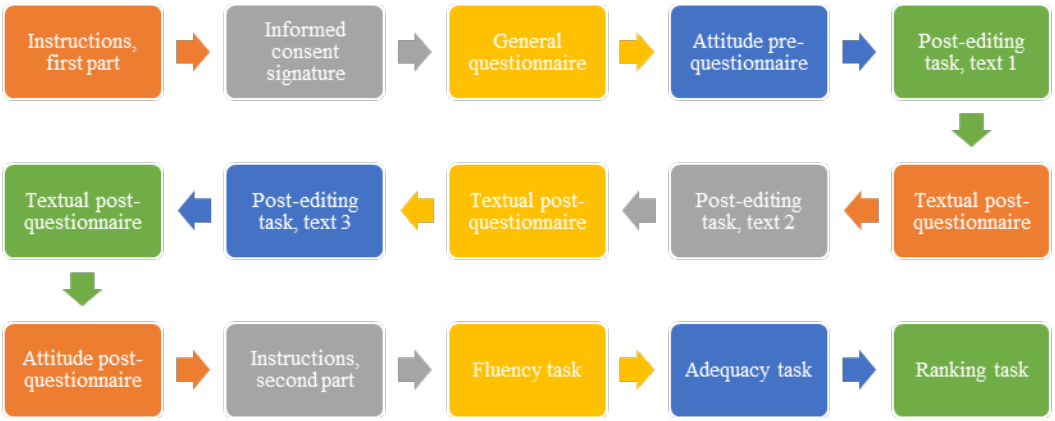


Figure 1: Experiment execution

DOMAIN	LANGS	MODEL	BLEU	METEOR	TER
MTOOL	ES-DE	SMT	19.830‡	35.260	69.378
		NMT	27.715† ∗	41.471	62.203
		GT	12.265	25.668	85.055
ELEV	ES-FR	SMT	62.524‡	74.627	25.550
		NMT	64.185†	76.062	23.100
		GT	18.857	37.955	63.050
INTORG	EN-ES	SMT	29.617	51.978	55.837
		NMT	32.726	54.467	50.620
		GT	33.024	55.068	50.646

Table 3: Results on automated machine translation metrics

5.3. Attitude Evaluation

Using scales from 1 to 10, participants were also asked to give their opinions on seven ratings in relation to their attitude towards MT and PE, namely the quality of raw machine-translated texts, the usefulness of MT for translators, their inclination to use MT as a text to depart from, their interest in PE, the boredom and the mental effort involved in PE, and the quality of post-edited machine-translated texts. These questions were presented twice, both before and after the PE task via pre- and post-questionnaires. This was aimed to see whether the actual post-editing task had in any way influenced their previous attitudes.

5.4. Experiment Execution

The experiment was divided in two parts to be performed at the participants' best convenience, as its expected length according to the pilot test was around 4 hours. Participants were informed via e-mail of the tasks to be carried out in each part. The instructions for the first part were to fill in a questionnaire on participant demographics and previous professional experience, and a pre-questionnaire on their attitude towards MT and PE.

Then, they were required to post-edit three texts. The order was balanced to minimise the fatigue and order-of-presentation effects, and was indicated individually to each participant on a separate email. After post-editing each text, they were requested to fill in the corresponding quality assessment post-questionnaire.

As a last step, they were asked to fill in a post-questionnaire on their attitude containing the same questions as the pre-questionnaire. Once they had finished the first part, they were asked to perform the fluency and adequacy evaluation tasks, and the rank comparison task (see Section 5.1.) for the same 150 segments.

The overall process is summarised in Figure 1.

5.5. Participants

Participants were selected following an a priori non-probabilistic purpose sampling (Bryman, 2012) based on subjects who met the following criteria: they had to be professional translators in the considered specific language pair and also native speakers of the target language.

A total of 52 participants took part in the experiment, distributed as follows: 11 female and 4 male EN-ES professional translators whose age ranged from 26 to 48 in the INTORG domain, 17 female and 5 male ES-DE professional translators ranging from 33 to 67 years old in the MTOOL domain, and 14 female and 1 male ES-FR professional translators whose ages ranged from 27 to 64 in the ELEV domain. Data from one participant in the MTOOL domain were not recorded due to technical problems. All participants but one in the MTOOL experiment had reached first cycle university studies.

6. Results

We first present results in terms of automated metrics, followed by a condensed representation of the human evaluation outcomes. A summary of all results is then added and discussed.

TASK	MEASURE	GT	NMT	SMT
SEGMENT-LEVEL QUALITY	ADEQUACY	3.56 ±0.70 † ‡	3.25 ±0.85 ∗	2.67 ±1.02
	FLUENCY	3.00 ±0.86 † ‡	2.75 ±0.97 ∗	2.00 ±0.93
	RANKING	1.39 ±0.61 † ‡	1.75 ±0.70 ∗	2.36 ±0.72
DOCUMENT-LEVEL QUALITY	FLUENCY	7.25 ±1.36 ‡	6.90 ±1.92 ∗	5.15 ±1.95
	ADEQUACY	8.67 ±0.78 †	7.90 ±0.94	7.54 ±1.56
	NECESSITY	5.83 ±2.08 ‡	7.00 ±1.90	7.92 ±1.18
	EASINESS	6.33 ±2.27	5.45 ±2.77	5.38 ±1.89
	EFFORT	6.90 ±2.31	7.64 ±2.62	7.38 ±1.04
PRODUCTIVITY	SPEED	1505 ±907.91 ‡	1236 ±560.60	957 ±375.83
	WORK	14.91 ±16.38 † ‡	19.76 ±16.91 ∗	27.28 ±18.56

Table 4: INTORG mean results and standard deviations for all human assessments

TASK	MEASURE	GT	NMT	SMT
SEGMENT-LEVEL QUALITY	ADEQUACY	2.55 ±0.96	3.25 ±0.86 † ∗	2.56 ±1.02
	FLUENCY	1.91 ±0.91	2.92 ±0.96 † ∗	1.91 ±1.06
	RANKING	2.04 ±0.72 ‡	1.49 ±0.72 † ∗	2.14 ±0.76
DOCUMENT-LEVEL QUALITY	FLUENCY	3.33 ±1.15	5.00 ±1.65 † ∗	3.31 ±2.17
	ADEQUACY	4.83 ±2.21	6.92 ±1.44 †	5.54 ±2.54
	NECESSITY	8.67 ±0.78	7.08 ±1.98 †	8.46 ±1.45
	EASINESS	4.92 ±2.97	5.25 ±2.67	4.62 ±3.07
	EFFORT	8.25 ±2.18	7.41 ±2.19	8.31 ±1.89
PRODUCTIVITY	SPEED	1018 ±500.40	1207 ±630.81	996 ±483.26
	WORK	37.56 ±21.51	20.49 ±20.82 † ∗	37.14 ±20.21 ‡

Table 5: MTOOL mean results and standard deviations for all human assessments

6.1. Automated metrics

We computed the performance of each model on the same test sets used for the human evaluations, in terms of BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005) and TER (Snover et al., 2006). For a closer comparison between automated and human evaluations, all machine-translated files were evaluated on cased detokenised output. Results are shown in Table 3.[3]

The first noticeable result is the strong benefit of domain adaptation for the MTOOL and ELEV domains, with large scoring differences on all metrics using either an SMT or an NMT domain-adapted system over generic GT engines. The only scenario where this result is not confirmed involves the INTORG domain. In this case, as previously described, the domain is the least restricted of the three, covering world news and events for which large amounts of training data are freely available. This wide scope domain demonstrates the convergence of various systems when in-domain data is not a marked provider of the most relevant information. Results from this domain also show the competitive scores achievable with comparatively small amounts of training data when compared to generic engines trained on significantly larger amounts of data. Overall, domain adaptation appears to be a necessary step to optimise translation quality, despite recent progress in the development of large generic natural machine translation systems.

A second important result is the effectiveness of current neural machine translation for narrow domains. Recent work had shown the need of large amounts of training data for NMT modelling, showing that SMT performed comparatively better in low resource scenarios (Zoph et al., 2016). Our experiments feature two narrow domains, with low amounts of parallel training data and high domain specificity, for which fine-tuned NMT models achieved the best results.[4] This outcome was obtained using simple fine tuning over generic models, an approach which has some inherent limitations such as need to restrict the adapted models to the vocabulary of the existing generic network. Domain modelling is thus limited in this approach, with domain-specific vocabulary handled via additional mechanisms such as unknown source word copies. Improved methods of domain adaptation for NMT are thus likely to provide gains to an already strong baseline for narrow domains.

Finally, in terms of automated metrics, SMT performed well in two out of three domains, reaching statistically comparable results to the ones obtained with domain-adapted

[3] Statistical significance was computed for the BLEU metric on the merged files for each domain via bootstrap resampling (Koehn, 2004). † indicates statistical significance, at $p < 0.05$, between NMT and GT; ‡ between SMT and GT; and ∗ between NMT and SMT.

[4] Note that, for the INTORG and ELEV domains, our NMT models were trained on more data than their SMT counterparts, the latter being built following a standard set-up where in-domain and out-of-domain data are not merged and only a portion of the out-of-domain data is selected. Additional experiments not reported here showed that using the entire out-of-domain dataset for SMT did not provide significant improvements over the approach reported here.

TASK	MEASURE	GT	NMT	SMT
SEGMENT-LEVEL QUALITY	ADEQUACY	2.40 ±0.96	3.56 ±0.68 †	3.43 ±0.77 ‡
	FLUENCY	1.99 ±1.06	3.06 ±0.90 †	2.91 ±0.98 ‡
	RANKING	2.33 ±0.68	1.47 ±0.63 †	1.47 ±0.68 ‡
DOCUMENT-LEVEL QUALITY	FLUENCY	2.90 ±1.83	7.40 ±1.74 †	7.10 ±1.83 ‡
	ADEQUACY	5.10 ±2.42	8.10 ±0.93 †	8.10 ±0.93 ‡
	NECESSITY	8.60 ±1.88	5.40 ±2.60 †	5.60 ±2.74 ‡
	EASINESS	5.20 ±2.49	7.00 ±2.12	6.10 ±1.62
	EFFORT	8.10 ±1.27	7.10 ±2.47	5.80 ±2.22 ‡
PRODUCTIVITY	SPEED	881 ±294.85	1462 ±582.93 †	1477 ±485.63 ‡
	WORK	39.44 ±21.02	11.12 ±13.56 †	10.72 ±13.54 ‡

Table 6: ELEV mean results and standard deviations for all human assessments

NMT, although with absolute scores consistently below those achieved with NMT. It is worth noting that the SMT results were obtained with domain adaptation techniques that have been extensively researched and time-tested over the years. Thus, it is unlikely that different domain adaptation methods for statistical machine translation would provide significant gains overall, which in turn places domain-adapted NMT as the currently optimal approach in terms of automated metrics.

6.2. Human evaluation

In this section, we first present the results in terms of metrics, both objective and subjective, and then summarise the eventual changes in perception of MT and PE for the translators who participated in the evaluation.

6.2.1. Metrics

A statistical analysis was carried out using IBM SPSS V. 20, setting the significance level at 0.05. For qualitative data such as adequacy, fluency and ranking at segment level, chi-square tests were used to compare the distribution of the assessments by language pair. Data relating to textual quality were considered discrete numerical variables, so that a Mann-Whitney U test was used for the comparison of groups. For the continuous numerical variables PE speed and work, normality of distributions was assessed by using the Shapiro-Wilk test. As data were not normally distributed, the Bonferroni-corrected Mann-Whitney U test was used for multiple comparisons.

Intra-class correlation coefficient (ICC) estimates and their 95% confidence intervals were used as inter-rater reliability indexes, which for all three domains resulted in excellent levels of reliability (all above 0.92) for the quality assessment variables at the segment level.

In the INTORG domain, the means of all aspects assessed in the case of GT indicated better results than those obtained with either NMT or SMT. Thus, the means were lower in ranking, necessity, effort and work, and higher in the other categories where higher scores indicate better results. In turn, all NMT means but one (effort) showed better results than those of SMT, as shown in Table 4.

Examining post-editing edit distances, shown in Figure 2(a), confirms the ranking of the systems in this domain. Thus, in the case of GT more than 37% of the segments had an edit distance of 0 and 22% an edit distance of 1. For these same two distances, SMT featured 11.5% and 18.8% of the segments, respectively, whereas for NMT the proportions were 19.9% and 24.3%, respectively. Additionally, GT exhibits a gradual reduction in the percentage of segments as the number of edits increases, whereas for NMT the number of segments increases between distances of 0 and 1, and SMT has a higher percentage of segments with an edit distance of 2, beyond 20%.

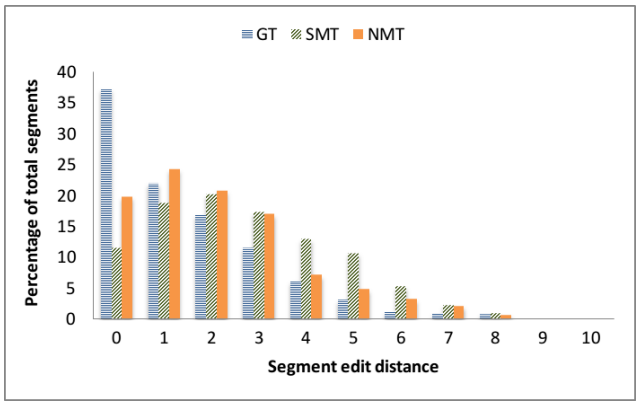
The conclusions are different in the MTOOL domain. As shown in Table 5, the leading position for all means in this case was for NMT, with GT and SMT presenting very close means in all items assessed. The distribution of results in terms of edit distances, shown in Figure 2(b), illustrates the differences between systems in this narrow domain. The first noticeable result is the extremely large difference for edit distances of 0, with 34.42% for NMT as opposed to 9.56% for GT and 11.43% for SMT. The NMT system cumulates more than 60% of the segments in the lowest edit distances, from 0 to 2, as opposed to 27% for GT and 25% for SMT. In terms of human evaluation, domain-adapted NMT was thus the optimal system in the MTOOL domain, with GT and ST showing comparable results.

In the ELEV domain, NMT had the lead again, as shown in Table 6. However, in this case SMT means were closer to NMT's means, which left GT as the worst performing system. This ranking of the systems is again illustrated by the distribution of edits shown in Figure 2(c). For both SMT and NMT, the dominant edit distance was 0, with 45.9% and 46.5%, respectively, whereas for GT the most frequent edit distances were 5 and 6, with 17.7% and 18.5%, respectively. In this domain, which exhibits a comparable domain-specificity to MTOOL but larger amounts of in-domain data, domain adaptation with NMT appears to be the optimal option, but the SMT system shows a comparable strong performance, with only the generic GT system performing markedly worse on all metrics.

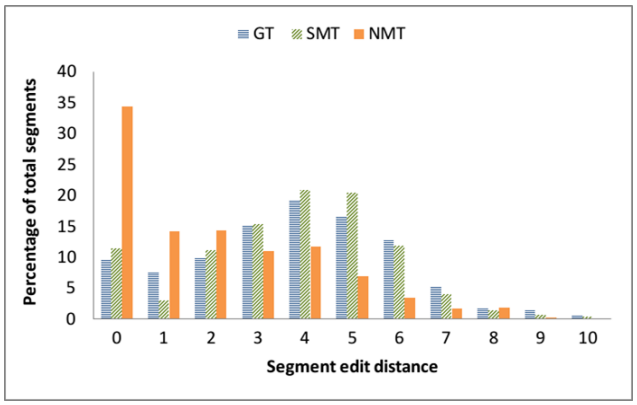
6.2.2. Attitude

We now summarise the results regarding the translators' changes in attitude before and after completing the tasks.

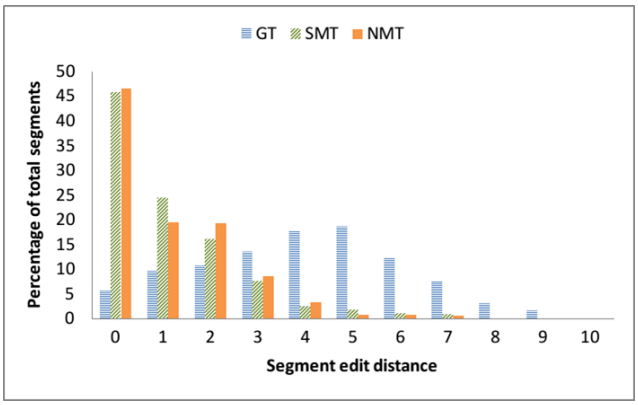
For INTORG, there was a positive evolution regarding aspects such as quality and utility of machine translation, as well as inclination towards using machine-translated texts as a starting point for translation. On the other hand, interest in post-editing lowered slightly, while boredom and



(a) Edit distances in the INTORG domain



(b) Edit distances in the MTOOL domain



(c) Edit distances in the ELEV domain

Figure 2: Edit distances per domain

MEASURE	INTORG	MTOOL	ELEV
ADEQUACY	r = 0.10, $p > 0.05$	r = -0.09, $p > 0.05$	r = -0.08, $p > 0.05$
FLUENCY	r = 0.36, $p < 0.001$	r = 0.48, $p < 0.001$	r = 0.61, $p < 0.001$
RANKING	r = -0.20, $p < 0.001$	r = -0.30, $p < 0.001$	r = -0.68, $p < 0.001$
SPEED	r = 0.04, $p > 0.05$	r = 0.05, $p > 0.05$	r = -0.26, $p < 0.001$
WORK	r = -0.51, $p < 0.001$	r = -0.50, $p < 0.001$	r = -0.69, $p < 0.001$

Table 7: Correlations between BLEU and human assessments

TASK	MEASURE	DOMAIN								
		ELEV			MTOOL			INTORG		
		GT	NMT	SMT	GT	NMT	SMT	GT	NMT	SMT
SEGMENT-LEVEL QUALITY	ADEQUACY	3 †	1 ‡	2 ∗	3 †	1	2 ∗	1 †	2 ‡	3 ∗
	FLUENCY	3 †	1 ‡	2 ∗	2 †	1 ‡	3 ∗	1 †	2 ‡	3 ∗
	RANKING	3 †	1 ‡	1 ∗	2 †	1 ‡	3 ∗	1 †	2 ‡	3 ∗
DOCUMENT-LEVEL QUALITY	FLUENCY	3 †	1 ‡	2 ∗	2 †	1	3 ∗	1 †	2 ‡	3 ∗
	ADEQUACY	3 †	1 ‡	1	3 †	1	2	1 †	2	3
	NECESSITY	3 †	1 ‡	2	3 †	1	2	1	2 ‡	3
	EASINESS	3	1	1	2	1	3	1	2	3
	EFFORT	3	2 ‡	1	2	1	3	1	3	2
PRODUCTIVITY	SPEED	3 †	2 ‡	1	2	1	3	1	2 ‡	3
	WORK	3 †	2 ‡	1	3 †	1	2 ∗	1 †	2 ‡	3 ∗
AUTOMATED METRICS	BLEU	3 †	1 ‡	2 ∗	3 †	1	2 ∗	1	2	3
	METEOR	3	1	2	3	1	2	1	2	3
	TER	3	1	2	3	1	2	1	2	3

Table 8: Summary of comparative results

the perception of the mental effort needed for post-editing increased. Inferential statistical analysis showed that there were statistically significant changes, with a p-value less than 0.05, only in the case of perception of machine translation quality and boredom.

In the MTOOL domain, the attitudes evolved negatively in all aspects except usefulness of machine translation and interest in post-editing. Perception of the quality of machine translation lowered, as did the interest in using machine-translated texts as input for translation. Results were statistically significant only in the case of perception of cognitive effort, which increased after completion of the task.

Finally, for the ELEV domain all positive attitude indicators increased, except for quality of post-edited texts, which maintained the same score, and boredom, which also increased. Thus, after completion of the task, the perception of quality of machine translation increased, as did its perceived usefulness and the interest in using it as input. The required mental effort was also perceived as lower after completion of the task. The changes were not statistically significant in any of the aspects, though, with p-values above 0.05.

Although attitude changes were not statistically significant in most cases, they are in line with the results on the previously discussed metrics, with an overall increase in positive perception of the post-editing task in the domain with markedly better translations, namely ELEV, and mixed results for the other two domains.

6.3. Summary

As shown in Table 7, there are relevant correlations between segment-level BLEU and the human assessments for fluency, ranking and WORK, particularly in the ELEV domain. Thus, the higher the fluency and ranking results and the lower the WORK, the higher the BLEU metric obtained. Although BLEU usually shows higher correlations with human judgements at the document level than at the segment level, in these experiments the correlations were significant.

Results in terms of both automated metrics and human assessments show an almost perfect match, as seen in Table 8, which includes the position each engine occupies taking into account every aspect assessed and the statistical significance of the differences observed between each pair of systems. Thus, in the INTORG domain GT is almost unanimously considered the best system by all items assessed, while SMT is deemed the worst performing one. Likewise, NMT is the system obtaining the best results in the MTOOL scenario, while GT and SMT vie for the last position. In the ELEV domain, NMT obtains again the best results, although closely followed by SMT, which clearly leaves GT as the worst classified system.

Considering the different domains selected for the experiments, their specificity as well as the amount of available in-domain data, the results were not unexpected but the experiments performed provide a quantified view of the impact of domain adaptation. Thus, for the two domains that were more specific, domain-adapted systems in one form or another provided clear advantages that are reflected in all metrics, automated, based on human subjective evaluation, or based on objective post-editing metrics.

Overall, adapting neural machine translation systems to a specific domain proved the optimal approach, performing better where domain-specificity was higher, and competing with a large state-of-the-art generic translation system while being trained on only a relatively small amount of data overall. This result shows the progress of NMT in general, as it performed better than statistical machine translation systems even in the case of highly specific domains. Note also that the NMT systems performed better overall than the SMT ones in terms of adequacy as well, in contrast with the results described in (Castilho et al., 2017a), where neural models performed better than statistical ones in terms of fluency, but not always in terms of adequacy.

It is worth noting also that, as shown by the ELEV domain, SMT systems could remain competitive in domain adaptation scenarios, although it is likely that future more sophisticated domain-adaptation methods for NMT will likely extend the gap between the approaches.

7. Conclusions

We have described the evaluation of the benefits of domain adaptation for machine translation under different scenarios that involve unrelated domains and language pairs, with varying degrees of domain specificity and amounts of training data. Our protocols include domain-adapted statistical and neural machine translation systems, as well as a large generic online system, thus addressing the main options currently available in terms of automated translation.

The human evaluation, which involved professional translators, covered quality assessments, post-editing productivity measurements, as well as attitude evaluations. An in-depth statistical analysis of the results was provided, along with an evaluation of eventual changes in perception of the task by the participants.

Overall, results in terms of both automated metrics and human assessments show the benefits of domain adaptation, with marked gains across the board for domain-adapted systems on all metrics for the more specific domains. Although not unexpected, given that domain-specific knowledge could be expected to positively impact data-driven translation systems, these results have been substantiated on a wide range of aspects.

Finally, the reported experiments show the comparative effectiveness of domain-adapted neural machine translation across the board, confirming that the paradigm shift that has taken place in the field towards neural machine translation can be considered adequate as well for the highly-specific scenarios that are common in the translation industry.

8. Acknowledgements

This work was partially funded by the Spanish Ministry of Economy and Competitiveness via the AdapTA project (RTC-2015-3627-7). We would like to thank MondragonLingua Translation & Communication as coordinator of the project, the anonymous LREC reviewers for their helpful feedback and suggestions, and the translators who participated in the experiments. Anna Matamala is a member of Transmedia Catalonia, a research group funded by the Catalan government under SGR call (2017SGR113).

9. Bibliographical References

Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 355–362. Association for Computational Linguistics.

Azpeitia, A., Etchegoyhen, T., and Martinez García, E. (2017). Weighted set-theoretic alignment of comparable sentences. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 41–45.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, Ann Arbor, Michigan*, volume 29, pages 65–72.

Bisazza, A., Ruiz, N., Federico, M., and Kessler, F.-F. B. (2011). Fill-up versus interpolation methods for phrase-based smt adaptation. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 136–143.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Logacheva, V., Monz, C., et al. (2016). Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, volume 2, pages 131–198.

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to machine translation. *Computational linguistics*, 16(2):79–85.

Bryman, A. (2012). *Social Research Methods*. Oxford University Press, Oxford.

Castilho, S., Moorkens, J., Gaspari, F., Calixto, I., Tinsley, J., and Way, A. (2017a). Is neural machine translation the new state of the art? *The Prague Bulletin of Mathematical Linguistics*, 108(1):109–120.

Castilho, S., Moorkens, J., Gaspari, F., Sennrich, R., Sosoni, V., Georgakopoulou, Y., Lohar, P., Way, A., Miceli Barone, A. V., and Gialama, M. (2017b). A Comparative Quality Evaluation of PBSMT and NMT using Professional Translators. In *Proceedings of Machine Translation Summit XVI*, Nagoya, Japan.

Crego, J., Kim, J., Klein, G., Rebollo, A., Yang, K., Senellart, J., Akhanov, E., Brunelle, P., Coquard, A., Deng, Y., et al. (2016). Systran's pure neural machine translation systems. *arXiv preprint arXiv:1610.05540*.

Eetemadi, S., Lewis, W., Toutanova, K., and Radha, H. (2015). Survey of data-selection methods in statistical machine translation. *Machine Translation*, 29(3-4):189–223.

Eisele, A. and Chen, Y. (2010). MultiUN: A Multilingual Corpus from United Nation Documents. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, pages 2868–2872. European Language Resources Association (ELRA), 5.

Etchegoyhen, T., Bywood, L., Fishel, M., Georgakopoulou, P., Jiang, J., Van Loenhout, G., Del Pozo, A., Maucec, M. S., Turner, A., and Volk, M. (2014). Machine translation for subtitling: A large-scale evaluation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 46–53.

Etchegoyhen, T., Azpeitia, A., and Martinez García, E. (2017). Exploiting Relative Frequencies for Data Selection. In *Proceedings of the 16th Machine Translation Summit*, volume 1: Long Papers, pages 170–184.

Foster, G. and Kuhn, R. (2007). Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 128–135, Stroudsburg, PA, USA. Association for Computational Linguistics.

Freitag, M. and Al-Onaizan, Y. (2016). Fast domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06897*.

Gascó, G., Rocha, M.-A., Sanchis-Trilles, G., Andrés-Ferrer, J., and Casacuberta, F. (2012). Does more data always yield better translations? In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 187–197, Stroudsburg, PA, USA. Association for Computational Linguistics.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of the 20th Annual Conference of the European Association for Machine Translation*.

Koehn, P. and Germann, U. (2014). The impact of machine translation quality on human post-editing. In *Proceedingds of the Workshop on Humans and Computer-Assisted Translation*, pages 38–46.

Koehn, P. and Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 102–121. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL*, pages 177–180. Association for Computational Linguistics.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 388–395.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the 10th Machine Translation Summit*, pages 79–86.

Koponen, M. (2010). Assessing machine translation quality with error analysis. In *Electronic proceedings of the*

KäTu symposium on translation and interpreting studies, volume 4, pages 1–12.

Luong, M.-T. and Manning, C. D. (2015). Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

Pinnis, M., Skadiņa, I., and Vasiļjevs, A. (2013). Domain adaptation in statistical machine translation using comparable corpora: case study for english latvian it localisation. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 224–235. Springer.

Plitt, M. and Masselot, F. (2010). A productivity test of statistical machine translation post-editing in a typical localisation context. *The Prague bulletin of mathematical linguistics*, 93:7–16.

Sennrich, R. (2012). Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 539–549. Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231.

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In *Proceedings of the 8th Language Resources and Evaluation Conference*, pages 2214–2218.

Toral, A. and Sánchez-Cartagena, V. M. (2017). A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1063–1073, Valencia, Spain, April. Association for Computational Linguistics.

Toral, A., Gaspari, F., Kumar Naskar, S., and Way, A. (2011). Comparative evaluation of research vs. online MT systems. In *Proceedings of the 15th Conference of the European Association for Machine Translation*.

Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V., and Nagy, V. (2005). Parallel corpora for medium density languages. In *Recent Advances in Natural Language Processing (RANLP 2005)*, pages 590–596.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575.

Upping the Ante: Towards a Better Benchmark for Chinese-to-English Machine Translation

Christian Hadiwinoto, Hwee Tou Ng
Department of Computer Science
National University of Singapore
{chrhad, nght}@comp.nus.edu.sg

Abstract

There are many machine translation (MT) papers that propose novel approaches and show improvements over their self-defined baselines. The experimental setting in each paper often differs from one another. As such, it is hard to determine if a proposed approach is really useful and advances the state of the art. Chinese-to-English translation is a common translation direction in MT papers, although there is not one widely accepted experimental setting in Chinese-to-English MT. Our goal in this paper is to propose a benchmark in evaluation setup for Chinese-to-English machine translation, such that the effectiveness of a new proposed MT approach can be directly compared to previous approaches. Towards this end, we also built a highly competitive state-of-the-art MT system trained on a large-scale training set. Our system outperforms reported results on NIST OpenMT test sets in almost all papers published in major conferences and journals in computational linguistics and artificial intelligence in the past 11 years. We argue that a standardized benchmark on data and performance is important for meaningful comparison.

Keywords: machine translation, benchmark, Chinese-to-English

1. Introduction

Over the years, there have been many published papers on machine translation (MT), proposing novel ideas by showing improvements over certain baselines. However, a paper often has a different experimental setup from the others. These variations include the approach (algorithm) and dataset. Ideally, research and development work on MT should be based on a benchmark of system setup with good performance. Otherwise, there is no support in asserting that a proposed method advances the state of the art.

Unfortunately, for Chinese-to-English MT, two widely spoken human languages and one of the most well-studied language translation directions in MT, there is no widely accepted standard benchmark for evaluation, comprising a standardized training set, development set, and test set. Throughout the past decade, Chinese-to-English translation has been most commonly performed on NIST OpenMT[1] test sets, trained on parallel and monolingual corpora from the Linguistic Data Consortium (LDC)[2].

Our goal in this paper is to propose a benchmark in evaluation setup for Chinese-to-English machine translation, such that the effectiveness of a new proposed MT approach can be directly compared to previous approaches. Towards this end, we also built a highly competitive state-of-the-art MT system trained on a large-scale training set. Our system outperforms reported results on NIST OpenMT test sets in almost all papers published in major conferences and journals in computational linguistics and artificial intelligence in the past 11 years.

The rest of this paper is organized as follows. Section 2 describes our MT approach. Section 3 elaborates our experimental setup. Section 4 presents our experimental results. Section 5 describes related work. Finally, Section 6 gives the conclusion.

[1] www.nist.gov/itl/iad/mig/open-machine-translation-evaluation

[2] catalog.ldc.upenn.edu

2. Neural Machine Translation

We built a neural machine translation (NMT) system based on the encoder-decoder approach with attention mechanism (Bahdanau et al., 2015). This NMT approach encodes an input sentence into a continuous representation by an encoder recurrent neural network (RNN) and produces translation output by a decoder RNN. The decoder RNN, through an attention mechanism, looks into different parts of the encoded input sentence while decoding is in progress.

2.1. Encoder-Decoder Model with Attention

Given a target language sentence $\mathbf{y} = (y_1, ..., y_n)$ and the corresponding source language sentence $\mathbf{x} = (x_1, ..., x_m)$, the neural machine translation model is formulated as

$$p(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{n} p(y_i|y_1, ..., y_{i-1}, \mathbf{x}) \quad (1)$$

in which the probability of the target word y_i at time step i is computed by the decoder RNN as follows:

$$p(y_i|y_1, ..., y_{i-1}, \mathbf{x}) = F(y_i, y_{i-1}, s_i, c_i) = t_i[y_i] \quad (2)$$

where F is a function to compute the probability of the word y_i to be generated at time step i and t_i is a vector having the size of the target language vocabulary, in which each vector dimension $t_i[y]$ stores the probability of a word y, computed as follows:

$$t_i = \text{softmax}(W_t(\tanh(U_t s_i + V_t E[y_{i-1}] + C_t c_i + b_t))) \quad (3)$$

where U_t, V_t, and C_t are matrices mapping the hidden state s_i, the embedding of the previous word $E[y_{i-1}]$, and the context vector c_i respectively to an intermediate vector representation, with b_t being the bias vector. Then

W_t transforms the intermediate vector representation to a vocabulary-sized probability vector.
The decoder hidden state at a time step i is computed by

$$s_i = g_y(E[y_{i-1}], s_{i-1}, c_i) \tag{4}$$

where g_y is the RNN unit function to compute the current hidden state given the hidden state of the previous time step, the previous word embedding, and the context.
Equation 3 indicates that the target word to be generated at a given time step takes into account the context vector c_i, which is a weighted sum of each annotation vector h_j, representing the source language sentence at position j:

$$c_i = \sum_{j=1}^{m} \alpha_{ij} h_j \tag{5}$$

in which the scalar weight α_{ij} for each h_j is computed by a softmax function:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{m} \exp(e_{ik})} \tag{6}$$

where e_{ij} is computed by

$$e_{ij} = v_a^T \tanh(W_a s'_i + U_a h_j + b_a) + \beta \tag{7}$$

where W_a and U_a are the weight matrices and b_a is the bias vector to compute a vector, which is then converted by the weight vector v_a and the bias term β into a scalar e_{ij}, i.e., the degree of matching between the target word at time step i and the input word at position j. This is conceptually a soft alignment model.
To compute the decoding hidden state s_i in Equation 4, we adopt an approach that incorporates the context c_i from the attention mechanism by using two transitions (Sennrich et al., 2017b). The decoder hidden state function $g_y(E[y_{i-1}], s_{i-1}, c_i)$ in Equation 4 first passes the embedding $E[y_{i-1}]$ of the input word y_{i-1} to the first recurrent unit function, resulting in an intermediate hidden state s'_i, which is computed by the decoder recurrent unit functions $g_{y,1}$ as:

$$s'_i = g_{y,1}(E[y_{i-1}], s_{i-1}) \tag{8}$$

and is passed to Equation 7. Then, the second recurrent unit function $g_{y,2}$ processes the context c_i defined in Equation 5 and the intermediate hidden state s'_i as follows:

$$s_i = g_{y,2}(c_i, s'_i) = g_{att}(E[y_{i-1}], s_{i-1}, c_i) \tag{9}$$

where $g_{att}(E[y_{i-1}], s_{i-1}, c_i)$ is a composition of $g_{y,1}$ and $g_{y,2}$. It is to be noted that as the decoder generates an output word y_i at current time step i, the input word at the time step is y_{i-1}. The recurrent unit function is described further in Section 2.2.
We made use of the bidirectional encoder RNN, where each annotation vector h_j is a concatenation of the forward and the backward RNN hidden states, $\overrightarrow{h_j}$ and $\overleftarrow{h_j}$ respectively, defined as follows:

$$h_j = [\overrightarrow{h}_j; \overleftarrow{h}_j] \tag{10}$$

$$\overrightarrow{h}_j = \overrightarrow{g}_x(E[x_j], \overrightarrow{h}_{j-1}) \tag{11}$$

$$\overleftarrow{h}_j = \overleftarrow{g}_x(E[x_j], \overleftarrow{h}_{j+1}) \tag{12}$$

where $\overrightarrow{g}_x$ is the forward RNN unit function to compute the RNN hidden state at the current encoding position j given the embedding of the current word $E[x_j]$ and the hidden state at the previous position, while $\overleftarrow{g}_x$ is the backward RNN unit function to compute the hidden state at j given the word embedding $E[x_j]$ and the hidden state at the next position.
Training an end-to-end NMT model is conducted by the back-propagation through time (BPTT) algorithm, which updates the parameters of the RNN while the time steps are unrolled, to minimize a cost function. The parallel training corpus is divided into mini-batches, each consisting of N parallel sentences. The NMT model parameters are updated in each mini-batch.
Translation decoding is performed by a beam search algorithm, which produces translation output sequentially in the target language order. NMT decoding proceeds by generating one word at each time step.
In NMT, as described in Equation 3, computing the probability involves mapping the hidden state vector to a vector with the dimension of the vocabulary size. Therefore, to make computation tractable, the NMT vocabulary size is limited. To cope with the limitation of the vocabulary size, we adopt fragmentation of words into sub-words of character sequences through the byte pair encoding (BPE) algorithm (Sennrich et al., 2016). This algorithm finds the N most frequent character sequences of variable length, through N character merge operations, and splits less frequent words based on this list of character sub-sequences.

2.2. Recurrent Unit Function

To compute the hidden state representations in Equations 4 and 10–12, we made use of recurrent unit functions with gate mechanism to control the flow of information from the input and the previous hidden state. There are two commonly adopted gate mechanisms in the encoder-decoder RNN NMT model, namely the long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) and the gated recurrent unit (GRU) (Cho et al., 2014).
The LSTM RNN unit consists of a memory cell μ_j and three gates, i.e., the input gate ι_j that controls the intensity of the new information to be stored in the memory cell, the forget gate f_j that controls how much to remember or to forget from the previous memory cell, and the output gate o_j that controls how much information is output to the hidden state from the memory. At each time step j, given the input χ_j, the hidden state η_j is formulated as:

$$\begin{aligned} \eta_j &= LSTM(\chi_j, \eta_{j-1}) \\ &= o_j \circ \tanh(\mu_j) \end{aligned} \tag{13}$$

where

$$\begin{aligned} \iota_j &= \sigma(W_\iota \chi_j + U_\iota \eta_{j-1} + b_\iota) \\ f_j &= \sigma(W_f \chi_j + U_f \eta_{j-1} + b_f) \\ \mu_j &= f_j \circ \mu_{j-1} + \iota_j \circ \tanh(W_\mu \chi_j + U_\mu \eta_{j-1} + b_\mu) \\ o_j &= \sigma(W_o \chi_j + U_o \eta_{j-1} + b_o) \end{aligned}$$

W and U denote the weight matrices transforming the input embedding and the previous hidden state into the corresponding outputs, and b denotes the bias vectors.

Meanwhile, the recurrent unit for GRU at each time step j consists of two gates, i.e., the update gate z_j and the reset gate r_j. At each time step j, given the time step input χ_j, the hidden state η_j is formulated as:

$$\begin{aligned}\eta_j &= GRU(\chi_j, \eta_{j-1}) \\ &= (1 - z_j) \circ \eta_{j-1} + z_j \circ \underline{\eta}_j \end{aligned} \tag{14}$$

where

$$\begin{aligned}\underline{\eta}_j &= \tanh(W\chi_j + U[r_j \circ \eta_{j-1}] + b) \\ z_j &= \sigma(W_z\chi_j + U_z\eta_{j-1} + b_z) \\ r_j &= \sigma(W_r\chi_j + U_r\eta_{j-1} + b_r)\end{aligned}$$

W and U denote the weight matrices transforming the input embedding and the previous hidden state to the corresponding outputs (denoted by the subscript), and b denotes the bias vectors.

As shown in Equations 13 and 14, both LSTM and GRU make use of gate mechanisms to control the information flow from the input and the hidden state. But unlike LSTM, GRU does not have the memory cell and the output gate. GRU proposes the hidden state $\underline{\eta}_j$ and interpolates each dimension with that of the previous hidden state, controlled by the update gate z_j. Meanwhile, the reset gate r_j controls the intensity of the previous hidden state to be taken into account in the current pre-computed hidden state.

The LSTM encoder re-defines Equations 11 and 12 respectively as:

$$\begin{aligned}\overrightarrow{h}_j &= \overrightarrow{g}_x(E[x_j], \overrightarrow{h}_{j-1}) = \overrightarrow{LSTM}(E[x_j], \overrightarrow{h}_{j-1}) \\ \overleftarrow{h}_j &= \overleftarrow{g}_x(E[x_j], \overleftarrow{h}_{j+1}) = \overleftarrow{LSTM}(E[x_j], \overleftarrow{h}_{j+1})\end{aligned}$$

while the GRU encoder re-defines the two equations respectively as:

$$\begin{aligned}\overrightarrow{h}_j &= \overrightarrow{g}_x(E[x_j], \overrightarrow{h}_{j-1}) = \overrightarrow{GRU}(E[x_j], \overrightarrow{h}_{j-1}) \\ \overleftarrow{h}_j &= \overleftarrow{g}_x(E[x_j], \overleftarrow{h}_{j+1}) = \overleftarrow{GRU}(E[x_j], \overleftarrow{h}_{j+1})\end{aligned}$$

with the embedding representation $E[x_j]$ of word x_j at position j as the input to the recurrent unit function.

For decoding, in Equations 8 and 9, both $g_{y,1}$ and $g_{y,2}$ can be instantiated by LSTM (Equation 13) as:

$$\begin{aligned}g_{y,1}(E[y_{i-1}], s_{i-1}) &= LSTM_1(E[y_{i-1}], s_{i-1}) \\ g_{y,2}(c_i, s'_i) &= LSTM_2(c_i, s'_i)\end{aligned}$$

or with GRU (Equation 14) as:

$$\begin{aligned}g_{y,1}(E[y_{i-1}], s_{i-1}) &= GRU_1(E[y_{i-1}], s_{i-1}) \\ g_{y,2}(c_i, s'_i) &= GRU_2(c_i, s'_i)\end{aligned}$$

where the input to the LSTM or GRU function for $g_{y,1}$ is the embedding of the previous word $E[y_{i-1}]$ and the input for $g_{y,2}$ is the context vector c_i.

2.3. Deep Recurrent Layers

Following (Sennrich et al., 2017a), we adopt deep RNN models for encoding and decoding. There are two alternatives to achieve this, namely the deep stacked RNN and deep transition RNN. The deep stacked RNN passes the whole input sequence to the first layer of RNN and feeds the sequence of the output hidden representations to the next layer of RNN. This is done subsequently depending on the number of RNN layers. Meanwhile, the deep transition RNN passes an input at each time step through a series of transitions (i.e., recurrent unit functions) and passes the hidden state of the last transition to the next time step.

2.3.1. Deep Stacked RNN

We adopt the deep stacked RNN that involves residual connection, summing the output of the previous RNN layer with the computed hidden state of the current RNN layer, and alternation of direction (Zhou et al., 2016), as illustrated in Figure 1. In this model, for each layer l and time step j, we need to distinguish between the computed hidden state of the current RNN unit function without residual connection, i.e., $\overrightarrow{h}^l_j$, and the hidden state which includes the residual connection, i.e., $\overrightarrow{w}^l_j$. The alternation of direction is designed such that the odd-numbered and even-numbered RNN layers process the sequence in the left-to-right and right-to-left directions respectively. In the stacked RNN with layer depth D_x, the forward encoder hidden state at time step j in Equation 11, $\overrightarrow{h}_j$, is computed as follows:

$$\begin{aligned}\overrightarrow{h}_j &= \overrightarrow{g}_x(E[x_j], \overrightarrow{h}_{j-1}) = \overrightarrow{w}^{D_x}_j \\ \overrightarrow{w}^{D_x}_j &= \overrightarrow{h}^{D_x}_j + \overrightarrow{w}^{D_x-1}_j \\ \overrightarrow{w}^1_j &= \overrightarrow{h}^1_j = \overrightarrow{g}^1_x(E[x_j], \overrightarrow{h}^1_{j-1}) \\ \overrightarrow{h}^{2k}_j &= \overrightarrow{g}^{2k}_x(\overrightarrow{w}^{2k-1}_j, \overrightarrow{h}^{2k}_{j+1}), \text{for } 1 < 2k \leq D_x \\ \overrightarrow{h}^{2k+1}_j &= \overrightarrow{g}^{2k+1}_x(\overrightarrow{w}^{2k}_j, \overrightarrow{h}^{2k+1}_{j-1}), \text{for } 1 < 2k+1 \leq D_x \\ \overrightarrow{w}^l_j &= \overrightarrow{h}^l_j + \overrightarrow{w}^{l-1}_j, \text{for } 1 < l \leq D_x\end{aligned}$$

In the above equations, $\overrightarrow{g}^l_x$ is the forward encoder recurrent unit function of a layer l in the deep RNN stack. It can be instantiated with LSTM or GRU. While the above equations compute the forward encoder hidden state, the backward encoder hidden state $\overleftarrow{h}_j$ is computed similarly by changing the arrow direction from right ($\rightarrow$) to left ($\leftarrow$) and swapping $j-1$ with $j+1$.

Since at each time step, the decoder has no knowledge of the next word, there is no alternation of direction therein. In addition, we only use the recurrent function with attention in the first layer of the RNN stack s^1_i, while the deeper layers are simple RNN without attention. Therefore, the decoder RNN with layer depth D_y computes the hidden state s_i in Equation 4, as follows:

$$\begin{aligned}s_i &= g_y(E[y_{i-1}], s_{i-1}, c_i) = w^{D_y}_i \\ w^{D_y}_i &= s^{D_y}_i + w^{D_y-1}_i \\ s^{D_y}_i &= g^{D_y}_y(w^{D_y-1}_i, s^{D_y}_{i-1}) \\ w^1_i &= s^1_i = g^1_{att}(E[y_{i-1}], s^1_{i-1}, c_i) \\ s^l_i &= g^l_y(w^{l-1}_i, s^l_{i-1}), \text{for } 1 < l \leq D_y \\ w^l_i &= s^l_i + w^{l-1}_i, \text{for } 1 < l \leq D_y\end{aligned}$$

The first RNN stack layer, $s^1_i = g^1_{att}(E[y_{i-1}], s^1_{i-1}, c_i)$, is a composition of two recurrent unit functions like in Equa-

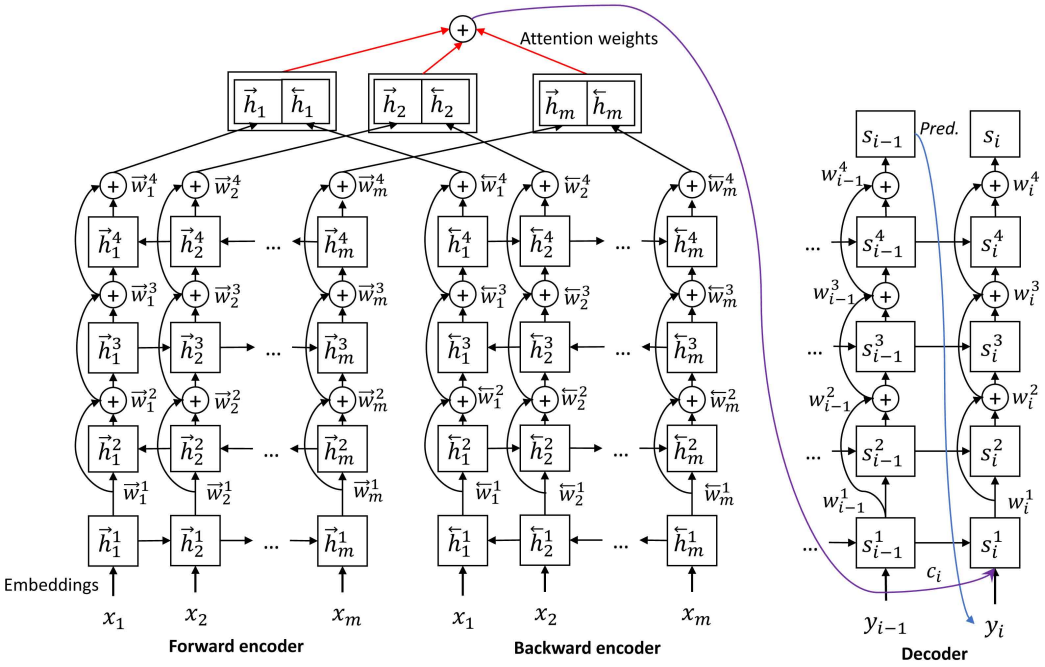


Figure 1: An illustration of a deep stacked RNN model (Zhou et al., 2016) with encoder stack depth (D_x) of 4 and decoder stack depth (D_y) of 4.

tions 8 and 9, i.e.,

$$
\begin{aligned}
s^1_i &= g^1_{att}(E[y_{i-1}], s^1_{i-1}, c_i) \\
s'_i &= g^1_{y,1}(E[y_{i-1}], s^1_{i-1}) \\
s^1_i &= g^1_{y,2}(c_i, s'_i)
\end{aligned}
$$

Like the encoders, the decoder recurrent unit function g^l_y at each layer l can be instantiated by LSTM or GRU.

2.3.2. Deep Transition RNN

The deep transition RNN (Miceli-Barone et al., 2017) involves a number of layers within a time step j through which an input word is fed, as illustrated in Figure 2. The recurrent unit function of each layer l is defined as a transition, which outputs an intermediate state $\overrightarrow{h}_{j,l}$ for the encoder and $s_{j,l}$ for the decoder. With L_x transitions, the hidden state representation at j is equivalent to the output of the last layer, so for the forward encoder state, Equation 11 defines $\overrightarrow{h}_j$ as:

$$
\begin{aligned}
\overrightarrow{h}_j &= \overrightarrow{g}_x(E[x_j], \overrightarrow{h}_{j-1}) = \overrightarrow{h}_{j,L_x} \\
\overrightarrow{h}_{j,1} &= \overrightarrow{g}_{x,1}(E[x_j], \overrightarrow{h}_{j-1,L_x}) \\
\overrightarrow{h}_{j,l} &= \overrightarrow{g}_{x,l}(0, \overrightarrow{h}_{j,l-1}), \text{for } 1 < l \leq L_x
\end{aligned}
$$

where the input to the first recurrent unit transition is $E[x_j]$, the embedding of the input word at j, while the subsequent higher level transitions only receive the output from the previous transition and do not receive the input word. The reverse hidden state $\overleftarrow{h}_j$ in Equation 12 is computed similarly by substituting $j-1$ with $j+1$. The encoder recurrent unit function $\overrightarrow{g}_{x,l}$ at each layer l can be instantiated by LSTM or GRU.

While the baseline shallow decoder RNN already contains two transitions, without and with attention respectively, the deep transition decoder RNN is extended similarly to the deep transition encoder RNN, such that the decoder hidden state with depth L_y is computed as:

$$
\begin{aligned}
s_{i,1} = s'_i &= g_{y,1}(E[y_{i-1}], s_{i-1,L_y}) \\
s_{i,2} &= g_{y,2}(c_i, s_{i,1}) \\
s_{i,l} &= g_{y,l}(0, s_{i,l-1}), \text{for } 2 < l \leq L_y \\
s_i &= s_{i,L_y}
\end{aligned}
$$

Similarly, the decoder recurrent unit function $g_{y,l}$ at each layer l can be instantiated by LSTM or GRU.

3. Experimental Setup

3.1. Datasets

We conducted experiments using the parallel training corpora from LDC to test on the NIST test sets. In addition, we also conducted experiments on the United Nations Parallel Corpus (Ziemski et al., 2016), following (Junczys-Dowmunt et al., 2016). We used the pre-defined training, development, and test sets of the corpus following (Junczys-Dowmunt et al., 2016) and conducted NMT experiments accordingly.

We pre-processed our parallel training corpora by segmenting Chinese sentences, which originally have no spaces to demarcate words, and tokenizing English sentences to split punctuation symbols from words. Chinese word segmentation was performed by a maximum entropy model (Low et al., 2005) trained on the Chinese Penn Treebank (CTB) segmentation standard.

To alleviate the effect of rare words in NMT, we fragmented words to sub-words through the byte pair encoding (BPE) algorithm (Sennrich et al., 2016) with 59,500 merge operations. All our training sentences are lowercased.

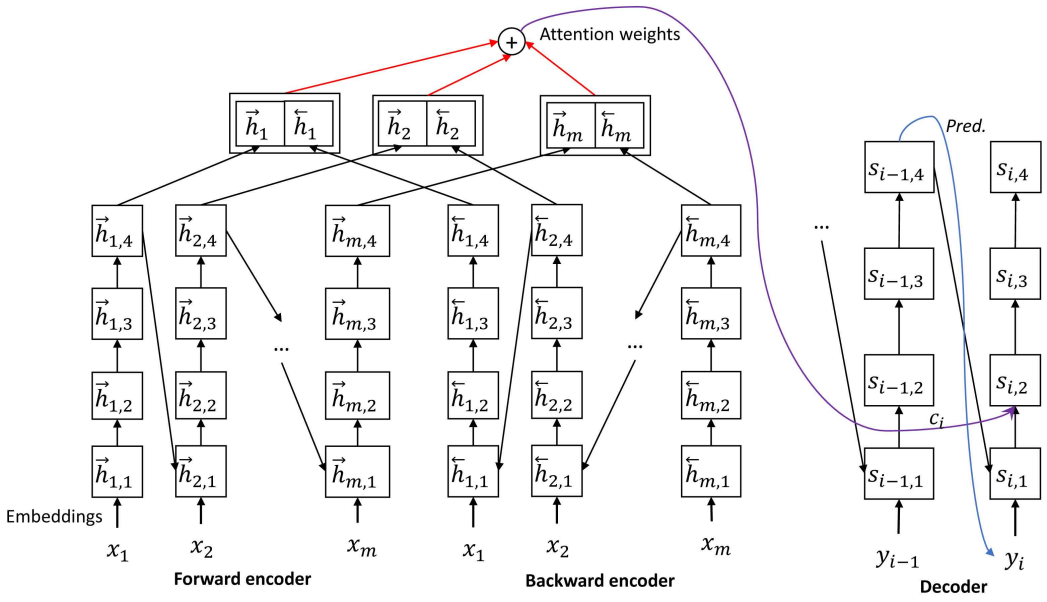


Figure 2: An illustration of a deep transition RNN model (Miceli-Barone et al., 2017) with 4 encoder transitions ($L_x = 4$) and 4 decoder transitions ($L_y = 4$).

3.1.1. LDC Corpora

We divide the LDC corpora we used into older corpora[3] and newer corpora[4]. Due to the dominant older corpora, we duplicate the newer corpora of various domains ten times to achieve better domain balance.

In addition to the parallel sentences, we also utilized a large amount of monolingual English texts, consisting of the English side of FBIS parallel corpus (LDC2003E14) and all the sub-corpora of the English Gigaword Fourth Edition (LDC2009T07). Altogether, the combined corpus consists of 107M sentences and 3.8B tokens. Each individual Gigaword sub-corpus (i.e., AFP, APW, CNA, LTW, NYT, and Xinhua) is used to train a separate N-gram language model. The English side of FBIS is also used to train another separate language model (LM). These individual language models are then interpolated to build one single large LM, via perplexity tuning on the English side of the development data. We use this LM for translation output re-ranking.

To recover the original casing on the translation output, we trained a statistical MT recaser model by using Moses (Koehn et al., 2007) on the English side of FBIS parallel text and the Xinhua portion of English Gigaword Fourth Edition.

Due to computation time and memory consideration, parallel sentences in the corpora that are longer than 50 sub-words in either Chinese or English are discarded. In the end, the final parallel training examples consist of 7.65M sentence pairs, 169M Chinese sub-word tokens (equivalent to 166M word tokens), and 186M English sub-word tokens (equivalent to 184M word tokens).

Our translation development (tuning) set is MTC corpus version 1 (LDC2002T01) and version 3 (LDC2004T07). This development set has 1,928 sentence pairs in total, 49K Chinese word tokens and 58K English word tokens on average across the four reference translations. Our translation test set consists of the NIST MT evaluation sets from 2002 to 2006, and 2008[5]. Altogether in the test sets, there are 7,497 sentence pairs, 192K Chinese word tokens, and 237K English word tokens on average across the four reference translations.

3.1.2. UN Parallel Corpus

The training set of the UN Parallel Corpus, after preprocessing and filtering those exceeding 50 sub-words, consists of 9.73M parallel sentence pairs, 207M Chinese sub-word tokens (equivalent to 204M word tokens), and 225 English sub-word tokens (equivalent to 223M word tokens). We also utilized larger English monolingual text, i.e., all the English side of the UN Parallel Corpus before length filtering, consisting of 11.3M sentences and 335M word tokens. The development set of the UN Parallel Corpus contains 4,000 sentence pairs, 107K Chinese word tokens, and 118K English word tokens, while the test set contains 4,000 sentence pairs, 106K Chinese word tokens, and 118K English word tokens. There is only one reference translation in the development and test sets of the UN Parallel Corpus.

3.2. NMT Model Parameters

We built our neural machine translation (NMT) system by using Nematus (Sennrich et al., 2017b), an open-source NMT toolkit which implements the encoder-decoder NMT architecture with attention mechanism. Our system is based on the NMT system in (Sennrich et al., 2017a). We built an

[3]LDC2002E18, LDC2003E14, LDC2004E12, LDC2004T08, LDC2005T06, and LDC2005T10.

[4]LDC2007T23, LDC2008T06, LDC2008T08, LDC2008T18, LDC2009T02, LDC2009T06, LDC2009T15, LDC2010T03, LDC2013T11, LDC2013T16, LDC2014T04, LDC2014T11, LDC2014T15, LDC2014T20, and LDC2014T26.

[5]LDC2010T10, LDC2010T11, LDC2010T12, LDC2010T14, LDC2010T17, and LDC2010T21.

ensemble model consisting of 4 independent models, which are the cross product of two different deep RNN architectures, i.e., deep stacked RNN and deep transition RNN, and two different recurrent unit functions, i.e., GRU and LSTM. For all our models, the word embedding dimension is 500, and the hidden layer dimension is 1,024. Our deep stacked RNN contains 4 stack layers on each of the encoder and decoder. Meanwhile, our deep transition RNN contains 4 encoder transitions and 8 decoder transitions.

Training for each individual model progresses by updating the model parameters at each mini-batch of 40 sentence pairs to minimize the negative log-likelihood loss function on the parallel training data. We use the Adam algorithm (Kingma and Ba, 2015) with learning rate of 0.0001. At each update, we clip the gradient norm to 1.0. We apply layer normalization (Ba et al., 2016) on the model parameters for faster convergence and tie the target-side embedding with the transpose of the output weight matrix (Press and Wolf, 2017). Model parameters are saved at every checkpoint of 10,000 update iterations. At this stage, the negative log-likelihood loss function on the development set is checked. Training stops when there has been no improvement over the lowest loss function value on the development set for 10 consecutive checkpoints.

The main difference between our system and (Sennrich et al., 2017a) is that while they only built NMT models with GRU, we also made use of LSTM. Another difference is in the usage of the larger monolingual English text. They built a synthetic Chinese-English parallel corpus by translating the monolingual English text to Chinese with an English-to-Chinese (reverse direction) NMT model and appended those sentence pairs to their parallel training corpus. Meanwhile, we exploited the English monolingual corpus by building a 5-gram language model to re-rank the k-best translation outputs produced by our NMT system.

4. Experimental Results

For experiments using the LDC corpora, translation quality is measured by case-insensitive BLEU (Papineni et al., 2002), for which the brevity penalty is computed based on the shortest reference (NIST-BLEU)[6]. Statistical significance testing between systems is conducted by bootstrap resampling (Koehn, 2004).

As shown in Table 1, among the individual model types, the deep stacked LSTM NMT model gives the best performance. However the best result is achieved by the ensemble of 4 independent model types, combining deep stacked and deep transition architectures with GRU and LSTM recurrent unit functions. This ensemble model gives an improvement of 4.40 BLEU points over the best deep stacked LSTM model. Applying re-ranking by the N-gram language model on top of the ensemble system of 4 independent models gives a further improvement of 0.11 BLEU point on average, which gives the best result of our system.

We are interested in comparing our 4-model ensemble before re-ranking with the ensemble of 2 models with only one RNN unit function (but two different deep layers) and with only one deep layer (but two different RNN units). As shown in Table 2, our 4-model ensemble outperforms every 2-model ensemble, and the improvement is statistically significant ($p < 0.01$).

We also compare the best results of our system on NIST test sets with the best results reported in published papers on Chinese-to-English MT systems in major computational linguistics and artificial intelligence publication venues[7]. Over the years 2007–2017 (both years inclusive), our 4-model ensemble system with re-ranking achieves a higher BLEU score than the best results reported in almost all (402 out of 403) papers[8].

Note that the LDC training datasets used in the published papers that we compare to may not be the same as ours. This incomparability would not have happened if there were a widely adopted, standardized dataset for training.

In addition, since there are two ways of computing BLEU scores with respect to word casing, i.e., case-insensitive and case-sensitive, we have taken care to compare BLEU scores using the same word casing, that is, by comparing our case-sensitive BLEU scores only to case-sensitive BLEU scores published previously, and similarly for case-insensitive BLEU scores.

Moreover, there are two ways of computation with respect to brevity penalty calculation involving multiple reference translations, namely "shortest", used in NIST-BLEU, and "closest", used in IBM-BLEU. The former sets brevity penalty against the shortest reference translation while the latter sets brevity penalty against the reference translation whose length is the most similar to the system translation output. We have also taken this into account by comparing NIST-BLEU with NIST-BLEU, and IBM-BLEU with IBM-BLEU[9].

For UN Parallel Corpus experiments, translation quality is measured by case-insensitive BLEU using the script provided by Moses[10], following the evaluation setup in (Junczys-Dowmunt et al., 2016)[11].

As shown in Table 3, our best result is obtained by an ensemble of 4 independent models with k-best output re-ranking using N-gram LM trained on the whole English side of the UN Parallel Corpus. Our system with re-ranking is 0.3 BLEU point better than without re-ranking, and the improvement is statistically significant ($p < 0.05$). Both of our systems achieve higher BLEU scores than the best published result for Chinese-to-English translation reported in (Junczys-Dowmunt et al., 2016).

[6] `ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v11b.pl`

[7] CL journal, TACL, ACL, COLING, EMNLP, NAACL, SSST, WMT, AAAI, and IJCAI.

[8] In one paper out of 403 papers (Huang et al., 2013), testing was performed on 4 test subsets: the news subset and the web subset of NIST06 and NIST08. Our system is only marginally worse (by less than 0.1 BLEU point) on the news subset of NIST06, and is better by a large margin (by 2–6 BLEU points) on the other 3 subsets.

[9] `ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a.pl`

[10] The `multi-bleu.perl` script in the Moses distribution.

[11] Personal communication.

Dataset	GRU		LSTM		4mod-ens	
	dstack	dtrans	dstack	dtrans	no re-ranking	with re-ranking
NIST02	41.27	43.28	44.03	42.12	46.82**	46.94
NIST03	41.78	42.38	42.49	41.87	47.42**	47.58
NIST04	43.75	44.33	45.11	43.97	49.12**	49.13
NIST05	41.71	42.52	43.40	41.97	47.72**	47.78
NIST06	42.27	43.18	43.43	42.19	49.19**	49.37
NIST08	35.28	36.11	36.78	35.55	41.36**	41.48
Average	41.01	41.97	**42.54**	41.28	**46.94****	**47.05**

Table 1: Experimental results in BLEU (%) of our NMT systems on NIST data set from LDC. Each individual model is obtained by cross-combining two different deep RNN architectures, i.e., deep stacked (**dstack**) and deep transition (**dtrans**) RNN, with two different recurrent unit functions, i.e., **GRU** and **LSTM**, without k-best re-ranking. The ensemble of 4 model types (**4mod-ens**) is obtained by taking the best model from each individual model type. This setting is tested both without and with re-ranking. Statistical significance testing was done to compare **4mod-ens** with the best individual model type, **dstack-LSTM** ($**$: significant at $p < 0.01$).

Layer / Unit	dstack	dtrans	dstack +dtrans
GRU	41.01	41.97	45.15
LSTM	42.54	41.28	45.34
GRU+LSTM	44.72	45.02	**46.94**$^{**\dagger\dagger\ddagger\ddagger\#\#}$

Table 2: Experimental results in BLEU (%) on NIST data set showing different combinations of RNN **unit** functions and deep **layers**. The RNN unit functions include GRU, LSTM, and the ensemble of the two. The deep layers include deep stacked (**dstack**) and deep transition (**dtrans**) RNN, and the ensemble of the two. Statistical significance testing is shown to compare our 4-model ensemble with the ensemble of **dstack+dtrans GRU** ($**$: significant at $p < 0.01$), with **dstack+dtrans LSTM** ($\dagger\dagger$: significant at $p < 0.01$), with **dstack GRU+LSTM** ($\ddagger\ddagger$: significant at $p < 0.01$), and with **dtrans GRU+LSTM** ($\#\#$: significant at $p < 0.01$).

Published	Ours (4mod-ens)	
	no reranking	with reranking
53.1	55.0	55.3*

Table 3: Experimental results in BLEU (%) on the test set of the UN Parallel Corpus of the best published result in (Junczys-Dowmunt et al., 2016) and our system, without and with k-best re-ranking with N-gram language model. Statistical significance testing shows the comparison between our 4-model system with re-ranking and without re-ranking ($*$: significant at $p < 0.05$).

5. Related Work

Establishing standards for the state of the art by publicly accessible resources is important in research. In speech recognition, for instance, there has been work on building a virtual machine as a means of collaboratively building a state-of-the-art system for speech recognition (Metze et al., 2013), aiming at realizing a standardized state-of-the-art system in a collaborative manner. While we do not provide any virtual machines, we have a similar intention of making available a state-of-the-art MT system.

On NMT, Denkowski and Neubig (2017) argued that experiments should be based on a strong baseline system to ensure that a newly proposed approach indeed improves over the best prior published approaches. They performed their experiments on WMT and IWSLT tasks (but not on Chinese-to-English translation) which have fixed training, development, and test sets. The problem for Chinese-to-English MT is greater in that there is no pre-defined set of training data that must be used for experiments, and various groups used different tuning sets and reported their results on different NIST OpenMT test sets. In addition, the lack of a standardized benchmark is not limited to neural MT approaches, but has been widespread since statistical MT approaches began to be tested on Chinese-to-English translation.

6. Conclusion

The problem of lack of consistent experimental setups for Chinese-to-English MT poses a challenge in evaluating newly proposed approaches over the pre-existing state of the art. This can be avoided if there is a clear benchmark which consists of standardized training, development, and test sets, as well as a common benchmark system setup. In this paper, we have shown that our proposed MT approach can be used to build a competitive system on the NIST OpenMT test sets that outperforms systems in almost all 403 published papers in the past 11 years.

We encourage Chinese-to-English MT experiments to use our common benchmark consisting of standard data and evaluation. As the NIST dataset from LDC is widely used in the Chinese-to-English MT research community, we have put up a scoreboard listing the scores achieved by all prior published papers when evaluated on the NIST dataset and released the source code and translation output of our best NMT system[12]. By doing so, we hope future work can make more meaningful comparisons to previous MT research.

[12] https://github.com/nusnlp/c2e-mt-benchmark

7. Bibliographical References

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *CoRR*, abs/1607.06450.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734.

Denkowski, M. and Neubig, G. (2017). Stronger baselines for trustable results in neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 18–27.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Huang, Z., Devlin, J., and Zbib, R. (2013). Factored soft source syntactic constraints for hierarchical machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 556–566.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is neural machine translation ready for deployment? A case study on 30 translation directions. In *Proceedings of the 13th International Workshop on Spoken Language Translation*.

Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL 2007 Demo and Poster Sessions*, pages 177–180.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395.

Low, J. K., Ng, H. T., and Guo, W. (2005). A maximum entropy approach to Chinese word segmentation. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, pages 161–164.

Metze, F., Fosler-Lussier, E., and Bates, R. (2013). The speech recognition virtual kitchen. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association (Interspeech-2013)*, pages 1858–1860.

Miceli-Barone, A. V., Helcl, J., Sennrich, R., Haddow, B., and Birch, A. (2017). Deep architectures for neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 99–107.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Press, O. and Wolf, L. (2017). Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Short Papers*, pages 157–163.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Miceli-Barone, A. V., and Williams, P. (2017a). The University of Edinburgh's neural MT systems for WMT17. In *Proceedings of the Conference on Machine Translation: Shared Task Papers*, pages 389–399.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli-Barone, A. V., Mokry, J., and Nădejde, M. (2017b). Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68.

Zhou, J., Cao, Y., Wang, X., Li, P., and Xu, W. (2016). Deep recurrent models with fast-forward connections for neural machine translation. *Transactions of the Association for Computational Linguistics*, 4(1):371–383.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The United Nations Parallel Corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

eSCAPE:
a Large-scale *S*ynthetic *C*orpus for *A*utomatic *P*ost-*E*diting

Matteo Negri[(1)], **Marco Turchi**[(1)], **Rajen Chatterjee**[(1,2)], **Nicola Bertoldi**[(1)]
[(1)]Fondazione Bruno Kessler, [(2)]University of Trento
Trento, Italy
{negri, turchi, chatterjee, bertoldi}@fbk.eu

Abstract

Training models for the automatic correction of machine-translated text usually relies on data consisting of (*source*, *MT*, *human_post-edit*) triplets providing, for each source sentence, examples of translation errors with the corresponding corrections made by a human post-editor. Ideally, a large amount of data of this kind should allow the model to learn reliable correction patterns and effectively apply them at test stage on unseen (*source*, *MT*) pairs. In practice, however, their limited availability calls for solutions that also integrate in the training process other sources of knowledge. Along this direction, state-of-the-art results have been recently achieved by systems that, in addition to a limited amount of available training data, exploit artificial corpora that approximate elements of the "gold" training instances with automatic translations. Following this idea, we present eSCAPE, the largest freely-available *S*ynthetic *C*orpus for *A*utomatic *P*ost-*E*diting released so far. eSCAPE consists of millions of entries in which the *MT* element of the training triplets has been obtained by translating the *source* side of publicly-available parallel corpora, and using the *target* side as an artificial human post-edit. Translations are obtained both with phrase-based and neural models. For each MT paradigm, eSCAPE contains 7.2 million triplets for English–German and 3.3 millions for English–Italian, resulting in a total of 14,4 and 6,6 million instances respectively. The usefulness of eSCAPE is proved through experiments in a general-domain scenario, the most challenging one for automatic post-editing. For both language directions, the models trained on our artificial data always improve MT quality with statistically significant gains. The current version of eSCAPE can be freely downloaded from: `http://hltshare.fbk.eu/QT21/eSCAPE.html`.

Keywords: Automatic Post-editing, Machine Translation

1. Introduction

Automatic post-editing (APE) for machine translation (MT) aims to fix recurrent errors made by the MT decoder by learning from correction examples. As a post-processing step, APE has several possible applications, especially in *black-box* scenarios (*e.g.* when working with a third-party translation engine) in which the MT system is used "as is" and is not directly accessible for retraining or for more radical internal modifications. In such scenarios, as pointed out by Chatterjee et al. (2015), APE systems can help to: *i)* improve MT output by exploiting information unavailable to the decoder, or by performing a deeper text analysis that is too expensive at the decoding stage; *ii)* provide professional translators with improved MT output quality to reduce (human) post-editing effort, and *iii)* adapt the output of a general-purpose MT system to the lexicon/style requested in a specific application domain.

The training of APE systems usually relies on data sets comprising (*source*, *MT*, *human_post-edit*) triplets, in which the *source* sentence in a given language has been automatically translated to produce the *MT* element that, in turn, has been manually corrected to produce the *human_post-edit*. In this supervised learning setting, the goal is to learn from the training data (and possibly generalise) the appropriate corrections of systematic errors made by the MT system, and apply them at test stage on unseen (*source*, *MT*) pairs. BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) computed against reference human post-edits are the standard evaluation metrics for the task. Their respective improvements and reductions are usually compared against the baseline scores obtained by the original MT output that has been left untouched (*i.e.* raw, non post-edited translations).

Early works on this problem date back to (Allen and Hogan, 2000; Simard et al., 2007), which addressed the problem as a "monolingual translation" task in which raw MT output in the target language has to be translated, in the same language, into a fluent and adequate translation of the original source text. Although the general monolingual translation approach to the problem is still the same, over the years the proposed solutions evolved in several ways, first by refining the decoding approach and then, in the last couple of years, by radically changing the core APE technology.

Decoding refinements successfully explored, for instance, the integration of *source* information for enhanced (joint, context-aware) input representation, either in the standard phrase-based MT (PBMT) framework (Béchara et al., 2011) or in more elegant batch factored models(Chatterjee et al., 2016) and online PBMT models (Chatterjee et al., 2017b). More recently, radical paradigm changes followed the "neural revolution" witnessed in the MT field. The current state of the art is indeed represented by single/multi-source neural APE systems, the former relying on the log-linear combination of monolingual and bilingual models (Junczys-Dowmunt and Grundkiewicz, 2016), and the latter learning from source and target information in a joint fashion (Chatterjee et al., 2017a). Recent works addressed the problem by also integrating external information such as word-level quality estimation scores (Chatterjee et al., 2017c) as a way to guide neural APE decoding towards better corrections.

Unsurprisingly, the impressive gains achieved by the neu-

ral solutions come at the cost of a much higher data demand compared to the PBMT methods. To overcome this problem, the latest published results on neural APE have been obtained by exploiting synthetically-created data during training (Junczys-Dowmunt and Grundkiewicz, 2016; Junczys-Dowmunt and Grundkiewicz, 2017; Variš and Bojar, 2017; Hokamp, 2017; Chatterjee et al., 2017a).

These trends, which emerged after three rounds of the APE task organised within the Conference on Machine Translation (WMT) (Bojar et al., 2015; Bojar et al., 2016; Bojar et al., 2017), clearly indicate that: *i)* information from the *source* text is definitely useful to train reliable APE models, and *ii)* the limited availability of "gold" training corpora made of (*source*, *MT*, *human_post-edit*) triplets calls for workarounds to unleash the full potential of state-of-the art but data-demanding neural systems.

The eSCAPE corpus presented in this paper meets such demand by providing APE research with a large-scale synthetically-created data set consisting of millions of triplets for two language pairs: English–German and English–Italian. Starting from a collection of publicly-available parallel corpora, it was built by automatically translating the *source* element of each sentence pair both with phrase-based an neural MT models, and using the original *MT* element as representative of a possible human correction.

This paper reports on the initial part of a roadmap aiming at more ambitious objectives. Future releases of the corpus will indeed include larger volumes of instances (translated with both MT paradigms) covering a larger spectrum of language combinations. The following sections provide an overview of the existing resources (Section 2), a description of eSCAPE (Section 3) and a discussion of experiments with the corpus (Section 4).

2. Related Work: Existing APE Corpora

The growing interest towards APE has to confront with the hard truth of data scarcity. Although nowadays post-edited data are a clear by-product of industrial translation workflows, the largest part of the daily work done by professional translators focuses on proprietary or copyright data that cannot be released. Though present in the industrial sector (as confirmed by recent works coming from big players like SYSTRAN (Crego et al., 2016) or eBay (Mathur et al., 2017)), APE technology is still more a matter of in-house development rather than a framework motivating free data sharing.

The few existing corpora that are usable for APE research can be classified into two types: *i)* the aforementioned "gold" data sets made of (*source*, *MT*, *human_post-edit*) triplets, and *ii)* the synthetic ones, to which our eSCAPE corpus belongs, in which some elements of the triplets derive from automatic translation. The remainder of this section provides an inventory of the existing APE corpora. As also shown in Table 1, the global picture is quite fragmented, with domain-specific data sets covering different language pairs, containing different types of post-edits and, most importantly, usually featuring a relatively small size.

2.1. "Gold" corpora

The Autodesk Post-Editing Data corpus (Zhechev, 2012)[1] is one of such resources. It mainly covers the domain of software user manuals, with English sentences translated with Autodesk's in-house PBMT system into several languages (simplified and Traditional Chinese, Czech, French, German, Hungarian, Italian, Japanese, Korean, Polish, Brazilian Portuguese, Russian, Spanish) with between 30,000 and 410,000 segments per language. Post-edits are made by professional translators.

Part of the Autodesk corpus has been used by Chatterjee et al. (2015) to compare different APE techniques in a controlled setting. For six target languages (Czech, German, Spanish, French, Italian and Polish), this subset comprises around 16,000 (*source*, *MT*, *human_post-edit*) triplets that share the same English *source*. To ease the replicability of their experiments and the reuse of the selected triplets, the authors released the scripts used for data extraction.[2]

Another useful resource is described in (Potet et al., 2012).[3] It consists of 10,881 triplets in which a French source sentence taken from several news corpora is translated into English by a PBMT system. Post-edits were collected using Amazon Mechanical Turk following strict control reviewing procedures to guarantee correction quality.

Two smaller corpora are respectively described in (Specia et al., 2010) and (Specia, 2011). The former consists of 4,000 English sentences from Europarl (Koehn, 2005), which were translated into Spanish by a PBMT system and manually post-edited by professional translators. The latter, which covers the news domain, includes 2,525 French–English PBMT translations and 1,000 English–Spanish translations with professional post-edits.

Other useful data have been released by the organisers of the WMT APE task. The first round of the task (Bojar et al., 2015) presented participants with around 12,000 English–Spanish training data drawn from the news domain, with translations derived from a PBMT system. A peculiarity of this corpus is that post-edits were collected from a non-professional crowdsourced workforce, with possible drops in terms of reliability and consistency.[4]

The second round of the task (Bojar et al., 2016) presented participants with a corpus released within the EU project QT21,[5] the same used for the WMT'16 quality estimation task. It comprises 13,000 English–German training data drawn from the information technology (IT) domain, with source sentences translated by a PBMT system and post-edits collected from professional translators. The combination of domain specificity and higher post-editing quality resulted in significant gains over the baseline.

[1] https://autodesk.app.box.com/Autodesk-PostEditing

[2] https://bitbucket.org/turchmo/apeatfbk/src/master/papers/ACL2015/

[3] http://www-clips.imag.fr/geod/User/marion.potet/index.php

[4] This is a possible cause of the poor results achieved by participants: none of them was indeed able to beat the APE task baseline represented by a "*do-nothing*" system that leaves all the raw MT translations unmodified.

[5] http://www.qt21.eu/

Corpus	Type	Languages	Domain	Size	Post-edits
(Specia et al., 2010)	Gold	En-Es	LEGAL	4K	Professional
(Specia, 2011)	Gold	Fr-En/En-Es	NEWS	2.5K/1K	Professional
(Zhechev, 2012)	Gold	En-Ch/Cs/Ff/De/Hu/It/Ja/Ko/Pl/Br/Pt/Ru/Es	IT	30K-410K	Professional
(Potet et al., 2012)	Gold	Fr-En	NEWS	11K	Crowd
(Bojar et al., 2015)	Gold	En-Es	NEWS	12K	Crowd
(Bojar et al., 2016)	Gold	En-De	IT	13K	Professional
(Bojar et al., 2017)	Gold	En-De/De-En	IT/MEDICAL	13K/26K	Professional
(Junczys-Dowmunt and Grundkiewicz, 2016)	Artificial	En-De	IT	4.3M	-

Table 1: Inventory of existing APE corpora

The third round of the task (Bojar et al., 2017) focused on both English–German and German–English data (also in this case provided by the QT21 project (Specia et al., 2017) and shared with the WMT'17 quality estimation task). English–German training data are drawn from the IT domain and consist of around 13,000 triplets. German–English training data, instead, come from the medical domain and comprise around 26,000 triplets. In both cases, translations were produced by a customised PBMT system and post-edited by professional translators.

2.2. Synthetic corpora

The use of synthetic resources aims to overcome the aforementioned problem of "gold" data scarcity with approximate solutions. This can be done in different ways. Several previous works have shown the viability of mimicking the ideal scenario in which the training triplets include actual human post-edits of machine-translated text by learning, instead, from the weaker connection between the MT output and external references. Though with variable margins, (Oflazer and El-Kahlout, 2007; Béchara et al., 2011; Rubino et al., 2012) report translation quality improvements in the PBMT scenario with post-editing components trained on (*source*, *MT*, *reference*) triplets. To the best of our knowledge, though potentially useful to APE research, none of such previous works released reusable datasets.
When moving to the data-demanding neural framework, data scarcity becomes a major problem that definitely calls for the external support of artificial corpora that are orders of magnitude larger than the current training sets.

A widely used resource, described in (Junczys-Dowmunt and Grundkiewicz, 2016), was included in the training set of the winning (and almost all) submissions to the last two English–German rounds of the APE task at WMT (IT domain). It consists of 4.3 million instances created by first filtering a subset of IT-related sentences from the German Common Crawl corpus[6], and then by using two English–German and German–English PBMT systems trained on in-domain IT corpora for a round-trip translation of the selected sentences (De→En→De). The final triplets were created by using: *i)* the English translations as (artificial) *source* sentences, *ii)* the round-trip German translations as (artificial) uncorrected *MT* output, and *iii)* the original German sentences as (artificial) *post-edits*.

[6] commoncrawl.org

By construction, this artificial data set approximates the quality of gold corpora by trying to keep a weak connection between the "post-edits" and the MT output. Keeping such connection, however, comes at the cost of having two levels of potential noise in the data, namely the possible errors introduced by the German–English translation needed to produce the *source* element of each triplet, and those of the English–German translations performed to produce the *MT* output. The approach we adopted to create the eSCAPE corpus, instead, follows a different strategy. As described in the next section, we start from parallel data and perform one single automatic translation step to produce the *MT* element of our triplets. The connection between MT output and "post-edits" is hence weaker than in (Junczys-Dowmunt and Grundkiewicz, 2016) due to the fact that our "post-edits" are actually independent reference translations of the source sentences. However, the possible noise introduced by translation errors can only affect one element of our triplets. Analysing the trade-off between translation noise and *MT-post-edits* proximity is out of the scope of this work but it is definitely an interesting aspect for future investigations.

3. The eSCAPE Corpus

The eSCAPE corpus[7] consists in two datasets (En-De and En-It) made of (*source*, *MT*, *reference*) triplets, where the *MT* segment is obtained by translating the *source* both with phrase-based and neural MT models. Its creation started from parallel (*source*, *target*) data collected from the WEB by merging several corpora belonging to various domains. Table 2 lists all the corpora used, indicating their domain and size in terms of number of sentences. Since some data sets, such as PatTR and Common Crawl, are only available in one language pair (En-De), the total number of sentences is different between the two language directions (En-De is twice larger than En-It). Apart from PatTR[8] and Common Crawl, all the datasets are available in the OPUS repository[9]. Before building the translation systems and producing the *MT* segments, all the corpora reported in Table 2 have been concatenated and shuffled (to avoid blocks of sentences belonging to the same domain) removing duplicates. This resulted in 7,258,533

[7] http://hltshare.fbk.eu/QT21/eSCAPE.html

[8] http://www.cl.uni-heidelberg.de/statnlpgroup/pattr/

[9] http://opus.lingfil.uu.se

English–German and 3,357,371 English–Italian sentence pairs.

Corpus	Domain	En-De	En-It
Europarl v7	LEGAL	1,920,209	1,909,115
ECB	LEGAL	11,317	193,047
Common Crawl	MIXED	2,399,123	-
JRC Acquis	LEGAL	719,372	810,979
News Commentary v11	NEWS	242,770	40,009
Ted Talks	MIXED	143,836	181,874
EMEA	MEDIC.	1,108,752	1,081,134
PatTR in-domain	MEDIC.	1,848,303	-
Wikipedia Titles	MEDIC.	10,406	-
Gnome	IT	28,439	319,141
Ubuntu	IT	13,245	21,014
KDE4	IT	224,035	175,058
OpenOffice	IT	42,391	-
PHP	IT	39,707	35,538
TOTAL		8,853,762	4,128,128

Table 2: List of datasets merged in the eSCAPE corpus.

3.1. MT systems

Driven by the need of translating these large quantities of source segments, the ModernMT toolkit (Bertoldi et al., 2017) has been used as translation system to generate both phrase-based and neural outputs. ModernMT is a new open-source MT software that consolidates the current state-of-the-art MT technology into a single and easy-to-use product. The toolkit adapts to the context in real-time and is capable of learning from (and evolving through) interaction with users, with the final aim of increasing MT-output utility for the translator in a real professional environment.

To avoid the risk of translating source segments that are in the training set, the collected sentence pairs were split in *4* slices: *3* parts were used to train the MMT models and the remaining one was translated. In this cross-validation setting, one sentence pair has been processed once for each experiment, either in training or in test.

For **phrase-based MT**, ModernMT uses high-performance embedded databases to store parallel and monolingual language data and associated statistics. Instead of pre-computing phrase-based feature function scores, these are computed on the fly, at translation time, from raw statistics. This allows the MMT toolkit to significantly speed up the training and test processes, to easily scale to large quantities of data, and to adapt on-the-fly to new domains. Training and test of the phrase-based models were run in parallel on several CPUs for around one week. Final performance, computed on a subset of the data, is 36.76 BLEU points for English–German and 38.08 for English–Italian.

For **neural MT**, the toolkit builds on the extension of a generic neural MT system based on the Nematus toolkit (Sennrich et al., 2017)[10], implementing the encoder-decoder-attention model architecture by (Bahdanau et al., 2014). Such extension consists in an internal dynamic memory, storing external user translation memories (TMs). When ModernMT receives a translation query, it quickly analyses its context, recalls from its memory the most related translation examples, and instantly adapts its neural network to the query (Farajian et al., 2017). Training and test of the neural models were run on one GPU (NVIDIA Tesla K80) for around three weeks. Final performance is 38.17 BLEU points for English–German and 41.01 for English–Italian.

To give the possibility for experiments on domain-adaptation for APE, each eSCAPE triplet is associated to a label indicating the name of the corpus from which the original (*source*, *reference*) pair was extracted.

4. Experiments

To test the usefulness of the eSCAPE corpus, we run APE experiments for both the language pairs covered by the data set. En-De and En-It data were first tokenised and then split into dev (2,000 triplets), test (10,000) and training (the remaining instances). For the sake of comparison, we performed the same data splits for both the phrase-based and for the neural-based section of the corpus.

As APE system, we chose the best system at this year's round of the WMT APE shared task (Chatterjee et al., 2017a). It consists in a neural multi-source model, in which the source and the MT segment are encoded separately and then merged together by a feed-forward network layer. A shared dropout is applied to both source and MT encoders. In this multi-source architecture both the encoders are trained jointly.

In our experiments, the hyper-parameters of all the systems in both language directions were the same. The vocabulary was created by selecting 50,000 most frequent sub-words, following the BPE approach of Sennrich et al. (2016b). Word-embedding and GRU (gated recurrent unit) hidden-state sizes were both set to 1024. Network parameters were optimized with Adagrad (Duchi et al., 2011) with a learning rate of 0.01. Source and target dropout was set to 10%, whereas encoder and decoder hidden states, weighted source context, and embedding dropout was set to 20% (Sennrich et al., 2016a). The batch size was set to 100 samples, with a maximum sentence length of 50 sub-words.

During training, the performance of the APE system was monitored on the development set and, at the end of the training phase, the model with highest BLEU score was used to post-edit the test set. The results are reported in Table 3 where, for both the language pairs and for both phrase-based and neural-based artificial data, the performance of the APE systems is compared against the "*do-nothing*" APE baseline (i.e. a system that leaves all the raw MT output unmodified).

It is interesting to note that APE systems outperform the baselines in both language settings with statistically significant gains.[11] This holds true both when they are trained

[10] https://github.com/rsennrich/nematus

[11] Although we consider the measured gains as a good indicator of the usefulness of using eSCAPE for training APE models, a study involving human evaluation would allow us to draw definite conclusions. Such a costly study, however, falls out of the scope of this paper and is left for future work.

	En–De	En–It
Phrase-based MT		
Do-Nothing baseline	36.76	38.08
APE	**38.15**	**39.80**
Neural MT		
Do-Nothing baseline	38.17	41.01
APE	**39.21**	**42.15**

Table 3: Neural APE results (BLEU score improvements are statistically significant with $p < 0.05$ computed with paired bootstrap resampling (Koehn, 2004)).

and tested on artificial data built from phrase-based models (+1.39 on En–De, +1.72 on En–It), and when training and test are performed on artificial data derived from neural models (+1.04 on En–De, +1.14 on En–It).[12]

The observed gains vary for the two language pairs (with highest results on En–It) and depending on the type of data used. Concerning this latter aspect, the higher quality of neural MT output results in lower gains on both language settings. This confirms previous outcomes from the WMT APE task: the higher the baseline (i.e. the BLEU score of the raw MT output), the lower the number of correction patterns that can be learned from the training data and the possibility to leverage their applicability to test data (Bojar et al., 2017).

Differently from the most recent shared evaluation settings (*i.e.* WMT'16 and WMT'17), in which neural APE has been tested in narrow domains, our results indicate that APE systems trained on the eSCAPE corpus can be also effective in the more challenging mixed-domain condition, where the correction rules are sparse across different domains, hence difficult to be learned and generalized. Considering the negative outcomes of the WMT'15 pilot task, which proposed a challenging evaluation setting based on general news data in which none of the participants was able to beat the "*do-nothing*" baseline, this is an interesting finding that adds value to our resource.

The BLEU score improvements also confirm the findings of (Oflazer and El-Kahlout, 2007; Béchara et al., 2011; Rubino et al., 2012) and extend them to neural APE. In fact, they report translation quality improvements in the PBMT scenario with an APE trained on *source*, *MT*, and independent *reference*. This suggests that, despite the aforementioned weak connection between the MT output and the "post-edits" of our triplets, APE models can be effectively trained on the eSCAPE corpus.

5. Conclusion and Future Work

We presented the eSCAPE corpus, a large-scale *S*ynthetic *C*orpus for *A*utomatic *P*ost-*E*diting consisting of millions of (*source*, *MT*, *post-edit*) triplets created via machine translation. eSCAPE is designed to support the recent trends in automatic post-editing, which show a clear predominance of data-demanding neural approaches. To cope with such demand, the current version of the corpus contains millions of triplets for two language pairs: English–German (14.4 millions) and English–Italian (6.6 millions). For both language pairs, half of the artificial data is obtained via phrase-based translation, while the other half is produced by better performing neural MT models. Having the same source sentences translated with both paradigms aims to enable future comparisons in the application of APE technology on the two types of output. The size of the corpus (the largest of its kind) is expected to ease model training and yield further state of the art improvements. Our preliminary experiments on mixed-domain data confirm this expectation: though trained on artificially-created instances, APE models significantly outperform baseline results in both language directions, independently from the MT technology underlying the data generation process. The work reported in this paper is the initial step of a more ambitious roadmap aimed to extend the resource with more data covering a larger spectrum of domains and language combinations. The current version of eSCAPE can be freely downloaded from: `http://hltshare.fbk.eu/QT21/eSCAPE.html`.

[12]Though interesting, other settings in which the two sections of eSCAPE are either combined or alternatively used one for training and one for test fall out of the scope of this paper and are left for future investigation.

Acknowledgments

This work has been partially supported by the EC-funded H2020 projects QT21 (grant agreement no. 645452) and ModernMT (grant agreement no. 645487). The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K80 GPU used for this research.

6. Bibliographical References

Allen, J. and Hogan, C. (2000). Toward the Development of a Post Editing Module for Raw Machine Translation Output: A Controlled Language Perspective. In *Proceedings of the Third International Controlled Language Applications Workshop (CLAW-2000)*, pages 62–71, Seattle, Washington, April.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Béchara, H., Ma, Y., and van Genabith, J. (2011). Statistical Post-Editing for a Statistical MT System. In *Proceedings of the 13th Machine Translation Summit*, pages 308–315, Xiamen, China, September.

Bertoldi, N., Cattoni, R., Cettolo, M., Farajian, M., Federico, M., Caroselli, D., Mastrostefano, L., Rossi, A., Trombetti, M., Germann, U., et al. (2017). MMT: New Open Source MT for the Translation Industry. In *Proceedings of The 20th Annual Conference of the European Association for Machine Translation (EAMT)*, pages 86–91.

Bojar, O., Chatterjee, R., Federmann, C., Haddow, B., Huck, M., Hokamp, C., Koehn, P., Logacheva, V., Monz, C., Negri, M., Post, M., Scarton, C., Specia, L., and Turchi, M. (2015). Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal, September.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Jimeno Yepes, A., Koehn, P., Logacheva, V., Monz, C., Negri, M., Neveol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., Verspoor, K., and Zampieri, M. (2016). Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, August.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi, M. (2017). Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214, Copenhagen, Denmark, September.

Chatterjee, R., Weller, M., Negri, M., and Turchi, M. (2015). Exploring the Planet of the APEs: a Comparative Study of State-of-the-art Methods for MT Automatic Post-Editing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 156–161, July.

Chatterjee, R., C. de Souza, J. G., Negri, M., and Turchi, M. (2016). The FBK Participation in the WMT 2016 Automatic Post-editing Shared Task. In *Proceedings of the First Conference on Machine Translation*, pages 745–750, Berlin, Germany, August. Association for Computational Linguistics.

Chatterjee, R., Farajian, M. A., Negri, M., Turchi, M., Srivastava, A., and Pal, S. (2017a). Multi-source Neural Automatic Post-Editing: FBK's participation in the WMT 2017 APE shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 630–638, Copenhagen, Denmark, September.

Chatterjee, R., Gebremelak, G., Negri, M., and Turchi, M. (2017b). Online Automatic Post-editing for MT in a Multi-Domain Translation Environment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 525–535, Valencia, Spain, April.

Chatterjee, R., Negri, M., Turchi, M., Federico, M., Specia, L., and Blain, F. (2017c). Guiding Neural Machine Translation Decoding with External Knowledge. In *Proceedings of the Second Conference on Machine Translation*, pages 157–168, Copenhagen, Denmark, September.

Crego, J. M., Kim, J., Klein, G., Rebollo, A., Yang, K., Senellart, J., Akhanov, E., Brunelle, P., Coquard, A., Deng, Y., Enoue, S., Geiss, C., Johanson, J., Khalsa, A., Khiari, R., Ko, B., Kobus, C., Lorieux, J., Martins, L., Nguyen, D., Priori, A., Riccardi, T., Segal, N., Servan, C., Tiquet, C., Wang, B., Yang, J., Zhang, D., Zhou, J., and Zoldan, P. (2016). SYSTRAN's Pure Neural Machine Translation Systems. *arXiv preprint 1610.05540*.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, pages 2121–2159.

Farajian, M. A., Turchi, M., Negri, M., Bertoldi, N., and Federico, M. (2017). Neural vs. Phrase-Based Machine Translation in a Multi-Domain Scenario. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 280–284, Valencia, Spain, April.

Hokamp, C. (2017). Ensembling Factored Neural Machine Translation Models for Automatic Post-Editing and Quality Estimation. In *Proceedings of the Second Conference on Machine Translation*, pages 647–654, Copenhagen, Denmark, September.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2016). Log-linear Combinations of Monolingual and Bilingual Neural Machine Translation Models for Automatic Post-Editing. In *Proceedings of the First Conference on Machine Translation*, pages 751–758, Berlin, Germany, August. Association for Computational Linguistics.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2017). An Exploration of Neural Sequence-to-Sequence Architectures for Automatic Post-Editing. In *Proceedings of the The 8th International Joint Conference on Natural Language Processing*, Taipei, Taiwan, November.

Koehn, P. (2004). Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the Empirical Methods on Natural Language Processing (EMNLP 2004)*, pages 388–395, Barcelona, Spain.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the Tenth Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket, Thailand, September.

Mathur, P., Ueffing, N., and Leusch, G. (2017). Generating Titles for Millions of Browse Pages on an e-Commerce Site. In *Proceedings of the 10th International Conference on Natural Language Generation (INLG2017)*, pages 158–167, Santiago de Compostela, Spain, September.

Oflazer, K. and El-Kahlout, I. D. (2007). Exploring Different Representational Units in English-to-Turkish Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 25–32, Prague, Czech Republic.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania.

Potet, M., Esperança-Rodier, E., Besacier, L., and Blanchon, H. (2012). Collection of a Large Database of French-English SMT Output Corrections. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, pages 4043–4048, Istanbul, Turkey.

Rubino, R., Huet, S., Lefèvre, F., and Linarès, G. (2012). Statistical Post-Editing of Machine Translation for Domain Adaptation. In *Proceedings of the European Association for Machine Translation (EAMT)*, pages 221–228, Trento, Italy, May.

Sennrich, R., Haddow, B., and Birch, A. (2016a). Edinburgh Neural Machine Translation Systems for WMT 16. In *Proceedings of the First Conference on Machine Translation*, pages 371–376, Berlin, Germany, August.

Sennrich, R., Haddow, B., and Birch, A. (2016b). Neural Machine Translation of Rare Words with Subword Units.

In *Proceedings of the 54th Annual Meeting on Association for Computational Linguistics*, pages 1715–1725, Berlin, Germany.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017). Nematus: a Toolkit for Neural Machine Translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain, April.

Simard, M., Goutte, C., and Isabelle, P. (2007). Statistical Phrase-Based Post-Editing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 508–515, Rochester, New York, April.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, August.

Specia, L., Cancedda, N., and Dymetman, M. (2010). A Dataset for Assessing Machine Translation Evaluation Metrics. In *Seventh Conference on International Language Resources and Evaluation, LREC 2010*, pages 3375–3378, Valletta, Malta.

Specia, L., Harris, K., Blain, F., Burchardt, A., Macketanz, V., Skadina, I., Negri, M., and Turchi, M. (2017). Translation Quality and Productivity: A Study on Rich Morphology Languages. In *Proceedings of the 16th Machine Translation Summit*, pages 55–71, Nagoya, Japan, September.

Specia, L. (2011). Exploiting Objective Annotations for Measuring Translation Post-editing Effort. In *15th Conference of the European Association for Machine Translation*, EAMT, pages 73–80, Leuven, Belgium.

Variš, D. and Bojar, O. (2017). CUNI System for WMT17 Automatic Post-Editing Task. In *Proceedings of the Second Conference on Machine Translation*, pages 661–666, Copenhagen, Denmark, September.

Zhechev, V. (2012). Machine Translation Infrastructure and Post-Editing Performance at Autodesk. In *AMTA 2012 Workshop on Post-Editing Technology and Practice (WPTP 2012)*, pages 87–96, San Diego, CA, USA.

Evaluating Machine Translation Performance on Chinese Idioms with a Blacklist Method

Yutong Shao[1], **Rico Sennrich**[2], **Bonnie Webber**[2], **Federico Fancellu**[2]
[1]School of Electronic Engineering and Computer Science, Peking University
5 Yiheyuan Road, Haidian District, Beijing, China
sythello@pku.edu.cn

[2]School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh, United Kingdom,
rico.sennrich@ed.ac.uk, bonnie@inf.ed.ac.uk, s1260346@sms.ed.ac.uk

Abstract

Idiom translation is a challenging problem in machine translation because the meaning of idioms is non-compositional, and a literal (word-by-word) translation is likely to be wrong. In this paper, we focus on evaluating the quality of idiom translation of MT systems. We introduce a new evaluation method based on an idiom-specific *blacklist* of literal translations, based on the insight that the occurrence of any blacklisted words in the translation output indicates a likely translation error. We introduce a dataset, CIBB (Chinese Idioms Blacklists Bank), and perform an evaluation of a state-of-the-art Chinese→English neural MT system. Our evaluation confirms that a sizable number of idioms in our test set are mistranslated (46.1%), that literal translation error is a common error type, and that our blacklist method is effective at identifying literal translation errors.

Keywords: Chinese-English machine translation, evaluation, idiom translation, blacklist method, CIBB dataset

1. Introduction

Idioms are a special figure of speech that are non-compositional and non-literal, though occasionally share surface realizations with literal language uses (Salton et al., 2014b). Idioms are considered highly problematic for a wide variety of NLP tasks (Sag et al., 2002). This belief also holds true for machine translation, because MT systems often make the assumption that meaning is compositional, which is not true for idioms. The compositionality assumption leads to *literal translation errors*, the word-by-word translation of idioms, resulting in a translation that is confusing and not understandable. Therefore, idiom translation is a hard problem in MT and has attracted considerable research interest (Cap et al., 2015; Salton et al., 2014b; Anastasiou, 2010).

Given the difficulty of idiom translation in MT, it would be helpful to have a method to evaluate idiom translation performance. There is a wide range of methods for evaluating the performance of MT systems, but none of them are satisfactory for the targeted evaluation of idiom translation. The most straightforward method is human evaluation. While human evaluation is highly valuable, it is desirable to develop complementary automatic methods that are low-cost and fast, thus allowing for more rapid and frequent feedback cycles. Popular automatic MT metrics such as BLEU (Papineni et al., 2002) are inexpensive, but are unsuitable for a targeted evaluation.

This paper tries to fill this gap by presenting a method to assess the quality of idiom translations. We introduce a new method called "blacklist method" for performance evaluation on idioms, which is based on the intuition that a literal translation of the components of the idiom is likely to be wrong, and easy to spot by defining a blacklist of words that indicate a likely literal translation error.

We perform a case study on a special class of Chinese idioms that typically consist of 4 characters, called "cheng2 yu3". Actually, not all these 4-character words satisfy the definition of idioms. Some words are *semantically transparent*, which means they are compositional and can be translated literally. This kind of words are less problematic and less necessary to evaluate the systems' performance on. In this research we will only focus on those *semantically non-transparent* words, which have different literal meanings and idiomatic meanings. We will subsequently refer to them as "Chinese idioms".

We also introduce the CIBB dataset[1] for actually executing this evaluation on Chinese→English MT systems. Based on this dataset, we conduct experiments on a state-of-the-art NMT system. From the experiments we draw the following conclusions:

1. Idiom translation remains an open problem in Chinese→English NMT
2. Literal translation error is still a prevalent error type
3. The blacklist method is effective at detecting literal translation errors.

2. Related Work

2.1. Global Evaluation Metrics

Global evaluation metrics are metrics that evaluate the overall performance of MT systems and allow automatically calculation. There are many well-known global evaluation metrics, such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), TER (Snover et al., 2006), etc. However, these metrics only provide global evaluation and are unable to evaluate MT systems' performance on specific aspects. Therefore, they are unsatisfactory in evaluating idiom translation performance.

[1]This dataset is released at `https://github.com/sythello/CIBB-dataset`

2.2. Test Suite Methods

Test suite methods construct a set of sentences that focus on specific types of difficulties in MT. Typically, we design a set of sentences in the source language for the MT system to translate, and a scoring method to evaluate the translations. The sentence set and the scoring method are designed so that the score assigned to a system indicates the system's performance on the focused difficulty. This kind of methods makes up for the drawbacks of global evaluation metrics that they cannot assess a system's performance on specific issues.

There are many previous works belonging to this category. (Isabelle et al., 2017) proposed a *challenge set approach* to evaluate English→French MT systems' performance on divergence problems. The English sentences in the challenge set are chosen so that their closest French equivalent will be structurally divergent from them in some crucial way. (Burchardt et al., 2017) constructed a test suite for English→German MT systems. This test suite covers a wide variety of linguistic phenomena, such as ambiguity, composition, function words, multi-word expressions and so on. (Burlot and Yvon, 2017) introduced a new scheme to evaluate the performance of English→MRL (morphologically rich languages) MT systems on morphological difficulties. The test suite they built consists of three parts, focusing on a system's morphological adequacy (generating different morphological features in different contexts), fluency (word agreement) and certainty (generating the same morphological features in different contexts), respectively. Evaluation is based on automatic morphological analysis of the MT output. (Sennrich, 2017) proposed a method to construct the test suite automatically for evaluating English→German NMT systems on word agreement, polarity, transliteration, etc. The test suite is made up with minimal translation pairs, where a reference translation is paired with a contrastive translation which introduces a single translation error, allowing to measure the sensitivity of a neural MT (NMT) system towards this type of error. The score on the test suite is also obtained automatically by calculating the precision of the NMT system to assign a higher probability to the correct translation than to the contrastive translation in each translation pair. While this method allows for an automatic large-scale evaluation of specific errors, it only measures the probability of pre-defined translations, and is less suitable if the types of errors are relatively unpredictable. Even if we only focus on literal translation errors, it is hard to align the idiom with its translation in the reference (because the translation of idioms can be very flexible) and replace it with literal translation without destroying the coherence of the whole sentence.

Test suite methods can be divided into manual-construction methods and automatic-construction methods, based on whether the test suite construction is automatic. They can also be divided into manual-evaluation methods and automatic-evaluation methods, based on whether the scoring process is automatic. Combining the two classification criteria, we have 3 typical categories of test suite methods:

1. Automatic construction, automatic evaluation: large test suites and efficient evaluation.

2. Manual construction, automatic evaluation: small test suites but efficient evaluation.

3. Manual construction, manual evaluation: small test suites and laborious evaluation.

According to the classification criteria given above, (Isabelle et al., 2017) and (Burchardt et al., 2017) are manual construction, manual-evaluation test suite methods; (Burlot and Yvon, 2017) and (Sennrich, 2017) are automatic-construction, automatic-evaluation test suite methods.

2.3. Automatic Error Detection Methods

Automatic error detection methods complement global evaluation metrics in another way, by providing algorithms to detect specific kinds of errors in the translation automatically. Previously, there have been many valuable works on automatic error detection. (Zeman et al., 2011) introduced Addicter, which can detect many translation error types, such as missing word, untranslated word, extra word, form error, etc. It is based on the word alignment between the reference and the hypothesis. (Popovic, 2011) introduced Hjerson, which detects similar error types as Addicter, while it is based on the dynamic programming algorithm for calculating Word Error Rate (WER). However, both Addicter and Hjerson have some drawbacks in common. First, they work on a word-by-word basis, so the error types they can detect are rather restricted. Also, they do not match well enough with human annotators, implied by the experiment results in (Zeman et al., 2011).

2.4. Idiom Translation and Literal Translation

Idioms have long been considered as a hard problem for machine translation in many language pairs. Experiments in (Salton et al., 2014a) showed that on sentences containing idioms, a standard phrase-based English→Brazilian-Portuguese MT system achieves about half the BLEU score of the same system when applied to sentences that do not contain idioms. Among all the translation errors caused by idioms, *literal translation errors* are believed to be an important error type. (Manojlovic et al., 2017) demonstrated that literal translations predominate in the output of a phrase-based English↔Croatian MT system when translating sentences with idioms. According to our preliminary observations, literal translation errors also occur often in state-of-the-art Chinese→English NMT systems.

In order to improve the performance of idiom translation, (Carpuat and Diab, 2010) investigate two strategies: treating idioms and multiword expressions as an atomic unit, and adding a phrase-level feature that identifies multiword expressions. They find that both strategies improve the translation of non-compositional expressions. (Salton et al., 2014b) propose a substitution method that replaces idioms in the source sentence with their literal meaning before translation; after translation, the translation of the literal meaning is replaced with a target language idiom, if possible.

3. Blacklist Method

The "blacklist" method we are going to describe is used for detecting *literal translation errors*, which means the system translates an idiom word-by-word and thus gets a wrong translation, as described in section 1. According to related works and our own observations introduced in section 2.4., we hypothesize that literal translation errors represent a majority of idiom translation errors. We further hypothesize that we can easily identify literal translation errors by checking the translation for words that represent the meaning of a subsequence of the source idiom, but which should not appear in the true, idiomatic translation. These words make up the *blacklist* for the idiom, which we manually create. For the example in Table 1, if a machine translation system is fed with a Chinese sentence containing this idiom, and the system outputs a translation containing "bamboo" or "chest", then we say the translation *trigger the blacklist* and therefore will be judged as a literal translation error.

Idiom	胸有成竹
Idiomatic translation (correct)	Be very ready; have a well-thought-out plan
Literal translation (incorrect)	Have a well-formed bamboo in one's chest
Blacklist	**bamboo, chest**

Table 1: Example of blacklist.

Using the concept of blacklist, here we give the whole process of "blacklist method" evaluation:

1. Build an idiom list with idioms that we can build blacklist for. To be more specific, we choose idioms that contain one or more characters whose direct translations should not exist in translation of the whole idiom.
2. Build a blacklist for each idiom on the list. The blacklist consists of the direct translation of the characters mentioned in the last step.
3. Gather source language (Chinese) sentences containing idioms on the list. Note that the method itself does not need reference translations. Nevertheless, if someone is not a speaker of the source language but wishes to get some ideas about the detected literal translation errors, or to check whether the detection is correct, then using translation pairs is more desirable than monolingual sentences.
4. Feed all the sentences to the MT system to get the translations.
5. Calculate the percentage of translations triggering the blacklist, which is the evaluation score for the system.

We draw on an existing idiom list for step 1, and perform step 2 manually. Steps 1-3 form the construction procedure and only need to be conducted once; steps 4-5 form the evaluation procedure that needs to be conducted on different systems.

Advantages and Disadvantages According to the classification criteria introduced in section 2.2., our blacklist method is a manual-construction, automatic-evaluation test suite method. Therefore, the main advantage of the blacklist method is that, after creating the blacklist, large-scale evaluation is inexpensive and reproducible. The selection of proper idioms and the construction of a blacklist for each idiom is feasible by a bilingual speaker, and future work may even try to automate this. After the idiom list and blacklists are determined, we can scale up the set of translation pairs as much as we need, using online bilingual or even monolingual datasets. Also, we expect the blacklist method to achieve a high precision, because the definition of "blacklist" is actually closely related to literal translation errors. On the other hand, the drawback of this method is that the method is restricted to only one error type, literal translation errors, and will not detect any other type of errors such as deletions or repetitions of the idiom. Hence, recall is uncertain.

4. Dataset Construction

In CIBB, we provide a list of 50 Chinese idioms, each paired with an idiom-specific blacklist, and 1194 Chinese→English translation pairs, each containing an idiom on the list.

Idioms and Blacklists We downloaded about 30000 Chinese idioms from the following websites:

- http://www.gsdaquan.com
- http://chengyu.t086.com
- http://bcc.blcu.edu.cn

After excluding all the idioms that never appeared in the training data of our NMT system, there are about 9000 idioms left. Among these 9000 idioms, we observed some samples of them and selected 50 idioms with different frequencies in the training data. According to our observation, idioms with very high frequency in the training data are generally translated well, so we focus on lower-frequency idioms. Meanwhile, we cannot expect a system to learn to translate idioms with too low frequency. Therefore, we selected idioms appearing between 7 and 1000 times in the training data. We further select only idioms whose translation is non-compositional, and create a blacklist for each idiom.

Translation Pairs The translation pairs were extracted from OpenSubtitles2016 dataset (Lison and Tiedemann, 2016), where we searched for Chinese→English translation pairs with idioms on our list. In order to balance the frequency of all the idioms in the translation pairs, preventing the majority being taken up by only a few idioms, we restricted the maximum occurrences of any idiom to be 40. Under such restrictions, we extracted a total of 1194 translation pairs.

5. Experiments

The objective of our experiments is to evaluate the effectiveness of the blacklist method at detecting translation errors, especially the literal translation errors, by its precision, recall, as well as the correlation with BLEU. Also, we want to test to what extent idiom translation is a problem for a current state-of-the-art NMT system.

5.1. The MT System

As a representative of the current state of the art in NMT, we evaluate the Edinburgh NMT system for the WMT17 shared news translation task (Sennrich et al., 2017), which was ranked tied best for Chinese→English. The system is an attentional encoder-decoder, and its training data is constrained to the training data provided at WMT17, namely *News Commentary v12*, *UN Parallel Corpus V1.0*, the *CWMT Corpus*, and back-translated monolingual data from the *News Crawl Corpus*. On the Chinese side, the system uses Jieba[2] for word segmentation, and BPE for subword segmentation (Sennrich et al., 2016). More details about the model architecture can be found in the system description.

Word Segmentation Our MT system performs both word segmentation and subword segmentation on the Chinese texts. It is worth noting that different approaches of word segmentation may lead to different results in our test. The test method focuses on literal translation errors, which can only happen if an idiom is segmented into several parts, not if the idiom is unsegmented and treated as a single unit. Treating the idiom as a single unit may be an effective approach to prevent literal translation errors, but may increase vocabulary size and/or cause other types of errors that are not captured by the blacklist. Evaluating the effect of (sub)word segmentation on idiom translation remains the subject of future work.

5.2. Experiment Setup

We first translate all the 1194 Chinese source sentences into English using the Edinburgh WMT17 system introduced above. Then we apply blacklist method to all the translations. For those translations triggering the blacklist, we manually count the number of correct and incorrect translations, as well as the number of literal translation errors. For those not triggering the blacklist, we randomly sample and manually evaluate 100 translations to estimate error rates for this group. We only focus on errors with respect to idioms; errors of other aspects are ignored.

5.3. Experiment Results

Among all the 1194 translations, 145 triggered the blacklist and 1049 translations did not. We conducted manual evaluation on all the translation triggering the blacklist, and 100 random sampled translations not triggering the blacklist. The results are shown in Table 2.

First of all, the overall 46.1% (551/1194) error rate and 11.9% (142/1194) literal translation error rate implies that idiom translation is still problematic for a state-of-the-art MT system and literal translation is an important error type in idiom translation. Furthermore, among the 145 translations triggering the blacklist, 142 were incorrect literal translations; only 3 of them were actually correct ones but triggered the blacklist in some other ways (an example of this is provided in section 5.5.2.). For the translations not triggering the blacklist, according to our evaluation on 100 examples, 61 of them were correct and 39 were incorrect, while no literal translation errors was found. We thus estimated that for all the 1049 translations not-triggering the blacklist, 640 are correct translations while 409 are incorrect, and there is no literal translation error. This means that our blacklist method has a very high precision of 97.9% (142/145) and recall of 100% (142/142) of catching literal translation errors. If we regard the blacklist method as a method to detect wrong idiom translations of any type, the precision is unchanged, and we still have a recall of about 25.8% (142/551), which means the blacklists can catch a considerable amount of errors in all the translations. Among errors that the blacklist method does not identify, deletion errors are the most prevalent category.

	Correct	Incorrect	Incorrect Literal	Total
Not triggering	640*	409*	0*	1049
Triggering	3	142	142	145
Total	643	551	142	1194

Table 2: Results of our test on Edinburgh WMT17 system. Figures followed by (*) are estimated using 100 random samples out of 1049.

5.4. Idiom Translation and BLEU

We test the interaction of our evaluation method and BLEU We calculated the BLEU score for four different sets of translations: A random sample of 1000 sentences from OpenSubtitles2016, our CIBB test set of 1194 sentences containing an idiom, all translations triggering the blacklist and all translations not triggering the blacklist. The results are listed in Table 3. We can see that the BLEU score for translations of idioms is only about half the BLEU score of randomly sampled translations, in line with results from previous work (Salton et al., 2014a). This confirms our hypothesis that translating sentences with idioms is hard for state-of-the-art NMT systems. Also, the BLEU score of translations triggering the blacklist is lower than the translations not triggering the blacklist, indicating that the blacklist method is useful at identifying low-quality translations, even without a reference.

test set	BLEU
Random 1000 samples	11.85
With idioms	6.35
Blacklist triggered	5.64
Blacklist not triggered	6.44

Table 3: BLEU scores for different sets of translations.

[2] https://github.com/fxsjy/jieba

5.5. Examples

Here we provide some examples for different types of translations we discussed in section 4.2.

5.5.1. Correctly Detected Errors

Idiom	说三道四
Meaning	Gossip
Literal	Speak three and four
Blacklist	**three** four
SRC	医生说了你不能对我说三道四
REF	The therapist said you're not allowed to judge me.
TRANS	The doctor said that you can't say **three** things to me.

Table 4: Example for correctly detected errors.

In the example shown in Table 4, the word "three" is the literal translation of 三, but should not appear in the correct idiomatic translation Therefore, the occurrence of "three" in the translation triggers our blacklist, correctly indicating a literal translation error.

5.5.2. False Positives

Idiom	谈笑风生
Meaning	Talk cheerfully and humorously
Literal	Talking and laughing generate winds
Blacklist	**wind**
SRC	他们谈笑风生而我们却要在这里吹风
REF	Burke's up there, too laughing it up with the President while we're stuck down here.
TRANS	They *talk and laugh*, but we're going to blow the ***wind*** right here

Table 5: Example for false positives.

The example shown in Table 5 demonstrates a false positive. While the idiom is translated correctly into "talk and laugh", "wind" appears in another place of the source sentence, and that triggered the blacklist. Future work could involve further constraints, such as taking into account alignment information, to further reduce false positives.

5.5.3. Not Detected Errors

In this example shown in Table 6, the idiom meaning "full of energy" or "actively" is incorrectly translated into "have to". However, as this is not a literal translation error, our blacklist method is unable to catch it. This is a limitation of the blacklist method, which is only designed to capture literal translation errors.

Idiom	生龙活虎
Meaning	Full of energy
Literal	Lively dragon and tiger
Blacklist	dragon tiger
SRC	你明明生龙活虎到处走
REF	You were so actively walking around just then
TRANS	You **have to** go all over the place

Table 6: Example for not detected errors.

6. Conclusion and Future Work

We introduced the blacklist method for evaluating the performance of MT systems on idioms. This method works by automatically detecting literal translation errors and calculating the error rate. The results of our experiments have shown that the blacklist method is useful for detecting this kind of errors. The experiments also confirm that idiom translation remains an open problem for NMT systems. We introduced the dataset CIBB which is used for executing blacklist method evaluation on Chinese→English MT systems. The dataset contains 1194 Chinese→English translation pairs covering 50 Chinese idioms.
In the future, this work may be developed in following directions:

- Our current idiom list consists of 50 idioms, and we can further extend the idiom list and refine the blacklist to improve the performance of the blacklist evaluation method.
- An automatic identification of idioms, and automatic construction of the blacklist would facilitate the transfer of the evaluation method to other language pairs. We note that there is related work on automatic identification of non-compositional expressions that could enable this (Melamed, 1997).
- While a blacklist-based evaluation has shown high precision and recall at identifying literal translation errors, it is blind towards other error types, such as deletion errors. We note that related research has focused on the identification and prevention of deletion errors via measuring the ability of models to reconstruct the source sentence from the translation (Li and Jurafsky, 2016; Tu et al., 2017). We consider it interesting that reconstruction-based methods may be blind towards literal translation errors, which means that these two methods are complementary and could potentially be combined.

More broadly, a blacklist-based evaluation is attractive in that it can identify some types of translation errors without access to human reference translation. It could thus

prove beneficial for quality estimation in a post-editing environment. Finally, we hope that our evaluation results and dataset will spark future research on improving idiom translation in MT. We could revisit strategies from phrase-based MT, such as forcing idioms to be represented as an atomic unit (Carpuat and Diab, 2010), although this would have undesirable side effects in neural MT such as increasing the size of the network vocabulary.

7. Acknowledgements

This research is supported by the Summer Research Visitor Programme between the School of Electronic Engineering and Computer Science at Peking University and the Institute for Language, Cognition and Computation in the School of Informatics at the University of Edinburgh.

8. Bibliographical References

Anastasiou, D. (2010). *Idiom Treatment Experiments in Machine Translation*. Cambridge Scholars Publishing.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72.

Burchardt, A., Macketanz, V., Dehdari, J., Heigold, G., Peter, J.-T., and Williams, P. (2017). A Linguistic Evaluation of Rule-Based, Phrase-Based, and Neural MT Engines. *The Prague Bulletin of Mathematical Linguistics*, 108(1):159–170.

Burlot, F. and Yvon, F. (2017). Evaluating the morphological competence of Machine Translation Systems. In *WMT*, pages 43–55. Association for Computational Linguistics.

Cap, F., Nirmal, M., Weller, M., and Schulte im Walde, S. (2015). How to Account for Idiomatic German Support Verb Constructions in Statistical Machine Translation. In *MWE@NAACL-HLT*, pages 19–28. The Association for Computer Linguistics.

Carpuat, M. and Diab, M. (2010). Task-based Evaluation of Multiword Expressions: a Pilot Study in Statistical Machine Translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 242–245, Los Angeles, California, June. Association for Computational Linguistics.

Isabelle, P., Cherry, C., and Foster, G. (2017). A Challenge Set Approach to Evaluating Machine Translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2476–2486. Association for Computational Linguistics.

Li, J. and Jurafsky, D. (2016). Mutual Information and Diverse Decoding Improve Neural Machine Translation. *CoRR*, abs/1601.00372.

Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In *LREC*. European Language Resources Association (ELRA).

Manojlovic, M., Dajak, L., and Bakaric, M. B. (2017). Idioms in state-of-the-art Croatian-English and English-Croatian SMT systems. In *MIPRO*, pages 1546–1550. IEEE.

Melamed, I. D. (1997). Automatic Discovery of Non-Compositional Compounds in Parallel Data. In *EMNLP*. ACL.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). Bleu: a Method for Automatic Evaluation of Machine Translation. In *ACL*, pages 311–318. ACL.

Popovic, M. (2011). Hjerson: An Open Source Tool for Automatic Error Classification of Machine Translation Output. *Prague Bull. Math. Linguistics*, 96:59–68.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A. A., and Flickinger, D. (2002). Multiword Expressions: A Pain in the Neck for NLP. In *CICLing*, volume 2276 of *Lecture Notes in Computer Science*, pages 1–15. Springer.

Salton, G., Ross, R. J., and Kelleher, J. D. (2014a). An Empirical Study of the Impact of Idioms on Phrase Based Statistical Machine Translation of English to Brazilian-Portuguese. In *HyTra@EACL*, pages 36–41. The Association for Computer Linguistics.

Salton, G., Ross, R. J., and Kelleher, J. D. (2014b). Evaluation of a Substitution Method for Idiom Transformation in Statistical Machine Translation. In *MWE@EACL*, pages 38–42. The Association for Computer Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Barone, A. V. M., and Williams, P. (2017). The University of Edinburgh's Neural MT Systems for WMT17. In *WMT*, pages 389–399. Association for Computational Linguistics.

Sennrich, R. (2017). How Grammatical is Character-level Neural Machine Translation? Assessing MT Quality with Contrastive Translation Pairs. In *EACL (2)*, pages 376–382. Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200.

Tu, Z., Liu, Y., Shang, L., Liu, X., and Li, H. (2017). Neural Machine Translation with Reconstruction. In *AAAI*, pages 3097–3103. AAAI Press.

Zeman, D., Fishel, M., Berka, J., and Bojar, O. (2011). Addicter: What Is Wrong with My Translations? *Prague Bull. Math. Linguistics*, 96:79–88.

Appendix

idiom	translation	literal translation
手无寸铁	Unarmed	Have no iron in one's hand
雪上加霜	Rub salt into the wound; exacerbate	Frost form on the snow
背井离乡	(Be forced to) leave one's home	Leave the well and hometown
五花八门	Of a wide variety	Five flowers and eight gates
立竿见影	Have an immediate effect	Put up a stick and see the shadow
烟消云散	Disappear, vanish	Vanish like smoke and cloud
大刀阔斧	Bold, drastic, macroscopic, not consider much of the details	Big knife and axe
不速之客	Uninvited guest; unwelcome guest	Not invited (speed) guest
冷嘲热讽	Sarcasm; irony	Cold and hot sarcasm
迎刃而解	(Problem) be easily solved	Break on the knife blade
蛛丝马迹	Traces, clues	Spider silk and horce trace
亡羊补牢	Better late than never	Mend the pen after losing some sheep
说三道四	Gossip	Speak three and four
锦上添花	Embellish what is already beautiful	Add flowers to beautiful cloth
马马虎虎	Careless / Just so-so, passable	Like horses and tigers
胆战心惊	Be terror-stricken	Gall trembling and heart frightened
易如反掌	Very easy, a piece of cake	As easy as turning one's hand
开门见山	Come straight to the point/question	Open door and see the mountain
胸有成竹	Have a well-thought-out plan	Have a bamboo in one's chest
蠢蠢欲动	Be restless to do something; be ready to do something	Be restless to move like worms (stupid)
洗耳恭听	Be all ears; listen carefully	Wash one's ears to listen politely
五光十色	Colorful	Five lights and ten colors
九霄云外	Far, far away	Out of nine clouds
推心置腹	Sincerely; heart-to-heart	Push hearts and settle the stomach
谈笑风生	Talk cheerfully and humorously	Talking and laughing generate winds
凤毛麟角	Extremely rare	Pheonix fur and kylin horn
灰飞烟灭	Vanish; be destroyed (like ashes and smoke)	Ash(grey) fly and smoke vanish
星罗棋布	Spread all over the place	Stars spread and men deployed
望尘莫及	Too far behind to catch up	Only see the dust and cannot catch up
天马行空	In a powerful and unconstrained style	Sky horse traveling in the sky
呼之欲出	Vivid / Coming out soon	Call it and it will show up
抛砖引玉	Make some introductory remarks to set the ball rolling	Throw bricks to attract jades
添油加醋	Add highly coloured details; distort, exaggerate	Add oid and vinegar
守株待兔	Wait around aimlessly for a windfall that is unlikely to come	Wait by a tree for rabbits
板上钉钉	Be fixed; be clinched	Nail on the board
顺手牵羊	Walk off with sth.; steal sth. when walking by	Take away a sheep when walking by
呆若木鸡	Be dumb-struck (as a wooden chicken)	Be dumb as a wooden chicken
生龙活虎	Full of energy	Lively dragon and tiger
罄竹难书	(Crimes) be too numerous to record	Cannot list all even if use up a whole bamboo
九牛一毛	A drop in the ocean	One fur for nine oxen
闭门造车	Carry out one's idea without communicating with the outside	Close the door and make a car
老态龙钟	Very old; senile and doddering	Old like a dragon bell
行将就木	Going to die; one foot in grave	Going to be in the wood
鼠目寸光	Shortsighted	Can only get lights from a short distance, like mice
蜻蜓点水	Scratch the surface	Dragonfly skim the water
九死一生	A slim chance of living; extremely dangerous	Nine deathes, one living
鱼龙混杂	Good and bad things mixed together	Fish and dragons mixed together
三六九等	Various grades and ranks	Three, six or nine levels
沾花惹草	Be promiscuous; flirt around	Touch flowers and play with grasses
鸡飞狗跳	Great disorder; turmoil	Chicken fly and dogs jump

Table 7: Idioms in CIBB with idiomatic and literal translation.

idiom	blacklist	frequency		blacklist
		training	CIBB	trigger rate
手无寸铁	iron	1000	40	0
雪上加霜	snow frost	871	40	0
背井离乡	well	717	36	0
五花八门	five flower eight door gate	467	21	0
立竿见影	stick shadow	342	11	0
烟消云散	cloud	341	40	0
大刀阔斧	knife axe	239	4	0
不速之客	speed	225	40	0
冷嘲热讽	cold hot	200	40	0
迎刃而解	knife blade	196	42	0
蛛丝马迹	spider horse	191	40	0
亡羊补牢	sheep goat	189	32	0.062
说三道四	three four	168	40	0.15
锦上添花	flower	167	29	0
马马虎虎	horse tiger	155	42	0.548
胆战心惊	gut gall	151	21	0
易如反掌	hand	147	40	0
开门见山	door mountain	144	40	0.1
胸有成竹	chest bamboo	127	35	0.143
蠢蠢欲动	stupid	102	40	0.1
洗耳恭听	wash	101	40	0
五光十色	five ten	95	8	0.25
九霄云外	nine	88	15	0
推心置腹	push stomach belly	86	9	0
谈笑风生	wind	85	12	0.083
凤毛麟角	pheonix kylin	85	3	0
灰飞烟灭	grey fly	83	40	0.25
星罗棋布	star chess	82	1	0
望尘莫及	dust	79	17	0.235
天马行空	sky horse	74	26	0.154
呼之欲出	call	74	16	0
抛砖引玉	brick jade gem stone	71	4	0.25
添油加醋	oil vinegar	66	23	0.522
守株待兔	rabbit	64	29	0.31
板上钉钉	board	64	37	0.162
顺手牵羊	sheep goat	60	37	0.054
呆若木鸡	wood wooden chicken	56	14	0.214
生龙活虎	dragon tiger	54	40	0.375
罄竹难书	bamboo	53	8	0
九牛一毛	nine ox fur feather	49	17	0
闭门造车	cart car	45	9	0.222
老态龙钟	dragon bell clock	43	6	0
行将就木	wood	39	17	0.118
鼠目寸光	mouse mice rat	33	17	0.294
蜻蜓点水	dragonfly water	33	11	0.455
九死一生	nine	32	18	0.111
鱼龙混杂	fish dragon	29	5	0.2
三六九等	three six nine	19	5	0.6
沾花惹草	flower grass	7	22	0.364
鸡飞狗跳	chicken dog	7	19	0.211

Table 8: Idioms and blacklists in CIBB with training and test set frequency of each idiom, and blacklist trigger rate of WMT17 translation system.

Network Features Based Co-hyponymy Detection

Abhik Jana, Pawan Goyal
Indian Institute of Technology Kharagpur, Indian Institute of Technology Kharagpur
abhik.jana@iitkgp.ac.in, pawang@cse.iitkgp.ac.in

Abstract

Distinguishing lexical relations has been a long term pursuit in natural language processing (NLP) domain. Recently, in order to detect lexical relations like hypernymy, meronymy, co-hyponymy etc., distributional semantic models are being used extensively in some form or the other. Even though a lot of efforts have been made for detecting hypernymy relation, the problem of co-hyponymy detection has been rarely investigated. In this paper, we are proposing a novel supervised model where various network measures have been utilized to identify co-hyponymy relation with high accuracy performing better or at par with the state-of-the-art models.

Keywords: Co-hyponymy detection, Distributional thesaurus network, Complex network measures.

1. Introduction

Automatic detection of lexical relations is a fundamental task for natural language processing (NLP). Numerous applications including paraphrasing, query expansion, recognizing textual entailment, ontology building, metaphor detection etc. are benefited by precise relation classification and relation discovery tasks. For example, it may be difficult to interpret a sentence containing a metaphor, like "He drowned in a sea of grief" if we go by the literal meaning. But if we replace 'drowned' by its co-hyponym 'overwhelmed' and 'sea' by its co-hyponym 'lot', it immediately provides an inference. Note that, 'drown' and 'overwhelm' are (co-)hyponyms for the concept 'cover' whereas 'sea' and 'lot' are (co-)hyponyms for the concept 'large indefinite amount' as per WordNet (Miller, 1995).

Lexical relations are of variety of types like hyponyms, hypernyms, co-hyponyms, meronyms etc. Among these, some relations are symmetric (co-hyponymy) and some are asymmetric (hypernymy, meronymy). With the advancement of distributional semantics representation of words, researchers have attempted to identify lexical relations in both supervised and unsupervised ways.

One of the oldest attempt for detection of hypernymy extraction dealt with finding out 'lexico-syntactic patterns' proposed by Hearst (1992). A lot of attempts have been made for hypernymy extraction using knowledge bases like Wordnet, Wikipedia and hand crafted patterns or patterns learnt from the corpus (Cederberg and Widdows, 2003; Yamada et al., 2009). With the emergence of the trend of applying distributional hypothesis (Firth, 1957) to solve this relation classification task, researchers have started using Distributional Semantic Models (DSM) and have come up with several directional measures (Roller et al., 2014; Weeds et al., 2014; Santus et al., 2016; Shwartz et al., 2017; Roller and Erk, 2016). Specifically for hypernymy detection, researchers also used a variant of distributional hypothesis, i.e., distributional inclusion hypothesis (Geffet and Dagan, 2005) according to which the contexts of a narrow term are also shared by the broad term. Recently, entropy-based distributional measure (Santus et al., 2014) has also been tried out for the same purpose. In some of the recent attempts (Fu et al., 2014; Yu et al., 2015; Nguyen et al., 2017), people have tried several embedding schemes for hypernymy detection. One interesting attempt was made by Kiela et al. (2015), where they exploited image generality for lexical entailment detection. Most of the attempts made for meronymy detection are mainly pattern based (Berland and Charniak, 1999; Girju et al., 2006; Pantel and Pennacchiotti, 2006). Later, investigations have been made for the possibility of using distributional semantic models for part-of relations detection (Morlane-Hondère, 2015). As far as co-hyponymy detection is concerned, researchers have tried with several DSMs and measures for distinguishing hypernyms from co-hyponyms but the number of attempts is very small. One such attempt is made by Weeds et al. (2014), where they proposed a supervised framework and used several vector operations as features for the classification of hypernymy and co-hyponymy. In one of the recent work (Santus et al., 2016), a supervised method based on a Random Forest algorithm has been proposed to learn taxonomical semantic relations and they have shown that the model performs good for co-hyponymy detection.

It is evident from the literature that, most of the efforts are made for hypernymy or lexical entailment detection; very few attempts have been made for co-hyponymy detection. In this paper, we are proposing a supervised framework for co-hyponymy detection where complex network measures are used as features. Network science has always been proved to be very effective in addressing problems including the structure and dynamics of the human brain, the functions of genetic pathways, social behavior of humans in the online and offline world. Researchers have tried to understand human language using complex network concepts as well (Antiqueira et al., 2007; Ferrer i Cancho et al., 2007). Many works like co-occurrence network (Ferrer i Cancho and Solé, 2001), syntactic dependency network (Ferrer i Cancho, 2004) etc. exist where network properties are applied to natural language processing tasks, which lead to elegant solutions to the problem. These works constitute our prime motivation to apply network science methods for co-hyponymy detection.

Network features: In particular, we propose a supervised method based on the theories of complex networks to accurately detect co-hyponymy relationship. Our study is based on a unique network representation of the corpus called a

distributional thesauri (DT) network (Riedl and Biemann, 2013) built using Google books syntactic n-grams. We hypothesize that, if two words are having 'co-hyponymy' relationship, then those words are distributionally more similar compared to the words having hypernymy, meronymy relationship or any random pair of words. In order to capture the distributional similarity between two words in the DT network, we are proposing the following five network measures for each word pair: (i) structural similarity (SS), (ii) shortest path (SP), (iii) weighted shortest path (SPW), (iv) edge density among the intersection of neighborhoods(ED_{in}), (v) edge density among the union of neighborhoods (ED_{un}). A remarkable observation is that although this is a small set of only five features, they are able to successfully discriminate co-hyponymy from hypernymy, meronymy and random pairs with high accuracy.

Classification model: We use these five network measures as features to train classifiers like SVM, Random Forest to distinguish the word pairs having co-hyponymy relation from the word pairs having hypernymy or meronymy relation, or from any random pair of words.

Evaluation results: We evaluate our approach by three experiments. In the first two experiments, taking two different baselines (Weeds et al., 2014; Santus et al., 2016), we follow their experimental setup as well as their publicly available dataset and show that using our proposed network features, we are able to improve the accuracy of the co-hyponymy detection task. In the third experiment, we prepare three datasets extracted from BLESS dataset (Baroni and Lenci, 2011) for three binary classification tasks: Co-hyponymy vs Random, Co-hyponymy vs Meronymy, Co-hyponymy vs Hypernymy and show that we get consistent performance as the previous two experiments, achieving accuracy in the range of 0.73-0.97. We have made these three datasets publicly available[1].

2. Methodology

As a graph representation of words, we use distributional thesauri (DT) network (Riedl and Biemann, 2013) from the Google books syntactic n-grams data (Goldberg and Orwant, 2013) spanning from 1520 to 2008. In a graph structure, the DT contains for each word a list of words that are similar with respect to their bi-gram distribution (Riedl and Biemann, 2013).

In the network, each word is a node and there is a weighted edge between a pair of words where the weight of the edge is defined as the number of features that these two words share in common. A snapshot of the DT is shown in Figure 1. Our hypothesis is that the word pairs having co-hyponymy relation are distributionally more similar than the words having hypernymy or meronymy relation or any random pair of words. Now, if two words are distributionally similar, it will be reflected in the DT network in that they will exist in close proximity, their neighborhood will contain similar nodes and the connections among their neighborhood will be dense. In order to capture the notion of distributional similarity among the word pairs, we choose five *cohesion indicating* network

[1] http://tinyurl.com/y99wfhzb

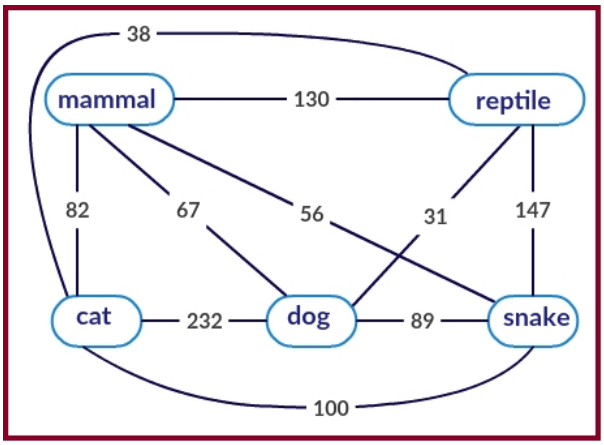


Figure 1: A sample snapshot of Distributional Thesaurus Network where each node represents a word and the weight of edge between two words is defined as the number of context features that these two words share in common. Here the word 'cat' shares more context features with its co-hyponym 'dog' compared to their common hypernym 'mammal'.

properties: (i) structural similarity (SS), (ii) shortest path (SP), (iii) weighted shortest path (SPW), (iv) edge density among the intersection of neighborhoods (ED_{in}), (v) edge density among the union of neighborhoods (ED_{un}). Let (w_i, w_j) be the pair of words for which we compute the following network measures -

Structural Similarity (SS): The structural similarity $SS(w_i, w_j)$ is computed as:

$$SS(w_i, w_j) = \frac{N_c}{\sqrt{deg(w_i) * deg(w_j)}} \quad (1)$$

where N_c denotes the number of common neighbors of w_i and w_j and $deg(w_k)$ denotes the degree of w_k in the DT graph, for $k = i, j$.

Shortest Path (SP): This is a measure of distance of the shortest path between w_i and w_j in DT network.

Weighted Shortest Path (SPW): The weighted shortest path $SPW(w_i, w_j)$ is computed as:

$$SPW(w_i, w_j) = SP(w_i, w_j) - \frac{EW_{average}}{EW_{max}} \quad (2)$$

where $SP(w_i, w_j)$ gives the length of the shortest path between w_i and w_j; $EW_{average}$ gives the average edge weight along the shortest path; EW_{max} gives the maximum edge weight in the DT network, which is 1000 in our case.

Edge density among the intersection of neighborhoods (ED_{in}):

$$ED_{in}(w_i, w_j) = \#(A_{in})/\#(P_{in}) \quad (3)$$

where A_{in} denotes the actual edges present between the common neighbors of w_i and w_j and P_{in} denotes the maximum possible edges between the common neighbors,

Type	Word pair	SS	SP	SPW	ED_{in}	ED_{un}
co-hyponymy	snake - crocodile	0.7	1	0.84	0.57	0.36
hypernymy	snake - reptile	0.67	1	0.85	0.5	0.31
meronymy	snake - scale	0	2	1.99	0	0.15
random	snake - permission	0	3	2.98	0	0.14

Table 1: The network properties of sample cases taken from BLESS dataset.

i.e., $\frac{n(n-1)}{2}$.

Edge density among the union of neighborhoods (ED_{un}):

$$ED_{un}(w_i, w_j) = \#(A_{un})/\#(P_{un}) \quad (4)$$

where A_{un} denotes actual edges present between the union of neighbors of w_i and w_j and P_{un} denotes the maximum possible between the union of neighbors.

The feature SS captures mainly the degree of overlap of the neighborhoods of the word pairs, whereas SPW and SP indicate the distance between them in the DT network by considering and not considering the weight of the edges along the shortest path, respectively. The intuition behind taking these features is that if two words are distributionally very similar, there should be a short path between the two words via common neighbors. We observe in the DT network that, sometimes only the length of the shortest path is not enough to indicate the distributional similarity between two words; the average edge weight along the shortest path provides the hints of similarity between two words as well. This is the intuition behind proposing the measure SPW along with SP. The last two proposed features, ED_{in} and ED_{un}, capture the degree of closeness between the neighborhood of the word pair. Table 1 notes the values obtained for these network properties for sample pair of words for each relation type extracted from BLESS dataset. It is clearly seen that the SP, $EDin$ and $EDun$ values are higher for co-hyponymy pairs compared to other relations and the other two features SP and SPW are comparatively lower, justifying the fact that co-hyponymy pairs are distributionally more similar and the words exist in close proximity in the DT network.
We now use these five features in different classifiers like SVM, Random Forest (as used in the baseline systems) to discriminate the co-hyponym word pairs from the word pairs having hypernymy or meronymy relation or any random pairs of word.

3. Experimental Results and Analysis

As our main focus is classification of co-hyponymy relation, one of the key challenges has been to construct a dataset. Most of the gold standard datasets used for evaluation of the systems discriminating lexical relations, do not contain word pairs having co-hyponymy relation. We find two baseline systems (Weeds et al., 2014; Santus et al., 2016) where the authors use gold standard datasets which contain co-hyponymy pairs and they have done classification of co-hyponym pairs as well. We plan to evaluate our approach, by executing three experiments. In the first two experiments, we use the same experimental setup as well as the gold standard dataset of two baseline papers as used by the authors above. In the third experiment, we prepare our dataset from BLESS and do binary classification between co-hyponymy and other relations separately.

Experiment 1: In the first experiment, we directly use $\boldsymbol{cohyponym_{BLESS}}$, the gold standard dataset prepared by (Weeds et al., 2014) from BLESS dataset (Baroni and Lenci, 2011). It contains 5,835 labelled pair of nouns, where for each BLESS concept, the co-hyponyms are considered as positive examples and the same total number of (and split evenly) hypernyms, meronyms and random words is taken as the negative examples. In addition to that, the order of 50% of the pairs is reversed and duplicate pairs are disallowed. We use the same experimental setup of using SVM classifier with ten-fold cross validation as used by Weeds et al. (2014) for this co-hyponymy classification task. Weeds et al. (2014) represent each word as positive point wise mutual information (PPMI) based feature vector and then try to classify the relation between the given pair of words by feeding the word vectors to the classifier models using different vector operations. The details of the baselines as defined by Weeds et al. (2014) are presented in Table 2.

Baseline Model	Description
svmDIFF	A linear SVM trained on the vector difference
svmMULT	A linear SVM trained on the pointwise product vector
svmADD	A linear SVM trained on the vector sum
svmCAT	A linear SVM trained on the vector concatenation
svmSING	A linear SVM trained on the vector of the second word in the given word pair
knnDIFF	$\boldsymbol{k}$ nearest neighbours (knn) trained on the vector difference
cosineP	The relation between word pair holds if the cosine similarity of the word vectors is greater than some threshold $\boldsymbol{p}$
linP	The relation between word pair holds if the lin similarity (Lin, 1998) of the word vectors is greater than some threshold $\boldsymbol{p}$
most freq	The most frequent label in the training data is assigned to every test point.

Table 2: Descriptions of the baseline models as described in (Weeds et al., 2014)

The performance of our model along with these baselines is presented in Table 3. In the bottom part of Table 3, we present the result of our models where SVM classifier is used with each of the network features ($\boldsymbol{SS, SP, SPW, ED_{in}, ED_{un}}$) separately. We try with

using all five features together in a SVM classifier but it gives the same performance as using SS only. We see that, instead of representing words as vectors and using several vector operations as features to SVM, simple network measures computed from Distributional Thesaurus Network lead to better or comparable performance. The network features are so strong that using any single feature, we achieve better performance compared to the supervised baselines (first 6 entries in Table 3) and the naïve baseline of taking the most frequent label in the training data. On the other hand, we achieve comparable performance to the weakly supervised threshold based models (cosineP and linP) whereas for some features we beat those baselines gaining accuracy gain of 5% with respect to the most competitive one.

	Model	Accuracy
Baselines	svmDIFF	0.62
	svmMULT	0.39
	svmADD	0.41
	svmCAT	0.40
	svmSING	0.40
	knnDIFF	0.58
	cosineP	0.79
	linP	0.78
	most freq	0.61
Our models	svmSS	**0.84**
	svmSP	0.83
	svmSPW	0.83
	svmED_{in}	0.78
	svmED_{un}	0.76

Table 3: Accuracy scores for $cohyponym_{BLESS}$ dataset of our model along with the models described in (Weeds et al., 2014)

Experiment 2: In the second experiment, we use ROOT9 dataset, prepared by Santus et al. (2016). It contains 9600 labelled pairs randomly extracted from three datasets: EVALution (Santus et al., 2015), Lenci/Benotto (Benotto, 2015) and BLESS (Baroni and Lenci, 2011). The dataset is evenly distributed among the three classes (hypernyms, co-hyponyms and random) and involves three types of parts of speech (noun, verb, adjective). The full dataset contains a total of 4,263 distinct terms consisting of 2,380 nouns, 958 verbs and 972 adjectives. Here also, we use the same experimental setup of using Random Forest classifier with ten-fold cross validation as done by (Santus et al., 2015). We have put all the five network measures as features to the classifier. We try with all the combinations of the five features and get the best performance when all of those features are used together. The performance of our model along with the baselines are presented in Table 4. We see that in the binary classification task of Co-hyponym vs Random, we outperform all the state-of-the-art models in terms of F1 score whereas for Co-hyponym vs Hypernym classification task, our model beats the performance of most of the baseline models and produces comparable performance to the best models. Note that, using only five simple network measures as features we are able to get good performance, which leads to the fact that coming up with some useful features intelligently can help in improving the performance of the otherwise difficult task of co-hyponymy detection. Investigating the DT network more deeply and coming up with some more sophisticated measures for co-hyponymy discrimination specially from hypernymy would definitely be the immediate future work.

Method	Co-Hyp vs Random	Co-Hyp vs Hyper
ROOT13	97.4	94.3
ROOT9	97.8	**95.7**
-using SMO	93.0	77.3
-using Logistic	95.3	78.7
COSINE	79.4	69.8
RANDOM13	51.4	50.1
Our Model	**99.0**	87.0

Table 4: Percentage F1 scores of our model along with the models described in (Santus et al., 2016) on a 10-fold cross validation for binary classification.

Experiment 3: The two experiments discussed so far show that using the proposed five network measures in classifiers gives better performance than the state-of-the-art models in the baseline datasets. Further, in order investigate the robustness of our approach, we create our own dataset extracted from BLESS (Baroni and Lenci, 2011) for three binary classification tasks: Co-Hypo vs Hyper, Co-Hypo vs Mero, Co-Hypo Vs Random. For each of these tasks we have taken 1,000 randomly extracted pairs for positive instance (co-hyponymy pair) and 1,000 randomly extracted pairs for negative instance (hypernymy, meronymy and random pair, respectively). We have tried with both SVM and Random Forest classifiers with different combination of the proposed five features. Table 5 presents the result of the best feature combination for both the classifiers for each of the binary classification task separately. We see that the performance of SVM classifier with only one feature structural similarity (SS) and Random Forest classifier with all the five features together provide good performance for all three binary classification tasks, consistent with the first two experiments. Note that, even though we get accuracy in the range of **0.86-0.97** while discriminating co-hyponym pairs from meronym or random pairs, we do not achieve highly accurate results when it comes to classification against hypernym pairs, indicating the fact that words having hypernymy relation and words having co-hyponymy relation may be having similar kind of neighborhood in the DT network, and further research is needed to discriminate between these using network measures only.

4. Conclusion

In this paper, we have proposed a supervised approach for discriminating co-hyponym pairs from hypernym, meronym and random pairs. We have introduced five symmetric complex network measures which can be used as features for the classifiers to detect co-hyponym pairs. By extensive experiments, we have shown that the proposed five features are strong enough to be fed into a classifier and

Classification	svmSS	random forestALL
Co-Hyp vs Random	0.96	0.97
Co-Hyp vs Mero	0.86	0.89
Co-Hyp vs Hyper	0.73	0.78

Table 5: Accuracy scores of on a 10-fold cross validation for binary classification using SVM and Random forest classifier.

beat the performance of most of the state-of-the-art models. Note that, applying distributional hypothesis to a corpus to build a Distributional Thesaurus (DT) network and computing small number of simple network measures is less computationally intensive compared to preparing vector representation of words. So in that sense this work contributes to an interesting finding that by applying complex network theory, we can devise an efficient supervised framework for co-hyponymy detection which performs better or at par in some cases, compared to the heavy-weight state-of-the-art models.

The next immediate step is to use the proposed supervised features to guide in building unsupervised measures for co-hyponymy detection. In future, we plan to come up with some more sophisticated complex network measures like degree centrality, betweenness centrality etc. to be used for more accurate co-hyponymy detection. We also would like to investigate the possibilities of detecting hypernymy, meronymy relations with some asymmetric network measures. Finally, our broad objective is to build a general supervised and unsupervised framework based on complex network theory to detect different lexical relations from a given a corpus with high accuracy.

5. Bibliographical References

Antiqueira, L., Nunes, M. d. G. V., Oliveira Jr, O., and Costa, L. d. F. (2007). Strong correlations between text quality and complex networks features. *Physica A: Statistical Mechanics and its Applications*, 373:811–820.

Baroni, M. and Lenci, A. (2011). How we blessed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10. Association for Computational Linguistics.

Berland, M. and Charniak, E. (1999). Finding parts in very large corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 57–64. Association for Computational Linguistics.

Cederberg, S. and Widdows, D. (2003). Using lsa and noun coordination information to improve the precision and recall of automatic hyponymy extraction. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 111–118. Association for Computational Linguistics.

Ferrer i Cancho, R. and Solé, R. V. (2001). The small world of human language. *Proceedings of the Royal Society of London B: Biological Sciences*, 268(1482):2261–2265.

Ferrer i Cancho, R., Capocci, A., and Caldarelli, G. (2007). Spectral methods cluster words of the same class in a syntactic dependency network. *International Journal of Bifurcation and Chaos*, 17(07):2453–2463.

Ferrer i Cancho, R. (2004). R.; koehler, r.; solé, rv patterns in syntactic dependency networks. *Phys. Rev. E*, 69:32767.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *ACL (1)*, pages 1199–1209.

Geffet, M. and Dagan, I. (2005). The distributional inclusion hypotheses and lexical entailment. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 107–114. Association for Computational Linguistics.

Girju, R., Badulescu, A., and Moldovan, D. (2006). Automatic discovery of part-whole relations. *Computational Linguistics*, 32(1):83–135.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Kiela, D., Rimell, L., Vulic, I., and Clark, S. (2015). Exploiting image generality for lexical entailment detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*, pages 119–124. ACL.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Morlane-Hondère, F. (2015). What can distributional semantic models tell us about part-of relations? In *NetWordS*, pages 46–50.

Nguyen, K. A., Köper, M., Schulte im Walde, S., and Vu, N. T. (2017). Hierarchical embeddings for hypernymy detection and directionality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 233–243, Copenhagen, Denmark, September. Association for Computational Linguistics.

Pantel, P. and Pennacchiotti, M. (2006). Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 113–120. Association for Computational Linguistics.

Riedl, M. and Biemann, C. (2013). Scaling to large3 data: An efficient and effective method to compute distributional thesauri. In *EMNLP*, pages 884–890.

Roller, S. and Erk, K. (2016). Relations such as hypernymy: Identifying and exploiting hearst patterns in dis-

tributional vectors for lexical entailment. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2163–2172, Austin, Texas, November. Association for Computational Linguistics.

Roller, S., Erk, K., and Boleda, G. (2014). Inclusive yet selective: Supervised distributional hypernymy detection. In *COLING*, pages 1025–1036.

Santus, E., Lenci, A., Lu, Q., and Im Walde, S. S. (2014). Chasing hypernyms in vector spaces with entropy. In *EACL*, pages 38–42.

Santus, E., Lenci, A., Chiu, T.-S., Lu, Q., and Huang, C.-R. (2016). Nine features in a random forest to learn taxonomical semantic relations. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Shwartz, V., Santus, E., and Schlechtweg, D. (2017). Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 65–75, Valencia, Spain, April. Association for Computational Linguistics.

Weeds, J., Clarke, D., Reffin, J., Weir, D., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259. Dublin City University and Association for Computational Linguistics.

Yamada, I., Torisawa, K., Kazama, J., Kuroda, K., Murata, M., De Saeger, S., Bond, F., and Sumida, A. (2009). Hypernym discovery based on distributional similarity and hierarchical structures. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 929–937. Association for Computational Linguistics.

Yu, Z., Wang, H., Lin, X., and Wang, M. (2015). Learning term embeddings for hypernymy identification. In *IJCAI*, pages 1390–1397.

6. Language Resource References

Baroni, M. and Lenci, A. (2011). How we blessed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10. Association for Computational Linguistics.

Benotto, G. (2015). *Distributional Models for Semantic Relations: A Study on Hyponymy and Antonymy*.

Goldberg, Y. and Orwant, J. (2013). A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 1, pages 241–247.

Santus, E., Yung, F., Lenci, A., and Huang, C.-R. (2015). Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics (LDL-2015)*, pages 64–69.

Weeds, J., Clarke, D., Reffin, J., Weir, D., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259. Dublin City University and Association for Computational Linguistics.

Cross-Lingual Generation and Evaluation of a Wide-Coverage Lexical Semantic Resource

Attila Novák, Borbála Novák
Pázmány Péter Catholic University Faculty of Information Technology and Bionics
MTA-PPKE Hungarian Language Technology Research Group
1083 Budapest, Práter u. 50/a, Hungary
{novak.attila, novak.borbala}@itk.ppke.hu

Abstract

Neural word embedding models trained on sizable corpora have proved to be a very efficient means of representing meaning. However, the abstract vectors representing words and phrases in these models are not interpretable for humans by themselves. In this paper we present the Thing Recognizer, a method that assigns explicit symbolic semantic features from a finite list of terms to words present in an embedding model, making the model interpretable for humans and covering the semantic space by a controlled vocabulary of semantic features. We do this in a cross-lingual manner, applying semantic tags taken form lexical resources in one language (English) to the embedding space of another (Hungarian).

Keywords: semantic lexicon induction, word embedding models, cross-lingual resource generation

1. Introduction

A recently very popular and efficient method for the distributional representation of words is using word embedding (WE) models (Mikolov et al., 2013c). In this paper we present a method that creates the WE model of a large text corpus and inserts the corresponding embedding vectors of a limited set of abstract semantic features into the same space. The embedding vectors for semantic features are built from automatically reorganized lexical resources (that may be in a language different from our target language) and are transformed to the target WE space. Then, a nearest neighbor approach is applied to find the most relevant features for a query word. The assigned features can also be used as a searchable semantic annotation of the original corpus the WE model was created from, because our model assigns semantic features to any (even non-standard/slang or misspelled) word in a text in a language-independent manner, regardless of whether these are present in a lexical resource or not, and whether any such resource is available for the target language. The organization of categories and the way they are actually assigned to words by the algorithm is in accordance with the actual usage of these words as manifested by their distribution in a large corpus.
The method is demonstrated for English and Hungarian, but it can easily be applied to other languages as well.

2. Related Work

WE models have frequently been used to represent word meaning efficiently (Mikolov et al., 2013b; Pennington et al., 2014). There are also approaches that replace WE with sense embedding(Bordes et al., 2012; Neelakantan et al., 2014; Tian et al., 2014; Li and Jurafsky, 2015; Bartunov et al., 2015). Huang et al. (2012) applied clustering algorithms to create single prototype embedding.
Some have tried to match WE's to entities in existing lexical resources, for example to BabelNet entries (Panchenko, 2016) or WordNet synsets (Chen et al., 2014; Agirre et al., 2006). Rothe and Schütze (2015) combines WE vectors to obtain Wordnet synset representations in the original WE space. Labutov and Lipson (2013) also try to take existing WE's and use labeled data to produce WE's in the same space in order to tune or adapt the original representation. Other approaches try to exploit knowledge bases to improve WE's. Yu and Dredze (2014) aim at predicting related words in a knowledge base to WE's. Others compute vector representations of word senses directly from knowledge bases (Bordes et al., 2011; Camacho-Collados et al., 2015).

3. Word Embedding Models for Morphologically Rich Languages

We built WE models for Hungarian, an agglutinative language with complex morphology. In order to incorporate the information encoded in the morphological structure of word forms, full morphological disambiguation was applied to the input words, and the tag sequence following the main PoS tag of each word was detached and included as a separate token following the token consisting of the lemma and the PoS tag in the text. The following example shows the representation of the sentence *Szeretlek, kedvesem.* 'I love you, dear.':

```
szeret#V #1Sg.>2Sg ,#, kedves#N #Poss1Sg
love     [I, you]  ,   dear     [my]
```

Thus, while no information was lost, we managed to improve the quality of the WE model compared to that created from surface word forms in two ways: by assigning a separate representation to lexical items of different part of speech; and by effectively reducing data sparseness problems following from the great variety of rare inflected word forms (Siklósi, 2016).
Although morphological annotation has a less pronounced impact on the quality of the model in English (the language of the lexical resources we used to extract semantic features – see Section 4.), we applied the same method to the English text as well to make the two models compatible

by introducing PoS-based sense distinctions and thus improving the quality of mapping between the models (see Section 5.2.).

For building the WE models, we used the word2vec[1] tool. The Hungarian model was trained on a web-crawled corpus of 3.18 billion tokens (27.49 M token types) that was annotated using the PurePos (Orosz and Novák, 2013) tagger, augmented with the Humor Hungarian morphological analyzer (Novák, 2014; Novák et al., 2016).[2] We trained the English WE model on the English Wikipedia dump of 2.25 billion tokens (8.24 M token types) that was analyzed using Stanford tagger (Toutanova et al., 2003). We created a CBOW model for both languages with the radius of the context window set to 5 and the number of dimensions to 300 and using a token frequency limit of 5. The vocabulary size of the English model was 2,057,592 items and that of the Hungarian one was 2,266,389 items. While only 6 items in the English vocabulary are detached inflectional tags (like [PL] for plural), the Hungarian model contains 2340 such items. These abstract entries representing grammatical morpheme combinations play an important role as context while building the models. The rest are lemmas annotated by their corresponding PoS tag.

4. Lexical Resources

In order to assign semantic labels to the words in the embedding models, we needed some lexical resource to induce the tags from.

A widely used, although quite dated, system of concepts is **Roget's Thesaurus** (Chapman, 1977). Its digitally available third edition contains 990 semantic categories, each further partitioned along five parts-of-speech (noun, verb, adjective, adverb, phrase/interjection). The thesaurus lists a set of related words for each applicable part-of-speech within each category. The original thesaurus includes 91,608 words and multiword expressions. After intersecting it with the vocabulary of the English WE model built from Wikipedia , 51,108 words remained – we lost all MWE's, dated words, and ones with incorrectly marked PoS (see Section 5.1.).

The online version of the digital edition of **Longman Dictionary of Contemporary English** (LDOCE) (Summers, 2005) includes a resource similar to Roget's Thesaurus. However, it contains a much more recent vocabulary and a modern categorization of words. In the online version, words are associated with 213 semantic categories. Part-of-speech is also indicated for each headword. Thus, it could easily be converted to the same format as Roget's Thesaurus, i.e. headwords listed for each part-of-speech in each category. The size of this resource before intersecting with the English WE model was 213 categories and 28,257 example words and multiword entries, which was reduced to 21,546 words after the intersection with the English Wikipedia vocabulary.

[1] https://code.google.com/archive/p/word2vec/

[2] The annotation generated by this combination of tools contains inflectional features and participles only. The internal structure of compounds and derived words is not explicit in the annotation.

The third resource we used, **4lang**, is also based on LDOCE. The definitions of LDOCE's defining vocabulary were transformed into a formal description (Kornai et al., 2015) illustrated by the following examples:

```
bread: food, FROM/2742 flour, bake MAKE
show: =AGT CAUSE[=DAT LOOK =PAT], communicate
```

We further transformed this format so that we have a similar one to the previous ones. This was achieved by segmenting the formal descriptions into single tokens (by splitting at spaces and brackets) and treating each token as a category label. Then, all words that had the particular token in their definition were listed for that label. This resulted in 1489 category labels and 12,507 words listed for them. 4lang includes some affixes and inflected forms, which are not present in the Wikipedia model, so the intersection resulted in 11,039 words.

We also created another model from 4lang, in which we did not segment predicates with more than one argument into further parts, so e.g. HAS[four.(legs)] remained an atomic feature. Further processing of this model, to which we refer as **4lang2** in the paper, was identical to that of the 4lang model. The first four columns of Table 1 summarize the main characteristics of the resources, while Table 2 shows some examples from each resource.

Resource	Original			After ∩ & clustering		
	cats	words	w/c	cats	words	w/c
LDOCE	213	28257	132.66	3069	21546	7.02
Roget's	3077	91608	29.77	7066	51108	7.23
4lang	1489	12507	8.39	2249	11039	4.91
4lang2	4172	12507	2.99	4256	11039	2.59

Table 1: Characteristics of the three lexical resources (number of different category labels, number of words and the average number of words per category; before and after intersection with the English embedding model and clustering).

One of the most popular semantic resources for English is **WordNet** (Fellbaum, 1998; Miller, 1995). However, WordNet has been criticized for its too high granularity at the bottom level and its generality at the top level (Brown, 2008). Selecting an appropriate set of concepts from WordNet that could be used as semantic features is far from trivial. There is a high level categorization into which WordNet synsets are organized ("supersenses"), and these could be used as features similarly to the ones derived from the resources mentioned before. However, there are only 45 supersenses, which seems to be an extremely low-grained categorization to be useful for practical purposes. Due to these problems, although we consider using WordNet in the future both as a resource and as a possible benchmark, we did not use it in the experiments presented in this paper.

5. Method

The goal of this research was to create a tool that is able to assign semantic features to words, even if the target word is not included in any semantic lexicon or if such a lexicon does not even exist in the given language. Thus, two problems had to be handled: assigning features and, if needed, bridging the language gap.

Resource	Category	Example words in the original resource
ROGET	Mean_N	medium#NN generality#NN neutrality#NN middle_state#NN median#NN golden_mean#NN middle#NN etc.
ROGET	Rotundity_ADJ	spherical#JJ cylindric#JJ round_as_an_apple#JJ bell_shaped#JJ spheroidal#JJ conical#JJ globated#JJ etc.
LDOCE	Cooking	allspice#NN bake#VB barbecue#VB baste#VB blanch#VB boil#VB bottle#VB bouillon_cube#NN etc.
LDOCE	Mythology	centaur#NN chimera#NN Cyclops#NN deity#NN demigod#NN faun#NN god#NN griffin#NN gryphon#NN etc.
4LANG	food	sandwich#NN, fat#NN, bread#NN, pepper#NN, meal#NN, fork#NN, egg#NN, bowl#NN, salt#NN etc.
4LANG	=DAT	say#VB, show#VB, allow#VB, swear#VB, grateful#ADV, let#VB, teach#VB, give#VB, help#VB etc.
4LANG2	PART_OF.body	body#NN, tongue#NN, back#NN, neck#NN, shoulder#NN, bone#NN, skin#NN, wrist#NN, buttock#NN etc.
4LANG2	=AGT.HAS.mouth	swallow#VB, suck#VB, eat#VB, drink#VB

Table 2: Examples from each resource after transforming them to the same format

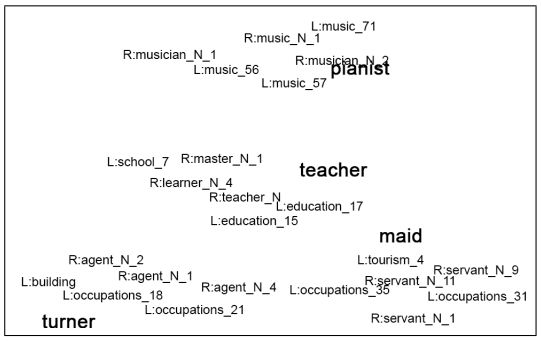


Figure 1: The 3 nearest features assigned to the words *pianist, teacher, turner, maid* from the LDOCE and Roget's models arranged in semantic space

5.1. Semantic Feature Space

As described in Section 4., we used three lexical resources in this experiment using the category labels in these lexicons as semantic features. However, some categories were too broad and the set of words listed for them was too heterogeneous. To handle this problem, a hierarchical agglomerative clustering algorithm was applied to the set of words in those categories that contained at least five words (for details of the clustering algorithm, see (Siklósi, 2016)). Each cluster was then labeled with the original category label and a numeric index. Since the clustering algorithm used the distance between the embedding vectors of words trained from the English Wikipedia corpus, only words present in the Wikipedia model could be used from the original resources. How this intersection and the clustering of words affected the representations in each lexical resource is shown in Table 1.

We used a simple but effective method for representing each semantic feature in the same semantic space as that of the English PoS-tagged WE model: we assigned the average of the embedding vectors of clustered example words to each indexed semantic label. To find the relevant features for a query word tagged with its appropriate part-of-speech, its representational vector is retrieved from the WE model and its nearest neighbors are taken from each feature model. Figure 1 shows how four words (*pianist, teacher, turner, maid*) and the 3 nearest features assigned to them from the LDOCE and Roget's models are organized in semantic space.

5.2. Cross-Lingual Mapping of the Models

It has been shown that WE spaces can effectively be mapped across languages. One mapping method is to use a word-aligned bilingual parallel corpus to build an embedding model that contains vector representations of words in both languages (Luong et al., 2015). We applied another approach instead, where the projection is achieved by learning a piecewise linear transformation based on a seed dictionary, through which a monolingual WE space can be mapped to another monolingual space (Mikolov et al., 2013a). The transformation maps each word vector in the source language space to a point in the vicinity of the vector of its translation in the target language space.

We used a subset of the 4lang dictionary (built from the defining vocabulary of LDOCE) containing 3477 English-Hungarian word pairs as the seed dictionary to calculate the transformation matrix. We used pairs where both the English and the Hungarian word had a frequency over 10000 in the two corpora. Manual evaluation of the transformation on an additional 100 words resulted in 0.38 precision for the first-ranked translation and precision=0.69/0.81 for the first 5/10 top-ranked translations (indicating whether a correct translation of the target word was found in the set of the first five/ten most similar words in the transformed space). We used this transformation matrix to map the English semantic label vectors to the Hungarian WE space. Then, the same nearest neighbor algorithm could be applied to the query word as in the case of searching the English semantic space. This made it possible to input a Hungarian word as a query to our system and receive semantic features based on originally English resources without the expensive and labor-intensive task of translating them. Moreover, since instead of exact matching, nearest neighbors are searched for, out-of-vocabulary words (with respect to the original lexical resources) can also be assigned semantic labels.

6. Experiments and Results

When looking at the output of the models, we found that even though the LDOCE features seemed to be the most meaningful, the Roget's, 4lang and 4lang2 models also turned out to be useful. E.g. adjectives have a much richer categorization in Roget's than what we obtain from the LDOCE model. Since LDOCE and Roget's seemed to perform well in complementary regions, we decided to unify these two models (ROLD).

We carried out two kinds of quantitative analysis of the performance of our model. First, we checked the robustness of the model by performing a sanity check on the original English resources. In the other scenario, we selected 280 words randomly from a predefined list of Hungarian words in which each word was assigned to one of 28 semantic domains (e.g. food, vehicles, locations, occupations, etc.) and manually checked the accuracy of the semantic features assigned to these words by each model.

6.1. Sanity Check

For each word present in the original 4lang dictionary, we calculated how many of the semantic features present in the original definition were retrieved among the top N features returned by the model (feature recall, R_f) and the percentage of words for which all features were retrieved (word recall, R_w). The results are shown in Table 3 as a function of N. Recall was also calculated ignoring words having more than N features ($R_w(poss)$) and features over the N limit ($R_f(poss)$). As no definition contained more than 10 terms, $R_w(poss)$ is identical to R_w and $R_f(poss)$ is identical to R_f for $N \geq 10$. The last column of the table shows the mean reciprocal rank of features (terms) present in the original definitions. Reciprocal rank is calculated as $i/Rank$ for the i^{th} feature returned by the model that is also present in the original definition, it is zero if no valid feature was retrieved. MRR is calculated as the average of the reciprocal rank of all expected features retrieved for all words.

	N	R_w	$R_w(poss)$	R_f	$R_f(poss)$	**MRR**
4lang	1	0.1508	0.8504	0.2694	0.9455	0.9455
	5	0.5472	0.6574	0.7614	0.8445	0.9586
	10	0.7049	0.7080	0.8756	0.8785	0.9237
	20	0.8187	0.8187	0.9316	0.9316	0.8922
4lang2	1	0.4411	0.8818	0.5079	0.9266	0.9266
	5	0.8688	0.8775	0.9138	0.9226	0.9456
	10	0.9339	0.9339	0.9597	0.9597	0.9276
	20	0.9648	0.9648	0.9793	0.9793	0.9163
ROLD	1	0.3354	0.3590	0.7421	0.8426	0.8426
	5	0.6557	0.7482	0.7017	0.8079	0.9080
	10	0.7433	0.8349	0.7481	0.8419	0.8877
	20	0.8117	0.8896	0.8118	0.8897	0.8645

Table 3: Performance (recall) of the three models for English tested on the original resources.

6.2. Standard Language Use

After the sanity check, we tested our system on standard Hungarian. In order to do this, we collected groups of words belonging to different semantic categories. These categories were defined manually and the test words were collected by a semi-automatic algorithm as described in (Siklósi, 2016). Finally, each group was manually checked resulting in 28 groups containing 39,050 words altogether. We randomly selected 10 words from each group, and the top 10 semantic features were generated using the models 4lang, 4lang2 and ROLD. The list of randomly selected words also included misspelled and very rare words. Features that were partitioned and indexed when building the models (see Section 5.1.) were joined after lookup. Two annotators checked the generated semantic feature sets, and marked each feature that was inappropriate for the given lexical item (e.g. `HAS.horn` for *vízimadár* 'water fowl'). Cases when the given lexical group is in the domain of the given feature (e.g. the domain of `HAS.horn` is animals) and completely inappropriate features (e.g feature `dig` for *csűr* 'barn' in the buildings group) were not differentiated: they were all simply marked wrong. Inter-annotator agreement was found to be substantial (Cohen's kappa=0.734). Results of the evaluation are shown in Table 4[3]. The table shows semantic feature accuracy (acc: the ratio of correctly assigned features) in each category for each model. We also automatically computed feature "domain accuracy" (d-acc): here we ignored feature assignment errors where the same feature was marked adequate for another test word in the same domain. The table also shows the number of different features (#F) each model assigned to the test words in each domain, and the number of features that were marked wrong for *any* of the test words in the given domain (#B). The overall feature accuracy of the 4lang-derived models was nearly 75%, while the combined ROLD model achieved over 80% feature accuracy. The feature space of the ROLD model is less fine-grained in some domains (e.g.food or clothing) than that derived from 4lang definitions (this is indicated by the lower number of different features assigned by the ROLD model) and this results in higher accuracy. Note that the domain accuracy of 4lang features is much higher than feature accuracy, it is about 90%. The worst average accuracy was obtained on colors: lists of things having specific colors or patterns and the high number of color terms themselves generated too much noise.

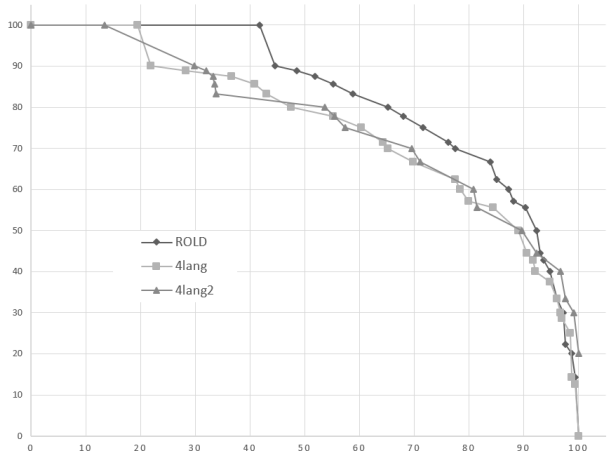


Figure 2: The distribution of feature precision for the three models ROLD, 4lang and 4lang2.

Figure 2 shows the distribution of the precision of features per word. The ROLD model assigned only appropriate features to 42% of the 290 test words, and precision was over 70% for over 75% of the words. 4lang and 4lang2 had 100% precision for 20% and 13.4% of the test words, respectively. All models had over 50% precision for about 90% of the words. The precision of 4lang2 was over 20% for all test words.

6.3. Proper Names and Non-standard Language

The WE models our method is based on also reflect world knowledge as represented in the corpus from which they are generated from. This enables our model to assign features to proper names of various types, such as names of people, institutions, fictional creatures, or even abbreviations as shown in Table 5. In the names section of the table, some famous people are shown, one of them is Hungarian (*Béla Bartók*, a Hungarian composer). It can be seen

[3]Due to length limits, we included only selected categories in the extended abstract.

	4lang					4lang2					ROLD				
Group name	acc	d-acc	#F	#B	most frq. features	acc	d-acc	#F	#B	most frq. features	acc	d-acc	#F	#B	most frq. features
Units of measure	62.22	74.44	55	26	unit measure	63.64	74.75	59	27	unit measure	70.00	81.67	34	13	Measurement Computers
Electronics	86.36	90.91	38	7	machine device	82.00	88.00	43	13	equipment machine	78.26	85.51	35	13	Technology Recording
Diseases	88.71	94.62	37	13	bad body	87.88	95.45	46	10	bad bad(situation)	94.23	98.08	24	2	Illness Disease_N
Animals	69.23	94.87	35	20	wild animal	67.19	94.79	35	18	animal HAS.wings	84.21	93.86	24	9	Animal_N Animals
Kitchen utensils	79.49	91.03	31	14	instrument contain	76.00	89.00	39	18	food.IN metal	92.31	92.31	24	3	Receptacle_N Daily_life
Food	57.14	95.24	26	14	food COOK	64.52	96.77	23	10	food material	97.44	97.44	7	1	Food Food_N
Vehicles	84.71	95.29	34	10	vehicle engine	74.55	92.73	45	15	vehicle ‹HAS.engine›	75.00	88.16	27	7	Journey_N Vehicle_N
Clothes	68.35	96.20	16	6	WEAR garment	71.00	87.00	34	14	garment cloth	100.00	100.00	12	0	Clothing_N Clothes
Disciplines	77.91	86.05	51	14	science educate	76.00	87.00	52	15	science knowledge	88.24	92.94	48	10	Education Knowledge_N
Water	87.64	98.88	24	7	water valley	84.00	99.00	29	8	land ON.earth	98.25	100.00	18	1	Geography Geology
Geographic areas	83.91	96.55	31	8	land valley	76.53	96.94	29	10	land natural	87.14	94.29	27	7	Geography Geology
Natural events	85.37	91.46	50	11	wind atmosphere	85.86	91.92	57	13	weather water	72.29	80.72	45	19	Meteorology Nature
Mountains, hills	79.89	89.66	32	11	hill land	80.30	92.42	27	13	hill mountain	93.86	98.25	17	4	Geography Nature
Cities	85.29	94.12	27	6	city place	82.61	92.75	25	8	city place	87.84	90.54	40	8	Abode_N Geography
Locations	85.94	93.75	31	7	country land	87.93	93.10	22	5	country land	81.54	86.15	41	10	Geography Government_N
Buildings	76.47	85.29	30	12	building place	86.00	92.00	39	11	place building	87.50	94.44	30	9	Abode_N Buildings
Groups of humans	90.00	96.47	55	9	group purpose	85.57	90.21	52	15	institution structure	85.71	92.86	55	14	Organizations Receptacle_N
Human relationship	74.73	95.60	43	14	=POSS KNOW	78.00	98.00	39	13	HAS.parent companion	82.35	88.24	48	12	Auxiliary_N Friend_N
Athletes	59.09	81.82	34	19	PLAY game	59.00	74.00	44	19	sport person	71.95	84.15	27	14	Strength_N Other_sports
Occupations	71.76	76.47	52	21	profession science	75.00	80.00	51	20	profession person	84.38	90.63	39	10	Occupations Scholar_N
Time	79.41	86.76	42	13	sunset monday	57.50	71.25	50	29	period ON.earth	46.43	69.64	31	21	Chronology Celebration_N
Events	60.53	76.32	39	20	invite holiday	76.53	92.86	35	16	FOR.pleasure period	81.40	94.19	35	14	Amusement_N Leisure
Colors	45.98	89.66	20	11	colour shade	40.86	84.95	18	14	colour light	53.33	68.33	28	16	Colours Color_Adj
Attributes of humans	67.76	95.39	48	18	stupid good	75.26	93.16	60	22	strange heavy	85.47	88.83	100	24	Love_Adj Badness_Adj
Attributes of food	71.43	86.81	53	16	taste COOK	82.65	89.80	46	10	material sweet	100.00	100.00	23	0	Food Food_dish
Verbs of movement	71.25	88.75	24	10	rush long	52.00	61.00	29	19	go rush	46.39	61.86	49	30	Velocity_Vb Journey_Vb
Verbs of free-time	50.59	75.29	29	21	relax <person>	60.00	77.00	46	24	relax lack(work)	59.79	75.26	61	31	Outdoor Endearment_Vb
Verbs of decay	69.88	84.34	43	16	after CAUSE	55.67	74.23	54	26	slip die	68.48	75.00	69	25	Death_Vb Chemistry
All	**74.74**	**90.07**	**564**	**266**		**73.86**	**88.34**	**584**	**295**		**80.36**	**87.93**	**561**	**252**	

Table 4: Performance of the models 4lang, 4lang2 and ROLD on test words from different semantic groups. **acc**: feature accuracy, **d-acc**: domain accuracy of features, **#F**: different features, **#B**: features marked wrong at least once.

Bartók	4L: music art *poem *poet *poetry WRITE sound *text musician 4L2: art *poem *poet music HAS.rhythm entertainment sound sequence *text MAKE.beautiful RL: Music Music_N Performing
Obama	4L: country government politician @United_States state LEAD *place president republic 4L2: country politician @United_States country.HAS place MAKE.law state *@Soviet_Union politics RL: Officials Government_N Government Politics_N Authority_N Director_N Council_N
MTA	4L: institution group society *president *republic educate science purpose *person people 4L2: institution society group educate science HAS.purpose study structure people RL: Occupations Education *Receptacle_N College *Geology Skill_N Organizations
ELTE	4L: educate institution study student degree science numbers atom *GIVE 4L2: educate institution study science *name *part knowledge public *system RL: College Education Knowledge_N School_Adj Language_N
PPKE	4L: educate institution science group study student degree society *sleeve @Catholic_Church 4L2: educate study institution science knowledge group religion *system job HAS.purpose RL: College School_Adj Education Occupations School

Table 5: Examples of features returned for proper names and abbreviations of names of institutions from the models

that each person is assigned features that provide information about them. Thus, the model can be queried even for names one is not familiar with, and relevant features will be provided. This also holds for names with lower frequency in the corpus, as long as the name itself is unique.

Table 5 also contains the abbreviated name of some organizations. *ELTE* is for Eötvös Loránd University, while *PPKE* is for Pázmány Péter Catholic University. While both of them are educational institutions, *ELTE* is a state university, but *PPKE* is catholic, and this difference is reflected by the set of features assigned to them in addition to their relation to science and education.

The same applies to slang terms, including many short diminutive forms. These are abundant in the web-crawled corpus, mainly coming from often heated discussions in user comments and fora, and many of them have strong emotional connotations. These are neatly reflected by the semantic tags assigned to them in addition to the ones reflecting the basic meaning of the term, e.g. 'Deceiver', 'Obstinacy', 'Ignorance', 'Thief', 'Crime', 'Politics' 'Race relations' 'Psychology, Psychiatry', 'stupid', 'criminal' in addition to 'person' for derogative terms like *nyugger* 'pensioner', *proli* 'proletarian', *bolsi* 'bolshevik' or *cigó* 'Gypsy'.

7. Conclusions

We have shown that the meaning implicitly represented in word embedding models can be transformed into a set of symbolic features that can be used as semantic annotation. This can also be done across languages, thus relevant semantic tags can be assigned to words in a language that lacks appropriate semantic resources. Despite its simplicity, our system, the Thing Recognizer, performs this surprisingly efficiently also for names and words that cannot be expected to be included in manually created lexical semantic resources.

Acknowledgments

This research has been implemented with support provided by grants FK125217 and PD125216 of the National Research, Development and Innovation Office of Hungary financed under the FK17 and PD17 funding schemes.

8. Bibliographical References

Agirre, E., Martínez, D., de Lacalle, O. L., and Soroa, A. (2006). Evaluating and optimizing the parameters of an unsupervised graph-based wsd algorithm. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, TextGraphs-1, pages 89–96, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bartunov, S., Kondrashkin, D., Osokin, A., and Vetrov, D. (2015). Breaking sticks and ambiguities with adaptive skip-gram. *arXiv preprint arXiv:1502.07257*.

Bordes, A., Weston, J., Collobert, R., and Bengio, Y. (2011). Learning structured embeddings of knowledge bases. In *AAAI*.

Bordes, A., Glorot, X., Weston, J., and Bengio, Y. (2012). Joint learning of words and meaning representations for open-text semantic parsing. In *In Proceedings of 15th International Conference on Artificial Intelligence and Statistics*.

Brown, S. W. (2008). Choosing sense distinctions for wsd: Psycholinguistic evidence. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT-Short '08, pages 249–252, Stroudsburg, PA, USA. Association for Computational Linguistics.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2015). A unified multilingual semantic representation of concepts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 741–751, Beijing, China, July. Association for Computational Linguistics.

Chapman, R. (1977). *Roget's International Thesaurus*. Harper Colophon Books. Crowell.

Chen, X., Liu, Z., and Sun, M. (2014). A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1025–1035, Doha, Qatar, October. Association for Computational Linguistics.

Christiane Fellbaum, editor. (1998). *WordNet: an electronic lexical database*. MIT Press.

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 873–882, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kornai, A., Ács, J., Makrai, M., Nemeskey, D. M., Pajkossy, K., and Recski, G. (2015). Competence in lexical semantics. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 165–175, Denver, Colorado, June. Association for Computational Linguistics.

Labutov, I. and Lipson, H. (2013). Re-embedding words. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 489–493, Sofia, Bulgaria, August. Association for Computational Linguistics.

Li, J. and Jurafsky, D. (2015). Do multi-sense embeddings improve natural language understanding? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1722–1732, Lisbon, Portugal, September. Association for Computational Linguistics.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Bilingual word representations with monolingual quality in mind. In *NAACL Workshop on Vector Space Modeling for NLP*, Denver, United States.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013a). Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Mikolov, T., Yih, W., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751.

Miller, G. A. (1995). Wordnet: A lexical database for English. *COMMUNICATIONS OF THE ACM*, 38:39–41.

Neelakantan, A., Shankar, J., Passos, A., and McCallum, A. (2014). Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar, October. Association for Computational Linguistics.

Novák, A., Siklósi, B., and Oravecz, C. (2016). A new integrated open-source morphological analyzer for Hungarian. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Novák, A. (2014). A new form of Humor – mapping constraint-based computational morphologies to a finite-state representation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Orosz, Gy. and Novák, A. (2013). PurePos 2.0: a hybrid tool for morphological disambiguation. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, pages 539–545, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Panchenko, A. (2016). Best of both worlds: Making word sense embeddings interpretable. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Rothe, S. and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1793–1803, Beijing, China, July. Association for Computational Linguistics.

Siklósi, B. (2016). Using embedding models for lexical categorization in morphologically rich languages. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing: 17th International Conference, CICLing 2016*, Konya, Turkey, April. Springer International Publishing, Cham.

Summers, D. (2005). *Longman Dictionary of Contemporary English*. Longman Dictionary of Contemporary English Series. Longman.

Tian, F., Dai, H., Bian, J., Gao, B., Zhang, R., Chen, E., and Liu, T.-Y. (2014). A probabilistic model for learning multi-prototype word embeddings. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 151–160, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yu, M. and Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 545–550, Baltimore, Maryland, June. Association for Computational Linguistics.

Advances in Pre-Training Distributed Word Representations

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, Armand Joulin
Facebook AI Research
{tmikolov, egrave, bojanowski, cpuhrsch, ajoulin}@fb.com

Abstract

Many Natural Language Processing applications nowadays rely on pre-trained word representations estimated from large text corpora such as news collections, Wikipedia and Web Crawl. In this paper, we show how to train high-quality word vector representations by using a combination of known tricks that are however rarely used together. The main result of our work is the new set of publicly available pre-trained models that outperform the current state of the art by a large margin on a number of tasks.

Keywords: fastText, word2vec, word vectors, pre-trained

1. Introduction

Pre-trained continuous word representations have become basic building blocks of many Natural Language Processing (NLP) and Machine Learning applications. These pre-trained representations provide distributional information about words, that typically improve the generalization of models learned on limited amount of data (Collobert et al., 2011). This information is typically derived from statistics gathered from large unlabeled corpus of text data (Deerwester et al., 1990). A critical aspect of their training is thus to capture efficiently as much statistical information as possible from rich and vast sources of data.

A standard approach for learning word representations is to train log-bilinear models based on either the skip-gram or the continuous bag-of-words (cbow) architectures, as implemented in *word2vec* (Mikolov et al., 2013a) and *fastText* (Bojanowski et al., 2017)[1]. In the skip-gram model, nearby words are predicted given a source word, while in the cbow model, the source word is predicted according to its context. These architectures and their implementation have been optimized to produce high quality word representations able to transfer to many tasks, while maintaining a sufficiently high training speed to scale to massive amount of data.

Recently, word2vec representations have been widely used in NLP pipelines to improve their performance. Their impressive capability at transfering to new problems suggests that they are capturing important statistics about the training corpora (Baroni and Lenci, 2010). As can be expected, the more data a model is trained on, the better the representations are at transferring to other NLP problems. Training such models on massive data sources, like Common Crawl, can be cumbersome and many NLP practitioners prefer to use publicly available pre-trained word vectors over training the models by themselves. In this work, we provide new pre-trained word vectors that show consistent improvement over the currently available ones, making them potentially very useful to a wide community of researchers.

We show that several modifications of the standard word2vec training pipeline significantly improves the quality of the resulting word vectors. We focus mainly on known modifications and data pre-processing strategies that are rarely used together: the position dependent features introduced by Mnih and Kavukcuoglu (2013), the phrase representations used in Mikolov et al. (2013b) and the use of subword information (Bojanowski et al., 2017).

We measure their quality on standard benchmarks: syntactic, semantic and phrase-based analogies (Mikolov et al., 2013b), rare words dataset (Luong et al., 2013), and as features in a question answering pipeline on Squad question answering dataset (Rajpurkar et al., 2016; Chen et al., 2017).

[1] https://fasttext.cc/

2. Model Description

In this section, we briefly describe the cbow model as it was used in word2vec, and then explain several known improvements to learn richer word representations.

2.1. Standard cbow model

The cbow model as used in Mikolov et al. (2013a) learns word representations by predicting a word according to its context. The context is defined as a symmetric window containing all the surrounding words. More precisely, given a sequence of T words $w_1, \ldots, w_T$, the objective of the cbow model is to maximize the log-likelihood of the probability of the words given their surrounding, i.e.:

$$\sum_{t=1}^{T} \log\ p\left(w_t \mid C_t\right), \tag{1}$$

where C_t is the context of the t-th word, e.g., the words $w_{t-c}, \ldots w_{t-1}, w_{t+1}, \ldots, w_{t+c}$ for a context window of size $2c$. For now on, we assume that we have access to a scoring function between a word w and its context C, denoted by $s(w, C)$. This scoring function will be later parametrized by the word vectors, or representations. A natural candidate for the conditional probability in Eq. 1 is a softmax function over the scores of a context and words in the vocabulary. This choice is however impractical for large vocabulary. An alternative is to replace this probability by independent binary classifiers over words. The correct word is learned in contrast with a set of sampled negative candidates. More precisely, the conditional probability of a word w given its context C in Eq. (1) is replaced

by the following quantity:

$$\log\left(1+e^{-s(w,C)}\right)+\sum_{n\in N_C}\log\left(1+e^{s(n,C)}\right), \qquad (2)$$

where N_C is a set of negative examples sampled from the vocabulary. The objective function maximized by the cbow model is obtained by replacing the log probability in Eq. (1) by the quantity defined in Eq. (2), i.e.:

$$\sum_{t=1}^{T}\left[\log\left(1+e^{-s(w_t,C_t)}\right)+\sum_{n\in N_{C_t}}\log\left(1+e^{s(n,C_t)}\right)\right].$$

A natural parametrization for this model is to represent each word w by a vector v_w. Similarly, a context is represented by the average of word vectors $u_{w'}$ of each word w' in its window. The scoring function is simply the dot product between these two quantities, i.e.,

$$s(w,C)=\frac{1}{|C|}\sum_{w'\in C}u_{w'}^T v_w. \qquad (3)$$

Note that different parametrizations are used for the words in a context and the predicted word.

Word subsampling. The word frequency distribution in a standard text corpus follows a Zipf distribution, which implies that most of the words belongs to small subset of the entire vocabulary (Li, 1992). Considering all the occurences of words equally would lead to overfit the parameters of the model on the representation of the most frequent words, while underfitting on the rest. A common strategy introduced in Mikolov et al. (2013a) is to subsample frequent words, with the following probability p_{disc} of discarding a word:

$$p_{disc}(w)=1-\sqrt{t/f_w} \qquad (4)$$

where f_w is the frequency of the word w, and $t>0$ is a parameter.

2.2. Position-dependent Weighting

The context vector described above is simply the average of the word vectors contained in it. This representation is oblivious to the position of each word. Explicitly encoding a representation for both a word and its position would be impractical and prone to overfitting. A simple yet effective solution introduced in the context of word representation by Mnih and Kavukcuoglu (2013) is to learn position representations and use them to reweight the word vectors. This position dependent weighting offers a richer context representation at a minimal computational cost.

Each position p in a context window is associated with a vector d_p. The context vector is then the average of the context words reweighted by their position vectors. More precisely, denoting by P the set of relative positions $[-c,\ldots,-1,1,\ldots,c]$ in the context window, the context vector v_C of the word w_t is:

$$v_C=\sum_{p\in P}d_p\odot u_{t+p}, \qquad (5)$$

where $\odot$ is the pointwise multiplication of vectors.

2.3. Phrase representations

The original cbow model is only based on unigrams, which is insensitive to the word order. We enrich this model with word n-grams to capture richer information. Directly incorporating the n-grams in the models is quite challenging as it clutters the models with uninformative content due to huge increase of the number of the parameters. Instead, we follow the approach of Mikolov et al. (2013b) where n-grams are selected by iteratively applying a mutual information criterion to bigrams. Then, in a data pre-processing step we merge the words in a selected n-gram into a single token. For example, words with high mutual information like "New York" are merged in a bigram token, "New_York". This pre-processing step is repeated several times to form longer n-gram tokens, like "New_York_City" or "New_York_University". In practice, we repeat this process $5-6$ times to build tokens representing longer ngrams. We used the *word2phrase* tool from the word2vec project[2]. Note that unigrams with high mutual information are merged only with a probability of 50%, thus we still keep significant number of unigram occurrences. Interstingly, even if the phrase representations are not further used in an application, they effectively improve the quality of the word vectors, as is shown in the experimental section.

2.4. Subword information

Standard word vectors ignore word internal structure that contains rich information. This information could be useful for computing representations of rare or mispelled words, as well as for mophologically rich languages like Finnish or Turkish. A simple yet effective approach is to enrich the word vectors with a bag of character n-gram vectors that is either derived from the singular value decomposition of the co-occurence matrix (Schütze, 1993) or directly learned from a large corpus of data (Bojanowski et al., 2017). In the latter, each word is decomposed into its character n-grams N and each n-gram n is represented by a vector x_n. The word vector is then simply the sum of both representations, i.e.:

$$v_w+\frac{1}{|N|}\sum_{n\in N}x_n. \qquad (6)$$

In practice, the set of n-grams N is restricted to the n-grams with 3 to 6 characters. Storing all of these additional vectors is memory demanding. We use the hashing trick to circumvent this issue (Weinberger et al., 2009).

3. Training Data

We used several sources of text data that are publicly available and the Gigaword dataset, as described in Table 1. In particular, we used English Wikipedia from June 2017, from which we used the meta pages archive which resulted in a text corpus with more than 9 billion words [3]. Further, we used all news datasets from `statmt.org` from years 2007 - 2016, the UMBC corpus (Han et al., 2013), the English Gigaword, and Common Crawl from May 2017[4].

[2] `https://github.com/tmikolov/word2vec`

[3] `https://dumps.wikimedia.org/enwiki/latest/`

[4] `https://commoncrawl.org/2017/06`

In case of the Common Crawl, we wrote a simple data extractor based on a unigram language model that retrieves the documents written in English and discards low quality data. The same approach can be in fact used to extract text data for many other languages from Common Crawl.
We decided to perform no complex data normalization or pre-processing, as we want the resulting word vectors to be very easily used by a wide community (the text normalization can be done on top of the published word vectors as a post-processing step). We only used a publicly available $tokenizer.perl$ script from the Moses MT project[5]. We observed that de-duplicating large text training corpora, especially Common Crawl, significantly improves the quality of the resulting word vectors.

Corpus	Size [billion]
Wikipedia meta-pages	9.2
Statmt.org News	4.2
UMBC News	3.2
Gigaword	3.3
Common Crawl	630

Table 1: Training corpora and their size in billions of words after tokenization and sentence de-duplication.

4. Results

Further we report results for models trained on either the Common Crawl, or on a combination of the Wikipedia, Statmt News, UMBC and Gigaword. This is comparable to corpora that other models that attempted to improve upon word2vec were trained on, notably the GloVe model from the Stanford NLP group (Pennington et al., 2014). Although a careful analysis performed in Levy et al. (2015) shows that the original word2vec is faster to train, produces more accurate models and takes significantly less memory than the GloVe algorithm, the availability of large pre-trained GloVe models proved to be a useful resource for many researchers who do not have time to train their own model on very large dataset like the Common Crawl.
We used the cbow architecture described in Section 2.1. with window size 5 for the baseline models and window size 15 for the models that learn position-dependent weights (described in Section 2.2.). We used 10 negative examples for training with the negative sampling and threshold for subsampling frequent words set to $t = 10^{-5}$.
In Table 2 we report results on the word analogies from Mikolov et al. (2013a) using baseline cbow model trained on Common Crawl with de-duplicated sentences, with phrases (we used 6 iterations of building the phrases by merging bigrams with high mutual information), and with the position-dependent weighting as used in Mnih and Kavukcuoglu (2013). The training itself took three days on a single multi-core machine.
In Table 3, we can see comparison between cbow as implemented in the fastText library (Bojanowski et al., 2017) and the GloVe models trained on comparable corpora. The

[5] `https://github.com/moses-smt`

Model	Sem	Syn	Tot
cbow + uniq	79	73	76
cbow + uniq + phrases	82	78	80
cbow + uniq + phrases + weighting	87	82	85

Table 2: Accuracies on semantic and syntactic analogy datasets for models trained on Common Crawl (630B words). By performing sentence-level de-duplication, adding position-dependent weighting and phrases, the model quality improves significantly.

87% accuracy in the word analogy tasks is to our knowledge the best published result so far by a large margin. We improved this result further to 88.5% accuracy by adding the sub-word features. We also report state of the art performance on the Rare Words dataset (Luong et al., 2013). Finally, we show that the new fastText vectors can be very useful in some popular Question Anwering tasks such as Squad (Rajpurkar et al., 2016). In a setup that is further described in Chen et al. (2017), we did observe significant improvement of the accuracy by using the new pre-trained models.

Model	Analogy	RW	Squad
GloVe Wiki + news	72	0.38	77.7%
fastText Wiki + news	**87**	0.50	78.8%
GloVe Crawl	75	0.52	78.9%
fastText Crawl	85	**0.58**	**79.8%**

Table 3: Results on Word Analogy, Rare Words and Squad datasets with fastText models trained on various corpora (see Table 1) or Common Crawl (see Table 2), and comparison to GloVe models trained on comparable datasets.

The models trained on the Wikipedia and News corpora, and on the Common Crawl, were published at the `fasttext.cc` website and are available to the NLP researchers. Further, we did experiment with the phrase-based analogy dataset introduced in Mikolov et al. (2013b), and achieved 88% accuracy using the model trained on Crawl, which again is to our knowledge the new state of the art result. We plan to release the model containing all the phrases in the near future.
Finally in Table 4, we use a script provided by Conneau et al. (2017) to measure the influence of different pre-trained word vector models on several text classification tasks (MRPC, MR CR, SUBJ, MPQA, SST and TREC). This is equivalent to running the fastText library in a supervised mode without finetuning the word representations (Joulin et al., 2016). Overall, the new fastText word vectors result in superior text classification performance.

5. Discussion

In this work, we have focused on providing very high quality set of pre-trained word and phrase vector representations. Our findings indicate that improvements can be

	Corpora	MRPC	MR	CR	SUBJ	MPQA	SST	TREC	Average
GloVe	Wiki+news	71.9/81.0	75.7	78.1	91.5	86.9	78.1	66.6	79.7
GloVe	Crawl	72.0/80.7	78.0	79.6	91.8	88.0	80.0	84.2	82.0
fastText	Wiki+news	72.9/**81.6**	77.8	80.3	92.2	**88.3**	81.1	**85.0**	82.5
fastText	Crawl	**73.4**/**81.6**	**78.2**	**81.1**	**92.5**	87.8	**82.0**	84.0	**82.7**

Table 4: Comparison of different pre-trained models on supervised text classification tasks.

achieved by training well-known algorithms on very large text datasets, and that using certain tricks can provide further gains in quality. Notably, we have found it very important to de-duplicate sentences in large corpora such as the Common Crawl before training the models. Next, we have used an algorithm for building the phrases in a preprocessing step. Finally, adding the position-dependent weights and subword features to the cbow model architecture gave us the final boost of accuracy. The models described in this paper are freely available to researchers and engineers at the fastText webpage, and we hope that these will be useful in various projects that use textual data.

6. Acknowledgements

We thank Marco Baroni and German Kruszewski for useful discussions and suggestions, Adam Fisch for help with the experiments on Squad dataset, and Qun Luo for suggesting to use the position-dependent weighting at the word2vec discussion forum.

7. Bibliographical References

Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Han, L., Kashyap, A. L., Finin, T., Mayfield, J., and Weese, J. (2013). UMBC_EBIQUITY-CORE: Semantic Textual Similarity Systems. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, June.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Li, W. (1992). Random texts exhibit zipf's-law-like word frequency distribution. *IEEE Transactions on information theory*, 38(6):1842–1845.

Luong, T., Socher, R., and Manning, C. D. (2013). Better word representations with recursive neural networks for morphology. In *CoNLL*, pages 104–113.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mnih, A. and Kavukcuoglu, K. (2013). Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in neural information processing systems*, pages 2265–2273.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Schütze, H. (1993). Word space. In *Advances in neural information processing systems*, pages 895–902.

Weinberger, K., Dasgupta, A., Langford, J., Smola, A., and Attenberg, J. (2009). Feature hashing for large scale multitask learning. In *ICML*.

Integrating Generative Lexicon Event Structures into VerbNet

Susan Windisch Brown[1], James Pustejovsky[2], Annie Zaenen[3], Martha Palmer[1]
[1]University of Colorado, [2]Brandeis University, [3]Stanford University
{susan.brown, martha.palmer}@colorado.edu, jamesp@cs.brandeis.edu, azaenen@stanford.edu

Abstract

This paper focuses on specific changes to the semantic representations associated with classes of verbs in the English lexical resource VerbNet (Schuler, 2005). The new form has been restricted to first-order representations to simplify use by and integration with planners. More significantly, the modifications incorporate the Generative Lexicon's event structure, with temporal ordering of subevents associated with explicit predications over the verb's arguments. These changes allow for greater flexibility in representing complex events, for a more consistent treatment of the oppositions inherent in change-of-state classes, and for a more nuanced portrayal of the Agent's role.

Keywords: semantics, event representation, lexicon

1. Introduction

Natural language processing has been moving from shallow semantic parsing to deeper semantic analysis of text in order to better support knowledge representation and reasoning systems. Understanding sentences requires more than identifying events and participants and giving them thematic role labels. In particular, it is essential to recognize any temporal sequencing within the event and any changes in state that might have occurred (Pustejovsky, 2005; Mani and Pustejovsky, 2012).

The language resource VerbNet (Kipper et al., 2006) is a promising source of such information. It is a hierarchical, domain-independent verb lexicon that groups verbs into classes based on similarities in their syntactic and semantic behavior (Schuler, 2005). Each class in VerbNet defines a set of members, thematic roles for the predicate-argument structure of these members, selectional restrictions on the arguments, and frames consisting of a syntactic description and a corresponding semantic representation. It has long been used for semantic role labeling and other inference-enabling tasks (Shi and Mihalcea, 2005; Giuglea and Moschitti, 2006; Loper et al., 2007). Automatic disambiguation of a verb's VerbNet class has also improved (Abend et al., 2008; Brown et al., 2014; Kawahara and Palmer, 2014). Efforts to use its semantic representations (Zaenen et al., 2008; Narayan-Chen et al., 2017) have revealed a need to revise them for consistency and greater expressiveness, in particular, a clearer representation of subevents.

Recent work in Generative Lexicon (GL) has focused on further articulating the semantics of subevent structure in language (Pustejovsky and Moszkowicz, 2011; Pustejovsky, 2013). Hence a reasonable undertaking is to revise VerbNet to take advantage of GL's progress in representing subevent structure while preserving VerbNet's strengths in linking predicate argument structure, thematic roles and semantic representations.

The remainder of this paper will describe the changes being made to VerbNet's semantic representations and the reasons behind those changes. Section 2 briefly describes the role semantic representations play in VerbNet and breaks down the structure of the "old" semantic representations in version 3.3. Section 3 goes into more detail on the drawbacks of these representations and the requests from users for new functionality. An overview of GL event structure is given in section 4, highlighting how it can fulfill the needs identified in section 3. Section 5 uses VerbNet change of location and change of state classes to illustrate the new representations to be released in VerbNet version 4.0.

2. VerbNet and Its Representation of Events

Each VerbNet class contains semantic representations that are compatible with the member verbs and the syntactic frames of the class. This pairing of each syntactic frame in a class with a semantic representation is a unique feature of VerbNet that emphasizes the close interplay of syntax and semantics. The semantic information is expressed as a conjunction of semantic predicates, such as **motion, perceive** or **cause** and an event variable **E**. Some of these are meant to describe the participants during various stages of the event evoked by the syntactic frame. For example, one of the intransitive frames in the class Run-51.3.2 is shown in (1), with the final 4 lines making up the semantic representation:

(1) *The horse ran into the barn.*
NP V PP
Theme V Destination

motion(during(E), Theme)
path_rel(start(E), Theme, Initial_location, ch_of_loc, prep)
path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
path_rel(end(E), Theme, Destination, ch_of_loc, prep)

The arguments of each predicate are represented using the thematic roles for the class. These roles provide the link between the syntax and the semantic representation. Each participant mentioned in the syntax, as well as necessary but unmentioned participants, is accounted for in the semantics. For example, the second component of the first

path_rel semantic predicate above includes an unidentified Initial_location.[1] Temporal sequencing is indicated with the second-order predicates start, during and end, which are included as arguments of the appropriate first-order predicates. A similar sentence with an Agent causing the motion, such as *John herded the sheep into the barn,* would add **cause**(Agent, E) to the semantic representation in (1).
The semantic representations associated with a class capture generalizations about the semantic behavior of the member verbs as a group. For some classes, such as the Battle-36.4 class, the verbs are semantically quite coherent (e.g., *battle, skirmish, war*) and the semantic representation is correspondingly precise.

(2) *Sparta warred with Athens.*
NP V PP
Agent V {with} Co-Agent

social_interaction(during(E), Agent, Co-Agent)
conflict(during(E), Agent, Co-Agent)
possible_contact(during(E), Agent, Co-Agent)
manner(Hostile, Agent, Co-Agent)

Other classes, such as Other Change of State-45.4, contain widely diverse member verbs (e.g., *dry, gentrify, renew, whiten*). The representation must be very general to apply to all the verbs. The following representation for this class ignores the specific type of state change in the example sentence (i.e., from wet to dry) in order to be general enough for any verb in the class when used in a basic transitive sentence.

(3) *John dried the clothes.*
NP V NP
Agent V Patient

path_rel(start(E), Initial state, Patient, ch_of_state, prep)
path_rel(result(E), Result, Patient, ch_of_state, prep)
cause(Agent, E)

Additional, more precise semantic information can be associated with each individual verb in a class. To that end, verb-specific features that differentiate member verbs within a class are currently being added to certain classes (Pustejovsky et al., 2016). Our goal in the revisions described here, however, is to enhance the accuracy and expressiveness of a class's representations while preserving the generalizations that can be made across all the verbs in a class.

3. The Impetus for Change

Over the years, VerbNet has undergone several revisions, either to expand its coverage (Kipper et al., 2008), to improve the clarity and consistency of its components (Bonial et al., 2011; Hwang, 2014), or in response to users' needs. Minor changes had been made to VerbNet's semantic representations, such as adding new semantic predicates, consolidating similar predicates, and standardizing the types and number of arguments particular predicates take. Efforts to use the semantic predicates in various tasks revealed several weaknesses that led us to undertake this current revision. Zaenen et al. (2008) used VerbNet predicates to inform change of location inferences, such as inferring from the statement *The diplomat left Bhagdad* that before the event, the diplomat was in Bhagdad. This is exactly the sort of information VerbNet's semantic representations were designed to provide. However, such information was not consistently supplied across all of the classes that dealt with motion. For several motion classes, End(E) was given but not Start(E). Other classes that dealt with the change of location of participants relative to each other (e.g., gather, mix) did not include a motion predicate at all. Although they found that in many cases VerbNet did support adequate inferencing, its inconsistencies lessened its usefulness.
Several of the omissions that were found stemmed from the practice of only including syntactic arguments in the semantic representations, ignoring any possible adjuncts. Including more semantic information about subevents in the representation, whether that information is instantiated in the syntax or not, was an important desideratum for the new representations.
A recent effort to use VerbNet in human-computer interaction (Narayan-Chen et al., 2017) found that a few aspects of the semantic representation could be altered both to facilitate the interaction between the language parsing and the planning components of the system and to increase the expressiveness of the representation.
First, attempts to use VerbNet information in robotics applications showed the need for a switch to a first-order logic representation. In addition, more specific causal and temporal relations were desired. The current method of indicating causation, for example, simply had an Agent and the event variable E as arguments to a Cause predicate. This is somewhat misleading in that it could imply that the Agent causes all of (E), including whatever state exists at Start(E).
Second, the temporal sequencing of an event is sometimes more complex than what can be expressed with only Start(E), During(E) and End(E). For example, a 'throw' event involves a Theme in motion during the event and an Agent in contact with Theme, but only in contact for part of the period denoted by During(E). The ability to subdivide these three periods seemed in order. For some events of this nature, we attempted to show this sequence in version 3.3 by positing two events, E_0 and E_1, and temporally subdividing those with During and End:

(4) *Mary threw the ball.*
NP V NP
Agent V Theme

exert_force(during(E_0), Agent, Theme)
contact(end(E_0), Agent, Theme)
¬ **contact**(during(E_1), Agent, Theme)
motion(during(E_1), Theme)
cause(Agent, E_1)

[1] Each **path_rel** predicate also has a **prep** slot which is a placeholder for the information contributed by specific prepositions to be passed to inferencing for further semantic processing.

As we will see, GL event structure and temporal sequencing of subevents solves this problem more logically and transparently, while preserving the idea that this sentence describes only a single event E.

Finally, representing simultaneity of subevents or gaps in the temporal sequencing is difficult with the apparently continuous sequence of before, during and after. In a cutting event, for example, an Agent is performing an action that results in a change of state in the Patient. The end of the cutting action temporally meets the beginning of the final state of the Patient. But this is not always the case in a causal event. With *John dried the clothes,* John may be doing something, such as waving a hair drier in front of the clothes, that does continue until the final end state is reached. Or John may have hung the clothes outside on a clothes line, in which case, his action does not temporally meet the final "dry" end state. For the representation of a generic "dry" event, we do not want an assumption that the actions taking place During(E) are necessarily contiguous with End(E). Therefore, in addition to having a means of identifying more subevents, we would like to have a means of indicating the specific temporal relations between them.

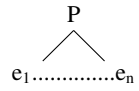
4. Generative Lexicon's Event Structure

Many of the issues described in section 2 are resolved by adopting aspects of the event structure as modeled in Generative Lexicon. Classic GL characterizes the different Aktionsarten in terms of structured subevents (Pustejovsky, 1995). Different event types can be represented as typed feature structures or in the form of tree structures, as below.

(5) a. STATE: a simple event, evaluated without referring to other events: *be sick*, *love*, *know*



b. PROCESS: a sequence of events identifying the same semantic expression: *run*, *push*, *drag*



c. TRANSITION: an event identifying a semantic expression evaluated with respect to its opposition: *give*, *open*; *build*:
Binary transition (achievement): $\neg\phi \in S_1$, and $\phi \in S_2$



Complex transition (accomplishment): $\neg\phi \in P$, and $\phi \in S$



The basic event types are the states and processes, which can represent independent events or be combined to derive complex events (transitions). Subevents within an event are ordered by **temporal relations** and **relative prominence** or **headedness**. Regarding temporal relations, two subevent orderings are of relevance here. One subevent may precede the other in a strictly sequential relation $< \circ$ (Allen's "meet" relation (Allen, 1984)), with the first subevent leading to the second, as with causatives (e.g., *build*), inchoatives (e.g., *arrive*), and ditransitive transfer verbs (e.g., *give*). In transaction events such as *sell*, *buy*, and *marry* ('get married to'), both subevents overlap in time ($\circ$).

Notice that, unlike primitive predicates, subevents can be quantified in the logical form of the sentence, in the same way that arguments can be.

(6) a. The destroyer is sinking the boat.
$\exists e_1 \exists x \exists y$[sink_act$(e_1, x, y) \wedge$ destroyer$(x) \wedge$ boat(y)]

b. The destroyer sank the boat.
$\exists e_1 \exists e_2 \exists x \exists y$[sink_act$(e_1, x, y) \wedge$ destroyer$(x) \wedge$ boat$(y) \wedge$ sink_result$(e_2, y) \wedge e_1 < e_2$]

c. The boat sank.
$\exists e_2 \exists e_1 \exists y \exists x$[sink_result$(e_2, y) \wedge$ boat$(y) \wedge$ sink_act$(e_1, x, y) \wedge e_1 < e_2$]

The logical form of the causative (6b) differs from the inchoative (6c) only in the explicit identification of a specific causer.

In subsequent work within GL, event structure has been integrated with dynamic semantic models in order to represent the attribute modified in the course of the event (the location of the moving entity, the extent of a created or destroyed entity, etc.) as a sequence of states related to time points or intervals. This way, in addition to describing the event in terms of discrete phases, we identify what attribute is changing and how it is changing over the event. The resulting event structure representation is called a **Dynamic Event Model** (Pustejovsky and Moszkowicz, 2011; Pustejovsky, 2013). Starting with the view that subevents of a complex event can be modeled as a sequence of frames, a dynamic event model explicitly labels the transitions that move an event from frame to frame.[2] We believe that, in order to adequately model change, the VerbNet representation must track the change in the assignment of values to attributes in the unfolding of the event. This includes making explicit any *predicative opposition* denoted by the verb. For example, simple transitions (achievements) encode either an intrinsic predicate opposition (*die* encodes going from $\neg dead(e_1, x)$ to $dead(e_2, x)$), or a specified relational opposition (*arrive* encodes going from $\neg loc_at(e_1, x, y)$ to $loc_at(e_2, x, y)$). Creation predicates and accomplishments generally also encode predicate oppositions.

A dynamic approach to modeling updates makes a distinction between formulae, ϕ, and programs, π. A formula is interpreted as a classical propositional expression, with assignment of a truth value in a specific state in the model (Harel et al., 2001). For our purposes, a state is a set of propositions with assignments to variables at a specific frame. We can think of atomic programs as input/output

[2]The resulting structure is equivalent to a Labeled Transition System (van Benthem, 1991), and is consistent with the approach developed in (Fernando, 2009; Fernando, 2013).

relations, i.e., relations from states to states, and hence interpreted over an input/output state-state pairing (cf. (Naumann, 2001)).
The model encodes three kinds of representations: (i) predicative **content** of a frame; (ii) **programs** that move from frame to frame; and **tests** that must be satisfied for a program to apply. These include: pre-tests, while-tests, and result-tests.

5. VerbNet's New Semantic Representations

A Generative Lexicon-inspired subevent structure fulfills the need for greater expressiveness and clarity in VerbNet representations that we identified in section 3. In this section we will describe the global changes we are making and then present their application in a few of the more complex types of events in VerbNet.
The greatest change is switching from a tripartite division of the temporal span of any event to a system of numbered subevents, which can be increased or decreased to accommodate the complexity of the event. This change eliminates the second-order logic of Start(E), During(E) and End(E), which was necessary to ease the integration of the representations with a robot planning system. It also allows for more nuanced temporal relationships between the subevents, as described in section 3.
The default assumption in this new schema is that e_1 precedes e_2, which precedes e_3, and so on. When appropriate, however, more specific predicates can be used to specify other relationships, such as **meets**(e_2, e_3) to show that the end of e_2 meets the beginning of e_3, or **while**(e_2, e_3) to show that e_2 and e_3 are co-temporal. The latter can be seen in section 5.1 with the example of accompanied motion.
Another important change is the way in which causation is represented. Previously, the representation implied that an event as a whole was being caused by an Agent, using **cause**(Agent, E).

(7) *The lion tamer jumped the lions through the hoop.*
NP V NP PP
Agent V Theme Trajectory

motion(during(E), Theme)
path_rel(start(E), Theme, ?Initial_location, ch_of_loc, prep)
path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
path_rel(end(E), Theme, ?Destination, ch_of_loc, prep)
cause(Agent, E)

In the new version, we focus on one subevent as being the cause of another. Thus, something an agent does (e.g., **do**(e_2, Agent)) causes a state change or another event (e.g., **motion**(e_3, Theme)), which would be indicated with **cause**(e_2, e_3).

(8) *The lion tamer jumped the lion through the hoop.*
has_location(e_1, Theme, ?Initial_Location)
do(e_2, Agent)
motion(e_3, Theme, Trajectory)
cause(e_2, e_3)
has_location(e_4, Theme, ?Destination)

(See sections 4.1-4.3 for further examples.)
A more minor adjustment concerns the path_rel predicate, which was introduced earlier in the revision process to highlight the commonalities among different types of change events and to provide greater consistency in the existing VerbNet representations (Hwang, 2014). At the request of some users, we are substituting more specific predicates for the general **path_rel** predicate, such as **has_location**, **has_state** and **change_value**. This shifts some information that was included within the variables and constants of the **path_rel** predicate out to the new predicates themselves. We are however maintaining a common subevent pattern for change of location, change of possession and change of state events that closely mirrors that introduced by Hwang (2014).
Events that include some sort of change from one location to another or one state to another compose the majority of classes in VerbNet and include some of the more complex event types. Therefore, we will use examples from the change of state and change of location classes to illustrate the new VerbNet representations.

5.1. Change of Location

The Run-51.3.2 class is a typical change of location class, with such member verbs as *run, march,* and *gallop.* The most basic change of location semantic representation (9) begins with a state predicate **has_location**, with a subevent argument e_1, a Theme argument for the object in motion, and an Initial_location argument. The motion predicate is underspecified as to the manner of motion in order to be applicable to all 97 verbs in the class. A final **has_location** predicate indicates the Destination of the Theme at the end of the event. Not all of the thematic roles included in the representation are necessarily instantiated in the sentence. Any uninstantiated roles in a frame are preceded by a question mark, such as Initial_location and Trajectory in (9).

(9) *The rabbit hopped across the lawn.*
has_location(e_1, Theme, ?Initial_Location)
motion(e_2, Theme, Trajectory)
has_location(e_3, Theme, ?Destination)

This representation collapses the information in two semantic predicates in the old VerbNet representation (10): the **path_rel** predicate indicating the Trajectory of the Theme with the motion predicate.

(10) *The rabbit hopped across the lawn.*
motion(during(E), Theme)
path_rel(start(E), Theme, ?Initial_location, ch_of_loc, prep)
path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
path_rel(end(E), Theme, ?Destination, ch_of_loc, prep)

This pattern of an initial state followed by a transition to another state forms the basis for more complex events, such a caused change of location. The representation in (9) is augmented in (11) with both a DO and a CAUSE predicate.

(11) *The farmer herded the sheep into the meadow.*
has_location(e_1, Theme, ?Initial_Location)
do(e_2, Agent)
motion(e_3, Theme, ?Trajectory)
cause(e_2, e_3)
has_location(e_4, Theme, Destination)

For many classes, the causal action has more specific semantic components in common across the member verbs, and, therefore, the predicates can be more specific than the underspecified DO predicate in the Run-51.3.2 class. For example, the Push-12 class represents the Agent's action with **contact** and **exert_force** predicates (12).

(12) *John pushed the plate to the edge of the table.*
has_location(e_1, Theme, ?Initial_Location)
contact(e_2, Agent, Theme)
exert_force(e_2, Agent, Theme)
motion(e_3, Theme, ?Trajectory)
has_location(e_4, Theme, Destination)

This new version highlights several of the advantages we have been discussing in comparison to the old (13).

(13) *John pushed the plate to the edge of the table.*
cause(Agent, E)
contact(during(E), Agent, Theme)
exert_force(during(E), Agent, Theme)
path_rel(start(E), Theme, ?Initial_location, ch_of_loc, prep)
path_rel(during(E), Theme, Trajectory, ch_of_loc, prep)
path_rel(end(E), Theme, ?Destination, ch_of_loc, prep)
motion(during(E), Theme)

On a superficial level, the new representation is more transparent to human readers, with the starting and ending states indicated with has_location predicates, and the numbered subevents clearly stepping through the temporal sequence. More fundamentally, the numbered subevents allow us to divide what was previously grouped as During(E) into separate subevents, one involving the interaction of the Agent and Theme and one involving the motion of the Theme.
Finally, an example from the Accompany-51.7 class illustrates how the new schema represents accompanied motion (14).

(14) *Elena guided Frank through the building.*
has_location(e_1, Theme, ?Initial_Location)
has_location(e_2, Agent, ?Initial_Location)
motion(e_3, Agent, Trajectory)
motion(e_4, Theme, Trajectory)
has_location(e_5, Agent, ?Destination)
has_location(e_6, Theme, ?Destination)
while(e_3, e_4)

The predicate **while** allows us to indicate that both the Agent and Theme are in motion simultaneously.

5.2. Change of State

The representations for changes of state have two basic patterns, depending on whether the change is between absolute states or along a value continuum. The first is illustrated in (15), the representation for the Die-42.4 class.

(15) *John died.*
alive(e_1, Patient)
¬**alive**(e_2, Patient)

For less semantically coherent classes, such as the Other_cos-45.4 class, the type of state must be underspecified, as in (16). In that case, the opposition between the initial and the result states must be explicitly shown.

(16) *The balloon burst.*
has_state(e_1, Patient, Initial_State)
opposition(Initial_State, V_Result)
has_state(e_2, Patient, V_Result)

Like the underspecificity of the **do** predicate, **has_state** allows us to reference initial states and final states general enough to apply to all the verbs in a class. The **do** predicate is used in situations in which the Agent's action causes another subevent but we really can't determine what that action is without further context. In many of change of state classes, however, we can further identify the final state by extracting information from the verb itself. In (16), the verb 'burst' tells us the final state of the Patient. The same holds for the other verbs from the class, such as *dry, blacken* or *triple.* We have introduced V_Result both as an indicator that the semantic representation can be further refined in context using the lexical features of the specific verb and as a placeholder for that information.
V_Result also allows us to distinguish between the change of state introduced by the verb and a further change of state introduced by a resultative construction.

(17) *The clothes dried wrinkled.*
Theme V Result
has_state(e_1, Patient, Initial_State)
has_state(e_2, Patient, V_Result)
has_state(e_2, Patient, Result)
opposition(Initial_State, V_Result)
opposition(Initial_State, Result)

A second type of change of state involves a change along a scale, such as the events in the Calibratible_cos-45.6.1 class.

(18) *The price of oil rose by 500% from $5 to $25.*
has_val(e_1, Patient, Initial_State)
change_value(e_2, DIRECTION, Extent, Attribute, Patient)
has_val(e_3, Patient, Result)

The members of this class have verb-specific features, either increase (e.g., *rise*), decrease (e.g., *fall*) or fluctuate (e.g., *vary*). DIRECTION, one of the arguments of **change_value**, is a variable whose value can be found in context from the particular verb's verb-specific feature.

6. Conclusion

This paper has focused on specific changes to the semantic representations associated with classes of verbs in VerbNet. We have restricted the representation language to first-order representations to simplify use by and integration with planners. A larger change has been modifications to incorporate GL's event structure, with temporal ordering of subevents associated with explicit predications over the verb's arguments. This allows for greater flexibility in representing complex events, for a more consistent treatment of the oppositions inherent in change-of-state classes, and for a more nuanced portrayal of the Agent's role.

7. Bibliographical References

Abend, O., Reichart, R., and Rappoport, A. (2008). A supervised algorithm for verb disambiguation into verbnet classes. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 9–16. Association for Computational Linguistics.

Allen, J. (1984). Towards a general theory of action and time. *Arificial Intelligence*, 23:123–154.

Bonial, C., Corvey, W., Palmer, M., Petukhova, V. V., and Bunt, H. (2011). A hierarchical unification of lirics and verbnet semantic roles. In *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on*, pages 483–489. IEEE.

Brown, S. W., Dligach, D., and Palmer, M. (2014). Verbnet class assignment as a wsd task. In *Computing Meaning*, pages 203–216. Springer.

Fernando, T. (2009). Situations in ltl as strings. *Information and Computation*, 207(10):980–999.

Fernando, T. (2013). Segmenting temporal intervals for tense and aspect. In *The 13th Meeting on the Mathematics of Language*, page 30.

Giuglea, A.-M. and Moschitti, A. (2006). Semantic role labeling via framenet, verbnet and propbank. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 929–936. Association for Computational Linguistics.

Harel, D., Kozen, D., and Tiuryn, J. (2001). Dynamic logic. In *Handbook of philosophical logic*, pages 99–217. Springer.

Hwang, J. D. (2014). *Identification and representation of caused motion constructions*. Ph.D. thesis, University of Colorado at Boulder.

Kawahara, D. and Palmer, M. (2014). Single classifier approach for verb sense disambiguation based on generalized features. In *LREC*, pages 4210–4213.

Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2008). A large-scale classification of english verbs. *Language Resources and Evaluation*, 42(1):21–40.

Loper, E., Yi, S.-T., and Palmer, M. (2007). Combining lexical resources: mapping between propbank and verbnet. In *Proceedings of the 7th International Workshop on Computational Linguistics, Tilburg, the Netherlands*.

Mani, I. and Pustejovsky, J. (2012). *Interpreting motion: Grounded representations for spatial language*. Number 5. Oxford University Press.

Narayan-Chen, A., Graber, C., Das, M., Islam, M. R., Dan, S., Natarajan, S., Doppa, J. R., Hockenmaier, J., Palmer, M., and Roth, D. (2017). Towards problem solving agents that communicate and learn. In *Proceedings of the First Workshop on Language Grounding for Robotics*, pages 95–103.

Naumann, R. (2001). Aspects of changes: a dynamic event semantics. *Journal of semantics*, 18(1):27–81.

Pustejovsky, J. and Moszkowicz, J. (2011). The qualitative spatial dynamics of motion. *The Journal of Spatial Cognition and Computation*.

Pustejovsky, J., Palmer, M., Zaenen, A., and Brown, S. (2016). Verb meaning in context: Integrating verbnet and gl predicative structures. In *Proceedings of the LREC 2016 Workshop: ISA-12, Potoroz, Slovenia*, volume 2016.

Pustejovsky, J. (1995). *The Generative Lexicon*. Bradford Book. Mit Press.

Pustejovsky, J. (2005). Generative lexicon and type theory. ESSLLI Summer School, 2005, Edinburgh, Scotland, August.

Pustejovsky, J. (2013). Dynamic event structure and habitat theory. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)*, pages 1–10. ACL.

Schuler, K. K. (2005). Verbnet: A broad-coverage, comprehensive verb lexicon.

Shi, L. and Mihalcea, R. (2005). Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *International conference on intelligent text processing and computational linguistics*, pages 100–111. Springer.

van Benthem, J. F. A. K. (1991). Logic and the flow of information.

Zaenen, A., Bobrow, D. G., and Condoravdi, C. (2008). The encoding of lexical implications in verbnet predicates of change of locations. In *LREC*.

8. Language Resource References

Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2006). Extensive classifications of english verbs. In *Proceedings of the 12th EURALEX International Congress*, Turin, Italy.

FontLex: A Typographical Lexicon based on Affective Associations

Tugba Kulahcioglu, Gerard de Melo
Rutgers University – New Brunswick
Department of Computer Science, Piscataway, NJ, USA
{tugba.kulahcioglu, gerard.demelo}@rutgers.edu

Abstract

The task of selecting suitable fonts for a given text is non-trivial, as tens of thousands of fonts are available, and the choice of font has been shown to affect the perception of the text as well as of the author or of the brand being advertized. Aiming to support the development of font recommendation tools, we create a typographical lexicon providing associations between words and fonts. We achieve this by means of affective evocations, making use of font–emotion and word–emotion relationships. For this purpose, we first determine font vectors for a set of ten emotion attributes, based on word similarities and antonymy information. We evaluate these associations through a user study via Mechanical Turk, which, for eight of the ten emotions, shows a strong user preference towards the fonts that are found to be congruent by our predicted data. Subsequently, this data is used to calculate font vectors for specific words, by relying on the emotion associations of a given word. This leads to a set of font associations for 6.4K words. We again evaluate the resulting dataset using Mechanical Turk, on 25 randomly sampled words. For the majority of these words, the responses indicate that fonts with strong associations are preferred, and for all except 2 words, fonts with weak associations are dispreferred. Finally, we further extend the dataset using synonyms of font attributes and emotion names. The resulting FontLex resource provides mappings between 6.7K words and 200 fonts.

Keywords: typography, font, emotion

1. Introduction

It is not uncommon for people to spend several minutes looking for the right font, but finally ending up using the default one (Fox, 2010). One would indeed be well-advised to spend some effort on typographic choices, as the choice of font has been shown to be able to affect the perception of the text as well as of the author (Juni and Gross, 2008; Shaikh, 2007b; Shaikh et al., 2007), or of associated products and brands (Fligner, 2013).

The task of selecting a suitable font is particularly burdensome for graphic designers, as their profession calls for such decisions to be made on a regular basis, and the mere use of a neutral font may negatively affect the perception of the related brand or product (Shaikh, 2007a; Shaikh, 2007b).

Even more severe than opting for a neutral font is to end up picking an ill-suited font typeface. This is commonly described as the font being *incongruent* with the underlying meaning or theme (e.g., writing the word "*happy*" with a font perceived as unhappy). As previous studies have revealed, the use of incongruent fonts not only increases the response time of users (Lewis and Walker, 1989; Hazlett et al., 2013), but can also have a particularly detrimental effect on the perception of the related product (Fligner, 2013; Childers and Jass, 2002; Van Rompay and Pruyn, 2011).

To make things worse, the task of font selection is becoming ever more challenging as the number of available fonts keeps increasing. Google Fonts[1] as of October 2017 provides a catalog of 822 font families, while broader font sharing websites[2] typically serve several thousands.

Despite the obvious need, the assistance offered by current tools remains very limited. Some websites (dafont.com, 2017; Bloch, 2017), and recently also word processing tools such as Microsoft Word, provide a categorized presentation of fonts for users to explore, based on visual attributes as well as also certain semantic ones. In addition to the exploratory approach, O'Donovan et al. (2014) present a method to recommend fonts that are semantically similar to the current font selection. In Qiao (2017), vector representations are used to generate font pairs. The visualization by Data Scope Analytics (2017) aims to help users in discovering aesthetically pleasing font pairs. Previous work, however, neglects the content of the text to be formatted, and in particular neglects the affective dimension of human perception.

Towards the aim of supporting the development of font recommendation tools based on the *textual content* and the associated *affect* of the message, in this study, we produce FontLex, a typographical lexicon that maps 6,721 words to a set of 200 fonts. As our main method, we rely on word–emotion and font–emotion associations, and connect words with fonts via their affective associations. This gives rise to 200-dimensional vector embeddings that capture the strength of the association between a given word with each of the 200 considered fonts. In an additional step, we also gather synonyms of the font attribute names from WordNet, which enables us to obtain font vectors for these words more directly from the font vectors of the related attribute.

The rest of this paper is organized as follows. First of all, Section 2. reviews related work on semantic attributes of fonts and on font recommendation techniques. Section 3. presents our method to predict emotion–font scores and evaluates it through a user study. Section 4. presents our method to predict word–font scores using the previously obtained emotion–font scores, and evaluates it through a further user study. Subsequently, Section 6. describes the semantic extension of the dataset using synonym relation-

[1] `https://fonts.google.com`

[2] For instance, `https://www.dafont.com/` and `http://www.1001fonts.com/`

ships. Finally, Section 7. concludes the paper and outlines plans for future work.

2. Related Work

We begin with a review of previous studies and tools that have approached the topic of semantic attributes of fonts or the goal of recommending fonts.

2.1. Semantic Attributes of Fonts

Through a crowdsourced study, O'Donovan et al. (2014) associate 200 fonts with 37 semantic attributes (e.g., happy). They ask users to pick one of two presented fonts for a given attribute, and then based on these selections assign scores between 0 and 100 for each font–attribute combination. The resulting dataset is publicly available[3] and will be discussed further in Section 3.

Kulahcioglu and de Melo (2018) extend the above crowdsourced dataset using deep Convolutional Neural Network (CNN) embeddings as a means of obtaining a similarity measure between fonts. To predict semantic attribute scores for a font outside the dataset, the authors take weighted averages of the nearest four font scores, as determined by the embeddings. Based on leave-one-out cross validation test results, the method is able to predict scores with around 9% mean absolute error.

In an online survey conducted by Shaikh et al. (2006), the characteristics of 20 fonts are assessed with respect to 15 adjective pairs (e.g., stable – unstable). The fonts are presented using alphabetic, numeral, and common symbols.

Further studies (Velasco et al., 2014; Velasco et al., 2015) analyze the relationship between visual font characteristics and taste attributes (sweet, sour, etc.) through user studies. They conclude that round fonts exhibit an association with sweet taste.

Finally, many font-focused websites (Sam Berlow and Sherman, 2017; dafont.com, 2017; Bloch, 2017) allow contributors to tag fonts with attributes, some of which are more semantic than visual.

2.2. Font Recommendation

O'Donovan et al. (2014) present a method of proposing fonts that are similar to a given font that is currently being used. In their experiments, they find that semantic attributes are more conducive to predicting the similarity of fonts than geometrical features. Thus, making use of a set of semantic attributes, they learn a font similarity metric based on crowdsourced comparisons, in which users need to assess which of two presented fonts is more similar to a provided reference font.

Wang et al. (2015) rely on a deep learning approach to find similar fonts. It is claimed that a qualitative comparison of both methods reveals this approach as producing better results than the former one by O'Donovan et al. (2014).

Using vector representations, Qiao (2017) aims to identify fonts that are both contrasting and complementary. The system can either propose a novel pair of fonts, or suggest a second font for an already specified one. The vector representations are provided online[4].

The force-directed graph visualization[5] developed by Data Scope Analytics (2017) displays 458 fonts and 1,807 co-usages gathered by Sam Berlow and Sherman (2017). The visualization[6] in Ho (2017) displays around 800 font embeddings mapped into a 2D space.

Several websites, including those of Sam Berlow and Sherman (2017), Canva.com (2017) and Mills (2017), provide font pair suggestions gathered from users or from other web sources.

2.3. Impact of Font Choices

A number of Stroop-style studies have been conducted to investigate the effect of font characteristics on perception. Hazlett et al. (2013) asked users to judge whether a displayed word is positive or negative, comparing 5 fonts and 25 words that are all strongly associated with positive or negative emotion. The results indicate that congruent typefaces yield faster responses. Lewis and Walker (1989) ask users to press a left hand key if the words *slow* or *heavy* appear, versus a right hand one if *fast* or *light* appears. In a second experiment, they display related words (e.g., *fox*) instead of the original words (e.g., *fast*) to ensure that the user needs to grasp the meaning of the displayed word. In both experiments, they repeat the tasks with congruent and incongruent fonts, finding that the former significantly reduce the response time.

In terms of survey-style studies, Juni and Gross (2008) present newspaper articles using two different fonts. Their survey reveals that the same text is perceived as more humorous or angry when read in a certain font compared to another. Shaikh (2007b) presents documents to participants using congruent, incongruent, and neutral fonts, while soliciting ratings to assess the perception of the document (e.g., as *exciting*) as well as the perceived personality of the author (e.g., in terms of trustworthiness). The findings show strong effects across the assessed font types with respect to the perception of documents, whereas congruent and neutral fonts appear to evoke similar perceptions of an author's personality.

Shaikh et al. (2007) study the effect of the choice of font on email perception. Their results suggest that fonts with low congruency may result in different perceptions of an email than fonts with medium to high congruency. A similar study on the perception of a company website (Shaikh, 2007a) demonstrates that neutral and low congruency fonts can negatively affect a company's perception in terms of professionalism, believability, trust, and intent to act on the site.

Many studies in marketing analyze font effects, especially in packaging design. For instance, Fligner (2013) shows that fonts associated with the attribute *natural* increase the perceived *healthfulness* of products when used in their packaging, particularly if the products' intrinsic cues (e.g., being fat-free) and extrinsic ones (e.g., being sold at Whole

[3] http://www.dgp.toronto.edu/~donovan/font/

[4] https://github.com/Jack000/fontjoy

[5] https://datascopeanalytics.com/fontstellations/

[6] http://fontmap.ideo.com/

	anger	anticipation	disgust	fear	joy	negative	positive	sadness	surprise	trust
1	¬calm	fresh	clumsy	bad	happy	bad	strong	¬happy	dramatic	strong
2	clumsy	formal	bad	capitals	playful	strong	¬bad	gentle	happy	calm
3	capitals	dramatic	sloppy	¬calm	graceful	sharp	happy	¬graceful	¬sharp	¬bad

Table 1: Top three closest attributes, where ¬ indicates attributes that are negated

Foods Market) also concur. Childers and Jass (2002) establish that the semantic attributes of a font bear an impact on user perception for both high and low engagement levels. Through experiments using bottled water of a fictional brand, Van Rompay and Pruyn (2011) finds additional evidence that the congruence between fonts and other design elements influence the perception of brand credibility, aesthetics, and value.

3. Emotion Mapping

In this section, we describe our method to obtain font scores for the emotion attributes that shall later, in the following section, be used to obtain font scores for words in an existing emotion lexicon.

3.1. Method

Our method assumes as input a set of fonts $\mathcal{F}$ that are described in terms of a set of font attributes $\mathcal{A}$. For this, we rely on the crowdsourced data from O'Donovan et al. (2014), which for a given font $f \in \mathcal{F}$ provides scores in $[0, 100]$ for each attribute $a \in \mathcal{A}$. From this data, we derive $|\mathcal{F}|$-dimensional vectors $\vec{a} \in [0, 1]^{|\mathcal{F}|}$ for each font attribute $a \in \mathcal{A}$. For this, we simply transform the dataset to consider the fonts for a given font attribute, normalizing scores to $[0, 1]$.

Then, to induce FontLex, we first generate $|\mathcal{F}|$-dimensional font vectors for a set of emotion attributes $\mathcal{E}$. Subsequently, using existing word–emotion associations, we will infer $|\mathcal{F}|$-dimensional font vectors for words such that each component of such a vector quantifies the strength of the association between a word and a font.

As the set of emotions $\mathcal{E}$, we consider the ten emotion attributes used in EmoLex (Mohammad and Turney, 2013). Our first step is to map these $e \in \mathcal{E}$ to vectors $\vec{e} \in R^{|\mathcal{F}|}$ that characterize their association with fonts $f \in \mathcal{F}$ in our data. To achieve this, we proceed as follows. For each emotion $e \in \mathcal{E}$, we determine the $k = 3$ most similar font attributes $a \in \mathcal{A}$, as shown in Table 1. To decide on this value, we have carried out leave-one-out tests on the crowdsourced seed dataset (O'Donovan et al., 2014). Although the average overall success of the method in terms of the mean error was slightly higher for higher k than 3, we found that for $k = 3$ the most attributes attained their highest scores. Also considering the complexity of the negation decisions as will be described shortly, we opted to use the closest $k = 3$ neighbors.

We rely on word2vec distances $d(e, a)$, using cosine distances on the standard word2vec Google News pretrained model[7], to determine similarity scores $\mathrm{sim}(e, a)$ between emotion names and font attribute names as below:

$$\mathrm{sim}(e, a_i) = \frac{1}{k-1} \frac{\sum_{\substack{j=1 \\ i \neq j}}^{k} d(e, a_j)}{\sum_{j=1}^{k} d(e, a_j)} \tag{1}$$

One aspect that needs to be addressed, however, is the widely known fact that distributional models of semantics tend to conflate synonyms with antonyms. Hence, we first define

$$\vec{\mu}(e, a) = \begin{cases} \vec{1} - \vec{a} & \text{if } a \text{ is assessed as an antonym of } e \\ \vec{a} & \text{otherwise,} \end{cases} \tag{2}$$

where $\vec{1}$ is an $|\mathcal{F}|$-dimensional vector of ones. Thus, for those words that are assessed as antonyms, we do not use the regular font vector $\vec{a}$, but instead consider an inverted vector, in which we subtract each value from the maximum value of 1. The assessment is performed manually. For relationships such as between *anger* and *calm*, determining antonym relationships was straightforward. However, for some more challenging decisions, such as *negative* and *sharp*, we evaluated both options and discussed the obtained results with a graphic designer before making the final decision. In Table 1, attributes labelled as antonyms are marked with a "¬" symbol.

To obtain font vectors $\vec{e}$ for emotions $e \in \mathcal{E}$, we compute

$$\vec{e} = \sum_{i=1}^{k} \mathrm{sim}(e, a_i)\, \vec{\mu}(e, a_i) \tag{3}$$

where the a_i are the k most similar attributes, as described above. Thus, the font vectors are a weighted average of the vectors for related attributes, after possibly inverting their respective vectors.

3.2. Results

Figure 1 depicts the top 3 fonts that are most strongly associated with the ten emotion attributes, whereas Figure 3 shows the three fonts for each emotion that are found to have the weakest associations. Figure 2 shows sample fonts that are predicted to be neutral in terms of the respective emotion, which are ranked in the middle of the ranked font list. In all figures, the emotion names are rendered using the corresponding fonts.

The fonts that are strongly associated with emotions share some special characteristics. For instance, for *joy*, we encounter handwriting-style typefaces, whereas for *disgust*, we find display fonts with salient stylization. It should also be noted that not all fonts that share these characteristics are

[7] https://code.google.com/archive/p/word2vec/

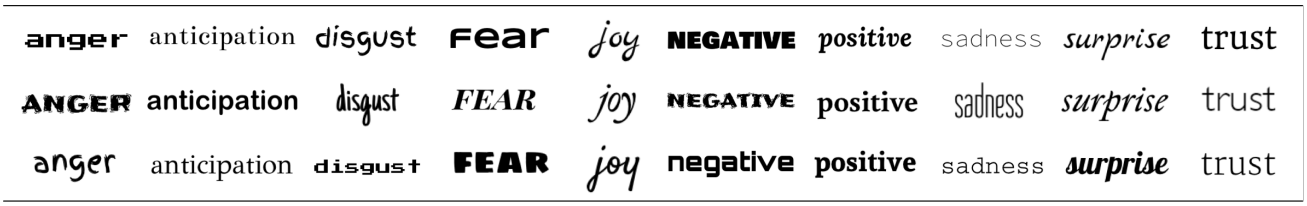


Figure 1: Emotion attributes rendered using the three most congruent fonts as predicted by our method. The renderings on the first line uses the fonts ranked 1st, the second line uses fonts ranked 2nd, and the third line uses fonts ranked 3rd.

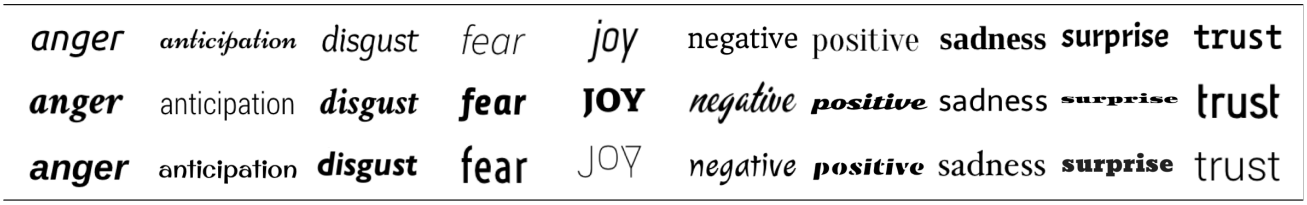


Figure 2: Emotion attributes rendered using the neutral fonts as predicted by our method. The renderings on the first line use the fonts ranked 99th, the second line uses fonts ranked 100th, and the third line uses fonts ranked 101st.



Figure 3: Emotion attributes using the three most incongruent fonts as predicted by our method. The renderings on the first line use the fonts ranked 198th, the second line uses fonts ranked 199th, and the third line uses fonts ranked 200th.

strongly associated with these emotions, since the relationships between emotion attributes and font characteristics are not straightforward (Kulahcioglu and de Melo, 2018).



Figure 4: An example task for *positive*. The second and fifth fonts are congruent, the third and fourth is incongruent and the first is neutral.

3.3. Evaluation

To assess the quality of the obtained emotion font score predictions, we carry out a user study.

3.3.1. User Study

For each of the ten emotion attributes, we generated four tasks with different random font choices. An example is given in Figure 4. Each task includes 5 fonts, two congruent fonts selected randomly among the top-scoring 10 fonts for that emotion, two incongruent fonts selected randomly among the lowest-scoring 10 fonts for that emotion, and one neutral font selected randomly among the ten fonts that are in the middle of the ranked list of fonts. In each task, the user is requested to select the image that best represents the word. As described above, the available options include the same word presented using five different fonts.

	Congruent	Neutral	Incongruent
Expected value	40.00	20.00	40.00
anger	74.04	14.42	11.54
anticipation	28.85	34.62	36.54
disgust	70.19	10.58	19.23
fear	78.85	5.77	15.38
joy	91.35	4.81	3.85
negative	59.80	15.69	24.51
positive	60.00	23.81	16.19
sadness	46.15	16.35	37.50
surprise	72.12	8.65	19.23
trust	62.50	20.19	17.31
Average	64.38	15.49	20.13

Table 2: Evaluation Results (in %) for Emotions

Each task is carried out by 30 participants in Mechanical Turk, all from the United States, with at least 5,000 approved hits and an overall approval rating of 97% or more. We used counterbalancing, i.e., half of the users received the tasks in the reverse order from the other half. We also used three validation tasks, and eliminated results of three participants who incorrectly answered all three of them.

3.3.2. Evaluation Results

Table 2 summarizes the results of this user study. The *congruent* column lists the percentages of selections in which



Figure 5: Selected words rendered using the three most congruent fonts as predicted by our method. The renderings on the first line uses the fonts ranked 1st, the second line uses fonts ranked 2nd and the third line uses fonts ranked 3rd.



Figure 6: Selected words rendered using the three most congruent fonts as predicted by our method. The renderings on the first line uses the fonts ranked 99th, the second line uses fonts ranked 100th and the third line uses fonts ranked 101st.

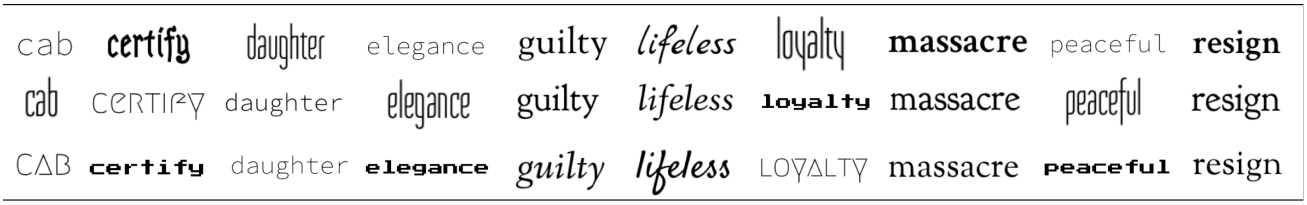


Figure 7: Selected words rendered using the three most incongruent fonts as predicted by our method. The renderings on the first line uses the fonts ranked 198th, the second line uses fonts ranked 199th and the third line uses fonts ranked 200th.

the congruent fonts (those in the top 10 for that word) are preferred. Similarly, the *neutral* and *incongruent* columns list the percentages of choices of neutral and incongruent fonts, respectively. The first row lists the expected value assuming the null hypothesis of a uniform distribution over the five choices, of which 2 are congruent, 1 neutral, and 2 incongruent.

The average is 64.38% for congruent font preferences. Compared to the expected value of 40%, this shows a strong trend toward the fonts predicted to be congruent, hence validating our results in general. Similarly, the preferences for the fonts that are found to be incongruent by our method was much lower than the expected value, with an average of only 20.13%.

However, a detailed look at the values for individual emotion attributes reveal that the performance differs between them. The strongest preference is obtained for *joy*, with a value of 91.35%, whereas the lowest is for *anticipation* with 28.85%. Another comparably low value is obtained for *sadness*, with a congruency of 46.15%. This suggests that different emotions may differ in how saliently and uniquely they are associated with visual font characteristics (cf. Section 5.).

4. Lexical Mapping

The next phase involves computing font vectors for words.

4.1. Method

EmoLex (Mohammad and Turney, 2013) provides binary emotion association indicators between words and the emotion attributes $e \in \mathcal{E}$ listed in Table 1. There are 6,468 words with at least one emotion association in their data. For words w in this set, we consider their data as providing vectors $\vec{w}_{\mathrm{E}} \in [0,1]^{|\mathcal{E}|}$.

To generate a font vector $\vec{w}_{\mathrm{F}}$ for a word w, we compute

$$\vec{w}_{\mathrm{F}} = \frac{1}{\|\vec{w}_{\mathrm{E}}\|_1} \mathbf{M}_{\mathrm{E}} \vec{w}_{\mathrm{E}} \tag{4}$$

where $\|\vec{w}_{\mathrm{E}}\|_1$ denotes the ℓ_1 norm of $\vec{w}_{\mathrm{E}}$ and $\mathbf{M}_{\mathrm{E}} = [\vec{e}_1 \ldots \vec{e}_{|E|}]$, i.e., a matrix with columns that capture the font vectors for the emotions $e \in \mathcal{E}$ (in the same order as captured in $\vec{w}_{\mathrm{E}}$).

4.2. Results

Figure 5 shows the top three congruent fonts associated with ten sample words, Figure 7 shows the most incongruent three fonts for the same words, and Figure 6 shows sample fonts that are predicted to be neutral for the respective words. In all images, the words are rendered using the corresponding fonts. These words are among those used in the evaluation user study in the following section.

4.3. Evaluation

We evaluate the dataset through a user study. In the following, we provide details on the design and the results of this study.

Expected value	Congruent 40.00	Neutral 20.00	Incongruent 40.00	Corresponding Emotion Attributes AG	AN	D	F	J	N	P	SA	SU	T
appreciation	70.59	15.69	13.73	0	0	0	0	1	0	1	0	0	1
cab	53.85	11.54	34.62	0	0	0	0	0	0	1	0	0	0
certify	79.59	6.12	14.29	0	0	0	0	0	0	0	0	0	1
conformance	61.54	19.23	19.23	0	0	0	0	0	0	1	0	0	0
congenial	42.86	20.41	36.73	0	0	0	0	0	0	1	0	0	0
daughter	70.59	13.73	15.69	0	0	0	0	1	0	1	0	0	0
elegance	76.00	16.00	8.00	0	1	0	0	1	0	1	0	0	1
guiilty	49.02	21.57	29.41	1	0	0	0	0	1	0	1	0	0
instruct	55.77	28.85	15.38	0	0	0	0	0	0	1	0	0	1
kill	75.00	5.77	19.23	0	0	0	1	0	1	0	1	0	0
lifeless	32.00	26.00	42.00	0	0	0	1	0	1	0	1	0	0
loyalty	76.92	9.62	13.46	0	0	0	0	0	0	1	0	0	1
massacre	56.00	16.00	28.00	1	0	1	1	0	1	0	1	0	0
medley	40.38	36.54	23.08	0	0	0	0	0	0	1	0	0	0
murky	82.35	5.88	11.76	0	0	1	0	0	1	0	1	0	0
noble	72.55	15.69	11.76	0	0	0	0	0	0	1	0	0	1
oracle	52.00	16.00	32.00	0	1	0	0	0	0	1	0	0	1
outcome	64.71	17.65	17.65	0	0	0	0	0	0	1	0	0	0
peaceful	64.00	12.00	24.00	0	1	0	0	1	0	1	0	1	1
persistent	65.38	9.62	25.00	0	0	0	0	0	0	1	0	0	0
precedence	56.86	15.69	27.45	0	0	0	0	0	0	1	0	0	1
resign	20.00	18.00	62.00	1	0	1	1	0	1	0	1	0	0
shameful	50.00	28.85	21.15	0	0	0	0	0	1	0	1	0	0
tickle	63.46	9.62	26.92	0	1	0	0	1	0	1	0	1	1
verified	64.71	23.53	11.76	0	0	0	0	0	0	1	0	0	1
Average	59.85	16.78	23.37										

Table 3: Evaluation Results (in %) and Emotion Associations for Words in the User Study. (AG: Anger, AN: Anticipation, D: Disgust, F: Fear, J: Joy, N: Negative, P: Positive, SA: Sadness, SU: Surprise, T: Trust)



Figure 8: An example task for the word *certify*. The second and fifth fonts are congruent, the first and third is incongruent, and the fourth is neutral.

4.3.1. User Study

For our study, we consider 25 words randomly selected from the set of words with at least one salient font association. For this purpose, we consider any of the 3,882 words that have a score of 0.75 or higher in any of the components of their respective font vectors. For each of the random 25 words, we generated two tasks with different random font choices. We have reduced the number of tasks to two, compared to the four tasks used in the previous section, to keep the total number of tasks reasonable for each participant.

An example task for the word *certify* is given in Figure 8. Each task includes 5 fonts, two congruent fonts selected randomly among the top-scoring 5 fonts for that word, two incongruent fonts selected randomly among the lowest-scoring 5 fonts for that word, and one neutral font selected randomly among the three fonts that are in the middle of the ranked list of fonts for the word. The decision to use 5 fonts as opposed to 10 is again based on considerations regarding the workload per user.

Each task involves a user being requested to select the image that best represents the word. As described above, the available options include the same word presented using five different fonts. Each task is carried out by 30 participants in Mechanical Turk, all from the United States, with at least 5,000 approved hits and an overall approval rating of 97% or more. We used counterbalancing and eliminated results of one participant that accidentally completed both of the original and reversed task sessions. We have also used three validation tasks, and eliminated results of one participant that incorrectly answered both of the two validation tasks.

4.3.2. Evaluation Results

Table 3 summarizes the evaluation results for the 25 randomly selected words as described above. The *congruent* column lists the percentages of selections in which the congruent fonts (those in the top 5 for that word) are preferred. Similarly, the *neutral* and *incongruent* columns list the percentages of choices of neutral and incongruent fonts, respectively.

The average is 59.85% for congruent font preferences, which shows that the consensus between our data and the users were strong. The strongest preference is obtained for

the word *murky*, with a value of 82.35%, whereas the lowest is for the word *resign* with 20.00%. Similarly, the average for the incongruent preferences was only 23.37%, bearing further witness to the quality of the results. Only two out of twenty-five words, namely *lifeless* and *resign*, received congruent preferences that are less than the expected value of 40%. Such results are expected, given that different words may differ in the strength and uniqueness of their associations (cf. Section 5.).

Table 3 also displays the corresponding emotions for the words used in the evaluation, allowing us to analyze the relationship between the success of the two datasets. In some cases, words associated with the same set of emotions obtained similar user ratings, such as *instruct*, *noble*, *precedence*, and *verified*. Whereas in some cases, words with the same emotion set obtained quite divergent ratings: *massacre* and *resign*.

5. Discussion on Results

We have introduced two datasets that connect emotions and words with fonts in terms of real-valued scores. Besides showing strong support for the datasets, the user evaluations also revealed that the performance varies for different emotions and words. Below, we discuss the potential sources for these differences.

For the emotion–font dataset, one reason for the differences between results could be the varying potential of fonts to represent or evoke different emotions (Kulahcioglu and de Melo, 2018). This could be observed in the results for *anticipation*, for which determining a font type may prove difficult even for an experienced graphic designer. It is also observed that emotions with higher arousal, namely *anger*, *disgust*, *fear*, *joy*, and *surprise*, received higher congruent user preferences compared to other emotions, which may be a direction that merits further analysis.

The second reason may be a lack of appropriate similar attributes in the crowdsourced seed dataset. Looking at Table 1, it could be argued that *joy* has semantically close neighbors in the dataset, whereas this is not the case for *anticipation*.

For the word–font dataset, checking the underlying emotion connections using Table 3 may shed some light on the differences. Recalling that the lowest performing emotion–font scores are for *anticipation* and *sadness*, one might expected that words associated with these emotions are prone to showing fewer user preferences that are congruent. The words associated with *anticipation*, namely *elegance*, *oracle*, *peaceful*, and *tickle*, do not seem to possess the same difficulty, as the lowest preference for these words is 52% (for *outcome*), which shows a strong preference.

On the other hand, among the words associated with *sadness*, the words *lifeless* and *resign* do not show such strong preferences. One might conjecture that this stems from low-performing emotion–font associations. However, a detailed look reveals that *kill* and *massacre* have the same underlying emotion associations as *lifeless* and *resign*, respectively. The fact that the fonts for *kill* and *massacre* received strong support from users indicates that the word–emotion associations might have played a role. Some words may have inaccurate or missing emotion associations, or some words may have weaker emotional associations than others, which is not reflected in the binary scheme used by EmoLex. Using a dataset with real-valued scores instead of binary associations might help to capture the latter case.

Fortunately, overall, both datasets have received strong support from users, with around 60% and 64% of the average user preferences towards the fonts found to be congruent by our datasets. Only for two words out of twenty-five, incongruent fonts are preferred more frequently than chance would predict, i.e. $\frac{2}{5} = 40\%$. In contrast, for 23 words, congruent fonts are preferred more frequently than chance would predict. Despite the subjective nature of font preferences and associations, we observe that there is a clear correspondence between the fonts chosen by our method and those assessed as appropriate by the human participants.

6. Extension via Semantic Relationships

Finally, we extend the dataset and increase its accuracy by accounting for semantic relationships given by WordNet (Fellbaum, 1998). For all attribute words in $\mathcal{E} \cup \mathcal{A}$, in total 47 attributes (37 original font attributes and 10 emotion attributes for which our study has computed font vectors), we gather the set of words that share a common synset with the attribute names (such as the words *deadening*, *dull*, *hohum*, *irksome*, *slow*, *tedious*, *tiresome* and *wearisome* for the font attribute *boring*). We then go through this list manually to exclude any synonyms with an irrelevant meaning (such as the word *building complex* for the font attribute *complex*). The remaining synonyms are assigned the font vectors of the words in $\mathcal{E} \cup \mathcal{A}$. This results in 364 additional word-font assignments, 112 of which override the ones from the methods in Sections 3. and 4. While small in number, these provide for particularly salient associations.

7. Conclusions and Future Work

To the best of our knowledge, no existing tool or resource provides semantic font recommendation support in which the meaning of the text is computationally matched with the semantic attributes of the fonts. Our study aims to support the development of such font recommendation tools.

Following this aim, we have created FontLex[8], a dataset that maps 6.7K words to 200 fonts. These derive mainly from the affective associations between words and fonts. Our evaluation shows an average of 55.95% of selections evincing a preference for the fonts recommended by the dataset. This is a strong result given the subjective nature of such preferences. Our ongoing work is broadening this even further based on further semantic relationships.

As part of the future work, we plan to further expand the dataset by making use of font attributes such as *thin*, *wide*, and *angular*, and their connections with objects, as opposed to the more abstract focus in this paper.

Acknowledgments

Gerard de Melo's research is supported in part by the DARPA SocialSim program. Tugba Kulahcioglu presents her thanks to Anthony Porturas for discussions on an earlier version of this study.

[8] http://gerard.demelo.org/fonts/

8. Bibliographical References

Bloch, M. (2017). 1001 Fonts. http://www.1001fonts.com. Accessed: 2017-07-29.

Canva.com. (2017). Canva font combinations. https://www.canva.com/font-combinations/. Accessed: 2017-07-29.

Childers, T. L. and Jass, J. (2002). All dressed up with something to say: Effects of typeface semantic associations on brand perceptions and consumer memory. *Journal of Consumer Psychology*, 12(2):93–106.

dafont.com. (2017). http://www.dafont.com. Accessed: 2017-07-29.

Data Scope Analytics. (2017). Fontstellations. https://datascopeanalytics.com/blog/fontstellations/. Accessed: 2017-07-29.

Fellbaum, C. (1998). Wordnet: an eletronic lexical database. cambridge, massachusetts, eua.

Fligner, A. (2013). *The effect of packaging typeface on product perception and evaluation*. Ph.D. thesis, The Ohio State University.

Fox, D. (2010). Free will for typeface selection: Myth or reality? *Usability News*, 12(1).

Hazlett, R. L., Larson, K., Shaikh, A. D., and Chaparo, B. S. (2013). Two studies on how a typeface congruent with content can enhance onscreen communication. *Information Design Journal*, 20(3).

Ho, K. (2017). Organizing the world of fonts with ai. https://medium.com/@kho98. Accessed: 2017-07-29.

Juni, S. and Gross, J. S. (2008). Emotional and persuasive perception of fonts. *Perceptual and motor skills*, 106(1):35–42.

Kulahcioglu, T. and de Melo, G. (2018). Predicting semantic signatures of fonts. In *Proceedings of the 12th IEEE International Conference on Semantic Computing*. IEEE.

Lewis, C. and Walker, P. (1989). Typographic influences on reading. *British Journal of Psychology*, 80(2):241–257.

Mills, H. (2017). Font pair. http://fontpair.co/. Accessed: 2017-07-29.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

O'Donovan, P., Libeks, J., Agarwala, A., and Hertzmann, A. (2014). Exploratory font selection using crowdsourced attributes. *ACM Trans. Graph.*, 33(4):92:1–92:9.

Qiao, J. (2017). Fontjoy – Font pairing made simple. http://fontjoy.com/. Accessed: 2017-07-29.

Sam Berlow, S. C. and Sherman, N. (2017). Fonts In Use. https://fontsinuse.com/. Accessed: 2017-07-29.

Shaikh, A. D., Chaparro, B. S., and Fox, D. (2006). Perception of fonts: Perceived personality traits and uses. *Usability News*, 8(1):1–6.

Shaikh, A. D., Fox, D., and Chaparro, B. S. (2007). The effect of typeface on the perception of email. *Usability News*, 9(1):1–7.

Shaikh, A. D. (2007a). The effect of website typeface appropriateness on the perception of a company's ethos. *Usability News*, 9(2).

Shaikh, A. D. (2007b). *Psychology of onscreen type: Investigations regarding typeface personality, appropriateness, and impact on document perception*. Ph.D. thesis.

Van Rompay, T. J. and Pruyn, A. T. (2011). When visual product features speak the same language: Effects of shape-typeface congruence on brand perception and price expectations. *Journal of Product Innovation Management*, 28(4):599–610.

Velasco, C., Salgado-Montejo, A., Marmolejo-Ramos, F., and Spence, C. (2014). Predictive packaging design: Tasting shapes, typefaces, names, and sounds. *Food Quality and Preference*, 34:88 – 95.

Velasco, C., Woods, A. T., Hyndman, S., and Spence, C. (2015). The taste of typeface. *i-Perception*, 6(4).

Wang, Z., Yang, J., Jin, H., Shechtman, E., Agarwala, A., Brandt, J., and Huang, T. S. (2015). Deepfont: Identify your font from an image. In *Proc. ACM Multimedia*, pages 451–459. ACM.

Multi-layer Annotation of the Ṛgveda

Oliver Hellwig[1] **, Heinrich Hettrich**[2] **, Ashutosh Modi**[3] **, Manfred Pinkal**[3]

[1]SFB 991, Düsseldorf University, ohellwig@phil-fak.uni-duesseldorf.de
[2]Würzburg University, heinrich.hettrich@uni-wuerzburg.de
[3]Department of Language Science and Technology, Saarland University, {ashutosh,pinkal}@coli.uni-saarland.de

Abstract

The paper introduces a multi-level annotation of the ṚGVEDA, a fundamental Sanskrit text composed in the 2. millenium BCE that is important for South-Asian and Indo-European linguistics, as well as Cultural Studies. We describe the individual annotation levels, including phonetics, morphology, lexicon, and syntax, and show how these different levels of annotation are merged to create a novel annotated corpus of Vedic Sanskrit. Vedic Sanskrit is a complex, but computationally under-resourced language. Therefore, creating this resource required considerable domain adaptation of existing computational tools, which is discussed in this paper. Because parts of the annotations are selective, we propose a bi-directional LSTM based sequential model to supplement missing verb-argument links.
Keywords: Sanskrit, multi-layer corpus, verb-argument structures, Ṛigveda

1. Introduction

In this paper, we introduce a multi-layer annotation of the complete ṚGVEDA (ṚV). The ṚV is central for research in Indo-European linguistics, because it is the oldest sample of this language family for which a sizeable text corpus has been transmitted (Witzel, 1995). The Vedic core corpus consists of four collections of hymns called Vedas ("knowledge"), which deal mainly with the worship of the Vedic pantheon and details of the ritual. The ṚV is the oldest among these four collections. It may have been composed around 1,500 BCE, has been transmitted orally for at least two millenia, and has, in spite of its age, remained a foundational text for the religious, cultural, and linguistic history of South Asia. Ideas and actors mentioned in the ṚV are constantly referred to in later texts produced on the Indian subcontinent (Gonda, 1975).

The corpus presented in this paper merges the phonetic, morphological, and lexical analyses for each word of the ṚV with a verb-argument (VA) annotation that links each verbal form to its main syntactic arguments. The first part of the paper describes how the annotation was performed, and provides a quantitative overview of size and structure of the resulting corpus. The second part of the paper presents a basic argument identification algorithm for Vedic Sanskrit. The VA annotation was created in a linguistic research context, but not as a building block of an NLP pipeline. As a consequence, parts of the case semantic information are not encoded explicitely, but need to be supplemented by a human reader. We will use the presented argument identification algorithm for complementing these missing parts of the VA annotation, and, later on, for annotating other central texts of the Vedic corpus.

This paper makes two important contributions. First, it introduces a novel resource for Sanskrit with deep linguistic annotations, ranging from the phonetic up to the syntactic level. The full annotated corpus is available via `https://git.adwmainz.net/open/rigveda` under the Creative Commons Attribution 4.0 International Public License. We expect that our ṚV annotation will become a standard reference resource for (diachronic) Indo-European linguistics, and for religious and cultural studies. Second, we describe how we merged two independent linguistic annotations to build a large digital corpus for a challenging South-Asian language, even though Vedic Sanskrit is strongly under-resourced from the viewpoint of NLP. Moreover, initial experiments with an argument identification algorithm that are reported in this paper, open up interesting perspectives for future research in automatic verb-argument detection.

The rest of the paper is organized as follows. Section 2. gives an overview of related research in Sanskrit CL and Vedistic studies. Sections 3. and 4. describe the morpho-lexical and verb-argument annotation of the data, and the necessary domain adaptation. Section 5. describes how these two annotation levels were merged into a single consistent format. Section 6. describes the algorithm developed for argument identification, and Sec. 7. summarizes the paper.

2. Related Research

Several authors studied the Vedic case system in general as well as the semantic functions of individual cases (Haudry, 1977; Hettrich, 2007; Kulikov, 2009). Detail studies such as Dahl (2014) also assessed how certain semantic roles are realized at the morpho-syntactic level in early Vedic. While these contributions focus strongly on language use in the ṚV, other important aspects of Vedic syntax such as word order or morpho-syntactic alignment were only studied for later Vedic prose (Delbrück, 1888) or with limited material from the ṚV. We are confident that our multi-layer annotation of the ṚV will provide the basis for large-scale studies of such phenomena in the oldest layer of Vedic.

To our knowledge, there exist no computational processing tools nor publicly available annotated corpora for the Vedic language. Hellwig (2009) introduced a stochastic morpho-lexical tagger for classical Sanskrit, which was extended to early Vedic for this paper. The systems described by Huet (2006), Kulkarni and Shukla (2009), and Jha (2009) aim at classical Sanskrit and are strongly influenced by the Pāṇinian framework of grammar, so that an extension to Vedic is not easily feasible.

Input text in Devanagari	वासयोषसः		श्रवसे
Input text in Latin	vāsayoṣasaḥ		śravase
Sandhi split	vāsaya $_{(a+u=o)}$ uṣasaḥ		śravase
Lexemes	vāsay	uṣas	śravas
Morphology	2. sg., imp.	acc. pl.	dat. sg.
Word meanings	to make shine	dawn	fame
Translation	**"For fame make the dawns shine."**		

Figure 1: Levels of linguistic analysis for ṚV, 1.134.3. Abbreviations: sg.: singular, imp.: imperative; pl.: plural, acc.: accusative, dat.: dative. Translation taken from Jamison and Brereton (2014, 304)

3. Morpho-lexical Annotation

We used a tagger that was originally developed for Classical Sanskrit (Hellwig, 2015a) for creating a morpho-lexical analysis of the ṚV in the edition of van Nooten and Holland (1994). This tagger produces all possible tokenizations of the input text that consist of morphologically and lexically valid word forms. Tokenization of Sanskrit is a challenging task, because individual words are merged by a set of phonetic rules called Sandhi ("connection"), whose resolution is non-deterministic and, therefore, guided by the morphological, lexical, and semantic composition of a sentence.[1] This tokenization step results in a trellis of possible readings for each line of text. A dynamic programming approach that operates with a trigram language model (Brants, 2000) is used to find the most probable lexical path through this trellis. Final fine-grained morphological decisions are made by applying a Conditional Random Field (Lafferty et al., 2001) model to the most probable lexical path. The solutions are ordered by decreasing linguistic probability, given the data from the language model. The first author of this paper finally validated all proposed system analyses in a manual correction step, resulting in a morphological and lexical gold annotation of the complete ṚV. Figure 1 shows a schematic overview of annotation levels for a part of hymn ṚV, 1.134.

3.1. System Adaptation

Although Classical Sanskrit developed out of a late form of Vedic Sanskrit, which was described by the grammarian Pāṇini, they represent two separate layers of Old Indo-Aryan. Therefore, we needed to perform domain adaptation of the tagger in three linguistic areas:

1. The inflectional system of Vedic Sanskrit has preserved Indo-European traits that are extinct in Classical Sanskrit, and formation of verbal forms is partly opaque in older Vedic.[2] We extended the morphological rule base and the full form dictionary of the tagger on per case basis, using Macdonell (1916). Figure 2 sets the number of newly added verbal forms (y-axis) in relation to the progress of annotation (x-axis). The plot shows that the number of cases in which we had to extend the full form database manually decreases over time, indicating improving adaptation to the new linguistic domain.

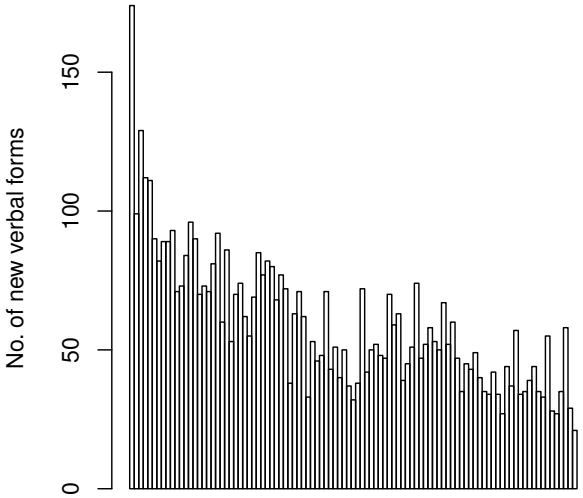


Figure 2: Distribution of the 5,908 newly added verbal forms over the annotation process (x-axis), illustrating the increasing domain adaptation of the tagger.

2. Due to the chronological distance of approximately 1,000 years and fundamental differences in genres and topics, Vedic and Classical Sanskrit use rather different vocabularies (Hellwig, 2017). Vedic texts in general and especially the ṚV contain many words that have disappeared in Classical Sanskrit. In addition, lexical semantics differ strongly between Vedic and Classical Sanskrit. The noun *vadha*, for example, can denote a tool for killing in the ṚV (e.g., ṚV 10.102.3), while it only denotes the act of killing in Classical Sanskrit. Bayesian models of semantic change (Frermann and Lapata, 2016) or diachronically motivated word embeddings (Hamilton et al., 2016) are not easily applicable, because the Vedic subcorpus is small,[3] and the text historical research in older Sanskrit literature is full of uncertainties (Fosse, 1997). The lexical database of the tagger was adapted to the Vedic vocabulary using the specialized dictionary of Grassmann (1873), and Geldner's German translation of the text (Geld-

[1] The words *rājā* 'the king' and *uvāca* 'he said', for example, are merged into the string *rājovāca* by the Sandhi rule *ā+u=o*. Sandhi complicates the linguistic processing of Sanskrit, because different Sandhi rules can result in the same merged phoneme. In the given example, the merged phoneme *o* could also have been produced by the rules *a+u*, *a+ū*, or *ā+ū*. Refer to Kielhorn (1888, 6ff.) and Hellwig (2015b) for further details.

[2] Consider the form *akrān* "he shouted", which is derived from the root *krand* by the following sequence of phonetic operations: lengthening of the root vowel (*krānd*), hardening of the final consonant (*krānt*), adding inflectional suffixes expressing number and person to the root (*krāntt*), dropping multi-consonant clusters at the end of the root (*krān*), and prefixing an augment (*akrān*).

[3] It may contain less than 5 million tokens, 5-10% of which have been analyzed so far.

ner, 1951 1957), from which we have integrated 923 and 1,599 new lexical meanings, respectively.

3. Basic syntactic structures have changed considerably from Vedic to Classical Sanskrit. While Vedic Sanskrit uses syntactic constructions such as relative and subordinate adverbial clauses (Hettrich, 1988; Hock, 2013), Classical Sanskrit tends to express subordination through compounding (Lowe, 2015) and absolutives (Tikkanen, 1991). In order to integrate syntactic changes in the tagging process, its trigram language model was split into two submodels, one trained on Classical Sanskrit, and one on the combination of the Classical and Vedic subcorpora.

3.2. Evaluation

This paper focuses on resource building, and the corpus of old Vedic is, more or less, restricted to the ṚV. Therefore, we did not evaluate the performance of the adapted tagger systematically. In order to get an idea of its performance, the main author of this paper recorded errors made by the system when analyzing the two hymns 8.102 and 8.103, both of which are dedicated to Agni, the god of fire and sacrifice. The evaluation distinguishes between three types of errors (Hellwig, 2015a):

1. Tokenization error: The system fails to propose the correct tokenizing split of a string. Example: *upastutāsaḥ*; correct: no split ('those [nom. pl. m.] who are praised'); system proposal: *upastutā-asaḥ* ('you [nom. sg. f.] will be the praised one'). Tokenization errors invalidate the analysis of a whole string.

2. Lemmatization error: The system fails to choose the right lexeme in a correctly tokenized string. Example: *rātahavyaḥ*: correct 'who has bestowed the oblation' (*rāta* = past participle of the verb *rā* 'to give'); system proposal: 'the oblation of Rāta' (*rāta* = name of a man).

3. Morphological error: Both preceding steps are solved correctly, but the system proposes a wrong morphological analysis of a token. Example: *haviṣkṛtaḥ*: correct: 'of him who prepares the oblation' (gen. sg.); system proposal: nom. pl. ('those who prepare the oblation'). Note that this form has 13 valid morphological readings.

Table 1 reports the error levels for the two Agni hymns. The highest error rates are observed on the morphological level. Given the morphological complexity of Vedic Sanskrit, this outcome is not really surprising. On the other hand, the system has a remarkably high tokenization accuracy. This somehow unexpected result is to a large degree due to the scholarly preprocessing of the ṚV, whereby Sandhis are resolved as far as possible in order to facilitate word search. When run on unpreprocessed (*saṃhitā*) texts, the system will certainly make a higher number of tokenization errors.

Type	Number	Proportion
8.102 (208 strings, 228 tokens)		
Tokenization	7	3.4%
Lemmatization	5	2.1%
Morphology	15	6.6%
8.103 (189 strings, 209 tokens)		
Tokenization	4	2.1%
Lemmatization	10	4.8%
Morphology	22	10.5%

Table 1: Error evaluation for the two hymns 8.102 and 8.103. Proportions are calculated w.r.t. the number of strings for tokenization, and the number of tokens for lemmatization and morphology.

4. Verb-Argument Annotation

The second component of the corpus provides verb-argument structure. It is based on the manual annotation of 27,104 verbal heads in the complete ṚV by one of the authors of this paper (Hettrich, 2001; Hettrich, 2007), an expert on Indo-European languages, and Vedic Sanskrit in particular. Verbs and dependents are connected with labelled edges. In many cases, part of speech and semantic type of the head noun are annotated in addition.

The original motivation for the annotation was its use in personal linguistic research. As a consequence, (1) the used inventory of relation types was independently developed and does not follow a standard dependency-grammar or role-semantic annotation schema, and (2) the labelling is selective, i.e., relations that are evident and can be easily supplemented by the reader are not labelled.

Concerning problem (1), the verb-argument annotations basically refer to the level of grammatical relations, with a number of semantically motivated refinements (for example, *comitative*, *separative*, *partitive*, *oblique agent*). The statistics across all 54,038 manually labelled edges shows a distribution with about 20 frequently occurring labels, from instrumental (2286) and adverbial (2153) to comparison (71) and predicative adverbial (20). The long tail of infrequent tags comprises standard tags labeled as uncertain or unusual, or combinations of them. We left the label inventory unchanged, leaving the mapping to coarser or more standard relation inventories (e.g., Universal Dependency Grammar) to the user.

Problem (2), the incompleteness of annotation, is a greater challenge. In particular, edges are missing for 9,585 occurrences of subjects and 8,573 occurrences of direct objects (non-oblique cases). In these cases, the whole construction is manually tagged to express the information that it comes with a subject and/ or object, but the location of the dependent is not disambiguated in the Sanskrit text. Section 5. describes how we deal with these cases.

5. Merging the Annotation Layers

Because morpho-lexical and verb argument annotations were performed independently of each other, and their results were stored in different data formats, we needed to merge them into a uniform multi-layer format. Merging was performed in two steps. In the first step, we established a mapping between the individual verbal roots that serve as heads of the verbal constructions (strings *vāsaya* in Fig. 3). As Indological researchers use different systems for en-

coding the lemmata of verbal roots, and for disambiguating homonymous verbal roots, this step involved a large amount of manual intervention, including the definition of 67 mapping rules between verbal roots.[4] 867 verbal heads can neither be mapped directly nor by applying the 67 special mapping rules. Moreover, we encountered 927 cases of copula omission, in which the verb-argument annotation does not disambiguate the arguments of the unexpressed copula.[5] These cases require a manual annotation step.
In the second step, we connect the arguments with their verbal heads. Apart from the problems with lemma encoding (see above), most non-oblique arguments are not disambiguated (see Sec. 4.). We deal with undisambiguated arguments in two ways:

1. 96.4% of all 24,861 disambiguated arguments occur in the same line of text as their verbal heads. Therefore, if verb v has an undisambiguated argument in case c, and the morpho-lexical annotation records exactly one word w with case c in the same text line as v, w is automatically disambiguated as the argument of v. This step produces 2,329 heuristic argument annotations.

2. If the preconditions for applying this heuristic are not met, because one line of text contains more than one word with case c, we use the labeling algorithm described in Section 6. for pre-annotating the arguments, and ask a human annotator to correct the output of the labeler.

The merged annotation contains 21,218 verbal heads, 20,438 of which are linked with a verbal form, while the remaining ones constitute sentences with missing copulae. Each verb has an average of 2.2 arguments (verbs without arguments: 6,399; with one arg.: 7,350; with 2-4 args.: 7,222; with more than 4 args.: 247).[6]

6. An Algorithm for Argument Identification

As mentioned in Sec. 4. and 5., the verb-argument annotation is selective. Therefore, we designed a basic argument identification algorithm that supports the re-annotation of non-oblique cases. Semantic role labeling is an active field of research in CL, and distinguishes between argument identification and argument classification (Gildea and Jurafsky, 2002). A wide range of learning algorithms such as probabilistic frameworks (Gildea and Jurafsky, 2002),

[4]The verbal roots are referenced by strings in the VA annotation and by unique numeric IDs on the morpho-lexical level. The 67 mapping rules need to disambiguate homonymous verbal roots such as *vas*, which can mean "to dwell" (*vasati*), "to wear" (*vaste*), or "to shine" (*ucchati*).

[5]The VA annotation indicates that a line of text contains a copula construction, but does not disambiguate the involved nominatives. – Use of copulae is optional in Sanskrit, with a strong tendency of not using it. So, the statement "Rāma is rich" can be expressed as *rāmo dhanyo 'sti* (*[a]sti* is the copula), or, more frequently as *rāmo dhanyaḥ*. Combined with the lack of punctuation, copula omission makes many Sanskrit texts highly ambiguous.

[6]Differences to numbers given for the VA annotation in Sec. 4. are due to mapping problems during the merging process.

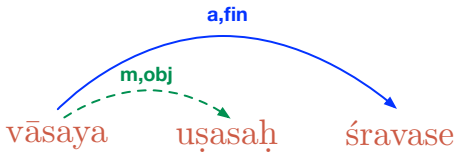


Figure 3: Full annotation of ṚV, 1.134.3 (refer to Fig. 1 for the morpho-lexical annotation). Labels on the arcs indicate the syntactic functions (obj[ect], fin[al]) and coarse word semantic classes (m = human, a = generic expression) of the arguments. Dashed arcs indicate arguments that are not disambiguated in the VA annotation.

Large Margin classifiers (Pradhan et al., 2008), and (recurrent/recursive) neural networks (Collobert et al., 2011; Roth and Lapata, 2016) was applied for semantic role labeling, and most authors emphasize the importance of high-quality parse trees as input features. While parse tree are not available for Vedic nor for Classical Sanskrit, the morpho-lexical annotations (Sec. 3.) provide a rich set of linguistic features that can be assumed to serve as proxies for syntactic relations in a weakly configurational language such as Sanskrit. Consequently, role identification is more challenging than role classification in Vedic Sanskrit, because role classification can make use of morphological features, as soon as a relation between a verb and an argument has been established. Under these circumstances, semantic role labeling can be viewed as a binary sequence annotation task: Given a sequence of noun forms and a verb, the classifier should select those noun forms that are arguments of the verb. Following recent research in CL and ML, we choose a recurrent neural network for this task.

Argument identification is performed in two steps. In the first step, an input line is processed with the morpho-syntactic tagger (Sec 3.), to get verbal forms V and their possible arguments A (all nouns and adjectives). In the second step, a neural network based sequential model is used to predict, if an argument at a given position t is related to a verb v or not. Note that the model works with text lines instead of sentences, because Sanskrit has not punctuation that indicates sentence ends. A single line of text can therefore contain more than one main verb. The neural network model has two bi-directional recurrent layers (Schuster and Paliwal, 1997). Each recurrent layer consists of 100 LSTM cells (Hochreiter and Schmidhuber, 1997). Prediction is done with a binary softmax activation function at the output. The model is trained by optimizing cross-entropy cost function and RMSProp schedule (Tieleman and Hinton, 2012), using an initial learning rate of 0.001, and a batch size of 8. Figure 4 shows the architecture for the model.

Each word of a text line is fed sequentially to the model. For each input word, lookup matrices are used to obtain the embeddings for the corresponding lemma and morphological information (e.g. case, gender, etc.). The embeddings for the lemma and morphological information are concatenated, and go as input to the first LSTM layer. The lemma embeddings are obtained by pre-training a word2vec skip-gram model (Mikolov et al., 2013) on the full corpus of

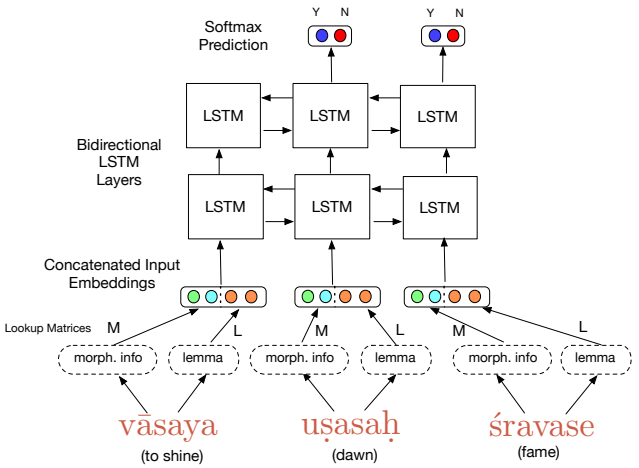


Figure 4: Bidirectional LSTM based neural network model for argument identification, unfolded for the task of labeling the sentence *vāsaya uṣasaḥ śravase* ("For fame make the dawns shine"; see Fig. 1). The model is trained to predict the words *uṣas* and *śravas* as argument of the verb *vāsay*.

Config.	P	R	F	N
all cases	72.74	69.35	71.00	20743
oblique cases	76.55	79.11	77.81	14318
non-obl. cases	60.74	46.55	52.71	6425
nom.	56.36	40.67	47.25	2606
acc.	61.84	47.69	53.85	2714
ins.	78.12	82.08	80.05	5078
dat.	75.51	81.66	78.46	3744
abl.	72.46	78.76	75.48	881
gen.	58.21	38.41	46.28	930
loc.	79.20	82.47	80.80	3685

Table 2: P(recision), R(ecall), and F score of the recurrent argument identifier. Column N gives the total number of training samples available for each configuration.

Classical and Vedic Sanskrit. The embeddings corresponding to morphological information are randomly initialized and learnt during training.

Since the dataset is small, the model is evaluated using 10-fold cross-validation with disambiguated and heuristically annotated arguments (Sec. 1). The model obtains an overall F score of 71.00 for roles in all cases, and of 77.81 when only oblique cases are considered (see Tab. 2). While this result compares favorably with results reported for verb argument detection (identification) tasks in English (Carreras et al., 2008; Das et al., 2013), one should keep in mind that the use of morpho-lexical gold information for Sanskrit, missing punctuation, and the small size of the Sanskrit corpus, when compared with corpora of modern languages, make a direct comparison impossible.

7. Summary

We have described the construction of a large-scale, multi-level annotation of the ṚGVEDA. In spite of the linguistic challenges raised by this complex Sanskrit text, we managed to merge two independent data sources into a consistent corpus, which we expect to become a standard reference tool in linguistic and cultural research. Post-processing and disambiguation of non-oblique arguments is work in progress. We designed an algorithm for argument identification that supports us in this ongoing task. In the future, we plan to extend this algorithm into a full-fledged verb-argument labeler for Sanskrit. Such a labeler will first be applied to Vedic texts that resemble the ṚV on the linguistic level (e.g., the metrical Atharvaveda), and later to the large corpus of Vedic prose texts.

Acknowledgements

Research for this project was partially funded by the Cluster of Excellence "Multimodal Computing and Interaction" of German Science Foundation (DFG). We thank the Akademie der Wissenschaften und der Literatur Mainz for hosting the annotated corpus.

References

Brants, T. (2000). TnT - a statistical part-of-speech tagger. In *Proceedings of the 6th Applied NLP Conference*, Seattle.

Carreras, X., Litkowski, K. C., and Stevenson, S. (2008). Semantic Role Labeling: An introduction to the special issue. *Computational Linguistics*, 34(2).

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Dahl, E. (2014). The morphosyntax of the experiencer in early Vedic. In Silvia Luraghi et al., editors, *Perspectives on Semantic Roles*, pages 181–204. John Benjamins Publishing Company.

Das, D., Chen, D., Martins, A. F. T., Schneider, N., and Smith, N. A. (2013). Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56.

Delbrück, B. (1888). *Altindische Syntax*. Verlag der Buchhandlung des Waisenhauses, Halle.

Fosse, L. M. (1997). *The Crux of Chronology in Sanskrit Literature*. Scandinavian University Press, Oslo.

Frermann, L. and Lapata, M. (2016). A Bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, 4:31–45.

Geldner, K. F. (1951–1957). *Der Rig-Veda. Aus dem Sanskrit ins Dt. übersetzt und mit einem laufenden Kommentar versehen von Karl Friedrich Geldner*. Harvard University Press, Cambridge, Mass.

Gildea, D. and Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288.

Gonda, J. (1975). *Vedic literature (Saṃhitās and Brāhmaṇas)*. A History of Indian Literature, Vol. I, Fasc. 1. Otto Harrassowitz, Wiesbaden.

Grassmann, H. (1873). *Wörterbuch zum Rig-veda*. Otto Harrassowitz, Wiesbaden.

Hamilton, W. L., Leskovec, J., and Jurafsky, D. (2016). Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the ACL*, pages 1489–1501.

Haudry, J. (1977). *L'emploi des cas en Vedique*. Les Hommes et les Lettres. Service de reproduction des thèses, Université de Lille III.

Hellwig, O. (2009). SanskritTagger, a stochastic lexical and POS tagger for Sanskrit. In Gérard Huet, et al., editors, *Sanskrit Computational Linguistics. First and Second International Symposia*, Lecture Notes in Artificial Intelligence, 5402, pages 266–277, Berlin. Springer Verlag.

Hellwig, O. (2015a). Morphological disambiguation of Classical Sanskrit. In Cerstin Mahlow et al., editors, *Systems and Frameworks for Computational Morphology*, pages 41–59, Cham. Springer.

Hellwig, O. (2015b). Using Recurrent Neural Networks for joint compound splitting and Sandhi resolution in Sanskrit. In Zygmunt Vetulani et al., editors, *Proceedings of the 7th LTC*, pages 289–293.

Hellwig, O. (2017). Coarse semantic classification of rare nouns using cross-lingual data and recurrent neural networks. In *Proceedings of the IWCS*.

Hettrich, H. (1988). *Untersuchungen zur Hypotaxe im Vedischen*. de Gruyter, Berlin, New York.

Hettrich, H. (2001). Die Erarbeitung einer Kasussyntax des Ṛgveda auf der Grundlage elektronisch gespeicherter Daten. In *Maschinelle Verarbeitung altdeutscher Texte*, pages 73–81. Max Niemeyer Verlag, Tübingen.

Hettrich, H. (2007). *Materialien zu einer Kasussyntax des Ṛgveda*. Universität Würzburg, Würzburg.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hock, H. H. (2013). Some issues in Sanskrit syntax. In Peter M. Scharf et al., editors, *Proceedings of the Seminar on Sanskrit syntax and discourse structures*, Paris.

Huet, G. (2006). Lexicon-directed segmentation and tagging of Sanskrit. In B. Tikkanen et al., editors, *Themes and Tasks in Old and Middle Indo-Aryan Linguistics*. Motilal Banarsidass, Delhi.

Jamison, S. W. and Brereton, J. P. t. (2014). *The Rigveda: the Earliest Religious Poetry of India*. Oxford University Press, New York.

Jha, G. N. (2009). Inflectional morphology analyzer for Sanskrit. In Gérard P. Huet, et al., editors, *Sanskrit Computational Linguistics*, pages 219–238. Springer, Berlin, Heidelberg.

Kielhorn, F. (1888). *Grammatik der Sanskrit-Sprache*. Dümmler Verlag, Berlin.

Kulikov, L. (2009). Evolution of case systems. In Andrej Malchukov et al., editors, *The Oxford Handbook of Case*, pages 439–457. Oxford University Press.

Kulkarni, A. and Shukla, D. (2009). Sanskrit morphological analyser: Some issues. *Indian Linguistics*, 70(1-4):169–177.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Lowe, J. J. (2015). The syntax of Sanskrit compounds. *Language*, 91(3):71–115.

Macdonell, A. A. (1916). *A Vedic Grammar for Students*. Clarendon Press, Oxford.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of ICLR Workshop*.

Pradhan, S. S., Ward, W., and Martin, J. H. (2008). Towards robust semantic role labeling. *Computational Linguistics*, 34(2):289–310.

Roth, M. and Lapata, M. (2016). Neural semantic role labeling with dependency path embeddings. In *Proceedings of the 54th Annual Meeting of the ACL*, pages 1192–1202.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*.

Tikkanen, B. (1991). On the syntax of Sanskrit gerund constructions: A functional approach. In Hans Heinrich Hock, editor, *Studies in Sanskrit Syntax. A Volume in Honor of the Centennial of Speijer's Sanskrit Syntax (1886-1986)*, pages 197–207. Motilal Banarsidass Publishers, Delhi.

van Nooten, B. and Holland, G. (1994). *Rig Veda: A Metrically Restored Text with an Introduction and Notes*. Harvard oriental series. Harvard University.

Witzel, M. (1995). Early Indian history: Linguistic and textual parametres. In George Erdosy, editor, *The Indo-Aryans of Ancient South Asia. Language, Material Culture and Ethnicity*, volume 1, pages 85–125. Walter de Gruyter, Berlin, New York.

The Natural Stories Corpus

Richard Futrell[1], Edward Gibson[1], Harry J. Tily[2], Idan Blank[1], Anastasia Vishnevetsky[1], Steven T. Piantadosi[3], and Evelina Fedorenko[4,5]
[1]MIT Department of Brain and Cognitive Sciences [2]Netflix, Inc.
[3]University of Rochester Department of Brain and Cognitive Sciences
[4]Massachusetts General Hospital Department of Psychiatry
[5]Harvard Medical School Department of Psychiatry
{futrell, egibson, iblank, evelina9}@mit.edu,
hal.tily@gmail.com, staseyvi@mail.med.upenn.edu

Abstract

It is now a common practice to compare models of human language processing by comparing how well they predict behavioral and neural measures of processing difficulty, such as reading times, on corpora of rich naturalistic linguistic materials. However, many of these corpora, which are based on naturally-occurring text, do not contain many of the low-frequency syntactic constructions that are often required to distinguish between processing theories. Here we describe a new corpus consisting of English texts edited to contain many low-frequency syntactic constructions while still sounding fluent to native speakers. The corpus is annotated with hand-corrected Penn Treebank-style parse trees and includes self-paced reading time data and aligned audio recordings. Here we give an overview of the content of the corpus and release the data.

Keywords: Cognitive modeling, reading time, psycholinguistics

1. Introduction

It is becoming a standard practice to evaluate theories of human language processing by comparing their ability to predict behavioral and neural reactions to fixed standardized corpora of naturalistic text. This method has been used to study several dependent variables which are believed to be indicative of human language processing difficulty, including word fixation time in eyetracking (Kennedy et al., 2013), word reaction time in self-paced reading (Roark et al., 2009; Frank et al., 2013), BOLD signal in fMRI data (Bachrach et al., 2009), and event-related potentials (Dambacher et al., 2006; Frank et al., 2015).

The more traditional approach to evaluating psycholinguistic models has been to collect psychometric measures on hand-crafted experimental stimuli designed to tease apart detailed model predictions. While this approach makes it easy to compare models on their accuracy for specific constructions and phenomena, it is hard to get a sense of how models compare on their coverage of a broad range of phenomena. Comparing model predictions over standardized texts makes it is easier to evaluate coverage.

Although the corpus approach has these advantages, the existing corpora currently used are based on naturally-occurring text, which is unlikely to include the kinds of sentences which can crucially distinguish between theories. Many of the most puzzling phenomena in psycholinguistics, and the phenomena which have been used to test models, have only been observed in extremely rare constructions, such as multiply nested relative clauses. Corpora of naturally-occurring text are unlikely to contain these constructions. More generally, models of human language comprehension are more likely to make distinct predictions for sentences that cause difficulty for humans, rather than for sentences that are easy to process. For instance, models of comprehension difficulty based on memory integration cost during parsing (Gibson, 2000; Lewis and Vasishth, 2005) will predict effects when the memory spans required for parsing are large, but most syntactic dependencies in naturally-occurring text are short (Temperley, 2007; Futrell et al., 2015). In general, situations that cause high processing difficulty might be rare in naturally-occurring text, because text written and edited in order to be easily understood.

Here we attempt to combine the strength of experimental approaches, which can test theories using targeted low-frequency structures, and corpus studies, which provide broad-coverage comparability between models. We introduce and release a new corpus, the Natural Stories Corpus, a series of English narrative texts designed to contain many low-frequency and psycholinguistically interesting syntactic constructions while still sounding fluent and coherent. The texts are annotated with hand-corrected Penn Treebank style phrase structure parses, and Universal Dependencies parses automatically generated from the phrase structure parses. We also release self-paced reading time data for all texts, and word-aligned audio recordings of the texts. We hope the corpus can form the basis for further annotation and become a standard test set for psycholinguistic models.[1]

2. Related Work

Here we survey datasets which are commonly used to test psycholinguistic theories and how they relate to the current release.

The most prominent psycholinguistic corpus for English is the **Dundee Corpus** (Kennedy, 2003), which contains

[1]The corpus is available from http://github.com/languageMIT/naturalstories. This corpus is distributed under an Attribution-NonCommercial-ShareAlike (CC BY-NC-SA) license, allowing free modification and re-distribution of the corpus so long as derivative work is released under the same terms.

51,501 word tokens in 2,368 sentences from British newspaper editorials, along with eyetracking data from 10 experimental participants. A dependency parse of the corpus is released in Barrett et al. (2015). Like in the current work, the eyetracking data in the Dundee corpus is collected for sentences in context and so reflects influences beyond the sentence level. The corpus has seen wide use (Demberg and Keller, 2008; Mitchell et al., 2010; Frank and Bod, 2011; Fossum and Levy, 2012; Smith and Levy, 2013; van Schijndel and Schuler, 2015; Luong et al., 2015).

The **Potsdam Sentence Corpus** (Kliegl et al., 2006) of German provides 1138 words in 144 sentences, with cloze probabilities and eyetracking data for each word. Like the current corpus, the Potsdam Sentence Corpus was designed to contain varied syntactic structures, rather than being gathered from naturalistic text. The corpus consists of isolated sentences which do not form a narrative, and during eyetracking data collection the sentences were presented in a random order. The corpus has been used to evaluate models of sentence processing based on dependency parsing (Boston et al., 2008; Boston et al., 2011) and to study effects of predictability on event-related potentials (Dambacher et al., 2006).

The **MIT Corpus** introduced in Bachrach et al. (2009) has similar aims to the current work, collecting reading time and fMRI data over sentences designed to contain varied structures. This dataset consists of four narratives with a total of 2647 tokens; it has been used to evaluate models of incremental prediction in Roark et al. (2009), Wu et al. (2010), and Luong et al. (2015).

The **UCL Corpus** (Frank et al., 2013) consists of 361 English sentences drawn from amateur novels, chosen for their ability to be understood out of context, with self-paced reading and eyetracking data. The goal of the corpus is to provide a sample of typical narrative sentences, complementary to our goal of providing a corpus with low-frequency constructions. Unlike the current corpus, the UCL Corpus consists of isolated sentences, so the psychometric data do not reflect effects beyond the sentence level.

Eyetracking corpora for other languages are also available, including the **Postdam-Allahabad Hindi Eyetracking Corpus** (Husain et al., 2015) and the **Beijing Sentence Corpus of Mandarin Chinese** (Yan et al., 2010).

3. Corpus Description

3.1. Text

The Natural Stories corpus consists of 10 stories of about 1000 words each, comprising a total of 10,245 lexical word tokens in 485 sentences. The stories were developed by A.V., E.F., E.G. and S.P. by taking existing publicly available texts and editing them to use many subject- and object-extracted relative clauses, clefts, topicalized structures, extraposed relative clauses, sentential subjects, sentential complements, local structural ambiguity (especially NP/Z ambiguity), idioms, and conjoined clauses with a variety of coherence relations. The texts and their sources are listed in Table 1.

The mean number of lexical words per sentence is 21.1, around the same as the Dundee corpus (21.7). Figure 1 shows a histogram of sentence length in Natural Stories as compared to Dundee. The word and sentence counts for each story are given in Table 2. Each token has a unique code which is referenced throughout the various annotations of the corpus.

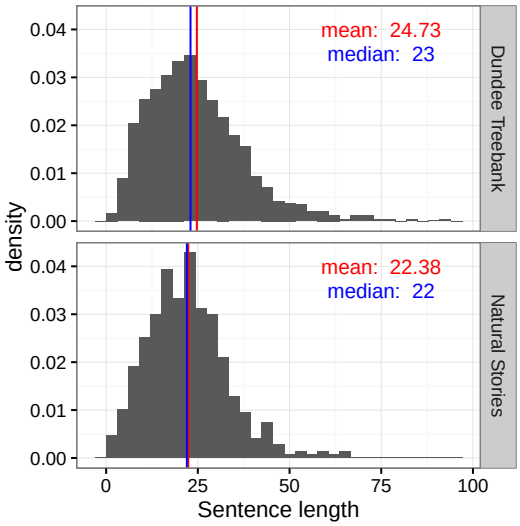


Figure 1: Histograms of sentence length (in tokens, including punctuation) in Natural Stories and the Dundee corpus.

> If you were to journey to the North of England, you would come to a valley that is surrounded by moors as high as mountains. It is in this valley where you would find the city of Bradford, where once a thousand spinning jennies that hummed and clattered spun wool into money for the long-bearded mill owners. That all mill owners were generally busy as beavers and quite pleased with themselves for being so successful and well off was known to the residents of Bradford, and if you were to go into the city to visit the stately City Hall, you would see there the Crest of the City of Bradford, which those same mill owners created to celebrate their achievements.

Figure 2: Sample text from the first story.

In Figure 2 we give a sample of text from the corpus (from the first story).

3.2. Parses

The texts were parsed automatically using the Stanford Parser (Klein and Manning, 2003) and hand-corrected. Trace annotations were added by hand. We provide the resulting Penn Treebank-style phrase structure parse trees. We also provide Universal Dependencies-style parses (Nivre, 2015) automatically converted from the corrected parse trees using the Stanford Parser.

3.3. Self-Paced Reading Data

We collected self-paced reading (SPR) data (Just et al., 1982) for the stories from 181 native English speakers over Amazon Mechanical Turk. Text was presented in a dashed

Story	Title	Source Title	Source Author
1	Boar	The Legend of the Bradford Boar	E. H. Hopkinson
2	Aqua	Aqua, or the Water Baby	Kate Douglas Wiggin
3	Matchstick	The Little Match-Seller	Hans Christian Andersen
4	King of Birds	The King of the Birds	Brothers Grimm
5	Elvis	Elvis Died at the Florida Barber College	Roger Dean Kiser
6	Mr. Sticky	Mr. Sticky	Mo McAuley
7	High School	Bullies	Sarah Cleaves
8	Roswell	Roswell UFO incident	Wikipedia
9	Tulips	Tulip mania	Wikipedia
10	Tourette's	Tourette Syndrome Fact Sheet	NINDS

Table 1: Stories with titles and sources.

Story	# Words	# Sentences
1	1073	57
2	990	37
3	1040	55
4	1085	55
5	1013	45
6	1089	64
7	999	48
8	980	33
9	1038	48
10	938	43

Table 2: Summary of stories by length.

moving window display; spaces were masked. Each participant read 5 stories per HIT. 19 participants read all 10 stories, and 3 participants stopped after one story. Each story was accompanied by 6 comprehension questions. We discarded SPR data from a participant's pass through a story if the participant got less than 5 questions correct (89 passes through stories excluded). We also excluded RTs less than 100 ms or greater than 3000 ms. Figure 3 shows histograms of RTs per story.

3.3.1. Inter-Subject Correlations

In order to evaluate the reliability of the self-paced reading RTs and their robustness across experimental participants, we analyzed inter-subject correlations (ISCs). For each subject, we correlated the Spearman correlation of that subject's RTs on a story with average RTs from all other subjects on that story. Thus for each story we get one ISC statistic per subject. Figure 4 shows histograms of these statistics per story.

3.3.2. Psycholinguistic Sanity Checks

As a sanity check for our RT data, we checked that basic psycholinguistic effects obtain in it. Some of the most robust predictors of reading time are frequency, word length, and surprisal (Kliegl et al., 2004; Smith and Levy, 2013). More frequent words are read more quickly, longer words are read more slowly, and more surprising words (as determined using e.g. an n-model) are read more slowly. Here we check whether these well-known effects can be found in our SPR corpus.

To do this, we fit a regression models to predict reading time based on each of the three predictors individually. Specifically, we fit a model predicting reading time from log frequency, one predicting reading time from word length (measured in orthographic characters), and one predicting reading time from log probability under a trigram model. Word and trigram counts are collected from the Google Books n-grams corpus, summing over years from 1990 to 2013; we make these counts available with the corpus. Each regression is a mixed-effects regression with subject and story as random intercepts (models with random slopes did not converge), meaning that we control for by-subject and by-story variability.

Predictor	$\hat{\beta}$	Std. Error	t value
Log Frequency	-2.61	0.08	-32.27
Log Trigram Probability	-2.19	0.09	-23.90
Word Length	4.21	0.12	35.72

Table 3: Regression coefficients from individual mixed-effects regressions predicting RT for each of the three predictors log frequency, log trigram probability, and word length. We predict and find negative effects of log frequency and log probability and a positive effect of word length. All p values are < 0.001.

The results of the regressions are shown in Table 3; we report results from the maximal converging models. In keeping with well-known effects, increased frequency and trigram probability both lead to faster reading times, and word length leads to slower reading times. These results show that basic psycholinguistic effects are present in our SPR data.

3.4. Syntactic Constructions

Here we give an overview of the low-frequency or marked syntactic constructions which occur in the stories. We coded sentences in the Natural Stories corpus for presence of a number of marked constructions, and also coded 200 randomly selected sentences from the Dundee corpus for the same features. The features coded are listed and explained in Appendix A. Figure 5 shows the rates of occurrence for these marked constructions per sentence in the two corpora. From the figure, we see that the Natural Stories corpus has especially high rates of nonlocal VP conjunction, nonrestrictive SRCs, idioms, adjective conjunction, noncanonical ORCs, local NP/S ambiguities, and it-clefts.

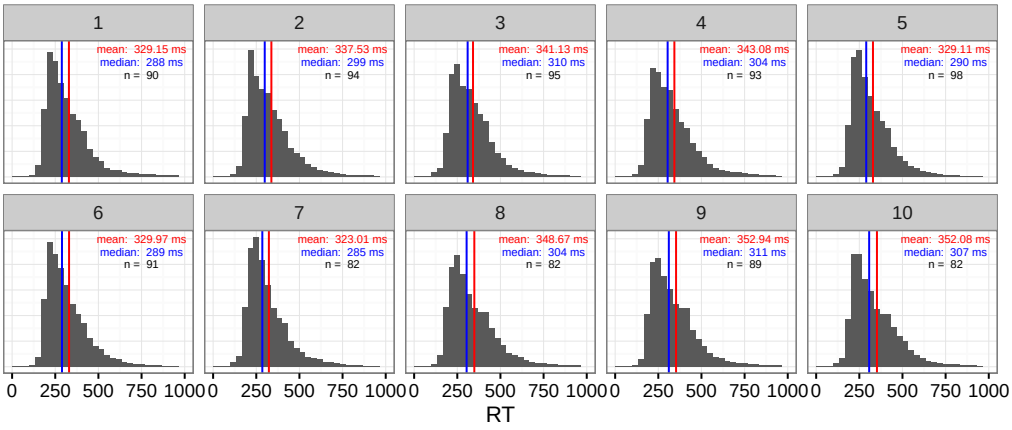


Figure 3: Histograms of SPR RTs per story, after data exclusion.

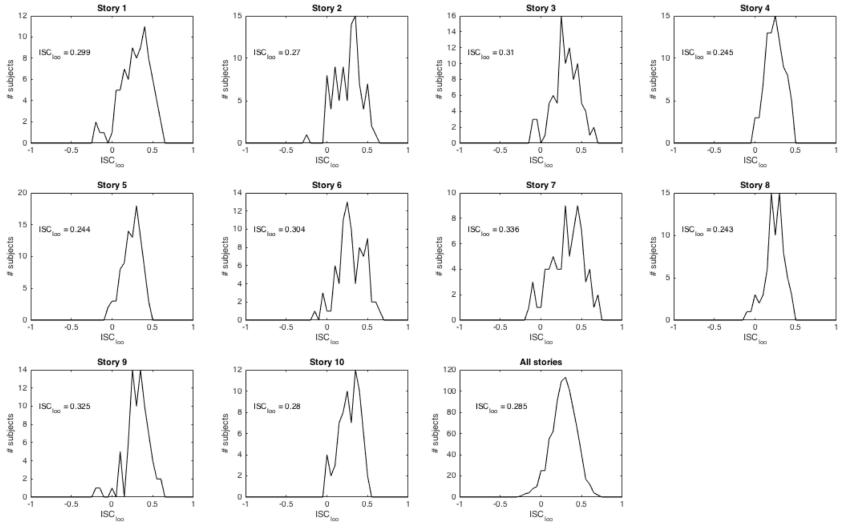


Figure 4: Leave-one-out Inter-Subject Correlations (ISCs) of RTs per story. In the panels, ISC_{loo} gives the average leave-one-out ISC for that story.

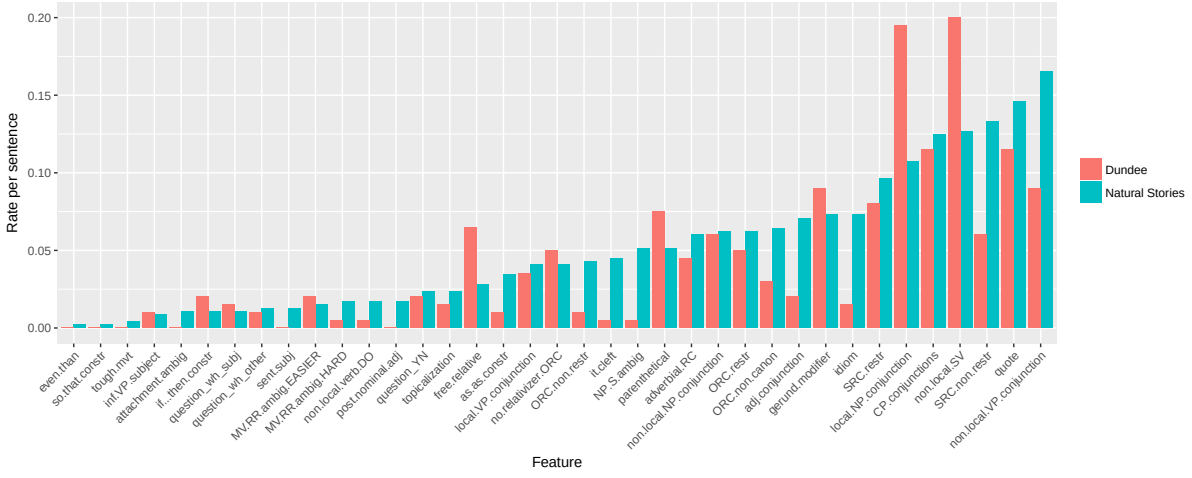


Figure 5: Rates of marked constructions in the Natural Stories corpus and in 200 randomly sampled sentences from the Dundee corpus.

4. Conclusion

We have described a new psycholinguistic corpus of English, consisting of edited naturalistic text designed to contain many rare or hard-to-process constructions while still sounding fluent. We believe this corpus will provide an important part of a suite of test sets for psycholinguistic models, exposing their behavior in uncommon constructions in a way that fully naturalistic corpora cannot. We also hope that the corpus as described here forms the basis for further data collection and annotation.

Acknowledgments

This work was supported by NSF DDRI grant #1551543 to R.F., NSF grants #0844472 and #1534318 to E.G., and NIH career development award HD057522 to E.F. The authors thank the following individuals: Laura Stearns for hand-checking and correcting the parses, Suniyya Waraich for help with syntactic coding, Cory Shain and Marten van Schijndel for hand-annotating the parses for traces, and Kyle Mahowald for help with initial exploratory analyses of the SPR data. The authors also thank Nancy Kanwisher for recording half of the stories (the other half was recorded by E.G.), Wade Shen for providing initial alignment between the audio files and the texts, and Jeanne Gallee for hand-correcting the alignment.

5. Bibliographical References

Bachrach, A., Roark, B., Marantz, A., Whitfield-Gabrieli, S., Cardenas, C., and Gabrieli, J. D. E. (2009). Incremental prediction in naturalistic langauge procesing: An fMRI study. Unpublished manuscript.

Barrett, M., Agić, Ž., and Søgaard, A. (2015). The Dundee treebank. In *The 14th International Workshop on Treebanks and Linguistic Theories (TLT 14)*, pages 242–248.

Boston, M. F., Hale, J., Kliegl, R., Patil, U., and Vasishth, S. (2008). Parsing costs as predictors of reading difficulty: An evaluation using the Potsdam Sentence Corpus. *Journal of Eye Movement Research*, 2(1).

Boston, M. F., Hale, J. T., Vasishth, S., and Kliegl, R. (2011). Parallel processing and sentence comprehension difficulty. *Language and Cognitive Processes*, 26(3):301–349.

Dambacher, M., Kliegl, R., Hofmann, M., and Jacobs, A. M. (2006). Frequency and predictability effects on event-related potentials during reading. *Brain Research*, 1084(1):89–103.

Demberg, V. and Keller, F. (2008). Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210.

Fossum, V. and Levy, R. (2012). Sequential vs. hierarchical syntactic models of human incremental sentence processing. In *Proceedings of the 3rd Workshop on Cognitive Modeling and Computational Linguistics*, pages 61–69. Association for Computational Linguistics.

Frank, S. L. and Bod, R. (2011). Insensitivity of the human sentence-processing system to hierarchical structure. *Psychological Science*, 22(6):829–834.

Frank, S. L., Monsalve, I. F., Thompson, R. L., and Vigliocco, G. (2013). Reading time data for evaluating broad-coverage models of English sentence processing. *Behavior Research Methods*, 45(4):1182–1190.

Frank, S. L., Otten, L. J., Galli, G., and Vigliocco, G. (2015). The ERP response to the amount of information conveyed by words in sentences. *Brain and Language*, 140:1–11.

Futrell, R., Mahowald, K., and Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Gibson, E. (2000). The dependency locality theory: A distance-based theory of linguistic complexity. In Alec Marantz, et al., editors, *Image, Language, Brain: Papers from the First Mind Articulation Project Symposium*, pages 95–126.

Husain, S., Vasishth, S., and Srinivasan, N. (2015). Integration and prediction difficulty in Hindi sentence comprehension: Evidence from an eye-tracking corpus. *Journal of Eye Movement Research*, 8(2).

Just, M. A., Carpenter, P. A., and Woolley, J. D. (1982). Paradigms and processes in reading comprehension. *Journal of Experimental Psychology: General*, 111(2):228.

Kennedy, A., Pynte, J., Murray, W. S., and Paul, S.-A. (2013). Frequency and predictability effects in the dundee corpus: An eye movement analysis. *Quarterly Journal of Experimental Psychology*, 66(3):601–618. PMID: 22643118.

Kennedy, A. (2003). The Dundee corpus [CD-ROM]. The University of Dundee, Psychology Department.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430.

Kliegl, R., Grabner, E., Rolfs, M., and Engbert, R. (2004). Length, frequency, and predictability effects of words on eye movements in reading. *European Journal of Cognitive Psychology*, 16:262–284.

Kliegl, R., Nuthmann, A., and Engbert, R. (2006). Tracking the mind during reading: The influence of past, present, and future words on fixation durations. *Journal of Experimental Psychology: General*, 135(1):12.

Lewis, R. L. and Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29(3):375–419.

Luong, M.-T., O'Donnell, T. J., and Goodman, N. D. (2015). Evaluating models of computation and storage in human sentence processing. In *CogACLL*, page 14.

Mitchell, J., Lapata, M., Demberg, V., and Keller, F. (2010). Syntactic and semantic factors in processing difficulty: An integrated measure. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 196–206.

Nivre, J. (2015). Towards a universal grammar for natural language processing. In *Computational Linguistics and Intelligent Text Processing*, pages 3–16. Springer.

Roark, B., Bachrach, A., Cardenas, C., and Pallier, C. (2009). Deriving lexical and syntactic expectation-based measures for psycholinguistic modeling via incremental

top-down parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 324–333. Association for Computational Linguistics.

Smith, N. J. and Levy, R. (2013). The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319.

Temperley, D. (2007). Minimization of dependency length in written English. *Cognition*, 105(2):300–333.

van Schijndel, M. and Schuler, W. (2015). Hierarchic syntax improves reading time prediction. In *Proceedings of NAACL*.

Wu, S., Bachrach, A., Cardenas, C., and Schuler, W. (2010). Complexity metrics in an incremental right-corner parser. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1189–1198.

Yan, M., Kliegl, R., Richter, E. M., Nuthmann, A., and Shu, H. (2010). Flexible saccade-target selection in Chinese reading. *The Quarterly Journal of Experimental Psychology*, 63(4):705–725.

A Syntactic features coded in Section 3.4.

Here we describe the syntactic features of the corpus which were reported in Section 3.4.. Where necessary, we give examples of each syntactic feature. We categorize the features into conjunction features, relative clause features, ambiguity features, displacement features, and miscellaneous.

Conjunction

- Local/nonlocal VP conjunction. Conjunction of VPs in which the head verbs are adjacent (local) or not adjacent (nonlocal). Local example: *The man sang and danced*. Nonlocal example: *The man sang a song and danced a dance*.
- Local/nonlocal NP conjunction. Conjunction of NPs in which the head nouns are adjacent (local) or not adjacent (nonlocal). Local example: *Rewarded with land and fame*. Nonlocal example: *The people of Bradford and the people who knew them.*
- Sentential conjunction. Conjunction of sentences. Example: *I sang and you danced*.
- CP conjunction. Conjunction of CPs with explicit quantifiers. Example: *I know that you are a doctor and that you are a criminal.*

Relative clauses

- Restrictive/nonrestrictive SRC. Subject-extracted relative clauses with either restrictive or nonrestrictive semantics. We marked relative clauses as restrictive if they served to restrict the domain of possible referents and nonrestrictive if they simply provided extra information. Restrictive example: *The man that knows Bob*. Nonrestrictive example: *The snow, which was white, fell everywhere*.
- Restrictive/nonrestrictive ORC. Object-extracted relative clauses with either restrictive or nonrestrictive semantics. Example: *The man that Bob knows*.
- No-relativizer ORC. An object-extracted relative clause without an explicit relativizer, e.g. *The man Bob knows*.
- Noncanonical ORC. An object-extracted relative clause where the subject of the relative clause is not a pronoun. Example: *The man that the woman knows*.
- Adverbial relative clause. An relative clause with an extracted adverbial. Example: *the valley where you would find the city of Bradford*.
- Free relative clause. A relative clause not modifying a noun. Example: *What I know is that Bob is a doctor.*

Ambiguity

- NP/S ambiguity. A local ambiguity where it is unclear momentarily whether a clause is an NP or the subject of a sentence. For example, in the sentence *I know Bob is a doctor*, after reading *I know Bob* it is not clear whether Bob is an NP object of *know* or the beginning of an embedded clause.
- Main Verb/Reduced Relative ambiguity (easy/hard). A local ambiguity between a main verb and a reduced relative clause. For example, *The horse raced past the barn fell*. We divide these into easy and hard cases based on the annotators' judgment about how confusing the local ambiguity is in context.
- PP attachment ambiguity. A global ambiguity where a PP could attach to one of two NPs. For example, in a sentence such as *The daughter of the colonel on the balcony*, it is not clear whether it is the daughter or the colonel who is on the balcony.

Displacement

- *Tough* movement. Cases where an adjective is modified by an infinitive verb phrase from which an object has been extracted. Example: *The point is hard to see*.
- Parentheticals. Additional material that interrupts or lies outside the syntactic structure of the rest of the sentence; constructions that would be marked as "parataxis" in Universal Dependencies. These do not necessarily have to be marked with orthographic parentheses. Example: *There was once, legend has it, a fearful boar*.
- Topicalization. Cases where an NP is moved to the front of a sentence to serve as its topic. Example: *The history of Korea, I know nothing about.*
- Question with *wh* subject. Questions with *wh*-movement of the subject. Example: *Who walked into the room?*
- Question with other *wh* word. Questions with *wh*-movement of anything other than the subject. Example: *Who did Bob see?*

Miscellaneous

- Nonlocal SV. The appearance of any material between a verb and the head of its subject. Example: *The man with the hat ran away.*
- Nonlocal Verb/DO. The appearance of any material between a verb and its direct object. Example: *The man ate quickly the sandwich.*
- Gerund modifiers. A case of a verb phrase modifying a noun. Example: *The man walking down the street is tall*.
- Sentential subject. A sentence where the subject is an embedded clase. Example: *The fact that Bob is a doctor is interesting*.
- Postnominal adjectives. Adjectives which follow their nouns. Example: *The moon, full and bright*.
- Idiom. Any idiomatic expression, such as *busy as beavers*.
- Quotation. Any directly-reported speech. Example: *The woman said "I am here"*.
- It-clefts. Example: *It was Mary that Bob saw*.
- even...than construction. Example: *Even taller than Mary.*
- if...then construction. Example: *If you go, then I go.*
- as...as construction. Example: *Bob was as angry as Mary.*
- so...that construction. Example: *Bob was so angry that he was shaking.*
- Yes-no Question. Example: *Is Mary here?*

Semi-automatic Korean FrameNet Annotation over KAIST Treebank

Younggyun Hahm, Sunggoo Kwon, Jiseong Kim, Key-Sun Choi
Semantic Web Research Center, School of Computing, KAIST
291 Daehak-ro, Yuseong-gu, Deajeon, South Korea
{hahmyg, fanafa, jiseong, kschoi}@kaist.ac.kr

Abstract

This paper describes a project for constructing FrameNet annotations in Korean over the KAIST treebank corpus to scale up the Korean FrameNet resource. Annotating FrameNet over raw sentences is an expensive and complex task, because of which we have designed this project using a semi-automatic annotation approach. This paper describes the approach and its expected results. As a first step, we built a lexical database of the Korean FrameNet, and used it to learn the model for automatic annotation. Its current scope, status, and limitations are discussed in this paper.

Keywords: FrameNet, Semantic Role Labeling, Corpus Annotation

1. Introduction

FrameNet is a large lexical database that has rich annotations to represent the meanings of text using semantic frames (Baker et al., 1998; Fillmore et al., 2003). FrameNet has been considered a useful resource for various applications such as question answering systems (Shen and Lapata, 2007, Hahm et al., 2016), information extraction (Surdeanu et al., 2003), and dialog systems (Chen et al., 2013). Lately, researchers have shown increasing interest in multilingual FrameNet (Borin et al., 2010; You and Lui, 2005; Meurs et al., 2008; Subirats and Petruck, 2003; Burchardt et al., 2006). A Korean FrameNet project built the Korean FrameNet resource by translating English FrameNet annotations and Japanese FrameNet into its equivalent in Korean (Park et al., 2014; Kim et al., 2016).

One of the purpose of the FrameNet annotation is to build a frame-semantic parser to understand the meaning of a text. Some studies have built frame-semantic parser for English by using full-text annotation and partially annotated exemplar sentences to train their models (Das et al., 2010; Swayamdipta et al., 2017; Yang and Mitchell, 2017). For example, the state-of-the-art frame-semantic parser uses nearly 139k exemplar sentences for training data and it generally introduces a 3–4 F1 gain for parsing (Yang and Mitchell., 2017). In comparison, the Korean FrameNet has full-text annotations for 5,025 sentences and 8,200 lexical units (LUs).

In this paper, we report an ongoing project to construct Korean FrameNet annotations to scale up the amount of annotations. A task to annotate the frame-semantics manually over raw sentences has been formalized by Ruppenhofer et al. (2006). It is an expensive and complex task, because the annotators would need to choose proper frame-semantics for each target word, and its corresponding frame elements for the arguments (1,221 frame-semantics are defined in FrameNet 1.7). Therefore, we have used the existing Korean resources to bootstrap the annotation task. A target corpus for the FrameNet annotation task is the KAIST treebank (Choi et al., 1994). It includes 31,086 sentences with morphological analysis, part-of-speech (POS) tagging, and dependency parsing. It is used in the Korean Universal Dependency treebank (Choi, 2013).

This paper briefly introduces the FrameNet annotation over the KAIST treebank project, and presents its scope in Section 2. The related research tasks are described in Section 3, which are separated into the current state of the research and ongoing tasks. The evaluation of the current progress is discussed in Section 4, and conclusions are presented in Section 5.

2. Problem Definition

2.1 Problem Statement and Workflow

The goal of the project is to annotate FrameNet over the KAIST treebank. Figure 1 shows the workflow of these tasks. In the first step, the semantic frame is automatically annotated over the target corpus, which is the KAIST treebank, by using a model learned from the Korean FrameNet.

First, the target identification module identifies the target words which evoke the frame-semantics, and then the frame identification module and the frame element identification module identify the proper frame-semantics for the given target and its corresponding arguments (subsection 3.2). These modules use the lexical units (LUs) and the full-text annotations in the Korean FrameNet. To support this purpose, we built the lexical database of the Korean FrameNet (subsection 3.1) as a follow-up of the previous study by Park et al. (2014). By this process, the FrameNet annotation over the KAIST treebank was conducted automatically, which is a *silver standard*.

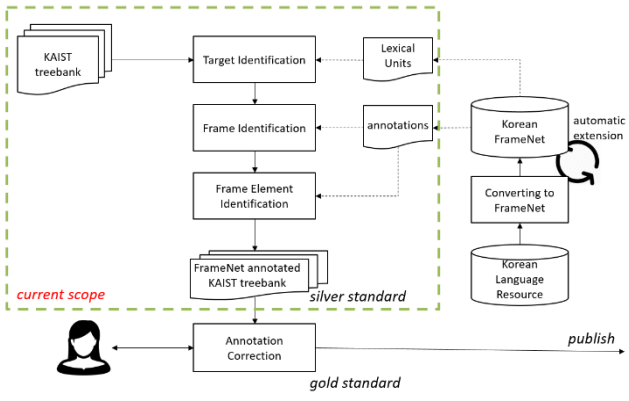


Figure 1: Workflow diagram of FrameNet annotation over the KAIST treebank.

To scale up the LUs in the Korean FrameNet, the task of converting the Sejong Electronic Dictionary (Hong, 2007) to the Korean FrameNet has been included as an ongoing task in this project (subsection 3.3). Then, the annotator would perform corrections of the *silver standard* annotations (subsection 3.4). In this paper, we report the

current scope and state of the project, and discuss the ongoing tasks and expected results. Figure 1 shows the workflow of this project.

2.2 Scope

In the FrameNet annotation tasks, the frame-semantics would be assigned to the target words. We considered general nouns, verbs, and adjectives alone in the KAIST treebank as the target words. The converted LUs from the Sejong Dictionary were not considered because it does not meet the gold standard yet. The annotation correction task is outside of the scope of this paper, however, the preparation tasks on ongoing works are briefly described.

3. Task Description

3.1 Building Korean FrameNet Database

Korean FrameNet is a resource that has been manually translated from English and Japanese FrameNet annotations into Korean. To use the Korean FrameNet, which was constructed in our previous work (Kim et al., 2016) as well as the training data, we first collected the LUs from the annotations. In the original annotations in English, the target words would be tokenized by white space; however, the translated Korean target words consist of multiple morphemes. For example, the target word 'visiting' is translated to '방문한', which would be tokenized into morphemes as a noun '방문'(visit), an adjective derivational suffix '하' which transforms the noun to the verb form, and adnominal ending 'ㄴ' which transforms the verb to the adjective form by combining it with other morphemes. To build a dictionary, we collected LUs along with their various form while retaining their grammatical and semantic meaning. We obtained all the target words from the annotations and then pruned specific morphemes, such as endings, *josa* (Korean postpositions), and affixes, as part of a lemmatization task in Korean. Then, 8,200 LUs were collected, which consisted of a lemma, its POS, and its corresponding frame-semantics. In this task, we corrected the errors in the processes, and performed the POS tagging and morphological analysis manually for 450 cases.

FrameNet is a lexical database that includes not only LUs but also syntactic realization patterns (i.e. valence patterns) for the frame elements of each LU. For example, an LU is 'visit' when it is used as a verb, and has a frame-semantics "`Visiting`"; the frame elements are annotated in full-text annotations, such as `agent`, `entity`, `place`, and so on. Moreover, these frame elements have a grammatical role in sentences (e.g., subject of the sentence), and have phrase types (e.g., noun phrase). These patterns are useful resources to the identify frame elements in a text. We parsed full-text annotations in the Korean FrameNet in the dependency syntax, and collected the valence patterns for each frame element of the LUs. For instance, the LU '`가르치.v.Educational_teaching`' (teach.v) consists of its lemma, POS, and the corresponding frame-semantics. It also has the valence patterns for each frame element; for example, the frame element `teacher` has the role of the subject in its annotations, and it has a *josa* '가,' which gives the nouns a subjective role. All LUs are updated on the Korean FrameNet webpage[1].

3.2 Automatic FrameNet Annotation

The FrameNet annotation task is generally separated into three steps (Das et al., 2010). A system 1) identifies a target word in the text, 2) identifies its proper frame-semantics, and then 3) identifies its frame elements.

As described in Section 2, the target words in the KAIST treebank are specified into three types—general nouns, verbs, and adjectives. Table 1 shows the number of target words of each type. For the target identification task, all words in the three types are considered as the target words.

Contents	Counts (total)	Counts (unique)
# of sentences	31,086	
# of general nouns	99,784	15,180
# of verbs	69,889	6,120
# of adjectives	26,559	1,807

Table 1: Statistics of the KAIST treebank

3.2.1 Frame Identification

Frame identification is a disambiguation task specified in the FrameNet terminology. First, the candidates for the frame-semantics are generated for a given target word, and then the most suitable frame-semantics is selected. In our task, for a given target word, a list of frame-semantics and their annotations were collected from the Korean FrameNet database that was built as described in subsection 3.1.

In the frame identification task, the frame-semantics of the target words would be disambiguated by their surrounding contexts (Baker et al., 1998). It means that if a given target word has a similar context as an LU in the Korean FrameNet, its proper frame-semantics would be a frame-semantics of the LU.

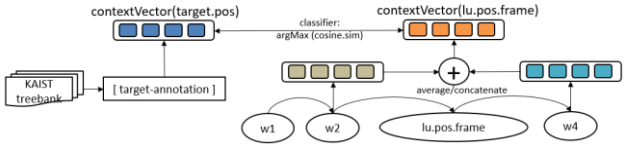


Figure 2: Identify the frame-semantics for a target word by comparing a context vector of the LU, which was learned from the Korean FrameNet annotations.

In this paper, to select a suitable frame-semantics for the target words, we have borrowed ideas from the concepts of synset embedding (Rothe and Schütze, 2015) and doc2vec (Le and Mikolov, 2014). Our model 1) learns the context vectors of each LU (called the *frame embedding*) from the full-text annotation in the Korean FrameNet, 2) generates the context vector for the given target word from the input text in the same way to generate the *frame embeddings*, 3) determines a similarity score between the context vectors of the target word and the *frame embeddings*, and then 4) chooses the most similar context vector and LU to get its

[1] http://framenet.kaist.ac.kr

frame-semantics. Figure 2 shows a snapshot of these operations.

In this paper, the context words are defined as 1-hop connected nodes with a target word in the dependency path, and the context vector is generated by averaging the sum of the word embeddings of each context word. The similarity score is calculated as the cosine similarity between the context vector of the target word and the *frame embeddings*.

Ongoing studies

The method described above uses a limited scope of the surrounding context of a target word and an LU. Hermann et al. (2014) uses a joint model with word embedding and syntactic structure, and Swayamdipta et al. (2017) uses rich syntactic features to identify the frame-semantics. Generating the *frame embeddings* from rich features would be the next step.

3.2.2 Frame Element Identification

As described in subsection 3.1, each LU has a list of valence patterns for its frame elements. In this paper, the valence patterns are applied by a rule-based approach. In other words, if a given target word in a sentence is disambiguated as a specific frame-semantics by the target identification task, its frame elements are identified if only it matched with the valence patterns. For example, if a given target word is 'teach.v' and its frame-semantics is `Educational_teaching`, then the frame element `teacher` is annotated for a phrase which is matched with the valence pattern for the grammatical function and its phrase type.

Ongoing studies

Identifying a boundary of frame elements and its type (i.e., the frame element tag) is still a challenge in the frame-semantic parsing task. Täckström et al. (2015) relied on the dependency features and some heuristic rules, and Yang and Mitchell (2017) used a joint model to jointly assign the frame-semantics and frame elements. In our project, our purpose of using automatic FrameNet annotation is to construct a *silver standard* corpus that would be corrected manually. Next, we are focused on the task that can identify frame elements well. To accomplish this, we are studying methods to generate valence patterns with richer syntactic features than using only grammatical functions and phrase types for high recall performance.

3.3 Automatic Expansion of FrameNet LUs

The Sejong Dictionary is known to have sufficient coverage for Korean corpus such as Wikipedia (Hahm et al., 2014). It includes not only a list of lexemes but also their predicate-arguments structure in PropBank style, their exemplar sentences, and English translations (words or phrases). Our project includes the task of automatic extension of the Korean FrameNet by converting the Sejong Dictionary to FrameNet. As a first step, we have chosen a representative word from the translations of a lexeme, and matched it with LUs in the English FrameNet by string matching approach (Levenshtein similarity > 0.95).

Ongoing studies

In this paper, we do not use the LUs derived from the Sejong Dictionary because it has not yet been manually validated, and the selection of a representative word from a translation is conducted by heuristic rules. Nevertheless, the Sejong Dictionary appears to be a useful resource that can improve the Korean FrameNet in terms of LU and exemplar sentences.

3.4 Manual Annotation Correction

To publish the resource, the FrameNet-annotated KAIST treebank corpus would be validated as a gold standard corpus to prevent error propagation issues in the training process. The result of the annotation correction is beyond the scope of this paper; however, the preparatory work for manual annotation correction is reported in this section.

WebAnno 3.2[2] (Biemann et al., 2017) is considered as the user interface. WebAnno provides a function to suggest proper candidate tags for a given word by using *constraints*. In the FrameNet annotation task, the frame-semantics candidates for a given word are shown at the top of the list of frame-semantics candidates, and it would prevent the annotators from wasting time searching for suitable frame-semantics tags. We generate the *constraints* from the Korean FrameNet database. For example, if a given word exist in the Korean FrameNet, its corresponding frame-semantics tags (i.e., annotated frame-semantics in Korean FrameNet full-text annotations) are shown at the top of the list of candidate tags.

4. Evaluation

For the evaluation of the results, we separate the Korean FrameNet into a training set and a test set. The test set consists of sentences that include one token LU (i.e. excluding the white space in the LU) categorized into three types—general nouns, verbs, and adjectives. The LUs in a test set have more than two frame-semantics candidates. The training set consists of 13,001 sentences, and the test set has 1,816 sentences.

Models	**Accuracy**
Random	67.29
Sentence Embedding	73.57
Frame Embedding	76.21

Table 2: Results of the frame identification task

Table 2 shows the results of the frame identification task that is described in subsubsection 3.2.1. The random model chooses the frame-semantics randomly from a list of candidates, and the sentence embedding model learns sentence embedding from the Korean FrameNet full-text annotations with the frame-semantics annotation on the target words. For example, in the sentence 'I go home,' the word 'go' is assigned to the frame-semantics `Motion`; the sentence embedding model learns the sentence embedding

[2] https://github.com/webanno/webanno

from a list ['I', 'go/Motion', 'home'] using doc2vec implementation[3]. This model learns all contexts in the sentence without a word order. The *frame embedding* model learns the context vectors for each LU in the Korean FrameNet (subsubsection 3.2.1), therefore, it would keep more relevant contexts as compared to the sentence embedding model. To generate the *frame embeddings*, we used the Korean Wikipedia[4] as a training corpus with the settings of dimensions = 100 and window size = 3.

Types	Total coverage	Word coverage
General Noun	47.82%	8.7%
Verb	68.42%	17.38%
Adjective	40.31%	6.03%

Table 3: Coverage of Korean FrameNet over the KAIST treebank

The frame embedding model also performed the annotation of frame-semantics over the KAIST treebank. For 31,086 sentences, 99,784 frame-semantics were annotated (3.42 per sentence), and 61,579 frame elements were annotated (0.62 per frame-semantics). Table 3 shows the results of the frame-semantics annotations in terms of coverage. Korean FrameNet has an overall coverage of about 57 percent; however, it does not show good coverage for each word shown in the KAIST treebank. It means that the scaling up of LUs is required to annotate FrameNet over the KAIST treebank, as described in subsection 3.3. When compared with the Korean FrameNet, 6.81 frame elements were annotated per frame-semantics in the Korean FrameNet. The task of increasing the coverage of the valence patterns also remains a challenge.

5. Conclusion

This paper describes the FrameNet annotation over the KAIST treebank, and discusses its scope and the current status. To bootstrap the annotation task, we designed the project using a semi-automatic annotation approach. We built the Korean FrameNet database, and used it to learn models to annotate automatically. We discovered that there are some limitations when using only Korean FrameNet, because of which the automatic extension of the Korean FrameNet task and the manual annotation correction task have been included in this project. Several ongoing studies are currently underway to address the challenges identified and described in this paper.

6. Acknowledgements

This work was supported by Institute for Information & Communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (R0124-16-0002, Emotional Intelligence Technology to Infer Human Emotion and Carry on Dialogue Accordingly)

7. Bibliographical References

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1* (pp. 86-90). Association for Computational Linguistics.

Biemann, C., Bontcheva, K., de Castilho, R. E., Gurevych, I., and Yimam, S. M. (2017). Collaborative Web-based Tools for Multi-layer Text Annotation. *The Handbook of Linguistic Annotation', Text, Speech, and Technology book series, Springer Netherlands*.

Borin, L., Dannélls, D., Forsberg, M., Gronostaj, M. T., and Kokkinakis, D. (2010). The past meets the present in Swedish FrameNet++. In *14th EURALEX international congress* (pp. 269-281).

Burchardt, A., Erk, K., Frank, A., Kowalski, A., Padó, S., and Pinkal, M. (2006). The SALSA corpus: a German corpus resource for lexical semantics. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006)* (pp. 969-974).

Chen, Y. N., Wang, W. Y., and Rudnicky, A. I. (2013). Unsupervised induction and filling of semantic slots for spoken dialogue systems using frame-semantic parsing. In *Automatic Speech Recognition and Understanding (ASRU)*, 2013 IEEE Workshop on (pp. 120-125). IEEE.

Choi, J. D. (2013). Preparing Korean data for the shared task on parsing morphologically rich languages. *arXiv preprint arXiv:1309.1649*.

Choi, K. S., Han, Y. S., Han, Y. G., and Kwon, O. W. (1994). KAIST tree bank project for Korean: Present and future development. In *Proceedings of the International Workshop on Sharable Natural Language Resources* (pp. 7-14).

Das, D., Schneider, N., Chen, D., and Smith, N. A. (2010). SEMAFOR 1.0: A probabilistic frame-semantic parser. *Language Technologies Institute, School of Computer Science, Carnegie Mellon University*.

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to framenet. *International journal of lexicography*, 16(3), 235-250.

Hahm, Y., Nam, S., and Choi, K. S. (2016). QAF: Frame Semantics-based Question Interpretation. In *Proceedings of the Open Knowledge Base and Question Answering Workshop*.

Hahm, Y., Kim, Y., Won, Y., Woo, J., Seo, J., Kim, J., Park, S., Hwang, D., and Choi, K. S. (2014). Toward matching the relation instantiation from DBpedia ontology to Wikipedia text: fusing FrameNet to Korean. In *Proceedings of the 10th International Conference on Semantic Systems* (pp. 13-19). ACM.

Hermann, K. M., Das, D., Weston, J., and Ganchev, K. (2014). Semantic Frame Identification with Distributed Word Representations. In *ACL 2014* (pp. 1448-1458).

Hong, J.S. (2007). Development Research Paper of 21 century Sejong Plan Electronic Dictionary. (11-1370252-000063-10), *the Ministry of Culture and Tourism, The National Institute of the Korean Language, 2007. (in Korean)*

Kim, J., Hahm, Y., and Choi, K. S. (2016). Korean FrameNet Expansion Based on Projection of Japanese FrameNet. In *COLING 2016*.

Kingsbury, P., and Palmer, M. (2002). From TreeBank to PropBank. In *LREC 2002* (pp. 1989-1993).

Le, Q., and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st*

[3] https://radimrehurek.com/gensim/models/doc2vec.html

[4] https://dumps.wikimedia.org/kowiki/20170601/

International Conference on Machine Learning (ICML-14) (pp. 1188-1196).

Meurs, M. J., Duvert, F., Béchet, F., Lefevre, F., and De Mori, R. (2008). Semantic Frame Annotation on the French MEDIA corpus. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).

Park, J., Nam, S., Kim, Y., Hahm, Y., Hwang, D., and Choi, K. S. (2014). Frame-semantic web: a case study for korean. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume 1272* (pp. 257-260). CEUR-WS. org.

Rothe, S., and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. *arXiv preprint arXiv:1507.01127*.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R., and Scheffczyk, J. (2006). FrameNet II: Extended theory and practice. *https://framenet.icsi.berkeley.edu/* .

Shen, D., and Lapata, M. (2007). Using Semantic Roles to Improve Question Answering. In *Emnlp-conll* (pp. 12-21).

Subirats, C., and Petruck, M. (2003). Surprise: Spanish FrameNet. In *Proceedings of CIL* (Vol. 17, p. 188).

Surdeanu, M., Harabagiu, S., Williams, J., and Aarseth, P. (2003). Using predicate-argument structures for information extraction. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1* (pp. 8-15). Association for Computational Linguistics.

Swayamdipta, S., Thomson, S., Dyer, C., and Smith, N. A. (2017). Frame-Semantic Parsing with Softmax-Margin Segmental RNNs and a Syntactic Scaffold. *arXiv preprint arXiv:1706.09528*.

Täckström, O., Ganchev, K., and Das, D. (2015). Efficient inference and structured learning for semantic role labeling. *Transactions of the Association for Computational Linguistics*, 3, 29-41.

Yang, B., and Mitchell, T. (2017). A Joint Sequential and Relational Model for Frame-Semantic Parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 1258-1267).

You, L., and Liu, K. (2005). Building chinese framenet database. In *Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on* (pp. 301-306). IEEE.

Handling Normalization Issues for Part-of-Speech Tagging of Online Conversational Text

Géraldine Damnati[1], Jeremy Auguste[2], Alexis Nasr[2], Delphine Charlet[1]
Johannes Heinecke[1], Frédéric Béchet[2]
(1) Orange Labs, Lannion, France
{geraldine.damnati, delphine.charlet, johannes.heinecke}@orange.com
(2) Aix-Marseille Univ, Université de Toulon, CNRS, LIS
{jeremy.auguste, alexis.nasr, frederic.bechet}@univ-amu.fr

Abstract

For the purpose of POS tagging noisy user-generated text, should normalization be handled as a preliminary task or is it possible to handle misspelled words directly in the POS tagging model? We propose in this paper a combined approach where some errors are normalized before tagging, while a Gated Recurrent Unit deep neural network based tagger handles the remaining errors. Word embeddings are trained on a large corpus in order to address both normalization and POS tagging. Experiments are run on Contact Center chat conversations, a particular type of formal Computer Mediated Communication data.

Keywords: Part of Speech Tagging, Computer Mediated Communication, Spelling Error Correction

1. Introduction

Contact Center chat conversation is a particular type of noisy user generated text in the sense that it is a formal Computer Mediated Communication (CMC) interaction mode. It shares some normalization issues with other CMC texts such as chatroom conversations or social media interactions but unlike the aforementioned cases, the professional context implies some specificities. For instance, contact center logs are hardly prone to Internet slang. Another characteristic is that they are dyadic conversations with asymmetric levels of orthographic or grammatical errors. Agents may write with mistakes but are usually recruited for their linguistic skills, and can rely on predefined utterance libraries. Customers on the other hand can make mistakes for several different reasons, be it their educational background, linguistic skills, or even the importance they pay to the social perception of the errors they would make. Some of them will make no mistake at all while some others will misspell almost every word.

The purpose of this paper is to perform POS tagging on this particular type of Noisy User Generated text. Our goal is to study to which extent it is worth normalizing text before tagging it or directly handling language deviations in the design of the tagger. We will show that a good compromise is to handle some of the errors through lexical normalization but also to design a robust POS tagger that handles orthographic errors. We propose to use word embeddings at both levels: for text normalization and for POS tagging.

2. Related work

Text normalization has been studied for several years now, with different perspectives over time. When studying SMS style language, researchers tried to handle new phenomena including voluntary slang shortcuts through phonetic models of pronunciation (Toutanova and Moore, 2002; Kobus et al., 2008). Recently, the effort has been more particularly set on Social Media text normalization with specific challenges on Twitter texts (Baldwin et al., 2015), which has been shown to be more formal (Hu et al., 2013) that what is commonly expected. The typology of errors is slightly different and most recent works focus on one-to-one lexical errors (replacing one word by another). The availability of large corpora has led to the design of normalization lexicons (Han et al., 2012) that directly map correct words to there common ill-formed variants. (Sridhar, 2015) learns a normalization lexicon and converts it into a Finite State Transducer. More recently, the construction of normalization dictionaries using word embeddings on Twitter texts were performed for Brazilian Portuguese (Bertaglia and Nunes, 2016). In this paper, we focus on out-of-vocabulary words. We propose to generate variants of such words using a lexical corrector based on a customized edit distance and to use word embeddings as distributed representations of words to re-rank these hypotheses thanks to contextual distance estimation.

In order to adapt POS tagging systems for noisy text, several approaches have proposed to use word clusters provided by hierarchical clustering approaches such as the Brown algorithm. (Owoputi et al., 2013) use word clusters along with dedicated lexical features to enrich their tagger in the context of online conversations. (Derczynski et al., 2013) use clustering approaches to handle linguistic noise, and train their system from a mixture of hand-annotated tweets and existing POS-labeled data. (Nasr et al., 2016) address the issue of training data mismatch in the context of online conversations and show that equivalent performance can be obtained by training on a small in domain corpus rather than using generic POS-labeled resources.

3. Text normalization

Our text normalization process operates in two steps, the first one produces in-lexicon variants for an out of lexicon form. The second one reranks the forms produced by the first step, using a distributional distance. The first step is based on a lexicon and an edit distance while the second relies on word embeddings. We focus on one-to-one normalization, avoiding the issue of agglutinations or split words.

3.1. Defining a target lexicon

In order to generate correct hypotheses of an out of vocabulary form, we need to define a target lexicon. A lexicon should both reflect general language common terms and company related specific terms. If the general common terms lexicon is very large, a lexical corrector would have more chances to propose irrelevant out of domain alternatives. Hence, we have chosen to reduce the size of our lexicon by selecting words that appear more than 500 times in the French Wikipedia, resulting in 36,420 words. Additionally a set of 388 manually crafted domain specific terms was added to the lexicon. The latter were obtained by selecting words in the manually corrected training corpus that were not covered by the general lexicon. Finally, as case is not a reliable information in such data we reduce all words of the lexicon to their lower case form. Contrastive experiments have been run but are not reported in this extended absract, showing that the choice of the lexicon is important for the whole process. Including a general knowledge lexicon from Wikipedia is more helpful for correcting Agent errors than for correcting Customer errors.

3.2. Edit-distance based normalization

The corrector built with this lexicon is based on the Damerau-Levenshtein (DL) distance. The code of the lexical corrector is available at the above mentioned url [1]. In contrast to standard DL we assign weights to error types: missing or superfluous diacritics only add 0.3 to the distance. Additionally, adjacent letters on the keyboard (like an `e` instead of an `r`, which sits just next to each other on QWERTY and AZERTY keyboards), add 0.9 to the edit-distance. Letter transpositions (such as `teh` instead of `the`) also account for 0.9. All other differences account for 1 in the global distance. These weights are configurable and have been optimized for our task.

Words to be processed are all transformed to their lower case form before applying the corrector with the lower case lexicon described in 3.1. The original case is reintroduced before applying the POS tagger.

The lexical corrector provides a list of candidates for correction, until a maximum cost is reach. This upper bound is proportional to the word length n in terms of number of letters and is computed as follows: $max_cost = n \times \gamma$

In these experiments γ is set to 0.3. Here again contrastive experiments can be provided showing the impact of the γ parameter.

As we are dealing with formal interactions,we did not apply the modification on the edit distance proposed by (Hassan and Menezes, 2013) where edit distance is computed on consonant skeletons, nor do we use Longest Common Subsequence Ratio (LCSR) as it didn't reveal to be helpful in our case.

3.3. Rescoring with word embeddings

The edit distance based variant generation process described above does not take into account the context of a word when generating variants. In order to take it into account, we propose to rescore the hypothesized alternatives using a distance metric derived from the cosine similarity between word embeddings. We have gathered a large amount of unannotated chat conversations from the same technical assistance domain, resulting in a 16.2M words corpus, denoted BIG. For the particular purpose of lexical normalization we are more interested in paradigmatic associations than in syntagmatic associations. Hence `word2vec` is used with a small window size of 4. Furthermore, in order to capture as many tokens as possible we have chosen to keep all tokens occurring at least twice in the corpus when learning the word embeddings. The lexicon produced contains 43.4K forms.

Let w be an observed form and $\alpha_i(w)$ be the i^{th} alternative proposed by the edit distance based lexical corrector. Let V_{emb} be the vocabulary of the word vector model estimated on the large unannotated corpus, and v_w denote the vector of word w. The word embeddings based distance $d_{emb}(w, \alpha_i(w))$ is defined as $1 - \cos(v_w, v_{\alpha_i(w)})$. If either v or $\alpha_i(w)$ does not belong to V_{emb}, $d_{emb}(w, \alpha_i(w))$ is set to 1, meaning that it will not have any effect on the re-scoring process. Let $C(w, \alpha_i(w))$ be the edit cost provided by the lexical corrector between w and the proposed alternative $\alpha_i(w)$, the rescoring process simply consists in multiplying the edit score by the distance derived from the embeddings.

$$C_{emb}(w, \alpha_i(w)) = C(w, \alpha_i(w)) \times d_{emb}(w, \alpha_i(w))$$

4. Part of speech tagging

The part of speech tagger used in our experiment is based on Gated Recurrent Units (GRU). GRUs, introduced by (Cho et al., 2014), are recurrent neural networks that work in a similar fashion than LSTMs. GRUs are simpler than LSTMs: they do not have an output gate, and the input and forget gates are merged into an update gate. This property allows GRUs to be computationally more efficient.

The ability of GRUs to handle long distance dependencies make them suitable for sequence labeling tasks, such as POS tagging. Our tagger uses a bidirectional GRU making use of past and future features for each specific word in a sentence. The bidirectionnal GRU consists of a forward layer and a backward layer which outputs are concatenated. The forward layer processes the sequence from the start to the end, while the backward layer processes it from the end to the start.

The input of the network is a sequence of words with their associated morphological and lexical features. The words are encoded using a lookup table which associates each word with its word embedding representation. These word embeddings can be initialized with pretrained embeddings and/or learned when training the model. For the morphological and typographic features, we use a boolean value for the presence of an uppercase character as the first letter of the word as well as the word suffixes of length 3 and 4 represented as onehot vectors. Finally, we also input as onehot vectors external lexicon information, constructed using the *Lefff* lexicon (Sagot, 2010). Such vectors represent the possible part-of-speech labels of a word. On the output layer, we use a softmax activation. During training, categorical

[1] https://github.com/Orange-OpenSource/lexical-corrector

cross-entropy is used as the loss function and the Adam optimiser (Kingma and Ba, 2014) is used for the gradient descent optimisation.

5. Experiments and results

The corpus used for our experiments has been extracted from chat conversation logs of a French technical assistance contact center. A set of 91 conversations has been fully manually corrected and POS tagged. This corpus has been split in two equal parts: the TRAIN part being used to train the POS tagger and the TEST part for evaluation. Both sets contain around 17K words, with 5.4K words from the Customer side 11.6K words from the Agent side.

The typology of errors follow the one proposed in (Nasr et al., 2016). `DIACR` stands for *diacritic* errors which are common in French, `APOST` for missing or misplaced *apostrophe*, `AGGLU` for *agglutinations* and `SPLIT` for words split into two words. It is common in French to find confusions `INFPP` between past participles and infinitives for verbs ending with `er` (`j'ai changé` ↔ `j'ai changer`). Morpho-syntactic inflection `INFL` in French is error prone as it is common that different inflected forms of a same word are homophones. `MOD1C` correspond to one modified character (substituted, deleted or inserted) or when two adjacent letters are switched.

5.1. Text Normalization Evaluation

In Table 1, we present the results of the text normalization steps, on the whole corpus. $editonly$ refers to text processed by the edit-based correction. $editembed$ refers to the full correction process with semantic rescoring based on word embedding distances. We show in the first line the amount of word errors that are potentially correctable by the proposed approach (*i.e.* errors leading to Out-of-Vocabulary words) and the remaining subset of word errors which can not be corrected by our approach (errors resulting in in-vocabulary words and words discarded from the correction process). Among the total amount of 1646 erroneous words, 53% (870) are potentially correctable. The other 47% are words that do appear in our lexicon. After the edit-based correction step, 76.7% of these errors have been corrected, leading to 202 remaining errors. When rescoring with semantic similarity, the $editembed$ approach enables to correct 5% additional words, leading to an overall correction of 81.6% of the potentially correctable errors. It is worth noticing that in our approach, as the lexicon used in the correction step is not exhaustive, we observe 80 added errors due to the fact that some words, which were correct in the raw text, but not present in the lexicon, have been erroneously modified into an in-vocabulary form. Overall, the word error rate (WER) on raw text was 4.37% and is reduced to 2.81% after $editonly$, and to 2.7% after $editembed$ semantic rescoring. When restricting the corpus to the Customer messages, the initial WER reaches 9.82% and the normalization process leads to 5.07%.

Detailed error numbers according to the type of errors, on TEST only, can be found in Table 3. As expected, the proposed approach is efficient for diacritics, apostrophes and 1 letter modifications (`DIACR, APOST, MOD1C`). However it is inefficient for agglutination `AGGLU` and `SPLIT`, for confusion of verbal homophonic forms (`INFPP`) and for inflexion errors (`INFL`).

	raw	$editonly$	$editembed$
# of correctable err.	870	202	160
# of non correctable err.	776	776	776
overall WER	4.37	2.81	2.70
CUST. WER	9.82	5.35	5.07
AGENT WER	1.73	1.58	1.55

Table 1: Evaluation of normalization. Number of correctable and non-correctable errors (second and third lines) and word error rates (lines four to six). Third and fourth columns indicate the errors after normalizing the text with respectively the edit distance based and the word embedding distance based normalization.

5.2. Part of speech tagging results

Three different taggers have been trained on the corrected version of the train corpus[2]. They differ in the embeddings that were used to represent the words. The first tagger does not use any pre-trained embeddings, the second uses embeddings trained on the raw corpus while the third one uses embeddings trained on the automatically corrected corpus. Three versions of the test corpus have been taken as input to evaluate the taggers. The raw version, the gold version, which has been manually corrected and the auto version, which has been automatically corrected. The accuracy of the three taggers on the three versions of the test corpus are represented in Table 2. POS accuracy has been computed on the whole TEST corpus as well as on subsets of the TEST corpus produced by the agents and the customers.

input	ALL	AGENT	CUST.
no pretrained embeddings			
gold	95.37	96.39	93.39
auto	93.83	95.52	90.54
raw	93.07	95.31	88.70
embeddings trained on raw corpus			
gold	95.36	96.37	93.40
auto	94.25	95.78	91.25
raw	94.01	95.77	90.60
embeddings trained on corrected corpus			
gold	95.35	96.35	93.42
auto	94.13	95.62	91.24
raw	93.43	95.52	89.37

Table 2: POS tagging accuracy of the three taggers on the test corpus. For each tagger results are given on three versions of the corpus: manually corrected (gold), automatically corrected (auto) and raw. The two last columns indicate accuracy on the Agent and Customer parts of the corpus.

Table 2 shows that the three taggers reach almost the same performances on the gold version of the TEST corpus. The

[2]Taggers trained on the raw versions of the corpus yielded lower results.

best performances on the raw TEST corpus are obtained by the second tagger, which word embeddings have been trained on the raw BIG corpus. This result does not come as a surprise since the raw TEST corpus contains spelling errors that could have occurred in the raw BIG corpus and therefore have word embedding representations. Although the tagger that uses pretrained word embeddings yields better results than the first tagger, it is still beneficial to automatically correct the input prior to tagging. Table 2 also shows that the benefits of using word embeddings trained on the raw BIG corpus is higher on the customer side, which was also expected since this part of the corpus contains more errors. Using embeddings trained on the automatically corrected BIG corpus doesn't yield any further improvements, suggesting that the initial embeddings trained on the raw corpus already capture the relevant information.

The influence of the spelling errors on the tagging process is analysed in Table 3. Each line of the table corresponds to one type T of spelling error. The left part of the table presents the results on the raw version of the test data and the right part, on its corrected version. The first column in each part is the total number of occurrences of type T errors, the second column is the number of type T errors that also correspond to a tagging error and the third column, the part of T errors that correspond to a tagging error (ratio of columns 1 and 2). The tagger used here is the second one, which uses word embeddings trained on raw data. Table 3 shows that the type of spelling error that is the more POS error prone is the `INFPP` type, which almost always lead to a tagging error. More generally, the table shows that the correction process tends to correct errors that are not very harmful to the tagger. This is especially true for the diacritic errors: the correction process corrects 67% of them but the number of tagging errors on this type of spelling errors is only decreased by 32.3%. Actually the remaining diacritic errors are typically errors on frequent function words in French that have different categories (`où` $\leftrightarrow$ `ou`, `à` $\leftrightarrow$ `a` meaning `where` $\leftrightarrow$ `or`, `to` $\leftrightarrow$ `have`).

	raw			auto		
ERR Type	Spell	Tag	ratio	Spell	Tag	ratio
DIACR	250	96	38.40	81	65	80.25
APOST	47	5	10.64	11	3	27.27
MOD1C	135	44	32.59	77	26	33.77
AGGLU	57	47	82.46	54	46	85.19
SPLIT	31	24	77.42	31	24	77.42
INFPP	29	26	89.66	29	26	89.66
INFL	84	9	10.71	77	8	10.39
OTHER	50	20	40.00	40	22	55.00

Table 3: POS tagging errors with respect to spelling error types, on raw and on corrected input. Column *Spell* is the number of errors of the corresponding type, Column *Tag* is the number of errors of the type that also correspond to tagging errors, column *ratio* is the ratio of column *Tag* and *Spell*.

6. Conclusion

We have shown in this paper that word embeddings trained on a noisy corpus can help for both tasks of correcting misspelled words and POS tagging noisy input. We have also quantified the impact of spelling errors of different categories on the POS tagging task. We plan as future work to combine both processes in a single one that performs both POS tagging and correction.

Acknowledgements

This work was partially funded by the French Agence Nationale pour la Recherche (ANR) under the grant ANR-15-CE23-0003

7. Bibliographical References

Baldwin, T., De Marneffe, M. C., Han, B., Kim, Y.-B., Ritter, A., and Xu, W. (2015). Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text (WNUT 2015), Beijing, China*.

Bertaglia, T. F. C. and Nunes, M. d. G. V. (2016). Exploring word embeddings for unsupervised textual user-generated content normalization. *Proceedings of the Workshop on Noisy User-generated Text (WNUT 2016), Osaka, Japan*.

Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013). Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *RANLP*, pages 198–206.

Han, B., Cook, P., and Baldwin, T. (2012). Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 421–432. Association for Computational Linguistics.

Hassan, H. and Menezes, A. (2013). Social text normalization using contextual graph random walks. In *ACL (1)*, pages 1577–1586.

Hu, Y., Talamadupula, K., Kambhampati, S., et al. (2013). Dude, srsly?: The surprisingly formal nature of twitter's language. In *Proceedings of the International AAAI Conference on Web and Social Media*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Kobus, C., Yvon, F., and Damnati, G. (2008). Normalizing sms: are two metaphors better than one? In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 441–448. Association for Computational Linguistics.

Nasr, A., Damnati, G., Guerraz, A., and Bechet, F. (2016). Syntactic parsing of chat language in contact center conversation corpus. In *17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 175.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL-HLT*, pages 380–390.

Sagot, B. (2010). The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *7th international conference on Language Resources and Evaluation (LREC 2010)*.

Sridhar, V. K. R. (2015). Unsupervised text normalization using distributed representations of words and phrases. In *Proceedings of NAACL-HLT*, pages 8–16.

Toutanova, K. and Moore, R. C. (2002). Pronunciation modeling for improved spelling correction. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 144–151. Association for Computational Linguistics.

Multi-Dialect Arabic POS Tagging: A CRF Approach

Kareem Darwish*, Hamdy Mubarak*, Mohamed Eldesouki*, Ahmed Abdelali*,
Younes Samih†, Randah Alharbi‡, Mohammed Attia⋆, Walid Magdy‡, Laura Kallmeyer†
*QCRI, †University of Dusseldorf, ‡University of Edinburgh, ⋆Google Inc.
{kdarwish,hmubarak,mohamohamed,aabdelali}@hbku.edu.qa,
samih@phil.hhu.de, s1581951@sms.ed.ac.uk, attia@google.com,
wmagdy@inf.ed.ac.uk, kallmeyer@phil.hhu.de

Abstract

This paper introduces a new dataset of POS-tagged Arabic tweets in four major dialects along with tagging guidelines. The data, which we are releasing publicly, includes tweets in Egyptian, Levantine, Gulf, and Maghrebi, with 350 tweets for each dialect with appropriate train/test/development splits for 5-fold cross validation. We use a Conditional Random Fields (CRF) sequence labeler to train POS taggers for each dialect and examine the effect of cross and joint dialect training, and give benchmark results for the datasets. Using clitic n-grams, clitic metatypes, and stem templates as features, we were able to train a joint model that can correctly tag four different dialects with an average accuracy of 89.3%.

Keywords: Arabic dialects, POS tagging, CRF

1. Introduction

Part-of-speech (POS) tagging is important for a variety of applications such as parsing, information extraction, and machine translation. Though much work has focused on POS tagging of Modern Standard Arabic (MSA), work on Dialectal Arabic (DA) POS tagging is rather scant with POS tagged corpora for most dialects being nonexistent or of limited availability. Dialectal POS tagging is becoming increasingly important due to the ubiquity of social media, where users typically write in their dialects to match how they speak in their daily interactions. Dialectal text poses interesting challenges such as lack of spelling standards, pervasiveness of transformative morphological operations, such as word merging and letter substitution or deletion, in addition to lexical borrowing from foreign languages. Existing work on dialectal POS tagging focuses on building resources and tools for each dialect separately (Duh and Kirchhoff, 2005; Habash et al., 2013). The rationale for the separation is that different dialects have different affixes, make different lexical and word ordering choices, and are influenced by different foreign languages. However, performing reliable dialect identification to properly route text to the appropriate POS tagger may be problematic, because conventional dialectal identification may lead to results lower than 90% (Darwish et al., 2014). Thus, building a POS tagger that performs reliably across multiple dialects without the need for dialect identification is desirable.

In this paper, we present new POS-tagging annotations on a dialectal dataset that is composed of social media text from Twitter for four major Arabic dialects, namely Egyptian (EGY), Levantine (LEV), Gulf (GLF), and Maghrebi (MGR) (Eldesouki et al., 2017; Samih et al., 2017a). For each dialect, we tagged 350 tweets using an extended version of the Farasa tagset (Darwish et al., 2017). We extended the tagset to account for tweet-specific tokens, namely hashtags, user mentions, emoticons and emojis, and URL's. We created 5-fold partitions for cross-validation with 70/10/20 train/dev/test splits for each dialect. We used the new dataset to train dialectal POS taggers for each dialect separately to test the effectiveness of the taggers on test data from the same dialect or from different dialects. We also experimented with cross-dialect and joint training to see if POS tagging of one dialect can benefit from data from other dialects. We show that joint models can perform on average at par with dialect specific models. For all our experiments we used a Conditional Random Fields (CRF) sequence labeler.

The contributions of this paper are as follows:

1. We present new dialectal POS tagging annotations on a multi-dialectal tweet dataset.
2. We report benchmark results on DA POS tagging.
3. We show that we can develop an effective joint model for POS tagging of different dialects without the need for dialect identification or dialect specific models.

2. Background

The scarcity of dialectal resources have hampered research in the area of DA, despite efforts from large institutions (ex. LDC) and programs (ex. TIDES, GALE and BOLT). Limited resources were made available for researchers with limited size and coverage. CallHome Egyptian Colloquial Arabic (ECA)[1] was the first attempt for a corpus to address this shortage released in 1997. The corpus is a collection of transcripts that cover five to ten minute segments taken from 120 unscripted telephone conversations between native speakers of Egyptian Arabic. Levantine Colloquial Arabic[2] as well as the Iraqi Arabic Conversational Telephone Speech[3] are two additional resources to cover dialectal Arabic that were built between 2004 and 2006 (Maamouri et al., 2004). The most recent dataset in this

[1] https://catalog.ldc.upenn.edu/LDC97T19
[2] https://catalog.ldc.upenn.edu/LDC2005S14
[3] https://catalog.ldc.upenn.edu/LDC2006T16

series was BOLT Egyptian Arabic SMS/Chat and Transliteration dataset[4], which is considerably the largest resource for dialectal Arabic (over a million words) even though it covers only Egyptian. The availability of these resources is stinted given the license requirement. On the other hand, researchers used ad-hoc resources or small datasets that were curated locally and not widely available. Graja et al. (2010) created the Tunisian Dialect Corpus Interlocutor (TuDiCoI) which contains 893 utterances, 3,404 words from dialectal conversations between Tunisian railway staff. Bouamor et al. (2014) used a collection of 2,000 sentences in Egyptian dialect as a seed to build a multi-dialectal Arabic corpus. The seed sentences were translated by native speaker to their own dialects to create a parallel corpus of Standard Arabic, Egyptian, Tunisian, Jordanian, Palestinian and Syrian Arabic, in addition to English. Cotterell and Callison-Burch (2014) extended the work of Al-Sabbagh and Girju (2010) and Zaidan and Callison-Burch (2011) to build a larger collection of commentaries from five Arabic newspapers and tweets for automatic dialect identification. Duh and Kirchhoff (2005) used ECA to build a POS tagger for Egyptian with the support of the MSA ATB. An accuracy of 69.83% with a coverage of 74.10% was achieved. Habash et al. (2013) released a new adaptation for MADA (Roth et al., 2008), extending it to cover Egyptian as well.

The emergence of social media platforms and their support to languages other than Latin -bidirectional/left-to-right or right-to-left helped dialectal presence to be more apparent than ever. This created a new need for newer resource with wider coverage. Our work attempts to fill some of this gap by providing a collection of tweet data that covers 4 major Arabic dialects as well as POS annotation of the data, which is unique and a first to be open for researchers.

3. Data Description

We used the dialectal Arabic dataset described by Eldesouki et al. (2017) and Samih et al. (2017b), which includes a set of 350 tweets for four major Arabic dialects that were manually segmented. The size of the dataset is as follows:

Dialect	No of Tweets	No of Words
Egyptian (EGY)	350	7,481
Levantine (LEV)	350	7,221
Gulf (GLF)	350	6,767
Maghrebi (MGR)	350	6,400

The words in the dataset were segmented in place without any modification or standardization attempts (ex. CODA (Habash et al., 2012)), and the segmentation guidelines aimed to generate a number of segments that match the correct number of POS tags for a word.

We used the POS tagset described by Darwish et al. (2017) which has 18 tags for MSA POS tagging, and we added 2 dialect-specific tags (namely PROG_PART, and NEG_PART), and 4 tweet-specific tags (namely HASH, EMOT, MENTION, and URL). Table 1 contains description of the newly added tags[5].

[4]https://catalog.ldc.upenn.edu/LDC2017T07

[5]Buckwalter transliteration is used in the paper

POS	Description	Example
PROG_PART	Progressive Part.	بنكتب (**b**nktb) "we are writing"
NEG_PART	Negation Part.	ماكانش (mAkAn**$**) "he was not"
HASH	Hashtag	#يا_رب (#yA_rb) "#O_God"
EMOT	Emoticon/Emoji	:)
MENTION	Mention	@mohamedAli
URL	URL	http://t.co/EF5cW

Table 1: Dialect-specific and tweet-specific POS tags

Segmentation and POS tagging were applied on the original raw text without any correction as suggested by Eldesouki et al. (2017) to overcome the need for standardization of different dialectal writings proposed in CODA by Habash et al. (2012). For example the word ومبيقولوش (wmbyqwlw$) "and they are not saying" is segmented as و+م+ب+يقول+و+ش (w+m+b+yqwl+w+$) and tagged as: CONJ+PART+PROG_PART+V+PRON+NEG_PART.

Words are white-space and punctuation separated while hashtags, emotions, mentions and URL's are considered as single words without internal segmentation. Data is formatted in CoNLL format: Words are split into tokens (clitics), and POS is provided for each token. In our annotation scheme, tokens, words, and sentences are separated by token boundary tag (TB), word boundary tag (WB), and end of sentence tag (EOS) respectively as shown in Table 2. Tagging was performed by a native speaker for each dialect. Then, multiple rounds of quality control and revision were performed to obtain high accuracy and consistency across dialects.

Index	Token	POS
0	ب (b) (present cont. particle)	PROG_PART
0	TB	TB
0	حب (Hb) "I love"	V
0	WB	WB
1	اسمع (AsmE) "I listen"	V
1	WB	WB
..	..	..
n	EOS	EOS

Table 2: Data format for segmentation and POS tagging

Figure 1 compares the distribution of POS tags in the four dialects against a sample of 350 MSA sentences from ATB with similar number of words (7,385 words). From this figure, some interesting observations can be made. For example, the four dialects are generally similar to each other in their POS distribution, while MSA shows substantial divergence. For example, nouns, adjectives, prepositions, numbers, and definite articles appear more frequently in MSA than in dialects, while on the other hand dialects show higher frequency of verbs, pronouns and particles. Our justification for this noticeable disparity is that the POS distribution is affected by the genre. The MSA text is from the formal news domain with a special focus on facts and entities, while the dialects are informal expressions with a focus on events, attitudes, and conversations. Another observa-

tion is that MSA has more noun suffixes and grammatical case endings, while dialects have more progressive particles and negation suffixes. This variance is related more to the linguistic nature of the language rather than the genre.

4. Experiments and Evaluation

4.1. Experimental Setup

For the experiments that we conducted, we used the CRF++ implementation of a CRF sequence labeler with L2 regularization and default value of 10 for the generalization parameter "C". We conducted three sets of experiments.

In the **first**, we tested the effectiveness of different features including different size contexts, metatypes, and stem templates, which we describe later.

In the **second**, we used the training and dev parts for each split for every dialect for training and then we used the test parts for all dialects testing. In these experiments, we were interested in knowing the POS tagging effectiveness when training and testing data are from the same dialect or from different dialects. High results for cross dialect training and testing may indicate closeness between dialects.

In the **third** set of experiments, we trained on all the train and dev parts of all the dialects jointly and tested on test sets of all the dialects. We were interested in determining if POS tagging from one dialect can benefit from added training data from other dialects.

For our experiments, given a sequence of clitics $c_n...c_{-2}, c_{-1}, c_0, c_1, c_2...c_m$, where we assumed perfect segmentation, we used the following features for each clitic (c_0):

- **Clitic n-grams.** a combination of clitic unigram features $\{c_{-1};c_0;c_1\}$, bigram features $\{c_{-2}^{-1};c_{-1}^{0};c_0^1;c_1^2\}$, and alternatively trigram features $\{c_{-2}^{0};c_{-1}^{1};c_0^2\}$ and 4-gram features $\{c_{-3}^{0};c_{-2}^{1};c_{-1}^{2};c_0^3\}$.

- **Clitic metatypes.** We defined a set of 10 "metatypes" that we heuristically determined. They include: Hashtag (if clitic starts with "#"); Mention (if clitic starts with "@"); URL (if clitic starts with "http"); Emoticon/emoji (if it appears in a list of 2,730 emoticons/emojis that we constructed); Retweet (if the clitic is "RT"); Foreign (if it contains non-Arabic letters); Number (if it matches Arabic or Hindi numerals or a gazetteer of written out numbers that we obtained from Farasa (Abdelali et al., 2016)); Punctuation (if it matches punctuations in the UTF8 codepage); Arabic (if it contains Arabic letters only); and Other for all other clitics. Using metatypes was shown to be effective for MSA POS tagging (Darwish et al., 2017).

- **Clitic stem templates.** Arabic words are typically derived from a closed set of roots that are placed in so-called stem templates to generate stems. For example, the root *ktb* can be fit in the template *CCAC* to generate the stem *ktAb* (book). Stem templates may overwhelmingly have one POS tag (e.g., *yCCC* is overwhelmingly a V) or favor one tag over another (e.g., *CCAC* is more likely a NOUN than an ADJ). This was shown to be effective for MSA POS tagging (Darwish et al., 2017), and we were curious to see if this would be effective for dialects also, particularly given the overlap between MSA and dialectal Arabic. We used Farasa to determine stem templates (Abdelali et al., 2016).

For all the experiments, we trained on the training and dev parts and tested on the test part. As mentioned earlier, we also randomly selected 350 MSA sentences from Arabic Penn Treebank (ATB) and treated MSA as a language variety. Doing so would allow us to observe the divergence of dialects from MSA and the relative effectiveness of using a small dataset compared to much more data.

4.2. Evaluation

We conducted the the following sets of experiments:

Set 1: In this set, we examined the effectiveness of the different features by when training and testing on the same dialect. We evaluated the following feature sets:

- Baseline (BL): clitic n-grams only, where we used the aforementioned combination of clitic unigrams, bigrams, and trigrams.
- Baseline + stem template (+ST)
- Baseline + metatype (+MT)
- A combination of clitic unigrams and bigrams + stem template + metatype (+ST+MT (2g))
- A combination of clitic unigrams, bigrams, and trigrams + stem template + metatype (+ST+MT (3g))
- A combination of clitic unigrams, bigrams, trigrams, and 4-grams + stem template + metatype (+ST+MT (4g))

Table 3 reports on the word-level accuracy for the different experiments. To demonstrate the difference between word-level accuracy, which we report here, and clitic-level accuracy, consider the phrase هـ+يلعب ب+ال+كورة (h+ylEb b+Al+kwr+m – "he will play with the ball") with correct POS tags FUTURE_PART+V PREP+DET+NOUN+NSUFF. If tagger erroneously tagged the phrase as FUTURE_PART+V PREP+DET+ADJ+NSUFF, then word-level accuracy would be 1/2 while clitic level accuracy would be 5/6. Since we used 5-fold cross validation, we report on the average across all folds. As the results show, using clitic n-grams only yielded the lowest results. Using stem template and metatype features improved results over using clitic n-grams alone with the combination of both features leading to even greater gain. When combined with stem template and metatype features, a combination of clitic unigrams and bigrams (2g) yielded the best results edging the use of higher order n-grams.

Set 2: Next, we were interested in determining cross-dialect training results to see if dialects can learn from each other and whether models from one dialect can generalize to other dialects. For all experiments, we used a combination of clitic unigrams and bigrams with stem template and metatype features (+ST+MT (2g)). Table 4 reports on cross-dialect results. Not surprisingly, the best results for

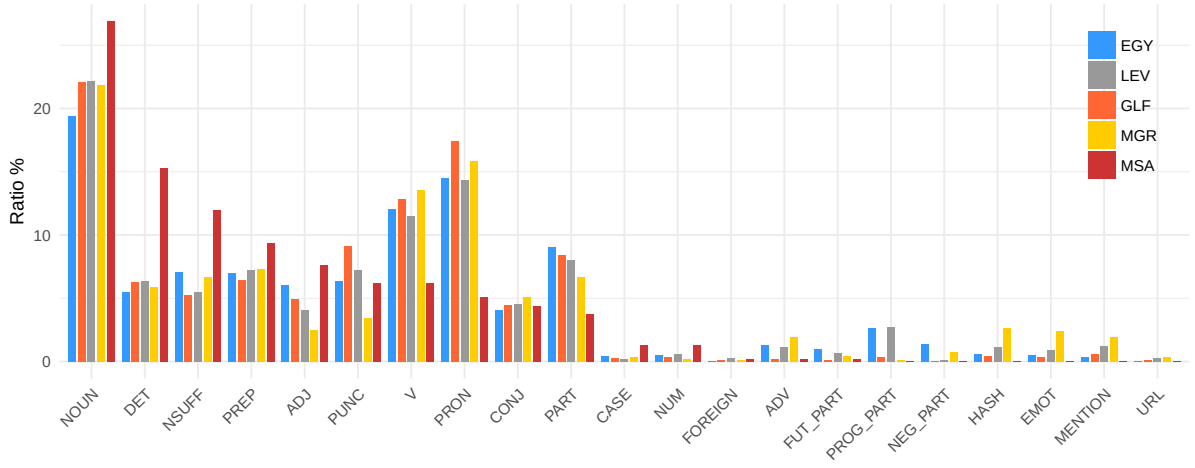


Figure 1: Distribution of POS tags per dialect and MSA. Tags are ordered according to average usage in MSA and DA

each dialect were obtained when the training and test sets were from the same dialect. Among the different dialects (excluding MSA), Maghrebi suffered the most when the training set was from another dialect, and conversely training on Maghrebi yielded the worst results for all the other dialects. This may indicate that Egyptian, Levantine, and Gulf were closer together and Maghrebi was most dissimilar to all of them. Further, training on MSA and testing on dialects yielded significantly lower results compared to training on dialects and testing on MSA. This may imply that the affixes that we observed in dialects are a superset of those observed in MSA.

Set 3: Lastly, we were curious to see if dialects can benefit from the addition of data from other dialects during training. Thus we combined the training and dev parts for all dialects and tested on the test part of each dialect. Table 5 reports on the results of joint learning with and without the inclusion of MSA. As the results show, Egyptian and Levantine benefited from the additional training data, while Maghrebi, Gulf, and MSA did not. The difference in accuracy (either positive or negative) ranged between 0.2% and 0.7%. On average across dialects only, the addition of MSA data marginally affected POS tagging effectiveness for different dialects. We suspect that if we use MSA tweets, instead of news sentences that we obtained from ATB, to match the genre of the dialect data would lead to greater improvement. Having a joint model that performs at par or better than dialect specific models across dialects is highly advantageous as it would avoid the need for dialect identification. The results show that a joint model may outperform dialect-specific models.

5. Error Analysis

To evaluate the strengths and weaknesses of our system, we analyzed the predicted results and the various types of errors made by the system. For such, we assessed the top 10 error types for each dialect and MSA for which correct POS of whole words is different than the guessed POS by the system. These errors represent 74%, 70%, 78%, 72% and 85% of all errors in EGY, LEV, GLF, MGR and MSA respectively. Next, we compiled together these errors and sorted them according to their average. Results are shown in Figure 1. On the average, 50% of all errors in DA are due

	MSA	EGY	LEV	GLF	MGR
BL	90.0	88.8	80.8	81.9	81.7
+ST	92.8	91.4	84.3	85.9	84.2
+MT	90.7	90.3	84.5	83.5	86.1
+ST+MT (2g)	**93.6**	**92.9**	**87.9**	**87.8**	**88.3**
+ST+MT (3g)	93.1	92.4	87.5	87.3	88.0
+ST+MT (4g)	92.5	91.6	86.8	86.6	87.2

Table 3: Per dialect training: baseline (BL), stem templates (+ST), and metatypes (+MT), and combined (+ST+MT) varying clitic n-grams (2g, 3g, and 4g).

	Training Set				
Test Set	MSA	EGY	LEV	GLF	MGR
MSA	**93.6**	76.1	76.1	76.9	72.7
EGY	54.5	**92.9**	74.3	78.1	72.7
LEV	52.0	74.7	**87.9**	73.5	69.8
GLF	58.7	78.8	76.7	**87.8**	74.4
MGR	50.8	71.1	73.1	70.1	**88.3**
Avg	61.9	**78.7**	77.6	77.3	75.6

Table 4: Cross dialect training using clitic bigrams (2g), stem templates, and metatypes as features.

to incorrect classification of nouns as verbs or adjectives and vice versa. This ratio increases to 65% in MSA.

Table 6 shows examples of the top error types across all dialects and MSA which represent 71% of all errors. Some of these errors are due to the fact that the words were not seen in training data. However, many of these words, such as منطقية (mnTqyp – "logical") and شوارع ($wArE – "streets"), are words that overlap between MSA and dialects and would exist in a large MSA corpus, such as ATB. Habash and Rambow (2006) and Mubarak (2017) reported an overlap of 60% for LEV verbs and 66% for EGY respectively. This suggests that domain adaptation with MSA data or using word embeddings that are trained on a large Arabic corpus would help overcome such errors.

Another source of errors is due to the lack of writing standards and handling words without correcting their spelling mistakes as in words وإسم (w<sm – "and name") and مأساه (m>sAh – "tragedy").

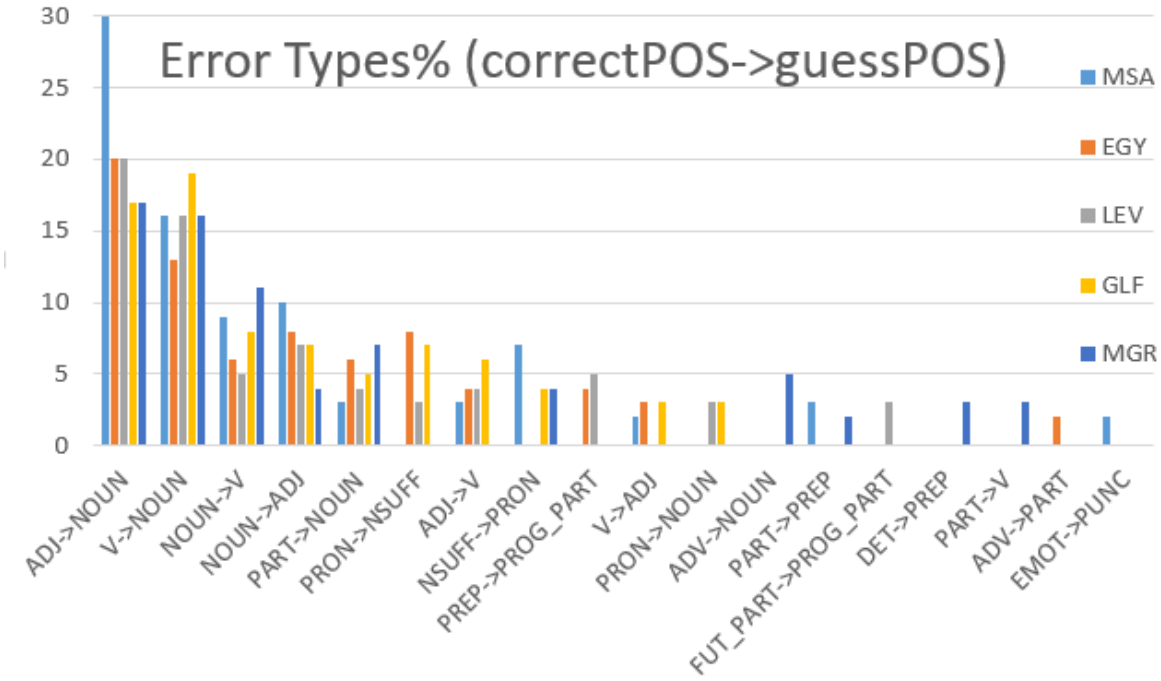


Figure 2: Distribution of Error types

	on self	joint (DA only)	joint (MSA&DA)
MSA	**93.6**	82.5	92.5
EGY	92.9	93.2	**93.4**
LEV	87.9	**88.6**	**88.6**
GLF	**87.8**	87.2	87.4
MGR	**88.3**	87.7	87.6
Avg (DA only)	89.2	89.2	**89.3**
Avg (MSA&DA)	**90.1**	87.8	89.9

Table 5: Results of joint learning

6. Conclusion and Future Work

In this paper, we introduced a new dataset for POS tagging of four major Arabic dialects that we constructed from tweets. We plan to provide the data freely to the community including our training, dev, and test splits. We also built POS taggers for the four dialects using a CRF sequence labeler using clitic n-gram features, stem templates, and clitic metatypes. Further, we show that we can train a joint model using data from all the dialects to train a POS tagger with comparable results to mono-dialectal training and testing, alleviating the need for dialect identification prior to POS tagging.

For future work, we plan to explore two distinct directions, namely:

- the use Brown clusters (Brown et al., 1992). Brown clustering is a hierarchical clustering of words based on their context and produces a kind of word embeddings that can be learned from large unlabeled texts. The rationale for using it here is that similar words, particularly those that share the same POS tag, tend to appear in similar contexts (Owoputi et al., 2013; Stratos and Collins, 2015).
- the use of Deep Neural Networks (DNN). DNNs have the advantage of alleviating the need for specific feature engineering including long distance relationships. Further, character-level models may be able to learning morphological patterns automatically.

7. Bibliographical References

Abdelali, A., Darwish, K., Durrani, N., and Mubarak, H. (2016). Farasa: A fast and furious segmenter for arabic. In *HLT-NAACL Demos*, pages 11–16.

Al-Sabbagh, R. and Girju, R. (2010). Mining the web for the induction of a dialectical arabic lexicon. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, pages 288–293.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A multidialectal parallel corpus of arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1240–1245.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Cotterell, R. and Callison-Burch, C. (2014). A multi-dialect, multi-genre corpus of informal written arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 241–245.

Darwish, K., Sajjad, H., and Mubarak, H. (2014). Verifiably effective arabic dialect identification. In *EMNLP*, pages 1465–1468.

Darwish, K., Mubarak, H., Abdelali, A., and Eldesouki, M. (2017). Arabic pos tagging: Dong't abandon feature en-

Error Type	Avg%	Example (Dialect)
ADJ->NOUN	21	غير منطقية (gyr mnTqyp) “not logical” (EGY)
V->NOUN	16	أعتقد ما في مجال (AEtqd mA fy mjAl) “I believe there is...” (LEV)
NOUN->V	8	لا شوارع (lA $wArE) “no streets” (GLF)
NOUN->ADJ	7	المغرب والجزائر خاوة (Almgrb wAljzAyr xAwp) “Morocco and Algeria are brothers” (MGR)
PART->NOUN	5	مثلما يستطيع الكثير (mvlmA ystTyE Alkvyr) “as many can do” (MSA)
PRON->NSUFF	4	عاجبه حالـه (Ejbh hAl+h) “likes his circumstances” (EGY)
ADJ->V	3	احنا فاهمين (AHnA fAhmyn) “we understand” (LEV)
NSUFF->PRON	3	عايشين مأساه (Ey$yn m>sA+h) “living in a tragedy” (EGY)
PREP->PROG-PART	2	عاملها بهايدك (EAmlhA b+Aydk) “you did it by your hand” (LEV)
V->ADJ	2	كي يصير (ky ySyr) “to become” (MSA)

Table 6: Error types examples from different DAs and MSA

gineering just yet. *WANLP 2017 (co-located with EACL 2017)*, page 130.

Duh, K. and Kirchhoff, K. (2005). Pos tagging of dialectal arabic: a minimally supervised approach. In *Proceedings of the acl workshop on computational approaches to semitic languages*, pages 55–62. Association for Computational Linguistics.

Eldesouki, M., Samih, Y., Abdelali, A., Attia, M., Mubarak, H., Darwish, K., and Kallmeyer, L. (2017). Arabic multi-dialect segmentation: bi-lstm-crf vs. svm. *arXiv:1708.05891v1 [cs.CL]*.

Graja, M., Jaoua, M., and Hadrich Belguith, L. (2010). Lexical study of a spoken dialogue corpus in tunisian dialect. In *The international arab conference on information technology (ACIT), benghazi–libya*.

Habash, N. and Rambow, O. (2006). Magead: a morphological analyzer and generator for the arabic dialects. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 681–688. Association for Computational Linguistics.

Habash, N., Diab, M. T., and Rambow, O. (2012). Conventional orthography for dialectal arabic. In *LREC*, pages 711–718.

Habash, N., Roth, R., Rambow, O., Eskander, R., and Tomeh, N. (2013). Morphological analysis and disambiguation for dialectal arabic. In *Hlt-Naacl*, pages 426–432.

Maamouri, M., Buckwalter, T., and Cieri, C. (2004). Dialectal arabic telephone speech corpus: Principles, tool design, and transcription conventions. In *NEMLAR International Conference on Arabic Language Resources and Tools, Cairo*, pages 22–23.

Mubarak, H. (2017). Analysis and quantitative study of egyptian dialect on twitter. In *17TH INTERNATIONAL CONFERENCE ON WEB ENGINEERING*. ICWE 2017.

Owoputi, O., O’Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL-HLT 2013*, pages 380–390. Association for Computational Linguistics.

Roth, R., Rambow, O., Habash, N., Diab, M., and Rudin, C. (2008). Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. *Proceedings of the Conference of American Association for Computational Linguistics (ACL08)*.

Samih, Y., Attia, M., Eldesouki, M., Mubarak, H., Abdelali, A., Kallmeyer, L., and Darwish, K. (2017a). A neural architecture for dialectal arabic segmentation. *WANLP 2017 (co-located with EACL 2017)*, page 46.

Samih, Y., Eldesouki, M., Attia, M., Darwish, K., Abdelali, A., Mubarak, H., and Kallmeyer, L. (2017b). Learning from relatives: Unified dialectal arabic segmentation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 432–441.

Stratos, K. and Collins, M. (2015). Simple semi-supervised pos tagging. In *VS@ HLT-NAACL*, pages 79–87.

Zaidan, O. F. and Callison-Burch, C. (2011). The arabic online commentary dataset: an annotated dataset of informal arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41. Association for Computational Linguistics.

A Corpus for Modeling Word Importance in Spoken Dialogue Transcripts

Sushant Kafle, Matt Huenerfauth
Rochester Institute of Technology
1 Lomb Memorial Drive, Rochester, NY
sushant@mail.rit.edu, matt.huenerfauth@rit.edu

Abstract

Motivated by a project to create a system for people who are deaf or hard-of-hearing that would use automatic speech recognition (ASR) to produce real-time text captions of spoken English during in-person meetings with hearing individuals, we have augmented a transcript of the Switchboard conversational dialogue corpus with an overlay of word-importance annotations, with a numeric score for each word, to indicate its importance to the meaning of each dialogue turn. Further, we demonstrate the utility of this corpus by training an automatic word importance labeling model; our best performing model has an F-score of 0.60 in an ordinal 6-class word-importance classification task with an agreement (concordance correlation coefficient) of 0.839 with the human annotators (agreement score between annotators is 0.89). Finally, we discuss our intended future applications of this resource, particularly for the task of evaluating ASR performance, i.e. creating metrics that predict ASR-output caption text usability for DHH users better than Word Error Rate (WER).

Keywords: Word Importance Annotation, Speech Recognition Evaluation, Word Importance Prediction

1. Introduction

There has been increasing interest among researchers of speech and language technology applications to identify the importance of individual words, for the overall meaning of the text. Depending on the context of how the importance of a word is defined, this task has found use in varieties of applications such as text summarization (Hong and Nenkova, 2014; Yih et al., 2007), text classification (Sheikh et al., 2016), or speech synthesis (Mishra et al., 2007).

Our laboratory is currently designing a system to benefit people who are deaf or hard-of-hearing (DHH) who are engaged in a live meeting with hearing colleagues. In many settings, sign language interpreting or professional captioning (where a human types the speech, displayed as text on a screen for the user), are unavailable, e.g. in impromptu conversations in the workplace. A system that uses automatic speech recognition (ASR) to generate captions in real-time could display this text on mobile devices for DHH users, but text output from ASR systems inevitably contains errors. Thus, we were motivated to understand which words in the text were most important to the overall meaning, to inform our evaluation of ASR accuracy for this task.

In this paper, we present a word-importance annotation of transcripts of the Switchboard corpus (Godfrey et al., 1992). While our overall goal is to produce measures of ASR accuracy for our caption application; to demonstrate the use of this corpus, in this paper, we present models that predict word-importance in spoken dialogue transcripts.

1.1. ASR Evaluation

ASR researchers generally report the performance of their systems using a metric called Word Error Rate (WER). The metric considers the number of errors in the output of the ASR system, normalized by the number of words human actually said in the audio recording. While WER has been the most commonly used intrinsic measure for the evaluation of ASR, there have been criticisms of WER (McCowan et al., 2004; Morris et al., 2004), and several researchers have recommended alternative measures to better predict human task-performance in applications that depend on ASR (Garofolo et al., 2000; Mishra et al., 2011; Kafle and Huenerfauth, 2016).

Among these newly proposed metrics, a common theme has been: rather than simply counting the number of errors, it would be better to consider the importance of the individual words that are incorrect - suggesting that it would be better to more heavily penalize systems that make errors on words that are important (with the definition of importance based on the specific application or task). This approach of penalizing errors differentially has been shown to be useful in various application settings, e.g. in our research for DHH users, we have found that an evaluation metric designed for predicting the usability of an ASR-generated transcription as a caption text for these users could benefit from word importance information (Kafle and Huenerfauth, 2017). However, estimating the importance of a word has been challenging for our team thus far, because we have lacked corpora of conversational dialogue with word-importance annotation, for training a word-importance model.

1.2. Word Importance Estimation

Prior research on identifying and scoring important words in a text has largely focused on the task of keyword extraction, which involves identifying a set of descriptive words in a document that serves as a dense summary of the document. Several automatic keyword extraction techniques have been investigated over the years, including unsupervised methods using, e.g. Term Frequency x Inverse Document Frequency (TF-IDF) weighting (HaCohen-Kerner et al., 2005), word co-occurrence probability estimation (Matsuo and Ishizuka, 2004) – as well as supervised methods that leverage various linguistic features from text to achieve strong predictive performance (Liu et al., 2011; Liu et al., 2004; Hulth, 2003; Sheeba and Vivekanandan, 2012).

While this conceptualization of word importance as a keyword-extraction problem has led to positive results in

the field of text summarization (Litvak and Last, 2008; Wan et al., 2007; Hong and Nenkova, 2014), this approach may not generalize to other applications. For instance, given the sometimes meandering nature of topic transition in spontaneous speech dialogue (Sheeba and Vivekanandan, 2012), applications that process transcripts of such dialogue may benefit from a model of word importance that is more local, i.e. based on the importance of a word at sentential, utterance, or local dialogue level, rather than at a document-level. Furthermore, the dyadic nature of dialogue, with interleaved contributions from multiple speakers, may require special consideration when evaluating word importance. In this paper, we present a corpus with annotation of word importance that could be used to support research into these complex issues.

2. Defining Word Importance

In eye-tracking studies of reading behavior, researchers have found that readers rarely glance at every word in a text sequentially: Instead, they sometimes regress (glance back at previous words), re-fixate on a word, or skip words entirely (Rayner, 1998). This research supports the premise that some words are of higher importance than others, for readers. Analyses of eye-tracking recordings have revealed a relationship between these eye-movement behaviors and various linguistic features, e.g. word length or word predictability. In general, readers' gaze often skips over words that are shorter or more predictable (Rayner et al., 2011).

While eye-tracking suggests some features that may relate to readers' judgments of word importance, at least as expressed through their choice of eye fixations, we needed to develop a specific definition of word importance in order to develop annotation guidelines for our study. Rather than ask annotators to consider specific features, e.g. word length, which may pre-suppose a particular model, we instead took a functional perspective, with our application domain in mind. That is, we define word importance for spontaneous spoken conversation as the degree to which a reader of a transcript of the dialogue would be unable to understand the overall meaning of a conversational utterance (a single turn of dialogue) if that word had been "dropped" or omitted from the transcript. This definition underlies our annotation scheme (in 3 1) and suits our target application, i.e. evaluating ASR for real-time captioning of meetings.

In addition, for our annotation project, we defined word-importance as a single-dimensional property, which could be expressed on a continuous scale from 0.0 (not important at all to the meaning of the utterance) to 1.0 (very important). Figure 1 illustrates how numerical importance scores can be assigned to words in a sentence – in fact, this figure displays actual scores assigned by a human annotator working on our project. Of course, asking human annotators to assign specific numerical scores to quantify the importance of a word is not straightforward. In later sections, we discuss how we attempt to overcome the subjective nature of this task, to promote consistency between annotators, as we developed this annotated resource (see Section 3 1). Section 4 characterizes the level of agreement between our annotators on this task.

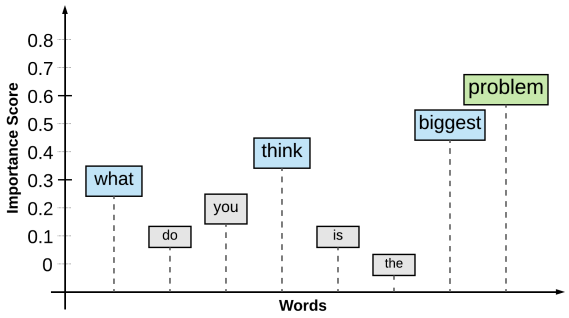


Figure 1: Visualization of importance scores assigned to words in a sentence by a human annotator on our project, with the height and font-size of words indicating their importance score (and redundant color coding: green for high-importance words with score above 0.6, blue for words with score between 0.3 and 0.6, and gray otherwise).

3. Corpus Annotation

The Switchboard corpus consists of audio recordings of approximately 260 hours of speech consisting of about 2,400 two-sided telephone conversations among 543 speakers (302 male, 241 female) from across the United States (Godfrey et al., 1992). In January 2003, the Institute for Signal and Information Processing (ISIP) released written transcripts for the entire corpus, which consists of nearly 400,000 conversational turns. The ISIP transcripts include a complete lexicon list and automatic word alignment timing corresponding to the original audio files[1].

In our project, a pair of annotators have assigned word-importance scores to these transcripts. As of September 2017, they have annotated over 25,000 tokens, with the overlap of approximately 3,100 tokens. With this paper, we announce the release[2] of these annotations as a set of supplementary files, aligned to the ISIP transcripts. Our annotation work continues, and we aim to annotate all of the Switchboard corpus and with a larger group of annotators.

3.1. Annotation Scheme

To reduce the cognitive load on annotators and to promote consistency, we created the following annotation scheme:

Range and Constraints. Each word is assigned a numeric score between [0, 1], where 1 indicates a high importance score; the numeric score has the precision of 0.05. Importance scores are not meant to indicate an absolute proportion of the utterance's meaning represented by each word, i.e. the scores do not have to sum to 1.

Methodology. Given an utterance (a speaker's single turn in the conversation), the annotator first considers the overall meaning conveyed by the utterance, with the help of the previous conversation history (if available). The annotator then scores each word based on its (direct or indirect) contribution to the utterance's meaning, using the rubric described in the Interpretation and Scoring section below.

[1] https://www.isip.piconepress.com/projects/switchboard/
[2] http://latlab.ist.rit.edu/lrec2018

Range	Description
[0 - 0.3)	Words that are of least importance - these words can be easily omitted from the text without much consequence.
[0.3 - 0.6)	Words that are fairly important - omitting these words will take away some important details from the utterance.
[0.6 - 1]	Words that are of high importance - omitting these words will change the message of the utterance quite significantly.

Table 1: Guidance for the annotators to promote consistency and uniformity in the use of numerical scores.

Rating Scheme. To help annotators calibrate their scores, Table 1 provides some recommendations for how to select word-importance scores in various numerical ranges.

Interpretation and Scoring. Annotators should consider how their understanding of the utterance would be affected if this word had been "dropped," i.e. replaced with a blank space ("_____"). Since these are conversations between pairs of speakers, annotators should consider how much the other person in the conversation would have difficulty understanding the speaker's message if that word had been omitted, i.e. if they had not heard that word intelligibly.

4. Inter-annotator Agreement

There were 3,100 tokens in our "overlap" set, i.e. the subset of transcripts independently labeled by both annotators. This set was used as the basis for calculating inter-annotator agreement. Since scores were nearly continuous (ranges [0,1] with a precision of 0.05), we computed the Concordance Correlation Coefficient (ρ_c), also known as Lin's concordance correlation coefficient, as our primary metric for measuring the agreement between the annotators. This metric indicates how well a new test or measurement (X) reproduces a gold standard or measure (Y). Considering the annotations from one annotator as a gold standard, we can generalize this measure to compute the agreement between two annotators. Like other correlation coefficients, ρ_c also ranges from -1 to 1; 1 being the score of perfect agreement.

Concordance between the two measures can be characterized by the expected value of their squared difference as:

$$E[(Y - X)^2] = (\mu_y - \mu_x)^2 + \sigma_x^2 + \sigma_y^2 - 2\rho\sigma_x\sigma_y \quad (1)$$

where, ρ is the correlation coefficient, μ_x and μ_y are the means of the population of the variables X and Y, and σ_x and σ_y are their standard deviation. The expectation score coefficient (between -1 and 1) is calculated as follows:

$$\rho_c = \frac{2\rho S_x S_y}{(\bar{Y} - \bar{X})^2 + S_x^2 + S_y^2} \quad (2)$$

where, ρ_c is the correlation coefficient, $\bar{X}$ and $\bar{Y}$ are the mean of X and Y, and S_x and S_y are standard deviations.

We obtained an agreement score (ρ_c) of 0.89 between our annotators, which we interpret as an acceptable level of agreement, given the subjective nature of the task of quantifying word importance in spoken dialogue transcripts.

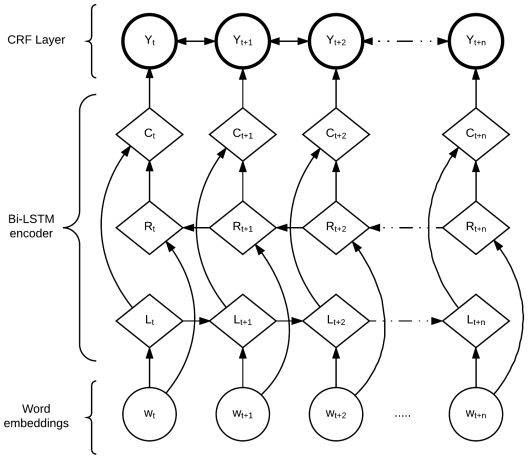


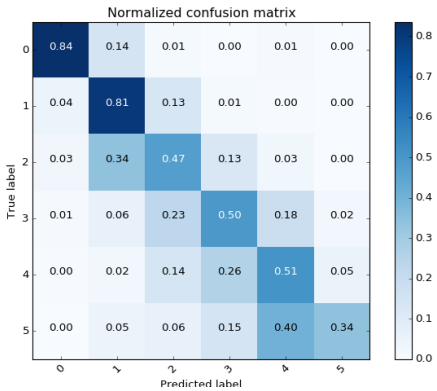
Figure 2: =
[General unfolded network structure of our model, adapted from (Lample et al., 2016). The bottom layer represents word-embedding inputs, passed to bi-directional LSTM layers above. Each LSTM takes as input the hidden state from the previous time step and word embeddings from the current step, and outputs a new hidden state. C_i concatenates hidden representations from LSTMs (L_i and R_i) to represent the word at time i in its context.]

5. Automatic Prediction

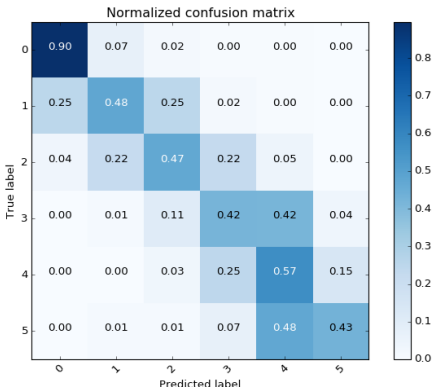
To demonstrate the use of this corpus, we trained a prediction model, by adopting the neural architecture described in (Lample et al., 2016) consisting of bidirectional LSTM encoders with a sequential Conditional Random Field (CRF) layer on top. Our input word tokens were first mapped to a sequence of pre-trained distributed embeddings (Pennington et al., 2014) and then combined with the learned character-based word representations to get the final word representation. As shown in Figure 2, the bidirectional LSTM encoders are used to create a context-aware representation of each word. The hidden representations from each LSTM were concatenated to obtain a final representation, conditioned on the whole sentence. The CRF layer uses this representation to look for the most optimal state (Y) sequence through all the possible state configurations.

The neural framework was implemented using Tensorflow, and the code is publicly available[3]. The word embeddings were initialized with publicly available pre-trained glove vectors (Pennington et al., 2014). The embeddings for characters were set to length 100 and were initialized randomly. The LSTM layer size was set to 300 in each direction for word- and 100 for character-level components. Parameters were optimized using the Adam (Kingma and Ba, 2014) optimizer, with the learning rate initialized at 0.001 with a decay rate of 0.9, and sentences were grouped into batches

[3]https://github.com/SushantKafle/speechtext-wimp-labeler



(a) Normalized confusion matrix for LSTM-CRF



(b) Normalized confusion matrix for LSTM-SIG

Figure 3: Confusion matrices for each model for classification into 6 classes: c_1 = [0, 0.1), c_2 = [0.1, 0.3), and so forth.

of size 20. We applied a dropout with a probability of 0.5 during training on word embeddings.

We investigated two variations of this model: (i) a bidirectional LSTM model with sequential CRF layer on top (LSTM-CRF) treating the problem as a discrete classification task, (ii) a new bidirectional LSTM model with a sigmoid layer on top (LSTM-SIG) for a continuous prediction. The LSTM-CRF models the prediction task as a classification problem, using a fixed number of non-ordinal class labels. In contrast, the LSTM-SIG model provides a continuous prediction, using a sigmoid nonlinearity to bound the prediction scores between 0 and 1. Using a square loss, we train this model to directly learn to predict the annotation scores, similar to a regression task.

5.1. Evaluation and Discussion

Partitioning our corpus as 80% training, 10% development, and 10% test sets, we evaluated our model using two measures: (i) total root mean square error (RMS) - the deviation of the model predictions from the human-annotations and, (ii) F_1 measure in a classification task - the ability of the model to predict human-annotations categorized into a group of classes. To evaluate performance in terms of classification, we discretized annotation scores into 6 classes: [0, 0.1), [0.1, 0.3), [0.3, 0.5), [0.5, 0.7), [0.7, 0.9), [0.9, 1].

Table 2 summarizes the performance of our models on the test set, presenting average scores for 5 different configurations, to compensate for outlier results due to randomness in model initialization. While the LSTM-CRF had a better (higher) F-score on the classification task, its RMS score was worse (higher) than the LSTM-SIG model, which may be due to the limitation of the model as discussed in Section 5.

Model	**RMS**	F_1 **(macro)**
LSTM-CRF	0.154	**0.60**
LSTM-SIG	**0.120**	0.519

Table 2: Model performance in terms of RMS deviation and macro-averaged F_1 score, with best results in **bold** font.

Confusion matrices in Figure 3 provide a more detailed view of the classification performance of each model. Since the LSTM-SIG was trained to optimize the accuracy of its continuous predictions, rather than its discrete assignment of instances to classes, it is not surprising to see a "wider diagonal" in the confusion matrix in Figure 3(b), which indicates that the LSTM-SIG model was more likely to misclassify words using ordinally adjacent classes. The figure illustrates that both models were worse at classifying words with importance scores in the middle range [0.3, 0.7).

Treating our human-annotations as ground truth, we also computed the concordance correlation coefficient to measure the agreement between the human annotation and each model. The average correlation between the human annotator and the LSTM-CRF model was higher ($\rho_c = 0.839$), as compared to the LSTM-SIG model ($\rho_c = 0.826$).

6. Conclusions and Future Work

We have presented a new collection of annotation of transcripts of the Switchboard conversational speech corpus, produced through human annotation of the importance of individual words to the meaning of each utterance. We have demonstrated the use of this data by training word-importance prediction models, with the best model achieving an F_1 score of 0.60 and model-human agreement correlation of 0.839. In future work, we will collect additional human annotations for additional sections of the corpus. This research is part of a project on the use of ASR to provide real-time captions of speech for DHH individuals during meetings, and we plan to incorporate these word-importance models into new word-importance-weighted metrics of ASR accuracy, to better predict the usability of ASR-produced captions for these users.

7. Acknowledgement

This material was based on work supported by the National Technical Institute for the Deaf (NTID). We thank Tomomi Takeuchi and Michael Berezny for their contributions.

8. Bibliographical References

Garofolo, J. S., Auzanne, C. G., and Voorhees, E. M. (2000). The trec spoken document retrieval track: A success story. In *Content-Based Multimedia Information Access-Volume 1*, pages 1–20. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

Godfrey, John J and Holliman, Edward C and McDaniel, Jane. (1992). *SWITCHBOARD: Telephone speech corpus for research and development*.

HaCohen-Kerner, Y., Gross, Z., and Masa, A. (2005). Automatic extraction and learning of keyphrases from scientific articles. *Computational linguistics and intelligent text processing*, pages 657–669.

Hong, K. and Nenkova, A. (2014). Improving the estimation of word importance for news multi-document summarization. In *EACL*, pages 712–721.

Hulth, A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223. Association for Computational Linguistics.

Kafle, S. and Huenerfauth, M. (2016). Effect of speech recognition errors in text understandability for people who are deaf or hard-of-hearing. In *Proceedings of the 7th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT)*. Interspeech.

Kafle, S. and Huenerfauth, M. (2017). Evaluating the usability of automatically generated captions for people who are deaf or hard of hearing. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility*. ACM.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 260–270.

Litvak, M. and Last, M. (2008). Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, pages 17–24, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liu, B., Li, X., Lee, W. S., and Yu, P. S. (2004). Text classification by labeling words. In *AAAI*, volume 4, pages 425–430.

Liu, F., Liu, F., and Liu, Y. (2011). A supervised framework for keyword extraction from meeting transcripts. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(3):538–548, March.

Matsuo, Y. and Ishizuka, M. (2004). Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 13(01):157–169.

McCowan, I. A., Moore, D., Dines, J., Gatica-Perez, D., Flynn, M., Wellner, P., and Bourlard, H. (2004). On the use of information retrieval measures for speech recognition evaluation. Technical report, IDIAP.

Mishra, T., Prud'hommeaux, E. T., and van Santen, J. P. (2007). Word accentuation prediction using a neural net classifier. In *SSW*, pages 246–251.

Mishra, T., Ljolje, A., and Gilbert, M. (2011). Predicting human perceived accuracy of asr systems. In *INTERSPEECH*, pages 1945–1948.

Morris, A. C., Maier, V., and Green, P. (2004). From wer and ril to mer and wil: improved evaluation measures for connected speech recognition. In *Eighth International Conference on Spoken Language Processing*.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rayner, K., Slattery, T. J., Drieghe, D., and Liversedge, S. P. (2011). Eye movements and word skipping during reading: effects of word length and predictability. *Journal of Experimental Psychology: Human Perception and Performance*, 37(2):514.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372.

Sheeba, J. and Vivekanandan, K. (2012). Improved keyword and keyphrase extraction from meeting transcripts. *International Journal of Computer Applications*, 52(13).

Sheikh, I., Illina, I., Fohr, D., and Linares, G. (2016). Learning word importance with the neural bag-of-words model. In *ACL, Representation Learning for NLP (Repl4NLP) workshop*.

Wan, X., Yang, J., and Xiao, J. (2007). Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In *ACL*, volume 7, pages 552–559.

Yih, W.-t., Goodman, J., Vanderwende, L., and Suzuki, H. (2007). Multi-document summarization by maximizing informative content-words. In *IJCAI*, volume 7, pages 1776–1782.

9. Language Resource References

Godfrey, John J and Holliman, Edward C and McDaniel, Jane. (1992). *SWITCHBOARD: Telephone speech corpus for research and development*.

Dialogue Structure Annotation for Multi-Floor Interaction

David Traum[1], Cassidy Henry[2], Stephanie Lukin[2], Ron Artstein[1]
Felix Gervitz[3], Kimberly A. Pollard[2], Claire Bonial[2], Su Lei[1],
Clare R. Voss[2], Matthew Marge[2], Cory J. Hayes[2], Susan G. Hill[2]
[1]USC Institute for Creative Technologies, [2]U.S. Army Research Laboratory, [3]Tufts University
Playa Vista, CA 90094, Adelphi, MD 20783, Medford, MA 02155
traum@ict.usc.edu

Abstract

We present an annotation scheme for meso-level dialogue structure, specifically designed for multi-floor dialogue. The scheme includes a *transaction unit* that clusters utterances from multiple participants and floors into units according to realization of an initiator's intent, and *relations* between individual utterances within the unit. We apply this scheme to annotate a corpus of multi-floor human-robot interaction dialogues. We examine the patterns of structure observed in these dialogues and present inter-annotator statistics and relative frequencies of types of relations and transaction units. Finally, some example applications of these annotations are introduced.

Keywords: dialogue structure annotation, human-robot interaction, multiparty dialogue

1. Introduction

We present an annotation scheme for meso-level dialogue structure (Traum and Nakatani, 1999), specifically designed for multi-floor dialogue. The scheme includes both a *transaction unit* for clustering utterances from multiple participants and floors that contribute to realization of an initiating participant's intent, and *relations* between individual utterances within the unit. While there are standard annotation schemes for both dialogue acts (Bunt et al., 2012) and discourse relations (Prasad and Bunt, 2015), these schemes do not fully address the issues of dialogue structure. Of particular interest to us, and not previously addressed in other schemes, are cases in which the units and relations span across multiple conversational floors. Dialogues can be characterized by distinct information states (Traum and Larsson, 2003). These include sets of participants, participant roles (e.g. active, ratified participant vs. overhearer), turn-taking or floor-holding, expectation of how many participants will make substantial contributions at a time (Edelsky, 1981), and other factors. Often distinct dialogues with different information states are going on at the same time. There are a number of ways in which such dialogues can be related to each other, including:

- having the same purpose but distinct participants, e.g., teams competing in a trivia contest to come up with the answer first.
- co-located such that participants in one can observe and possibly comment on the other, such as groups of people sitting at different tables at a restaurant.
- having one or more (but not all) participants in common, where such participants are *multicommunicating* (Reinsch et al., 2008), e.g., someone in a meeting is texting with one or more people outside the meeting.

In the multicommunicating case, the multiple dialogues that a multicommunicator is part of might involve completely separate topics or be more closely related, such that satisfaction of the goals of one depends on actions in the other. For example, a question arising in a meeting might be conveyed and answered over the text channel. We use the term *multi-floor dialogue* to refer to cases in which the high-level dialogue purposes are the same, and some content is shared, but other aspects of the information state, such as the participant structure and turn-taking expectations, are distinct. Situations of distributed decision-making and action are quite common, e.g., in a restaurant where some people take the customer's order and others make the food, or in military units, where orders are relayed through the chain of command. In some cases, where all parties can hear all communication, we can view this as multi-party dialogue within a single floor, but in other cases not all the communications are available to all participants – this is a case of multi-floor dialogue. We are particularly interested in capturing the latter case.

In the next section, we present the annotation scheme. This is applied to a corpus of human-robot dialogue (section 3), and the scheme is shown to have high inter-annotator reliability (section 4). Our objectives for the annotation are two-fold. First, we seek to explore how multi-floor dialogue works and characterize different kinds of multi-party, multi-floor contributions. This is addressed by analysis of dialogue annotated with this scheme and the kinds of patterns of interaction that are observed (see section 5). Second, we use data from the corpus annotated with this scheme to serve as training and evaluation data for creating automated multicommunicators (see section 6).

2. Annotation Scheme

We annotate two aspects of Dialogue Structure at the *meso-level* (bigger than a single speaker-turn, but smaller than a complete dialogue activity) (Traum and Nakatani, 1999). First, we look at *intentional structure* (Grosz and Sidner, 1986), consisting of units of dialogue utterances that all have a role in explicating and addressing an initiating participant's intention. Second, we look at the relations between different utterances within this unit, which reveal

Expansions	relate utterances that are produced by the same participant within the same floor.
Responses	relate utterances by different participants within the same floor.
Translations	relate utterances in different floors.

Table 1: Top Level Corpus Relations

how the information state of participants in the dialogue is updated as the unit is constructed. To fully understand the intentional and interactional structure, it is also necessary to include aspects of micro-level meaning (e.g. dialogue acts) and macro-level meaning (e.g. dialogue purpose); however those aspects are beyond the scope of the current paper.

We call the main unit of intentional structure a *transaction unit*, following Sinclair and Coulthard (1975) and Carletta et al. (1996). A transaction unit (TU) contains an initial message by one speaker and all subsequent messages by the same and other speakers across all floors to complete the intention. For example, a transaction may consist of an instruction initiated by one participant in one floor that is relayed by a multicommunicator to another floor, and then performed by another participant of the second floor, in addition to various sorts of feedback between pairs of participants. For a TU we focus on the lowest level of dialogue in which intentions are fulfilled across speakers. In some kinds of dialogues, particularly complex negotiations or problem-solving, intentional structure can be recursive, such that the purpose of one segment partially contributes to the purpose of a higher-level segment (Grosz and Sidner, 1986). Other types of dialogues have a flatter structure, including transactions that contribute to an overall dialogue purpose, but with few, if any, levels in between. Each utterance-level message is assigned to at most one TU, and the TU is defined by the set of constituent utterances. At most points in a dialogue, there is only one active TU, however there are occasions where there are multiple active TUs, when a new one is started before the previous one has been completed.

We also model the internal structure of TUs as *relations* between pairs of utterances within the unit. Each relation is annotated by coding a relation-type and an antecedent for each utterance after the first in a transaction. Thus, each transaction can be viewed as a tree structure, with the first utterance as root (having no relation-type or antecedent annotations). While relations often exist between an utterance and multiple previous utterances, to simplify the annotation, we code only the most direct, recent such relation. This practice is common for many annotation efforts, e.g. the "code-high" principle from (Condon and Cech, 1992). In the future, we plan to use inference rules to derive some "indirect relations" from what has been annotated.

We have developed a taxonomy of relation types based on how a new utterance is connected to its antecedent. At the highest level, we distinguish relations by the combination of the participants who produced the utterances (often called "speakers", even if the communication was not using speech) and floors that the utterances are part of (Table 1).

Each of these types has one or more subtypes. For expansions, we indicate how the intention is expressed across multiple utterances. Utterances that add additional content are termed *continues*, and utterances that remove or replace some content are called *corrections*. Utterances that do neither, but reiterate some content, are termed *summarizations*. Finally, utterances that consist primarily of explicit discourse markers that link a preceding utterance to a following one are termed *link-next*.

For translations, there is one subtype for each source and target floor combination. Thus, a dialogue with two floors would have two translate relations, while one with three floors would have up to six. We also include two other types: *quotation* and *comment*, where some content is conveyed across floors but not the same illocutionary force as the original.

We also annotate several types of responses, many of which in turn have sub-types. The main types of responses are summarized in Table 2. These cover positive and negative feedback on contact, perception, understanding, and attitudinal reaction (Allwood et al., 1992), as well as perfomance status and other relevance relations. The *acknowledgement*, *clarification*, and *question-response* relation types have multiple sub-types, as shown in Table 3. Acknowledgements all indicate a claim or demonstration of understanding of the antecedent. However, the subtypes also indicate the status of an instructed action: whether it has been started or completed, or whether the responder thinks it can or will be done. Clarifications all indicate a lack of ability to fully understand and act on an instruction, with subtypes indicating problems with receiving a message, the message being incomplete, requesting clarification, or providing a repair elicited by another participant. Question responses can be either answers or non-answers that address a question but do not directly provide an answer.

processing:	positive feedback (Allwood et al., 1992) at the perception level, but lack of feedback at higher levels.
acknowledgement:	positive feedback at the understanding level, indicating *grounding* (Clark and Schaefer, 1989; Traum, 1994), with subcategories indicating attitudinal reaction, commitment and performance status of an instructed action.
clarification:	negative feedback of understanding, with subcategories representing different strategies for repair.
question-response:	the antecedent is a question and the response indicates understanding and some attempt to address the question,
reciprocal response:	response indicates the same or similar content as the antecedent e.g. reciprocal greetings.
3rd turn feedback:	a response to a response - often an evaluation of the response.
other:	a response not fitting the other categories.

Table 2: Response Relation Types

acknowledgment	(listed in order of likelihood of action being successfully performed)
ack-done	ack that a command or prior planned act has been completed successfully
ack-doing	ack that the speaker understands the command and is starting to do it
ack-wilco	ack of a command and promise to do it in the future (includes acceptance with something like "ok")
ack-understand	express or show understanding without commitment to action or agreement. Includes repetitions of what was said, affirmative cue words like "uh-huh".
ack-try	ack of a command and promise to try to do it (but not necessarily)
ack-unsure	ack of understanding of a command, expressing uncertainty about whether it can/will be done. Not clearly an ack-cant or ack-try, but also more than ack-understand because of some explicit statement of doubt about possibility or future action.
ack-cant	expression that the previous command was understood but can't be executed.
clarification	
req-clar	request for clarification – indicates that something in the prior utterance was not clear, and asks the other speaker to do something about it (such as answer a question or confirm a trial). E.g. a command was not specified well enough to be unambiguously confirmed and carried out.
clar-repair	providing a clarification to a prior utterance , after prompting by another (other-initiated self-repair).
missing info	indicates a specific part of the antecedent was not interpretable well enough to act on, but not requesting further action (e.g "I don't know which object you are referring to"). The other party has the option of whether to clarify-repair or move on and do something else.
nack	indicates that the antecedent could not be understood well enough to act on, but not explicitly requesting action (e.g. "no copy" or "I don't understand")
req-repeat	request to repeat a prior utterance
clar-repeat	providing (other-initiated self) repeat, after prompting to repeat with a req-repeat
question-response	
answer	an answer to a question, other than a clar-repair or clar-repeat.
Non-Answer-Response (NAR)	addresses question without providing an answer. E.g. explains why an answer won't be given, or the question is not relevant, or a helpful suggestion of how the requested information might be arrived at.

Table 3: Response Sub-Relations

3. Initial Domain Application: Distributed Human-Robot Interaction

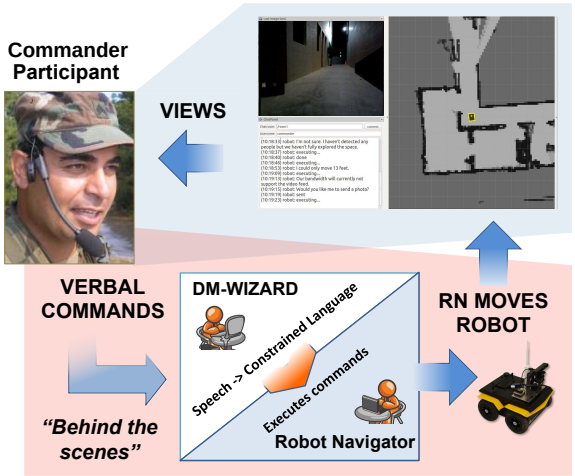


Figure 1: Domain Application Experimental Setup: Human-Robot Interaction with Wizards

We first apply this annotation scheme to a corpus of human-robot interaction, taken from a project with a long-term goal to create an autonomous robot intelligence that can collaborate with remotely located human participants on exploration and navigation tasks. In the initial versions, a human "Commander" tasks the robot verbally, and gets feedback via multiple modalities, including text messages, a live 2D-map built from the robot's LIDAR scanner, and still photos captured from the robot's front-facing camera. In order to collect sufficient information about the type of language used by a Commander (Marge et al., 2017), and provide training data to support development of appropriate language processing components, the development of the autonomous human-robot interaction begins with a series of "Wizard of Oz" experiments (Marge et al., 2016; Bonial et al., 2017), where the robot is controlled by two wizards, with an internal communication floor, distinct from the floor used by the Commander to communicate with the robot. The wizards include a Dialogue Manager (DM-Wizard, or DM) who handles communication to the Commander and "speaks" via text messages (Bonial et al., 2017) and a Robot Navigator (RN-Wizard, or RN) who teleoperates the robot based upon commands issued by the Commander relayed by the DM (Figure 1).

This Wizard of Oz communicative setting thus involves multi-floor dialogue: three participants (Commander, DM-Wizard, and RN-Wizard), two floors (Commander to DM, called "left", and DM to RN, called "right") and four distinct message streams. The DM-Wizard is multicommunicating and can translate from one floor to the other. When the DM translates Commander speech to the "right" floor, this is called translation-r, and when they translate from the RN to the "left" floor, this is termed translate-l. The RN and the Commander cannot speak to one another directly.

4. Corpus and Annotation

A total of 60 dialogues (up to 20 min. each) were collected from 20 Commander participants. Aligned transcripts were produced for all dialogues in the form of Tables 4–7, by transcribing Commander and RN speech, and aligning with

	Left Floor		Right Floor		Annotations		
#	**Commander**	**DM→Commander**	**DM→RN**	**RN**	**TU**	**Ant**	**Rel**
1	move forward three feet				1		
2		ok			1	1	ack-wilco
3			move forward 3 feet		1	1	translation-r
4				done	1	3	ack-done
5		I moved forward 3 feet			1	4	translation-l

Table 4: Example Minimal TU. The ack- prefix indicates a type of acknowledgement.

	Left Floor		Right Floor		Annotations		
#	**Commander**	**DM→Commander**	**DM→RN**	**RN**	**TU**	**Ant**	**Rel**
1	face west				1		
2	and take a photo				1	1	continue
3			face west, photo		1	2*	translation-r
4		executing...			1	2*	ack-doing
5				image sent	1	3	ack-done
6		sent			1	5	translation-l

Table 5: Example Extended-Link TU. The ack- prefix indicates a type of acknowledgement. * indicates that the most direct antecedent is part of a sequence connected by expansions (e.g. utterances 1 and 2 are both being translated or responded to, but 2 is the most recent antecedent).

DM text messages. The right three columns include annotations of dialogue structure, with Transaction Unit (TU), Antecedent (Ant), and Relation-type (Rel) for each.
Because all participants in the dialogues are guiding the robot to accomplish a small set of search and exploration tasks, many of the TUs in this corpus involve instructions initiated by the Commander, which are translated into a simplified form by the DM and passed along to the RN-Wizard, who then carries out those instructions by teleoperating the robot. We call TUs that include only these components and acknowledgements (and translations of acknowledgements) "Minimal TUs" – an example of which is shown in Table 4, or "Extended-link" TUs, like Table 5, depending on whether they include a single instruction or a sequence of multiple instructions. However, there are also units involving questions and clarifications, and other types of dialogue moves, when dialogue ensues to repair any Commander instructions that are unclear (perhaps due to garbled speech), ambiguous (as to a referent in the physical environment), or impossible (given the constraints of the physical environment). Table 6 includes two example TUs involving questions and responses. Table 7 shows another example of two TUs, however in this case one involves a repair, and the second one commences before the first one has been completed.

4.1. Inter-annotator Reliability for this annotation scheme and corpus

Inter-annotator reliability was calculated separately on three different markables: antecedents, relation types, and transaction units. An initial sample of 3 dialogues (482 utterances) was annotated by up to 5 coders using an early version of the coding manual; this was followed by several iterations of refining the coding manual, after which a second sample (1 dialogue, 314 utterances) was annotated by 6 coders. Results of the agreement tests are in Table 8.

After each round of agreement testing, disagreements were reconciled, and the annotation guidelines were revised to clarify issues of disagreement. In the initial version, there were confusions about how many distinct instructions were part of the same transaction unit. The guidelines were revised to make clear that continuations of an instruction would be considered part of the same TU only until the DM started acting on them (with feedback and/or a translation). Subsequent Commander instructions would be seen as starting a new TU, even if they follow semantically from the previous instruction. For example, in Table 7, the instruction in line 6 to take another picture is seen as new TU, even though the previous TU has not been completed. Another added guideline allowed the antecedent to be designated as a sequence (via a *) rather than just a single utterance.

A major contributor to remaining disagreement in the second test was the case where compound instructions are both presented and translated in multiple utterances. In this case, the second translation was both a translation of the second instruction as well as a continuation of the first translation. The initial guidelines indicated that the most recent relation was to be annotated, but in this case it was felt that translations were more important to capture than continuations, and marking translations of individual utterances (where appropriate) was more informative than translations of whole sequences, so the guidelines were updated to consider the most important recent relation. An example is shown in Table 9. Here, the whole sequences of 6,7,10

	Left Floor		Right Floor		Annotations		
#	**Commander**	**DM→Commander**	**DM→RN**	**RN**	**TU**	**Ant**	**Rel**
1	how many window openings do you see in front of you				1		
2		three			1	1	answer
3	do you see a yellow flashlight				2		
4		processing...			2	3	processing
5		I'm not sure			2	3	answer
6		If you describe an object, you can help me to learn what it is.			2	3	non-answer response

Table 6: Example including two QA TUs

	Left Floor		Right Floor		Annotations		
#	**Commander**	**DM→Commander**	**DM→RN**	**RN**	**TU**	**Ant**	**Rel**
1	move to where you see the first cone				1		
2		I'm not sure which object you are referring to. Can you describe it in another way, using color or its location?			1	1	request-clarification
3	move to the cone on the right a red cone on the right				1	2	clarification-repair
4			move to face the cone on the right		1	3	translation-r
5		executing...			1	3	ack-doing
6	take another picture				2		
7				done	1	4	ack-done
8		done			1	7	translation-l
9			image		2	6	translation-r
10				image sent	2	9	ack-done
11		sent			2	10	translation-l

Table 7: Multifloor example: two partially interleaved TUs (Repair and minimal)

Markable Type	Agreement		Distance Metric
	Initial	Second	
Antecedents	0.72–0.82	0.78	Nominal[a]
Relation Types	0.77–0.82	0.89	Nominal[a]
Transaction Units	0.48–0.70	0.93	MASI[b]

[a]Krippendorff (1980) [b]Passonneau (2006)

Table 8: Inter-annotator agreement (Krippendorff's α)

and 5,8, 9 are acknowledgements and translate-r of the sequence of 1,3, respectively. So 10 and 9 are both continuations within their local sequence as well as direct translation and acknowledgment of 3.

We expect that agreement would improve following the final guidelines, but as it was already fairly high, we did not do a final test.

5. Analysis of Corpus Annotation

Table 10 shows the distributions of relation types and major subtypes in the annotated corpus. The high percentage of translations and relatively low percentage of expansions indicate a high degree of multi-communicating and relatively low complexity in intra-turn discourse structure, respectively.

We also examined the tree-structures of TUs, which reveals that 644 unique TU patterns make up all the 2230 observed TUs in the collected corpora. These patterns were classified into the following taxonomy of TU types: Minimal, Extended-Link, Repair, Question-Answer, and Other.

Minimal TUs consist of a single instruction from the Commander that is well formed and that the DM passes on in a single instruction to the RN. Minimal transactions include a single translation-r, an acknowledgment back to the Commander, a successive acknowledgment from the RN, and finally a translation-l back to the Commander. Minimal TUs

	Left Floor		Right Floor		Annotations		
#	**Commander**	**DM→Commander**	**DM→RN**	**RN**	**TU**	**Ant**	**Rel**
1	go one hundred eighty degrees				1		
2		done,sent			0	0	translation-l
3	and take a picture				1	1	continue
4		ok			1	3*	ack-understand
5			turn 180		1	1	translation-r
6		I will turn around 180 degrees			1	1	ack-wilco
7		and. . .			1	6	link-next
8			then. . .		1	5	link-next
9			send image		1	3	translate-r
10		I will send a picture			1	3	ack-wilco
11		turning. . .			1	3*	ack-doing
12				uh done and sent	1	9*	ack-done
13		done, sent			1	12	translation-l

Table 9: Example TU containing complex instructions conveyed across floors in parts. Note that line 2 is from the previous TU, but its antecedents are not shown here so it is labelled 0.

Type	**Subtypes**	**#**	**%**
Translation		4287	39
	translation-r	2355	21
	translation-l	1911	17
	comment	21	< 1
	quotation	0	0
Expansion		1583	14
	continue	1175	11
	link-next	337	3
	correction	50	< 1
	summarization	20	< 1
Response		5193	47
	acknowledgment	3998	36
	done	2015	18
	doing	1357	12
	wilco	592	5
	understand	34	< 1
	try	15	< 1
	unsure	14	< 1
	can't	11	< 1
	clarification	569	5
	req-clar	266	2
	clar-repair	237	2
	missing info	36	< 1
	nack	20	< 1
	repeat	8	< 1
	processing	315	3
	question-response	212	2
	answer	84	1
	non-answer	11	< 1
	other	48	< 1
	3rd turn feedback	37	< 1
	reciprocal response	14	< 1

Table 10: Corpus Relation frequency

make up 48% of all TUs. An example of a minimal TU is shown in Table 4. Another example is the second TU in Table 7.

Extended-Link TUs consist of more than one well-formed instruction from the Commander that the DM passes on in one or more instructions to the RN. Common examples of this include cases where a Commander asks the robot to move to a particular landmark and take a picture of it. An example is shown in Table 5. Extended-Link patterns make up 26% of all TUs.

Repair TUs contain an instruction that requires a clarification; the instruction is not actionable (e.g., it is not well-formed, or missing information) and an exchange must occur to rectify it. 9% of TUs are a repair and successfully resolve the conflict (TU 1 in Table 7 is an example of a successful repair TU). 2% of TUs involving a repair were not resolved, and might have been abandoned or a new and unrelated instruction was issued.

Question-Answer TUs contain a question and an answer or other response that sometimes involves conference between each floor, but does not involve repairing an instruction. Question-Answer TUs are different from the Repair TU; Repair TUs and Question-Answer TUs may contain a question, answer, reciprocal response or 3rd turn feedback, but a Question-Answer TU does not involve malformed instructions, thus requiring no clarification. Table 6 shows two examples of a QA TUs. In 1% of TUs, after a question is answered, the answer, as an instruction, is passed to the RN to complete the action, for example, the DM asks if the Commander would like a picture, and they respond "yes". In 4% of TUs, a question is answered but not passed as an executable instruction to the RN, as in the instance of a Commander asking about the robot's capabilities.

Other TUs that do not fall into these categories include instructions that were abandoned or interrupted and do not contain any questions or repairs (11% of TUs).

Classification and examination of TUs show consistency throughout the task for relaying well-formed instructions, with 74% of TUs being Minimal or Extended-Link, indicative of instructions that can be translated to the RN and then executed. It is encouraging to observe that the majority of malformed instructions in Repair TUs are successfully resolved and a command is executed by the RN (82% of TUs classified as Repair). On the other hand, the majority of questions in QA TUs do not involve an executable instruction (82%), suggesting that after new information is received from another speaker, the Commander comes to a decision about what further instructions should be issued or should be abandoned and re-issued as new instructions.

6. Applications of Annotated Data

The dialogue structure annotations have been used for two purposes so far: characterizing types of instructions, and automating the dialogue manager. In (Marge et al., 2017), TUs, were used to help characterize the content of instructions people would formulate to a robot before it had a chance to respond. The initial TU instruction and all expansions before any feedback were called *instruction units*, and used to contrast use of absolute vs. relative coordinates. This formulation was used to gain insight into which aspects of these initial instructions changed during the course of participants interaction with the robot.

The approach toward automating the DM involves using the annotated corpus data to train a statistical text classifier in the NPC Editor platform (Leuski and Traum, 2011). To process the data for input to the classifier, we extracted all utterances from the annotated corpus that were produced by the DM. These included the Translation relations (translation-l, translation-r) and the various Response relations (Acknowledgements, Clarifications, and Question-Responses). This gave us a large training set which contained the input-response pairings that were processed by the DM in our experiments. An example training pair for the translation-r relation is "move to the cone on the right a red cone on the right" $\rightarrow$ "move to face the cone on the right" (see Table 7). In this way, the commander's actionable instruction was translated to the RN to carry out. In cases where the instruction was not actionable (e.g., "Move forward") the mapped response was often a clarification request directed to the commander (e.g., "How far would you like me to move forward?"). After training on these pairings, the classifier learned to translate commands and provide appropriate feedback to many of the input utterances.

While a more thorough evaluation of the system is work in progress, reasonably high accuracy on the most common commands was observed: we were able to achieve over 80% accuracy on first dialogue manager response on a held-out corpus of 6 dialogues from our corpus. The TU patterns were also used to develop dialogue manager policies to engage in these patterns, such as the type and quantity of feedback given and how to sequence feedback to the commander with translations to the robot navigator.

7. Conclusion

We have presented a new annotation scheme for meso-level dialogue structure in multi-floor dialogue. The scheme covers Transaction units that accomplish collaborative goals, sometimes across multiple floors, and relations between individual utterances in the transaction unit,. The scheme has been used to annotate an initial corpus with two floors, in a human-robot interaction scenario. We have presented statistics of the different types of relations and transaction structures present in the corpus, as well as introduced some of the ways that the annotated corpus is being used.

8. Acknowledgements

The effort described here has been supported by the U.S. Army. Any opinions, content or information presented does not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

9. Bibliographical References

Allwood, J., Nivre, J., and Ahlsen, E. (1992). On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9.

Bonial, C., Marge, M., Artstein, R., Foots, A., Gervits, F., Hayes, C. J., Henry, C., Hill, S. G., Leuski, A., Lukin, S. M., Moolchandani, P., Pollard, K. A., Traum, D., and Voss, C. R. (2017). Laying Down the Yellow Brick Road: Development of a Wizard-of-Oz Interface for Collecting Human-Robot Dialogue. In *AAAI Fall Symposium*.

Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A. C., Hasida, K., Petukhova, V., Belis-Popescu, A., and Traum, D. (2012). ISO 24617-2: A semantically-based standard for dialogue annotation. In *International Conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey, May.

Carletta, J., Isard, A., Isard, S., Kowtko, J., Doherty-Sneddon, G., and Anderson, A. (1996). HCRC dialogue structure coding manual. Technical Report 82, HCRC.

Clark, H. H. and Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science*, 13:259–294.

Condon, S. and Cech, C. (1992). Manual for coding decision-making interactions. unpublished manuscript, updated May 1995, available at: ftp://sls-ftp.lcs.mit.edu/pub/multiparty/coding_schemes/condon.

Edelsky, C. (1981). Who's got the floor? *Language in Society*, 10(3):383–421.

Grosz, B. J. and Sidner, C. L. (1986). Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Krippendorff, K., (1980). *Content Analysis: An Introduction to Its Methodology*, chapter 12, pages 129–154. Sage, Beverly Hills, CA.

Leuski, A. and Traum, D. (2011). NPCEditor: Creating virtual human dialogue using information retrieval techniques. *AI Magazine*, 32(2):42–56.

Marge, M., Bonial, C., Pollard, K. A., Artstein, R., Byrne, B., Hill, S. G., Voss, C., and Traum, D. (2016). Assessing agreement in human-robot dialogue strategies: A tale of two wizards. In *International Conference on Intelligent Virtual Agents*, pages 484–488. Springer.

Marge, M., Bonial, C., Foots, A., Hayes, C., Henry, C., Pollard, K. A., Artstein, R., Voss, C. R., and Traum, D. (2017). Exploring Variation of Natural Human Commands to a Robot in a Collaborative Navigation Task. In *RoboNLP*.

Passonneau, R. (2006). Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In *Proc. of LREC*.

Prasad, R. and Bunt, H. (2015). Semantic relations in discourse: The current state of iso 24617-8. In *Proceedings 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-11)*, pages 80–92.

Reinsch, N. L., Turner, J. W., and Tinsley, C. H. (2008). Multicommunicating: A practice whose time has come? *Academy of Management Review*, 33(2):391–403.

Sinclair, J. M. and Coulthard, R. M. (1975). *Towards an analysis of Discourse: The English used by teachers and pupils.* Oxford University Press.

Traum, D. and Larsson, S. (2003). The information state approach to dialogue management. In Jan van Kuppevelt et al., editors, *Current and New Directions in Discourse and Dialogue*, pages 325–353. Kluwer.

Traum, D. R. and Nakatani, C. H. (1999). A two-level approach to coding dialogue for discourse structure: Activities of the 1998 working group on higher-level structures. In *Proceedings of ACL 1999 Workshop: Towards Standards and Tools for Discourse Tagging*, pages 101–108.

Traum, D. R. (1994). *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. thesis, Department of Computer Science, University of Rochester. Also available as TR 545, Department of Computer Science, University of Rochester.

Effects of Gender Stereotypes on Trust and Likability in Spoken Human-Robot Interaction

Matthias Kraus, Johannes Kraus, Martin Baumann, Wolfgang Minker
Institute of Communications Engineering, Department of Human Factors
University of Ulm, Germany
{matthias.kraus, johannes.kraus, martin.baumann, wolfgang.minker}@uni-ulm.de

Abstract

As robots enter more and more areas of everyday life, it becomes necessary for them to interact in an understandable and trustworthy way. In many regards this requires a human-like interaction pattern. This research investigates the influence of gender stereotypes on trust and likability of humanoid robots. In this endeavor, explicit (name and voice) and implicit gender (personality) of robots have been manipulated along with the stereotypicality of a task. 40 participants interacted with a NAO robot to gain feedback on a task they were working on and rated the perception of the robot cooperation partner. While no gender stereotypes were found for the explicit gender, implicit gender showed a strong effect on trust and likability in the stereotypical male task. Participants trusted the male robot more and rated it as more reliable and competent than the female personality robot, while the female robot was perceived as more likable. These findings indicate that for gender stereotypes in robot interaction a differentiation between explicit and implicit stereotypical features have to be drawn and that the task context needs consideration. Future research may look into situational variables that drive stereotypification in human-robot interaction.

Keywords: Human-Robot Interaction, Trust, Gender Stereotypes

1. Introduction

Due to the recent rapid growth of artificial intelligence, robots will no longer be mere substitutions for labor-intensive and repetitive tasks like for example in industrial automation. As they manage to handle more and more complex tasks, they will soon find their way into our daily lives and become our companions. In order to be accepted and trusted as a *social* companion, future robots need to possess human-like characteristics and social skills. Hence, several studies investigated the effects of gender and personality features in human-robot interaction (Jonsson and Dahlbäck, 2013; Park et al., 2012; Tay et al., 2014; Löckenhoff et al., 2014). Exemplarily, the results Park et al. (2012) showed that people who interacted with a robot of similar personality (extraversion - introversion) felt more comfortable than those who engaged with a robot of different personality. According to Joosse et al. (2013) when considering the robot's gender and personality, the task context in which the interaction takes place is also of great importance. They found that preferences for robot personalities depend on the robot's role and the stereotype perception people hold for certain tasks. For example, extroverted participants perceived similarity attraction when the robot was a tour guide, while introverted subjects perceived similarity attraction when the robot was a cleaner. Hence, the consistency of a robot's behavior and the context of a given task or role seems to play a role for perception of the robot.

In the scope of this work, we integrated gender stereotypical communication style (male - female personality traits) and gender typical characteristics (male - female voice) into a spoken dialogue robot-based assistance system and investigated interaction in a stereotypical female (baby health care) and stereotypical male task (taxi ordering) scenario. Furthermore, we considered the effects of the different configurations on trust, and on other related variables like predictability or competence, as well as on the likability of the robot. According to a stereotype perspective, the emphasis of this study was the comparison of matching (e.g. male personality traits - male task scenario) and non-matching conditions (e.g. female voice - male task scenario).

The outline of this paper is as follows: in Section 2 an overview of related work is provided. Particularly, we will focus on general gender traits and existing stereotypes as well as on trust and likability in human-machine interaction. Section 3 deals with the integration of gender stereotypes in spoken human-robot dialogue using the humanoid robot Aldebaran NAO. Here, we describe the implementation of explicit (name and voice) and implicit (personality traits) gender characteristics in the interaction. Furthermore, the experimental setup is described in detail. In Section 4 the results of the study are presented. Subsequently, the paper is concluded in Section 5 with a discussion of the found results and an outlook to future work.

2. Related Work

2.1. Gender Traits and Stereotypes

There are different personality traits which are attributed uniformly to men or women (Huddy and Terkildsen, 1993). Löckenhoff et al. (2014) investigated the perceived gender differences in five-factor personality traits (Big-Five) in terms of different nations and age groups and whether theses perceived differences reflect already assessed gender differences in personality accurately. They concluded that women were perceived as more open, more conscientious, and more agreeable than men. There were also higher ratings for women in warmth (facet of extraversion) and anxiety, depression, self-consciousness and vulnerability (facets of neuroticism), whereas men were higher rated in terms of excitement seeking and assertiveness. These perceived differences were consistent across age groups and nations and reflected closely the assessed gender differences in personality (self- and observer-ratings).

Otterbacher et al. (2017) investigated the strength and content of gender stereotypes in image search algorithms using a trait adjective list based on Abele et al. (2008). The developed trait list contains stereotypically adjectives describing both female and male personality traits. For example, representative adjectives for the female trait warmth are attributes like emotional, warm or fair. In contrast, the male trait competence was described by adjectives like competent, consistent or intelligent. The results are in line with common ground of social psychological research findings, where female traits are more correlated to warmth and male traits rather correlated to competence (Ruble et al., 1998). Otterbacher et al. (2017) pointed out that regardless which wording participants use for their descriptions, the two dimensions (warmth and competence) can be found in a cluster analysis. These findings are in line with Paulhus and Trapnell (2008) who postulate the Big Two, a concept paralleling the Big Five for gender-specific traits. The Big Two are communion (warmth) and agency (competence) (Bakan, 1966). Abele and Bruckmüller (2011) mention similar classifications: the adjectives warmth, friendliness, trustworthiness (female traits) and ambitious, competent, self-confident (male) describe typical gender traits.

2.2. Trust and Likability in Human-Machine Interaction

Trust has been investigated in regard to human-machine interaction over the last 20 years. It has been found to play a role in many different professional and everyday domains like automated driving (Hergeth et al., 2016) or online shopping (Gefen et al., 2003). Trust can be defined as "the attitude that an agent will help achieve an individual's goals in a situation characterised by uncertainty and vulnerability" (Lee and See, 2004, p.51). In human-robot teamwork, trust reflects the person's willingness to accept information and suggestions provided by the robot (Hancock et al., 2011).

In human-human interaction, the likability of the social interaction partner can be judged within seconds and this judgment has been found to be a strong predictor for interaction with a robot (Clark and Rutter, 1985; Robbins and DeNisi, 1994). Thus, besides trust likability of the robot acting as a social interaction partner (Bartneck et al., 2008) seemed to be worthwhile considering in our study as a dependent variable.

2.3. Hypotheses

The two papers Tay et al. (2014) and Tay et al. (2013) investigated the impact of occupational roles (security - health care), personality (extroverted - introverted), and gender (male - female) of a robot on user acceptance. Personality was manipulated by non-verbal cues, while the gender of the robot was varied by gender-specific names and voices (male - female). In these experiments subjects accepted robots with conformed gender and personality to the respective role stereotype more. In contrast to these studies, we choose a more content-related manipulation of personality traits and stereotypes to the robot with verbal cues. Additionally, in this research a broader range of outcome variables was measured, e.g., trust in the robot.

According to our preceding considerations, we tested the following hypotheses:

H1: Implicit male gender of the robot lead to higher ratings of trust (H1.1), reliability (H1.2), predictability (H1.3), and competence (H1.4) in the male task scenario (matching-condition) as compared to female personality traits.

H2: Implicit female gender of the robot lead to higher ratings of trust (H2.1), reliability (H2.2), predictability (H2.3), and competence (H2.4) in the female task scenario (matching-condition) as compared to male personality traits.

H3: Explicit male gender of the robot leads to higher ratings of trust (H3.1), reliability (H3.2), predictability (H3.3), and competence (H3.4) in the male task scenario (matching-condition) as compared to a female voice.

H4: Explicit female gender of the robot leads to higher ratings of trust (H4.1), reliability (H4.2), predictability (H4.3), and competence (H4.4) in the female task scenario (matching-condition) as compared to a male voice.

H5: Female personality traits of the robot lead to higher ratings of likability in the female (H5.1) and the male task scenario (H5.2) as compared to male personality traits.

3. Integration of Gender Stereotypes in Spoken Human-Robot Dialogue

For integrating gender stereotypes in human-robot interaction, the humanoid robot Aldebaran NAO produced by the SoftBank Robotics Group was set up as a dialogue partner in two stereotypical scenarios. A main feature of the NAO robot is the ability to engage in a multimodal interaction with users through speech, gestures and gaze. Since this work focuses on verbal interaction, only the speech capability of NAO was used to manipulate the variables of interest, while the other features were fixed in "autonomous life" [1] mode. Gender of the NAO robot was manipulated at two ways reflecting the two independent variables of the study: first, an explicit gender manipulation by typical male vs. female names and voices, and second, an implicit gender manipulation by gender specific personality traits through modeling gender specific wording and communication style of the robot's utterances. Furthermore, as a third independent variable gender stereotypical tasks were included in the study. For a stereotypical female area of work the health care and for a typical male domain the taxi domain were chosen. In each scenario, users had to solve several scripted tasks in a Wizard-of-Oz setup, where NAO evaluated the users' action by providing either positive oder negative spoken feedback.

[1] In this mode, NAO is in an upright position and wags slightly waiting for input. The head of NAO is oriented to nearest recognised person.

3.1. Design of Gender-Specific Personality Traits, Scenarios and Voice

In order to manipulate explicit gender of the robot's voice, the designed utterances of the robot were transformed into spoken language using IBM Watson Text to Speech Demo (available at http://text-to-speech-demo.mybluemix.net/) with the two available German voices (female - male) and played to the user.
The implicit gender (personality) of the robot was modeled according to the findings of Abele and Bruckmüller (2011). Hence, the robot's stereotypical male utterances were designed to be *dominant*, *confident* and *assertive*, whereas the stereotypical female utterances were *agreeable* and *warm*. In order to make the two conditions comparable, the meaning and general content of the utterances were kept constant in both conditions. In the following an exemplary utterance for negative feedback for each condition in the taxi scenario is depicted:

NAO male personality

> *"This is not correct. Due to several road construction works and traffic jams taxis C,D and A would require more time to reach the destination than taxi B. Keep this in mind for the future, please"*

NAO female personality

> *"Unfortunately, you picked the wrong answer. There is a better solution. Taxi C, D and A would have force themselves through far more road construction works and traffic jams than taxi B. However, I'm sure that you are conscious of that by now. So don't worry."*

In the female personality condition, the feedback was more submissive and tolerant in case of wrong answers by the user, which was intended to convey an impression of warmth and agreeableness of the robot. Contrary, the male NAO's feedback was rather strict and uncommunicative in order to give the user the perception of a competent and self-confident dialogue partner. The selection of the respective scenarios was derived from the findings of Williams and Best (1977). In the taxi ordering domain, the user was told to imagine that he would be working for a taxi company and was responsible for taking requests and sending the right amount of taxis to the correct places. In order to act in the company's best interests, the user should choose the most profitable option but also ensure satisfaction of the users' needs. As this task strongly relates to the adjectives logical, rational and methodical which were deemed stereotypical male attributes, the taxi ordering scenario was selected as male domain. In the baby health care domain, the user was told to be responsible for the nursing of a baby and had to make decisions on the appropriate handling of several situations, like the feeding or bathing of the baby. This task is strongly associated with the need of social skills and warmth. Additionally, people expected emotional competence and sensitive handling according to Williams and Best (1977). Therefore, this scenario was chosen as a task for a stereotypical female domain.

3.2. Experimental Setup

3.2.1. Participants

40 German participants were recruited at Ulm University and received 6 Euro in return for their participation. Two participants had to be excluded due to technical issues with the NAO robot. As a consequence, the data of 38 participants could be further analysed. 12 subjects were female, 26 were male, and the age ranged from 19 to 50 years with a mean age of M = 26.34 (SD = 7.38).

3.2.2. Experimental Design and Manipulations

	Task Order Scenario	
Gender Traits	**Baby Health Care**	**Taxi Ordering**
	Male voice Male personality	Male voice Male personality
	Female voice Male personality	Female voice Male personality
	Male voice Female personality	Male voice Female personality
	Female voice Female personality	Female voice Female personality

Table 1: Experimental conditions of the study: eight study groups from the combination of three independent variables with 2 levels.

In our experiment, we assessed trust, reliability, predictability, competence, acceptance and likability of the robot as dependent variables. Each variable was measured with items from established and with items from established and validated scales that were translated into German and slightly modified for content and study context. All scales were assessed with a 7-point Likert scale ranging from 1="totally disagree" to 7="totally agree". A 2x2x2 mixed factorial experimental design was conducted with the robot's implicit and explicit genders as between-independent and the task scenarios (baby - taxi) as within-independent variables. Additionally, the order of the scenarios was randomised leading to an overall of eight study groups to which participants were randomly assigned (see Table 1). As a cover story the subjects were told that they had to test a new robotic assistant for training which will be deployed in the company they were allegedly working for. Specifically, they had to work on five questions per scenario which referred to the specific task scenario at hand. The respective tasks were presented on a laptop computer, which was also used by the participants to fill in the questionnaires. The answers the subjects had to give were predefined, i.e. they had to give either a correct or a wrong answer per question corresponding to a script they were handed. Each time, the robot provided positive or negative feedback through speech after the participant uttered the scripted answer. As a Wizard-of-Oz paradigm was applied in this experiment, the system's feedback was triggered remotely by the wizard from an external desktop.

3.2.3. Experimental Procedures

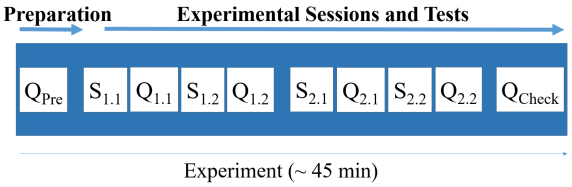


Figure 1: Procedure of experiment. After two scenario sessions (S), dependent variables were assessed. The final questionnaire of the experiment gives insights to the manipulation of the independent variables.

After the welcome procedure, the subjects were provided with first instructions and details about the study. As a second step, they had to read and sign the informed consent, and had to fill a pre-test questionnaire about their personality. Before the first interaction cycle, they received information about the tasks and the procedure of the study. This included details about the feedback functionality of NAO and about the task to rate the interaction with the robot. At the beginning of each scenario, NAO introduced him- or herself. Subsequently, the participant had to work on the first two questions, fill in a questionnaire to assess the dependent variables, and continue with the remaining three questions on the respective task. The scenario was then ended with the completion of the same questionnaire in order to gain a more robust evaluation. The same procedure was repeated for the second task scenario. In conclusion, a last questionnaire containing manipulation checks, demographics and possible confounding variables had to be filled out.

4. Results

All scales used to assess the dependent variables showed acceptable Cronbach's Alphas (all alphas $>$ .82; scales ranging from a minimum of four to a maximum of seven items). No significant outliers were found. To rule out confounding group differences for the study conditions, we controlled for the subject's acceptance of technical and electronic devices, their preliminary trust towards assistive systems, their experience with spoken dialogue systems, as well as their attitude towards (humanoid) robots as confounding variables. All 2x2 ANOVAs testing these differences did not show any significant group differences (all p-values $>$ .05/3). In addition, participants age and gender was similarly distributed in the different experimental groups.

A manipulation check was conducted with a series of independent t-tests concerning the gender and quality of the used voices, gender-specific personalities of the robot, as well as gender-stereotypical task scenarios. Therefore, we tested each gender-manipulated variable on perceived masculinity and femininity by the user. In both scenarios the manipulation of the gender-specific voices worked (all p-values $<$ .05). The manipulation of the gender-specific personality of the robot also worked in both scenarios (see Table 2). However, contrary to the generated "empathic" female personality being recognised as more female in both scenarios (all p-values $<$ .05), the generation of a "dominant" male personality was not perceived as significantly more male by the participants. The taxi ordering and baby healthcare scenario were correctly rated as a typically male and respectively female task ($p < .05$). In order to guarantee comparability of the two voice conditions exopect for perceived gender, six additional characteristics of the voices were assessed. Figure 2 shows the measured quality of each voice using six subjective metrics. Except for the naturalness and pleasantness of the voice ("The voice is natural" $[F(1, 36) = 8, 155, p < .05]$; "The robot is a pleasant conversation partner" $[F(1, 36) = 2, 502, p < .05]$) there were no significant differences.

We measured the dependent variables at two times during the interaction with the robot. The first time of measurement was to assess the initial reaction towards the robot while the second time of measurement sought to measure a stabilised evaluation after some interaction with the robot was experienced. Following the reasoning that trust and the assessment of its components competence, reliability and predictability fluctuates during early interaction with a new system (Lee and See, 2004; Hoff and Bashir, 2015), for these dependent variables only the averaged scale values of

Gender Personality Manipulation Check				
	Taxi		**Baby**	
	Male *M (SD)*	Female *M (SD)*	Male *M (SD)*	Female *M (SD)*
Male Personality Check	4.80 (1.49)	4.54 (1.01)	5.06 (1.06)	4.62 (1.13)
Female Personality Check	2.62 (1.11)	4.25 (1.12)	2.99 (1.31)	4.57 (1.03)

Table 2: Descriptive statistics of the manipulation check regarding the different personality traits of NAO.

	Gender Personality			
	Taxi		**Baby**	
	Male *M (SD)*	Female *M (SD)*	Male *M (SD)*	Female *M (SD)*
Trust	5.57 (.67)	4.95 (1.13)	4.90 (1.02)	5.00 (1.11)
Reliability	5.61 (.71)	4.96 (1.39)	4.98 (1.03)	5.09 (1.05)
Predictability	5.86 (.88)	4.98 (1.54)	4.99 (1.28)	4.90 (1.34)
Acceptance	5.05 (.90)	5.02 (1.27)	4.96 (1.06)	5.09 (1.29)
Likability	4.04 (.94)	5.15 (1.23)	4.51 (.74)	5.24 (1.09)
Competence	5.89 (.77)	5.14 (1.34)	5.59 (1.03)	5.34 (1.30)

Table 3: Descriptive statistics with reference to gender personality and task scenario.

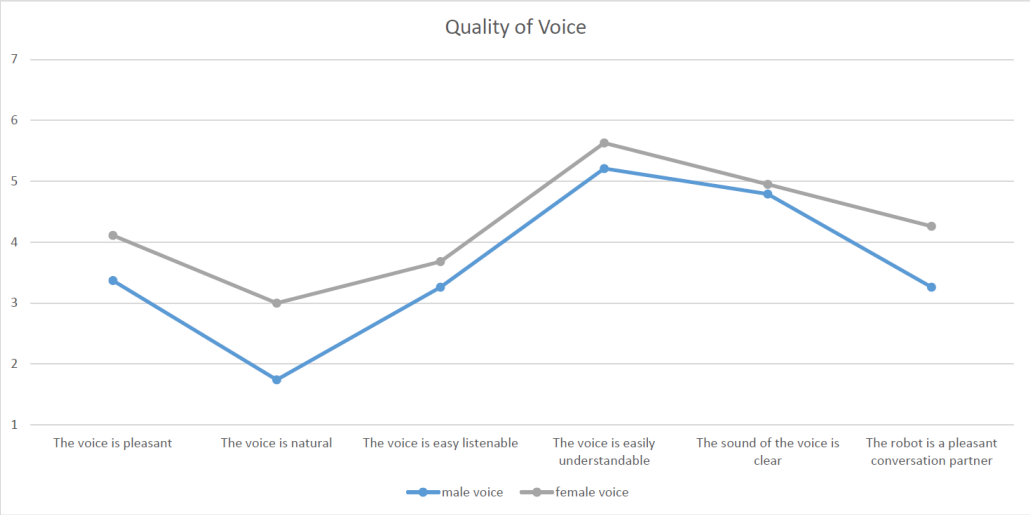


Figure 2: Diagram showing the rated scales for the measurement of quality of the used voices. The quality was rated on a 7-point Likert scale for six items.

the second time measurement was included in the analysis. For acceptance and likability an aggregate of the two points of measurement was used.

As a starting point, three-way mixed ANOVAs for all dependent variables were conducted. The three-way interaction of all independent variables did not yield promising results (all $p > .05$). Therefore, it is not further considered in the following. Similarly, all interactions and main effects including the explicit gender missed significance. This lead to the decision to leave this factor out of further analyses for this study (all $p > .05$). For all further analyses, the dependent variable acceptance showed no significant differences - neither for the gender manipulations nor for the different scenarios (all $p > .05$). The two-way interaction of the factors scenario (babycare vs. taxi) and implicit gender were significant for trust $[F(1, 34) = 5.968, p < .05]$ and predictability $[F(1, 34) = 4.553, p < .05]$. For both variables the robot was significantly rated higher for the male personality as compared to the female personality in the taxi ordering scenario (see Table 3 for detailed descriptives). Although the p-values for reliability $[F(1, 34) = 3.454, p = .072]$, competence $[F(1, 34) = 2.676, p = .111]$ and liking $[F(1, 34) = 2.475, p = .125]$ missed significance, in face of the comparatively low power of this analyses and the small sample size of this study, we conducted post-hoc t-tests to further inspect these results for all dependent variables.

These test revealed no significant effect for competence in both scenarios (all $p > .05$). For competence there was no significant difference found for the implicit gender in the baby scenario, but in the taxi scenario a robot with a male personality was rated significantly more competent $[t(36) = 2.047, p < .05]$. Conversely, likability was rated significantly higher for the female personality of the robot in both the baby care $[t(36) = 2.352, p < .05]$ and the taxi ordering scenario $[t(36) = 3.064, p < .05]$. For visualisation, the dependent variables are presented for both tasks in Figure 3.

5. Discussion and Future Work

The study provides evidence that gender stereotypes can indeed be replicated in the context of human-robot interaction and that their consideration in spoken dialogue systems can positively affect the human-machine interaction.

The results showed that the personality condition has a major effect on the interaction within the stereotypically male-designed taxi ordering domain. The male personality was perceived more trustworthy, reliable, competent and predictable than NAO's female personality condition (hypothesis H1 verified; H1.2 not significant). In the stereotypically female-designed baby health care domain our assumption to expect the reversed effect was not validated and showed only a slight tendency regarding trust and reliability (hypothesis H2 rejected). An explanation why the expected effects could not be observed in the female domain poses the design of the scenario itself.

The baby health care scenario was designed in accordance with attributes like emotional competence, warmth and social skills. As the study was conducted in a highly fictionous environment and the subjects had to make scripted decisions from a rather distant point of view, participants possibly could not form an emotional bond to the task and did not require the robot to be warm or emotionally competent. Another explanation could be the rather masculine appearance of NAO itself. As visual cues play an important role in the perception of gender stereotypes (Hall, 1978), the masculinity of NAO could have had an negative effect on the female personality condition and the perceived gender of the robot.

Besides, the fact that the manipulation of the male personality of NAO did not work as well as the manipulation of the female personality traits may have contributed to a smaller difference in the perception of the robot. This especially seemed to hold true for the female task scenario, which could not be pushed by the masculinity of the task and hence resulted in almost no visible distinction.

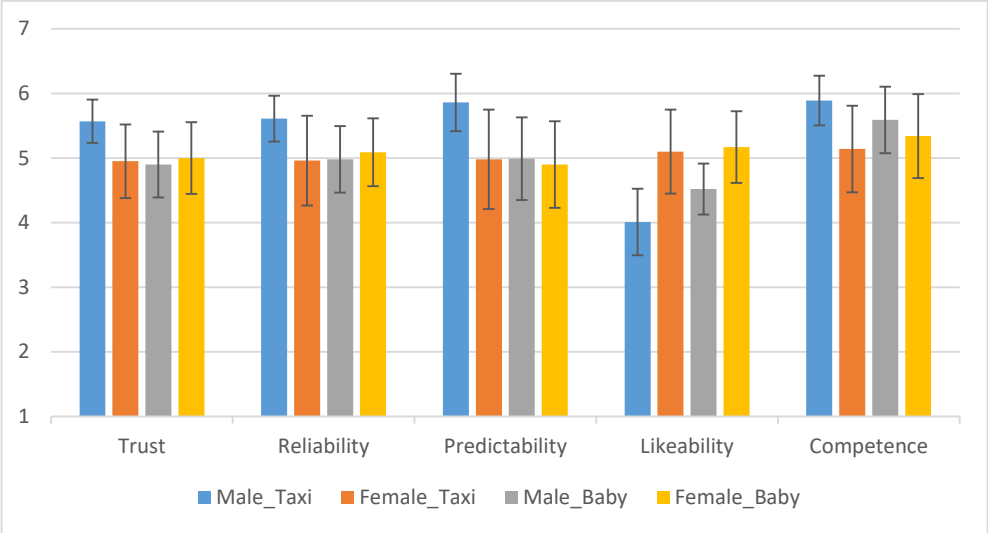


Figure 3: Bar charts for the variables trust, reliability, predictability, likability and competence. The terms "Male" and "Female" refer to the implicit gender of the robot, while results are provided for both task scenarios. Mean values can be taken from Table 3.

Surprisingly, the voice condition had no effect on the interaction in the stereotypical domains (hypothesis H3 and H4 rejected). Hence, it could be possible that personality traits have a larger impact on gender stereotypes than a female or male sounding voice. In order to prove this claim further investigation is necessary. Measurement of the quality of the used voices showed a significant difference in naturalness of the male and female voice. However, even though the female voice was perceived as more natural, this circumstance did not lead to measurable effects in the experiment. As one would expect that a more natural voice would result in more trust in the interlocutor, this forms a kind of paradox, but reinforces the impression that personality traits have a bigger impact than voice alone.

Furthermore, the study revealed that gender personalities effect significantly the likability of human-machine interaction. Since the female personality traits were designed more agreeable and warm, the female personality was perceived more likable in both scenarios (H5 verified; H5.2 not significant). An interesting finding was that trust and likability did not always seem to correlate, but it seems that robots of different characteristics seem to be more trustworthy in different domains - despite the fact that robots of a more female personality are liked more irrespective of the domain under investigation. Hence, a trustworthy system is not necessarily more likable. Is it possible to maintain both trustworthiness and likability in spoken human-robot interaction? Further research should provide more insight on this coherence. Additionally, likability and acceptance do not seem to interdepend gender-personality wise. Thus, robots of all genders are accepted but may show differences in likability depending on the personality. This implicates for human-robot interaction designers to not only concentrate on a high likability of their systems because rather emotionally cold systems may be equally accepted.

In order to avoid the drawbacks of this experimental setup in future work, we plan on conducting the study without a humanoid robot and turn to a embedded voice-only control system like Amazon Echo. In doing so, we could eliminate the visual side-effects of perceived gender stereotypes. Furthermore, a more sophisticated approach for modeling the female domain is necessary. Therefore, the future participants should not work on scripted tasks, but make their own decisions in a more realistic environment. Hence, subjects could get emotionally involved in the task. Most importantly, a higher sample size and a better manipulation of the male personality characteristics are necessary to give further insight in this quite interesting topic.

6. Bibliographical References

Abele, A. E. and Bruckmüller, S. (2011). The bigger one of the "big two"? preferential processing of communal information. *Journal of Experimental Social Psychology*, 47(5):935–948.

Abele, A. E., Uchronski, M., Suitner, C., and Wojciszke, B. (2008). Towards an operationalization of the fundamental dimensions of agency and communion: Trait content ratings in five countries considering valence and fre-

quency of word occurrence. *European Journal of Social Psychology*, 38(7):1202–1217.

Bakan, D. (1966). The duality of human existence: An essay on psychology and religion.

Bartneck, C., Croft, E., and Kulic, D. (2008). Measuring the anthropomorphism, animacy, likeability, perceived intelligence and perceived safety of robots. In *Metrics for HRI workshop, technical report*, volume 471, pages 37–44.

Clark, N. K. and Rutter, D. R. (1985). Social categorization, visual cues, and social judgements. *European Journal of Social Psychology*, 15(1):105–119.

Gefen, D., Karahanna, E., and Straub, D. W. (2003). Trust and tam in online shopping: An integrated model. *MIS quarterly*, 27(1):51–90.

Hall, J. A. (1978). Gender effects in decoding nonverbal cues. *Psychological bulletin*, 85(4):845.

Hancock, P. A., Billings, D. R., Schaefer, K. E., Chen, J. Y., De Visser, E. J., and Parasuraman, R. (2011). A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors*, 53(5):517–527.

Hergeth, S., Lorenz, L., Vilimek, R., and Krems, J. F. (2016). Keep your scanners peeled: Gaze behavior as a measure of automation trust during highly automated driving. *Human factors*, 58(3):509–519.

Hoff, K. A. and Bashir, M. (2015). Trust in automation: Integrating empirical evidence on factors that influence trust. *Human Factors*, 57(3):407–434.

Huddy, L. and Terkildsen, N. (1993). Gender stereotypes and the perception of male and female candidates. *American Journal of Political Science*, pages 119–147.

Jonsson, M. and Dahlbäck, N. (2013). In-car information systems: Matching and mismatching personality of driver with personality of car voice. In *International Conference on Human-Computer Interaction*, pages 586–595. Springer.

Joosse, M., Lohse, M., Pérez, J. G., and Evers, V. (2013). What you do is who you are: The role of task context in perceived social robot personality. In *Robotics and automation (ICRA), 2013 IEEE international conference on*, pages 2134–2139. IEEE.

Lee, J. D. and See, K. A. (2004). Trust in automation: Designing for appropriate reliance. *Human factors*, 46(1):50–80.

Löckenhoff, C. E., Chan, W., McCrae, R. R., De Fruyt, F., Jussim, L., De Bolle, M., Costa Jr, P. T., Sutin, A. R., Realo, A., Allik, J., et al. (2014). Gender stereotypes of personality: universal and accurate? *Journal of Cross-Cultural Psychology*, 45(5):675–694.

Otterbacher, J., Bates, J., and Clough, P. (2017). Competent men and warm women: Gender stereotypes and backlash in image search results. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 6620–6631. ACM.

Park, E., Jin, D., and del Pobil, A. P. (2012). The law of attraction in human-robot interaction. *International Journal of Advanced Robotic Systems*, 9(2):35.

Paulhus, D. L. and Trapnell, P. D. (2008). Self-presentation of personality. *Handbook of personality psychology*, 19:492–517.

Robbins, T. L. and DeNisi, A. S. (1994). A closer look at interpersonal affect as a distinct influence on cognitive processing in performance evaluations. *Journal of Applied Psychology*, 79(3):341.

Ruble, D. N., Martin, C. L., and Berenbaum, S. A. (1998). Gender development. *Handbook of child psychology*.

Tay, B. T. C., Park, T., Jung, Y., Tan, Y. K., and Wong, A. H. Y. (2013). When stereotypes meet robots: The effect of gender stereotypes on people's acceptance of a security robot. In *International Conference on Engineering Psychology and Cognitive Ergonomics*, pages 261–270. Springer.

Tay, B., Jung, Y., and Park, T. (2014). When stereotypes meet robots: the double-edge sword of robot gender and personality in human–robot interaction. *Computers in Human Behavior*, 38:75–84.

Williams, J. E. and Best, D. L. (1977). Sex stereotypes and trait favorability on the adjective check list. *Educational and Psychological Measurement*, 37(1):101–110.

A Multimodal Corpus for Mutual Gaze and Joint Attention in Multiparty Situated Interaction

Dimosthenis Kontogiorgos[1], Vanya Avramova[1], Simon Alexanderson[1], Patrik Jonell[1], Catharine Oertel[1 2], Jonas Beskow[1], Gabriel Skantze[1], Joakim Gustafson[1]

[1] KTH Royal Institute of Technology, Sweden
[2] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Abstract

In this paper we present a corpus of multiparty situated interaction where participants collaborated on moving virtual objects on a large touch screen. A moderator facilitated the discussion and directed the interaction. The corpus contains recordings of a variety of multimodal data, in that we captured speech, eye gaze and gesture data using a multisensory setup (wearable eye trackers, motion capture and audio/video). Furthermore, in the description of the multimodal corpus, we investigate four different types of social gaze: referential gaze, joint attention, mutual gaze and gaze aversion by both perspectives of a speaker and a listener. We annotated the groups' object references during object manipulation tasks and analysed the group's proportional referential eye-gaze with regards to the referent object. When investigating the distributions of gaze during and before referring expressions we could corroborate the differences in time between speakers' and listeners' eye gaze found in earlier studies. This corpus is of particular interest to researchers who are interested in social eye-gaze patterns in turn-taking and referring language in situated multi-party interaction.

Keywords: multimodal situated interaction, social eye-gaze, referential gaze, joint attention, mutual gaze, reference resolution

1. Introduction

In this corpus we combine verbal and non-verbal communication cues to model shared attention in situated interaction. We capture multimodal cues using a multisensory setup in order to extract information on visual attention during deictic references in collaborative dialogue. We made use of a moderator who had the task of leading the interaction and making sure that both participants were involved in the task and in the conversation. During the interaction the moderator used referring expressions and gaze to direct the participants' attention to objects of interest on a large touch screen (figure 1).



Figure 1: Participants used referring expressions and gaze to direct each other's attention while moving objects on a large display.

Our purpose of collecting the corpus presented in this paper is to study social eye gaze in multiparty interaction. The goal is to create visual attention models that make it possible for our robots to direct humans' attention to certain objects. There are several novelties with our corpus: Firstly, it is a three-party interaction with and without an interactive touch screen where we have synchronised data streams of gaze targets, head direction, hand movement and speech (run through ASR with word timings). Second, one of the three participants is a mediator that has the role of encouraging the participants to reconsider their decisions and to foster collaboration. In all recordings we use the same person as the mediator, which makes it possible for us to build coherent verbal and non-verbal behavioural models. We will use these to develop a robot that can be used as a moderator in similar collaborative tasks. To our knowledge there is no other publicly available corpus with these features.

During their interaction, the participants collaborated to furnish a virtual apartment using a collection of available furniture objects. It was their task to discuss and decide on which objects they would use, given that they had a limited budget. The moderator had the role of leading the discussion. However, this does not change the fact that this is an example of dynamic multiparty situated interactions that (Bohus and Horvitz, 2009) define as an open-world dialogue. When discussing the furniture objects, the participants naturally made use of a combination of verbal referring expressions and non-verbal cues such as deictic gestures and referential gaze. We are particularly interested in the participants' gaze behaviour just before and during verbal referring expressions. When analysing our corpus we find that listeners and speakers display different visual behaviour in some cases.

In the next sessions, we provide an overview of the state-of-the-art in multimodal multiparty corpora and relevant works in various types social eye gaze. We further describe our process in data collection and experiment design, and our methods for automatic eye gaze extraction. Finally, we go through the collected data and give an overview of the participants' gaze behaviour during the task-oriented dialogues and our findings supported by the relevant literature.

2. Background

2.1. Multiparty multimodal corpora

Over the last decade, more and more multimodal multiparty corpora have been created, such as the ones described in (Carletta, 2007; Mostefa et al., 2007; Oertel et al., 2014; Hung and Chittaranjan, 2010; Oertel et al., 2013; Stefanov and Beskow, 2016). (Carletta, 2007) and (Mostefa et al., 2007) fall into the category of meeting corpora, (Hung and Chittaranjan, 2010) and (Stefanov and Beskow, 2016) are examples of games corpora, and (Oertel et al., 2014) a job interviewing corpus. The corpus from (Oertel et al., 2013) in contrast to the corpora listed above tries to escape the lab environment and gathers data of multi-party interactions "in the wild".

In (Carletta, 2007), (Mostefa et al., 2007) and (Oertel et al., 2013) the visual focus of interlocutors is divided; i.e. there are stretches in the corpus in which interlocutors' main focus of attention is on each other and there are stretches in the corpus where they mainly focus on e.g. the white board or sheets of papers which are laying in front of them. However, with the technical set-up used at the recordings at the time, it was hard to infer the exact points the participant were looking. In (Oertel et al., 2014) and (Hung and Chittaranjan, 2010) participants' visual focus of attention is solely focused on the other participants (no other objects are present during the recordings).

Another example is (Stefanov and Beskow, 2016) who, in their corpus, study the visual focus of attention of groups of participants. For this, they recorded groups of three participants while they were sorting cards. They also recorded groups of three participants, discussing their travel experiences without any objects present that might distract the visual focus of attention.

Finally, while (Lücking et al., 2010) is not a multiparty corpus, it should be mentioned here as it is similar to the corpus described in this paper, particularly well suited for the study of referring expressions. In terms of experimental setup, the corpus recording described in this paper is most similar to (Stefanov and Beskow, 2016).

2.2. Social eye-gaze

Social eye gaze refers to the communicative cues of eye contact between humans and is usually referred to by 4 main types (Admoni and Scassellati, 2017): 1. Mutual gaze where both interlocutors' attention is directed at each other, 2) Joint attention where both interlocutors focus their attention on the same object or location, 3) Referential gaze which is directed to an object or location and often comes together with referring language and 4) Gaze aversions that typically avert from the main direction of gaze - i.e. the interlocutor's face.

Joint attention is of particular importance for communication. It provides participants with the possibility to interpret and predict each other's actions and react accordingly. In the current corpus for example, the modeling of joint attention is of particular importance as it provides participants with the possibility to track the interlocutors' current focus of attention in the discourse (Grosz and Sidner, 1986). A common quality of joint attention is that it may start with mutual gaze to establish where is the attention of the interlocutor and end towards the referential gaze direction to the most salient object (Admoni and Scassellati, 2017).

Also, as participants are very likely to be focused more on the display than each other, joint attention will be crucial to discern whether they are paying attention to each other. Modelling of joint attention is also crucial when developing fine-grained models of dialogue processing (Schlangen and Skantze, 2009), which for example makes it possible for a dialogue system to give more timely feedback (Meena et al., 2014). With regards to multi-party interaction there are also recent studies which model the situation in which the interaction takes place, in order to manage several users talking to the system at the same time (Bohus and Horvitz, 2010), and references to objects in the shared visual scene (Kennington et al., 2013).

2.3. Multimodal reference resolution

A reference is typically a symbolic representation of a linguistic expression to a specific object or abstraction. Durining early attempts in verbal communications between humans and machines researchers used rule-based systems to disambiguate words and map them to referent objects in virtual worlds (Winograd, 1972). Research has also focused in disambiguating language using multimodal cues; starting in the late 70s with Richard Bolt's "Put-That-There" (Bolt, 1980), to recent approaches using eye-gaze (Mehlmann et al., 2014; Prasov and Chai, 2008; Prasov and Chai, 2010), head pose (Skantze et al., 2015), and pointing gestures (Lücking et al., 2015). Gross et. al recently explored the variability and interplay of different multimodal cues in reference resolution (Gross et al., 2017).

Eye gaze and head direction have been shown to be good indicators of object saliency in human interactions; researchers have developed computational methods to construct saliency maps and identify humans' visual attention (Bruce and Tsotsos, 2009; Sziklai, 1956; Borji and Itti, 2013; Sheikhi and Odobez, 2012). Typically speakers direct the listeners' attention to objects using verbal and non verbal cues and listeners often read the speaker's visual attention during referring expressions to get indications on the referent objects.

3. Data collection

3.1. Scenario

We optimised for variation in conversational dynamics by dividing the current corpus recordings into two conditions. In the first condition the moderator facilitated a discussion about participants' experience on the topic of sharing an apartment. In this condition no distracting objects were existent and the screen on the large display was off. In the second condition the participants were asked to collaborate on decorating an apartment (Figure 2) that they should imagine they would be moving in together. They were given an empty flat in which they had to decide where to place furniture which they could buy from the store. The stores were provided as extra screens on the application, that they could go through to get new pieces of furniture. They were given a limited budget which fostered their decisions and discussions on what objects to choose. Given the variety

of available objects they would need to compromise their choices and collectively decide where to place objects.

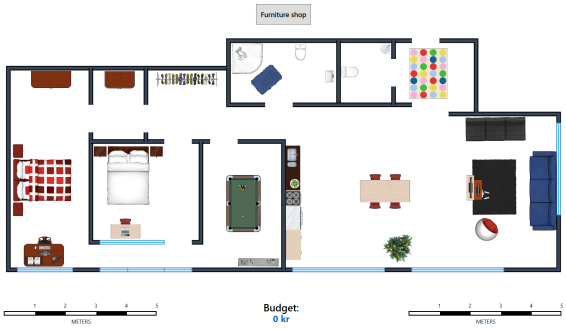


Figure 2: The application running on the large display. The objects were "bought" from the furniture shop that had a variety of available furniture.

3.2. Participants

We collected data from 30 participants with a total of 15 interactions. Our moderator (native US-English speaker) was present on all 15 sessions facilitating the structure of the interactions and instructing participants on their role for completing the task. The moderator was always at the same part of the table and the two participants were sitting across the moderator (Figure 1). The mean age of our participants was 25.7; 11 were female and 19 were male and the majority of them were students or researchers at KTH Royal Institute of Technology in Stockholm, Sweden.

3.3. Corpus

Our corpus consists of a total of 15 hours of recordings of triadic interactions. All recordings (roughly one hour each) contain data from various sensors capturing motion, eye gaze, audio and video streams. Each session follows the same structure: The moderator welcomes the participants with a brief discussion on the topic of moving in a flat with someone, and thereafter introducing the setup and scenario of the two participants planning their moving in the same apartment using the large display. During the interactions we collected a set of multimodal data: a variety of input modalities that we combined to get information on participants' decision making and intentions.

Out of the 15 sessions, 2 sessions had no successful eye gaze calibration and were discarded. One session had synchronisation issues on the screen application which was also discarded. Last, one session had large gaze data gaps and was discarded as well. The rest 11 sessions of the aggregated and processed data from the corpus are further described on the rest of the paper and are available at: `https://www.kth.se/profile/diko/page/material`.

3.4. Experimental setup

Participants were situated around a table which had a large touch display. On the display there was an application we developed to facilitate their planning of a moving in together in a flat scenario. We gave participants a pair of gloves with reflective markers and eye tracking glasses (Tobii Glasses 2[1]) which also had reflective markers on to track their position in space. The room was surrounded with 17 motion capture cameras positioned in such a way that both gloves and glasses are always on the cameras' sight.

The moderator was also wearing eye tracking glasses. Since our aim is to develop models for a robot without hands, the moderator was not wearing gloves with markers and was instructed to avoid using hand gestures. There were two cameras placed on the table capturing facial expressions and a regular video camera at a distance recording the full interaction for annotation purposes. On the glasses we also placed lavalier microphones (one per participant), in such a way that we capture the subject's voice separated from the rest of the subjects' speech and with a volume consistency.

The participants collaborated in the given scenario for 1 hour where they discussed and negotiated to form a common solution in apartment planning. At the end of the recording, we asked participants to fill a questionnaire on their perception of how the discussion went, their negotiations with the other participant and finally some demographic information. All participants were reimbursed with a cinema ticket.

3.5. Motion capture

We used an OptiTrack motion capture system[2] to collect motion data from the subjects. The 17 motion capture cameras collected motion from reflective markers on 120 frames per second. To identify rigid objects in the 3d space we placed 4 markers per object of interest (glasses, gloves, display) and captured position (x, y, z) and rotation (x, y, z, w) for each rigid object. While 3 markers are sufficient for capturing the position of a single rigid body, we placed a 4th marker on each object for robustness. That way, if one of the markers was not captured we would still identify the rigid object in space.

3.6. Eye gaze

We were interested in collecting eye gaze data for each participant in order to model referential gaze, joint attention, mutual gaze and gaze aversion in multiparty situated dialogue (figure 3). In order to capture eye gaze in 3D space we used eye tracking glasses with motion capture markers. This made it possible to accurately identify when a participant's gaze trajectory intersected objects or the other interlocutors. It also made it possilbe to capture gaze aversion, i.e. when participants gazed away when speaking or listening to one of the participants.

Gaze samples were on 50Hz and the data was captured by tracking the subjects' pupil movements and pupil size and a video from their point of reference. We placed reflective markers on each pair of glasses, and were therefore able to identify their gaze trajectory in x and y on their point of reference and then resolve it to world coordinates using the glasses' relevant position in 3d space.

The glasses using triangulation from both eyes' positions, also provided a z value, which would refer to the point

[1] `http://www.tobiipro.com/`
[2] `http://optitrack.com/`

where the trajectories from the two eyes meet. That point in space (x, y, z) we used to resolve the eye gaze trajectory in 3d space.

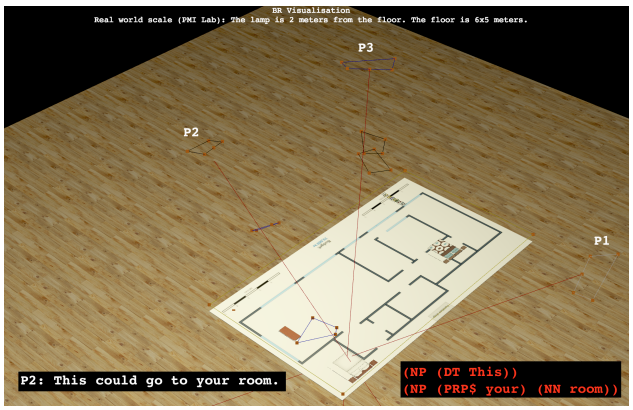


Figure 3: By combining synchronised data from motion capture and eye tracking glasses, we captured the participants' eye gaze in 3d space but also head pose and gestures.

3.7. Speech and language

We recorded audio from each participant using channel separated lavalier microphones attached to the eye tracking glasses. Each microphone captured speech that we later used to automatically transcribe using speech recognition and resolve the spoken utterances to text. We used a voice activity detection filter in all channels to separate captured speech from other speakers. We further used Tensorflow's[3] neural network parser to form syntactic representations of the automatically transcribed natural language text.

3.8. Facial expressions

In order to extract facial expressions we placed GoPro cameras in front of the participants and moderator on the table. We also recorded each session from a camera placed on a tripod further in the room to capture the interaction as a whole and for later usage in annotating the corpus.

3.9. Application

We built an application to enable the interaction with several virtual objects (figure 2). The application consisted of two main views: a floor plan and a furniture shop. The floor plan displayed on the large multi-touch display was used by the participants during their interactions as the main area to manipulate the objects while the shops were used to collect new objects.

The shop screens were divided by room categories such as kitchen or living room. They naturally induced deictic expressions as the participants referred to the objects during their negotiations using both language and referential gaze (i.e. "it", the desk", "the bed"). The floor plan on the contrary was the main display were they could manipulate objects by placing them in rooms and used referring language to these virtual locations (i.e. "here", "there", "my room").

[3] https://www.tensorow.org/versions/r0.12/tutorials/syntaxnet/

The floor plan view also displayed the apartment scale, and the budget the participants could spend for objects. The budget was limited and, after initial pilots, it was decided to have a value that would be enough to satisfy both participants' furniture selections but also limited enough to foster negotiations on what objects they would select. They were also allowed to only choose one of each item in order to foster negotiations further.

The application maintained event-based logging at 5 fps to keep track of the interaction flow, object movement and allocations. Each event carried time stamps and relevant application state, as well as positions and rotation data for all objects. By placing markers in the corners of the screen it could be placed into the motion capture coordinate system. This allowed us to get the virtual object events in 3D space and capture the participants' visual attention to these objects.

3.10. Subjective measures

At the end of each session we gave participants a questionnaire to measure their impression on how the discussion went and how well they thought they had collaborated when decorating the appartment. We used these measures to measure dominance, as well as collaborative behaviour and personality. All participants were asked to fill personality tests before coming to the lab; the tests included introversion and extroversion measures (Goldberg, 1992).

3.11. Calibration

The sensors we used required calibration in order to successfully capture motion and eye movements. We calibrated all 17 cameras positioning at the beginning of all recordings, while the eye tracking glasses required calibration on each subject separately. That is due to the fact that each participant's eye positions vary but also how their eyes move during saccades.

4. Annotations

We annotated referring expressions to objects on the display by looking at what object the speaker intended to refer to. Speakers typically drew their interlocutors' attention to objects using deictic expressions; referring language, referential gaze, as well as mutual gaze. For every dialogue there was a set of references to objects not included in our simulated environment -"I also have a fireplace in my flat, but I do not use it a lot". However, we only annotated references that can be resolved to objects existing in our simulated environment, i.e. the application on the touch display. The references we can resolve in this environment are therefore only a subset of all possible references in the dialogue.

For the annotations we used the videos of the interactions together with the ASR transcriptions and the current state of the app (visible objects or current screen on the large display). We defined each referring expression by looking at the time of the speaker's utterance. The timing was defined as from the ASR transcriptions that were synchronised with the gaze and gesture data. Utterances were split into inter pausal units (IPUs) that were separated by silent pauses longer than 250ms (Georgeton and Meunier, 2015).

The ASR transcriptions were used as input to the Tensorflow syntactic parser where we extracted the part-of-speech (POS) tags. Similarly to (Gross et al., 2017), as linguistic indicators we used full or elliptic noun-phrases such as "the sofa", "sofa", "bedroom"; pronouns such as "it" or "this" and "that"; possessive pronouns such as "my", "mine"; and spatial indexicals such as "here" or "there". The ID of the salient object or location of reference was identified and saved on the annotations. In some cases there was only one object referred to by the speaker, but there were also cases where the speaker referred to more than one object in one utterance (i.e. "the table and the chairs"). The roles of the speaker and listeners varied in each expression. In total we annotated 766 referring expressions out of 5 sessions, roughly 5 hours of recordings which was about one/third of our corpus.

5. Data processing

The variety of modalities was post-processed and analysed, where in particular combined eye gaze and motion capture provided an accurate estimation of gaze pointing in 3D space. We used a single-world coordinate system in meters on both eye gaze and motion capture input streams and we identified the gaze trajectories, head pose and hand gestures. We started with a low-level multimodal fusion of the input sensor streams and further aggregated the data together with speech and language to high-level sequences of each participant's speech and visual attention. Each data point had automatic speech transcriptions, syntactic parsing of these transcriptions, gesture data in the form of moving objects and eye gaze in the form of target object or person.

5.1. Data synchronisation

We used sound pulses from each device to sync all signals in a session. The motion capture system sent a pulse on each frame captured, while the eye tracking glasses sent 3 pulses every 10 seconds. The application also sent a pulse every 10 seconds. All sound signals were collected on the same 12 channel sound card which would then mark the reference point in time for each session.

Audio signals were also captured on the soundcard, therefore we were able to identify a reference point that would set the start time for each recording. That was the last of the sensors to start, which was one of the eye tracking glasses. Apart from the sound pulses, we used a clapperboard with reflective markers on its closed position which would mark the start and the end of each session. We used that to sync the video signals and as a safety point in case one of the sound pulses failed to start. Since the application on the display sent sync signals, we used it to mark the separation in time of the two experimental conditions. The first one (free discussion) ended when the app started which would lead to the second condition (task-oriented dialogue).

5.2. Visualisation and visual angle threshold

As illustrated in figure 3 we calculated the eye gaze trajectories by combining motion capture data and eye tracking data per frame. We implemented a visualisation tool[4] in WebGL to verify and evaluate the calculated eye gaze in 3d space. Using this tool we were able to qualitatively investigate the sections of multimodal turn taking behaviour of participants by visualising their position and gaze at objects or persons, along with their transcribed speech and audio in wav format. We also used this tool to empirically define the visual angle threshold for each eye tracker on the accuracy of gaze targets. During pilots we asked participants to look at different objects ensuring there are angle variations to identify the threshold of the gaze angle to the dislocation vector (between the glasses and the salient object).

5.3. Automatic annotation of gaze data

We extracted eye-gaze data for all participants and all sessions and noticed that there were gaps between gaze data points quite often and in some sessions more than in others. We applied smoothing to eliminate outliers and interpolated gaps in a 12 frame step (100ms on our 120 fps data). Typically an eye fixation is 250ms but no smaller than 100ms (Rayner, 1995) which defined our smoothing strategy. The data was then filtered in the same time window to only provide data points with fixations and not include saccades or eye blinks.

There were however, gaps that were longer than 12 frames, caused by lack of data from the eye trackers. In such cases the eye trackers had no information on the eyes' positions which means that there was no knowledge on if a speaker/listener looked at the referent object. In such cases of no gaze data, we went through the manually annotated referring expressions and checked the relevant frames of the gaze data. If at least one of the participants had no gaze data, we would discard the relevant referring expression from our analysis. After cleaning those cases we had remaining 582 expressions out of the 766 initially annotated[5] The average error of gaze data loss we had was 40.8%. The session with the max gaze error rate was 71.5% while the min was 26.9%.

During an referring expression participants' visual attention spanned through a) a variety of visible objects on the screen, b) their interlocutors or c) none of the above which we assumed on this corpus to be gaze aversion. The prominent objects of visual attention were identified by calculating each participant's visual angle α, between their gaze vector g to the dislocation vector o for every visible object on the screen for every frame.

$$\alpha_{ij} = \arccos\left(\frac{\vec{g_{ij}} \cdot \vec{o_{ij}}}{|\vec{g_{ij}}|\,|\vec{o_{ij}}|}\right) \tag{1}$$

Similarly, we approximated each person's head with a sphere with radius of 0.2m, and automatically annotated all frames where the gaze vector intersected the sphere as visual attention towards that person.

Finally, after filtering for eye fixations, we calculated the proportional gaze likelihood per object:

$$P(o_i \mid t_i) = \frac{c(o_i, t_i)}{c(t_i)} \tag{2}$$

[4] Available at `https://www.kth.se/profile/diko/page/material`

[5] The 5 sessions chosen for annotation were the ones with the smallest percentage of gaze data loss.

For each object o_i we gathered the proportional gaze data points and counted the amount of time the object was gazed during the time t of an utterance. As a gaze target we defined the area around any object that is within the threshold of the visual angle defined above. In many cases more than one objects competed for saliency in the same gaze target area. We therefore defined as joint attention not the group of objects gazed by the interlocutors but the gaze at the same area around the prominent object given the visual angle threshold.

6. Results

In the current section, we describe gaze patterns as observed between and across the two conditions, and proportional gaze during and before referring expressions. In figure 4, we show the proportional eye-gaze of the moderator as well as the participants' eye gaze on the display. As can be observed, the participants spent more time gazing on the display than at the moderator or at each other, which was expected to observe during a task-oriented dialogue.

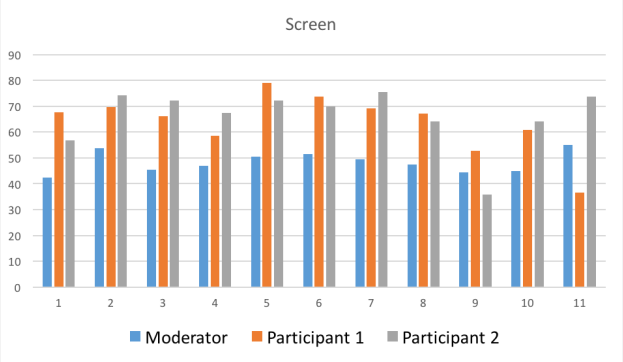


Figure 4: Proportional amount of participant eye-gaze per session on the display during the second condition (task-oriented dialogue).

For each utterance a set of prominent objects was defined from the annotations. Given the gaze targets per participant, we calculated the proportional gaze from the speaker and the listeners during the time of the utterance and exactly 1 second before the utterance. Since all interactions were triadic, there were two listeners and one speaker at all times. To compare across all utterances, we looked at the mean proportional gaze of the two listeners to the area of the prominent object to define them as the listener group. We then compared the gaze of the listener group to the speakers gaze close to the referent objects. We also looked at the proportional combined gaze to other objects during the utterance (all other objects that have been gazed at during the utterance), gaze at the speaker and averted gaze. In figure 5, the blue colour refers to the proximity area of the referent object from the speaker's utterance while orange refers to the gaze at all other known objects combined on the virtual environment. Grey is for the gaze to the listeners or the speaker during the utterance and yellow for averted gaze.

6.1. Eye gaze during referring expressions

We looked at all references to objects (N = 582) and compared the means of the proportional gaze of the speaker to the proportional listeners' gaze to the proximity area of the referent objects, the rest of the gazed objects and gaze towards each other. We conducted paired sample t-tests and found significant difference between the speaker gaze to the proximity area to the referent object (M = 46.69, Std.Error = 1.37) against the listener gaze on the area around the referent object (M = 39.45, Std.Error = 1.11) with [t = 4.942, p = 0.001]. There was no significant difference however on the speaker's gaze to the listeners against the gaze from the listeners to the speaker [t = -0.816, p = 0.415] (mutual gaze).

There was also a significant difference on the proportional gaze of the speaker to the area around the referent object (M = 46.69, Std.Error = 1.37) against the proportional gaze to other objects during the same utterances (M = 32.41, Std.Error = 1.25), [t = 5.887, p = 0.001]. However, no significant difference was found on the same case for the listeners' gaze [t = -0.767, p = 0.444].

6.2. Eye gaze before referring expressions

Previous studies have revealed that speakers typically look at referent objects about 800-1000ms before mentioning them (Staudte and Crocker, 2011). We therefore looked at 1 second before the utterance for all utterances (N = 582) and the gaze proportions around the area to the object(s) that were about the be referred to. We compared the speaker's gaze to the referent objects to the listeners' gaze using paired sample t-tests. The speaker's gaze (M = 49.72, Std.Error = 1.52) was different than the listener's gaze (M = 28.53, Std.Error = 1.18), [t = 12.928, p = 0.001]. No significant difference was found on the mutual gaze during the time before the referring expression [t = -1.421, p = 0.156].

There was also a significant difference on the proportional gaze of the speaker to the referent object (M = 49.72, Std.Error = 1.52), towards gaze on other objects (M = 33.07, Std.Error = 1.37), [t = 6.154, p = 0.001].

Finally, we compared the proportional gaze of the speaker to the referent object during the utterance and in the 1 second period before the utterance, however there was no significant difference, [t = -1.854, p = 0.064]. There was a difference however, as expected, on the listeners gaze during (M = 39.45, Std.Error = 1.11) and before (M = 28.5309, Std.Error = 1.18) the utterance, [t = 9.745, p = 0.001].

6.3. Eye gaze timing

Further, we investigated when each interlocutor turned their gaze to the object that was referred to verbally. In 450 out of the 582 cases the speaker was already looking at the object in a time window of 1 second prior to their verbal reference to it. On average during the 1s window they turned their gaze to the object 0.796s before uttering the referring expression [t = 83.157, p = 0.001], which is supported by the literature.

At least one of the listeners looked at the referent objects during the speaker's utterance in 537 out of the 582 cases. On average they started looking at the proximity area of the

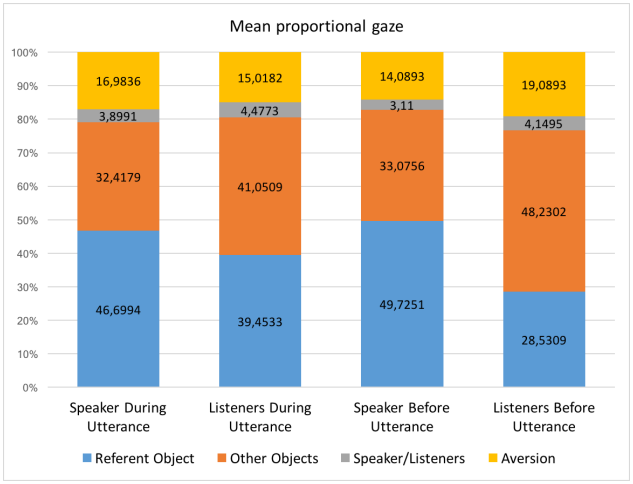


Figure 5: Mean proportional gaze per referring expression for speaker and listeners. a) The speaker's gaze during the time of the reference. b) The combined listeners' gaze during the speaker's referring expression. c) The speaker's gaze during the 1 second period before the reference. d) The combined listeners' gaze during the 1 second period before the speaker's referring expression.

referent object 344.5ms after the speaker started uttering a referring expression [t = 19.355, p = 0.001].

6.4. Mutual gaze and joint attention

We extracted all occasions where the group was jointly attending the area around the referent object, but also the occasions where either the speaker or the none of the listeners looked at the object: table 1.

N = 582	BU [-1..0]	DU [0..t]
None	58	21
Both	339	479
Speaker only	111	24
Listeners only	74	58

Table 1: Joint attention to the referent object area before (BU) and during (DU) speaker's references

During the preliminary analysis of the corpus we noticed that in many cases at least one of the listeners was already looking at the area around the referent object before the speaker would utter a referring expression (figure 5). It was our intuition that the salient object was already in the group's attention before. We looked at -1s before the utterance and automatically extracted these cases, as can be seen on table 2.

N = 582	BU [-1s]
None	163
Both	168
Speaker only	155
Listeners only	96

Table 2: Count of occasions where the referent object has already been in focus of attention (-1s before speaker's referring expression)

Finally we looked at the occasions where the interlocutors established mutual gaze. The table below shows cases where the speaker looked at one of the listeners or where at least one of the listeners looked at the speaker during and before referring expressions. Mutual gaze indicates where the speaker and one of the listeners look at each other, and as can be seen this is very rare during or before referring expressions.

N = 582	BU [-1..0]	DU [0..t]
Mutual gaze	10	20
Speaker at Listeners	52	72
Listeners at Speaker	80	143

Table 3: Count of occasions where the speaker looked at the listeners, the listeners looked at the speaker and mutual gaze during and before the speaker's referring expressions

7. Discussion

The presented corpus contains recordings of a variety of multimodal data, is processed and annotated and provides researchers with the possibility to explore multi-modal, multi-party turn-taking behaviours in situated interaction. While its strength lies in the rich annotation and the variation in conversational dynamics it also has some limitations.

One of the limitations is the granularity of eye-gaze annotation. Even though a high-end eye-gaze tracking system was used it was not always possible to disambiguate which of the potential objects were being gazed at. Given a visual angle threshold we are provided with a certain confidence measure defined by the angle difference to each object towards identifying the objects of attention. That limited our analysis to the area around the object rather than a precise measure of the object itself. Another limitation is that we did not analyse the object movements or the participants' pointing gestures. These might explain some of the visual attention cases prior to the verbal referring expressions.

Moreover, the use of the eye tracking glasses and motion capture equipment had the disadvantage that they were quite intrusive. Participants complained about fatigue at the end of the interactions and it was also qualitatively observed that their gaze-head movement coordination changed once wearing the glasses.

In most occasions the listeners and the speakers were looking at the area of the referent object before the referring expression was uttered which could potentially mean that the object was already on the group's attention. In some cases the listeners' visual attention was brought to the object area by referring language. Similarly to the referred literature this potentially shows that gaze has indications on the salient objects in a group's discussion and that can be used for reference resolution. It is also our intuition that objects that establish higher fixation density during referring expressions are considered to be more salient and can potentially resolve the references.

There are a few cases where neither the speaker nor the listener looked at the referent objects; in such cases text saliency algorithms (Evangelopoulos et al., 2009) or other

multimodal cues such as pointing gestures (Lücking et al., 2015) could be combined to resolve the reference to the salient objects.

Typically speakers direct the listeners' attention using verbal and non-verbal cues and listeners often read the speaker's visual attention during referring expressions to get indication on the referent objects. As in literature we found that speakers and listeners gazed at each other a lot during the references to establish grounding on the referent objects. In very few cases however, they also established mutual gaze (looking at each other at the same time) during those references.

8. Conclusions

The current paper presents a corpus of multi-party situated interaction. It is fully transcribed and automatically annotated for eye-gaze, gestures and spoken language. Moreover, it features an automatic eye-gaze annotation method where the participant's gaze is resolved in 3d space; a visualisation tool is also used to qualitatively examine parts or the whole corpus in terms of conversational dynamics, turn taking and reference resolution. We annotated object references and investigated the proportional gaze from both the perspective of the speaker and the listeners. Finally, we quantitatively described the data of the corpus and gave further indications on how the corpus can be useful by the research community. Both the annotated corpus and visualisations are available at: `https://www.kth.se/profile/diko/page/material`.

9. Acknowledgements

The authors would like to thank Joseph Mendelson for moderating the sessions and helping with the recordings of this corpus. We would also like to acknowledge the support from the EU Horizon 2020 project BabyRobot (687831) and the Swedish Foundation for Strategic Research project FACT (GMT14-0082).

10. Bibliographical References

Admoni, H. and Scassellati, B. (2017). Social eye gaze in human-robot interaction: a review. *Journal of Human-Robot Interaction*, 6(1):25–63.

Bohus, D. and Horvitz, E. (2009). Models for multiparty engagement in open-world dialog. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 225–234. Association for Computational Linguistics.

Bohus, D. and Horvitz, E. (2010). Facilitating multiparty dialog with gaze, gesture, and speech. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, page 5. ACM.

Bolt, R. A. (1980). *'Put-that-there': Voice and gesture at the graphics interface*, volume 14. ACM.

Borji, A. and Itti, L. (2013). State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207.

Bruce, N. D. and Tsotsos, J. K. (2009). Saliency, attention, and visual search: An information theoretic approach. *Journal of vision*, 9(3):5–5.

Carletta, J. (2007). Unleashing the killer corpus: experiences in creating the multi-everything ami meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

Evangelopoulos, G., Zlatintsi, A., Skoumas, G., Rapantzikos, K., Potamianos, A., Maragos, P., and Avrithis, Y. (2009). Video event detection and summarization using audio, visual and text saliency. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 3553–3556. IEEE.

Georgeton, L. and Meunier, C. (2015). Spontaneous speech production by dysarthric and healthy speakers: Temporal organisation and speaking rate. In *18th International Congress of Phonetic Sciences, Glasgow, UK, submitted*.

Goldberg, L. R. (1992). The development of markers for the big-five factor structure. *Psychological assessment*, 4(1):26.

Gross, S., Krenn, B., and Scheutz, M. (2017). The reliability of non-verbal cues for situated reference resolution and their interplay with language: implications for human robot interaction. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 189–196. ACM.

Grosz, B. J. and Sidner, C. L. (1986). Attention, intentions, and the structure of discourse. *Computational linguistics*, 12(3):175–204.

Hung, H. and Chittaranjan, G. (2010). The idiap wolf corpus: exploring group behaviour in a competitive role-playing game. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 879–882. ACM.

Kennington, C., Kousidis, S., and Schlangen, D. (2013). Interpreting situated dialogue utterances: an update model that uses speech, gaze, and gesture information. *Proceedings of SIGdial 2013*.

Lücking, A., Bergmann, K., Hahn, F., Kopp, S., and Rieser, H. (2010). The bielefeld speech and gesture alignment corpus (saga). In *LREC 2010 workshop: Multimodal corpora–advances in capturing, coding and analyzing multimodality*.

Lücking, A., Pfeiffer, T., and Rieser, H. (2015). Pointing and reference reconsidered. *Journal of Pragmatics*, 77:56–79.

Meena, R., Skantze, G., and Gustafson, J. (2014). Data-driven models for timing feedback responses in a map task dialogue system. *Computer Speech & Language*, 28(4):903–922.

Mehlmann, G., Häring, M., Janowski, K., Baur, T., Gebhard, P., and André, E. (2014). Exploring a model of gaze for grounding in multimodal hri. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 247–254. ACM.

Mostefa, D., Moreau, N., Choukri, K., Potamianos, G., Chu, S. M., Tyagi, A., Casas, J. R., Turmo, J., Cristoforetti, L., Tobia, F., et al. (2007). The chil audiovisual cor-

pus for lecture and meeting analysis inside smart rooms. *Language Resources and Evaluation*, 41(3-4):389–407.

Oertel, C., Cummins, F., Edlund, J., Wagner, P., and Campbell, N. (2013). D64: A corpus of richly recorded conversational interaction. *Journal on Multimodal User Interfaces*, 7(1-2):19–28.

Oertel, C., Funes Mora, K. A., Sheikhi, S., Odobez, J.-M., and Gustafson, J. (2014). Who will get the grant?: A multimodal corpus for the analysis of conversational behaviours in group interviews. In *Proceedings of the 2014 Workshop on Understanding and Modeling Multiparty, Multimodal Interactions*, pages 27–32. ACM.

Prasov, Z. and Chai, J. Y. (2008). What's in a gaze?: the role of eye-gaze in reference resolution in multimodal conversational interfaces. In *Proceedings of the 13th international conference on Intelligent user interfaces*, pages 20–29. ACM.

Prasov, Z. and Chai, J. Y. (2010). Fusing eye gaze with speech recognition hypotheses to resolve exophoric references in situated dialogue. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 471–481. Association for Computational Linguistics.

Rayner, K. (1995). Eye movements and cognitive processes in reading, visual search, and scene perception. In *Studies in visual information processing*, volume 6, pages 3–22. Elsevier.

Schlangen, D. and Skantze, G. (2009). A general, abstract model of incremental dialogue processing. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 710–718. Association for Computational Linguistics.

Sheikhi, S. and Odobez, J.-M. (2012). Recognizing the visual focus of attention for human robot interaction. In *International Workshop on Human Behavior Understanding*, pages 99–112. Springer.

Skantze, G., Johansson, M., and Beskow, J. (2015). Exploring turn-taking cues in multi-party human-robot discussions about objects. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 67–74. ACM.

Staudte, M. and Crocker, M. W. (2011). Investigating joint attention mechanisms through spoken human–robot interaction. *Cognition*, 120(2):268–291.

Stefanov, K. and Beskow, J. (2016). A multi-party multimodal dataset for focus of visual attention in human-human and human-robot interaction. In *Proceedings of the 10th edition of the Language Resources and Evaluation Conference (LREC 2016, 23-28 of May.* ELRA.

Sziklai, G. (1956). Some studies in the speed of visual perception. *IRE Transactions on Information Theory*, 2(3):125–128.

Winograd, T. (1972). Understanding natural language. *Cognitive psychology*, 3(1):1–191.

Improving Dialogue Act Classification for Spontaneous Arabic Speech and Instant Messages at Utterance Level

AbdelRahim A. Elmadany[1], Sherif M. Abdou[2], Mervat Gheith[3]

[1]Department of Computer Science, Deanship of Community Service and Continuing Education, Jazan University, KSA
[2]Department of Information Technology, Faculty of Computers and Information, Cairo University, Egypt
[3]Department of Computer Science, Institute of Statistical Studies and Research (ISSR), Cairo University, Egypt
aelmadany@jazanu.edu.sa, sh.ma.abdou@gmail.com, mervat_gheith@yahoo.com

Abstract

The ability to model and automatically detect dialogue act is an important step toward understanding spontaneous speech and Instant Messages. However, it has been difficult to infer a dialogue act from a surface utterance because it highly depends on the context of the utterance and speaker linguistic knowledge; especially in Arabic dialects. This paper proposes a statistical dialogue analysis model to recognize utterance's dialogue acts using a multi-classes hierarchical structure. The model can automatically acquire probabilistic discourse knowledge from a dialogue corpus were collected and annotated manually from multi-genre Egyptian call-centers. Extensive experiments were conducted using Support Vector Machines classifier to evaluate the system performance. The results attained in the term of average F-measure scores of 0.912; showed that the proposed approach has moderately improved F-measure by approximately 20%.

Keywords: Dialogues Language Understanding, Dialogue Acts Classification, Spoken Dialogues, Instant Messages, Natural Language Understanding

1. Introduction

The most important and difficult part in human-computer interaction system "i.e. Dialogue System" is understanding what the user needs? This task is called "language understating component" or somewhere "Dialogue Acts (DAs) classification." DAs classification task is labeling the speaker's intention in producing a particular utterance with short words; the DAs terminology approximately is the equivalent of the speech act of Searle (1969), and DAs is different based on dialogue systems domains (Elmadany *et al.*, 2015b). Since 1999, the research in DAs area has increased after spoken dialogue systems been a commercial reality. Hence, the development of dialogue systems has focused on some of the conversational roles such acts which can perform because it is closely linked to the field of computational linguistics (Stolcke *et al.*, 2000). DAs is used practically in many live dialogue systems such as Airline Travel Information Systems such as ATIS (Seneff *et al.*, 1991), DARPA (Pellom *et al.*, 2001), and VERBMOBIL (Wahlster, 2000).

Recently, the development of dialogue interaction systems has gained considerable attention, but most of the resources and systems are built so far tailored to English and other Indo-European languages. The development of the dialogue systems for other languages as Arabic is required. So, the Arabic dialogue acts classification's task has gained focus because it is a key player in Arabic language understanding to building these systems. The motivation of this paper comes from the point of view "building automatic language understanding component for Egyptian dialect dialogues".

The paper focuses on inquiry–answer dialogues from the call-centers domain because it receives or transmits a large volume of information inquiries from customers. In this research, we have selected Customer-Service entities from Banks, Flights, and Mobile Networks Operators call-centers.

In this paper, we are referring to an *utterance* as a small unit of speech that corresponded to a single act (Webb, 2010; Traum and Heeman, 1997). In speech research community, *utterance* definition is a slightly different; it refers to a complete unit of speech bounded by the speaker's silence while we refer to the complete unit of speech as *a turn*. Thus, a single *turn* can be composed of many *utterances*. *Turn* and *utterance* can be the same definition when the *turn* contained one utterance as defined and used in (Graja *et al.*, 2013).

To develop a language understanding model for either spoken dialogue or instant messages, there are four major issues are required:

- Dialogue acts schema.
- Annotated corpora with the dialogue act schema
- Turn segmentation into utterances classifier
- Utterance labeling classifier (i.e. dialogue act classifier)

The annotated Egyptian dialect dialogues corpus were built utilizing manually collected data from Egyptian call-centers (Elmadany *et al.*, 2014, 2015a). During annotation process, it is being noted that; the Egyptian turns are almost long and contains many utterances as noticed during data collection. Consequently, turn segmentation into utterances for Egyptian Arabic dialogues model namely 'USeg' (Elmadany *et al.*, 2015d) has been built, which a machine learning approach based on context without relying on punctuation, text diacritization or lexical cues. Finally, (Elmadany *et al.*, 2015c) have been proposed a dialogue act classifier based on chunking concepts and depending on a set of sentential and contextual features. The sentential features contain four features: Utterance-Words, Words Part-Of-Speech (POS) Tags, Speaker Name, and Utterance start a label. The contextual features contain only one feature: the previous utterance act.

In this paper, we improved (Elmadany *et al.*, 2015c) dialogue act classifier. We proposed an utterances labeling with suitable act model for Egyptian dialect inquiry-answer dialogues using multi-classes hierarchical structure. The classification model has been built using two-layer hierarchical structure. In the first layer, each utterance is classified into one of six categories: Dialogue Structure, Social Obligation, Question, Answer, Social Courtesy, or

Other. In the second layer, each utterance has been classified as individual acts based on their class 'i.e. category' which is determined in the first layer. The proposed model depends on a set of sentential and contextual features. To train and evaluate the proposed model, a corpus that contains spoken dialogues and Instant Messages (IM) for Egyptian Arabic has been used; and the model results are compared with manually annotated utterances by experts.

This paper presents three major contributions. First, the selected features and hierarchal structure has moderately improved the dialogue acts classification in the term of the average F-measure approximately 20%. Second, the proposed approach does not rely on a number of classes as used in binary classification, instead; it uses only two models (one for each layer). Third, the proposed method is suitable for working on Egyptian dialect either spontaneous speech dialogue or instant messages.

This paper is organized as follows: section 2 presents a literature review of acts classification, section 3 presents s the proposed classification model, section 4 describes the dataset, experimental setup, and experiments results, and finally, the conclusion and feature works are reported in section 5.

2. Literature Review of Acts Classification

The initial state of speech act classification has been addressed by Searle (1969) based on Austin (1962) work as a fundamental concept of linguistic pragmatics, analyzing, for example, what it means to ask a question or make a statement. Although major dialogue theories treat Dialogue acts as a central notion, the conceptual granularity of the Dialogue act labels used varies considerably among alternative analyses, depending on the application or domain (Webb and Hardy, 2005). Within the field of computational linguistics, recent work, closely linked to the development and deployment of spoken language dialogue systems, has focused on some of the conversational roles such acts can perform. Therefore, Dialogue act recognition is considered an important component of most spoken dialogue systems.

Many statistical models have been applied to dialogue acts classification. N-gram models can be considered the simplest method of DA classification based on some limited sequence of previous DAs as in (Hardy *et al.*, 2004; Webb, 2010; Webb and Hardy, 2005; Webb *et al.*, 2005) and sometimes used with Hidden Markova Model (HMM) as in (Boyer *et al.*, 2010). In addition, there are other approaches such as Transformation-Based Learning (TBL) as in (Samuel *et al.*, 1998), and Naïve Bayesian as in (Grau *et al.*, 2004).

Most of the previous researchers on dialogue acts classification addressed two types of feature: (1) Sentential features reflecting the linguistic characteristics of the surface utterance, which are extracted by a linguistic analyzer, such as a morphological analyzer, syntactic parser or semantic analyzer. (2) Contextual features reflecting the relationship between the current utterance and the previous utterance. In an actual dialogue, a speaker can express an identical meaning using different surface utterances based on the speaker's personal linguistic background. For this reason, it is impossible to directly compute the sentential probability because sentences are too various to find identical surface forms. To overcome this problem, researchers assume that a syntactic pattern generalizes these surface utterances using syntactic features to represent the sentential features such as sentence type, main verbs, auxiliary verbs and clue words (Choi *et al.*, 2005).

Kang et al (Kang *et al.*, 2013) proposed a model for classification speech acts for Koran language based on two-layer hierarchal structure using binary Support Vector Machine (SVM) classifiers They used the sentential features that are composed of words annotated with POS tags and POS bi-grams of all the words in an utterance and used the speech act of the previous utterance only as a contextual feature.

The proposed approach is mainly different from a Kang et al approach in three aspects - the architecture of the hierarchal structure and the selected feature set. First, the number of SVM models within their architecture requires more processing. The Authors' approach is mainly constructed using 19 SVMs models; the tested utterances are passed through the classifiers of the first layer (3 SVMs classifiers), and finally classified into one speech act among the speech acts included in the assigned type by the classifiers of the second layer (6 SVMs for Question, 7 SVMs for Response, and 3 SVMs for Other).

The proposed approach is mainly constructed using two models, one for each layer. Therefore, we think that our approach model is faster than binary classification and it can be more an efficient dialogue act classification model in real-time systems. The second difference is the number of models in multi-classes classification "our approach" not affected with a number of dialogue acts or classes but models numbers are affected when used binary classification as in the authors' approach. The third difference has they used a limited feature set that might be suitable for Koran whereas there are many features that can be used such as the relation between the speaker's dialogue act and the utterance surface, while our feature set includes rich features consisting of sentential and contextual features. For instance, speaker name, the number of utterance words, previous category, previous speaker… etc.

In fact, there are very few efforts have addressed dialogue acts classification for Arabic. (Shala *et al.*, 2010) used Naïve Bayes and Decision Trees. (Bahou *et al.*, 2008) used utterances semantic labeling based on the frame grammar formalism. (Lhioui *et al.*, 2013) used syntactic parser context-free grammar with HHM. (Graja *et al.*, 2013) used Conditional Random Fields (CRF) to semantically label spoken Tunisian dialect turns. (Hijjawi *et al.*, 2013; Hijjawi *et al.*, 2014)used Arabic function words such as "هل" "do/does", "كيف" "How" to classify questions and non-questions utterances with Decision Tree Classifier.

The proposed approach is mainly different from the previously mentioned approaches in three aspects. First, these approaches not used the hierarchal structure to solve the classification problem. Second, we provide a feature set which differed from the feature set in these approaches. We used rich features consisting of sentential and contextual features such as speaker name, the number of utterance words, previous category, previous speaker… etc. The third difference is these approaches were designed and applied on MSA or Tunisian dialect which fully differed from Egyptian dialect.

To the best of our knowledge, there is only one published work for understanding Egyptian Arabic or Egyptian

dialect proposed by (Elmadany *et al.*, 2015c). They have presented a dialogue act classifier based on chunking concepts and depending on a set of sentential and contextual features. The sentential features contain four features: Utterance-Words, Words Part-Of-Speech (POS) Tags, Speaker Name, and Utterance start a label. The contextual features contain only one feature: the previous utterance act.

3. Utterance Labelling Model

All Let $U_{1,n}$ denote a dialogue which consists of a sequence of n utterances, U_1, U_2 …U_n, and let $DA_{1,n}$ denote the dialogue act sequences of $U_{1,n}$. Then, the dialogue act of current *utterance* can be formally defined as:

$$DA(U_i) \approx argmax_{S_{i,j}} P(CA_{i,j}|SF_i)P(CA_{i,j}|DA(U_{i-1}))$$

$DA(U_i)$ denotes the dialogue act of the i[th] utterance (U_i) and $CA_{i,j}$ denotes jth candidate dialogue act of the ith utterance (U_i), given a dialogue including n utterances ($U_{1,n}$). Therefore, we assume that the current dialogue act $DA(U_i)$ is dependent on the sentential features set (SF_i) of current utterance (U_i) and the dialogue act $DA(U_{i-1})$ of the previous utterance (U_{i-1}) (Choi *et al.*, 2005).

Using the utterances meta information can help dialogue acts classification process (Kim *et al.*, 2010; Ivanovic, 2005, 2008) and know what happened before current utterance can help the classification task (Sridhara *et al.*, 2009; Eugenio *et al.*, 2010). Moreover, there is a strong relationship between the speaker's dialogue act and the surface *utterances* expressing that dialogue act (Andernach, 1996; Kang *et al.*, 2013; Choi *et al.*, 2005). For instance, the speaker utters a sentence, which most well expresses his/her intention (act) so that the hearer can easily understand what the speaker's dialogue act is. On the other hand, the speaker type Operator or Customer of the current utterance can help to determine the act of utterance. For instance, the act "Service-Question" is related to the customer because he connected to service support service to asking about a provided service, but the act "Other-Question" and "Choice-Question" are related to operator because the operator asking the client about his name or choosing the client to select one of the provided services. Therefore, the sentential features represent the relationship between the dialogue acts and the surface *utterances*. In a real dialogue, the speaker utters identical contents with various surface utterances according to his personal linguistic knowledge. In addition, knowing the previous utterances acts sequence in the dialogue help the classifier to predict the dialogue act of current utterance. For instance, the act "Agree" and "Disagree" is almost followed by the "Confirm-Question" act.

The first layer of the proposed model depends on seven sentential features: Utterance-Words, Utterance-length, POS, First-Verb, Is-Part-Of-Turn, Speaker Name, and Cues; and two contextual features: speaker name of the previous utterance, and dialogue act of the previous utterance.

The second layer of the proposed model depends on eight sentential features are used: the seven sentential features as the first layer plus the main category of current utterance. So, the sentential features are Utterance-Words, Utterance-length, POS, First-Verb, Is-Part-Of-Turn, Speaker Name, Cues, and the main category of current utterance. Also, it depends on three contextual features: contextual features as the first layer plus the previous main category. So, contextual features are the main category of previous utterance, speaker name of the previous utterance, and dialogue act of the previous utterance. Table 1 shows the used sentential features in the two layers with their possible values.

Sentential Features	Values
Utterance-Words	Uni-grams and bi-grams of utterance words
Utterance-Length	Number of utterance words
POS	Sequence of words Part-Of-Speech tags
First-Verb	One of four types: active (a), passive (p), not applicable (na), and undefined (u)
Is-Part-Of-Turn	Yes, or No
Speaker, Previous Speaker	Operator or Customer
Cue-Word and Cue-Phrase	yes, no, ok, Thank you, etc. (total of 241)
− The main category of current utterance − The main category of the previous utterance	One of six main categories: Dialogue Structure, Turn Management, Social Obligation, Question, Answer, Social Courtesy, or Argumentation

Table 1. Sentential Features

In the first layer, the class feature is excluded from sentential features because that is what a need to classify for is. In the second layer, the predicted class will add to sentential features. On another hand, the output of the training phase (i.e. the classification model) is used in the prediction phase to generate the final utterance act classification. In this study, WEKA[1] (Hall *et al.*, 2009), a comprehensive workbench with support for a large number of machine learning algorithms, is utilized as the development environment of the machine learning based component. The SVM algorithm is applied using SMO.

4. Empirical Evaluation

4.1 Dialogues Corpus for Egyptian Dialect

We used a corpus of real spoken dialogue in the Egyptian dialect which used in (Elmadany *et al.*, 2015c), this corpus is called JANA. JANA is a multi-genre corpus of Arabic dialogues labeled for Arabic Dialogues Language Understanding (ADLU) at utterance level and comprising Spontaneous Speech Dialogues (SSD) and Instance Messages (IM) for Egyptian dialect (Elmadany et al., 2016).

SSD has been recorded since August 2013, and it contains 52 phone calls recorded from Egyptian's banks and Egypt Air Company call-centers with an average duration of two hours of talking time after removing ads from calls. It

[1] http://www.cs.waikato.ac.nz/ml/weka/

consists of human-human discussions about providing services e.g. Create new bank account, service request, balance check and flight reservation. IM dialogues contain 30 chat dialogues, collected from mobile network operator's online-support. JANA consists of approximately 3001 turns with average 6.7 words per turn, containing 4725 utterances with average 4.3 words per utterance, and 20311 words.

4.2 Experimental Results

Experimental results are presented across five datasets: Banks dataset, Flights dataset, IM dialogues dataset, combined spoken dataset (Banks and Flights), and combined dataset (Banks, Fights, and IM). Three different functions (or classifiers) are applied separately to each dataset, including SVM classifier which is supported in WEKA toolkit via SMO and built-in classifier.

Our preliminary experimental results showed that one-vs-one approach achieves the best performance in this task. Therefore, we used one-*vs*-one classification approach and the predicted probabilities are coupled using Hastie and Tibshirani's pairwise coupling method (Hastie and Tibshirani, 1998).

In this study, the evaluation is conducted based on a 10-fold cross-validation method to avoid over-fitting in which the available data set is divided into 10 folds and for each fold, a classifier is induced. The classifier is derived from 9 folds and tested on the remaining fold. The WEKA tool provides the functionality of applying the conventional k-fold cross-validation for evaluation with each classifier and then having the results represented in the aforementioned standard measures.

The first layer is classified the main category of current utterance. In the second layer, we added the classified main categories of the current and previous utterances to feature set for recognizing the dialogue act of current utterance. To test the performance of hierarchical structure in dialogue act classification and due to the lack of published works in dialect acts classification on Egyptian dialect over spontaneous dialogues either spoken or instant messages. Table 2 are illustrated the results of the proposed systems performances in terms of average F-measure when applied on Bank, Flights, IM, Combined Spoken (Banks, Flights), Combined (Banks, Flights, IM) Datasets.

Banks	Flights	IM	Spoken	Combined Dataset
0.913	0.902	0.909	0.909	0.912

Table 2. The results of applying our system on Bank, Flights, IM, Combined Spoken, Combined Datasets

According to the empirical results illustrated in Table 2, the overall experimental results show that the spoken dialogues highest performance than instant messages dialogues over all classifiers, and the results are much closed when applied our system using the three classifiers.

So, the results show the highest performance in acts such as Turn-Assign, Agree, SelfIntroduce, Greeting, Service-Answer, and Inform. The results show very good performance in acts such as Disagree, Service-Question, and Confirm-Question. The results show good results in acts such as Suggest and low performance in Promise, Offer, and Correct acts. The low performance due to the low counts or not exist in the training for these acts. For instance, acts 'Closing', 'Promise', and 'Offer' is not existence (i.e. N/A) in collected IM dialogues and Promise, Offer, and Correct acts are rarely existing.

In the hierarchal method, if the first layer would incorrectly classify the main category, the second layer will be classified incorrectly. For example, if the first layer is classified the main category of the current utterance as "Social Obligation", then in the second layer must choose one of four acts "Apology, Greeting, SelfIntroduce, Thanking". To solve these problems, we used the results of the first layer "main category of the current utterance and main category of the previous utterance" as features in the second layer to choose dialogue act from the 26 acts.

In Arabic dialect, especially in Egyptian Arabic, there are some words/phrases can be used in many situations with a different meaning. For example, if the operator asks " أي استفسار تاني حضرتك؟" "any other service sir?", The customer can answer "شكرا". The word "شكرا" here means "there is no other service is need" and that refers to "NO" disagree act but actually the word "شكرا" refers to thanks but here based on the dialogue it refers to NO. Also, the word "نعم" refers to "YES" agree act but sometimes used as misunderstanding sign. So, used features have solved these problems. So, the experimental results show that our system overcomes the ambiguation problem due to using the dialogue structure features such as previous act, speaker, and main category. The proposed system gives 0.909 for 'Thanking' acts, and 0.876 for 'Disagree' act in the term of F-measure. The most failure of our system due to either the rare existence or low counts of some acts in the training datasets, or there are some utterances needs to deeply semantic analysis. For instance, if the operator's utterance such as "ولكن طبعا لازم يكون عدي عليها 6 شهور" (Make sure you must get it since 6 months) and the customer has responded such as "لا لا هي عدي عليها 4 سنين" (No No it since 4 years). The system classifies the customer utterance as 'Disagree' act because it contains "لا" (No) in spite of the customer has agreed on the operator warning.

Two-Layer Hierarchical Structure based on Binary Classification	Two-Layer Hierarchical Structure based on Multi-Classification	
43.28 second	One-vs-One	One-vs-All
	19.89 second	22.94 second

Table 3. The comparison between Two-Layer Hierarchical Structure based on Binary Classification and Multi-Classification when our system (using SMO classifier) is applied on Combined Datasets

(Kang *et al.*, 2013) has been approved using a two-layer hierarchical structure based on binary classification to solve dialogue act classification is much faster than binary classification and reported it needs only about 40%of running time of the binary classification model. The experiments results verify that the running time of two-layer hierarchical structure based on multi-classification in

the training phase is much faster than a two-layer hierarchical structure based on binary classification.

Table 3 shows the comparison between Two-Layer Hierarchical Structure based on Binary Classification and Multi-Classification when our system (using SMO classifier) is applied on Combined Datasets.

Training Models	Test datasets (Macro F-Measure)		
	Banks	Flights	IM
Banks Dataset	--	0.855	0.786
Flight Dataset	0.857	--	0.782
IM Dataset	0.762	0.778	--
All Datasets (70% train, 30% test)	0.891	0.864	0.864

Table 4. The comparison results of applying proposed model on Flights and IM datasets, and Banks when using each dataset as training model

Finally, to test the generality of the proposed model on inquiry-answer domains, we trained the system using a corpus from one domain and tested the system using a corpus from a different domain. Table 4 shows the results of applying our system in the term of Macro F-measure on Flights and IM datasets, and Banks dataset when used each dataset as a training model. The results achieved the highest performance when Banks dataset has used to train the system and testing the system using others datasets 'unseen' (Flights and IM datasets). It is worth noting that we can achieve surprisingly good classification accuracy using this method.

To compare the results obtained using the proposed model with others, previous speech act analysis models in Arabic dialogues. Table 5 shows these others, previous models of different types, and their performance. We report the performance of each model as they reported and an evaluation metric that is used in their papers. So, we notice that using a hierarchical structure in dialogue acts classification has proved it's comparatively higher efficiency and improved the previous system (Elmadany *et al.*, 2015c) results in more than 20% in the term of F-measure using same experimental setup and data.

Classification model	**Data Type**	**Feature set**	**Measurement**	**Score**
(Bahou et al., 2008)	– Speech – Tunisian national railway – MSA	– Normalization – Morphological analysis – Semantic Analysis – Lexical – Semantic frames of the utterance.	F-Measure	0.7179
(Shala *et al.*, 2010)	– Speech – Newspaper & TV – MSA	– Initial words in the utterance – Parts-of-Speech – Named Entity Recognition – SVM, NB & J48	F-Measure	0.4173
(Lhioui *et al.*, 2013)	– Speech – Tunisian Dialect	– Context-free grammar augmented with probabilities associated with rules – HMM for creating the stochastic model	F-Measure	0.7379
(Graja *et al.*, 2013)	– Speech – Tunisian national railway – Tunisian Dialect	– Conditional Random Fields (CRF) – lexical normalization – Morphological analysis and lemmatization – Annotate word by word	F-Measure	0.8652
(Hijjawi *et al.*, 2013) (Hijjawi *et al.*, 2014)	– Instant Messages – MSA	– Arabic function words – focused on classifying questions and non-questions utterances – NB & Decision Tree	Accuracy	0.8741
(Neifar *et al.*, 2014)	– Speech – Tunisian national railway – Tunisian Dialect	– Based on (Bahou *et al.*, 2008) – Lexical database – Conceptual segmentation	F-Measure	Dataset A = 0.7322 Dataset B = 0.9298
(Dbabis *et al.*, 2015)	– Speech – TV Programs – Dialect	– Lexical – Morphological – Discursive and structural features – SVM, NB, and J48	F-Measure	0.522
(Graja *et al.*, 2015)	– Speech – Tunisian national railway – Tunisian Dialect	– improved their previous model (Graja *et al.*, 2013) – Adding a new lexicon of the domain (Railways inquiry domain-based ontology).	F-Measure	0.8845
(Elmadany *et al.*, 2015c)	**– Egyptian Dialect Dialogues (JANA corpus)**	**– Chunking concepts - Utterance-Words – Words - Part-Of-Speech (POS) Tags - Speaker Name – Utterance start a label -Previous utterance act**	**F-Measure**	**0.7036**

Table 5. Performance of the proposed model and other previous models

5. Conclusion

This paper proposes an effective dialogue acts classification model using a multi-classes hierarchical model based on the two-layer hierarchical structure for an understanding of the Arabic dialogues task for Egyptian dialect at the utterance level. The proposed classifier has been tested using a corpus consisting of spontaneous speech dialogues and IM for Egyptian dialect, and the obtained results are very promising. In the future work, a plan is recommended to improve the classifier by adding general cues for the call-centers domain, morphological features, and dialect words treatments. Moreover, we would like to enrich the corpus with inquiry-answer dialogues from other domains e.g. Online Markets, and Railway Networks to cover 1000 Arabic dialogues.

6. References

Andernach, T. 1996. *A machine learning approach to the classification of dialogue utterances*. In Proceedings of NeMLaP-2.98–109.

Bahou, Y., *et al.* 2008. *Towards a Human-Machine Spoken Dialogue in Arabic*. In Proceedings of Workshop on HLT & NLP within the Arabic world: Arabic Language and local languages processing: Status Updates and Prospects, at the 6th Language Resources and Evaluation Conference (LREC'08). Marrakech, Maroc.

Boyer, K., *et al.* 2010. *Dialogue Act Modleing in a Complex Task-Orinted Domain*. In Proceedings of SIGDIAL. Tokyo, Japan:297-305.

Choi, W. S., *et al.* 2005. An Integrated Dialogue Analysis Model for Determining Speech Acts and Discourse Structures. *IEICE TRANS. INF. & SYST.*, E88–D(1):150-157.

Dbabis, S. B., *et al.* 2015. *Automatic dialogue acts Annotation within Arabic debates*.16th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2015). Cairo, Egypt.

Elmadany, A., *et al.* 2016. JANA: A Human-Human Dialogues Corpus for Egyptian Dialect (LDC2016T24) Linguistic Data Consortium. Philadelphia.

Elmadany, A. A., *et al.* 2014. Arabic Inquiry-Answer Dialogue Acts Annotation Schema. *IOSR Journal of Engineering (IOSRJEN)*, 04(12-V2):32-36.

Elmadany, A. A., *et al.* 2015a. *JANA: An Arabic Human-Human Dialogues Corpus*.Recent Trends in Information Systems (ReTIS), 2015 IEEE 2nd International Conference on. Jadavpur University, Kolkata, India:347-352.

Elmadany, A. A., *et al.* 2015b. A Survey of Arabic Dialogues Understanding for Spontaneous Dialogues and Instant Message. *International Journal on Natural Language Computing (IJNLC)*, 4(2):75-94.

Elmadany, A. A., *et al.* 2015c. Towards Understanding Egyptian Arabic Dialogues. *International Journal of Computer Applications*, 120(22):7-12.

Elmadany, A. A., *et al.* 2015d. Turn Segmentation into Utterances for Arabic Spontaneous Dialogues and Instant Messages. *International Journal on Natural Language Computing (IJNLC)*, 4(2):111-123.

Eugenio, B. D., *et al.* 2010. Dialogue act classification, higher order dialogue structure, and instance-based learning. *Dialogue and Discourse*, 1(2):1-24.

Graja, M., *et al.* 2013. *Discriminative Framework for Spoken Tunisian Dialect Understanding*.2nd International Conference on Statistical Language and Speech Processing, SLSP 2014.102–110.

Graja, M., *et al.* 2015. Statistical Framework with Knowledge Base Integration for Robust Speech Understanding of the Tunisian Dialect. *IEEE/ACM Transactions On Audio, Speech, And Language Processing*, 23(12).

Grau, S., *et al.* 2004. *Dialogue Act Classification using a Bayesian Approach*. In Proceedings of 9th Conference Speech and Computer.

Hall, M., *et al.* 2009. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1).

Hardy, H., *et al.* 2004. *Data-driven strategies for an automated dialogue system*. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics.

Hastie, T., and R. Tibshirani. 1998. Classification by pairwise coupling. *The Annals of Statistics*, 26(2):451-471.

Hijjawi, M., *et al.* 2013. *User's Utterance Classification Using Machine Learning for Arabic Conversational Agents*. In Proceedings of 5th International Conference on Computer Science and Information Technology (CSIT).223-232.

Hijjawi, M., *et al.* 2014. *ArabChat: an Arabic Conversational Agent*. In Proceedings of 6th International Conference on Computer Science and Information Technology (CSIT).227-237.

Ivanovic, E. 2005. *Automatic utterance segmentation in instant messaging dialogue*. In Proceedings of The Australasian Language Technology Workshop.241-249.

Ivanovic, E. 2008. Automatic instant messaging dialogue using statistical models and dialogue acts. Masters Research thesis. University of Melbourne.

Kang, S., *et al.* 2013. Hierarchical speech-act classificationfor discourse analysis. *Pattern Recognition Letters*, 34(10):1119–1124.

Kim, S. N., *et al.* 2010. *Classifying dialogue acts in one-on-one live chats*. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2010).862–871.

Lhioui, C., *et al.* 2013. *A Combined Method Based on Stochastic and Linguistic Paradigm for the Understanding of Arabic Spontaneous Utterances*. In Proceedings of CICLing 2013, Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science. Samos, Greece, 7817(2):549-558.

Neifar, W., *et al.* 2014. *Implementation of a Symbolic Method for the Tunisian Dialect Understanding*.

In Proceedings of 5th International Conference on Arabic Language Processing (CITALA'2014). Oujda, Maroc.

Pellom, B., *et al.* 2001. *University of Colorado Dialog Systems for Travel and Navigation*. In Proceedings of HLT '01: Proceedings of the First International Conference on Human Language Technology Research. USA.

Samuel, K., *et al.* 1998. *Dialogue act tagging with transformation-based learning*. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics.

Seneff, S., *et al.* 1991. *Interactive Problem Solving and Dialogue in the ATIS Domain*. In Proceedings of HLT '91: Proceedings of the Workshop on Speech and Natural Language. USA:354-359.

Shala, L., *et al.* 2010. *Automatic Speech Act Classification In Arabic*. In Proceedings of Subjetividad y Procesos Cognitivos Conference 14(2):284-292.

Sridhara, V. K. R., *et al.* 2009. Combining lexical, syntactic and prosodic cues for improved online dialog act tagging. *Computer Speech & Language*, 23(4):407–422.

Stolcke, A., *et al.* 2000. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics*, 26(3):339-373.

Traum, D., and P. A. Heeman. 1997. Utterance units in spoken dialogue. *Dialogue processing in spoken language systems* Springer. Berlin Heidelberg,125-140.

Wahlster, W. 2000. *Verbmobil: Foundations of Speech-To-Speech Translation* Springer.

Webb, N. 2010. Cue-Based Dialogue Act Classification. Ph.D. dissertation. University of Sheffield, England.

Webb, N., and H. Hardy. 2005. *Data-Driven Language Understanding for Spoken Language Dialogue* American Association for Artificial.

Webb, N., *et al.* 2005. *Dialogue Act Classification Based on Intra-Utterance Features*. In Proceedings of the AAAI Work-shop on Spoken Language Understanding.

Data Management Plan (DMP) for Language Data under the New General Data Protection Regulation (GDPR)

Pawel Kamocki, Valérie Mapelli, Khalid Choukri
ELDA
9 rue des Cordelières, 75013 Paris
{kamocki | mapelli | choukri} @elda.org

Abstract
The EU's General Data Protection Regulation (GDPR) of 27 April 2016 will apply from 25 May 2018. It will reinforce certain principles related to the processing of personal data, which will also affect many projects in the field of Natural Language Processing. Perhaps most importantly, the GDPR will introduce the principle of accountability, according to which the data processor shall be able to demonstrate compliance with the new rules, and that he applies 'privacy by design and by default'. In our opinion, a well-drafted Data Management Plan (DMP) is of key importance for GDPR compliance; indeed, the trend towards the adoption of a DMP, particularly in EU-funded research projects, has been more vivid since 2017, after the extension of the Horizon 2020 Open Data Pilot. Since 2015, ELRA also proposes its own template for the Data Management Plan, which is being updated to take the new law into account. In this paper, we present the new legal framework introduced by the GDPR and propose how the new rules can be integrated in the DMP in order to increase transparency of processing, facilitate demonstration of GDPR compliance and spread good practices within the community.

Keywords: data management plan, data protection, anonymisation, personal data

1. Introduction

In the years 2017-2018, the language resources community, especially in the European Union, has been confronted with certain important changes. First of all, the Regulation 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation, hereinafter: GDPR) will apply from 25 May 2018. This new legislative act will repeal the Directive 95/46/EC and unify data protection laws across the whole European Union. Unlike a directive, a regulation applies directly in all the Member States, it is therefore essential for researchers and companies established in the EU[1] and processing language data to comply with the GDPR.

Secondly, the trend towards the adoption of Data Management Plans has recently been more visible. This is partly due to the extension of the Open Research Data Pilot to all the thematic areas of the Horizon 2020 Programme (hereinafter: H2020). Indeed, from 2017 on, all the research data in the H2020-funded projects is open by default (with opt-outs still possible); these projects are therefore obliged to adopt a Data Management Plan (DMP) addressing the issues of Findability, Accessibility, Interoperability and Reusability (FAIR) of the data (see art. 29.3 of the H2020 Model Grant Agreement). A template for a DMP is available on the European Commission's web portal[2]. Of course, the importance of a DMP is not limited to H2020-funded projects — in fact, many other projects and institutions have adopted DMPs long before the extension of the Open Research Data Pilot. For example in the US, the National Science Foundation (NSF) requires all grant proposals to include a DMP 'of no more than two pages'.

Since 2015, ELRA has proposed a model for a DMP concerning specifically language resources (Choukri et al., 2016).

The purpose of the proposed paper is to draw the attention of the community on how certain issues related to the processing of personal data should be integrated in a DMP to ensure compliance with the GDPR. It is therefore useful to examine the general framework of the GDPR and then to see how the DMPs should be modified in order to adapt to the new law.

2. General Data Protection Regulation – a summary of the new framework

Most of the changes introduced by the GDPR are of evolutionary rather than revolutionary nature; the light has been shed on some grey areas, but the main principles remain largely the same (2.1). The most important addition from the point of view of the language community is probably the reinforcement of the obligations related to data processing (2.2).

2.1. General framework of the GDPR – what's old, what's new?

The notions of personal data and processing remain unchanged for the most part. Personal data are defined, just like in the Directive 95/46/EC, as *"any information relating to an identified or identifiable natural person"* (art. 4 no. 1 of the GDPR). According to the opinion of the Article 29 Data Protection Working Party (hereinafter: WP29)[3], the notion covers not only 'objective' information (i.e. facts), but also 'subjective' information (opinions and assessments). The information 'relates to a person' not only if it is 'about' a person (the 'content' element), but also if it is used to evaluate or influence the status or behaviour of the person (the 'purpose' element), or if it has an impact on

[1] …and not only; in fact, the GDPR applies also to entities established outside of the EU if they offer goods and services to physical persons in the Union, or if they monitor the behavior of physical persons on the Union's territory (art. 3 of the GDPR).

[2] http://ec.europa.eu/research/participants/data/ref/h2020/grants_manual/hi/oa_pilot/h2020-hi-oa-data-mgt_en.pdf

[3] WP29 is a working group made up of a representative of the data protection authority of each EU Member State, the European Data Protection Supervisor and the European Commission, created by art. 29 of the Directive 95/46/EC. Under the GDPR, it will be replaced by the European Data Protection Board.

the person's interests or rights (the 'result' element) (WP29, 2007). For the definition of personal data to be met, the person that the information relates to may be identified (i.e. singled out of a group), but also identifiable (i.e. possible to be singled out) directly (e.g. by a name or by an identification number) or indirectly (e.g. by a unique combination of various factors, such as sex, date of birth and postal code) (WP29, 2007). The concept of personal data is therefore extremely broad; it refers not only, as some may believe, to data containing named entities. In particular, all the unaltered video and voice recordings involving physical persons shall be regarded as personal data, as they often allow to identify the speaker. Indeed, recital 26 of the GDPR specifies that in order to determine whether a natural person is identifiable *"account should be taken of all the means reasonably likely to be used".* The recital continues: *"To ascertain whether means are reasonably likely to be used to identify the natural person, account should be taken of all objective factors, such as the costs of and the amount of time required for identification, taking into consideration the available technology at the time of the processing and technological developments".* It is therefore expressly stated that identifiability may change over time – data that do not allow to identify the data subject today, may identify him tomorrow. Consequently, it is necessary to periodically review the results of anonymization processes. On the other hand, data that do not relate to any natural person or that do not allow to identify the natural person that they relate to (because they have been successfully anonymised) shall be regarded as anonymous data and can be freely processed.

The notion of processing is also very broad; it is defined as *"any operation or set of operations which is performed on personal data (…) whether or not by automated means"* (art. 4 no. 2 of the GDPR). In particular, this includes collection, storage, consultation, adaptation, but also erasure. Personal data have to be processed in a way that is lawful (see below), fair and transparent (art. 5.1(a) of the GDPR). The data can only be collected for specified, explicit and legitimate purposes, and not further processed in a manner incompatible with these purposes (purpose limitation — art. 5.1(b) of the GDPR). The data that are being processed have to be limited to what is necessary in relation to the purposes of processing (data minimization – art. 5.1(c) of the GDPR) and only kept for as long as necessary (storage limitation — art. 5.1, (e) of the GDPR). They should be accurate and, when necessary, kept up to date (art. 5.1(d) of the GDPR). Moreover, they should be processed in a manner that ensures appropriate security, *"including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures"* (art. 5.1(f) of the GDPR).

The processing is lawful if one of the conditions set forth in art. 6 of the GDPR is met. Chiefly, it is the case when the data subject (i.e. the person that the data refer to) has given his consent to the processing (art. 6.1(a) of the GDPR). The GDPR contains some important precisions when it comes to the conditions for valid consent (see esp. art. 4 no. 11 of the GDPR): most importantly, it should be freely given, specific (i.e. limited to a specific purpose of processing), informed (i.e. prior to giving his consent, the data subject should be informed at least about the identity of the data controller and the purposes of the processing – recital 42 of the GDPR) and unambiguous. It can be withdrawn by the data subject at any moment (art. 7.3 of the GDPR). On the other hand, consent does not necessarily have to be given in a written document – it can also be an oral statement or any other unambiguous indication of the data subject's agreement to the processing[4].

Apart from consent, other legal grounds for processing are also possible; from the perspective of the language resources community, the most important of these alternative grounds is the pursuit of legitimate interests of the controller (arguably, research can be such a legitimate interest). Processing can indeed be based on this ground, unless the legitimate interests of the controller are overridden by the interests or fundamental rights and freedoms of the data subject (art. 6.1(f) of the GDPR). In assessing this, account should be taken of the reasonable expectations of the data subjects, as well as the applied safeguards (WP29, 2014).

This brings us to the last point of this paragraph: one of the new and interesting additions in the GDPR which is the introduction of the notion of pseudonymisation, absent from the Directive 95/46/EC. Pseudonymisation is defined as *"the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organisational measures to ensure that the personal data are not attributed to an identified or identifiable natural person"* (art. 4 no. 5 of the GDPR). Pseudonymisation is specifically not equivalent to anonymization (pseudonymised data are still to be regarded as personal data – recital 26 of the GDPR), but it is a safeguard measure that can be taken into account in assessing the risks of processing for the data subject.

2.2. Reinforced obligations of the data controller

The data controller is an entity that determines the purposes and means of processing of personal data. The GDPR considerably reinforces the obligations of the data controller.

Most importantly, the GDPR introduces the principle of accountability, according to which the controller shall be responsible for compliance of the processing with the principles of the GDPR, and able to demonstrate this compliance (art. 5.2 of the GDPR). In particular, whenever processing is based on the data subject's consent, the controller should be able to demonstrate this consent (art. 7.1 of the GDPR). It is also his responsibility to implement appropriate technical and organisational measures and, when necessary, policies in order to ensure compliance with the GDPR (art. 24 of the GDPR).

Moreover, such technical and organizational measures should be implemented *"both at the time of the determination of the means for processing and at the time of the processing itself".* This is particularly important to ensure that the principle of data minimization (see above) is respected. Therefore, the data controller is obliged to implement

[4] For more information about consent, see: WP29, 2017.

"data protection by design and by default" (art. 25 of the GDPR).

The controller is also (with some exceptions) obliged to maintain a written record of his processing activities (see art. 30 of the GDPR for details). In addition to that, when processing presents a high risk to rights and freedoms of natural persons, the controller is obliged to carry out, prior to the processing, an assessment of the impact of the envisaged processing (art. 35 of the GDPR) (WP29, 2017a).

In order to comply with the transparency principle, the controller should also adopt appropriate measures to provide the data subject with information listed in articles 13 and 14 of the GDPR, and in particular the identity of the controller, the purpose of the processing, the recipients of envisaged data transfers, the period for which the data will be stored and the rights of the data subject with regards to his personal data (see art. 15 through 21 of the GDPR).

3. Data Management Plan in H2020

The H2020 DMP Template consists of six sections: (1) Data Summary, (2) FAIR (Findable, Accessible, Interoperable and Re-Usable) Data, (3) Allocation of Resources, (4) Data Security, (5) Ethical aspects, (6) Other.

Since under the GDPR data processors have to implement the *"privacy by design and by default"* approach, and shall be able to demonstrate their compliance with this and other GDPR rules according to the accountability principle (see above), in the projects involving processing of personal data it is useful to integrate certain principles of the GDPR in the DMP, even using the exact wording of the GDPR. While this is not by itself sufficient to prove actual compliance with the GDPR, it would undoubtedly help demonstrate the processor's proactive attitude and spread good practices within the community. A well-drafted (in plain, understandable language) DMP should also be made available to data subjects in order to comply with the transparency requirement (see above). Moreover, it is an excellent basis for the record of personal data processing activities which processors of such data are obliged to keep.

We will now see how different GDPR rules can be incorporated into various sections of the DMP.

3.1. Data Summary

The Data Summary is an introductory section which, among others, states the purpose of data collection. In our opinion, it is useful not only to clearly state the purpose of processing in that section (according to the principle of purpose limitation of art. 5.1(b) of the GDPR), but also to mention the principles of data minimisation (art. 5.1(c) of the GDPR) and data accuracy (art. 5.1(d) of the GDPR), as well as on which legal ground personal data are being processed (i.e. consent or some alternative ground).

3.2. Data Accessibility

Section 2.2 of the H2020 DMP Template addresses data accessibility. It is this section that is supposed to explain how and to whom the data is going to be made available. In the terminology of the GDPR this corresponds to the *"envisaged transfers"* of data, of which the data processor have to inform the data subject. Moreover, the data subject has the right to access the data (according to art. 15 of the GDPR), which should also, in our opinion, be stated in this section of the DMP.

3.3. Data Re-Usability

Section 2.4 of the H2020 DMP Template concerns the reusability of the data. While this section normally deals with IPR licensing questions, in our view it is important to mention three aspects related to data protection (and which also have to do with data security): anonymisation, storage limitation (art. 5.1(e) of the GDPR) and purpose limitation (art. 5.1(b) of the GDPR). Indeed, it is useful to state in this section that personal data will not be re-used for purposes incompatible with the purpose for which they were initially collected (as stated in the Data Summary section) and stored for longer than necessary to achieve this purpose. As soon as the purpose can be achieved without processing personal data, the data shall be anonymised.

3.4. Allocation of Resources

Section 3 of the H2020 DMP Template addresses *"Allocation of Resources"*. While the primary concern of this section is the estimation and coverage of financial costs of processing, it should also *"clearly identify responsibilities for data management in [the] project"* (H2020, 2016). Therefore, this is where the data controller should be clearly identified (the entity that defines the means and purposes of data processing), and data processors (entities processing the data on behalf of the controller) should be named. It is also useful to restate some of the rules regarding the relation between the data controller and data processors (art. 28 of the GDPR), such as the one according to which the processor *"processes personal data only on documented instructions from the controller"* and *"shall not engage another processor without prior specific or general written authorisation of the controller"*.

3.5. Data Security

Section 4 of the H2020 DMP Template is fundamental from the point of view of GDPR-compliance. Indeed, the principle of data integrity and confidentiality (art. 5.1(f) of the GDPR) requires that data shall be processed *"in a manner that ensures appropriate security (…), including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures"*. Apart from specific security measures adopted in the project (such as pseudonymisation — see art. 32 of the GDPR), this section should also expressly state that the data are processed according to this general principle.

3.6. Ethical Issues

Section 5 of the H2020 DMP Template is the appropriate place to mention the results of a data protection impact assessment (art. 35 of the GDPR) if such an assessment has been carried out. Moreover, it is useful to mention in this section that such an assessment will be carried out whenever required by law. In our opinion, this section is also a good place to mention that any security breaches will be promptly reported to the national supervisory authority (as required by art. 33 of the GDPR) and communicated to the data subjects (as per art. 34 of the GDPR).

4. The role of ELRA/ELDA

In order to help the language resources community adapt to the new rules, ELRA will review its DMP. While distributing language resources, ELDA will also act as an intermediary between data subjects and the data controller. Moreover, it also proposes a legal helpdesk[5], freely available to those who want to learn more about the new Regulation.

4.1. ELRA Data Management Plan

Since 2015, ELRA proposes its own DMP template, designed specifically with language resources in mind (Choukri et al., 2016). Recently this DMP has been reviewed and adapted to the GDPR. The "Project Description" section now includes a statement on whether personal data are processed within the project. If the answer is in the positive, other elements of the DMP are modified accordingly. In particular:

○ the "Project Description" section:

- clearly and unambiguously identifies the data controller (or joint controllers);
- contains a statement that if a data processor is involved, he shall *« processes personal data only on documented instructions from the controller »* and *« shall not engage another processor without prior specific or general written authorisation of the controller »*;
- states the purpose for which personal data are processed and restates the principle of purpose limitation (art. 5.1 (b) of the GDPR);

○ the "Data Acquisition" section:

- restates the principles of data minimisation (art. 5.1 (c) of the GDPR) and data accuracy (art. 5.1 (d) of the GDPR);
- clearly identifies the legal ground for processing of personal data (consent or an alternative ground) and restates the principle of lawfulness (art. 6 of the GDPR);

○ the "Legal Issues and Ethics" section:

- restates the obligation to anonymise personal data as soon as possible in relation to the purposes of processing;
- restates the obligation to carry out a Data Protection Impact Assessment and presents the results of the DPIA if it was carried out;

○ the "Sustainability" section:

- reminds that according to the principle of data accuracy (art. 5.1 (personal data have to be kept up to date;
- reminds that the results of anonymization shall be periodically reviewed to take into account technological progress in identification techniques;

○ the "Data Storage Section" restates the principle of data integrity and confidentiality (art. 5.1 (f) of the GDPR), and the principle of storage limitation (art. 5.1 (e) of the GDPR);

○ the "Data Access and Sharing" section:

- lists envisaged data transfers;
- reminds of the data subject's right of access (art. 15 of the GDPR).

These changes are intended to educate the community about the current legal framework and to spread good practices. The new ELRA DMP will also – if applied and followed by the users – facilitate the demonstration of compliance with the GDPR, required under the principle of accountability (art. 5.2 of the GDPR, see above).

4.2. ELDA as an intermediary

While distributing language resources, ELDA will also play the role of an intermediary between the data subject and the data controller. This role will consist primarily of providing the data subjects with information required by the GDPR (art. 13 and 14 of the GDPR), according to the transparency principle. ELDA will also guarantee the data subject's right to withdraw his consent to further processing of his personal data (as well as the right to erasure, restriction and objection) by enabling him to communicate his wishes directly to ELDA which will then transmit it to the controller.

4.3. ELDA's Legal Helpdesk

ELDA's Legal Helpdesk will also provide, free of charge, any member of the community with information regarding the GDPR and recommendations on how to comply with the new rules. Moreover, ELDA can also offer assistance (including on-site) regarding data protection and data management.

5. Closing remark — the role of the Codes of Conduct

In the previous sections, we have established that a well-drafted DMP is essential for demonstrating compliance with the GDPR, and in particular the principles of transparency and accountability. However, it is not the only tool to demonstrate compliance with the GDPR. Another such tool would be a community-wide Code of Conduct regarding processing of personal data for Natural Language Processing purposes. Indeed, art. 40 of the GDPR encourages the adoption of Codes of Conduct by *"associations and other bodies representing categories of controllers or processors"*. Such codes have to be approved, registered and published by the national supervisory authority; it may be even granted universal validity by the European Commission (and "complement" the GDPR). The GDPR also specifies the process of monitoring compliance with such codes of conduct (art. 41), and even certification processes (art.

[5] http://www.elra.info/en/services-around-lrs/legal-support-helpdesk/

42). The adoption of such a code of conduct would require a substantial, internationally coordinated effort on behalf of the language resources community, but the reward (simplified demonstration of compliance and lower transaction costs) may be well worth it.

If such an international Code of Conduct is adopted, or if national codes are adopted at the level of some Member States, the adherence to such a Code should also be mentioned in the DMP.

6. Bibliographical References

Choukri, K.; Mapelli, V.; Mazo, H.; Popescu V. (2016). ELRA Activities and Services. In: Proceedings of LREC 2016, pp. 463-468.

H2020 Programme (2016). Guidelines on FAIR Data Management in Horizon 2020, version 3.0, 26 July 2016.

H2020 Programme (2017). Annotated Model Grant Agreement, Version 4.0.1, 20 June 2017.

Regulation 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data.

WP29 (Article 29 Data Protection Working Party) (2007). Opinion 4/2007 on the concept of personal data.

WP29 (Article 29 Data Protection Working Party) (2014). Opinion 06/2014 on the notion of legitimate interests of the data controller under Article 7 of Directive 95/46/EC.

WP29 (Article 29 Data Protection Working Party) (2017). Guidelines on Consent under Regulation 2016/679.

WP29 (Article 29 Data Protection Working Party) (2017a). Guidelines on Data Protection Impact Assessment (DPIA) and determining whether processing is "likely to result in a high risk" for the purposes of Regulation 2016/679.

We Are Depleting Our Research Subject as We Are Investigating It: In Language Technology, more Replication and Diversity Are Needed

António Branco
University of Lisbon
NLX-Natural Language and Speech Group, Department of Informatics
Faculdade de Ciências
Campo Grande, 1749-016 Lisboa, Portugal
antonio.branco@di.fc.ul.pt

Abstract

In this paper, we present an analysis indicating that, in language technology, as we are investigating natural language we are contributing to deplete it in the sense that we are contributing to reduce the diversity of languages. To address this circumstance, we propose that more replication and reproduction and more language diversity need to be taken into account in our research activities.

Keywords: science ethics, science policy, language technology, human language diversity, replication of research results, language policy.

1. Introduction

Natural language is a most extraordinary object of scientific inquiry lending itself to be researched at least as a referential symbolic system, a socially effective type of behavior or a class of specialized mental activities, and hopefully one day as a principled unified combination of all its dimensions. As it is at the core of what distinctively human nature may be, the approximately 7 000 human languages existing in our planet are a most valuable treasure trove for scientific inquiry on the human brain, mind and behavior, and for advancing our understanding of ourselves and finding better technological solutions that improve our life and heal us.

While informed laypersons are aware of the dramatic consequences of the depletion of important natural resources, from energy to bio-diversity and including potable water, ozone layer among several others, they are much less, or not at all aware of the threat hanging over language diversity. Around one third of the world languages are at present vulnerable to become extinct according to UNESCO (Moseley, 2010). In this paper we start by pondering on the impact that the very development of language science and technology at large is having on language diversity. We will then proceed with this analysis by narrowing the focus of our reflection into a case study in the realm of language resources. With the present paper, we aim at fostering the debate and action on how language scientists and the research activities on language science and technology, including on the development of language resources, can be much more mindful of the language diversity issue and must bring it to the center stage of its mission. It is not only natural and human heritage but also their very own object of research that is being eroded as they are investigating it and because they are investigating it they way that investigation is being undertaken.

2. Monolingual vortex

In the prevailing model for the promotion and funding of research, science progress is mostly driven by the societal priorities identified in the different countries and entities supporting its development. The growth of scientific knowledge is thus asymmetric among different areas, different disciplines inside a given area, different topics inside a given discipline, etc.

Human language science and technology is no exception. As in each country research tends to be prioritized mostly towards its official or predominant language(s), a most substantial asymmetry here is the different research effort devoted to different languages. As a matter of fact, this asymmetry could not be more extreme given that it tells apart one language, viz. English, from all other languages. This was neatly captured in the study of (Mariani and Frankopoulo, 2012), whose findings are summarized in Figure 1, where the research effort devoted to English is five times larger than the effort devoted to the second most researched language, confirmed by an independent study based on a different sample (Rehm and Uszkoreit, 2013), p. 10.

English is the predominant language of the United States, the country that is the world superpower, and of a number of other economically and scientifically highly developed countries, including the United Kingdom, Canada and Australia. All in all, it is the language of around 1/4 of the world GDP — with the second language with the largest share, Chinese, with only half of that GDP value, followed by a long tail of other languages related to economies with GDPs all below half of the Chinese score. But this extreme asymmetry is not explained only by the funding model of science. It results also from how science is produced and to a Matthew effect of accumulated advantage this induces.

At some point or other of their doctoral research on language technology, most students from a non English speaking country had to face the decision on whether they pick their mother tongue or English as the object language of their study. Although English is not their native language, it has many more language resources, processing tools and applications available that can be reused, and many more published research results on which further progress can be built, and this offers a very clear picture: Adopting English

as the object language for their research substantially enhances their chances that more new results are produced in less time, and hence their chances of eventually getting more papers accepted for publication, a more visible dissertation and a better career.

We all know many colleagues and students that when faced with this individual choice understandably opted for their immediate best interest and chose English. Regrettably, in collective terms this represents the diverting of the very few resources available in non English speaking countries to support, again, research on English, thus further widening the gulf between advancing the research on English and on the other languages, in detriment of the latter.

This is a self-reinforcing draining effect that is active at many levels that further reinforce each other: When senior researchers decide which research themes to pursue that could be more rewarding for their promotion; when the heads of research units decide which research lines to support that enhances the chances of a better assessment outcome and more future funding for their units, etc.

And all this is strengthened by the fast science funneling effect, where replication and reproduction of previous results are not being produced, accepted or published in almost all venues, not even for English.

It is as if one would have decided, in a counter-factual world, to establish the scientific realm of Biology by researching only one species, or a half dozen of them at best. Certainly these researchers would come across many aspects that we would know that are common and universal to all species and living beings, but they would have no means to figure that out, or even to hypothesize that that was the case given they would have no access to the other thousands of species.

3. Monolingual cyberworld

Along human history, natural language was gone through technological shocks, among which some of the most well known, for instance, are the advent of writing or of the printing press. New technologies have permitted an enhanced usage of languages, allowing to break temporal, spatial and social limitations of face to face communication.

But this usually comes at the cost of a reduction in language diversity. Languages that due to historical or economical circumstances were not technologically prepared or did not receive the benefits of the new technology tended to be abandoned by their speakers. A well known example is the decline and eventual extinction of many languages and dialects that were not used as vehicular languages by the newspapers in previous centuries (Wright, 2016).

Leaving aside the violent cases where languages are banned or their speakers are decimated, languages get extinct because their speakers abandon it in favor of another language. And the causes are basically the same as the ones that lead emigrants to abandon their rural villages to move to big metropolis: The new language is felt to grant more competitive advantages than the one that is getting abandoned, and eventually extinct, in their search for a better life.

Natural language is undergoing a technological shock with unprecedented historical and civilizational consequences. Language technology will allow to overcome further limitations of communication. It will allow speakers that do not share a common language to instantly communicate with each other, supported by automatic translation services. And it will permit to communicate seamlessly with all sort of devices, digital services and artificial agents in natural language.

Even more than in previous technological shocks, language diversity is under the risk of being drastically depleted. The digital world is bringing new disruptive forms of living and working with wide new competitive advantages that cannot be ignored. Natural languages that will be technological prepared will be major channels and instruments to get access to those benefits and will constitute a most relevant competitive advantage. They will be an irresistible attracting pole for speakers of other languages that will not undergo sufficient technological preparation, in yet another instantiation of the Matthew effect of accumulated advantage.

It turns out that language science and technology is neglecting a vast portion of its research object, by neglecting the research on and the technological preparation of the vast majority of the languages. At this point, it is also evident that this risks to contribute for the depletion of its very own research object. The more we funnel our research into one language, or a handful of languages, the more we are contributing for the risk of others to get extinct in the digital age, and thus for further funneling, and eventually locking, ourselves into doing research in one language.

Under the current historical circumstances, ignoring language diversity in our research is not only neglecting a vast portion of our research object and delaying our scientific progress. It is also contributing to reduce language diversity and eventually compromising our chances to get to understand our very research object, human language. Given the way we are doing language science and technology, one could well say that we are contributing to the depletion of our very own research object as we are investigating it. A quite singular situation for the scientific ethos.

4. Multilingual replication

The funneling and depletion effects commented on above can be traced back to the asymmetries underlying how scientific activities happen to be nowadays economically and socially sustained and deployed. These are asymmetries that are common to all scientific areas and disciplines and are inducing all sorts of biases compromising not only the effectiveness but also the integrity of the scientific endeavor.

This discussion has grown in importance as the resources allocated to and societal impact of scientific activities have been expanding (e.g. (Stodden, 2013), (Aarts and others, 2013), to the point that it has crossed the borders of the research world and made its appearance in important mass media and was brought to the attention of the general public (e.g. (Nail, 2011), (Zimmer, 2012), (Begley, 2012), (Begley and Ellis, 2012), (Hiltzik, 2013), (Economist, 2013)). The immediate motivation for this increased interest is to

be found in a number of factors, including the realization that for some published results, their replication is not being obtained (e.g. Prinz et al. (2011), Begley and Ellis (2012)); that there may be problems with the commonly accepted reviewing procedures, even besides their possible lack of quality, where deliberately falsified submissions, with fabricated errors and fake authors, get accepted even in respectable journals (e.g. Bohannon (2011)); that the expectation of researchers vis a vis misconduct, as revealed in inquiries to scientists on questionable practices, scores higher than one might expect or would be ready to accept (e.g. Fanelli (2009)); among several others.

Underneath these immediate causes, a number of factors have been pointed out, including career and promotion pressure too biased for quantity; widespread disinterest on negative results as an intrinsic part of the scientific progress; widespread disfavoring of activities of replication by funding agencies; poor or non existent retraction procedures for results that are eventually noticed to be wrong or flawed after having been published; ideological pressure to get immediate financial return from research results; etc. In Bill Frezza's bold opinion, the financial pressure on the scientific system "has created a moral hazard to scientific integrity no less threatening than the moral hazard to financial integrity that recently destroyed our banking system." (Frezza, 2011).

Given the way — together with other colleagues from other disciplines — we are doing science and technology, one could well say that we are contributing to the depletion of the conditions of possibility of our very own research activity. Another quite singular situation for the scientific ethos.

Against this background, it is compelling to advocate that like in other scientific areas, we very much need to foster practices that enhance reproducibility and replicability of research results and bring them to the center stage of our scientific activities, amplifying pioneering initiatives like the 4REAL workshop (Branco et al., 2016). Following the text introducing a new Special Section of the Language Resources and Evaluation journal on reproducibility and replicability (Branco et al., 2017): "Reproduction of results entails arriving at the same overall conclusion(s), as opposed to finding identical values for some measure (Drummond, 2009), (Dalle, 2012), (Buchert and Nussbaum, 2011); that is, to appropriately validate a set of results, scientists should strive to reproduce the same answer to a given research question by different means, possibly by re-implementing an algorithm or evaluating it on a new dataset. Replication has a somewhat more limited aim, typically involving running the exact same system under the same conditions in order to arrive at the same output result."

In a previous occasion, we have motivated this need on the interest of securing the integrity and quality of the research results in our area (Branco, 2013). In the context of the present paper, this need gets further reinforced as a key measure to counteract the funneling and depletion effects that were commented on above and that in the long run appear as self-defeating our own scientific endeavor.

It is important that results obtained when working on some object language(s) are reproduced and replicated with those same language(s) and also with other languages. In the long-term interest of our research area and research subject, it is important that this becomes accepted and encouraged as a first-class citizen practice of our scientific activities.

5. Multilingual diversity

The funneling and depletion effects commented on above can be traced back to overall asymmetries that are common to all scientific areas and disciplines, including ours. This calls for our community to be aligned with and be a major contributor for global correctives initiatives, like paying due attention to replication and reproduction of results in scientific research.

But there are biasing effects that emerge as specific of our area given its particular characteristics and the specific nature of its research subject, and call for responses that should be specific. And for a problem to be addressed and corrected, the first basic requirement is that there is sufficient awareness that it exists.

Anecdotal evidence that in language technology,[1] language diversity is obliterated and that this is not being perceived as an issue, can be found on how titles happen to be chosen for papers in international venues. The few publications whose results are obtained working with an object language different from English typically have an explicit mention to that language in the title. The vast majority of the papers, in turn, which takes English as an object language, makes no reference to English in the title, and many times, not even in the body of the articles. It is compelling to envisage this socially accommodated behavior as a manifestation of a collective unconscious assumption — by all authors, from both sorts of papers alike — that English is "the" natural language by default, and the other languages are just a source of additional exotic or picturesque details.

More seriously than the inessential wording of titles, this bias has been endured by researchers who receive reviews for their papers whose object languages do not include English. If one pays attention to their shared stories during coffee breaks in conferences, one come to realize that more often than not they are questioned by anonymous reviewers whether their results also hold for English (but not, say, for Finnish, Farsi, Hindi, Japanese or any other one of 7 000 languages in the world), or even advised that English should be tried for the paper to be considered mature to be submitted for publication.

Anecdotal aspects aside, the bias this is illustrating has a decisive impact on how our research activities are fostered and our results have been pursued. In this extended abstract, we will focus on one particular example, meant to be illustrative.

As a language resource, WorddNet is a most well known and important asset in language technology. As a multiparty open research initiative, it is a most successful one in our area. And as a case of replication, it is a most prominent one, with an ever growing number of WordNets in construction for particular languages. This is why WorNet offers a telling example that diversity is needed and can be promoted.

[1] With honorable exceptions, like the research communities gathering around LREC/ELRA conferences and only a very few others.

There have been a number of initiatives, including EuroWordNet, MultiWordNet, BalkaNet, etc. (Vossen, 1998), (Pianta et al., 2002), (Tufiş et al., 2000), where concepts that are from different WordNets and are semantically equivalent are co-indexed with each other. As semantic equivalence is a transitive relation, it suffices that each concept, in a particular language/WordNet, is indexed with an equivalent concept, in any other language/WordNet. However, in practice concepts from all languages other than English have been connected to concepts of only one other language, namely English. In practice, no sustained studies, development tools or alternative multilingual ensembles that support a different approach have been pursued.

When there are two equivalent concepts that can be lexically expressed in two languages other than English, but that cannot expressed in English, that equivalence has remained unrecorded. This has funneling effects that once again brings superior development effort and competitive advantage to English. The WordNet for this language has the widest translational homomorphism, built though at the cost of a significant share of the resources deployed for and during the construction of the WordNets for the other the languages. And at the cost that translational equivalence is eventually sub-optimally registered among other languages.

More recently, this funneling effect had been further fostered as a side effect of other initiatives, including Open Multilingual WordNet, BabelNet, etc. (Bond and Paik, 2012), (Navigli and Ponzetto, 2012), whose goal is to gather ensembles of WordNets mapped among themselves, rather than just co-indexing their concepts. As these ensembles start being increasingly used and cited in the literature, the existence of the individual WordNets gets obfuscated. Given these ensembles appear as a convenient one-stop reference, even when only one particular WordNet in them is needed, the former are the favored reference. Citations to individual WordNets are thus vanishing, and with them the incentives and the research productivity indicators that researchers and funding entities need in order to support the continued research on other languages/WordNets.

As a first possible step contributing towards mitigating these funneling effects, we have proposed the undertaking of a Pluricentric Global Wordnet (Branco et al., 2018). But our goal here is not to motivate and present this notion. Rather, WordNet is being offered just as one illustrative case — among possibly many existing ones — of asymmetric biases that may be specific to our field, and to each one of our research topics, and that need to be addressed with specific responses that go on a par with an increased attention to reproduction and replication of results.

Such specific responses are needed to secure and enhance diversity in a wide range of dimensions in our research activities. We need more language diversity in every aspect of our procedures in our scientific activity, ranging from how we set up the reviewing of papers in conferences, to how we conceive our research questions and deploy our priorities around every one of our research topics or subareas, and including crucially how we raise and lobby for the funding of our activities.

Certainly, the need for more language diversity echoes, even if at a different level, the overall need for more diversity in our area, including the diversity in terms of methodological approaches, gender, etc., that have also started to be identified at other venues (Nivre, 2017).

6. Final remarks

In this paper, we presented an analysis aimed at bringing to light two processes that are induced by our research activities in language technology and whose combination are one of the major contributions for the depletion of language diversity. One the one hand, our research focus mostly in one language, English. On the other hand, natural language is undergoing a historical technological shock that is reducing the social and economical competitive advantages for the vast majority of individual languages other than English. Each one of these processes is an instance of and is propelling a powerful Mathew effect of accumulated advantage, which get even further aggravated by the compounding effect of their confluence and the exponential magnifying combination with each other.

Besides raising awareness about this circumstance as a major issue questioning our practice as scientists, in this paper, we also propose measures to mitigate and counteract this unwelcome contribution of language technology for the depletion of its own research subject.

A set of measures results from the alignment with an emerging global trend in science that is urging for greatly increasing the replication and reproduction of research results, for the sake of securing the overall credibility of the scientific knowledge and endeavor. Language technology also needs and benefits from aligning as most and as rapidly as possible with this trend. This process needs and should — and provides a major opportunity — to extend replication and reproduction to languages other than English.

Another set of measures are specific to our area and result from bringing to light and counteracting the non assumed viewpoint that English is the natural language by default. We need more language diversity in our scientific procedures. This should trigger an overall collective process of renewing our activities, ranging from how we set up the reviewing of submitted papers in conferences, to how we conceive our research questions and deploy our priorities around every one of our research topics or subareas, and including crucially how we lobby for the external support to our activities.

Acknowledgment

The present research was partly supported by the Infrastructure for the Science and Technology of the Portuguese Language (CLARIN Língua Portuguesa), and by the ANI/3279/2016 grant.

7. References

Aarts, A. et al. (2013). Estimating the reproducibility of psychological science. *Science*, 349.

Begley, G. and Ellis, L. (2012). Drug development: Raise standards for preclinical cancer research. *Nature*, 483:531–533.

Begley, S. (2012). In cancer science, many "discoveries" don't hold up. March 28, 2012, online edition. Reuters.

Bohannon, J. (2011). Who's afraid of peer review? *Science*, 341:60–65.

Bond, F. and Paik, K. (2012). A survey of wordnets and their licenses. In *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, pages 64–71.

António Branco, et al., editors. (2016). *Proceedings of the Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language (4REAL), of LREC2016*. ELRA.

Branco, A., Cohen, K. B., Vossen, P., Ide, N., and Calzolari, N. (2017). Replicability and reproducibility of research results for human language technology: Introducing an lre special section. *Language Resources and Evaluation*, 51:1–5.

Branco, A., Branco, R., Saedi, C., and Silva, J. (2018). Browsing and supporting pluricentric global wordnet, or just your wordnet of interest. In *Proceedings of LREC 2018*.

Branco, A. (2013). Reliability and meta-reliability of language resources: Ready to initiate the integrity debate? In Sandra Kuebler, et al., editors, *Proceedings of the 12th Workshop on Treebanks and Linguistic Theories (TLT12)*, pages 27–36. Bulgarian Academy of Sciences.

Buchert, T. and Nussbaum, L. (2011). Leveraging business workflows in distributed systems research for the orchestration of reproducible and scalable experiments. In Anne Etien, editor, *9ème édition de la conférence MAnifestation des JEunes Chercheurs en Sciences et Technologies de l'Information et de la Communication - MajecSTIC 2012 (2012)*.

Dalle, O. (2012). On reproducibility and traceability of simulations. In *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pages 1–12. IEE.

Drummond, C. (2009). Replicability is not reproducibility: nor is it good science. In *Proceedings of the Evaluation Methods for Machine Learning Workshop at the 26th ICML*.

Economist. (2013). Unreliable research: Trouble at the lab. *The Economist*, October 19, 2013, online edition. http://www.economist.com/news/briefing/21588057-scientists-think-science-self-correcting-alarming-degree-it-not-trouble.

Fanelli, D. (2009). How many scientists fabricate and falsify research? a systematic review and meta-analysis of survey data. *PLOS ONE*. doi:10.1371/journal.pone.0005738.

Frezza, B. (2011). The financially driven erosion of scientific integrity. *Real Clear Markets*, December 5, 3011, online edition. http://www.realclearmarkets.com/articles/2011/12/05/the-financially-driven-erosion-of-scientific-integrity-99401.html.

Hiltzik, M. (2013). Science has lost its way, at a big cost to humanity. *Los Angeles Times*, October 17, 2013, online edition. http://www.latimes.com/business/la-fi-hiltzik-20131027,0,1228881.column#ixzz2lT8zjZWD.

Mariani, J. and Frankopoulo, G. (2012). Language matrices and the language resource impact factor. In *PAROLE Workshop, Lisbon, 18-19 October 2012*.

Christopher Moseley, editor. (2010). *Atlas of the World's Languages in Danger*. UNESCO Publishing, third edition.

Nail, G. (2011). Scientists' elusive goal: Reproducing study results. *The Wall Street Journal*, December 2, 2011, online edition. http://online.wsj.com/news/articles/SB10001424052970 20376480457705 9841672541590.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Nivre, J. (2017). Presidential address ACL 2017: Challenges for ACL. *The 55th Annual Meeting of the Association for Computational Linguistics (ACL)*. https://pt.slideshare.net/aclanthology/joakim-nivre-2017-presidential-address-acl-2017-challenges-for-acl.

Pianta, E., Bentivogli, L., and Girardi, C. (2002). Multiwordnet: Developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*.

Prinz, F., Schlange, T., and Asadullah, K. (2011). Believe it or not: how much can we rely on published data on potential drug targets? *Nature Reviews Drug Discovery*, 10:712.

Georg Rehm et al., editors. (2013). *META-NET Strategic Research Agenda for Multilingual Europe 2020*. Springer.

Stodden, V. (2013). Resolving irreproducibility in empirical and computational research. *Institute of Mathematical Statistics Bulletin Online*. http://bulletin.imstat.org/2013/11/resolving-irreproducibility-in-empirical-and-computational-research.

Tufiş, D., Cristea, D., and Stamou, S. (2000). Balkanet: Aims, methods, results and perspectives - a general overview. *Romanian Journal of Information Science and Technology Special Issue*, 7:9–42.

Vossen, P. (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer.

Wright, S. (2016). *Language Policy and Language Planning: From Nationalism to Globalisation*. Palgrave Macmillan UK, second edition.

Zimmer, C. (2012). A sharp rise in retractions prompts calls for reform. *The New York Times*, April 16, 2012, online edition. http://www.nytimes.com/2012/04/17/science/rise-in-scientific-journal-retractions-prompts-calls-for-reform.html.

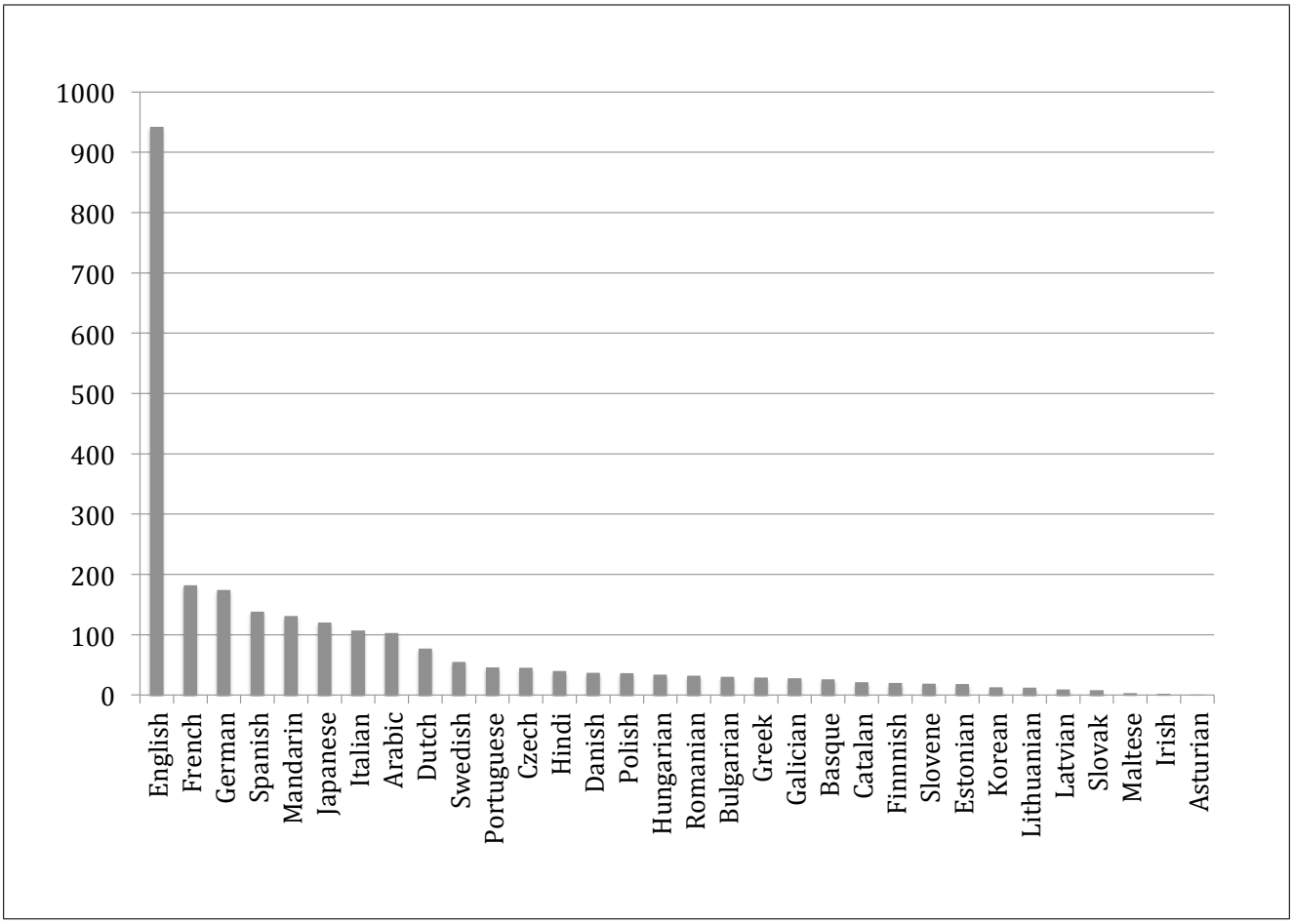


Figure 1: Number of references to data sets and processing tools per language at top scientific conferences between 2010 and in 2012, from (Mariani and Frankopoulo, 2012).

Lessons Learned: On the Challenges of Migrating a Research Data Repository from a Research Institution to a University Library

Thorsten Trippel and Claus Zinn
University of Tübingen
Wilhelmstrasse 19, 72074 Tübingen, Germany
{thorsten.trippel, claus.zinn}@sfs.uni-tuebingen.de

Abstract

The transfer of research data management from one institution to another infrastructural partner is all but trivial, but can be required, for instance, when an institution faces reorganisation or closure. In a case study, we describe the migration of all research data, identify the challenges we encountered, and discuss how we addressed them. It shows that the moving of research data management to another institution is a feasible, but potentially costly enterprise. Being able to demonstrate the feasibility of research data migration supports the stance of data archives that users can expect high levels of trust and reliability when it comes to data safety and sustainability.

Keywords: Research Data Management, Data Repositories, Data Migration

1. Introduction

Good scientific practice requires that research data created and studied by scientists is archived. The sustainable archiving of research data is a complex manner as it includes the entire life cycle of data. In the different phases of this life cycle, many human factors are involved, often placed in different organizational structures, and making use of many technological frameworks.

The sustainable management of research data is a noble aim, but there is no single golden path to sustainability. Also, the path might suddenly encounter a road block, when for instance, an existing archival infrastructure faces a discontinuation because a research institution faces closure or a new research orientation. Here, the sustainability of research data management depends on another institution being able and willing to take over the data.

Though the scholars are partly responsible for archiving, they can hardly be responsible for running the archive. Research infrastructures and networks of institutions claim that they fulfill this responsibility, using certified technical infrastructure and processes. The cooperation between institutions in such infrastructures and networks strengthens the overall reliability of each partner, but the expression of intent can be severely tested if one partner discontinues its service.

In this paper, we report on the following use case: research data has been collected, evaluated, catalogued, and made accessible by a research institution in an exemplary manner. It is assumed that this data centre is discontinued, but that the research data should remain available. All research data therefore needs to be migrated, in the given case from a linguistics department to a discipline independent data facility operated by an institutional infrastructure, here a university library and computing centre, which takes care of all data and guarantees its access for the foreseeable future.

For this use case, we have devised a migration concept that has uncovered a number of challenges that need to be addressed to make such as hand-over of research data management a success.

2. Background

The work reported in this paper stems from the NaLiDa project, which was divided into two main phases. The first funding phase aimed at the construction of an infrastructure for the long-term archival of linguistic resources with technology and workflows that are manageable and sustainable. The infrastructure was to be built within the research institution that creates all data, the department of linguistics at the University of Tübingen. In the second phase, the NaLiDa project took on board the two infrastructural units of the University of Tübingen, the university library and the computing centre. The aim was to explore how to best transfer the management of research data to these units for the long-time archiving of linguistic resources. Also the university library wanted to learn about the processes required to ingest all resources' metadata into their catalogues. With library catalogues connected with the research data repositories of the computing centre, users would profit from easy-to-use access points.

It turned out that the transfer of research data management is no easy matter, and that many technological and organizational hurdles exist and need to be dealt with. In the remainder of this section, we describe the management of research data at both the research institution and at the university library. The aim is to identify the commonalities and differences of Research Data Management (RDM).

2.1. RDM at the discipline specific-data centre

RDM has a technical and an organizational perspective.

2.1.1. Technical Backbone

The technical backbone at the time of the migration process comprised the following four key components:

- a Fedora Commons 3 repository (which in the meantime has been ported to Fedora Commons 4), see [U6].
- ProAI: a repository-neutral, Java web application supporting the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH), see [U7].

- ProFormA: a form-based editor for metadata management (Dima et al., 2012b) (in the meantime replaced by Comedi (Lyse et al., 2015)), and
- ERDO: a web portal for research data ingestion and maintenance (Dima et al., 2012a).

While the first two items are open-source applications, the latter two are in-house developments. The ProFormA editor is targeted at users to easily instantiate CMDI-based metadata schemas; and the ERDO web portal is used to support the data ingestion workflow, see below.

2.1.2. Workflows

The discipline oriented data centre closely cooperates with the data providers, usually the researchers who created or gathered all data. All parties involved are committed to follow FAIR, a set of guiding principles to make data Findable, Accessible, Interoperable, and Re-usable (Wilkinson, 2016), see also [U11].

Both parties initialize the archiving process in a cooperative manner. The data providers decide on the granularity of the data to be archived, but get help from the data centre staff. Such decisions can be helped by consulting, for instance, the criteria of the ISO 24619 standard on the assignment of persistent identifiers to language resources (see (ISO 24619, 2011), section 6).

The next step is to collect and upload all relevant individual files to the repository system. Depending on the type of the research data (*e.g.*, lexical resources, experimental studies, or text corpora), an appropriate CMDI-based metadata schema is selected (ISO 24622-1, 2015). An initial provision of the metadata is given by the research data creators, who presumably know their data best. However, as the data providers are not necessarily archiving experts, they consult with the data centre's archivist to answer any questions. Usually, metadata provision is an iterative process between both parties, where research data providers add missing pieces of information, and where archiving experts may curate the data.

The discipline oriented data centre is committed to open access. In practise, however, there are often cases where language related research data is subjected to restrictive data usage licenses. This is the case, for instance, when research data makes use of third party data, which is turn is published under a restrictive license, or when research data involves potentially privacy infringing data collections. In some cases, researchers would like to choose an open license for their data (*e.g.*, `https://creativecommons.org/licenses/by-nc-nd/4.0/`) but want to be consulted before it is given to an interested party. In any case, the data providers – in close consultation with the archivists – assign appropriate license and access rights to the data, varying from "open to the general public" to "protected, individual permission required per dataset".

Whenever the data provider is *not* affiliated with the data centre's institution, the rights and duties between depositor and depositee are laid down in a depositing agreement. The agreement specifies, for instance, that the depositor (i) is the owner of the intellectual property rights of the data, (ii) warrants that the dataset (and its metadata) does not contain false or misleading information, (iii) assures that the dataset does not violate or infringe any copyright, trademark, patent or intellectual property rights of third parties, and so on. The agreement grants the depositee, for instance, the rights to distribute the dataset in electronic form, to make available its metadata records through its catalogues, or to assign digital object identifiers that link metadata records with the data they describe. A good example for a deposit agreement is given in [U12], but clearly, such (legally binding) documents must be drawn up on a case by case basis.

Once the data providers have finalized the provision of metadata and access restrictions, the archivists take over, adding elements unknown to the data providers, including technical information on the submitted files (*e.g.*, checksums, file sizes, storage locations) and references to authority files if available, see (Zinn et al., 2016). Additionally the archivists start a quality assurance process for all files.

At the end of the quality assurance phase, the archival objects receive a persistent identifier according to ISO 24619. In our study the Handle system is being used [U5]. With the persistent identifiers, the archival objects are finally archived in the repository system. This process involves the publication of the metadata via the OAI-PMH protocol, see [U8]. The metadata now also includes access information to the data such as location, contact information, and license/access rights.

To honor the access rights attached to research data, the archival system implements a system for authorization, which is based on the built-in access control system by Fedora Commons. XACML authorization rules define users with name and password, and a role allowing or denying specific operations for archived objects:

```
<user name="guest" password="xxx">
   <attribute name="fedoraRole">
      <value>user</value>
   </attribute>
</user>
```

2.2. RDM at the University Library and Computing Centre

2.2.1. Technical Backbone

The technical infrastructure at the university library and computing centre makes use of the following software:

- the Fedora Commons 4 repository,
- the software Apache Solr/Lucene for indexing [U9],
- *Docuteam Packer* for the creation of packages of archival files [U10],
- ingest software for the archival and validation of digital objects, and
- portal software for research data access and rights management.

The latter two items are in-house developments.

```
<Policy>
  <Subjects>
     <AttributeValue
         DataType="http://www.w3.org/2001/XMLSchema#string">
        A04_admin
     </AttributeValue>
  </Subjects>
                                          ==>
  <Resource>
     <AttributeValue
       DataType="http://www.w3.org/2001/XMLSchema#string">
       FID:183
     </AttributeValue>
  </Resource>
<Policy>
```

Handle-PID	ACL Expression
10900.1/FID:183	+u:A04_admin

Figure 1: From NaLiDa-ACLs to library and computing centre's ACLs.

2.2.2. Workflows

At the university library, the archival process starts with a pre-ingest of the research data using the Docuteam Packer software. During the pre-ingest, the researcher gathers and structures her data into a machine-readable package, and describes it with metadata. In this process, researchers are supported by the staff of the library and computing centre. As a result, a Submission Information Package (SIP) is created that contains all research data and its metadata in the EAD/METS format (Encoded Archival Description / Metadata Encoding Transmission Standard), see [U3].

In-house software is then used to read the resulting package and validate its content for correctness and completeness. Upon successful validation, the package is being ingested into the library's digital archive, which is based on the Fedora Commons 4 repository system. As part of the ingestion process, each resource is assigned a unique persistent identifier (PID) of the Handle system. Also, all metadata is being ingested into an Apache–Solr Server.

All access to archived digital objects is performed via a purpose-built portal software. The portal gives a web-based access, and for this it makes use of a database that holds information about access rights to digital objects, and information about users and their authorization records. Authorization is defined via *Access Control Lists (ACL)*. The database associates with each PID an ACL.

When a user logs into the portal system, his user data is retrieved from the database. Users will only be able to access a resource when their credentials feature in the resource' ACL. Here, the authorization system distinguishes three access categories: roles, users, and groups. An ACL can contain any number of instructions along these categories that are either tagged as "+ (grant)" or "- (revoke)". For authentication, the central authentication server of the University of Tübingen is used. The user ids from the authentication server correspond to the user ids in the portal's database.

2.3. Commonalities and Differences

Our two data centres share common features, but there are also differences. Both rely on the same repository backend and hence share a large common technological ground. Also both centres use handle-based persistent identifiers, though with different prefixes. Major differences exist in the workflow, which is more generic at the library and computing centre; here, the research institution, naturally, offers more discipline-specific support. This is also exemplified by the different metadata schemas; here, the discipline-specific institution makes available CMDI profiles for different types of resources, while the library makes use of EAD (Encoded Archival Description), which does not discriminate against resource types. Both data centres also manage access rights differently. Here, the library-based archive has a stricter regime in place, which is also embedded in the university's authentication system.

3. Migration Concept

We outline migration issues along three dimensions.

3.1. Authentication and Authorization

The authentication and authorization procedures for accessing research data differ considerably. At the web portal of the library and computing centre, authentication is embedded in the university's central LDAP server whereas the discipline-specific NaLiDa repository uses a proprietary authentication procedure that is captured by locally maintained XML-based documents.

To address this issue, the library and computing centre needs to complement the usage of the central LDAP server with a local server that will also be consulted for user authentication. The local LDAP server will register all NaLiDa users that do not have a valid university-based id (that is, their id is not part of the central LDAP server).

With regard to authorization, both approaches use a role-based access management based on Access-Control-Lists (ACLs). However, there are differences in the use of ACLs, and where they are stored. In the library and computing centre, no user of the web portal is granted write access to digital objects. Once a digital object is ingested, it cannot be changed. In the NaLiDa repository, the archiving workflow allows ERDO users to modify the digital objects prior to their publication by the archivists. It is clear that digital objects, once transferred to the repository of the library and computing centre's repository, cannot be changed thereafter. Any NaLiDa-based access rights that grant the writing of digital objects will be revoked.

Technically, the NaLiDa repository used the functionality for access management as provided by the Fedora Commons 3 software (xacml-2.0-policy-schema). The library and computing centre repository uses a different approach that is decoupled from the repository software, the aforementioned database-driven approach. This approach helps migrating the NaLiDa-based access rights as any ACL can be mirrored to a corresponding database entry. Here, all *XACML Subjects* are mapped to users of the library and computing centre; the roles for administrator and user also have their correspondence in the repository of the library and computing centre, and all XACML READ Actions can be transformed into equivalent *grant* and *revoke* statements. All WRITE statements will not be migrated. Fig. 1 illustrates the migration of access rights from one repository system to the other.

3.2. Metadata Harmonization

There is a profound difference in the metadata used to describe research data. While the NaLiDa team uses the CMDI-Framework, which follows the ISO 24622-1 standard, the library and computing centre uses the Encoded Archival Description (EAD) scheme. The transfer of research data must hence include a transformation (crosswalk) from one metadata scheme to another.

The crosswalk is based upon an existing conversion from CMDI-based metadata profiles to Dublin Core [U1] and MARC 21 [U2], see (Zinn et al., 2016). The conversion is hand-tailored to all profiles used in the NaLiDa repository (*e.g.*, for the description of corpora and tools), and aims at limiting the loss of information for these profiles.

The metadata harmonization makes use of those conversions by first converting CMDI-based metadata to MARC 21. Then, a crosswalk from MARC 21 to EAD is being performed. This crosswalk is well documented and used in the library world [U4]. The conversion of discipline-specific metadata to generic bibliographic metadata profits from the use of authority records (Trippel and Zinn, 2016). The ingestion process at the library and computing centre will need to make use of the conversion service.

3.3. Persistent Identifier Management

Persistent identifiers can cause problems. While both repositories make use of Handle-based PIDs, they use different identifier prefixes and different local handle servers to resolve them. The NaLiDa-based PIDs are resolved using a local Handle-Server at the GWDG (Gesellschaft für wissenschaftliche Datenverarbeitung Göttingen), and the library and computing centre uses its own handle server with their own prefix (10900.1). This server will need to take over the resolving process for handles. Here, the new PIDs automatically created during the ingesting workflow at the library and computing centre's repository will need to be mapped to the existing PIDs that stem from the NaLiDa repository. The PID stemming from the GWDG will then point to the new library and computing centre based PID, which in turn will point to the corresponding research data in the library and computing centre. Note that such PID assignment involves a third party (at the GWDG) so that the PID mapping can only be partially mechanised.

4. Discussion and Conclusion

The authors are not aware of reported similar efforts in linguistics or related research areas. The migration of research data from one data repository to another is bound to take place from time to time in many institutions, but seems to get rarely reported and published. Also note that there is large degree of freedom that governs such enterprises. Readers who managed to migrate from, say the Fedora 3 repository system to its Fedora 4 successor will be aware of the many technical subtleties and intricacies involved, even if such migration is taking place within a single institution.[1] As a result, many design and migration decisions may well differ across institutions, which makes it hard to generalize. In this case study, we outlined the migration of research data from one data repository to another one. We assumed that all data is being migrated, and ignored a potential step to re-evaluate all data with regard to data obsolescence. The migration study profited from a common technological base as both archives used the Fedora Commons repository system. Still, there were issues that we needed to address.

Access restrictions, once imposed to research data, can become a significant hurdle for data migration. Here, we advocate a strong commitment to Open Data. Restricted access to data should be avoided, potentially at the cost of moratoria where restrictions must be lifted after a limited period of time. Ideally, legal agreements in favour of open access should be drafted when research data is deposited for the first time as it might be harder to amend any agreements at the time of the data migration.

The authentication and authorization infrastructure (AAI) that we described in this paper was limited to the level of the discipline-specific data centre at the institution level, or the wider university-wide level for RDM at the university library and computing centre. Ideally, digital repositories should support users from the outside, too. It should be possible, for instance, to make available resources with a "CC BY-NC-ND 4.0" license (permitting the non-commercial use of research data) to other researchers world-wide. Here, the migration of data to the more generic infrastructure will likely increase data accessibility as the university library and computing centre is in a better position to support an international authentication infrastructure such as `https://www.eduroam.org`.

The conversion of metadata formats can also be a challenging undertaking. Here, research institutions might have very expressive means to describe their research data, whereas library institutions often strictly adhere to bibliographic metadata standards such as MARC 21 or EAD. In our case, the information loss is significant as the conversion process went from CMDI to MARC 21, and from MARC 21 to EAD. Also, only the EAD description enters the library catalogue. To a large part, the value of a repository is rooted in the metadata that is used to describe its content. If the migration of research data implies a degradation of its corresponding metadata quality, then this is very unfortunate, in particular, when so much effort has been

[1] See `https://wiki.duraspace.org/display/FF/Training+-+Migrating+from+Fedora+3+to+Fedora+4`.

undertaken to describe research data is the most descriptive way possible. Here, we advocate to keep the rich original metadata attached to the digital object so that a maximum amount of information about the research data is preserved.

The migration of data from one data centre to another is a non-trivial undertaking. Migration costs can be significantly lowered when both data centres make use of good practices and standards. A common technological base eases the migration process considerably, but parties should be aware of issues such as access rights, metadata conversion, and the new resolving of persistent identifiers. There are other issues that might be taken into account, for instance, when the new data centre requires all research data to be bundled and ingested at a different level of granularity.

The migration of research data from one data centre to another needs to be carefully planned, and sufficient time and personnel should be allotted to ensure a smooth transition. From our description in Sect. 3, it should be clear that only parts of the migration process can be fully automated so that migration costs increase almost linearly with the number of digital objects to be migrated.

In an ideal world, the receiving end has a fully functional technical and organizational setup in place, but in reality many universities have only started to establish eScience centres that must cater for the needs of many different disciplines. When data migration must happen during the start-up of such an eScience centre, extra time must be allocated.

Also, be prepared that many smaller issues may materialize well after the actual migration. Here, data depositors might feel the most important service deterioration. When those researchers handed over their data to the discipline-specific data centre, they were given their data to colleagues they know, and despite well established workflows, their were informal communication channels were it was easily possible to amend data, metadata, or access rights. When research data is now managed at the infrastructural institution of the university, those informal settings are replaced by official routes. Here, the depositee is not both a linguist and archivist (who caters for the few), but just an archivist (who caters for the many). With greater distance to the respective discipline, it is likely that discipline-specific metadata schemes (such as CMDI profiles for speech corpora) will be superseded by generic bibliographic metadata. This makes it harder for others to find and evaluate research data for their studies. In consideration of this factor, it is advisable to choose the data-receiving archive with care. If possible, choose an archive that values and enforces rich metadata and which guarantees that all data and metadata are indexed in a widely known and searchable resource.

5. Acknowledgments

This work has been supported by the German Research Foundation (DFG reference no. 88614379), and the SFB 833 data management project INF (DFG reference no. 75650358). The data centre cooperates closely with the CLARIN-D centre in Tübingen which is funded by the German Federal Ministry of Education and Research (BMBF). We would like to thank the anonymous reviewers for their valuable feedback.

6. Web Resources

[U1] The Dublin Core Metadata Initiative, see www.dublincore.org.

[U2] The MARC 21 standard, see www.loc.gov/marc/bibliographic.

[U3] The EAD standard, see https://www.loc.gov/ead/.

[U4] The MARC to EAD crosswalk, see http://www.loc.gov/ead/ag/agappb.html#sec4.

[U5] The Handle system, see https://www.handle.net.

[U6] The Fedora repository platform, see http://fedorarepository.org.

[U7] ProAI, see http://proai.sourceforge.net.

[U8] The OAI-PMH protocol, see https://www.openarchives.org/pmh.

[U9] Apache Lucene and Solr, see http://lucene.apache.org/solr.

[U10] Docuteam packer, see https://www.docuteam.ch/en/products/it-for-archives/software.

[U11] The FAIR principles, see https://www.force11.org/group/fairgroup/fairprinciples.

[U12] Example of a deposit agreement (University of Reading, UK), see http://researchdata.reading.ac.uk/deposit_agreement.html

7. Bibliographical References

Dima, E., Henrich, V., Hinrichs, E., Hinrichs, M., Hoppermann, C., Trippel, T., Zastrow, T., and Zinn, C. (2012a). A Repository for the Sustainable Management of Research Data. In N. Calzolari, et al., editors, *Proceedings of the 8th. International Conference on Language Resources and Evaluation (LREC'12)*. ELRA.

Dima, E., Hoppermann, C., Hinrichs, E., Trippel, T., and Zinn, C. (2012b). A Metadata Editor to Support the Description of Linguistic Resources. In N. Calzolari, et al., editors, *Proceedings of the 8th. International Conference on Language Resources and Evaluation (LREC'12)*. ELRA.

ISO 24619. (2011). Language resource management - Persistent identification and sustainable access (PISA). International Standard.

ISO 24622-1. (2015). Language resource management – Component MetaData Infrastructure – Part 1: The Component Metadata Model. International Standard.

Lyse, G. I., Meurer, P., and Smedt, K. D. (2015). Comedi: A component metadata editor. *Selected Papers from the CLARIN 2014 Conference, Linköping University Electronic Press*, 116(8):82–98.

Trippel, T. and Zinn, C. (2016). Enhancing the quality of metadata by using authority control. In *5th. Workshop on Linked Data in Linguistic (LDL-2016) at LREC'16*.

Wilkinson, M. D. *et al.*. (2016). The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, 3(160018). http://dx.doi.org/10.1038/sdata.2016.18.

Zinn, C., Trippel, T., Kaminski, S., and Dima, E. (2016). Crosswalking from CMDI to Dublin Core and MARC 21. In N. Calzolari, et al., editors, *Proceedings of the 10th. International Conference on Language Resources and Evaluation (LREC'16)*. ELRA.

Introducing NIEUW: Novel Incentives and Workflows for Eliciting Linguistic Data

Christopher Cieri, James Fiumara, Mark Liberman, Chris Callison-Burch*, Jonathan Wright
University of Pennsylvania Linguistic Data Consortium and *Department of Computer and Information Science
{ccieri, jfiumara, myl, jdwright} @ldc.upenn.edu, {ccb} @cis.upenn.edu

Abstract

This paper introduces the NIEUW (Novel Incentives and Workflows) project funded by the United States National Science Foundation and part of the Linguistic Data Consortium's strategy to provide order of magnitude improvement in the scale, cost, variety, linguistic diversity and quality of Language Resources available for education, research and technology development. Notwithstanding decades of effort and progress in collecting and distributing Language Resources, it remains the case that demand still far exceeds supply for all of the approximately 7000 languages in the world, even the most well documented languages with global economic and political influence. The absence of Language Resources, regardless of the language, stifles teaching and technology building, inhibiting the creation of language enabled applications and, as a result, commerce and communication. Project oriented approaches which focus intensive funding and effort on problems of limited scope over short durations can only address part of the problem. The HLT community instead requires approaches that do not rely upon highly constrained resources such as project funding and can be sustained across many languages and many years. In this paper, we describe a new initiative to harness the power of alternative incentives to elicit linguistic data and annotation. We also describe changes to the workflows necessary to collect data from workforces attracted by these incentives.

Keywords: novel incentives, workflows, language resources

1. Introduction & Motivation

The Human Language Technology (HLT) community has benefitted from massive contributions of linguistic data from data centers, governments and groups around the world. Nevertheless, potential still remains largely untapped because the LRs that fuel development fall far short of need whether measured by volume, data type, or language coverage. Data centers regularly receive requests for data sets that they cannot supply even for the dozen languages with the greatest populations and *gross linguistic product*. A 2010 survey (METANET) of the Language Resources (LRs) required to build the HLTs that would support speakers of a given language working in the information age found that none of the language of the European Union, not even English, were fully supplied and warned that "*21 out of 30 European languages could become extinct in the digital world*". Beyond a few exceptional cases: Mandarin, Modern Standard Arabic, Japanese, language spoken primarily outside the EU, suffer even greater LR deficits.

Although the absence of language resources hinders the development of technologies to support international commerce, the problem becomes truly acute during emergent situations such as natural disasters and refugee crises. The International Association of Conference Interpreters warned in 2008: "*Ending a conflict and delivering emergency and humanitarian aid across language barriers represents a major challenge, for which few of the organisations entrusted with operations in the field are well equipped. This problem is compounded by the fact that there is a chronic shortage of interpreters in zones of crisis and war willing to work in the line of fire or in areas of natural disaster.*"

HLTs can make a critical contribution towards disaster relief as we see in the work of Verma et al. (2011) who were able to identify tweets that provided situational awareness with 80% accuracy using purpose built HLTs. However, their system required LRs not currently available for most of the world languages.

Some US government programs have begun to address the problem. DARPA LORELEI is developing technologies to provide situational awareness based on disaster related communications in low resource languages. Despite DARPA's track record for managing such common task projects to produce effective technologies, LORELEI's impressive array of resources will be available for at most a few dozen of the worlds 7000 languages.

Today's approaches to LR creation to support HLT development cannot hope to address world need. This is not only due to the total effort required to create these resources, but also due to the reliance on finite project-oriented funding and collection. Language resource developers and the HLT community must augment these efforts by rethinking the way we collect and annotate linguistic data, the incentives that we offer, the workforces who react to such incentives, the workflows that maximize the efficiency of such workforces and the downstream processing necessary to make the best use of the data and judgments that result from such new approaches.

2. An Incentives-Aware Model of Language Resource Creation

We envision any process used to collect linguistic data as comprised of the interaction of several components: task, incentives, workforce, workflow and processing. Each task has an inherent difficulty which may be mitigated by careful interface design. Task difficulty determines the level of education, skill set and commitment required of the humans who provide the data required to accomplish the task. At the same time the incentives offered attract different contributors who require custom workflows and interfaces in order to maximize their productivity. The tuple of task incentive, workforce and workflow produces an output that may again require customized processing before it can be exploited by a human language technology. Greenfield, Chan and Campbell (2016) make this clear when they write: "*While annotators who have been trained as professional linguists are able to annotate accurately and consistently from dense annotation guidelines, the amateur annotators who serve as workers on crowdsourcing platforms are not similarly motivated to create the best annotations possible*."

Additionally, data created specifically to support HLT research and development have mostly employed the single

incentive of monetary compensation. However, this approach is ineffective when there is a lack of funding or when potential data contributors are motivated by factors other than financial gain. Within the NIEUW project, we will consciously engineer incentives, workforces and workflows to produce output for specific purposes.

3. Incentives in Language Resource Development for HLT

Although the use of alternative incentives is not unknown, it is still relatively uncommon within HLT though some examples follow. Campbell (2016) describes multiple data collections that support research into the production of expressive speech. These efforts augmented monetary compensation with sustenance, curiosity, fun, access to recording equipment and data and unusual social opportunities such as human-robot interactions. The resulting data varied in many ways including regionalisms and emotive speech. In Mitsuzawa et al. (2016) consumers provide reviews via the Fuman Kaitori Center initially to communicate some dissatisfaction with a product or service but also to receive spendable credit based upon the size, quality and compliance of the review. The mixed incentives result in variation in the data such as duplications, marginally useful posts, varied spelling of named entities and inaccurate metadata that, the authors believed, must subsequently be corrected by expert annotators.

Crowd workers may be motivated by the quality of the interface design and the desire to maintain a high approval rating in a reputation market as well as any monetary incentives. By improving interface design, Greenfield, Chan and Campbell (2016) elicited higher quality data without having to offer highly compensated work which can tend to attract a mercenary element in some crowd working communities. Phrase Detectives (Poesio et al. 2016) offers the incentives of entertainment, interesting readings, a variable point system, experience levels, leaderboards, socializing and a lottery. The Great Language Game has elicited millions of language labels via the incentives of information, entertainment, competition and status.

Tyson et al. (2016) show that the difference between the corporate mission of About.com and the motivations of content creators leads to an end product that must be post-processed in order to add the proportion of cross document links optimal to sustain the site. The DialRC Center at CMU offers the novel incentives of access to information and the promise of an improved customer experience in real world transit interactions (Eskenazi et al., 2016).

4. Novel Incentives outside HLT

Although the HLT communities stands to benefit significantly from the use of novel incentives, one finds they are employed more frequently and effectively outside the community.

LibriVox[1] organizes volunteers who created audiobooks from out-of-copyright literary works and place them explicitly in the public domain. The initiative has created well more than 10,000 audiobooks recording more than 50,000 hours of speech in the process. Although most books are in English, at least 31 other languages are represented and the non-English material continues to grow. LibriVox recordings could support multiple HLTs including language and speaker recognition, speech-to-text and text-to-speech. Prahallad, Toth and Black (2007) developed text-to-speech systems from a dozen hours of LibriVox audio, compared those systems with more traditional data sources and concluded that "*a voice could be successfully built from large multi-paragraph speech using automatic segmentation tools*."

LibriVox volunteers make enormous contributions because they believe in the LibriVox mission, enjoy reading aloud, want to help maintain the art of storytelling, and enjoy collaborating. A small number eventually receive paid work through audiobook companies.

Other research disciplines collect data and judgments from a crowd attracted by non-monetary incentives. The citizen science site Zooniverse[2] has recruited more than one millions volunteers who contribute to more than 80 different projects by completing tasks such as identifying movement in star fields, classifying animal species, and transcribing museum records. Zooniverse's well-designed interfaces and highly tuned tasking are incentives for participation along with the motivations of learning and discovery, contributing to scientific advancement, interacting with a community who share these goals, and entertainment.

5. NIEUW Directions in Language Research Development

In order to help address the data needs mentioned above, LDC's NIEUW project is building tools to dramatically increase the store of LRs by employing techniques proven to work in multiple scientific disciplines and industry. NIEUW is supported by a 3-year Research Infrastructure grant from the U.S. National Science Foundation

Social media, games with a purpose, and citizen science have shown that human resources are effectively limitless for some activities. By creating an infrastructure that enables the ongoing construction of scalable data collection and annotation activities available to the public via the web and mobile devices and designed with appropriate incentive models, we will enhance LR development well beyond what project-dependent, direct funding alone can accomplish.

We recognize that in order to make the best use of non-traditional labor, we must offer a variety of incentives that are packaged into coherent clusters to appeal to large classes of potential contributors. It appears that several such overlapping communities already exist: 1) language students and professionals such as linguists, transcriptionists and translators who work directly with language data but would benefit from improved tools; 2) citizen scientists who are motivated to contribute to and participate in linguistic research and technology development; 3) game players who seek entertainment and competition. NIEUW is creating web-based portals for each of these communities populated with language collection and annotation activities that appeal to the respective communities through alternative incentives and task design strategies. The size and complexity of the tasks and activities will also be matched to the contributor community. Activities will initially be created by LDC and later by collaborators using a toolkit developed for the

[1] www.librovox.org

[2] www.zooniverse.org

project. By allowing researchers to develop their own activities, the infrastructure not only serves the larger research community but also creates a sustainable and growing resource for data creation not limited to any particular project goal. Unlike collection and annotation strategies created for specific projects and then allowed to lie fallow once the project's needs are met or the funding has been depleted, the NIEUW infrastructure is always available with little impediment to participation.

5.1 Language Professionals

NIEUW will create a portal dedicated to the specific needs, skills, and motivations of language professionals and students. In addition to having interest and expertise in language, professional linguists, language teachers and students can contribute data and judgments of great value to HLT research. Increasingly, language professionals and their students transcribe their own audio data in order to exploit big data approaches that leverage, for example, speech activity detection, forced alignment, and automatic vowel formant extraction. However, uneven availability of supporting infrastructure, data and tools hinders efficient use of time that could be dedicated to actual annotation and learning.

We will address this problem by adapting LDC web-based transcription tools for use within the Language Professionals portal. Teachers and other language professionals will be able to upload audio or select from LDC holdings including the Penn Sociolinguistic Archive comprised of ~6000 recordings made by William Labov and his students over the past 50 years across worldwide English speaking communities. Performance will be evaluated upon small amounts of audio for which gold standard transcriptions exist. The portal will track progress and accuracy which teachers can use to evaluate students. Where appropriate, speech activity detection, forced alignment, and phonetic classifiers will simplify the task and enhance the output which will be made available to researchers in standard formats.

Other planned activities include creating templates for deploying surveys of linguistic typology and eliciting related translations, including re-implementing the Afranaph surveys on the typology of African languages. Questions may include words or sentences requiring translation or questions whose answers are a combination of controlled vocabulary and example sentences.

5.2 Citizen Linguists

Public contributions to scientific research have a long history. Prominent early examples include Edmund Halley soliciting the public to help map solar eclipses in the eighteenth century (Pasachoff, 1999) and the Audubon Society's annual Christmas Bird Count which began in 1900 (Root, 1988). New digital technologies such as the internet, social media, and smart phones have increased the public's ability to engage in scientific research.

Our plans for a Citizen Linguist portal learn from the success of Zooniverse and similar efforts. Although many projects on Zooniverse involve classifying astronomical and zoological images, the success of transcription projects such as Shakespeare's World demonstrates the ability to crowdsource linguistic annotation with proper tasking, tool design, and incentives. Notwithstanding concerns about accuracy, contributions from citizen science can yield high quality results and achieve a throughput that far exceeds that of individual researchers or even small teams of experts.

A portal for citizen linguists attract similar levels of participation as other citizen science communities perhaps more as language is a common experience for nearly every human on the planet. A number of citizen linguist activities will require only native competence. Additionally, connections between language and identity make local pride and cultural preservation powerful incentives. Unified design and branding will encourage a dedicated community of participants. Contributors provide only user name and email during registration though some activities will require, for example demographical and attitudinal metadata which has been shown to correlate with linguistic variation. The portal will mediate access to multiple projects, research results, blogs, news and community fora and provide public recognition for participant contributions. While appealing graphics and simplified tasking and tool design to sustain contributor interest will be important, the portal will also rely on the contributors' desire to participate in language science and technology research, and in linguistic and cultural promotion and preservation.

Like all of the proposed portals, initial activities will be created by LDC with the future ability for researchers to create their own collection and annotation projects using the toolkit. Initial planned activities and tasks include the following.

There are a number of examples of projects that have used crowdsourcing to elicit contributions of recorded speech from participants across the globe. For example, Phonemica[3] preserves Chinese language and culture by collecting contributed stories in 12 different languages with English translations and a map that plots contributor's location. NIEUW will include activities w here citizen linguists record speech samples via computer or mobile device or by telephone via an 800 number.

Another citizen linguist activity will collect neurotypical control data for comparison with clinical populations. A variety of language tasks are among the tools used by clinicians to evaluate patients with Autism Spectrum Disorders or Neurodegenerative Disease. Thousands of hours of recordings are archived, but their value would be enhanced by an increase in matched neurotypical controls. NIEUW will implement versions of elicitation tasks used by the University of Pennsylvania Center for Autism Research as well as open source personality and speech production and perception tests to provide neurotypical controls for comparison to the clinical data.

LDC will create another activity that is a continuation of the GlobalTIMIT project. TIMIT (Garofalo et al., 1993) is the most popular corpus in the history of HLT and LDC has received numerous requests for TIMIT style corpora in other languages. In TIMIT, native speakers read 10 English sentences constructed to maximize the number and combination of sounds. In GlobalTIMIT, naturally occurring sentences replace the somewhat artificial phonetically rich sentences. To date we have applied this method to Thai, Mandarin, Non-Native English and Ga. NIEUW will include an activity with a simple to use

[3] http://phonemica.net

interface to record GlobalTIMIT sentences in many languages.

5.3 Games and Gamified Activities

Human computation and *games with a purpose* (GWAP) target online gamers who "generate useful data" while playing a fun game (Law and von Ahn, 2011). The NIEUW games portal will contain a variety of language games and gamified activities created in house and by collaborators. An early version of this portal, LingoBoingo,[4] is already available and currently includes the language games Phrase Detectives, Tile Attack, Zombilingo, and Jeux des Mots. LDC will have added its own Language ID game to the portal by the time this paper is published.

Inspired by the Great Language Game, our Language I.D. Game will ask players to listen to short audio clips and identify the language spoken. However, our version will improve tracking, language choices, educational potential and, the ability to collect new judgements. Main rounds of the game will present clips where the language is known and ask the player to choose from possible answers with the correct language always an option. These judgments both provide data about language confusability and also serve to demonstrate game player competence in correctly identifying specific languages. Using that knowledge, bonus rounds will be offered where the game player is presented clips of languages for which the correct answer is suspected, but not known. A variety of techniques such as voting algorithms or obtaining multiple judgments for each clip will yield accurate language identification for raw language data.

The game elements of the activity encourage participation. Players score points for every correct answer, but are eliminated after a certain number of incorrect answers. Scoreboards and advanced rounds encourage competition and continued play. As a player increases their score, the scoreboard changes from personal to daily to weekly and so on in order to encourage higher levels of play. Periodic feedback can meet multiple incentives by providing educational information that may also help players improve their score. This information may include summaries of languages provided by Ethnologue, diagrams of language family trees, and information relevant to making game judgements such as phonological properties of a given language.

We are currently advertising the LingoBoingo web portal on a variety of forums from Facebook and Twitter to Linguist List in order to develop brand recognition, solicit and maintain game player participation, and to develop relationships with collaborators who wish to include their own language games on the portal.

6. Infrastructure

Our experience in developing hundreds of language collection and annotation tasks and tools leads us to conclude that a great deal of the technical infrastructure needed to host annotation exercises shared common features whether the exercise appears superficially as a game or a serious working environment. Linguistic data is presented, selected and segmented, segments are classified and labels, annotation records are stored along with information concerning the source media, time and task and contributors. In addition, when multiple contributors make decisions concerning the same segment, it is necessary to determine how to model that variation. For example, should one expose the variation or use a voting algorithm or other method to reduce the variable answers to a single preferred answer.

Given the potential for sharing underlying infrastructure across traditional annotation, citizen science and language games, NIEUW will develop a toolkit for building data collection and annotation exercises inspired by LDC's WebAnn toolkit which has been used since 2011 by hundreds of LDC contributors to collect tens of millions of linguistic judgements. WebAnn is a single application that presents different GUIs to the user by reusing fixed components while granting control to an annotation task developer who is generally not a software developer. The application has continued to mature in its ability to allow managers to control their work from a redesigned layout manager for tool widgets to a more sophisticated assignment creation feature that tracks the managers' input and reports back on failures.

The NIEUW Toolkit will add support for annotating image and video and improve ease of use for activity designers especially when laying out GUIs, defining workflow, reporting progress and extracting stable corpora. Of particular importance, given the new contributor communities, will be algorithms for task assignment and the integration of variable responses into useable corpora. Planning for the popularity other efforts have experienced, we will also optimize for speed in responding to human actions. The resulting toolkit will accept data contributions that are keyboarded or uploaded and will deploy modules that connect to external data sources including our partners' user databases, social media accounts, smart phones and tablets, and LDC's telephone collection platforms. It will be capable of presenting media in whole or part, addressing segments of text by word or character offset, audio and video by time offset and image or video frame by polygons defined in Cartesian space. It will support annotation schemas involving free text, lists, controlled vocabularies or trees for complex taxonomies. Task designers who are not programmers will use tools to lay out GUIs, specify workflows and task assignments and select algorithms for converting annotations that include discrepancies into stable corpora.

The toolkit will also support tutorials, tests and the evaluation of contributor performance versus a gold standard or other contributors. Project managers will have access to tools for monitoring progress and adjusting task assignment. The NIEUW toolkit will include a suite of *gamification* components: progress meters, leaderboards and badges. To support the Citizen Linguist portal, it will include the capacity to add introductory material, blogs and message boards to each activity. The NIEUW version of WebAnn will be released as open source and data collected with this NSF funding will be released at no cost through the LDC.

7. Conclusion

This paper sketched the importance of incentives in language resource development and their impact on workforces, workflows and post-processing for a specific

[4] https://lingoboingo.org

HLT task. The community has experience with various monetary incentives and the necessity to condition found data. However, the HLT community has spent relatively little time trying to consciously engineer incentives and workflows. We described several instances above but believe the field now needs to benchmark its data creation scale and cost against external efforts that have been much more effective. Innovation in language resource creation, employing novel incentives, workforces and workflows is critical if the field is ever to seriously address the demand for HLTs for the world's languages.

8. Acknowledgements

The authors would like to acknowledge the support of the National Science Foundation via CISE Research Infrastructure (CRI) grants CRI CI-P 1629923 and CRI CI-NEW 1730377. We would also like to thank the participants in the workshops on novel incentives held at LREC 2016 and later that same year at LDC. Their contributions also made this paper possible.

9. Bibliographic References

Campbell, Nick. (2016). Herme & Beyond; the Collection of Natural Speech Data. 10th edition of the Language Resources and Evaluation Conference (LREC), 23-28 May 2016, Portorož Slovenia.

Erard, Michael. (2007). The Wealth of LibriVox: Classic texts, amateur audiobooks, and the grand future of online peer production, Reason 39:1, p. 46.

Eskenazi, Maxine, Sungjin Lee, Tiancheng Zhao, Ting Yao Hu, Alan W Black, Unconventional Approaches to Gathering and Sharing Resources for Spoken Dialog Research. 10th edition of the Language Resources and Evaluation Conference (LREC), 23-28 May 2016, Portorož Slovenia.

Garofolo, J., L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren and V. Zue. (1993). "TIMIT Acoustic- Phonetic Continuous Speech Corpus, LDC93S1," Linguistic Data Consortium, Philadelphia.

Greenfield, Kara, Kelsey Chan, Joseph P. Campbell, A Fun and Engaging Interface for Crowdsourcing Named Entities. 10th edition of the Language Resources and Evaluation Conference (LREC), 23-28 May 2016, Portorož Slovenia.

International Association of Conference Interpreters. (2008). Interpreting in Zones of Crisis and War: http://aiic.net/page/2979/interpreting-in-zones-of-crisis-and-war/lang/1

Kominek, John, Alan W Black. (2003). CMU ARCTIC databases for speech synthesis, CMU Technical Report CMU-LTI-03-177, Ver. 0.95, Pittsburgh, PA. Carnegie Mellon University.

Law, E. and L. von Ahn. (2011). Human Computation, San Rafeal, CA: Morgan & Claypool Publishers.

Lewis, M. Paul, Gary F. Simons, Charles D. Fennig (eds.). (2016). Ethnologue: Languages of the World, Nineteenth edition. Dallas, Texas: SIL International. Online version: http://www.ethnologue.com.

METANET. 2010. META-NET White Paper Series: Press Release, http://www.meta- net.eu/whitepapers/press-release-en, accessed March 16, 2016.

Mitsuzawa, Kensuke. Maito Tauchi, Mathieu Domoulin, Masanori Nakashima, Tomoya Mizumoto, FKC Corpus: a Japanese Corpus from New Opinion Survey Service.

Pasachoff, J. M. (1999). "Halley as an eclipse pioneer: his maps and observations of the total solar eclipses of 1715 and 1724," Journal of Astronomical History and Heritage, vol. 2, no. 1, pp. 39-54.

Poesio, Massimo, Jon Chamberlain, Udo Kruschwitz and Chris Madge, Novel Incentives for Phrase Detectives. 10th edition of the Language Resources and Evaluation Conference (LREC), 23-28 May 2016, Portorož Slovenia.

Prahallad, Kishore, Arthur R Toth, Alan W Black. 2007. Automatic Building of Synthetic Voices from Large Multi-Paragraph Speech Databases, Proceedings of Interspeech, Antwerp, Belgium.

Root, T. (1988). Atlas of Wintering North American Birds: An Analysis of Christmas Bird Count Data, Chicgo, IL: University of Chicago Press.

Székely, Éva, João P. Cabral, Peter Cahill, Julie Carson-Berndsen. 2011. Clustering Expressive Speech Styles in Audiobooks Using Glottal Source Parameters, Interspeech.

Tyson, Na'im, Jonathan Roberts, Jeff Allen, Matt Lipson, Evaluation of Anchor Texts for Automated Link Discovery in Semi-structured Web Documents. 10th edition of the Language Resources and Evaluation Conference (LREC), 23-28 May 2016, Portorož Slovenia.

Verma, S., S. Vieweg, W. J. Corvey, L. Palen, J. H. Martin, M. Palmer, A. Schram, and K. M. Anderson, "Natural Language Processing to the Rescue? Extracting "Situational Awareness" Tweets During Mass Emergency," in *Proceedings of International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.

Three Dimensions of Reproducibility in Natural Language Processing

K. Bretonnel Cohen[1,2], Jingbo Xia[3], Pierre Zweigenbaum[2], Tiffany J. Callahan[1], Orin Hargraves[4], Foster Goss[5], Nancy Ide[6], Aurélie Névéol[2], Cyril Grouin[2], and Lawrence E. Hunter[1]

[1]Computational Bioscience Program, University of Colorado School of Medicine
[2]LIMSI, CNRS, Université Paris-Saclay
[3]Huazhong Agricultural University
[4]University of Colorado Boulder
[5]Department of Emergency Medicine, University of Colorado
[6]Vassar College

Abstract

Despite considerable recent attention to problems with reproducibility of scientific research, there is a striking lack of agreement about the definition of the term. That is a problem, because the lack of a consensus definition makes it difficult to compare studies of reproducibility, and thus to have even a broad overview of the state of the issue in natural language processing. This paper proposes an ontology of reproducibility in that field. Its goal is to enhance both future research and communication about the topic, and retrospective meta-analyses. We show that three dimensions of reproducibility, corresponding to three kinds of claims in natural language processing papers, can account for a variety of types of research reports. These dimensions are reproducibility of a *conclusion,* of a *finding,* and of a *value.* Three biomedical natural language processing papers by the authors of this paper are analyzed with respect to these dimensions.

Keywords: methodology, reproducibility, repeatability, replicability, replicatability

1. Introduction

The journal *Language Resources and Evaluation* recently published an editorial on reproducibility in language processing. The editorial announced a new topic for the journal, named the associate editors for the topic, called for papers on the topic, and defined a number of relevant terms (Branco et al., 2017).

Before the editorial had appeared, the authors (who include two of the authors of this paper) had already submitted a correction: on further assessment of the literature, they had realized that they should reverse the definitions of two crucial terms (viz. *reproducibility* and *replicability*)[1]. It was a textbook example of publication of an analysis that turned out not to hold, and of how such an analysis should be handled.

It is especially striking that the topic of the correction was the definition of two rather central terms—failures of reproducibility have been so present in both the global scientific and the public consciousness that one could reasonably expect that there would be at least a broad consensus about the terminology that is used to discuss the phenomenon. And yet: even a cursory (and certainly a careful) review of the literature shows that no such consensus exists. Indeed, the community is not even *close* to a consensus. In definitions in the literature (multiple examples omitted from the abstract due to space constraints), one observes at least the following *three* terms used frequently to refer to the same *two* concerns: *replicability, repeatability,* and *reproducibility.* Less frequently, one also sees the terms *commensurate, valid,* and *validity,* and all of these can be involved in the definition of *rigor* (Kilicoglu, 2017). It is not uncommon to see *reproducibility* and *replicability* or *repeatability* used interchangeably in the same paper.

As to the things that may or may not be replicated or reproduced, the literature includes the *experiment* itself; specific *values,* measured or calculated; *findings*; *conclusions;* and confirmation or non-confirmation of a *hypothesis.* These terms that are used to define *reproducibility* are explicitly defined less frequently than are *reproducibility, replicability,* etc., so in the end, one is often not sure what, exactly, would count as "reproducible" or not, even in the presence of a definition. The situation is further complicated by the introduction of modifiers, e.g. to refer to *weak reproducibility.* Biological research literature often involves additional occasional confusion with the noun *replicate,* a count noun (one can have two or more replicates, or none) that refers to copies of a macromolecule under study (Vaux et al., 2012).

This lack of consensus definitions related to reproducibility is a problem because without them, we cannot compare studies of reproducibility. A number of such studies have appeared very recently, and in general, the results have been depressing. Multiple studies over the course of the past two years have reported widespread failures of reproducibility (Collaboration and others, 2015; Collberg et al., 2015). They range from unusually large-scale studies in psychology (Collaboration and others, 2015), to surprisingly large ones in computer science (Collberg et al., 2015), to case studies in natural language processing (Schwartz, 2010; Borgholt et al., 2015; Cohen et al., 2016; Gomes et al., 2016; Névéol et al., 2016; Cassidy and Estival, 2017; Kilicoglu, 2017; Mieskes, 2017). Yet, it is still quite difficult to get even a rough sense of the actual scale of the problem in natural language processing, because the lack of agreement about what exactly is being assessed makes it difficult to compare findings across papers on reproducibility issues.

[1]https://link.springer.com/article/10.1007%2Fs10579-017-9386-7

To address this problem of a lack of consensus definitions, this paper proposes a set of dimensions of reproducibility. Perhaps counter-intuitively, we first give the definition of *replicability* or *repeatability* that we assume in the paper:

- ***Replicability*** or ***repeatability*** is a property of an *experiment:* the ability to repeat—or not—the experiment described in a study.

Thus, we reserve the terms *replicability* or *repeatability* for the ability to repeat an experiment's methods (in the case of natural language processing, with the same data). As (Drummond, 2009) puts it, "replicability. . . means to exactly replicate the original experiment"—the *experiment* must be repeated, but whether or not the same values are obtained, the same findings emerge, or the same conclusion is reached is not relevant to the question of whether or not the experiment has been replicated. In the lexicon of (Goodman et al., 2016), which discusses reproducibility and replicability from the perspective of the broader field of both computational and non-computational science, this corresponds to *methods reproducibility*.

We differentiate between replicability or repeatability on the one hand, and reproducibility on the other. We propose the following:

- ***Reproducibility*** is a property of the *outcomes* of an experiment: arriving—or not—at the same conclusions, findings, or values.

With the disjunction *conclusions, findings, or values,* we see a likely cause of differing assessments of whether or not a "paper" has been reproduced: a subsequent study could *replicate or repeat* the *experiment* described in that paper, or vary the methodology in some small but interesting way, and arrive at the same or different values, *or* findings, *or* conclusions. Our proposal of three separate *dimensions of reproducibility* begins with the hypothesis that the lack of consensus about definitions of reproducibility is directly related to the fact that there are these multiple ways in which a paper might, or might not, be supported by subsequent work. Problems then arise when a single one of these dimensions is isolated and labelled as *reproducibility.* The result is essentially one of polysemy—with its attendant ambiguity.

To address this issue, the paper proposes a set of three things that we will refer to as *dimensions* of reproducibility, using here the sense of the word *dimension* as *one of a group of properties whose number is necessary and sufficient to determine uniquely each element of a system* [2]. The proposed dimensions are then evaluated with an adequacy test—we ask the question of whether or not publications in natural language processing can be mapped to the proposed dimensions. If a dimension were found not to be relevant to any paper in natural language processing, that would constitute evidence that it is not a valid dimension, or at least not a very useful one. On the other hand, if an analysis of publications in natural language processing resulted in very disparate aspects of papers being lumped into the same dimension, that would be consistent with the hypothesis that the dimension in question needed to be split into finer-grained categories.

To avoid punishing our colleagues for graciously sharing their findings with the community, we do this analysis with our own papers. This has the methodological shortcoming that it almost certainly introduces an element of bias into the analysis. For example, it is almost certainly the case that we assume things to be obvious in those papers that probably are not obvious to very many people outside of our own laboratories. On the other hand, it has the methodological advantage that we are intimately familiar with the papers, and so failures to find the relevant aspects of the papers in question are very unlikely to be due to not being familiar with the topic areas, or the methodologies, or the rationales behind the analyses. If they were not our own papers, all of these factors would certainly be possible confounds. In any case, we return to this methodological shortcoming in the Discussion section, and discuss some ways that it might be avoided in future work.

2. The proposed dimensions

We begin by establishing and constraining the scope of these dimensions.

- We exclude issues related to the ability to repeat the experiments reported in a paper. We define that above as *replicability* or *repeatability.*
- We take the unit of analysis as a paper. This could include conference papers, journal articles, or chapters in an edited volume. We exclude longer works, such as books, as well as shorter ones, such as published abstracts.

  There is an impressive amount of research going back to at least 1993 (Yentis et al., 1993) on the topic of subsequent publication of work that was originally presented at medical conferences in abstract form. It is clear from these publications that there are things related to reproducibility to be investigated in abstracts, as well (Yentis et al., 1993; Scherer et al., 1994; Marx et al., 1999; Sanders et al., 2001; Bydder et al., 2004; Byerly et al., 2000; Jackson et al., 2000; Oliver et al., 2003; Herbison, 2004; Ng et al., 2004; Autorino et al., 2006; Balasubramanian et al., 2006; Peng et al., 2006; Rao et al., 2006; Smollin and Nelson, 2006; Scherer et al., 2007; Dahllöf et al., 2008; Harel et al., 2011; Varghese et al., 2011). However, since they are not a common publication type in natural language processing and this is a paper about reproducibility in natural language processing, we do not have the requisite data to establish whether or not these dimensions apply to them.

Within this scope, then, we propose the following three dimensions of reproducibility:

1. Reproducibility of a **conclusion**.
2. Reproducibility of a **finding**.
3. Reproducibility of a **value**.

We now expand on those dimensions.

[2] Merriam-Webster.com, merriam-webster.com/ dictionary/dimension

2.1. Reproducibility of a conclusion

By *conclusion,* we mean a broad induction that is made based on the results of the reported research. Some examples of conclusions from our papers include:

- *The abstracts of scientific journal articles and the bodies of scientific journal articles have meaningfully different structural and linguistic characteristics.* This conclusion in our paper (Cohen et al., 2010), which was quite clearly stated—it actually formed the title of the paper—was important at the time (and presumably now) because it demonstrated the importance of what has since become a major theme in biomedical natural language processing. Prior to this, the majority of biomedical natural language processing papers treated only the abstracts of scientific journal articles; full text was becoming widely available, and this paper's conclusion supported the idea that there would be a crucial need for a very new research direction.
- *Clinical documents and scientific literature have very different distributional characteristics at multiple levels of negation.* This conclusion in the paper (Cohen et al., 2017a) appeared in the context of a recent paper by Wu et al. (Wu et al., 2014) that had concluded that negation was very much **not** a solved problem in natural language processing—despite the fact that many papers had suggested that it is—and that in order for it to *become* a solved problem in natural language processing, the way forward was not to annotate *more* data from the same genres in which it was already available, but to annotate data with *different distributional characteristics* from the data that was already available. Wu et al.'s conclusion was well-reasoned, but at the time it was difficult to act on it, as there wasn't actually much published data on what those distributions actually were. Our paper then showed some of the relevant characteristics of the distributions. In that context, our paper presented methodologies for doing apples-to-apples comparisons of the distributions of negation at both the phrasal and the sub-word (affixal) levels, as well as showing that those can vary completely independently.

  One might ask whether a "conclusion" is even capable of being reproduced. Data and objects can be reproduced, but to the extent that conclusions happen in people's minds, it is difficult to claim that they can be demonstrated to be the same.

  So, it is important to specify that by "conclusion," we mean an explicit statement in a paper. The fact that scientific papers often include a section labelled *Conclusions* should give even the most stalwart logical positivist (for a classic example in linguistics, consider Leonard Bloomfield (Bloomfield, 1936)) some confidence that such things exist.

2.2. Reproducibility of a finding

By *finding,* we mean a relationship between the values for some reported figure of merit with respect to two or more dependent variables. Two values could be equal—or not. These may be direct measurements (e.g. counts of true and false positives) or calculated numbers (e.g. a p-value less than some value for alpha, or not), but there must be a comparison involved. Findings of specific relationships between values—an F-measure higher with one classification algorithm than another, a strength of lexical association that is stronger in one genre than another—are at the heart of applications of natural language processing in the digital humanities (Moretti et al., 2008) and the essential starting point for natural-language-processing-based approaches to social science (Chateauraynaud, 2003; Née, 2017). More generally, they lie at the very heart of evaluation in natural language processing, where the most common trope is to compare the performance of one system as measured by some figure of merit to that of another (Resnik and Lin, 2010).

In contrast with a conclusion, a finding is a repeatable discovery, whereas a conclusion is not—it is instead a broader statement inferred (justifiably or not) from one or more findings. A finding deals with computable properties of some entity; a conclusion does not, but rather makes a statement that the findings support or lead to. Two papers could have the same *findings* but reach different *conclusions* based on those findings because the conclusions of a paper are based on an interpretation of its findings—two researchers might interpret a given set of findings quite differently. Some examples of findings from our papers include:

- *Explicit phrasal negation is normally distributed in the abstracts of scientific journal articles and in the bodies of scientific journal articles.* This finding, reported in (Cohen et al., 2017a), was important in the context of that paper because it constrained a central aspect of the methodology of the work—the statistical hypothesis tests that could be applied to the raw data.
- *Negation is normally distributed in scientific journal articles and in clinical documents.* The finding was derived from a statistical hypothesis test that showed that in the cases of both document types, the p-value of a Shapiro-Wilk test was less than 0.05. This finding in our paper is notable in this context because it is a clear example of a finding in the previous paper *not* being reproduced. The finding was even more significant in this paper than in the previous one, due to the motivation that was described above for this specific paper—the need to know the distributional characteristics of negation in a variety of types of biomedical text—in contrast to the previous paper, which studied textual characteristics of the genre more broadly.

In the lexicon of (Goodman et al., 2016), the dimension of reproducibility of a finding corresponds to *results reproducibility.*

2.3. Reproducibility of a value

By *value,* we mean a number, whether measured (e.g. a count of false positives) or calculated (e.g. a standard deviation). Actual values are important in finding constants, e.g. the coefficient of a long-tail distribution (see the extensive discussion of relevant topics for language in (Muller, 1977; Tweedie and Baayen, 1998; Baayen, 2001)), or the best smoothing value when calculating relative frequencies (Kilgarriff, 2012).)

Shannon's early work on the entropy of written English text provides an example of a language-related value that stimulated an enormous amount of academic work, some of which has been evaluated with respect to the extent to which it does or does not reproduce the values reported in (Shannon, 1951). For example, (Cover and King, 1978) used a very different method from Shannon's original one and found a value of 1.3 bits for the entropy of written English. The paper explicitly states that this value "agrees well with Shannon's estimate," suggesting that the authors considered their value to have reproduced Shannon's original value in (Shannon, 1951)[3]. In a very different tone, (Brown et al., 1992) reported an upper bound of exactly 1.75 bits, but did not explicitly compare that to previous findings, although it is clear from the paper that they considered it different from—and better than—previously reported values. As the authors put it:

> We see this paper as a gauntlet thrown down before the computational linguistics community.

A relevant value from our papers that was *not* reproduced is the mean value for the frequency of negation. We reported this in our papers (Cohen et al., 2010) and (Cohen et al., 2017a). They were different by roughly a factor of 2, even though we used the same corpus in both cases.

This is especially notable because the second of these papers is completely replicable, and yet we were later unable to reproduce our initial value. There was a doubly non-reproducible value here—the value in (Cohen et al., 2010) was not reproduced in (Cohen et al., 2017a), and that value in turn was not reproduced when we later repeated the experiment—all in our own papers.

The dimension of reproducibility of a value does not have an equivalent in the lexicon of (Goodman et al., 2016), perhaps because that paper points out a number of problems in determining whether or not studies have the "same" values, even when we can avail ourselves of statistical hypothesis tests.

3. Meta-analyses of some papers in natural language processing

3.1. Case study: A paper that was replicable but only partly reproducible

The motivation for this work came from a paper by Wu et al. (Wu et al., 2014) that discussed the potentially misleading nature of much recent work on negation in natural language processing of biomedical text. The contention of that paper was that in order to achieve reportable results that give a better estimate of the reality of performance on negation, one needs data with different distributional properties than the data that has been used in previous research on the topic. To address this, our paper undertook a study of two kinds of negation in two kinds of biomedical texts. We studied explicit phrasal negation (e.g. *is not involved in*) and sub-word or affixal negation (e.g. *unknown*) in biomedical journal articles and in physician progress notes on Intensive Care Unit patients from the MIMIC II database (Saeed et al., 2002). This was a quantitative study that did hypothesis tests on the rates of the two kinds of negation, finding that phrasal negation was more common in clinical texts than in scientific journal articles, while affixal negation was more common in scientific journal articles than in clinical texts—a surprising finding, given the relative amounts of research on negation in the two genres (much more on clinical text than on scientific text).

3.2. Why this example?

We select this study to illustrate the application of the proposed dimensions of reproducibility because it is the most heavily evaluated work, with respect to both replicability (the ability to repeat the *experiments*) and reproducibility (the *outcomes* of that experiment), that we have ever done. The reasons that we say that:

1. First, we made a very deliberate effort to archive *all* data and *all* code for this study on GitHub[4].
2. Second, we then had two trainees repeat the experiments, during the course of which one of the students found a bug in the analysis code, suggesting that they examined it quite closely.

In addition to the great effort that was made to ensure the replicability of this project—and we note that it appears to have been a very strong effort, as indicated by the fact that the student was able to repeat the experiment closely enough to locate a bug in the code—we had the opportunity to do a fortuitous assessment of the reproducibility of the work, because the night before giving a talk on this work, we found *another* bug in the code. That then gave us an opportunity to see whether or not the original conclusion, findings, and values would be *reproduced* after we fixed the bug.

So, this was a rather unusual piece of research, both in terms of the documented efforts that went into ensuring its replicability, and the unexpected opportunity to assess its reproducibility.

Additionally, there's this: the first author of the work is an associate editor for reproducibility issues in natural language processing. Three of the four authors of the paper are actively involved in research on reproducibility. It is difficult to say that they were not aware of how difficult of a problem this is, and it is clear from the GitHub repository (see Footnote 1) and from the replicability check that they had a student do that they were making a concentrated effort to ensure the reproducibility of the work.

3.3. What happened

The research was carried out, and the paper written, with no more stress or problems than one would expect. The analyses were all done in R markdown, as is often recommended in order to ensure the replicability of an analysis. (Gandrud, 2013; Leeper, 2014; Wickham and Grolemund, 2016). All data, code, and outputs of analyses were put in a publicly accessible GitHub repository, as is also often recommended

[3] We do not evaluate that claim ourselves because Shannon's paper actually reports a range of values; it is clear from the quotation that (Cover and King, 1978) felt that they reproduced Shannon's value, but it is not clear to us exactly how they came to that conclusion from the range of values in Shannon's 1951 paper.

[4] github.com/KevinBretonnelCohen/NegationDistribution

in order to ensure replicability (Pedersen, 2008). The paper was written and submitted to a large conference on biomedical informatics.

While the paper was under review, two trainees (one doctoral student and one post-doctoral fellow) were asked to check out the GitHub repository and repeat the work. This immediately led to two observations:

1. We had forgotten to upload two data files.
2. Replicating the analysis required editing some paths in the R code.

The two forgotten data files were then uploaded, and the fact that some editing of paths in the R code was required was duly noted. (The email chain that documents the chain of events here has been uploaded to the GitHub repository as a series of screen shots.) With that done, the trainees were able to replicate the analysis. (Note that we use the term *replicate* here because the analysis technique is a part of the experimental method, rather than one of the *outcomes* of the experiment.)

At that point, one of the trainees attempting to replicate the work noticed a bug in the R markdown file for calculating the inter-annotator agreement: two file names had been reversed. The code was fixed and the analysis was rerun. The calculated value for inter-annotator agreement changed, but the overall finding of relative incidence of negation still held, so neither this finding, nor the overall conclusion of the paper, was affected.

The paper was accepted for publication in the conference proceedings. The value for inter-annotator agreement in the paper that had been based on the incorrect file names was replaced with the correct value; again, neither the finding nor the conclusion was affected.

The night before giving the talk, one of the authors was finalizing the slides. Figuring that the most explicit way to demonstrate what had been done in counting the phrasal explicit negatives (e.g. *no* and *not*) would be to show the regular expression that had been used to detect them, he looked at the code in order to copy that regular expression into the slides. A sinking feeling ensued: he had written that code, he knew what he had intended for it to do, and it was not at all likely that the code in question had done it. The regular expression would work fine on the clinical data, which had been converted to one token per line in a preprocessing step. However, the scientific journal articles had not undergone this preprocessing, and the regular expression would need to have a global switch (which directs the regular expression engine to match as many times as the pattern occurs in the input, rather than just once) in order to do the count properly. Without that global switch, the code would only find at most one explicit phrasal negative in a line; since the corpus contained one paragraph per line, that meant that the code would find at most one explicit phrasal negative per *paragraph.*

Because all of the data and code was available on the GitHub site, repeating the counts and the subsequent analysis was literally a matter of about two minutes' work. When this was done, the following emerged:

- The counts of explicit phrasal negatives in the clinical documents had not changed; the counts of explicit phrasal negatives in the scientific journal articles had changed quite a bit. The mean frequency of explicit phrasal negation in the scientific journal articles was now much higher than it had been.
- Contrary to what we had published in the paper, the distribution of frequencies in the scientific journal articles was *not* normal. This meant that the t-test that had been used for hypothesis testing in the published version of the paper should not be used—rather, a non-parametric hypothesis test should have been used. Happily, when a Wilcoxon signed rank test was then done, the means were still significantly different, p = 2.138e-15.

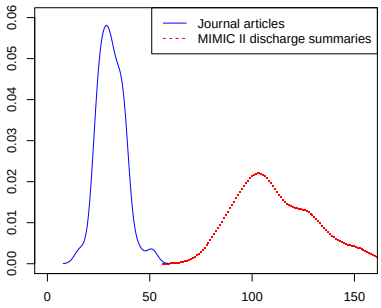


Figure 1: MIMIC II progress notes mean = 111 per 10,000-word sample, CRAFT corpus mean = **31** per 10,000-word sample. **Welch 2-sample t-test: t = -27.092, df = 53.822, p-value $<$ 2.2e-16**.

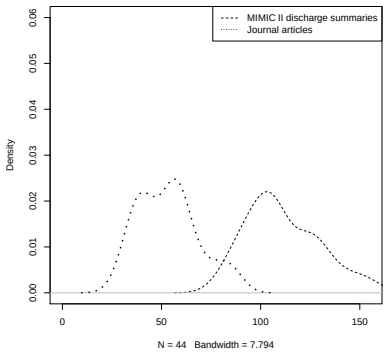


Figure 2: MIMIC II progress notes mean = 111 per 10,000-word sample, CRAFT corpus mean = **53** per 10,000-word sample. **Wilcoxon (Mann-Whitney U) W = 5.5, p-value = 2.138e-15**.

3.4. Analysis in terms of the three proposed dimensions of reproducibility

Working upwards from the most granular dimension to the most general one, we find:

Values One of the values was *not* reproduced. The mean for the scientific journal articles was much higher after the bug fix than before it. In contrast, the value for the clinical documents *was* reproduced.

Findings The finding that the frequencies of explicit phrasal negation in the scientific journal articles were normally distributed was *not* reproduced. In contrast, the finding that the mean of the frequencies of explicit

phrasal negation in the scientific journal articles was statistically significantly lower than the mean in the clinical documents *was* reproduced.

Conclusion Since of those two findings, only the finding that the mean of the frequencies of explicit negation in the scientific journal articles was lower than in the clinical documents was used to support the conclusion of the paper—that the distribution of explicit phrasal negation is different in the two genres—the conclusion *was* reproduced.

To summarize: the conclusion of the paper and the finding that was used to support that conclusion were both reproduced. The other finding, and the value that led to that finding, were not reproduced.

Metric	Before	After
Mean	31/10K words	53/10K words
Distribution	normal	bimodal

Table 1: Mean and distribution of explicit phrasal negation in scientific journal articles before and after fixing the bug. The *Before* values are the values published in (Cohen et al., 2017a). The *After* values are the values after we fixed the bug.

3.5. Case study: A paper on reproducibility whose conclusion is not reproducible

Cohen et al. (Cohen et al., 2016) published a case study on reproducibility that involved evaluating two R libraries for biomedical text mining. Both of those libraries provided connections to a web-based service. They concluded that reproducing the original work was difficult, but not impossible. Even before the paper went to press, that had ceased to be the case. As the authors put it:

> [T]wo hours after we submitted this paper for review, one of the libraries stopped working completely. . . . . the behavior [of the library] has not been the same since. . . and so far, we have been unable to reproduce our previous results. The research that relied on the library is at a standstill.

3.6. Case study: An attempt to reproduce an influential paper that was unable to reproduce its findings

Gomes et al. (Gomes et al., 2016) described a paper on their results when they replicated an influential approach to domain adaption. As the work is described, they were able to *replicate* the *methodology.* However, they were not able to reproduce the findings: the performance of their machine translation system did improve in one translation direction, but not in the other. In contrast, the original paper had shown improved performance in all cases that it examined. As the authors put it:

> While we were able to improve the Portuguese-to-English translation of in-domain texts using the. . . technique, the [method] did not outperform the in-domain trained baseline in the English-to-Portuguese direction.

This is an example of the dimension of reproducibility of a finding because it consists of a failure to reproduce the relative values of the system under test with respect to the baseline system. The dimension of reproducibility of a value is not relevant here because values were not being directly compared. Neither is the dimension of reproducibility of a conclusion relevant, since the paper does not make one beyond the statement about the findings.

4. Discussion and Conclusions

4.1. Is there really a problem here?

This paper is motivated by the claim of the existence of a problem of lack of consensus on terminology. Is there really such a problem? Related literature is consistent with the claim that there is. For example, a 2016 paper from one of the scientists most responsible for the recent notion of a "reproducibility crisis" in science notes that

> The language and conceptual framework of "research reproducibility" are nonstandard and unsettled across the sciences. . . As the movement to examine and enhance the reliability of research expands, it is important to note that some of its basic terms—reproducibility, replicability, reliability, robustness, and generalizability—are not standardized.

(Goodman et al., 2016)

Is that "problem in theory" really a "problem in practice"? Goodman et al. suggest that it is:

> This diverse nomenclature has led to confusion. . . about what kind of confirmation is needed to trust a given scientific result.

(Goodman et al., 2016)

In the field of natural language processing, work on the topic has concluded that such problems exist, as well. Ten years ago, (Pedersen, 2008) discussed the extent to which replicability and reproducibility issues go right to the heart of our field's claim to being an empirical science, and 9 years later, Olorisade et al. showed that the problem is still quite widespread (Olorisade et al., 2017). Fokken et al. showed that it is a difficult problem to address—the repercussions are grave (Fokkens et al., 2013).

As Goodman et al. point out, not knowing what kind of confirmation is needed to trust a given "scientific fact"—presumably, what we report in computational linguistics meetings and journals—has a very practical consequence. Not knowing what kind of confirmation is needed prevents us from operationalizing the solutions to the problems pointed out by writers on the topic of replicability and repeatability problems in natural language processing—we have no "clear operational criteria for what constitutes successful replication or reproduction" (Goodman et al., 2016). As we show in this paper in our case study on negation, the things that we often think are sufficient for ensuring replicability (e.g. shared data, the use of publicly available repositories to share our code, and markdown languages) are clearly not *in and of themselves* sufficient.

4.2. What is the origin of the problem?

Where does that lack of consensus come from, and does the source of the consensus tell us anything about the possible success (or lack thereof) of any proposal to address it?
On some level, we can trace the lack of consensus to a case of synonymy: the words *reproducibility* and *repeatability* are close enough to synonymous in general English that they often appear in each other's definitions. For example, in monolingual English dictionaries, we see:

- *replicate,* sense 3: *to repeat, duplicate, or reproduce, esp. for experimental purposes.* (Random House Unabridged 1999).

This is reflected in the very ways that scientists themselves define the terms when they write about the topics. For example:

> *. . .* ***reproducibility*** *means that the process of establishing a fact, or the conditions under which the same fact can be observed, is* ***repeatable****.*

(Teten, 2016), cited in (Atmanspacher et al., 2014)—our emphasis.
Previous work has established three things about reproducibility in natural language processing: it is important (Pedersen, 2008; Schwartz, 2010; Branco et al., 2017), it can be quite difficult to achieve (Fokkens et al., 2013; Névéol et al., 2016), and the causes of reproducibility problems can be well-hidden—see (Johnson et al., 2007; Cohen et al., 2017b), as well as the bug that we report in this paper.

4.3. Definitions of dimensions of reproducibility in the larger context of natural language processing

The bigger picture in which this work is situated is that of a lack of a fully developed epistemology of computational linguistics and natural language processing. Enormous advancements in this area have come from the shared task model of evaluation (Hirschman, 1990; Hirschman, 1994; Jones and Galliers, 1995; Resnik and Lin, 2010; Hirschman, 1998; Chapman et al., 2011; Huang and Lu, 2015), from the development of a science of evaluation in our field (Daelemans and Hoste, 2002; Voorhees et al., 2005; Buckley and Voorhees, 2017), and from the development of a science of annotation (Palmer et al., 2005; Ide, 2007; Wilcock, 2009; Pustejovsky and Stubbs, 2012; Stubbs, 2012; Styler IV et al., 2014; Bonial et al., 2017; Green et al., 2017; Ide and Pustejovsky, 2017; Savova et al., 2017). But, large holes remain in our development of an epistomology of computational linguistics and natural language processing that integrates these strengths of our field and also explores the relationships between natural language processing; computational and corpus linguistics; artificial intelligence, theoretical linguistics, and cognitive science (Cori et al., 2002). (See also (Cori and Léon, 2002) for a discussion of how issues of definition of what our field is affect that epistemology and (Bès, 2002; Habert and Zweigenbaum, 2002; Amblard, 2016) for how taxonomization of methodologies in natural language processing, computational linguistics, and engineering interact with it).

4.4. Novel observations

The work reported here allows some observations that to our knowledge have not been made before. First: reproducibility is not a binary, you-are-or-you-aren't condition—it is more nuanced than that, as can be seen from the examples of the dimensions, as well as from the extended case study.
Second: despite suggestions to the contrary, there are no "silver bullets" where reproducibility *or replicability* is concerned. The work that is described in the case study made heavy use of the most-commonly-advocated architectures for enhancing both replicability *and* reproducibility—and yet, we were initially not able to replicate the experiments. Once we could, we found that although the conclusion was reproducible, a crucial value and a key finding were not. The distinction between *replicability* and *reproducibility* that we make from the outset of this paper, along with the three dimensions of reproducibility that this paper proposes, allowed us to make these distinctions. A failure to distinguish between the ability to replicate an experiment and to reproduce its outcomes would not allow for a description of these circumstances, and a binary *reproduced/not reproduced* distinction would not allow us to do so, either.
We also note that trying to replicate the work was very productive—it led not only to discovering that some files were absent from the repository (which directly affects the repeatability of the work), but it led directly to the finding of the first bug. This might be surprising in the context of (Drummond, 2009)'s strong stance *against* the very notion that replicability is valuable. He makes the same distinction between replicability (the ability to repeat an experiment's methods) and reproducibility, and says the following about replicability in a paper titled *Replicability is not Reproducibility—Nor is it Good Science*:

> It would cause a great deal of wasted effort by members of our community. . . I am also far from convinced that it will deliver the benefits that many think it will. I suspect that, at best, it would serve as little more than a policing tool, preventing outright fraud.

The analysis of the case of our study on negation is a clear example of the success of what (Goodman et al., 2016) call "a proof-of-principle study. . . sufficient to show that [a phenomenon is] possible:" *pace* Drummond, our attempt to replicate an experiment improved our science.

4.5. Conclusions

We have shown examples from the natural language processing and computational linguistics literature of all three of the proposed dimensions of reproducibility—cases where conclusions, findings, and values were reproduced, and cases where they were not. We have also shown that the value for one dimension is not dependent on the others. For example, in the extended analysis of (Cohen et al., 2017a), we showed a case where a value was not reproduced and a finding was not reproduced, but the conclusion was. In the discussion of (Gomes et al., 2016), we showed a

case where a finding was not reproduced, but neither values nor conclusions were relevant to asking whether or not the earlier paper as a whole had been reproduced. In (Cohen et al., 2016), we see a paper whose conclusion is not reproducible, independent of specific values or findings. Taken together, these suggest that the proposed dimensions of reproducibility are, indeed, applicable to research in natural language processing. We have also shown how they map to definitions of the relevant phenomena in other work on reproducibility in science more broadly.

Moving forward, what can be done with the dimensions proposed in this paper that could not be done before? With this more nuanced set of definitions of reproducibility, we can better understand the state of the science in our field. Once we know what that state is, then we can build on the suggestions of papers like (Pedersen, 2008; Fokkens et al., 2013; Olorisade et al., 2017) that make concrete recommendations about dealing with issues of reproducibility in natural language processing—and make it better.

While we do this, we should be charitable to each other other, recognizing that failures of reproducibility can occur even in spite of the best intentions of the researchers. Facing our reproducibility problems will probably be painful for the field, but in the end, it will be of benefit to all of us, and to our science.

Acknowledgements

The work reported here was supported by NIH grants LM008111 and LM009254 to Lawrence E. Hunter. Cohen's work was supported by Hunter's grants, by grant AHRQ R21HS024541-01 to Foster Goss (as was Goss's), and by generous funding from Labex DigiCosme (project ANR11LABEX0045 DIGICOSME), operated by ANR as part of the program Investissement d'Avenir Idex Paris-Saclay (ANR11 IDEX000302), as well as by a Jean d'Alembert fellowship from the Fondation Campus Paris-Saclay as part of the Investissement d'Avenir program operated by ANR. TJC's work was supported by a supplement to NIH grant LM009254 to Hunter and Callahan. PZ and AN received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant 676207. The authors thank:

- Nicoletta Calzolari
- The participants in the workshop on *The role of text mining and image processing in fostering responsible research practices* at the 5th World Conference on Research Integrity

. . . for their insightful comments.

5. Bibliographical References

Amblard, M. (2016). Pour un TAL responsable. *Traitement Automatique des Langues*, 57(2):21–45.

Atmanspacher, H., Lambert, L. B., Folkers, G., and Schubiger, P. A. (2014). Relevance relations for the concept of reproducibility. *Journal of the Royal Society Interface*, 11(94):20131030.

Autorino, R., Quarto, G., Sio, M. D., Lima, E., Quarto, E., Damiano, R., Oliviero, R., Osorio, L., Marcelo, F., and D'Armiento, M. (2006). Fate of abstracts presented at the world congress of endourology: are they followed by publication in peer-reviewed journals? *Journal of endourology*, 20(12):996–1001.

Baayen, R. H. (2001). *Word frequency distributions*, volume 18. Springer Science & Business Media.

Balasubramanian, S., Kumar, I., Wyld, L., and Reed, M. (2006). Publication of surgical abstracts in full text: a retrospective cohort study. *The Annals of The Royal College of Surgeons of England*, 88(1):57–61.

Bès, G. G. (2002). La linguistique entre science et ingénierie. *Traitement Automatique des Langues*, 43(2):57–81.

Bloomfield, L. (1936). Language or ideas? *Language*, pages 89–95.

Bonial, C., Conger, K., Hwang, J. D., Mansouri, A., Aseri, Y., Bonn, J., O'Gorman, T., and Palmer, M. (2017). Current directions in English and Arabic PropBank. In *Handbook of Linguistic Annotation*, pages 737–769. Springer.

Borgholt, L., Simonsen, P., and Hovy, D. (2015). The rating game: Sentiment rating reproducibility from text. In *EMNLP*, pages 2527–2532, Lisbon, Portugal.

Branco, A., Cohen, K. B., Vossen, P., Ide, N., and Calzolari, N. (2017). Replicability and reproducibility of research results for human language technology: introducing an LRE special section.

Brown, P. F., Pietra, V. J. D., Mercer, R. L., Pietra, S. A. D., and Lai, J. C. (1992). An estimate of an upper bound for the entropy of english. *Computational Linguistics*, 18(1):31–40.

Buckley, C. and Voorhees, E. M. (2017). Evaluating evaluation measure stability. In *ACM SIGIR Forum*, volume 51, pages 235–242. ACM.

Bydder, S. A., Joseph, D. J., and Spry, N. A. (2004). Publication rates of abstracts presented at annual scientific meetings: how does the Royal Australian and New Zealand College of Radiologists compare? *Journal of Medical Imaging and Radiation Oncology*, 48(1):25–28.

Byerly, W. G., Rheney, C. C., Connelly, J. F., and Verzino, K. C. (2000). Publication rates of abstracts from two pharmacy meetings. *Annals of Pharmacotherapy*, 34(10):1123–1127.

Cassidy, S. and Estival, D. (2017). Supporting accessibility and reproducibility in language research in the Alveo virtual laboratory. *Computer Speech & Language*, 45:375–391.

Chapman, W. W., Nadkarni, P. M., Hirschman, L., D'Avolio, L. W., Savova, G. K., and Uzuner, O. (2011). Overcoming barriers to NLP for clinical text: the role of shared tasks and the need for additional creative solutions.

Chateauraynaud, F. (2003). *Prospéro. Une technologie littéraire pour les sciences humaines*. CNRS.

Cohen, K. B., Johnson, H. L., Verspoor, K., Roeder, C., and Hunter, L. E. (2010). The structural and content aspects of abstracts versus bodies of full text journal articles are different. *BMC bioinformatics*, 11(1):492.

Cohen, K. B., Xia, J., Roeder, C., and Hunter, L. (2016).

Reproducibility in natural language processing: A case study of two R libraries for mining PubMed/MEDLINE. In *Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*, pages 6–12, Portoroz, Slovenia.

Cohen, K. B., Goss, F., Zweigenbaum, P., and Hunter, L. E. (2017a). Translational morphosyntax: Distribution of negation in clinical records and biomedical journal articles. In *MEDINFO*, Hangzhou, China.

Cohen, K., NÃ©vÃ©ol, A., Xia, J., Hailu, N., Hunter, L., and Zweigenbaum, P. (2017b). Reproducibility in biomedical natural language processing. In *Proc AMIA Annu Symp*.

Collaboration, O. S. et al. (2015). Estimating the reproducibility of psychological science. *Science*, 349(6251):aac4716.

Collberg, C., Proebsting, T., and Warren, A. M. (2015). Repeatability and benefaction in computer systems research. *University of Arizona TR 14*, 4.

Cori, M. and Léon, J. (2002). La constitution du TAL. *Traitement Automatique des Langues*, 43(3):21–55.

Cori, M., David, S., and Léon, J. (2002). Pour un travail épistémologique sur le TAL. *Traitement Automatique des Langues*, 43(3).

Cover, T. and King, R. (1978). A convergent gambling estimate of the entropy of english. *IEEE Transactions on Information Theory*, 24(4):413–421.

Daelemans, W. and Hoste, V. (2002). Evaluation of machine learning methods for natural language processing tasks. In *3rd International conference on Language Resources and Evaluation (LREC 2002)*. European Language Resources Association.

Dahllöf, G., Wondimu, B., and Maniere, M.-C. (2008). Subsequent publication of abstracts presented at the international association of paediatric dentistry meetings. *International journal of paediatric dentistry*, 18(2):91–97.

Drummond, C. (2009). Replicability is not reproducibility: nor is it good science.

Fokkens, A., Van Erp, M., Postma, M., Pedersen, T., Vossen, P., and Freire, N. (2013). Offspring from reproduction problems: What replication failure teaches us. In *ACL (1)*, pages 1691–1701.

Gandrud, C. (2013). *Reproducible research with R and R studio*. CRC Press.

Gomes, L., van Noord, G., Branco, A., and Neale, S. (2016). Seeking to reproduce "Easy Domain Adaptation". In *Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*, page 1, Portoroz, Slovenia.

Goodman, S. N., Fanelli, D., and Ioannidis, J. P. (2016). What does research reproducibility mean? *Science translational medicine*, 8(341):341ps12–341ps12.

Green, M., Hargraves, O., Bonial, C., Chen, J., Clark, L., and Palmer, M. (2017). Verbnet/ontonotes-based sense annotation. In *Handbook of Linguistic Annotation*, pages 719–735. Springer.

Habert, B. and Zweigenbaum, P. (2002). Régler les règles. *Traitement Automatique des Langues*, 43(3):83–105.

Harel, Z., Wald, R., Juda, A., and Bell, C. M. (2011). Frequency and factors influencing publication of abstracts presented at three major nephrology meetings. *International archives of medicine*, 4(1):40.

Herbison, P. (2004). Full publication of abstracts of randomised controlled trials published at international continence society meetings'. *Neurourology and urodynamics*, 23(2):101–103.

Hirschman, L. (1990). Natural language evaluation. In *Speech and Natural Language*, Hidden Valley, Pennsylvania.

Hirschman, L. (1994). Human language evaluation. In *Human Language Technology*, Hidden Valley, Pennsylvania.

Hirschman, L. (1998). The evolution of evaluation: Lessons from the message understanding conferences. *Computer Speech & Language*, 12(4):281–305.

Huang, C.-C. and Lu, Z. (2015). Community challenges in biomedical text mining over 10 years: success, failure and the future. *Briefings in bioinformatics*, 17(1):132–144.

Ide, N. and Pustejovsky, J. (2017). *Handbook of Linguistic Annotation*. Springer.

Ide, N. (2007). Annotation science from theory to practice and use.

Jackson, K. R., Daluiski, A., and Kay, R. M. (2000). Publication of abstracts submitted to the annual meeting of the Pediatric Orthopaedic Society of North America. *Journal of Pediatric Orthopaedics*, 20(1):2.

Johnson, H. L., Cohen, K. B., and Hunter, L. (2007). A fault model for ontology mapping, alignment, and linking systems. In *Pacific Symposium on Biocomputing*, page 233, Maui, Hawaii. NIH Public Access.

Jones, K. S. and Galliers, J. R. (1995). *Evaluating natural language processing systems: An analysis and review*, volume 1083. Springer Science & Business Media.

Kilgarriff, A. (2012). Getting to know your corpus. In *International conference on text, speech and dialogue*, pages 3–15. Springer.

Kilicoglu, H. (2017). Biomedical text mining for research rigor and integrity: tasks, challenges, directions. *Briefings in Bioinformatics*.

Leeper, T. J. (2014). Archiving reproducible research with r and dataverse. *R Journal*, 6(1).

Marx, W. F., Cloft, H. J., Do, H. M., and Kallmes, D. F. (1999). The fate of neuroradiologic abstracts presented at national meetings in 1993: rate of subsequent publication in peer-reviewed, indexed journals. *American journal of neuroradiology*, 20(6):1173–1177.

Mieskes, M. (2017). A quantitative study of data in the NLP community. *EACL 2017*, page 1.

Moretti, F., Dobenesque, E., and Jeanpierre, L. (2008). *Graphes, cartes et arbres: modèles abstraits pour une autre histoire de la littérature*. Les prairies ordinaires.

Muller, C. (1977). *Principes et méthodes de statistique lexicale*. Classiques Hachette.

Née, E. (2017). *Méthodes et outils informatiques pour l'analyse des discours*. Presses universitaires de Rennes.

Névéol, A., Cohen, K., Grouin, C., and Robert, A. (2016).

Replicability of research in biomedical natural language processing: a pilot evaluation for a coding task. In *Proceedings of the LOUHI Seventh International Workshop on Health Text Mining and Information Analysis*, pages 78–84, Austin, Texas.

Ng, L., Hersey, K., and Fleshner, N. (2004). Publication rate of abstracts presented at the annual meeting of the American Urological Association. *BJU international*, 94(1):79–81.

Oliver, D., Whitaker, I., and Chohan, D. (2003). Publication rates for abstracts presented at the British Association of Plastic Surgeons meetings: how do we compare with other specialties? *British journal of plastic surgery*, 56(2):158–160.

Olorisade, B. K., Brereton, P., and Andras, P. (2017). Reproducibility of studies on text mining for citation screening in systematic reviews: Evaluation and checklist. *Journal of biomedical informatics*, 73:1–13.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Pedersen, T. (2008). Empiricism is not a matter of faith. *Computational Linguistics*, 34(3):465–470.

Peng, P. H., Wasserman, J. M., and Rosenfeld, R. M. (2006). Factors influencing publication of abstracts presented at the AAO-HNS Annual Meeting. *Otolaryngology-Head and Neck Surgery*, 135(2):197–203.

Pustejovsky, J. and Stubbs, A. (2012). *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. O'Reilly Media, Inc.

Rao, A. R., Beatty, J. D., Laniado, M., Motiwala, H. G., and KARIM, O. (2006). Publication rate of abstracts presented at the British Association of Urological Surgeons Annual Meeting. *BJU international*, 97(2):306–309.

Resnik, P. and Lin, J. (2010). Evaluation of NLP systems. *The handbook of computational linguistics and natural language processing*, 57:271–295.

Saeed, M., Lieu, C., Raber, G., and Mark, R. G. (2002). MIMIC II: a massive temporal ICU patient database to support research in intelligent patient monitoring. In *Computers in Cardiology, 2002*, pages 641–644. IEEE.

Sanders, D., Carter, M., Hurlstone, D., Lobo, A., and Hoggard, N. (2001). Research outcomes in British gastroenterology: an audit of the subsequent full publication of abstracts presented at the British Society of Gastroenterology. *Gut*, 49(1):154–155.

Savova, G., Pradhan, S., Palmer, M., Styler, W., Chapman, W., and Elhadad, N. (2017). Annotating the clinical text–MiPACQ, ShARe, SHARPn and THYME corpora. In *Handbook of Linguistic Annotation*, pages 1357–1378. Springer.

Scherer, R. W., Dickersin, K., and Langenberg, P. (1994). Full publication of results initially presented in abstracts: a meta-analysis. *Jama*, 272(2):158–162.

Scherer, R. W., Langenberg, P., Von Elm, E., et al. (2007). Full publication of results initially presented in abstracts. *Cochrane Database Syst Rev*, 2(2):MR000005.

Schwartz, L. (2010). Reproducible results in parsing-based machine translation: the JHU shared task submission. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 177–182, Uppsala, Sweden. Association for Computational Linguistics.

Shannon, C. E. (1951). Prediction and entropy of printed english. *Bell Labs Technical Journal*, 30(1):50–64.

Smollin, C. G. and Nelson, L. S. (2006). Publication of abstracts presented at 2001 NACCT. *Journal of Medical Toxicology*, 2(3):97–100.

Stubbs, A. (2012). Developing specifications for light annotation tasks in the biomedical domain. In *Third Workshop on Building and Evaluating Resources for Biomedical Text Mining*, page 71, Istanbul, Turkey.

Styler IV, W. F., Bethard, S., Finan, S., Palmer, M., Pradhan, S., de Groen, P. C., Erickson, B., Miller, T., Lin, C., Savova, G., et al. (2014). Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143.

Teten, H. (2016). *Enzyklopädie Philosophie und Wissenschaftstheorie: Bd. 3: G–Inn*. Springer-Verlag.

Tweedie, F. J. and Baayen, R. H. (1998). How variable may a constant be? measures of lexical richness in perspective. *Computers and the Humanities*, 32(5):323–352.

Varghese, R. A., Chang, J., Miyanji, F., Reilly, C. W., and Mulpuri, K. (2011). Publication of abstracts submitted to the annual meeting of the pediatric orthopaedic society of north america: is there a difference between accepted versus rejected abstracts? *Journal of Pediatric Orthopaedics*, 31(3):334–340.

Vaux, D. L., Fidler, F., and Cumming, G. (2012). Replicates and repeats—what is the difference and is it significant?: A brief discussion of statistics and experimental design. *EMBO reports*, 13(4):291–296.

Voorhees, E. M., Harman, D. K., et al. (2005). *TREC: Experiment and evaluation in information retrieval*, volume 1. MIT press Cambridge.

Wickham, H. and Grolemund, G. (2016). *R for data science: import, tidy, transform, visualize, and model data*. " O'Reilly Media, Inc.".

Wilcock, G. (2009). Introduction to linguistic annotation and text analytics. *Synthesis Lectures on Human Language Technologies*, 2(1):1–159.

Wu, S., Miller, T., Masanz, J., Coarr, M., Halgrim, S., Carrell, D., and Clark, C. (2014). Negation's not solved: generalizability versus optimizability in clinical natural language processing. *PloS ONE*, 9(11):e112774.

Yentis, S., Campbell, F., and Lerman, J. (1993). Publication of abstracts presented at anaesthesia meetings. *Canadian journal of anaesthesia*, 40(7):632–634.

Content-Based Conflict of Interest Detection on Wikipedia

Udochukwu Orizu, Yulan He
School of Engineering and Applied Science, Aston University, UK
{orizuus,y.he9}@aston.ac.uk

Abstract

Wikipedia is one of the most visited websites in the world. On Wikipedia, Conflict-of-Interest (CoI) editing happens when an editor uses Wikipedia to advance their interests or relationships. This includes paid editing done by organisations for public relations purposes, etc. CoI detection is highly subjective and though closely related to vandalism and bias detection, it is a more difficult problem. In this paper, we frame CoI detection as a binary classification problem and explore various features which can be used to train supervised classifiers for CoI detection on Wikipedia articles. Our experimental results show that the best F-measure achieved is 0.67 by training SVM from a combination of features including stylometric, bias and emotion features. As we are not certain that our non-CoI set does not contain any CoI articles, we have also explored the use of one-class classification for CoI detection. The results show that using stylometric features outperforms other types of features or a combination of them and gives an F-measure of 0.63. Also, while binary classifiers give higher recall values (0.81∼0.94), one-class classifier attains higher precision values (0.69∼0.74).

Keywords: Wikipedia, Conflict-of-Interest Detection, Bias, Stylometric features, One-class classification.

1. Introduction

A key feature of Wiki sites is to allow people from all over the world to add or modify articles anonymously and without consequence. This enables people with malicious intentions to use articles to promote or to discredit target products, services, organisations, or individuals.

Conflict of Interest (CoI) is defined as a situation in which a person or organisation is involved in multiple interests, financial interest, or otherwise; one of which could possibly corrupt the motivation of the individual or organisation[1]. According to Wikipedia, content on Wikipedia and other Wiki-media projects "must be written from a neutral point of view (NPOV)"[2]. NPOV refers to representing neutral and without bias all of the significant views that have been published by reliable sources. CoI editing happens when an editor contributes to Wikipedia about themselves or their relationships such as family, friends, clients, employers, and financial links, etc. Often times, CoI editing does not comply with NPOV.

CoI editing is strongly discouraged by Wikipedia as it undermines the public's confidence in it, and causes public embarrassment to the individuals being promoted. It is easy to assume that CoI is just bias; however while it is not possible for CoI to exist without bias, bias can often exist in the absence of a CoI. One's beliefs and desires can lead to biased editing, but that does not constitute a CoI.

The growth of Wikipedia makes it increasingly difficult for both Wikipedia users and administrators to manually monitor articles. Taking two example documents from our dataset, one is an article classified as CoI while the other is not:

- **CoI example:** *Kaizaad Kotwa, born in Mumbai, India, is an award winning professor and writer, actor, director, producer and designer. Currently he is a professor of theatre and film at Ohio State University in Columbus, Ohio. He recently won the Griffin Society Award for Best Professor and in 2007 was named one of the top professors in Ohio. He is the co-owner and co-Artistic Director of Poor-Box Productions, along with his mother Mahabanoo Mody-Kotwal, a famous actor, director and producer in India.*

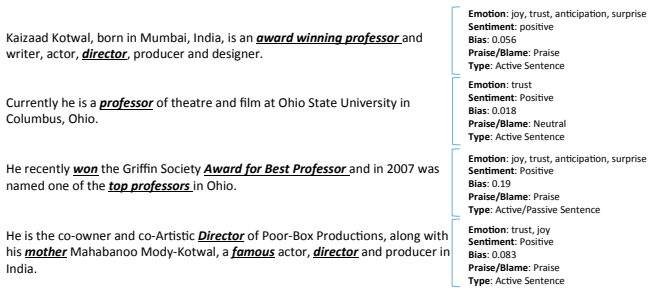
- **Non-CoI example:** *"Enrica Zunic" is the pseudonym of "Enrica Lozito", an Italian science-fiction writer. She lives and works in Turin. Her work is partly inspired by her activities with Amnesty International. In 2003 she won the Premio Italia award for science fiction.*

Using our proposed approach, a number of interesting features are identified as shown in Figure 1. It can be observed that the CoI example when compared to the non-CoI one contains more subjective sentences, bias sentences, emotion and more praise/blame expressions.

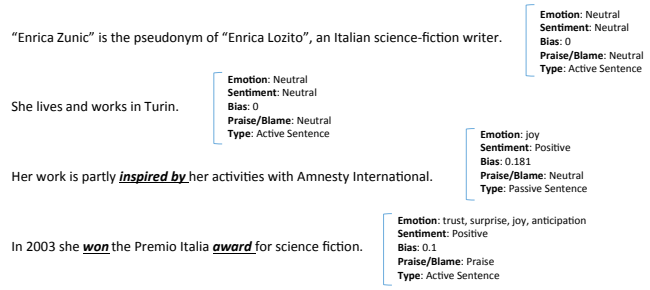
Our main aim in this work is to detect CoI articles based solely on the content of the articles without relying on any related metadata. We explore a rich set of features including stylometric features, the presentational features by focusing on the existence of Rhetorical Structure Theory's (RST's) presentational relations, various forms of language biases and implicit/explicit emotions. We then investigate using different combinations of features to train supervised binary classifiers for CoI detection. Our results show that the best result of 0.67 in F-measure is obtained when training Support Vector Machines (SVMs) from a combination of all features. Also, further combining various features with document-level representations either in the form of bag-of-words or dense representations by combining pre-trained word vectors does not bring any performance gains. As we only have the labeled CoI class, but not the non-CoI class, we have also explored the use of one-class classification for CoI detection. The results show that using stylometric features outperforms other types of features or a combination

[1] https://en.wikipedia.org/wiki/Conflict_of_interest

[2] https://en.wikipedia.org/wiki/Wikipedia:Neutral_point_of_view



(a) CoI example article



(b) Non-CoI example article

Figure 1: Two sample documents with features identified by our approach. Words/phrases underlined in text are those which be found in an emotion or sentiment lexicon. Due to space constraint, we only show some key features here such as the emotion/sentiment label, bias score, praise/blame indicator, and sentence type.

of them. Also, one-class classifier gives higher precision values compared to binary classifiers. To the best of our knowledge, we are the first to carry out automatic CoI detection on Wikipedia articles based solely on text content. Our main contributions are summarised below:

- We have built a CoI dataset which contains 3,280 CoI articles and 3,450 non-CoI articles, which could be used in future research on CoI detection;
- We have proposed a set of features based on our research of existing work close to CoI detection and analysis of the data collected and have identified the most effective features through extensive experiments on our CoI dataset;
- We have also investigated the effectiveness of using one-class classification for CoI detection.

The problem of content-based CoI has never been investigated before. We believe that our work will inspire further development of automated systems for CoI detection based on text content.

2. Related Work

There is no prior work on CoI detection from text. But CoI is closely related to vandalism and bias. As such, we review existing work on vandalism and bias detection from textual data, with focus on Wikipedia articles.

Vandalism can be defined as any modification of content made in a cautious effort to compromise the integrity of Wikipedia (West et al., 2010). Early tools consist of bots that would label vandalism using handcrafted rules encoding heuristic vandalism patterns. Such bots include ClueBot[3], MartinBot[4], etc. These bots' typical rules were limited and some of the features they examined include: the amount of text inserted or deleted, the ratio of capital letters, and the presence of vulgarisms detected.

Chin et al. (2010) looked at constructing statistical language models of an article from its revision history. According to their approach, if inappropriate content is added to the article, then the compression level is lower than it would be for text which is similar to existing content. This approach has a drawback that it tends to label as vandalism any large addition of material, regardless of its quality, while overlooking the small additions of vandalism. The idea of using reputation systems to aid in vandalism detection was advanced in (Zeng et al., 2006; Adler and De Alfaro, 2007). West et al. (2010) applied the concept of reputations to editors and articles. They proved that the broader use of meta-data can be very effective.

Potthast et al. (2010) presented a comprehensive overview of what types of features have been employed for vandalism detection. Early approach (Potthast et al., 2008) used manual inspection to construct a feature set based on meta data and content-level properties and built a classifier using logistic regression. They achieved 83% in precision and 77% in recall. Other similar machine learning approaches for vandalism detection include those proposed in (Smets et al., 2008; Itakura and Clarke, 2009; Mola-Velasco, 2012). Harpalani et al. (2011) hypothesised that vandalism edits have unique linguistic properties in common. They based their approach on stylometric analysis of vandalism edits using probabilistic context-free grammar models. Their approach outperformed features based on shallow patterns and achieved 77% in recall.

Recasens et al. (2013) analysed real instances of human edits designed to remove bias from Wikipedia articles. The analysis uncovers two classes of bias: framing bias, such as praising or perspective-specific words link to subjectivity; and epistemological bias, related to whether propositions that are presupposed or entailed in the text are undisputedly accepted as true. They found that features based on subjectivity and sentiment lexicons are very helpful in detecting bias. Callahan and Herring (2011) examined cultural bias based on Wikipedia's NPOV policy.

Bhosale et al. (2013) presented work on detecting promotional content in Wikipedia. They looked at the content features, structural features, network features, edit history features, overall sentiment score, trigram language models and PCFG language models. They found that the stylometric features influenced results the most.

When an edit is made on Wikipedia, the editor can either register for an account or edit anonymously. When done anonymously, Wikipedia uses the IP address to identify and distinguish the article instead of a username. WikiScanner or WikiWatchdog listed "anonymous" edits related to real-world organisations. They work by comparing a list of all IP addresses that have made edits to Wikipedia with IP

[3] https://en.wikipedia.org/wiki/User:ClueBot
[4] https://en.wikipedia.org/wiki/User:MartinBot

addresses which belong to real world organisations and returning a list of "anonymously" edited articles made from the organisations' IP addresses. Although WikiScanner or WikiWatchdog can be potentially used for CoI detection, they suffer from a number of limitations, for example, they don't analyse the content itself and don't consider edits done by registered users. Also, to avoid the detection by WikiScanner or WikiWatchdog, one would simply make an edit from a IP address not belonging to a real world organisation.

3. Our Approach

We address the CoI detection problem as binary classification which determines if a given document belongs to the category of CoI or non-CoI. We make the following hypotheses:

1. Since CoI is a sub-category of the "NPOV disputes" Wikipedia category, CoI articles inherit various linguistic and stylometric characteristics from their parent Wikipedia categories including those typically found in vandalism and bias;

2. CoI articles contain more subjective sentences than non-CoI articles;

3. The presentation of content in CoI articles will tend to increase the reader's interest/regard for the subject matter;

4. Since the choice of words projects opinions and preferences, CoI articles likely contain more expressions of implicit or explicit emotions.

In this section, we explore a rich set of features to test our hypotheses above and to train supervised classifiers for CoI detection.

3.1. Stylometric Features

Stylometric features attempt to recognise patterns of style in text. These techniques have been traditionally applied to attribute authorship (Reddy et al., 2016; Stamatatos, 2009; Argamon et al., 2009), opinion mining (Panicheva et al., 2010), and forensic linguistics (Turell, 2010; Olsson and Luchjenbroers, 2013). We create a list of features selected from previous research work in vandalism and bias as mentioned in the Related Work section. Since not all features are relevant to our CoI detection task, We perform feature selection using the implementation of InfoGain and Chi-Square available in Weka[5] to eliminate insignificant features. We also include the nine universal dependency groups[6], detection of which is done using the Stanford Dependency Parser[7]. The final set of features is listed in Table 1. This set of features is relating to Hypothesis 1.

3.2. Bias Features

In (Recasens et al., 2013), two major classes of bias in Wikipedia edits have been discussed, *framing bias* and *epistemological bias*. The former is realised by subjective words or phrases linked with a particular point of view, while the latter is related to linguistic features that subtly focus on the believability of a proposition. We use the same classes of bias as discussed in (Recasens et al., 2013) and identify existence of the classes in a Wikipedia article based on a bias lexicon[8]. We also consider other words/phrases which may introduce bias as illustrated in the Wikipedia's manual of style/Words to Watch[9]. The bias features are shown in Table 2. This set of features is relating to Hypothesis 1 and 2.

3.3. Presentational Features

Rhetorical Structure Theory (RST) is a discourse theory, which offers an explanation of the coherence of texts. It provides a way to describe the relations among text and has been used to successfully analyse a variety of text types (Taboada, 2006; Taboada and Mann, 2006). In RST, presentational relations are those whose intended effect is to increase some inclination in the reader or acceptance of the content (Mann and Thompson, 1987).
We focus our work on identifying the existence of presentational relations[10] using cue words as relation signals. We use 10 presentational relations as shown in Table 3, as they increase readers' acceptance of text in one form or the other. We built a simple cue phrase detector with phrases provided in various RST research (Taboada, 2006) and relation nucleus/satellite positioning described in (Mann and Thompson, 1987). This set of features is relating to Hypothesis 3.

3.4. Emotion Features

We focus on Ekman's six basic emotions (*joy, sadness, anger, surprise, fear, disgust*) and implement both explicit and implicit emotions detection. Emotions can be expressed explicitly by using "emotion-bearing words" or implicitly without such words. For explicit emotions, we use a simple lexicon-based approach with negation handling based on a modified version of the NRC lexicon (Mohammad and Turney, 2013); and for implicit emotions, we use the rule-based approach (Udochukwu and He, 2015). In addition, we also perform polarity detection (*positive* and *negative*) using majority voting based on the lexicon matching results obtained with three sentiment lexicons, SentiWordNet (Esuli and Sebastiani, 2005), AFINN (Hansen et al., 2011) and the Subjectivity Lexicon (Wilson et al., 2005). We implement a contextual valence shifter as described in (Polanyi and Zaenen, 2006) to detect polarity change in context. Apart from emotion and polarity features, we also consider the expressions of *blame* and *praise* as additional features using the method proposed in (Orizu and He, 2016)

[5] http://www.cs.waikato.ac.nz/ml/weka/

[6] http://universaldependencies.org/docsv1/u/dep/index.html

[7] http://nlp.stanford.edu/software/stanford-dependencies.shtml

[8] http://www.mpi-sws.org/~cristian/Biased_language.html

[9] https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Words_to_watch

[10] http://www.sfu.ca/rst/01intro/definitions.html

Feature Name	Description
Sentence Level	
Average_Sentence_Length	Average length of the sentences in the document
Average_Unique_Word_Count	Average # of unique words per sentence
Average_Punctuation	Average number of punctuations per sentence
Adjective_Rate	Rate of adjectives per sentence
CC_Rate	Rate of coordinating conjunctions per sentence
Pronouns_Rate	Rate of pronouns per sentence
Word_Count_Score	Total # of words / Total # of sentences
Unique_POS_per_Sentence	Rate of unique Part-of-Speech (POS) tags per sentence
Document Level	
Sentence_Count	Total # of sentences in the document
Unique_Word_Count	Total # of unique words in the document
No_of_Verbs	Total # of verbs in the document
No_of_CC	Total # of coordinating conjunctions in the document
No_of_CompAdverbs	Total # of comparative adverbs in the document
No_of_Adjectives	Total # of adjectives in the document
Special_clausal_dependents	Total # of special clausal dependents in the document
Active_Sentences	Total # of non-passive sentences
Non_core_dependents_of_clausal_predicates	Total # of non-core dependents of clausal predicates
Core_dependents_of_clausal_predicates	Total # of core dependents of clausal predicates
Noun_dependents	Total # of Noun dependents
Compounding_and_unanalyzed	Total # of Compounding and unanalyzed dependencies
Case-marking, prepositions, possessive	Total # of Case-marking, prepositions, possessive
Coordination	Total # of Coordination dependencies
Loose_joining_relations	Total # of loose joining relations
Sentence_head_and_Unspecified_dependency	Total # of Sentence head and Unspecified dependency
Complexity_Score	Text complexity score

Table 1: Stylometric features.

Bias	Subtypes
Epistemological	Factive verbs / Entailments / Assertives / Hedges
Framing	Subjective terms / Intensifiers
Others	Puffery / Contentious labels / Unsupported attributions / Expressions of doubt / Editorialising

Table 2: Bias features and subtypes.

Relation Name	Intention of W
Antithesis	R's positive regard for N is increased
Background	R's ability to comprehend N increases
Concession	R's positive regard for N is increased
Enablement	R's potential ability to perform the action in N increases
Evidence	R's belief of N is increased
Justify	R's readiness to accept W's right to present N is increased
Motivation	R's desire to perform action in N is increased
Preparation	R is more ready, interested or oriented for reading N

Table 3: The 10 presentational relations used (N stands for nucleus, R for reader and W for writer).

for detection. This set of features is relating to Hypothesis 4.

4. Experiments

4.1. Data

We construct our dataset by collecting 4,050 articles from Wikipedia which have been categorised as conflict of interest (CoI) items[11]. This CoI category is a sub-category of "NPOV disputes". Wikipedia encourages its editors to pick an article from this category and decide whether it meets its notability policy[12]. If one believes the article should be kept, he/she needs to review the text to ensure that it complies with NPOV. This human categorisation of Wikipedia articles will be our basis for evaluating our results.

In order to build a dataset containing both CoI and non-CoI articles, for each CoI article, we randomly select non-CoI articles from its first associated Wikipedia category. For example, a CoI article might be associated with two categories, "1932 births" and "Living people". We randomly select a non-CoI article from the category "1932 births". This resulted in a total of 4,600 non-CoI articles selected from over 100 Wikipedia categories. We have considered various criteria for the selection of non-CoI articles such as age of article, number of views, editor information. We found that identifying a threshold on these meta-data that cuts across the various categories and sectors would require a fine-tooth comb. For example, an article maybe older but has fewer views than a newer

[11] https://en.wikipedia.org/wiki/Category:Wikipedia_articles_with_possible_conflicts_of_interest

[12] https://en.wikipedia.org/wiki/Wikipedia:Notability

article OR articles from a particular category may have more views than other categories. As a result, we chose the random selection approach as long as the article was from the same category as a CoI article and did not belong to CoI disputes category. We focus on the article content as our means of classification and ignore the meta information provided by Wikipedia such as the editor(s) of a Wikipedia edit, time and date of creation, associated IP address, etc. The dataset is available at `https://www.dropbox.com/s/ivap8zvjpjymfrj/sql-import-coi-dataset.zip?dl=0`.

4.2. Preprocessing

We pre-process the dataset by removing the top 1% articles and the lower 5% of the articles based on the document length. This reduces the total number of document to 3,280 CoI articles and 3,450 non-CoI articles. The vocabulary size for the dataset is 52,302. We then carry out sentence splitting and tokenisation, stopword removal, stemming and remove words occurred less than ten times. For implicit emotion detection and blame/praise detection, we also perform part-of-speech (POS) tagging using the Stanford POS Tagger[13], word sense disambiguation (WSD) using the classic Lesk algorithm for WSD in NLTK[14], and dependency parsing using the Stanford Dependency Parser. To represent documents, apart from the commonly used bag-of-words approach, we also consider using doc2vec (Le and Mikolov, 2014) which modifies the word2vec algorithm (Mikolov et al., 2013) for unsupervised learning of continuous representations for larger blocks of text, such as sentences, paragraphs or entire documents. Recent work in the area of NLP has shown it to be a strong alternative for both bag-of-words and bag-of-n-grams models. We use Gensim[15] which has an implementation of doc2vec. We ignore words occurred less than 10 times and generate a vector representation of each article using the pre-trained vectors from the Google News dataset[16] with about 100 billion words, 300-dimensional vectors. The size of the context window we use is 3 before and after the predicted word. The final generated document vectors have 100 dimensions.

4.3. Feature Selection

Here we aim to identify the features that are mostly useful for prediction of CoI. We use Correlation-based Feature Subset Selection (CFS) and Information Gain Ratio (IGR) to rank features on all our feature sets from the training set and merge the top 15 features as listed in Table 4. Most of the top features are Stylometric features (74%) followed by the Emotion (21%) and Bias (5%) features. We also found that no Presentational features appear in the top 15 positions. The feature selection results indicate that stylometric features are very important in determining whether an article should be classified as CoI. Among various emotion features, *Blame*, *Praise*, *Polarity_Score* and *Suprise* seem more important than others. The *Bias_Score* is also relevant, but less important compared to many Stylometric or some Emotion features. Presentational features do not seem to contribute much to CoI detection.

Feature	Set	Description
Blame	*Emotion*	Total # of expressions of "Blame"
Praise	*Emotion*	Total # of expressions of "Praise"
Polarity_Score	*Emotion*	Aggregated polarity score of the document
Surprise	*Emotion*	Total # of expressions of "Surprise"
Active_Sentences	*Stylometric*	Total # of non-passive sentences
Non_core_dependents _of_clausal_predicates	*Stylometric*	Total # of non-core dependents of clausal predicates
Average_Sentence_Length	*Stylometric*	Average length of sentences in the document
Average_Unique_Word_Count	*Stylometric*	Average # of unique words per sentence
No_of_CC	*Stylometric*	Total # of coordinating conjunctions in the document
CC_Rate	*Stylometric*	Rate of coordinating conjunctions per sentence
Adjective_Rate	*Stylometric*	Rate of adjectives per sentence
Pronouns_Rate	*Stylometric*	Rate of pronouns per sentence
Sentence_Count	*Stylometric*	Total # of sentences in the document
Coordination	*Stylometric*	Total # of Coordination dependencies
Word_Count_Score	*Stylometric*	Total # of words / Total # of sentences
Unique_POS_per _Sentence	*Stylometric*	Rate of unique Part-of-Speech (POS) tags per sentence
Complexity_Score	*Stylometric*	Text complexity score
Special_clausal_dependents	*Stylometric*	Total # of special clausal dependents
Bias_Score	*Bias*	Aggregated bias score of the document

Table 4: Merged features from the feature selection results from Correlation-based Feature Subset Selection (CFS) and Information Gain Ratio (IGR).

4.4. Binary Classification Results

We train supervised classifiers including Support Vector Machines (SVMs), Maximum Entropy (MaxEnt) and Naïve Bayes (NB) using various feature sets and different combinations of them. Ten-fold cross validation is used and the results are averaged over 10 such runs.

We can observe from Table 5 that among the four feature sets, *Stylometric* gives the best performance followed by *Emotion*. This is consistent with our feature selection results discussed in Section 4.3.. It also confirms our hypothesis that the writing styles of editors of CoI articles are similar. *Bias* and *Presentational* features appear to be less useful. This shows that CoI is more than just bias. Presentational features had no member appeared in the top 20 features ranked by CFS or IGR. Although SVM or MaxEnt trained from *Presentational* or *Bias* features give much worse results compared to other feature sets, NB trained from these two types of features sets performs only slightly worse than trained from *Stylometric* or *Emotion* features.

We have also tried with combinations of different features sets. For both SVM and MaxEnt, the best performance is given by *All features*. SVM achieves much higher recall than precision with an overall F-measure of 0.67. MaxEnt gives more balanced precision and recall values, but with slightly worse F-measure compared to SVM. We also notice that using *Best features* as listed in Table 4 does not lead to improved performance for SVM or MaxEnt. However, the *Best features* set boosts the recall value to 0.94 for NB, although it only gives the precision value of 0.51.

We have next experimented with document representations using Bag-of-Words (BOW) weighted by TFIDF or doc2vec, and a combination of BOW or doc2vec with various feature sets. But they do not give any improvements, showing that CoI classification is not relevant to words presented in documents. Due to the page limit, we do not re-

[13] `http://nlp.stanford.edu/software/tagger.shtml`

[14] `http://www.nltk.org/howto/wsd.html`

[15] `http://radimrehurek.com/gensim/`

[16] `https://code.google.com/archive/p/word2vec/`

Feature Sets	SVM			MaxEnt			NB		
	P	R	F	P	R	F	P	R	F
Stylometric	0.57	0.74	0.64	0.60	0.61	0.60	0.50	0.91	0.65
Presentational	0.07	0.06	0.06	0.57	0.36	0.44	0.49	0.90	0.63
Bias	0.18	0.16	0.17	0.57	0.26	0.35	0.48	0.91	0.63
Emotion	0.26	0.16	0.24	0.56	0.39	0.46	0.49	0.92	0.64
Stylometric+Emotion	0.57	0.73	0.64	0.60	0.61	0.61	0.50	0.91	0.65
Stylometric+Emotion+Bias	0.57	0.73	0.64	0.61	0.61	0.61	0.50	0.91	0.65
All features	0.58	**0.81**	**0.67**	**0.64**	**0.67**	**0.66**	**0.62**	0.51	0.56
Best features	**0.61**	0.30	0.40	0.61	0.65	0.63	0.51	**0.94**	**0.66**

Table 5: Conflict-of-Interest (CoI) detection results in Precision, Recall and F-measure using SVM, MaxEnt and NB with various feature sets.

port the results here.

4.5. One-Class Classification

In Section 4.4., we train supervised classifiers from a dataset containing both CoI and non-CoI documents for binary classification. One problem we encountered is that there is no-degree of assurance that the items in our non-CoI category are purely non-CoI documents, as they where merely selected randomly from the same Wikipedia categories as CoI articles, with no concrete certainty that they are all non-CoI. Our problem could be potentially solved by one-class classification (Manevitz and Yousef, 2001; Khan and Madden, 2009), in which one of the target class is well represented by instances in the training data with little or no other class present. The problem of One-class classification is harder than the problem of conventional classification as a result of the one-sided nature of the dataset. One-class classification makes it difficult to decide which attributes should be used to best separate target and non-target (i.e., CoI and non-CoI in our case).

In (Schölkopf et al., 2001), adapting SVM to the one-class classification problem has been proposed. Essentially, the input data are first mapped into a high dimensional feature space via a kernel. The origin is considered as the only member of the second class. Then the algorithm iteratively finds the maximal margin hyperplane which best separates the training data from the origin. In our experiments here, we used one-class SVM implementation in the LIBSVM[17] with default parameters.

Feature Sets	Precision	Recall	F-Measure
Stylometric	**0.74**	**0.55**	**0.63**
Presentational	0.69	0.52	0.59
Bias	0.72	0.54	0.62
Emotion	0.73	0.55	0.62
All Features	0.72	0.53	0.61
Best features	0.73	0.54	0.62

Table 6: CoI detection results using one-class classification.

Table 6 shows the CoI detection results using one-class classification by 10-fold cross validation trained on the CoI-related documents only. It can be observed that using *Stylometric* features gives the best results compared to other feature sets although the improvement in F-measure compared to the *Bias* or *Emotion* features is only marginal. We also notice that the precision values, which are in the rage of 0.69 to 0.74, are much higher than those achieved based on binary classification where the typical precision values are between 0.58 and 0.64. However, the recall values are lower (0.52∼0.55 cf. 0.81∼0.94). This shows that if we aim to achieve high recall values for CoI detection, then binary classification should be used. However, if high precision values are more desirable, then one-class classification should be used instead.

4.6. Comparison with an Existing Approach to Vandalism Detection

There is no prior approach to content-based CoI detection from Wikipedia. Existing work to bias or vandalism detection often made use of metadata such as anonymity, edit frequency, author reputation, etc., and performed classification at the sentence-level. As we do not have the relevant metadata available and there are no sentence-level annotations in our dataset, directly comparing our approach with existing work is difficult. Nevertheless, we re-implemented an approach proposed in (Mola-Velasco, 2012) in which their best F-measure and AUC were achieved using LogitBoost and Random Forest, respectively, ranking in the first place of the PAN'10 Wikipedia vandalism detection task (Potthast et al., 2010). Since we do not have edit histories available, we exclude features relating to edit histories and only extract other stylometric features and features analogous to vulgarism frequency and vulgarism impact and train LogitBoost for 500 iterations. The results in comparison to our best ones are listed in Table 7. It can be observed that both our binary and one-class classifiers outperform LogitBoost with the performance gain in F-measure ranging from 6% to 10%.

Method	Precision	Recall	F-Measure
LogitBoost	0.56	0.58	0.57
SVM (binary)	**0.74**	0.55	0.63
SVM (one-class)	0.58	**0.81**	**0.67**

Table 7: Comparison with an existing approach to vandalism detection.

[17] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

4.7. Discussion

Our finding of the importance of stylometric features confirms our original hypothesis in Section 3. that CoI will inherit linguistic and stylometric features from its parent Wikipedia Category. But our hypothesis that presentation relations would affect CoI was not supported by our experimental results. We found that our hypothesis on CoI articles being more subjective holds true based on the experiment results. Also, the hypothesis that CoI articles contain more expressions of implicit and explicit emotions is also supported from our experimental results.

Our feature selection results show that *Blame*, *Praise* and *Polarity_Score* are discriminative features for the CoI class as they are ranked in the top 3 positions by CFS. However, in binary classification results, using features from the *Emotion* category gives worse results compare to the *Stylometric* category, although it outperforms both *Presentational* and *Bias* categories. The same observation holds for one-class classification. Using *Stylometric* features consistently outperform other feature sets for both binary and one-class classification. Also, it seems that articles with a higher rate of coordinating conjunctions and adjectives per sentence have a higher chance of belonging to the CoI category.

5. Conclusions and Future Work

The work presented here tackles a unique problem for the automatic detection of Conflict of Interest (CoI) articles in Wikipedia entries based on the content of the articles. We have shown that the CoI detection task is a complex problem but with carefully engineered feature sets, it is possible to identify CoI articles with an F-measure of 0.67 using SVM. We have also found that out of four different sets of features, *Stylometric* features help the most with CoI detection. In addition to binary classification, we have experimented with one-class classification and shown that while binary classification gives higher recall values, one-class classification attains higher precision values.

In future work, we intend to explore other types of features extracted from metadata of Wikipedia articles such as editors' information, editing history and associated IP addresses, and evaluate their impact on the performance of CoI detection. It is possible that articles in different Wikipedia categories might follow different writing styles (e.g., Wikipedia entries about people and about organisations). One possible direction is to build category-specific classifiers for CoI detection. Finally, to avoid expensive feature engineering, it is possible to learn feature representations and classifiers simultaneously by investigating various deep learning architectures.

6. Bibliographical References

Adler, B. T. and De Alfaro, L. (2007). A content-driven reputation system for the wikipedia. In *Proceedings of the 16th international conference on World Wide Web*, pages 261–270.

Adler, B. T., De Alfaro, L., Mola-Velasco, S. M., Rosso, P., and West, A. G. (2011). Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 277–288.

Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. (2009). Automatically profiling the author of an anonymous text. *Communications of the ACM*, 52(2):119–123.

Bhosale, S., Vinicombe, H., and Mooney, R. J. (2013). Detecting promotional content in wikipedia. In *EMNLP*, pages 1851–1857.

Callahan, E. S. and Herring, S. C. (2011). Cultural bias in wikipedia content on famous persons. *Journal of the Association for Information Science and Technology*, 62(10):1899–1915.

Chin, S.-C., Street, W. N., Srinivasan, P., and Eichmann, D. (2010). Detecting wikipedia vandalism with active learning and statistical language models. In *Proceedings of the 4th workshop on Information credibility*, pages 3–10.

Esuli, A. and Sebastiani, F. (2005). Determining the semantic orientation of terms through gloss classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 617–624.

Hansen, L. K., Arvidsson, A., Nielsen, F. Å., Colleoni, E., and Etter, M. (2011). Good friends, bad news-affect and virality in twitter. *Future information technology*, pages 34–43.

Harpalani, M., Hart, M., Singh, S., Johnson, R., and Choi, Y. (2011). Language of vandalism: Improving wikipedia vandalism detection via stylometric analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 83–88.

Itakura, K. Y. and Clarke, C. L. (2009). Using dynamic markov compression to detect vandalism in the wikipedia. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 822–823.

Khan, S. S. and Madden, M. G. (2009). A survey of recent trends in one class classification. In *Irish Conference on Artificial Intelligence and Cognitive Science*, pages 188–197.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196.

Manevitz, L. M. and Yousef, M. (2001). One-class svms for document classification. *Journal of Machine Learning Research*, 2(Dec):139–154.

Mann, W. C. and Thompson, S. A. (1987). *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013).

Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. 29(3):436–465.

Mola-Velasco, S. M. (2012). Wikipedia vandalism detection through machine learning: Feature review and new proposals: Lab report for pan at clef 2010. *arXiv preprint arXiv:1210.5560*.

Olsson, J. and Luchjenbroers, J. (2013). *Forensic linguistics*. A&C Black.

Orizu, U. and He, Y. (2016). Detecting expressions of blame or praise in text. In *the 10th edition of the Language Resources and Evaluation Conference*, Portoro, Slovenia, May, 2016.

Panicheva, P., Cardiff, J., and Rosso, P. (2010). Personal sense and idiolect: combining authorship attribution and opinion analysis. In *the 7th International Conference on Language Resources and Evaluation*, Malta, 2010.

Polanyi, L. and Zaenen, A. (2006). Contextual valence shifters. *Computing Attitude and Affect in Text*, 20:1–10.

Potthast, M., Stein, B., and Gerling, R. (2008). Automatic vandalism detection in wikipedia. In *European Conference on Information Retrieval*, pages 663–668.

Potthast, M., Stein, B., and Holfeld, T. (2010). Overview of the 1st international competition on wikipedia vandalism detection. In *CLEF (Notebook Papers/LABs/Workshops)*.

Recasens, M., Danescu-Niculescu-Mizil, C., and Jurafsky, D. (2013). Linguistic models for analyzing and detecting biased language. In *ACL (1)*, pages 1650–1659.

Reddy, T. R., Vardhan, B. V., and Reddy, P. V. (2016). A survey on authorship profiling techniques. *International Journal of Applied Engineering Research*, 11(5):3092–3102.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471.

Smets, K., Goethals, B., and Verdonk, B. (2008). Automatic vandalism detection in wikipedia: Towards a machine learning approach. In *AAAI workshop on Wikipedia and artificial intelligence: An Evolving Synergy*, pages 43–48.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the Association for Information Science and Technology*, 60(3):538–556.

Taboada, M. and Mann, W. C. (2006). Rhetorical structure theory: Looking back and moving ahead. *Discourse studies*, 8(3):423–459.

Taboada, M. (2006). Discourse markers as signals (or not) of rhetorical relations. *Journal of pragmatics*, 38(4):567–592.

Turell, M. T. (2010). The use of textual, grammatical and sociolinguistic evidence in forensic text comparison. *International Journal of Speech, Language & the Law*, 17(2).

Udochukwu, O. and He, Y. (2015). A rule-based approach to implicit emotion detection in text. In *International Conference on Applications of Natural Language to Information Systems*, pages 197–203.

West, A. G., Kannan, S., and Lee, I. (2010). Detecting wikipedia vandalism via spatio-temporal analysis of revision metadata? In *Proceedings of the Third European Workshop on System Security*, pages 22–28.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Zeng, H., Alhossaini, M. A., Ding, L., Fikes, R., and McGuinness, D. L. (2006). Computing trust from revision history. Technical report, Stanford Univ Ca Knowledge Systems LAB.

Word Affect Intensities

Saif M. Mohammad
National Research Council Canada
saif.mohammad@nrc-cnrc.gc.ca

Abstract

Words often convey affect—emotions, feelings, and attitudes. Further, different words can convey affect to various degrees (intensities). However, existing manually created lexicons for basic emotions (such as anger and fear) indicate only coarse categories of affect association (for example, associated with anger or not associated with anger). Automatic lexicons of affect provide fine degrees of association, but they tend not to be accurate as human-created lexicons. Here, for the first time, we present a manually created affect intensity lexicon with real-valued scores of intensity for four basic emotions: anger, fear, joy, and sadness. (We will subsequently add entries for more emotions such as disgust, anticipation, trust, and surprise.) We refer to this dataset as the *NRC Affect Intensity Lexicon*, or *AIL* for short. AIL has entries for close to 6,000 English words. We used a technique called best–worst scaling (BWS) to create the lexicon. BWS improves annotation consistency and obtains reliable fine-grained scores (split-half reliability > 0.91). We also compare the entries in AIL with the entries in the *NRC VAD Lexicon*, which has valence, arousal, and dominance (VAD) scores for 20K English words. We find that anger, fear, and sadness words, on average, have very similar VAD scores. However, sadness words tend to have slightly lower dominance scores than fear and anger words. The Affect Intensity Lexicon has applications in automatic emotion analysis in a number of domains such as commerce, education, intelligence, and public health. AIL is also useful in the building of natural language generation systems.

Keywords: emotion intensity, emotion lexicon, emotion analysis, crowdsourcing, best–worst scaling, sentiment analysis

1. Introduction

Words often convey affect—emotions, feelings, and attitudes. Some words have affect as a core part of their meaning. For example, *dejected* and *wistful* denotate some amount of sadness (and are thus associated with sadness). On the other hand, some words are associated with affect even though they do not denotate affect. For example, *failure* and *death* describe concepts that are usually accompanied by sadness and thus they connotate some amount of sadness. Lexicons of word–affect association have numerous applications, including: tracking brand and product perception, tracking support for issues and policies, tracking public health and well-being, literary analysis, and developing more natural dialogue systems. Past work on manually compiling affect lexicons has focused on denotative words (Wiebe et al., 2005; Francisco and Gervás, 2006; Strapparava and Valitutti, 2004). A notable exception to this is the NRC Emotion Lexicon, which includes words that are associated with (or connotate) an emotion (Mohammad and Turney, 2013).

Words can be associated with different intensities (or degrees) of an emotion. For example, most people will agree that the word *outrage* is associated with a greater degree of anger (or more anger) than the word *irritate*. However, existing manually created affect lexicons for basic emotions such as anger and fear do not provide scores for the intensity of the emotion. Annotating instances for fine-grained intensity of affect is a substantially more difficult undertaking than categorical annotation. It is particularly hard to ensure consistency (both across responses by different annotators and within the responses produced by the same annotator).

Best–Worst Scaling (BWS) is an annotation scheme that addresses these limitations by employing comparative annotations (Louviere, 1991; Cohen, 2003; Louviere et al., 2015; Kiritchenko and Mohammad, 2017). Annotators are given n items at a time (an n-tuple, where $n > 1$ and commonly $n = 4$). They are asked which item is the *best* (highest in terms of the property of interest) and which is the *worst* (least in terms of the property of interest). When working on 4-tuples, best–worst annotations are particularly efficient because each best and worst annotation will reveal the order of five of the six item pairs (i.e., for a 4-tuple with items A, B, C, and D, if A is the best, and D is the worst, then A $>$ B, A $>$ C, A $>$ D, B $>$ D, and C $>$ D).

We can calculate real-valued scores of association between the items and the property of interest from the best–worst annotations for a set of 4-tuples (Orme, 2009; Flynn and Marley, 2014). The scores can be used to rank items by the degree of association with the property of interest. It has been empirically shown that three annotations each for $2N$ 4-tuples is sufficient for obtaining reliable scores (where N is the number of items) (Louviere, 1991; Kiritchenko and Mohammad, 2016).[1] Kiritchenko and Mohammad (2017) showed through empirical experiments that BWS produces more reliable and more discriminating scores than those obtained using rating scales.

Here, for the first time, we create an affect intensity lexicon with real-valued scores of association for four basic emotions (anger, fear, joy, and sadness) using best–worst scaling. For a given word and emotion X, the scores range from 0 to 1. A score of 1 means that the word conveys the highest intensity (amount) of emotion X. A score of 0 means that the word conveys the lowest intensity (amount) of emotion X. We will refer to this lexicon as the *NRC Affect Intensity Lexicon (AIL)*. AIL includes entries for close to 6,000 English words. It includes common English terms as well as terms that are more prominent in social media platforms, such as Twitter. It includes terms that are associ-

[1] At its limit, when $n = 2$, BWS becomes a *paired comparison* (Thurstone, 1927; David, 1963), but then a much larger set of tuples need to be annotated (closer to N^2).

ated with emotions to various degrees. For a given emotion, this even includes some terms that may not predominantly convey that emotion (or that convey an antonymous emotion), and yet tend to co-occur with terms that do. Antonymous terms tend to co-occur with each other more often than chance, and are particularly problematic when one uses automatic co-occurrence-based statistical methods to capture word–emotion connotations. Thus, it is particularly beneficial to have manual annotations of affect intensity for these terms.

We show that repeat annotations of the terms in the Affect Intensity Lexicon with independent annotators lead to affect association scores that are close to the scores obtained originally (Spearman Rank correlations of 0.92; Pearson correlation: 0.91). The fine-grained scores obtained with BWS and the high correlations on repeat annotations indicate that BWS is both markedly discriminative (helps identify small differences in affect intensity) and markedly reliable (provides stable outcomes).

We also compare the entries in AIL with the entries in the *NRC VAD Lexicon*, which has valence, arousal, and dominance (VAD) scores for 20K English words. We find that anger, fear, and sadness words, on average, have very similar VAD scores. However, sadness words tend to have slightly lower dominance scores than fear and anger words. The Affect Intensity Lexicon has applications in automatic emotion analysis in a number of domains such as commerce, education, intelligence, and public health. AIL is also useful in the building of natural language generation systems. We have made the NRC Affect Intensity Lexicon freely available for, non-commercial, research purposes.[2]

We begin with a brief overview of the related work (Section 2), followed by a description of how we created the NRC Affect Intensity lexicon (Section 3). In Section 4, we study the valence, arousal, and dominance scores of words in the Affect Intensity Lexicon. In Section 5, we present experiments on the reliability of the annotations. In Section 6, we outline various applications of the NRC Affect Intensity lexicon. Finally, in Section 6, we present concluding remarks.

2. Related Work

Psychologists have argued that some emotions are more basic than others (Ekman, 1992; Plutchik, 1980; Frijda, 1988; Parrot, 2001).[3] Thus, most work on capturing word–emotion associations has focused on a handful of emotions, especially since manually annotating for a large number of emotions is arduous. In this project, we focus on four emotions common among the many proposals for basic emotions (Plutchik, 1980; Ekman, 1992; Parrot, 2001): anger, fear, joy, and sadness.

There is a large body of work on creating valence or sentiment lexicons, including the General Inquirer (Stone et al., 1966), ANEW (Nielsen, 2011; Bradley and Lang, 1999), MPQA (Wiebe et al., 2005), NRC VAD Lexicon by (Mohammad, 2018), and the lexicon by Warriner et al. (2013). The work on creating lexicons for categorical emotions such as joy, sadness, fear, etc, is comparatively small. WordNet Affect Lexicon (Strapparava and Valitutti, 2004) has a few hundred words annotated with the emotions they evoke.[4] It was created by manually identifying the emotions of a few seed words and then marking all their WordNet synonyms as having the same emotion. The NRC Emotion Lexicon was created by crowdsourcing and it includes entries for about 14,000 words and eight Plutchik emotions (Mohammad and Turney, 2013; Mohammad and Turney, 2010).[5] It also includes entries for positive and negative sentiment.

Most prior work in sentiment analysis describes machine learning systems trained and tested on data with coarse categorical annotations. This is not surprising, because it is difficult for humans to directly provide valence (sentiment) scores at a fine granularity. A common problem is inconsistencies in annotations among different annotators. One annotator might assign a score of 7.9 to a word, whereas another annotator may assign a score of 6.2 to the same word. It is also common that the same annotator assigns different scores to the same word at different points in time. Further, annotators often have a bias towards different parts of the scale, known as *scale region bias*. Despite this, a key question is whether humans are able to distinguish affect at only four or five coarse levels, or whether we can discriminate across much smaller affect intensity differences.

Best–Worst Scaling (BWS) was developed by Louviere (1991), building on some ground-breaking research in the 1960's in mathematical psychology and psychophysics by Anthony A. J. Marley and Duncan Luce. However, it is not well known outside the areas of choice modeling and marketing research. Within the NLP community, BWS has thus far been used for creating datasets for relational similarity (Jurgens et al., 2012) and word-sense disambiguation (Jurgens, 2013). Mohammad (2018) used best–worst scaling to annotate about 20K words for valence, arousal, and dominance. In this work, we use BWS to annotate words for intensity (or degree) of basic emotions. With BWS we address the challenges of direct scoring, and produce more reliable emotion intensity scores. Further, this will be the first dataset that will also include emotion scores for words common in social media.

There is growing work on automatically determining word–emotion associations (Mohammad and Kiritchenko, 2015; Mohammad, 2012; Strapparava and Valitutti, 2004). These automatic methods often assign a real-valued score representing the degree of association. Further, these association scores are likely to be somewhat correlated with the intensity of the emotion. The Affect Intensity Lexicon can be used to judge the quality of the automatic lexicons, and also to explore the extent of correlation between emotion association and emotion intensity.

[2] www.saifmohammad.com/WebPages/AffectIntensity.htm

[3] However, they disagree on which emotions (and how many) should be classified as basic emotions—some propose 6, some 8, some 20, and so on.

[4] http://wndomains.fbk.eu/wnaffect.html

[5] http://www.purl.org/net/saif.mohammad/research

3. NRC Affect Intensity Lexicon

We now present how we created the NRC Affect Intensity Lexicon. The two sub-sections below describe how we chose the terms to be annotated and how we annotated the chosen terms, respectively.

3.1. Term Selection

We chose to annotate commonly used English terms, as well as terms common in social media texts, so that the resulting lexicon can be applied widely. Twitter has a large and diverse user base, which entails rich textual content.[6] Tweets have plenty of non-standard language such as emoticons, emojis, creatively spelled words (*happee*), hashtags (*#takingastand, #lonely*) and conjoined words (*loveumom*). Tweets are often used to convey one's emotions, opinions towards products, and stance over issues. Thus, emotion analysis of tweets is particularly compelling. Therefore, apart from common English terms, we also chose to annotate terms common in tweets.

Since most words do not convey a particular emotion to a marked degree, annotating all words for all emotions is sub-optimal. Thus, for each of the eight emotions, we created separate lists of terms that satisfied either one of the two properties listed below:

- The word is already known to be associated with the emotion (although the intensity of emotion it conveys is unknown).
- The word has a tendency to occur in tweets that express the emotion.

With these properties in mind, for our annotation, we included terms from two separate sources:

- The words listed in the NRC Emotion Lexicon that are marked as being associated with any of the Plutchik emotions.
- The words that tend to co-occur more often than chance with emotion-word hashtags in a large tweets corpus. (Emotion-word hashtags, such as *#angry, #fear,* and *#happiness*, act as noisy labels of the corresponding emotions.)

Since the NRC Emotion Lexicon (Mohammad and Turney, 2013; Mohammad and Turney, 2010) includes only those terms that occur frequently in the Google n-gram corpus (Brants and Franz, 2006), these terms satisfy the 'commonly used terms' criterion as well.

The Hashtag Emotion Corpus (Mohammad, 2012) has tweets that each have at least one emotion-word hashtag. The emotion-word hashtags corresponding to the eight basic Plutchik emotions. As mentioned before, we consider the emotion-word hashtags as (noisy) labels of the corresponding emotions. For every word that occurred more than ten times in the corpus, we computed the pointwise mutual information (PMI) between the word and each of the emotion labels. If a word has a greater-than-chance tendency to occur in tweets with a particular emotion label, then it will have a PMI score that is greater than 0. For each emotion, we included all terms in the Hashtag Emotion Corpus (Mohammad, 2012) that had a PMI > 1. Note that this set of terms included both terms that are more common in social media communication (for example, *soannoyed, grrrrr, stfu*, and *thx*) as well as regular English words.[7]

3.2. Annotating for Affect Intensity with Best–Worst Scaling

For each emotion, the annotators were presented with four words at a time (4-tuples) and asked to select the word that conveys the highest emotion intensity and the word that conveys the lowest emotion intensity. $2 \times N$ (where N is the number of words to be annotated) distinct 4-tuples were randomly generated in such a manner that each word is seen in eight different 4-tuples, and no two 4-tuples had more than two items in common. We used the script provided by Kiritchenko and Mohammad (2016) to obtain the 4-tuples to be annotated.[8] A sample questionnaire is shown below.

Words Associated With Most And Least Anger

Words can be associated with different degrees of an emotion. For example, most people will agree that the word *condemn* is associated with a greater degree of anger (or more anger) than the word *irritate*. The goal of this task is to determine the degrees of anger associated with words. Since it is hard to give a numerical score indicating the degree of anger, we will give you four different words and ask you to indicate to us:

- the word that is associated with the MOST anger
- the word that is associated with the LEAST anger

A rule of thumb that may be helpful is that a word associated with more anger tends to occur in many angry sentences, whereas a word associated with less anger tends to occur in fewer angry sentences.

Important Notes

- Some words, such as *furious* and *irritated*, are not only associated with anger, they also explicitly express anger. Others do not express anger, but they are associated with the emotion; for example, argument and corruption are associated with anger. To be selected as 'associated with MOST anger' or 'associated with LEAST anger', a word does not have to explicitly express anger.
- Some words have more than one meaning, and the different meanings may be associated with different degrees of anger. If one of the meanings of the word is strongly associated with anger, then base your response on that meaning of the word.

---

[6] Twitter is an online social networking and microblogging service where users post and read messages that are up to 140 characters long. The posts are called tweets.

[7] Some of the terms included from tweets were deliberate spelling variations of English words, for example, *bluddy* and *sux*.

[8] http://saifmohammad.com/WebPages/BestWorst.html

Dataset	#words	Location of Annotators	Annotation Item	#Items	#Annotators	MAI	#Q/Item	#Best–Worst Annotations
anger	1,483	USA	4-tuple of words	2,966	119	4	2	12,212
fear	1,765	USA	4-tuple of words	3,530	82	4	2	14,129
joy	1,268	USA	4-tuple of words	2,536	76	4	2	10,365
sadness	1,298	USA	4-tuple of words	2,596	76	4	2	10,429
Total	**5,814**							**47,135**

Table 1: Summary details of the current annotations done for the NRC Affect Intensity Lexicon. MAI = median number of annotations per item. Q = questions.

EXAMPLE

Q1. Identify the term associated with the MOST anger.

- tree
- grrr
- boiling
- vexed

Ans: boiling

Q2. Identify the term associated with the LEAST anger.

- tree
- grrr
- boiling
- vexed

Ans: tree

The questionnaires for other emotions are similar.

We setup four crowdsourcing tasks corresponding to the four basic emotions. The 4-tuples of words were uploaded for annotation on the crowdsourcing platform, CrowdFlower.[9] We obtained annotations from native speakers of English residing in the United States of America. Annotators were free to provide responses to as many 4-tuples as they wished. The annotation tasks were approved by the National Research Council Canada's Institutional Review Board, which reviewed the proposed methods to ensure that they were ethical.

About 5% of the data was annotated internally beforehand (by the author). These questions are referred to as gold questions. The gold questions are interspersed with other questions. If one gets a gold question wrong, they are immediately notified of it. If one's accuracy on the gold questions falls below 70%, they are refused further annotation, and all of their annotations are discarded. This serves as a mechanism to avoid malicious or random annotations. In addition, the gold questions serve as examples to guide the annotators.

Kiritchenko and Mohammad (2016) showed that using just three annotations per 4-tuple produces highly reliable results. In task settings, we specified that we needed annotations from four people for each word.[10] However, because of the way the gold questions work in CrowdFlower, they were annotated by more than four people. Nonetheless, the median number of annotations is four (same as the minimum number of annotations). A total of 47,135 pairs of responses (best and worst) were obtained (see Table 1).

[9] http://www.crowdflower.com

[10] Note that since each word occurs in eight different 4-tuples, each word is involved in $8 \times 4 = 32$ best–worst judgments.

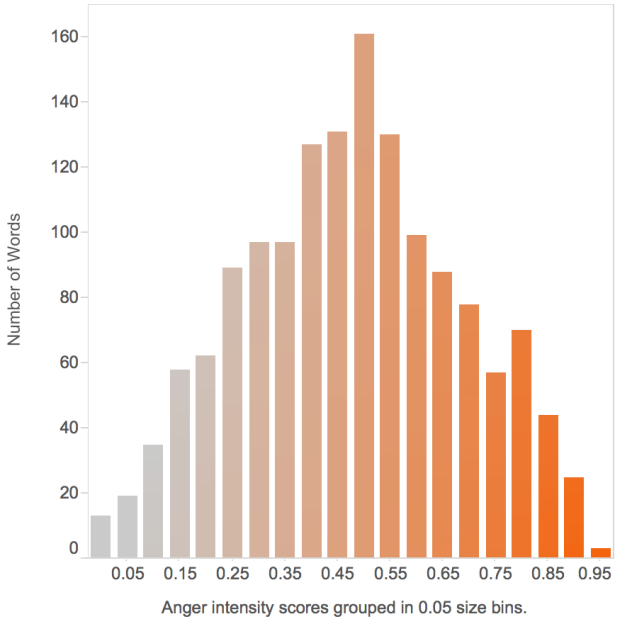


Figure 1: A histogram of word–anger intensities. Anger intensity scores are grouped in bins of size 0.05. The colors of the bars go from gray to orange in increasing order of affect intensity.

Annotation Aggregation: The intensity scores were calculated from the BWS responses using a simple counting procedure (Orme, 2009; Flynn and Marley, 2014): For each item, the score is the proportion of times the item was chosen as having the most intensity minus the proportion of times the item was chosen as having the least intensity. The scores range from -1 (least emotion intensity) to 1 (the most emotion intensity). Since degree of emotion is a unipolar scale, we linearly transform the -1 to 1 scores to scores in the range 0 (least emotion intensity) to 1 (the most emotion intensity). We refer to the full list of words along with their real-valued scores of affect intensity as the *NRC Affect Intensity Lexicon*.

Distribution of Scores: Figure 1 shows a histogram of word–anger intensities. The words are grouped into bins of scores 0–0.05, 0.05–0.1, and so on until 0.95–1. Observe that the intensity scores have a normal distribution. The histograms for other emotions have a similar shape.

Table 1 gives a summary of the number of items annotated and the number of annotations obtained. Table 2 shows some example entries from the lexicon. The lexicon is made freely available.

Word	Anger	Word	Fear	Word	Joy	Word	Sadness
outraged	0.964	*horror*	0.923	*sohappy*	0.868	*sad*	0.844
brutality	0.959	*horrified*	0.922	*superb*	0.864	*suffering*	0.844
satanic	0.828	*hellish*	0.828	*cheered*	0.773	*guilt*	0.750
hate	0.828	*grenade*	0.828	*positivity*	0.773	*incest*	0.750
violence	0.742	*strangle*	0.750	*merrychristmas*	0.712	*accursed*	0.697
molestation	0.742	*tragedies*	0.750	*bestfeeling*	0.712	*widow*	0.697
volatility	0.687	*anguish*	0.703	*complement*	0.647	*infertility*	0.641
eradication	0.685	*grisly*	0.703	*affection*	0.647	*drown*	0.641
cheat	0.630	*cutthroat*	0.664	*exalted*	0.591	*crumbling*	0.594
agitated	0.630	*pandemic*	0.664	*woot*	0.588	*deportation*	0.594
defiant	0.578	*smuggler*	0.625	*money*	0.531	*isolated*	0.547
coup	0.578	*pestilence*	0.625	*rainbow*	0.531	*unkind*	0.547
overbearing	0.547	*convict*	0.594	*health*	0.493	*chronic*	0.500
deceive	0.547	*rot*	0.594	*liberty*	0.486	*injurious*	0.500
unleash	0.515	*turbulence*	0.562	*present*	0.441	*memorials*	0.453
bile	0.515	*grave*	0.562	*tender*	0.441	*surrender*	0.453
suspicious	0.484	*failing*	0.531	*warms*	0.391	*beggar*	0.422
oust	0.484	*stressed*	0.531	*gesture*	0.387	*difficulties*	0.421
ultimatum	0.439	*disgusting*	0.484	*healing*	0.328	*perpetrator*	0.359
deleterious	0.438	*hallucination*	0.484	*tribulation*	0.328	*hindering*	0.359

Table 2: Example entries for four emotions in the NRC Affect Intensity Lexicon. For each emotion, the table shows every 100th and 101st entry, when ordered by decreasing emotion intensity.

4. Relationships of the Basic Emotions with Valence, Arousal, and Dominance

Even though the basic emotions model has long enjoyed the attention of psychologists, the valence–arousal–dominance (VAD) model (Russell, 2003) is also widely accepted. According to the VAD model of affect, individual emotions are points in a three-dimensional space of valence (positiveness–negativeness), arousal (active–passive), and dominance (dominant–submissive). Both the basic emotions model and the VAD model offer different perspectives that help our understanding of emotions. However, there is little work relating the two models of emotion with each other. Much of the past work on textual utterances such as sentences and tweets, is based on exactly one or the other model (not both). For example, corpora annotated for emotions are either annotated only for the basic emotions (Strapparava and Mihalcea, 2007; Mohammad and Bravo-Marquez, 2017b) or only for valence, arousal, and dominance (Yu et al., ; Mohammad et al., 2017; Nakov et al., 2016). Mohammad and Kiritchenko (2018) created the first dataset of tweets manually annotated for multiple affect dimensions from both the basic emotion model and the VAD model. For each emotion dimension, they annotated the data for coarse classes (such as no anger, low anger, moderate anger, and high anger) and also for fine real-valued scores indicating the intensity of emotion (anger, sadness, valence, etc.). They present an analysis of emotion intensities of tweets and their relationship with valence.

Similar to the situation for textual corpora, *words* have been annotated largely either just for valence, arousal, and dominance (*ANEW* (Bradley and Lang, 1999), the *Warriner Lexicon* (Warriner et al., 2013), and the *NRC VAD Lexicon* (Mohammad, 2018)) or just for association with basic emotions (the NRC Emotion Lexicon (Mohammad and Turney, 2013; Mohammad and Turney, 2010)). Since all the words in the Affect Intensity Lexicon also have entries in the NRC VAD Lexicon (Mohammad, 2018), we now examine the relationship between the valence, arousal, and dominance scores across different basic emotions.

Emotion	Avg. Valence	Avg. Arousal	Avg. Dominance
anger	0.26	0.66	0.46
fear	0.29	0.66	0.48
joy	0.77	0.52	0.60
sadness	0.24	0.58	0.38

Figure 2: Average valence, arousal, and dominance scores for each basic emotion. The cells are in shades of green with the darkness proportional to the score: lighter shades indicate low scores and darker shades indicate high scores.

4.1. Valence, Arousal, and Dominance of Words in the Affect Intensity Lexicon

The NRC VAD Lexicon (Mohammad, 2018) has valence, arousal, and dominance scores for over 20,000 commonly used English terms. It was created using best–worst scaling in a similar approach described earlier in this paper for obtaining emotion intensity scores. The three sets of scores range from 0 (lowest valence, arousal, and dominance) to 1 (highest valence, arousal, and dominance).

For each of the words in the Affect Intensity Lexicon, we looked up their entries in the NRC VAD Lexicon for scores of valence, arousal, and dominance. Figure 2 shows the average scores for each of the basic emotions. Figure 3 shows the scatter plot of the Affect Intensity Lexicon words across the orthogonal valence-arousal space.

Observe that as expected, joy words have much higher valence scores (are much more positive) on average than the anger, fear, and sadness words. Joy words also have

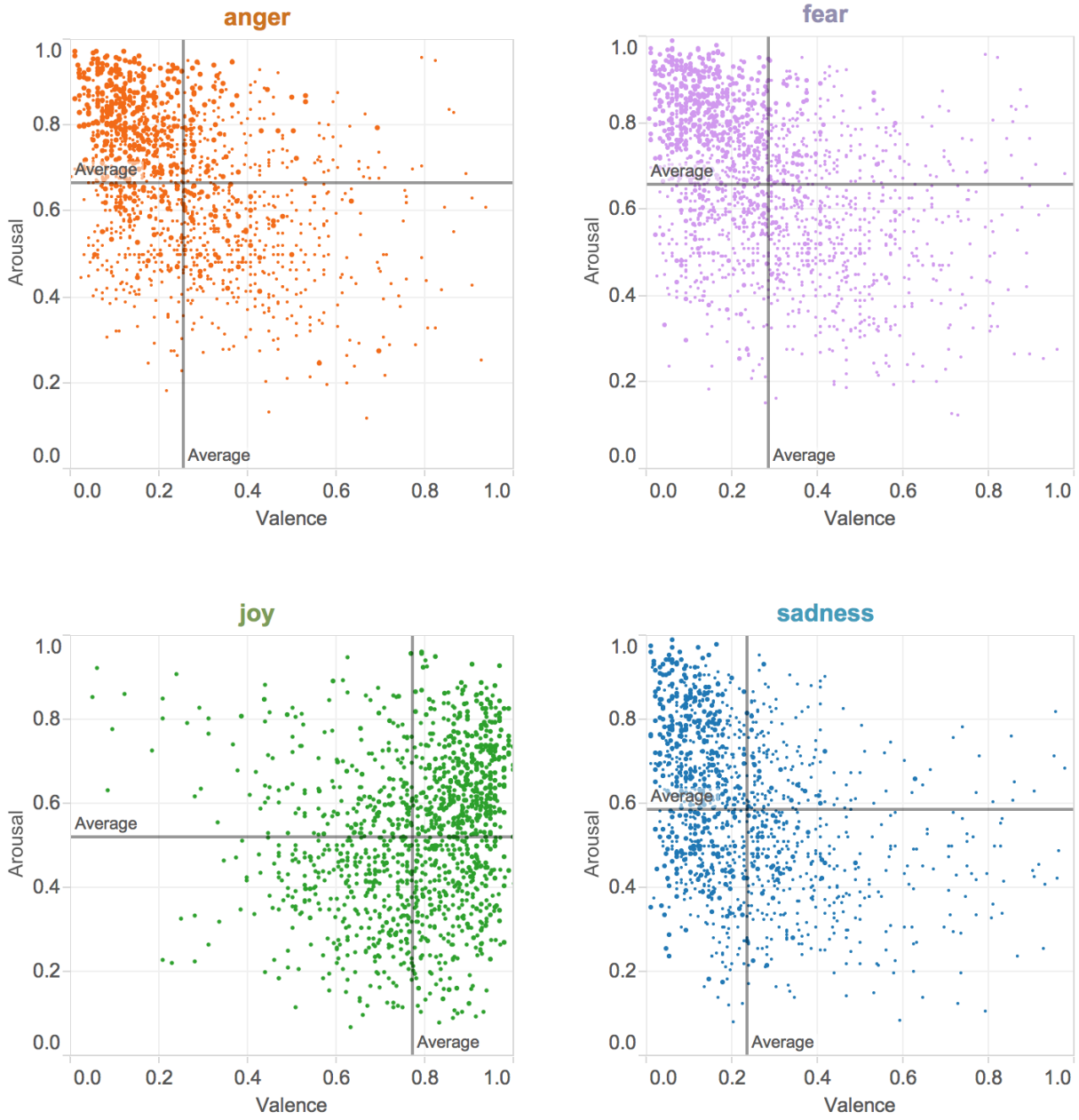


Figure 3: Valence–Arousal scatter plots for words associated with each of the four basic emotions. For example, the anger plot on the top left has points for every anger word in the Affect Intensity Lexicon. The position of the point indicates its valence and arousal scores (as obtained from the NRC VAD Lexicon). The size of the point is proportional to the intensity of anger (as obtained from AIL). The size of the point is proportional to the intensity of the corresponding emotion.

somewhat lower arousal scores (are more passive) on average than sadness words, which in turn have somewhat lower average arousal scores than anger and fear. Interestingly, anger and fear have a very similar profile of average VAD scores. Sadness words, on average, have the lowest average valence, followed by fear and anger.

To determine whether the dominance–arousal space separates the three negative emotion words from each other, we generated the corresponding scatter plots as well. See Figure 4. Observe that words conveying negative emotions can belong to a wide and overlapping range of arousal and dominance scores. The range of scores now overlaps markedly with the joy words as well. Figure 5 in the Appendix shows the scatter plots for the valence–dominance space.

Overall, we observe that the three negative emotions can be conveyed by words having a wide range of values for valence, arousal, and dominance.[11] Let the words with emotion intensity scores greater than 0.5 be called the *upper-half subset*. The upper half subset includes words expressing medium to high emotion intensity. Table 3 lists, for each emotion, the top four words that have highest and lowest valence, arousal, and dominance scores in the upper-half subset of the emotion. Note that for the negative emotions, the highest valence entries in the upper-half subset of anger are still expected to be somewhat negative.[12]

[11] The range is limited to the lower half of valence, but knowing valence is not sufficient to determine the precise basic emotion.

[12] For example, amongst the moderate-to-high anger terms, the highest valence term is still expected to be somewhat negative.

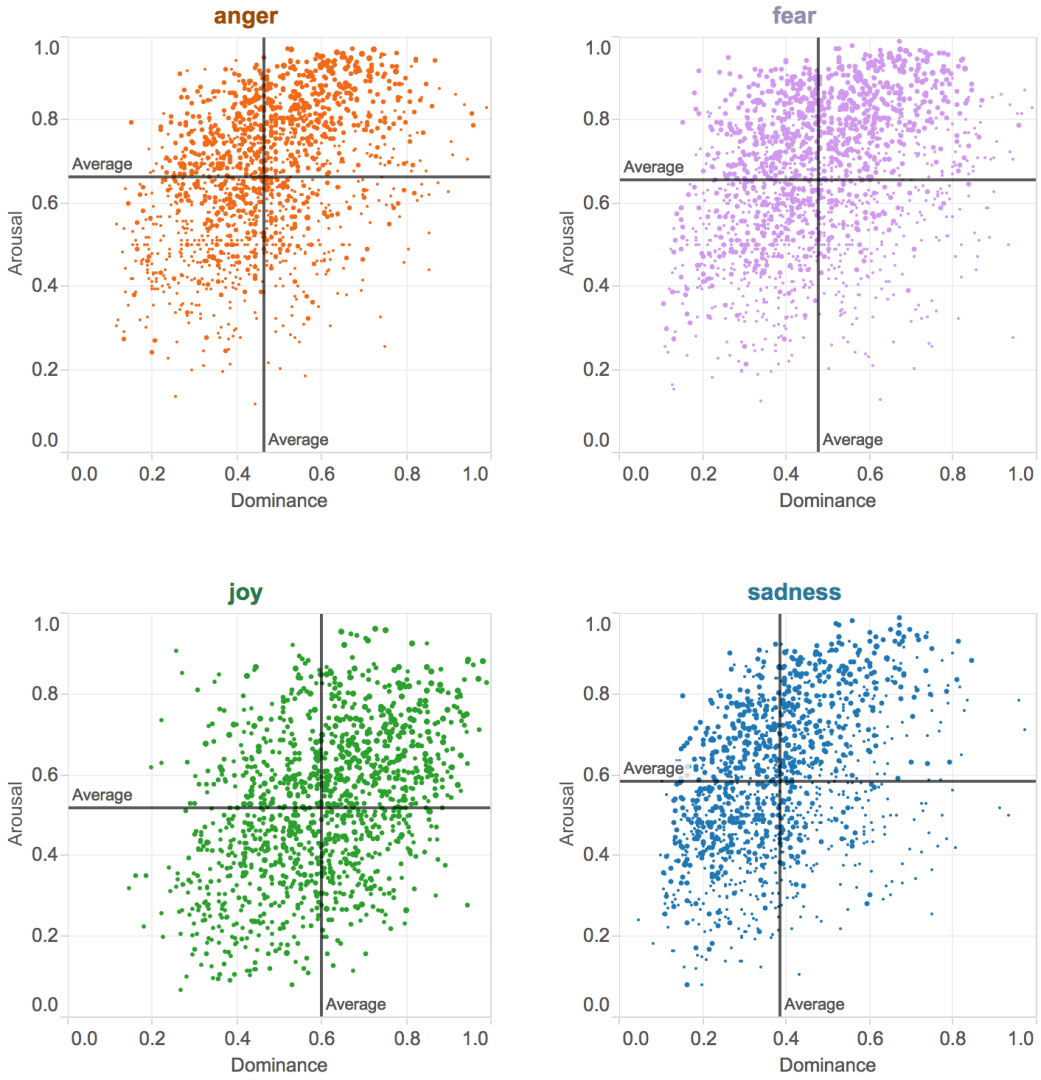


Figure 4: Dominance–Arousal scatter plots for words associated with each of the four basic emotions.

Emotion	V↑↑	V↓↓	A↑↑	A↓↓	D↑↑	D↓↓
anger > 0.5	*blaze*	*shit*	*homicide*	*batter*	*domination*	*casualty*
	glare	*homicide*	*terrorism*	*tiredofit*	*battle*	*idiots*
	incense	*murderous*	*violently*	*causality*	*overbearing*	*slave*
	temper	*terrorist*	*enraged*	*cross*	*dictatorial*	*dishonest*
fear > 0.5	*seize*	*nightmare*	*abduction*	*senile*	*domination*	*defenseless*
	meltdown	*afraid*	*exorcism*	*coma*	*projectiles*	*hopeless*
	retribution	*homicide*	*homicide*	*stalk*	*dictator*	*cowardly*
	enforce	*murderer*	*violently*	*hopeless*	*beastly*	*casualty*
joy > 0.5	*generous*	*raving*	*elated*	*stressfree*	*powerful*	*silly*
	happily	*zeal*	*excitation*	*peaceful*	*success*	*heheh*
	love	*silly*	*euphoria*	*serenity*	*triumphant*	*weeeee*
	magnificent	*boisterous*	*erotic*	*tranquility*	*winning*	*snuggles*
sadness > 0.5	*meltdown*	*bankruptcy*	*abduction*	*nothingness*	*warfare*	*defenseless*
	console	*disheartening*	*exorcism*	*alone*	*earthquake*	*weakly*
	insurmountable	*homicide*	*homicide*	*emptiness*	*bomber*	*hopeless*
	longing	*pain*	*terrorism*	*senile*	*unforgiving*	*pity*

Table 3: The top four words that have highest and lowest valence (V), arousal (A), and dominance (D) scores, while also having an emotion intensity score greater than 0.5 (in the upper-half subset). The emotion intensity scores are taken from the NRC Affect Intensity Lexicon and valence, arousal, and dominance scores are taken from the NRC VAD Lexicon. ↑↑ indicates the highest score entries. ↓↓ indicates the lowest score entries.

Emotion	Spearman	Pearson
anger	0.906	0.912
fear	0.910	0.912
joy	0.925	0.924
sadness	0.904	0.909

Table 4: Split-half reliabilities (as measured by Pearson correlation and Spearman rank correlation) for the anger, fear, joy, and sadness entries in the NRC Affect Intensity Lexicon.

5. Reliability of the Annotations

One cannot use standard inter-annotator agreement to determine quality of BWS annotations because the disagreement that arises when a tuple has two items that are close in emotion intensity is a useful signal for BWS. For a given 4-tuple, if respondents are not able to consistently identify the word that has highest (or lowest) emotion intensity, then the disagreement will lead to the two words obtaining scores that are close to each other, which is the desired outcome. Thus a different measure of quality of annotations must be utilized.

A useful measure of quality is reproducibility of the end result—if repeated independent manual annotations from multiple respondents result in similar intensity scores, then one can be confident that the scores capture the true emotion intensities. To assess this reproducibility, we calculate average *split-half reliability (SHR)* over 100 trials. SHR is a commonly used approach to determine consistency in psychological studies, that we employ as follows. All annotations for an item (in our case, tuples) are randomly split into two halves. Two sets of scores are produced independently from the two halves. Then the correlation between the two sets of scores is calculated. If the annotations are of good quality, then the correlation between the two halves will be high. Table 4 shows the split-half reliabilities for the anger, fear, joy, and sadness entries in the NRC Affect Intensity Lexicon. Observe that both the Pearson correlation and the Spearman rank correlations are above 0.9, indicating a high degree of reproducibility. Note that SHR indicates the quality of annotations obtained when using only half the number of annotations; the correlations obtained when repeating the experiment with four annotations for each 4-tuple is expected to be higher than 0.91. Thus 0.91 is a lower bound on the quality of annotations obtained with four annotations per 4-tuple.

6. Applications and Future Work

The NRC Affect Intensity Lexicon has many applications including automatic emotion analysis in a number of domains such as commerce, education, intelligence, and public health. The AIL was already used by several teams that participated in the WASSA-2017 shared task on Emotion Intensity in Tweet (Mohammad and Bravo-Marquez, 2017b) as well as the SemEval-2018 Task 1: Affect in Tweets (Mohammad et al., 2018) (including the teams that came first in both shared tasks). AIL is also useful in the building of natural language generation systems.

We are currently using the NRC Affect Intensity Lexicon along with tweets datasets that were annotated for emotion intensity (*Tweet Emotion Intensity Dataset* (Mohammad and Bravo-Marquez, 2017a)), to test the extent to which people convey strong emotions in tweets using high-intensity emotion words. We will also use the lexicon to identify syllables that consistently tend to occur in words with strong affect associations. This has implications in understanding how some syllables and sounds have a tendency to occur in words referring to semantically related concepts. Identifying emotions associated with a syllable is also useful in generating names for literary characters and commercial products.

The lexicon also has applications in the areas of digital humanities and literary analysis, where it can be used to identify high-intensity words. The NRC Affect Intensity Lexicon can also be used as a source of gold intensity scores to evaluate automatic methods of determining word affect intensity.

7. Conclusions

This paper describes how we created the *NRC Affect Intensity Lexicon*—a crowdsourced lexicon that captures word–affect intensities for four basic emotions: anger, fear, joy, and sadness. We used a technique called best–worst scaling (BWS) to obtain fine-grained scores (and word rankings). BWS addresses issues of annotation consistency that plague traditional rating scale methods of annotation. We show that repeat annotations of the terms in the Affect Intensity Lexicon with independent annotators lead to affect association scores that are close to the scores obtained originally (split-half reliability: $rho = 0.92$, $r = 0.91$). The fine-grained scores obtained with BWS and the high correlations on repeat annotations indicate that BWS is both markedly discriminative (helps identify small differences in affect intensity) and markedly reliable (provides stable outcomes).

The Affect Intensity Lexicon has applications in automatic emotion analysis as well as in understanding affect composition—how affect of a sentence is impacted by the affect of its constituent words. We will continue to add entries for other emotions such as disgust, trust, surprise, and anticipation. We will use the lexicon to study the role emotion words play in high emotion intensity tweets, using the Tweet Emotion Intensity Dataset that has intensity scores for whole tweets. We will also use the lexicon to determine syllables and phonetic sounds that are associated with particular affect categories, that is, syllables that tend to occur more often than average in affect-associated words. The lexicon is made freely available.

Acknowledgments

Many thanks to Svetlana Kiritchenko and Tara Small for helpful discussions.

8. Bibliographical References

Bradley, M. M. and Lang, P. J. (1999). Affective norms for English words (ANEW): Instruction manual and affective ratings. Technical report, The Center for Research in Psychophysiology, University of Florida.

Brants, T. and Franz, A. (2006). Web 1t 5-gram version 1. *Linguistic Data Consortium*.

Cohen, S. H. (2003). Maximum difference scaling: Improved measures of importance and preference for segmentation. Sawtooth Software, Inc.

David, H. A. (1963). *The method of paired comparisons*. Hafner Publishing Company, New York.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3):169–200.

Flynn, T. N. and Marley, A. A. J. (2014). Best-worst scaling: theory and methods. In Stephane Hess et al., editors, *Handbook of Choice Modelling*, pages 178–201. Edward Elgar Publishing.

Francisco, V. and Gervás, P. (2006). Automated mark up of affective information in English texts. In Petr Sojka, et al., editors, *Text, Speech and Dialogue*, volume 4188 of *Lecture Notes in CS*, pages 375–382. Springer.

Frijda, N. H. (1988). The laws of emotion. *American psychologist*, 43(5):349.

Jurgens, D., Mohammad, S. M., Turney, P., and Holyoak, K. (2012). Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluation*, pages 356–364, Montréal, Canada, 7-8 June.

Jurgens, D. (2013). Embracing ambiguity: A comparison of annotation methodologies for crowdsourcing word sense labels. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Atlanta, GA, USA.

Kiritchenko, S. and Mohammad, S. M. (2016). Capturing reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, San Diego, California.

Kiritchenko, S. and Mohammad, S. M. (2017). Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of The Annual Meeting of the Association for Computational Linguistics (ACL)*, Vancouver, Canada.

Louviere, J. J., Flynn, T. N., and Marley, A. A. J. (2015). *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge University Press.

Louviere, J. J. (1991). Best-worst scaling: A model for the largest difference judgments. Working Paper.

Mohammad, S. M. and Bravo-Marquez, F. (2017a). Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada.

Mohammad, S. M. and Bravo-Marquez, F. (2017b). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mohammad, S. M. and Kiritchenko, S. (2015). Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence*, 31(2):301–326.

Mohammad, S. M. and Kiritchenko, S. (2018). Understanding emotions: A dataset of tweets to study interactions between affect categories. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

Mohammad, S. M. and Turney, P. D. (2010). Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the NAACL-HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, LA, California.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Mohammad, S. M., Sobhani, P., and Kiritchenko, S. (2017). Stance and sentiment in tweets. *Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media*, 17(3).

Mohammad, S. M., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, USA.

Mohammad, S. (2012). #Emotional Tweets. In *Proceedings of the Joint Conference on Lexical and Computational Semantics*, pages 246–255, Montréal, Canada.

Mohammad, S. M. (2018). Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Submitted*.

Nakov, P., Rosenthal, S., Kiritchenko, S., Mohammad, S. M., Kozareva, Z., Ritter, A., Stoyanov, V., and Zhu, X. (2016). Developing a successful semeval task in sentiment analysis of twitter and other social media texts. *Language Resources and Evaluation*, 50(1):35–65.

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC Workshop on 'Making Sense of Microposts': Big things come in small packages*, pages 93–98, Heraklion, Crete.

Orme, B. (2009). Maxdiff analysis: Simple counting, individual-level logit, and HB. Sawtooth Software, Inc.

Parrot, W. (2001). *Emotions in Social Psychology*. Psychology Press.

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. *Emotion: Theory, research, and experience*, 1(3):3–33.

Russell, J. A. (2003). Core affect and the psychological construction of emotion. *Psychological review*, 110(1):145.

Stone, P., Dunphy, D. C., Smith, M. S., Ogilvie, D. M., and associates. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. The MIT Press.

Strapparava, C. and Mihalcea, R. (2007). Semeval-2007 task 14: Affective text. In *Proceedings of SemEval-2007*, pages 70–74, Prague, Czech Republic.

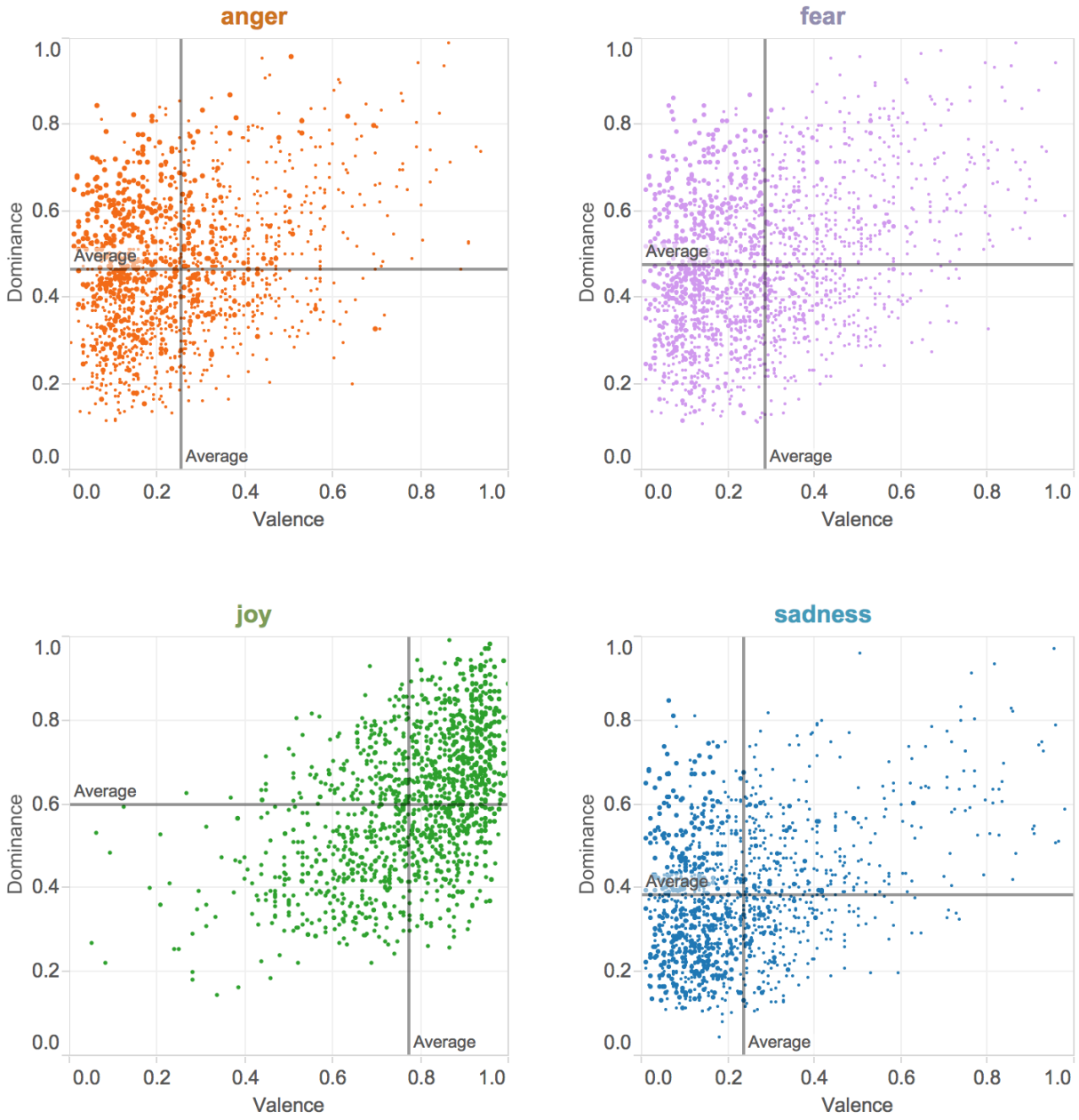


Figure 5: Valence–Dominance scatter plots for words associated with each of the four basic emotions. The size of the point is proportional to the intensity of the corresponding emotion. For example, the anger plot on the top left has points for every anger word in the Affect Intensity Lexicon. The position of the point indicates its valence and dominance scores (as obtained from the NRC VAD Lexicon). The size of the point is proportional to the intensity of anger (as obtained from the Affect Intensity lexicon).

Strapparava, C. and Valitutti, A. (2004). Wordnet-Affect: An affective extension of WordNet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 1083–1086, Lisbon, Portugal.

Thurstone, L. L. (1927). A law of comparative judgment. *Psychological review*, 34(4):273.

Warriner, A. B., Kuperman, V., and Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, 45(4):1191–1207.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2-3):165–210.

Yu, L.-C., Lee, L.-H., Hao, S., Wang, J., He, Y., Hu, J., Lai, K. R., and Zhang, X.-j.). Building Chinese affective resources in valence-arousal dimensions. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, San Diego, California.

9. Appendix

Figure 5 show scatter plots for the four basic emotions in the valence-dominance space.

Representation Mapping: A Novel Approach to Generate High-Quality Multi-Lingual Emotion Lexicons

Sven Buechel & Udo Hahn
{sven.buechel|udo.hahn}@uni-jena.de
Jena University Language and Information Engineering (JULIE) Lab
Fürstengraben 27, 07743 Jena, Germany
http://www.julielab.de

Abstract

In the past years, sentiment analysis has increasingly shifted attention to representational frameworks more expressive than semantic polarity (being positive, negative or neutral). However, these richer formats (like Basic Emotions or Valence-Arousal-Dominance, and variants therefrom), rooted in psychological research, tend to proliferate the number of representation schemes for emotion encoding. Thus, a large amount of representationally incompatible emotion lexicons has been developed by various research groups adopting one or the other emotion representation format. As a consequence, the reusability of these resources decreases as does the comparability of systems using them. In this paper, we propose to solve this dilemma by methods and tools which map different representation formats onto each other for the sake of mutual compatibility and interoperability of language resources. We present the first large-scale investigation of such representation mappings for four typologically diverse languages and find evidence that our approach produces (near-)gold quality emotion lexicons, even in crosslingual settings. Finally, we use our models to create new lexicons for eight typologically diverse languages.

Keywords: Automatic Construction of Emotion Lexicons, Representation Mapping, Models of Emotion

1. Introduction

In the past two decades, the NLP-based analysis and prediction of affective states, as performed by sentiment analysis systems, has received enormous interest (Liu, 2015). Starting with simple positive-negative polarity distinctions on the word or text level (Hatzivassiloglou and McKeown, 1997; Pang et al., 2002), research in sentiment analysis has since then shifted towards more nuanced and challenging tasks, e.g., sentiment compositionality (Socher et al., 2013), aspect-level assessments (Schouten and Frasincar, 2016) or stance detection (Sobhani et al., 2016). In parallel, psychologically more advanced and more expressive *representation formats* for affective states have been proposed, like Basic Emotions (Ekman, 1992) or Valence-Arousal-Dominance (Bradley and Lang, 1994). However, there is currently no consensus in the literature what scheme should be used as a common ground. Rather, there are a multitude of competing formats often motivated by the needs of concrete applications or the availability of user-labeled social media data (Desmet and Hoste, 2013; Li et al., 2016).

While such decisions for a specific format may be perfectly reasonable in a specific research setting, on the flip-side, this proliferation of competing formats may seriously hamper progress in sentiment analysis for two reasons, at least. First, language resources are less reusable (if at all) as gold standards and, second, with the growing number of representation formats meaningful comparisons between predictive systems become harder (if not impossible).

One way to resolve this dilemma is to develop techniques to automatically translate between such formats. This task of *emotion representation mapping* (EMOMAP) was introduced only very recently to NLP by Buechel and Hahn (2017b). Their work came up with an emotion-labeled corpus which, in part, is annotated with two different emotion formats both being highly predictive for each other. In a follow-up study, Buechel and Hahn (2017a) examined the potential of EMOMAP as a substitute for manual annotation, yet their comparison was restricted to only two emotion lexicons. Comparable work has (to the best of our knowledge) only been done in psychology. However, this stream of work does not target the goal of predictive modeling (Stevenson et al., 2007; Pinheiro et al., 2017). In NLP, a task related to EMOMAP is *emotion prediction* on the level of words, sentences, or texts (Wang et al., 2016; Sedoc et al., 2017) where, in contrast to EMOMAP, the target unit does not already need to bear annotations from another format. Thus, emotion prediction algorithms constitute a reasonable baseline for EMOMAP (see Section 4.).

This contribution puts emphasis on emotion lexicons developed in psychology. Although highly relevant for sentiment analysis, those resources have mostly been neglected by NLP researchers as the discussion of related work in Section 2. reveals. Making use of this valuable work, we here conduct the first thorough evaluation of EMOMAP for emotion lexicon construction on four typologically diverse languages and find strong evidence that the quality of the output we generate is on a par with a gold standard when compared to human performance (see Section 4.). Finally, we exploit our models to create novel emotion lexicons for eight different languages (including low-resource ones; Section 5.). The lexicons as well as the source code for building them are publicly available (see Section 6.).

2. Data

Models of emotion are typically subdivided into *discrete* (or *categorical*) and *dimensional* ones (Stevenson et al., 2007; Calvo and Mac Kim, 2013). Discrete models are centered around particular sets of emotional categories deemed fundamental. Ekman (1992), for instance, identifies six *Basic Emotions* (Joy, Anger, Sadness, Fear, Disgust and Surprise). In contrast, dimensional models consider emotions to be composed out of several influencing factors (mainly

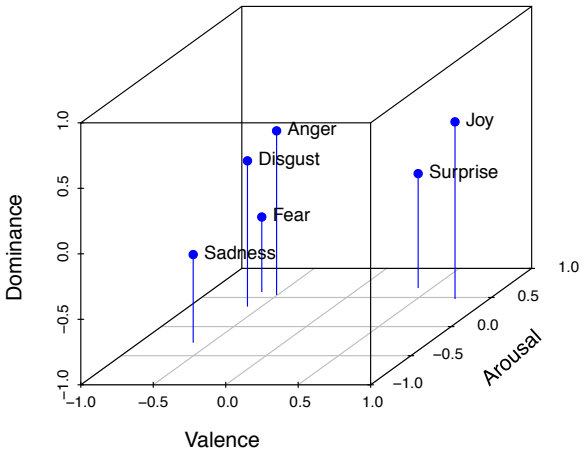


Figure 1: Affective space spanned by the Valence-Arousal-Dominance model, together with the position of six Basic Emotions; positions determined by Russell and Mehrabian (1977); figure adapted from Buechel and Hahn (2016).

two or three). These are often referred to as *Valence* (corresponding to the concept of polarity), *Arousal* (a calm–excited scale), and *Dominance* (perceived degree of control over a (social) situation)—the VAD model (see Figure 1 for an illustration of the relationship between VAD dimensions and Basic Emotion categories). The last dimension, Dominance, is sometimes omitted, leading to the VA model.

In contrast to NLP where many different formats are being used lexical resources in psychology almost exclusively subscribe to VA(D) or Basic Emotions (typically omitting Surprise; the BE5 format). Over the years, a considerable number of resources built on these premises have emerged from psychological research labs for various languages. Table 1 enumerates published resources based on these two approaches (27 in total covering 13 languages, including low-resource ones such as Finnish and Indonesian). To the best of our knowledge, the vast majority of them has neither been used nor referenced in NLP research.

In this paper, we restrict ourselves to the VAD and BE5 format. In more detail (following the conventions of our emotion lexicons), each VAD dimension receives a value from the interval $[1, 9]$ where '1' means "most negative/calm/submissive", '9' means "most positive/excited/dominant" and '5' means "neutral". Conversely, values for BE5 categories range in the interval $[1, 5]$ where '1' means "absence" and '5' means "most extreme" expression of the respective emotion.[1] Consequently, the VAD and BE5 formats are conceptually different from one another insofar as VAD dimensions are bi-polar, whereas BE5 categories are uni-polar.

Our work is based on the condition that some pairs of data sets in Table 1 are complementary in the sense that, when combining these lexicons, a subset of the entries they contain are then described according to *both* emotion formats, VAD *and* BE5. This condition is illustrated for three lexical items in Table 2.

[1] Although these intervals are fairly well established conventions, in some data sets different rating scales are used, nevertheless. In these cases, we linearly transformed the ratings so that they match the defined intervals.

Reference	Lang.	Format	# Entries
Warriner et al. (2013)	en	VAD	13,915
Stevenson et al. (2007)	en	BE5	1,034
Bradley and Lang (1999)	en	VAD	1,034
Stadthagen-Gonzalez et al. (2017)	es	VA	14,031
Ferré et al. (2017)	es	BE5	2,266
Guasch et al. (2015)	es	VA	1,400
Redondo et al. (2007)	es	VAD	1,034
Hinojosa et al. (2016a)	es	VA+BE5	875
Hinojosa et al. (2016b)	es	+D	875
Võ et al. (2009)	de	VA	2,902
Briesemeister et al. (2011)	de	BE5	1,958
Schmidtke et al. (2014)	de	VAD	1,003
Kanske and Kotz (2010)	de	VA	1,000
Imbir (2016)	pl	VAD	4,905
Riegel et al. (2015)	pl	VA	2,902
Wierzba et al. (2015)	pl	BE5	2,902
Yu et al. (2016)	zh	VA	2,802
Yao et al. (2017)	zh	VA	1,100
Monnier and Syssau (2014)	fr	VA	1,031
Ric et al. (2013)	fr	V+BE5	524
Moors et al. (2013)	nl	VAD	4,299
Sianipar et al. (2016)	id	VAD	1,490
Palogiannidi et al. (2016)	gr	VAD	1,034
Montefinese et al. (2014)	it	VAD	1,121
Soares et al. (2012)	pt	VAD	1,034
Eilola and Havelka (2010)	fi	VA	210
Davidson and Innes-Ker (2014)	sv	VA	100

Table 1: List of VA(D) and BE5 lexicons with empirically gathered ratings from human subjects; including reference, language code (according to ISO 639-1), emotion representation format, and number of lexical entries.

Word	V	A	D	J	A	S	F	D
sunshine	8.1	5.3	5.4	4.2	1.2	1.3	1.3	1.2
terrorism	1.6	7.4	2.7	1.1	3.1	3.4	3.7	2.7
orgasm	8.0	7.2	5.8	4.2	1.3	1.3	1.5	1.2

Table 2: Three lexical items and their emotion values in VAD (second column group) and BE5 (third column group) format. VAD scores are taken from Warriner et al. (2013), BE5 scores were automatically derived (see Section 5.).

From the resources listed in Table 1, we identified such complementary pairs and merged them into four lexicons which serve as gold data for the subsequent experiments:

- **English**: Bradley and Lang (1999) intersected with Stevenson et al. (2007) yielded 1,034 overlapping entries.
- **Spanish**: Redondo et al. (2007) intersected with Ferré et al. (2017) yielded 1,012 overlapping entries.
- **Polish**: Imbir (2016) intersected with Wierzba et al. (2015) yielded 1,272 overlapping entries.
- **German**: Schmidtke et al. (2014) intersected with Briesemeister et al. (2011) yielded 318 overlapping entries.

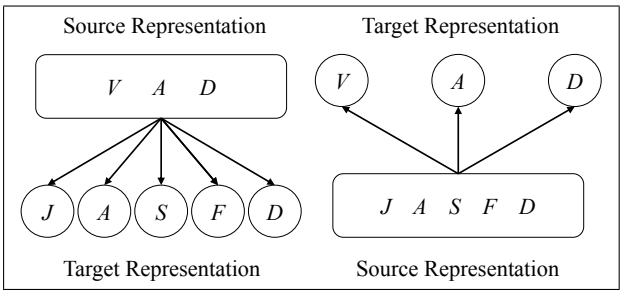


Figure 2: Illustration of the two representation mapping procedures: VAD2BE5 (left) vs. BE52VAD (right). Each arrow represents an individual kNN model.

3. Method

Given an emotion lexicon in VAD format, our goal is to map its ratings onto the BE5 format and vice versa. We employ a simple, yet surprisingly efficient, method proposed by Buechel and Hahn (2017b): For each of the dimensions or categories of the target representation (VAD or BE5, respectively), we train a single supervised model which employs each of the dimensions/categories of the source representation as features (e.g., *one* model to predict Joy, given Valence, Arousal, and Dominance scores as input; see Figure 2 for a graphical illustration of the general scheme).

In a pilot study, we compared different learning algorithms including linear regression, *k* nearest neighbor regression (kNN), support vector regression (using different kernels), random forests, as well as feed-forward neural networks. To our surprise, all of them performed equally well (with only negligible differences). Thus, kNN was selected due to its simplicity.[2] Note that the feature set is extremely small (either three or five variables for mapping *onto* BE5 or VAD, respectively) so that using more complex methods (e.g., more sophisticated neural architectures) seems a waste of efforts.

4. Experiments

We here present the first large-scale evaluation of emotion representation mapping (EMOMAP). Our methodology, at the same time, leads to the automatic construction of emotion lexicons for four typologically diverse languages. We consider one monolingual and two crosslingual set-ups, i.e., training and testing data from the same or different language(s), respectively. Those three different mapping strategies are illustrated in Figure 3.

The performance of the EMOMAP approach will be measured as Pearson correlation (r) between our automatically predicted values and human gold ratings. In general, the Pearson correlation between two data series $X = x_1, x_2, ..., x_n$ and $Y = y_1, y_2, ..., y_n$ takes values between $+1$ (perfect positive correlation) and -1 (perfect negative correlation). It is computed as

$$r_{xy} := \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (1)$$

where $\bar{x}$ and $\bar{y}$ denote the mean values for X and Y, respectively.

[2] We use the `scikit-learn.org` implementation.

These measurements will then be compared, first, with the current state-of-the-art in word-level emotion prediction (as baseline), and, second, with human inter-study reliability (as ceiling). Both comparisons will, for different reasons, be limited to the VAD model.

4.1. Baseline and Ceiling

Word-level emotion prediction (automatically deriving the emotion of a word from scratch; see Section 1.) serves as a reasonable baseline since it produces the same output as EMOMAP, yet does not require the target words to have already been annotated in a different emotion format (other than the output representation).

Sedoc et al. (2017) evaluated their approach to word-level emotion prediction on the data set compiled by Bradley and Lang (1999) using 10-fold cross-validation. They report measurements of $r = 0.806$ for Valence and $r = 0.615$ for Arousal. Concerning the other affective dimensions and categories, we are not aware of any other system predicting numerical scores for them. Thus, we will restrict our comparison to Valence and Arousal.

For comparison against the human ceiling, we found eight pairs of emotion lexicons with partially overlapping entries distributed over four languages. For each of these pairs, we computed their inter-study reliability (ISR), i.e., the Pearson correlation between the ratings from the two respective studies for each affective dimension (see Table 3). Again, because we only found lexicons with overlapping VAD (not BE5) entries, we restrict this comparison to VAD representations.

We stipulate that for all ISR values from Table 3, the *minimum* for each affective dimension constitutes the most relevant score of comparison. The rationale for this assumption is as follows: If our approach happens to outperform this minimal value, one cannot be certain that manual annotation leads to better results than using our automatic procedure. In this situation, we assume that the computational approach would almost always be preferred over manual annotation efforts. Accordingly, the following correlation values were identified as minimum inter-study reliabilities: $r = .948$ for Valence, $r = .709$ for Arousal, and $r = .794$ for Dominance (henceforth, jointly referred to as ISR_{min}).

Note that comparing against the ISR is a much harder test than comparing against inter-annotator agreement (IAA): Since the former is based on the mean rating of many raters, this aggregated judgment is more stable than individual ratings, thus resulting in higher correlation values compared to its IAA counterpart.[3]

4.2. Monolingual Evaluation

The first part of our analysis concerns train and test data originating from the same language (respectively data set); see left part of Figure 3. For each of our (language-wise) four gold lexicons (cf. the final paragraph of Section 2.), we train kNN models to map between VAD and BE5 representations back and forth according to the scheme from Fig-

[3] For numerical emotion ratings, IAA is typically computed in a leave-one-out fashion and can thus be interpreted as how well a single human annotator predicts the gold value (Strapparava and Mihalcea, 2007; Buechel and Hahn, 2017b).

	Val	Aro	Dom	#Overlap
Imbir (2016) vs. Riegel et al. (2015)	**.948**	.733	—	1,272
Guasch et al. (2015) vs. Stadthagen-Gonzalez et al. (2017)	.949	**.875**	—	1,298
Bradley and Lang (1999) vs. Warriner et al. (2013)	.952	.760	**.794**	1,027
Guasch et al. (2015) vs. Hinojosa et al. (2016a)	.968	.777	—	134
Guasch et al. (2015) vs. Redondo et al. (2007)	.969	.844	—	316
Hinojosa et al. (2016a) vs. Stadthagen-Gonzalez et al. (2017)	.970	**.709**	—	636
Schmidtke et al. (2014) vs. Kanske and Kotz (2010)	.971	.788	—	169
Redondo et al. (2007) vs. Stadthagen-Gonzalez et al. (2017)	**.976**	.755	—	1,010

Table 3: Inter-study reliabilities between different data sets (measured in r). Minimum and maximum values for each VAD dimension (respectively) in bold.

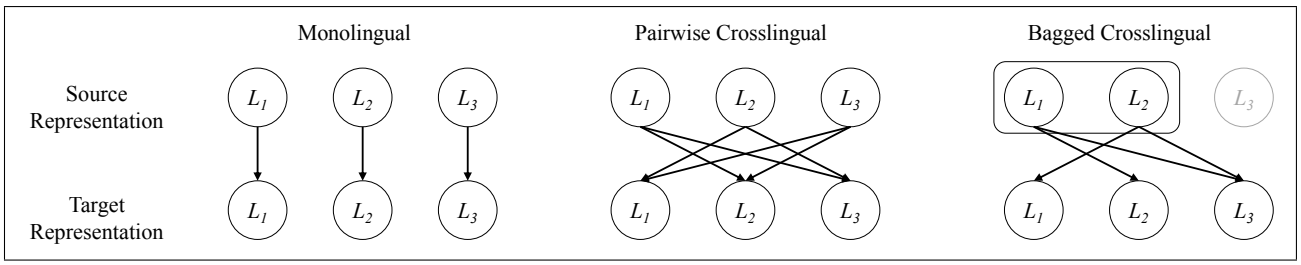


Figure 3: Illustration of the three mapping strategies applied in Section 4. exemplified for the languages L_1, L_2 and L_3.

ure 2. Training and testing was done using 10-fold cross-validation (9:1 train/test split). The k-parameter was fixed to 20, based on a pilot study (eliminating the need for a dev set). The results are presented in Table 4, upper section.

As can be seen, the outcome is overall favorable for our approach. In general, it works about equally well in both mapping directions (VAD2BE5 and BE52VAD) with average values (over VAD dimensions and BE5 categories, respectively) of $r \geq 77\%$. The results on the English and Spanish gold lexicons are better than for the Polish, yet worst for the German one (which is also the smallest). In comparison with the baseline (see above; English data set only), our EMOMAP approach performs more than 15 percentage points better for Valence and more than 10 percentage points better for Arousal. Even more surprisingly, compared to the human ceiling, we find that our approach outperforms the ISR_{min} in 9 out of 12 cases (again, only failing to do so on the Polish and German data set). In those 9 cases where we outperformed the ISR_{min}, we conducted a one-tailed one sample t-test based on the 10 individual cross-validation results (Dietterich, 1998) finding significant differences in 6 of these cases ($p < .05$; marked with asterisk in Table 4).

We conclude that, in the monolingual set-up, EMOMAP performs on a par with (if not superior to) manual annotation for mapping *onto* VAD. Thus, its results can be considered as true gold data. For BE5, we cannot draw the same conclusion due to a lack of data on inter-study reliability. However, since the performance figures for the VAD2BE5 mapping are equally high, we may quite safely assume that the Basic Emotion ratings can be attributed high quality as well.

4.3. Pairwise Crosslingual Evaluation

In the two crosslingual set-ups (training and test data drawn from different languages, respectively data sets), we make use of the fact that our models do not rely on any language-specific information since the categories/dimensions describe (supposedly universal) affective states rather than linguistic entities. Thus, models trained on one language could, in theory, be applied to another without any adaptation.

Let us, first, address *pairwise* comparisons. That is, for each language, we train our kNN models on the entirety of the respective data set and then test on all the remaining languages individually (illustrated in Figure 3; resulting in a total of 12 language pairs). Since, this set-up uses fixed training and test sets, there is no need for cross-validation. The results are given in Table 4 (middle section).

Overall, the values remain astonishingly high. As can be seen, for mapping BE52VAD, the results are quite favorable for Valence with correlation values ranging well above 90% of correlation. On this dimension, our approach still outperforms the baseline by over a 15%-points margin and even surpasses the human ceiling in more than half of the cases, five of them being statistically significant. Since different from Section 4.2., we now have a fixed test set, we use a one-tailed z-test ($p < .05$) based on z-transformed correlation values (Cohen, 1995).

In contrast to Valence, the performance for Arousal and Dominance may suffer quite substantially in the crosslingual approach, depending on the combination of training and testing languages. While there is almost no performance loss for combinations of English and Spanish, the correlation decreases the most for combinations of Polish and German (especially for predicting Dominance), possibly due to data sparsity of the lexicons involved.

This outcome led us to conclude that the relationship between VAD and BE5 ratings is not fully constant across different languages (respectively data sets). Rather it seems to depend on subtle semantic differences between the translational equivalents of the affective dimensions/categories,

Experiment	Language	Val	Aro	Dom	**Av$_{VAD}$**	Joy	Anger	Sadn	Fear	Disg	**Av$_{BE5}$**
monolingual	English	.966*	.723	.833*	**.841**	.958	.870	.864	.864	.790	**.869**
	Spanish	.970*	.736	.855*	**.854**	.957	.847	.828	.870	.744	**.849**
	Polish	.944	.761*	.740	**.815**	.932	.845	.803	.784	.814	**.836**
	German	.950	.762*	.637	**.783**	.923	.793	.680	.851	.602	**.770**
crosslingual (pairwise)	es2en	.963*	.714	.794	**.824**	.948	.830	.853	.835	.780	**.849**
	pl2en	.962*	.598	.776	**.778**	.955	.845	.836	.832	.765	**.847**
	de2en	.952	.445	.762	**.720**	.952	.861	.836	.855	.746	**.850**
	en2es	.966*	.737*	.811	**.838**	.948	.791	.806	.826	.694	**.813**
	pl2es	.961*	.634	.701	**.765**	.941	.744	.763	.766	.665	**.776**
	de2es	.959*	.498	.842*	**.767**	.942	.794	.785	.839	.640	**.800**
	en2pl	.938	.655	.653	**.749**	.924	.816	.800	.751	.795	**.817**
	es2pl	.934	.663	.552	**.717**	.918	.755	.762	.653	.768	**.771**
	de2pl	.920	.674	.497	**.697**	.914	.815	.759	.700	.739	**.785**
	en2de	.940	.615	.583	**.713**	.915	.789	.678	.849	.584	**.763**
	es2de	.953	.618	.645	**.739**	.904	.789	.692	.840	.579	**.761**
	pl2de	.934	.691	.358	**.661**	.907	.768	.655	.788	.529	**.730**
crosslingual (bagged)	English	.963*	.714	.794	**.824**	.948	.830	.853	.835	.780	**.849**
	Spanish	.966*	.737*	.811	**.838**	.948	.791	.806	.826	.694	**.813**
	Polish	.939	.645	.629	**.738**	.926	.781	.780	.700	.769	**.791**
	German	.949	.635	.632	**.739**	.917	.799	.692	.844	.551	**.761**

Table 4: Results of the monolingual (Section 4.2.) and crosslingual (Sections 4.3. and 4.4.) evaluation in Pearson's r. Language 'a2b' denotes mapping from language *a* (source) to language *b* (target). Significant values are marked with '*' (compared to ISR$_{min}$; $p < .05$; VAD only), averages over VAD and BE5 (respectively) in bold.

cultural differences, or variations in the annotation guidelines, suggesting that the above assumption of language independence (not so surprisingly) may not fully hold.

In contrast to these partly inconclusive results, the outcome for mapping VAD2BE5 is much more favorable for EMOMAP and easy to describe. Compared to the monolingual set-up (relative to the target language), the drop of the average performance amounts to only a few percentage points (< 5 in most cases). Thus, the predictions for BE5 are much more robust compared to the VAD predictions which might be an effect of the respective source representation.

We conclude that in the pairwise crosslingual set-up, EMOMAP still performs really well in many cases. Yet depending on the language pair, the performance may degrade (much more severely so for mapping BE52VAD).

4.4. Bagged Crosslingual Evaluation

As evident from the last section, the performance of our mapping approach may vary depending on source and target language. However, different from the last experiment, when constructing new emotion lexicons in a crosslingual fashion, there is no need to restrict the training set to only one language. Instead, because no language-specific features are used, we may merge training data from multiple languages if this leads to a more robust predictive model. However, since not all languages (respectively data sets) seem to match well, there is no guarantee that more data sets always help boosting performance. In line with these considerations, the goal of the last experiment is to identify the best group of training data sets for automatically creating novel lexicons and to estimate their quality.

For each combination of gold lexicons of bag sizes two[4] to four and each target language, we train our models on the entirety of the bag of training data (except the one designated for testing, should it also belong to the training data) and then test on the target language; see the third data scenario in the right part of Figure 3.

In line with Section 4.3., we found that the average performance over VAD behaved less robust across different combinations of training data (ranging between $r = .730$ to .786) compared to BE5 ($r = .771$ to .806). Concerning the average over *all* emotions, the combination of Polish and German, once again, performed worst ($r = .755$), whereas the combination of English and Spanish worked best ($r = .794$; i.e., for testing on English, Spanish was used for training and vice versa, while for testing on the remaining languages, training was done on English *and* Spanish). Consequently, this combination of gold data was used for creating novel lexicons in the crosslingual set-up (see Section 5.).

Table 4 (bottom section) displays the results of this crosslingual experiment for the best performing bag of training data (comprising the English and Spanish gold lexicons, only). Thus, these performance data serve as an estimate of the quality of the novel emotion lexicons presented in Section 5. Overall, we find that the results are again favorable for our approach. The correlation with the English and Spanish data set, not surprisingly, is stronger than with the Polish and German one, confirming that these data sets form a better basis for generalization (the effect being more obvious for VAD than for BE5). In general, Valence, Joy, and

[4] If only one lexicon would be used for training, this one could not also be used for testing, thus making the different combinations incomparable.

	Language	#VAD	#BE5
monolingual	English		12,888
	Spanish	1,254	
	German	1,641	683
	Polish		3,633
crosslingual	Italian		1,121
	Portuguese		1,034
	Dutch		4,299
	Indonesian		1,490

Table 5: Automatically constructed gold quality lexicon resources, '#' indicates the number of previously unrated lexical units for a specific representation format.

	Joy	Anger	Sadness	Fear	Disgust
Mean	2.11	1.62	1.61	1.66	1.59
Median	1.86	1.38	1.38	1.42	1.37
Min	1.07	1.14	1.21	1.17	1.11
Max	4.40	3.38	3.81	3.74	3.26
StDev	0.79	0.47	0.47	0.49	0.47

Table 6: Descriptive statistics for the automatically constructed English BE5 lexicon.

Anger can be predicted with consistently high correlation, whereas for the other dimensions/categories we find occasional negative outliers.

Comparing our results to human reliability (in VAD only), we find that our models are superior to human ISR_{min} in 7 from 12 cases (including all cases on the English and Spanish data set). In 3 of these cases, the difference is statistically significant ($p < .05$). In comparison to the baseline (on English, VA only), our approach still clearly outperforms state-of-the-art word-level emotion prediction by a 15 and 10 percentage point margin for Valence and Arousal, respectively.

We conclude that even if no gold data for a given language are available, EMOMAP still performs comparably to human reliability when utilizing appropriate sets of training data. Some dimensions and categories seem to be reliable across data sets, whereas for others the performance may degrade, depending on the target data set. Yet, the lexicons derived in this set-up can still be attested *near-gold* quality.

Joy	Anger	Sadness	Fear	Disgust
christmas	*killer*	*chemo*	*insanity*	*felony*
happiness	*gang*	*worthless*	*motherfucker*	*enraged*
magical	*revenge*	*gonorrhea*	*terrorism*	*traitorous*
fun	*die*	*nausea*	*attacker*	*dishonesty*
enjoyment	*massacre*	*virus*	*bullshit*	*chauvinist*
bonus	*attacker*	*amputation*	*murderous*	*mistrust*
oasis	*sue*	*unhappiness*	*dangerous*	*gory*
fantastic	*hijacker*	*unsanitary*	*tragedy*	*hostile*
happy	*nigger*	*molester*	*arrest*	*racist*
sunshine	*penniless*	*lynching*	*rape*	*cellulite*

Table 7: Top 10 entries per Basic Emotion in automatically constructed English BE5 lexicon.

	V	A	D	J	A	S	F	D
V	-	–.18	+.72	+.92	–.83	–.82	–.75	–.87
A	-	-	–.18	–.03	+.58	+.46	+.67	+.41
D	-	-	-	+.68	–.66	–.76	–.68	–.61
J	-	-	-	-	–.66	–.61	–.59	–.69
A	-	-	-	-	-	+.92	+.95	+.91
S	-	-	-	-	-	-	+.91	+.85
F	-	-	-	-	-	-	-	+.82
D	-	-	-	-	-	-	-	-

Table 8: Correlation matrix (in r) for automatically constructed English BE5 (JASFD) lexicon combined with the data by Warriner et al. (2013) (VAD).

5. Construction of New Emotion Lexicons

After the positive evaluation of EMOMAP for four typologically diverse languages, our main contribution is to apply the created models to a wide variety of data sets which so far bear emotion ratings for *one* format only (either VAD or BE5). Based on our preceding experiments, we claim that these have gold quality (using the monolingual approach, Section 4.2.) or near-gold quality (using the crosslingual approach, Section 4.4.). We constructed a total of nine emotion lexicons covering eight languages (including low-resource ones, such as Dutch and Indonesian). Table 5 depicts the number of lexical items for which we have generated previously unknown VAD or BE5 ratings per language. For illustration, we provide an analysis of the English BE5 lexicon (by far the largest resource constructed in this manner) in the remainder of this section.

Table 6 provides fundamental statistical characteristics of this newly developed data set. As can be seen, Joy ratings have higher mean, standard deviation and range than all the other categories. This suggests that a larger portion of lexical items expresses at least a moderate degree of Joy, whereas the other Basic Emotions are expressed less often and to a smaller extent. Table 7 lists the ten entries with the highest values for each Basic Emotion category. Obviously, the automatically derived ratings align well with our intuition, thus granting face validity to our approach.

Finally, Table 8 provides correlation values between the BE5 categories and VAD dimensions (ratings for the latter were taken from Warriner et al. (2013)). Joy displays a moderate negative correlation with the other Basic Emotions while these in turn have strong positive correlation among each other. Unsurprisingly, Valence displays a strong positive correlation with Joy and strong negative correlations with the remaining BE5 categories. Lastly, Arousal is uncorrelated with Joy but displays moderate positive correlation with Anger, Sadness, Fear and Disgust. These findings are consistent with empirically determined emotion values, thus validating our claims concerning the good quality of the constructed resources (Wierzba et al., 2015; Hinojosa et al., 2016a).

6. Conclusion

Progress in emotion analysis is hampered by a multitude of heterogeneous and, in the end, mutually incompatible emotion representation formats. In this paper, we performed the first large-scale analysis of *representation map-*

ping as a means to mediate between these heterogeneous formats. Our simple, yet highly effective, supervised approach makes use of the wide range of emotion lexicons already developed in various psychology labs.

We could show that, in the monolingual setup, automatic representation mapping outperforms human inter-study reliability and, therefore, produces *gold quality* data. In the crosslingual set-up, our approach still performs comparable to manual annotation though less robust than in the first set-up for some affective dimensions or categories, thus rendering *near-gold quality* entries. In both set-ups, mapping existing ratings to another format performs way better than the state-of-the-art in emotion prediction. Hence, we conjecture that our approach paves the way to greatly improve interoperability and re-use of lexical resources in this field.

Lastly, we applied our technique to produce (near) gold quality emotion lexicons for eight typologically diverse languages, including low-resourced ones. These resources (together with our code) are available via `github.com/JULIELab/EmoMap`.

7. Bibliographical References

Bradley, M. M. and Lang, P. J. (1994). Measuring emotion: The Self-Assessment Manikin and the semantic differential. *Journal of Behavior Therapy and Experimental Psychiatry*, 25(1):49–59.

Bradley, M. M. and Lang, P. J. (1999). Affective Norms for English Words (ANEW): Stimuli, instruction manual and affective ratings. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, Gainesville, FL.

Briesemeister, B. B., Kuchinke, L., and Jacobs, A. M. (2011). Discrete Emotion Norms for Nouns: Berlin Affective Word List (DENN–BAWL). *Behavior Research Methods*, 43(2):441.

Buechel, S. and Hahn, U. (2016). Emotion analysis as a regression problem: Dimensional models and their implications on emotion representation and metrical evaluation. In *ECAI 2016 — Proceedings of the 22nd European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS 2016)*, volume 285 of *Frontiers in Artificial Intelligence and Applications*, pages 1114–1122.

Buechel, S. and Hahn, U. (2017a). A flexible mapping scheme for discrete and dimensional emotion representations: Evidence from textual stimuli. In *CogSci 2017 — Proceedings of the 39th Annual Meeting of the Cognitive Science Society*, pages 180–185.

Buechel, S. and Hahn, U. (2017b). EMOBANK: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *EACL 2017 — Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: Short Papers, pages 578–585.

Calvo, R. A. and Mac Kim, S. (2013). Emotions in text: Dimensional and categorical models. *Computational Intelligence*, 29(3):527–543.

Cohen, P. R. (1995). *Emperical Methods for Artificial Intelligence*. MIT Press, Cambridge, MA.

Davidson, P. and Innes-Ker, Å. (2014). Valence and arousal norms for Swedish affective words. Technical Report Volume 14, No. 2, Lund University.

Desmet, B. and Hoste, V. (2013). Emotion detection in suicide notes. *Expert Systems with Applications*, 40(16):6351–6358.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923.

Eilola, T. M. and Havelka, J. (2010). Affective norms for 210 British English and Finnish nouns. *Behavior Research Methods*, 42(1):134–140.

Ekman, P. (1992). An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200.

Ferré, P., Guasch, M., Martínez-García, N., Fraga, I., and Hinojosa, J. A. (2017). Moved by words: Affective ratings for a set of 2,266 Spanish words in five discrete emotion categories. *Behavior Research Methods*, 49(3):1082–1094.

Guasch, M., Ferré, P., and Fraga, I. (2015). Spanish norms for affective and lexico-semantic variables for 1,400 words. *Behavior Research Methods*, 48(4):1358–1369.

Hatzivassiloglou, V. and McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *ACL-EACL 1997 — Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics & 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 174–181.

Hinojosa, J. A., Martínez-García, N., Villalba-García, C., Fernández-Folgueiras, U., Sánchez-Carmona, A., Pozo, M. A., and Montoro, P. R. (2016a). Affective norms of 875 Spanish words for five discrete emotional categories and two emotional dimensions. *Behavior Research Methods*, 48(1):272–284.

Hinojosa, J. A., Rincón-Pérez, I., Romero-Ferreiro, M. V., Martínez-García, N., Villalba-García, C., Montoro, P. R., and Pozo, M. A. (2016b). The Madrid Affective Database for Spanish (MADS): Ratings of dominance, familiarity, subjective age of acquisition and sensory experience. *PLoS One*, 11(5):e0155866.

Imbir, K. K. (2016). Affective Norms for 4900 Polish Words Reload (ANPW_R): Assessments for valence, arousal, dominance, origin, significance, concreteness, imageability and, age of acquisition. *Frontiers in Psychology*, 7:#1081.

Kanske, P. and Kotz, S. A. (2010). Leipzig affective norms for German: A reliability study. *Behavior Research Methods*, 42(4):987–991.

Li, S., Xu, J., Zhang, D., and Zhou, G. (2016). Two-view label propagation to semi-supervised reader emotion classification. In *COLING 2016 — Proceedings of the 26th International Conference on Computational Linguistics*, volume Technical Papers, pages 2647–2655.

Liu, B. (2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press, New York, NY.

Monnier, C. and Syssau, A. (2014). Affective norms

for French words (FAN). *Behavior Research Methods*, 46(4):1128–1137.

Montefinese, M., Ambrosini, E., Fairfield, B., and Mammarella, N. (2014). The adaptation of the Affective Norms for English Words (ANEW) for Italian. *Behavior Research Methods*, 46(3):887–903.

Moors, A., De Houwer, J., Hermans, D., Wanmaker, S., van Schie, K., Van Harmelen, A.-L., De Schryver, M., De Winne, J., and Brysbært, M. (2013). Norms of valence, arousal, dominance, and age of acquisition for 4,300 Dutch words. *Behavior Research Methods*, 45(1):169–177.

Palogiannidi, E., Koutsakis, P., Iosif, E., and Potamianos, A. (2016). Affective lexicon creation for the Greek language. In *LREC 2016 — Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 2867–2872.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *EMNLP 2002 — Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 79–86.

Pinheiro, A. P., Dias, M., Pedrosa, J., and Soares, A. P. (2017). Minho Affective Sentences (MAS): Probing the roles of sex, mood, and empathy in affective ratings of verbal stimuli. *Behavior Research Methods*, 49(2):698–716.

Redondo, J., Fraga, I., Padrón, I., and Comesaña, M. (2007). The Spanish adaptation of ANEW (Affective Norms for English Words). *Behavior Research Methods*, 39(3):600–605.

Ric, F., Alexopoulos, T., Muller, D., and Aubé, B. (2013). Emotional norms for 524 French personality trait words. *Behavior Research Methods*, 45(2):414–421.

Riegel, M., Wierzba, M., Wypych, M., Żurawski, L., Jednoróg, K., Grabowska, A., and Marchewka, A. (2015). Nencki Affective Word List (NAWL): The cultural adaptation of the Berlin Affective Word List–Reloaded (BAWL–R) for Polish. *Behavior Research Methods*, 47(4):1222–1236.

Russell, J. A. and Mehrabian, A. (1977). Evidence for a three-factor theory of emotions. *Journal of Research in Personality*, 11(3):273–294.

Schmidtke, D. S., Schröder, T., Jacobs, A. M., and Conrad, M. (2014). ANGST: Affective Norms for German Sentiment Terms, derived from the affective norms for English words. *Behavior Research Methods*, 46(4):1108–1118.

Schouten, K. and Frasincar, F. (2016). Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(3):813–830.

Sedoc, J., Preoţiuc-Pietro, D., and Ungar, L. H. (2017). Predicting emotional word ratings using distributional representations and signed clustering. In *EACL 2017 — Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: Short Papers, pages 564–571.

Sianipar, A., van Groenestijn, P., and Dijkstra, T. (2016). Affective meaning, concreteness, and subjective frequency norms for Indonesian words. *Frontiers in Psychology*, 7:#1907.

Soares, A. P., Comesaña, M., Pinheiro, A. P., Simões, A., and Frade, C. S. (2012). The adaptation of the Affective Norms for English Words (ANEW) for European Portuguese. *Behavior research methods*, 44(1):256–269.

Sobhani, P., Mohammad, S. M., and Kiritchenko, S. (2016). Detecting stance in tweets and analyzing its interaction with sentiment. In **SEM 2016 — Proceedings of the 5th Joint Conference on Lexical and Computational Semantics @ ACL 2016*, pages 159–169.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP 2013 — Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Stadthagen-Gonzalez, H., Imbault, C., Pérez-Sánchez, M. A., and Brysbært, M. (2017). Norms of valence and arousal for 14,031 Spanish words. *Behavior Research Methods*, 49(1):111–123.

Stevenson, R. A., Mikels, J. A., and James, T. W. (2007). Characterization of the Affective Norms for English Words by discrete emotional categories. *Behavior Research Methods*, 39(4):1020–1024.

Strapparava, C. and Mihalcea, R. (2007). SemEval 2007 Task 14: Affective text. In *SemEval 2007 — Proceedings of the 4th International Workshop on Semantic Evaluations @ ACL 2007*, pages 70–74.

Võ, M. L. H., Conrad, M., Kuchinke, L., Urton, K., Hofmann, M. J., and Jacobs, A. M. (2009). The Berlin Affective Word List Reloaded (BAWL–R). *Behavior Research Methods*, 41(2):534–538.

Wang, J., Yu, L.-C., Lai, K. R., and Zhang, X. (2016). Community-based weighted graph model for valence-arousal prediction of affective words. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(11):1957–1968.

Warriner, A. B., Kuperman, V., and Brysbært, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, 45(4):1191–1207.

Wierzba, M., Riegel, M., Wypych, M., Jednoróg, K., Turnau, P., Grabowska, A., and Marchewka, A. (2015). Basic emotions in the Nencki Affective Word List (NAWL BE): New method of classifying emotional stimuli. *PLoS One*, 10(7):e0132305.

Yao, Z., Wu, J., Zhang, Y., and Wang, Z. (2017). Norms of valence, arousal, concreteness, familiarity, imageability, and context availability for 1,100 Chinese words. *Behavior Research Methods*, 49(4):1374–1385.

Yu, L.-C., Lee, L.-H., Hao, S., Wang, J., He, Y., Hu, J., Lai, K. R., and Zhang, X. (2016). Building Chinese affective resources in valence-arousal dimensions. In *NAACL-HLT 2016 — Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 540–545.

Unfolding the External Behavior and Inner Affective State of Teammates through Ensemble Learning: Experimental Evidence from a Dyadic Team Corpus

Aggeliki Vlachostergiou[1], Mark Dennison[2], Catherine Neubauer[2], Stefan Scherer[3], Peter Khooshabeh[2], Andre Harrison[2]
[1]Image,Video and Multimedia Systems Lab, National Technical University of Athens, Athens, Greece
[2]US Army Research Lab, Los Angeles, CA, USA
[3]Institute of Creative Technologies, University of Southern California, Los Angeles, CA, USA
aggelikivl@image.ntua.gr, {mark.s.dennisson.ctr, catherine.e.neubauer2.ctr, peter.khooshabehadeh2.civ, andre.v.harrison2.civ}@mail.mil, scherer@ict.usc.edu

Abstract

The current study was motivated to understand the relationship between the external behavior and inner affective state of two team members ("instructor"-"defuser") during a demanding operational task (i.e., bomb defusion). In this study we assessed team member's verbal responses (i.e., length of duration) in relation to their external as well as internal affective cues. External behavioral cues include defuser's verbal expressions while inner cues are based on physiological signals. More specifically, we differentiate between "defusers'" physiological patterns occurring after the "instructor's" turns according to whether they belong to a short or a long turn-taking response interval. Based on the assumption that longer turn-taking behaviors are likely to be caused by demanding cognitive task events and/or stressful interactions, we hypothesize that inner mechanisms produced in these intense affective activity intervals will be reflected on defuser's physiology. A dyadic team corpus was used to examine the association between the "defusers" physiological signals following the "instructor's" questions to predict whether they occurred in a short or long turn-taking period of time. The results suggest that an association does exist between turn taking and inner affective state. Additionally, it was our goal to further unpack this association by creating diverse ensembles. As such, we studied various base learners and different ensemble sizes to determine the best approach towards building a stable diverse ensemble that generalizes well on the external and inner cues of individuals.
Keywords: Turn-takings, Physiological features, MEAP, Dyadic Team Corpus, Ensemble Learning

1. Introduction

Lack of emotional expressivity is one of the main deficits that characterizes periods of stress when team members perform highly cognitive cooperative tasks. In cases such as this, teammates may find it more difficult to express their conscious feelings and show different patterns in perceiving and conveying emotional information when working together to meet a common goal(Jones and George, 1998; Prati et al., 2003). In light of these observations, having a way to monitor the internal state of teammates within such contexts might provide us new insights with respect to the mechanisms of their interaction and affectivity.

During high workload and high stress tasks, the sympathetic nervous system is accountable for activating glands and organs that are responsible for defending the body from perceived threats. This activation is associated with changes in arousal that are further influenced by emotion, cognition or attention. Stress results in increased sympathetic activity and can be tracked for example through bodily reactions, such as an increase in heart rate, greater blood flow to extremities and an increase in the respiration rate etc. Thus, a combination of more than one physiological indicator would be considered a more sensitive measure of changes in stress and can be used to provide estimations of emotion, arousal and general cognition (McEwen, 2007).

In this paper, we shed light on the association between two team members' physiological states and their speech, measured via their conversational turn-taking duration. Team members in highly-demanding operational tasks do not often notice triggers that cause them to be emotionally and mentally stressed (Murphy, 1996; Stein, 2001). Thus, they might communicate with their teammates and express their emotions in ways that may not be noticed in observable audio-visual cues. For instance, one would expect that asking an individual to disarm a simulated bomb would result in high levels of (internal) stress. This inherent gap between teammates' external observable behavior and their inner affective state is not well understood and can be potentially bridged by monitoring their physiology. The duration of response utterances is also reported to be very important, as it can be indicative of conflicting mental and stress procedures (Raux and Eskenazi, 2009). Because physiological indicators reflect aspects of underlying mental states and specifically the amount of distress (El-Sheikh et al., 1989), we explore whether physiological signals of long and short response utterance durations exhibit different physiological patterns. To further capture and interpret this ongoing and evolving interplay, we examine the use of two ensemble learning strategies. We believe that the investigation of physiological changes during the response periods can provide a better understanding of a team dynamics.

2. Related Work

There has been a lot of research on dialogue dynamics as well as the relationship of stress to underlying physiology, but these fields have largely been separate in the literature. For example, work on turn-taking behavior in dialogue systems (Raux and Eskenazi, 2009) could benefit from an understand of interpersonal dynamics of physiological stress response during a cooperative task (Dennison et al., 2016). Links between turn-taking behavioral responses and physiology have been studied for assessing how adults' anger

levels affected children (El-Sheikh et al., 1989). Moreover, previous studies have shown the advantages of using ensemble learning in both unimodal (Schuller et al., 2005b; Scherer et al., 2008; Schels and Schwenker, 2010) and multimodal behavioral analysis(Glodek et al., 2011; Schels et al., 2012; Schuller et al., 2005a).

To the best of our knowledge, there is no experimental evidence of applying ensemble learning to study the link between external behavior of turn-taking responses (Section 4.) and inner affective states inferred from physiological signal indicators (Section 5.) of two teammates (instructor-defuser) trying to disarm a simulated bomb (Section 3.)(Neubauer et al., 2016). This work is an effort to unfold this association based on the experimental evidence from the Dyadic Team Corpus. Our results indicate that physiological patterns convey information about the defuser's inner state, because they differ according to the duration of turn-taking behavioral replies with respect to the instructor's turns (Section 6.2.). Finally, our results are further enhanced through ensemble learning methods, which outperform the individual base learners in most cases and interesting observations are discussed in the Section 7..

3. Corpus Description

The dyadic cooperative team corpus (Neubauer et al., 2016) employed a 2x2 between subjects design resulting in a total of 2 experimental conditions with 20-gender-matched pairs in the following two conditions: The Ice Breaker conversation (IB) condition which consisted of allowing teammates to garner familiarity through a series of "getting to know you" questions prior to the start of the task and the Control (CT) condition, where teammates simply began the task with no prior familiarity. The corpus consists of a series of simulated "bomb defusion" scenarios. In each scenario one team member served as the "defuser" and one team member served as the "instructor". The "instructor" was given a manual with instructions on how to diffuse the bomb. The "instructor" was told that it was their responsibility to provide information that would allow the defuser to successfully complete the task. After each separate task the team members switched roles (i.e., each team member was given the opportunity to be both the "defuser" and the "instructor" twice during the main task), which resulted in a total of 4 main tasks, each lasting an average of 5mins. For this work, we take into account only 10-gender-matched pairs (5 from each condition) and we examine only the case in which participant A is the instructor and participant B is the defuser, (i.e. we didn't examine the case of asking members to switch roles).

4. Turn-taking behavioral responses

One of the main indicators of an interactional speech episode is often called a "turn-taking" and is defined as the time duration between the end of someone's turn and the beginning of the other interlocutor's corresponding turn. Turn-taking responses may span from very short to very long, which may indicate shorter or longer emotional and stressful episodes. In a similar way, in our teammate corpus, longer turn taking behavioral responses provided valuable information about the defuser's perceived cognition and affective state, reflected their external observable as well as implicit inner affective states. We choose to investigate that type of interactional context between the two teammates, motivated by the fact that the instructor's behavior is more controllable, thus minimizing the effect of the instructor's variability on the defuser's behavior.

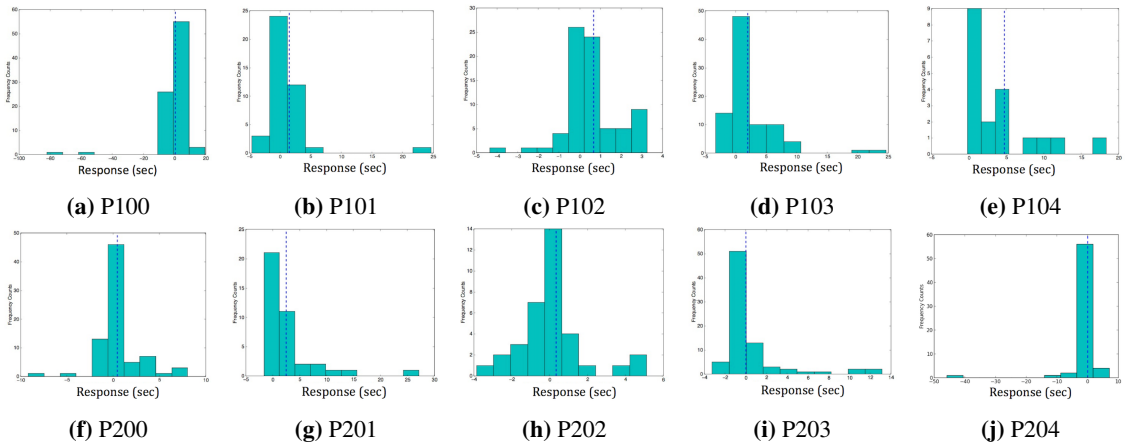
To further distinguish between short and long turn-taking behavioral responses (Figure 1) we draw a threshold at the 70th percentile of response values. This threshold was computed empirically after plotting the histograms of turn-taking behavioral response instances from the data of each defuser separately. Negative values of this measure mean that the defuser started talking before the instructor had finished the current turn. Phenomena such as overlapped speech and very short utterances are aligned with high levels of stress in highly-demanding operational tasks (Heldner and Edlund, 2010).

After carefully inspecting various turn-taking behavioral instances in our corpus, we came across a number of interesting tendencies. There were examples during the interactional context in which the instructor explained how the blue and red wires are connected. In that case, the defuser's reply is short (i.e., the defuser uses words such as ok/yes/no). Then, as a follow up, the instructor explained with more detail the process of bomb defusion to check whether the defuser is really following his instructions. In the first case, where the statement is simple and elicits low cognitive effort, a short reply occurred, while a long one occurred in the second case, where the defuser repeated the instructor's guidelines to confirm that he correctly understood the task. In this case, it was expected that the defuser was much more mentally alert.

5. Extraction of Physiological Features

A BIOPAC MP150, with a standard lead II electrode configuration was used to record electrocardiography (ECG), real-time changes in blood pressure, and impedance cardiography (ZKG). Continuous data were recorded for each participant throughout each task and analyzed offline. The raw time series for each task segmented into thirty second intervals relative to the end of each session, such that a few seconds from the beginning of the "bomb defusion" task were cut out. This was done because a minimum of 30 seconds of cardiovascular data are necessary for further analysis. The Moving Ensemble Average Program (MEAP)(Cieslak, 2017) was used to extract features from the data by computing an ensembled average over each epoch.

We extracted 24 cardiovascular features. Some of these features, presented in Table 1, included heart rate (HR), LVET (left ventral (systolic) ejection time), p_time, s_time, t_time, x_time, systole_time, pre-ejection period (PEP), ventricular contractility (VC), cardiac output (CO) and total peripheral resistance (TPR). PEP is the time from the onset of the heart muscle depolarization to the opening of the aortic valve. When PEP decreases, VC increases. VC has been shown to be related to task engagement (Newlin and Levenson, 1979; Richter and Gendolla, 2009; Spangler and Friedman, 2015; Seery, 2011). CO is the amount of blood pumped in liters per minute. TPR reflects vasodilation (more blood flow) and vasoconstriction (less blow



(a) P100 **(b)** P101 **(c)** P102 **(d)** P103 **(e)** P104

(f) P200 **(g)** P201 **(h)** P202 **(i)** P203 **(j)** P204

Figure 1: Frequency counts (y axis) for 10 defusers in IB and CT conditions and distributions of their responses (x axis) measured in seconds with respect to the instructor's turns. The vertical dashed blue line distinguishes between the short and long replies.

Participants	Condition	Selected Physiological Features
P100	IB	s_time, systole_time
P101	IB	hr
P102	IB	lvet,p_time, x_time
P103	IB	hr
P104	IB	t_time
P200	CT	diastole_time
P201	CT	hr
P202	CT	pep
P203	CT	hr
P204	CT	hr

Table 1: Most frequently selected Physiological Features for ten defusers, five from the IB and CT conditions respectively during the "bomb defusion" task. The features are labeled as follows: heart rate (hr), left ventral (systolic) ejection time (LVET) and pre-ejection period (pep).

flow), which are related to parasympathetic and sympathetic activity, respectively. Prior work has shown that TPR unambiguously increases when an individual is in a threat state and decreases in a challenge state, whereas CO either remains unchanged or decreases in a threat state and increases in a challenge state (Tomaka et al., 1997).

6. Experiments

The purpose of our experiment is to unfold the direct link between levels of external socio-cognitive behavioral demand and inner-affective mechanisms. Through an ensemble learning task we attempted to exploit different behaviors of the selected base learners to enhance the accuracy of our overall learning system. Our aim is to show that defusers' physiological patterns differ between periods of longer and shorter replies and that there exist a range of stress levels across defusers.

6.1. Methodology

Feature Selection: Due to the fact that some of the features are highly correlated, we reduce the set with correlation feature selection (CFS). CFS selects features that correlate with the class label but are not correlated with previously selected features.

Ensemble Learning: To maintain diversity of base learners (Dietterich, 2002) we use two heterogeneous learner schemes: Voting (Shipp and Kuncheva, 2002) and Meta-learning (Wolpert, 1992). Under the Voting scheme, we combine the individual base by applying the average combination rule to the outputs. Meta-learning employs several base learners to get class predictions, which are then used by a meta-learning algorithm during the training phase to predict when the base learners are incorrect. Additionally, we comment on the ensemble size of the learners. Both ensemble schemes were built by combining the following base learners: K-Nearest Neighbor with K=5 (KNN)[1], Naive Bayes (NB), Decision Tree (DT), Random Tree (RT), Support Vector Machines with RBF kernel (SVM-RBF), Multi-layer Perceptron (MLP) and Random Forest (RF)(Breiman, 2001)[2]. The experiments were performed using leave-one-instance-out cross validation, where instance denotes a turn-taking behavioral response. We applied this approach for every defuser separately, for both conditions IB and CT respectively, as we wanted to investigate the unique individual trends of each defuser participant with respect to their behavior, their physiology and their their experimental condition (Ice Breaker or Control).

6.2. Results

The individual base learners and the ensemble learning methods chosen for our study are shown in Table 2.

Our experimental results range from 43.75% to 88.89%, suggesting that physiological signals contain information relevant to the amount of behavioral verbal replies. Additionally, we notice a great difference in performance across defusers, underlying once again the individual traits of every defuser. Particularly, the selected physiological cues of defusers P100, P102, P202, P203 and P204 appear to be more closely associated to the type of behavioral reply instances (short/long) compared to the corresponding patterns of P101, P103, P104 and P201 defusers.

[1] K=5 was empirically found to give better performance considering the limitation of small number of instances for 2 defusers (P104 and P202 with 9 and 10 instances respectively).

[2] This learner as a base one is robust and works relatively well without excessive need of meta parameter tuning.

		Base Learners							Ensemble Learners	
Participants	Condition	KNN(5)	NB	DT	RT	SVM-RBF	MLP	RF	Voting	Meta-learning
P100	IB	68.97	**75.86**	62.07	**75.86**	72.41	65.52	**75.86**	72.41	72.41
P101	IB	43.75	56.25	50.00	50.00	43.75	50.00	50.00	43.75	**62.50**
P102	IB	**80.00**	73.33	73.33	73.33	40.00	66.67	73.33	73.33	73.33
P103	IB	34.38	46.88	53.13	43.75	59.38	53.13	43.75	59.38	**62.50**
P104	IB	66.67	44.44	**88.89**	66.67	55.56	66.67	66.67	55.56	55.56
P200	CT	77.78	66.67	**88.89**	**88.89**	**88.89**	77.78	**88.89**	**88.89**	83.33
P201	CT	43.75	43.75	56.25	18.75	31.25	50.00	25.00	31.25	**62.50**
P202	CT	70.00	80.00	**90.00**	60.00	60.00	80.00	80.00	60.00	70.0
P203	CT	**88.00**	**88.00**	**88.00**	72.00	**88.00**	84.00	72.00	**88.00**	**88.00**
P204	CT	**77.78**	66.67	**77.78**	**77.78**	**77.78**	66.67	66.67	**77.78**	66.67

Table 2: The individual base learners and the ensemble learning methods chosen for ten defusers, five from the IB and CT conditions respectively during the "bomb defusion" task. The best method(s) for every participant is highlighted.

Ensemble learning performance and ensemble size: We notice that ensembles for the two different combination schemes either outperform the best individual base learner (P101, P103, P201) or reach similar performance with that (P200, P203, P204). After experimenting with a size ranging from 5 to 10 base learners, we present only those, whose combination determined the best approach towards building a stable diverse ensemble that generalizes well on the external and inner cues of individual. Furthermore, we have experimented with an odd and even number of ensemble size. Experimentally, we found that using an odd number for the ensemble size provides a higher learning performance. That could be explained if we consider that, when an even number of base learners is used, there is a potential for a tie when half of the base learners vote for one class while the other half vote for the opposite class.

Most frequently selected physiological features: We elaborate on the features presented in Table 1 in terms of their importance with respect to the "bomb defusion" task. We observe that for 50% of the defusers the most selected physiological signal is HR. Based on this, we assume that HR is associated with arousal levels and is of high importance for the examined task for this work. Regarding the remaining selected physiological features, we notice that these features range across defusers. This finding enhances the original assumption of uniqueness of individual personal traits across participants. Finally, the former finding is also aligned with our experimental results and observations that suggest that the "teammate prior familiarity" parameter does not have an impact on our task.

7. Discussion

As discussed in Section 6.2., there is a wide variability across defusers with respect to the given task. This observation indicates that there might be mechanisms triggered in defusers with high learning accuracy, reflected their physiological signals, which are not present in defusers with low learning performance (i.e. P101, P103, P201). It is also noteworthy that for these three defusers, the selected physiological feature is HR. To further elaborate on this tendency, we go through the audiovisual recordings and the HR signals. We notice that there is a difference in the arousal levels (i.e., stress) with respect to the type of behavioral replies (short/long) and that arousal affectivity is present both in short and long turn-taking responses, depending on the defuser.

More specifically, we come across examples of defusers who took a long time to respond after having given a wrong answer once and were asked to try again to confirm the bomb defusion steps. Hence, it appears that the task was a sufficiently stressful stimulus for them. In these long turn-taking examples, it is also reasonable to assume that high cognitive activity or stressor events occurred. At the same time, high levels of arousal are noticed in short turn-taking examples, in which for example the defuser uses words such ok/yes/no. This tendency is not aligned with the "bomb defusion" task, considering that we were expecting that short turn-taking examples would reflect low levels of arousal. On the contrary, our observation suggests that even though there may be no obvious (audible/visible) signals of arousal, physiological signals may provide a complementary, not overlaid though, view of a person's state. This finding is of particular importance, especially in cognitively demanding tasks in which one of the teammates manipulates the discussion and is also aligned with previous research studies (Tomaka et al., 1997; Gellatly and Meyer, 1992; Calkins and Fox, 2002).

8. Conclusions and Future Work

This study provides an analysis of physiological signals in a dyadic team "bomb defusion" scenario in association with their expressive behavioral cues. The results suggest that physiological responses convey information about the defuser's inner state. They also reflect the amount of the defuser's verbal responses with respect to a stimuli and can be further linked with the amount of underlying socio-cognitive activity, which is not always obvious through traditional observational methods. Last, we proposed two existing ensemble learning methods which are new to the field of generalizing external and inner cues of speakers, showing that these methods can yield improvements over traditional analysis methods.

One of the limitations of our study is its reliance on a small part of the corpus. Future plans include the analysis and discussion of the identified trends over the whole corpus, as well as the examination of the uniqueness of personal traits of every participant after switching roles (each team member was given the opportunity to be both the defuser and the instructor). Also, considering that this study relied on observational cues concerning turn taking dura-

tion measures, we believe that the examination of expressive cues with a more detailed analysis of participants' lexical features will provide an insight into whether these can be linked with their inner physiological signals. In terms of the lexical features, we would like to focus on the number of words, the length of the utterances, the number of laughs, the richness of the vocabulary as well as the use of backchannels in terms of short feedback such as "mmhmm", "yeah'.

Additionally, the investigation of singular pronouns (I, me, mine), assents (OK, yes), non-fluencies (hm, umm), fillers (I mean, you know) prepositions or words indicating prior familiarity could extend the pool of the used features. Apart from that, we intend to apply more advanced lexical modeling such as topic modeling, to better capture word usage, word choice and to unfold all aspects of the defuser's specific grammar employed in such stressful interactions. But, mostly we do believe that such an investigation could provide an insight with respect to the relevant vocabulary that is used in such particular tasks and the speaking style of every team member while sessions progress. This corpus serves as a technical springboard for developing dialogue agents that not only capture turn-taking behavior in a stressful task, but also underlying physiological states during the dyadic task.

9. References

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Calkins, S. D. and Fox, N. A. (2002). Self-regulatory processes in early personality development: A multilevel approach to the study of childhood social withdrawal and aggression. *Development and Psychopathology*, 14(3):477–498.

Cieslak, M. (2017). Moving ensemble averaging program. `https://github.com/mattcieslak/MEAP`.

Dennison, M., Neubauer, C., Passaro, T., Harrison, A., Scherer, S., and Khooshabeh, P. (2016). Using cardiovascular features to classify state changes during cooperation in a simulated bomb diffusal task. In *Proceedings of the Physiologically Aware Virtual Agents Workshop*. IEEE.

Dietterich, T. G. (2002). Ensemble learning. *The handbook of brain theory and neural networks*, 2:110–125.

El-Sheikh, M., Cummings, E. M., and Goetsch, V. L. (1989). Coping with adults' angry behavior: Behavioral, physiological, and verbal responses in preschoolers. *Developmental Psychology*, 25(4):490.

Gellatly, I. R. and Meyer, J. P. (1992). The effects of goal difficulty on physiological arousal, cognition, and task performance. *Journal of Applied Psychology*, 77(5):694.

Glodek, M., Tschechne, S., Layher, G., Schels, M., Brosch, T., Scherer, S., Kächele, M., Schmidt, M., Neumann, H., Palm, G., et al. (2011). Multiple classifier systems for the classification of audio-visual emotional states. *Affective Computing and Intelligent Interaction*, pages 359–368.

Heldner, M. and Edlund, J. (2010). Pauses, gaps and overlaps in conversations. *Journal of Phonetics*, 38(4):555–568.

Jones, G. R. and George, J. M. (1998). The experience and evolution of trust: Implications for cooperation and teamwork. *Academy of management review*, 23(3):531–546.

McEwen, B. S. (2007). Physiology and neurobiology of stress and adaptation: central role of the brain. *Physiological reviews*, 87(3):873–904.

Murphy, L. R. (1996). Stress management in work settings: a critical review of the health effects. *American Journal of Health Promotion*, 11(2):112–135.

Neubauer, C., Woolley, J., Khooshabeh, P., and Scherer, S. (2016). Getting to know you: a multimodal investigation of team behavior and resilience to stress. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 193–200. ACM.

Newlin, D. B. and Levenson, R. W. (1979). Pre-ejection period: Measuring beta-adrenergic influences upon the heart. *Psychophysiology*, 16(6):546–552.

Prati, L. M., Ceasar, D., Ferris, G. R., Ammeter, A. P., and Buckley, M. R. (2003). Emotional intelligence, leadership effectiveness, and team outcomes. *International Journal of Organizational Analysis*, 11(1):21.

Raux, A. and Eskenazi, M. (2009). A finite-state turn-taking model for spoken dialog systems. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 629–637. ACL.

Richter, M. and Gendolla, G. (2009). The heart contracts to reward: monetary incentives and pre-ejection period. *Psychophysiology*, 46(3):451.

Schels, M. and Schwenker, F. (2010). A multiple classifier system approach for facial expressions in image sequences utilizing gmm supervectors. In *20th International Conference on Pattern Recognition*, pages 4251–4254. IEEE.

Schels, M., Glodek, M., Meudt, S., Schmidt, M., Hrabal, D., Böck, R., Walter, S., and Schwenker, F. (2012). Multi-modal classifier-fusion for the classification of emotional states in woz scenarios. In *1st International Conference on Affective and Pleasurable Design*, pages 5337–5346.

Scherer, S., Schwenker, F., and Palm, G. (2008). Emotion recognition from speech using multi-classifier systems and rbf-ensembles. *Speech, Audio, Image and Biomedical Signal Processing using Neural Networks*, pages 49–70.

Schuller, B., Müller, R., Lang, M., and Rigoll, G. (2005a). Speaker independent emotion recognition by early fusion of acoustic and linguistic features within ensembles. In *Ninth European Conference on Speech Communication and Technology*.

Schuller, B., Reiter, S., Muller, R., Al-Hames, M., Lang, M., and Rigoll, G. (2005b). Speaker independent speech emotion recognition by ensemble classification. In *IEEE International Conference on Multimedia and Expo,(ICME) 2005.*, pages 864–867. IEEE.

Seery, M. D. (2011). Challenge or threat? cardiovascular indexes of resilience and vulnerability to potential

stress in humans. *Neuroscience & Biobehavioral Reviews*, 35(7):1603–1610.

Shipp, C. A. and Kuncheva, L. I. (2002). Relationships between combination methods and measures of diversity in combining classifiers. *Information fusion*, 3(2):135–148.

Spangler, D. P. and Friedman, B. H. (2015). Effortful control and resiliency exhibit different patterns of cardiac autonomic control. *International Journal of Psychophysiology*, 96(2):95–103.

Stein, F. (2001). Occupational stress, relaxation therapies, exercise and biofeedback. *Work: A Journal of Prevention, Assessment and Rehabilitation*, 17(3):235–245.

Tomaka, J., Blascovich, J., Kibler, J., and Ernst, J. M. (1997). Cognitive and physiological antecedents of threat and challenge appraisal. *Journal of Personality and Social Psychology*, 73(1):63.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 5(2):241–259.

Understanding Emotions: A Dataset of Tweets to Study Interactions between Affect Categories

Saif M. Mohammad and Svetlana Kiritchenko
National Research Council Canada
{saif.mohammad,svetlana.kiritchenko}@nrc-cnrc.gc.ca

Abstract

Human emotions are complex and nuanced. Yet, an overwhelming majority of the work in automatically detecting emotions from text has focused only on classifying text into positive, negative, and neutral classes, and a much smaller amount on classifying text into basic emotion categories such as joy, sadness, and fear. Our goal is to create a single textual dataset that is annotated for many emotion (or affect) dimensions (from both the basic emotion model and the VAD model). For each emotion dimension, we annotate the data for not just coarse classes (such as anger or no anger) but also for fine-grained real-valued scores indicating the intensity of emotion (anger, sadness, valence, etc.). We use Best–Worst Scaling (BWS) to address the limitations of traditional rating scale methods such as inter- and intra-annotator inconsistency by employing comparative annotations. We show that the fine-grained intensity scores thus obtained are reliable (repeat annotations lead to similar scores). We choose Twitter as the source of the textual data we annotate because tweets are self-contained, widely used, public posts, and tend to be rich in emotions. The new dataset is useful for training and testing supervised machine learning algorithms for multi-label emotion classification, emotion intensity regression, detecting valence, detecting ordinal class of intensity of emotion (slightly sad, very angry, etc.), and detecting ordinal class of valence (or sentiment). We make the data available for the recent SemEval-2018 Task 1: Affect in Tweets, which explores these five tasks. The dataset also sheds light on crucial research questions such as: which emotions often present together in tweets?; how do the intensities of the three negative emotions relate to each other?; and how do the intensities of the basic emotions relate to valence?

Keywords: emotion intensity, valence, arousal, dominance, basic emotions, crowdsourcing, sentiment analysis

1. Introduction

Emotions are central to how we perceive the world, how we make sense of it, and how we make day-to-day decisions. Emotions are also complex and nuanced. Even though humans are known to perceive hundreds of different emotions, there is still little agreement on how best to categorize and represent emotions. According to the basic emotion model (aka the categorical model) (Ekman, 1992; Plutchik, 1980; Parrot, 2001; Frijda, 1988), some emotions, such as joy, sadness, fear, etc., are more basic than others, and that these emotions are each to be treated as separate categories. Each of these emotions can be felt or expressed in varying intensities. Here, *intensity* refers to the degree or amount of an emotion such as anger or sadness.[1] As per the valence–arousal–dominance (VAD) model (Russell, 2003), emotions are points in a three-dimensional space of valence (positiveness–negativeness), arousal (active–passive), and dominance (dominant–submissive).

Both the categorical model and the dimensional model of emotions have a large body of work supporting them, and offer different perspectives that help our understanding of emotions. However, there is very little work relating the two models of emotion with each other. Much of the past work on textual utterances such as sentences and tweets, is based on exactly one or the other model (not both).[2] For example, corpora annotated for emotions are either annotated only for the basic emotions (Mohammad and Bravo-Marquez, 2017b; Strapparava and Mihalcea, 2007; Alm et al., 2005) or only for valence, arousal, and dominance (Yu et al., ; Mohammad et al., 2017; Nakov et al., 2016).

Within Natural Language Processing, an overwhelming majority of the work has focused on classifying text into positive, negative, and neutral classes (valence classification), and a much smaller amount on classifying text into basic emotion categories such as joy, sadness, and fear. A key obstacle in developing algorithms for other emotion-related tasks, especially those involving fine-grained intensity scores, is the lack of large reliably labeled datasets.

The goal of this work is to create, for the first time, a large single textual dataset annotated for many emotion (or affect) dimensions (from both the basic emotion model and the VAD model). Specifically, we annotate tweets for the emotions of people that posted the tweets—emotions that can be inferred solely from the text of the tweet. For each emotion dimension, we annotate the data for not just coarse classes (such as anger or no anger) but also for fine-grained real-valued scores indicating the intensity of emotion (anger, sadness, valence, etc.). The datasets can be used to train many different kinds of emotion analysis systems. Further, as (Mohammad and Bravo-Marquez, 2017a) showed, correlations across emotions means that training data for one emotion can be used to supplement training data for another emotion. We choose Twitter as the source of the textual data we annotate because tweets are self-contained, widely used, public posts, and tend to be rich in emotions. However, other choices such as weblogs, forum posts, and comments on newspaper articles are also suitable avenues for future work. Similarly, annotating for the emotions of the reader or emotions of those mentioned in the tweets are also suitable avenues for future work.

Mohammad and Bravo-Marquez (2017b) created the first datasets of tweets annotated for anger, fear, joy, and sadness intensities. Given a focus emotion, each tweet is annotated for intensity of the emotion felt by the speaker using a technique called *Best–Worst Scaling (BWS)*.

[1] Intensity is different from *arousal*, which refers to the extent to which an emotion is calming or exciting.

[2] There is some work on words that are annotated both for association to basic emotions as well as for valence, arousal, and dominance (Mohammad, 2018).

Dataset	Source of Tweets	Annotated In 2016	2017
E-c	Tweets-2016	-	✓
	Tweets-2017	-	✓
EI-reg, EI-oc	Tweets-2016	✓	-
	Tweets-2017	-	✓
V-reg, V-oc	Tweets-2016	-	✓
	Tweets-2017	-	✓

Table 1: The data and annotations in the AIT Dataset.

BWS is an annotation scheme that addresses the limitations of traditional rating scale methods, such as inter- and intra-annotator inconsistency, by employing comparative annotations (Louviere, 1991; Louviere et al., 2015; Kiritchenko and Mohammad, 2016; Kiritchenko and Mohammad, 2017). Annotators are given n items (an n-tuple, where $n > 1$ and commonly $n = 4$). They are asked which item is the *best* (highest in terms of the property of interest) and which is the *worst* (lowest in terms of the property of interest). When working on 4-tuples, best–worst annotations are particularly efficient because each best and worst annotation will reveal the order of five of the six item pairs. For example, for a 4-tuple with items A, B, C, and D, if A is the best, and D is the worst, then A $>$ B, A $>$ C, A $>$ D, B $>$ D, and C $>$ D. Real-valued scores of association between the items and the property of interest can be calculated from the BWS annotations (Orme, 2009; Flynn and Marley, 2014). Mohammad and Bravo-Marquez (2017b) collected and annotated 7,100 tweets posted in 2016. We will refer to the tweets alone as *Tweets-2016*, and the tweets and annotations together as the *Emotion Intensity Dataset* (or, *EmoInt Dataset*). This dataset was later used in the 2017 WASSA Shared Task on Emotion Intensity (EmoInt).[3]

We build on that earlier work by first compiling a new set of tweets posted in 2017 and annotating the new tweets for emotion intensity in a similar manner. We will refer to this new set of tweets as *Tweets-2017*. Similar to the work by Mohammad and Bravo-Marquez (2017b), we create four subsets annotated for intensity of fear, joy, sadness, and anger, respectively. However, unlike the earlier work, here a common dataset of tweets is annotated for all three negative emotions: fear, anger, and sadness. This allows one to study the relationship between the three basic negative emotions. The full set of tweets along with their emotion intensity scores can be used for developing automatic systems that predict emotion intensity (*emotion intensity regression*, or *EI-reg*, systems).

We also annotate tweets sampled from each of the four basic emotion subsets (of both Tweets-2016 and Tweets-2017) for degree of valence. This data can be used for developing systems that predict sentiment intensity (*valence regression*, or *V-reg*, systems). Annotations for degree of arousal and dominance are ongoing, and will be described in a subsequent paper. We leave the annotations for intensity of other basic emotions such as anticipation, disgust, and surprise for future work.

In addition to knowing a fine-grained score indicating degree of intensity, it is also useful to qualitatively ground the information on whether the intensity is high, medium, low, etc. Thus we manually identify ranges in intensity scores that correspond to these coarse classes. For each of the four emotions E, the 0 to 1 range is partitioned into the classes: *no E can be inferred, low E can be inferred, moderate E can be inferred,* and *high E can be inferred*. This data can be used for developing systems that predict the ordinal class of emotion intensity (*EI ordinal classification*, or *EI-oc*, systems). Since valence is a bi-polar scale, we partition the 0 to 1 range into: *very negative, moderately negative, slightly negative, neutral or mixed, slightly positive, moderately positive,* and *very positive mental state of the tweeter can be inferred*. This data can be used to develop systems that predict the ordinal class of valence (*valence ordinal classification*, or *V-oc*, systems).[4]

Finally, the full Tweets-2016 and Tweets-2017 datasets are annotated for the presence of eleven emotions: anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, and trust. This data can be used for developing *multi-label emotion classification*, or *E-c*, systems.

Table 1 shows the two stages in which the annotations were done: in 2016 as described in the work by Mohammad and Bravo-Marquez (2017b) and in 2017 as described in this paper. Together, we well refer to the joint set of tweets from Tweets-2016 and Tweets-2017 along with all the emotion-related annotations described above as the *SemEval-2018 Affect in Tweets Dataset* (or *AIT Dataset* for short), since this data was used to create the training, development, and test sets in the SemEval-2018 shared task of the same name – *SemEval-2018 Task 1: Affect in Tweets* shared task (Mohammad et al., 2018).[5] The shared task evaluates automatic systems for EI-reg, EI-oc, V-reg, V-oc, and E-c in three languages: English, Arabic, and Spanish.

We show that the intensity annotations in the AIT dataset have a high split-half reliability (between 0.82 and 0.92), indicating a high quality of annotation. (Split half reliability measures the average correlation between scores produced by two halves of the annotations—higher correlations indicate stable and consistent outputs.) The annotator agreement on the multi-label emotion annotations (E-c) is also well above the random agreement.

We show that certain pairs of emotions often present together in tweets. For example, the presence of anger is strongly associated with the presence of disgust, the presence of optimism is strongly associated with the presence of joy, etc. For some pairs of emotions (e.g., anger and disgust), this association is present in both directions, while for other pairs (e.g., love and joy), the association is markedly stronger in only one direction. We calculate the extent to which the intensities of affect dimensions correlate. Amongst anger, fear, and sadness the correlations are close to zero. Finally, we identify the tweets for which two affect scores correlate and the tweets for which they do not.

[3] http://saifmohammad.com/WebPages/EmoInt2017.html

[4] Note that valence ordinal classification is the traditional sentiment analysis task most commonly explored in NLP literature. The classes may vary from just three (positive, negative, and neutral) to five, seven, or nine finer classes.

[5] http://www.saifmohammad.com/WebPages/affectintweets.htm

2. The Affect in Tweets Dataset

We now present how we created the Affect in Tweets Dataset. For simplicity, we will describe the procedure as if all the tweets were collected at the same time. However, as stated earlier in the introduction, some tweets were collected in 2016 (part of the EmoInt dataset).

2.1. Compiling Tweets

We first compiled tweets to be included in the four EI-reg datasets corresponding to the four basic emotions: anger, fear, joy, and sadness. The EI-oc datasets include the same tweets as in EI-reg, that is, the Anger EI-oc dataset has the same tweets as in the Anger EI-reg dataset, the Fear EI-oc dataset has the same tweets as in the Fear EI-reg dataset, and so on. However, the labels for EI-oc tweets are ordinal classes instead of real-valued intensity scores. The V-reg dataset includes a subset of tweets from each of the four EI-reg emotion datasets. The V-oc dataset has the same tweets as in the V-reg dataset. The E-c dataset includes all the tweets from the four EI-reg datasets. The total number of instances in the E-c, EI-reg, EI-oc, V-reg, and V-oc is shown in the last column of Table 5.

2.1.1. Basic Emotion Tweets

For each of the four basic emotions, our goal was to create a dataset of tweets such that:

- The tweets are associated with various intensities (or degrees) of emotion.
- Some tweets have words clearly indicative of the basic emotion and some tweets do not.

A random collection of tweets is likely to have a large proportion of tweets not associated with the focus emotion, and thus annotating all of them for intensity of emotion is suboptimal. To create a dataset of tweets rich in a particular emotion, we used the following methodology. For each emotion X, we selected 50 to 100 terms that were associated with that emotion at different intensity levels. For example, for the anger dataset, we used the terms: *angry, mad, frustrated, annoyed, peeved, irritated, miffed, fury, antagonism,* and so on. For the sadness dataset, we used the terms: *sad, devastated, sullen, down, crying, dejected, heartbroken, grief, weeping,* and so on. We will refer to these terms as the *query terms*.

We identified the query terms for an emotion using many different ways to improve the overall diversity of the collected tweets:

- We looked up the *Roget's Thesaurus* to find categories that had the focus emotion word (or a close synonym) as the head word.[6] We chose all words listed within these categories to be the query terms for the corresponding focus emotion.
- We looked up a table of commonly used emojis to identify emojis associated with the four emotions.
- We identified simple emoticons such as ':)', ':(', and ':D' that are indicative of happiness and sadness.
- We identified synonyms of the four emotions in a word-embeddings space created from 11 million tweets with emoticons and emotion-word hashtags using word2vec (Mikolov et al., 2013).

The full list of query terms is made available on the SemEval-2018 Task 1 website.

We polled the Twitter API, over the span of two months (June and July, 2017), for tweets that included the query terms. We collected more than sixty million tweets. We discarded re-tweets (tweets that start with RT) and tweets with URLs. We created a subset of the remaining tweets by:

- selecting at most 50 tweets per query term;
- selecting at most one tweet for every tweeter–query term combination.

This resulted in tweet sets that are not heavily skewed towards any one tweeter or query term.

We randomly selected 1400 tweets from the joy set for annotation of intensity of joy. For the three negative emotions, we first randomly selected 200 tweets each from their corresponding tweet collections. These 600 tweets were annotated for all three negative emotions so that we could study the relationships between fear and anger, between anger and sadness, and between sadness and fear. For each of the negative emotions, we also chose 800 additional tweets, from their corresponding tweet sets, that were annotated only for the corresponding emotion. Thus, the number of tweets annotated for each of the negative emotions was also 1400 (600 common to the three negative emotions + 800 unique to the focus emotion). In 100 randomly chosen tweets from each emotion set (joy, anger, fear, and sadness), we removed the trailing query term (emotion-word hashtag, emoticon, or emoji) so that our dataset also includes some tweets with no clearly emotion-indicative terms.

Thus, the EI-reg dataset included 1400 new tweets for each of the four emotions. These were annotated for intensity of emotion. Note that the EmoInt dataset already included 1500 to 2300 tweets per emotion annotated for intensity. Those tweets were not re-annotated. The EmoInt EI-reg tweets as well as the new EI-reg tweets were both annotated for ordinal classes of emotion (EI-oc) as described in Section 2.2.3.

The new EI-reg tweets formed the EI-reg development (dev) and test sets in the AIT task; the number of instances in each is shown in the third and fourth columns of Table 5. The EmoInt tweets formed the training set. Manual examination of the new EI-reg tweets later revealed that it included some near-duplicate tweets. We kept only one copy of such pairs and discarded the other tweet. Thus the dev. and test set numbers add up to a little lower than 1400.

2.1.2. Valence, Arousal, and Dominance Tweets

Our eventual goal is to study how valence, arousal, and dominance (VAD) are related to joy, fear, sadness, and anger intensity. Thus, we created a single common dataset to be annotated for valence, arousal, and dominance, such that it includes tweets from the EI-reg datasets as described below. Specifically, the VAD annotation dataset of 2600 tweets included:

[6] The *Roget's Thesaurus* groups words into about 1000 categories. The head word is the word that best represents the meaning of the words within the category. The categories chosen were: 900 Resentment (for anger), 860 Fear (for fear), 836 Cheerfulness (for joy), and 837 Dejection (for sadness).

- From the new EI-reg tweets:
  - all 600 common negative emotion tweets,
  - 600 randomly chosen joy tweets,
- From EmoInt EI-reg tweets:
  - 600 randomly chosen joy tweets,
  - 200 each, randomly chosen tweets, for anger, fear, and sadness.

To study valence in sarcastic tweets, we also included 200 tweets that had hashtags #sarcastic, #sarcasm, #irony, or #ironic (tweets that are likely to be sarcastic). Thus the V-reg set included 2,600 tweets in total. The V-oc set included the same tweets as in the V-reg set.

2.1.3. Multi-Label Emotion Classification Tweets

We selected all of the 2016 and 2017 tweets in the four EI-reg datasets to form the E-c dataset, which is annotated for presence or absence of 11 emotions.

2.2. Annotating Tweets

We annotated all of our data by crowdsourcing. The tweets and annotation questionnaires were uploaded on the crowdsourcing platform, CrowdFlower.[7] All annotators for our tasks had already consented to the CrowdFlower terms of agreement. They chose to do our task among the hundreds available, based on interest and compensation provided. Respondents were free to annotate as many questions as they wished to. All the annotation tasks described in this paper were approved by the National Research Council Canada's Institutional Review Board, which reviewed the proposed methods to ensure that they were ethical.

About 5% of the tweets in each task were annotated internally beforehand (by the authors). These tweets are referred to as gold tweets. The gold tweets were interspersed with other tweets. If a crowd-worker got a gold tweet question wrong, they were immediately notified of the error. If the worker's accuracy on the gold tweet questions fell below 70%, they were refused further annotation, and all of their annotations were discarded. This served as a mechanism to avoid malicious annotations.

2.2.1. Multi-Label Emotion Annotation

We presented one tweet at a time to the annotators and asked two questions. The first was a single-answer multiple choice question:

> *Q1. Which of the following options best describes the emotional state of the tweeter?*
> *– anger (also includes annoyance, rage)*
> *– anticipation (also includes interest, vigilance)*
> *– disgust (also includes disinterest, dislike, loathing)*
> *– fear (also includes apprehension, anxiety, terror)*
> *– joy (also includes serenity, ecstasy)*
> *– love (also includes affection)*
> *– optimism (also includes hopefulness, confidence)*
> *– pessimism (also includes cynicism, no confidence)*
> *– sadness (also includes pensiveness, grief)*
> *– surprise (also includes distraction, amazement)*
> *– trust (also includes acceptance, liking, admiration)*
> *– neutral or no emotion*

[7] http://www.crowdflower.com

The second question was a checkbox question, where multiple options could be selected:

> *Q2. In addition to your response to Q1, which of the following options further describe the emotional state of the tweeter? Select all that apply.*

This question included the same first eleven emotion choices, but instead of neutral, the twelfth option was 'none of the above'. Example tweets were provided in advance with examples of suitable responses.

On the CrowdFlower task settings, we specified that we needed annotations from seven people for each tweet. However, because of the way the gold tweets were setup, they were annotated by more than seven people. The median number of annotations was still seven. In all, 303 people annotated between 10 and 4,670 tweets each. A total of 87,178 pairs of responses (Q1 and Q2) were obtained (see Table 4).

Annotation Aggregation: We determined the primary emotion for a tweet by simply taking the majority vote from the annotators. In case of ties, all emotions with the majority vote were considered the primary emotions for that tweet. We aggregated the responses from Q1 and Q2 to obtain the full set of labels for a tweet. We wanted to include not just the primary emotion, but all others that apply, even if their presence was more subtle. One of the criticisms for several natural language annotation projects has been that they keep only the instances with high agreement, and discard instances that obtain low agreements. The high agreement instances tend to be simple instantiations of the classes of interest, and are easier to model by automatic systems. However, when deployed in the real world, natural language systems have to recognize and process more complex and subtle instantiations of a natural language phenomenon. Thus, discarding all but the high agreement instances does not facilitate the development of systems that are able to handle the difficult instances appropriately.

Therefore, we chose a somewhat generous aggregation criteria: if more than 25% of the responses (two out of seven people) indicated that a certain emotion applies, then that label was chosen. We will refer to this aggregation as *Ag2*. If no emotion got at least 40% of the responses (three out of seven people) and more than 50% of the responses indicated that the tweet was neutral, then the tweet was marked as neutral. In the vast majority of the cases, a tweet was labeled either as neutral or with one or more of the eleven emotion labels. 107 tweets did not receive sufficient votes to be labeled a particular emotion or to be labeled neutral. These very-low-agreement tweets were set aside. We will refer to the remaining dataset as E-c (Ag2), or simply *E-c*, data.

Since we used gold tweets interspersed with other tweets in our annotations, the amount of random or malicious annotations was small, identified, and discarded. Further, annotators had the option of choosing *neutral* if they did not see any emotion, and had no particular reason to choose an emotion at random. These factors allow us to use a 25% threshold for aggregation without compromising the quality of the data. Manual random spot-checks of the 25%–40% agreement labels by the authors revealed

	anger	antic.	disg.	fear	joy	love	optim.	pessi.	sadn.	surp.	trust	neutral
% votes	23.2	9.8	20.0	11.6	27.8	7.7	16.9	8.2	19.1	5.0	4.8	10.9
Ag2: % tweets labeled	36.1	13.9	36.6	16.8	39.3	12.3	31.3	11.6	29.4	5.2	5.0	2.7
Ag3: % tweets labeled	29.0	4.1	25.4	12.3	33.7	6.5	18.0	3.7	19.7	1.6	0.8	2.9

Table 2: **Applicable Emotion:** Percentage of votes for each emotion as being applicable (Q1+Q2) and the percentage of tweets that were labeled with a given emotion (after aggregation of votes).

	anger	antic.	disg.	fear	joy	love	optim.	pessi.	sadn.	surp.	trust	neutral
% votes	20.0	5.4	5.7	9.6	24.9	1.6	5.6	1.6	12.1	1.6	1.0	10.9
% tweets labeled	23.8	2.7	3.5	10.5	28.8	0.8	4.8	0.6	12.8	1.1	0.2	10.2

Table 3: **Primary Emotion:** Percentage of votes for each emotion as being the primary emotion (Q1) and the percentage of tweets that were labeled as having a given primary emotion (after aggregation of votes).

that the annotations are reasonable. Nonetheless, in certain applications, it is useful to train and test the systems on higher-agreement data. Thus, we are releasing a version of the E-c data with 40% as the cutoff (at least 3 out of 7 annotators must indicate that the emotion is present). We will refer to this aggregation as *Ag3*, and the corresponding dataset as *E-c (Ag3)*. 1,133 tweets did not receive sufficient votes to be labeled a particular emotion or to be labeled neutral when using Ag3. Note that all further analysis in this paper, except that pertaining to Table 2, is on the *E-c (Ag2)* data, which we will refer to simply as *E-c*.

Class Distribution: The first row of Table 2 shows the percentage of times each emotion was selected (in Q1 or Q2) in the annotations. The second and third rows show the percentage of tweets that were labeled with a given emotion using Ag2 and Ag3 for aggregation, respectively. The numbers in these rows sum up to more than 100% because a tweet may be labeled with more than one emotion. Observe that joy, anger, disgust, sadness, and optimism get a high number of the votes. Trust and surprise are two of the lowest voted emotions. Also note that with Ag3 the percentage of instances for many emotions drops below 5%.

The first row of Table 3 shows the percentage of times each emotion was selected as the primary emotion (in Q1). The second row shows the percentage of tweets that were labeled with having a given emotion as the primary emotion (after taking the majority vote). Observe that joy, anger, sadness, and fear are often the primary emotions. Even though optimism was often voted for as an emotion that applied (Table 2), Table 3 indicates that it is predominantly not the primary emotion.

2.2.2. Annotating Intensity with Best–Worst Scaling

We followed the procedure described by Kiritchenko and Mohammad (2016) to obtain BWS annotations. For each affect category, the annotators were presented with four tweets at a time (4-tuples) and asked to identify the tweeters that are likely to be experiencing the highest amount of the corresponding affect category (most angry, highest valence, etc.) and the tweeters that are likely to be experiencing the lowest amount of the corresponding affect category (least angry, lowest valence, etc.). $2 \times N$ (where N is the number of tweets in the emotion set) distinct 4-tuples were randomly generated in such a manner that each item was seen in eight different 4-tuples, and no pair of items occurred in more than one 4-tuple. We will refer to this procedure as *random maximum-diversity selection (RMDS)*. RMDS maximizes the number of unique items that each item co-occurs with in the 4-tuples. After BWS annotations, this in turn leads to direct comparative ranking information for the maximum number of pairs of items.

It is desirable for an item to occur in sets of 4-tuples such that the maximum intensities in those 4-tuples are spread across the range from low intensity to high intensity, as then the proportion of times an item is chosen as the best is indicative of its intensity score. Similarly, it is desirable for an item to occur in sets of 4-tuples such that the minimum intensities are spread from low to high intensity. However, since the intensities of items are not known beforehand, RMDS is used.

Every 4-tuple was annotated by four independent annotators.[8] The questionnaires were developed through internal discussions and pilot annotations. They are available on the SemEval-2018 AIT Task webpage.

Between 118 and 220 people residing in the United States annotated the 4-tuples for each of the four emotions and valence. In total, around 27K responses for each of the four emotions and around 50K responses for valence were obtained (see Table 4).[9]

Annotation Aggregation: The intensity scores were calculated from the BWS responses using a simple counting procedure (Orme, 2009; Flynn and Marley, 2014): For each item, the score is the proportion of times the item was chosen as having the most intensity minus the percentage of times the item was chosen as having the least intensity.[10] We linearly transformed the scores to lie in the 0 (lowest intensity) to 1 (highest intensity) range.

Distribution of Scores: Figure 1 shows the histogram of the V-reg tweets. The tweets are grouped into bins of scores 0–0.05, 0.05–0.1, and so on until 0.95–1. The colors for the bins correspond to their ordinal classes as determined from the manual annotation described in the next sub-section. The histograms for the four emotions are shown in Figure 3 in Appendix 6.1.

[8] Kiritchenko and Mohammad (2016) showed that using just three annotations per 4-tuple produces highly reliable results. Note that since each tweet is seen in eight different 4-tuples, we obtain $8 \times 4 = 32$ judgments over each tweet.

[9] Gold tweets were annotated more than four times.

[10] Code for generating tuples from items using RMDS, as well as for generating scores from BWS annotations: http://saifmohammad.com/WebPages/BestWorst.html

Dataset	Annotation Scheme	Location of Annotators	Annotation Item	#Items	#Annotators	MAI	#Q/Item	#Annotations
E-c$^{\text{Tw16,Tw17}}$	categorical	World	tweet	11,090	303	7	2	174,356
EI-reg$^{\text{Tw17}}$								
anger	BWS	USA	4-tuple of tweets	2,780	168	4	2	27,046
fear	BWS	USA	4-tuple of tweets	2,750	220	4	2	26,908
joy	BWS	USA	4-tuple of tweets	2,790	132	4	2	26,676
sadness	BWS	USA	4-tuple of tweets	2,744	118	4	2	26,260
V-reg$^{\text{Tw16,Tw17}}$	BWS	USA	4-tuple of tweets	5,134	175	4	2	49,856
Total								**331,102**

Table 4: Summary details of the current annotations done for the SemEval-2018 Affect in Tweets Dataset. These annotations were done on a set of 11,288 unique tweets. The superscript indicates the set of source tweets: Tw16 = Tweets-2016, Tw17 = Tweets-2017. MAI = Median Annotations per Item. Q = annotation questions. (This table does not include details for the EI-reg annotations done on the data from Tweets-2016 in earlier work (EI-reg$^{\text{Tw16}}$).)

2.2.3. Identifying Ordinal Classes

For each of the EI-reg emotions, the two authors of this paper independently examined the ordered list of tweets to identify suitable boundaries that partitioned the 0–1 range into four ordinal classes: *no emotion, low emotion, moderate emotion,* and *high emotion*. Similarly the V-reg tweets were examined and the 0–1 range was partitioned into seven classes: *very negative, moderately negative, slightly negative, neutral or mixed, slightly positive, moderately positive,* and *very positive mental state can be inferred.*[11]

Annotation Aggregation: The two authors discussed their individual annotations to obtain consensus on the class intervals. The V-oc and EI-oc datasets were thus labeled.

Class Distribution: The legend of Figure 1 shows the intervals of V-reg scores that make up the seven V-oc classes. The intervals of EI-reg scores that make up each of the four EI-oc classes are shown in Figure 3 in Appendix 6.1. Figure 2 shows the distribution of the tweets with hashtags indicating sarcasm or irony in the seven V-oc classes. Observe that a majority of these tweets are in the 'neutral or mixed' class. This aligns with the hypothesis that often sarcastic tweets indicate mixed emotions as on the one hand, the speaker may be unhappy about a negative event or outcome, but on the other hand, they choose to express themselves through humor. Figure 2 also shows that many of the sarcastic tweets convey a negative valence, and that sarcastic tweets conveying positive valence of the speaker are fewer in number.

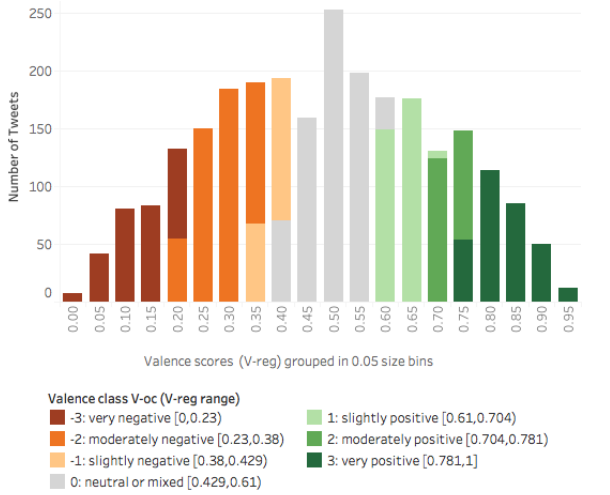


Figure 1: Valence score (V-reg) and class (V-oc) distribution.

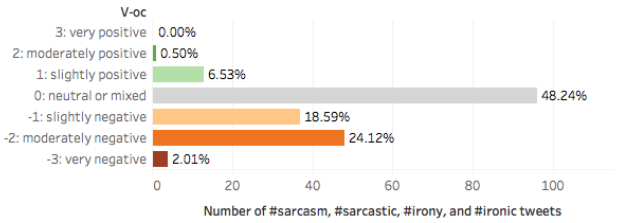


Figure 2: Valence class (V-oc) of tweets with sarcasm and irony indicating hashtags.

2.3. Training, Development, and Test Sets

Table 4 summarizes key details of the current set of annotations done for the *SemEval-2018 Affect in Tweets (AIT) Dataset*. AIT was partitioned into training, development, and test sets for machine learning experiments as described in Table 5. All of the tweets that came from Tweets-2016 were part of the training sets. All of the tweets that came from Tweets-2017 were split into development and test sets.[12]

Dataset	train$^{\text{Tw16}}$	dev$^{\text{Tw17}}$	test$^{\text{Tw17}}$	Total
E-c	6,838	886	3,259	10,983
EI-reg, EI-oc				
anger	1,701	388	1,002	3,091
fear	2,252	389	986	3,627
joy	1,616	290	1,105	3,011
sadness	1,533	397	975	2,905
V-reg, V-oc	1,181	449	937	2,567

Table 5: The number of tweets in the SemEval-2018 Affect in Tweets Dataset. The superscript indicates the set of source tweets: Tw16 = Tweets-2016, Tw17 = Tweets-2017.

[11] Valence is a bi-polar scale; hence, more classes.

[12] This split of Tweets-2017 was first done such that 20% of the tweets formed the dev. set and 80% formed the test set – independently for EI-reg, EI-oc, V-reg, V-oc, and E-c. Then we moved additional tweets from the test sets to the dev. sets such that a tweet in any dev. set would not occur in any test set.

Annotations	Inter-Rater Agreement	Fleiss' κ
Primary emotion (Q1)		
random	8.33	0.00
E-c	41.53	0.32
All applicable emotions (Q1+Q2)		
random	41.67	0.00
E-c: avg. for all 12 classes	83.38	0.21
E-c: avg. for 4 basic emotions	81.22	0.40

Table 6: Annotator agreement for the Multi-label Emotion Classification (E-c) Dataset.

3. Agreement and Reliability of Annotations

It is challenging to obtain consistent annotations for affect due to a number of reasons, including: the subtle ways in which people can express affect, fuzzy boundaries of affect categories, and differences in human experience that impact how they perceive emotion in text. In the subsections below we analyze the AIT dataset to determine the extent of agreement and the reliability of the annotations.

3.1. E-c Annotations

Table 6 shows the inter-rater agreement and Fleiss' κ for the multi-label emotion annotations. The inter-rater agreement is calculated as the percentage of times each pair of annotators agree. This measure does not take into account the fact that agreement can happen simply by chance. Fleiss' κ, on the other hand, calculates the extent to which the observed agreement exceeds the one that would be expected by chance (Fleiss, 1971). It is debatable if there is a need to correct for chance agreement, therefore we present both measures.[13] *E-c* shows the scores for the labeling of the primary emotion. The numbers for all applicable emotions are calculated by taking the average of the agreement/Fleiss' κ scores for each of the twelve labels individually. *E-c: 4 basic emotion classes* shows the averages for the four basic emotions, which are also the most frequent in the E-c dataset. The individual scores for each of the twelve classes are shown in Appendix 6.3. For the sake of comparison, we also show the score obtained by randomly choosing the predominant emotion, and the score obtained by randomly choosing whether a particular emotion applies or not.[14] Observe that the scores obtained through the actual annotations are markedly higher than the scores obtained by random guessing. Not surprisingly, the Fleiss' κ scores (chance-corrected agreement) are higher when asked to select only the primary emotion than when asked to identify all emotions that apply (since agreement on the more subtle emotion presence cases is expected to be low). The Fleiss' κ scores are also markedly higher on the frequently occurring four basic emotions, as compared to the full set.

3.2. EI-reg and V-reg Annotations

For real-valued score annotations, a commonly used measure of quality is reproducibility of the end result—if repeated independent manual annotations from multiple respondents result in similar intensity rankings (and scores), then one can be confident that the scores capture the true emotion intensities. To assess this reproducibility, we calculate average *split-half reliability (SHR)*, a commonly used approach to determine consistency (Kuder and Richardson, 1937; Cronbach, 1946; Mohammad and Bravo-Marquez, 2017b). The intuition behind SHR is as follows. All annotations for an item (in our case, tuples) are randomly split into two halves. Two sets of scores are produced independently from the two halves. Then the correlation between the two sets of scores is calculated. The process is repeated 100 times, and the correlations are averaged. If the annotations are of good quality, then the average correlation between the two halves will be high.

	Spearman	Pearson
Emotion Intensity		
anger	0.89	0.90
fear	0.84	0.85
joy	0.90	0.91
sadness	0.82	0.83
Valence	0.92	0.92

Table 7: Split-half reliabilities in the AIT Dataset.

Table 7 shows the split-half reliabilities for the AIT data. Observe that correlations lie between 0.82 and 0.92, indicating a high degree of reproducibility. Past work has found the SHR for sentiment intensity annotations for words, with 6 to 8 annotations per tuple to be 0.95 to 0.98 (Mohammad, 2018; Kiritchenko and Mohammad, 2016). In contrast, here SHR is calculated from whole sentences, which is a more complex annotation task and thus the SHR is expected to be lower than 0.95.

4. Associations Between Affect Dimensions

The AIT dataset allows us to study the relationships between various affect dimensions.

Co-occurrence of Emotions: Since we allow annotators to mark multiple emotions as being associated with a tweet in the E-c annotations, it is worth examining which emotions tend to frequently occur together. For every pair of emotions, i and j, we calculated the proportion of tweets labeled with both emotions *i* and *j* out of all the tweets annotated with emotion *i*.[15] (See Figure 5 in Appendix 6.5. for the co-occurrence numbers.) The following pairs of emotions have scores greater than 0.5 indicating that when the first emotion is present, there is a greater than 50% chance that the second is also present: anger–disgust, disgust–anger, love–joy, love–optimism, joy–optimism, optimism–joy, pessimism–sadness, trust–joy, and trust–optimism. In case of some pairs such as anger and disgust, presence of either one is strongly associated with the presence of the other, whereas in case of other pairs such as love and joy, the association is markedly stronger only in one direction. As expected, highly contrasting emotions such as love and disgust have very low co-occurrence scores.

[13] http://www.john-uebersax.com/stat/kappa2.htm http://www.agreestat.com/book3/bookexcerpts/chapter2.pdf

[14] See Appendix 6.4. for details.

[15] Note that the numbers are calculated from labels assigned after annotator votes are aggregated (Ag2).

V-reg–EI-reg	all data	the emotion present
valence–joy	0.79 (607)	0.65 (496)
valence–anger	-0.73 (598)	-0.40 (282)
valence–sadness	-0.73 (603)	-0.47 (313)
valence–fear	-0.60 (600)	-0.09 (175)

Table 8: Pearson correlation r between valence and each of the four emotions on the subset of the Tweets-2017 that is annotated for both valence and a given emotion. The numbers in brackets indicate the number of instances.

Correlation of Valence and Emotion Intensity: The real-valued scores for V-reg and EI-reg allow us to calculate the correlations between valence and the intensities of the annotated emotions. Table 8 shows the results. For every valence–emotion pair, only those instances are considered for which both valence and emotion intensity annotations are available. Observe that valence is found to be moderately correlated with joy intensities. The correlation is lower when we consider only those instances that have some amount of joy (EI-oc class is low, moderate, or high joy). Table 8 also shows that valence is inversely moderately correlated with anger, fear, and sadness. The correlation drops considerably for valence–fear, when examining only those data instances that have some amount of fear.

For any given tweet, we will refer to the ratio of one affect score to another affect score as *affect–affect intensity ratio*, or AAIR. If two affect dimensions are correlated (at least to some degree), then the AAIRs and the differences from the average help identify the tweets for which the two affect scores correlate and the tweets for which they do not.

For each affect dimension pair shown in Table 8, we calculate the AAIRs for the emotion-present tweets. Since valence (positiveness) is inversely correlated with each of the three negative emotions, for these emotions we calculate the AAIR with negativeness ($1 - valence$). We then examine those tweets for which the ratio is much greater than the average, as well as the tweets for which the ratio is much lower than the average. (Table 11 in Appendix 6.2. shows example tweets for both kinds.) For the valence–negative emotion tweet sets, the AAIR tends to be higher than the average AAIR when the tweet conveys a different negative emotion. For example, a tweet may have a high negativeness score (a low valence score) and a low fear score because it conveys a high amount of anger. Often the AAIR is lower than the average AAIR (low negativeness and high negative emotion), when the tweet expresses optimism, confidence, or resolve, despite a negative situation. Both of the above occur frequently in the valence–fear-present set of tweets, resulting in the particularly low correlation scores. Examination of the valence–joy tweets reveals that the AAIRs are higher than the average AAIR (i.e., high valence and low joy) when tweets convey positive emotions other than joy such as optimism, satisfaction, and relief. (See examples in Table 11.)

Correlations of the Intensities of Pairs of Negative Emotions: As mentioned earlier, we chose to annotate a common set of 600 tweets for intensity of anger, fear, and sadness. We can thus calculate the extent to which these scores are correlated. Table 9 shows the results. Observe that the scores are in the range from 0.5 to 0.65 for the full set; however, the scores are much closer to 0, when considering only those tweets where both emotions are present (have EI-oc labels of low, moderate, or high emotion). This suggests that when the emotions are present, the intensities are largely not correlated with each other. Table 11 in Appendix 6.2. shows example tweets whose AAIRs were markedly higher or lower than the average—tweets whose scores were high for one emotion, but low for the other emotion. Table 9 results also imply that when a particular emotion is not present, then the intensities correlated moderately. This is possibly because in the absence of the emotion, the BWS annotators ranked tweets as per valence. For example, a person who tweeted a happy thought will likely be marked least angry more often than the person who tweeted a neutral thought.

EI-reg–EI-reg	all data	both emotions present
fear–sadness	0.64 (668)	0.09 (174)
anger–sadness	0.62 (616)	0.08 (224)
anger–fear	0.51 (599)	-0.13 (124)

Table 9: Pearson correlation r between each pair of the negative emotions on the subset of the Tweets-17 that is annotated for both emotions. The numbers in brackets indicate the number of instances in each case.

5. Summary and Future Work

We created a new affectual tweets dataset of more than 11,000 tweets such that overlapping subsets are annotated for a number of emotion dimensions (from both the basic emotion model and the VAD model). For each emotion dimension, we annotated the data not just for coarse classes (such as anger or no anger) but also for fine-grained real-valued scores indicating the intensity of emotion (anger, sadness, valence, etc.). We crowdsourced the data annotation through a number of carefully designed questionnaires. We used Best–Worst Scaling to obtain fine-grained real-valued intensity scores (split-half reliability scores > 0.8)).

The new dataset is useful for training and testing supervised machine learning algorithms for a number of emotion detection tasks. We made the dataset freely available via the website for *SemEval-2018 Task 1: Affect in Tweets* (Mohammad et al., 2018).[16] Subsequently, Spanish and Arabic tweet datasets were also created following the methodology described here (Mohammad et al., 2018). The SemEval task received submissions from 72 teams for five different tasks, each with datasets in English, Arabic, and Spanish.

The Affect in Tweets Dataset is also useful for shedding light on research questions about the relationships between affect categories. We calculated the extent to which pairs of emotions co-occur in tweets. We showed the extent to which the intensities of affect dimensions correlate. We also calculated affect–affect intensity ratios which help identify the tweets for which the two affect scores correlate and the tweets for which they do not.

We are currently annotating the dataset for arousal and dominance. With those additional annotations, we can explore how valence, arousal, and dominance change across tweets with low to high anger/joy/sadness/fear intensity.

[16] http://www.saifmohammad.com/WebPages/affectintweets.htm

6. Bibliographical References

Alm, C. O., Roth, D., and Sproat, R. (2005). Emotions from text: Machine learning for text-based emotion prediction. In *Proceedings of the Joint Conference on HLT–EMNLP*, Vancouver, Canada.

Cronbach, L. (1946). A case study of the splithalf reliability coefficient. *Journal of educational psychology*, 37(8):473.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3):169–200.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Flynn, T. N. and Marley, A. A. J. (2014). Best-worst scaling: theory and methods. In Stephane Hess et al., editors, *Handbook of Choice Modelling*, pages 178–201. Edward Elgar Publishing.

Frijda, N. H. (1988). The laws of emotion. *American psychologist*, 43(5):349.

Kiritchenko, S. and Mohammad, S. M. (2016). Capturing reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, San Diego, California.

Kiritchenko, S. and Mohammad, S. M. (2017). Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of The Annual Meeting of the Association for Computational Linguistics (ACL)*, Vancouver, Canada.

Kuder, G. F. and Richardson, M. W. (1937). The theory of the estimation of test reliability. *Psychometrika*, 2(3):151–160.

Louviere, J. J., Flynn, T. N., and Marley, A. A. J. (2015). *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge University Press.

Louviere, J. J. (1991). Best-worst scaling: A model for the largest difference judgments. Working Paper.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Mohammad, S. M. and Bravo-Marquez, F. (2017a). Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada.

Mohammad, S. M. and Bravo-Marquez, F. (2017b). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mohammad, S. M., Sobhani, P., and Kiritchenko, S. (2017). Stance and sentiment in tweets. *Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media*, 17(3).

Mohammad, S. M., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Mohammad, S. M. (2018). Word affect intensities. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

Nakov, P., Rosenthal, S., Kiritchenko, S., Mohammad, S. M., Kozareva, Z., Ritter, A., Stoyanov, V., and Zhu, X. (2016). Developing a successful semeval task in sentiment analysis of twitter and other social media texts. *Language Resources and Evaluation*, 50(1):35–65.

Orme, B. (2009). Maxdiff analysis: Simple counting, individual-level logit, and HB. Sawtooth Software, Inc.

Parrot, W. (2001). *Emotions in Social Psychology*. Psychology Press.

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. *Emotion: Theory, research, and experience*, 1(3):3–33.

Russell, J. A. (2003). Core affect and the psychological construction of emotion. *Psychological review*, 110(1):145.

Strapparava, C. and Mihalcea, R. (2007). Semeval-2007 task 14: Affective text. In *Proceedings of SemEval-2007*, pages 70–74, Prague, Czech Republic.

Yu, L.-C., Lee, L.-H., Hao, S., Wang, J., He, Y., Hu, J., Lai, K. R., and Zhang, X.-j.). Building Chinese affective resources in valence-arousal dimensions. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, San Diego, California.

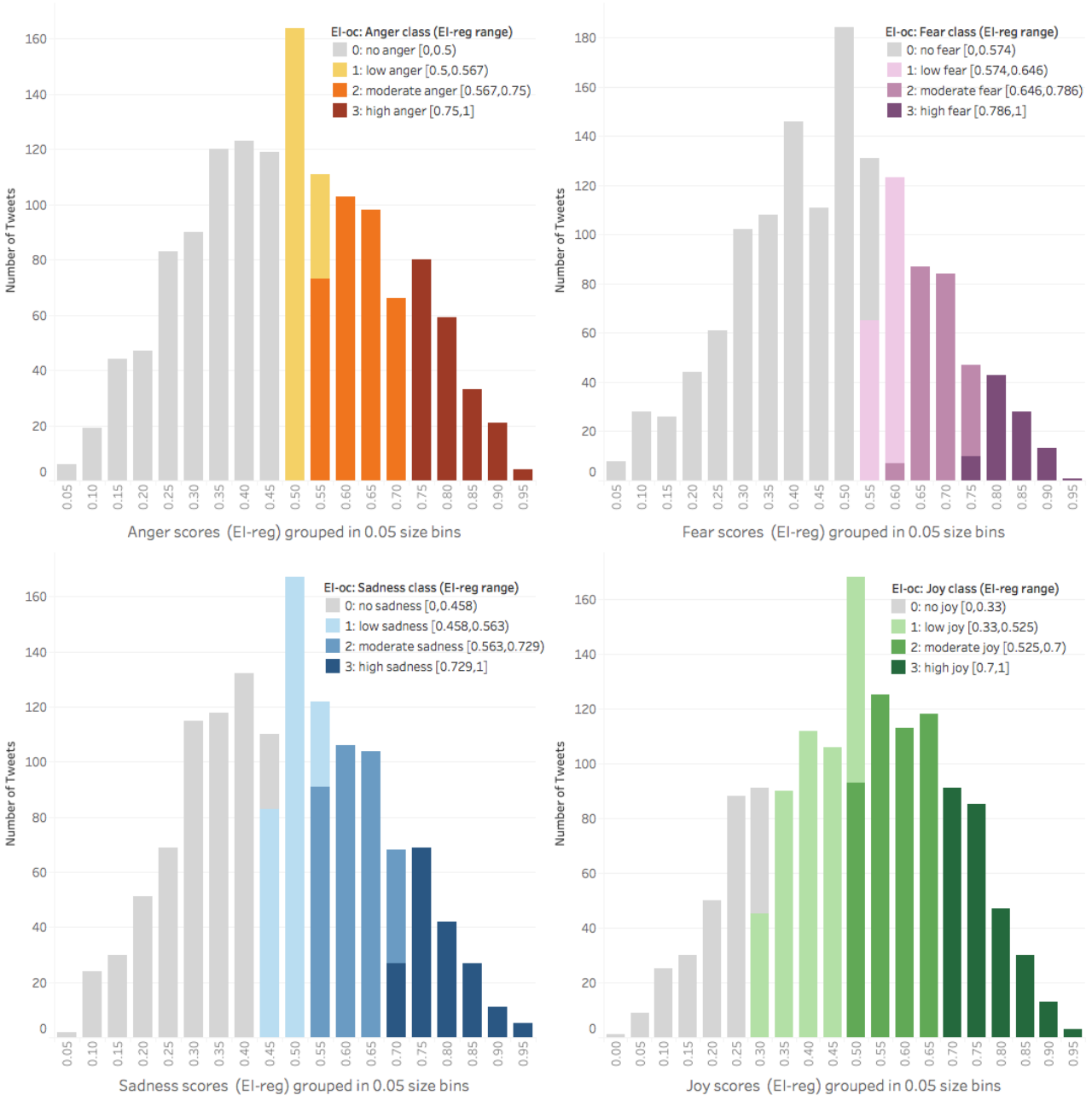


Figure 3: Emotion intensity score (EI-reg) and ordinal class (EI-oc) distributions for the four basic emotions in the SemEval-2018 AIT development and test sets combined. The distribution is similar for the training set, which was annotated in earlier work.

Appendix

6.1. Distributions of the EI-reg Tweets

Figure 3 shows the histograms of the EI-reg tweets in the anger, joy, sadness, and fear datasets. The tweets are grouped into bins of scores 0–0.05, 0.05–0.1, and so on until 0.95–1. The colors for the bins correspond to their ordinal classes: no emotion, low emotion, moderate emotion, and high emotion. The ordinal classes were determined from the EI-oc manual annotations.

6.2. Relationships Between Affect Dimension Pairs

Figure 4 shows the valence of tweets in the EI-reg and EI-oc datasets. Observe that, as desired, using the chosen query terms led to the joy dataset consisting of a majority positive tweets and the anger, fear, and sadness datasets consisting of a majority negative tweets. Table 11 shows pairs of example tweets whose AAIRs are markedly higher and lower from the average AAIR, respectively. Such tweets shed light on why the two affect dimensions are not perfectly correlated (or perfectly inversely correlated).



Figure 4: Valence of tweets in the EI-reg and EI-oc datasets.

6.3. Per-Emotion Annotator Agreement in the E-c Annotations

Table 10 shows the per-emotion annotator agreement for the Multi-label Emotion Classification (E-c) Dataset. Observe that the Fleiss' κ scores are markedly higher for the frequently occurring four basic emotions (joy, sadness, fear, and anger), and lower for the less frequent emotions. (Frequencies for the emotions are shown in Table 2.) Also note, that the agreement is low for the neutral class. This is not surprising because the boundary between neutral (or no emotion) and slight emotion is fuzzy. This means that often at least one or two annotators indicate that the person is feeling some joy or some sadness, even if most others indicate that the person is not feeling any emotion.

6.4. E-c: Random Guess Agreement Calculation

When randomly guessing whether an emotion applies or not, half of the annotators ($\frac{n}{2}$) are expected to choose the emotion, and the other half are expected not to choose the emotion. So, there are $\frac{n}{4}(\frac{n}{2}-1)$ pairs of the annotators that agree that the emotion is present, and the same number of pairs that agree that the emotion does not apply. All the other pairs disagree. There are $\frac{n}{2}(n-1)$ total number of the annotator pairs. So, the Inter-Rater Agreement, which is the percentage of the annotator pairs that agree, is $\frac{n-2}{2(n-1)}$. For $n = 7$, IRA is 41.67%.

Emotions	Inter-Rater Agreement	Fleiss' κ
anger	79.23	0.41
anticipation	83.05	0.04
disgust	74.68	0.20
fear	87.45	0.38
joy	78.91	0.47
love	88.83	0.21
optimism	77.15	0.18
pessimism	86.27	0.08
sadness	79.27	0.32
surprise	91.20	0.07
trust	91.17	0.04
neutral	83.33	0.14

Table 10: Applicable emotions (Q1+Q2): Per-emotion annotator agreement for the annotations in the E-c data.

6.5. Emotion–Emotion Co-occurrence

Figure 5 shows the proportion of tweets in the E-c dataset annotated with each pair of emotions. For a pair of emotions, say from row i and column j, the number in cell *(i,j)* shows the proportion of tweets labeled with both emotions i and j out of all the tweets annotated with emotion i.[17])

[17]Note that the numbers are calculated from labels assigned after annotator votes are aggregated (Ag2).

Emotion	anger	anticipat..	disgust	fear	joy	love	optimism	pessimis..	sadness	surprise	trust
anger	1.00	0.07	0.81	0.15	0.09	0.01	0.07	0.11	0.35	0.04	0.01
anticipation	0.18	1.00	0.19	0.16	0.47	0.08	0.45	0.07	0.16	0.12	0.11
disgust	0.80	0.07	1.00	0.17	0.08	0.01	0.05	0.14	0.40	0.04	0.01
fear	0.31	0.13	0.37	1.00	0.13	0.02	0.13	0.19	0.35	0.04	0.03
joy	0.08	0.17	0.08	0.06	1.00	0.29	0.62	0.02	0.10	0.06	0.09
love	0.03	0.09	0.02	0.02	0.94	1.00	0.66	0.01	0.07	0.05	0.11
optimism	0.08	0.20	0.06	0.07	0.78	0.26	1.00	0.03	0.09	0.05	0.13
pessimism	0.34	0.09	0.45	0.27	0.08	0.01	0.07	1.00	0.71	0.03	0.01
sadness	0.43	0.08	0.49	0.20	0.13	0.03	0.10	0.28	1.00	0.03	0.01
surprise	0.25	0.31	0.25	0.14	0.49	0.11	0.29	0.06	0.20	1.00	0.07
trust	0.06	0.31	0.04	0.09	0.69	0.26	0.78	0.02	0.07	0.07	1.00

Figure 5: The proportion of tweets in the E-c dataset annotated with each pair of emotions.

AD1–AD2	Intensity of AD1	Intensity of AD2	Example Tweet
Valence – Emotion			
valence–anger	V=↓	A=↓	*Man I feel like crap today*
	V=↑	A=↑	*Up early. Kicking ass and taking names. #offense.*
valence–fear	V=↓	F=↓	*@altontowers Loves the rich!!! Fuck us 'working class folk' and our kids! #fuming #joke #scandalous #disgusting*
	V=↑	F=↑	*Heading to PNC to get the ball going for my MA! #goingforit #nervous #excited*
valence–joy	V=↓	J=↑	*One of the greathorrible moments as a professor is seeing a wonderful student leaving your university to pursue his/her true passion.*
	V=↑	J=↓	*Nothing is more #beautiful than a #smile that has struggled through tears. #optimism [muscles emoji]*
valence–sadness	V=↓	S=↓	*It's 2017 and there still isn't an app to stop you from drunk texting #rage*
	V=↑	S=↑	*@[masked] keep your head clear an focused. Do not let T intimidate you or use your children to silence you! Hate when a man does that!*
Negative Emotion – Negative Emotion			
anger–fear	A=↓	F=↑	*Going to sleep was a bad idea i had a horrible nightmare abt what i hate the most in a nightmare but its fine im ok*
	A=↑	F=↓	*Don't fucking tag me in pictures as 'family first' when you cut me out 5 years ago. You're no one to me.*
fear–sadness	F=↓	S=↑	*This kind of abuse is UNBELIEVABLE and an absolute disgrace. It makes me sad to see this #dismayed*
	F=↑	S=↓	*I am having anxiety right now because I don't know it's gonna happen*
sadness–anger	S=↓	A=↑	*I hate when stupid ass shit irritate me*
	S=↑	A=↓	*Found out the peraon i liked wanted to that someone else #sadness*

Table 11: Pairs of example tweets whose AAIRs are markedly higher and lower from the average AAIR, respectively, for various affect dimension (AD) pairs. Such tweets shed light on why the two affect dimensions are not perfectly correlated (or perfectly inversely correlated)

When ACE met KBP[1]: End-to-End Evaluation of Knowledge Base Population with Component-level Annotation

Bonan Min[†], Marjorie Freedman[‡], Roger Bock[†], Ralph Weischedel[‡]
[†] Raytheon BBN Technologies, Cambridge, MA 02138
{bonan.min, roger.bock}@raytheon.com
[‡] USC/Information Sciences Institute, Marina del Rey, CA 90292
{mrf, weisched}@isi.edu

Abstract

Building a Knowledge Base from text corpora is useful for many applications such as question answering and web search. Since 2012, the Cold Start Knowledge Base Population (KBP) evaluation at the Text Analysis Conference (TAC) has attracted many participants. Despite the popularity, the Cold Start KBP evaluation has several problems including but not limited to the following two: first, each year's assessment dataset is a pooled set of query-answer pairs, primarily generated by participating systems. It is well known to participants that there is pooling bias: a system developed outside of the official evaluation period is not rewarded for finding novel answers, but rather is penalized for doing so. Second, the assessment dataset, constructed with lots of human effort, offers little help in training information extraction algorithms which are crucial ingredients for the end-to-end KBP task. To address these problems, we propose a new unbiased evaluation methodology that uses existing component-level annotation such as the Automatic Content Extraction (ACE) dataset, to evaluate Cold Start KBP. We also propose bootstrap resampling to provide statistical significance to the results reported. We will then present experimental results and analysis.

Keywords: Information Extraction, Knowledge Base Population, evaluation

1. Introduction

Automatically constructing a Knowledge Base (KB) of entities and relations from unstructured text, has long been a goal of Natural Language Processing (NLP). The task, named Knowledge Based Population (KBP), will unlock the huge potential in unstructured text for applications such as questions answering and web search.

Since 2012, the National Institute of Standards and Technology (NIST) has run the TAC Cold Start KBP evaluation, which measures performance of KBP. As the successor to the Message Understanding Conference (MUC) (Grishman and Sundheim, 1996) and Automatic Content Extraction (ACE) (Doddington et al., 2004) evaluations, Cold Start KBP evaluates a system's ability to automatically construct a KB from text. It uses a large corpus of 50,000-90,000 documents which have not gone through a careful selection process. In Cold Start KBP evaluation, a system is required to submit a KB [2] of entities and relations, constructed automatically from the corpus by algorithms.

How can one evaluate the quality of a KB? The Cold Start KBP evaluation[2] measures it by probing the KB with two types of queries: 1-hop (e.g., *which organization(s) is(are) founded by Bill Gates?)* or 2-hop (e.g., *in which city(-ies) is(are) the organization(s) founded by Bill Gates headquartered?*). The evaluation software traverses a KB and finds all answers to the 1-hop and 2-hop queries. Human annotators then annotate the correctness of a system answer by checking whether it is sufficiently justified in the source corpus. The process is performed over all submitted KBs[3].

While the Cold Start KBP evaluation directly measures end-to-end performance on KBP, it has several problems:

- The scores will vary by number of participants and the amount of answers they produced. Furthermore, the scores aren't comparable from year to year, therefore it is hard to measure progress.
- Given the high cost of the assessment process, the query set has typically been small relative to the schema size. For example, the 2016 query set contains only 317 queries - not a large number for 42 relation types.
- The evaluation suffers from severe pooling bias. Chaganty et al. (2017) show that the Cold Start KBP evaluation is significantly and systematically biased against systems that make novel predictions. For a system that does not participate in each year's evaluation, the pool is likely to not contain a significantly large fraction of correct answers. Therefore, recall will be significantly underestimated. Precision will also be estimated incorrectly because of novel answers that are not assessed.
- The assessment dataset is at the end-to-end (query-answer pair) level. It offers little for improving the components of a KBP system. A standard approach (Ji and Grishman, 2011) to KBP is to integrate a range of Information Extraction (IE) technologies including: named entity recognition, within document coreference, relation extraction, and cross document coreference. The KBP dataset cannot be used for (re)training any of the component level algorithms.
- The KBP assessment dataset annotation lacks component level annotation to support error analysis. A KBP system developer must trace the cause of an error. On the Slot Filling subtask alone, Min et al. (2012)

[1] Inspired by the movie "*When Harry Met Sally…"* in which two friends with drastically different personalities found each other to be the love of their life.

[2] The schema of the KB and the evaluation procedure, are defined in the task description, available at https://tac.nist.gov/2017/KBP/ColdStart/guidelines/TAC_KBP_2017_ColdStartTaskDescription_1.0.pdf

[3] A time-limited manual run is conducted and used to increase the size of the answer pool.

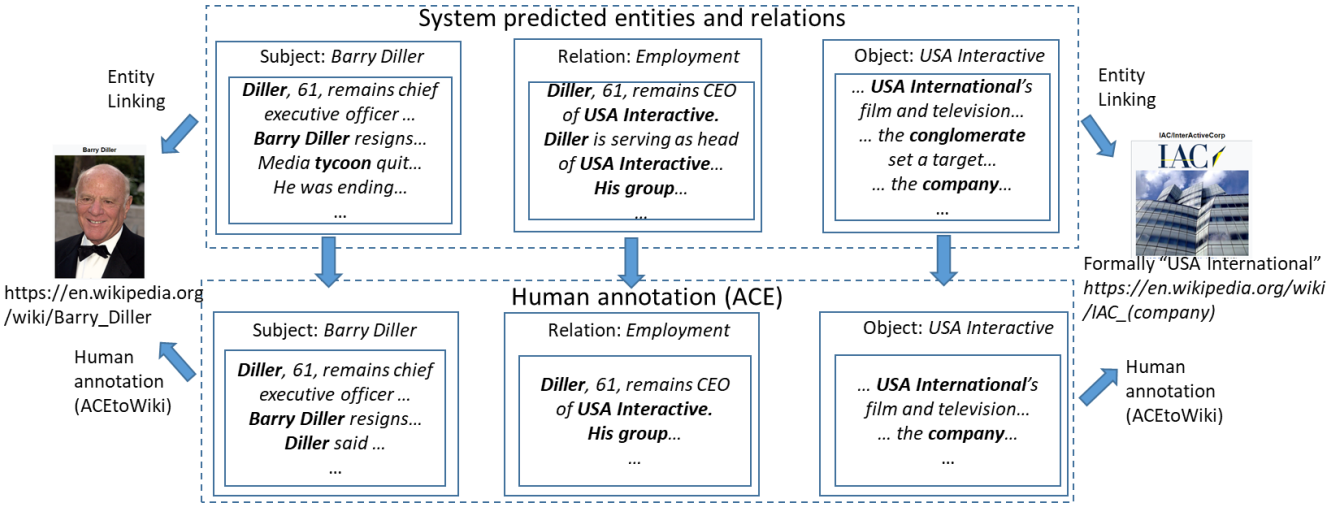


Figure 1 Aligning a system-predict relation to ground truth. The arrows show the direction of alignment.

shows that error analysis requires significant manual effort even with the help of KBP assessments.

- While it may seem that participants could accurately estimate the quality over the overall KB by measuring performance on traditional information extraction tasks (e.g. with standard relation extraction, coreference, and name-entity recognition datasets and metrics), experience has shown that improvements in an enabling technology do not translate to improvements in overall knowledge base quality. We hypothesize this is due to the differences in focus between sentence-by-sentence information extraction and a task that examines a corpus as whole. As an example, KB quality is impacted less by finding additional instances of the same very common fact, where as in sentence-by-sentence extraction finding an additional instance of a fact that has been seen in the previous sentence is weighted equivalently to finding a new fact.

To address these problems, we propose a novel evaluation method that uses existing information extraction resources such as the ACE training corpus (Walker et al, 2016) to evaluate Cold Start KBP. It has no pooling bias, does not rely on carefully selected queries, and can be used to measure progress since no additional annotation is required for a new system. The dataset has component-level annotation, therefore it supports fine-grained analysis of errors and can be used for training/improving IE components. Furthermore, we augment the method with bootstrap resampling to provide statistical significance. We present experiments and analysis.

2. Related Work

The TAC Cold Start KBP evaluation provides a corpus of 50,000 to 90,000 documents. A system is expected to produce a KB of 5 entity types and 42 relation types defined in the TAC schema[2]. The evaluation software probes each submission KB with 1-hop or 2-hop queries and obtains a set of answers. A separate time-limited human answer-finding round is also conducted to add more answers. The pooled answer set, accompanied with justification text in the original corpus, are provided to human annotators to be assessed as correct, incorrect, or redundant. A system is measured by the precision and recall of its answers to queries, using the pool of assessed answers. The evaluation is subject to pooling bias. Lacking identification of which component the error stems from, the annotation is not very useful for analyzing the error nor can it be used to improve or (re)train new component-level models for system improvement.

Recently, Chaganty et al. (2017) proposed an on-demand evaluation framework for Cold Start KBP. Observing that newly developed systems suffer from significant pooling bias, they proposed to use crowdsourcing to annotate newly found answers on-demand, and an importance sampling strategy for unbiased evaluation. Although the collected annotation contains EDL (entity discovery and linking) and Slot filling annotation, it still is not sufficiently useful for fine-grained error analysis such as mention tagging, coreference, etc. Moreover, the annotation is collected at the end-to-end level; therefore it is not straightforward to use it to train component-level algorithms. Furthermore, the cost for each new system is about $300. The total cost could potentially be very large if many systems need to be evaluated on-demand, e.g. to support variations in parameters and/or algorithms.

3. A Novel Evaluation Method

Resources We use the ACE 2005 English training corpus (Walker et al., 2006) as the evaluation corpus. The ACE corpus consists of articles from weblogs, broadcast news, newsgroups, and broadcast conversations; it is annotated exhaustively with mentions, coreference, and relations. To augment the ACE corpus with entity linking annotation, we use ACEtoWiki[4] (Bentivogli et al., 2010), an auxiliary dataset which extends the English ACE 2005 corpus annotation with ground-truth links to Wikipedia.

The ACE dataset is widely used to evaluate IE components such as named entity recognition, within-document coreference, and relation extraction. Since there is a plethora of work (Nadeau and Sekine, 2007; Luo, 2005) on component-level evaluations, we will refer interested readers to these papers. As a benchmark dataset for IE components, ACE helps KBP system developers to find places to improve and understand

[4] We manually remove links to under-specified pages or links to groups of entities, e.g., links to *People*, *Presidents of the United States*, *Country*, *Politician*, *etc*. This resulted in removal of 14% links.

where the errors are. The ACE dataset has also been used extensively for training entity (Nadeau and Sekine, 2007) and relation tagging models (Zhou et al., 2005).

We will focus on the end-to-end evaluation and describe how to use the ACE and ACEtoWiki augmentation to evaluate KBP. At a high level, the idea is to first align system predicted relation triples *{<subject, relation, object>}* to the ground truth triples {*<subject', relation', object'>}*, and then generate relation paths of a single or multiple hops with each hop being a relation triple. The resulting ground truth relation paths, can be used as reference for measuring how accurate a system is (precision) and its coverage (recall). Figure 1 shows an example of the process to align system predicted relation triples to the ground truth. We need to perform alignment at the following levels:

Document-level entity: An alignment is found if *>P%* (*P=50* in the experiments) named mentions in a system-predicted entity cluster can be aligned to one of the mentions in a corresponding ground truth entity. For example, a system-generated *Barry Diller* entity will be aligned to a ground truth entity *e* if more than 50% of the system-tagged mentions are in *e*. As illustrated in Figure 1, we perform entity alignment for the relation's subject *Barry Diller*, and object *USA Interactive*.

Entity Linking: For each entity, we use ACEtoWiki dataset to find a Wikipedia page for each named mention (if the page exists). The Wikipedia page of the most named mentions will be used as the ground truth page for the entity. For the names that are unlinkable to any Wikipedia pages, we cluster them by exact string match. This generates a unique corpus-level entity for one or multiple coreferential document-level entities.

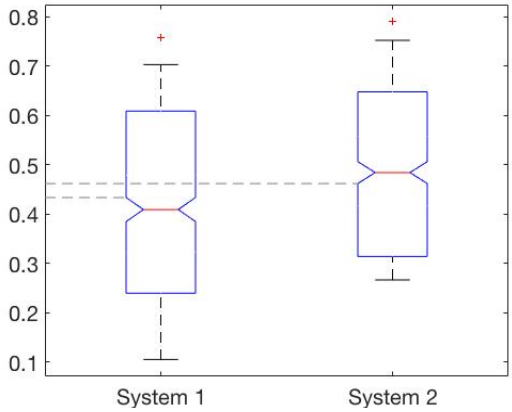


Figure 2 Illustration of scores intervals and comparing of system performance. The box shows 95% (top) and 5% (bottom). Outliers (red dots) are above the top horizontal line or below the bottom horizontal line. Medians are represented with red lines.

Relation: A system-predicted relation can be aligned to a ground truth relation *r*, if its subject, relation type and object all can be aligned to the corresponding fields of *r*. We found a relation type match if the system-predicted type is exactly the same type in the ground truth. We didn't enforce relation provenances being equal since we focus on corpus-level end-to-end KBP.

Relational paths: To support evaluation of 1-hop, 2-hop, and multi-hop queries (e.g., answering a 2-hop question *in which city(-ies) the organization(s) founded by Bill Gates is(are) headquartered?*), we generate *n*-hop relational paths in the form of $< e_1, r_1, r_2, \dots, r_n, e_{n+1} >$. e_1, e_{n+1} are head and tail entities. $r_1, r_2 \dots$ are relation types. We define one-hop paths (relation triples) as R_1:

$$R = R_1 = \{< e_1, r_1, e_2 >\}$$

Further, we define two-hop paths R_2 as the following:

$$R_2 = R_1 \hat{\times} R = \{< e_1, r_1, r_2, e_3 > | \forall < e_1, r_1, e_2 >, < e_2, r_2, e_3 > \in R\}$$

and n-hop paths R_n:

$$R_n = R_{n-1} \hat{\times} R = \{< e_1, r_1, r_2, \dots, r_n, e_{n+1} > | \forall < e_1, r_1, e_2, r_2, e_3, \dots, r_{n-1}, e_n > \in R_{n-1}, \forall < e_n, r_n, e_{n+1} > \in R\}$$

We generate two sets of multi-hop paths with the above mentioned equation: R_n from ground truth annotation, and R'_n from the system predicted tuples. We compare R'_n to R_n, and calcualte preicsion, recall and F1:

$$P = \frac{|R_n \cap R'_n|}{R'_n}, R = \frac{|R_n \cap R'_n|}{R_n}, F1 = \frac{2PR}{P+R}$$

Bootstrap re-sampling: The ACE English dataset contains 596 documents. Similar datasets (e.g., Rich ERE (Song et al., 2015)) with entity and relation annotation contains a similar number of documents. To show the statistical variance of the scores as well as to show whether any increase in scores is statistically significant, we propose bootstrap resampling (Efron and Tibshirani, 1994), which has been applied in tasks such as measuring Machine Translation performance (Koehn, 2004).

Bootstrap resampling works as follows: Assume we can only measure performance with *n* (*n=596* for ACE) documents, randomly drawn from some large corpus in the ideal world (on which it would be too expensive to annotate exhaustively). We compute precision, recall and F1 with the above-mentioned method on the *n* documents. We could sample another test set of *n* documents from the original *n* documents with replacement, and compute the scores again. We repeat this for a sufficiently large number of times (e.g., 1000 times) and produce many measures of the performance. These sampled scores can be used for

- Estimating an interval [*a*, *b*] which approaches the 90% confidence interval for scores of test set of size *n*. To do so, we will sort the scores and then drop the top 5% and bottom 5%. We show the interval as a box as illustrated for *System 1* in Figure 2. Outliers are above the top horizontal line or below the bottom horizontal line.
- Estimating whether an improvement in score is statistically significant. To do so, we construct a pair of bootstrap resampling scores, one for a baseline system *System 1*, and the other for an improved system *System 2*. Both are illustrated in Figure 2. The notch of a box shows the confidence interval which is normally based on the median$\pm 1.15 * IQR/\sqrt{n}$ (the interquartile range (IQR) is the 25 to 75 percentage). Notches (Chambers et al., 1983) are useful in offering a rough guide to significance of difference of medians; if the notches of two boxes do not overlap, this offers evidence of a statistically significant difference between (95% confidence) the medians. The two notches in Figure 2

don't overlap, it shows the improvements (as defined by the improvement in median) is statistically significant.

4. Implementation and Experiments

As described in Section 3, we use the ACE English dataset as the evaluation dataset. We use relation subtypes since subtypes represent concrete relations such as *Employement*, *Subsidiary* instead of their categorical counterparts (types defined in ACE) such as *Organization-Affiliation* and *Part-Whole*. We do not include *User-Owner-Inventor-Manufacturer*, *Citizen-Resident-Religion-Ethnicity*, and *Lasting-Personal* since the meaning of these relations are not clear - each can be further divided into finer-grained types.

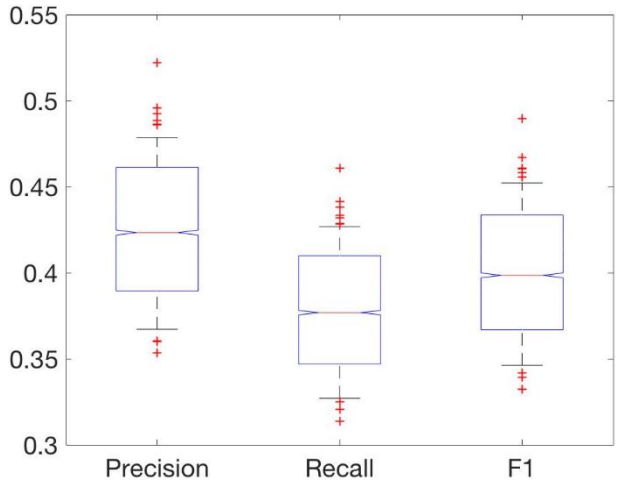
Table 1 System performance on the ACE corpus.

	Precision	Recall	F1
1-hop	0.427	0.38	0.402
2-hop	0.24	0.156	0.188
3-hop	0.103	0.09	0.095

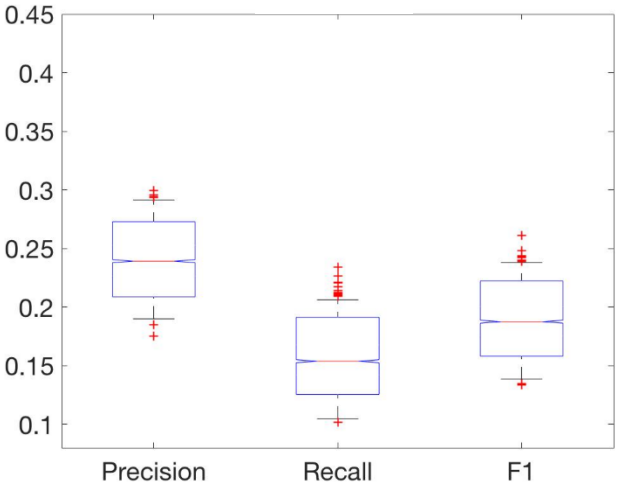
We apply a state-of-the-art Cold Start KBP system (Min et al., 2017; Min and Freedman, 2016) on the ACE English document sets. Since TAC KBP has different type sets for relations, we only apply the ACE relation extraction system in Min and Freeman (2016)[5] to support the ACE relation schema.

Table 1 shows the performance on the set of 596 ACE documents without any sampling approach. We measure performance up to three hops (e.g., "*Where is the organization Bill Gates's mother works for?*"). Both precision and recall drop as we add hops. This shows the known problem of error multiplication in Cold Start KBP: Errors accumulate along the paths and renders the end results less precise and have lower coverage.

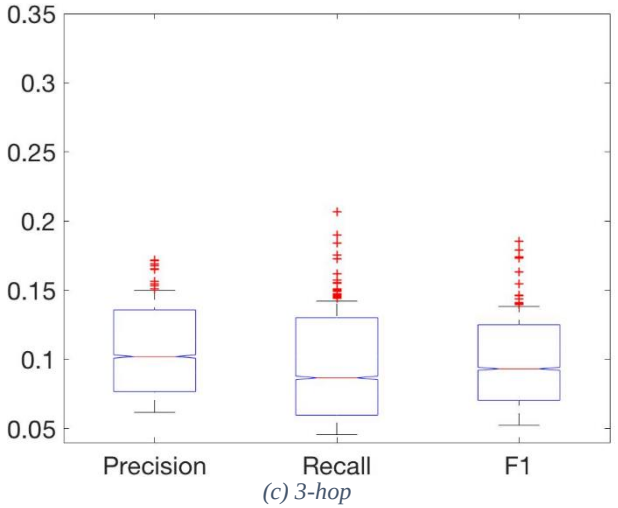
We further experimented with bootstrap resampling. We ran the experiment 1000 times, sampling documents with replacement. Figure 3 shows the results on 1-hop (Figure 3a), 2-hop (Figure 3b), and 3-hop (Figure 3c) respectively. Similar to Table 1, precision, recall and F1 also decrease as hop increases. The variance (length of boxes as well as bars indicating outliers) is not very large. This indicates that we could obtain very accurate estimates for each measure with a large sampling size (1000 for our experiments). In addition, the notches are very small (<1 point). This offers an accurate way to measure statistically significant improvements to the system – for scores obtained with bootstrap resampling for a new system, if the notch of the new scores didnot overlap with the current notch, it indicates a statically significant difference. The small notches on all experiments show that we could measure statistically significant improvements fairly accurately.



(a) 1-hop



(b) 2-hop



(c) 3-hop

Figure 3 System performance with bootstrap resampling on 1-hop (top), 2-hop (middle), and 3-hop (bottom). The scores are generated with 1000 bootstrap resampling runs. The top and bottom lines of the boxes show 95% and 5% percentile respectively. The notch (though very small due to the low variance in median scores) shows confidence intervals of median scores.

5. Conclusion and Future Work

We present a novel method for evaluating end-to-end Knowledge Base Population with component-level annotation. Our method makes use of existing component-

[5] One of the main relation extraction components in Min and Freedman (2016) is a set of statistical models trained with the ACE training dataset. Its decoding output can be mapped into KBP relation types, given the similarity between ACE and KBP. We use the unmodified ACE type output to support direct assessment on the ACE dataset.

level annotation such as ACE. It also includes bootstrap resampling approaches for measuring statistical significance of the results. Our next step is to apply the approach to other datasets such as the rich ERE (Entity, Relation and Events) (Song et al., 2015) annotation dataset.

6. Acknowledgements

This work was supported by DARPA/I2O Contract No. FA8750-13-C-0008 under the DEFT program. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

7. Bibliographical References

Luisa Bentivogli, Pamela Forner, Claudio Giuliano, Alessandro Marchetti, Emanuele Pianta, Kateryna Tymoshenko. Extending English ACE 2005 Corpus Annotation with Ground-truth Links to Wikipedia. In Proceedings of COLING 2010 Workshop on "The People's Web Meets NLP: Collaboratively Constructed Semantic Resources", Beijing, China, August 28, 2010.

Arun Chaganty, Ashwin Paranjape, Percy Liang and Christopher D. Manning. Importance sampling for unbiased on-demand evaluation of knowledge base population. In Proceedings of EMNLP 2017.

John M. Chambers, William S. Cleveland, Beat Kleiner, and Paul A. Tukey. 1983. "Comparing Data Distributions." In Graphical Methods for Data Analysis, 62. Belmont, California: Wadsworth International Group.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, Ralph Weischedel. The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation. In Proceedings of LREC 2004.

Bradley Efron and R. J. Tibshirani. 1994. An Introduction to the Bootstrap. CRC Press.

Ralph Grishman and Beth Sundheim, Message Understanding Conference - 6: A Brief History. In Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, Denmark, 1996.

Paul McNamee, Hoa T. Dang, Heather Simpson, Patrick Schone, Stephanie Strassel. An Evaluation of Technologies for Knowledge Base Population. In Proceedings of LREC 2010.

Bonan Min and Marjorie Freedman. BBN's 2016 System for Cold Start Knowledge Base Population. In Proceedings of the Text Analysis Conference (TAC).

Bonan Min, Marjorie Freedman and Talya Meltzer. Probabilistic Inference for Cold Start Knowledge Base Population with Prior World Knowledge. In Proceedings of EACL 2017.

Bonan Min and Ralph Grishman. Challenges in the Knowledge Base Population Slot Filling Task. In Proceedings of LREC 2012.

Heng Ji and Ralph Grishman. Knowledge base population: Successful approaches and challenges. In Proceedings of ACL 2011.

Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In Proceedings of EMNLP 2004.

David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. Lingvisticæ Investigationes, Volume 30, Issue 1, 2007, pages: 3 – 26.

Xiaoqiang Luo. On coreference resolution performance metrics. In Proceedings of HLT-EMNLP 2005

GuoDong Zhou, Jian Su, Jie Zhang and Min Zhang. Exploring various knowledge in relation extraction. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics

8. Language Resource References

Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant and Xiaoyi Ma. From Light to Rich ERE: Annotation of Entities, Relations, and Events. In Proceedings of the 3rd Workshop on EVENTS at the NAACL-HLT 2015.

C. Walker, S. Strassel, J. Medero, K. Maeda. ACE 2005 multilingual training corpus. Linguistic Data Consortium, Philadelphia, 2006.

Simple Large-scale Relation Extraction from Unstructured Text

Christos Christodoulopoulos, Arpit Mittal
Amazon Research Cambridge
{chrchrs, mitarpit}@amazon.co.uk

Abstract

Knowledge-based question answering relies on the availability of facts, the majority of which cannot be found in structured sources (e.g. Wikipedia info-boxes, Wikidata). One of the major components of extracting facts from unstructured text is Relation Extraction (RE). In this paper we propose a novel method for creating distant (weak) supervision labels for training a large-scale RE system. We also provide new evidence about the effectiveness of neural network approaches by decoupling the model architecture from the feature design of a state-of-the-art neural network system. Surprisingly, a much simpler classifier trained on similar features performs on par with the highly complex neural network system (at 75x reduction to the training time), suggesting that the features are a bigger contributor to the final performance.

Keywords: relation extraction, distant supervision, unstructured text

1. Introduction

Knowledge-based question answering relies on the availability of facts – usually in the form of triples, stored in large-scale knowledge bases (KBs) e.g. Freebase (Bollacker et al., 2008), DBPedia (Auer et al., 2007). There are two main sources of facts for such a KB: structured data (e.g. Wikipedia info-boxes, Wikidata) or unstructured text. Undeniably, the former type of knowledge extraction is very accurate and has been the main source of knowledge behind the major industrial knowledge bases. However, the facts extracted from structured sources cover a limited set of high-importance relations, leaving a large number of them implicitly (or explicitly) mentioned in unstructured text (McCallum, 2005).

In order to ground the following presentation, we will present a typical problem from the factual knowledge extraction domain with the following unstructured text from a Wikipedia page:

> "*Carrie Fisher wrote several semi-autobiographical novels, including Postcards from the Edge.*"

The purpose of a fact extraction system is to extract the following facts of the form of `predicate` (*subject*, *object*): **instance of (postcards from the edge, novel)**, and **author of (postcards from the edge, carrie fisher)**, where the first part is a *relation*, and the other parts are the left and right *entities* participating in that relation.

Typically three tasks are involved in generating facts: Entity Recognition, Entity Resolution (or Entity Linking), and Relation Extraction (RE). Entity Recognition and Resolution deal with the task of translating surface strings to KB entities. This includes nominal or pronominal coreference resolution: we should be able to extract the same entity even if the text stated that 'Fisher wrote…' (instead of resolving e.g. to Bobby Fisher) or 'She wrote…' (provided that Carrie Fisher's name was mentioned in a previous sentence). Relation Extraction extracts relation triples (or facts) involving those entities with appropriate relations (also part of the KB schema). Each of these components could be built and operated in isolation, but they affect the performance of each other.

In this paper, we examine the task of RE focusing on extracting knowledge to enrich a large-scale KB (∼billions of facts). We consider a state-of-the-art model that has been applied to hyponymy detection and present a thorough analysis of its application to datasets derived from Wikidata and Alexa KB, a proprietary large-scale triple KB that powers Amazon's Alexa. We also present a new way of generating distant supervision for relation extraction with a simple yet effective way of reducing the noise for the entity resolution.

2. Related work

Relation Extraction is the NLP task of extracting structured semantic relations between entities from natural (unstructured) text. Formally, it can be defined as identifying semantic relations between (resolved) entities and normalise these relations by mapping them to a predefined KB schema. In the NLP community, the RE task evolved out of the Information Extraction projects like MUC in the 1990s (see Chinchor et al. (1993) for an overview) and ACE in the 2000s (Doddington et al., 2004). In both projects the main focus was the automatic extraction of *events* rather than relations (the main difference being that an event is a special type of fact that involves actor entities and occurs at a specific time point) in a limited set of domains (e.g. bombings, company mergers, etc.). This meant that in both projects the number of relations marked for extraction was very limited (3 relations in MUC and 24 in ACE with 7k relation instances for 40k entity mentions).

Starting with those projects, the task of RE was thought of as a pipeline, where the entities were first detected, resolved to a standard schema, and then the RE system would determine which of the possible relations was expressed (if any) between any given pair of entities. Much of the earlier work explored a variety of different features, such as syntactic phrase chunking and constituency parsing (Bunescu and Mooney, 2005; Jiang and Zhai, 2007; Qian et al., 2008), and semantic knowledge like WordNet (Zhou et al., 2005), although Jiang and Zhai (2007) showed that the more complex features might actually hurt the performance

of an SVM-based RE system. The work of Shwartz et al. (2016), that we closely follow, is also using both semantic and syntactic features, by combining the dependency paths between entities, with word embedding representations of both the entities and the lemmas in the dependency paths.

Another related area is relation extraction for Open Information Extraction (OpenIE). Some of the more representative projects in the area, like Reverb (Fader et al., 2011) and more recently ClauseIE (Del Corro and Gemulla, 2013) use syntactic information (PoS tagging / chunking, and dependency parsing respectively) to extract entity and relation phrases. However, unlike OpenIE, we are interested in normalized entities and relations (i.e. that map to a knowledge base).

In this work, we follow a common way of producing training examples for RE is to use *distant supervision* (Craven et al., 1999; Mintz et al., 2009): the assumption is that if any sentence mentions two entities which we know (from a KB) participate in a specific relation, that sentence must be evidence for that relation. In the area of distant supervision, there are two relevant research directions. The first is to use it for directly enriching KBs from unstructured text, as well as leverage the KBs to generate the distant supervision labels (Poon et al., 2015; Parikh et al., 2015). The second direction attempts to reduce the noise in distant supervision labels. A first line of approaches, starting with Data Programming (Ratner et al., 2016), uses generative models to combine multiple sources of weak supervision (e.g. automatically extracted from a KB, rules generated by experts etc.) in order to predict disagreements and overlaps between them and create a noise-aware posterior distribution of predictions. An extension of this approach is Socratic Learning (Varma et al., 2017) which uses the differences in the predictions of the generative model and the main classification system to discover discriminating features and add them back to the generative model. As these approaches require multiple sources of weak supervision, we examine another line of projects which works by aggregating the support sentences[1] for each entity pair (Riedel et al., 2010; Hoffmann et al., 2011). This is the approach that Shwartz et al. (2016) and the current work follow.

2.1. HypeNET

A recent paper (Shwartz et al., 2016) proposed HypeNET, a new method for RE that integrated dependency path information with distributional semantic vector representation of the entities. The authors applied this method to extract hyponyms (i.e. **instance of** relations) and also made a new version of their system publicly available.[2] The training examples used (entity/relation triples) come from a number of sources like WordNet (Miller, 1995), Yago (Hoffart et al., 2013), DBPedia and Wikidata, and the source of the linguistic features (part-of-speech tags, dependency paths, noun phrases) was the 2015 dump of Wikipedia, processed using the spaCy system[3]. Their proposed system achieved by far the best results on their dataset. Since **instance of** is one of the most often used relations (most of the uses are implicit, during inference), we decided to investigate HypeNET as the base of our RE system.

The training examples used by the authors of HypeNET consisted of facts about only one relation. We wanted to build a system that works on multiple relations at a very large scale. Hence, in this work we use two different dataset sources: Wikidata, a publicly-available large-scale KB to aid reproducibility, as well as the larger Alexa KB, built by combining a hand-curated ontology with publicly available data from Wikidata, Wikipedia, Freebase, DBPedia, and other sources.

3. Distant supervision

Following the technique presented in Mintz et al. (2009), and the implementation in HypeNET, we needed to generate training examples where entities X and Y are connected by a relation in the KB and also appear together in the same sentence. When we applied the distant supervision technique presented in HypeNET to our datasets (both Wikidata and Alexa KB) we got poor annotations (see Figure 4(top) for some examples from Alexa KB and section 6.1. for evaluation on both datasets). This could be attributed to the large volume of entities and their corresponding denotations in the KBs, which resulted in a number of ambiguous situations. For instance, "*Chicago*" could denote both the **city** and the **broadway musical show**. In the following section we present our new technique of filtering denotations used for Entity Resolution. This method allows our RE system to scale much better than the original method.

3.1. Page-specific gazetteers

We created a new type of entity gazetteer, based on the main entity of a Wikipedia page, and the knowledge about that entity we have in the KB. The new system, presented on the top dashed box in Figure 1, starts with a Wikipedia URL, retrieves its corresponding ID from the KB for that URL (the main entity), and then extracts entities that are connected directly to the main entity (one-hop distance in the KB graph), by going through all the relations the main entity is involved in (except those involving string literals) and returning the entities on the other side of those relations. For each of the related entities, we collect its denotational strings into a purpose-built gazetteer. Figure 2 shows an example KB subgraph for a target entity (in this case George Springate); it contains all the entities immediately connected to it with relations such as **graduate of** or **instance of**. Also appearing in the graph, are the denotation strings for each one of the related entities.

Note that this approach will reduce the number of extracted entities compared to the original method, but will dramatically improve both the coverage for non-NP entities and precision of entity resolution. One way to increase the recall of this system would be to consider entities with a distance of >1 (entities related to entities related to the main entity). Figure 4 (bottom) shows results obtained by performing entity resolution using page-specific gazetteers. Those examples, as well as the results in section 6.1. show that the noise in the data is significantly reduced.

[1] By support sentences we mean any sentence in the dataset that contains both entities.

[2] https://github.com/vered1986/LexNET

[3] https://spacy.io

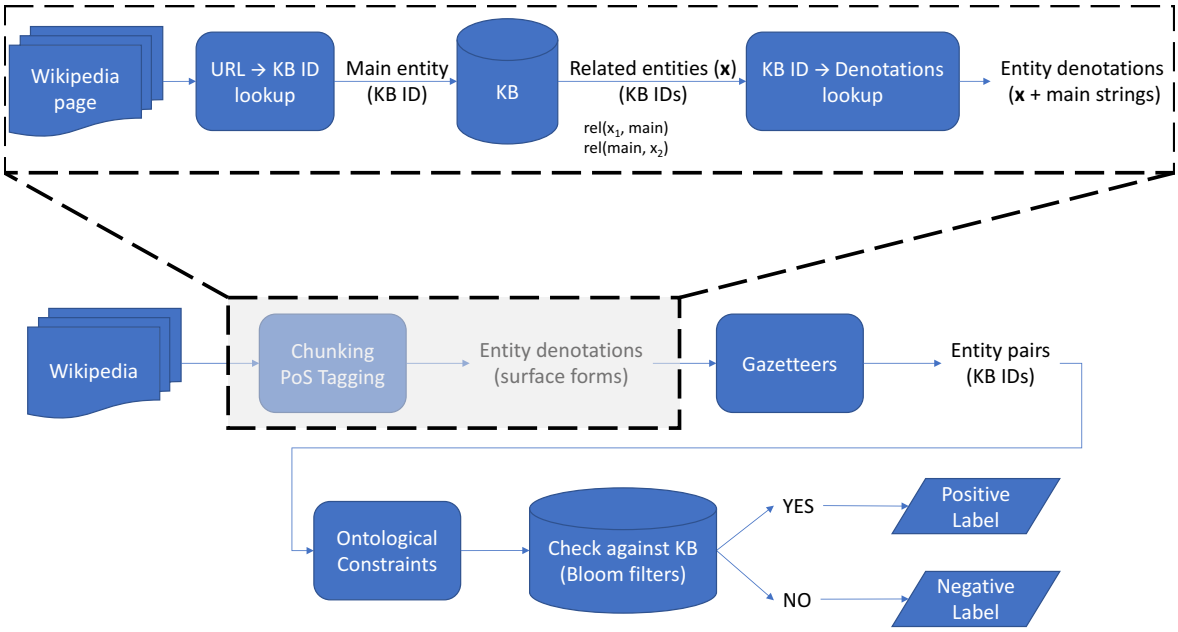


Figure 1: The distant supervision pipeline with page-specific gazetteers. The grey box represents the entity resolution system of Shwartz et al. (2016).

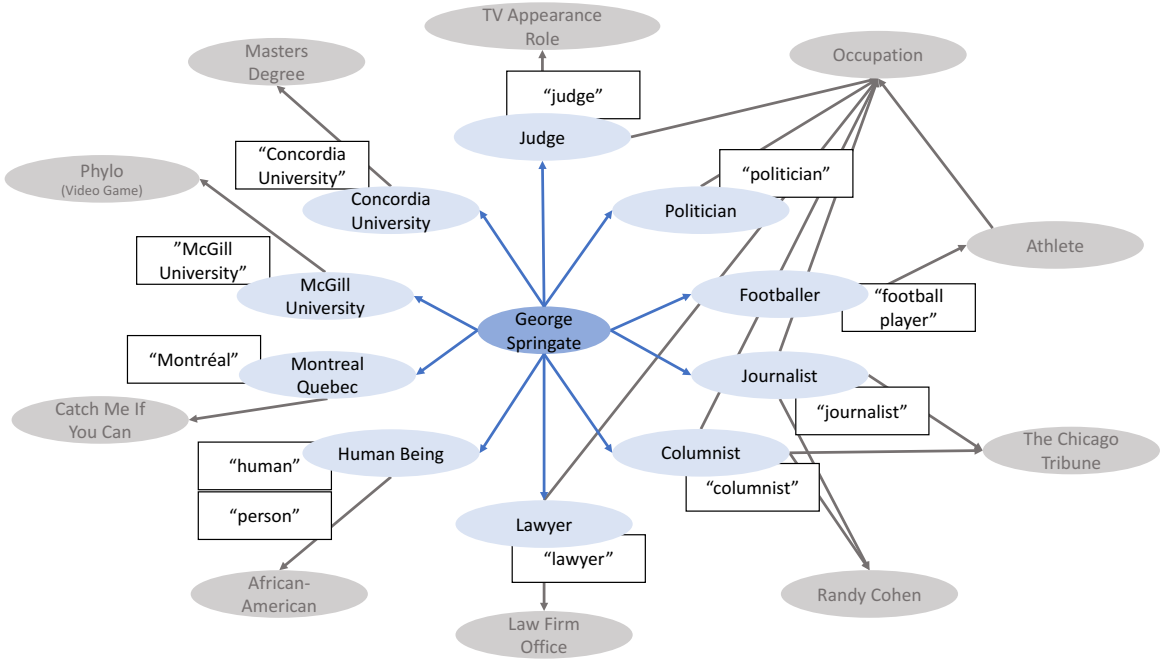


Figure 2: Example KB entry for George Springate. The lightly (blue) shaded ovals represent entities that are within one hop from the main entity and the white boxes are their denotations. The grey ovals represent entities that are two hops away.

3.2. Annotation pipeline

The bottom half of Figure 1 presents the distant supervision generation process adapted from Shwartz et al. (2016) to work with our data. In the original work, the text is processed to split and tokenise the sentences, tag the parts of speech and separate the noun phrases (NPs) – these are the candidate entities. They then construct the dependency path between each possible pair of entities. Each noun/NP pair is checked against the KB for distant supervision. keep only the entities/paths that appear in the list of labelled examples. They also filter out entity pairs that have infrequent paths (occurring fewer than five times), and pairs whose path is more than five tokens long. However, as discussed in the beginning of this section, this approach introduces a lot of noise. To avoid this problem, we use the page-specific gazetteer and a greedy string matching system to scan through the unstructured text and assign KB IDs to the longest-matching substring in a sentence.

The final step was to generate the annotation labels themselves. To do that, we examine each possible pair of entities to see if they participate in the target relation. For the Wikidata KB, we simply checked whether the target relation existed as a property in the data. Considering the large size of Alexa KB, database lookup operations could be very expensive. In order to speed up the lookup for each X rel Y triple, we used two methods. First, before checking against the KB though, we ensure that the pair conforms with the class signature of the relation ('Ontological Constraints'). For example, only a **geographical location** can be the left entity in the **birthplace of** relation. Second, instead of relying on database queries, we used Bloom filters (Bloom, 1970) – a memory efficient probabilistic data structure that can be used to test if an entity is a member of a set. The compression value of a Bloom filter is governed by the accepted false positive rate. We set the false positive rate to 0.001 for our experiments.

Since for any given pair of entities it is much more likely that they are not going to be related, we only keep a small fraction of the negative instances. Following Shwartz et al. (2016), we use a 4:1 negative to positive ratio.

4. Isolating HypeNET features

To discover the effectiveness of the approach of Shwartz et al. (2016), we wanted to separate HypeNET's neural architecture from its input features and use those features with different (and simpler) classifiers. HypeNET's main advantage is that it integrated dependency path features with distributional information about the word lemmas along the path and left and right entities. As our goal was to generate discrete features to be used with more traditional classifiers, we opted for using Brown clusters (Brown et al., 1992) instead of the 50-dimensional GloVe vectors (Pennington et al., 2014) used by Shwartz et al. (2016). The Brown clusters were pre-trained on the Reuters Corpus Vol. 1 (Lewis et al., 2004) using 3,200 clusters.

After evaluating different feature configurations (see section 6.7.), the resulting features were as follows: for each entity pair and for each support, we extracted the dependency path between them and concatenated the lemma, 4-bit prefix of Brown cluster of the lemma, part of speech, dependency relation, and path direction information; to that we added the strings and 4-bit Brown cluster prefix of the left and right entities. The features from different supports were concatenated into one feature list. For example, given the following sentences containing the entity pair **carrie fisher, star wars**: "*In 1977, Fisher starred in George Lucas' film Star Wars*", and "*Fisher became known for playing Princess Leia in the Star Wars film series*". The following is the full list of discrete features extracted, where each space-separated token is a distinct feature, and X and Y are used to replace the left and right entities:

Carrie_Fisher/0111	X/0000/NOUN/nsubj/>
star/0011/VERB/ROOT	in/1101/ADP/prep/<
film/0010/NOUN/pobj/<	Y/0000/NOUN/appos/<
X/0000/NOUN/nsubj/>	become/1111/VERB/ROOT
know/1111/VERB/acomp/<	for/1101/APD/prep/<
play/1111/VERB/pcomp/<	in/1101/APD/prep/<
Y/0000/NOUN/pobj/<	Star_Wars/0011

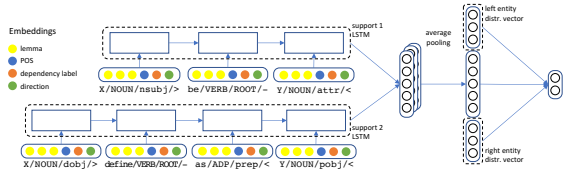


Figure 3: The HypeNET model architecture, reproduced from Shwartz et al. (2016).

4.1. Using a MaxEnt classifier

In the first set of experiments, we used a standard Maximum Entropy classifier from MALLET toolkit (McCallum, 2002) with the discrete features described above. The parameters and settings were kept to their defaults (LBFGS optimizer, with a Gaussian prior variance of 1).

4.2. Using the fastText model

Joulin et al. (2016) recently introduced fastText: a very efficient classifier composed of a simple linear model with a rank constraint. The architecture of the system is very similar to that of Mikolov et al. (2013) except that instead of predicting the middle word in a window, the classifier is predicting a label. For fastText, the input features are token ngrams which are embedded into a single hidden value and fed into a hierarchical softmax classifier. For our experiments, we used fastText's default settings, except for the number of ngrams, which we set to 4.

4.3. Using the HypeNET model

The original version of HypeNET (Figure 3) combines the dependency path-based features with the distributional information in its neural net architecture: for each entity pair, each support (dependency path) token is encoded by a set of embedding layers – one for each linguistic component – and passed into an LSTM layer. The LSTM layers for the whole path are merged by an average pooling layer and the distributional representation of the entities (via embedding layers) is added. Finally, a softmax layer makes a binary classification decision.

We implemented our own version of HypeNET code using Keras (Chollet, 2015) and optimized the learning objective using the Adam optimizer. We modified the basic HypeNET model by making the following changes: i) we allowed the training of word embeddings for lemmas (after initializing them with GloVe embeddings), ii) we replaced the uni-directional LSTM with bi-directional LSTM[4].

5. Evaluation

We want to examine a varied set of connections between the left and right entities, so in addition to the **instance of** relation (P31 in Wikidata) that connects objects to classes, we will examine **birthplace of** (P19) that connects a location entity to a person entity, and **part of** (P527) which links objects to their meronyms. When evaluating against Alexa

[4]The performance of the resulting model was slightly better on the Alexa KB dataset, achieving an F-score value of 94.29±0.21, compared to 94±0.15 for the basic model across the three trials (with a threshold value of 0.5) for **instance of** relation.

His studies were interrupted by army service and at the *end* of the *war* he was forced to return...
instance of (the second world war, cause of death)

In the *intro* to the *song*, Fred Durst makes reference to...
instance of (intro 15367, song)

Turner also released one *album* and several *singles* under the moniker Repeat.
instance of (the singles the 2011 album, album)

Call Your Girlfriend was written by Robyn, Alexander Kronlund and Klas Åhlund, with the latter producing the *song*.
instance of (call your girlfriend 3, song)

Forget Her is a *song* by Jeff Buckley.
instance of (forget her, song)

The *Subei Mongol Autonomous County* is an autonomous *county* within the prefecture-level city of Jiuquan in the northwestern Chinese province of Gansu.
instance of (subei mongol autonomous county, chinese county)

Figure 4: Entity resolution results for the distant supervision training data using Alexa KB and the original pre-processing system of Shwartz et al. (2016) (top), and the new page-specific gazetteers (bottom). The matched strings in the original sentences are *highlighted*.

	Relation	HypeNET μ(F-score)	HypeNET σ	fastText μ(F-score)	fastText σ	MaxEnt F-score
Wikidata	**instance of**	93.90	0.21	**96.44**	0.01	58.45
	birthplace of	92.06	0.90	**93.05**	0.07	66.72
	part of	48.73	2.59	**72.87**	0.16	45.13
Alexa	**instance of**	94.29	0.21	94.31	0.03	83.93
	birthplace of	85.57	0.26	**87.63**	0.01	80.83
	applies to	81.98	1.78	**86.17**	0.01	65.27

Table 1: Mean and standard deviation of the F-score values at 0.5 threshold across three trials for the Wikidata and Alexa KBs, using the MaxEnt, fastText, and HypeNET (our Keras implementation with word embeddings training and bi-directional LSTMs). The MaxEnt model did not have any variance across the trials.

KB, we replaced **part of** with **applies to**, a relation that links an attribute to an object and has no correspondence in Wikidata. We will use the Wikidata KB as a first source of evaluation, and switch to the Alexa KB for a more in depth exploration.

We evaluate all models on a sample of 50K examples for training, 10K examples for validation and test respectively for all relations (except **part of** for which we could only collect 22K training examples). Each example is the collection of all the sentences supporting a $X\ rel\ Y$ triple that have been annotated by the distant supervision system of section 3.. We examine the effect of grouping supports in section 6.4..

6. Results and discussion

We ran each of the following experiments three times (with random initialization) to obtain a measure of variance for their results.

6.1. Distant supervision

The goal of the method presented in section 3.1. was to reduce the number of false positives at the cost of introducing some amount of false negatives (due to missing entities, missing denotations, or missing KB facts). In order to quantify the effect of the new method, we manually annotated 1,000 **instance of** distant supervision examples produced by our new method and the original method used by Shwartz et al. (2016). The original method yielded 67% false positive and 3% false negative examples; the page-specific gazetteer solution returned only 1% false positives and 39% false negatives. After more analysis, 62% of the false negatives (or 24% of the total examples) were cases were the KB contained the **subclass of** relation, which we consider a separate relation (although in the data collected by Shwartz et al. (2016) from Yago and Wikidata it is conflated with **instance of**). The results are similar when using the Wikidata KB: around 1% false positives but only 5% false negatives[5] of which 89% were cases where the KB contained similar relations (like **occupation** for people, or **taxon** for species).

Figure 4 presents a qualitative comparison of the two methods on our KB. We can see that the two problems of spuri-

[5]The lower rate of false negatives can be partially attributed to the exact lookup, instead of the Bloom filters used with Alexa KB.

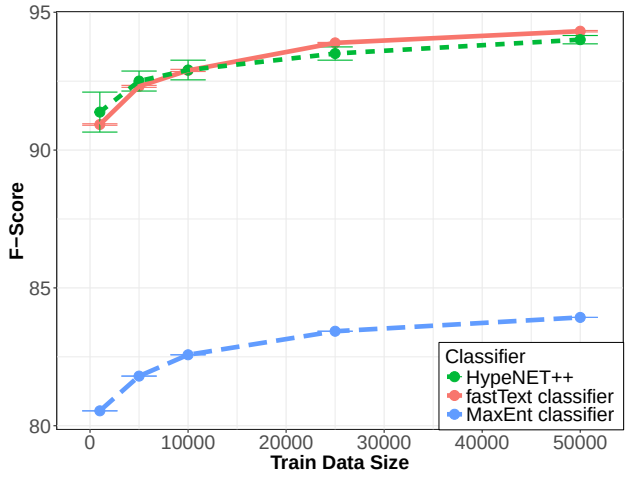


Figure 5: Effect of training data size on the Alexa KB dataset for the **instance of** relation for HypeNET and fastText models at 0.5 confidence threshold.

ous entity matching (e.g. "*end*" to **cause of death**) and non-standard noun-phrase entities (e.g. "*call your girlfriend*") have been successfully addressed by the page-specific gazetteer.

6.2. Model comparison

The results comparing the performance and generalizability of the models (over the three relations) are shown in Table 1. The main takeaway from these results is that the more advanced architecture of HypeNET does not offer a significant advantage over that of fastText when used with (almost) the same input features. As an added benefit, the fastText classifier is dramatically faster than the HypeNET model, with a reduction of training time from around 75 minutes to less than a minute. However as the results of the MaxEnt model show, the features alone are not enough. It is fastText's (and HypeNET's) ability to create higher-dimensional representations of these discrete features that provide the best results.

6.3. Training data size

Another parameter we wanted to explore was the impact of size of the training data since we plan to target relations with fewer training examples in the future. We evaluate the F-scores of the HypeNET, fastText, and MaxEnt models for the **instance of** relation on the Alexa KB dataset.

The results are shown in Figure 5. Note that the numbers in the figure refer to entity pairs, not individual supports (sentences). As expected, the performance of all systems keeps increasing when more training examples are used, but there are two interesting observations to be made. The first is the relative variance of the fastText versus the HypeNET model, especially for the case of fewer than 25k examples. The second is that even with 1,000 training examples, the F-score of both HypeNET and fastText models is above 90%.

6.4. Grouping supports

We also wanted to investigate the effect of grouping the supports (sentences) for each entity pair. As mentioned earlier (Section 2.), this had been proposed as a method to reduce noise in the distant supervision labels (Hoffmann et al., 2011). In Shwartz et al. (2016), the grouping was performed by the mean pooling layer; in the case of the fastText-based system, we simply concatenate the feature tokens from all the supports and feed them into the single hidden layer.

	inst. of	**appl. to**	**bp of**
grouped	94.31	86.17	87.63
ungrouped	93.85	80.90	85.09

Table 2: Effect of grouping supports for each X *rel* Y triple using the fastText classifier on the Alexa KB data (threshold of 0.5).

	inst. of	**appl. to**	**bp of**
satellites	94.31	86.17	87.63
w/o satellites	93.69	85.85	84.42

Table 3: Effect of using dependency path satellite nodes for each X *rel* Y triple using the fastText classifier on the Alexa KB data (threshold of 0.5).

For each of the three relations, we ran the fastText model with and without grouping each entity pair's supports, using exactly the same features in both cases. The Table 2 presents the results. Interestingly, the effect on **instance of** is much smaller than on the other two relations. One possible explanation could be that the page-specific gazetteer method is producing fewer false positives for that relation; more likely, the supports for **birthplace of** and **applies to** are more diverse than those of **instance of**, making their grouping more useful to the classifier.

6.5. Using satellite nodes

We also looked at the role of the dependency path *satellite nodes* (words to the left and right of the entities). This type of features has also been adopted by various systems including Mintz et al. (2009) and Shwartz et al. (2016), and we wanted to establish a basis for its effectiveness across multiple relations. The results, shown in Table 3, show that the clearest advantage of the satellite nodes is for the **birthplace of** relation; for the other two the performance without satellite notes is marginally different. This suggests that the immediate context of the left and right entities is less informative for the **instance of** and **applies to** relations, possible because of the more limited ways that the expression of these relations are syntactically constructed.

	inst. of	**appl. to**	**bp of**
(1) 5 supports	94.55	86.26	87.56
all supports	94.33	85.92	87.63

Table 4: Effect of using all the supports for each X *rel* Y triple using the fastText classifier on the Alexa KB data (threshold of 0.5).

	inst. of	appl. to	bp of
(1)-Brown	94.20	85.93	87.51
(1)-lemma	94.17	84.15	86.65
(1)-POS	94.15	85.93	87.71
(1)-dep	93.59	85.42	86.53
(1)-X/Y entities	93.63	83.89	86.95
X/Y only	91.15	74.20	81.15
full sentence	86.70	77.77	87.09

Table 5: F-score results for the three relations on the Alexa KB dataset. The baseline system (1) is the fastText classifier using the 5 most frequent supports for each X rel Y triple, (1)-dep refers to the system with both the dependency relation and direction features removed, the last system uses all the (lowercased) words in each support as features.

6.6. Using all supports

A comparison is made for the fastText models trained using the 5 most frequent supports for each triple with the ones trained using all available supports. As shown in Table 4, reducing the supports to the most frequent ones slightly increases the performance (except in the case of **birthplace of**) even though on average more than 18K training examples, and more than 3K test examples contain more than 5 supports.

6.7. Feature ablation

As a final step in our exploration, we wanted to measure the impact of each of the features used by the system of Shwartz et al. (2016). Table 5 presents the feature ablation results on the Alexa KB data using the fastText classifier. We compare the full set of features presented in section 4. against feature sets without the Brown clusters, word lemmas, POS tags, dependency information, and the X and Y entities (and their Brown cluster). We also show the results of the system using only the X and Y entities and just the words of the supporting sentences (without extracting the dependency path between entities).

The main takeaway is that for the **instance of** and **applies to** relations, the structure induced by the dependency parser is critical for the system's performance. One explanation is that these relations are not always lexically defined (sometimes expressed with just the verb 'to be' across long subordinate clauses). For the **birthplace of** relation, the system using the full sentence is on par with the best dependency-supported version suggesting that there are strong lexical cues that signify them (like 'born in', or just the presence of a city name).

7. Conclusion

In this paper, we have explored the feature design and network architecture of the HypeNET RE system, and presented a new mechanism for extracting distant supervision data based on our large-scale KB. We found that by replacing HypeNET network architecture with a simple fastText model similar performance is achieved. The main difference between these two architectures is the mechanism of producing the high-dimensional representations: in HypeNET, LSTMs are used, which maintain dependency over longer contexts of dynamic length; in fastText, the window size for the ngrams is fixed. From our experiments, we can infer that dynamic-length context modelling did not bring any gains. Furthermore, we evaluated the effect of grouping of supports and satellites nodes features for various relations. The results from these experiments provide a solid ground to build RE systems for more relations. There are obvious extensions to the current approach, such as using a more sophisticated method for grouping the supports (e.g. an ensemble-based method) and we investigate these in future work.

Beyond architecture improvements, there are two main focus areas for the immediate future: generalising the system to cover very large number ($\sim$1k) of relations, and reducing the sources of noise. The former should be relatively straightforward given the existing architecture for extracting the dataset and training the system. The main obstacle will be to combine the results of the multiple RE systems (one for each relation group) into a single classifier.

Finally, distant supervision as a method itself introduces some errors since not all sentences that mention both entities of a fact express that fact (e.g. the relation **is the director of** between **steven spielberg** and **saving private ryan** is not expressed in the sentence "*The level of violence in Saving Private Ryan makes sense because Spielberg is trying to show ...*"). Going further, we would like to expand the manual annotations to the training/validation sets to assist or replace the distant supervision.

8. Bibliographical References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The semantic web*, pages 722–735.

Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Bunescu, R. C. and Mooney, R. J. (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 724–731. Association for Computational Linguistics.

Chinchor, N., Lewis, D. D., and Hirschman, L. (1993). Evaluating message understanding systems: an analysis of the third message understanding conference (MUC-3). *Computational linguistics*, 19(3):409–449.

Chollet, F. (2015). Keras. `https://github.com/fchollet/keras`.

Craven, M., Kumlien, J., et al. (1999). Constructing biological knowledge bases by extracting information from text sources. In *ISMB*, volume 1999, pages 77–86.

Del Corro, L. and Gemulla, R. (2013). ClausIE: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*, pages 355–366. ACM.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The automatic content extraction (ACE) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G. (2013). YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, 194:28–61.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics.

Jiang, J. and Zhai, C. (2007). A systematic exploration of the feature space for relation extraction. In *HLT-NAACL*, pages 113–120.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. (2004). RCV1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

McCallum, A. K. (2002). Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu`.

McCallum, A. (2005). Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9):48–57.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Parikh, A. P., Poon, H., and Toutanova, K. (2015). Grounded semantic parsing for complex knowledge extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Poon, H., Toutanova, K., and Quirk, C. (2015). Distant supervision for cancer pathway extraction from text. In *Pacific Symposium on Biocomputing*, pages 120–131.

Qian, L., Zhou, G., Kong, F., Zhu, Q., and Qian, P. (2008). Exploiting constituent dependencies for tree kernel-based semantic relation extraction. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 697–704. Association for Computational Linguistics.

Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., and Ré, C. (2016). Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*, pages 3567–3575.

Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. *Machine learning and knowledge discovery in databases*, pages 148–163.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. *arXiv preprint arXiv:1603.06076*.

Varma, P., Yu, R., Iter, D., De Sa, C., and Ré, C. (2017). Socratic learning: Correcting misspecified generative models using discriminative models. *arXiv preprint arXiv:1610.08123*.

Zhou, G., Su, J., Zhang, J., and Zhang, M. (2005). Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

Joint Learning of Sense and Word Embeddings

Mohammed Alsuhaibani Danushka Bollegala
Department of Computer Science, University of Liverpool.
{m.a.alsuhaibani, danushka.bollegala}@liverpool.ac.uk

Abstract

Methods for learning lower-dimensional representations (embeddings) of words using unlabelled data have received a renewed interested due to their myriad success in various Natural Language Processing (NLP) tasks. However, despite their success, a common deficiency associated with most word embedding learning methods is that they learn a single representation for a word, ignoring the different senses of that word (polysemy). To address the polysemy problem, we propose a method that jointly learns sense-aware word embeddings using both unlabelled and sense-tagged text corpora. In particular, our proposed method can learn both word and sense embeddings by efficiently exploiting both types of resources. Our quantitative and qualitative experimental results using unlabelled text corpus with (a) manually annotated word senses, and (b) pseudo annotated senses demonstrate that the proposed method can correctly learn the multiple senses of an ambiguous word. Moreover, the word embeddings learnt by our proposed method outperform several previously proposed competitive word embedding learning methods on word similarity and short-text classification benchmark datasets.

Keywords: Sense Embeddings, Word embeddings, Labelled Data, Unlabelled Data

1. Introduction

The ability to accurately represent the meanings of words is a fundamental requirement for many natural language processing (NLP) tasks. By using accurate word representations, it is possible to improve the performance of downstream NLP applications such as name entity recognition (NER) (Turian et al., 2010), word similarity measurement (Huang et al., 2012), sentiment analysis (Dhillon et al., 2015), word analogy detection (Bollegala et al., 2014), syntactic parsing (Socher et al., 2013) and dependency parsing (Bansal et al., 2014). Moreover, compositional approaches can be used to compute phrase-, sentence- or document-level embeddings from word embeddings (Baroni et al., 2014). Consequently, various methods have been proposed recently that *embed* words in lower-dimensional dense vector spaces, for example, using word co-occurrence information such as skip-gram with negative sampling (SGNS), continuous bag-of-words model (CBOW) (Mikolov et al., 2013) and Global Vectors (GloVe) (Pennington et al., 2014), to name a few.

A common limitation associated with existing prediction-based word embedding learning methods is that they represent each word by a *single* vector, ignoring the potentially multiple senses of a word. For example, consider the ambiguous word *bank* that could mean either a *financial institution* or a *river-bank*. The two senses of *bank* are significantly different, and embedding both senses to the same point is inadequate.

Several solutions have been proposed in the literature to overcome this limitation and learn *sense embeddings*, which capture the sense related information of words. For example, Reisinger and Mooney (2010) proposed a method for learning sense-specific high dimensional distributional vector representations of words, which was later extended by Huang et al. (2012) using global and local context to learn multiple sense embeddings for an ambiguous word. Neelakantan et al. (2014) proposed multi-sense skip-gram (MSSG), an online cluster-based sense-specific word representations learning method, by extending SGNG. Unlike SGNG, which updates the gradient of the word vector according to the context, MSSG predicts the nearest sense first, and then updates the gradient of the sense vector.

Aforementioned methods apply a form of word sense discrimination by clustering a word contexts, before learning sense-specific word embeddings based on the induced clusters to learn a fixed number of sense embeddings for each word. In contrast, a nonparametric version of MSSG (NP-MSSG) (Neelakantan et al., 2014) estimates the number of senses per word and learn the corresponding sense embeddings. On the other hand, Iacobacci et al. (2015) used a Word Sense Disambiguation (WSD) tool to sense annotate a large text corpus and then used an existing prediction-based word embedding learning method to learn sense and word embeddings with the help of the sense information obtained from the BabelNet (Navigli and Ponzetto, 2010) sense inventory. Similarly, Camacho-Collados et al. (2015) used the knowledge in two different lexical resources: WordNet (Miller, 1995) and Wikipedia. They use the contextual information of a particular concept from Wikipedia and WordNet synsets prior to learning two separate vector representations for each concept.

Above-mentioned methods for learning word and sense embeddings require either (a) sense inventories (dictionaries defining the different senses of a word), and (b) word sense taggers that can be applied on unlabelled corpora to generate sense-labelled training data, or (c) manually sense-annotated corpora. Unfortunately, such resources are either under developed or not available for most resource poor languages. On the other hand, methods that learn only word embeddings such as SGNS, CBOW, GloVe etc. can operate on unlabelled corpora. It remains unclear whether unlabelled data can help the process of learning sense embeddings, thereby reducing the manual effort required for creating sense tagged corpora for learning sense embeddings. Revisiting our previous example, only few instances of the word *bank* might be annotated in the labelled data with its sense as a financial institute, however, there might be many other words such as *cash*, *ATM*, *transaction* etc.

that co-occur with bank that could contribute useful information about this particular sense towards the embedding of *bank*. Importantly, such word-level co-occurrences can be obtained purely using unlabelled texts, which are comparatively easier to obtain than sense-labelled texts.

In this paper, we propose a method that uses a large collection of unlabelled texts and a comparatively smaller collection of sense-labelled sentences to learn both word and sense embeddings simultaneously. Our proposed method randomly initialises each word w_i and each of its senses s_{ij} with unique embedding vectors, and update those vectors such that the rank loss between words and senses that co-occur in unlabelled or labelled contexts is minimised over the entire vocabulary of words. In particular, we do not require sense lexicons or dictionary definitions (glosses) of words/senses in this process. Moreover, the proposed method works in an *online* fashion, where we require only a single pass over the data considering one sentence at a time. This is particularly attractive when learning from large collections of unlabelled texts, such as the ukWaC corpus (Baroni et al., 2009) used in our experiments.

We conduct three sets of experiments to evaluate the word/sense embeddings learnt by the proposed method. First, (in § 3.1.) we create a *pseudo sense-labelled* corpus by replacing either two or four words by a unique identifier to create an artificially sense tagged corpus. This approach enables us to generate arbitrarily large sense-labelled data considering different frequency levels of the ambiguous words. Our experimental results on this dataset show that the proposed method can indeed learn word embeddings that are sensitive to the different senses appearing in the dataset. Second, (in § 3.2.) we use the learnt word embeddings to compute the semantic similarity between two words for word-pairs that have been rated by humans. Third, (in § 3.3.) we use the word embeddings learnt by the proposed method to represent textual reviews on several benchmark datasets to solve short-text classification task. Those experiments reveal that by incorporating unlabelled data, we can indeed learn better word embeddings that are sensitive to the word senses compared to what we would get if we had used only labelled data, which is encouraging given the abundance of unlabelled text corpora. Moreover, the experiment shows that by considering the senses in the learning process we can not only learn better sense embeddings, but it also improves the accuracy of the word embeddings as well.

2. Learning Sense Aware Word Embeddings

We propose a method to jointly embed words and their senses in the same lower-dimensional dense vector space. To explain our method, let us consider the lemma of the *target word* $l_i \in \mathcal{V}$ for which we are interested in learning a word embedding $\boldsymbol{l}_i \in \mathbb{R}^d$ in some d-dimensional real space. Here, $\mathcal{V}$ is the vocabulary of words and we use bold fonts to denote word/sense embedding vectors. Given an unlabelled (i.e. not sense-tagged) corpus $\mathcal{U}$, let us denote the set of contexts in which l_i occurs by $\mathcal{K}_i$. Here, for example, a context can be a window of fixed/dynamic length, a sentence or a document. Next, let us consider the lemma of a *context word* l_n that co-occurs with l_n, denoted by $l_n \in \mathcal{K}_i$.

Inspired by the negative sampling method used in SGNS, we would like to learn the embeddings of l_i and l_n close to each other than a word $l_m (\notin \mathcal{K}_i)$ that does not co-occur with l_i. We sample $l_n \sim P_u$ from the unigram distribution P_u such that words that are frequent in the corpus (therefore likely to occur in a given sentence) but do not co-occur with l_i as the *negative* examples. We define the hinge loss J_{ww} for predicting l_n over l_m in all contexts $\mathcal{K}(l_i)$ over the entire vocabulary by

$$J_{ww} = \sum_{l_i \in \mathcal{V}} \sum_{l_n \in \mathcal{K}_i} \sum_{\substack{l_m \sim P_u \\ l_m \notin \mathcal{K}_i}} \max\left(-\boldsymbol{l_i}^\top \boldsymbol{l_n} + \boldsymbol{l_i}^\top \boldsymbol{l_m} + 1, 0\right) \quad (1)$$

J_{ww} can be computed using unlabelled data and does not involve sense embeddings.

We require that the word embeddings must be able to predict not only the co-occurrences of a context word in contexts where a target word occurs, but also must be able to predict the senses associated with the target and contexts words. To model such word vs. sense co-occurrences, given a sense-tagged corpus $\mathcal{L}$, we compute the hinge loss J_{ws} associated with predicting the correct sense s_{nt} of the context word l_n and a randomly sampled sense s_{mg} from the distribution of senses in unigrams P_s that does not occur with l_i as follows:

$$J_{ws} = \sum_{l_i \in \mathcal{V}} \sum_{s_{nt} \in \mathcal{K}_i} \sum_{\substack{s_{mg} \sim P_s \\ s_{mg} \notin \mathcal{K}_i}} \max\left(-\boldsymbol{l}_i{}^\top \boldsymbol{s}_{nt} + \boldsymbol{l}_i{}^\top \boldsymbol{s}_{mg} + 1, 0\right) \quad (2)$$

Here, P_s is computed by counting the occurrences of senses in $\mathcal{L}$.

Likewise, we can compute the hinge loss J_{sw} for predicting a context word using the sense s_{ij} of the target word over a randomly sampled word $l_m \sim P_u$ as follows:

$$J_{sw} = \sum_{l_i \in \mathcal{V}} \sum_{l_n \in \mathcal{K}_i} \sum_{\substack{l_m \sim P_u \\ l_m \notin \mathcal{K}_i}} \max\left(-\boldsymbol{s}_{ij}{}^\top \boldsymbol{l_n} + \boldsymbol{s}_{ij}{}^\top \boldsymbol{l}_m + 1, 0\right) \quad (3)$$

Finally, we require that sense embeddings must be able to predict the correct sense s_{nt} of a context word given the sense s_{ij} of the target word. This requirement is captured by the hinge loss given by (4), where the inner-product between $\boldsymbol{s}_{ij}$ and $\boldsymbol{s}_{nt}$ must be greater than with $\boldsymbol{s}_{mg}$, a randomly sampled sense $s_{mg} \sim P_s$, as given by (4).

$$J_{ss} = \sum_{l_i \in \mathcal{V}} \sum_{s_{nt} \in \mathcal{K}_i} \sum_{\substack{s_{mg} \sim P_s \\ s_{mg} \notin \mathcal{K}_i}} \max\left(-\boldsymbol{s}_{ij}{}^\top \boldsymbol{s}_{nt} + \boldsymbol{s}_{ij}{}^\top \boldsymbol{s}_{mg} + 1, 0\right) \quad (4)$$

We combine the four losses given above into a single linearly-weighted objective given by (5), for some $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$ coefficients.

$$J = J_{ww} + \lambda_1 J_{ws} + \lambda_2 J_{sw} + \lambda_3 J_{ss} \quad (5)$$

We find the word embeddings $\boldsymbol{l}_i$, $\boldsymbol{l}_m$, $\boldsymbol{l}_n$ and sense embeddings $\boldsymbol{s}_{ij}$, $\boldsymbol{s}_{nt}$, $\boldsymbol{s}_{mg}$, $\boldsymbol{s}_{ft}$ such that J is minimised. For this purpose, we compute the partial derivatives of J w.r.t. word and sense embeddings and use stochastic gradient descent (SGD) with initial learning rate set to 0.01.

The derivatives of the objective function given by (5) w.r.t. the variables are given as follows:

$$\frac{\partial J}{\partial \boldsymbol{l_i}} = \begin{cases} \begin{cases} -\boldsymbol{l_n} + \boldsymbol{l_m} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_1 \begin{cases} -\boldsymbol{s_{nt}} + \boldsymbol{s_{gm}} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{6}$$

$$\frac{\partial J}{\partial \boldsymbol{l_n}} = \begin{cases} \begin{cases} -\boldsymbol{l_i} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_2 \begin{cases} -\boldsymbol{s_{ij}} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{7}$$

$$\frac{\partial J}{\partial \boldsymbol{l_m}} = \begin{cases} \begin{cases} \boldsymbol{l_i} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_2 \begin{cases} \boldsymbol{s_{ij}} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{8}$$

$$\frac{\partial J}{\partial \boldsymbol{s_{ij}}} = \begin{cases} \lambda_2 \begin{cases} -\boldsymbol{l_n} + \boldsymbol{l_m} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{l_n} - \boldsymbol{l_m}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_3 \begin{cases} -\boldsymbol{s_{nt}} + \boldsymbol{s_{gm}} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{9}$$

$$\frac{\partial J}{\partial \boldsymbol{s_{nt}}} = \begin{cases} \lambda_1 \begin{cases} -\boldsymbol{l_i} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_3 \begin{cases} -\boldsymbol{s_{ij}} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{10}$$

$$\frac{\partial J}{\partial \boldsymbol{s_{gm}}} = \begin{cases} \lambda_1 \begin{cases} \boldsymbol{l_i} & \text{if } \boldsymbol{l_i}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \\ +\lambda_3 \begin{cases} \boldsymbol{s_{ij}} & \text{if } \boldsymbol{s_{ij}}^\top(\boldsymbol{s_{nt}} - \boldsymbol{s_{gm}}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{cases} \tag{11}$$

3. Experiments and Results

We conduct three sets of experiments to evaluate the embeddings learnt by the proposed method. First, in § 3.1., we qualitatively evaluate the ability of the proposed method to discover known senses in a pseudo-labelled dataset. Second, in § 3.2., we compare the word embeddings learnt by the proposed method against prior work on multiple word similarity benchmarks. Third, in § 3.3., we use the word embeddings learnt by the proposed method to solve short-text classification task and compare the performance against prior work.

3.1. Qualitative Analysis

To verify that the proposed method can learn sense embeddings for the the different senses of an ambiguous word as expected, we conduct the following experiment. We create a pseudo sense-tagged corpus by replacing all occurrences of *two* words by an artificial word in a corpus and tag the mentions of original words as different senses of the artificial word. Due to space limitations, only a selected few examples are shown in Table 1, where we select words with different frequencies (ratio of frequencies indicated within brackets in the first column). For example, we replace *dog* and *chairman* with the artificial ambiguous word *dogychairman* with two senses corresponding dog and chairman. Using ukWaC as the unlabelled corpus, we produced a pseudo-labelled corpus following this procedure. This approach enables us to create arbitrarily large sense-tagged corpora with known senses (and frequencies), which is useful for verifying that the proposed method is working as expected.

We run the proposed method independently on the (a) unlabelled corpus, and (b) the combination of unlabelled and pseudo-labelled corpora to compute word (in the case of both (a) and (b)) and sense ((b) only) embeddings. The nearest neighbouring words (computed using the cosine similarity between the learnt 300 dimensional embeddings) for setting (a) (third column) and for setting (b) (fourth and fifth columns) are shown in Table 1. From Table 1 we see that the nearest neighbours of the word embeddings learnt using only the unlabelled corpus are a mixture of the multiple senses of the ambiguous artificial word. On the other hand, the sense embeddings learnt by the proposed method using both unlabelled and labelled data enable us to produce coherent neighbourhoods, capturing a single sense of the artificial ambiguous word.

Naturally, ambiguous words tend to have more than two different senses, hence it is important for the proposed method to be able capture those multiple senses. For this purpose, in Table 2, we conduct the same experiment reported in Table 1 and described above, however with the difference that instead of replacing all occurrences of *two* words by an artificial word in a corpus, we replace all occurrences of *four* words. This approach allow us to verify the ability of the proposed method to capture the multiple senses of the ambiguous word. From Table 2 we can see that the proposed method was able to detect the correct single sense for each word (fourth, fifth, sixth and seventh column) using both labelled and unlabelled corpora, unlike the mixture of the various senses (third column) produced by using only unlabelled corpus. It is worth noting that even with the frequency variation of the selected *four* words to be replaced by the artificial ambiguous word, the proposed method was able to correctly capture the correct senses.

To further illustrate the ability of the proposed method for learning the sense and word embeddings, we use t-SNE (Maaten and Hinton, 2008) to project the word embeddings to a two-dimensional space as shown in Figure 1. Nearest neighbours of *dogychairman* and its two senses are highlighted. We see that the proposed method successfully learns the different senses of the ambiguous word in the embedding space. For example, the *dog* sense of *dogychaieman* has neighbours such as *dogs, cats* and *pet*, whereas the *chairman* sense has *executive, president* and *director* as the neighbours.

Words	Unique Identifier (ambigious word)	Nearest Neighbours unlabelled corpus	Nearest Neighbours joint (labelled+unlabelled) corpora sense#1	sense#2
career (0.8) africa (0.2)	careeryafrica	south, australia, development,education, professional, developing, china,west, seeking, experience, job, academic	careers, professional, profession, graduate, academic, employment, training, development, job, successful, skills, pursue	india, europe, asia, south, kenya, australia, china, african, southern, countries, pacific, brazil
stock (0.7) dance (0.3)	dancystock	market, music, shares, exchange, company, rolling, markets, art, mix, stocks, theatre, dancing	stocks, market, markets, price, exchange,purchase, prices, investment, company, shares, trading, products	dancing, music, musical, jazz, theatre, art, singing, ballet, drama, artists, opera, song
sea (0.6) chapter (0.4)	chapterysea	river, ocean, introduction, atlantic, island, coastal, section, shore, coast, above, waters,north	ocean, river, coast, mountains, bay, atlantic, shore, beach, coastal, island, sand, water	introduction, section, chapters, book, summary, article, describes,act, review, notes, paragraph,report
dog (0.5) chairman (0.5)	dogychairman	executive, cat, chief, president, bob, david, director,john, horse, cats, brown,fox	cat, puppy, pet, horse, cats,dogs, rat, girl, breed, sheep, horses, boy	executive, chief, committee, director, treasurer, secretary, john, vice, officer, turner, superintendent, deputy

Table 1: Nearest Neighbours of the learnt sense (*two senses*) and word embeddings.

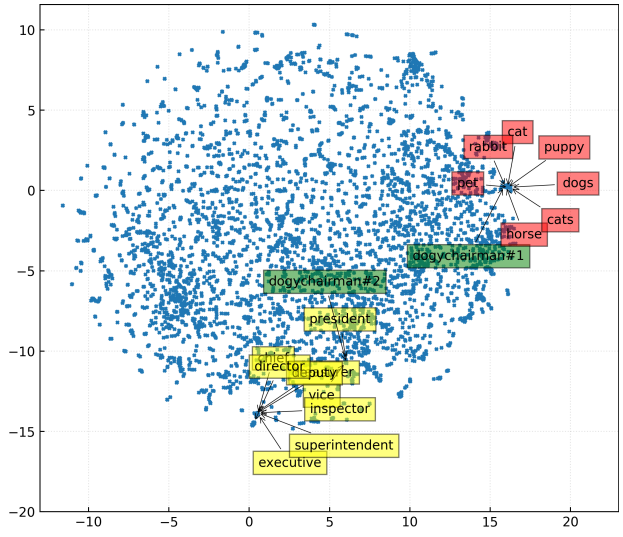


Figure 1: t-SNE projection of word/sense embeddings. Green labels show the two sense embeddings for *dogychairman*, whereas yellow and red labels show the nearest neighbours for the two senses. Best viewed in colour.

3.2. Word Similarity

To empirically compare the proposed method against prior work, we use ukWaC as the unlabelled corpus and SemCor (Miller et al., 1993) as the sense-tagged corpus, and learn word and sense embeddings using the proposed method. We set the context window to 10 tokens to the right and left of a word in the sentence. We used 5 negative samples for both words $\boldsymbol{l}_m$ and senses $\boldsymbol{s}_{mg}$ with 0.75 as a uniform sampling rate. The proposed model converged to a solution with 20 training epochs. We used the Rubenstein-Goodenough (**RG**, 65 word-pairs) (Rubenstein and Goodenough, 1965) dataset as a validation dataset to tune the hyperparameters λ_1, λ_2 and λ_3 given in (5). In particular, we vary the values of the coefficients λ_1, λ_2 and λ_3 and learn the sense and word embeddings using the proposed method afore measuring the Spearman correlation on **RG** dataset. Next, λ_1, λ_2 and λ_3 values are selected based on the highest reported correlation score.[1]

Next, we measure the cosine similarity between two words in human similarity benchmarks using their embeddings, and measure Spearman correlation coefficient between human similarity ratings and computed cosine similarities. A higher correlation with human similarity ratings implies that the word embeddings learnt by the proposed method accurately capture the semantics of the words.

We use several benchmark datasets in our evaluations: WordSim353 (**WS**, 353 word-pairs) (Finkelstein et al., 2002), Miller-Charles (**MC**, 30 word-pairs) (Miller and Charles, 1998), rare words dataset (**RW**, 2034 word-pairs) (Luong et al., 2013), Stanford's contextual word similarities (**SCWS**, 2023 word-pairs) (Huang et al., 2012), **MEN** test collection (3000 word-pairs) (Bruni et al., 2012) and the SimLex-999 (**SimLex**, 999 word-pairs) (Hill et al., 2016).

In Table 3, we compare several word embedding learn-

[1]Setting $\lambda_1 = \lambda_2 = \lambda_3 = 10$ performed consistently well in our experiments.

Words	Unique Identifier (ambigious word)	Nearest Neighbours (unlabelled corpus)	Nearest Neighbours joint (labelled+unlabelled) corpora sense#1	sense#2	sense#3	sense#4
national (0.5) road (0.2) forest (0.2) faith (0.1)	nationalroad	forests,woods, regional,local, international, scottish, lane,junction, roads,belief, british,wales	international, regional, local, scottish, wales,british, association, institute, government,agency, european, scotland	lane,junction, roads,hill, street, park, avenue,bridge, traffic, highway, crossing,mile	forests,woods, valley, forestry, park,deer, hills,habitat, wood, trees, hill,river	belief,christian, religion,god, christ,jesus, christians,religious, christianity, islam, gospel,holy
policy (0.4) farm (0.2) family (0.2) medical (0.2)	policyfarm	policies,farms, farmer,families, medicine,strategy, government,issues, farmers,friends, parents, dental	policies,strategy, government,issues, strategic,development, research,framework, economic,governance, public, legislation	farms,farmer, farming,farmers, dairy,cattle, mill,barn,sheep, cottage,village, agricultural	families,friends, home,parents, relatives,mother, father,children, house,husband, wife, child	medicine, dental, health,healthcare, physicians,doctors, clinical,nursing, physician,veterinary, care,specialist
minister (0.3) transport (0.3) blood (0.2) climate (0.2)	ministertrans	warming,prime, transportation, deputy,secretary, rail,change, liver, ministers,bleeding, global,freight	prime,ministers, deputy,secretary, government,mp,mr, ministry,blair, spokesman,president, chancellor	transportation, infrastructure, local,bus,freight, buses,passenger, airports,public, roads,travel,rail	liver,bleeding, glucose,oxygen, kidney,fluid, skin,cholesterol, lung,tissue, stomach,cells	warming,emissions, global,pollution, environmental,ozone, change,greenhouse, environment,carbon, weather,impacts
council (0.3) album (0.3) floor (0.3) urban (0.1)	councilalbum	albums,borough, floors,songs, councils,basement, song,rural, band,county, debut,district	borough,councils, county,district, authority,committee, housing,community, government,executive, local,authorities	albums,songs, song, band, ep,debut, tunes,cd, punk,singer, pop, tracks	floors,basement, roof,bathroom, room,ceiling, ground, kitchen, bedroom,flat, flooring,lounge	rural,cities, areas,landscape, city,sustainable, development,spaces, communities,towns, transport,suburban

Table 2: Nearest Neighbours of the learnt sense (*four senses*) and word embeddings.

Model	WS	MC	RW	SCWS	MEN	SimLex
CBOW	0.587	0.569	0.251	0.523	0.654	0.291
SGNS	0.633	**0.746**	0.259	0.582	0.677	0.356
GloVe	0.465	0.664	0.265	0.483	0.701	0.327
MSSG	0.658	0.738	0.152	0.632	0.676	0.341
NP-MSSG	0.653	0.715	0.153	**0.639**	0.674	0.355
Proposed	**0.668**	0.702	**0.282**	0.606	**0.734**	**0.372**

Table 3: Performance of the proposed method in comparison with prior work evaluated on word similarity benchmark datasets.

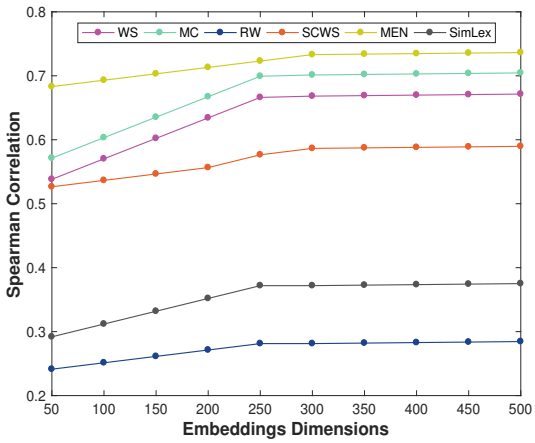


Figure 2: Accuracy vs Dimensionality of the word embeddings evaluated on the **WS**, **MC**, **RW**, **SCWS**, **MEN** and **SimLex** datasets.

ing methods such as sense-insensitive embeddings **CBOW**, **SGNG** (Mikolov et al., 2013) and **GloVe** (Pennington et al., 2014), and sense-sensitive embeddings **MSSG** and **NP-MSSG** (Neelakantan et al., 2014) for learning sense embeddings. We limit the comparison to the state-of-the-art methods for which source codes are publicly available such that we can train all methods on the same datasets and same dimensionality (i.e 300) for a fair comparison.

From Table 3, we see that the proposed method reports the best perfomance in most benchmark datasets, except for the smallest dataset **MC** and the contextual dataset **SCWS**. Table 3 shows that using a sense-tagged corpus is not only beneficial for learning sense embeddings, but also helps in learning better word embeddings. For example, the proposed method reports the highest score among all other models in two of the largest word similarity datasets **MEN** and **SimLex**. **NP-MSSG** reports the best performance in **SCWS** where sentential information is available, which shows an advantage of cluster-based models of capturing the senses. However, the proposed method significantly outperforms (Fisher transformation at $p < 0.05$) **NP-MSSG** and **MSSG** in **RW**, **MEN** and **SimLex**.

Figure 2 shows the effect of the dimensionality of the embeddings learnt by the proposed method. Overall, in all benchmarks, the proposed method is able to learn accurate

Model	CR	TR	MR	SUBJ
CBOW	72.81	73.31	67.40	82.35
SGNS	76.07	72.87	69.41	83.55
GloVe	76.17	73.25	**70.40**	85.10
MSSG	75.53	72.03	69.13	84.98
NP-MSSG	73.82	70.34	68.52	85.05
Proposed	**76.33**	**73.75**	69.26	**85.15**

Table 4: Performance of the proposed method in comparison with prior work evaluated on short-text classification datasets.

word embeddings with as small as 50 dimensions. Moreover, the performance gradually increases with the dimensionality, reaching a peak around 300 dimensions.

3.3. Short-Text Classification

To further empirically compare the embeddings learnt by the proposed method against prior work, we used short-text classification as another extrinsic task for evaluating the word embeddings. We followed the same experimental settings used in subsection 3.2. to learn word and sense embeddings using the proposed method. Next, we used the following four binary short-text classification datasets to solve the classification task: customer reviews dataset (**CR**) (Hu and Liu, 2004) (1494 instances (925 positive and 569 negative)), Stanford sentiment treebank (**TR**)[2] (1806 test instances (903 positive and 903 negative)), movie reviews dataset (**MR**) (Pang and Lee, 2005) (10662 instances (5331 positive and 5331 negative)), and the subjectivity dataset (**SUBJ**) (Pang and Lee, 2004) (10000 instances (5000 positive and 5000 negative)). Each review is represented as a bag-of-words. Next, for each bag, we compute the centroid of the embeddings to represent the review. We then train a binary logistic regression classifier with the train data portion of each dataset, and measure the classification accuracy using the corresponding test data portion.

Table 4 shows the result of our model against other methods on the short-text classification task. Overall, from Table 4, we can see that the proposed method reports the best performance results for most of the benchmark datasets. In particular, the proposed method obtains the best results on **TR**, **CR** and **SUBJ**, whereas **GloVe** obtains the best performance on **MR**. Table 4 supports the conclusion drawn from Table 3 that it is beneficial to consider a sense-labelled corpus to obtain better word embeddings not only to learn sense embeddings. For example, in **SUBJ** which is among the largest short-text classification datasets, the proposed method reports the highest performance. Moreover, the proposed method significantly improves over the sense-sensitive **MSSG** and **NP-MSSG** on both **TR** and **CR**.

4. Conclusion

We proposed a method for jointly learning word and sense embeddings using both an unlabelled corpus and a sense-tagged corpus. The purposed method embed words and their senses in the same lower-dimensional space. Both qualitative and quantitative experiments using unlabelled text corpus with manually sense-tagged and pseudo sense-tagged corpora show that the proposed method can accurately learn word and sense embeddings. Moreover, our experiments on multiple word similarity and short-text classification benchmark datasets show that the proposed method learns accurate word embeddings by modelling senses.

[2]https://nlp.stanford.edu/sentiment/treebank.html

5. Bibliographical References

Bansal, M., Gimpel, K., and Livescu, K. (2014). Tailoring continouos word representations for dependency parsing. In *Proc. of ACL*.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Baroni, M., Bernardi, R., and Zamparelli, R. (2014). Frege in space: A program of compositional distributional semantics. *Linguistic Issues in Language Technology*, 9:5–110.

Bollegala, D., Maehara, T., Yoshida, Y., and ichi Kawarabayashi, K. (2014). Learning word representations from relational graphs. In *Proc. of AAAI*, pages 2146 – 2152.

Bruni, E., Boleda, G., Baroni, M., and Tran, N. K. (2012). Distributional semantics in technicolor. In *Proc. of ACL*, pages 136–145.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2015). Nasari: a novel approach to a semantically-aware representation of items. In *HLT-NAACL*, pages 567–577.

Dhillon, P. S., Foster, D. P., and Ungar, L. H. (2015). Eigenwords: Spectral word embeddings. *Journal of Machine Learning Research*, 16:3035–3078.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., z. Solan, Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20:116–131.

Hill, F., Reichart, R., and Korhonen, A. (2016). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proc. KDD*, pages 168–177.

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proc. of ACL*, pages 873 – 882.

Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2015). Sensembed: Learning sense embeddings for word and relational similarity. In *Proc. of ACL-IJCNLP*, pages 95–105.

Luong, M.-T., Socher, R., and Manning, C. D. (2013). Better word representations with recursive neural networks for morphology. In *CoNLL*.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Mikolov, T., Chen, K., and Dean, J. (2013). Efficient estimation of word representation in vector space. In *Proc. of International Conference on Learning Representations*.

Miller, G. and Charles, W. (1998). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

Miller, G. A., Leacock, C., Tengi, R., and Bunker, R. T. (1993). A semantic concordance. In *Proc. of the Workshop on Human Language Technology*, pages 303–308.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Navigli, R. and Ponzetto, S. P. (2010). Babelnet: Building a very large multilingual semantic network. In *Proc. of ACL*, pages 216–225.

Neelakantan, A., Shankar, J., Passos, A., and McCallum, A. (2014). Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proc. of EMNLP*, pages 1059–1069, October.

Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. ACL*, page 271.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proc. of ACL*, pages 115–124.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: global vectors for word representation. In *Proc. of EMNLP*, pages 1532 – 1543.

Reisinger, J. and Mooney, R. J. (2010). Multi-prototype vector-space models of word meaning. In *Proc. of NAACL*, pages 109–117.

Rubenstein, H. and Goodenough, J. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8:627–633.

Socher, R., Bauer, J., Manning, C. D., and Andrew Y., N. (2013). Parsing with compositional vector grammars. In *Proc. of ACL*, pages 455–465.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. In *ACL*, pages 384 – 394.

Comparing Pretrained Multilingual Word Embeddings on an Ontology Alignment Task

Dagmar Gromann, Thierry Declerck
Technical University Dresden, DFKI GmbH
Nöthnitzer Str. 46, D-01187 Dresden, Germany, Stuhlsatzenhausweg 3, D-66123, Saarbrücken, Germany,
dagmar.gromann@gmail.com, declerck@dfki.de

Abstract

Word embeddings capture a string's semantics and go beyond its surface form. In a multilingual environment, those embeddings need to be trained for each language, either separately or as a joint model. The more languages needed, the more computationally cost- and time-intensive the task of training. As an alternative, pretrained word embeddings can be utilized to compute semantic similarities of strings in different languages. This paper provides a comparison of three different multilingual pretrained word embedding repositories with a string-matching baseline and uses the task of ontology alignment as example scenario. A vast majority of ontology alignment methods rely on string similarity metrics, however, they frequently use string matching techniques that purely rely on syntactic aspects. Semantically oriented word embeddings have much to offer to ontology alignment algorithms, such as the simple Munkres algorithm utilized in this paper. The proposed approach produces a number of correct alignments on a non-standard data set based on embeddings from the three repositories, where FastText embeddings performed best on all four languages and clearly outperformed the string-matching baseline.

Keywords: word embeddings, ontology alignment, multilingual resources, distributional semantics, comparison, evaluation

1. Introduction

Word embeddings constitute a distributed word representation to leverage the semantics of words by mapping them to vectors of real numbers, where each dimension of the embedding represents a latent feature of the word (Turian et al., 2010). They have been shown to be very successful in many NLP tasks (Mikolov et al., 2013; Camacho-Collados et al., 2016) and also ontology alignment (Zhang et al., 2014). Evaluations of embeddings mostly focus on English standard datasets with high frequency single words (Baroni et al., 2014; Riedl and Biemann, 2017) and the few available multilingual comparisons, such as by Camacho-Collados et al. (2016), usually focus on one type of embedding. This paper proposes the use of a non-standard, domain-specific and multilingual dataset with multi-word expressions to compare three different pretrained embedding repositories. Re-use of existing embeddings is attractive since no training time or expertise in learning embeddings is required.

Embeddings abstract away from the string's surface form, which is not the case with syntactic similarity metrics, such as the Levenshtein edit distance (Levenshtein, 1966), conventionally used in ontology alignment (see Cheatham and Hitzler (2013) for a comparison). A semantic representation of words is important in ontology alignment for two major reasons. First, it allows to consider synonyms and terminological variants. For instance, "Schuhe"@de (*shoes*) should have a lower similarity when compared to "Schule"@de (*school*) than compared to its actual synonym "Fußbekleidung"@de (*footwear*). Second, shortened strings, such as abbreviations, co-occur in the same context as their full form but differ strongly in their surface form. However, the computational cost of training word embeddings increases proportionally with a rising number of languages considered.

In this paper, we evaluate three existing embedding libraries, that is, Polyglot (Al-Rfou et al., 2013), a FastText (Bojanowski et al., 2016), and a word2vec embedding repository (Park, 2017) on an ontology alignment task in four languages: English, Spanish, German, and Italian. We compare the embedding libraries to a Jaccard baseline, which is a string-matching technique that has been shown to perform well on multilingual ontology alignment (Cheatham and Hitzler, 2013). For this task, two multilingual ontologies that exist in all four languages with enough overlap to allow for an alignment are needed and the structure of each ontology should be the same in all its languages, whereas the structure of the two multilingual ontologies might not be exactly the same. We test the application of word embeddings to the mapping of ontology labels of two lightweight industry classification ontologies: the GICS (Global Industry Classification Standard[1]) and the ICB (Industry Classification Benchmark[2]) classification systems. To reduce the Out Of Vocabulary (OOV) ratio, we present and utilize a decomposition for German compounds. Our main contribution is, on the one hand, an experiment with existing embedding repositories on a new kind of task, namely multilingual, domain-specific ontology alignment, and, on the other hand, a cost-effective semantic string matching method for multilingual ontology alignment. We also publish the code utilized in this experiment[3].

As regards structure, we first describe the utilized embedding repositories and ontologies before we specify the chosen methodology including the compound decomposition, vector concatenation, and ontology alignment method. Section 5 quantifies and describes the obtained results, which are discussed in Section 6. We conclude by providing some related approaches and concluding remarks.

[1] https://www.msci.com/gics
[2] http://www.icbenchmark.com/
[3] https://github.com/dgromann/OntologyAlignmentWithEmbeddings

2. Embedding Resources

Polyglot (Al-Rfou et al., 2013) represents a repository of word embeddings in more than 100 languages trained on the Wikipedia corpus for each language (Al-Rfou et al., 2013). The used vocabulary consists of the 100,000 most frequent words in each language and the vectors use a dimensionality of 64. The embeddings were trained using a neural network implementation in Theano (Bergstra et al., 2010). Surface forms of words are mostly preserved, that is, a minimum of normalization is applied, which allows for a querying of the resource without major preprocessing for most languages (German is an exception in our dataset, see 4.1. for details).

The FastText embeddings (Bojanowski et al., 2016) are available in 294 languages, use the most common vector dimensionality of 300 and are trained on the Wikipedia corpus using FastText (Bojanowski et al., 2016). In contrast to the other resources, this embedding library considers subword information of the vocabulary in the corpus. By using a skipgram model, a vector representation for each character n-gram is produced and words are the sum of their character representations.

A repository of word2vec embeddings (Park, 2017), called word2vec for short from now on, available in more than 30 languages was used as a comparative library to fastText and trained on the Wikipedia corpus using the popular word2vec approach (Mikolov et al., 2013). The dimensionality of the vectors is also 300, but no subword information was considered here. The cited word2vec embedding library does not contain English, which is why we trained the English word embeddings using the exact same method as for the other languages with the code provided in the repository (Park, 2017) and also compare to the results obtained with the pretrained embeddings trained on the Google News corpus with dimensionality 300 (Mikolov, 2013).

3. Ontology Alignment Data Set

Industry classification systems enable the comparison of companies across borders and languages. They help investors to diversify their asset portfolios by sorting stocks into sectors and industries. Thus, a portfolio manager can choose stocks from different classes to mitigate the risk of centering extensively on one sector or industry. However, due to numerous and often competing classification systems, the resulting resources are inconsistent on a taxonomic and terminological level. We utilize two widely accepted classification systems in our multilingual alignment method in English, German, Spanish, and Italian: the Global Industry Classification Standard (GICS) and the Industry Classification Benchmark (ICB). The English version of GICS has been translated to other languages, which means that all languages of GICS are fully parallel in structure, which is also true for ICB.

3.1. Global Industry Classification Standard (GICS)

The Global Industry Classification Standard (GICS) represents industry sectors in a lightweight ontology developed by MSCI and Standard & Poor's[4]. The GICS structure consists of 10 sectors, 24 industry groups, 68 industries and 154 sub-industries. GICS is offered in 11 different languages. It contains 256 labels for each language, that is, a total of 2,816 labels. For our experiment dealing with 4 languages, we have thus 1,024 labels. Each sub-industry is equipped with a natural language definition.

3.2. Industry Classification Benchmark (ICB)

The Industry Classification Benchmark (ICB) developed by Dow Jones and FTSE[5] consists of four major levels. There are ten main industries which are subcategorized in an increasingly finer classification into 19 supersectors, 41 sectors and 114 subsectors. Each stock is uniquely classified into one of the 114 subsectors, which consequently uniquely classifies it into one of the sectors, supersectors, and industries. ICB is offered in 14 languages and contains 184 labels for each language, that is, 2,576 labels in total for all languages. For our experiment with four of those languages, we have thus 736 labels. Each subsector is equipped with a natural language definition.

3.3. Comparing ICB and GICS

Both systems classify a company according to its primary revenue source, apply four major levels to their structure and have a comparable number of subcategories. We compare the ten top levels of both hierarchies. Apart from the consumer related sector they seem to be very similar, four of them are even exact string matches. One major difference is to be found in the consumers section. GICS differentiates between staples and discretionary containing both goods and services, whereas ICB distinguishes consumer goods from consumer services. As this regards the top-level classification, it is an important aspect to be considered in the alignment strategy. The terms used to designate equivalent categories differ substantially.

Both classifications apply unique integers for indexing the concepts, to which the labels are associated. While GICS and ICB have both four conceptual levels, they use each a different strategy for encoding the taxonomic positions. GICS adds 2 digits per level ($15 > 1510 > 151010 > 15101010$), while ICB increases the numbers for marking each level ($1000 > 1300 > 1350 > 1353$). Both classifications at times use identical strings to label elements at different levels, e.g. "Banks"@en (ICB8300, ICB8350, ICB8355).

4. Methodology

We use an element-level matching algorithm to calculate the semantic similarity between sets of multilingual labels. The embeddings for each word in each label are retrieved individually and then combined. The similarity is calculated based on a cosine function, the most frequently used similarity metric for embeddings, between the combined vectors of each label in each ontology.

[4]See respectively http://www.msci.com/products/indices/sector/gics/ and http://www.standardandpoors.com/indices/gics/en/us

[5]See http://www.ftse.com/Indices/Industry_Classification_Benchmark/index.jsp

4.1. Preprocessing

Our preprocessing focuses on optimizing short textual sequences for word embedding retrieval since we align ontology labels. To this end, we remove stop words, numbers, and punctuation and decompose complex German compounds to reduce the OOV rate. We decided against further preprocessing, such as lemmatization or stemming.

Normalization

To minimize the OOV value, we repeat the retrieval attempt with several case representations of the word, that is, upper case initial letter, lower casing all letters, and representing all characters as upper case (title casing).

German Noun Decomposition

Compounding languages, such as German, allow a potentially infinite creation of new words, which cannot possibly be covered by a single embedding repository. Decomposition of such word constructs does indeed reduce this potential infinity to a nearly finite set of words that are used in compounds, and which have a much higher probability of being covered by the utilized repositories. But decomposition is not a trivial task. Fortunately, we can use two data sources in our experiment: the GermaNet list of 66,059 compounds together with the explicit description of their components[6]. The second resource we use is generated directly from the data of GICS and ICB: a program first traverses all the labels and definitions used in the German version of the classification systems and collects all the words used. In a second run, words that are re-used in larger strings are marked as components of a compound and thus also as independent words. This strategy allows to significantly increase the number of German words covered by the individual repositories and bring them closer to the other languages (see Table 1 for details).

4.2. Vector Composition

In order to obtain the best estimation of the similarity of two labels $sim(label_1, label_2)$ each label is split into its k number of individual words. We then retrieve the vector representation $\vec{v}_i$ for each word from the individual embedding repositories, so we query for the vector $\vec{v}_i : \forall w_k \in label_i$ for all words w_k in each label $label_i$.

Since OOV occurrences are possible, we need to compose the vectors in a way that allows for the indication of missing words in a longer sequence of words representing the label. To this end, we adapt the concept of lexical semantic vectors (Konopik et al., 2016). We create a combined vocabulary $L = unique(label_1 \cup label_2)$ of all unique words in $label_1$ of GICS and $label_2$ of ICB as represented in Figure 1. Similarity of the first word of L and the first word of $label_1$ is calculated as the cosine metric of their respective embeddings, which we retrieve from the embedding repositories (in this example we used Polyglot embeddings). Then, the first word of L is compared to the second word of $label_1$ until the first word of L has been compared to all words of $label_1$. We create a vector $\vec{m}$ that contains the maximum cosine similarities between each word of L and all other words in $label_1$. For instance, the "Renewable" embedding in L obtains the highest similarity (1.0) with the "Renewable" embedding in $label_1$ of GICS. The process is iterated for the second word of L to obtain the second value of vector $\vec{m}$. To obtain $\vec{n}$, the words of L are compared to all words of $label_2$ as described for $label_1$. For instance, the "Renewable" embedding of L obtains the highest similarity value (0.695) with the "Conventional" embedding of $label_2$ of ICB. The dimensionality of the resulting vectors $\vec{m}$ and $\vec{n}$ depends on the number of words in L. The similarity $sim(label_1, label_2)$ is calculated as the cosine value between $\vec{m}$ of GICS and $\vec{n}$ of ICB for each of their labels.

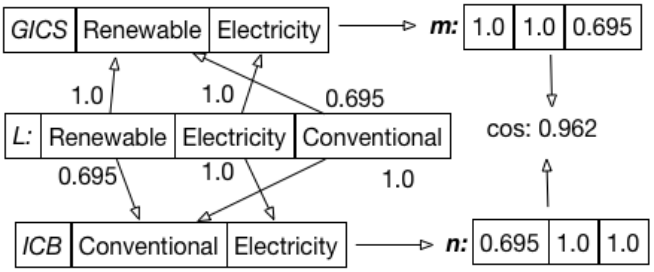


Figure 1: Vector Comparison Method to Measure Semantic Distance

If a word is not in the vocabulary of the embedding repository, we do not have an embedding to calculate the cosine similarity with other word embeddings. To solve this problem, we fill the slot of this word in the final vector $\vec{m}$ respectively $\vec{n}$ with the average of all other values[7]. This means that we sum all calculated cosine similarities for in-vocabulary words of the label with an OOV and divide the sum by the number of in-vocabulary elements of the same label. The resulting value provides us with the cosine-similarity value in vector $\vec{m}$ respectively $\vec{n}$ for the OOV words as depicted Figure 2, where "Nondurable" is not in the embedding library, which in this example case is the Polyglot library.

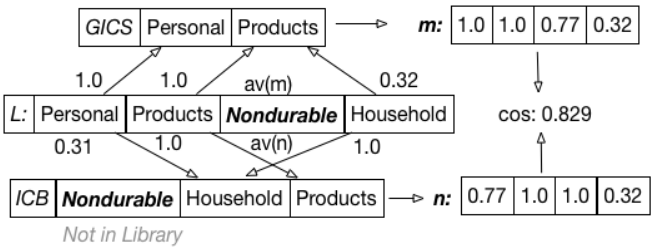


Figure 2: Handling OOV in Vector Composition

4.3. Ontology Alignment Task

Two ontologies modeling the same domain frequently differ due to design choices of the engineers. To enable their

[6] http://www.sfs.uni-tuebingen.de/lsd/compounds.shtml

[7] We tested with values of -1, close to zero, and the average similarity of all other words in the same label for OOV words and found a better performance using the average similarity of all other values in the vector.

re-use, extension, and comparison, ontology alignment is useful. It has been shown that string similarity metrics alone can achieve results competitive to state-of-the-art ontology alignment systems (Cheatham and Hitzler, 2013). For instance, the Jaccard distance, which is based on the number of words two strings have in common divided by the total number of words of the two strings, has been found to be very effective in multilingual ontology alignment scenarios (Cheatham and Hitzler, 2013). For this reason we chose Jaccard as the baseline for measuring the performance of the proposed embedding-based approach. However, linguistic condensation strategies, such as compounding or abbreviations, and the lack of semantic context can pose a serious challenge to string-based ontology alignment methods.

Ontology alignment represents the problem of identifying a set of correspondences $\{e_1, e_2, sim | e_1 \in O_1, e_2 \in O_2\}$ where e_1 is an element in O_1 and e_2 is an element of O_2 and sim represents the confidence of each individual correspondence in the set of alignments. In this first experiment, our entities e correspond to class labels only. The correspondence is represented as an equivalence relation between two entities $e_1 \equiv e_2$, where each entity is represented by its natural language label. This correspondence relation is based on the assumption that a one-to-one matching between entities exists. We repeat the alignment process for each language-specific ontology, since they are provided separately.

To align the two input ontologies in each language, we employ the Munkres assignment algorithm (Munkres, 1957). Given a non-negative `n x n` matrix it optimizes the alignment of the i-th row with the j-th column. Our similarity matrix represents all GICS labels as rows and all ICB labels as columns and the value if the i-th row and j-th column represents their similarity $sim_{ij} = (label_i, label_j)$.

4.4. Cross-Lingual Experiment

Cross-lingual approaches to ontology labels range from benefiting from an alignment across languages to complement elided content in labels (e.g. Gromann and Declerck (2014)) to cross-lingual ontology matching (e.g. Spohr et al. (2011)). Even though our paper focuses on a multilingual alignment process, we conducted a small experiment utilizing cross-lingual information. In order to benefit from the parallel structure across languages, we aggregate each similarity value and related ICB identifier from each language for a given GICS element. In other words, for each GICS element we obtain four ICB ids and four similarity values. We first count the frequency of occurrence of each ICB id in the list of four ids. If the id occurs more than once, it is selected as the chosen alignment target for this specific GICS element. If no id occurs more than once, we select that ICB id with the highest associated similarity value. For instance, the element GICS 15102010 with label "Construction Materials" is mapped to the following ICB elements[8]: "Heavy Construction" (ICB 2357) in English, "Household Goods & Home Construction" (ICB 3720) in Italian, "Building Materials & Fixtures" (ICB 2353) in German and Spanish. The assigned similarity values are $[2357 : 0.812, 3720 : 0.797, 2353 : 0.884, 2353 : 0.924]$ where the last two values correspond to first the German and then the Spanish cosine value. In this example our simple algorithm selects ICB 2353 because it occurs twice, which is a correct mapping that corresponds to the manual alignment. If it had occurred only once, our basic approach would have still selected ICB 2353 since it also has the highest similarity value with 0.924 in Spanish.

[8]For a better understanding of the example we only provide the English labels of the ontology elements here.

4.5. Evaluation

In order to evaluate our method, two experts created a manual alignment of GICS and ICB elements based on their labels and natural language descriptions in English. This monolingual mapping is sufficient, since the other language versions of each standard are translated from English and remain structurally equivalent to the source language. We calculate the inter-annotator agreement to be 0.75. Alignments that reached no agreement were evaluated by a third expert. The automatically created alignment is compared to the manually created alignment, which is how we obtain the metrics presented in Section 5. We only consider elements that can directly be mapped across the two ontologies - elements that require a one-to-many mapping are ignored for this first experiment. For instance, "Marine" (GICS 20303010) in GICS is defined as any maritime transportation of goods and passengers excluding cruise ships, while ICB differentiates between "Marine Transportation" (ICB 2773) for commercial markets, such as container shipping, and "Travel & Tourism" (ICB 5759) providing passenger transportation including ferry companies. Thus, a direct mapping of the GICS concept to one ICB element could not be established in this particular example. Our final dataset contains 155 labels from each ontology in each language, that is, a total of 620 labels per ontology across all languages combined.

5. Results

To quantify our comparison, we first evaluate the coverage of our vocabulary by each embedding library. We continue to quantify the performance of all embedding libraries by F-score on the described dataset and in comparison to a string-matching baseline.

5.1. OOV scores

To provide a better explanation of the F-measure results, Table 1 provides statistics on Out Of Vocabulary (OOV) scores for each set of multilingual embeddings across both ontologies, ICB and GICS, without duplicates. To increase coverage, each word of a label not directly available in the library was queried again with different case settings, i.e., lower, upper, and title case. For German, the compound decomposition described in Section 4.1. was conducted.

Reasons for an inability to find specific words from a label in the embeddings library varied strongly with each library. Polyglot had difficulty finding abbreviations, such as "REITs" (Real Estate Investment Trusts), and unusual compounds, such as "Multi-Utilities". FastText only struggled with unusual compounds in all languages but Italian, where nouns with articles constituted the biggest problem,

	Polyglot	FastText	word2vec
English	21	5	41
Italian	22	17	43
German*	70 (198)	38 (99)	125 (218)
Spanish	21	4	43

*value in brackets is before decomposition

Table 1: Word Coverage Across Resources

e.g. "all'Ingrosso". Articles where no problem for Polyglot and FastText handled abbreviations without any difficulty. So the type of further preprocessing that would be required to further increase coverage differs with the embedding library. The extraordinarily high number of OOVs in German can be attributed to high frequent ellipses in the vocabulary, such as "Abfall-" (waste) in the label " Abfall- und Entsorgungsdienstleister" (ICB 2799 "Waste and Disposal Services"), and complicated compounds, such as "Arzneimitteleinzelhändler" (ICB 5333 "Drug Retailer"). Results for word2vec could be improved for Spanish and Italian by lemmatizing, however, for German the major problem are the compounds that are not considered in the library. Using the proposed decomposition method, we could improve on this situation as shown in Table 1 where the values in brackets represent the OOV words prior to decomposing.

5.2. Alignment Statistics

A generated similarity matrix calculated on the basis of the individual word embeddings is submitted to the simple Munkres algorithm to compare the obtained alignment to the manual gold standard alignment. Table 2 quantifies the comparison across embedding repositories and to a Jaccard baseline. The English embeddings of Polyglot seem to largely outperform all other tested languages of the same repository. The same could be observed for the other repositories in Table 2. All embeddings were trained on the Wikipedia corpus but different methods were utilized in obtaining the embeddings. It can be seen from Table 2 that FastText and its encoded subword information outperforms the other embedding representations. However, the simple string-matching Jaccard similarity baseline outperforms Polyglot in all languages and outperforms the word2vec pretrained embeddings in German.

	Jaccard	Polyglot	FastText	word2vec
English	0.692	0.652	**0.830**	0.760 (0.826*)
Italian	0.488	0.385	**0.686**	0.517
German	0.473	0.434	**0.652**	0.418
Spanish	0.495	0.360	**0.745**	0.582
All	0.678	0.675	**0.812**	0.750

* Using embeddings trained on the Google News Corpus.

Table 2: Embedding Comparison by F-Score on Ontology Alignment Task

The first four rows of Table 2 quantify the results of all four languages, while row five describes a cross-lingual optimization experiment. Results of this first experiment quantified in Table 3 show that FastText has the highest correspondence across all languages, since it has the highest number of recurring ICB ids across languages. Incidentally FastText also has the highest F-measure in all languages, while Polyglot, the repository with the lowest F-measure of all methods also has the lowest correspondence of ICB target ids across languages. This simple cross-lingual comparison will be replaced by a more principled cross-lingual approach in future work.

	Jaccard	Polyglot	FastText	word2vec
most_common	89	71	**138**	107
high_sim	66	84	17	47

Table 3: Cross-Lingual Experiment

Interestingly this cross-lingual experiment leads to quite different F-scores as can be seen from the last row of Table 2. In general, the composition across languages has a positive impact on the F-score in Italian, German, and Spanish in all cases. However, Jaccard, FastText, and word2vec provide better results in the monolingual English setting, while the composition of results across languages outperforms the English results for Polyglot.

6. Discussion

When using pretrained embeddings, the handling of OOV is an important issue. Decomposition and preprocessing for the German data proposed in this paper could still be refined to include for instance ellipses resolution (Gromann and Declerck, 2014). While FastText provides a comparatively high coverage of vocabulary across all languages, the other libraries could benefit from some more refined preprocessing of the GICS and ICB labels. Nevertheless, the performance of the pretrained embeddings on domain-specific multi-word labels of lightweight ontologies is very promising, in particular FastText.

FastText outperforms all the other three methods in all languages. We attribute the success of this embedding library to two main factors: i) the library has fewer OOV words than the other embedding repositories, ii) subword information is considered when training the embeddings and each embedding is the aggregated result of its character n-grams. It seems as if this more morphologically oriented type of embedding in FastText is more adequate for domain-specific multi-word expressions as found in ontologies. FastText is also the one repository with most corresponding results across all languages, as our small cross-lingual comparison shows.

One of the main assumptions for Polyglot's performance below the Jaccard baseline is the high degree of OOV words in the library in all languages. However, word2vec has a higher OOV rate and provides better results than Polyglot. Thus, it can be followed that the settings and parameters of the training method of the embeddings make a difference since all three repositories are trained on the Wikipedia corpus and applied to the same task in this paper but differ in resulting F-score. Those parameters include the chosen dimensionality of the embedding, which in case of Polyglot

is 64 as opposed to 300 in the other libraries. The factor of dimensionality has been shown to have a substantial impact on the accuracy that can be obtained with the vector representations (Mikolov et al., 2013). Of course, also the corpus chosen for training has a large impact as can be seen by the comparison to the word2vec embeddings trained on the Google News Corpus.

Similarity in the vector space is not necessarily semantic similarity, but might be any type of relatedness. Thus, labels such as "Renewable Electricity" and "Conventional Electricity" might obtain a very high cosine value as shown in our vector composition example even though they are not synonymous and in fact almost are opposites. We would expect this problem to be more prominent in this highly domain-specific scenario, however, as can be seen from the good results obtained by FastText, this difference in similarity relation does not seem to be detrimental to the overall application of word embeddings to a label-centric ontology alignment task.

In our results, English embeddings obtain better results than embeddings used in other languages to align ontology labels. Both ontologies were originally produced by English-speaking companies in English and then translated to the other languages. It has been shown that non-native language and translations are closer to each other than they are to the native language (Rabinovich et al., 2016). Thus, this difference in accuracy cannot necessarily be attributed to the embeddings but more likely to the input labels. For this purpose it would be interesting to repeat the experiment on a multilingual and structurally parallel standard dataset.

We believe that this method can also be applied to other interesting scenarios. The similarity between the labels also hints at the similarity between the entire resources. Thus, this method could potentially be used to find similar resources in a repository of ontologies, which is offset in comparison to simple string matching by embeddings and multilinguality. It can also be used to do a lightweight checking of structural problems in ontologies. For instance, it can be considered bad practice to assign identical or almost identical labels to different concepts. Identical surface forms and equivalences can be detected using the proposed method.

7. Related Approaches

Two main lines of research are related to the proposed comparison of word embeddings on the task of multilingual ontology alignment: (i) comparisons of word embeddings in general, (ii) use of word embeddings on ontology alignment tasks. Most embedding comparisons focus on high frequent English words for various tasks (Baroni et al., 2014; Riedl and Biemann, 2017) or if multilingual, evaluate specific embeddings (Camacho-Collados et al., 2016). However, approaches for multilingual, domain-specific multi-word expressions are hard to find for embedding comparisons. The use of word embeddings in ontology-related tasks is a rather recent development. Zhang et al. (Zhang et al., 2014) utilize word embeddings they learn from Wikipedia texts to match the OAEI 2013 benchmark ontologies and three real-world ontologies including Freebase in English. In their embedding comparison, the ones trained solely on Wikipedia performed best but in an overall evaluation a hybrid embedding and edit-distance method outperformed the others.

8. Conclusion

In this initial experiment we evaluate the use of pretrained word embeddings in four languages for the task of real-world domain-specific ontology alignment. We propose a method that is able to handle missing embeddings for individual words applied to a non-standard dataset. One of the reasons for this decision is our interest in domain-specific labels and multilingual, aligned contents. Furthermore, we were interested in a real-world scenario that also has a practical value for industry. Pretrained embedding libraries achieve promising results, particularly FastText with a greater consideration for morphological structures seems very apt for the task of string-based ontology alignment. Future work will consider the integration of taxonomic and axiomatic knowledge from the ontologies with the embeddings to improve the alignment results as well as the utilization of existing knowledge-rich embeddings, e.g. ones that integrate ConceptNet structures into their representation. Furthermore, we are interested in the further utilization of cross-lingual information in the alignment process, such as treating OOV words in one language by using word embeddings available in another language.

Acknowledgement

The DFKI contribution to this work has been partially funded by the BMBF project "DeepLee - Tiefes Lernen für End-to-End-Anwendungen in der Sprachtechnologie" with number 01-W17001, and the project QT21, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 645452.

9. Bibliographical References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria, August. Association for Computational Linguistics.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL (1)*, pages 238–247.

Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., and Bengio, Y. (2010). Theano: A cpu and gpu math compiler in python. In *Proc. 9th Python in Science Conf*, pages 1–7.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2016). Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence*, 240:36–64.

Cheatham, M. and Hitzler, P. (2013). String similarity metrics for ontology alignment. In *International Semantic Web Conference*, pages 294–309. Springer.

Gromann, D. and Declerck, T. (2014). A cross-lingual correcting and completive method for multilingual ontology labels. In *Towards the Multilingual Semantic Web*, pages 227–242. Springer.

Konopik, M., Prazák, O., Steinberger, D., and Brychcín, T. (2016). Uwb at semeval-2016 task 2: Interpretable semantic textual similarity with distributional semantics for chunks. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 803–808.

Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38.

Rabinovich, E., Nisioi, S., Ordan, N., and Wintner, S. (2016). On the similarities between native, non-native and translated texts. *arXiv preprint arXiv:1609.03204*.

Riedl, M. and Biemann, C. (2017). There's no 'count or predict' but task-based selection for distributional models. In *12th International Conference on Computational Semantics (IWCS)*. ACL (W17).

Spohr, D., Hollink, L., and Cimiano, P. (2011). A machine learning approach to multilingual and cross-lingual ontology matching. In *International Semantic Web Conference*, pages 665–680. Springer.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Zhang, Y., Wang, X., Lai, S., He, S., Liu, K., Zhao, J., and Lv, X. (2014). Ontology matching with word embeddings. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 34–45. Springer.

10. Language Resource References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. https://sites.google.com/site/rmyeid/projects/polyglot.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Pre-trained word vectors. https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md.

Mikolov, T. (2013). word2vec: Tool for computing continuous distributed representations of words. https://code.google.com/archive/p/word2vec/.

Park, K. (2017). Pre-trained word vectors of 30+ languages. https://github.com/Kyubyong/wordvectors.

A Large Resource for Patterns of Verbal Paraphrases

Octavian Popescu[1], Ngoc Phuoc An Vo[2], Vadim Sheinin[3]
IBM Research, Yorktown Heights, US
[1]o.popescu@us.ibm.com, [2]ngoc.phuoc.an.vo@ibm.com, [3]vadims@us.ibm.com

Abstract

Paraphrases play an important role in natural language understanding, especially because there are fluent jumps between hidden paraphrases in a text. For example, even to get to the meaning of a simple dialog like *I bought a computer. How much did the computer cost?* involves quite a few steps. A computational system may actually have a huge problem in linking the two sentences as their connection is not overtly present in the text. However, it becomes easier if it has access to the following paraphrases: [HUMAN] buy [ARTIFACT] $\Longleftrightarrow$ [HUMAN] pay [PRICE] for [ARTIFACT] $\Longleftrightarrow$ [ARTIFACT] cost [HUMAN] [PRICE], and also to the information that *I* **IsA** [HUMAN] and *computer* **IsA** [ARTIFACT]. In this paper we introduce a resource of such paraphrases that was extracted by investigating large corpora in an unsupervised manner. The resource contains tens of thousands of such pairs and it is available for academic purposes.

Keywords: verbal paraphrase, deep learning.

1. Introduction

When two phrases can be interchanged in a text without altering the meaning of the whole, we observe a paraphrasing relationship. Paraphrasing is a fundamental property of natural languages, and it is normally recognized as "saying the same thing with different words". Proposing a paraphrasing relation is a hard task for natural language processing (NLP) systems. The task of recognition of paraphrases was proposed in various SemEval competitions (Butnariu et al., 2009; Mihalcea et al., 2010; Specia et al., 2012; Xu et al., 2015) as an independent task or as a part of larger tasks like semantic similarity, textual entailment, etc. The resource we created is instrumental for all these tasks. Some paraphrases are made out of nominal phrases that contain only ostensible nouns and their adjectival determiners, like in *a fourteen year old boy* $\Longleftrightarrow$ *a teenager*. Another type of paraphrases are the ones that involve a verbal constituent, like *abandon a kid* $\Longleftrightarrow$ *ignore parental obligations*. This later type includes nominalizations, that is, even if the phrase has only noun constituents, at least one of them is a noun coming from a verb, like *abandoning a kid*. One major difference between nominal vs. verbal paraphrases is that the first ones are basically context independent, that is they can be substituted in a text directly, while the second are context dependent, their replacement in a sentence requires changes in the syntactic and semantic role of their complements and adjuncts.

The resource we compiled contains a list of pairs, each member being centered on a verb and its arguments. A pair is a valid verbal paraphrase relation given a certain context that is represented via types associated with each argument. For a given sentence that contains only one verb phrase, we can extract a set of paraphrases. For example, in Figure 1, for the sentence *I pay 1,200 for a laptop from Bestbuy*, we present a few valid verbal paraphrase extracted from the resource. In this example *X, Y, Z, U* are variable representing the head of syntactic components. [MONEY], [HUMAN], [ORG], [ARTIFACT] represent features that the variable must carry in order for a pair to be a valid paraphrase.

The verbal paraphrases occurring in this resource are patterns of verbal phrases, that is, they represent a generalization over various real instances in a text. The names of the features occurring on different syntactic slots in a paraphrase pattern are unimportant, but the class of the words that define the respective feature is important.

2. Related Work

One of the main ideas for the acquisition and recognition of verbal paraphrases was introduced in a seminal paper (Lin and Pantel, 2001). At the core of this approach lies the fact that paraphrases occur in the same context. A statistical approach based on the mutual information measure can filter out pairs of paraphrases from a given corpora.

However, this approach cannot solve two important problems: first, it is not the words by themselves that make two expressions paraphrases but, it is actually the role these words play inside the whole sentence; second it is not clear how the complements and adjuncts are aligned between the pairs. Even if a word is very frequent, like *I* or *you*, it is the feature [HUMAN] carried by both that is actually relevant for the meaning of the verbal phrases. The second problem is very challenging, as the same type of constituent can appear in different syntactic positions in the two expressions. For example, the adjunct [SHOP] in "[HUMAN] pay [PRICE] for [ARTIFACT] from [SHOP]" must appear in the subject position in "[SHOP] sell [ARTIFACT] to [HUMAN] for [PRICE]". In this paper we describe a technique able to cope with these problems which lead to the building of a resource of pattern paraphrases.

3. Pattern Paraphrases

The technique to extract pattern paraphrases is driven by the idea behind chain clarifying relationships (see among others (Popescu and Magnini, 2007; Kawara et al., 2014; Popescu, 2013; Popescu et al., 2014)). A chain clarifying relationship holds between the components of a verbal phrase if there is a unique combinations of senses that is legitimate. For example, in *I saw the river's bank.* vs. *I saw*

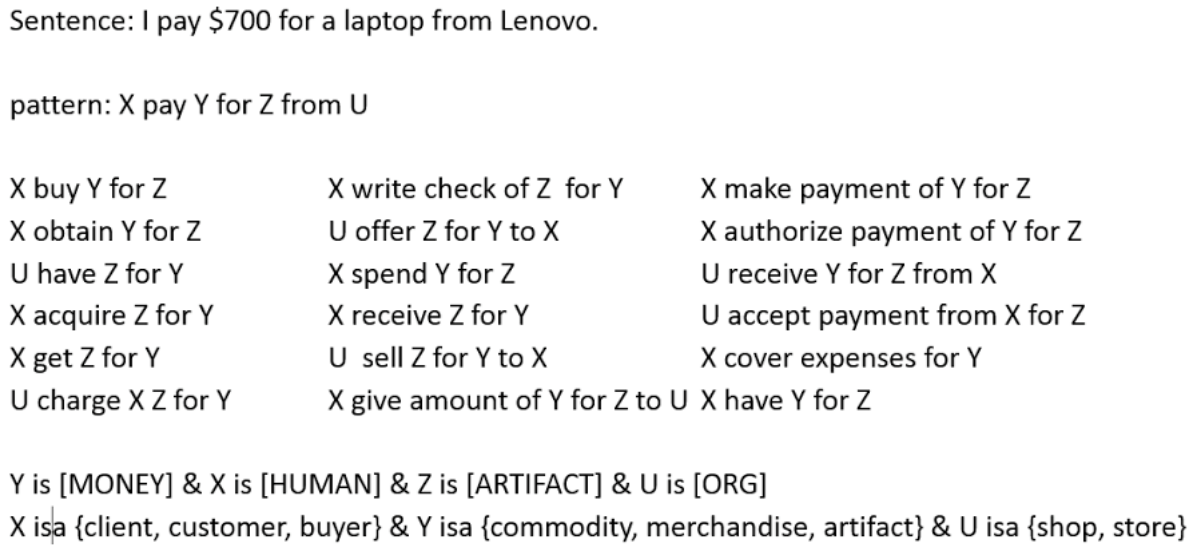


Figure 1: Examples of verbal paraphrases.

a problem the verb *see* has two different meaning, *perceive by sight* vs. *to understand*. Also, *bank* has two meanings too, *sloping land* vs. *financial institution*, and *problem* has two meanings as well: *state of difficulty*, *question raised*. The combination of senses *perceive by sight a state of difficulty* is not legitimate, and neither *understand a financial institutions* is. In fact, in the sentence *I saw the river bank*, *river* imposes the *sloping land* reading to *bank*, which in turn imposes the *perceive by light* meaning on the verb *see*. That is why we talk about a chain clarifying relationship - words trigger the sense of other words in a chain like relation, as long as the words are components of phrases that have only one combination of senses possible.

The chain clarification relation is not defined by words, which are just instances of lexical units bearing certain features. In the example above, any word which is defined by the [PHYSICAL OBJECT] feature imposes the meaning *perceive by light* to the verb *see*. From this point of view, both *apple* and *book* have the same role, as both are carrying the [PHYSICAL OBJECT] feature. However, this similarity is restricted to the chain clarifying relationship for the verb *see*. While *apple* and *book* are antagonistic with respect to the verb *devour* as they impose two different chain clarifications for this verb, namely *eat* vs. *read avidly*.

Pattern paraphrases are pairs of chain clarifying relationships. The meaning of the whole verbal phrase is preserved, thereby creating a paraphrase relationship, by the fact that the same meaning of the verb and the same features are used.

4. Extracting Pattern Paraphrases from Large Corpora

4.1. Extracting Sub-categorization Framework for Verbal Phrases

The first step in the unsupervised extraction of pattern paraphrases is to consider a large corpus that is already parsed. We used Gigaword, LDC2012T21 (Napoles et al., 2012). For each verb, we extracted the verbal dependents. Due to parser errors, there are many such dependency paths that are noise. To filter them out we used COMLEX, http://nlp.cs.nyu.edu/comlex/, (Grishman et al., 1994). In the case where the direct object was governing a prepositional phrase, this prepositional phrase was included in the dependency path. In Figure 2 we see an example of such dependencies:

the *nsubj, dobj, iobj, prep_** is the head word of the nominal group having the respective role in the dependency path, $v_$ marks the verb. As can be seen in this example, we also considered the partial paths, so the same sentence may lead to several instances of paths.

We further removed low frequency verbs, low frequency paths so that from an initial 1, 244, 793, 787 paths we filtered out a large number of them and we arrived to 391, 410, 259 paths that represent the closest approximation to a verb sub-categorization frame we could get. These paths contain 7, 922, 730 nouns in different syntactic positions and 25,812 verbs, which lead to 487,703 verbal phrases. These paths represent the input to a feed forward neural network that predicts the similarity of context. In a sense, we implemented a generalization of the original Lin algorithm that finds the dependencies paths that have the most similar context. From another point of view, we could think of the model created by this NN as dependency paths embedding. See Figure 3. e focused primarily on verbal groups, where a verbal group is defined by the following regular expression over dependency paths: $[sbj] + [obj|objprep]+[iobj]+[prepP|]*[prt]+v$ where `sbj` is the subject, obj is the object, obj prep matches the object and its governed preposition, if any, `iobj` is the indirect object, `prepP` is the attached prepositional group with its head, `prt` matches particles. For example, the following fragments of the dependency paths are matched by the above regular expression: $putprt_upwith$, $putobj_questionon$, $john_sbjwalkto_store$, $leaf_sbjtouchhim_objonface$.

The obtained model cannot be used directly to predict paraphrasing, but its output represents a large list of candidates.

nsubj_army v_strengthen dobj_hold_hold
nsubj_army v_strengthen dobj_hold_hold prep_on_city_city
nsubj_army v_strengthen prep_on_city_city
v_make dobj_way_way
v_make dobj_way_way prep_for_neighborhoods_neighborhood
v_make prep_for_neighborhoods_neighborhood
nsubj_he_he v_support dobj_seizure_seizure
v_support dobj_seizure_seizure
nsubj_ministers_minister v_say
nsubj_rebels_rebel v_rebuff dobj_efforts_effort

Figure 2: Example of dependency path extracted from GigaWord.

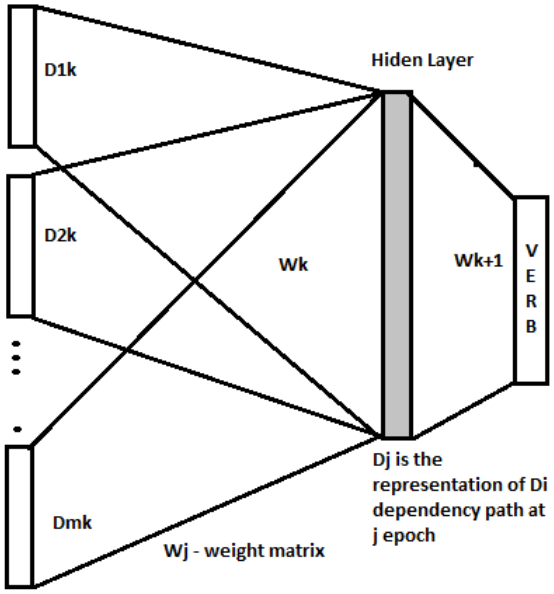


Figure 3: Dependency paths RNN.

The number of candidate pairs is 193, 628, 633. However, most of these pairs do not make it after the next filtering step.

4.2. Boostrapping from Mono-sense Verbal Phrases

In order to find the pattern paraphrases we need to find classes of words that are common between two candidates. However, due to the noise, we cannot get an accurate system of classes. Rather, we implemented a bootstrapping approach. For this, we used Zipf's law: the ambiguity of words is inversely proportional to its frequency rank. We started from verbs that according to WordNet are non ambiguous, therefore they have just one sense. We also considered linking verbs, *make, get, have, be etc.* together with their direct object and the propositional group, like *make way for* in Figure 4. These are mono-sense expressions. Then, we considered only the high probability paraphrases for these, which contain the more ambiguous verbs. We keep in separate classes the verbs from the later category, those multi-sense expressions, according to the mono-sense verb they paraphrase. If a multi-sense verbal expression occurs with two different mono-senses in a paraphrase relationship, then it is discarded. This bootstrapping process continues till we reach the most ambiguous verbs. In Figure 4 we show an example of the bootstrapping process. The *make way for* is a mono-sense verbal phrase, unlike *create* or *pave*. But the fact that at step 1 we determined that *make way for* and *create, pave* are valid candidates for paraphrasing leads to the creation of a cluster inside all the occurrences of *create*, and a cluster inside *pave*. The same happens for very ambiguous verbs like create or accommodate. All the occurrences inside this cluster can be paraphrased via MAKE WAY FOR. At the next step we will compute a precise contextual definition of these clusters.

4.3. Finding the Set of Features

The best way to find a set of features would be to have the agentive subject for each verb, like *buyer* for *buy*, with its preferred adjuncts in the set of paraphrases extracted from corpus. However, this kind of constructions are hardly present in a news corpus, as a sentence like *buyer buys*

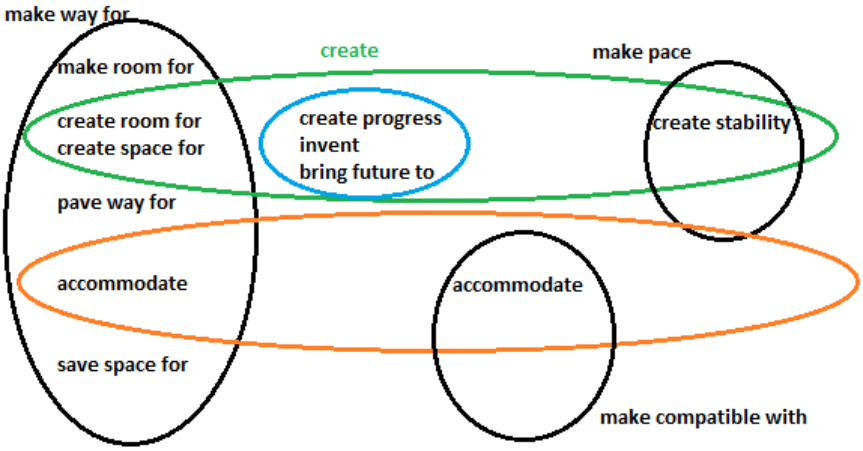


Figure 4: Bootstrapping from mono-sense verbs towards ambiguous verbs.

products is never used. We need to build the features for representing the pattern paraphrases in a bottom up approach, that is by finding the most general words that individualize that cluster vs all other clusters. In order to find the set of features for each verb separately we start from the clusters found at the previous step. Ideally, each cluster corresponds to a distinct sense of the verb. Inside each cluster, by considering the set of respective paraphrases, we compute the mutual information for each syntactic slot together with the word occurring in that syntactic slot, and rank them. At the top of this ranking we find the best representative words for that meaning, together with their syntactic functions. in the case of agentive verbs, we compute the Lin distance (Lin, 1998) on Wordnet(Miller, 1995) between it and the set of words occurring in that syntactic position, and we select the closest ones, for example *nsubj_customer , nsubj_client, nsubj_buyer* are the winners for verb *buy*. So, like in Figure 1, the paradigmatic set for variable **X** denoting the subject position is formed by these. Now, on the basis of the mutual information computed above, we find the most likely complements and their closest neighbors according to the Lin measure. The next step is to generalize the most likely fillers of verbal phrases as much as possible. This was carried out using the hypernym function from WordNet via SUMO ontology (Niles and Pease, 2003). Each word is replaced by its direct hypernym as long as the newly created form is not found in two clusters. In Figure 5 we present schematically this generalization process for three classes for the verb *move*. The three cluster identified at the previous steps, C1, C2 and C3 have different fillers for subject and object position respectively. The process of feature generalization is carried out as long as the obtained form of the pattern stay within the original cluster, that is there is no form that exist in two clusters at any time. For example , the first cluster and the third cluster in Figure 5 collide on object position, so the generalization this syntactic position stops shortly, while for for subject position it can go on further.

4.4. Seeds - Mono sense and frequent

The first observation is that the set of paraphrases generated by the Lin algorithm with class embedding is very accurate when the ambiguity of the target word is low and the number of occurrences is high. In this case the noise in classification is as low as it could be and thus the class context describes precisely the correct usage of the target word. The second observation is that class embedding preserves the meaning of the verbal group, so the semantic similarity between the set of correct paraphrases must be very high. The third observation is that the senses of the verbal group are paraphrased differently by using class embedding and thus a void intersection of class embedding indicates that the set of candidate paraphrases are indeed correct paraphrases. These observations suggest the following post filtering over the paraphrase candidate strategy:

- S1 Identify low ambiguity, high frequency verbal groups
- S2 Consider their set of candidate paraphrase;
- 3 Find the subset that minimizes the semantic distance
- S4 Consider the candidate paraphrase for each verbal group in this subset
- S5 Retain only the paraphrases that are common in these candidate sets
- S6 Repeat step 1 for the verbal groups in the retained paraphrase until the semantic distance is below a fixed threshold.

At S1 we used WordNet to decide on the ambiguity, and we used a linear combination of Lin distance with Roget similarity at step 3. The algorithm above produces a repository of paraphrases for each verbal groups. We obtained highly accurate paraphrases for 75,000 verbal groups, each verbal group being paraphrased in average with more than 250 paraphrases.

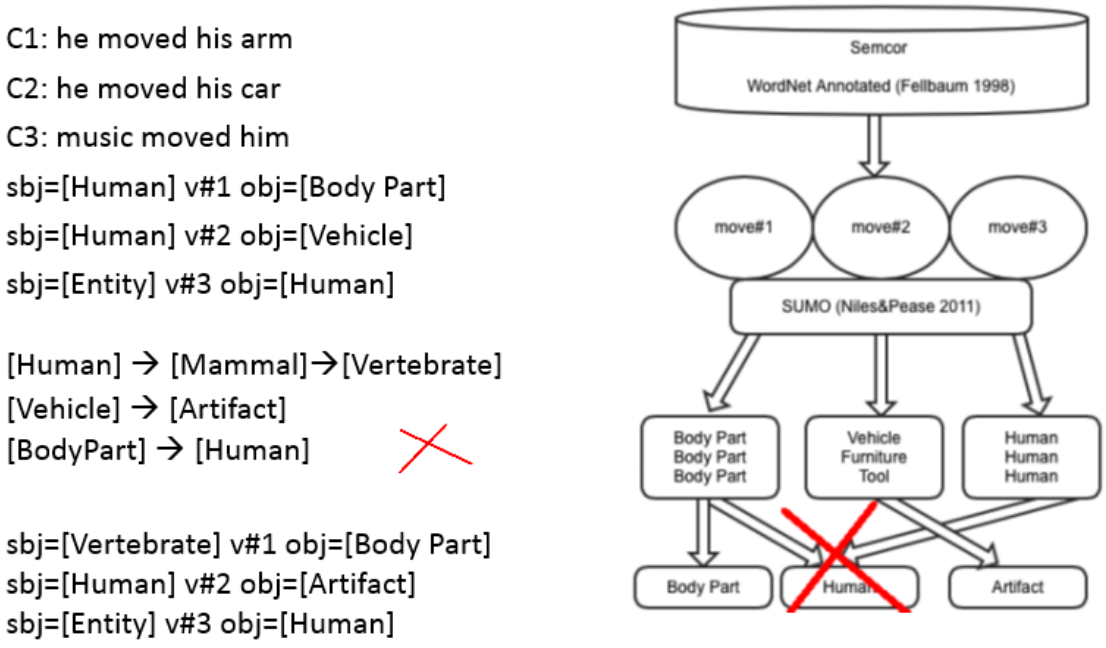


Figure 5: Generalization of features.

4.5. Slot Alignment between Paraphrases

The slot alignment is carried out via a computation of the most probable combination of arguments. This computation takes place in two steps. At the first step we consider the maximally probable configuration for each pair of paraphrases and at the second step we choose from this set the one that is the most probable considering all possible paraphrases into a cluster. Let's consider again Figure 1.

After the generalizing the slots, we have the pairs of the verbal group only, that is, we know that *buy, obtain, make a payment for, sell, get from, pay* etc. can enter a paraphrasing relationship in the same class for the verb *buy* (step 1&2 above) and we also know that this class has [CUSTOMER], [ARTIFACT], [ORG] as features for subject, direct object and prepositional group for the verb buy (step 3). The verbs *obtain, make payment for, sell, get from, pay* have their own syntactic slots for slightly different features, as the generalization process does not necessarily lead to the same features, but to variants of them, for example *client* vs. *customer*, or the same feature occurs in several slots.

First we employ a chain conditional formula for each pair of paraphrases in order to get the first one-to-one alignment. Given the form of one verbal phrase, we compute the probability that another verbal phrase has a certain realization. For example, we compute the probability that the verb *sell* has a certain configuration as $p(nsubj = [human_1], dobj = [artifact], prepTO = [human_2] \mid v = buy, nsubj = [human_2], dobj = [artifact], prepFROM = [human_1])$ (we use indexes to distinguish same feature in different syntactic position). In general, given two paraphrases in the same cluster, with $_t$ denoting the target slots, we compute $argmax_{X_t,Y_t,Z_t} p(X_t, Y_t, Z_t)$ given the distribution of $X_c, Y_c, Z_c, v_c)$ of source pattern and v_t, V_c the target and source verbs respectively. For this probability we use the chain formula and we calculate the necessary independent probabilities over the whole corpus.

$$p(X_t, Y_t, Z_t \mid v_t, v_c, X_c, Y_c, Z_c) \approx \quad (1)$$
$$p(v_t \mid v_c, X_C, Y_c, Z_c)* \quad (2)$$
$$p(X_t \mid v_c, v_t, X_C, Y_C, Z_C)* \quad (3)$$
$$p(Y_t \mid X_t, v_t, , v_t, X_C, Y_C, Z_C)* \quad (4)$$
$$p(Z_t \mid Y_t, Z_t, v_c, , v_t, X_C, Y_C, Z_C) \quad (5)$$

(2) is the probability of v_t and v_c being paraphrase relation when the words of $-c$ are used, (3),(4) and (5) represent the probability of each slot for v_t for a given word, given that the v_t and v_c are in the same cluster of paraphrases.

An example of clusters: Cluster 1 X buys Y from Z vs. Z sell Y to X X make payment of W for Y at Z vs. Y cost W at Z X in[PER], Y in [ARTIFACT], Z in [ORG], W in [MONEY] Cluster 2 X provide assistance for Y vs. X deal with Y vs. X attend over Y, Y in [ACTIVITY] Cluster 3 X acquire Y vs. X become affected with Y , X in [PERS], Y in disease Cluster 4 X acquire Y vs. Y work for X , X in [ORG], Y in [PERS]

5. Evaluation Experiments

To evaluate paraphrases is a very difficult task, because there is not a gold standard. For limited data, human experts can verify manually the validity of some of them. However, our approach, PP, produces hundreds of different paraphrases for each verbal phrase.

We selected a set of 100 verbal expressions among which *abandon, be expert on, begin, buy, employ, expect, have address on, manage, plan , produce, solve work on, write* etc. We implemented the original DIRT algorithm (Lin and Pantel, 2001) and ran it for these 100 verbs over GigaWord, call it D100_G. We also considered word2vec with

	min	max	average
D100_G	40	80	70
S_w2V	30	90	76
G_w2V	30	90	85
PPDB	60	90	79
PP	70	100	90

Table 1: Precision s-level

	min	max	average
D100_G	24	33	27
S_w2V	37	62	58
G_w2V	37	68	59
PPDB	40	70	71
PP	60	100	82

Table 2: Precision m-level

	Precision			Recall		
	min	max	average	min	max	average
D100_G	8	12	8	8	42	20
S_w2v	15	25	17	49	73	58
G_W2V	15	27	19	49	77	60
PPDB	10	68	55	60	79	67
PP	40	78	68	69	94	79

Table 3: Precision l-level

	interagremment
s level	97
m level	93
l level	85

Table 4: annotator inter-agreement

the standard Google news model, call it S_w2v, and trained form GigaWord, call it G_w2v. Finally, we considered the set of paraphrases from PPDB (Ganitkevitch et al., 2013). The PPDB has a few levels of accuracy, s which very precise, small coverage, m, the medium precision and coverage, and l that is the large coverage, lower precision. As in PPDB, there are instances of paraphrases at the sentence level, we extracted 100 sentences from GigaWord for each verbal phrases, for a total of 10,000 sentences. We carried out two evaluation experiments. The first one focuses on pairs of verbal paraphrases, without considering any context. The second one considers the context around the verbal phrases in a given sentence and proposes a new paraphrase , if available. This second experiment cannot be carried out for DIRT, or w2v because these approach do not handle the context, so the systems evaluated here are ours, PP, and PPDB.

5.1. Pair to Pair paraphrase evaluation

For the 100 chosen verbs, we put together all the paraphrases created by each approach. For the DIRT and word2vec approaches we have to set a threshold under which two pairs are not consider paraphrases, as these approaches compute a score for each possible pair. We consider the first 10, 40 and 400 pairs, which create three levels of precision which we roughly equate with the s,m,l levels from PPDB. These thresholds were not exactly a random choice, because 10 is the average number existing in VerbOcean (Chklovski and Pantel, 2004), a paraphrase resource created with DIRT algorithm, while 40 is the standard number of similar phrases returned by word2vec. We also ranked the PP created by our approach based on the probability of occurrence of each pattern. In this way , we could have the same levels of 10, 40 and 400 paraphrases. So we create three distinct test corpora where each verb had the first 10, 40 and 400 returns from our approach, DIRT, word2vec and PPDB, respectively. There are not exactly 4000, 16 000, and 160 000 pairs of paraphrases, as some of the above resources may not have provided the required number of paraphrases. In the end we have three test corpora for the s,m,l level. Our experiment consists in extracting random samples from each of the test corpus and in evaluating their accuracy. We can now estimate how many pairs were correct on average for each approach, and how many correct paraphrases were contributed to the pool of correct paraphrases by each approach. We have three annotators, each one checking 2,500 pairs for correctness, out of which 250 were from the s level, 750 from the m level and 1,500 from the l level . Out of these 7, 500 pairs, 900 where common to all three annotators in order to compute their inter-agreement. The first three tables belows summarize the results of the evaluation for each of the s, m, l levels, and the fourth one shows the inter-agreement percentage.

5.2. Contextual Paraphrasing

There are 10,000 sentences that contain the chosen verbs that we extracted from Gigaword. For this sentences we can compare the accuracy of the whole text, not only of the verb. That is we can compare the effectiveness of paraphrase replacement in a specified context. Only our approach and PPDB can be compared in this experiment, as for DIRT and word2vec there is no immediate way to carry it out as this approaches do not contain contextual information. We considered the large level in order to maximize the chance that a given sentence matches an existing paraphrase in PPDB. From the 10,000 sentences we selected a random sample of 1,500 sentences and we gave them to the same three annotators, that is 500 for each. 90 sentences were common to all three annotators in order to observe their inter-agreement. The pp approach produced the correct replacement in the sentences in 46% of the sentences, while a suitable paraphrase was found in ppdb only in 19%. The inter agreement was 76%.

6. Conclusion and Further Work

We have compiled in a unsupervised way a large resource of pattern paraphrases that is available for academic purposes. A pattern is defined by a verb and a set of features that can occur on a specified syntactic position. A pattern matches some constituents in a given sentence by instantiating the features with corresponding words. The pattern is paraphrased by other patterns which do not necessarily

assign the same syntactic roles to those constituents. Each pattern corresponds to a set of specific paraphrases which involve different other verbs. There are a few directions that could be exploited in order to increase the quality of this resource. For the moment, there are no paraphrases for noun phrases, including the ones that may contain adjectival determiners. This is a direction that we would like to exploit further. The adverbs were not taken into account when we extracted dependency paths, but they may play a role in the determination of certain pattern paraphrases. Another direction for improvement is to fill the gaps in the pattern set for certain verbs, that is, the algorithms acknowledges that some patterns have not been found yet, but their instances are present in text.

7. Bibliographical References

Butnariu, C., Kim, S. N., Nakov, P., Ó Séaghdha, D., Szpakowicz, S., and Veale, T. (2009). Semeval-2010 task 9: The interpretation of noun compounds using paraphrasing verbs and prepositions. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 100–105. Association for Computational Linguistics.

Chklovski, T. and Pantel, P. (2004). Verbocean: Mining the web for fine-grained semantic verb relations. In *EMNLP*, volume 4, pages 33–40.

Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2013). Ppdb: The paraphrase database. In *HLT-NAACL*, pages 758–764.

Grishman, R., Macleod, C., and Meyers, A. (1994). Comlex syntax: Building a computational lexicon. In *Proceedings of the 15th conference on Computational linguistics-Volume 1*, pages 268–272. Association for Computational Linguistics.

Kawara, D., Pederson, D., Popescu, O., and Palmer, M. (2014). Inducing example-based semantic frames from a massive amount of verb uses. In *Proceedings of the EACL 2014*. Association for Computational Linguistics.

Lin, D. and Pantel, P. (2001). Dirt@ sbt@ discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328. ACM.

Lin, D. (1998). An information-theoretic definition of similarity. In *ICML*, volume 98, pages 296–304.

Mihalcea, R., Sinha, R., and McCarthy, D. (2010). Semeval-2010 task 2: Cross-lingual lexical substitution. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 9–14. Association for Computational Linguistics.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Napoles, C., Gormley, M., and Van Durme, B. (2012). Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 95–100. Association for Computational Linguistics.

Niles, I. and Pease, A. (2003). Mapping wordnet to the sumo ontology. In *Proceedings of the ieee international knowledge engineering conference*, pages 23–26.

Popescu, O. and Magnini, B. (2007). Sense discriminative patterns for word sense disambiguation. In *SCAR workshop, NODALIDA*.

Popescu, O., Palmer, M., and Hacks, P. (2014). Mapping cpa onto ontonotes. In *Proceedings of the 9th International Conference on Language Resources and Evaluation - LREC14*.

Popescu, O. (2013). Learning corpus pattern with finite state automata. In *Proceedings of the ICSC 2013*.

Specia, L., Jauhar, S. K., and Mihalcea, R. (2012). Semeval-2012 task 1: English lexical simplification. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 347–355. Association for Computational Linguistics.

Xu, W., Callison-Burch, C., and Dolan, B. (2015). Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *SemEval@ NAACL-HLT*, pages 1–11.

Building Parallel Monolingual Gan Chinese Dialects Corpus

Fan XU, Mingwen WANG, Maoxi LI
School of Computer Information Engineering, Jiangxi Normal University
Nanchang 330022, China
{xufan, mwwang, mosesli}@jxnu.edu.cn

Abstract

Automatic language identification of an input sentence or a text written in similar languages, varieties or dialects is an important task in natural language processing. In this paper, we propose a scheme to represent Gan (Jiangxi province of China) Chinese dialects. In particular, it is a two-level and fine-grained representation using Chinese character, Chinese Pinyin and Chinese audio forms. Guided by the scheme, we manually annotate a Gan Chinese Dialects Corpus (GCDC) including 131.5 hours and 310 documents with 6 different genres, containing news, official document, story, prose, poet, letter and speech, from 19 different Gan regions. In addition, the preliminary evaluation on 2-way, 7-way and 20-way sentence-level Gan Chinese Dialects Identification (GCDI) justifies the appropriateness of the scheme to Gan Chinese dialects analysis and the usefulness of our manually annotated GCDC.

Keywords: Parallel Corpus, Monolingual, Gan Chinese Dialects

1. Introduction

Automatic language identification of an input sentence or a document is an important task in Natural Language Processing (NLP), especially when processing speech or social media messages. Currently, the interest in language resources and its computational models for the study of similar languages, varieties and dialects has been growing substantially in the last few years (Zampieri et al, 2014, 2015, 2017; Malmasi et al., 2016). Meanwhile, an increasing number of dialect corpus and corresponding computational models have been released for Catalan, Russian, Slovene, etc. However, no free corpus has been released for the similar, varieties or dialects in Mandarin Chinese. Therefore, in this paper, we focus on the corpus building and its computational model design for the closely related parallel monolingual Gan Chinese languages.

As we all know, Chinese is spoken in different regions, with distinct differences among regions. There are different expressions for a same concept among the closely related Gan Chinese languages, varieties and dialects. For example, 今里 jin li 'today', 今家 jin jia 'today', 今宁 jin ning 'today', 今兜 jin dou 'today', 今朝 jin zhao 'today' are the valid expressions in Nanchang, Yichun, Jian, Fuzhou and Yingtan district in Jiangxi province (Gan in short) of China, respectively. Although these expressions are different, they have the same semantic meanings. They all refer to 今天 jin tian 'today' in Mandarin Chinese (called 普通话 Putonghua 'common language' in Mainland China).

More specifically, firstly, we present a scheme to handle Gan Chinese dialects which is a fine-grained representation using Chinese character, Chinese Pinyin and Chinese audio forms. Based on the scheme, we manually annotate a parallel Gan Chinese Dialects Corpus (GCDC) consists of 310 documents with 6 different genres (news, official document, story, prose, poet, letter and speech) from 19 different Gan regions. As a byproduct, the corpus contains the parallel Gan Chinese audio with 131.5 hours. Besides, we conduct a preliminary experiment on the proposed GCDC through the sentence-level Gan Chinese Dialects Identification (GCDI) task. The simple but effective character Chinese Pinyin uni-gram yields a strong baseline on 2-way, 7-way and 20-way Gan dialects discrimination. The overall accuracy can reach to 78.64% on the fine-grained 20-way classification, which shows the automatic Gan Chinese dialects identification should be feasible. The evaluation result justifies the appropriateness of the scheme to Gan Chinese dialects analysis and the usefulness of our manually annotated GCDC.

The rest of this paper is organized as follows. Section 2 overviews related work. In Section 3, we present the scheme to deal with Gan Chinese dialects. Section 4 describes the annotation and an annotation instance of the GCDC. In Section 5, we present our preliminary experiment for the sentence-level Gan Chinese dialects identification on the proposed GCDC, and we conclude this work in Section 6 and present future directions.

2. Related Work

In this section, we describe the representative dialect corpus and its corresponding discrimination models.

2.1 Parallel Corpus

In the past decade, several parallel corpora among different languages have been proposed, e.g. Chinese-English (Ayan and Dorr, 2006), Japanese-English (Takezawa et al., 2002) and French-English (Mihalcea and Pedersen, 2003). They are annotated either at word-level or phrase-level alignment between two different languages (bilingual). Recently, many researchers pay attention to the parallel corpora only in the closely related languages (monolingual), varieties and dialects (Zampieri et al, 2014, 2015, 2017; Malmasi et al., 2016) which containing Bulgarian, Macedonian, etc. and a group containing texts written in a set of other languages. However, none parallel corpora in the closely related languages in the Gan dialects has been freely released so far. The representative certain scale parallel corpora is the Greater China Region (GCR) corpora (Xu et al., 2015) which focus on Mandarin with simplified and traditional scripts.

2.2 Dialects Identification Models

Generally speaking, language identification among different languages is a task that can be solved at a high accuracy. For example, Simoes et al. (2014) achieved 97% accuracy for discriminating among 25 unrelated languages. However, it is generally difficult to distinguish between related languages or variations of a specific language (see Zampieri et al, 2014, 2015 for example). To

be more specific, Ranaivo-Malancon (2006) proposed features based on frequencies of character n-grams to identify Malay and Indonesian. Zampieri and Gebre (2012) found that word uni-grams gave very similar performance to character n-gram features in the framework of the probabilistic language model for the Brazilian and European Portuguese language discrimination. Tiedemann and Ljubesic (2012); Ljubesic and Kranjcic (2015) showed that the Na ̈ve Bayes classifier with uni-grams achieved high accuracy for the South Slavic languages identification. Grefenstette (1995); Lui and Cook (2013) found that bag-of-words features outperformed the syntax or character sequences-based features for the English varieties. Besides these works, other recent studies include: Spanish varieties identification (2014), Arabic varieties discrimination (Elfardy and Diab,2013; Zaidan and Callison-burch, 2014; Salloum et al.,2016; Tillmann et al.,2014) and Persian and Dari identification (Malmasi and Dras, 2015); Indian languages identification (Murthy and Kumar, 2006).

3. Annotation Scheme

In this section, we present the scheme to Gan Chinese dialects which has two-level partitions and three forms.

3.1 Two-level Partitions

Gan Chinese is spoken in different regions in Jiangxi province of China, with distinct differences between two regions. To be specific, Table 2 shows a two-level Gan dialects partition is provided. The first level contains six regions of Gan dialects (Yan Sen, 1986), such as 昌靖片 'chang jing region', 宜萍片 'yi ping region', 吉莲片 'ji lian region', 抚广片 'fu guang region', 鹰弋片 'yin yi region', 客家话 'Hakka'. The six regions are further divided into 19 sub-regions in the second level. For example, 昌靖片 'chang jing region' contains 5 sub-regions, such as 新建 'xinjian', 南昌 'nanchang' and so on.

3.2 Three Forms

Chinese Pinyin: Pinyin is basically the alphabet for the Chinese language. The Pinyin system was invented to help people pronounce the sound of the Chinese characters. It is a Romanization system used to learn Mandarin. It transcribes the sounds of Mandarin using the Western (Roman) alphabet. It is well know that pronunciation is vital for any language. Therefore, we annotate Chinese Pinyin into our corpus. Figure 1 and Table 1 show pitch contours of lexical tones in Mandarin (Chen et al., 2016). In our corpus annotation, we annotate the pitch height with 1, 2, 3 and 4 accordingly.

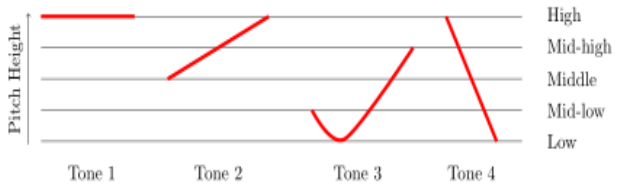


Figure 1: Pitch contours of lexical tones in Mandarin. The vertical axis denotes pitch height, whereas the horizontal staves indicate different tone levels within one's comfortable vocal range.

Tone	Pitch Contour	English Equivalent
1	High-level	Singing
2	High-rising	Question-final intonation; e.g., What?
3	Dipping	No equivalent; e.g., nǐhǎo, hello
4	Falling	Curt commands; e.g., Stop!

Table 1: Lexical Tones in Mandarin.

Chinese character: We observe that the same concept can be expressed using different linguistic expressions for the different region of Gan dialects as mentioned in the Introduction section.

Chinese audio: Furthermore, we present Chinese audio as a byproduct in this corpus. It consists of the parallel Gan Chinese audio and mandarin Chinese sound for each document.

4. Gan Chinese Dialects Corpus

In this section, we address the key issues with the GCDC annotation.

4.1 Annotator Training

The annotator team consists of a Ph.D. in Chinese linguistics as the supervisor (senior annotator) and 19 undergraduate students from different 19 Gan regions in Chinese linguistics as annotators. An annotator of a given region works only in data of his/her area. The annotation is done in four phases. In the first phase, the annotators spend 1 month on learning the principles of scheme. In the second phase, the annotators spend 1 month on independently annotating the same 30 documents, and another 1 month on crosschecking to resolve the difference and to revise the guidelines. In the third phase, the annotators spend 2 months on annotating the remaining 280 documents. In the final phase, the supervisor spends 1 month carefully proofread all 310 documents.

4.2 Corpus Statistics

Currently, the GCDC corpus consists of the representative 19 sub-regions of Gan dialects and their statistics as shown in Table 2. Given the above scheme, we annotate parallel 310 XML-style documents with 6 different genres (news, official document, story, prose, poet, letter and speech), containing 218 newswire documents from Chinese Treebank 6.0 with Linguistic Data Consortium (LDC) catalog number LDC2007T36, and other 92 documents for the remaining genres from the internet using Baidu search engine with official document, story, prose, poet, letter and speech as key words. Specifically, we don't have parallel sentences for each variant of each sentence in all documents, and the documents included differ among the dialects but are all parallel with respect to a Mandarin translation. We require the annotators to annotate the documents included differ among the dialects but are all parallel with respect to a Mandarin translation. As a byproduct, it has the 131.5-hour audios, wherein 69.0 hours Gan dialects sound and 62.50 hours Putonghua sound, and the total number of sentence in the corpus is 3,878. Table 2 shows the statistics in detail with the number of non-news genre are shown in parentheses.

Dialects region (level-1)	Dialects location (level-2)	Number of document	Number of sentence
昌靖片 chang jing region	新建 xinjian	10(4)	-
	南昌 nanchang	10(3)	-
	湖口 hukou	10(2)	-
	都昌 douchang	6(6)	-
	靖安 jingan	16(0)	-
	#total	52(15)	353(101)
吉莲片 ji lian region	吉安 jian	10(2)	-
	吉水 jishui	8(6)	-
	永丰 yongfeng	21(12)	-
	#total	39(20)	362(138)
抚广片 fu guang region	进贤 jingxian	10(2)	-
	东乡 dongxiang	14(2)	-
	抚州 fuzhou	17(12)	-
	#total	41(16)	230(105)
宜萍片 yi ping region	丰城 fengcheng	10(4)	-
	宜丰 yifeng	5(3)	-
	萍乡 pingxiang	7(5)	-
	#total	22(12)	165(99)
鹰弋片 yin yi region	余干 Yugan	10(4)	-
	上饶 shangrao	8(8)	-
	#total	18(12)	110(63)
客家话 Hakka	赣州 ganzhou	24(12)	-
	兴国 xinguo	12(2)	-
	大余 dayu	10(3)	-
	#total	46(17)	294(120)
普通话 Putonghua		156(92)	1113(625)
#total		310	3878

Table 2: Corpus statistics.

4.3 Quality Assurance

It is very challenging to check the agreement between annotators. We focus on 鹰弋片'yin yi region', and require another 2 annotators from this region to annotate 30 documents, 173 sentences, from the corpus. We calculate the annotation consistency value which is 0.93 for Pinyin. Due to the homophone phenomenon is obvious in Chinese character, we don't calculate the agreement for Chinese character. The high inter-annotator consistency in Chinese Pinyin guarantees the corpus's quality.

4.4 An Annotation Instance

Table 3 describes an annotation instance of GCDC for clarity.

```
<?xml version="1.0" encoding="GB2312" ?>
<document>
<voiceInfo>
  <Region>昌靖片 chang jing region</Region>
  <Location>新建 xinjian </Location>
  <Sex>女 Female</Sex>
  <Age>19</Age>
  <Genre>新闻 news </Genre>
  <Chanel>手机 mobile phone</Chanel>
  <FangyanTime>38 seconds </FangyanTime>
  <PutongTime>36 seconds</PutongTime>
  <FangyanFile>chtb_2946_fangyan.wav</FangyanFile>
  <PutongFile>chtb_2946_putong.wav</PutongFile>
</voiceInfo>
<sentence count="1">
  <putongContent>据 报道 ： 星期六 印度 和 巴基斯坦 军队 ， 在 科什米尔 停火线 一带 又 发生 了 新的 冲突 。According to a report, new conflicts in Kashmir ceasefire area were occurred between India and Pakistan on Saturday.
  </putongContent>
  <ganContent>居 报倒 ： 星期六 印度 和 巴基斯坦 军队 ， 赖 科什米尔 停我线 一带 又 发生 的 新个 冲突 。According to a report, new conflicts in Kashmir ceasefire area were occurred between India and Pakistan on Saturday.
  </ganContent>
  <putongPinyin>Ju4 Bao4Dao3 : Xing1Qi2Liu4 Yin4Du4 He2 Ba1Ji1Si1Tan3 Ju1Dui4 , Zai4 Ke1Shen2Mi3Er3 Ting2Huo3Xian4 Yi2Dai4 You4 Fa1Sheng1 Le1 Xin1De1 Chong1Tu1 .
  </putongPinyin>
  <ganPinyin>Ju4 Bao4Dao3 : Xing1Qi2Liu4 YIN4Du4 He2 Ba1Ji1Si1Tan3 Ju1Dui4 , Lai4 Ke1Shen2Mi3Er3 Ting2Wo3Xian4 Yi2Dai4 You4 Fa1Sheng1 De4 Xin1Ge4 Chong1Tu1 .
  </ganPinyin>
</sentence>
<sentence count="2">
  <putongContent>巴基斯坦 方面 说 ： 最近 发生 在 平泊尔 地区 的 冲突 中 ， 有 5 名 印度 士兵 被 打死 ， 很多 士兵 被 打伤 。It was reported by Pakistan that five Indian soldiers were killed and many soldiers were wounded in recent clashes in Pingboer area.
  </putongContent>
  <ganContent>巴基斯坦 方面 挖 ： 最将 发生 赖 平泊尔 那里 个 冲突 里面 ， 有 5 个 印度 当兵个 被 打死的 ， 好多 当兵个 被 打伤的 。It was reported by Pakistan that five Indian soldiers were killed and many soldiers were wounded in recent clashes in Pingboer area.
  </ganContent>
  <putongPinyin>Ba1Ji1Si1Tan3 Fang1Mian4 Shuo1 : Zui4Jin4 Fa1Sheng1 Zai4 Ping2Bo2Er3 Di4Qu1 De1 Chong1Tu1 Zhong1 , You3 Wu3 Ming2 Yin4Du4 Shi4Bing1 Bei4 Da3Si3 , Hen3Duo1 Shi4Bing1 Bei4 Da3Shang1.
  </putongPinyin>
  <ganPinyin>Ba1Ji1Si1Tan3 Fang1Mian4 Wa1 : Zui4Jiang1 Fa1Sheng1 Lai4 Ping2Bo2Er3 Na4Li3 Ge4 Chong1Tu1 Li3Mian4 , You3 En3 Ge4 Yin4Du4 Dang11Bing1Ge4 Bei4 Da3Si3De1 , Hao3Duo1 Dang11Bing1Ge4 Bei4 Da3Shang1De1 .
  </ganPinyin>
</sentence>
</document>
```

Table 3: An annotation instance for Gan dialects.

The example comes from file chtb2946 of CTB (Chinese Tree Bank) released by the LDC. The <voiceInfo> section describes the detail information, such as region, location, sex and age of annotator, genre type, record channel, duration and file. The <sentence> section demonstrates the specific contents including Chinese character and Chinese Pinyin, containing <putongContent> section refers to the Chinese character in Mandarin, <ganContent> section represents the Chinese character in specific Gan dialect. <putongPinyin> section indicates the Chinese Pinyin in Mandarin, while <ganPinyin> section means the Chinese Pinyin in specific Gan dialect. The whole corpus are available through the LREC 2018 repository.

5. Preliminary Experimentation

As mentioned in the Introduction section, automatic language identification of an input text is an important task in Natural Language Processing (NLP) because somebody must determine the language of the text before applying tools trained on specific language. For the sentence-level language identification, a user is given a single sentence, and the user needs to identify the language. Below, we recast the sentence-level dialects identification in the Gan dialects as a multi-class classification problem. Firstly, we will describe some features. Then, these features are fed into a classifier to determine the dialect of a sentence.

5.1 Features

In this section, we represent the character-level N-gram features.

Chinese Character Pinyin N-gram: According to the related work (Nikola and Denis, 2015; Cagri and Taraka, 2016), n-grams with $n \leq 3$ are effective features for discriminating general languages. Also, Cagri and Taraka (2016) showed their simple linear SVM model with n-gram feature is quite useful and hard to beat by current neural network models. Compared with English, no space exists between words in Chinese sentence. Therefore, we use character uni-grams, bi-grams and tri-grams in Chinese Pinyin as features. We take Pinyin with lexical tones or without it as different two kinds of features.

Chinese Character N-gram: While our corpus provides both Chinese character and Chinese Pinyin simultaneously, we also present Chinese Character uni-grams, bi-grams and tri-grams as features. This is because sometime Pinyin is not available in a specific situation.

5.2 Classifier and Evaluation Metric

Classifier: After extracting the above proposed features, we train a single multi-class linear kernel support vector machine using LIBLINEAR (Fan et al., 2008) for Gan Chinese dialects identification. They adopt the default parameters such as verbosity level with 1, trade-off between training error and margin with 0.01, slack rescaling, zero/one loss.

Evaluation Metric: We report system's performance using accuracy, which is the ratio of the number of the correctly predicted sentence divided by the total number of sentence for Gan dialects.

For the Gan dialects dataset, we generate three scenarios using 5-fold cross validation:

(1) **2-way detection**: We try to distinguish between two groups of dialects, the ones is 普通话 'Putonghua', and the others are the left 19 sub-regions of Gan dialects;

(2) **7-way detection**: The level-1 Gan dialects of 昌靖片 'chang jing region', 吉莲片 'ji lian region', 抚广片 'fu guang region', 宜萍片 'yi ping region', 鹰弋片 'yin yi region', 客家话 'Hakka' and 普通话 'Putonghua' as shown in Table 2 are considered;

(3) **20-way detection**: We detect both Mandarin and other level-2 19 Gan dialects of 新建 xinjian, 南昌 nanchang, 湖口 hukou, etc. as shown in Table 2 are all considered.

5.3 Experimentation Results

In this section, we report the experiment results for the Gan Chinese dialects identification on our dataset.

5.3.1 Results on Chinese Character Pinyin

Table 4 shows the performance on Chinese character Pinyin feature. As can be seen, the character uni-gram Pinyin feature yields the best performance on both news and other types of genres. Obviously, the performance of 2-way classification is higher than both 7-way and 20-way language discrimination. Strangely, the performance of the more fine-grained 20 different dialect labels task achieves higher results than the identification of only 7 labels. We attribute it to the parallel nature of the corpus. Basically, the performance is increased with the increment of the number of training data.

Domain	Way	Lexical tones	acc. (%)
Chinese Character uni-gram Pinyin			
News genre	2-way	Y	**85.94**
		N	85.52
	7-way	Y	**73.44**
		N	72.91
	20-way	Y	**78.64**
		N	75.19
Other genres	2-way	Y	**74.04**
		N	71.61
	7-way	Y	**68.96**
		N	66.99
	20-way	Y	**69.37**
		N	68.22
Chinese Character bi-gram Pinyin			
News genre	2-way	Y	63.56
		N	68.93
	7-way	Y	50.19
		N	53.43
	20-way	Y	50.59
		N	52.63
Other genres	2-way	Y	47.40
		N	52.89
	7-way	Y	43.39
		N	46.46
	20-way	Y	40.57
		N	45.08

Table 4: The performance of sentence-level Gan dialects identification using Chinese character Pinyin. 'Y' stands for the corpus with lexical tones, 'N' indicates none.

In addition, lexical tones in uni-gram Pinyin reflect the fine-grained characteristic of Gan dialects. Using lexical

tones is better than without it. We also conduct the tri-gram case, but the performance is lower than bi-gram about 20%. Compared with uni-gram, there are much noise in both bi-gram and tri-gram features. The proposed uni-gram features significantly outperforms the bi-gram ones with $p<0.05$ using paired t-test for significance. It shows the effectiveness of the proposed Chinese character Pinyin feature.

More specifically, the accuracy for each level-1 Gan dialects for news domain with Chinese character uni-gram Pinyin feature is reported in Table 5. As shown, we gain the best identification performance for 普通话 'Putonghua', while the accuracy of 鹰弋片 'yin yi region' is the worst one. The reason is that the difference between 鹰弋片 'yin yi region' and 普通话 'Putonghua' is not obvious enough as shown in Table 6, also we have enough training data for 普通话 'Putonghua'.

Dialect	acc. (%)
昌靖片 chang jing region	68.57
抚广片 fu guang region	76.09
客家话 Hakka	67.24
吉莲片 ji lian region	80.56
普通话 Putonghua	**95.50**
宜萍片 yi ping region	84.85
鹰弋片 yin yi region	22.73

Table 5: Accuracy of each level-1 Gan dialects on news domain.

To be more specific, we report the confusion table for each level-1 Gan dialect using Chinese character uni-gram Pinyin in Table 6. As can be seen, most instances have been correctly classified. Due to the challenge of discrimination for the closely related languages in the Gan dialects, some instances still have been falsely classified. For example, we can know that some instances falsely classified from 昌靖片'chang jing region' to'普通话' Putonghua (20) is similar to those from 鹰弋片'yin yi region' to 普通话 'Putonghua' (15). The reason is that the 昌靖片 'chang jing region' and 鹰弋片 'yin yi region' are closed to 普通话 'Putonghua'.

		Predicted label						
		L_1	L_2	L_3	L_4	L_5	L_6	L_7
True label	L_1	**48**	2	0	0	20	0	0
	L_2	2	**35**	2	1	6	0	0
	L_3	0	0	**39**	0	19	0	0
	L_4	1	1	4	**58**	8	0	0
	L_5	1	2	2	4	**212**	0	1
	L_6	1	0	1	0	3	**28**	0
	L_7	1	0	1	0	**15**	0	5

Table 6: Confusion table of each level-1 Gan dialects using uni-gram Pinyin with tones on news domain. Remark: L_1 stands for 昌靖片'chang jing region', L_2 indicates 抚广片'fu guang region', L_3 donates 客家话 'Hakka', L_4 refers to 吉莲片'ji lian region', L_5 means 普通话 'Putonghua', L_6 represents 宜萍片'yi ping region', L_7 embodies 鹰弋片'yin yi region'.

5.3.2 Results on Chinese Character

Table 7 shows the performance on Chinese character. Again, the character uni-gram feature yields best performance on both news and other type of genres. It also yields promising results on the extremely difficult fine-grained 20-way language classification.

Feature	Domain	Way	acc. (%)
Character uni-gram	News genre	2-way	**89.43**
		7-way	**76.03**
		20-way	**79.70**
	Other genres	2-way	**79.08**
		7-way	**67.70**
		20-way	**67.42**
Character bi-gram	News genre	2-way	73.27
		7-way	52.66
		20-way	52.02
	Other genres	2-way	58.71
		7-way	45.91
		20-way	44.35

Table 7: The performance of sentence-level Gan dialects identification using Chinese character.

6. Conclusions

In this paper, we annotate a parallel Gan Chinese Dialects Corpus (GCDC) based on different levels of modularity (written and spoken data) with different layers of annotations and transcription. Meanwhile, we conduct a preliminary experiment on the proposed GCDC through sentence-level Gan Chinese dialects identification task on different levels of granularity. The simple but effective character Chinese Pinyin and character uni-gram yields a strong baseline, especially on the 20-way Gan dialects discrimination, which shows the fine-grained automatic Gan Chinese dialects identification should be feasible.

In future work, we would like to explore more features without the need of using the Pinyin notation, enlarge the scale of the corpus, and test other classifiers. Furthermore, we will finally investigate how dialect identification can help other NLP tasks.

7. Acknowledgments

The authors would like to thank the anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China under Grant No. 61772246, 61402208, 61462044, 61462045, 61662031 and the Social Science Planning Project of Jiangxi Province under Grant No. 17YY05 and the humanities and Social Sciences projects of the Jiangxi Provincial Education Department under Grant No. YY17211.

8. Bibliographical References

Nancy F. Chen, Darren Wee, Rong Tong, Bin Ma, Haizhou Li. (216). Large-scale characterization of non-native mandarin Chinese spoken by speakers of European origin: Analysis on iCALL. *Speech Communication*, 84:46-56.

Cagri Coltekin, taraka Rama. (2016). Discriminating similar languages : experiments with linear SVMs and neural networks. In *Proceedings of the third workshop on NLP for similar languages, varieties and dialects*, pages 15-24, Osaka, Japan, December 2016.

Heba Elfardy and Mona T Diab. (2013). Sentence level dialect identification in Arabic. *In Proceedings of ACL*, pages 456-461.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xang-Rui Wang, and Chih-Jen Lin. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9: 1871-1874.

Necip Fazil Ayan and Bonnie J Dorr. (2006). Going beyond aer: An extensive analysis of word alignments and their impact on MT. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual Meeting of the Association for Computational Linguistics*, pages 9-16.

Gregory Grefenstette. (1995). Comparing two language identification schemes. In *Proceedings of Analisi Statistica dei Dati Testuali (JADT)*, pages 263-268.

Nikola Ljubesic and Denis Kranjcic. (2015). Discriminating between closely related languages on twitter. *Informatica*, 39(1):1-8.

Marco Lui and Paul Cook. (2013). Classifying English documents by national dialect. In *Proceedings of Australasian Language Technology Workshop*, pages 5-15.

Bali Ranaivo-Malancon. (2006). Automatic identification of close languages-case study: Maly and Indonesian. *ECTI Transactions on Computer and Information Technology*, 2:126-134.

Wolfgang Maier and Carlos Gomez-Rodriguez. (2014). Language variety identification in Spanish tweets. In *Proceedings of the LT4CloseLang Workshop*, pages 25-35.

Shervin Malmasi and Mark Dras. (2015). Automatic Language Identification for Persian and Dari texts. In *Proceedings of PACLING*, pages 59-64.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubesic, Preslav Nakov, Ahmed Ali, Jorg Tiedemann. (2016). Discriminating Between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task. In *Proceedings of the COLING VarDial Workshop*, pages 1-14.

Rada Mihalcea and Ted Pedersen. (2003). An Evaluation Exercise for Word Alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1-10.

Kavi Narayana Murthy and G Bharadwaja Kumar. (2006). Language identification from small text samples. *Journal of Quantitative Linguistics*, 13(01):57-80.

Matthew Purver. (2015). A Simple Baseline for Discriminating Similar Languages. In *Proceedings of the LT4VarDial*, pages 1-5.

Wael Salloum, Heba Elfardy, Linda Alamir-Salloum, Nizar Habash, and Mona Diab. (2014). Sentence level dialect identification for machine translation system selection. In *Proceedings of ACL*, pages 772-778.

Alberto Simoes, Jose Joao Almeida, and Simon D Byers. (2014). Language identification: a neural network approach. In *Proceedings of the 3rd Symposium on Languages, Applications and Technologies, SLATE'14*, pages 252-265.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. (2002). Toward a Broad-coverage Bilingual Corpus for Speech Translation of Travel Conversations in the Real World. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC)*, pages 147-152.

Jorg Tiedemann and Nikola Ljubesic. (2012). Efficient discrimination between closely related languages. In *Proceedings of COLING*, pages 2619-2634.

Christoph Tillmann,Yaser Al-Onaizan, and Saab Mansour. (2014). Improved Sentence-Level Arabic Dialect Classification. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages*, pages 110–119.

Fan Xu, Xiongfei Xu, Mingwen Wang, Maoxi Li. (2015). Building Monolingual Word Alignment Corpus for the Greater China Region. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages*,Varieties and Dialects, pages 85–94, Hissar, Bulgaria, September 10, 2015. Association for Computational Linguistics.

Sen Yan. The Partitions of Jiangxi Dialects. *Dialect*, 1986,(1):19-38 (in Chinese).

Omar F Zaidan and Chris Callison-Burch. (2014). Arabic dialect identification. *Computational Linguistics*, 40(1): 171-202.

Marcos Zampieri and Binyam Gebrekidan Gebre. (2012). Automatic identification of language varieties: The case of Portuguese. In *Proceedings of KONVENS*, pages 233-237.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubesic, Preslav Nakov, Ahmed Ali, Jorg Tiedemann, Yves Scherrer, Noemi Aepli. (2017). Findings of the VarDial Evaluation Compaign. In *Proceedings of the EACL VarDial workshop*, pages 1-15.

Marcos Zampieri, Liling Tan, Nikola Ljubesic, and Jorg Tiedemann. (2014). A report on the DSL shared task 2014. In *Proceedings of the COLING VarDial Workshop*, pages 58-67.

Marcos Zampieri, Liling Tan, Nikola Ljubesic, Jorg Tiedemann, and Preslav Nakov. (2015). Overview of the DSL shared task 2015. In *Proceedings of the RANLP LT4VarDial workshop*, pages 1-9.

A Recorded Debating Dataset

Shachar Mirkin[1], Michal Jacovi[1], Tamar Lavee[1,2], Hong-Kwang Kuo[2], Samuel Thomas[2], Leslie Sager[3], Lili Kotlerman[1], Elad Venezian[1], Noam Slonim[1]

[1]IBM Research – Haifa, Israel, [2]IBM Watson – Yorktown Heights, New York, USA, [3]Tadiad, Tel Aviv, Israel

{shacharm,michal.jacovi,lili.kotlerman,eladv,noams}@il.ibm.com, tamar.lavee1@ibm.com,

{hkuo,sthomas}@us.ibm.com,sagerleslie@gmail.com

Abstract

This paper describes an English audio and textual dataset of debating speeches, a unique resource for the growing research field of computational argumentation and debating technologies. We detail the process of speech recording by professional debaters, the transcription of the speeches with an Automatic Speech Recognition (ASR) system, their consequent automatic processing to produce a text that is more "NLP-friendly", and in parallel – the manual transcription of the speeches in order to produce gold-standard "reference" transcripts. We release 60 speeches on various controversial topics, each in five formats corresponding to the different stages in the production of the data. The intention is to allow utilizing this resource for multiple research purposes, be it the addition of in-domain training data for a debate-specific ASR system, or applying argumentation mining on either noisy or clean debate transcripts. We intend to make further releases of this data in the future.

Keywords: debating technologies, computational argumentation, argumentation mining, automatic speech recognition

1. Introduction

Computational argumentation and debating technologies aim to automate the extraction, understanding and generation of argumentative discourse. This field has seen a surge in research in recent years, and involves a variety of tasks, over various domains, including legal, scientific writing and education. Much of the focus is on argumentation mining, the detection of arguments in text and their classification (Palau and Moens, 2009), but many other tasks are being addressed as well, including argument stance classification (Sobhani et al., 2015; Bar-Haim et al., 2017), the automatic generation of arguments (Bilu and Slonim, 2016), identification of persuasive arguments (Wei et al., 2016), quality assessment (Wachsmuth et al., 2017a) and more. Multiple datasets are available for such research, mostly in English, such as the Internet Argument Corpus (Walker et al., 2012), that consists of numerous annotated political discussions in internet forums, ArgRewrite (Zhang et al., 2017), a corpus of argumentative essay revisions, and the datasets released by IBM Research as part of the Debater Project (Rinott et al., 2015; Aharoni et al., 2014). Lippi and Torroni (2016) list several additional such datasets. Further, Wachsmuth et al. (2017b) have released an argument search engine over multiple debating websites, and Aker and Zhang (2017) have initiated the projection of some datasets to languages other than English, such as Chinese.

All of the above are based on written texts, while datasets of spoken debates, outside of the political domain, are scarce. A spoken debate differs from a written essay or discussion not only in structure and content, but also in style as in any other case of spoken vs. written language. Zhang et al. (2016) made available transcripts from the Intelligence Squared[1] debating television show[2]. The transcripts of the show are available on the show's site, and while they are of high quality, they do not match the audio recordings precisely, requiring substantial additional effort, if one wishes, for example, to use them as ASR training data.

With this paper we release a dataset of 60 audio speeches, recorded specifically for debating research purposes. We describe in detail the process of producing these speeches and their automatic and manual transcripts. This is a first batch of a larger set of recordings we intend to produce and release in the future.

2. Recording the Speeches

We recorded short speeches about debatable topics, with experienced speakers. This section describes the recording process.

Recruiting and training the speakers Our team of speakers are all litigators or debaters, fluent or native English speakers, experienced in arguing about any given topic. The recruitment and training of the speakers included several steps. First, we interviewed potential speakers to evaluate their ability to argue about a topic when given only a short time to prepare. Then, we provided candidates with an essay to read aloud and record. Candidates were given technical guidelines to ensure high recording quality, including microphone configuration instructions and recording best-practices such as to record in a quiet environment, to use an external microphone and to maintain a fixed distance from the microphone while speaking. After listening to these recordings, we provided the speakers with feedback and repeated the process until the essay recordings were of good quality for the naked ear. Next, we provided each candidate with two motions (e.g. "we should ban boxing") and asked them to record a spontaneous speech supporting each motion, after a 10 minute preparation.

All recordings – three per candidate (one reading and two spontaneous speeches) – were processed through automatic speech recognition and were sent to manual transcription, as described in the next sections. Comparing the automatic and manual transcripts, we computed the system's Word Error Rate (WER, the sum of substitution, deletion and in-

[1] http://www.intelligencesquaredus.org

[2] http://www.cs.cornell.edu/~cristian/debates/

sertion error rates) for each speech, and accepted candidates for whom the WER was below a pre-defined threshold of 10%. That, to make sure that our ASR system is reasonably successful on their speeches.

The recording process All speakers received a list of motions, each with an ID and a short name (to be easily identified by human readers), and background information extracted from Debatabase[3] or Wikipedia. The speakers were directed to spend up to 10 minutes reviewing the motion's topic and preparing their arguments, and then immediately start recording themselves arguing in its favor for 4-8 minutes. The speakers were instructed not to search for further information about the topic beyond the provided description. The idea is to prevent multiple debaters who record a speech about the same topic from reaching the same resources (in particular debating websites), which may reduce the diversity of the ideas presented in the speeches. Example 1 shows a part of background information for the topic "doping in sports".

Example 1 (Topic background information)

At least as far back as Ben Johnson's steroid scandal at the 1988 Olympics, the use of performance-enhancing drugs in sports had entered the public psyche. Johnson's world record sprint, his win, and then, the stripping of his gold medal made news around the world. However, performance-enhancing drugs in sports do not begin with Johnson ...

3. Automatic Speech Processing

Every recorded speech was automatically transcribed by a speaker-independent deep neural network ASR system. The system's acoustic model was trained on over 1000 hours of speech from various broadband speech corpora including broadcast news shows, TED talks[4] and Intelligence Squared debates[5]. We used a 4-gram language model with a vocabulary of 200K words, trained on several billion words that include transcripts of the above speech corpora and various written texts, such as news articles.

The ASR system we used is similar to those described in (Soltau et al., 2013; Soltau et al., 2014). We trained speaker-independent convolutional neural network (CNN) models on 40 dimensional log-mel spectra augmented with delta and double delta features. Each frame of speech is also appended with a context of 5 frames. The first CNN layer of the model has 512 nodes attached with 9×9 filters. Outputs from this layer are then processed by a second feature extraction layer, also with 512 nodes but using a set of 4×3 filters. The outputs from the second CNN layer are finally passed to 5 fully connected layers with 2048 hidden units each, to predict scores for 7K context-dependent states. This speaker-independent ASR system performs on average at 8.4% WER on the speeches we release with this paper.

[3] `http://idebate.org/debatabase`

[4] `https://www.ted.com/`

[5] We semi-automatically aligned the transcripts and the audio, to overcome the inconsistency problem mentioned in Section 1.

Once a speech has been automatically transcribed, we obtain a text in the format shown in Example 2. Each token (including sentence boundary and silence markers `<s>`, `<s/>`, `~SIL`) is followed by the start and end time of its utterance, in seconds, relative to the beginning of the recording segment.

This format is the basis for two versions of the data that we release for each speech: an automatically processed "clean" ASR version, and a manually transcribed one. The steps for obtaining the former are described in Section 3.1. The production of manual transcripts is described in Section 4.

Example 2 (Raw ASR output)

```
<s>[0.000,0.660] we[0.660,0.830] should
[0.830,1.060] allow[1.060,1.470] doping
[1.470,2.010] in[2.010,2.240] sports
[2.240,2.920] </s>[2.920,3.100] so
[3.100,3.280] by[3.280,3.390] this
[3.390,3.580] we[3.580,3.710] mean
[3.710,4.110] </s>[4.110,4.140] steroids
[4.140,4.950] </s>[4.950,5.080] human
[5.080,5.390] growth[5.390,5.680] hormone
[5.680,6.150] and[6.150,6.290] other
[6.290,6.500] similar[6.500,6.960] drugs
[6.960,7.490] ~SIL[7.490,7.670] should
[7.670,7.890] be[7.890,8.000] allowed
[8.000,8.330] in[8.330,8.430] pro
[8.430,8.840] and[8.840,9.010] amateur
[9.010,9.400] sports[9.400,10.030] </s
>[10.030,10.070]
```

3.1. ASR transcripts

To obtain a "clean" version of the raw ASR output stream, we post-process it, as detailed below. After this processing, the text in Example 2 is converted to the text in Example 3.

- **Removal of timing information**.
- **Removal of non-textual tokens:** Silence markers, `~SIL`, appear whenever a relatively long pause has been detected in the speech; sentence boundary tags, `<s>` and `</s>`, denote predicted beginnings and ends of sentences. These are the result of the fact that the ASR language model was trained not only on spoken language transcripts, but also on written texts that contain punctuation marks. We have experimentally determined that, for our data, these predictions are not reliable enough to be utilized for sentence splitting on their own and used a dedicated method for this purpose, as described below. We also remove tags such as `%hes`, denoting unspecified speaker's hesitation, as well as other tokens denoting hesitation that were transcribed explicitly, such as `ah`, `um` or `uh`.
- **Abbreviations reformatting:** The ASR-produced underscored abbreviation (initialism) format (`i_b_m`) is replaced with the standard all-caps one (`IBM`).
- **Automatic punctuation and sentence splitting:** The automatically transcribed text contains no punctuation marks. In downstream tasks, such as syntactic parsing, long texts are often difficult to handle, and we consequently split the stream of ASR output into sentences. Unlike typical sentence-splitting methods, whose

main goal is to disambiguate between periods that mark end-of-sentence and those denoting abbreviations, here the text contains no periods, hence a different method is required. We employed a bidirectional Long Short-term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) to predict commas and end-of-sentence periods over the ASR output. This neural network was trained on debate speeches, like the ones we share in this paper, and on TED talks, taken from the English side of the French-English parallel corpus from the IWSLT 2015 machine translation task (Cettolo et al., 2012). [6]

- **Capitalization:** We apply basic truecasing to the text: capitalizing sentences' first letters and occurrences of "I". We have experimented with more sophisticated truecasing tools and abstained from employing them to the released texts due to mixed results.

Example 3 (Clean ASR output)

```
We should allow doping in sports.
So by this, we mean steroids, human growth
hormone and other similar drugs should be
allowed in pro and amateur sports.
```

4. Manual Transcription

As mentioned, the ASR process produces imperfect texts. In order to obtain a "reference" text – a precise transcript of the speech – we employ human transcribers to post-edit the automatic transcript, i.e. correct its mistakes.

Transcribers selection and training We invited 15 candidates to train as transcribers, all of which are native or fluent English speakers, experienced in linguistic annotation tasks. As a first test, we asked them to transcribe the same four speeches, after carefully reading the guidelines. We used their outcomes for creating ground-truth transcripts: for each speech, we compared its transcripts pair-wise, listened carefully to points of differences, and created a "gold-transcript" that resolved all differences between the individual transcripts. Using these four gold-transcripts, we scored the work of the individual transcribers, and accepted as transcribers nine of the candidates whose transcripts were at least 98% accurate. They were further trained by transcribing ten speeches each, and getting feedback on them upon our review. Once done, we considered them "experienced transcribers".

Transcription methodology In our experience, starting from initial transcripts produced by ASR can halve the time necessary to produce reference transcripts, while maintaining similar transcript quality. This is particularly true if the ASR is highly accurate since it reduces the number of corrections the human transcriber has to make. One should be aware, however, that this procedure can introduce bias, depending on how conscientious the human transcriber is. An inexperienced or less conscientious transcriber may neglect to correct some ASR mistakes.

It is also easier for human transcribers to process shorter segments of speech, especially if they have to listen multiple times to unclear segments. Hence, to speed up the process of human transcription, the audio and transcript are first segmented by cutting them at silences longer than 500ms. Excessively long audio segments are then further divided at their longest silences, which must be at least 100ms. Note that the resulting segments do not necessarily correspond to linguistic boundaries or to where punctuation marks should be placed. Instead, in spontaneous speech, a person may pause in the middle of a sentence when faced with an increased cognitive load, e.g. when trying to recall a word. Similar methods of using ASR output as a basis for manual transcription were applied, e.g., by (Park and Zeanah, 2005) and (Matheson, 2007), for the purpose of transcribing interviews for interview-based research.

The human transcribers used Transcriber[7], a tool for assisting manual annotation of speech signals through a graphical user interface. The tool synchronizes the text with the audio, and allows the human transcriber to review the text while listening to the audio, and easily pause, fix, annotate, and continue listening from a selected segment.

On average, the time needed for manual transcription by experienced transcribers was approximately five times the duration of the audio file. An example of the input to the tool – the output of the above-mentioned segmentation process – is presented in Example 4. The output of the post-edition, which uses the same format, is shown in Example 5.

The guidelines used for manual transcription explain how to deal with cases such as speaker hesitation, repetitions and utterance of incomplete words, what punctuation marks to use[8], how to write abbreviations, numbers, etc. The main principles are that the transcripts should be accurate with respect to the source, capture as much signal as possible, and that they should maintain a uniform format that can be easily parsed in subsequent processing.[9]

Example 4 (Input for manual transcription)

```
<Sync time="18.020"/>
doping is the use of performance enhancing
drugs
<Sync time="21.290"/>
at what i
<Sync time="22.030"/>
am talking about sports i am of course
referring to
<Sync time="25.015"/>
a competitive sports
<Sync time="26.630"/>
for example the olympics
<Sync time="28.320"/>
or other kinds of competitions
<Sync time="30.040"/>
like a true the fonts
<Sync time="31.800"/>
and etcetera
```

[6] This is a simplified version of (Pahuja et al., 2017).

[7] http://trans.sourceforge.net/en/presentation.php; We used version 1.5.1

[8] The ASR does not produce punctuation marks; it turned out that the transcribers preferred adding them, as it made the text more readable. Punctuation also makes the texts more accessible for analysis and annotation and may be helpful for some automatic processing tasks.

[9] The transcription guidelines are shared with the released data.

Example 5 (Output of manual transcription)

```
<Sync time="18.020"/>
doping is the use of performance enhancing
drugs .
<Sync time="21.290"/>
uh when i
<Sync time="22.030"/>
am talking about sports i am of course
referring to
<Sync time="25.015"/>
uh competitive sports ,
<Sync time="26.630"/>
for example the olympics
<Sync time="28.320"/>
or other kinds of competitions
<Sync time="30.040"/>
like uh tour de france
<Sync time="31.800"/>
uh etcetera ,
```

4.1. Reference Transcripts

Some of the annotations in the post-edited transcripts are mostly useful for ASR training, as in the case of word mispronunciation and its correction (e.g. "lifes/lives"), while others contain signals that may also be useful for downstream text processing.

Our approach in producing the reference transcripts was to remove all non-textual annotations, producing a text-only version of the transcription, that can be used as-is, e.g. for argument extraction. From the Transcriber's output, we first removed all SGML tags and merged the lines into a single stream. We then removed incomplete words and mispronounced words (replacing them with the correct pronunciation); similarly to the raw ASR post-processing, we removed annotations, hesitations, reformatted abbreviations and applied basic truecasing. Then, we detokenized the text, i.e. removed any unnecessary spaces between tokens, for example, before a punctuation mark. Lastly, we applied automatic spell-checking to detect typos and formatting errors, and sent the identified instances of possible typos for review. Example 6 shows the text segment from Example 5 after going through this cleaning.

Example 6 (Clean reference transcript)

```
Doping is the use of performance enhancing
drugs.
When I am talking about sports I am of
course referring to competitive sports, for
 example the olympics or other kinds of
competitions like tour de france etcetera,
```

5. Dataset

The dataset we created was generated through the process described in the previous sections. We release all file types, including raw and clean versions, to enable research based on various signals, including audio-based ones, such as prosody or speech rate, and to allow performing different post-processing. Table 1 summarizes the files that are obtained and released for each debatable topic.

Extension	Description
`wav`	Recorded speeches
`asr`	Raw automatic transcripts
`asr.txt`	Clean automatic transcripts
`trs`	Manual transcripts
`trs.txt`	Clean manual transcripts (references)

Table 1: Summary of the dataset file types.

ID	Topic	Speeches	WER (%)
1	Violent video games	6	7.4
21	One-child policy	5	8.3
61	Doping in sports	5	7.7
101	Affirmative action	5	9.6
121	Boxing	5	9.6
181	Multiculturalism	2	8.5
381	The monarchy	5	7.3
482	Cultivation of tobacco	3	8.2
483	Freedom of speech	5	6.7
602	School vouchers	5	7.2
644	Year-round schooling	1	8.9
681	Intellectual property	3	10.9
701	Endangered species	2	6.8
841	Blasphemy	3	9.3
881	Holocaust denial	3	9.8
945	Infant circumcision	2	11.2

Table 2: List of motion topics in our dataset, and the number of speeches per topic. The right column shows the average WER across speeches of the topic, when using the speaker-independent ASR model.

The dataset we release includes 60 speeches for 16 motions from (Rinott et al., 2015), recorded by 10 different speakers.[10] Table 2 provides details about the recordings included in the dataset.

There is a large variance in WER across different debate recordings, and between different speakers. The WER of any specific debate can vary depending on the degree of mismatch with the ASR acoustic and language models. Examples of mismatch include differences in speaker voice, speaking style and rate, audio capture (microphone type and placement), ambient noise, word choice and phrasing, etc. By reducing mismatch through model adaptation of speaker-dependent acoustic models, the WER can be significantly reduced. For instance, with adaptation using about 15 minutes of a speaker's data, WER of a speech from topic 61 was reduced from 12.9% to 8.6%, and of a speech from topic 483, from 12.2% to 9.7%.

The dataset is freely available for research at `https://www.research.ibm.com/haifa/dept/vst/mlta_data.shtml`.

6. Acknowledgements

We wish to thank the many speakers and transcribers that took part in the effort of creating this dataset.

[10] Currently, the list contains only a single female speaker; we are making an effort to recruit more female debaters.

7. Bibliographical References

Aker, A. and Zhang, H. (2017). Projection of argumentative corpora from source to target languages. In *Proceedings of the 4th Workshop on Argument Mining*, pages 67–72.

Bar-Haim, R., Edelstein, L., Jochim, C., and Slonim, N. (2017). Improving claim stance classification with lexical knowledge expansion and context utilization. In *Proceedings of the 4th Workshop on Argument Mining*, pages 32–38.

Bilu, Y. and Slonim, N. (2016). Claim synthesis via predicate recycling. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, page 525.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technology*, 16:10:1–10:25.

Matheson, J. L. (2007). The voice transcription technique: Use of voice recognition software to transcribe digital interview data in qualitative research. *Qualitative Report*, 12(4):547–560.

Pahuja, V., Laha, A., Mirkin, S., Raykar, V., Kotlerman, L., and Lev, G. (2017). Joint learning of correlated sequence labelling tasks using bidirectional recurrent neural networks. In *Proceedings of INTERSPEECH*, Stockhold, Sweden.

Palau, R. M. and Moens, M.-F. (2009). Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM.

Park, J. and Zeanah, A. E. (2005). An evaluation of voice recognition software for use in interview-based research: a research note. *Qualitative Research*, 5(2):245–251.

Rinott, Ruty and Dankin, Lena and Perez, Carlos Alzate and Khapra, Mitesh M and Aharoni, Ehud and Slonim, Noam. (2015). *Show Me Your Evidence-an Automatic Method for Context Dependent Evidence Detection.* Association for Computational Linguistics (ACL).

Sobhani, P., Inkpen, D., and Matwin, S. (2015). From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining, ArgMining@HLT-NAACL*, pages 67–77.

Soltau, H., Kuo, H., Mangu, L., Saon, G., and Beran, T. (2013). Neural network acoustic models for the DARPA RATS program. In *Proceedings of Interspeech*. ISCA.

Soltau, H., Saon, G., and Sainath, T. (2014). Joint training of convolutional and non-convolutional neural networks. In *Proceedings of ICASSP*. IEEE.

Wachsmuth, H., Naderi, N., Habernal, I., Hou, Y., Hirst, G., Gurevych, I., and Stein, B. (2017a). Argumentation quality assessment: Theory vs. practice. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 250–255.

Wachsmuth, H., Potthast, M., Al Khatib, K., Ajjour, Y., Puschmann, J., Qu, J., Dorsch, J., Morari, V., Bevendorff, J., and Stein, B. (2017b). Building an argument search engine for the web. In *Proceedings of the 4th Workshop on Argument Mining*, pages 49–59.

Wei, Z., Liu, Y., and Li, Y. (2016). Is this post persuasive? ranking argumentative comments in the online forum. In *The 54th Annual Meeting of the Association for Computational Linguistics*, page 195.

Zhang, J., Kumar, R., Ravi, S., and Danescu-Niculescu-Mizil, C. (2016). Conversational flow in Oxford-style debates. In *Proceedings of HLT-NAACL*.

Zhang, F., B. Hashemi, H., Hwa, R., and Litman, D. (2017). A corpus of annotated revisions for studying argumentative writing. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1568–1578, Vancouver, Canada, July. Association for Computational Linguistics.

8. Language Resource References

Aharoni, Ehud and Polnarov, Anatoly and Lavee, Tamar and Hershcovich, Daniel and Levy, Ran and Rinott, Ruty and Gutfreund, Dan and Slonim, Noam. (2014). *A benchmark dataset for automatic detection of claims and evidence in the context of controversial topics*. Association for Computational Linguistics (ACL).

Mauro Cettolo and Christian Girardi and Marcello Federico. (2012). *WIT^3: Web Inventory of Transcribed and Translated Talks*. The European Association for Machine Translation (EAMT).

Rinott, Ruty and Dankin, Lena and Perez, Carlos Alzate and Khapra, Mitesh M and Aharoni, Ehud and Slonim, Noam. (2015). *Show Me Your Evidence-an Automatic Method for Context Dependent Evidence Detection.* Association for Computational Linguistics (ACL).

Marilyn Walker and Jean Fox Tree and Pranav Anand and Rob Abbott and Joseph King. (2012). *A Corpus for Research on Deliberation and Debate*. European Language Resources Association (ELRA).

Building a Corpus from Handwritten Picture Postcards: Transcription, Annotation and Part-of-Speech Tagging

Kyoko Sugisaki, Nicolas Wiedmer, Heiko Hausendorf
German department, University of Zurich
Schönberggasse 9, 8001 Zurich, Switzerland
{sugisaki, nicolas.wiedmer, heiko.hausendorf}@ds.uzh.ch

Abstract

In this paper, we present a corpus of over 11,000 holiday picture postcards written in German and Swiss German. The postcards have been collected for the purpose of text-linguistic investigations on the genre and its standardisation and variation over time. We discuss the processes and challenges of digitalisation, manual transcription, and manual annotation. In addition, we developed our own automatic text segmentation system and a part-of-speech tagger, since our texts often contain orthographic deviations, domain-specific structures such as fragments, subject-less sentences, interjections, discourse particles, and domain-specific formulaic communicative routines in salutation and greeting. In particular, we demonstrate that the CRF-based POS tagger could be boosted to a domain-specific text by adding a small amount of in-domain data. We showed that entropy-based training data sampling was competitive with random sampling in performing this task. The evaluation showed that our POS tagger achieved a F1 score of 0.93 (precision 0.94, recall 0.93), which outperformed a state-of-the-art POS tagger.

Keywords: postcard corpus, POS tagging, German

1. Introduction

In this paper, we report the construction of the language resource *Ansichtskartenkorpus* (*[anko]*), 'picture postcard corpus', containing over 11,000 holiday postcards written in Standard German and Swiss German. They were manually transcribed and annotated with structural and discourse-related information, and then automatically annotated with text segmentation, lemma and part-of-speech (POS) information.

We will first characterise the texts contained in the resource (Section 2), and then describe their manual transcription and annotation before outlining the development of a NLP toolkit for text segmentation and POS annotation (Section 3).

2. Data Source

The holiday postcards were collected at the University of Zurich from 2009 to present day for the purpose of text-linguistic investigations on the genre and its standardisation and variation over time. The postcards included in our corpus were sent by post from people on holiday, mainly from Switzerland but also from Italy, Germany and other European countries to their family, friends, colleagues and neighbours living in the German-speaking area of Switzerland. About 95% of the cards (11,760 cards) were written mainly in Standard German. The remaining part of the corpus is comprised of postcards written mainly in Swiss German. Although the postcards were dated from 1898 to 2016, the majority were written in the 1980s (22%) and 1990s (19%). On average, a post card contains 50 words, while individual post cards vary from one to 350 words.

3. Corpus Construction

In this section, we describe the process of digitalisation, transcription, and annotation carried out manually and automatically to build the corpus of the collected postcards.

3.1. Overall Pipeline: From Digitalisation to XML with Linguistic Annotation

Because the collected holiday postcards were in paper format, we first scanned the front and back of each card. We then considered using an optical character recognition (OCR) system to extract the texts from the scanned images. However, the postcards were handwritten in German, and OCR systems do not work well for handwritten texts in languages other than English. Therefore, we decided on manual transcription for which we developed a web-based tool. The user interface is illustrated in Figure 3.2. Each scanned card was integrated into the tool. The tool displayed the front and back images of each card on the left side and the transcription and annotation forms on the right side. Thus, the transcribers could directly transcribe handwriting, mark paragraphs, note textual discourse structures (e.g. greetings) and enter metainformation (e.g. dates). The data were then saved in a MySQL database, which we then converted to an XML representation. We then incorporated our automatic annotations in the XML: 1) text segmentation, 2) lemma and 3) POS tags.

3.2. Transcription and Manual Annotation

The picture postcards written in Standard German were transcribed and annotated by four transcribers in a typing office in Germany. The Swiss German postcards were transcribed and annotated by a student whose native language is Swiss German. To ensure the quality of the transcription and the manual annotation, during the process of transcription and annotation, three students checked samples, corrected them manually and gave feedback to the typing office.

Our corpus consisted of the main texts as primary data and textual properties as metadata. A picture postcard consists of two sides – the front side and the back side. The front side of a modern postcard typically includes images of tourist attractions and landscapes, including the name of the location, whereas the back side consists of an address field on the

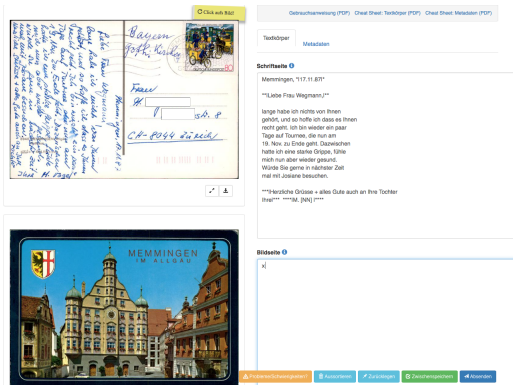


Figure 1: Web-based manual transcription/annotation tool

(A) Word/lemma features	
A1	**Word form**: real word forms
A2	**Normalized word form**: all lower case and without ü
A3	**Character type of unit**: word form is categorised into the following classes: (1) all special characters (2) all numbers (3) capitalized (4) all alphabets without capitalization (5) mixed of all possible character without capitalization
A4-7	**Suffix**: the last 4, 3, 2, 1 character of words, respectively.
A8	**Lemma**: generated by TreeTagger
(B) POS	
B1	**POS**: generated by TreeTagger
B2	**POS**: generated by Stanford POS tagger
(C) Semantic cluster features	
C1-2	**Brown clustering**: Brown clustering is used in 4 digits (D1) and all digits (D2)
C3	**Word2Vec**
C4	**Fasttext**

Table 1: Features for CRF-based POS tagging

right and a message field on the left. During the transcription process, the message field was transcribed and regarded as primary data. The address field (e.g. name, postal code, location and country of the receiver) was considered metadata, including latent information, such as the genders of both receiver and the sender, as well as the presence of sketches drawn by the latter.

In addition, our transcribers annotated textual discourse-related information during the transcription process. The message field of a holiday postcard is generally structured as follows: 1) a preface (date, sometimes location, temperature or weather); 2) a salutation (e.g. *Dear Heidi*); 3) the main message; 4) greeting including closing (e.g. *Cheers*); 5) the signature of the sender. During the transcription, the preface, salutation, greeting and signature were marked directly on the text. Each beginning and end of these discourse zones were marked with unique markdowns. The markdowns consisted of character sequences that hardly appeared in the main text. The salutation was marked as star star bar `**|`, and the closing was marked as `|**`. For example, the salutation *Dear Heidi* in the main text was annotated as `**|Dear Heidi|**`. Hence, minimal annotation was required, and the mapping to XML opening and closing tag was straightforward.

We considered that sensitive data in the corpus should be explicitly coded. The picture postcards often contained private information, such as the name and address of the receiver, the telephone number or even the bank account number of the sender. Therefore, the transcribers did not include such sensitive information but coded as `[Vertraulich]` (i.e. 'confidential') in the message field. In particular, family names are coded as `[NN]` (i.e. the short form of *Nachname* or 'family name'). The sensitive data in the address field were marked as such to ensure that they will not be released in the corpus.[1]

3.3. Automatic Text Segmentation

The primary texts were segmented into paragraphs, sentences and words. The segmented texts were then structured in a XML representation.

Generally in German, punctuation segments a text into sentences, and spaces are used to segment a sentence into words. However, this rule of thumb was not always applicable to the sentence segmentation of the postcards, particularly with regard to the following cases: 1) punctuation was a part of a token with preceding characters; and 2) punctuation was absent. Case 1 refers to abbreviations (e.g. *z.B.* instead of *zum Beispiel* or 'for example') and brand or proper names (e.g. *Sat.1*), which is also common in Standard German orthography. Case 2 refers to freestanding lines, which typically ended with a wide blank space or extra line spacing, and which often omitted punctuation, such as titles, subtitles, addresses, dates, greetings, salutations and signatures (Official German Orthography, 2006). Dates, greetings, salutations and signatures belong to the core text zones of postcards. In addition, freestanding lines were often extended to the end of the paragraph in the texts of the postcards. Furthermore, the following use of punctuations is also common in postcards, which differs from Standard German orthography: (a) repeated punctuation (e.g.,*!!!*,*???*,.....) in order to emphasise words, phrases and sentences; (b) the use of emotional pictograms that are typically composed of punctuation (e.g. *:),;-)*). Based on these peculiarities, we developed a statistical sequential sentence segmentation system that differentiates punctuations into Case (1) and the sentence boundary, and deliberately handles Case (2) (Sugisaki, 2017).

With regard to tokenisation, the texts of the postcards showed a frequent use of contractions, which is also common in internet-based and computer-mediated communication (Bartz et al., 2013). In the contractions, the verb was often combined with the pronoun *es*, 'it', and delimitated by an apostrophe (e.g. *gibt's* instead of *gibt es*, 'gives it'). The apostrophe was sometimes omitted (e.g. *gibts*). Nonetheless, not only verbs are concatenated with the pronoun, but also in 'wh question' words (*wenn's/wo's* instead of *wenn es/wo es*, 'when/where it') and prepositions (*auf's* instead of *aufs* or *auf das*, 'on the'). Based on this observation, we developed a simple rule-based tokeniser in which *'s* was separated from the remaining part of the token if it was not a noun. If it was a noun, the *'s* was considered a genitive marker and part of the token. We used TreeTagger (Schmid, 1995) to obtain the POS information. However, in the case of contractions without apostrophes, TreeTagger does not provide an accurate POS tag. Contractions without apostrophes do not belong to standard orthographies, which might cause this difficulty. We observed that frequently used verbs, such as *give, be* and *have* often occurred with the reduced pronoun *s* without an apostrophe. Therefore, we created a list of these verbs and some wh question words in order to separate *s* from them.

[1] A sample of our corpus will be available at `http://ansichtskartenprojekt.de`

3.4. Part-of-speech Tagging

The segmented tokens were further annotated with POS tags that were integrated into the XML representation. We developed a POS tagger for the postcards. The texts comprised a mixture of Standard German and Swiss German. In addition, the targeted texts were in written form, but conceptually, they were in near-oral language (Koch and Oesterreicher, 2008; Dürscheid, 2016). An off-the-shelf POS tagger is typically trained on a corpus of newspapers written in Standard German. A newspaper article belongs to the category of a prototypical written language in both form and concept. Furthermore, it contains fewer orthographical deviations. Therefore, we experimented with features and training data to determine the best method for optimising the accuracy of the tagger applied to the postcard text in this study.

3.4.1. Experimental Setting

In the experiments, we used the tagging method of conditional random fields (CRF) (Lafferty et al., 2001). CRF is a supervised machine learning method for sequences. For the experiments, we created the following three data sets:

1. TüBa-D/Z v. 10, Tübinger Baumbank des Deutschen/Zeitungskorpus (Telljohann et al., 2012), which is a German newspaper corpus (1.787.801 tokens, henceforth *TüBa*). In our first experiments, approximately 80% of the TüBa (803.040 tokens (henceforth, *TüBa80*) were used as training data, and 20% of the TüBa tokens were used as test data (252.784 tokens, henceforth TüBa20). In the second experiment, we used all the TüBa (*TüBa100*) tokens as training data. In addition, we used a cross-validation data set (2.239 tokens) in all experiments.

2. NOAH's Corpus of Swiss German Dialects (henceforth, *NOAH*) (Hollenstein and Aepli, 2014) is a Swiss German corpus (94.306 tokens) that contains a variety of texts (blogs, reports, Wikipedia, etc.). In our experiments, we used the corpus as training data.

3. From the Ansichtskartenkorpus, or 'picture postcard corpus'(henceforth, *ANKO*), we first manually annotated 200 postcards to derive the test data. The test data were sampled randomly from the corpus and divided into two sets: 100 cards for the experiment (5.048 tokens, henceforth *ANKO-TEST*) and 100 cards for the evaluation (5.341 tokens, henceforth *ANKO-EVAL*). In addition, we manually annotated 1,500 sentences for the experiments. The sentences were used as training data, and they were sampled in three ways: 1) 300 sentences were selected randomly (henceforth, *ANKO-R*); 2) 1,200 sentences were selected based on word 4-gram-based entropy scores according to four measurements. We describe the entropy sampling method in Section 3.4.3. In our experiments, we used the Standard German sub-corpus of *ANKO*.

The set of linguistic features used in our experiments is provided in Table 1. The features were divided into (A) word and lemma, (B) POS features generated by the POS tagger TreeTagger (Schmid, 1995) and the Stanford POS Tagger (Toutanova et al., 2003); and (C) semantic clusters generated by unsupervised machine learning methods, that is, Brown clustering[2] (Brown et al., 1992), [3] (Mikolov et al., 2013) and fasttext[4] (Bojanowski et al., 2016).
In the following subsections, we describe the experiments using the set of linguistic features and the data sets.

3.4.2. Features

We trained CRF models on the training set of TüBa and tested them on the test set of TüBa and ANKO. We trained four different types of features (A to C in Table 1) separately and all features in context window 0 (i.e., current tokens). The results are shown in Table 2. As expected, tagging accuracy (F1 score) was lower if the training data and test data were derived from different domains. Regardless of the test data, the best features were the word and lemma features (A). The morphosyntactic analysis using the existing POS taggers showed a lower performance, and the semantic features (B) did not achieve high accuracy. However, the combination of these three types of features outperformed the word/lemma features. We extended the feature sets of (A), (B) and (C) from context window 0 (current tokens) to 5 left and right context windows. The results are shown in Table 1. The main finding was that the window side did not affect the accuracy as much as expected. However, the wider context window size slightly improved the accuracy of the test set of TüBa. Therefore, we conducted further experiments using the combination of the feature sets (A), (B) and (C) in context windows 0 to 5.

3.4.3. Training Data

In this section, we investigate the following challenges: 1) how to boost the tagging accuracy in texts with mixed languages and 2) whose domain and morphosyntactic distribution were different from newspapers.
To handle the first challenge, we added the Swiss German training data, NOAH. The results are shown in Table 3. The addition of the Swiss German training data produced results that were similar to those of the model that was trained only on TüBa100, but it did not improve the tagger.
To address the second challenge, we added small amounts of five types of training data from ANKO to the TüBa100 and NOAH training data. The first in-domain training data were randomly selected from ANKO. The remaining data sets were selected using a cross entropy score. Cross entropy is a variant of perplexity that is used to compare different probability models. The score is measured as follows (Jurafsky

[2]For Brown clustering, we used the implementation of P. Liang. To create 100 clusters, we trained the model on TüBa100, NOAH, ANKO (normalized word form). The first 4 digits and all digits are used as features.

[3]For word2vec, we used gensim with parameters skip-gram, 500 dimensions, context window 5. For K-means clustering, we used the scikit-learn to create 30 clusters.

[4]We used the fasttext with parameters, CBOW, 200 dimensions, context window 5, 5 word ngrams. For K-means clustering, we used the scikit-learn to build 20 clusters.

Context window	0				0-1	0-3	0-5
Feature	**Feature A**	**Feature B**	**Feature C**	**Feature A-C**			
TüBa-Test	.968 (.968,.968)	.960 (.960,.961)	.893 (.893,.894)	.974 (.974,.974)	.977 (.977,.977)	**.978 (.978,.978)**	**.978 (.978,.978)**
ANKO-Test	.883 (.886,.881)	.848 (.850,.846)	.795 (.796,.794)	**.897 (.900,.895)**	.895 (.897,.893)	.892 (.895,.890)	.895 (.897,.893)

Table 2: Experiments with features in context window 0, 0-1, 0-3, 0-5: Training data =TüBa80:F1 score (precision, recall)

(2009, pp. 117)):

$$H(w_1 \ldots w_n) = -\frac{1}{N} logP(w_1 \ldots w_n) \quad (1)$$

The goal of the in-domain training data selection was the automatic selection of a small number of in-domain sentences that might improve the tagging accuracy. Ideally, the in-domain sentences to be selected were not observed in the training in TüBa and NOAH but were typical in ANKO. We considered two methods: 1) ranking-based entropy scoring (henceforth, method [A]) and 2) difference-based entropy scoring (henceforth, method [B]). Ranking-based entropy scoring is a measurement of how informative in-domain sentences are based on a language model trained on out-of-domain data. The entropy scores were ranked in order from high to low. In this method, in-domain sentences with high entropy scores were assumed distinct from the out-of-domain data and thus more informative. This method is compatible with Axelrod and Gao (2011) in which perplexity was used instead of cross entropy. We inspected the top 300 sentences. They included salutations, greetings, signatures and dates. These discourse types are typical in postcards but are rarely included in a newspaper corpus. In contrast, difference-based entropy is a measurement of differences in entropy scores based on a language model trained on both out-of-domain and in-domain sentences. In-domain sentences were considered informative if the difference in score was large. This method is based on Moore and Lewis (2010). We inspected the top 300 sentences. These sentences were similar to those selected by the ranking-based entropy scores, and they were a mixture of typical discourse structures.

However, the selected sentences did not include in-domain interpersonal and fragmental sentence patterns typically used in private communication. Thus, we did not find any sentences whose subject was in the first or second person, such as *Danke für Deine Karte.* ('Thank you for your card') or fragments such as *sind glücklich hier oben gelandet* ('happily landed up here above'). Here, we found that the variance in higher entropy scores was high in TüBa (mean: 10, variance: 914) and low in ANKO (mean: 1, variance: 3), which indicated that the difference-based entropy scores were mainly guided by the TüBa scores. Therefore, these two methods selected similar sentences.

To detect typical main sentences in ANKO, we introduced two methods: in-domain ranking-based entropy score (henceforth, method [C]) and a difference-ranking-based entropy score (henceforth, method [D]). Method (C) was used to select the sentences with lowest entropy scores based on a language model trained on the in-domain data. In method (D), we simply ranked the entropy scores trained on TüBa and on ANKO, and we ordered the difference in ranking from high to low.

Training data	Test
TüBa100	.899 (.902,.897)
TüBa100 + NOAH	.898 (.901,.896)
TüBa100 + NOAH + 100 ANKO-A	.910 (.913,.908)
TüBa100 + NOAH + 100 ANKO-B	.908 (.911,.906)
TüBa100 + NOAH + 100 ANKO-C	.922 (.924,.920)
TüBa100 + NOAH + 100 ANKO-D	.926 (.928,.924)
TüBa100 + NOAH + 100 ANKO-R	.931 (.934,.929)
TüBa100 + NOAH + 300 ANKO-R/A/B/C/D	.941 (.943,.939)

Table 3: Experiments with training data with features (A), (B), and (C), and test on the *ANKO-TEST*: F1 score (precision, recall)

We experiment on these domain-data selection methods (A)-(D) with random selection (R) as our baseline. For that, we manually annotated 300 sentences for the training set (R) and (A)-(D). The results are shown in Table 3. The 300 sentences selected by the (D) method outperformed the other three entropy-based sampling methods, which indicated that ranked-difference-based entropy scoring is a viable sampling method, particularly if differences in the variance of the entropy scores between out-of-domain and in-domain data are large. However, the selected sentences did not outperform the in-domain data that were selected at random. Finally, we tested the models trained on TüBa, NOAH and 1,200 training sentences in the postcards, which achieved the best F1 score of 0.94.

3.4.4. Evaluation

To evaluate the developed POS tagger, we created a test set that was derived from the postcard corpus (ANKO-EVAL). We re-trained the CRF model with the features A, B, and C and the training data, TüBa100, NOAH, ANKO (i.e. ANKO-Test, all ANKO in-domain training sentences R/A/B/C/D). For the comparison, we used TreeTagger. The evaluation revealed that our POS tagger achieved a F1 score of 0.93 (precision 0.94, recall 0.93), which outperformed TreeTagger's F1 score of 0.86 (precision 0.86, recall 0.86).

4. Conclusion

In this paper, we described the process of digitalising, transcribing and annotating of over 11,000 handwritten postcards. In particular, we demonstrated that the POS tagger could be boosted to a domain-specific text by adding a small amount of in-domain data. We showed that entropy-based training data sampling was competitive with random sampling in performing this task. In future work, we will test our POS tagger on text that is written in Swiss German.

5. Acknowledgments

This work has been funded under SNSF grant 160238. We thank all the project members, Joachim Scharloth, Noah Bubenhofer, Selena Calleri, Maaike Kellenberger, David Koch, Marcel Naef, Dewi Josephine Obert, Jan Langenhorst, Michaela Schnick.

6. Bibliographical References

Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 355–362.

Bartz, T., Beißwenger, M., and Storrer, A. (2013). Optimierung des Stuttgart-Tübingen-Tagset für die linguistische Annotation von Korpora zur internetbasierten Kommunikation: Phänomene, Herausforderungen, Erweiterungsvorschläge. *JLCL*, 28:157–198.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Dürscheid, C. (2016). *Einführung in die Schriftlinguistik*. Vandenhoeck & Ruprecht, Göttingen, 5 edition.

Hollenstein, N. and Aepli, N. (2014). Compilation of a Swiss German dialect corpus and its application to PoS tagging. *COLING 2014*, page 85.

Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Pearson Education International, Upper Saddle River, New Jersey, 2nd edition.

Koch, P. and Oesterreicher, W. (2008). Mündlichkeit und Schriftlichkeit von Texten. In Nina Janich, editor, *Textlinguistik: 15 Einführungen*. G. Narr.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML)*, pages 282–289.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers (ACLShort)*, pages 220–224.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop.*, Dublin, Ireland.

Sugisaki, K. (2017). Word and sentence segmentation in german: Overcoming idiosyncrasies in the use of punctuation in private communication. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology (GSCL)*.

Telljohann, H., Hinrichs, E. W., Sandra, K., Heike, Z., and Kathrin, B. (2012). Stylebook for the Tübingen treebank of written German (TüBa-D/Z). Technical report, Universität Tübingen.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL '03)*, pages 173–180.

A Lexical Tool for Academic Writing in Spanish based on Expert and Novice Corpora

Marcos García Salido, Marcos Garcia, Milka Villayandre, Margarita Alonso-Ramos
Universidade da Coruña, Universidad de León
Facultade de Filoloxía, Rúa Lisboa 7, Campus da Zapateira, 15008 A Coruña, Spain
Facultad de Filosofía y Letras, Campus de Vegazana s/n, 24071 León, Spain
{marcos.garcias, marcos.garcia.gonzalez, margarita.alonso}@udc.gal
milka.villayandre@unileon.es

Abstract

The object of this article is to describe the extraction of data from a corpus of academic texts in Spanish and the use of those data for developing a lexical tool oriented to the production of academic texts. The corpus provides the lexical combinations that will be included in the afore-mentioned tool, namely collocations, idioms and formulas. They have been retrieved from the corpus controlling for their keyness (i.e., their specificity with regard to academic texts) and their even distribution across the corpus. For the extraction of collocations containing academic vocabulary other methods have been used, taking advantage of the morphological and syntactic information with which the corpus has been enriched. In the case of collocations and other multiword units, several association measures are being tested in order to restrict the list of candidates the lexicographers will have to deal with manually.

Keywords: academic writing, corpus, writing aid, multi-word units, collocations

1. Introduction

One of the challenges faced by university students is producing texts written following the conventions of academic discourse. Writers of English, Dutch or French – to cite a few European languages – have at their disposal several production-oriented tools (Kübler and Pecman, 2012; D'Hertefelt et al., 2014; Granger and Paquot, 2015). Writers of Spanish can find several guidelines dealing mostly with structural features of academic texts, but, to the best of our knowledge, there is no resource (on paper or in electronic format) where they can search for lexical combinations in order to compose academic texts.

This paper presents a tool oriented to the production of academic texts in Spanish – the *Herramienta de Ayuda a la Redacción de textos Académicos*, henceforth, HARTA – based on a corpus of academic texts. This tool will mainly provide indications as to how to use vocabulary typical of the academic genre in order to build complete texts. More particularly, the tool will focus on *academic lexical combinations* (ALCs). By ALCs we mean recurrent sequences of words that may or may not be semantically compositional and fulfill rhetorical functions such as giving examples, introducing conclusions, expressing certainty or probability, etc.

Three types of ALCs will form the dictionary module: *collocations, idioms* and *formulas*. The ACLs occurring in the corpus will be classified into these three categories and treated accordingly in the tool, following the criteria proposed by Mel'čuk (2015) within the framework of Explanatory and Combinatorial Lexicology. *Idioms* are non compositional multiword combinations, such as *punto de vista* 'perspective', or *sin embargo* 'however', which will have their own entries along with monolexemic lexical units. *Collocations* are compositional combinations of two lexical units, one of which – the collocate – is chosen depending on the other – the base, such as *formular (una) hipótesis* 'to formulate a hypothesis' or *adoptar un punto de vista* 'to adopt a perspective'. Collocations will be provided in the entry of their bases (e.g., *formular* under the headword *hipótesis* and *adoptar* under the headword *punto de vista*). Finally, *formulas* are compositional combinations for which neither their meanings nor their encoding are freely selected. For instance, if speakers want to express the idea «Now I will rephrase what has been said before» in Spanish, they are not free to choose the meaning 'to put', in contrast to Eng. *to put it differently*, nor can they encode the meaning of 'differently' other than *de otra manera* or *de otro modo* (cf. *dicho de otra manera/*diferentemente/??de manera diferente*). Formulas will be given entries specifying their discourse function: e.g. *dicho de otra manera*: 'used to rephrase a previously introduce idea, argument, etc.' (for more details the reader is referred to Alonso-Ramos et al. (2017)).

Along with the expert subcorpus from which the ALCs which will be included in HARTA have been extracted, a novice writers subcorpus is being compiled containing texts of students in bachelor and master degrees. In what follows, we offer a description of both subcorpora, present some of the results obtained from them, and give an account of their incorporation in HARTA.

2. Corpus description

The HARTA Corpus consists of two subcorpora. The first subcorpus is devoted to the written production of experts and is made up of research articles published in scientific journals and originally written in Spanish. The core of this subcorpus (234 research articles) stems from the Spanish part of the SERAC 2.0 corpus (InterLAE Research Group, 2008). This core has been supplemented with 180 further articles in order to obtain four balanced subsections in terms of their size (see below).

The other subcorpus covers the production of novice writers, and is similar in conception to BAWE (Gardner and

Nesi, 2013) or CALE (Callies and Zaytseva, 2013), among others. It is a collection of bachelor and master degree theses publicly available from institutional repositories. Currently, 125 texts have been incorporated to this subcorpus and xml-marked, 77 bachelor's degree and 48 master's degree theses, amounting to a total of ca. 1.5 million running words.

After completing the incorporation of novice texts, each subcorpus will contain a total of ca. two million words. Consequently, the completed corpus will be similar in size to other corpora exploited for the creation of lexical resources in languages such as English (Coxhead, 2000; Paquot, 2010) or French (Tutin and Kraif, 2016).

2.1. Scientific domains

Both subcorpora are equally divided into four thematic sections: (i) Arts and Humanities, (ii) Biological and Health Sciences, (iii) Physical Science and Engineering, and (iv) Social Sciences and Education. Although there are standards with regard to the classification of scientific fields (UNESCO, 1978), these standards do not seem to have gained acceptance when it comes to compile corpora of academic discourse. Thus, the French SCIENTEXT corpus is divided in ten scientific domains (Tutin and Kraif, 2016), Coxhead (2000) distinguishes four big scientific areas (Arts, Commerce, Law, and Science) in turn subdivided into 27 subareas, Paquot (2010) draws a big divide between hard and soft sciences and further breaks down the former into four sections and the latter into six, etc.

The four thematic areas in both HARTA subcorpora are balanced for size as measured in number of words, like in the case of Coxhead (2000) or Paquot (2010). In this respect, the Spanish part of SERAC 2.0 had to be modified, since it was balanced with respect to the number of texts, rather than the number of words per domain. In its final versions, each subcorpus will include ca. 500,000 words for scientific domain.

2.2. Markup

HARTA corpus is xml marked. This markup makes explicit (a) editorial metadata of the texts included (author, year, article title, journal title, and, in the case of novice writers, university and degree – bachelor vs master) and (b) descriptive metadata relative to the scientific domains and textual sections – introduction, body, etc. – (see Figure 1).

As for text structure markup, all texts must include an obligatory *body* section and optional peripheral sections or subdivisions such as *introduction, methods, conclusion*, and *footnotes*. Even though research articles' structure is quite standard (cf. Swales and Feak (2004), among others), we have decided to keep most textual sections as optional, given the relatively loose structural conventions of some domains (e.g. articles on literature). The information provided by this kind of markup will enable sophisticated searches both to users of HARTA and to researchers, so that they can restrict their queries to specific domains or to particular sections (e.g., abstracts, conclusion sections, etc.).

```
<HARTA>
    <header>
        <article title="" author=""></article>
        <deposit year="" university=""/>
        <class type="" domain="" subdomain=""/>
        <size words=""/>
    </header>
    <content>
        <abstract><p>...</p></abstract>
        <introduction><p>...</p></introduction>
        <body><p>...<i>...</i>...
                <b>...</b>...</p></body>
        <conclusion><p>...</p></conclusion>
        <footnotes><p>...</p></footnotes>
    </content>
</HARTA>
```

Figure 1: HARTA's xml-markup

2.3. Morphological annotation and parsing

The expert corpus has been tokenized and lemmatized with LinguaKit (Garcia and Gamallo, 2016) and PoS-tagged with FreeLing (Padró and Stanilovsky, 2012). Subsequently, we have used UDPipe (Straka et al., 2016) to perform dependency parsing using universal dependencies (Nivre et al., 2016).

3. Preliminary findings

The corpus described in the previous section will feed HARTA with lexical units and combinations thereof retrieved from it. Such data will constitute the raw materials of which HARTA will be made after a manual revision on the part of lexicographers. The information extracted from the corpus is being or will be pre-processed by means of different techniques before handing it to the lexicographers. In this section we give an account of such pre-processing techniques.

3.1. Academic Word List

An Academic Word List (AWL) has been extracted from the HARTA expert subcorpus. The items of the list fulfill two requirements: (i) being specific or "key" to the academic corpus and (ii) having an even distribution through the whole expert subcorpus in order to discard discipline specific terms. The *keyness* or specifitiy of the list has been established by comparing the distribution of the lemmas corresponding to content words present in the HARTA expert subcorpus with their distribution in a non-academic corpus. Following Paquot (2010), we have used a corpus of fiction narrative as a strongly contrasting reference corpus: the fiction narrative part of LEXESP (Sebastián-Gallés et al., 2000). To determine whether each lemma was significantly more frequent in the academic corpus we used the Wilcoxon-Mann-Whitney (WMW) rank test (Paquot and Bestgen, 2009; Lijffijt et al., 2014) and discarded the pairs yielding a p-value equal or greater than 0.001. In order to apply this test, each lemma has been assigned a series of ranks derived from their frequency in the sections of the academic and fiction corpora. As sections of the academic corpus we used its very division in articles and we divided the fiction corpus in fragments of 5,000 words. Subsequently we obtained the frequency of each lemma in each

section, normalised it per 5,000 words and transformed it into the ranks that fed the WMW test.
Although the test results ultimately derive from information relative to counts in corpus sections, it is not clear whether it is sensitive to evenness of distribution – see Paquot and Bestgen (2009), who attribute this quality only to the t-test. For that reason, we also used Gries's Deviation of Proportions (DP) (Gries, 2008), a coefficient indicative of the evenness of distribution of the elements of a corpus. DP values near 0 correspond to the absence of differences with respect to the expected distribution, whilst values close to 1 are suggestive of highly skewed distributions. For our AWL we have kept lemmas with values under 0.5.
The resulting academic list contains 1080 lemmas of content words, i.e. nouns, adjectives, verbs and adverbs. The breakdown of the list into these four parts of speech can be seen in Table 1.

Part of Speech	No. of instances
noun	333
adjective	235
verb	384
adverb	128

Table 1: AWL breakdown by part-of-speech

This list contains the potential candidates for the collocation bases that will be included in HARTA. The final decision, however, will be made after a manual exam by expert lexicographers. Predictably, not all four categories will be equally productive in this respect. Thus, nouns are much more interesting than the other three categories as collocational bases.

3.2. Formulas and uninflected idioms list

Some ALCs tend to occur as invariable strings. This is the case of idioms with prepositional structures, such as *sin embargo* 'however', *a través de* 'by means of', *en lo que respecta a* 'as far as X is/are concerned' and formulas such as *en otras palabras* 'in other words', *como se ha visto (anteriormente/más arriba, etc.)* 'as seen (before)'. To obtain such sequences, we extracted n-grams and filter them by frequency (10 occurrences per million words, one of the thresholds conventionally used for the identification lexical bundles; see Biber et al. (1999)) and by dispersion with the same criteria indicated in Section 3.1. So far, we have extracted bi-grams, tri-grams and four-grams.
A frequency threshold alone in the case of bi-grams performs poorly and does not even distinguish combinations produced by chance from others (Bestgen, 2014). However, among bi-grams some interesting ALCs can be found, e.g.: *no obstante* 'notwithstanding', *cabe esperar* 'it should be expected', etc. For that reason, we resisted the idea of discarding bi-gram extraction and added to the frequency threshold other association measures – namely, pointwise Mutual Information, Backwards Transition Probability and Forward Transition Probability, cf. Appel and Trofimovich (2017) for the latter two. Whereas the precision in retrieving phraseological expressions benefits from these measures in the case of bi-grams, this effect is not so evident with longer n-grams. Table 2 shows the result of manually checking the top hundred items from n-grams lists sorted by the above-mentioned association measures.

	Freq.	MI	BTP	FTP
bi-gr	.09	.25	.67	.05
tri-gr	.39	.74	.59	.54
four-gr	.48	.52	.60	.38

Table 2: Precision of association measures in identifying phraseological n-grams

3.3. Collocations and inflected idioms list

Collocations and certain idioms (especially verbal ones) are not necessarily continuous invariable strings. This is especially evident in the case of collocations. Thus, for instance, *formular hipótesis* may occur as *nos lleva a formular la siguiente hipótesis* 'leads us to formulate the following hypothesis', *formulen y revisen sus hipótesis* 'formulate and revise their hypotheses', etc.
In order to extract the relevant information from such highly variable configurations we have resorted to the syntactic annotation of the expert subcorpus. We have extracted dependency triples "relation(head,dependent)" of the following relations: "amod(*estudio* 'study', *prospectivo* 'prospective')", "obj(*base* 'foundation', *sentar* 'lay')" and "nsubj(*consenso* 'consensus', *existir* 'exist')". Currently, we are running tests with several association measures (t-score, pointwise mutual information, etc.) in order to determine which one performs better in identifying collocations from the corpus. Once the novice corpus is completed, our intention is to extract the same types of ACLs in the novice writer subcorpora as well and compare their use in both types of writers in order to discover differences between them and know better the needs of the possible users of the tool.

4. Incorporating CLA's into HARTA

After having been extracted from the expert subcorpus, CLA candidates will be examined by lexicographers who will decide on their inclusion in the tool. In order to facilitate the examination of candidates, two applications have been developed: one for the treatment of collocations and another for idioms and formulas. In the case of collocations, a list of candidates extracted as explained in Section 3.3 is displayed, so that the lexicographer can easily select a particular candidate for its inclusion in HARTA or discard it. Collocations are sorted by their base, and collocates of the same base are displayed in decreasing order of association strength – as determined by an association measure. For each collocation candidate the sentences in which it occurs in the corpus are also displayed so that lexicographers can select representative examples of its use (Figure 2). After being revised by the lexicographers' team, collocations will be sent to HARTA and to the *Diccionario de colocaciones del español* (Alonso-Ramos, M., 2004).
The treatment of formulas and idioms will be slightly different. First, the lexicographers will encounter lists of n-

grams and will be able to choose among the association measures mentioned in Section 3.2 to sort the candidates.

Given the apparent interaction of association measures and n-gram length (cf. Table 2 above), this feature of the application seems particularly interesting. Once the lexicographers decide that a given n-gram qualifies either as a formula or an idiom, they will proceed to manually edit them. Formulas will be assigned a rhetorical function (e.g. presenting conclusions, expressing a contrast, quoting other pieces of research, etc.; see Figure 3, field FUNCIÓN DISCURSIVA). Likewise, possible variants of one formula will have to be introduced manually. Such variants are cases where two or more n-grams show slight formal differences, but perform the same rhetorical functions (e.g. *como se ha dicho más arriba/como hemos dicho más arriba* 'as has been said before/as we have said before'). Since determining whether two n-grams perform the same rhetorical function or not will in all probability require manual analysis of concordances (Salazar, 2014), this process cannot be automatized.

Idioms will be given part-of-speech information (in the field CLASE DE PALABRA). In principle, it could be problematic to define idioms in terms of a single part-of-speech, since at some level they have internal phrase structure. In terms of their meaning, however, they behave as single lexical units and syntactically they can be assimilated to adjectives, adverbs, prepositions, etc. (Mel'čuk, 2006). Idioms' part-of-speech in HARTA will reflect their behavior as blocks, rather than their internal structure.

5. Conclusion

The present paper describes the treatment and exploitation of a corpus compiled as a data source for a dictionary cum writing-aid directed to writers of academic texts in Spanish (HARTA). The project is currently ongoing research. So far, the expert subcorpus has been marked-up, part-of-speech tagged and parsed according to the design described above. The novice writers subcorpus is on the process of being compiled and marked-up.

The expert subcorpus has provided material for the HARTA: an AWL and an idiom list of Spanish have been extracted which provide the candidates for entry headwords of the dictionary. Likewise, we extracted a set of collocations according to the methods indicated in Section 3.3 and containing lemmas of the AWL, taking advantage of the annotation and parsing of the expert subcorpus. Future research will include the manual revision of these results in order to include them in HARTA with different entry structures depending on their phraseological status. Additionally, studies comparing the use of phraseology by experts and novice writers will be carried out with a view to knowing better the needs of the latter group when writing academic texts.

We plan to make the tool accessible online once it is completed. The users of this tool will not only have access to a writing aid, but also to the expert and novice writers' subcorpora through the tool interface.

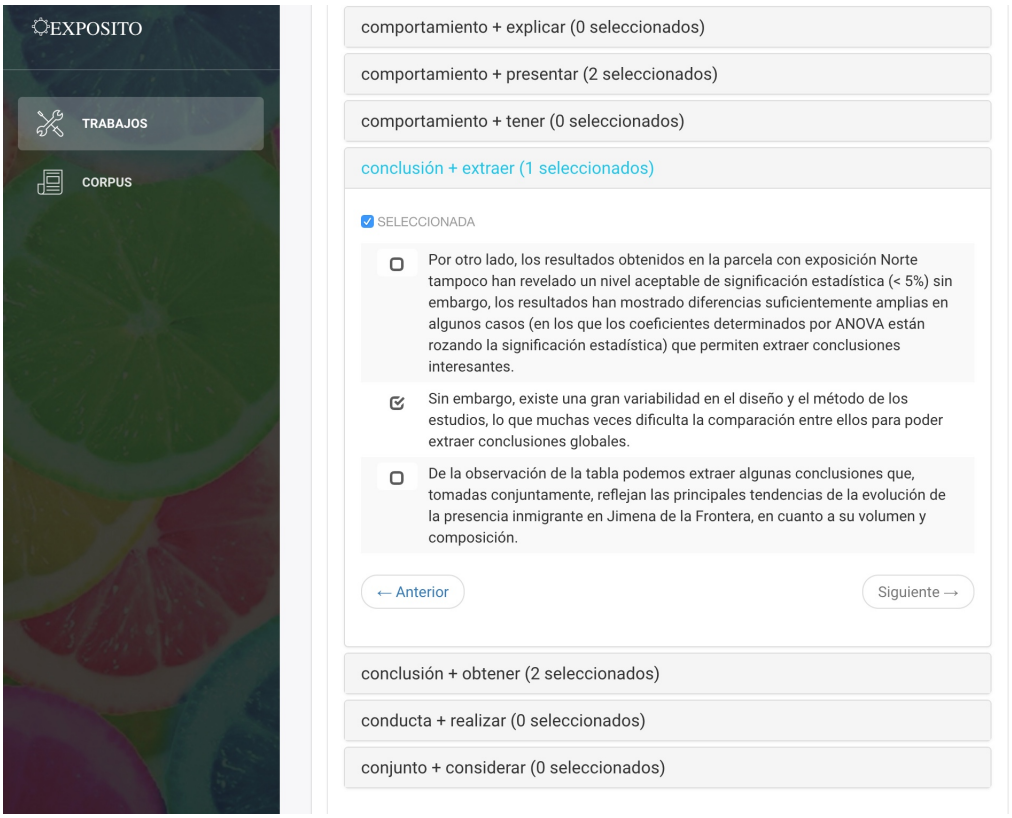


Figure 2: Editing collocations

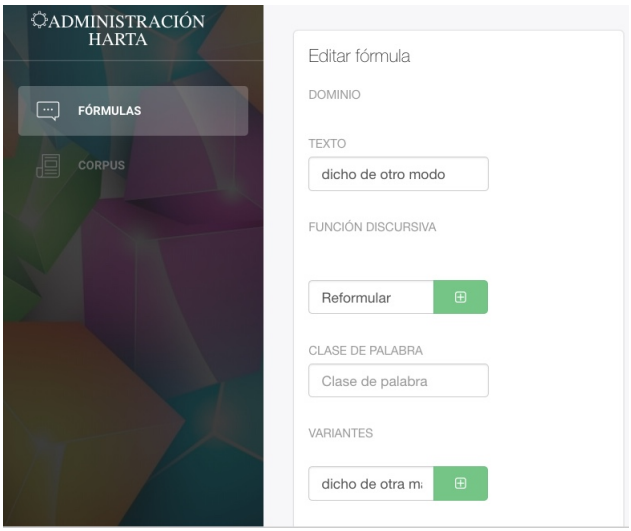


Figure 3: Editing idioms and formulas

6. Acknowledgements

The work presented in this paper has been partially supported by the Spanish Ministry of Economy and Competitiveness (MINECO) and FEDER Funds of the European Commission under the contract number FFI2016-78299-P and the post-doctoral grant Juan de la Cierva (IJCI-2016-29598), and by the Galician Government (post-doctoral grant ED481D 2017/009; RELEX network ED431D R2016/046).This publication reflects the views only of the authors, and InterLAE cannot be held responsible for any use which may be made of the information contained herein.

7. Bibliographical References

Alonso-Ramos, M., García-Salido, M., and Garcia, M. (2017). Exploiting a Corpus to Compile a Lexical Resource for Academic Writing : Spanish Lexical Combinations. In I. Kosem, et al., editors, *Proceedings of 2017 eLex Conference*, pages 571–586, Leiden. eLex.

Appel, R. and Trofimovich, P. (2017). Transitional probability predicts native and non-native use of formulaic sequences. *International Journal of Applied Linguistics*, 27:1–20.

Bestgen, Y. (2014). Extraction automatique de collocations : Peut-on étendre le test exact de Fisher à des séquences de plus de 2 mots ? In *12es Journées internationales d'Analyse statistique des Données Textuelles*, Paris.

Biber, D., Finegan, E., Johansson, S., Conrad, S., and Leech, G. (1999). *Longman Grammar of Spoken and Written English*. Longman, 1 edition.

Callies, M. and Zaytseva, E. (2013). The Corpus of Academic Learner English (CALE). *Dutch Journal of Applied Linguistics*, 2(1):126–132.

Coxhead, A. (2000). A new academic word list. *TESOL Quarterly*, 34(2):213–238.

D'Hertefelt, M., De Wachter, L., and Verlinde, S. (2014). Writing aid dutch. supporting students' writing skills by means of a string and pattern matching based web application. In Susan Zvacek, et al., editors, *Proceedings of the 6th International Conference on Computer Supported Education: Vol. 1*, pages 486–491. SCITEPRESS.

Garcia, M. and Gamallo, P. (2016). Yet another suite of multilingual NLP tools. In J. P. Leal J. L. Sierra-Rodríguez et al., editors, *Languages, Applications and Technologies. Communications in Computer and Information Science*, pages 65–75. Springer, Cham.

Gardner, S. and Nesi, H. (2013). A classification of genre families in university student writing. *Applied Linguistics*, 34(1):25–52.

Granger, S. and Paquot, M. (2015). Electronic lexicography goes local: Design and structures of a needs-driven online academic writing aid. *Lexicographica: international annual for lexicography*, 31(1):118–141.

Gries, S. T. (2008). Dispersions and adjusted frequencies in corpora. *International Journal of Corpus Linguistics*, 13(4):403–437.

Kübler, N. and Pecman, M. (2012). The ARTES bilingual LSP dictionary. from collocation to higher order phraseology. In S. Granger et al., editors, *Electronic Lexicography*, pages 187–210. Oxford University Press, Oxford.

Lijffijt, J., Nevalainen, T., Saily, T., Papapetrou, P., Puolamaki, K., and Mannila, H. (2014). Significance testing of word frequencies in corpora. *Digital Scholarship in the Humanities*, 31:374–397.

Mel'čuk, I. (2006). Parties du discours et locutions. *Bulletin de la Société de linguistique de Paris*, 101(1):29–65.

Mel'čuk, I. (2015). Clichés, an Understudied Subclass of Phrasemes. *Yearbook of Phraseology*, 6(1):55–86.

Nivre, J., Marneffe, M.-C. D., Ginter, F., Goldberg, Y., Manning, C. D., Mcdonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666. European Language Resources Association (ELRA).

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In Nicoletta Calzolari, et al., editors, *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC2012)*, pages 2473–2479. European Language Resources Association (ELRA).

Paquot, M. and Bestgen, Y. (2009). Distinctive words in academic writing: a comparison of three statistical tests for keyword extraction. In Andreas H. Jucker, et al., editors, *Corpora: Pragmatics and Discourse*, pages 247–269, Amsterdam. Rodopi.

Paquot, M. (2010). *Academic vocabulary in learner writing*. Continnuum, London.

Salazar, D. (2014). *Lexical Bundles in Native and Non-native Scientific Writing: Applying a Corpus-based Study to Language Teaching*. Studies in corpus linguistics. John Benjamins Publishing Company, Amterdam/Philadelphia.

Sebastián-Gallés, N., Martí, M. A., Carreiras, M. F., and

Vega, F. C. (2000). *LEXESP: léxico informatizado del español*. Edicions de la Universitat de Barcelona.

Straka, M., Hajic, J., and Straková, J. (2016). Udpipe: Trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666. European Language Resources Association (ELRA).

Swales, J. and Feak, C. (2004). *Academic Writing for Graduate Students: Essential Tasks and Skills*. Academic Writing for Graduate Students. University of Michigan Press.

Tutin, A. and Kraif, O. (2016). Routines semantico-rhétoriques dans l'écrit scientifique de sciences humaine: l'apport des arbres lexico-syntaxiques récurrents. *Revue de linguistique et de didactique des langues*, 53:119–141.

UNESCO. (1978). Recommendation concerning the international standardization of statistics on science and technology. Technical report, UNESCO.

8. Language Resource References

Alonso-Ramos, M. (2004). *Diccionario de colocaciones del español*. http://www.dicesp.com/paginas.

InterLAE Research Group. (2008). *Spanish English Research Article Corpus*. http://www.interlae.com/serac.

Framing Named Entity Linking Error Types

Adrian M.P. Braşoveanu*, Giuseppe Rizzo‡, Philipp Kuntschik†, Albert Weichselbraun†, Lyndon J.B. Nixon*
* MODUL Technology GmbH,
Am Kahlenberg 1, 1190 Vienna, Austria
‡ ISMB, Via Pier Carlo Boggio 61, 10138 Torino, Italy
† Swiss Institute for Information Research, University of Applied Sciences Chur,
Pulvermühlestrasse 57, 7000 Chur, Switzerland
{adrian.brasoveanu,lyndon.nixon}@modul.ac.at
giuseppe.rizzo@ismb.it
{philipp.kuntschik,albert.weichselbraun}@htwchur.ch

Abstract

Named Entity Linking (NEL) and relation extraction forms the backbone of Knowledge Base Population tasks. The recent rise of large open source Knowledge Bases and the continuous focus on improving NEL performance has led to the creation of automated benchmark solutions during the last decade. The benchmarking of NEL systems offers a valuable approach to understand a NEL system's performance quantitatively. However, an in-depth qualitative analysis that helps improving NEL methods by identifying error causes usually requires a more thorough error analysis. This paper proposes a taxonomy to frame common errors and applies this taxonomy in a survey study to assess the performance of four well-known Named Entity Linking systems on three recent gold standards.
Keywords: Named Entity Linking, Linked Data Quality, Corpora, Evaluation, Error Analysis

1. Introduction

A Named Entity Linking (NEL) system identifies, classifies and links entity mentions from a text to their Knowledge Base (KB) references. A NEL system can also be used for extracting factual knowledge from a text in order to use it for Knowledge Base Population (KBP). The typical components of a NEL system reflect the logical steps of identifying and linking the entities: identification and classification (to a defined type like person, organization or location) of the entity mentions in a text (Named Entity Recognition and Classification or NERC), linking to the referent KB, and clustering of the remaining unlinked entities (Ji and Nothman, 2016). Some of these steps might be performed jointly if the chosen architecture supports it (e.g., neural architectures). Numerous current architectures used for NEL systems include: *(i)* graph-based disambiguation which uses the links between the entities found in the text in order to rank the best candidates; *(ii)* statistical disambiguation typically focused on classic machine learning algorithms or heuristics (e.g., SVM, Conditional Random Fields, etc.) or *(iii)* neural models (e.g., LSTM). When evaluating a system, at least three other components are needed: a dataset (usually a labeled corpus or gold standard), a certain KB version (e.g., DBpedia 2016-04), and a scorer that computes measures such as accuracy, precision, recall, and F-measure.

These measures compare the NEL systems quantitatively, but do not help system designers improve their approaches. If, for example, one wants to improve the precision of a system because it affects its perceived quality, the number of false positives needs to be decreased. Reducing the number of errors (false positives and false negatives) requires a thorough analysis of the evaluation results. A good method to do this is to use the primary error analysis results from the TAC-KBP inspired neleval[1] scorer (Hachey et al., 2014), which is limited to indicating whether the returned results are correct, incorrect, extra (i.e. a named entity does not occur in the gold standard) or missing. If possible, a more detailed explanation of each error should be included, as it could potentially lead to the rapid updating of systems, datasets and KBs, especially since all these components could potentially trigger errors during an evaluation. In fact, since some of these components will not necessarily be under the system developer's control as gold standards, KBs, or scorers are probably developed by third-parties, we might argue that such an explanation is not only beneficial, but that it should become standard practice.

Given the increasing complexity of the NEL systems and evaluations, as a first step towards an automated error classification system, we propose a taxonomy focused around a set of types (e.g., Knowledge Base, Dataset, Annotator, NIL Clustering, Scorer, etc.) and sub-types (causes) as reflected through the various steps of a NEL system (e.g., partial match, wrong link returned due to KB redirect). Such a classification effort allows us to perform multiple tasks when processing the results of the error analysis: *(i)* create a transparent and reproducible method to publish detailed evaluation results[2] on top of the well-known TAC-KBP scorer; *(ii)* evaluate and improve the quality of the labelled data (e.g., Knowledge Base dump, gold standards) used in the evaluation; *(iii)* help NEL system designers to improve their systems by fixing the errors at the level where they are produced (e.g., we can report the Knowledge Base or gold standard errors to the creators of these resources).

The remainder of this paper is organized as follows: Section 2. discusses related work; Section 3. presents the fram-

[1] https://github.com/wikilinks/neleval

[2] The annotation guideline can be found at https://github.com/modultechnology/nel_errors/tree/master/guideline.

ing of the problem and the reasoning process behind the proposed taxonomy; Section 4. applies this taxonomy in a survey study aimed at analyzing different gold standards and NEL systems using DBpedia as the referent KB; and Section 5. concludes the paper and outlines future research.

2. Related Work

TAC-KBP challenges (Ji and Nothman, 2016) are focused on the rapid prototyping of NEL systems for various languages. Each participant needs to submit one or several runs of their system and write a paper. In addition to the new yearly datasets, TAC-KBP participants are given access to previous datasets for training purposes. The event's annual overview reviews the best approaches and shortlists the challenges to be solved for the next years. The goal is to find approaches to solve these errors in next year's competition. In the 2016 edition's overview (Ji and Nothman, 2016), ample space was given to the approaches used for trilingual knowledge transfer, weak/strong mention boundary detection and to the within-document and cross-document coreference resolution error propagation, whereas type discovery (e.g., discovering entities that are not in the current schema), massive multilingual Entity Discovery and Linking (e.g., for hundreds of languages instead of just three) or streaming data were considered as challenges for 2017. It has to be noted that one of the most popular scorers used today (neleval) is based on the lessons learned from the TAC-KBP challenges (Hachey et al., 2014). Radford (Radford, 2015), a former TAC-KBP participant, analyzed his systems and presented types of errors encountered in them, but tailored its taxonomy of errors around the peculiarities of each system.

Issues observed in well-known gold standards are reported in (van Erp et al., 2016), but the results are described through a general set of features (such as dataset types, confusability, prominence) and not through the results obtained by different NEL systems. The paper also notes that the efficacy of NEL evaluations needs to be improved by removing the dependency on black-box evaluations (e.g., evaluations in which the participants only see global results and not mention-level results) as they do not allow to improve upon the results. We address this research challenge by building our experiments on top of the TAC-KBP scripts and evaluation format (Hachey et al., 2014) to provide a more detailed error analysis.

In (Heinzerling and Strube, 2015), authors present a system for performing visual error analysis built on top of the TAC-KBP evaluation formats, but it only supports a subset of the error types covered by our method.

In (Cornolti et al., 2013), authors define a set of annotation tasks (such as Annotate to Wikipedia – A2W, Concepts to Wikipedia – C2W) and propose an automated evaluation system that measures per-task performance. A sequel to Cornolti's work, GERBIL (Usbeck et al., 2015) is a large-scale evaluation system that allows us to compare the output of different NEL systems. It is typically used as an alternative to TAC-KBP style evaluations in literature. GERBIL uses gold standards in the NIF format (Natural Language Processing Interchange Format), an RDF format designed to allow the sharing of both textual and annotation resources and ease the interplay between NLP tools. A basic error analysis based on Gerbil can be performed with EAGLET, but it is only focused on seven error types which can be classified as dataset errors (Jha et al., 2017) and does not include other large error classes. The EAGLET pipeline contains a preprocessing module and a rule-based module for identifying dataset errors, resulting annotations being reviewed by a human annotator. It has to be noted that similarly to our approach, the observed errors are annotated and judged by humans. Two of the gold standards we experiment in our work (KORE50 and Reuters128) are also integrated in GERBIL[3].

3. Classification of Errors in NEL Evaluations

Current NEL scorers, such as GERBIL or TAC-KBP do not allow to perform a closer inspection and framing of the evaluated NEL annotations. While GERBIL does not provide any mechanism through which to access the errors at mention level, the TAC-KBP results can be processed to obtain a primary error analysis limited to the validity of the results (e.g., results are marked as wrong link, extra link or missing). While such information is valuable, adding a semantic layer on top of it can offer researchers and developers the insights they need to improve their tools. In order to capture the logical reasoning that has produced the error we have considered the following error types:

KB A Knowledge Base error is an error discovered in a particular KB version (e.g., DBpedia 2015-10). May include wrong mappings (e.g., a person's name that points to a year) or missing entities.

DS Dataset errors are typically the errors produced by the human annotators during the annotation process which can still be found in a certain gold standard version. Most common DS errors are missing or wrong annotations (e.g., incomplete surface form).

AN Annotator errors refer to the output of an automated annotation system that follows the classic NEL phases (e.g., NERC, linking, relation extraction or graph disambiguation). This is the largest error class and includes errors like wrong type, wrong link, wrong surface form, etc.

NIL NIL Clustering errors refer to the output of the NIL Clustering components. Known errors include missing surface forms for some of the entities or new links in recent version of the KB.

SE Scorer or evaluation errors explain the errors reported by an evaluation script when the AN and DS outputs are similar. These types of errors are extremely hard to spot. A common error is related to how redirect links are scored.

We have started our investigations by examining the output of the TAC-KBP scorer. We have followed the strategy of collecting and verifying the mentions (surface forms), types

[3] http://gerbil.aksw.org/gerbil/config

NEL System			Gold Standard		Error	
Entity Link$_s$	**ET$_s$**	**SurfaceForm**	**Entity Link$_g$**	**ET$_g$**	**Type**	**Cause**
dbr:Bruce_Willis	ORG	expiration	-	-	KB	Redirects
de.dbr:2009	LOC	2009	-	-	KB	Wrong Type
dbr:United_States	LOC	U.S.	-	-	DS	Missing Annotation
dbr:New_York_City	LOC	New York	dbr:New_York	LOC	DS	Wrong Annotation
de.dbr:Berlin	LOC	Berlin	dbr:Berlin	LOC	DS	Different Language
dbr:JFK	PER	Kennedy	dbr:JPK	PER	AN	Same-Type
dbr:Beck	ORG	Beck	dbr:Jeff_Beck	PER	AN	Cross-Type
dbr:Barack_Obama	PER	Malia Obama	NIL	PER	NIL	Wrong Cluster
NIL	ORG	Knicks	dbr:New_York_Knicks	ORG	NIL	Partial Match
dbr:Miles_Davis	PER	Davis	dbr:Miles_davis	PER	SE	Correct Redirect

Table 1: Examples of the most common errors detected in the three gold standards investigated in this paper. The entries that have gold links were marked as *wrong-link* and the others as *extra* in TAC-KBP primary error analysis. ET represents the entity type. Subscripts *s* and *g* denote the *system* (annotator) and *gold standard* (dataset).

and links for all entities present in a text. The template used for describing a new error cause in the *Annotation Guideline* contains the **scope** (mention, type, link), **similarity** to other error causes, a **general description** of the error, **examples** and **comments**. We aim to improve this template in time using community feedback. Our initial goal was simply to find an easy way to report such errors through a method that would later allow us to easily classify them.

Based on the experiments performed in Section 4., we have defined error causes for each error class as illustrated in Table 1. In a first phase we have focused on the first two large classes: KB and DS in order to remove any doubts related to the AN or NIL errors.

A typical **KB** error looks like the entity *de.dbr:2009* that has been marked as a location in the German DBpedia version 2015-10 (Table 1) or the surface form *expiration* that was tagged with *dbr:Bruce_Willis* due to a KB redirect. A lot of the errors that occur on this level are simply due to the fact that most systems do not use the live versions of the KB but rather dumps that are published at certain intervals (e.g., DBpedia dumps are published every 6 months, whereas Wikidata dumps are published weekly). It is quite often the case that missing links from a previous dump were updated in the meantime.

The **DS** errors are generally instances of wrong annotations due to various causes: typos (we found many cases in which dots were missing from geographic abbreviations), a different language than the target one (e.g., German DBpedia instead of English), partial matches (e.g., geo entities missing parts of their name). DS errors are perhaps the hardest to agree on as each gold standard could have different annotation guidelines. However, we think these must be judged both against the original guideline that was used to create them, but also using common sense, especially if there is an intention of integrating multiple datasets in a single evaluation tool (e.g., as it was done with GERBIL).

AN errors include abbreviation conflicts (e.g., for *Kent.* abbreviation, the annotator returns *Kent, UK* instead of *Kentucky*), same-type disambiguation errors (e.g., *Bill Clinton* returned instead of *Bill Gates* or *Hillary Clinton*), cross-type disambiguation errors (e.g., when an entity with a different type is returned), or generic terms (e.g., when words like *ship* or *Admiral* are returned instead of the real entities that are near them, like *Hansa Stavanger container ship* or *Admiral Thad W. Allen*). Largely the AN errors depend on the algorithms and settings that were chosen for a specific tool. These kind of errors can only be removed by fixing the tool.

NIL clustering errors include entity mentions being shared among multiple clusters (e.g., role/title or last name appearing in a different cluster than the full name of a person) or partial macthes (e.g., *Knicks* used for *N.Y. Knicks*). Or clusters that are generated of lexical equal entities but they hold different semantic meanings in the context they are used. This large error class is, unfortunately, dependent on the annotation system. Early systems like those presented in Radford's work (Radford et al., 2011) were known to be sub-optimal due to their lack of integration with the linking process, whereas more recent implementations are integrative and also include advanced co-reference resolution algorithms as described in recent TAC-KBP initiatives (Ji and Nothman, 2016).

While **SE** errors are not as frequent like the other categories, a classic example is represented by correct redirects not counted as such (see the Miles Davis row in Table 1). In order to correctly identify such errors there is a need to compare the results of different evaluation tools on the same datasets, but this goes beyond the purpose of the current paper.

It has to be noted that in some cases an error can appear due to multiple causes (e.g., a partial match causes a different entity to be returned by the NEL system due to a wrong KB redirect - this case offering both an AN and a KB error). Such cases are of course hard to interpret correctly, but we have chosen to follow the logical order: KB - DS - AN - NIL - SE and try to always place the error on the first layer on which it can occur. This might not always be optimal, but it should help developers better reason about how these errors are produced.

The reasoning process used to identify such cases starts with reading the text in order to understand the context. Then the surface forms, types and mentions that are present

in AN and DS results are examined in order to find the most likely error cause (e.g., surface form does not seem to have any connection with the returned entity, wrong link or wrong type). If the error cause cannot be found in AN or DS results, then the KB entry is examined for additional clues. This reasoning chain is repeated until a good error cause can be determined, and if the error cause is not present in the Annotation Guideline, a new entry is added to this guideline and is formatted according to the proposed template.

4. Experimental Setup

A set of experiments was performed on three gold standards with four NEL systems in order to better understand the reasoning used for explaining the errors, the agreement among human annotators who evaluated the system results, and the feasibility of creating an automatic semantic error analysis system for NEL evaluations.

4.1. Datasets and Tools

We have only selected datasets that were known to have been annotated manually and were published in the NIF format. If the dataset publication mentioned that it was created automatically, it had only one annotator or is rather a baseline than a gold standard, we have not included it in our experiment. While it can be argued that the methodology described in the previous section can also be applied for such datasets, we thought it is best to first select datasets that match our criteria for reasonable gold standards. We consider extending this methodology for any type of datasets in future work.

We have applied the described methodology for identifying and classifying the error classes and error types introduced in the previous section to evaluations performed on four state-of-the-art off-the-shelf NEL systems, namely DBpedia Spotlight (Daiber et al., 2013), Babelfy (Moro et al., 2014), AIDA (Hoffart et al., 2011), and Recognyze (Weichselbraun et al., 2015) while annotating three well-known benchmark datasets, namely Reuters128 (Röder et al., 2014), KORE50 (Hoffart et al., 2012), and RBB150 (Brasoveanu et al., 2016).

AIDA and Babelfy are graph-disambiguation frameworks, DBpedia Spotlight is a statistical disambiguation framework, whereas Recognyze used heuristics at the time of the experiments. We have selected two graph-disambiguation frameworks simply because it seemed to be the best paradigm for performing NEL at the time.

KORE50 consists of English sentences from five different domains, Reuters128 includes full news media articles in English, and RBB150 contains German television subtitles. To reduce the manual annotation workload, we processed a subset of documents for each corpus and focused only on the false positives.

4.2. Experimental Methodology

We have then used the four NEL systems in order to automatically annotate the first 50 texts from each gold standard and collect the false positives that could signal eventual problems. DBpedia Spotlight was the only annotator used for both languages, while Recognyze was only used for German. The gold standards were converted into the TAC-KBP format from the NIF format (Hellmann et al., 2012), a format that allows for easy interchange of NLP data (e.g., gold standard annotations, NEL system results). The TAC-KBP scorer (Hachey et al., 2014) was used for the evaluations. A set of runs with the webservices of the investigated NEL systems was produced for each gold standard. Only three types presented in TAC-KBP 2014 evaluations were selected: Person, Organisation and Location as we considered that the systems were already well-trained to handle them. TAC-KBP 2015 has introduced the convention of splitting the Location class into separate classes for GPE, Location and Facility, but all the other challenges still consider only few types. The types of the entities were inferred from the DBpedia links returned by the systems, as not all systems return the types directly. We have chosen to keep all well-known classes for the three entity types as they appear in DBpedia, YAGO and schema.org ontologies and have not included fine-grained typing in the current version. While fine-grained typing is in our research agenda, our current goal is to refine this methodology based on feedback received from third-party users.

We checked all the entities that were marked as having no other type than *owl:Thing* and discovered that in some cases they represented the merging of multiple entities (e.g., *Kenneth and Mamie Clark*), family names (e.g., *dbr:Reuter*), redirects (e.g., *Carnival Cruise Lines*) or even annotations in other languages (e.g., links from German DBpedia in an English corpora). Only entities that had the three types we were aiming for (person, location, organization) or were annotated to different languages in the gold standard were considered. We have then ran the TAC-KBP scorer and collected the results and the false positives from the primary error analysis (the classification of links as correct, wrong, missing or extra returned by neleval) for each run.

We automatically created supersets with all the identified errors available for a particular gold standard. Such a superset included all the data related to the errors available from the NEL systems and gold standard, each error being identified by the mention, type and link from both the AN and DS (if available), but also by several fields that we have later used to create our error annotations (e.g., fields like document, span, error type, error cause or presence in the gold standard). These supersets were annotated independently by two human annotators and inter-annotator agreement scores were computed. A third human annotator went through the results along with the two human annotators and resolved the inconsistencies. While creating human annotations is costly, we considered it a necessary step in order to validate our taxonomy, but also to create a gold standard that can be used towards the automated classification of errors.

All the errors were annotated with respect to single-language evaluations (e.g., English, German). While multilingual evaluations start to become more common, there are less datasets for such tasks available, and most of them are rather baselines then gold standards.

Along with this paper we also publish the human annotation guidelines to foster their reuse, to promote common annotation standards and to advance the state of the art in

Dataset	Spotlight	Babelfy	AIDA	Recognyze
Reuters128	125	172	102	-
KORE50	23	18	44	-
RBB150	113	-	-	65

Table 2: False Positive counts returned by the investigated NEL systems.

Dataset	κ	FPs	KB	DS	AN	SE
Reuters128	0.653	302	9	42	251	0
KORE50	0.689	59	2	1	55	1
RBB150	0.877	176	2	70	104	0

Table 3: Total count of False Positives (FPs) and error types (KB, DS, AN, SE) in all systems.

NEL error analysis[4].

4.3. Results and Discussion

The examples presented in Table 1 were collected during the experiments. Table 2 shows a quantitative analysis of false positives generated by the NEL system. Table 3 reports on the inter-annotator agreement Fleiss's κ (Fleiss, 1971). The Fleiss's κ agreement figures show that there is a high agreement between the human annotators (a value between 0.61 and 0.80 denotes substantial agreement, while a value between 0.81 and 1.00 denotes an almost perfect agreement). NIL Clustering errors were not taken into account in this experiment due to the fact that the examined systems included no such components. As it can be seen only one gold standard can be considered as having high-quality annotations (KORE50), due to its low number of KB errors and high agreement with annotations provided by state-of-the-art NEL tools. The Reuters 128 dataset that is filled with many popular entities, in contrast, suffers from numerous redirects and multiple surface forms that lead to the largest number of KB errors.

The RBB150 gold standard contains the largest number of errors overall, although many of these errors are caused by partial matches due to the fact that annotations for person-type entities included their roles (e.g., President Barack Obama), whereas most NEL systems returned these entities without roles (e.g., Barack Obama), therefore the dataset's annotation rules should be changed to be more in line with other datasets or provide different settings for full and partial matches.

Both Reuters128 and RBB150 have cases of entities annotated in a different language than the original language of the processed text (e.g., German annotations in an English text). This should be considered an error only in a single language evaluation, whereas in a multilingual evaluation such output is not only desirable, but it is encouraged.

We have noticed that entities that are classified as Person lead to a substantial agreement among the NEL systems, while entities that are classified as Organization are often annotated with their full suffixes in the gold standards, but recognized without them. Location proved to be the source of many inconsistencies in gold standards and KB, due to demonyms automatically annotated to countries (therefore Annotator errors) or lack of clear rules for annotating long names (e.g., Columbus, OH might be a single entity, while for Rome, Italy there will be two annotations).

A good application that can help improve the quality of semantic data is the rapid publishing of the evaluation results. Before turning the output into NIF and reporting issues to KB or DS designers, we advise practitioners to establish several rules upfront, for example to clarify how many tools and human annotators would need to be in agreement. While we recommend at least an agreement between 75% of the tools and two human annotators, the final criteria will always depend on the use case and end goals of system or challenge designers.

5. Conclusion

In this paper we described a taxonomy to identify errors in gold standards and errors generated by NEL systems. The taxonomy has been tested in an experimental environment that has involved two human annotators. A manual annotation and classification of the errors identified in the evaluation results demonstrates the usefulness and potential of the suggested schema for identifying error classes and improving the underlying datasets, KBs and NEL components.

While the taxonomy and the evaluation method presented here are still in an early stage (e.g., only false positives were considered and the focus was mostly on the big error classes), a few applications already show lots of promise in using such an inductive data-driven approach of fixing gold standards, knowledge bases, NEL system annotations, and overall providing a better support to error analyses for NEL evaluations.

Using human annotators in order to annotate the errors is a current limitation of the method presented in this paper due to time consumption and costs, but it was necessary as a first step towards automating error analysis. The agreement scores denote the fact that the method can be widely deployed and by converting these annotations in the NIF format we can share our results with the maintainers of KBs and gold standards in order to help them to improve their services.

The main advantage of this method is the fact that it allows us to identify and explain most of the large error classes as long as the NEL systems considered include the conventional components (even if some of these components are merged). This work can also be adapted for different NEL evaluation systems besides TAC-KBP (e.g., GERBIL) with the condition to be able to access the NEL system runs and the gold standards.

6. Acknowledgements

The InVID project (http://www.invid-project.eu/) has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 687786. The Job-Cockpit project (www.htwchur.ch/job-cockpit) was funded by the Swiss Commission for Technology and Innovation (CTI).

[4] https://github.com/modultechnology/nel_errors

7. Bibliographical References

Brasoveanu, A., Nixon, L. J. B., Weichselbraun, A., and Scharl, A. (2016). A regional news corpora for contextualized entity discovery and linking. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.*, pages 3333–3338.

Cornolti, M., Ferragina, P., and Ciaramita, M. (2013). A framework for benchmarking entity-annotation systems. In Daniel Schwabe, et al., editors, *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 249–260. International World Wide Web Conferences Steering Committee / ACM.

Daiber, J., Jakob, M., Hokamp, C., and Mendes, P. N. (2013). Improving Efficiency and Accuracy in Multilingual Entity extraction. In Marta Sabou, et al., editors, *I-SEMANTICS 2013 - 9th International Conference on Semantic Systems, ISEM '13, Graz, Austria, September 4-6, 2013*, pages 121–124. ACM.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382.

Hachey, B., Nothman, J., and Radford, W. (2014). Cheap and easy entity evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 464–469. The Association for Computer Linguistics.

Heinzerling, B. and Strube, M. (2015). Visual error analysis for entity linking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, System Demonstrations*, pages 37–42. ACL.

Hellmann, S., Lehmann, J., Auer, S., and Nitzschke, M. (2012). NIF combinator: Combining NLP tool output. In Annette ten Teije, et al., editors, *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, volume 7603 of *Lecture Notes in Computer Science*, pages 446–449. Springer.

Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 782–792. ACL.

Hoffart, J., Seufert, S., Nguyen, D. B., Theobald, M., and Weikum, G. (2012). KORE: keyphrase overlap relatedness for entity disambiguation. In Xue-wen Chen, et al., editors, *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 545–554. ACM.

Jha, K., Röder, M., and Ngomo, A. N. (2017). All that glitters is not gold - rule-based curation of reference datasets for named entity recognition and entity linking. In Eva Blomqvist, et al., editors, *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 305–320.

Ji, H. and Nothman, J. (2016). Overview of tac-kbp2016 tri-lingual edl and its impact on end-to-end kbp. In *Eighth Text Analysis Conference (TAC)*. NIST.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Radford, W., Nothman, J., Curran, J. R., Hachey, B., and Honnibal, M. (2011). Naïve but effective NIL clustering baselines - CMCRC at TAC 2011. In *Proceedings of the Fourth Text Analysis Conference, TAC 2011, Gaithersburg, Maryland, USA, November 14-15, 2011*. NIST.

Radford, W. (2015). *Linking Named Entities to Wikipedia*. Ph.D. thesis, School of Information Technologies, Faculty of Engineering and IT, The University of Sydney.

Röder, M., Usbeck, R., Hellmann, S., Gerber, D., and Both, A. (2014). N^3 - A collection of datasets for named entity recognition and disambiguation in the NLP interchange format. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 3529–3533.

Usbeck, R., Röder, M., Ngomo, A. N., Baron, C., Both, A., Brümmer, M., Ceccarelli, D., Cornolti, M., Cherix, D., Eickmann, B., Ferragina, P., Lemke, C., Moro, A., Navigli, R., Piccinno, F., Rizzo, G., Sack, H., Speck, R., Troncy, R., Waitelonis, J., and Wesemann, L. (2015). GERBIL: General entity annotator benchmarking framework. In *Proceedings of the 24th International Conference on World Wide Web, WWW 2015*, pages 1133–1143.

van Erp, M., Mendes, P. N., Paulheim, H., Ilievski, F., Plu, J., Rizzo, G., and Waitelonis, J. (2016). Evaluating entity linking: An analysis of current benchmark datasets and a roadmap for doing a better job. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.*, pages 4373–4379.

Weichselbraun, A., Streiff, D., and Scharl, A. (2015). Consolidating Heterogeneous Enterprise Data for Named Entity Linking and Web Intelligence. *International Journal on Artificial Intelligence Tools*, 24(2):1–31.

A FrameNet for Cancer Information in Clinical Narratives: Schema and Annotation

Kirk Roberts[1], **Yuqi Si**[1], **Anshul Gandhi**[1], **Elmer V. Bernstam**[1,2]
[1] School of Biomedical Informatics
[2] Division of General Internal Medicine, Department of Internal Medicine, McGovern Medical School
The University of Texas Health Science Center at Houston
Houston, TX USA
kirk.roberts@uth.tmc.edu

Abstract

This paper presents a pilot project named Cancer FrameNet. The project's goal is a general-purpose natural language processing (NLP) resource for cancer-related information in clinical notes (i.e., patient records in an electronic health record system). While previous cancer NLP annotation projects have largely been ad hoc resources to address a specific and immediate information need, the frame semantic method employed here emphasizes the information presented in the notes themselves and its linguistic structure. To this end, three semantic frames (targeting the high-level tasks of cancer diagnoses, cancer therapeutic procedures, and tumor descriptions) are created and annotated on a clinical text corpus. Prior to annotation, candidate sentences are extracted from a clinical data warehouse and de-identified to remove any private information. The frames are then annotated with the three frames totaling over thirty frame elements. This paper describes these steps in the pilot project and discusses issues encountered to evaluate the feasibility of general-purpose linguistic resources for extracting cancer-related information.
Keywords: clinical information extraction, cancer, frame semantics

1 Introduction

Important medical information about cancer patients is often only available in free text (natural language) notes in electronic health records (EHRs). This information is frequently needed for research, quality improvement, surveillance, and other important functions. However, the manual abstraction of this information can be incredibly time-consuming and expensive, often making it infeasible for both early-stage research and clinical quality improvement projects. Thus, natural language processing (NLP) approaches can offer a tremendous service in oncology.

Existing NLP systems for cancer-related information generally fall under a type of biomedical NLP task known as *phenotyping*, which is the task of identifying patients that meet a certain set of criteria. Phenotyping specifications are often highly task-specific, which frequently yields one-off NLP datasets and algorithms that do not generalize to similar phenotyping tasks. For example, one phenotype method may identify lung cancer patients with a tumor of at least 2cm in diameter, while another method may identify lung cancer patients with at least two tumors that are 1cm in diameter. Such methods are quite similar, yet often produce incompatible annotations and algorithms.

The key insight is that phenotyping methods often conflate extraction (identifying relevant portions of text) and reasoning (determining if the extracted text fulfills the task's needs). By separating these steps and focusing on general-purpose extraction, it will be possible to easily and rapidly develop phenotyping methods. This is because the bulk of the effort is typically spent in the extraction step (annotating data and developing NLP algorithms), while the reasoning step is often a straightforward set of rules. Continuing the example above, an NLP system capable of extracting all tumor references and their sizes from text would easily meet the needs of both phenotyping methods (along with potentially many others).

The challenge then becomes how to develop general-purpose extraction algorithms for clinical text, especially when it isn't necessarily clear a priori what information needs to be extracted. Luckily, *frame semantics* provides a useful framework for developing such a resource. In frame semantics, a word or phrase evokes a frame of semantic knowledge that describes the characteristic attributes associated with a concept. For example, for *tumor*, the frame would likely contain elements that describe the size, location, and morphology of the tumor. The set of frame elements can either be defined a priori by a subject expert or added iteratively based on the data (this work combines both approaches).

The specification of a set of frames combined with annotated examples is referred to as a FrameNet, the best known being Berkeley FrameNet (Baker et al., 1998). But other FrameNets exist as well, notably domain-specific FrameNets. This paper describes a pilot project to build such a domain-specific resource—referred to hereafter as Cancer FrameNet—that focuses on cancer-related information. The goal of the pilot is to test the feasibility of a much larger resource covering the depth and breadth of cancer information in patient records. Even as a pilot, however, this resource is still sizable, covering three important frames, 22 lexical units, and nearly 8 thousand annotated sentences.

There are several potential pitfalls for frame annotation in clinical notes, thus the need for a feasibility pilot. These issues include the consistency of cancer-related information in clinical notes: is cancer too complex with too many variables to be reliably annotated (both manually and by an automatic NLP system)? Another issue is the overlapping information of frames: are there really semantically distinct

concepts that can be formed into different frames? Finally, frame annotation is typically limited to sentence and clause context (excluding implicit information), but does this apply to cancer-related information in clinical notes?

It should be noted that there is a second major challenge in the generalizability of clinical NLP systems. This involves the portability of algorithms from one institution's clinical notes to another, as oftentimes these can be drastically different. While we acknowledge the critical importance of this problem, we make no attempt to solve it here. All data described in this paper come from a single institution. It is our hope, however, that upon establishing a general-purpose resource, annotated frames from additional institutions can be added in order to improve inter-institutional generalizability.

The rest of this paper is structured as follows: Section 2 outlines previous work in both domain-specific FrameNets and cancer-related information extraction. Section 3 describes the pre-annotation process: where the data came from, how it is extracted and prepared for annotation, including how it is de-identified to protect patient privacy. Section 4 lays out the frames covered in the pilot project: what they are, why they were chosen, and the current frame elements. Section 5 details the annotation process. Finally, Section 6 discusses the potential ramifications of Cancer FrameNet on cancer information extraction, including its strengths and weaknesses, and how it might or might not overcome some of the aforementioned pitfalls.

2 Background

The theory of frame semantics has spawned many practical natural language resources. Most prominent is the Berkeley FrameNet project (Baker et al., 1998; Baker et al., 2015). which is intended to be an open-domain encoding of common knowledge (commercial transactions, transportation, crime, international affairs, etc.). The FrameNet construction methodology has been ported to many other languages (Heppin and Gronostaj, 2014; Lin et al., 2015; Rezhake and Kuerban, 2015; Ohara, 2016). More interesting, many domain-specific FrameNets exist, ranging from soccer (Schmidt, 2006; Torrent et al., 2014) to sentiment (Ruppenhofer, 2013) to disability (Savova et al., 2005) to cellular pathways (Dolbey et al., 2006; Dolbey, 2009). No such resource targets the types of cancer-related information found in EHR notes.

While not explicitly based on a FrameNet-style approach, a significant amount of work has focused on NLP systems for extracting cancer-related information from EHRs. A sampling of the types of information extracted include: a frame-like representation of radiological findings (Taira et al., 2001); procedures, tumor stages, and various biomarker scores (Xu et al., 2004); tumor and node staging (McCowan et al., 2007); Gleason score, tumor stages, and margin status (D'Avolio et al., 2008); histology, site, dimension, and various tumor types (Coden et al., 2009); tumor progression (Cheng et al., 2010); colonoscopy status (Denny et al., 2010); colonoscopy quality measures (Harkema et al., 2010); Gleason score, Clark level, and Breslow depth (Napolitano et al., 2010); cancer history (Wilson et al., 2010); counts of examined and positive tumors and nodes (Martinez and Li, 2011); pancreatic cancer predictors (Zhao and Weng, 2011); tumor staging and biomarkers (Segagni et al., 2012); pain in prostate cancer patients (Heintzelman et al., 2013); highest level of pathology, number of removed adenomas (Imler et al., 2013); tumor, node, metastases, and ACPS stages (Martinez et al., 2013); liver cancer status (Ping et al., 2013); change of event state (Vanderwende et al., 2013); twenty-two staging indicators (Ashish et al., 2014); volume, size, and location (Wang et al., 2014); and diagnosis, hormone receptor status, tumor size, and number of positive nodes (Napolitano et al., 2016). This synopsis understates the number of extracted information types, but it is still clear that there is a significant breadth of information as well as consistent areas of overlap.

A more direct comparison to our work is the recent work in the DeepPhe project (Savova et al., 2017). DeepPhe takes a document-level approach to extracting cancer information, which is more appropriate for certain data types than the sentence-based approach proposed below. Most crucially, it is unknown how well their document-level approach generalizes to other institution's data. On the other hand, while the pilot project discussed in this paper focuses on a single institution as well, we hypothesize that a frame-based method targeting information at the sentence level will result in greater potential for generalization across institutions.

3 Preparing Clinical Narratives

Several steps are necessary to prepare clinical narratives for frame annotation: clinical notes must be retrieved from the clinical data warehouse, lexical units (see Section 4) and their proper context must be extracted, then private patient information must be de-identified. All of this must be done within a secure HIPAA-compliant environment.

The clinical notes are derived from the UT Physicians clinics, a chain of outpatient clinics in the Houston area. The snapshot available from the data warehouse contains more than 260,000 notes with more than 175 million tokens. While by no means a large corpus by EHR standards, it contains sufficient data for a pilot evaluation. This project was deemed exempt by the Committee for the Protection of Human Subjects, the UTHealth Institutional Review Board, under protocol number HSC-SBMI-13-0549.

For each lexical unit (see Section 4), every sentence containing the lexical unit is extracted from the note corpus. Sentence segmentation is by no means a simple task in clinical data (Miller et al., 2015; Zweigenbaum et al., 2016). Stanford CoreNLP (Manning et al., 2014) was used to identify initial sentences, but due to the lack of punctuation in clinical notes, these often constituted multiple (sometimes dozens) of sentences (newlines are often used as end-of-sentence markers, but frequently newlines do not end sentences). As a result, several high-precision rules were used to prune down sentence length. Ultimately, a human annotator was required to remove words not part of the lexical

unit's proper sentence boundary. This was done in conjunction with the de-identification stage below.

Clinical notes contain significant amounts of copy-pasting and templated sentences, so a random sample of sentences might contain substantial numbers of duplicates and near-duplicates. To maximize the diversity of the annotations, the sentences were sorted by TF-IDF cosine distance.

The final preparation step is to de-identify the notes, removing any protected health information (PHI) and replacing it with a placeholder. In the United States, the Health Insurance Portability and Accountability Act (HIPAA) mandates the de-identification of 18 categories of information:

(1) names
(2) geographic localities smaller than a state
(3) dates and ages over 89
(4) telephone numbers
(5) fax numbers
(6) e-mail addresses
(7) Social Security numbers
(8) medical record numbers
(9) health plan numbers
(10) account numbers
(11) certificate/license numbers
(12) vehicle & license numbers
(13) device identifiers
(14) web URLs
(15) IP addresses
(16) biometric identifiers
(17) full face photographic images
(18) any other identifying number/characteristic/code

We expanded on this to cover all ages and geographic localities, as well as other types of identifying information as described in Stubbs and Uzuner (2015). Furthermore, one of our lexical units is frequently a surname in the data: these sentences are completely discarded. Both human annotation and an in-house automatic system (Lee et al., 2017) were used. To reduce bias, the human de-identification occurred first, then the automatic system provided additional suggestions that the human may have missed. The automatic de-identifications were all manually verified to reduce the proliferation of false positives common with de-identification systems. Finally, at future stages of the annotation, as detailed below, annotators always have the option of identifying further PHI missed by this process.

4 Initial Frame Schemas

Based on the existing literature, three common phenotyping tasks were selected: (1) identification of patients with a particular cancer diagnosis, (2) identification of patients with a particular cancer treatment, and (3) identification of patients with particular tumor characteristics. To reduce the complexity of the second task, and to focus on data more likely to be found in outpatient notes, treatments are limited to surgical procedures (i.e., excluding medications, chemotherapy, etc.). These three tasks yield three frames (i) CANCER_DIAGNOSIS, (ii) CANCER_THERAPEUTIC_PROCEDURE, and (iii) TUMOR_DESCRIPTION. We also define an abstract root frame, CANCER_MASTER_FRAME, from which all three inherit elements. This enables frame elements (attributes) that are universal, such as negation and certainty. For each of these frames, an expert in cancer informatics (EVB) helped devise a list of lexical units:

CANCER_DIAGNOSIS: *adenocarcinoma*, *cancer*, *carcinoma*, *leukemia*, *lymphoma*, *malignancy*, *malignant*, *melanoma*, *myeloma*, *sarcoma*

CANCER_THERAPEUTIC_PROCEDURE: *colectomy*, *hysterectomy*, *lymphadenectomy*, *mastectomy*, *palliative*, *pancreatectomy*, *prostatectomy*, *radiation*, *whipple*

TUMOR_DESCRIPTION: *lesion*, *mass*, *tumor*

For each of these lexical units, the process described in Section 3 was followed until up to 500 de-identified sentences were available for each lexical unit. Five of the lexical units (*lymphadenectomy*, *myeloma*, *pancreatectomy*, *sarcoma*, and *whipple*) had fewer than 500 sentences in the corpus.

The elements (attributes) of each frame were determined by an iterative process. First, an initial set of elements was proposed by the cancer expert. During the course of the annotation, new elements were frequently proposed by the annotators. Elements with sufficient frequency and importance–as determined by the cancer expert–were added to the frame schema. For example, FAMILY-HISTORY (for the CANCER_DIAGNOSIS frame), EXTENT (for CANCER_THERAPEUTIC_PROCEDURE), and RECURRENCE (for TUMOR_DESCRIPTION) were added after the start of annotation. It is expected that further annotation will yield additional changes to the schema. The set of frame elements, including brief definitions, is shown in Table 1. Note that some elements (e.g., STATUS, PATIENT) are part of all three frames, but not the CANCER_MASTER_FRAME as these are not expected to necessarily apply to future frames.

5 Annotation

The annotation process largely followed standard linguistic annotation practices (Pustejovsky and Stubbs, 2013). Notably, the sentences containing candidate lexical units were double-annotated then reconciled with the help of a third individual. All frame annotation was performed in Brat (Stenetorp et al., 2012). See Figure 1 for examples.

Two special annotations, whose functionality was briefly mentioned earlier, deserve more attention here. First, as shown in Table 1, there is a special "???" element that annotators can use to indicate potentially useful information that may later result in the creation of a new element. (see Figure 1 for an example). As the disease is so complex, there is simply too much information associated with cancer to include elements for all possible types of information. So the ??? element allows for the prioritization of information based on the actual frequency in the clinical notes. Second, the annotation denoted as ERROR is used by

Frame Element	Description
CANCER_MASTER_FRAME	
CERTAINTY	Certainty/hedging of frame (e.g., *possible*, *likely*)
DATETIME	Temporal information for the frame (often reference to PHI element)
POLARITY	Existence/negation of frame (e.g., *no*, *positive*)
???	Used for other phrases in the text the annotator feels is important, but do not have a corresponding frame element
CANCER_DIAGNOSIS	
DESCRIPTION	Other frame with further information (e.g., TUMOR_DESCRIPTION)
FAMILYHISTORY	Specifies a family member with the diagnosis (as opposed to the PATIENT)
HISTOLOGY	Histological description (e.g., *carcinoma*), can be lexical unit
LOCATION	Part of body associated with the cancer
PATIENT	Reference to the patient (e.g., *patient*, *female*)
QUANTITY	Some quantitative measure of the cancer
STATUS	Diagnostic status (e.g., *history*, *ongoing*)
CANCER_THERAPEUTIC_PROCEDURE	
AGENT	Agent performing the procedure (e.g., *surgeon*)
COMPLICATION	Unexpected, undesirable outcome of procedure (e.g., *nausea*)
EXTENT	Extent of the procedure, often how much of the mass is removed (e.g., *complete*)
LOCATION	Part of body procedure targets
PATIENT	Reference to the patient (e.g., *patient*, *female*)
RESULT	Result of the procedure (e.g., *successful*, *negative*)
STATUS	Procedure status (e.g., *planned*, *postoperative*)
TUMOR_DESCRIPTION	
LOCATION	Part of body tumor is located in, often ambiguous (e.g., *lymph nodes*)
MALIGNANCY	Whether the tumor is benign or malignant
MARGINSTATUS	Description of tumor margin (e.g., *superficial edge*)
METASTATIS	Whether the tumor has metastasized
PATIENT	Reference to the patient (e.g., *patient*, *female*)
QUANTITY	Some quantitative measure of the tumor
RECURRENCE	Whether the tumor has recurred
RESECTABILITY	Indicator of whether tumor is resectable
MORPHOLOGY	Morphology of tumor
SIZE	Diameter/volume of tumor, including unit (e.g., *3-4 mm*)
SIZETREND	Trend in tumor size over time (e.g., *increasing*, *decreasing*, *stable*)
STAGE	Stage number (e.g., *stage IV*)
STATUS	Tumor status (e.g., *present*, *active*)
SUBTUMOR	Link to another TUMOR_DESCRIPTION that further describes this tumor, especially if this is describing a group of tumors

Table 1: Cancer FrameNet pilot frames and their elements.

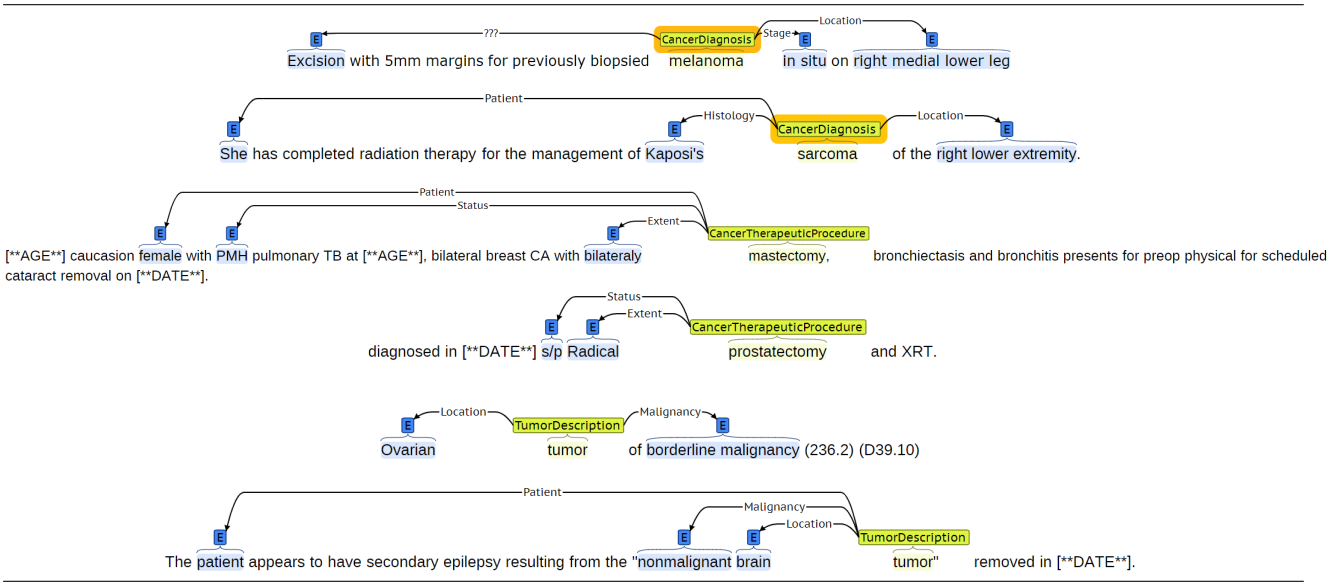


Figure 1: Example annotations.

Information Type	Frequency
Sentences	7,961
Frame Instances	7,163
Average Sentence Length	18
CANCER_DIAGNOSIS	3577
adenocarcinoma	474
cancer	419
carcinoma	495
leukemia	364
lymphoma	473
malignancy	487
melanoma	450
myeloma	191
sarcoma	200
CANCER_THERAPEUTIC_PROCEDURE	2204
colectomy	281
hysterectomy	428
lymphadenectomy	68
mastectomy	459
palliative	133
pancreatectomy	34
prostatectomy	276
radiation	470
whipple	55
TUMOR_DESCRIPTION	1382
lesion	524
mass	352
tumor	506

Table 2: Frequencies of frame instances in the corpus.

annotators to mark both PHI that was missed (thankfully a rare occurrence) and sentence boundaries that should have been removed by the process described in Section 3. Thus ERRORs indicate "sentences" that must be altered before the annotations can be considered final.

Annotation Statistics The annotation was completely performed in 90 hours, taking approximately one minute per sentence, and additional time for reconciliation. Descriptive statistics of the annotated corpus are provided in Table 2. A total of 7,961 sentences are annotated with 7,163 frame-evoking lexical units (out of a total of 8,206 candidate lexical units). Specifically, there are 3,577 CANCER_DIAGNOSIS frames, 2,204 CANCER_THERAPEUTIC_PROCEDURE frames, and 1,382 TUMOR_DESCRIPTION frames. In terms of frame elements (e.g., LOCATION, PATIENT, HISTOLOGY), CANCER_DIAGNOSIS had an average of 3.2 elements per frame instance, CANCER_THERAPEUTIC_PROCEDURE had an average of 1.7, and TUMOR_DESCRIPTION had an average of 1.1 elements. The most common elements which are shared across frames are PATIENT (2,749 instances), STATUS (2,214), LOCATION (2,190), CERTAINTY (999), and POLARITY (638). See Table 3 for more frame element details.

Annotation Agreement Inter-annotator agreement results are shown in Table 4. While observed agreement for frames–shown in Table 4(a)–is high (around 90%) for all three frames, the fairly high levels of expected agreement (73-77%) result in at best moderate κ agreement for CANCER_THERAPEUTIC_PROCEDURE and TUMOR_DESCRIPTION (0.46 and 0.58, respectively). However, κ agreement for CANCER_DIAGNOSIS is excellent (0.84). The reason for the high expected agreements is the lack of ambiguity in many of the lexical units (e.g., *adenocarcinoma* is almost always used to indicate a diagnosis), but each of the frames have at least one lexical unit that has high levels of ambiguity (e.g., *cancer*, *radiation*, and *mass*). Table 4(b) shows the F_1 agreement for frame elements. It is unsurprising to see a wide range of agreements, from the very high (e.g., RESECTABILITY, EXTENT, LOCATION) to the quite low (e.g., RESULT, COMPLICATION, AGENT). Notably, the elements with lowest agreement tend to be relatively rare in the corpus (e.g., 6 AGENTs and 71 COMPLICATIONs compared to 553 EXTENTs and 2,190 LOCATIONs), so low agreement was likely due to a lack of data for calibration.

(a) Frame Agreement

Frame	Observed Agreement	Expected Agreement	κ
CANCER_DIAGNOSIS	0.96	0.73	0.84
CANCER_THERAPEUTIC_PROCEDURE	0.88	0.77	0.46
TUMOR_DESCRIPTION	0.89	0.74	0.58

(b) Frame Element Agreement

Frame Element	Overall F_1
AGENT	0.30
CERTAINTY	0.69
COMPLICATION	0.33
DATETIME	0.69
EXTENT	0.87
FAMILYHISTORY	0.84
HISTOLOGY	0.66
LOCATION	0.82
MALIGNANCY	0.87
MARGINSTATUS	0.61
METASTASIS	0.57
MORPHOLOGY	0.56
PATIENT	0.83
POLARITY	0.77
QUANTITY	0.41
RECURRENCE	0.70
RESECTABILITY	0.88
RESULT	0.34
SIZE	0.83
SIZETREND	0.60
STAGE	0.80
STATUS	0.75

Table 4: Annotator agreement.

6 Discussion & Conclusion

In this paper, we presented Cancer FrameNet, a pilot project focuses on building a cancer-related clinical narrative resource for developing NLP systems. We introduced an annotation schema consisting of three frames (CANCER_DIAGNOSIS, CANCER_THERAPEUTIC_PROCEDURE, and TUMOR_DESCRIPTION), as well as a corpus annotated according to the schema which consists of almost eight thousand sentences. Our primary goal is to inform the developmental process for an extended Cancer FrameNet resource, with the secondary goal of informing the develop-

(a) CANCER_DIAGNOSIS elements

Lexical Unit	CERTAINTY	DATETIME	POLARITY	FAMILY HISTORY	HISTOLOGY	LOCATION	PATIENT	STAGE	QUANTITY	STATUS
adenocarcinoma	112	10	16	46	91	376	119	134	1	84
cancer	61	8	30	85	12	335	197	25	0	101
carcinoma	98	0	24	15	49	419	176	63	5	92
leukemia	37	7	15	167	6	4	96	1	0	91
lymphoma	87	16	28	56	41	77	202	22	0	142
malignancy	192	5	209	9	2	122	195	6	1	46
malignant	6	1	3	0	0	5	3	0	0	1
melanoma	55	17	43	96	19	197	181	49	9	156
myeloma	38	4	18	46	0	10	73	2	0	44
sarcoma	20	3	5	34	2	101	69	3	0	63

(b) CANCER_THERAPEUTIC_PROCEDURE

Lexical Unit	CERTAINTY	DATETIME	POLARITY	AGENT	COMPLICATION	EXTENT	LOCATION	PATIENT	RESULT	STATUS
colectomy	7	7	4	0	4	197	56	77	1	194
hysterectomy	21	14	13	1	9	53	40	183	1	237
lymphadenectomy	3	7	2	0	0	3	48	20	0	38
mastectomy	15	15	5	2	7	191	188	223	16	306
palliative	18	6	0	1	0	0	3	72	6	64
pancreatectomy	1	2	0	0	3	15	25	8	2	24
prostatectomy	9	12	5	0	34	89	65	109	6	208
radiation	25	11	90	2	10	5	117	253	3	149
whipple	3	5	1	0	4	0	2	27	2	41

(c) TUMOR_DESCRIPTION elements

Lexical Unit	CERTAINTY	DATE TIME	POLARITY	MALIG-NANCY	MARGIN STATUS	METAS-TASIS	MORPH-OLOGY	PATIENT	QUANTITY	RECUR-RENCE	RESECT-ABILITY	SIZE	SIZE TREND	STAGE	STATUS
lesion	64	5	42	20	10	9	60	145	14	6	26	51	36	9	44
mass	50	7	48	10	3	10	17	115	0	2	25	54	12	2	16
tumor	77	6	37	42	5	18	24	206	2	11	96	34	27	15	73

Table 3: Frequencies of frame elements in the corpus.

ment of further such corpora in other clinical domains. Additionally, we plan to utilize the corpus as training data for a future NLP system. However, due to the large variety of information types in cancer, and the restriction to a single institution, this corpus is of limited utility in developing complete and robust cancer information extraction methods. For example, important textual information such as post-treatment status, medication, and genomic & molecular testing results are critically important in the "precision medicine" era of cancer treatment.

Apart from the limitations related to the corpus and missing elements in the schema, another complex issue is the granularity of the frames. For example, a reasonable phenotyping task might be to find cases of cancer of any original that have metastasized to the lymph nodes. However, a common phrase used to describe biopsy results is "*with lymph node involvement*", which does not necessarily specify directly whether it is a lymph node cancer (e.g., non-Hodgkin lymphoma) or a metastasis, even if it is likely the latter. Further, encoding hypothetical phrases such as "*if tumor untreated*" presents schematic difficulties, though these could likely be overcome given sufficient data for initial exploration.

On the whole, however, frame semantics provides a flexible means of encoding important cancer information without getting bogged down in the minute details of standardizing clinical representations (a major barrier to interoperability in structured clinical data). Rules can be developed on top of the extracted frames (the "reasoning" step we describe in the Introduction) that make the necessary assumptions to utilize imperfect data. For instance, in the lymph node example, the phenotyping algorithm could exclude patients who only have a diagnosis related to the lymph node, thus likely excluding the non-metastasis cases. It is also important to recognize that, given the presence of structured data in the EHR, NLP is often seen as an (imperfect) means of supplementing (also imperfect) structured data. In this context, given that the vast majority of sentences were represented quite well using the proposed frames, this approach appears quite promising.

Another potential pitfall of using frame semantics for cancer involves the ability of organizing information into compact frames. Put another way, if almost all cancer frames shared the exact same frame elements, then a frame-based schema would be a poor fit. Instead, we found frames to work quite well in this regard. First, three frame elements were used in the CANCER_MASTER_FRAME (CERTAINTY, POLARITY, and DATETIME), but these actually generalize to just about all frames, well beyond cancer. Second, two frame elements were used in all three frames, PATIENT and STATUS. The former might ultimately be a better fit for CANCER_MASTER_FRAME, but the STATUS element has different semantics for each frame (e.g., the status of someone's cancer versus the status of the surgery to remove a tumor). Third, two elements, LOCATION and STAGE, were in two frames (CANCER_DIAGNOSIS and TUMOR_DESCRIPTION), but given the inter-relatedness of having a tumor and being diagnosed with cancer, this is not particularly surprising. Finally, beyond these cases, the remaining 16 elements were unique to a single frame, suggesting that frame-based representations are a good fit for cancer-related EHR text.

A common problem with clinical text is the mixture of structured data and true natural language prose. Commonly, structured descriptions are automatically or semi-automatically integrated with the clinical narrative. While frame semantics can handle such cases (often trivially), they are of limited value given the likely duplication of that same information in the structured part of the EHR data.

However, in this pilot project we limited ourselves to just the sentence-level scope for frame annotation, so determining better ways to handle this kind of data in future projects will be important.

On the other hand, our limitation to sentences provides a useful starting point for cross-institutional data sharing. The privacy concerns surrounding sharing of complete records largely goes away when individual sentences are manually stripped of their PHI and any potential linking information back to the original record. The ability to gather multiple institutions' data, all organized according to the same frame semantic schema, into a single FrameNet would be highly valuable to the clinical NLP community. These frames could then be mapped to existing structured clinical data standards, such as FHIR and OHDSI.

The one notable exception to sentence-based annotation, and something we would strongly consider changing in a future such project, is the use of sentences at the frame identification stage. While individual frame elements can easily be annotated at the sentence level, the word sense disambiguation task of determining whether a phrase invokes a particular frame is sometimes difficult. This is usually limited to cases where the sentence is in fact a fragment (e.g., a single item in a bulleted list), but this is a common phenomenon in EHR text and therefore worth taking careful consideration in how to handle. It may be sufficient to provide a human annotator with the extra-sentential context, but limit the machine to simply the sentence itself for classification. Alternatively, we could relax the notion of sentences in the case of fragments to provide more context. These compromise strategies could overcome one of the primary issues the annotators struggled with, while likely not having that significant an impact on the resource for training NLP systems.

7 Acknowledgements

This work was supported by the U.S. National Library of Medicine, National Institutes of Health (NIH) (awards R00 LM012104 and R01 LM01068); the National Center for Advancing Translational Sciences (NIH awards UL1 TR000371, TL1 TR000369, KL1 TR000370, UL1 TR001105); UTHealth DSRIP; the Cancer Prevention Research Institute of Texas (CPRIT) Data Science and Informatics Core for Cancer Research (RP170668) and Precision Oncology Decision Support Core (RP150535); and the Patient-Centered Outcomes Research Institute (PCORI) grants "Privacy Preserving Interactive Record Linkage (PPIRL) via Information Suppression" and CDRN-1306-04608.

8 Bibliographical References

Ashish, N., Dahm, L., and Boicey, C. (2014). University of California, Irvine-Pathology Extraction Pipeline: the pathology extraction pipeline for information extraction from pathology reports. *Health Informatics Journal*, 20(4):288–305.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*.

Baker, C. F., Schneider, N., Petruck, M. R. L., and Ellsworth, M. (2015). Getting the Roles Right: Using FrameNet in NLP. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorial Abstracts*.

Cheng, L., Zheng, J., Savova, G., and Erickson, B. (2010). Discerning tumor status from unstructured MRI reports: completeness of information in existing reports and utility of automated natural language processing. *Journal of Digital Imaging*, 23(2):119–132.

Coden, A., Savova, G., Sominsky, I., Tanenblatt, M., Masanz, J., Schuler, K., Cooper, J., Guan, W., and de Groen, P. (2009). Automatically extracting cancer disease characteristics from pathology reports into a disease knowledge representation model. *Journal of Biomedical Informatics*, 42(5):937–949.

D'Avolio, L., Litwin, M., Rogers, S., and Bui, A. (2008). Facilitating Clinical Outcomes Assessment through the Automated Identification of Quality Measures for Prostate Cancer Surgery. *Journal of the American Medical Informatics Association*, 15(3):341–348.

Denny, J., Peterson, J., Choma, N., Xu, H., Miller, R., Bastarache, L., and Peterson, N. (2010). Extracting timing and status descriptors for colonoscopy testing from electronic medical records. *Journal of the American Medical Informatics Association*, 17(4):383–388.

Dolbey, A., Ellsworth, M., and Scheffzyk, J. (2006). BioFrameNet: A Domain-specific FrameNet Extension with Links to Biomedical Ontologies. In *Proceedings of the Biomedical Ontology in Action Workshop at KR-MED*, pages 86–94.

Dolbey, A. (2009). *BioFrameNet: A FrameNet Extension to the Domain of Molecular Biology*. Ph.D. thesis, UC Berkeley.

Harkema, H., Chapman, W., Saul, M., Dellon, E., Schoen, R., and Mehrotra, A. (2010). Developing a natural language processing application for measuring the quality of colonoscopy procedures. *Journal of the American Medical Informatics Association*, 18(Suppl 1):i150–i156.

Heintzelman, N., Taylor, R., Simonsen, L., Lustig, R., Anderko, D., Haythornthwaite, J., Childs, L., and Bova, G. (2013). Longitudinal analysis of pain in patients with metastatic prostate cancer using natural language processing of medical record text. *Journal of the American Medical Informatics Association*, 20(5):898–905.

Heppin, K. F. and Gronostaj, M. T. (2014). Exploiting FrameNet for Swedish: Mismatch? *Construction and Frames*, 6(1):52–72.

Imler, T., Morea, J., Kahi, C., and Imperiale, T. (2013). Natural language processing accurately categorizes findings from colonoscopy and pathology reports. *Clinical Gastroenterology and Hepatology*, 11(6):689–694.

Lee, H.-J., Wu, Y., Zhang, Y., Xu, J., Xu, H., and Roberts, K. (2017). A hybrid approach to automatic deidentification of psychiatric notes. *Journal of Biomedi-*

cal Informatics.

Lin, L., Chen, H., and Bi, Y. (2015). The Designing and Construction of Domain-oriented Vietnamese-English-Chinese FrameNet. In *Proceedings of Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 53–65.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Martinez, D. and Li, Y. (2011). Information extraction from pathology reports in a hospital setting. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1877–1882.

Martinez, D., Cavedon, L., and Pitson, G. (2013). Stability of Text Mining Techniques for Identifying Cancer Staging. In *Proceedings of the 4th Workshop on Health Document Text Mining and Information Analysis*.

McCowan, I., Moore, D., Nguyen, A., Bowman, R., Clarke, B., Duhig, E., and Fry, M. (2007). Collection of cancer stage data by classifying free-text medical reports. *Journal of the American Medical Informatics Association*, 14(6):736–745.

Miller, T. A., Finan, S., Dligach, D., and Savova, G. (2015). Robust Sentence Segmentation for Clinical Text. In *Proceedings of the AMIA Annual Symposium*, pages 112–113.

Napolitano, G., Fox, C., Middleton, R., and Connolly, D. (2010). Pattern-based information extraction from pathology reports for cancer registration. *Cancer Causes & Control*, 21(11):1887–1894.

Napolitano, G., Marshall, A., Hamilton, P., and Gavin, A. (2016). Machine learning classification of surgical pathology reports and chunk recognition for information extraction noise reduction. *Artificial Intelligence in Medicine*, 70:77–83.

Ohara, K. H. (2016). Universality of Frames: A View from Japanese FrameNet. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (Tutorial)*.

Ping, X.-O., Tseng, Y.-J., Chung, Y., Wu, Y., Hsu, C., Yang, R., Huang, G., Lai, F., and Liang, J. (2013). Information extraction for tracking liver cancer patients' statuses: from mixture of clinical narrative report types. *Telemedicine Journal and e-Health*, 19(9):704–710.

Pustejovsky, J. and Stubbs, A. (2013). *Natural Language Annotation for Machine Learning*. O'Reilly Media.

Rezhake, M. and Kuerban, A. (2015). Design of the Uyghur FrameNet Desktop. *Lecture Notes on Software Engineering*, 3(1):53–56.

Ruppenhofer, J. (2013). Anchoring Sentiment Analysis in Frame Semantics. In *Veredas*, pages 66–81.

Savova, G., Harris, M., Pakhomov, S., and Chute, C. (2005). Frame Semantics and the Domain of Functioning, Disability and Health. In *Proceedings of the AMIA Annual Symposium*, page 1006.

Savova, G. K., Tseytlin, E., Finan, S., Castine, M., Miller, T., Medvedeva, O., Harris, D., Hochheiser, H., Lin, C., Chavan, G., and Jacobson, R. S. (2017). DeepPhe: A Natural Language Processing System for Extracting Cancer Phenotypes from Clinical Records. *Cancer Research*, 77(21).

Schmidt, T. (2006). Interfacing Lexical and Ontological Information in a Multilingual Soccer FrameNet. In *Proceedings on OntoLex*, pages 75–81.

Segagni, D., Tibollo, V., Dagliati, A., Zambelli, A., Priori, S., and Bellazzi, R. (2012). An ICT infrastructure to integrate clinical and molecular data in oncology research. *BMC Bioinformatics*, 13(Suppl 4):S5.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstration Session at EACL 2012*, pages 102–107.

Stubbs, A. and Uzuner, Ö. (2015). Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/UTHealth corpus. *Journal of Biomedical Informatics*, 58:S20–S29.

Taira, R. K., Soderland, S. G., and Jakobovits, R. M. (2001). Automatic Structuring of Radiology Free-Text Reports. *Radiographics*, 21:237–245.

Torrent, T., Salomão, M. M., Campos, F., Braga, R., Matos, E., Gamonal, M., Gonçalves, J., Souza, B., Gomes, D., and Peron, S. (2014). Copa 2014 Brazil: a frame-based trilingual electronic dictionary for the Football World Cup. In *Proceedings of the International Conference on Computational Linguistics (COLING): System Demonstrations*, pages 10–14.

Vanderwende, L., Xia, F., and Yetisgen-Yildiz, M. (2013). Annotating Change of State for Clinical Events. In *Proceedings of the 1st Workshop on EVENTS: Definition, Detection, Coreference, and Representation*.

Wang, H., Zhang, W., Zeng, Q., Li, Z., Feng, K., and Liu, L. (2014). Extracting important information from Chinese Operation Notes with natural language processing methods. *Journal of Biomedical Informatics*, 48:130–136.

Wilson, R., Chapman, W., Defries, S., Becich, M., and Chapman, B. (2010). Automated ancillary cancer history classification for mesothelioma patients from free-text clinical reports. *Journal of Pathology Informatics*, 1:24.

Xu, H., Anderson, K., Grann, V. R., and Friedman, C. (2004). Facilitating cancer research using natural language processing of pathology reports. In *Studies in Health Technology and Informatics*, volume 107(Pt 1), pages 565–572.

Zhao, D. and Weng, C. (2011). Combining PubMed Knowledge and EHR Data to Develop a Weighted Bayesian Network for Pancreatic Cancer Prediction. *Journal of Biomedical Informatics*, 44(5):859–868.

Zweigenbaum, P., Grouin, C., and Lavergne, T. (2016). Supervised classification of end-of-lines in clinical text with no manual annotation. In *Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM)*, pages 80–88.

A New Corpus to Support Text Mining for the Curation of Metabolites in the ChEBI Database

Matthew Shardlow[1], Nhung Nguyen[1], Gareth Owen[2], Steve Turner[2], Claire O'Donovan[2], Andrew Leach[2], John McNaught[1], Sophia Ananiadou[1]

National Centre for Text Mining, University of Manchester[1], European Bioinformatics Institute[2]

{matthew.shardlow, nhung.nguyen, john.mcnaught, sophia.ananiadou}@manchester.ac.uk,

{gowen, turner, odonovan, arl}@ebi.ac.uk

Abstract

We present a new corpus of 200 abstracts and 100 full text papers which have been annotated with named entities and relations in the biomedical domain as part of the OpenMinTeD project. This corpus facilitates the goal in OpenMinTeD of making text and data mining accessible to the users who need it most. We describe the process we took to annotate the corpus with entities (*Metabolite*, *Chemical*, *Protein*, *Species*, *Biological Activity* and *Spectral Data*) and relations (*Isolated From*, *Associated With*, *Binds With* and *Metabolite Of*). We report inter-annotator agreement (using F-score) for entities of between 0.796 and 0.892 using a strict matching protocol and between 0.875 and 0.963 using a relaxed matching protocol. For relations we report inter annotator agreement of between 0.591 and 0.693 using a strict matching protocol and between 0.744 and 0.793 using a relaxed matching protocol. We describe how this corpus can be used within ChEBI to facilitate text and data mining and how the integration of this work with the OpenMinTeD text and data mining platform will aid curation of ChEBI and other biomedical databases.

Keywords: Text Mining, Corpus, Bioinformatics

1. Introduction and Background

The ChEBI (Chemical Entities of Biological Interest) database (Degtyarenko et al., 2008; Hastings et al., 2016) is a freely available, electronic dictionary and ontology of small molecules. ChEBI was created to help researchers in the field of molecular biology who need to know the structure, names, and properties of the small molecules that they encounter in their research. There are a number of freely-available chemical databases. Most of them are created by an automatic 'pipeline' process and contain information on polymers, industrial chemicals, synthetic intermediates, etc. Their sheer size creates problems for users, as any search may result in hundreds or even thousands of answers. For non-expert users it is very difficult to determine which, if any, is the compound they are really looking for. By contrast, the focus of the ChEBI database is on high quality rather than quantity.

ChEBI is manually curated and focuses on the requirements of the molecular biology community. Manual curation assures the high quality of the database, but it also makes it an expensive database to produce, particularly so for our focus of metabolites. Users who would like to add a new metabolite, may only know a research code or a trivial name from which it is not possible to deduce a structure. The curator then has to search through scientific literature to find as much information as possible about the new metabolite. The information is likely to include:

- In which species is it present?
- Does it have any interesting biological properties, applications, etc? (E.g., Biological activity)
- Is there any spectral data available that indicates the structure?
- From which chemical's metabolism does it derive?

All of this information ideally needs to be supported by appropriate citations to publications in the scientific literature. Whilst we cannot automate every step in the curation process, we can use text mining to facilitate the curator throughout the process.

This work is part of the OpenMinTeD project[1] (Oudenhoven and Pontika, 2017), which is developing a text and data mining framework consisting of an integrated registry, metadata schema, text mining workflow service and annotation viewer. The OpenMinTeD project works with data providers to give users access to large databases of open access publications (Knoth and Pontika, 2016). The OpenMinTeD project is also promoting a number of community standards for text and data mining (Przybyła et al., 2016; Ba and Bossy, 2016; Peters, 2016), such as UIMA (Ferrucci and Lally, 2004), maven[2] and Docker (Merkel, 2014), which we have adopted in our work. The OpenMinTeD project has also made extensive efforts to facilitate the legal interoperability of text mining software and content. From which, we have adopted lessons and advice for our work (Margoni and Dore, 2016; Labropoulou et al., 2016)

To facilitate curation in ChEBI through text mining, we will deploy this corpus and any tools derived from it via the OpenMinTeD platform. The platform provides many advantages to both the data provider and consumer, including persistent availability, support and training, sustainability of the platform, reusability of tools and redeployability of tools to new scenarios. In addition, the tools we make available will be usable by audiences other than our intended target (curators of ChEBI), fulfilling the project's goal of promoting open science throughout communities which stand to benefit from text and data mining.

The intended purpose of this corpus is to provide training

[1] https://openminted.eu

[2] https://maven.apache.org/

data for both named entity recognition and relation extraction tools. We briefly describe approaches to each of these areas below.

Named entity recognition (Nadeau and Sekine, 2007) is a common text mining task in which an algorithm is used to identify parts of an unstructured text that can be categorised according to a given schema. Named entity recognition has been applied across diverse corpora including but not limited to Twitter (Derczynski et al., 2015; Baldwin et al., 2015), biomedical papers (Munkhdalai et al., 2015; Wang et al., 2015) and other areas. Approaches for named entity recognition range from using dictionaries (Cohen and Sarawagi, 2004) to using regular expressions and other rules (Kluegl et al., 2016; Chiticariu et al., 2010), using machine learning approaches such as the conditional random field (Lu et al., 2015), or more recently leveraging advances in the field of deep learning (Chiu and Nichols, 2015; Lample et al., 2016).

Relation extraction (Aggarwal and Zhai, 2012) is a task in information extraction that requires an algorithm to link together two named entities according to a given schema which defines the meaning of the link. Relation extraction requires named entity recognition to be performed as a prerequisite. Relation extraction may be performed using supervised (Nguyen and Grishman, 2015; Pons et al., 2015) or unsupervised methods (Quan et al., 2014). Simple heuristics such as two entities in close proximity may serve as a baseline, but are less powerful than measures which take structural and context features of the named entities into account.

2. Corpus Construction

We annotated a set of 200 abstracts and 100 full papers with the entities that were of interest to us, as well as relations between these entities. Although we could have leveraged existing corpora for some of the entity types that we were interested in, we found that there were very few corpora dealing with metabolites in the fine-grained annotations that we were interested in. In addition, whereas corpora may have contained the entity types of interest to us, no suitable corpora existed containing the relation types we wished to annotate.

We first defined the entity types that were of interest to us. These were determined as follows. The definitions of these entities were driven by the curators of ChEBI, who also participated in the annotation process.

Metabolite: A chemical which has been produced by, detected in, or isolated from a living organism, where this is clear from the context of the paper (e.g. Nitrosobenzene, 11-Deoxycorticosterone, sclerotiorin).

Chemical: Any name that is used to define 'small' chemicals (those that are not proteins, nucleic acids, etc.). Includes molecules, salts, class names (e.g. benzoate esters; indole alkaloids; etc.) and groups (parts of molecules) - e.g. methyl group; benzyl substituent; alanine residue…)

Protein: Any protein or large polypeptide (usually one that is too big to be drawn by normal chemical drawing software). All enzymes and receptors are considered to be proteins in our scheme (e.g. 4-Dihydroxyphenylalanine decarboxylase, Dopa decarboxylase).

Species: Any entity referring to a formal name for a living organism or from which the name can be inferred (e.g. 'volunteer', 'patient' implies 'human').

Biological Activity: An effect/consequence that a chemical entity has on a biological system. Examples may include affecting the activity (e.g. by inhibition or activation) of a particular enzyme; growth regulator; antimicrobial agent; apoptosis inducer; anti-inflammatory; flavour enhancer; etc.

Spectral Data: data arising from spectrometry. e.g., 1H-NMR, 13C-NMR, MS, X-ray, IR, etc. Where two or more spectroscopic techniques are present, each should be tagged separately.

We also defined the following types of relation between the entities. The relations help us to answer the questions about a new metabolite that a curator may have, as outlined in Section 1..

Isolated_from(Metabolite, Species): A metabolite was isolated from or detected in a specific species.

Associated_with(Chemical / Metabolite, Biological Activity / Spectral Data): A chemical or metabolite is linked to a particular biological activity or spectral data.

Binds_With(Chemical / Metabolite, Protein): A chemical or metabolite interacts with (e.g.binds) and affects the behaviour of a biological target.

Metabolite of(Metabolite, Chemical): A metabolite is derived from the metabolism of a related compound.

We developed guidelines containing these definitions. The guidelines provided examples of the annotations, as well as specific information for each category that helped the annotators to agree in ambiguous cases. The guidelines were updated after each round of annotation in accordance with feedback from the annotators.

We selected 200 abstracts for annotation from PubMed, according to the criteria that each abstract should contain at least one of the relation types that we were looking for. We performed double annotation for all 200 abstracts to ensure the consistency and validity of our annotations. We engaged two annotators, both of whom were actively involved in the curation of the ChEBI database. We initially annotated 20 abstracts and evaluated inter-annotator agreement. We found some discrepancies between the annotators' interpretations of the categories and so we discussed these with the annotators and updated our definitions and the guidelines accordingly. We proceeded to annotate a further 20 abstracts, after which we obtained a higher agreement on all categories. We performed further resolution between the annotators and updated the guidelines, with further rounds of 60 and then 100 abstracts to bring the total number of doubly annotated abstracts up to 200.

Following on from this large round of double annotation, we proceeded to singly annotate the full text of 100 papers. It has been shown elsewhere in the literature that the annotation of full texts is preferential over the annotation of abstracts (Westergaard et al., 2017). We extracted full texts using the API provided by the Elsevier developer portal.[3] We chose 100 papers for annotation, again according to the criteria that they should contain at least one relation type of interest to us. Both annotators contributed to these annotations, although each full paper was only annotated by one annotator.

Both the corpus and guidelines associated with this paper will be made available via the OpenMinTeD platform, as well as in the supplementary material to this work, upon final submission.

3. Corpus Statistics

We calculated inter annotator agreement using the F-score statistic, as is the norm for text mining tasks. We could not use the more common Kappa statistic, as this takes the true negative rate into account, which is not appropriate for named entity recognition where all tokens or spans that have not been annotated will be considered true negatives. We calculate the F-score using both a strict and relaxed matching protocol as described below. The matching protocols are further explained in Table 1.

In the strict matching protocol, the annotators are considered to agree on a named entity only if they have annotated the exact same span and assigned the same entity type. In this case, the annotation will be considered a true positive. To obtain the false positive and false negative rate, we consider one annotator to be the 'gold standard'. A false negative is assigned if the gold annotator has created an annotation that is not present in the other annotations. A false positive is assigned if the gold annotator has not created an annotation that is present in the other annotations. We follow the same protocol for relations, where a true positive is assigned if both terms match exactly, as well as the type of the relation.

In the relaxed matching protocol, the annotators are considered to agree on a named entity only if the spans of two entities overlap by at least one character and the category is the same. False positives and false negatives are assigned as above when a matching term cannot be found for either the gold standard or the other annotator. Relations are assigned as a true positive if the terms match using the relaxed matching criteria and the category is the same.

The strict matching protocol may be overly punitive in cases where the annotators clearly agree on an annotation, but disagree about exactly which part of a term should be covered by the annotation. For example, consider the annotation *'poly aromatic hydrocarbons'*. One annotator may annotate the whole span as a chemical, whereas the other annotates the span *'aromatic hydrocarbons'*. In this case, the annotators both agree that there is a chemical at this point, the disagreement is around whether to include the term 'poly' in the annotation or not. According to the strict matching protocol this would be regarded as a false negative for the first annotator and a false positive for the second. According to the relaxed matching criteria, this would be considered a true positive. The relaxed matching criteria may be overly lenient, as even one character in common could signify a true positive. However, we hope that by providing both measures the reader will have the best tools to interpret our results. We have provided our results for inter annotator agreement in Table 2.

We can see from the results in Table 2 that we were able to attain a high level of agreement for entities between the annotators on the set of 200 abstracts in our corpus. It is clear to see that using the relaxed matching protocol yields an increase in agreement over the strict matching protocol. For the entities, the largest increase is for *Spectral Data*, where using relaxed matching gives an increase of 0.122, indicating that the annotators often disagree about the exact boundaries of *Spectral Data*. The *Species* category was the named entity with the smallest increase when using relaxed matching of only 0.05, indicating that the annotators rarely disagreed on the exact boundaries of each *Species* annotation. Overall, the highest agreement was attained for the *Species* annotation when using the strict matching protocol and for *Spectral Data* when using the relaxed matching protocol.

The agreement for relations is lower than for entities. Each relation covers two entities, so there is greater scope for the annotators to disagree on how these entities should be related. Furthermore, if the annotators have not agreed on the scope of one or both entities in the relation then it will not be considered a match according to the strict protocol. Accordingly, we can see that using the relaxed protocol for relations generally gives a larger performance boost than for entities alone. The largest increase is for the *Associated With* category, where relaxed matching yields an increase of 0.197. The *Associated With* category covers the *Spectral Data* entity which had the largest increase of all entities when using relaxed matching vs. strict, which may explain the increase for *Associated With*. The smallest increase within the relations is for *Binds With* where an increase of only 0.051 is recorded. This covers the *Metabolite*, *Chemical* and *Protein* entities, all of which had similarly small increases. Overall, the agreement for relations under relaxed matching gives scores between 0.744 for *Binds With* and 0.793 for *Associated With*. These figures indicate that agreement is high, if not perfect, and that annotators may not always agree on the exact boundaries of each annotation.

The final corpus contains 200 doubly annotated abstracts and 100 singly annotated full papers. The full papers portion of the corpus is much longer than that of the abstracts portion as each full paper is much longer than a single abstract. We present statistics on the full corpus in Table 3.

We can see from Table 3 that the full papers are indeed much richer than the abstracts. For entities, the average full paper contain between 12.08 (*Species*) and 28.57 (*Spectral Data*) times more entities than the average abstract. The increase in data is lower for relations, where the average full paper contains between 2.32 (*Metabolite Of*) and 8.09 (*Associated With*) times more relations than the average ab-

[3] https://dev.elsevier.com/

	annotator 1 benzoate esters annotator 2 are	annotator 1 benzoate esters annotator 2 are	annotator 1 benzoate esters are
Strict	Match	No Match	No Match
Relaxed	Match	Match	No Match

Table 1: Three possible annotation scenarios and the results of the strict and relaxed matching protocol. The first column shows an 'exact match' where both annotators have highlighted the same term. This is considered a match by both protocols. The second column shows a partial match, where the annotators have agreed on the annotation, but disagreed on the scope. The strict matching protocol does not consider this a match, whereas the relaxed matching protocol does consider this a match. The final column shows that annotator 1 made an annotation, where annotator 2 did not. Neither protocol would consider this case to be a match.

Category	**Strict**	**Relaxed**
Metabolite	0.821	0.875
Chemical	0.865	0.950
Protein	0.866	0.944
Species	0.892	0.942
Biological Activity	0.796	0.904
Spectral Data	0.841	0.963
Isolated From	0.591	0.766
Associated with	0.596	0.793
Binds With	0.693	0.744
Metabolite Of	0.623	0.789

Table 2: The agreement between annotators on entities (top) and relations (bottom) using strict and relaxed matching protocols. All reported values are F-score.

Category	**Abstracts**	**Full Papers**	**Scale**
Documents	200	100	—
Words	160.34	3722.80	23.22
Sentences	10.37	112.15	10.81
Metabolite	3.30	48.45	14.68
Chemical	14.49	213.29	14.72
Protein	4.14	63.01	15.22
Species	2.72	32.87	12.08
Biological Activity	2.98	47.51	15.94
Spectral Data	0.28	8.00	28.57
Isolated From	1.29	8.00	6.20
Associated With	4.45	36.01	8.09
Binds With	1.32	6.18	4.68
Metabolite Of	0.75	1.74	2.32

Table 3: The averaged statistics per document for both abstracts and full papers in our corpus. The first line of data shows the total number of each document type. It is clear that the full papers contain much more data than the abstracts alone. We present general stats (top), entities (middle) and relations (bottom). The third column shows the magnitude of the increase in number of available entities when using full texts as opposed to abstracts.

stract. The lower increase in relations, may be because they are more difficult for the annotator to spot in full papers, where a relation may span several paragraphs.

It is interesting to note that whilst the number of words in a full paper is on average 23.22 times greater than in abstracts, most of the increases for entities and relations are below this number (all except for *Spectral Data*). This demonstrates that whilst full papers are richer in availability of entities and relations, they are less densely packed with entities and relations than abstracts. This implies that a larger volume of entities and relations could be found by processing the same number of words from abstracts than from full papers. However, full papers (where available) provide much more information about the entities they contain and so are important for information extraction tasks where important information may not be reported in the abstract.

The *Chemical* entity is the most frequently reported entity in both the abstracts and full papers in our corpus, whereas *Spectral Data* is the the least frequent. The documents we have chosen are from biomedical journals and so it is unsurprising that they mention chemicals with such high frequency. *Spectral Data* is surprisingly low in abstracts, indicating that authors do not commonly report this entity in the abstract, but instead report it in the full text of the paper.

The *Associated With* relation is the most frequent relation in both the abstracts and full papers, whereas the *Metabolite Of* relation is the least frequent. *Associated With* is a broad category that covers several entity types, which may explain why it is more frequent than the other more narrowly scoped relations.

4. Conclusion

We have described our new corpus containing 200 abstracts and 100 full papers annotated with entities and relations that are useful for automating the curation process of the ChEBI database. This corpus will be made available as part of the OpenMinTeD project for use in text mining applications. We will use the corpus to train text mining tools capable of detecting the entities and relations contained within the corpus. These text mining tools will be made available via the OpenMinTeD platform for use in the curation of ChEBI, as well as for use by other teams of curators who face similar problems in automating their curation workflows.

5. Bibliographical References

Aggarwal, C. C. and Zhai, C. (2012). *Mining text data*. Springer Science & Business Media. Section 2.3.

Ba, M. and Bossy, R. (2016). Interoperability of corpus processing work-flow engines: the case of alvisnlp/ml in openminted. In Richard Eckart de Castilho, et al., editors, *Proceedings of the Workshop on Cross-Platform Text Mining and Natural Language Processing Interoperability (INTEROP 2016) at LREC 2016*, pages 15–18, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Baldwin, T., de Marneffe, M.-C., Han, B., Kim, Y.-B., Ritter, A., and Xu, W. (2015). Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135.

Chiticariu, L., Krishnamurthy, R., Li, Y., Reiss, F., and Vaithyanathan, S. (2010). Domain adaptation of rule-based annotators for named-entity recognition tasks. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 1002–1012, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chiu, J. P. C. and Nichols, E. (2015). Named entity recognition with bidirectional LSTM-CNNs. *CoRR*, abs/1511.08308.

Cohen, W. W. and Sarawagi, S. (2004). Exploiting dictionaries in named entity extraction: Combining semi-markov extraction processes and data integration methods. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 89–98, New York, NY, USA. ACM.

Degtyarenko, K., de Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., AlcÃ¡ntara, R., Darsow, M., Guedj, M., and Ashburner, M. (2008). ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Research*, 36(suppl1):D344–D350.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., and Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49.

Ferrucci, D. and Lally, A. (2004). Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Hastings, J., Owen, G., Dekker, A., Ennis, M., Kale, N., Muthukrishnan, V., Turner, S., Swainston, N., Mendes, P., and Steinbeck, C. (2016). Chebi in 2016: Improved services and an expanding collection of metabolites. *Nucleic acids research*, 44(D1):D1214–D1219.

Kluegl, P., Toepfer, M., Beck, P.-D., Fette, G., and Puppe, F. (2016). Uima Ruta: Rapid development of rule-based information extraction applications. *Natural Language Engineering*, 22(1):1–40.

Knoth, P. and Pontika, N. (2016). Aggregating research papers from publishers' systems to support text and data mining: Deliberate lack of interoperability or not? In Richard Eckart de Castilho, et al., editors, *Proceedings of the Workshop on Cross-Platform Text Mining and Natural Language Processing Interoperability (INTEROP 2016) at LREC 2016*, pages 1–4, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Labropoulou, P., Piperidis, S., and Margoni, T. (2016). Legal interoperability is-sues in the framework of the openminted project: a methodological overview. In Richard Eckart de Castilho, et al., editors, *Proceedings of the Workshop on Cross-Platform Text Mining and Natural Language Processing Interoperability (INTEROP 2016) at LREC 2016*, pages 60–63, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics.

Lu, Y., Ji, D., Yao, X., Wei, X., and Liang, X. (2015). CHEMDNER system with mixed conditional random fields and multi-scale word clustering. *Journal of Cheminformatics*, 7(1):S4, Jan.

Margoni, T. and Dore, G. (2016). Why we need a text and data mining exception (but it is not enough). In Richard Eckart de Castilho, et al., editors, *Proceedings of the Workshop on Cross-Platform Text Mining and Natural Language Processing Interoperability (INTEROP 2016) at LREC 2016*, pages 57–59, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Merkel, D. (2014). Docker: Lightweight linux containers for consistent development and deployment. *Linux J.*, 2014(239), March.

Munkhdalai, T., Li, M., Batsuren, K., Park, H. A., Choi, N. H., and Ryu, K. H. (2015). Incorporating domain knowledge in chemical and biomedical named entity recognition with word representations. *Journal of Cheminformatics*, 7(1):S9.

Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Nguyen, T. H. and Grishman, R. (2015). Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 39–48, Denver, Colorado, June. Association for Computational Linguistics.

Oudenhoven, M. and Pontika, N. (2017). Learning about text and data mining: The future of open science. In *Proccedings of the Open Science Conference*, Berlin, Germany.

Peters, W. (2016). Tackling resource interoperability: Principles, strategies and models. In Richard Eckart de Castilho, et al., editors, *Proceedings of the Workshop on Cross-Platform Text Mining and Natural Language Processing Interoperability (INTEROP 2016) at LREC 2016*, pages 34–37, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Pons, E., Becker, B., Akhondi, S., Afzal, Z., van Mulligen, E., and Kors, J. (2015). Religator: chemical-disease relation extraction using prior knowledge and textual information. In *Proceedings of the Fifth BioCreative Challenge Evaluation Workshop, Sevilla, Spain*, pages 247–253.

Przybyła, P., Shardlow, M., Aubin, S., Bossy, R., Eckart de Castilho, R., Piperidis, S., McNaught, J., and Ananiadou, S. (2016). Text mining resources for the life sciences. *Database*, 2016(0):baw145.

Quan, C., Wang, M., and Ren, F. (2014). An unsupervised text mining method for relation extraction from biomedical literature. *PloS one*, 9(7):e102039.

Wang, X., Yang, C., and Guan, R. (2015). A comparative study for biomedical named entity recognition. *International Journal of Machine Learning and Cybernetics*, Sep.

Westergaard, D., Stærfeldt, H.-H., Tønsberg, C., Jensen, L. J., and Brunak, S. (2017). Text mining of 15 million full-text scientific articles. *bioRxiv*.

Parallel Corpora for the Biomedical Domain

Aurélie Névéol[†], Antonio Jimeno Yepes[*], Mariana Neves[‡], Karin Verspoor[⋆]
[†] LIMSI, CNRS, Université Paris-Saclay, F-91405 Orsay
[*] IBM Research Australia
[‡] German Federal Institute for Risk Assessment (BfR), Germany
[⋆] University of Melbourne, Australia
aurelie.neveol@limsi.fr, antonio.jimeno@au1.ibm.com
mariana.lara-neves@bfr.bund.de, karin.verspoor@unimelb.edu.au

Abstract

A vast amount of biomedical information is available in the form of scientific literature and government-authored patient information documents. While English is the most widely used language in many of these sources, there is a need to provide access to health information in languages other than English. Parallel corpora can be leveraged to implement cross-lingual information retrieval or machine translation tools. Herein, we review the extent of parallel corpus coverage in the biomedical domain. Specifically, we perform a scoping review of existing resources and we describe the recent development of new datasets for scientific literature (the EDP dataset and an extension of the Scielo corpus) and clinical trials (the ReBEC corpus). These corpora are currently being used in the biomedical task in the Conference on Machine Translation (WMT'16 and WMT'17), which illustrates their potential for improving and evaluating biomedical machine translation systems. Furthermore, we suggest additional applications for multilingual natural language processing using these resources, and plan to extend resource coverage to additional text genres and language pairs.

Keywords: Parallel corpus, biomedical domain, multilingual applications

1. Introduction

Machine translation (MT) is currently being used for a variety of tasks and domains. It is known to play an important role in supporting readers' access to textual documents in a language other than their native language or for communicating in real time. The accuracy of MT systems has improved in recent years thanks to the availability of large collections of parallel and/or comparable corpora. In turn, these resources could be leveraged by deep learning methods, which created a paradigm shift for MT.

MT plays an important role in the health domain. For instance, it has the potential to enable patients to read documents written in a language in which they are not fluent and to hold a conversation with foreign health professionals in case of accidents or health issues in a foreign country. Further, it allows patients to access health information which is only available in a foreign language, for instance, in the case of disease outbreak with origin in other countries (e.g., Zika virus outbreak in Brazil (Bueno, 2017)).

MT can also support researchers to access scientific literature only available in a foreign language, for instance, when working on tropical diseases specific of a region or even when moving to another country for research purposes (Walker, 2016). Finally, MT can also support the biomedical natural language processing (BioNLP) domain when processing documents in languages other than English for which no specific NLP tools are available. This is often the case for clinical discharge reports that are usually only available in the local language. In such cases, researchers could translate the original document into English and rely on state-of-the-art BioNLP tools that are available for English (**?**). The biomedical and health domain is well known for its complex nomenclature, for which specific language resources and tools have been developed, e.g., lemmatizers (Liu et al., 2012). Therefore, specific training and test datasets are also necessary to precisely translate biomedical document across languages. However, despite its importance for the general population and researchers, there are very few parallel and comparable corpora specific for this domain.

In this paper, we present an overview of the state-of-the-art on parallel and comparable corpora for the biomedical domain. In a scoping review of existing resources, we characterize the resources available by language pairs and document type and provide pointers to more in-depth descriptions of the resources. Additionally, we present the parallel corpora that we assembled and built, such as EDP (French/English), ReBEC (Neves, 2017) (Portuguese/English) and Scielo (Neves et al., 2016) (French/English, Portuguese/English and Spanish/English). For the latter, we provide details on the corpus construction, insights on the data and their utilization for the biomedical task (Bojar et al., 2016; Jimeno Yepes et al., 2017) of the Conference for Machine Translation (WMT). All corpora are available in our repository in GitHub[1].

2. Related Work

One of the first efforts that involved the development of large-scale shareable parallel corpora for the biomedical domain was the OPUS collection that contained medical documents from the European Medicines Agency (EMEA) (Tiedemann, 2012)[2]. A number of biomedical parallel (Widdows et al., 2002; Ozdowska et al., 2005; Deleger et al., 2009) and comparable corpora (Chiao and Zweigenbaum, 2004) have been used for terminology translation only. Similarly, the Mantra project (Kors et al., 2015;

[1] http://github.com/biomedical-translation-corpora/corpora
[2] `http://opus.lingfil.uu.se/EMEA.php`

Hellrich et al., 2014) provided corpora of biomedical articles automatically annotated for named-entity recognition for English, Spanish, French, German and Dutch. The corpora included MEDLINE titles, EMEA documents and patents in the biomedical field. While the goal of this project was to leverage annotation transfer from English to other languages to expand terminology coverage in languages other than English, to our knowledge, the corpus has not been used for machine translation.

After general purpose machine translation systems were found to perform poorly on medical text (Zeng-Treitler et al., 2010), the use of domain-specific data was investigated to improve MT system performance in the biomedical field. MEDLINE titles and terms from the Unified Medical Language System (UMLS) (Lindberg et al., 1993) were investigated first, due to easy availability (Wu et al., 2011; Jimeno Yepes and Névéol, 2013). Abstract sentences were also found useful but difficult to obtain and share due to license issues (Jimeno Yepes et al., 2013).

Recent work relied on a corpus of Cochrane Systematic Review Abstracts translated from English to French by professional translators at the French Cochrane Center. Preliminary work showed this dataset to be a good resource for domain-specific machine translation (Neveol et al., 2013). Follow-up work further developed the corpus using post-edited machine translation, which allowed the collection of a rich annotated parallel corpus(Ive et al., 2016)[3].

To encourage the community to take an interest in biomedical MT, recent challenges specifically provided targeted resources for system training and evaluation. The medical translation task at WMT 2014 (Bojar et al., 2014) included some parallel biomedical collections: MuchMore, various patents, Wikipedia titles, Khesmoi search queries, and vocabulary lists extracted from the UMLS. The biomedical track at WMT 2016 (Bojar et al., 2016) provided new resources for French, Portuguese and Spanish with the Scielo corpus (Neves et al., 2016). The biomedical track at WMT 2017 (Jimeno Yepes et al., 2017) continued to offer new resources for French (EDP corpus), Portuguese and Spanish (extension of the Scielo corpus). The task also relied on the UFAL corpus, which comprises some of the previously released sources (EMEA, patents, ...) as well as newly crawled online patient information covering a total of ten language pairs.

Previous work also reported comparable biomedical corpora collected from the Web for Spanish, Arabic and Japanese (Moreno-Sandoval and Campillos-Llanos, 2013). However, text seem to come from different sources and no study on the equivalence between the texts in the various languages seemed to have been carried out.

Table 1 presents a list of biomedical parallel corpora used in the literature. We provide information regarding the text genre and language pairs covered by each source as well as pointers to a description of the resources.

In the following sections, we describe more specifically the recent development of the EDP, ReBec and Scielo corpora.

3. Method for Corpus Development

For all corpora we produced, we carried out the following procedure: (a) document retrieval or download; (b) document parsing and processing; (c) document (sentence) alignment; and (d) quality checking.

Document retrieval and download. Document retrieval varies depending on the document collection, some are readily available for download while others need to be crawled from the corresponding Web site.

Document parsing and processing. After download, the documents need to be parsed in order to extract the relevant text, e.g., title and abstract in the case of scientific publications. Documents in the different languages are then paired based on identifiers in the source websites. This step depends on the format of the obtained documents, whether XML or HTML format. Finally, we split the document into sentences using standard NLP tools.

Document (sentence) alignment. The corpora described in this section do not result from organized professional translation. For this reason, the texts were not translated sentence by sentence as is often the case for professional translation of technical documents. Empirical inspection of the corpora suggests that while some of the documents reflect sentence by sentence translation, others were created more freely and the content in one language could be structured differently in the other language. We made the hypothesis that documents could nonetheless be aligned at the sentence level and we relied on automatic tools for performing the alignment. We identified alignment tools based on an evaluation of alignment for literary texts which is a genre that also features fuzzy alignment(Xu et al., 2015).

Quality checking. After automatically aligning the sentences of the documents, we manually checked a sample of our corpora. This was carried out using the Appraise (Federmann, 2010) tool, and we evaluated whether the aligned sentences where correct or whether more information was available in one language or the other. Native speakers of each foreign language were responsible for this task.

4. Application to Three Biomedical Corpora

Here we describe the three corpora that we developed and highlight the differences regarding the particular tools that we used for the various steps above.

EDP We identified five open access CC-BY journals, referenced EDP Sciences[4] as having content in French and in English: the articles were originally written in French but the journals also publish the titles and abstracts in English, using a translation provided by the authors. Three journals are listed by the publisher under *Health*: "Actualités Odonto-Stomatologiques" and "Médecine Buccale Chirurgie Buccale", which are journals addressing dentistry and "Les Cahiers de Myologie", a journal addressing muscle medicine. Two journals are listed under *Life & Environmental Sciences*: "Cahiers Agriculture" and "Oilseeds and fats, Crops and Lipids". A list of the journal URLs was ob-

[3] http://www.translatecochrane.fr/corpus/

[4] http://www.edpsciences.org

Corpus	genre	languages (other than English)	reference
Cochrane	Systematic Review (SR) abstracts	fr	(Ive et al., 2016)
COPPA, PaTr	Patents	de,fr	(Bojar et al., 2014)
EDP	Article titles and abstracts	fr	ibid.
EMEA	Medication description	cs,da,de,el,es,et,fi,fr,hu,it lt,lv,mt,nl,pl,pt,ro,sk,sl,sv	(Tiedemann, 2012)
Himl*	Patient information and SR abstracts	cs,de,fr	(Jimeno Yepes et al., 2017)
Khresmoi	Short medical search queries	cs,de,fr	(Pecina et al., 2013)
MEDLINE	Article titles	de,es,fr,hu,pl,tu	(Wu et al., 2011)
MuchMore Springer	Article titles and abstracts	de	(Widdows et al., 2002)
ReBEC	Clinical Trial summaries	pt	(Neves, 2017)
Santé Canada	Patient information	fr	(Deleger et al., 2009)
Scielo	Article titles and abstracts	es,fr,pt	(Neves et al., 2016)
UFAL*	Medical web crawl	cs,de,es,fr,hu,pl,ro,sv	(Jimeno Yepes et al., 2017)
UMLS	Metathesaurus	cs,da,de,el,es,eu, et,fi,fr,he,hu,hr it,ja,ko,lt,lv,nl,no,pl,pt,sv,tr,zh	(Lindberg et al., 1993)

Table 1: Overview of biomedical parallel corpus. We use ISO 639-1 two-letter language codes. A star indicates resources that include previously developed corpora as well as new data

Corpus	Tokens	Count method
EDP		
EN	56,684	`wc -w` on txt files
FR	62,333	`wc -w` on txt files
ReBEC		
EN	625,881	reported by (Neves, 2017)
PO	665,325	reported by (Neves, 2017)
Scielo		
EN	20,337,385	script `BioC2txtWithCounts.py`
ES	21,651,629	available on GitHub
EN	525,866	reported by (Neves et al., 2016)
FR	735,486	reported by (Neves et al., 2016)
EN	18,769,613	script `BioC2txtWithCounts.py`
PT	18,573,561	available on GitHub

Table 2: Content of open biomedical parallel corpus.

tained [5] and crawled [6] on March 15, 2017. The html pages were parsed to extract the titles and abstracts in French and English as well as the author names. Any articles lacking some of this information were discarded.

The dataset was pre-processed for sentence segmentation using the Stanford CoreNLP toolkit[7] for use in the WMT17 biomedical task. A manual reference for sentence segmentation was then created independently by revising baseline segmentation after the punctuation marks: full stop, interrogation point, exclamation point and colon.

Based on the manually validated sentence segmentation, the dataset was aligned automatically at the sentence level using YASA(Lamraoui and Langlais, 2013). Manual evaluation conducted on a sample set suggests that 94% of the sentences are correctly aligned, with about 20% of the sentence pairs exhibiting additional content in one of the languages.

MEDLINE vernacular titles MEDLINE indexes journals in languages other than English that publish a title and abstract in English. In this case, MEDLINE citations include the title of the article in the vernacular language in addition to English. This has been used to develop parallel corpora to train machine translation methods (Wu et al., 2011; Jimeno Yepes et al., 2013).

We have retrieved the MEDLINE citations for articles in French, Spanish and Portugese available before the first WMT biomedical task [8]. We collected titles in English and vernacular (Spanish, French and Portuguese). Titles are already aligned since they typically can be considered as one sentence. It can be noted that while our work was limited to the languages of interest in the WMT biomedical track, parallel titles and/or abstracts may also be retrieved for other languages. For instance, the query `chinese [la]` returns 286,151 results on September 29, 2017, and parsing the MEDLINE xml result file could yield several thousand aligned titles and abstract sentences for the relevant citations.

ReBEC As already described in (Neves, 2017), the construction of the ReBEC corpus followed the workflow described in the previous section. The Website site of the Brazilian Clinical Trials Registry [9] provides ways to easily download the trials in XML format, which was further

[5] Using `http://www.xsitemap.com/`

[6] Using the perl utility wget

[7] `https://stanfordnlp.github.io/CoreNLP/`

[8] using the pubmed queries `french [la]`, `spanish[la]`, `portugese [la]`.

[9] http://www.ensaiosclinicos.gov.br/

parsed. However, given the various elements (sub-sections) in a trial, e.g., inclusion criteria, exclusion criteria, and given that some of these appear multiple times in the document, the automatic alignment of parallel documents is not straightforward.

Scielo Scielo (Scientific Electronic Library Online) [10] is a database of open access scientific publications with a focus on developing and emerging countries, and especially on Latin America. All publications in Scielo are available under either the Creative Commons Attribution-Noncommercial 3.0 Unported (cc-by-nc) or Attribution 3.0 Unported (cc-by) licenses, which makes all documents suitable for redistribution and research purpose.

We developed a corpus based on Scielo (Scielo corpus (Neves et al., 2016)) using the following procedure. We crawled the Scielo site and retrieved articles periodically from Scielo. Our crawling has its starting point in the pages that list all journals from the "Biological Sciences" and "Health Sciences" subjects. These categories are used to compose the two datasets, with the corresponding names, of our corpus. Despite being distinct categories in Scielo, these are overlapping categories, as there are many journals that belong to both of them. From the list of journals, it is possible to retrieve a list of all issues of a particular journal, which is available in the regional web sites of Scielo in distinct countries, such as Brazil, Chile or Colombia. The HTML page of the journal's list of issues was further parsed to retrieve the page containing the list of articles of a given issue.

Finally, we downloaded the page of a particular article and parsed the HTML code in order to extract the title and the abstract of each publication. Titles and abstracts were subsequently stored and indexed in the SAP HANA database. All translations of the abstracts in Scielo are the original texts provided by the authors of the publications, who are presumably not professional translators, and who may not have native proficiency in both languages. After the initial version of the corpus produced in 2016, we are using the same procedure to update the corpus on a yearly basis for the ENES and ENPT language pairs. For ENFR, there were no new documents added in 2017.

5. Results

5.1. Datasets Descriptive Statistics

Table 2 presents detailed statistics of the contents of the biomedical parallel corpora that we developped. Table 3 presents an overview of the corpora with the training and test set splits that were offered thoroughout the WMT campaigns.

5.2. Quality Assessment

We also provide a summary of the correct alignment rate for the various corpora, as shown in Table 4. The alignment was automatically carried out using the respective tools as previously described and a sample was manually checked using the Appraise tools for ReBEC and Scielo, and manual inspection of text files for EDP.

[10] http://www.scielo.org

For EDP the manual reference for sentence segmentation provides an evaluation of Stanford sentence segmentation, which comes to 0.77 F-measure on the French portion and 0.81 F-measure on the English portion. Overall, error analysis reveals that the segmentation errors produced by the Stanford tool mainly result from segmentation of the section titles in structured abstracts (Introduction, Material and Method, Results...) which were considered as separate segments by the manual reference but not by the tools. Other errors occur due to organism names (e.g. `E. coli`, which may cause a sentence boundary to be set after "E.").

5.3. Data format

All corpora presented in the previous section are available from GitHub in the BioC format (Comeau et al., 2013), a standard XML format in the BioNLP community.

6. Discussion

In this section we present a short discussion on some interesting topics that raised during both the corpus construction and its use in our shared tasks.

6.1. Lessons learned during corpus construction.

The challenges of building parallel corpora for the biomedical include the identification of high quality relevant data that can be shared with the community. Technical issues then lie with the identification of adequate tools for sentence segmentation and alignment.

Sentence segmentation: we relied on tools which are non-specific for the biomedical domains, such as Stanford CoreNLP, OpenNLP and SAP HANA. However, we did observe issues. A specific discussion of sentence segmentation errors is reported in (Neves, 2017) for ReBEC. For the EDP corpus, we used intially used Stanford CoreNLP for sentence segmentation (in the version of the corpus distributed at WMT17). Then, we manually validated sentence segmentation in both languages in order to create a reference corpus that may be used to train and evaluate sentence segmentation tools. Therefore, Updated versions of the corpus reflect the manual sentence segmentation.

Sentence Alignment: GMA was used for Scielo and ReBEC. Due to difficulties to install GMA, Yasa(Lamraoui and Langlais, 2013) was used for EDP; however, Yasa may be limited to the language pair en/fr. We can refer readers to (Xu et al., 2015) for a discussion and evaluation of alignment tools for a specialized domain (literary texts). Nevertheless, both tools provided good automatic alignments (?). Additionally, GMA was used for two languages (es and pt) and two document types (scientific publications and clinical trials).

6.2. Differences across the corpora.

Despite the corpora presented in this work have been converted into a similar layout, we observed some differences across the results we obtained. These differences are mostly related to particularities of the corpora, such as its format. One such example is the lower rate of correct alignments for the ReBEC corpus that was due to problems when parsing the document format rather that the alignment tool itself, though some few errors could have come from GMA.

Training sets	fr/en	pt/en	es/en
MEDLINE	612,797/idem	74,286/idem	285,408/idem
Scielo	1,135/∼9,500	83,839/∼650,000	93,528/∼750,000
ReBEC		1188/∼23,000	

Test sets	fr/en	pt/en	es/en	en/fr	en/pt	en/es
Scielo	500/∼5,000	1000/∼8,000 189/1,897	1000/∼9,000 158/1,180	500/∼5,000	1,000/∼8,000 188/1,806	1000/∼9,000 158/1,082
EDP	85/699			84/750		

Table 3: Overview of the training and test sets. We present the number of documents and sentences in each corpus. Statistics for the MEDLINE dataset corresponds to both documents and sentences, given that it consists only of titles. For the Scielo test set collection, we present details of the WMT'16 (first row) and WMT'17 (second row) test sets.

Corpus	Rate of correct alignment
EDP	94%
ReBEC	67%
Scielo	79%-85%

Table 4: Summary of the rate of correct alignment for the corpora. No alignment was necessary for the MEDLINE titles.

6.3. Gaps to be addressed.

We can see from table 1 that there is no clinical corpus or datasets from the social media. Also, some languages benefit from better coverage than others depending on the corpora: DE, ES, FR, PT. Finally, we do not yet cover Asian languages, which we plan to address in the future through collaboration with members of the BioNLP community. Typically, we intend to create an abstract collection from MEDLINE as indicated in section 4.

7. Conclusion

We presented a scoping review of the various parallel corpora that are available for the biomedical domain. To our knowledge, this is the first survey of biomedical parallel corpora. In addition we detailed the development of corpora that we recently provided for training and evaluating biomedical machine translation systems. The collections cover a total of four languages (including English) and various types of documents, such as scientific publications and clinical trials, from various sources and databases. Further, these corpora have been evaluated on the scope of two shared tasks and are freely available for the scientific community either for MT or other NLP tasks. Finally, future work will contribute towards the inclusion of additional languages, e.g., German, as well as other documents types, e.g., health-related news and clinical reports.

8. Acknowledgements

This work was supported in part by the Agence Nationale pour la Recherche (French National Research Agency) under grant number ANR-13-JCJC-SIMI2-CABeRneT. The authors would like to thank Arthur Boyer for his contribution to the EDP corpus.

9. Bibliographical References

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., Soricut, R., Specia, L., and Tamchyna, A. (2014). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Logacheva, V., Monz, C., Negri, M., Neveol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., Verspoor, K., and Zampieri, M. (2016). Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation (WMT16) at the Conference of the Association of Computational Linguistics*, pages 131–198. 0.

Bueno, F. T. C. (2017). Vigilância e resposta em saúde no plano regional: um estudo preliminar do caso da febre do Zika vírus. *Ciêcia & Saúde Coletiva*, 22:2305 – 2314, 07.

Chiao, Y. and Zweigenbaum, P. (2004). Aligning words in french-english non-parallel medical texts: effect of term frequency distributions. In *Stud Health Technol Inform*, volume 107, pages 23–7.

Comeau, D. C., Islamaj Doğan, R., Ciccarese, P., Cohen, K. B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M., Valencia, A., Verspoor, K., Wiegers, T. C., Wu, C. H., and Wilbur, W. J. (2013). Bioc: a minimalist approach to interoperability for biomedical text processing. *Database*, 2013:bat064.

Deleger, L., Merkel, M., and Zweigenbaum, P. (2009). Translating medical terminologies through word alignment in parallel text corpora. *J Biomed Inform*, 42(4):692–701, Aug.

Federmann, C. (2010). Appraise: An open-source toolkit for manual phrase-based evaluation of translations. In *In LREC*.

Hellrich, J., Clematide, S., Hahn, U., and Rebholz-Schuhmann, D. (2014). Collaboratively annotating multilingual parallel corpora in the biomedical domain - some mantras. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Ive, J., Max, A., Yvon, F., and Ravaud, P. (2016). Diagnosing high-quality statistical machine translation using traces of post-edition operations. In *Proceedings of the LREC 2016 Workshop: Translation evaluation - From*

fragmented tools and data sets to an integrated ecosystem, pages 55–62, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Jimeno Yepes, A. and Névéol, A. (2013). Effect of additional in-domain parallel corpora in biomedical statistical machine translation. In *Proceedings of the 4th International Workshop on Health Document Text Mining and Information Analysis with the Focus of Cross-Language Evaluation (Louhi 2013)*.

Jimeno Yepes, A., Prieur-Gaston, E., and Neveol, A. (2013). Combining medline and publisher data to create parallel corpora for the automatic translation of biomedical text. *BMC Bioinformatics*, 14(1):146.

Jimeno Yepes, A., Névéol, A., Neves, M., Verspoor, K., Bojar, O., Boyer, A., Grozea, C., Haddow, B., Kittner, M., Lichtblau, Y., Pecina, P., Roller, R., Rosa, R., Siu, A., Thomas, P., and Trescher, S. (2017). Findings of the WMT 2017 Biomedical Translation Shared Task. In *Proceedings of the Second Conference on Machine Translation (WMT17) at the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, Copenhagen, Denmark, September.

Kors, J. A., Clematide, S., Akhondi, S. A., van Mulligen, E. M., and Rebholz-Schuhmann, D. (2015). A multilingual gold-standard corpus for biomedical concept recognition: the mantra gsc. *Journal of the American Medical Informatics Association*, 22(5):948–956.

Lamraoui, F. and Langlais, P. (2013). Yet another fast, robust and open source sentence aligner. time to reconsider sentence alignment? In *XIV Machine Translation Summit*, Nice, France, Sept.

Lindberg, D., Humphreys, B., and McCray, A. (1993). The unified medical language system. *Yearb Med Inform*, 1:41–51.

Liu, H., Christiansen, T., Baumgartner, W. A., and Verspoor, K. (2012). Biolemmatizer: a lemmatization tool for morphological processing of biomedical text. *Journal of Biomedical Semantics*, 3(1):3, Apr.

Moreno-Sandoval, A. and Campillos-Llanos, L. (2013). Design and annotation of multimedica - a multilingual text corpus of the biomedical domain. *Procedia - Social and Behavioral Sciences*, 95:33 – 39. Corpus Resources for Descriptive and Applied Studies. Current Challenges and Future Directions: Selected Papers from the 5th International Conference on Corpus Linguistics (CILC2013).

Neveol, A., Max, A., Ivanishcheva, Y., Ravaud, P., Zweigenbaum, P., and Yvon, F. (2013). Statistical machine translation of systematic reviews into French. In *Proc Workshop on optimizing understanding in multilingual hospital encounters – TIA 2013*, pages 10–13.

Neves, M., Jimeno Yepes, A., and Névéol, A. (2016). The Scielo Corpus: a Parallel Corpus of Scientific Publications for Biomedicine. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Neves, M. (2017). A parallel collection of clinical trials in portuguese and english. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 36–40, Vancouver, Canada, August. Association for Computational Linguistics.

Ozdowska, S., Neveol, A., and Thirion, B. (2005). Traduction compositionnelle automatique de bitermes dans des corpus anglais/franÃ§ais alignÃ©s. In *Proceedings of the 6th International Conference on Terminology and Artificial Intelligence*, pages 83–94.

Pecina, P., Dušek, O., Hajič, J., and Urešová, Z. (2013). Khresmoi query translation test data 1.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Walker, C. (2016). Learn the local lingo to get ahead. *Nature*, 7607(534):425–7, Jun 16.

Widdows, D., Dorow, B., and Chan, C.-K. (2002). Using parallel corpora to enrich multilingual lexical resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*. European Language Resources Association (ELRA).

Wu, C., Xia, F., Deleger, L., and Solti, I. (2011). Statistical machine translation for biomedical text: are we there yet? In *Proc AMIA Annu Symp*, pages 1290–9, Nov.

Xu, Y., Max, A., and Yvon, F. (2015). Sentence alignment for literary texts. *Linguistic Issues in Language Technology*, 12(6), Oct.

Zeng-Treitler, Q., Kim, H., Rosemblat, G., and Keselman, A. (2010). Can multilingual machine translation help make medical record content more comprehensible to patients? In *Stud Health Technol Inform*, volume 160, pages 73–7.

Medical Entity Corpus with PICO Elements and Sentiment Analysis

Markus Zlabinger[1], Linda Andersson[1], Allan Hanbury[1], Michael Andersson[2], Vanessa Quasnik[3], Jon Brassey[4]
[1]TU Wien, [2]Stockholm University, [3]Uppsala University, [4]Trip Database
Austria, Sweden, United Kingdom
[1]forename.surname@tuwien.ac.at, [2]micche@gmail.com, [3]vanessa@quasnik.de, [4]jon.brassey@tripdatabase.com

Abstract

In this paper, we present our process to establish a PICO and a sentiment annotated corpus of clinical trial publications. PICO stands for Population, Intervention, Comparison and Outcome — these four classes can be used for more advanced and specific search queries. For example, a physician can determine how well a drug works only in the subgroup of children. Additionally to the PICO extraction, we conducted a sentiment annotation, where the sentiment refers to whether the conclusion of a trial was positive, negative or neutral. We created both corpora with the help of medical experts and non-experts as annotators.

Keywords: sentiment analysis, PICO, medical corpus, annotation

1. Introduction

Text mining, like data mining or knowledge discovery, is a process to discover implicit knowledge and potentially useful patterns from large text collections. For machine learning (ML) algorithms, it is essential that the labeled data is designed in such a way that the optimal learning for an algorithm is achieved.

We extracted PICO (Figure 1) elements from the text of publications of clinical trial results in order to improve a medical search mechanism of clinical questions. In this paper, we present the process of creating an annotated, phrase-level PICO (Population, Intervention, Comparison, Outcome) corpus and a sentence-level sentiment analysis corpus. In contrast, in other PICO annotation approaches (Boudin et al., 2010; Kim et al., 2011; Wallace et al., 2016), the PICO elements were only labeled on a sentence-level or abstract-level.

The PICO elements are associated with different aspects of noun phrases and domain terminology. The Population (P) elements generally consist of a patient description (e.g. children, men) with one or more post modifications; e.g., *patient over forty with type 2 diabetes*. The Intervention (I) respectively Comparison (C) describes a treatment method (e.g. drug treatment, surgery) and the Outcome (O) describes the aim of a conducted study (e.g. reduce pain).

Our contributions can be summarized in three main points: i) creation of a corpus dataset labeled with PICO elements on a phrase-level, ii) a sentiment analysis of the Outcome, i.e. if an Intervention had a positive effect on a target population or not, and iii) description of a new annotation methodology for community annotation of medical data, which requires less domain-specific knowledge.

The rest of the paper is organized as follows: In Section 2, existing PICO corpora and annotation schemata are presented. Section 3 contains a description of the data and the developed annotation interfaces. The conclusion and future work are described in Section 4.

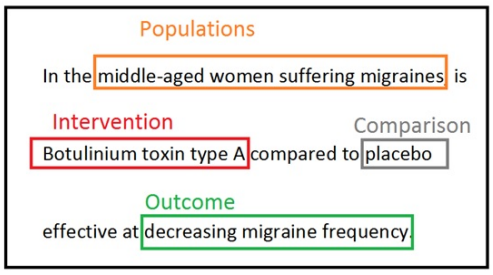


Figure 1: Example of PICO elements in a sentence

2. Related Work

As mentioned, there are existing datasets based on PICO elements. For instance, (Huang et al., 2006) presented PICO frames as a knowledge representation by analysing 59 real-world, primary-care clinical questions. They obtained the questions from the Family Practice Inquiries Network (FPIN) and Parkhurst Exchange. (Demner-Fushman and Lin, 2007) annotated PICO elements in 275 PubMed abstracts and used the data to build a clinical question-answering system.

(Kim et al., 2011) established a corpus of 1,000 medical abstracts, which were manually annotated with 6 different categories (Background, Population, Intervention, Outcome, Study Design, Others) on a sentence-level by medical experts. (Boudin et al., 2010) extracted 260,000 abstracts from PubMed. They limited their domain to English abstracts with publication dates from 1999 to 2009; and used the publication categories Humans, Clinical Trial (CT) and Randomized Controlled Trial (RCT). They exploited the sentence headings that occur in some abstracts (e.g. Results, Methods, ...). Some of these headings are an indicator for certain PICO elements; for example, the headings *Participants* or *Sample* indicate a Population. Based on the sentences that occur within certain headings, they created a dataset and evaluated several ML classifiers on it.

In (Wallace et al., 2016), they exploited an already existing

semi-structured resource, the Cochrane Database of Systematic Reviews (CDSR). They derived supervised distant supervision from the CDSR resource in order to obtain data to train a PICO extraction model. In a follow-up study (Marshall et al., 2017), they used a modified version of this semi-automatic annotated dataset to create a prototype to extract and synthesize medical evidence information from clinical trial articles.

In all of the above studies, domain experts were involved in the data creation process. An expert annotation task is defined as a task that requires several years of knowledge or a specific profession in order to understand and conduct the task correctly (Xia and Yetisgen-Yildiz, 2012). Expert annotators are rather expensive and given the amount of data required to train the end-applications, are a bottleneck in the development of domain-specific text mining applications.

Moreover, the expert annotation schema is not a guarantee for high quality annotations, since domain knowledge alone is not enough to annotate sought entities. The technical aspect of annotation also requires an understanding of what an algorithm can learn from the labeled data (Xia and Yetisgen-Yildiz, 2012). A well designed annotation task requires expert knowledge in the text domain as well as in fields related to linguistics, computational linguistics and Natural Language Processing (NLP), i.e. when annotating an entity, it is important to consider what an algorithm can learn from it. The technical gap between knowledge of the text domain and the requirement of the algorithms to learn the sought entities has been a topic of several publications (Uzuner et al., 2010; Yetisgen-Yildiz et al., 2010; Xia and Yetisgen-Yildiz, 2012). Due to the technical gap between domain experts and artificial intelligence experts, the annotation design for domain specific text genres generally are time consuming, expensive and the outcome is uncertain.

There are also less expensive annotation schemata, such as Crowdsourcing or Community Annotation. These could be alternatives if the domain specific annotation task is designed well. The Crowdsourcing schema makes use of the online labour via annotation providers such as Amazon's Mechanical Turk or Crowd Flower. Depending on the task, these schemata can obtain good results at a low cost (Yetisgen-Yildiz et al., 2010). In the Community Annotation schema, annotations are gathered from the research community that is interested in a particular task and thereby have some pre-knowledge of the target domain and task, which can be beneficial (Uzuner et al., 2010).

3. The Annotation Process

As mentioned in Section 1, the final goal was to extract the PICO elements and the sentiment from medical publications. Since there was no appropriate data available to create an automatic PICO approach or sentiment classifier, we created it ourselves with the help of expert and non-expert annotators. The PICO annotation task requires a different level of linguistic information and also information from different medical domains, which further increases the complexity of the annotation task. For example, the decision of whether labeling an element as Population or Intervention depends on the context; i.e., in a different context, a drug can be part of a Population and in another context, part of an Intervention (see Table 1 for examples).

In addition, to the PICO annotation, we included a sentiment analysis annotation for a subset of the PICO labeled data. For the sentiment analysis, we defined three classes: positive, negative and neutral. The class selection depends on the outcome of the Intervention compared to the Comparison. That is, if an Intervention is better (e.g. more effective, less adverse effects) than its Comparison, the sentiment is positive. On the other hand, an Intervention that did not perform better than its Comparison, should be classified as negative. All other cases are classified as neutral.

We developed two annotation interfaces (one for the sentiment and one for the PICO) and tested them with a group of 6 persons who come from different backgrounds: linguists, biologists, medical experts and students. Step by step, we updated the interface to create a more effective annotation environment for the annotators. The main goal of these updates was to improve the agreement between the annotators; because, if not even humans can agree on where the PICO elements are located or what sentiment a publication has, an algorithmic approach will most certainly also fail to do so. In addition, better agreements mean that the resulting dataset is more reliable and therefore it will be easier to create a well performing automatic detection approach.

3.1. Data Collection

We were provided with 1.5M PubMed titles and abstracts from Trip[1] of which a subsample was used in this first attempt to establish a PICO and sentiment corpus. Since not all publication types are of interest for sentiment analysis and PICO extraction, we used exclusively Randomized Control Trials (RCTs), which contain the following key components: an intervention-arm (*aspirin*), a comparison-arm (*placebo*), an outcome (*Aspirin is more effective than placebo*) and finally, a group of people who are randomly assigned to the intervention-arm or comparison-arm (*men with headache were randomized to either [...]*).

3.2. Annotation Infrastructure

All interfaces for the sentiment and the PICO annotation were implemented by using a mixture of HTML5, JavaScript and PHP5. The submitted annotations were saved in a MySQL database. The publications (i.e. abstract, title) and the user information (i.e. username, user id, etc.) were also stored in the MySQL database.

To increase the agreement between annotators, we provided two guideline documents: one for the PICO annotation task and one for the sentiment annotation task. The guidelines can be seen as a reference manual that can be referred to for difficult cases, but also as an introduction on how to accurately identify the important text parts that should be annotated. The guidelines were updated based on the annotations that we got from the users in small-scale test runs.

3.3. The PICO Annotation Tool

In this section, the different prototypes of the PICO interface are presented. In total, we created three versions of the PICO annotation tool. For each version, the agreement

[1] https://www.tripdatabase.com/

Example	Population	Intervention
Adverse effects of aspirin in men who take vitamin C regularly	men who take vitamin C regularly	aspirin
Adverse effects of vitamin C in men	men	vitamin C
Effects of paracetamol in patients who underwent bankart repair	patients who underwent bankart repair	paracetamol
Bankart repair in patients with shoulder instability	patients with shoulder instability	Bankart repair

Table 1: Depending on the context, treatment methods can be part of a Population or an Intervention.

between annotators was computed. The aim was to successively reach better agreements after each interface update.

3.3.1. First Prototype (version 0)

In the first version of the annotation tool (see Figure 2), the user was asked to mark text within the title or the abstract. Then the marked text could be assigned to one of the four PICO classes with a single button click. Additionally, we allowed open text input for cases where the PICO information was only implicitly stated; for example, *placebo-controlled trial* would mean that the Comparison is placebo. We also offered an advice system that was based on rules crafted with Stanford's TokensRegex (Chang and Manning, 2014). TokensRegex is a rule based framework and used for information extraction. It is similar to regular expressions, but applied over NLP components (e.g. part of speech tags, word tokens) rather than single characters.

3.3.2. Second Prototype (version 1)

Since the open text input from version 0 lead to low inter-annotation agreements of about 20%, we developed a more restricted interface with respect to user interaction for the second prototype. In this version, we first split the publication text into sentences and then each sentence into its tokens. To do so, we used Stanford's CoreNLP (Manning et al., 2014), which is an NLP toolkit that includes sentence splitting and tokenization. To give additional guidance, we provided semantic labels for some of the tokens; for example, diseases, drugs or persons were labeled, as illustrated in Figure 3. The semantic labels for the medical information were generated by using GATE's BioYodie pipeline (Wu et al., 2018), which is a tool for Named Entity Recognition in medical documents. To label Person elements, we simply used a static lookup list that consisted of 44 person keywords (e.g. patients, seniors, children).

In order to do an annotation in the second prototype, the annotator selects one sentence and afterwards, he/she selects the start and end token of the PICO element by simply clicking on them, i.e. open text input was prohibited in this version's interface. Afterwards, a pop-up window opens where the PICO type is selected (see Figure 3). Finally, the annotator selects one of the four PICO classes.

3.3.3. Third Prototype (version 2)

In the third prototype (final version), two changes were made: First, we decided to drop the Outcome from the PICO annotation task since it appeared to be too diverse to reach a reasonable inter-annotator agreement, i.e. only PIC was annotated. Second, we introduced a confidence selection where the annotator could state how confident he/she was, in his/her annotation. We offered three options: Low Confidence, Medium Confidence and High Confidence (default). With the third prototype, we achieved acceptable agreements of around 45% for the Intervention/Comparison, and 55% for the Population, in a small-scale test run. The third prototype is illustrated in Figure 3, which is, besides the two mentioned changes, identical to the previously described interface (i.e. version 1).

We decided to establish the first version of the dataset using majority voting; e.g., if two or more annotators labeled the same part of a text as Population, we considered it as a Population annotation. Based on this majority voting strategy, we started the final annotation run with our six annotators. We distributed 50 unique documents to each annotator and then 50 community documents, which were identical for all 6 annotators. This document distribution was repeated until a total of 500 documents were assigned to each annotators' account. To sum it up, each annotator annotated 250 unique and 250 community documents, which makes a total of $(6 \times 250) + 250 = 1750$ annotated RCTs.

We observed from the annotated dataset that the experts had a tendency to add the design (e.g. randomized, blind) of the trial to either the Population or Intervention. For the Population, they occasionally forgot to mark the entire noun phrase. Meanwhile, the non-experts had difficulties in identifying Populations that had no reference to a Person entity (e.g. *apsirin in headache* VS *aspirin in men with headache*).

3.4. The Sentiment Annotation Tool

For a subset of the PIC annotated corpus, the sentiment was annotated. We differentiated between two types of RCTs:

- Type 1: The abstract contains a conclusion section, as is the case for the abstract shown in Figure 4. In this case, we asked the annotator to select a sentiment of either positive, neutral (default) or negative. Afterwards, by clicking submit, the sentiment annotation is saved in the database.

- Type 2: The abstract does not contain a conclusion section, as is the case for the abstract shown in Figure 5. In this case, we asked the annotators to click on the first sentence where he/she thinks that the conclusion starts. This clicked sentence and all subsequent ones were then listed and a sentiment could be selected for each one.

With the developed interface, it was possible to achieve an inter-annotator agreement of 80%. Note: Before computation of the agreement, the negative and neutral classes were merged, since negative sentiments occurred too rarely (in $\sim 10\%$ of the cases).

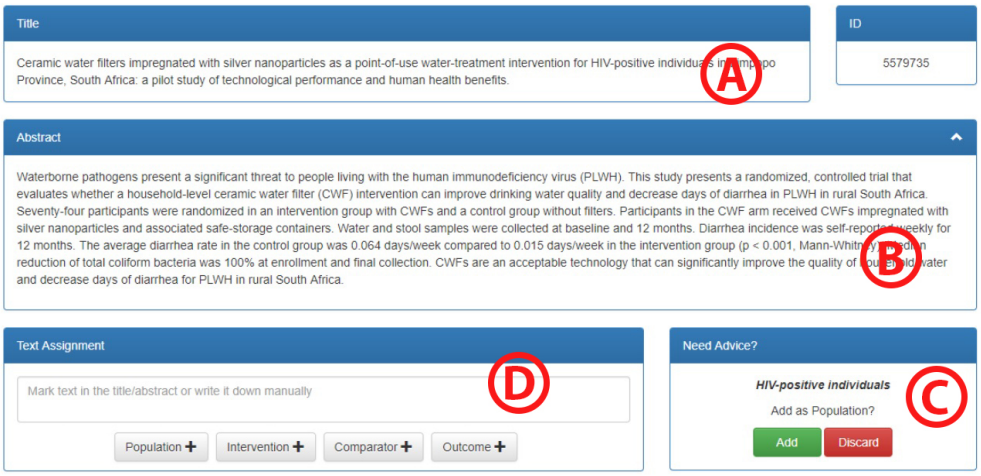


Figure 2: First version of the PICO annotation tool: (A) Title, (B) abstract, (C) advice system and (D) open text input.

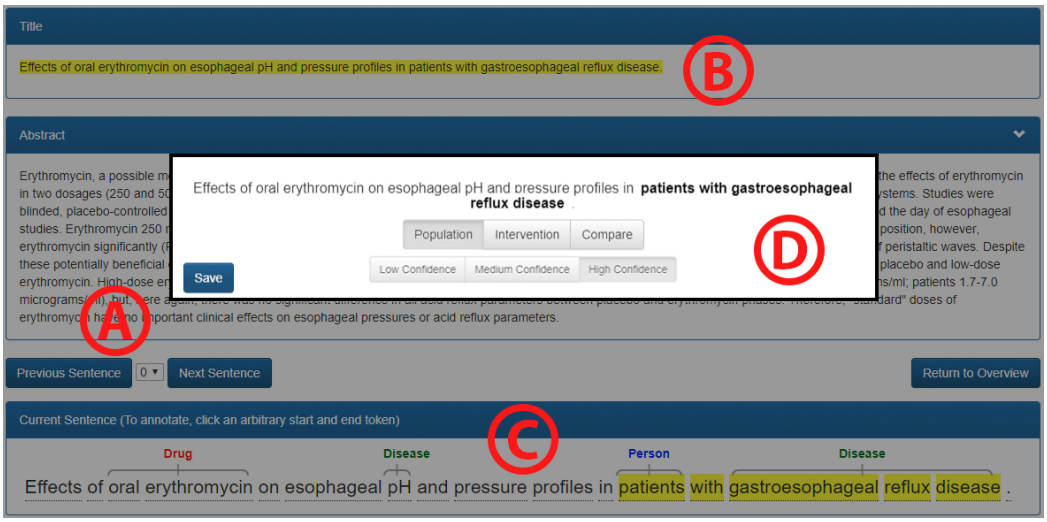


Figure 3: Final version of the PICO annotation tool: (A) Sentence navigation, (B) active sentence (yellow background), (C) active sentence split into single word units (tokens) and finally, after selecting a start and end token, a pop-up window (D) is shown and used to submit an annotation for either P, I or C.

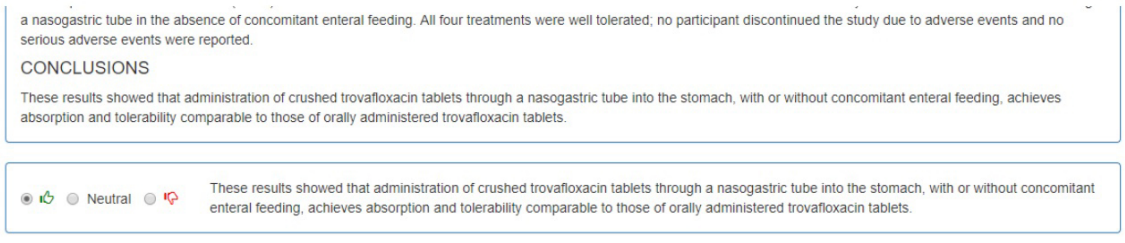


Figure 4: (Type 1) The conclusion sentence(s) are shown immediately.

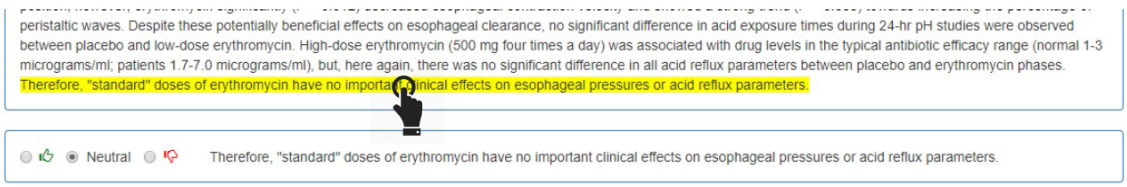


Figure 5: (Type 2) The starting sentence of the conclusion is selected by the annotator.

Since we already reached reasonable agreements in the first version of the annotation tool, we started the final annotation run in which we distributed 200 community and 200 unique documents to each annotator.

4. Conclusion

We have presented the process of establishing a PIC annotated corpus on a phrase-level. We collected a total of 1750 annotated RCTs (250 overlapping) by the annotations of both experts and non-experts. We also labeled a smaller set of these RCTs (1,400) with a sentiment. From our first version of the annotation interface, we increased the annotation agreement from 20% to 55% for the PIC elements. For the sentiment annotation analysis, we reached agreements of 80%.

We have developed an annotation tool for PIC and sentiment analysis, ready to be used in community annotation tasks. Furthermore, we discovered that the PIC annotation task can be conducted by non-experts, if the data is prelabeled with semantic categories, such as persons, drugs or diseases. We only observed minor annotation differences between non-experts and experts. The next step is to turn these two annotation tasks into a community annotation effort in order to collect more annotated data. As soon as we have increased the data, we plan to release part of the corpus to the research community.

5. Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 644753 (KConnect).

6. Bibliographical References

Boudin, F., Nie, J.-Y., Bartlett, J. C., Grad, R., Pluye, P., and Dawes, M. (2010). Combining classifiers for robust PICO element detection. *BMC Medical Informatics and Decision Making*, 10(1):29.

Chang, A. X. and Manning, C. D. (2014). TokensRegex: Defining cascaded regular expressions over tokens. Technical Report CSTR 2014-02, Department of Computer Science, Stanford University.

Demner-Fushman, D. and Lin, J. (2007). Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1):63–103, March.

Huang, X., Lin, J., and Demner-Fushman, D. (2006). PICO as a knowledge representation for clinical questions. In *AMIA 2006 Symposium Proceedings*, pages 359–363.

Kim, S. N., Martinez, D., Cavedon, L., and Yencken, L. (2011). Automatic classification of sentences to support evidence based medicine. *BMC Bioinformatics*, 12(2):S5.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Marshall, I., Kuiper, J., Banner, E., and Wallace, B. (2017). Automating biomedical evidence synthesis: Robotreviewer. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2.

Uzuner, Ö., Solti, I., Xia, F., and Cadag, E. (2010). Community annotation experiment for ground truth generation for the i2b2 medication challenge. *Journal of the American Medical Informatics Association*, 17(5):519–523.

Wallace, B. C., Kuiper, J., Sharma, A., Zhu, M. B., and Marshall, I. J. (2016). Extracting PICO sentences from clinical trial reports using supervised distant supervision. *Journal of Machine Learning Research*, 17(132):1–25.

Wu, H., Toti, G., Morley, K. I., Ibrahim, Z. M., Folarin, A., Jackson, R., Kartoglu, I., Agrawal, A., Stringer, C., Gale, D., Gorrell, G., Roberts, A., Broadbent, M., Stewart, R., and Dobson, R. J. (2018). SemEHR: A general-purpose semantic search system to surface semantic data from clinical notes for tailored care, trial recruitment, and clinical research. *Journal of the American Medical Informatics Association*.

Xia, F. and Yetisgen-Yildiz, M. (2012). Clinical corpus annotation: challenges and strategies. In *Proceedings of the Third Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM' 2012) in conjunction with the International Conference on Language Resources and Evaluation (LREC), Istanbul, Turkey*.

Yetisgen-Yildiz, M., Solti, I., Xia, F., and Halgrim, S. R. (2010). Preliminary experience with Amazon's mechanical turk for annotating medical named entities. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 180–183. Association for Computational Linguistics.

Word Embedding Approach for Synonym Extraction of Multi-Word Terms

Amir Hazem and Béatrice Daille
Laboratoire des Sciences du Numérique de Nantes (LS2N)
Université de Nantes, 44322 Nantes Cedex 3, France
Amir.Hazem@univ-nantes.fr,Beatrice.Daille@univ-nantes.fr

Abstract

The acquisition of synonyms and quasi-synonyms of multi-word terms (MWTs) is a relatively new and under represented topic of research. However, dealing with MWT synonyms and semantically related terms is a challenging task, especially when MWT synonyms are single word terms (SWTs) or MWTs of different lengths. While several researches addressed synonym extraction of SWTs, few of them dealt with MWTs and fewer or none while MWTs synonyms are of variable lengths. The present research aims at introducing a new word-embedding-based approach for the automatic acquisition of synonyms of MWTs that manage length variability. We evaluate our approach on two specialized domain corpora, a French/English corpus of the wind energy domain and a French/English corpus of the breast cancer domain and show superior results compared to baseline approaches.

Keywords: Synonym extraction, Multi-word terms, Compositionality, Word embeddings

1. Introduction

Synonyms acquisition has mainly concerned single word terms (SWTs) using a variety of approaches such as: lexicon-based approaches (Blondel and Senellart, 2002), multilingual approaches (Wu and Zhou, 2003; van der Plas and Tiedemann, 2006; Andrade et al., 2013), distributional approaches (Lin, 1998; Hagiwara, 2008), etc. However, exploring multi-word terms (MWTs) and their synonyms or semantically related terms can be useful in many applications such as: word sense disambiguation, machine translation, information retrieval, text simplification, etc. MWTs are motivated combinations that clearly convey the concept they designate. The requirement of term transparency argues in favor of compositional semantics for complex terms. Compositionality means that the whole meaning can be deduced from the meaning of its components and the syntactic rule by which they are combined (Partee et al., 1990). Pirrelli et al. (2010) claim that the most productive compounds are compositional (at least weak compositional) constructions. Synonymic variants of multi-words exhibit multiple phenomena ranging from compositional multi-word terms synonyms of the same length such as: *wind turbine*/*wind machine*[1]; MWT synonyms of variable length such as: *wind farm*/*wind power plant*; to non compositional MWT synonyms such as: *pole tower*/*mast*.

Few works addressed the acquisition of MWT synonyms. The main approaches that have been proposed in the experimental literature deal with the acquisition of synonyms of MWTs that are compositional and often of the same length. Synonym extraction approaches implement the principle of compositionality by substituting parts of the MWT by synonyms provided by a dictionary (Hamon and Nazarenko, 2001), or by distributional analysis (Hazem and Daille, 2014).

It has been recently shown that words, phrases, sentences, paragraphs and more generally, pieces of texts of any length can be efficiently represented by word embeddings using operations on vectors and matrices like addition or multiplication (Mitchell and Lapata, 2010; Mikolov et al., 2013b; Socher et al., 2011; Mikolov et al., 2013b; Le and Mikolov, 2014; Kalchbrenner et al., 2014; Kiros et al., 2015; Wieting et al., 2016; Arora et al., 2017; Hazem et al., 2017). For phrase representation, Mikolov et al. (2013b) have shown for instance that the embedding vector of the phrase *Volga river* is similar to the addition of the embedding vector of *Volga* and the embedding vector of *river*. The addition property that word embbeding models exhibit offers key information for representing phrases and by extension MWTs and there synonyms or quasi-synonyms. Drawing inspiration from these findings and based on the principle of compositionality and distributed approaches, we propose several techniques based on word embedding models to deal with synonyms acquisition of MWTs. More specifically, we extend the work of Hazem and Daille (2014) and explore synonym extraction of single word terms and multi-word terms of variable lengths. Our first proposition is an extension of the Semi-compositional approach using word embeddings to extract synonyms of parts of MWT. Our second proposition is a Full-compositional approach based on the additive property of word embeddings to extract synonyms of the entire MWT. We conduct several experiments on two specialized datasets that is: a French/English wind energy corpus and a French/English breast cancer corpus. The obtained results of the proposed approaches outperform the state of art baseline approaches.

The remainder of this paper is organized as follows. Section 2. describes the state of art approaches as well as our proposed techniques. Section 3. describes the different linguistic resources used in our experiments. The experimental setup and the obtained results on the wind energy and the breast cancer corpora are respectively presented in Sections 4. and 5. Section 6. initiates a discussion regarding the obtained results and finally, we conclude our work in Section 7.

2. Approaches

In this section we first describe the two main baseline approaches that deal with MWTs synonyms acquisition that

[1] In the renewable energy domain.

is, the compositional approach and the semi-compositional approach. Then, we develop our proposed techniques that is: semi-compositional word embeddings and full-compositional word embeddings approaches. Except the last approach, all these methods hypothesise that MWT semantics is compositional and thus that a synonym of a MWT could be obtained by substituting one of the component parts by a synonymic expression at a given syntactic position. They differ according to how they provide synonym components.

2.1. Compositional Approach

The compositional approach substitutes one of the component of the MWT by one of its synonyms provided by a synonym dictionary. The synonym MWT is considered as valid if and only if it can be found in the corpus. For instance, given the MWT *collecteur général* 'general collector' extracted from the wind energy corpus (cf. Section 3.), several synonyms of *général* are proposed by dictionary of synonyms: *habituel*, *ordinaire*, *commun*, ... The MWT *collecteur commun* 'common collector' which is the correct synonym of *collecteur général* is validated as it occcurs in the wind energy corpus.

Hamon and Nazarenko (2001) defined three rules to extract synonymy relations by assuming a compositional semantics. Given the multi-word candidate terms $CCT_1 = (T_1, E_1)$ and $CCT_2 = (T_2, E_2)$ and $syn(CT_1, CT_2)$ a synonym relation between the candidate terms CT_1 and CT_2, the following inference rules are used:

- R_1: $T_1 = T_2 \wedge syn(E_1, E_2) \supset syn(CCT_1, CCT_2)$
- R_2: $E_1 = E_2 \wedge syn(T_1, T_2) \supset syn(CCT_1, CCT_2)$
- R_3: $syn(T_1, T_2) \wedge syn(E_1, E_2) \supset syn(CCT_1, CCT_2)$

In rule R_1, the heads are identical and the expansions are synonymous, while in rule R_2 heads are synonymous and expansions are identical. Finally, R_3 is a generalization of rules R_1 and R_2. If the compositional approach of Hamon and Nazarenko (2001) is based on a dictionary of synonyms, it can be generalized using external resources that provide semantically related words such as Wordnet for instance. Nonetheless, this approach remains resource dependent. To alleviate this drawback, Hazem and Daille (2014) proposed an approach based on distributional analysis that does not need a dictionary of synonyms or external thesauri and extracts synonyms and semantically related words automatically from the corpus. We present their approach in the next section.

2.2. Semi-Compositional Approach

Like the compositional approach, the semi-compositional variant is based on the principle of compositionality of MWTs. The main difference lies on the nature of the substituted elements of the MWT. It is no longer constrained by the sole relation of synonymy like in Hamon and Nazarenko (2001). Hazem and Daille (2014) generalized the substitution on MWT elements to semantically related terms of any type. They extended the compositional rules R_1 and R_2 by replacing $syn(CCT_1, CCT_2)$ which means synonym relation between CCT_1 and CCT_2 by $sem(CCT_1, CCT_2)$, which means semantic relation between CCT_1 and CCT_2. R_1^G corresponds to the generalized rule R_1 (respectively, R_2^G corresponds to the generalized rule R_2) and T_1, T_2, E_1, E_2 can be MWTs. In addition, they remove the rule R_3 relying on the results of Hamon and Nazarenko (2001) where they have shown that R_3 is the less productive and reliable rule. They obtained the two following rules:

- $R_1^G : T_1 = T_2 \wedge sem(E_1, E_2) \supset sem(CCT_1, CCT_2)$
- $R_2^G : E_1 = E_2 \wedge sem(T_1, T_2) \supset sem(CCT_1, CCT_2)$

For example, the synonym of *énergie renouvelable* 'renewable energy' can be obtained by first extracting each part of the MWT; then, finding the semantically related words of *énergie* 'energy' and/or *renouvelable* 'renewable' with distributional methods; finally, filtering all expressions using monolingual specialized corpora. In the next paragraph we introduce the distributional approach that is used to extract semantically related terms.

Distributional Approach Instead of using a dictionary that will provide synonyms of each lexical element of the MWT, another way to do it is by exploiting distributional relationships. The distributional approach is based on the assumption that words with similar meanings are more likely to share similar contexts. Hence, each word is represented by its context which corresponds to all its surrounding words in the corpus. The surrounding words are often delimited by a window of size n (n is often small 3, 5 or 7 words). Hereafter the main steps of the distributional approach:

- The context vector $v_{w_i^s}$ of a given source word w_i^s is first built. The vector $v_{w_i^s}$ contains all the words that co-occur with w_i^s within a window of n words that surround w_i^s. Let us denote by $occ(w_i^s, w_j^s)$ the co-occurrence count of w_i^s and a given word of its context w_j^s.
- The process of building context vectors is repeated for all words of the specialized corpus.
- Words of the context vectors are weighted using association measures such as the point-wise mutual information (noted MI) (Fano, 1961), the log-likelihood (noted LLR) (Dunning, 1993) or the discounted odds-ratio (noted LO) (Laroche and Langlais, 2010). These measures aim at strengthening the correlation between a word and all the words of its context vector.
- To extract the semantically related words of a given source word w_i^s, a similarity measure such as the cosine similarity (Salton and Lesk, 1968) (noted COS) or the weighted Jaccard index (noted JAC) (Grefenstette, 1994) is applied between $v_{w_i^s}$ and all the target vectors of the corpus $v_{w_i^s}^t$.

- The semantically related candidates of the word w_i^s are the target words ranked according to their similarity scores.

2.3. Word Embeddings Approaches

We introduce two new techniques for synonyms extraction of multi-word terms. The first technique called *Semi-compositional word embeddings*, follows the principle of the semi-compositional approach based on distributional analysis (Hazem and Daille, 2014). It mainly differs in the procedure of extracting SWTs synonyms or semantically related terms which are parts of MWTs. The second technique called *Full-compositional word embeddings*, is inspired by the idea that phrases can be represented by an element-wise sum of the word embeddings of semantically related words of its parts (Mikolov et al., 2013b). It also follows the principle of sentence representation performed by an element wise addition of word embeddings of its parts (Wieting et al., 2016; Arora et al., 2017; Hazem et al., 2017). We adapt this idea and apply it to MWTs. We also experiment the word-embedding state of art approach of Mikolov et al. (2013b) to extract MWTs. We refer to this baseline as *Distributed representation of phrases* and denote it by *Mikolov* approach.

2.3.1. Semi-Compositional Word Embeddings

The Semi-compositional word embeddings approach is also based on the composition of the elements of MWTs. It can be considered as a variant of the semi-compositional approach introduced in (Hazem and Daille, 2014). The difference resides in the manner of extracting semantically related terms of the SWTs ($sem(E_1, E_2)$ and $sem(CCT_1, CCT_2)$). If to do so, Hazem and Daille (2014) use distributional approach as introduced in Subsection 2.2., here we use distributed models (Mikolov et al., 2013b). We explore the two well-known word embedding representation: the Skip-gram model and the continuous bag-of-word model (CBOW).

2.3.2. Full-Compositional Word Embeddings

The Full-compositional word embeddings approach aims at extracting MWTs synonyms of any length. It provides a joint representation for all the MWTs which facilitates MWTs comparison. If the compositional property is applied after-hand in the previous approaches which is problematic when MWTs are of lengths higher than two, the Full-compositional word embeddings approach integrates it beforehand thanks to the additive property of embedding models. All the MWTs are represented by a single embedding vector. Each MWT is first characterized by an element wise sum of its word embedding elements. Then, the cosine similarity measure is applied to extract MWTs synonyms. The implementation of the Semi-compositional and the Full-compositional approaches can be found here `https://github.com/hazemAmir/FullComp.git`

2.3.3. Distributed Representation of Phrases

We apply Mikolov et al. (2013b) approach originally introduced for phrases to MWTs synonyms extraction. Basically, the approach is two-fold. First, (i) we detect and extract all the MWTs of the corpus, then (ii) we consider them as single tokens and build embedding vectors based on their contexts as it is usually done for words by the skip-gram and the CBOW models. Hence, each MWT is characterized by a single embedding vector. Finally, we use the cosine similarity to extract MWTs synonyms. In this approach, the compositionality property is not taken into account. Also, due to the relatively smaller number of MWTs comparing to SWTs, especially in specialized domains, it might be difficult to build efficient embedding models of MWTs. Nonetheless and for a matter of comparison, it is interesting to report the results of this approach.

CBOW and Skip-gram are two distributed representations introduced by Mikolov et al. (2013b) that capture linguistic regularities, namely the Continuous Bag-of-Words (CBOW) model and the Skip-gram model. The principle of the CBOW model is to combine the representations of surrounding words to predict the word in the middle, while the training objective of the Skip-gram model is to learn how to predict the surrounding words based on the representations of the middle word. If CBOW and Skip-gram exhibit similar architectures, CBOW is faster and is more suitable for large datasets while Skip-gram gives better word representations when monolingual data is small (Mikolov et al., 2013a).

3. Data and Resources

In this section, we describe the data and the different resources used in our experiments.

3.1. Corpora

The experiments have been carried out on the French/English specialized corpus from the domain of wind energy of 400,000 words[2] and the French/English specialized corpus from the domain of breast cancer of 500,000 words.

Wind energy corpus is part of the TTC project[3] and has been crawled from the web using *Babouk* (Groc, 2011) crawler. As search engine requests, several technical words have been used such as *wind*, *energy* and *renewable* for English and *vent*, *énergie* and *renouvelable* for French.

Breast cancer corpus has been extracted from *Istex* portal[4] using as keywords *breast cancer* for English and *cancer du sein* for French. The gathered documents concern the period ranging from 2001 to 2015.

The wind energy and breast cancer corpora have both been pre-processed using tokenization, part-of-speech tagging, and lemmatization.

[2] `http://www.lina.univ-nantes.fr/?Ressources-linguistiques-du-projet.html`

[3] `www.ttc-project.eu/index.php/releases-publications`

[4] `https://api.istex.fr/documentation/`

3.2. Reference Lists

Reference lists have been built from various terminological resources. Only the resources that list synonymic terms in their terminological records have been examined. In such lists or databases, synonyms are not systematically present, and for records including them, synonymic variants are various. Many of them are terms related by other types of semantic relations, such as near-synonymy or hypernymy.

For the French part of the wind energy corpus, we selected the French MWT pairs from the *Terminalf*[5] linguistic resource. From 84 MWTs of the wind energy domain, we obtained 34 French MWT synonyms as a result of filtering out SWT synonyms and after checking that the MWT synonyms occur in the specialized corpora. For English, we selected the MWT pairs from the glossary of wind energy from the online book (Gipe, 2004) and from the linguistic resource *Termium* [6]. As a result of filtering and of corpus projection, we obtained 20 English MWT pairs.

This method has been reiterated in order to build the lists of synonyms of multi-word terms in the breast cancer domain. *Termium* has been used. Here again, by discarding the same types of variants for wind energy. After filtering with breast cancer corpus in each language, the lists of reference of the breast cancer domain contain 20 French terms and 16 English terms associated with their synonyms.

The small size of the reference lists can be explained by the small size of the specialized corpora which contain few specialized terms and few synonymic variants. But a more plausible explanation is that the majority of these synonymic variants are contextual. It is difficult for a terminologist to predict and to detect all synonymic variants that can be produced. Contextual synonymic variants do not generally appear in a dictionary resource (Kremer et al., 2014). To evaluate the Full-compositional approach we built a reference list that contains only pairs of synonyms of variable lengths. Following the same procedure for the above described lists, we built a reference list of 10 French and 9 English pairs of synonyms on the wind energy corpus. Here again the small size of the reference lists is due to the lack of synonyms of variable lengths however it is interesting to use these list as a preliminary result.

4. Experimental Setup

For all the experiments the mean average precision *MAP* (Manning et al., 2008) is used to evaluate the quality of the different approaches.

$$MAP = \frac{1}{|W|} \sum_{i=1}^{|W|} \frac{1}{Rank_i} \quad (1)$$

where $|W|$ corresponds to the size of the evaluation list, and $Rank_i$ corresponds to the ranking of a correct synonym candidate i.

[5] http://terminalf.scicog.fr

[6] http://www.btb.termiumplus.gc.ca/tpv2alpha/alpha-eng.html?lang=eng

English term synonyms

aerogenerator	wind turbine generator
windmill	wind turbine
mast	pole tower
rotor-swept area	reference area
wind farm	wind power plant
vertical axis wind turbine	darrieus rotor
wind turbine	wind machine
power supply	energy supply
power plant	electricity plant
savonius model	savonius type
energy output	energy production
wind farm	wind power station
sea wind farm	offshore wind farm
wind turbine	aeroturbine

French term synonyms

éolienne	moulin à vent
rotor de Savonius	anémomètre
générateur synchrone	alternateur
éolienne à axe horizontal	moulin à hélice
parc éolien	implantation
éolienne à axe vertical	rotor de Darrieus
aérogénérateur	turbine éolienne
aéromoteur	moteur éolien
énergie renouvelable	énergie durable
centrale électrique	centrale éolienne
unité de stockage	dispositif de stoskage
arbre primaire	arbre lent
force du vent	vitesse du vent
aérogénérateur	générateur éolien

Table 1: Examples of English/French synonyms and quasi-synonyms of MWTs recorded in terminology banks of the wind energy domain.

4.1. Dictionary-based Method

We used as first baseline the method proposed in Hamon and Nazarenko (2001). To extract French synonyms of single-word terms we used the on-line dictionary *DES* [7]. *DES* contains 49,168 entries and 201,511 synonym relations. The initial database has been constructed from seven dictionaries. The extraction of English synonyms has been conducted using the lexical database *WordNet* [8]. *WordNet* contains approximately 117,000 synsets. The main relation among words in *WordNet* is synonymy.

4.2. Distributional Method Settings

Using the *distributional method*, three main parameters need to be set: the size of the window used to build the context vectors (Morin et al., 2007; Gamallo, 2008), the association measure (the log-likelihood (Dunning, 1993), the point-wise mutual information (Fano, 1961), the discounted odds-ratio (Laroche and Langlais, 2010),...) and the similarity measure (the weighted Jaccard index

[7] http://www.crisco.unicaen.fr/des/synonyms

[8] http://wordnetweb.princeton.edu/perl/webwn/

(Grefenstette, 1994), the cosine similarity (Salton and Lesk, 1968),...). To build the context vectors we chose a 7-window size. We used MI, LLR and LO as association measures and COS and JAC as similarity measures. We refer to the distributional-based approaches by: Semi-Comp (MI-COS), Semi-Comp (LO-COS) and Semi-Comp (LLR-JAC). Other combinations of parameters were assessed, but on average the chosen parameters turned out to give the best performance.

4.3. Word Embeddings Settings

The second baseline is the distributed representation-based approach that we denote by $Mikolov$. For word embeddings, we used as settings a window size ranging from 1 to 20 words[9], negative sampling of 5, sampling of 1e-3 and training over 15 iterations. We applied both Skip-gram and CBOW models[10] to create vectors of dimension ranging from 50 to 800 dimensions. We used hierarchical softmax for training the Skip-gram model. In the proposed approaches, SG100 stands for using skip-gram (100 dimensions) and CBOW300 stands for CBOW (300 dimensions).

5. Results

The experimental results conducted on the French/English wind energy and breast cancer corpora are presented in Tables 2 and 3.

Method	French	English
Hamon&Nazarenko	0.25	3.63
Mikolov	4.56	6.78
Semi-Comp (MI-COS)	27.4	32.6
Semi-Comp (LO-COS)	26.8	27.2
Semi-Comp (LLR-JAC)	31.4	36.1
Semi-Comp (SG50)	30.9	50.3
Semi-Comp (SG100)	**34.9**	55.9
Semi-Comp (SG200)	34.8	52.7
Semi-Comp (CBOW50)	23.0	49.0
Semi-Comp (CBOW100)	23.7	49.4
Semi-Comp (CBOW200)	23.8	49.4
Full-Comp (SG100)	27.3	57.8
Full-Comp (SG200)	28.9	**58.4**
Full-Comp (SG300)	28.5	55.3
Full-Comp (CBOW50)	22.6	47.0
Full-Comp (CBOW100)	20.1	45.1
Full-Comp (CBOW200)	21.6	44.5

Table 2: Results (MAP%) on the wind energy corpus.

First, we observe the very low results of Hamon&Nazarenko approach. This can be explained by the lack of synonymy relations for SWTs part of MWTS. Second, we observe the slightly better results but still low of Mikolov approach. Here, the small size of the datasets is certainly one of the main reasons that can explain the results. Indeed, embedding models of MWTs can't be efficient with small data size. Concerning the Semi-Comp approach, we notice higher results for both distributional-based and embeddings-based approaches

Method	French	English
Hamon&Nazarenko	4.92	7.03
Mikolov	8.37	9.12
Semi-Comp (MI-COS)	19.9	12.6
Semi-Comp (LO-COS)	27.1	11.0
Semi-Comp (LLR-JAC)	13.9	13.3
Semi-Comp (SG50)	32.1	15.0
Semi-Comp (SG100)	**32.2**	15.2
Semi-Comp (SG300)	27.9	9.60
Semi-Comp (CBOW50)	29.1	15.1
Semi-Comp (CBOW100)	29.2	15.3
Semi-Comp (CBOW300)	29.4	15.8
Full-Comp (SG100)	25.6	17.4
Full-Comp (SG200)	28.0	**18.9**
Full-Comp (SG300)	30.5	16.0
Full-Comp (CBOW100)	24.9	10.6
Full-Comp (CBOW200)	24.9	11.6
Full-Comp (CBOW300)	25.0	10.5

Table 3: Results (MAP%) on the breast cancer corpus.

with a better performance for our word-embeddings adaptation (Semi-Comp(SG))[11]. Overall, the best results are mainly obtained by the Full-Comp approach for English datasets and by the Semi-Comp(SG) approach for French datasets.

Figures 1 shows the performance of the semi-compositional approach (noted SemiCBOW and SemiSG[12]) and the full-compositional approach (noted FullCBOW and FullSG) using CBOW and Skip-gram models while varying the context window size (from 1 to 20) and the dimension size (from 50 to 800). We observe that the best results are obtained using small window size and small dimension size. Overall, the FullSG approach obtains the best performance followed by the SemiSG. The SemiCBOW and FullCBOW obtain lower results in general. The best combination is w=5 and dim=100 for FullSG, w=1 and dim=50 for SemiSG, w=3 and dim=400 for SemiCBOW and w=3 and dim=50 for FullCBOW.

To evaluate the Full-compositional approach we did an extra experiment only on synonyms of variable lengths using the wind energy corpus for French and English. We obtained a MAP score of 10.2% and a recall of 66.6% for English and 4.46% of MAP score and a recall of 40% for French. The state of art and Semi-comp proposed approaches can't be applied for this experiment because they don't deal with MWTs length variability. If the results of Full-Comp approach are still low, this approach offers an alternative to pairs of MWTs synonyms that have different lengths.

6. Discussion

Synonyms extraction of MWTs can be addressed using different strategies. When using compositionality property, dictionary-based approach is beneficial when the dictionary of SWT's synonyms is available as shown in (Hamon and Nazarenko, 2001). However, in many cases this resource is difficult to obtain, one interesting alternative

[9]Figures 1 shows the best window size for each approach.

[10]To train word embedding models we used the gensim toolkit (Rehurek and Sojka, 2010).

[11]Except for the Fr breast cancer dataset.

[12]SemiSg with SG that stands for Skip-gram.

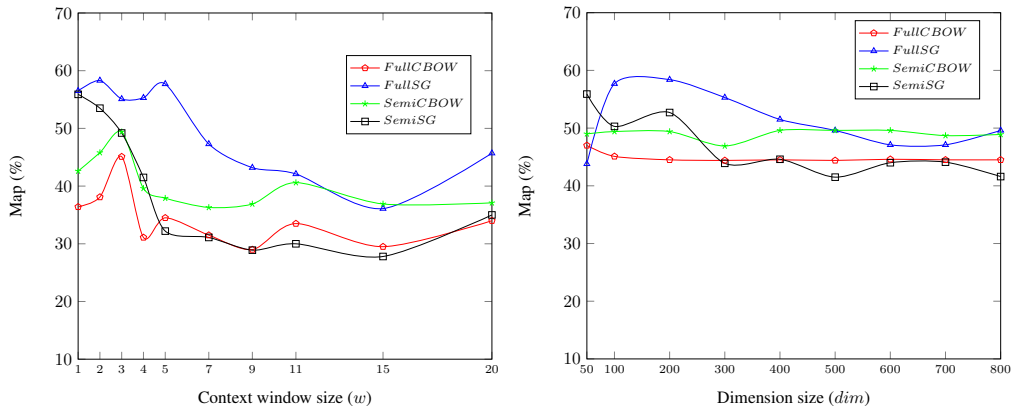


Figure 1: Semi-Comp and Full-Comp comparison while varying the window and dimension size of CBOW and Skip-gram models on the wind energy corpus.

is the distributional-based semi-compositional approach as shown in (Hazem and Daille, 2014). Hence, extracting automatically synonyms of parts of MWT turned out to give better results than looking for them in a dictionary as reported in Tables 2 and 3. With the boom of word embeddings, a straightforward extension of the distributional-based semi-compositional approach is the use of word embeddings to extract synonyms of SWTs. This is the first contribution of this paper. Here again we notice over the results, better performance in most cases using Skip-gram and CBOW models. If the above mentioned approaches are suitable for synonyms extraction of MWTs, they can hardly deal with MWTs synonyms of variable lengths. For instance to extract the synonym of *vertical axis wind turbine* which is *darrieus rotor*, it is not obvious to know that the four-gram length synonym is a bigram in this example. The above cited approaches should know this information or experience all the n-grams possibilities to extract this type of synonyms which is clearly laborious. One alternative which is the second contribution of this paper is the Full-compositional approach. Taking advantage of the additive property of word embeddigs, a MWT can be represented by a single embedding vector which is the result of adding the embedding vectors of its parts. If the Full-compositional approach achieved promising results on the variable length reference lists, the main problem remains its productivity. The question is how to deal with duplicates in the candidates. Filtering is necessary to alleviate repetitive n-grams in different positions. We believe that using sophisticated filtering process[13] based on linguistic patterns for instance, should improve the performance of the Full-Compositional approach. We will pursue this direction in the near future.

7. Conclusion

In this paper, we have proposed different word embeddings approaches for synonyms extraction of MWTs. We have shown that using word embeddings with compositionality and additive composition improve the results comparing to baseline approaches.The full compositional approach which is length independent for MWT representation, has shown the best results in almost all the experiments. If the results on the variable length experiment are still low due to the productivity of this approach, the preliminary results are encouraging since no specific filtering process has been applied. For the future we will pursue this direction by giving more attention to relations between synonyms of variable lengths and their linguistic patterns.

[13]In the variable length FullComp evaluation, we only applied n-grams frequency filtering.

8. Acknowledgments

The research leading to these results has received funding from the French National Research Agency under grant ANR-17-CE38-0008 HORAE (Hours - Recognition, Analysis, Editions) project and the CominLabs excellence laboratory financed by the National Research Agency under reference ANR-10-LABX-07-01 (project LIMAH `limah.irisa.fr`).

9. Bibliographical References

Andrade, D., Tsuchida, M., Onishi, T., and Ishikawa, K. (2013). Synonym acquisition using bilingual comparable corpora. In *International Joint Conference on Natural Language Processing (IJCNLP'13)*, Nagoya, Japan.

Arora, S., Yingyu, L., and Tengyu, M. (2017). A simple but tough to beat baseline for sentence embeddings. In *Proceedings of the 17th International Conference on Learning Representations (ICLR'17)*, pages 1–11.

Blondel, V. D. and Senellart, P. (2002). Automatic extraction of synonyms in a dictionary.

Dunning, T. (1993). Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61–74.

Fano, R. M. (1961). *Transmission of Information: A Statistical Theory of Communications*. MIT Press, Cambridge, MA, USA.

Gamallo, O. (2008). Evaluating two different methods for the task of extracting bilingual lexicons from comparable corpora. In *Proceedings of LREC 2008 Workshop on Comparable Corpora (LREC'08)*, pages 19–26, Marrakech, Marroco.

Gipe, P. (2004). *Wind power: renewable energy for home, farm, and business*. Chelsea Green Pub. Co.

Grefenstette, G. (1994). *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publisher, Boston, MA, USA.

Groc, C. D. (2011). Babouk : Focused Web Crawling for Corpus Compilation and Automatic Terminology Extraction. In *Proceedings of 10th International Conferences on Web Intelligence (WIC'11)*, pages 497–498, Lyon, France.

Hagiwara, M. (2008). A supervised learning approach to automatic synonym identification based on distributional features. In *Proceedings of the ACL-08: HLT Student Research Workshop*, pages 1–6, Columbus, Ohio, June. Association for Computational Linguistics.

Hamon, T. and Nazarenko, A. (2001). Detection of synonymy links between terms: experiment and results. In *Recent Advances in Computational Terminology*, pages 185–208. John Benjamins.

Hazem, A. and Daille, B. (2014). Semi-compositional method for synonym extraction of multi-word terms. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Hazem, A., el amel Boussaha, B., and Hernandez, N. (2017). Mappsent: a textual mapping approach for question-to-question similarity. Recent Advances in Natural Language Processing, RANLP 2017, 2-8 September, 2017, Varna, Bulgaria.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. *CoRR*, abs/1404.2188.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In C. Cortes, et al., editors, *Advances in Neural Information Processing Systems 28*, pages 3294–3302. Curran Associates, Inc.

Kremer, G., Erk, K., Padó, S., and Thater, S. (2014). What substitutes tell us - analysis of an "all-words" lexical substitution corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 540–549.

Laroche, A. and Langlais, P. (2010). Revisiting Context-based Projection Methods for Term-Translation Spotting in Comparable Corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, pages 617–625, Beijing, China.

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. *CoRR*, abs/1405.4053.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98, pages 768–774, Stroudsburg, PA, USA. Association for Computational Linguistics.

Manning, D. C., Raghavan, P., and Schütze, H. (2008). Introduction to information retrieval. Cambridge University Press.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013a). Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Mitchell, J. and Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1439.

Morin, E., Daille, B., Takeuchi, K., and Kageura, K. (2007). Bilingual Terminology Mining – Using Brain, not brawn comparable corpora. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07)*, pages 664–671, Prague, Czech Republic.

Partee, B., Meulen, A., and Wall, R. (1990). *Mathematical Methods in Linguistics*. Studies in Linguistics and Philosophy. Springer Netherlands.

Pirrelli, V., Guevara, E., and Baroni, M. (2010). Computational issues in compound processing. In Sergio Scalise et al., editors, *Cross-disciplinary issues in compounding*, volume 311 of *Current issues in linguistic theory*, pages 271–285. John Benjamins Publishing Company, Amsterdam/Philadelphia.

Rehurek, R. and Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

Salton, G. and Lesk, M. E. (1968). Computer evaluation of indexing and text processing. *Journal of the Association for Computational Machinery*, 15(1):8–36.

Socher, R., Huang, E. H., Pennington, J., Ng, A. Y., and Manning, C. D. (2011). Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Advances in Neural Information Processing Systems 24*.

van der Plas, L. and Tiedemann, J. (2006). Finding synonyms using automatic word alignment and measures of distributional similarity. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics ACL'06*, Sydney, Australia.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2016). Towards universal paraphrastic sentence embeddings. *International Conference on Learning Representations, CoRR*, abs/1511.08198.

Wu, H. and Zhou, M. (2003). Optimizing synonym extraction using monolingual and bilingual resources. In *In Proceedings of the second international workshop on Paraphrasing*, page 72.

A Large Automatically-Acquired All-Words List of Multiword Expressions Scored for Compositionality

Will Roberts, Markus Egg
Humboldt-Universität zu Berlin
Unter den Linden 6, 10099 Berlin, Germany
{will.roberts, markus.egg}@anglistik.hu-berlin.de

Abstract

We present and make available a large automatically-acquired all-words list of English multiword expressions scored for compositionality. Intrinsic evaluation against manually-produced gold standards demonstrates that our compositionality estimates are sound, and extrinsic evaluation via incorporation of our list into a machine translation system to better handle idiomatic expressions results in a statistically significant improvement to the system's BLEU scores. As the method used to produce the list is language-independent, we also make available lists in seven other European languages.

Keywords: multiiword expressions, compositionality, machine translation

1. Introduction

Multiword expressions (MWEs) are phraseological units, which consist of more than one lexeme and exhibit some kind of idiosyncrasy (Sag et al., 2002); such idiosyncrasy may be lexical (*ad hoc*), syntactic (*by and large*), semantic (*middle of the road*), pragmatic (*all aboard*), or statistical (*black and white* but not *white and black*; these are commonly known as *collocations*) (Baldwin and Kim, 2010).

In this paper, we present a new linguistic resource, in the form of a large automatically-acquired all-words list of MWEs, which aims to support future research into semantically idiosyncratic MWEs. Semantically idiosyncratic MWEs, or *idiomatic expressions*, are *non-compositional* in that their meanings cannot be predicted from their parts; these expressions are used frequently to make language more fluent (Jackendoff, 1997), and often contain word senses not found in other contexts. Thus, identifying non-compositional MWEs presents a clear challenge for fields such as automatic machine translation (MT), information retrieval, natural language understanding, natural language generation, question answering, text summarisation, and word sense disambiguation (McCarthy et al., 2007). In recent years, there has been considerable interest in the MWE community in automatically estimating compositionality (Biemann and Giesbrecht, 2011; Reddy et al., 2011; Schulte im Walde et al., 2013; Salehi et al., 2015); however, to the best of our knowledge, this work has hardly been applied to real-world NLP tasks. We set out to distribute a convenient resource representing the best practices gleaned from this work, by automatically scoring the expressions on our list for compositionality.

This paper is structured in the following way: Section 2. lists previous work in this area, while section 3. details our acquisition method. Our resource is then evaluated intrinsically against manually-produced gold standards in section 4., and extrinsically, inside a MT system in section 5..

2. Related work

While the resource introduced in this paper is an all-words list acquired automatically, most existing MWE resources are produced manually and focus on a single part of speech (e.g., noun-noun compounds, verb-noun constructions, verb particle constructions, adjective-noun constructions);[1] some examples of these are used in Section 4..

Other more general resources include machine-readable dictionaries that happen to list MWEs; examples include the TED-MWE bilingual dictionary (Monti et al., 2015), with 2,484 automatically-extracted aligned EN-IT MWEs, and BabelNet (Navigli and Ponzetto, 2012), some of whose 8.5 M entries in 271 languages are MWEs.

MWE research dealing with compositionality tends to focus on methodologies rather than producing resources. There are also MWE compositionality resources that are not targeted towards natural language processing, such as Martinez and Schmitt (2012), who produce a list of 505 non-compositional English phrases for teaching English as a second language. In contrast to the monolingual method we make use of here, some methods to estimate compositionality do so by measuring the relative difficulty of translating an expression into another language; an example is Villada Moirón and Tiedemann (2006), who leveraged parallel corpora to extract Dutch MWEs. However, for languages such as Basque this approach is not feasible, because parallel corpora are very limited in size and number and restricted to few languages (Leturia et al., 2009; Leturia, 2012).

3. Acquisition of non-compositional MWEs

We collect lexical co-occurrence statistics on all words in the English Wikipedia, using the WikiExtractor tool[2] to retrieve plain text from the April 2015 dump (ca. 2.8B words), and using simple regular expressions to segment sentences and words, and remove URLs and punctuation. We perform no POS tagging, lemmatisation, case normalisation, or removal of numbers or symbols; MWE acquisition using unlemmatised text in this way may be useful for capturing the morphological or syntactic fixedness of some idiomatic

[1] Losnegaard et al. (2016) offers a recent survey and `http://multiword.sourceforge.net/`, a list of MWE resources.

[2] `https://github.com/bwbaugh/wikipedia-extractor`

MWEs (e.g., identifying *spill the beans* but not *spill the bean*[3]). We collect word frequency information with the SRILM language modelling toolkit (Stolcke, 2002)[4], counting n-grams ($n \leq 3$), treating MWEs as contiguous[5] bigrams and trigrams), and identify MWE candidates by computing the Poisson collocation measure (Quasthoff and Wolff, 2002)[6] for all bigrams and trigrams (ca. 23M n-grams). This method should be readily extensible to include longer n-grams.

The Poisson measure we use is chosen after an empirical evaluation of several commonly used association measures[7]:

chi χ^2: $\sum_{i,j} \frac{[f_{ij}-f'_{ij}]^2}{f'_{ij}}$

conf confidence (Omiecinski, 2003) $\max[\frac{P(AB)}{P(A)}, \frac{P(AB)}{P(B)}]$

mi mutual information: $\sum_{i,j} P_{ij} \log(\frac{P_{ij}}{P'_{ij}})$

pe permutation entropy (Zhang et al., 2006)

poisson Poisson collocation measure (Quasthoff and Wolff, 2002) $\frac{f'(xy)-f(xy)\log f'(xy)+\log[f(xy)!]}{\log N}$. This is identical up to a constant factor with the "log likelihood measure" introduced by Dunning (1993).

poissonT Poisson balanced for trigrams

ps Piatetsky-Shapiro (Piatetsky-Shapiro, 1991) $P(AB) - P(A)P(B)$

psT Piatetsky-Shapiro balanced for trigrams $P(ABC) - P(A)P(B)P(C)$

ttest t-test: $\frac{f(AB)-f'(AB)}{\sqrt{f(AB)[1-f(AB)/N]}}$

ttestT t-test balanced for trigrams

We estimate the quality of these rankings by searching for known collocations and multiword expressions, and finding the ranks of these known expressions in the lists. We define a good association measure as one which tends to rank these known expressions highly (as operationalised by the Mean Reciprocal Rank). For this comparison, we use manually-constructed lists of multiwords intended as gold standards in MWE acquisition work:

English noun compound (NC) (Nakov, 2008)

English verb particle constructions (VPC) (Baldwin, 2005)

[3]For example, Villada Moirón and Tiedemann (2006) found lemmatisation to be unhelpful for identifying non-compositional MWEs, because of the tendency of idiomatic MWEs to display more morphosyntactic fixedness than literal text.

[4]`http://www.speech.sri.com/projects/srilm/`

[5]Note that, while many MWEs are contiguous (e.g., *in a nutshell*), some may be non-contiguous (e.g., *take a (long) bath*).

[6]This measure is almost identical to the log-likelihood ratio introduced by Dunning (1993).

[7]For a more complete list of association measures commonly used in the MWE acquisition literature, the reader is referred to (Pecina, 2008).

	NC	VPC
χ^2	7.7	2.8
`conf`	1.2	1.1
`mi`	55.0	64.5
`pe`	2.2	2.6
`poisson`	66.3	74.6
`poissonT`	70.0	78.5
`ps`	34.0	77.6
`psT`	32.3	74.7
`ttest`	38.7	79.0
`ttestT`	35.7	73.8

Table 1: Mean Reciprocal Rank ($\times 10^{-7}$) by association measure for two test corpora. Higher values are better.

Score	MWE	Cosine similarities		
0.005	*a front for*	—	0.005	—
0.012	*red tape*	−0.056	0.081	
0.191	*stops short of*	0.285	0.097	—

Table 2: Some compositionality-scored MWE candidates.

Table 1 lists the results for the association measures on these two corpora, demonstrating that the Poisson measure works best for our task.

We then automatically score the million most strongly associated n-grams (i.e., roughly the top 5% of the Poisson-ranked list) for compositionality. Compositionality scores are assigned using a method based on the work of Salehi et al. (2015), which represents the current state of the art. Using word2vec (Mikolov et al., 2013)[8] with the parameters[9] found to be most effective by Baroni et al. (2014), we build a word embedding vector for every simplex word in the vocabulary (ca. 1M types), as well as for each MWE candidate. We then compute the cosine similarity of the vector representation for a MWE candidate with the vectors of its constituent words, and take the arithmetic mean. In scoring the compositionality of a candidate, we do not measure the cosine similarity of the MWE with any stop words it may contain, as stop words may be assumed to be semantically uninformative[10]. Table 2 presents several extracted MWE candidates with their computed compositionality scores and shows that cosine similarity scores with determiners (*a*) and prepositions (*for* and *of*) are ignored.

The embedding vectors are trained on the extracted Wikipedia text, where each occurrence of a MWE candidate is greedily replaced with a single token representing the MWE as a word-with-spaces. The string rewriting is performed efficiently using the Aho-Corasick algorithm (Aho and Corasick, 1975). This greedy rewriting procedure cannot deterministically handle n-grams which overlap with

[8]`https://code.google.com/p/word2vec/`

[9]Continuous bag of words model with 400-dimensional vectors, window size 5, subsampling with $t = 10^{-5}$, negative sampling with 10 samples. We build vectors only for tokens observed 20 times or more in the corpus.

[10]Stop words are taken here to be the 50 most frequent words in the vocabulary.

	Found	Total	Spearman ρ	Pearson's r
F_ENC	631	1042	0.458	0.473
R_ENC	61	90	0.615	0.603
MC_VPC	48	117	0.432	0.379
D_ADJN	64	68	0.525	0.581
MC_VN	132	638	0.392	0.395

Table 3: Correlation of our compositionality-ranked list against manually-constructed gold standards.

other n-grams, so we sort the MWE candidates into 10 disjoint batches such that, for any two candidates e_1, e_2 in the same batch, e_1 is neither a substring, nor a superstring of e_2, and there is no prefix of e_1 which is a suffix of e_2. This sorting is performed greedily, by processing candidates in order of decreasing Poisson score, and assigning each candidate to the first batch for which this property obtains; candidates which cannot be assigned to a batch (ca. 6.8%) are discarded. Each batch thus results in a word embedding model for all single words in the vocabulary, and some subset of the MWE candidates; after computing the compositionality scores, we recombine the candidates from all batches to produce a single list that is sorted in order of increasing compositionality, containing 917,647 expressions.

4. Intrinsic Evaluation

We conducted an in-vitro evaluation of the compositionality scores by measuring correlations against several gold standard datasets from the MWE compositionality literature, which contain human judgements of how predictable the meaning of a MWE is from its constituent words. The datasets are:

F_ENC (Farahmand et al., 2015) 1,042 noun compounds (e.g., "cat fight", "chicken breast", "crash course", etc.) annotated by five judges, with some filtering, resulting in a 5-point Likert scale. Inter-annotator agreement by Fleiss' κ was 0.62. Yazdani et al. (2015) report $\rho = 0.410$ on this dataset.

R_ENC (Reddy et al., 2011) 90 noun compounds (e.g., "snail mail", "guilt trip", etc.) annotated over Amazon Mechanical Turk using a 6-point Likert scale. Inter-annotator agreement by averaged Spearman correlation between rankings was $\rho = 0.686$. Salehi et al. (2015) reported achieving $r = 0.796$.

MC_VPC (McCarthy et al., 2003) 117 verb-particle pairs (e.g., "rule out", "clamp down", etc.) annotated by 3 judges, with averaged scores on a 11-point Likert scale. Inter-annotator agreement with Kendall's Coefficient of Concordance is reported to be $W = 0.594$. The original paper reports $\rho = 0.49$ using a method based on measuring the size of overlap in synonyms of the phrasal verb and in those of the bare ("simplex") verb, using an automatically constructed thesaurus.

D_ADJN (Biemann and Giesbrecht, 2011) $58 + 10 = 68$ compounds (Adj-NN compounds only) from the training and validation sets of the Disco 2011 Shared Task (e.g., "mental health", "soft drink", "small group", etc.). Annotated over Amazon Mechanical Turk using a 11-point Likert scale, with scores averaged over judges. No inter-annotator agreement figures are available. Krčmář et al. (2013) achieved $\rho = 0.54$ using a LSA-based model.

MC_VN (McCarthy et al., 2007) This subset of the resource constructed by Venkatapathy and Joshi (2005) contains 638 verb-object pairs (e.g., "lend money", "turn back", "watch television", etc.) annotated by two judges using a 6-point Likert scale. This list also contains some non-contiguous items (e.g., "lose temper", "beg question", etc.) not found in our list. Inter-annotator agreement by Kendall's $\tau = 0.61$; Spearman rank correlation between annotators: $\rho = 0.71$. Kiela and Clark (2013) reported $\rho = 0.461$.

Table 3 shows the correlation of our compositionality scores against these gold standards. The table lists the size of each gold standard dataset, and its overlap with our resource.
The compositionality ranking accords well with human judgements, with correlation scores not far from the state of the art, and 10–30 percentage points below the human inter-annotator agreement.. In the case of the largest resource, F_ENC, we are not aware of a better correlation than the one we report here. The list is positively correlated with all gold standard judgements, representing a variety of parts of speech, and all correlations are statistically significant. This demonstrates the validity of our compositionality scoring.
The n-gram statistics we collect contain 1,562 of a total 1,931 items from the gold standards; we can take this number to be the count of these compounds which are attested in the English Wikipedia. Our compositionality-ranked list contains only 912 items from the gold standards. Part of this decrease represents the MWE candidates which are discarded due to low association measure scores, and part likely results from MWE candidates lost because they could not be assigned to a batch for compositionality computation. Note that the largest number of missing compounds come from MC_VN, which, as noted, contains many discontinuous (non-n-gram) compounds.

5. Extrinsic Evaluation: MT

To evaluate the utility of our resource for NLP applications, we conduct an extrinsic evaluation by incorporating MWE knowledge into an automatic English-Spanish translation system.
TectoMT (Žabokrtský et al., 2008) is a linguistically sophisticated hybrid MT system which uses a combination of statistical and rule-based components in a modular pipeline model to analyse source language up to a highly abstract (*tectogrammatical*) level of representation; this so-called t-tree is a dependency tree structure containing only nodes for autosemantic words. The morphosyntactic properties of the nodes (t-nodes) in this t-tree are represented by *formemes*, which encode grammatical roles and complements (e.g., *n:subj* for a noun in subject position, or *n:for+X* for a noun preceded by the preposition *for*).
In the *transfer* stage, translation is performed by first **copying** the source language t-tree structure into the target lan-

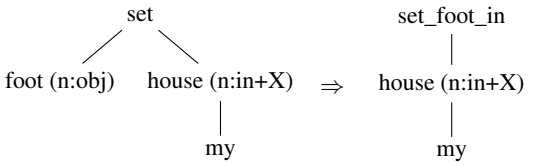


Figure 1: Tectogrammatical reduction of multiple t-nodes (representing the non-compositional MWE *set foot in*) into a single composite t-node.

guage; the formemes and lemmas on each tree node are then translated into the target language using a maximum entropy translation model. The copying done in the first step means that the translation produced by the system is (on the tectogrammatical level) structurally identical to the input; thus, the system operates using the strong assumption that translation can be performed *isomorphically*. The translation model is learnt by analysing parallel corpora using the TectoMT pipeline, and then inducing maxent models for lemma and formeme translation from Giza++ alignments; in the experiments we report here, we train our models on Europarl (Koehn, 2005). Following transfer, further pipeline components generate successively concrete representations in the target language, until the system can produce a linearised string of words as its output translation.

The isomorphicity assumption built into TectoMT generally works well, but is problematic when the source language contains non-compositional MWEs. Therefore, we integrate lexical information about MWEs in the form of our compositionality-ranked list into the TectoMT pipeline, by collapsing multiple t-nodes in a t-tree which represent a single MWE into a single composite t-node. Note that this paradigm will only work for MWEs which can be translated into single lexical nodes in the target language; MWEs which are translated by other multiwords will result in translation failures (i.e., insertion or deletion errors). However, we expect that such failures will happen relatively infrequently[11].

For this work, we perform some semi-automatic filtering of our MWE list, removing several of the more common errors that we observed, using a simple pattern-based filter (e.g., discarding those candidates which begin or end with a conjunction or some form of the copula). We also discard some MWE candidates which are superstrings or substrings of another MWE candidate with a lower compositionality score, when the two candidates have very similar word embedding vectors. This results in the removal of around 11% of the candidates from the list, leaving us with 817,592 MWE candidates.

Immediately prior to the transfer stage, we identify MWEs in the source language greedily by searching on word forms in the input and finding their corresponding t-nodes in the t-tree; we match only sets of nodes in the t-tree that are fully connected to each other by dependency relations (i.e., which are *treelets*). In this search, MWEs with lower compositionality scores are preferred; ties are broken arbitrarily by taking the leftmost match. Successfully matched MWE instances have their lemma altered to a word-with-spaces representation, and are collapsed by deleting dependent MWE nodes and rearranging arguments so that these depend on the new composite node. Figure 1 shows the reduction performed in the analysis of a successfully matched MWE instance[12].

Performing this analysis during training of the TectoMT system allows the translation model to learn how to translate English MWEs observed in the training corpus into Spanish. We record all MWEs seen during training, and use only this list for analysis during testing, to ensure that no MWEs in the test corpus are reduced for which the trained translation model has not learnt any translations (which would create new out-of-vocabulary items). This has the effect of filtering our MWE candidate list, so that, at test time, only those expressions found in the translation training corpus are used to analyse the test data. We manipulate the compositionality value θ as an independent variable, using a threshold to control the number and compositionality of MWEs that are analysed in the source text. For example, with $\theta \leq 0.1$ we restrict the MWE candidate list to contain only those items whose compositionality score is less than or equal to 0.1. Table 4 shows the number of MWEs found in the English section of Europarl for different values of the threshold.

Threshold	Types	Tokens
$\theta \leq 0.1$	1,093	32,956
$\theta \leq 0.2$	5,020	174,015
$\theta \leq 0.5$	90,133	2,808,015

Table 4: Counts of MWEs observed during TectoMT training on Europarl with varying compositionality thresholds.

We train four English-Spanish models on Europarl: a baseline model, which does not analyse MWEs, and three MWE-enabled models, using threshold values of $\theta \leq 0.1$, $\theta \leq 0.2$, and $\theta \leq 0.5$. We test these models on the ACL 2008 shared translation task (Callison-Burch et al., 2008), containing 2,000 sentences (ca. 55 K words) from Europarl. We also build a MWE-rich test corpus by filtering the test split of Europarl (Oct.–Dec. 2000), retaining only sentences that contain one or more highly non-compositional ($\theta \leq 0.1$) MWEs from our list. This produces a small English-Spanish test corpus of 518 sentences (ca. 18K words).

Case-insensitive BLEU scores (Papineni et al., 2002) summarising our results are presented in Table 5, which also shows the counts of MWEs observed during testing. On both test sets, we observe a similar pattern: Analysing MWEs improves translation over the baseline model, but only when using low values of the compositionality threshold; performance falls below the baseline as this threshold is increased. This effect is expected, because it is likely that composite t-nodes representing compositional English MWEs cannot be adequately translated by single lexemes in Spanish.

On the ACL 2008 test set, we observe an absolute improvement over the baseline of $+0.18$ BLEU points (1% relative)

[11] Cf. Uresova et al. (2013), who found in the Parallel Czech-English Dependency Treebank that most verbal MWEs are not translated by other MWEs.

[12] In this example, the preposition *in* has been encoded in the formeme of the t-node under it (*house*) by the TectoMT system, but our analysis will still find this treelet because it can find *set* and *foot*.

Experiment	MWE Counts		BLEU
	Types	Tokens	
ACL 2008 shared task			
Baseline			12.55
$\theta \leq 0.1$	7	17	**12.73** $*$
$\theta \leq 0.2$	39	74	12.66
$\theta \leq 0.5$	715	1,175	11.99
MWE-rich test set			
Baseline			11.59
$\theta \leq 0.1$	20	71	11.39
$\theta \leq 0.2$	37	99	**11.83**
$\theta \leq 0.5$	299	449	11.28

Significance relative to the baseline: $*$: $p < 0.01$

Table 5: TectoMT experimental results: BLEU scores of different MWE-enabled models on two test corpora.

when using the lowest value of the compositionality threshold; this effect is statistically significant at the $p < 0.01$ level[13]. Increasing the threshold to $\theta \leq 0.2$, the improvement is smaller but still positive; the effect is not significant. On the MWE-rich test set, the $\theta \leq 0.2$ model obtains an absolute improvement over the baseline of $+0.24$ BLEU (2% relative); due to the small test corpus size, this effect is not significant ($p = 0.066$). The $\theta \leq 0.1$ model, by contrast, performs more poorly than the baseline. Error analysis does not conclusively explain this, but we have observed the model making mistakes due to instances of non-compositional MWEs, such as *came into force*, which happen to have literal translations in Spanish (*entró en vigor*). The $\theta \leq 0.2$ model appears to contain helpful MWEs, such as *(on) the one hand*, which help to offset these errors.

It is interesting to note that the improvement to BLEU scores is out of proportion to the number of MWEs analysed at test time; for instance, the best improvement seen on the ACL 2008 test set occurs when TectoMT finds only 17 instances of MWEs in the test corpus. We have observed this phenomenon while training models on other parallel corpora, and while using other test sets—sometimes this results in better-than-baseline performance on test sets containing no MWEs at all. We surmise that treating non-compositional MWEs while training TectoMT allows the translation model to learn to ignore spurious translations of polysemous verbs (e.g., *come, enter, set*) and nouns (e.g., *point, term*) which enter into idiomatic expressions; that is, when learning to translate a particular lexeme, the model is not distracted by the translations of MWEs which include that lexeme. E.g., suppose that the analysis of the parallel corpora couples *come to terms* with its Spanish translation *llegar a un acuerdo*. If we identify the English expression as a MWE, we make sure that there is no spurious analysis of *terms* as the English equivalent of *acuerdo* 'agreement' regardless of whether or not *come to terms* shows up in the material to be translated automatically.

[13]In this paper, significance tests use bootstrap resampling, and one-tailed p values are reported (Koehn, 2004). We use the MT-ComparEval software (Klejch et al., 2015), `https://github.com/choko/MT-ComparEval`.

6. Conclusion

We have introduced a new automatically-acquired all-words list of MWEs, automatically ranked for compositionality. Evaluation against manually-created gold standards validates our compositionality scores, and incorporating our list into a MT system to detect idiomatic language gave a statistically significant improvement to the system's BLEU scores.

We used the same language-independent method to build compositionality-ranked lists for other languages (Bulgarian, Czech, German, Spanish, Basque, Dutch, and Portuguese); we make these lists available here without evaluation.

7. Acknowledgements

This work was supported by the EC's FP7 (FP7/2007- 2013) under grant agreement number 610516: "QTLeap: Quality Translation by Deep Language Engineering Approaches".

8. Bibliographical References

Aho, A. V. and Corasick, M. J. (1975). Efficient string matching: An aid to bibliographic search. *Communications of the ACM*, 18(6):333–340.

Baldwin, T. and Kim, S. N. (2010). Multiword expressions. In Nitin Indurkhya et al., editors, *Handbook of Natural Language Processing*, pages 267–292. CRC Press, Boca Raton, USA, second edition.

Baldwin, T. (2005). Deep lexical acquisition of verb–particle constructions. *Computer Speech & Language*, 19(4):398–414.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore. Association for Computational Linguistics.

Biemann, C. and Giesbrecht, E. (2011). Distributional semantics and compositionality 2011: Shared task description and results. In *Proceedings of the Workshop on Distributional Semantics and Compositionality*, pages 21–28. Association for Computational Linguistics.

Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., and Schroeder, J. (2008). Further meta-evaluation of machine translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 70–106. Association for Computational Linguistics.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Farahmand, M., Smith, A., and Nivre, J. (2015). A multiword expression data set: Annotating non-compositionality and conventionalization for English noun compounds. In *Proceedings of the 11th Workshop on Multiword Expressions*, pages 29–33, Denver, CO. Association for Computational Linguistics.

Jackendoff, R. (1997). *The architecture of the language faculty*. MIT Press, Cambridge, MA.

Kiela, D. and Clark, S. (2013). Detecting compositionality of multi-word expressions using nearest neighbours

in vector space models. In *Proceedings of the Short Papers of the Conference on Empirical Methods in Natural Language Processing (EMNLP-13)*, pages 1427–1432.

Klejch, O., Avramidis, E., Burchardt, A., and Popel, M. (2015). MT-ComparEval: Graphical evaluation interface for machine translation development. *Prague Bulletin of Mathematical Linguistics*, 104(1):63–74.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In Dekang Lin et al., editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona. Association for Computational Linguistics.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT Summit 2005*, pages 79–86.

Krčmář, L., Ježek, K., and Pecina, P. (2013). Determining compositionality of word expressions using word space models. In *Proceedings of the 9th Workshop on Multiword Expressions*, pages 42–50, Atlanta, GA. Association for Computational Linguistics.

Leturia, I., San Vicente, I., and Saralegi, X. (2009). Search engine based approaches for collecting domain-specific Basque-English comparable corpora from the Internet. In *Proceedings of the 5th International Web as Corpus Workshop (WAC5)*, pages 53–61.

Leturia, I. (2012). Evaluating different methods for automatically collecting large general corpora for Basque from the web. In *COLING 2012*, pages 1553–1570.

Losnegaard, G. S. G., Sangati, F., Escartín, C. P., Savary, A., Bargmann, S., and Monti, J. (2016). PARSEME survey on MWE resources. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Martinez, R. and Schmitt, N. (2012). A phrasal expressions list. *Applied Linguistics*, 33(3):299–320.

McCarthy, D., Keller, B., and Carroll, J. (2003). Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*, pages 73–80. Association for Computational Linguistics.

McCarthy, D., Venkatapathy, S., and Joshi, A. K. (2007). Detecting compositionality of verb-object combinations using selectional preferences. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 369–379. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Monti, J., Sangati, F., and Arcan, M. (2015). TED-MWE: A bilingual parallel corpus with MWE annotation. In *Proceedings of the Second Italian Conference on Computational Linguistics (CLiC-it 2015)*, pages 193–197, Trento, Italy.

Nakov, P. (2008). Paraphrasing verbs for noun compound interpretation. In *Proceedings of the LREC Workshop: Towards a Shared Task for Multiword Expressions (MWE 2008)*, pages 46–49.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Omiecinski, E. R. (2003). Alternative interest measures for mining associations in databases. *IEEE Transactions on Knowledge and Data Engineering*, 15(1):57–69.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Pecina, P. (2008). A machine learning approach to multiword expression extraction. In *Proceedings of the LREC Workshop: Towards a Shared Task for Multiword Expressions (MWE 2008)*, pages 54–57.

Piatetsky-Shapiro, G. (1991). Discovery, analysis, and presentation of strong rules. In Gregory Piatetsky-Shapiro et al., editors, *Knowledge Discovery in Databases*, chapter 13, pages 229–238. MIT Press, Cambridge, MA.

Quasthoff, U. and Wolff, C. (2002). The Poisson collocation measure and its applications. In *Proceedings of the 2nd International Workshop on Computational Approaches to Collocations*, Vienna, Austria.

Reddy, S., McCarthy, D., and Manandhar, S. (2011). An empirical study on compositionality in compound nouns. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 210–218, Chiang Mai. Asian Federation of Natural Language Processing.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for NLP. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2002)*, pages 1–15, Mexico City, Mexico.

Salehi, B., Cook, P., and Baldwin, T. (2015). A word embedding approach to predicting the compositionality of multiword expressions. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 977–983. Association for Computational Linguistics.

Schulte im Walde, S., Müller, S., and Roller, S. (2013). Exploring vector space models to predict the compositionality of German noun-noun compounds. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 255–265.

Stolcke, A. (2002). SRILM: An extensible language modeling toolkit. In John H. L. Hansen et al., editors, *Proceedings of the 7th International Conference on Spoken Language Processing, ICSLP2002 - INTERSPEECH 2002*, Denver, CO. ISCA.

Uresova, Z., Sindlerova, J., Fucikova, E., and Hajic, J. (2013). An analysis of annotation of verb-noun idiomatic combinations in a parallel dependency corpus. In *Proceedings of the 9th Workshop on Multiword Expressions*, pages 58–63, Atlanta. Association for Computational Linguistics.

Venkatapathy, S. and Joshi, A. K. (2005). Relative compo-

sitionality of multi-word expressions: A study of verb-noun (VN) collocations. In *Natural Language Processing–IJCNLP 2005*, pages 553–564. Springer, Berlin, Heidelberg.

Villada Moirón, B. and Tiedemann, J. (2006). Identifying idiomatic expressions using automatic word alignment. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006): Workshop on multiword expressions in a multilingual context*, pages 33–40, Trento.

Yazdani, M., Farahmand, M., and Henderson, J. (2015). Learning semantic composition to detect non-compositionality of multiword expressions. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1733–1742, Lisbon. Association for Computational Linguistics.

Žabokrtský, Z., Ptáček, J., and Pajas, P. (2008). TectoMT: Highly modular MT system with tectogrammatics used as transfer layer. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 167–170. Association for Computational Linguistics.

Zhang, Y., Kordoni, V., Villavicencio, A., and Idiart, M. (2006). Automated multiword expression prediction for grammar engineering. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties (MWE 06)*, pages 36–44, Sydney. Association for Computational Linguistics.

A Hybrid Approach for Automatic Extraction of Bilingual Multiword Expressions from Parallel Corpora

Nasredine Semmar
CEA, LIST, Vision and Content Engineering Laboratory
F-91191, Gif-sur-Yvette, France
nasredine.semmar@cea.fr

Abstract

Specific-domain bilingual lexicons play an important role for domain adaptation in machine translation. The entries of these types of lexicons are mostly composed of MultiWord Expressions (MWEs). The manual construction of MWEs bilingual lexicons is costly and time-consuming. We often use word alignment approaches to automatically construct bilingual lexicons of MWEs from parallel corpora. We present in this paper a hybrid approach to extract and align MWEs from parallel corpora in a one-step process. We formalize the alignment process as an integer linear programming problem in order to find an approximated optimal solution. This process generates lists of MWEs with their translations, which are then filtered using linguistic patterns for the construction of the bilingual lexicons of MWEs. We evaluate the bilingual lexicons of MWEs produced by this approach using two methods: a manual evaluation of the alignment quality and an evaluation of the impact of this alignment on the translation quality of the phrase-based statistical machine translation system Moses. We experimentally show that the integration of the bilingual MWEs and their linguistic information into the translation model improves the performance of Moses.

Keywords: Bilingual lexicon, Multiword expression, Terminology extraction, Domain adaptation, Statistical machine translation

1. Introduction

A MultiWord Expression (MWE) is a combination of words for which syntactic or semantic properties of the whole expression cannot be obtained from its components (Sag et al., 2002). Such units could be collocations, compound words, named entities, idioms, etc. They constitute an important part of the lexicon of any natural language (Jackendoff, 1997). Bilingual lexicons of MWEs play a vital role in Machine Translation (MT) and Cross-Language Information Retrieval (CLIR) because for a specific domain the specialized vocabulary is largely dominated by MWEs. The manual construction of these lexicons is costly and time-consuming. Word alignment approaches are often used to automatically construct bilingual lexicons from parallel corpora. Several word alignment approaches have been explored (Daille et al., 1994; Barbu, 2004) and many automatic word alignment tools are available, such as Giza++ (Och and Ney, 2000). However, most of these tools are efficient only to align single words (Fraser and Marcu, 2007). In this paper, we describe and evaluate a hybrid approach to automatically extract and align MWEs from an English-French parallel corpus. In contrast to traditional approaches for MWEs alignment which consist in firstly identifying monolingual MWEs candidates and secondly applying alignment to find bilingual correspondences, our approach extracts and aligns MWEs in a one-step process.

The remainder of the paper is organized as follows. We define in Section 2 the notion of Multiword Expression and describe different types of MWEs with examples. In Section 3, we survey previous works addressing the tasks of extracting and aligning MWEs from parallel corpora. Section 4 introduces our hybrid approach to build bilingual lexicons of MWEs from sentence aligned parallel corpora. The experimental results are reported and discussed in Section 5. Finally, we present in Section 6 the conclusion and future work.

2. Multiword Expressions

In Natural Language Processing (NLP), a multiword expression refers to a non-compositional sequence of words whose exact and unambiguous meaning, connotation and syntactic properties cannot be derived from the meaning or connotation of its components (Sag et al., 2002). MWEs are frequently used in written texts and constitute a significant part of the language lexicon. Sag et al. (2002) classify multiword expressions into two main categories: lexicalized phrases and institutionalized phrases. Lexicalized phrases "have at least partially idiosyncratic syntax or semantics, or contain "words" which do not occur in isolation". Institutionalized phrases are "semantically and syntactically compositional, but statistically idiosyncratic".

2.1 Lexicalized Phrases

In a decreasing order of lexical rigidity, lexicalized phrases are broken down into three classes: fixed expressions, semi-fixed expressions and syntactically-flexible expressions.

Fixed expressions are non-compositional sequences of words. They are syntactically and morphologically rigid and undergo neither internal modification nor morphological and syntactical variations (e.g. "*nest of vipers*" in English or "*pomme de terre*" in French). To determine whether or not a sequence of words is a fixed expression, we can use linguistic criteria such as using synonyms or adding words between its components (e.g. "*nest of many black vipers*" in English or "*pomme de jolie terre lointaine*" in French). Fixed expressions can be considered as single entries in the dictionary.

A semi-fixed expression is a non-compositional sequence of words whose components do not contribute to its figurative meaning. Semi-fixed expressions should respect a strict word order and some of them undergo limited lexical and morphological variability such as inflection and some variation in the reflexive form. According to their

characteristics, they can be broken down into three basic categories: non-decomposable idioms, proper names and some compound nominals (Sag et al., 2002).

Non-decomposable idioms do not undergo syntax variability but their components accept lexical changes such as pronominal reflexivity form (e.g. "*wet him-self*", "*wet themselves*"), verbal inflection ("*kick the bucket*", "*kicked the bucket*") or passivization (e.g. "*briser le silence*" or "*le silence est brisé*" in French). Proper Names "are syntactically highly idiosyncratic" (Sag et al., 2002). They can be complex with two or three proper names as components, including person, place and organization names. Compound nominals are syntactically unalterable and undergo number inflection (e.g. "*car park(s)*" in English or "*pomme(s) de terre*" in French).

Unlike semi-fixed expressions, syntactically-flexible expressions undergo a wide degree of syntactic variation such as passivization (e.g. "*The cat was let out of the bag*") and allow external elements to intervene between their components (e.g. "*slow the car down*"). This type of expressions includes verb-particle constructions, decomposable idioms. Particle verbs constructions are made up of a verb whose meaning is modified by one or more particles. They can be either semantically idiosyncratic such as "*brush up on*" or compositional such as "*take after*", "*look out*", "*go back*" and "*run over*". Decomposable idioms tend to be syntactically flexible to some degree that is unpredictable. Semantically, they behave as if their components were linked parts contributing independently to the figurative interpretation of the expression as a whole.

2.2 Institutionalized Phrases

Institutionalized phrases are semantically and syntactically fully compositional, but statistically idiosyncratic (Sag et al., 2002). They occur in a high frequency and their idiosyncrasy is statistical rather than linguistic. They generally allow one available meaning. Institutionalized phrases often refer to "collocations", described as sequences of words that statistically have a high probability to appear together whether they are contiguous or not (e.g. "*make a difference*").

3. Related Work

Automatic identification of MWEs from texts is a real challenge in Natural Language Processing. This is due to the diversity and the complexity of their lexical, syntactic and semantic characteristics (Moon, 1998; Riehemann 2001; Sag et al. 2002). Two approaches have emerged to extract bilingual MWEs from parallel corpora. The first approach consists of acquiring translations of MWEs from parallel corpora in one-step (DeNero and Klein, 2008; Marchand and Semmar 2011). DeNero and Klein (2008) consider, on the one hand, MWEs as phrases composed of contiguous sequences of words that encapsulate enough context to be translatable, and on the other hand, that the problem of finding an optimal alignment between bilingual MWEs can be cast as an integer linear program. Marchand and Semmar (2011) used an approach which followed to some extent that of DeNero and Klein (2008) while they added two scoring functions based on co-occurrence and a seed single word bilingual dictionary. The second approach for extracting bilingual MWEs from parallel corpora, firstly, identifies monolingual MWEs candidates and then applies alignment techniques to find bilingual correspondences (Daille et al., 1994; Blank 2000; Barbu 2004; Deng et al., 2005; Samuelsson et al., 2007; MacCartney et al., 2008; Lefever et al., 2009; Semmar et al., 2011; Bouamor et al., 2012). In the second approach, MWEs extraction can be processed by using symbolic methods based on linguistic patterns (Dagan et al., 1994; Okita et al., 2010; Bouamor et al., 2012), or, through statistical approaches which use automatic measures to rank MWEs candidates (Pearce 2002; Evert and Krenn 2005; Zhang et al. 2006; Villavicencio et al. 2007; Vintar et al., 2008). Finally, MWEs extraction can be done by using hybrid approaches, which combine statistical information with some kinds of linguistic information such as syntactic and semantic properties (Baldwin and Villavicencio 2002; Van de Cruys and Villada Moiron 2007; Caseli et al., 2010). Dagan and Church (1994) proposed to use syntactic analysis to extract terminology. MWEs are then extracted by grouping linguistically related terms. In the same way, Okita et al. (2010) proposed to link across two languages MWEs according to their syntactic and lexical information. Tufis and Ion (2007) introduce a linguistic approach in which they claim that MWEs keep in most cases the same morpho-syntactic structure in the source and target languages. Statistical approaches also have proven to be useful in collecting bilingual MWEs from parallel corpora. Kupiec (1993) introduced the use of machine learning algorithms such as the Expectation Maximization (EM) to extract MWEs. Similarly, Vintar and Fiser (2008) proposed to extract bilingual MWEs by translating MWEs from a well-known language (English) to a low resource language (Slovene) by using machine translation. They have shown that their translation-based approach performs better than using linguistic approaches. However, they did not combine these two kind of approaches. The combination of such approaches enables to extract finer MWEs. In this way, Wu and Chang (2004) and later Boulaknadel et al. (2008), proposed to use syntactic and statistical analysis to extract bilingual MWEs from a parallel corpus. The main aspect of their approach is a monolingual parsing to extract MWEs combined with statistical detection in each language, then, they confront candidates from each side to find bilingual MWEs. Other approaches proposed to use machine translation to translate MWEs candidates found with a syntactic analysis (Seretan and Wehrli, 2007).

4. Building Bilingual Lexicons of MWEs

The process of building MWEs bilingual lexicons from parallel corpora is composed of the following two steps:

1. MWEs extraction and alignment using scoring functions.
2. MWEs candidates filtering using morpho-syntactic patterns.

4.1 Extraction and Alignment of MWEs

In this section, we describe our approach to extract and align MWEs from an English-French parallel corpus in a one-step process (Marchand and Semmar, 2011; Semmar and Marchand, 2017; Semmar and Laib, 2017). This approach is hybrid because it considers the global task of identification and alignment of MWEs as an optimization

problem and it uses external linguistic resources: a seed single word bilingual dictionary and morpho-syntactic patterns. It handles MWEs which are composed of contiguous units. As the only restriction we made is the contiguity of MWEs, the alignment task is a NP-hard problem. We formalize, then, the alignment task as an integer linear programming problem to find an approximated optimal solution (DeNero and Klein, 2008; Marchand and Semmar, 2011).

In this formalization, a sentence pair consists of two word sequences *e* and *f*, e_{ij} is the MWE from between-word positions *i* to *j* of *e*, and f_{kl} is the MWE from between-word positions *k* to *l* for *f*. A link is an aligned pair of MWEs, denoted (e_{ij}, f_{kl}). Each e_{ij} is allowed to be linked with several f_{kl} and each f_{kl} with several e_{ij}. An alignment *a* of the sentence pair (*e*; *f*) is a segmentation of the two sentences in MWEs with the set of links between these MWEs. We use a real-valued function ϕ (objective function) to score links.

$$\phi : \{e_{ij}\} \times \{f_{kl}\} \rightarrow R$$

The score of an alignment *a* is the product of all the links inside it:

$$\phi(a) = \prod_{(e_{ij}, f_{kl}) \in a} \phi(e_{ij}, f_{kl})$$

In order to find the alignment (segmentation + links) that maximizes this score, we, first, introduce binary variables A_{ijkl} denoting whether a link exists between e_{ij} and f_{kl}. Furthermore, we introduce binary indicators E_{ij} and F_{kl} that denote whether some (e_{ij}, .) and (., f_{kl}) appear in *a*, respectively. Finally, we use $W_{ijkl} = log(\Phi(e_{ij}, f_{kl}))$ to transform the product into a sum. When optimized, the integer program yields the optimal alignment[1]:

$$\begin{cases} \max \sum_{i,j,k,l} W_{i,j,k,l} A_{i,j,k,l} & \\ \forall\, x : 1 \leq x \leq |e| & \sum_{i,j:i<x\leq j} E_{i,j} = 1 \quad (1) \\ \forall\, y : 1 \leq y \leq |f| & \sum_{k,l:k<y\leq l} F_{k,l} = 1 \quad (2) \\ \forall\, i, j & \sum_{k,l} A_{i,j,k,l} \geq E_{i,j} \quad (3) \\ \forall\, k, l & \sum_{i,j} A_{i,j,k,l} \geq F_{k,l} \quad (4) \\ \forall\, i, j, k, l & 2 \cdot A_{i,j,k,l} \leq E_{i,j} + F_{k,l} \quad (5) \end{cases}$$

Under the following constraints:

$$\begin{cases} 0 \leq i < |e|, & 0 < j \leq |e|, & i < j \\ 0 \leq k < |f|, & 0 < l \leq |f|, & k < l \end{cases}$$

Constraints (1) and (2) indicate that a word is inside exactly one MWE. Constraint (3) ensures that each MWE in the selected partition of *e* appears in at least one link (and likewise constraint (4) for *f*). Finally, constraint (5) ensures that if a link exists between e_{ij} and f_{kl} ($A_{ijkl} = 1$) then e_{ij} and f_{kl} are in the selected partitions of *e* and *f*. This constraint allows a MWE to be aligned with several other MWEs. This integer program can work with any real-valued scoring function.

Because the only restriction we made on MWEs is their contiguity, the alignment task model can handle the following MWEs:

- Compound nouns: A sequence of words acting as a single noun. These compounds could be proper nouns or common nouns.
- Phrasal verbs: Collocations containing a verb followed by a preposition.
- Verb constructions: Concatenations of a verb and a noun collocation.
- Verb phrase idioms: Verb phrases whose semantics are non-compositional.
- Verb-prepositional phrase constructions: Verbs attached to prepositional phrases without compositional semantics.

The integer linear program describing the alignment task can work with any scoring function. To solve this program, we used two scoring functions.

4.1.1 Scoring Based on Co-occurrence of MWEs

We use a sentence-aligned corpus to compute the co-occurrence score. For each MWE, we consider its presence or absence in each sentence, and thus, the score between two MWEs e_{ij} and f_{kl} is computed as follows:

$$\phi_c(e_{ij}, f_{kl}) = \frac{\sum_{s' \in S} N_{s'}(e_{ij}) \times N_{s'}(f_{kl})}{\sum_{s \in S} N_s(e_{ij}) + N_s(f_{kl}) - N_s(e_{ij}) \times N_s(f_{kl})}$$

Where $N_s(e_{ij})$ is *1* if the phrase e_{ij} of the first language is present in the sentence *s* of the corpus *S* and *0* otherwise. $N_s(f_{kl})$ is similar for the other language. Note that if none of e_{ij} or f_{kl} appears in the whole corpus, the score is set to *0*. Indeed, if two MWEs appear exactly in the same bi-sentences, they are probably translation of each other and the score will be *1*.

As expected with this scoring function, if the program finds an unknown word or if the word co-occurs with no other word in the translated sentence, all the links containing this word will obtain a score equal to *0*. Therefore, the global score of the alignment will be also equal to *0* whatever the other links because the scoring function is multiplicative. In order to overcome this limit, we used an external linguistic resource: a seed bilingual dictionary.

4.1.2 Scoring Based on a Bilingual Dictionary

The bilingual dictionary provides several word-to-word alignments. We want to comply with these alignments as often as possible as we infer that they are mostly correct. The dictionary also gives negative alignment information. Of course, if two words are not aligned by the dictionary we can't take for sure that they shouldn't, and we have to take that into account. The dictionary score is calculated with the following formula:

[1] We used the open source solver GLPK (www.gnu.org/s/glpk/).

$$\phi(e_{ij}, f_{kl}) = \frac{a \times R_1 + b \times R_0}{a \times R_1 + b \times R_0 + c \times N_1 + d \times N_0}$$

R_1 is the number of respected links, R_0 is the number of respected non-links, N_1 is the number of non-respected links, and N_0 is the number of non-respected non-links. The coefficients *a*, *b*, *c* and *d* can be adapted to balance the relative influence of the four terms. We analyzed a small corpus that allowed us to empirically choose the use of the following values: $a = b = c = 1$ and $d = 0.5$. The score is calculated for each part of the bilingual MWEs and then the two of them are multiplied. We have to take into account R_0 and N_0 because otherwise the whole bi-sentence would be the optimal segmentation.

As we can see, this score has a double effect. First, it gives a high score if the bilingual MWEs respect dictionary word-to-word alignment. Second, due to R_0, it sets a threshold score for unknown couples. Both effects can have a positive role in alignment task as we will see in the examples below. The dictionary-based score is not intended to be used separately. It is mixed with co-occurrence score. We used an English-French bilingual dictionary containing 243539 entries with doubles[2]. It is important to point out here that the entries of the English-French bilingual dictionary are in lemmas forms. Therefore, to take full advantage of this dictionary, it is preferable to lemmatize the parallel corpus before extracting and aligning MWEs. However, as some surface forms are similar to lemmas in English and French languages, we experimented the two possibilities. The parallel corpus has been lemmatized using the multilingual analyzer LIMA (Besançon et al., 2010).

4.2 Filtering MWEs Candidates

The result of the previous step (Extraction and alignment of MWEs) is a list of alignment links candidates. Each link is composed of a MWE in the source language and its translation candidate in the target language. This step covers all the categories of MWEs (Compound nouns, Phrasal verbs, etc.).

In order to increase the accuracy of this step, we filter the results, on the one hand, by removing the longer MWEs if shorter MWEs occur in these candidates, and on the other hand, by selecting only MWEs which match with a list of morpho-syntactic patterns (Table 1). The MWEs candidates are composed of sequences of words of size $n \geq 2$ that follow the most frequent Part-Of-Speech patterns. Part-Of-Speech tags of the components of each MWE are provided by the multilingual analyzer LIMA after processing the parallel corpus. We have built manually the list of morpho-syntactic patterns by analyzing the sequences of Part-Of-Speech tags corresponding to the MWEs candidates provided by the first step. We have also used the patterns derived by other research works (Bouamor et al., 2012). At the end, we obtained a set of 25 patterns, most of which are related to noun phrases. However, it is important to note that a same pattern in a source language could have several patterns in the target language. It is for instance the case of the English pattern "Adj-Noun-Noun" which have three equivalent patterns in French "Noun-Adj-Prep-Noun", "Noun-Noun-Adj" and "Noun-Prep-Noun-Adj".

Contrary to the work of Bouamor et al. (2012), we consider a MWE in the target language as a translation of a MWE in the source language only if the morpho-syntactic pattern of the source MWE has an equivalent morpho-syntactic pattern in the target language. This led us to take a decision on the set of MWEs which are most probable to be entries of the bilingual lexicon. Indeed, the objective of using filtering morpho-syntactic is to identify and separate only the strongest possible MWEs from among the list of all possible MWEs candidates. Naturally, this step increases the precision of the alignment but at the same time it decreases the recall.

English Pattern	Equivalent French Pattern
Adj-Noun	Adj-Noun
Adj-Noun	Noun-Adj
Noun-Noun	Noun-Prep-Noun
Noun-Noun	Noun-Noun
Adj-Noun-Noun	Noun-Adj-Prep-Noun
Adj-Noun-Noun	Noun-Noun-Adj
Adj-Noun-Noun	Noun-Prep-Noun-Adj
Noun-Prep-Noun	Noun-Prep-Noun
Noun-Noun-Noun	Noun-Prep-Noun-Det-Noun
Noun-Noun-Noun	Noun-Prep-Noun-Noun
Adj-Adj-Noun	Noun-Adj-Adj
Noun-Noun-Noun-Noun	Noun-Prep-Noun-Noun-Prep-Det-Noun
Adj-Noun-Noun-Noun	Noun-Prep-Det-Noun-Prep-Noun-Adj
Adj-Adj-Noun-Noun	Noun-Noun-Adj-Adj

Table 1: Some English and French filtering morpho-syntactic patterns (Adj refers to an Adjective, Prep to a Preposition, and Det to a Determiner).

5. Experimental Results

The quality of alignment of MWEs and the impact of using MWEs on machine translation have been evaluated, firstly, manually, by comparing the results of our approach with a reference alignment; and secondly automatically by using the results of our MWEs alignment approach to build the translation model of the state-of-the-art statistical machine translation system Moses (Koehn et al., 2007).

5.1 Manual Evaluation

Our hybrid approach for MWEs alignment and the baseline Giza++ (Och and Ney, 2000) have been evaluated using the evaluation metrics defined in (Mihalcea et al., 2003). The

[2] http://catalog.elra.info/product_info.php?products_id=666.

corpus used to evaluate the performance of the English-French MWE aligners is composed of a set of 1992 parallel sentences extracted from Europarl (European Parliament Proceedings). This parallel corpus is composed of 46265 English words and 49332 French words and has been used to build manually the reference alignment by the Yawat tool (Germann, 2008). Alignment with Giza++ was achieved in source–target and target–source directions and the results were merged using the union heuristic.

At first glance, we can see that the combination of the scoring using co-occurrence and the scoring based on the bilingual dictionary with the filtering patterns provides the best performance of our MWEs alignment approach. It clearly appears that keeping only MWEs candidates that have equivalent morpho-syntactic patterns in source and target languages has had a significant impact on the precision of the alignment. This filtering step certainly has improved the precision but the recall has dropped.

MWEs Aligner	Precision	Recall	F-measure
Baseline (Giza++)	0.83	0.37	0.51
Co-occurrence	0.61	0.63	0.61
Co-occurrence + Bilingual dictionary	0.85	0.54	0.66
Co-occurrence + Bilingual dictionary + Filtering patterns	0.95	0.52	0.67

Table 2: Performance of Giza++ and our MWEs alignment approach.

We observed after aligning some sentences that when both sentence structures are similar, our MWEs aligner performs well. The segmentation is word to word or MWE to MWE depending on what is more frequent in the corpus. Moreover, the surjective formulation of the problem allows our approach to detect expressions in two parts. We can see in the following example that both the English words "*role*" and "*play*" are linked to the French word "*rôle*" (Figure 1).

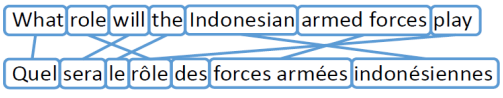


Figure 1: Example of a correct alignment with only the co-occurrence socre.

We have also observed some improvements due the information provided by the bilingual dictionary, as presented in Figure 2. In this example, the bilingual dictionary provides the alignments: "*be/être*", "*decided/décidé*" and "*there/y*". Therefore, our MWEs aligner reconstructs the whole expression "*is to be decided on there/doit y être décidé*". Moreover, the links "*concrete/concret*" and "*programme/programme*" are consolidated by the bilingual dictionary.

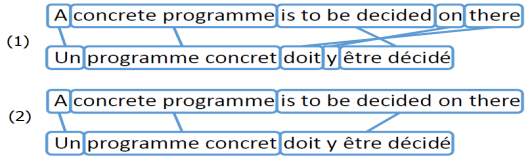


Figure 2: Improvement of alignments (1) Alignment without the bilingual dictionary and (2) Alignment with the bilingual dictionary.

Finally, it should be noted that the link "*is to be decided on there/doit y être décidé*" is abandonned after the step of filtering because no morpho-syntactic pattern matches this expression.

5.2 Evaluation through a Translation Task

The unavailability of a reference alignment of a significant size for MWEs does not allow us to achieve a large scale evaluation. That's why we considered evaluating the impact of MWEs on the quality of translation by integrating the results of our MWEs alignment approach in the training corpus used to extract the translation model of the phrase based statistical machine translation system Moses. We used the factored translation model (Koehn and Hoang, 2007) as our baseline system. It is an extension of the phrase-based model which enables the use of additional linguistic information at the word level such as morphology and Part-Of-Speech. Note that in Moses translation models are produced by the word alignment tool Giza++.

The factored translation model operates on lemmas instead of surface forms. The translation process is then broken up into the following mapping steps:

1. Translate the lemmas of the source language into lemmas in the target language.
2. Generate surface forms given the lemma and linguistic information (Morphology and Part-Of-Speech).

The goal of these experiments is to study in what respect bilingual MWEs are useful to improve the performance of Moses. In Moses, phrase tables are the main knowledge source for the machine translation decoder. The decoder consults these tables to figure out how to translate an input sentence into the target language. These tables are built automatically using Giza++. In order to integrate into Moses the bilingual lexicon which is extracted automatically by our MWEs alignment approach, we add the extracted bilingual lexicon as a parallel corpus and retrain the translation model.

5.2.1 Data and Experimental Setup

In order to study the impact of the bilingual lexicon of MWEs on the performance of Moses, we conducted our experiments on two English-French parallel corpora (Table 3): Europarl (European Parliament Proceedings) and Emea (European Medicines Agency Documents). These corpora were extracted from the open parallel corpus OPUS (Tiedemann, 2012). We achieved three runs and two test experiments for each run: In-Domain and Out-Of-Domain. For this, we randomly extracted 500 parallel sentences rom

Europarl as an In-Domain corpus and 500 pairs of sentences from Emea as Out-Of-Domain corpus. The domain vocabulary is represented in the case of the baseline (Giza++) by the specialized parallel corpus Emea which is added to the training data (Europarl). For our MWEs alignment approach, the domain vocabulary corresponds to the bilingual lexicon of MWEs extracted from the specialized corpus. This bilingual lexicon of MWEs is added to the training corpus (Europarl). It is important to note here that the word alignment tool Giza++ is used to generate the translation tables for both methods (baseline and our approach). In other words, for the baseline (Giza++), the translation table is generated from the parallel corpus which is the concatenation of the general-purpose training data (Europarl) and the domain-specific data (Emea). For our MWEs aligner, the translation table is generated from the parallel corpus which is the concatenation of the general-purpose training data (Europarl) and the bilingual lexicon of MWEs extracted from the domain-specific data (Emea).

Run n°.	Training (# sentences)	Tuning (# sentences)
1	150K+10K (Europarl+Emea)	2K+0.5K (Europarl+Emea)
2	150K+20K (Europarl+Emea)	2K+0.5K (Europarl+Emea)
3	150K+30K (Europarl+Emea)	2K+0.5K (Europarl+Emea)

Table 3: Corpora details used to train Moses language and translation models (K refers to 1000)

5.2.2 Results and Discussion

The performance of the SMT system Moses is evaluated using the BLEU score (Papineni et al., 2002) on the two test sets for the three runs described in the previous section. Note that we consider only one reference per sentence. The obtained results are reported in tables 4 and 5. As shown in tables 4 and 5, for In-Domain texts, Moses achieves a relatively high BLEU score and the scores of Moses when using the results of our MWEs alignment approach are better than those when we use the baseline (Giza++) in all the runs. Again, the best performance for both In-Domain and Out-Of-Domain texts is achieved using the combination of the scoring using co-occurrence and the scoring based on the bilingual dictionary with the filtering morpho-syntactic patterns.

In addition, we explored the use of LSTM (Long Short-Term Memory) recurrent neural network language models (Hochreiter and Schmidhuber, 1997) for rescoring the 100-best translations proposed by the SMT system Moses. This has been limited to only the third run for both In-Domain texts and Out-Of-Domain texts. When we experimented the LSTM to rerank the 100 hypotheses, the BLEU score (corresponding to the combination of the scoring using co-occurrence and the scoring based on the bilingual dictionary with the filtering morpho-syntactic patterns) increases to 35.82 (+1.49 BLEU points) for In-Domain texts and to 25.53 for Out-Of-Domain texts (+0.9 BLEU points).

Run n°.	In-Domain (Europal)			
	Baseline (Giza++)	Co-occurrence	Co-occurrence + Bilingual dictionary	Co-occurrence + Bilingual dictionary + Filtering patterns
1	32.62	32.69	32.71	32.72
2	33.81	33.88	33.89	33.91
3	34.25	34.30	34.32	34.33

Table 4: BLEU scores of Moses for In-Domain texts.

Run n°.	Out-Of-Domain (Emea)			
	Baseline (Giza++)	Co-occurrence	Co-occurrence + Bilingual dictionary	Co-occurrence + Bilingual dictionary + Filtering patterns
1	22.96	23.03	23.06	23.07
2	23.30	23.37	23.39	23.41
3	24.55	24.59	24.62	24.63

Table 5: BLEU scores of Moses for Out-Of-Domain texts.

Because the BLEU score reports only global improvements and does not necessarily reveal the impact of the domain vocabulary (represented by the bilingual lexicon of MWEs extracted with our word alignment approach) on the translation quality of Moses, we manually analyzed some examples of translations drawn from the Out-Of-Domain test corpus (Table 6). We noted after analyzing the translation results of the specialized test corpus (Emea) that in some cases errors come from the training parallel corpus. For instance, the English word "*hypertension*" is sometimes translated as the uniterm "*hypertension*" such as in the bilingual sentence "*Cases of hypertensive crisis have been reported with duloxetine, especially in patients with pre-existing hypertension./Des cas de crise hypertensive ont été rapportés avec la duloxétine, en particulier chez des patients présentant une hypertension préexistante.*", and sometimes translated as the multiterm "*hypertension artérielle*" such as in the bilingual sentence "*The initiation of treatment with XERISTAR is contraindicated in patients with uncontrolled hypertension that could expose patients to a potential risk of hypertensive crisis./L'instauration du traitement par XERISTAR est contre-indiquée chez les patients présentant une hypertension artérielle non équilibrée qui pourrait les exposer à un risque potentiel de crise hypertensive.*". In the example of Table 6, the baseline system provides for the word "*hypertension*" the translation "*hypertension*" and our MWE alignment approach provides for this word the translation "*hypertension artérielle"*. Of course, both translations are correct.

Similarly, both the baseline and our MWE alignment approach translate correctly the multiword expression "*increase in blood pressure/augmentation de la pression artérielle*". On the other hand, as we can see, some translations provided when using the baseline and when we use our approach have many spelling and grammatical errors and are very approximate. As examples, we may mention the translations of the expressions "*has been*

associated/a été associé" and "*in some patients/dans certains patients*". These results can be explained by the fact that, on the one hand, statistical machine translation toolkits like Moses have not been designed with grammatical error correction in mind, and on the other hand, these two expressions have not been considered by our alignment approach as being MWEs. Indeed, even if after the scoring function based on co-occurrence, these two expressions have been identified as MWEs, but the filtering step based on morpho-syntax patterns takes that possibility away (no patterns for these expressions). This is one of the major weaknesses of MWEs alignment approaches based on patterns. Applying morpho-syntax patterns to filter the list of MWEs candidates increases the precision of the alignment but at the same time it decreases the recall.

Example Input (Emea): Duloxetine has been associated with an *increase in blood pressure* and clinically significant *hypertension* in some patients.	
Translation reference	La duloxétine a été associée à une *augmentation de la pression artérielle* et à une *hypertension artérielle* cliniquement significative chez certains patients.
Translation when using the Baseline (Giza++)	Duloxetine a été associé à une *augmentation de la pression artérielle* et de différence cliniquement significative *hypertension* dans certains patients.
Translation when using our MWE aligner (Co-occurrence + Bilingual dictionary + Filtering patterns)	Duloxetine a été associé à une *augmentation de la pression artérielle* et de *hypertension artérielle* cliniquement significative dans certains patients.

Table 6: Translations produced by Moses for a sentence from the Emea corpus.

For the multiword expression "*clinically significant hypertension*", the translation proposed by Moses when using the baseline provides an ungrammatical and meaningless translation (*clinically significant hypertension/différence cliniquement significative hypertension*).

6. Conclusion and Future Work

This paper presented, on the one hand, a hybrid approach to extract and align MWEs from a parallel corpus in a one-step process, and on the other hand, an experimental evaluation of the impact of integrating the results of this MWEs alignment approach on the performance of the statistical machine translation system Moses. We have more specifically shown that adding external knowledge (bilingual lexicons and filtering linguistic patterns) to the co-occurrence scoring function improves significantly the precision of the MWEs alignment approach. We have also showed that the results of the SMT system Moses can be improved by rescoring its n-best translations using a LSTM language model. This study offers several open issues for future work. First, we expect to use machine learning approaches to extend the morpho-syntactic patterns to take into account other forms of MWEs. The second perspective is to explore the integration of bilingual MWEs into other machine translation systems such as neural machine translation ones. We also expect to adapt our MWEs alignment approach to new language pairs such as English-Arabic and French-Arabic.

7. Acknowledgements

This research work is supported by the ASGARD project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 700381.

8. Bibliographical References

Baldwin, T., Villavicencio, A. (2002). Extracting the unextractable: A case study on verb–particles. In Proceedings of the 6th conference on natural language learning (CoNLL).

Barbu, A. M. (2004). Simple linguistic methods for improving a word alignment algorithm. In Proceedings of the 7th International Conference on the Statistical Analysis of Textual Data.

Blank, I. (2000). Terminology extraction from parallel technical texts. *Parallel text processing*, Springer.

Bouamor, D., Semmar, N., and Zweigenbaum, P. (2012). Identifying bilingual Multiword expressions for statistical machine translation. In Proceedings of the eighth international conference on Language Resources and Evaluation (LREC).

Boulaknadel, S., Daille, B., and Aboutajdine, D. (2008). A multi-word term extraction program for arabic language. In Proceedings of the sixth international conference on Language Resources and Evaluation (LREC).

Caseli, H., Ramisch, C., Nunes, M., and Villavicencio, A. (2010). Alignment-based extraction of multiword expressions. *Language Resources & Evaluation*, 44.

Dagan, I. and Church, K. (1994). Termight: Identifying and translating technical terminology. In Proceedings of the fourth conference on applied natural language processing.

Daille, B., Gaussier, E., and Langé, J. M. (1994). Towards automatic extraction of monolingual and bilingual terminology. In Proceedings of the 15th conference on Computational linguistics (ACL).

Deng Y. and Byrne W. (2005). HMM Word and Phrase Alignment for Statistical Machine Translation. In Proceedings of Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP).

DeNero, J. and Klein, D. (2008). The complexity of phrase alignment problems. In Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies (ACL HLT).

Evert, S. and Krenn, B. (2005). Using small random samples for the manual evaluation of statistical association measures. *Computer Speech and Language* 19(4).

Fraser, A. and Marcu, D. (2007). Measuring word alignment quality for statistical machine translation. *Association for Computational Linguistics*, Volume 33, Number 3.

Germann, U. (2008). Yawat: Yet Another Word Alignment Tool. In Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies (ACL HLT).

Hochreiter, S. and Schmidhuber J. (1997). Long short-term memory. *Neural Computation*, Volume 9 Issue 8, MIT Press Cambridge, MA, USA.

Jackendoff, R. (1997). The architecture of the language faculty. *The MIT Press*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: open source toolkit for statistical machine translation. In Proceedings of the 45th annual meeting of the Association for Computational Linguistics (ACL).

Koehn, P. and Hoang, H. 2007. Factored translation models. In Proceedings of the 7th Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning.

Kupiec, J. (1993). An algorithm for finding noun phrase correspondences in bilingual corpora. In Proceedings of the 31st annual meeting on Association for Computational Linguistics (ACL).

Lefever E., Macken L., Hoste V. (2009). Language-independent bilingual terminology extraction from a multilingual parallel corpus. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (ACL).

MacCartney B., Galley M., and Manning C. D. (2008). A Phrase-Based Alignment Model for Natural Language Inference. In Proceedings of the Conference on Empirical Methods in Natural Language Processing.

Marchand, M. and Semmar, N. (2011). A Hybrid Multi-Word Terms Alignment Approach Using Word Co-occurrence with a Bilingual Lexicon. In Proceedings of the fifth language and technology conference: human language technologies as a challenge for computer science and linguistics.

Mihalcea, R. and Pedersen, T. (2003). An evaluation exercise for word alignment. In: Proceedings of the HLT-NAACL 2003 Workshop on Building and using parallel texts: data driven machine translation and beyond - Volume 3.

Moon, R. (1998). Fixed Expressions and Idioms in English. *Oxford: Clarendon Press*.

Och, F. J. and Hermann, N. (2000). Improved statistical alignment models. In Proceedings of the 38th annual meeting on Association for Computational Linguistics (ACL).

Okita, T., Guerra, A. M., Graham Y., and Way, A. (2010). Multi-word expression-sensitive word alignment. In Proceedings of the 4th International Workshop on Cross Lingual Information Access at the 23rd International Conference on Computational Linguistics (COLING).

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics (ACL).

Pearce, D. (2002). A comparative evaluation of collocation extraction techniques. In Proceedings of the third international conference on language resources and evaluation.

Riehemann, S. Z. 2001. A Constructional Approach to Idioms and Word Formation. *Ph.D. Thesis, Stanford University*.

Sag, I., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for NLP. In Proceedings of third CICLing: International Conference on Computational Linguistics and Intelligent Text Processing (CICLing).

Samuelsson Y. and Volk M. (2007). Automatic Phrase Alignment: Using Statistical N-Gram Alignment for Syntactic Phrase Alignment. In Proceedings of the Sixth Workshop on Treebanks and Linguistic Theories.

Semmar N., Servan C., de Chalendar G., and Le NY B. (2010). A Hybrid Word Alignment Approach to Improve Translation Lexicons with Compound Words and Idiomatic Expressions. In Proceedings of the 32nd Translating and the Computer conference.

Semmar, N. and Marchand, M. (2017). Une approche hybride pour la construction de lexiques bilingues d'expressions multi-mots à partir de corpus parallèles. In Proceedings of the 24ème conférence sur le Traitement Automatique des Langues Naturelles (TALN).

Semmar, N. and Laib, M. (2017). Building Multiword Expressions Bilingual Lexicons for Domain Adaptation of an Example-Based Machine Translation System. In Proceedings of the Recent Advances in Natural Language Processing (RANLP).

Seretan, V. and Wehrli, E. (2007). Collocation translation based on sentence alignment and parsing. In Proceedings of the 14ème conférence sur le Traitement Automatique des Langues Naturelles (TALN).

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In Proceedings of the eighth international conference on Language Resources and Evaluation (LREC).

Tufis, D. and Ion, R. (2007). Parallel corpora, alignment technologies and further prospects in multilingual resources and technology infrastructure. In Proceedings of the 4th International Conference on Speech and Dialogue Systems.

Van de Cruys, T. and Villada Moiron, B. (2007). Semantics-based multiword expression extraction. In Proceedings of the workshop on a broader perspective on multiword expressions.

Villavicencio, A., Kordoni, V., Zhang, Y., Idiart, M., and Ramisch, C. (2007). Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In Proceedings of the joint conference on empirical methods in natural language processing and computational natural language learning.

Vintar, S. and Fiser D. (2008). Harvesting multi-word expressions from parallel corpora. In Proceedings of the sixth international conference on Language Resources and Evaluation (LREC).

Wu, C. and Chang, J. S. (2004). Bilingual collocation extraction based on syntactic and statistical analyses. Computational Linguistics and Chinese Language Processing, Vol. 9, No. 1, Association for Computational Linguistics and Chinese Language Processing.

Zhang, Y., Kordoni, V., Villavicencio, A., and Idiart, M. (2006). Automated multiword expression prediction for grammar engineering. In Proceedings of the workshop on multiword expressions: Identifying and exploiting underlying properties.

No more beating about the bush: A Step towards Idiom Handling for Indian Language NLP

Ruchit Agrawal, Vighnesh Chenthil Kumar, Vigneshwaran Muralidharan, Dipti Sharma
FBK Trento, IIIT Hyderabad, Cardiff University, IIIT Hyderabad
ragrawal@fbk.eu, vighnesh.ck@students.iiit.ac.in, muralidaranV@cardiff.ac.uk, dipti@iiit.ac.in

Abstract

One of the major challenges in the field of Natural Language Processing (NLP) is the handling of idioms; seemingly ordinary phrases which could be further conjugated or even spread across the sentence to fit the context. Since idioms are a part of natural language, the ability to tackle them brings us closer to creating efficient NLP tools. This paper presents a multilingual parallel idiom dataset for seven Indian languages in addition to English and demonstrates its usefulness for two NLP applications - Machine Translation and Sentiment Analysis. We observe significant improvement for both the subtasks over baseline models trained without employing the idiom dataset.

Keywords: Idioms, Machine Translation, Sentiment Analysis, Indian Languages

1. Introduction

Idioms pose a problem to most NLP applications (Sag et al., 2002), including sentiment analysis, question answering, machine translation, parsing and so on. One of the most negatively affected subtasks among these is Machine Translation (MT) (Salton et al., 2014a). While parallel corpora can be used by MT systems to learn the language constructs, thereby generating decent translations from source to target language; the same cannot be said for the learning of idioms. Most machine translation systems existent today fail when it comes to the handling of idioms (Table 1). Past research (Salton et al., 2014a) has come up with results stating that a standard Statistical Machine Translation (SMT) system tends to achieve only about half the BLEU score of the same system when applied to sentences containing idioms, as compared to those that do not.

Since idioms encode a very specific kind of linguistic knowledge, it is not easy to learn their automatic handling computationally, without an idiom database. This makes idiom handling a challenging problem for various NLP subtasks including sentiment analysis and question answering in addition to MT. The situation is far worse for Indian languages, a majority of which are low resource languages (Post et al., 2012) with regard to the availability of NLP tools, and yet representing 1.3 billion native speakers. Moreover, these languages are under-studied, while also exhibiting linguistic properties that make idiom handling for various NLP subtasks even more challenging.

In this paper, we present a multilingual parallel dataset that maps 2208 commonly used idioms in English to their translations in seven Indian languages: Hindi, Urdu, Bengali, Tamil, Gujarati, Malayalam and Telugu[1]. The idioms are also annotated with the appropriate sentiments that they channel, and their meanings in the respective languages. We demonstrate the enhancement obtained using our resource for two major applications - machine translation and sentiment analysis. We observe a significant improvement in performance on conducting baseline experiments for the above mentioned tasks.

[1]This dataset is available at goo.gl/receLs

2. Related Work

One of the earliest known work in idiom handling is a comparative study (Volk, 1998) between two contemporary translation systems, namely machine translation and translation memory systems. The study concluded that neither of the systems could handle idioms, and proposed a method of integrating both the systems along with idiom databases to form a phrase archive, which could then be recognised more efficiently by the translation systems. A popular idiom corpus was the one built for Japanese (Hashimoto and Kawahara, 2008). This resource contains Japanese phrases labelled as either idiomatic or literal, which helps to better understand the semantics of the sentence. Another well known work on idiom handling is a system to identify idiomatic expressions from a large bilingual English-Korean corpus, using phrasal alignments to make sense of phrases as well as words, instead of the previously explored word alignment method that purely made sense of words alone (Lee et al., 2010). (Post et al., 2012) crowdsourced a parallel corpus between English and six Indian languages namely: Bengali, Hindi, Malayalam, Tamil, Telugu, and Urdu. They compared the translational capabilities of their model with regard to Google Translate. However, there was no research specific to the domain of idioms in this work. Meanwhile, efforts to efficiently translate idioms resulted in a system that implemented a substitution method (Salton et al., 2014b). This system was tested on a parallel corpus of English and Brazilian-Portuguese, and would first substitute idioms with their literal meanings before translation, and later on substitute these literal meanings back to idioms after. There has been considerable work done on multiword expressions (MWE) in the last two years with regard

Table 1: Performance on Google Translate on idiomatic sentences.

Source	John is known for beating about the bush.
Target	जॉन बुश के बारे में मारने के लिए जाना जाता है।
Transliteration	john bush ke baare mein maarane ke lie jaana jaata hai.
Gloss	John bush GEN about LOC beat-INF PUR know-PASS-MASC-PRES
Meaning	John is known to be hitting in the matter of bush.
Source	The show kept me in stitches the entire time.
Target	शो ने मुझे पूरे समय टाँके में रखा।
Transliteration	sho ne mujhe poore samay taanke mein rakha.
Gloss	Show ERG I-DAT total time stitch LOC keep-PST
Meaning	The show kept me in stiches (injured) the entire time.

Legend: *GEN - Genitive case, LOC - Locative case, DAT - Dative case, PASS - Passive voice, MASC - Masculine gender, PRES - Present tense, PST - Past tense, INF - Infinitive form of a verb, PUR - Purpose of an action*

Category	Number of Idioms
Very negative (- -)	196
Somewhat negative (-)	657
Neutral (0)	726
Somewhat positive (+)	503
Very positive (++)	126
Total	2208

Table 2: Sentiment Annotation Statistics of our Dataset

to Indian languages. A prominent work was the detection of MWEs for Hindi language, mainly noun compounds and noun+verb compounds, using Word Embeddings and WordNet-based features (Patel and Bhattacharyya, 2015). Another important work was the annotation of MWEs for Hindi and Marathi, and classifying them into either compound nouns or light verb constructions (Singh et al., 2016). A very recent work on the topic of idiom handling (Liu and Hwa, 2016) is a system that implements a phrasal substitution by replacing idioms with their corresponding meanings and transforming the meanings to fit the context of the sentence with the right conjugation. So far, there has been no significant work done on creation of a multilingual idiom dataset. To the best of our knowledge, our resource is the first of its kind for Indian languages.

3. Creation of IMIL

3.1. Data Collection

A significant number of idioms in reference materials are ones that are seldom used, thereby hindering their effectiveness (Liu, 2003). We strive to create a multilingual parallel idiom dataset that covers the most commonly used idioms in everyday English, so that it can be used effectively for different NLP applications. We crawled the web through relevant websites to extract over 5000 idioms, their respective meanings, and their sample usages[2]. The list of the websites crawled through is provided here [3]. We then perform an intersection of the list of idioms obtained with those compiled from other well known American English corpora, including the American National Corpus (ANC) (Ide and Suderman, 2004); Michigan Corpus of Academic Spoken English (MICASE) (Simpson et al., 2002), and Brown Corpus (Francis and Kucera, 1979).

The compilation of idioms from the above mentioned corpora was done using the method proposed by (Muzny and Zettlemoyer, 2013). A threshold count of 25 was set for eliminating non-frequent idioms after performing the intersection. The intersection and elimination was done for the removal of inaccurate programmatic detections but at the same time ensuring that it is a frequently occurring idiom. We were then able to filter out and consolidate a list of 2208 most commonly used idioms, thus rendering the application of the corpus as close to natural human language as possible.

3.2. Annotation Guidelines

We create a parallel idiom dataset for seven Indian languages in addition to English. The English idioms extracted in the first phase (Section 3.1.) are translated to the following languages :Hindi, Urdu, Malayalam, Bengali, Gujarati, Tamil and Telugu. The annotation for each language is performed by three native speakers and later verified by two professional linguists to deal with language specific idiosyncrasies. The resulting dataset is called "Idiom Mapping for Indian Languages" (IMIL).

Following are the guidelines that the annotators were asked to follow:

1. An idiom, its meaning, and a sample usage is provided in English followed by slots for the seven languages. If there is an equivalent idiom in the target language, then the corresponding idiomatic translation is to be provided. In case this is not possible, a phrasal translation that aptly conveys the meaning of the source idiom is to be added instead. This information is to be mentioned along with the translation, using the tags 'P' (phrasal) or 'I' (idiomatic).

[2]This was done using various python libraries, primarily BeautifulSoup4.

[3]https://goo.gl/s4R4uH

2. In case it is neither possible to find an appropriate idiom nor an equivalent phrasal translation, "Skip" has to be entered in the target slot.

3. The sentiment of each idiom should be marked at each node in its parsed tree structure (detailed in Section 4.2.). The annotation scheme along with the statistics for each sentiment is given in Table 2.

4. Experiments and Results

We demonstrate the application of our dataset by conducting experiments for two different NLP tasks:

4.1. Machine Translation

Statistical Machine Translation (SMT) (Koehn, 2009) and Neural Machine Translation (NMT) ((Sutskever et al., 2014), (Cho et al., 2014), (Bahdanau et al., 2014)) are the two major MT paradigms today which require large parallel corpora for training. Such corpora containing sufficient idiomatic sentences are not available for Indian languages. We employ $IMIL$ and conduct experiments to analyse MT quality when the system is fed with an idiom mapping in addition to the parallel training corpora.

We employ the multilingual Indian Language Corpora Initiative (ILCI) corpus (Jha, 2010) for training [4]. It contains 50,000 sentences from the health and tourism domains aligned across eleven Indian languages. We choose three Indo-Aryan languages (Hindi, Bengali, Urdu) and one Dravidian language (Telugu) as candidate languages for our experiments to maintain brevity. We employ preprocessing to eliminate misalignments - the resultant dataset has a size of 47,382 sentences (Training - 44000, Validation - 1382, Test - 2000). We create 250 manually annotated sentences with idiomatic usages, of which 50 are appended to the validation set, and 200 to the test set. The resultant size of the training set, validation set and test set ($Test_{Concat}$) is 46,200, 1432 and 2200 sentences respectively. We conduct experiments using both NMT as well as SMT approaches. For the former, we concatenate $IMIL$ to the $ILCI$ training set, yielding a training set containing 46200 sentences. We train an NMT model on this set using the architecture proposed by (Bahdanau et al., 2014) and call it NMT_{IMIL}. We compare the performance of this model with a baseline NMT model trained on the $ILCI$ train set (44000 sentences). This model is called NMT_{Base}. The results obtained are given in Table 3. Although NMT_{IMIL} produces better output than NMT_{Base} in terms of BLEU score, the translation quality obtained is found to be substandard on manual inspection due to inadequate inflectional learning. It is, however, much better as compared to the literal translation produced by NMT_{Base} for idioms.

Additionally, we train a Phrase Based Statistical Machine Translation system (PBSMT) (Zens et al., 2002) using our dataset as an additional resource for the phrase table generation. We use Moses (Koehn et al., 2007) for phrase extraction as well as lexicalized reordering as proposed by (Kunchukuttan et al., 2014). We append the 2208 idioms to the phrase-tables rather than concatenating them to the training set[5]. The training set is thus 44000 sentences with the other splits remaining the same as mentioned above. We also add an additional feature in the phrase table to indicate whether the idiom can have a non-idiomatic usage as well, i.e. 0 if it cannot and 1 if it can[6]. We compare the performance with a standard PBSMT model trained on the $ILCI$ parallel corpus, called $PBSMT_{Base}$. We observe significantly higher improvement in scores (an average inrease of 2.69 % BLEU) using this method than that obtained using NMT (0.73 % BLEU). This can be attributed to a more sophisticated handling of idioms using phrase tables rather than direct concatenation to the training corpus.

4.1.1. Discussion

Although the performance of the translation system improves with the inclusion of our idiom dataset (IMIL), there are a few issues that we noted in the idiomatic translations produced by the system. An idiom is not a fixed multi-word expression but allows considerable variation in how the idiomatic expression is going to be realized in a sentence based on syntactic and morphological properties of (a) the tokens inside the idiom (b) the composite expression itself. If the equivalent idiom in the target language belongs to a similar syntactic category the translation is likely to be correct. Where the syntactic categories of the expressions differ, the translation quality is affected.

Let us take two idioms for illustration: 'without batting an eyelid' and 'cannot stomach someone or something'. The syntactic category of the two idioms are PP and NP respectively which determines how they are used in a sentence. The first idiom can be used as a part of a Verb Phrase like 'VP(VP (uttering a lie) PP(without batting an eyelid)). If the equivalent idiom in target language can be used as a part of a verb phrase just like English and hence the translation sounds good. For example, when the system output for Telugu translation is "(VP (VP(saṅkōcapaḍakuṇḍā) VP(abad'dhaṁ annāḍu))", the translation is perfectly okay. Even though the target phrase "(VP saṅkōcapaḍakuṇḍā)" is not a PP like in English, it can constitute a larger verb phrase just like in English. In the second idiom, the target language phrase learnt from our parallel dataset is not a verb phrase but a noun phrase 'bardaasht ke baahar'. Hence a source sentence 'He could not stomach the truth' when translated as 'wah sach bardaasht ke baahar hai' is not a good translation because of this syntactic incompatibility of the target

[4]This corpus is available on request from TDIL: https://goo.gl/VHYST

[5]We use the xml markup feature provided by Moses for suggesting phrasal translations to the decoder.

[6]This facilitates learning of the decoder for idioms having possible literal usage as well.

Table 3: Impact of IMIL on Sentiment Analysis

S1	John is always beating about the bush
Base	(2 (2 John) (3 (3 (2 (2 is) (2 always)) (2 (2 beating) (2 (2 about) (2 (2 the) (2 bush))))) (2 .)))
IMIL	(2 (2 John) (1 (1 (2 (2 is) (2 always)) (1 (2 beating) (2 (2 about) (2 (2 the) (2 bush))))) (2 .)))
S2	He hit the ceiling when he came to know the truth.
Base	(3 (2 He) (2 (2 (2 (3 hit) (2 (2 the) (2 ceiling))) (2 (2 when) (2 (2 he) (2 (2 came) (2 (2 to) (2 (2 know) (2 (2 the) (3 truth))
IMIL	(1 (2 He) (1 (1 (0 (3 hit) (2 (2 the) (2 ceiling))) (2 (2 when) (2 (2 he) (2 (2 came) (2 (2 to) (2 (2 know) (2 (2 the) (3 truth))
S3	Mary is in the pink of health.
Base	(2 (2 (2 Mary) (2 (2 is) (2 (2 in) (2 (2 (2 the) (2 pink)) (2 (2 of) (2 health))))) (2 .)))
IMIL	(3 (2 (2 Mary) (3 (2 is) (3 (2 in) (4 (2 (2 the) (2 pink)) (2 (2 of) (2 health))))) (2 .)))

Legend: *S1, S2, S3 : Sample Sentences, 0: Extremely negative, 1: Negative, 2: Neutral, 3: Positive, 4: Extremely positive, Base:* $Stanford_{Base}$*, IMIL:* $Stanford_{IMIL}$

Model	Direction	Bengali	Urdu	Telugu
$PBSMT_{Base}$	hin=>	28.42	41.38	14.82
	hin<=	28.17	42.64	19.47
NMT_{Base}	hin=>	25.73	39.57	11.43
	hin<=	26.42	43.51	15.14
$PBSMT_{IMIL}$	hin=>	**31.84**	**44.74**	**16.46**
	hin<=	**32.06**	43.97	**21.94**
NMT_{IMIL}	hin=>	25.91	40.18	12.18
	hin<=	27.81	**44.15**	15.97

Table 4: Results obtained on $Test_{concat}$ by our models in terms of BLEU score. *hin: Hindi*

	Compositional	Actual
break a leg	-	+
kick the bucket	0	-
apple of my eye	0	++
under the weather	0	-

Table 5: Examples of non-compositionality of sentiments in idioms

idiom. Idioms have to be matched for their syntactic compatibility while translating them. Secondly, there are components in idioms which are determined by the other tokens outside the idiom. e.g. 'worth one's salt' is realized as 'worth his salt', 'worth her salt' and so on agreeing with the subject. These changes should be accommodated in the target language as well.

As part of future work, the automatic generation of the bidirectional lexical and phrasal translation probabilities as proposed by (Klementiev et al., 2012) can be explored along with the feature addition in the phrase table for further improvement in performance for languages where large monolingual corpora are available. This could facilitate the coverage of words and phrases surrounding the idiom by the the decoder in addition to the idiom itself.

4.2. Sentiment Analysis

This is one of the most interesting applications of the database due the non-compositional behavior of idioms in terms of semantic as well as sentiment information. Table 4 gives some of such examples, motivating the need for a sentiment-annotated idiom database.

Model	CALA	CARLA
$Stanford_{Base}$	67.01	70.23
$Stanford_{IMIL}$	68.73	73.56

Table 6: Sentiment Analysis results on $Test_{Concat}$. *CALA: Combined Approximate Label Accuracy. CARLA: Combined Approximate Root Label Accuracy.*

This is the primary motivating factor for the need of an idiom sentiment database like $IMIL$, which can help towards better Sentiment Analysis, especially the phrase-level approaches ((Wilson et al., 2005), (Socher et al., 2013)). $IMIL$ can be employed for Sentiment Analysis for any candidate language among the languages in consideration. Due to space constraints, we demonstrate the application of $IMIL$ to Sentiment Analysis for English, using Recursive Neural Tensor Networks (RNTNs) proposed by (Socher et al., 2013). The RNTNs can learn the phrase sentiments from a sentiment treebank containing trees with a sentiment annotated at each node in the parsed tree structure of a sentence. Our dataset is seamlessly integratable with the Stanford Sentiment Treebank, since the sentiment annotation scheme (mentioned in Section 3.2.) is in alignment with the method employed by (Socher et al., 2013).

We generate the parse trees for the 2208 idioms from $IMIL$ and annotate them with sentiments at each node level. We append this treebank to the training set employed by (Socher et al., 2013). The model trained on this set is called $Stanford_{IMIL}$. For development and testing, we append 50 and 200 annotated sentence trees with idiomatic usage to the original validation and test sets respectively. The resultant statistics are as follows: Training set - 10744 trees, validation set - 1151 trees, test set - 2410 trees. The test set is called $Test_{Concat}$. We compare the performance with a baseline model trained on the original training set employed by (Socher et al., 2013). We call this model $Stanford_{Base}$. The results of both the models on $Test_{Concat}$ are given in Table 5. $Stanford_{IMIL}$ shows significant improvement over $Stanford_{Base}$. We observe that although the training is done on only idiomatic phrase trees than sentence

trees, $Stanford_{IMIL}$ produces a 1.52 % increase for Combined Approximate Label Accuracy and a 3.33 % increase for Combined Approximate Root Label Accuracy. This is attributed to the ability of the RNTN to learn the sentiment at higher nodes of the tree from the subtrees using the tensor-based composition function.

4.2.1. Discussion

We inspect the outputs generated by the model after it is trained using our parallel dataset IMIL[7]. We plot the sentiment trees using the outputs generated by StanfordBase and StanfordIMIL. Table 3 shows the performance of the model for idiomatic sentences, before and after training with IMIL. [8]. It can be observed that the model is able to handle the non-compositional behavior of idioms with respect to sentiments on being trained with IMIL as additional data. This would be very challenging to accomplish in the absence of labelled sentiment trees for idioms. StanfordIMIL is also able to learn the correct sentiment trees for the entire idiomatic sentences, although the training is done only on the idiom phrases.

5. Conclusion

This paper is an effort in the direction of idiom handling for various Natural Language Processing tasks, with an emphasis on Indian languages. We present $IMIL$, a multilingual parallel idiom dataset consisting of 2208 idioms, spanning across seven languages in addition to English. We demonstrate its usefulness for two applications, namely Machine Translation and Sentiment Analysis. We conclude that Phrase-based SMT is better able to handle idiomatic sentences than Neural Machine Translation, producing an average increase of 2.69 % BLEU score over a baseline model trained over the same corpus. A promising improvement is also observed for Sentiment Analysis, primarily due to the inability of the baseline model to learn the non-compositional sentiments of idioms, which is addressed with the presence of an idiom sentiment dataset. We conclude that $IMIL$ is a valuable resource with potential applications in varied NLP subtasks, especially with regard to Indian languages.

6. References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bhattacharyya, P. (2017). Indowordnet. In *The WordNet in Indian Languages*, pages 1–18. Springer.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Fellbaum, C. (1998). Towards a representation of idioms in wordnet. In *Proceedings of the workshop on the use of WordNet in Natural Language Processing Systems (Coling-ACL 1998)*.

Francis, W. N. and Kucera, H. (1979). Brown corpus manual. *Brown University*, 2.

Hashimoto, C. and Kawahara, D. (2008). Construction of an idiom corpus and its application to idiom identification based on wsd incorporating idiom-specific features. In *Proceedings of the conference on empirical methods in natural language processing*, pages 992–1001. Association for Computational Linguistics.

Ide, N. and Suderman, K. (2004). The american national corpus first release. In *LREC*.

Jha, G. N. (2010). The tdil program and the indian langauge corpora intitiative (ilci). In *LREC*.

Klementiev, A., Irvine, A., Callison-Burch, C., and Yarowsky, D. (2012). Toward statistical machine translation without parallel corpora. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 130–140. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Koehn, P. (2009). *Statistical machine translation*. Cambridge University Press.

Kunchukuttan, A., Mishra, A., Chatterjee, R., Shah, R., and Bhattacharyya, P. (2014). Sata-anuvadak: Tackling multiway translation of indian languages. *pan*, 841(54,570):4–135.

Lee, H.-G., Kim, M.-J., Hong, G., Kim, S.-B., Hwang, Y.-S., and Rim, H.-C. (2010). Identifying idiomatic expressions using phrase alignments in bilingual parallel corpus. In *PRICAI*, pages 123–133. Springer.

Liu, C. and Hwa, R. (2016). Phrasal substitution of idiomatic expressions. In *HLT-NAACL*, pages 363–373.

Liu, D. (2003). The most frequently used spoken american english idioms: A corpus analysis and its implications. *Tesol Quarterly*, 37(4):671–700.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Muzny, G. and Zettlemoyer, L. (2013). Automatic idiom identification in wiktionary. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1417–1421.

Osherson, A. and Fellbaum, C. (2010). The representation of idioms in wordnet. In *Principles, Construction and Application of Multilingual Wordnets. Proceedings of the Fifth Global WordNet Conference (GWC 2010), Mumbai, India. Narosa Publishing House*.

[7]We choose English due to smooth integrability with Stanford SA

[8]A graphical view is provided at https://goo.gl/5Wcqba

Patel, D. S. S. B. K. and Bhattacharyya, P. (2015). Detection of multiword expressions for hindi language using word embeddings and wordnet-based features. In *12th International Conference on Natural Language Processing*, page 291.

Post, M., Callison-Burch, C., and Osborne, M. (2012). Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 401–409. Association for Computational Linguistics.

Sag, I., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. *Computational Linguistics and Intelligent Text Processing*, pages 189–206.

Salton, G., Ross, R., and Kelleher, J. (2014a). An empirical study of the impact of idioms on phrase based statistical machine translation of english to brazilian-portuguese.

Salton, G., Ross, R., and Kelleher, J. (2014b). Evaluation of a substitution method for idiom transformation in statistical machine translation.

Simpson, R. C., Briggs, S. L., Ovens, J., and Swales, J. M. (2002). The michigan corpus of academic spoken english. *Ann Arbor, MI: The Regents of the University of Michigan*.

Singh, D., Bhingardive, S., and Bhattacharya, P. (2016). Multiword expressions dataset for indian languages. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Volk, M. (1998). The automatic translation of idioms. machine translation vs. translation memory systems. *Machine Translation: Theory, Applications, and Evaluation, An Assessment of the State-of-the-art, St. Augustin, Gardez Verlag*.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Zens, R., Och, F. J., Ney, H., and Vi, L. (2002). Phrase-based statistical machine translation. *KI*, 2479:18–32.

Sentence Level Temporality Detection using an Implicit Time-sensed Resource

Sabyasachi Kamila, Asif Ekbal, Pushpak Bhattacharyya
Department of Computer Science and Engineering, IIT Patna
Patna, India
{sabysachi.pcs16,asif,pb}@iitp.ac.in

Abstract

Temporal sense detection of any word is an important aspect for detecting temporality at the sentence level. In this paper, at first, we build a temporal resource based on a semi-supervised learning approach where each Hindi-WordNet synset is classified into one of the five classes, namely *past*, *present*, *future*, *neutral* and *atemporal*. This resource is then utilized for tagging the sentences with *past*, *present* and *future* temporal senses. For the sentence-level tagging, we use a rule-based as well as a machine learning-based approach. We provide detailed analysis along with necessary resources.

Keywords: Temporal Sense Detection, Semi-supervised Machine Learning, Sentence Level Temporality Detection

1. Introduction

Over the last few years, 'temporality' has drawn a significant attention to the community of Natural Language Processing (NLP) and Information Retrieval (IR). Time is an intrinsic property that aids in ordering events in a sequential order from the past to present to future. This ordering of events is very crucial in analyzing a document. Some of the applications where temporality plays an important role include automatic summaries (Allan et al., 2001), question-answering (Schockaert et al., 2006), clustering (Alonso et al., 2009), similarity of documents (Jatowt et al., 2013) etc. Queries can specify temporality both explicitly and implicitly. Queries like *"World Cup 2011"*, *"Indian Prime Minister 2000"* etc. denote the temporality explicitly, whereas the queries like *"Chomsky's childhood"*, *"Recent Bollywood songs"* etc. correspond to implicit temporality. All these highlight the significance of time in refining and ranking the results retrieved from a search engine.

In a survey, Joho et al. (2013) claimed that most of the time user queries need to be addressed with *recent* information. However, many situations demand the *past* or *future* related information. For example, the query *"*ग्लोबल वार्मिंग की वर्तमान स्थिति *(globala vArmiMga kI vartamAna sthiti - The current status of global warming.)"*[1] requires *present* related information whereas the queries like *"*डिजिटल अर्थव्यवस्था में भारत के लिए अवसर क्या है? *(DijiTala arthavyavasthA meM bhArata ke lie avasara kyA hai?- What are the opportunities for India in the Digital Economy?)"*, *"*अशोक का इतिहास *(ashoka kA itihAsa- History of Ashoka)"* need *future* and *past* related information, respectively. Here, tense related information does not help but the implicit temporal keywords *'current'*, *'opportunities'* and *'history'* help in finding the temporal information of the respective queries.

1.1. Motivation and Problem Definition

Most of the earlier studies, for example, TempEval tasks (Verhagen et al., 2009; Verhagen et al., 2010; UzZaman et al., 2013) in the computational linguistics, have concentrated on identifying the temporal expressions, event expressions and various relations among these. These studies tried to address the temporal aspects of information with the help of linguistic constructs such as the presence of temporal expressions like before, now, after etc., document creation time (DCT), or explicit time expressions.

Let us consider the following two example sentences: **Sentence-I**: *You should live in the present.* **Sentence-II**: *She gave him a nice present.* When these two sentences are subjected as input to the SUTime tagger,[2] we observe that, for both the sentences, the word 'present'[3] is tagged as a temporal expression. However, it should be temporal only for the first sentence. When these two sentences are subjected as input to the HeidelTime tagger,[4] no temporal mention is found in either of the sentences.

In Hindi-WordNet (Bhattacharyya, 2010), the word *"*कल *(kala)"* has 8 senses in total, both *temporal* and *atemporal*. Let us consider the following three example sentences: **Sentence-I**: *"*यह लेख कल के अखबार में है *(yaha lekha kala ke akhabAra meM hai-This article is in yesterday's newspaper.)"*; **Sentence-II**: *"*कल कौनसी परीक्षा है? *(kala kaunasI parIkShA hai?- Which examination is scheduled tomorrow?)"*; **Sentence-III**: *"*नये-नये कल का निर्माण हो रहा है *(naye-naye kala kA nirmANa ho rahA hai-New machines are being built)"*. Here, the same word *"*कल *(kala)"* corresponds to *'yesterday'*, *'tomorrow'* and *'machines'* in the first, second and third sentence, respectively. These denote the *past* and *future* senses in the first and second sentence, respectively, and *atemporal* in the third sentence. Unless the contextual information is taken into account these cannot be disambiguated appropriately.

It is evident from the existing literature that there is a lack of attention in detecting the implicit temporal sense of words. In order to capture such implicit temporal senses, we propose an effective technique for determining the temporal sense of each synset of the

[1] Henceforth, all the Hindi examples are represented by Hindi glosses, ITRANS representations and using equivalent English translation.

[2] `http://nlp.stanford.edu/software/sutime.html`

[3] The word 'present' has noun Part-of-Speech (PoS) tag in both the sentences.

[4] `http://heideltime.ifi.uni-heidelberg.de/heideltime/`

Hindi-WordNet. We augment each synset of the Hindi-WordNet with one of the five temporal tags, namely *past*, *present*, *future*, *neutral*, and *atemporal*. For example, the synset *"*प्राचीन *(prAchIna-Ancient)"*, *"*मौजूदा *(maujUdA-Existing)"* and *"*आगामी *(AgAmI-Forthcoming)"* correspond to *past*, *present* and *future* time sense, respectively. The synset *"*अयोग्य *(ayogya-Unworthy)"* is characterized as *'atemporal'* as it does not depict any time sense. There are also some synsets, such as *"*सुबह *(subaha-Morning)"* that clearly represent a time sense, but cannot be specifically categorized to *past*, *present* or *future*. Such kind of instances are denoted as *neutral*.

At first, we propose a semi-supervised machine learning framework for detecting temporal word senses. The process initiates learning with a set of seed instances for each class, and then iteratively expands it following various expansion strategies. The temporal resource, Tempo-Hindi-WordNet that we build will definitely be an effective resource for the efficient temporal information access in the resource-poor languages like Hindi which is one of the widely spoken languages worldwide and one of the official languages in India. We show how this resource can be utilized for sentence-level temporal tagging.

Our present study is inspired from the prior works (Dias et al., 2014; Hasanuzzaman et al., 2014), where the authors attempted to annotate each synset of English WordNet with four temporal dimensions, namely *past*, *present*, *future* and *atemporal*. Our work differs from these existing works in terms of the following points: (i). present work attempts to build a temporal resource that can facilitate temporal information access in Hindi; (ii). new expansion strategies including word-embedding based techniques are proposed; and (iii). two approaches (i.e. rule-based and machine learning-based) for sentence-level temporality detection are developed. The present work also differs from an earlier work reported in (Pawar et al., 2016) in terms of expansion strategies, quality of temporal resource created, and application of the resource developed for sentence-level temporality detection in two different domains, *viz.* newswire and Twitter.

2. Word-level Temporal Sense Detection

Due to the unavailability of annotated dataset, we adapt a semi-supervised learning strategy for temporal word sense detection.

2.1. Seed Data Creation

We manually prepare a seed set based on the synsets of Hindi-WordNet. Three individuals (with post-graduate level knowledge) were asked to annotate the seed set based on the word knowledge and the information available in the gloss, and it was found to have a substantial multi-rater kappa agreement (Fleiss, 1971) of 0.73 among the annotators. The tag was finalized based on majority voting. The seed consists of 96 synsets, out of which 48 are *atemporal* and the rest are equally distributed among the *past*, *present*, *future* and *neutral*. While creating this, special care was taken to ensure that it is not biased towards any specific temporal class or Part-of-Speech (PoS) category. It is to be noted that in the Hindi-WordNet *samay*-time is biased towards the 'noun' PoS category.

2.2. Gold Standard Set for Evaluation

In order to evaluate the Tempo-Hindi-WordNet, we manually prepare a gold standard test set with the synsets taken from the Hindi-WordNet.[5] Same persons who created the seed set were employed for this annotation with the help of similar kind of information. Multi-rater kappa agreement was found to be 0.63 which gives an idea about the level of difficulty involved, as humans are also not in agreement for a number of decisions. One of the considerations was the fundamental fact that the core concepts of words do not exist in many cases; these are rather defined by the contextual information. For example, the synset *"*आवश्यकता (AvashyakatA-requirement) - आवश्यक होने की अवस्था या भाव (Avashyaka hone kI avasthA yA bhAva-State of necessity) has a connotative sense of *future*. However, from the inspection to the WordNet gloss, it was found not to have any time sense. For a second example, the synset *"*इमरजेंसी (imarajeMsI - Emergency)- संकट या विपत्ति का समय (saMkaTa yA vipatti kA samaya-Time of crisis or disaster)" describes a situation or condition of emergency where we can call something *"emergency"* by looking at its effects in the recent *past* or *present*. We cannot surely confirm a situation to be emergent that has not yet happened. Hence, it can have both the *past* and *present* time senses.

Many idiomatic synsets such as *"*धूप-छाँह *(dhUpa-ChA.Nha)*- बारी-बारी से आने वाला अच्छा और बुरा समय *(bArI-bArI se Ane vAlA achChA aura burA samaya - ups and downs of life where good and bad times come alternately)"* etc. are very difficult to annotate properly. From the meaning represented in the gloss we can conclude that the synset signifies a time period (denoting neutral) or a state of life (denoting atemporal). However, majority agree it to be of atemporal type. Finally, instances of gold standard are annotated based on the majority agreement. The gold standard set finally contains 180 instances: 16 past, 8 present, 13 future, 22 neutral and 121 atemporal.

2.3. Framework

We propose a hierarchical classification framework for solving the problem. In the first level, we distinguish temporal vs. atemporal. In the second level, we classify temporal instances into *past*, *present*, *future* and *neutral* categories. Initial set of seed set is iteratively expanded using various expansion strategies. Steps of the algorithm are shown in Algorithm 1.

2.4. Expansion Strategies

We propose two expansion strategies: *"Confidence based Expansion (CBE)"* and *"Semantic Distance based Expansion (SDE)"*.

1. In the first model (i.e. CBE), we select the most informative instances based on the prediction confidence of the classifier.

[5] While creating models, we excluded these instances from the Hindi-WordNet for processing.

Algorithm 1 Basic steps of temporal resource creation

1: Select initial set of seed words.
2: **repeat**
3: Train the model on the training instances created from the seed set.
4: Evaluate the model developed on the rest of the synsets of Hindi-WordNet (in an incremental manner).
5: Expand the seed set according to the chosen expansion strategy.
6: **until** cross-validation accuracy drops.
7: Classify the Hindi-WordNet using the final trained model.

2. In our second model (i.e. SDE), we select the most informative instances based on the semantic distance rather than the classifier's confidence.

Prototype Vector Generation: For representing the instances, we create a 'prototype vector' from the glosses of synset, synonyms, hyponyms and hypernyms. We assume that the temporal senses of a synset are propagated through various semantic relations of the Hindi-WordNet, and hence the information represented in their glosses will provide an important evidence. Vectors of two semantically related synsets could assist in spreading or detecting connotative temporality. Such a vector representation should be able to quickly refine the classifier's decision boundaries.

For example, semantic relations such as hypernym and hyponym detect connotative temporality as hypernym is a generalization and hyponym is a specialization of the synset. For example, *"*विराम_काल *(virAma kAla-rest period)"* is the kind of *"*काल *(kAla-period)"*. Here, *"*काल *(kAla-period)"* is the hypernym and *"*विराम_काल *(virAma kAla- rest period)"* is a hyponym. Both of these indicate temporality. As we encode both hyponyms and hypernyms, one's presence ensures other's inclusion through expansion.

We use Word2vec tool (Mikolov et al., 2013) for generating word embedding vectors. The model is trained on Bojar's corpus (Bojar et al., 2014) of around 44 million Hindi sentences for the training of Word2Vec using Skip-gram model with the dimension set to 200 and window size set to 7. For each content word of the synset, hyponyms, hypernyms and their glosses, we extract the corresponding vector of 200 dimension. All these vectors are averaged over to create a 'prototype vector'. If there are m content words then the prototype vector is generated as shown in the Equation 1.

$$\frac{\sum_{i=1}^{m} WE(w_i)}{m} \quad (1)$$

where, m is the number of content words in the glosses of synset, synonyms, hypernyms and hyponyms; WE(w_i) is the word embedding vector of the i^{th} token.

2.4.1. Confidence based expansion (CBE)

This expansion strategy makes use of the classifier's confidence as a mean to expand the initial seed list. Higher the value of confidence, more is the chance of its belongingness in the expanded list. We use three classification algorithms, namely Support Vector Machine (SVM) (Joachims, 2002), Naive Bayes (NB) (John and Langley, 1995) and Decision Tree (DT) (Quinlan, 1993).

Each classifier is trained with the feature vectors generated from the initial seed instances and tested with the rest of the Hindi-WordNet synsets. All the training and test instances are represented by *'prototype vectors'*. Instances which are predicted with higher confidence are considered to be the useful samples. Such instances are given more priority during expansion so as to preserve the connotational properties of the initial seed entities intact. In every iteration, we add the instances to the training set in such a way that the ratio of the instances of different classes are maintained at par with the initial class distribution. For expansion, we execute the following steps: (i). for each class, we select the top-10 instances from the test set based on the classifier's confidence; (ii). exclude such instances from the test set for evaluation in the next iteration; (iii). add the selected instances to the initial training set; and (iv). perform 10-fold cross validation experiment on the expanded training set. This process is repeated in an iterative fashion. If the cross-validation accuracy does not increase in the consecutive 3 iterations, then we terminate the process. Finally, we select the model that shows the best performance during all the iterations. This final selected model is used for classifying the entire Hindi-WordNet.

2.4.2. Semantic distance based expansion (SDE)

In this method, we expand the seed set in such a way that the newly added instances are always semantically closer to the existing seed instances. Unlike CBE, this method does not depend on the classifiers' decisions, rather it relies on the semantic distance between the two vectors. All the training and test instances are represented by *'prototype vectors'*. For each test instance, we measure its distance from all the training vectors by computing the cosine similarities. We choose the new candidate instances to be added to the initial training set in such a way that: (i). added instances are closer to the existing seed entities; and (ii). rejected instances seem to be dissimilar to the existing ones.

We select 10 most similar instances (based on cosine similarity) for each class from the test set, and add them to the training set in each iteration. We exclude such instances from the test set in the next iteration. We stop iterating when the cross-validation accuracy does not increase in the consecutive three iterations.Like the CBE based model, we finally fix the model that produces the highest cross-validation accuracy.

Through this process we ensure that in every run, a good quality of new instances are added to the existing training set. As the semantically closer instances are added during the expansion process, we believe that it preserves the soundness property. The process is more effective in detecting connotative temporal properties of the data as we expand our knowledge base by inducing word-embedding vectors and other WordNet semantic relations.

2.5. Results and Analysis

In this section we report the experimental results along necessary analysis.

2.5.1. Experiments: Cross-validation

Results of 10-fold cross-validation are reported in Table 1 for the hierarchical SDE based classification approach. It is to be noted that this model quickly converges but still attains better accuracy. It shows precision, recall and F-measure values of 85.53%, 89.66% and 87.55%, respectively for SVM. The SVM classifier performs better than DT and NB. Tempo-Hindi-WordNet that we obtain at the end contains 1,572 past, 3,650 present, 2,822 future, 5,429 neutral and 130,413 atemporal instances.

Iteration		1	2	...	9	10
DT	precision	61.80	70.56		78.89	78.85
	recall	65.89	72.45		78.91	78.81
	F-measure	63.78	71.49		78.90	78.83
NB	Iteration	1	2	.....	9	10
	precision	62.23	67.28		73.78	74.20
	recall	64.88	68.03		76.92	74.10
	F-measure	63.53	67.65		75.32	74.15
SVM	Iteration	1	2	....	12	13
	precision	72.87	73.76		84.67	85.53
	recall	62.56	63.78		89.23	89.66
	F-measure	67.32	68.40		86.89	87.55

Table 1: Iteration-wise 10-fold cross-validation results. Here, we report the average performance of all the classes for the SDE based approach.

For CBE based expansion technique, experiments on 10-fold cross-validation yield precision, recall and F-measure values of 86.88%, 85.32% and 86.14%, respectively. In both the cases, SVM performs better-may be due to its robustness in efficiently handling high dimensional feature space. Finally, we observe the following statistics of the Tempo-Hindi-WordNet using CBE: 973 past, 431 present, 4,302 future, 1,977 neutral and 135,183 atemporal instances.

2.5.2. Experiments: Gold Standard

We evaluate our various models on the gold standard test set. We also show the evaluation on easy-to-classify instances.[6] The basis of showing this evaluation is to demonstrate how our classifier performs with respect to the humans. We report the results using all the expansion strategies in Table 2 for the first level classes in the hierarchy (i.e. temporal vs. atemporal). The results corresponding to the "Gold set" column denotes the overall performance (easy+hard instances). It shows that the SDE based approach performs better. The state-of-the-art system corresponds to the resource created in (Pawar et al., 2016) that was based on classifier's confidence and made use of only gloss-based features. The results reported for state-of-the-art system are for gold standard set.

Evaluation results of the second level (i.e. finer level) classification in the hierarchy are shown in Table 3. This again shows that the semantic distance-based instance selection strategy is more effective compared to the classifiers' confidence based selection strategy.

As evident from the experimental results, in both coarse (i.e. first level) as well fine-grained (i.e. second level) classification scenarios, word embedding and semantic relation based techniques improve the efficiency at a greater extent. Recall improves at a much faster rate, indicating the efficiency of word embedding features in correctly retrieving more and more instances. The gain in overall performance signifies the fact that the classifier is not only robust in handling easy-to-classify instances, but also generalizes well at predicting hard-to-classify instances. Although in CBE based method there is a drop in precision compared to the state-of-the-art system, our proposed model shows considerably higher F-measure due to the significant gain in recall. This phenomenon ensures that our current model is able to find a good trade-off between easy and hard cases.

[6]Correspond to the examples annotated by all humans with 100% agreement.

Hierarchical First Level	**CBE**		**SDE**	
	Gold set	Easy cases	Gold set	Easy cases
Precision	80.40	80.80	88.90	91.80
Recall	82.80	88.40	89.90	95.50
F-measure	81.60	84.50	89.40	93.60
	State-of-the-art			
	Precision: 84.70, **Recall:** 72.40, **F-measure:** 78.10			

Table 2: Results of gold standard set with different expansion strategies for the first level in the hierarchy. Here, CBE: denotes candidate selection based on classifier's confidence score, SDE: denotes candidate selection based on semantic distance based measurement.

Hierarchical Second Level	**CBE**		**SDE**	
	Gold set	Easy cases	Gold set	Easy cases
Precision	72.50	72.30	73.83	74.02
Recall	69.60	76.90	70.82	77.35
F-measure	71.02	74.53	72.29	75.65
	State-of-the-art			
	Precision: 54.76, **Recall:** 55.23, **F-measure:** 54.99			

Table 3: Results with different expansion strategies for the second level classes in the hierarchy

We make a Multi-rater agreement (Fleiss, 1971) with the classification model and humans' annotations. While we look at the agreement, it was observed that for easy-to-classify instances, there is a considerably high agreement (with more than 85%) among machines (i.e. classifier) and humans. It was also observed that, for the instances where annotators had dis-agreement, classifier was also not able to properly classify-this was confirmed by an expert (non-annotator).

From our further analysis we come up with the following observations: (i). instances where both human and machine commit mistakes: for *"*नया *(nayA-new)* - जिसकी रचना अभी-अभी की गई हो *(jisakI rachanA abhI-abhI kI gaI ho-Which has just been created.)"*, machine assigns *'future'* whereas human assigns *'neutral'*. However, this should be *'present'* as confirmed by an expert. (ii). instances for which human makes mistakes but machine does not: e.g., *'*तुरंत *(turaMta-Immediately)* - शीघ्रता से या बिना विलम्ब किए *(shIghratA se yA binA vilamba kie-Hastily or without delay.)'* is tagged as *'atemporal'* by human, but *'neutral'* by machine. (iii). instances where machine makes mistake but human correctly predicts: e.g., *"*ताजा *(tAjA-fresh)* - हाल ही का *(hAla hI kA-recent)"* is tagged as *'atemporal'* by the machine, but *'present'* by human.

2.6. Error Analysis

We closely analyze the outputs of the classifiers to understand the behaviors of each expansion as well as classification strategy.

CBE: In this model, most of the miss-classified instances are also found to be difficult to the human annotators as temporal senses in these synsets are not directly denoted. As an example, *"*मरणासन्न (maraNAsanna- Moribund) - जो मरने के बहुत समीप हो *(jo marane ke bahuta samIpa ho- One who is very close to death)*" is classified as *'neutral'* even though it connotes 'futuristic' temporal sense.

SDE: This is the most effective model, hence reduces the errors significantly. The model miss-classifies those instances which either do not have any denotative temporal evidence in their glosses or fall into the difficult-to-classify cases, i.e. the human annotators are not even in perfect agreement while classifying.

We observe that both the models have complimentary behaviors. There are instances which are correctly predicted by SDE, but CBE fails and the vice-versa. Significance t-test (De Winter, 2013) confirms that the performance improvement in SDE-based approach over the CBE-based approach is statistically significant.

3. Sentence Level Temporality Detection

As an application of Tempo-Hindi-WordNet that we develop, we evaluate its effectiveness for detecting temporality at the sentence level. Each sentence is classified with one of the three temporal classes, namely *past*, *present* and *future*. As there was no sentence level temporally tagged corpus, we manually create it for benchmarking. Three experts (with post-graduate level knowledge) were asked to manually annotate two kinds of datasets: (i). The first set contains 940 sentences of ILTIMEX corpus (Ramrakhiyani and Majumder, 2015) with 281, 533 and 126 instances of past, present and future, respectively; (ii). The second dataset contains 210 tweets chosen from SAIL dataset (Patra et al., 2015) with 18, 166 and 26 instances of past, present and future, respectively. We find inter-annotator multi-rater kappa agreement (Fleiss, 1971) of 0.80.

We develop two models based on rules and supervised machine learning.

3.1. Rule-based Approach

We define a set of generic rules which we apply for determining the temporal sense of any sentence for both Twitter and Newswire text. We apply the same set of rules for the following two cases: (i). Temporal sense of each word sense in the sentence is detected using our temporal resource. The most suitable sense of each word in the sentence is determined using an unsupervised Most Frequent Sense (MFS) algorithm (Bhingardive et al., 2015). (ii). We identify the tense of each word in a sentence using a Hindi Morphological Analyzer.[7] Verbs with the tense information (past, present or future) are used for developing the rule-based system.

We depict the rules in Algorithm 2.

[7] http://www.cfilt.iitb.ac.in/~ankitb/ma/

Algorithm 2 Basic Steps of Rule-based Approach.

1: If majority words in a sentence belong to a particular temporal/tense category *t* then label it as *t*.
2: If the words in the sentence are equally distributed among the three classes then
   2.1. Label the sentence as present if the classes are only past and present;
   2.2. Label the sentence as future if the classes are only present and future;
   2.3. Label the sentence as future if the classes are only past and future;
   2.4. Label the sentence as future if all the three classes occur.
3: Class label is assigned at random in case no temporal/tense word is detected in the sentence.

Experimental results of this rule-based approach are shown in Table 4. It shows that the classifier created based on our temporal resource performs better than the system based on the tense information. Significance t-test (De Winter, 2013) confirmed that the performance improvement in our resource-based approach over the tense-based approach is statistically significant.

	Tense based	Temporal Resource based
ILITIMEX Corpus	(63.67, 63.18, 63.43)	**(64.90, 67.90, 66.37)**
Twitter Corpus	(45.58, 53.10, 49.06)	**(61.78, 69.65, 65.48)**

Table 4: Results using rule-based approach. Here, (x, y, z): precision, recall, and F-score.

3.2. Supervised Machine Learning Approach

We develop a SVM-based model with the following set of features.

Unigrams(UN): Word unigrams of sentence are used as features of the classifier.

Tense Synset (TenseS): Synsets of words containing tense information are used as the features. Tense information is detected by the same Hindi morphological analyzer (c.f. Section 3.1.).

Temporal Synset(TempS): WodNet synsets of temporal words present in a sentence are used as features. We use Tempo-Hindi-WordNet to determine the temporal sense.

Results of machine learning-based approach are reported in

	UN	UN+TenseS	UN+TempS
ILTIMEX corpus	86.92	87.42	**88.98**
Twitter corpus	84.29	84.61	**86.23**

Table 5: Results of 10-fold cross validation accuracy for machine learning based approach

Table 5. It shows that the best result is achieved when the unigrams and temporal synset features are used together. Significance *t-test* shows that the performance improvement with this feature combination is statistically significant over the others.

3.3. Analysis of Results

In order to study the behaviors of sentence-level taggers, we analyse the outputs of the classifiers. It is found that a number of errors were contributed due to the incorrect sense marking by the MFS disambiguation algorithm (Bhingardive et al., 2015). Let us consider the following example, *"फिलहाल, सब इंस्पेक्टर तीन सप्ताह के बेड रेस्ट पर है (philahAla, saba iMspekTara tIna saptAha ke beDa resTa para hai-Currently, the sub inspector is in three week's bed rest)"*. Here, the MFS algorithm fails to identify the proper sense of the word *"फिलहाल (philahAla-Currently)"*, and thus can not directly detect temporality.

The temporal resource-based system correctly classifies many instances where the tense-based system fails. Let us consider the following example: *"हमारे पास मैच जीतने का मौका है (hamAre pAsa maicha jItane kA maukA hai-We have a chance to win the match)"* which refers to the *'future'* event. Here, the tense-based system classifies it as *'present'*, but our temporal resource-based classifier very correctly tags it as *'future'*. There are instances where the Hindi morphological analyzer fails to detect any tense information. For example, *"अब क्यों नहीं करती नारी अधिकार की बात (aba kyo.m nahii.m karatii naarii adhikaara kii baata-now, why do not you talk about women's right)"*. Here, our temporal resource-based tagger correctly classifies it as *'present'* with the help of temporal keyword *"अब (aba-now)"*. There are also some counter examples where the temporal resource-based classifier fails, but the tense-based classifier behaves properly. For example, *"मौका (maukA-opportunity)"* is a word having connotative *future* time sense. When this word appears in a sentence like *"हमने मैच जीतने का मौका गवा दिया था (hamane maicha jItane kA maukA gavA diyA thA-We missed an opportunity to win the match.)"*, it actually refers to a *past* time sense which is captured correctly by the tense-based model, whereas our resource-based system mis-classifies it as *'future'*.

Our close analysis reveals that the behaviors of rule-based and machine learning-based approaches are very often complimentary in nature, i.e. there are instances where rule-based model succeeds but the machine learning-based approach fails and the vice-versa. For example, consider the following sentence: *"उसके खिलाफ दर्ज केस वापस लेने के लिए सीबीआई पहले ही अर्जी दाखिल कर चुकी है (usake khilApha darja kesa vApasa lene ke lie sIbIAI pahale hI arjI dAkhila kara chukI hai-CBI has already filed an application for withdrawing the case against him)"*. Here, the rule-based approach classifies it as *'present'* but the machine learning-based approach correctly classifies it as *'past'*. In the following sentence: *"उन्हें चाजुन से कड़ी टक्कर मिलने की संभावना है (unheM chAjuna se kaDI Takkara milane kI saMbhAvanA hai-He is likely to get tough competition from Chajun)"*, the machine learning-based approach incorrectly predicts it as *present* but the rule-based approach correctly predicts it as *future*.

There are also some instances where both the rule-based and the machine learning-based approaches fail. For example, *"इस बार उनकी नजरें गोल्ड मेडल पर हैं (isa bAra unakI najareM golDa meDala para haiM-This time her eyes are on gold medal)"*. Here, both rule-based and machine learning-based methods incorrectly classify the sentence as *present*. However, this is actually an instance of *future*. In order to perform quantitative analysis we create confusion matrix that shows that the system is mostly confused in discriminating *present* from the *future* classes.

4. Conclusions

In this paper, we have presented a framework for sentence-level temporality detection in Hindi. In order to achieve this, we propose a semi-supervised learning framework for finding temporal sense of each word in the sentence. This classifies the entire Hindi-WordNet into five classes. We have used three learning algorithms and several expansion strategies. A gold standard test set is also created to perform detailed evaluation. Finally, we show how the temporal resource can be used for temporality detection at the sentence level. We develop two versions: rule-based and machine learning-based. These have been evaluated on two different domain corpora, namely Twitter (informal text) and newswire (formal text). Evaluation shows that such a temporal resource will facilitate research in temporal IR/NLP. Our proposed method is generic and can be adapted to other languages and domains with the availability of minimal resource such as the WordNet.

In future, we will like to investigate a hybrid expansion strategy for resource creation where probabilistic expansion and semantic distance based expansion will be joined together to exploit each other's merit. For sentence-level tagging, we will explore deep learning based methods.

5. Acknowledgements

Asif Ekbal acknowledges Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

6. Bibliographical References

Allan, J., Gupta, R., and Khandelwal, V. (2001). Temporal Summaries of News Topics. In *SIGIR 2001: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 10–18, New Orleans, Louisiana, USA.

Alonso, O., Gertz, M., and Baeza-Yates, R. A. (2009). Clustering and Exploring Search Results using Timeline Constructions. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM*, pages 97–106, Hong Kong, China.

Bhingardive, S., Singh, D., V, R., Redkar, H. H., and Bhattacharyya, P. (2015). Unsupervised Most Frequent Sense Detection using Word Embeddings. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1238–1243, Denver, Colorado, USA.

De Winter, J. C. (2013). Using the Student's T-test with Extremely Small Sample Sizes. *Practical Assessment, Research & Evaluation*, 18(10).

Dias, G. H., Hasanuzzaman, M., Ferrari, S., and Mathet, Y. (2014). TempoWordNet for Sentence Time Tagging. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, pages 833–838, Geneva, Switzerland.

Fleiss, J. L. (1971). Measuring Nominal Scale Agreement among Many Raters. *Psychological Bulletin*, 76(5):378–382.

Hasanuzzaman, M., Dias, G., Ferrari, S., and Mathet, Y. (2014). Propagation Strategies for Building Temporal Ontologies. In *14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 6–11, Gothenburg, Sweden.

Jatowt, A., Yeung, C. A., and Tanaka, K. (2013). Estimating Document Focus Time. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13*, pages 2273–2278, San Francisco, CA, USA.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms.* Kluwer Academic Publisher, Dordrecht, Netherlands.

John, G. H. and Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo, CA.

Joho, H., Jatowt, A., and Blanco, R. (2013). A Survey of Temporal Web Search Experience. In *22nd International World Wide Web Conference, WWW '13*, pages 1101–1108, Rio de Janeiro, Brazil.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781.

Patra, B. G., Das, D., Das, A., and Prasath, R. (2015). Shared Task on Sentiment Analysis in Indian Languages (SAIL) Tweets - An Overview. In *Mining Intelligence and Knowledge Exploration - Third International Conference, MIKE 2015*, pages 650–655, Hyderabad, India.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA.

Ramrakhiyani, N. and Majumder, P. (2015). Approaches to Temporal Expression Recognition in Hindi. *ACM Trans. Asian & Low-Resource Lang. Inf. Process.*, 14(1):2.

Schockaert, S., Ahn, D., De Cock, M., and Kerre, E. E. (2006). Question Answering with Imperfect Temporal Information. In *Flexible Query Answering Systems*, pages 647–658. Springer, Berlin, Germany.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J. F., Verhagen, M., and Pustejovsky, J. (2013). SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013*, pages 1–9, Atlanta, Georgia, USA.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Moszkowicz, J., and Pustejovsky, J. (2009). The Tempeval Challenge: Identifying Temporal Relations in Text. *Language Resources and Evaluation (LRE)*, 43(2):161–179.

Verhagen, M., Saurí, R., Caselli, T., and Pustejovsky, J. (2010). Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval@ACL 2010*, pages 57–62, Uppsala, Sweden.

7. Language Resource References

Bhattacharyya, P. (2010). Indowordnet. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

Bojar, O., Diatka, V., Rychlý, P., Stranák, P., Suchomel, V., Tamchyna, A., and Zeman, D. (2014). HindEnCorp - Hindi-English and Hindi-only Corpus for Machine Translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 3550–3555, Reykjavik, Iceland.

Pawar, D., Hasanuzzaman, M., and Ekbal, A. (2016). Building Tempo-HindiWordNet: A Resource for Effective Temporal Information access in Hindi. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016*, Portorož, Slovenia.

Comprehensive Annotation of Various Types of Temporal Information on the Time Axis

Tomohiro Sakaguchi, Daisuke Kawahara and Sadao Kurohashi
Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
sakaguchi@nlp.ist.i.kyoto-u.ac.jp, {dk, kuro}@i.kyoto-u.ac.jp

Abstract

In order to make the temporal interpretation of text, there have been many studies linking event and temporal information, such as temporal ordering of events and timeline generation. To train and evaluate models in these studies, many corpora that associate event information with time information have been developed. In this paper, we propose an annotation scheme that anchors expressions in text to the time axis comprehensively, extending the previous studies in the following two points. One of the points is to annotate not only expressions with strong temporality but also expressions with weak temporality, such as states and habits. The other point is that various types of temporal information, such as frequency and duration, can be anchored to the time axis. Using this annotation scheme, we annotated a subset of Kyoto University Text Corpus. Since the corpus has already been annotated predicate-argument structures and coreference relations, it can be utilized for integrated information analysis of events, entities and time.

Keywords: Time annotation, Time anchoring, Time axis

1. Introduction

Everyday many texts are generated on the Web, and a huge amount of texts have been accumulated so far. To extract knowledge about a certain topic from this large amount of texts, we need an information analysis technology to integrate, summarize and compare related texts. In order to analyze texts written at different times or texts referring to different times, it is necessary to interpret the temporal information implied in the texts. There have been many studies and tasks to understand the relationship between event information and time information in text. For example, temporal ordering of events that estimates the temporal relations of event-event and event-time was studied in TempEval 1, 2, 3 (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013), and the timeline generation task that links event and time in multiple documents was studied in SemEval 15 (Minard et al., 2015).

In order to train models and evaluate results in these tasks, corpora in which event information is correlated with time information in text have been developed (Pustejovsky et al., 2003; Cassidy et al., 2014; Reimers et al., 2016). In these studies, expressions which have clear temporality were annotated, but in order to know how people understand texts from the perspective of time, it is essential to know how the expressions with weak temporality are interpreted. To understand temporal information in text exhaustively, we propose an annotation scheme that anchors various expressions to the time axis, reflecting personal interpretation of text and common sense. Using this scheme, we annotate Kyoto University Text Corpus (Kawahara et al., 2002), which is a Japanese newspaper corpus annotated with predicate-argument structures and coreference relations.

The points of our annotation scheme are two-fold. One of the points is to annotate various expressions that can have temporality. We annotate not only expressions with strong temporality but also expressions with weak temporality. Many previous studies annotate "events" that express situations that happen or occur, which are defined in the guideline of TimeML (Sauri et al., 2006). Therefore, expressions as in the following example are not annotated.

(1) *Businesses are **emerging** on the Internet so quickly that no one, including government regulators, can keep track of them.*

However, the temporal information of expressions other than "event" also can be a clue to understand text. In the case of the above example, the temporal information of *"emerging,"* i.e., several years ago to the present, should be annotated to clarify the temporal common sense implied in the text. Therefore, we annotate all the expressions that can have temporality, that is, all the predicates and the eventive nouns in text. Annotators judge whether the expressions have temporality, and annotate the corresponding time tags. The other point of our annotation scheme is that various types of time information such as frequency and duration can be anchored to the time axis. Reimers et al. (2016) proposed an annotation scheme that represents an event period using its starting and ending points. However, it cannot represent "non-continuous period" or "a period in a long duration" as in the following examples.

(2) *He **plays** baseball every Sunday.*

(3) *I will **take** a business trip for three days next week.*

(4) *He often used to **have** a tea with us.*

In this paper, we introduce new time tags that can more accurately anchor various types of time information to the time axis.

By annotating various types of temporal information with the expressive time tags, personal interpretation of text and common sense appear as tag disagreements. In this research we consider that such disagreements are also important in understanding how time information is interpreted, and thus we do not eventually integrate time tags annotated by several annotators into one. Instead, we introduce an annotation method that keeps differences in interpretation and only corrects obvious annotation errors.

Using the annotation scheme, we annotated 113 documents with 4,534 expressions in Kyoto University Text Corpus. 80% of the expressions are judged to have temporality, and approximately 30% of them are annotated with the notation newly proposed in this paper. Since the corpus has already been annotated with predicate-argument structures and coreference relations, our annotation makes it possible to utilize for integrated information analysis of events, entities and time. The annotated corpus is publicly available.[1]

2. Related Work

There are many corpora which associate event information with time information, and they can be roughly divided into two approaches. One approach is annotating temporal relations between events. Pustejovsky et al. (2003) annotated events and times based on the TimeML guideline, and relations between event-event, event-time, and event-time. Originally, the annotation was sparse because there were only the relations which are judged to be important by annotators, but TempEval competitions (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013) annotated all the relations in same sentence to improve the coverage.

Kolomiyets et al. (2012) annotated temporal order relations with the nearest event expressions in a corpus of children's stories. Cassidy et al. (2014) annotated all temporal relations in the same sentence and neighbouring sentences.

The other approach is anchoring events to the time axis. Huang et al. (2016) annotated one of five temporal status categories with events in newspaper articles on civil unrest: *Past, On-going, Future Planned, Future Alert, Future Possible*.

Reimers et al. (2016) anchored with finer granularity. Their smallest granularity is day. They divided events into two types: single day event and multiple day event. The former is annotated with the date on which the event occurred, and the latter is annotated with the start and end dates of the event. For example, ***sent*** in the following sentence, an event which ends in one day, is annotated with *1980-05-26*, and ***spent***, an event spanning multiple days, is annotated with *beginPoint=1980-05-26 endPoint=1980-06-01*.

(5) He was ***sent*** into space on May 26, 1980. He ***spent*** six days aboard the Salyut 6 spacecraft.

In the case that the exact event date is not mentioned, notations *before* and *after* are used. In the following sentence, ***appointed*** is annotated with *after 1996-01-01 before 1996-12-31*, and ***part*** is annotated with *beginPoint=after 1984-10-01 before 1984-10-31 and endPoint=after 1984-10-01 before 1984-10-31*.

(6) In 1996 he was ***appointed*** military attache at the Hungarian embassy in Washington. [...] McBride was ***part*** of a seven-member crew aboard the Orbiter Challenger in October 1984

In their annotations, about 60% of all the events end in a day, and about 40% are events that span multiple days. 56% of the former have precise dates, and of the latter, 20% have precise start dates and 16% have precise end dates.

In this research, we extend the anchoring to the time axis approach, and propose annotation scheme that can deal with various time information in text.

3. Annotation Scheme

We annotate expressions which consist of all predicates and eventive nouns in text (hereinafter referred to as "target expressions"). We first apply morphological analysis to text and extract base phrases of predicates and eventive nouns. Annotators first judge whether the expressions have temporality. Expressions that have temporality are annotated with time tags which represent the corresponding time value in consideration of the document creation time (DCT) and the context. When an expression is judged to have no temporality, it is annotated with the time tag *not applicable (t:n/a)*.

A time tag that has temporality is expressed as a Time Base Unit (TBU) or a combination thereof (Table 1). TBU represents a specific time point, and TBUs are divided into three types. There are four ways of combining these, which enable to represent various types of time information. Reimers et al. (2016) used only TBU 1 and combinations a and c in Table 1. As in the previous studies, the finest granularity of time tags is day.

Although we annotate the Japanese corpus, we use English examples to explain our annotation scheme below.

3.1. Judgement of Temporality

In order to judge the temporality of a target expression, annotators consider whether it implies a change in the behavior or state between the past and future. In the case that the expression focuses on a change, it has temporality. In the following examples, ***go*** in example (7) and ***thriving*** in example (8) have temporality, and ***eat*** in example (9) has no temporality. Note that ***thriving*** in example (8) is an expression which is not subject to annotate in the previous studies.

(7) *He will **go** to Kyoto tomorrow.*

(8) *Language processing research is **thriving**.*

(9) *Rabbits **eat** grass.*

3.2. Time Base Unit (TBU)

3.2.1. Date Tag

The temporal information of a date expression is represented by annotating the time value in *t* tag. The time value notation in Japanese TimeBank Corpus (Asahara et al., 2014) is used, such as *t:YYYY* and *t:YYYY-MM-DD*. To reduce the annotation cost, the date tag of the document creation time can be written as *t:DCT*.

For example, ***arrived*** in the following sentence is annotated with *t:2017-04-28*.

(10) [DCT: 2017-04-29] *The president **arrived** in New York yesterday.*

Unlike the previous studies, our annotation scheme allows time tags with larger granularity than day. For example, ***hot*** in the following example is annotated with *t:2016-08*.

[1] http://nlp.ist.i.kyoto-u.ac.jp/index.php?KUTBC

Temporality	Time Base Unit (TBU)	1. Date tag (Day, Month, Year)	e.g., *t:1995-01-05*
		2. Vague time tag (Past, Present, Future)	e.g., *t:PRESENT*
		3. Relative tag (Time Coreference, Utterance Day)	e.g., *t:election*
	Combinations of TBUs	a. Interval between TBUs (TBU∼TBU)	e.g., *t:1995-01-05∼1995-01-07*
		b. Specific span in a TBU (span)	e.g., *t:1995-01,span:P1W*
		c. Unspecific span in a TBU (partial span)	e.g., *t:1995-01,span:part*
		d. Repetition of TBUs (freq)	e.g., *t:1995-01,freq:2/P1W*
No Temporality			e.g., *t:n/a*

Table 1: List of time tags

(11) [DCT: 2017-04-29] *It was **hot** last August.*

This tag does not necessarily mean exactly from August 1st to 31st. As the expression "August" is different from "August 1st to 31st," the period of the corresponding time tag is somewhat vague. The granularity of the time tags in this paper implies such vagueness.

3.2.2. Vague Time Tag

There are many expressions that represent vague time in text. In the following sentence, it is not clear when and how long ***live*** represents in the past.

(12) *I used to **live** in Hiroshima.*

Reimers et al. (2016) interpreted this expression as "a period from one day to another day until today" and annotated it with *beginPoint=before DCT endPoint=before DCT*. In our annotation scheme, some special tags are introduced. The vague past, present and future are represented as *t:PAST*, *t:PRESENT* and *t:FUTURE*, respectively. *t:PRESENT* includes not only today but also a little past and future. In the following sentence, ***bring*** is annotated with *t:PRESENT* since it represents not only today but also a little before and after today.

(13) *You can **bring** liquids on domestic flights.*

To represent the past and future, *t:PAST-M*, *t:PAST-Y*, *t:FUTURE-M* and *t:FUTURE-Y* tags are also available according to the temporal distance. *t:PAST-M* represents a few months ago and *t:PAST-Y* represents a few years ago. For more than a few years ago, or when the granularity is unknown, *t:PAST* is used. It is the same for future.
There are other vague time expressions. In the case of expressions that represent numerical ambiguity, such as "around 1980" or "about 3 years", *ap* (approximately) is attached to the ambiguous numerical value of the time tag. In the following sentence, ***built*** is annotated with *t:1980ap*.

(14) *The hotel was **built** around 1980.*

3.2.3. Relative Tag

In texts with few temporal expressions, such as novels, it is difficult to anchor events to the time axis. In such a case, the TimeBank Corpus' annotation scheme, i.e., annotating the temporal relation between events, provides richer information. Therefore, in the case where the specific date is unknown but the temporal relation with another phrase in the same sentence is known, that phrase is used as a time value (*Time Coreference*). In the following sentence, though the date on which the demonstration took place is unknown, it can be understood that it is the day after the election. In this case, ***held*** is annotated with *t:election+P1D*, using the notation of the duration expressions in TimeBank Corpus (see subsection 3.3.2. for details).

(15) *The day after the election, a large demonstration was **held**.*

If there are two or more phrases that can be referred to, priority is given as follows and one with the highest priority is selected: 1. phrase with absolute time value tag, 2. phrase with the closest distance.
In conversational sentences and interviews, the date of the speech is often unknown. If the date of the utterance cannot be guessed from the context, the date can be described as *t:UD* (*Utterance Day*). In the following sentence, ***work*** is annotated with *t:UD*. Note that ***said***, an expression outside the utterance, is annotated with the absolute time value.

(16) *"I have no choice but to **work** hard from now," said the director.*

3.3. Combinations of TBUs

3.3.1. Interval between TBUs

The interval between TBUs is represented by connecting the starting TBU and the ending TBU with ∼. This notation corresponds to the *beginPoint* and *endPoint* tags in Reimers et al. (2016). If either of the starting or ending TBU cannot be guessed, it is omitted. The time tag of ***busy*** in the following sentence is *t:∼2017-04-28*.

(17) [DCT: 2017-04-29] *I was **busy** up until yesterday.*

3.3.2. Span in a TBU

A part of the period in a long TBU, e.g., a part of the period in August, is represented by combining the *t* tag representing the large period and the *span* tag representing the small period. When the length of the small period is guessed, the *span* tag is represented using the notation of the duration expressions defined in the Japanese TimeBank Corpus. For example, three years is represented as *span:P3Y*, three weeks is represented as *span:P3W* and three days is represented as *span:P3D*. In the following sentence, ***am going*** is annotated with *t:2017-05,span:P1W*.

(18) [DCT: 2017-04-29] *I **am going** to London for a week next month.*

If the length of the small period cannot be guessed, it is represented as *span:part*. The *span:part* tag is equivalent to the *before* and *after* tags in Reimers et al. (2016).

	Annotator1	Annotator2	Annotator3	Average
Date tags	1195 (26.4%)	1145 (25.3%)	938 (20.7%)	1093 (24.1%)
Year	35 (0.8%)	47 (1.0%)	16 (0.4%)	33 (0.7%)
Month	9 (0.2%)	14 (0.3%)	3 (0.1%)	9 (0.2%)
Day	1151 (25.4%)	1084 (23.9%)	919 (20.3%)	1051 (23.2%)
Vague time tags ∗	617 (13.6%)	375 (8.3%)	249 (5.5%)	414 (9.1%)
t:PRESENT	520 (11.5%)	257 (5.7%)	195 (4.3%)	324 (7.2%)
t:PAST	40 (0.9%)	25 (0.6%)	17 (0.4%)	27 (0.6%)
t:FUTURE	57 (1.3%)	89 (2.0%)	31 (0.7%)	59 (1.3%)
t:ap	0 (0.0%)	4 (0.1%)	6 (0.1%)	13 (0.3%)
Relative tags ∗	215 (4.7%)	135 (3.0%)	313 (6.9%)	410 (8.0%)
Tags including Time Coreference	138 (3.0%)	58 (1.3%)	207 (4.6%)	134 (3.0%)
Tags including *t:UD*	77 (1.7%)	77 (1.7%)	106 (2.3%)	87 (1.9%)
Interval between TBU (∼)	387 (8.5%)	562 (12.4%)	842 (18.6%)	597 (13.2%)
Specific span in a TBU (span) ∗	540 (11.9%)	447 (9.9%)	550 (12.1%)	512 (11.3%)
Date + span	46 (1.0%)	69 (1.5%)	96 (2.1%)	70 (1.6%)
∼ + span	482 (10.6%)	357 (7.9%)	434 (9.6%)	424 (9.4%)
Vague + span	12 (0.3%)	21 (0.5%)	20 (0.4%)	18 (0.4%)
Unspecific span in a TBU (span:part)	455 (10.0%)	561 (12.4%)	478 (10.5%)	498 (11.0%)
Date + span:part	36 (0.8%)	56 (1.2%)	46 (1.0%)	46 (1.0%)
∼ + span:part	373 (8.2%)	475 (10.5%)	391 (8.6%)	413 (9.1%)
Vague + span:part	46 (1.0%)	30 (0.7%)	41 (0.9%)	39 (0.9%)
Repetition of TBU (freq)	46 (1.0%)	52 (1.2%)	47 (1.0%)	48 (1.1%)
No temporality (*t:n/a*)	1071 (23.6%)	1077 (23.8%)	1060 (23.4%)	1069 (23.6%)
Tags marked in the second step	8 (0.2%)	180 (4.0%)	57 (1.3%)	82 (1.8%)
Newly proposed tags	1372 (30.3%)	957 (21.1%)	1112 (24.5%)	1147 (25.3%)
All	4534	4534	4534	4534

Table 2: Distribution of annotated time tags. Time tags with ∗ are newly proposed in this paper.

3.3.3. Repetition of TBU

There are many target expressions that are not represented as continuous periods, such as "every Sunday" and "once every three days." Target expressions occurring across multiple days repeatedly are represented with *freq* tag.
The *freq* tag can be used in three ways.

1. When the repetition is expressed as a number of occurrences during a certain period, such as "twice a week" and "once every three days," the *freq* tag is represented as *the number of times / period*. In the following sentence, ***go*** is annotated with *t:2016-07∼DCT,freq:2/P1W*.

   (19) [DCT: 2017-04-29] *I **go** to the pool twice in a week since July 2016.*

2. When the repetition is expressed as a repetition of specific date, such as "every 25th day" and "every Sunday," the date is used as a value of the *freq* tag. The Japanese TimeBank Corpus' notation is extended by allowing to include the symbol @ in each part of *YYYY-MM-DD* in the sense that it can represent any number. In the following sentence, ***is held*** is annotated with *t:PRESENT,freq:@@@@-@@-25*.

   (20) [DCT: 2017-04-29] The Tenjin market ***is held*** on the 25th of every month.

3. When the repetition or the frequency cannot be guessed from the context, one of the following four abstract tags is used: *usually, often, sometimes* and *rarely*. In the following sentence, ***go*** is annotated with *t:PRESENT,freq:sometimes*.

   (21) [DCT: 2017-04-29] I sometimes ***go*** to Starbucks.

	Strict	Relax
The first step	0.417	0.719
The final result	0.554	0.802
The first step (Excluding *t:n/a*)	0.380	0.803
The final result (Excluding *t:n/a*)	0.526	0.867
[Reimers+ 16]	0.617	0.912

Table 3: Inter-annotator agreement computed by Krippendorff's α.

4. Annotation Study

4.1. Annotation Method

Using our annotation scheme, we annotated a subset of documents in Kyoto University Text Corpus. The subset consists of 4,534 target expressions in 113 documents. Kyoto University Text Corpus is a Japanese newspaper corpus that was manually annotated with various linguistic information, such as predicate-argument structures and coreference relations.
The time tags were annotated by three annotators. Since we annotate expressions whose interpretation varies depending on the individual's common sense, we do not eventually combine the annotators' tags into one. We introduce a two-step annotation method that keeps the interpretation

Agreed between annotators			Disagreed between annotators		
Pair of abstracted time tags		Frequency	Pair of abstracted time tags		Frequency
n/a	n/a	800	PRES	n/a	110
DAY	DAY	740	DAY	n/a	105
∼DAY,span	∼DAY,span	142	DAY	∼DAY,span	104
PRES	PRES	113	∼DAY,span	∼DAY,span	77
DAY∼,span	DAY∼,span	54	DAY	∼DAY	73
DAY∼DAY	DAY∼DAY	38	PRES	∼DAY,span	59
YEAR	YEAR	12	PRES	∼DAY	49
DAY∼FUTURE	DAY∼FUTURE	10	PRES	PAST∼DAY	49
All		2045	All		2275

Table 4: Frequency of agreed/disagreed time tags in the first step in the strict metric

Agreed between annotators			Disagreed between annotators		
Pair of abstracted time tags		Frequency	Pair of abstracted time tags		Frequency
n/a	n/a	800	PRES	n/a	110
DAY	DAY	741	DAY	n/a	105
∼DAY,span	∼DAY,span	219	DAY	DAY	44
PRES	PRES	113	DAY∼,span	n/a	36
∼DAY,span	DAY	90	∼DAY,span	n/a	31
DAY∼,span	DAY∼,span	85	DAY∼FUTURE	n/a	20
DAY	∼DAY	67	DAY∼FUTURE,span	n/a	19
∼DAY,span	PRES	59	DAY∼DAY	n/a	14
All		3533	All		787

Table 5: Frequency of agreed/disagreed time tags in the first step in the relaxed metric

of other annotators and modifies only obvious annotation errors. The document set is divided into three parts. Each annotator annotates two of them in the first step, and the remaining one is annotated in the second step. In the first step, each annotator independently annotates, and in the second step they annotate tags by confirming the others' tags in the first step. If an obvious error is found in the already annotated tags, it is just marked. The marked tags are 2% of the total and are treated as missing values in the analysis in section 5.

4.2. Distribution of the Annotated Time Tags

The distribution of the annotated time tags is shown in Table 2. Approximately 80% of the target expressions have temporality, and 55% of them are TBU and the others are combinations of TBUs. The date tags account for approximately 25%, while the vague time tags and the relative tags account for approximately 10%. Since the domain of annotation is newspaper, the majority of target expressions are directly anchored to the time axis. The *freq* tag, representing repetition, is hardly used, i.e., 1% of the whole. The time tags that are newly proposed in this paper account for 25% of the whole.

4.3. Inter-Annotator Agreement

We compute the inter-annotator agreement using Krippendorff's α (Krippendorff, 2004; Hayes and Krippendorff, 2007). Following Reimers et al. (2016), two metrics are utilized. One is a strict metric that measures whether the time tags completely match. The other is a relaxed metric that permits partial matching. If the time tags are overlapped even for one day, they are regarded as matched, and if they do not overlap at all, they are regarded as mismatched. Table 3 shows the agreement at each step. "Excluding *t:n/a*" means an agreement computed excluding the expressions in which one or more annotators annotated with *t:n/a*.

Comparing the first step and the final result of the annotation process, the latter agreement increased significantly. This is because while the documents are annotated independently in the first step, annotators can check others' tags in the second step. When the target expressions annotated with *t:n/a* are excluded, the relaxed agreement increased significantly. It shows that the difficulty of temporality judgement is a cause of lowering the agreement in relaxed metric. Compared with previous studies, the agreement in the strict metric is particularly low. Due to the increase of the variation of the time tags, annotators' interpretations can be reflected a lot, and it became difficult to agree completely.

5. Disagreement Analysis

In order to analyze the annotated time tags without being limited to specific values, we abstract them from the aspect of granularity. For example, for the *t* tag, *t:1994-12-31* is abstracted as *DAY*, *t:∼1994-12-31* is abstracted as *∼DAY* and *t:1994* is abstracted as *YEAR*. For the *span* tag and the *freq* tag, their values are omitted. For example, *t:∼1994-12-31,span:P1D* and *t:∼1994-12-31,span:part* are both abstracted as *∼DAY,span*.

In this section, we analyze the results of the first stage, where annotators independently annotated. Tables 4 and 5 show disagreements in the strict and relaxed metrics respec-

tively. Table 4 indicates that in the strict metric, about 70% of agreed tags are *DAY* and *n/a*, and most of the disagreements are the judgement of temporality and the interpretation of date and period such as *DAY* and ∼*DAY*. Table 5 indicates that most of the disagreements in the relaxed metric are the judgement of temporality. It indicates that most of the tags that were disagreed due to the interpretation between date and period in the strict metric overlap the spans, and they are consistent in the relaxed metric.
In the following subsection, we analyze the disagreement of temporality judgment and the disagreement of interpretation of the date and the period with actual examples.

5.1. Judgement of Temporality

In the relaxed metric, the biggest cause of disagreements is that the judgement of temporality varies depending on annotators. When one annotator tags *n/a*, the other annotates *n/a* (76.6%), *DAY* (5.3%), *PRESENT* (5.0%), ∼*DAY,span* (1.7%) in order of frequency. This means that 75% of *n/a* tags agree, and if it is not the case, one annotates the *DAY* or *PRESENT* tag at a rate of 40%. Many of these expressions represent states, positions and organizations, and the judgment is divided according to whether it is interpreted as permanent or as a temporal period.
In the following sentence, one annotated *t:PRESENT* and the other annotated *t:n/a*.

(22) 大統領官邸の**ある**中心部
The city center where the presidential official residence ***exists***

The annotator who recognized temporality interpreted that there is a possibility that the place of the presidential office may change in the future, while the other interpreted it as semi-permanent.

5.2. Interpretation of Date and Period

As Reimers et al. (2016) pointed out, it is difficult to judge whether an event ends in one day or is held for several days from a text. It is also not easy to clarify the beginning and ending date of an event. Such vagueness appears as disagreements among *DAY*, ∼*DAY*, ∼*DAY,span*, *DAY*∼, *DAY*∼*,span* and *PRESENT* in this annotation scheme. Among them, the disagreement between *DAY* and ∼*DAY,span* often occurs. In many cases, *DAY* is DCT, which means that it is difficult to interpret whether it occurred at the written date or before that.
In the following sentence, it is difficult to judge the duration of the event ***resists*** from the text. One annotated *t:DCT* and the other annotated *t:*∼*DCT,span:part*. While the former interpreted that the event occurred in a day, the latter interpreted as a longer period.

(23) しかしドゥダエフ政権部隊は頑強に**抵抗**、双方の死者は数百人に達する見込みだ。
But the Dudaev regime strongly ***resists***, and the death toll will reach hundreds.

One of the difficulties is due to the domain being newspaper. In the following sentence, one annotated *t:DCT* and the other annotated *t:*∼*DCT,span:P1D*. While the former interpreted that it happened on the date when the article was written from the promptness of newspaper, the latter interpreted that it was not necessarily so.

(24) 外相は、「非民営化・再国営化」の基本方針を**打ち出した**。
The Foreign Minister ***has laid out*** the basic policy of "privatization and re-nationalization."

Thus, the major cause of the disagreements among the annotators is that there are multiple interpretations depending on the context and common sense, closely related to the writing style and theme of newspaper.

6. Conclusion

In this paper, we described a new annotation scheme for comprehensively annotating temporal information in texts reflecting personal interpretation and common sense. Using this scheme, we annotated a subset of a Japanese newspaper corpus, and the new tags account for approximately 25% of the all tags.
Though we annotated newspaper articles in this research, it seems that the writing style and its character are one of causes of the annotation disagreement. In the future, we would like to try annotating on a corpus other than newspaper such as Web texts.

Acknowledgments

This work was partially supported by JST CREST Grant Number JPMJCR1301 including AIP challenge program, Japan. We also thank Manami Ishikawa, Marika Horiuchi and Natsuki Nikaido for their careful annotation.

7. References

Asahara, M., Kato, S., Konishi, H., Imada, M., and Maekawa, K. (2014). BCCWJ-Timebank: Temporal and Event Information Annotation on Japanese Text. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 19, Number 3, September 2014*.

Cassidy, T., McDowell, B., Chambers, N., and Bethard, S. (2014). An Annotation Framework for Dense Event Ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506, Baltimore, Maryland, June. Association for Computational Linguistics.

Hayes, A. F. and Krippendorff, K. (2007). Answering the Call for a Standard Reliability Measure for Coding Data. *Communication Methods and Measures*, 1(1):77–89.

Huang, R., Cases, I., Jurafsky, D., Condoravdi, C., and Riloff, E. (2016). Distinguishing Past, On-going, and Future Events: The Eventstatus Corpus. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 44–54. Association for Computational Linguistics.

Kawahara, D., Kurohashi, S., and Hasida, K. (2002). Construction of a Japanese Relevance-tagged Corpus. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, Las Palmas, Canary Islands - Spain, May. European Language Resources Association (ELRA).

Kolomiyets, O., Bethard, S., and Moens, M.-F. (2012). Extracting narrative timelines as temporal dependency structures. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, The 50th annual meeting of the association for computational linguistics, Jeju, Republic of Korea, 8-14 July 2012*, pages 88–97. ACL.

Krippendorff, K. (2004). *Content Analysis: An Introduction to Its Methodology (second edition)*. Sage Publications.

Minard, A.-L., Speranza, M., Agirre, E., Aldabe, I., van Erp, M., Magnini, B., Rigau, G., and Urizar, R. (2015). SemEval-2015 Task 4: TimeLine: Cross-Document Event Ordering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 778–786, Denver, Colorado, June. Association for Computational Linguistics.

Pustejovsky, J., Hanks, P., Saurí, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Beth Sundheim, D. D., Ferro, L., and Lazo, M. (2003). The timebank corpus. In *Proc. Corpus Linguistics 2003*, pages 647–656.

Reimers, N., Dehghani, N., and Gurevych, I. (2016). Temporal Anchoring of Events for the Timebank Corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2195–2204, Berlin, Germany, August. Association for Computational Linguistics.

Sauri, R., Littman, J., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006). TimeML Annotation Guidelines, Version 1.2.1.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). SemEval-2007 Task 15: TempEval Temporal Relation Identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80. Association for Computational Linguistics.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010). SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62. Association for Computational Linguistics.

Agreements and Disagreements in Temporal Processing: An Extensive Error Analysis of the TempEval-3 Systems

Tommaso Caselli†, **Roser Morante**‡
†CLCG – University of Groningen, ‡CLTL Lab – VU Amsterdam
The Netherlands
†t.caselli@gmail.com|rug.nl, ‡r.morantevallejo@vu.nl

Abstract

In this article we review Temporal Processing systems that participated in the TempEval-3 task as a basis to develop our own system, that we also present and release. The system incorporates high level lexical semantic features, obtaining the best scores for event detection (F1-Class 72.24) and second best result for temporal relation classification from raw text (F1 29.69) when evaluated on the TempEval-3 data. Additionally, we analyse the errors of all TempEval-3 systems for which the output is publicly available with the purpose of finding out what are the weaknesses of current approaches. Although incorporating lexical semantics features increases the performance of our system, the error analysis shows that systems should incorporate inference mechanisms and world knowledge, as well as having strategies to compensate for data skewness.

Keywords: temporal processing, error analysis, written corpora

1. Introduction

Any discourse, spoken or written, contains temporally connected linguistic mentions, such as events and temporal expressions (*timexes*). Relations between these mentions can be meaningfully interpreted by using models of time, which allow to connect events on a timeline (temporal anchoring) and to understand complex sequences of events (temporal ordering). Temporal relations (TRs) provide a model and a set of properties to account for the connections between pairs of entities.

Temporal Processing (TP) is a task consisting in automatically identifying and classifying basic entities and their relations, such as event-event (**e-e**), and event-timex (**e-t**). Temporally aware Natural Language Processing (NLP) systems are crucial not only to generate timelines and storylines (Vossen et al., 2015), but also in decision support systems, summarization and textual entailment applications, question answering systems, and document archiving, among others. Since the release of the TimeBank corpus (Pustejovsky et al., 2003) there has been a renewed interest in the area of TP, which has resulted in the celebration of several evaluation campaigns[1] and in the creation of corpora and tools in languages other than English. [2] The TempEval-3 campaign (UzZaman et al., 2013) is the latest campaign on open-domain TP in English. The major contribution of TempEval-3 is the release of a platform for the development and evaluation of end-to-end TP systems based on the TimeML mark-up language (Pustejovsky et al., 2003).

After the TempEval-3 evaluation a new dataset has been released, the TimeBank-Dense corpus (Cassidy et al., 2014). This corpus has been developed to address one of the major shortcomings of the TempEval-3 dataset, namely lack of connectivity between all possible **e-e** and **e-t** pairs, by completing the transitive closure. The outcome of this approach is a very dense temporal graph consisting of 12,713 annotated TRs, with a 6.3 ratio of relations to events and timexes, which is much higher than the 0.8 ratio of the TempEval-3 data, where 11,098 TRs were manually annotated. Density has been obtained by forcing the annotators to always provide an answer. Additionally, the set of TRs has been simplified with respect to the one used in TempEval-3. Only 6 values are used instead of the original 14: `BEFORE`, `AFTER`, `INCLUDES`, `IS_INCLUDED`, `SIMULTANEOUS`, and `VAGUE`. The value `VAGUE` applies to cases where there is no TR between **e-e** and **e-t** pairs, and to cases where a TR exists but the specific value cannot be reliably determined. Not surprisingly, the `VAGUE` value is the most frequently used (selected 5910 times).

This paper focuses on analysing errors of several TP systems that participated in the TempEval-3 campaign, to identify their limitations. Our study builds on the work by Derczynski (2013), who proposes a classification of TR errors as a result of analysing the output of systems participating in the TempEval-2 campaign. However, our error analysis is different due to differences in the setting of the two competitions: the test data of TempEval-2 is a subset of the TimeBank corpus, whereas in TempEval-3 the evaluation is conducted on a new set of manually annotated documents and the entire TimeBank corpus is made available for training; systems participating in TempEval-2 produced a simplified set of TRs, whereas systems participating in TempEval-3 are asked to provide the full set of fine-grained TRs which have been proposed in TimeML and annotated in the TimeBank corpus; and in TempEval-2 systems were not evaluated on an end-to-end approach, whereas in TempEval-3 they were.

Two are the contributions of this paper: first, we review state-of-the-art TP systems to identify their properties (i.e. features and learning algorithm), common characteristics, and limitations. Based on that we have developed our own

[1] TempEval (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013), Clinical TempEval (Bethard et al., 2016; Bethard et al., 2017), Q-A TempEval (Llorens et al., 2015).

[2] For an extended list of available TimeBanks see (Caselli and Sprugnoli, 2017).

system, which we release to the public.[3] Secondly, we have conducted an extensive error analysis by comparing the output of different systems, including our own, to provide a better understanding of the limitations and issues that still need to be addressed in this task.

The remainder of the paper is structured as follows: Section 2. explains the TP task in general and as formulated in the TempEval-3 evaluation exercise. Section 3. reviews the TP systems that participated in the TempEval-3 competition whose output is publicly available, highlighting commonalities, differences, and limitations. The results of the error analysis are presented in Section 5. and Section 6. for event trigger detection and temporal relation classification, respectively. Finally, Section 7. puts forward conclusions and future work.

2. Task Description

TP is a concatenation of 4 subtasks: identification and classification of linguistic mentions that denote events (ES); detection and normalization of timexes (TE); identification of **e-e** and **e-t** pairs (TD); and classification of valid temporal relations according to a predefined set of values (TC).

TempEval-3 is a follow-up of two previous evaluation campaigns (TempEval and TempEval-2), with the difference that the task of TP is evaluated from an end-to-end perspective, i.e. systems should produce full temporally annotated documents starting from raw text. The TempEval-3 datasets are compliant with the TimeML Annotation Guidelines (Saurı et al., 2006). In particular, an *event* is defined as any linguistic mention, including verbs, nouns, adjectives and prepositional phrases, which denotes something that happens, occurs, or describes states/circumstances in which something obtains or holds true. Each event mention is further characterized by a set of 5 attributes: class, tense, aspect, polarity, and modality.

Timexes are defined as lexical items which denote a time, a date, a duration, or a set (e.g. *noon*, *yesterday*, *two days ago*, *yearly*), extending previous annotation initiatives such as TIDES (Ferro et al., 2002) and STAG (Setzer, 2001). Finally, the set of possible TRs is based on Allen's temporal intervals consisting of a total of 14 possible values: `BEFORE`, `AFTER`, `INCLUDES`, `IS_INCLUDED`, `BEGINS`, `ENDS`, `BEGUN_BY`, `ENDED_BY`, `SIMULTANEOUS`, `IAFTER`, `IBEFORE`, `DURING`, `DURING_INV`, `IDENTITY`. The value `IDENTITY` is actually non-temporal, but it is used to identify coreference relations between event mentions.

For the TempEval-3 campaign extra training data were provided, by automatically annotating almost 600,000 tokens for event, timexes, and TRs. Additionally, a new test data was released with manual gold annotations (20 articles, 8,000 ca. tokens). Evaluation was conducted by means of a new evaluation measure, aimed at assessing the *temporal awareness* of end-to-end systems (UzZaman et al., 2013). Temporal awareness measures the ability of a system to identify and classify TRs. This includes the correct identification and classification of the temporal entities participating in the TR, i.e. event mentions and timexes.

[3] https://github.com/cltl/TimeMLEventTrigger

Table 1 contains the number of events in the TempEval-3 data splitted by part-of-speech (POS), and Table 2, the distribution per value of TRs for the manually annotated (Training-Gold and Test) and the automatically annotated (Training-Auto) training data .

Event POS	Training-Gold	Training-Auto	Test
Verb	5837	65813	539
Noun	2450	13489	169
Adjective	202	473	23
Preposition	10	0	1
Other	19	879	15
Overall	11108	80654	749

Table 1: Events per POS in the TempEval-3 dataset.

TLINK Value	Training Gold	Training Auto	Test
BEFORE	3701	45581	330
AFTER	1361	35241	200
INCLUDES	1523	5062	91
IS_INCLUDED	2287	16029	177
BEGINS	110	7	2
ENDS	77	0	3
BEGUN_BY	70	5	3
ENDED_BY	137	0	2
SIMULTANEOUS	580	10894	93
IAFTER	49	84	10
IBEFORE	65	8	8
DURING	280	0	2
DURING_INV	0	0	1
IDENTITY	858	0	15
Overall	11098	112911	937

Table 2: TRs per temporal value in the TempEval-3 dataset.

3. Temporal Processing Systems: a Review

In order to develop our own out-of-competition TP system, we analyzed first the best systems from TempEval-3 that targeted either the event extraction and classification subtask only (Task B in the TempEval-3 guidelines) or the end-to-end temporal relation identification and classification subtask (Task C in the TempEval-3 guidelines, which includes Task B as well). In total we review 6 unique systems (5 for event detection and classification only and 4 for the full TP).

3.1. Event detection and classification

The event detection and classification task is addressed by all systems using supervised discrete machine learning classifiers such as Conditional Random Fields (CRFs) (Kolya et al., 2013; Bethard, 2013), Logistic Regression (Kolomiyets and Moens, 2013), and Maximum Entropy (Chambers, 2013; Jung and Stent, 2013). Most of the systems (4 out of 5) adopted the same learning model also for event classification. Overall, 17 features are represented in the learning models, which can be aggregated in 5 groups:

- Basic morpho-syntactic features: token, lemma, stem, parts-of-speech (POSs), token's affix and/or suffix, among others.
- Syntactic features: constituency/dependency syntax relations; governing verb lemma, verb chunks.
- Contextual features: context windows of token, lemma, POS; and tokens polarity, among others.
- Semantic features, limited to semantic roles.
- Lexical semantic features, limited to WordNet synsets and hypernyms.

For the event detection task the learning models are more complex in terms of features than for the classification task. Semantics and lexical semantics features are used by less systems (2 systems for event detection and only 1 system for classification), and they are limited to WordNet data only. F1 scores range from 78.2 to 81.05 for event detection, but decrease for event class (from 52.69 to 71.88), and other attributes, such as tense (from 49.7 to 61.63) and aspect (from 63.2 to 73.5). The analysis of systems shows that:

- The identification of event attributes benefits from the use of event detection features only when a rich feature set is used that includes extended syntactic (constituent and dependency parsing) and semantic information (semantic roles).
- Reducing the feature complexity for event attribute identification does not help a machine learning system to generalize enough to beat a baseline based on the most frequent values (Bethard, 2013), but it allows machine learning systems to perform better than rule-based solutions, at least for this dataset (Kolya et al., 2013).
- Varying the composition of the training data (i.e., only gold data (Chambers, 2013) vs. gold data plus silver data (Jung and Stent, 2013)) affects the final results of the specific problem addressed, by either improving the results, like for the Event Trigger and Class subtask, or by downgrading them, as in the case of the TR substask.

3.2. Temporal relation detection and classification

The temporal relation detection and classification task is addressed as a supervised multi-class classification task. Systems use either a single classifier (Maximum Entropy (Chambers, 2013); CRFs (Kolya et al., 2013)) or a combination of two classifiers (SVM and Logistic Regression (Kolomiyets and Moens, 2013); SVM and Maximum Entropy (Bethard, 2013)).
Three of four systems solve the task in a two-step approach: recognition of eligible temporal relations and assignment of the temporal values. Only one system (Bethard, 2013) uses a single step approach, introducing the value NORELATION for negative examples. All systems incorporate different classifiers for different subsets of relations (**e-e**, **e-t**, and event-document creation time (DCT) pairs (**e-dct**)). Only two systems (Chambers, 2013; Kolya et al., 2013) incorporate classifiers for intra- and inter-sentence relations, while the others deal only with intra-sentence relations. Finally, two systems (Bethard, 2013; Kolomiyets and Moens, 2013) use a reduced set of temporal values, while the others adopted the full 14 temporal values.
The feature set for classification of TRs is larger than for event detection and classification, up to 29 features per system, and scattered. There are specific features for some sub-types of TRs (e.g. syntactic path between **e-t** pairs, timex tokens, and linear order in the text, among others). Most of the features fall into the same categories of the event detection and classification task, although some extra features are used: tense and aspect values, order of presentation of the events, presence of temporal prepositions/adverbs, and type of timexes, which are grounded in linguistic theories of time (Reichenbach, 1947; Comrie, 1985; Declerck, 1986). Features which account for discourse structure and world knowledge are either missing or simplified (e.g. only WordNet synsets).
The best system obtains 30.98 F1 for global temporal awareness (Bethard, 2013). It uses a reduced set of temporal relations (3 for **e-dct** and 2 for **e-t** and **e-e**), and models only intra-sentence relations.

4. CRF4TimeML: A New TP System

Based on our study of participating systems, we developed a new end-to-end TP system, CRF4TimeML. Similarly to previous work, we used a single learner and we split the task in multiple subtasks. The system is based on a cascade of 7 CRF classifiers. At this stage of development, we have chosen to use a discrete classifier (CRF), relying more on feature engineering than on distributed feature representations, such as word embeddings, and neural network architectures. This choice is motivated by two main reasons: i) we aim at understanding both the fitness and the limitations of the selected features for a high-level semantic task such as TR identification and classification; ii) we intend to establish a relation between the errors made by the systems, including ours, and the features used.
CRF4TimeML has been designed taking as reference efficient existing systems that incorporate discrete classifiers. In particular, all classifiers we developed share with previous systems basic morpho-syntactic features, such as token, lemma, POS, and dependency relations. However, we have added lexical semantic information by using not only WordNet synsets, but also VerbNet classes and FrameNet frames, obtained from the alignments in the Predicate Matrix (Lacalle et al., 2014). The pre-processing of data is performed with state-of-the-art tools, such as the Stanford CoreNLP (Manning et al., 2014) and the NewsReader NLP pipelines (Agerri et al., 2014). [4]

4.1. Event Detection and Classification

The event detection and classification task is performed by 4 different classifiers share the basic morpho-syntactic and

[4]The timex detection and normalization task is performed using a state-of-the-art system (Bethard, 2013), available at `https://bitbucket.org/qwaider/textpro-en`.

lexico-semantic features: one for the identification of event triggers, and 3 for the identification of attributes, namely Class, Tense, and Aspect. Each classifier has then its own set of specific features:

- The Event Trigger classifier is extended with semantic roles, context windows of [+/-2] for token, lemmas, and POS.
- The Class classifier uses the typed dependency from an event token to the root token of the sentence, plus semantic roles.
- The Tense classifier is extended with context windows of [-2] tokens, and POS.
- The Aspect classifier employs (predicted) tense, a combination of tense and POS, and context windows of [-4] tokens, and POS.

We used the TempEval-2 test data as a development set to fine-tune the context windows and features. This allowed us to observe that:

- The quality of pre-processsing tools concerning lemmatization, POS tagging, and syntactic dependencies affects the quality of the Event Trigger classifier, especially for Recall.
- The best scores for Event Trigger and Class are obtained by using manually and automatically created training data. Although training with automatically generated data causes replication of errors, for these 2 subtasks, with high variability of realizations, it can positively affect the learning process by providing more positive examples for the sparse classes. Nevertheless, for the other attributes, which have less variability and are more dependent on specific combinations of co-textual data, automatically generated data lower the quality of the classifiers (e.g. the values PRESENT and PERFECTIVE for tense and aspect can only be correctly derived by identifying the relationship between an auxiliary and the main verb).
- Syntactic and semantic features have the biggest impact on the classifiers performance: using only morpho-syntactic and context window features gives an F1 of 82.1 for Event Trigger detection, which increases to 88.2 when adding lexical semantics features only, and reaches 90.9 when lexical semantics information is combined with syntactic information.

The best results of the CRF4TimeML system are obtained with the following configuration: [5]

- Pre-processing was performed with NewsReader NLP for semantic roles, and Stanford CoreNLP for tokenization, morpho-syntax, and dependency parsing.
- Adding the automatically generated training data to the manually created training data only for the Event Trigger and Class classifiers.
- Adding additional semantic information: VerbNet classes and FrameNet frames.

Table 3 provides the results of the CRF4TimeML system for Event Detection and Classification, as well as the results of the other TempEval-3 systems.

System	Trigger	Class	Tense	Aspect
CRF4TimeML	**81.87**	**72.24**	60.87	73.18
(Jung and Stent, 2013)	81.05	71.88	59.47	73.5
(Bethard, 2013)	78.81	67.87	**61.63**	71.61
(Chambers, 2013)	80.3	67.48	60.86	**73.28**
(Kolya et al., 2013)	78.62	52.69	58.62	72.14

Table 3: F1 system results on the TempEval-3 Event Detection and Classification Task (Event Trigger, Class, Tense, and Aspect).

CRF4TimeML obtains the best F1 scores for Event Trigger and Class, while the results for Tense and Aspect are slighlty lower than the scores of the best systems. According to the TempEval-3 ranking, CRF4TimeML would be first. However, differences in results are not statistically significant after performing the McNemar's test ($p>0.05$).

4.2. Temporal Relation and Classification

The TR task is addressed by means of 3 multi-class CRF classifiers, one for each pair of temporal entities (**e-dct**, **e-t**, and **e-e** pairs), which predict the 14 TimeML temporal values. Similarly to existing systems, we target only intra-sentence relations for **e-t** and **e-e** pairs, given that the number of cross-sentence relations in the training data is low. The classifiers are trained with the gold data set only, plus additional relations from Bethard et al. (2014). Different pairs have been normalized with respect to the directionality of the relation in order to reduce the variability of the temporal values. This resulted in the following ordering of pairs: i) relations involving an event and a timex, including the DCT, have been represented as **e-t/e-dct** pairs; ii) relations involving event pairs have been normalized according to the linear order of presentation of the events in the sentences.

In this task the system uses predicted event triggers, which in the learning model are represented with the morpho-syntactic and lexical-semantic features used in the Event Detection and Classification task plus the predicted values for class, tense, and aspect.

Each TP classifier uses additional specific features, namely:

- **e-dct**: TimeML type of the DCT (DATE or TIME).
- **e-t**: Additional features for each each timex: the TimeML type, token(s), dependency relation, head, lemma, POS, and a combination of the POS, dependency relation, and head POS. In addition to this, we have included: the dependency path connecting the event and the timex, the token(s) of any temporal signal between the event and the timex, the token of a

[5] Details for replicating the experiments and results are available at `https://github.com/cltl/TimeMLEventTrigger`

temporal signal at the beginning of the sentence, and the distance between the two elements in the pairs.

- **e-e**: Features about the connection of the two events (typed dependency paths; typed dependency paths and POS; typed dependency paths, tokens, and POS); a combination of the tense values of the two events; a combination of the tense and aspect values of the two events; the distance between the two events in the sentence; the token of a temporal signal before the first event in the pair; the token(s) of any temporal signal between the two events; presence of any other events between the elements in the pair; a combination of the events classes.

One the known problems of the TempEval-3 dataset is that the annotation of TRs is not complete because the temporal closure of the relations between all temporal entities is not available. Following the solution proposed in the TimeBank Dense corpus (Cassidy et al., 2014), we have assumed that in the test data the temporal closure of all events and timexes has been calculated, so that all events are temporally connected to each other. With this biased model we aim at better evaluating the completeness of the test data, by identifying pairs which are correct but not annotated.
Finally, we use the 14 temporal relation values, rather than simplifying the set to the most frequent ones in the training data only.
Table 4 contains the results of our TP system, compared to the results of the reviewed systems.

System	F1	P	R
CRF4TimeML	29.69	23.86	**39.29**
(Bethard, 2013)	**30.98**	**34.08**	28.40
(Chambers, 2013)	27.28	31.25	24.20
(Kolya et al., 2013)	24.61	19.17	34.36
(Kolomiyets and Moens, 2013)	19.01	17.94	20.22

Table 4: Results for TempEval-3 Task C (Temporal Processing from raw text).

Our system qualifies as the second best on this task (F1 29.69). The higher recall and lower precision is a consequence of assuming a full temporal connection of the entities. Systems that are designed based on this assumption tend to over-generate TRs. Breaking down these results per type of entity pairs (see also Table 5), our system has the best F1 for **e-e** pairs (25.51), while the best score for the other participating system is obtained by Chambers (2013) (F1 19.01). Results are different for **e-t** and **e-dct** pairs. In both cases Bethard (2013) obtains the best scores, with an F1 of 41.41 for **e-t** and 24.75 for **e-dct**. Our system, on the other hand, scores only 27.59 F1 for **e-t**, and a competitive 23.48 for **e-dct**. The different scores for the **e-t** pairs and the rest indicates that more (and better) annotated data are needed on specific pairs of temporal entities, namely **e-dct** and **e-e**.

5. Event Triggers: What is it wrong?

We analyzed the errors made by all systems presented in the previous section for the event detection and classification subtask. As for event detection, of the 749 gold events, 64% are correctly detected by all systems, 10% by 5, 3.6% by 4, 4% by 3, 4.4% by 2, 4.4% by 1, and 9.4% by 0 systems. From the events that all systems correctly detect 91.87% are verbs, 7.08% nouns, 0.83% adjectives, and 0.20% other. From the events that no system correctly detects 67.60% are nouns, 18.30% are adjectives, 11.26% other, and 2.81% prepositions. These numbers indicate that more systems agree for events with POS verb, than for events with POS noun and that events with POS noun are more difficult to detect.
The high number of verbs being correctly detected might reflect the annotation decisions stated in the TimeML annotation guidelines, which are strongly leaning towards the verb category (Pustejovsky et al., 2003). This impacts the results of the systems, which are well trained to detect events expressed by prototypical POS (i.e. verbs), while detecting events with less prototypical POS remains a challenge. This is coherent with statistics from the training data, where 80.5% of events are verbs. In this sense we are confronted with a very standard characteristic of NLP gold data sets, namely class imbalance. As in many other NLP tasks, a good system will have to be able to deal with the sparse examples that belong to the long tail of data distribution.
From the events with POS noun that no system detects, 3 are proper nouns affected by metonymy, as in the Example 1 where *Everest* is a proper noun that refers to the event 'climbing the Everest'. Solving this cases would require a system to apply inference mechanisms.

1. He said: "Lowe was a brilliant, kind fellow who never sought the limelight ... and 60 years on from **Everest** his achievements deserve wider recognition."

Additionally, for all events that no system detects correctly we checked if they occur in the training corpus. We found that out of 71, 4 occur less than 5 times and 2 around 40 times, but with a different POS. The rest do not occur in the training corpus. This raises the question of how reliable systems are when confronted to previously unseen data. It also confirms the well-known dependence on training data of discrete models and their limitations to generalize.
As for event classification, 43.95% of the examples are correctly classified by all systems, 22.12% by 5, 7.96% by 4, 5.75% by 3, 5.16% by 2, 6.19% by 1, and in 8.84% of the cases no system finds the right solution. The events that all systems correctly classify belong mostly to the classes OCCURRENCE (74.16%) and REPORTING (21.81%), which are the most frequent classes in the training set (61.71% and 14.36%). The distribution of classes where all systems fail is as follows: STATE (43.33%), ASPECTUAL (23.33%), I_STATE (10%), OCCURRENCE (8.33%), I_ACTION (6.66%), REPORTING (5%), and PERCEPTION (3.33%). This indicates again that the most difficult cases belong to low-represented classes in the training data.

6. Temporal Relations: When is it wrong?

For the error analysis of the TR subtask we look at three aspects: i) how many and what type of relations are incorrectly classified by all systems; ii) what type of processing

	[e-dct]			[e-t]			[e-e]		
System	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**
CRF4TimeML	21.95	13.66	**55.80**	27.71	20.72	41.81	**25.51**	22.40	**29.61**
(Bethard, 2013)	**24.75**	16.54	49.17	**41.41**	**38.46**	**44.84**	16.57	**35.62**	10.80
(Chambers, 2013)	19.72	**25.66**	16.02	40.65	37.18	**44.84**	19.01	25.52	15.15
(Kolya et al., 2013)	20.76	13.04	50.82	27.05	21.32	36.96	18.50	15.10	23.86
(Kolomiyets and Moens, 2013)	21.15	13.17	53.59	1.66	2.66	1.21	9.88	13.92	7.66

Table 5: Results for Temporal Relations Detection and Classification per type of TLINK.

requirements are needed to solve cases where all systems fail; and, iii) to which extent False Positives (FP) identified by our own system are correct.

Table 6 presents the errors of systems per type of TR. For a case to be considered correct, both the pair of temporal entities and the TR value have to be correct with respect to the gold standard data.

Correct	**[e-dct]**	**[e-t]**	**[e-e]**
All systems	16 (10.70%)	0 (0%)	2 (0.45%)
4 systems	45 (30.20%)	35 (30.07%)	9 (2.04%)
3 systems	31 (20.80%)	31 (27.19%)	28 (6.34%)
2 systems	12 (8.05%)	8 (7.01%)	79 (17.91%)
1 system	23 (15.43%)	18 (15.78%)	150 (34.01%)
No system	22 (14.76%)	22 (19.29%)	173 (39.22%)

Table 6: Number of correct solutions per number of systems and type of relation in the TLINK task.

The first observation is that there is no pair of **e-t** where all systems "agree" in providing a correct answer. A similar observation holds for the **e-e** pairs, although, in this case, only in 2 instances all systems provide a correct answer. These figures, especially for the **e-t** and **e-e** relations, clearly point out that both anchoring and ordering TRs are complex tasks when performed from raw text data.

Furthermore, the complexity of both tasks is increased, in this case, by divergences in how the training and test data have been annotated. [6] For instance, when looking at the percentage of events realized by nouns and verbs in the manually annotated TempEval-3 training data, we find that 43.78% of events are not temporally connected, while in the test data they amount to 20.11%. The lack of annotated TRs in the training data makes the two-step learning algorithm weak, since the success of TR classification depends on having found first the right pairs of entities which stand in a TR. This is also an additional motivation for having assumed, in this work, the temporal closure of **e-e** and **e-t** pairs.

With the exception of one system (Bethard, 2013) that uses a restricted set of temporal values, all system outputs are as skewed as in the training data. In particular, systems tend to predict the values BEFORE, AFTER, and SIMULTANEOUS for **e-e** pairs, IS_INCLUDED for **e-t** pairs, and BEFORE, INCLUDES, IS_INCLUDED, and AFTER for **e-dct** pairs.

Concerning **e-e** relations, we have analysed also the impact of tense and aspect values. According to tense and aspect semantics (Comrie, 1985), differences in tense forms between sequences of events are primary hints for correct TR identification and classification. We look at the cases where all systems fail, but we found that the percentage of errors in sequences with different tense and aspect values is similar to the percentage of errors in sequences of events with the same tense and aspect values (38.38% and 40.76%, respectively). This raises questions about choices in the TimeML Annotation Scheme concerning i) the granularity of tense and aspect attributes, and ii) their annotation methodology. For instance, having more fine-grained values for the Past temporal dimension, thus allowing to differentiate between a simple past and a past perfect, may positively affect the task. Furthermore, the TimeML surface-based annotation philosophy should not be applied to the tense and aspect values because rather than surface forms, what is actually needed is a contextual interpretation of these values.

We have analyzed the **e-e** pairs in terms of parts-of-speech. Apparently, TRs between events with different parts-of-speech is less prone to errors than pairs with the same parts-of-speech values (25.26% and 38.15%, respectively, when looking at the cases where all systems fail to correctly classify the TR).

As a result of a detailed error analysis, we provide a classification of errors into the 6 categories listed below.[7] Except for *Error*, the categories refer to the processing requirement that the system should have fulfilled in order to correctly identify and classify a TR.

- **Iconicity**: the system should interpret the linear order of presentation of the entities.
- **Signaled**: the system needs to process an explicit temporal signal (e.g. *before*, *since*, and similar) that connects the elements in the pair.
- **Inference**: the relation can be identified and classified through inference via other existing relations (e.g. two events linked to two different timexes can be ordered by means of the comparison of the values of timexes only).
- **Grammar**: the system needs to infer the relation via grammatical information (e.g. tense and aspect values), and/or syntactic dependencies between the elements in a pair.
- **World Knowledge**: the relation can be classified by applying knowledge about event semantics (including *aktionsaart*), discourse structure, factuality profiling of events, script and frame knowledge, and general world knowledge.

[6] On this point see also (Cassidy et al., 2014; Orasmaa and Kaalep, 2017)

[7] 4 classes: Iconicity, Inference, Signaled and World Knowledge have been proposed in Derczynski (2013).

- **Error**: the gold temporal relation is either wrong or in dispute.

We summarize our findings in Table 7. The following error analysis is based on all cases of **e-t** and **e-dct** pairs and is limited to 50% of the **e-e** pairs.

Processing requirement	**[e-dct]**	**[e-t]**	**[e-e]**
Iconicity	0 (0%)	0 (0%)	4 (4.59%)
Signaled	0 (0%)	9 (40.9%)	9 (10.34%)
Inference	6 (27.27%)	5 (22.72%)	21 (24.13%)
Grammar	4 (18.18%)	1 (4.5%)	20 (22.98%)
World Knowledge	11 (50%)	3 (13.63%)	32 (36.78%)
Error	1 (4.5%)	4 (18.18%)	1 (1.14%)

Table 7: TLINK error classification of errors for cases where all systems fail.

With the exception of the **e-dct** pairs, world knowledge has a less prominent role than expected, especially for **e-e** pairs. Previous studies (Van Der Meer et al., 2002; McRae and Matsuki, 2009; Caselli and Prodanof, 2009) have shown that event knowledge is highly salient and, actually, plays a preeminent role in the identification and classification of temporal relations by humans. Nevertheless, as our data show, the impact of world knowledge is limited to 36.78% of the cases for **e-e** pairs and only 13.63% for **e-t** ones. Theses figures seem to be in-line with other studies on error analysis for high-level semantic tasks. For instance, Clark et al. (2007) in their error analysis of the RTE-3 dataset show that world knowledge has an impact on the correct resolution of text/hypothesis pairs only for 36% of the cases. We report below some examples of temporal entity pairs which could be correctly solved only by taking into account world knowledge.

2. (**DCT:2013-03-21**) **Lodged** in his brain is an untreatable and inoperable cancerous tumor [...]. [WSJ_20130321_1145.tml] `TLINK e-dct:INCLUDES`

3. He won the Gusher Marathon, **finishing** in **3:07:35**$_{DURATION}$. [WSJ_20130321_1145.tml] `TLINK e-t:ENDS`

4. Over 200,000 Belfast customers were **affected** by a blackout but power is starting to be **restored**. [bbc_20130322_1600.tml] `TLINK e-e:INCLUDES`

An average of circa 24% for TRs between the three types of entity pairs could be addressed and (possibly) correctly solved if inference mechanisms were applied. Being able to keep track of which entities have been already temporally connected and, most importantly, with which values, may positively impact the resolution of the task. In example 5, the temporal relation between the event pair **fail**-**creates** can be solved if the TR between the event **last year** and the timex *fail* is also taken into account and not just, for instance, the tenses of the two events.

5. Greece may well have been too big to **fail** *last year*, but Cyprus, which **creates** less than one-half percent [...] [nyt_20130321_cyprus.tml] `TLINK e-e:BEFORE`

Grammatical information has an important role for **e-e** relations. The following cases could have been solved if grammatical structure (Example 6), and tense and aspect values (Example 7) were correctly processed and taken into account.

6. [...] it would consider **keeping** a tower, *if* the airport **convinces** the agency [...] [CNN_20130322_248.tml] `TLINK e-e:AFTER`

7. "We are **growing**$_{present:progressive}$ in number", **said**$_{past:none}$ Senator Amy Klobuchar [...] [nyt_20130321_women_senate.tml] `TLINK e-e:INCLUDES`

A subset of errors could have been avoided by having access to the contextual interpretation of the tense and aspect values, as illustrated in Example 7. For both events, we report the TimeML values of tense and aspect.

8. The season **started**$_{past:none}$ a month ago, **sparking**$_{present:none}$ concerns [...] [AP_20130322.tml] `TLINK e-e:SIMULTANEOUS`

Another cause of errors is incorrect processing of pairs where one of the elements is a reporting verb. These cases require careful processing of the linguistic context in order to identify the correct TR between the reported events and the reporting verbs. To correctly process event pairs that contain a reporting verb, it seems important to distinguish the type of reported speech, i.e. whether it is direct, indirect, or mixed. Reported speech tends to maintain an internal coherence and a specific temporal dimension which may differ with respect to the one signaled by the reporting verb and, in some cases, may also not fit the expected sequence of tenses.

9. [...] gale force wind would **blow** snow ...they **added**. [bbc_20130322_1600.tml] `TLINK e-e:AFTER`

10. "We **appear** to be getting close [...]" Jhung **added**. [AP_20130322.tml] `TLINK e-e:INCLUDES`

Additional errors are found in inter-sentential **e-e** relations. Overall, they represent 26.96% of the analyzed data. Interestingly, the correct processing of inter-sentential **e-e** pairs cannot be related to a particular processing requirement, since errors are distributed in the 6 categories reported in Table 7.

Focusing on the errors of our system, we have measured the impact of the semantic features by removing the lexical semantic features. On the test data, the scores drop 3.63 points of temporal awareness. [8] As for the False Positives (FP), our system produces 555 FP for **e-dct** relations; 301 for **e-t**, and 571 for **e-e**. We manually checked 15% of the

[8] The official scorer computes the temporal awareness including inferred temporal link, but it does not output them.

test files to establish whether the temporal links predicted by the system are correct. Out of 81 **e-dct** links, 74 of them are valid, which results in 48 (64.86%) links correctly classified. The same applies to the **e-e** pairs, where out of 54 system output links, 44 are valid with 26 (59.09%) correctly classified. As for **e-t** pairs, we have identified 40 possible links, with 33 valid links. Contrary to the other cases, only 9 (27.27%) **e-t** links are correct due to an over-generation of the IS_INCLUDED value.

7. Conclusions and Future Work

In this paper we have focused on TP in the framework of TempEval-3. We have reviewed the 5 top performing systems to gain insights into their architectures and features. We found that no system has used rich lexical semantic information as a means to encode world knowledge information. We developed a new end-to-end TP system that, by incorporating rich lexical semantic information, performs better than all systems for the Event Detection and Classification task, (F1-Class 72.24) and qualifies second on the TR Identification and Classification task (F1 29.69). Additionally, we performed an error analysis by comparing the output of all the systems focusing on the cases where no system can give a correct answer.

A detailed error analysis shows that there are easy and difficult cases both for event trigger detection and TR processing. Summing up, for event detection and classification problems arise when non-prototypical POS and polysemous lexical items are involved, while for TRs the difficulty lies in the creation of the pairs.

Concerning the classification errors of temporal relations, we observe that inference phenomena and world knowledge have a prominent role. As for inference, the analysis of data suggests that a two step strategy should be followed: first, provide a temporal anchor to the events by addressing first the **e-t** and **e-dct** pairs, and then use this information to enrich models to learn **e-e** pairs. Sieve-based architectures expanded with transitivity rules like the one proposed by Chambers et al. (2014) are addressing this problem in the right way but they require "densely" annotated data, or else the transitivity rules fail.

We showed that rich lexical semantic information is beneficial for the TP task but not enough. Recent work (Mirza and Tonelli, 2016; McDowell et al., 2017) has shown on a different dataset (TimeBank-Dense corpus) that word embeddings positively contribute[9] to the classification of TRs between **e-e** pairs both when occurring in the same sentence and in different sentences.

As a result of our error analysis, we would like to stress the following aspects as possible future directions. Firstly, we need more data, systematically annotated, in the line of the TimeBank-Dense corpus. At the same time, we think that it is important to densely annotate only those temporal entities (namely events) which are *actually* in a temporal relation, avoiding to introduce temporal chains with events which are not temporally connected. This would make the task more difficult but also more natural with respect to how humans reconstruct a document's timeline. Secondly, systems for TP should be able to "keep track" of the incremental nature of the document. So far, almost all approaches (also those that recently targeted the task via Neural Networks (Choubey and Huang, 2017)) have assumed that each sentence is a discourse universe of its own, while sentences occur in a text, which is a unitary and coherent message. Finally, the granularity of temporal relations should be reduced as most of the fine-grained values are very rare (and in some cases very hard to annotate), thus having a negative impact in the development of robust machine learning models.

[9] McDowell et al. (2017) report an increase of 5% in F1 in the TimeBank-Dense corpus

8. Acknowledgements

The work presented in this paper was funded by the Netherlands Organization for Scientific Research (NWO) via the Spinoza grant, awarded to Piek Vossen in the project "Understanding Language by Machines".

9. Bibliographical References

Agerri, R., Bermudez, J., and Rigau, G. (2014). Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *LREC*, volume 2014, pages 3823–3828.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *COLING-ACL '98: Proceedings of the Conference*, pages 86–90, Montreal, Canada.

Bethard, S., Chambers, N., McDowell, B., and Cassidy, T. (2014). An annotation framework for dense event ordering. In *52nd Annual Meeting of the Association for Computational Linguistics*, 52, Baltimore, MD, USA, June. Association for Computational Linguistics.

Bethard, S., Savova, G., Chen, W.-T., Derczynski, L., Pustejovsky, J., and Verhagen, M. (2016). Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, California, June. Association for Computational Linguistics.

Bethard, S., Savova, G., Palmer, M., and Pustejovsky, J. (2017). Semeval-2017 task 12: Clinical tempeval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572, Vancouver, Canada, August. Association for Computational Linguistics.

Bethard, S. (2013). Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 10–14.

Caselli, T. and Prodanof, I. (2009). Robust temporal processing: from model to system. *Research in Computing Science. Special Issue: Natural Language Processing and its Applications*, pages 29–40.

Caselli, T. and Sprugnoli, R. (2017). In *It-TimeML and the Ita-TimeBank: Language Specific Adaptations for Temporal Annotation*, pages 969–988. Springer.

Cassidy, T., McDowell, B., Chambers, N., and Bethard, S. (2014). An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume*

2: Short Papers), pages 501–506, Baltimore, Maryland, June. Association for Computational Linguistics.

Chambers, N., Cassidy, T., McDowell, B., and Bethard, S. (2014). Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284.

Chambers, N. (2013). NavyTime: Event and Time Ordering from Raw Text. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 73–77, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Choubey, P. K. and Huang, R. (2017). A sequential model for classifying temporal relations between intra-sentence events. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1803, Copenhagen, Denmark, September. Association for Computational Linguistics.

Clark, P., Murray, W. R., Thompson, J., Harrison, P., Hobbs, J., and Fellbaum, C. (2007). On the role of lexical and world knowledge in rte3. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 54–59. Association for Computational Linguistics.

Comrie, B. (1985). *Tense*, volume 17. Cambridge University Press.

Declerck, R. (1986). From Reichenbach (1947) to Comrie (1985) and beyond. *Lingua*, 70(4):305 – 364.

Derczynski, L. (2013). *Determining the Types of Temporal Relations in Discourse*. Ph.D. thesis, University of Sheffield.

Christiane Fellbaum, editor. (1998). *WordNet: an electronic lexical database*. MIT Press.

Ferro, L., Gerber, L., Mani, I., Sundheim, B., and Wilson, G., (2002). *Instruction Manual for the Annotation of Temporal Expressions*. MITRE, Washington C3 Center, McLean, Virginia.

Jung, H. and Stent, A. (2013). Att1: Temporal annotation using big windows and rich syntactic and semantic features. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 20–24, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Kolomiyets, O. and Moens, M.-F. (2013). Kul: Data-driven approach to temporal parsing of newswire articles. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 83–87, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Kolya, A. K., Kundu, A., Gupta, R., Ekbal, A., and Bandyopadhyay, S. (2013). JU_CSE: A CRF based approach to annotation of temporal expression, event and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 64–72. Citeseer.

Lacalle, M. L. D., Laparra, E., and Rigau, G. (2014). Predicate matrix: extending semlink through wordnet mappings. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Llorens, H., Chambers, N., UzZaman, N., Mostafazadeh, N., Allen, J., and Pustejovsky, J. (2015). Semeval-2015 task 5: Qa tempeval - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800, Denver, Colorado, June. Association for Computational Linguistics.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. System Demonstrations*, pages 55–60.

McDowell, B., Chambers, N., Ororbia II, A., and Reitter, D. (2017). Event ordering with a generalized model for sieve prediction ranking. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 843–853.

McRae, K. and Matsuki, K. (2009). People use their knowledge of common events to understand language, and do so as quickly as possible. *Language and linguistics compass*, 3(6):1417–1429.

Mirza, P. and Tonelli, S. (2014). An analysis of causality between events and its relation to temporal information. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2097–2106, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Mirza, P. and Tonelli, S. (2016). On the contribution of word embeddings to temporal relation classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2818–2828.

Orasmaa, S. and Kaalep, H.-J. (2017). Can we create a tool for general domain event analysis? In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 192–201.

Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003). The TimeBank corpus. 2003:40.

Reichenbach, H. (1947). *Elements of symbolyc logic*. the Free Press.

Saurı, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006). Timeml annotation guidelines version 1.2. 1.

Setzer, A. (2001). *Temporal information in newswire articles: an annotation scheme and a corpus study*. Ph.D. thesis, University of Sheffield.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events,

and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Van Der Meer, E., Beyer, R., Heinze, B., and Badel, I. (2002). Temporal order relations in language comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 28(4):770.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80. Association for Computational Linguistics.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010). Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62. Association for Computational Linguistics.

Vossen, P., Caselli, T., and Kontzopoulou, Y. (2015). Storylines for structuring massive streams of news. In *Proceedings of the First Workshop on Computing News Storylines*, pages 40–49, Beijing, China, July. Association for Computational Linguistics.

10. Language Resource References

Predicate Matrix Working Group. (2017). *Predicate Matrix*. http://adimen.si.ehu.es/web/PredicateMatrix, Unspecified, ISLRN 264-387-270-241-5.

Pustejovsky, J., Verhagen, M., Sauri, R., Littman, J., Gaizauskas, R., Katz, G., Mani, I., Knippen , R., and Setzer, A. (2014). *TimeBank 1.2*. Linguistic Data Consortium: Web Download, 1.0, ISLRN 717-712-373-266-4.

VV.AA. (2017a). *VerbNet*. unspecified, ISLRN 606-348-320-105-2.

VV.AA. (2017b). *WordNet*. unspecified, ISLRN 379-473-059-273-1.

Annotating Temporally-Anchored Spatial Knowledge by Leveraging Syntactic Dependencies

Alakananda Vempala and **Eduardo Blanco**
Human Intelligence and Language Technologies Lab
University of North Texas
AlakanandaVempala@my.unt.edu, eduardo.blanco@unt.edu

Abstract

This paper presents a two-step methodology to annotate temporally-anchored spatial knowledge on top of OntoNotes. We first generate potential knowledge using syntactic dependencies, and then crowdsource annotations to validate the potential knowledge. The resulting annotations indicate how long entities are or are not located somewhere, and temporally anchor this information. Crowdsourcing experiments show that spatial inferences are ubiquitous and intuitive, and experimental results show that they can be done automatically.

Keywords: Semantics, Information extraction, Spatial knowledge

1. Introduction

Extracting spatial meaning from text is of utmost importance in natural language understanding. Efforts focused on spatial meaning—both corpora development and automatic tools—have become popular. Existing approaches to extract spatial knowledge usually focus on extracting locations of events, someone or something. For example, semantic role labeling (Palmer et al., 2005) determines who did what to whom, when and where, e.g., *Thelma Gutierrez [went]*$_{\text{verb}}$ *[inside the forensic laboratory where scientist are trying to solve this mystery]*$_{\text{ARG}_4}$, where ARG$_4$ indicates the END POINT of event *went*. Efforts targetting locations of entities include geo-locating Twitter users (Liu and Inkpen, 2015), and pairing companies with the location of their headquarters (Mintz et al., 2009) e.g., *[IBM's]*$_{\text{company}}$ *headquarters in [New York]*$_{\text{location}}$.

Determining the temporal span where the spatial knowledge holds is not extensively researched. From the sentence *John parked Jamie's car at the Highland Garage*, we can infer that *John* and the *car* are certainly located at *the Highland Garage* minutes before and during parking, and that *John* will leave shortly after *parking* whereas the *car* will be at the garage for a few days but not months. We can also infer that *Jamie* will probably be at *the Highland Garage* at some point after *parking* to pick up his car.

This paper presents (1) a two-step methodology to extract temporally-anchored spatial knowledge by manipulating syntactic dependencies, and a (2) crowdsourced corpus annotated with temporally-anchored spatial knowledge.

The work presented here extends our previous work (Vempala and Blanco, 2016), which only manipulated semantic roles. We show that additional temporally-anchored spatial knowledge can be extracted by leveraging syntactic dependencies. We release a new corpus that annotates how long entities are and are *not* located somewhere, and temporally anchor this spatial information.[1]

2. Background

We work on top of OntoNotes (Hovy et al., 2006) as it is a well known corpus with text from various domains. Ontonotes contains over 64,000 sentences. It annotates, among other linguistic information, part-of-speech tags, parse trees, named entities and co-reference chains. We use the CoNLL- 2011 Shared Task distribution (Pradhan et al., 2011), and transform the gold parse trees into syntactic dependencies using Stanford CoreNLP (Manning et al., 2014). De Marneffe and Manning (2008) present and exemplify the Stanford dependencies, and Weischedel and Brunstein (2005) the named entity types used in OntoNotes.

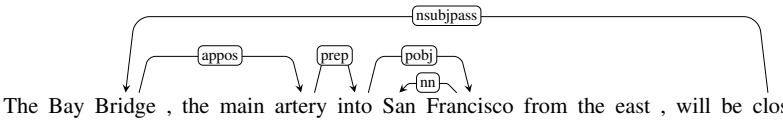
We use the term *temporally-anchored spatial knowledge* to refer to information regarding whether a given x is or is not located at some location y, and for how long with respect to an event. We use the notation LOCATION(x, y) to indicate the spatial relation between x and y. We use the term potential spatial knowledge to refer to spatial relations LOCATION(x, y) that are yet to be validated.

There are 2 types of relations LOCATION(x, y): (1) those whose arguments x and y are semantic roles of some verb, and (2) those whose arguments x and y are not semantic roles of any verb. Type (1) can be further divided into type (1a) if x and y are roles of the same verb, and type (1b) if x and y are roles of different verbs. In the sentence *John called Google's office for Bill's appointment*, the relation LOCATION(*John*, *Google's office*) is of type (1) and LOCATION(*Bill*, *Google's office*) is of type (2). Also, in the example *Officer Jack found the missing diamond at a warehouse owned by Mr. Walker*, LOCATION(*Jack*, *warehouse*) is of type (1a) and LOCATION(*Mr. Walker*, *warehouse*) is of the type (1b). Our previous work (Vempala and Blanco, 2016) extracts spatial knowledge of type (1). In Section 3, we detail the approach that leverages syntactic dependencies to extract spatial knowledge of type (1) and (2).

3. Corpus Creation

We follow a two-step methodology to annotate temporally-anchored spatial knowledge on top of OntoNotes. First, we manipulate syntactic dependencies and named entities to generate potential spatial knowledge. Second, we gather crowdsourced annotations to either discard or validate the potential knowledge.

[1] Available at https://alakanandav.bitbucket.io/



x	*y*	y_{verb}
The Bay Bridge	San Francisco	closed

Figure 1: Sample sentence and potential spatial knowledge generated using syntactic dependencies.

3.1. Generating Potential Additional Spatial Knowledge

We enforce the restrictions below to generate potential spatial relations LOCATION(*x*, *y*):

1. *y* is a GPE or LOC named entity;
2. *x* is a PERSON, FAC, PRODUCT or WORK_OF_ART named entity from the same sentence than *y*;
3. *y* is reachable from y_{verb}, where y_{verb} is the closest verb going up the dependency tree from *y*; and
4. y_{verb} is not the verb *to be* or *to have*.

We defined Restrictions 1–2 because we are interested in locations of named entities and assigning spatial information to other entities (e.g., DATE) is nonsensical. Restriction 3 reduces the annotation effort. Restriction 4, however, was designed after pilot annotations revealed that no temporally-anchored spatial knowledge could be inferred from them.

OntoNotes annotates 19,478 GPEs and 1,858 LOCs (potential *y*s); and 18,823 PERSONs, 1,080 FACs, 734 PRODUCTs, and 1,253 WORK_OF_ARTs (potential *x*s). Pairing all potential *x*s and *y*s within a sentence results in 10,136 pairs (Restrictions 1–2). Enforcing Restriction 3 reduces the number of pairs to 9,351, and enforcing Restriction 4 further reduces the number to 8,775. Out of these 8,775 pairs, 7,029 have a PERSON as *x*, 951 a FAC, 411 a PRODUCT, and 384 a WORK_OF_ART.

Figure 1 presents sample sentence and potential spatial relations generated using dependencies.

3.2. Crowdsourcing Annotations

After generating potential spatial knowledge, it must be validated manually. To do so, we crowdsourced annotations using CrowdFlower and asking questions in plain English. More specifically, for each potential pair (*x*, *y*), annotators were asked "After reading the sentence above, is *x* located at *y* before / during / after y_{verb}?" The annotation interface showed the original sentence with *x* and *y* highlighted, and no further information. Annotators were instructed to answer questions based on the sentence provided, and to not use prior knowledge about *x* or *y*.

After pilot annotations, it became clear that answering the above question with `yes` or `no` is suboptimal. First, the question is sometimes nonsensical because (a) *x* cannot be literally located anywhere, or (b) y_{verb} is a state, thus the meaning of before / during / after y_{verb} is unclear.

Second, sometimes there is not enough information in the sentence to unambiguously determine whether *x* is or is not located at *y* with respect to y_{verb}. Recall that potential pairs are generated automatically, so some will inevitably be spurious. The final interface forces annotators to choose from one of the following coarse-grained labels for each temporal anchor (before / during / after y_{verb}):

- `yes` or `no` if *x* is (or is *not*) located at *y* before / during / after y_{verb};
- `inv` if asking the question for *x* is nonsensical; and
- `unk` if the question is intelligible but the answer is unknown, i.e., neither `yes` nor `no`.

Additionally, if the coarse-grained label is `yes`, annotators had to choose a fine-grained label:

- Before and after: `secs`, `mins`, `days`, `weeks`, `months`, `years`, or `inf` for infinite. They were instructed to choose the longest unit of time possible (e.g., `days` means for a few days but less than a week).
- During: `entire` if *x* is located at *y* for the entire duration of y_{verb}, `some` otherwise.

Out of the 8,775 (*x*, *y*) pairs automatically generated, we collected three annotations for 25% of pairs per y_{verb} (total: 1,689). Among these relations, 478 belong to type (1) (a: 227, b: 251) and 1,211 belong to type (2) i.e., *x* and *y* belong to the same semantic role or are not the heads of any semantic role.

4. Corpus Analysis

Figure 2 shows percentages of coarse-grained labels per temporal anchor (before, during, after and all). Overall (bottom right sub figure), only 3.20% questions are invalid, and annotators answered with `yes` or `no` 74.28% of questions (`yes`: 51.77%, `no`: 22.51%), i.e., almost 75% of potential spatial knowledge is deemed correct by annotators. Percentages per named entity type of *x* follow similar trends overall, but WORK_OF_ART has more `inv` labels (18.05%) than the rest (0.85%–2.87%), and PRODUCT has more `yes` labels than the rest (62.65% vs. 45.83%–53.23%). The percentages per temporal anchor indicate that more temporally-anchored spatial knowledge can be extracted for *before* than *after* (52.87% + 28.89% = 81.76% vs. 52.34 + 22.68% = 75.02%). Also, more potential spatial knowledge can be extracted for *during* than *before* and *after* (63.47% + 21.79% = 85.26%).

Percentages for fine-grained labels are shown in Table 1. For *during*, the vast majority of labels (91.22%) are `entire`, and only 8.77% are `some`. For *before* and *after*, most labels are either `years` (49.28% and 36.72%) or `inf` (34.42% and 45.19%). Other labels (`secs`, `mins`, ..., `months`) are uncommon (1.10%–10.55%). Because of the unbalanced distribution, we experiment with clustered fine-grained labels $<$`years` and $\geq$`years`.

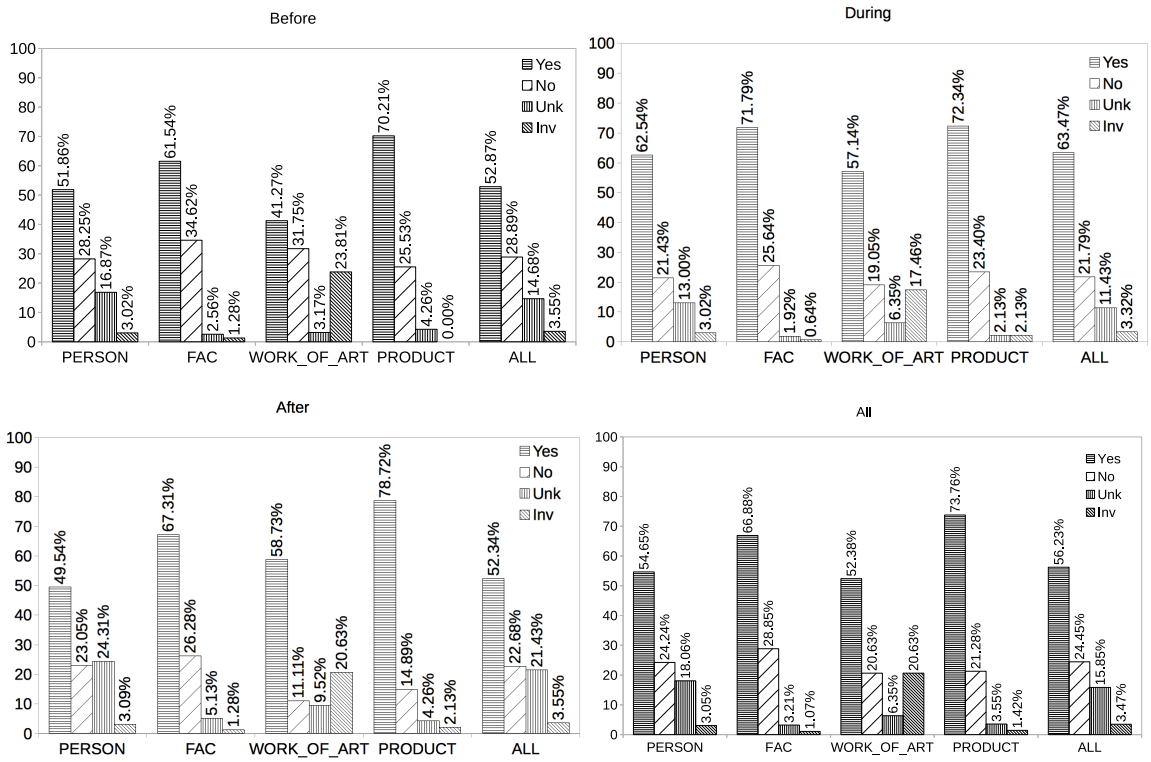


Figure 2: Percentages of coarse-grained labels per temporal anchor (top left: before, top right: during, bottom left: after, and bottom right: all). Percentages are divided by the named entity type of *x*.

	some	entire	secs	mins	hours	days	weeks	months	years	inf
Before	n/a	n/a	2.48	1.83	6.39	2.09	2.09	1.43	49.28	34.42
During	8.77	91.22	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
After	n/a	n/a	1.30	2.34	10.55	2.73	1.69	2.47	36.72	42.19
All	3.65	37.94	1.10	1.22	4.95	1.41	1.10	1.13	25.11	22.37

Table 1: Percentage of fine-grained labels for instances annotated with coarse-grained label `yes`.

	r	% instances such that		
		3/3	2/3	0/3
Before	0.62	31.20	56.42	12.37
During	0.63	36.64	51.56	11.78
After	0.59	26.34	60.86	12.78
All	0.59	31.39	56.28	12.31

Table 2: Annotation quality for coarse-grained labels. We show weighted Pearson correlations(r) between annotators and the majority label, and percentage of pairs (x, y) for which 3, 2 and none of the annotators agree (out of 3).

	r	% instances such that				
		3/3	2/3	0/3	2/2	0/2
Before	0.62	39.43	46.69	13.88	88.53	11.47
During	0.60	38.57	48.62	12.81	84.94	15.06
After	0.58	29.67	54.64	15.68	87.78	12.22
All	0.62	36.19	49.8	14.01	87.04	12.96

Table 3: Annotation quality for fine-grained labels. We show weighted Pearson correlations(r) between annotators and the majority label and percentages of pairs (x, y) for which annotators agree. We divide the percentages into pairs in which 3 or 2 annotators agreed on the coarse-grained label *yes* (3/3, 2/3 or 0/3; and 2/2 or 0/2).

4.1. Annotation Quality

A majority coarse-grained label exists in over 87% of pairs (Table 2, 3/3 or 2/3 agreed). We calculated weighted Pearson correlation between annotators and the majority label following this mapping: `yes`:1, `no`:-1, `unk` and `inv`:0; correlations range between 0.59 and 0.62. Fleiss Kappa agreements (not shown in Table 2) range between 0.51 and 0.55, which are considered moderate (Landis and Koch, 1977). We believe Pearson is better suited than Kappa, as not all disagreement are equally bad (e.g., `yes` vs. `no` and `unk` vs. `inv`).

Table 3 presents a similar analysis for fine-grained labels. A majority label exists in 86% of pairs when 3/3 annotators agreed on the coarse-grained label `yes`, and 87.04% when 2/3 annotators agreed. Pearson correlations range from 0.58 to 0.60, and overall Kappa is 0.56 (not shown).

4.2. Annotation Examples

In table 4, we present real examples from the annotated corpus. In Sentence 1, the annotators chose the label `years`

Sentence	Before		During		After	
	C	F	C	F	C	F
Sentence 1: [A {US}$^{\text{GPE}}$ poll]$_{\text{ARG}_0}$ [shows]$_{verb}$ [President {Clinton}$^{\text{PERSON}}$ and his wife, {First}$^{\text{ORDINAL}}$ Lady {Hillary Rodham Clinton}$^{\text{PERSON}}$ are the man and woman most admired by {Americans}$^{\text{NORP}}$]$_{\text{ARG}_1}$.						
x: Hillary Rodham Clinton, y: US, y_{verb}: *shows*	yes	years	yes	entire	yes	years
x: President Clinton, y: US, y_{verb}: *shows*	yes	years	yes	entire	yes	years
Sentence 2: [{Venezuela}$^{\text{GPE}}$'s leftist President]$_{\text{ARG}_0}$ has [awarded]$_{verb}$ [{Fidel Castro}$^{\text{PERSON}}$]$_{\text{ARG}_2}$ [the key to the city of {Caracas}$^{\text{GPE}}$]$_{\text{ARG}_1}$.						
x: Fidel Castro, y: Caracas, y_{verb}: awarded	yes	days	yes	entire	yes	days
x: Fidel Castro, y: Venezuela, y_{verb}: awarded	no	n/a	no	n/a	no	n/a
Sentence 3: On {Capitol Hill}$^{\text{LOC}}$ {today}$^{\text{DATE}}$, senators were asking [the general who]$_{\text{ARG}_0}$ [sent]$_{verb}$ [{the "USS Cole"}$^{\text{WORK_OF_ART}}$]$_{\text{ARG}_1}$ [into the port of {Aden}$^{\text{GPE}}$ in {Yemen}$^{\text{GPE}}$]$_{\text{ARG}_2}$, why he made that decision.						
x: "USS Cole", y: Yemen, y_{verb}: sent	no	n/a	no	n/a	yes	weeks
x: "USS Cole", y: Aden, y_{verb}: sent	no	n/a	no	n/a	yes	weeks
x: "USS Cole", y: Capitol Hill, y_{verb}: asking	no	n/a	no	n/a	no	n/a
Sentence 4: [{Yesterday}$^{\text{DATE}}$]$_{\text{ARGM-TMP}}$, [{Afghanistan}$^{\text{GPE}}$'s ruling {Taliban}$^{\text{ORG}}$]$_{\text{ARG}_0}$ [denied]$_{verb}$ [{Bin Laden's}$^{\text{PERSON}}$ involvement in the {Yemeni}$^{\text{NORP}}$ attack]$_{\text{ARG}_1}$.						
x: Bin Laden, y: Afghanistan, y_{verb}: denied	unk	n/a	unk	n/a	unk	n/a

Table 4: Annotation examples for the generated pairs. We show coarse- and fine-grained annotations (C and F respectively); y_{verb} denotes the first verb going up the dependency tree from y, curly brackets and superindices indicate named entities, and square brackets and subindices indicate semantic roles of y_{verb}

for before, after and `entire` for during because *Hillary Rodham Clinton* (x in pair 1) and President *Clinton* (x in pair 2) will be located in *US* (y) with respect to the y_{verb} *shows* for all the three temporal anchors. The label `years` can also be justified because the sentence states that *Hillary Rodham Clinton* is the wife of *US President Clinton*. Also, this is a type 2 spatial relation since x and y are not semantic roles of any verb.

From Sentence 2, we can say that *Fidel Castro* (x) is located in the *Caracas* city at least for a few `days` before, after and for the `entire` time during y_{verb} *awarded* took place. Also, the annotators correctly interpreted that *Fidel Castro* is not located in *Venezuela* before, during and after *awarded* took place.

From Sentence 3, the annotators could infer that the *USS Cole* (x) is not located in *Yemen* (y in pair 1) or *Aden* (y in pair 2) before and during y_{verb} *sent* took place. They also inferred that it will be located in *Yemen* and *Aden* at least for a few `weeks` after *sent* took place. Also, it can be justified that *USS Cole* is not present in *Capitol Hill* before, during and after y_{verb} *asking* took place.

In Sentence 4, we cannot say anything about the location of *Bin Laden* (x) from the sentence, so the annotators choose the label `unk` for all the three temporal anchors.

5. Experiments

In this section, we present learning experiments with the corpus. Each LOCATION(x, y) relation has three labels corresponding to temporal anchors before, during and after, thus we generate 3 instances per relation. We perform two classification tasks: (1) coarse-grained classification to predict `yes`, `no` or `unk`, and (2) clustered fine-grained classification to predict $\geq$ `years`, $<$`years`, `no` or `unk`. This classification is inspired by the previous work by Pan et al. (2006) who predict event durations.

We discard all the instances with `inv` label. We found it advantageous to train one classifier per temporal tag. We divide the instances into 80% train and 20% test by ensuring all the LOCATION(x, y) relations generated from a particular sentence belong to either the train or test sets. We use scikit-learn (Pedregosa et al., 2011) to train one SVM per temporal anchor and tune the C and γ parameters using 10-fold cross-validation and grid search over the train set. Results are reported on the corresponding test set.

Table 5 lists the feature set we experiment with. *Basic* features encode basic information regarding argument x, location y, x_$verb$ and y_$verb$. NE features categorize the argument x and location y based on their named entity types. Syntax features capture dependency structure of x and y. Semantic features add information regarding spatial and temporal roles.

6. Results

We performed experiments using gold-standard linguistic annotations as well as predicted linguistic annotations. The gold POS tags, parse-trees, semantic roles, dependencies and named entities are taken from the CoNLL release and the predicted linguistic information is obtained using SyntaxNet (Andor et al., 2016). The baseline systems predict the most frequent label per temporal anchor and obtain an overall F-score of 0.46 (0.29 for before, 0.49 for during and 0.29 for after).

Results with coarse-grained and clustered fine-grained labels obtained with all features per temporal anchor using gold standard linguistic information are presented in Table 6. Models trained with all features perform best with respect to all temporal anchors. In general, results with before and during are better than results with after.

Results with coarse-grained and clustered fine-grained labels obtained with all features per temporal anchor using predicted linguistic information is presented in Table 7. The

Type	Feature	Description
Basic	1	whether *x* occurs *before* or *after* *y*
	2	number of tokens between *x* and *y*
	3–4	number of tokens in *x* and *y*
	5–8	words and POS tags of the heads of *x* and *y*
	9–16	words and POS tags of the first and last tokens in *x* and *y*
	17–24	words and POS tags of the previous and next tokens to *x* and *y*
	25–26	word and POS tag of the closest verb to y going up the dependency tree
NE	27–28	named entity types of x and y
Syntax	29–30	outgoing dependencies from heads of x and y
	31–34	outgoing dependencies from first and last tokens of x and y
	35–38	outgoing dependencies from previous and next tokens to x and
Semantic	39–40	counts of ARGM-LOC and ARGM-TMP roles in the sentence

Table 5: Feature set to determine whether *x* is (or is not) located at *y*, and for how long.

		Before			During			After			Overall		
		P	R	F	P	R	F	P	R	F	P	R	F
Coarse-grained	Yes	0.62	0.71	0.66	0.66	0.84	0.74	0.58	0.63	0.61	0.62	0.73	0.67
	No	0.55	0.52	0.54	0.33	0.21	0.26	0.46	0.47	0.47	0.45	0.40	0.42
	Unk	0.32	0.23	0.27	0.35	0.12	0.18	0.38	0.32	0.35	0.35	0.22	0.27
	Avg	0.54	0.56	0.55	0.54	0.60	0.56	0.50	0.51	0.50	0.53	0.56	0.54
Clustered fine-grained	≥years	0.56	0.64	0.60	n/a	n/a	n/a	0.60	0.57	0.58	0.58	0.61	0.59
	<years	0.44	0.15	0.22	n/a	n/a	n/a	0.33	0.12	0.18	0.39	0.14	0.20
	entire	n/a	n/a	n/a	0.67	0.82	0.74	n/a	n/a	n/a	0.67	0.82	0.74
	some	n/a	n/a	n/a	0.00	0.00	0.00	n/a	n/a	n/a	0.00	0.00	0.00
	No	0.55	0.60	0.58	0.30	0.28	0.29	0.48	0.66	0.55	0.44	0.51	0.47
	Unk	0.33	0.28	0.31	0.31	0.10	0.16	0.44	0.40	0.42	0.36	0.26	0.30
	Avg.	0.51	0.52	0.51	0.51	0.57	0.52	0.50	0.51	0.50	0.51	0.53	0.51

Table 6: Precision recall and F1-score with coarse-grained labels and clustered fine-grained labels using features extracted from gold standard linguistic annotations for the best system per temporal anchor

		Before			During			After			Overall		
		P	R	F	P	R	F	P	R	F	P	R	F
Coarse-grained	Yes	0.57	0.83	0.67	0.67	0.88	0.76	0.56	0.56	0.56	0.60	0.76	0.66
	No	0.52	0.44	0.47	0.32	0.17	0.22	0.41	0.45	0.43	0.42	0.35	0.37
	Unk	0.00	0.00	0.00	0.67	0.08	0.14	0.24	0.21	0.23	0.30	0.10	0.12
	Avg.	0.45	0.55	0.49	0.58	0.62	0.56	0.44	0.44	0.44	0.49	0.54	0.50
Clustered fine-grained	≥years	0.57	0.57	0.57	n/a	n/a	n/a	0.60	0.39	0.47	0.59	0.48	0.52
	<years	0.43	0.18	0.25	n/a	n/a	n/a	0.25	0.17	0.20	0.34	0.18	0.23
	entire	n/a	n/a	n/a	0.64	0.88	0.74	n/a	n/a	n/a	0.67	0.82	0.74
	some	n/a	n/a	n/a	0.00	0.00	0.00	n/a	n/a	n/a	0.00	0.00	0.00
	No	0.52	0.61	0.56	0.33	0.17	0.22	0.40	0.62	0.49	0.42	0.47	0.42
	Unk	0.24	0.22	0.23	0.38	0.12	0.18	0.32	0.36	0.34	0.31	0.23	0.25
	Avg.	0.49	0.50	0.49	0.51	0.59	0.52	0.50	0.51	0.50	0.50	0.53	0.50

Table 7: Precision recall and F1-score values for best systems with coarse-grained labels and clustered fine-grained labels using features extracted from predicted linguistic annotations.

coarse-grained and clustered fine-grained results with models trained using predicted linguistic information obtained an overall F1-score of 0.50 (vs. 0.54 and 0.51) with a test set of 655 (vs. 1076, 60% overlap with gold test set). When semantic roles are used extract potential spatial knowledge (Vempala and Blanco, 2016) the overlap between predicted and gold test set is only 30%.

7. Conclusions

We have presented an approach to determine whether selected named entities are located or *not* located somewhere, and specify *when* with respect to an event. Crowdsourcing experiments show that annotating this kind of temporally-anchored spatial knowledge can be done by non-experts. Most of the pairs (74.28%, Figure 2) automatically generated are validated by annotators (coarse-grained labels `yes` and `no`). Importantly, working with named entities and syntactic dependencies instead of semantic roles allows us to generate more potential spatial knowledge and obtain better results in a realistic scenario, i.e., with predicted linguistic annotations.

8. Bibliographical References

Andor, D., Alberti, C., Weiss, D., Severyn, A., Presta, A., Ganchev, K., Petrov, S., and Collins, M. (2016). Glob-

ally normalized transition-based neural networks. *CoRR*, abs/1603.06042.

De Marneffe, M.-C. and Manning, C. D. (2008). Stanford typed dependencies manual. Technical report, Technical report, Stanford University.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: the 90% Solution. In *NAACL '06: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers on XX*, pages 57–60, Morristown, NJ, USA. Association for Computational Linguistics.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1).

Liu, J. and Inkpen, D. (2015). Estimating user location in social media with stacked denoising auto-encoders. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 201–210, Denver, Colorado, June. Association for Computational Linguistics.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore, August. Association for Computational Linguistics.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Pan, F., Mulkar, R., and Hobbs, J. R. (2006). An annotated corpus of typical durations of events. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 77–82. Citeseer.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27, Portland, Oregon, USA, June. Association for Computational Linguistics.

Vempala, A. and Blanco, E. (2016). Beyond plain spatial knowledge: Determining where entities are and are not located, and for how long. In *ACL (1)*.

Weischedel, R. and Brunstein, A. (2005). BBN pronoun coreference and entity type corpus. Technical report, Linguistic Data Consortium, Philadelphia.

Contextualized Usage-Based Material Selection

Dirk De Hertog, Piet Desmet
ITEC, KU Leuven, imec
Etienne Sabbelaan 51, 8500 Kortrijk, Belgium
{dirk.dehertog, piet.desmet}@kuleuven.be

Abstract

In this paper, we combine several NLP-functionalities to organize examples drawn from corpora. The application's primary target audience are language learners. Currently, authentic linguistic examples for a given keyword search are often organized alphabetically according to context. From this, it is not always clear which contextual regularities actually exist on a syntactic, collocational and semantic level. Showing information at different levels of abstraction will help with the discovery of linguistic regularities and thus improve linguistic understanding. Practically this translates in a system that groups retrieved results on syntactic grounds, after which the examples are further organized at the hand of semantic similarity within certain phrasal slots. Visualization algorithms are then used to show focused information in phrasal slots, laying bare semantic restrictions within the construction.

Keywords: Language Learning, Constructions, Corpus-based examples, Distributional Analysis

1. Introduction

We present the integration of NLP-functionalities to help with the selection of contextualized usage-based examples to assist language learners and other end users that could benefit from a distributional linguistic analysis. Usage-based material, in the form of examples drawn from corpora, like the Keyword in Context (KWIC) method, are thought of as valuable resources for studying languages. KWIC allows for the user to search for a keyword, retrieves a set of examples from a corpus and presents them ordered alphabetically according to context words. However, from examples alone, it is not always immediately clear which contextual regularities exist at the syntactic, collocational and semantic level. The lack of such additional structure and the required effort to derive it single-handedly have proven demotivational in practical settings.

We propose to use several NLP-methods to provide additional information by which the examples could be further organized in a meaningful way. This would add value as it would clarify linguistic regularities at a glance. Practically this translates in a system that looks for a certain keyword and groups the retrieved examples first according to syntactic grounds, then to semantic similarity within certain contextual phrasal slots.

By doing so, we aim to lessen the gap between corpus-based examples and cognitive understanding of linguistic regularities with the end-user. We start from the basic premise that learning presumes understanding (Krashen, 1981; Chapelle, 1996); in this respect, we argue that full disambiguation facilitates linguistic understanding and thus enhances word knowledge and improves vocabulary retention. The procedure is inspired by collostructional analysis (Anatol and Gries, 2003), which treats syntactic constructions as a disambiguating factor. For the semantic understanding of words, word sense is arguably linked to the actual construction in which it is used. Conversely, the intended word sense presupposes correct syntactic and/or collocational use for it to be understood. A retrieval that links both types of information should therefore prove highly informative.

We first explain what we interpret as cognitive understanding from a pedagogical and linguistic point of view. We then offer technical operationalizations to achieve the intended semantic disambiguation. Section 4 discusses expert opinions and proposes a future validation of the results. We wrap up with a short summary and point to ongoing and future work.

2. Towards Linguistic Understanding

Traditional teaching practice often distinguishes between acquisition of vocabulary and syntax. Syntax governs the grammatical rules found in a language, while vocabulary items fill certain syntactic slots following those grammatical rules.

Such a simple dichotomy is being challenged from different viewpoints. For example, within the field of Second Language Acquisition it is accepted that word knowledge is more complex than a simple slot-filler approach and also involves knowledge about lexical collocations, constructions and semantics (Nation, 2001). Within Cognitive Linguistics, the dichotomy is reinterpreted as a cline that ranges from the lexical to the syntactic, with semantic meaning and functional interpretation as a central linguistic principle (Croft, 2001).

From a linguistic point of view, Cognitive Linguistics defends a usage-based model of language. In such a model, all linguistic knowledge flows from the actual linguistic utterances a person encounters. It posits a mechanism of distributional interpretation of language into patterns (Tomasello, 2003), after which the patterns acquire a certain semantic interpretation themselves. Bybee (2006), for instance, exemplifies a mechanism of exemplar-based categorization of lexical items to result in grammatical patterns. The exemplars in their turn relate to the pattern as specific instantiations, called constructs. "The major idea behind exemplar theory is that the matching process has an effect on the representations themselves; new tokens of experience are not decoded and then discarded, but rather they impact memory representations." (p.716)

We aim to stimulate a similar process of linguistic understanding, acquired automatically by native speakers in second language learners, stimulating a distributional

understanding of an unknown word in a fully specified linguistic context (Nagy, Herman and Anderson, 1985; Sternberg, 1987). Moreover, it is widely accepted in Second Language Acquisition studies that exposure to new material stimulates vocabulary knowledge (Nation, 2001). Taking into account that the procedure requires a certain level of proficiency it seems best suited for advanced students that wish to expand on their existing vocabulary knowledge.

3. Selection Procedure

We emulate a full contextual disambiguation at the hand of two processing steps. The first uses the (abstract) constructions in which a word occurs and interprets it as a structural context. The second looks within that structure for analogue lexical examples that are semantically related. The double selection procedure thus highlights semantic constraints, shown through positive evidence in the form of examples, within a single construction. The proposed method has been applied to a compiled corpus of freely available literary works of approximately 100 million words in order to exemplify the procedure.

3.1. Constructional Constraints

In order to help the language learner discover constructional analogies (syntactic regularities) from examples, we propose to provide language input and enrich it with abstract constructional information. To derive abstract constructional information, we start from POS-tags, a word's syntactic category such as Noun or Verb, as formal superficial word-representations and derive contiguous patterns. Formally, a selected sequence of POS-tags thus stands for a construction, while the lexical word combinations matching that pattern are exemplar-based instantiations.

The selection of patterns is based on an actual example that poses difficulties for the learner. The retrieved examples from the corpus are matched to the syntagmatic axis of the problematic example on two levels: the target word (as a lexical center) and the syntactic constructions (constructional patterns) in which the lexical item functions.

Take for instance Ex. 1, in which *obvious* is set as the target word for which we want more distributional information.

Ex. 1. He made a rather **obvious** remark.

Pron Verb Det Adv Adj **Noun**

To get a better understanding of the word, we use corpus-based examples that share the keyword *obvious* and the constructions in which it participates. As such, rather than taking context-words such as *rather* and *remark*, as lexical collocations, we use them to determine the partly abstract constructions *Adv obvious* and *obvious Noun*. The corpus is then used to retrieve paradigmatically analogous constructs, resolving the abstract slot, as seen in Table 1.

The target word participates in a number of constructions that are increasingly complex. Using the number of different lexical types (as opposed to tokens) for a certain construction is a direct quantification of how prominent a particular structural context is for the word at hand. It is used for our purposes as an objective measure to quantify which constructional slots are most interesting to provide semantic structure for. From a practical perspective, those slots largely coincide with immediate dependencies of the target word. Ex. 2 shows a single construction, for which the numbers in brackets signify how many different word-types occur in the slot.

Ex. 2. Det (3) Adv (6) OBVIOUS Noun (14)

	the	perfectly	fact
	the	apparently	consideration
	an	equally	conviction

Lexical Example	Derived Constructions	Analogous Constructs	Lexical Types
Obvious remark	Obvious Noun	Obvious point Obvious reason …	290
Rather obvious	Adv obvious	Fairly obvious Very obvious …	58
Rather obvious remark	Adv obvious Noun	Fairly obvious fact Very obvious reason …	28

Table 1: Lexical Examples, Derived Constructions, Analogous Constructs and Lexical Types (output Step 1)

3.2. Semantic Similarity

To help the learner discover semantic structure in the retrieved constructions, we order the words according to their semantic similarity with the initial example. Because we see semantic restrictions as being related to certain constructional patterns, we aim to highlight the imposed paradigmatic restrictions with the given example in an unsupervised fashion.

We use a self-implemented version of Semantic Vector Spaces (SVS) (Lund and Burgess, 1996; Turney and Pantel, 2010; Mikolov, 2013) to order words according to certain criteria. A vector space model is defined by three distinct parameters: the contexts included as features, a weighting function for the collocational corpus counts, and a similarity metric to compare the vectors. Each parameter has an impact on the resulting measure of similarity. For instance, choices for the first parameter include the inclusion/exclusion of function words as context features and the window size that states how many context words surrounding the target are taken into account. Small context sizes that include function words will directly shift the vector space to capture a more syntactic interpretation,

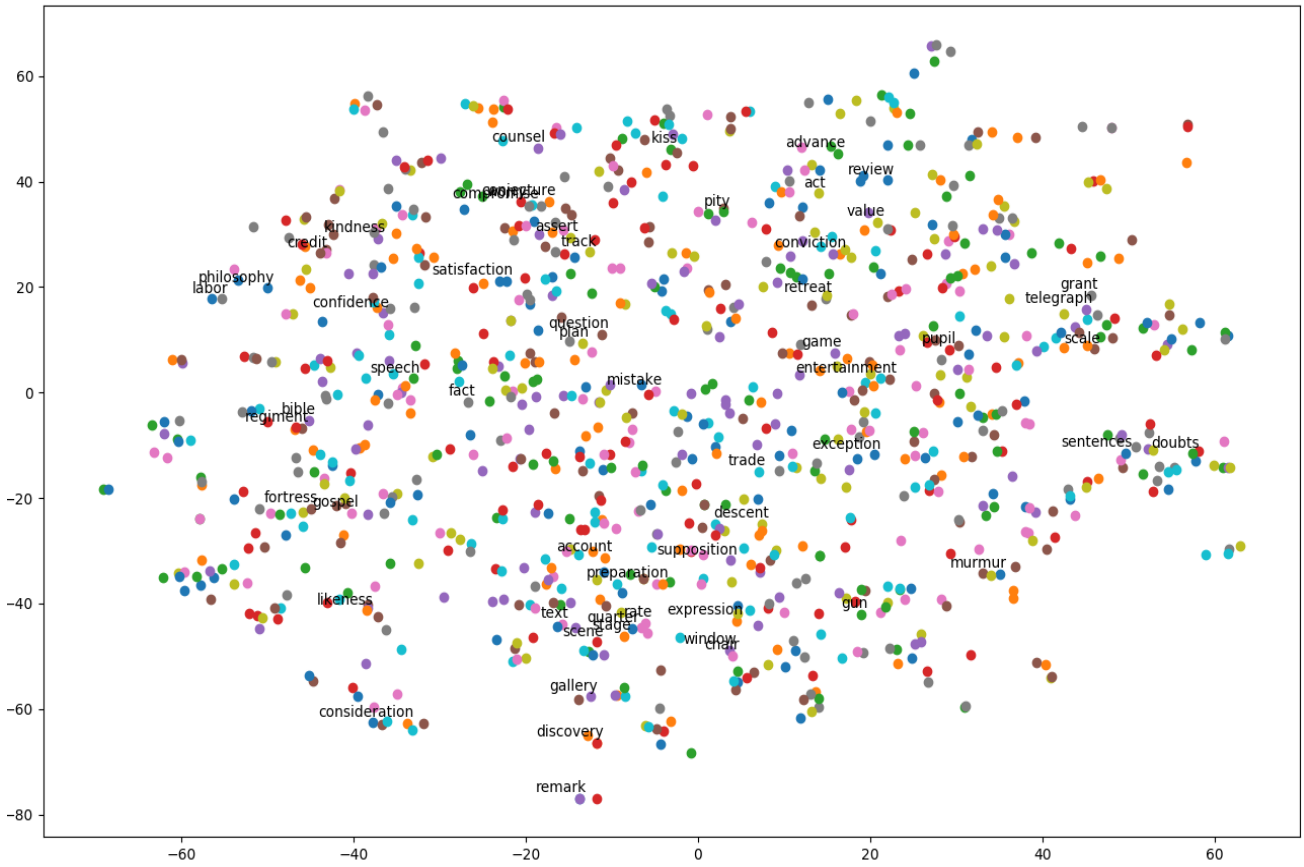


Figure 1: t-SNE visualization of 'remark'

while large context vectors that focus on content words will shift the interpretation towards a topical one.

We choose the parameter settings in such a way that the model captures both topical and syntactic word information. To achieve this, we use a word-order maintaining context window of three at either side of the target word, we include function words and bigram features, and we use as a weighting function a binary metric. The latter downgrades the importance of (highly frequent) function words, while it puts more weight on words with moderate frequencies. Also, it partially overcomes any data sparsity that might evolve from the inclusion of bigram context features. The chosen distance function is the Jaccard coefficient, which measures the number of shared features of each vector (the overlap).

After the identification of the Noun and the Adverb slot as having the most variation, we apply the described procedure to Ex. 2, which yields the results as seen in Table 2. The learner could infer from this that *rather* in combination with *obvious* specifies a degree, while *remark* can be interpreted as some sort of statement, both contributing and leading to a better distributional understanding of *obvious* and how it functions in this specific context.

3.3. Visualization of semantic clusters

As a further refinement to the above ranking, we also carried out experiments to visualize the word relationships at the hand of vclust (Suzuki and Shimodaira, 2006) and t-SNE (Van der Maaten and Hinton, 2008). We hypothesize that such visualizations will trigger a better understanding of which words within the phrasal slot are highly related to the relevant word, but also highlight which words have a set of distinctive features within the semantic space.

Target Word	Paradigmatic Alternatives	Jaccard coefficient
rather	very	0.429
	pretty	0.283
	apparently	0.234
	equally	0.220
	fairly	0.200
	perfectly	0.199
remark	question	0.256
	reason	0.248
	fact	0.229
	point	0.227
	expression	0.218
	plan	0.203
	consideration	0.190
	conviction	0.151
	tendency	0.126
	begging	0.112
	monument	0.104
	supposition	0.085
	inference	0.078

Table 2: Paradigmatic alternatives, ordered by semantic similarity to the target word (output Step 2)

Vclust is a hierarchical clustering algorithm based on multiscale bootstrap resampling. It overcomes the arbitrary input bias from a normal clustering algorithm by randomly

shuffling the provided input, effectively checking for consistency and robustness of the achieved clustering. T-SNE is a low-dimensional visualization of the achieved semantic vectors. Both visualizations are attractive as they offer an (implicit) unsupervised clustering which are poised to stimulate human interpretation.

Figure 1 shows the results of a 2D-representation of the semantic space for 'remark' using the t-SNE algorithm. The feature vectors of the 1000 nearest-neighbours (according to the Jaccard distance) have been selected to create a sufficiently large and meaningful space for the subsequent algorithm to work on. We reduce the high dimensionality of the selected vectors to 50 using Truncated Singular Value Decomposition (Halko, 2011). To account for usability, we try to avoid visual information overload by randomly labeling 90 words from the initial set of 1000 in addition to the 13 paradigmatic alternatives of 'remark' from Table 2. These pieces of information are given as input to the t-SNE algorithm; perplexity is set at 10 motivated by empirical evidence that our vector space is good at determining close neighbours, but that the effect wears off quickly. The number of iterations is set to 2000.

4. Expert Opinion and Validation Proposal

We asked 3 language teaching experts for their opinion on the usefulness of such techniques for possible applications in language learning. The experts pointed to applications that enhance receptive (comprehension) and productive vocabulary knowledge. The techniques could be integrated in for instance a reading aid, an activity mainly focused on receptive understanding. However, all experts expressed doubt whether language learners, especially high school students, would put in the extra effort to solve the linguistic puzzle laid out before them simply to understand a word, while easier alternatives, such as linked dictionary could be made available. They thus dismissed the usefulness of adding more structure from a didactic point of view for a mechanism (concordancing) which remains largely unused. One expert however thought of the technique as an excellent feedback mechanism to supplement the correction of errors as it would improve understanding why using a certain word is not correct in a specific context.

In a future validation scheme, we will therefore focus on the mechanism to serve as feedback for written exercises and tests in conjunction with automatic error analysis. Computations can in this case be performed offline and sent as a report. For language acquisition, it is of particular interest to investigate the influence of focused corpus-based information on the acquisition of certain lexical and syntactic patterns. As such the technique could prove a valid substitute, or at least a welcome addition for teacher feedback.

5. Summary and future work

We presented a set of NLP-functionalities that further structure examples retrieved for keyword-based searches. The examples are first grouped based on syntactic grounds, using POS-tags as an abstract representation of words from which constructions are derived. After this the examples are further organized at the hand of semantic similarity within certain phrasal slots. We use a self-implemented vector space model that emphasizes similarity both on a functional and a topical level. Visualization algorithms are then used to present focused information in phrasal slots, laying bare semantic relatedness of words, but also semantic restrictions within the construction.

We asked 3 experts for their opinion on the procedure's usefulness in language learning. While they were hesitant for its qualities to increase repective knowledge, they were more positive for it to be used as an automatic form of feedback.

While we intend to apply the procedure in language learning tools, we would like to point towards its utility for linguists, translators, lexicographers and other language professionals. The tools are generally intended to enable the study of language; the semantic interpretation of words and the proper understanding how to use words in context, both on a constructional, lexical and semantic level.

6. Bibliographical References

Bybee, J. (2006). From usage to grammar: the mind's response to repetition. *Language* 82(4), 711-733.

Anatol, S. and Gries, S. Th. (2003). Collostructions: Investigating the interaction between words and constructions. *International Journal of Corpus Linguistics* 8 (2), 209-43.

Chapelle, C. (1998). Multimedia call: lessons to be learned from research on instructed SLA. *Language Learning & Technology*, 2(1), 21–36.

Chinkina, M. and Meurers, D. (2016). Linguistically Aware Information Retrieval: Providing Input Enrichment for Second Language Learners. Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications (BEA), 188-198.

Croft, William A. (2001) Radical Construction Grammar: Syntactic Theory in Typological Perspective. Oxford University Press.

Goldberg, Adele E. (2006). Constructions at Work. The Nature of Generalization. Oxford University Press.

Halko, N., Martinsson, P.-G. and Trop, J.A. (2011). Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *Society for Industrial and Applied Mathematics,* 53(2), 217-288.

Krashen, S. (1981). *Second Language Acquisition and Language Learning*. Oxford: Pergamon Press

Lund, K. and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, and Computers*, 28 (2), 203–208.

Mikolov T., Sutskever I., Chen K, Corrado GS., Dean J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 3111-3119.

Nagy W., Herman P., and Anderson R. C. (1985). Learning words from context. *Reading Research Quarterly*, 20(2), 233–253.

Nation, I. S. P. (2001). Learning vocabulary in another language. Cambridge, UK: Cambridge University Press.

Sternberg R. J. (1987). Most vocabulary is learned from context. *The nature of vocabulary acquisition*, 89–106.

Suzuki R. and Shimodaira, H. (2006). Pvclust: an R package for assessing the uncertainty in hierarchical clustering. *Bioinformatics* 22(12), 1540-1542.

Tomasello, M. (2003). *Constructing a Language: A Usage-Based Theory of Language Acquisition.* Harvard University Press.

Turney, P.D. and Pantel, P. (2010). From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research*, 37(1), 141–188.

Van der Maaten, L.J.P. and Hinton, G.E. (2008). Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research* 9:2579-2605.

CBFC: a parallel L2 speech corpus for Korean and French learners

Hiyon Yoo*, Inyoung Kim[+]
*Laboratoire de Linguistique formelle CNRS 7110, Université Paris Diderot, Place Paul Ricoeur, Paris 75013, France
[+] Naver Labs Europe, 6 chemin de Maupertuis, Meylan 38240, France
*yoo@linguist.univ-paris.diderot.fr , [+]inyoung.kim@naverlabs.com

Abstract
In this paper, we present the design of a bilingual corpus of French learners of Korean and Korean learners of French using the same experimental design. This language resource contains mainly speech data, gathered among learners with different proficiency levels and in different speaking contexts (read and spontaneous speech). We aim at providing a translated and annotated corpus to the scientific community which can be used for a large array of purposes in the field of theoretical but also applied linguistics.

Keywords: L2 corpora, Korean, French, phonetics, prosody, phonetic segmentation, alignment

1. Introduction

In the last years, especially with the development of new methodologies and new technologies in corpus and computational linguistics, language corpora have become more and more common and needed in linguistic research. Corpora vary a lot in size, uses or presentations but it appears clearly that doing research using linguistic corpora brings along new questions but also new analyses of linguistic phenomena. Recently, we saw a rise of large corpora for second language acquisition (cf. among others, Granger 2003 and 2012, Hawkins and Buttery 2009). Beside the fact that few corpora are freely available to the research community (see however Milde and Gut 2002, Tortel 2008, Herment et al. 2012a and 2012b, and Gut 2009), a glimpse at the « Learner corpora around the world » database (http://www.uclouvain.be/en-cecl-lcworld.html) reveals that:
- Most existing corpora concern English
- Most existing corpora are written and not spoken
- The pair of languages French-Korean is quasi-inexistent.

Still, the use of large corpora allows a better evaluation of possible correlations between the learner's L1 (first language), his grammatical competence and his proficiency level in L2 (second language) (following for example the Common European Framework of Reference for Languages). Corpus-based studies can be used to determine how some morpho-syntactic phenomena are acquired in English as a foreign language (see for example the project English Profile, Hawkins and Buttery 2009), how students' pronunciation of L2 can be influenced by his L1 (I-PFC project, Racine et al. 2011, 2012) and so on. Depending on how the corpus is built, such corpora can also be useful for understanding learners' opinion of the target language during education.

In this paper, we present a bilingual corpus of French learners of Korean and Korean learners of French which aims at giving an answer to this missing linguistic resource by putting in parallel two languages, French and Korean.

There are today numerous corpora of French as L1, be they written or spoken (the Frantext corpus (http://www.frantext.fr/), TCOF corpus (http://www.cnrtl.fr/corpus/tcof/), PFC (http://www.projet-pfc.net/) and so on). For instance, the TUFS corpus is a representative example of corpora gathered in Aix–en-Provence where the main concern is spontaneous French. Another ambitious project is the PFC corpus, which aims at gathering data of spoken French all over the world, using the same methodology. Thus, the PFC project is based on spoken tasks only (there are no written tasks), with reading tasks (words, short texts) and also spontaneous speech (mainly interviews). One of the recent developments of PFC was to include foreign learners of French (the I-PFC project, Racine et al. 2012) following the same corpus collection protocol but adding tasks in respects to the L1). The I-PFC includes also Korean learners of French. Moreover, the I-PFC concerns only French as a second language and the equivalent for Korean using the same material is not available. The CEFLE corpus (Corpus Ecrit de Français Langue Etrangère) developed at the Lund University (http://projekt.ht.lu.se/cefle/information) is an illustration of a written corpus of French as a Second Language, but concerns Swedish learners.

Most existing corpora for Korean are written (the Korean National Corpus http://www.sejong.or.kr/user/main.do) or concern the English-Korean pair (the Gachon Korean Corpus). The I-PFC is the only project that presents the pair of language French-Korean but, as mentioned before, concerns only Korean learners of French and more specifically on their phonological competence (Han, 2011). Moreover, there are not any published results yet, and it is impossible to say how much the project has advanced.

Section 2 presents in detail the experimental design of the CBFC corpus; section 3 gives information on how segmentation and annotation was dealt.

2. The CBFC[1] Project

The "Corpus Bilangue Français Coréen" (CBFC) is a L2 corpus putting in parallel French learners of Korean and Korean learners of French. The corpus has been gathered following a unique protocol for the two populations and is made to be as complete as possible, with both written and spoken data, but with special emphasis on the prosodic aspects of L2 speaker's productions. Our aim is to make available to the research community interested in L2 phonology and prosody from this resource for the French-Korean language pair. In this section, we detail the protocol we used for gathering the spoken data.

[1] CBFC stands for "Corpus Bilangue Français Coréen" (French Korean Bilingual Corpus). The term *bilangue* is to be distinguished from *bilingue* in that we are not dealing with a bilingual corpus but a parallel corpus between two languages.

2.1 Subjects and Recording Procedure

A total of 22 subjects participated to the recordings of the speech data: one native speaker for each language, and 10 learners for each language. We established a linguistic profile for each speaker, with information concerning their knowledge of other languages, their L2 proficiency and so on. Some of the detail of the participants is given in tables 1 and 2.

French subjects for L2 Korean	Age*	Level**	Bilingual	Other known second languages
F01	18	2y		English, German
F02	17	4y		English, Arabic, Spanish
F03	18	3y	Lingala	English, Spanish
F04	17	3y	Turkish	English, Spanish,
F05	27	2y	Antillian creole	English, Russian
F06	18	1.5y	Lingala	English, Italian, Spanish,
F07	17	2.5y	Italian	English, Italian
F08	19	5y		English, Russian, Italian
F09	n/a	6m		
F10	n/a	2y		

Korean subjects for L2 French	Age*	Level**	Stay in France***	Education
M01	23	1y	21d	International relations
F01	19	2y	15d	French language and literature
F02	15	5y	15d	French language and literature
F03	21	1m	15d	Economy
F04	16	5y	1m	French language and literature
M02	19	4y	n/a	French language and literature
F05	26	8y	8y	Costumier
M03	25	3.7y	3.7y	Cinema
F06	20	7y	7y	Architecture
M04	31	3y	3y	Pianist

Table 1: Details of French and Korean participants: *age when subject started learning L2, **years of study, ***period time of staying at the immerging country; years(y), months(m), days(d)

Recording sessions took place at the recording room of University Paris Diderot in France. We used a portable Roland R-26 and a XLR microphone. Each recording session lasted from 40 to 80 minutes depending on the subject. The subjects had the possibility to make a pause between two different tasks.

As subjects were recruited in Paris, the two language groups show different characteristics. As it is shown in Table 1, Korean participants had English as the only second language, and only one female participant considered herself as bilingual. There were several languages learned as second language, and five French participants considered themselves as bilingual. Korean participants had various education background compared to French who were recruited at the department of Korean language in Paris Diderot University.

2.2 The Corpus

The corpus is based on the COREIL protocol which was initially developed for the English/French (Delais-Roussarie and Yoo 2011) and extended to other language pairs such as Spanish/French (see Santiago, F. & E. Delais-Roussarie. (2012), Delais-Roussarie, Santiago, F. & Yoo H. 2015)). It is inspired by the AGILE corpus Voormann, H. & U. Gut. 2008, Gut 2009. This protocol which was initially designed for the study of L2 intonation and phrasing presents the advantage to be modular, and it is easy to add or remove tasks. It can also be adapted following the performance level of speakers. We thus enriched the initial protocol in order to obtain specific type of sentences.

We distinguish in this corpus three different groups of tasks:

- Reading
- Spontaneous Speech in monologue, dialogues, and interviews.
- ToBI questionnaire

The reading task is composed by three different texts: small monologues, small dialogues like those which appear in learners' textbooks (see Figure 1) and an excerpt of *Little Prince* (from Saint Exupéry). The texts are rich in type of sentences, and therefore, we expect subjects to produce various intonational contours in order to signal interrogatives or other illocutionary forces.

Server: *Bonjour. Une table pour deux personnes ?*
Client: *Oui, nous sommes deux. Vous avez un espace non-fumeur ?*
Server: *Bien sûr. Vous préférez cette table, ou celle-ci, près de la fenêtre ?*
Client: *Plutôt celle-ci.*
Server: *Très bien. Installez-vous. Voici le menu.*

Figure 1: Example of a dialogue reading task in L2 French

Spontaneous speech was obtained during four different tasks:



Figure 2: Images used for the image description task

- **Image description**: subjects are invited to describe and comment freely the image which is shown. Figure 1 gives an idea of the type of images they had to describe.
- **Question guided monologue**: the examiner asks questions and subjects are invited to answer freely.
- **Questionnaire completion**: subjects are provided with a questionnaire they had to fill, after interviewing the examiner.
- "**Who am I?**" **game**: subjects are invited to play a game where they had to ask questions in order to guess the face their co-gamer has chosen. Figure 3 gives an example of the type of data that were obtained. The motivation of the game setup is to obtain interrogative type of sentences with a coherent context given to each participant.

- mʌri-ga	noransɛg-iejo
hair-NOM	yellow-DEC *(Is the hair yellow?)*
- ne *(Yes.)*	
- kopsɯlmʌri-ejo	
curly-DEC *(Does he or she has curly hairs?)*	
- jak'an *(A little)*	
- luk'a *(Lucas?)*	
- madʒajo	
correct-DEC *(It's correct)*	

Figure 3: Example of data from the *who am I?* game task

For the last task, we used the ToBI questionnaire used for gathering data of the IARI project and intonation in romance (Frota S. & Prieto P. 2015). In this project, the idea was to adopt the same framework and the same methodology in order to obtain comparable intonational contours for romance languages. This kind of questionnaire presents the advantage to force the subjects to perform certain utterances for a specific given context. However, it is a rather difficult task for beginners or intermediate learners.

Table 3 gives a mean value of the speech time for each task.

L2 French		Mean speech time (minutes)
Read speech	Texts	2.6
	Petit Prince	3.7
	Dialogues	4.5
Spontaneous speech	Monologue (Q-R)	16
	Image describing	5.3
	Free questionings	5.3
	Questioning game	9
Questionnaire ToBI		20
TOTAL		66

L2 Korean		Mean speech time (minutes)
Read speech	Texts	5
	Dialogues	8.6
Spontaneous speech	Monologue (Q-R)	14.2
	Image describing	5.8
	Free questionings	4.6
	Questioning game	7.1
Questionnaire ToBI		28
TOTAL		73.3

Table 3: Speech time by speech type for L2 French and L2 Korean

3. Segmentation and Annotation

As noted by Lüdeling and Hirschmann (2015), coding the learners' errors is a difficult task and it is important to minimize interpretation. For the moment, we proceeded at an orthographic transcription of the data. We then used semi-automatic tools in order to provide a segmentation and an annotation at the segmental level.

3.1 French Productions

French productions were aligned using the EasyAlign software (Goldman 2011). EasyAlign is an executable software that can be used in Windows environment only, which adds a plugin into the Praat software (2017). When provided with an orthographic transcription of the sound file, it gives as a result is a multi-tier annotation within the Textgrid in Praat, offering a segmentation into phones and words.

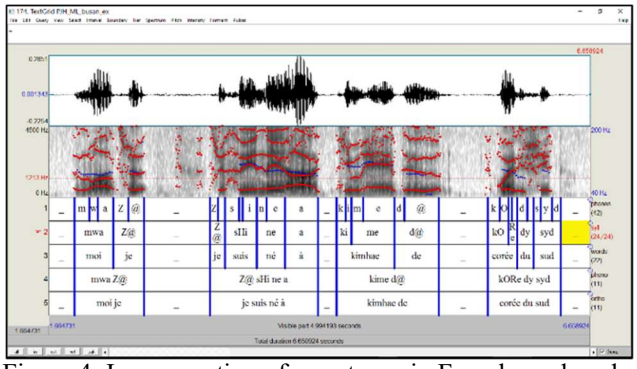


Figure 4: Image caption of a sentence in French produced by a Korean learner (M01, L2 French)

Figure 4 gives an image caption of the segmentation and annotation obtained after manual correction of the boundaries. EasyAlign provides five different tiers (phone, syllable, word, phonetization and orthographic level).

3.2 Korean Productions

Since the EasyAlign software is not yet available for Korean, we had to use another tool for segmenting and annotating Korean data. Korean data were aligned with the automatic alignment function supplied in the 'Interval' menu in Praat (version 6.0.25). The automatic alignment is proposed for many languages, and we tested the available option 'Korean-test'. We selected all the option boxes, i.e., 'word', 'phonemes', 'silences', and this function creates two subordinate tiers; '/word' tier and a '/phon' tier.

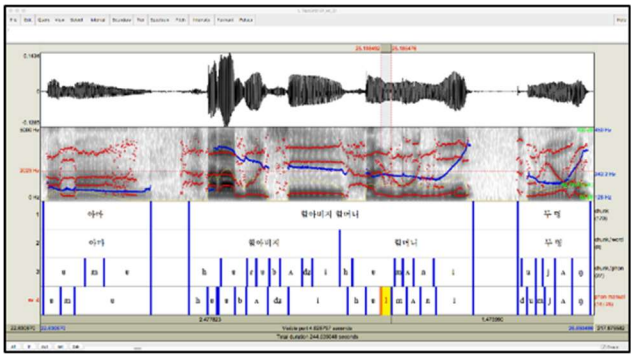


Figure 5: Image caption of a sentence in Korean produced by a French learner (F05, L2 Korean)

When Korean orthograph hangul is correctly transcribed and aligned in an interval tier (tier1 in Figure 5, named *chunk*), the second /*word* tier and the third /*phon* tier are created automatically within the boundary aligned to the first tier. The /*word* tier creates an *oejeol* unit, which can be a lexical word or a combination of a lexical word with a grammatical morpheme in Korean grammar. Consequently, when two lexical words are aligned as one unit in the first interval tier, the /*word* boundary of the two tiers between the words should be corrected also. Additionally, most of the time, the coda /l/ is ignored, and Korean intervocalic velar lenis stop is transcribed as /q/, and /g/ transcribes initial velar lenis stop by the automatic aligner. The fourth tier is duplicated from the third and manually corrected (tier 4).

In addition to the three tiers of segmentation and annotation, a *prosody* tier and a misc(miscellaneous) tier are created for the prosody annotation(Figure 6). The prosodic tier is completed manually, and the misc tier serves for remarks.

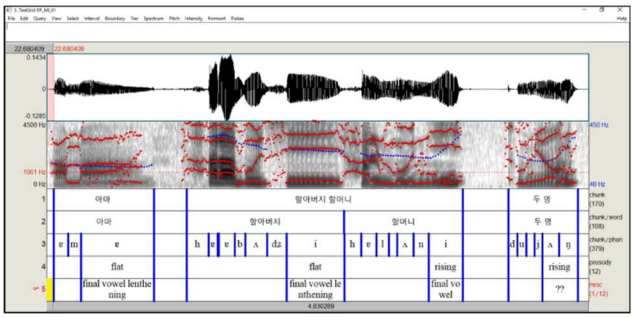


Figure 6: Example of an alignment after correction of the /*phon* tier and two additional tiers for prosody (F05, L2 Korean)

4. Conclusions and Perspectives

In this paper, we present the experimental design of a parallel L2 learner corpus for the pair of languages Korean/French. We aim at providing raw but controlled parallel material for research in comparative linguistics, (with special attention to syntax, morphology, phonology and prosody), but also in applied linguistics, and bring to teachers, material for building assessments and language evaluation tools. We believe that such a parallel corpus can enable to evaluate the weight and role of the learner L1 as well as the differences and/or similarities between L1 and L2 acquisition. We will contribute at bringing to the scientific community working on Korean and French but also second language acquisition a valuable linguistic resource, and more collaborations to be established.

At the moment, we are finalizing annotation corrections for Korean data, and we are working on the deposition of the linguistic resource.

5. Acknowledgements

This project was initiated by co-funding from University Paris Diderot and Seoul National University, and followed by Labex Empirical Foundations for Linguistics funding.

6. Bibliographical References

Boersma, Paul & Weenink, David (2017). Praat: doing phonetics by computer [Computer program]. Version 6.0.33, retrieved 26 September 2017 from http://www.praat.org/

Common European Framework of Reference for Languages: Learning, Teaching, Assessment, Language Policy Unit, Strasbourg (www.coe.int/lang-CEFR)

Delais-Roussarie E. and H. Yoo. (2011) Learner corpora and prosody: from the COREIL corpus to principles on data collection and corpus design. Poznań Studies in Contemporary PSCIL 47: 28-39.

Delais-Roussarie, E. Santiago, F. & Yoo, H. (2015) the extended COREIL corpus: first outcomes and methodological issues, Satellite workshop of ICPhS 2015, August 2015 Glasgow

Detey, S., Durand, J., Laks, B. and Lyche, C. (2016) (éds). Varieties of Spoken French: a source book. Oxford: Oxford University Press.

Frota, S. & Prieto, P. (2015), Intonation in romance, OUP.

Gut, U. (2009). Non-native speech. A corpus-based analysis of phonological and phonetic properties of L2 English and German. Frankfurt: Peter Lang.

Granger, S. (2012). Learner Corpora. In: C.A. Chapelle, The Encyclopedia of Applied Linguistics, Wiley-Blackwell: Oxford, 2012. 978-1-4051-9473-0.

Granger, S. (2003). The International Corpus of Learner English: A New Resource for Foreign Language Learning and Teaching and Second Language Acquisition Research. TESOL 37 (3). 538-546.

Goldman J.-Ph. (2011) *EasyAlign: an automatic phonetic alignment tool under Praat* Proceedings of InterSpeech, September 2011, Firenze, Italy (http://latlcui.unige.ch/phonetique/easyalign.php)

Gut, U. (2009): Non-native speech. A corpus-based analysis of phonological and phonetic properties of L2 English and German. Frankfurt: Peter Lang.Han, M.-H. (2011). Fautes de prononciation des Coréens apprenant le français et correction phonétique. Synergies Corée 2 : 73-82.

Hawkins, J. & Buttery, P. (2009). "Using learner language from corpora to profile levels of proficiency: insights from the English Profile Programme". In Taylor, L. & Weir, C. J. (dir.), Language testing matters: investigating the wider social and educational impact of assessment. Cambridge: Cambridge University Press. pp. 158-175.

Herment, S., Tortel,A., Bigi, B., Loukina, A., Hirst, D. J., Kochanski, G. (2012a). AixOx, a multi-layered learners corpus: automatic annotation, Proceedings of the 4th International Conference on Corpus Linguistics (4 : 2012 mars 21-24 : Jaèn, Spain).

Herment, S., Loukina, A., Tortel, A. (2012b). AixOx. Available on SLDR (Speech Language Data Repository), http://sldr.org/sldr000784/fr

Lüdeling, A., Hirschmann, H. (2015). Error annotation. In: Granger, S., Gilquin, G., Meunier, F.,(eds), The Cambridge Handbook of Learner Corpus Research Cambridge: Cambridge University Press.

Milde & Gut 2002 "A prosodic corpus of non-native speech", In Proceedings of Speech Prosody 2002, Aix-en-Provence, pp.503-506.

Racine, I., Zay, F., Detey, S. and Kawaguchi, Y. (2011), « De la transcription de corpus à l'analyse interphonologique: enjeux méthodologiques en FLE ». In Travaux Linguistiques du CerLiCO 24, Rennes: PUR.

Racine, I., Detey, S., Zay, F. and Y. Kawaguchi (2012). Des atouts d'un corpus multitâches pour l'étude de la phonologie en L2 : l'exemple du projet « Interphonologie du français contemporain » (IPFC). In A. Kamber et C. Skupiens (éds). Recherches récentes en FLE. Berne: Peter Lang.

Santiago, F. & E. Delais-Roussarie. (2012). Acquiring phrasing and intonation in French as a second language: The case of Yes-No questions produced by Mexican Spanish Learners. In Ma, Q., Ding, H. & Hirst, D. [Eds.]. Proceedings of Speech Prosody 2012, Shanghai, China, 338-341

Tortel, A. 2008. ANGLISH : base de données comparative L1 & L2 de l'anglais lu, répété, parlé. TIPA 27: 111-122.

Voormann, H. & U. Gut. 2008. Agile Corpus Creation. Corpus Linguistics and Linguistic Theory 4 (2): 235-251.

7. Resources

CEFLE corpus: http://projekt.ht.lu.se/cefle/information

Frantext project: http://www.cnrtl.fr/corpus/frantext/

IARI project: Prieto, P., Borràs-Comes, J., & Roseano, P. (Coords.) (2010-2014). Interactive Atlas of Romance Intonation. Web page: <http://prosodia.upf.edu/iari/>

Korean National Corpus : http://www.sejong.or.kr/user/main.do

Learner corpora around the world database: http://www.uclouvain.be/en-cecl-lcworld.html

PFC Project : http://www.projet-pfc.net/

TUFS project: http://www.coelang.tufs.ac.jp/multilingual_corpus/fr/index.html?contents_xml=corpus&menulang=en

TCOF project: http://www.cnrtl.fr/corpus/tcof/

SW4ALL: a CEFR-Classified and Aligned Corpus for Language Learning

Rodrigo Wilkens, Leonardo Zilio and Cédrick Fairon
Centre de Traitement Automatique du Langage - CENTAL
Université catholique de Louvain - UCL
{rodrigo.wilkens, leonardo.zilio, cedrick.fairon}@uclouvain.be

Abstract

Learning a second language is a task that requires a good amount of time and dedication. Part of the process involves the reading and writing of texts in the target language, and so, to facilitate this process, especially in terms of reading, teachers tend to search for texts that are associated to the interests and capabilities of the learners. But the search for this kind of text is also a time-consuming task. By focusing on this need for texts that are suited for different language learners, we present in this study the SW4ALL, a corpus with documents classified by language proficiency level (based on the CEFR recommendations) that allows the learner to observe ways of describing the same topic or content by using strategies from different proficiency levels. This corpus uses the alignments between the English Wikipedia and the Simple English Wikipedia for ensuring the use of similar content or topic in pairs of text, and an annotation of language levels for ensuring the difference of language proficiency level between them. Considering the size of the corpus, we used an automatic approach for the annotation, followed by an analysis to sort out annotation errors. SW4ALL contains 8,669 pairs of documents that present different levels of language proficiency.

Keywords: SLA, CEFR classification, Aligned corpus

1. Introduction

Learning a second language is a process that requires exposition to texts, especially for the acquisition of vocabulary (Rott, 1999). To retrieve texts that match learners' language level (or proficiency) it is possible to use a corpus carefully designed for language learners, or one can search for them in the web. Following these alternatives, systems can dig up texts aiming at finding those that are best suited to the language skills of a learner. Examples of these systems are REAP (Heilman et al., 2008), FLAIR (Chinkina et al., 2016), and READ-X (Miltsakaki and Troutt, 2007), which use the web as a corpus. The use of web allow the learners to interact with a huge diversity of texts, which makes it easier to find those that correspond to their interests. But most of the web texts retrieved by search engines require a high language proficiency, even for native speakers (Vajjala and Meurers, 2013). On the other hand, the use of an off-line corpus ensures content quality, while hindering the search for different text topics.

This dichotomy of text sources enforces different types of restrictions on the systems. An alternative for trying to get the best of both approaches is to use the web as source of texts, but restricting it to trustful domains. This is similar to SourceFinder's method, in which on-line newspapers and magazines are downloaded and processed as text sources (Passonneau et al., 2002; Sheehan et al., 2007). However, this kind of approach doesn't allow for an easy update of the texts, because the content is stored off-line, and an update would require a rerun of the whole corpus compilation process. Another reliable type of text source that is adequate to language learners is corpora made up of simplified texts, such as the Weekly Reader, the Simple English Wikipedia, and the BBC Bitesize. This type of source generally represents an attempt to make texts more accessible, by adapting or simplifying them to present a language that should be easier to understand for a non-native speaker. Wikipedia also has concerns on the comprehension abilities of its users, so that, for the English language, there is a simplified version, addressing the needs of natives with low literacy, but also the needs of learners of English. This type of resource presents a simplified version of a source article, serving as a facilitator for the communication of knowledge for those with less language skill, but it doesn't present an information about to what extent the text is simplified. It doesn't explain the simplification strategies applied to each text or for what target reader each text was simplified, and this, in the case of language learning, which categorizes learners in different levels, is crucial to better inform the learner about which texts would be at an understandable level. For instance, the Common European Framework of Reference for Languages (CEFR) classifies learners in six language levels (ranging from A1 to C2), while other frameworks, like the Cambridge ESOL classifies them in 5 levels, and still other frameworks use score ranges, like TOEFL and IELTS. Without this information about the language level, or some other information about the adequacy to a given target reader, language resources that present a simplified version are, per se, not well suited for language learners, because the simplification requirements may not be of use for the needs of specific learners. As such, another layer with a more pedagogically relevant classification is needed.

This study aims at automatically determining the CEFR level of pairs of original and simplified texts, so that a corpus of paired texts pertaining to different language levels can be used in a language learning framework.[1] This would allow learners to compare text structures from different levels that describe the same content, while also allowing for the selection of topics of interest. To this end, we annotated language levels in an aligned version of the English

[1]The developed resource is available at `http://cental.uclouvain.be/resources/smalla_smille/sw4all/`

Wikipedia[2] (EW) and Simple English Wikipedia[3] (SEW), and filtered pairs of texts that are associated to different levels, but that refer to the same topic or content. This process resulted in a resource that we called *Simple Wikipedia for Aligned Language Learning* (SW4ALL).

This paper is organized as follows: Section 2. presents systems that classify texts for the purpose of language learning; Section 3. describes the training corpus, the detailed training methodology and its results, and the application of the trained model to the aligned EW-SEW; Section 4. is where the evaluation of the classification model is presented; Section 5. contains a description of the resulting annotated corpus; and, finally, Section 6. is reserved for our final remarks on this study.

2. Related Work

The search of texts that can improve language learning skills, and, at the same time, be able to match the learners' interests is a very time-consuming activity for language teachers. Aiming to reduce the time allocated to this task, the SourceFinder (Passonneau et al., 2002; Sheehan et al., 2007) allows teachers to search for documents classified in different language levels (according to the Graduate Record Examinations curriculum) by means of keywords. One of the advantages of SourceFinder is the use of off-line texts, which allows the processing of the texts with several NLP tools without delay to the user. On the other hand, it only allows the search for text content and grammar structures.

Using online documents, the REAP (Heilman et al., 2008), READ-X (Miltsakaki and Troutt, 2007), and LAWSE (Ott and Meurers, 2011) systems allow the users (learners or teachers) to search texts by means of keywords and to filter them according to readability measures. Those systems identify the text readability by applying traditional readability scores (Flesch-Kincaid measure (Kincaid et al., 1975), and Gunning Fog Index (Gunning, 1952)) that are based on shallow cues (e.g. number of words per sentences and syllables per word). These measures have the advantage of a fast annotation process, but they are not accurate, and they require the users to deal with a score that may not be familiar to them.

Using a more accurate text classification method, the FLAIR system (Chinkina et al., 2016; Chinkina and Meurers, 2016) dynamically crawls, annotates, and classifies the 20 first results of a search engine. The FLAIR text classification is based on parsing information and on the official English language curriculum of schools in Baden-Württemberg (Chinkina et al., 2016).

Taking into account the pedagogical function of these systems, a major point is their ability to address documents that are readable by learners. However, text length-based readability scores weigh only sentence length and word difficulty, ignoring factors such as cohesion (Bruce et al., 1981). Recently, Xia et al. (2016) compared syntactic and length-based features for text classification according to language level, and identified that adding syntactic features on top of length-based features improved the classification results, but using only length-based features presented a better result than the syntactic features alone.

Regarding the representation of the texts in features, there is a huge variety of options in the literature. They can be grouped into 6 categories: length-based (e.g. word and sentence length), lexical (e.g. proportion of words in a list of easy words), morphological (e.g. part of speech), syntactic (e.g. presence of passive voice), semantic (e.g. word polysemy), and language model (e.g. n-gram model perplexity). The syntactic features could be split into two groups, depending on how the parser is used. Usually, a parser-based annotation of features follows the same process as the morphological annotation: simple counts of parser annotation. However, some studies, such as François and Fairon (2012) and Callan and Eskenazi (2007), used information beyond parsed tags.

All those systems focus on presenting a readable text, but some of them go beyond that, presenting exercises for supporting the learning activity in a more active way (e.g. REAP). In spite of all the effort to present readable and interesting texts to learners, those systems do not indicate how learners can improve their skills by using the indicated texts.

3. Methodology

Regarding the objective of automatically determining pairs of texts that present good examples of different language levels, we trained a classifier and applied it to the aligned English Wikipedia (EW) and Simple English Wikipedia (SEW). This annotated resource was named Simple Wikipedia for Aligned Language Learning (SW4ALL). In this section, we present first the alignment between the two Wikipedia versions (Section 3.1.). We then move on to the resources needed to build the classifier: the training corpus (Section 3.2.) and the feature set (Section 3.3.). Finally, we discuss the application of the classifier to the aligned EW-SEW (Section 3.4.).

3.1. Aligned Wikipedias

The Wikipedia is a collaborative encyclopedia with huge amounts of texts available in several languages. In English, there are two version, one that focus just on encyclopedic information, and the other that requires the content to be written in a simplified way. Comparing the vocabulary of the two encyclopedias, Coster and Kauchak (2011) identified that 96% of the words in the simple version are found in the other version, and 87% of the words in the normal version are found in the simple version. This overlap is also found at the n-gram level. Regarding the alignment of the Wikipedias, there are different versions, and in this paper we opted for the version organized by Kauchak (2013), in which the texts were aligned both at the document and sentence level. Kauchak (2013) downloaded and cleaned all articles from the Wikipedias (removing stubs and navigation pages), resulting in 60K articles each. The difference in the number of sentences between the Wikipedia versions is partially because some articles from SEW present just partial information. Indeed, the sentence level alignment, also presented by Kauchak (2013), was possible in only 28% of

[2] https://wikipedia.org/

[3] https://simple.wikipedia.org/

the sentences from the SEW (and in 4.25% of the sentences from EW).

3.2. Training Corpus

Focusing on improving the learners' writing skills, we opted to use a corpus of texts written by language learners. In that way we are able to present texts compatible with learners' productive skills, making it easier for them to identify the structures that exist in those texts. So, by applying a model that was trained on texts produced by learners to the Wikipedias, we expect that the structures in the text will be familiar to the language learners, while also providing an authentic source of information, because the texts of the Wikipedias were written by native speakers.

In this study, we used the EF-Cambridge Open Language Database (EFCAMDAT) (Geertzen et al., 2013) as training corpus. This corpus is divided according to the Common European Framework of Reference for languages (CEFR) (Verhelst et al., 2009), containing a total of 532 thousand documents (33 million tokens) written by 83,385 learners of 137 nationalities, and each document has an evaluation score and an associated topic (e.g. introducing yourself by email). The data is distributed into three mains levels, each with two sublevels (all referenced by a letter and a number): basic (breakthrough or A1; waystage or A2), independent (threshold or B1; vantage or B2), and proficient (effective operational proficiency or C1; mastery or C2). The EFCAMDAT corpus contains an unbalanced number of documents per level (e.g. 151 thousand documents for A2 and 23 thousand for B2), so we selected 9,000 random documents from each level, and also filtered out those documents that did not achieve an evaluation score higher than 80%, because, in these cases, learners' errors could have an impact on the machine learning approach (Pilán et al., 2016). The result of this process is a corpus of 40,946 documents (9,000 for levels A1, A2, B1, and B2, and 4,946 for C1[4]).

3.3. Feature Annotation

The annotation process was three-fold: first the documents were automatically parsed with the Stanford Parser (Manning et al., 2014), and then a series of features were annotated, including the ones developed by project SMILLE (Zilio and Fairon, 2017), which have a good performance, comparable to state-of-the-art parsers in the labeling task (Zilio et al., 2017a; Zilio et al., 2017b). Finally, we annotated the documents with readability scores. The annotations were grouped in four categories, inspired by Xia et al. (2016): length-based, morphological, syntactic, and readability.

In addition to these sets of features, we also took into account the grouping of these morphological and syntactic features according to two criteria: CEFR level (e.g., A1, A2, etc.) in which they should be learned[5], which resulted in 5 grammatical and 5 word features; and type of grammatical structure (still respecting the CEFR levels division; for instance, connectives, which are learned on CEFR level B1, were put together in one set, but modals, which are learned in different levels, were separated in two sets). These sets of features were called pedagogical feature sets.

3.4. Annotating the Aligned EW-SEW

The model trained on the Cambridge Corpus data was applied to the aligned version of the English Wikipedia (EW) and the Simple English Wikipedia (SEW), so that we could observe which pairs of texts are suitable for contrasting different CEFR levels of English. Based on the premises of the SEW that the texts should be simpler than the EW, they should at least be on the same CEFR level as their EW counterpart, so we considered that all pairs for which the system classified the SEW text as having a higher CEFR level than the EW were bad for SW4ALL. Conversely, all the pairs for which the system presented the SEW text as being from a lower CEFR level than its EW counterpart were considered good for the resource. Pairs which presented the same level for both Wikipedias were further analyzed for level tendencies, as we discuss in Section 5..

4. Model evaluation

The first topic to be addressed in this evaluation is the quality of the model used to annotate the corpus. First, in Section 4.1., we evaluate the prediction power of each feature. Then, we discuss the corpus size impact on the model's performance (Section 4.2.).

4.1. Feature selection

To compare the features' quality, we ranked all of them according to the Gain Ratio algorithm (Frank et al., 2009), an entropy-based feature selection algorithm which ranks each feature according to its pertinence for separating the classes. We observed that 5 pairs of features (2 morphological, 2 syntactic and 6 pedagogical) presented a low score difference (< 0.00001) in the pair. This happens because the value of each pair comes mostly from one feature, so the group effect is not observed. As such, we changed the rank position of these features to bear the same value of those with which there was no substantial difference, aiming to make a fairer comparison. We also removed 12 features scored as zero by the Gain Ratio algorithm[6].

The rank distribution of the different types of features is presented in Figure 1, which displays the overall importance of pedagogical features for the model. However, it is important to notice that the rank is toped by length-based features, followed by morphological features. Interestingly, in the top 10 features, 5 are pedagogical, and the best of them is grammar-based, while the others are all vocabulary-based. Since the rank distribution does not seem to follow a normal distribution, we looked at median and quartiles of the feature types. Ranking them, we observed this, in ascending order: pedagogical, morphological, length-based,

[4]We used all documents that were scored over 80% C1 data, and we did not use the C2-level documents due to the low number of documents.

[5]For allocating each structure to a given CEFR level, we used SMILLE's pedagogical model.

[6]From the features scored as zero, one is length-based, one is vocabulary-based, four are morphological, four are syntactic, and two are pedagogical.

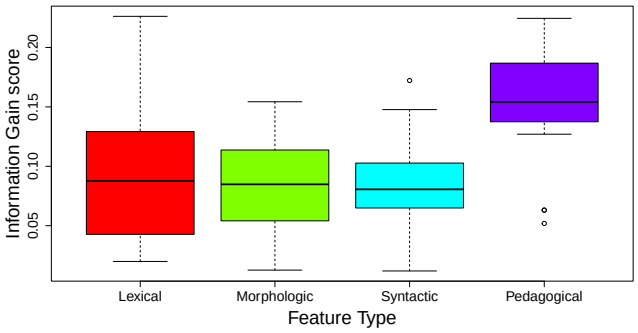


Figure 1: Score of each type of feature according to the Information Gain algorithm

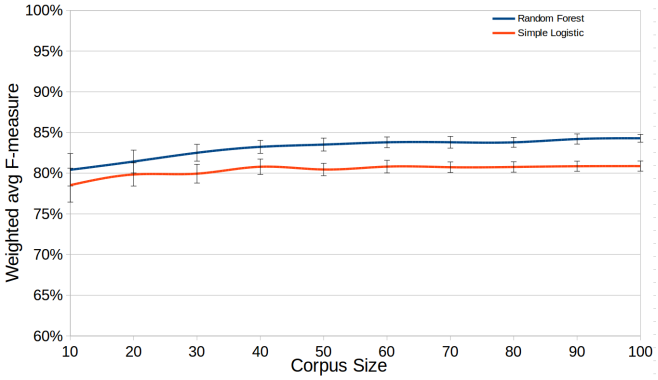


Figure 2: Average f-measure and standard deviation for the ten-fold cross-validation of the Simple Logistics and Random Forest models

syntactic, and readability-based features. Considering the rank distribution of the features, we observed that the Q1 (top 25% of the features) was ranked lower than 16.25 for length-based, 23.25 for pedagogical, 23 for morphological, 47.75 for syntactic, and 66 for readability features.

The results suggest that length-based features have great relevance for the level classification, followed by the pedagogical features. They also indicate no large difference between the morphological and syntactic features. The result of the pedagogical features is not a surprise, since they are the association of morphological and syntactic features to a pedagogical curriculum.

4.2. Corpus size

Along with the feature weight, the corpus size is an important feature in machine learning. So, to evaluate the quality increase of our model in relation to corpus size, we performed ten experiments with varying corpus sizes, from 10% to 100% of the corpus. In each test, we used all feature sets and performed a ten-fold cross-validation with the Simple Logistic and the Random Forest algorithms, as can be seen in Figure 2. The model performance was evaluated in terms of precision, recall and f-measure.

The f-measure increases an average of 0.15% for each increment of 10% in corpus size. However, in regard to statistical confidence, we identify a significant increase of f-measure only when the corpus is increased by at least 20% (1,590 instances), and no difference was observed in sizes larger than 40%. Despite the nonexistence of statistical difference in larger samples, they present a smaller standard deviation. In other words, the result is more reliable using larger samples.[7]

To the best of our knowledge, there are no studies addressing the classification of texts written by English learners. However, in the literature there are some studies that are similar to ours. For instance, Pilán et al. (2016) address the same task, but using the MERLIN corpus (Wisniewski et al., 2013), which contains documents written in Czech, German, and Italian (80% of f-measure), and Xia et al. (2016) employ a similar set of features, but using the Cambridge English Exams dataset, which is made up of text written by native speakers (80% of accuracy). Still, in an effort to establish a basis of comparison, using corpus sizes that were similar to those studies, we achieved an F1 of 82% and an accuracy of 80%. If we consider the full corpus, we have an F1 of 84.7%.

5. Results

For developing a CEFR-classified corpus, we annotated the aligned texts of the English Wikipedia (EW) and the Simple English Wikipedia (SEW). Applying a conservative approach, we considered it a classification error when the pairs of documents had a lower language level annotated for the EW. From more than 60 thousand pairs of texts, a good amount (10,225) was classified as having the same level in both Wikipedias. For these, we further looked at the distribution as a tie breaker. For instance, a pair in which both texts were classified as B1 was investigated to see if the distribution tended to show that the SEW text was easier than the text from the EW. This process identified 2,223 pairs of documents for which the SEW version tends to present a lower level. The system classified 9,222 pairs as having an SEW document that was classified as at least one level lower than its EW counterpart, which forms a more reliable set of pairs for language learning purposes. This process left us with 11,445 pairs of texts in which the SEW document was deemed to present a lower level in relation to its EW counterpart.

To ensure the quality of the resource, we turned ourselves once again to the SEW assumptions, which indicate that texts should explain complex concepts to the user, while also splitting complex sentences, so as to form shorter, simpler sentences for the reader, but presenting the same content and with even longer texts (due to the explanations). With this information in mind, we cleaned from the corpus pairs in which the SEW text had a size (in number of words) of 90% or less than its EW counterpart, for the pair would almost certainly not present the same content, let alone a SEW version with more explicitations[8]. This cleaning process removed 1,359 pairs of documents from

[7] Analyzing the results of the model, even with a larger corpus, we expect that a similar performance should be achieved.

[8] We did not restrict a maximum size, because it would be impractical to establish how much explicitations or sentence splitting would be too much.

the good sample, resulting in a subcorpus from the aligned EW-SEW containing 10,086 documents.

As a final step for ensuring the reliability of our classification for a user of the resource, we clustered the distribution of probabilities for each level from the classifier using the k-means[9] algorithm Arthur and Vassilvitskii (2007), so as to distinguish how well our classified data matched the assumptions of the SEW. We organized the clusters in suited or non-suited according to the purity as a measure of confidence. We were able to identify documents that probably present labeling error from the annotator and documents that are correctly classified. Considering only those documents that had a confidence score of more than 95%, SW4ALL consists of 6,394 pairs of documents (63% of all the documents that were considered suited), but, by relaxing this confidence to all of those above 85%, the size of the resource increases to 8,669 pairs of documents (86% of the documents that were considered suited), while maintaining a good confidence in the classification.

6. Conclusion

In this paper, we presented SW4ALL, a resource that focuses on the contrast of aligned texts that belong to different CEFR levels. This resource could be employed by teachers or students to compare grammar, vocabulary and general text structure of texts in different levels, but with roughly the same content.

A classification model was trained using an annotated version of the EFCAMDAT corpus, and the model was then applied to classify pairs of aligned documents from the English Wikipedia (EW) and its simplified version, the Simple English Wikipedia (SEW). After further analysis, pairs of documents in which the document from the SEW were classified as having a lower level, or a tendency to have a lower level, were used in SW4ALL, resulting in a total of 8,669 pairs of documents.

The pairs of documents present the same content or topic, so that SW4ALL can be a rich resource for aiding teachers and learners that wish to compare different linguistic strategies for writing a similar content, providing an interesting option for improving the learning of English as a second language.

7. Acknowledgements

The authors would like to thank the Walloon Region (Projects BEWARE n. 1510637 and 1610378) for support, and Altissia International for research collaboration.

8. Bibliographical References

Arthur, D. and Vassilvitskii, S. (2007). k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics.

Bruce, B., Rubin, A., and Starr, K. (1981). Why readability formulas fail. *IEEE Transactions on Professional Communication*, (1):50–52.

Callan, J. and Eskenazi, M. (2007). Combining lexical and grammatical features to improve readability measures for first and second language texts. In *Proceedings of NAACL HLT*, pages 460–467. sn.

Chinkina, M. and Meurers, D. (2016). Linguistically aware information retrieval: Providing input enrichment for second language learners. In *BEA@ NAACL-HLT*, pages 188–198.

Chinkina, M., Kannan, M., and Meurers, D. (2016). Online information retrieval for language learning. *ACL 2016*, page 7.

Coster, W. and Kauchak, D. (2011). Learning to simplify sentences using wikipedia. In *Proceedings of the workshop on monolingual text-to-text generation*, pages 1–9. Association for Computational Linguistics.

François, T. and Fairon, C. (2012). An ai readability formula for french as a foreign language. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 466–477. Association for Computational Linguistics.

Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I. H., and Trigg, L. (2009). Weka-a machine learning workbench for data mining. In *Data mining and knowledge discovery handbook*, pages 1269–1277. Springer.

Gunning, R. (1952). The technique of clear writing.

Heilman, M., Zhao, L., Pino, J., and Eskenazi, M. (2008). Retrieval of reading materials for vocabulary and reading practice. In *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, pages 80–88. Association for Computational Linguistics.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *ACL (1)*, pages 1537–1546.

Kincaid, J. P., Fishburne Jr, R. P., Rogers, R. L., and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Technical Training Command Millington TN Research Branch.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Miltsakaki, E. and Troutt, A. (2007). Read-x: Automatic evaluation of reading difficulty of web text. In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, pages 7280–7286. Association for the Advancement of Computing in Education (AACE).

Ott, N. and Meurers, D. (2011). Information retrieval for education: Making search engines language aware. *Themes in Science and Technology Education*, 3(1-2):9–30.

Passonneau, R., Hemat, L., Plante, J., and Sheehan, K. M. (2002). Electronic sources as input to gre® reading

[9]The k value was set as 10.

comprehension item development: Sourcefinder prototype evaluation. *ETS Research Report Series*, 2002(1).

Pilán, I., Volodina, E., and Zesch, T. (2016). Predicting proficiency levels in learner writings by transferring a linguistic complexity model from expert-written coursebooks. In *COLING*, pages 2101–2111.

Rott, S. (1999). The effect of exposure frequency on intermediate language learners incidental vocabulary acquisition and retention through reading. *Studies in second language acquisition*, 21(4):589–619.

Sheehan, K. M., Kostin, I., and Futagi, Y. (2007). Sourcefinder: A construct-driven approach for locating appropriately targeted reading comprehension source texts. In *Workshop on Speech and Language Technology in Education*.

Vajjala, S. and Meurers, D. (2013). On the applicability of readability models to web texts. In *Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 59–68.

Verhelst, N., Van Avermaet, P., Takala, S., Figueras, N., and North, B. (2009). *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press.

Wisniewski, K., Schöne, K., Nicolas, L., Vettori, C., Boyd, A., Meurers, D., Abel, A., and Hana, J. (2013). Merlin: An online trilingual learner corpus empirically grounding the european reference levels in authentic learner data. In *ICT for Language Learning 2013, Conference Proceedings, Florence, Italy. Libreriauniversitaria. it Edizioni*.

Xia, M., Kochmar, E., and Briscoe, E. (2016). Text readability assessment for second language learners.

Zilio, L. and Fairon, C. (2017). Adaptive system for language learning. In *Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on*, pages 47–49. IEEE.

Zilio, L., Wilkens, R., and Fairon, C. (2017a). Enhancing grammatical structures in web-based texts. In *Proceedings of the 25th EUROCALL*, pages 839–846. Accepted.

Zilio, L., Wilkens, R., and Fairon, C. (2017b). Using nlp for enhancing second language acquisition. In *Proceedings of Recent Advances in Natural Language Processing*, pages 839–846.

9. Language Resource References

Geertzen, J., Alexopoulou, T., and Korhonen, A. (2013). Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*.

Towards a Diagnosis of Textual Difficulties for Children with Dyslexia

Solen Quiniou, Béatrice Daille
Université de Nantes, LS2N
2 rue de la Houssinière, 44322 Nantes Cedex 3, France
{Solen.Quiniou, Beatrice.Daille}@univ-nantes.fr

Abstract

Children's books are generally designed for children of a certain age group. For underage children or children with reading disorders, like dyslexia, there may be passages of the books that are difficult to understand. This can be due to words not known in the vocabulary of underage children, to words made of complex subparts (to pronounce, for example), or to the presence of anaphoras that have to be resolved by the children during the reading. In this paper, we present a study on diagnosing the difficulties appearing in French children's books. We are more particularly interested on the difficulties coming from pronouns that can disrupt the story comprehension for children with dyslexia and we focus on the subject pronouns *"il"* and *"elle"* (corresponding to the pronoun *"it"*). We automatically identify the pleonastic pronouns (*e.g.*, in *"it's raining"*) and the pronominal anaphoras, as well as the referents of the pronominal anaphoras. We also detect difficult anaphoras that are more likely to lead to miscomprehension from the children: this is the first step to diagnose the textual difficulties of children's books. We evaluate our approach on several French children's books that were manually annotated by a speech therapist. Our first results show that we are able to detect half of the difficult anaphorical pronouns.
Keywords: anaphora, dyslexia, children's book, French

1. Introduction

The democratization of books on digital tablets has allowed the design of new methods to support people with reading troubles. Children's book publishers have thus proposed adaptations for young readers, with the use of specific typefaces, larger margins and text spaces, or refined illustrations to help them to better understand the content of the text. These adaptations have also proven their efficiency on Web browsers as they have allowed readers with dyslexia to better use their short-term memory on the text and on its meaning (Parilova et al., 2016). Research works have been mainly focused on using audio interfaces, as speech dictation, and screen readers to offer new features in digital books addressed to readers with dyslexia (Sitbon et al., 2007) rather than on supporting the difficulties coming from the content of the text itself.

Our work is complementary of the previous approaches as it only considers the content of the text. Indeed, we are interested in evaluating the difficulties that occur in a text, in the form of textual ambiguities, and that can lead to difficulties in the comprehension of the story, especially for children with reading troubles like dyslexia. There are ambiguities at various levels: at the phonetic level such as the French word *"fils"* that can be translated as *"son"* or *"threads"* depending on the context and which is either a singular noun, or a plural noun (furthermore, its pronunciation is different in both cases); at the lexical level, *e.g.* due to collocations like *"il pleut des cordes"* that translates into *"it's raining cats and dogs"* (whereas *"cordes"* usually translates into *"ropes"*); at the pragmatical level as in *"il lui parle"* (*"he talks to him/her"*) where *"il"* is an anaphorical pronoun which referent has to be found between the preceding masculine nominal groups and *"lui"* is also an anaphorical pronoun which referent can be either a masculine, or a feminine nominal group.

To our knowledge, in the field of natural language processing, the difficulties coming from dyslexia have been studied in the writing of children with dyslexia (Rello et al., 2016; Rauschenberger et al., 2016) but not in their reading. Evaluating the difficulty of a text is a field of natural language processing that has been recently studied (François and Watrin, 2011; Gala et al., 2014; Ho Dac et al., 2016; Tanguy et al., 2016; Müller et al., 2016). Nonetheless, these works rather focused on predicting the lexical complexity and the graduation of words in a text whereas we are more interested in establishing a diagnosis of the difficulties appearing in a text but also to propose an explanation for the identified difficulties. Indeed, our goal is to allow the visualization of the difficulties on digital children's books and also to be able to visualize the explanations of the detected difficulties.

2. Reading Difficulties of Children with Dyslexia

According to the French dictionary of speech therapy (Brin-Henry et al., 2011), dyslexia is " the term used to name all the specific and durable troubles that manifest themselves when a person (child or adult) identifies written words during reading. Dyslexic troubles persist throughout the life course of the person. The World Health Organization estimates that 8 % to 12 % of the world's population is affected by these dyslexic troubles. "

In practice, dyslexia can express itself in various ways: omission, substitution, sound inversion in words; confusion between mirror letters (*"d/b"* and *"p/q"*) and between close sounds (*"ch/j"* and *"d/t"*); decoding difficulty; guessing words relying on their first letters or on the meaning of the sentence; difficulties to recog-

Book title	Code	Age group	#Words	#Personal pronouns	#*"il"*	#*"elle"*
Ali Baba et les quarante voleurs	ALI	8-12	2 458	258	80	29
L'arbre et le bûcheron	ARB	8	1 517	161	42	2
Le buveur d'encre	BUV	7	1 002	161	34	2
Dans le ventre du cheval de Troie	CHE	9	2 623	266	50	11
Emporté par le vent	EPV	12-15	3 005	253	30	142
Nos étoiles contraires	NEC	14	35 484	3 055	595	234

Table 1: Presentation of the children's books in the corpus

nize words (small lexical stock); difficulties for irregular words; skipping words or lines; comprehension difficulties due to the attention focused on the decoding; misreading a word for another one (paralexia).

In this work, we focus on ambiguities caused by pronouns, and especially anaphoras (Botley and McEnery, 2000). For children with dyslexia, they can lead to comprehension difficulties in several situations:

- in the French language, the pronoun *"il"* can be either pleonastic, or anaphorical. When the pronoun is pleonastic, if the children do not detect it, they will be looking for a referent that does not exist;

- the pronouns *"lui"* and *"leur(s)"* (corresponding to *"her/him/it"* and *"their"*) can be either feminine, or masculine. For example, in the sentence *"Il lui(leur) laissa la vie sauve"* (*"He spared his/her(their) life(s)"*), we do not know if the referent of *"lui(leur)"* is feminine or masculine;

- when the referent is located several sentences before the pronoun, the children will have trouble to find it. For example, in ALI (see Section 3.), we have the following translated sentences: *"Ali Baba is happy for his brother. Good thing that Cassim is married to a rich heir! Good thing that he is an important merchant of the city! But too bad that Cassim does not want to see him anymore."*. In the last sentence, *"him"* refers to *"Ali Baba"* but it appears 3 sentences before this pronoun;

- in dialogues, the pronouns *"je"* and *"tu"* (*"I"* and *"you"*) do not always refer to the same person as it depends on the person who is talking. It makes it hard for children with dyslexia to follow the dialogue, especially in the case of long dialogues;

- words like *"en"* or *"le/la/les"* can be either pronouns, or prepositions and determiners: it makes it hard to understand them. For example, in *"il se déplaçait en silence"* (*"he was moving silently"*), *"en"* is a preposition, whereas in *"l'eau coule ; il y en a partout"* (*"water flows; it is everywhere"*), *"en"* is a pronoun;

- when there is an anaphorical chain, *i.e.* a sequence of pronouns corresponding to the same referent, it might be helpful to annotate the first pronoun because it is more difficult to retrieve than the following ones. For example, in ALI, we have the following translated anaphorical chain: *"Still trembling, Ali Baba comes down the tree. He knows that he must quickly leave this place! He should not get involved in this! This is too dangerous! He is only a modest logger!"*

- when the referent is located after the pronoun (*i.e.* in a cataphora), finding the referent will be harder for the children. For example, in ALI, we have the following translated cataphora: *"Cassim's wife goes in the kitchen to pick up the jar herself. When she gets back home, Ali Baba's wife plunges the jar in the gold coins."* Here, *"she"* refers to *"Ali Baba's wife"* and not to *"Cassim's wife"*;

- when several clitics pronouns appear between the subject pronoun and the verb, it is difficult to identify the referent of each pronoun. For example, in ARB, we have *"je te l'ai dit"* (*"I told it to you"*) where *"te"* and *"l'"* are two pronouns.

3. Corpus

The raw corpus is first described and the manual annotation process is then presented.

3.1. Corpus Used in this Study

The corpus is provided by Mobidys[1], a startup that publishes digital books for children with dyslexia. The corpus gathers full-texts or extracts of French children's books (some of them have been adapted to children with dyslexia by rewriting the text of the books). Table 1 gives the corpus features: the age group to which the books are addressed, the number of words, the number of personal pronouns, and the number of *"il"* and *"elle"* pronouns. We can see that the proportion of personal pronouns is different between the books and goes from 8.4% (for EPV) to 16.1% (for BUV): books addressed to younger children contain a higher proportion of personal pronouns as compared

[1] www.mobidys.fr

to books addressed to teenagers. This gives an indication on the textual difficulty: a text with a higher proportion of pronouns is more likely to contain a higher number of ambiguities and thus to be more difficult to understand for children with dyslexia. Furthermore, *"il"* pronouns represent 11.9% to 31.0% of the personal pronouns whereas *"elle"* pronouns represent 1.2% to 56.1% of them: they thus represent about one third of the personal pronouns in the corpus.

3.2. Corpus Annotation

To evaluate the detection of pleonastic pronouns and anaphoras, a speech therapist student (in her fourth year of study) annotated the occurrences of *"il"* pleonastic pronouns, and of *"il"* and *"elle"* pronominal anaphoras with their referents. She also judged the difficulty of the pronominal anaphoras. Table 2 summarizes these numbers for each book[2] (the rate of difficult anaphoras is computed according to the number of anaphorical pronouns).

Text	#Pleonastic pronouns	#Anaph. pronouns	#Difficult anaphoras
ALI	7	102	10 (9.8%)
ARB	9	35	9 (25.7%)
BUV	10	26	7 (26.9%)
CHE	20	41	2 (4.9%)
EPV	8	164	18 (11.0%)
NEC	17	85	7 (8.2%)

Table 2: Number of manual annotations of *"il"* and *"elle"* pronouns in the corpus

Anaphorical pronouns represent the majority of the *"il"* and *"elle"* pronouns (only *"il"* pronouns can be pleonastic). In ARB and BUV, a quarter of the anaphoras were considered difficult whereas the books are addressed to young children (7-8 year olds).

4. Pronoun Detection and Resolution

Difficult anaphora detection consists in three main steps: the pleonastic pronoun detection, the anaphora resolution and, the difficult anaphora diagnosis.

4.1. Pleonastic Pronoun Detection

In the French language, the *"il"* pronoun is either a subject pronoun, or a pleonastic pronoun. As the anaphora resolution only applies to argument pronouns, it is necessary to distinguish between the two types of *"il"* pronouns. To do so, we define rules over sequences of tokens with their grammatical features, each rule being in the form of a regular expression. Specific components were developed to deal with regular expressions on annotations, such as TokenRegex(Chang and Manning, 2014). We use PyRATA[3], available for Python, which uses part-of-speeches, inflectional forms, and lemmas in the regular expressions. From the set of rules written by Danlos (2005) to disambiguate the *"il"* pronouns, we kept 15 rules: those exclusively describing a pleonastic use, those including typical collocations, and a subpart of those describing current pleonastic sentence structures with an extraposed nominal subject. Here are examples of rules, for each category:

```
(1) word="il" word="ne"? lemme@"meteo"
(2) word="il" lemme="ne"? lemme="y"
    lemme="en"? lemme="avoir"
(3) word="il" word="ne"? pos="PRO:PER"?
    lemme="avoir"? word="pas"? lemme="manquer"
```

Rule (1) recognizes pleonastic sentences with weather verbs belonging to the meteorological semantic class, such as *"il neige"* (*"it snows"*). Rule (2) describes the typical collocation *"il y a"* (*"there is"*). Rule (3) expresses the pleonastic sentence structure where the subject of the verb *"manquer"* (*"to lack"*) occurs as a direct object and the subject is the pleonastic pronoun, *"il manque du pain"* (*"there is a bread shortage"*). We do not take into account Danlos rules belonging to the formal language register as it does not characterize children's book. Furthermore, rule (3) is ambiguous because it can miscategorize the *"il"* pronoun as pleonastic when it is anaphorical. Indeed, in *"il manque une évaluation "* (*"an evaluation is missing"*), *"il"* is pleonastic, but in *"il manque de confiance en lui"* (*"he lacks self-confidence"*), *"il"* is anaphorical. We tag these ambiguous structures as UNDETERMINED as they are likely to confuse children with dyslexia.

4.2. Anaphora Resolution

To perform pronominal anaphora resolution, we apply the knowledge-poor approach of Mitkov (2002): it only requires part-of speech tagging and chunk identification as the linguistic preprocessing. The algorithm identifies the nominal chunks that precede an anaphorical pronoun, within a distance of two sentences, then checks the inflectional agreement with the anaphora and finally applies indicators to rank the nominal chunks. Each indicator gives either a positive or a negative score. The nominal chunk with the highest combined score is chosen as the antecedent of the anaphorical pronoun. We use RDRPOSTagger (Nguyen et al., 2014) to extract the nominal chunks. This tagger is designed for French and gives the gender and the number of a word: these informations are used for the inflectional agreement part of the anaphora resolution.
Mitkov (2002) listed 10 indicators. We kept 6 indicators as such: definiteness, givenness, lexical reiteration, non prepositional noun phrases, collocation pattern preference, and referential distance. We adapted two other indicators, *i.e.* section heading preference

[2]Due to a lack of time, only the first pronouns of NEC were annotated. The annotation of the remaining pronouns is in progress.

[3]https://github.com/nicolashernandez/PyRATA

and domain terminology preference, to the type of our texts. To represent the section heading preference, we only used the book title as there is no section. To take into account the domain terminology preference, we used the list of characters of the book. This list has to be manually supplied. We assign a higher score to main characters than to secondary or peripheral characters. We removed two indicators, indicating verbs and immediate reference, as they did not fit with linguistic features of children's books. We apply the eight indicators on each nominal chunks and we only keep the fourth first chunks with a combined score greater than two. When no antecedent is found, we assign the antecedents of the previous anaphorical pronoun to this anaphorical pronoun.

4.3. Difficult Anaphora Diagnosis

We consider as difficult anaphoras, the following anaphorical pronouns:

- pronouns with a distance of more than two sentences with the antecedent;
- pronouns with a high number of antecedents, *i.e.* three ore more;
- pronouns at the start of an anaphorical chain;
- pronouns belonging to an anaphorical chain of five pronouns or more;
- pronouns with several character names among the antecedents.

Nonetheless, difficulties due to anaphorical chains were not always tagged as such in the reference corpus. For example, in ALI, there is an anaphorical chain of seven anaphorical pronouns but none of them were considered to be difficult.

5. Experimental Results

The results of our experiments on the 3 parts of the pronoun detection and resolution are given in Table 3 with respect to the manual annotation of the corpora. They are discussed in the following subsections.

Text	Impers. pronouns		Resolved pronouns		Difficult anaphoras	
	REC	**PR**	**REC**	**PR**	**REC**	**PR**
ALI	85.7%	100%	88.2%	89.1%	60.0%	16.7%
ARB	77.8%	100%	74.3%	74.3%	33.3%	17.6%
BUV	90.0%	100%	73.1%	73.1%	42.9%	60.0%
CHE	90.0%	100%	68.3%	68.3%	50.0%	5.0%
EPV	25.0%	100%	72.6%	71.3%	27.8%	23.8%
NEC	52.9%	100%	49.4%	47.2%	42.9%	6.0%

Table 3: Recall (REC) and precision (PR) on the pronoun detection and resolution in the corpus

5.1. Pleonastic *"il"* Detection

For all the texts, the precision reaches 100%. For the first four texts, the recall can reach 100% if we consider the UNDETERMINED pronouns to be pleonastic ones. For EPV, the recall goes to 62.5% by adding the UNDETERMINED pronouns. The remaining pleonastic pronouns correspond to pronouns used with the verbs *"pouvoir"* (*"can"*) and *"être"* (*"be"*). These verbs are more often used with anaphorical pronouns than with pleonastic ones. Furthermore, the negative form does not appear in the regular expressions: considering it would allow an additional increase in the recall.

5.2. Anaphora Resolution

First results are quite satisfactory, except for NEC. This is mainly due to anaphorical chains because if the referent of the first pronoun is wrong, the mistake will be propagated to the other pronouns of the anaphorical chain. Another source of mistakes occur when the referent is not present in one of the two preceding sentences but in a sentence before these two sentences. The last source of mistakes comes from the POS tagger which wrongly tags some chunks as nominal ones and thus allows them to be candidate referents. For example, in NEC, 20% of the anaphorical *"il"* referents are not nominal chunks: it includes *"ai pardonné"* (*"has forgiven"*), *"allume"* (*"turn on"*), or *"musclé"* (*"strong"*).

5.3. Difficult Anaphora Diagnosis

The recall of difficult anaphoras is better than the precision which is quite low. Our approach tends to over detect difficult anaphoras. It is also due, in part, to the POS tagger and the misdetection of nominal chunks (or the misdetection of the frontiers of the chunks). Indeed, each time a pronoun corresponds to an unseen nominal chunk, this pronoun is considered to be difficult. For example, in ARB, *"l'arbre"* (*"the tree"*) and *"à l'arbre"* (*"to the tree"*) are 2 detected nominal chunks: the second one should be just *"l'arbre"* and not a new referent that would be also further detected as a difficult anaphora.

6. Conclusion

In this paper, we have presented a first attempt to diagnose difficulties in children's books. Our work focused on distinguishing pleonastic pronouns from anaphorical pronouns, and on recovering the referents of anaphorical pronouns for *"il"* and *"elle"* pronouns (corresponding to the *"it"* pronoun). We also proposed a first attempt at identifying difficult anaphoras, *i.e.* anaphoras that are more likely to cause difficulties in the comprehension of children with dyslexia.

Currently, we are designing an experimental evaluation on the detection of difficult anaphoras with children suffering from dyslexia as well as with other children. In future works, we want to extend the diagnosis to other pronouns as well as to the vocabulary used in the text (as compared to children age groups) and to the detection of words with complex subparts to read.

7. Acknowledgements

This work is part of the AmbiDYS project which has been funded by the RFI OIC (Région Pays de la Loire). We would also like to thank Kevin Espasa and David Kerbrat who worked on this project as interns as well as the Mobidys startup which provided the children's books we worked on.

8. Bibliographical References

Botley, S. P. and McEnery, T. (2000). *Corpus-based and Computational Approaches to Discourse Anaphora*. John Benjamins Publishing Company.

Brin-Henry, F., Courrier, C., Lederlé, E., and Masy, V. (2011). *Dictionnaire d'orthophnie - 3e édition*. Ortho Edition.

Chang, A. X. and Manning, C. D. (2014). TokensRegex: Defining cascaded regular expressions over tokens. Technical Report CSTR 2014-02, Department of Computer Science, Stanford University.

Danlos, L. (2005). Ilimp: Outil pour repérer les occurrences du pronom impersonnel il. In *Actes de TALN*, pages 123–132.

François, T. and Watrin, P. (2011). Quel apport des unités polylexicales dans une formule de lisibilité pour le français langue étrangère ?

Gala, N., François, T., Bernhard, D., and Fairon, C. (2014). Un modèle pour prédire la complexité lexicale et graduer les mots.

Ho Dac, L. M., Muller, S., and Delbar, V. (2016). L'anticorrecteur : outil d'évaluation positive de l'orthographe et de la grammaire.

Mitkov, R. (2002). *Anaphora resolution*. Longman.

Müller, A., François, T., Roekhaut, S., and Fairon, C. (2016). Classification automatique de dictées selon leur niveau de difficulté de compréhension et orthographique.

Nguyen, D. Q., Nguyen, D. Q., Pham, D. D., and Pham, S. B. (2014). RDRPOSTagger: A Ripple Down Rules-based Part-Of-Speech Tagger. In *Proceedings of the Demonstrations at EACL*, pages 17–20.

Parilova, T., Mrvan, B., Mizi, B., and Hldka, E. (2016). Emerging technology enabling dyslexia users to read and perceive written text correctly.

Rauschenberger, M., Rello, L., Füchsel, S., and Thomaschewski, J. (2016). A language resource of german errors written by children with dyslexia. In *Proc. of LREC*.

Rello, L., Baeza-Yates, R., and Llisterri, J. (2016). DysList: An annotated resource of dyslexic errors. In *Proc. of LREC*.

Sitbon, L., Bellot, P., and Blache, P. (2007). éléments pour adapter les systèmes de recherche d'information aux dyslexiques. *Revue TAL*, 48(3):1–26.

Tanguy, L., Fabre, C., and Mercier, C. (2016). Analyse d'une tâche de substitution lexicale : quelles sont les sources de difficultés ?

Coreference Resolution in FreeLing 4.0

Montserrat Marimon*, Lluís Padró†, Jordi Turmo†
*Universitat Pompeu Fabra. Barcelona, Spain.
montserrat.marimon@upf.edu

†TALP Research Center.
Universitat Politècnica de Catalunya. Barcelona, Spain.
{padro,turmo}@cs.upc.edu

Abstract

This paper presents the integration of RelaxCor into FreeLing. RelaxCor is a coreference resolution system based on constraint satisfaction that ranked second in the CoNLL-2011 shared task. FreeLing is an open-source library for NLP with more than fifteen years of existence and a widespread user community. We present the difficulties found in porting RelaxCor from a shared task scenario to a production enviroment, as well as the solutions devised. We present two strategies for this integration and a rough evaluation of the obtained results.

Keywords: FreeLing, Coreference Resolution, CoNLL-2011, relaxation labeling

1. Introduction

FreeLing[1] is an open-source multilingual language processing library providing a wide range of analysis functionalities for several languages.

The project is conceived as a library that can be called from a user application in need of analysis services. The software is open-source, distributed under an GNU Affero General Public License[2], and dual-licensed to companies that embed it in their commercial products or online services.

The open-source approach has been very fruitful during the fifteen years of life of the project (the first version was released on 2003). The amount of accumulated downloads during this time is over 200,000. Contributions from the user community combined with the increasing availability of open source language resources has made it possible to extend the number of supported languages from three (English, Spanish and Catalan) to fourteen (adding German, French, Italian, Portuguese, Russian, Norwegian, Asturian, Welsh, Galician, Croatian and Slovene).

FreeLing offers a wide variety of language processing modules, though not all modules are available for all languages. Most relevant modules are: Language identification, tokenization and sentence splitting, lemmatization, date/time detection, numbers detection, multiword expressions detection, physical magnitudes detection, named entity recognition and classification, PoS tagging, SAMPA phonetic encoding, word sense disambiguation, shallow parsing, constituency parsing, dependency parsing, semantic role labelling, coreference resolution, semantic graph extraction, and document summarization.

More details about the FreeLing project can be found in (Padró and Stanilovsky, 2012) and in the online documentation in the project website.

One remarkable extension in version 4.0 was the inclusion of a Coreference Resolution module, based on RelaxCor, the second-ranked system in CoNLL-2011 shared task (Sapena et al., 2011).

However, academic shared tasks have a very specific scenario, which does not necessarily match the real-world settings in which a system like FreeLing is required to operate. Thus, considerable efforts must be devoted to the integration of a module successful in the laboratory, such as RelaxCor, in a production pipeline.

In this paper we describe how this coreference resolution module was integrated in FreeLing (for English and Spanish), as well as the encountered obstacles and solutions devised. We also present an alternative configuration for RelaxCor using hand-written constraints instead of the automatically learnt constraints used in CoNLL shared task.

The next section briefly summarizes related work. Section 3. overviews the basic idea behind RelaxCor and the main difficulties presented by its integration in FreeLing. Sections 4. and 5. describe an alternative set of hand-written constraints and compare its performance with the machine-learned model. Finally, Section 6. concludes.

2. Related Work

There are several open-source suites other than FreeLing that offer state-of-the-art level NLP functionalities. The most remarkable, for being open-source, widely used and offering a set of functionalities comparable to FreeLing are: Stanford CoreNLP (Manning et al., 2014), Apache OpenNLP[3], NLTK (Bird et al., 2009) and IXA Pipes (Agerri et al., 2014).

There are also other systems of NLP-related software, such as GATE or UIMA, which are not language analysis pipelines themselves, but architectures or frameworks to integrate existing components.

The above mentioned suites largely differ with respect to the used programming language, offered APIs, processing speed, supported languages, customization or retraining capabilities, whether they are more developer-oriented or end-

[1] http://nlp.lsi.upc.edu/freeling
[2] http://www.gnu.org/copyleft/agpl.html
[3] https://opennlp.apache.org

user oriented, etc. Thus, a detailed comparison is out of the scope of this paper.

Regarding coreference resolution, Stanford CoreNLP includes an updated version of the first-ranked system in CoNLL 2011 shared tasks, supporting English and Chinese. Apache OpenNLP offers a basic support for English. IXA Pipes do not ship a coreference resolution module out-of-the-box, but third-party provided modules are available for Spanish and English. Finally, the latest FreeLing version offers coreferences for Spanish and English.

The first attempt to establish a common evaluation framework for coreference systems was carried out in SemEval 2010 (Recasens et al., 2009), which offered data sets for 6 languages (including English and Spanish). Later, CoNLL-2011 and CoNLL-2012 shared tasks (Pradhan et al., 2011; Pradhan et al., 2012) proposed similar tasks that have been a reference framework since then. The 2011 edition included only English, and was won by Stanford rule-based system (Lee et al., 2013). The 2012 edition included English, Arabic, and Chinese, and was won by a neural network based system (Fernandes et al., 2012), which has been the main trend in the state of the art since then.

3. Integration of RelaxCor into FreeLing

3.1. RelaxCor

RelaxCor is a coreference resolution system based on constraint satisfaction. The coreference resolution problem is represented as a graph with mentions in the vertices which are connected to each other by edges. Edges are assigned a weight that indicates the confidence whether the mention pair corefers or not. More specifically, an edge weight is the sum of the weights of the constraints that apply to that mention pair.

The knowledge used by the system is encoded in constraints, each of which has a confidence score. The larger the score absolute value, the more reliable the constraint is and the stronger effect when applied. The sign of the constraint confidence score indicates whether a pair or a group of mentions may corefer (positive) or not (negative). Only constraints over pairs of mentions are used in the current version of RelaxCor, though the model can handle higher-order constraints. Constraints and their confidence scores can be obtained from any source, including manual encoding or automatic acquisition from a training corpus.

Figure 3.1. shows (a simplified version of) the graph corresponding to the text:

> *FC Barcelona president Joan Laporta has warned Chelsea off star striker Leonel Messi. Aware of Chelsea owner Roman Abramovich's interest in the young Argentine, Laporta said last night: "I will answer as always, Messi is not for sale and we do not want to let him go."*

Constraints are applied to every pair of mentions in the text, and a compatibility score for that pair is computed (represented by edges in the graph). Many pairs remain unconnected if constraints find no evidence neither for nor against joining them. The algorithm will partition the graph, keeping together pairs with high compatibility and setting apart nodes with negative scores.

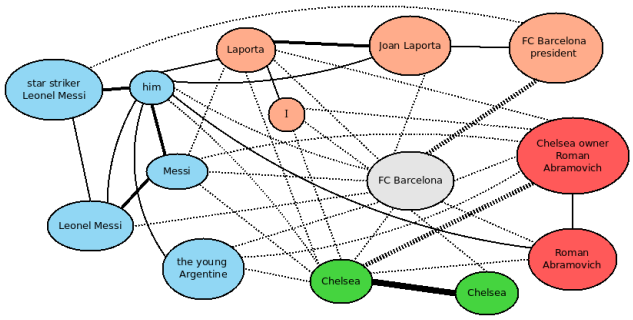


Figure 1: Graf produced from example text. Dotted lines represent negative compatibility scores. Solid lines represent positive scores. Line thickness is proportional to score absolute value. Colors represent the final groups created.

For instance, constraints assign large negative value to nodes corresponding to named entities of different classes (person and organization entities in this example), and large positive values to nodes of the same kind with similar names (e.g. Both mentions of *Chelsea*, or *Messi* and *Leonel Messi*). There are also nodes such as *him* that are compatible with many mentions, and the final decision depends both on the strength of their connections, and the conections among their neighbors (e.g. *him* not only has stronger connections with *Messi* and *star striker Leonel Messi*, but they both also share a strong connection with *Leonel Messi*, which reinforces all of them gluing together).

RelaxCor uses relaxation labeling for the resolution process. Relaxation labeling is an iterative algorithm that performs function optimization based on local information. It has been widely used to solve a variety of NLP problems. A vector of probability values is maintained for each vertex/mention. Each vector element corresponds to the probability that the mention belongs to a specific entity among the potential entities in the document. During the resolution process, the probability values are updated according to the edge weights and probability vectors of the neighboring vertices. The larger the edge weight, the stronger the influence exerted by the neighboring distributions. The process stops when there are no more changes in the probability vectors.

The relaxation labeling approach combines mention pair classification and linking in one step. Thus, decisions are taken considering the entire set of mentions, which ensures consistency and avoids local classification decisions.

Mentions are modeled as feature sets, and contain a variety of information (gender, number, person, PoS, sense, etc). Constraints use these features to establish a compatibility value among pairs of mentions (e.g. Two mentions with different gender will have a negative compatibility. Mentions that are pronouns and have the same person, gender, and number will have a positive compatibility, etc.). Note that semantic and syntactic information can also be used in the constraints.

RelaxCor was first proposed in SemEval-2010 (Sapena et al., 2010), and an improved version (Sapena et al., 2011) ranked second in CoNLL-2011. Extended details on how

RelaxCor works can be found in (Sapena et al., 2013).
We selected RelaxCor as the coreference resolution module to be integrated in FreeLing for the following reasons

- It was developed in UPC, so FreeLing team has a deep understanding of the algorithm and the code, which eases the portability. Also, the underlying algorithm is Relaxation Labelling which was already part of FreeLing, since it is used by one of the PoS taggers.
- RelaxCor uses a very general approach: The problem is modeled in terms of graph vertices encoded as feature sets and constraints among them, which makes it easy to adapt the code for new languages or even for new tasks.
- The constraint-based approach allows the addition of new languages with a relatively low cost: If no training data is available to use machine learning techniques, constraint can be hand-encoded as presented in this paper.

3.2. Integration Issues

Coreference Resolution is a complex NLP tasks, since it requires a lot of information from previous analysis steps: Tagging and parsing are required to identify candidate mentions and to discover syntactic relations among them that are relevant to the task (e.g. appositions, relative clauses, etc.). Named Entity detection and classification is also crucial to establish whether two names may refer to the same entity. Word sense disambiguation and Semantic Role Labelling also provide relevant information in some cases.
This large amount of dependences largely increases the difficulty of integrating a module developed in one specific scenario into a production pipeline, each with its own settings and dependences.
Some relevant issues that we had to take into account are:

- The input data in CoNLL-2011 shared task follows tokenization conventions that not necessarily match those used by FreeLing.
- The input data in CoNLL-2011 shared task uses a PoS tagset that has some differences with that used by FreeLing.
- The input data in CoNLL-2011 shared task contains a gold constituency parse which can be used to detect mentions. The module integrated in FreeLing will have to resort to the output of its own parser, which not only will contain errors, but also uses different labels and syntactic structures.
- The constituency parser in FreeLing is rule based and uses a simple strategy. Thus, it does not perform at the same accuracy level as a statistical parser.
- The input data in CoNLL-2011 shared task contains speaker information in some dialog documents, but FreeLing has no dialog or speaker detection module.

To tackle with these issues, we took the following integration decisions:

- To build two mention detectors:
  - Another one based on dependency trees, that would use the output of FreeLing statistical dependency parser, which offers more robust and accurate results. This detector follows FreeLing linguistic team criteria to establish what a candidate mention is, which are not necessarily the same followed in CoNLL shared tasks.[4]
- To adapt the mention feature extraction, as well as the syntactic checks required by some constraints, to FreeLing PoS tagset and syntactic labels and structures. Note that this had to be done twice: for constituency trees and for dependency trees.
- To taylor the train/development/test corpus to match FreeLing tokenization criteria. We ported the coreference annotations to the corpus retokenized according to FreeLing criteria, and we excluded from the corpus those documents where unsolvable retokenization clashes prevented the safe mapping of the gold annotations.
- To exclude dialog documents with speaker information from the corpus.

4. Coreference Resolution in FreeLing

As mentioned in section 3.2. we integrated in FreeLing two versions of RelaxCor: One using constituency parsing to detect mentions and to extract syntactic information and the other using dependency parsing.

4.1. Constituency parsing version

The constituency parsing based module consists of a straightforward translation of the original RelaxCor from Perl to C++, adapting PoS tagset, constituent labels and syntactic structures. The goal was to have a FreeLing module that could use machine learning models learned on the CoNLL shared task, to avoid costly re-training and parameter-tuning procedures. Given differences between the criteria used in the training corpus and the output of FreeLing preprocessing stages, we expect this module to perform worse once integrated in FreeLing than it did in the shared task, so we will consider it as a baseline.

4.2. Dependency parsing version

The dependency parsing based module uses more accurate parsing trees, raising the quality of mention detector and syntactic information used in constraints. Since constituent information is not available here, originally trained models can not be used. One option would be retraining the module using a corpus adapted to the new criteria, which would require a costly re-annotation effort. Thus, we opted by developing a set of hand written constraints, inspired on the approach proposed in (Lee et al., 2013), the winner system in CoNLL-2011, that defines ten sieves of decreasing precision rules, applied in a cascaded schema.

[4]Our mention detector considers a mention the span of any subtree headed by a noun, a personal pronoun, or a relative pronoun, except those where the head is the word *what* or a temporal noun (*day*, *year*, *morning*, *minute*, etc.). No filtering of embedded mentions or pleonastic pronouns is performed.

In our case, relaxation labeling applies all constraints simultaneously, thus we need to emulate the cascading using compatibility scores of different ranges (e.g. more releable rules have scores around 50, while less precise constraints have scores about 5–10), so that higher-precision rules overweight any contradicting lower-precision rule. In addition, our model does not follow an entity-mention approach but a mention-pair approach, thus some sieves can be only approximated. Finally, we need to add some default rules that favor the creation of singletons in absence of strong enough coreference evidence, to avoid ending in trivial solutions where all mentions belong to the same group.

Next, we overview the sieves proposed by Stanford and present a few samples of the rules we encoded in each of them:

- **Sieve 1**: Mentions with the same speaker in dialog documents or when direct speech is used. We only deal with the later, since no speaker identification is available in FreeLing.
  Sample rules:
  +50 if both mentions are personal pronoun "I" and both are inside the object of the same reporting verb (*say*, *tell*, *ask*, etc.) or inside the same quotation.
  -25 if one mention is a personal pronoun and the other is not the same pronoun and both are inside the object of the same reporting verb, or inside the same quotation.

- **Sieves 2–3**: Mentions containing the same text (either the whole mention or up to the head word).
  Sample rules:
  +50 if both mentions are of the same type (named entity, pronoun, noun phrase), not nested in one another and their texts match completely.
  +25 if both mentions are of the same type and their texts match up to the head word.

- **Sieve 4**: Mentions appearing in specific constructions (e.g. relative clauses, appositions and predicative constructions).
  Sample rules:
  +50 if both mentions are in apposition.
  +50 if both mentions are in a predicative structure.
  +50 if one mention is a relative pronoun and has the other as syntactic antecedent.

- **Sieves 5–7**: Mentions that have the same head, or the head of one matches some word in the other.
  Sample rules:
  +25 if one mention is a named entity or noun phrase, the other is not a pronoun, they have the same head and their modifiers are compatible.

- **Sieves 8–9**: Mentions headed by the same proper name.
  Sample rules:
  +50 if both mentions are named entities and they have the same head and the same semantic class.

- **Sieve 10**: Pronominal coreference.
  Sample rules:
  +15 if one mention is a 3rd-person, non-relative pronoun, the other is a named entity or noun phrase, they have morphological agreement and they belong to compatible semantic classes (person, man, woman, non-person, organization, location).

Since the used features and rules are very general, we also evaluated their performance on Spanish, using the annotated corpus provided by SemEval-2010 Task 1 (Recasens et al., 2009). After adapting the lexicon and syntactic criteria used by the features and rules, we obtained a coreference resolution system with a performance comparable to our version for English.

5. Experiments and Results

In this section we present the experimental setting we used to roughly evaluate the results of the integration.

We evaluated two systems: (1) the original RelaxCor ported to C++, using the output of FreeLing constituency parser and the original model trained on CoNLL-2011 data, and (2) the alternative version, using the output of FreeLing dependency parser and hand-written rules. Both systems were also applied to Spanish (after adapting configuration files with relevant lexica, PoS tags, syntactic labels, etc).

The used data was a subset of CoNLL-2012, excluding documents containing dialogs or severe tokenization mismatches (accounting for about 16-17% of the documents). The original CoNLL-2012 test and development sections were used as development corpus for the hand-written rules, and the train section of CoNLL-2012 was used as test. See Table 1 for a summary of corpus sizes.

		#doc orig.	#doc	#sent	#tok
English	Devel.	639	493	6,090	101,957
	Test	2,273	1,952	18,637	348,831
Spanish	Devel.	308	261	1,079	29,285
	Test	875	719	2,615	77,749

Table 1: Sizes of the used development and test corpus. Column *#doc orig* shows the number of documents in the original CoNLL corpus. The other columns show the sizes of the corpus after filtering dialog documents and tokenizaton mismatches.

Table 2 shows the results for both developed versions computed using the latest version (v8.01) of official CoNLL-2012 scorer. Rows marked *MD* present mention detection scores. Other rows present different performance metrics, including CoNLL-2011 and 2012 official metrics (average of MUC, B^3, and CEAF-e).

Several issues must be taken into account when interpreting the results:

- The development and test corpus partitions are not the same used in CoNLL shared tasks. Moreover, dialog documents and documents where tokenization could not be mapped where excluded.

		Constituency parsing, ML constraints						Dependency parsing, hand-written constraints					
		English			Spanish			English			Spanish		
Corpus	Metric	R	P	F_1	R	P	F_1	R	P	F_1	R	P	F_1
Devel.	*MD*	*65.80*	*62.87*	*64.30*	*28.55*	*55.90*	*37.80*	*66.66*	*64.97*	*65.80*	*55.85*	*62.05*	*58.79*
	MUC	54.76	49.37	51.93	19.80	41.91	26.89	52.48	54.30	53.38	36.98	43.89	40.14
	B^3	41.12	28.06	33.36	15.60	42.53	22.83	36.88	45.58	40.77	34.81	44.72	39.15
	CEAF-m	39.18	37.45	38.29	24.12	47.22	31.93	46.60	45.42	46.00	45.07	50.07	47.44
	CEAF-e	28.54	33.89	30.98	22.80	38.58	28.66	43.42	35.62	39.13	46.73	45.81	46.27
	BLANC	42.69	28.20	32.01	10.34	31.65	15.59	40.44	48.36	42.63	27.40	36.30	31.06
	CoNLL	**41.47**	**37.10**	**38.75**	**19.40**	**41.00**	**26.12**	**44.26**	**45.16**	**44.42**	**39.50**	**44.80**	**41.85**
Test	*MD*	*57.57*	*54.25*	*55.86*	*28.07*	*52.73*	*36.64*	*59.81*	*57.35*	*58.56*	*59.08*	*60.64*	*59.85*
	MUC	43.46	38.48	40.82	19.46	39.89	26.16	45.67	46.64	46.15	38.99	42.57	40.70
	B^3	35.04	27.92	31.08	15.40	39.76	22.20	36.16	42.31	39.00	35.76	42.68	38.91
	CEAF-m	38.60	36.40	37.47	24.02	45.14	31.36	46.22	44.32	45.25	46.25	47.47	46.86
	CEAF-e	29.23	32.95	30.98	22.46	35.65	27.56	42.97	35.74	39.02	48.03	43.55	45.68
	BLANC	35.48	23.26	27.50	10.41	31.04	15.59	33.74	39.37	35.30	29.21	34.01	31.21
	CoNLL	**35.91**	**33.11**	**34.29**	**19.10**	**38.43**	**25.30**	**41.60**	**41.56**	**41.39**	**40.92**	**42.93**	**41.76**

Table 2: Results of both RelaxCor integration strategies for English and Spanish.

- The machine-learning version was trained on CoNLL-2011 data, and is being evaluated on CoNLL-2012.
- Criteria used in the hand-written rule version to define which mentions are considered singletons are different from those used in CoNLL data.
- Our hand-written rules mark some coreferences (e.g. relative pronouns or predicative noun phrases) that are not marked in CoNLL data.

For all this, results presented here can not be compared to the state-of-the-art in CoNLL shared tasks, but only taken as a self-contained evaluation.

From this point of view, results show that both English models get a similar score on mention detection. Regarding the final coreference score, the distance between the training and test scenarios severely hampers the accuracy of the machine-learned models, while the hand-written model, being developed and tested in the target corpus, obtains significantly higher scores.

When applying the models to Spanish, the machine-learned model suffers a big drop in both measures, as one would expect. But the manual model –more easily tunable–, obtains results for Spanish in the same range than English.

6. Conclusions and Further Work

We have presented two strategies to integrate RelaxCor –a coreference resolution system developed for CoNLL-2011 shared task– into FreeLing. We have discussed encountered problems and solutions undertaken. We used pre-trained machine-learned models, as well as hand-written constraints, and evaluated the results. Even the evaluation is not comparable to the state of the art given the differences in the used corpus and criteria, we believe that the provided modules will be useful to FreeLing users.

There are still open lines to pursue: The machine learning version could be retrained and tested on an adapted version of CoNLL-2012. Constraint compatibility scores for hand-written rules could be automatically assigned using a training corpus. The utility of the modules could be assessed via indirect evaluation. Finally, the hand-written model could be adapted to other languages in FreeLing.

7. Acknowledgements

This research was partially funded by the Spanish Government via project Graph-MED (TIN2016-77820-C3-3-R).

8. Bibliographical References

Agerri, R., Bermudez, J., and Rigau, G. (2014). IXA pipeline: Efficient and ready to use multilingual NLP tools. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, Reykjavik, Iceland, May.

Bird, S., Loper, E., and Ewan, K. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

Fernandes, E., dos Santos, C., and Milidiú, R. (2012). Latent structure perceptron with feature induction for unrestricted coreference resolution. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 41–48. Association for Computational Linguistics.

Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M., and Jurafsky, D. (2013). Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 4(39):885–916.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May. ELRA.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In

Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task, pages 1–27. Association for Computational Linguistics.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, CoNLL '12, pages 1–40, Stroudsburg, PA, USA. Association for Computational Linguistics.

Recasens, M., Martí, T., Taulé, M., Màrquez, L., and Sapena, E. (2009). Semeval-2010 task 1: Coreference resolution in multiple languages. *SEW-2009 Semantic Evaluations: Recent Achievements and Future Directions*.

Sapena, E., Padró, L., and Turmo, J. (2010). Relaxcor: A global relaxation labeling approach to coreference resolution. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 88–91. Association for Computational Linguistics.

Sapena, E., Padró, L., and Turmo, J. (2011). Relaxcor participation in conll shared task on coreference resolution. pages 35–39. Association for Computational Linguistics.

Sapena, E., Padró, L., and Turmo, J. (2013). A constraint-based hypergraph partitioning approach to coreference resolution. *Computational Linguistics*, 39(4):847–884, December.

BASHI: A Corpus of Wall Street Journal Articles Annotated with Bridging Links

Ina Rösiger
Institute for Natural Language Processing
University of Stuttgart, Germany

Abstract

This paper presents a corpus resource for the anaphoric phenomenon of bridging, named BASHI. The corpus consisting of 50 Wall Street Journal (WSJ) articles adds bridging anaphors and their antecedents to the other gold annotations that have been created as part of the OntoNotes project (Weischedel et al., 2011). Bridging anaphors are context-dependent expressions that do not refer to the same entity as their antecedent, but to a related entity. Bridging resolution is an under-researched area of NLP, where the lack of annotated training data makes the application of statistical models difficult. Thus, we believe that the corpus is a valuable resource for researchers interested in anaphoric phenomena going beyond coreference, as it can be combined with other corpora to create a larger corpus resource. The corpus contains 57,709 tokens and 459 bridging pairs and is available for download in an offset-based format and a CoNLL-12 style bridging column that can be merged with the other annotation layers in OntoNotes. The paper also reviews previous annotation efforts and different definitions of bridging and reports challenges with respect to the bridging annotation.

Keywords: Corpus Resource, Bridging, Anaphora, Wall Street Journal, OntoNotes, English

1. Introduction

Bridging is an anaphoric phenomenon where the interpretation of a bridging anaphor, sometimes also called associative anaphor (Hawkins, 1978), is based on the non-identical associated antecedent.

The associated NLP task of bridging resolution is about linking these anaphoric noun phrases and their antecedents, where both do not refer to the same referent, but are related in a way that is not explicitly stated. Bridging anaphors are thus discourse-new, but dependent on previous context.

(1) I went to a wedding last weekend. **The bride** was a friend of mine.[1].

(2) What is the book about? **The answer** isn't trivial.

One can think about bridging anaphors as expressions with an implicit argument, e.g. *the bride* (at a wedding) or *the answer* (to this question).

1.1. Motivation

Compared to coreference resolution, which has become one of the standard NLP tasks, with its own track at most NLP conferences, the progress in bridging resolution is much slower. The main issue for most researchers aiming to apply statistical algorithms to this task is the lack of training data. While coreference resolution has about 35,000 coreferent pairs in their standard benchmark dataset OntoNotes (taking into account the transitivity of coreference chains), most datasets for bridging commonly comprise around 400 - 600 pairs (of course, bridging anaphors are also much rarer than coreference anaphors). Note that a benchmark dataset for bridging has not yet been established. In order to tackle the lack of available training data, several smaller corpora could be combined to create a larger corpus resource, including the corpus presented in this paper. The ISNotes corpus (Markert et al., 2012) contains bridging annotations, with 633 bridging pairs. Grishina (2016) recently described a parallel corpus of German, English and Russian texts with 432 German bridging pairs that have been transferred to their English and Russian counterparts. The corpus has, to the best of our knowledge, not yet been made publicly available. One of the newest corpora is the GUM corpus (Zeldes, 2017), a corpus of 64,000 tokens annotated with bridging links and coarse-grained information status.

During the preparation of the camera-ready version of this paper, the first shared task on bridging resolution was announced[2]. As a data basis, the second release of the ARRAU corpus (first released in Poesio and Artstein (2008)) was used, which contains 5512 bridging pairs in three different domains: news text, dialogue and narrative text. This is, as far as we know, currently the biggest corpus resource containing bridging pairs. However, only a small subset of the annotated pairs contains truly anaphoric bridging anaphors, which is why annotated corpus resources like the one presented in our paper are still beneficial (c.f. Section 1.2. for the distinction between referential and lexical bridging).

The resolution of bridging links is important because it can help in tasks which use the concept of textual coherence, for example Barzilay and Lapata (2008)'s entity grid or Hearst (1994)'s text segmentation. They might also be of use in higher-level text understanding tasks such as textual entailment (Mirkin et al., 2010) or summarisation based on argument overlap (Kintsch and van Dijk, 1978; Fang and Teufel, 2014).

1.2. Bridging: One Term, Many Phenomena

Bridging has been studied in many theoretical studies (Clark, 1975; Hawkins, 1978; Hobbs et al., 1993; Asher and Lascarides, 1998) as well as in corpus and computational studies (Fraurud, 1990; Poesio et al., 1997; Vieira

[1] Anaphors are marked in bold face, their antecedents are underlined

[2] http://anawiki.essex.ac.uk/dali/crac18/crac18_shared_task.html

and Teufel, 1997; Poesio and Vieira, 1998; Poesio et al., 2004; Nissim et al., 2004; Nedoluzhko et al., 2009; Lassalle and Denis, 2011; Baumann and Riester, 2012; Cahill and Riester, 2012; Markert et al., 2012; Hou et al., 2013b; Hou et al., 2013a; Hou, 2016; Zikánová et al., 2015; Grishina, 2016; Roitberg and Nedoluzhko, 2016; Riester and Baumann, 2017).

One big issue is that, unlike in work on coreference resolution, these studies do not follow an agreed upon definition of bridging. On the contrary, many different phenomena have been described as bridging. As a result, guidelines for bridging annotation differ in many respects so that they cannot be easily combined to create a larger bridging corpus resource. The latter would however be necessary to further research in this area, as statistical approaches to bridging resolution are limited due to the limited corpus size, cf. for example Hou (2016).

This section will present the different phenomena that have in previous research been treated as bridging and will make a suggestion for an approach that aims at a broad definition of bridging that is compatible with many previous studies.

One issue that came up in the early work on bridging and is still present in some work is the **overlap with coreferent anaphora**. Clark (1975) proposed a very broad definition, including anaphoric use of NPs that have an identity relation with their antecedent, e.g. in

(3) I met a man yesterday. **The man** stole all my money.

While it is nowadays non-controversial that these coreferent cases should not fall under the label of bridging, the more difficult cases of coreference where the anaphor and the antecedent do not share the same head but are in a synonymy, hyponymy or metonomy relation, are sometimes treated as bridging, e.g. in Poesio and Vieira (1998), among others.

(4) I met a man yesterday. **The bastard** stole all my money.

Clark (1975) and Asher and Lascarides (1998) also included **rhetorical relation or connection cases**, e.g. in

(5) John partied all night yesterday. He's going to get drunk **again** today.

While these are interesting cases of anaphoric use, most work nowadays limits the anaphor to nominal referring expressions.

Another important point of discussion is the question whether **definiteness** should be a requirement for bridging anaphors. Many studies (Poesio and Vieira, 1998; Baumann and Riester, 2012; Rösiger, 2016), among many others, have excluded indefinite expressions as potential bridging candidates as indefinite expressions introduce new information that can be processed without the context of the previous discourse. Löbner (1998) suggested that bridging anaphors can also be indefinite, as these indefinite expressions can occur in part-whole or part-of-event relations, with the consequence that many studies have linked them as bridging (e.g. in ISNotes, and others).

(6) I bought a bicycle. **A tire** was already flat.

Riester and Baumann (2017) suggested to restrict the annotation of bridging to definite expressions as part of their information status annotation of referring expressions (r-level) and to treat lexical relations (in indefinite and definite expressions) on another level (called the l-level). We agree with the opinion that definite bridging cases are different from indefinite cases and should, when both are treated as bridging, be labelled as different types of bridging.

Another common issue is **the restriction of bridging to pre-defined relations**, such as part-of, set-membership, possession or event relations, e.g. in the Switchboard corpus, (Nissim et al., 2004). Some corpora do not make such limitations (e.g. ISNotes). We believe that bridging is a versatile phenomenon that cannot be captured with pre-defined relations. Furthermore, some work (e.g. ISNotes) has excluded certain relations, e.g. comparative anaphora (Markert et al., 2012), from the bridging category arguing that they can be found by surface markers, such as *other, another*, etc., e.g. in

(7) About 200,000 East Germans marched in Leipzig and thousands more staged protests in **three other cities**.

Comparative anaphora have different properties than "regular bridging" cases, as they indicate co-alternativity, e.g. a relationship on equal terms, between the antecedent and the anaphor, while for typical bridging cases, the relation between the anaphor and the antecedent is a hierarchical one, with the bridging anaphor being subordinate to the antecedent.

While many approaches distinguish only between coreferent anaphors that refer to the same referent as their antecedent and bridging anaphors that refer to a different referent, Recasens and Hovy (2010; Recasens et al. (2012) has introduced a third concept, the concept of **near-identity** which has been picked up by others (e.g. Grishina (2016)). Near-identity is defined to hold between an anaphor and an antecedent whose referents are almost identical, but differ in one of four respects: name metonomy, meronymy, class or spatio-temporal functions.

(8) Iran maintains diplomatic relations with 99 members of the United Nations. **Tehran** and the P5+1 came to a historic agreement to end economic sanctions.

We prefer to stay with the two-class categorisation of coreference and bridging and argue that in cases where *Iran* and *Tehran* are both used to refer to the Iranian government, they should be considered coreferent. In cases where they do not refer to the same referent, but a related entity, they can in principle be considered bridging. However, in this case, *Tehran* is not anaphoric, which leads us on to the following important distinction.

Referential vs. lexical bridging

We propose the terms referential and lexical bridging to distinguish two different phenomena which are currently both defined as bridging. **Referential bridging** describes bridging anaphors that are truly anaphoric in the sense that they need an antecedent in order to be interpretable, as in

(9) The city is planning a new townhall and **the construction** will start next week.

Referential bridging is often a subclass of (referential) information status annotation. The corpus ISNotes (Markert et al., 2012) is one example of a corpus which solely includes referential bridging.
Lexical bridging describes lexical semantic relations between certain words, i.e. *Spain* and *Europe* being in a meronymy relation. These cases are not anaphoric, as the interpretation of *Spain* does not depend on the antecedent *Europe*. Lexical bridging is often annotated when certain pre-defined relations are defined as bridging. The second release of the ARRAU corpus (first released in Poesio and Artstein (2008)), as used in the first shared task on bridging resolution, for example contains both referential and lexical bridging, with the majority of the bridging links being lexical bridging pairs.
It should be noted that lexical and referential bridging are two different phenomena with completely different properties, although, for sure, they can co-occur in one and the same expression, such as in

(10) a house ... *the door*.

In this paper, we only focus on referential bridging, as we think that these are the bridging cases which are most interesting from a discourse understanding point of view. Also, the task of lexical bridging resolution is related to work that has been done in the NLP community on detecting semantic relations between words (c.f. e.g. Shwartz and Dagan (2016).

1.3. Our Proposed Approach

Our annotation guidelines are on the one hand broad enough to cover many cases, following these principles

- Bridging anaphors have to be anaphoric, i.e. not interpretable without an antecedent (=referential bridging only)
- Bridging relations are not restricted to certain predefined relations;
- bridging anaphora can be definite or indefinite (but we use two different labels to distinguish them);
- bridging antecedents can be nominal entities or events (VP or clauses) .

On the other hand, we propose a clear separation from other tasks:

- No overlap with coreference resolution: context-dependent anaphors that refer to the same entity as their antecedent are considered "given" information (independent of their surface realisation), and thus covered by coreference resolution;
- bridging anaphors are context-dependent expressions that do not refer to the same entity as their antecedent, but to a related entity;
- we focus on referring expressions, excluding rhetorical or connection cases: anaphors are nominal, antecedents can be nominal, verbal or clauses.

The annotation guidelines are tailored to Germanic languages like English and German as they focus on the distinction between definiteness and indefiniteness. The idea of a broad, but clear definition of bridging without an overlap with the concept of coreference can of course also be applied to other languages.

2. Corpus Creation

We annotate 50 articles from the WSJ that are already part of OntoNotes. The articles were selected blindly, but we exluded articles that were already annotated as part of the ISNotes corpus (Markert et al., 2012) and those articles that give an overview of what happened in a certain time frame, thus containing several separate discourses in one document. The corpus is named BASHI, bridging anaphors hand-annotated inventory[3]. It is a relatively small corpus, but because of its categorised bridging links it can be combined with many other corpus resources (e.g. ISNotes), in order to create a larger corpus resource.

3. Annotation Scheme

3.1. Markables

Markables (and thus candidates for bridging anaphors) are all NPs that have been gold annotated in the OntoNotes corpus (Weischedel et al., 2011). Pre-marked NPs in OntoNotes include

- nominal phrases: *the president*
- proper names: *Mr. Bush*
- quantifier phrases: *all the products*
- pronouns: personal, possessive, demonstrative, reflexive

If the annotator thinks that an NP has not been pre-marked, he or she added a markable to the set of markables (this is rarely the case).

3.2. Non-markables

The pre-marked NPs do not include

- nominal premodification: the *US* president
- interrogative or relative pronouns

The annotators are told to mark the longest span of the NP that refers to the entity, including determiners and adjectives, dependent PPs and relative clauses.

[3]Bashi can mean "bridge" in Japanese.

(11) There have been concerns that the Big Board's basket could attract investors with a short term perspective who would rapidly turn over the product, thus increasing volatility.

3.3. Bridging Anaphors

We only annotate referential bridging. This means that bridging anaphors are discourse-new, anaphoric expressions which are dependent on the previous context, and for which the text presents an antecedent NP which does not stand in the relation of identity, but in some other form of relation to the associative phrase. The antecedent may be an associate in a typical relation such as part-of, part-of-event or any kind of associate as long as there is a clear relation between the two phrases.

(12) We use a classifer to distinguish between the two categories.
The training data consists of ...

(13) My sister celebrated her birthday last weekend. I offered to help her make **the cake**.

(14) Our correspondent in Egypt is reporting that **the opposition** is holding a rally against **the constitutional referendum**.

Often, the anaphor is lacking an implicit argument (the antecedent) which enables the interpretation of the expression. This is also reflected in the bridging definition of Roitberg and Nedoluzhko (2016) (called genitive bridging) where they restrict bridging cases to those that can form a genitive construction with the antecedent. While genitive constructions might be a bit too restrictive and the use of genitive constructions is very language-dependent, we agree that bridging pairs can often be seen as head-argument constructions.

(15) **the opposition** (in Egypt)

(16) **the answer** (to this question)

3.3.1. Definite Use

Most bridging anaphors are definite NPs. Note that bare singulars can sometimes also count as definite, in cases where the insertion of the definite article is more plausible than the insertion of an indefinite article. Bare plurals usually count as indefinites.

(17) I went into the room. **The windows** were broken.

(18) We performed the experiments using ... . **Evaluation** is done by means of 10-fold cross validation.

3.3.2. Indefinite Use

Some bridging anaphors are indefinite expressions. In this case, we label the NP as indefinite and link it to the preferred antecedent. Indefinite cases of bridging are typically either part-of or part-of-event relations. As a general rule, indefinite expressions always introduce new information that can be interpreted without context. Nevertheless, we annotate them as bridging in cases where we feel that the interpretation strongly benefits from an argument, i.e. the antecedent.

(19) I bought a bicycle. **A tire** was already flat.

(20) Afghanistan ... **Millions of refugees** would rush home.

3.3.3. Comparative Anaphors

Comparative anaphors have been excluded from the bridging category and treated as a separate category in the ISNotes corpus. We include them in the bridging cases, but label them as comparative and link the comparative markable to the antecedent.

(21) About 200,000 East Germans marched in Leipzig and thousands more staged protests in **three other cities**

(22) President Bush, the Canadian prime minister and **14 other members of the Committee**.

3.4. Antecedents

As a general principle, one antecedent has to be chosen. In special cases, e.g. comparative cases where two antecedents are needed, the annotator may create two or several links.

(23) President Bush, the Canadian prime minister and **14 other members of the Committee**.

We include nominal and abstract antecedents, where the anaphors links back to a VP or a clause.

(24) What is the meaning of life? **The answer** cannot be expressed in one sentence.

The antecedent should be the best semantically related expression. In case of several possible antecedents, the closest should be chosen.
Bridging should not be used as a substitution category for aggregated coreference, where we need two coreference links to for example state that *all sides* involves *the media* and *the congressman* (in a context where these two expressions do not appear in a coordination).

3.5. Link Types

As there are different types of links covered under the term bridging in previous annotation efforts, we distinguish a number of bridging types, for purely pragmatic reasons. The phenomena can then be studied separately, if needed, or certain anaphor types can be excluded when merging data from different source corpora. Cases of the category bridging-contained, as described in Baumann and Riester (2012), is not annotated as bridging because it is not an anaphoric phenomenon and as such a special case where the antecedent modifies the bridging anaphor.

(25) *the windows in the room*

(26) *the mother's room* or *her room*

The annotated bridging link categories are the following: (i) definite bridging links, (ii) indefinite bridging links and

(iii) comparative bridging links. Cataphoric bridging links are not allowed.

4. Annotation Procedure

The annotation is done using the annotation tool Slate (Kaplan et al., 2012) [4]. The markables, i.e. the gold annotated NPs in OntoNotes, are presented in green. Coreferent entities shown in red are already marked and can thus not be marked as bridging anaphors. Exceptions are the first mention in a coreference chain, which can of course be of the category bridging. We refrain from annotating attributes in order not to complicate the annotation process. The annotation involves two annotators (both graduate students in computational linguistics, who have previously been involved in information status annotation) for five WSJ articles, to establish the inter-annotator agreement. The gold annotations for this subset are then merged, and differences between the two versions are resolved via discussion between the annotators. The rest of the corpus is annotated by a single annotator.

5. Difficult Annotation Decisions

Some cases of bridging are very clear, particularly for definite anaphors that occur in a well-defined relation with their antecedent, e.g. whole-part (*the house - the window*). In this case, it is obvious that the definite anaphor requires the antecedent for its interpretation.

5.1. Generic Use vs. Bridging

Other cases are less clear, and they are often a question of generic use vs. bridging. Consider the following example that is taken from the Wall Street Journal and is thus concerned with the US (which is often not explicitly stated, but obvious given the WSJ's location).

(27) **The police** would be waiting.

The question whether *the police* is a generic reference to the concept police or whether a bridging link should be placed between *the police* and *the US* is not obvious. When does such an entity need an antecedent or when does it simply add (optional) information? In cases of obvious generic use, we do not link the two entities. If we get the feeling that we are not speaking about the generic class *police*, but more specifically about the police in, say, Baltimore, we link the two entities. As a general rule, if the entity is interpretable on its own, we do not link it, e.g. in

(28) When you annotate a text, **bridging anaphors** are the most difficult issue.

Still, this distinction remains a little vague.

5.2. Unused/Mediated vs. Bridging

Another difficult choice is the distinction between the information status category unused (sometimes called mediated) and bridging, i.e. in a case like

(29) Iran ... **foreign secretary Mottaki**

where some people might consider this a bridging case, as the *foreign secretary Mottaki* is probably not interpretable alone for a typical WSJ reader without the mentioning of *Iran* first. However, others might argue that his discourse referent might already be identified by his name.
Furthermore, while we typically assume entities like *the moon* to be unique, known entities, and thus of the category unused/mediated, there might be contexts where there are several moons, and one might want to link *the moon* to the entity *the earth* via a bridging relation.

5.3. Determining a Single Antecedent

In some contexts, the writer/speaker introduces a topic into the discourse and then talks about aspects referring to this topic. In cases where there are several noun phrases representing this topic it is not always obvious which NP should be chosen as the antecedent.

(30) No age group is more sensitive than younger voters, like Ms. Ehman. A year ago this fall, voters under 30 favored George Bush by 56 to 39 % over Michael Dukakis, [..]. Voters in **the same age group** backed Democrat Florio 55% to 20 % over Republican Courter.

It is relatively obvious that *the same age group* is a bridging anaphor, but whether *younger voters, like Ms. Ehman*, *Ms. Ehman* or *voters under 30* should be chosen as the antecedent remains unclear (and does not really make a big difference in terms of the interpretation of the anaphor).

6. Resulting Corpus

As can be seen in Table 1, the corpus consists of 459 bridging links, 114 of which contain an indefinite anaphor, 275 a definite anaphor and 70 are comparative anaphors. Out of these 70 comparative anaphors, 12 have more than one link to an antecedent. The corpus contains 57,709 tokens.

Bridging links	459
Definite	275
Indefinite	114
Comparative	70

Table 1: Corpus statistics for the gold bridging corpus

6.1. Inter-Annotator Agreement

Five WSJ articles have been annotated by a second annotator, in order to assess the inter-annotator-agreement. Table 2 shows the agreement for the respective categories. We only report the observed agreement, as the expected agreement for linking markables is considered extremely low (as one can potentially link every NP with all preceeding NPs) and can thus be neglected.
It can be seen that the agreement is high for comparative anaphora: as these almost always occur with surface markers such as *other, another, etc.*, they can be easily spotted. The agreement for the chosen antecedent is also higher, as they are typically local antecedents in a rather narrow window. As expected, the agreement for anaphor detection as

[4] Annotation guidelines:
http://www.ims.uni-stuttgart.de/institut/mitarbeiter/roesigia/guidelines-bridging-en.pdf

Bridging anaphor type	anaphor			anaphor+antecedent		
	same	diff.	agreement	same	diff.	agreement
Definite	34	13	73.9%	30	17	63.8%
Indefinite	15	11	57.7%	11	15	42.3%
Comparative	12	2	85.2%	10	4	71.4%
Total	31	25	70.9%	51	36	59.3%

Table 2: Inter-annotator agreement on five WSJ articles

well as for full bridging resolution is higher for definites than for indefinites. This confirms our hypothesis that for definites, it is easier to decide whether they are anaphoric or not. Overall, for anaphor detection, we achieve an agreement of 70.9% and 59.3% agreement for the overall links. As the overall agreement on the bridging links is rather low (also for other corpora), one could think about evaluating the task of bridging resolution differently than with the typical precision/recall metrics, particularly for contexts such as Example (29).

6.2. Format and Download

The corpus is made available in the form of a download link[5]. The download contains the annotations in an offset-based XML format as well as CoNLL-12 style columns. For the single anaphor type categories (definite, indefinite, comparative) we have created separate columns, as well as one joint column which contains all the bridging links. As the OntoNotes data has to be obtained separately via the LDC, the download will include instructions on how to merge the annotations with the actual corpus data and the annotations in the OntoNotes release (words, part-of-speech, coreference, etc.).

7. Conclusion

We have presented BASHI, a corpus of 50 WSJ articles which adds bridging anaphors and their antecedents to the other gold annotations that have been created as part of the OntoNotes project (Weischedel et al., 2011). As the bridging links contain information about the type of the bridging anaphor (definite, indefinite, comparative), it is compatible with many other bridging corpora and can thus be used to create a bigger corpus resource, which would be required for further advances using statistical methods. The corpus contains 57,709 tokens and 459 bridging pairs and is available for download in an offset-based format and a CoNLL-12 style bridging column that can be merged with the other annotation layers in OntoNotes.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) via the SFB 732, project A6. Many thanks to Janis Pagel for annotating five articles and to Arndt Riester for commenting on the paper as well as coming up with the corpus name.

[5] http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/bashi.html

8. Bibliographical References

Asher, N. and Lascarides, A. (1998). Bridging. *Journal of Semantics*, 15(1):83–113.

Barzilay, R. and Lapata, M. (2008). Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Baumann, S. and Riester, A. (2012). Referential and lexical givenness: Semantic, prosodic and cognitive aspects. *Prosody and meaning*, 25:119–162.

Cahill, A. and Riester, A. (2012). Automatically acquiring fine-grained information status distinctions in German. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 232–236. Association for Computational Linguistics.

Clark, H. H. (1975). Bridging. In *Proceedings of the 1975 workshop on Theoretical issues in natural language processing*, pages 169–174. Association for Computational Linguistics.

Fang, Y. and Teufel, S. (2014). A summariser based on human memory limitations and lexical competition. In *Proceedings of the EACL*. Association for Computational Linguistics. (to appear).

Fraurud, K. (1990). Definiteness and the processing of noun phrases in natural discourse. *Journal of Semantics*, 7(4):395–433.

Grishina, Y. (2016). Experiments on bridging across languages and genres. In *CORBON@ HLT-NAACL*, pages 7–15.

Hawkins, J. A. (1978). Definiteness and indefiniteness: A study in reference and grammaticality prediction. atlantic highlands.

Hearst, M. A. (1994). Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 9–16.

Hobbs, J. R., Stickel, M. E., Appelt, D. E., and Martin, P. (1993). Interpretation as abduction. *Artificial Intelligence*, 63:69–142.

Hou, Y., Markert, K., and Strube, M. (2013a). Cascading collective classification for bridging anaphora recognition using a rich linguistic feature set. In *EMNLP*, pages 814–820.

Hou, Y., Markert, K., and Strube, M. (2013b). Global inference for bridging anaphora resolution. In *Proceedings of NAACL-HLT*, pages 907–917.

Hou, Y. (2016). *Unrestricted Bridging Resolution*. Ph.D. thesis.

Kaplan, D., Iida, R., Nishina, K., and Tokunaga, T. (2012). Slate – a tool for creating and maintaining annotated cor-

pora. *Journal for Language Technology and Computational Linguistics*, pages 89–101.

Kintsch, W. and van Dijk, T. A. (1978). Toward a model of text comprehension and production. *Psychological Review*, 85(5):363–394.

Lassalle, E. and Denis, P. (2011). Leveraging different meronym discovery methods for bridging resolution in french. *Anaphora Processing and Applications*, pages 35–46.

Löbner, S. (1998). Definite associative anaphora. *manuscript) http://user. phil-fak. uniduesseldorf. de/˜ loebner/publ/DAA-03. pdf*.

Markert, K., Hou, Y., and Strube, M. (2012). Collective classification for fine-grained information status. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 795–804. Association for Computational Linguistics.

Mirkin, S., Dagan, I., and Padó, S. (2010). Assessing the role of discourse references in entailment inference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL 2010, pages 1209–1219. Association for Computational Linguistics.

Nedoluzhko, A., Mírovskỳ, J., Ocelák, R., and Pergler, J. (2009). Extended coreferential relations and bridging anaphora in the prague dependency treebank. In *Proceedings of the 7th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2009), Goa, India*, pages 1–16.

Nissim, M., Dingare, S., Carletta, J., and Steedman, M. (2004). An annotation scheme for information status in dialogue. *LREC 2004*.

Poesio, M. and Artstein, R. (2008). Anaphoric Annotation in the ARRAU Corpus. In *International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco, May.

Poesio, M. and Vieira, R. (1998). A corpus-based investigation of definite description use. *Computational linguistics*, 24(2):183–216.

Poesio, M., Vieira, R., and Teufel, S. (1997). Resolving bridging references in unrestricted text. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 1–6. Association for Computational Linguistics.

Poesio, M., Mehta, R., Maroudas, A., and Hitzeman, J. (2004). Learning to resolve bridging references. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 143. Association for Computational Linguistics.

Recasens, M. and Hovy, E. (2010). A typology of near-identity relations for coreference (nident). In *LREC*.

Recasens, M., Marti, M. A., and Orasan, C. (2012). Annotating near-identity from coreference disagreements. In *LREC*, pages 165–172.

Riester, A. and Baumann, S. (2017). The reflex scheme-annotation guidelines.

Roitberg, A. and Nedoluzhko, A. (2016). Bridging corpus for russian in comparison with czech. In *CORBON@ HLT-NAACL*, pages 59–66.

Rösiger, I. (2016). Scicorp: A corpus of english scientific articles annotated for information status analysis. In *LREC*.

Shwartz, V. and Dagan, I. (2016). Path-based vs. distributional information in recognizing lexical semantic relations. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V)*, pages 24–29. The COLING 2016 Organizing Committee.

Vieira, R. and Teufel, S. (1997). Towards resolution of bridging descriptions. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 522–524. Association for Computational Linguistics.

Weischedel, R., Pradhan, S., Ramshaw, L., Palmer, M., Xue, N., Marcus, M., Taylor, A., Greenberg, C., Hovy, E., Belvin, R., et al. (2011). Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*.

Zeldes, A. (2017). The gum corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612, Sep.

Zikánová, Š., Hajicová, E., Hladká, B., Jínová, P., Mírovskỳ, J., Nedoluzhko, A., Poláková, L., Rysová, K., Rysová, M., and Václ, J. (2015). Discourse and coherence. *From the Sentence Structure to Relations in Text. Institute of Formal and Applied Linguistics*.

SACR: A Drag-and-Drop Based Tool for Coreference Annotation

Bruno Oberle
LiLPa (EA 1339), Université de Strasbourg
14 rue Descartes, F-67084 Strasbourg Cedex, France
oberleb@unistra.fr

Abstract

This paper introduces SACR, an easy-to-use coreference chain annotation tool, which is used to annotate large corpora for Natural Language Processing (NLP) applications. Coreference annotation is usually considered as costly both in terms of time and human resources. So, in order to find the easiest annotation strategy, we will first of all compare several annotation schemes implemented in existing tools. Since interface ergonomics is also an important part of our research, we then focus on identifying the most helpful features to reduce the strain for annotators. In the next section of the paper, we present SACR in details. This tool has been developped specifically for coreference annotation, and its intuitive user interface has been designed to facilitate and speed up the annotation process, making SACR equally suited for students, occasional and non-technical users. In order to create coreference chains, elements are selected by clicking on the corresponding tokens. Coreference relations are then created by drag-and-dropping expressions one over the other. Finally, color frames around marked expressions help the user to visualize both marked expressions and their relations. SACR is open source, distributed under the terms of the Mozilla Public License, version 2.0, and freely available online.

Keywords: coreference annotation, annotation tool, coreference chain, interface ergonomics

1. Introduction and Context

The development of statistical methods in automatic language processing leads to an increased need for annotated resources. In particular, training algorithms that aim to detect coreference[1] requires manually annotated texts. Because such a task is costly in terms of time and human resources, there are few annotated corpora with coreference relations. In French, there is only one such a corpus large enough to be used with supervised methods: the ANCOR corpus (Muzerelle et al., 2014), with 488,000 lexical units, 116,000 referring expressions (only nouns and pronouns) and 51,300 relations. But it contains only spoken French.

The "Democrat" project[2] aims, among others, at providing a large corpus annotated with coreference chains. It is expected to have one million words and 100,000 elements of coreference chains. Several annotation strategies have been discussed (Landragin et al., 2017) in order to balance scientific needs with annotation speed.

The main purpose of this paper is to introduce SACR[3], a tool specifically designed to facilitate and speed up annotation of coreference chains. We first compare coreference annotation strategies used by different tools, both in terms of annotation scheme (section 2) and interface usability (section 3), before turning to a presentation of SACR (section 4), and how it is used by annotators in the Democrat project.

2. Coreference chain annotation

According to (Habert, 2005), any annotation task may be divided into three steps; for coreference annotation, these steps are:

1. delimiting and marking referring expressions (that is, a chunk of text that *refers* to some entity, the *referent*, in the extralinguistic world),

2. annotating a set of features for each referring expression (*e.g.* the part of speech of the syntactic head or its grammatical function),

3. linking coreferential expressions to build coreference chains (that is, the set of all the linguistic expressions that refer to the same referent).

We could also add a fourth step which would consist in annotating relations and/or chains (*e.g.* with the type of the referent: a person, an organization, a location, an idea, etc.). Some of these stages may be automated (Landragin, 2011; Poudat and Landragin, 2017), chiefly the annotation of parts of speech, for which well performing tools exist. The other steps require a manual annotation, since no tool is (yet) effective enough to rely on, at least for French. Coreference annotation is the most demanding among these steps, and this is what we will discuss in this section.

Several annotation schemes may be used to annotate coreference relations. First, each relation between two coreferential expressions may be annotated separately (for example, between "cat" and "animal" then between "animal" and "it"). Chains are then built afterwards by transitivity (if "animal" is coreferential with both "cat" and "it", that means

[1] There is coreference when two linguistic expressions refer to the same *referent*, that is, the same entity. For example, in *My cat is drinking milk. It is really thirsty*, both *my cat* and *it* refer to the same entity (the cat I own) and thus are *coreferential*. All expressions that refer to the same entity are said to be in a *coreference chain*.

[2] *DEscription et MOdélisation des Chaînes de Référence: outils pour l'Annotation de corpus (en diachronie et en langues comparées) et le Traitement automatique* "Description and modelling of reference chains: tools for corpus annotation (with diachronic and cross-linguistic approaches) and automatic processing".

[3] SACR is an acronym for *Script d'Annotation des Chaînes de Référence* "Script for Coreference Chain Annotation". It is freely available at `http://boberle.com/projects/sacr`.

that "cat" and "it" are also coreferential). This scheme offers fine-grained annotation possibilities, since it is possible to add a set of features for each relation (is the relation anaphoric? cataphoric? etc.). This method is tedious, though, and a quicker way is to put all the coreferential expressions in one set (a coreference chain), without annotating individual relations. The annotator simply mark "cat", "animal" and "it" as belonging to the same set.

Both schemes are possible with most of the tools that allow coreference annotation. Glozz (Widlöcher and Mathet, 2012) and Analec (Landragin et al., 2012) define a three-level model: *units* are used to mark chunks of texts, that is, in our case, referring expressions; *relations* to annotate binary relations; *schemata* to make coreference chains, by linking either the relations or the units. The other tools have similar features, with different names: MMAX2 (Müller and Strube, 2006) calls the Glozz-like relations "pointer-type relations" and the schemata "set-type relations" where BRAT (Stenetorp et al., 2012; Bra, 2014) uses the terms "binary relations" and "equivalence relations". GATE (Cunningham et al., 2013; Cunningham et al., 2011) seems to allow only generic sets, and not binary relations.

Annotation with schemata in Glozz and Analec has been tested during the MC4 project[4] (Landragin, 2011; Mélanie-Becquet and Landragin, 2014); but this requires to build a schema for each new coreference chain, which involves an extra work.

Rather than building a schema for each chain, a better strategy is to build coreference chains from annotations: they can be easily and automatically deduced if the name of the referent is recorded as a feature of each referring expression: a chain is then the set of referring expressions that have the same referent name. For example, if the expressions "cat", "animal" and "it" have the same referent name "John's cat" recorded as a feature, then it must be that they are in the same coreference chain.

This is a change of perspective: annotators do not build the chain themselves, but focus on finding the referent for each referring expression. This has been found to be quicker and easier than the building of chains *via* the schema-like strategy and is the method currently used in the Democrat project (Landragin et al., 2017).

Consequently, the tool used to annotate must only allow marking referring expressions and recording referent names. Almost every annotation tool can be used as long as annotators can mark tokens and add a feature set to them, so that a tool like UAM CorpusTool (O'Donnell, 2008), which does not (yet) allow annotation of relations, can be used. Even a basic XML editor like Oxygen[5] may be used.

The Democrat project is currently using TXM (Heiden, 2010), a platform to which Analec has been added as an extension. Referent names are added as a feature to each referring expression, and the coreference chains are created automatically afterwards, as schemata computed from these names. But adding and managing dozens of referent names is tedious and the process may be optimized with a dedicated user interface.

3. User interface ergonomics

Usually annotation tools are developed with little attention for the comfort of annotators (Fort, 2012), even if annotating requires both intellectual and physical efforts, and if user interface should take both into consideration (Müller and Strube, 2006).

Physical efforts come first from the fact that annotators must type a referent name for each referring expression. This can be facilitated by offering a default unique name when the referring expression is marked, by auto-completing the name when the referent has been entered previously or even by copying the name from an expression to another by a simple drag-and-drop operation. This also has the advantage to prevent typing errors and name misspellings. Some annotation schemes require all referring expressions to be annotated, and not only those that are related to other expressions, because having all the referring expressions marked allows more refined analysis, like comparing isolated expressions in contrast to coreferential ones. One annotation strategy to ease the work of annotators is to allow them to type a code (like "SI" for "singleton"[6]) when the expression is not related to another: annotations marked with this code will not be included when building chains.

Delimiting and marking referring expressions can also be at times irritating with some tools that do not tokenize the text in "words". While this can be useful for some languages such as Chinese, it is most often an issue when trying to mark an expression around a apostrophe or a comma.

A better visualization of marked expressions helps to reduced the cognitive load. Gate and Analec just highlight marked expressions with only one color, while MMAX offers the possibility to surround them with brackets. BRAT uses colors, but it is Glozz which is the most helpful tool here since it draws colored frames around marked expressions. This is especially important when expressions are nested, and it often happens, when all referring expressions are to be annotated, and not only those which are part of a chain. But Glozz shows relations as arrows, which tend to obstruct the text beneath them.

Keeping track of all the referents encountered so far in a text is difficult yet necessary to be able either to link coreferential expressions or to type exactly the same referent name entered for previously encountered referents. Glozz and Analec (when used with schemata) require the creation of a schema for each chain, so chains are clearly identified. Referents that do not give rise to a chain (a schema) are more problematic. Analec, when used as an extension of TXM (without schemata but with referent names), offers an auto-completing list of referent names. GATE and MMAX2 have similar features. But sometimes the mere referent name is not enough for annotators who need a list of all the expressions associated with a referent, in order to decide how to annotate a new referring expression. Only Glozz offers users an easy way to list all the elements in a given chain.

[4] *Modélisation Contrastive et Computationnelle des Chaînes de Coréférence*, project from the French *Centre National de la Recherche Scientifique*.

[5] `https://www.oxygenxml.com/`

[6] A *singleton* is a referring expression that is not related to another one.

Several authors (Poudat and Landragin, 2017; Felt et al., 2010) have pointed out that some annotations can be made automatically, for example the annotation of parts of speech. CCASH (Felt et al., 2010) is entirely built around this idea, and has integrated modules (called "widgets") that automate simple annotation tasks. But it is difficult to create custom widgets. Tools that have no such integrated modules may be paradoxically more flexible since they allow the use of other programs *via* a common exchange format. For example, referring expressions may be manually marked with one dedicated tool, parts of speech may be automatically added with some other specialized tool, etc. In the Democrat project (Poudat and Landragin, 2017), using a chunker, like SEM (Tellier et al., 2012), has been considered to pre-annotate the text, so annotators would only have to create coreference relations.

All these highlights are not found at once in general purpose annotation and analysis tools, but require a dedicated tool, specifically designed to ease the coreference chain annotation process. SACR was designed in the light of this idea.

4. Presentation of SACR

4.1. Overview

SACR (figure 1) is an open source, ready-to-use annotation tool. Written in HTML, CSS and JavaScript, it is a single-page application that works with Firefox or Chromium/Google Chrome. It can be used immediately, with no installation overhead, by virtually anyone on any platform. It is available online, but can also be downloaded for offline use.

Referring expressions are marked by clicking on their first and last words, coreference relations are made by dragging one expression and dropping it on another. There is almost no learning time and the tool is thus well suited for students of literary background or non-technical annotators as well as more expert users.

It is dedicated to annotation and users are expected to use some other tools (for example TXM) for the analyses. This has allowed us to optimize its interface specifically for the annotation process.

4.2. Creating annotations

Markables may be words (that is, a serie of letters) or characters for languages where characters are words (*e.g.* Chinese). A click on the first and last markables is enough to make a referring expression with a default unique name. Rather than having to type the referent name for each expression, a simple drag-and-drop operation is used to copy the name: this create a new coreference relation, which is symbolized by the coloring of the two linked expressions. When adding more related expressions, they get the same color, so each chain is identified with a unique color, while singletons (referring expressions not related to any other one) are left gray. Using eye-catching colors for the most prominent referents allows to easily identify them and to rely heavily on the drag-and-drop without having to carefully read and type referent names (*e.g.* if a pronoun "he" refers to "Paul", and "Paul" is in red, just drag the "he" and drop it on one of the red elements). Default names may be changed (chiefly for the most prominent referents), but this is not necessary, and annotators may want to rely only on the colors.

When it comes to annotate features like parts of speech or grammatical functions (or check them if they are the output of an automatic annotating system), a special edition mode allows annotators to use shortcuts (*e.g.* "d" for a noun with a definite article, "p" for a personal pronoun, etc.): the next expression is automatically selected and presented in the middle of the screen when annotators have made their choice. This way, one keystroke is enough for each annotation so that annotating or checking a feature for one hundred referring expressions requires only one hundred keystrokes, and nothing more.

SACR does not impose a predefined annotation scheme: users can use as many features as they want (parts of speech, grammatical functions, morphological informations, number and type of modifiers, named entity types, speakers (for oral transcriptions), etc.) and choose the tagset and the corresponding shortcuts. The annotation scheme is saved into a separate file and so may be used for several texts or corpora.

4.3. Helping the annotator

Annotation visualization is an important part of SACR. Marked expressions are surrounded by colored frames, allowing an easy view of several levels of nested expressions (for instance, a genitive in a relative clause). This is especially critical when all the referring expressions of a text must be marked.

A popup window (figure 2) lists all the referents and all the expressions associated with each referent. So annotators can look for already annotated expressions and decide whether they must attach, or not, the expression they are currently working on to a previously defined referent. Drag-and-dropping is also possible to link two somewhat distant expressions. This list may also be used to check if an expression has not been related to the wrong coreference chain.

Referring expressions may be filtered in several ways. It is possible to show only expressions from a given chain. Users can also search expressions with specific features, for example all expressions with a certain part of speech (all pronouns, etc.) or grammatical function (all subjects, etc.), if these features are annotated. The search can be made with a literal or with a regular expression, so it is possible to combine criteria (*e.g.* searching all nouns and pronouns). Such searches are useful to check annotations at the end of the annotation process, for example by looking for expressions lacking any value for a feature (which means that an annotator has forgotten to annotate it).

SACR does not perform any automatic annotation, but its simple import/export text format allows converting to and from other tools that can perform such annotations. A nominal chunker may be used as a pre-treatment so that annotators do not need to mark referring expressions, but only to link them. Several tools have been tested, including TreeTagger (Schmid, 1994; Schmid, 1995) and SEM (Tellier et al., 2012), but they are limited and add more work (too many wrong results) than they solve, since correcting

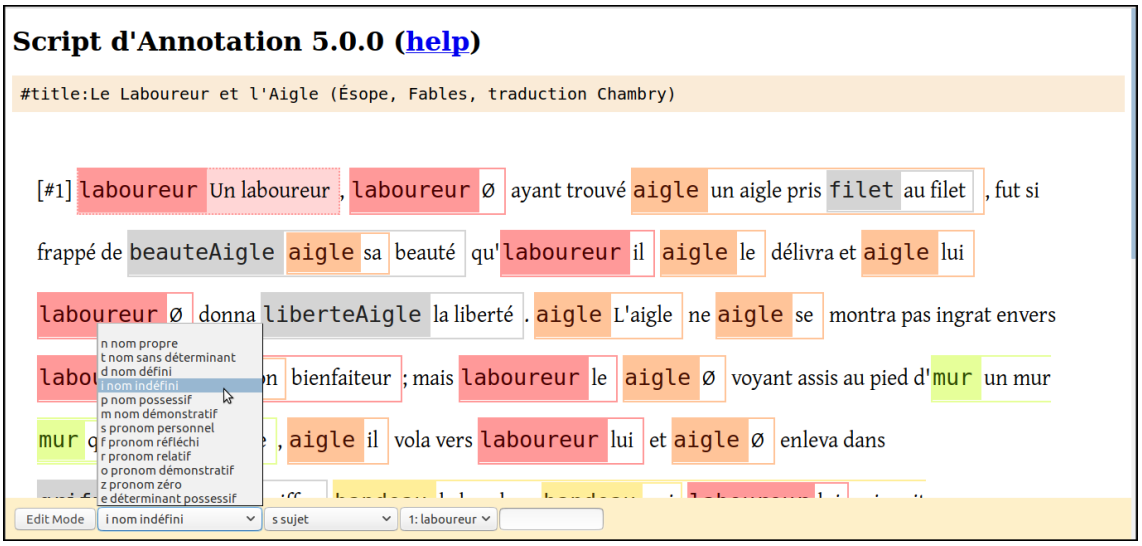


Figure 1: Annotating coreference and parts of speech with SACR.



Figure 2: List of all referring expressions grouped by chains. Elements of a chain can be collapsed or expanded, and it is possible to drag-and-drop any element of the list from or to the main window, or vice-versa, to add a coreference relation.

wrong annotations requires an additional cognitive workload; furthermore, referring expressions that are not detected by the tool may be missed by annotators if they rely too much on the tool output. Features like parts of speech may also be annotated with other tools, and checked within SACR.

4.4. Output format

Annotations made in SACR are stored inline, that is, within the text. The aim is to offer an easy-to-parse format that can be converted to other formats with the help of simple scripts. Such scripts exist for example to convert to and from the Glozz format (Widlöcher and Mathet, 2012) (a format in which annotations are deported and that can be imported in tools such as Analec or TXM) or the CONLL-2011 format. Inline annotations also allow the modification of the text itself, for example for spelling correction. Stand-off annotations, on the other hand, offer other advantages like the possibility to easily add annotation layers (*e.g.* output from a part-of-speech tagger and a parser). Since Glozz offers such a stand-off format and there are converters from and to the Glozz format, it is easy to combine the advantages of both type of annotations storage.

The format includes any metadata that are needed for the program (colors), the organization of the corpus (identifier, title, author, source, etc.), and even the text structure (headings of different levels; sections like “introduction” or “conclusion”; etc.).

The ouput file itself contains the text with the marked expressions enclosed between curly brackets. Features are stored near the opening bracket. Here is a short example:

```
#title:  Le laboureur et l'Aigle

{laboureur:categorie="i
nom indéfini",fonction="s
sujet",head="1",expansion="" Un
laboureur}, {laboureur:categorie="z
pronom zéro",fonction="s
sujet",head="0",expansion="" Ø}
ayant trouvé {aigle:categorie="i
nom indéfini",fonction="v compl
(verbe)",head="1",expansion="v"
un aigle pris {filet:categorie="d
nom défini",fonction="c
circonstant",head="1",expansion="" au
filet}}...
```

The file describing the features to be annotated is also a simple text file listing all the values allowed for a property,

or offering a text box for annotators to complete:

```
PROP:name=categorie
n nom propre
t nom sans déterminant
d nom défini
...

PROP:name=fonction
s sujet
v compl (verbe)
n compl (nom)
...

PROP:name=head,type=head

PROP:name=expansion,type=text
```

A special type of property, called `head`, is set to let annotators choose the syntactic head of the referring expression (which is, in some annotation schemes, required to be a syntactic phrase).

Annotators may define the feature set they want, with the features and values adapted for specific tasks and/or languages. SACR may also be used without any feature set defined: in this case, annotators just have to mark referring expressions and create coreference relations.

4.5. Using SACR: User feedback and experiment

This tool has been easily used by several annotators, both students and linguists, even when they were not at ease with computers. The preferred way of annotating is to mark all the referring expressions of a paragraph: this first reading allows annotators to get the meaning of the passage. Then coreference relations are marked by drag-and-dropping expressions one over the other. Finally, when the whole text has been annotated with coreference, users may annotate features for each expression (this process may be automated by an external tool).

We conducted a small-scale experiment to compare SACR to Analec (Landragin et al., 2012) as integrated in TXM (Heiden, 2010). Three master's students in linguistics and computational linguistics have been trained to use both SACR and TXM, and have been asked to mark referring expressions and coreference relations in a 18,900 token long corpus of fables (easy to understand and annotate) and legal European texts (more challenging to understand and annotate, with referring expressions sometimes spanning several lines and nested several levels deep).

When using SACR, annotators have taken 55 % less time on average to complete the task, which means that annotation with SACR is about twice as fast as with Analec-TXM. We had expected difficult legal texts to be annotated even faster with SACR than simpler texts like fables, but this was not the case: the reduction in time is similar for both types of texts.

Beyond coreference, SACR has been used for other annotation tasks that necessitated a quick and user-friendly interface. It can handle any annotation scheme that requires to mark chunks of text and to annotate a feature set.

4.6. Limitations

SACR is dedicated to referring expression and coreference chain annotation, with or without a set of features for each expression. While it may be used to annotate other type of linguistic phenomena (named entities, specific vocabularies, etc.), its focus on coreference chains implies some limitations. Most notably, there is no possibility of discontinuous annotations because each referring expression is assumed to be a syntactic phrase. This is also the reason why there cannot be any overlapping marked expression, but only nested ones. This limitation fits the technical choice of using HTML as a way of representation. Furthermore, we have made the choice of considering coreference chain as a set of referring expressions, without the possibility of annotating each relation in particular. For example, while it is possible to annotate features for expressions A and B, is not possible to annotate specific features for the *relation* from A to B.

5. Conclusion

In this paper, we presented SACR, a new easy-to-use, annotator-friendly tool designed and optimized for coreference chain annotation. It requires no installation and is ready-to-use even by non-technical users. Marking referring expressions is done by clicking on the first and last words of an expression, and coreference chains by drag-and-dropping expressions one over the other.

6. Acknowledgements

This work was supported by the Democrat project (*DEscription et MOdélisation des Chaînes de Référence: outils pour l'Annotation de corpus (en diachronie et en langues comparées) et le Traitement automatique*), from the French National Research Agency (ANR) (ANR-15-CE38-0008).

7. Bibliographical References

(2014). BRAT: Brat Rapid Annotation Tool. `http://brat.nlplab.org`.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Cunningham, H., Tablan, V., Roberts, A., and Bontcheva, K. (2013). Getting more out of biomedical documents with gate's full lifecycle open source text analytics. *PLOS Computational Biology*, 9(2):1–16.

Felt, P., Merkling, O., Carmen, M., Ringger, E., Lemmon, W., Seppi, K., and Haertel, R. (2010). CCASH: A web application framework for efficient, distributed language resource development. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Fort, K. (2012). *Les ressources annotées, un enjeu pour l'analyse de contenu: vers une méthodologie de l'annotation manuelle de corpus*. Ph.D. thesis, Paris 13.

Habert, B. (2005). Portrait de linguiste(s) à l'instrument. *Texto!*, 10(4).

Heiden, S. (2010). The TXM Platform: Building Open-Source Textual Analysis Software Compatible with the TEI Encoding Scheme. In Ryo Otoguro, et al., editors, *24th Pacific Asia Conference on Language, Information and Computation*, pages 389–398, Sendai, Japan. Institute for Digital Enhancement of Cognitive Development, Waseda University.

Landragin, F., Poibeau, T., and Victorri, B. (2012). Analec: a new tool for the dynamic annotation of textual data. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Landragin, F., Potier, J., and Bothua, M. (2017). Annotation manuelle d'expressions référentielles: expérimentations pour simplifier les prises de décisions et optimiser le processus. In *9èmes Journées Internationales de la Linguistique de Corpus (JLC 2017)*, Grenoble, France.

Landragin, F. (2011). Une procédure d'analyse et d'annotation des chaînes de coréférence dans des textes écrits. *Corpus*, 10:61–80.

Mélanie-Becquet, F. and Landragin, F. (2014). Linguistique outillée pour l'étude des chaînes de référence: questions méthodologiques et solutions techniques. *Langages*, 195.

Müller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. In Sabine Braun, et al., editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a. M., Germany.

Muzerelle, J., Lefeuvre, A., Schang, E., Antoine, J.-Y., Pelletier, A., Maurel, D., Eshkol, I., and Villaneau, J. (2014). ANCOR Centre, a large free spoken french coreference corpus: Description of the resource and reliability measures. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

O'Donnell, M. (2008). Demonstration of the UAM CorpusTool for text and image annotation. In *Proceedings of the 46th annual meeting of the Association for Computational Linguistics on human language technologies: Demo session*, pages 13–16. Association for Computational Linguistics.

Poudat, C. and Landragin, F. (2017). *Explorer un corpus textuel*. Champs linguistiques. De Boeck.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to german. In *Proceedings of the ACL SIGDAT-Workshop*.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

Tellier, I., Duchier, D., Eshkol, I., Courmet, A., and Martinet, M. (2012). Apprentissage automatique d'un chunker pour le français. In Gilles Sérasset Georges Antoniadis, Hervé Blanchon, editor, *TALN2012*, volume 2 of *Actes de la conférence conjointe JEP-TALN-RECITAL 2012*, pages 431–438, Grenoble, France.

Widlöcher, A. and Mathet, Y. (2012). The glozz platform: A corpus annotation and mining tool. In *Proceedings of the 2012 ACM symposium on Document engineering*, pages 171–180. ACM.

Deep Neural Networks for Coreference Resolution for Polish

Bartłomiej Nitoń, Paweł Morawiecki, and Maciej Ogrodniczuk
Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
{bartek.niton, pawel.morawiecki}@gmail.com, maciej.ogrodniczuk@ipipan.waw.pl

Abstract

The paper presents several configurations of deep neural networks aimed at the task of coreference resolution for Polish. Starting with the basic feature set and standard word embedding vector size we examine the setting with larger vectors, more extensive sets of mention features, increased number of negative examples, Siamese network architecture and a global mention clustering algorithm. The highest results are achieved by the system combining our best deep neural architecture with the sieve-based approach – the cascade of rule-based coreference resolvers ordered from most to least precise. All systems are evaluated on the data of the Polish Coreference Corpus featuring 540K tokens and 180K mentions. The best variant improves the state of the art for Polish by 0.53 F1 points, reaching 81.23 points of the CoNLL metric.

Keywords: coreference resolution, deep neural network, Polish

1. Introduction

Coreference resolution, the task of clustering textual fragments that refer to the same entity in the discourse world, has been successfully tackled for Polish in numerous configurations, starting with a rule-based model (Ogrodniczuk and Kopeć, 2011) through machine-learning (Kopeć and Ogrodniczuk, 2012) and projection-based approaches (Ogrodniczuk, 2013) up to the newest multi-pass sieve setting (Nitoń and Ogrodniczuk, 2017). In this paper we present the first deep neural network resolver for Polish, further improving the state of the art. For English, the state-of-the-art coreference resolution systems are also based on deep neural networks (Clark and Manning, 2016), (Wiseman et al., 2016). We were inspired and motivated by these works.

The data for our experiments, as for all previous configurations, come from the Polish Coreference Corpus (Ogrodniczuk, 2015, PCC), a large corpus of Polish general nominal coreference manually annotated over the texts of the National Corpus of Polish (Przepiórkowski et al., 2012)[1] and Rzeczpospolita Corpus (Presspublica, 2002). The corpus features broad understanding of mentions (e.g. with included relative clauses or appositions, nesting, discontinuities and zero anaphora) and contains almost 1800 documents from 14 genres, 540K tokens, 180K mentions and 128K coreference clusters.

Coreference scores on the test set are measured using gold mentions on input with *MUC* (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), and *CEAFE* (Luo, 2005) metrics averaging them according to the CoNLL-2011 approach (Pradhan et al., 2011) to track influence on different coreference dimensions (the B^3 measure being based on mentions, *MUC* on links, and *CEAFE* on entities). *CEAFM* (Luo, 2005) and *BLANC* (Recasens and Hovy, 2011) are also presented for consideration. Metrics were calculated using *Scoreference*[2], a mention detection and coreference resolution evaluation tool (Ogrodniczuk et al., 2015).

[1] Pol. Narodowy Korpus Języka Polskiego (NKJP), see `http://nkjp.pl`.

[2] `http://zil.ipipan.waw.pl/Scoreference`

2. The Baseline

In all our experiments we used 90% texts from the *PCC* as the training set and 10% as the test set. Text type balance was maintained in this division.

Our neural networks return a single output (a value between 0 and 1), which is interpreted as the probability of two mentions being coreferent. Mentions are then linked into coreference chains with a certain *clustering algorithms*. We experimented with both *mention-based* and *entity-based* settings. The *mention-based* algorithm connects each anaphor with an antecedent for which neural network returned the best prediction score. The *entity-based* algorithm connects each anaphor with a mention group for which the neural network returned the best average prediction score (which is average prediction between the anaphor and each mention being part of the tested mention group).

For both types of algorithms the prediction must be higher than selected *connection threshold*, i.e. the value above which two mentions are considered coreferent. Each experiment (excluding *Experiment 3*) was tested on a set of various different pre-selected *connection threshold* values: *0.5*, *0.75*, *0.85*, *0.95*, and *0.99*.

2.1. Input Features

Each training features vector gathers information about antecedent, anaphor and antecedent-anaphor pair. Each mention features vector consists of:

- word embedding vectors (Wawer, 2015) for the mention head word, the first word in the mention, two words preceding the mention and two words following the mention
- averages of embeddings vectors calculated for five words preceding the mention, five words following the mention, the words of the mention and the words of the sentence in which the mention occurred
- binary features marking whether the mention is of a

nominal type[3], pronominal type[4], a zero type[5] or other.

Each pair's features vector consists of the distances between mentions in the pair measured in words and in mentions[6], and a set of binary features marking whether:

- mentions in a pair intersect
- mentions are identical (two features: without lemmatization or using lemmatized mentions strings, obtained with Morfeusz morphological analyser[7] (Woliński, 2014) and Pantera tagger[8] (Acedański, 2010))
- mentions are in the same sentence
- mentions are in the same paragraph
- one mention is an acronym of the other
- the antecedent contains the rarest (in terms of frequency) word from the anaphora[9].

In this experiment we used the word embedding vectors of the size 50. Each training example (pair of mentions) has 1147 features (554 for each mention and 39 pair features describing their relations) and is labeled with 1 or 0 marking whether mentions are coreferent or not.

2.2. Network Architecture

Input features described above are concatenated into a single vector and act as input to our neural network. Thus, the network takes an input vector of 1147 units and is passed through a fully connected network with a single output (a value between 0 and 1). The output is interpreted as the probability of two mentions being coreferent. The network has 3 hidden layers, where a number of units in subsequent layers are 500, 300, and 100. In hidden layers we use RECTIFIED LINEAR UNIT — RELU (Nair and Hinton, 2010) as an activation function and a sigmoid function in the output layer.

2.3. Training Details

The network is trained by finding the parameters (weights) to minimize the loss function. Regarding the loss, we follow a typical choice, namely a binary cross entropy function. During the training, the loss was minimized with ADAM (Kingma and Ba, 2014) for 2 epochs with mini-batches of size 128. We experimented with longer training (more epochs) but the network became overfitted. We used batch normalization (Ioffe and Szegedy, 2015) in each hidden layer and the network was regularized using dropout (Srivastava et al., 2014) with a rate of 0.2.
Part of input features consists of word embeddings and these vectors are treated as static and are not modified during training.
Training set had 426 thousands pairs of mentions, equally split between positive and negative pairs. The neural network model was implemented with KERAS (Chollet and others, 2015) using TENSORFLOW (Abadi et al., 2016) as a backend. For training we used the GPU (K40 TESLA) and the training was completed within a few minutes (around 2 minutes per epoch). The implemented models are publicly available at `http://zil.ipipan.waw.pl/Corneferencer`.

2.4. The Results

First we evaluated the neural network model on the test set consisting of 40K mention pairs. Our baseline model accuracy is 72.27%, which means approximately 72% of examples are classified correctly.
Then we evaluated the neural network on whole texts (not only selected mention pairs) from the test set using THE CORNEFERENCER[10] system specially implemented for this task. The best score was acquired for *the mention-based* clustering algorithm with *the connection threshold* 0.99 (see row labeled as *Baseline* in Table 1 for results).

3. The Experiments

3.1. Experiment 1: Larger Vectors

After experimenting with the basic feature set we tested different architectures in pursuit of a better, more robust model. The first improvement featured larger word embedding vectors (of the size 300 instead of 50), which gave 6647 features for each training example. However, despite much richer embeddings, we did not observe any significant improvements in the evaluation metrics. The best results were acquired for *mention-based* clustering algorithm with 0.99 *connection threshold* (see *Experiment 1* in Table 1). It might be the case that 50-value embeddings are just enough to capture similarities (or any other relations) relevant to our task.

3.2. Experiment 2: More Features

In the next step we brought back embeddings vector size to 50 and added extra input features to the training examples. We selected the features proved best in other coreference resolution systems for Polish, e.g. the model described in

[3]Nominal mentions are all nominal phrases whose syntactic head is a noun marked with a `subst` (general noun) or `ger` (gerund) tags (see `http://nkjp.pl/poliqarp/help/en.html` for a concise tag descriptions).

[4]Pronominal mentions are first-, second- (annotated as `ppron12`) or third-person pronouns (`ppron3`).

[5]Zero mentions are marked with tags corresponding to verbal forms (`fin`, `praet`, `bedzie`, `winien`, `aglt`, and `impt`).

[6]Distances are binned into one of the buckets [0,1,2,3,4,5-7,8-15,16-31,32-63,64+,discontinuous] and then represented as binary features (last bucket is reserved for situation when one mention is between parts of second discontinuous mention)

[7]`http://sgjp.pl/morfeusz/`

[8]`http://zil.ipipan.waw.pl/PANTERA`

[9]For the purpose of checking word rarity we used a word frequency list extracted from the balanced subcorpus of the National Corpus of Polish (Przepiórkowski et al., 2012).

[10]CORNEFERENCER (`http://zil.ipipan.waw.pl/Corneferencer`) is a neural network based tool for performing coreference resolution. It is the final product of the research described in this article, it was used to get system annotation for each experiment using for this task pretrained neural networks. Default CORNEFERENCER configuration is the one described as *Experiment 5*.

(Ogrodniczuk et al., 2015) and in (Nitoń and Ogrodniczuk, 2017).
Additional binary mention features are e.g. features marking whether the mention:

- is in first or second person
- starts with a demonstrative pronoun
- starts with a demonstrative pronoun and is nominal
- starts with a demonstrative pronoun and is pronominal or zero
- is a reflexive pronoun
- is first in a sentence
- is a personal pronoun or zero mention (false, if not one of them)
- head contains a digit
- contains a letter
- is post modified (a head word is not the last word in the mention).

Additional binary pair features are features marking whether:

- distance between mentions in sentences is 1, 2 or more (3 features)
- their gender values agree (without distinction of masculine gender into subtypes)
- the string of one mention starts with second mention's string
- the string of one mention ends with second mention's string
- the string composed of the initial letters of all the capitalized words in the mention string produces a string matching a head word of the second mention
- mentions are in the same sentence, the anaphor is pronominal, and the antecedent is the first in paragraph
- mentions are in the same sentence, their persons and numbers agree, and the antecedent is the first in paragraph
- mentions are in adjacent sentences, are adjacent mentions (without any other mention in between), their persons and numbers agree and the anaphor is pronominal
- mentions are in adjacent sentences, are adjacent mentions and the anaphor is pronominal
- they safisfy additional conditions for six knowledge-based features — 3 PLWORDNET-based and 3 WIKIPEDIA-based, closely described in (Ogrodniczuk et al., 2015).

We also added string kernel features matching whole mentions or their heads (2 features).
As suspected, the features which are working well in other systems also significantly increased the evaluation metrics of our solution. Best results were acquired for *mention-based* clustering algorithm with 0.95 *connection threshold* (see *Experiment 2* in Table 1).

3.3. Experiment 3: Siamese Networks

Next we tried a different network architecture called the Siamese network (Bromley et al., 1994). Networks of this type are particularly useful for tasks that involve finding similarity or a relationship between two comparable things. The network consists of two identical subnetworks (weights are shared) to process two inputs followed by another module which produces the final output. We used here same embeddings vector size and features as in *Experiment 2* with the difference that one network uses all mention features of the antecedent and features corresponding to the tested mention pair and the other uses all mention features for the anaphora and also mention pair features. So we are using same pair features at the input of both networks.
Typically, Siamese networks are applied to determine whether two faces belong to the same person or to figure out whether two signatures come from the same person. Unfortunately, this architecture did not bring us any improvement over the baseline results (see *Experiment 3* in Table 1 for the best acquired, in this experiment, results).

3.4. Experiment 4: More Negative Examples

In the next experiment we used features, embeddings vector size, and architecture from *Experiment 2* but extended the training set by additional 600 thousands negative pairs of mentions, also including singletons. Dominance of negative examples over positive is a typical situation in real texts, where most pairs are not coreferent. Thus our new training set should correspond better to a real test scenario. The best results were obtained for *mention-based* clustering algorithm with 0.85 *connection threshold* (see *Experiment 4* in Table 1) and improve the metrics by over 3%.

3.5. Experiment 5: *All2all* Mention-based Clustering Algorithm

The *mention-based* detection algorithm, in its base form, considers only mentions preceding the mention to be clustered. In this experiment we checked all possible mention pairs regardless their positions in the text. We used here the same configuration (embeddings vector size, network architecture, features) as in *Experiment 4*.
Best results were acquired for 0.85 *connection threshold* (see *Experiment 5* in Table 1). We refer later to this clustering algorithm as *all2all*.

3.6. Experiment 6: Mixed Architecture

In the last experiment we simulated mixing the sieve-based architecture described in (Nitoń and Ogrodniczuk, 2017) with our best neural system configuration (*Experiment 5*). To acquire this we preprocessed input data with the sieve-based coreference resolver using different sieve configurations and then by CORNEFERENCER tool using the *all2all*

System	MUC [%]			B3 [%]			CEAFM [%]		
	P	R	F1	P	R	F1	P	R	F1
Baseline	71.87	40.15	51.52	**94.87**	79.35	86.42	78.65	78.65	78.65
Experiment 1	64.96	44.46	52.79	91.72	80.43	85.71	77.79	77.79	77.79
Experiment 2	62.80	59.30	61.00	87.64	83.72	85.64	78.85	78.85	78.85
Experiment 3	55.67	55.07	55.37	84.41	82.39	83.39	75.69	75.69	75.69
Experiment 4	**72.64**	59.66	65.51	91.08	83.95	**87.37**	**82.08**	**82.08**	**82.08**
Experiment 5	69.31	65.28	67.23	87.19	86.01	86.59	81.14	81.14	81.14
Experiment 6	70.34	**68.12**	**69.21**	86.76	**86.72**	86.74	81.69	81.69	81.69

System	CEAFE [%]			BLANC [%]			CoNLL
	P	R	F1	P	R	F1	[%]
Baseline	77.02	**90.37**	83.16	**85.08**	60.08	65.42	73.70
Experiment 1	77.99	87.66	82.54	78.23	62.91	67.59	73.68
Experiment 2	82.76	84.57	83.65	76.57	68.16	71.54	76.76
Experiment 3	81.19	81.54	81.37	70.54	66.13	68.05	73.37
Experiment 4	84.33	90.24	87.19	76.97	69.65	**72.70**	80.02
Experiment 5	85.92	87.88	86.89	68.45	74.03	70.85	80.24
Experiment 6	**87.21**	88.29	**87.75**	68.10	**74.71**	70.86	**81.23**

Table 1: Comparison of coreference resolution scores for different experiments with neural networks

System	MUC [%]			B3 [%]			CEAFM [%]		
	P	R	F1	P	R	F1	P	R	F1
Ruler	51.38	65.61	57.63	78.78	84.99	81.76	74.57	74.57	74.57
Bartek–3	61.14	67.90	64.34	84.08	86.09	85.07	79.81	79.81	79.81
Bartek–S1	70.30	65.35	67.73	**87.91**	85.38	86.63	**81.74**	**81.74**	**81.74**
Neural	69.31	65.28	67.23	87.19	86.01	86.59	81.14	81.14	81.14
Mixed	**70.34**	**68.12**	**69.21**	86.76	**86.72**	**86.74**	81.69	81.69	81.69

System	CEAFE [%]			BLANC [%]			CoNLL
	P	R	F1	P	R	F1	[%]
Ruler	84.89	75.65	80.00	70.69	68.53	69.55	73.13
Bartek–3	86.99	83.22	85.06	**75.67**	73.01	**74.26**	78.16
Bartek–S1	86.56	**88.96**	87.74	70.19	71.73	70.93	80.70
Neural	85.92	87.88	86.89	68.45	74.03	70.85	80.24
Mixed	**87.21**	88.29	**87.75**	68.10	**74.71**	70.86	**81.23**

Table 2: Comparison of coreference resolution systems

clustering algorithm. As we can see in Table 1 it brings some improvement for coreference resolution even over sieve-based solution (see Table 2). We think that is due to the fact that such system uses more complex mechanisms in cases where simple rules fail. It also merges initial (detected by sieve system) mention groups by hardest links between their mentions based on the prediction made by the neural network.

Best results were acquired while preprocessing data with full set of sieves described in (Nitoń and Ogrodniczuk, 2017) as best configuration and 0.95 *connection threshold* (see *Experiment 6* in Table 1).

4. Summary

Table 2 presents comparison of our new coreference resolution strategies (*Neural* and *Mixed*) with *Bartek–S1*, sieve-based solution described in (Nitoń and Ogrodniczuk, 2017) and two existing coreference resolution systems for Polish described in detail in (Ogrodniczuk et al., 2015). RULER is a simple rule-based tool with design following (Haghighi and Klein, 2007) and BARTEK–3 is an adaptation of the BART system for Polish, being a machine learning-based solution.

The comparison shows that using solely neural network-based system we can almost reach the state of the art for coreference resolution score for Polish. Combining the sieve-based architecture and the best acquired neural network configuration has led to the best score for Polish coreference resolution ($\tilde{0}.5\%$ improvement in *CoNLL* over the best sieve-based system). We think that there is still room for improvement, specifically by trying different neural architectures and/or using knowledge from sieves in the training phase of a neural net. The main disadvantage of using neural networks is the clustering time, which is way longer than in compared approaches, therefore it is not the best solution for real-time working tools.

5. Acknowledgements

The work reported here was carried out within the research project financed by the Polish National Science Centre (contract number 2014/15/B/HS2/03435).

6. Bibliographical References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., and Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283.

Acedański, S. (2010). A Morphosyntactic Brill Tagger for Inflectional Languages. In Hrafn Loftsson, et al., editors, *Advances in Natural Language Processing: Proceedings of the 7th International Conference on NLP (IceTAL 2010)*, pages 3–14. Springer Berlin Heidelberg, Berlin, Heidelberg.

Bagga, A. and Baldwin, B. (1998). Algorithms for Scoring Coreference Chains. In *Proceedings of The 1st International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pages 563–566.

Bromley, J., Guyon, I., Lecun, Y., Säckinger, E., and Shah, R. (1994). Signature Verification using a "Siamese" Time Delay Neural Network. In *In NIPS Proc*.

Chollet, F. et al. (2015). Keras. `https://github.com/fchollet/keras`.

Clark, K. and Manning, C. D. (2016). Deep reinforcement learning for mention-ranking coreference models. *arXiv preprint arXiv:1609.08667*.

Haghighi, A. and Klein, D. (2007). Unsupervised Coreference Resolution in a Nonparametric Bayesian Model. In John A. Carroll, et al., editors, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 848–855. The Association for Computational Linguistics.

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Francis R. Bach et al., editors, *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org.

Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

Kopeć, M. and Ogrodniczuk, M. (2012). Creating a Coreference Resolution System for Polish. In Nicoletta Calzolari, et al., editors, *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 192–195, Stambuł. European Language Resources Association.

Luo, X. (2005). On Coreference Resolution Performance Metrics. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 25–32, Vancouver, Canada. Association for Computational Linguistics.

Nair, V. and Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. In Johannes Fürnkranz et al., editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814. Omnipress.

Nitoń, B. and Ogrodniczuk, M. (2017). Multi-pass sieve coreference resolution system for Polish. In Jorge Gracia, et al., editors, *Proceedings of the 1st Conference on Language, Data and Knowledge (LDK 2017)*, number 10318 in Lecture Notes in Artificial Intelligence, pages 222–236. Springer International Publishing, Berlin.

Ogrodniczuk, M. and Kopeć, M. (2011). Rule-based coreference resolution module for Polish. In *Proceedings of the 8th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2011)*, pages 191–200, Faro, Portugal.

Ogrodniczuk, M., Głowińska, K., Kopeć, M., Savary, A., and Zawisławska, M. (2015). *Coreference in Polish: Annotation, Resolution and Evaluation*. Walter De Gruyter.

Ogrodniczuk, M. (2013). Translation- and Projection-Based Unsupervised Coreference Resolution for Polish. In Mieczysław A. Kłopotek, et al., editors, *Proceedings of the 20th International Conference Intelligent Informa-*

tion Systems, volume 7912 of *Lecture Notes in Computer Science*, pages 125–130. Springer-Verlag, Berlin, Heidelberg.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). CoNLL-2011 Shared Task: Modeling Unrestricted Coreference in OntoNotes. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, CONLL Shared Task '11, pages 1–27, Stroudsburg, PA, USA. Association for Computational Linguistics.

Adam Przepiórkowski, et al., editors. (2012). *Narodowy Korpus Języka Polskiego [Eng.: National Corpus of Polish]*. Wydawnictwo Naukowe PWN, Warsaw.

Recasens, M. and Hovy, E. H. (2011). BLANC: Implementing the Rand Index for Coreference Evaluation. *Natural Language Engineering*, 17:485–510.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D., and Hirschman, L. (1995). A Model-Theoretic Coreference Scoring Scheme. In *Proceedings of the 6th Message Understanding Conference (MUC-6)*, pages 45–52.

Wawer, A., (2015). *Sentiment Dictionary Refinement Using Word Embeddings*, pages 186–193. Springer International Publishing, Cham.

Wiseman, S., Rush, A. M., and Shieber, S. M. (2016). Learning Global Features for Coreference Resolution. *arXiv preprint arXiv:1604.03035*.

Woliński, M. (2014). Morfeusz reloaded. In Nicoletta Calzolari, et al., editors, *Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014*, pages 1106–1111, Reykjavík, Iceland. ELRA.

7. Language Resource References

Maciej Ogrodniczuk. (2015). *Polish Coreference Corpus*. Institute of Computer Science, Polish Academy of Sciences, `http://zil.ipipan.waw.pl/PolishCoreferenceCorpus`.

Presspublica. (2002). *Rzeczpospolita Corpus*. Dawid Weiss, `http://www.cs.put.poznan.pl/dweiss/rzeczpospolita`.

SzegedKoref: A Hungarian Coreference Corpus

Veronika Vincze[1,2], **Klára Hegedűs**[3], **Alex Sliz-Nagy**[4], **Richárd Farkas**[4]
[1]MTA-SZTE Research Group on Artificial Intelligence
vinczev@inf.u-szeged.hu
[2]Department of General Linguistics, University of Szeged
[3]Department of Psychology, University of Szeged
klarahegedus92@gmail.com
[4]Institute of Informatics, University of Szeged
rfarkas@inf.u-szeged.hu
a.sliz.nagy@gmail.com

Abstract

In this paper we introduce SzegedKoref, a Hungarian corpus in which coreference relations are manually annotated. For annotation, we selected some texts of Szeged Treebank, the biggest treebank of Hungarian with manual annotation at several linguistic layers. The corpus contains approximately 55,000 tokens and 4000 sentences. Due to its size, the corpus can be exploited in training and testing machine learning based coreference resolution systems, which we would like to implement in the near future. We present the annotated texts, we describe the annotated categories of anaphoric relations, we report on the annotation process and we offer several examples of each annotated category. Two linguistic phenomena – phonologically empty pronouns and pronouns referring to subordinate clauses – are important characteristics of Hungarian coreference relations. In our paper, we also discuss both of them.

Keywords: coreference, corpus, Hungarian

1. Introduction

In order to avoid unnecessary repetitions and redundancy, speakers can use a wide variety of expressions when referring to the same entity or event in the world. Languages usually offer several lexical and grammatical tools for this purpose. One of the grammatical tools to express identity is coreference, which is used when two (or more) linguistic units refer to the same entity/individual in the world. Coreference relations are most frequently expressed by pronouns, adverbs and nouns (mostly, nouns denoting gender or position such as *girl* or *sergeant*). At the lexical level, it is mostly synonyms that can contribute to lexical variability.

In this paper we introduce the SzegedKoref corpus, in which coreference relations are manually annotated. For annotation, we selected some texts of Szeged Treebank. It is the biggest treebank of Hungarian that contains manual annotation at several linguistic layers (Csendes et al., 2005). We present the annotated texts, we describe the annotated categories of anaphoric relations – pronominal, nominal, adverbial and verbal coreference and subtypes of nominal coreference (hypernyms, synonyms etc.), and we offer several examples of each annotated category. We also mark zero anaphors and pronouns coreferential with subordinate clauses since these are two linguistic phenomena of Hungarian that deserve special attention from the viewpoint of coreference resolution.

2. Related Work

There are several coreference corpora available for many languages, for instance, OntoNotes contains coreference annotation for English, Chinese and Arabic (Weischedel et al., 2011; Pradhan et al., 2007). This database formed the training and test sets of the CoNLL-2011 (Pradhan et al., 2011) and CoNLL-2012 (Pradhan et al., 2012) shared tasks, which aimed at automatic coreference resolution.

There is coreference annotation in the DIRNDL and ANCOR_Centre corpora, containing German and French spoken language data (Muzerelle et al., 2014; Björkelund et al., 2014). As for Japanese, the corpus NAIST Text contains coreference annotation, together with predicate-argument structure (Iida et al., 2007). A large coreference corpus is also available for Polish (Ogrodniczuk et al., 2014; Ogrodniczuk et al., 2013b), moreover, there are annotated coreference corpora for Dutch (Hendrickx et al., 2008) and Czech (Nedoluzhko et al., 2009) as well. Recently, Ghaddar and Langlais (2016) reported on WikiCoref, a coreference corpus of English Wikipedia articles.

A small dataset with manual coreference annotation was earlier published for Hungarian (Miháltz, 2012). In contrast, here we present our large corpus, SzegedKoref, which has been manually annotated for coreference data. Due to its size, the corpus can be used for training and evaluating machine learning-based systems, which is nowadays the most popular approach used for coreference resolution (Pradhan et al., 2012).

In morphologically rich languages like Hungarian, some issues might occur concerning the annotation process of coreference relations. It is the treatment of phonologically empty pronouns that is particularly important among others, as already emphasized for Polish (Ogrodniczuk et al., 2013a). Moreover, pronouns referring to subordinate clauses should also be paid special attention in Hungarian. In our paper, we will focus on both of these phenomena.

3. The Corpus

As the Szeged Corpus (Csendes et al., 2005) contains annotation for several linguistic layers (POS-tags, constituency and dependency syntax), we selected those texts for coreference annotation, in order to enrich their linguistic struc-

ture. Since it is preferred to annotate coreference relations in longer comprehensive texts instead of using very short texts, we also needed to select the appropriate subcorpora of the Szeged Corpus. For this reason, we finally decided to neglect the subcorpus containing short business news, where each piece of news consisted of only 1-2 sentences, hence annotation was not carried out in this subcorpus. Instead, we chose to focus on student essays and newspaper articles, which are comprehensive texts of considerable length and are expected to contain various coreference relations.

3.1. Annotation Principles

During annotation, mentions (i.e. mostly noun phrases that refer to a concept) were first marked, then antecedents were linked to the heads referring to the same entity. The type of coreference is also marked in the data, that is, pronominal, nominal, adverbial and verbal coreference. We also paid attention to derivational anaphors, i.e. cases where the antecedent and the head refer to the same action/entity but belong to different parts of speech (for instance, an action is expressed by a verb first, then it is referred to with a noun or participle). Categories are shown below:

- pronominal anaphor: *Ismertem **a lányt**, **aki** épp átjött az úton.* "I knew **the girl who** was just crossing the street."
- nominal anaphor:
  - Repetition: *Józsi este találkozott **a lánnyal**. **A lány** piros ruhát viselt.* "Joe met **the girl** last night. **The girl** was wearing a red dress."
  - Variant: ***Pálffy János gróf** személyében magyar főparancsnokot neveztek ki a császári sereg élére. **Pálffy** tárgyalásokat kezdett Károlyi Sándor báróval.* "A Hungarian colonel – **Earl János Pálffy** – was chosen to lead the imperial army. **Pálffy** initiated negotiations with Baron Sándor Károlyi."
  - Synonym: *Józsi kapott **egy biciklit**. Másnap **az új kerékpárral** jött munkába.* "Joe got **a new bike**. The next day he came to work with **his new bicycle**."
  - Hypernym: *Az udvaron volt **egy kutya**. **Az állat** keservesen ugatott.* "There was **a dog** in the yard. **The animal** was barking desperately."
  - Hyponym: *Az udvaron volt **egy kutya**. **Szegény uszkár** meg volt kötve.* "There was **a dog** in the yard. **The poor poodle** was tied."
  - Meronym: *Jól játszott **a csapat**, **a kapus** különösen kiemelkedett a mezőnyből.* "**The team** was playing well, **the goalkeeper** especially had an excellent performance."
  - Holonym: *Defektes lett **a jobb első kerék**, így **az autónak** ki kellett állnia a versenyből.* "**The first right wheel** got a puncture, so **the car** had to finish the race."
  - Epithet: ***Józsi** nem tudott bejutni, mert **a szerencsétlen** otthon hagyta a kulcsot.* "**Joe** could not enter the flat because **the poor one** forgot his key at home."
  - Apposition: *Pálffy tárgyalásokat kezdett **Rákóczi megbízottjával, Károlyi Sándor báróval**.* "Pálffy initiated negotiations with **Rákóczi's representative**, **Baron Sándor Károlyi**."
- adverbial anaphor: *Elindultunk **a hotelba**, a többiekkel **ott** találkozunk.* "We have left for **the hotel**, we will meet the others **there**."
- verbal anaphor: *Juli **elénekelt** tegnap egy dalt, ma pedig Józsi is **így tett**.* "Julie **sang a song** yesterday, and Joe **did so** today."
- derivational anaphor: *Józsi mindig **énekel** a fürdőben. **Az éneklés** nagyon zavarja a többi lakót.* "Joe always **sings** in the bathroom. **His singing** annoys the other tenants."

As for nominal anaphors, we also marked their semantic categories, for instance, whether there is a synonymy/hypernymy/holonymy relation between the head and mention (e.g. *kutya* "dog" – *állat* "animal"), whether the head is simply repeated (e.g. *kutya* "dog" – *kutya* "dog") or whether a variant is used (e.g. *Albert Einstein – Einstein*). Derivational relations were also marked between the head and the anaphor (e.g. *Pista hangosan **énekelt**. Az **ének** nagyon zavarta a szomszédját.* "Steve was **singing** loudly. His **song** annoyed his neighbour.").

In Hungarian, zero pronouns also mean a challenge to coreference resolution systems. As the Hungarian verbal paradigm differentiates between verb forms referring to a definite object and verb forms referring to an indefinite one on the one hand, and verbs are also conjugated differently for each person and number on the other hand, there is no need to explicitly mark pronominal subjects and objects in the sentence and so, they can be deduced from context. Furthermore, pronominal possessors might also remain hidden in possessive constructions, due to nominal inflection. From the viewpoint of coreference resolution, all this entails that the anaphor might not be present in the sentence as a separate token, only as a zero pronoun (pro). Thus, before the annotation process started, they had had to be inserted into the text. The following example illustrates this process:

> *Látta a kertjében.* → ***proSUBJ** látta **proOBJ** a **proPOSS** kertjében.*
> see-PAST-3SGOBJ the garden-3SGPOSS-INE
> → proSUBJ see-PAST-3SGOBJ proOBJ the proPOSS garden-3SGPOSS-INE
> "He saw it in his garden."

Here, the words equivalent to the English pronouns *he*, *it* and *his* are missing from the original Hungarian sentence and instead, zero pronouns were automatically inserted into the text before the manual annotation process, so they are also annotated in the data. The insertion took place on the

basis of linguistic rules and morphological and syntactic constraints.
Pronouns referring to subordinate clauses were also marked as coreferent with the subordinate clause they are referring to, no matter they occurred in their overt or zero form. In contrast with English, Hungarian may use a pronoun in the matrix clause that can function as an argument of the main verb and is coreferent with the subordinate clause. For instance, compare these two sentences:

> *Mondtam **proOBJ**, hogy mindjárt itt a karácsony.*
> say-PAST-1SGOBJ proOBJ , that soon here the Christmas
> "I told you that Christmas is almost here."

> ***Azt** mondtam, hogy mindjárt itt a karácsony.*
> it-ACC say-PAST-1SGOBJ , that soon here the Christmas
> "I told you that Christmas is almost here." (lit. "I told **it** to you that Christmas is almost here."

In these sentences, the overt pronoun *azt* and the zero pronoun *proOBJ* were annotated as coreferent with the clause *mindjárt itt a karácsony*.

3.2. Annotation Process

Annotation was carried out by two annotators, who were trained in linguistics and supervised by a linguist expert. The MMAX2 tool was employed for annotation, which allows multilayer annotation and makes it possible to visually track coreference chains during annotation (Müller and Strube, 2006). A sample of the annotated texts is shown in Figure 1.
In order to measure inter-annotator agreement rate, a small sample of 10 documents were annotated by both annotators. Their agreement rate was 0.95 (in terms of F-score), with regard to mention identification.

3.3. Statistical Data

Currently, the corpus contains 309 sentences and 9,782 tokens from the newspaper domain and 3,712 sentences and 45,981 tokens from the student essay subcorpus. Altogether, there are 400 texts, 4021 sentences and 55,763 tokens in the current version of the corpus.
There are 2191 anaphoric chains in the student essay subcorpus and 265 in the newspaper domain, adding up to 2456 anaphoric chains altogether. As shown in Table 1, the most frequent types of anaphor are pronominal anaphors and repetition, indicating that automatic coreference resolution systems should pay extra attention to these categories. Figure 2 tells us that repetitions, hypernyms and adverbial anaphors are much more frequent in the student essays than in the newspaper articles. However, synonyms and appositions are more widely applied in newspaper texts.
The distribution of the anaphoric categories shows a statistically significant difference (χ^2-test, $p<0.01$), hence there are domain differences in the use of anaphoric categories. Later on, we intend to annotate other domains of texts for coreference in order to check what the most characteristic anaphoric categories are for each domain.

Zero pronoun	**Student essays**	**Newspaper**	**Total**
subject	594	119	713
object	181	9	190
possessive	212	128	340
Total	987	256	1243

Table 2: Anaphoric zero pronouns.

Table 2 shows that there are many zero pronouns that form part of an anaphoric chain, what is more, about 67% of pronominal anaphors involve a zero pronoun. Hence, coreference resolution systems should be prepared for the efficient treatment of Hungarian zero pronouns.

4. Possible Uses of the Corpus

Coreference corpora and coreference resolution algorithms might be useful for several purposes. For instance, information extraction systems might exploit coreference relations, since information related to a specific entity might be collected from the text not only by searching for the exact name of the entity but also by finding elements that are coreferent with it.
On the other hand, machine translation applications might also profit from coreference resolution. Although Hungarian does not make use of a grammatical gender for nouns and pronouns, it can be essential to know whether a given pronoun (e.g. *őt* "him" or "her") refers to a male or female person as this information is crucial in finding the proper equivalent of the pronoun in another language that uses grammatical gender. With the antecedent of the pronoun identified, the system may be able to select the personal pronoun of the appropriate gender.
The thorough investigation of types of coreference, as well as the detailed analysis of zero pronouns and pronouns referring to clauses, might be also fruitful for both theoretical linguistics and natural language processing.

5. Conclusions

Here we introduced the SzegedKoref corpus, in which coreference relations are manually annotated. The corpus contains selected texts of Szeged Treebank, the biggest treebank of Hungarian with manual annotation at several linguistic layers. We presented the basic annotation principles and some statistical data on the annotated corpus. Due to its size, the corpus can be exploited in training and testing machine learning based coreference resolution systems, which we would like to implement in the near future.
The corpus is freely available for research and educational purposes at `http://rgai.inf.u-szeged.hu/SzegedTreebank`.

6. Acknowledgements

This research was supported by the project "Integrated program for training new generation of scientists in the fields of computer science", no EFOP-3.6.3-VEKOP-16-2017-0002 and further supported by the EU-funded Hungarian grant EFOP-3.6.1-16-2016-00008. The project has been supported by the European Union and co-funded by the European Social Fund.

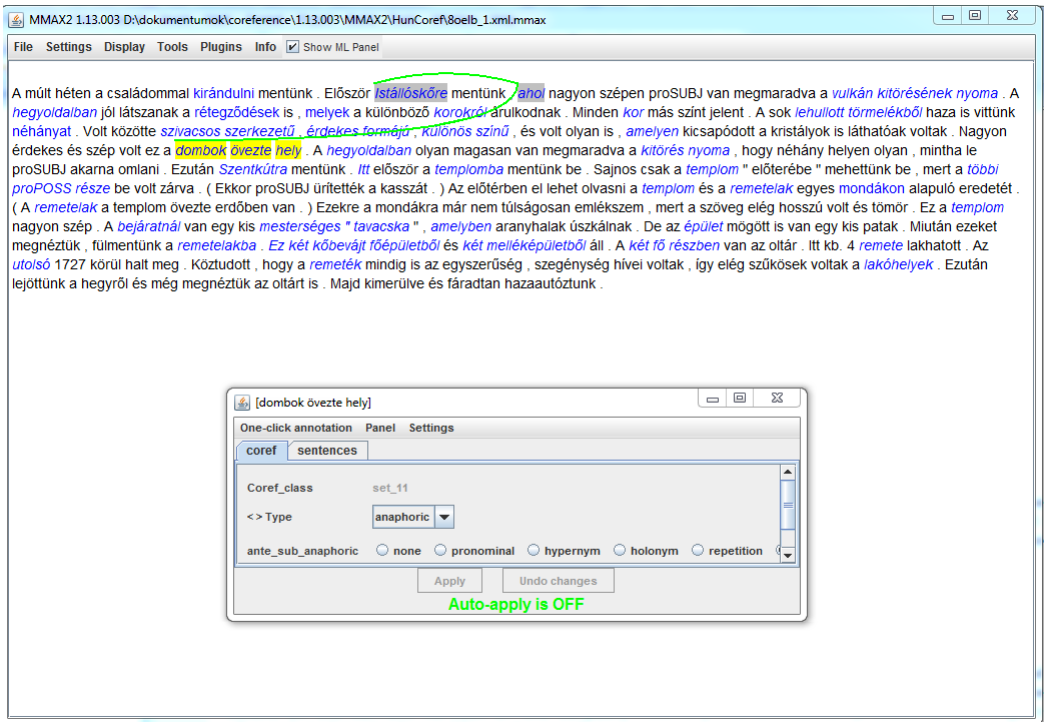


Figure 1: The MMAX2 annotation tool.

Anaphor	Student essays	%	Newspaper	%	Total	%
pronominal	1531	33.51	320	39.22	1851	34.37
repetition	1176	25.74	86	10.54	1262	23.44
synonym	329	7.20	252	30.88	581	10.79
hypernymy	445	9.74	0	0.00	445	8.26
holonymy	350	7.66	34	4.17	384	7.13
epitheton	17	0.37	23	2.82	40	0.74
apposition	117	2.56	70	8.58	187	3.47
adverbial	339	7.42	1	0.12	340	6.31
verbal	5	0.11	0	0.00	5	0.09
derivational	76	1.66	30	3.68	106	1.97
other	184	4.03	0	0.00	184	3.42
Total	4569	100	816	100	5385	100

Table 1: Types and frequency of anaphors.

The research of Richárd Farkas was funded by the János Bolyai Scholarship. Veronika Vincze was supported by the UNKP-17-4 New National Excellence Program of the Ministry of Human Capacities.

7. Bibliographical References

Björkelund, A., Eckart, K., Riester, A., Schauffler, N., and Schweitzer, K. (2014). The Extended DIRNDL Corpus as a Resource for Coreference and Bridging Resolution. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3222–3228, Reykjavik, Iceland. European Language Resources Association (ELRA).

Csendes, D., Csirik, J., Gyimóthy, T., and Kocsor, A. (2005). The Szeged TreeBank. In Václav Matousek, et al., editors, *Proceedings of the 8th International Conference on Text, Speech and Dialogue, TSD 2005*, Lecture Notes in Computer Science, pages 123–132, Berlin / Heidelberg, September. Springer.

Ghaddar, A. and Langlais, P. (2016). Wikicoref: An english coreference-annotated corpus of wikipedia articles. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Hendrickx, I., Bouma, G., Coppens, F., Daelemans, W., Hoste, V., Kloosterman, G., Mineur, A.-M., Vloet, J. V. D., and Verschelde, J.-L. (2008). A Coreference Corpus and Resolution System for Dutch. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Iida, R., Komachi, M., Inui, K., and Matsumoto, Y. (2007).

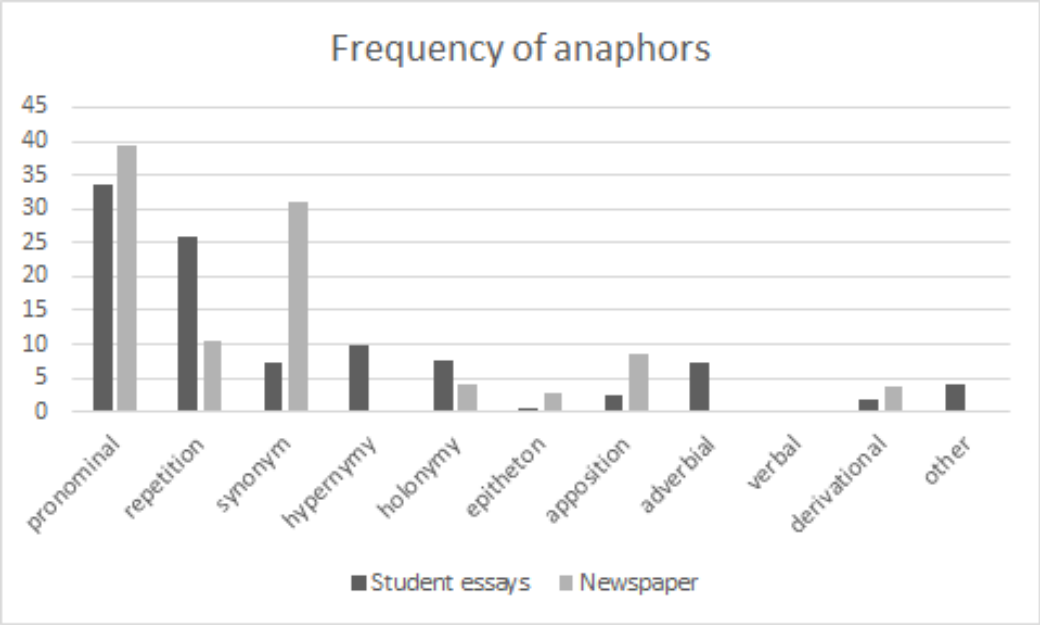


Figure 2: Frequency of anaphor types.

Annotating a Japanese Text Corpus with Predicate-Argument and Coreference Relations. In *Proceedings of the Linguistic Annotation Workshop*, pages 132–139, Prague, Czech Republic, June. ACL.

Miháltz, M. (2012). Tudásalapú koreferencia- és birtokosviszony-feloldás magyar szövegekben. *Általános Nyelvészeti Tanulmányok*, XXIV:151–166.

Müller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. In Sabine Braun, et al., editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a.M., Germany.

Muzerelle, J., Lefeuvre, A., Schang, E., Antoine, J.-Y., Pelletier, A., Maurel, D., Eshkol, I., and Villaneau, J. (2014). ANCOR_Centre, a large free spoken French coreference corpus: description of the resource and reliability measures. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 843–847, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Nedoluzhko, A., Mírovskỳ, J., Ocelák, R., and Pergler, J. (2009). Extended coreferential relations and bridging anaphora in the Prague Dependency Treebank. In *Proceedings of the 7th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2009), Goa, India*, pages 1–16.

Ogrodniczuk, M., Głowińska, K., Kopeć, M., Savary, A., and Zawisławska, M. (2013a). Interesting Linguistic Features in Coreference Annotation of an Inflectional Language. In Maosong Sun, et al., editors, *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, volume 8202 of *Lecture Notes in Computer Science*, pages 97–108. Springer Berlin Heidelberg.

Ogrodniczuk, M., Głowińska, K., Kopeć, M., Savary, A., and Zawisławska, M. (2013b). Polish coreference corpus. In *6th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, volume 3, pages 494–498. Wydawnictwo Poznańskie.

Ogrodniczuk, M., Kopeć, M., and Savary, A. (2014). Polish Coreference Corpus in Numbers. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3234–3238, Reykjavik, Iceland. European Language Resources Association (ELRA).

Pradhan, S. S., Ramshaw, L., Weischedel, R. M., MacBride, J., and Micciulla, L. (2007). Unrestricted Coreference: Identifying Entities and Events in OntoNotes. In *ICSC*, pages 446–453. IEEE Computer Society.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). CoNLL-2011 Shared Task: Modeling Unrestricted Coreference in OntoNotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, CONLL Shared Task '11, pages 1–27, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Weischedel, R., Hovy, E., Marcus, M., Palmer, M., Belvin, R., Pradan, S., Ramshaw, L., and Xue, N. (2011). OntoNotes: A Large Training Corpus for Enhanced Processing. In *Handbook of Natural Language Processing and Machine Translation*.

A Corpus to Learn Refer-to-as Relations for Nominals

Wasi Uddin Ahmad and Kai-Wei Chang
Department of Computer Science, University of California, Los Angeles, California, USA
{wasiahmad, kwchang}@cs.ucla.edu

Abstract

Continuous representations for words or phrases, trained on large unlabeled corpora are proved very useful for many natural language processing tasks. While these vector representations capture many fine-grained syntactic and semantic regularities among words or phrases, it often lacks coreferential information which is useful for many downstream tasks like information extraction, text summarization etc. In this paper, we argue that good word and phrase embeddings should contain information for identifying refer-to-as relationship and construct a corpus from Wikipedia to generate coreferential neural embeddings for nominals. The term *nominal* refers to a word or a group of words that functions like a noun phrase. In addition, we use coreference resolution as a proxy to evaluate the learned neural embeddings for noun phrases. To simplify the evaluation procedure, we design a coreferential phrase prediction task where the learned nominal embeddings are used to predict which candidate nominals can be referred to a target nominal. We further describe how to construct an evaluation dataset for such task from well known OntoNotes corpus and demonstrate encouraging baseline results.

Keywords: Nominals, Coreference, Refer-to-as relation

1. Introduction

Understanding relations between words and phrases is a long-standing problem in natural language processing. Various resources are collected and utilized in order to understand different types of relations between words, including synonymy, antonymy (Turney, 2008) and hierarchical relationships such as hyponymy and hypernymy (Fu et al., 2014). Coreference (a.k.a refer-to-as) relation is another important type of relations between words and phrases and has a wide range of potential applications. Previous work (Feng et al., 2015) found both textual and visual information helpful to learn refer-to-as relations between words. However, annotated data for coreferent phrases are missing. In this work, we develop a dataset from Wikipedia which can aid in learning and evaluating refer-to-as relations between a group of words which act as a noun phrase. Furthermore, the developed dataset can be leveraged to construct semantic space representations for the coreferential nominals.

Learning semantic representations for words from large unlabeled corpora (ex., Wikipedia) using co-occurrence statistics has a long history in natural language processing (Deerwester et al., 1990; Lund and Burgess, 1996; Collobert and Weston, 2008). More recent works (Mikolov et al., 2013; Pennington et al., 2014) uses log-bilinear models to learn continuous representations of words on large corpora efficiently. While these vector representations capture fine-grained syntactic and semantic regularities among words or phrases, it often lacks coreferential information. For example, "phd student" and "graduate fellow" can be co-referred to each other and this relationship should be recognized by semantic representations.

Refer-to-as relation information can benefit many natural language processing applications such as question answering (Morton, 1999), information extraction (Humphreys et al., 1997; Zelenko et al., 2004) etc. So, in this paper, we focus on the task of resolving refer-to-as relation between nominals. We design a coreferential phrase prediction task by simplifying the coreference resolution task to evaluate the utility of our proposed corpus. The automatic resolution of identifying surface forms (a.k.a mentions) which co-refer to the same abstract entity is a challenging task with a long history in computational linguistics. For example, given a paragraph, "A female motorist wearing a blue shirt abruptly made a left turn, ignoring the officer's attempt to initiate a traffic stop. The driver continued to drive erratically to Annapolis Road.", an automatic system has to recognize "A female motorist wearing a blue shirt" and "the driver" refer to the same entity. To deal with such a task, the underlying system has to decide whether two noun phrases are compatible and whether they can narratively replace each other. This requires a high-level understanding of the semantics of words and phrases.

In order to learn representations which can capture the coreferential relationship between nominals, we propose a corpus extracted from Wikipedia. To evaluate the learned representation of nominals from our proposed corpus, we use a simplified form of coreference resolution task and use it as a proxy to evaluate the learned representation of noun phrases. In the following, we provide an example to illustrate the advantage of utilizing coreferential information from Wikipedia in resolving nominal coreference.

Example: The September 11 attacks *by al-Qaeda killed 2,996 people and caused at least $10 billion in property and infrastructure damage. It was* the deadliest incident *for firefighters and law enforcement officers in the history of the United States.*

Resolving the nominal mention, *the deadliest incident* to the mention *September 11 attacks* requires help from external knowledge source. For example, if we know *incident* can be linked to *attacks* from a knowledge source, ex., Wikipedia, then we can resolve the coreference in the provided example. We argue that good representations of noun phrases should contain sufficient information to identify such coreferential relationship. So, we train deep neural networks to learn phrase representations based on our pro-

posed corpus and design a coreferential phrase prediction task to evaluate the learned representations. This evaluation is complementary to common word embeddings evaluation on synonymy, hypernymy and hyponymy (Fu et al., 2014) and is more close to human's perception of word and phrase meanings.

The traditional coreference resolution task is a supervised clustering task. Given a document, a system clusters all the mentions in the article into equivalent classes, such that each class contains mentions refer to the same entity. Despite we can plug-in word and/or phrase representations into a coreference resolution system and evaluate the soundness of the representations based on the end performance of the system but it is hard to measure the contribution of the individual word/phrase representations. As our main goal is to evaluate representations of noun phrases, we specifically focus on the nominals since previous works (Durrett and Klein, 2014) show that resolving nominal mentions are one of the hardest categories in coreference resolution.

Therefore, we propose a ranking evaluation procedure based on the intuition that many state-of-the-art approaches are based on mention ranking model (Denis and Baldridge, 2008). Given a target mention and a list of candidate mentions, the goal in our setting is to rank the mentions in the candidate list based on how likely it is co-referred with the target mention without considering the context. In this way, we can directly compare the contributions of the mention representations in a model. We propose a phrase level task where we consider the entire mention boundaries [1] as a noun phrase. We demonstrate how to construct such mention ranking evaluation data from Ontonotes v5.0 corpus and present the results of preliminary experiments as a proof-of-concept.

2. Dataset Construction

We construct our proposed corpus on the Wikipedia dump from March 06, 2016 which contains 16.4M (approx.) articles where 6.5M (approx.) articles are redirected to another Wikipedia article. We considered rest of the 9.9M non-redirected Wikipedia articles to generate the dataset that will be used to learn the coreferential relationship between nominals. We treat each article on Wikipedia as representing an entity (or concept or idea), and the anchor text of inlinks as a mention of the entity. All the wiki links present in a Wikipedia article are extracted and tagged with appropriate part-of-speech using Stanford log-linear POS tagger. We consider the anchor text of the hyperlinks as mentions. Since we are focusing on nominals, we tag the noun phrases to identify nominals from rest of the noun phrases. Nominals are derived from the noun mentions by following a simple rule, *all non-capitalized nouns are nominals*. For example, *daughter* is counted as a nominal but *Professor* is not. The development tool along with the constructed corpus is publicly available. [2]

Number of articles	16,388,870
Number of redirected articles	6,466,828
Number of non-redirected articles	9,922,042
Unique noun mentions	26,660,798
Unique nominal mentions	2,512,347
Unique nominal mentions ($1 \leq$ mention length ≤ 30)	1,428,441

Table 1: Corpus description extracted from Wikipedia

Additionally, we consider the nominals which do not contain noun phrase but are linked to a Wikipedia article whose title contains noun phrase. For example, *self-governed* mention is linked to an article titled as *self-governance* is counted though *self-governed* is an adjective but *self-governance* is a noun phrase. Finally, nominal mentions with length 1 or more than 30 are filtered out for our experiment. Table 1 lists the complete details of the generated dataset.

3. Learning Phrase Embeddings

We propose to use coreference relationship to evaluate word and phrase embedding. It is important to note that, co-occurrence and coreference are not the same concepts. For example, "lazy dog" and "attractive cat" are very close in the low dimensional embedding space because of high co-occurrence of the words "dog" and "cat" but they cannot be co-referred. On the other hand, "phd candidate" and "graduate student" can be co-referred to each other.

As we mentioned, we reduce the coreference resolution problem to an antecedent ranking problem, since many state-of-the-art models (Wiseman et al., 2015; Chang et al., 2013; Durrett and Klein, 2013) are a variant of the mention-ranking model (Denis and Baldridge, 2008). We define the antecedent ranking problem as, given a target mention, the goal is to rank all antecedent mentions in the same document based on predicted scores such that the antecedent mentions referred to the target mention are ranked on the top. We propose to evaluate phrase embeddings to estimate the usefulness of embeddings to train supervised learning algorithms for antecedent ranking. We train an end-to-end model to produce phrase embeddings based on the training set, tune the model on the development set, and compute accuracy of the ranking based on evaluation dataset. We allow a model to use a pre-trained word embedding for initialization purpose. We construct the training and development dataset from the corpus we generated from Wikipedia and the evaluation dataset based on OntoNotes V5.0 corpus (Hovy et al., 2006) that used in the CoNLL shared task 2012 for coreference resolution (Pradhan et al., 2012). Table 2 lists the complete details of the extracted dataset.

We get coreferential mention clusters from the corpus we generate as described in section 2. Negative examples are

[1] Different corpus may have different definitions of mention boundaries. For example, Ontonotes defines the largest noun phrase that represents an entity as mention, while ACE annotation uses the shortest noun phrase to identify a mention. In this work, we follow the definition in Ontonotes corpus.

[2] https://github.com/wasiahmad/mining_wikipedia/tree/master/WikiMiner

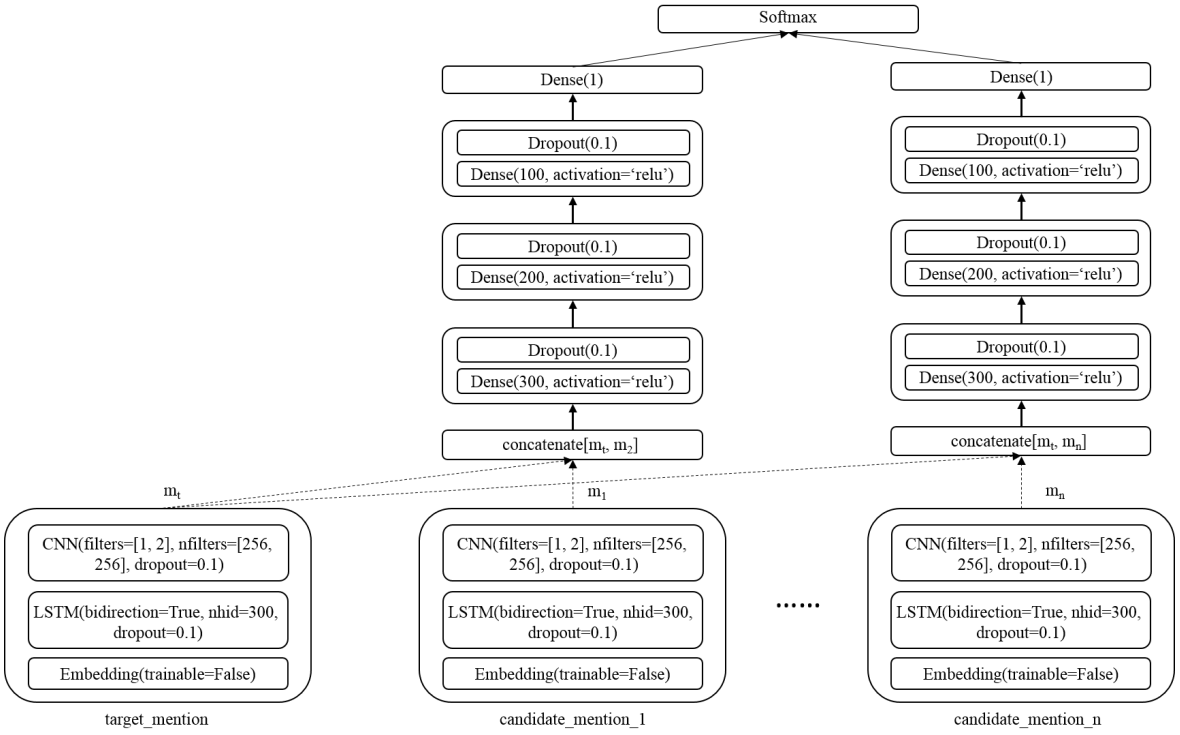


Figure 1: Neural network architecture to rank candidate mentions of a target mention

Train (src: Wikipedia)	Total nominal coref. chain	78,665
	Avg. candidates per chain	24
	Total unique terms	35,939
Development (src: Wikipedia)	Total nominal coref. chain	8,354
	Avg. candidates per chain	18
	Total unique terms	6,686
Test (src: CoNLL)	Total nominal coref. chain	623
	Avg. candidates per chain	12
	Total unique terms	2,839

Table 2: Data Description

selected using random sampling based on cosine similarity distribution. To compute similarity, we simply use pre-trained word embeddings (Pennington et al., 2014). Examples of train/development dataset is provided in table 3.

Baselines: The "Mention Embeddings" (an unsupervised approach) baseline simply take the average of the word vectors in a phrase as the phrase embedding, and compute the cosine similarity to score phrase pairs. Formally, given the embeddings of n_p words of a phrase p, $w_1, \ldots, w_{n_p}$, the phrase embedding $E(p)$ is:

$$E(p) = \frac{1}{n_p} \sum_{k=1}^{n_p} w_k$$

where $E(p)$, $w_k \in R^{d_e}$ and d_e is a hyper-parameter, indicating word embedding size. Then, we compute the similarity score of the phrase pair as follows.

$$\text{Sim}(p_1, p_2) = \text{cosine}(p_1, p_2) = \frac{p_1^T p_2}{||p_1||||p_2||}$$

The "Mention Embeddings + FFNN" baseline construct mention representations like previous baseline but compute score using a two-layer feed-forward neural network.

$$\text{Sim}(p_1, p_2) = \sigma(u^T \tanh(W[E(p_1), E(p_2)] + b))$$

where $W \in R^{d_e \times d_e}, b, u \in R^{d_e}$, and $[E(p_1), E(p_2)]$ represents concatenation of the phrase embedding pair.

Inspired by (Chiu and Nichols, 2016; Ma and Hovy, 2016), we also provide a more sophisticated baseline by using bidirectional long-short term memory (LSTM) (Hochreiter and Schmidhuber, 1997) and convolutional neural network (CNN) (LeCun et al., 1998; Kalchbrenner et al., 2014) in combinations to construct mention representation and using a feed-forward neural network to score mention antecedent pairs. A general architecture for the proposed baseline methods is shown in Fig. 1.

We use a shallow bi-directional LSTM with hidden size $\tilde{h}$ to encode contextual embeddings $\tilde{p}_t$ of each word in the phrase,

$$\overrightarrow{h}_t = LSTM(\overrightarrow{h}_{t-1}, w_t), t = 1, \ldots, n_p$$
$$\overleftarrow{h}_t = LSTM(\overleftarrow{h}_{t+1}, w_t), t = n_p, \ldots, 1$$
$$\tilde{p}_t = [\overrightarrow{h}_t, \overleftarrow{h}_t]$$

Where $\tilde{p}_t \in R^h$ and $h = 2\tilde{h}$.

To construct mention embeddings, we apply convolution operation on the contextual embeddings, $\tilde{p}_t$. We can view the convolution operation as sliding window based feature extractor which captures the n-gram contextual features. A convolution operation involves a filter $w \in R^N (N = nd_e)$, which is applied to a window of n words to produce an n-gram contextual feature. The filter is applied to each possible window of n words in the mention to produce a feature

Target Mention	Positive Candidates	Negative Candidates
protein sequence	amino acid sequencing, chain of amino acids, peptide sequence, protein primary structure	metabolic enzymes, biological mutations, periodic sequence, nucleotide sequence
general election	whole coalition, upcoming election, the previous election, election campaign, legislative election	the constitutional amendment, election win, the presidential election, democratic political values
aerial bomb	aerial bombardment, bombing, bomb attack	nuclear bomb technology, terror attacks, attack ground targets, atomic weapon
highway construction	roads, road building equipment, road work construction, street construction, road building	highway marker, construction yard, railway and highway bridge, construction superintendent

Table 3: Example of positive and negative coreference clusters generated from Wikipedia

Model	NLL-Loss	MAP	P@1	P@5	R@1	R@5
Mention Embeddings	1.7389	**0.5452**	**0.5185**	0.2374	**0.3715**	0.7630
Mention Embeddings + FFNN	1.7836	0.4632	0.4995	0.2317	0.3516	0.7888
Bidirectional-LSTM + CNN + FFNN	**1.6731**	0.4884	0.4719	**0.2475**	0.3476	**0.8025**

Table 4: Performance of baseline methods.

map. Then max pooling is applied over the feature map to select one feature for one filter. In this way, all the contextual features generated by a predefined number of filters are concatenated to produce the mention representation.

A two-layer feed-forward neural network is used to score mention antecedent pairs. Embedding of the mention and it's candidate antecedent are concatenated and given as an input to the feed-forward network. To add non-linearity after each layer, we use rectified linear units in the feed-forward network.

Hyper-parameter Tuning. We carefully tune parameters on the development set. Parameters of the model were learned using mini-batch SGD with Adam (Kingma and Ba, 2014) for optimization with the two momentum parameters set to 0.9 and 0.999 respectively. Initial learning rate was set to 0.001. We use mini-batches of size 64 and early stopping criterion to stop training if validation accuracy does not improve for 3 iterations. For the hidden size of BiLSTM, we consider the range $[150, 300, 600]$ and found 300 results in best performance. We use convolution filters of size 1 and 2 with 256 feature maps each. Gradient clipping technique (5.0) (Graves, 2013) and dropout (0.1) (Srivastava et al., 2014) were used. We use pre-trained GloVe embeddings (Pennington et al., 2014) and fix them during training. Out-of-vocabulary words were initialized with zero vectors.

4. Evaluation Metrics and Baseline Results

In this section, we present the details of evaluation metrics and the performances of baseline methods.

Evaluation Metrics. In tradition, researchers treat coreference resolution as a supervised clustering problem and evaluate system performance by clustering metrics (Pradhan et al., 2014). However, this evaluation metric does not align with our goal of ranking antecedents for a given target mention. Therefore, we evaluate the performance by Mean Average Precision (MAP), Precision at k ($P@k$), Recall at k ($R@k$). We also report the negative log-likelihood loss for the baseline methods.

Results. The performance comparison based on the evaluation dataset between baseline models is presented in Table 4. For our proposed phrase embeddings evaluation task, averaging word vectors is a strong baseline, even without accessing training dataset, it achieves better results to the trained neural network based approaches. This could be due to the noise coming from the unlabeled corpus generated from Wikipedia. We need to deal with the noise to capture a better coreferential relationship between phrases and we are leaving this as our future work.

5. Related Works

Distributed word representations, also known as, word embeddings, typically represent words with dense, low-dimensional and real-valued vectors. Word embeddings have been empirically shown to preserve linguistic information, such as the semantic relationship between words (Mikolov et al., 2013; Pennington et al., 2014) and it helps to learn algorithms to perceive underlying semantics of the targeted task. Over the past few years, researchers have been studying different ways for evaluating word embeddings, including using hypernym-hyponym relation (Fu et al., 2014), word similarity task (Levy and Goldberg, 2014), word analogy taks (Levy et al., 2015), POS tagging task (Lin et al., 2015) and phrase-based machine translation (Zou et al., 2013). Our proposed evaluation approach is complementary to the previous ones. Besides, it is suitable to be used for evaluating phrase embedding.

Our work is inspired by previous works on supervised coreference research which show that incorporating external knowledge can improve the performance of a coreference system (CR). A variety of approaches (Ng, 2007; Ponzetto and Strube, 2006; Haghighi and Klein, 2009) have

been shown to benefit from using external resources such as Wikipedia (Strube and Ponzetto, 2006; Ponzetto and Strube, 2007; Singh et al., 2012; Spitkovsky and Chang, 2012), WordNet (Harabagiu et al., 2001; Soon et al., 2001) or YAGO (Suchanek et al., 2007). (Rahman and Ng, 2011) examined the utility of three major sources of world knowledge and applied them to two learning-based coreference models and found improved performance when knowledge extracted from different sources are exploited in combination rather than individually. Also, previous works (Ogrodniczuk, 2013) verified that nominal facts extracted from world knowledge resources effectively perform and can be used as a source of pragmatic knowledge for coreference resolution.

6. Conclusion and Future Work

In this work, we propose a corpus to learn refer-to-as relations for nominals extracted from Wikipedia in an unsupervised way. Also, we carry out an extrinsic evaluation of phrase embeddings which can aid in resolving nominal coreference. We simplified the coreference resolution problem and presented it as an antecedent ranking task. We have provided several baseline techniques to demonstrate the efficacy of phrase embeddings in resolving nominal mentions. In future, we are interested to investigate, whether word and phrase embeddings can be trained in such a way that learned representations can capture coreferential relationship along with other linguistics regularities and patterns.

Acknowledgements

This work was supported in part by National Science Foundation Grant IIS-1760523 and an NVIDIA Hardware Grant.

7. Bibliographical References

Chang, K.-W., Samdani, R., and Roth, D. (2013). In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Chiu, J. and Nichols, E. (2016). Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Denis, P. and Baldridge, J. (2008). Specialized models and ranking for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 660–669. Association for Computational Linguistics.

Durrett, G. and Klein, D. (2013). Easy victories and uphill battles in coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Seattle, Washington, October. Association for Computational Linguistics.

Durrett, G. and Klein, D. (2014). A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*, 2:477–490.

Feng, S., Ravi, S., Kumar, R., Kuznetsova, P., Liu, W., Berg, A. C., Berg, T. L., and Choi, Y. (2015). Refer-to-as relations as semantic knowledge. In *AAAI*, pages 2160–2166.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *ACL (1)*, pages 1199–1209.

Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Haghighi, A. and Klein, D. (2009). Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1152–1161. Association for Computational Linguistics.

Harabagiu, S. M., Bunescu, R. C., and Maiorano, S. J. (2001). Text and knowledge mining for coreference resolution. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–8. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Humphreys, K., Gaizauskas, R., and Azzam, S. (1997). Event coreference for information extraction. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 75–81. Association for Computational Linguistics.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Levy, O. and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Lin, C.-C., Ammar, W., Dyer, C., and Levin, L. (2015). Unsupervised pos induction with word embeddings. *arXiv preprint arXiv:1503.06760*.

Lund, K. and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, 28(2):203–208.

Ma, X. and Hovy, E. H. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. The Association for Computer Linguistics.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Morton, T. S. (1999). Using coreference for question answering. In *Proceedings of the Workshop on Coreference and its Applications*, pages 85–89. Association for Computational Linguistics.

Ng, V. (2007). Shallow semantics for coreference resolution. In *IJcAI*, volume 2007, pages 1689–1694.

Ogrodniczuk, M. (2013). Discovery of common nominal facts for coreference resolution: Proof of concept. In *Mining Intelligence and Knowledge Exploration*, pages 709–716. Springer.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Ponzetto, S. P. and Strube, M. (2006). Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 192–199. Association for Computational Linguistics.

Ponzetto, S. P. and Strube, M. (2007). Knowledge derived from wikipedia for computing semantic relatedness. *J. Artif. Intell. Res.(JAIR)*, 30:181–212.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 1–40. Association for Computational Linguistics.

Pradhan, S., Luo, X., Recasens, M., Hovy, E., Ng, V., and Strube, M. (2014). Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, June.

Rahman, A. and Ng, V. (2011). Coreference resolution with world knowledge. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 814–824. Association for Computational Linguistics.

Singh, S., Subramanya, A., Pereira, F., and McCallum, A. (2012). Wikilinks: A large-scale cross-document coreference corpus labeled via links to wikipedia. *University of Massachusetts, Amherst, Tech. Rep. UM-CS-2012-015*.

Soon, W. M., Ng, H. T., and Lim, D. C. Y. (2001). A machine learning approach to coreference resolution of noun phrases. *Computational linguistics*, 27(4):521–544.

Spitkovsky, V. I. and Chang, A. X. (2012). A cross-lingual dictionary for english wikipedia concepts. In *LREC*, pages 3168–3175.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Strube, M. and Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, volume 6, pages 1419–1424.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM.

Turney, P. D. (2008). A uniform approach to analogies, synonyms, antonyms, and associations. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 905–912. Association for Computational Linguistics.

Wiseman, S. J., Rush, A. M., Shieber, S. M., and Weston, J. (2015). Learning anaphoricity and antecedent ranking features for coreference resolution. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics.

Zelenko, D., Aone, C., and Tibbetts, J. (2004). Coreference resolution for information extraction. In *Proceedings of the ACL Workshop on Reference Resolution and its Applications*, pages 9–16.

Zou, W. Y., Socher, R., Cer, D. M., and Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398.

Sanaphor++: Combining Deep Neural Networks with Semantics for Coreference Resolution

Julien Plu[◊], Roman Prokofyev[*], Alberto Tonon[*], Philippe Cudré-Mauroux[*], Djellel Eddine Difallah[*], Raphaël Troncy[◊], Giuseppe Rizzo[∞]

[◊]EURECOM, France {julien.plu,raphael.troncy}@eurecom.fr

[*]eXascale Infolab, University of Fribourg {roman.prokofyev,alberto.tonon,philippe.cudre-mauroux,djelleleddine.difallah}@unifr.ch

[∞]ISMB, Italy giuseppe.rizzo@ismb.it

Abstract

Coreference resolution has always been a challenging task in Natural Language Processing. Machine learning and semantic techniques have improved the state of the art over the time, though since a few years, the biggest step forward has been made using deep neural networks. In this paper, we describe Sanaphor++, which is an improvement of a top-level deep neural network system for coreference resolution—namely Stanford deep-coref—through the addition of semantic features. The goal of Sanaphor++ is to improve the clustering part of the coreference resolution in order to know if two clusters have to be merged or not once the pairs of mentions have been identified. We evaluate our model over the CoNLL 2012 Shared Task dataset and compare it with the state-of-the-art system (Stanford deep-coref) where we demonstrated an average gain of 1.13% of the average F1 score.

Keywords: Deep Learning, Coreference Resolution, Entity Linking

1. Introduction

The task of coreference resolution aims to identify which mentions in a text refer to the same real-world entity. Although coreference resolution is mostly studied as a clustering problem, it has also been studied as a Semantic Web problem by using Named Entity Recognition (NER) and Named Entity Linking (NEL) approaches. We define a Semantic Web problem as a problem where we exploit the semantics represented in a knowledge base that is published on the Web. Coreference resolution is an important aspect of text understanding and has numerous applications such as Entity Linking (see, for instance, the first two editions of the Open Knowledge Extraction challenge (Nuzzolese et al., 2015; Nuzzolese et al., 2016)). As an example of coreference resolution, in the following piece of text *"Emmanuel Macron is the new French president. He has been elected with a large majority"*, the mentions *Emmanuel Macron* and *He* will be disambiguated to the same entity, e.g. `http://dbpedia.org/resource/Emmanuel_Macron`. The task of coreference resolution is considered as one of the most challenging in Natural Language Processing (NLP). As an example of its challenging nature, in the following sentence *"Curie shared the 1903 Nobel Prize in Physics with her husband, Pierre Curie"*, it is clear that the clusters {*Pierre Curie*} and {*Curie, her*} are disjoint and do not refer to the same entity, but it is ambiguous whether the pair of mentions *Pierre Curie* and *Curie* are coreferent or not. Actually, without the mention *her*, one does not know if the mention *Curie* refers to *Pierre Curie* or *Marie Curie*.

The contributions of this work are:

1. A new approach that leverages both deep learning and Semantic Web techniques to solve an NLP problem;
2. A model that is integrated into a widely used NLP toolkit, namely the Stanford CoreNLP;
3. A thorough evaluation showing that our technique improves the results over the standard CoNLL2012 Shared Task dataset compared to the state-of-the-art methods by 1.13% in terms of the average F1 score.

2. Related Work

Stanford deep-coref (Clark and Manning, 2016b) takes inspiration from multiple existing methods and implements them using a deep neural network. As a starting point, the framework pre-trains a cluster-ranking model that takes advantage of entity-level information, with a neural mention-ranking model inspired from (Wiseman et al., 2015). In (Wiseman et al., 2016), the authors extend their previous mention-ranking model (Wiseman et al., 2015) by integrating entity-level information taken from the output of a recurrent neural network running over the candidate antecedent-clusters. Nevertheless, this is a simple change with respect to their original mention-ranking model, but not a true clustering model as the deep-coref cluster ranker is. Coreference resolution systems, such as joint inference (McCallum and Wellner, 2003; Poon and Domingos, 2008; Haghighi and Klein, 2010) and those that construct coreference clusters incrementally (Luo et al., 2004; Yang et al., 2008; Raghunathan et al., 2010), integrate entity-level information. Stanford deep-coref takes inspiration from the second kind of system and, particularly, from a combination of cluster-ranking (Rahman and Ng, 2011; Ma et al., 2014) and easy-first clustering strategies (Stoyanov and Eisner, 2012; Clark and Manning, 2015). While most of the previous systems used handcrafted features to integrate linguistic constraints, Stanford deep-coref, in addition, uses a learning-to-search approach inspired from (Chang et al., 2015) in order to learn from data the entity-level distributed representation. Although Stanford deep-coref provides very good results, it often has issues when resolving a coreference that involves entities. For example, in the sentence *"Marie Curie shared the 1903 Nobel Prize in Physics with her husband, Pierre*

Curie", it outputs the following cluster: {*Marie Curie, her, Pierre Curie*} and not the two clusters {*Marie Curie, her*}, {*Pierre Curie*} as expected.

Finally, there also exists a category of coreference resolution approaches that uses structural knowledge from external data sources (Strube and Ponzetto, 2006; Ponzetto and Strube, 2006; Bryl et al., 2010; Uryupina et al., 2011) such as Wikipedia, YAGO or WordNet. SANAPHOR (Prokofyev et al., 2015) belongs to that category and uses DBpdia and YAGO to help disambiguate the different entities that are involved into a coreference cluster. SANAPHOR is plugged to the output of the Stanford decoref (Lee et al., 2011) coreference resolution system, and improves its results by linking the different entities by deciding if a cluster has to be merged with another one, or if it has to be split based on the type, and the link (from YAGO or DBpedia) of the disambiguated entities.

3. SANAPHOR and Stanford deep-coref

This section describes both methods implemented by SANAPHOR and Stanford deep-coref in order to detail their inner-workings and, then, have a clear understanding of the implications in our approach.

3.1. SANAPHOR

SANAPHOR receives as input the clusters of coreferences generated by the Stanford decoref coreference resolution system. Each cluster is a set of mentions extracted from the original text. Each mention comes in the form of a string and, potentially, an associated headword (the most salient word in the mention). The mentions can be either entities, pronouns, or determinants. The resolution then proceeds in two steps by *i)* representing entities with their semantic counterparts whenever possible, and *ii)* optimizing the clusters by merging or splitting them according to their semantic representation. For the first step, it uses an entity linking component in order to link the mentions that might be an entity against DBpedia. It focus on precision rather than recall by doing a strict match over an inverted index over DBpedia, in order to be, as sure as possible, that the mention corresponds to an entity. It uses Wikipedia redirect pages in order to handle the entities that have multiple possible aliases. In case a mention corresponds to an ambiguous entity (i.e. entities associated to a Wikipedia disambiguation page directly), it is discarded. Once a mention is linked, it uses a mapping between DBPedia and YAGO ontologies provided by the TRank Hierarchy (Tonon et al., 2013) to map DBPedia types onto YAGO types.

For the second step, SANAPHOR makes use of the semantic features previously computed in order to optimize the clusters provided by Stanford decoref. The first part of this optimization is the spliting of each cluster by comparing each mention pairwise. A cluster is split following three different cases: *i)* the two mentions being compared have been properly linked against DBpedia and these DBpedia links are different, *ii)* the two mentions being compared have not been properly linked against DBpedia but successfuly typed against the YAGO ontology and those YAGO types are different, or *iii)* over the two mentions being compared, one has been properly linked against DBpedia and the other one against the YAGO ontology, and their YAGO type are different. Since a coreference cluster might also contain non-annotated mentions, they identify the words that belong exclusively to one of the mentions, then assign all the other mentions to one of the new clusters based on the overlap of their words with the exclusive words of each new cluster. After applying these heuristics, new clusters are created and then possibly merged. In the end, clusters are merged *i)* if they share at least one mention that refers to the same entity, or *ii)* if the type of two mentions share the same hierarchy.

3.2. Stanford Deep-coref

Stanford deep-coref basically consists in one big neural network where each component can be seen as three different sub-networks, and where each sub-network has a specific task. These three sub-networks are used to train the cluster-ranking model. The first sub-network is the mention-pair encoder that produces distributed representations for pairs of mentions by passing relevant features through a feed-forward neural network. The input of this sub-network is composed of multiple features that can be grouped in five categories: embedding features, mention features, document genre, distance features and string matching features. More details about the features are given in the original paper (Clark and Manning, 2016b).

This mention-pair encoder is used as a feature for the two other sub-networks: the mention-ranking model, and the cluster-pair encoder. The former scores pairs of mentions by passing their representations (from the mention-pair encoder) through a single neural network layer. The latter produces distributed representations for pairs of clusters by applying a pooling operation over the representations of relevant mention pairs (i.e. pairs where one mention is in each cluster). More precisely, it concatenates the results of max-pooling and average-pooling. The mention-ranking model is pre-trained before the final cluster-ranking model is fed. The final neural network, the cluster-ranking model, is trained with the output of the pre-trained mention-ranking model, and with the cluster-pair encoder. It is important to notice that the mentions are sorted in descending order according to their score from the mention-ranking model before they are used as features. The cluster-ranking model scores pairs of clusters and the scores it produces are leveraged to determine if candidates must be merged or not.

4. Sanaphor++

In this section, we detail the different steps of Sanaphor++ and, in particular, how we extend Stanford deep-coref and SANAPHOR into one single method. The first step is to create a new logic that extends SANAPHOR to take into account ambiguous and novel or emergent entities. The second step is to extend Stanford deep-coref to handle the new logic described in the first step.

4.1. Ambiguous entities

Sanaphor++ is able to handle coreference for ambiguous entities. In SANAPHOR, an ambiguous entity is a mention for which the basic entity linking method finds multiple candidates, such as *Paris* that might refers to: *Paris*

in France, *Paris in Texas*, *Paris Hilton*, *Paris the movie* or even *Paris the band*. To be able to handle those cases, we switch to a more robust entity linking system, ADEL (Plu, 2016), when we encounter an ambiguous entity. ADEL is a hybrid entity linking approach being agnostic to the kind of text (e.g. newswire, tweets, subtitles), the knowledge base used to disambiguate the entities (e.g. DBpedia, Musicbrainz), the type of entities to extract (e.g. Person, Date, Numbers, Location), and the language of the docucment.

4.2. Novel Entities

The second stage is to handle novel or emergent entities, that is, entities that do not exist (yet) in the knowledge base being used, in our experiment, DBpedia. The case of a novel entity occurs when both the SANAPHOR original entity linking method and ADEL give an empty result. In that case, we rely on the Stanford NER annotator that is integrated into the mention extraction process detailed in 4.3.. In that process, mentions come with their NER type attached. The model used to attach these types has been trained with the CoNLL2012 Shared Task dataset that contains the types of mentions defined as entities. Finally, like for the linking that maps DBpedia types to YAGO types, we have defined a mapping that links the NER types to YAGO types. Nevertheless, these NER types are high level types such as *Person* or *Organization* and they do not give a lot of details with respect to their semantics. Despite this lack of information on the deep semantic of these entities, it is still worth to handle because it helps the system to split clusters where cases that mix, for example, *Person* and *Organization*. This is often the case when a company is named according to the family name of the owner. Once these two new cases are handled, we need to find a way to modify Stanford deep-coref in order to make it able to take into account this extended logic: handling ambiguous and novel entities.

4.3. New Mention-pair Ranking Model

The final step is to make Stanford deep-coref able to handle the new logic seen before, to have the final Sanaphor++ pipeline. Thus, after a thorough study of Stanford deep-coref, we have found that it shares two common parts with SANAPHOR: *i)* the cluster merging part and the cluster-ranking model, and *ii)* the cluster splitting part and the mention-ranking model.

The cluster-ranking model has a very complex structure and modifying it to take into account the merging features was impractical, such that we decided to leave this as future work. Therefore, we decided to create a new three-dimensional vector, one dimension for each semantic feature: *i)* if the entities of the two mentions are the same, *ii)* if the type of the two mentions are the same, and *iii)* if the type of one mention is included into the hierarchy of the other one. They are then concatenated with the original vector in the input layer of the mention-pair encoder and then to the mention-pair ranking model.

The extraction of the new features must be implemented in Stanford deep-coref. To do so, we have to put the new process to extract the semantic features directly into the one that takes care of the Stanford deep-coref features. The CoNLL2012 Shared Task dataset is composed of three datasets: train, dev and test. The extraction of the mentions for the train and the set {dev, test} datasets is done differently. About the training dataset, the mentions are directly extracted from the annotated gold standard, whereas for the set {dev, test} datasets, the mentions are extracted using the Stanford mention annotator (see (Clark and Manning, 2015)), where its goal is basically to extract mentions from the text. It means that the annotations in the {dev, test} datasets are not used at all, because they are the evaluation datasets. Once the mentions have been extracted, either via the dataset or via the Stanford mention annotator, they are all put into different feature computation process, one for each kind of feature: embedding, mention, document, distance, string matching and now the semantic feature, i.e. the 3-dimensional vector. In order for the neural network to take into account these new features, we must modify its training objective and more precisely the mistake-specific cost function. The new mistake-specific cost function is represented in Equation 1.

$$\Delta(a, m_i) = \begin{cases} \alpha_{FN} & \text{if} \quad a = NA \wedge \Gamma(m_i) \neq \{NA\} \\ \alpha_{FA} & \text{if} \quad a \neq NA \wedge \Gamma(m_i) = \{NA\} \\ \alpha_{WL} & \text{if} \quad a \neq NA \wedge a \notin \Gamma(m_i) \\ 0 & \text{if} \quad a \in \Gamma(m_i) \vee e_a \in \Omega(m_i) \vee t_a \in \Phi(m_i) \vee T(e_a) \in \Phi(m_i) \end{cases} \tag{1}$$

Where NA indicates an empty antecedent; $\Gamma(m_i)$ denotes the set of true antecedents of m_i (i.e. mentions preceding m_i that are coreferent with it or $\{NA\}$ if m_i has no antecedent); a is a possible antecedent of m_i; e_a denotes the entity of a; $\Omega(m_i)$ denotes the set of the entities of the true antecedents of m_i; t_a denotes the type of a; $\Phi(m_i)$ denotes the set of the types of the true antecedents of m_i (i.e. this set includes also all the types that belongs to the hierarchy of a type); $T(e_a)$ denotes the type of the entity of a. The goal of this new mistake-specific cost function, is to make understand to the network if two mentions are likely to be compatible together and being a pair or not. We detail the clauses of this new function:

1. The first clause stands for the *false new* antecedents. It means that if the antecedent is an empty antecedent and if the set of true antecedents is not equal to empty, then this antecedent is likely to be a wrong pair;

2. The second clause stands for the *false anaphoric* antecedents. It means that if an antecedent is not an empty antecedent and if the set of true antecedents is equal to empty, then this antecedent is likely to be a wrong pair;

3. The third clause stands for the *wrong link* antecedents. It means that if an antecedent is not empty and this antecedent does not belong to the set of true antecedents, then this antecedent is likely to be a wrong pair;

4. The fourth clause stands for a correct coreferent decision. It means that if an antecedent is in the set of the true antecedents, or if the entity of the antecedent is in the set of the true entities, or if the type of the antecedent is in the set of the true types, or if one type

	MUC			B^3			CEAF-E			
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Avg F1**
Sanaphor++	**65.81**	**74.65**	**69.95**	**58.84**	**62.37**	**60.55**	**52.47**	**58.64**	**55.39**	**61.96**
Stanford deep-coref	64.3	72.93	68.34	57.46	60.91	59.14	52.11	58.24	55	60.83

Table 1: Sanaphor++ and Stanford deep-coref results

	BLANC		
	Precision	**Recall**	**F1**
Sanaphor++	**65.88**	**54.97**	**59.93**
Sanaphor	60.63	55.16	57.11
Stanford decoref	60.61	55.07	57.04
Stanford deep-coref	61.48	50.98	55.7

Table 2: Sanaphor++, Sanaphor, Stanford decoref and Stanford deep-coref BLANC results

belonging to the hierarchy of type of the antecedent is in the set of the true types, then this antecedent is likely to be a good pair.

The error penalties α_{FN}, α_{FA} and α_{WL} are hyperparameters that must be defined at the beginning of the training. We keep the values set by the original network respectively (α_{FN}, α_{FA}, α_{WL}) = (0.8, 0.4, 1.0).
Finally, we run the training of this new model with the same hyper parameters than the original Stanford deep-coref. A negative effect of adding these new features into the network is that it increases the training time from 3 days to 5 days. Once the training is done, it is possible to save the model, with the help of scripts provided by the author of Stanford deep-coref, into a format that can directly be used by Stanford CoreNLP toolkit (Manning et al., 2014).

5. Evaluation

5.1. Metrics

Many metrics have been proposed to evaluate the performance of coreference resolution systems, such as *MUC* (van Deemter and Kibble, 2000), B^3 (Bagga and Baldwin, 1998), or *CEAF* (Luo, 2005), including multiple variants of them. *MUC* counts the minimum number of links between mentions to be inserted or deleted when mapping a system response to a gold standard key set. B^3 overcomes the shortcomings of the *MUC* score, instead of looking at the links, it computes precision and recall for all mentions in the document, which are then combined to produce the final precision and recall numbers for the entire output. *CEAF* is calculated based on the best mapping between coreference expressions or entities, thus results in two types of *CEAF*: expression-based (*CEAF-M*) and entity-based (*CEAF-E*). Finally, *Avg F1* is an average of the F1 scores of the three previous ones. Afterward, to have a full comparison among the different systems we will use the most recent metric, BLANC (Recasens and Hovy, 2011).

5.2. Experimental Results and Settings

We evaluate our system on standard datasets from the CoNLL-2012 Shared Task on Coreference Resolution (Pradhan et al., 2012). We compare Sanaphor++ with the most recent version of Stanford deep-coref based on a deep reinforcement learning (Clark and Manning, 2016a) in Table 1. Finally, we compare Sanaphor++, Sanaphor, Stanford deep-coref and Stanford decoref with the BLANC score in Table 2.

The Sanaphor++ model has been exported into a Stanford CoreNLP Framework compliant format, in order to be interoperable and foster the usage through the Stanford CoreNLP Framework. We have run the Sanaphor++ model and the Stanford deep-coref model over the CoNLL2012 test dataset. The provided Stanford deep-coref model in the Stanford CoreNLP Framework is designed for *real-world* usage and gets lower scores than the ones provided in the corresponding paper (Clark and Manning, 2016a), because it does not take into account the CoNLL specific features such as speaker or document genre. For this reason, we have also designed our model to discard those features positioning our experimental setup in the worst experimental setup conditions.

As shown in Table 1, the new semantic logic brought by Sanaphor++ allows to compute a better mention-pair score, as all the scores are improved compared to Stanford deep-coref. While the results are promising, the new logic provides also wrong clusters, such as in the piece of text: *How does* {*the copyright thing*} *work on a Live365 stream? [...] About* {*the JW thing*}*, I work with a couple of them...*, the mentions *the copyright thing* and *the JW thing* have a high pair score since they share common surface forms and the same type, both are novel entities and have been typed to *THING*. The case also appears with ambiguous entities, such as in the piece of text: *It is she who asks if* {*Michael*} *and John could come too. [...] or that can't be tied to* {*Michael Jackson*}, the mentions *Michael* and *Michael Jackson* have a high pair score, because they share a common surface form, the same type and the same link. The first *Michael* refers to the character in Peter Pan which is completely different from the artist *Michael Jackson*. The problem is that *Michael* is highly ambiguous in this sentence.

Results from SANAPHOR have not been reported because it has been built over the Stanford decoref that uses the CoNLL specific features and then cannot be compared with the actual models of Sanaphor++ and Stanford deep-coref.

6. Conclusion and Future Work

We have presented a coreference resolution system that is able to capture the semantic of the entities into a single method based on two different systems that use divergent methods. The newly created model can be used through the Stanford CoreNLP toolkit as it uses the same structure as Stanford deep-coref. Finally, the results have shown improvements over the CoNLL 2012 Shared Task compared to Stanford deep-coref of 1.13% in terms of the average F1 score.

As future work, we want to integrate the merging logic from SANAPHOR into the cluster-ranking model in order to improve the clustering merge decision of this model. We will also investigate the impact on the training time and we will envisage possibilities to reduce it. We would like to evaluate how much the used entity linking system (here, ADEL) impacts the results by using and comparing with other entity linking systems. Finally, our last goal is to have models in other languages such as French.

7. Acknowledgments

We would like to thank Kevin Clark for his help in understanding and using Stanford deep-coref. This work has been partially supported by the French National Research Agency (ANR) within the ASRAEL project (ANR-15-CE23-0018), the French Fonds Unique Interministériel (FUI) within the NexGen-TV project and the innovation activities 3cixty (14523) and PasTime (17164) of EIT Digital (`https://www.eitdigital.eu`).

8. Bibliographical References

Bagga, A. and Baldwin, B. (1998). Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*.

Bryl, V., Giuliano, C., Serafini, L., and Tymoshenko, K. (2010). Using background knowledge to support coreference resolution. In *Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence*.

Chang, K., He, H., III, H. D., and Langford, J. (2015). Learning to search for dependencies. *CoRR*.

Clark, K. and Manning, C. (2015). Entity-centric coreference resolution with model stacking. In *Association for Computational Linguistics (ACL)*.

Clark, K. and Manning, C. (2016a). Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*.

Clark, K. and Manning, C. (2016b). Improving coreference resolution by learning entity-level distributed representations. *arXiv preprint arXiv:1606.01323*.

Haghighi, A. and Klein, D. (2010). Coreference resolution in a modular, entity-centered model. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., and Jurafsky, D. (2011). Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Conference on Natural Language Learning (CoNLL) Shared Task*.

Luo, X., Ittycheriah, A., Jing, H., Kambhatla, N., and Roukos, S. (2004). A mention-synchronous coreference resolution algorithm based on the bell tree. In *ACL*.

Luo, X. (2005). On coreference resolution performance metrics. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*.

Ma, C., Doppa, J., Orr, J., Mannem, P., Fern, X., Dietterich, T., and Tadepalli, P. (2014). Prune-and-score: Learning for greedy coreference resolution. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*.

McCallum, A. and Wellner, B. (2003). Toward conditional models of identity uncertainty with application to proper noun coreference. In *Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web*.

Nuzzolese, A., Gentile, A., Presutti, V., Gangemi, A., Garigliotti, D., and Navigli, R. (2015). The First Open Knowledge Extraction Challenge. In *12^{th} European Semantic Web Conference (ESWC)*.

Nuzzolese, A., Gentile, A., Presutti, V., Gangemi, A., Meusel, R., and Paulheim, H. (2016). The Second Open Knowledge Extraction Challenge. In *13^{th} European Semantic Web Conference (ESWC)*.

Plu, J. (2016). Knowledge Extraction in Web Media: At The Frontier of NLP, Machine Learning and Semantics. In *25^{th} World Wide Web Conference (WWW), PhD Symposium*.

Ponzetto, S. and Strube, M. (2006). Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*.

Poon, H. and Domingos, P. (2008). Joint unsupervised coreference resolution with markov logic. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*.

Prokofyev, R., Tonon, A., Luggen, M., Vouilloz, L., Difallah, D., and Cudré-Mauroux, P. (2015). Sanaphor: Ontology-based coreference resolution. In *International Semantic Web Conference (1)*.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A

multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*.

Rahman, A. and Ng, V. (2011). Narrowing the modeling gap: A cluster-ranking approach to coreference resolution. *J. Artif. Intell. Res. (JAIR)*.

Recasens, M. and Hovy, E. (2011). Blanc: Implementing the rand index for coreference evaluation. *Natural Language Engineering*.

Stoyanov, V. and Eisner, J. (2012). Easy-first coreference resolution. In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, 8-15 December 2012, Mumbai, India*.

Strube, M. and Ponzetto, S. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*.

Tonon, A., Catasta, M., Demartini, G., Cudré-Mauroux, P., and Aberer, K. (2013). Trank: Ranking entity types using the web of data. In *International Semantic Web Conference (1)*.

Uryupina, O., Poesio, M., Giuliano, C., and Tymoshenko, K. (2011). Disambiguation and filtering methods in using web knowledge for coreference resolution. In *FLAIRS Conference*.

van Deemter, K. and Kibble, R. (2000). On coreferring: Coreference in muc and related annotation schemes. *Computational Linguistics*.

Wiseman, S., Rush, A., Shieber, S., and Weston, J. (2015). Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*.

Wiseman, S., Rush, A., and Shieber, S. (2016). Learning global features for coreference resolution. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*.

Yang, X., Su, J., Lang, J., Tan, C. L., Liu, T., and Li, S. (2008). An entity-mention model for coreference resolution with inductive logic programming. In *ACL*.

ANCOR-AS: Enriching the ANCOR Corpus with Syntactic Annotations

Loïc Grobol*†, Isabelle Tellier*, Éric de la Clergerie†, Marco Dinarelli*, Frédéric Landragin*
* Lattice ; † ALMAnaCH, Inria
1 rue Maurice Arnoux, 92120 Montrouge, France ; 2 rue Simone Iff, 75589 Paris, France
{loic.grobol, isabelle.tellier, marco.dinarelli, frederic.landragin}@ens.fr,
eric.de_la_clergerie@inria.fr

Abstract

This paper presents ANCOR-AS, an enriched version of the ANCOR corpus. This version adds syntactic annotations in addition to the existing coreference and speech transcription ones. This corpus is also released in a new TEI-compliant XML format.

Keywords: Coreference, Parsing, Oral Data, TEI

1. Introduction

Since its inception, even as far back as (Hobbs 1986), automatic detection of coreference and anaphora has made consequent use of rich syntactic knowledge. To this day, most of the state-of-the-art systems make use of at least some information inferred from syntax analyses available in coreference-annotated treebanks. Until now, however, no French coreference corpus had publicly available syntactic analysis, which led previous works on automatic coreference detection to either use ad-hoc automatic parsing and shallow syntactic analysis or to use manually annotated mentions, thus avoiding the problem of mention detection.

In an effort to at least start to address this lack of resource, we present ANCOR-AS, an enriched version of ANCOR (Muzerelle et al. 2014) — the current reference corpus for coreference in French — that includes syntactic analysis obtained in the Universal Dependencies framework (Nivre et al. 2016) through state-of-the-art automatic parsing techniques. ANCOR-AS builds upon our previous efforts in (Grobol, Landragin, and Heiden 2017) to develop a standard format for coreference annotations and extends our proposal for TEI-compliant (TEI consortium 2016) reference annotations to the case of syntactic dependencies.

Our preliminary experiments with using automatic syntactic analysis for mention detection, described in (Grobol, Tellier, et al. 2017) give us confidence that in the absence of gold syntactic analysis, automatic parsing is a valuable asset for developing an end-to-end coreference detection system. Moreover, unrelated works on ANCOR, such as Temporal@ODIL (Antoine et al. 2017), are already using semi-automatic parsing as a support for further manual annotations, which suggests that standardized syntactic annotations would be useful for further uses of ANCOR beyond coreference.

2. Context

2.1. Related works

As mentioned earlier, there already exist several coreference treebanks, most notably the AnCoRa corpus (Taulé, Martí, and Recasens 2008) (for Spanish and Catalan), the Prague Dependency Treebank (Nedoluzhko et al. 2016) (for Czech) and the omnipresent OntoNotes (Pradhan, Hovy, et al. 2007). To this day, the latter is still the largest coreference corpus, with nearly 3M words in three languages – Arabic, English and Chinese – and the evaluation standard for coreference detection system since its use for the CoNLL-2011 (Pradhan, Ramshaw, et al. 2011) and CoNLL-2012 (Pradhan, Moschitti, et al. 2012) coreference detection shared tasks. Also relevant for our case is the NXT-Switchboard corpus (Calhoun et al. 2010), a coreference treebank of oral English transcriptions.

For some other languages, only coreference corpora with rich morphological or shallow syntactic annotations exist, such as the Polish Coreference Corpus PCC (Ogrodniczuk et al. 2015) or the EPEC corpus (Soraluze et al. 2012) (for Basque) and, more recently, the Summ-it++ corpus (Fonseca et al. 2016), a coreference corpus of Portuguese enriched with various automatic annotations.

For French however, there is no corpus of either kind, and though the reference treebank for French has been enriched with annotations for named entities (Sagot, Richard, and Stern 2012), this is merely a subset of the annotations needed for coreference.

2.2. Coreference for French and the ANCOR corpus

ANCOR is, for now, the only currently publicly available[1] large-scale coreference-annotated corpus of French, with around 418k words. It is composed of French speech transcriptions, mainly from the ESLO corpus (Baude and Dugua 2011), with coreference and morphosyntactic annotations for noun phrases and pronouns including singleton mentions, but no linguistic annotations of other elements.

More precisely, noun phrases and pronouns in ANCOR are annotated with

- Gender, number and part of speech
- Definiteness (indefinite, definite, demonstrative or expletive form)
- Inclusion or not in a prepositional phrase
- Named entity type (for named entities)
- "NEW" for the first mention of a coreference chain

[1] (Tutin et al. 2000) is another large-scale anaphora corpus but it is not publicly available.

```
<div type="section" xml:id="s2">
  <timeline>
    <when absolute="3.531" xml:id="t2.0"/>
    [...]
  </timeline>
  <u start="#t7.0" who="#spk2" xml:id="u7"
   end="#t7.19">
    [...]
    <tei:w xml:id="u7-w76">au</tei:w>
    <tei:w xml:id="u7-w77">moment</tei:w>
    <tei:w xml:id="u7-w78">où</tei:w>
    <tei:w xml:id="u7-w79">je</tei:w>
    <tei:w xml:id="u7-w80">me</tei:w>
    <tei:w xml:id="u7-w81">suis</tei:w>
    <tei:w xml:id="u7-w82">marié</tei:w>
    <tei:w xml:id="u7-w83">en</tei:w>
    <tei:w xml:id="u7-w84">juillet</tei:w>
    <tei:w xml:id="u7-w85">soixante-sept</tei:w>
  </u>
</div>
```

Figure 1: Speech transcription annotations in ANCOR-AS

```
<standOff>
  <annotation type="coreference">
    <spanGrp type="unit" subtype="mention">
      <span from="#u7-w76" to="#u7-w77"
       xml:id="m31"/>
      <span from="#u7-w84" to="#u7-w85"
       xml:id="m32"/>
    </spanGrp>
    <linkGrp type="relation" subtype="coreference">
      <link target="#m31 #m32" xml:id="r20"/>
    </linkGrp>
    <linkGrp type="schema" subtype="chain">
      <link target="#m31 #m32 #m40" xml:id="c12"/>
    </linkGrp>
  </annotation>
</standOff>
```

Figure 2: Coreference annotations in ANCOR-AS (fragments)

Another coreference corpus for French, Democrat, is under development since 2015 and is planned for release in 2019. It will include various type of documents in written French, including historic texts. The XML-TEI-URS format (Grobol, Landragin, and Heiden 2017), the basis of the format we use for ANCOR-AS, has been developed to provide a standard format in which the Democrat corpus could also be released

3. Interoperable Annotations for Coreference, Syntax and Transcription

The format of ANCOR-AS is based on the XML-TEI-URS format described in (Grobol, Landragin, and Heiden 2017) an example of which is given in fig. 1 and fig. 2.

While the original intend of XML-TEI-URS was to provide standard way of describing coreference relations, it inherits the versatility of its inspiration: the URS metamodel developed by Widlöcher and Mathet (2012) for the Glozz annotation platform. The URS metamodel provides a framework to describe relations between linguistic units with additional characterisations, making it straightforward to describe coreference relations, but can be extended to other types of annotations. In particular, describing dependency syntactic analysis is easy, since dependencies can be seen as simple typed directed relations. The XML-TEI-URS format is a TEI-compliant XML serialisation of the URS metamodel, thus enabling us to encode both coreference and syntactic annotations in a standard and interoperable way. We also chose to make those annotations stand-off, to avoid cluttering the source text too much and clashing with the speech transcription annotations. Figure 3 shows a (partial) example of syntactic annotations in XML-TEI-URS.

More precisely, a given syntactic analysis of part of an utterance in ANCOR is encoded using two layers of annotations. First, a description of the words involved using TEI `<span>` elements, that in our implementation refer to tokens `<w>` in the transcription by their `@xml:id` attribute. The `@n` attribute is also used to denote the word's index (in the Universal Dependencies sense).

The second layer is a representation of the dependency relations between words using `<link>` elements with a `@target` attribute set to `"#head_id #dependent_id"`.

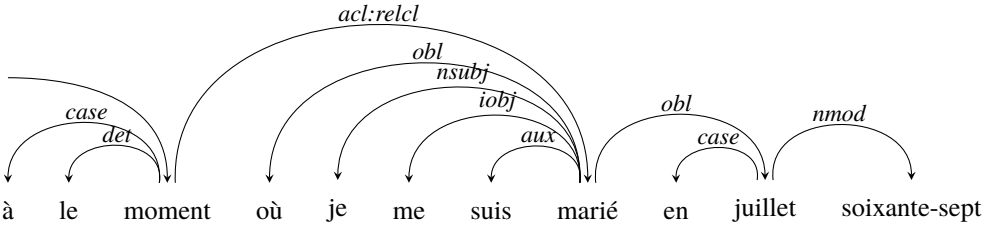
Both of theses layers are complemented by ISO feature structures (ISO 2006) describing the additional informations provided in the Universal Dependency framework, e.g. dependency types, word POS, lemma…

The only significant extension needed to adapt XML-TEI-URS to syntactic annotations was the addition of a way to describe syntactic words that are not directly accessible as a source text span. Figure 3 has an example of such an issue, as the source text token "*au*" has to be expanded into two syntactic words "*à*" and "*le*" to comply with the Universal Dependencies guidelines. These cases are dealt with by using an `<expan>` element to mark the expansion, thus providing `<w>` elements that can then be referred to in the usual way. This mirrors the recommendation of Universal Dependencies for dealing with multiword tokens. This way allows us to keep using the segmentation we use for coreference annotations to link the source text with its syntactic analysis, though it should be noted that other pointing mechanisms, such as those described in (Bański et al. 2016) might have been used, since XML-TEI-URS does not impose any particular one.

4. Enriching ANCOR with automatic syntactic annotations

For the time being, manual syntax analysis of ANCOR is out of reach, and so we settled on using automatic parsing to develop ANCOR-AS. For this iteration, we chose the DYALOG-SRNN parser (De La Clergerie, Sagot, and Seddah 2017) for its good (though unofficial) results for French in the CoNLL 2017 multilingual parsing shared task (Zeman et al. 2017). In particular, some upcoming resources for the syntactic analysis of oral French, such as those coming from the ORFEO project (Debaisieux, Benzitoun, and Deulofeu 2016), might lead us to re-evaluate our choice of parser, or at least to train it on a dataset that includes speech transcriptions.

In an effort to compensate for the fact that this parser has not been designed for speech transcriptions, we are also applying some pre-processing to the corpus in order to make it more easily parsable. Most notably, we filter out purely phatic discourse elements (such as "*hm hm*" or "*oui*"), incomplete words and fillers ("*euh*", "*ben*") before parsing (while of course preserving them in the transcription content) and segment overly-long utterances into sentences using sim-



(a) Syntactic analysis (subtree) for "*au moment où je me suis marié en juillet soixante-sept*"

```
<standOff>
  <annotation type="syntax">
    <div type="tree" xml:id="tree10">
      <div type="multiword-token">
        <expan xml:id="tree10-w6-7" n="6-7" corresp="#u7-w76">
          <w xml:id="u7-w76.1">à</w>
          <w xml:id="u7-w76.2">le</w>
        </expan>
      </div>
      <spanGrp type="unit" subtype="word">
        [...]
        <span target="#u7-w76.1" n="6" xml:id="tree10-w6" ana="#tree10-w6-fs"/>
        <span target="#u7-w76.2" n="7" xml:id="tree10-w7" ana="#tree10-w6-fs"/>
        <span target="#u7-w77" n="8" xml:id="tree10-w8" ana="#tree10-w6-fs"/>
        [...]
      </spanGrp>
      <linkGrp type="relation" subtype="dependency">
        [...]
        <link target="#tree10-w8 #tree-10-w6" xml:id="tree10-d6" ana="#tree10-d6-fs"/>
        <link target="#tree10-w8 #tree-10-w7" xml:id="tree10-d7" ana="#tree10-d7-fs"/>
        <link target="#tree10-w3 #tree-10-w8" xml:id="tree10-d8" ana="#tree10-d8-fs"/>
        [...]
      </linkGrp>
      <div type="dependency-fs">
        [...]
        <fs xml:id="tree10-d7-fs">
          <f name="type"><symbol value="det"></f>
        </fs>
        <fs xml:id="tree10-d8-fs">
          <f name="type"><symbol value="obl"></f>
        </fs>
        [...]
      </div>
      <div type="word-fs">
        [...]
        <fs xml:id="tree10-w6-fs">
          <f name="upostag"><symbol value="DET"></f>
          <f name="Definite"><symbol value="Def"></f>
          [...]
        </fs>
        [...]
      <div>
    </div>
  </annotation>
</standOff>
```

(b) Part of the corresponding XML serialisation

Figure 3: Syntactic annotations in ANCOR-AS

ple heuristics inspired by the work done for the Rhapsodie (Lacheret et al. 2014) and projects.

To ease the use of these syntactic annotations, we also provide some basic links with the coreference annotations by every mention with its syntactic head. It is obvious, though, that an automatic syntactic analysis should not be expected to be perfect, and that, in particular, the manually annotated mention spans might not match perfectly with subtrees in the syntactic analysis, as exemplified by fig. 4. In these cases, we associate mentions with the root of their minimal covering subtree, and annotate as such the dependency relations that we know to be spurious.

Though these syntactic analyses are not perfect (and unsurprisingly so, since automatic parsing of spontaneous speech is still very much an open issue), our experiments in (Grobol, Tellier, et al. 2017) give us hope that they can be of use for automatic coreference detection. Furthermore, from the perspective of the development of a real-world end-to-end coreference detection pipeline, gold-standard syntactic annotations might not be as pertinent, since such a system would still have to be able to use automatic syntactic analysis to deal with unlabeled data. Thus, while we would certainly welcome any effort of manual annotation on ANCOR, we do not consider it an absolute necessity for the avancement of automatic coreference detection for French, especially considering the recent avancements of machine learning techniques for knowledge-poor and inexact data.

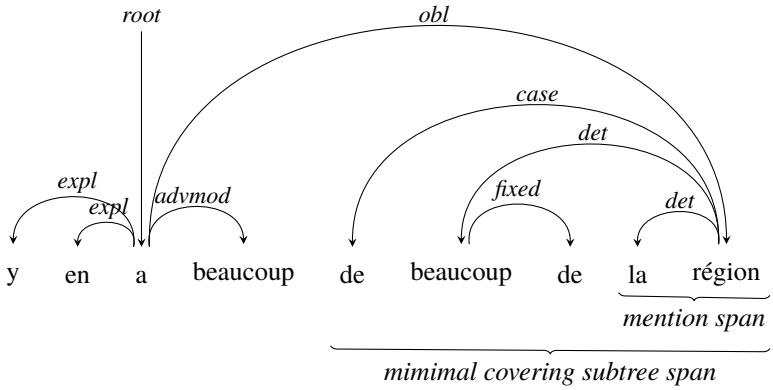


Figure 4: Bad match between syntactic analysis and mention span

5. Conclusion

In this paper, we presented an enriched version of ANCOR, which includes state-of-the-art automatic syntactic analysis and manual coreference, morphosyntactic and speech transcription annotations in a TEI-compliant format. The resulting resource is intended to serve as a stepping stone, both for the development of similar and improved coreference corpora and for the application to French of the most recent automatic coreference detection methods.

Furthermore, the specificities of coreference phenomena in spontaneous speech (such as coreferences in disfluencies, use of spatial deictics…) have not seen much interest from a corpus-based approach. We hope that providing ANCOR – one of the rare spontaneous speech corpora with coreference annotations – in an easier to use and richer format will help researchers explore this topic.

It is also our hope that this work will serve as a proof of feasibility for complex referential linguistic annotations within the TEI guidelines, at least for uses in interchange and archive formats. In this perspective, this application to dependency syntax of the XML-TEI-URS format — initially developed for coreference annotations — proves that this format is versatile enough to be used for a large class of annotation frameworks. For instance, adapting this work to add constituent-based syntactic analysis or temporal annotations (as in other ongoing projects on ANCOR) would not require significant changes to the annotation model we used here.

While further improvements to this linguistic resource are planned, the current version is available at `http://lattice.cnrs.fr/Grobol-Loic` with the same copyleft license as ANCOR (Creative Common BY-SA/BY-NC-SA).

6. Acknowledgements

This work is part of the "Investissements d'Avenir" overseen by the French National Research Agency ANR-10-LABX-0083 (Labex EFL).

This work has been supported by the ANR DEMOCRAT (Description et modélisation des chaînes de référence: outils pour l'annotation de corpus et le traitement automatique) project ANR-15-CE38-0008.

7. Bibliographical references

Antoine, J.-Y. et al. (Sept. 2017). Temporal@ODIL Project: Adapting ISO-TimeML to Syntactic Treebanks for the Temporal Annotation of Spoken Speech. In H. Bunt, editor, *Thirteenth Joint ISO-ACL Workshop on Interoperable Semantic Annotation*. ACL Special Interest Group on Computational Semantics (SIGSEM) and ISO TC 37/SC 4 (Language Resources) WG 2. Montpellier, France.

Bański, P. et al. (Sept. 2016). *Wake up, standOff!* TEI Conference 2016. Wien, Austria.

Baude, O. and Dugua, C. (2011). (Re)faire le corpus d'Orléans quarante ans après : quoi de neuf, linguiste ? *Corpus*. Varia, 10: 99–118.

Calhoun, S. et al. (Dec. 2010). The NXT-format Switchboard Corpus: a rich resource for investigating the syntax, semantics, pragmatics and prosody of dialogue. *Language Resources and Evaluation*, 44.4: 387–419.

De La Clergerie, É., Sagot, B., and Seddah, D. (Aug. 2017). The ParisNLP entry at the ConLL UD Shared Task 2017: A Tale of a #ParsingTragedy. In *Conference on Computational Natural Language Learning*, pages 243–252. Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. Vancouver, Canada.

Debaisieux, J.-M., Benzitoun, C., and Deulofeu, H.-J. (June 2016). Le projet ORFEO: Un corpus d'études pour le français contemporain. *Revue Corpus*. Corpus de français parlé et français parlé des corpus, 15: 91–114.

Fonseca, E. et al. (May 23–28, 2016). Summ-it++: an Enriched Version of the Summ-it Corpus. In N. Calzolari et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož, Slovenia. European Language Resources Association (ELRA).

Grobol, L., Landragin, F., and Heiden, S. (Sept. 2017). Interoperable annotation of (co)references in the Democrat project. In H. Bunt, editor, *Thirteenth Joint ISO-ACL Workshop on Interoperable Semantic Annotation*. ACL Special Interest Group on Computational Semantics (SIGSEM) and ISO TC 37/SC 4 (Language Resources) WG 2. Montpellier, France.

Grobol, L., Tellier, I., et al. (June 2017). Apports des analyses syntaxiques pour la détection automatique de mentions dans un corpus de français oral. In *TALN 2017*. Actes de la 24e Conférence sur le Traitement Automatique des Langues Naturelles (TALN). Association pour le Traitement Automatique des Langues (ATALA). Orléans, France.

Hobbs, J. R. (1986). Resolving Pronoun References. In B. J. Grosz, K. Sparck-Jones, and B. L. Webber, editors, *Readings in Natural Language Processing*, pages 339–352. San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

ISO/TC 37/SC 4 (Apr. 2006). *ISO 24610-1:2006 Language resource management – Feature structures – Part 1: Feature structure representation*. Reference. Geneva, CH: International Organization for Standardization.

Lacheret, A. et al. (May 2014). Rhapsodie: a Prosodic-Syntactic Treebank for Spoken French. In *Language Resources and Evaluation Conference*. Reykjavik, Iceland.

Muzerelle, J. et al. (May 2014). ANCOR Centre, a Large Free Spoken French Coreference Corpus: Description of the Resource and Reliability Measures. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavík, Ísland. European Language Resources Association (ELRA).

Nedoluzhko, A. et al. (May 23–28, 2016). Coreference in Prague Czech-English Dependency Treebank. In N. Calzolari et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož, Slovenia. European Language Resources Association (ELRA).

Nivre, J. et al. (May 23–28, 2016). Universal Dependencies v1: A Multilingual Treebank Collection. In N. Calzolari et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož, Slovenia. European Language Resources Association (ELRA).

Ogrodniczuk, M. et al. (2015). *Coreference in Polish: Annotation, Resolution and Evaluation*. Walter De Gruyter.

Pradhan, S., Hovy, E., et al. (2007). OntoNotes: A Unified Relational Semantic Representation. In *Proceedings of the International Conference on Semantic Computing*, pages 517–526. ICSC '07. Washington, DC, USA. IEEE Computer Society.

Pradhan, S., Moschitti, A., et al. (2012). CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *Proceedings of the joint Conference on Empirical Methods in Natural Language Processing (EMNLP) and Computational Natural Language Learning (CoNLL)*, pages 1–40. CoNLL '12. Jeju, Korea. Association for Computational Linguistics.

Pradhan, S., Ramshaw, L., et al. (2011). CoNLL-2011 Shared Task: Modeling Unrestricted Coreference in OntoNotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27. CONLL Shared Task '11. Portland, Oregon. Association for Computational Linguistics.

Sagot, B., Richard, M., and Stern, R. (June 2012). Annotation référentielle du Corpus Arboré de Paris 7 en entités nommées. In G. Antoniadis, H. Blanchon, and G. Sérasset, editors, *Traitement Automatique des Langues Naturelles (TALN)*. Volume 2 - TALN. Actes de la conférence conjointe JEP-TALN-RECITAL 2012. Grenoble, France.

Soraluze, A. et al. (Sept. 2012). Mention detection: First steps in the development of a Basque coreference resolution system. In J. Jancsary, editor, *Proceedings of KONVENS 2012*, pages 128–136. Main track: oral presentations. Wien, Austria. ÖGAI.

Taulé, M., Martí, M. A., and Recasens, M. (May 2008). AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC-08)*. ACL Anthology Identifier: L08-1222. Marrakech, Morroco. European Language Resources Association (ELRA).

TEI consortium, editor (Dec. 15, 2016). *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. Version 3.1.0. TEI P5: Guidelines for Electronic Text Encoding and Interchange. URL: `http://www.tei-c.org/Guidelines/P5` (visited on 10/09/2017).

Tutin, A. et al. (2000). Annotating a large corpus with anaphoric links. In *Third International Conference on Discourse Anaphora and Anaphora Resolution (DAARC2000)*, page 2. United Kingdom.

Widlöcher, A. and Mathet, Y. (2012). The Glozz Platform: A Corpus Annotation and Mining Tool. In *Proceedings of the 2012 ACM Symposium on Document Engineering*, pages 171–180. DocEng '12. Paris, France. ACM.

Zeman, D. et al. (2017). CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19.

ParCorFull: a Parallel Corpus Annotated with Full Coreference

Ekaterina Lapshinova-Koltunski, Christian Hardmeier, Pauline Krielke
Saarland University, Uppsala University, Saarland University
e.lapshinova@mx.uni-saarland.de, christian.hardmeier@lingfil.uu.se, pauline.krielke@gmail.com

Abstract

In this paper, we describe a parallel corpus annotated with full coreference chains that has been created to address an important problem that machine translation and other multilingual natural language processing (NLP) technologies face – translation of coreference across languages. Recent research in multilingual coreference and automatic pronoun translation has led to important insights into the problem and some promising results. However, its scope has been restricted to pronouns, whereas the phenomenon is not limited to anaphoric pronouns. Our corpus contains parallel texts for the language pair English-German, two major European languages. Despite being typologically very close, these languages still have systemic differences in the realisation of coreference, and thus pose problems for multilingual coreference resolution and machine translation. Our parallel corpus with full annotation of coreference will be a valuable resource with a variety of uses not only for NLP applications, but also for contrastive linguists and researchers in translation studies. This resource supports research on the mechanisms involved in coreference translation in order to develop a better understanding of the phenomenon.

Keywords: coreference, full coreference, cross-lingual coreference resolution, coreference annotation, linguistic annotation, machine translation, multilingual NLP

1. Introduction

We present a parallel corpus containing full coreference annotation that has been created to address an important problem affecting machine translation (MT) and multilingual NLP technologies: translation of coreference across languages. The corpus is available from the LINDAT repository at `http://hdl.handle.net/11372/LRT-2614`.

Texts of various genres often contain recurring references to objects and other discourse entities, realised with a variety of linguistic devices such as noun phrases (NPs), pronouns or other linguistic means. Devices referring to the same entity are said to corefer. The coreference relation is shared across all languages. However, languages differ considerably in the range of linguistic means triggering this relation (Kunz and Steiner, 2012; Kunz and Lapshinova-Koltunski, 2015; Novák and Nedoluzhko, 2015). The choice between these referring expressions is governed by language-specific constraints. Differences in their realisation give rise to transformation patterns used to create coherent translations. In translation, references in the source language (SL) must be rendered with appropriate linguistic devices from the repertoire of the target language (TL), with different constraints. For instance, pronouns and adjectives in German (DE) are subject to grammatical gender agreement, whereas in English (EN), only person pronouns have this marking and adjectives are unmarked.

Recent research in multilingual coreference and automatic pronoun translation has led to important insights into the problem and some promising results, but a working solution for coreference translation in an end-to-end MT has not been demonstrated yet. Research on automatic coreference translation has been restricted to pronouns, but the phenomenon is by no means limited to anaphoric pronouns. Example (1) illustrates a coreference chain expressing a relation of comparison, where we have a closed class of explicit markers for establishing this type of relation in English (*imaginary ones*). German, however, uses an elliptical noun phrase (*imaginäre [...]*).

(1) *...I would make an effort to tell them we have real sciences, hard sciences, we don't need [imaginary ones]. - ...ich würde mir extra Mühe geben, ihnen zu erzählen, dass wir richtige Wissenschaften haben, hieb- und stichfeste Wissenschaften, wir brauchen [keine imaginären].*

Moreover, negation in German can be expressed not only with the adverb *nicht*, but also with the indefinite pronoun *kein*. This pronoun changes its form depending on the case, number and gender (*keine* in example (1) is a plural accusative), which also influences the form of the following adjective: *imaginären* (plural accusative). This form is dependent on the antecedent (Wissenschaften) of the nominal ellipsis. A translation error such as an incorrect inflection (*imaginäre/imaginäres/imaginärer*) may destroy the coreference chain. Interpretating referential expressions is therefore essential for correct translation.

Even where the systemic options for coreference devices coincide, we can find frequent alternations in the use of demonstrative pronouns in German as in example (2).

(2) *We work for prosperity and opportunity because [they]'re right. [It]'s the right thing to do. – Wir arbeiten für Wohlstand und Chancen, weil [das] richtig ist. Wir tun [damit] das Richtige ('We work for prosperity and opportunity because [that] is right. We do [thereby] the right').*

The English example uses the personal pronouns *they* and *it* to refer to the entities *prosperity* and *opportunity* in the first case, and to the event *working for prosperity and opportunity* in the second. The example translation from a parallel corpus uses the demonstrative *das* and the demonstrative deictic *damit*, referring to the event *working for prosperity and opportunity* in both cases, but encoding an additional

logico-semantic relation of instrumentality in the second. This is one of the typical cases of translation between English and German where the coreference relation as such is preserved, but it is not coreference between exactly the same entities in both cases, and it is semantically enriched by the instrumental relation in the second. An MT system is very likely to output the personal pronoun *sie* instead of *das* (*Wir arbeiten für Wohlstand und Chancen, weil [sie] richtig sind*) making this pronoun refer to the entities and not the event, which would sound less natural in German.

2. Related Work

The challenge of translating pronouns has been a recurring topic in recent studies. There are a few corpus-based studies of coreference translation (Novák and Nedoluzhko, 2015; Novák et al., 2013; Guillou and Webber, 2015). For the languages under analysis, it has been empirically shown to be a relevant problem (Hardmeier and Federico, 2010; Guillou, 2016). In the MT community, the awareness of the problem has been increased with three recent shared tasks on pronoun translation (Hardmeier et al., 2015; Guillou et al., 2016; Loáiciga et al., 2017). In recognition of the difficulty of the problem, test suite-based evaluation methods for pronoun translation have been proposed (Guillou and Hardmeier, 2016; Bawden et al., 2017).

At the same time, coreference translation and multilingual coreference resolution is still a complex problem, as we observe a widespread lack of understanding of this phenomenon. Existing coreference resolution tools are known to be unreliable as they introduce an unacceptable number of errors, and therefore manually annotated parallel resources are absolutely indispensable for the development of coreference-aware MT systems and other multilingual language technologies, including cross-lingual coreference resolution (Grishina, 2017; Novák and Žabokrtský, 2014; Green et al., 2011), information extraction (Lee et al., 2012; Zelenko et al., 2004) and question answering (Morton, 1999; Hartrumpf et al., 2008). Most existing coreference corpora are not parallel. The only resources for the language pair English-German that are known to us include the GECCo corpus (Lapshinova-Koltunski and Kunz, 2014), the ParCor corpus (Guillou et al., 2014) and the multilingual coreference corpus described by (Grishina and Stede, 2015). The first corpus contains annotations of the source texts only and is available with restrictions on some texts. The second resource considers only pairwise annotation of anaphoric pronouns and their antecedents. The third corpus, although containing annotations of all referring expressions appearing in a coreference chain, is very small (ca. 11,000 words per language).

For this reason, we have created an English-German parallel corpus which contains annotation of full coreference chains on the basis of the ParCor corpus. The annotation scheme takes inspiration from the schemes used in all the three resources mentioned above (Kunz, 2012; Guillou et al., 2014; Grishina and Stede, 2016). In contrast to other existing coreference schemes that were designed for monolingual datasets, these were elaborated for a multilingual corpus and will allow us to obtain uniform nominal coreference annotations which facilitates extension to further languages in the future.

3. Annotation Categories

A detailed description of categories and disambiguation rules are needed to guarantee consistency throughout the whole process of annotation. Our annotation guidelines are based on the three existing ones described by Grishina and Stede (2016), Guillou et al. (2014) and Kunz (2012). They address the segmentation of nominal elements, the annotation of different antecedent and anaphora types and examples of various problematic cases (Lapshinova-Koltunski and Hardmeier, 2017).

Segmentation Annotated elements (markables) include: Pronouns, nouns, nominal phrases or elliptical constructions that are parts of a coreference pair (antecedent-anaphora), as well as verbal phrases or clauses being antecedents of event anaphora.

Types of antecedents In our framework, we define two different types of antecedents: entities and events. Entities can either be represented by a pronoun or an NP. Events can be represented by a VP as in (3-a), a clause as in (3-b) or a set of clauses. Antecedents can be split as in (3-b) (multiple elements "prosperity and oportunity" constitute one antecedent – all components of the antecedent are linked to the referring expression "they"). If there is no explicit antecedent (in some cases, a referring expression is anaphoric, but no specific antecedent can be found in the text) the position of the antecedent is left open.

(3) a. *... you have to basically [combine everything you learned from project one and project two]. ultimately [that]'s the goal .*
b. *[We work for [prosperity] [and opportunity]] because [they]'re right. [It]'s the right thing to do.*

Types of anaphora We include two types of referring expressions (anaphors) into our analysis: Pronouns and nominal phrases. Coreferring **pronouns** include personal, demonstrative, relative and reflexive pronouns. Note that demonstrative pronouns may also refer to locations (*there, here*) and time (*then, now*). We also include pronominal adverbs in the category of demonstrative pronouns. Pronominal adverbs are formed by replacing a preposition and a pronoun, like *gegen+das* → *dagegen* in example (4). They exist in both English and German, but are used differently. In German, they are very common, but in English, they sound rather archaic and are generally avoided. adverbs are not considered in most coreference annotation schemes. However, they constitute around 8% of all referring expressions in the German language[1] and are especially frequent in spoken and spoken-like language.

(4) *Viele Amerikaner haben Probleme mit [Rassismus]; doch wir sind [dagegen] immun.*

Coreferring **nominal phrases** include proper names (*Herr Almeida Freire* in example (5-a)), nominal premodifiers as

[1]This number is based on the annotations available in the German part of the GECCo corpus(Lapshinova-Koltunski and Kunz, 2014).

in (5-b), full nominal phrases (used with a definite article or a demonstrative modifier as in example (5-c)) and nominal phrases with quantifiers (*all people* in the meaning *all these people*). Generic nouns can co-refer with definite full NPs or pronouns, but not with other generic nouns, see (5-d).

(5) a. *In [seiner] EWSA-Stellungnahme zum "Bericht der Kommission zur Beobachtung des Handelsmarktes" schreibt [Herr Almeida Freire]...*
b. *The unionists used to be [[EU] supporters], but now they are questioning how [it] has developed...*
c. *This past spring, the U.S. Department of Education issued [a report, The Condition of Education 2000]. [The report] found that...*
d. *[Computers] are expensive. But [they] are useful. Computers cost a lot of money.*

As shown in example (1) above, linguistic chains may also include *substitution* and *ellipsis* in addition to referring expressions. These trigger a type reference relation (as opposed to a relation of identity) between referents belonging to the same class (Kunz and Steiner, 2013; De Beaugrande and Dressler, 1981). In substitution patterns, the referring expression is replaced with another element (example (6-a)). In ellipsis, it is completely left out, and the reference is implicit (example (6-b)).

(6) a. *Do you prefer the blue shirt or [the red shirt]? – I would like the red [one].*
b. *...if I take any one of these balls... and I count how many [neighboring balls] that there are around it, the answer's always twelve [].*

Substitution and ellipsis are mostly analysed within separate chains in other studies. We include them into our framework, since they often occur in similar contexts as coreference if considered cross-lingually, as again was shown in example (1). In our framework, substitution and ellipsis are subdivided into their structural types, according to the omitted/substituted element: nominal ellipsis (7-a), verbal substitution (7-b) and clausal substitution in (7-c).

(7) a. *You might have to come up afterwards to count but if I take any one of these balls in the middle and I count how many [neighboring balls] that there are around it, the answer's always twelve [].*
b. *You'll see that it had [to accommodate] an incredible range of functions much more elaborate than any temple or palace in the past would have [done].*
c. *[Does everybody have a handout, for today]? If [not] Aaron's got handouts.*
d. *[So, well, any more questions]? – [no], okay, ...*
e. *[How many slices do you want]? - "[Two]", I said.*

Following Menzel (2017), we also define two additional classes for ellipsis: yes-no type as in (7-d) and mixed type (a combination of nominal and verbal or clausal) as illustrated in (7-e).

Another category that is considered here but is excluded from most analyses is that of comparative reference, which does not trigger co-reference in the strict sense. Together with other cases (substitution and ellipsis) it instead involves type reference, co-classification or "sloppy identity", see (Kunz and Steiner, 2012). The linguistic means signaling comparative reference include such words as *same, equal, identical* or particular adjectives in the comparative form. We distinguish between general and particular comparison, the first referring to a general relation of comparison between two entities (8-a) and the latter referring to particular comparative features of two entities (8-b).

(8) a. So what do you think happened to [these design students] ? [...] We did the exercise again with [the same students] .
b. That car over there is very [fast] . But well, my uncle drives an even [faster] one.

4. Annotation Process

The annotations were created with the annotation tool MMAX2, including all of the above mentioned categories. The annotation scheme created for this task allows the annotator to define each markable as a certain mention type (pronoun, NP, VP or clause). The mentions can be defined further in terms of their cohesive function (antecedent, anaphoric, cataphoric, comparative, substitution, ellipsis, extratextual, pleonastic-it, apposition). Antecedents can either be annotated as simple or split, and as entity or event. For anaphoric expressions the scheme includes singular/plural agreement with the antecedent and subject/non-subject position of the expression. The annotation scheme also covers pronoun type (personal, possessive, demonstrative, reflexive, relative) and modifier types of NPs (possessive, demonstrative, definite article, or none for proper names). An example of the MMAX2 interface with a visualisation of a coreference chain is illustrated in Figure 1.

All annotations were performed by highly experienced well-trained annotators with linguistic background in order to ensure maximum accuracy.

5. Data Selection

We used existing resources and extended them with: (1) complete annotation of full coreference chains; (2) additional referring expressions to achieve full coreference chains. The resources we are using as a basis include the ParCor corpus (Guillou et al., 2014) and the dataset used for the DiscoMT workshop shared task (Hardmeier et al., 2015). To increase the variety in register and genre, we included some additional data taken from the test sets of the news translation shared task at the Conference on Machine Translation (Bojar et al., 2017, WMT17). Table 1 provides an overview of the total number of tokens included into our corpus.

For the DiscoMT dataset, the existing annotations covered only English. We extended the existing dataset into a parallel corpus by added the corresponding translation from

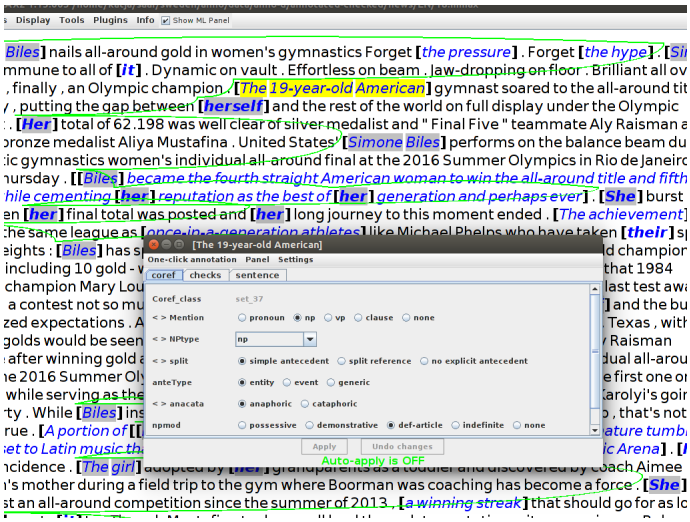


Figure 1: A coreference chain visualised in MMAX2

language	ParCor	DiscoMT	WMT news	total
English	31,971	39,764	10,644	82,379
German	30,305	37,452	10,593	78,350
total	62,276	77,216	21,237	160,729

Table 1: Corpus data

English into German. Overall, we completed the annotations by adding all types of referring expressions 71,735 tokens of the English data and 30,305 of the German data. Around 48,000 tokens of data in German (translations of the English TED talks contained in the DiscoMT data and the news texts from the WMT data), as well as 10,644 tokens of the English data did not contain any annotations and were thus annotated from scratch. The total number of tokens in the annotated corpus amounts to ca. 160,000.

The annotated resource that we have created represents a reasonably-sized data set for training coreference resolution components that can be used for MT or other cross-lingual applications. It is comparable in size (with a larger amount of text, but fewer annotated mentions) to the ARRAU corpus (Poesio and Artstein, 2008), which features a similarly rich coreference annotation and covers a greater variety of genres, but does not include multilingual parallel text. Although the amount of data is not enough to train an MT system, this dataset will be large enough for MT tuning, testing and evaluation, which is an important improvement over the existing data situation.

6. Annotation Results

We present an overview of the annotated structures (absolute numbers) in Table 2 below.

In total, the corpus contains about 15,000 annotated mentions at the moment. The annotated mentions are classified according to their morpho-syntactic type: pronouns (pronoun), nominal phrases (np), verbal phrases (vp) and clauses (clause). This differentiation was introduced for a practical reason, as it permits classifying mentions further according to their function or the role in a coreference chain. As seen from the table, German texts contain more markables, i.e. more referring expressions.

	English	German	total
pronoun	4,650	4,269	8,919
np	2,485	2,611	5,096
vp	133	132	265
clause	335	312	647
total mentions	7,603	7,324	14,927

Table 2: Annotated mentions and their subcategories

	English	German	total
number of chains	2,319	2,425	4,744
average chain length	2.94	2.81	2.87

Table 3: Annotated chains

The number of full coreference chains in the data amounts to 4,744 (see Table 3). We also calculate the average chain length (total number of mentions/total number of chains). The German translations contain more chains than their English sources, but on average, these chains are shorter.

To evaluate the reliability of the annotated coreference chains, we created a second annotation of two files in each language. The inter-annotator dataset included TED talks 785 and 790 and was composed of 6,253 English and 5,975 German tokens. As a measure of inter-annotator agreement, we computed the mention overlap and entity-based CEAF scores (Luo, 2005) between the two annotations, treating our regular annotator as the hypothesis to be evaluated and

the second annotator as the reference. The scores were calculated with the CoNLL reference scorer implementation (Pradhan et al., 2014) and are shown in Table 4 as a macro-average over the two documents.

	Precision	Recall	F-score
English			
mentions	89.20%	73.89%	80.71%
CEAF*e*	82.90%	67.13%	74.13%
German			
mentions	84.80%	69.76%	76.54%
CEAF*e*	72.53%	60.36%	65.88%

Table 4: Inter-annotator metrics for coreference chains

As seen from the table, we observe a better agreement for the English texts. We suppose that the reason for the greater disagreement for German texts is the complexity of the linguistic structures triggering coreference in this language. However, a more detailed analysis of the agreement results is needed to understand the reasons. We plan to do this in future work.

We also performed automatic inconsistency checks to prove if the annotated data contains any (1) marked mentions outside of chains; (2) antecedents of chains that are not marked as first elements of chains; (3) some other error types. The detected errors were then corrected by the annotators. Besides that, the annotator added the following categories (that were not included into the annotation scheme at the very beginning): (a) bare nouns; (b) indefinite nouns and (c) quantifiers (both) as demonstratives.

7. Conclusion and Future Work

The differences in coreference realisation in multiple languages present a huge challenge to machine translation and are of interest for contrastive linguists and researchers in translation studies. A parallel corpus with full annotation of coreference is a valuable resource with a variety of uses. The corpus will help us study the mechanisms involved in coreference translation in order to develop a better understanding of the phenomenon. It will serve as a resource for creating and evaluating coreference-aware MT systems without having to rely on notoriously inaccurate automatic coreference resolvers. Finally, it can also be used as a training and development resource for the creation of multilingual or monolingual coreference resolution systems. Moreover, we address the demand for better approaches to evaluate complex linguistic phenomena that are not covered by existing annotation schemes.

8. Acknowledgements

The annotation work was funded by the European Association for Machine Translation. Christian Hardmeier was supported by the Swedish Research Council under grants 2012-916 and 2017-930.

9. Bibliographical References

Bawden, R., Sennrich, R., Birch, A., and Haddow, B. (2017). Evaluating discourse phenomena in neural machine translation. *ArXiv e-prints*, 1711.00513.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi, M. (2017). Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark, September. Association for Computational Linguistics.

De Beaugrande, R.-A. and Dressler, W. U. (1981). *Einführung in die Textlinguistik*. Niemeyer, Tübingen.

Green, S., Andrews, N., Gormley, M. R., Dredze, M., and Manning, C. D. (2011). Cross-lingual coreference resolution: A new task for multilingual comparable corpora. Technical Report 6, HLTCOE, Johns Hopkins University.

Grishina, Y. and Stede, M., (2016). *Parallel coreference annotation guidelines.*

Grishina, Y. (2017). Combining the output of two coreference resolution systems for two source languages to improve annotation projection. In *Proceedings of the Third Workshop on Discourse in Machine Translation (DiscoMT), EMNLP 2017*, Copenhagen, Denmark.

Guillou, L. and Hardmeier, C. (2016). PROTEST: A test suite for evaluating pronouns in machine translation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 636–643, Portorož, Slovenia. European Language Resources Association (ELRA).

Guillou, L. and Webber, B. (2015). Analysing ParCor and its translations by state-of-the-art SMT systems. In *Proceedings of the Second Workshop on Discourse in Machine Translation (DiscoMT), EMNLP 2015*, Lisbon, Portugal.

Guillou, L., Hardmeier, C., Smith, A., Tiedemann, J., and Webber, B., (2014). *ParCor 1.0: Pronoun Coreference Annotation Guidelines*. Edinburgh, Uppsala.

Guillou, L., Hardmeier, C., Nakov, P., Stymne, S., Tiedemann, J., Versley, Y., Cettolo, M., Webber, B., and Popescu-Belis, A. (2016). Findings of the 2016 WMT shared task on cross-lingual pronoun prediction. In *Proceedings of the First Conference on Machine Translation (WMT16)*, Berlin, Germany.

Guillou, L. (2016). *Incorporating Pronoun Function into Statistical Machine Translation*. Ph.D. thesis, School of Informatics. University of Edinburgh.

Hardmeier, C. and Federico, M. (2010). Modelling pronominal anaphora in statistical machine translation. In *Proceedings of the IWSLT (International Workshop on Spoken Language Translation)*, pages 283–289, Paris, France.

Hardmeier, C., P., Nakov, S., Stymne, J., Tiedemann, Y., Versley, and Cettolo, M. (2015). Pronoun-focused MT and cross-lingual pronoun prediction: Findings of the 2015 DiscoMT shared task on pronoun translation. In

Proceedings of the Second Workshop on Discourse in Machine Translation (DiscoMT), EMNLP 2015, pages 1–16, Lisbon, Portugal.

Hartrumpf, S., Glöckner, I., and Leveling, J. (2008). Coreference resolution for questions and answer merging by validation. In Carol Peters, et al., editors, *Advances in Multilingual and Multimodal Information Retrieval: 8th Workshop of the Cross-Language Evaluation Forum, CLEF-200, Budapest, Hungary, September 19-21, 2007*, pages 269–272. Springer, Berlin, Heidelberg.

Kunz, K. and Lapshinova-Koltunski, E. (2015). Cross-linguistic analysis of discourse variation across registers. *Special Issue of Nordic Journal of English Studies*, 14(1):258–288.

Kunz, K. and Steiner, E. (2012). Towards a comparison of cohesive reference in English and German: System and text. In M. Taboada, et al., editors, *Contrastive Discourse Analysis. Functional and Corpus Perspectives*. Equinox, London.

Kunz, K. and Steiner, E. (2013). Cohesive substitution in English and German: A contrastive and corpus-based perspectivet. In Karin Aijmer et al., editors, *Advances in Corpus-Based Contrastive Linguistics. Studies in honour of Stig Johansson*, pages 201–232. John Benjamins, Amsterdam.

Kunz, K., (2012). *Richtlinien für die Korrektur von kohäsiven Referenzmitteln*.

Lapshinova-Koltunski, E. and Hardmeier, C., (2017). *Coreference Corpus Annotation Guidelines*, December.

Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 489–500, Jeju Island, Korea.

Loáiciga, S., Stymne, S., Nakov, P., Hardmeier, C., Tiedemann, J., Cettolo, M., and Versley, Y. (2017). Findings of the 2017 DiscoMT shared task on cross-lingual pronoun prediction. In *Proceedings of the Third Workshop on Discourse in Machine Translation (DiscoMT), EMNLP 2017*, pages 1–16, Copenhagen, Denmark.

Luo, X. (2005). On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 25–32, Vancouver, British Columbia, Canada.

Menzel, K. (2017). *Understanding English-German contrasts: a corpus-based comparative analysis of ellipses as cohesive devices*. Ph.D. thesis, Universität des Saarlandes, Saarbrücken.

Morton, T. S. (1999). Using coreference for question answering. In *Proceedings of the Workshop on Coreference and its Applications*, CorefApp-99, pages 85–89, College Park, Maryland.

Novák, M. and Nedoluzhko, A. (2015). Correspondences between Czech and English coreferential expressions. *Discours*, 16.

Novák, M. and Žabokrtský, Z. (2014). Cross-lingual coreference resolution of pronouns. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 14–24, Dublin, Ireland.

Novák, M., Žabokrtský, Z., and Nedoluzhko, A. (2013). Two case studies on translating pronouns in a deep syntax framework. In *Proceedings of the 6th International Joint Conference on Natural Language Processing*, pages 1037–1041, Nagoya, Japan.

Pradhan, S., Luo, X., Recasens, M., Hovy, E., Ng, V., and Strube, M. (2014). Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 30–35, Baltimore, Maryland.

Zelenko, D., Aone, C., and Tibbetts, J. (2004). Coreference resolution for information extraction. In *Proceedings of the Conference on Reference Resolution and Its Applications*.

10. Language Resource References

Grishina, Y. and Stede, M. (2015). Knowledge-lean projection of coreference chains across languages. In *Proceedings of the 8th Workshop on Building and Using Comparable Corpora*, page 14, Beijing, China.

Guillou, L., Hardmeier, C., Smith, A., Tiedemann, J., and Webber, B. (2014). ParCor 1.0: A parallel pronoun-coreference corpus to support statistical MT. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC-2014)*, Reykjavik, Iceland.

Lapshinova-Koltunski, E. and Kunz, K. (2014). Annotating cohesion for multillingual analysis. In *Proceedings of the 10th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, pages 57–64, Reykjavik, Iceland.

Poesio, M. and Artstein, R. (2008). Anaphoric annotation in the ARRAU corpus. In Bente Maegaard Joseph Mariani Jan Odijk Stelios Piperidis Daniel Tapias Nicoletta Calzolari (Conference Chair), Khalid Choukri, editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.

An Application for Building a Polish Telephone Speech Corpus

Bartosz Ziółko[1,2], Piotr Żelasko[1], Ireneusz Gawlik[1,3], Tomasz Pędzimąż[1,2], Tomasz Jadczyk[2]
[1]AGH University of Science and Technology, Poland,
Department of Computer Science, Electronics and Telecommunications,
al. Mickiewicza 30, Kraków, Poland, www.dsp.agh.edu.pl
[2]Techmo, Kraków, Poland, techmo.pl
[3]Grupa Allegro Sp. z o.o., Poland
bartosz.ziolko@techmo.pl, pzelasko@agh.edu.pl, igawlik@live.com, {pedzimaz, jadczyk}@techmo.pl

Abstract

The paper presents our approach towards building a tool for speech corpus collection of a specific domain content. We describe our iterative approach to the development of this tool, with focus on the most problematic issues at each working stage. Our latest version synchronizes VoIP call management and recording with a web application providing content. The tool was already used and applied for Polish to gather 63 hours of automatically annotated recordings across several domains. Amongst them, we obtained a continuous speech corpus designed with an emphasis on optimal phonetic diversification in relation to the phonetically balanced National Corpus of Polish. We evaluate the usefulness of this data against the GlobalPhone corpus in the task of training an acoustic model for a telephone speech ASR system and show that the model trained on our balanced corpus achieves significantly lower WER in two grammar-based speech recognition tasks - street names and public transport routes numbers.

Keywords: speech recognition, language resources, corpus recording tools

1. Introduction

Speech corpora acquisition is a significant problem in voice applications development, however, its magnitude is often under-appreciated. In this paper, we address this problem by exploring different means of how speech recordings can be collected. The process we describe is iterative - as we will show, at each iteration the previous approach is scrutinised and improved upon, which allows for acquisition of the recordings on a larger scale than possible before, while improving user experience, and thus the quality of the data collected.

Specifically, we describe a tool designed to improve our Automatic Speech Recognition (ASR) system performance, specialised for conversations over telephone - SARMATA (Ziółko et al., 2015). To that end, we focus on improving the training corpus by collecting more and more data, as Polish is still quite limited in the availability of speech corpora regarding their sizes and diversity. Our main assumptions were:

- training data for telephone ASR should be recorded through a telecommunication channel in order to condition the ASR on the channel effects met in a production environment,
- speech content should be as diversified as possible,
- it is faster and more cost efficient to produce speech for a given transcription than to annotate existing recordings.

Our ASR is based on a Deep Neural Network (DNN) architecture (Ziółko et al., 2015). At the moment, we are able to train the DNN with approx. 4 million parameters on as little as around 100 hours of speech. However, the de facto standard in the industry is to use sizeable training corpora, which total durations are measured in hundreds or even thousands of hours (Amodei and others, December 2015; Xiong and others, January 2017). Therefore, we expect to achieve better model generalisation and word error rates (WER) with more data, and thus aim in training SARMATA on at least 1000 hours of speech. To achieve this goal without sacrificing quality of the data, we chose to record sentences from National Corpus of Polish (NCP) (Przepiórkowski et al., 2012) and parts of Polish Wikipedia, choosing phrase subsets with optimal phonetic diversification. This process is described in details in section 3.1.

2. Early Development Stage

The present discussion has focused on the collection of a large quantity of continuous speech recordings. However, our first attempts at recorded speech collection were focused on a much simpler case, i.e. isolated digits, keywords and commands recognition, as described in (Żelasko et al., 2016). During that time, we used a free VoIP softphone application to call speakers whom we wanted to record, and manually annotated the recordings. Not surprisingly, this approach did not yield a considerable amount of data, and was not satisfactory in the long term.

In preparation for the next iteration, in which we wanted to record about 1200 different street names, we established that the recordings have to be automatically annotated for our approach to be viable. To this end, we developed a simple C++ VoIP call recorder application, which could be called by a person to be recorded. At the beginning of the call, the user listened to a pre-recorded invitation prompt with instructions and asked to prepare the list of street names he wants to record and select appropriate number on the DTMF keyboard. The street names were separated into 12 disjoint sets, each sized at about a hundred utterances, and so each key on the keyboard could be assigned to a different phrase set. Upon selecting the phrase set, the user had to read the entire list in correct order, and wait for a beep between each street name. The recording

was then automatically cut at the moments where beep was played and annotated according to the ordering of the items in a given set.
We need to bear in mind, that the timing was crucial in this process, as long waiting times between beeps can quickly become exhausting for the speaker, yet on the other hand, giving too short an interval for reading led to prematurely and incorrectly cut recordings. We observed that setting the time interval between beeps as constant was found unsatisfactory, because some street names could be very long, e.g. "Osiemnastego Bielskiego Batalionu Desantowo Szturmowego" (eng. 18th Bielski assault and landing battalion), while others are very short, e.g. "Agawy" (eng. "Agave street"), which in turn led user waiting for a very long time between each beep. We solved the problem by developing a heuristic for predicting how much time T should pass between beeps, given by a linear function of the number n of non-whitespace characters in the utterance. In practice, we found that using an offset of 1.5 seconds and adding approx. 300 milliseconds per syllable allowed most of our speakers to read at an optimal pace. We approximated the number of syllables as half the number of characters in an utterance. The relation is given by

$$T = 1500 \text{ ms} + n * 150 \text{ ms} . \quad (1)$$

Using this application, we gathered a total of several hours of streets names recordings. However, it was still far from perfect and had several problems. Firstly, we still needed to manually verify that the data was properly cut and annotated, and fix the mistakes. Secondly, the process was confusing for some speakers, causing them to wait for a few beeps, which introduced offsets in some of the annotations, which were later corrected. Also, due to the simplicity of the application, only a single speaker could be recorded at a time. Seeing as the process of data acquisition was still largely manual (including the selection of phrase set and addition of new recordings to the corpus), we designed and implemented a more robust solution.

3. The Application

The first major version of our tool consists of a front-end web application and a back-end server application. The front-end part is capable of displaying a previously prepared sentence set to the user, and establishing a phone call between the server and the users phone, enabling the user to dictate the phrases over the phone.
Each user, upon entering the website, is led through the following steps:

- Webform, allowing to enter users nickname, telephone number and gender. The user also chooses the phrase set to be recorded (see Fig. 1). The user may also access this screen through a specially crafted URL with pre-filled data, so that he does not have to fill any field.
- Initial invitation screen, with instructions related to the recording process. (see Fig. 2).
- Terms of usage screen, with necessary agreement statement. That screen includes a "Call me" button (see Fig. 3).
- Screen, which informs about ongoing call. User is presented with the phone number used to call them. (see Fig. 4).
- After the call is answered, the screen automatically changes to display the phrases to be read (see Fig. 5). To advance to the next phrase, the user can click on the "Next" button or press space bar. The recorded utterance is stored on the machine, where the back-end server application is being run.

Many speakers misspoke sentence or react emotionally to read texts. This generated additional work on the verification of the quality of recordings and as a result reduced the amount of received recordings. Most speakers were aware of committed mistakes. In order to solve this issue, we gave the speakers an option to repeat the phrase or to go back to the previous one. Both actions result in overriding previously recorded phrase on the server.
In order not to burden our users with the payment for the recording session, we decided to take advantage of VoIP service providers, allowing us to call the speaker directly. Due to that, the costs of the calls were covered from the corpus maker side and much lower due to a large scale payment plan.
Back-end part of the application was implemented in C++. In order to be able to connect with most of the Polish VoIP service providers, we chose Session Initiation Protocol (SIP) as our call handling protocol, and Real-Time Transport Protocol (RTP) for transfer of audio packets. The audio packets encoding varies for each call session, as it is negotiated between call endpoints, and in our case is one of: G.722 HD, G.711u, G.711a, G.729, G.726-32, G.722, iLBC or GSM - however, it is possible that audio packets are transcoded at a node between call endpoints, which results in a diverse range of channel modifications in the obtained recordings. Multiple recording sessions are allowed, by means of thread-per-session approach. Front-end content is served statically, and communication between back-end and front-end parts relies on XHR calls. REST interface, needed for communication with application front-end part, is implemented with the use of Mongoose, open source, embedded C++ HTTP server library (Hammel, 2010).
Front-end part is implemented as a Single Page Application (SPA), using AngularJS 1.4 library (Jain et al., 2014). We took into consideration unexpected events, such as call failures. Persistent state exchange from back-end to front-end is implemented using AJAX polling technique. Use of such technique allows us to detect closing of browser window, and in effect ending the call with server side event. To avoid the abuse of the service, a lack of interaction with front-end part also results in ending the call.

3.1. Speech diversification

It has been shown that phonetically balanced training corpora tend to improve the efficacy of the resulting ASR system (Uraga and Gamboa, 2004) (Wang, 1998). In order to acquire this property, we extracted optimal phonetically diversified phrase sets from the NCP and Polish pages of



Figure 1: The application starts with collecting information about the speaker (from the top): the nickname, phone number, gender and the choice of a phrase set. All the information can be also inserted by a link to speed up the process.

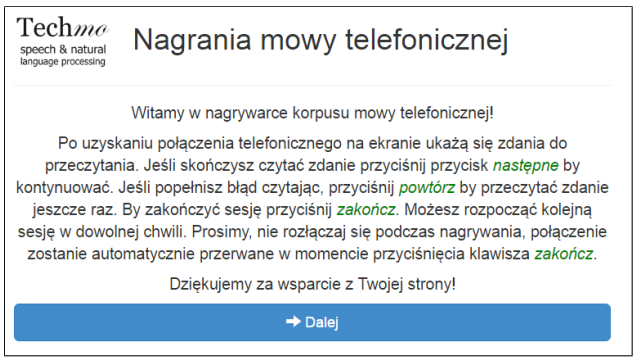


Figure 2: After user fills the questionnaire, a short explanation of how the application works and what is expected from the speaker is displayed.

Wikipedia. As such, a relatively small subcorpus of selected phrases had retained phonetic distribution which is close to the distribution of the whole corpus. We have decided to implement two approaches for records gathering.
In the first one, we created a huge corpus of randomly selected phrases, and let multiple users iterate over following sentences for as long as they were willing to. When multiple speakers were recording, each one has received a sentence that was enqueued in a circular phrase-list. That approach, however, did not guarantee that the built corpora would be balanced both in terms of voice diversity as well as phonetically balanced for each speaker.
In the second approach, we prepared phonetically balanced phrase subsets of fixed size by selection of specific sentences. However, simple approach based on random subsets selection of feasible size (less than 10 minutes of continuous speech) did not retain expected phonetic properties.



Figure 3: The user is informed that the recordings may be further processed, demonstrated and sold as part of a corpus, without revealing information about the users identity. In order to proceed, the user has to agree.



Figure 4: Information that the user is being called by the server (after he agrees to the terms of service).

To overcome this issue, we developed a subset selection heuristic algorithm (see Algorithm 1), allowing us to prepare diverse, phonetically balanced phrase subsets for each speaker.
In the proposed approach, we transformed each NCP sentence into a bag of phonemes (BOP) representation, so that each sentence was represented by a fixed length phoneme count vector. Generated phrase subset was initialized with a single, randomly selected sentence. Then, we iteratively expanded the subset with new sentences. The choice of the new sentence is based on phonetic similarity, so that the BOP of each selected sentence closely matches the difference in phoneme distribution between NCP and current state of the subset, by means of cosine measure. Additional constraint on sentence length was applied, reducing bias towards short sentences.

3.2. Issues encountered during corpus collection

We encountered some issues, which resulted in some parts of our acquired data to be unnatural. The first one was a result of wrong gender inflection in sentences displayed to speakers, i.e. males were sometimes given female sentences and the other way around. The second source of unnatural recordings was unusually large number of swear



Figure 5: The main working screen consists of a display of the phrase to be read (here 'Stare Miasto'), information about the next phrase (here 'Grzegorzki') and steering buttons - from the left (previous, repeat and next) and the large red one - finish.

Algorithm 1 NCP subset selection procedure

```
function SELECTNCPSUBSET(corpus, subsetSize)
    BOPSentences ← TOBOP(corpus)
    Subset ← EMPTYSET()
    Initial ← RANDOM(BOPSentences)
    INSERT(Subset, Initial)
    while SIZE(Subset) < subsetSize do
        Expected ← PHONETICDISTDIFF(Subset, BOPSentences)
        Selected ← LOOKUP(BOPSentences, Expected)
        INSERT(Subset, Selected)
        REMOVE(BOPSentences, Selected)
    end while
end function
```

words in NCP. Two speakers actually stopped participation in the recordings, because they refused to read such phrases. Some other speakers encountered problems while reading unusual words, especially archaic ones.

A possible field for improvement is automatic progress to a next phrase. In our application it was arranged by a click of a button while it could be arranged automatically by silence detection and (or) on-line recognition. Also, we did not provide automatic check of the content by application of ASR.

The application was created for collection of a relatively small corpus. Due to that it did not provide an efficient user and content control tool. For our goals it was not necessary, but it could prove problematic in a large, speaker-diversified corpus acquisition scenario.

We did not control the environment of the speakers we were recording. Even though we asked them to record in silent conditions some of the recording where done with background noise.

We also considered recording with a separate WebRTC channel through a PC microphone, in parallel to telephone call. This would have provided us with the same speech recording, transferred via different media. At the time, we considered it to be too complex and in the first version we decided not to do it.

We plan to address those issues in the next iteration of our tool.

4. Results

4.1. Collected resources

We managed to gather several corpora using various versions of the tool. Most notably, we collected a general-purpose corpus of continuous speech from the NCP and Wikipedia phrase sets, which consists of 15600 utterances, uttered by a total of 163 speakers (63 female and 100 male). This is more than 25 hours of continuous speech.

Other collected corpora are domain-specific. The Commands corpora consists of utterances which often appear in Interactive Voice Response (IVR) scenarios, such as basic commands (e.g. "*go back*"), numbers and dates, as well as more specific vocabulary, such as names of medical specializations. The *Bus* corpus is a collection of recordings of the bus line numbers in Cracow. The only large collection we gathered before application 1.0 is the *Streets* corpus, as mentioned in section 2.. Details about those corpora are presented in table 1.

Corpus	Speakers	Utterances	Duration	Size[MB]
Continuous	163	15685	25:05	2860
Commands	72	10846	09:05	1166
Bus	130	17991	17:21	1969
Streets	47	10507	12:09	1380
Total	434	55029	63:40	7375

Table 1: Detailed information about sets of collected recordings. The duration is given in [hh:mm] format.

	GlobalPhone	Continuous	Both
Streets	50.41%	5.21%	3.59%
Bus	51.97%	11.47%	8.29%

Table 2: Word error rates (WER) of the SARMATA ASR system trained exclusively on: I. GlobalPhone; II. Continuous corpus; III. Sum of GlobalPhone and Continuous. The test corpora are the Streets and Bus corpora.

4.2. Speech recognition

In order to ascertain the quality of gathered recordings, we incorporated them into training of our SARMATA ASR (Ziółko et al., 2015). The baseline system was trained using the Polish training subset of the GlobalPhone corpus (Schultz, 2002). Second variant of the system was trained on the *Continuous* corpus, and finally the third variant was trained on both of these corpora. For evaluation, we used previously mentioned *Bus* and *Streets* corpora. The results are shown in table 2.

Training our system on the *Continuous* corpus resulted in a major improvement of the word error rate (WER) during evaluation. We attribute this result to several factors. Firstly, our corpus consists only of the telephone speech, and GlobalPhone, contrary to its name, does not. This results in a mismatch between the spectral characteristics of the training and evaluation recordings in case of a system trained on the GlobalPhone. Another reason is the larger amount of data available in the *Continuous* corpus - 25 hours versus GlobalPhone's 15 hours. Finally, our recordings are of higher quality and generally not as noisy as those encountered in GlobalPhone.

Combining both GlobalPhone and *Continuous* corpora introduced additional diversity in the recordings, further improving the ASR acoustic model generalisation and reducing WER.

5. Conclusions

We presented an iterative process of designing a tool for automated corpus collection. Throughout its development, we observed and addressed subsequent issues resulting from automating the annotation process. Current version of the tool provides a user-friendly approach towards recording continuous speech. As a result of our work, we were able to obtain 63 hours of quality recordings, which we use both for training of the acoustic models and for ASR evaluation. We observed a significant WER improvement with regard to the baseline system evaluated on telephonic speech.

6. Acknowledgments

This project has received funding from the European Union's Horizon2020 research and innovation programme under grant agreement No 761349.

7. Bibliographical References

Amodei, D. et al. (December 2015). Deep speech 2: End-to-end speech recognition in english and mandarin. *arXiv*.

Hammel, M. (2010). Mongoose: an embeddable web server in c. *Linux Journal*, 2010.

Jain, N., Mangal, P., and Mehta, D. (2014). Angularjs: A modern mvc framework in javascript. *Journal of Global Research in Computer Science*, 5(12).

Wang, H.-m. (1998). Statistical analysis of mandarin acoustic units and automatic extraction of phonetically rich sentences based upon a very large chinese text corpus. *Computational Linguistics*, 3(2):93–114.

Xiong, W. et al. (January 2017). The microsoft 2016 conversational speech recognition system. *arXiv*.

Ziółko, B., Jadczyk, T., Skurzok, D., Żelasko, P., Gałka, J., Pędzimąż, T., Gawlik, I., and Pałka, S. (2015). SARMATA 2.0 automatic Polish language speech recognition system. *Interspeech, Dresden*.

8. Language Resource References

Przepiórkowski, A., Bańko, M., Górski, R., and Lewandowska-Tomaszczyk, B. (2012). *Narodowy Korpus Jezyka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Schultz, T. (2002). Globalphone: A multilingual speech and text database developed at Karlsruhe University. In *Proceedings of the ICSLP*, pages 345–348.

Uraga, E. and Gamboa, C. (2004). Voxmex speech database: design of a phonetically balanced corpus. In *Fourth International Conference on Language Resources and Evaluation*.

Żelasko, P., Ziółko, B., Jadczyk, T., and Skurzok, D. (2016). AGH corpus of Polish speech. *Language Resources and Evaluation*, 50:585–601.

CPJD Corpus: Crowdsourced Parallel Speech Corpus of Japanese Dialects

Shinnosuke Takamichi, Hiroshi Saruwatari
Graduate School of Information Science and Technology, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
{shinnosuke_takamichi, hiroshi_saruwatari}@ipc.i.u-tokyo.ac.jp

Abstract

Public parallel corpora of dialects can accelerate related studies such as spoken language processing. Various corpora have been collected using a well-equipped recording environment, such as voice recording in an anechoic room. However, due to geographical and expense issues, it is impossible to use such a perfect recording environment for collecting all existing dialects. To address this problem, we used web-based recording and crowdsourcing platforms to construct a crowdsourced parallel speech corpus of Japanese dialects (CPJD corpus) including parallel text and speech data of 21 Japanese dialects. We recruited native dialect speakers on the crowdsourcing platform, and the hired speakers recorded their dialect speech using their personal computer or smartphone in their homes. This paper shows the results of the data collection and analyzes the audio data in terms of the signal-to-noise ratio and mispronunciations.

Keywords: crowdsourcing, Japanese dialect, speech, transcription

1. Introduction

A public corpus of low-resourced languages can accelerate related studies, such as natural language, spoken language, and speech signal processing. Thanks to recent improvements in machine learning techniques for these research areas (v. d. Oord et al., 2016; Hinton et al., 2012; Takamichi et al., 2017), the accuracies in the rich-resourced languages (such as English, Chinese, and Japanese) have become high, and attention is shifting to more challenging tasks, including language and speech processing of dialects. Dialect processing is one of the topics being actively targeted in machine translation (Salloum et al., 2014), speech recognition (Hirayama et al., 2014), speech perception (Jacewicz and Fox, 2017), and speech synthesis (Masmoudi et al., 2016). For studies on Japanese dialect, (Yoshino et al., 2016) collected a parallel database of Japanese dialects with voice recording in their well-equipped recording environment. However, geographical and expense issues make it impossible to use such a perfect recording environment for all existing Japanese dialects.

In this work, we collected parallel data of the common language and several dialects in Japanese using a web-based recording and crowdsourcing platform. We recruited native dialect speakers on the crowdsourcing platform, and the hired speakers of 21 dialects uttered 250 sentences for each. They also converted text in the common language into their dialect, and read the dialect text in the web-based recording platform available on personal computers and smartphones. Thus, the resultant resources consist of texts (one common language and 21 dialects), nine hours of speech data, and their geographic contexts (i.e., areas of the dialect). We analyzed the collected speech data from perspectives of the speech signal and spoken language processing. This paper shows the signal-to-noise ratio (SNR) results for the recording environments and the number of mispronunciations.

2. Existing resources of Japanese dialects and their problems

Several text/speech corpora of Japanese dialects have been compiled (Kubozomo, 2001 2008; National Institute for Japanese Language and Linguistics, 2016). For example, the database at National Institute for Japanese Language and Linguistics (National Institute for Japanese Language and Linguistics, 2016) includes text and speech data spoken by native dialect speakers. However, the recording setting is not suitable for spoken language processing. For instance, it is hard for current acoustic modeling techniques to model such spontaneous speech recorded in such a poor environment. Yoshino et al. (Yoshino et al., 2016) collected parallel data of Japanese dialects. They constructed parallel text between the common language and dialects in Japanese and had native dialect speakers read the dialect texts in their recording studio. Because the reading-style speech data was recorded in the well-equipped recording environment, their corpus is useful for spoken language processing research. In addition, because their corpus includes the parallel text of the common language and dialects, it is also useful for natural language processing research. However, dialects that can be collected in such a perfect environment are very limited and collecting many dialects is unrealistic because of geographical and expense issues. For example, travel expenses from the speaker's hometown (e.g., in the countryside where a rare dialect is spoken) to the recording studio have to be covered, and the cost significantly increases when the variety of dialects to record is large. Also, elderly people of countrysides is hard to come to the recording studio due to their physical burden.

Crowdsourcing can greatly reduce the time and money needed for building speech corpora (Hughes et al., 2010; Gutkin et al., 2016). The crowdsourced data is noisy compared to the data obtained in the perfect environment, but it is expected to solve above issues. In this work, using a web-based recording platform, we collected the speech data recorded in an indoor recording environment with comparably stationary audio noise. This is because the effects of such environments can be alleviated more easily than (National Institute for Japanese Language and Linguistics, 2016). Moreover, we collected small amounts of speech data of a variety of dialects rather than large amounts of speech data of one dialect. This is because there is a

strong relationship between the it is known that phonetic and prosodic properties of dialects are strongly related to the geographical relationship (Preston, 1999), and mixed dialect models that take into account this relationship will improve the accuracy of dialect modeling.

3. Collection of the CPJD corpus

This section explains how we collected the CPJD corpus. First, we prepared sentences of the common language in Japanese so that the native dialect speakers could convert it to their dialects. Then, the speakers were recruited on the crowdsourcing platform, and the hired speakers converted the sentences and read the dialect sentences. Finally, we constructed the corpus comprising dialect sentences, their read speech, the name of the dialect, and the geographic contexts of the dialect.

3.1. Sentences of the common language

We selected sentences of the common language so that the native dialect speakers could convert then into their dialects. These sentences were randomly selected from a blog category of the BCCWJ (Balanced Corpus of Contemporary Written Japanese) corpus (Maekawa, 2014) and the KNB (Kyoto University and NTT Blog) corpus (Hashimoto et al., 2011). We expect the blog sentences are comparably informal and suitable for speaking informal dialects. Sentences including geographic words, e.g., a place name, were removed. In addition, words that are unsuitable for current daily life were changed into suitable ones, e.g., "cell phone" was changed to "smartphone."

3.2. Web-based recording platform

We prepared a web-based speech recording platform available on personal computers and smartphones. Using *Recorder.js*[1], we prepared buttons to start or stop an audio recording. The platform also has a spectrum and waveform plots so that speakers can check their voice. The spectrum is produced in real time during a recording, and the waveform is produced after the recording. Automatic mispronunciation detection, voice activity detection, noise detection, and voice gain control function were not implemented.

3.3. Recruiting native dialect speakers

We hired native dialect speakers to collect text and audio data. Before hiring them, we asked recruited speakers to tell us the name of their dialect and its prefecture on the crowdsourcing platform, and we hire some of the speakers so that many areas of the country would be covered. The hired speakers first converted sentences of the common language into their dialect and then read the converted sentences on the web-based recording platform. The conversion and recording were done in their indoor recording environments. The speakers were instructed not to include extraneous sounds like coughing in their recordings, and we let the speakers convert honorific expressions in the common language text into plain expressions to make it easier for them to convey their dialect.

[1] `https://github.com/mattdiamond/Recorderjs`

Table 1: List of collected dialects. The numbers correspond to the prefecture index in Fig. 1. Each dialect was spoken by just one native speaker (except Nara-ben). (The suffix "ben" means dialect. For example, "Hokkaido-ben" is a dialect of Hokkaido.)

Area	Dialect
Hokkaido	Hokkaido-ben (1)
Tohoku	Tsugaru-ben (2), Akita-ben (3), Iwaki-ben (4)
Kanto	Saitama-ben (5)
Chubu	Kanazawa-ben (6), Tohshuu-ben (7), Hukui-ben (8)
Kinki	Kyoto-ben (9), Kyo-kotoba (9), Nara-ben (10), Osaka-ben (11)
Chugoku	Okayama-ben (12), Izumo-ben (13), Hiroshima-ben (14)
Shikoku	Tosa-ben (15), Awa-ben (16), Iyo-ben (17)
Kyushu	Fukushima-ben (18), Miyazaki-ben (19), Morokata-ben (19, 20)
Okinawa	None

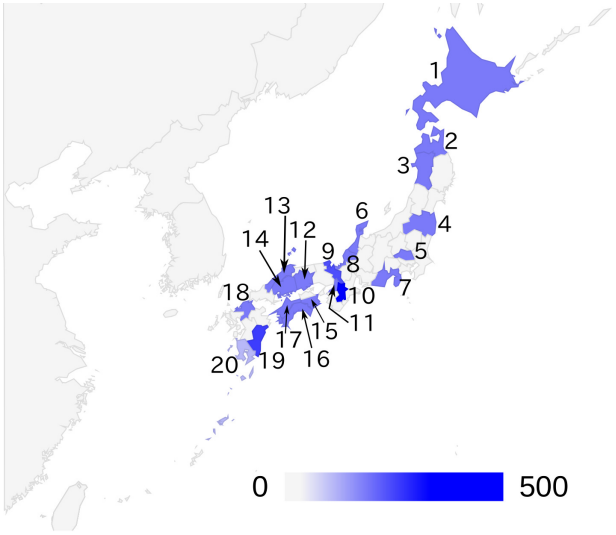


Figure 1: The number of collected utterances for each prefecture in Japan. The map chart was drawn using Google chart API (`https://developers.google.com/chart/`), but we moved the position of the legend and removed some areas outside of Japan for a clear illustration. The prefecture boundary is not always the same as the boundary of the dialect's area.

3.4. Data correction

After data collection, we manually identified misrecorded or mispronounced (e.g., hesitation or filler) data and manually added commas between breath groups.

Table 2: Examples of parallel text data. For comparison, the text translated into English is also listed. As described in Section 3.4, we corrected the text data after data collection, but the texts shown here was not corrected.

Dialect	Texts	Phonemes
Common	できるだけスマートフォンひとつで身の回りのこと全て片付けようとしているようだ.	d e k i r u d a k e s u m a: t o f o N h i t o ts u d e m i n o m a w a r i n o k o t o s u b e t e k a t a z u k e y o u t o sh i t e i r u y o u d a
Miyazaki-ben (19)	なるだけスマートフォンひとつで身んまわりんこと全部片付けようとしちょるみたいやね.	n a r u d a k e s u m a: t o f o N h i t o ts u d e m i N m a w a r i N k o t o z e N b u k a t a z u k e y o u t o sh i ch o r u m i t a i y a n e
Tsugaru-ben (2)	でぎるだげスマートフォンばりで身の回りのことまるっととっけるんた.	d e g i r u d a g e s u m a: t o f o N b a r i d e m i n o m a w a r i n o k o t o m a r u q t o t o q k e r u N t a
Cf. Translation	It seems like everything around here is done as much as possible on a smartphone.	-
Common	これからこの機能が加わったからといって特別ハッピーなわけでもない.	k o r e k a r a k o n o k i n o u g a k u w a w a q t a k a r a t o i q t e t o k u b e ts u h a q p i: n a w a k e d e m o n a i
Kyo-kotoba (9)	これからこの機能が加わったからゆうて特別ハッピーなわけでもあらへん.	k o r e k a r a k o n o k i n o u g a k u w a w a q t a k a r a y u u t e t o k u b e ts u h a q p i: n a w a k e d e m o a r a h e N
Awa-ben (16)	これからほの機能が加わったからといって特別ハッピーなわけやないし.	k o r e k a r a h o n o k i n o u g a k u w a w a q t a k a r a t o i q t e t o k u b e ts u h a q p i: n a w a k e y a n a i sh i
Cf. Translation	I would not be especially happy even if this function were added in the future.	-

4. Analysis of the CPJD corpus

4.1. Corpus specification and examples

To recruit the native dialect speakers, we used the crowdsourcing service Lancers[2], which is one of the biggest crowdsourcing platforms in Japan. To cover a large number of areas, we asked speakers to name their dialect and home prefecture, and hired speakers who can speak dialects we had not yet collected. The recruiting period ran for five days during April and May 2017. Hired speakers were paid approximately $45 for their work. The number of common language sentences to be converted and read was 250. The audio sampling rate was 44.1 or 48 kHz. The number of hired speakers was 22, nine male and 13 female speakers. Gender labels were added by our annotator after recording. The number of prefectures where dialects were collected was 20. The average duration of recorded speech for each speaker was 24 minutes and 36 seconds (including non-speech regions). The total duration was approximately nine hours. Compared to (Yoshino et al., 2016) (3.5 hours of speech data, eight prefectures), the CPJD corpus contains large amounts of speech data and covers more prefectures than previous corpora do.

Table 1 lists the collected dialects, and Fig. 1 maps their areas. We can see that a variety of areas and dialects were collected in this work. Table 2 shows examples of collected sentences and transcribed phonemes. The meanings of sentences are the same; however, there are small differences in minor word choices.

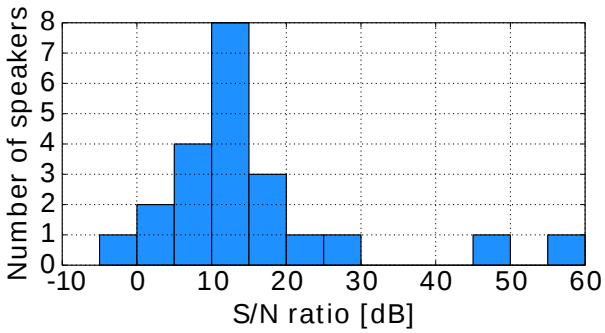


Figure 2: Histogram of SNRs. The values are averaged for each speaker.

Table 3: Statistics of SNR [dB]. The histogram is shown in Fig. 2.

Worst	Best	Mean	Median
-2.1	57.8	15.1	12.6

4.2. Analysis

We analyzed the collected audio data. First, for use in speech signal processing research, we calculated SNR (signal-to-noise ratio) of the speech data using the decision-directed method (Plapous et al., 2006). The high SNR means that speech was recorded in the clean (well-equipped) environment. The histogram of SNRs for each speaker is shown in Fig. 2, and the SNR statistics are summarized in Table 3. We can see that two speakers had a

[2] http://www.lancers.jp

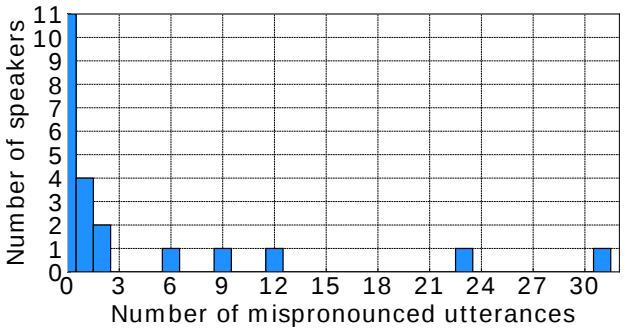


Figure 3: Histogram of mispronounced utterances. The values are averaged for each speaker. Automatic mispronounciation detection was not used for the recordings. Mispronounce were manually labeled after the recordings.

good recording environment (45-through-60 dB), but most of the speakers environments were less than ideal.

Next, for use in spoken language processing research, we calculated the number of mispronounced utterances. We manually detected mispronunciations where the text and speech were mismatched. Fig. 3 shows the histogram of the number of mispronounced utterances for each speaker. Half of the speakers made no mispronunciations, but a few speakers made more than 20 mistakes within their 250 utterances. This suggests the need to improve the recording platform.

5. Conclusion

We constructed a corpus of parallel data of Japanese dialects, called the CPJD corpus, using web-based recording and crowdsourcing platforms. The corpus contains sentences of one common language and 21 dialects in Japanese, nine hours of speech data read by dialect speakers, and their geographic contexts. We analyzed the collected speech data in terms of signal-to-noise ratio and the number of mispronounced utterances. In future work, we will build speech synthesis systems using the CPJD corpus and do the preparation to make the corpus available to the public.

Acknowledgment

Part of this work was supported by the SECOM Science and Technology Foundation.

6. Bibliographical References

Gutkin, A., Ha, L., Jansche, M., Pipatsrisawat, K., and Sproat, R. (2016). TTS for low resource languages: A Bangla synthesizer. In *Proc. LREC*, pages 2005–2010, Paris, France.

Hinton, G., Deng, L., Yu, D., Dahl, G., r. Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine of IEEE*, 29(6):82–97.

Hirayama, N., Yoshino, K., Itoyama, K., Mori, S., and Okuno, H. G. (2014). Automatic speech recognition for mixed dialect utterances by mixing dialect language models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32(2):373–382.

Hughes, T., Nakajima, K., Ha, L., Vasu, A., Moreno, P., and LeBeau, M. (2010). Building transcribed speech corpora quickly and cheaply for many languages. In *Proc. INTERSPEECH*, pages 1914–1917, Chiba, Japan, Sep.

Jacewicz, E. and Fox, R. A. (2017). Dialect perception by older children. In *Proc. INTERSPEECH*, pages 354–358, Stockholm, Sweden, Aug.

Masmoudi, A., Ellouze, M., Bougares, F., Esètve, Y., and Belguith, L. (2016). Conditional random fields for the tunisian dialect grapheme-to-phoneme conversion. In *Proc. INTERSPEECH*, pages 1457–1461, San Francisco, U.S.A., Sep.

Plapous, C., Marro, C., and Scalart, P. (2006). Improved signal-to-noise ratio estimation for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6):2098–2108.

Preston, D. R. (1999). *Handbook of perceptual dialectology*, volume 1. John Benjamins Publishing.

Salloum, W., Elfardy, H., Alamir-Salloum, L., Habash, N., and Diab, M. (2014). Sentence level dialect identification for machine translation system selection. In *Proc. ACL*, volume 2 (Short Papers), pages 772–778, Baltimore, Maryland.

Takamichi, S., Tomoki, K., and Saruwatari, H. (2017). Sampling-based speech parameter generation using moment-matching network. In *Proc. INTERSPEECH*, pages 3961–3965, Stockholm, Sweden, Aug.

v. d. Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W., and Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. abs/1609.03499.

Yoshino, K., Hirayama, N., Mori, S., Takahashi, F., Itoyama, K., and Okuno, H. G. (2016). Parallel speech corpora of japanese dialects. In *Proc. LREC*, pages pp. 4652–4657, Portoroz, Slovenia, May.

7. Language Resource References

C. Hashimoto and S. Kurohashi and D. Kawahara and K. Shinzato and M. Nagata. (2011). *Construction of a Blog Corpus with Syntactic, Anaphoric, and Sentiment Annotations* `http://nlp.ist.i.kyoto-u.ac.jp/kuntt/#ga739fe2`. Journal of Natural Language Processing (in Japanese).

H. Kubozomo. (2001-2008). *Accent Database of Koshikijima Japanese,* `http://koshikijima.ninjal.ac.jp/`.

K. Maekawa. (2014). *Balanced corpus of contemporary written Japanese* `http://pj.ninjal.ac.jp/corpuscenter/bccwj/`. Proceedings of the 6th Workshop on Asian Language Resources.

National Institute for Japanese Language and Linguistics. (2016). *Database of Spoken Dialects all over Japan: Collection of Japanese Dialects* `http://pj.ninjal.ac.jp/publication/catalogue/hogendanwa_db/outline.htm` *(in Japanese).*

Korean L2 Vocabulary Prediction: Can a Large Annotated Corpus be Used to Train Better Models for Predicting Unknown Words?

Kevin P. Yancey, Yves Lepage
Waseda University
2-7 Hibikino, Wakamatu-ku, Kitakyushu city, Fukuoka, Japan
kpyancey@fuji.waseda.jp, yves.lepage@waseda.jp

Abstract

Vocabulary knowledge prediction is an important task in lexical text simplification for foreign language learners (L2 learners). However, previously studied methods that use hand-crafted rules based on one or two word features have had limited success. A recent study hypothesized that a supervised learning classifier trained on a large annotated corpus of words unknown by L2 learners may yield better results. Our study crowdsourced the production of such a corpus for Korean, now consisting of 2,385 annotated passages contributed by 357 distinct L2 learners. Our preliminary evaluation of models trained on this corpus show favorable results, thus confirming the hypothesis. In this paper, we describe our methodology for building this resource in detail and analyze its results so that it can be duplicated for other languages. We also present our preliminary evaluation of models trained on this annotated corpus, the best of which recalls 80 % of unknown words with 71 % precision. We make our annotation data available.

Keywords: lexical simplification, vocabulary knowledge prediction, crowdsourcing

1. Introduction

Our goal is to build a text simplification system for L2 learners of Korean to aid reading comprehension and expedite language acquisition through reading. Reading, and extensive reading in particular (i.e., the practice of reading large amounts of easy, entertaining text), has been shown to have many benefits to language acquisition, including improving grammar (Krashen, 2003) , writing (Mason and Krashen, 1997), and listening skills (Elley, 1991), as well as reading proficiency and motivation (Crawford Camiciottoli, 2001).

A task critical to simplifying texts for L2 learners is estimating the proficiency level required to understand a word (i.e., the word's "complexity"), so that unknown words can be predicted and simpler replacements can be selected. Many past proposals focused on rules that used word frequency, length, or some combination thereof as a proxy for word complexity (Devlin and Tait, 1998; Bott et al., 2012; Shardlow, 2013), but these features do not correlate perfectly with word complexity for L2 learners. More recently, Tack et al. (2016) proposed a model that measured a word's complexity as its first level of occurrence within a corpus of graded textbooks for L2 learners, evaluating it against a corpus annotated by four L2 learners of French. However, their model was only able to correctly classify 40 % of the unknown words.

Tack et al.'s paper hypothesized that better results might be obtained by compiling a much larger annotated corpus and casting the problem as a supervised learning problem. In this paper, we will explore this alternative. We will describe our methodology for constructing a large annotated corpus by crowdsourcing via the Internet, and will evaluate the predictive capability of several models built from this corpus via supervised learning, comparing our results to previous approaches. Finally, we will describe the contents of this annotated corpus and make it available to other researchers to build their own models and compare with our results.

2. Building an L2 Annotated Corpus of Unknown Korean Words

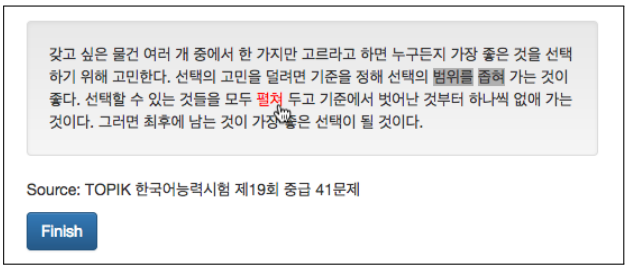
To collect a training corpus, we crowdsourced the annotation of Korean text via an Internet survey, where L2 learners annotated the words they did not understand.

2.1. Corpus Selection

We used graded synthetic texts extracted from the reading sections of past Test of Proficiency in Korean (TOPIK) exams (National Institute for International Education Development in Korea, 2017), which is the standard Korean language proficiency test administered by the South Korean government. This test rates L2 learners' Korean proficiency on a scale of 6 levels, which correspond roughly to the 6 levels in the Common European Framework of Reference (CEFR) (Won, 2016). Prior to July 2014, TOPIK exams came in three difficulty levels, or "grades": beginner (for levels 1-2), intermediate (for levels 3-4), and advanced (for levels 5-6). We automatically downloaded, parsed, and extracted reading passages from 25 different TOPIK exams for each of these three grades. Among these extracted texts, we excluded passages containing blanks or special characters that are used as markers for the exam questions. We created the corpus using the remaining 263 short passages, consisting of 31 beginner passages, 143 intermediate passages, and 89 advanced passages, most having 3–6 sentences each.

2.2. Designing a Survey to Annotate Unknown Words

The survey was conducted via a website custom-built for the purpose. When annotators first started the survey, they were asked to fill out a basic questionnaire about their native language, Korean proficiency level, and reasons for studying Korean. They were then presented with a series



Words turn red as the annotator hovers the cursor over them. Unknown words that have been clicked by the annotator are highlighted with a dark gray background.

Figure 1: Survey passage annotation

0 - Did not understand the passage at all.

1 - Understood the general topic only.

2 - Mostly understood the passage.

3 - Understood all or almost all the passage.

Figure 2: Annotator self-assessed passage comprehension scale

of passages selected semi-randomly from the corpus described in the previous section. So that passages would not be too difficult for annotators, the selection algorithm estimated the annotator's level and provided passages that were roughly one or two TOPIK levels higher so that the annotator would be expected to know roughly 80–90 % of the words in each passage. Annotators were asked to read each passage, without a dictionary, and highlight the words they did not understand by clicking on them (see Figure 1).[1] Once the annotator finished the passage, the selected words were annotated as unknown, and all the remaining words were annotated as known. After annotating each passage, annotators were asked to rate their comprehension of the passage according to the scale defined in Figure 2.

2.3. Maximizing Annotation Submissions

Each annotator was asked to finish at least 2 passages so that we would have enough data to estimate each annotator's level in addition to learning which words they did or did not know. However, to maximize the amount of data we could gather from each participant, we added the following gamification features to encourage users to annotate additional passages:

Points Annotators were awarded points for each passage completed.

Titles Annotators earned titles for earning certain numbers of points. They were regularly presented with a status screen showing them their points earned and their progress towards earning the next title.

Rankings Annotators were ranked by points earned and could view their current ranking (and that of other contributors) on a rankings page of the website.

To make it easy for annotators of various language backgrounds to contribute, the website's UI was available in 5 languages: English, Chinese, Japanese, French, and (for those whose native language wasn't supported) Korean.
We promoted the survey on forums and social media where L2 learners of Korean were likely to find it: mainly the websites waygook.org (an online forum for foreigners living in Korea), reddit.com/r/Korean (an online forum for L2 learners of Korean), iTalki (an online language learning website), and Facebook. Additionally, having seen the survey's announcement, a person with a very large number of followers helped promote the survey via Twitter.

Language[2]	# of Ann. Passages	% of Ann. Passages	# of Annotators	% of Annotators
English (en)	1,450	60.8 %	279	78.2 %
Korean (ko)[3]	262	11.0 %	1	0.3 %
Danish (da)	258	10.8 %	1	0.3 %
Chinese (zh)	87	3.6 %	16	4.5 %
Portuguese (pt)	79	3.3 %	8	2.2 %
Spanish (es)	49	2.1 %	6	1.7 %
German (de)	39	1.6 %	11	3.1 %
Vietnamese (vi)	17	0.7 %	3	0.8 %
Other (but known)	56	2.3 %	18	5.0 %
Unknown	88	3.7 %	14	3.9 %

Table 1: Annotated passages by native language

3. Analysis of Annotation Results

In 3 months time, we collected a total of 2,385 annotated passages from 357 distinct annotators of varying levels, countries, and language backgrounds. Extracting the annotations and removing duplicates (see Section 4.1.) resulted in 48,622 distinct annotator/baseword pairs, each annotated as either known or unknown.

3.1. Demographics of Annotators

The annotated corpus includes native speakers of 17 languages (see Table 1) living in a variety of countries (see Table 2).

3.2. Factors Impacting the Volume of Annotated Submissions

In this section, we discuss some of the factors that affected the quantity of annotations received. Here we refer to those who visited the survey website as "visitors", a superset of those who contributed annotations (i.e., the "annotators").

[1] Written Korean text is spaced with 어절s (pronounced "eojeol"), which are comprised of an inflected word form followed by some number of particles. To keep the annotation process simple, we used 어절s as the unit of annotation.

[2] When not supplied by the annotator in the initial questionnaire, native language was inferred from the preferred language indicated in the annotator's HTTP headers.

[3] A native Korean speaker also submitted annotations while proofreading the essays for us, as she also encountered some words she had not known in some of the advanced passages, so we included her annotations in the dataset to help identify very complex words.

Country	# of Ann. Passages	% of Ann. Passages	# of Annotators	% of Annotators
Japan (JP)	399	16.7 %	6	1.7 %
United States (US)	383	16.1 %	88	24.6 %
Denmark (DK)	258	10.8 %	1	0.3 %
Philippines (PH)	239	10.0 %	50	14.0 %
Korea, Republic of (KR)	186	7.8 %	32	9.0 %
Australia (AU)	102	4.3 %	14	3.9 %
Indonesia (ID)	95	4.0 %	18	5.0 %
Other/Unknown	723	30.3 %	148	41.5 %

Table 2: Annotated passages by country

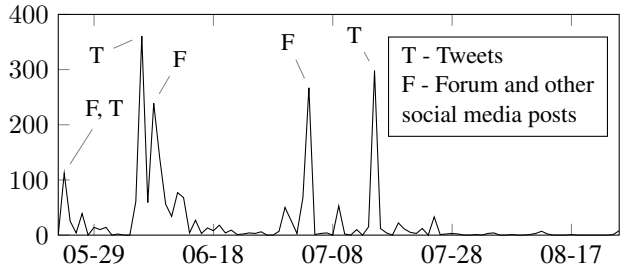


Figure 3: # of annotated passages by date

3.2.1. Promotion through Forums and Social Media

We tended to receive a large volume of annotated passages immediately following announcements on online forums and social media, followed by a much smaller volume of annotated passages in the following days or weeks. This is seen in Figure 3, where the letter T marks the dates promotional tweets were sent out and the letter F marks the dates posts were made to forums and social media websites.

We use the timing of annotated passages received and the sources of web traffic (see Table 3) to discern which websites worked best for recruiting annotators. In our case, promotion by the aforementioned Twitter user, who has over 3,500 followers with an interest in Korean popular culture, resulted in the most annotated passages. This user independently sent tweets about our survey on 3 separate occasions, resulting roughly 300 new annotated passages each time. The second largest group of annotators came from Reddit, whose contributors also showed a great diversity of language levels.

3.2.2. Gamification

Annotators gave positive feedback about the gamification features. Figure 4 also shows its positive effect on the number of annotated passages received, where we find that 29 % of annotators completed enough passages to earn the first

[4]While a sizable number of visitors came from Facebook, for the most part the timings of these visits did not correlate with survey receipt of annotated passages.

Source	# of Visitors	% of Visitors
Twitter	968	41.96 %
Facebook[4]	222	9.62 %
Reddit	155	6.72 %
Waygook.org	99	4.29 %
Other/Unknown	863	37.40 %

Table 3: Sources of annotation survey website visitors

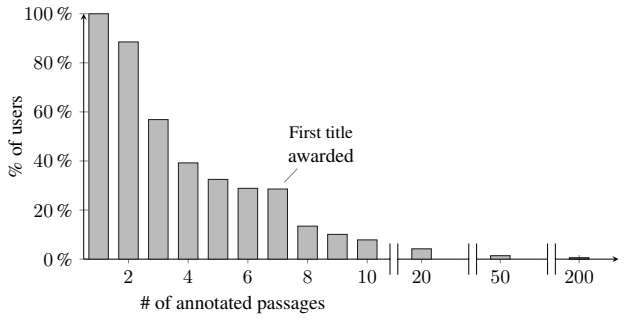


Figure 4: # of annotated passages per annotator

"title", but that the submission rate of annotated passages drops off sharply after that point.

3.2.3. Login Accounts and Privacy Concerns

Feedback from visitors and website usage statistics show that requiring annotators to register accounts negatively impacted the number of visitors who chose to annotate passages. Initially, a number of would-be annotators raised privacy concerns over using social logins, fearing they might be used to collect personally identifying information about them. After adding an option to register "anonymously" using only an arbitrary username and password that is not linked to an email or social login, website usage statistics still showed that as many as 50 % of visitors left the website once they reached the registration page.

3.3. Quality & Inter-annotator Agreement

The literature indicates that annotation tasks of this kind tend to have low inter-annotator agreement by conventional measures. Paetzold and Specia (2016) reported a Krippendorff's Alpha agreement coefficient (Hayes and Krippendorff, 2007) of 0.244, which they hypothesized was due to differences in language backgrounds and proficiency levels of the annotators. Tack et al. (2016) also noted that there can be high variation in the annotation of content words, even among annotators of the same level and language background.

The In-Corpus Model described in Section 4.1. predicts unknown words for each annotator based on how other annotators annotated those same words. So, we use this model as a tool to measure agreement between each annotator in a way that accounts for level differences and identify annotators whose annotations may be unreliable. We do this by computing the cross entropy between each annotator's annotations and the model's predictions. We then compare this to the distribution of cross entropies that result if the same words are labeled as known/unknown at random in the same proportion. This analysis showed that 74 % of annotators (which corresponds to 90 % of all annotations) submitted data that agrees with the model significantly better than randomly annotated data (p=0.01). [5]

[5]Another 8 % of annotators either did not select any words as unknown, or selected all words as unknown. A manual review of the remaining annotator's annotations suggests that some performed the annotation task backwards: selecting the words they did know instead of the words they did not know. Many others did

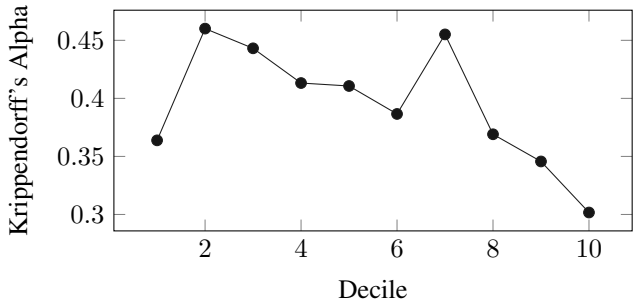


Figure 5: Inter-nnotator Agreement for Annotators Grouped into Deciles by Estimated Proficiency Level

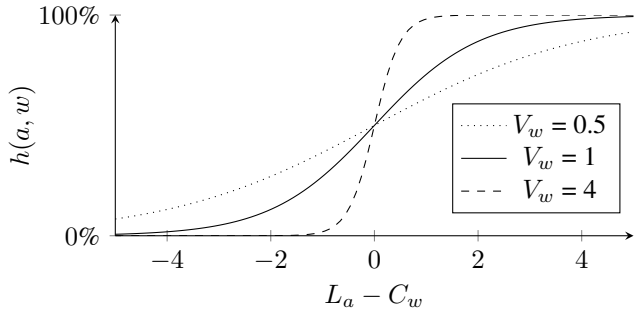


Figure 6: Probability of a word being known as predicted by h with different word complexity variances V_w

Within this 74 % of annotators, comparing annotators of similar estimated proficiency level by grouping them into deciles, we get an average Krippendorff's Alpha among the deciles of 0.394 (see Figure 5). By computing the alpha pairwise, we see the effect that the difference in proficiency level has on inter-annotator agreement: pairs of annotators with less than one percentile difference have an alpha of 0.312 on average, but those with at least a 40 percentile difference in proficiency level have negative alphas (indicating systemic disagreement) on average.

4. Unknown Word Prediction Models

In this section we demonstrate that by using a large annotated corpus as a labeled training dataset, better unknown word prediction models can be built. We evaluate three prediction models built from this annotated corpus: an In-Corpus Model that is limited to predicting words found within the training corpus, and two general models based on Support Vector Machines (SVMs). We compare the performance of these models to three baseline models. Finally, we investigate how large of an annotated corpus is needed to maximize the performance of these models.

4.1. In-Corpus Model

We start by preprocessing the annotated corpus, turning it into a labeled training dataset: each annotation's word is normalized to its baseword by removing particles, inflections, and morphological suffixes,[6] and duplicate annotations are removed so that there is at most one annotation per annotator/baseword pair (see "Labeled Dataset" in Appendix for details).

A manual review of this data, however, reveals that many of the self-reported Korean proficiency levels of the annotators are unreliable: frequently being 1–3 levels higher than what the annotator's actual level appears to be. This leaves us with no option but to estimate annotators' proficiency levels based on their annotations. But, we cannot estimate the proficiency levels of each annotator without considering the complexity of words they have annotated, which is what we're trying to learn, so this becomes a chicken-and-egg problem.

We solve this problem by learning the annotator proficiency levels and word complexities simultaneously using a method that is similar to logistic regression. We do this by first defining an equation, h, that models how the probability of an annotator knowing a word is related to the annotator's level and to the word's complexity, and then applying gradient descent to estimate the annotator proficiency levels and word complexities simultaneously.

For this model to make sense, h needs to be an S-shaped curve so that the estimated probability of a reader knowing a word increases monotonically as the difference between the annotator's level (denoted L_a) and the word's complexity (denoted C_w) increases (see Figure 6). Thus, we choose to define h as follows:

$$h(a,w) = \phi\left(V_w^{-1}(L_a - C_w)\right) \quad (1)$$

where a is the annotator, w is the word, and ϕ is the sigmoid function $\phi(x) = 1/(1+e^{-x})$. The word complexity variance, V_w, is added to control the steepness of the curve for each word.[7] With equation h defined, we define a loss function over our training dataset using the cross entropy formula, and apply gradient descent to find the annotator levels, L, and word complexities, C, that maximize the likelihood of the observed annotations.

A limitation of the In-Corpus Model is that it can only estimate the complexity of words whose baseword occurs in the training set (i.e., "seen" words). Because our annotated corpus consists of only 263 short passages, this limits our In-Corpus Model to making predictions for the words included in one of the 3,612 word families that correspond to the basewords found in our corpus. Authentic texts usually contain many words outside these word families. So, in the next section, we introduce a general model based on Support Vector Machines (SVMs) that works for both seen and unseen words.

4.2. General Models that Handle Unseen Words

We build two general models using SVMs. For input features, we use the annotator level learned by the In-Corpus Model and a variety of word features. Platt scaling (Platt and others, 1999) is used to convert the SVM's output classification scores into probabilities. We evaluate two SVM-

not annotate enough examples of unknown words for this analysis to conclusively show if their annotations were genuine.

[6] So that, for example, 경제 "economy" and경제적으로 "economically" are treated as the same word

[7] So that $h(r,w)$ is continuous for all learned variables, we replaced V with $e^{V'_w}$ where $V'_w = \ln V_w$, and learn V' instead of V directly. Thus, the final equation becomes $h(r,w) = \phi\left(e^{-V'_w}(C_w - L_a)\right)$.

Model	Cross Entropy	Accuracy at $t = 0.5$	Precision at 80 % Recall
Seen Words Only			
In-Corpus Model	0.328	85.5 %	73.1 %
General Models			
SVM (RBF)	0.341	84.3 %	70.8 %
SVM (Linear)	0.361	83.0 %	67.4 %
Baseline Models			
TOPIK Word Level	0.393	81.0 %	62.1 %
Word Frequency Only	0.399	80.9 %	61.0 %
Word Frequency & Length	0.398	81.0 %	61.2 %

Table 4: Evaluation metrics of prediction models

based models: one using a linear kernel, and the other using the RBF (Radial Basis Function) kernel.

To select word features useful to the classification problem, we use Pearson's correlation to compare the word complexities estimated by the In-Corpus Model to various word features. This yields 23 candidate word features, which are listed in Table 5. We use Recursive Feature Elimination with Cross Validation (RFECV) (Guyon et al., 2002) to further reduce the list of features and avoid over-fitting, but find that only the Noun feature harms the models.

4.3. Experiments

We use 10-fold cross validation repeated 10 times to evaluate these three models and compare their performance to three baseline models. We also run experiments training the best model on subsets of the training data to investigate how much annotation data is needed to achieve good results.

4.3.1. Baseline Models

The General SVM Models described in the previous section incorporate a comprehensive set of word features, which is practical only because we have a sufficiently large labeled dataset for training. Since such a resource is often lacking, many previous approaches have often used only one or two features as proxies for word complexity. Most commonly, word frequency, word length, or some combination thereof has been used, such as was done in Bott et al. (2012). For L2 learners, however, defining word complexity by the word's level of first occurance within a graded corpus may produce more accurate results, as was investigated by Tack et al. (2016).

To determine if similar results could be achieved with these approaches, we build three baseline models in the same manner described in the previous section, but using only the word features proposed by these approaches: one using only log inverse word frequency, one using both log inverse word frequency and word length, and one using word level within a graded corpus. For the last of these, we use the TOPIK exams as our graded corpus.

4.3.2. Model Evaluation Metrics

Each of these models predict the probability that a given word is unknown by a given reader. Since the cost of misclassifying an unknown word will, in practice, often be different than that of misclassifying a known word, it is useful to consider discrimination thresholds other than 50 %. For brevity, we will call this discrimination threshold t.

Using the collected annotations as the gold standard, we evaluate these models using 10-fold cross validation repeated 10 times, comparing the predictions to the actual classes given by the annotations. We measure three metrics:

Cross Entropy - The cross entropy between the predicted probabilities and the actual classes. This metric measures the prediction accuracy of the model across all possible values of t, but can be difficult to interpret.

Accuracy at $t = 0.5$ - The percentage of annotations where the predicted class matched the actual class when the classification threshold, t, is 50 %.

Precision at 80 % Recall - The precision[8] of the model if t is set such that recall[9] is 80 %.

The last of these is the most relevant metric for our purposes because our analysis[10] indicates that identifying 80 % of unknown words should be sufficient for intermediate readers to understand 95 % of the words when reading advanced texts, which is normally sufficient for comprehension (Laufer and Ravenhorst-Kalovski, 2010).

4.3.3. Experimental Results

The results of our experiments are shown in Table 4. The In-Corpus Model achieves 85 % accuracy (see Table 4). By adjusting the discrimination threshold, it is able to identify 80 % of the annotators' unknown words with a precision of 73 %.

Unlike the In-Corpus Model, the General SVM Model using the RBF kernel can predict words not found int he training corpus, and achieves similar performance, having 71 % precision at 80 % recall. The General SVM Model using the linear kernel performs a little worse, getting only 67 % precision.

The best of the baseline models is the one that uses the TOPIK Word Level feature, but was substantially outperformed by our proposed general models, achieving only 62 % precision at 80 % recall. From this we conclude that this feature alone does not provide enough information to make accurate predictions, nor does the combination word frequency and word length.

To investigate how the size of the annotated corpus affects performance, we use the same cross validation procedure to evaluate the General SVM (RBF) Model while training on only a percentage of the training data available during each cross-validation fold, repeating the experiment with progressively larger percentages (see Figure 7). While the model performs better when a larger percentage of the training data is used, the gains are almost negligible after the percentage exceeds 60 % (or about 30,000 distinct annotator/baseword pairs). Measuring precision at 80% recall, there is less than 1 percentage point difference between the model trained on 60 % of the available data and the one trained on 100 % of the available data.

[8] Precision is the percentage of words predicted to be unknown that were actually unknown.

[9] Recall is the percentage of unknown words predicted as unknown.

[10] We computed this by calculating the density of advanced words (i.e., words that do not appear beginner or intermediate TOPIK exams) in advanced TOPIK tests.

Feature Name	Corpus	Pearson's r[†]	Definition
TOPIK Beg. Inverse Log WF	TOPIK Beginner	0.50	The log inverse word frequency (i.e., $-\ln f(w)$), of the word sampled from the corpus.
TOPIK Int. Inverse Log WF	TOPIK Intermediate	0.51	
TOPIK Adv. Inverse Log WF	TOPIK Advanced	0.38	
Inverse Log WF	Exquisite Corpus[‡]	0.50	
TOPIK Beg. DF	TOPIK Beginner	−0.53	The document frequency (DF) of the word in the corpus.
TOPIK Int. DF	TOPIK Intermediate	−0.49	
TOPIK Adv. DF	TOPIK Advanced	−0.42	
TOPIK Beg. Informativeness	TOPIK Beginner	0.02[*]	The informativeness of the word in the corpus, calculated as the Log Inverse WF divided by the phonemic length.
TOPIK Int. Informativeness	TOPIK Intermediate	0.05[*]	
TOPIK Adv. Informativeness	TOPIK Advanced	−0.01[*]	
Informativeness	Exquisite Corpus[‡]	0.08	
TOPIK DF $>$ 0 Level	TOPIK (all levels)	0.49	The lowest TOPIK level at which the word's document frequency (DF) is greater than the threshold.[b]
TOPIK DF $>$ 10 Level	TOPIK (all levels)	0.48	
TOPIK DF $>$ 20 Level	TOPIK (all levels)	0.42	
Phonemic Length	n/a	0.14	The number of 글자 (i.e., Korean vowels and consonants) in the spelling of the word.[c]
Syllabic Length	n/a	0.08	The number of syllable blocks in the spelling of the word.[d]
English Cognate	n/a	−0.12	A flag, 0 or 1, that indicates whether the word is a cognate of an English word.
Noun	n/a	0.05	The word's part of speech as determined by the `twitter-korean-text` API (Ryu, 2017) encoded using one-of-n.
Proper Noun	n/a	0.08	
Adjective	n/a	−0.01[*]	
Verb	n/a	−0.05	
Adverb	n/a	−0.04[*]	
Other POS	n/a	−0.07	

[†] The Pearson correlation between word complexity (as estimated by the In-Corpus Model) and the word feature.
[‡] Word frequencies for this general corpus are provided by the `wordfreq` Python package (Speer et al., 2016).
[*] Correlation not statistically significant ($p < 0.01$).
[a] Exquisite Corpus compiles texts from a variety of sources, including Wikipedia, Reddit, Twitter, and movie subtitles. Word frequencies for this corpus are provided by the `wordfreq` Python package (Speer et al., 2016).
[b] TOPIK Level is encoded as 0 for beginner, 1 for intermediate, 2 for advanced, or 3 for words whose document frequency (DF) is less than the threshold for all levels.
[c] For example, 앉다 (to sit) would have phonemic length of 5 because it consists of the letters (ㅏ, ㄴ, ㅈ, ㄷ, ㅏ).
[d] For example, 앉다 (to sit) has a syllabic length of 2 because it consists the blocks of 앉 and 다.

Table 5: Candidate word features for SVM models

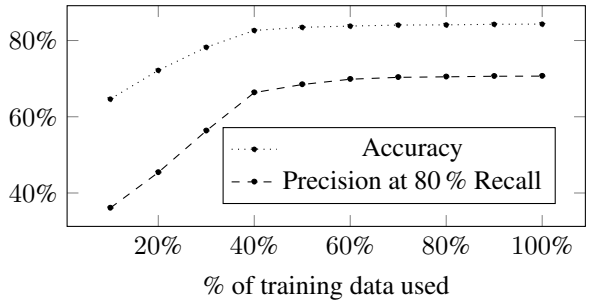


Figure 7: Evaluation metrics of SVM (RBF) Model by % of training data used

5. Conclusion and Future Work

We have demonstrated that crowdsourcing and gamification can be used to gather a large annotated corpus of words unknown by L2 learners. Furthermore, we have shown that using such a corpus as a labeled training dataset and incorporating a comprehensive set of word features, it is possible to train models that outperform previous approaches that use only a few features. The best of our models recalled 80 % of unknown words with 71 % precision, compared to the baseline's 62 %.

Our experiments also showed that similar results can be achieved with only 30,000 distinct annotator/word pairs in the training dataset. Training on larger annotated corpora showed only very small increases in performance.

There are many relevant features that our models do not yet account for: such as the word's context, attached particles, inflections, synonyms, and similarity to known words, as well as the reader's native language. In a future work, we plan to investigate how these features can be used to improve our model's accuracy.

We make our annotation data available for researchers who wish to train and/or evaluate L2 unknown word prediction models for Korean. This comes in the form of two resources: a standoff annotated corpus, and a labeled dataset that we extracted from that annotated corpus. The resources are explained in detail in the Appendix.

6. Acknowledgements

We thank the South Korea's National Institute for International Education (NIIED) for giving us permission to use passages from past TOPIK exams, and Mr. Seo Chaneweon for helping us through the approval process. We also thank Skye Payne for helping promote our survey via Twitter and the lead author's wife, Hanna Yancey, for assisting with Korean proofreading and translation. Finally, we especially thank all the annotators who took the time to take our survey.

Appendix: Description of Annotated Corpus & Labeled Dataset

Our annotation data is available for download at `http://lepage-lab.ips.waseda.ac.jp/korean-l2-unknown-words`. In this section, we describe the content and format of this resource.
The annotation data is provided in the form of 2 resources, both available in JSON and XML formats:

Labeled Dataset - A preprocessed list of what words were known and unknown by each annotator, suitable for training and validation of most unknown word prediction models.

Standoff Annotated Corpus - The fully detailed annotation data published as a standoff annotated corpus.

The fields available in each dataset are listed in Table 6. The following sections explain the individual resources in further detail.

Labeled Dataset

Each annotated token is normalized to its base word form with suffixes removed using the `twitter-korean-text` API (Ryu, 2017), and duplicate annotations are removed so that there are is at most one annotation per annotator/word pair. If the duplicate annotations are from different passages, the annotation from the earliest submitted passage is used (since we assume they may have learned the meaning of the word from the previous reading). If they are from the same passage, the word is considered to be unknown if any of the occurrences of it in that passage were annotated as unknown.

Standoff Annotated Corpus

The original annotations are available as a standoff annotated corpus. This is useful, for instance, for training prediction models that take the word's context into account, which is not possible with just the labeled dataset.
In order to extract the original text referenced by the standoff corpus, the TOPIK exam PDFs will have to be downloaded, converted to text, and stripped of whitespace.[11] However, most text extraction tools will not extract the text of these particular PDFs properly. To make it easier to correctly resolve the character offsets against these PDFs, we provide a python script that will correctly extract the text and resolve the character offsets.

[11] In order to publish research that use these copywritten documents, one must obtain permission from the National Institute for International Education Development (`www.niied.go.kr/eng/index.do`).

Bibliographical References

Bott, S., Rello, L., Drndarevic, B., and Saggion, H. (2012). Can spanish be simpler? lexsis: Lexical simplification for spanish. In *CoLing*, pages 357–374.

Crawford Camiciottoli, B. (2001). Extensive reading in English: Habits and attitudes of a group of italian university efl students. *Journal of Research in Reading*, 24(2):135–153.

Devlin, S. and Tait, J. (1998). The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic databases*, pages 163–173.

Elley, W. B. (1991). Acquiring literacy in a second language: The effect of book-based programs. *Language learning*, 41(3):375–411.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422.

Hayes, A. F. and Krippendorff, K. (2007). Answering the call for a standard reliability measure for coding data. *Communication methods and measures*, 1(1):77–89.

Krashen, S. D. (2003). *Explorations in language acquisition and use*. Heinemann Portsmouth, NH.

Laufer, B. and Ravenhorst-Kalovski, G. C. (2010). Lexical threshold revisited: Lexical text coverage, learners' vocabulary size and reading comprehension. *Reading in a foreign language*, 22(1):15.

Mason, B. and Krashen, S. (1997). Extensive reading in English as a foreign language. *System*, 25(1):91–102.

Paetzold, G. and Specia, L. (2016). Semeval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Shardlow, M. (2013). A comparison of techniques to automatically identify complex words. In *ACL (Student Research Workshop)*, pages 103–109.

Tack, A., François, T., Ligozat, A.-L., and Fairon, C. (2016). Evaluating lexical simplification and vocabulary knowledge for learners of french: Possibilities of using the flelex resource. In *Proceedings of LREC 2016: Tenth International Conference on Language Resources and Evaluation*.

Won, Y. (2016). Common european framework of reference for language (cefr) and test of proficiency in korean (topik). *International Journal of Area Studies*, 11(1):39–58.

Object/Field Name - Description	Datatype	Resource(s)[†]
Annotator		
`annotator_id` - A unique identifier for the annotator, assigned randomly.	integer	LD, SAC
`joined_datetime` - Indicates when the annotator first registered with the survey website.	datetime	LD, SAC
`reported_TOPIK_level` - The annotator's self-assessed TOPIK level that they reported in their questionnaire. Annotators below level 1 were instructed to chose level 1.	integer	LD, SAC
`estimated_language_level` - The annotator's relative language level as estimated by an analysis of his/her submitted annotations (see Section 4.1.). 0 is the average level among all annotators, and greater values indicate higher levels of proficiency.	float	LD, SAC
`native_language` - The ISO 639-1 code for the annotator's self-reported native language (selected from the list English, Chinese, Japanese, German, Russian, Vietnamese, French, Thai, Italian, or Spanish) or blank if "other" was selected.	string	LD, SAC
`browser_preferred_language` - The ISO 639-1 code for the annotator's preferred language according to the annotator's HTTP header.	string	LD, SAC
`country` - The ISO 3166 alpha-2 code for the country from which we received the annotator's most recent annotated passage, as inferred from the annotator's IP address.	string	LD, SAC
`studyreason_korean_popculture` - The annotator indicated that they were studying Korean because of an interest in Korean popculture.	boolean[*]	LD, SAC
`studyreason_family_or_friends` - The annotator indicated that they were studying Korean to communicate with family or friends.	boolean[*]	LD, SAC
`studyreason_live_in_korea` - The annotator indicated that they were studying Korean because they were living or planning to live in Korea.	boolean[*]	LD, SAC
`studyreason_study_in_korea` - The annotator indicated that they were studying Korean because they wished to study in Korea.	boolean[*]	LD, SAC
`studyreason_work_related` - The annotator indicated that they were studying Korean because of work of career-related reasons.	boolean[*]	LD, SAC
Passage		
`source` - The TOPIK test number, level, and question the passage was extracted from.	string	SAC
`url` - The URL of the file on the TOPIK website from which the passage was extracted.	string	SAC
`level` - The TOPIK level of the test from which the passage was extracted.	string	SAC
`offset` - The character offset of the start of the passage.	integer	SAC
`length` - The character length of the passage, not counting whitespace.	integer	SAC
`submitted_datetime` - Indicates when the annotated passage was submitted.	datetime	SAC
`annotation_duration_seconds` - The amount of time, in seconds, that the annotator spent reading and annotating the passage.	integer	SAC
`comprehension_rating` - The annotator's self-reported comprehension rating of the passage, using the scale provided in Figure 2.	integer	SAC
Annotation		
`offset` - The character offset of the start of the word, relative to the beginning of the TOPIK exam document.	integer	SAC
`length` - The length of the word in characters.	integer	SAC
`checksum` - A hash of the word to use as a checksum to verify that the correct word has been extracted from the original text.	integer	SAC
`base_word` - The base word form of the annotated word, with derivational and inflectional suffixes removed, as provided by the `twitter-korean-text` API (Ryu, 2017).	string	LD
`unkown` - True if the word was annotated as unknown, otherwise false.	boolean	LD, SAC

[†] Indicates the language resources that the field is applicable to: "LD" for Labeled Dataset and "SAC" for Standoff Annotated Corpus.

[*] This section of the questionnaire was not added to the website until July. For those annotators who never completed this section of the questionnaire, this field is blank.

Table 6: Dataset fields

Language Resource References

National Institute for International Education Development in Korea. (2017). *Past TOPIK Exams*. `http://www.topik.go.kr/`.

Ryu, Will Hohyon. (2017). *twitter-korean-text API*. `https://github.com/twitter/twitter-korean-text`, 4.0.

Speer, Robert and Chin, Joshua and Lin, Andrew and Nathan, Lance and Jewett, Sara. (2016). *wordfreq Python Package*. Zenodom, 1.6.1.

Crowdsourcing-based Annotation of the Accounting Registers of the Italian Comedy

Adeline GRANET[1], Benjamin HERVY[1], Geoffrey ROMAN-JIMENEZ[1], Marouane HACHICHA[1], Emmanuel MORIN[1], Harold MOUCHÈRE[1], Solen QUINIOU[1], Guillaume RASCHIA[1], Françoise RUBELLIN[2], Christian VIARD-GAUDIN[1]

[1]LS2N, UMR CNRS 6004, Université de Nantes, France

[2]L'AMO, EA 4276, Nantes Université de Nantes, France

{firstname.lastname}@univ-nantes.fr

Abstract

In this paper, we present a double annotation system for new handwritten historical documents. We have 25,250 pages of registers of the Italian Comedy of the 18^{th} century containing a great variety and amount of information. A crowdsourcing platform has been set up in order to perform labeling and transcription of the documents. The main purpose is to grasp budget data from the all 18^{th} century and to create a dedicated database for the domain's experts. In order to improve, help and accelerate the process, a parallel system has been designed to automatically process information. We focus on the titles field, segmenting them into lines and checking candidate transcripts. We have collected a base of 971 title lines.

Keywords: handwriting recognition, crowdsourcing, historical data

1. Introduction

The CIRESFI project[1] sets out to reassess a theatrical heritage that has often been considered as inferior to that of the two major, royally-privileged theaters (the Opera and the French Comedy). By studying the theater that was excluded from the system of privileges, the Italian Theater, we analyze the questions of acculturation and institutionalization including the fusion of the Opera-Comique with the Italian Comedy. By employing a mass of untapped and unpublished resources (27,544 pages of registers available at the BnF[2]), this program will take a decidedly fresh look at emerging forms of creation and the changes in the entertainment economy. To this end, CIRESFI takes up a technological challenge such as creating a tool for handwriting recognition and create an interactive database.

Information retrieval is tedious in old handwritten documents for humanities and social science researchers. The digitization of collections facilitates their consultation but it is necessary to extract information to make them fully exploitable. Sometimes, handwriting recognition systems could be set up but they require to be trained on an existing ground truth. Within the CIRESFI project, this assumption is not fulfilled since the corpus is very large, diverse and quite new w.r.t. the information extraction task. Crowdsourcing allows the use of volunteers to annotate historical documents according to our needs: (i) indicate the type of a page; (ii) detect the different areas and identify information type; (iii) transcribe as required; (iv) validate or correct previous transcriptions. In order to support and boost this label-and-transcribe process, we have implemented an automatic annotation enrichment method. The detected areas are segmented into lines. Then, the transcripts associated with these zones, if any, are reworked and validated manually. Figure 1 illustrates this approach.

2. Registers of Italian Comedy

The studied documents are registers of the Italian Comedy which record the daily, monthly and annual receipts from 1716 to 1791. In addition to these valuable information, there are additional annotations concerning the historical context through 63 seasons.

The corpus is made of 25,250 pages over the official 27,544 pages of the accounting registers since the remaining pages are not available in high quality. We have identified several types of page in addition to accounts: cover, blank page, resident statement, and introduction. The most common type is the daily accounts.

Figure 2 shows a daily account with the date and titles of the plays; receipts in the left column, and expenses in the right column; followed by actor names; and sometimes, notes.

The pages have undergone severals changes over the century. Firstly, this layout of information moved in four approaches using more or less than one page for all information. Secondly, at the beginning of the century, the Italian actors drafted the accounts themselves. Later in the century, a cashier was hired to write these accounts. Drafting language has evolved from the various Italian dialects to French.

These data show the evolution of writing which is interesting from a historical point of view but also for the natural handwriting language processing. Table 1 presents some relevant points.

Using the data extracted from the crowdsourcing process, we aim at designing an automatic reading system that focuses on the title field. This contains different levels of information ranging from a short list of the plays performed that day to specific information about a given play. The latter may be the number of acts, if it is a specific piece from the Italian Comedy, how many times it was played, if it was played in special places, or in front of the king's court.

[1]French National ANR-14-CE31-0017 program.

[2]*Bibliothèque Nationale de France*. Digital library: `http://gallica.bnf.fr/accueil/`

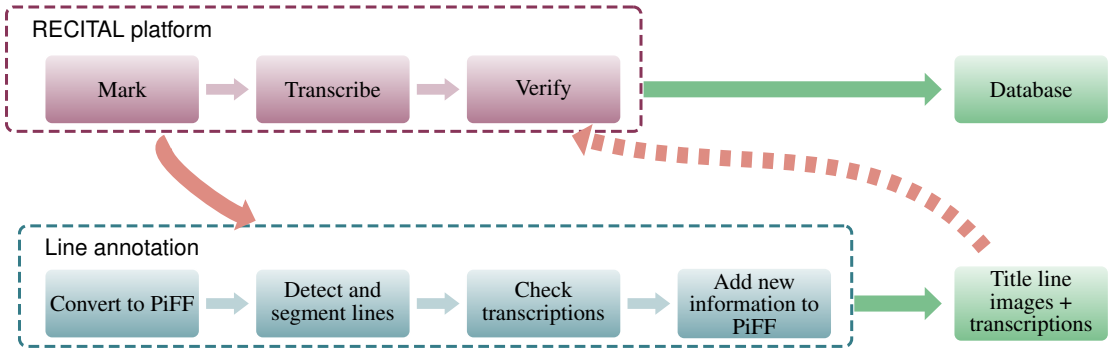


Figure 1: Dataflow of our approach for Italian Comedy documents.

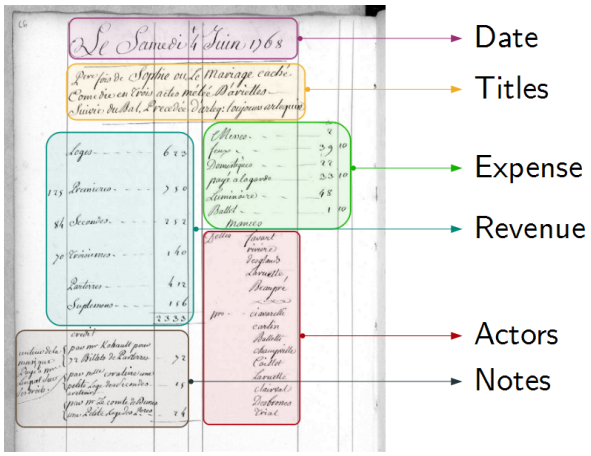
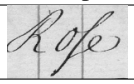
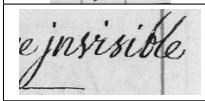
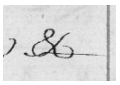
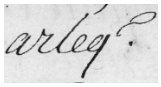


Figure 2: Example of a daily budget record for the Italian Comedy with identification fields.

Table 1: Special characters and abbreviations in the Italian Comedy documents.

Example	Transcription	Relevant Point
	"Rose"	's' has two form : long and short
	"Invisible"	using 'i' or 'j' made no difference
	"etc"	strong abbreviation to replace several words
	"arlequin"	weak abbreviation to cut few characters

3. Related work

The processing steps to be performed are to detect, identify, transcribe and validate the handwritten information contained in these documents. Severals approaches are available : perform all tasks automatically or with a participatory system. The option of having these documents transcribed by at least one specialist was rejected at the outset.The time spent to achieve this, would count in years for one person (see. section 4.3.).

As far as handwritten recognition is concerned, state of the art techniques and methods already exist. OCR softwares such as Abby®FineReader are widely used in production but failed to perform satisfying OCR results on our datasets, even on simple pages, probably due to features listed in table 1. Moreover, handwriting recognition systems could be set up but they require to be trained on an existing ground truth that we do not have for our heterogeneous dataset.

To save and re-use easily all information from the analyzed document, different XML format exits. The best-known format is the Text Encoding Initiative (TEI) dedicated to the representation of the textual components such as the transcription of one play. The main problem is that despite the great complexity of this format, it does not allow us to easily bind spatial information to transcripts as well as their type. Another one, Page Analysis and Ground-truth Elements(Pletschacher and Antonacopoulos, 2010) (Page XML) allows to store image features, layout structure and content page. However, the diversity and complexity of data contained in a daily page of Italian comedy require types and tags to be more specific and hold-back all levels of annotation. This is for those reasons we choose to use a new format called Pivot File Format (PiFF)(Mouchere et al., 2017).

4. Crowdsourcing Platform: RECITAL

Due to the previously described heterogeneity of the documents, a pure OCR-based approach to annotate the accouting registers is doomed to failure. Conversely, labeling and transcription are well-suited HITs (Human Intelligence Tasks) for crowdsourcing (Chittilappilly et al., 2016). The RECITAL crowdsourcing (CS) platform[3] is a fork of ScribeAPI[4], a framework for label-and-transcribe tasks of OCR resistant text-based documents. Although CS approaches to transcription of text-based documents are nowadays usual in digital humanities projects, the RECITAL workflow integrates a pre-labeling step and shows a high level of complexity. Indeed, we need to classify a typology of hundreds of different categories of revenues and expenditures. In addition, we also need to identify dates, titles, and many other information like actors and actresses names, cashiers, etc. Those information are written in different languages, by different people and in different types of documents (daily budgets, annual records, etc.) over the decades covered by the corpus.

[3] http://recital.univ-nantes.fr
[4] https://github.com/zooniverse/scribeAPI

4.1. Overall Workflow for Crowdsourcing

Basically, the workflow (see fig. 3) is composed of 3 activities and follows a sequential implementation:

1. **marking**: this step consists for a worker in classifying the displayed page. 8 different page types can be picked, among which three (covers, blank pages and unclassifiable) close the process and make the page retired. Then, depending on what page type was picked, a sequence of 5 "screens" is suggested. Each screen proposes about 10 marks to identify the different kinds of information (revenues, expenditures, names, etc.). Workers can mark as much elements as possible and will be asked at the end of the sequence if everything in the page has been labeled. They can also stop working whenever they want within the activity.

2. **transcribing**: this step consists in transcribing the text that has been marked in the previous step. Workers have the ability to label the mark as illegible or report the mark as misplaced if necessary.

3. **verifying**: when at least 2 different transcriptions have been proposed, they are submitted as a vote to other workers in order to achieve a consensus.

The CS annotation process offers a very large number of micro tasks, taking all types of documents together (see fig.3): 133 different elements can be classified among 8 types of documents. The marking activity is divided into a sequence of general categories of information to mark (from global to types of revenues, by the way of types of expenditures, names of actors and actresses, etc.).

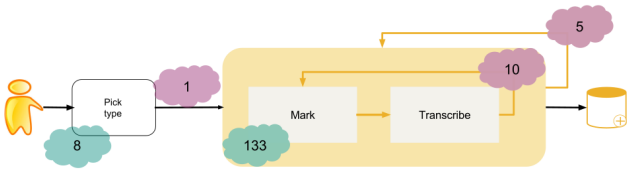


Figure 3: Simplified view of the overall workflow for one page. Voting activity is not represented. Figures in blue represent the numbers of tasks within the activity whereas figures in red represent overall existing categories.

4.2. Crowdsourcing Setup

Task assignment Marking, transcribing and verifying activities can be freely selected by a worker. Within one activity, a random task among available ones is assigned to the worker. But one can also get a direct access to a specific register of accounting records[5] to work on. Moreover, workers (if authenticated) cannot operate the verification on a transcription they made.

Management of outliers During the workflow, workers can report unclassifiable pages (for the page type label), but also mistakes (misplaced mark for example) done by other workers. Finally, illegible parts can be reported. After two reports from two distinct workers, the element (page/mark) is retired and flagged accordingly in the database.

Consensus achievement Each mark is submitted for transcription to at least 2 distinct workers. Approximate matching offered by lossy algorithms (case and punctuation insensitive comparison, ignoring whitespaces) is used to increase the ability to achieve consensus as suggested in (Matsunaga et al., 2016). If it yields to the same transcript, the proposition is accepted and the mark is retired from the workflow. Otherwise, a new vote is submitted. During the vote task, a worker can either choose one of the already existing transcripts or rather propose a new one. In case of distinct transcriptions, we want to reach a consensus based on majority voting. The threshold for majority is 75% of voters, having a number of voters between 3 and 10. When 10 distinct workers have voted without reaching a majority, the vote is closed and the annotation is considered as a dissensus.

4.3. Monitoring

Working hours We estimate that there is an average of 30 information to be transcribed per page. Based on existing data (timestamps of users' actions) on the platform, we can approximate the average time spent by one of the experts on each information: 23.5 seconds for marking, and 13 seconds for transcription. Thus, an expert would spent approximately 26 minutes per page. Given the number of 1540 hours worked per year, it would take almost 3.5 years for one full-time expert to complete the transcription of the 25,250 pages !

Users and answers The RECITAL platform is hosting 25,250 out of the 27,544 total pages. At the date of the 23^{th} of January 2018, 68 540 tasks (see Fig. 4) have been performed by 314 workers. Though this number is an indicator of the activity on the platform and reveals the worker's engagement, it is not a good overview of the overall progress in labeling and annotating the corpus. So we computed (fig. 5) the number of marks, transcriptions and pending (or closed) votes per page for each page of each register (one register per year).

Annotations and consensus On a total of 46,504 marks created (defining an area to be transcribed), 43.6% were transcribed by at least one worker. Among the marks that have at least 1 transcription, 76.3% have been transcribed by only one worker, 11% have been completed (case when 2 distinct workers make the same transcription successively), 10.1% are pending votes (at least 2 different transcriptions), and 2.6% (536 cases) have yield a consensus (we only have 7 cases of dissensus, e.g. consensus failures, in our dataset).

4.4. Limits and Perspectives

At this step of the CS process, we are able to identify some limitations and perspectives both in the methodology and the underlying framework.

Document ordering Figure 5 illustrates the issue caused by displaying direct access to documents. Therefore, we added a condition on the workflow based on an arbitrary

[5] Accounting records are grouped by year in registers. The volume of pages per register varies between 192 and 590.

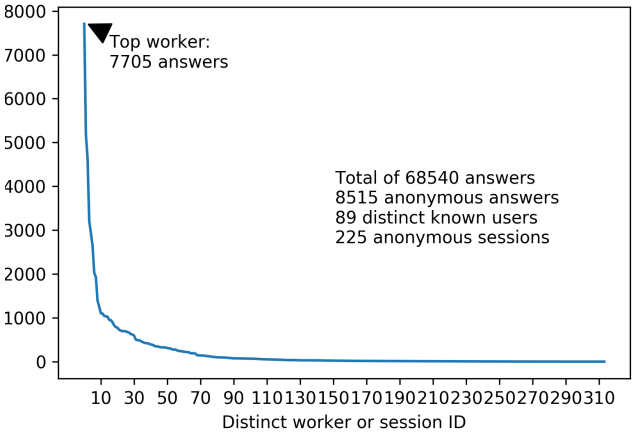


Figure 4: Distribution of answers (taking all activities together) per worker at the date of 2018-01-23. Only the workers that completed at least one task are considered.

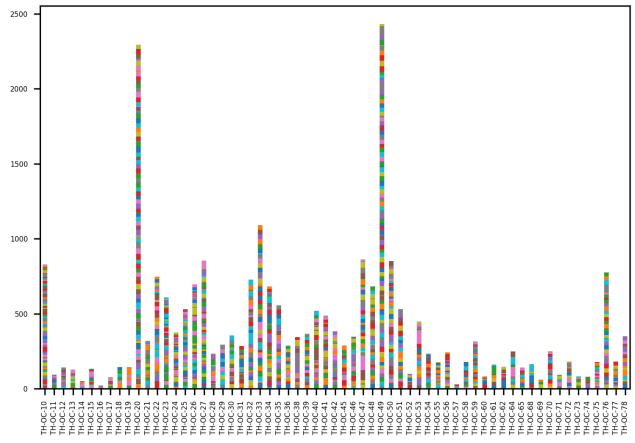


Figure 5: Overview of crowdsourcing progress at the date of 2018-01-23 by accounting registers (time-based ordering on x-axis). Colored bars are single pages and their size depends on the number of marks, transcriptions and consensus existing for this page.

order. This order is defined as follows: accounting records after 1747 are much more easier to mark and transcribe so should be prioritized. This is expected to drastically reduce time-to-complete document annotation and improve user engagement.

Free *vs.* controlled transcription Annotation may yield to slightly different transcripts from distinct workers. It depends of course on the worker's expertise, but also her own judgment of the expected result. For instance, a date like "*Du Mardy 12 Juillet 1768*" (in French) can be transcribed as a facsimile, or can be interpreted and transcribed to a canonical form like "*mardi 12/07/1768*". Normalizing is good for data post-processing and undesirable for pattern recognition. Abbreviations, approximate handwriting quality, multiple writers and multilingual documents highlight the importance and the difficulty to set up instructions, help content and input controls in such crowdsourcing workflow.

Convergence to consensus Lossy algorithms to merge transcripts by similarity are expected to reduce the effort towards reaching consensus by majority voting. Besides, it has been shown (Little et al., 2010) that an iterative updating process can be quite efficient to achieve the consensus in a transcription task.

User engagement Online platforms can improve user engagement through rewards, challenges, community management with discussion forum, etc. Such simple feedback and motivational techniques have been proved useful when you face a small crowd of volunteers inherently interested in the task (Clematide et al., 2016).

User profiling There exist state-of-the-art methods to insert fake tasks with known results to classify users (*ELICE for Expert Label Injected Crowd Estimation*) or learn from disagreement.
Finally, an interesting perspective is related to the combination of CS results and supervised segmentation and transcription. This idea is described in the next section.

5. Line Transcription

CS platforms requires a lot of workers to reach a consensus. In order to accelerate the annotation process, an automatic recognition would be very helpful. Such a system requires a training stage with annotated text lines. Figure 1 illustrates how the first collected data (even if they weren't validate yet) are used to simplify this huge annotation task. As a first step we focused on the title field, but we plan to extend to other types of field.

Converting to PiFF Firstly, we select and convert all data related to title field into a Pivot File Format (PiFF)[6]. This format is a solution promoting the exchange of information between different systems. At each step, the document can be enriched by the new associated results. The PiFF file format allows to store locations of polygons (text areas in our case) and one or several annotations attached to each of them.

Detecting and segmenting lines The coordinates specified by the users are used to crop the full title area. Then, the text line extraction algorithm (TLA) proposed by (Arvanitopoulos and Süsstrunk, 2014) is applied on it. It provides a new polygon for each line in this area. At this step, all of them are added to the PiFF without to sort error line. We associate those new polygons with all candidate transcriptions given by user for this area.

Checking transcriptions Our aim is to create a ground truth for handwriting recognition system focused on title line. So, we build a simple user interface (see Figure 6) to check if:

- the current polygon is correct, i.e it's not void line or just a thread;
- the transcription is a perfect match with the current polygon, i.e sort facsimiles and standardized transcripts.

In case several transcripts have been proposed for a title area, they are all proposed in order to associate the best one

[6] https://gitlab.univ-nantes.fr/mouchere-h/PiFFgroup

or to correct the closest. When the user valids her transcriptions and polygons for one page, once again, the associated PiFF is upgraded with new information. We validated 971 title line images and their transcriptions.

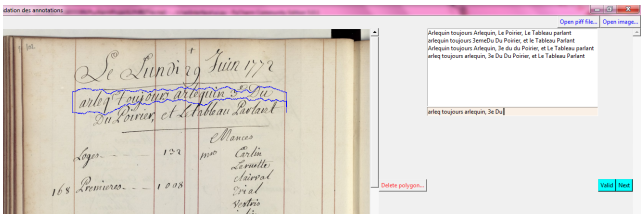


Figure 6: User interface to check manually transcriptions for title line image. The left part shows the polygons and area. The blue polygon is the result from TLA. The right part provides all candidate transcriptions. The user can validate her choice, pass to next polygon (without validation) or delete one polygon with its candidate transcriptions.

This assisted annotation system allowed us to create new resource of labeled title line images and formalize collected information in a dynamic format. The automatic handwriting recognition of our project can start the training phase with title line images.
Currently, the interaction is unilateral but in the future we are going to make it bilateral by feeding the RECITAL platform with the polygons obtained automatically with DMOS (Couasnon, 2001) for the mark information. Furthermore, we have a recognition system consisting of a Convolutional Neural Network (CNN) for the automatic features extraction; and a BLSTM-CTC neural network for the transcription part as described in (Granet et al., 2018). This deep neural network for handwriting recognition is going to provide more transcriptions for the database. This is illustrated in Figure 1 by the red dotted arrow. We can use those new data at two locations in the workflow. Firstly, the neural network results giving sufficient confidence are going to be proposed for the verifying task to accelerate crowdsourcing process. Another part of the data are going to be used to compare with crowdsourcing final data for checking data quality.

6. Conclusion

We presented two types of production to study the economy of Italian Comedy through the accounting registers. The hybrid system combining crowdsourcing platform and annotation system allows to identify and annotate unknown documents. This allowed us to collect information in several forms: 25,250 digitized pages from accounting registers; a database of crowdsourced transcriptions, and a new handwriting database focusing on the title plays, that includes 971 images. All PiFF documents produced are going to be distributed with Creative Commons Attribution-Non Commercial-Share Alike 3.0 Unported License.

7. Bibliographical References

Arvanitopoulos, N. and Süsstrunk, S. (2014). Seam carving for text line extraction on color and grayscale historical manuscripts. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, pages 726–731. IEEE.

Chittilappilly, A. I., Chen, L., and Amer-Yahia, S. (2016). A Survey of General-Purpose Crowdsourcing Techniques. *IEEE Transactions on Knowledge and Data Engineering*, 28(9):2246–2266, sep.

Clematide, S., Furrer, L., and Volk, M. (2016). Crowdsourcing an OCR Gold Standard for a German and French Heritage Corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 975–982, Portorož. European Language Resources Association (ELRA).

Couasnon, B. (2001). Dmos: A generic document recognition method, application to an automatic generator of musical scores, mathematical formulae and table structures recognition systems. In *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*, pages 215–220. IEEE.

Granet, A., Morin, E., Mouchere, H., Quiniou, S., and Viard-Gaudin, C. (2018). Transfer learning for handwriting recognition on historical documents. In *7nd International Conference on Pattern Recognition Applications and Methods, ICPRAM 2018*.

Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. (2010). Exploring iterative and parallel human computation processes. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, pages 68–76, New York, NY, USA. ACM.

Matsunaga, A., Mast, A., and Fortes, J. A. (2016). Workforce-efficient consensus in crowdsourced transcription of biocollections information. *Future Generation Computer Systems*, 56(C):526–536, mar.

Mouchere, H., Kermorvant, C., Rojas, A., Coustaty, M., Chazalon, J., and Couasnon, B. (2017). Piff: a pivot file format. In *Proceedings of the 1st International Workshop on Open Services and Tools for Document Analysis*. IEEE.

Pletschacher, S. and Antonacopoulos, A. (2010). The page (page analysis and ground-truth elements) format framework. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 257–260. IEEE.

FEIDEGGER: A Multi-modal Corpus of Fashion Images and Descriptions in German

Leonidas Lefakis, Alan Akbik, Roland Vollgraf
Zalando Research
Mühlenstraße 25, 10243 Berlin
{firstname.lastname}@zalando.de

Abstract

The availability of multi-modal datasets that pair images and textual descriptions of their content has been a crucial driver in progress of various text-image tasks such as automatic captioning and text-to-image retrieval. In this paper, we present FEIDEGGER, a new multi-modal corpus that focuses specifically on the domain of fashion items and their visual descriptions in German. We argue that such narrow-domain multi-modality presents a unique set of challenges such as fine-grained image distinctions and domain-specific language, and release this dataset to the research community to enable study of these challenges. This paper illustrates our crowdsourcing strategy to acquire the textual descriptions, gives an overview over the FEIDEGGER dataset, and discusses possible use cases.

Keywords: Crowdsourcing, Multi-modalily, Fashion, German

1. Introduction

Recent years have seen a renewed interest in text-image multi-modality and have seen the emergence of tasks such as automatically generating textual captions for a given image (Karpathy and Fei-Fei, 2015; Lu et al., 2016), using plain text descriptions to query images (Socher et al., 2014), and matching lexical tokens or constituents to regions in an image (Ma et al., 2015). This interest is driven by advances in multi-modal deep learning for computer vision (Vinyals et al., 2015; Karpathy et al., 2014) on the one hand, as well as the availability of paired text-image datasets on the other.

Multi-modal text-image datasets. Commonly cited multi-modal datasets either consist of user-captioned images from the web, such as FLICKR (Hodosh et al., 2013; Plummer et al., 2015) and online news (Hollink et al., 2016), or images for which crowd workers have produced visual descriptions (Rashtchian et al., 2010), such as the popular COCO dataset (Lin et al., 2014). In both cases, the textual data directly describes image content, thus enabling the above-mentioned lines of research.

However, these datasets are often restricted to English language text and typically of relatively broad domain; The FLICKR caption datasets for instance contain images including landscapes, animals, and everyday scenes while the COCO dataset is similarly broad but contains more items per image. This makes such datasets difficult to apply for study of multi-modality in more narrow domains. In the domain of fashion items for instance, images are broadly similar and are often distinguished only by fine-grained differences such as the material, the neckline, brand logos, the cut and the style of the hem. Similarly, the language used in fashion is domain-specific, tailored specifically to highlight such fine-grained differences. We argue that such narrow domains present unique research challenges that require specialized multi-modal datasets.

A multi-modal text-image corpus for fashion. With this paper, we introduce a novel dataset for research in narrow-domain multi-modality, called FEIDEGGER[1]. Contrary to previous datasets, we restrict the domain to images of one type of fashion item, namely dresses, and German-language visual descriptions. The dataset consists of 8,700 fashion items, each with a high resolution image and 5 independently collected textual descriptions of the item. The images are of each fashion item alone in front of a white background. Crowd workers were instructed to inspect each image and then produce a plain text description of the fashion item. For an example item in this dataset, see Figure 1.

In the remainder we provide a description of our dataset and describe the design of our crowdsourcing approach. We qualitatively analyze the obtained data and find that the crowdsourced descriptions are of high quality and are finely detailed, which we attribute to our careful choices of experimental parameters. Finally, we provide a description of how we package this data and discuss expected use cases. We release FEIDEGGER to the research community to further research in narrow-domain multi-modality.

2. Dataset Creation

In constructing FEIDEGGER we employed a crowdsourcing approach to produce accurate and succinct descriptions of each fashion image that make reference to fine-grained non-generic image features. Our pipeline required careful monitoring of individual workers' performance using automated evaluation of test-questions as well as performing pilot studies. However, we did not place very high demands on language correctness in terms of spelling and grammar. Rather we accepted average language use as might be expected in user reviews or forums on the web. The details of our pipeline are given in the following sections.

2.1. Task Design: Pilot Study

We first conducted a pilot study using the crowdsourcing platform CROWDFLOWER[2] to test the design of our crowdsourcing task and identify potential quality issues.

[1] A rough acronym of "fashion image data and descriptions in German"

[2] https://www.crowdflower.com/

Image 1



Description 1

Langes weißes Kleid mit Bugs Bunny Musterung an der Seite des Kleides. runder Ausschnitt und kurze Ärmel.

(engl.) Long white dress with Bugs Bunny pattern at the side of the dress. Round neckline and short sleeves.

Description 2

Schlauchkleid in weiß mit sehr kurzen Armen und einem Bugs Bunny Aufdruck auf der linken Seite. Arme und Hals haben einen schwarzen Streifen.

(engl.) Tube dress in white with very short sleeves and a Bug Bunny print on the left side. The sleeves and the neck have a black stripe.

Figure 1: Example item in FEIDEGGER: For each fashion item we provide an image and 5 crowdsourced descriptions (only 2 presented here). The image is always a still of the fashion item itself in front of a white background. Textual descriptions are in German and typically consist of 2-4 short sentences. English translations are provided for the purpose of illustration in this Figure, but are not part of the dataset.

Task design. Given the image of a fashion article workers were instructed to provide a German language description of the item. They were instructed to go into detail and write about 5 sentences. In order to discourage non-native speakers to participate in the task, we provided instructions in German language only and required workers to first complete a German-language tutorial.

Study parameters and results. We conducted the initial study over 1000 fashion items and restricted workers to a maximum of 50 descriptions each. We restricted the pool of workers to (a) those based in a German-speaking country and (b) *level 3* workers, who are the highest ranked workers according to the internal CROWDFLOWER system of evaluation.

Upon manual inspection of the results we found the produced textual descriptions to be of a very high quality, likely due to our very restrictive parameters in selecting workers. However, we also identified the following issues:

Short descriptions Although instructed to provide descriptions at reasonable length, some crowd workers provided very short descriptions, sometimes only a few words in length.

Unspecific descriptions We found some descriptions of reasonable length to be generic descriptions that some crowd workers simply re-used for each fashion item, sometimes directly copied from the tutorial examples.

Non-German text Finally, there were a number of instances in which workers had responded in another language than German, such as Polish.

While the issues of short and non-German crowd answers are fairly straightforward to address with automatic verification methods, the problem of catching workers that provide unspecific, non-matching or low-quality descriptions proved inherently more difficult. We therefore decided to adopt two strategies for increasing quality, namely curating workers and automatic quality checks. In the next two sections, we give an overview of each.

2.2. Building a Pool of Curated Workers

Our first and most important measure was to identify a pool of workers that could consistently and reliably create content to the level of quality we required. To this end, we constructed a crowdsourcing task that was open to any worker meeting the minimum requirements, hereafter referred to as the *trial task*. This task mirrored the task design of the actual crowdsourcing, but restricted each worker to provide a maximum of 100 descriptions, which we treated as a sample of the workers' ability to provide high quality descriptions. This sample was then manually assessed by experts.

Results. The task was executed for a period of 3 weeks. Approximately 150 distinct crowd workers participated in the task, of which we admitted 50 into the pool of curated workers. These workers were used for the final crowdsourcing task of generating image descriptions.

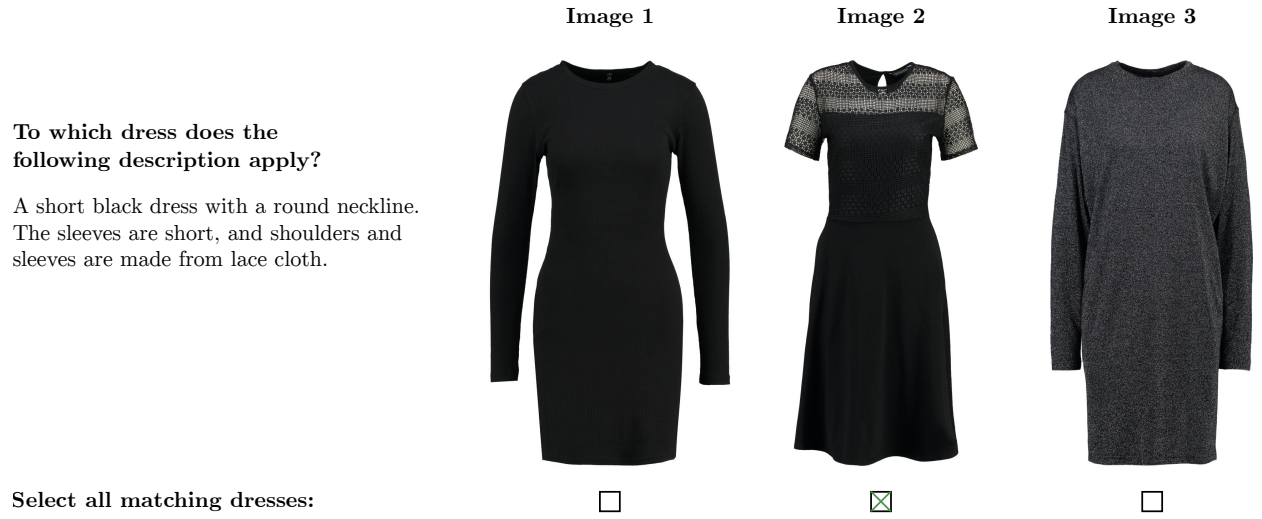


Figure 2: Example evaluation task. A description is paired with three images, one of which is the original source image for which the description was generated. The evaluation worker is instructed to select all images to which the description applies. In this example, the worker selects *image 2*, which is the correct source. *Note: Original task is in German, this example was translated for readability.*

2.3. Automatic Quality Control

In addition to curating workers, we employed three simple methods for automatic quality control to ensure that workers would not submit short or unrelated descriptions.

The first was to add a simple regular expression that checked whether each submitted text contained at least 10 distinct words. Similarly, the second checked whether at least one word of the submitted text was a German stopword, to identify responses that were either in a different language or not well-formed. Though these checks were rather coarse, they were nonetheless successful in filtering workers and descriptions to those of reliable quality. A manual inspection of both the cleared and rejected results showed that no non-German text was erroneously accepted while in only one case was work in German rejected as non-German.

The third was to introduce a series of test questions within the data presented to the workers. These consisted of images that had been manually inspected and for which a series of words at least one of which should appear in any reasonable description had been determined. For instance, for the images in Figure 1, we would expect the German words for black (*schwarz*) and white (*weiß*) to be mentioned in the description, as well as some synonym of either rabbit or Bugs Bunny. Workers that failed automatic quality control were removed from the pool and their work discarded.

2.4. Full Crowdsourcing Task

After building up the pool of curated workers and establishing the quality controls, we ran a large crowdsourcing experiment. Our goal was to annotate 8,764 fashion items with 5 descriptions each. Annotation was completed after 4 weeks of crowdsourcing.

3. Quality Estimation

In order to assess the quality of the gathered descriptions, we set up a separate crowdsourcing task. Since our main goal in creating FEIDEGGER was to acquire detailed, non-generic descriptions, the evaluation task was set up to evaluate both whether the descriptions were accurate *and* discriminative.

Evaluation task. To accomplish this, we designed the *evaluation task* as follows: Each crowd-generated description was paired with three images of fashion articles. One of the three images was the *source image*, i.e. the image for which the description had been produced. The other two images were other items that are visually similar to the source image, determined using pre-computed image embeddings over a large fashion catalogue (Bracher et al., 2016). One such example pairing of a description and three similar images is depicted in Figure 2.

Given such a pairing, a crowd worker was asked to select all images to which the description applies. Note that the worker was not informed that the description's origin was

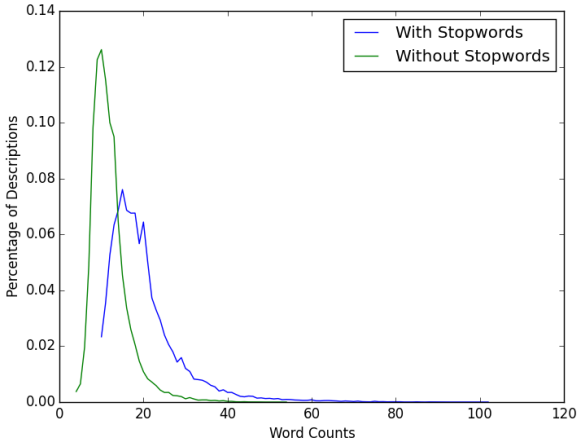


Figure 3: Plot of word count frequencies.

only of one of the images and was given the choice to select more than one image. Thus the task tested both for correctness (whether the source image was chosen) and discriminativeness (whether other images beyond the source image were also chosen). Figure 2 illustrates one such evaluation task.

Experimental results. We ran experiments on 4,000 description-image triplets. In 96.5% of cases the worker deemed the description relevant to at least one of the images. Furthermore, in 97% of those cases the worker picked the correct image as being relevant to the description, while in 96.35% of the cases the worker chose only that image. These results indicate that descriptions are generally of high quality and discriminatively match the source image.

Length of descriptions. To give an overview of the length of crowd-provided descriptions, we computed statistics on word count, as illustrated in Figure 3: Descriptions have an average total of 20.26 words, with a median of 18, and consist of, on average, 2.23 sentences. Stopwords make up roughly 40% of the data.

4. Data Release and Outlook

We release the data via a specifically dedicated website. This data may be useful for experiments various text-image tasks such as captioning and image retrieval, to compare the quality of approaches that work well on general datasets such as COCO and FLICKR with narrow-domain data, and to research approaches that work well in this domain.

Future work will focus on extending the scope of the crowdsourced image description along two dimensions. On the one hand, we will include other types of fashion items besides dresses, such as shoes and shirts. On the other hand, we aim to repeat crowdsourced data gathering efforts for languages other than German, such as English, French, and Dutch. The medium-term goal is to create a multi-modal dataset that for each fashion item contains descriptions in several languages.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no 732328 ("FashionBrain").

5. Bibliographical References

Bracher, C., Heinz, S., and Vollgraf, R. (2016). Fashion dna: Merging content and sales data for recommendation and article mapping. *arXiv preprint arXiv:1609.02489*.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Hollink, L., Bedjeti, A., van Harmelen, M., and Elliott, D. (2016). A corpus of images and text in online news. In *LREC 2016, 10th International Conference on Language Resources and Evaluation*.

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137.

Karpathy, A., Joulin, A., and Li, F. F. F. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Lu, J., Xiong, C., Parikh, D., and Socher, R. (2016). Knowing when to look: Adaptive attention via a visual sentinel for image captioning. *arXiv preprint arXiv:1612.01887*.

Ma, L., Lu, Z., Shang, L., and Li, H. (2015). Multimodal convolutional neural networks for matching image and sentence. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2623–2631.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. (2010). Collecting image annotations using amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 139–147. Association for Computational Linguistics.

Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., and Ng, A. Y. (2014). Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Toward a Lightweight Solution for Less-resourced Languages: Creating a POS Tagger for Alsatian Using Voluntary Crowdsourcing

Alice Millour, Karën Fort
STIH/Université Paris-Sorbonne, France
{alice.millour, karen.fort}@paris-sorbonne.fr

Abstract

We present here the results of an experiment aiming at crowdsourcing part-of-speech annotations for a less-resourced French regional language, Alsatian. We used for this purpose a specifically-developed slightly gamified platform, `Bisame`. It allowed us to gather annotations on a variety of corpora covering some of the language dialectal variations. The quality of the annotations, which reach an averaged F-measure of 93%, enabled us to train a first tagger for Alsatian that is nearly 84% accurate. The platform as well as the produced annotations and tagger are all freely available. The platform can easily be adapted to other languages, thus providing a solution to (some of) the less-resourced languages issue.

Keywords: crowdsourcing, less-resourced languages, POS tagging, Alsatian

1. Introduction

Despite the progress made in unsupervised learning, manually annotated corpora are still necessary both to develop and to evaluate natural language processing (NLP) tools. However, building such corpora is notoriously expensive (see, for example (Böhmová et al., 2001)). For less-resourced languages, the (lack of) availability of language experts represents yet another obstacle to overcome.

We hypothesized that contributing to the creation of NLP tools for their language would be an incentive for non-expert speakers, especially for languages known for their speakers' activism, to participate in such a project. We therefore developed a lightweight crowdsourcing platform, `Bisame`[1], which enables participants to collaboratively produce part-of-speech (POS) annotations. We tested it on a French regional language with the appropriate specificities, i.e. activism and easy access to the Internet: Alsatian.

We present here the related work performed on Alsatian, POS tagging less-resourced languages and crowdsourcing linguistic annotations. We then describe the methodology we used and the results we obtained. Finally, we discuss the limits of this work.

2. Related Work

2.1. Alsatian

Alsatian is a generic term for the continuum of Germanic dialects spoken in Alsace and part of Moselle, two diglossic regions where French and Alsatian dialects coexist.[2] Beyond this generic glotonym lies a set of dialectal subsystems (Malherbe, 1983). In fact, low Alemannic, the main variant of Alsatian, can itself be divided into two subsets: Northern (NV) and Southern (SV) low Alemannic. Besides, Strasbourg presents a slightly different variant of Northern low Alemannic (STRV), tinted with Franconian, as evidenced by the use of *Dienschtdàj* for "Tuesday", instead of the Alemannic *Zischdig*. Thus, the 7 (phonetically different) forms *Baum*, *Bààm*, *Bœm* (SV); *Bauim*, *Bàuim*, *Bäum* (NV); *Baam* (STRV) can be found for the word "tree". Additionally to the dialectal variation and although the Orthal spelling guidelines (Crévenat-Werner and Zeidler, 2008) have existed since 2006, no orthographic convention is recognized yet as the legitimate standard among the speakers. Consequently, elided forms such as *m'r* for *mir* ("we"), or *d'* for *die* ("the") coexist depending on the spelling habits of the writer and independently of the variant.

This two level variation scheme results in a great heterogeneity of the existing written forms for a given word.

So far, works on POS tagging Alsatian corpora are very exploratory: only Bernhard and Ligozat (2013) have proposed a method based on the transposition of grammatical words into German, enhancing the performances of the German tagger on the transposed corpora. This method reaches up to 85% accuracy but has a low potential for improvement.

2.2. POS Tagging Less-resourced Languages

A variety of methods have been developed to overcome the data scarcity issue regarding the POS tagging task. They may involve additional resources such as bilingual corpora used with recurrent neural networks (see for instance (Zennaki et al., 2016)), annotation projection (see for instance (Agić et al., 2016)), or weakly supervised methods (see for instance (Li et al., 2012), showing the benefits of using the `Wiktionary` as an external resource). No such resources are available for Alsatian[3], we therefore decided to rely only on freely available raw corpora.

2.3. Crowdsourcing Linguistic Annotations

Crowdsourcing consists in issuing an open call encouraging people (a "crowd") to participate in producing data (encyclopaedia entries, a drawing, a vote etc.), today mainly through the Internet. Many taxonomies of crowdsourcing have been proposed (see (Geiger et al., 2011) for an overview). We suggest to consider it along two axes (Fort, 2016): i) the remuneration (or not) of the activity and

[1] See: `http://bisame.paris-sorbonne.fr`.

[2] In spite of the decrease of the family linguistic transmission, a study registered 550,000 Alsatian speakers (Barre and Vanderschelden, 2004).

[3] The Alsatian `Wiktionary` has disappeared, being merged with the Alemannic `Wikipedia`. The Alsatian section of this `Wikipedia` contains around 50,000 words once very similar articles have been excluded.

ii) the awareness of the participants that they are producing data (as this can be hidden underneath a playful interface, for instance). This typology allows to distinguish between transparent, voluntary crowdsourcing (for instance, Wikipedia), microworking platforms which propose rather transparent tasks and a micro-remuneration (such as Amazon Mechanical Turk) and games with a purpose, that more or less hide the task being performed (like, for example, in Phrase Detectives[4] (Poesio et al., 2013) or ZombiLingo[5] (Guillaume et al., 2016)).

Amazon Mechanical Turk has been used by many researchers, directly or through CrowdFlower, including to have POS tags annotation produced (Hovy et al., 2014). In addition to the ethical issues it raises (Fort et al., 2011), this kind of platform is not adapted to the languages we target, as very few (if not none) microworkers are fluent speakers of these languages. Besides, microworking platforms do not allow to train annotators, only to test them.

Games with a purpose have proven efficient at getting good quality linguistic data at lower cost than traditional methods (Chamberlain et al., 2013). Yet, developing a full-fledged game is a long-term endeavor which requires a range of skills (Web development, gaming mechanisms knowledge, user experience design, advertising) and has to be made profitable in the long run. [6] We thus chose to develop a lightweight platform with very few playful elements so far, namely a basic point system, a scoreboard, and a progress bar indicating the evolution of the annotation state of the corpus. We are not aware of any other voluntary crowdsourcing application for POS tagging annotation.[7] However, related tasks have been successfully achieved by volunteers, such as the annotation of suicide notes (Pestian et al., 2012) or text message translation in an humanitarian emergency context (Munro, 2013).

Finally, there exist some generic platforms for citizen science, such as Crowd4U[8] or Zooniverse[9], but none presents a linguistic application yet.[10]

3. Methodology

3.1. Tagset

For the sake of adaptability, we chose to work with the universal POS tagset (see Appendix I), which synthesizes the tagsets of 22 languages and can easily be adapted to the needs of each languageintroduced by (Petrov et al., 2012).[11] In fact, the only modification we initially made was to have the X category ("Others", a catch-all category hard to interpret) to match only the cases of code-switching which cannot be analyzed as loan words.

We further became aware that the contractions that we do not automatically split (such as *am*: *an+dem* ("at the")) generated confusion and frustration among the participants. We thus introduced the ADP+DET category, and accordingly corrected the existing annotations to abide by this new tagset.

3.2. Corpora and Lexicons

The training corpus (T), annotated via the platform, consists of 333 sentences, adding up to 6,878 tokens. The low quantity of raw corpus as well as the urge to produce a freely available annotated corpus have forced us to follow a pragmatic approach and to gather an "opportunistic" corpus (McEnery and Hardie, 2011). By definition, this generates a bias in term of content: 80% of the corpus is made of articles from the Alemannic Wikipedia, 20% being a novel kindly provided by a participant-author.

Our evaluation corpus (E) is made of 4 texts adding up to 1,468 tokens (102 sentences), manually annotated by expert linguists from LiLPa, in Strasbourg. As shown in Table 1, both corpora are made of at least two variants.

Name	Nb. Sentences (Nb. tokens)	Content
T_{SV}	267 (5,110)	Wikipedia articles
T_{STRV}	66 (1,768)	Novel
E_{SV}	47 (875)	Wikipedia articles
$E_{\text{NV},1}$	26 (362)	Theater piece
$E_{\text{NV},2}$	29 (231)	Recipes

Table 1: Description of the training and evaluation corpora.

We have also integrated two lexicons to the tagger training process: i) a lexicon of grammatical words (determiners, pronouns, prepositions, conjunctions, particles) and frequent verbs and adverbs, summing up to 322 entries, which has been compiled by Bernhard and Ligozat (2013), ii) a lexicon with more than 40,000 entries from the Office for Alsatian Language and Culture (OLCA) bilingual lexicons , a bilingual dictionary compiled by the Culture and Heritage of Alsace Association (ACPA) and a multilingual French-German-Alsatian dictionary (Adolf, 2006).

The integration of these various sources increases the coverage of the dialectal and scriptural variants. We can for instance find the following entries for the word "elbow": *Elleböje* (OLCA), *Elleboje* (ACPA); *Ellaboja* (OLCA), *Ällabooga* (ACPA).

3.3. Preprocessing of the Corpora

The gathered texts were tokenized using a specific Python script, which we completed when cases of wrong segmentation –due to unknown spelling habits– were brought to our attention by the participants. For instance *'r* can either be considered as a separated token when placed after a verb (e.g. *hät'r*, "he has") or as part of a token containing an elided vowel (e.g. *d'r*, "the").

Both the training and evaluation corpora have been pre-annotated with two taggers: i) the Stanford POS Tagger (Toutanova et al., 2003) applied to the texts after

[4] See: http://anawiki.essex.ac.uk/phrasedetectives/.

[5] See: http://www.zombilingo.org.

[6] For more details, see (Lafourcade et al., 2015).

[7] Such a platform might exist though, as some did not get any dedicated scientific publication, like LanguageQuiz: http://quiz.ucomp.eu/.

[8] See: https://crowd4u.org.

[9] See: https://www.zooniverse.org.

[10] A specific platform is under development as we write this paper: https://lingoboingo.org/.

[11] See: http://universaldependencies.org/u/pos/all.html.

a transposition of grammatical words in German, following the methodology defined in (Bernhard and Ligozat, 2013). and ii) `MElt` (Denis and Sagot, 2012), that we regularly trained on the annotated corpus. These pre-annotations were used to provide suggestions to the participants: when the taggers disagree, the two categories they produce are proposed to the participants, while when the agree, the consensual tag can be directly validated: this fastens the annotation process on frequent and weakly ambiguous words and has led to an increase of annotated sequences during one annotation session.[12]

3.4. Annotating with `Bisame`

To be granted access and actually produce annotations, participants must go through a four sentences training phase during which they must annotate correctly every token. The production phase also consists in annotating a sequence of 4 sentences, from which one is taken from the evaluation corpus and is used to give the participant a confidence score at the end of each sequence. The confidence score given to a participant P having produced $NbAnn_{Ref}$ annotations on sentences coming from C_{Ref} is the ratio of correct categorizations:

$$Score_P = \frac{NbAnn_{Ref,Correct}}{NbAnn_{Ref}}$$

We set $ScoreAnn_{T,P,C_i}$, the confidence score for an annotation produced by P with the category C_i, to $Score_P$ at the time of the annotation. Each token being annotated by different annotators with potentially concurrent tags, we further decide on a unique category C_T: a confidence score for each category C_i is calculated averaging the scores of the Ann_{T,P_j,C_i} produced:

$$Score_{T,C_i} = \frac{\sum_j ScoreAnn_{T,P_j,C_i}}{\sum_{i,j} ScoreAnn_{T,P_j,C_i}}$$

and $C_T = \arg\max_i(Score_{T,C_i})$. For instance, in the sentence ***Dr Mentelin hàt sina Stroßburger Drukaréi grinda.*** ("Mentelin has founded his Strasbourg printing house"), the token $T =$ ***Stroßburger*** has been annotated with three different tags $\{C_{i=1..3}\}$ by five participants, $\{P_{j=1..5}\}$. Table 2 illustrates this case, that results in choosing $C_{\textit{Stroßburger}} =$ADJ, which is the correct tag.

C_i	$ScoreAnn_{T,P_j,C_i}$	$Score_{T,C_i}$
PROPN	0.935	0.24
ADJ	0.875	**0.69**
	0.846	
	0.938	
NOUN	0.25	0.07

Table 2: Choosing the most probable tag for a given token.

4. Results

4.1. Annotated Corpus

So far, 202 people have created an account, 72 have completed the training phase, and 46 have collaboratively produced 18,917 annotations. The platform was released in May 2016 but the annotations have mainly been produced during short periods of time adding up to 73 days. Our experience confirms a well-known phenomenon described in (Chamberlain et al., 2013): a minority of participants contributes a lot (in our case the 10 most active participants produced almost 90% of the annotations).

Among the annotations, 8,244 were added on the evaluation corpus, thus enabling us to evaluate the quality of the collected annotations. The accuracy and weighted average F-score for these annotations reach 93%. This quality is above that obtained that obtained by Hovy et al. (2014) (80% accuracy) when crowdsourcing POS tags on Twitter data *via* `CrowdFlower`. This shows the benefits of being able to train the participants when designing a crowdsourcing task of this kind. Accordingly with Guillaume et al. (2016), we observe that the quality of the annotation raises with participation: with twice as many annotations, the average F-score has raised by more than 40%. The remaining 10,673 annotations were used to annotate a raw corpus of 6,878 tokens (Alice Millour, Karën Fort, 2017).

4.2. Tagger Performance

	E_{SV}	$E_{\text{NV},1}$	$E_{\text{NV},2}$	E_{SV}+$E_{\text{NV},1}$+$E_{\text{NV},2}$
T_{SV}	**83.7**	78.7	71.3	77.9
Unk. Tokens	40%	65%	62%	52%
$T_{\text{SV}} + T_{\text{STRV}}$	82.3	**82.8**	**71.8**	**79.1**
Unk. Tokens	40%	37%	61%	47%

Table 3: Accuracy of the trained taggers.

Table 3 shows the accuracy of the two `MElt` taggers we trained, according to the training corpus we used. The first corpus contains only the Southern variant, while the second experiment includes the corpus written in the Strasbourg variant, which is closer to Northern low Alemannic. The addition of the lexicons described in Section 3.2. to the training of `MElt` led to an increase of nearly 31% of the accuracy on unknown words, reaching 70% on average, and to improve the overall performances by 6%. Unsurprisingly, the best performance (83.7%) is reached when the training (T_{SV}) and evaluation (E_{SV}) corpora are made of the same variant of the language. We observe that adding the Strasbourg variant to the training process positively impacts the performance on the Northern variant corpora while leading to a drop in accuracy on the Southern variant corpus. This emphasizes the need to integrate the variation to the training process, and not to treat Alsatian dialects as a whole.

Overall, an analysis of the F-scores per tag shows that the lowest performances (lower than 0.5) concern the less represented tags (PART, SCONJ, SYM, ADP+DET) as well as the X category which represents 2% of the evaluation corpus and a F-score of 0.3. While the tag proportions are roughly the same between the training and the test corpus (both taken as a whole), as only the PROPN and X are slightly overrepresented, the distribution of tags is not balanced among the evaluation corpora.[13] In particular, the category PROPN is proportionally 2.2 times more present in E_{SV} than in the training corpus and is mistaken for NOUN in almost 60% of the cases. Another frequent error is the

[12]The two taggers agree in 50% of the cases, and this consensual tag is corrected by the participants in 12% of the cases.

[13]The full tag distributions can be found in Appendix II.

confusion between AUX and VERB (in 30% of the cases) and between VERB and ADJ (in 25% of the cases).
Our performance on Alsatian is lower than what has been obtained with similar amounts of resources for other languages: for instance, with 100 sentences (less than 3,000 tokens), Fort and Sagot (2010) trained a tagger (MElt) reaching 86.6% accuracy on section 23 of the Penn Treebank. On a less canonical language, Vergez-Couret et al. (2014) showed that the Talismane parser (Urieli, 2013) can be trained for Occitan (another French regional language) to reach 89% accuracy with 2,500 tokens, using a lexicon of 225,000 entries. We therefore think that beyond dialectal variation, orthographic inconsistencies might be the source of the lower performances obtained on Alsatian.

5. Discussion

5.1. Corpus Size and Evaluation

The small size of our evaluation corpus has led us to assess the quality of the crowdsourced annotations on a very small set of cases. The corpora also revealed unbalanced in terms of tag proportion. We are now looking into ways of building a minimal reference corpus specifically designed to address the difficulties inherent to POS tagging. In order to collect more freely available corpora we are also planning to ask participants to contribute to raw corpus building within the platform, following a suggestion from Liberman (2016).
Another consequence of the lack of reference corpus is that we could not evaluate the tagger we trained on a corpus which had not already been used to train the participants to the task, or to evaluate them. Consequently, a bias exist in our evaluation of the tagger. Moreover, Fort and Sagot (2010) have shown that pre-annotation can have a negative impact on the quality of annotations, especially on less-trained participants. This bias has not been evaluated so far on our platform. Finally, we identified recurring errors in the participants annotations due either the task complexity or to unclear guidelines. The most frequent errors concern a confusion between ADJ and ADV categories, and AUX which is mistaken for VERB in 75% of the cases. We also noticed a confusion between code-switching (annotated with X) and loanwords (annotated with their part of speech). We thus plan to work on adapting the guidelines we provide to tackle these difficulties.
Improving our user evaluation method is also necessary, as a manual inspection of the annotations has revealed that a participant with a high confidence score (0.95, the average being 0.82) had produced some bad quality annotations (some were recurring errors, others were due to the bad automatic translations provided by his browser), without significant impact on his confidence score.

5.2. Motivating the Participants

With respect to our crowdsourcing experience, the hypothesis we initially made regarding the motivation of the speakers could not be completely validated: official structures and local media have revealed ineffective to advertise the platform, and recruitment using social networks has revealed time consuming. The platform is not a game, and motivating the participants to contribute and to come back to contribute again has been challenging. So far, only 37% of the participants came back on the platform at least twice. Nonetheless, we have observed that adding a scoreboard and a progress bar have had a positive effect on participation and quantity of data produced during a session (Millour and Fort, 2017). We are therefore considering the development of a couple of simple gamification features to make the annotation task less tedious. In fact, according to the feedbacks we gathered from the participants, we believe that diversifying the available tasks and enhancing the community feeling within the platform to develop the *social* incentive (Poesio et al., 2013) would help heading towards a more autonomous platform.

5.3. Dealing with Variations

This experiment showed that, at least from a NLP point of view, Alsatian should not be considered as a unified language, as we observed that a multi-dialect training corpus can lead to a drop in performance on some portions of the evaluation corpus. Moreover, the use of external lexicons has proven efficient to enhance performance. Our future work will therefore involve focusing on crowdsourcing a multi-variant tag dictionary to complement our token-supervised (labeled sentences) with a type-supervised (tag dictionaries) approach, as defined and suggested by Garrette and Baldridge (2013). We also intend to address orthographic variation in the manner of Samardzic et al. (2015) on Swiss German, normalizing to a single representation before training the tagger.

6. Conclusion

Thanks to the voluntary crowdsourcing platform we developed, we collected 18,917 annotations, thereby building the first open source POS annotated corpus for Alsatian. We used this corpus to develop the first tagger specific to Alsatian. The quality of the annotations gathered (93%) as well as the tagger performance (reaching 83.7% in accuracy) show that our method is valid. Nonetheless, some improvements with regard to both language specific considerations and methodological points should be provided to tackle the obstacles and biases we identified and discussed in Section 5. The platform source code is freely available on GitHub[14] under the CeCILL v2.1 license.[15] It can be adapted to any languages (an instance already exists for Guadeloupean Creole) for which i) a minimal reference, ii) an open source raw corpus, iii) adapted annotation guidelines and, if available, iv) a baseline tagger exist. Both the corpora and the tagger model are freely available under the CC BY-NC-SA license.[16]

Acknowledgements

Part of this work has been funded by the French Ministry of Culture (DGLFLF), through the PLURAL project. We wish to thank all the participants of Bisame for their motivation and valuable comments, as well as D. Bernhard and L. Steiblé (LiLPa) for their help.

[14] See: https://github.com/alicemillour/Bisame.
[15] See: http://www.cecill.info/.
[16] See: https://bisame.paris-sorbonne.fr/corpora.

Appendix

Appendix I

Open classes	ADJ	ADV	INTJ	NOUN	PROPN	VERB		
Closed classes	ADP	AUX	CONJ	DET	NUM	PART	PRON	SCONJ
Others	SYM	X	PUNCT					

Table 4: The universal POS tagset.[17]

Appendix II

	T_{SV} (`Wikipedia` articles)	T_{STRV} (Novel)	E_{SV} (`Wikipedia` articles)	$E_{\text{NV},1}$ (Theater piece)	$E_{\text{NV},2}$ (Recipes)
ADJ	6%	3%	3%	3%	6%
ADP	12%	7%	10%	7%	9%
ADP+DET	3%	2%	4%	3%	2%
ADV	6%	5%	5%	7%	6%
AUX	4%	6%	6%	2%	0%
CCONJ	4%	4%	3%	6%	7%
DET	11%	8%	13%	8%	12%
INTJ	0%	0%	0%	1%	0%
NOUN	17%	10%	14%	9%	21%
NUM	3%	0%	4%	0%	3%
PART	1%	1%	0%	1%	0%
PRON	6%	6%	3%	12%	2%
PROPN	3%	1%	8%	4%	0%
PUNCT	13%	34%	10%	19%	13%
SCONJ	1%	1%	0%	2%	1%
SYM	1%	0%	0%	0%	0%
VERB	10%	9%	8%	11%	18%
X	1%	0%	9%	4%	0%

Table 5: Tag distribution in the training and evaluation corpora.

7. References

7.1. Bibliographical References

Adolf, P. (2006). *Dictionnaire comparatif multilingue: français-allemand-alsacien-anglais*. Midgard, Strasbourg, France.

Agić, Ž., Johannsen, A., Plank, B., Martínez, H. A., Schluter, N., and Søgaard, A. (2016). Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:301–312.

Barre, C. and Vanderschelden, M. (2004). *L'enquête "étude de l'histoire familiale" de 1999 - Résultats détaillés*. INSEE, Paris.

Bernhard, D. and Ligozat, A.-L. (2013). Hassle-free POS-Tagging for the Alsatian Dialects. In Marcos Zampieri et al., editors, *Non-Standard Data Sources in Corpus Based-Research*, ZSM Studien, pages 85–92. Shaker. Volume 5.

Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2001). The Prague dependency treebank: Three-level annotation scenario. In Anne Abeillé, editor, *Treebanks: Building and Using Syntactically Annotated Corpora*. Kluwer Academic Publishers.

Chamberlain, J., Fort, K., Kruschwitz, U., Lafourcade, M., and Poesio, M. (2013). Using games to create language resources: Successes and limitations of the approach. In Iryna Gurevych et al., editors, *The People's Web Meets NLP*, Theory and Applications of Natural Language Processing, pages 3–44. Springer Berlin Heidelberg.

Crévenat-Werner, D. and Zeidler, E. (2008). *Orthographe alsacienne - Bien écrire l'alsacien de Wissembourg à Ferrette*. Jérôme Do Bentzinger.

Denis, P. and Sagot, B. (2012). Coupling an annotated corpus and a lexicon for state-of-the-art pos tagging. *Lang. Resour. Eval.*, 46(4):721–736, December.

Fort, K. and Sagot, B. (2010). Influence of pre-annotation on POS-tagged corpus development. In *Proc. of the ACL Linguistic Annotation Workshop (LAW)*, pages 56–63, Uppsala, Sweden, July.

Fort, K., Adda, G., and Cohen, K. B. (2011). Amazon mechanical turk: Gold mine or coal mine? *Computational Linguistics*, 37(2):413–420.

Fort, K. (2016). *Collaborative Annotation for Reliable Natural Language Processing*. Focus series. ISTE Wiley.

Garrette, D. and Baldridge, J. (2013). Learning a part-of-speech tagger from two hours of annotation. In *Proc. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-13)*, pages 138–147, Atlanta, GA, USA, June.

Geiger, D., Seedorf, S., Schulze, T., Nickerson, R. C., and Schader, M. (2011). Managing the crowd: Towards a taxonomy of crowdsourcing processes. In *Proc. of the Americas Conference on Information Systems (AMCIS)*, Detroit, MI, USA, August.

Guillaume, B., Fort, K., and Lefebvre, N. (2016). Crowdsourcing complex language resources: Playing to annotate dependency syntax. In *Proc. of the 26th International Conference on Computational Linguistics (COLING): Technical Papers*, pages 3041–3052, Osaka, Japan, December.

Hovy, D., Plank, B., and Søgaard, A. (2014). Experiments with crowdsourced re-annotation of a pos tagging data set. In *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL) (Volume 2: Short Papers)*, pages 377–382, Baltimore, MD, USA, June.

Lafourcade, M., Joubert, A., and Le Brun, N. (2015). *Games with a Purpose (GWAPS)*. Wiley-ISTE, July.

Li, S., Graça, J. a. V., and Taskar, B. (2012). Wiki-ly supervised part-of-speech tagging. In *Proc. of the 2012 Joint Conference on Empirical Methods in Natural Language Processing (EMNLP) and Computational Natural Language Learning (CoNLL)*, pages 1389–1398, Jeju Island, Korea, July.

Liberman, M. (2016). Oral histories: Linguistic documentation as social media. In *Novel Incentives and Engineering Unique Workflows (NIEUW)*, pages 38–39, Philadelphia, PA, USA, October.

Malherbe, M. (1983). *Les langages de l'humanité (une encyclopédie des 3000 langues parlées dans le monde)*. Collection Bouquins. Laffont.

McEnery, T. and Hardie, A. (2011). *Corpus Linguistics:*

[17] We reproduce here the classification proposed by Petrov et al. (2012) without validating all their choices: some "closed classes", such as prepositions (ADP), being highly productive.

Method, Theory and Practice. Cambridge Textbooks in Linguistics. Cambridge University Press.

Millour, A. and Fort, K. (2017). Why do we Need Games? Analysis of the Participation on a Crowdsourcing Annotation Platform. In *Games4NLP*, Valencia, Spain, April.

Munro, R. (2013). Crowdsourcing and the crisis-affected community: lessons learned and looking forward from mission 4636. *Journal of Information Retrieval*, 16(2).

Pestian, J. P., Matykiewicz, P., and Linn-Gust, M. (2012). What's in a note: Construction of a suicide note corpus. *Biomedical Informatics Insights*, 5:1–6.

Petrov, S., Das, D., and McDonald, R. (2012). A universal part-of-speech tagset. In *Proc. of Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, May.

Poesio, M., Chamberlain, J., Kruschwitz, U., Robaldo, L., and Ducceschi, L. (2013). Phrase detectives: Utilizing collective intelligence for internet-scale language resource creation. *ACM Trans. Interact. Intell. Syst.*, 3(1):3:1–3:44.

Samardzic, T., Scherrer, Y., and Glaser, E. (2015). Normalising orthographic and dialectal variants for the automatic processing of swiss german. In *Proc. of the 7th Language and Technology Conference*, Poznań, Poland, November.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180, Edmonton, Canada, May.

Urieli, A. (2013). *Robust French syntax analysis: reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. Ph.D. thesis, Toulouse le Mirail-Toulouse II University.

Vergez-Couret, M., Urieli, A., and Foix, F. (2014). Postagging different varieties of occitan with single-dialect resources. In *Proc. of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects (VarDial)*, pages 21–29, Dublin, Ireland, August.

Zennaki, O., Semmar, N., and Besacier, L. (2016). Inducing Multilingual Text Analysis Tools Using Bidirectional Recurrent Neural Networks. In *Proc. of International Conference on Computational Linguistics (COLING)*, Osaka, Japan, December.

7.2. Language Resource References

Alice Millour, Karën Fort. (2017). *Crowdsourced POS Tagged Corpus of Alsatian*.

Crowdsourced Corpus of Sentence Simplification with Core Vocabulary

Akihiro Katsuta, Kazuhide Yamamoto
Nagaoka University of Technology
1603-1, Kamitomioka Nagaoka, Niigata 940-2188, JAPAN
{katsuta,yamamoto}@jnlp.org

Abstract

We present a new Japanese crowdsourced data set of simplified sentences created from more complex ones. Our simplicity standard involves all rewritable words in the simplified sentences being drawn from a core vocabulary of 2,000 words. Our simplified corpus is a collection of complex sentences from Japanese textbooks and reference books together with simplified sentences generated by humans, paired with data on how the complex sentences were paraphrased. The corpus contains a total of 15,000 sentences, in both complex and simple versions. In addition, we investigate the differences in the simplification operations used by each annotator. The aim is to understand whether a crowdsourced complex-simple parallel corpus is an appropriate data source for automated simplification by machine learning. The results, that there was a high level of agreement between the annotators building the data set. So, we believe that this corpus is a good quality data set for machine learning for simplification. We therefore plan to expand the scale of the simplified corpus in the future.

Keywords: Corpus, Crowdsourcing, Simplification

1. Introduction

Simplification task is the process of rewriting a complex text into a simpler form while preserving its meaning. Simplified texts play an important role in providing accessible and easy-to-understand information for a wide range of users who find it difficult to understand texts that have not been simplified due to their linguistic complexity. Attempts have been made to automate the simplification process for various languages, including English, Spanish, Brazilian Portuguese, and Portuguese. We have been conducted research on simplification since a few years ago (Moku et al., 2012). Recent studies have treated text simplification as a monolingual machine translation problem in which simple and synonymous sentences are generated using statistical machine translation (Wubben et al., 2012; Xu et al., 2016).

As with statistical machine translation using bilingual parallel corpora, text simplification therefore requires a monolingual parallel corpus for training. In the case of English, there are PWKP (Zhu et al., 2010) and Wikipedia Datasets (Coster and Kauchak, 2011; Kauchak, 2013) made from wiki and Swiki. In German, a simplification corpus with a scale of 7,000 sentences (Caseli et al., 2009), In Italian the PaCCSS-IT (Brunato et al., 2016) with a scale of 63,000 sentences, In Spanish a simplification corpus do exist made by hand by the few rules (Mitkov and Štajner, 2014; Štajner et al., 2015), and so on.

However, only English corpora are publicly available, such as a large-scale simplified English corpus obtained from pairs of Wikipedia and Simple English Wikipedia entries. For Japanese, there is no other than our simplification corpus (Maruyama and Yamamoto, 2018). We have already conducted the experiment of simplification using corpus (Maruyama and Yamamoto, 2017). Currently, most of the simplification research that has been done for Japanese has emphasized paraphrasing word units (Kajiwara et al., 2013; Kajiwara and Yamamoto, 2015; Hading et al., 2016). For that reason, we consider building a simplified corpus to be the most important task for studying automated simplification of Japanese.

In this paper, we therefore present a human-generated simplified corpus where the simplification operations are based on simple vocabulary restriction rules. Recent research on corpus building has shown that simplification processes based on short lists of simple rules are more time efficient and consistent (Mitkov and Štajner, 2014).

In order to express all the usual things of daily conversation level, we set a frame of total number in advance, and artificially selected what we call core vocabulary. We unify these simplification criteria by defining a core vocabulary of 2,000 words and asking annotators to simplify complex sentences into plainer ones using only these 2,000 words. Our simplification is aimed at vocabulary compression of sentences as we usually use everyday. Therefore, we do not consider other simplification parameters such as difficult grammar and sentence length (E.g. simplification including summarization tasks such as Simple Wikipedia is not covered). Also, since we are interested in the approaches different annotators take to the simplification process, we investigate the differences in the simplification operations used for the simplified corpus.

2. Experimental Design

2.1. Core Vocabulary

We chose 2,000 words, based on the UniDic[1] word units that preserved the most meaning for sentences in the Tanaka corpus[2], and defined these words as the core, or simple Japanese, vocabulary. The Tanaka corpus of Japanese-English parallel corpus that was translated as part of the classwork by the Japanese college students and contains

[1] https://ja.osdn.net/projects/unidic
[2] http://www.edrdg.org/wiki/index.php/Tanaka_Corpus

	Version		Example
(a)	Original		彼女は、あなたが考えているような 女の子 ではない。
	Simplified	(1)	彼女は、あなたが考えているような 女 ではない。
		(2)	彼女は、あなたが考えているような 女性 ではない。
		(3)	彼女は、あなたが考えているような 少女 ではない。
(b)	Original		このスープは ほんのわずか 塩が たりない。
	Simplified	(1)	このスープは 本当に少しだけ 塩が 不足している。
		(2)	このスープは ちょっとだけ 塩が 不十分だった。
		(3)	このスープは 少し 塩が 少ない。
(c)	Original		今日私は道で見つけた キー を 拾い上げた。
	Simplified	(1) - (3)	今日私は道で見つけた 鍵 を 拾った。

Table 1: Example of simplification sentences in the corpus

150 thousand sentences. It is many of the sentence pairs have been derived from textbooks. Therefore, we consider that many the sentences of the daily conversation level are included. And it is also possible to combine simple sentences with English sentences.
This core vocabulary had the following features.

1. It consisted mainly of simple, frequently used words.

2. It also included some words that did not meet the first condition, but were necessary to explain certain concepts. (e.g. red, green, and blue were necessary to explain the concept of color.)

Symbols (such as punctuation marks), unique words (such as proper nouns), and words that were not present in UniDic (such as English words) were excluded from the core vocabulary.

2.2. Simplification Task

We used "CrowdWorks[3]," the crowdsourcing platform in Japan to gather Japanese workers. The purpose of this research is to build a human-generated simplified corpus. The text that we targeted for simplification consisted of the Japanese sentences in the Tanaka corpus. We used crowdsourcing to take complex source sentences containing words outside the core vocabulary, and translate them into simple sentences. We extracted sentences of between 7 and 65 words from the Tanaka corpus and simplified them using crowdsourcing. A total of 34,300 sentences were divided into seven parts, and these parts were then assigned to different annotators for simplification (4,900 sentences each). For evaluation, we additionally asked them to simplify the same 100 sentences. The rules we requested the annotators to use for simplification were as follows.

1. Translate each complex sentence using only the core vocabulary, preserving as much of its meaning as possible.

2. Do not paraphrase words that have been excluded from the core Vocabulary (see Section 2.1.), or the target word when explaining the meaning of a word in a sentence.

[3]https://crowdworks.jp/

	Original	Simplified
#Sentences	(4,900)34,300	(4,900)34,300
Vocabulary size	(5,846)14,429	(2,084) 3,649
Avg. sentence length	19.18	21.16

Table 2: Statistics for the simplified corpus (In parentheses are the average for each annotator)

	Original	Simplified
#Sentences	100	(100) 700
Vocabulary size	397	(398) 681
Avg. sentence length	16.56	19.41

Table 3: Statistics for the evaluation simplified corpus (In parentheses are the average for each annotator)

Ideally, machine translation systems should be trained on a corpus built by a small number of annotators, to prevent annotation variability. We therefore asked the annotators to translate most of the sentences by themselves.

3. Simplification Corpus

Major phenomena of simplification in the evaluation corpus is shown in Table 1. The same optimal word may not always be substituted, because the annotator may not know all 2,000 words and there are multiple ways of expressing many concepts. Despite having asked the annotators to make substitutions using the core words, there were instances where, for example, "女の子 (girl)" was replaced by "女 (female)" or "女性 (woman)" rather than "少女 (girl)", as shown in (a) of the table, it is observed that the optimal word is not always chosen for substitution, because the annotator may not know it is in the core words. In example (b), all the annotators paraphrased the sentences to preserve its meaning, but they used different words. Example (c) shows an instance where all three annotators simplified a sentence in the same way.
Overall, 34,300 complex sentences using a vocabulary of 14,429 words, were selected for simplification. After simplification, the new corpus used a vocabulary of only 3,649 words (in Table 2). So, there are 1,649 words excluded from substitution in the 34,300 sentences. And, when 7

BLEU	Frequency
[0.0 , 0.2)	0
[0.2 , 0,3)	4
[0.3 , 0.4)	2
[0.4 , 0.5)	8
[0.5 , 0.6)	7
[0.6 , 1.0]	0

Table 4: Frequency of inter-agreement annotator

at annotators each simplified 100 sentences, the number of vocabulary increased from 397 to 681 (in Table 3). We consider that the reason for the increase is due to the difference of simplified such as (a) or (b) in Table 1.
The length of the simplified sentence is longer in both Table 2 and 3, so we can see that sentences tend to be longer if they paraphrase without losing their meaning. For example in the following cases:

Original:
“そこに署名してください。”
Simplified:
“そこに名前を書いてください。”
“そこにあなたの名前を書いてください。”

Original:
“彼はこのビジネスで名声を築いた。”
Simplified:
“彼はこの仕事で有名になった”
“彼はこの仕事で社会的に評価され尊敬を集める立場を作った。”

Although the latter is closer meaning, we can see that a sentence is longer.
In the next section, we investigate the 100 sentences of evaluation corpus that were given to all annotators.

3.1. Agreement between Annotators

3.1.1. Automated Evaluation

We assessed the level of agreement between annotators to analyze annotation variability. We computed this in terms of the BLEU score in the same way as (Mitkov and Štajner, 2014). This looks at the level of agreement between the simplifications, with higher values indicating more similar results. Because annotator is 7 people, BLEU between 21 patterns annotator was calculated. Table 4 shows the level of inter-annotator agreement in terms of the BLEU scores. Most of BLEU is distributed between 0.4 and 0.6, and there are a few around 0.3.
Mitkov and Štajner (2014) built a simplified corpus based on simple rules, finding that the BLEU scores of their three annotators were between 0.44 and 0.53. The BLEU of our simplified corpus (Maruyama and Yamamoto, 2018) range between 0.58 to 0.63. Since the corpus we created is more controllable, it is highly agreement. In crowdsourcing, since difference how much preserve meaning for each annotator, the case that BLEU become low occurs. However, it is considered that the quality of simplification corpus is high because about 70% of inter-agreement exceeds 0.4.

3.1.2. Manual Evaluation

We also analyzed the quality of the simplified sentences and the simplification operations applied by each annotator. We used the frequencies with which the simplification operations were selected and the qualities of the resulting sentences as indicators of agreement between annotators.
First, we analyzed the simplification operations applied by the annotators to each of the 100 common sentences. We did not count the number of operations applied in a sentence, but just whether or not it was applied at least once. The simplification operations were mainly classified into the following four types:

1. Word substitution (WS)
   A word is paraphrased by a synonym.
   Original:
   “彼から ずいぶん 長い間 便り がない。”
   Simplified:
   “彼から かなり 長い間 手紙 がない。”

2. Phrase substitution (PS)
   Two or more consecutive words are paraphrased.
   Original:
   “私たちのところに、不意の来客 が**あった**。”
   Simplified:
   “私たちのところに、突然訪ねてきた客 が**いた**。”

3. Deletion (D)
   A word is removed from the original sentence.
   Original:
   “彼は私の肩を いっぱつ 打った。”
   Simplified:
   “彼は私の肩を打った。”

4. Insertion (I)
   A word is added to the original sentence.
   Original:
   “冬休みはどのように過ごしましたか。”
   Simplified:
   “冬 の 休みはどのように過ごしましたか。”

Second, we opted to manually evaluate the quality of each sentences, in addition to the above analysis. Following the criteria in Table 5, we asked human evaluators to assess, on a scale of 1-4 (where higher marks denote better sentences), two aspects of the presented sentences: grammaticality (G) and meaning preservation (M).
The resulting numbers of simplification operations, together with the average evaluations from five annotators, are shown for each annotator in Table 6.
The total number of simplification operations for each annotator is not 100, because in some cases the annotators performed several different operations on a single sentence. The most common operation we observed was replacing a word/phrase in the original sentences with a word/phrase from the core vocabulary. In addition, the grammar quality and level of meaning preservation were both high overall, and exhibited little variation between annotators, so the simplification process can be regarded as both consistent and reliable.

Grammar (G)	
Evaluation	Criterion
1	The sentence does not make any sense at all.
2	The sentence is hard to understand due to grammatical mistake.
3	The sentence is fairly good except for minor mistakes.
4	The sentence is free from grammatical mistakes.
Meaning (M)	
Evaluation	Criterion
1	The meaning is unrelated to that of the original sentence.
2	The meaning is related, but the original sentence cannot be guessed.
3	The meaning of the sentence is roughly the same, but it is a little ambiguous.
4	The sentences have the same meaning.

Table 5: Criteria for evaluating the simplified sentences

	WS	PS	D	I	G	M
(1)	60	52	3	5	3.75	3.57
(2)	74	41	2	5	3.87	3.65
(3)	68	39	11	5	3.92	3.57

Table 6: Simplification operation frequencies and human evaluation results (the "G" and "M" columns show the mean grammaticality, and meaning preservation scores, respectively)

4. Conclusions and Future Work

In this paper, we have created a simplified corpus for automated simplification research by crowdsourcing. This is a large simplification corpus, suitable for machine learning, that was produced by a small number of annotators and which shows low annotation variability. Examining the level of agreement between annotators, we found almost high agreement in either the manual or automated evaluations, and therefore regard, this corpus as consistent and reliable. In addition, we found that limiting the core vocabulary to 2,000 words was advantageous for controlling annotation variability. This work also demonstrates that a high-quality simplification corpus can easily be built by crowdsourcing.

In the future, we aim to expand this simplification corpus and use it to create an automated simplification system by machine learning. This corpus will be released.

Acknowledgments

This work was supported in part by JSPS KAKENHI Grants-in-Aid for Challenging Research (Exploratory) Grant ID 17K18481.

References

Brunato, D., Cimino, A., Dell'Orletta, F., and Venturi, G. (2016). Paccss-it: A parallel corpus of complex-simple sentences for automatic text simplification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 351–361.

Caseli, H. M., Pereira, T. F., Specia, L., Pardo, T. A., Gasperin, C., and Aluísio, S. M. (2009). Building a brazilian portuguese parallel corpus of original and simplified texts. *Advances in Computational Linguistics, Research in Computer Science*, 41:59–70.

Coster, W. and Kauchak, D. (2011). Simple english wikipedia: a new text simplification task. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 665–669. Association for Computational Linguistics.

Hading, M., Matsumoto, Y., and Sakamoto, M. (2016). Japanese lexical simplification for non-native speakers. *NLPTEA 2016*, page 92.

Kajiwara, T. and Yamamoto, K. (2015). Evaluation dataset and system for japanese lexical simplification. In *Proceedings of the ACL-IJCNLP 2015 Student Research Workshop*, pages 35–40.

Kajiwara, T., Matsumoto, H., and Yamamoto, K. (2013). Selecting proper lexical paraphrase for children. In *Proceedings of The 25th Conference on Computational Linguistics and Speech Processing*, pages 59–73.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st annual meeting of the association for computational linguistics (volume 1: Long papers)*, volume 1, pages 1537–1546.

Maruyama, T. and Yamamoto, K. (2017). Sentence simplification with core vocabulary. *Proceedings of the International Conference on Asian Language Processing*, pages 363–366.

Maruyama, T. and Yamamoto, K. (2018). Simplified corpus with core vocabulary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'18)*.

Mitkov, R. and Štajner, S. (2014). The fewer, the better? a contrastive study about ways to simplify. In *Proceedings of the Workshop on Automatic Text Simplification-Methods and Applications in the Multilingual Society (ATS-MA 2014)*, pages 30–40.

Moku, M., Yamamoto, K., and Makabi, A. (2012). Automatic easy japanese translation for information accessibility of foreigners. In *Proceedings of Coling-2012 Workshop on Speech and Language Processing Tools in Education*, pages 85–90.

Štajner, S., Calixto, I., and Saggion, H. (2015). Automatic text simplification for spanish: comparative evaluation of various simplification strategies. In *Proceedings of the international conference recent advances in natural language processing*, pages 618–626.

Wubben, S., Van Den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of*

the Association for Computational Linguistics: Long Papers-Volume 1, pages 1015–1024. Association for Computational Linguistics.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Zhu, Z., Bernhard, D., and Gurevych, I. (2010). A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd international conference on computational linguistics*, pages 1353–1361. Association for Computational Linguistics.

Simplification corpus	
Original	ナンシーは最初に来た 女の子 だった。
	(Nancy was the first girl to come.)
Simplified	ナンシーは最初に来た少女だった
Original	追って 通知 があるまで 会合 は 延期 された。
	(The meeting was put off till further notice.)
Simplified	追って連絡があるまで会の予定は遅れることになった。
Original	ドアが パッ と 勢い 良く あい た。
	(The door burst open.)
Simplified	ドアが突然開いた。
Original	この 騒音 は我慢出来ない。
	(I cannot put up with this noise.)
Simplified	このようにうるさい所にはいられない。
Original	いやぁ、昨日は 入れ食い でねえ。
	(They were biting like crazy yesterday.)
Simplified	いや、昨日はたくさん魚が手に入ってねえ。
Original	例の スキャンダル はそういつまでも 臭い ものに フタ と いう わけにはいくまい。 いずれ 人は 嗅ぎつけ てしまうさ。
	(I don't think we can keep the lid on the scandal much longer; people are bound to find out.)
Simplified	例のあまり良くない話はそういつまでも無かったことにはできまい。 やがて人は気づいてしまうさ。
Evaluation simplifie corpus	
Original	明日の午後、いつでもお 出で ください。
	(Come to see me at any time tomorrow afternoon.)
Simplified	明日の午後、いつでも来てください。 (7)
Original	近頃 は いかが ですか。
	(How are you these days?)
Simplified	最近はどうですか。(4) 調子はどうですか。 最近は、調子はどうですか。 最近はどのようにお過ごしですか。
Original	彼はこの ビジネス で 名声 を 築い た。
	(He worked up a good reputation through this business.)
Simplified	彼はこの仕事で有名になった。(3) 彼はこの仕事で名を上げた。 彼はこの事業で有名になった。 彼はこの仕事で高い評価を手にした。 彼はこの仕事で社会的に評価され尊敬を集める立場を作った。
Original	彼から ずいぶん 長い間 便り がない。
	(I haven't heard from him for ages.)
Simplified	彼からかなり長い間手紙がない。 彼からかなり長い間手紙が来ない。 彼からかなり長い間連絡がない。 彼から長い間連絡がない。 彼からとても長い間連絡がない。 彼らからは長い間手紙が来ていない。 彼から非常に長い間、手紙が来ない。
Original	彼らは 雇い主 に 忠実 だ。
	(They are loyal to their master.)
Simplified	彼らは使用人に誠実によく働く人だ。 彼らは主人を信頼して従っている。 彼らは上司に従う。 彼らは働いているところの主人によく従っている。 彼らは使用者の言うことによく従う。 彼らは社長の命令をよく聞く。 彼らは経営者に大事にされている。

Table 7: Example of Simplification Corpus. The underline words in the original sentences are complex words. The numbers are the number of same simplified sentences.

A Multilingual Wikified Data Set of Educational Material

Iris Hendrickx[1], Eirini Takoulidou[3], Thanasis Naskos[3], Katia Lida Kermanidis[3], Vilelmini Sosoni[2], Hugo de Vos[1], Maria Stasimioti[2], Menno van Zaanen[4], Panayota Georgakopoulou[6], Markus Egg[5], Valia Kordoni[5], Maja Popovic[5], Antal van den Bosch[1]

[1]Centre for Language Studies, Radboud University,Nijmegen, Netherlands

[1]Department of Foreign Languages, Translation and [2]Interpreting, Ionian University, Corfu, Greece

[3]Department of Informatics, Ionian University, Corfu, Greece

[4]Department of Communication and Information Sciences, Tilburg University, The Netherlands

[5]Department of English Studies, Humboldt−Universität zu Berlin, Germany

[6]Deluxe Entertainment Services Group, Athens, Greece

i.hendrickx@let.ru.nl;rinoulit@gmail.com;anaskos@ionio.gr; vilelmini@hotmail.com; kerman@ionio.gr; stasimioti.maria@gmail.com; mvzaanen@uvt.nl; yota.georgakopoulou@bydeluxe.com; kordonie@anglistik.hu-berlin.de; maja.popovic@anglistik.hu-berlin.de; markus.egg@anglistik.hu-berlin.de;a.vandenbosch@let.ru.nl

Abstract

We present a parallel wikified data set of parallel texts in eleven language pairs from the educational domain. English sentences are lined up to sentences in eleven other languages (BG, CS, DE, EL, HR, IT, NL, PL, PT, RU, ZH) where names and noun phrases (entities) are manually annotated and linked to their respective Wikipedia pages. For every linked entity in English, the corresponding term or phrase in the target language is also marked and linked to its Wikipedia page in that language. The annotation process was performed via crowdsourcing. In this paper we present the task, annotation process, the encountered difficulties with crowdsourcing for complex annotation, and the data set in more detail. We demonstrate the usage of the data set for Wikification evaluation. This data set is valuable as it constitutes a rich resource consisting of annotated data of English text linked to translations in eleven languages including several languages such as Bulgarian and Greek for which not many LT resources are available.

Keywords: MOOCs, crowdsourcing, wikification

1. Introduction

TraMOOC (Translation for Massive Open Online Courses) is an EU-funded project that aims to improve access to educational material in MOOCs by providing dedicated machine translation (MT) solutions. The educational content, including video lecture subtitles, forum posts, and quiz questions and answers, is translated from English into eleven European and BRIC languages: Bulgarian, Chinese, Croatian, Czech, Dutch, German, Greek, Italian, Polish, Portuguese, and Russian.

Here, we focus on the collection of parallel texts in eleven language pairs that are annotated and linked with respect to the entities occurring in the text. We consider names and noun phrases that contain the topical information about a text as entities. To identify such entities, information from Wikipedia is used. Each entity in the text is linked to its corresponding Wikipedia page in the respective language.

The fine-grained information encapsulated in this data set can be used for a wide range of applications. We use the data set for an in-depth analysis and evaluation of machine translation output. In addition to word-based evaluations (e.g., BLEU), a semantic evaluation can be performed, as the links to the Wikipedia pages in the various target languages provide an additional source of information. Such implicit MT evaluation aims to judge the MT quality between source and target language without using an explicit (manual) translation step. Alternatively, the data set is suited for other multilingual tasks that focus on semantic aspects such as the cross-lingual Semantic Textual Similarity task (Agirre et al., 2016; Cer et al., 2017). Furthermore, the data set can be used as multilingual training material for the development of novel wikification tools (e.g., Tsai and Roth (2016)), or tools that automatically detect and link topics in a text to their respective Wikipedia pages.

In the remainder of this paper we discuss related work in Section 2., the creation of this data set via crowdsourcing (Section 3.), the difficulties that we encountered using crowdsourcing for such complex annotation task (Section 4.) and the outcomes of this process in Section 5.. We also briefly discuss the use case of implicit translation evaluation for which we created the data set in Section 6. and conclude in Section 7..

2. Related Work

Comparable and related types of data sets are those created for the evaluation of wikification tools (Mihalcea and Csomai, 2007). The Illinois Wikifier was evaluated on English material annotated with Wikipedia links (Ratinov et al., 2011). This evaluation set consists of Wikipedia pages and news articles. In the context of automatic word sense disambiguation, there is another related multilingual data set from Semeval-2015 task 13 (Moro and Navigli, 2015) containing links to BabelNet (Navigli and Ponzetto, 2012) and Wikipedia pages with news articles in three languages (of which only Italian matches the languages targeted in the TraMOOC project).

These data sets do not cover all the languages we are interested in and, additionally, do not target the educational domain. Therefore, a new data set needed to be created. This data set is intended as both tuning material for the developed implicit machine translation system evaluation tool and for testing the final machine translation systems. The data set also gives us insights into the coverage of

high		low	
EN	5546	CS	397
DE	2140	BG	238
NL	1919	HR	181
IT	1407	EL	141
RU	1445		
PL	1259		
PT	987		
ZH	985		

Table 1: Number of Wikipedia articles per language in K articles (stats from 08-01-2018)

Wikipedia of the eleven project languages for topics in the educational domain, and thus testing the limits of this implicit MT evaluation method. We can only measure translation success for those topics for which an equivalent Wikipedia page exists in the target language. Tables 1 shows the number of available pages for each of the languages of the TraMOOC project. The English Wikipedia is by far the largest, consisting of more than 5,5 million articles; it is likely that many detected source topics do not have a corresponding topic in the target language. Some of our target languages have many Wikipedia pages such as Dutch and German who have at least a million articles for each language, while Greek and Croatian are supported by a much smaller number of Wikipedia pages, and we can expect this method to be less effective for them.

Due to the specialized nature of this task and its multilingual aspects, we chose to use crowdsourcing to collect the data. Even though using a non-expert crowd may perhaps lead to lower quality results, crowdsourcing has the main benefit of access to people speaking the different languages.

3. Data Selection

Our aim was to collect wikification annotations for 500 to 1,000 sentences from parallel educational texts for each language pair. We used three existing parallel text resources as the basis for the sentence selection so as to cover a broad range of online courses and to cover all eleven language pairs. In particular, we use parallel texts from course material of the Coursera MOOC platform, the Iversity MOOC platform, and the QCRI Educational Domain (QED) Corpus (formerly known as QCRI AMARA Corpus) (Abdelali et al., 2014). The Iversity data consists of (i) manually translated MOOC data, and (ii) MT output of English MOOC data produced by the first MT software prototype that was developed in the first year of the TraMOOC project. Both Coursera and QED material consist of MOOC subtitles that were translated using crowdsourcing in other projects unrelated to TraMOOC. The QED corpus consists of a large collection of files and each file contains a number of subtitles from MOOC video lectures in a particular language. The aligned files (parallel corpus) share the same first part of the file name. However, not all files with the same name contain the same amount of text. Hence, a filtering process was applied to select only those files where both the English file and the target language file contain the same number of sentences.

A mix of the data collected from the three parallel corpora was used to cover the number of sentences needed for all eleven target languages. Hence, in most of the cases both QED and Coursera material is used (with the exception of Croatian where only a small set of sentences in the Coursera corpus was available). Iversity MOOC material was available for three language pairs, i.e., EN-EL, EN-IT, EN-PT; this was combined with a sample from the Coursera corpus to reach the number of sentences required.

We deliberately chose to select sentences from different resources as each resource has its own peculiarities. For example, the QED texts include partially aligned sentences or are aligned at the clause level, while the Coursera data often aligns multiple sentences to multiple sentences. The quality of the translation and of the sentence alignment also varies per language and resource. Some sentences are badly aligned and the aligned parts do not contain the same information. This is particularly problematic in the context of multilingual wikification.

The actual text snippets used in the annotation collection task were manually selected. They consist of around 20–50 consecutive sentences each. Special care was taken to make sure that the sentences in the data set did not contain repetitions (e.g., the sentence "welcome to this lecture" is commonly used in all courses) and did not contain speaker interjections in textual representation of speech (such as "mmm", [NOISE], or [MUSIC]). Also, particularly long paragraphs were not selected, so as to maintain the microtasking nature of the activity.

The selected source and target sentences were automatically tokenized using the multilingual tokenizer Ucto[1] (van Gompel et al., 2017). Ucto has language-specific rules for the tokenization of several languages including Dutch, English, German, Italian, Portuguese, and Russian. For the other languages, generic language-independent settings of Ucto were used except for the Chinese language where we applied the Stanford Word Segmenter (Chang et al., 2008).

4. Annotation via Crowdsourcing

Crowdsourcing has been used extensively for annotating corpora due to being a cheap and fast means to collecting human intelligence input, compared to requesting expert-based intervention (Wang et al., 2013). Crowdsourcing approaches vary from voluntary work and gaming to paid microtasks (Bougrine et al., 2017). Applications involve, but are not limited to, the annotation of speech corpora (Bougrine et al., 2017; Su et al., 2017), of named entities (Bontcheva et al., 2017), of domain-specific concepts (Good et al., 2014) of relation extraction (Liu et al., 2016). Researchers have shown particular interest in issues pertaining to ethical implications (Cohen et al., 2016), as well as best practices for obtaining optimal quality output (Sabou et al., 2014). Best practice guidelines are followed in the present work also, after experimentation with varying

[1]Ucto is freely available at `https://languagemachines.github.io/ucto/`.

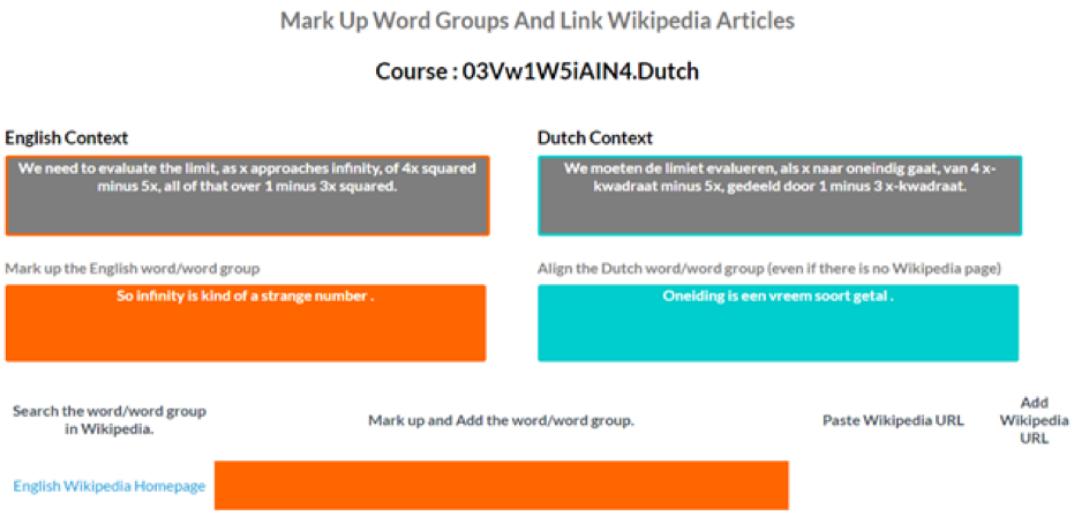


Figure 1: CrowdFlower interface for manual annotation.

parameterization schemata, and involve task decomposition into simple microtasks, the appropriate crowd choice, the appropriate crowd reward choice, data preparation, task design, task completion time, quality control, task monitoring and crowd evaluation.

Given pairs of aligned texts, the entity annotation (wikification) task consists of identifying and annotating names and noun phrases in the texts that can be linked to their corresponding Wikipedia pages with the same meaning.

To collect this information, annotators (called contributors in the context of crowdsourcing) are presented with an English sentence and its translation. Such a sentence pair is displayed along with its context (which consists of sentences of the entire paragraph in both English and the target language, as well as the related course title). Contributors are then instructed to analyze the sentences to identify names and nouns that are found in Wikipedia. If such an entity is identified, contributors mark them up in both the English and translated sentences. Annotating an item consists of highlighting the words describing the entity as well as providing a URL to the corresponding Wikipedia page in the correct language. It may be the case that an entity can be found in the English Wikipedia, but not in the Wikipedia of the target language. In this case, the phrase is still identified, but a null value is assigned for the Wikipedia link in the target language.

As this activity differs greatly from regular crowdsourcing tasks, detailed instructions with examples are provided to the contributors. Contributors were asked to mark up only those entities that had an Wikipedia page for that entity with the same meaning. In case multiple possible entities could be marked, contributors should mark the longest possible phrase.

(1) Agrippa 's Trilemma states that there are three options if we try to prove the truth .

In example 1 both [Agrippa] and [Trilemma] have their own separate Wikipedia page, but we annotate the longest word group [Agrippa's Trilemma] that points, via redirection, to the Wikipedia page entitled 'Münchhausen trilemma'. Wikipedia uses redirection links to link synonyms to the same page. In this example only two entities are to be marked, [Agrippa's Trilemma] and [truth]. The noun phrase 'options' does not express a clear concept for which a unique Wikipedia page exists and should not be marked. Such annotation decisions are sometimes difficult to make and partly subjective. For that reason each sentence is annotated by three different contributors to only keep those annotations where at least two of the three contributors agreed upon.

A trial instance ('test question') is provided to the contributor, which has gold standard annotations against which the contributor's annotation is checked. The same approach is used to validate the accuracy of the contributors' work and assure the quality of the data collected during the task. We used these test questions to monitor the trustworthiness of the contributors and flag malicious users as untrusted. These test questions were manually translated and annotated by professional translators. Around 30 test questions per language pair were prepared.

For the collection of the annotations, the CrowdFlower (CF) platform[2] was used. This choice was driven by several practical concerns, such as payment options, configurability, quality control mechanisms as well as the size of the contributor crowd. Especially the configurability of the CrowdFlower platform had a major impact on the choice of platform, as the wikification annotation task is more complex than most crowdsourcing annotation tasks.

We show a picture of the CF interface in Figure 1. The English sentence and corresponding sentence in the target language are shown next to each other in the CF interface and some preceding sentences are shown in the gray box above each sentence. Samples of consecutive sentences were presented to contributors in order to help them pick the correct Wikipedia pages for topics that have multiple meanings and, thus, multiple candidate Wikipedia links. When sentences are presented in context, this context aids the disambiguation process. Furthermore, by using several consecutive sentences, the contributors could work more efficiently

[2]CrowdFlower: `https://www.crowdflower.com`.

when encountering a topic that was already mentioned in the previous sentence.

The example sentence shown in Figure 1 was taken from the QED corpus. The short English sentence to be annotated, 'So infinity is kind of a strange number', is more easily comprehensible as part of a mathematical context when reading the preceding sentence 'We need to evaluate the limit, as x approaches infinity, of 4x squared minus 5x, all of that over 1 minus 3x squared.'. Both the English term 'number' and its Dutch translation 'getal' are ambiguous terms that can potentially be linked to multiple candidate Wikipedia pages but in this example only the mathematical reading is appropriate (`https://en.wikipedia.org/wiki/Number` and `https://nl.wikipedia.org/wiki/Getal_(wiskunde)`), and the same holds for the concept 'infinity' (linked to `https://en.wikipedia.org/wiki/Infinity`), 'oneindig' in Dutch, `https://nl.wikipedia.org/wiki/Oneindigheid`).

The annotation task was launched with an initial set of a 1,000 sentences for each language pair. In total, funding was available for 3,000 annotations of sentence pairs per language. As the contributors were also paid when annotating the test questions, which make up 30% of the total annotations, we could maximally expect to gather 700 items per language pair as we planned to use a redundancy of three in this task for quality purposes.

CrowdFlower allows for configuration settings that restrict the contributors according to several measures. Initially, we only allowed contributors that are residents of the countries in which the language was spoken. Unfortunately, the annotation process progressed very slowly for most of the languages and for some languages hardly any annotations were collected. Not only the availability of the contributors (limited by the restriction), but also the complexity of the task severely limited data annotation. The country limitation was therefore relaxed for all languages except PT for which we did have sufficient contributors. For Chinese specifically, we could not gather sufficient contributors via the CrowdFlower platform, and for this data set we asked a professional translator to annotate 200 sentences.

Lang pair		English			Target language		
	#s	slen	#t	#t/#s	slen	#t	#t/#s
BG	618	14.9	765	1.2	13.5	747	1.2
CS	688	14.4	712	1.2	11.7	702	1.2
DE	785	15.3	911	1.2	14.5	900	1.1
EL	627	23.0	923	1.2	22.9	912	1.3
HR	597	19.1	712	1.2	15.4	703	1.2
IT	782	22.3	1494	1.2	21.9	1477	1.3
NL	689	16.9	831	1.2	15.8	815	1.1
PL	569	16.2	653	1.3	13.5	638	1.3
PT	696	21.0	1289	1.3	20.2	1275	1.3
RU	837	17.4	1117	1.2	14.8	1088	1.2
ZH	604	19.6	856	1.2	14.0	759	1.1

Table 2: Data set statistics listing the number of sentence pairs (#s), the number of identified topics for both English and the target language (#t), average sentence length in tokens (slen), and average number of annotated topics per sentence (#t/#s).

5. Outcomes

At the end of the annotation process, we collected all annotations from the CrowdFlower platform and filtered the sentences to keep only those for which we had at least three trustworthy contributors and kept all annotations made by at least two annotators. This resulted in a data set in which for each language pair at least 500 sentences were annotated, as is shown in the second column in Table 2. When we compare the number of topics annotated in English (column 4 of Table 2) and in its corresponding topics in the target language (column 7), we can see that only for a handful of topics no equivalent phrase was available in the translation but in the vast majority the corresponding topic was marked. The average number of annotated topics per sentence is rather low, between 1.1 and 1.3 on average for all language pairs.

We also performed a manual inspection of the filtered trustworthy annotations to verify whether they are indeed correct, and to what extent they are complete. We took a small sample of 20 sentences for each language pair and counted the number of topics in English with their corresponding Wikipedia links. This analysis showed that annotations gathered via crowdsourcing were not complete; many topics that had a corresponding English Wikipedia page were not marked. For some languages even 50% of the topics that could have been marked were missing in this small sample. There are two main reasons for this. On the one hand, the annotators were not all consistent in their annotations and an entity would be marked by one annotators but not by all. As we only take into account those annotations made by at least two annotators to guard quality, we are losing these annotations. Also, many annotators chose to mark a minimum of entities instead of marking all options. However, this manual inspection showed us that the topics that were marked, were correctly marked, linked and also correctly linked to the topic in the target language (in this small sampling 90–100% correct). In terms of precision, we achieved a satisfying result.

The resulting data set will be made available through the EU (according to the H2020 Open Research Data Pilot) for research purposes after the end of the project, excluding the sentences sampled from Coursera as these were restricted by copyright to project-internal usage only. Therefore the publicly available data set contains around 40% less sentence pairs than the full data set shown in Table 2. The numbers of sentences in the publicly available dataset are shown in the second column in Table 3.

Table 3 represents the upper bound of Wikipedia coverage score by checking how many of the English manual labels had a corresponding Wikipedia page in the target language. This shows how many of the cases received a null value in the annotation process as discussed above. As expected, both Greek and Croatian have a low upper bound of around 60% indicating that due to the small number of Wikipedia pages for these languages, linking translations

lang	#s	upper bound
BG	461	76.23
HR	27	68.29
CS	529	82.31
DE	490	91.64
EL	467	62.20
IT	503	89.36
NL	482	87.10
PL	442	79.74
PT	529	92.95
RU	487	88.61
ZH	217	80.62

Table 3: The percentage of entities in English sentences that actually have an equivalent Wikipedia page in the target language.

lang	#s	precision
BG	461	56.21
HR	27	83.33
CS	529	57.63
DE	490	54.55
EL	467	57.14
IT	503	61.41
NL	482	52.17
PL	442	45.45
PT	529	64.41
RU	487	52.94
ZH	217	68.00

Table 4: Comparison between manual labels and labels produced by English Illinois Wikifier.

via Wikipedia is less informative for these languages than for the languages with better coverage.

6. Usage

We developed the dataset presented in this paper for the purpose of creating and tuning a method for implicit translation evaluation. We make use of the fact that most Wikipedia pages have translations in many other languages. We apply the following method. The input for evaluation is an English source sentence and its translation in the target language. We apply a Wikifier to find and link the entities in the English source data to their relevant Wikipedia pages. We check whether any entities identified in the source text have corresponding Wikipedia pages in the target languages using the inter-language links present in Wikipedia. Next we verify whether the target sentence indeed contain this corresponding translation, checking both for synonyms and lemmatized versions of the entity. When such a match is found, we count this as a correct entity translation, or as an error when no match was found.

The full details on the experiments for implicit evaluation are beyond the scope of this paper. To demonstrate the usability of the created data set we show the results of applying a state-of-the-art publicly available Wikifier on the English sentences in the data set. We apply the Illinois Wikifier (Cheng and Roth, 2013) that uses both global and local context cues (Ratinov et al., 2011) to disambiguate ambiguous names and concepts that have more than one possible Wikipedia page. The Wikifier first detects names and concepts in the text and then aims to link these entities to Wikipedia pages. In Table 4 we show a comparison between the manually labeled Wikipedia links and the links assigned by the Wikifier. The first column shows the number of sentences in the data set. We computed how many of the predicted Wikipedia links were correct (precision) shown in the second column. We observe that between 45 and 68% of the manual labeled entities was also marked by the Wikifier (the HR data set achieves 83% but it only contains 27 sentences). This is considerably lower than the outcome scores reported by (Cheng and Roth, 2013). When inspecting the mismatches between the Wikifier and the data set it becomes clear that the Wikifier, which is tuned on news articles, focuses mostly on named entities which are less present in the educational material. Typical educational entities like 'number' and 'infinity' from the example sentence in figure 1 are not recognized as entities by the Wikifier.

7. Conclusion and Discussion

In this article, we describe the process of creating a wikification data set. This dataset consists of pairs of sentences for eleven language pairs that are manually annotated with names and noun phrases. Each annotated item is linked to its corresponding Wikipedia page in the language of the sentence, if such a page exists. For the annotation process, we used a popular crowdsourcing platform to find contributors from the countries where these languages were spoken. The produced annotations are precise and correct, but lack in coverage (recall). We gave a brief example of how the data set can be used for the evaluation of translation or the evaluation of automatic wikification.

Malicious behavior in crowdsourcing platforms is a common phenomenon. Contributors are mainly interested in quick payment, making the least possible effort. The erroneously marked entities indicated that contributors paid little attention to the detailed instructions, which should be expected in a complex task, such as entity annotation. Furthermore, after a number of submissions, contributors get trained on the task process, and often adopt malicious behavior in order to cheat the system, despite the fact that we put in 30% test questions in the crowdsourcing setup. In a characteristic example, in a sentence with multiple entities, even decent contributors annotated only one, just to receive the reward easily and quickly.

The wikification annotation task is more complex than traditional crowdsourcing annotation tasks. Annotation of items in sentences should only be done if corresponding Wikipedia pages can be found, which means that an external source of information (Wikipedia) is needed to annotate the sentences. Additionally, after marking the items, the URL to the Wikipedia page should also be given. The same has to happen in the sentence in the other language of the language pair. This process consists of several complex

actions.

Given the complexity of the task, it is hard to find contributors who are interested in performing the task. Because the contributors also need to be fluent in the two languages of the language pair, this resulted in very limited availability of contributors for this task. Even when the initial country restriction was dropped, this still resulted in a small crowd for the task.

Ideally, the crowdsourcing platform opens a crowdsourcing job up to a large crowd all over the world. In practice, however, we did not easily get sufficient contributors for this specific task. We expect this is partly due to the complexity of the task in connection to the payment rate: we ended up with annotations by contributors from several low-income economies like Philippines, Venezuela, and Indonesia. For future entity annotation tasks, it would be worth considering using another crowdsourcing platform (e.g., MTurk) to target an a-priori competent pool of contributors, taking advantage of the platform features (e.g., qualified contributors with a specific success rate threshold in related past tasks).

We did consider reducing the complexity of the task by splitting the entire annotation process in several smaller steps, such as identification in English, identification in the other language, and linking to Wikipedia in both languages. Such approach is suggested by Kittur and colleagues (Kittur et al., 2011) who showed that a complex task can be broken down into smaller tasks. Such an alternative approach had several disadvantages. First, we would have to create multiple interfaces for the different types of tasks and setting up crowdsourcing jobs for each of the steps. After annotation, the annotations in the two languages would also need to be linked to each other. In the end, such a step-wise approach would have been much more costly and time consuming, while at the same time it is unlikely that it would increase annotation throughput. The most time consuming part of this annotation task was the step to look up each potential name and candidate in Wikipedia. If we were to split the task in multiple steps, this time consuming part would have to be performed multiple times.

While crowdsourcing is an excellent means of collecting annotated data for several purposes, it seems to be less suitable for complex tasks as the one described here. The task needs to be clear, straightforward, and annotation should not consist of several actions. Having strict requirements on the people in the crowd (such as country or language restrictions) severely limits the availability of the crowd, which in turn reduces the annotation speed.

From a financial point of view, crowdsourcing, even for complex tasks, allows for the collection of annotations that are otherwise simply not feasible. In terms of value for money, crowdsourcing leads to results that we could not have achieved in any other way. When attempting crowdsourcing data collection, care has to be taken on how to setup the tasks to maximise the use of the crowd, as well as find automated ways to ensure continuous monitoring of the crowd contributors' quality to ensure the successful completion of the task.

Acknowledgements

This work was completed as part of the TraMOOC project (Translation for Massive Open Online Courses) funded by the European Commission under H2020-ICT-2014/H2020-ICT-2014-1 under grant agreement number 644333.

References

Abdelali, A., Guzman, F., Sajjad, H., and Vogel, S. (2014). The AMARA Corpus: Building Parallel Language Resources for the Educational Domain. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Agirre, E., Banea, C., Cer, D. M., Diab, M. T., Gonzalez-Agirre, A., Mihalcea, R., Rigau, G., and Wiebe, J. (2016). Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of SemEval@ NAACL-HLT*, pages 497–511.

Bontcheva, K., Derczynski, L., and Roberts, I. (2017). Crowdsourcing named entity recognition and entity linking corpora. In *Handbook of Linguistic Annotation*, pages 875–892. Springer.

Bougrine, S., Cherroun, H., and Abdelali, A. (2017). Altruistic crowdsourcing for arabic speech corpus annotation. *Procedia Computer Science*, 117:137–144.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada, August. Association for Computational Linguistics.

Chang, P.-C., Galley, M., and Manning, C. D. (2008). Optimizing chinese word segmentation for machine translation performance. In *Proceedings of the third workshop on statistical machine translation*, pages 224–232. Association for Computational Linguistics.

Cheng, X. and Roth, D. (2013). Relational inference for wikification. In *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1787–1796. Association for Computational Linguistics (ACL).

Cohen, K. B., Fort, K., Adda, G., Zhou, S., and Farri, D. (2016). Ethical issues in corpus linguistics and annotation: Pay per hit does not affect effective hourly rate for linguistic resource development on amazon mechanical turk. In *ETHics In Corpus collection, Annotation and Application workshop*.

Good, B. M., Nanis, M., Wu, C., and Su, A. I. (2014). Microtask crowdsourcing for disease mention annotation in pubmed abstracts. In *Pacific Symposium on Biocomputing Co-Chairs*, pages 282–293. World Scientific.

Kittur, A., Smus, B., Khamkar, S., and Kraut, R. E. (2011). Crowdforge: Crowdsourcing complex work. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 43–52, New York, NY, USA.

Liu, A., Soderland, S., Bragg, J., Lin, C. H., Ling, X., and Weld, D. S. (2016). Effective crowd annotation for relation extraction. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 897–906.

Mihalcea, R. and Csomai, A. (2007). Wikify!: Linking documents to encyclopedic knowledge. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 233–242, New York, NY, USA. ACM.

Moro, A. and Navigli, R. (2015). Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297, Denver, Colorado, June. Association for Computational Linguistics.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Ratinov, L., Roth, D., Downey, D., and Anderson, M. (2011). Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1375–1384. Association for Computational Linguistics.

Sabou, M., Bontcheva, K., Derczynski, L., and Scharl, A. (2014). Corpus annotation through crowdsourcing: Towards best practice guidelines. In *Proceedings of LREC*, pages 859–866.

Su, R., Shi, S., Zhao, M., and Huang, H. (2017). Utilizing crowdsourcing for the construction of chinese-mongolian speech corpus with evaluation mechanism. In *International Conference of Pioneering Computer Scientists, Engineers and Educators*, pages 55–65. Springer.

Tsai, C.-T. and Roth, D. (2016). Cross-lingual wikification using multilingual embeddings. In *Proceedings of HLT-NAACL*, pages 589–598.

van Gompel, M., van der Sloot, K., and van den Bosch, A., (2017). *Ucto: Unicode Tokeniser,Reference Guide, version 0.9.6*. Centre for Language Studies, Radboud University Nijmegen, January. URL: https://languagemachines.github.io/ucto/.

Wang, A., Hoang, C. D. V., and Kan, M.-Y. (2013). Perspectives on crowdsourcing annotations for natural language processing. *Language resources and evaluation*, 47(1):9–31.

Using Crowd Agreement for Wordnet Localization

Amarsanaa Ganbold*, Altangerel Chagnaa*, Gábor Bella†
*Department of Information and Computer Science, SEAS, National University of Mongolia
Ikh surguuliin gudamj – 3, Sukhbaatar district, 14201 Ulaanbaatar, Mongolia
†Department of Information and Computer Science, University of Trento
via Sommarive, 5, 38123 Trento, Italy
{amarsanaag, altangerel}@num.edu.mn, gabor.bella@unitn.it

Abstract

Building a wordnet from scratch is a huge task, especially for languages less equipped with pre-existing lexical resources such as thesauri or bilingual dictionaries. We address the issue of costliness of human supervision through crowdsourcing that offers a good trade-off between quality of output and speed of progress. In this paper, we demonstrate a two-phase crowdsourcing workflow that consists of a synset localization step followed by a validation step. Validation is performed using the inter-rater agreement metrics *Fleiss' kappa* and *Krippendorf's alpha*, which allow us to estimate the precision of the result, as well as to set a balance between precision and recall. In our experiment, 947 synsets were localized from English to Mongolian and evaluated through crowdsourcing with the precision of 0.74.

Keywords: crowdsourcing evaluation, inter-rater agreement, synset localization, wordnet

1. Introduction

Lexical-semantic resources like WordNet are a fundamental resource for many NLP and semantic applications. Building such resources traditionally involves the collaboration of a large number of professionals, such as psycholinguists and lexicographers, in order to obtain a high-quality end result. To mitigate the cost of WordNet construction, automatic and semi-automatic approaches have been successfully used, based on existing bilingual resources such as dictionaries and thesauri. Such methods, however, are hardly applicable to less-resourced languages that do not already have rich thesauri and dictionaries.

In this paper we introduce an alternative approach of localizing wordnet synsets through crowdsourcing. The process consists of two phases: in phase one, workers build synsets by contributing synonymous words while in phase two they validate the correctness of words provided earlier. Validation results are combined and assessed using two alternative statistical metrics of *inter-rater agreement*, the idea being that higher levels of agreement corresponds to higher precision. We evaluate this hypothesis with the help of a gold standard corpus. The evaluation provides us with an insight on the efficiency of crowdsourcing and allows us to optimize the coefficient parameters of these metrics for increased precision.

2. Related Work

Ontology localization, as described by Espinoza et al. (2009) and Cimiano et al. (2010), presents an alternative approach to addressing the cost of building lexical-semantic resources. An approach based on ontology label translation (Arcan and Buitelaar, 2013) was developed to provide a knowledge-based extension to a statistical machine translation system. However, automatically translated multilingual terms often suffer from quality issues. A preliminary crowdsourcing model for less-resourced languages (Benjamin and Radetzky, 2014) allows the building of lexicons with the help of Internet users. They concluded that extensive manipulation and review by language experts was necessary in order to obtain high-quality linguistic data and to capture a large diversity of knowledge. Crowdsourcing techniques were used to build wordnets from scratch through word sense acquisition (Biemann and Nygaard, 2010), to bootstrap a wordnet through translation between two languages (Wijesiri et al., 2014), and to annotate the correctness of synset words of an automatically developed wordnet (Fišer et al., 2014). In (Lanser et al., 2016), a two-stage workflow for translation-based crowdsourcing of ontology lexicons was designed and evaluated, with positive results. The difference of our approach lies in the fact that we evaluate synset words using inter-rater agreement.

3. Synset Localization

A wordnet synset is a set of synonymous words in a natural language that represents a lexical concept. A lot of such concepts can be mapped to equivalent or very similar concepts across languages, a principle without which bilingual dictionaries would not be possible. For instance, the English synset {`hovel, hut, hutch, shank, shanty`} is equivalent to the Mongolian synset {овоохой, оромж} and both synsets represents the concept `small crude shelter used as a dwelling`. This equivalence allows us to reuse the semantic structure of an existing wordnet (in this case, English) and to produce a partial wordnet (in this case, Mongolian) by following some basic principles of *ontology localization* (Ganbold et al., 2014b; Espinoza et al., 2009). While localization also involves well-known problems related to language diversity, such as conceptualizational mismatches or lexical gaps (Giunchiglia et al., 2017; Bentivogli and Pianta, 2000), we consider these problems as a separate research issue and prefer to focus in this paper on the more straightforward problem of synset localization. Synset localization provides a set of words in a target language that is equivalent to a given synset in a source language. Rather than a literal translation of words between languages, the localization of synsets involves finding the most appropriate words in the target language to express the

same meaning. Providing the words of a synset is a non-trivial task that involves the understanding of precise semantic distinctions with respect to hypernym and hyponym synsets, often based on psycholinguistic and lexicographic expertise. In this paper we examine to what extent crowdsourcing may offer a viable alternative to expert involvement for the specific task of synset localization.

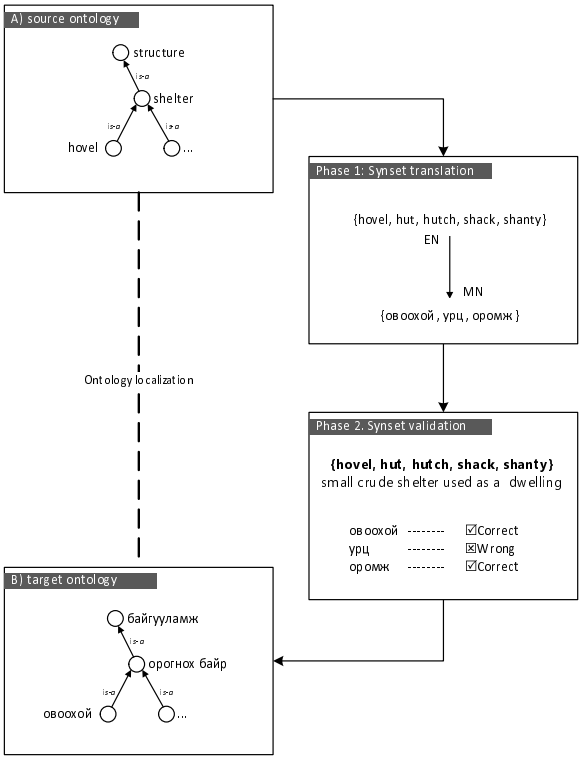


Figure 1: Crowdsourcing workflow of the synset localization. The concepts represented in the *ontology* boxes are labeled by a single word of their representative synsets.

We specify a two-phase crowdsourcing workflow (Figure 1) which consists of a translation and a validation phase. In the translation phase we ask several (in our case, five) workers to provide words for a synset in the target language (in our case, Mongolian, MN) that is equivalent to a given synset in the source language (English, EN). The number of words can be different in the two synsets. Thus, the same synset will be assigned to several workers who may end up suggesting many candidate words for the target synset.
The translation task allocated to workers provides them with synset words, the gloss of the synset, example sentences, the part of speech, as well as images from ImageNet[1] which might help workers in understanding the underlying meaning. If the concept cannot be localized because of a nonexistent lexicalization in the target language (e.g., the concept of *sea port* is not lexicalized in Mongolian) , a worker may mark it as a lexical gap. Workers can also skip tasks that they find difficult to translate or that contain words that are unknown to them. The HIT (Human Intelligence Task) had the following instructions:
Question: *Provide the most appropriate Mongolian word(s) for the following concept represented in English. Please follow the instructions below:*

- *A word must be a lexical unit which is one of a headword in dictionary, a phrase or a restricted collocation.*
- *A word must be written in Mongolian Cyrillic.*
- *A word can be case insensitive.*
- *Words should be separated by semicolon (;).*
- *If you think the concept cannot be localized, please mark it as a lexical GAP.*

A word is not acceptable if includes the following:

- *a Latin letter or words, or digits*
- *a special character (dot, colon etc.) except hyphens (-) which is acceptable in some Mongolian words*

Warning:

- *Image(s) may not exactly represent the concept in some cases.*
- *Please skip the task if you are not familiar with the concept and try to provide words for the next concept.*

4. Synset Validation

In the second phase, workers are asked to validate all distinct candidate words in the synset by annotating the words with three categories: *Correct*, *Wrong*, and *Unknown*.
Our validation task design had the following instructions:
Question: *Validate the Mongolian word(s) which can represent following concept. Please follow the instructions:*

- *assign a category of Correct, Wrong and Unknown for each word;*
- *assign the Wrong category if the word has spelling errors;*
- *assign the Wrong or the Unknown category for all words, if you think the concept is GAP and, assign the Correct category to the GAP entry.*

This evaluation is not acceptable if you do following:

- *randomly evaluate the words;*
- *assign a single category to all words and repeated several times.*

Validation results were collected on a word-by-word basis. We did not consider word ranks (the order of relevance of words) for this paper.
On the synset level, validation results were aggregated to compute the *inter-rater agreement* using the statistical metrics *Fleiss' kappa* (eq. 1) and *Krippendorff's alpha* (eq. 2). These metrics can measure agreement between a fixed number of workers who provided words. The higher the values of these metrics (close to 1), the more certainty we have that the workers have correctly classified the synset

[1] http://image-net.org/

words into correct and incorrect ones, and ultimately the more confident we are of the precision of the result.

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \quad (1)$$

where $\bar{P} - \bar{P}_e$ is the degree of achievable agreement and $1 - \bar{P}_e$ is the degree of agreement actually achieved. $\bar{P}$ is the mean of all proportions, where a proportion is the agreement for a given candidate word. $\bar{P}_e$ is the sum of squares of weighted votes of each category. $\kappa = 1$ means the raters have total agreement over *Correct* or *Wrong* or *Unknown*, while $\kappa < 0$ means that there is no agreement among raters. If the value is $0 < \kappa \leq 1.0$, there is some level of agreement among raters. In our case, the optimal value for moderate agreement can depend on factors such as words, categories, and raters.

$$\alpha = 1 - \frac{D_o}{1 - D_e} \quad (2)$$

While κ calculates the number of categories of each word in a synset, α computes the agreement from the data that includes category labels per annotator of the words in the synset. The range of α values is same as κ . In eq. 2, D_o is the disagreement observed and D_e is the disagreement expected by chance.

Alpha allows missing data, i.e., no categorization for some words. The *Unknown* category can be considered as missing data as this option is selected by annotators who do not know what category to assign to the word.

For each synset, when a certain agreement is reached, we extract words that have a majority of *Correct* votes, considered as the most suitable for the target synset. This way we combine the workers' contributions for the translation task. To the best of our knowledge, there are no experiments on the use of Fleiss' kappa and Krippendorff's alpha which can be used to combine workers' contributions. These metrics meet our requirements of validation of synset localization.

5. Experiments and Results

We used *CrowdCrafting*[2] for recruiting workers because of a limited presence of Mongolian speakers on platforms such as *Amazon Mechanical Turk* and *CrowdFlower*. CrowdCrafting is free for scientific projects with volunteer contributors. In phase 1, the total of 77 web users were asked to translate 947 manually built synsets from the *space domain*, that is, the subtree under the high-level synsets of *space* in (Ganbold et al., 2014b; Giunchiglia et al., 2009). In phase 2, 75 web users were asked to validate the results of phase 1. In total, contributors have completed 9,490 tasks and have introduced 6,442 words[3].

In order to evaluate contributions from the crowd, we compiled a gold standard from the space domain in Mongolian, covering all synsets that were included in the crowdsourcing experiment. The gold standard corpus was created by five language experts using an *expert sourcing* methodology described in (Ganbold et al., 2014b; Ganbold et al., 2014a). The methodology involved translation by bilingual translators and a two-level validation process where as many iterations were performed on each translation as necessary for reaching a high-quality result through inter-expert agreement.

Source	# of synsets	# of words
Space domain	943	1,436
Expert validation	889	1,813
Total	**943**	**2,627**

Table 1: Size of the gold standard *space domain* corpus and of the crowd-translated corpus post-validated by experts.

While expert sourcing provides a high-quality reference corpus, it does not guarantee exhaustiveness: crowd workers may come up with alternative yet acceptable lexicalizations. For this reason we completed the evaluation based on our *a priori* gold standard by an *a posteriori* expert validation step on the output of workers, carried out by three language experts. Table 1 shows the size of the gold standard corpus and of the crowd translations for which expert validation was provided. The higher number of words in the expert-validated corpus indicates that the crowds managed to provide correct new lexicalizations that were not present in the gold standard, i.e., that had not been though of by experts.

Evaluation of the entire crowdsourced corpus yielded an overall precision of 0.74. Our main research goal, however, was the evaluation of inter-rater agreement as a tool for controlling the precision of the output. For this purpose, we first computed inter-rater agreements on each synset separately. Then we created increasingly smaller subsets of synsets corresponding to a certain minimum level of inter-rater agreement. For example, to understand the effect of imposing an agreement threshold of 0.5, we ran an evaluation only on synsets with an agreement of 0.5 or higher. The result of these evaluations is shown in Figure 2 where the *x* axis indicates agreement thresholds and the *y* axis indicates precision, recall, and F-measure.

These values were computed separately for the three different agreement metrics, namely, *Fleiss' kappa*, *Krippendorff's alpha*, and *Krippendorff's alpha/NA*. *Alpha/NA* is an alternative form of Krippendorf's alpha where the *Unknown* category is considered as missing data and is not included in calculations.

The first observation we can make from the results is that precision monotonically increases with inter-rater agreement for all three metrics. From this we conclude that inter-rater agreement generally is a reliable tool to control the precision of the outcome, at least for synset localization tasks. We did not obtain a significant difference related to the three different metrics employed (alpha tends to report somewhat more conservative agreement results than the two other metrics for the same precision). At the same time, we observe that for higher agreement levels (above 0) the rapidly increasing precision is accompanied by an even more rapidly falling recall. This indicates to us that, at

[2]https://crowdcrafting.org

[3]Data collected during the two phases are available at https://crowdcrafting.org/project/mongolian-lkc and at https://crowdcrafting.org/project/mongolian-lkc-evaluation under CC-BY-SA license.

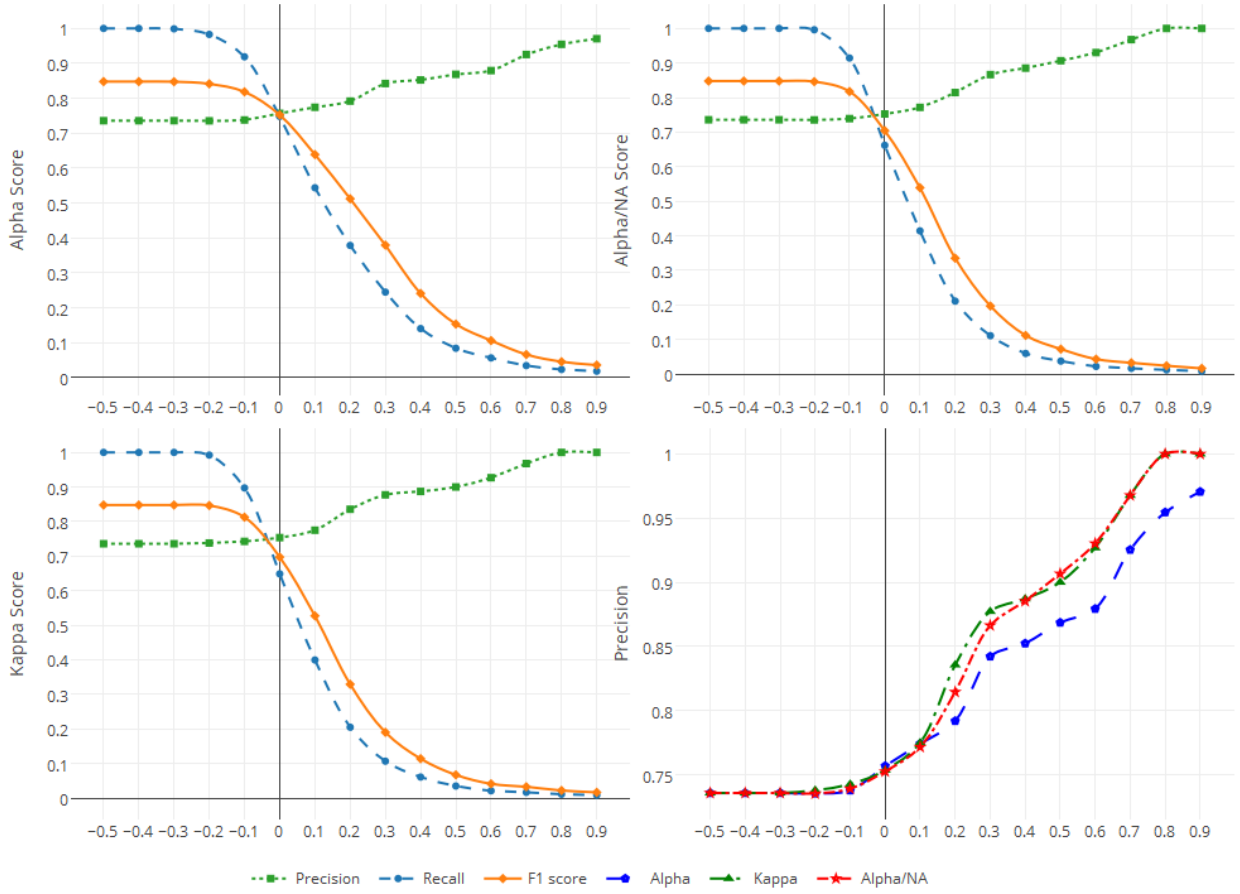


Figure 2: Experiment scores versus agreement values (x-axis)

least in our experiment, inter-annotator agreement was not frequent. Our hypothesis is that both the level of agreement and the overall precision could be improved in at least two distinct ways: by a more careful selection of workers based on competence (which may result in a less heterogeneous output) and by a more sophisticated allocation of tasks to workers. For example, by allocating all hyponyms of a synset to the same worker the *differentia* among them would become more evident and would possibly result in better-quality localizations. We leave these improvements for future work.

6. Conclusion and Future Work

We presented a two-phase crowdsourcing workflow for wordnet localization, consisting of a synset translation phase and of a subsequent validation phase. Two statistical metrics, *Fleiss' kappa* and *Krippendorff's alpha*, were used to compute inter-rater agreement. The overall precision of crowdsourced localizations was 0.74. We found that both agreement metrics can be successfully used to control the precision of the result, the latter monotonically increasing starting from the agreement threshold of about -0.1. However, choosing an agreement threshold significantly higher than this value decreased recall considerably, a fact that hints at a generally low level of worker agreement. In future work we plan to address this point by a more careful selection of workers and of localization tasks. As future work, we will refine our crowdsourcing workflow to improve F1-score as well as our crowd task generation algorithm. Further goal is to apply this crowdsourcing approach to get a usably-sized Mongolian wordnet.

7. Acknowledgements

The research has received funding from the Mongolian Science and Technology Fund under grant agreement SSA_024/2016. The result described in this paper is part of the long-term project *Mongolian Local Knowledge Core*, partially supported by the National University of Mongolia under grant agreement P2017-2383.

8. Bibliographical References

Arcan, M. and Buitelaar, P. (2013). Ontology Label Translation. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*, pages 40–46. The Association of Computational Linguistics.

Benjamin, M. and Radetzky, P. (2014). Multilingual Lexicography with a Focus on Less-Resourced Languages : Data Mining , Expert Input , Crowdsourcing , and Gamification Acquiring Lexical Data for LRLs. *9th edition of the Language Resources and Evaluation Conference*.

Bentivogli, L. and Pianta, E. (2000). Looking for lexical gaps. In *In Proceedings of the Ninth EURALEX International Congress*, pages 663–669.

Biemann, C. and Nygaard, V. (2010). Crowdsourcing WordNet. In *Proceedings of the 5th Global WordNet conference*, pages 5659–5664, Mumbai, India.

Cimiano, P., Montiel-ponsoda, E., Buitelaar, P., and Espinoza, M. (2010). A Note on Ontology Localization. *Applied Ontology*, 5:127–137.

Espinoza, M., Montiel-Ponsoda, E., and Gómez-Pérez, A. (2009). Ontology localization. In *Proceedings of the fifth international conference on Knowledge capture - K-CAP '09*, K-CAP '09, pages 33–40, New York, New York, USA. Universidad Politécnica de Madrid, Spain, ACM Press.

Fišer, D., Tavčar, A., and Erjavec, T. (2014). sloWCrowd: a Crowdsourcing Tool for Lexicographic Tasks. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3471–3475, Reykjavik, Iceland, 5. European Language Resources Association (ELRA).

Ganbold, A., Farazi, F., and Giunchiglia, F. (2014a). An Experiment in Managing Language Diversity Across Cultures. In *eKNOW 2014 : The Sixth International Conference on Information, Process, and Knowledge Management*, number c, pages 51–57, Barcelona, Spain.

Ganbold, A., Farazi, F., Reyad, M., Nyamdavaa, O., and Giunchiglia, F. (2014b). Managing Language Diversity Across Cultures : the English-Mongolian Case Study. *International Journal on Advances in Life Sciences*, 6(3):167–176.

Giunchiglia, F., Dutta, B., and Maltese, V. (2009). Faceted Lightweight Ontologies. In *Conceptual Modeling Foundations and Applications*, volume 5600, pages 36–51.

Giunchiglia, F., Batsuren, K., and Bella, G. (2017). Understanding and exploiting language diversity. In *IJCAI International Joint Conference on Artificial Intelligence*, pages 4009–4017.

Lanser, B., Unger, C., and Cimiano, P. (2016). Crowdsourcing Ontology Lexicons. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3477–3484. European Language Resources Association (ELRA).

Wijesiri, I., Gunathilaka, B., Wimalasuriya, D., Paranavithana, R., Gallage, M., Lakjeewa, M., Dias, G., and de Silva, N. (2014). Building a WordNet for Sinhala. In *GWC 2014: Proceedings of the 7th Global Wordnet Conference*.

Translation Crowdsourcing: Creating a Multilingual Corpus of Online Educational Content

Vilelmini Sosoni[1], Katia Lida Kermanidis[2], Maria Stasimioti[1], Thanasis Naskos[2], Eirini Takoulidou[2], Menno van Zaanen[3], Sheila Castilho[4], Panayota Georgakopoulou[5], Valia Kordoni[6], Markus Egg[6]

[1]Department of Foreign Languages, Translation and Interpreting, Ionian University, Corfu, Greece
[2]Department of Informatics, Ionian University, Corfu, Greece
[3]Department of Communication and Information Sciences,Tilburg University, The Netherlands
[4]ADAPT Centre, Dublin City University, Dublin, Ireland
[5]Deluxe Media Europe, Deluxe Entertainment Services Group, London, UK
[6]Department of English Studies, Humboldt-Universität zu Berlin, Germany
vilelmini@hotmail.com; kerman@ionio.gr; stasimioti.maria@gmail.com; anaskos@ionio.gr; rinoulit@gmail.com; mvzaanen@uvt.nl; sheila.castilho@adaptcentre.ie; yota.georgakopoulou@bydeluxe.com; kordonie@anglistik.hu-berlin.de; markus.egg@anglistik.hu-berlin.de

Abstract

The present work describes a multilingual corpus of online content in the educational domain, i.e. Massive Open Online Course material, ranging from course forum text to subtitles of online video lectures, that has been developed via large-scale crowdsourcing. The English source text is manually translated into 11 European and BRIC languages using the CrowdFlower platform. During the process several challenges arose which mainly involved the in-domain text genre, the large text volume, the idiosyncrasies of each target language, the limitations of the crowdsourcing platform, as well as the quality assurance and workflow issues of the crowdsourcing process. The corpus constitutes a product of the EU-funded TraMOOC project and is utilised in the project in order to train, tune and test machine translation engines.

Keywords: parallel corpus, MOOCs, online educational text, crowdsourcing

1. Introduction

Massive Open Online Courses (MOOCs) have been growing rapidly in number of enrollees and participating universities, in impact, and in target groups. According to 2016[1] statistics, they involve around 60 million students worldwide. In their majority, MOOC users are non-native English language speakers, and the language barrier proves to be the main obstacle towards further expansion of the MOOC market, as most courses are offered in English only.

The EU-funded TraMOOC (Translation for Massive Open Online Courses) project aims at providing machine translation solutions for the educational content available in MOOCs, thus enhancing access to online education. The content genre varies from video lecture subtitles, slides, assignments, and quiz text, to course discussion forum text (Kordoni et al., 2016). Given that the source language is English, in-domain trained and tested translation engines are built for 11 European and BRIC target languages (Bulgarian, Chinese, Croatian, Czech, Dutch, German, Greek, Italian, Polish. Portuguese, and Russian).

In an attempt to achieve optimal translation output, the TraMOOC goal is to develop as much in-domain data as possible, especially for the language pairs that are not adequately supported by the required infrastructure. To this end, crowdsourcing is adopted (as an alternative to using professional translators) for collecting translations on a large scale for all language pairs involved.

The present work describes the in-domain source data collected, the crowdsourcing experiment and the resulting multilingual parallel dataset, taking into consideration the challenges imposed by the text genre, the number of language pairs involved, the large data volume aimed at, the quality assurance of the experimental process, and the related crowdsourcing workflow issues.

2. Related Work

The creation of parallel data on a large scale for new language-pairs requires intensive human effort and availability of experts. For most language-pairs, the small number of expert translators available or the lack of access to fluent bilingual speakers makes it difficult and expensive to create parallel corpora for training machine translation systems. Recent research has looked at obtaining translations via crowdsourcing, in particular for low resource languages (Ambati & Vogel, 2010; ; Zaidan & Callison-Burch, 2011; Post, Callison-Burch, Osborne, 2012). Crowdsourcing as an approach to activate or use the knowledge and skills of a large group of people in order to solve problems has existed for a long time (cf. Ellis, 2014). Nowadays, it leverages Web 2.0 tools (O'Reilly, 2007) in order to take a job normally performed by a designated person and having it done by a large, undefined, and dispersed number of participants (Howe, 2008). In the area of translation, crowdsourcing has actually been used widely in the past years for the translation of online content. As Jiménez-Crespo (2017) observes, Facebook has used it for the translation of its social networking site and its user interfaces, Amara and TED for audiovisual practices, Kiva and the platform Trommons from the Rosetta Foundation for non-profit initiatives.

For the generation of parallel corpora, the most widely used crowdsourcing platform is MTurk[2] (Ambati & Vogel, 2010; Zaidan & Callison-Burch, 2011; Post, Callison-Burch, Osborne, 2012), although Negri &

[1] https://www.class-central.com/report/mooc-stats-2016/

[2] https://www.mturk.com/

Mehdad (2010) used CrowdFlower[3] for the creation of a bilingual Textual Entailment corpus for the English/Spanish language pair taking advantage of an already existing monolingual English RTE corpus.

Although the benefits from collecting corpora using crowdsourcing techniques are numerous, gathering data is cheap, quick to acquire, and varied in nature, it does not go without carrying risks such as quality control and workers that try to "game" the system.

Taking the pros and cons into account, the present work extends previous work by using the CrowdFlower platform for the translation of both formal and informal English sentences of educational content to both low- and high-resource languages (11 European and BRIC target languages) and by applying various quality measures and features.

3. Source Data

The source data are comprised of online educational course material in English. More specifically, they contain lecture subtitles and quiz assessment text (considered henceforth *formal* text), and course forum discussion text that students share among themselves and/or the instructor for posting questions, clarifications, opinions etc. (considered henceforth *informal* text). The topics of the courses varied from technical (e.g. Finance) to humanities (e.g. Future of Storytelling).

3.1 Data Sources

The English source text comes from several different channels. Henceforth, a 'segment' is a piece of text between two consecutive CR/LF characters.

Iversity.org. A large part of the dataset (~35,000 segments), formal and informal, originated from the MOOC provider Iversity.org.

Videolectures.NET. Videolectures.NET is a library of online educational video lectures. 800 segments of lecture subtitles (formal text) of the 'Complexity Science' course were included in our dataset.

Coursera. 27,000 segments originated from subtitles of online courses provided by Coursera (formal text). The number of courses exceeded 280, and varied between 'Web Applications', 'Public Policy', and 'Art History'.

QED. 28,000 segments were transcripts of video lectures (formal text) selected from the QCRI Educational Domain Corpus (QED)[4].

3.2 Data Description

Regarding the formal text, several properties rendered its processing quite challenging. On the one hand, it exhibited a high frequency of domain specific terms and expressions, named entities, scientific formulas, and words unknown to crowdworkers, as well as to any system for posterior processing. On the other hand, the subtitle genre contained spontaneous speech properties, truncated sentences, elliptical formations, disfluencies, repetitions, interjections and fillers.

Example 1: *What? What? He's going to score, he's in! Whoops. And I've got, ooh, here we go.*

Example 2: *Sharing Economy is the way of build resources and get to go ahead on the path, where we got new creation, trust, together and we feel better for the responsibilities.*

Example 3: *Hello, Imstuding fashion design and my aim it's to become a sustainable and ethical fashion designer.*

Regarding the informal text, it presented all the properties of social media text: slang, misspellings ('supa' instead of 'super'), lexical variants, abbreviations, acronyms, multilingual tokens, unorthodox syntax structures, disfluencies, awkward word choices, and repetitions.

Example 4: *Puuurrrfect!*

Example 5: *truthfully , i have no idea how much i reason per day day.*

Course title	No of segments	
	Formal	Informal
Business Analysis	378	5148
Contemporary Architecture	258	5219
Crystals and Symmetry	144	3338
Dark Matter	379	4443
Gamification Design	319	9121
Public Speaking	113	-
Web Design	270	1523
Critical Thinking	550	500
Social Innovation	550	500
Monte Carlo Methods in Finance	700	500
Modeling and Simulation using Matlab	150	440
Future of Storytelling	550	460
Total	4361	31192

Table 1: Segment size per course in the Iversity.org data.

3.3 Data Preparation

In order for the data to be appropriately formatted for further processing, it had to undergo a preparation phase, which included:

Conversion into plain text. The text is freed from all markup and meta information, and converted into plain UTF-8 text format, using Python and UNIX-based shell scripts. Special characters were removed, along with non-

[3] https://www.crowdflower.com/

[4] http://alt.qcri.org/resources/qedcorpus

content lines, and multiple or trailing whitespace characters.

Tokenization, and sentence segmentation. The plain text data was tokenized into words and punctuation. Correct segmentation involved the removal of incomplete segments, and/or of segments that contained multiple sentences.

Example 6: *and often*

Example 7: *say the ... in the case of the peacock's tail , say,*

Also, due to the subtitle transcription process, which was undertaken to a large extent by non-experts, and due to the presence of spoken language characteristics in the subtitle text (incomplete/unfinished sentences, intersentential change of topic), very often a segment includes the ending part of the previous sentence and the starting part of the next, as shown in the following example.

Example 8: *this is his name. Five years ago, whenever I took over as*

Scripts were built to deal with incorrect segmentation, although they were not completely error-free, because automatic deep understanding of the text was essential, but not feasible. The toolset is available online[5].

Markup of special elements. Some textual elements, such as URLs and emoticons, are automatically replaced with special tags. This would ensure that translators would not try to translate these elements, whilst this abstraction is also considered beneficial for the MT systems, as the latter would be correctly trained to leave them untranslated.

Data setup for the translation crowdsourcing activity. A set of 5000 segments was selected to constitute the tuning and testing set for the upcoming machine translation experiments. These segments were taken from the pool of Iversity.org, and the entire set of Videolectures.NET segments, and were translated by at least two and at most three workers per target language for redundancy purposes. The rest of the segments were translated once per target language.

4. Crowdsourcing Experiments

A total of 2050 workers participated in the crowdsourcing experiments, which took place from March to June 2017. They were given clear and detailed instructions, both general and target-language specific, on how to translate the English segments presented to them. Instructions contained specific examples to help workers deal with typical linguistic (e.g. the translation of acronyms, proper names etc.) or formatting (e.g. dealing with punctuation issues) challenges appearing in the text. Segments were presented in a stand-alone, out-of-context manner, but workers were provided with the course title, when available, in order to get a better understanding of the context. No terminological lexicon was provided to them, as this was not feasible due to the vast number of diverse topics. In order to keep the crowdsourcing task as simple as possible, no further annotation was required by the workers, e.g. pertaining to the grammaticality of the input segment.

Quality assurance was supported by a test mode, where workers were asked to answer a set of test questions, i.e., choose the best translation for a source segment among three candidates. Thereby, the workers' accuracy level was determined. The test mode occurred before the first translation questions, but also during the entire process task for continuous monitoring of the workers' quality. Once the accuracy level of a worker dropped below a certain threshold, the workers could not continue working on the task. Additionally, the worker was blocked to continue working on other crowdsourcing experiments in the TraMOOC project.

Performing accuracy measurements this way is a trade-off. Using question types that are different from those in the task allows workers to easily identify the test questions. Once these are answered correctly, the actual translations can, for example, be generated randomly. The results showed that some workers used hill-climbing with different accounts to collect the correct answers. The entire process of filling in the data can then be automated, resulting, obviously, in unusable translations.

To make sure that only correct translations were collected, close and constant, albeit to a large extent automated, monitoring of the workers' input proved to be the most important element for ensuring quality annotations, banning spammers and removing worthless input.

5. Multilingual Corpus of Online Educational Content

The outcome of the crowdsourcing activity we launched on CrowdFlower was the collection of a large dataset of parallel corpora in the 11 languages of the project, although the number of segments varied per language (Figure 1). Participants who followed the instructions and showed competent behavior (no blank inputs, no random translations, accuracy above 80% threshold) were labeled "trusted" and their translations were accepted as such. As shown in Figure 1 below, the Italian and Croatian languages collected over 90,000 'trusted' translated segments. The Italian language collected the smallest amount of untrusted judgements (less than 2000 judgments) and the vast majority of Italian translated segments came from contributors located in Italy (~90% of the participants). Regarding the Croatian language, the large amount of collected translations is considered a significant pillar of linguistic infrastructure, taking into consideration that there are no sufficiently large parallel corpora available, and that it is the most weakly NLP-resource-supported language in the project. Due to malicious behavior of the contributors, the Croatian

[5] https://github.com/mixstef/tramooc

language collected over 10,000 untrusted judgments. Another interesting result is that half of the contributors were from Serbia followed by Bosnia and Herzegovina in second place. Datasets for Russian and Portuguese followed suit with more than 80,000 translated segments. These languages collected ~8000 and ~6000 untrusted judgements respectively. In the Portuguese crowdsourcing task only crowd workers who were located in Portugal (40%) and Brazil (60%) took part, while in the Russian task there were mostly contributors located in the Russian Federation (~75%). Greek and Polish crowdsourced translations reached over 70,000 segments, which is a remarkable amount for such low-resource languages. The amount of untrusted judgements was relatively small for both languages (less than 5000 and 7000 respectively). In the first case, the majority of the contributors were from Greece (90%) and in the second case from Poland (85%). Bulgarian and Chinese tasks gathered ~60,000 segments. Regarding the Bulgarian language, the amount of collected data is quite satisfactory, taking into consideration that it is a low-resource language. The percentage of untrusted judgements was 10% and the contributors were from Bulgaria (~70%), FYROM (~12%) and the Russian Federation (~8%). The untrusted judgements for the Chinese language were around 7000. Most of the contributors were located in Hong Kong, followed by China, Malaysia and Taiwan. German and Czech translated segments were above 50,000. The untrusted judgments were around 7000 and 5000 respectively. In the first case, the contributors were from Germany (~80%), followed by contributors from Austria (~10%). In the second case, the contributors were from the Russian Federation, Poland, Czech Republic and Venezuela. For the Dutch language, however, we managed to collect only ~40,000 translations (less than half of Italian task), with ~1000 untrusted judgements. The contributors were mainly from the Netherlands (~50%) and Belgium (~40%).

Examples of trusted and untrusted segments are provided:

Trusted segments:
EN: *An interface is simply the point where two entities meet.*
EL: *Μια διεπαφή είναι απλώς το σημείο όπου δυο οντότητες συναντιούνται*

EN: *And I get C d theta dt.*
DE: *Und ich bekomme C d theta dt.*

Untrusted segments:
EN: *What Gamification, Game Thinking and Games are*
EL: *What Gamification, σκέψης παιχνίδι και τα παιχνίδια είναι*

EN: *Nope, I am good*
DE: *Amet, quidem sit accusamus et eveniet, repudiandae culpa, quam aut*

The provision of partly translated sentences (example of the EN-EL language pair), Latin text (example of the EN-DE language pair) and blank translations were common practices among "untrusted" contributors and such cases were spotted in all language pairs.

The language-specific workflow data serves as a means to assess the size of the crowd channels linked to the particular crowdsourcing platform. It turns out that, while certain languages are satisfactorily supported by crowd channels of workers that speak them, others are not. Furthermore, channel support is not aligned with Natural Language Processing (NLP) resource support. Languages satisfactorily equipped with NLP tools are not necessarily supported by large crowdsourcing channels (e.g. Dutch), and vice versa.

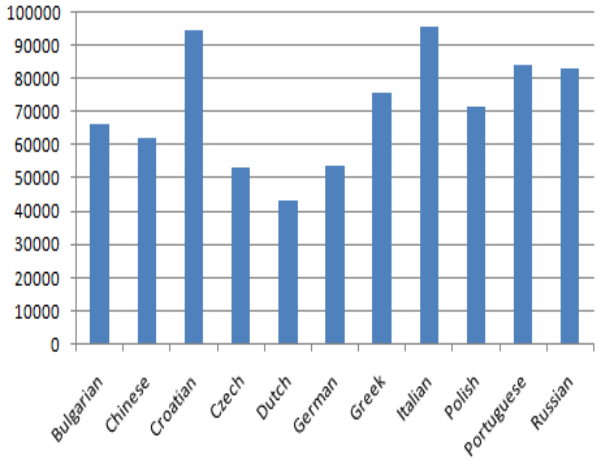


Figure 1: Number of trusted translated segments per target language.

As mentioned earlier, the primary use of the collected data is for developing translation engines for online course material. To this end, it was deemed particularly important to gather the data in the strongly supervised manner described earlier, so as to reach a satisfactory quality level that would prove beneficial for the MT systems (Behnke et al., 2018).

6. Conclusion

The multilingual corpus of online course material described in this article was developed manually in 11 languages via crowdsourcing. The challenges encountered due to the genre of the video lecture transcripts (i.e. spontaneous speech) and the social media forum text, as well as crowdsourcing workflow issues for some of the languages are presented. Language-specific workflow phenomena serve as an indication of the size of the crowd channels supported by the platform for every language.

Close monitoring of the crowdsourcing process proved to be key in addressing the aforementioned challenges, and ensuring the required quality threshold of the provided annotations. The process led to a satisfactory percentage of trusted judgements, resulting in a large-scale multilingual corpus of online course material. The corpus will be made available through the EU (according to the H2020 Open Research Data Pilot) for research purposes after the end of the project, and taking into account copyright restrictions imposed by each source.

Acknowledgements

This work was performed as part of the TraMOOC project (Translation for Massive Open Online Courses), funded by the European Commission under H2020-ICT-2014/H2020-ICT-2014-1 under grant agreement number 644333.

References

Ambati, V., & Vogel, S. (2010). Can crowds build parallel corpora for machine translation systems?. In Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk (pp. 62-65). Association for Computational Linguistics.

Behnke et al. (2018). Improving Machine Translation of Educational Content via Crowdsourcing. Proceedings of the International Conference on Language Resources and Evaluation, Miyazaki, Japan (to appear).

Ellis, S. (2014). A History of Collaboration, a Future in Crowdsourcing: Positive Impacts of Cooperation on British Librarianship. Libri 64 (1), pages 1-10.

Jiménez-Crespo, M. A. (2017) Crowdsourcing and Online Collaborative Translations: Expanding the limits of Translation Studies. Amsterdam/Philadelphia: John Benjamins.

Kordoni et al. (2016). Enhancing Access to Online Education: Quality Machine Translation of MOOC Content. Proceedings of the International Conference on Language Resources and Evaluation, pages 16-22, Portoroz, Slovenia.

Negri, M., & Mehdad, Y. (2010). Creating a bi-lingual entailment corpus through translations with mechanical turk: $100 for a 10-day rush. In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (pp. 212-216). Association for Computational Linguistics.

O'Reilly, T. (2007) What is Web 2.0: Design patterns and business models for the next generation of software. International Journal of Digital Economics, 65, pages 17-37.

Post, M., Callison-Burch, C., Osborne, M. (2012). Constructing parallel corpora for six Indian languages via crowdsourcing. In Proceedings of the Seventh Workshop on Statistical Machine Translation (pp. 401-409). Association for Computational Linguistics.

Zaidan, O. F., & Callison-Burch, C. (2011). Crowdsourcing translation: Professional quality from non-professionals. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 1220-1229). Association for Computational Linguistics.

Building an English Vocabulary Knowledge Dataset of Japanese English-as-a-Second-Language Learners Using Crowdsourcing

Yo Ehara
Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology
2-4-7 Aomi, Koto-ku, Tokyo, Japan
i@yoehara.com, y-ehara@aist.go.jp

Abstract

We introduce a freely available dataset for analyzing the English vocabulary of English-as-a-second language (ESL) learners. While ESL vocabulary tests have been extensively studied, few of the results have been made public. This is probably because 1) most of the tests are used to grade test takers, i.e., placement tests; thus, they are treated as private information that should not be leaked, and 2) the primary focus of most language-educators is how to measure their students' ESL vocabulary, rather than the test results of the other test takers. However, to build and evaluate systems to support language learners, we need a dataset that records the learners' vocabulary. Our dataset meets this need. It contains the results of the vocabulary size test, a well-studied English vocabulary test, by one hundred test takers hired via crowdsourcing. Unlike high-stakes testing, the test takers of our dataset were not motivated to cheat on the tests to obtain high scores. This setting is similar to that of typical language-learning support systems. Brief test-theory analysis on the dataset showed an excellent test reliability of 0.91 (Chronbach's alpha). Analysis using item response theory also indicates that the test is reliable and successfully measures the vocabulary ability of language learners. We also measured how well the responses from the learners can be predicted with high accuracy using machine-learning methods.

Keywords: Vocabulary Test, Item Response Theory, Crowdsourcing

1. Introduction

Supporting someone involves enabling him/her is overcoming a difficulty. Thus, we first need to detect what he/she has difficulty with. This holds true for supporting second language learners: we first need to detect what they have difficulty with (Ehara et al., 2010; Ehara et al., 2012; Ehara et al., 2013; Ehara et al., 2014; Ehara et al., 2016). To detect such difficulties, we need to test them. The knowledge of a language is too large, and testing all types of knowledge imposes a heavy burden on them. This is where vocabulary tests come into play. Vocabulary tests measure language learners' knowledge of words, especially their meaning. Compared to other aspects of language, such as syntax, learners' vocabulary can be easily measured and can be used to understand what kind of difficulties that language learners face: the largeness of vocabulary, or a set of words, is easily measured due to its size. Its elements, or words, are also countable. The difficulty of words is roughly measured by their frequency in a corpus: the rarer a word, the more difficult it may be to learners. Most importantly, whether a learner knows a typical meaning of a word can be easily measured through multiple-choice questions, the results of which (i.e., learners' responses to the test) are machine-readable and can be easily automatically scored if the correct answers of the test are known. Therefore, the results of a vocabulary test given to a language learner provide essential information to support his/her language learning. This is why this type of test is frequently used as a placement test from which language learners are categorized into a "class" with similar difficulties in using the second language that they are learning.

Although the results of vocabulary tests are important for supporting language learners, few datasets are, however, available for building and evaluating language-support systems. This is probably because 1) most of the tests are used to grade test takers, i.e., placement tests; thus, they are treated as private information that should not be leaked, and 2) the primary focus of most language-educators is how to measure their students' ESL vocabulary, rather than the test results of other test takers.

In this paper, we introduce a freely available dataset for analyzing English vocabulary of English-as-a-second language (ESL) learners. In our dataset, 100 learners answered 100 well-tested vocabulary questions. Unlike a typical setting in which test takers are in classrooms of a language-learning course, we employed 100 learners via crowdsourcing, which means they were paid.

The test results in which test takers are paid are more suitable for a dataset to be used in building and evaluating language-support systems than those in which test takers are not paid. Vocabulary tests can be high-stakes tests when they are used for placement: test-takers have high stakes if they obtain good scores in the test, so they have strong motivation to obtain high scores. In a high-stakes test, language learners tend to pretend to have knowledge on the second language. Vocabulary tests as a high-stakes test are common as placement tests, but are not common in language-support systems because users know that they may be provided with incorrect support if they pretend. Thus, our dataset is more similar to a realistic environment in which language learners do not have motivation to exaggerate their knowledge of second language vocabulary.

We conducted a brief statistical analysis on our dataset, which showed that it is effective in terms of measuring ability. We also measured how accurately the responses from the learners can be predicted using machine-learning methods.

2. Related Work

Many previous papers have reported on the *analysis* vocabulary-test data using datasets not publicly available. Typically, one hundred to several hundred participants participated in the vocabulary tests in these studies. (Culligan, 2015) compared three typical test formats and showed that all three have high reliability and measure the vocabulary ability of language learners. Their results are based on vocabulary-test data of 54 words collected from 167 university students.

Recently, large experiments on an English vocabulary test were conducted with over 1,000 test takers and hundreds of words (Webb et al., 2017); however, their raw test-result data are not publicly available. This large number of test takers was provided by language teachers worldwide.

To the best of our knowledge, no vocabulary-test result datasets is publicly available. This is presumably because the data are usually collected from language classes on a volunteer basis. This method of collecting vocabulary-test results may be beneficial for classroom teaching because the environment under which a dataset is taken is a classroom and the applications to which the dataset are used are also for classrooms. However, this is not the case for developing educational software, in which participants are more diverse than in typical classrooms.

The vocabulary dataset by Ehara et al. (2010) is a publicly-available vocabulary knowledge data [1]. This dataset collects the results of $12,000$ word questions by 16 people. Notable differences between their dataset and our dataset are: 1) theirs were collected in a self-report manner, thus, strictly speaking, their dataset is not "test results". 2) the number of test-takers are 100 in our dataset while theirs are merely 16. Thus, our dataset is more accurate for small size of vocabulary while theirs focus on the size of vocabulary to be tested with sacrificing accurateness.

3. Dataset

The purpose of building our dataset was to use it for building and evaluating language-support systems. Such systems are used by a diverse population of users. This setting is very different from that of testing in classrooms of a language course in which the characteristics of test takers are usually limited to a category, for example, university students of the same year. To meet the purpose of this dataset, we employed a diverse population of test takers via crowdsourcing.

The data for the dataset were collected via a crowdsourcing service called Lancers, one of the major crowdsourcing companies in Japan. We used the vocabulary size test (VST) (Nation and Beglar, 2007) for this dataset. This test was designed to measure the vocabulary size of each learner. In this test, test takers are asked to answer 100 vocabulary questions. Each question has four options and only one of the options is correct. This means that each learner has a 25% chance to answer each question correctly regardless of their vocabulary knowledge. We employed 100 test takers. We paid each test taker 383 Yen (approximately 3.5 USD). An example question in the vocabulary size test (Nation and Beglar, 2007) is as follows:

microphone: Please use the <microphone>.

a machine for making food hot

b machine that makes sounds louder

c machine that makes things look bigger

d small telephone that can be carried around

To compare the test results to other test results, we limited the test takers to those who had previously taken the TOEIC test [2], which is a popular English proficiency tests in Japan. The choice of this test as a reference is simply because of the number of test takers and the internationality of the test. For example, although we know that the TOEIC test is popular in only certain Asian countries including Japan and other tests such as TOEFL [3] are more universally popular, we chose TOEIC as a reference because, we cannot collect enough test-takers on any crowdsourcing service popular in Japan. We did not include English proficiency tests popular almost solely in Japan, such as the "Eiken" test, an English proficiency test popular with Japanese high-school students While we required our test takers to have a TOEIC test score, it did not matter when they took the test. The reason of this is also not to limit the diversity of test takers. Since many people do not take a language-proficiency test after they graduate from universities, it is easily speculated that limiting the time when the learners took a proficiency test would lead to limiting the diversity of test takers.

4. Analysis using Test Theory

4.1. Notation

First, let us introduce some notations. Let us consider the case in which test takers respond to problems. Problems to be solved are called *items* in psychology or psychological statistics. Let the set of test takers (learners) be I and the set of problems be J. Each test taker $i \in I$ answers item $j \in J$. Each problem can be scored in binary format: let y_{ij} be how i answers j: $y_{ij} = 1$, i.e., correctly or $y_{ij} = 0$, i.e., incorrectly.

4.2. Chronbach's Alpha

Chronbach's alpha is a measure of *test reliability*. Test reliability implies whether the test can be used to measure uni-dimensional hidden characteristics of test takers to be estimated, which we usually call "ability". Another interpretation of Chronbach's alpha is that it measures *internal consistency*. That is, it measures how well items that can be used to measure similar characteristics of test takers would result in similar response patterns of test takers.

Using the notations explained above, t is defined as follows:

$$\alpha = \frac{|J|}{|J|-1}\left(1 - \frac{\sum_{j=1}^{|J|} P_j(1-P_j)}{\sigma_X^2}\right) \qquad (1)$$

[1] http://yoehara.com/esl-vocabulary-dataset/

[2] https://www.ets.org/toeic

[3] https://www.ets.org/toefl

where σ_X is the variance of all responses, and P_j represents the proportion of correct answers to the item (i.e., problem) j.

In our dataset, the Chronbach's alpha was 0.91, which is regarded as "excellent" (George, 2011; Kline, 2013; DeVellis, 2016). This means that our dataset is highly reliable.

4.3. Item Response Theory

Item response theory (IRT) (Baker and Kim, 2004) is usually used for analyzing test-result data including language-test data. Item response theory is used like a name of a field rather than a specific models. However, a two-parameter model (**2PL**) and one-parameter model (**1PL**) are frequently used for analysis. With both models, it is assumed that each problem is independent: whether a test taker answered a problem correctly has no influence on whether he/she answered the other problems correctly. The **1PL** is sometimes called the Rasch model (Rasch, 1960).

The **2PL** is a generalization of **1PL**. It models the probability that test taker i correctly responds to a problem j in the following equation. Let σ denote the logistic sigmoid function, i.e., for $t \in \mathcal{R}; \sigma(t) = \frac{1}{1+\exp(-t)}$.

$$P(y_{ij} = 1|i, j) = \sigma\left(a_j(\theta_i - b_j)\right) \quad (2)$$

The model has two item parameters a_j and b_j, and one test-taker parameter θ_i, which is called the *ability* parameter and denotes the ability of i. Parameter b_j is called the *difficulty* parameter and denotes the difficulty of j. Since the logistic sigmoid function is a monotonously increasing function, the larger θ_i is, the more likely that i will correctly answer j. On the other hand, the larger b_j is, the less likely that i will correctly answer j. Since $\sigma(0) = 0.5$, i is more likely to respond correctly to j if and only if $\theta_i > b_j$ and vice versa.

Parameter a_j is called a *discrimination* parameter and has a more complicated definition. Briefly speaking, it shows how well j discriminates low-ability test takers from high-ability test takers. In other words, if a_j becomes larger, the difference in the probability to respond correctly between high and low test takers becomes larger.

Equation 2 can be graphically drawn. The curve showing probability (i.e., y-axis) against θ_i (i.e., x-axis) is called an *item characteristic curve* (ICC). It shows how difficult a j is for an i whose ability is the value on the x-axis. We show an example of the ICCs for some of the calculated words in Figure 1.

4.4. Test Information Function

A test information function is an important concept in IRT. It shows how reliable a test is in the form of a function against *ability*. Intuitively, a test is unreliable for test takers too skilled or un-skilled for whom the test is designed. For example, an elementary school math test cannot reliably measure the abilities of computer-science math students. Therefore, the informative-ness of the test results can be given as a function of the ability of test takers. This is the underlying idea behind the test information function. Figure 2 shows the test information function of our dataset [4] We can see that it contains information in a wide range of abilities.

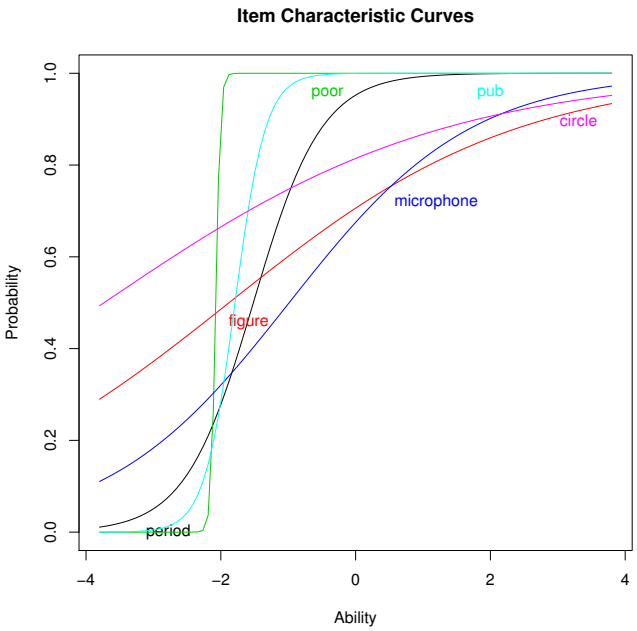


Figure 1: Examples of ICCs in our dataset

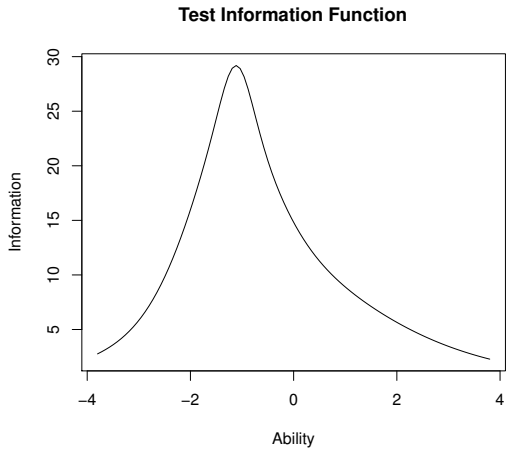


Figure 2: Test information function of our dataset

4.5. Evaluation of dataset's reliability using TOEIC scores

In this dataset, we limited test takers to those who had previously taken the TOEIC test. The TOEIC test thoroughly examines test takers's ability. It takes about 2 hours to take the test; thus, it imposes a heavy burden on test takers. The TOEIC test is mainly used as a proficiency test of English in Japan and Korea. We evaluated the reliability of our

[4] When drawing Figure 2, we removed four words that were outliers in the data, namely, "poor", "pub", "octopus", and "puma".

dataset by analyzing the correlation between the test-takers' θ_is and their TOEIC scores.

In Figure 3, we show the TOEIC score against the calculated θ_i of each test-taker. A linear-regression analysis shows that θ_i is a good estimator of TOEIC score. A TOEIC score is estimated by $86.50 \times$ ability parameter + 703.08 ($p < 0.001$).

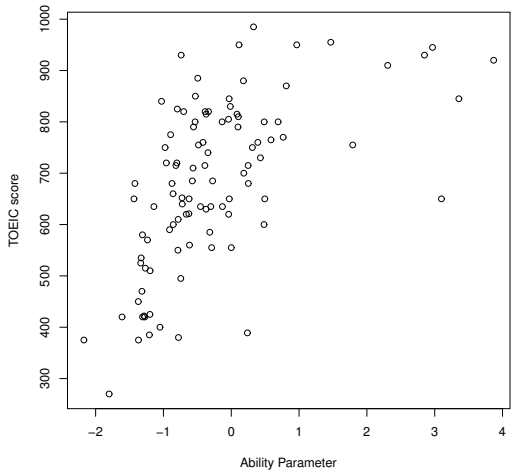


Figure 3: TOEIC scores against calculated ability parameters

A high correlation with the TOEIC score also decreases test-taker burden: taking the VST we used takes about 20 minutes. This high correlation means that we can practically measure learners' English proficiency over the Web using only 20 minutes of vocabulary testing.

5. Prediction Experiment

A developer typically needs to know the difficulty level of many words to develop educational software. For example, to support readers, the reading support system by Ehara et al. (2013) requires the difficulty parameters of the words appearing throughout a given text.

To obtain the values of the difficulty and discrimination parameters of the words in a large vocabulary set, however, testing all the words in the set imposes too much burden on the test takers, who are potential users of educational systems. A workaround for this problem is as follows: we first calculate the values of difficulty parameters and discrimination parameters for a small vocabulary set using IRT. Then, using these calculated parameters as training data, we regress the parameters with features such as *word embeddings*, or vector representations of words, to predict the parameters of the words outside this small vocabulary set.

We conducted a prediction experiment. Among the 100 words used in the vocabulary test, 92 were available in Wikipedia. We further divided these 92 words into 70 words for training and 22 words for testing. We prepared word embeddings using the word2vec toolkit (Mikolov et al., 2013) over the entire English Wikipedia. We used skip-gram as the algorithm and the size of the vectors was set at 300. To predict the parameters from the word embeddings, we applied support vector regression (SVR) with a linear and radial basis function (RBF) kernel.

As a result, the difficulty parameters were predicted with a root mean squared error (RMSE) of 3.632, and the discrimination parameters were predicted with an RMSE of 1.020 using SVR with a linear kernel. Presumably, due to the small training dataset, the SVR with an RBF kernel was overfitted, and its RMSE scores were worse than those in the case when linear kernels were used. Considering that difficulty values typically range from -5.0 to 5.0 and discrimination values range from 0.0 to 5.0, our results suggest that predicting difficulty and discrimination only from word embeddings without using data from learners is difficult. These results also suggest that our dataset is valuable because important statistics calculated from the data cannot be easily predicted from typical word features such as word embeddings.

6. Conclusion

We introduced a dataset of vocabulary-test results. Our dataset is freely available to the public and has high reliability. The calculated vocabulary ability has high correlation with the TOEIC English proficiency test ($p < 0.001$). We analyzed the reliability of our dataset using Chronbach's alpha and IRT.

We also conducted an experiment to predict the IRT parameters using word embeddings. Future work includes making this prediction more reliable.

7. Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 15K16059.

8. Bibliographical References

Baker, F. B. and Kim, S.-H. (2004). *Item Response Theory: Parameter Estimation Techniques*. Marcel Dekker, New York, second edition.

Culligan, B. (2015). A comparison of three test formats to assess word difficulty. *Language Testing*, 32(4):503–520.

DeVellis, R. F. (2016). *Scale development: Theory and applications*, volume 26. Sage publications.

Ehara, Y., Shimizu, N., Ninomiya, T., and Nakagawa, H. (2010). Personalized reading support for second-language web documents by collective intelligence. In *Proceedings of the 15th international conference on Intelligent user interfaces (IUI 2010)*, pages 51–60, Hong Kong, China. ACM.

Ehara, Y., Sato, I., Oiwa, H., and Nakagawa, H. (2012). Mining words in the minds of second language learners: learner-specific word difficulty. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, Mumbai, India, December.

Ehara, Y., Shimizu, N., Ninomiya, T., and Nakagawa, H. (2013). Personalized reading support for second-language web documents. *ACM Transactions on Intelligent Systems and Technology*, 4(2).

Ehara, Y., Miyao, Y., Oiwa, H., Sato, I., and Nakagawa, H. (2014). Formalizing word sampling for vocabulary prediction as graph-based active learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1374–1384, Doha, Qatar, October. Association for Computational Linguistics.

Ehara, Y., Baba, Y., Utiyama, M., and Sumita, E. (2016). Assessing translation ability through vocabulary ability assessment. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI*, pages 3712–3718.

George, D. (2011). *SPSS for windows step by step: A simple study guide and reference, 17.0 update, 10/e*. Pearson Education India.

Kline, P. (2013). *Handbook of psychological testing*. Routledge.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Nation, P. and Beglar, D. (2007). A vocabulary size test. 31(7):9–13.

Rasch, G. (1960). *Probabilistic Models for Some Intelligence and Attainment Tests*. Danish Institute for Educational Research, Copenhagen.

Webb, S., Sasao, Y., and Ballance, O. (2017). The updated vocabulary levels test. *ITL - International Journal of Applied Linguistics*, 168(1):33–69.

Chinese Relation Classification using Long Short Term Memory Networks

Linrui Zhang, Dan Moldovan
The University of Texas at Dallas
800 West Campbell Road ; MS EC31, Richardson, TX 75080 U.S.A
linrui.zhang@utdallas.edu, moldovan@hlt.utdallas.edu

Abstract

Relation classification is the task to predict semantic relations between pairs of entities in a given text. In this paper, a novel Long Short Term Memory Network (LSTM)-based approach is proposed to extract relations between entities in Chinese text. The shortest dependency path (SDP) between two entities, together with the various selected features in the path, are first extracted, and then used as input of an LSTM model to predict the relation between them. The performance of the system was evaluated on the ACE 2005 Multilingual Training Corpus (Walker et al., 2006), and achieved a state-of-the-art F-measure of 87.87% on six general type relations and 83.40% on eighteen subtype relations in this corpus.

Keywords: Long Short Term Memory Networks, Shortest Dependency Path, Chinese Relation Classification

1. Introduction

The task of relation classification is to predict semantic relations between pairs of entities. Formally, the goal is to predict the semantic relations between a head entity e_h and a tail entity e_t from a given sentence. For example, in the phrase "俄罗斯总统 Russian President", the relations between "Russia (ORG)" and "President (PER)" are ORG-AFF which indicate that the "President" is affiliated to an organization "Russia" , in ACE annotation ORG-AFF(Russia, President), and to be more specific, "President" is the employee of "Russia", EMPLOYMENT(Russia, President). In ACE corpus 2005, six general relations such as ORG-AFF and PART-WHOLE, and eighteen subtype relations such as OWNERSHIP, LOCATED, and EMPLOYMENT are defined.

The feature-based, kernel-based and deep learning-based methods are the most popular models for relation extraction in the literature. The basic idea of the feature-based approach is to treat relation extraction as a classification problem. Different kinds of features are extracted from text and then fed into a classifier. Such work includes (Kambhatla, 2004), (GuoDong et al., 2005) and (Moldovan and Blanco, 2012). However, these approaches suffer from error propagation problems. If the features are not well selected, the errors will be added up until the end. (Jiang and Zhai, 2007) systematically analyzed the effectiveness of different features for relation extraction on a large feature space and concluded that just using the basic unit features from each feature space (sequence, syntactic and dependency relation) can achieve reasonably good performance, and adding more complex features may not benefit the result. The kernel-based approach is to compute a kernel function to measure the similarity between two data objects. Such work includes (Zelenko et al., 2003), (Culotta and Sorensen, 2004), (Bunescu and Mooney, 2005), (Zhang et al., 2006), (Zhou et al., 2007), (Wang, 2008) and (Plank and Moschitti, 2013). The key issue of the kernel-based approach is the slow training and prediction time, so it is not good enough to process big data. With the development of deep learning, a series of neural network-based models are proposed, such as recursive neural network-based approaches (Socher et al., 2012),(Ebrahimi and Dou, 2015), and convolutional neural network-based approaches (Zeng et al., 2014), (Santos et al., 2015), and (Nguyen and Grishman, 2015).

Long short-term memory (Hochreiter and Schmidhuber, 1997) can capture long-term dependencies in sequences, so they could be used to model sequential data naturally. Recently they have been used frequently in many NLP tasks (Cho et al., 2014). Shortest dependency paths (SDP) have proven to be highly useful to relation extraction (Ebrahimi and Dou, 2015). They can to the utmost avoid involving irrelevant words in the path from one entity to another. (Xu et al., 2015) combined LSTMs and SDPs together and proposed a SDP-LSTM model for an English relation classification task on the SemEval-2012 dataset. Intrigued by this idea, we built a simplified SDP-LSTM model for Chinese relation classification on the ACE 2005 corpus and obtained state-of-the-art results. To the best of our knowledge, we are the first to implement an LSTM network for Chinese relation classification. This paper is organized as follows: We first present related work, then describe the system in detail, including the SDP and features for Chinese, and explain how to apply them to the LSTM Model. After that we present the details of our experiment, such as the corpus, the training details and the tools used. Finally, we compare our work with previous work in the literature, and show the effectiveness of different features on the final results.

2. Related Work

Research on Chinese relation extraction is quite limited compared with the progress with English. This may be due to two reasons. First, the Chinese language makes less use of function words and morphology (Levy and Manning, 2003), which makes it harder to extract syntactic information from it. Second, the lack of relevant corpora and tools also slows down the progress of research in Chinese. (Che et al., 2005) (Kebin et al., 2007) (Huang et al., 2008) (Yu et al., 2010), and (Dandan et al., 2012) proposed different kernel based approach to Chinese relation extraction. (Li et al., 2008) (Zhang et al., 2008),(Zhang et al., 2011) published a series of papers on feature-based approaches. In particular, they designed nine positional structures of enti-

ties and focused on the effectiveness of the positional feature on Chinese relation extraction. (Chen et al., 2014) proposed a novel Omni-word feature which takes advantage of Chinese sub-phrases, together with soft constraints for Chinese relation extraction, and got a 90.35% F1 score on Type relation, and a 75.44% F1 score on subType relation. Other interesting work such as (Chen et al., 2012) introduces a Deep Belief Network model that can handle high-dimensional feature space (Lin et al., 2010) worked on a mixed model that combined feature-based and kernel-based models together.

Word vector representation is the foundation of deep learning techniques for NLP. There are two popular models for word embedding: the word2vec model, which is promoted by Google (Mikolov et al., 2013), and the GloVe model, which is promoted by Stanford University (Pennington et al., 2014). Low-dimensional, dense word embeddings can effectively alleviate sparsity by sharing statistical strength between similar words, and can provide a good starting point to construct features of words and their interactions (Chen and Manning, 2014).

Long short term memory (LSTM) networks were first proposed by Hochreiter in 1997 (Hochreiter and Schmidhuber, 1997). An LSTM could be viewed as a complex activation unit that has an input gate, a forget gate, an output gate, a new memory cell and a final memory cell. Figure 1 shows the complete representation of a long short term memory unit.[1]

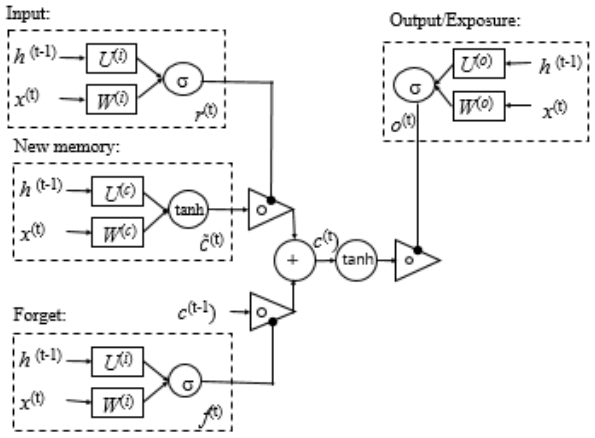


Figure 1: The complete representation of a long short term memory unit.

The mathematical formulation of LSTM units is as follows:

The input gate is to decide if the input x_t is worth being preserved based on the input word x_t and the past hidden state h_{t-1}.

$i_t = \sigma(W^{(i)}x_t + U^{(i)}h_{t-1})$

The forget gate f_t makes an assessment on whether the past memory cell is useful to compute the current memory cell.

$f_t = \sigma(W^{(f)}x_t + U^{(f)}h_{t-1})$

The output gate is to separate the final memory c_t from the hidden state.

$o_t = \sigma(W^{(o)}x_t + U^{(o)}h_{t-1})$

The new memory generation cell is used to generate a new memory $\tilde{c}_t$ by input work x_t and the past hidden state h_{t-1}.

$\tilde{c}_t = tanh(W^{(c)}x_t + U^{(c)}h_{t-1})$

The final memory cell produces the final memory c_t by summing the advice of the forget gate f_t and input gate i_t

$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t$

$h_t = o_t \circ tanh(c_t)$

3. Approach

The pipeline of the system is as follows: we first find the shortest dependency path (SDP) between two given entities in a sentence. Second, we extract the selected features along the words in the SDP. Finally, we concatenate the distributated representation of the words and their features into dense vectors, and feed them as input of LSTM models.

3.1. The Shortest Dependency Path

Dependency parsing captures the dependence relations between words, and when compared with constituency paths, dependency paths have a better ability to encode information. In dependency parsing, the dependency relations and words will form a dependency graph. The edges are the dependency relations and the vertices are the words. Finding the shortest dependency path between words may be mapped into finding the shortest path between two vertices in the dependency graph. We used Stanford CoreNLP (Manning et al., 2014) for dependency parsing, and NetworkX (Hagberg et al., 2008) to get the shortest path in the dependency graph. Figure 2 and 3 show an example to map from the dependency graph to shortest dependency path of sentence "We went home after comforting her sister (我们在安慰完妹妹后也回家了).".

3.2. Feature selection

We extracted four kinds of features: word embedding, Part-of-Speech tags, entity type, and entity subtype.

We use Google's word2vec model and trained our word vectors on Chinese Wikipedia data. Part-of-Speech tags are extracted by Stanford CoreNLP. The type and subtype of the entities are already annotated by the corpus. The ACE corpus defines seven general types of entities, and each general type could be subcategorized into subtypes. There are 44 total subtypes of entities.

All the features use distributed representations and are only applied to the words that are on the shortest dependency path. For each word, we concatenated it with all its features into a dense vector and fed it as the input of the LSTM model. Figure 4 shows the feature representation of the sentence.

3.3. Model Structure

In general, the model has three layers: one LSTM layer, one dropout layer and one softmax layer. An overview of the model is shown in Figure 5.

3.3.1. The LSTM layer

The LSTM network takes input from the data directly. The input of the LSTM network should be in three dimensions.

[1] http://web.stanford.edu/class/cs224n/lecture_notes/cs224n-2017-notes5.pdf

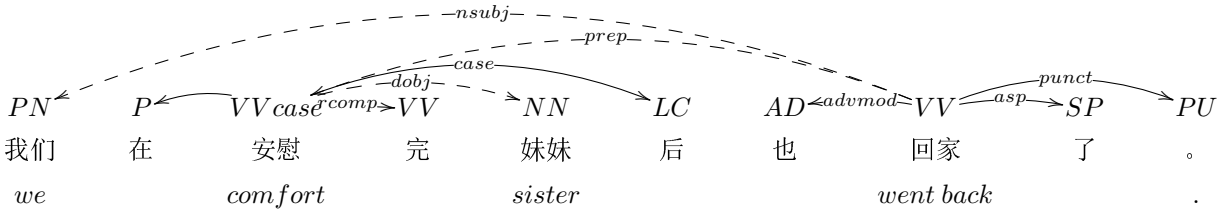


Figure 2: The dependency graph of the sentence "We went back after comforting her sister."

$$we \xleftarrow{nsubj} went\ back \xrightarrow{prep} comfort \xrightarrow{dobj} sister$$

Figure 3: The shortest dependency path between "we "and "sister ".

Word		POS		Type		SubType
We		PN		PER		$Group$
go	+	VV	+	$\emptyset$	+	$\emptyset$
$comfort$		VV		$\emptyset$		$\emptyset$
$sister$		NN		PER		$Individual$

Figure 4: feature representation of the sentence.

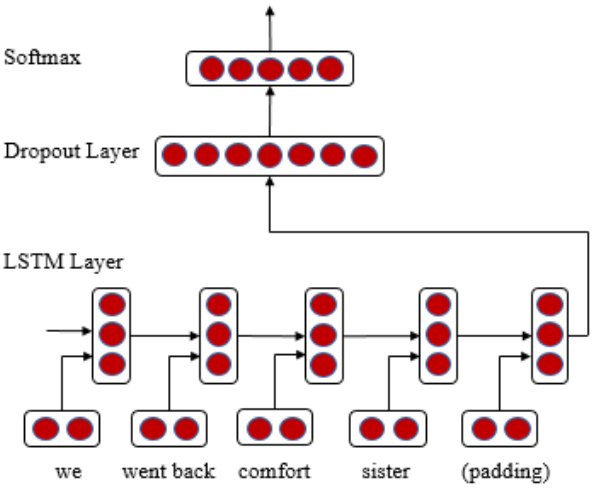


Figure 5: The strcture of our model.

The first dimension is the batch size, which is set during the experiment. The second dimension is the size of the input, which is the size of the dense vector that concatenates the word and its features. The third dimension is the time step dimension of the LSTM, which in our case is the sequence length of the Shortest Dependency Path (SDP). However, different instances have the different lengths of SDPs, so we use padding techniques that unify the SDP length of all the training examples.

3.3.2. The Dropout layer

A dropout layer is built on top of LSTM layer. It is used to reduce overfitting by randomly delete features from network during the training in order to prevent the features from excessively co-adaptation (Hinton et al., 2012) (Srivastava et al., 2014). To be more formal, a kept_prob rate of probability P is initialized. During the training process, 1- P of the features will be randomly deleted, however during the test process we do not delete any features, as a result, the testing process will have the same magnitude as the training process, and the overfitting will be prevented.

3.3.3. The softmax layer

A softmax layer is usually used as the output layer for classification problems in Deep Learning. It takes the output of the dropout layer and outputs the probability distribution of the candidate classes. In our case, the outputs are the six general type relations and the eighteen subtype relations between the entities.

3.3.4. Learning

The training goal is to minimize the cross-entropy error between the softmax layer outputs (the probabilistic distribution of the predicated relations) and the one-hot representation of the gold annotated relations. The mathematic representation of the training process is as follows:

$$E(\theta) = -\sum_{n}\sum_{k} t_n^k \log y_n^k + \frac{\lambda}{2}||\theta||^2$$

t is the gold annotated relations. y is the predicated relations from the softmax layer. λ is the regularization rate. θ is the model parameters we are trying to learn.

4. Experiments and Results

The model is implemented using Google's Tensorflow(Abadi et al., 2016). All the layers and the training object used the default packages included in Tensorflow. We did not make any changes to these packages.

4.1. Corpus and experiment details

The performance of the system was tested on the ACE 2005 training corpus Chinese data set. There are in total positive 7985 instances, and we split the corpus into 80% training and 20% testing.

We used the pretrained word vector on Chinese Wikipedia data and set the vector length to 60. Since there is no model proposed to train the Part-of-Speech tags and entity types, and they are not sparse even with one-hot representation,

Systems	features	Type F1	SubType F1
(Li et al., 2008)	Entity Type, Entity SubType 9 Position Strucute, N-grams Wordlist	70.29%	67.41%
(Zhang et al., 2011)	Entity Type, Entity SubType 9 Position Strucute, N-grams	70.43%	67.86%
(Dandan et al., 2012)	Convolution Tree Kernel	69.00%	66.60%
Ours	Entity Type, Entity SubType Word Embedding, Part-Of-Speech	**87.87%**	**83.40%**

Table 1: Performance of Relation Extraction Systems

we randomly initialized them and set the vector length to 20 for each of them, so the input size of the LSTM model is 120 dimensions. Because most relations in the ACE corpus are short-range relations, and some of them even within single noun phrases, we set the time step size to 5, and used padding to unify the sequence length. We also set the number of hidden units in the LSTM layer to 64 and kept the prob_rate on the dropout layer at 0.5.

4.2. Results

Table 2 shows the final results and effectiveness of different features to the result. From Table 2, we can conclude that all the selected features are effective and will benefit the results, and among those, word embedding is the most important feature. There is another popular feature, "positional structure" that is frequently used in previous research, and we believe our results will be further improved if we include it. This feature has been used particularly with the ACE corpus. However, modern corpus is no longer annotated with this feature, which is why we do not include it.

Systems	features	Type F1	SubType F1
LSTM	Word Embeddings	78.55%	72.92%
	+ POS	+1.63%	+1.67%
	+ Entity Type	+6.38%	+9.23%
	+ Entity SubType	+5.22%	+7.82%
	Overall	87.87%	83.40%

Table 2: The experiment results

4.3. Compare with state-of-the-art System

We compared our results with those of three previous approaches that were evaluated on the same corpus. Table 1 shows the comparison of the features/kernel use and the F1 scores of type and subtype relations between our approach and that of others. Previous works on this corpus focused on relation extraction problem. The different between relation extraction and relation classification is that relation extraction needs to detect if a relation utters corresponding to some entity pairs before predicting the relation between them. In practise, we could reduce the relation extraction problem into relation classification problem by removing negative instances (that do not have relations between entities). (Nguyen and Grishman, 2015) discovered that the system performace will improve about 20% from relation extraction to relation classification in ACE corpus. Considering this discovery, our system is still superior to the others. More importantly, our model does not combine tremendous feature engineering work, and this re-confirms its advantages.

4.4. Error analysis

Errors most frequently happen at the feature selection step. Stanford CoreNLP may make mistakes analyzing Chinese sentences. As with the entire pipeline model, it has error propagation issues.

Optimizing the feature embedding is another way to improve the results. For example, when we face a complex noun phrase that is not in the pretrained word vector, we may split it into several small elements, and then take the average of each element embedding. It is a simple strategy, but not the best. Besides, we randomly initialized the embeddings for the rest of features, however, as stated in (Chen and Manning, 2014), similar POS such as "NNS" and "NN", "VB" and "VBZ" should be clustered together and have similar embeddings, and it is also the same for entity type and subtype features.

5. Conclusion and Future Work

In this paper, we present an LSTM network with shortest dependency path model to extract relations between entities in Chinese. The results show that the model is effective on the ACE 2005 training corpus. Compared with traditional feature-based and kernel-based methods, deep learning models (neural network based methods) can easily achieve better results with fewer features.

There are several avenues for future work. The correct representation of the feature embedding is very important in deep learning models, and since Chinese is a character based language, some researchers propose that instead of using word embedding, we may use character embedding instead. In this case, we will no longer have segmentation problems, and have less feature engineering work. Since few deep learning based works have been done to Chinese, we would like to try more deep learning models and analyze their performance. Besides, our model is not very language-sensitive, so we will extend our work to other languages.

6. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Bunescu, R. C. and Mooney, R. J. (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 724–731. Association for Computational Linguistics.

Che, W., Jiang, J., Su, Z., Pan, Y., and Liu, T. (2005). Improved-edit-distance kernel for chinese relation extraction. In *Proceedings of IJCNLP*, pages 132–137.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Emnlp*, pages 740–750.

Chen, Y., Zheng, D.-Q., and Zhao, T.-J. (2012). Chinese relation extraction based on deep belief nets. *Ruanjian Xuebao/Journal of Software*, 23(10):2572–2585.

Chen, Y., Zheng, Q., and Zhang, W. (2014). Omni-word feature and soft constraint for chinese relation extraction. In *ACL (1)*, pages 572–581.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Culotta, A. and Sorensen, J. (2004). Dependency tree kernels for relation extraction. In *Proceedings of the 42nd annual meeting on association for computational linguistics*, page 423. Association for Computational Linguistics.

Dandan, L., Yanan, H., and Longhua, Q. (2012). Exploiting lexical semantic resource for tree kernel-based chinese relation extraction. *Natural Language Processing and Chinese Computing*, pages 213–224.

Ebrahimi, J. and Dou, D. (2015). Chain based rnn for relation classification. In *HLT-NAACL*, pages 1244–1249.

GuoDong, Z., Jian, S., Jie, Z., and Min, Z. (2005). Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

Hagberg, A., Swart, P., and S Chult, D. (2008). Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL).

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huang, R., Sun, L., and Feng, Y. (2008). Study of kernel-based methods for chinese relation extraction. *Information Retrieval Technology*, pages 598–604.

Jiang, J. and Zhai, C. (2007). A systematic exploration of the feature space for relation extraction. In *HLT-NAACL*, pages 113–120.

Kambhatla, N. (2004). Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 22. Association for Computational Linguistics.

Kebin, L., Fang, L., Lei, L., and Ying, H. (2007). Implementation of a kernel-based chinese relation extraction system [j]. *Journal of Computer Research and Development*, 44(8):1406–1411.

Levy, R. and Manning, C. (2003). Is it harder to parse chinese, or the chinese treebank? In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 439–446. Association for Computational Linguistics.

Li, W., Zhang, P., Wei, F., Hou, Y., and Lu, Q. (2008). A novel feature-based approach to chinese entity relation extraction. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 89–92. Association for Computational Linguistics.

Lin, R., Chen, J., Yang, X., and Xu, H. (2010). Research on mixed model-based chinese relation extraction. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 1, pages 687–691. IEEE.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Moldovan, D. I. and Blanco, E. (2012). Polaris: Lymba's semantic parser. In *LREC*, pages 66–72.

Nguyen, T. H. and Grishman, R. (2015). Relation extraction: Perspective from convolutional neural networks. In *VS@ HLT-NAACL*, pages 39–48.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Plank, B. and Moschitti, A. (2013). Embedding semantic similarity in tree kernels for domain adaptation of relation extraction. In *ACL (1)*, pages 1498–1507.

Santos, C. N. d., Xiang, B., and Zhou, B. (2015). Classifying relations by ranking with convolutional neural networks. *arXiv preprint arXiv:1504.06580*.

Socher, R., Huval, B., Manning, C. D., and Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1201–1211. Association for Computational Linguistics.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way

to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Walker, C., Strassel, S., Medero, J., and Maeda, K. (2006). Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57.

Wang, M. (2008). A re-examination of dependency path kernels for relation extraction. In *IJCNLP*, pages 841–846.

Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., and Jin, Z. (2015). Classifying relations via long short term memory networks along shortest dependency paths. In *EMNLP*, pages 1785–1794.

Yu, H., Qian, L., Zhou, G., and Zhu, Q. (2010). Chinese semantic relation extraction based on unified syntactic and entity semantic tree. *Journal of Chinese Information Processing*, 24(5):17–23.

Zelenko, D., Aone, C., and Richardella, A. (2003). Kernel methods for relation extraction. *Journal of machine learning research*, 3(Feb):1083–1106.

Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al. (2014). Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344.

Zhang, M., Zhang, J., Su, J., and Zhou, G. (2006). A composite kernel to extract relations between entities with both flat and structured features. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 825–832. Association for Computational Linguistics.

Zhang, P., Li, W., Wei, F., Lu, Q., and Hou, Y. (2008). Exploiting the role of position feature in chinese relation extraction. In *LREC*.

Zhang, P., Li, W., Hou, Y., and Song, D. (2011). Developing position structure-based framework for chinese entity relation extraction. *ACM Transactions on Asian Language Information Processing (TALIP)*, 10(3):14.

Zhou, G., Zhang, M., Ji, D.-H., and Zhu, Q. (2007). Tree kernel-based relation extraction with context-sensitive structured parse tree information. In *EMNLP-CoNLL*, volume 2007, pages 728–736.

The UIR Uncertainty Corpus for Chinese: Annotating Chinese Microblog Corpus for Uncertainty Identification from Social Media

Binyang Li[1], Jun Xiang[1], Le Chen[1], Xu Han[2, 7*], Xiaoyan Yu[2], Ruifeng Xu[3], Tengjiao Wang[4], Kam-fai Wong[5, 6]

[1] School of Information Science and Technology, University of International Relations
[2] College of Information Engineering, Capital Normal University
[3] Harbin Institute of Technology, Shenzhen Graduate School
[4] School of Electronics Engineering and Computer Science, Peking University
[5] Systems Engineering and Engineering Management, The Chinese University of Hong Kong
[6] MoE Key Lab on High Confidence Software Technologies CUHK Sub-Lab
[7] Collaborative Innovation Center of Mathematics and Information Science, Capital Normal University
{byli, jxiang, lchen}@uir.edu.cn; {hanxu, xyyu}@cnu.edu.cn; xuruifeng@hit.edu.cn; kfwong@se.cuhk.edu.hk

Abstract

Uncertainty identification is an important semantic processing task, which is critical to the quality of information in terms of factuality in many NLP techniques and applications, such as question answering, information extraction, and so on. Especially in social media, the factuality becomes a primary concern, because the social media texts are usually written wildly. The lack of open uncertainty corpus for Chinese social media contexts bring limitations for many social media oriented applications. In this work, we present the first open uncertainty corpus of microblogs in Chinese, namely, the UIR Uncertainty Corpus (UUC). At current stage, we annotated 40,168 Chinese microblogs from Sina Microblog. The schema of CoNLL 2010 have been adapted, where the corpus contains annotations at each microblog level for uncertainty and 6 sub-classes with 11,071 microblogs under uncertainty. To adapt to the characteristics of social media, we identify the uncertainty based on the contextual uncertain semantics rather than the traditional cue-phrases, and the sub-class could provide more information for research on handing uncertainty in social media texts. The Kappa value indicated that our annotation results were substantially reliable.

Keywords: Chinese Microblog, Uncertainty Annotation, UIR Uncertainty Corpus

1. Introduction

"Uncertainty - in its most general sense - can be interpreted as lack of information: the receiver of the information (i.e., the hearer or the reader) cannot be certain about some pieces of information" (Szarvas et al. 2012). Many natural language processing problems are required to be solved based on the veracity of information, including question answering, information extraction, and so on, are extraordinarily need the uncertainty identification technique (Li et al., 2014).

Most participants of the CoNLL-2010 Shared Task, aiming at identifying uncertain expressions from the biological papers and Wikipedia articles, utilized linguistics features, such as lexical cues, Part-Of-Speech (POS), to detect the uncertain sentences from the given texts. (Farkas et al., 2010).

Nowadays, with the advancement of social media, there are more and more text contents containing a large amount of casual or word-of-mouth expressions. The quality of information in social media in terms of factuality has become a primary concern (Wei et al. 2013). Generally speaking, only the factual information with high credibility has the value for use. However, most expressions in social media are published with the hypothesis and episteme which cannot be decided whether it is true or false at that moment. Some previous studies investigated the existence of uncertain statements and found that they were widespread among social media, e.g., Facebook, Twitter, Sina Weibo, etc. According to the statistic of tweet dataset (Wei et al., 2013), there were at least 18.91% of tweets possess uncertainty characteristic. As a result, the quality of information in social media has become a primary concern for many social media oriented tasks, such as rumor detection (Qazvinian et al., 2011) and credibility evaluations (Jaworski et al., 2014). Therefore, uncertainty identification is significant for users to synthesize information to derive reliable interpretation.

To do this, there are a few such corpora: the primal corpora are BioScope corpus annotating uncertainty and their scope in biological papers and the Wikipedia dataset for CoNLL 2010 shared task (Vincze, 2010).

Similar work for Chinese has been reported by Ji (2010), who constructed the Chinese corpus from newspapers for Chinese uncertainty identification. However, their data was annotated exactly based on the cue-phrases and size of corpus was a bit small for some uncertainty detection systems. In addition, Chinese uncertain cue phrases were not always indicating the uncertainty of sentences. For instance, "好像(*likely*)" could express uncertain with the *Possible* sub-class of uncertainty, or regard as metaphor implied certainty based on the implicit semantics.

More importantly, unlike the biological papers and Wikipedia articles, the texts in social media are usually short and informal. Due to the word count limit and casual expression, many cue phrases are expressed in substandard shape or even omitted from sentences. In this case, the uncertain semantics will be implicitly conveyed by the whole sentence rather than by the explicit cue phrases.

* Corresponding Author

In this work, we target on annotating social media texts for uncertainty identification in Chinese language. Our corpus was collected from Sina Microblogs (also referred as Sina Weibo) during the 2010 Shanghai World Expo. After removing error and invalid sentences, 11,071 out of 40,168 microblogs were annotated as uncertain.

To summary, this paper has the following contributions:

(1) we are the first to annotate a Chinese social media corpus, namely the UIR Uncertainty Corpus (UUC), for uncertainty identification task;

(2) UUC corpus is constructed by 40,168 Chinese Sina Microblogs and 11,071 were labeled uncertain following our proposed scheme for Chinese with the *Kappa* value indicating substantially reliable;

(3) UUC corpus cannot only contained certainty or uncertainty classification but also annotated six sub-categories of uncertainty as *Possible*, *Condition*, *Doxastic*, *Hope*, *External*, *Question*.

The remainder of this paper is organized as follows: Section 2 gives the related work and illustrate the differences between this work and the previous work. Section 3 details the annotation scheme of uncertainty in social media texts. Section 4 presents the annotation results and some important statistics of the corpus. Section 5 concludes this work.

2. Related Work

Most of the current uncertainty annotation work followed the task of uncertainty identification in CoNLL 2010, in which the uncertainty texts were annotated based on the cue-phrases. It is because that the uncertainty cues commonly exist in uncertain sentences and express the . It is true that the cue words are an efficient indicator for uncertainty, but these cues are usually invalid for Chinese in social media context. To identify uncertainty clearly, our first priority is to figure out the types of uncertainty in microblogs. Kiefer (2005) pointed out that uncertainty can be divided into *Epistemic* and *Hypothetical*. *Epistemic* contains *Possible* and *Probable*. In fact, the two sub-classes are fairly similar in Chinese. Wei (2013) observed *Question* and *External* frequently appeared on these posts or comments which reveals uncertainty. Considering *Dynamic*, one sub-class of *Hypothetical*, hardly existing in Chinese microblogs, we removed this label in our annotation scheme.

At present, there are several corpora in different domains:

(1) BioScope corpus (Vincze et al., 2008) annotated uncertainty, negation sentences and their scope in biomedical texts containing 20,879 sentences from 3,236 documents.

(2) the dataset for CoNLL'2010 shared task (Vincze, 2010) consisted of biological part of BioScope corpus and the selection of Wikipedia articles, which annotated uncertain sentences and cues.

(3) Uncertainty Corpus in complex spoken dialogue systems derived from 120 digital dialogues recording from 60 students, totaling 2,171 turns for students and 2,531 turns for tutor.

(4) The Scientific Literature Corpus for Chinese (Chen et al., 2013) including 19 full papers annotated negation, uncertainty and their cues as well as scope.

(5) Feng Ji (2010) presented a corpus annotated uncertain cues and sentences about 50 news containing 8,000 sentences.

Although in different fields have existed uncertainty corpora, there is no standard uncertain corpus for Chinese in social media texts. The expression of uncertainty in social media is fairly different from that in formal text in a sense. Therefore, our study seeks to fill this gap by presenting the Sina Weibo corpus, which consists of 40,168 microblogs annotated for uncertainty and their sub-classes.

3. Annotation Scheme

3.1 Types of Uncertainty in Chinese Microblogs

The annotation scheme adopted in this work refer to the uncertainty categories of tweets by Wei, et al. (2013). However, due to the different habits or uses of languages between English and Chinese, there are some adjustments in our annotation scheme. Generally, uncertainty of the sentences can be divided into *Epistemic* and *Hypothetical*. For *Epistemic*, there are two sub-classes *Possible* and *Probable*. Compared to the Chinese uncertainty corpora, we found that the sub-classes of *Possible* and *Probable* are very similar in Chinese uncertainty expressions. These two sub-classes are merged as one class *Possible*. In addition, we also found that of all the microglobs there was no expressions belonging to the sub-class of *Dynamic* in Chinese social media. So, we also remove the sub-class of *Dynamic*. On the contrary, we found that in Sina microblogs, there were a lot of sentences expressing the uncertainty statement with expected semantics.

Base on the above observations, we propose a variant of uncertainty sub-class in Chinese social media corpus by removing the *Dynamic* type and adding the sub-class of *Hope*, as shown in Figure 1.

We modify and redefine the uncertainty scheme into six sub-classes as follows:

- *Question*: Sentences are used to ask for information or to test someone's knowledge.
- *External*: Speaker repeat exactly what someone else has said or written.
- *Doxastic*: Expresses the speaker's beliefs and hypotheses.
- *Hope*: Expresses the speaker want something to happen or be true and to believe that it is possible or likely.
- *Possible*: Expresses something it can be done or achieved.
- *Condition*: Proposition under condition.

Examples of each sub-class of uncertain sentences are selected from the corpus:

(1) *Doxastic*

我相信，华夏文艺复兴正在走来。

I believe that the Chinese Renaissance is coming.

(2) *Question*

网游客户会对媒体品质有影响吗？

Will online visitors have an impact on media quality?

(3) *External*

据说那个叫苏琴的人已经当了年级组长

It is said that the person who called Suqin has been a grade leader

(4) *Hope*
希望我们 302 宿舍以后可以这样，嘿嘿
Hope our 302 dormitory later can be so, heihei
(5) *Condition*
如果这是真的，不忍想象他们心中藏着怎样的悲痛。
If this is true, can not bear to imagine how the hearts of their hidden grief.
(6) *Possible*
今早一饭四菜，好像做多了
Four dishes this morning, it seems to do more

3.2 Annotation Process

During our annotation process, 3 annotators were trained to annotate the dataset independently. Given a collection of microblogs $M=\{m_1, m_2, m_3, m_4, \ldots m_n\}$, the annotation task is to label each microblog m_1 as either uncertain or certain.

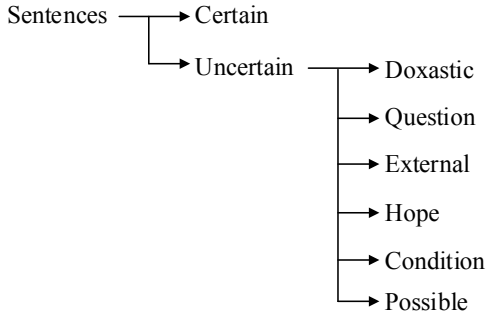


Figure 1: The annotation scheme for Chinese uncertainty annotation in social media.

Moreover, the uncertainty assertions are to be identified in terms of the judgments about author's intended meaning or the implicit semantics rather than the presence of uncertain cue-phrase. The sub-classes are also required to be further labeled according to the scheme as shown in Figure 1.

4. Results

The corpus we annotated was collected from Sina Weibo during the 2010 Shanghai World Expo. Currently, we have already annotated 40,168 Chinese microblogs after removing error and invalid sentences, in which we found 11,071 uncertain microblogs. This annotation work made out the UIR uncertainty corpus (UUC) in Chinese towards social media texts. Compared to the previous work on Chinese uncertainty corpus, UUC is the first Chinese corpus for uncertainty identification in social media, and the quantity of annotation is much larger than other Chinese uncertainty corpus in newspaper (Ji et al., 2010). Therefore, we can conclude some characteristics of uncertain expressions in Chinese and we further annotated six sub-class labels under uncertainty as shown in Table 1.

The Kappa coefficient (Carletta, 1996) indicating inter-annotator agreement was 85.68% for the certain/uncertain binary classification which means our results were substantially reliable. The disagreements in annotation work mainly come from the conflict understanding to certain ambiguous phrases, e.g., "有木有" (did you…?), "如果" (if), and we also list the specific statistics of the agreement in each sub-class, as shown in Table 2.

Refer to Table 1, there are 11,071 out of 40,168 microblogs are labeled as uncertain accounting for 27.56%. *Question* is the uncertainty category with the most microblogs, followed by *Possible*. However, only 117 microblogs are labeled as *External*. We argued that there might be other uncertain sub-classes in such microblogs apart from *External*. But only can we select one more obvious label between these for these microblogs, which might replace *External*.

As illustrated in Table 2, most of the annotation results were considerably consistent. However, the *Kappa* coefficients for *External* and *Doxastic* indicated lower than the others since the amount of microblogs for *External* and *Doxastic* was smaller compared with others. Besides, our scheme is designed for single label annotation, there exists the situation that more than one label should be annotated for the same microblog.

Microblog#	40,168
Certainty#	29,097
Uncertainty#	11,071
Possible#	2,973
Condition#	1,029
Doxastic#	470
Hope#	612
External#	117
Question#	5,870

Table 1: Statistics of annotation results

Certainty/Uncertainty	0.8568
Possible#	0.9141
Condition#	0.9651
Doxastic#	0.8502
Hope#	0.8804
External#	0.8572
Question#	0.9237

Table 2: The *Kappa* value statistics for each sub-class

Corpus	UUC	Ji's
Total #	40,168	10,000
Uncertain#	11,071	2,858
Without Cue#	2,466	23
Percentage	22.28%	0.8%

Table 3: Comparison between UUC and Ji's corpus on cue phrases percentage

Since our argument for the annotation in social media is that the word count limit and casual expression, many cue phrases are expressed in substandard shape or even omitted from sentences. We also list the statistics between two Chinese corpus for uncertainty identification, our UUC and Ji's (2010). Note that among 2,858 uncertain sentences in the Ji's corpus, there are only 23 uncertain sentences (0.8%) not containing cue phrases. On social media dataset, however, 22.28% uncertain sentences are not containing cue phrases. So the statistics also prove our assumption.

5. Conclusion and Future Work

In this paper, we have presented an uncertain annotation scheme for Chinese microblogs and the first open Chinese uncertainty corpus in social media texts. 11,071 out of 40,168 Chinese Sina Microblogs were labeled uncertain following our proposed scheme with the *Kappa* value over 0.85 that indicated substantially reliable. Specifically, we

define six sub-classes for the uncertain classification which made the corpus more detailed and useful for other related NLP applications in the future.

In the future, we are planning to annotate more uncertain microblogs about the hot issue for the development of social media applications, such as rumor detection, user profile, etc.

6. Acknowledgements

This paper is funded by the National Natural Foundation of China 61502115, 61602326, U1636103, U1536207, 61572043, 61672361, 61632011, the Hong Kong Applied Science and Technology Research Institute Project 7050854, the Fundamental Research Fund for the Central Universityies 3262015T70, 3262017T12, and the Shenzhen Foundational Research Funding 20170307150024907. We would also like to thank Nuo Chen, Ning Jiang, Lingchen Lian, Kaimin Zhou, and Haotian Ren for their work on corpus.

7. References

Castor, A. and Pollux, L. E. (1992). The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

Grandchercheur, L.B. (1983). Vers une modélisation cognitive de l'être et du néant. In S.G Paris, G.M. Olson, & H.W. Stevenson (Eds.), *Fondement des Sciences Cognitives*. Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 6-38.

Strötgen, J. and Gertz, M. (2012). Temporal tagging on different domains: Challenges, strategies, and gold standards. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), pages 3746–3753, Istanbul, Turkey, may. European Language Resource Association (ELRA).

Superman, S., Batman, B., Catwoman, C., and Spiderman, S. (2000). Superheroes experiences with books. The Phantom Editors Associates, Gotham City, 20th edition.

Vincze, V. (2016). Detecting Uncertainty Cues in Hungarian Social Media Texts.

Wei, Z., Chen, J., Gao, W., Li, B., Zhou, L., & He, Y., et al. (2013). An Empirical Study on Uncertainty Identification in Social Media Context. Meeting of the Association for Computational Linguistics (pp.58-62).

Qazvinian, V., Rosengren, E., Radev, D. R., & Mei, Q. (2011). Rumor has it: identifying misinformation in microblogs. Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John Mcintyre Conference Centre, Edinburgh, Uk, A Meeting of Sigdat, A Special Interest Group of the ACL (pp.1589-1599). DBLP.

Jaworski, W., Rejmund, E., & Wierzbicki, A. (2014). Credibility Microscope: Relating Web Page Credibility Evaluations to Their Textual Content. Ieee/wic/acm International Joint Conferences on Web Intelligence (Vol.1, pp.297-302). IEEE Computer Society.

Vincze, V., Szarvas, G., Farkas, R., Móra, G., & Csirik, J. (2008). The bioscope corpus: biomedical texts annotated for uncertainty, negation and their scopes. Bmc Bioinformatics, 9(Suppl 11), S9-S9.

Vincze V. The CoNLL-2010 shared task: learning to detect hedges and their scope in natural language text[C]// Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2010:1-12.

Feng Ji, Xipeng Qiu, Xuanjing Huang. (2010).Exploring uncertainty sentences in Chinese. In The 6th China Conference on Information Retrieval

Kiefer, F. (2005). Lehetoseg es szuksegszeruseg [Possibility and necessity]. Tinta Kiadó, Budapest.

Chen, Z., Zou, B., Zhu, Q., & Li, P. (2013). The Scientific Literature Corpus for Chinese Negation and Uncertainty Identification. Chinese Lexical Semantics. Springer Berlin Heidelberg.

Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. Computational Linguistics, cmp-lg/9602004(2), págs. 249-254.

EventWiki: A Knowledge Base of Major Events

Tao Ge[1,2], Lei Cui[2], Baobao Chang[1], Zhifang Sui[1], Furu Wei[2], Ming Zhou[2]
[1]MOE Key Laboratory of Computational Linguistics, Peking University, Beijing, China
[2]Microsoft Research Asia, Beijing, China
{tage, lecu, fuwei, mingzhou}@microsoft.com, {chbb, szf}@pku.edu.cn

Abstract

This paper introduces a new resource called *EventWiki* which is, to the best of our knowledge, the first knowledge base resource of major events. In contrast to most existing knowledge bases that focus on static entities such as people, locations and organizations, our *EventWiki* concentrate on major events, in which all entries in *EventWiki* are important events in mankind history. We demonstrate that *EventWiki* is a very useful resource for information extraction regarding events in Natural Language Processing (NLP), knowledge inference and automatic knowledge base construction.

Keywords: EventWiki, event knowledge base, major event, resource, event-driven, information extraction, knowledge base construction, knowledge inference

1. Introduction

Knowledge bases (KBs) are large collections of facts about entities such as people, organizations and locations, which have attracted growing interest in recent years because they are useful in various studies and applications such as question-answering, semantic parsing, information extraction and knowledge inference.

Most existing knowledge bases (e.g., Freebase) focus on static entities (e.g., people, locations, and organizations) and their relations. Few of them, however, pay much attention to events and event-related knowledge; instead, they just treat events as normal entities and do not specially distinguish them, which makes it hard to identify event entries. Even though some facts in KBs can be treated as events, they tend to entity-centric and insufficient to describe major and complicated events (e.g., *2011 Tohoku earthquake* and *2014 World Cup*). In fact, event knowledge is very important for many applications and its related tasks (e.g., event extraction and event detection) have been extensively studied in both NLP and Data mining community in the past decades. Unfortunately, due to the lack of rich event-related knowledge and annotations, studies related to event knowledge discovery are either limited (e.g., in event extraction evaluation under Automatic Content Extraction project[1], only events of 33 predefined event types are studied) or shallow (e.g., clustering-based event detection).

To provide rich event knowledge and a dataset for event-related studies, this paper introduces a new knowledge base resource – *EventWiki* is a knowledge base of events, which collects 21,275 major events of 95 types from Wikipedia[2]. We will show in the following section that *EventWiki* contains rich information about events (e.g., event taxonomy, event schemas and event relations), which makes it useful for a variety of studies and applications such as information extraction, knowledge inference and knowledge base construction. To the best of our knowledge, *EventWiki* is the first knowledge base of events which covers such large numbers of major event entries.

[1]http://projects.ldc.upenn.edu/ace/

[2]https://en.wikipedia.org

2. EventWiki

Compared to static entities (e.g., person, location and organization), there is less work focusing on event knowledge discovery due to a lack of event-centric knowledge bases. Although Wikipedia and some existing knowledge bases (e.g., DBPedia) contains information of major events, there is no explicit indicator that indicates whether a page refers to an event or an static entity (e.g. a person), or event-event relation links, which makes it difficult to distinguish these events from overwhelming static entities and directly use these resources for event-related knowledge discovery.

For better studying event knowledge, we manually identify event pages from Wikipedia, build links between events to develop an event knowledge called *EventWiki*. As introduced before, entries in *EventWiki* are all major events that happened throughout mankind history. For an event entry, there are 4 kinds of information in *EventWiki*: event type, event infobox, event summary and full text description of the event. In addition, for some events, there are relations between them. Table 1 shows the structure of an event entry where $e \in E$ is an event entry in *EventWiki*, T is the set of 95 event types, and $rel_i : e_{k_i}$ means that e are related with e_{k_i} and their relation is rel_i. A concrete example is shown in Figure 1 that presents the basic structure of *EventWiki* where event information and event-event relations are clearly illustrated.

Event: e	
event type	$t \in T$
infobox	$\{slot_1$:$value_1$, $slot_2$: $value_2, \cdots\}$
summary	text
full text	text
relation	$\{rel_1$: e_{k_1}, rel_2: $e_{k_2}, \cdots\}$

Table 1: Structure of an event entry.

In this section, we introduce our resource – *EventWiki* as follows: we first introduce meta-data (i.e., event types and event schema), and then present the information of an event

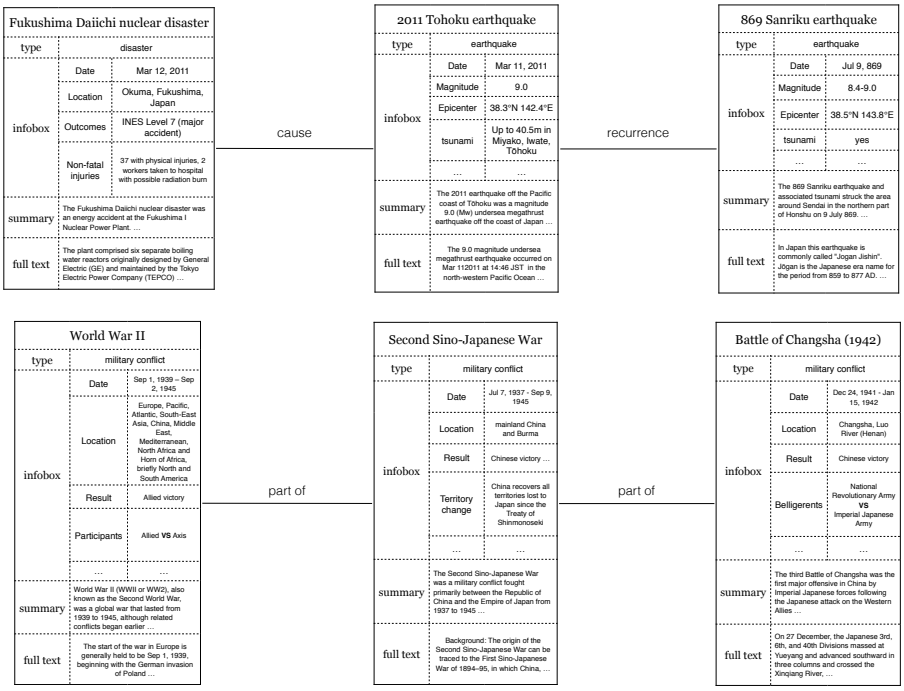


Figure 1: A (small) part of events in *EventWiki*, showing the basic schema of *EventWiki*.

entry in detail in *EventWiki* and the application of this resource.

2.1. Meta-data: Event types and schemas

As meta-data for describing event information, event types and schemas are very important information in many applications such as event extraction. However, there is no large amounts of annotated data for event typing or event schema extraction. As a result, most studies regarding event extraction are either limited (only focused on a small number of pre-defined event types) or shallow (just consider a verb as an event).

In *EventWiki*, there are 95 event types in total, derived from infobox's template types in Wikipedia by manually screening from English Wikipedia dump on December 3, 2015. Every event entry in *EventWiki* belongs to one of these 95 event types.

For each event type, we extract the schema of this event type by finding out the most frequent non-empty slots from the infobox of events belonging to this type. The most frequent slots of an event type can be used to describe key information of an event of this type and thus can be considered as the schema of the event type. Table 2 shows the schemas of 6 event types.

With the information, it is easy to learn a classifier for event classification and event schema extraction. For event classification, each event entry in *EventWiki* with its event type can be treated as training data. By using such information, it is possible to train an event classification model which can be used to classify events reported in news articles every day for further event knowledge discovery. Likewise, *EventWiki* provides valuable resources for event schema extraction. For events that belong to one of 95 types in *EventWiki*, the event schema of the event type in *EventWiki* can be directly used as the schema for event extraction. Even if an event does not belong to any of those 95 types, we still can use rich information in *EventWiki* to infer its event schema. For example, text information and slots in the infobox of an event entry in *EventWiki* can be used as training data to learn what slots an event should have given its text description. For example, a model may learn from *EventWiki* that if "magnitude" frequently appear in text description of an event, "magnitude" should be one slot of this event in its schema.

2.2. Events in *EventWiki*

As mentioned before, each event entry has 4 kinds of information (i.e., event type, event infobox, event summary and full text description) and event-event relation information. We call those 4 kinds of information intra-event information because they describe the information of an event while we call event-event relation inter-event information since it presents relations between events.

2.2.1. Intra-event information

In this section, we introduce the intra-event information in detail.

- Event type: event type presents the type of an event which must be one of 95 event types. In *EventWiki*, an event's type is extracted by analyzing the template type of the first infobox of its Wikipedia ariticle.
- Event infobox: as most static entities in Wikipedia, events also have infoboxes which describe most important information of events. We extract events' infoboxes from Wikipedia. Note that slots (i.e., attributes) in an event's infobox depends on the event's type.
- Event summary: a summary of an event is a short text that summarizes the event, which is obtained by ex-

aircraft accident	olympic event	wildfire	military conflict	earthquake
date	date	date	place	date
site	teams	location	date	location
occurrence_type	gold	fatalities	combatants(1,2)	magnitude
crew	silver	area	result	depth
fatalities	bronze	source	partof	casualties
operator	venue	injuries	commanders(1,2)	countries affected
passengers	competitors	building	strength(1,2)	origintime
origin	nations	landuse	casaulties(1,2)	tsunami
destination	win_value	cost	territory	intensity

Table 2: Examples of schemas of 6 event types

tracting the first paragraph of an event's text description at Wikipedia.

- Full text description: In contrast to event summary, full text description contains all text information in an event page at Wikipedia.

2.2.2. Inter-event information

Inter-event information is relations between events. For example, in Figure 1, we can see the relation between *Fukushima Daiichi nuclear disaster* and *2011Tohoku earthquake*.

For the current version *EventWiki*, inter-event information is mainly obtained by analyzing events' infoboxes in which some event-event relations are presented. For example, by analyzing the infobox of *Second Sino-Japanese War*, we can extract the relation *partof(Second Sino-Japanese War, World War II)*. In total, we extracted 9,713 event-event relations from infoboxes.

In addition to infoboxes of events, many event-event relations are described by text. For example, the relation between *2011 Tohoku earthquake* and *869 Sanriku earthquake* is expressed by the following text in *2011 Tohoku earthquake*:

> This megathrust earthquake was a recurrence of the mechanism of the earlier *869 Sanriku earthquake*, which has been estimated as having a magnitude of at least 8.4 Mw, which also created a large tsunami that inundated the Sendai plain.

However, it is difficult to automatically obtain such event-event relations in high accuracy from text. Fortunately, it is possible to exploit structure of a page at Wikipedia to help harvest event-event relations, as shown in Figure 2. In this way, we extracted additional 540 event-event relations.

2.3. Applications

We show that event information in *EventWiki* can be very useful for the following tasks and applications:

2.3.1. Event extraction

Event extraction is an important information extraction task. In this paper, we model event extraction as infobox generation problem for an event. Intuitively, an event infobox contains key information of an event. If an event infobox can be accurately generated, we can say that the information of an event is perfectly extracted. Usually, event infobox generation problem can be divided into three subtasks: event classification, event schema extraction and slot filling. As mentioned before, *EventWiki* can provide rich information for event classification and event schema extraction. Moreover, it is extremely useful for training a slot filling model for event extraction.

Slot and value pairs in the infobox (intra-event information) in *EventWiki* can be used as weak (distant) supervision for training a slot filling model, as (Reschke et al., 2014) did. For example, for the slot value pair *"magnitude: 9.0"* in *2011 Tohoku earthquake*, we first find out the sentences which *"9.0"* appears in. The context information of *"9.0"* can be used as features and *"magnitude"* is used as the label of *"9.0"* for training a slot filling model.

2.3.2. Event-event relation extraction and inference

As slot filling for event extraction, we can also use inter-event information in *EventWiki* to train an event-event relation extraction model using distant supervision strategy. For an event relation triple, we can find out the sentences that mention both events in the triple. For example, for the triple *partof(Second Sino-Japanese War, World War II)*, the following sentence is found for generating training instance for relation extraction:

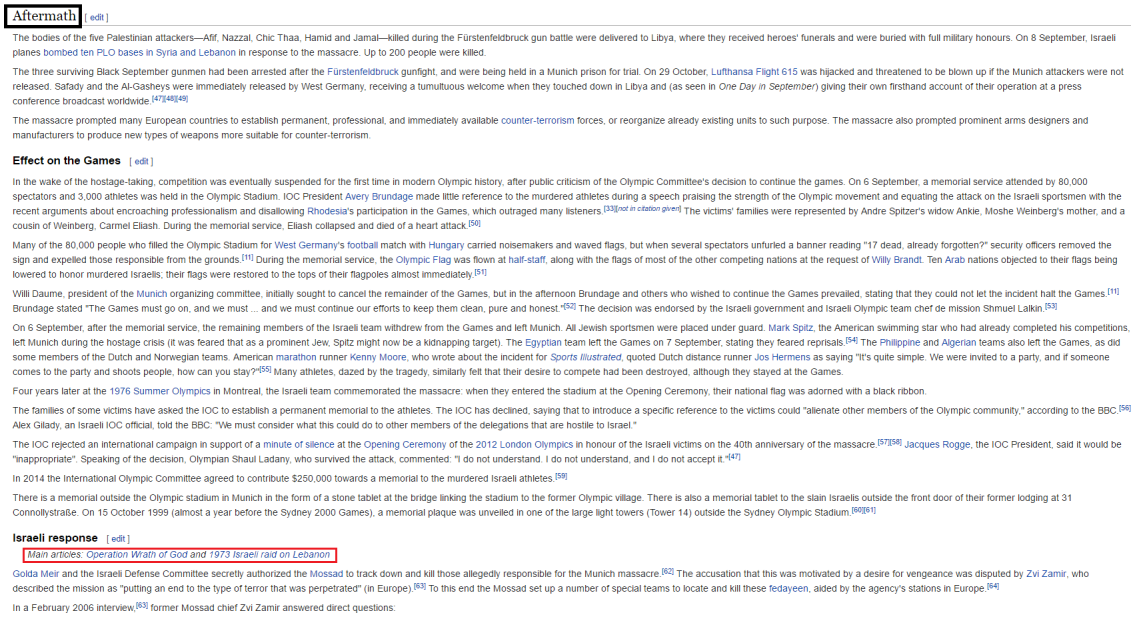
> As the Western Allies entered the war against Japan, *the Sino-Japanese war* would become part of a greater conflict, the Pacific theatre of *World War II*. (label: *partof*)

To the best of our knowledge, even though relation extraction has been extensively studied in NLP, most of them focus on relations between static entities (e.g., people and locations) and there are very few resources for event-event relation extraction. *EventWiki* can plug the gap and make it possible to study event-event relation extraction problem.

With event-event relations, it is also possible to apply knowledge inference approaches to *EventWiki* to infer event-event relations, which is a novel task and has not been well studied. For example, with *partof(Second Sino-Japanese War, World War II)* and *partof(Battle of Changsha, Second Sino-Japanese War)*, we can infer *partof(Battle of Changsha, World War II)*, which would be for useful for building event taxonomy.

2.3.3. Event summary generation

EventWiki is a good resource for event summary generation. As introduced before, each event entry in *EventWiki*



Aftermath [edit]

The bodies of the five Palestinian attackers—Afif, Nazzal, Chic Thaa, Hamid and Jamal—killed during the Fürstenfeldbruck gun battle were delivered to Libya, where they received heroes' funerals and were buried with full military honours. On 8 September, Israeli planes bombed ten PLO bases in Syria and Lebanon in response to the massacre. Up to 200 people were killed.

The three surviving Black September gunmen had been arrested after the Fürstenfeldbruck gunfight, and were being held in a Munich prison for trial. On 29 October, Lufthansa Flight 615 was hijacked and threatened to be blown up if the Munich attackers were not released. Safady and the Al-Gasheys were immediately released by West Germany, receiving a tumultuous welcome when they touched down in Libya and (as seen in *One Day in September*) giving their own firsthand account of their operation at a press conference broadcast worldwide.[47][48][49]

The massacre prompted many European countries to establish permanent, professional, and immediately available counter-terrorism forces, or reorganize already existing units to such purpose. The massacre also prompted prominent arms designers and manufacturers to produce new types of weapons more suitable for counter-terrorism.

Effect on the Games [edit]

In the wake of the hostage-taking, competition was eventually suspended for the first time in modern Olympic history, after public criticism of the Olympic Committee's decision to continue the games. On 6 September, a memorial service attended by 80,000 spectators and 3,000 athletes was held in the Olympic Stadium. IOC President Avery Brundage made little reference to the murdered athletes during a speech praising the strength of the Olympic movement and equating the attack on the Israeli sportsmen with the recent arguments about encroaching professionalism and disallowing Rhodesia's participation in the Games, which outraged many listeners.[33][*not in citation given*] The victims' families were represented by Andre Spitzer's widow Ankie, Moshe Weinberg's mother, and a cousin of Weinberg, Carmel Eliash. During the memorial service, Eliash collapsed and died of a heart attack.[50]

Many of the 80,000 people who filled the Olympic Stadium for West Germany's football match with Hungary carried noisemakers and waved flags, but when several spectators unfurled a banner reading "17 dead, already forgotten?" security officers removed the sign and expelled those responsible from the grounds.[11] During the memorial service, the Olympic Flag was flown at half-staff, along with the flags of most of the other competing nations at the request of Willy Brandt. Ten Arab nations objected to their flags being lowered to honor murdered Israelis; their flags were restored to the tops of their flagpoles almost immediately.[51]

Willi Daume, president of the Munich organizing committee, initially sought to cancel the remainder of the Games, but in the afternoon Brundage and others who wished to continue the Games prevailed, stating that they could not let the incident halt the Games.[11] Brundage stated "The Games must go on, and we must ... and we must continue our efforts to keep them clean, pure and honest."[52] The decision was endorsed by the Israeli government and Israeli Olympic team chef de mission Shmuel Lalkin.[53]

On 6 September, after the memorial service, the remaining members of the Israeli team withdrew from the Games and left Munich. All Jewish sportsmen were placed under guard. Mark Spitz, the American swimming star who had already completed his competitions, left Munich during the hostage crisis (it was feared that as a prominent Jew, Spitz might now be a kidnapping target). The Egyptian team left the Games on 7 September, stating they feared reprisals.[54] The Philippine and Algerian teams also left the Games, as did some members of the Dutch and Norwegian teams. American marathon runner Kenny Moore, who wrote about the incident for *Sports Illustrated*, quoted Dutch distance runner Jos Hermens as saying "It's quite simple. We were invited to a party, and if someone comes to the party and shoots people, how can you stay?"[55] Many athletes, dazed by the tragedy, similarly felt that their desire to compete had been destroyed, although they stayed at the Games.

Four years later at the 1976 Summer Olympics in Montreal, the Israeli team commemorated the massacre: when they entered the stadium at the Opening Ceremony, their national flag was adorned with a black ribbon.

The families of some victims have asked the IOC to establish a permanent memorial to the athletes. The IOC has declined, saying that to introduce a specific reference to the victims could "alienate other members of the Olympic community," according to the BBC.[56] Alex Gilady, an Israeli IOC official, told the BBC: "We must consider what this could do to other members of the delegations that are hostile to Israel."

The IOC rejected an international campaign in support of a minute of silence at the Opening Ceremony of the 2012 London Olympics in honour of the Israeli victims on the 40th anniversary of the massacre.[57][58] Jacques Rogge, the IOC President, said it would be "inappropriate". Speaking of the decision, Olympian Shaul Ladany, who survived the attack, commented: "I do not understand. I do not understand, and I do not accept it."[47]

In 2014 the International Olympic Committee agreed to contribute $250,000 towards a memorial to the murdered Israeli athletes.[59]

There is a memorial outside the Olympic stadium in Munich in the form of a stone tablet at the bridge linking the stadium to the former Olympic village. There is also a memorial tablet to the slain Israelis outside the front door of their former lodging at 31 Connollystraße. On 15 October 1999 (almost a year before the Sydney 2000 Games), a memorial plaque was unveiled in one of the large light towers (Tower 14) outside the Sydney Olympic Stadium.[60][61]

Israeli response [edit]

Main articles: Operation Wrath of God and 1973 Israeli raid on Lebanon

Golda Meir and the Israeli Defense Committee secretly authorized the Mossad to track down and kill those allegedly responsible for the Munich massacre.[62] The accusation that this was motivated by a desire for vengeance was disputed by Zvi Zamir, who described the mission as "putting an end to the type of terror that was perpetrated" (in Europe).[63] To this end the Mossad set up a number of special teams to locate and kill these fedayeen, aided by the agency's stations in Europe.[64]

In a February 2006 interview,[63] former Mossad chief Zvi Zamir answered direct questions:

Figure 2: Part of "Munich massacre" article at Wikipedia. We use the structural information to harvest event-event relations such as *aftermath(Munich massacre, 1973 Israeli raid on Lebanon)*.

has a summary and full text description, which can be seen as naturally annotated data for training a summarization model that generates a summary from full text description. In addition, it can even be used as training data for generating a summary of an event from an event's infobox or generating a text description of a (slot,value) pair of an event, which is very useful for question answering systems based on knowledge bases.

2.3.4. Event-centric knowledge base construction

As we introduced, *EventWiki* is a knowledge base of events which is collected from pages at Wikipedia that are manually constructed and edited. It is an example for event knowledge base and can be helpful for automatic event knowledge base construction.

Given a collection of documents about a major event, we can use *EventWiki* to predict the event's type, extract the event's schema, generate its infobox, identify the relations with other major events and generate a summary for it. When we finish all the steps above, the profile of the event is built, which can be used as an event entry for an event knowledge base. In other words, the above five steps can be seen as the subtasks of event knowledge base construction.

Among 21,275 events in *EventWiki*, we select 1,517 events happening during 1995-2010 and find[3] corresponding news articles of these events in English Gigaword (Parker et al., 2011), which can used to test results of each step and evaluate end-to-end performance.

3. Related Resources

Knowledge bases are useful resources for various applications in many fields. Among the exitsing knowledge bases such as WordNet[4] and Gene Ontology[5], knowledge bases of general world knowledge such as Freebase[6], DBPedia[7] and YAGO (Mahdisoltani et al., 2014) have received most attention. Most of the knowledge bases mainly focus on static entities and their related facts but pay little attention to influential events.

To the best of our knowledge, the most related resource to ours is the data released by (Reschke et al., 2014), which contains 193 air crash events and their corresponding news articles. In some sense, this data can be seen as a small knowledge base of air crashes. In contrast, our *EventWiki* contains much more major events of a variety of types and rich knowledge about events, which is the first complete event-centric knowledge base as far as we know.

4. Conclusion and Future Work

In this paper, we introduce a novel knowledge base – *EventWiki* which is the first knowledge base of major events. We introduce its structure and information contained in it, and show its potential value for NLP and automatic KB construction.

In future, we plan to enrich the event-event relation information by adopting a semi-automatic approach for analyzing relations between events in text. Additionally, we will regularly update the knowledge base to ensure that it contains the latest events and plan to provide an online interface and APIs for easier access.

5. Acknowledgments

The research work is supported by the National Science Foundation of China under Grant No. 61772040. The contact author is Zhifang Sui.

[3] We use information retrieval package Lucene (Jakarta, 2004) to find relevant documents to these events and manually check the results.

[4] https://wordnet.princeton.edu/

[5] http://geneontology.org/

[6] https://www.freebase.com/

[7] http://wiki.dbpedia.org/

6. Bibliographical References

Jakarta, A. (2004). Apache lucene-a high-performance, full-featured text search engine library.

Mahdisoltani, F., Biega, J., and Suchanek, F. (2014). Yago3: A knowledge base from multilingual wikipedias. In *7th Biennial Conference on Innovative Data Systems Research*. CIDR Conference.

Reschke, K., Jankowiak, M., Surdeanu, M., Manning, C. D., and Jurafsky, D. (2014). Event extraction using distant supervision. In *LREC*, pages 4527–4531.

7. Language Resource References

Parker, Robert and Graff, David and Kong, Junbo and Chen, Ke and Maeda, Kazuaki. (2011). *English Gigaword Fifth Edition*. Linguistic Data Consortium, 5.0, ISLRN 911-942-430-413-0.

Annotating Spin in Biomedical Scientific Publications: the case of Randomized Controlled Trials (RCTs)

Anna Koroleva, Patrick Paroubek
LIMSI-CNRS,
Bât. 508, rue John von Neumann, campus Université d'Orsay, F-91405 Orsay
{koroleva, pap}@limsi.fr

Abstract
In this paper we report on the collection in the context of the MIROR project of a corpus of biomedical articles for the task of automatic detection of inadequate claims (spin), which to our knowledge has never been addressed before. We present the manual annotation model and its annotation guidelines and describe the planned machine learning experiments and evaluations.

Keywords: spin, annotation scheme, biomedical articles

1. Introduction

Merriam Webster dictionary defines spin doctor as *"a person (such as a political aide) responsible for ensuring that others interpret an event from a particular point of view"*[1]. In the context of the MIROR[2] project, we address spin in biomedical scientific publications, where it refers to misleading presentation of scientific results, in particular in articles reporting randomized controlled trials (RCTs), an important type of clinical trial. In our case, spin consists in presenting the examined treatment as having greater beneficial effects than the experiments show (Boutron et al. 2010; Boutron et al. 2014; Haneef et al. 2015; Yavchitz et al. 2016). Spin in RCTs affects clinical decision-making (Boutron et al. 2014) and results in distorted presentation of research findings in media (Yamamoto & Takagi 2005; Hall et al. 2009). We present here the first steps aiming at proposing an algorithm for automatic spin identification in biomedical abstracts, something which to the best of our knowledge has not been attempted before. We present here the construction and annotation of a corpus of medical publication extracted from PubMed Central[3] (PMC) about RCT and describe the annotation model and guidelines.

1.1 On spin types

From previous research on spin classification (Boutron et al. 2010; Lazarus et al. 2015, Yavchitz et al. 2012], we can outline three main types and their subtypes of spin in RCT reports:

1. misleading reporting of study results:

- selective reporting of outcomes (omission of the primary outcome; focus on statistically significant results different from the main outcome);
- occulting adverse events;
- misleading reporting of study design;
- linguistic spin (beautifying formulations);
- discarding limitations;
- selective citation of other studies

2. inadequate interpretation of the results:

- claiming a beneficial effect of the intervention despite statistically non-significant results;
- claiming an equivalent effect of the interventions for statistically non-significant results;
- claiming that the treatment is safe for statistically non-significant safety outcomes;
- concluding a beneficial effect despite no comparison test performed;
- interpretation of the results according to statistical significance instead of clinical relevance;

3. inadequate extrapolation:

- inadequate extrapolation from the population, interventions or outcome actually assessed in the study to a larger population, different interventions or outcomes;
- inadequate implications for clinical practice.

Example of spin putting focus on secondary result ("*improved PFS and response for treatment*") instead of the main result, object of the experiment ("*survival rate*"):

> *"This study demonstrates improved PFS and response for the treatment A compared with comparator B, although this did* ***not result in improved survival*** *".*

Fig. 1. Example of spin (focus on secondary result)

In the rest of this paper, we present our linguistic model of spin (section 2), the annotation scheme (section 3), the the annotation guidelines (section 4), conclusions and plans for future work (section 5).

2. Model of spin

To the best of our knowledge, this is the first attempt at addressing the analysis of spin in the biomedical literature from a Natural Language Processing point of view. Spin detection overlaps partially with previous works in NLP, in particular objectivity/subjectivity identification (Wiebe

[1]https://www.merriam-webster.com/dictionary

[2]http://miror-ejd.eu/

[3]https://www.ncbi.nlm.nih.gov/pmc/

et al. 2005), sentiment analysis (Pak 2012), fact checking (Nakashole & Mitchell 2014) or deception detection (Hancock et al. 2010; Litvinova et al. 2017); a point to note is that these works address texts of general domain while we deal with spin in biomedical texts. We regard spin detection as a task most closely related to deception detection. Deception is defined as a deliberate act of communicating information that the speaker/author believes to be false, with the intention to induce listeners/readers to believe a distorted presentation of the topic. Strictly speaking, spin is not necessarily a form of deception, as the intention is difficult to establish most of the time, e.g., spin in abstracts may be conditioned by limited space; by author's wish to report the results that he/she perceives to be most important; by unclear/absent reporting guidelines; by lack of training etc. However, spin is similar to deception for what concerns its impact and the method required to detect it from textual content only (Mihalcea et al. 2009).

Spin can be considered as the most serious form of incomplete or incoherent reporting of trial data and results (omission or inconsistent presentation of information). We aim at creating a general model that would be able to represent the information about a trial data and all possible realizations of spin in reporting.

For trial data, we choose to follow the information structure accepted in trial registries (official catalogues for registering clinical trials, containing in a structured form trial data provided by the investigators who carry out a trial).

Start Date ICMJE	September 2006
Primary Completion Date	*Not Provided*
Current Primary Outcome Measures ICMJE **(submitted: June 8, 2007)**	Sleep quality assessed by the Leed Sleep Evaluation Questionnaire, measured after one month of treatment
Original Primary Outcome Measures ICMJE	*Same as current*
Change History	No Changes Posted
Current Secondary Outcome Measures ICMJE **(submitted: June 8, 2007)**	• Sleep quality assessed by the Leed Sleep Evaluation - Questionnaire, measured 15 days after treatment withdrawal • Sleep quality assessed by a Sleep diary completed each day during all the study period by the participants • Sleep efficiency measured by ambulatory actigraphy (2 period) • Evolution of melatonin/6sulfatoxymelatonin ratio (before and after treatment) • Clinical General Impression of the clinician, before and after treatment. • Safety of the treatment (adverse event reporting)

Fig. 2. Excerpt from an RCT description queried on ClinicalTrials.gov with the keywords: '*RCT insomnia France*'

Trial registries may slightly vary regarding the level of detalisation used for information presentation, so we reviewed several registries (ClinicalTrials.gov[4], ISRCTN[5], plus some national registries) and generalized the categories used. We compiled the following list of data describing a trial:

- Information about interventions: intervention name, dosage, administration schedule, treatment duration;
- Information about participants: age, gender, health condition, health type, nationality/ethnicity, recruitment country/region; information regarding intervention assigned; other information. Can be represented in a form of a list of inclusion and exclusion criteria, that can contain all of the above information;
- Trial methods / trial design: allocation concealment, allocation type, blinding, sample sizes for groups examined, study type, study subtype, trial phase, statistical tests used;
- Trial objectives / outcomes (with their methods of measurement and associated time points);
- Data about registration: registration number, registration time;
- Financing: sponsors;
- Hypothesis, hypothesis type;
- General information: medical domain;
- Summary.

We also introduced some other categories that are not typically present in registries but that are relevant to trial description: limitations and reported statistical measures

In order to be able to capture instances of spin, we further need to reflect the following phenomena:

1. Incomplete reporting, which can take many forms, but we are most interested in omission of information that is normally supposed to be present in a well-reported abstract, such as:

- clear definition of the primary outcome;
- results for primary outcome;
- results for non-significant secondary outcomes;
- information about adverse events (their absence should be stated explicitly).

Omission of some other types of information (design, methods, statistical tests used, etc.) should not be considered as spin but rather as incomplete reporting acting as 'spin facilitator' hindering fact checking.

2. Incoherent reporting:

- primary outcome described in the trial registry differs from the primary outcome described in the text;
- patient population reported in the abstract does not correspond to the population studied in its qualitative characteristics (age, gender, etc.);
- reported results do not correspond to trial design;
- the compared treatments are reported to be similar when the design does not allow to conclude on similarity (i.e. the trial is not a 'non-inferiority' of 'equivalence' trial);
- within-group comparison reported when the trial objective was not to examine changes within groups (i.e. the trial is not a 'before-and-after trial');

[4]https://clinicaltrials.gov/

[5]https://www.isrctn.com/ (International Standard Randomised Controlled Trial Number Register)

- focus on significant secondary outcomes instead of primary outcome;
- positive conclusions are made (efficacy stated, treatment recommended for use) when the primary outcome is not significant.

Incoherence or incompleteness of reporting can be established by checking the completeness of the abstract, discrepancies between abstract and article body or between trial registry entry (if available) and article content. We thus work with two types of documents: articles and registry entries. For articles, the model comprises information about its structure: its division into title, abstract and body text, for registries we rely on their internal structure, in general a tabular form holding short pieces of text or data.

3. Annotation scheme

We proposed a description of an algorithm of spin detection elsewhere (Koroleva & Paroubek 2017a). The main steps are the following:

- dividing a given article into title, abstract and body text; finding results and conclusions within the abstract;
- identifying positive evaluation of the studied treatment in results/conclusions of the abstract;
- extracting elements of trial data relevant to spin assessment, such as outcomes, patient population, statistical significance of results;
- extracting relation between elements of trial data, such as an outcome and its statistical significance;
- extracting specific constructions possibly related to spin (see below);
- final assessment of spin: checking if the information in the results and conclusions of the abstract corresponds to the extracted trial data, for example, if the pre-defined outcomes are reported correctly or if the positive evaluation of the treatment is supported by statistically significant results.

We propose here an annotation scheme comprising the information elements relevant for the future algorithm. Our annotation scheme is implemented in XML and includes several levels of information:

1. Document type (article/registry entry).

2. Structural information (for articles). For this annotation level we adopt the existing annotation scheme used in PubMed[6], simplified for our needs. Our scheme includes journal name, article title, authors list, abstract, body text, bibliography. Within abstracts, Results and Conclusions sections are marked.

3. Elements describing the trial (what was studied and how: compared interventions, outcomes, population studied, statistical measures used, etc.): we introduce a separate tag for each type of trial data. This decision is motivated by the fact that we need specific sets of attributes for different types of trial data, and we need to introduce particular relations for specific types of trial information. As outcome is the most important type of trial data for spin detection, for outcomes (or trial objectives) we use several tags that are needed to distinguish between different specific constructions:

The <Prol>primary outcome measure will be</Prol> <Out type=”Prim” Status=”Declared”>**QoL**</Out>, assessed with the ALS Assessment Questionnaire...

Fig. 3. Example of annotation for a primary (attribute type is Prim) outcome (Out) explicitly declared, with the annotation of its linguistic marker (Prol).

The type of an outcome can have three different type attribute values: Prim (primary) / Sec (secondary) / None (undefined). Outcome has also an attribute ‘status’ which can have two values: Declared when it is explicitly stated in the text to be an outcome (e.g. Fig 3), which is its value by default and Reported, when the outcome is only reported in results or conclusion section without referring explicitly to its nature.

Our <Prol>secondary aim is</Prol> <Aim type=”Sec”>**to describe the costs**</Aim> associated with RESERVE-DSD.

Fig. 4. The AIM is the objective of the trial.

4. Relations between elements of trial data: relations that link a pair of elements that describe different features of a single concept, e.g. an outcome with its method of measurement or with its time points, or an intervention to its dosage, administration schedule, etc.

5. Particular constructions of interest:

- Positive evaluations of treatment (positive results regarding the treatment);
- Statements of similarity between treatments regarding their efficacy or safety;
- Within-group comparisons (statements of changes that occur within a group receiving the studied treatment, without comparing it to the group receiving the control treatment);
- Recommendations to use treatment.

These include: i) an analysis which shows that the ethnic difference in performance in this 2006/7 <Subj>**cohort of Year 3 students**</Subj> was similar in size to that in <Subj>**previous cohorts on the course**</Subj> [see Additional file 2]

Fig. 5. Example of a similarity statement. Subj – trial subjects.

A problem that arises with this type of information consists in deciding which fragment of text should be

[6] https://www.ncbi.nlm.nih.gov/corehtml/query/DTD/index.shtml

annotated. Normally these constructions comprise a whole proposition, but we can as well highlight some words that are the most 'representative' of the meaning of each construction. We choose to annotate the smallest possible fragments that are indicators of relevant constructions. The motivation behind this decision is the need to make the annotation as clear and simple as possible for annotators, and the fact that, having annotation on word level, we can easily expand it to the sentence level.

6. Annotation for spin: annotation level that is meant to capture all the cases of incoherence and incompleteness regarding the types of information enumerated above. This type of annotation resembles most to a well-known task of relation annotation, but here the most important is not to capture relation that holds between two text fragments, but to mark the cases when there is no relation when we expect it to exist. For example, a relation between a declared primary outcome in article text or protocol and a corresponding reported outcome in abstract means no spin, but a declared primary outcome with no related reported outcome is a case of spin. A similarity statement is not spin if the trial was of equivalence type, but it is a spin if there is no text fragment indicating that the trial belongs to equivalence trials.

To annotate this information, we follow the system accepted in TimeML (Pustejovsky et al. 2003) annotation for relations: we introduce empty tags that contain reference to IDs of fragments that are linked in case of good reporting; in case of incoherence/incompleteness, the tag contains ID of the present text fragment. These tags have an attribute 'spin' that is set to 'yes' or 'no'.

Another form of actual spin or of 'spin facilitator' is omitting some information about methods, design or results in the abstract, e.g. not stating clearly the primary outcome. For this type of omission, we do not need to refer to an ID, we only need an empty tag to mark which type of information is omitted in abstract.

Thus, the annotation for spin is done on the lowest level: as a relation between text fragments. We can then calculate the value of 'spin' attribute for the whole text.

Figure 6 shows an example of text with spin (the example comes from the appendix of Boutron et al. 2014) and the process of assessment of outcome-related spin.

<Abstract>Abstract

<...>

<Res>Results

<...> <Out ID="1" Type="None" Status="Reported">**The International Union Against Cancer R0 resection rate**</Out> was 81.9% after treatment A as compared with 66.7% with surgery alone (P = .036). The surgery-only group had more <Out ID="2" Type="None" Status="Reported">**lymph node metastases**</Out> than the treatment A group (76.5% v 61.4%; P = .018). <...> A <Out ID="3" Type="None" Status="Reported">**survival**</Out> benefit could not be shown (hazard ratio, 0.84; 95% CI, 0.52 to 1.35; P = .466). </Res>

<Concl>Conclusion

This trial showed a significantly increased <Out ID="4" Type="None" Status="Reported">**R0 resection rate**</Out> but failed to demonstrate a <Out ID="5" Type="None" Status="Reported">**survival**</Out> benefit. </Concl>

<BodyText> <...>

The primary end point of this trial was <Out ID="6" Type="Prim" Status="Declared">**overall survival**</Out>. <...></BodyText>

Fig. 6. Example of annotation of spin for an abstract

The first step in annotating this text would be to annotate all outcomes reported in the abstract (IDs 1 – 5) and the declared primary outcome (ID 6). The following steps to fully annotate all types of spin related to primary outcome would be the following:

1) Check and mark if there is a definition of the primary outcome in the abstract. Here it is absent (full text if abstract omitted for the sake of space) – we conclude incomplete reporting.

2) Check and mark if the declared primary outcome is present among the reported outcomes. Here it can be considered to correspond to the outcomes 3 and 5 – we conclude correct reporting.

3) Check and mark if the primary outcome is presented correctly according to its importance: it should be presented in the first place without regard to significance of results; there should be no focus on other outcomes. In this abstract, the insignificant primary outcome is presented after significant secondary ones – we conclude incoherent reporting (focus on secondary outcomes).

4. Annotation guidelines

We plan to combine automatic annotation as first stage, and manual annotation to correct and complete the annotation. We do not aim at manually annotating all the types of information. Most of the trial data not directly relevant to spin detection will be marked automatically only in trial registry entries, where information is highly structured. We do not thus cover them in the annotation guidelines.

We described our algorithms of automatic pre-annotation in our previous works (Koroleva & Paroubek 2017a, Koroleva & Paroubek 2017b). These algorithms aim at extracting/annotating the following:

- text structure: separating results and conclusiong sections in abstracts;
- various constructions defining trial outcomes, with special attention to the primary one, for example:

1. The primary outcome is <Out Type="Prim" Status="Declared"> emotional distress (symptoms of depression , anxiety , diabetes-specific stress , and general perceived stress) </Out> .

2. This project has one primary aim : to measure <Out Type="Prim" Status="Declared"> the impact of continuity of midwifery care </Out> compared to routine care on restricting excessive gestational weight gain in obese women.

3. Sample size A power calculation was carried out for the primary outcome (<Out Type="Prim" Status="Declared"> health related quality of life measured on the York version of the SF-12</Out>).

- comparative constructions that are often used to report the trial results. These constructions usually include some of the following elements: compared patient groups, compared treatments, outcomes that serve as basis for comparison. We mainly focus on extracting outcomes:

1. <Subj> Patients with TC asthma </Subj> has significantly higher<Out Type="None" Status="Reported"> AQLQ scores </Out> compared to those with NTC asthma.

2. Muscarinic agonists appear to reduce <Out Type="None" Status="Reported"> the potency of beta-agonist bronchodilation </Out> , possibly through an effect on adenylyl cyclase 17.

3. <Out Type="None" Status="Reported"> Levels of hs-CRP </Out> increased modestly in the ABC / 3TC arm compared with the TDF / FTC the arm.

- Description of studied population:

1. We studied <Subj> <Aim Type="None"> 19 consecutive unselected patients who met the ARDS criteria of the American European Consensus Conference 21. </Aim> </Subj>

2. A total of <Subj> 32 patients aged 12 to 17 years with severe , active and refractory JoAS </Subj> were enrolled in a multicenter , randomized , double-blind , placebo-controlled parallel study of 12 weeks.

These annotations, although not perfectly correct and complete, are hoped to reduce workload for annotators: in case pre-annotation is completely correct or completely erroneous, the annotators will simply need to validate/reject it, reducing the number of cases requiring manual annotation.

The current pre-annotated corpus includes 3938 articles on randomized controlled trials in various medical domains, extracted from PubMed Central. This corpus will serve as basis for manual annotation.

We will split manual annotation into several stages that would differ regarding their complexity and thus the skills required from the annotators.

Some of the tasks are relatively easy and can be done by annotators who do not have special knowledge in medical domain. We consider that the tasks that fall into this group are: explicit descriptions of outcomes, mentions of patient population, statistical measures (p-value), confidence intervals.

Some other types of information require some special knowledge of medical domain as understanding of medical terms is needed to correctly interpret the meaning of sentences and categorize text fragments as representing a certain type of trial data/construction. Following tasks fall into this category: reported outcomes, similarity statements, within-group comparisons, evaluations related to treatment.

The final task of spin annotation (i.e. marking parts of the text that represent coherent and complete reporting for chosen concepts, and marking cases when there is incoherence/incompleteness) is an even more difficult task. The concept of spin in biomedical domain is not completely formally defined yet, experts in the domain often disagree on classifying a certain phenomenon as spin or not. For example, some experts regard absence of explicit definition of the primary outcome in the abstract of an article as definite spin, while others consider it to represent incomplete reporting but less important than spin. Besides, mismatch between information in the abstract and in the article (e.g. change of outcomes studied and reported) is not spin if it has valid scientific justification, which should be provided in the article. Extraction of such justifications and assessment of their validity would be necessary to conclude on absence or presence of spin, but it falls outside scope of our work.

Thus, there are several difficulties that we should take into account when developing annotation guidelines:

1) Some of the tasks require at least some level of special medical knowledge, so it is likely that the annotators will not be linguists and will not have experience in corpus creation/annotation. This fact should be taken into account when choosing terminology (no specific linguistic terms) and when defining the task (e.g., be clear about annotating coordinated elements as separate elements and not one element).

2) Choice of the annotation tool to be used should take into account the complexity of the task but also the involvement of non-linguists in annotation process. From the point of view of functionality, the tool should at the very least be able to capture relations, potentially embedded. This requirement makes tools not allowing relation annotation, such as WebAnnotator (Tannier 2012), not appropriate. After testing and comparing several tools, we chose the Glozz platform (Widlöcher & Mathet 2012) as the one that best corresponds to the needs of the task of full linguistic annotation of spin. Glozz is a flexible and powerful tool that allows to annotate units (text fragments), their relations and schemes (which can be seen as higher-level relations that can include one or more units, relations or other schemes) which covers all possible instances of incompleteness or incoherence in reporting.

However, demonstration of text annotation with Glozz to a medical expert showed that it does not meet the requirements of non-linguist annotators: ease of installation of the tool, amount of time needed for training for a person without previous experience in corpus annotation, complexity of guidelines describing the task. Consequently, we decided to replace the task of linguistic annotation of texts by a set of simpler tasks (in the form of questions) such as the following:

- validation/correction of primary outcomes found at the pre-annotation stage;

- validation/correction of reported outcomes found at the pre-annotation stage;

- establishing if two given (extracted at previous stages) outcomes refer to the same concept;

- identification of similarity statements in the Results and Conclusions of abstracts;

- identification of within-group comparisons in the Results and Conclusions of abstracts;

- identification of other positive evaluation of the studied treatment in the Results and Conclusions of abstracts.

We plan to use a web-based survey tool (such as LimeSurvey[7]) to create questionnaires containing these questions, generated on the basis of pre-annotation. Using survey tools for corpus annotation is not typical. Our decision is motivated by several reasons: survey tools are usually available online and thus do not require any complex installation procedures (survey participants can access the survey simply by following a link received by email); survey tools are widely used in medical community and are familiar to the community. This fact reduces time needed for annotators to learn how to use the tool. Besides, breaking the task into simple questions, independent one from another, allows to include into each question a brief guideline on how to answer, thus in most cases annotators will not need to refer to an extensive external annotation guide. Answering simple questions is also likely to cause fewer discrepancies between annotators than full annotation of spin.

3) In case of full linguistic annotation of spin, we should clearly define which pieces of text to annotate. We anticipate some difficulties in cases when elements of trial data get embedded one into another. The guidelines should explain whether to annotate these elements as embedded or as two separate instances linked by a certain type of relation (e.g. outcome and its method of measurement).

4) Given the complexity of the task, we need to clarify the definition of what should be considered to be spin. For this, we need to strictly define the types of spin that we focus on, describe in detail which pieces of information are relevant to these types of spin. Taking into account lack of agreement between experts in detailed definition of spin, for our current annotation project we decided to avoid using the notion "spin" and focus on tasks that are relatively simpler and clearer, such as: annotating outcomes; marking if pairs of extracted outcomes refer to the same concept; annotating specific constructions of interest, such as similarity statements or within-group comparisons. This information would allow to estimate with a certain probability that an article does or does not contain spin, but the final decision is left to the human readers of the article.

5) The task of developing guidelines must be fulfilled in close collaboration with experts in medical domain and in the domain of spin in medical texts, in order to verify that all the definitions regarding medical concepts and spin are correct.

[7] https://www.limesurvey.org/

5. Conclusions and future work

In this paper we described our approach to creation of a corpus of biomedical articles annotated for spin (distorted reporting) and its supporting information. We briefly outlined the proposed algorithm of spin detection and summarized our work on automatic pre-annotation of the corpus. Consequently, we described the annotation scheme that we developed for spin annotation. We discussed the process of creating the annotation guidelines, provided some thoughts as for choice of annotation tool and outlined expected challenges.

Our future tasks include running a pilot survey to validate usability of survey format for our task and evaluate the adequacy and clarity of the questions for annotators. Consequently we will proceed to a full-scale survey project.

Acknowledgements.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 676207.

6. Bibliographical References

1. Boutron I., Altman D.G., Hopewell S., Vera-Badillo F., Tannock I., Ravaud P. Impact of spin in the abstracts of articles reporting results of randomized controlled trials in the field of Cancer: the SPIIN randomized controlled trial. J Clin Oncol, 32, 4120–4126 (2014)
2. Boutron I., Dutton S., Ravaud P., Altman D.G. Reporting and interpretation of randomized controlled trials with statistically nonsignificant results for primary outcomes. JAMA, 303, 2058–2064 (2010)
3. Conroy N. J., Rubin V. L., & Chen Y., Automatic Deception Detection: Methods for Finding Fake News, Proceedings of ASIST 2015, St. Louis, MO, USA, November (2015)
4. Hall M., Frank E., Holmes G., Pfahringer B., Peter R., Witten I. H. The weka data mining software: An update. SIGKDD Explorations, 11(1) (2009)
5. Hancock J. T., Beaver D. I., Chung C. K., Frazee J., Pennebaker J. W., Graesser A., & Cai Z., Social language processing: A framework for analyzing the communication of terrorists and authoritarian

regimes, Behavioral Sciences of Terrorism and Political Aggression, Vol. 2, No. 2, pp 108–132, May (2010)
6. Haneef R., Lazarus C., Ravaud P., Yavchitz A., Boutron I. Interpretation of results of studies evaluating an intervention highlighted in Google Health News: a cross-sectional study of news. PLoS ONE, 10(10) (2015)
7. Koroleva A., Paroubek P. Automatic detection of inadequate claims in biomedical articles: first steps. Proceedings of Workshop on Curative Power of MEdical Data (2017)
8. Koroleva A., Paroubek P. On the contribution of specific entity detection and comparative construction to automatic spin detection in biomedical scientific publications. Proceedings of The Second Workshop on Processing Emotions, Decisions and Opinions at The 8th Language and Technology Conference (2017)
9. Lazarus C., Haneef R., Ravaud P., Boutron I. Classification and prevalence of spin in abstracts of non-randomized studies evaluating an intervention. BMC Med Res Methodol., 15:85 (2015)
10. Litvinova O., Seredin P., Litvinova T. & Lyell J., Deception detection in Russian texts, Proc. of the Student Research Workshop at EACL2017, Valencia, Spain, April (2017)
11. Mihalcea R., and Strapparava C., The Lie Detector: Explorations in the Automatic Recognition of Deceptive Language. Proceedings of ACL, pp. 309-312, Singapore, (2009).
12. Nakashole N., & Mitchell T.M., Language-Aware Truth Assessment of Fact Candidates, Proceedings of the 52nd Annual Meeting of the ACL, pp 1009–1019, Baltimore, Maryland, USA, June (2014)
13. Pak A., Automatic, adaptive, and applicative sentiment analysis, PhD. dissertation, U. Paris-Sud, https://tel.archives-ouvertes.fr/tel-00717329
14. Pustejovsky J., Castaño J., Ingria R., Saurí R., Gaizauskas R., Setzer A. and Katz G. TimeML: Robust Specification of Event and Temporal Expressions in Text. IWCS-5, Fifth International Workshop on Computational Semantics (2003)
15. Tannier X. WebAnnotator, an Annotation Tool for Web Pages. Proceedings of the 8th International Conference on Language Resources and Evaluation (2012)
16. Widlöcher A., Mathet Y. The Glozz platform: a corpus annotation and mining tool. ACM Symposium on Document Engineering (2012)
17. Wiebe J., & Rilof E., Creating Subjective and Objective Sentence Classiers from Unannotated Texts, Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing) (2005)
18. Yamamoto Y., Takagi T. A sentence classification system for multi biomedical literature summarization. Proceedings of the 21st International Conference on Data Engineering Workshops (2005)
19. Yavchitz A., Boutron I., Bafeta A., Marroun I., Charles P., Mantz J., et al. Misrepresentation of randomized controlled trials in press releases and news coverage: a cohort study. PLoS Med, 9:e1001308 (2012)
20. Yavchitz A., Ravaud P., Altman D.G., Moher D., Hrobjartsson A., Lasserson T., Boutron I. A new classification of spin in systematic reviews and meta-analyses was developed and ranked according to the severity. Journal of Clinical Epidemiology, 75, 56-65 (2016)

Visualization of the occurrence trend of infectious diseases using Twitter

Ryusei Matsumoto, Minoru Yoshida, Kazuyuki Matsumoto, Hironobu Matsuda, Kenji Kita
Institute of Technology and Science, University of Tokushima
2-1, Minami-josanjima, Tokushima, 770-8506, Japan
c501306048@tokushima-u.ac.jp, mino@is.tokushima-u.ac.jp, matumoto@is.tokushima-u.ac.jp,
c501637026@tokushima-u.ac.jp, kita@is.tokushima-u.ac.jp

Abstract

We propose a system for visualizing the epidemics of infectious diseases. We apply factuality analysis both to disease detection and location estimation for accurate visualization. We tested our methods for several infectious diseases, and show that our method performs well on various diseases.

Keywords: Twitter, disease surveillance, location estimation

1. Introduction

Information related to human life such as illness and infectious diseases is important for many people. However, people have limited kinds of sources of information on diseases such as TV news and newspapers (including Internet news.) On the other hand, SNS including Twitter is a promising source of another type of information about diseases. It has a highly real-time nature and can obtain a large amount of information about the event occurring just now. The final goal of this research is to provide users with real-time infection maps automatically created from Twitter posts. To this end, we propose a system to obtain how many people are infected by the given disease in each area. The proposed system extracts the tweets that contain the disease name, and estimates the location of the user. The system performs factuality analysis of mentions both for infection events and user locations for more accurate estimation. Finally, an infectious disease map is created using the result of the location estimation.

2. Existing Research

Several researches that combine Twitter and illness have been done so far (Charles-Smith et al., 2015). For example, the system by Aramaki et al. (Aramaki et al., 2011; Iso et al., 2016; Kanouchi et al., 2015; Kitagawa et al., 2015) extracts tweets related to influenza posted on Twitter, conducted factuality analysis using SVMs on extracted tweets for detecting epidemic influenza. They also provide the "influ kun" service[1] which extracts tweets related to influenza from Twitter and provides a epidemic map of influenza overlaid on a map of Japan. Other existing research on disease surveillance with location estimation used simple geocoding (Li and Cardie:, 2013), geotagged tweets (Sadilek et al., 2012) and user profiles (Dredze et al., 2013; Paul and Dredze, 2011). Our system is different from this service in that it is aimed at various kinds of diseases other than influenza, and estimate users location even if the user location is not explicitly provided in the form of geotag or user profiles.

There have also been many researches on estimating user locations from tweets (contents) (Chandra et al., 2011; Cheng et al., 2010; Han et al., 2012; Sugiya et al., 2013). Most of the research use the geotagged tweets as training data and estimate the positions of users/tweets based on the word-geotag collocations. The goal of those previous location estimation researches is to estimate the locations vaguely with best-effort approaches. On the other hand, our goal is to provide the positions of infected people as accurately as possible. For that reason, our location estimation algorithm is different from the previous ones: we only focus on the location names found in tweets, and *do the factuality analysis on them* to determine whether the user actually is in the position. To the best of our knowledge, this is the first research to take factuality analysis approach to location estimation for Twitter users.

Some of the researches take time-aware approaches, which estimate the given location name is for the current position, or future position, etc. (Li and Sun, 2014; Suzuki et al., 2015) Our problem setting is similar to them, but different from in that our goal is to estimate where the user live in by excluding not only the non-current-position-indicating tweets but also non-living-position-indicating tweets, which is difficult only by lexical approaches such as '"now" indicates the current position', etc.

Our contribution is two-fold. First, we report how the factuality analysis proposed for influenza generalize to other diseases by testing it on several disease names with their accuracies. Second, we propose a new task of factuality analysis of location indication when the location name itself is clearly provided.

3. Proposed method

We define our problem as the task to perform random search on Twitter with a disease name, and determine the living place of the user by looking at tweets of the user as few as possible. We assume this setting because in the real system it will be needed to estimate user location as lightly as possible.

3.1. Acquisition of tweets related to infectious diseases

In this research, we use TwitterAPI to acquire tweets. We use names or synonyms of seven kinds of infectious diseases that are prevalent in the summer as the search target.

[1] http://mednlp.jp/influ_map/

・As a result of inspection at the hospital, it was Helpangina
・Suspicion of Helpangina
・it's prevalent, Helpangina
・@XXX Is it the Helpangina that is in fashion now?

Figure 1: Examples of Helpangina tweets (translated from Japanese)

As a result of inspection at the hospital, it was Helpangina. 1
It may be Helpangina. -1
I am scared of Helpangina. -1

Figure 2: Examples of training data on infectious diseases

The targeted infections are described in detail in Section 4. As an example, we explain an experimental method of conducting a search using an infection term "Helpangina". Figure 1 shows an example of the acquired tweets. The reason for using synonyms is that many of the infectious diseases targeted in this research were more likely to contain synonyms in tweets than the official names. As an example, for the infectious diseases "epidemic parotitis", another name for this infection is "mumps". Generally, "mumps" is used more often than "epidemic parotitis". Using such synonyms thus increases the number of retrieved tweets.

3.2. Factuality analysis on infectious diseases

Factuality analysis of disease events are performed by using SVMs, which is the same way as the method by Aramaki et al. (Aramaki et al., 2011) The tweet which indicates infections is a positive example, and taken as a negative example otherwise. The training data for each disease consist of 1,000 tweets (500 positive and 500 negative) and the test data consists of 400 tweets for each disease. Figure 2 shows an example of training data. As a meaning of labels, "1" is infected, "-1" is not infected

3.3. Location estimation

We also performed the factuality analysis for location indications. As mentioned above, our problem setting is that we estimate the user's living area given *the tweets as few as possible*. This is because in the real situation it takes much cost to obtain whole list of tweets from one user, which makes it difficult to use user's whole tweets for location estimation. We assume the situation in which we search random tweet collections, collect disease-related tweets, look at the user's timeline and find one tweet with location name, and determine if the location tweet is valid. This makes it possible to construct the system which are more comprehensive for surveillance by being able to refer to more users.

3.3.1. Tweet Acquisition

We constructed the test data for location estimation for other 100 users, extracting his/her representative location tweet by obtaining the latest tweets that contain location name in his/her timeline[2]. We used the location information database for geotagging which consists of words expressing location information such as prefecture name, city name, spot name such as station name. We constructed this database by compling the data from *the Japanese location data download service*[3], which provides a list of city/street names with their location metadata such as its latitude, longitude, and corresponding prefecture name. The database also includes a list of major train station names and famous leisure spots with their corresponding prefecture names, which was manually added by the authors. The estimation is a prefecture-level, which means that we estimate the prefecture the user lives in by matching location names found in the tweets with our database.

We manually exclude the tweet if the tweet is travel-related one by extracting and checking the tweets before and after the target tweet. (Figure 3 shows an example.) [4]

It's a learning trip from tomorrow!
↓
End day 1
↓
End day 2
↓
I forgot to buy a Tokyo banana. let's buy(Detect「Tokyo」)
↓
It seems long and it was a short learning trip!

Figure 3: Examples of tweets representing travel

It seems that I can return to Tokyo earlier than I expected. 1
I will go to Tokyo on my own weekend. -1
I should buy it. Should I get off at Hakata? -1

Figure 4: Example of training data on location information.

3.3.2. Factuality analysis on location estimation

We constructed the training data that consist of 1,000 tweets (500 positive and 500 negative ones.) A positive example means a tweet that can be determined to live in a place of a word representing location information, a negative example means a tweet that can be determined that the user does not live in that location. Figure 4 shows an example of training data. As a meaning of labels, "1" is what the contributor tweets in the location where the per-son lives, "-1" is other things.

[2] The users whose timeline did not include any location names were excluded from the test data.

[3] http://nlftp.mlit.go.jp/isj/

[4] We are currently trying to automate this process. Determination is done by the same way as other classification, i.e., using SVMs. Again, our problem setting is to estimate if the given one tweet is the one during travel or not. We constructed the training data consisting of 200 positive and 200 negative tweets, and 50 test tweets. Positive tweets are the tweets that indicate the user is in travel, and negative tweets are other ones. We obtained 62% accuracy (31/50).

Infectious disease	Accuracy[%]
Helpangina	83.75
Mumps	66.25
Infectious gastroenteritis	79.75
Epidemic conjunctivitis	72.00
Mycoplasma	84.00
Pool heat	68.50
Hand-foot-and-mouth disease	67.75

Table 1: Experimental results of factuality analysis on infectious diseases

4. Experiments

4.1. Factuality analysis for infectious disease events

In this research, we focus on the following seven infectious diseases, for which the results of accuracy evaluation of factuality analysis are shown.

1. Helpangina
2. Mumps
3. Infectious gastroenteritis
4. Epidemic conjunctivitis
5. Mycoplasma
6. Pool heat
7. Hand-foot-and-mouth disease

Because it targets infectious diseases prevailing in the summer, tweets collected from July to August are used as experimental data. Regarding location estimation, it shows the result of position estimation for the newly acquired twitter account. We used word unigrams as features to represent each tweet for SVM training/classification.
Table 1 shows the evaluation results of the accuracy of factuality analysis using SVMs for the seven targeted infectious diseases. The accuracy was 66-84[%].

4.2. Factuality analysis for location estimation

As described in previous sections, 100 user accounts are newly selected for evaluation of location estimation. The acquisition conditions are as follows.

1. Location information is described on the profile
2. Does not include "BOT", "Bot", "bot" in one or more of Twitter client name, account name, display name, and profile

(1) of the above conditions is aimed at easily judging whether or not the result of estimating the position information from the tweet sentence is correct. Also, users who describe location information on their profile tend to have more tweets containing words that describe the area in which they live, than tweets that do not describe location information. (2) is aimed to exclude Bot accounts.
Table 2 shows the evaluation results. The accuracy was 78[%].

	Accuracy[%]
Factuality analysis on location estimation	78.00

Table 2: Experimental results of factuality analysis on location estimation

	Accuracy[%]
Helpangina	81.00
Mumps	79.00
Infectious gastroenteritis	76.00
Epidemic conjunctivitis	62.00
Mycoplasma	78.00
Pool heat	71.00
Hand-foot-and-mouth disease	65.00

Table 3: Experimental results of factuality analysis on location estimation

We also evaluated the accuracy of factuality analysis for locations for users infected with diseases used in Sec. 4.1. Table 3 shows the results. The accuracy was 62-81[%].

5. Error Analysis

5.1. Infectious diseases detection

We observed the variation of accuracy across the diseases. For example, infectious diseases such as "mumps" are fundamentally children's infections. In this research, we treat it as a positive case even if the person related to the user is infected, not the person who posted the tweet sentence. As an example, when there is a tweet sentence "My son has caught a mumps", that tweet sentence is a positive example. Also, There was a tendency that there were a lot of tweets showing that they are prevalent in schools where their children attend, such as "The mumps in my school where my children are attending are popular", "The child of the kindergarten the daughter passes through is suffering from mumps". For this reason, accuracy is considered to be low with respect to infectious diseases prevailing in children such as mumps and hand-foot-and-mouth disease. In order to improve the accuracy more, it is considered effective to increase the learning data or reduce the deviation.

5.2. Location estimation

As a factor of misclassification for location estimation, some tweets contained two or more location words, many of which were misclassified. Although such tweets are not majority of the data, we need a method to choose the appropriate word among several location-indicating words for further improvement of accuracy. In order to improve the accuracy more, it is considered effective to increase training data, improvement of noise removal method, increase of location information database. Currently, our noise removal method uses only special symbols such as musical notes. Removing more special symbols and emoticons will contribute to improving accuracy. Currently the location database contains only nationwide known locations or events, such as "Tokyo Sky Tree" and "Tottori Dune Hill" for the spot name, "Tokyo Game Show" for the event name,



Figure 5: Example of Infectious Disease Map.

so it is thought that accuracy will improve if we add even a little popular locations/events.

6. Visualization of location information

Figure 5 shows the infectious disease map created based on the result of estimating the location of Helpangina infected person. The mark shown on the map shows the following meaning.

Red mark: One infected person,

Blue mark and number: 2 to 10 people infected,

Yellow mark and number: More than 10 people infected.

7. Conclusions

We proposed a method to create an infectious disease map automatically from Twitter. We perform factuality analysis both for disease infection events and location indication for improved accuracy. Experimental results showed that our method perform well on various kinds of diseases. Future work includes the increase of training data and location databases to improve the system accuracy.

8. Bibliographical References

Aramaki, E., Maskawa, S., and Morita, M. (2011). Twitter catches the flu: Detecting influenza ep-idemics using twitter. In *Proceedings of EMNLP 2011*, pages 1568–1576.

Chandra, S., Khan, L., and Muha-ya, F. B. (2011). Estimating twitter user location using social interactions-a content based approach. In *Proceedings of SocialCom/PASSAT 2011*, pages 838–843.

Charles-Smith, L. E., Reynolds, T. L., Cameron, M. A., Conway, M., Lau, E. H., Olsen, J. M., Pavlin, J. A., Shigematsu, M., Streichert, L. C., Suda, K. J., and Corley, C. D. (2015). Using social media for actionable disease surveillance and outbreak management: a systematic literature review. *PloS one*, 10(10).

Cheng, Z., Caverlee, J., and Lee, K. (2010). You are where you tweet: a content-based ap-proach to geo-locating twitter users. In *Proceedings of CIKM 2010*, pages 759–768.

Dredze, M., Paul, M. J., Bergsma, S., and Tran, H. (2013). Carmen: A twitter geolocation system with applications to public health. In *AAAI Workshop on Expanding the Boundaries of Health Informatics Using AI (HIAI)*.

Han, B., Cook, P., and Baldwin, T. (2012). : Geolocation prediction in social media data by finding location indicative words. In *Proceedings of COLING 2012*, pages 1045–1062.

Iso, H., Wakamiya, S., and Aramaki, E. (2016). Forecasting word model: Twitter-based influenza surveillance and prediction. In *Proceedings of COLING 2016*, pages 76–86.

Kanouchi, S., Komachi, M., Oka-zaki, N., Aramaki, E., and Ishikawa, H. (2015). Who caught a cold ? - identifying the subject of a symptom. In *Proceedings of ACL (1) 2015*, pages 1660–1670.

Kitagawa, Y., Komachi, M., Ara-maki, E., Okazaki, N., and Ishikawa, H. (2015). Disease event detection based on deep modality analy-sis. In *Proceedings of ACL (Student Research Workshop) 2015*, pages 28–34.

Li, J. and Cardie:, C. (2013). Early stage influenza detection from twitter. In *CoRR abs/1309.7340*.

Li, C. and Sun, A. (2014). Fine-grained location extraction from tweets with temporal awareness. In *Proceedings of SIGIR 2014*, pages 43–52.

Paul, M. J. and Dredze, M. (2011). You are what you tweet: Analyzing twitter for public health. In *Proceedings of ICWSM 2011*.

Sadilek, A., Kautz, H. A., and Silenzio, V. (2012). Predicting disease transmission from geo-tagged micro-blog data. In *Proceedings of AAAI 2012*.

Sugiya, T., Sirakawa, M., Hara, T., and Nishio, S. (2013). User position estimation method when posting tweets with supervised reinforcement learning. In *Proceedings of IPSJ SIG Technical Reports*.

Suzuki, Y., Kaji, N., Yoshinaga, N., and Toyoda, M. (2015). Geolocation estimation of mi-crobloggers utilizing their recent posts (in japanese). In *Proceedings of the 7th forum on Data Engineering and Information Management (DEIM)*.

Reusable workflows for gender prediction

Matej Martinc, Senja Pollak
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
{matej.martinc, senja.pollak}@ijs.si

Abstract

This paper presents a system for author profiling (AP) modeling that reduces the complexity and time of building a sophisticated model for a number of different AP tasks. The system is implemented in a cloud-based visual programming platform ClowdFlows and is publicly available to a wider audience. In the platform, we also implemented our already existing state of the art gender prediction model and tested it on a number of cross-genre tasks. The results show that the implemented model, which was trained on tweets, achieves results comparable to state of the art models for cross-genre gender prediction. There is however a noticeable decrease in accuracy when the genre of a test set is different from the genre of the train set.

Keywords: author profiling, workflows, reusability and replicability

1. Introduction

In recent years, the data from the user-generated content (UGC) has become a popular resource for studies in natural language processing (NLP) and text mining, especially due to its accessibility, size and a near real-time publishing. Messages from the social media by personal users display a trend towards more personal content (Naaman et al., 2010), which makes it a useful dataset for learning about the users' characteristics, related to demographics, psychological profile or mental health. The field that uses text documents to discover users' attributes automatically is known as author profiling (AP) and includes tasks, such as the prediction of author's gender (Rangel et al., 2017), which is also at the core of this paper, but also the prediction of author's age (Rangel et al., 2016), personality type (Verhoeven et al., 2016) or language variety (Rangel et al., 2017).

While deep learning approaches are gradually taking over different areas of NLP, the best approaches to AP still use more traditional classifiers and require extensive feature engineering (Rangel et al., 2017). This test-driven approach relies heavily on trying out different feature and parameter configurations, which can be very time consuming. To accelerate this process of model building, we propose the implementation of a system that enables faster model prototyping by employing the visual programming (VP) paradigm. The VP paradigm simplifies the representation of complex procedures into visual arrangements of their building blocks and consequently enables faster and more efficient model prototyping and model fine-tuning.

The contribution of this paper is two-fold and is anchored in recent trends in reusability and replicability in NLP. Our AP system implementation is built in an online visual programming environment, which is to the best of our knowledge the first of its kind. The proposed approach to constructing prototype AP models through visual workflows of NLP components contributes to the reusability in NLP on the level of specific components, as well as on the level of the entire gender prediction models. The pretrained models for five languages (four of which we developed for PAN 2017 (Martinc et al., 2017)) can be reused in new experiments and tested on new datasets. This has been showcased in cross-genre experiments in Section 4, which also show that our model, trained on tweets, is robust and achieves results comparable to the models built specifically for the cross-genre task. On the other hand, specific components and the entire workflow can be reused for building new models for AP as well as for other text classification tasks. Even if the PAN community already encourages code sharing (`https://github.com/pan-webis-de`), the simplified visual representation reduces the technical skills required for building complete systems and opens the experimentation also to the users outside of the programming community (e.g., linguists) which is very important in such an interdisciplinary field as AP. The experimentation is also easier/faster, since no specific software installation is required.

By publishing the workflows of our experiments, this work contributes to the paradigm of transparent experimentation and replicability. Since our experiments are proposed as online workflows, they can be easily reproduced and used for comparison of results.

2. Background technologies and related work

The platform used for the implementation of the system for rapid prototyping, implementation and testing of models is ClowdFlows (Kranjc et al., 2012), (`http://clowdflows.org`). ClowdFlows is a cloud-based web application for composition, execution and sharing of interactive data mining workflows.

ClowdFlows enables visual programming and has a GUI in which a user connects processing components (i.e. widgets) into executable pipelines (i.e. workflows) on a design canvas. It has a web based GUI for building workflows, runs in all major browsers and requires no installation.

ClowdFlows visual programming platform was chosen for the implementation of the system for AP modeling because of its open source nature that allows anybody to expand its widget stack and because of its system for publishing workflows together with the data, allowing reproducibility of our results. None of the considered open source alternatives, such as the GATE platform (Cunningham, 2002)

(General Architecture for Language Engineering) or the open source implementation of the IBM's Unstructured Information Management Architecture (Apache UIMA) (Ferrucci and Lally, 2004) enable workflow and data sharing. TextFlows (Perovšek et al., 2016), a fork of ClowdFlows specialized for NLP and text mining workflows, was also initially considered for the implementation of the AP modeling system. The reason why this platform was not chosen were its format restrictions, or more specifically, the enforced Annotated Document Corpus (ADC) format, which would increase the memory consumption of the system, prolong the execution time and put heavy restrictions on the size of the input corpora.

Even if—to the best of our knowledge—the AP tools have not yet been made available in form of executable workflows, the field of AP has a strong research tradition. Approaches for predicting author's age (Rangel et al., 2016), language variety (Rangel et al., 2017) or personality type (Verhoeven et al., 2016) have been proposed, while gender prediction has the strongest tradition. The earliest attempts at AP that covered gender identification started with Koppel et al. (2002), who used parts of the BNC, but continued on other corpora, such as the ACL corpus of scientific papers (Vogel and Jurafsky, 2012) or UGC corpora, esp. in the context of PAN shared tasks (Rangel et al., 2017). This paper also builds upon our contribution to the PAN 2017 shared task (Martinc et al., 2017), but decomposes the approach to the reusable building blocks, makes the models available for new experimentation and tests the model on cross-genre settings.

3. Widgets for rapid model prototyping

We implemented new tools in the ClowdFlows platform that would enable its users a fast test driven development in the field of AP. These tools are presented as graphical components (widgets) that can be connected into workflows and range in functionality from tools for corpus representation and NLP to more general classification tools. This section briefly describes the newly implemented tools and how they can be combined into a workflow for building state of the art classification models:

Load Corpus From CSV widget takes a comma-separated values (CSV) file containing the corpus with texts and annotations and converts it into a Pandas dataframe (McKinney, 2011) that enables further processing. Lines of the dataframe present documents and columns present different text annotations or the texts itself. The same format applies to the input CSV file. The first line of the loaded CSV file should also contain names of the columns.

Select Corpus Attribute widget takes a corpus in a dataframe form as an input and returns only the column of the dataframe defined by the user. This enables extracting only the part of the corpus that is needed, reducing the computational complexity of further processing. The column is returned as a Python list.

Remove Punctuation widget takes a column of the corpus (in the form of a Python list), usually the one containing texts, and returns a list of texts without the most common punctuation characters.

Remove Stopwords widget takes a column of the corpus containing texts and returns a list of texts without stopwords. The user can choose between stopwords lists for English, Portuguese, Slovenian or Spanish.

Tweet Cleaner widget takes a column of the corpus containing texts and by default replaces hashtags, mentions and URL-s with filler tokens (HTTPURL, TWEETMENTION and HASHTAG). Hashtags, mentions and URL-s can also be removed if the user chooses the *remove mode* in the widget parameters.

Affix Extractor widget takes a column of the corpus as an input and returns token affixes. The user can define the length and type of affixes (prefixes, suffixes and punctuation affixes, i.e. *beg-puncts* (Sapkota et al., 2015).

Count Patterns widget takes a column of the corpus containing texts and counts the specified patterns in the text. The user can specify the patterns as a comma separated list of word tokens or phrases, or select from already defined patterns, such as character floods and emojis. The user can also select if he wants raw counts or counts that are normalized by the number of characters in the text. By default, the widget returns a list containing the counts for each text in the corpus, however the widget can also return an integer representing the count of a specific pattern for the whole corpus.

Emoji Sentiment widget (Novak et al., 2015) takes a column of the corpus containing texts as an input and returns a sentiment score for every text. The sentiment is calculated as the sum of sentiment values of all the emojis in the text (no normalization is used).

TF-IDF Vectorizer widget takes a text column of the corpus as an input and returns a vector of calculated TF-IDF weights. The user can choose the type of tokens for which he wants the TF-IDF values, such as word, character and word bound character n-grams with different n values. Minimum and maximum document frequency of the token can be defined, meaning that all the tokens not fitting in the desired interval will be ignored. The tokens can optionally be lowercased before the TF-IDF calculation and different versions of TF-IDFs can be calculated (with smooth IDF and sublinear TF). The widget relies heavily on the TF-IDF vectorizer implemented in the Scikit library (Pedregosa et al., 2011).

Feature union widget takes heterogeneous features and a target value as inputs and returns a dataset containing the union of the features that can be feed as an input to different Scikit classifiers. The user can define weights for different types of features which influence the penalization of these features during the classifier learning phase.

Cross Validation widget takes a classifier and a dataset produced by a *Feature union* widget as an input

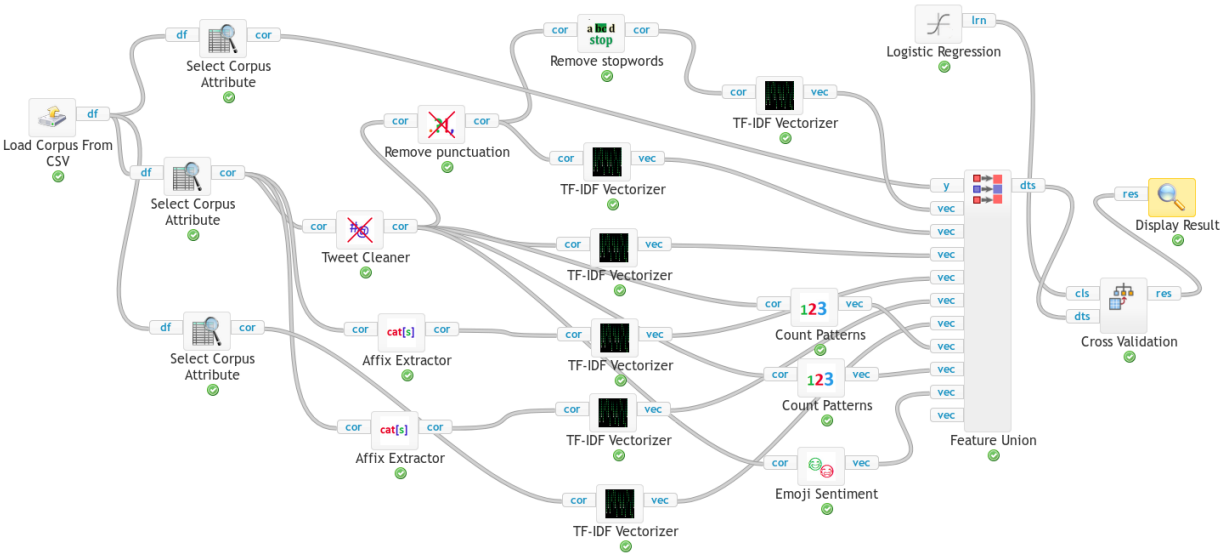


Figure 1: The model for gender classification implemented in the ClowdFlows platform (publicly available at `http://clowdflows.org/workflow/10620/`).

and conducts cross-validation of the classifier on the dataset. The user can define the number of folds and the metrics for evaluation (accuracy, precision, recall and f-score).

Display result widget is a simple widget that displays the results of the cross-validation test.

Besides the above mentioned widgets, ClowdFlows also contains different Scikit classifiers (Pedregosa et al., 2011) that can be used in model creation, such as Naive Bayes classifier, Logistic regression, Support Vector Classification, Linear Support Vector Classification, K-nearest neighbour classifier and Decision tree classifier. All these classifiers have specific parameters that can be tweaked for better performance.

The workflow in Figure 1 presents how we can combine the developed NLP widgets into a workflow that trains state of the art gender classification model. The workflow structure is nearly identical to the structure of our model used in the PAN competition (Martinc et al., 2017). We load the subcorpus[1] of the PAN 2017 corpus containing documents by 500 authors (each document contains around 100 concatenated tweets written by the author), part of speech (POS) tags and labeled gender from the CSV file by the *Load Corpus From CSV* widget. We use three *Select Corpus Attribute* widgets to extract text, gender labels and POS tags. Gender labels are directly connected to the *Feature Union* widget as a target value, while we do some further processing on text and POS tags. POS tags are TF-IDF weighted and converted into POS trigrams with the help of the *TF-IDF Vectorizer*. For text we build three levels of preprocessing (in the first level we just remove mentions, hashtags and URL-s, in the second level we additionally remove punctuation and in the third level we also remove stopwords), which are all later TF-IDF weighted. We also build different affix features with the *Affix extractor*, pattern counts features (character flood and emoji counts) created by *Count pattern* widgets and calculate document sentiment. All these heterogeneous features are combined in the *Feature Union* widget which builds a dataset. This dataset is used in the cross-validation experiments using Logistic Regression as a classifier and accuracy as a measure. The results are displayed in the *Display Result* widget.

Since this workflow is publicly available (`http://clowdflows.org/workflow/10620/`) it can be used as a template for building and testing models for many different AP tasks. Going to the published url of the workflow and clicking on the *Launch workflows! (Create a copy)* button generates a copy of the workflows together with its data, that a user can change without affecting the original workflow. This way, the published workflow can be modified to better fit some other AP task and data by, for example, adding/removing features and tweaking parameters, which is fast and simple because of the visual programming paradigm.

4. Model Reusability

Fast prototyping is important for the users who wish to develop new models quickly. On the other hand, there are users who just wish to use an already developed and trained model. For example, if we have a dataset and we want to annotate it with the gender of their author, pretrained models can be an optimal solution. For this reason we took our model for gender classification developed during the PAN 2017 shared task[2] and packed it in a widget *Gender Classifier* (for details see (Martinc et al., 2017)). The widget takes

[1]The number of tweets was limited to 50,000 to comply with the Twitter terms of use: `https://developer.twitter.com/en/developer-terms/agreement-and-policy`.

[2]The model achieved the second highest overall accuracy (for English gender prediction, the accuracy was 80,71%).

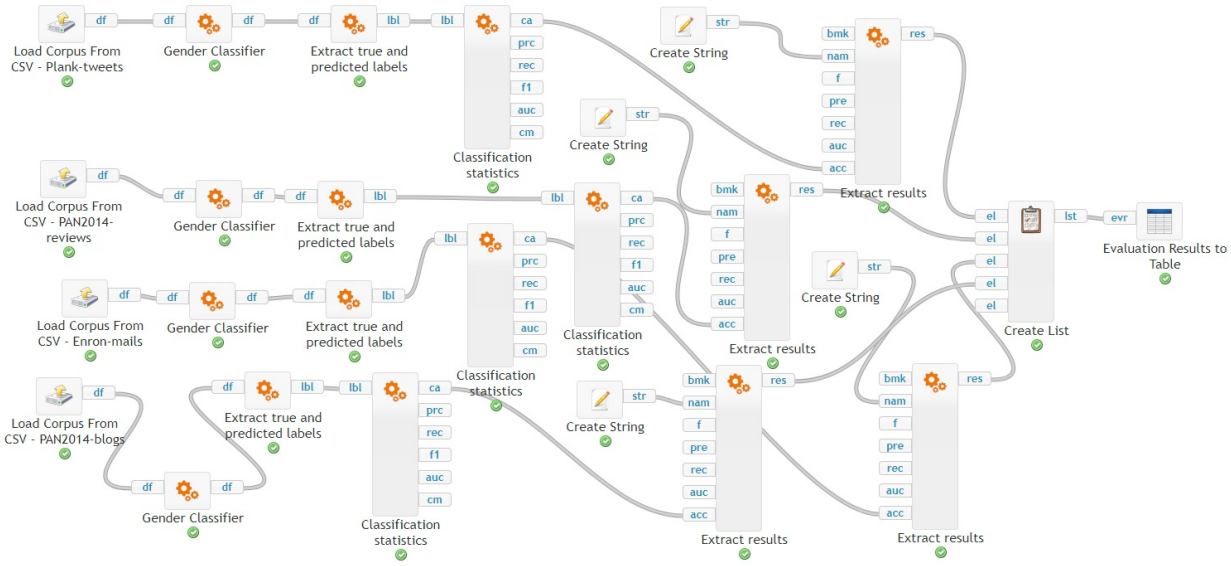


Figure 2: The workflow for cross-genre experiments conducted in the ClowdFlows platform (publicly available at `http://clowdflows.org/workflow/11042/`).

a Pandas dataframe as input and returns a dataframe with an additional column with predicted gender labels. The user needs to define the name of the column containing text documents as a parameter and choose the language of the text (English, Spanish, Portuguese, Arabic or Slovenian).

Even if the gender classification model was developed for the classification of tweets, we tested it not only on a separate corpus of tweets, but also in a cross-genre classification. The PAN 2016 AP shared task (Rangel et al., 2016)—and additional research conducted by part of the winning team (Medvedeva et al., 2017)—dealt with cross-genre classification, so we are able to compare the results that our model achieved on corpora of blogs and reviews with the other state of the art models built specifically for cross-genre performance.

Figure 2 shows the workflow (available at `http://clowdflows.org/workflow/11042/`) for the cross-genre experiments conducted on four corpora from different genres of UGC with labeled gender.

PAN2014-blogs (Rangel et al., 2014)[3]: the dataset consists of 2,278 blogs by 146 authors (73 male and 73 female). The blogs of the same author were concatenated to a single document, representing one author.

PAN2014-reviews : the PAN 2014 AP train set contains hotel reviews from 4,160 authors (2,080 male and 2,080 female). As above, the reviews of the same author were grouped to a single document.[3]

Plank-tweets (Plank and Hovy, 2015)[4]: dataset consists of 561 male and 939 female authors and is already preprocessed in terms of retweet removal and concatenation of tweets by the same author. We removed 378 female authors to balance the two classes for easier comparison with other datasets that are also balanced.

Enron-mails dataset (Klimt and Yang, 2004)[5] contains 517,431 e-mails of Enron employees, 105 male and 43 female. In this corpus emails are arranged by folders for specific users. The body of each e-mail was parsed by removing the header and reply text. We balanced the corpus by removing 62 male authors and concatenated e-mails of the same author to a single document.

We loaded all four corpora from the CSV files and converted them into a Pandas dataframe. The gender was predicted by our *Gender Classifier* widget. We used *Extract true and predicted labels* widget for extracting true gender labels and the labels predicted by the classifier that are used to calculate different metrics – we were interested in the accuracy of the model for which we used the *Classification statistics* widget. Widget *Extract results* is used to properly format the output of the widgets *Classification statistics* and *Create String* (which is used for adding a name of the corpus for the final table representation). *Create list* groups all the results, which are then finally presented in the table form with the *Evaluation Results to Table* widget. Results (Table 1) show that the genre of the train set matters. Our gender classification model, which was trained on tweets, proved far more accurate on tweets (Plank-tweets) than on other genres. Our model achieved very similar accuracies on blogs and reviews and achieved the lowest accuracy on emails. We can compare the results of our model to the results of the winners of the PAN 2016 AP shared task (Medvedeva et al., 2017). In the experiments where their model was trained on English tweets (train set from the PAN 2016 AP competition) they achieved 73.43% accuracy on PAN2014-blogs and 50% accuracy on PAN2014-reviews while mails were not one of their test corpora. The

[3]PAN2014 data: `http://pan.webis.de/data.html`

[4]`https://bitbucket.org/bplank/wassa2015/src`

[5]`https://www.cs.cmu.edu/~./enron/`

Corpus	Accuracy
PAN2014-blogs	65.75%
Plank-tweets	**74.69%**
Enron-mails	53.41%
PAN2014-reviews	63.73%

Table 1: Results of the cross-genre experiments (majority classifier for all the corpora is 50%). The model was trained on the tweets from the PAN 2017 AP training set and tested on four different genres (tweets, blogs, mails, reviews).

results show that our model—even if it was not built with cross-genre classification in mind—gives more constant accuracies for the two compared genres, achieving higher accuracy on reviews and lower on blogs.

5. Conclusion

This paper presents a system for rapid AP model prototyping together with the results of cross-genre gender classification experiments. The implemented system reduces the complexity and time of building a sophisticated model for a number of different AP tasks and also gives the user enough freedom and flexibility to allow for thorough exploration of different possibilities. We also demonstrate that our model for gender classification, that was developed for tweet classification in the PAN 2017 task, can be used on other genres and gives results that are comparable to the models built for cross-genre classification (we achieve higher results for reviews, but lower for blogs). There is however a noticeable decrease in accuracy when the genre of a test set is different from the genre of the training set. The presented work contributes to the replicability and reusability in NLP since all the components, models and experiments are available to the research community in the form of online workflows and widgets. In future work, the models for other languages will be tested on available datasets, while the constructed workflows will be used for other AP tasks.

Acknowledgements

The authors acknowledge the financial support from the Slovenian Research Agency through core research programme Knowledge Technologies (P2-0103) and the basic research project Resources, Tools and Methods for the Research of Nonstandard Internet Slovene (J6-6842). The authors also acknowledge the project ClowdFlows Data and Text Analytics Marketplace on the Web (CF-Web), which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 754549. We would also like to thank Iza Škrjanec and Katja Zupan for their contribution in the scope of the PAN 2017 shared task.

6. Bibliographical References

Cunningham, H. (2002). Gate, a general architecture for text engineering. *Computers and the Humanities*, 36(2):223–254.

Ferrucci, D. and Lally, A. (2004). Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Klimt, B. and Yang, Y. (2004). The Enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning*, pages 217–226. Springer.

Koppel, M., Argamon, S., and Shimoni, A. R. (2002). Automatically categorizing written texts by author gender. *Literary and Linguistic Computing*, 17(4):401–412.

Kranjc, J., Podpečan, V., and Lavrač, N. (2012). ClowdFlows: A cloud based scientific workflow platform. In Peter A. Flach, et al., editors, *Proc. of ECML/PKDD (2)*, volume 7524 of *LNCS*, pages 816–819. Springer.

Martinc, M., Škrjanec, I., Zupan, K., and Pollak, S. (2017). Pan 2017: Author profiling-gender and language variety prediction. *CLEF 2017 Evaluation Labs and Workshop – Working Notes Papers*.

McKinney, W. (2011). Pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, pages 1–9.

Medvedeva, M., Haagsma, H., and Nissim, M. (2017). An analysis of cross-genre and in-genre performance for author profiling in social media. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 211–223. Springer.

Naaman, M., Boase, J., and Lai, C.-H. (2010). Is it really about me?: Message content in social awareness streams. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative work*, pages 189–192. ACM.

Novak, P. K., Smailović, J., Sluban, B., and Mozetič, I. (2015). Sentiment of emojis. *PloS one*, 10(12):e0144296.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, 12:2825–2830.

Perovšek, M., Kranjc, J., Erjavec, T., Cestnik, B., and Lavrač, N. (2016). TextFlows: A visual programming platform for text mining and natural language processing. *Science of Computer Programming*, 121:128—-152.

Plank, B. and Hovy, D. (2015). Personality traits on twitter -or- how to get 1,500 personality tests in a week. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*. Lisbon, Portugal.

Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., and Daelemans, W. (2014). Overview of the author profiling task at PAN 2014. In *CLEF 2014 Evaluation Labs and Workshop – Working Notes Papers*. CEUR.

Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., and Stein, B. (2016). Overview of the 4th Author Profiling Task at PAN 2016: Cross-genre evaluations. In *CLEF 2016 Working Notes*. CEUR-WS.org.

Rangel, F., Rosso, P., Potthast, M., and Stein, B. (2017). Overview of the 5th Author Profiling Task at PAN 2017:

Gender and Language Variety Identification in Twitter. In Linda Cappellato, et al., editors, *Working Notes Papers of the CLEF 2017 Evaluation Labs*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, September.

Sapkota, U., Bethard, S., Montes-y-Gómez, M., and Solorio, T. (2015). Not all character n-grams are created equal: A study in authorship attribution. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 93–102.

Verhoeven, B., Daelemans, W., and Plank, B. (2016). Twisty: A multilingual twitter stylometry corpus for gender and personality profiling. In *LREC*.

Vogel, A. and Jurafsky, D. (2012). He said, she said: Gender in the ACL anthology. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 33–41. ACL.

Knowing the Author by the Company His Words Keep

Armin Hoenen[1], **Niko Schenk**[2]
[1]CEDIFOR, [2]Applied Computational Linguistics Lab (ACoLi)
Goethe University, Frankfurt am Main, Germany
{hoenen,nschenk}@em.uni-frankfurt.de

Abstract

In this paper, we analyze relationships between word pairs and evaluate their idiosyncratic properties in the applied context of authorship attribution. Specifically, on three literary corpora we optimize word pair features for information gain which reflect word similarity as measured by word embeddings. We analyze the quality of the most informative features in terms of word type relation (a comparison of different constellations of function and content words), similarity, and relatedness. Results point to the extraordinary role of function words within the authorship attribution task being extended to their pairwise relational patterns. Similarity of content words is likewise among the most informative features. From a cognitive perspective, we conclude that both relationship types reflect short distance connections in the human brain, which is highly indicative of an individual writing style.

Keywords: Word pairs, word relations, authorship attribution

1. Introduction

In traditional linguistics, there exists a famous saying that one should know a word by the company it keeps (Firth, 1957), which informally describes the meaning of a particular word by the context in which it occurs. In this paper, we investigate sets of frequently appearing similar words and their relations. To this end, we make use of word embeddings as a substitute and ground our work on related applications of distributed word representations and their applications (Mikolov et al., 2013a; Mikolov et al., 2013b; Mikolov et al., 2013c), which allows us to investigate relationships and even associations between words. Word embeddings have become popular recently and have been successfully applied to a variety of NLP tasks. Generally, word embeddings are believed to capture distributional similarity with an implication to semantic similarity. Kiela et al. (2015), for example, investigate relatedness and similarity and find embeddings to be specializable for both phenomena.

In our work, we focus our word pair-based methodology on the particular domain of authorship attribution. From an interpretation perspective, we conjecture that our proposed method allows to inspect not only single words in isolation and ask *what kind of word choices may characterize certain authors* (Marsden et al., 2013), but also to focus on properties of latent syntactic and semantic relations *between* words.

2. Word Relations & Human Cognition

In this study a special focus is laid on word pairs and their relations. Those can be understood as edges in a wider network and thus represent a first step towards modeling true semantic connections: Although finding a direct neurological correlate for words is a rather complex and controversial endeavour, many methods and theories even in psychology implicitly or explicitly operate with such networks. In computational linguistics, for instance, word nets have become a popular way of rationalizing relationships between concepts manifested through words. Especially theories on the mental lexicon belong to this sphere, cf. Elman (2004); Libben and Jarema (2002).

Accepting some neural implication of words, a most fundamental distinction is between function words and content words. The former carry few to no meaning, have been characterized as belonging to closed classes not easily incorporating neologisms and finally they function as markers of grammatical, syntactic or discourse related functions. Content words on the other hand are the meaningful referential actants or arguments around which all relations are built. Cognitively, both fundamental word classes imply different activity localized at different regions in the brain, see for instance Gordon and Caramazza (1982); Bradley and Garrett (1983); Friederici (1985); Diaz and McCarthy (2009). Shore (1995) referring to Bates et al. (1988) mention a left vs. right hemisphere contrast on the one hand and an anterior vs. posterior opposition on the other.

One possible hypothesis is thus that *function word - content word* (FC) connections display other, distinct properties from *content word - content word* (CC) and *function word - function word* (FF) connections, when inspecting word pair relations.

Neurologically, the latter two (CC & FF) could happen to be local connections (which could point to dendritic connectivity), whilst the former could span longer distances (pointing to axonic connectivity). In neurocomputation it is well perceived and taken as a basis for models that neurons have two main types of connectors passing on electrical signals: dendrites and axons (Hameroff, 2010). Dendrites transmit activation via synaptic gaps to neighboring cells. Activation spreads through neighbor's neighbors and so forth but decreases in strength with distance. New dendrites are formed even in adult life (Tavosanis, 2012) and thus their connectivity patterns may be more subject to differences over time and between individuals. Axons (usually one per neuron) are threshold dependent (i.e. they only "fire" once the activation of the cell has surpassed a critical activation level). They transmit and reinforce the signal information over longer distances (Purves et al., 2006, pp.1050). Consequently, if function words and content words are located in different brain regions, their connectivity type should not be characterized exclusively by dendritic

connections. Since neuroimaging techniques may be hardly able to visualize differentiatingly on-the-fly axonic activation pathways in contrast to dendritic ones the claims made here are most likely better testable through other than direct neurological imaging techniques.[1] If to this end distributional similarity would entail or be correlated with the cognitive connection between distributionally similar words, the properties of such connections could be reflected by the textual data. If FF and CC connections roughly represented neural short distance conncetions whereas FC connections represented long distance connections, if furthermore short distance connections are characterized more by dendritic connectivity, then a plausible hypothesis investigated in this paper could be that FF and CC connections are more plastic (they are more subject to interindividual differences) than FC connections which involve less flexible axonic connectivity. In turn this would entail that in authorship attribution those features (CC and FF) would be more informative, since they are more individual.

Authorship attribution is the task of identifying the unknown author of a textual document. An overview of current authorship attribution methods is given by Stamatatos (2009), consider also Rudman (1997); Baayen et al. (2002); Burrows (2002); Koppel and Schler (2003); Argamon (2008); Luyckx and Daelemans (2008); Spracklin et al. (2008); Luyckx (2011); Pennebaker (2011); Rybicki and Eder (2011); Smith and Aldridge (2011); Marsden et al. (2013); Evert et al. (2015); Eder et al. (2016); Markov et al. (2016). While in authorship attribution studies have focussed on various different aspects and linguistic levels, a look onto word pairs with an implication to word type has—to the best of our knowledge—not yet been taken. We fill this gap and look at the connection types when using the author as a class label in a machine learning setup by optimizing the feature vectors. This allows us to distinguish between more and less informative features.

3. Related Work

Relationships between two and more words have been researched for instance in psycho- and computational linguistics. Naturally, binary word relations are a subset of n-ary word relations. The identification of pairwise semantic word relations such as synonymy, hypernymy, meronymy, but also noun-compound relations, relations between named entities or semantically typed relations are among the areas of research where word pairs have been focussed (Hearst, 1992; Pantel et al., 2004; Roark and Charniak, 1998; Berland and Charniak, 1999; Costello et al., 2006; Strube and Ponzetto, 2006).

The specific type of relations, we will focus on here is function words and content words with an implication to authorship. Typically function words are high frequency words and the most frequent words of texts strongly tend to be function words, cf. Islam and Hoenen (2013). In determining word pair relations, a study using high frequency words vs. low frequency words is reported by Davidov and Rappoport (2006). Their focus is not authorship attribution but the identification of word categories. A more fine grained analysis using word class instead of the rough binary difference function-content word was for instance pursued by Hasegawa et al. (2004), who found the relations between named entities to be informative or Widdows and Dorow (2002), who extract informative noun noun relations.

In the psycholinguistic and linguistic literature, some studies focus on the distinction between function and content words albeit with no explicit focus on authorship and seldom at word pairs. For instance, Corver and van Riemsdijk (2001) inspect some syntactic, distributional and lexical patterns and structures for function and content words. Bell et al. (2009) consider the predictability of English content and function words in discourse. On a related note, word pair features have received special attention in the recognition of implicit discourse relations (Biran and McKeown, 2013). Lexical access latencies, which are correlated with frequency are investigated for instance by Segalowitz and Lane (2000). While thus in the computational linguistics literature, word pair relations have been extensively researched, a focus or look to the categories function vs. content word is rather rare. On the other hand (psycho-) linguistic literature often applied these categories, but a focus on word pairs is rather rare.

Looking to authorship attribution, traditionally function words have been used much, moreover the application of both function and content words is not new, see for instance Koppel et al. (2009). Garcia and Martin (2006) look at the ratio of function to content words of a text. As a third (sub)field, authorship attribution thus uses both the distinction between function and content words and focusses on authorship, but to our best knowledge, the use of pairwise word patterns is rather rare and such a use under the distinction between content and function words novel.

4. Experiment

4.1. Corpora

We use two corpora provided by the computational stylistics group[2] since they cover two languages for which large word nets do exist: an English prose corpus, *A Short Collection of British Fiction* and a German one, *German Prose*. A third corpus comes from the Japanese Institute for Japanese Language and Linguistics,[3] the Meiroku corpus, containing short Japanese newspaper articles from the 19th century.[4] The three corpora vary in a large number of parameters, such as number of texts (German 66, English & Japanese 26), the number of authors (German 21, English 10, Japanese 13), the sizes of the respective texts, the time span of text creation, the genre and most of all in language and writing system, for more details see Table 1. The variety is of such a dimension, that any similar result obtained on all corpora has a very low likeliness to be caused by inherent corpus or sampling similarity.

[1] Though slightly outdated, Koch and Zador (1993) describe how dendrites are barely visualizable on microscopes at the time of publication.

[2] https://sites.google.com/site/computationalstylistics/

[3] https://www.ninjal.ac.jp/english/database/type/corpora/

[4] http://pj.ninjal.ac.jp/corpus_center/cmj/meiroku/

language	mean (T)	sd (T)	mean (S)	range (T)
EN	245,292	182,066	16,501	41,129 − 973,341
DE	84,187	101,221	5,070	13,993 − 607,144
JP	1,949	816	91	533 − 4,863

Table 1: Authorship corpora employed in this study and their properties. T = tokens, S = sentences.

4.2. Collecting Word Pairs

As a preprocessing step, we lowercase all texts and delete all punctuation symbols as we are interested in word relations only. For each text, we construct word embeddings using *word2vec*[5] with default settings. For any word in a specific text T_i being part of a corpus C, one can obtain the m most similar words according to the embeddings of this text alone, $0 < m < |T_i| - 1$. The word embedding vectors v_j for all words $w_j \in T_i$ of a specific text have a fixed number of dimensions (here 100). We choose the n most frequent words of the union of all texts in the corpus and denote this set MFW, $0 < n < \sum_{i=1}^{|C|} |T_i|, T_i \in C$. For each most frequent word $mfw_i \in MFW$, we collect the m most similar neighbors[6] for each text T_i. If the most frequent word is not in the text, the set contains m times ϵ. Otherwise, similarity for the current most frequent word with each other word $\in T_i$ is defined through some established vector similarity (here: cosine similarity) between the word pairs' vectors, $\delta(w_j, w_k) = cos(v_j, v_k)$. Thus, for each text T_i, for each word $w_i \in MFW$, we obtain a set with m most similar neighbors in T_i. The superset containing all such sets from one text is called W_t and the set of all W_t is called W_c.[7]

In the next step, we collect the set of all unique word pairs (w_i, n_i), where w_i is a most frequent word and n_i is any (most similar) neighbor of w_i in any text, $w_i \in MFW$, $n_i \in W_c$, $w_i \neq n_i$. The set of all unique such pairs U is used to construct a $|U|$-dimensional vector. For each word pair $(u_1, u_2) \in U$ with u_1 the most frequent word and u_2 a similar neighbor, a vector for each text is constructed and assigned 1 if the respective text's W_t in the set corresponding to the actual most frequent word's (u_1) neighbors contains u_2, otherwise the value is 0. For optimization in a machine learning setting, we assign the author as class label to the so constructed vector.

For an illustration see Table 2. Columns represent the feature vectors per text, which are used for optimization. 1 means the text at hand contains this word pair as one which according to the embeddings of the text has a non trivial similarity relation (the right word is among the n most similar 'neighbors' of the left, most frequent word given the embeddings of the actual text). 0 means such a similarity relation is absent. The last row contains the author of the text as class label and the first column makes features (word pairs) explicit. We choose the most frequent $1,000$ words and the most similar 100 neighbors, since they have been shown to yield good performances on an authorship attribution experiment (Hoenen, 2017). We then conduct an optimization for Information Gain (IG) on the above described binary vectors (1 per text), using the WEKA machine learning environment (Hall et al., 2009). IG reduces the set of dimensions of the original vectors to the most informative ones by iteratively analysing additive performative gains (and ranks features for their informativity). We analyze the most informative features (word pairs) looking at their connection type (FC,CC,FF). In order to gain more insight, we also analyze the semantic similarity of CC pairs. For English, we additionally look at synonymy and relatedness.

Feature	T_1	T_2	...
the-then	1	0	
basket-grandma	1	0	
because-ten	0	1	
green-red	1	0	
car-automobile	1	1	
car-engine	0	0	
...	...	...	
Class	author1	author2	

Table 2: Author/text-wise representation of feature vectors.

For identifying function words, we bootstrap a function word lexicon, applying a heuristic: we extract all words not tagged as one of the tags starting with or equalling (NN,NE,ADV,ADJA,ADJD,TRUNC,V,FM,XY) for German from the Tiger corpus (Brants et al., 2004); for English from the Brown corpus (Francis and Kucera, 1979) all words not tagged as or having a tag starting with (v,to+vb,rb,n,jj,fw, punctuation-tags); for Japanese, *meishi, doushi, keyoushi, keijoushi, eitango, romajimon, kanbun* and *fukushi* were excluded. We give the proportions of incidences of each connection type before and after optimization.

For semantic similarity of CC pairs, we look at all features, where both words are present in WordNet (Miller, 1995; Fellbaum, 1998) or GermaNet (Hamp and Feldweg, 1997; Henrich and Hinrichs, 2010). We then compute the average similarity of all synsets of both, according to Jiang and Conrath (1997), which has been shown by Gurevych and Niederlich (2005) to have a high correlation with human judgements. We compute the proportion of wordpairs similar above a threshold (here 0.5) among all WordNet or GermaNet pairs.

For computational modeling, we employ the German standard interfaces provided from GermaNet, for English we

[5] https://code.google.com/archive/p/word2vec/

[6] We call these words neighbors, which is a more or less arbitrary choice made in order to emphasize on conceptual and functional proximity.

[7] A further level of abstraction allowing also cross language comparisons based on word embeddings would be the use of (normalized) ranks or rank distances in a neighbor vector instead of tokens.

use WS4J[8] and for Japanese the JAWJAW wrapper of the Japanese WordNet.[9] We extracted the first translation of each term, then we proceeded with the English terms.[10] For English, we additionally look at synonymy and relatedness, using the same datasets as Kiela et al. (2015). These are: a) lists from Nelson et al. (2004) reflecting experiments on free association pointing to how related any two terms are and b) a resource on synonymy, the MyThes thesaurus developed by the OpenOffice.org project.[11] Synonymy, compare Freitag et al. (2005) and sources therein, refers to words which appear in extremely similar contexts and can be used to express the same meaning. We look at all word pairs, where both words were related according to those resources and give numbers for how many of those were carried over into the optimization.

5. Results

The investigated numbers of features ($|U|$) were $251,348$ initial features for German, $668,242$ for English and $2,696$ for Japanese. An optimization through IG reduces the set of features in such a way, that the classification result is optimized (better with the new set, than with the old full set) if possible. The algorithm assigns each feature an estimated informativity and retains only those in the set of optimized features which surpass a certain threshold of informativity. WEKA records these IG assignments per feature and allows to inspect them in closer detail. Here, exemplarily we look at which features survive if we set the IG threshold at 0.1 including quite uninformative but no contradictory features, and 0.5 at the level of which already less than 10% of the features are present. The retained features are thus the most informative features (best characterize) for the author, cf. Tables 3–6 for the results.[12]

In Tables 3-6, the column *retained* gives the percentage of features surpassing a certain informativity level (IG thresh). The reduced feature set at IG thresh 0.5 contained overall $39,345$ features for English, where the entire feature set had contained $668,242$, thus the proportion in column retained is $\frac{39,345}{668,242} = 0.06$.

Column FF in Table 3 gives the proportion of all FF features, which at the current level of informativity have been retained within the more effective reduced feature set, given (denominator) all FF features in the entire set. FF features in entire feature set: $17,590$ (EN), $23,334$ (DE), $1,270$ (JP).

Analogously, in Table 4, column FC gives the proportion of all retained FC features. For English, the entire feature set contained $173,394$ FC features. Within the reduced feature set $12,608$ features were FC features, thus $\frac{12,608}{173,394} = 0.07$ is the percentage of FC features retained given all FC features measured. Initially there were $173,394$ FC features for English, $124,121$ (DE), 1310 (JP).

Corpus	IG thresh	retained	FF	Diff
English	0.1	0.97	0.99	+0.02
German	0.1	0.72	0.93	+0.23
Japanese	0.1	0.9	0.92	+0.02
English	0.5	0.06	0.14	+0.08
German	0.5	0.007	0.018	+0.11
Japanese	0.5	0.06	0.1	+0.04

Table 3: Statistics on function word pair features.

Corpus	IG thresh	retained	FC	Diff
English	0.1	0.97	0.98	+0.01
German	0.1	0.72	0.71	−0.01
Japanese	0.1	0.9	0.9	±0
English	0.5	0.06	0.07	+0.01
German	0.5	0.007	0.007	±0
Japanese	0.5	0.06	0.03	−0.03

Table 4: Statistics on function word–content word pair features.

Table 5 is analogous to the above tables, CC gives the percentage of retained CC features in contrast to overall retained features. $477,258$ (EN) CC features initially, $103,893$ (DE), 116 (JP).

For English, using the above mentioned data sets, synonymy and relatedness could be tested, results shown in Table 6. As in the above tables, percentages of retained synonymic/related features given all synonymic/related features in the initial set are contrasted to the overall rate of feature retention.

Finally, Table 7 shows the percentage of features which are captured by a word net (both words of feature present) and have a semantic word net based similarity of more than 0.5. Column Sim gives their proportion on the entire initial feature set, while Sim retained gives the percentage in the reduced set of retained features (optimization). For German $12,510$ word pairs of the initial set were such that both words were covered by GermaNet. $5,600$ of those were similar above 0.5 semantic similarity (not IG thresh). This ratio is given in column Sim. In the reduced feature set, a subset of $10,310$ word pair features were covered by GermaNet with $4,716$ being similar above the similarity threshold. Thus through optimization the proportion of similar features given testable features increased, the amount of increase is displayed in column Diff. A column named Diff is used analogously also in the other tables and allows to see which features increase proportions in the optimized feature sets.

6. Discussion

The results (especially in columns labeled Diff) were similar in all three languages and corpora. This is despite vast differences in the corpora and in the features extracted (for the time being in a binary representation). It entails that some independent explanation should hold for the observed phenomena. The proportion of informative FF connections was larger than for FC connections, which in turn

[8] https://code.google.com/archive/p/ws4j/

[9] http://nlpwww.nict.go.jp/wn-ja/index.en.html

[10] JAWJAW provides similarity sets, but we found none of the actual pairs of the data.

[11] https://www.openoffice.org/lingucomponent/thesaurus.html

[12] For illustration, percentages are rounded to the second decimal digit or where necessary to the third.

Corpus	IG thresh	retained	CC	Diff
English	0.1	0.97	0.96	−0.01
German	0.1	0.72	0.68	−0.04
Japanese	0.1	0.9	0.69	−0.21
English	0.5	0.06	0.05	−0.01
German	0.5	0.007	0.005	−0.02
Japanese	0.5	0.06	0.009	−0.051

Table 5: Statistics on content word pair features.

Corpus	IG thresh	retained	syn/rel	Diff
English/syn	0.1	0.97	0.99	+0.02
English/syn	0.5	0.06	0.14	+0.08
English/rel	0.1	0.97	0.99	+0.02
English/rel	0.5	0.06	0.16	+0.1

Table 6: Statistics on informative synonyms (MyThes thesaurus) and informative related terms Nelson et al. (2004).

was larger than for CC connections. This would contradict the initial hypothesis that FF and CC connections show more individual plasticity and are thus author informative.

The current results could for instance be correlated with frequency rather than connection type. However, such a conclusion does not necessarily reflect the overall picture. Whereas there are very few and mostly very frequent function words implying a denser network and shorter pairwise paths (more effective dendritic coactivation), content words form much larger networks. This entails that only a restricted number of them might be susceptible to their neighbors spreading (dendritic) activation. These could be semantically similar terms which could align for instance with graded priming effects. The results in all three languages have shown that despite being overwhelmingly CC connections, semantically similar word pairs are author informative just as FF connections. Furthermore, for English this could be shown to hold true also for related and synonymic terms which further strengthens the assumption, that some kind of semantic relationship entails proximity. Finally, similar CCs were author informative despite their lower frequencies.

Additionally, the more informative the features became the larger was the proportion of retained similar word pair features (this trend was robust and is not just a consequence of the displayed IG limits). This may entail that what authors convey to be similar concepts is within the limits of similarity of the overarching language largely individual. Such a conclusion would nicely align with hypothesis of larger general semantic transparency and adaptability, see for instance related hypotheses in Eger et al. (2016). Since cross linguistically, the structures of synonymy (and ultimately word nets) seem to differ to certain extents, it seems plausible that interindividual differences to similar extents can exist within speakers of the same language. However, function words and functional morphemes such as regular endings are rigid and shared between speakers as to the mapping of form and function.

Corpus	IG thresh	Sim	Sim retained	Diff
English	0.1	0.001	0.001	±0
German	0.1	0.45	0.46	+0.01
Japanese	0.1	0.95	0.97	+0.02
English	0.5	0.001	0.002	+0.001
German	0.5	0.45	0.56	+0.11
Japanese	0.5	0.95	1	+0.05

Table 7: Statistics on word pair features retained, where both words according to the wordNets of their languages are similar above a threshold (here:0.5).

Furthermore as suspected above, if similarity is rather realized as dendritic short distance connection neurologically, a logical consequence would be a large plasticity since dendrons can adapt even in adult life, probably quicker so than axons the architecture and components of which are more complex. FF connections seem to be highly individual (again they could refect short distance dendritic connectivity). This time another plausible factor is their high frequency through which different pairs become more easily distinguishable. The important role of function words in authorship attribution, see for instance Burrows (2002); Pennebaker (2011) would be extended stating that not only frequencies and choice of function words are largely individual, but also patterns of their combination.

Yarowsky (1995); Yarowsky (1993) show that the trend for words to exhibit only one sense in a collocation is much less pronounced in collocations with function words. Thus, in his context it was also a crucial feature of word pair relations whether they were function or content words. Techniques such as pattern extraction for flexible patterns as presented in Schwartz et al. (2013) use word groups which can be larger than word pairs. These contain at least two high frequency words (often probably function words) and they have been shown to perform well and add value (as opposed to using only character n-gram or word n-gram features) on the identification of authors of very short Twitter messages, a disproportionally difficult authorship attribution task. This finding would align well with the here hypothesized findings.

7. Conclusion and Outlook

We presented a method, which uses word embeddings to identify pairwise word relations based on distributional criteria. We used these gained from single texts to optimize for recognition of an author and analysed the relations of author informative word pairs. Looking at these in terms of relation type showed similar patterns across three heterogeneous languages and corpora. The results point to similarity of content words (independent of word choice and lexicon) being subject to individual (authorial) fluctuation within the general characteristics of a language system although the extent of variability is not clear. Furthermore, the important role of function words for authorship attribution could be extended to relational patterns between them. A possible interpretation of FF connections and CC connections for similar words which deserves further research is that they could reflect short distance connections in the brain and by that token be highly adaptable and individual.

8. Bibliographical References

Argamon, S. (2008). Interpreting Burrows's delta: geometric and probabilistic foundations. *Literary and Linguistic Computing*, 23(2):131–147.

Baayen, H., van Halteren, H., Neijt, A., and Tweedie, F. (2002). An experiment in authorship attribution. In *6th JADT*, pages 29–37.

Bates, E., Bretherton, I., and Snyder, L. (1988). *From first words to grammar*. Cambridge University Press.

Bell, A., Brenier, J. M., Gregory, M., Girand, C., and Jurafsky, D. (2009). Predictability effects on durations of content and function words in conversational english. *Journal of Memory and Language*, 60(1):92–111.

Berland, M. and Charniak, E. (1999). Finding parts in very large corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 57–64. Association for Computational Linguistics.

Biran, O. and McKeown, K. (2013). Aggregated Word Pair Features for Implicit Discourse Relation Disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Volume 2: Short Papers*, pages 69–73, Sofia, Bulgaria.

Bradley, D. C. and Garrett, M. F. (1983). Hemisphere differences in the recognition of closed and open class words. *Neuropsychologia*, 21(2):155–159.

Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., König, E., Lezius, W., Rohrer, C., Smith, G., and Uszkoreit, H. (2004). Tiger: Linguistic interpretation of a german corpus. *Research on Language and Computation*, 2(4):597–620.

Burrows, J. (2002). 'Delta': a Measure of Stylistic Difference and a Guide to Likely Authorship. *Literary and Linguistic Computing*, 17(3):267–287.

Corver, N. and van Riemsdijk, H. (2001). *Semi-lexical categories: the function of content words and the content of function words*, volume 59. Walter de Gruyter.

Costello, F. J., Veale, T., and Dunne, S. (2006). Using wordnet to automatically deduce relations between words in noun-noun compounds. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 160–167. Association for Computational Linguistics.

Davidov, D. and Rappoport, A. (2006). Efficient unsupervised discovery of word categories using symmetric patterns and high frequency words. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 297–304. Association for Computational Linguistics.

Diaz, M. T. and McCarthy, G. (2009). A comparison of brain activity evoked by single content and function words: An fmri investigation of implicit word processing. *Brain Research*, 1282:38—-49.

Eder, M., Kestemont, M., and Rybicki, J. (2016). Stylometry with R: A package for computational text analysis. *R Journal*, 16(1).

Eger, S., Hoenen, A., and Mehler, A. (2016). Language classification from bilingual word embedding graphs. In *Proceedings of COLING 2016*. ACL.

Elman, J. L. (2004). An alternative view of the mental lexicon. *Trends in cognitive sciences*, 8(7):301–306.

Evert, S., Proisl, T., Vitt, T., Schöch, C., Jannidis, F., and Pielström, S. (2015). Towards a better understanding of Burrows's Delta in literary authorship attribution. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, pages 79–88, Denver, Colorado, USA. Association for Computational Linguistics.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

Firth, J. R. (1957). *Papers in linguistics, 1934-1951*. Oxford University Press.

Francis, W. N. and Kucera, H. (1979). Brown corpus manual. Technical report, Department of Linguistics, Brown University, Providence, Rhode Island, US.

Freitag, D., Blume, M., Byrnes, J., Chow, E., Kapadia, S., Rohwer, R., and Wang, Z. (2005). New experiments in distributional representations of synonymy. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 25–32. Association for Computational Linguistics.

Friederici, A. D. (1985). Levels of processing and vocabulary types: Evidence from on-line comprehension in normals and agrammatics. *Cognition*, 19(2):133–166.

Garcia, A. M. and Martin, J. C. (2006). Function words in authorship attribution studies. *Literary and Linguistic Computing*, 22(1):49–66.

Gordon, B. and Caramazza, A. (1982). Lexical decision for open-and closed-class words: Failure to replicate differential frequency sensitivity. *Brain and Language*, 15(1):143–160.

Gurevych, I. and Niederlich, H. (2005). Computing semantic relatedness of GermaNet concepts. In *Bernhard Fisseni, Hans-Christian Schmitz, Bernhard Schröder & Petra Wagner (Eds.), Sprachtechnologie, mobile Kommunikation und linguistische Ressourcen:Proceedings of Workshop "Applications of GermaNet II" at GLDV'2005*, pages 462–474. Peter Lang.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD Explorations*, 11(1):10–18.

Hameroff, S. (2010). The "conscious pilot"—dendritic synchrony moves through the brain to mediate consciousness. *Journal of biological physics*, 36(1):71–93.

Hamp, B. and Feldweg, H. (1997). Germanet - a lexical-semantic net for german. In *In Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Hasegawa, T., Sekine, S., and Grishman, R. (2004). Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 415. Association for Computational Linguistics.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th con-*

ference on Computational linguistics-Volume 2, pages 539–545. Association for Computational Linguistics.

Henrich, V. and Hinrichs, E. (2010). Gernedit - the germanet editing tool. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Hoenen, A. (2017). Using word embeddings for computing distances between texts and for authorship attribution. In *Proceedings of the 22nd International Conference on Applications of Natural Language to Information Systems (NLDB)*.

Islam, Z. and Hoenen, A. (2013). Source and translation classification using most frequent words. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1299–1305.

Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc of 10th International Conference on Research in Computational Linguistics, ROCLING'97*.

Kiela, D., Hill, F., and Clark, S. (2015). Specializing Word Embeddings for Similarity or Relatedness. In Lluís Màrquez, et al., editors, *EMNLP*, pages 2044–2048. The Association for Computational Linguistics.

Koch, C. and Zador, A. (1993). The function of dendritic spines: devices subserving biochemical rather than electrical computation. *Journal of Neuroscience*, 13:413–413.

Koppel, M. and Schler, J. (2003). Exploiting stylistic idiosyncrasies for authorship attribution. In *Proceedings of IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis*, volume 69, page 72.

Koppel, M., Schler, J., and Argamon, S. (2009). Computational methods in authorship attribution. *Journal of the Association for Information Science and Technology*, 60(1):9–26.

Libben, G. and Jarema, G. (2002). Mental lexicon research in the new millennium. *Brain and language*, 81(1):2–11.

Luyckx, K. and Daelemans, W. (2008). Authorship attribution and verification with many authors and limited data. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 513–520. Association for Computational Linguistics.

Luyckx, K. (2011). *Scalability Issues in Authorship Attribution*. Proefschriften UA-LW: taalkunde. VUB University Press.

Markov, I., Gómez-Adorno, H., Sidorov, G., and Gelbukh, A. (2016). Adapting cross-genre author profiling to language and corpus. In *Proceedings of the CLEF*, pages 947–955.

Marsden, J., Budden, D., Craig, H., and Moscato, P. (2013). Language Individuation and Marker Words: Shakespeare and His Maxwell's Demon. *PLoS ONE*, 8(6):63–88.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Mikolov, T., tau Yih, W., and Zweig, G. (2013c). Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2013)*. Association for Computational Linguistics.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Nelson, D. L., Mcevoy, C. L., and Schreiber, T. A. (2004). The University of South Florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, pages 402–407.

Pantel, P., Ravichandran, D., and Hovy, E. (2004). Towards terascale knowledge acquisition. In *Proceedings of the 20th international conference on Computational Linguistics*, page 771. Association for Computational Linguistics.

Pennebaker, J. W. (2011). *The secret life of pronouns*. Bloomsbury Press.

Purves, W., Sadava, D., Hillis, D., Orians, G., and Heller, H. (2006). *Biologie*. 7th Edition, German version edited by J. Makl. Spektrum.

Roark, B. and Charniak, E. (1998). Noun-phrase co-occurrence statistics for semiautomatic semantic lexicon construction. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 1110–1116. Association for Computational Linguistics.

Rudman, J. (1997). The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities*, 31(4):351–365.

Rybicki, J. and Eder, M. (2011). Deeper Delta across genres and languages: do we really need the most frequent words? *Literary and Linguistic Computing*, 26(3):315–321.

Schwartz, R., Tsur, O., Rappoport, A., and Koppel, M. (2013). Authorship attribution of micro-messages. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1880–1891.

Segalowitz, S. J. and Lane, K. C. (2000). Lexical access of function versus content words. *Brain and language*, 75(3):376–389.

Shore, C. M. (1995). *Individual differences in language development*, volume 7. Sage Publications.

Smith, P. W. H. and Aldridge, W. (2011). Improving Authorship Attribution: Optimizing Burrows' Delta Method. *Journal of Quantitative Linguistics*, 18(1):63–88.

Spracklin, L., Inkpen, D., and Nayak, A. (2008). Using the complexity of the distribution of lexical elements as a feature in authorship attribution. In *LREC*.

Stamatatos, E. (2009). A Survey of Modern Authorship Attribution Methods. *J. Am. Soc. Inf. Sci. Technol.*, 60(3):538–556.

Strube, M. and Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, volume 6, pages 1419–1424.

Tavosanis, G. (2012). Dendritic structural plasticity. *Developmental neurobiology*, 72(1):73–86.

Widdows, D. and Dorow, B. (2002). A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Yarowsky, D. (1993). One sense per collocation. In *Proceedings of the ARPA Human Language Technology Workshop, Princeton*, pages 266–271.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196. Association for Computational Linguistics.

Towards a Gold Standard Corpus for Variable Detection and Linking in Social Science Publications

Andrea Zielinski and Peter Mutschke
GESIS - Leibniz Institute for the Social Sciences
{[andrea.zielinski,peter.mutschke]}@gesis.org

Abstract

In this paper, we describe our effort to create a new corpus for the evaluation of detecting and linking so-called *survey variables* in social science publications (*e.g.*, *"Do you believe in Heaven?"*). The task is to recognize survey variable mentions in a given text, disambiguate them, and link them to the corresponding variable within a knowledge base. Since there are generally hundreds of candidates to link to and due to the wide variety of forms they can take, this is a challenging task within NLP. The contribution of our work is the first gold standard corpus for the variable detection and linking task. We describe the annotation guidelines and the annotation process. The produced corpus is multilingual – German and English – and includes manually curated word and phrase alignments. Moreover, it includes text samples that could not be assigned to any variables, denoted as negative examples. Based on the new dataset, we conduct an evaluation of several state-of-the-art text classification and textual similarity methods. The annotated corpus is made available along with an open-source baseline system for variable mention identification and linking.

Keywords: Text mining, semantic textual similarity, paraphrase detection, linking

1. Introduction

There is a growing trend to integrate research data into the scientific publication process. Open Science encourages scientific practices in which all research data should be interlinked and contextualized to enhance reproducibility and reusability of research results. Ideally, publications that report on a result of an empirical study should contain a direct link to the cited dataset and lead the reader directly to the research data that underlies the publication. However, in practice, this metadata is often missing. The potential of text and data mining technologies to automatically detect dataset citations has been addressed, *e.g.*, in the International Workshop on Mining Scientific Publications[1] and 4REAL Workshop[2](Cohen et al., 2016; Branco, 2012; Fokkens et al., 2013).

Interesting work in this direction has been carried out by Mariani et al. (2016) who seek to retrieve mentions of language resources (*e.g.*, corpora, lexica listed in the LRE map) by analyzing the content of the proceedings of the Language Resources and Evaluation Conference (LREC), *i.e.*, to discover the most relevant topics in this field. However, specialized solutions for specific use cases are still required within certain scientific areas.

For instance, social sciences publications often discuss the results of survey studies. A survey generally consists of several hundreds of variables, each of them representing a single survey question (*e.g.*, *Do you believe in Heaven?*). Social science papers, however, only focus on a particular selection of variables. In order to establish links between data and publications on a fine-grained level it is therefore necessary to link not only on study name level but also on the level of survey variables.

The problem of automatically linking a data citation text fragment to the corresponding dataset has been addressed in the INFOLIS project (Boland et al., 2012). However, advanced algorithms that are able to identify the survey variable mentions used in the underlying study and link them to a specific survey variable identifier in a knowledge base are still lacking.

We will refer to the problem as Variable Detection and Linking task, *i.e.*, given a set of variables of a particular survey and topic, all relevant mentions that refer to one of these variables are to be identified.

Within computational linguistics, the problem can be framed in two different ways. It can be conceptualized as an extension of the entity linking problem (Erbs et al., 2011; Rao et al., 2013), attempting to link citations to variables in a knowledge base or it can be phrased as a Recognizing Textual Entailment (RTE) problem, where the system should be able to identify whether a sentence entails a given candidate hypothesis or not (Dagan et al., 2013; Bentivogli et al., 2009).

Specifically the *textual entailment search* task (cf. (Harabagiu and Hickl, 2006), a variation of the RTE task, is adequate: Thus, the question and each answer option form the Hypothesis (H) and the system should be able to retrieve candidate entailing sentences from the document, defined as the Text (T). Also, an undirected relationship between the pairs of texts might hold, as in the related task of detecting Semantic Textual Similarity (STS) (Agirre et al., 2013).

The format of the dataset is released in the same fashion as the RTE data, consisting of pairs of Text (T) and Hypothesis (H), *i.e.*, in our scenario T corresponds to variable mentions in scientific documents and H corresponds to questions and answer sets obtained from the variable data catalog.

Negative pairs are created as combinations of verified Ts with *other* Hs, that is, Hs from the same topic but that cannot be linked to T. In correspondence to the real-life application setting, we also selected a high proportion of unrelated sentences. While the first set of positive and negative T/H pairs can be used for the subtask of *Variable Disambiguation*, the second set of unrelated sentences has been

[1] https://wosp.core.ac.uk/

[2] Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language

created for the subtask of *Variable Detection*.
Furthermore, we investigated the diverse types of citations in a qualitative and quantitative study. It is the first corpus for this task and we will make it available to the research community. It has a broad coverage of linkage types, showing that lexical semantics is important for obtaining good performance. The new corpus is intended to drive the development of NLP methods for the detection and linking of variable mentions and can be used for benchmarking them.

2. Use Case Description

The Variable Detection and Linking task is to automatically augment a plain text document with links to variables in order to annotate salient social science concepts. It assumes the existence of a knowledge base, covering all variables of interest. In social science, survey variables are generally listed in data catalogs such as ISSP[3] or ALLBUS[4]. While each survey (or questionnaire) is composed of a specific set of survey variables, only a subset of them might be cited in a publication. An illustration for the variable linking task is shown in Figure 1.
In our setting, the task focuses on classifying and linking mentions to one (or more) of the variable classes that are identified by a unique identifier in ALLBUS[5]. Variable-level information includes the question and subquestion text, an associated topic and a predefined set of answers (*i.e.*, the majority of survey questions are closed and respondents have to mark their choices w.r.t. the given response options). The wording of the questions and answers is generally well chosen according to common practices in survey design. An example of a survey variable is provided in Figure 2.

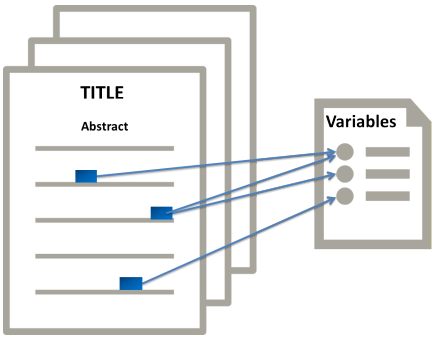


Figure 1: Linking Survey Variable Mentions in Scientific Publications

Identifying mentions of survey variables is a particular challenge in social science publications, because they usually appear in a wide variety of forms. Authors in this field often do not quote a variable literally but tend to paraphrase it, as exemplified in (1), or summarize more than one variable in a single sentence, as illustrated in example (2). For instance, the paraphrase in example (1) is used in our dataset to refer to variable v278 (cf. Fig. 2).

[3]International Social Survey Programme `http://www.issp.org/menu-top/home/`

[4]`https://www.gesis.org/en/allbus/`

[5]Variable IDs extracted from ALLBUS CUM. 1980-2014 for German and English (Allerbeck et al., 2016; Lepsius et al., 2016)

```
<variable v_id="v278" lang="English"
surey="ALLBUS_cumulated">
      <v_label>OPINION ON DUAL CITIZENSHIP
      </v_label>
      <v_topic>Ethnocentrism and
      Minorities</v_topic>
      <v_question>Using the scale on the card,
      please indicate the extent to which you agree
      with each statement. </v_question>
      <v_subquestion>Foreigners living in Germany
      should be able to acquire German citizenship
      without having to give up their own
      citizenship, dual citizenship should be
      possible. </v_subquestion>
      <v_answer a_id="1">Not available</v_answer>
      <v_answer a_id="2">Completely
      disagree</v_answer>
      <v_answer a_id="3">Disagree</v_answer>
      <v_answer a_id="4">Indifferent</v_answer>
      <v_answer a_id="5">Agree</v_answer>
      <v_answer a_id="6">Completely agree</v_answer>
      <v_answer a_id="7">No answer</v_answer>
   </variable>
```

Figure 2: Example of a Survey Variable

> Results of more recent public opinion such as the Allbus Survey conducted in 2006 show that the majority of Germans continue to reject the idea of generally granting dual citizenship. (1)

The text fragment in (2) can be linked to the Allbus variables v274, v275, v276 and v277. All of them have a common main question *Do you have any personal contact with foreigners living in Germany?* but differ in their respective subquestions, *e.g.*, *...at work?*, *...in your neighborhood?* etc.

> Encounters between Germans and foreigners can take place in different spheres of life, at work, in the neighborhood, in the family or in the circle of friends and acquaintances. (2)

Our dataset consists of German and English mentions and variables. For instance, sentence (3) originates from a German scientific publication (synonymous to (2)) and can be linked to the corresponding German survey variable v275 (*i.e.*, *Haben Sie persönlich Kontakte zu in Deutschland lebenden Ausländern, und zwar an Ihrem Arbeitsplatz?*).

> Begegnungen zwischen Deutschen und Ausländern können in verschiedenen Lebensbereichen stattfinden, bei der Arbeit, in der Nachbarschaft, in der Familie oder im Freundes- und Bekanntenkreis. (3)

The problem of identifying survey variable mentions in texts can be defined as a multi-label classification task (Zielinski and Mutschke, 2017): given a set of sentences S $\subseteq \{s_1, .., s_i\}$ and variables V $\subseteq \{v_1, .., v_j\}$, a function needs to be defined $h : S \rightarrow V$. Each sentence s is represented by a single instance which can be associated with one (or more) class label(s), including *unrelated* as a label, in case the mention cannot be assigned to any of the variables.

3. General Annotation Procedure

Our benchmark dataset contains data from 100 scientific publications compiled from the Social Science Open Access Repository[6] (SSOAR) which all carry an established link to the survey study ALLBUS. It covers 20 general-domain topics such as *economy*, *political attitudes and participation*, *attitudes towards marriage, family and partnership*, and *use of media*.[7]

The Variable Corpus is a development corpus of English and German data given as single structured XML files, one for each language. The corpus consists of 415 *positive* and 505 *negative* sentence pairs hand-tagged by two social science students when in their judgment a mention in the text can be linked to a variable or not. For example, given the variable "Do you believe in Heaven?" and a citation "Two million inhabitants believe in God and Heaven", the annotators picked the citation and the variable as a positive pair. The citations extracted from the scientific documents provide a set of roughly synonymous sentences representing different linguistic realizations of a particular variable in the knowledge base. Positive sentence pairs and negative pairs can be used jointly for *Variable Disambiguation*. Moreover, 865 *unrelated* sentences have been selected and can be used for the subtask of *Variable Detection*.

The annotation procedure consists of the following steps:

1. Annotators read all documents from beginning to end and search for text passages that refer to any of the variables of the underlying survey. If so, they look them up in the variable catalog and retrieve the variable ID, text, question and answer set.

2. Annotators build negative pairs from topically-related variables. The mention-variable pairs are cases where annotators disagreed and/or with a high degree of confusability. These cases might be particularly informative because they are near the decision boundary.

3. Annotators select additional sentences from the abstract and/or the same paragraph in which a mention occurs. In our setting, approximately three quarter of the corpus is made up of *unrelated* sentences.

4. Validation: In a final pass, all given sentence pairs are revisited to resolve any remaining inconsistencies. Any pdf-to-text errors in the text and line-breaks were removed as a part of the preformatting.

In step 2, annotators choose alternative variables such as, *e.g.*, *v1328*, *i.e.*, *"Have you had German citizenship since birth?"* and *v261* which relates to the question *"whether to grant German citizenship only to persons that were born in Germany"*. These are used to build negative variable-mention pairs.

Then, in step 3, sentences from the abstract and the same passage are extracted. For instance, sentence (4) immediately follows (1) and elaborates on the same topic.

> Asked to indicate the degree to which they agreed or disagreed with the idea of allowing immigrants to naturalize without relinquishing their former citizenship, 40 percent of respondents strongly agreed and 54 percent rejected the idea to some degree. (4)

4. Annotation Guidelines

Setting up a clear annotation guideline is important for defining the task properly. The disambiguation of variable mentions with respect to a predefined set of variables is sometimes difficult due to ambiguities, vagueness or identification of only partial matches. For this task, we have defined the following annotation guidelines for human annotators:

- *The sentence containing the variable mention should be self-contained*, *i.e.* it should be a suitable reference also when seen in isolation from the context. Annotators thus need to identify the spans of text that most accurately reflect the contents of the variable.
- *Mentions referring to more than one variable* should be assigned all valid variable IDs.
- *Linking mentions at the correct level of granularity*, *i.e.* if alternatively more general or more specific variables exist, they should not be selected.

Our aim was to exhaustively identify all links in the publications and include them as text samples in our gold standard corpus. However, there are some exceptional cases where samples were too vague or part of a table and have therefore not been included in the corpus. For instance, if the reference involves not a single variable but can only be achieved by selecting a whole set of variables, we opted not to include this sample in the corpus. Also, if a survey question cannot be understood without the previous question, we discard the sample. This might happen because the interviewer generally asks a standardized list of questions in consecutive order.

For a few control variables – these are generally used to describe the population selected for the study – the question text was missing and had to be added manually[8].

A design decision was to restrict the length of text samples to a sentence. However, in the variable corpus, the *local context* of the variable is provided, *i.e.*, the whole paragraph in which the mention occurs, so that the similarity of the context of the mention with the associated variable can be exploited. All mentions from the document define the *global context* of the variable, which make it possible to investigate the semantic coherence between co-occurring mentions in a document.

4.1. Sub-sentential Alignments

In order to assess the difficulty of the task, we also explore the dataset in relation to possible sub-sentential alignments and context dependencies for all positive pairs.

[6] http://www.ssoar.info/ssoar/

[7] Variables are assigned to thematic categories based on the CESSDA topic classification, cf. https://dbk.gesis.org/DBKSearch/Topics.asp

[8] For instance, for the variable assessing the respondent's age, we chose *Please tell me your age.*

4.1.1. Linguistic Annotation Layer

Two computational linguistics students annotated and aligned all positive sentence pairs using the paraphrase typology and tagset of Vila (2015) which has been created for addressing individual paraphrase phenomena. In their work, 24 paraphrase types have been defined, ranging from morpho-lexical changes (*e.g.*, derivational changes, lexical substitutions), to syntax-based (*e.g.*, negation switching, ellipsis), discourse-based (modality changes), and semantic-based changes.

An overview of the frequency of the different types of word and phrase alignments between corresponding sentence pairs in our English and German corpus is provided in Figure 3, including examples in Table 1. The most frequent types are identity mappings, followed by (local) lexical-semantic variations and (global) discourse-based modifications. In the case of identity mappings (*i.e.*, aligned phrases that are exactly the same in wording), token-level overlap is 16,1% and character-level overlap makes up 16,75%, when normalized by the length of the question text[9]. Note that discourse-related modifications such as conversion from direct to indirect speech are frequent in our use case and not relevant for the judgment on meaning preservation. They also imply changes regarding modality, tenses, adverbials, pronouns, and often go along with argument variation.

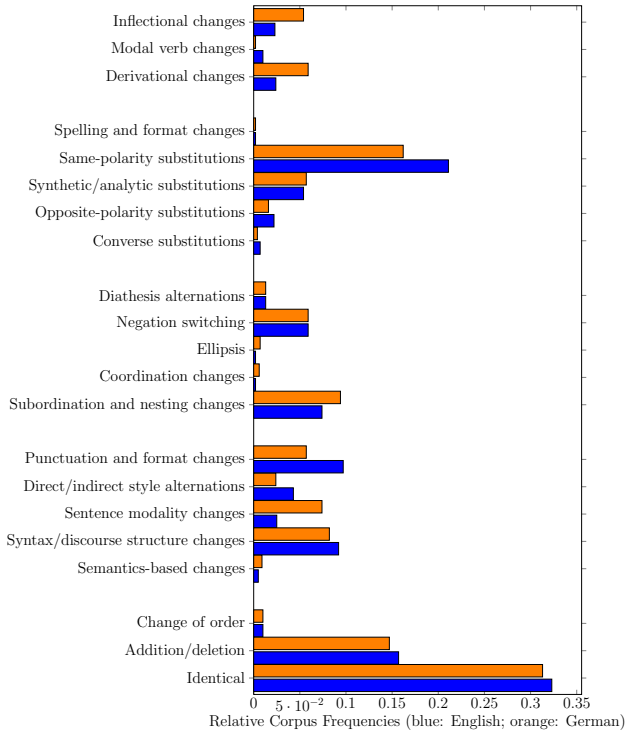


Figure 3: Linguistic Phenomena identified in the annotated corpus (in percentage terms)

4.1.2. Semantic Types of Linkages

Different phenomena regarding the semantic type of linkage could be observed at the sentence level. Special to our setting is that the relation between mention and variable is not necessarily symmetric, *e.g.*, an entailment might hold.

- EQUIVALENT: [fare dodging] ⇔ [use public transport without buying a valid ticket]
- MORE-SPECIFIC: [income tax return] ⇔ [tax return]
- MORE-GENERAL: [lead to problems] ⇔ [reason for shortages]

Moreover, the alignment is mono-lingual and non-exhaustive, *i.e.* it is not required that the entire reference sentence is semantically equivalent to the variable text.

[9]We compute the percentage of lexical overlap between two sentences T and H as: $|T \cap H|/|H|$

Phenomena	Example
Same polarity substitution	make lower - reduce
Opposite polarity substitution	fear - prefer not to
Converse substitution	to work hard - for individual achievement
Derivational change	immigrants - immigration
Inflectional change	walk - walking
Modal verb change	must - has to
Syntax/discourse structure	priorization - choose most important
Diathesis alternation	should have what they need - should get the money they need
Subordination and nesting	belong to a Christian denomination - being Christian
Spelling and format	quotations (”)
Direct/indirect style	What is your opinion? Should social benefits be cut in the future, or should they be extended? - The first question asks whether “social benefits” should be cut or extended [..].
Sentence modality change	How often do you pray? - The average number of children for respondents who never pray was only 1.39 compared to 2.06 for those who pray daily.

Table 1: Linguistic Phenomena in the English Dataset

4.2. Context Clues

Generally, classifying short text is a challenging task because only little context is available and word co-occurrence information cannot be reliably exploited. In our dataset, however, cue words often co-occur with a variable mention in the same sentence or text passage, as shown in example (5) and (6), respectively. Such trigger terms might introduce the speaker (*e.g.*, respondents of a survey), reporting verbs (*e.g.*, verbs expressing opinion or factuality),

measure and assessment verbs, or figures and percentages.

In this case the *figures from Eurobarometer show* that those with worst expectations *report* on average a *7.33%* lower life satisfaction [..]. (5)

In 2009 the *respondents* from *33 European countries were asked by Eurobarometer* whether they expected [..]. (6)

4.3. Corpus Statistics

Table 2 reports some key statistics about our collected datasets. Our benchmark corpus comprises 504 English and 638 German sentences from 35 English and 34 German documents (out of 50 English and 50 German documents) which contained variable mentions. The average length of the sentences extracted from scientific publications is 28 tokens for English and 24 for German, while the average length of variables is 24 tokens for English and 19 tokens for German, considering the question and subquestion text. Pairing the mentions with respective variables results in 466 English and 454 German sentence pairs.

Corpus for Variable Mention Detection

Sentences	#Related	#Unrelated
English	126	378
German	151	487

Corpus for Variable Mention Disambiguation

Sentence Pairs	#Positive	#Negative
English	194	272
German	221	233

Table 2: Corpus Statistics

We also computed the cardinality of the dataset S (*i.e.*, the mean of the number of labels of the instances that belong to S) and the density of S (*i.e.*, the mean of the number of labels of the instances that belong to S divided by $\mathbf{card}(L)$). Label density is between 1 and 7, with a mean of 1.54 and 1.46 as shown in Table 3.

Sentences	#English	#German
Label Density	1.54	1.46
Cardinality	3.69	3.00

Table 3: Label Density and Cardinality

4.4. Inter Annotator Agreement

Inter-Annotator Agreement (IAA) measured with Cohen's Kappa is relatively high, *i.e.*, 80% on sentence level[10] and 91.5% after a reconciliation among the annotators.
On the sub-sentential level, we focused on the agreement w.r.t. the paraphrase type rather than the phrase boundaries, yielding an agreement of approx. 85%.

[10]Average Kappa level of 0.78 corresponding to 'substantial agreement' (Landis and Koch, 1997).

5. Experiments

This section presents results of the baseline approach on our German and English corpus. We evaluate the two steps *Variable Detection* and *Variable Disambiguation* separately: In the first step, we seek to detect occurrences of variable mentions. In the second step, we consider a given set of variables as candidates for all relevant sentences and seek to assign the proper variable ID. Automatically produced annotations are then compared to ground-truth data. We experimented with prominent NLP and ML algorithms adopted to the related tasks RTE, STS and Entity Linking, and tested their effectiveness on out task. We used DKPro Core (de Castilho and Gurevych, 2014), a linguistic pipeline based on UIMA to pre-process the corpus. To facilitate further research on the new resource, we provide a baseline variable linking system based on DKPro-TC (Daxenberger et al., 2014) and DKPro-Similarity (Bär et al., 2012) with models trained on several standard text similarity datasets, *e.g.*, the Microsoft Research Paraphrase Corpus (MSR-Paraphrase)(Dolan et al., 2004)[11].

5.1. Baseline for Variable Detection: Classification using VSM

As a baseline to the Variable Detection task, we adopt a shallow approach based on machine learning applied to lexical features extracted from the dataset. Accordingly, linguistic expressions are treated as a bag of words, using the variable questions and subquestions for training and the mentions for testing. In order to overcome the problem of a lack of training data – in particular because there is only one example for each class – the training dataset is augmented with additional features from WordNet and GermaNet (*i.e.*, synonyms, hypernyms and derivational forms) and keyword terms from *TheSoz* (Zapilko et al., 2013)[12].
For the two-way classification task, *i.e.*, *related* class (any variable ID) versus *unrelated* class, we focus on a high recall which makes it possible to filter out false positives in a later processing step. Table 4 shows the performance in terms of (macro-averaging) precision and recall. As we hypothesized, including only the lemmas in the feature vector yields relatively low recall on the minority class. Best results in this regard are achieved when training the model by expanding the feature space with semantic relations from the lexical databases, and using it to classify the test data, which is based on lemmas. Moreover, results are constistently better for English than for German, mainly due to the high rate of German compounds that have not been splitted into their component parts.

5.2. Baseline for Variable Disambiguation: Similarity Metrics

We have also conducted experiments based on text similarity scores, including *e.g.*, *greedy-string tiling*, *levenshtein*, *longest common subsequence*, *character n-gram* and *BLEU*. In this configuration, similarity scores for pairs

[11]https://github.com/openminted/uc-tdm-socialsciences

[12] Thesaurus for the Social Sciences http://lod.gesis.org/thesoz/

English Baseline for Variable Detection

	P	R	P	R	MAP	MAR
	related		*unrelated*			
Lemma	0.81	0.18	0.59	0.97	0.70	0.57
+Lexical Resources	0.63	0.47	0.65	0.78	0.64	0.63
Train/Test on F_{Lex}/F_{Lem}	0.48	**0.91**	0.76	0.21	0.62	0.56

German Baseline for Variable Detection

	P	R	P	R	MAP	MAR
	related		*unrelated*			
Lemma	0.37	0.16	0.66	0.85	0.51	0.51
+Lexical Resources	0.37	0.16	0.66	0.85	0.51	0.51
Train/Test on F_{Lex}/F_{Lem}	0.36	**0.82**	0.71	0.23	0.54	0.53

Table 4: Baseline Results for Variable Detection based on the Naïve Bayes Classifier with 3 different configuations. *Lemma*: The classifier is trained and tested on lemmas; *+Lexical Resources*: The classifier is trained and tested on lemmas enriched with features from lexical resources; and F_{Lex}/F_{Lem}: When training the classifier, features from lexical resources are integrated, while testing is carried out on the lemma forms.

of variable description and their mention are calculated, combing the set of extracted features into one feature vector and feeding it into a Simple Logistic Regression classifier. For comparison, we also ran the Sequential Minimal Optimization (SMO) classifier which performed slightly better on our datasets. Evaluation results for the task of *Variable Disambiguation* based on 10-fold cross-validation are reported in terms of accuracy, *i.e.*, only for instances that belong to the *related class*. The accuracy in our multi-label classification setting is the proportion of labels correctly classified of the total number (predicted and actual) of labels for that instance averaged over all instances.

Algorithm	English	German
Logistic Regression Classifier	60.39%	65.25%
SMO	63.68%	66.81%

Table 5: Baseline Evaluation Results for Variable Disambiguation in terms of accuracy, based on the precision/recall for each class label over the *related class* dataset

6. Related Work

While various benchmark datasets have been developed for the shared tasks in semantic relatedness and textual entailment (Bentivogli et al., 2017) (Agirre et al., 2013), no resource exists so far for the Variable Detection and Linking task. There are major differences which makes the task interesting, summed up in Table 6, w.r.t. the following characteristics:

a) Context Dependency: Should information outside the sentence pairs be taken into account?

b) Class Distribution: Is the corpus balanced or unbalanced in terms of related and unrelated sentences? Is it balanced in terms of positive and negative sentence pairs?

c) Partial Entailment: Only some 'facets' within the sentences match

d) Domain: Formal (*e.g.*, scientific publications, news) versus informal domains (*e.g.*, forums, blogs)

Our corpus differs from other corpora in related applications in various ways: a) the local context (*i.e.*, paragraph) in which the mention occurs is provided so that context similarity clues can be taken into account; b) while the majority of sentences belong to the *unrelated* class, class distribution according to the *positive* and *negative* class is almost balanced; c) semantic equivalence or entailment relationships can generally be observed only in parts of the sentences; and d) the corpus has been compiled from German and English scientific publications.

Because of the fact that the major bottleneck of our use case is the high variability due to different linguistic realizations of the same variable, the RTE semantic search scenario seems most appropriate to our use case. Due to the lack of training data, no prior knowledge on the likelihood of a link can be exploited, as is usually done in entity linking.

7. Conclusion

We have introduced a new dataset which has been created for the *Variable Detection and Linking* task and originates from the needs within the social sciences. We intend to make the corpus freely available to the research community under a Creative Commons license, along with the annotation guidelines. We have proposed a pipeline that includes several stages: a) pre-processing, b) Variable Detection and c) Variable Disambiguation, and evaluated it on our German and English datasets. We first applied a Naïve Bayes Classifier on BoW lexical features extended with *WordNet/GermaNet* and *TheSoz* terms to achieve a high recall and then ran a more precision-oriented SMO classifier based on string similarity features. While this approach is flexibel and can easily adapt to any new repertoire of survey variables, experimental results show that due to the small number of available training instances this is a challenging task within NLP. Yet, we think that the dataset will foster research in this field and lead to enhanced solutions that might also take into account the local and global context of the variable mention, and exploit the answer set of the variables.

8. Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation programme (H2020-EINFRA-2014-2) under grant agreement No. 654021 (OpenMinTeD)[13].

[13] http://openminted.eu/

Datasets	RTE1-4	RTE5-6	RTE8	WikiQA	STS-par	SEM-QA	iSTS'16	VDL'18
Context Given	No	Yes	No	No	No	No	No	Yes
Balanced Class Distribution	Yes	No	No	No	Yes	Yes	Yes	No
Partial Entailment	No	No	Yes	No	No	No	Yes	Yes
Domain	News	News	Scholar	Wikipedia	News	Forum	News	Scholar

Table 6: Dataset Characteristics *(Bentivogli, 2017)*. VDL'18 is our Variable Detection and Linking Corpus.

9. Language Resource References

Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., and Guo, W. (2013). SEM 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In *In* SEM 2013: The Second Joint Conference on Lexical and Computational Semantics*.

Allerbeck, K., Allmendinger, J., Andreß, H.-J., Bürklin, W., Diekmann, A., Feger, H., Fetchenhauer, D., Huinink, J., Kiefer, M. L., Kreuter, F., Kühnel, S., Kurz, K., Lepsius, M. R., Liebig, S., Mayer, K. U., Meulemann, H., Müller, W., Opp, K. D., Pappi, F. U., Scheuch, E. K., Schmitt-Beck, R., Solga, H., Trappe, H., Wagner, M., Westle, B., and Ziegler, R. (2016). Allgemeine Bevölkerungsumfrage der Sozialwissenschaften ALLBUS - Kumulation 1980-2014.

Bär, D., Biemann, C., Gurevych, I., and Zesch, T. (2012). UKP: Computing Semantic Textual Similarity by Combining Multiple Content Similarity Measures. In *SEM 2012: The First Joint Conference on Lexical and Computational Semantics (SemEval 2012)*, pages 435–440, Montréal, Canada, 7-8 June.

Bentivogli, L., Clark, P., Dagan, I., and Giampiccolo, D. (2009). The Fifth PASCAL Recognizing Textual Entailment Challenge. In *TAC*.

Bentivogli, L., Dagan, I., and Magnini, B. (2017). The recognizing textual entailment challenges: Datasets and methodologies. In *Handbook of Linguistic Annotation*, pages 1119–1147. Springer.

Boland, K., Ritze, D., Eckert, K., and Mathiak, B. (2012). Identifying references to datasets in publications. In *International Conference on Theory and Practice of Digital Libraries*, pages 150–161. Springer.

Branco, A. (2012). Reliability and Meta-reliability of Language Resources: Ready to initiate the integrity debate? In *The 12th workshop on treebanks and linguistic theories*.

Cohen, K. B., Xia, J., Roeder, C., and Hunter, L. (2016). Reproducibility in natural language processing: A case study of two R libraries for mining PubMed/MEDLINE. In *LREC 4REAL Workshop: Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*, pages 6–12.

Dagan, I., Roth, D., Sammons, M., and Zanzotto, F. M. (2013). Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies*, 6(4):1–220.

Daxenberger, J., Ferschke, O., Gurevych, I., and Zesch, T. (2014). DKPro TC: A java-based framework for supervised learning experiments on textual data. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 61–66, Baltimore, Maryland, June.

de Castilho, R. E. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT at COLING*, pages 1–11.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04.

Erbs, N., Zesch, T., and Gurevych, I. (2011). Link discovery: A comprehensive analysis. In *Fifth IEEE International Conference on Semantic Computing (ICSC)*, pages 83–86. IEEE.

Fokkens, A., Van Erp, M., Postma, M., Pedersen, T., Vossen, P., and Freire, N. (2013). Offspring from Reproduction Problems: What replication failure teaches us. In *ACL (1)*, pages 1691–1701.

Harabagiu, S. and Hickl, A. (2006). Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics*, pages 905–912.

Lepsius, M. R., Kreuter, F., Liebig, S., Kurz, K., Kühnel, S., Kiefer, M. L., Huinink, J., Fetchenhauer, D., Solga, H., Scheuch, E. K., Schmitt-Beck, R., Pappi, F. U., Müller, W., Opp, K. D., Meulemann, H., Trappe, H., Mayer, K. U., Feger, H., Diekmann, A., Bürklin, W., Andreß, H.-J., Allmendinger, J., Allerbeck, K., Wagner, M., Westle, B., and Ziegler, R. (2016). German General Social Survey (ALLBUS) - Cumulation 1980-2014.

Mariani, J., Paroubek, P., Francopoulo, G., and Hamon, O. (2016). Rediscovering 15+ 2 years of discoveries in language resources and evaluation. *Language Resources and Evaluation*, 50(2):165–220.

Rao, D., McNamee, P., and Dredze, M. (2013). Entity linking: Finding extracted entities in a knowledge base. In *Multi-source, multilingual information extraction and summarization*, pages 93–115. Springer.

Vila, M., Bertran, M., Martí, M. A., and Rodríguez, H. (2015). Corpus annotation with paraphrase types: New annotation scheme and inter-annotator agreement measures. *Language Resources and Evaluation*, 49:77–105.

Zapilko, B., Schaible, J., Mayr, P., and Mathiak, B. (2013). Thesoz: A SKOS representation of the thesaurus for the social sciences. *Semantic Web*, 4(3):257–263.

Zielinski, A. and Mutschke, P. (2017). Mining social science publications for survey variables. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 47–52.

KRAUTS: A German Temporally Annotated News Corpus

Jannik Strötgen[α], Anne-Lyse Minard[β δ], Lukas Lange[α], Manuela Speranza[β], Bernardo Magnini[β]
[α] Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[β] Fondazione Bruno Kessler, Trento, Italy
[δ] Univ Rennes, Inria, CNRS, IRISA, Rennes, France
{jannik.stroetgen,llange}@mpi-inf.mpg.de, {minard,manspera,magnini}@fbk.eu

Abstract

In recent years, temporal tagging, i.e., the extraction and normalization of temporal expressions, has become a vibrant research area. Several tools have been made available, and new strategies have been developed. Due to domain-specific challenges, evaluations of new methods should be performed on diverse text types. Despite significant efforts towards multilinguality in the context of temporal tagging, for all languages except English, annotated corpora exist only for a single domain. In the case of German, for example, only a narrative-style corpus has been manually annotated so far, thus no evaluations of German temporal tagging performance on news articles can be made. In this paper, we present KRAUTS, a new German temporally annotated corpus containing two subsets of news documents: articles from the daily newspaper DOLOMITEN and from the weekly newspaper DIE ZEIT. Overall, the corpus contains 192 documents with 1,140 annotated temporal expressions, and has been made publicly available to further boost research in temporal tagging.

Keywords: temporal tagging, corpus annotation, TIMEX3

1. Introduction

Temporal tagging – the extraction and normalization of temporal expressions from texts – is an important task towards improved natural language understanding. Once temporal expressions have been detected in a text, their semantics can be assigned to them in a standard format so that applications can exploit not only the surface forms of temporal expressions, but also their meaning. For instance, applications in event / timeline extraction (Minard et al., 2015; Cornegruta and Vlachos, 2016; Spitz and Gertz, 2016), question answering (Llorens et al., 2015) and (temporal) information retrieval (Kanhabua et al., 2015) can exploit temporal tagging output. Thus, temporal tagging has become a vibrant research area, and several new temporal taggers have been made available and new strategies have been developed. However, as was shown in previous work (Mazur and Dale, 2010; Strötgen and Gertz, 2013; Bethard et al., 2016; Tabassum et al., 2016), different types of documents pose different challenges for temporal tagging such that domain-sensitive normalization strategies are required (Strötgen and Gertz, 2016). To judge the performance of temporal taggers and new methods, evaluations need to be performed on diverse text types, e.g., on news articles and narrative-style Wikipedia documents.

In contrast to many natural language processing tasks, there has also been some effort towards multilinguality in the context of temporal tagging, e.g., research competitions were organized not only for English but covered further languages such as Spanish and Italian (Verhagen et al., 2010; Caselli et al., 2014). Despite its importance, German has not been part of any of these challenges so far. In addition, HeidelTime is the only publicly available temporal tagger for German, and only narrative style corpora have been manually annotated so far. Thus, no proper evaluations of German temporal tagging performance on news articles can be carried out. Therefore, HeidelTime's German temporal tagging quality has only been evaluated on narrative texts using the WikiWarsDE and AncientTimes corpora.

In this paper, we present our effort in developing KRAUTS, a new temporally annotated corpus in German containing two subsets of news documents: articles from the daily newspaper DOLOMITEN and from the weekly newspaper DIE ZEIT. For annotating temporal expressions in the corpus, we developed annotation guidelines for German temporal tagging by using the guidelines defined for Italian (Caselli and Sprugnoli, 2015) as a starting point. Overall, the corpus contains 192 documents with 1,140 annotated temporal expressions, and the corpus as well as the annotation guidelines have been made publicly available to further boost research in temporal tagging.[1]

2. Related Work

The task of temporal processing has gained interest in recent years, in particular thanks to the TempEval tasks at SemEval (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013; Llorens et al., 2015; Bethard et al., 2016). Temporal tagging is a subtask of temporal processing and consists of the identification of temporal expressions in texts and their normalization to some standard format. Strötgen and Gertz (2016) present a complete overview of the task as well as a survey of the resources, tools, etc. They focus on the description of domain-sensitive temporal tagging and multilingual taggers.

The annotation of temporal expressions follows in most cases the TimeML annotation guidelines (Pustejovsky et al., 2003) developed first for English. They have then been adapted to other languages such as Italian (Caselli and Sprugnoli, 2015), Spanish (Saurí et al., 2009) and French (Bittar, 2010). However until now no adaptation of the guidelines to German has been done. The two corpora of narratives in German, AncientTimes (Strötgen et al., 2014) and WikiWarsDE (Strötgen and Gertz, 2011), have been manually annotated but following the English TimeML guidelines without further specifying language-

[1] http://github.com/JannikStroetgen/KRAUTS

specific adaptations. WikiWarsDE is the German counterpart of the English WikiWars corpus (Mazur and Dale, 2010) and AncientTimes is a small multilingual corpus containing documents about history.

Driven by the above-mentioned shared tasks, many temporal taggers have been developed. Some of these can process several languages, such as TIPSem (Llorens et al., 2010) for English and Spanish, TimePro (Mirza and Minard, 2014) (a module of TextPro[2]) for English, Italian and French, and HeidelTime (Strötgen and Gertz, 2013) for 13 languages, including German, as well as its automatic extension as a baseline temporal tagger for more than 200 languages (Strötgen and Gertz, 2015). Strötgen et al. (2014) performed an evaluation of HeidelTime on two German corpora of narratives: WikiWarsDE and AncientTimes. They reported F1-scores of 87.7 and 78.0 for strict match, and value F1-scores[3] of 80.4 and 82.2 on WikiWarsDE and AncientTimes, respectively.

Our work consists of defining TimeML guidelines for German and annotating a corpus following these guidelines. Using the newly annotated corpus, we report evaluation results for HeidelTime, which to the best of our knowledge is the only temporal tagger available for German.

3. Corpus Description

KRAUTS (Korpus of newspapeR Articles with Underlined Temporal expressionS) consists of two subsets: articles of the daily, regional newspaper DOLOMITEN and articles of the nationwide weekly newspaper DIE ZEIT. The corpus is composed of 192 documents with a total of 75,678 tokens. Table 1 contains some statistics about both subsets. Details about annotated temporal expressions will be given in Section 5.

3.1. Dolomiten

The DOLOMITEN subset consists of 142 articles published between 2009 and 2016. DOLOMITEN is a local newspaper from South Tyrol (Italy) written in the local variant of German. Therefore, the articles contain words and phrases which are not used in High German, including the temporal expression *heuer* which translates to "this year".

Two students, supervised by two expert annotators, performed the manual annotation of temporal expressions. 100 DOLOMITEN articles were annotated starting from raw text, while the remaining 42 were first pre-annotated with the HeidelTime tool and then checked and corrected manually in order to speed-up the annotation process.

3.2. Die Zeit

In the context of a Bachelors thesis on a time-centric analysis of German news articles (Lange, 2017), 50 documents of the German weekly newspaper DIE ZEIT were manually annotated by two annotators – but without first adapting the English annotation guidelines to German in a concise way. This resulted in several discussions about non-uniform annotations, and it was concluded that proper annotation guidelines for German are required to achieve high quality manual annotations. In the context of the collaboration between the Fondazione Bruno Kessler and the Max Planck Institute for Informatics, the 50 articles have been re-annotated following the newly developed annotation guidelines for German (cf. Section 4).

Compared to the documents in the DOLOMITEN part of the KRAUTS corpus, the DIE ZEIT articles are very long (an average of 885 tokens vs. 221 per article, respectively). In addition, as they are sometimes rather non-standard news articles, annotating temporal expressions in these documents is probably more challenging, even for humans.

	DOLOMITEN	DIE ZEIT	KRAUTS
# documents	142	50	192
# tokens	31,422	44,256	75,678
tokens/doc	221	885	394

Table 1: Statistics about the KRAUTS corpus.

4. German-specific Guidelines

As German presents some language-specific phenomena which have to be taken into account when performing any annotation task, it is not possible to apply the English temporal annotation guidelines to German in a straightforward way. The adaptations to be done mainly affect the extent of temporal expressions. In particular, in German there are compounds which can sometimes contain temporal expressions (e.g., *Diskussionabende* "Evenings of discussion") and there are contractions of prepositions and articles (e.g., *im*: *in* + *dem* "in the"). It is widely accepted that annotations of temporal expressions should always start and end at a token boundary, whereas the specific morphology of German would lead to annotating a subpart of the tokens in the case of compounds and contractions. This illustrates that the English annotation guidelines cannot be directly applied, as these state that articles, when present, are part of temporal expressions while prepositions are not.

In order to develop the guidelines needed for the annotation of temporal expressions in German, we selected the It-TimeML guidelines (Caselli and Sprugnoli, 2015) as a reference. The choice was motivated not only by the fact that these guidelines are very well-defined and detailed, but also by the fact that in Italian, as in German, it is also possible to contract articles and prepositions. Thus, the Italian guidelines are a more natural choice than the English ones when adapting annotation guidelines to German.

The new guidelines we produced are summarized in the document *Examples and Guidelines for Annotation of Temporal Expressions (<TIMEX3>) in German*, which is an annex to the It-TimeML guidelines. It is available for download on the It-TimeML website[4] and linked from the KRAUTS website. The annex contains the extensions needed to adapt the It-TimeML guidelines to the specific morpho-syntactic features of German, as well as many German annotated examples to illustrate the application to the German language of the relevant It-TimeML guidelines.

[2] http://textpro.fbk.eu

[3] "value F1-score" consists of evaluating both the recognition of the temporal expressions with relaxed match and the correctness of the normalization value.

[4] https://sites.google.com/site/ittimeml/documents

The numbering of the examples in the annex is the same as in the It-TimeML guidelines.

4.1. Compounds

Compound words, lexemes that consists of more than one lexical element, are very frequent in German. For example, *Werktag* is composed of *Werk* "work" and *Tag* "day" and means "working day". We can see in this example that compounds can contain lexical elements with temporal meaning.

According to our guidelines, a compound containing a temporal trigger has to be annotated if the syntactic head of the compound is a temporal trigger. If the syntactic head is not a temporal trigger, on the other hand, the compound should not be annotated, even if it contains a temporal trigger. For example, *Diskussionsabende* in (1) is annotated because the syntactic head *abend* "evening" is a temporal trigger, whereas *Monatsblatt* in (2) is not annotated (the syntactic head is *blatt* "leaflet", which is not a temporal trigger).

(1) Weiters werden im Jugendtreff <TIMEX3>zwei Diskussions**abende**</TIMEX3> veranstaltet.
[Furthermore, two evenings of discussion will be organized in the youth center.]

(2) Jedes **Monats**blatt behandelt ein eigenes Thema.
[Each monthly leaflet addresses a specific subject.]

4.2. Prepositions, Articles, and Contractions

Following the general TimeML rule, articles (whatever their case) are included in the extent of the temporal expressions (3), while prepositions are excluded (4); as for contractions of prepositions and definite articles, we adopted the Italian guidelines and so they are not included in the extent of the temporal expression (5). This decision leaves open the possibility of marking contractions in a future step, as they often include prepositions used as indicators of temporal relations. According to the TimeML framework, these are to be marked as SIGNAL if the full task of temporal annotation and not just temporal tagging is the goal.

(3) <TIMEX3>**der** nächste Tag</TIMEX3> [the next day]

(4) **um** <TIMEX3>15:00 Uhr</TIMEX3> [at 3 pm]

(5) **im** <TIMEX3>nächsten Jahr</TIMEX3> [in_the next year]

5. Corpus Annotation

In the KRAUTS corpus, we have annotated the following types of temporal expressions: DATE (calendar unit), TIME (time of the day), DURATION (period of time), and SET (reoccurring temporal expression), and the following attributes: tid (identifier), type (type of the temporal expression), value (normalized value), anchorTimeID (ID of the temporal expression to which the marked expression is linked), beginPoint (begin point of a DURATION), endPoint (end point of a DURATION), freq (frequency of a SET), quant (quantifier of a SET) and mod (temporal modifier).

According to the TimeML annotation guidelines, TIMEX3 tags with no extent (i.e., empty TIMEX3 tags) are introduced, for example, to deal with unspecified time points, which are sometimes needed to anchor durations. Nevertheless, in most prior work, empty TIMEX3 tags have not been used, neither in annotated corpora nor by TIMEX3-compliant temporal taggers.

However, in order to represent durations in a better way, empty TIMEX3 tags should be annotated. We thus followed the organizers of the Italian temporal tagging challenge EVENTI (Caselli et al., 2014) and the developers of the MEANTIME corpus (Minard et al., 2016), who were the first, and, to the best of our knowledge, the only researchers so far who have annotated empty TIMEX3 tags in documents which have resulted in publicly available corpora (the EVENTI corpus for Italian and the MEANTIME corpus contains English, Dutch, Italian, and Spanish temporally annotated news articles). An example of an empty TIMEX3 tag is given in (6): The duration *vor einem Monat* "one month ago" (*vor* is outside of the TIMEX3 tag as it is a preposition) is annotated in the text and an empty TIMEX3 tag of type DATE is added which represent the date of one month ago to anchor the duration.

(6) *(DCT: 2018-01-17, t0)* ... vor <TIMEX3 tid="t1" type="DURATION" value="P1M" beginPoint="t2" endPoint="t0">einem Monat</TIMEX3> ...
<TIMEX3 tid="t2" type="DATE" value="2017-12-17" anchorTimeID="t0"/>
[... one month ago ...]

	DOLOMITEN	DIE ZEIT	KRAUTS
DATE	376	358	734
TIME	98	12	110
DURATION	94	144	238
SET	19	39	58
Total	587	553	1,140
Empty tag	30	41	71

Table 2: Annotation statistics (in the first part of the table we give the number of text-consuming TIMEX3).

In Table 2, we provide information about the distribution of the different types of temporal expressions in the corpus. In total, KRAUTS contains 1,140 (text-consuming) temporal expressions and 71 empty TIMEX3 tags. 64% of the temporal expressions are of type DATE.

The DOLOMITEN subset contains 587 temporal expressions, a large proportion of which are dates. We observe a rather high number of temporal expressions of type TIME with regards to the type of texts (newspaper articles). We can explain it by the presence of local event announcements in the DOLOMITEN newspaper.

The DIE ZEIT subset contains 553 temporal expressions, with a majority of dates. Compared to the DOLOMITEN subset, it contains few time expressions and many duration and set expressions. The rather low number of time expressions is probably due to the fact that DIE ZEIT is a weekly

newspaper so that such fine-granular expressions are less important. Duration and set expressions often occur in articles belonging to categories (such as "travel") that are not typical news categories. This also shows that the DIE ZEIT subset contains very diverse articles and several some of the documents can be considered as rather less typical news articles compared to articles of a daily newspaper – which increases the difficulty of temporal tagging the DIE ZEIT subset of the KRAUTS corpus.

6. Evaluating HeidelTime on KRAUTS

In Table 3, we present the evaluation of HeidelTime, performed with the TempEval-3 scorer[5], on KRAUTS.[6] We have performed the evaluation on three sections of the corpus separately: DOLOMITEN-42 (the subpart of the Dolomiten articles pre-annotated with HeidelTime and revised manually), DOLOMITEN-100 (the Dolomiten articles annotated manually starting from raw texts), and DIE ZEIT. For comparison, the best system for relaxed matching at TempEval-3 on English news documents, SUTime, obtained an F1-score of 90.32 on relaxed match, 79.57 on strict match and 67.38 on value, and the overall best system, HeidelTime, obtained an F1-score of 90.30 on relaxed match, 81.34 on strict match, and 77.61 on value.

The results obtained on DOLOMITEN-42 are higher than those obtained on the other two sections; this can be explained considering that the DOLOMITEN-42 articles had been pre-annotated with HeidelTime and consequently the final annotation (after manual revision) might still be slightly biased.

The results on the DIE ZEIT articles are the lowest, probably because the articles are very long and thus have a more complex temporal discourse structure. In addition, some articles are written in a narrative rather than news-style fashion – due to the characteristics of the weekly newspaper in general – which led to incorrect normalizations.

In Table 4, we give the detailed results obtained for each type of temporal expression. We can observe that the best results are obtained for temporal expressions of type DATE and DURATION. The results for type SET are low, but it should be noticed that the corpus contains very few of them so that few false positives and false negatives lower the score significantly.

We have analyzed the annotations made by HeidelTime on the corpus DOLOMITEN-42 (corpus of 42 files pre-annotated with HeidelTime). We counted three false positives: the age of a person (which is not to be marked as a temporal expression), a four digit number that was not a year, and an occurrence of "Christmas" which did not refer to Christmas as a time period but, more in general, as a subject. As false negatives we found many expressions where the time of day (e.g., "um 20.30 Uhr" [at 20:30]) was not an hour on the dot, as well as expressions with the ordinal number of the week (e.g., "der dritten Woche" [the third week]). Introducing a few new rules will prevent the tool from leaving out these temporal expressions in the future.

	strict	relaxed	type	value
DOLOMITEN-42	79.4	83.1	82.5	80.0
DOLOMITEN-100	71.4	81.3	76.9	66.2
DIE ZEIT	69.7	79.3	75.4	62.8

Table 3: HeidelTime evaluation results (in terms of F1-score) on the three subsets of KRAUTS.

	DATE	TIME	DURAT.	SET
DOLOMITEN-42	84.8	77.1	87.5	50.0
DOLOMITEN-100	83.3	51.7	66.1	54.6
DIE ZEIT	78.8	76.2	74.5	37.5

Table 4: HeidelTime evaluation results (relaxed match in terms of F-measure) on the three subsets of KRAUTS.

7. Conclusions

In this paper, we defined specific TimeML guidelines for German starting from It-TimeML, the Italian TimeML guidelines. Following these new guidelines we annotated a corpus of newspaper articles: the KRAUTS corpus. It is the first news corpus for temporal tagging in German. It consists of 192 articles from a daily, regional newspaper and a weekly newspaper, with 1,140 annotated temporal expressions.

As a benchmark for the evaluation of automatic systems, we have exploited KRAUTS to evaluate HeidelTime, the temporal tagger for German, which has only been evaluated against narrative-style corpora for German so far. On two of the three subparts of the corpus (DOLOMITEN-100 and DIE ZEIT) it obtained F1-scores of around 70 and 80, respectively for strict and relaxed match.

KRAUTS contains different kinds of news articles which differ from the length of the documents and the proportion of each type of TIMEX3. This is not enough in order to develop and evaluate a generic temporal tagger. For German, narrative-style temporal annotated documents are also available. It will now be interesting to also annotate some colloquial texts, such as tweets or emails.

8. Acknowledgments

We thank Sara Baino and Martina Coser for their contribution, which consisted of manually annotating the DOLOMITEN articles. This work has been partially funded by the EUCLIP_RES project under the FESR Program of the Autonomous Province of Bolzano - South Tyrol.

9. Bibliographical References

Bethard, S., Savova, G., Chen, W.-T., Derczynski, L., Pustejovsky, J., and Verhagen, M. (2016). SemEval-2016 Task 12: Clinical TempEval. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval'16, pages 1052–1062. Association for Computational Linguistics.

[5] www.cs.york.ac.uk/semeval-2013/task1/index.html

[6] "Strict" and "relaxed" refer to the evaluation of the extent of the temporal expressions; "type" and "value" represent the evaluation of the respective attributes. The F1-score is computed taking into account the recognition of the temporal expressions with relaxed match and the identification of the attribute. The TempEval-3 scorer does not evaluate the empty TIMEX3 tags, so they are not part of the presented evaluation.

Bittar, A. (2010). ISO-TimeML Annotation Guidelines for French, version 1.0. Technical report.

Caselli, T. and Sprugnoli, R. (2015). It-TimeML, TimeML Annotation Guidelines for Italian, version 1.4. Technical report.

Caselli, T., Sprugnoli, R., Speranza, M., and Monachini, M. (2014). EVENTI: EValuation of Events and Temporal INformation at Evalita 2014. In *Proceedings of the Fourth International Workshop EVALITA*, EVALITA 2014, pages 27–34.

Cornegruta, S. and Vlachos, A. (2016). Timeline Extraction Using Distant Supervision and Joint Inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16, pages 1936–1942. Association for Computational Linguistics.

Kanhabua, N., Blanco, R., and Nørvåg, K. (2015). Temporal Information Retrieval. *Found. Trends Inf. Retr.*, 9(2):91–208.

Lange, L. (2017). Time in Newspaper – A Large-scale Analysis of Temporal Expressions in News Corpora. Bachelor's thesis, Universität des Saarlandes, Max Planck Institute for Informatics, Saarland Informatics Campus.

Llorens, H., Saquete, E., and Navarro, B. (2010). TIPSem (English and Spanish): Evaluating CRFs and Semantic Roles in TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval'10, pages 284–291. Association for Computational Linguistics.

Llorens, H., Chambers, N., UzZaman, N., Mostafazadeh, N., Allen, J., and Pustejovsky, J. (2015). SemEval-2015 Task 5: QA TempEval - Evaluating Temporal Information Understanding with Question Answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval'15, pages 792–800. Association for Computational Linguistics.

Mazur, P. and Dale, R. (2010). WikiWars: A New Corpus for Research on Temporal Expressions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP'10, pages 913–922. Association for Computational Linguistics.

Minard, A.-L., Speranza, M., Agirre, E., Aldabe, I., van Erp, M., Magnini, B., Rigau, G., and Urizar, R. (2015). SemEval-2015 Task 4: TimeLine: Cross-Document Event Ordering. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval'15, pages 778–786. Association for Computational Linguistics.

Minard, A.-L., Speranza, M., Urizar, R., Altuna, B., van Erp, M., Schoen, A., and van Son, C. (2016). MEANTIME, the NewsReader Multilingual Event and Time Corpus. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, LREC'16, pages 4417–4422. ELRA.

Mirza, P. and Minard, A.-L. (2014). FBK-HLT-time: A Complete Italian Temporal Processing System for EVENTI-Evalita 2014. In *Proceedings of the Fourth International Workshop EVALITA*, EVALITA 2014.

Pustejovsky, J., Castaño, J. M., Ingria, R., Saurí, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003). TimeML: Robust Specification of Event and Temporal Expressions in Text. In *New Directions in Question Answering*, pages 28–34.

Saurí, R., Saquete, E., and Pustejovsky, J. (2009). Annotating Time Expressions in Spanish TimeML Annotation Guidelines. Technical report.

Spitz, A. and Gertz, M. (2016). Terms over LOAD: Leveraging Named Entities for Cross-Document Extraction and Summarization of Events. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'16, pages 503–512. ACM.

Strötgen, J. and Gertz, M. (2011). WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions. In *In Proceedings of the Conference of the German Society for Computational Linguistics and Language Technology*, GSCL'11, pages 129–134.

Strötgen, J. and Gertz, M. (2013). Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298.

Strötgen, J. and Gertz, M. (2015). A Baseline Temporal Tagger for all Languages. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, EMNLP'15, pages 541–547. Association for Computational Linguistics.

Strötgen, J. and Gertz, M. (2016). *Domain-sensitive Temporal Tagging*. Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers, San Rafael, CA.

Strötgen, J., Bögel, T., Zell, J., Armiti, A., Canh, T. V., and Gertz, M. (2014). Extending HeidelTime for Temporal Expressions Referring to Historic Dates. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, LREC'14, pages 2390–2397. ELRA.

Tabassum, J., Ritter, A., and Xu, W. (2016). TweeTime : A Minimally Supervised Method for Recognizing and Normalizing Time Expressions in Twitter. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP'16, pages 307–318. Association for Computational Linguistics.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of 7th International Workshop on Semantic Evaluation*, SemEval'13, pages 1–9. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). SemEval-2007 Task 15: TempEval Temporal Relation Identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval'07, pages 75–80. Association for Computational Linguistics.

Verhagen, M., Saurí, R., Caselli, T., and Pustejovsky, J. (2010). SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval'10, pages 57–62. Association for Computational Linguistics.

CogCompNLP: Your Swiss Army Knife for NLP

Daniel Khashabi[1], Mark Sammons[1], Ben Zhou[2] , Tom Redman[2]
Christos Christodoulopoulos[3] , Vivek Srikumar[4] , Nicholas Rizzolo[5] , Lev Ratinov
Guanheng Luo[2] , Quang Do[6] , Chen-Tse Tsai[7] , Subhro Roy[8] , Stephen Mayhew[1]
Zhili Feng[9] , John Wieting[10] , Xiaodong Yu[2] , Yangqiu Song[11] , Shashank Gupta[2]
Shyam Upadhyay[1] , Naveen Arivazhagan[5] , Qiang Ning[2] , Shaoshi Ling[2] , Dan Roth[1,2]

[1]Univ. of Pennsylvania [2]Univ. of Illinois at Urbana-Champaign [3]Amazon Research Cambridge [4]Univ. of Utah [5]Google
[6]Anduin Transactions [7]Bloomberg L.P. [8]MIT CSAIL [9]Univ. of Wisconsin-Madison [10]Carnegie Mellon University [11]HKUST

Abstract

Implementing a Natural Language Processing (NLP) system requires considerable engineering effort: creating data-structures to represent language constructs; reading corpora annotations into these data-structures; applying off-the-shelf NLP tools to augment the text representation; extracting features and training machine learning components; conducting experiments and computing performance statistics; and creating the end-user application that integrates the implemented components. While there are several widely used NLP libraries, each provides only partial coverage of these various tasks. We present our library CogCompNLP which simplifies the process of design and development of NLP applications by providing modules to address different challenges: we provide a corpus-reader module that supports popular corpora in the NLP community, a module for various low-level data-structures and operations (such as search over text), a module for feature extraction, and an extensive suite of annotation modules for a wide range of semantic and syntactic tasks. These annotation modules are all integrated in a single system, Pipeline, which allows users to easily use the annotators with simple direct calls using any JVM-based language, or over a network. The sister project CogCompNLPy enables users to access the annotators with a Python interface. We give a detailed account of our system's structure and usage, and where possible, compare it with other established NLP frameworks. We report on the performance, including time and memory statistics, of each component on a selection of well-established datasets. Our system is publicly available for research use and external contributions, at: `http://github.com/CogComp/cogcomp-nlp`.

1. Motivation

Natural Language Processing (NLP) is one of the fastest-growing fields both in research and industrial work. Tools that provide analysis of text by identifying syntactic and semantic elements (e.g. named entity recognizers, syntactic parsers, semantic role labelers) are widely used - typically many such components in combination - as inputs to a more specialized (and complex) application. However, the process of managing and aggregating these tools and their inputs and outputs is typically labor-intensive and error-prone, requiring significant engineering effort before the target application can be built and evaluated. With the introduction of new, complex tasks such as event detection or cross-document coreference, and the increasing commercial demand for text analysis, it is critical to build software frameworks that give easy access to a wide range of existing NLP annotators and support straightforward extension to others.

We introduce CogCompNLP, an NLP ecosystem published under an academic-use license, which is designed for management, aggregation, and application of NLP analytic components. It comes with a suite of NLP components for syntactic and shallow semantic analysis, but also word and phrase similarity metrics. It provides essential support for text processing applications, including classes for text cleaning and for reading a number of popular NLP corpora. In addition to describing some key functionalities, we illustrate the use of CogCompNLP components in developing a Semantic Role Labeling application: reading data from a corpus; augmenting the resulting data-structures with NLP components using the NLP pipeline; extracting features for input to machine learning algorithms; training classifiers using LBJava –another CogComp project (Rizzolo and Roth, 2010); serializing the system outputs; and adding the new SRL application to the pipeline for other applications to use.

2. Terminology

The CogCompNLP framework builds on the conceptual design and data-structures described in (Clarke et al., 2012; Sammons et al., 2016). Here we give a brief summary of the main structures and keywords used. A View is a data-structure which contains an annotation structure of a text; examples are tokens, lemmas or dependency parse trees. An Annotator is a class which produces a View given a text, and potentially some other Views. The main data-structure used is TextAnnotation, which contains a document (e.g. a phrase, a sentence, a paragraph) and its various Views.

3. Framework Design

A high-level view of the system is depicted in Figure 1. The boxes show modules and edges show the dependencies between them (with the targets being the dependencies). We describe each of these modules in the following sections.

Core Utilities. Contains the fundamental data-structures and operators; hence many of the other modules depend on it. A selection of important basic functionalities supported by this module are:

- SQL-like operations on TextAnnotation for extracting patterns
- Experiment utilities, such as P/R/F1 reporting, statistical significance testing and cross-validation helpers
- String pattern-matching algorithms
- Utilities for reading and writing files, resources and annotations.

		Sentence splitting	Tokenizing	Lemmatizing	Part of Speech tagging	Shallow Parsing	NER (4 labels)	Extended NER (15-20 labels)	Constituency Parsing	Dependency Parsing	Quantity Normalization	Verb-sense Classification	Temporal Normalization	Mention Detection	Comma SRL	Preposition SRL	Propbank (verb) SRL	Nombank (nominal) SRL	Coreference resolution	Relation Extraction	Sentiment	OpenIE	Wikifier
	CogCompNLP (ours)	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Java	CoreNLP	✓	✓	✓	✓		✓	✓	✓	✓			✓	✓					✓	✓	✓	✓	
Java	OpenNLP	✓	✓	✓	✓	✓	✓		✓										✓				
Python	SpaCy	✓	✓	✓	✓	✓	✓	✓	✓	✓													
Python	NLTK	✓	✓	✓	✓	✓	✓																
Python	TextBlob	✓	✓	✓	✓	✓															✓		

Table 1: Summary of major annotators included in well-known NLP packages,

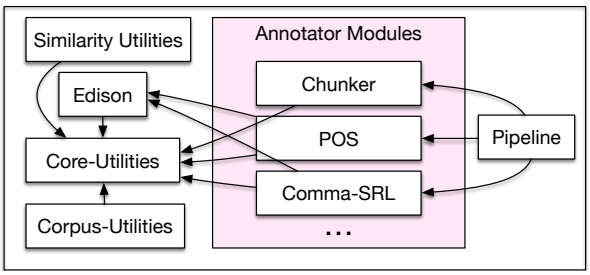


Figure 1: Diagram of the components in CogCompNLP and how they depend on each other. Each module to which an arrow is pointing at, is a dependee project. The arrows indicate the direction of dependence; for example Chunker depends on Core-Utilities.

The key data-structures for interaction between components – TextAnnotation, View, and Constituent – are illustrated in Figure 2. The TextAnnotation contains the raw source text together with its tokenization and other annotation layers added either by reading corpora annotations or applying NLP components. Token indexes are used to align constituents from different views, but each Constituent tracks its character offsets in the raw text. The figure shows Part-of-Speech, Named Entity, Quantity, and Semantic Role Label annotations, each in a separate View.

Corpus Utilities. While often overlooked and purely an engineering effort, implementing code that reads data correctly and efficiently can be a time-consuming task. CogCompNLP's corpus reader module includes NLP corpus readers that populate TextAnnotation objects. A few of the important datasets supported by this module are:

- Propbank Semantic Role Labeling
- Treebank Shallow Parse
- PennTreebank Constituency Parse
- Nombank Semantic Role Labeling
- ACE 2004/2005 Named Entity, Relation, Co-reference, and Event
- Ontonotes 5.0 POS, Named Entity, Syntactic Parse, and Semantic Role Labeling
- TAC/ERE Event, Relation, and Named Entity

Similarity Utilities. For calculating semantic similarity between words, phrases, and entities using both structured and distributional representations. Each similarity function compares objects (words, phrases, named entities, sentences) and returns a score indicating how similar they are. Depending on the inputs, different algorithms are available:

- *Word Similarity:* For computing the similarity between two words. The following representations are currently supported: `word2vec` (Mikolov et al., 2013), `paragram` (Wieting et al., 2015), `esa` (Gabrilovich and Markovitch, 2007), `glove` (Pennington et al., 2014), `wordnet` (Do et al., 2009), `phrase2vec` (Yin and Schütze, 2014). Here is a sample usage:

```
String representation = "esa";
WordSim ws = new WordSim(representation);
ws.compare("word", "sentence"); // 0.37
```

- *Named-Entity Similarity:* Comparing named entities requires a different class of algorithm. CogCompNLP's current algorithm is based on (Do et al., 2009):

```
NESim nesim = new NESim();
nesim.compare("Donald Trump", "Trump"); // 0.9
```

- *Phrasal Similarity:* Algorithms to combine lexical-level systems to make sentence-level decisions (Do et al., 2009):

```
Metric llm = new LLMStringSim(config);
String s1 = "Jack bought Alex's car";
String s2 = "Alex sold his car to Jack.";
llm.compare(s1, s2); // 0.75
```

Edison. Feature extraction is a crucial element in design of NLP systems. Edison (Sammons et al., 2016) is a feature extraction framework that uses the data-structures of CogCompNLP core-utilities to extract features to be used by machine learning algorithms. Edison enables users to define feature extraction functions that take as input the Views and Constituents created by CogCompNLP's Annotators. This makes it possible to not only develop feature sets like words, n-grams, and paths in parse trees, which work with a single View, but also more complex features that combine information from several Views.

This library has been successfully used to facilitate the feature extraction for several higher level NLP applications like Semantic Role Labeling (Punyakanok et al., 2005), Coreference resolution (Rizzolo and Roth, 2016) and, Textual Entailment (Sammons et al., 2010), which use information across several Views over text to make a decision.

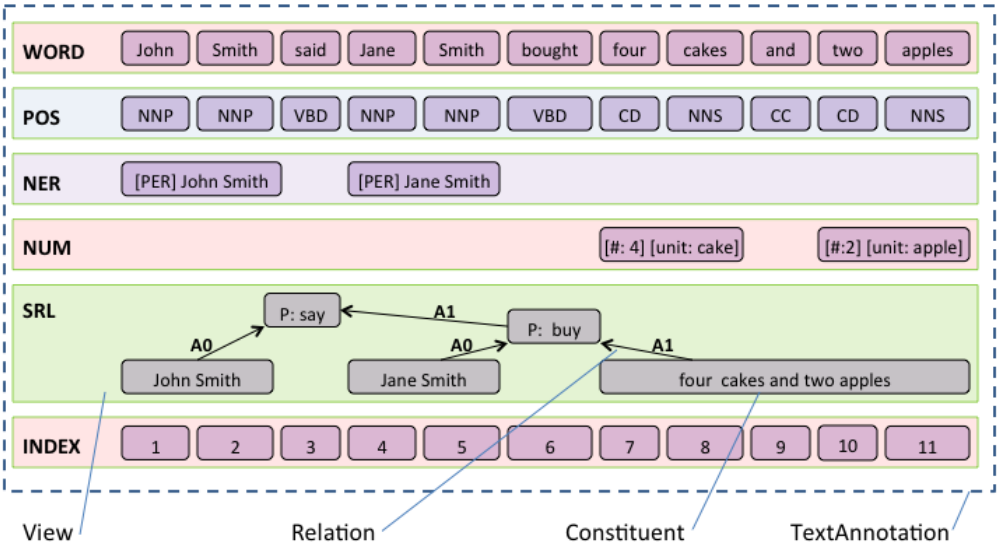


Figure 2: Illustration of the TextAnnotation, View, Constituent, and Relation data-structures in the Core Utilities module.

```
// assume 'srlTa' is a partially annotated text that comes from an earlier step
TextAnnotation srlTa = ...
AnnotatorService pipeline = PipelineFactory.buildPipeline(ViewNames.POS, ViewNames.NER_CONLL);
TextAnnotation augmentedSrlTa = pipeline.annotateTextAnnotation(srlTa);
List<Constituents> list = augmentedSrlTa.getView(ViewNames.POS).getConstituents();
System.out.println(list);  // (NNP Pierre) (NNP Vinken) (, ,) (CD 61) (NNS years) (JJ old) (, ,)...
```
Figure 3: Code snippet for using pipeline

Annotators. Given a set of corpus readers that load input data into unified data-structures, we build NLP annotators which share many features from EDISON and train with LBJAVA (Rizzolo and Roth, 2010) or Illinois-SL (Chang et al., 2015). This means that they are very easy to retrain or adapt to new domains and new languages. COGCOMPNLP includes a wide range of annotators and supporting functionality. Many of these annotators are state-of-the-art and are widely usable across different tasks.

The complete list of syntactic and semantic annotations we currently support is provided in Table 1, which also shows NLP components of other NLP frameworks. To the best of our knowledge this is the only NLP framework with such a large variety of syntactic and semantic components (in comparison, Stanford's CORENLP offers 14 components, OPENNLP 8, SPACY 9).

All of the the modules in our systems are actively updated to improve their quality, while maintaining their interoperability. These systems typically began as individual projects in previous work and have evolved over the years. In some cases the performance of components has been reduced for engineering purposes (e.g. faster speed, smaller memory footprint, etc). In Table 2 we present a qualitative assessment of the major components in COGCOMPNLP. The components have state-of-the-art quality or very close to the best existing results.

Pipeline. With all the Annotators generating the same data-structures, the PIPELINE project provides a simple interface to access Annotator components either individually or as a group, with a single function call. Use of PIPELINE is illustrated in Figure 3. A demo of PIPELINE is accessible online at `http://nlp.cogcomp.org`.

Accessibility from other programming languages. The majority of the implementation is in Java and hence COGCOMPNLP is easily accessible to JVM-based languages with direct calls.

To go beyond memory limitations of making direct calls to PIPELINE and make it available to other programming languages, the PIPELINE can be made accessible over a network. Using an internal web-server, PIPELINE can be made available over a network, making it accessible for a variety of programming languages. Essentially a user can instantiate an instance of PIPELINE server on a single machine (with sufficient memory), which can be shared by many users and queried from different languages.

We have created a Python interface, COGCOMPNLPY[1] which works with direct calls (using PyJnius and Cython (Behnel et al., 2011)) as well as over-network calls to the Java back-end. Here is an example snippet showing how to annotate a sentence with COGCOMPNLPY:

```
from ccg_nlpy import remote_pipeline
pipeline = remote_pipeline.RemotePipeline()
text = "Hello, how are you. I am doing fine"
ta = pipeline.doc(text)
print(ta.get_pos)
# (UH Hello) (, ,) (WRB how) (VBP are) (PRP you)...
```

4. Related Work

One important aspect of our work is the collection of the major NLP annotators. Table 1 contains a summary of components that exist in other well-established NLP libraries. CORENLP (Manning et al., 2014) is a popular

[1] `https://github.com/CogComp/cogcomp-nlpy`

Task	Dataset	Measure	Setting	Result
Tokenization	MASC (Ide et al., 2010)	Accuracy	–	97
POS (Roth and Zelenko, 1998)	Penn Treebank (Bies et al., 2015)	F1	–	96.13
NER (Ratinov and Roth, 2009; Redman et al., 2016; Tsai and Roth, 2016)	CoNLL (T. Kim Sang and De Meulder, 2003)	F1	–	91.12
	Ontonotes (Hovy et al., 2006)	F1	–	84.61
	MUC-7 (Chinchor, 1998)	F1	–	88.37
	Enron (Shetty and Adibi, 2004)	F1	–	77.21
	TAC-KBP 2016 EDL shared task	F1	English	88.3
		F1	Spanish	85
		F1	Chinese	79.3
Shallow Parse (Punyakanok and Roth, 2000)	CoNLL 2000 (Sang and Buchholz, 2000)	F1	–	93.58
Temporal Normalization (Zhao et al., 2012)	TempEval3 (UzZaman et al., 2013)	Exact match F1 / Relaxed match F1	Temporal Span Extraction	79.35/ 83.4
		F1	Temporal normalization, given a predicted temporal span	70.45
Mention Detection	ACE-05 (Walker et al., 2006)	F1	Head detection	89.6
		F1	Boundary detection given the head	89.45
	ERE	F1	Head detection	81.7
		F1	Boundary detection given the head	88.74
Relation Extraction (Chan and Roth, 2011)	ACE 2005 (Walker et al., 2006)	F1	Gold mention - Coarse Type	62.54
		F1	Gold mention - Fine Type	58.35
Taxonomic Relations (hypernyms, hyponyms, and co-hypernyms)	Test-I of Do and Roth (2012)	Accuracy	–	86.1
Comma SRL (Arivazhagan et al., 2016)	(Arivazhagan et al., 2016)	F1	–	83.6
Preposition SRL (Srikumar and Roth, 2013)	(Srikumar and Roth, 2013)	F1	–	90.26
Verb SRL (Punyakanok et al., 2004)	PropBank (Palmer et al., 2005a)	F1	–	76.22
Nominal SRL(Punyakanok et al., 2004)	NomBank (Meyers et al., 2004)	F1	–	66.97
Coreference (Samdani et al., 2014)	CoNLL-12 (Pradhan et al., 2012)	Average of F1 score of MUC, B^3	Gold mentions	77.05
			Predicted Mentions	60
	ACE-04 (Doddington et al., 2004)	B^3	Gold mentions	79.42
			Predicted Mentions	68.27
Wikifier (Tsai and Roth, 2016)	TAC-KBP 2016 EDL shared task	F1	English: NER+ link to freebase	76.5
		F1	Spanish: NER+ link to freebase	75.1
		F1	Chinese: NER+ link to freebase	70.6

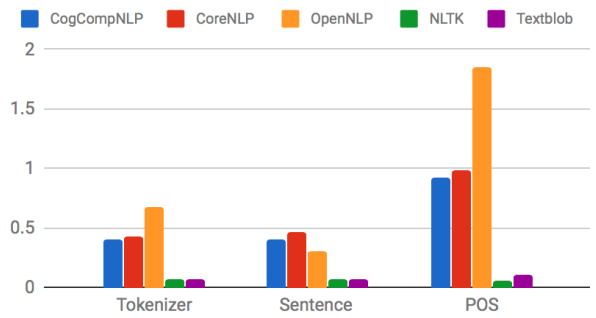
Table 2: Qualitative evaluation of major components included in CogCompNLP.

library which contains various NLP components. While missing a few modules (OpenIE and Constituency parsing), CogCompNLP contains high-level annotators for Semantic Role Labeling of verbs, nouns, commas, and prepositions, while also offering a much wider range of supporting functionality. Compared to OpenNLP (Baldridge, 2005), SpaCy[2], NLTK (Bird, 2006) and TextBlob (Loria et al., 2014), CogCompNLP has many extra annotators. While CogCompNLP is implemented in Java, its sister project CogCompNLPy makes it accessible in Python.

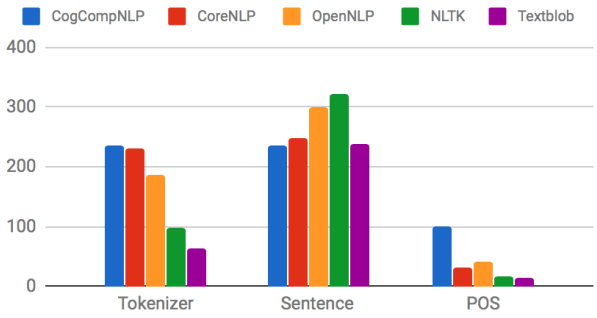
Memory and speed comparison. To measure speed and memory, we compile a collection of English raw text files extracted from a news corpus (NYT files from English Gigaword v5). The collected plain text corpus has 999 files, 665233 words (using linux `'wc -w'` command).
To measure time and memory, we use the GNU `time` command, which is programming-language agnostic. It reports the time that it takes to run the program from start to end, and it also captures the maximum resident set size of the process during its lifetime. In reporting time we use three definitions: (a) *wall-clock* time is the time that a clock on the wall (or a stopwatch in hand) would measure as having elapsed between the start and end of the process. (b) *user time* is the amount of time spent in user code, and (c) *system time* the amount of time spent in the kernel. For each of the systems we use their latest available version (summarized in Table 3).

Often NLP systems are implemented as a sequence of interdependent components, and one would ideally measure their specifications using only the subsequence up to and including the component being evaluated. However, not all systems have the same ordering of the components internally. In what follows, we show the speed/memory it takes

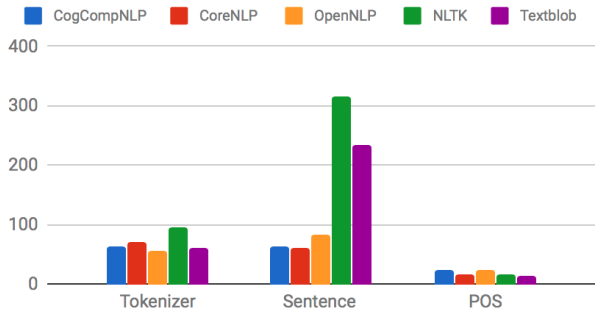
[2] `https://spacy.io`



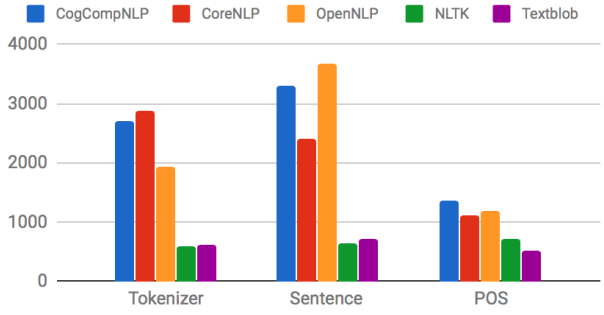
(a) Memory (GB): **lower is better**. Python-based tools have lower memory footprint.



(b) Speed: Wall clock time (thousand tokens / second): **higher is better**.



(c) Speed: User time (thousand tokens / second): **higher is better**.



(d) Speed: System time (thousand tokens / second): **higher is better**.

Figure 4: Speed and memory comparison between major NLP pipelines. SPACYis not shown in any of the figures, since its components are not easy to separate.

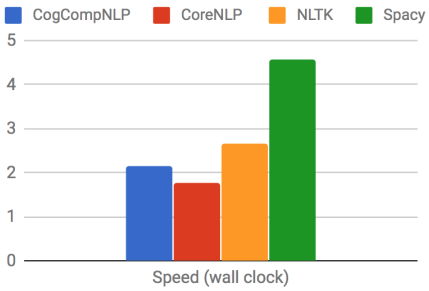


Figure 5: Speed: Wall clock time (thousand tokens / second): **higher is better**. The systems are set to produce *Tokenizing, Sentence Splitting, POS tagging and Named Entity Recognition*. TEXTBLOB is not presented since it does not have an NER.

	Tool	Version
Software	COGCOMPNLP	4.0.2
	CORENLP	3.8.0
	SPACY	2.0.5
	NLTK	3.2.5
	TEXTBLOB	0.15.1
	Maven	3.5.2
	Java	1.8.0-151
	Python	3.5.2
Hardware	CPU	12 × Intel Xeon six core 3.2GHz
	Memory	32GB

Table 3: Specs of the software and hardware used in our evaluation. For a fair comparison, we used the same machine to run all components.

to run the component being evaluated, with everything not needed being turned off.

The results of the speed evaluation for the three tasks of sentence-splitting, tokenizing, and POS tagging are summarized in Figure 4. Overall, JDK-based systems tend to need more memory (subfigure (a)). Differences in speed are often related to the task at hand; for example for POS tagging or tokenizing we do not see any significant difference.

In Figure 4 we do not show any results on SPACYsince unlike other tools, in SPACY we did not find a clean-cut way to study individual modules in isolation, In order to compare other tools with SPACY we perform another experiment where all the systems annotate Tokenizing, Sentence Splitting, POS tagging and Named Entity Recognition together. The results of this speed evaluation is depicted in Figure 5.

5. SRL: A Sophisticated Application

In this section we illustrate the ways COGCOMPNLP supports development and evaluation of complex NLP applications using the task of Semantic Role Labeling (SRL) as an example. SRL identifies predicate-argument structures in

```
String corpusDir = "/local/path/to/ontonotes5.0/corpus/";
OntonotesReader srlReader = new OntonotesSrlReader(corpusDir);
while (srlReader.hasNext()) {
    TextAnnotation documentSrl = srlReader.next();
    PredicateArgumentView srlView = (PredicateArgumentView) documentSrl.getView(ViewNames.SRL_VERB);
    for (Constituent verbPredicate : srlView.getPredicates()) {
        for (Relation argumentRelation : srlView.getArguments(verbPredicate)) {
            String argumentType = argumentRelation.getRelationName();
            Constituent argument = argumentRelation.getTarget();
            ...
    }
}
```

Figure 6: Code snippet for reading OntoNotes and iterating over SRL predicates

```
// assume our new component is called 'MySrlApp'
Annotator mySrlApp = new MySrlApp(); // initialize the SRL annotator
AnnotatorService pipeline = PipelineFactory.buildPipeline(ViewNames.Lemma, ViewNames.POS);
pipeline.addAnnotator(mySrlApp); // add the annotator to an pipeline
String text = "John Smith said Jane Smith bought four cakes and two apples.";
TextAnnotation outputTa = pipeline.createAnnotatedTextAnnotation(text); // annotate with annotators
boolean useJson = true;
SerializationHelper.serializeTextAnnotationToFile(outputTa, "outputFile.json", useJson);
```

Figure 7: Code snippet for adding a new component to the pipeline. The expected output is shown in Figure 2.

text; while nouns, prepositions, and other word types may express predicates we focus here on verb SRL (Palmer et al., 2005b), using the OntoNotes 5.0 corpus (Weischedel et al., 2013). Due to space requirements, we will focus on representative elements of the task in some steps, but the overall illustration should allow other NLP researchers to easily implement the relevant steps for themselves and the tasks they are interested in.

5.1. Reading the Data

The OntoNotes corpus has many layers of NLP annotations stored in a deep directory structure. The CogCompNLP Ontonotes reader loads each document and its annotations into a TextAnnotation data-structure (see the indicated Views in Figure 2, which shows only one representative sentence). The user can iterate over the predicate argument structures to process each in turn as indicated in the code snippet in Figure 6.

5.2. Adding more NLP information

In order to predict SRL structures, we want to use other syntactic and semantic information represented by the source text, such as the outputs from a syntactic parser and a named entity recognizer. Figure 3 illustrates the use of CogCompNLP's Pipeline to add these annotations to the same data-structure returned by the Ontonotes reader in section 5.1.

5.3. Candidate Extraction & Feature Generation

SRL requires a number of component predictions that are assembled to produce the final output. One such component identifies argument boundaries, and a typical approach is to generate a number of candidates and then predict for each whether it is a valid SRL argument. The code snippet in Figure 8 enumerates the syntactic parse constituents and applies Edison feature extractors to generate inputs for a machine learning component.

5.4. Training and Evaluating a Classifier

Now that we have features extracted, we can generate examples for a classifier by comparing candidate boundaries with gold standard SRL argument boundaries to determine the appropriate label. These examples can be collected and passed to a learning component defined using LBJava (Rizzolo and Roth, 2010), as shown in Figure 9. (see (Rizzolo and Roth, 2010) for a detailed overview and examples of LBJava).

The learner's performance can be easily computed using classes from CogCompNLP's core-utilities module. The library provides a range of supporting functionality for cross-validation and computation of experimental statistics that goes well beyond what we can illustrate here.

5.5. Using the New NLP Component

Once a new component/application has been developed, its outputs can easily be made available to other applications either via serialization, or by making the component an Annotator and adding it to the pipeline. Figure 7 shows how these tasks can be done.

6. Conclusion

CogCompNLP is a mature, well-architected, and largely well-documented NLP framework available under an academic license. It has been developed with the twin goals of making it easy to obtain NLP annotations from off-the-shelf components, and developing and evaluating new applications. In this paper we explain the NLP tools it contains; the pipeline application for applying these tools to input text; its word and phrase similarity metrics. In an earlier work (Sammons et al., 2016) we have described its support for feature extraction. We have illustrated its use to develop sophisticated NLP applications by showing how to implement Semantic Role Labeling, a key NLP component.

```
 /** predicate for selecting candidate predicates (here, verbs) */
private static class IsVerb extends Predicate<Constituent> {
    // expects POS constituent
    @Override
    public Boolean transform(Constituent c) {
        return c.getLabel().startsWith("V");
    }
}

/** extract positive and negative examples for training. */
public static List<Pair<String, Pair<Constituent, Constituent>>> generateExamples(TextAnnotation ta) {
    List<Pair<String, Pair<Constituent, Constituent>>> examples = new ArrayList<>();
    PredicateArgumentView verbSrlView = (PredicateArgumentView) ta.getView(ViewNames.SRL_VERB);
    IQueryable<Constituent> verbs =
            new QueryableList<>(ta.getView(ViewNames.POS).getConstituents()).where(new IsVerb());
    TreeView parseView = (TreeView) ta.getView(ViewNames.PARSE_STANFORD);

    for (Constituent verb : verbs) {
        List<Constituent> allArgCandidates = getArgCandidates(parseView, verb.getStartSpan());
        for (Constituent candidate: allArgCandidates) {
            String label = "negative";
            if (findGoldArgMatch(candidate, verb, verbSrlView)) label = "positive";
            examples.add(new Pair(label, new Pair(candidate, verb)));
        }
    }
    return examples;
}

/** check in SRL View whether candidate argument, predicate match a gold argument, predicate pair */
public static boolean findGoldArgMatch(Constituent candidate, Constituent verb, PredicateArgumentView srlView) {
    List<Constituent> srlArgMatches =
            srlView.getConstituentsMatchingSpan(candidate.getStartSpan(), candidate.getEndSpan());
    Set<Constituent> srlPredMatches = new HashSet<>(
            srlView.getConstituentsMatchingSpan(verb.getStartSpan(), verb.getEndSpan()));
    srlPredMatches.retainAll(new HashSet<>(srlView.getPredicates()));

    for (Constituent srlArgMatch : srlArgMatches)
        for (Relation rel : srlArgMatch.getIncomingRelations())
            if (srlPredMatches.contains(rel.getSource())) return true;

    return false;
}

/** use syntactic parse to identify parse constituents near the predicate */
public static List<Constituent> getArgCandidates(TreeView parseView, int predicateIndex) {
    // consider siblings and child nodes of predicate
    List<Constituent> argumentCandidates = new ArrayList<>();
    Tree<Constituent> t = parseView.getConstituentTree(0);
    Tree<Constituent> predicateTree = t.getYield().get(predicateIndex).getParent();
    Tree<Constituent> predicateParent = predicateTree.getParent();

    for (Tree<Constituent> sibling : predicateParent.getChildren())
        argumentCandidates.add(sibling.getLabel());

    for (Tree<Constituent> child : predicateTree.getChildren())
        if (child != predicateTree) argumentCandidates.add(child.getLabel());

    return argumentCandidates;
}
```

Figure 8: Code snippet for generating verb SRL argument examples.

Acknowledgement

The authors would like to thank all the other contributors to the project. This material is partly based on research sponsored by DARPA under agreement number FA8750-13-2-0008. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. This work was also supported by Contract HR0011-15-2-0025 with the US Defense Advanced Research Projects Agency (DARPA). Approved for Public Release, Distribution Unlimited. This work was partly funded by a grant from the Allen Institute for Artificial Intelligence (allenai.org); by Google; by NSF grant BCS-1348522; and by NIH grant R01-HD054448. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any research sponsors listed above.

References

Arivazhagan, N., Christodoulopoulos, C., and Roth, D. (2016). Labeling the semantic roles of commas. In *Proc. of the Conference on Artificial Intelligence (AAAI)*, 2.

Baldridge, J. (2005). The OpenNLP project. *URL: http://opennlp. apache. org/index. html,(accessed 2 February 2012)*.

Behnel, S., Bradshaw, R., Citro, C., Dalcin, L., Seljebotn, D. S., and Smith, K. (2011). Cython: The best of both worlds. *Computing in Science & Engineering*, 13(2):31–39.

```
// a trivial classifier, to return a feature for a predicate-argument structure
public class SrlWordAndPos extends Classifier {
    private WordAndPos wordAndPos = new WordAndPos();
    @Override
    public FeatureVector classify(Object arg) {
        Pair<Constituent, Constituent> argAndPred = ((Pair<String, Pair<Constituent, Constituent>>) arg).getSecond();
        return FeatureUtilities.getLBJFeatures(FeatureUtilities.conjoin(
            wordAndPos.getFeatures(argAndPred.getFirst()), wordAndPos.getFeatures(argAndPred.getSecond())));
    }
}
```

```
// LBJava code, to define a classifier, its input features and the expected output
import SrlWordAndPos;

discrete ArgumentIdentifier(Pair<String, Pair<Constituent, Constituent>> argAndPred) <-
learn IsArgument
  using SrlWordsAndPos
  from new SrlArgumentReader(Constants.corpusHome) 50 rounds
  with SparseNetworkLearner {
    SparseAveragedPerceptron.Parameters p = new SparseAveragedPerceptron.Parameters();
    p.learningRate = .1;
    baseLTU = new SparseAveragedPerceptron(p);
  }
```

Figure 9: LBJava code for training and evaluating an SRL argument identifier classifier. The `ArgParser` argument is a class that extends LBJava's `Parser` interface, and returns an argument example (label 'positive' or 'negative', argument constituent, and predicate constituent) for each call of its `next()` method.

Bies, A., Mott, J., and Warner, C. (2015). English news text treebank: Penn treebank revised.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Chan, Y. and Roth, D. (2011). Exploiting syntactico-semantic structures for relation extraction. In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Portland, Oregon.

Chang, K.-W., Upadhyay, S., Chang, M.-W., Srikumar, V., and Roth, D. (2015). Illinoissl: A java library for structured prediction. *arXiv preprint arXiv:1509.07179*.

Chinchor, N. A. (1998). Overview of muc-7/met-2. Technical report, SCIENCE APPLICATIONS INTERNATIONAL CORP SAN DIEGO CA.

Clarke, J., Srikumar, V., Sammons, M., and Roth, D. (2012). An nlp curator (or: How i learned to stop worrying and love nlp pipelines). In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*, 5.

Do, Q. and Roth, D. (2012). Exploiting the wikipedia structure in local and global classification of taxonomic relations. *Journal of Natural Language Engineering (JNLE)*, 18(2):235–262, 4.

Do, Q., Roth, D., Sammons, M., Tu, Y., and Vydiswaran, V. (2009). Robust, light-weight approaches to compute lexical similarity. *Computer Science Research and Technical Reports, University of Illinois*, page 94.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.

Gabrilovich, E. and Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJcAI*, volume 7, pages 1606–1611.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Ide, N., Fellbaum, C., Baker, C., and Passonneau, R. (2010). The manually annotated sub-corpus: A community resource for and by the people. In *Proceedings of the ACL 2010 conference short papers*, pages 68–73. Association for Computational Linguistics.

Loria, S., Keen, P., Honnibal, M., Yankovsky, R., Karesh, D., Dempsey, E., et al. (2014). Textblob: simplified text processing. *Secondary TextBlob: Simplified Text Processing*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The nombank project: An interim report. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Palmer, M., Gildea, D., and Kingsbury, P. (2005a). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Palmer, M., Gildea, D., and Kingsbury, P. (2005b). The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106, March.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In

Joint Conference on EMNLP and CoNLL-Shared Task, pages 1–40. Association for Computational Linguistics.

Punyakanok, V. and Roth, D. (2000). Shallow parsing by inferencing with classifiers. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*, pages 107–110.

Punyakanok, V., Roth, D., Yih, W., and Zimak, D. (2004). Semantic role labeling via integer linear programming inference. In *Proc. of the International Conference on Computational Linguistics (COLING)*, pages 1346–1352, Geneva, Switzerland, 8.

Punyakanok, V., Roth, D., and Yih, W. (2005). The necessity of syntactic parsing for semantic role labeling. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1117–1123.

Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*, 6.

Redman, T., Sammons, M., and Roth, D. (2016). Illinois named entity recognizer: Addendum to ratinov and roth '09 reporting improved results. Technical report.

Rizzolo, N. and Roth, D. (2010). Learning based java for rapid development of nlp systems. In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta, 5.

Rizzolo, N. and Roth, D. (2016). Integer linear programming for co-reference resolution. In Roland Stuckardt Massimo Poesio et al., editors, *Anaphora Resolution: Algorithms, Resources, and Applications*. Springer-Verlag.

Roth, D. and Zelenko, D. (1998). Part of speech tagging using a network of linear separators. In *Coling-Acl, The 17th International Conference on Computational Linguistics*, pages 1136–1142.

Samdani, R., Chang, K.-W., and Roth, D. (2014). A discriminative latent variable model for online clustering. In *Proc. of the International Conference on Machine Learning (ICML)*.

Sammons, M., Vydiswaran, V., and Roth, D. (2010). Ask not what textual entailment can do for you... In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Uppsala, Sweden, 7. Association for Computational Linguistics.

Sammons, M., Christodoulopoulos, C., Kordjamshidi, P., Khashabi, D., Srikumar, V., Vijayakumar, P., Bokhari, M., Wu, X., and Roth, D. (2016). Edison: Feature extraction for nlp, simplified. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proc. of the International Conference on Language Resources and Evaluation (LREC)*. European Language Resources Association (ELRA).

Sang, E. F. T. K. and Buchholz, S. (2000). Introduction to the conll-2000 shared task chunking. In *CoNLL/LLL*, pages 127–132. Association for Computational Linguistics.

Shetty, J. and Adibi, J. (2004). The enron email dataset database schema and brief statistical report. *Information sciences institute technical report, University of Southern California*, 4(1):120–128.

Srikumar, V. and Roth, D. (2013). Modeling semantic relations expressed by prepositions. 1:231–242.

T. Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Tsai, C.-T. and Roth, D. (2016). Illinois cross-lingual wikifier: Grounding entities in many languages to the english wikipedia. In *Proc. of the International Conference on Computational Linguistics (COLING) Demonstrations*, 12.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 1–9.

Walker, C., Strassel, S., Medero, J., and Maeda, K. (2006). Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., et al. (2013). Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*.

Wieting, J., Bansal, M., Gimpel, K., Livescu, K., and Roth, D. (2015). From paraphrase database to compositional paraphrase model and back. *arXiv preprint arXiv:1506.03487*.

Yin, W. and Schütze, H. (2014). An exploration of embeddings for generalized phrases. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 41–47.

Zhao, R., Do, Q., and Roth, D. (2012). A robust shallow temporal reasoning system. In *Proc. of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 6.

A Framework for the Needs of Different Types of Users in Multilingual Semantic Enrichment

Jan Nehring[+], **Felix Sasaki**[++]
[+]German Research Centre for Artifical Intelligence Berlin, Germany
[++]Cornelsen, Berlin, Germany
jan.nehring@dfki.de, felix@sasakiatcf.com

Abstract

The FREME framework bridges Language Technologies (LT) and Linked Data (LD). It establishes workflows between LT and LD in a well defined, coherent way. FREME addresses common challenges that both researchers and industry face when integrating LT and LD: interoperability, "silo" solutions and the lack of adequate tooling. Usability, reusability and interoperability are often attributes of frameworks and toolkits for LT and LD. In this paper, we take a novel approach: We define user types and user levels and describe how they influence design decisions in a LT and LD processing framework. In this way, we combine research outcomes from various communities: language technology, linked data and software interface engineering. This paper explains the different user types and how FREME addresses the specific needs of each user type. Core attributes of FREME are usability, reusability and interoperability.

Keywords: language resource, language technology, usability, user types, linked data, software interface engineering

1. Introduction

Language Technology (LT) profits from the current rise of Artificial Intelligence and is being more and more integrated in commercial applications. The growing amount of digital content calls for technologies to process and enrich digital content in an automated manner. Both industry and academia need adequate tooling to process digital content. The amount of available tools is growing, but each tool uses its own data format. Merging annotations of digital content lacks a unified approach (Hellmann et al., 2013; Sanderson et al., 2013). Knowledge resources have become a key component of current systems in Artifical Intelligence (Flati et al., 2014). Construction and exploitation of such knowledge sources has gained attraction from both researchers (Mitchell, 2005; Navigli and Ponzetto, 2012) and big industry players like Google (Singhal, 2012) or IBM (Ferrucci, 2012). A popular approach to store and connect knowledge sources is Linked Data (LD).

The FREME framework bridges Language Technologies and Linked Data (Dojchinovski et al., 2016). It establishes workflows between LT and LD in a well defined, coherent way. FREME addresses common challenges that both researchers and industry face when integrating LT and LD: interoperability, "silo" solutions and the lack of adequate tooling (Sasaki et al., 2015a).

Usability, reusability and interoperability are often attributes of frameworks and toolkits for Language Technologies (LT) and Linked Data (LD). Examples are (Bachmann-Gmur, 2013; Hinrichs et al., 2010; Klein et al., 2017; Noji and Miyao, 2016). In this paper, we take a novel approach: We define user types and user levels and describe how they influence design decisions in a LT and LD processing framework. In this way, we combine research outcomes from various communities: language technology, linked data and software interface engineering. This paper explains the different user types and how FREME addresses the specific needs of each user type. Core attributes of FREME are usability, reusability and interoperability.

2. Background

This section first explains the different communities that are being addressed and bridged by the FREME framework. Then it explains the Natural Language Processing Interchange Format (NIF).

2.1. Bridging Language Resources, Language Technology and Engineering

Many approaches exist to integrate LD and LT, e.g. for Named Entity Linking (Ehrmann et al., 2016; Usbeck et al., 2014), Machine Translation (Srivastava et al., 2017) or Sentiment Analysis (Paul Buitelaar and Strapparava, 2013) but combining the two is still cumbersome and lacks a unified approach. Often LT profits from the combination with knowledge (Navigli and Ponzetto, 2012; Ristoski and Paulheim, 2016) but bridging language tools and knowledge sources in a well established, easy to use and coherent workflow is still a challenge. Researchers bridge LD and LT by storing the output of language tools using LD formats (Hellmann et al., 2013; Sanderson et al., 2013). Furthermore several challenges arise when these worlds meet, e.g. many different content formats to process; adaptability and avoiding "silo solutions"; usability and a lack of adequate tooling (Sasaki et al., 2015b). The FREME framework provides solutions to overcome these hardships.

2.2. The Natural Language Processing Interchange Format

FREME uses the Natural Language Interchange Format (NIF) as a common broker format to ensure that different LT and LD services are interoperable. NIF is an RDF/OWL based format that defines a common vocabulary to describe NLP annotations (Hellmann et al., 2013). NIF addresses the interoperability of NLP tools on three layers: It is based on Linked Data (structural layer) and a selection of ontologies to describe common NLP terms and concepts (conceptual layer). NIF aware applications are accessible via REST services (access layer). In the FREME terminology, such services are called e-Service.s

Every e-Service in FREME uses NIF or plaintext as input format and produces NIF as output format. Therefore, it is possible to feed the output of one e-Service in the next e-Service to form pipelines.

```
@prefix itsrdf: <http://www.w3.org/2005/
11/its/rdf#> .
@prefix nif: <http://persistence.uni-leipzig
.org/nlp2rdf/ontologies/nif-core#> .

<http://freme-project.eu/#char=0,18>
  a nif:String , nif:RFC5147String ,
    nif:Context ;
  nif:beginIndex "0"
    ^^xsd:nonNegativeInteger ;
  nif:endIndex "18"
    ^^xsd:nonNegativeInteger ;
  nif:isString "Welcome to Berlin!"
    ^^xsd:string ;
  itsrdf:mtConfidence "0.725583686873588" ;
  itsrdf:target "Willkommen in Berlin!"@de .
```

Listing 1: Example of NIF for a translation

Listing 1 shows an example of NIF in the turtle serialization format. It shows the translation of an English document to German. The document is identified by a URL. This URL has further annotations, for example the textual content "Welcome to Berlin!" or the begin and end indizes of the annotation. The translation (itsrdf:target) and the confidence value (itsrdf:mtConfidence) of the translation system are expressed via the ITSRDF vocabulary.

3. User types

The design of FREME evolves around the following user types:

User level 0 is a technology specialist in digital content management, big data and multilingual and semantic technologies. He features specialised skills relevant to a certain domain. User level 0 can create FREME and its components. Innovation is generated at this user level and the outcomes are utilized by user levels 1, 2 and 3.

User level 1 is a data expert and a technology user. He has skills relevant to LD vocabularies and LD technologies. These users deploy FREME for working with concrete datasets. They build on the technologies developed by user level 0 and provide the basis for user levels 2 and 3.

User level 2 builds user interfaces and applications using FREME APIs and technologies provided by the lower user levels. His skill set is specialised on application development and knowledge of LT and LD technologies is usually low. The interface and application developer uses the high-level APIs provided by user level 1 to provide the basis for user level 3. Examples for user level 2 are web site architects or application developers.

User level 3 is the end user, e.g. a content creator, translator / localiser, publisher or others that create, process or consume content. He uses GUIs provided by user level 2 and is often not aware that the technologies he uses builds on FREME.

4. The FREME Framework

This section explains the architecture of FREME and its components in context of its user roles and the core attributes usability, reusability and interoperability.

4.1. Architecture of FREME

The FREME framework exhibits a set of LT and LD services as HTTP APIs. FREME applications are client / server applications with the user being the client and the FREME API being the server. Figure 1 shows a birds-eye view on the architecture.

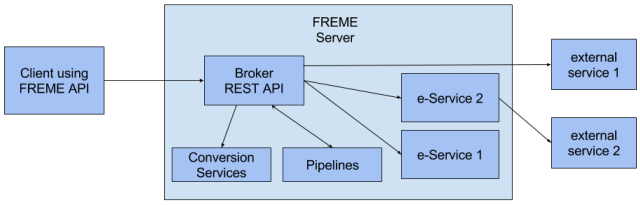


Figure 1: Architecture of FREME

The mandatory element of every FREME installation is the broker which acts as the entry point for every HTTP request. It redirects the request to the target e-Service or other module. An e-Service is a NLP enrichment service, e.g. Named Entity Recognition or Sentiment Analysis. Every e-Service uses NIF as input and output format. Each e-Service exhibits at least one HTTP endpoint that executes a certain LT or LD task. Every e-Service adds annotations to an existing NIF document. These annotations are called enrichments in the FREME terminology.

We chose this architecture for several reasons:

- The architecture allows a division of labour between language technology experts (user level 0-1) and application developers (user level 2). The language technology experts maintain the FREME server while the application developers merely use the service.
- Clients can use any programming language and any operating system.
- Further the clients can be lightweight because the server performs heavy processing.

(Sasaki et al., 2016) explain the architecture in more detail. The FREME framework offers the following e-Services out of the box:

- e-Entity for named entity recognition, classification and linking to the LD cloud
- e-Translation for statistical machine translation
- e-Terminology for terminology annotation

(Sasaki et al., 2016) and the FREME documentation give a more detailed overview about the services integrated in FREME. Other users of the framework provide additional e-Services. The project Digital Curation Technologies provides the following e-Services compatible with FREME (Rehm et al., 2017; Rehm and Sasaki, 2016):

- named entity recognition and linking

- event detection
- geograhical localisation
- temporal expression analysis
- text classification
- text clustering
- monolingual and cross-lingual event detection
- single and multi-document summarisation
- machine translation

4.2. Concepts for different user levels

This section explains how the design of FREME helps the different user levels. This section explains benefits mainly for user level 1 and 2.

4.2.1. High level pipelines for easy integration

More and more LT tools exist but integrating these tools and managing inputs and outputs is still patchy. Each tool uses its own input and output format. Partially this problem is being addressed by NIF but FREME has an additional approach to ease the integration of LT and LD pipelines.
The individual steps of a traditional pipeline in the sense of e.g. UIMA (Ferrucci et al., 2009) or Gate (Cunningham et al., 2011) operate on a low level of abstraction, e.g. sentence splitting or tokenization. In FREME atomic steps operate on a high level of abstraction. So a single pipeline step can be e.g. Named Entity Enrichment, Machine Translation or similar. This means that a single step in a FREME pipeline often consists of a pipeline itself. These high level pipelines hide the low level complexity of the NLP tooling from the user. A single pipeline step is self-contained and does all the pre-processing of the data it needs itself. In this way, it is possible to use NLP services without detailed knowledge of their inner workings. Further one can rapidly develop FREME enabled pipelines and exchange processing steps at will. The NLP services do not pass the results of the low-level processing along the pipeline so this approach might lead to a lower performance since some steps may need to be done twice. So this approach is a trade-off between usability and performance.
The benefits for this approach are located at user level 1 and 2 because they can make use of the easy pipelines. User level 0 has to be aware of this concept so he can create services accordingly.

4.2.2. Accessing Linked Data Cloud without SPARQL

User level 2 usually has no expertise in using LD and SPARQL. Therefore, these technologies are often considered challenging. In order to be able to exploit the benefits of Linked Data without using SPARQL, FREME has introduced a division of labor between user level 1 and 2 using a mechanism called e-Link template. An e-Link template is a SPARQL CONSTRUCT query that is executed on a NIF document. It can for example extract all annotations of entities of type ”city” in the NIF document and then enrich the document with museums or other tourist attractions located in these cities. Another use case is to fetch additional information about persons like the birthday or a picture. These templates are stored by the FREME API. User level 1 defines these templates while user level 2 uses them. During the FREME project e-Link templates were used in several occasions and we found that they were very useful for both user levels because each could concentrate on what they do best and it established a clear workflow. Further it allows pipelining workflow without any hard-coded elements because defining e-Link templates requires configuration instead of programming. (Brümmer et al., 2016) explain the e-Link service in more detail.

4.2.3. Use Linked Data workflows without Linked Data knowledge using SPARQL filters

FREME has developed another concept to allow using LD based workflows without knowledge of LD. Although internally the pipelines use LD only, it allows other input and output formats.
FREME supports a wide range of input formats apart from LD, e.g. plaintext, HTML or PDF. In addition, it has a build in mechanism called SPARQL filter to convert LD output of the pipeline to a tabular format. A SPARQL filter is a SPARQL SELECT query that is applied to the NIF as the last step of the pipeline. A SPARQL SELECT query converts the graph based RDF data to tabular data which can be represented in Comma Separated Values, JSON or similar. User Level 1 defines the SPARQL converter, then user Level 2 uses it without getting in touch with Linked Data. Figure 2 shows an example of a pipeline that uses PDF as input format and tabular data as output format.



Figure 2: Example pipeline that uses PDF as input fornat and tabular data as output format

Another use case for the SPARQL filter API is simplification of the output. Pipeline output often contains a lot of information while only a subset of information is of interest. The SPARQL filter can reduce the size of the output and e.g. output person annotations only and strip all other information.

4.2.4. Use Linked Data workflows without Linked Data knowledge using e-Internationalization and XSLT

Using the XSLT API FREME can integrate in XML based workflows. The XSLT API can convert a document between several XML data formats using XSLT stylesheets that are stored on the server. E-Internationalization provides functionality to convert data from HTML5, which is also XML based, to NIF and back. Then NIF is converted to HTML5, enrichments will be embedded in HTML5 using the Internationalization Tag Set . Using this workflow, it is possible to define a pipeline that accepts XML as input and produces XML as output. This workflow is useful for user level 2. Figure3 shows an example of a pipeline that uses XML as input and output format.

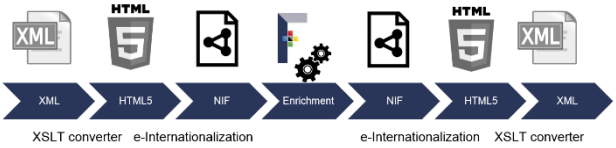


Figure 3: Example pipeline using XML as input and output format

4.2.5. Ready to use language resources, data resources and services

FREME offers a range of data services ready to use under an open license. This is useful for user level 1, although user level 2 can also benefit from the quick integration of these data services.

- Named entity recognition, classification and linking in 6 languages (English, German, Dutch, French, Italian, Russian) trained on the DBPedia Abstracts corpus (Brümmer et al., 2016)
- Datasets: DBPedia, Geopolitical Ontology, ORCID (Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, 2012), Statbel, Global Airports, Cordis, VIAF (Bennett et al., 2006), ONLD, GRID, FAO (Kim et al., 2013)
- e-Link templates: FREME offers a series of e-Link templates that can enrich NER annotations with information from the Linked Open Data cloud.
- SPARQL converters: We offer a series of converters to store RDF data as CSV for easier integration.
- FREME offers a set of common XSLT stylesheets to convert between XML formats using the XSLT converter service.

These LT and LD services are available through the official live instance of FREME. Further they can be downloaded and integrated in several ways into an on-premise FREME installation.

Several datasets were created or converted to Linked Data and integrated directly in the FREME framework. The rest of this section explains these datasets, in-depth information and downloads are located on Datahub[1]. These datasets help user level 1 and 2 because they can be used without integration work.

The DBPedia Abstracts corpus (Brümmer et al., 2016) contains the abstracts of wikipedia articles in seven languages. The dataset provided training data for the e-Entity Named Entity Recognition service. Further it provides data for Named Entity Linking.

The Statbel corpus contains RDF conversion of datasets from "Statistics Belgium" which aims at collecting, processing and disseminating relevant, reliable and commented data on Belgian society.

OpenFlights.org contains a dataset about airport names, their locations, codes and other related info. The Global Airports dataset is an RDF version of this data. In FREME it was used to enrich text for the tourism domain.

Open Researcher and Contributor ID (ORCID) is an index to uniquely identify scientific and other academic authors (Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, 2012). The 2014 version was converted to RDF.

The Global Research Identifier Database (GRID) contains information about worldwide research organizations. For FREME it was converted to RDF.

The Community Research and Development Information Service (CORDIS) is a dataset about EU fundet research projects which was converted to RDF during the FREME project.

Other datasets that were already previously available and are now ready to use out of the box are VIAF (Bennett et al., 2006) and FAO (Kim et al., 2013).

5. Related Work

Apache Stanbol (Bachmann-Gmur, 2013), Weblicht (Hinrichs et al., 2010) and NLP curator (Clarke et al., 2012) are available as web services in a Software as a Service manner. The Unstructured Information Management Architecture (UIMA) (Ferrucci, 2012) has an extension to turn it into a web API (The Apache UIMA Development Community, 2008).

None of the above mentioned systems use a standardized data exchange format based on Linked Open Data. Since UIMA and Gate (Cunningham et al., 2011) are very popular their data formats have turned into quasi standards because of their wide adoption across several tools.

The Speech Analytics Platform (Batista et al., 2016) integrates several speech processing modules. It was developed with the aim to make usability of the modules as easy of possible. It has similar design principles as FREME: It is accessible as an API and provides a simple workflow to add new services. Further it can be used in a Graphical User Interface from the web browser.

Clarin Weblicht (Hinrichs et al., 2010) is similar to FREME because it also provides a web based execution environment and pipelines can span several APIs. Clarin Weblicht makes it easy to create pipelines in a web interface. It has a special emphasis on usability and interoperability because it targets users which do not have a technical background (Hinrichs and Krauwer, 2014). These users are from user levels 1 and 3.

The Jigg framework (Noji and Miyao, 2016) provides a methodology to integrate services for user level 1. Jigg suggests a workflow that relies on wrapping tools like Stanford Core NLP (Manning et al., 2014) with a Java objects, exchanging these objects in a pipeline and outputing the results in a proprietary XML format.

None of the above mentioned systems uses FREMEs concept of high level pipeline components. Pipelines in the sense of above systems are low level and therefore lack the easy integration.

Now the relation between FREME and other NLP tools is explored: There is a series of ready to use tools for certain NLP tasks, e.g. the widely used Stanford CoreNLP (Manning et al., 2014), Apache OpenNLP (The Apache Foun-

[1] https://old.datahub.io/dataset?tags=freme-project

dation, 2017) or the Extended Named Entity Recognition API (Nguyen et al., 2017). The OpenMT toolkit (Klein et al., 2017) provides Machine Translation that has a similar focus on extensibility and modularity like FREME. These tools have a narrower focus than the FREME framework and can be integrated into FREME as e-Services. In this way only user level 0 is concerned with the inner workings and data formats of the tool. Subsequent user levels can utilize these tools like any other FREME e-Service and benefit from the FREME advantages.

Another background of FREME is research in interface engineering. Other research has identified three stakeholders affected by APIs and frameworks: API designers who implement the APIs. API users that implement applications using the work of the API designers and finally consumers that use the applications (Myers and Stylos, 2016). Research about API design argues that interoperability, reusability and interoperability are core features of APIs, along with other features such as learnability, security and expressiveness (Myers, 2017).

6. The FREME ecosystem

The FREME project was a two year innovation action funded by the Horizon 2020 program that ran from January 2015 - January 2017 under the lead of the German Research Centre for Artificial Intelligence. It consisted of a consortium of three research institutes and four industry partners. The goal was, among others, to help the partners bring multilingual, semantic technologies to the market. In the course of the FREME project the FREME framework described in this paper was developed. FREMEs business partners came from different domains of digital publishing, content recommendation, localization and internationalization and agriculture and food data.

6.1. The FREME community

The usefulness and the open nature of the framework attracted other users from outside of the consortium. The project Digital Curation Technologies (DKT) builds on the FREME framework also and adds new NLP services to the framework. Further the ADAPT research centre hosts a FREME server to provide a ready to use NLP API for its data scientists.

7. FREME in a real world scenario

The development of FREME was triggered by a number of use cases provided by the industry partners of the project. This section explains one of the use cases and the experiences made by this effort.

The use case was using FREME as part of the backend of a content recommendation platform. A backend service crawled websites and fed the documents into a FREME pipeline. Afterwards the features generated by the pipeline were used to recommend websites to the reader. The workflow used a clear separation of work between an application developer (user level 2) who developed the frontend and the data scientists (user level 1) who created the pipeline. The development of the backend from the data science side used rapid prototyping and different approaches were tried and compared to each other. During rapid prototyping the high level services proved to be useful because it was easy and straight forward to change the NLP pipeline. Further the format coverage aspect of FREME proved to be useful because the HTML documents could be fed into the pipeline without time consuming preprocessing. At the end of the pipeline the SPARQL filter service converted the output from NIF to easy processable tabular data which contains only the important information from the NLP pipeline, stripping all unnecessary information.

During this project user level 0 created the underlying FREME services which are independent from the specific use case. User level 1 configured the services and provided the necessary knowledge sources. User level 2 could use the services without the need of a deep understanding of the underlying technology. User level 3 used the content recommendation without even noticing the FREME pipelines working in the backend. The user levels could not be totally separated. User level 0 and 1 often where the same people. Also user level 2 wanted to learn as much as possible about the underlying technology so he could be able to perform certain configuration tasks of the pipeline himself.

In this use case the differentiation between user levels and the impact of this differentiation on the design of FREME proved to be very useful because it established a clear workflow and every user level could focus on what he or she can do best. Especially the flexibility to quickly try out new approaches, mostly by configuration and without coding, was very useful.

8. Conclusion and Future Work

The aforementioned concepts proved to be useful to deploy applications that use both LD and LT in industry use cases. The FREME framework provides easy to use, reusable and interoperable services with a special focus on bridging the knowledge gaps between application developers and technology experts. The high level pipeline components made integration easy and flexible. Currently the work on the FREME framework focuses on the creation of new services and maintaining the core of the framework so it stays up to date. There are plans to augment FREME with big data processing capabilities. Other approaches focus on integrating FREME with cloud infrastructure providers like Amazon and executing FREME in a lambda function to create services that are scalable on demand.

Acknowledgments. This research was partially supported by the German Federal Ministry of Education and Research (BMBF) through the project BBDC (01IS14013E).

9. Bibliographical References

Bachmann-Gmur, R. (2013). *Instant Apache Stanbol*. Packt Publishing.

Batista, F., Curto, P., Trancoso, I., Abad, A., Ferreira, J., Ribeiro, E., Moniz, H., de Matos, D. M., and Ribeiro, R. (2016). {SPA:} Web-based Platform for easy Access to Speech Processing Modules. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation {LREC} 2016, Portoro{ž}, Slovenia, May 23-28, 2016.*

Brümmer, M., Dojchinovski, M., and Hellmann, S. (2016). DBpedia Abstracts: A Large-Scale, Open, Multilingual

NLP Training Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Clarke, J., Srikumar, V., Sammons, M., and Roth, D. (2012). An NLP Curator (or: How I Learned to Stop Worrying and Love NLP Pipelines). In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Dojchinovski, M., Sasaki, F., Gornostaja, T., Hellmann, S., Mannens, E., Salliau, F., Osella, M., Ritchie, P., Stoitsis, G., Koidl, K., Ackermann, M., and Chakraborty, N. (2016). FREME: Multilingual Semantic Enrichment with Linked Data and Language Technologies. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4180–4183.

Ehrmann, M., Nouvel, D., and Rosset, S. (2016). Named Entity Resources - Overview and Outlook. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Ferrucci, D., Lally, A., Verspoor, K., and Nyberg, E. (2009). Unstructured Information Management Architecture ({UIMA}) Version 1.0. OASIS Standard.

Ferrucci, D. A. (2012). Introduction to "This is Watson". *IBM J. Res. Dev.*, 56(3):235–249.

Flati, T., Vannella, D., Pasini, T., and Navigli, R., (2014). *Two is bigger (and better) than one: The wikipedia bitaxonomy project*, volume 1, pages 945–955. Association for Computational Linguistics (ACL).

Hellmann, S., Lehmann, J., Auer, S., and Bruemmer, M. (2013). Integrating NLP using Linked Data. In *The 12th International Semantic Web Conference (ISWC2013)*, pages 97–112.

Hinrichs, E. and Krauwer, S. (2014). The {CLARIN} {Research} {Infrastructure}: {Resources} and {Tools} for e-{Humanities} {Scholars}. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 1525–1531.

Hinrichs, E., Hinrichs, M., and Zastrow, T. (2010). WebLicht: Web-based LRT Services for German. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, pages 25–29, Stroudsburg, PA, USA. Association for Computational Linguistics.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *CoRR*, abs/1701.0.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The {Stanford} {CoreNLP} Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mitchell, T. (2005). Reading the Web: A Breakthrough Goal for AI. \url{http://www.cs.cmu.edu/\~{}tom/pubs/AImagazine-7-2005.html}.

Myers, B. A. and Stylos, J. (2016). Improving API Usability. *Commun. ACM*, 59(6):62–69.

Myers, B. A. (2017). Human-centered Methods for Improving API Usability. In *Proceedings of the 1st International Workshop on API Usage and Evolution*, WAPI '17, page 2, Piscataway, NJ, USA. IEEE Press.

Navigli, R. and Ponzetto, S. P. (2012). {B}abel{N}et: {T}he Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, 193:217–250.

Nguyen, T. D., Mai, K., Pham, T.-H., Nguyen, M. T., Nguyen, T.-V. T., Eguchi, T., Sasano, R., and Sekine, S. (2017). Extended Named Entity Recognition {API} and Its Applications in Language Education. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, {ACL} 2017, Vancouver, Canada, July 30 - August 4, System Demonstrations*, pages 37–42.

Noji, H. and Miyao, Y. (2016). Jigg: {A} Framework for an Easy Natural Language Processing Pipeline. In Sameer Pradhan et al., editors, *Proceedings of {ACL-2016} System Demonstrations, Berlin, Germany, August 7-12, 2016*, pages 103–108. Association for Computational Linguistics.

Paul Buitelaar, Mihael Arcan, C. A. I. J. F. S.-R. and Strapparava, C. (2013). Linguistic linked data for sentiment analysis. In *2nd Workshop on Linked Data in Linguistics (LDL- 2013): Representing and linking lexicons, terminologies and other language data*, Pisa, Italy.

Rehm, G. and Sasaki, F. (2016). Digital Curation Technologies. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation (EAMT˜2016)*, Riga, Latvia.

Rehm, G., Moreno Schneider, J., peter Bourgonje, Srivastava, A., Nehring, J., Berger, A., König, L., Räuchle, S., and Gerth, J. (2017). Event Detection and Semantic Storytelling: Generating a Travelogue from a large Collection of Personal Letters. In *Proceedings of the Events and Stories in the News Workshop*, pages 42–51, Vancouver, Canada, aug. Association for Computational Linguistics.

Ristoski, P. and Paulheim, H. (2016). Semantic Web in data mining and knowledge discovery: A comprehensive survey. *Web Semantics: Science, Services and Agents on the World Wide Web*, 36:1–22.

Sanderson, R., Ciccarese, P., and de Sompel, H. V. (2013). Designing the W3C open annotation data model. In Hugh C Davis, et al., editors, *WebSci*, pages 366–375. ACM.

Sasaki, F., Gornostay, T., Dojchinovski, M., Osella, M., Mannens, E., Stoitsis, G., Ritchie, P., and Koidl, K. (2015a). Introducing FREME: Deploying Linguistic Linked Data. In *The 4th Workshop on the Multilingual Semantic Web: co-located with ESWC 2015, Portoro{ž}, Slovenia*.

Sasaki, F., Gornostay, T., Dojchinovski, M., Osella, M., Mannens, E., Stoitsis, G., Ritchie, P., and Koidl, K. (2015b). Introduction to FREME: Data meets Language meets Business. In *The 1st International Workshop on the Use of Multilingual Language Resources in Knowledge Representation Systems (MLKREP2015), Vienna, Austria*.

Sasaki, F., Dojchinovski, M., and Nehring, J. (2016). Chainable and Extendable Knowledge Integration Web Services. In *First Workshop on Knowledge Extraction and Knowledge Integration - ISWC 2016*.

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. *Official Google Blog, May*.

Srivastava, A., Rehm, G., and Sasaki, F. (2017). Improving Machine Translation through Linked Data. *The Prague Bulletin of Mathematical Linguistics*, 108(1):355–366, jan.

The Apache Foundation. (2017). Apache OpenNLP.

The Apache UIMA Development Community. (2008). UIMA Simple Server User Guide.

Usbeck, R., Ngomo, A.-C. N., Röder, M., Gerber, D., Coelho, S., Auer, S., and Both, A. (2014). AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data. In Peter Mika, et al., editors, *The Semantic Web – ISWC 2014*, volume 8796 of *Lecture Notes in Computer Science*, pages 457–471. Springer International Publishing.

10. Language Resource References

Bennett, R., Hengel-Dittrich, C., O'Neill, E. T., and Tillett, B. B. (2006). VIAF (Virtual International Authority File): Linking Die Deutsche Bibliothek and Library of Congress Name Authority Files.

Brümmer, M., Dojchinovski, M., and Hellmann, S. (2016). DBpedia Abstracts: A Large-Scale, Open, Multilingual NLP Training Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Haak, Laurel L. and Fenner, Martin and Paglione, Laura and Pentz, Ed and Ratner, H. (2012). ORCID: a system to uniquely identify researchers. *Learned Publishing*, 25(4).

Kim, S., Iglesias-Sucasas, M., and Viollier, V. (2013). The FAO Geopolitical Ontology: A Reference for Country-Based Information. *Journal of Agricultural & Food Information*, 14(1):50–65.

The LREC Workshops Map

Roberto Bartolini, Sara Goggi, Monica Monachini, Gabriella Pardelli
Istituto di Linguistica Computazionale del CNR "Antonio Zampolli"
Via Giuseppe Moruzzi 1 – 56124 Pisa, Italy
{roberto.bartolini, sara.goggi, monica.monachini, gabriella pardelli}@ilc.cnr.it

Abstract

The aim of this work is to present an overview of the research presented at the LREC workshops over the years 1998-2016 with the aim to shed light on the community represented by workshop participants in terms of country of origin, type of affiliation, gender.
There has been also an effort towards the identification of the major topics dealt with as well as of the terminological variations noticed in this time span. Data has been retrieved from the portal of the *European Language Resources Association* (ELRA) which organizes the conference and the resulting corpus made up of workshops titles and of the related presentations has then been processed using a term extraction tool developed at ILC-CNR.

Keywords: corpus creation, terminology, LREC

1. Introduction

Over the years the increasing availability of online open access documentation has permitted to gather information for various types of analysis. For this work we decided to retrieve the information about all workshops organized as satellite events of the *International Conference on Language Resources and Evaluation* (LREC) in its ten editions (1998-2016): data has been retrieved from the portal of the *European Language Resources Association* (ELRA) which organizes the conference.
The aim is to monitor the research in the field of Language Resources (LRs) and Language Technology (LT) presented at LREC workshops and describe the thematic and terminological trend over these 18 years. In the introductory pages to the Proceedings of LREC 2000, Antonio Zampolli wrote: "The workshops, whose Proceedings are published in separate volumes, have contributed greatly to the scientific relevance of the Conference, both for the quality of the work and the choice of topics." (Zampolli, 2000).
During the ten editions of the conference its satellite workshops have witnessed the participation of 6039 researchers from both academy and industry, coming from all continents. This heterogeneity results in a authorship which is composite for country, affiliation, sector and also gender; as for the latter, the female representation is relevant, as well as the percentage of single authors.

2. Some Remarks on Topics

The topics dealt with and the papers related to these topics are the focus of the work: we created a corpus (WS-CORPUS) made of the titles of all the workshops and their related papers presented at LREC since 1998. In ten editions of the conference 206 workshops have been organized around the most varied subjects, some of them as single events but many ended up constituting a "series" over the years.

The workshop on Sign Language, for example, has been organized at LREC since 2004 by the same researchers; while a workshop on the so-called "minority/less-resourced/under-resourced" languages[1] is always present at LREC but the organizing teams have been different over this time span: in the first Granada edition a workshop was organized by researchers working on the development of resources for the indigenous minority languages of Europe ("Language Resources for European Minority Languages"); since then Workshops on this topic have always been organized at LREC but it is worth mentioning – and investigating - the terminological shift from the term "minority languages" used in 1998, 2000 and 2004 to "less-resourced languages" in the three following editions and finally to the term "under-resourced language" which establishes itself in 2014 and is used again in 2016. Research in this field range from the creation of LRs (in the first two editions but again in 2010), to the development of the needed language technologies up to machine translation (in two editions) for these specific languages.
In Granada was also organized a workshop titled "Speech Database Development for Central and Eastern European Languages" with the aim of promoting speech technologies in Eastern Europe.
This theme will appear again a decade later at LREC 2010 where a workshop titled «Exploitation of Multilingual Resources and Tools for Central and (South) Eastern European Languages» was organized: to be noted the additional geographical specification (*South*) for a better definition of the linguistic area and the substitution of the outdated term *database* with *resources*.
«The same holds for well established or emerging linguistic knowledge representation frameworks, which can only benefit from embedding components for Central, Eastern and Southern European languages» (Stelios Piperidis et alii, 2010).

[1] From the English Oxford Living Dictionaries: "A language spoken by a minority group, if different from that of the majority. Origin 1920s; earliest use found in American Journal of International Law".
https://en.oxforddictionaries.com/definition/minority_language

Still focusing on languages, at LREC 2012 in Istanbul the first workshop on Indian languages is organized («WILDRE –the first 'Workshop on Indian Language Data: Resources and Evaluation»); and WILDRE will take place in the next two editions as well, in Reykjavik and Portoroz.
A special focus on Arabic and more generally on Semitic languages dates back to the 2002 edition in Las Palmas and covers most of following editions where also two tutorials took place (in 2006 and 2008).
Specifically for Arabic, the last two LREC editions the same group of researchers organized the OSACT workshop («Open/Free–Source Arabic Corpora and Processing Tools»), adding the social media theme to the 2016 edition.
Invitations for submission on the topic related to speech technology and African languages converge in the «Second Workshop on African Language Technology (AfLaT)» presented at LREC 2010 in Malta while already in 2006 a first call had been launched for developing a network aimed at cooperation in the development of resources and tools for the African languages («Networking the development of language resources for African languages»).

3. WS-CORPUS Building and Method

The process of corpus building and analysis is split up in three steps: 1) creation of the corpus by acquiring the titles of all the LREC satellite workshops and their related papers; 2) data cleaning and processing using a NLP tool; 3) terminological analysis and comparison.
Data has been retrieved from the ELRA-ELDA portal where the Proceedings of all the LREC editions - except for the 1998 one – are stored (http://www.elra.info/en/lrec/proceedings/).
The workshops organized over the 18 years are 206 (while tutorials are 46), for an overall total of 2250 presentation titles which build up our small WS-CORPUS.
In the pre-processing phase data have been annotated for having detailed information on the authors' profile: a) year of the workshop, b) title of the workshop, c) title of the paper, d) author(s)'s name and surname, e) author(s) gender, f) author(s)'s affiliation, g) country, h) ISO code.

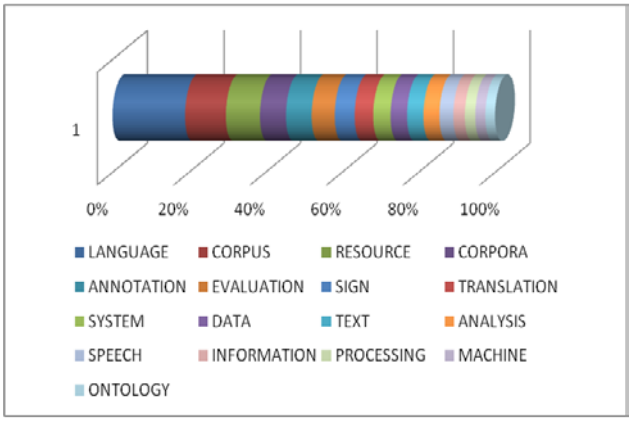
The corpus was then processed using a tool for term extraction (Goggi et al. 2015; 2016): this is a "pipeline" of different tools which extracts lexical knowledge from texts; in short, a rule-based system tool for knowledge extraction and document indexing.
The tool analyzes textual data and its result is an annotated text that allows for terminological extraction of relevant concepts. Within the WS-CORPUS it extracts a list of single (monograms) and multi-word terms (bigrams and trigrams) ordered by frequency with respect to the context.

3.1 Terminological Analysis

In order to correctly comment on the information retrieved from the corpus we should extract the most connotative features of the domain from a terminological point of view; this analysis will allow us to monitor the thematic trends as they appear and disappear from the LREC scene.
The connotative strength of words is not the same for all of them and their frequency is not the only factor to be taken into account for weighting the importance of the single terms in our corpus: as a matter of fact, also terms with just one occurrence within the corpus (hapax) might have a significant value once properly contextualized.

3.2 Terms Frequency

We start with the analysis of the most frequent monograms occurring in the corpus which will give a first terminological overview: starting from the most obvious ones such as *language*, *corpus*, *resource*, *corpora*, *annotation* and following with *evaluation*, *sign*, *translation*, *system*, *data*, *text*, *analysis*; terms with less than a hundred occurrences are *speech*, *information*, *processing*, *machine*, *ontology*. Apart from the general terms like *language*, *resource*, *data* and *text*, some domain-related terms can be retrieved among the ten most frequent ones: *corpus*, *corpora*, *ontology*, *system*, *machine*, *processing*, *annotation*, *evaluation*, *translation*.



Graph 1: Terms Frequency

3.2.1 Languages

The high frequency of the term "language" – rather obvious in itself – is reflected in the total number of spoken languages dealt with in the 200 LREC workshops: there are about sixty languages mentioned in the corpus, some of them with just one occurrence (hapax): <ALGERIAN ARABIC>, <ALSATIAN>, <ARANESE>, <ASTURIAN>, <BASHKIR>, <BELGIAN>, <BENGALI>, <BRETON>, <CREOLE>, <CROATIAN>, <IKOTA>, <JAPANESE> ,<KOMI>, <MAITHILI>, <MALAY> <MARTHI>, <NENET>, <NEPALI>, <OCCITAN>, <SAMI>, <SOMALI>, <SWAHILI>, <TAJIK>, etc.. These hapax might be substantially important because are the expression of an interest in developing LRs, corpora, tools, standards and infrastructures also for the less-studied languages cited above.
Research on a given language could be influenced by the venue where a conference takes place: the most significant example is the edition of Marrakech 2008 where the term "Arabic" reaches the highest number of

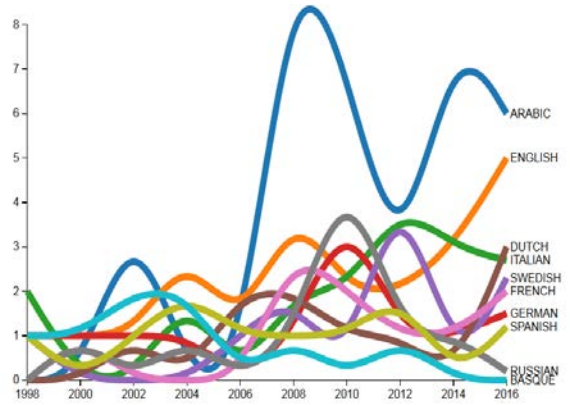
occurrences thanks to the organization of two events (a workshop and a tutorial) on the processing of dialects and local languages. And a workshop on Turkic languages has been organized only in Istanbul in 2012. But that this does not apply to all languages given that, for example, the workshop on Indian languages, firstly organized in Istanbul in 2012 has then been presented again both in Iceland and Slovenia proving then to be unrelated to the geographical location of the conference.
Graph 2 represents the temporal trend of the first ten languages retrieved from the corpus.

3.2.2 Sign Language

The survey on languages includes Sign Language (SL) as well: research on SL has been regularly presented at LREC since 2004 thus becoming a feature of the conference and constituting a "series" of seven consecutive editions. Within our corpus we can identify research on sixteen European and non-European SLs: ARABIC SIGN LANGUAGE - ArSL (2010); AMERICAN SIGN LANGUAGE - ASL (2004, 2010, 2012, 2014); BRITISH SIGN LANGUAGE (2010); CHINESE SIGN LANGUAGE - CSL (2004, 2010); FINNISH SIGN LANGUAGE - FinSS (2010, 2014, 2016); FRENCH SIGN LANGUAGE - LSF (2006, 2014,2016); GERMAN SIGN LANGUAGE - DGS (2010, 2012); GREEK SIGN LANGUAGE- GSL (2004); ITALIAN SIGN LANGUAGE- LIS (2004, 2006, 2010, 2014); HONG KONG SIGN LANGUAGE - HKSL (2008); SLOVENIAN SIGN LANGUAGE - (2016); SWEDISH SIGN LANGUAGE - STS (2012, 2014, 2016); TURKISH SIGN LANGUAGE - TİD (2014); SPANISH SIGN - LSE 2004, (2010); CZECH SIGN LANGUAGE - CzSL (2010); RUSSIAN SIGN LANGUAGE - RSL (2010).

3.2.3 Minority, Less-resourced and Under-resourced Languages

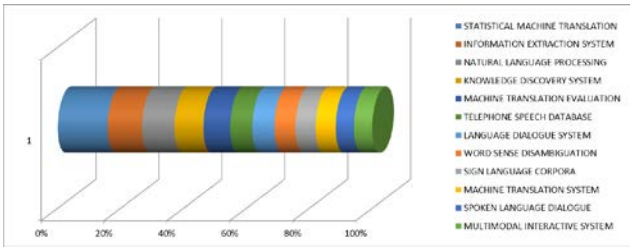
Workshops on this topic have been organized at LREC since the 1998 edition but what is worth mentioning – and investigating - is the terminological shift from the term "minority languages" used in 1998, 2000 and 2004 to "less-resourced languages" in the three following editions and finally to the term "under-resourced language" which establishes itself in 2014 and is used again in 2016.
In 2002 the focus of the workshop titled *Portability Issues in Human Language Technology (HLT*) is slightly different: «The primary objective of the workshop is to bring together participants from academia and industry to discuss and disseminate the current state of the art in multilingual research and development in the context of cross-language HLT transfer» .[2]
On the workshop website, organizers further specified the motivation behind the event: «There are more than 6000 languages in the world, yet only a small number possess the resources required for implementation of Human Language Technologies (HLT). This imbalance in technical resources available to languages of the world is likely to result in a significant linguistic divide that further exacerbates global social and economic inequities unless decisive action is taken relatively soon. One potential means of ameliorating this imbalance in technology resources is through encouraging research in the portability of human language technology for multilingual application».
Research in this field range from the creation of LRs (in the first two editions but again in 2010), to the development of the needed language technologies up to machine translation (in two editions) for these specific languages.



Graph 2: Languages

3.2.4 Trigrams

The analysis of the most frequent trigrams extracted from the corpus provides a list of the recurring topics: from statistical machine translation to multimodal systems, from machine translation evaluation to information extraction systems and so on.
Graph 3 shows the trigrams which identify the most investigated domains.



Graph 3: Trigrams

4. A case-study: WordNet

«WordNet® is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The resulting network of meaningfully related words and concepts can be navigated with the browser (link is external). WordNet is also freely and publicly available for download. WordNet's structure makes it a useful tool for computational linguistics and natural language processing»: this is the definiton of the resource available on the Princeton University website[3].

[2] http://www.lrec-conf.org/lrec2002/lrec/wksh/Portability.html

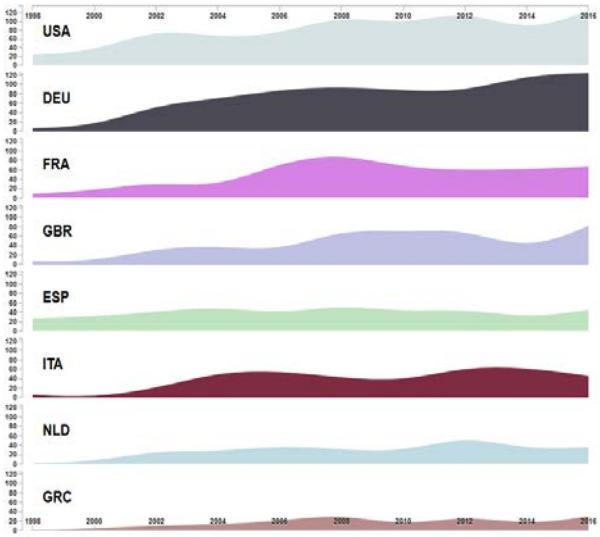
[3] https://wordnet.princeton.edu/

The WordNet resource is one of the most widely used all over the world since the early 90s and many related projects and extensions have been developed over the years: just to mention a few, in addition to the many WordNets in languages other than English, there is «SentiWordNet», a lexical resource for opinion mining; «BabelNet», a very large multilingual semantic network with millions of concepts; «FrameNet», a lexical database similar and referring to WordNet; the Lexical markup framework (LMF) which is an ISO standard for defining a common standardized framework for the construction of lexicons.
Unexpectedly just one workshop on WordNet has been organized at LREC, precisely at the 2002 edition: *Wordnet Structures and Standardization, and how these affect Wordnet Applications and Evaluation*.
In the Call for Papers of the workshop, the organizers state the importance of WordNet as follows: «During the last decade, WordNet has become a powerful resource in (computational) linguistics for various language processing tasks as well as for theoretical research issues. Due to the success of Princeton WordNet, numerous wordnets for further languages have been built or just been started. Therefore, guidelines and principles for comparing existing and acquiring new languages are of utmost importance in the field. Specific lexical properties of new languages should be accounted for. Wordnet developers of less-studied languages can profit from the experience made by the wordnet pioneers, and may also benefit from the feedback provided by wordnet appliers. The workshop will constitute a forum for sharing a common wordnet structure across languages».[4]
Papers on WordNet have obviously been presented at other Workshops during these ten editions and are spread over topics such as lexical semantics, computational lexicography, ontology, information extraction, machine translation, sentiment, annotation, semantic web, linked data, etc. We took note of a total of 38 papers related to WordNet presented at 21 workshops.
From the works presented it is possible to retrieve WordNets for the following languages: *Catalan WordNet, Galician WordNet* (1998); *Hungarian WordNet – Balkanet, Romanian WordNet, Estonian WordNet and GermaNet* (2002); *Czech WordNet - Prague Dependency TreeBank* (2004); *French WordNet* (2008); *Arabic WordNet – YAGO ontology, Estonian WordNet* (2010); *Slovene WordNet, Croatian WordNet* (2012); *IndoWordNet - Indian languages from Indo-Aryan, Dravidian, Quranic Arabic WordNet, Irish language - WordNet Gaeilge* (2014); *Konkani SentiWordNet* (2016).

5. Workshops Community

5.1 Community and Country

This paragraph is dedicated to give a first idea about the profile of LREC Workshops authors.
The inter-disciplinary dimension, the specialized themes and the geographical dislocation of its stakeholders are the requisites of attraction of the satellite LREC workshops community: over the years universities, research centers, governmental bodies and industries presented their own research experiences, the technological solutions tested and/or adopted thus facilitating the introduction of new paradigms as well as the giving up of obsolete models.
During these 18 years, 6039 researchers «participated» in the LREC workshops: 4116 from Universities, 402 from industries, 159 from Academies of Science and the rest from National Reserch Bodies (as the French CNRS or the Italian CNR), private research centers, foundations, governmental institutions, national or international organizations from all over the world. Coming to the geographical representativeness, Graph 4 illustrates the participation of the first ten countries of the ranking.
It must be said that «participation» does not necessarily means «attendance»; that is, we have records from the workshop proceedings and not from the list of actual participants to the event. There will surely be a discrepancy between these two figures, meaning that real attendance is much lower than the figures about authors coming from the published papers might suppose.
On the other hand, the following are the countries represented only once: Argentina (2012); Chile (2016); Cuba (2008); Cyprus (2012); Ethiopia (2012); Latvia (2012); Pakistan (2008); Thailand (2000); Vietnam (2012); Zambia (2006). Two countries record two occurrences (that is, two authors): Macao in 2004 and Colombia in 2016.
As stated above, the presence of two authors from these countries does not automatically means that they actually attended the workshops.



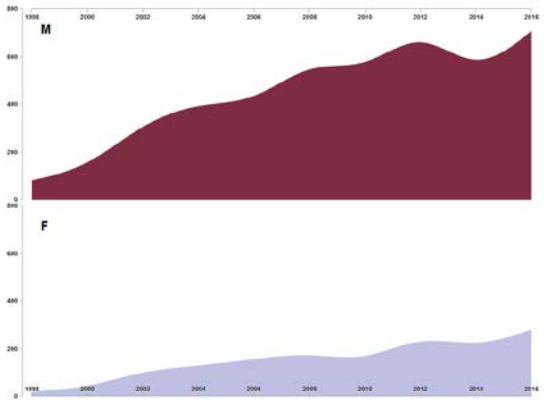
Graph 4: Countries

5.2 Community and Gender

In addition to the nationality of authors, we decided to extrapolate the information on their gender as well: this type of analysis is usually difficult due to the various ways of writing the names (full name, initials, middle initials). It was therefore needed a cleaning process for

[4] http://www.lrec-conf.org/lrec2002/lrec/wksh/Wordnets.html

being able to divide the authors by gender: for disambiguating the initials of the first names we used portals such as ACL Anthology, LREC Proceedings – Section author, in alphabetical order for all editions), Scopus, ISI WOS, Google Scholar Citations and social networks such as LinkedIn.
A few unresolved cases have been annotated with a question mark. Graph 4 shows the participation by gender to LREC Workshops: of course the names of those who participated to more than one edition – or even presented more than one paper at the same edition - have been counted only once. The results talk about a sort of clear preponderance of men: 1528 female participants vs 4435 male (76 are the unidentified authors).



Graph 5: Gender

6. Conclusions

This preliminary analysis of the WS-CORPUS gives a partial terminological overview of the research presented at LREC workshops; further investigations will have to be performed in order to being able to provide a diachronic view on the evolution of the topics treated, for highlighting the new ones which have emerged and those which rather disappeared from the scene.
In these 18 years many topics have been dealt with in these workshops: some major themes have been faced regularly over the conference editions, like for example the various aspects of multimodality (workshops at LREC have been organized since 2002), or issues related to standards and interoperability. Some specific topics such as emotion and sentiment analysis or biomedicine started being investigated at LREC 2006 and since then have been regularly presented; and themes such as application of visualization tools to LRs, controlled natural language applications and NLP applications to the Digital Humanities have recently emerged.
This study cannot be considered an exhaustive review of the field seen through the lenses of the LREC workshops because for the time being just a few features have been taken into account. More thorough analyses can be carried out both on the research side for detecting the thematic trends and on the "societal" side for better defining this heterogeneous community constituted by the scholars who participated to the numerous LREC workshops.

7. Bibliographical References

Benjamin M., Radetzky P. (2014). Multilingual Lexicography with a Focus on Less-Resourced Languages: Data Mining, Expert Input, Crowdsourcing, and Gamification. *Proceedings of CCURL 2014 Workshop: Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era - Ninth International Conference on Language Resources and Evaluation* (LREC'14). Reykjavik, Iceland. European Language Resource Association (ELRA), Paris. Pages 9-16.

Cunliffe D., Herring Susan C.(2005). Introduction to Minority Languages, Multimedia and the Web. *New Review of Hypermedia and Multimedia*, 11(2), 2005. Taylor & Francis. Pages 131-137.
doi = {10.1080/13614560512331392186}.

Francopoulo G., Mariani J., Paroubek. P. Vernier F. (2016). Providing and Analyzing NLP Terms for our Community, *Proceedings of the 5th International Workshop on Computational Terminology (Computerm2016)*, Osaka, Japan. The COLING 2016 Organizing Committee. Pages 94–103, http://www.aclweb.org/anthology/W16-4711

Goggi et al. (2015). Marine Planning and Service Platform (MAPS): An Advanced Research Engine for Grey Literature in Marine Science. *The Grey Journal* Volume 11: 3 (2015), ISSN: 1574-1796.
Also in D. Farace and J. Frantzen (Eds.), Proceeding of the *Sixteenth International Conference on Grey Literature Grey Literature Lobby: Engines and Requesters for Change, GL'16,* (Library of Congress Washington D.C., USA (GL-conference series, ISSN 1386-2316, Volume 16). TextRelease, Amsterdam, 2015. Pages 108-115.

Goggi S., Pardelli G., Bartolini R., Frontini F., Monachini M., Manzella G., De Mattei M. e Bustaffa F. (2016). A semantic engine for grey literature retrieval in the oceanography domain, *The Grey Journa*l, (12), 3, pp.155-161, ISSN: 1574-1796. Also in Dominic Farace, Jerry Frantzen (Eds.) (2016), Proceedings, Seventeenth International Conference on Grey Literature. A New Wave of Textual and Non-Textual Grey Literature, Amsterdam, TextRelease. Pages 104-111.

Johnston T. (2010). Adding value to, and extracting of value from, a signed language corpus through secondary processing: implications for annotation schemas and corpus creation. *Proceedings of the 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies. - Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA), Paris. Pages 137-142.

Mariani J., Paroubek, P., Francopoulo G., Hamon O. (2016). Rediscovering 15 + 2 years of Discoveries in Language Resources and Evaluation. *Language Resources and Evaluation*, 50(2):165–220.

Mapelli, V., Arranz, V., Carré, M., Mazo, H., Mostefa, D., and Choukri, K. (2012). ELRA in the heart of a cooperative HLT world. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation* (LREC'12). Istanbul, Turkey. European Language Resources Association (ELRA), Paris. Pages 55-59.

Nirenburg S., McShane M.(2009). Computational Field Semantics: Acquiring an Ontological-Semantic Lexicon for a New Language. In *Language Engineering for Lesser-Studied Languages*, Volume 21. Ioss Press. Pages 183-206.

Simpson H., Cieri C., Maeda K., Baker K., Onyshkevych B. (2008). Human Language Technology Resources for Less Commonly Taught Languages: Lessons Learned Toward Creation of Basic Language Resources. *Proceedings of the SALTMIL Workshop: Collaboration: interoperability between people in the creation of language resources for less-resourced languages - 6th International Conference on Language Resources and Evaluation* (LREC '08), Marrakech, Marocco. European Language Resource Association (ELRA), Paris. Pages 7-11.

Zampolli, A. (2000). Introduction of the Conference Chairman. In M. Gavrilidou, G. Carayannis, S. Markantonatou, S. Piperidis, G. Stainhaouer (Eds), *Proceedings of the 2nd International Conference on Language Resources and Evaluation* (LREC'00*)*. Volume I, Athens, Greece. European Language Resource Association (ELRA), Paris. Pages XV-XXI.

http://www.elra.info/en/lrec/proceedings/

http://www.lrec-conf.org/lrec1998/elra/workshops.html

https://wordnet.princeton.edu/

https://en.wikipedia.org/wiki/WordNet

Preserving Workflow Reproducibility: The RePlay-DH Client as a Tool for Process Documentation

Markus Gärtner[1*] Uli Hahn[2] Sibylle Hermann[3*]**
[1]Institute for Natural Language Processing [2]Communication and Information Centre
[3]University Library [*]University of Stuttgart [**]Ulm University
markus.gaertner@ims.uni-stuttgart.de, uli.hahn@uni-ulm.de
sibylle.hermann@ub.uni-stuttgart.de

Abstract

In this paper we present a software tool for elicitation and management of process metadata. It follows our previously published design idea of an assistant for researchers that aims at minimizing the additional effort required for producing a sustainable workflow documentation. With the ever-growing number of linguistic resources available, it also becomes increasingly important to provide proper documentation to make them comparable and to allow meaningful evaluations for specific use cases. The often prevailing practice of post hoc documentation of resource generation or research processes bears the risk of information loss. Not only does detailed documentation of a process aid in achieving reproducibility, it also increases usefulness of the documented work for others as a cornerstone of good scientific practice. Time pressure together with the lack of simple documentation methods leads to workflow documentation in practice being an arduous and often neglected task. Our tool ensures a clean documentation for common workflows in natural language processing and digital humanities. Additionally, it can easily be integrated into existing institutional infrastructures.

Keywords: reproducibility, metadata, documentation

1. Introduction

Several decades ago Claerbout and Karrenbach (1992) formulated the idea that in a research environment of ever increasing digitization the published articles are only advertisement for the actual scholarship. The scholarship itself then also includes all the scripts, tools, data and additional information that was used to create the results. This view is even more pronounced today, considering how already minor adjustments or changes in intermediary or preprocessing steps in typical natural language processing (NLP) tasks can have significant impact on the outcome of down-stream tasks, as has been exemplified by Elming et al. (2013). While it is not an ultimate guarantee for reproducibility, thorough documentation of workflows certainly aids in being able to reproduce results, evaluate the suitability of resources for a certain task or at least increases ite)usability for others and even for oneself.

In stark contrast to this, day-to-day research often exhibits a serious neglect of documentation efforts when it comes to minor details in a project's workflow. Reasons for this include for example the competitive and time pressure that largely dominates today's research practice. On the road to a publishable end-result it is easy to omit workflow documentation, especially if it requires a substantial amount of extra effort and there is no apparent gratification for it. If at all, such documentation is typically created retrospectively at a later time, where the risk that some details of the workflow might have already been lost is high.

The matter is further complicated by the diverse nature of workflows in the field of NLP, computational linguistics (CL) or digital humanities (DH). They draw from a vast pool of available resources and tools to intermix strictly automatized steps with purely manual work or any form of hybrids between those two. Resulting workflows can also be linear, branched or highly iterative with only small differences between recurring steps. They can further involve multiple persons working in an collaborative effort on the same data and joining results. Looking at this complexity one cannot emphasize enough the importance of providing detailed workflow and provenance information for published resources and results.

We previously proposed our design of a software tool (Gärtner et al., 2018) and an associated workflow metadata scheme to fill this need for a way of comfortably collecting process documentations. Our approach addresses documentation already during an active workflow top of the version control system Git[1]. It models workflows in a very generic way and is suitable to describe automatic steps as well as manual work. In order to increase sustainability our design uses several standards and established practices and offers simple interfaces for publishing and/or archiving individual stages within a workflow.

In this paper we focus on the applicability of our tool to typical tasks in NLP and CL. We show how common tasks fit into our metadata scheme and how the tool can be integrated into existing institutional infrastructures.

2. Related Work

For the task of documenting research processes we essentially distinguish two conceptually different types of systems, namely the ones used for workflow management and those for workflow tracking.

Used for setting up (and often even executing) workflows as collections of interdependent steps, *workflow management systems* (WMSs) contribute to the overall documentation effort prior to or during an active workflow. In contrast, a system for *workflow tracking* represents a more reactive approach and provides the documentation during or after a workflow. Looking at available WMS, the list of (commercial) systems for general-purpose or enterprise use is extensive. Their usability for specialized research

[1]https://git-scm.com/

workflows however is usually rather limited. In the context of certain research fields customized WMS instances have emerged, that allow researchers to build and execute workflows from catalogs of predefined analysis or processing steps. Popular web-based examples for this include *GenePattern*[2] (Reich et al., 2006) for genome research or in the field of NLP the *Language Application Grid* (Ide et al., 2016). The latter is an application of the *Galaxy* (Afgan et al., 2016) platform for biomedical analyses that has been tailored to the field of natural language processing. For local execution of NLP pipelines there have been approaches using the *Apache UIMA™project*, such as the *DKPro Core* (Eckart de Castilho and Gurevych, 2014) framework. With such systems the workflow documentation is very much covered by the actual description of the workflow setup if the system allows to export this kind of information.

When looking at the topic of workflow tracking, the core actions addressed often boil down to either monitor the data flow to/from processes or the physical changes to a designated set of resources. Tools like *YesWorkflow* (McPhillips et al., 2015) are examples of the first type and offer the ability to annotate data flows and operations on the code level. The implicit workflow can then be visualized based on the annotated information. The second type of workflow tracking systems are version control systems like Git[3] or Apache Subversion (SVN)[4], which are common in software development. They are intended for documenting complex collaborative development workflows exist in various flavors, such as centralized (SVN) versus decentralized (Git).

While the elaborate solutions listed above do provide a wide range of approaches to workflow documentation, they often are not applicable to a given research workflow (e.g. NLP/CL/DH): Their general focus on executable workflows makes them incompatible with workflows that also contain manual steps, such as annotation or curation tasks. Solutions such as version control provide the means of tracking very fine-grained changes, but leave out dedicated mechanism to formally describe what actions were conducted to cause those changes. Another important aspect of documentation systems regarding their usability is the level of technical expertise they require. This is especially true for technically less-skilled users, where a complex system is likely to act more as deterrent than encouragement. As a result it is not uncommon for researchers to document their workflows manually by means of a local Word or Excel file.

Besides approaches to documenting the actions in a workflow, it is also important to consider existing infrastructures that deal with metadata for the objects used in the workflow. As such initiatives like the *LRE Map* (Calzolari et al., 2010) or *CLARIN* (Hinrichs and Krauwer, 2014) and many others already provide wide coverage of metadata repositories for communities in computational linguistics and digital humanities. By no means a replacement of proper process documentation, linking to content of those infrastructures does already provide a valuable foundation to build on.

Also related to the topic of workflow documentation is the notion of resource provenance. Here the *PROV Family of Documents*[5] from the W3C Working Group provides models and exchange formats for describing provenance in a very expressive way. While not used as its native model for workflows, the RePlay-DH Client supports the PROV concept for the description of datasets that are exported.

Field	#	Description
Title	1	Short label of a step
Description	1	Human readable explanation of the actions performed
Person	0..n	Human subjects involved such as annotators or curators
Tool	0..1	Processing software used to generate the output
Input	0..n	All resources used to produce the output, including external things like annotation guidelines
Output	0..n	Resources generated or modified as result of the performed step
Properties	0..n	Custom metadata entries to store additional information in an organized (machine readable) way

Table 1: Overview of the top-level fields used in our process metadata scheme, their multiplicity and brief description.

3. Target Workflows

Workflows in RePlay-DH are assumed to be representable as directed acyclic graphs of individual steps and their dependencies. This is in line with observations regarding the most common workflow structures in Deelman et al. (2009). Our main targets are *data centered* workflows, hence the explicit `input` and `output` field in the metadata scheme described in the next section. However, it is noteworthy that the scheme is flexible enough to also model steps consisting of only title and description, e.g. ones that represent a cognitive progress in a workflow.

Under the data centered assumption, a single workflow step models the generation or modification of one or more resources by one or more actors[6], optionally with the aid of a software tool. The role of *actor* is implicitly assumed to be filled in by the user, but for situations with additional persons involved such a step can carry information to identify and describe those persons.

Naturally, the amount of individual actions that together form a complete workflow step is highly dependent on the actual workflow and we do not impose any direct constraints on the overall granularity of what can be recorded as a workflow step. We do however limit each step to contain at most one instance of processing software (see the `tool` field in Section 4). This is done to preserve proper information about individual input and output resources in the event of pipeline architectures where several tools are applied by the user consecutively.

[2] `http://www.genepattern.org`

[3] `https://git-scm.com/`

[4] `https://subversion.apache.org`

[5] `https://www.w3.org/TR/prov-overview/`

[6] We only count persons as actors in a workflow step, who perform manual work on the data. Merely executing an automatic processing tool or script does not suffice.

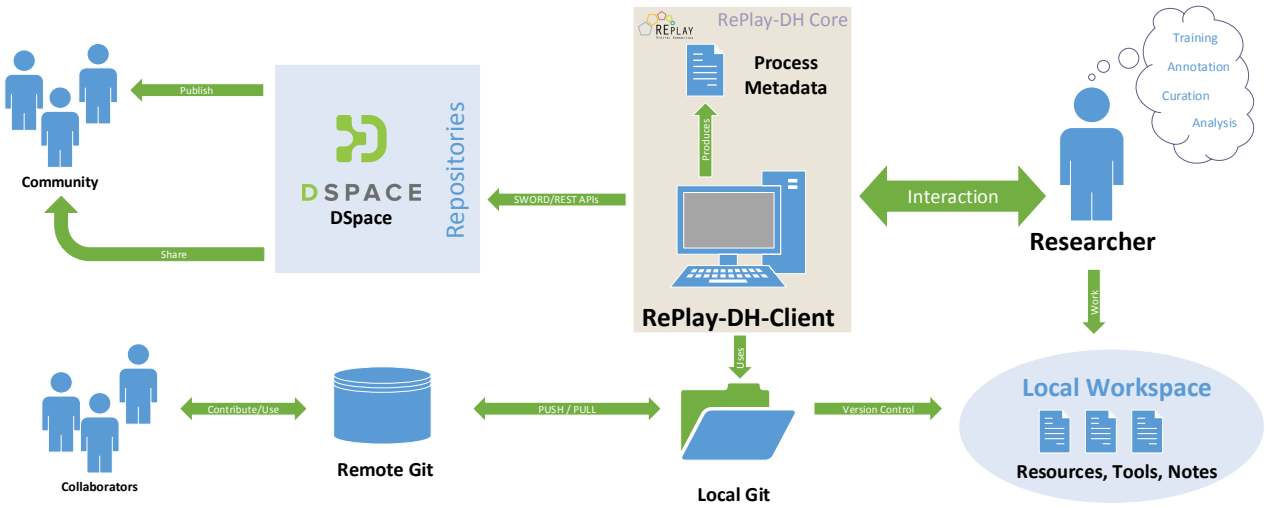


Figure 1: Overview of the RePlay-DH Client architecture and its interfaces to existing infrastructure components.

A recurring issue we noticed when documenting workflows involving very different processing tools, was the need to sometimes split resources into multiple fragments, to meet input requirements of certain tools. While not a problem for processing per se, it does however pose a challenge for workflow documentation. To avoid this, the RePlay-DH Client adds another organizational layer and lets the user group multiple physical resources (such as files) when creating process documentation, so that the amount of information within the metadata for a workflow step remains readily comprehensible.

4. Metadata

For the purpose of documenting workflows we distinguish between *process metadata* and *object metadata*. So far infrastructure initiatives in the NLP community have largely been focused on solutions for object related metadata, i.e. the detailed documentation of (finished) resources or tools in public repositories such as the CLARIN Virtual Language Observatory (VLO)[7] with its Component MetaData Infrastructure (CMDI)[8]. While some of those metadata schemes also provide the means to record information regarding the provenance or creation process of a resource, documenting that sort of workflow in detail is generally outside their scope.

Our approach is to model an entire workflow graph, including experimental steps or paths that led to dead-ends. It therefore features the individual workflow steps as central units and not only the one successful workflow path that resulted in the final output data. This is in contrast to classic approaches for documenting resource provenance such as PROV, which are focused on modeling provenance chains. We therefore designed a compact model and associated metadata scheme for recording process metadata in data centric workflows with the expressed goal of being able to build on the richness of existing object metadata infrastructures.

Table 1 gives a brief overview of the top-level fields for individual workflow steps in our metadata scheme. For a detailed explanation we refer to the original description in Gärtner et al. (2018). Of special importance in the context of this paper is the flexible mechanism used for identifying resources in our scheme and linking them to existing object metadata records. For this, every resource entry (`Person`, `Tool`, `Input`, `Output`) contains a collection of *typed identifiers*. An identifier consists of a *type* definition (similar to a namespace declaration) and the actual textual *id* itself, for instance `VLO-HANDLE` as type for entries in the VLO and the actual handle URL as id for a CMDI metadata record of the TIGER Corpus[9].

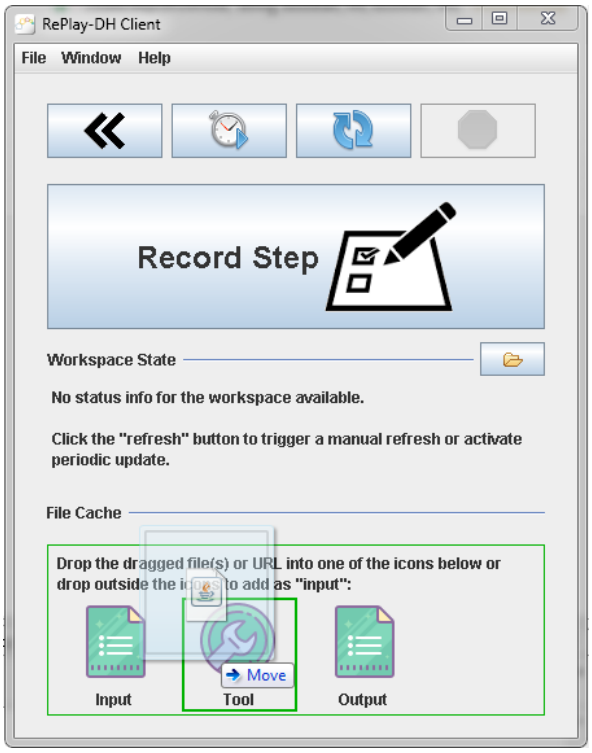
With this scheme our process metadata can link directly to object metadata entries in established repositories and reduce both redundancy and the overall effort required to describe resources in detail. For situations where no external repositories are available or needed, our client provides an integrated object metadata repository. This local repository can be used to store metadata records based on the Dublin Core scheme (Powell et al., 2005).

The RePlay-DH Client assists the user in the creation of metadata and uses JSON[10] for recorded workflow steps to serialize metadata into a representation that is stored alongside the data inside the underlying Git repository. We chose JSON for its simplicity, human readability (visible in Figure 3) and interoperability, which makes it easy to process exported instances of the process metadata with other tools or to extend the scheme for future needs.

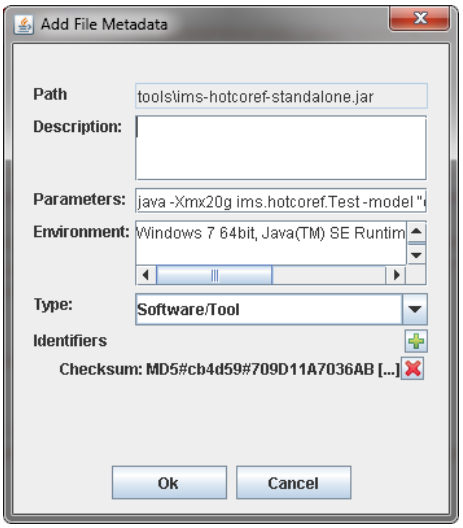
5. Architecture and Integration

The central product of our project is the RePlay-DH Client, the architecture and interfaces of which are depicted in Figure 1. Its main role is to bundle as many aspects of

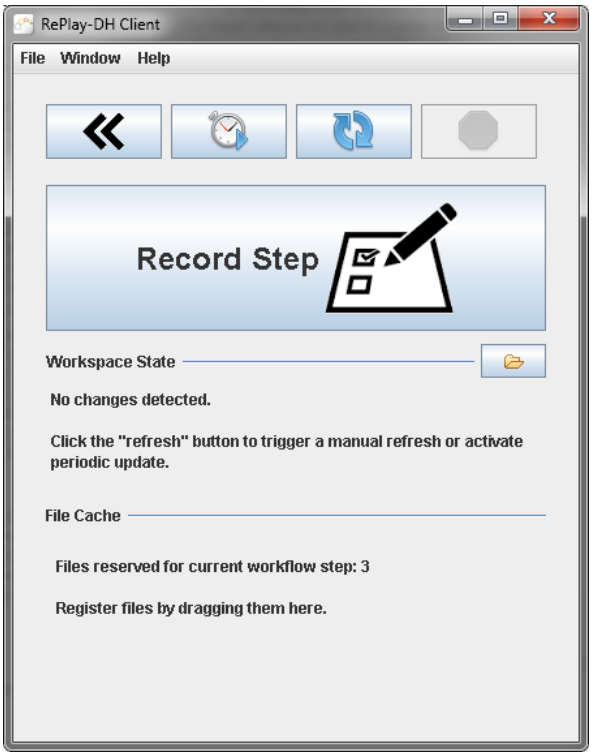
[7] `https://vlo.clarin.eu`
[8] `http://www.clarin.eu/cmdi`
[9] `http://hdl.handle.net/11022/1007-0000-0000-8E2D-F`
[10] `http://www.json.org/`



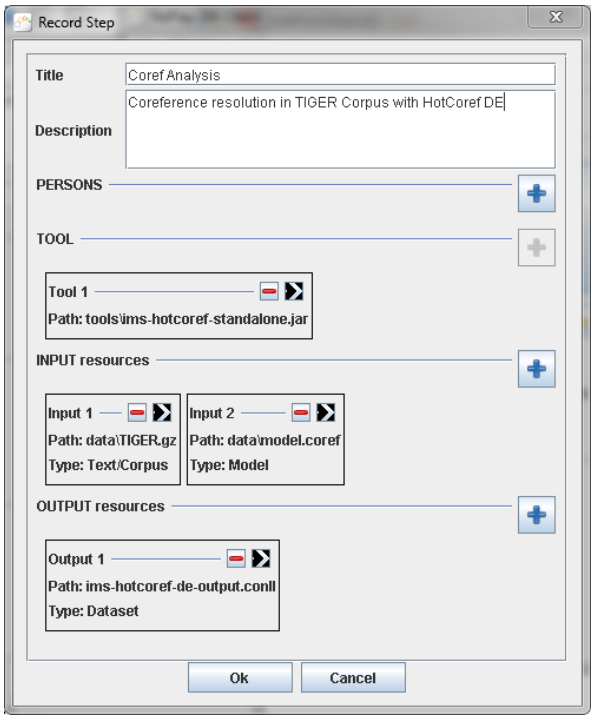
(a) Registering a resource as part of an ongoing workflow step via dragging a file onto the RePlay-DH Client interface.



(b) Recording metadata for the new resource.



(c) The RePlay-DH Client having three files registered as part of the current workflow step.



(d) Dialog for finally recording the active workflow step.

Figure 2: Phases in the process of registering resources (a) and their metadata (b) during a workflow step, marking them for later (c) and then finally recording the step itself (d).

```
{"title":"Coref Analysis",
  "description":"Coreference resolution on TIGER Corpus with HotCoref DE",
  "timestamp":"2017-12-27 17:13:12",
  "input":[
    {"systemId":"cbf13c01-d735-40e7-865c-477f4090e8c1",
    "type":"Text/Corpus",
    "identifiers":[
      {"type":"path","id":"data\\TIGER.gz"},
      {"type":"vlo-handle","id":"http://hdl.handle.net/11022/1007-0000-0000-8E2D-F"},
      {"type":"checksum","id":"MD5#7bca4b#452E036994C0F8E74D971D73861ADB2251"}]
    },
    {"systemId":"f73ad47c-8b4c-4afa-948a-dc5927a00261",,
    "type":"Model"
    "identifiers":[
      {"type":"path","id":"data\\model.coref"},
      {"type":"checksum","id":"MD5#1309#745412406E25E8B85EEFED6C58D1A8A574"}]
    }
  ],
  "tool":{
    "systemId":"53b0194f-d926-4cc1-b468-04feac22e404",
    "type":"Software/Tool",
    "environment":"Windows 7 64bit, Java(TM) SE Runtime Environment (build 1.8.0_161-b12)"
    "parameters":"java -Xmx20g -cp . ims.hotcoref.Test -model \"data/model.coref\" -in \"data/TIGER.gz\" -out \"
        ims-hotcoref-de-output.conll\" -cores \"4\" -lemmaBased -beam 20",
    "identifiers":[
      {"type":"checksum","id":"MD5#cb4d59#709D11A7036AB3778A3856E380564D1A6B"},
      {"type":"vlo-handle","id":"http://hdl.handle.net/11022/1007-0000-0000-8E69-B"},
      {"type":"path","id":"tools\\ims-hotcoref-standalone.jar"}]
  },
  "output":[
    {"systemId":"ff805df8-4e0e-43cf-8c25-d62cdfceeb45",
    "type":"Dataset",
    "identifiers":[
      {"type":"path","id":"ims-hotcoref-de-output.conll"},
      {"type":"checksum","id":"MD5#145e#EDC0555E82BDD9EF93E9A49CB0B52642ED"}]
    }
  ]
}
```


Figure 3: Example instance of process metadata for a single processing step in JSON. It describes an automatic analysis involving a coreference resolver as `tool`, a target corpus and trained model files as `input` and the result file as `output`. To conserve space, some values have been shortened and the default JSON layout has been condensed.

the workflow documentation process as possible in order to provide the researcher with a single optimized interface for that process. As mentioned in the introductory section, an omnipresent factor in today's research is time pressure. Therefore, a documentation tool's cost-benefit ratio (with 'cost' representing the effort required to use it) is most likely to be the decisive factor for its suitability. With this in mind our client is designed to be non-invasive and to minimize the documentation overhead, so that researchers can focus on their actual workflows. In its most basic configuration it functions without any external dependencies other than the libraries it ships with, making it very lightweight and flexible.

Git as foundation. Internally the client uses the popular and well-maintained JGit[11] library to put local workspace folders under version control and thereby monitor them. As a result no additional installation of any Git-related local software is required.

Each recording of a workflow step is wrapped into a Git commit and the associated process metadata for that step is stored in serialized form (JSON) as commit message. This provides a tight coupling of metadata and the observed physical changes of each workflow step within the Git commit graph.

As an added benefit of using a local Git repository comes also the possibility of connecting it to arbitrary remote repositories such as an institutional GitLab[12] instance. Use cases for this kind of interfacing are for example collaborative work or simply having an additional layer of backup available.

Incremental metadata construction. Delayed documentation, i.e. process metadata that gets created at a (significantly) later point in time than the actual actions it describes, runs the risk of having seemingly minor, but potentially important, steps omitted. To counter this, our client allows the user to build the final metadata for an active workflow step incrementally while working, "on the fly". Files (or URL strings) can be added via drag and drop functionality and immediately enriched with metadata. This process is depicted in the Figures 2a and 2b. The screenshots show the user dragging a tool in the form of a JAR file onto the client interface and then filling out the metadata form for the `tool` section of a workflow step (such as the parameters used to execute the tool on a command-line interface).

With this functionality the RePlay-DH Client enables users to document usage of a resource directly when actually using it. Entries cached this way (as seen in the lower part of Figure 2c) are then automatically added to the documentation of a workflow step when the user decides to record

[11] `https://eclipse.org/jgit/`

[12] `https://gitlab.com`

it. The final dialog in Figure 2d shows the three previously registered resources and the automatically detected output. Especially for automatic steps with a runtime of more than a few moments this can lead to a more efficient use of time for users and minimize the time overhead required for the documentation itself, besides the time that is used on the workflow.

Resource and metadata repositories. Our client is designed to directly interface with repository systems to make intermediate stages or the results of entire workflows available to others. Depending on the research context, different domains (Treloar et al., 2007) and restrictions apply. Innately the client works in private domains without external interfacing. In addition to that, the client also offers the possibility of collaborating in the shared (but not publicly open) domain. The possibility to share data within defined communities, is an important aspect when working with sensitive data.
For publishing partial or final results with a permanent identifier (DOI[13]) in the public domain the popular repository software DSpace[14] (Smith, 2002) is supported as one possible use case for the repository-client interaction.

Simplification and extensibility. The client implementation effectively shields the user from the complexity of underlying systems such as Git. This way we lower the barrier of entry to workflow documentation significantly and make it a convenient and accessible task for a wide target audience. The focus is on tracking changes of resources in a workflow and to assist the researcher in the documentation process by filling as many parts of the metadata as possible automatically (e.g. reuse information previously entered by the user for the same resource in another workflow step). For sharing and publishing workflow data we directly support a system frequently used for institutional repositories, as described above. Integration into existing institutional infrastructures is made possible by a plugin framework, allowing the RePlay-DH Client to be tailored to individual needs by adding custom implementations for interfaces.

Interoperability. Besides assisting in the elicitation of process metadata, the RePlay-DH Client also allows to export it. This export functionality is available for various formats and levels of granularity, such as an entire workflow graph, a certain workflow path or individual steps. While not directly derived from the PROV model or its extensions such as P-PLAN[15], our metadata scheme shares many aspects with those and transformation is a simple task. When exporting, the user can therefore choose between the native representation (serialized to JSON) or OWL-based variants. Due to the underlying plugin engine additional export formats can easily be integrated on demand.

6. Applicability

In this section we show in what ways the client is suitable for various types of NLP or DH tasks based on three examples and also how it can face recurring challenges when dealing with linguistic data or common processing steps.

[13] `https://www.doi.org/`
[14] `http://www.dspace.org`
[15] `http://www.opmw.org/model/p-plan/`

Manual annotation. Manually annotated corpora are one of the pillars of research in NLP. To evaluate a corpus for a given task researchers require precise knowledge about annotation guidelines, curation steps, automatic preprocessing or cleaning of the primary data. The ability to link to arbitrary resources as `input` for a step makes our metadata scheme well-suited for the first two aspects. Since the majority of annotation formats are represented textually, Git is a natural fit for tracking fine-grained changes made to them and ensuring their documentation.

Automatic processing. Quickly performed iterative and automatic processing steps are a hotbed for incomplete documentation when for example seemingly minor adjustments are left out. Our metadata scheme limits granularity of processing steps to have at most one `tool` instance. This way we ensure that no intermediary information gets lost. While this might appear demanding, the client supports mechanisms like drag & drop of files for usability and reduces the time required to document a single step.

No output. In the previous two data centered task types there is always a resource being modified or generated. To also document cognitive progress in a workflow, we allow the recording of steps that have no apparent effect on the resources in a workspace. Being essentially comparable to memos, they offer a great way to document insights gathered from a set of (`input`) resources.

Data size. Depending on the task, corpus or model resources can grow very large, making duplication caused by Git prohibitive. Our client allows users to exclude files from version control (either manually or based on a customizable size threshold), but to still cover them in the documentation. Combined with proper links to object metadata and documentation of previous steps, this can at least provide a level of reproducibility sufficient for many use cases.

Duplicates. With the lack of actually applied standards, it is quite common that within a single workflow resources are converted into multiple different formats for processing. While in principle still the same resource, those physically distinct instances could cause confusion in documentations. Our flexible approach to resource identification (see Section 4) allows proper unification of several physical instances of the same resource.

7. Pilot Project

For the design of our process metadata scheme described in Section 4 an associated annotation project was highly influential: In the manual annotation efforts for the gold standard in the GRAIN[16] release of the SFB732 Silver Standard Collection (Eckart and Gärtner, 2016) a setup very similar to RePlay-DH was used. Annotations were joined in a Git repository and the annotators had to enter formalized descriptions of their performed steps as part of their commits. These descriptions took the form of a preliminary version of the RePlay-DH process metadata and were provided in a simplified JSON format. Due to the lack of a dedicated support tool for eliciting the metadata (such as the client software presented in this paper) at the time, annotators were

[16] Also appearing as an article in this volume.

given textual templates to fill in relevant information in order to minimize overhead.
Subsequent automatic extraction of the metadata from the Git repository showed promising results in terms of usability for visualizing compositional information of the corpus. In addition, the pipelines creating automatic annotations for GRAIN were also designed in such a way that they create the same kind of process metadata as part of their analysis output. It is planned to release a curated version of the process metadata alongside the actual corpus data.

8. Availability

The tool itself is implemented in Java and runs on all major operating systems. Besides an installed Java Runtime Environment of version 8 or higher, no additional software is required for its use. Executable binaries, documentation and further information regarding the client software are available online[17]. Additional metadata for documentation is published in the framework of CLARIN [18].

9. Outlook

In this paper we presented the implementation of our previously proposed software tool for supporting process documentation by using version control as foundation. We contextualized the tool in the landscape of existing systems that deal with various aspects of workflow management or tracking applicable to tasks in NLP and DH. We also showed how our approach of separating different aspects of workflow documentation – namely the distinction between metadata describing *objects* used in a workflow and the *actions* performed – allows it to easily integrate into the diverse landscape of existing infrastructures and to better exploit the richness of available metadata repositories.
Discovery of available metadata records for resources in a workflow remains an open issue when it comes to external repositories. For future releases we plan to explore the feasibility of interfacing our client even more tightly with such systems to partly automate this discovery process. The possibility of defining metadata templates for entire steps which are frequently used in certain types of workflows is also something we intend to evaluate.

10. Acknowledgements

This work was funded by the Ministry for Science, Research and the Arts in Baden-Württemberg (MWK) via project RePlay-DH and the German Research Foundation (DFG) via the SFB 732, project INF.

11. Bibliographical References

Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Eberhard, C., Grüning, B., Guerler, A., Hillman-Jackson, J., Von Kuster, G., Rasche, E., Soranzo, N., Turaga, N., Taylor, J., Nekrutenko, A., and Goecks, J. (2016). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. *Nucleic Acids Res.*, 44(W1):W3–W10.

Calzolari, N., Soria, C., Gratta, R. D., Goggi, S., Quochi, V., Russo, I., Choukri, K., Mariani, J., and Piperidis, S. (2010). The lrec map of language resources and technologies. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Claerbout, J. and Karrenbach, M. (1992). Electronic documents give reproducible research a new meaning. In *Proc. 62nd Ann. Int. Meeting of the Soc. of Exploration Geophysics*, pages 601–604.

Deelman, E., Gannon, D., Shields, M., and Taylor, I. (2009). Workflows and e-science: An overview of workflow system features and capabilities. *Future Gener. Comput. Syst.*, 25(5):528–540, May.

Eckart, K. and Gärtner, M. (2016). Creating Silver Standard Annotations for a Corpus of Non-Standard Data. In Stefanie Dipper, et al., editors, *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, volume 16 of *BLA: Bochumer Linguistische Arbeitsberichte*, pages 90–96, Bochum, Germany.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Elming, J., Johannsen, A., Klerke, S., Lapponi, E., Martinez Alonso, H., and Søgaard, A. (2013). Down-stream effects of tree-to-dependency conversions. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, page 617–626, Atlanta, Georgia, June. Association for Computational Linguistics.

Gärtner, M., Hahn, U., and Hermann, S. (2018). Supporting sustainable process documentation. In Georg Rehm et al., editors, *Language Technologies for the Challenges of the Digital Age*, pages 284–291, Cham. Springer International Publishing.

Hinrichs, E. and Krauwer, S. (2014). The CLARIN Research Infrastructure: Resources and Tools for e-Humanities Scholars. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 1525–1531, May.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., DiPersio, D., Shi, C., Suderman, K., Verhagen, M., Wang, D., and Wright, J., (2016). *The Language Application Grid*, pages 51–70. Springer International Publishing, Cham.

McPhillips, T. M., Song, T., Kolisnik, T., Aulenbach, S., Belhajjame, K., Bocinsky, K., Cao, Y., Chirigati, F., Dey, S. C., Freire, J., Huntzinger, D. N., Jones, C., Koop, D., Missier, P., Schildhauer, M., Schwalm, C. R., Wei, Y., Cheney, J., Bieda, M., and Ludäscher, B. (2015).

[17] http://hdl.handle.net/11022/1007-0000-0007-C62F-6

[18] http://hdl.handle.net/11022/1007-0000-0007-C630-3

YesWorkflow: A user-oriented, language-independent tool for recovering workflow information from scripts. *CoRR*, abs/1502.02403.

Powell, A., Nilsson, M., Naeve, A., and Johnston, P. (2005). Dublin core metadata initiative - abstract model. White Paper.

Reich, M., Liefeld, T., Gould, J., Lerner, J., Tamayo, P., and Mesirov, J. P. (2006). GenePattern 2.0. *Nature Genetics*, 38(5):500–501, May.

Smith, M. (2002). Dspace: An institutional repository from the mit libraries and hewlett packard laboratories. In Maristella Agosti et al., editors, *Research and Advanced Technology for Digital Libraries*, volume 2458 of *Lecture Notes in Computer Science*, pages 543–549. Springer Berlin Heidelberg.

Treloar, A., Groenewegen, D., and Harboe-Ree, C. (2007). The data curation continuum: Managing data objects in institutional repositories. *D-Lib Magazine*, 13(9/10), September/October.

The ACoLi CoNLL Libraries: Beyond Tab-Separated Values

Christian Chiarcos, Niko Schenk
Applied Computational Linguistics (ACoLi) Lab
Goethe University Frankfurt, Germany
{chiarcos,nschenk}@em.uni-frankfurt.de

Abstract

We introduce the ACoLi CoNLL libraries, a set of Java archives to facilitate advanced manipulations of corpora annotated in TSV formats, including all members of the CoNLL format family. In particular, we provide means for (i) rule-based re-write operations, (ii) visualization and manual annotation, (iii) merging CoNLL files, and (iv) data base support.

The ACoLi CoNLL libraries provide command-line interface to these functionalities. The following aspects are technologically innovative and exceed beyond the state of the art: We support *every* OWPL (one word per line) corpus format with tab-separated columns, whereas most existing tools are specific to one particular CoNLL dialect. We employ established W3C standards for rule-based graph rewriting operations on CoNLL sentences. We provide means for the heuristic, but fully automated merging of CoNLL annotations of the same textual content, in particular for resolving conflicting tokenizations. We demonstrate the usefulness and practicability of our proposed CoNLL libraries on well-established data sets of the Universal Dependency corpus and the Penn Treebank.

Keywords: CoNLL data format, CoNLL-RDF, merging conflicting tokenization, interoperability

1. Background & Motivation

Since 1999, the Conference on Natural Language Learning (CoNLL)[1] has established a strong tradition of annually organized shared tasks within the NLP community. The addressed linguistic phenomena exhibit great diversity and included (but were not limited to) lexical semantics, semantic role labeling, dependency and discourse parsing, and coreference resolution. With their continuous progression in terms of linguistic complexity, the shared tasks reflect the maturation of statistical NLP, the dominating paradigm of computational linguistics in the 2000s. In many cases, successful participants established reference tools, and—as it allowed for comparative evaluation—the underlying formats (for training and test data) continued to be supported by succeeding NLP tools. This in fact has reinforced the global importance of the CoNLL format family and, as a result, the CoNLL format has ultimately become a de-facto standard within the language processing community.

CoNLL and other tab separated value (TSV)-based formats are widely used, and individual dialects come with considerable tool support, e.g., CoNLL-X, CoNLL-U or the formats of corpus tools like CWB (Evert and Hardie, 2011) or Sketch Engine (Kilgarriff et al., 2014). At the moment, however, there are considerable limitations with respect to

semantics: off-the-shelf database support is limited to CoNLL dialects with invariable number of columns, thus excluding semantic role labelling (since 2004) and discourse semantics (since 2015) in CoNLL,

syntax: while parsing CoNLL into an array is trivial, graph traversal and transformation on top of this requires considerable programming efforts,

tokenization: CoNLL presupposes one token (word) per line, if tokenization diverges, annotations can no longer be compared, and

interoperability: most tools are specific to one particular CoNLL dialect, positing constraints on columns and/or additional metadata as provided in comments.

We introduce the ACoLi CoNLL libraries in order to address these issues: They are applicable to every CoNLL TSV dialect, they provide flexible graph traversal and graph rewriting using existing W3C standards, they allow export into and import from off-the-shelf RDF triple stores, and support the automated merging of conflicting tokenizations. As shown in Fig. 1, the ACoLi CoNLL libraries can be used in different types of NLP architectures to complement existing NLP modules with an advanced level of interoperability: The **CoNLL Merge** package transforms and integrates *conflicting tokenizations* in CoNLL TSV, it *merges CoNLL files* by aggregating all (retokenized) columns of the source files, the **CoNLL-RDF** package yields an isomorphic rendering of CoNLL in *RDF*, thereby enabling *advanced manipulation of annotations* (graph rewriting) by SPARQL Update, *off-the-shelf database* support by RDF Triple/Quad Stores, and access with a W3C standardized *query language* (SPARQL). Both packages are implemented in Java and available under Apache 2.0 license.[2]

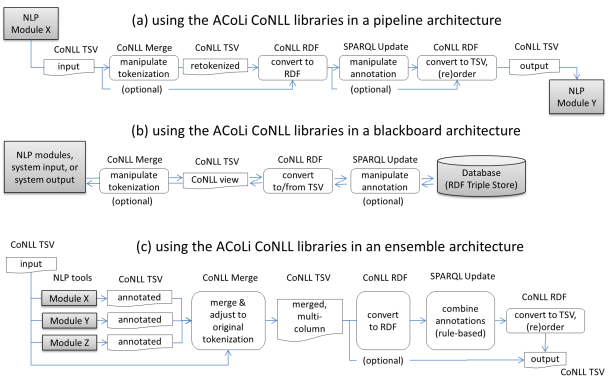


Figure 1: Using the ACoLi CoNLL libraries in different NLP architectures

[1] http:/www.signll.org/conll

[2] https://github.com/acoli-repo/conll-rdf, https://github.com/acoli-repo/conll

2. Manipulating TSV data

The **original data model** of CoNLL TSV formats is a sequence of variable-width tables, each representing a sentence, with rows in a fixed (sequential) order. Different CoNLL dialects differ in the choice and definition of columns, however, all are characterized by putting *one word per line* (OWPL), with annotations of these words stored in the respective rows. While basic manipulations such as reordering, merging, and dropping columns are simple and can be accomplished with off-the-shelf tools such as sed, advanced NLP problems often require manipulating *rows*: All CoNLL formats come with the limitation of imposing one single segmentation level on the text, but in complex NLP architectures, different modules may produce, require or produce files with deviating segmentations. Yet, tokenization strategies generally differ with respect to the research question, tokenizations can drastically *disagree*, so that multiple linguistic annotations on top of concurrent tokenizations of the same text usually require intense efforts for harmonization, as in the following phrase from the Penn Treebank (Marcus et al., 1999), resp. OntoNotes (Hovy et al., 2006):

```
a           DT        a      DT
                      19     CD
19-month    JJ        -      HYPH
                      month  NN
                      cease  NN
                      -      HYPH
cease-fire  NN        fire   NN      (wsj0655)
```

Problems multiply if an annotation inserts empty elements, e.g., the OntoNotes syntax in opposition to the OntoNotes NER. With the `CoNLLAlign` class in the **CoNLL Merge** package, we provide a fully automated solution for merging CoNLL files with concurrent tokenizations in three different modes:

default adopt the first tokenization as gold tokenization, lossless (for mismatches, add a new line after the mismatch, and mark it as an 'empty' non-word by `*RETOK*`)

-f force the first tokenization as gold tokenization (for mismatches, attach the annotations of the mismatched word to the annotations of the last line, using + as concatenator)

-split split tokens into longest common substrings, lossless (for mismatches, split annotations using the I[O]BE[S] scheme)

Figure 2 illustrates these strategies. We recommend the default option for the quantification and inspection of tokenization mismatches in preparation of NLP experiments or corpus conversion; `-f` for NLP pipelines, as this permits flexible retokenization against a target tokenization (at the price of information loss); `-split` for the fast integration of corpus data: annotations are anchored in the primary data (no artificial 'empty' elements), and the original annotation is recoverable.

CoNLL Merge builds on Myer's Diff (Myers, 1986), applied to the `WORD` (`FORM`) column, and is thus capable of tolerating spelling differences, e.g., for escaped special characters like brackets or quotes. With `-split`, it iterates on non-aligned sequences with a character-level diff, with sequences of matching characters pairs are merged into substrings.

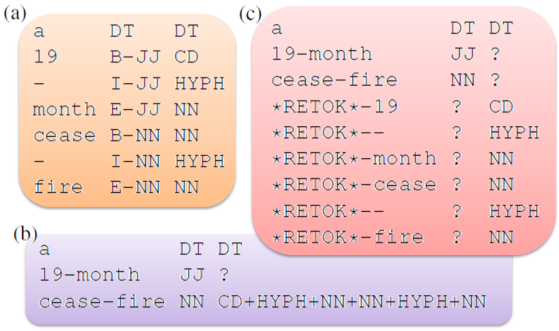


Figure 2: Merging concurrent tokenizations with **`-split`** (a), **`-f`** (b) and default options (c)

To our best knowledge, CoNLL Merge is the first publicly available tool for the fully automated, domain-independent merging of linguistic annotations of the same text with concurrent tokenizations. It does build on earlier work by Chiarcos et al. (2012), but while their implementation was specific to a complex XML standoff format with limited technological support from the wider community, we now provide an open source implementation for a popular de-facto standard for linguistic annotations in NLP.

3. Beyond Tab-Separated Values

Off-the-shelf database support for CoNLL TSV is currently limited to dialects with constant number of columns. However, this cannot be taken for granted, at least for semantic annotations, as the CoNLL representation of SRL, for example,[3] introduces an additional column *per frame instance*, i.e., sentences vary in the number of columns.

To facilitate CoNLL processing and querying, the **CoNLL-RDF** package provides a mapping from OWPL corpus formats to an isomorphic, and lossless representation in RDF (Chiarcos and Fäth, 2017). In order to provide a generic converter of CoNLL data, we must not rely on a fixed set or order of columns, but instead, expect user-provided labels. CoNLL-RDF provides a trivial of mapping comparable to CSV2RDF (Tandy et al., 2015), yet, a number of CoNLL-specific extensions are required (marked by * below):

- (*)Preserve CoNLL comments as comments, but do not interpret them.
- Assign every non-empty row a unique URI ('primary key') based on a user-provided base URI for the document, the sentence number and the word ID (or position): In the resource `https://github.com/UniversalDependencies/UD_German/blob/master/de-ud-dev.conllu`, the second word in the first sentence will receive the URI `https://github.com/`

[3] Similar problems exist with the representation of discourse relations for the 2015-2016 shared tasks.

UniversalDependencies/UD_German/blob/master/de-ud-dev.conllu#s1.2,[4] resp., :s1.2 in short.

- (*)Define every row as a word, and connect it to its successor using the NIF vocabulary (Hellmann et al., 2013)[5]: :s1.2 a nif:Word; nif:nextWord :s1.3 .
- Given a **user-provided list of column labels** (say, LEMMA for the second column in UD), we create datatype properties in the conll: namespace, and assign the word its corresponding annotation as a literal value, e.g., :s1.2 conll:LEMMA ''sein''.
- As an exception, the HEAD column is mapped to an object property ('foreign key'): :s1.2 conll:HEAD :s1.5 . This convention enables special handling of intra-sentential cross-references and is suitable for—but not restricted to—dependency syntax.
- (*)Special treatment for the user-provided label X-ARGs: If there is a column X, say, PRED, then assume that every word with a non-empty value for PRED introduces an additional column PRED-ARGi for its arguments. For every word w that has a non-empty annotation a in PRED-ARGi, and the word p_i that has the ith non-empty value in PRED, define w as argument of p_i: p_i conll:a w., e.g., :s1.2 conll:A1 :s1.1 .

In consequence, we obtain an isomorphic representation of the original CoNLL data structure in RDF which is semantically shallow,[6] but can be effectively queried, manipulated and serialized back into CoNLL using off-the-shelf RDF technology. In particular, this includes a rich infrastructure of databases, webservices, APIs, models for resource publication and linking (Chiarcos et al., 2013).

Even though it lacks formal semantics (by design), the CoNLL RDF model can also serve as a basis to transform CoNLL data into semantically well-defined formalisms such as POWLA (Chiarcos, 2012) or full-fledged NIF (Hellmann et al., 2013).

For the en-bloc conversion of CoNLL data to CoNLL-RDF, we provide the JAVA class CoNLL2RDF. Fig. 3 illustrates CoNLL-RDF sample data for the first sentence of the German UD development set, together with its rendering in CoNLL and other derived representations.

4. Advanced Graph Operations

A key advantage of an RDF representation is that a W3C-standardized query language for the flexible querying and manipulation of this data can be employed (Buil Aranda et al., 2013, SPARQL). As an example, consider querying paths in a dependency tree. SPARQL 1.1 property paths provide convenient means for complex path configurations in labeled graphs, supporting logical operations, reversal of direction and iterations of edges. While, after conversion from CoNLL TSV, dependency labels are stored as literal values of conll:EDGE, it is easy to transform them into object properties using a simple SPARQL Update statement:

```
INSERT {
  ?dep ?prop ?head .
} WHERE {
  ?dep conll:HEAD ?head .
  ?dep conll:EDGE ?edge .
  BIND(IRI(CONCAT(
    'http://universaldependencies.org/u/dep/',
?edge) AS ?prop))
}
```

As the example also illustrates, RDF resources can be defined in a way that their IRIs/URIs resolve against external resources, e.g., the dependency label nsubj yields the URL http://universaldependencies.org/u/dep/nsubj and thus a human-readable definition. For resources other than UD, one can thus directly link to *machine-readable* information and use this in a federated search, e.g., regarding linguistic annotations,[7] lexical entries from Wiktionary,[8] semantic frames,[9] multilingual word sense information,[10] or general concept knowledge.[11] After this transformation, it is now possible, for example, to retrieve all nominal subject of verbs, including those nested in conjunctions:

```
PREFIX ud:
  <http://universaldependencies.org/u/dep/>
SELECT ?verb ?nsubj
WHERE {
  ?verb conll:UPOS "VERB".
  ?verb ud:nsubj/ud:conj? ?nsubj.
  ?nsubj conll:UPOS "NOUN".
}
```

An interesting feature here is that complex graph configurations can be expressed and retrieved in a compact and human-readable way: here, an optional transition along a ud:conj edge is permitted, but not required.

SPARQL Update can not only be used for querying, but also for *manipulating* annotations. To facilitate processing data streams, the class CoNLLStreamExtractor in the **CoNLL RDF** package reads CoNLL from stdin, applies CoNLL2RDF sentence by sentence and returns valid CoNLL-RDF. In addition to this mere conversion functionality, the CoNLLStreamExtractor supports data manipulation by means of SPARQL Update: It takes as additional arguments a list of files containing SPARQL Update

[4] The UD sent_id is currently not used, because it only appears in a comment. However, future support for UD-specifics is possible.

[5] http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core

[6] Note that the conll: namespace used here is not connected with any ontology, but populated by properties as defined by the user (column labels) or in the data (values for X-ARGs columns).

[7] http://purl.org/olia

[8] http://kaiko.getalp.org/about-dbnary/

[9] http://premon.fbk.eu/

[10] http://babelnet.org/

[11] http://dbpedia.org/

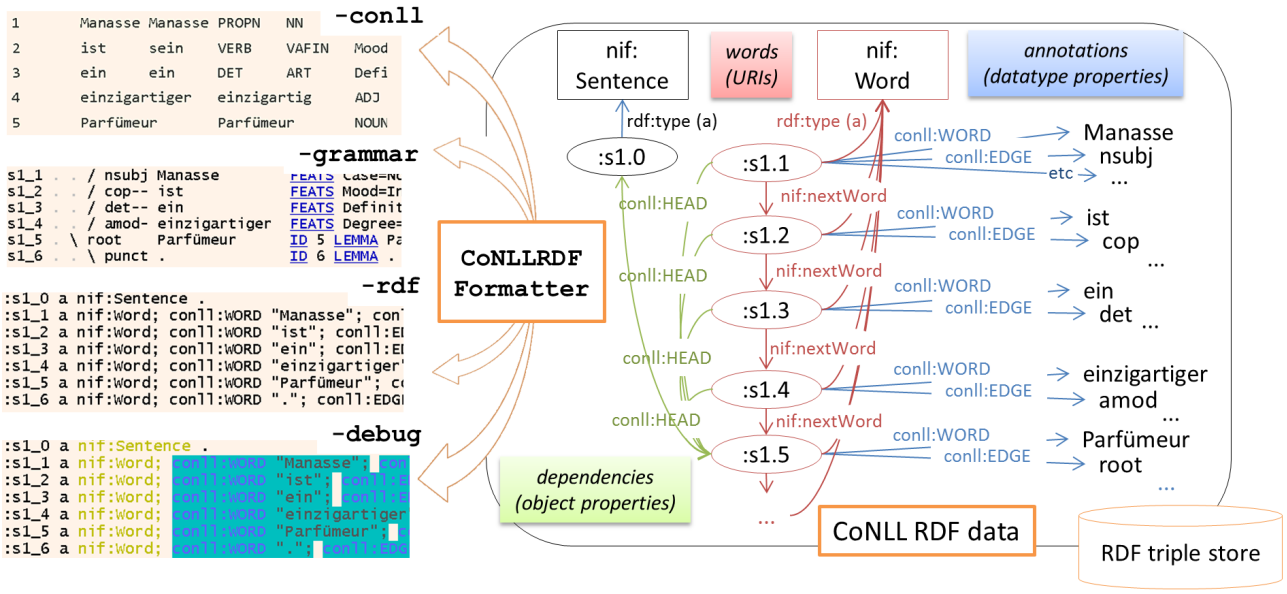


Figure 3: German sample sentence from UD, its CoNLL-RDF representation, and different serializations/visualizations

statements, introduced by the flag `-u`. Every individual file represents a module, which can be stacked to form a processing pipeline. Successively, these SPARQL Updates are applied to the individual sentences, thereby rewriting the RDF graph. We also support iterated applications, marked by an integer that can follow the respective SPARQL file in curly brackets.

In essence, the use of SPARQL 1.1 allows *unconstrained graph rewriting and enrichment*, and sample pipelines for different tasks have been developed, including, but not limited to

- combining `conll:HEAD` and dependency labels (`conll:EDGE`) into object properties,
- converting PCFG parses to dependency parses,
- transforming dependency annotations to produce UD-conformant 'semantic' dependencies,
- reducing various span annotations to the respective heads in the dependency annotation,
- stemming (using `BIND` and SPARQL string operations),
- chunking,
- rule-based dependency parsing (using a modified shift-reduce scheme), and
- supersense inference (by consulting RDF editions of VerbNet and WordNet).

5. Serializing CoNLL-RDF

CoNLL-RDF is an extensible RDF data model that can be serialized in any RDF format. The class `CoNLLRDFFormatter` in the **CoNLL RDF** package generates different serializations of CoNLL RDF data. It reads CoNLL-RDF in blocks of sentences from `stdin`, and, depending on its parameters, produces CoNLL TSV (`-conll`), a human-readable representation for dependency annotations (`-grammar`), *canonically formatted* CoNLL-RDF (default, `-rdf`), or canonically formatted CoNLL-RDF with syntax highlighting (on Un*x shells with color support, `-debug`), see Fig. 3.

Canonically formatted CoNLL-RDF has been designed as a compromise between Semantic Web standards and the low-level processability of CoNLL TSV: It is a highly constrained subset of Turtle that emulates OWPL formats.

- Sentences are separated by empty lines.
- Following eventual prefix declarations, first line defines the sentence as a `nif:Sentence`. For all except the first sentence of a corpus, the first line is preceded by a `nif:nextSentence` statement marking its position.
- The second line holds the first content word, defines it as a `nif:Word`, followed by its `conll:WORD` and other annotations, as well as a `nif:nextWord` statement pointing to the next word in the sentence. If the current word has no dependency annotation or is a `root`, its `conll:HEAD` is the sentence.
- CoNLL-RDF uses the Turtle separator ";" to enumerate the annotations assigned to a particular `nif:Word`, and "," to enumerate multiple values for the same annotation. CoNLL-RDF follows CoNLL in that all triples referring to one `nif:Word` are written in one line, concluded with ".".

The resulting format adopts many basic CoNLL conventions (comments starting with #, sentences separated by empty lines, one token per line, strictly ordered fields) and others modified (annotations separated by ; and identified by property names rather than position). Most importantly, canonically formatted CoNLL-RDF can be as easily processed with low-level string manipulations as the original CoNLL format, albeit not with an array as primary data

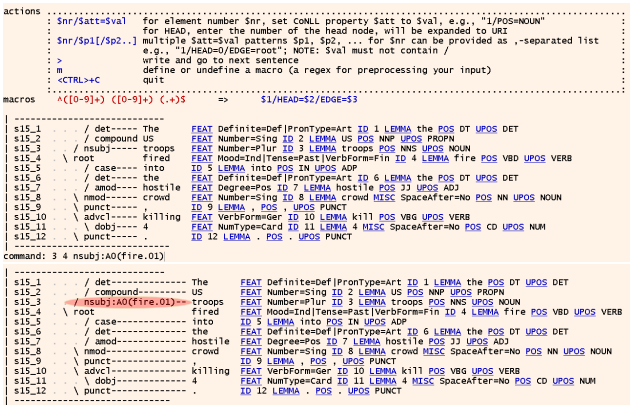


Figure 4: `CoNLLRDFAnnotator`: annotating arguments with PropBank roles (English UD corpus).

structure, but with a dictionary/hashmap. Yet, it is also possible to process canonically CoNLL-RDF with off-the-shelf RDF technology. The canonical format can be easily restored using the `CoNLLRDFFormatter`.

For NLP applications, the `CoNLLStreamExtractor` and `CoNLLRDFFormatter` can be connected in a pipeline, so that CoNLL TSV data is read from `stdin`, manipulated using SPARQL Update, and transformed back into CoNLL TSV for further processing with off-the-shelf NLP tools.

We also provide rudimentary support for manual annotation on Un*x shells with color coding: As an extension of our serialization routines, the `CoNLLRDFAnnotator` reads and writes CoNLL-RDF sentence by sentence, it visualizes it using the human-readable representation shown above, and, for every word in the sentence, it allows to overwrite every `conll:`-property with a new value. With user-provided macros (replacement rules), multiple features can be overwritten, e.g., with a pre-defined macro that allows to set `conll:HEAD` and `conll:EDGE` at the same time: The operation `3 4 nsubj:A0(fire.01)` in Fig. 4 means that the third word in the current sentence should point to the fourth as its head, with a dependency label that combines the UD dependency label with information about its PropBank frame.

6. Summary

In this paper, we introduce the ACoLi CoNLL libraries, a set of Java archives to facilitate advanced manipulations of corpora annotated in TSV formats, including all members of the CoNLL format family. These are based on our earlier research on corpus representation (Chiarcos, 2012; Chiarcos and Fäth, 2017) and structural interoperability (Chiarcos et al., 2012) with a focus on the popular CoNLL format family.

The primary goal of this effort is to facilitate interoperability between existing components in complex NLP architectures. The CoNLL Merge package provides a fully automated solution to the problem of concurrent tokenizations and leverages concurrent tokenizations of the same text by supporting fully automatized retokenization of annotated texts. The CoNLL-RDF package provides a transformation of CoNLL TSV into (and from) an isomorphic, but shallow RDF representation of CoNLL data, which facilitates database support and querying, and enables advanced manipulations of annotations CoNLL data by means of graph rewriting. We provide stream processing capabilities for these operations, by processing CoNLL and CoNLL-RDF data streams sentence by sentence, and export in different serializations, including CoNLL TSV, canonically formatted CoNLL-RDF with optional syntax highlighting, and a human-readable view for dependency annotations. In addition, a rudimentary annotation functionality is provided.

The ACoLi CoNLL libraries are designed as minimal software components that provide a layer of interoperability for glueing together heterogeneous modules and existing software components in complex NLP systems. CoNLL Merge allows to leverage tokenization differences and CoNLL-RDF permits a full rewrite of existing annotations. Beyond this, CoNLL-RDF can actually be used to implement rule-based NLP components, using SPARQL property paths and SPARQL Update for graph rewriting. As standalone application, they only provide a command-line interface to their functionalities.

With SPARQL Update, CoNLL-RDF provides a powerful, and W3C-standardized graph rewriting formalism which allows us to separate the transformation logic (SPARQL Update) from the conversion between different CoNLL dialects (CoNLLStreamExtractor, CoNLLRDFFormatter). Example transformations are provided with the release, more complete pipelines as well as rule-based NLP components developed in SPARQL Update will be subject to subsequet publications.

CoNLL Merge and CoNLL-RDF are designed to be applicable to every OWPL corpus format and thus depend on user input regarding the labels and types of columns. They impose minimal terminological constraints:

- Merging and visualization: The user has to identify the column that contains the primary data (we follow the original terminology in naming this `WORD` rather than `FORM`).
- RDF conversion: If a column `ID` is provided, it will be used to generate URIs. Columns with labels `HEAD` or X`-ARGs` are rendered as object properties ('foreign key') rather than string annotations.
- Visualization: For dependency relations and their labels, we expect the column labels `HEAD` and `EDGE`.
- RDF to TSV conversion: We export properties from the `conll:` namespace in the user-defined order.
- CoNLL RDF properties and IDs must conform to IRI conventions, i.e., the following characters are reserved: : / ? # [ ] @ $ & ' ( ) * + , ; =

No additional a priori naming conventions apply within the CoNLL-RDF core infrastructure. User- or usecase-specific SPARQL Update scripts do impose naming conventions. CoNLL Merge and CoNLL-RDF, together with sample data and sample scripts, are released under the Apache license 2.0 via our public Github repository `https://github.com/acoli-repo`.

Acknowledgments

Our research at Goethe University Frankfurt was supported by the project 'Linked Open Dictionaries (LiODi, 2015-2020)', funded by the German Ministry for Education and Research (BMBF).

7. References

Buil Aranda, C., Corby, O., Das, S., Feigenbaum, L., Gearon, P., Glimm, B., Harris, S., Hawke, S., Herman, I., Humfrey, N., Michaelis, N., Ogbuji, C., Perry, M., Passant, A., Polleres, A., Prud'hommeaux, E., Seaborne, A., and Williams, G. (2013). SPARQL 1.1 overview. Technical report, W3C Recommendation.

Chiarcos, C. and Fäth, C. (2017). Conll-rdf: Linked corpora done in an nlp-friendly way. In Jorge Gracia, et al., editors, *Language, Data, and Knowledge: First International Conference, LDK 2017, Galway, Ireland, June 19-20, 2017, Proceedings*, pages 74–88. Springer International Publishing, Cham.

Chiarcos, C., Ritz, J., and Stede, M. (2012). By all these lovely tokens... merging conflicting tokenizations. *Language resources and evaluation*, 46(1):53–74.

Chiarcos, C., McCrae, J., Cimiano, P., and Fellbaum, C. (2013). Towards open data for linguistics: Linguistic linked data. In Alessandro Oltramari, et al., editors, *New Trends of Research in Ontologies and Lexical Resources: Ideas, Projects, Systems*, pages 7–25. Springer Berlin Heidelberg, Berlin, Heidelberg.

Chiarcos, C. (2012). A generic formalism to represent linguistic corpora in rdf and owl/dl. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Evert, S. and Hardie, A. (2011). Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium. In *Proceedings of the Corpus Linguistics 2011 conference*, Birmingham. University of Birmingham.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *Proc. of the 12th International Semantic Web Conference (ISWC-2013)*, pages 98–113, Sydney, Australia, Oct. Springer.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovář, V., Michelfeit, J., Rychlý, P., and Suchomel, V. (2014). The sketch engine: ten years on. *Lexicography*, 1(1):7–36, Jul.

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., and Taylor, A. (1999). Treebank-3. LDC Catalog No.: LDC99T42, ISBN, 1-58563-163-9.

Myers, E. W. (1986). Ano(nd) difference algorithm and its variations. *Algorithmica*, 1(1):251–266.

Tandy, J., Herman, I., and Kellogg, G. (2015). Generating RDF from tabular data on the web. Technical report, W3C Recommendation.

What's Wrong, Python? – A Visual Differ and Graph Library for NLP in Python

Balázs Indig[12], András Simonyi[1], Noémi Ligeti-Nagy[1]
[1]MTA-PPKE Hungarian Language Technology Research Group
[2]Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
H-1083 Budapest, Práter street 50/A
{indig.balazs, simonyi.andras, ligeti-nagy.noemi}@itk.ppke.hu

Abstract

The correct analysis of the output of a program based on supervised learning is inevitable in order to be able to identify the errors it produced and characterise its error types. This task is fairly difficult without a proper tool, especially if one works with complex data structures such as parse trees or sentence alignments. In this paper, we present a library that allows the user to interactively visualise and compare the output of any program that yields a well-known data format. Our goal is to create a tool granting the total control of the visualisation to the user, including extensions, but also have the common primitives and data-formats implemented for typical cases. We describe the common features of the common NLP tasks from the viewpoint of visualisation in order to specify the essential primitive functions. We enumerate many popular off-the-shelf NLP visualisation programs to compare with our implementation, which unifies all of the profitable features of the existing programs adding extendibility as a crucial feature to them.

Keywords: Tools, Visualisation, Python

1. Introduction

In natural language processing (NLP), it is obviously fundamental to achieve good performance on the well-researched tasks; however, on a task currently under development, it is even more relevant to be aware of what have happened and where the program has gone wrong. The thorough analysis of the errors can lead to new ideas and methods, and finally, to better results. In most of the NLP tasks, this error inspection is carried out manually, while the number of tools supporting this work phase is low. The main reason of that is that for visual comparing there needs to be a good graphing library that displays output data correctly, which can be also hijacked to display visual differences in a form that helps human processing. Finally, a well-visualised good example can tell more of the inner-working of an algorithm than a thousand words or tables.

In scientific computing, especially in NLP, two programming languages serve as the source language of the majority of the new tools. JAVA is popular because of its maturity and cross-platform compatibility. In the job market, one can find many well-trained programmers developing in JAVA, the universities are still putting a great emphasis on teaching JAVA, and finally, the slowly evolving industrial applications are still using JAVA for new projects because of its adoption. However, Python is still gaining more and more popularity on the basis of its simplicity allowing fast prototyping and its ability to hide the performance of the critical parts – which are usually implemented in C++ – under a newbie-friendly Python API without sacrificing the overall running or development performance. The cheap work power of fast-trained Python programmers may turn the tide in the future. At the moment, main tools can be seen having both Python and JAVA APIs. Concerning the NLP visualisation task, there are mostly JAVA programs on the shelf, which are – strictly technically – not good as they can not be integrated well with Python programs. We stand with Python, therefore in this paper, we mostly sample Python programs along with a few others for comparison. We argue that the available solutions are not good enough for the future Python language NLP visualisation needs, therefore we created our own program in Python.

We describe the common NLP tasks of the past years to discover their common features, which are expected to be inherent in the methods and tasks of the following years. We gather the list of available tools to compare them and present our expectations for a new visual comparison program. Finally, we present our solution and our future plans.

2. NLP Tasks and Formats

The first *Conference on Computational Natural Language Learning (CoNLL)* was held in 1997 (Ellison, 1997). The first shared task, the *English noun-phrase chunking shared task* was announced in 2000 (Tjong Kim Sang and Buchholz, 2000) in a special format, the *CoNLL-2000 format*[1]. Later on, each year this format has evolved, extended and has become a de facto standard. The main idea behind the format is that each token is represented in a line of a text file and each of its features is represented as a field separated by a white-space – mostly a tabulator – character. The sentences are separated by a newline, and the order of the features has been standardised over the years. This so-called vertical format was a lightweight but machine-parseable alternative to XML by resembling / specialising the already existing *Tab Separated Values (TSV)* format[2]. The first field of each record (token) is always the actual token, the second is the lemma if any, and the third is the *Part of Speech*

[1]Ide et al. (2017) provides a broad overview of the state-of-the-art in standards development for language resources, including CoNLL and dependency annotations.

[2]A special hybrid version of TSV and XML format called *vertical file format* is used also in Sketch Engine (Kilgarriff et al., 2014) to represent corpora.

(POS) tag. The rest of the fields often differs and represents two kinds of structures.

One basic type of structure among tokens is *bracketing*. Many variants of bracketing task exist (e.g. NP-chunking, named-entity recognition) and many tasks can be interpreted as bracketing[3]. Basically, when a word or multiple subsequent words need to be labelled[4] – which implies bracketing – one can represent the task with the same data structure, and from our perspective, their visualisations are practically the same too.

Another basic type of structure being represented in various CoNLL formats comes from the further processing of the input sentence, where a one-level bracketing is insufficient. In *constituent parsing* immediate constituents are represented as brackets and it is needed to define higher-level relations between them to get the full parsing. These relations between the tokens (or groups) are represented by edges having a direction and a label, optionally. In *dependency parsing*, directed edges are always used and no artificial constituents are introduced to name the underlying groups of tokens. These two techniques can be represented in the data uniformly by using references to the token number to denote the edges. The resulting table cannot easily be read by humans but is practical for machines. This is the root cause why one would need a visual comparison tool.

There is a third type of commonly used structure which is not represented in the CoNLL format: the sentence alignment task. In the field of machine translation, tokens in a sentence should be reordered to get the correct translation. Sentence aligner tools also make mistakes, which, again, must usually be investigated by a human. In order to compare the alignments, one must see the two alignments with the matching and differing edges. Not to mention when one needs to see which part of the source sentence is translated to which part of the target sentence. This latter task involves bracketing and the alignment edges in conjunction.

We can see that most of the current NLP tasks involving supervised learning could profit from a visual comparison tool. A tool like this could help humans inspect the automatically generated output in view of the gold standard and increase the productivity of error analysis.

3. Traditional NLP centric visualisation and visual comparison tools

In this section we list the most important data visualisation libraries, focusing on their strength and weaknesses. There are a plethora of general data visualisation tools, that support many languages and are able to create interactive and publication-ready figures as well. In this paper, we sample only a few that suppose to shed light on the classes of the available programs regarding their good and bad sides.

In LaTeX there are many libraries that support general, linguistic and NLP focused graphing for vector-graphical publication-quality images, which use *PGF/TikZ* or other macros. Their common weakness is that they are optimised for printing and do not support interactive workloads; however, they have a nice output and are excellent for their specific task and easy to learn for LaTeX experts.

From the viewpoint of layout, one must consider *GraphViz* (Gansner and North, 2000) as it has advanced layout properties and can be customised with the standard DOT graph description language format for input, and it supports various output formats as well. It is, however, designed for non-interactive use and lacks the features needed for real interactive usage with a layout not easily customised.

As the de facto NLP framework for Python is *Natural Language Tool Kit (NLTK)* (Bird et al., 2009), it is a good starting point for searching for a proper base implementation. Its data structures and format handling capabilities are exceptional, but its interactive visualisation part is less extraordinary. It uses *GraphViz* for dependency graphs and a TKinter GUI for displaying the generated SVG figures in a cross-platform compatible way. As it lacks the layout code, only its input format and corpus handling capabilities are useful for our purpose.

In recent years, a new NLP framework in Python emerged from NLTK, which is called *Spacy* (Honnibal and Johnson, 2015) and it is gaining more and more popularity because of its speed and performance. It is a full-fledged NLP framework with a beautiful, modern, web-based visualisation module called *Displacy*. From our point of view, its main problem is that it does not support comparison and it has a heavy weight. Even if one would decouple the visualisation part and build a whole new comparison framework, the maintenance burden of the decoupled part would be too high because of the pace of Spacy's development.

Another novel emerging web-based visualisation library in Python is *Bokeh* (Bokeh Development Team, 2014), which is basically an interactive visualisation program designed for browsers around *D3.js* (Bostock et al., 2011) [5]. Its main goal is to release the users from the need to fiddle with JavaScript: it can be entirely customized from Python. The main problem of *Bokeh* is that it is not NLP centric and too general, and its API is rather abstract to be used directly. However, it can be used as a low-level graphing API for a custom layout engine as well as a basic WebUI.

We must mention the class of classic GUI-based or headless standard graphing APIs. The emerging *Matplotlib* (Hunter, 2007) is a great plotting API for Python, which is easier to interact with from Python than the other alternatives like *GNUPlot*[6]. These programs are able to create publication-ready figures in numerous supported exporting formats with the standard low-level primitives and also can be used as a base for interactive tasks as well. It is easy to write the NLP centric primitives with the standard ones and use the layout engine for our purposes.

For the specific task of NLP centric visual diffing we have found two off-the-shelf solutions, which are implemented in JAVA, and nowadays are not actively developed: one example is *MaltEval* (Nilsson and Nivre, 2008), which has

[3] For example, any kind of tokenisation task (e.g. paragraph, sentence or token splitting) can be interpreted as bracketing.

[4] Even POS-tagging can be interpreted as a labelled bracketing of each individual token.

[5] The goal is very similar to the *Computable Document Format* interactive demonstrations capability of the proprietary Wolfram's Mathematica, which can be tried at `http://demonstrations.wolfram.com/`

[6] `http://gnuplot.info/`

the typical anti-features we had faced with the aforementioned programs as well as they are too well-embedded into a framework: they cannot be hijacked for any purpose other than their well-defined role, which makes their usage for a slightly different task impossible, thus condemning them not to be used by a greater audience. It is hard to decouple them into modules to accommodate novel representations, thus forcing the user to reimplement much of the inner-workings, which is mostly a duplicate effort.

The other off-the-shelf example for the specified task is *What's Wrong With My NLP?*[7]. It has an unpleasant, old-style *graphical user interface (GUI)* (see Figure 5), which is considered clumsy and the rendered figure is a raster image, which can only be exported to EPS format. The internal representation is very promising from the aspect of extension to new formats of features. It is the best program currently available for our purpose.

As one can see, we checked multiple programs which could bring us closer to our goal: a lightweight library written with toolbox-philosophy in mind ready for versatile NLP-centric graphing and diffing in order to create a program, which could provide a human-friendly interactive comparison solution easily extendible to handle new formats and different layouts. We have found many well-defined or too abstract off-the-shelf implementations, that would have placed a large burden on us considering later development. Hence, we started to design a new implementation in Python with the followings in mind: (a) we tried not to constrain the edge drawing at all in our program, and (b) we tried to mix all the good features of the aforementioned programs in order to gain larger audience and satisfy the standard and non-standard comparison and visualisation needs as well. Some of the features were previously unavailable in any of the existing programs as their authors concentrated only on the standard NLP tasks. Our implementation is very close to *What's Wrong With My NLP?*. however, our goal was to use the resulting program in a pure Python environment and thus it only served as the base of our new Python implementation[8], which we will present in the next section in detail.

4. Our Solution: `LibWhatsWrong`

As described in the previous section, there is no real off-the-shelf solution in Python, which would help the interactive visual comparison of data and the visual, step-by-step debugging of a program's result with the possibility of exporting the resulting figures in a web-compliant format, that may also be embedded into scientific papers. The only similar solution implemented in JAVA has its own problems with the old-fashioned GUI and nowadays it is not actively developed.

Our goal is web-compliance in conjunction with the support of printable formats. To get started we have chosen the W3C standard *Scalable Vector Graphics (SVG)* as the final format, as there are many lightweight libraries to generate figures in this format in many languages. We selected *SVGWrite* [9] for the initial implementation. We designed an abstraction for the needed logical primitives as we wanted to be able to draw the standard drawing primitives with different drawing backends, such as *Matplotlib* or *Bokeh*.

We wanted to separate the input formats and corpus handling from the layout code, to encourage the potential users to implement even their really own format or utilise some other implementation of parsers of the widely-known data formats, for example, the ones in *NLTK* with our exposed API. This decision makes it possible in the future to quickly write a parser in Python for any specific format – based on our example implementations – and use it as a replacement. Between the input data – which is parsed into a unified form – and the description of the logical primitives for the drawing backend, lies the layout and the comparing code. Our implementation of layout generation and comparing is based on *What's Wrong With My NLP?*, with modifications to allow more control to customise the layout. One can create custom layouts easily by combining the layout primitives (which includes dependency, span and alignment layouts) or create new layouts from lower level logical or standard primitives.

The comparing process is fully separated from the layout computing. Numerous features of the primitives, which have a large impact on the complete laid out figure, can be set as parameters such as colour, font type, size, etc. This design of the core library allows one to use each step separately – even without the comparison – with his or her data in any supported or future format either from the graphical, command line or web UI or even from a Python program via an API. This fine-grained control was previously not available in *What's Wrong With My NLP?* or in any of the aforementioned solutions.

In our implementation it is also possible to have multiple edges from the same starting element to the same ending one or to form a loop by returning to the starting token, which is not a trivial requirement as it cannot be found in any of the aforementioned implementations[10] (see 5.. section for ideas and possible applications). The absence of this feature is caused by the fact that a well-formed dependency syntax tree cannot contain such edges, however, these types of errors are the ones which would be essential for a human corrector or may be required for a non-standard representation (see 5.. section for details).

The laid out figures can be exported to SVG, PDF and EPS formats currently and the range of the supported formats can be widened with the usage of *Matplotlib* as backend. To demonstrate the power of our implementation, we implemented a *GUI* around the library, that has all the features of *What's Wrong With My NLP?* – including interactive modification of the resulting figure –, but with a modern, sound GUI based on Qt 5 in PyQt (see Figure 6). There is also a simple demonstrative CLI for the basic drawing and comparing. Besides that, we plan to use *Bokeh* as UI and

[7] `https://code.google.com/archive/p/whatswrong/`

[8] We must note that with the proper knowledge of JAVA and the available libraries our main feature differences can be easily backported to *What's Wrong With My NLP?*to get a fully functioning future-proof NLP centric graph library and differ tool for JAVA.

[9] `https://github.com/mozman/svgwrite`

[10] The CoNLL formats do not support double edges and for example *Displacy* draws the two edges on each other

drawing backend to gain web UI functionality. In the next section, we present some useful ideas and possible applications for our library.

5. Possible applications

In this section, we would like to demonstrate the novelty and practicality of our application through four visualisation examples. We also enumerate some ideas regarding how to use our program, which might be useful for others in the future.

Although CoNLL format does not support double edges at all, and other applications simply draw two edges on each other if needed, correctly and visibly visualising two edges between the same two words is not only important because of our need to use the visualisation tool to improve the algorithm behind the given drawing, but also because there may be some parsing systems that specifically require and produce double edges. From a linguistic point of view, there are special relations between the constituents of a sentence that should be explored during the parsing: *focus*, for example, is a grammatical category that determines which part of the sentence contributes new, non-derivable, or contrastive information (Halliday, 1968). In Figure 1, we demonstrate how double edges – here an edge between the verb and its focus, and an edge between the verb and its object – are represented in our visualisation tool.

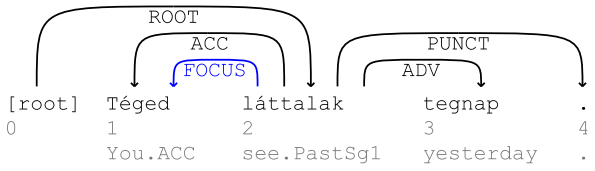


Figure 1: The visualisation of the dependency parsing of the sentence *Téged láttalak tegnap.* 'I saw YOU yesterday.'. The example demonstrates how our tool visualizes double dependency-like edges: one edge `ACC` starts from the verb *láttalak* and ends in its object, *téged*, while the other edge between the exact same tokens, `FOCUS`, means that the object here is the focus of the sentence (represented by the verb as the root). The two edges are drawn separately, and with different colours (blue and black), which indicates that the blue edge represents a connection on a different parsing level.

A loop – when an edge returns to the same token it started from –, is not part of the dependency formalism. However, some of the aforementioned tools support its visualisation, others do not. Drawing a loop is part of our solution, as we claim that it has an importance in parsing. For example, in some languages, like in German or in Hungarian, preverbal particles form one word with the verb. Therefore, the edge between a verb and its particle must return to the same token it started from (see Figure 2).

'The task of applying tags to each token in a sentence consecutively is called *sequential tagging*. In general, the tagger tries to assign labels to (neighbouring) tokens correctly. The well-known special cases of this task include Part-of-Speech tagging, Named-Entity Recognition (NER) and

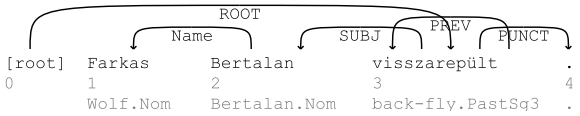


Figure 2: The visualisation of the dependency-like parsing of the sentence *Farkas Bertalan visszarepült.* 'Bertalan Farkas flew back.'. The example shows how our tool visualizes a loop. The edge `PREV` starts from the verb and ends in its verbal particle. In some languages, for example in Hungarian, the verbal particle and the verb often form one word, therefore the edge `PREV` must point to the same token it started from: the edge `PREV` connects the verb *repült* 'flew' and the verbal particle *vissza* 'back', but as they form one word the edge must form a loop.

chunking. In the latter, two *IOB tags* are used to determine a *well-formed one level bracketing* on the text.' (Indig and Endrédy, 2018)

Moreover, in languages, that do not have enough training data yet, or lack the specific programs to train a parser to do real syntactic parsing, chunking provides results which could substitute the parsing task in scenarios where only specific information is needed from the text. It is also essential for a linguist to see the specific examples and their frequencies in order to refine existing theories and recover annotation errors in the training set, which peculiarities have their own interesting aspects but cannot be harmonized with the automatic chunking process and its applications.

Figure 3 shows the visualisation of the chunking of a problematic Hungarian sentence. In Hungarian, noun phrases may end with an adjective even when followed by a noun that starts another noun phrase, despite the fact that the regular order of these categories within an NP is exactly `ADJ + NOUN`. Moreover, most of the adjectives can perfectly behave like a noun in the sentence. This is the reason why the chunking of the sentence in Figure 3 is not trivial: the word *cicaimádó* 'kitten-loving' is an adjective, but may bear the POS-tag `NOUN` because it often happens to behave like a noun in the sentences. However, in this case, this token is a true adjective being the modifier of the noun *író* 'writer'.

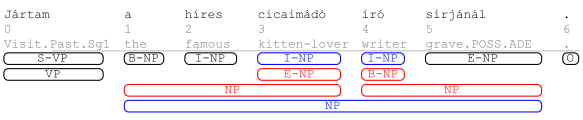


Figure 3: The visualisation of an NP-chunked sentence. The four lines under the sentence indicate two different chunking results in two forms: the first two lines show the differences in the `IOB` encoding and the last two lines show the differences in the chunks. The first and the last (blue) lines are the gold standards, and the middle ones (red) are the automatic results.

In machine translation, matching the corresponding words of the target and the source sentence is crucial for later steps. This task is called *alignment*. All of the source and target tokens must have their corresponding pair, but the order of the words can be different and multiple tokens can be

assigned to one and vice versa. The visual comparison of the gold standard and the automatic alignment (e.g. using different colours) is crucial to analyse errors in the input data and in the algorithm as well, in order to improve an aligning. In Figure 4, it can be seen how our tool visualises an alignment.

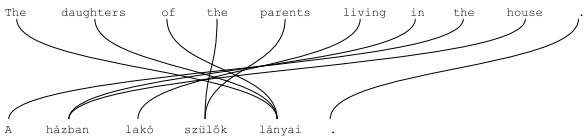


Figure 4: The visualisation of a sentence alignment. Some tokens have multiple corresponding pairs and vice versa.

Apart from the actual drawing and comparing, our program has an impressive feature set on the searching and filtering of the instances. One can filter by all the tokens and edge properties which can be a big help for the analyst to find, count, classify and correct errors. One can choose to create a visualized version on specific sentences, which matches some user-defined criteria in order to view a specific type of errors or sentences one by one or directly visualise the output of some parser like *Spacy* does.

6. Conclusion and Future Work

In this paper, we introduced a new NLP-centric visualisation and visual comparison library in Python, which adopts all the good features of the existing similar implementations while adding some novel key features so far not available in any of the current tools. Our idea was to keep the toolbox-philosophy in mind, write well-defined general modules and do not reimplement what is known to work. We also wanted to leave as much freedom as possible for the user to interact, (even dynamically) customise or extend the program for his or her needs even for non-standard formats or in-development purposes.

In the future, we want to use this program for multiple purposes, which involve interactive visual comparison and dynamic interactive visualisations. These applications will not only be interesting on their own but will also demonstrate the power of this library. We hope this program will be as useful for others as it is for us. The code is licensed under the LGPL 3.0 and is available at `https://github.com/ppke-nlpg/whats-wrong-python`.

7. References

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st edition.

Bokeh Development Team, (2014). *Bokeh: Python library for interactive visualization*.

Bostock, M., Ogievetsky, V., and Heer, J. (2011). D3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, December.

T. Mark Ellison, editor. (1997). *Proceedings of the 1997 Meeting of the ACL Special Interest Group in Natural Language Learning: Computational Natural Language Learning, CoNLL'97*. ACL.

Gansner, E. R. and North, S. C. (2000). An open graph visualization system and its applications to software engineering. *SOFTWARE - PRACTICE AND EXPERIENCE*, 30(11):1203–1233.

Halliday, M. A. K. (1968). Notes on Transitivity and Theme in English: Part 3. *Journal of Linguistics*, 4(2):179–215.

Honnibal, M. and Johnson, M. (2015). An Improved Non-monotonic Transition System for Dependency Parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal, September. Association for Computational Linguistics.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95.

Ide, N., Calzolari, N., Eckle-Kohler, J., Gibbon, D., Hellmann, S., Lee, K., Nivre, J., and Romary, L. (2017). Community Standards for Linguistically-Annotated Resources. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 113–165. Springer.

Indig, B. and Endrédy, I., (2018). *Gut, Besser, Chunker - Selecting the best models for text chunking with voting*. Springer International Publishing, Cham. In press.

Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovvář, V., Michelfeit, J., Rychlý, P., and Suchomel, V. (2014). The Sketch Engine: ten years on. *Lexicography*, pages 7–36.

Nilsson, J. and Nivre, J. (2008). MaltEval: an Evaluation and Visualization Tool for Dependency Parsing. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Tjong Kim Sang, E. F. and Buchholz, S. (2000). Introduction to the CoNLL-2000 Shared Task: Chunking. In *Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning - Volume 7*, CoNLL '00, pages 127–132, Stroudsburg, PA, USA. Association for Computational Linguistics.

8. Appendix

To demonstrate the main – visible – differences between the JAVA version of *What's Wrong With My NLP?* and our Python implementation, we captured screenshots of the two applications during operation. As can be seen in Figure 5, the old version has an unpleasant, old-style GUI with numerous different windows often not being resizeable (for example the window named *Show Properties*). One of the biggest problems with this version is its clumsiness: the various windows are handled as separate units, therefore, for example, it is difficult to move them to the foreground one by one when they are covered by another window. Our implementation (see Figure 6), on the other hand, comes with a modern, clean GUI with easy to use sections grouped into tabs.

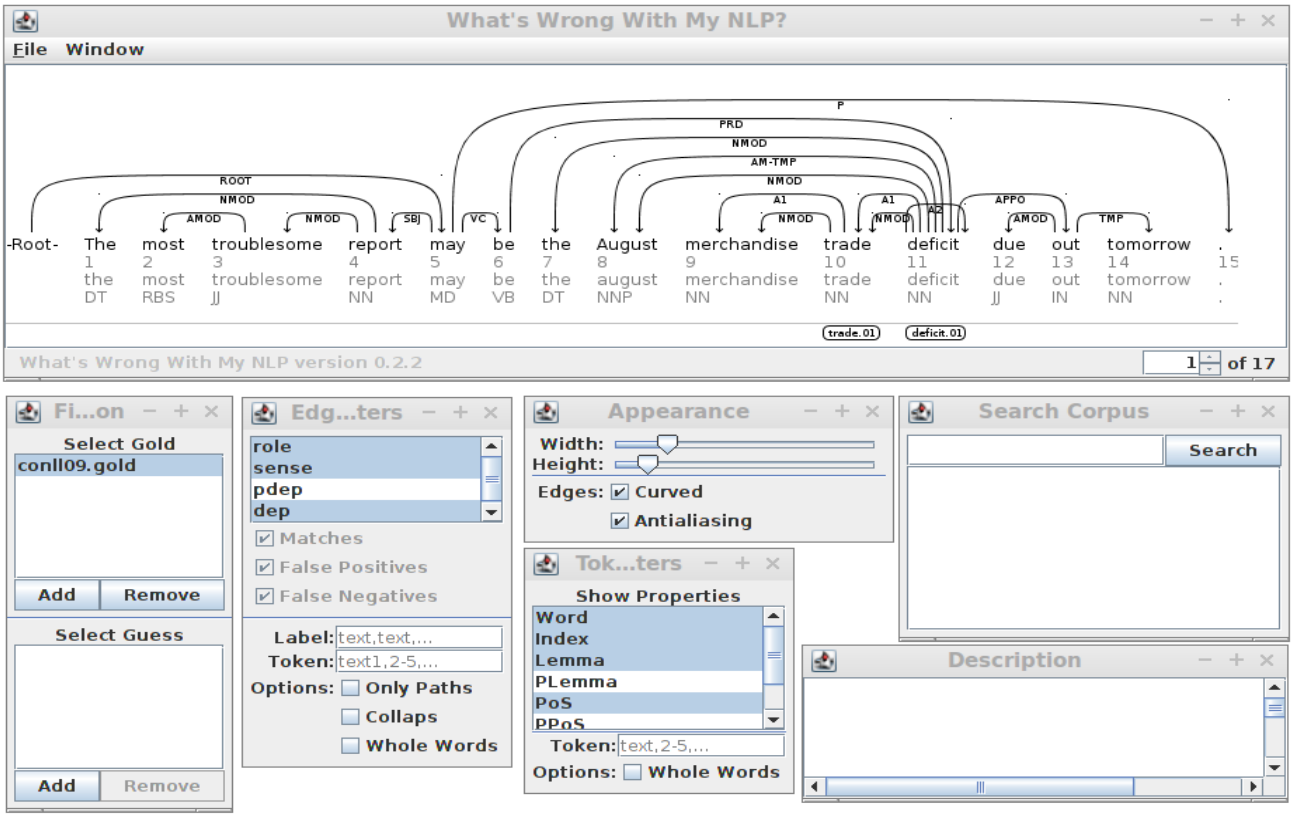


Figure 5: Screenshot of the JAVA version of *What's Wrong With My NLP?*

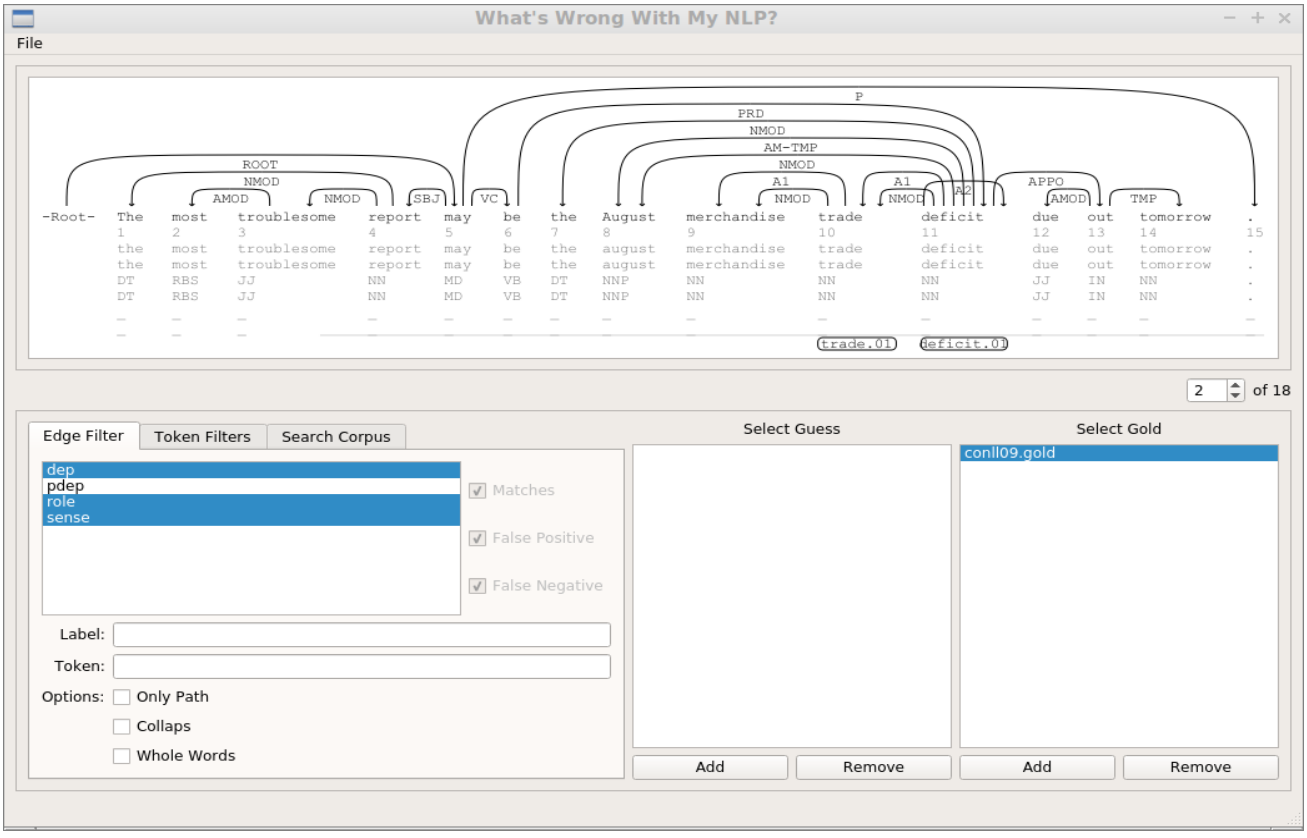


Figure 6: Screenshot of the Python version of *What's Wrong With My NLP?*

ScholarGraph :a Chinese Knowledge Graph of Chinese Scholars

Shuo Wang, Zehui Hao, Xiaofeng Meng, Qiuyue Wang
Renmin University of China, Hebei University
Beijing, Baoding in Hebei
shuowang@ruc.edu.cn, jane0331@126.com, xfmeng@ruc.edu.cn, qiuyuew@ruc.edu.cn

Abstract
Scholars and their academic information are widely distributed on the Web. Integrating these information and making association between them can play a catalytic role in academic evaluation and research. Since 2008, Web and Mobile Data Management Laboratory (WAMDM) in Renmin University of China began to collect Chinese literatures in more than 20 academic domains, and build a data integration system called ScholarSpace to automatically integrate the relevant chinese academic information from the chinese scholars and science. Focusing on the chinese scholars, ScholarSpace can give you an academic portrait about a chinese scholar with the form of knowledge graph. So the ScholarSpace can be tranformed into a knowledge graph called ScholarGraph. It includes the scholar information such as the affiliation, publications, teacher-student relationship, etc. ScholarGraph is a subset of the whole knowledge graph generated from the ScholarSpace and is published on the web page of WAMDM. ScholarGraph consists of more than 10,000,000 triples, including more than 9,000,000 entities and 6 relations. It can support the search and query about portrait of Chinese scholars and other relevant applications.

Keywords: knowledge graph construction,chinese scholars, entity, entity relation

1. Introduction

On the Web, many information about any person such as working place, living addresss, friends, personal journey can be searched online. The same things happened in the academia. The academic information related to a scholar are already available on the Web, but they are not effectively linked together. As shown in Figure 1, the relations are actually existing among scholars, journals and academic institutions. In fact, these relations are a small part of the reality. So it is necessary to build an integration system for the academic field.

The main character of the academia information is centered on scholars. Because the academic actions are completed by scholars. For example, a professor can publish a paper in a journal or a coference, supervise postgraduate students, cooperate with other scholars, apply for fundings, etc. The relations corresponding to these actions are generated by scholars.



Figure 1: Example of scholars' information.

So the data integration system needs to be focused on scholars. According to DBLP, we firstly built a Chinese data integration system – CDBLP – to collect Chinese literature information of the computer science community in China. Based on CDBLP[1], we expanded the collection from a single domain of computer science (Chen Wei et al., 2011) to 25 domains including education, management, archaeology, economics, etc. On this basis, we propose ScholarSpace system which was constructed for Chinese scholars to support Chinese search, Chinese query, Chinese journals' evaluation, etc. There is a wealth of knowledge in these integrated data. Many entities (Professor /Ph.D. /Journal /Paper) and relations (Author /PublishedIN /Affiliation /Advisor) are shown in Figure 1. So a knowledge graph – ScholarGraph – is generated from ScholarSpace. We select a partial data from ScholarSpace and transform them into SPO triples to represent the knowledge.

In Section 2, we focus on the structure design and function implemention of ScholarSpace system. We present the transformation from the ScholarSpace to the ScholarGraph in Section 3. We show the statistic and examples about the application of ScholarGraph in Section 4. At last, the conclusion and perspectives about the ScholarGraph is described in details.

2. Data Collection and Integration

The data collection and integration is the first step for generating a knowledge graph (Cowie and Lehnert, 1996 ; Zhao Jun et al., 2011). As mentioned above, the ScholarSpace system is so important to ScholarGraph that we can not ignore the introduction about it.

2.1 ScholarSpace Structure

As shown in Figure2, the ScholarSpace system is divided into four levels by the logical design that describes the data processing flow from the Web source to the user interface.

At the bottom level, data sources include semi-structured data or unstructured data such as personal homepage, research institutes' web pages, coferences' web pages, journals' web pages.

Above the bottom level, the data integration function model aims at designing the configuration files for each

Corresponding author is Xiaofeng Meng.
[1] http://www.c-dblp.cn/

kind of data sources that can collect the data from the Web to the database.

On the third level, data processing maintains three function parts as follow. Data cleaning needs to remove the information that is not relevant to the literature (such as documents of Call For Papers) or duplicated records from the integrated data on the Web. Entity extraction means to extract the information about authors, journals,

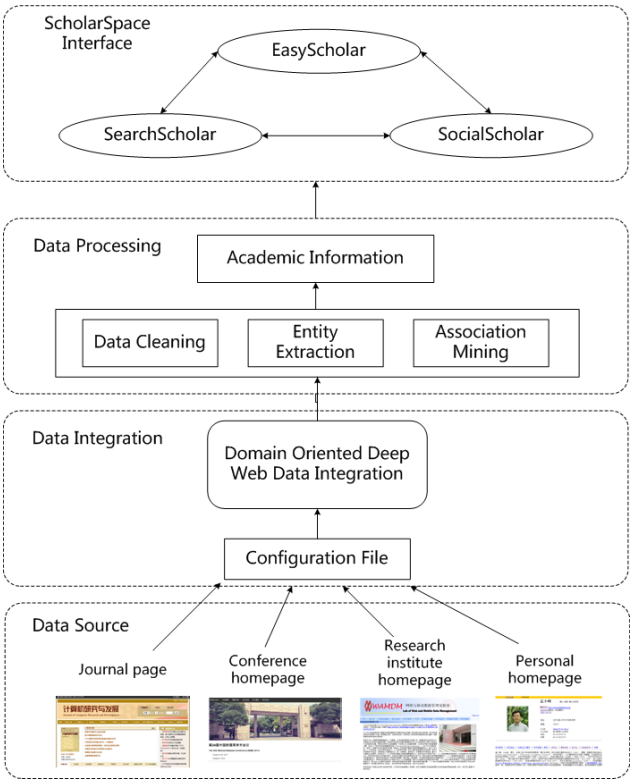


Figure 2: The logical design structure

coferences, institutes, etc. All of these objects can be described as entities with attributes and relations. Association mining can discover the associated entities and construct the relations between these entities.

On the top level, users can see three parts of ScholarSpace. The first and most important part is SearchScholar which can build the data set for the Chinese scholars and support the information query and retrieval. The second part is EasyScholar which can generate the scholar page based on SearchSholar and provide the association mining based on the author. The third part is named as SocialScholar. It builds a science community for academic communication and dissemination based on the SearchScholar and EasyScholar. The three parts support and promote each other.

2.2 ScholarSpace Storage

The ScholarSpace has more than thirty kinds of tables to store the information about the authors, the journals, the papers, etc. The kernel tables include c_papers, authortab, coauthor, disambiguation, affiliation, author_research_domain, clcs_intro, journals, projects, student, etc. The extraction and integration processing are shown in Figure 3. The author "杜治娟" is disambiguated through the table disambiguation and inserted properly into the table authortab. The paper "大数据融合研究：问题与挑战" is stored in the table c_papers. The institution "中国人民大学信息学院" is checked with the table affiliation and determined whether or not to insert it into the table. The project information about this paper are extracted and stored into the corresponding items of the table projects. There are more data extracted from the Web and stored into more tables for this paper.

3. ScholarGraph Construction

The ScholarGraph is a knowledge graph constructed based on ScholarSpace. SPO (subject-predicate-object) triples is a kind of decription forms for representing the human knowledge. For example, the sentence "姚明是篮球运动员" can be parsed into a triple "姚明-IsA-篮球运动员" as a SPO form in Chinese. ScholarGraph is a dataset which consists of SPO triples. As mentioned in Section 2, entities and their relations are extracted from the Web and stored into the tables of the database. So ScholarGraph system needs to transform the table into the triples describing the knowledge about the scholars. We need to manually define some extraction rules for recognizing the entities from database tables according to the values of database field and type. Firstly, the entities or properties are selected based on the rules. Secondly, the candidate entities and properties are manually filtered. Lastly, the relations between the entities are generated by the association mining method (Zen Wandan et al.,2006) and inspected manually.

In Figure 4, the transformation are shown from one or more tables to the triples. The paper "大数据融合研究：问题与挑战" is selected from the table c_papers. The properties of this paper include Title, Keywords, Keywords_en, Source_site, Year, CLC, etc. The triples about author "杜治娟" are generated from the table disambiguation and the table c_papers to describe the relation "Write" and the property "Name". The table affiliation and disambiguation can make up the triple for the relation "Affiliated".

4. Description and Application

ScholarGraph datasets are set up through processing databases as mentioned obove. The details of the datasets are described in in Section 4.1 and its application can be shown in Section 4.

4.1.1 Details of ScholarGraph Datasets

ScholarGraph datasets consist of 7 subsets for each scientific domain such as archaeology, computer,education, economy, geography, management and physics. The 7 subsets include Author, Affiliation, Journals,Name, Papers, and PublisheIN that can give the portraits for the Chinese scholars in one domain. We just select the necessary properties and relations from the database to construct the triples as shown in Table 1.

The value in the fourth column means that the triple does not represent relation but property. The statistics of the triple numbers are listed in Table 2. We select the data that we collected from 2013 to 2016. In fact, ScholarSpace began to integrate the Web data from 2000 and collected all papers published in the selected journals



Figure 3: Integration and storage processing in ScholarSpace

Subset	Subject Entity	Relation/Property	Object Entity/Value
Author	AuthorID	Write	PaperID
Affiliation	AuthorID	Affilated	Value
Journals	JournalID	Name	Value
	JournalID	BeginYear	Value
	JournalID	Cnki_code	Value
Name	AuthorID	Name	Value
Papers	PaperID	PaperTitle	Value
	PaperID	PaperKeywords	Value
	PaperID	PaperKeywords_en	Value
	PaperID	PaperYear	Value
	PaperID	PaperSource_site	Value
	PaperID	PaperCLC	Value
PublishedIN	PaperID	PublishedIn	JournalID
Advisors	AuthorID	GuideBy	AuthorID

Table 1: The patterns of the datasets

from the begining of the publication. So the complete dataset for the same journals has 10602497 triples. Table 2 just show 12 percent of the complete set published on the webpage of WAMDM. We do not mention the advisor subset because we just generate the teacher-student relation for computer science domain. And there are lots of authors that are unable to find their teachers or students. A data portrait for a Chinese scholar can be constructed from this dataset. For example, a scholar has relations with papers, journals and institutions. A scholar, an author entity to published papers in journals, includes relation « Write » as follows.

<http ://www.c-dblp.cn/computer/**AuthorID**/ 25447-0>

<http ://www.c-dblp.cn/computer/**Write**>

<http ://www.c-dblp.cn/computer/**PaperID**/99955>

And a paper entity can have a property triple to represent the title of this paper based on URI forms as follows.

Triple Number	Author	Journals	Name/ Affiliation	Papers	PublisheIN
Archaeology	5998	24	3661	22248	3708
Computer	39622	33	21025	84905	14128
Economics	69899	219	36167	301806	50301
Education	38371	108	20421	178620	29770
Geography	7006	18	3749	27817	4636
Management	58018	87	29715	189234	31539
Physics	94213	90	41237	151363	24802
Total	1584558				

Table 2: The statistics of the datasets

<http ://www.c-dblp.cn/computer/**PaperID**/99955>

<http ://www.c-dblp.cn/computer/Property/**PaperTitle**>

« 支持QoS保障的可信服务组合调度算法 »

A paper entity can also have a relation triple to represent the publishment of this paper as follow.

<http ://www.c-dblp.cn/computer/**PaperID**/99955>

<http ://www.c-dblp.cn/computer/**PublishedIn**>

<http ://www.c-dblp.cn/computer/**JournalID**/9>

As mentioned above, we got the information about the authors of a paper, the journal in which a paper is published, the institution of the author, etc. Further more, we infered the co-author, the interest domains of a scholar, the research topic of a scholar, etc. through the knowledge graph. So we can develop many applications with ScholarGraph (Li Hehan et al., 2015).

4.2 Applications of ScholarGraph

The application of ScholarGraph is constructed based on the knowledge graph of the scholar information. We have built an interactive interface for ScholarGraph through visualization tools. ScholarExplorer[2] is a webpage which can browse on the homepage WAMDM. In Figure 5, when you search the name "孟小峰", the results in five areas are shown on the webpage. In the middle of the page, a graph is centered by this author and links the author to the papers, journals, coauthors and institutions. In the left area, the students or teachers of this author are listed. In the right upper area, the academic statistics of this author and the properties are shown in text form. In the right lower area, user can find the word cloud of research domains about this author. In the bottom of the page, the publication years of the corresponding topic can be found. The highlight Chinese characters mean big data which is the topic, and the highlight spots are the papers related to the big data topic in the middle area. At the bottom, the highlight belts are the publication years of these papaers.

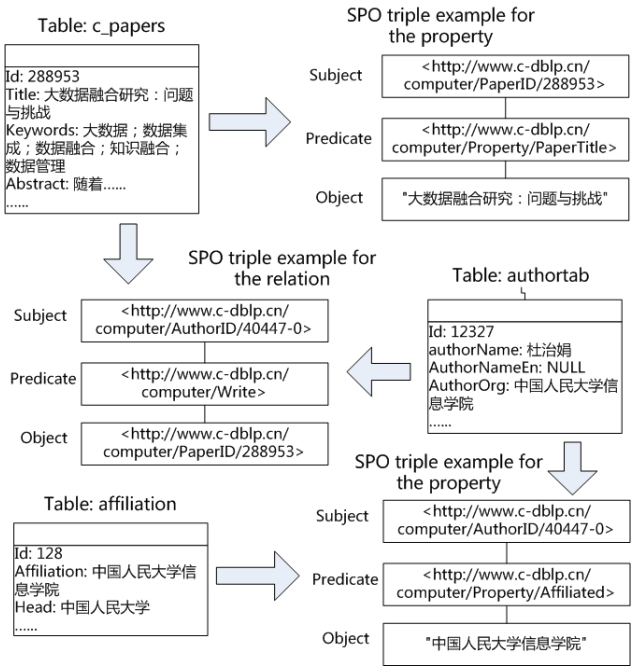


Figure 4: Examples for SPO triples generation

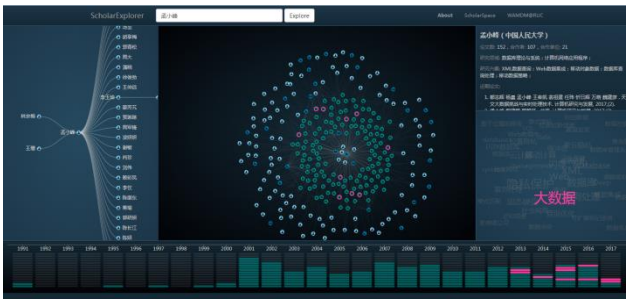


Figure 5: Application example of ScholarGraph - ScholarExplorer

Further more, we have taken adventage of the knowledge graph for the experts recommendation[3] when a paper or an item needs to review or check up. A user can input the information about the paper or the item, and the recommendation system can compute the suitability of experts in the related domain based on the ScholarGraph. Due to the word limit, we don't describe in details.

5. Conclusion and Future Work

ScholarGraph is built by the knowledge graph technique and firstly focuses on the Chinese scholars. We developed an automated system to integrate the Chinese academic information from Web and transform them into the knowledge graph for improving the application . The accuracy of ScholarGraph is relied on the manual evaluation. The triples are randomly selected to human judges and made the judgements. The average accuracy of thirteen relations or properties is 97.76%. We need to extend the domains and add the relations that exist in ScholarSpace but not in ScholarGraph in future.

6. Acknowledgements

This research was partially supported by the grants from the Natural Science Foundation of China (No. 61532010, 61379050, 61532016, 91646203, 61762082); the National Key Research and Development Program of China (No. 2016YFB1000602, 2016YFB1000603); the Fundamental Research Funds for the Central Universities, the Research Funds of Renmin University(No. 11XNL010); and the Science and Technology Opening up Cooperation project of Henan Province (172106000077) ; the Opening Project of State Key Laboratory of Digital Publishing Technology of Founder Group ; the Natural Science Foundation of Hebei Province (No. F2017201020）.

7. Bibliographical References

Chen Wei, Wang Zhongyuan, Yang Sen, Zhang Peng, Meng Xiaofeng. (2011). ScholarSpace : Computer Oriented Academic Space. Journal of Computer Research and Development, 48(S3), 395-399(in Chinese).

Cowie, J., & Lehnert, W. (1996). Information extraction : Natural language processing. Communications of The ACM, 39(1), 80-91.

Zhao Jun, Liu Kang, Zhou Guangyou, et al. (2011). Open information extraction. Journal of Chinese Information Processing, 25(6), 98-110(in Chinese).

Zeng Wandan, Zhou Xubo, Dai Bo, Chang Guiran, Li Chunping. (2006). Matrix Algorithm for Mining Association Rules. Computer Engineering, (02), 45-47(in Chinese).

Li Hehan, Meng Xiaofeng, Zou lei. (2015). Keyword query method oriented to the knowledge base of ScholarSpace. Journal on Communications, 36(12), 28-36(in Chinese).

[2] http://www.c-dblp.cn/scholarexplorer/

[3] http://www.c-dblp.cn/recommend.php

Enriching Frame Representations with Distributionally Induced Senses

Stefano Faralli[†], **Alexander Panchenko**[‡], **Chris Biemann**[‡], **Simone Paolo Ponzetto**[†]
[†]Data and Web Science Group, School of Business Informatics and Mathematics, Universität Mannheim, Germany
[‡] Language Technology Group, Department of Informatics, Universität Hamburg, Germany
{stefano,simone}@informatik.uni-mannheim.de
{panchenko,biemann}@informatik.uni-hamburg.de

Abstract

We introduce a new lexical resource that enriches the Framester knowledge graph, which links Framnet, WordNet, VerbNet and other resources, with semantic features from text corpora. These features are extracted from distributionally induced sense inventories and subsequently linked to the manually-constructed frame representations to boost the performance of frame disambiguation in context. Since Framester is a frame-based knowledge graph, which enables full-fledged OWL querying and reasoning, our resource paves the way for the development of novel, deeper semantic-aware applications that could benefit from the combination of knowledge from text and complex symbolic representations of events and participants. Together with the resource we also provide the software we developed for the evaluation in the task of Word Frame Disambiguation (WFD).

Keywords: distributional semantics, linked open data, frame semantics.

1. Introduction

Recent years have witnessed an impressive amount of work on the automatic construction of wide-coverage knowledge resources. Web-scale information extraction systems like NELL (Carlson et al., 2010) or Knowledge Vault (Dong et al., 2014) can acquire massive amounts of machine-readable knowledge from the Web, whereas projects like DBpedia (Bizer et al., 2009), YAGO (Rebele et al., 2016) or BabelNet (Navigli and Ponzetto, 2012) have turned collaboratively-generated content into large knowledge bases. However, all of these resources are entity-centric in that they are primarily built around the notion of *entities*, as either provided by an external resource (e.g., Wikipedia pages) or automatically discovered from text (e.g., by clustering entity mentions). The entities are most commonly represented by nouns, noun phrases, and named entities. Entities lie at the heart of the Semantic Web and are central to enable semantic technologies on a large scale (Dietz et al., 2017). Besides, they also provide the foundation for more complex semantic representations like event templates or frames (Fillmore, 1968), which are in the focus of our interest in this paper.

Recent work looked at ways to populate the Linked Open Data (LOD) cloud with wide-coverage information about semantic frames on the basis of Framester (Gangemi et al., 2016a), a frame-centric resource that is meant to act as a hub between several other linguistic resources that are already part of the LOD cloud. In parallel, researchers looked at ways to combine knowledge graphs of this kind with distributional semantics to include human-readable meaning representations based on semantic vector spaces within the Semantic Web ecosystem (Faralli et al., 2016). Consequently, in this paper we bring these two lines of research together and present a new resource that combines a wide-coverage symbolic repository of frames with human-readable distributional semantic representations from text.

In Table 3, we show an excerpt from a distributional-based sense inventory, a proto-conceptualization (PCZ) proposed by Faralli et al. (2016) and further developed by Biemann et al. (2018). The senses in this resource are induced automatically from text via clustering of graphs of semantically related words and linked to BabelNet and WordNet lexical resources. In particular, in the resource, each sense (e.g., *read#VB#1*) is defined by its related senses (*quote#VB#0*, *recite#VB#0*, and so on) which have been observed to co-occur with the sense. Sense inventories with related senses as well as other features, such as weighted hypernymy senses and context clues, can provide rich and human-readable knowledge to disambiguate word meanings in context, e.g., the meaning for the verb *to read* as 'read aloud' versus 'make sense' as in:

When we *hear* someone READ a text, our *understanding* of what we *hear* is usually spontaneous – the rules by which we *interpret* meaning.

Here, the reference to the 'Reading_aloud' frame from FrameNet[1] could be triggered on the basis of the occurrences of the verbs in the sentence: *hear*, *understanding* and *interpret*. Such connections, in turn, could be provided by a hybrid resource where distributional representations from text have been explicitly linked to semantic knowledge repositories, since hybrid resources of this kind have been shown in the past to improve performance on lexical understanding (Panchenko et al., 2017) as well as taxonomy learning and cleaning (Faralli et al., 2017).

In this paper, we bridge the gap between distributional and frame semantics by linking distributional semantic representations to Framester, a knowledge graph that acts as a hub between resources like FrameNet, BabelNet and DBpedia, among others. As a result of this, we introduce a new lexical resource that enriches the Framester knowledge graph with distributional features extracted from text, and show how this hybrid resource yields better results on the task of recognizing frames in running text.

Joining distributional and frame semantics builds upon and

[1] https://framenet.icsi.berkeley.edu/

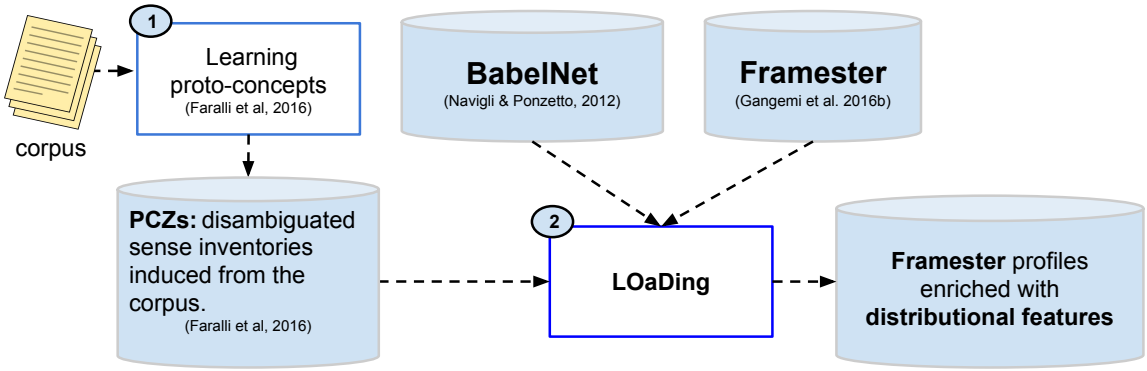


Figure 1: The overall resource construction workflow of our approach: distributionally induced sense representations extracted from a corpus (PCZ) are used to enrich the Framester symbolic knowledge graph.

BabelNet Synset ID	English Word Senses	Gloss
bn:00085007v	quote, cite	'To repeat someone's exact words.', 'To quote; to repeat, as a passage from a book, or the words of another.'
bn:00090740v	observe, mention, note, remark	'She observed that his presentation took up too much time.', 'They noted that it was a fine day to go sailing.'

Table 1: Excerpt of two BabelNet synsets.

Frame ID	Related BabelNet Synsets
Evidence	bn:00085007v, bn:00084633v, ...
Telling	bn:00085007v, bn:00083488v, ...
Communication	bn:00085007v, bn:00090740v, ...

Table 2: Excerpt of the Framester TransX profile.

extends our framework for combining symbolic and statistical meaning representations (Biemann et al., 2018): the main objective of this line of research is to 'join forces' across heterogeneous knowledge and semantic models in order to mutually enrich them and combine the strengths of the lexicographic tradition that describes linguistic information manually with the coverage and versatility of the corpus and data-driven approaches that derive meaning representations directly from the data. Bringing together the 'best of both worlds' has the potential to combine the benefits of wide-coverage symbolic (Gangemi et al., 2016b) and statistical (Chen et al., 2014) semantics for frame parsing, which, in turn, can be exploited for many different applications ranging from sentiment analysis (Recupero et al., 2015) all the way to content-based recommendations (De Clercq et al., 2014).

The contributions of this paper are the following ones:

1. We present the *LOaDing* lexical resource, an extension of Framester that adds distributional-based features (Faralli et al., 2016; Biemann et al., 2018) to frame representations (see Section 2.);

2. We evaluate the resource in the task of Word Frame Disambiguation (WFD), i.e., on the identification of frames in context (Gangemi et al., 2016b), and show significant improvements over the original Framester frame representations (see Section 3.).

Both resources and software produced in this work are available under a CC-BY 4.0 license.[2]

2. Enriching Framester with Symbolic Distributional Sense Representations

In Figure 1 we show the diagram describing the workflow used to add distributional features to the Framester profiles to enable a better word frame disambiguation.

2.1. Resources and Datasets

Our approach makes use of three linguistic resources:

BabelNet (Navigli and Ponzetto, 2012): a multilingual encyclopedic dictionary that connects concepts and named entities in a very large network of semantic relations (see Table 1).

Framester (Gangemi et al., 2016a): a linguistic LOD hub that provides links from FrameNet's frames to semantically related BabelNet senses (see Table 2). The current version consists of six different profiles: *Base* contains only manually curated links, whereas the other five (*DirectX*, *FrameBase*, *Fprofile*, *TransX* and *XWFN*) are automatically built extensions or subsets of the *Base* profile.

Proto-Conceptualizations (PCZs) (Faralli et al., 2016): a fully disambiguated sense inventory automatically induced from text, providing human-readable distributional semantic representations. In this resource, each sense is provided with (see Table 3): a) a weighted list of semantically-related and hypernymy senses; b) links to existing knowledge bases (i.e., BabelNet); c) context clues used to disambiguate the senses in the context.

[2] http://web.informatik.uni-mannheim.de/joint/

PCZ Sense ID	BabelNet Synset ID	Related Senses	Context Clues
quote#VB#0	bn:00085007v	cite#NN#1[1.0], interview#VB#0[0.8], mention#VB#1[0.7], publish#VB#0[0.6], review#VB#0[0.6], ...	in#IN#pcomp[24799.3], Register#NP#prep_on[21282.9], Track#NP#-vmod[16808.9], ...
mention#VB#1	bn:00090740v	attest#VB#0[1.0], describe#VB#0[0.8], document#VB#0[0.7], quote#VB#0[0.6], ...	Register#NP#prep_on[45477.7], Track#NP#-vmod[35850.4], say#VB#prepc_as[17041.8], ...

Table 3: Excerpt from a proto-conceptualization (PCZ) of two distributional-based senses. Numbers behind related senses are normalized similarity scores, numbers behind context clues are association scores.

Frame ID	Related BabelNet Synsets	Related PCZ Senses
Evidence	bn:00084633v[29.0], bn:00085007v[13.0], ...	quote#VB#0, ...
Telling	bn:00085007v[12.0], bn:00083488v[8.0], ...	quote#VB#0, ...
Communication	bn:00090740v[18.0], bn:00085007v[15.0], ...	mention#VB#1, quote#VB#0, ...

Table 4: Excerpt of a *LOaDing* TransX profile. Numbers behind related synsets bs are the distributional semantics-based weights $w(F, bs)$ we assign in Step 2.4 of our LOaDing approach.

2.2. Combination of Resources

The goal of our method is a combination of the three resources mentioned in the previous section as illustrated in Table 4. This is made possible by using BabelNet as a pivot, since Framester provides links from FrameNet to BabelNet (Table 2) while our PCZs are also linked to BabelNet (Table 3). As a first result of such a combination we created *LOaDing*, an extension of Framester's profiles where we provide weights for each frame-related BabelNet synset. We show in Table 4 an excerpt of the resulting extended Framester's TransX profile shown in Table 2.

The enriched frame representations are built in two main steps as illustrated in Figure 1:

Step 1: Learning a proto-conceptualization (PCZ). We apply the methodology from Biemann et al. (2018), to produce disambiguated sense inventories with entries for nouns and verbs.

Step 2: LOaDing a PCZ into Framester. For each frame F, e.g. *Communication*, and related BabelNet sysnet bs, e.g. *bn:00085007v*, we assign a weight $w(F, bs)$. In order to compute the weight, we create three bag-of-word representations for each of the three resources listed in the previous section:

1. $B_{bn} = BoW(bs)$ to represent bs including the counts for all the content words from the synset word senses and the glosses;

2. From the PCZ we collect all the entries $PCZ(bs)$ matching bs, for instance PCZ(*bn:00085007v*) $= \{$*quote#VB#0*, ... $\}$, and create a bag of words $B_z = BoW(PCZ(bs))$ by collecting all the weights of content words from the senses $s \in PCZ(bs)$ and the corresponding related senses.

3. We create a BoW from B_f by collecting all the content words from the description of the frame F.

4. Finally, we compute a relatedness score between a frame F and a BabelNet synset bs:

$$w(F, bs) = \sum_{w \in I} B_z.c(w) \times (B_f(F).c(w) + B_{bn}.c(w)),$$

where $I = B_z(bs) \cap (B_{bn}(bs) \cup B_f(F))$ and $A.c(w)$ equals the number of occurrences of the word w in A. For instance, with respect to the excerpts of Tables 3 and 4 we obtain $w($*Communication*, *bn:00085007v*$) = 15.0$.

3. Using the Enriched Representations for Word Frame Disambiguation

We evaluate our extensions of Framester profiles following the experimental setting of Gangemi et al. (2016b), and compare the extended and the original profiles in a task of Word Frame Disambiguation (WFD).

3.1. Dataset: FrameNet Full Text Documents

To create a silver standard we processed all 108 documents from the FrameNet 1.7 dataset (Baker et al., 1998) with BabelFy (Moro et al., 2014)[3]. By combining the original frame annotations with the automatically generated entity links we collected a total of 81,706 annotations, which we use in our experimental setting as a silver standard.

3.2. Word Frame Disambiguation

Following the WFD approach described in Gangemi et al. (2016b) we implemented a simple word frame disambiguator, where for each provided annotation in our silver standard we try to predict a frame label only on the basis of the BabelNet synsets generated through BabelFy. In order to provide the most suitable frame label F for the provided BabelNet synset label bs:

1. Creating the set of candidate frames $CF(bs)$, by collecting all the frame IDs for which bs is a semantically related BabelNet synsets (e.g., $CF($*bn:00085007v*$)=$ $\{$*Evidence*,*Telling*, *Communication*$\}$ see Table 4);

[3] http://babelfy.org

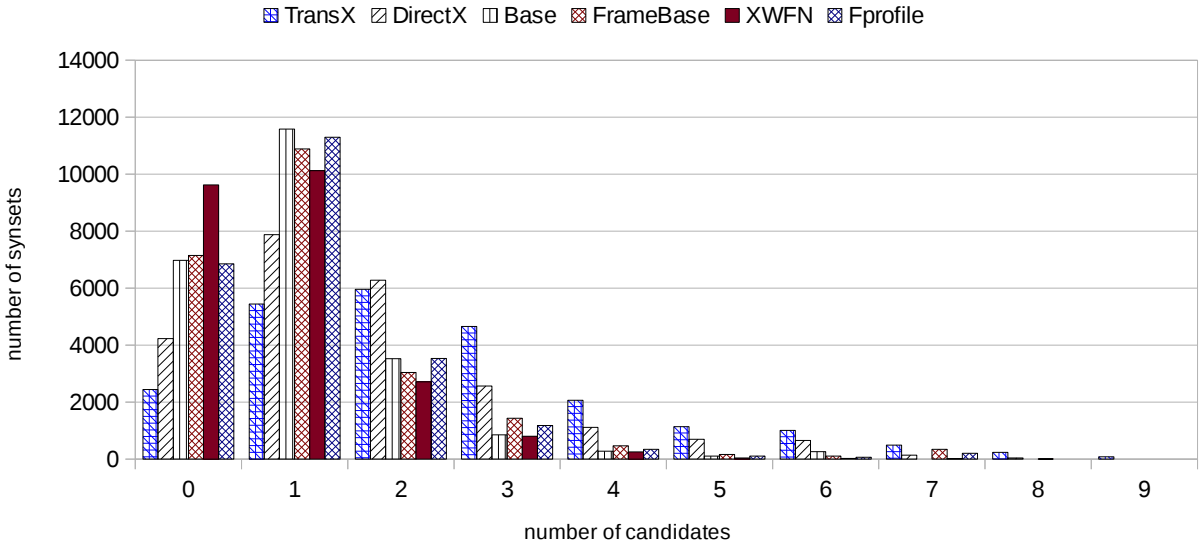


Figure 2: The distribution of synset counts per number of candidate frames (step 1 of our WFD approach, see Section 3.2.). The y axis counts the synsets bs for which $|CF(bs)| = x$.

	Base	FrameBase	XWFN	TransX	DirectX	Fprofile
oracle	67.60	67.74	57.85	77.82	73.74	68.16
original (cond)	53.13	52.73	45.44	**45.21**	46.30	53.04
wiki-n30-1400k (cond)	53.23	52.82	45.42	44.75	**46.58**	53.46
wiki-n30-1400k (inv)	**55.52**	**55.06**	**46.73**	28.24	45.39	**55.69**

Table 5: F1 performances on Word Frame Disambiguation across profiles and ranking methodologies.

2. Ranking all the candidate frames $f \in CF(bs)$ by computing the following scoring functions:

   - the score is equal to the conditional probability of a frame given a synset (*cond*):

   $$score_{cond}(f, bs) = \frac{w(f, bs)}{\sum_{b \in relatedSynsets(f)} w(f, b)};$$

   - the score is equal to the number of related synsets divided by the weight of the synsets (*inv*):

   $$score_{inv}(f, bs) = \frac{|relatedSynsets(f)|}{w(f, bs)}.$$

   Such a scoring function promotes the candidate frame that relates with the highest number of synsets $|relatedSynsets(f)|$, and it also penalizes the selection of candidates which are triggered by synsets having higher weights, in other words promoting candidates triggered by synsets in the 'long tail';

3. Finally, we select the candidate frame with the highest score.

We experimented with a variety of scoring functions based on weights $w(F, bs)$ and $|relatedSynsets(F)|$, and present only the two best performing ones here.
In Figure 2 we draw the distribution of the counts of synsets across the number of frames that are triggered in the phase of candidate selection (step 1 of our WFD approach), i.e., in the XWFN profile, we counted ten thousand synsets bs for which $|CF(bs)| = 1$.

3.3. Results

In Table 5, we show the resulting word frame disambiguation performances in terms of the F-score and compare across profiles and ranking methodologies. To better understand the limits of our 'silver standard' and the limits of ranking methodologies we introduced a so called 'oracle', which selects the labeled frame if present in the list of selected candidates independently from its position in the ranking. The results indicate that:

- For Base, FrameBase, XWFN and Fprofile profiles, adding distributional-semantic-based features lead to 1 to 2 points improvements of F1 in the WFD task;
- Our approach lowers the performance of DirectX profile and even drastically lowers the performances of the TransX profile;
- Overall, we noticed that the former two profiles are very dense by means of related senses per frame and lead to a potential high recall (see the oracle performances on the two profiles at 73.74%, 77.82% respectively) while introducing more noisy senses;
- Finally, the presence of many frequent and noisy senses let the $score_{inv}$ function outperform other function based on weights $w(F, bs)$ and $|relatedSynsets(F)|$.

To summarize this experiment: when using our induced weighting with the $score_{inv}$ ranking, we achieve a boost on Framester profiles that showed a good performance in the

original setting, while losing performance on inferior, more noisier profiles. The overall best performance was reached by our extension of the high-quality, manually created Base profile, followed by our extension to the Fprofile.

3.4. Error analysis

An in-depth analysis of the errors made during the experiments (see Section 3.3.) by our WFD approach (see Section 3.2.) allowed us to identify the following categories of errors:

- *Misaligned annotations:* our silver standard is automatically created by collecting BabelFy annotations for a total of 81,706 annotations from the original 93,358 FrameNet 1.7 annotations;
- *Entity linking errors:* we estimated that around 15% of the 81,706 BabelFy annotations[4] assigned a wrong sense, eventually triggering the selection of a wrong frame;
- *Ranking error:* errors where an inferior ranking leads to bad performance. These are a consistent part of the WFD errors and may be improved with more complex weighting/ranking approaches;
- *Profile errors:* related to a specific profile. In this category we identify errors due to: i) the slightly difference between some frame meaning (e.g., many errors are due to the prediction of the frame *Measure duration* instead of *Calendric unit*); ii) the frequent absence of related senses so that is not possible to trigger any frame, even for correct BabelFy annotations.

4. Conclusions

In this paper, we presented *LOaDing*, a novel lexical resource that connects distributional sense representations induced from text to symbolic frame representations. Hybrid semantic representations have been shown to enable complex semantic tasks like, for instance, end-to-end taxonomy induction (Faralli et al., 2017): they could provide an additional signal to improve state-of-the-art semantic parsing (as illustrated by our running example). As we have shown for word sense disambiguation for nouns (Panchenko et al., 2017), distributional information is able to considerably alleviate the sparsity inherent in knowledge-based methods. Consequently, in future work we would like to study the contribution and the potential of our proto-conceptualizations into more complex frame-centered tasks such as verb frame induction (Vulić et al., 2017). Our vision in the longer term is to exploit hybrid statistical and symbolic approaches to go beyond the vocabulary and relations of entity- and encyclopedic-centric resources to produce novel semantic representations of event templates and frames with a large coverage.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) within the JOIN-T project.

[4] A sample of 1000 annotations has been manually annotated as correct or wrong by two annotators, and a third annotator provided a final judgment only for the annotations in disagreement.

5. Bibliographical References

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley Framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Biemann, C., Faralli, S., Panchenko, A., and Ponzetto, S. P. (2018). A framework for enriching lexical semantic resources with distributional semantics. *Natural Language Engineering*, pages 1–48.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). DBpedia – A crystallization point for the web of data. *JWS*, 7(3):154–165.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Proceedings of AAAI*, pages 1306–1313, Atlanta, GA, USA. AAAI Press.

Chen, Y. N., Wang, W. Y., and Rudnicky, A. I. (2014). Leveraging frame semantics and distributional semantics for unsupervised semantic slot induction in spoken dialogue systems. In *Proceedings of IEEE Spoken Language Technology Workshop (SLT)*, pages 584–589, South Lake Tahoe, NV, USA. IEEE.

De Clercq, O., Schuhmacher, M., Hoste, V., and Ponzetto, S. P. (2014). Exploiting FrameNet for content-based book recommendation. In *Proceedings of CBRecSys Workshop at RecSys*, pages 14–20, Foster City, CA, USA. ACM.

Dietz, L., Kotov, A., and Meij, E. (2017). Utilizing knowledge graphs in text-centric information retrieval. In *Proceedings of WSDM*, pages 815–816. ACM.

Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of KDD*, pages 601–610, New York, NY, USA. ACM.

Faralli, S., Panchenko, A., Biemann, C., and Ponzetto, S. P. (2016). Linked disambiguated distributional semantic networks. In *Proceedings of ISWC*, pages 56–64, Kobe, Japan. Springer International.

Faralli, S., Panchenko, A., Biemann, C., and Ponzetto, S. P. (2017). The ContrastMedium algorithm: Taxonomy induction from noisy knowledge graphs with just a few links. In *Proceedings of EACL*, pages 590–600, Valencia, Spain. Association for Computational Linguistics.

Fillmore, C. J. (1968). The case for case. In *Universals in Linguistic Theory*, pages 2–88. Holt, Rinehart and Winston, New York, NY, USA.

Gangemi, A., Alam, M., Asprino, L., Presutti, V., and Recupero, D. R. (2016a). Framester: A wide coverage linguistic linked data hub. In *Proceedings of EKAW*, pages 239–254, Bologna, Italy.

Gangemi, A., Alam, M., and Presutti, V. (2016b). Word frame disambiguation: Evaluating linguistic linked data on frame detection. In *Proceedings of LD4IE Workshop*

at ISWC, pages 23–31, Kobe, Japan. Springer International.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *ArtInt.*, 193:217–250.

Panchenko, A., Faralli, S., Ponzetto, S. P., and Biemann, C. (2017). Using linked disambiguated distributional networks for word sense disambiguation. In *Proceedings of SENSE Workshop at EACL*, pages 1–7, Valencia, Spain. Association for Computational Linguistics.

Rebele, T., Suchanek, F. M., Hoffart, J., Biega, J., Kuzey, E., and Weikum, G. (2016). YAGO: A multilingual knowledge base from Wikipedia, Wordnet, and Geonames. In *Proceedings of ISWC*, pages 177–185, Kobe, Japan. Springer International.

Recupero, D. R., Presutti, V., Consoli, S., Gangemi, A., and Nuzzolese, A. G. (2015). Sentilo: Frame-based sentiment analysis. *Cognitive Computation*, 7(2):211–225.

Vulić, I., Mrkšić, N., and Korhonen, A. (2017). Cross-lingual induction and transfer of verb classes based on word vector space specialisation. In *Proceedings of EMNLP*, pages 2536–2548, Copenhagen, Denmark. Association for Computational Linguistics.

An Integrated Formal Representation for Terminological and Lexical Data included in Classification Schemes

Thierry Declerck[1,2], Kseniya Egorova[3], Eileen Schnur[1]
[1]DFKI GmbH, Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
[2]Austrian Academy of Sciences, Sonnenfelsgasse 19,1010 Vienna, Austria
[3]Saint Petersburg State University of Architecture and Civil Engineering,
2-nd Krasnoarmeiskaya St. 4, 190005 St. Petersburg, Russia
declerck@dfki.de, kseniya.a.egorova@gmail.com, eileen.schnur@dfki.de

Abstract

This paper presents our work dealing with a potential application in e-lexicography: the automatized creation of specialized multilingual dictionaries from structured data, which are available in the form of comparable multilingual classification schemes or taxonomies. As starting examples, we use comparable industry classification schemes, which frequently occur in the context of stock exchanges and business reports. Initially, we planned to follow an approach based on cross-taxonomies and cross-languages string mapping to automatically detect candidate multilingual dictionary entries for this specific domain. However, the need to first transform the comparable classification schemes into a shared formal representation language in order to be able to properly align their components before implementing the algorithms for the multilingual lexicon extraction soon became apparent. We opted for the SKOS-XL vocabulary for modelling the multilingual terminological part of the comparable taxonomies and for OntoLex-Lemon for modelling the multilingual lexical entries which can be extracted from the original data. In this paper, we present the suggested modelling architecture, which demonstrates how terminological elements and lexical items can be formally integrated and explicitly cross-linked in the context of the Linguistic Linked Open Data (LLOD).

Keywords: SKOS-XL, OntoLex-Lemon, Terminology, Lexicography

1. Introduction

The topic of extracting dictionaries from raw data was discussed in the context of the recently terminated European Network of e-Lexicography (ENeL) COST Action[1], and it was also the motto of the fifth biennial eLex conference[2] called "eLex 2017: Lexicography from scratch". The work presented in this paper takes its source within this context, meaning in detail that we aim to investigate the automated extraction of domain-specific multilingual lexicons from comparable classification schemes or taxonomies.

As a dataset for our investigation, we selected two comparable industry classification schemes that are used in various stock exchanges like Euronext, the New York Stock Exchange, the Toronto Stock Exchange, etc., or within business reports of companies. The decision to use these sources was also inspired by former ontology mapping work applied to this kind of data, as described in (Gromann and Declerck, 2014), building thus on related previous work. We planned to follow an approach based on cross-taxonomies and cross-languages string mappings to automatically detect candidate multilingual dictionary entries for this specific domain.

After an in-depth analysis of both data sources, which are available in Excel files, it soon became obvious that there is a necessity to proceed from a formally identical structure derived from the distinct sources. As a consequence of that, we decided to first transform the comparable classification schemes into a shared formal representation language in order to be able to properly align their components before implementing the algorithms for the domain specific multilingual lexicon extraction.

We opted for SKOS-XL[3] for modelling the multilingual terminological part of the comparable taxonomies and OntoLex-Lemon[4] for modelling the lexical items that can be extracted from the labels and the definitions included in the classification schemes.

As both vocabularies, SKOS-XL and OntoLex-Lemon, are using formal representation languages that supports the publication of terminological and lexical datasets on the Linguistic Linked Open Data (LLOD)[5] cloud, our approach can contribute to a significant increase of the linking of such terminological and lexical datasets in the LLOD framework.

In the next sections, we will first introduce the datasets, followed by the description of the formalisation in SKOS-XL of multilingual labels and definitions used in the two classification schemes. After that, we will show how the lexical items used in these labels and definitions can be modelled in OntoLex-Lemon, before displaying the suggested modelling architecture for integrating SKOS- XL and OntoLex-Lemon and finally illustrating how this integration can be published in the LLOD cloud.

2. The Data Sources

We are currently applying our approach on two comparable multilingual industry classification schemes: the In-

[1]See `http://www.elexicography.eu/` for more details.

[2]See `https://elex.link/elex2017/`.

[3]`https://www.w3.org/TR/skos-reference/skos-xl.html`.

[4]See `https://www.w3.org/2016/05/ontolex/` and (McCrae et al., 2017) for more details.

[5]See `http://linguistic-lod.org/llod-cloud` and (Chiarcos et al., 2012) for more details.

dustry Classification Benchmark (ICB)[6] and the Global Industry Classification Standard (GICS)[7]. Both classification schemes are using a four levels taxonomic structure, each level being indexed by a numeral combination associated with a short textual label, and in both classification schemes a definition text is added to the leaf element of the taxonomic structure.

ICB implements a taxonomic structure consisting of 10 industries, subdivided into 19 supersectors, which are further divided into 41 sectors including 114 subsectors, which are the leaf categories/labels that are equipped with a definition. The similar looking GICS consists of 11 sectors, subdivided in 24 industry groups, partitioned in 68 industries and 157 sub-industries, which are the leaf categories/labels to which the definitions are associated.

Figure 1 below shows an example of the taxonomic structure of ICB, in which its 4 levels are indicated by the increasing specification of numbers: $7000 > 7500 > 7530 > 7537$. Here, only the German and English labels are illustrated. Each leaf category/label is associated with a definition, which is shown by the example of index 7537, displayed in Figure 2.

ICB	**German Categories introducing the definition**	**English Categories introducing the definition**
Industry	7000 VERSORGER	7000 Utilities
Supersector	7500 Energieversorgung	7500 Utilities
Sector	7530 Elektrizität	7530 Electricity
Subsector	7537 Alternative Stromerzeugung	7537 Alternative Electricity

Figure 1: The four levels of the ICB classification with German and English labels.

In Figure 1, it can be observed that one and the same word of the source language (English) has been translated with different words in the target language, depending on its level in the taxonomy in which it occurs: "Electricity" is translated in German as being either "Elektrizität" (index 7530) or "Stromerzeugung" (index 7537) and "Utilities" as either "VERSORGER" (index 7000) or "Energieversorgung" (index 7500). We have no information about the reasons behind the existence of those different translations: if they are motivated by style considerations or by the position of the labels in the hierarchical structure remains unclear.

In Figure 2, two different types of statements can be noticed in the definitions. The first sentence, which could be considered an intensional description of the defined term/label, and the second sentence, which can be considered an extensional description, listing mainly subterms. It will be important for an automated extraction of (multilingual) terminologies and lexical elements to be able to distinguish those types of statements.

	7537 Alternative Electricity (EN) Alternative Stromerzeugung (DE)
English Definition	Companies generating and distributing electricity from a renewable source. Includes companies that produce solar, water, wind and geothermal electricity.
German Definition	Firmen, die Strom aus erneuerbaren Quellen erzeugen und vertreiben. Einschließlich Firmen, die Solar-, Wasser- und Windenergie sowie geothermische Energie erzeugen.

Figure 2: Definitions associated to the ICB leaf category/label with index 7537, in German and English.

Figures 3 and 4 display the similar structure of GICS, whereas the reader can observe the differences in labelling the classes and sub-classes on the one hand and the difference in the length of the provided definitions on the other.

In Figure 3, it can be seen that the designers of the taxonomy are using an indexing strategy that differs from the one for ICB, although both rely on numbers for this. It can also be noticed that in this concrete example, GICS uses the same German words to translate "Utilities" in the two categories in which the term occurs. The same remark applies to the two translations of the English noun "Electricity".

	German Categories introducing the definition	**English Categories introducing the definition**
Sector	55 Versorgungsbetriebe	55 Utilities
Industry Group	5510 Versorgungsbetriebe	5510 Utilities
Industry	551050 Unabhängige Energie- und Erneuerbare Elektrizitätshersteller	551050 Independent Power and Renewable Electricity Producers
Sub-Industry	55105020 Erneuerbare Elektrizität	55105020 Renewable Electricity

Figure 3: The four levels of the GICS classification with German and English labels.

In Figure 4, we can observe that the provided textual definitions are significantly longer than in the ICB case. Additionally, the provided definitions do not only offer a combination of intensional and extensional statements, but they also precise which terms should be excluded from the definition of the leaf category/label. This is why a specific algorithm should be applied to those definitions in order to be able to automatically extract terminologies and lexical items.

Those differences are motivating our proposal for a shared formal representation of the two multilingual classification schemes, so that their components can be more easily aligned and form the base for a more accurate lexical extraction. The vocabulary we selected for this modelling is SKOS-XL, which is introduced in section 3.

[6] http://www.icbenchmark.com. ICB covers 13 languages: Chinese, Danish, Estonian, French, Finnish, German, Icelandic, Italian, Japanese, Latvian, Lithuanian, Spanish and Swedish. For each language a different Excel file is available. English is the original language.

[7] https://www.msci.com/gics. GICS covers 10 languages: French, German, Italian, Japanese, Korean, Portuguese, Russian, Simplified Chinese, Traditional Chinese and Spanish. For each language a different Excel file is available. English is the original language.

	55105020 Renewable Electricity (EN) / Erneuerbare Elektrizität (DE)
English Definition	Companies that engage in the generation and distribution of electricity using renewable sources, including, but not limited to, companies that produce electricity using biomass, geothermal energy, solar energy, hydropower, and wind power. Excludes companies manufacturing capital equipment used to generate electricity using renewable sources, such as manufacturers of solar power systems, installers of photovoltaic cells, and companies involved in the provision of technology, components, and services mainly to this market.
German Definition	Unternehmen, die in der Herstellung und Verteilung von Elektrizität unter Verwendung von erneuerbaren Energien tätig sind. Eingeschlossen, aber nicht beschränkt auf, sind Unternehmen, die Elektrizität und/oder elektrische Leistung durch Energiequellen wie Biomasse, Biogase, Sonnen-, Wasser- und Windkraft herstellen. Ausgeschlossen sind Unternehmen, die in der Herstellung von Ausrüstungsgütern für Stromherstellung unter Verwendung erneuerbarer Energien tätig sind, wie Hersteller von Sonnenkraftanlagen, Installateure von photovoltaischen Zellen und Unternehmen, die ihre Technologie, Komponenten, und Dienste hauptsächlich diesem Markt anbieten.

Figure 4: Definitions associated to the GICS leaf category/label with index 55105020.

3. The SKOS-XL Modelling

We are quoting two sources for describing SKOS: "The Simple Knowledge Organization System (SKOS) is a common data model for sharing and linking knowledge organization systems via the Semantic Web."[8] and "SKOS is an RDF vocabulary for describing the basic structure and content of concept schemes such as thesauri, classification schemes, subject heading lists, taxonomies, 'folksonomies', other types of controlled vocabulary, and also concept schemes embedded in glossaries and terminologies"[9]. A number of taxonomies, classification schemes and terminologies have successfully been ported to SKOS[10].

As the quoted texts are pointing out, SKOS is a RDF-based vocabulary, and it is making use of RDF(S)[11] "annotation properties" like `rdfs:label` or `rdfs:comment`. Those annotation properties have been introduced in the RDF(S) vocabulary in order to equip OWL[12] ontological elements, like classes, properties or instances with additional meta-data and also human readable descriptions of the modelled knowledge objects. SKOS introduces three additional annotation properties that can be considered as a specialisation of `rdfs:label` for addressing terminological purposes: `skos:prefLabel`, `skos:altLabel` and `skos:hiddenLabel`. The values of such annotation properties are literals, and have as such no ontological status and can thus not be designated by a URI and consequently can not be used as a subject in RDF triples or as an object in an `owl:ObjectProperty`. SKOS alone would thus not allow us to formally state relations between the terms represented by the labels in the two classification schemes we are dealing with. Fortunately, the W3C community has proposed a remedy to this situation, and defined a corresponding recommendation called SKOS-XL.

SKOS-XL stands for "Simple Knowledge Organization System eXtension for Labels", providing additional support for describing and linking label elements of knowledge systems[13]. SKOS-XL is thus in a sense elevating the values of the `skos:prefLabel`, `skos:altLabel` and `skos:hiddenLabel` properties to the same level as concepts defined in the knowledge sources, supporting thus the cross-linking of labels or their linking to other formal objects. This is exactly the point that makes this formal representation language interesting for our purpose: In SKOS-XL concepts and labels that describe them are the same type of object/entities to which an URI can be associated. Relations between SKOS-XL labels can thus be explicitly and formally defined. A `skos:Concept` can relate to a `skosxl:Label` object via a `skosxl:prefLabel`, a `skosxl:altLabel` or a `skosxl:hiddenLabel` property and users can define all types of relations between `skosxl:Label` objects. This way we can state explicit relations between labels within one classification scheme but also between two or more classification schemes, as can be seen in the following section.

3.1. SKOS and SKOS-XL Encoding of the Components of the GICS and ICB Schemes

In the following two sections we give examples of the encoding of both the taxonomic concepts and the labels of the original classification schemes[14].

3.1.1. The Encoding of the Concepts

We define in one conceptual space "Industry_Classification" a `skos:ConceptScheme` for each taxonomy, as exemplified below in the TTL[15] SKOS-XL code listing 1 for the GICS scheme.

SKOS-XL_Encoding 1: skos:ConceptScheme for GICS

```
gics:ConceptScheme_GICS
  rdf:type skos:ConceptScheme ;
  rdfs:comment "GICS stands for \"Global
      Industry Classification Standard\".
      This structure is effective after
      close of business (US, EST) Wednesday
      - August 31, 2016"@en ;
  rdfs:label "GICS"@en ;
  skos:hasTopConcept gics:Concept_10 ;
  skos:hasTopConcept gics:Concept_55 ;
  .....
.
```

All 11 top-level concepts of GICS and all 10 top-level concepts of ICB are encoded as `skos:Concept` being in a `skos:topConceptOf` relation to their corresponding `skos:ConceptScheme` and in a `skos:narrower` to the concepts placed lower in the original taxonomy, as shown in code listing 2 for the GICS class 55 ("Utilities"), where the reader can see how we introduce the SKOS-XL

[8] Quoted from `https://www.w3.org/TR/skos-reference/skos-xl.html`.

[9] Quoted from `https://www.w3.org/2009/08/skos-reference/skos.rdf`.

[10] See for example `https://www.w3.org/2001/sw/wiki/SKOS/Datasets`.

[11] RDF stands for "Resource Description Framework" and RDF Schema is adding a data model for for the basic RDF vocabulary. See also `https://www.w3.org/TR/rdf-schema/`.

[12] OWL stands for "Web Ontology Language", a Semantic Web representation language for modelling knowledge. See also `https://www.w3.org/OWL/`.

[13] See also `http://lov.okfn.org/dataset/lov/vocabs/skosxl`.

[14] Due to limitation of space, we focus here on GICS, but the encodings are the same for ICB.

[15] TTL stand for "Terse RDF Triple" or more commonly "Turtle", a syntax for serializing RDF triples. See also `https://www.w3.org/TR/turtle/`.

property for linking to two skosxl:Label elements[16], which are exemplified further down, in the SKOS-XL codes 5 (for English) and 6 (for German).

SKOS-XL_Encoding 2: skos:topConceptOf

```
gics:Concept_55
  rdf:type skos:Concept ;
  rdfs:comment "Id of a top-level concept of
       GICS"@en ;
  rdfs:comment "This concept is in the
      domain of \"Sector\""@en ;
  rdfs:label "Utilities"@en ;
  skos:narrower gics:Concept_5510 ;
  skos:topConceptOf gics:ConceptScheme_GICS
      ;
  skosxl:prefLabel gics:Label_55_de ;
  skosxl:prefLabel gics:Label_55_en ;
.
```

All other concepts are encoded as a skos:Concept being in a skos:inScheme relation to the corresponding skos:ConceptScheme, and organized in a skos:broader relation to the concept immediately higher in the original taxonomy. The SKOS-XL code listing 3 gives as an example the GICS ID 5510 ("Utilities").

SKOS-XL_Encoding 3: skos:inScheme concept

```
gics:Concept_5510
  rdf:type skos:Concept ;
  rdfs:comment "Id of a top-level concept of
       GICS"@en ;
  rdfs:comment "This concept is in the
      domain of \"Industry Group\""@en ;
  rdfs:label "Utilities"@en ;
  skos:broader gics:Concept_55 ;
  skos:inScheme gics:ConceptScheme_GICS ;
  skos:narrower gics:Concept_551050 ;
  skosxl:prefLabel gics:Label_5510_de ;
  skosxl:prefLabel gics:Label_5510_en ;
.
```

Finally, The SKOS-XL code in listing 4 is displaying the SKOS encoding for a leaf catagory/label of GICS (class 55105020), including (partially) the definition.

SKOS-XL_Encoding 4: GICS leaf category

```
gics:Concept_55105020
  rdf:type skos:Concept ;
  rdfs:comment "Id of a top-level concept of
       GICS"@en ;
  rdfs:comment "This concept is in the
      domain of \"Sub-Industry\""@en ;
  rdfs:label "Renewable Electricity"@en ;
  skos:broader gics:Concept_551050 ;
  skos:definition "Companies that engage in
      the generation and distribution of
      electricity using renewable sources,
      including, but not limited to, ... ."
      @en ;
  skos:inScheme gics:ConceptScheme_GICS ;
  skosxl:prefLabel gics:Label_55105020_de ;
  skosxl:prefLabel gics:Label_55105020_en ;
.
```

[16]With maximally one skosxl:prefLabel per language. We do not display here all the languages listed in GICS.

3.1.2. The Encoding of the Labels

As already stated, we propose a SKOS-XL encoding for the labels of the original taxonomies, in order to be able to formally express relations between those, within one classification system or between both taxonomies. SKOS-XL code listings 5 and 6 show the basic information associated with the English and German labels of the top level concepts. Those labels are now encoded as instances of an owl:Class and no longer as simple literals as this was the case in SKOS. The German label is marked as being a translation of the English label. In both cases we indicate with lex:identical that the used label is identical to the label of the immediately lower category.[17]

SKOS-XL_Encoding 5: skosxl:Label of a GICS top level concept

```
gics:Label_55_en
  rdf:type skosxl:Label ;
  lex:identical gics:Label_5510_en ;
  rdfs:comment "Labels of a GICS concept"@en
      ;
  rdfs:label "Utilities"@en ;
  skosxl:literalForm "Utilities"@en ;
.
```

SKOS-XL_Encoding 6: German label as a translation of GICS 55 label

```
gics:Label_55_de
  rdf:type skosxl:Label ;
  lex:identical gics:Label_5510_de ;
  lex:isTranslationOf gics:Label_55_en ;
  rdfs:comment "Labels of a GICS concept"@en
      ;
  rdfs:label "Versorgungsbetriebe"@de ;
  skosxl:literalForm "Versorgungsbetriebe"
      @de ;
.
```

The next two TTL code listings are displaying the encodings for the last two levels of the original hierarchy. For the code listing 8 we do not reproduce the full definition, which can be found in Figure 4.

SKOS-XL_Encoding 7: An intermediate German Label

```
gics:Label_551050_de
  rdf:type skosxl:Label ;
  lex:isTranslationOf gics:Label_551050_en ;
  lex:lessSpecific gics:Label_5510_de ;
  lex:moreSpecific gics:Label_55105020_de ;
  rdfs:comment "Labels of a GICS concept"@en
      ;
  rdfs:label "Unabhaengige Energie\- und
      Erneuerbare Elektrizitaetshersteller"
      @de ;
  skosxl:literalForm "Unabhaengige Energie-
      und Erneuerbare
      Elektrizitaetshersteller"@de ;
.
```

[17]In the SKOS-XL encodings of the labels we keep the rdfs:label property, as it is very often queried by Semantic Web applications. So that we have a kind of redundancy to the skosxl:literalForm property.

SKOS-XL_Encoding 8: A German leaf category label

```
gics:Label_55105020_de
  rdf:type skosxl:Label ;
  lex:lessSpecific gics:Label_551050_en ;
  rdfs:comment "Labels of a GICS concept"@en
      ;
  rdfs:label "Erneuerbare Elektrizitaet"@de
      ;
  skos:definition "Unternehmen, die in der
      Herstellung und Verteilung von
      Elektrizitaet unter Verwendung von
      erneuerbaren Energien taetig sind.
      Eingeschlossen, aber nicht ... ."@de ;
  skosxl:literalForm "Erneuerbare
      Elektrizitaet"@de ;
.
```

Once this mapping from the two original classification schemes into a unified SKOS-XL representation has been solved, we started to investigate how the lexical elements contained in the labels can be described in a comparable formalism. We selected for this OntoLex-Lemon (McCrae et al., 2017)[18], as this model already includes a link between lexical items encoded in a standardized RDF vocabulary and the SKOS vocabulary (see Figure 5). In this case, we just need to consider SKOS-XL instead of SKOS for modelling the relation between the conceptual world and the lexicon modelling proposed by OntoLex-Lemon. As examples for this modelling we take instances of the ICB classification scheme (see Figure 1). We display first the SKOS-XL encoding for the German term “Stromerzeugung”, in code listing 9, and then in code listing 10 the corresponding ICB concept.

SKOS-XL_Encoding 9: The German label for ICB 7537

```
icb:Label_7537_de
  rdf:type skosxl:Label ;
  lex:isTranslationOf icb:Label_7537_en ;
  lex:lessSpecific icb:Label_7530_de ;
  rdfs:comment "Labels of a ICB concept"@en
      ;
  rdfs:label "Alternative Stromerzeugung"@de
      ;
  skos:definition """Firmen, die Strom aus
      erneuerbaren Quellen erzeugen und
vertreiben. Einschliesslich Firmen, die
    Solar-, Wasser- und
Windenergie sowie geothermische Energie
    erzeugen."""@de ;
  skos:related gics:Label_55105020_de ;
  skosxl:literalForm "Alternative
      Stromerzeugung"@de ;
.
```

In code listing 9 we give also an example on how we can now link two labels across distinct taxonomies, using for this the `skos:related` property, but any more specific property can be used. The reader can clearly see here the advantage of “elevating” labels to an ontological entity status.

[18]See also the corresponding W3C Community Report `https://www.w3.org/2016/05/ontolex/`.

SKOS-XL_Encoding 10: The encoding for the concept ICB 7537

```
icb:Concept_7537
  rdf:type skos:Concept ;
  rdfs:comment "Id of a concept of ICB"@en ;
  rdfs:comment "This concept is in the
      domain of \"Subsector\""@en ;
  rdfs:label "Alternative Electricity"@en ;
  skos:broader icb:Concept_7530 ;
  skos:definition "Companies generating and
      distributing electricity from a
      renewable source. Includes companies
      that produce solar, water, wind and
      geothermal electricity.{@en@" ;
  skos:inScheme icb:ConceptScheme_ICB ;
  skosxl:prefLabel icb:Label_7537_de ;
  skosxl:prefLabel icb:Label_7537_en ;
.
```

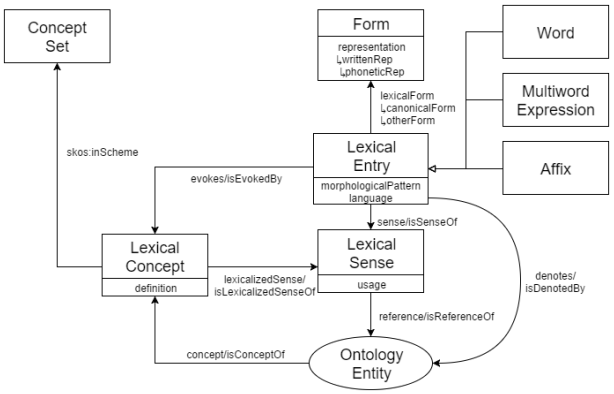


Figure 5: The core module of OntoLex-Lemon: Ontology Lexicon Interface. Graphic taken from `https://www.w3.org/2016/05/ontolex/`.

The following section describes our first steps in the OntoLex-Lemon modelling of lexical data included in ICB/GICS labels.

4. Modelling of the lexical Data

The core module of OntoLex-Lemon is displayed in Figure 5. There the reader can see how the lexical data is related to conceptual data encoded in SKOS. For our example concerning the German term “Stromerzeugung” (ICB ID 5737, see Figure 1), we just need to adapt the OntoLex-Lemon model, and integrate the SKOS-XL vocabulary instead of SKOS. This reflects also our view that terminological data should not be modelled as a lexical data, but rather within a representation framework conceived for terminologies, as this is the case for SKOS-XL. Code listing 11 is showing the suggestion for encoding the lexical item “Stromerzeugung” in OntoLex-Lemon.

SKOS-XL_Encoding 11: OntoLex-Lemon entry for ”Stromerzeugung”

```
ind_class_lemon:lex_7537_2
  rdf:type ontolex:MultiWordExpression ;
  lexinfo:termElement <http://tutorial-
      topbraid.com/ind_class#Label_7537de> ;
```

```
rdf:_1 ind_class_lemon:Component_1 ;
rdf:_2 ind_class_lemon:Component_2 ;
rdfs:label "Stromerzeugung"@de ;
<http://www.w3.org/ns/lemon/decomp#
    constituent> ind_class_lemon:
    Component_1 ;
<http://www.w3.org/ns/lemon/decomp#
    constituent> ind_class_lemon:
    Component_2 ;
ontolex:denotes <http://de.dbpedia.org/
    page/Stromerzeugung> ;
ontolex:evokes icb:Concept_7537 ;
.
```

The link between the lexical description and the terminology is established by using the property `ontolex:evokes`. As OntoLex-Lemon is based on the idea that lexical entries are getting their sense(s) by linking them to elements of ontologies, we are linking the entry to a DBpedia page, using the property `ontolex:denotes`. A nice feature is the fact that we can link the word "Stromerzeugung" to the SKOS-XL label 7537, stating that it is a part of it (`lexInfo:termElement`), and not only to the SKOS-XL concept 7537. We take advantage of the fact, that OntoLex-lemon is also supporting the modelling of compound words. We can decompose the word and link to its components ("Strom" and "erzeugung"). And from there to link the components to the related lexical entries "Strom" and "Erzeugung". Importantly, we can also link the component "Strom" directly to the right sense ("Electricity" versus "River")[19]. In doing this, we are fulfilling lexicographic requirements in the context of terminology and OntoLex-Lemon proved to be a very satisfactory modelling framework.

5. Linguistic Linked Data Cloud

The Linguistic Linked Open Data (LLOD) cloud[20] is an initiative to break the data silos of linguistic data and thus encourage NLP applications that can use data from multiple languages, modalities (e.g., lexicon, corpora, etc.) and develop novel algorithms. Looking at the current state of the LLOD, one can see that the data sets published in this cloud are classified along the lines of six categories:

- Corpora
- Terminologies, Thesauri and Knowledge Bases
- Lexicons and Dictionaries
- Linguistic Resource Metadata
- Linguistic Data Categories
- Typological Databases

Not all the date sets are equally linked to each other, and our approach can contribute in better linking the data sets in the fields of Terminologies, Thesauri and Knowledge Bases and those in the fields of Lexicons and Dictionaries.

[19]This approach is based on (Declerck and Lendvai, 2016)

[20]See `http://linguistic-lod.org/llod-cloud`.

6. Conclusion

We have implemented the integration of two different but closely related formal representation languages, SKOS-XL and OntoLex-Lemon, for encoding terminological and lexical data that are used in classification schemes as interrelated knowledge objects. This makes those data accessible in the Linked Open Data and also in the Linguistic Linked Open Data cloud[21]. This formalisation seemed to be a necessary pre-requisite for our original task, which consists in extracting multilingual domain-specific dictionaries from such classification systems. The next step in our work will consist in implementing the extraction algorithms based on the formal representation of the terms and the language data used in those terms.

7. Acknowledgements

The DFKI contribution to this work has been partially funded by the BMBF project "DeepLee - Tiefes Lernen für End-to-End-Anwendungen in der Sprachtechnologie" with number 01-W17001. The ACDH contribution is supported in part by the H2020 project "ELEXIS" with Grant Agreement number 731015. We would like to thank the participants of the ENeL WG3 meeting held in Budapest in February 2017 for comments on the very first steps of our investigation.

8. Bibliographical References

Chiarcos, C., Hellmann, S., and Nordhoff, S., (2012). *Linking Linguistic Resources: Examples from the Open Linguistics Working Group*, pages 201–216. Springer Berlin Heidelberg, Berlin, Heidelberg.

Declerck, T. and Lendvai, P. (2016). Towards a formal representation of components of german compounds. In Micha Elsner et al., editors, *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Humboldt University, ACL.

Gromann, D. and Declerck, T., (2014). *A Cross-Lingual Correcting and Completive Method for Multilingual Ontology Labels*, pages 227–242. Springer.

McCrae, J. P., Buitelaar, P., and Cimiano, P. (2017). The OntoLex-Lemon Model: development and applications. In *Proceedings of eLex 2017*.

[21]`http://linguistic-lod.org/llod-cloud`.

One event, many representations. Mapping action concepts through visual features.

Alessandro Panunzi, Andrea Amelio Ravelli, Lorenzo Gregori
Università degli Studi di Firenze, Florence, Italy
{alessandro.panunzi, lorenzo.gregori, andreaamelio.ravelli}@unifi.it

Abstract

This paper faces the problem of unifying the representation of actions and events in different semantic resources. The proposed solution exploits the IMAGACT visual component (video scenes that represent physical actions) as the linkage point among resources. By using visual objects, we connected resources responding to different scopes and theoretical frameworks, in which a concept-to-concept mapping appeared difficult to obtain. We provide a brief description of two experiments that exploit IMAGACT videos as a linkage point: an automatic linking with BabelNet, a multilingual semantic network, and a manual linking with Praxicon, a conceptual knowledge base of action. The aim of this work is to integrate data from resources with different level of granularity in order to describe the action semantics from a linguistic, visual and motor point of view.

Keywords: ontology linking, semantics, action verbs, knowledge representation

1. Introduction

Action verb interpretation is a basic issue for human-machine interaction systems that aim to process natural language instructions. The difficulty behind automatic action verb understanding comes out from the evidence that no one-to-one correspondence can be established between action predicates (lexical items in each natural language) and action concepts (mental representations of experienced events). The same action can be predicated by multiple verbs and, conversely, one verb can extend to multiple and different actions. Most of these verbs belong to the class of general verbs, which are characterized by a high ambiguity and high frequency in the use (Moneglia, 2010). In these circumstances, senses are often vague and overlapping, their discrimination is not clear, and this is a critical issue for their semantic representation. In fact, if we look at WordNet (Fellbaum, 1998), the proposed classification of general action verbs highlights two main issues: on one hand, a synset often encodes a variety of events that are cognitively conceived as different action concepts; on the other hand, it's frequent that a specific event is not clearly described by a unique verb sense, but it seems to be spread on more senses (belonging to different synsets), each one representing a possible conceptualization of the same event. The following examples explain these classification problems.

Example 1 shows a synset that represents a very general sense of *putting objects in a location* that can refer to actions of pouring, inserting or laying a body part. In the Example 2 there is an action of *beating up someone* (Figure 1), that can be correctly encoded by two senses of the verb *to beat*.

Example 1

bn:00090224v[1]: place, lay, put, set, pose. *Put into a certain place or abstract location*.

- John puts the wine in the glass;
- John puts the letter in the envelop;
- John puts his hand on Mary's shoulder.

Example 2

- **bn:00083248v**: beat up, beat, work over. *Give a beating to; subject to a beating, either as a punishment or as an act of aggression*;
- **bn:00083249v**: beat. *Hit repeatedly*.



Figure 1: The action *John beats/beats up/batters Paul*.

The action representation is even more difficult in a multilingual perspective, given that different languages operate different segmentations of the action domain; for example, classification methods built upon English language do not necessarily hold in other languages, and especially for typologically-different language families (Majid et al., 2007). It has been observed that often it is not possible to find an exact match between lexicalized action concepts in different languages, even with a fine-grained sense distinction (Moneglia and Panunzi, 2007). Moreover, one language could totally lack a lexical representation for a specific concept, whenever there is a lexical gap (Gregori and Panunzi, 2017).

[1] The IDs reported in Example 1 and 2 are taken from BabelNet taxonomy, that derives directly from WordNet. (see 2.2. for more details.

These considerations highlight a big issue in the creation of linguistic ontologies, where there is a need to divide the word meaning in a set of senses that are discrete, well-defined and related together through a predefined set of semantic relations. A word sense discrimination task is tricky and the existence of an universal set of word senses is questioned both theoretically (Wittgenstein, 1953; Pustejovsky, 1991; Croft and Cruse, 2004) and computationally (Kilgarriff, 1997; Resnik and Yarowsky, 1999; Cimiano et al., 2013).

The difficulty in finding a shared representation of concepts reflects in the existence of a wide variety of ontologies and lexical resources that are often bounded to a specific theoretical model and have different levels of granularity in their concept definition. In this context, finding matches between concepts encoded in different resources is a hard task.

We describe here a *visual mapping* methodology, that has been applied to connect together action concepts from different resources. Instead of a classic concept-to-concept mapping, visual mapping performs a concept-to-video linking. In fact, a video depicting an event is not subject to any linguistic constraint, and therefore the associated semantic information can be described in various manners. Starting from this observation we used videos to link concepts of different resources, that express independent event conceptualization according to their own theoretical framework.

By exploiting the videos featured in the IMAGACT ontology of action, we applied the visual mapping to connect BabelNet, a general multilingual semantic network, and Praxicon, a specific conceptual knowledge base of action.

2. Resources

2.1. IMAGACT

IMAGACT Visual Ontology of Action[2] (Moneglia et al., 2014) is a multimodal and multilingual resource that offers a novel integration of visual and linguistic information as complementary elements. The resource contains 1010 distinct action concepts as a result of an information bootstrapping from Italian and English spoken corpora. Metaphorical and phraseological usages have been excluded from the annotation process, in order to collect only the occurrences referring to physical actions.

Verbs in IMAGACT are divided into *action types*, according to their semantic variation. An action type gathers a group of actions, that are perceived as unitary from a cognitive point of view. Each type is linked to one or more video scenes (either 3D animations or filmed video clips) of performed actions, that act as prototypes for it. The verbs of each language referring to the same actions are linked to the same scenes, resulting in an interlinguistic and multimodal semantic network.

The ontology is in continuous development and, at present, contains 9 languages and 13 more that are under development, with an average of 730 action verbs per language.

This resource gives a broad picture of the variety of actions and activities that are prominent in everyday life and specifies the lexicon used to express each one in ordinary communication, in all the included languages.

2.2. BabelNet

BabelNet[3] (Navigli and Ponzetto, 2012) is a multilingual semantic network developed through the automatic mapping of the WordNet thesaurus and the Wikipedia encyclopedia. At present, BabelNet 3.7 contains 284 languages and it is the widest multilingual resources available for semantic disambiguation. Concepts and named entities are represented by BabelSynsets (BSs), unitary concepts identified by several kinds of information (semantic features, glosses, usage examples, images, etc.) and related to lemmas (in any language) which have a sense matching with that concepts. BSs are not isolated, but connected together into a huge network by means of the semantic relations inherited from WordNet.

2.3. Praxicon

Praxicon[4] is an ontology for the representation of action concepts, based on the Minimalistic Grammar of Action (Pastra and Aloimonos, 2012). In Praxicon, an action is expressed through motor concepts, specified in terms of 3 basic components: GOAL, TOOL and OBJECT. A wide part of this ontology is also linked with WordNet synsets and ImageNet images (Deng et al., 2009).

Praxicon makes a distinction between *Actions*, *Movements*, and *Events*[5]. Actions are sets of structured motor execution, intentionally performed by an agent with a tool to achieve a goal. The goal is a necessary component, so any non-voluntary motor activation is addressed as a Movement, but not as an Action. Finally, actions that are too complex to be described as a set of motor concepts, are considered Events and are out of the scope of the Praxicon resource.

3. Visual mapping at work

Herein we show how the visual mapping technique has been applied to link IMAGACT with BabelNet and Praxicon.

An example of the linking between IMAGACT, Praxicon and BabelNet can be observed in Figure 2, that shows a *beating* event with the parallel representation in the 3 resources.

3.1. IMAGACT and BabelNet

BabelNet concepts (the BSs) are interlinguistic: they gather all the word senses in different languages that are semantically equivalent (or almost equivalent). Conversely, IMAGACT action types encode small semantic differences, so they are more granular and language-dependent. Given these differences, an exact match between concepts is very rare; it's also hard to establish less strict semantic relations (e.g. *narrow-to-broad*), because the BSs boundaries are often fuzzy and the gloss is not always able to make a clear discrimination between them.

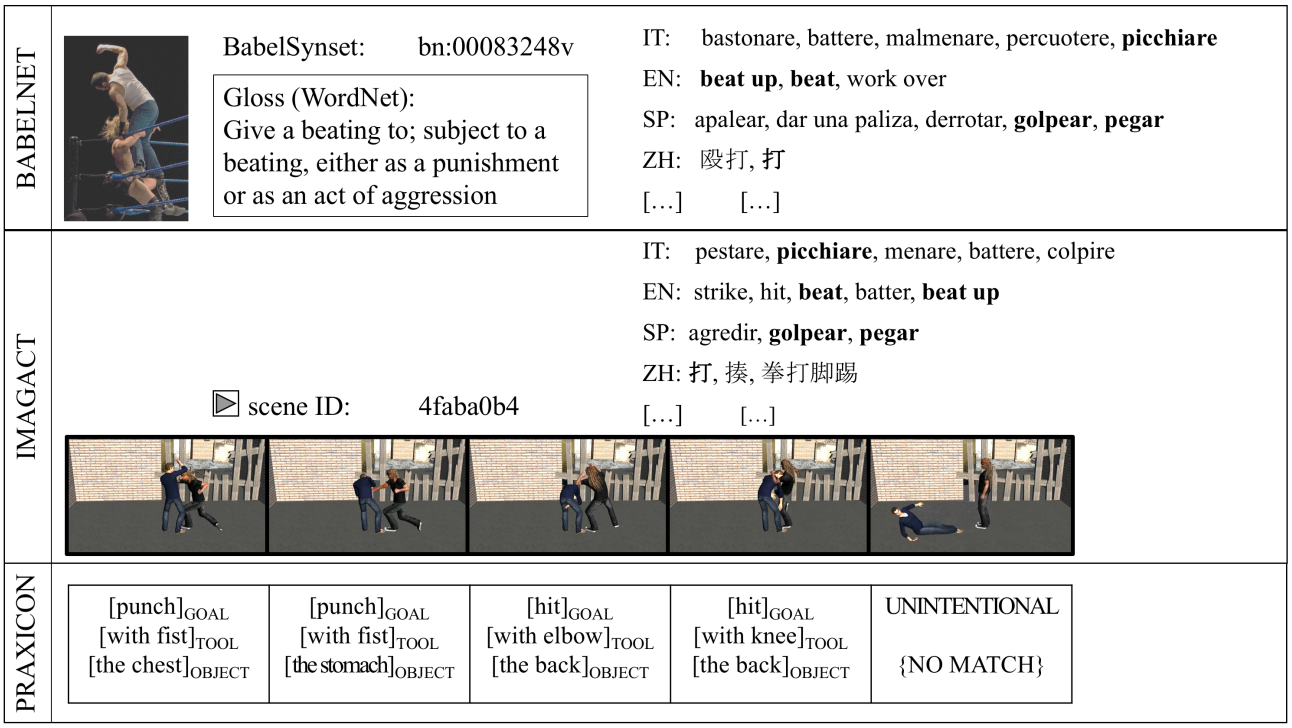
[2] http://www.imagact.it/

[3] http://www.babelnet.org/

[4] https://github.com/CSRI/PraxiconDB

[5] These categories have their own definition in the Praxicon framework. We use capital letters when referring to this specific meaning

Figure 2: An example of the resulting linking between BabelNet, IMAGACT and Praxicon.



In this case visual mapping solved the problem: in fact even for the BSs where the description is not precise, it's easy to say if a video is a good action prototype for it or not.

Given the multilingual nature of the two resources, we could exploit a rich lexical information, i.e. all the verbs in many languages related both to IMAGACT scenes and BabelNet BSs. The connections between BSs and scenes have been automatically established through a Machine Learning algorithm (Gregori et al., 2016).

In order to perform this linking, a dataset of 50 scenes and 57 BabelSynsets (2,850 human judgments in total) have been created[6]. Each ⟨BS, Scene⟩ pair has been evaluated to check if the scene is appropriate in representing the BS. Three annotators compiled the binary judgment table and we reported a Fleiss' kappa inter-rater agreement of 0.74, meaning that at least 2 annotators out of 3 gave the same value for each pair.

IMAGACT data belonging to 17 languages have been exploited to train the algorithm. We used three basic features: the number of verbs connected to the Scene, the number of verbs connected to the BS and the number of verbs that are shared between the Scene and the BS. In each pair ⟨BS, Scene⟩ these features have been calculated for the candidate BS and also for its neighbors, i.e. the other BSs connected though a semantic relation in the BabelNet semantic network.

A Support Vector Machine (SVM) classifier with a RBF kernel have been used to create the model. Table 1 reports Precision, Recall and F-measure.

[6]The manually annotated training set is published at `http://bit.ly/2jt2cD4`

	Baseline $th = 0.04$	**ML Algorithm** 27 features
Pr	0.580	0.833
Re	0.529	0.441
Fm	0.553	0.577

Table 1: Precision, Recall and F-measure of BSs to scenes linking task calculated on the test set for the algorithm and the baseline.

3.1.1. Results

Both the resources took an advantage from this linking: IMAGACT gained translation information for languages still not implemented in the Visual Ontology, and BSs referring to action verbs obtained a video representation. In Table 2, the detailed numbers of scenes and BSs connected through this linking are shown.

Table 2: IMAGACT-BabelNet linking results.

IM Scenes linked to BS	773
BS linked to Scenes	517
IM English Verbs related to Scenes	544
BabelNet English Verbs related to BS	1,100

3.2. IMAGACT and Praxicon

Similarly to the linking with BabelNet, the IMAGACT scenes have been used to connect the information from Praxicon, given that the definitions of concept in the two

resources are too different to obtain a proper and extensive match. In fact, the IMAGACT scenes can work as a visual representation for Praxicon concepts and, at the same time, Praxicon syntax could be used to analytically describe, from a motor point of view, all the low-level actions involved in the execution of more complex ones.
Differently from the previous linking, in this case it is a totally manual work, consisting in the analysis of each scene, the determination of the physical actions performed, and the annotation in Praxicon syntax of the motor executions.
IMAGACT scenes are specifically created to provide a prototypical representation of a lexicalized action concept: every scene is a reference of at least one action verb. For this reason, such a work of annotation allows to derive some interesting results about the relation between motor and lexical level.

3.2.1. Results

The linking with Praxicon is still in progress: the results are partial, but we believe that the integration between linguistic and motor knowledge on action is very relevant both for theoretical analysis and robotic applications. From one side an integrated resource is desirable to carry on deep investigations on the relation between language and action, that is a long debated subject in linguistics and neuroscience (Pustejovsky, 1991; Pulvermüller, 2005; Kemmerer and Gonzalez-Castillo, 2010). Praxicon is also exploited for robotic applications (Vitucci et al., 2016; Tsagarakis et al., 2007) and the integration with a linguistic-oriented resource like IMAGACT can be useful to enhance human-robot interaction through natural language.
The scene annotation has been accomplished on 281 IMAGACT scenes (∼28% of the total) and we obtained the following results[7]:

- 154 scenes (∼55%) have a one-to-one relation with Praxicon Action concepts;
- 64 scenes (∼23%) map on more than one Action concept;
- 30 (∼11%) are Events but not Actions (in the Praxicon framework);
- 19 scenes (∼7%) are Movement but not Actions (in the Praxicon framework);
- 14 scenes (∼5%) are unclear.

This data rises some interesting observations about the relation between action verbs and the motor information they express. Consider these two sentences:

1. *John pushes the door*;
2. *John opens the door*.

The verb *to push* (sentence 1) focuses on the performance of the physical action (the *pushing* event) and not on the result, that depends on contextual factors: a closed door will open by pushing; an open door will close by pushing. Otherwise the verb *to open* (sentence 2) has a specific focus on the result without providing information about the physical action required to reach it: *pushing*, *smashing*, *turning the key*, and so on.
Verbs that focus on the action performance (like *to push*, *to gallop* or *to brush*) reflect some motor features in their semantics. For example the action described in sentence 1 has some motor features that are also encoded in the semantics of *to push*: the application of a force on an object and the outbound direction of the movement. Conversely, verbs that focus on the action result (like *to break*, *to open* or *to hang*) do not encode specific motor features, given that the result is achieved by performing a set of different physical actions.
This difference mirrors in the annotated scenes: in fact scenes connected to verbs that focus on the performance have a one-to-one relation with Praxicon Action concepts, while scenes connected to verbs that focus on the result usually map on more than one Action concept.
Another thing that emerged from the annotation is the presence of some verbs (like *to drive*, *to clean* or *to rob*) that predicate complex activities, which are characterized by a high number of physical action that varies a lot depending on the context. In a sentence like *John drives the car*, the activity involves a sequence of actions performed within a loose temporal structure: *turning the steering wheel*, *pushing the pedals*, *moving the gearshift*, and so on. The scenes connected to these verbs are considered Events in the Praxicon Framework, and not Actions.
Finally, the scenes that depict a non-voluntary motor activation (like *John falls down*) does not have a goal, so they are not considered Actions, but Movements in Praxicon Framework.

[7] At the moment, due to the unfinished state of this task, the inter-annotator agreement have not been calculated.

4. Conclusions

We introduced the visual mapping methodology that allows resource linking through visual representations. This approach is particularly useful when it's hard to find relations between concepts, as in the representation of actions and events, because it does not force any kind of convergence between senses. For this reason, we feel confident that this methodology could be successfully applied also in other linking tasks involving multimodal resources.
Two case studies have been described: the linking of IMAGACT with BabelNet and Praxicon. In the first case we were dealing with two lexical-semantic resources having huge differences in sense discrimination, and for this reason it was hard to find inter-resource semantic relations. In the case of Praxicon we applied visual mapping to link IMAGACT with a resource of a different type, in which the concepts are motor and not linguistic. This allowed us to derive some preliminary considerations on the relation between linguistic and motor level in action semantics.

5. Bibliographical References

Cimiano, P., McCrae, J., Buitelaar, P., and Montiel-Ponsoda, E. (2013). On the role of senses in the ontology-lexicon. In *New trends of research in ontologies and Lexical resources*, pages 43–62. Springer.

Croft, W. and Cruse, D. A. (2004). *Cognitive linguistics*. Cambridge University Press.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Christiane Fellbaum, editor. (1998). *WordNet: an electronic lexical database*. MIT Press.

Gregori, L. and Panunzi, A. (2017). Measuring the italian-english lexical gap for action verbs and its impact on translation. *SENSE 2017*, page 102.

Gregori, L., Panunzi, A., and Ravelli, A. A. (2016). Linking imagact ontology to babelnet through action videos. *CLiC it*, page 162.

Kemmerer, D. and Gonzalez-Castillo, J. (2010). The Two-Level Theory of verb meaning: An approach to integrating the semantics of action with the mirror neuron system. *Brain and Language*, 112(1):54–76.

Kilgarriff, A. (1997). I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113.

Majid, A., Bowerman, M., van Staden, M., and Boster, J. S. (2007). The semantic categories of cutting and breaking events: A crosslinguistic perspective. *Cognitive Linguistics*, 18(2):133–20.

Moneglia, M. and Panunzi, A. (2007). Action predicates and the ontology of action across spoken language corpora. the basic issue of the semact project. In *M. Alcántara, T. Declerck, In International Workshop on the Semantic Representation of Spoken Language (SRSL7). Salamanca: Universidad de Salamanca*, pages 51–58.

Moneglia, M., Brown, S., Frontini, F., Gagliardi, G., Khan, F., Monachini, M., and Panunzi, A. (2014). The imagact visual ontology. an extendable multilingual infrastructure for the representation of lexical encoding of action. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Moneglia, Massimo; Panunzi, A., (2010). *Language, Cognition and Identity. Extension of the Endocentric/Esocentric Typology*, chapter I verbi generali nei corpora di parlato. Un progetto di annotazione semantica cross-linguistica, pages 27–46. Florence University Press.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Pastra, K. and Aloimonos, Y. (2012). The minimalist grammar of action. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 367(1585):103–117.

Pulvermüller, F. (2005). Brain mechanisms linking language and action. *Nature reviews. Neuroscience*, 6(7):576.

Pustejovsky, J. (1991). The syntax of event structure. *Cognition*, 41(1):47–81.

Resnik, P. and Yarowsky, D. (1999). Distinguishing systems and distinguishing senses: New evaluation methods for word sense disambiguation. *Natural language engineering*, 5(2):113–133.

Tsagarakis, N. G., Metta, G., Sandini, G., Vernon, D., Beira, R., Becchi, F., Righetti, L., Santos-Victor, J., Ijspeert, A. J., Carrozza, M. C., et al. (2007). icub: the design and realization of an open humanoid platform for cognitive and neuroscience research. *Advanced Robotics*, 21(10):1151–1175.

Vitucci, N., Franchi, A. M., and Gini, G. (2016). Programming a humanoid robot in natural language.

Wittgenstein, L. (1953). Philosophical investigations (gem anscombe, trans.).

Tel(s)-Telle(s)-Signs: Highly Accurate Automatic Crosslingual Hypernym Discovery

Ada Wan
University of Zurich
Zurich, Switzerland
ada.wan@uzh.ch

Abstract

We report a highly accurate hypernym discovery heuristic that works on unrestricted texts. This approach leverages morphological cues in French, but given any parallel data and word alignment tool, this proves to be a technique that can work reliably in other languages as well. We tested this method using two French-English corpora of different genres (medical and news) and attained near-perfect accuracy. The key idea is to exploit morphological information in the French trigger phrase *tel*(*s*)*-/telle*(*s*)*- que* (meaning "such as" in English) to uniquely identify the correct hypernym. This shows to be an inexpensive and effective heuristic also when there are multiple noun phrases preceding the trigger phrase, as in the case of prepositional phrase attachment causing ambiguity in interpretation and hypernym acquisition, indicating that this pattern in French is more informative than its English counterpart.

Keywords: hypernym discovery, information extraction, crosslingual knowledge transfer

1. Introduction

The present work focuses on discovering hypernyms in French (FR) using the trigger phrase *tel-*[1] *que* (meaning "such as" in English (EN)), exploiting morphological information to uniquely identify the correct hypernym. This shows to be an inexpensive and effective heuristic also when there are multiple noun phrases preceding the trigger phrase, as in the case of prepositional phrase attachment causing ambiguity in interpretation and hypernym acquisition, proving that this pattern in FR is more informative than its EN counterpart.

In simpler cases, where there is only one noun phrase (NP) preceding the trigger phrase *such as*, hypernym discovery is straightforward. For instance, in the sentence:

[1] *Les agrumes tels que l'orange, le citron ou le pamplemousse contiennent beaucoup de vitamin C.*[2]

and its EN translation:

"Citrus fruits such as oranges, lemons or grapefruits contain a lot of vitamin C."

the NPs preceding *such as* and *tels que* – (*citrus*) *fruits* and *agrumes*, respectively – are the hypernyms denoting the categories or classes in which the hyponyms *oranges*/*orange*, *lemons*/*citron*, and *grapefruits*/*pampelmousse* belong. But seldom does one notice that the bound morpheme *-s* in *agrumes* and *tels* in FR also reveals agreement between the two words and has a role to play in hypernym recognition.

It appears that one could use this morphological information in *tel-* to correctly identify the antecedent of this hyper-/hyponym relation, disambiguating cases where multiple nouns preceding *tel- que* (hereafter: *tq*) pose as plausible hypernym candidates. This would also resolve the issue brought about by the prepositional compound phrase construction, i.e. the combination of a noun phrase and a prepositional phrase (Lefever et al., 2014), alternatively formulated as the error caused by the "local nature of the patterns" (Ritter et al., 2009):

> "A sentence with ... *urban birds in cities such as pigeons* ... matches the pattern 'C such as E' with C bound to *city* and E bound to *pigeon*, leading to *city* as a hypernym of *pigeon*."

(Note that *cities*, the more local noun to *such as*, can be the hypernym in the sentence ... *urban birds in cities such as Paris, Rome* ....)

This paper demonstrates the reliability and accuracy of our proposed agreement pattern in hypernym discovery in FR. Once a hypernym is identified in FR in this manner, reliable extraction of hypernyms in other languages can also be expected given proper word alignment and bitexts.

2. Related Work

Ontological relations are commonly extracted from unstructured texts to build and extend semantic taxonomies or relational databases, such as *WordNet* and *DBpedia*, in addition to industry-specific knowledge bases. Hearst (1992) broke ground with her seminal work in ontological discovery which aimed to automatically acquire hyponyms in unrestricted texts. Both Yamada et al. (2009) and Lefever et al. (2014) noted the myriad of research in this area since then – many approached this topic with pattern-based methods, while others used clustering or distributional similarity ones as well as techniques with word class lattices and embeddings. Yet, except in work by Shinzato & Torisawa (2004), Tjong Kim Sang (2007), Bosma & Vossen (2010), Tjong Kim Sang et al. (2011), Fu et al. (2013), and Lefever et al. (2014), morphology (word-internal structure) has played little role in ontological relation extraction. All of these authors exploited morphology in a similar fashion – they treated the rightmost constituent or affix (roughly considering radicals in Chinese characters as affixes of sorts here) of a complex noun or a multi-word expression as

[1] *tel-* is used as a shorthand for *tel*, *tels*, *telle*, or *telles*, cf. Section 3.1.

[2] `http://bdl.oqlf.gouv.qc.ca/bdl/gabarit\_bdl.asp?id=2437`

the head noun and hence the hypernym. For example, in Japanese: *eiga* as the hypernym of *amerika-eiga* "American movie" and *nihon-eiga* "Japanese movie"; in Dutch: *beleid* "policy" as the hypernym of *landbouwbeleid* "agricultural policy" and *pijpleiding* "pipeline" as the hypernym of *offshore pijpleiding* "offshore pipeline"; and in Chinese: the radical 虫 "insect" as the hypernym of 蜻蜓 "dragonfly" and 企鹅 "penguin" as the hypernym of 皇帝企鹅 "emperor penguin". Lefever et al. also accounted for prepositional compound phrases by designating the head at the left edge of the compound phrase as the hypernym, e.g. *saneren* "remediation" as hypernym of *saneren van verontreinigde bodems* "remediation of contaminated soils". However, this account would not suffice for constructions requiring the more local noun to be the hypernym.

In addition to addressing the issue of how lexico-morphological information can be used in hypernym detection as in the work above, we aim to demonstrate in this paper how leveraging agreement information can help resolve ambiguity in hypernym identification and report how the reliability of these cues can transfer to high accuracy in extracting hypernyms in another language given proper word alignment.

3. Method

3.1. Approach

The discovery of our approach to hypernym recognition was a data-driven endeavor inspired by one lexico-syntactic pattern indicated in Hearst (1992) using *such as*:

NP_0 such as $\{\text{NP}_1,\ \text{NP}_2 \ldots, (\text{and}|\text{or})\}\ \text{NP}_n$

where it is implied that:

for all NP_i, $1 \leq \text{i} \leq \text{n}$, hyponym($\text{NP}_i$, NP_0), i.e. hypernym(NP_0, NP_i).

While examining EN-FR parallel sentences in the EMEA corpus, we learned that *tq* is the most common translation for *such as*. (*que* also has a variant *qu'* – mostly when the next word starts with a vowel.) We noticed that not only does the pattern "NP *tq* NP" manifest a hyper-/hyponym relationship as one would expect in EN, but the word *tel-* also agrees with the preceding noun which is the hypernym of said pattern. Generally, adjectives and certain pronouns in FR agree in gender and number with nouns which they modify or co-refer with. The *tel-* in *tq* is found to vary – in its masculine singular form *tel*, feminine singular *telle*, masculine plural *tels*, and feminine plural *telles*. There is always (at least) one noun preceding *tq* with matching agreement – distant or local. That should not be surprising as there is a rule in FR grammar prescribing that when *tq* introduces a comparison and appears before a single example or an enumeration, *tel-* is to agree with the noun that precedes it, i.e. the one that it exemplifies[3], as can be seen in sentence [1] in Section 1., as well as in [2] below:

[2] *À l'examen, vous pourrez utiliser des livres de référence tels que grammaires et encylopédies.*

(EN translation: "At the examination, you can use reference books such as grammars and encyclopedias.")

tels in [2] agrees with *livres de référence*, more precisely with the head noun *livres*, as they are both masculine plural (mp); it does not agree with other nouns in the sentence, e.g.: *référence* (feminine singular (fs)), *grammaires* (fp), or *encyclopédies* (fp)).

We hypothesize that the closest noun preceding *tq* that agrees with *tel-* in gender and number is the hypernym in the sentence in FR in the pattern[4]:

NP_x... ($\text{NP}_{y \neq x}$)* tel_x que $\{\text{NP}_1, \ldots (\text{et}|\text{ou})\}$ NP_n

where the subscripts x and y indicate gender and number marking, and where:

for all NP_i, $1 \leq \text{i} \leq \text{n}$, hyponym($\text{NP}_i$, NP_x), i.e. hypernym(NP_x, NP_i).

3.2. Experiment

We tested our hypothesis using two sets of FR-EN parallel corpora – the EMEA corpus made from PDF documents from the European MEdicines Agency[5] (Tiedemann, 2009) and the news commentary training data from the WMT 2014 shared task[6]. After tokenization, we filtered sentences with length over 100 words for the EMEA corpus and 200 for the WMT corpus. We used GIZA++ (Och and Ney, 2003) with the grow-diag-final-and heuristic (Koehn et al., 2007) for word alignment, and tagged the FR text using the morphological analyzer *morfette*[7] (Chrupala et al., 2008).

We selected the first 50 unique sentence pairs from each corpus for manual evaluation. For these 100 pairs of sentences, the questions we ask in the evaluation process are:

1. Is the closest noun preceding *tq* that agrees with *tel-* in gender and number the hypernym in the sentence?

2. Did *morfette* predict the correct outcome in question 1 above?

3. Is the hypernym in FR aligned with the hypernym in EN?

A positive result for question 1 would confirm that our hypothesis is accurate, that a hypernym in FR can be predicted using morphological information. A positive result for question 2 would show the hypernym can be easily extracted with the aid of a morphological analyzer like *morfette*. A positive result for question 3 would indicate that hypernyms in EN can also be automatically extracted by

[3] http://bdl.oqlf.gouv.qc.ca/bdl/gabarit_bdl.asp?id=2437

[4] Similar to the corresponding Hearst pattern in EN, our pattern assumes *tq* to be immediately surrounded by Ns/NPs, not verbs (e.g. passive participle in *such as indicated*) or subordinate clauses after *que* (esp. since *que*, like *that/as* in EN, can often be followed by a clausal structure beginning with e.g. a N-V sequence). There should also be no punctuation between *tel-* and its preceding NP.

[5] http://opus.lingfil.uu.se/EMEA.php

[6] http://statmt.org/wmt14/translation-task.html

[7] https://sites.google.com/site/morfetteweb/home

means of this heuristic with word alignment tools such as GIZA++.

4. Results and Discussion

Out of the 100 sentences we evaluated manually, 97 confirm our hypothesis. In 91 of these cases, *morfette* correctly tagged the hypernyms for gender and number, suggesting that the morphological cue can be readily extracted. Finally, crosslingual alignment of the hypernyms succeeded in 87.5[8] instances: the errors here were mostly due to the original sentence alignment being wrong, such that the EN sentences failed to contain the corresponding pattern in the first place. Table 1 provides a summary broken down by corpus.

We noticed that not only can our hypothesis disambiguate prepositional compound phrases but also potentially obviate the notion of headedness or syntactic parsing for the task of hypernym extraction. Below are sets of sample sentences from the EMEA corpus which will help elucidate (gender and number information, as provided by *morfette*, is suffixed here onto FR nouns and *tel-* with an underscore for easier reference: _ms (masculine singular), _fs (feminine singular), _mp (masculine plural), _fp (feminine plural)):

1. Long-distance agreement:

   FR: *APTIVUS, co-administré avec le ritonavir_ms à faible dose_ms, doit être utilisé avec précaution_fs chez les patients_mp pouvant présenter un risque_ms accru de saignement_ms en raison_fs d' un traumatisme_ms, d_ms' une chirurgie_fs ou d' antécédents médicaux autres, ou chez ceux recevant des* **traitements_mp** *connus pour augmenter le risque_ms de saignement_ms* **tels_mp que** *les antiagrégants_mp plaquettaires et les anticoagulants_mp, ou chez ceux qui prennent de la vitamine_fs E.*

   EN: *APTIVUS, co-administered with low dose ritonavir, should be used with caution in patients who may be at risk of increased bleeding from trauma, surgery or other medical conditions, or who are receiving* **medicinal products** *known to increase the risk of bleeding* **such as** *antiplatelet agents and anticoagulants or who are taking supplemental vitamin E.*

   Without knowing the meaning of most of the words in the sentence, we learn that *les antiagrégants plaquettaires* and *les anticoagulants* are hyponyms to *traitements* "medicinal products" in FR because *tels* tells us that the closest preceding noun having the same masculine plural ending is *traitements*, bypassing all nearer neighbors *risque* and *saignement*.

2. Nearest neighbor:

   FR: *Celles- ci incluent l'inhibition_fs de la libération_fs de* **cytokines_fp** *proinflammatoires* **telles_fp que** *IL-4, IL-6, IL-8, et IL-13 par les mastocytes_mp basophiles_fp humains, ainsi que l'inhibition_ms de l'expression_fs de la molécule_fs d'adhésion_fs P-sélectine_fs sur les cellules_mp endothéliales_mp.*

   EN: *These include inhibiting the release of proinflammatory* **cytokines such as** *IL-4, IL-6, IL-8, and IL-13 from human mast cells/ basophils, as well as inhibition of the expression of the adhesion molecule P-selectin on endothelial cells.*

   The first NP in the FR sentence in this sentence pair can be literally translated as "the inhibition of the release of proinflammatory cytokines such as IL-4...". If one had no idea what these hyponyms *IL-4*, *IL-6* etc. could be, one could potentially interpret these as kinds of inhibition or release. The head of this NP is *inhibition*, but it bears no relevance – *telles* agrees with a more local noun *cytokines*, and that's the hypernym of the NPs that immediately follow *que*.

3. Surface recognition:

   FR: • *Maladies_fp de l'œ sophage et autres* **facteurs_mp** *qui retardent le transit_ms œ sophagien* **tels_mp que** *sténose et achalasie_fs.*

   EN: • *Abnormalities of the oesophagus and other* **factors** *which delay oesophageal emptying* **such as** *stricture or achalasia.*

   From a purely structural perspective, there are multiple possible ways of parsing the above sentence pair, different scopes with conjunction and subordination. But the reading with *facteurs* (mp) as hypernym is the only one that makes sense in natural language and is also the only one that *tels* in the FR sentence allows for given our hypothesis. With our heuristic, there is no need for any deep analysis beyond some shallow morphological tagging.

In the news commentary data, there are more instances with *tq* followed by one single hyponym. All of the 3 instances where our hypothesis failed to predict the correct hypernym in FR fall into this class – in one case, *tel-* seems to agree with the hyponym, in another, it doesn't agree with anything in the sentence (possible human errors). The third case, shown below, exemplifies the limit of this approach – if multiple nouns agree with *tq*, morphology alone cannot always uniquely detect the correct hypernym. It should be noted, however, that we only observed this pattern once in our sample of 100 sentences. Here, *guerre* "war" should be the hypernym, but *usure* "attrition" was predicted:

FR: *Malheureusement, les Israéliens_mp peuvent leur prouver qu'ils ne réussiront pas à détruire Israël sans faire l'expérience_fs d'une guerre_fs d'usure_fs violente* **telle_fs que** *celle qu'ils mènent.*

EN: *Unfortunately, the Israelis can show the Arabs that they cannot destroy Israel only by enduring the violent war of attrition that the Arabs are pursuing.*

There are also more cases where a pronominal NP (e.g. *celle* "the one") follows *que*, as in the example above as well as in ... *des expériences telles que celle de l'Iran* ...

[8] 0.5 due to partial alignment in 1 multiword expression.

	total number of unique sentences evaluated	hypothesis correct	FR hypernym extractable (using morfette)	EN hypernym extractable (using GIZA++)	
				best	worst
EMEA	50	50 (100%)	45 (90%)	41.5/44 (94.32%)	41.5/50 (83%)
WMT14 news commentary	50	47 (94%)	46 (92%)	46 (92%)	

Table 1: Results (*best* indicates when sentences misaligned to the extent that the relevant pattern is not available in EN are excluded, *worst* is when these cases are included in the count of total evaluated)

(lit.: "... experiences such as the one of Iran ..."), EN: ... *Iran's experiences* ..., but these are rarely translated into EN literally. In these sentences with only one hyponym, our hypothesis is still able to account for the majority of them. To obtain higher accuracy, one may want to restrict the hypothesized pattern to an enumeration of NPs following *que*, as opposed to "one or more NPs".

5. Conclusion

We presented a lexico-morphological pattern that facilitates accurate extraction of hypernyms in French, leveraging a morphological cue that enables unique disambiguation of many instances which would require semantic knowledge in other languages such as English. Transferring the results of this approach from French to English with the aid of bitexts and word alignment was found to be feasible. We understand that this discovery has its limit in coverage. That said, we hope that our effort in bringing this somewhat latent observation to light will inspire researchers to pay heed to these often neglected features in language that are available with little to no cost.

6. Bibliographical References

Bosma, W. and Vossen, P. (2010). Bootstrapping language neutral term extraction. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Chrupala, G., Dinu, G., and van Genabith, J. (2008). Learning morphology with morfette. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Fu, R., Qin, B., and Liu, T. (2013). Exploiting multiple sources for open-domain hypernym discovery. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234, Seattle, Washington, USA, October. Association for Computational Linguistics.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, COLING '92, pages 539–545, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lefever, E., Van de Kauter, M., and Hoste, V. (2014). Evaluation of automatic hypernym extraction from technical corpora in English and Dutch. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, March.

Ritter, A., Soderland, S., and Etzioni, O. (2009). What is this, anyway: Automatic hypernym discovery. In *Learning by Reading and Learning to Read, Papers from the 2009 AAAI Spring Symposium, Technical Report SS-09-07, Stanford, California, USA, March 23-25, 2009*, pages 88–93.

Shinzato, K. and Torisawa, K. (2004). Acquiring hyponymy relations from web documents. In Daniel Marcu Susan Dumais et al., editors, *HLT-NAACL 2004: Main Proceedings*, pages 73–80, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Tiedemann, J. (2009). News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, et al., editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia.

Tjong Kim Sang, E., Hofmann, K., and de Rijke, M. (2011). Extraction of hypernymy information from text. In *Interactive Multi-modal Question-Answering. Theory and Applications of Natural Language Processing*, pages 223–245. Springer, June.

Tjong Kim Sang, E. (2007). Extracting hypernym pairs from the web. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 165–168, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yamada, I., Torisawa, K., Kazama, J., Kuroda, K., Murata, M., De Saeger, S., Bond, F., and Sumida, A. (2009). Hypernym discovery based on distributional similarity and hierarchical structures. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 929–937, Stroudsburg, PA, USA. Association for Computational Linguistics.

Disambiguation of Verbal Shifters

Michael Wiegand*, Sylvette Loda*, Josef Ruppenhofer†
*Spoken Language Systems, Saarland University, †Institute for German Language
*D-66123, Saarbrücken, Germany, †D-68161, Mannheim, Germany
michael.wiegand@lsv.uni-saarland.de, sylvette.loda@lsv.uni-saarland.de
ruppenhofer@ids-mannheim.de

Abstract

Negation is an important contextual phenomenon that needs to be addressed in sentiment analysis. Next to common negation function words, such as *not* or *none*, there is also a considerably large class of negation content words, also referred to as *shifters*, such as the verbs *diminish*, *reduce* or *reverse*. However, many of these shifters are ambiguous. For instance, *spoil* as in *spoil your chance* reverses the polarity of the positive polar expression *chance* while in *spoil your loved ones*, no negation takes place. We present a supervised learning approach to disambiguating verbal shifters. Our approach takes into consideration various features, particularly generalization features.

Keywords: sentiment analysis, negation modeling, word-sense disambiguation

1. Introduction

Negation is one of the most central linguistic phenomena. Therefore, negation modeling is essential to various common tasks in natural language processing, such as relation extraction (Sanchez-Graillet and Poesio, 2007), recognition of textual entailment (Harabagiu et al., 2006) and particularly sentiment analysis (Wiegand et al., 2010). In the latter task, a negation typically reverses the polarity of polar expressions. For example, in (1), the negated positive polar expression *pass* conveys negative polarity.

(1) Peter did [**not** [pass]$^+$]$^-$ the exam.
(2) Peter [**failed** to [pass]$^+$]$^-$ the exam.

So far, most research in negation modeling for sentiment analysis has focused on negation (function) words, such as the particle *not* or the adverbs *no* or *never*. However, among other word classes, particularly the content words, such as verbs, nouns and adjectives, there also exist words expressing negation. For example, in (2) the verb *failed* has a similar function as the negation word *not* in (1). These *negation content words*, which are also called ***shifters***, are often excluded from discussion since there does not (yet) exist a commonly accepted resource with these expressions. Even though the frequency of a single negation function word is much higher than that of a shifter, the overall number of shifters is significantly larger than those of negation function words. Schulder et al. (2017) identified 980 verbal shifters while the popular negation word lexicon proposed by Wilson et al. (2005) only includes 15 negation function words. Moreover, since content words are more ambiguous than function words, we also envisage shifters to be more ambiguous than negation function words.

In this paper, we address the ambiguity of shifters. We select a set of 20 ambiguous verbal shifters and try to disambiguate them automatically. We focus on verbal shifters since it has been recently shown that a large amount of such verbs exist and they are important to polarity classification (Schulder et al., 2017). Examples for ambiguous verbal shifters are *clear*, *spoil*, *cloud* or *slump* which in (3), (5), (7) and (9) convey negation while in (4), (6), (8) and (10) they do not.

(3) The well-known actor was [**cleared** [of murder]$^-$]$^+$.
(4) That highly controversial bill *cleared* the House and caused three days of riots in the streets.
(5) Political blunders [**spoiled** [their chance of being re-elected]$^+$]$^-$.
(6) On Valentine's day, all people should *spoil* their loved ones.
(7) Suspicions of drug use have [**clouded** [her prospects of a promotion]$^+$]$^-$.
(8) The solution *clouds* if you shake it.
(9) [[My spirits]$^+$ **slumped**]$^-$ at the sight of him.
(10) After a busy day, the successful businessman *slumped* into his armchair and watched TV.

Our notion of negation focuses on the impact of polar expressions within the scope of a particular negation word or shifter. This may include cases that are no full negation in the proper semantic sense. For instance, in (5) a *chance* of being re-elected still exists. However, it has been significantly reduced. In terms of polarity, this is typically interpreted as a shift from positive to negative polarity (i.e. a *slight chance* is considered less positive than a *great chance*). It is not a proper negation in the sense that the there is no chance at all.

Moreover, the ambiguous verbs may also often carry an intrinsic polarity. Whether or not a particular mention of such verb conveys some form of negation, however, depends on the fact whether it shifts the polarity of its arguments. For example, in (5) the verb *spoil* acts as a shifter as the positive polarity denoted by its argument (i.e. *their chance of being re-elected*) is shifted. In (6), on the other hand, the verb conveys a positive polarity (as *spoil* here means treating someone with a lot of care and kindness). The polarity of its direct object is not reversed, i.e. the loved ones still remain to be loved. Therefore, in this sentence the verb does not function as a shifter despite carrying an intrinsic polarity.

In this paper, we follow a **supervised learning** approach. We frame the task as a binary classification problem. Either a mention of a verbal shifter actually conveys negation or it does not. The aim of this research is three-fold: First, we want to find out whether this disambiguation is learnable at all. Secondly, we want to determine which types of features are helpful for this task. Thirdly, we examine what type of training data is necessary: Is it necessary to have training data for every ambiguous verbal shifter in order to disambiguate them automatically or is it possible to generalize over different ambiguous verbs and therefore make automatic predictions for mentions of *unknown* ambiguous shifters?

All labeled data produced as part of this research effort are publicly available.[1]

2. Data & Annotation

We selected a set of 20 ambiguous verbal shifters (Table 1). We particularly focused on those types of verbs which display a high degree of ambiguity. This was established by annotating the senses associated with those verbs. All of our 20 verbs contain between 36.3% and 60.0% senses according to WordNet (Miller et al., 1990) in which the verb denotes some form of negation.

For each of those ambiguous verbs 100 sentences were randomly extracted from the North American News Text Corpus (LDC95T21) resulting in 2000 sentences. We only took sentences into account that had between 10 and 40 tokens. Short sentences, particularly headlines, and very long sentences are often incorrectly parsed. Since some of our features rely on the output of a syntactic parser and it would be beyond the scope of this work to improve parsing quality for very short and long sentences, we focus on those sentences that are more likely to be correctly parsed.

Each of the 2000 sentences was manually annotated. The annotator had to decide for each sentence whether the ambiguous verbal shifter which it contained conveys some form of negation or not. Instead of directly annotating the sentences from scratch, we first looked at the set of all possible word senses of each ambiguous verbal shifter (according to WordNet) and decided for each word sense whether it conveys negation or not. In the actual annotation of the sentences, the annotators also first determined the particular word sense (according to WordNet) it conveyed and then made the final decision on the basis of the assigned word sense. We found that by this procedure we could notably improve interannotation agreement. On a subset of 600 sentences, an interannotation agreement of $\kappa = 0.79$ was measured. This level of agreement can be considered substantial (Landis and Koch, 1977).

On the entire gold standard, 62% of the mentions of the ambiguous verbal shifters were annotated as shifters, while the remaining 38% were found not to convey any form of negation. Table 2 summarizes the most important statistics of our gold standard.

[1] https://github.com/miwieg/lrec2018

clear, corrupt, cost, crumple, depress, disdain, dissolve, drain, eject, hold down, jam, overturn, paralyse, sink, slump, soothe, spoil, trash, tumble, worsen

Table 1: The 20 ambiguous verbs.

Property	Freq
Number of unique ambiguous verbal shifters	20
Number of annotated sentences	2000
Number of sentences per verbal shifter	100
Average number of tokens in sentence	26
Proportion of mentions conveying negation	62%

Table 2: Some statistics of the gold standard.

3. Feature Engineering

3.1. Surface-based Features: Bag of Words

Since we can consider our task as some type of text classification task, we should examine simple surface-based features, such as bag of words. Despite both their genericity and simplicity, these features are known to be very predictive.

3.2. Word Generalization Features

Even though we expect bag of words to be predictive for this task, we also anticipate this type of feature to suffer from data sparsity. With 2000 sentences, our gold standard is not very large. The mere fact that we conduct a text classification on the sentence level further exacerbates this problem. (Sentences contain considerable fewer words than larger units of texts which are usually considered for text classification, such as paragraphs or documents.) Therefore, we examine different types of word generalization methods. What all these methods try to accomplish is that a classifier is able to classify a given context with words not observed in the training data. This can be achieved by harnessing the similarity of those unknown words and words observed in the training data.

Table 3 illustrates for the ambiguous verb *spoil* that the objects for the sense *negation* and *no negation* are of a specific semantic type. For the sense *negation*, the objects represent some form of activity, while for the sense *no negation*, they typically represent some human being (or at least some animate entity). These observed selectional preferences are an indication that some form of generalization of the context words might help for this task.

Brown Clustering. A popular data-driven word generalization method is Brown Clustering (Brown et al., 1992). This is an unsupervised clustering method in which words with the similar distributional contexts are automatically assigned the same clusters. Clusters therefore represent a set

Sense	Contexts	Type
negation	spoil a(n) {effort|idea|fun}	*activity*
no negation	spoil one's {partner|girlfriend|spouse}	*human*

Table 3: Illustration of word generalization.

[Negation Sense	No Negation Sense
[**clear** someone of murder^{-}]$^{+}$	sky *cleared*
[**spoil** someone's efforts^{+}]$^{-}$	*spoil* one's spouse
[peace^{+} **crumples**]$^{-}$	*crumple* a shirt
[**cost** someone their inner peace^{+}]$^{-}$	*cost* some amount of money

Table 4: Polar expressions. to indicate negation sense.

of words rather than individual ones.

In our experiments, we induce 1000 clusters from the North American News Text Corpus. This is a standard configuration proven to yield good results (Turian et al., 2010). We induce the clusters using the SRILM-toolkit (Stolcke, 2002).

Word Embeddings. A more recent alternative to Brown clustering is the usage of word embeddings. Word embeddings are (dense) vector representations of words that are automatically induced from corpora. They are devised as a more robust alternative to bag of words. Unlike a *one-hot* bag-of-words vector representation where different words (no matter how similar they are in meaning) are always orthogonal to each other, embeddings allow different words which are distributionally similar, such as *partner* and *spouse*, also to have similar vector representations. In our experiments, we induce word embeddings using Word2Vec (Mikolov et al., 2013). The embeddings are induced from the North American News Text Corpus. In order to avoid overfitting, we leave the tool in its default configuration.

WordNet Hypernyms and Supersenses. We use WordNet (Miller et al., 1990) as a resource-based method for word generalization. WordNet is the largest lexical ontology for the English language. It is organized in word senses called *synsets*.[2] By considering hypernyms of a synset representing some set of lemmas as a feature, we enable similar words, i.e. synonyms or near-synonyms, to have a feature in common. While hypernyms are a fairly fine-grained form of generalization, we also consider *supersenses*, a set of coarse-grained classes, which have previously been found to be effective for sentiment analysis (Flekova and Gurevych, 2016).

3.3. Polarity Information

We assume that many ambiguous shifters convey a negation if they co-occur with polar expressions. This is illustrated in Table 4. Therefore, we count the number of polar expressions in a sentence to be classified. We identify such expressions with the help of the *Subjectivity Lexicon* (Wilson et al., 2005).

3.4. Focused Features

Our task can be considered as a word-sense disambiguation (WSD) task. Therefore, we should also consider a feature set established in previous work on WSD. Table 5 lists those features mainly inspired by Akkaya et al. (2009).

[2]Since we are not aware of any robust open-domain word-sense disambiguation software, we always consider the union of all synsets associated with a particular lemma.

Feature
subcategorization frame of verbal shifter
hypernym(s) of dependents of verbal shifter
supersense(s) of dependents of verbal shifter
is a polar expression among dependents of verbal shifter?
is verbal shifter coordinated with another verbal shifter?
words representing dependents of verbal shifter

Table 5: Focused features

	unknown verbs		known verbs	
Feature	**Acc**	**F1**	**Acc**	**F1**
bag of words	63.8	59.6	70.7	67.9
embeddings	64.4	60.8	70.8	68.1
bag of words + embed.	64.3	60.7	70.9	68.2

Table 6: Bag of words vs. word embeddings.

They have in common that they all only consider a very local context of the mention of the verbal shifter (i.e. typically its dependents). Some of these features incorporate syntactic information. We use the Stanford Parser (Chen and Manning, 2014) to obtain that information.

4. Experiments

We evaluate our features in a 10-fold crossvalidation. The classifier, we consider is a Support Vector Machine. As a tool we use SVM^{light} (Joachims, 1999). We consider two different evaluation settings. On the one hand, we arrange the verbs in such a way that the test data contain contexts of verbs which have not been observed in the training data. On the other hand, we ensure that all test data contain contexts of verbs that have been observed in the training data. As a baseline, we also list the performance of a majority-class classifier.

For the bag-of-words features, we experimented with different window sizes but found that using all words in a sentence provides best performance. For the word embeddings, however, we had to consider a fixed window size because we need to establish that all vectors representing an instance have the same dimensionality. In order to achieve this, we took the word embeddings of the words in a fixed context window and simply concatenated them to a large vector. We established that for the setting using observed verbal shifters, the optimal window size is $n = 8$ and for the setting of unknown verbs, the size is $n = 6$.

Table 6 compares the performance of bag of words and word embeddings and their combination. The performance of the two representations is very similar and there is no significant benefit in combining them. Consequently, we will exclusively employ the bag-of-words feature in our remaining experiments.

Table 7 compares different kinds of feature sets. It shows that we can significantly outperform our bag-of-words baseline on the setting dealing with unknown verbs. The performance on this setting is also notably worse than on observed verbs. Still, even on the former setting, we manage to outperform the majority-class baseline. This is an important result as it indicates that in order to learn this

	unknown verbs		known verbs	
Feature	**Acc**	**F1**	**Acc**	**F1**
majority	61.7	38.1	61.7	38.1
bag of words	63.8*	59.6*	70.7*	67.9*
all features	**66.8***	**63.1***	**70.8**	**68.1**

*: significantly better than previous feature using paired t-test ($p < 0.05$)

Table 7: Comparison of different features.

	unknown verbs		known verbs	
Feature	**Acc**	**F1**	**Acc**	**F1**
all features	66.8	63.1	70.8	68.1
w/o bag of words	66.8	63.3	70.8	68.1
w/o brown clusters	66.7	63.2	70.2	67.5
w/o hypernyms	65.0*	60.7*	69.8	67.1
w/o supersenses	66.2	62.1	70.7	68.1
w/o polarity	66.7	63.2	70.9	68.2
w/o focused	66.1	63.2	70.9	68.1

*: significantly worse than *all features* using paired t-test ($p < 0.05$)

Table 8: Ablation experiment.

disambiguation we do not require sentences with all possible ambiguous verbal shifters. However, our results also suggest that for such a setting a more sophisticated set of features is necessary.

Table 8 presents the results of an ablation experiment in which we compare the performance of the full feature set with a feature set where one type of feature is removed. The table shows that most feature types do not convey unique information since if they are removed, classification performance usually only drops marginally. There is only one notable exception: on the setting using unknown verbs the omission of WordNet hypernyms causes a significant drop in performance.

Figure 1 shows a learning curve of the most important feature sets. Due to the limited space of this paper, we only display the setting using unknown verbs. Judging by the slope of the curve, we could expect further improvements of classification performance by adding more training data. The table also shows that when more than 40% of the training data are used, the entire feature set systematically outperforms the bag-of-words baseline.

5. Related Work

With regard to WSD, our work follows the strand of research that argues for a coarse-grained set of sense inventories (Palmer et al., 2004; Hovy et al., 2006; Navigli, 2006; Snow et al., 2007). Coarse-grained sense inventories have an obvious practical advantage. They are easier to discriminate than fine-grained sense inventories. This applies to both human and automatic categorization. They typically also require fewer training data.

The work most closely related to ours is Akkaya et al. (2009) in that a coarse-grained sense inventory is examined for sentiment analysis. That work proposes two senses for expressions potentially conveying

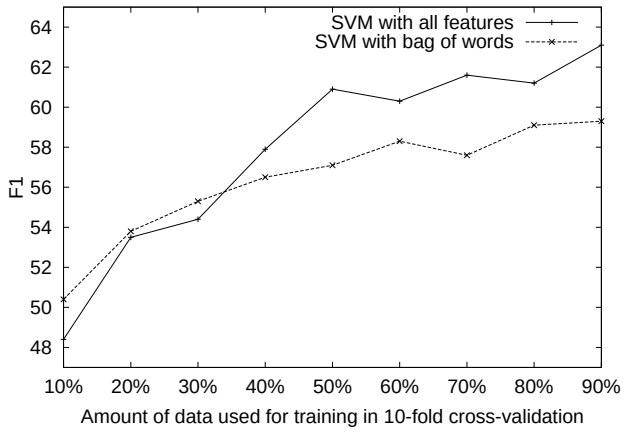


Figure 1: Learning curve on gold standard.

subjectivity, one sense in which the expression indeed conveys subjectivity and another in which it does not. We also consider two sense categories: a verb may convey negation or it may not.

To the best of our knowledge, this paper is the first work to address the ambiguity of shifters as a major research task. The most comprehensive study on negation with respect to polarity (Wilson et al., 2005) already identified the problem of negation words being ambiguous. Since the negation words that are considered in that work are predominantly negation function words, the problem is not considered that severe. In fact, those few cases of ambiguity that are identified concern the negation particle *not* and can be reliably disambiguated by a handful of simple phrasal patterns, that is, *not only* in (11), *not just* in (12) and *if not* in (13). Note that the polar expressions in the scope of the mentions of *not* in these three examples (i.e. *problem* in (11), *skill* in (12) and *worse* in (13)) all preserve their polarity in those contexts, that is, they are not negated by *not*. For our verbal shifters, such simple phrasal rules could not be identified on our dataset.

(11) This is *not* only a problem$^-$ that concerns this nation but the entire world population.

(12) Camouflage is *not* just a critical skill$^+$ in combat, it is the recipe for victory.

(13) The situation in Afghanistan is as bad, if *not* worse$^-$ than, under the Taliban.

6. Conclusion

We presented an approach to automatically disambiguate verbal shifters. The task was framed as a supervised learning approach. We found that, in principle, one can learn to distinguish these senses and that even classification of unknown shifters is possible. Here, particularly word generalization features are important. In general, very simple surface features, such as bag of words are already effective. Our learning curve suggests, however, that in order to produce a classifier with reasonable performance more labeled training data are required.

Acknowledgements

The authors were partially supported by the German Research Foundation (DFG) under grants RU 1873/2-1 and WI 4204/2-1.

7. Bibliographical References

Akkaya, C., Wiebe, J., and Mihalcea, R. (2009). Subjectivity Word Sense Disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 190–199, Singapore.

Brown, P. F., deSouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Chen, D. and Manning, C. D. (2014). A Fast and Accurate Dependency Parser using Neural Networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar.

Flekova, L. and Gurevych, I. (2016). Supersense Embeddings: A Unified Model for Supersense Interpretation, Prediction, Utilization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2029–2041, Berlin, Germany.

Harabagiu, S., Hickl, A., and Lacatusu, F. (2006). Negation, Contrast and Contradiction in Text Processing. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 755–762, Boston, MA, USA.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% Solution. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (HLT/NAACL)*, pages 57–60, New York, NY, USA.

Joachims, T. (1999). Making Large-Scale SVM Learning Practical. In B. Schölkopf, et al., editors, *Advances in Kernel Methods - Support Vector Learning*, pages 169–184. MIT Press.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at the International Conference on Learning Representations (ICLR)*, Scottsdale, AZ, USA.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. (1990). Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3:235–244.

Navigli, R. (2006). Meaningful Clustering of Senses Help Boost Word Sense Disambiguation. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING/ACL)*, pages 105–112, Sydney. Australia.

Palmer, M., Babko-Malaya, O., and Dang, H. T. (2004). Different Sense Granularities for Different Applications. In *Proceeddings of the HLT-NAACL-Workshop on Scalable Natural Language Understanding*, pages 49–56, Boston, MA, USA.

Sanchez-Graillet, O. and Poesio, M. (2007). Negation of protein-protein interactions: analysis and extraction. *Bioinformatics*, 23(13):i424–i432.

Schulder, M., Wiegand, M., Ruppenhofer, J., and Roth, B. (2017). Towards Bootstrapping a Polarity Shifter Lexicon using Linguistic Features. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, Taipei, Taiwan.

Snow, R., Prakash, S., Jurafsky, D., and Ng, A. Y. (2007). Learning to Merge Word Senses. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1005–1014, Prague, Czech Republic.

Stolcke, A. (2002). SRILM - An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, pages 901–904, Denver, CO, USA.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word Representations: A Simple and General Method for Semi-supervised Learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 384–394, Uppsala, Sweden.

Wiegand, M., Balahur, A., Roth, B., Klakow, D., and Montoyo, A. (2010). A Survey on the Role of Negation in Sentiment Analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 60–68, Uppsala, Sweden.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pages 347–354, Vancouver, BC, Canada.

Bootstrapping Polar-Opposite Emotion Dimensions from Online Reviews

Luwen Huangfu and Mihai Surdeanu
University of Arizona
Tucson, AZ 85719, USA
{huangfuluwen, msurdeanu}@email.arizona.edu

Abstract

We propose a novel bootstrapping approach for the acquisition of lexicons from unannotated, informal online texts (in our case, Yelp reviews) for polar-opposite emotion dimension values from the Ortony/Clore/Collins model of emotions (e.g., *desirable/undesirable*). Our approach mitigates the intrinsic problem of limited supervision in bootstrapping with an effective strategy that softly labels unlabeled terms, which are then used to better estimate the quality of extraction patterns. Further, we propose multiple solutions to control for semantic drift by taking advantage of the polarity of the categories to be learned (e.g., *praiseworthy* vs. *blameworthy*). Experimental results demonstrate that our algorithm achieves considerably better performance than several baselines.

Keywords: bootstrapping, semantic drift, limited supervision

1. Introduction

Bootstrapping is a lightly-supervised learning approach in which supervision comes in the form of a small number of initial examples (or seeds). While bootstrapping is an attractive choice for NLP, the limited supervision involved also yields important drawbacks. First, the training of bootstrapping models often "drifts" semantically from the original task into different tasks (e.g., from learning women names into learning flower names). Second, the lack of labeled data (i.e., only a small set of seed examples is annotated) impedes the capacity of the model to correctly assess the quality of the generated model during training.

In this paper we propose solutions for the above issues in the context of learning lexicons for the emotion dimensions (e.g., Desirability, Praise-/Blame-worthiness) necessary to assemble the Ortony/Clore/Collins (OCC) cognitive model of emotions (1990). According to the OCC framework, 22 different *emotion types* are generated from several *emotion dimensions*. In this work, we focus on the dimensions of Desirability, Praise-/Blame-worthiness and Likelihood because they are central emotion dimensions that contain both positive values (e.g., *desirable, praiseworthy, certain*) and negative values (e.g., *undesirable, blameworthy, likely*). For example, the emotion type *JOY* combines the dimensions Desirability with value *desirable* and Likelihood with value *certain*.

The contributions of this work are:

1. We propose multiple lightly-supervised solutions for the acquisition of emotion dimensions that control for semantic drift by taking advantage of the polarity of the classes to be learned (i.e., positive/negative appraisals).

2. We introduce an effective strategy to softly label unlabeled terms, i.e., unlabeled terms are assigned a value that indicates how close they are to a given category, and use these soft labels to better estimate the quality of extraction patterns in the above bootstrapping approaches.

3. We show that multiple resources (WordNet, word embeddings that project words in a continuous vector space that capture distributional similarity, and edit distance similarity) all help for the above two contributions, and are complementary to each other.

4. We empirically demonstrate that our approach outperforms several strong baselines for the acquisition of emotion dimensions lexicons from informal texts such as product reviews on the web.

2. Related Work

In the vast bootstrapping literature, a few works attempted to address the two limitations mentioned in the introduction. With respect to mitigating semantic drift, Kozareva and Hovy (2010) used stronger constraints for their lexicon extraction patterns, encouraging them to stay within the desired category to be acquired. Yangarber (2003) proposed "counter training", which introduces competition between the multiple categories (e.g., lexicon or event types) that are learned simultaneously (i.e., they are not allowed to overlap). This idea was generalized by the NELL system (Carlson et al., 2010). McIntosh and Curran (2010) extended counter training with negative categories that are discovered on the fly. Our approach is closest to counter training, with the extension that we propose multiple "soft" exclusion criteria.

With respect to the better handling of unlabeled data, Gupta and Manning (2014) improved the scoring of extraction patterns by predicting the labels of unlabeled terms, and using this information to better estimate the precision of the candidate patterns. Gupta and Manning (2015) extended this idea with a k nearest neighbors (kNN) formulation that expands the labeled training data with unlabeled entities that are close (according to kNN) to seed examples. Popescu and Etzioni (2005) applied a similar idea to the extraction of opinion words, where the unlabeled terms are labeled using a combination of syntactic, WordNet constraints, and relaxation labeling. Our work builds on these ideas with a simpler approach (no classifier is used). We also investigate more resources to measure the distance

Algorithm 1: Bootstrapping for emotion dimensions

input : A set of documents $\mathcal{D}$; seed words $\mathcal{S}$ for k emotion dimension values, $k \in \{$*praiseworthy*, *blameworthy*, *desirable*, *undesirable*$\}$

1 $\mathcal{Z} = \langle\rangle$ // stores extraction patterns for each k
2 $\mathcal{E} = \mathcal{S}$ // stores terms for each k
3 **foreach** epoch t **do**
  // expand the known set of terms:
4 **foreach** dimension value k **do**
    // we denote with $-k$ the dimension value opposite to k, e.g., if k is *desirable*, $-k$ is *undesirable*; if k is *praiseworthy*, $-k$ is *blameworthy*
5    $E(k) = \text{expandTerms}(\mathcal{E}(k), \mathcal{E}(-k), \mathcal{D})$
  // discover new extraction patterns:
6 **foreach** dimension k **do**
7    $P(k) = \text{extractAndRankPatterns}(E(k), E(-k))$
    // keep most relevant patterns:
8    $\mathcal{Z}(k) = \mathcal{Z}(k) + \text{getTop}(P(k))$
  // discover new terms:
9 **foreach** dimension k **do**
10   $T(k) = \text{extractAndRankTerms}(\mathcal{Z}(k), \mathcal{Z}(-k))$
    // keep most relevant terms:
11   $\mathcal{E}(k) = \mathcal{E}(k) + \text{getTopWithPolarityChecking}(T(k))$

output: $\mathcal{E}$

from known examples, ranging from WordNet (Miller et al., 1990) to `word2vec` (Mikolov et al., 2013), and show that they provide complementary information.

3. Approach

Algorithm 1 lists our proposed algorithm that extracts lexicons corresponding to emotion dimension values (or categories). Our algorithm builds on the traditional bootstrapping approach, which starts with a small set of seed examples, and alternates between learning extraction patterns and using them to discover new information (Riloff, 1996; McIntosh and Curran, 2008).

There are two fundamental differences between our approach and previous work. First, by using external information such as word embedding similarity, we expand the current set of acquired terms for each category, which are then *softly* labeled with category information (lines 4 – 5). The expanded term set is then used for the discovery of new extraction patterns (lines 6 – 8). Second, unlike McIntosh and Curran (2008)'s approach, which defined mutually exclusive categories, we remove the hard mutual-exclusivity constraint between categories. Instead, by taking advantage of the inherent polarity of the emotion dimension values, e.g., *desirable* is the opposite of *undesirable*, we introduce multiple soft constraints between opposite categories (k vs. $-k$). Such constraints are used throughout the algorithm (lines 5, 7, 10, 11). We detail these steps next.

3.1. Term Expansion

Expanding the terms (discussed below in this section) used for pattern ranking (see Section 3.2.) is important as it mitigates the sparsity of pattern-based methods with other complementary resources (akin to co-training (Blum and Mitchell, 1998)). We show in Section 4. that this step is important, especially in domain-specific settings, which are driven by "little data" scenarios.

The algorithm for term expansion relies on three resources: edit distance, word embeddings, and WordNet. We use these to generate candidate terms as follows: (a) We calculate the edit distance between every term in our corpus and the current term pool for category k ($\mathcal{E}(k)$); we consider new terms as candidates for category k if their edit distance is below a threshold (we used the same formula and threshold as (Gupta and Manning, 2014)). (b) We compute the average cosine similarity between every term's word embedding vector and the vectors of known terms in category k, and consider a term as a candidate if this similarity is above a threshold.[1] (c) Lastly, we use WordNet synonyms, derived wordforms, direct hypernyms and hyponyms of the term currently in the pool of category k as expansion candidates for k, and their antonyms as candidates for $-k$.

It is important to note that these three resources have complementary strengths and weaknesses. For example, edit distance naturally captures misspellings, but it also introduces false positives, e.g., "goods" as a candidate for the dimension containing "good". However, using the similarity of word embedding vectors mitigates this problem because the distributional similarity of the two words is low. On the other hand, antonyms (e.g., "good" vs. "bad") tend to have a high distributional similarity (Yih et al., 2012). WordNet addresses the latter problem, but its coverage is far more limited than that of word embedding models, which limits its applicability to domain-specific texts.

In order to allow these different resources to help each other, they have to interact. To do this, we assign to each candidate term produced above a score that combines the three resources:

$$\text{cscore}(c_k) = EDP - EDN + EMP - EMN + WNP - WNN \quad (1)$$

where c_k is a candidate term for category k. EDP and EDN use the discretized **e**dit **d**istance from the current **p**ositive/**n**egative entities (i.e., terms belonging to category k vs. $-k$) of Gupta and Manning (2014). EMP and EMN are the average cosine similarities between the **em**bedding vector of c_k and the embedding vectors of the terms in the **p**ositive category k (for EMP) and **n**egative category $-k$ (for EMN).

Lastly, WNP and WNN measure the overlap of the candidate term WordNet synset information with the terms in the **p**ositive and **n**egative category, respectively, as follows. In particular, the WNP score of a candidate term c_k is computed as the term overlap between the synonyms of c_k (Syn(c_k)) and the set of expanded terms in k from the previous epoch $E(k)^{t-1}$ plus the the term overlap between the antonyms of c_k (Ant(c_k)) and $E(-k)^{t-1}$.[2] This count is then normalized. WNN is computed similarly, but with the two categories (k and $-k$) flipped:

$$\text{WNP}(c_k) = \left(\frac{n_1}{|\text{Syn}(c_k)|} + \frac{n_4}{|\text{Ant}(c_k)|}\right) \times log\left(\sum_{l=1}^{4} n_l\right) \quad (2)$$

[1] We used word embeddings of 200 dimensions generated with `word2vec`'s skip-gram algorithm over the English Gigaword. We used 0.6 for the threshold. These values were not tuned.

[2] We used 1st and 2nd order synonyms and antonyms. For example, for each c_k, we construct Syn(c_k) from its synonyms and the synonyms of the synonyms. $E(k)^0$ is set to the seed terms of k.

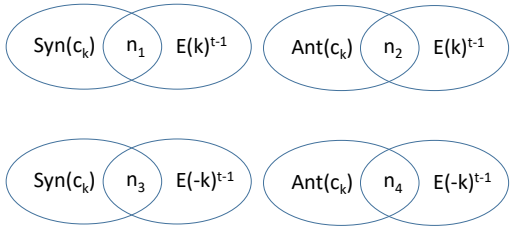


Figure 1: Illustration for the computation of the WNP and WNN scores for a term c_k, using the overlap between its synonyms (Syn(c_k)) and antonyms (Ant(c_k) vs. the category at hand (k) and its polar opposite ($-k$).

$$\text{WNN}(c_k) = (\frac{n_3}{|\text{Syn}(c_k)|} + \frac{n_2}{|\text{Ant}(c_k)|}) \times log(\sum_{l=1}^{4} n_l) \quad (3)$$

where n_1, n_2, n_3, and n_4 are the number of terms in the set intersections shown in Figure 1. The log component follows the intuition of (Riloff, 1996) to promote terms that are frequent; but here we adapt it to use the size of the overlap between WordNet synsets and the pools of known terms.

We use the score in Eq. 1 in multiple ways. First, we implement two "cautiousness" (Collins and Singer, 1999) constraints, i.e., we accept only candidate terms that: (a) have all the following three conditions true: EDP$\geq$ EDN, WNP$\geq$ WNN, EMP$\geq$ EMN (i.e., their association with the positive class is stronger than the one with the negative class under all resources); and (b) have at least one of the constraints satisfied: EDP$-$EDN $\equiv$ 1.0[3], WNP $-$ WNN $\geq \lambda_2$, EMP $-$ EMN$\geq \lambda_3$. Second, we use the score to estimate the quality of extraction patterns, as detailed next.

3.2. Pattern Learning

In this work, we define patterns similarly to McIntosh and Curran (2008), as five- or four-grams over surface tokens that include the term under consideration.

Our pattern learning algorithm expands upon existing methods by taking advantage of: (a) the set of expanded terms E, and (b) the polarity of the emotion dimensions to be learned. For each category k, each pattern (or template) t is assigned the following relevance weight:

$$\begin{gathered}\text{weight}(t \in \mathcal{Z}(k)) = \\ (\textstyle\sum_{c \in E(k)} \text{score}(t,c) - \sum_{c \in E(-k)} \text{score}(t,c)) \\ \times log(\textstyle\sum_c \text{frequency}(t,c))\end{gathered} \quad (4)$$

where frequency(t,c) captures the number of times a pattern t and a term c matched in the text; and score(t,c) measures the association between the pattern and the term c using the formula:

$$\text{score}(t,c) = log(\text{cscore}(c)\frac{p(t,c)^2}{p(t)p(c)} + 1) \quad (5)$$

Eq. 5 builds upon the squared mutual information (MI^2) formula of McIntosh and Curran (2008). We follow McIntosh and Curran (2008) by choosing squared MI over plain MI.[4] We weigh the MI^2 term by the association strength between term c and the current category k (i.e., cscore from Eq. 1). Further, we replace the hard mutual exclusive constraint from McIntosh and Curran (2008), which does not allow term and patterns to belong to multiple categories, with the soft constraint captured in Eq. 4, where opposite categories (k and $-k$) compete for pattern t.

Patterns are ranked in descending order of their weight, and the top M patterns are added to the cumulative pattern pool of category k in each epoch.

3.3. Term Learning

Lastly, we add new terms to the pool of known terms ($\mathcal{E}$) using the patterns previously learned. Terms are ranked by the following formula:

$$\begin{gathered}\text{weight}(c \in \mathcal{E}(k)) = \\ (\textstyle\sum_{t \in Z(k)} \text{score}(c,t) - \sum_{t \in Z(-k)} \text{score}(c,t)) \\ \times log(\textstyle\sum_t \text{frequency}(c,t))\end{gathered} \quad (6)$$

Eqs. 4 and 6 are nearly symmetrical for terms and patterns. Similarly, score(c,t) measures the association between the pattern and a term c using the formula:

$$\text{score}(c,t) = log(\text{tscore}(t)\frac{p(t,c)^2}{p(t)p(c)} + 1) \quad (7)$$

Eqs. 5 and 7 are symmetrical for terms and patterns, but in Eq. 7 tscore(t) is set to 1.0 because we rely solely on patterns from the pools of known patterns (i.e., no pattern expansion was implemented).

Similar to the previous step, the candidate terms are ranked in descending order of their weight, and the top N are added to the cumulative pool of terms.[5] Additionally, this step implements the "cautiousness" constraint from Section 3.1..

4. Experiments

Corpus: we are interested in learning from small datasets containing informal language. Here, we used a corpus of reviews for 1,439 sellers of e-cigarettes that we collected from Yelp. We downloaded all reviews from each page. The final corpus contains 1,600,151 tokens. The texts were stripped of URLs, then tokenized using CoreNLP (Manning et al., 2014). Note that our dataset contains only free text, without any hashtags or emoticons. We split data into training (75%) and testing (25%). We used the training dataset solely to tune the model's hyper parameters: λ_1, λ_2, and λ_3 from Section 3., as described later in this section. The tuning happened exclusively on the training dataset.

Terms and patterns: we consider terms to be single-word adjectives, nouns, verbs, or adverbs (e.g., "horrific"), and patterns to be 4- or 5-grams surrounding them (e.g., "avoid the _ burn lead" pattern captures the previous term).

[3]Note that the values of EDP and EDN are discretized edit distances, and can take only values of 0 or 1 (Gupta and Manning, 2014).

[4]Similar to (McIntosh and Curran, 2008), we observed that this formula performed better than $p(t|c)$ and regular MI.

[5]To mitigate the sensitivity to low frequencies, we set a candidate term's weight to 0 if $\sum_t \text{frequency}(c,t) \leq 3$ in Eq. 6. These terms are separately ranked using Eq. 1 instead.

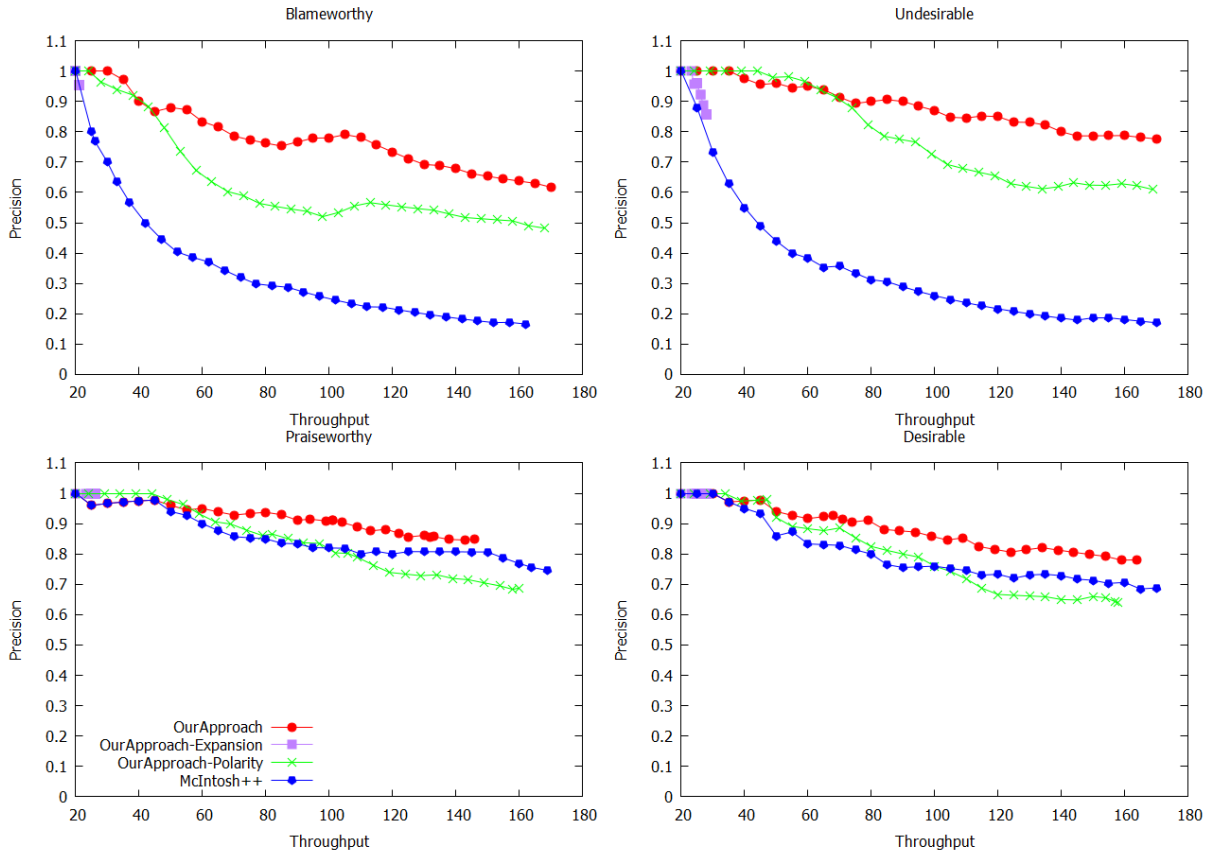


Figure 2: Precision/throughput curves of our approach and the three baselines. All systems were run for up to 30 epochs.

Evaluation measures: we report cumulative precision and throughput of terms for 30 epochs. Here, cumulative precision means the fraction of the words retrieved that are correct for the given category; cumulative throughput means the number of words that are retrieved by a system. All extracted terms were manually evaluated for their correctness, i.e., their membership in the corresponding category. There are three annotators (who were not authors), two of which are native speakers. We use the majority vote as the ground truth.

Baselines: we compare our system against three baselines: (a) *OurApproach – Expansion*: this baseline removes lines 4-5 in Algorithm 1 and uses the known terms ($\mathcal{E}$) rather than the expanded terms (E) for pattern ranking (Eq. 4); (b) *OurApproach – Polarity*: this model removes all polarity information: only the positive category is used in Eqs. 1, 2, 3, 4, and 6, and no cautionessness constraints in Section 3.1. and Section 3.3.; and (c) *McIntosh++*: our implementation of the mutually-exclusive approach of McIntosh and Curran (2008): no term expansion, no polarity, and categories are required to be mutually exclusive. Our extension to this system was to add log(frequency) to their ranking formula based on MI^2 (similar to Eq. 4 and 6); we found this performs better on our small data.

Hyper parameters: there are three hyper parameters in our approach: λ_1, λ_2 , and λ_3 (see Section 3.). Intuitively, we aim for a relatively high value for λ_1 because we prefer fewer, high-quality expanded terms (i.e., the expansion should be cautious). We aim for relatively low values for λ_2 and λ_3 because these are coupled with additional constraints in Section 3. that reduce the risk of introducing noise. We tuned all these parameters on the training dataset, and found that performance is best when $\lambda_1 = 0.5$, $\lambda_2 = 0.2$, $\lambda_3 = 0.1$.

Results: Figure 2 plots the cumulative term precision and throughput (i.e., number of terms learned in a given category) for our approach and the three baselines, for four emotion dimension categories. The figure shows that all our contributions are important. Term expansion yields a considerable improvement in both precision and throughput, especially for the negative emotion dimensions, which are more affected by sparsity. Polarity information consistently improves precision for all categories. Our approach has considerable higher precision than McIntosh++, at a small loss in throughput for two categories.

An ablation test (shown in Figure 3) of the resources used indicates that WordNet has the highest contribution to precision (e.g., yielding an increase of 15% (absolute) for *blameworthy*, and 20% for *undesirable*), and `word2vec` has the highest contribution to throughput (e.g., yielding an increase from 80 to 145 terms for *praiseworthy*). Edit distance has a small contribution to precision for *undesirable* and *desirable*, confirming that misspellings occur in informal texts, but it impacts negatively the *blameworthy* category, suggesting that it can also accumulate er-

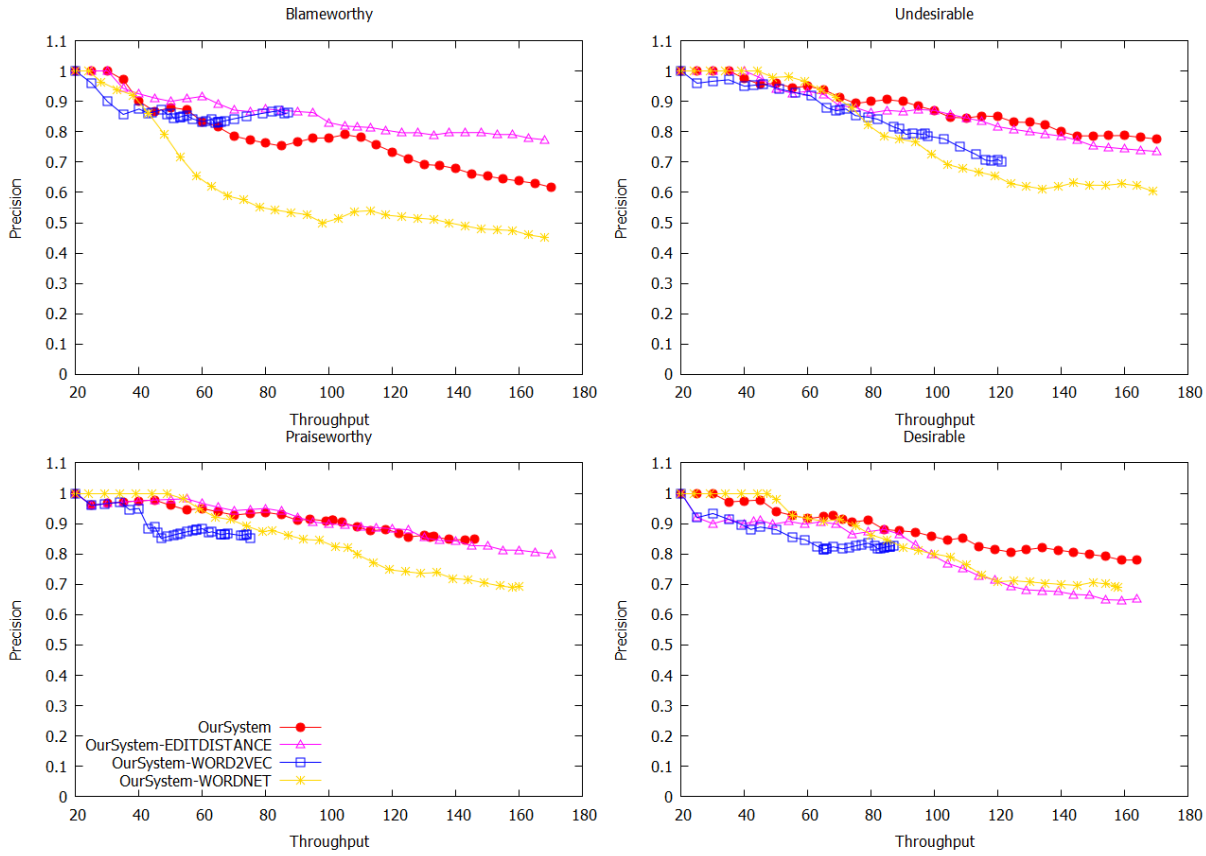


Figure 3: Ablation test for the three resources used in our approach: edit distance, `word2vec`, and WordNet. All systems were run for up to 30 epochs.

rors, e.g., by generating multiple spellings of terms that are incorrect.

5. Conclusion

We introduced a novel bootstrapping approach for the extraction of lexicons for polarized categories, in our case emotion dimensions of the OCC cognitive model of emotions (Ortony et al., 1990). We focused on small datasets containing informal texts, and made several contributions. First, we mitigated the sparsity of the data with a term expansion component that takes advantage of multiple resources: WordNet, word embeddings, and edit distance, and showed that these resources used have complementary contributions. Second, we addressed the semantic drift limitation of bootstrapping with multiple solutions that take advantage of the polarity of the classes to be learned. Our approach yields considerable higher precision than a traditional counter-training system (in some cases more than double), with only a small loss of throughput for some categories.

6. Bibliographical References

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Jr., E. R. H., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *AAAI*.

Collins, M. and Singer, Y. (1999). Unsupervised models for named entity classification. In *EMNLP*.

Gupta, S. and Manning, C. D. (2014). Improved pattern learning for bootstrapped entity extraction. In *CoNLL*.

Gupta, S. and Manning, C. D. (2015). Distributed representations of words to guide bootstrapped entity classifiers. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

Kozareva, Z. and Hovy, E. (2010). A semi-supervised method to learn and construct taxonomies using the web. In *EMNLP*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL*.

McIntosh, T. and Curran, J. R. (2008). Weighted mutual exclusion bootstrapping for domain independent lexicon and template acquisition. In *Proceedings of the Australasian Language Technology Association Workshop*.

McIntosh, T. (2010). Unsupervised discovery of negative categories in lexicon bootstrapping. In *EMNLP*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *ICLR Workshop*.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244.

Ortony, A., Clore, G. L., and Collins, A. (1990). *The Cognitive Structure of Emotions*. American Psychological Association.

Popescu, A.-M. and Etzioni, O. (2005). Extracting product features and opinions from reviews. In *EMNLP*.

Riloff, E. (1996). Automatically generating extraction patterns from untagged text. In *AAAI*.

Yangarber, R. (2003). Counter-training in discovery of semantic patterns. In *ACL*.

Yih, W., Zweig, G., and Platt, J. C. (2012). Polarity inducing latent semantic analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Sentiment-Stance-Specificity (SSS) Dataset: Identifying Support-based Entailment among Opinions.

Pavithra Rajendran, Danushka Bollegala, Simon Parsons
University of Liverpool, University of Liverpool, Kings College London
Pavithra.Rajendran@liverpool.ac.uk, danushka.bollegala@liverpool.ac.uk, simon.parsons@kcl.ac.uk

Abstract

Computational argumentation aims to model arguments as a set of premises that either support each other or collectively support a conclusion. We prepare three datasets of text-hypothesis pairs with support-based entailment based on opinions present in hotel reviews using a distant supervision approach. Support-based entailment is defined as the existence of a specific opinion (premise) that supports as well as entails a more general opinion and where these together support a generalised conclusion. A set of rules is proposed based on three different components — *sentiment*, *stance* and *specificity* to automatically predict support-based entailment. Two annotators manually annotated the relations among text-hypothesis pairs with an inter-rater agreement of 0.80. We compare the performance of the rules which gave an overall accuracy of 0.83. Further, we compare the performance of textual entailment under various conditions. The overall accuracy was 89.54%, 90.00% and 96.19% for our three datasets.

Keywords: argument mining, stance classification, structured argumentation

1. Introduction

Argument mining (Abbas and Sawamura, 2008; Palau and Moens, 2009) deals with the extraction of argument components and structures from natural language texts. It has drawn attention from both the argumentation and NLP communities with the introduction of ArgMining workshops [1]. In computational argumentation, an argument can be defined as a collection of premises together (linked argument) or individually (convergent argument) which are related to a conclusion (Palau and Moens, 2009). Each premise provides a *support* in the form of logical reasoning for, or evidence in support of, the conclusion to which it is connected.

It has been suggested that, in natural language texts, this support relation can be interpreted as meaning either (a) one premise is inferred from another premise (Janier et al., 2014) or (b) one premise provides evidence that supports another premise (Park and Cardie, 2014). In either case, it is natural to interpret the relationship as a form of entailment.

In this paper, we consider a subtype of entailment, which we term *support-based entailment*, where a support relation exists between the text and the hypothesis. Despite the unstructured nature of natural language texts, they provide meta-linguistic attributes such as stance, sentiment, and specificity that can be exploited for detecting support-based entailment.

We create a dataset of text-hypothesis pairs from opinions collected from a set of hotel reviews where the text provides support to the corresponding hypothesis. As an example, we consider online reviews that are comprised largely of opinionated texts that talk about various aspects of a product or service. Consider the examples shown in Fig. 1 where we have two different reviews with overall star ratings of 1 and 2. There we see a collection of *opinions*, that is sentence-level statements that talk about one or more aspects of a product or service. These are the basic units that we deal with in our work. Human annotation of argument structures and relation among them is a complicated task which is domain-dependent and hence manually annotating huge data is costly and difficult (Matthias and Stein, 2016). To combat this, we use a distant supervision approach by manually creating a set of rules based on meta-linguistic attributes such as *stance*, *sentiment* and *specificity*. These rules automatically label a set of sentences, which is then used to train a classifier for predicting support-based entailment.

Now, we give a few examples to explain how the three meta-linguistic attributes are useful. First, let us consider two opinions with the same sentiment as follows:

"not good enough for a Hotel charging these prices"

"the problem with the hotel is the staff"

Both these opinions have the same negative sentiment, but there exists no support or entailment relation between them. Suppose, we consider two opinions with the same stance (here, we refer stance as the standpoint taken towards a particular topic) as follows:

"the staff were helpful and polite"

"the staff was great"

Both these opinions have the same sentiment and stance towards the aspect *staff*. The only difference is that, in the first opinion the stance does not contain the stance expressed linguistically, whereas it is expressed in the second. Rajendran et al. (2017) use a supervised approach to classify opinions as implicit/explicit based on how the stance is expressed linguistically. This classification can help to relate the opinions as: the first opinion (text) supports as well as entails the second opinion (hypothesis).

Suppose, we consider two opinions as follows:

"the staff was great"

[1] https://argmining2017.wordpress.com/

"overall, great service!"

While these two opinions have the same positive sentiment and both are explicit opinions, the first opinion has a stance towards the aspect *staff* and the second opinion has a stance towards the aspect *service*. In such cases, sentiment and stance alone would be insufficient. If we were given a knowledge base that can relate *staff* with *service*, then, it can be useful to relate these opinions as: first opinion (text) supports as well as entails the second opinion (hypothesis). The remaining sections are given below.

- Section 2. gives an overview of the related works.
- Section 3. gives a description about the support-based entailment relation and the three meta-linguistic attributes – *sentiment*, *stance* and *specificity* that are useful to predict the same.
- Section 4. gives a description about the different support-based entailment rules (SER) that are proposed to predict the support-based entailment relation.
- Section 5. describes the SSS dataset that we create using opinions extracted a set of hotel reviews and SER.
- Section 6. describes the experiments carried out on the SSS dataset using existing textual entailment methods and the results are reported.
- Section 7. presents the conclusion of our work.

2. Related Work

A detailed study of previous work in argument mining has been presented by Lippi and Torroni (Lippi and Torroni, 2015; Lippi and Torroni, 2016). Few papers have dealt with the problem of mining arguments from online reviews (Wyner et al., 2012; Villalba and Saint-Dizier, 2012). Using computational argumentation techniques to deal with real-world problems such as opinion mining (Dragoni et al., 2016), sentiment analysis (Rajendran et al., 2016) and detecting deceptive reviews (Cocarascu and Toni, 2016) has been tackled so far. Boltuzic et al. (2014) combine stance, textual entailment and semantic similarity to identify relations between arguments and comments presented in a debate. We propose a way of detecting support-based entailment such that a specific opinion *supports* as well as *entails* a corresponding generalised opinion. Cabrio and Villata (2012) consider entailment to be a form of support relation that occurs between arguments present in debates. We differ from this work, since we define a support relation based on structured argumentation using three different components – sentiment, stance and specificity that can also predict entailment. Previously, Grosse et al. (2012) have explored constructing opinion analysis trees that aggregate opinions present in a Twitter dataset based on the specificity property. Our work is not to aggregate opinions but to construct argument structures that are able to persuade an audience towards a particular conclusion.

Stance classification (Mohammad et al., 2016; Augenstein et al., 2016; Anand et al., 2011) relates to classifying whether a given statement is for or against a known target, which is explicitly stated or not. Sobhani et al. (2016) investigate the relation between stance and sentiment on a set of Twitter data where the target need not be present explicitly. Ebrahimi et al. (2016) propose a model that integrates stance, sentiment and target features jointly as a three way interaction for classifying stance in a set of tweets. We use sentiment as a way of identifying stance present in opinions where the target is explicitly present. Also, we are interested in how the stance is expressed and use this as a feature to identify support-based entailment relation.

Textual entailment deals with identifying whether a hypothesis can be inferred from a given text, which is directional and differs from semantic similarity measures. Yokote et al. (2012) propose a model that transforms similarity measures into a non-linear transformation for predicting textual entailment. Zanzotto et al. (2005) investigate on identifying patterns based on subject verb relation to identifying entailment. In their paper, they argue that the logical entailment present between the text and hypothesis is not captured properly. In contrast, we are interested in a subtype of entailment that can predict the *support* relation based on argumentation theory.

3. Support-based Entailment

The three components of the proposed method are explained below. Based on these, we manually identify a set of support-based entailment rules (SER) for predicting the support-based entailment between a text (T) and a hypothesis (H).

Opinion and Premise: We take an *opinion* to be a sentence-level statement, which might be either positive or negative in sentiment, and talks about an aspect or several aspects of a product/service. For example, *service*, *location* are aspects of hotels in the hotel domain.

We consider a *premise* as a simple atomic unit that talks about one particular aspect. Hence, any opinion that talks about several aspects can be considered as a collection of several premises that may or may not be related.

Sentiment: The positive/negative sentiment of an opinion is taken into consideration. We ignore objective opinions as it cannot be used to match the global sentiment (overall star rating). As a first step we only consider TH pairs as opinions with the same sentiment.

Stance: Previously (Rajendran et al., 2017), we explained how to classify the stance expressed by an opinion as implicit/explicit. In both, the stance (for/against) is expressed by the reviewer, but explicit opinions have the stance explicitly expressed using (1) direct approval/disapproval or (2) words/phrases by the reviewer that have a stronger intensity of expression with respect to the topic in discussion. General cues such as *recommend*, *great*, *worst* indicate direct expressions and are useful in identifying explicit opinions. Specific cues that are related to domain-based targets can help in identifying implicit opinions. For

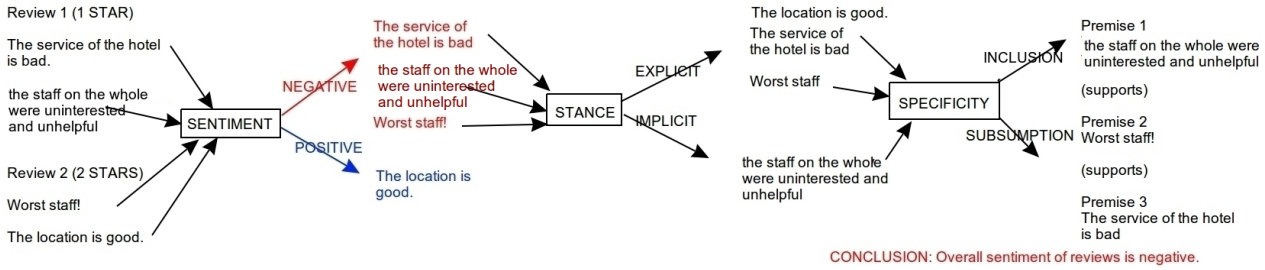


Figure 1: Opinions from two reviews are extracted and distinguished based on their local sentiment, stance, and specificity. All opinions that do not match the overall sentiment of the reviews are discarded. The rest of the opinions are then classified as explicit or implicit and using subsumption and inclusion relation, these opinions are combined such that one supports another.

example, *lightweight laptop* has a positive stance towards the target *laptop* whereas *the storyline of the book is lightweight* has a negative stance towards the target *book*. Also, opinions can express justification such as reasons that express stance implicitly. An example is provided in Fig. 1.

Specificity: A knowledge base (KB) is created based on the domain and the aspects present where one aspect is a sub-class of the other. Given such a KB, we describe three domain-based ontology relations between two premises that make use of the implicit/explicit nature of the opinions in which the aspects are present.

Suppose an aspect is present in a given opinion, we consider the opinion to contain a premise about that particular aspect. We thus represent each such premise as $\mathcal{P}(attr, op, stance)$ where $attr$ is the *aspect* present in an opinion *Op* which is classified as implicit/explicit and represented as *Stance*. We define the three relations below.

Def. 1 (Subsumption, $\sqsubseteq_{sub}$). Two premises present within an opinion, $\mathcal{P}(attr1, op1, exp) \sqsubseteq_{intrasub}$ (intra-subsumption) $\mathcal{P}(attr2, op1, exp)$ if *attr1* is a sub-class of *attr2*.
Two premises present in two different opinions, $\mathcal{P}(attr1, op1, exp) \sqsubseteq_{intersub}$ (inter-subsumption) $\mathcal{P}(attr2, op2, exp)$ if *attr1* is a sub-class of *attr2*.

Def. 2 (Inclusion, $\sqsubseteq_{inc}$). Two premises, one present in an implicit opinion and the other present present in an explicit opinion satisfies $\mathcal{P}(attr1, op1, imp) \sqsubseteq_{inc}$ (is-inclusive of) $\mathcal{P}(attr2, op2, imp)$ such that *attr1* and *attr2* are the same.

Def. 3 (Equivalence, $\equiv$). $\mathcal{P}(attr1, op1, exp) \equiv$ (equivalent) $\mathcal{P}(attr2, op2, exp)$ if *attr1* and *attr2* are same. $\mathcal{P}(attr1, op1, imp) \equiv$ (equivalent) $\mathcal{P}(attr2, op2, imp)$ if *attr1* and *attr2* are same.

4. Support-based Entailment Rules (SER)

Our definition of a premise states that an opinion with *n* aspects contains *n* premises. For example,

*"and the **service** from the **staff** was extremely poor"*

contains two premises, one about the *service* and the other about the *staff*.
We are not interested in decomposing the opinion into different premises based on the linguistic structure but instead focus on identifying text-hypothesis (TH) pairs. Our motivation behind creating the dataset is to identify TH pairs that can help in forming argument structures from these premises using implicit and explicit opinions. A simple structure would be of the form $(implicit_1, explicit_1, explicit_2)$ with different relations as follows:

- Inclusion relation between a premise present in $implicit_1$ and a premise in $explicit_1$. Both premises are about the same aspect.
- Intra-subsumption relation between two different premises present within $explicit_1$. The same can be said for $explicit_2$.
- Inter-Subsumption/Equivalence relation between a premise in $explicit_1$ and a premise in $explicit_2$.

All these relations require two premises. For every opinion (text or hypothesis), our rules are designed to consider atmost two premises at a time and whether those two premises are related or not. For example,

Op 1: *the **hotel** was exceptionally clean, the* service *was very friendly at all times and nothing seemed to be too much and the* location *is quiet and peaceful...*

Op 2: *this is very nice **hotel** that exceeded our expectations*

Op1 contains three premises $\mathcal{P}(hotel, Op1, imp)$, $\mathcal{P}(service, Op1, imp)$ and $\mathcal{P}(location, Op2, imp)$. Op2 contains one premise $\mathcal{P}(hotel, Op2, exp)$.
In the above example, we can consider atmost two premises at a time, which means we have the following premise pairs:-

- $(\mathcal{P}(hotel, Op1, imp), \mathcal{P}(service, Op1, imp))$
- $(\mathcal{P}(hotel, Op1, imp), \mathcal{P}(location, Op2, imp))$
- $(\mathcal{P}(service, Op1, imp), \mathcal{P}(location, Op2, imp))$

Rule	# Aspects (Text)	#Aspects (Hypothesis)	Text	Hypothesis	Relation
Rule 1	>1	>1	*a* $\sqsubseteq_{intrasub}$ *b*	*c* $\sqsubseteq_{intrasub}$ *d*	*b* $\sqsubseteq_{intersub}$ *d* or *b* $\equiv$ *d* and *a* $\sqsubseteq_{intersub}$ *c* or *a* $\equiv$ *c*
Rule 2	>1	1	*a* $\sqsubseteq_{intrasub}$ *b*	*c*	*b* $\sqsubseteq_{intersub}$ *c* or *b* $\equiv$ *c*
Rule 3	>1	1	*a,b* and not related	*c*	*a* $\sqsubseteq_{intersub}$ *c* and *b* $\sqsubseteq_{intersub}$ *c*
Rule 4	>1	1	*a,b* and not related	*c*	*a* $\equiv$ *c* or *b* $\equiv$ *c*
Rule 5	1	1	*a*	*c*	*a* $\sqsubseteq_{intersub}$ *c*
Rule 6	1	1	*a*	*c*	*a* $\equiv$ *c*
Rule 1	1	1	*a*	*c*	*a* $\sqsubseteq_{inc}$ *b*
Rule 2	1	>1	*a*	*b* $\sqsubseteq_{intrasub}$ *c*	*a* $\sqsubseteq_{inc}$ *b*
Rule 3	>1	>1	*a,b* and not related	*c* $\sqsubseteq_{intrasub}$ *d*	*a* $\sqsubseteq_{inc}$ *c* and *b* $\sqsubseteq_{inc}$ *d*
Rule 4	>1	1	*a,b* and not related	*c*	*a* $\sqsubseteq_{inc}$ *c* or *b* $\sqsubseteq_{inc}$ *c*
Rule 5	1	>1	*a*	*b,c* and not related	*a* $\sqsubseteq_{inc}$ *b* or *a* $\sqsubseteq_{inc}$ *c*
Rule 6	>1	>1	*a,b* and not related	*c,d* and not related	*a* $\sqsubseteq_{inc}$ *c* or *b* $\sqsubseteq_{inc}$ *d*

Table 1: Each proposed rule for subsumption (top) and inclusion (bottom) relation is presented. The number of aspects (premises) that must be present in text and hypothesis is given. Conditions that must hold true in text, hypothesis and between them is also given. Here, we consider *a*,*b*,*c* and *d* to represent the aspects (premises) present.

Rule	Text	Hypothesis	Relation
Rule 1	and the **service** from the **staff** was extremely poor (*staff*$_{text}$ $\sqsubseteq_{intrasub}$ *service*$_{text}$)	it is the worst **service** i have seen in a five star **hotel** (*service*$_{hyp}$ $\sqsubseteq_{intrasub}$ *hotel*$_{hyp}$)	*service*$_{text}$ $\sqsubseteq_{intersub}$ *hotel*$_{hyp}$, *staff*$_{text}$ $\sqsubseteq_{intersub}$ *service*$_{hyp}$, *service*$_{text}$ $\equiv$ *service*$_{hyp}$
Rule 2	**location** of the **hotel** is really well placed - you're in the middle of everything (*location*$_{text}$ $\sqsubseteq_{intrasub}$ *hotel*$_{text}$)	overall a very good **hotel** (*hotel*$_{hyp}$)	*hotel*$_{text}$ $\equiv$ *hotel*$_{hyp}$
Rule 3	weak **service** for very high **prices** (*service*$_{text}$, *prices*$_{text}$)	i would not plan to stay at this **hotel** again (*hotel*$_{hyp}$)	*service*$_{text}$ $\sqsubseteq_{intersub}$ *hotel*$_{hyp}$, *prices*$_{text}$ $\sqsubseteq_{intersub}$ *hotel*$_{hyp}$
Rule 4	weak **service** for very high **prices** (*service*$_{text}$, *prices*$_{text}$)	however this is probably the worst **service** we have ever experienced (*service*$_{hyp}$)	*service*$_{text}$ $\equiv$ *service*$_{hyp}$
Rule 5	great **location** (*location*$_{text}$)	i absolutely loved this **hotel** (*hotel*$_{hyp}$)	*location*$_{text}$ $\sqsubseteq_{intersub}$ *hotel*$_{hyp}$
Rule 6	i absolutely loved this **hotel** (*hotel*$_{text}$)	overall a very good **hotel** (*hotel*$_{hyp}$)	*hotel*$_{text}$ $\sqsubseteq_{intersub}$ *hotel*$_{hyp}$
Rule 1	**hotel** infrastructure is in need of serious upgrading (*hotel*$_{text}$)	so believe me when i say do not stay at this **hotel** (*hotel*$_{hyp}$)	*hotel*$_{text}$ $\sqsubseteq_{inc}$ *hotel*$_{hyp}$
Rule 2	the **staff** that we encountered were very friendly and helpful (*staff*$_{text}$)	and the **service** from the valet and front desk **staff** is very good (*staff*$_{hyp}$ $\sqsubseteq_{intrasub}$ *service*$_{hyp}$)	*staff*$_{text}$ $\sqsubseteq_{inc}$ *staff*$_{hyp}$
Rule 4	to their credit the management was more responsive and very apologetic for the condition of my **room** and the rude treatment by their **staff** (*room*$_{text}$, *staff*$_{text}$)	dissapointed from the **room** (*room*$_{hyp}$)	*room*$_{text}$ $\sqsubseteq_{inc}$ *room*$_{hyp}$
Rule 5	the **staff** was not friendly nor helpful (*staff*$_{text}$)	overall its a dark dated **hotel** let down badly by the unhelpful and rude **staff** (*hotel*$_{text}$, *staff*$_{hyp}$)	*staff*$_{text}$ $\sqsubseteq_{inc}$ *staff*$_{hyp}$

Table 2: Examples for different rules satisfying subsumption (top) and inclusion (bottom) relations.

- ($\mathcal{P}(hotel, Op2, exp)$,$\mathcal{P}(hotel, Op1, imp)$)
- ($\mathcal{P}(hotel, Op2, exp)$,$\mathcal{P}(service, Op1, imp)$)
- ($\mathcal{P}(hotel, Op2, exp)$,$\mathcal{P}(location, Op2, imp)$)

Out of these, ($\mathcal{P}(hotel, Op2, exp)$, $\mathcal{P}(hotel, Op1, imp)$) is related by the inter-subsumption relation. Further, if an opinion contains more than one premise, then rules based on a single premise cannot be considered. In the above example, Op2 can be considered for rules based on a single premise whereas Op1 cannot be considered.

Let us consider another case where a text that contains 3 premises *a,b* and *c* with *a* and *b* related. For a given hypothesis, one rule will be satisfied based on the related premises *a* and *b* while some other rule might be satisfied based on two premises that are not related (eg. *a* and *c*). We predict the support-based entailment in a TH pair if at least one of the rules is satisfied. This is to ensure that there are no

Data	Rev	Exp	Imp	Sub	Inc
FA	369	264	720	Rule 1: 14	Rule 1: 271
				Rule 2: 138	Rule 2: 25
				Rule 3: 27	Rule 3: 6
				Rule 4: 218	Rule 4: 619
				Rule 5: 193	Rule 5: 147
				Rule 6: 218	Rule 6: 344
SA	707	1001	4359	Rule 1: 92	Rule 1: 1790
				Rule 2: 566	Rule 2: 137
				Rule 3: 82	Rule 3: 55
				Rule 4: 344	Rule 4: 3418
				Rule 5: 842	Rule 5: 933
				Rule 6: 1834	Rule 6: 1799
UA	3271	564	5933	Rule 1: 34	Rule 1: 3708
				Rule 2: 467	Rule 2: 148
				Rule 3: 55	Rule 3: 33
				Rule 4: 119	Rule 4: 4726
				Rule 5: 428	Rule 5: 2189
				Rule 6: 1354	Rule 6: 3053

Table 3: In each dataset: total number of reviews (Rev) present, total number of explicit opinions (Exp) and implicit opinions (Imp) found and total number of TH pairs satisfying each rule in SER based on subsumption (Sub) and inclusive (Inc) relation is present.

duplicate pairs created.

If a text/hypothesis can contain a single premise or atmost two premises, then nine different combinations are possible based on whether inter-subsumption is present in the text/hypothesis or not. This holds for both subsumption-based and inclusion-based rules. Based on our definition of *support-based entailment*, a specific premise supports a more generalised premise. Thus, we ignore rules based on subsumption relation that look into hypothesis containing non-related premises. This means we have only six different combinations to deal with. Moreover, implicit opinions (text) cannot have any inter-subsumption relation and hence three of those combinations are ruled out. Thus, we have a total of six different rules based on each of inclusion and subsumption. These rules are present in Table 1.

Given two explicit opinions of same sentiment, we apply the rules based on the subsumption relation. Fistly, we check for intra-subsumption related premises within each text and hypothesis and apply the corresponding rules. If not, rules based on unrelated and single premises are applied. Given an implicit opinion and an explicit opinion of same sentiment, we apply the rules based on the inclusion relation. Single premises within the text and hypothesis are checked first and the corresponding rules are applied. Otherwise, hypotheses with related premises are considered and the rule is applied. Then, text and hypothesis with unrelated premises are considered and the rules are applied accordingly.

5. Sentiment-Stance-Specificity[2] (SSS) Dataset

We use an existing hotel reviews corpus, ArguAna (Wachsmuth et al., 2014b) to create our datasets. The data for each hotel contains a balanced set of reviews based on the overall star rating for that hotel. Each review contains manually annotated local sentiment of the statements (pos, neg or obj), aspects present and the overall star rating.

First, we create a knowledge base using a list of aspects extracted from the ArguAna corpus. For example, *(Location $\sqsubseteq_{sub}$ Hotel), (Service $\sqsubseteq_{sub}$ Hotel), (Cleanliness $\sqsubseteq_{sub}$ Hotel), (Staff $\sqsubseteq_{sub}$ Service), (Restaurant service $\sqsubseteq_{sub}$ Service) etc.*

We used the manually annotated dataset of 1288 implicit/explicit opinions created in (Rajendran et al., 2017) which was annotated by two annotators with an inter-rater agreement of Cohen's Kappa = 0.70. Finally, three different dataset were created for our experiment using the proposed rules (few examples in Table 2):

1. **Fully annotated (FA)** This contains a balanced set of 369 reviews from 15 different hotels present in the ArguAna corpus. As explained previously, the local sentiment of statements and aspects present in them are manually annotated. Further, using the definitions from (Rajendran et al., 2017), the extracted opinions are manually annotated as explicit or implicit. There are 264 explicit opinions and 720 implicit opinions present. The SER rules predicted 2220 TH pairs with support-based entailment.

2. **Semi-annotated (SA)** This contains a balanced set of 707 reviews from 33 different hotels present in the ArguAna corpus. Here, the extracted opinions are automatically classfied as explicit or implicit using an SVM-based classifier (Rajendran et al., 2017) with the following features:

   - Surface based features - Unigrams, bigrams and adjective-noun pairs (count of adjective-noun pairs present).
   - Average embedding based feature - For each word, we use the Glove-based (Pennington et al., 2014) word embedding and average these embeddings for an opinion.

   We train a linear SVM classifier using the Scikit-learn[3] package for an undersampled dataset containing 494 explicit opinions and 894 implicit opinions respectively. We use this undersampled data as our training data. We performed a cross-validation on the unbalanced data containing 494 explicit opinions and 1367 implicit opinions to obtain the cost parameter value C of the SVM as 1.0. The cross-validation accuracy of the training data using the above mentioned features is 80% for explicit opinions and 87% for implicit opinions respectively.

   There are 1001 explicit opinions and 4359 implicit opinions present. The SER rules predicted 11892 TH pairs with support-based entailment.

[2] goo.gl/cfBHc7

[3] scikit-learn.org

Experiment	FA	SA	UA
SER	89.54	90.00	96.19
Non-SER	76.18	72.69	88.01
Subsumption based SER	81.63	75.82	92.11
Subsumption based Non-SER	73.91	67.93	86.21
Inclusion based SER	95.83	96.49	97.68
Inclusion based NON-SER	76.87	73.84	88.31
Implicit-Explicit Entailment	75.94	71.03	87.89
Subsumption			
-Rule 1	100.0	83.69	100.0
-Rule 2	86.95	92.40	96.14
-Rule 3	44.44	52.43	80.0
-Rule 4	89.44	93.89	99.15
-Rule 5	62.69	46.67	83.64
-Rule 6	86.69	81.35	92.17
Inclusion			
-Rule 1	92.61	93.74	94.76
-Rule 2	96.0	95.62	96.62
-Rule 3	100.0	94.59	100.0
-Rule 4	97.25	98.50	98.47
-Rule 5	89.79	92.60	95.56
-Rule 6	95.63	97.72	98.59
Random sentiment (SER)	45.62	45.31	47.98
Random sentiment (Non-SER)	38.64	36.37	44.02

Table 4: An experiment was run on each dataset by (a) SER — TH pairs satisfying either of the six subsumption or six inclusion rules (b) Non-SER — TH pairs that do not satisfy any of the 12 rules. (c) Subsumption and Inclusion — TH pairs satisfying each individual rule and (d) Random sentiment — assigning sentiment of opinions present in TH pairs of SER and Non-SER randomly. Accuracy is reported.

3. **Unannotated (UA)** Reviews from 30 different hotels that are unannotated and not present in the ArguAna corpus are used. Here, the reviews are unbalanced. For each statement, local sentiment is automatically classfied as positive, negative or objective using the SVM-based classfier used in the ArguAna (Wachsmuth et al., 2014a) tool. All statements predicted as positive or negative were considered as opinions. We extract a list of aspects manually annotated in the ArguAna corpus and use this to identify aspects present in opinions. The opinions are automatically classfied as implicit or explicit as mentioned for the previous dataset.

   There are 564 explicit opinions and 5933 implicit opinions present. The SER rules predicted 16314 TH pairs with support-based entailment.

6. Experiments and Results

6.1. Performance of SER

In each of the above datasets, we predicted support-based entailment relation using the SER and present the total number of predicted cases in Table. 3. We extracted 160 TH pairs based on the SER as well as those that do not satisfy them. The proportion of TH pairs based on the SER is higher than those that do not satisfy them. We do this to understand whether the pairs extracted using SER rules are accepted by human annotators as well. Two annotators were asked to manually annotate whether the pairs satisfy support-based entailment or not. No information about the rules were provided. The inter-rater agreement was calculated using Cohen's Kappa as 0.80. To test the performance of the SER, we took the intersection of the two annotations as the ground truth data and the accuracy of the SER prediction was 0.83. We also considered the union of the two annotations as the ground truth data which gave the accuracy of the SER prediction as 0.93.

6.2. Performance of Textual Entailment

We use the Excitement Open Platform (EOP) (Magnini et al., 2014) to automatically predict textual entailment in support-based entailment relation and investigate using three different training sets – standard RTE-3 (Giampiccolo et al., 2007), SICK (Marelli et al., 2014) and EXCITEMENT (Kotlerman et al., 2015). The EOP tool takes a text and a hypothesis as input and predicts whether text (T) entails the hypothesis (H) or not. We use the TH pairs that are predicted as support-based entailment using the 12 different SER (Table. 1). Four different entailment decision algorithms (EDA) present in the EOP were used to test the support-based entailment present in the *Fully Annotated* dataset – MaxEntClassificationEDA, AdArteEDA, EditDistanceEDA and PSOEDA. Among these the MaxEntClassificationEDA which is based on the maximum entropy classifier gave the best performance with the RTE-3 dataset and overall accuracy of 89.54 % on the *FA* dataset and hence we use this classifier and the training data for other experiments.

We evaluate the performance of automatically predicting entailment by conducting the following experiments on the three different datasets. The accuracy of correct prediction in each of these experiments is listed in Table. 4 and we describe the experiments below.

Subsumption based SER Based on subsumption rules, two explicit opinions are paired with each other.

Subsumption based Non-SER Two explicit opinions are paired with each other if they do not match any of the subsumption rules.

Inclusion based SER Based on inclusion rules, an implicit opinion is paired with an explicit opinion.

Inclusion based Non-SER An implicit opinion is paired with an explicit opinion, if it does not match any of the inclusion rules.

SER We use pairs extracted in both **Subsumption based SER** and **Inclusion based SER**.

Non-SER We use pairs extracted in both **Subsumption based Non-SER** and **Inclusion based Non-SER**.

Subsumption Text-hypothesis pairs are extracted according to each individual subsumption rule.

Inclusion Text-hypothesis pairs are extracted according to each individual inclusive rule.

Implicit-Explicit Entailment Here, we predict entailment by pairing an explicit opinion with an implicit opinion as text and hypothesis without any rules. The only condition is that both must be of the same sentiment. This is to understand how textual entailment is able to differentiate between explicit and implicit opinions.

Random sentiment For each opinion in each pair present in SER and Non-SER, we randomly assign a local sentiment and predict *support based entailment* relation based on this misinformation.

From Table. 6. we can observe that the overall accuracy of SER outperforms that of Non-SER, which shows that our method is effective for predicting support-based entailment across all datasets. The individual cases, case 3 and 5 in the subsumption category do not perform better than the remaining cases. One reason could be that these two cases are strictly based upon the subsumption relation whereas the rest of them are a combination of both the subsumption and the equivalence relation. Given that these two cases are strictly based on the subsumption relation, it is evident that textual entailment does not depend on the domain ontology and does not consider specificity as a property for prediction.

There is not much difference among the cases present in inclusion, mainly because we differentiate between identical aspects based on the implicit/explicit opinion classification. It is best to compare the accuracy of inclusion-based SER with implicit-explicit entailment to analyse how the implicit/explicit classification affects textual entailment. The performance of inclusion-based SER is better and means that implicit/explicit opinion classification helps in better prediction.

We also experimented by randomly assigning incorrect sentiment (random sentiment baseline) and as expected the accuracy was lowered in comparison with SER.

It has to be noted that the inconsistency in the textual entailment results (Table. 6.) may be higher for the unannotated dataset, even though the results are higher. This is due to the following reasons: (1) the sentiment of the opinions as well as implicit/explicit classification are predicted automatically and (2) only a limited number of aspects are identified.

7. Conclusion

We present three datasets of TH pairs based on a subtype of entailment, which we term as support-based entailment that predicts the support relation between a specific premise and a generalised premise using sentiment, stance and specificity. A distant supervision approach is carried out by using a set of proposed rules based on three components — *sentiment*, *stance* and *specificity*. The performance of these rules against manually annotated 160 TH pairs is measured in terms of accuracy as 0.83. Experiments on the three datasets for the textual entailment task shows that the rules are able to predict the entailment relation but existing textual entailment method is not able to capture support-based entailment. We believe that our datasets will be useful to expedite research in argument mining.

8. Future Work

As part of future work, manually evaluating the unannotated/semi-annotated datasets would be a costly task. Instead, using semi-supervised approaches for automatically classifying implicit/explicit opinions can help in reducing the noise in labels. These datasets can also be useful for learning deep-learning models for predicting *support-based entailment* relation. We will need to evaluate whether such deep-learning models are able to capture the relation without any information such as *sentiment*, *stance* and *target* given explicitly. As of now, we consider only aspects that are explicitly present in an opinion. Given that a lot of existing work (Wang et al., 2011; Hai et al., 2011) in NLP have dealt with identifying explicit and implicit aspects present in online reviews, our work can benefit from this. Another direction for future work is to use the dataset to create argument structures similar to OVA+ structures (Janier et al., 2014).

9. Bibliographical References

Abbas, S. and Sawamura, H. (2008). A first step towards argument mining and its use in arguing agents and its. In *KES*, pages 149–157.

Anand, P., Walker, M., Abbott, R., Tree, J. E. F., Bowmani, R., and Minor, M. (2011). Cats rule and dogs drool!: Classifying stance in online debate. In *WASSA*, pages 1–9.

Augenstein, I., Rocktaschel, T., Vlachos, A., and Bontcheva, K. (2016). Stance detection with bidirectional conditional encoding. In *EMNLP*, pages 876–885.

Boltuzic, F. and Snajder, J. (2014). Back up your stance: Recognizing arguments in online discussions. In *ArgMining@ ACL*, pages 49–58.

Cabrio, E. and Villata, S. (2012). Combining textual entailment and argumentation theory for supporting online debates interactions. In *ACL*, pages 208–212.

Cocarascu, O. and Toni, F. (2016). Detecting deceptive reviews using argumentation. In *PrAISe*, pages 1–8.

Dragoni, M., Pereira, C. D. C., Tettamanzi, A. G., and Villata, S. (2016). Smack: An argumentation framework for opinion mining. In *IJCAI*, pages 4242–4243.

Ebrahimi, J., Dou, D., and Lowd, D. (2016). A joint sentiment-target-stance model for stance classification in tweets. In *COLING*, pages 2656–2665.

Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, B. (2007). The third pascal recognizing textual entailment challenge. In *ACL-PASCAL*, pages 1–9.

Grosse, K., Chesñevar, C. I., and Maguitman, A. G. (2012). An argument-based approach to mining opinions from twitter. In *AT*, pages 408–422.

Hai, Z., Chang, K., and Kim, J.-j. (2011). Implicit feature identification via co-occurrence association rule mining. *CiCLing*, pages 393–404.

Janier, M., Lawrence, J., and Reed, C. (2014). Ova+: an argument analysis interface. In *COMMA*, pages 463–464.

Kotlerman, L., Dagan, I., Magnini, B., and Bentivogli, L. (2015). Textual entailment graphs. *Natural Language Engineering*, 21:699–724.

Lippi, M. and Torroni, P. (2015). Argument mining: A machine learning perspective. In *TAFA*, pages 163–176.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. *TOIT*, 16(2):10.

Magnini, B., Zanoli, R., Dagan, I., Eichler, K., Neumann, G., Noh, T.-G., Pado, S., Stern, A., and Levy, O. (2014). The excitement open platform for textual inferences. In *ACL*, pages 43–48.

Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., and Zamparelli, R. (2014). A sick cure for the evaluation of compositional distributional semantic models. In *LREC*, pages 216–223.

Matthias, K. A.-K. H. W. and Stein, H. J. K. B. (2016). Cross-domain mining of argumentative text through distant supervision. In *NAACL-HLT*, pages 1395–1404.

Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X.-D., and Cherry, C. (2016). A dataset for detecting stance in tweets. In *LREC*, pages 3945–3952.

Palau, R. M. and Moens, M.-F. (2009). Argumentation mining: the detection, classification and structure of arguments in text. In *ICAIL*, pages 98–107.

Park, J. and Cardie, C. (2014). Identifying appropriate support for propositions in online user comments. In *ACL*, pages 29–38.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Rajendran, P., Bollegala, D., and Parsons, S. (2016). Assessing weight of opinion by aggregating coalitions of arguments. In *COMMA*, pages 431–438.

Rajendran, P., Bollegala, D., and Parsons, S. (2017). Identifying argument based relation properties in opinions. In *PACLING*, page to appear.

Sobhani, P., Mohammad, S., and Kiritchenko, S. (2016). Detecting stance in tweets and analyzing its interaction with sentiment. In *SEM*, pages 159–169.

Villalba, M. P. G. and Saint-Dizier, P. (2012). A framework to extract arguments in opinion texts. *IJCINI*, 6:62–87.

Wachsmuth, H., Trenkmann, M., Stein, B., and Engels, G. (2014a). Modeling review argumentation for robust sentiment analysis. In *COLING*, pages 553–564.

Wachsmuth, H., Trenkmann, M., Stein, B., Engels, G., and Palakarska, T. (2014b). A review corpus for argumentation analysis. In *CICLing*, pages 115–127.

Wang, H., Lu, Y., and Zhai, C. (2011). Latent aspect rating analysis without aspect keyword supervision. In *SIGKDD*, pages 618–626.

Wyner, A., Schneider, J., Atkinson, K., and Bench-Capon, T. J. (2012). Semi-automated argumentative analysis of online product reviews. In *COMMA*, pages 43–50.

Yokote, K., Bollegala, D., and Ishizuka, M. (2012). Similarity is not entailment - jointly learning similarity transformations for textual entailment. In *AAAI*, pages 1720–1726.

Zanzotto, F. M., Pazienza, M. T., and Pennacchiotti, M. (2005). Discovering entailment relations using textual entailment patterns. In *ACL*, pages 37–42.

Resource Creation Towards Automated Sentiment Analysis in Telugu (a low resource language) and Integrating Multiple Domain Sources to Enhance Sentiment Prediction

Gangula Rama Rohit Reddy, Radhika Mamidi
LTRC IIIT Hyderabad, LTRC IIIT Hyderabad
Hyderabad Telangana India, Hyderabad Telangana India
ramarohitreddy.g@research.iiit.ac.in, radhika.mamidi@iiit.ac.in

Abstract

Understanding the polarity or sentiment of a text is an important task in many application scenarios. Sentiment Analysis of a text can be used to answer various questions such as election prediction, favouredness towards any product etc. But the sentiment analysis task becomes challenging when it comes to low resource languages because the basis of learning sentiment classifiers are annotated datasets and annotated datasets for non-English texts hardly exists. So for the development of sentiment classifiers in Telugu, we have created corpora "Sentiraama" for different domains like movie reviews, song lyrics, product reviews and book reviews in Telugu language with the text written in Telugu script. In this paper, we describe the process of creating the corpora and assigning polarities to them. After the creation of corpora, we trained the classifiers that yields good classification results. Typically a sentiment classifier is trained using data from the same domain it is intended to be tested on. But there may not be sufficient data available in the same domain and additionally using data from multiple sources and domains may help in creating a more generalized sentiment classifier which can be applied to multiple domains. So to create this generalized classifier, we used the sentiment data from the above corpus from different domains. We first tested the performance of sentiment analysis models built using single data source for both in-domain and cross-domain classification. Later, we built sentiment model using data samples from multiple domains and then tested the performance of the models based on their classification. Finally, we compared all the three approaches based on the performance of the models and discussed the best approach for sentiment analysis.

Keywords: Sentiment analysis, in-domain, cross-domain, machine learning, reviews, support vector machine, bernoulli naive bayes.

1. Introduction

With the rapid increase of textual content on the internet, efficient text processing is very important for various applications. With the advancement of machine learning approaches, the issue of processing can be addressed to a decent level (Hirschberg and Manning, 2015). Automated sentiment analysis is one of the important research topics. For example: Sentiment analysis is very useful in social media monitoring as it allows us to gain an overview of the wider public opinion. Most of the sentiment analysis approaches use supervised machine learning algorithms or expert-defined lexicons.

The automated sentiment analysis is a challenging task because of the natural language processing overheads like intentions of the author and the sentiment of the text can change depending on the situation. It becomes a more challenging task for non-English texts because of the unavailability of annotated data sets. In this paper, we explain how to create suitable corpora for supervised machine learning approaches i.e how to extract data from various sources and annotate them using an appropriate method.

We created a corpus "Sentiraama" for multiple domains like movie reviews, song lyrics, product reviews and book reviews in Telugu. Telugu is an agglutinative Dravidian language spoken widely in India. It is the third most popular language in India after Hindi and Bengali. According to Ethnologue[1] list of most spoken languages worldwide, Telugu ranks fifteenth in the list, and a total of 85 million Telugu native speakers exist across the world. After creating the data, we performed sentiment analysis on the data available using supervised machine learning approaches in three different ways: **In-domain, Cross-domain, Generalized** i.e classifier is trained using data from multiple domains. We, then presented the performance of the classifier models created and discussed which approach would be better to increase the accuracy or performance. To our knowledge, this is the first work in Telugu sentiment analysis at document level.

The paper is structured as follows. In section 2, we presented the related work , in section 3, we discussed the challenges in creating corpus, in section 4, we described the creation of corpus in detail, in section 5, we discussed the statistics and experiences in building classifiers and finally, we presented concluding remarks in section 6 and discussed the ideas for further improvement of automated sentiment analysis.

[1]https://www.ethnologue.com/statistics/size

2. Related Work

2.1 Sentiment Analysis in English

Sentiment analysis systems have been applied to many different kinds of texts including customer reviews (Liu, 2015; Hu and Liu, 2004; McGuinness and Ferguson, 2004), newspaper headlines (Bellegarda, 2013), blogs (Neviarouskaya et al.,2011), novels (Boucouvalas, 2002), emails (Mohammad and Yang, 2013). Often these systems have to cater to the specific needs of the text such as formality versus informality, length of utterances, etc.

2.2 Sentiment Analysis in Telugu

Sentiment analysis of Telugu social media texts has several challenges. Telugu is a morphologically complex

language. Very little work is done on sentiment analysis in Telugu. Sentiment analysis systems have been applied to different kinds of Telugu texts including song lyrics (Abburi et al., 2016), News (Mukku et al., 2016; Naidu et al., 2018).

3. Corpus Requirements and Challenges in Creating Corpus

To train a sentiment classifier for texts, an appropriate dataset is required. Here we present a method for its creation.

Firstly, we identified the scenarios as follows:

(1) We discussed the task of building a sentiment classifier to analyze the sentiment of Telugu songs from its lyrics. The classifier should support various movie songs. The sentiment classification of song lyrics is especially challenging because songs may not contain any of the subjectivity clues in a general subjectivity lexicon, yet express positive or negative emotions.

(2) We discussed the task of building sentiment classifiers to analyze reviews in their respective domains like movie, product and book reviews. The main challenges in this are the usage of colloquial language and a large number of spelling mistakes and non grammatical constructions of sentences in the reviews.

We observe that in each domain the requirements differ a lot with respect to sentiment analysis.

(1) Songs generally reflect a person's emotion at a particular situation in a movie and the lyrics of the song play a key role in carrying that emotion. Automated sentiment analysis should be able to classify the emotion of the person in that situation. The corpus is annotated at document level with two sentiment labels: positive and negative. The sentiment annotations reflect how the emotion of a song is percieved by the people.

(2)Where as in the case of reviews of objects in multiple domains like movies, books and products, the common thing in all of these is the user opinion. We decided to create a corpus with two sentiment labels: positive and negative in each domain because each review is about the user liking or disliking the object. The corpus is annotated at document level.

4. Creating the Telugu Corpus

In order to apply a machine learning approach, a corpus matching the requirements of the scenario is needed. Since there is no corpus available, we created a new corpus "Sentiraama". We identified the necessary steps for the annotation process which include:

1) Deciding the sentiment definition and formulating detailed instructions for annotation.

2) Deciding classification type and document source.

3) Annotating the documents based on the formulated instructions.

In the following paragraphs, we present our sentiment definition and describe the corpus creation procedure.

4.1 Sentiment Definition and Annotation Procedure

Two annotators annotated all the dataset items in all the domains i.e song lyrics, movie reviews, product reviews and book reviews using a 2-value scale, distinguishing between positive and negative based on the specific procedure to each domain as mentioned below. After multiple meetings and discussions with them, a kappa score of 0.9 is achieved.

4.1.1 Telugu Song Lyrics

In order to gather a dataset of unique song lyrics, we mined song lyrics from two websites viz. a2zsonglyrics24.blogspot.com and telugulyrics.org. After mining the lyrics, we cleaned them of html tags and other extraneous text. This created a dataset of 339 Telugu song lyrics. Due to lack of resources and not many lyrics in Telugu script are available, we could mine very less songs. For each song in our dataset, annotators first went through the lyrics and annotated them. But annotating only based on lyrics would be misleading as it depends on the situational context in the movie. So, annotators learned the situation of the song in the movie and corrected the wrong annotations. Finally, to make the dataset completely error free, we cross checked the annotations by keeping the count of number of positive and negative subjectivity clues that occured in the top tags of the songs.

Corpus Statistics

As shown in Figure 1, the amount of positive songs present in Telugu are higher than those of negative songs. So when we collected the data , it got reflected. We were able to get 230 positive songs and 109 negative songs whose lyrics are present in Telugu script as shown in Table 1.

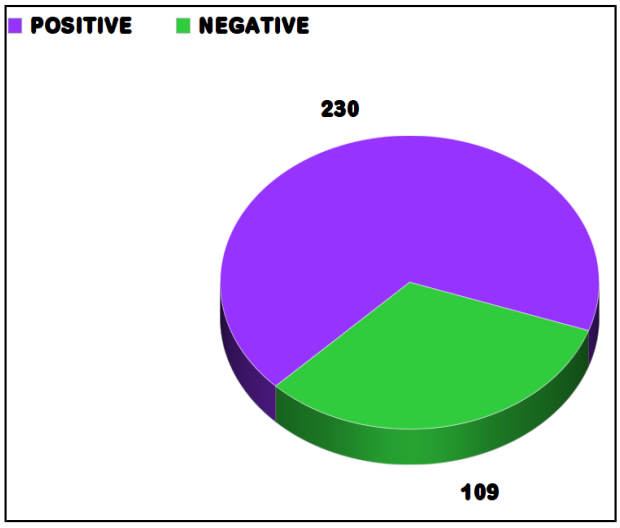


Figure 1: Pie diagram of song lyrics data.

Songs	Documents	Sentences	Words
Positive	230	4500	22500
Negative	109	2180	10900
Total	339	6680	33400

Table 1: Songs Corpus Statistics

Movie_Reviews	Documents	Sentences	Words
Positive	136	6041	60410
Negative	131	3959	39590
Total	267	10000	100000

Table 2: Movie_Reviews Corpus Statistics

4.1.2 Movie Reviews

Sentiment analyis in movie review is important because it helps in understanding how a movie is percieved by the viewer. But the very same challenge i.e the low availability of Telugu movie reviews arises even here. We tried to scrape most of the data available from different sites like tupaki.com , telugu.samayam.com and created a dataset with 267 movie reviews of more than a total of 10000 sentences. All the datasets were annotated in the following way:

The movies rated above 2.5 out of 5 by the reviewer are annotated as positive and less than 2.5 as negative. When the movie is rated as 2.5 in the review, we annotated it based on the last one line summarisation by the reviewer as it removes ambiguity. A recheck is made by going through the entire review and its annotation. Surprisingly, we found that no errors were found in the annotation made through this method.

Corpus Statistics

Here we got nearly the same amount of positive and negative reviews but the number of reviews were less. We were able to get 136 positive reviews and 131 negative reviews. The entire data consists of 10000 sentences. Though the number of reviews were less, it has decent amount of sentences. The advantage of having equal amount of data of both the types is that it prevents classification system from learning biases inherent in the dataset. Table 2 shows the corpus statistics.

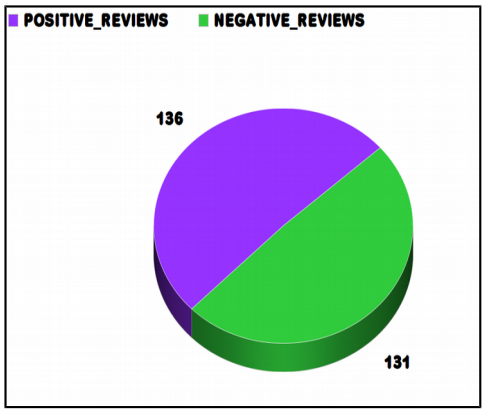


Figure 2: Pie diagram of movie reviews data.

4.1.3 Book Reviews and Product Reviews

The method of creation and procedure of annotation is the same for both of these i.e book reviews and product reviews. All the dataset items were annotated using a 2-value scale, distinguishing between positive and negative reviews. In all these cases, a review is marked as positive if a person or user who reviews is satisfied with it else it is marked as negative. But the main problem here is data gathering because we could find only 20-30 reviews written in Telugu language. A classifier model cannot be build with a such small data, so we decided to translate reviews from English to Telugu in order to get more data. The major challenge is that we should not lose the emotion conveyed by the user while translating. We tried using Google translate for the translation process but after translation the result was very bad as both the meaning and emotion conveyed were lost. So the reviews were manually translated by two experienced translators.

As shown in Figure 3, the translation was carried out in the following way: First the data was normalised and all the spelling mistakes were corrected. All the abbrevations were expanded so that the classifier could treat both abbrevation and its expansion as the same word. The numbers were retained as in Roman script. The translators were instructed to be faithful to the original text as much as possible and retain the same sentimental value. First they got the exact meaning and emotion of the review. Later they generate the review in Telugu such that it fits Telugu grammar and syntax and also carries same emotion. Though this process is accurate, it is time consuming.

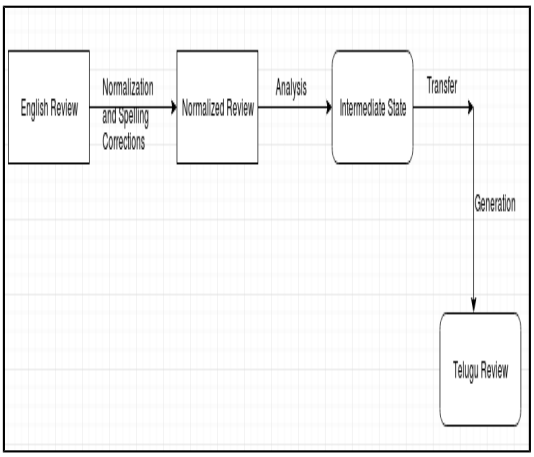


Figure 3: Translation Procedure.

In this way we were able to get 200 book reviews and 200 product reviews in each of which half of them are positive and rest of the half is negative. The annotation procedure here is going through the entire review and if the person is satisfied with the product and rated it above 2.5 out of 5 then review is annotated as positive. If the person is not satisfied with the product and rated it below 2.5 then the review is annotated as negative.

Corpus Statistics

It is necessary to equalize the dataset so that there is an equal number of positive and negative reviews. This prevents classification systems from learning biases inherent in the dataset. As shown in Figure 4, our dataset consists of 100 positive and 100 negative reviews in both the domains i.e book reviews and product reviews. Table 3 shows corpus statistics of books and Table 4 shows that of products.

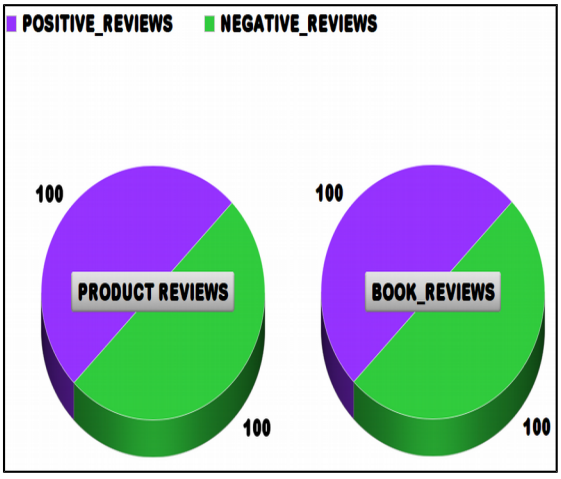


Figure 4: Pie diagrams of book reviews and product reviews.

Book_Review	Documents	Sentences	Words
Positive	100	1340	7001
Negative	100	2000	8030
Total	200	3340	15031

Table 3: Book_Reviews Corpus Statistics

Product_Review	Documents	Sentences	Words
Positive	100	2052	17390
Negative	100	2305	20104
Total	200	4357	37494

Table 4: Product_Reviews Corpus Statistics

5. Building and Analyzing Classifiers

5.1 Classifiers

We employ Bernoulli Naive Bayes as it has been found to perform well in text-related domain (Rish Irina, 2001). We also employ SVM as it gives good accuracy (Joachims T. , 1998). All of these are implemented using the scikit learn toolkit[2]. We evaluated our model applying the 10-fold cross-validation.

In Naive Bayes for text classification, the instance is assigned to the class which has the highest conditional probability of P(C|X), where C is the sentiment and X is the set of words for that instance.

In SVM, each data item is plotted as a point in n-dimensional space with the value of each feature being the value of a particular coordinate. Then, we performed classification by finding the hyper-plane that differentiate the classes very well as shown in Figure 5.

[2]http://scikit-learn.org/stable/supervised_learning.html

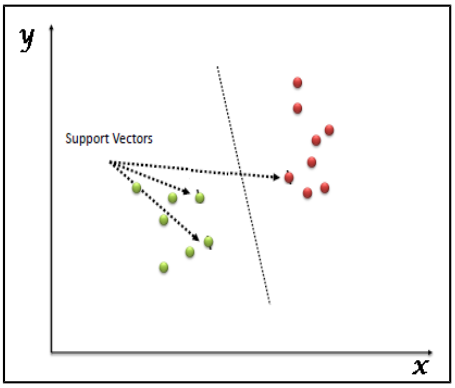


Figure 5: Support Vector Machines model.

5.2 Experimental Setup

We created four different datasets belonging to different domains. Now we use these datasets for training the classifier. We used Scikit learn[3] framework to learn a classifier based on these datasets. We first performed the in-domain sentiment analysis i.e both the training and the testing data will be from the same domain and then we evaluated the performance of the classifiers of each domain. Later we performed a cross-domain sentiment analysis i.e the training and the testing data are from different domains and evaluated the performance of the classifiers.

Finally we created a generalized classifier which is trained on data samples from all the domains and evaluated its performance.

5.3 Results

5.3.1 Single Source Data Sets(Same domain or in-domain)

Table 5 shows the results which we achieved using 10-fold cross-validation on the corpus using Bernoulli Naive Bayes classifier. Table 6 shows the results, we achieved using 10-fold cross-validation on the corpus using support vector machine.

The performance of both the classifiers was good with almost same accuracy with slight variations in precision and recall.

Thus when trained and evaluated on same domain the classifiers performed well but the problem arises when data is very sparse because the training data for classifier would be low which would affect accuracy. So we tried other approaches namely cross-domain and generalized approach.

[3]http://scikit-learn.org/stable/supervised_learning.html

Dataset	Precision	Recall	f1_score
Song Lyrics	Pos-100% Neg-75%	Pos-70% Neg-100%	84.03%
Movie Reviews	Pos-86% Neg-82%	Pos-77% Neg-89%	83.25%
Product Reviews	Pos-85% Neg-85%	Pos-85% Neg-85%	85%
Book Reviews	Pos-84% Neg-84%	Pos-84% Neg-84%	84%

Table 5: Results achieved by Bernoulli Naive Bayes classifier learned on the song lyrics, movies, products and book reviews evaluated using 10-fold cross-validation.

Dataset	Precision	Recall	f1_score
Song Lyrics	Pos-82% Neg-88%	Pos-90% Neg-78%	84.25%
Movie Reviews	Pos-88% Neg-79%	Pos-80% Neg-87%	83.3%
Product Reviews	Pos-91% Neg-80%	Pos-77% Neg-92%	84.499%
Book Reviews	Pos-88% Neg-81%	Pos-79% Neg-89%	84.034%

Table 6: Results achieved by Support Vector Machine classifier learned on the song lyrics, movies, products and book reviews evaluated using 10-fold cross-validation.

5.3.2 Cross-domain approach

In this method, we trained the classifies using datasets from one domain and tested its performance on other domains. Tables 7 shows the results, we achieved in cross-domain approach using Bernoulli Naive Bayes classifier on the corpus.

Training Dataset	Test Dataset	f1_score
Song lyrics	Movie reviews	49.5%
Song lyrics	Product reviews	53%
Song lyrics	Book reviews	50%
Movie reviews	Song lyrics	51.1%
Movie reviews	Product reviews	63.4%
Movie reviews	Book reviews	67.4%
Product reviews	Movie reviews	64.94%
Product reviews	Song lyrics	51.3%
Product reviews	Book reviews	76.8%
Book reviews	Movie reviews	61.32%
Book reviews	Product reviews	72.6%
Book reviews	Song lyrics	47%

Table 7: Results achieved by Bernoulli Naive Bayes classifier learned using cross-domain approach.

Training Dataset	Test Dataset	f1_score
Song lyrics	Movie reviews	52.2%
Song lyrics	Product reviews	60.9%
Song lyrics	Book reviews	56.7%
Movie reviews	Song lyrics	50%
Movie reviews	Product reviews	52.2%
Movie reviews	Book reviews	45.5%
Product reviews	Movie reviews	51.5%
Product reviews	Song lyrics	54.1%
Product reviews	Book reviews	50.5%
Book reviews	Movie reviews	46.6%
Book reviews	Product reviews	61.9%
Book reviews	Song lyrics	49%

Table 8: Results achieved by Support Vector Machine classifier learned using cross-domain approach.

We can clearly see in Table 7 and 8, a huge drop in f1 scores. Performing cross-domain analysis with classifiers trained performed significantly worse than in-domain classification.

5.3.3 Generalized Approach:

This section presents results for classifiers trained from a combination of datasets from all the domains. The results for using the combined approach can be found in Table 9. The dataset size refers to the number of instances found in the combined dataset. We used three training datasets namely

Case-1)Training set contains 80% of dataset from each domain.

Case-2)Training set contains 80% of dataset from all domains except the domain which is to be tested on, only 50% of that dataset is taken for training and other part will be used as test data.

Case-3)Training set contains 80% of dataset from all domains except the domain which is to be tested on, only 20% of that dataset is taken for training and other part will be used as test data.

Using the combined data set yields similar performance to using data set from same domain. But we can see a small

increase in accuracies in generalized approach.This is for case one. Where as from case-2 and case-3 we can say that even if large data from a domain is not available to train, we can generate good results using this generalized classifier. But the f1score would be little low compared to in-domain approach yet it is much better than that of cross-domain approach. So a generalized classifier here even solves the problem of sparsity to certain extent.

	Case1- 80%	Case2- 50%	Case3- 20%
Movie Reviews	Precision-86 Recall-86% f1_score-86%	Precision-84 Recall-83% f1_score-83.497%	Precision-81 Recall-81% f1_score-81%
Song lyrics	Precision-86 Recall-85% f1_score-85.497%	Precision-70 Recall-72% f1_score-70.98%	Precision-66 Recall-68% f1_score-66.985%
Product Reviews	Precision-87 Recall-87% f1_score-87%	Precision-82 Recall-81% f1_score-81.497%	Precision-79 Recall-78% f1_score-78.497%
Book Reviews	Precision-87 Recall-86% f1_score-86.497%	Precision-77 Recall-77% f1_score-77%	Precision-76 Recall-76% f1_score-76%

Table 9: Results achieved by Support Vector Machine classifier learned using Generalized approach.

5.3.4 Comparision

From the above results a commonality which we observe is that generalized classifier is the top group for evaluating various domain datasets, unlike classifiers trained from single domain, which performs better only in its domain and gives worst results for other domains. Generalized classifier is significantly better than classifier generated using cross-domain data sets and also better than in-domain classifier. Also it would gives little less but better accuracies when trained on small data than that of large data. So one classifier for all domains in a language would be a good idea than building a classifier for each domain.

6. Conclusion and Future Work

In this study, we presented a method of resource creation for sentiment analyis and a formal procedure to annotate them. We also set out to determine the performance of multi-domain sentiment analysis using data from all the domains available. We performed in-domain , cross-domain and generalized approaches and evaluated classifier's performance in each of these. We found out that using generalized (multi-source) sentiment classification would yield better results than that of in-domain and cross-domain classification.

6.1 Future Work

Future work will involve extending our methodology for sentiment classification from document level to aspect level and entity level. This would be very useful in understanding which features of products are liked by the users and which are not liked by them. This would result in the production of better products.

Using the dataset ''Sentiraama'' that we created, many machine learning algorithms and lexicon based sentiment analysis can be applied and performance can be measured and also improvements can be made.

We are also planning to work on political bias in Telugu newspapers based on this work of domain adaptation.

7. Acknowledgements

We would like to thank the annotators and translators.
This work is supported by KCIS project no: LTRC-CPH-KCIS-78

8. Bibliographical References

Julia-Hirschberg and Christopher D. Manning (July 2015) Advances in natural langage processing Science **349** (6425), 261-266. [doi: 10.1126/science.aaa8685]

Sandeep Sricharan Mukku, Nurendra Choudhary, and Radhika Mamidi. 2016. Enhanced sentiment classification of telugu text using ML techniques. In Proceedings of the 4th Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2016) colocated with 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York City, USA, July 10, 2016., pages 29–34.

Reddy Naidu, Santosh Kumar Bharti, Korra Sathya Babu, and Ramesh Kumar Mohapatra. 2018. Text summarization with automatic keyword extraction in telugu e-newspapers. In Smart Computing and Informatics, pages 555–564. Springer.

Harika Abburi, Eswar Sai Akhil Akkireddy, Suryakanth Gangashetti, and Radhika Mamidi. 2016. Multimodal sentiment analysis of telugu songs. In Proceedings of the 4th Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2016) co-located with 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York City, USA, July 10, 2016., pages 48–52.

Thorsten Joachims. (April 21 - 23, 1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In Proceeding ECML '98 Proceedings of the 10th European Conference on Machine Learning, Pages 137-142.

K. Mallareddy, Dr. (2012). Evolution of Telugu Language Teaching and Challenges to Present Curricular Trends. IOSR Journal of Humanities and Social Science. 5. 33-36. 10.9790/0837-0513336.

Heredia, Brian & Khoshgoftaar, Taghi & Prusa, Joseph & Crawford, Michael. (2016). Integrating Multiple Data Sources to Enhance Sentiment Prediction. 285-291. 10.1109/CIC.2016.046.

Lommatzsch, Andreas & Bütow, Florian & Ploch, Danuta & Albayrak, Sahin. (2017). Towards the Automatic Sentiment Analysis of German News and Forum Documents. 18-33. 10.1007/978-3-319-60447-3_2.

Rish, Irina. (2001). An Empirical Study of the Naïve Bayes Classifier. IJCAI 2001 Work Empir Methods Artif Intell. 3. .

Bing Liu. 2015. Sentiment Analysis - Mining Opinions, Sentiments, and Emotions. Cambridge University Press.

Minqing Hu and Bing Liu. 2004. Mining opinion features in customer reviews. In (McGuinness and Ferguson, 2004), pages 755–760.

Deborah L. McGuinness and George Ferguson, editors. 2004. Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence, July 25-29, 2004, San Jose, California, USA. AAAI Press / The MIT Press.

Jerome R. Bellegarda. 2013. Data-driven analysis of emotion in text using latent affective folding and embedding. Computational Intelligence, 29(3):506–526.

Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka. 2011. Affect analysis model: novel rule based approach to affect sensing from text. Natural Language Engineering, 17(1):95–135.

Anthony C Boucouvalas. 2002. Real time text-to-emotion engine for expressive internet communications. In Proceedings of International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP-2002).

Saif M. Mohammad and Tony Yang. 2013. Tracking sentiment in mail: How genders differ on emotional axes. CoRR, abs/1309.6347.

9. Language Resource References

The corpus ‘‘Sentiraama’’ was created at KCIS, LTRC, IIIT-Hyderabad. It will be available at ‘‘https://ltrc.iiit.ac.in/showfile.php?filename=downloads/sentiraama/’’.

Multilingual Multi-class Sentiment Classification Using Convolutional Neural Networks

Mohammed Attia*, **Younes Samih**†, **Ali Elkahky***, **Laura Kallmeyer**†
*Google Inc.; †University of Düsseldorf
New York, USA; Düsseldorf, Germany
{attia,alielkahky}@google.com
{samih,kallmeyer}@phil.hhu.de

Abstract

This paper describes a language-independent model for multi-class sentiment analysis using a simple neural network architecture of five layers (Embedding, Conv1D, GlobalMaxPooling and two Fully-Connected). The advantage of the proposed model is that it does not rely on language-specific features such as ontologies, dictionaries, or morphological or syntactic pre-processing. Equally important, our system does not use pre-trained word2vec embeddings which can be costly to obtain and train for some languages. In this research, we also demonstrate that oversampling can be an effective approach for correcting class imbalance in the data. We evaluate our methods on three publicly available datasets for English, German and Arabic, and the results show that our system's performance is comparable to, or even better than, the state of the art for these datasets. We make our source-code publicly available.

Keywords: sentiment analysis, sentiment detection, multi-class

1. Introduction

The social media has revolutionized the web by transforming users from being passive recipients of information into contributers and influencers. This has a direct impact on businesses, products and governance. Many of the users' posts are opinions about products and brands that impact other consumers' buying decisions and affect brand trustworthiness. Negative reviews circulated online may cause critical problems for the reputation, competitive power, and survival chances of any business. This in turn has led to some fundamental changes in how businesses approach their customers, gauge satisfaction, provide support and manage risks.

Sentiment analysis is formally defined as the task to identify and analyze subjective information of people's opinions in social media sources (Pham and Le, 2016; Yang et al., 2017). This field of study has recently attracted a lot of attention due to its implications for businesses and governments. The challenges of this task can be summarized in the following points:

- There is a large variety of expressions to denote a range of sentiments. Therefore a sentiment dictionary, no matter how large it could be, cannot list all the possible ways people can express their attitudes.
- Words change meaning depending on the context and the domain. For example, a "short cord" mostly indicates a negative opinion, while a "short boot time" signifies a positive one.
- There could be long-distance dependencies between different constituents of a sentence, and without knowing, for example, the scope of negation, the polarity of an adjective cannot be determined.
- People do not always reveal their opinion in an explicit way, as they could be indirect, subtle or ironical.

Users generally express a broad variety of sentiments with a wide range of degrees, but to simplify the task, sentiment analysis approaches have traditionally classified sentiments into either positive, negative or neutral.

This paper describes our system for multilingual, multi-class sentiment classification using Convolutional Neural Networks (CNNs). We evaluate our system on three datasets in three different languages, and we find that state-of-art results can be achieved without language-specific features or pre-trained word embeddings. We also find that data imbalance has a detrimental effect on multi-class classification. We use over-sampling to balance datasets by repeating instances to increase the size of minority classes, which generally led to significant improvements.

Our methods are language-independent in the sense that we do no rely on ontologies, lists of polarity lexical terms, morphological or syntactic information, thus avoiding the need to deal with out-of-vocabulary opinion words and language-specific features. Our source-code publicly available at `https://github.com/SamihYounes/senti-cnn`.

2. Related Work

The different approaches to sentiment analysis share the common theme of mapping a piece of text to a given label from a predefined set (Pang and Lee, 2008). This in essence is similar to some other NLP tasks, such as text categorization (a.k.a. document classification) and language identification. There exists, however, some research on the use of unsupervised methods (Lin and He, 2009).

Research on sentiment analysis is broadly categorized into two paradigms (Cambria, 2016), knowledge-based and statistics-based, depending on whether external language-dependent information and structured resources (such as POS taggers ans polarity lexicons) are included as features in the model or not.

In the first paradigm, machine learning methods are applied to sentiment classification assisted with knowledge-

based features (Mullen and Collier, 2004; Boiy and Moens, 2009; Godbole et al., 2007; Gamon, 2004). Within this domain, work could include a massive amount of prior linguistic knowledge and feature engineering. For example, Wilson et al. (2005) compiled a prior-polarity lexicon of 8,000 subjectivity clues (with around 33% positive, 60% negative and 7% neutral clues) and used 27 linguistic features, including the existence of prior-polarity clues, POS tags, context, use of intensifiers and pronouns, and document topic. This method, however, is unscalable, resource-intensive, and will be hard to adapt to different domains or less-resourced languages.

Some dedicated resources were also created for aiding with this task including the SentiWordNet (Esuli and Sebastiani, 2006; Baccianella et al., 2010) which annotates WordNet synsets according to their degrees of positivity, negativity, and neutrality, and SenticNet (Cambria et al., 2016), which provides the semantic, cognitive and affective information for over 14,000 concepts. Similar lexicons of varying scales were built for some other languages, including, for instance, German (Remus et al., 2010), Arabic (Badaro et al., 2014), and Spanish (Perez-Rosas et al., 2012).

In the second paradigm, using pure statistical methods for sentiment analysis was also successfully applied (Neethu and Rajasree, 2013; Maas et al., 2011; Tripathy et al., 2016). A pioneering work within this approach is the research of Pang et al. (2002) who employed three machine learning methods (Naive Bayes, maximum entropy classification, and support vector machines) to detect polarity in movie reviews. They relied on corpus-driven features using the bag-of-features framework which assumes $f_1, ..., f_m$ as a predefined set of m features (word unigrams and bigrams), $n_i(d)$ as the number of times f_i occurred in a document d, and then each document d is represented by the vector: $\vec{d} := (n_1(d), ..., n_m(d))$. This method is a precursor to the relatively recent word2vec word representation (Mikolov et al., 2013b; Mikolov et al., 2013a; Pennington et al., 2014).

Deep learning for sentiment analysis has also been presented in a number of papers such as (Glorot et al., 2011; Poria et al., 2016; dos Santos and Gatti, 2014). The basic idea with deep learning is to use hidden layers of neural nets to automatically capture the underlying factors that lead from the input to the output, eliminating the need for feature engineering.

3. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) (LeCun et al., 1995) are a powerful deep learning technique because they preserve the spatial structure of the data. They have been shown to produce state-of-the-art results in image processing, computer vision (Krizhevsky et al., 2012) and speech recognition (Graves et al., 2013). In recent years, CNNs have been successfully applied to NLP and document classification problems (Kim, 2014; Johnson and Zhang, 2014).

The input to CNNs is a feature map which corresponds to the pixels in an image or words in a sentence or document, or characters in words. This feature map is scanned in CNNs one area at a time by filters, assuming that filters slide, or convolve, around the feature map. The way CNNs adjust their filter weights is through backpropagation, which means that after the forward pass, the network is able to look at the loss function and make a backward pass to update the weights.

The CNN layer is followed by a pooling layer that compresses or generalizes over the CNN representations. It reduces the dimensionality of the CNN layer by downsampling the output and taking the maximum value as the feature corresponding to each filter.

The pooling layer is typically followed by a feed-forward fully connected layer that takes the features from the pooling layer and makes new combinations for further learning or final predictions.

4. Data Description

4.1. English - The Sanders Twitter Sentiment Corpus

The Sanders Twitter Sentiment dataset[1] (Sanders, 2011) consists of 5,513 tweets related to the products of four companies: Apple, Google, Microsoft, and Twitter. Tweets have been manually tagged as either positive, negative, neutral, or irrelevant with respect to the topic. The distribution of the tagset is in Table 1. We could not find information on how this dataset was annotated, e.g. annotation guidelines, number of annotators involved and whether it was annotated in-house or through crowd-sourcing.

positive	negative	neutral	irrelevant	Total
570	654	2,503	1,786	5,513

Table 1: The Sanders Twitter Sentiment dataset

4.2. German - Deutsche Bahn

The data of the GermEval shared task[2] (Wojatzki et al., 2017) consists of 21,824 messages from various social media and web sources intended for analyzing customer reviews about "Deutsche Bahn", the German public train operator with about two billion passengers annually. The data is split roughly into 90% for training and 10% for development as shown in Table 2. They also provide two test sets. The shared task's Subtask-B is on multi-class document-level polarity, which is about identifying whether the customer's opinion of "Deutsche Bahn" or travel is positive, negative or neutral. The sentence length in the training set ranges between 182 words and just two words. Test set 1 contains larger articles reaching up to 4,666 words in length.

Dataset	positive	negative	neutral
training	1,179	5,048	13,222
development	149	589	1,637
test 1	105	780	1,681
test 2	108	497	1,237

Table 2: Deutsche Bahn Dataset

The data was annotated by a team of six annotators (Wojatzki et al., 2017), and each document was annotated by

[1] http://www.sananalytics.com/lab

[2] https://sites.google.com/view/germeval2017-absa/

positive	negative	mixed	objective	Total
799	1,684	832	6,691	10,006

Table 3: ASTD Dataset

two annotators using WebAnno's curation interface. The documents were checked for consistency by a supervisor who decided on divergences and new issues.

4.3. Arabic - ASTD

The Arabic Sentiment Tweets Dataset (ASTD)[3] (Nabil et al., 2013) consists of 10,006 tweets which are classified as 'positive', 'negative', 'mixed', and 'objective'. The distribution of the tagset is shown in Table 3. The tweets were collected from EgyptTrends and were not related to any particular topic, but generally included comments on diverse political issues. The tweets were annotated through the Amazon Mechanical Turk by three annotators. Tweets that were assigned the same rating by at least two annotators were accepted, otherwise rejected.

A key observation in the three datasets is the noticeable imbalance between the labels as shown in Figure 1, where the "positive" label shows the most disproportionate distribution.

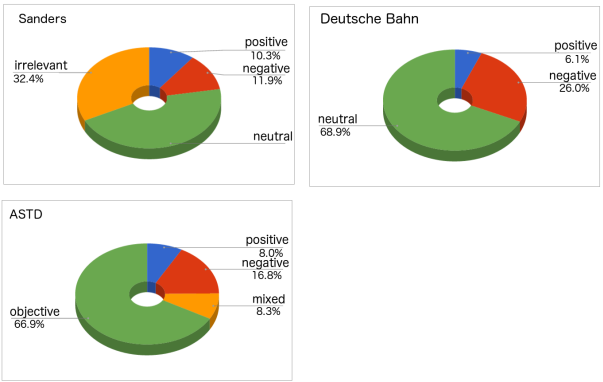


Figure 1: Tagset Distribution

5. System Description

We use a deep neural network model for predicting sentiment polarity. The architecture of our model, shown in Figure2, is straight-forward. The first layer in our model is a randomly-initialized word embedding layer that turns words in sentences into a feature map and preserves the spatial (contextual) information for each word. This is followed by a convolution neural network (CNN) layer that scans the feature map. This CNN layer has 300 filters and a width of 7, which means that each filter is trained to detect a certain pattern in a 7-gram window of words. Global max-pooling is applied to the output of each filter to take the maximum score of each pattern we search for though the text. The main function of the pooling layer is reduce the dimensionality the CNN representations by down-sampling the output and taking the maximum value as the feature corresponding to each filter. Those score are then supplied to a

[3]http://www.mohamedaly.info/datasets/astd

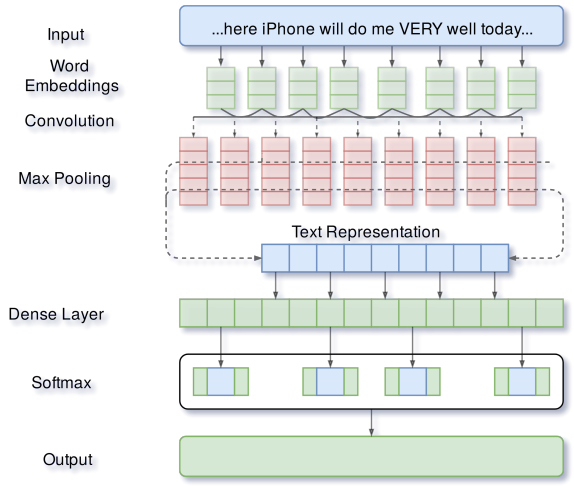


Figure 2: The Architecture of our sentiment detection model applied to an example tweet, best viewed in color. Here the model takes a tweet as its current input to predict its sentiment polarity.

single feed-forward (fully-connected) layer of size 600 and Relu activation to make further learning. Finally, the output of that layer goes through a Softmax layer that predicts the output classes. Table 4 and Figure 2 show the layer configuration and architecture of our model.

Layer	Output Shape	Params
embedding	(None, 300, 300)	10960800
conv1d	(None, 294, 300)	630300
glob_max_pool_1d	(None, 300)	0
dense_1	(None, 600)	180600
dense_2	(None, 3)	2404

Table 4: Neural Network Architecture

For processing the data, we perform manual tokenization on the data by inserting a space between words and punctuation marks. Then we ignore the long tail of the low-frequency words (heuristically setting the threshold at 3, i.e. ignoring words that occur three times or less), and remove URL addresses.We tried ignoring the top three most frequent words, assuming that they are semantically-irrelevant function words (e.g. punctuations, determiners and prepositions), but this led to lowering the performance.

6. Experiments and Results

We compare our system to the best scores published in the literature for each dataset. We consider the comparison meaningful only when there is a test set provided. Otherwise, we create our own splits of the data, and the comparison with previously published results are only indicative.

6.1. Results on Sanders Dataset

Sentiment analysis for English using the Sanders dataset has been reported in a number of papers. For example, Bravo-Marquez et al. (2013) extracted features from a number of lexical resources (such as OpinionFinder, AFINN and SentiWordNet) and applied a number of traditional ML classifiers (such as J48, Naive Bayes, Logistic

and SVM), and reported an accuracy of 70.1%. Da Silva et al. (2014) used the Ensemble method (training multiple learners to solve the same problem) and reported an accuracy score of 76.25%. Similarly Hassan et al. (2013) used a stack of learning models and a voting mechanism and obtained an accuracy of 76.30%. Using our system we achieved a significantly higher accuracy score of 78.3%. However, as there were no dedicated test set for this dataset we tested our system on a randomly selected subset of 20% of the data, which makes the comparison with the other systems not conclusive.

It is to be noted that Johansson and Lilja (2016) reported a higher score for the Sanders dataset of 84.38% using a lexicon, but they reported that the method did not scale well to other datasets including IMDB and Sentiment140. It is also not clear to us whether their experiments were limited to the binary polarity of positive and negative or they predicted the full range of classes.

6.2. Results on Deutsche Bahn

The Deutsche Bahn dataset has been used in the GermEval-2017 Shared Task on Aspect-based Sentiment in Social Media Customer Feedback. The best results on Test 1 were reported by Naderalvojoud et al. (2017), Hövelmann and Friedrich (2017) and Sidarenka (2017) who achieved accuracy scores of 74.94%, 74.786% and 74.47% respectively. Hövelmann and Friedrich (2017) used three German lexicons (combined in what they called the German SentiWordNet Lexicon) and fed them to a deep recurrent neural network (RNN) classifier to discover context-based sentiments. Hövelmann and Friedrich (2017) on their turn used a fastText classifier (the Facebook open-source library for text representation), enhanced with pretrained vectors and gradient boosted trees (GBTs) trained on bag-of-words (BOWs). Sidarenka (2017) used a hybrid approach by joining an SVM module trained on user-specified attributes with a bi-directional LSTM network.

We fine tune the hyper parameters of our system on the development set of the GermEval shared task. Our system gives an accuracy of 75.45% on Test 1 which is slightly above the results of the best system. We consider the comparison here meaningful as we are testing against the same benchmark set.

6.3. Results on the ASTD Dataset

The ASTD dataset has been used in (Nabil et al., 2013) and the best reported accuracy score is 69.1% using an SVM classifier. They tried to balance the data using undersampling which did not perform as well. Our system gives an accuracy of 67.93%, which is slightly below their results, but the evaluation is not conducted on the same set. As no test set was provided we randomly selected a 20% subset from the available data.

7. Dealing with Data Imbalance

We notice that the prediction results favor the majority classes at the expense of the minority classes as the system attempts to achieve high accuracy scores. Trying to resolve this issue, we need first to find an evaluation metrics that gives equal weight to the classes in the labelset, and second find a way to balance the data. For the evaluation measure, we choose the F1 score with a macro average as it calculates the f-score for each label, and outputs their unweighted mean, allowing each class to have the same weight as the other classes regardless of the number of instances.

To balance out the training data, we apply manual oversampling by repeating records of the minority classes. It is to be noted that SMOTE (Synthetic Minority Over-Sampling Technique) is sometime successfully applied to numerical data, but it is not applicable to textual data as is case with the data here. We compare the results using the macro F1 before and after oversampling for the three datasets.

For the English dataset, Tables 5 and 6 show the confusion matrix for the model's predictions without and with oversampling respectively. In oversampling we repeated the records for 'Positive' and 'Negative' three-folds. The oversampling technique works in improving the performance raising both the accuracy (from 78.60% to 79.57%) and the macro F-measure scores (from 69.13% to 70.23%).

	Negative	**Positive**	**Neutral**	**irrelevant**
Negative	56	12	39	0
Positive	10	50	36	5
Neutral	29	37	384	23
irrelevant	3	1	24	314
Accuracy	78.60			
F1 Macro	69.13			

Table 5: Confusion Matrix for the Sanders Dataset without Oversampling

	Negative	**Positive**	**Neutral**	**irrelevant**
Negative	55	4	46	2
Positive	11	47	39	4
Neutral	25	21	404	23
irrelevant	3	1	30	308
Accuracy	79.57			
F1 Macro	70.23			

Table 6: Confusion Matrix for the Sanders Dataset with Oversampling

For the German dataset, Table 7 show the confusion matrix for the model predictions for the test set without oversampling. Table 8 shows the results after oversampling 'Positive' and 'Negative' three folds. The oversampled model shows a better confusion matrix with an increase in the macro F1 score (from 49.00% to 45.84%) despite the decrease in Accuracy.

	Negative	**Positive**	**Neutral**
Negative	344	1	435
Positive	10	4	91
Neutral	89	4	1588
Accuracy	75.45		
F1 Macro	49.00		

Table 7: Confusion Matrix for the Deutsche Bahn Dataset without Oversampling

	Negative	Positive	Neutral
Negative	482	5	293
Positive	16	16	73
Neutral	259	28	1394
Accuracy	73.73		
F1 Macro	54.84		

Table 8: Confusion Matrix for the Deutsche Bahn Dataset with Oversampling

For the Arabic dataset, Table 9 show the confusion matrix without oversampling, and Table 10 shows the results after oversampling by repeating 'Positive' and 'Mixed' three folds, and double sizing the 'Negative'. The oversampled model again shows a better macro F1 score with an increase from 29.82% to 35.32%.

	Negative	Positive	Objective	Mixed
Negative	111	0	233	0
Positive	6	0	167	0
Objective	73	0	1249	0
Mixed	43	0	120	0
Accuracy	67.93			
F1 Macro	29.82			

Table 9: Confusion Matrix for the ASTD Dataset without Oversampling

	Negative	Positive	Objective	Mixed
Negative	89	2	237	16
Positive	10	16	145	2
Objective	30	73	1191	28
Mixed	24	4	118	17
Accuracy	65.58			
F1 Macro	35.32			

Table 10: Confusion Matrix for the ASTD Dataset with Oversampling

Oversampling the data, simply by replicating records, only tricks the model into giving greater weight to a certain class, but does not provide any essentially useful information to the system to base it's judgment on. Therefore, we recommend data annotation to selectively target more data to increase the size of the minority classes to allows the system to better understand and predict these classes.

8. Conclusion

In this paper we have presented our systems for multi-class sentiment classification and we found that the deep neural network model can outperform traditional methods that rely on language-specific feature engineering. We show that the class imbalance in the data can lead to degradation in the system performance, and point out that oversampling can be a helpful workaround for handling this imbalance.

Further we note that the system performs better on the Sanders dataset, followed by the Deutsche Bahn and the ASTD datasets. This could be connected to the observation that the ASTD dataset has the largest number of classes (5 classes compared to 4 in Sanders and 3 in Deutsche Bahn) and there could be an inverse correlation between the increased number of classes and the system performance.

Beside the class balance and the granularity of classes, we assume that the system performance could be also impacted by the quality of the annotation (in-house vs. crowdsourced) and whether the polarity is related to a specific or general topic.

9. Bibliographical References

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 2200–2204.

Badaro, G., Baly, R., Hajj, H., Habash, N., and El-Hajj, W. (2014). A large scale arabic sentiment lexicon for arabic opinion mining. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 165–173.

Boiy, E. and Moens, M.-F. (2009). A machine learning approach to sentiment analysis in multilingual web texts. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 12:526–558.

Bravo-Marquez, F., Mendoza, M., and Poblete, B. (2013). Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, Chicago, USA.

Cambria, E., Poria, S., Bajpai, R., and Schuller, B. (2016). Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives. *In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2666–2677.

Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems*, 31:102–107.

Da Silva, N. F., Hruschka, E. R., and Hruschka, E. R. (2014). Tweet sentiment analysis with classifier ensembles. *Decision Support Systems*, 66:170–179.

dos Santos, C. N. and Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. *In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78.

Esuli, A. and Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06)*, pages 417–422.

Gamon, M. (2004). Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. *In Proceedings of the 20th International Conference on Computational Linguistics (COLING'04)*.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Domain adaptation for large-scale sentiment classification: A deep learning approach. *In Proceedings of the Twenty-eight International Conference on Machine Learning, ICML*.

Godbole, N., Srinivasaiah, M., and Skiena, S. (2007). Large-scale sentiment analysis for news and blogs. *In*

ICWSM 2007 - International Conference on Weblogs and Social Media.

Graves, A., Mohamed, A.-r., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE.

Hassan, A., Abbasi, A., and Zeng, D. (2013). Twitter sentiment analysis: A bootstrap ensemble framework. In *Social Computing (SocialCom), 2013 International Conference on*, pages 357–364. IEEE.

Hövelmann, L. and Friedrich, C. M. (2017). Fasttext and gradient boosted trees at germeval-2017 tasks on relevance classification and document-level polarity. In *Proceedings of the GermEval 2017 Shared Task*, Berlin, Germany.

Johansson, H. and Lilja, A. (2016). Method performance difference of sentiment analysis on social media databases: Sentiment classification in social media.

Johnson, R. and Zhang, T. (2014). Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995.

Lin, C. and He, Y. (2009). Joint sentiment/topic model for sentiment analysis. *In Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 375–384.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., , and Potts, C. (2011). Learning word vectors for sentiment analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 142–15.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *In Proceedings of International Conference on Learning Representations (ICLR) 2013. arXiv:1301.3781v3*, pages 746–751, Scottsdale, AZ.

Mikolov, T., tau Yih, W., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT 2013*, pages 746–751, Atlanta, Georgia.

Mullen, T. and Collier, N. (2004). Sentiment analysis using support vector machines with diverse information sources. *Proceedings of EMNLP-04, 9th Conference on Empirical Methods in Natural Language Processing*.

Nabil, M., Aly, M., and Atiya, A. (2013). Astd: Arabic sentiment tweets dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 2515–2519, Lisbon, Portugal.

Naderalvojoud, B., Qasemizadeh, B., and Kallmeyer, L. (2017). Hu-hhu at germeval-2017 sub-task b: Lexicon-based deep learning for contextual sentiment analysis. In *Proceedings of the GermEval 2017 Shared Task*, Berlin, Germany.

Neethu, M. S. and Rajasree, R. (2013). Sentiment analysis in twitter using machine learning techniques. pages 1–5, July.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2:1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. *EMNLP 2002*, 10:79–86.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar.

Perez-Rosas, V., Banea, C., and Mihalcea, R. (2012). Learning sentiment lexicons in spanish. In *LREC*, volume 12, page 73.

Pham, D.-H. and Le, A.-C. (2016). A neural network based model for determining overall aspect weights in opinion mining and sentiment analysis. *Indian Journal of Science and Technology*, 9:1–6.

Poria, S., Chaturvedi, I., Cambria, E., and Hussain, A. (2016). Convolutional mkl based multimodal emotion recognition and sentiment analysis. *In Proceedings of the Twenty-eight International Conference on Machine Learning, ICML*, pages 439–448.

Remus, R., Quasthoff, U., and Heyer, G. (2010). Sentiws-a publicly available german-language resource for sentiment analysis. In *LREC*.

Sanders, N. J. (2011). Sanders-twitter sentiment corpus. *Sanders Analytics LLC*.

Sidarenka, U. (2017). Potts at germeval-2017 task b: Document-level polarity detection using hand-crafted svm and deep bidirectional lstm network. In *Proceedings of the GermEval 2017 Shared Task*, Berlin, Germany.

Tripathy, A., Agrawal, A., and Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *In Expert Systems With Applications*, 57:117–126.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. pages 347–354.

Wojatzki, M., Ruppert, E., Holschneider, S., Zesch, T., and Biemann, C. (2017). Germeval 2017: Shared task on aspect-based sentiment in social media customer feedback. In *Proceedings of the GermEval 2017 Shared Task*, Berlin, Germany.

Yang, K., Cai, Y., Huang, D., Li, J., Zhou, Z., and Lei, X. (2017). An effective hybrid model for opinion mining and sentiment analysis. *IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 465–466.

A Large Self-Annotated Corpus for Sarcasm

Mikhail Khodak, Nikunj Saunshi, Kiran Vodrahalli
Computer Science Department, Princeton University
35 Olden St., Princeton, New Jersey 08540
{mkhodak,nsaunshi,knv}@cs.princeton.edu

Abstract

We introduce the Self-Annotated Reddit Corpus (**SARC**), a large corpus for sarcasm research and for training and evaluating systems for sarcasm detection. The corpus has 1.3 million sarcastic statements — 10 times more than any previous dataset — and many times more instances of non-sarcastic statements, allowing for learning in both balanced and unbalanced label regimes. Each statement is furthermore *self-annotated* — sarcasm is labeled by the author, not an independent annotator — and provided with user, topic, and conversation context. We evaluate the corpus for accuracy, construct benchmarks for sarcasm detection, and evaluate baseline methods.

Keywords: sarcasm, classification, conversation

1 Introduction

Sarcasm detection is an important component in many natural language processing (NLP) systems, directly relevant to natural language understanding, dialogue systems, and text mining. However, detecting sarcasm is difficult because it occurs infrequently and is difficult for even humans to discern (Wallace et al., 2014). Despite these properties, existing datasets either have *balanced labels* — data with roughly the same number of examples of each label (González-Ibánez et al., 2011; Bamman and Smith, 2015; Joshi et al., 2015; Amir et al., 2016; Oraby et al., 2016) — or use humans to annotate sarcastic statements (Riloff et al., 2013; Swanson et al., 2014; Wallace et al., 2015).

In this work, we make available the first corpus[1] for sarcasm detection that has both unbalanced and self-annotated labels and does not consist of short text snippets from Twitter[2]. With more than a million examples of sarcastic statements, each provided with author, topic, and contex information, the dataset exceeds all previous sarcasm corpora by an order of magnitude in size. This is possible due to the comment structure of the social media site Reddit[3] as well as its frequently-used and standardized annotation for sarcasm.

Following a discussion of corpus construction and relevant statistics in Section 3, we discuss the quality of this dataset compared to alternative sources in Section 4, manually evaluating our corpus for noise. Then in Section 5 we use our dataset to construct suitable benchmarks for sarcasm detection systems and examine the performance of simple baseline methods and human evaluators on these subsets.

2 Related Work

Since our main contribution is a corpus and not a method for sarcasm detection, we point the reader to a recent survey by Joshi et al. (2016) that discusses many interesting efforts in this area. Note that many of the works the authors mention will be discussed by us in this section, with many papers using their own datasets and illustrating the need for common evaluation baselines.

Sarcasm datasets can largely be distinguished by the sources used to get sarcastic and non-sarcastic statements, the amount of human annotation, and whether the dataset is balanced or unbalanced. Reddit has been used before, notably by Wallace et al. (2015); while the authors allow unabalanced labeling, they do not exploit the possibility of using self-annotation and generate around 10,000 human-labeled sentences. Twitter is a frequent source due to the self-annotation provided by hashtags such as #sarcasm, #notsarcasm, and #irony (Reyes et al., 2013; Bamman and Smith, 2015; Joshi et al., 2015). As discussed in Section 4.2, its abbreviated language and other properties make Twitter a less attractive source for annotated comments. However, it is by far the largest raw source of data for this purpose and has led to some large unbalanced corpora in previous efforts (Riloff et al., 2013; Ptáček et al., 2014). A further source of comments is the Internet Argument Corpus (IAC) (Walker et al., 2012), a scraped corpus of Internet discussions that can be further annotated for sarcasm by humans or by machine learning; this is done by Lukin and Walker (2013) and Oraby et al. (2016), in both cases resulting in around 10,000 labeled statements.

3 Corpus Details

3.1 Reddit Structure and Annotation

Reddit is a social media site in which users communicate by commenting on *submissions*, which are titled posts consisting of embedded media, external links, and/or text, that are posted on topic-specific forums known as *subreddits*; examples of subreddits include `funny`, `pics`, and `science`. Users comment on submissions and on other comments, resulting in tree-like conversation structure such that each comment has a parent comment. We refer to *elements* as any nodes in the tree of a Reddit link (i.e., comments or submissions).

Reddit users have adopted a common method for sarcasm annotation consisting of adding the marker "/s" to the end of sarcastic statements; this originates from the HTML text delineation `<sarcasm>...</sarcasm>`. As with Twitter hashtags, using these markers as indicators of sarcasm is noisy (Bamman and Smith, 2015), especially since

[1] http://nlp.cs.princeton.edu/SARC/
[2] https://www.twitter.com
[3] https://www.reddit.com



Figure 1: A Reddit submission and one of its comments. Note the conventional annotation "/s" indicating sarcasm.

many users do not use the marker, do not know about it, or only use it where sarcastic intent is not otherwise obvious. We discuss the extent of this noise in Section 4.1.

3.2 Constructing SARC

Reddit comments from December 2005 have been made available due to web-scraping [4]; we construct our dataset as a subset of comments from January 2009-April 2017, comprising the vast majority of comments and excluding noisy data from earlier years. For each comment we provide a sarcasm label, author, the subreddit it appeared in, the comment score as voted on by users, the date of the comment, and identifiers linking back to the original dataset of all comments.

To reduce noise, we use several filters to remove noisy and uninformative comments. Many of these are standard preprocessing steps such as excluding URLs and limiting characters to be ASCII. To handle Reddit data, we also exclude comments that are descendants of sarcastic comments in the conversation tree, as annotation in such cases is extremely noisy, with authors agreeing or disagreeing with the previously expressed sarcasm with their own sarcasm but often with no marking.

Our raw corpus consists of three files:

1. An array in CSV format containing 533 million comments, of which around 1.3 million are sarcastic. This file only contains those comments whose authors know about the standard sarcasm annotation; this is determined by whether they have used the annotation in the same month as the comment was made or earlier. This limitation is added in order to reduce false negatives due to authors not annotating their sarcasm. Each row also contains the parent comment.

2. A hashtable in JSON format containing all comments and posts in the conversation thread of a sarcastic comment as well as all siblings of sarcastic comments.

3. An array in CSV format, with each row containing a sequence of comments leading up to a sarcastic comment, the (sarcastic and non-sarcastic) responses to the last element in that sequence, and the labels of those responses. Each element is given as a key to the previous file.

This raw corpus is very large and suitable for both large-scale machine learning and statistical analysis as well for deriving smaller benchmark tasks for evaluating sarcasm detection systems. These benchmarks, whether in the balanced or unbalanced regimes, require further subsampling of the corpus and an approach for dealing with noisy data in the face of sparse signals. We specify and evaluate an approach for doing so in Section 5.1, followed by the evaluation of learning algorithms on the output.

[4] http://files.pushshift.io/reddit

Corpus	Dataset	Sarcastic	Total
IAC	Joshi et al. '15	751	1502
	Oraby et al. '16	4.7K	9.4K
Twitter	Joshi et al. '16	4.2K	5.2K
	Bamman & Smith '15	9.7K	19.5K
	Reyes et al. '13	10K	40K
	Riloff et al. '13	35K	175K
	Ptáček et al. '13	130K	780K
Reddit	Wallace et al. '15	753	14124
	SARC	**1.34M**	**533M**

Table 1: SARC compared with previous sarcasm corpora. In addition to a million sarcastic comments our dataset also provides many millions more non-sarcastic statements by the same authors.

4 Corpus Evaluation

There are three major metrics of interest for evaluating our corpora: (1) size, (2) the proportion of sarcastic to non-sarcastic comments, and (3) the rate of false positives and false negatives. Of interest is also the quality of the text in the corpus and its applicability to other NLP tasks. Thus in this section we evaluate error in the raw corpus and provide comparison with other corpora used to construct sarcasm datasets. We also discuss the potential limitations of our approach.

4.1 Manual Evaluation

To investigate the noisiness of using Reddit as a source of self-annotated sarcasm we estimate the proportion of false positives and false negatives induced by our filtering. This is done by manually checking a random subset of 500 comments from SARC tagged as sarcastic and 500 tagged as non-sarcastic, with full access to the comment's context. A comment was determined to be false positive if "/s" tag was not an annotation but part of the sentence and a false negative if the comment author was clearly being sarcastic to the human rater. This procedure yielded a false positive rate of 1.0% and a false negative rate of 2.0%. Although the false positive rate is reasonable, the false negative rate is significant compared to the sarcasm proportion (0.25%), indicating large variation in the working definition of sarcasm and the need for methods that can handle noisy data in the unbalanced setting. In the balanced setting this is still a fairly small amount of noise.

4.2 Comparison with other Sources

As noted before, Twitter has been the most common source for sarcasm in previous corpora; this is likely due to the explicit annotation provided by its hashtags. However, using Reddit as a source of sarcastic comments holds many research advantages. Unlike Reddit comments, which are not

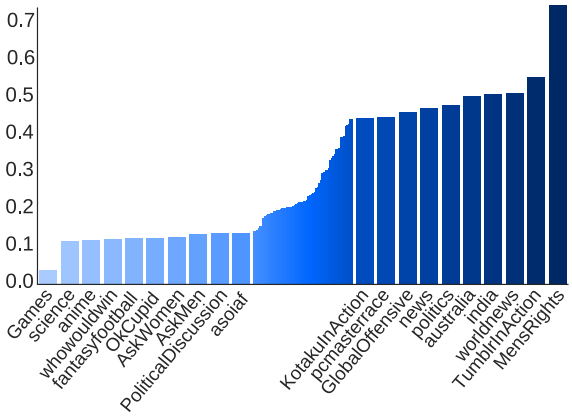


Figure 2: Sarcasm percentage for subreddits with more than a million comments in SARC. Well-moderated and special-interest forums such as `science` and `asoiaf` (referring to fantasy series *A Song of Ice and Fire*) have less sarcasm than controversial and less-moderated subreddits.

constrained by length and contain fewer hashtags, tweets are written in abbreviated English. Hashtagged tokens are also frequently used as a part of the statement itself (e.g. "that was #sarcasm"), blurring the line between text and annotation; on Reddit "/s" is generally only used as something other than annotation when its use as an annotation is being referred to (e.g. "you forgot the /s"). The full conversation context is also much easier to provide on Reddit due to the shallow tree structure of an individual post and its comments.

Furthermore, from a subsample of Twitter and Reddit data from July 2014 we determined that a vastly smaller percentage (.002% vs. .927%) of Twitter authors make use of sarcasm annotation (#sarcasm, #sarcastic, or #sarcastictweet). We hypothesize that Reddit users require sarcastic annotation more frequently and in a more standardized form because they are largely anonymous and so cannot rely on a shared context to communicate sarcasm. Finally, Reddit also benefits from having subreddits, which enable featurization and data exploration based on an explicit topic assignment.

The Internet Argument Corpus (IAC) has also been used as a source of sarcastic comments (Walker et al., 2012). The corpus developers found 12% of examples in the IAC to be sarcastic, which is a much nicer class proportion for sarcasm detection than ours. As the Reddit data consists of arbitrary conversations, not just arguments, it is not surprising that our sarcasm percentage is much smaller, even when accounting for false negatives; this property also makes our dataset more realistic. Unlike Reddit and Twitter, the IAC also requires manual annotation of sarcasm.

4.3 Limitations of Our Approach

There are a few noteworthy limitations to our method of collecting a self-annotated sarcasm dataset. Despite our efforts to filter noisy "/s" labels, there remain instances where no simple rule reliably eliminates incorrect labels. We describe the difficulties for both false positives and false negatives:

- **False positives** are instances where a comment is incorrectly labeled as sarcastic due to the presence of a "/s" tag. This case only occurs when a "/s" tag appears in the comment with a meaning different from indicating sarcasm. As previously noted, this possibility is more likely to occur if a user is unaware of the "/s" notation. Similarly, if "/s" is used to *refer to the convention of its use as an annotation*, the naive approach of merely detecting the "/s" string also fails. Finally, it is possible that "/s" has other connotations: For instance, in HTML, `<s>...</s>` denotes a strike-through. Therefore a subreddit focusing on the discussion of web programming, for example, might include instances where "/s" is used with a different meaning.

  To combat the first issue, we restrict to users aware of the sarcasm notation by ensuring they have used "/s" previously. This filter helps ensure that the users are aware of the semantic usage of "/s". For the second case, we only keep comments which have the "/s" at the *end* of the comment. All comments we inspected which terminated in "/s" used the annotation to indicate sarcasm. The third case amounts to solving word sense disambiguation, and we did not find a universally simple approach to reduce noise of this form. However, it is possible to reduce the likelihood of this form of sense mismatch by restricting to subreddits which are known to not have alternate senses for "/s" (e.g., politics).

- **False negatives** are instances where a comment is sarcastic, but not annotated with a "/s". False negatives are harder to detect than false positives since the portion of comments which have no sarcasm annotation is much larger than the portion that do. There are two primary ways a false negative can arise: Either a user does not know of the "/s" convention, or the user believes their use of sarcasm is obvious enough to warrant not including the tag. Notably, such a belief depends on what community the user is communicating in, who the user is communicating with (another user they routinely have arguments with, or a stranger), and also on prior comments on the thread. As noted previously, a comment which is sarcastic often spawns a chain of subsequent comments which are all sarcastic, but which lack the "/s" symbol. In short, context matters a lot for determining whether or not a sarcastic comment is obvious.

  The first issue is solved by our first filter. The second issue is difficult to address, and remains a limitation of our approach. We avoid the particular case of sarcastic comment chains by discarding the child comments of sarcastic comments in a thread.

All of our filters are validated by manual evaluation of the false positive and false negative rates as described in Section 4.1, which improved considerably after implementing the filters. Our manual evaluation approach has one central limitation: Though we provide local context to the human annotators, if the ability to distinguish the sarcastic intent of a comment relies on knowledge of, for instance, the

Method	all-bal*	pol-bal	pol-unbal†
Bag-of-Words	73.2	75.9	27.0
Bag-of-Bigrams	75.8	76.5	24.9
Sentence Embedding	71.0	76.0	26.7
Human (Average)	81.6	83.0	-
Human (Majority)	92.0	85.0	-
Random	50.0	50.0	10.2

* Used only features appearing at least 5 times in the corpus for Bag-of-Words and Bag-of-Bigrams.

† Used only features appearing at least 100 times in the corpus for Bag-of-Words and Bag-of-Bigrams. Measured as average F_1- score scaled by 100

Table 2: Accuracy percentage of baseline methods for sarcasm detection compared to human performance. Tests conducted in the balanced regime for the all subreddits task and balanced and unbalanced regimes for the politics subreddit task

commenter's comment history or relevant news, then human annotators may not perform well. We tried to resolve this issue by using a voting scheme which required several humans to agree about whether or not a comment with its context was sarcastic.

5 Benchmarks for Sarcasm Detection

A direct application of our corpus is for training and evaluating sarcasm detection systems. Thus we use the raw corpus described in Section 3 to construct several useful benchmarks for the task of classifying statements as sarcastic or non-sarcastic. All benchmarks provide the full conversation thread leading up to the target statements to the learning algorithm, along with comment metadata. Following their specification we consider a few context-free baseline methods depending only on linear classification over simple featurizations. Code to reproduce our results is provided at `https://github.com/NLPrinceton/SARC`.

5.1 Evaluation Task

In the most general case, we use the provided raw files to construct datapoints for systems to learn the following task: given a post and a sequence of comments, determine which comments among the responses to the last comment in the sequence are sarcastic. Thus each datapoint consists of a conversation thread followed by a series of responses and sarcasm labels. Performance on this task is measured by average precision, recall and F_1 scores.

Before constructing this subcorpus we first remove from consideration all comments that are not complete sentences and not between 2 and 50 tokens long, allowing for cleaner comments in the evaluation. Although the responses are still largely non-sarcastic, the proportion of sarcastic comments is much greater here as each datapoint must correspond to a thread where at least one sarcastic annotation occurred. In total we construct 8.44 million sequences, with the average proportion of sarcastic responses being 28.1%.

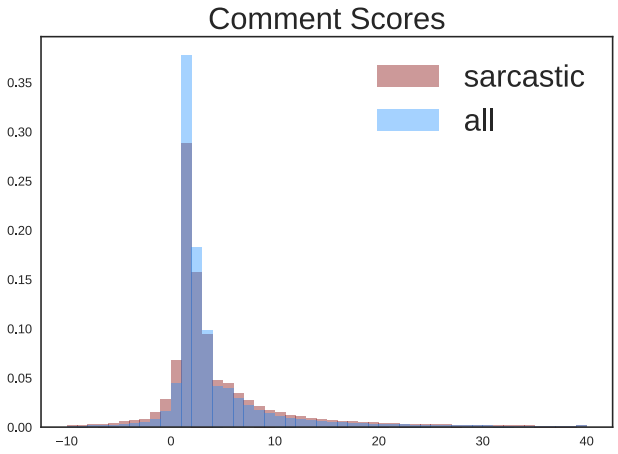


Figure 3: Score distributions of sarcastic and non-sarcastic comments in the raw SARC dataset.

5.1.1 Balanced Labels

We construct a balanced learning task by taking only one sarcastic and one non-sarcastic response from each set of responses to a comment sequence. The task then becomes one of picking which of two statements that share a context is sarcastic, with performance measured by accuracy.

While having only posts with at least one sarcastic response is useful, it also increases the false negative rate as comments warranting a sarcastic response often draw other sarcastic statements that are similar in content to the labeled sarcastic responses, but which themselves may not be labeled. Thus to reduce this issue when picking the non-sarcastic statement, we featurize all statements using the normalized sum of Common Crawl GloVe embeddings of the words and pick from only those non-sarcastic statements that have similarity ≤ 0.95 with the sarcastic statement (Pennington et al., 2014).

5.1.2 Politics

The difficulty of detecting sarcasm rests not only on the need to understand the context of previous statements but also on understanding background information on the topic being discussed. Even humans will struggle with sarcastic comments drawn from unfamiliar topics, for instance, obscure hobbies or art forms. Thus we also test human and machine performance on comments drawn solely from the `politics` subreddit, a topic for which all evaluators had sufficient background information. This subsample contains 17 thousand sequences, with the average proportion of sarcastic responses being 23.2%.

5.2 Methods

For the case of balanced labels, a simple, no-context baseline method for the above task is to featurize the two responses and to train a logistic regression classifier to distinguish between the sarcastic and non-sarcastic response as separate classes. On the testing set, we pick the response with the highest probability of being labeled sarcastic as the sarcastic one. We split both datasets we test on 80%-20% between train-test subsets and report the results of the following three approaches in Table 2.

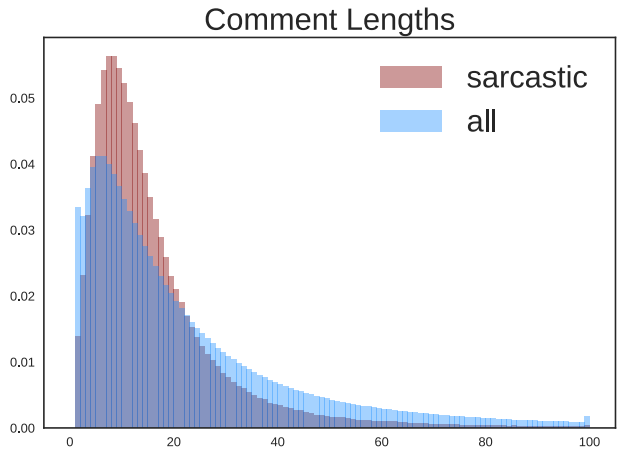


Figure 4: Length distributions of sarcastic and non-sarcastic comments in the raw SARC dataset.

5.2.1 Bag-of-n-Grams

The Bag-of-n-Grams representation consists of using a document's n-gram counts as features in a vector. We test two variants, the Bag-of-Words and the Bag-of-Bigrams. For the subsample containing all subreddits we use only those features that occur at least 5 times in the training comments. We considered including other comment features, such as comment length and scores, but empirical results and the distributions of these features (see Figures 3 and 4) indicate that they are not particularly informative. More sophisticated featurization, such as the noun-phrase and feature interaction indicators proposed by (Wallace et al., 2014), is left to future work.

5.2.2 Sentence Embeddings

Given a document, taking the elementwise sum of embeddings of its words provides a simple low-dimensional document representation. This particular technique of constructing word sequence featurizations has been previously studied and established as a strong baseline for a multitude of supervised NLP prediction tasks (Arora et al., 2017). We use 1600-dimensional GloVe representations trained on the Amazon product corpus, which is used instead of Common Crawl because of the semantic closeness between sentiment and sarcasm (McAuley et al., 2015).

5.2.3 Human

Human sarcasm detection performance was measured by giving 5 human evaluators 100 samples and asking them to perform the same task as the algorithm: determining which of two statements is sarcastic. We provide links to the evaluation survey for the full corpus[5] as well as the politics subreddit[6]. The full context was provided, and the final human classifier was taken as the majority vote of all 5 evaluators.

5.2.4 Random

We use a simple baseline where all responses are labeled sarcastic randomly and independently with a fixed probability. This probability is chosen as the average fraction of responses that are sarcastic in the training set.

[5] www.surveygizmo.com/s3/3878814/SARCmain
[6] www.surveygizmo.com/s3/3878798/SARCpol

Pos. n-Grams	Weights	Neg. n-Grams	Weights
obviously	1.79	:)	-1.37
clearly	1.66	lmao	-1.27
so fun	1.49	:(	-1.17
totally	1.39	:/	-1.17
good thing	1.35	, but	-1.10
shocked	1.32	lol	-1.00
shocking	1.23	the original	-0.98
m sure	1.15	wat	-0.97
omg	1.13	why	-0.96
how dare	1.13	oh god	-0.95

Table 3: Most positive and negative n-grams based on weights assigned by the Bag-of-Bigrams classifier. Positive (negative) weight for an n-gram implies it is a strong indicator for the comment being sarcastic (non-sarcastic). The weights indicate that positive n-grams are more important for linear classification of sarcasm.

5.3 Results

5.3.1 Baselines

The baselines in Table 2 perform reasonably well and much better than the random baseline, but none of them match human performance on either dataset. There is clear scope for improvement for machine learning methods, starting with the use of context provided to make better decisions about sarcasm. As evident in Table 2, Bag-of-Word and Bag-of-Bigram representations perform better than sentence embeddings; however, distributed representations may be necessary for incorporating context in future methods.

5.3.2 Human

As expected, human evaluators performed significantly better, both as a majority and on-average, than the baseline methods. There was significant but not perfect agreement among annotators: on the main dataset the Fleiss kappa score (Fleiss, 1971) was 0.5, indicating moderate agreement, while on the politics subsample it was 0.67, indicating substantial agreement. Interestingly, while individually human performance was worse on average on sequences drawn from all subreddits than on the politics subsample, taking a majority vote among humans led to much better performance in the former case. This performance boost indicates that while individuals may not have enough context for all topics of discussion on Reddit, in aggregate there is enough information to do well, even surpassing the performance on a well-known topic such as politics.

6 Conclusion

We introduce a large sarcasm dataset based on self-annotated Reddit comments. Both the raw data and evaluation subsamples are made freely available, with the former having over 1 million sarcastic sentences, larger than any existing dataset. We evaluate the baseline performance of simple machine learning methods and compare them with human performance. We hope that future users of this dataset will improve upon these benchmarks and find new ways of utilizing the large quantities of self-annotated information we provide.

7 Acknowledgements

We would like to acknowledge Angel Chang and Christiane Fellbaum for helpful discussion and suggestions.

8 Bibliographical References

Amir, S., Wallace, B. C., Lyu, H., Carvalho, P., and Silva, M. J. (2016). Modelling context with user embeddings for sarcasm detection in social media.

Arora, S., Liang, Y., and Ma, T. (2017). A simple but tough-to-beat baseline for sentence embeddings. Conference Paper at ICLR 2017.

Bamman, D. and Smith, N. A. (2015). Contextualized sarcasm detection on twitter. Association for the Advancement of Artificial Intelligence.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382.

González-Ibánez, R., Muresan, S., and Wacholder, N. (2011). Identifying sarcasm in twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 581–586. Association for Computational Linguistics.

Joshi, A., Sharma, V., and Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 757–762. Association for Computational Linguistics.

Joshi, A., Bhattacharyya, P., and Carman, M. J. (2016). Automatic sarcasm detection: A survey. arXiv.

Lukin, S. and Walker, M. (2013). Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. In *Proceedings of the Workshop on Language in Social Media*, pages 30–40. Association for Computational Linguistics.

McAuley, J., Pandey, R., and Leskovec, J. (2015). Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Oraby, S., Harrison, V., Reed, L., Hernandez, E., Riloff, E., and Walker, M. (2016). Creating and characterizing a diverse corpus of sarcasm in dialogue. In *Proceedings of the SIGDIAL 2016 Conference*, pages 31–41. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543. Association for Computational Linguistics.

Ptáček, T., Habernal, I., and Hong, J. (2014). Sarcasm detection on czech and english twitter. In *25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223.

Reyes, A., Rosso, P., and Veale, T. (2013). A multidimensional approach for detecting irony in twitter. *Data Knowledge Engineering*, 47(1).

Riloff, E., Qadir, A., Surve, P., Silva, L. D., Gilbert, N., and Huang, R. (2013). Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–741. Association for Computational Linguistics.

Swanson, R., Lukin, S., Eisenberg, L., Corcoran, T. C., and Walker, M. A. (2014). Getting reliable annotations for sarcasm in online dialogues. In *Language Resources and Evaluation Conference*.

Walker, M. A., Anand, P., Tree, J. E. F., Abbott, R., and King, J. (2012). A corpus for research on deliberation and debate. In *Language Resources and Evaluation Conference*.

Wallace, B. C., Choe, D. K., Kertz, L., and Charniak, E. (2014). Humans require context to infer ironic intent (so computers probably do, too. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 512–516. Association for Computational Linguistics.

Wallace, B. C., Choe, D. K., and Charniak, E. (2015). Sparse, contextually informed models for irony detection: Exploiting user communities, entities, and sentiment. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1035–1044. Association for Computational Linguistics.

HappyDB: A Corpus of 100,000 Crowdsourced Happy Moments

Akari Asai*, Sara Evensen†, Behzad Golshan‡, Alon Halevy‡, Vivian Li‡
Andrei Lopatenko‡, Daniela Stepanov‡, Yoshihiko Suhara‡, Wang-Chiew Tan‡, Yinzhan Xu†
*Univ. of Tokyo, †MIT, ‡Recruit Institute of Technology
akari-asai@g.ecc.u-tokyo.ac.jp, {sevensen, xyzhan}@mit.edu,
{behzad, alon, vivian, andrei, daniela, suharay, wangchiew}@recruit.ai

Abstract

The science of happiness is an area of positive psychology concerned with understanding what behaviors make people happy in a sustainable fashion. Recently, there has been interest in developing technologies that help incorporate the findings of the science of happiness into users' daily lives by steering them towards behaviors that increase happiness. With the goal of building technology that can understand how people express their happy moments in text, we crowd-sourced HappyDB, a corpus of 100,000 happy moments that we make publicly available. This paper describes HappyDB and its properties, and outlines several important NLP problems that can be studied with the help of the corpus. We also apply several state-of-the-art analysis techniques to analyze HappyDB. Our results demonstrate the need for deeper NLP techniques to be developed which makes HappyDB an exciting resource for follow-on research.
Keywords: science of happiness, positive psychology, happyDB corpus, crowdsourcing

1. Introduction

The science of happiness is an area of positive psychology that studies the factors that sustain people's happiness over time (Seligman, 2011; Fredrickson, 2009; Lyubomirsky, 2008). One of the interesting findings of the field (Diener et al., 1999) is that while 50% of our happiness is genetically determined, and only 10% of it is determined by our life circumstances (e.g., finances, job, material belongings), 40% of our happiness is determined by behaviors that are under our control. Examples of such behaviors include investing in long-term personal relationships, bonding with loved ones, doing meaningful work, and caring for one's body and mind. Consequently, positive psychologists have focused on devising methods to steer people towards those behaviors. Fostering happiness has also received attention at the national policy level – in a recent interview (Murthy, 2016) the U.S. Surgeon General claimed that fostering happiness is an important priority as one of the main ways to prevent disease and live a longer, healthier life.

Naturally, there has been recent interest to develop technologies that help users incorporate the findings of the science of happiness into their daily lives. Current applications that pursue this goal generally fall into one of the two categories: (1) applications that suggest relevant content to the users based on their answers to a predefined set of questions (Killingsworth, 2017; Happify, 2017; Happier, 2017) or (2) applications in which users can log their emotions in a journaling-style environment but that content is available mostly for their own reflection (Bliss, 2017; Mojo, 2017; DayOne, 2017).

Our work has been to develop a journal-like application where users express their happy moments using their own language, thereby allowing for more nuance in their description of what makes them happy. The ultimate goal of our app is that it should understand from the text which activities make the user happy and who else participated in those happy moments. The app can then provide a useful visualization of the user's happy moments, offer meaningful follow-up questions, and over time learn to suggest other activities that may benefit the user.

As we started working on this application we quickly realized that understanding the different aspects of happy moments is a challenging NLP problem that has received very little attention to date. In order to advance the state of the art for this problem, we set out to crowd-source HappyDB, a corpus of 100,000 moments that we released publicly at `http://rebrand.ly/happydb`.

This paper describes the HappyDB corpus, and outlines a few NLP problems that can be studied with it. We describe the application of a few state-of-the-art analysis techniques to the corpus resulting in several observations. We also discuss some additional annotations that we provide along with HappyDB that would be useful to anyone who wants to explore the corpus further. The upshot of all these analyses, however, is that there is a need for deeper NLP techniques in the analysis of happy moments (and of emotions expressed in text in general), and thus HappyDB provides an exciting opportunity for follow-up research.

In addition to the applications motivating our work, there are other areas in which a deeper understanding of happy moments can be useful. Of particular note is the analysis (by advertisers or third parties) of the sources of happiness relating to products and services from comments on social media. Viewed in that perspective, analyzing happy moments can also be seen as a refined analysis of sentiments (e.g., (Liu, 2012; Pang and Lee, 2008)).

HappyDB is a collection of sentences in which crowdworkers answered the question: *what made you happy in the past 24 hours* (or alternatively, the past 3 months). Naturally, the descriptions of happy moments exhibit a high degree of linguistic variation. Note that HappyDB is *not* a longitudinal dataset that follows individuals over a period of time. Some examples of happy moments are:

1. *My son gave me a big hug in the morning when I woke him up.*
2. *I finally managed to make 40 pushups.*
3. *I had dinner with my husband.*

Part of the work was done while Asai, Evensen, and Xu were at Recruit Institute of Technology.

4. *Morning started with the chirping of birds and the pleasant sun rays.*
5. *The event at work was fun. I loved spending time with my good friends and laughing.*
6. *I went to the park with the kids. The weather was perfect!*

Fully understanding a happy moment is obviously a problem that goes beyond natural language processing into the fields of psychology and philosophy. Here we take an NLP perspective on the problem and set a goal of understanding which activities happened in the happy moment and who participated in those activities. Evaluating which of these activities is the true cause of happiness adds another level of complexity. For example, even for the very simple happy moment *I had dinner with my husband*, the extracted activities could be "having dinner", "being with the husband", or something that is not explicitly in the text such as "having a date night without the children".
The following are several NLP-related problems that could be studied using HappyDB.

1. What are the activities described in a given happy moment? What other components besides activities are important in the happy moment? Which of these aspects are most central to the happy moment?

2. Can we discover common paraphrasings to describe activities that appear in happy moments?

3. Can we discover whether the cause of happiness in a particular happy moment is related to the expectation the person had? For example, a happy moment can be written as *I got to spend time with my son* versus *I spent time with my son*. In the first case it seems that the person was partially happy because they didn't expect to be able to spend time with their son.

4. Can we reliably remove extraneous text in a happy moment? For example can we transform, "*I am happy to hear that my friend is pregnant*" to "*My friend is pregnant*". Note that removing extraneous information can be very helpful in understanding which activity or event is the cause of happiness.

5. Can we create a useful ontology of activities that cause happiness and map happy moments onto that ontology. Such an ontology can be an important tool for recommending additional activities to the user.

Solutions to the questions raised above will require advances in NLP. In particular, we need techniques that go beyond analysis of the happy moments at the keyword level and perform deeper analyses such as semantic role labeling (into possibly a set of frames that leverage Framenet, Verbnet, and/or Propbank). Further analysis also needs to accommodate ungrammatical sentences such as "*Early morning in the beach, having breakfast with the family.*"
In this paper, we lay the groundwork for a deeper exploration of HappyDB. We begin by describing how HappyDB was collected and cleaned. We present some basic statistics about HappyDB demonstrating that it is a broad corpus. We



Figure 1: HappyDB's word cloud – an anecdotal overview of the corpus. The words "work" and "friend" appear most prominently in HappyDB; mentions of "wife" and "husband" occur about equally, and so do "son" and "daughter". However, "girlfriend" occurs more often than "boyfriend" (1960 vs. 1252 times), "night" appears more often than "morning" (3391 vs. 2736 times), and "dog" occur much more often than "cat" (2160 vs. 988 times).

compare the topics and the emotional content of our corpus with other corpora using standard state-of-the-art annotations which we are releasing with HappyDB. We also illustrate another interesting aspect of HappyDB: moments describing experiences from the last 24 hours are significantly different from those describing experiences from the last 3 months. Finally, we address the most basic research problem at the heart of HappyDB: classifying happy moments into categories. We show that even this problem is extremely challenging as it is closely related to the problem of mining expressed emotions in short sentences. We describe a set of crowd-sourced category annotations that will facilitate future research in the problem.

2. HappyDB: 100,000 happy moments :)

We collected 100,000 happy moments with Mechanical Turk (MTurk) over 3 months. The workers were asked to answer either: *what made you happy in the last 24 hours?* or, *what made you happy in the last 3 months?* HappyDB is split evenly between these two reflection periods. The majority of our workers are of age 20 to 40 years and from the USA. There are about the same number of male and female workers and the majority of our workers are single. More information about the demographics of the workers as well as our crowd-sourcing setup can be found in appendices A and B respectively. Along with the original 100,000 happy moments (which we refer to as the *original* HappyDB), we also released a cleaned version of HappyDB (which we refer to as the *cleaned* HappyDB), where some spelling mistakes are corrected (as described below) and some vacuous moments are removed. Each moment is also annotated with the reflection period (24 hours or 3 months) and with the demographic information of the worker providing it.

Cleaning HappyDB: Naturally, the collected happy moments can contain a variety of errors. In our cleaning pro-

Collection period	3/28/2017 – 6/16/2017
# happy moments	100,922
# distinct workers	10,843
# distinct words	38,188
Avg. # happy moments / worker	9.31
Avg. # words / happy moment	19.66

Table 1: Basic statistics on HappyDB

cess we dealt with two types of errors: (1) empty or single word sentences and (2) sentences with spelling errors. We removed any sentences with less than two words. To find the spelling errors, we compared all the words to a dictionary built from Norvig's text corpus (Norvig, 2007) as well as a complete list of English Wikipedia titles[1] which includes the name of many cities, locations and other known entities. We also performed a few edits on the dictionary to remove foreign language phrases as well as certain words such as 'Alot' and 'Iam' which are actual city names, but are more likely to be spelling errors. We found that only 2.7% of happy moments contain words not present in our dictionary. While this number seems small enough to justify removing such happy moments, we observe that certain words are more likely to be misspelled and could create a bias if we remove these happy moments. A specific example is that mentions of the word "son" is higher than "daughter" in the original corpus because the word "daughter" is more likely to be misspelled than the word "son". After fixing the typos using our technique (which we describe next), both words ended up having almost the same frequency. This example indicates that there is a need for the spell-corrector.

To fix the spelling issues, we experimented with various open-source spell correctors, but we didn't find them suitable for our task; they either didn't provide confidence scores for the corrections, or suggested corrections that would have a higher likelihood in other corpora, but not ours. For example, in the context of happy moments, the phrase "achive" is more likely to be a typo for "achieve" than for "active". Thus, we decided to develop a spell corrector that is tailored to the domain of HappyDB and only corrects typos that we are highly confident of. The details of our spell-corrector are presented in appendix C.

Some basic statistics: Table 1 shows some basic statistics of the original HappyDB. Figure 1 shows the word cloud for the cleaned HappyDB. The figure is mostly provided for anecdotal value and as a means to highlight the most frequent words in the corpus. As one proxy for the complexity of the sentences in HappyDB, we calculated the number of verbs in each sentence which are summarized in Figure 2. The data shows that 53% of the sentences have 3 verbs or more and 36% of the sentences have 4 verbs or more meaning that workers definitely expressed quite complex thoughts in their moments.

Diversity of contents in HappyDB: An important question concerning the utility of HappyDB is whether it covers happy moments from a variety of topics. To get a feel for the level of diversity, we identified 9 rather diverse top-

[1] https://dumps.wikimedia.org/enwiki/

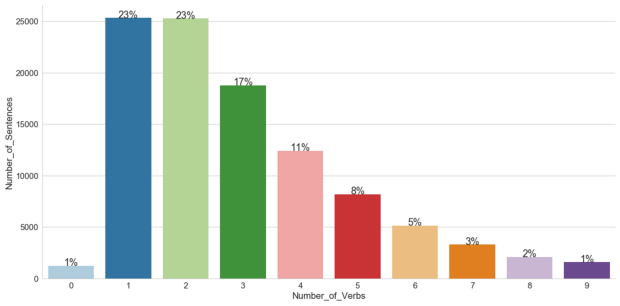


Figure 2: The distribution of the number of verbs per happy moment

Topics	% of Sentences in Topic	Size of Keywords List
people	46.0	478
family	26.4	423
food	16.2	1073
work	14.5	115
entertainment	8.8	156
exercise	8.4	558
shopping	8.4	35
school	5.5	47
pets	4.5	149
none	20.3	N/A

Table 2: HappyDB Topics Distributions

ics we saw occurring often in the corpus. The topics are "people", "family" (a subset of "people"), "pets", "work", "food", "exercise", "shopping", "school", and "entertainment". For each topic, we curated a list of keywords and regular expressions whose usage is almost exclusive to the topic. For example, the category "people" contains words describing family members, as well as other words that refer to people, like "hairdresser" or "neighbor", but it does not include "he", "she" or "they", as these words are sometimes used in reference to pets or inanimate objects. Additionally, if these pronouns refer to a person they should also have an antecedent which our dictionary should recognize. "People" also does not contain the word "I", since we are trying to capture interactions between people.

Table 2 shows the percentage of sentences in HappyDB found for each topic as well as the size of the list associated with each topic. Note that a happy moment may be related to multiple topics. For instance, "*running with my son*" is related to both "family" and "exercise". All of the keywords lists are disjoint except for "people": this is a superset of "family", and also contains some words from other topics, for example "co-worker" which is also in "work". We can observe that 80% of HappyDB pertains to these 9 topics. The remaining sentences that did not fit into any of these topics contain all sorts of topics, such as rare surprises ("*finding a $100 dollar bill inside my pants pockets*") or situations that turned out to be better than expected ("*There was almost no traffic today*"). None of these other categories covered a large enough portion of our corpus to justify adding them to our dictionaries. However, this suggests that there is a long tail of topics in the corpus.

Another perspective on the contents of HappyDB can be obtained by annotating the corpus with the popular semantic classes known as *supersense*. Supersense tags are defined in WordNet (Fellbaum, 1998) as lexicographic classes and are categorized into 15 verb classes (e.g., stative, cognition, communication, social, motion etc.) and 26 noun classes (e.g., person, artifact, cognition, food etc.).

We trained a supersense tagger with the SemEval-2016 dataset (Schneider and Smith, 2015) using CRF (Okazaki, 2007). The supersense annotated HappyDB is also provided as part of HappyDB. Table 3 shows the proportion of sentences for the top seven supersense labels in HappyDB. It also displays the proportion of supersense labels for sentences in other textual corpora from the Manually Annotated Sub-Corpus dataset[2]. As shown, the proportions of several of the top five labels for HappyDB are significantly higher than the other corpora which implies that these labels are potential features for identifying happy moments. Examples of some supersense classes and their frequencies in HappyDB are shown in Table 4.

2.1. Emotions in happy moments

To analyze the cognitive and emotional state of happy moments, we applied the sentiment lexicon Linguistic Inquiry and Word Count (LIWC), "a transparent text analysis program that counts words in psychologically meaningful categories", (Pennebaker et al., 2015b) on a sample of 500 happy moments; only a sample was chosen because of existing restrictions on the amount of requests to the LIWC commercial API. Table 5 shows some of the LIWC categories in which the scores for the 500 happy moments vary notably from those of other corpora (expressive writing, blog posts, and novels (Pennebaker et al., 2015a)). These categories are defined as follows: *analytic* refers to a measurement of the author's logical thinking, as opposed to narrative and informal thinking; *authentic* approximates how honest and disclosing the writing is; and *tone* measures how positive or negative the text is. As expected, HappyDB has a higher score for tone than any of the other corpora analyzed. More interestingly, the analytic score for HappyDB is quite high and very close to that of Novels, yet the authentic score (also quite high) is closer to that for Expressive Writing. Our analysis of LIWC scores suggest that our corpus is very disclosing and honest which makes it an ideal corpus for studying emotions expressed in text.

An alternative approach to analyze the emotions expressed in text is to use the Valence-Arousal-Dominance model (VAD) of emotion (Bradley and Lang, 1994; Warriner et al., 2013) which provides a score for each lemmatized word on a scale of pleasure-displeasure (valence), excitement-calm (arousal), and control-inhibition (dominance). To evaluate our data across these dimensions, we used the Warriner et al. database of 13,915 manually rated English lemmas, as averaged over at least 18 ratings for all three VAD features (Warriner et al., 2013). This is currently the largest available lexicon of VAD scores, and the VAD ratings covered 45.84% of the lemmatized words in HappyDB. The words which were not covered were mostly pronouns, articles, conjunctions, numbers, and proper nouns. Some examples of the highest and lowest scoring words across each dimension are listed in Table 6. We calculated a VAD score for HappyDB by taking the mean over the VAD score of words in the corpus. Interestingly, we observed that HappyDB's VAD score is similar to the *travel* section of the Guardian corpus (Brett and Pinna, 2013) (V≅6.2, A≅4.0, D≅5.7) and rather different from other sections such as *crime* or *banking*. This shows that the VAD scores (which we release as part of HappyDB) can help us quantify how emotional the content of the corpus is.

The conclusion from the analyses provided so far in this section is that HappyDB is a diverse corpus with content that is emotionally rich and covers various topics (e.g., "work", "leisure", "exercise" and etc.). Furthermore, while we used several techniques to extract general statistics about the content, diversity, and emotional content of HappyDB, there is clearly a need for deeper analysis of happy moments.

2.2. Comparing Reflection Periods

The analyses presented thus far, though rather rudimentary, already enable us to discover an important property of HappyDB, namely that there are important differences between the happy moments that reflect on the last 24 hours versus those that reflect 3 months back. In addition to being an important property of the corpus, these differences raise additional interesting research questions. We demonstrate these differences in two ways.

Pointwise Mutual Information Scores: For each reflection period we calculated pointwise mutual information (PMI) scores (Manning et al., 2008) for words in the cleaned happy moments, and compared the top nouns in each batch. Table 7 shows the top 10 nouns with the highest PMI scores in the 24 hours batch w.r.t. the other batch and vice-versa. The results suggest that moments reported in the 24 hour period tend to be activities that occur daily (e.g., foods, bedtime) and moments reported in the 3 months period tend to reflect infrequent occurrences like holidays or life events.

Topic Mentions by Reflection Period: We analyzed the incidence of different topics separately for each reflection period. In Table 8 we observe different distributions of topics for each reflection period, mainly in the categories "food", "school", "people", "family", and "entertainment". For instance, we observe that the categories "food" and "entertainment" have higher percentage of coverage in 24 hour (19.2%, and 9.6%) compared to the 3 months reflection period (13.1%, and 7.8%). Naturally, people are more likely to talk about a meal or a movie because these are more frequent daily events that are more likely to be remembered if they occurred recently. When people are asked to reflect on the past 3 months, they tend to remember events that are more prominent such as school, big achievements, and time spent with friends and family.

3. Categorizing Happy Moments

So far we have gained an understanding of HappyDB through some analysis of annotations that are frequently

[2]MASC v3.0.0 http://www.anc.org/data/masc/downloads/data-download/

	n.person	n.time	v.social	v.motion	v.possession	n.event	n.food
HappyDB	0.48	**0.41**	**0.31**	**0.26**	**0.22**	**0.18**	**0.13**
Blog	0.28	0.12	0.15	0.07	0.04	0.05	0.06
Newspaper	0.39	0.11	0.12	0.06	0.04	0.05	0.07
Twitter	**0.51**	0.11	0.11	0.07	0.03	0.04	0.05

Table 3: Distribution of top-7 most frequent supersense categories in HappyDB and other corpora. The number is the proportion of sentences that contain a particular supersense label. Bold-face denotes the highest value in a column.

n.person		n.food		v.emotion	
friend	9,493	food	1,511	enjoy	2,778
son	3,507	dinner	938	love	2,080
daughter	3,314	coffee	681	feel	1,399
wife	2,721	pizza	568	like	1,070
husband	2,687	breakfast	525	excite	533

Table 4: Most common words associated with frequently used supersense labels.

LIWC Category	**HappyDB**	**Expressive**	**Novels**	**Blogs**
analytic	69.19	44.88	70.33	49.89
authentic	77.14	76.01	21.56	60.93
tone	60.85	38.60	37.06	54.50

Table 5: Average LIWC Scores for (cleaned) HappyDB compared to those of other text corpora of expressive writing, novels, and blogs (Pennebaker et al., 2015a). All three categories provide scores in the range from 1 to 99.

Category	Low Scoring Words	High Scoring Words
Valence	murder (1.48) leukemia (1.47)	excited (8.11) happiness (8.48)
Arousal	librarian (1.75) calm (1.67)	rampage (7.57) lover (7.45)
Dominance	Alzheimer's (2.00) earthquake (2.14)	completion (7.73) smile (7.72)

Table 6: Examples of words with very high or low VAD scores (Warriner et al., 2013).

24 hours		**3 months**	
Word w	**Ratio** (X_w/Y_w)	**Word** w	**Ratio** (Y_w/X_w)
bedtime	15.0	valentine	45.0
custard	12.0	scenario	34.0
spoon	12.0	sorrow	24.0
burritos	10.0	gender	20.0
nachos	10.0	thousand	17.0
opener	10.0	custody	17.0
fool	10.0	faculty	16.0
dough	9.0	palace	14.0
hurry	9.0	propose	13.0
gossip	9.0	military	11.0

Table 7: Top PMI words from the two batches. X_w (resp. Y_w) denotes the probability of word w occurring in the 24 hours batch (resp. 3 months batch)

Topics	**% of sentences in topic**	
	24 Hours Reflection	**3 Months Reflection**
people	44.0	47.9
family	24.7	27.9
pets	4.6	4.4
work	14.4	14.5
food	19.2	13.1
exercise	8.7	8.1
shopping	7.6	9.0
school	4.3	6.7
entertainment	9.6	7.8
None	20.4	20.1

Table 8: Percentage of happy moments in each topic, separated by reflection period. All differences between the columns are statistically significant at $p < 10^{-3}$ except for pets, work, and none.

considered in the literature. In this section, we take a first step towards a deeper analysis of happy moments by trying to classify them into categories. Categorization is important for several reasons. First, it forms the basis for visualizing one's happy moments. Second, the techniques for analyzing happy moments may depend partially on the category they belong to. Finally, the category of a happy moment could trigger a conversation between an app and a user, and the course of conversation is clearly dependent on the category being discussed. For instance, the app's response to a happy moment about completing an exercise may be to congratulate the user, but the same response would be unacceptable if the user mentions that she is enjoying a beautiful scenery.

There is no consensus on a single set of categories for happy moments in positive psychology because they are often discussed under different names, with small variations and at different levels of granularity. We chose a set of categories inspired by research in positive psychology that also reflects the contents of HappyDB. These categories and a brief description of them are listed in Table 9. Note that *affection* refers to activity with family members and loved ones, while *bonding* refers to activities with other people in one's life.

We developed a multi-class classifier using Logistic Regression with a bag of words representation of happy moments as features. To obtain training data, we crowdsourced a batch of $15,000$ happy moments to obtain category labels. Every happy moment was shown to 5 workers, and we only considered labels that at least 3 workers agreed on. Table 10 shows the performance of our classifier using a 5-fold cross-validation setup. Clearly the clas-

Category	Definition	Examples
Achievement	With extra effort to achieve a better than expected result	Finish work. Complete marathon.
Affection	Meaningful interaction with family, loved ones and pets	Hug. Cuddle. Kiss.
Bonding	Meaningful interaction with friends and colleagues	Have meals w coworker. Meet with friends.
Enjoy the moment	Being aware or reflecting on present environment	Have a good time. Mesmerize.
Exercise	With intent to exercise or workout	Run. Bike. Do yoga. Lift weights.
Leisure	An activity done regularly in one's free time for pleasure	Play games. Watch movie. Bake cookies.
Nature	In the open air, in nature	Garden. Beach. Sunset. Weather

Table 9: The categories of happy moments

				% of moments	
Category	**Precision**	**Recall**	**F1**	**24 Hrs**	**3 Months**
Achievement	79.2	87.3	83.0	30.9	36.5
Affection	89.9	94.3	92.0	32.7	35.1
Bonding	91.9	87.1	89.4	10.4	10.8
Enjoy the moment	59.2	49.9	54.0	13.3	8.9
Exercise	85.3	59.3	69.9	1.5	0.8
Leisure	77.9	67.2	72.1	8.7	6.1
Nature	80.9	52.4	63.4	2.1	1.5

Table 10: Precision/recall/F1 scores for each category as well as the % of moments per category for each reflection period. All differences between the reflection periods are statistically significant at $p < 10^{-5}$ except for bonding.

sifier has room for further improvement on categories such as "Leisure" and "Enjoy the moment" which shows that word distributions are not sufficient for this task. Building a classifier with sufficiently high precision/recall scores on all categories can be a challenging task in general, as it usually involves inferring some information or context that is not explicitly mentioned in text.

We publish our crowd-sourced labels as part of HappyDB to provide a ground-truth for researchers interested in topic mining and clustering of short utterances. We also released our predicted results on the entire corpus as a baseline.

Table 10 also shows the percentage of moments classified into each category for both reflection periods, which further highlights the differences between reflection periods described in Section 2.. Notice that HappyDB has roughly the same number of moments for each reflection period. Thus, the percentage of moments classified in each category in HappyDB can be computed by taking the average of the last two columns in Table 10. The higher frequency of moments in "Exercise", "Nature", and "Leisure" under the 24 hours reflection period confirms our theory that daily tasks are sources of short-term happiness. Longer-term happiness is more likely to come from loved ones or achievements.

4. Related work

To the best of our knowledge, HappyDB is the first crowd-sourced corpus of happy moments which can be used for understanding the language people use to describe happy events. There has been recent interest in creating datasets in the area of mental health. Althoff et al. (Althoff et al., 2016) conducted a large scale analysis on counseling conversational logs collected from short message services (SMS) for mental illness study. They studied how various linguistic aspects of conversations are correlated with conversation outcomes. Mihalcea et al. (Mihalcea and Liu, 2006) performed text analysis on blog posts from LiveJournal (where posts can be assigned happy/sad tags by their authors). Lin et al. (Lin et al., 2016) measure stress from short texts by identifying the stressors and the stress levels. They classified tweets into 12 stress categories defined by the stress scale in (Holmes and Rahe, 1967).

Last year alone, there were multiple research efforts that obtained datasets via crowdsourcing and applied natural language techniques to understand different corpora. For example, SQuAD (Rajpurkar et al., 2016) created a large-scale dataset for question-answering. The crowdsourced workers were asked to create questions based on a paragraph obtained from Wikipedia. They employed MTurk workers with strong experience and a high approval rating to ensure the quality of the dataset. We did not select the workers based on their qualification for HappyDB as our task is cognitively easier than SQuAD's and we want to avoid bias in our corpus. HappyDB is similar to SQuAD in terms of the scale of the crowdsourced dataset. However, unlike SQuAD, which was designed specifically for studying the question answering problem, the problems that HappyDB can be used for are more open-ended.

5. Conclusion

We have published HappyDB, a broad corpus of happy moments expressed in diverse linguistic styles. We have also derived a cleaned version of HappyDB, added annotations, and presented our analysis of HappyDB based on these annotations. We made our dataset and most of our annotations publicly available to encourage further research in the science of happiness and well-being in general. We believe that HappyDB can spur research of the topic of understanding happy moments and more generally, the expression of emotions in text. The results of this research can translate to applications that can improve people's lives.

6. Bibliographical References

Althoff, T., Clark, K., and Leskovec, J. (2016). Large-scale Analysis of Counseling Conversations - An Application of Natural Language Processing to Mental Health. *TACL*.

Bliss. (2017). Gratitude journal - bliss. http://bliss31.com/.

Bradley, M. M. and Lang, P. J. (1994). Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, 25(1):49–59.

Brett, D. and Pinna, A. (2013). The Distribution of Affective Words in a Corpus of Newspaper Articles. *Procedia: social and behavioral sciences*, 95:621–629.

DayOne. (2017). Day one - a simple and elegant journal for iphone. http://dayoneapp.com/.

Diener, Suh, Lucas, and Smith. (1999). Subjective well-being: Three decades of progress. *Psychological Bulletin*, pages 276–302.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Fredrickson, B. (2009). *Positivity: Top-Notch Research Reveals the Upward Spiral That Will Change Your Life*. Harmony.

Happier. (2017). Happier: Gratitude journal, meditation, and celebrating the good around you. https://www.happier.com/.

Happify. (2017). Happify: Science-based happiness games activities. www.happify.com/.

Holmes, T. H. and Rahe, R. H. (1967). *The social readjustment rating scale*, volume 11. Journal of psychosomatic research.

Killingsworth, M. (2017). Track your happiness. https://www.trackyourhappiness.org.

Lin, H., Jia, J., Nie, L., Shen, G., and Chua, T.-S. (2016). What Does Social Media Say about Your Stress?. *IJCAI '16*.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Lyubomirsky, S. (2008). *The How of Happiness: A New Approach to Getting the Life You Want*. Penguin Books; Reprint edition (December 30, 2008).

Manning, C. D., Raghavan, P., Schütze, H., et al. (2008). *Introduction to information retrieval*. Cambridge university press Cambridge.

Mihalcea, R. and Liu, H. (2006). A corpus-based approach to finding happiness. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 139–144. AAAI.

Mojo. (2017). Mojo: My gratitude journal. http://welovemojo.com/.

Murthy, V. (2016). http://www.huffingtonpost.com/entry/surgeon-general-happiness-vivek-murthy_us_564f857ee4b0d4093a57c8b0.

Norvig, P. (2007). How to write a spelling corrector. http://norvig.com/spell-correct.html.

Okazaki, N. (2007). Crfsuite: a fast implementation of conditional random fields (crfs). http://www.chokkan.org/software/crfsuite/.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

Pennebaker, J. W., Boyd, R. L., Jordan, K., and Blackburn, K. (2015a). The development and psychometric properties of liwc2015. Technical report.

Pennebaker, J., Booth, R., Boyd, R., and Francis, M. (2015b). Linguistic inquiry and word count: Liwc2015. austin, tx: Pennebaker conglomerates.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *EMNLP*, pages 2383–2392.

Schneider, N. and Smith, N. A. (2015). A Corpus and Model Integrating Multiword Expressions and Supersenses. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1537–1547.

Seligman, M. E., Steen, T. A., Park, N., and Peterson, C. (2005). Positive psychology progress: empirical validation of interventions. *American psychologist*, 60(5):410.

Seligman, M. E. P. (2011). *Flourish: A New Understanding of Happiness, Well-being - and how to Achieve Them*. Nicholas Brealey.

Tversky, A. and Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science*, 211(4481):453–458.

Warriner, A. B., Kuperman, V., and Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior research methods*, 45(4):1191–1207.

A Demographic information of HappyDB crowdsourcing workers

The following tables represent the demographic distributions of the crowdsourcing workers who contributed to our HappyDB dataset.

Age	Ratio
10–20	4.22%
20–30	47.77%
30–40	29.34%
40–50	10.22%
50–60	5.89%
60–70	2.15%
70–80	0.30%
80–90	0.04%

Table 11: Age distribution.

Country	Ratio
USA	86.11%
IND	8.94%
CAN	0.61%
VEN	0.50%
GBR	0.44%
OTHERS	3.40%

Table 12: Country of residence distribution.

Gender	Ratio
Female	50.37%
Male	49.12%
Not specified	0.51%

Table 13: Gender distribution.

Marital status	Ratio
Single	52.66%
Married	40.52%
Divorced	5.22%
Separated	0.95%
Widowed	0.66%

Table 14: Marital status distribution.

Parenthood status	Ratio
No	59.64%
Yes	40.36%

Table 15: Parenthood status distribution.

B Data collection by crowdsourcing

We investigated several parameters before collecting our corpus. First, we investigated whether differences in our instructions to the workers will influence the happy moments collected. Second, we experimented with different windows of reflection (i.e., how far in the past did the happy moment occur). We did this to understand how the time period influences the content of happy moments. By analyzing the outcomes on batches of 300 moments that were collected by systematically varying these parameters, we embarked on a large-scale collection of $100,000$ happy moments. We also experimented with two platforms: Mechanical Turk and Crowd Flower. We did not notice a significant difference in the quality or content of the moments so we used Mechanical Turk.

Instructions for workers (I_1):
What made you happy today? Reflect on the past 24 hours, and recall three actual events that happened to you that made you happy. Write down your happy moment in a complete sentence. Write three such moments.
Examples of happy moments we are looking for:
- I went for a run in the neighborhood. I enjoyed the perfect weather.
- The offsite with colleagues was great fun. We had stimulating discussions.
- My son gave me a big hug in the morning when I woke him up.
- I finally managed to make 40 pushups.
- I enjoyed watching the sunset on the porch.

Examples of happy moments we are NOT looking for (e.g., events in distant past, partial sentence):
- The day I married my spouse
- My Dog

Figure 3: Instructions I_1 with positive examples

B1. Instructions for workers

Following (Seligman et al., 2005) who developed a questionnaire that included a question that asked for 3 good things that went well each day, we asked our MTurk workers for 3 happy moments that happened to them in the past 24 hours. To minimize the bias we introduce through our instructions, we carefully analyze the effect our examples of happy moments have on the way crowdsource workers report their moments. In our first batch of crowdsourcing, we gave concise instructions and positive examples of what we believed are legitimate happy moments (see Figure 3). Upon collecting our first batch of 300 happy moments, we tabulated the top 20 most frequent words (nouns/verbs/adjectives) that occur in the happy moments. It was surprising to observe that the words used in the positive examples often appear in the top 20 most frequent words. (See the first two columns of top of Table 16). For example, the bold words "morning", "enjoy", "woke", "son", and "great" (excluding "went"), which are words we use in our positive examples, appear highly among the top 20 words. Consequently, we experimented with an instruction set *without* the positive examples (I_2) and collected another 300 happy moments. The top of Table 16 shows the top 20 most frequent words. When we compare the moments collected under I_1 and I_2, it is evident that the bold words no longer appear highly ranked.
The bottom of Table 16 shows a quantitative representation of the *framing effect* (Tversky and Kahneman, 1981) of the I_1 instruction set. Here, $\mathcal{V}$ denotes the set of all words (no duplicates and no stopwords) obtained from our 5 positive examples in the I_1 instruction set. We counted the number of times words in $\mathcal{V}$ occur in the happy moments obtained from I_1 instruction and, respectively, I_2 instruction. In addition, we also counted the number of times words outside $\mathcal{V}$ occur in the respective batches of happy moments.

I_1	Count	I_2	Count
got	37	made	42
went	34	long	23
made	27	got	23
great	25	family	18
morning	21	went	18
friends	20	team	17
nice	16	favorite	17
super	16	go	16
may	15	work	15
bowl	14	game	15
enjoyed	14	movie	14
son	14	**morning**	14
work	14	friend	14
really	14	first	13
family	14	bought	13
see	13	dinner	12
dinner	13	really	12
woke	13	get	12
daughter	12	friends	11
coffee	12	saw	11

	# words in batch from $\mathcal{V}$	# words in batch not from $\mathcal{V}$
I_1	213	3,115
I_2	126	3,452

Table 16: The top shows the most used words in happy moments for instruction sets I_1 and I_2. The bottom shows the frequencies in I_1 and I_2 w.r.t. $\mathcal{V}$, the vocabulary in our positive examples.

A χ^2 analysis (Manning et al., 2008) shows that the presence of the 5 positive examples in I_1 does affect the word

usage of workers. Specifically, our null hypothesis is that the word usage of the happy moments is independent of whether the instruction set contained positive examples or not. The χ^2-test rejected the null hypothesis with p-value < 0.001. Hence, we conclude that MTurk workers were influenced by the positive examples in our instructions when reporting their happy moments and decided against positive example sentences in the instructions for collecting happy moments. It is interesting to note that the bottom of Table 16 shows also that the vocabulary of happy moments from I_1 instruction set is significantly minimized, 3,328 total words used in the I_1 batch versus 3,578 in the I_2 batch. From this analysis, we concluded that we should avoid using positive examples in our instructions. We also experimented with instructions that do not include negative examples. However, apart from some reduction in the number of low-quality happy moments, we did not detect significant differences between happy moments that are collected from instructions with or without negative examples. Hence, we included negative examples in our instructions for the workers.

C A Spell-Corrector for HappyDB

Here, we discuss the details of the spell-correction algorithm that we have created for HappyDB. Our main goal is to fix as many typos while introducing as little error as possible. To this end, we have decided to focus on a small set of corrections: typos that are within a Levenshtein distance of 1 of a valid word (i.e., one deletion, insertion, transposition, or replacement of a letter or a space).

The spell-correcting algorithm starts by finding the set of words within edit distance 1 of a typo and computes a confidence score $C(w)$, for each word w which we defined as $C(w) = \log(f_w)$ where f_w is the frequency of the word w. If w consists of two words (which occurs with replacement or insertion of a space character) then f_w is the lower frequency of the two words. We calculate these frequencies using a corpora which consists of Norvig's corpus and the portion of HappyDB that has no spelling errors. We observed that resulting corrector was biased toward splitting words into two, for example "outtdoors" was being replaced with "out doors" instead of "outdoors". This is because shorter words occur more frequently. For the same reason, words were often being replaced by an incorrect but shorter alternative, for example "helpd" being replaced with "help" (instead of "helped"). In an attempt to solve this problem, we refined the confidence score by adding two additional parameters: $0 \leq s \leq 1$, to discount the confidence of replacements with an inserted space, and l, to increase the confidence of longer words. This updated confidence score $C(w)$ can be written as

$$C(w) = \log(f_w)s^{b(w)} + l \times \text{len}(w)$$

where $b(w)$ is a simple indicator function which returns 0 if w consists of two words and returns 0 otherwise. Note that we are using the logarithmic frequency in our definition. The hypothesis is that shorter words occur exponentially more often than long words on average, in which case computing confidence as a function of the logarithmic frequency would yield better results. We also observed this effect in practice. The last step is to tune the parameters s and l. To tune these parameters, we took a random sample of 100 spelling errors from our data and manually made corrections. We then perform a grid search over possible values for l and s. Our experiment suggests that conservative values for l and s are 0.5 and 0.05, respectively. With these values, in a random sample of $3,000$ happy moments, all 74 detected typos were either corrected appropriately or left as is.

We applied our spell-corrector on the entire corpus, which automatically replaced $1,568$ words. The remaining typos were corrected by workers on CrowdFlower, as well as internal workers in our lab. Each word was evaluated by two judges, and if they agreed, the result was automatically applied. There were less than 500 words where the judges did not agree. In this case, we defaulted to the worker with higher confidence rating (our internal lab workers). In the remaining cases where the confidence ratings were equal, we left the word alone if either judge was unsure, otherwise chose a suggestion at random (in the majority of these cases, the answers varied in usage of space, punctuation, or capitalization, and not in content). In total, $2,218$ happy moments were modified with this method.

MultiBooked: A Corpus of Basque and Catalan Hotel Reviews Annotated for Aspect-level Sentiment Classification.

Jeremy Barnes, Patrik Lambert, Toni Badia
Pompeu Fabra University, Barcelona, Spain
{firstname.lastname}@upf.edu

Abstract

While sentiment analysis has become an established field in the NLP community, research into languages other than English has been hindered by the lack of resources. Although much research in multi-lingual and cross-lingual sentiment analysis has focused on unsupervised or semi-supervised approaches, these still require a large number of resources and do not reach the performance of supervised approaches. With this in mind, we introduce two datasets for supervised aspect-level sentiment analysis in Basque and Catalan, both of which are under-resourced languages. We provide high-quality annotations and benchmarks with the hope that they will be useful to the growing community of researchers working on these languages.

Keywords: basque, catalan, sentiment analysis, aspect-level, under-resourced, opinion mining, cross-lingual

1. Introduction

Sentiment analysis has become an established field with a number of subfields (aspect-level sentiment analysis, social media sentiment analysis, cross-lingual sentiment analysis), all of which require some kind of annotated resource, either to train a machine-learning based classifier or to test the performance of proposed approaches.

Although much research into multi-lingual and cross-lingual sentiment analysis has focused on unsupervised or semi-supervised approaches (A.R. et al., 2012; Perez-Rosas et al., 2012; Gao et al., 2015), these techniques still require certain resources (linked wordnets, seed lexicon) and do not generally reach the performance of supervised approaches.

In English the state-of-the-art for binary sentiment analysis often reaches nearly 90 percent accuracy (Tai et al., 2015; Kim, 2014; Irsoy and Cardie, 2014), but for other languages there is a marked drop in accuracy. This is mainly due to the lack of annotations and resources in these languages. This is especially true of corpora annotated at aspect-level. Unlike document- or tweet-level annotation, aspect-level annotation requires a large amount of effort from the annotators, which further reduces the likelihood of finding an aspect-level sentiment corpus in under-resourced languages. We are, however, aware of one corpus annotated for aspects in German (Klinger and Cimiano, 2014), although German is not a particularly low-resource language.

The movement towards multi-lingual datasets for sentiment analysis is important because many languages offer different challenges, such as complex morphology or highly productive word formation, which can not be overcome by focusing only on English data.

The novelty of this work lies in creating corpora which cover both Basque and Catalan languages and are annotated in such a way that they are compatible with similarly compiled corpora available in a number of languages (Agerri et al., 2013). This allows for further research into cross-lingual sentiment analysis, as well as introducing the first resource for aspect-level sentiment analysis in Catalan and Basque. The corpus is available at `http://hdl.handle.net/10230/33928` or `https://jbarnesspain.github.io/resources/`.

2. Related Work

In English there are many datasets available for document- and sentence-level sentiment analysis across different domains and at different levels of annotation (Pang et al., 2002; Hu and Liu, 2004; Blitzer et al., 2007; Socher et al., 2013; Nakov et al., 2013). These resources have been built up over a period of more than a decade and are currently necessary to achieve state-of-the-art performance.

Corpora annotated at fine-grained levels (opinion- or aspect-level) require more effort from annotators, but are able to capture information which is not present at document- or sentence-level, such as nested opinions or differing polarities of different aspects of a single entity. In English, the MPQA corpus (Wiebe et al., 2005) has been widely used in fine-grained opinion research. More recently, a number of SemEval tasks have concentrated on aspect-level sentiment analysis (Pontiki et al., 2014; Pontiki et al., 2015; Pontiki et al., 2016).

The Iberian peninsula contains two official languages (Portuguese and Spanish), as well as three co-official languages (Basque, Catalan, and Galician) and several smaller languages (Aragonese, Gascon). The two official languages do have available resources for sentiment at tweet-level (Villena-Román et al., 2013; Arruda et al., 2015), as well as at aspect-level (Agerri et al., 2013; Villena-Román et al., 2015; Almeida et al., 2015). The co-official languages, however, have almost none. The authors are aware of a small discourse-related sentiment corpus available in Basque (Alkorta et al., 2015), as well as a stance corpus in Catalan (Bosco et al., 2016). These resources, however, are limited in size and scope.

3. Data Collection

In order to improve the lack of data in low-resource languages, we introduce two aspect-level sentiment datasets to the community, available for Catalan and Basque. To collect suitable corpora, we crawl hotel reviews from `www.booking.com`. Booking.com allows you to search for reviews in Catalan, but it does not include Basque. Therefore, for Basque we crawled reviews from a number of other

websites that allow users to comment on their stay[1]
Many of the reviews that we found through crawling are either 1) in Spanish, 2) include a mix of Spanish and the target language, or 3) do not contain any sentiment phrases. Therefore, we use a simple language identification method[2] in order to remove any Spanish or mixed reviews and also remove any reviews that are shorter than 7 tokens. This finally gave us a total of 568 reviews in Catalan and 343 reviews in Basque, collected from November 2015 to January 2016.
We preprocess them through a very light normalization, after which we perform tokenization, pos-tagging and lemmatization using Ixa-pipes (Agerri et al., 2014).
Our final documents are in KAF/NAF format (Bosma et al., 2009; Fokkens et al., 2014). This is a stand-off xml format originally from the Kyoto project (Bosma et al., 2009) and allows us to enrich our documents with many layers of linguistic information, such as the pos tag of a word, its lemma, whether it is a polar word, and if so, if it has an opinion holder or target. The advantage of this format is that we do not have to change the original text in any way.

4. Annotation

For annotation, we adopt the approach taken in the OpeNER project (Agerri et al., 2013), where annotators are free to choose both the span and label for any part of the text.

4.1. Guidelines

In the OpeNER annotation scheme[3] (see Table 1 for a short summary), an annotator reads a review and must first decide if there is any positive or negative attitudes in the sentence. If there are, they then decide if the sentence is on topic. Since these reviews are about hotels, we constrain the opinion targets and opinion expressions to those that deal with aspects of the hotel. Annotators should annotate the span of text which refers to:

- **opinion holders**,
- **opinion targets**,
- and **opinion expressions**.

If any opinion expression is found, the annotators must then also determine the polarity of the expression, which can be STRONG NEGATIVE, NEGATIVE, POSITIVE, or STRONG POSITIVE. As the opinion holder and targets are often implicit, we only require that each review has at least one annotated opinion expression.
For the strong positive and strong negative labels, annotators must use clues such as adverbial modifiers ('very bad'), inherently strong adjectives ('horrible'), and any use of capitalization, repetition, or punctuation ('BAAAAD!!!!!') in order to decide between the default polarity and the strong version.

Is there a positive / negative attitude?	yes/no
Is the sentence on topic ?	yes/no
Is it to the point?	yes/no
IF YES TO ALL, ANNOTATE:	
What is the span of the expression?	choose span
Is the expression positive or negative?	choose
Is the expression strong?	choose
Is there an explicit target?	yes/no
If yes, what is the span?	choose span
Is there an explicit opinion holder	yes/no
If yes, what is the span?	choose span

Table 1: Simplified annotation guidelines.

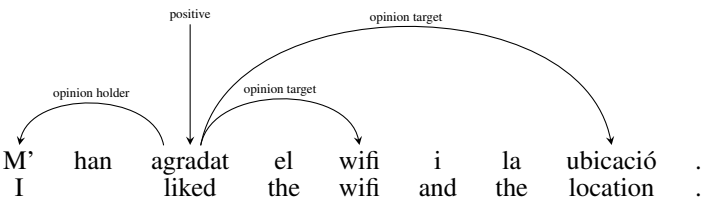


Figure 1: An opinion annotation following the annotation scheme detailed in Section 4.1..

4.2. Process

We used the KafAnnotator Tool (Agerri et al., 2013) to annotate each review. This tool allows the user to select a span of tokens and to annotate them as any of the four labels mentioned in Section 4.1..
The annotation of each corpus was performed in three phases: first, each annotator annotated a small number of reviews (20-50), after which they compared annotations and discussed any differences. Second, the annotators annotated half of the remaining reviews and met again to discuss any new differences. Finally, they annotated the remaining reviews. For cases of conflict after the final iteration, a third annotator decided between the two.
The final Catalan corpus contains 567 annotated reviews and the final Basque corpus 343.

4.3. Dataset Characteristics

The reviews are typical hotel reviews, which often mention various aspects of the hotel or experience and the polarity towards these aspects. An example is shown in Example
Statistics for the two corpora are shown in Table 2.

4.4. Agreement Scores

Common metrics for determining inter-annotator agreement, e.g. Cohen's Kappa (Cohen, 1960) or Fleiss' Kappa (Fleiss, 1971), can not be applied when annotating sequences, as the annotators are free to choose which parts

[1]We took reviews from a total of 35 different websites, including www.airbnb.com, www.atrapalo.com, www.nekatur.net, www.rentalia.es, www.toprural.es, and www.tripadvisor.com.

[2]We use the count of stopwords to predict the probability that a review is written in Spanish, Catalan, or Basque.

[3]http://www.opener-project.eu/

	Catalan	Basque
Number of Reviews	567	343
Average length in tokens	45	46.9
Number of Targets	2762	1775
Number of Expressions	2346	2328
Number of Holders	236	296

Table 2: Corpus Statistics

	Catalan	Basque
Number of Reviews	567	343
Targets	.767	.739
Expressions	.716	.714
Holders	.121	.259
Polarity (MSE)	1.53	2.7

Table 3: Inter-annotator agreement scores. *AvgAgr* score is reported for targets, expressions and holders and averaged mean squared error is reported for polarity.

of a sequence to include. Therefore, we use the *agr* metric (Wiebe et al., 2005), which is defined as:

$$\text{agr}(a||b) = \frac{|A \text{ matching } B|}{|A|} \quad (1)$$

where a and b are annotators and A and B are the set of annotations for each annotator. If we consider a to be the gold standard, agr corresponds to the recall of the system, and precision if b is the gold standard. For each pair of annotations, we report the average of the agr metric with both annotators as the temporary gold standard,

$$\text{AvgAgr}(a, b) = \frac{1}{2}\big[\text{agr}(a||b) + \text{agr}(b||a)\big] \quad (2)$$

Perfect agreement, therefore, is 1.0 and no agreement whatsoever is 0.0. Similar annotation projects (Wiebe et al., 2005) report $AvgAgr$ scores that range between 0.6 and 0.8 in general.

For polarity, we assign integers to each label (Strong Negative: 0, Negative: 1, Positive: 2, Strong Positive: 3). For each sentence of length n, we take the mean squared error (MSE),

$$\text{Mean Squared Error} = \frac{1}{n}\sum_{i=1}^{n}(A - B)^2 \quad (3)$$

where A and B are the sets of annotations for the sentence in question. This approach punishes larger discrepancies in polarity more than small discrepancies, i.e. if annotator 1 decides an opinion expression is STRONG NEGATIVE and annotator two that the same expression is POSITIVE, this will be reflected in a larger MSE score than if annotator 2 had chosen NEGATIVE. Perfect agreement between annotators would lead to a MSE of 0.0, with the maximum depending on the length of the phrase. For a phrase of ten words, the worst MSE possible (assuming annotator 1 labeled all words STRONG POSITIVE and annotator 2 labeled them STRONG NEGATIVE) would be a 9.0. We take the mean of all the MSE scores in the corpus.

Inter-annotator agreement is reported in Table 3.

The inter-annotator agreement for target and expressions is high and in line with previous annotation efforts (Wiebe et al., 2005), given the fact that annotators could choose any span for these labels and were not limited to the number of annotations they could make. This reflects the clarity of the guidelines used to guide the annotation process.

The agreement score for opinion holders is somewhat lower and stems from the fact that there were relatively few instances of explicit opinion holders. Additionally, Catalan and Basque both have agreement features for verbs, which could be considered an implicit mention of the opinion holder. This is not always clear, however. Finally, the mean squared error of the polarity scores shows that annotators generally agree on where and which polarity score should be given. Again, the mean squared error in this annotation scheme requires both annotators to choose the same span and the same polarity to achieve perfect agreement.

5. Difficult Examples

During annotation, there were certain sentences which presented a great deal of problems for the annotators. Many of these are difficult because of 1) **nested opinions**, 2) **implicit opinions reported only through the presence or absence of certain aspects**, or 3) **the difficulty to identify the span of an expression**. Here, we give examples of each difficulty and detail how these were resolved during the annotation process.

(1) Hotela bikaina zen , nahiz eta bertako
Hotel.ABS.SG great.ABS.SG be , although there.from
langileak ez bereziki jatorrak izan.
workers.ABS.PL not particularly friendly.ABS.PL were
'The hotel was great, although the workers there were not particularly friendly.'

In the Basque sentence in Example 1, we can see that there are two distinct levels of aspects. First, the aspect 'hotel', which has a positive polarity and then the sub-aspect 'workers'. We avoid the problem of deciding which is the opinion target by treating these as two separate opinions, whose targets are 'hotel' and 'workers'.

(2) Igerilekua zegoen.
pool.ABS.SG was
'There was a pool.'

If there was an implicit opinion based on the presence or absence of a desirable aspect, such as the one seen in Example 2, we asked annotators to identify the phrase that indicates presence or absence, i.e. 'there was', as the opinion phrase.

(3) Langileek emandako arreta
workers.ERG.PL given.COMP attention.ABS.SG
bikaina zen .
excellent.ABS.SG was
'The attention that the staff gave was excellent.'

Finally, in order to improve overlap in span selection, we instructed annotators to choose the smallest span possible that retains the necessary information. Even after several

iterations, however, there were still discrepancies with difficult examples, such as the one shown in Example 3, where the opinion target could be either 'attention', 'the attention', or 'the attention that the staff gave'.

6. Benchmarks

In order to provide a simple baseline, we frame the extraction of opinion holders, targets, and phrases as a sequence labeling task and map the NAF tags to BIO tags for the opinions in each review. These tags serve as the gold labels which will need to be predicted at test time. We also perform classification of the polarity of opinion expressions.

For the extraction of opinion holders, targets, and expressions we train a Conditional Random Field[4] (CRF) on standard features for supervised sequence labeling (word-, subword-, and part-of-speech information of the current word and previous words). For the classification of the polarity of opinion expressions, we use a Bag-of-Words approach to extract features and then train a linear SVM classifier[5]

For evaluation, we perform a 10-fold cross-validation with 80 percent of the data reserved for training during each fold. For extraction and classification, we report the weighted F_1 score. The results of the benchmark experiment (shown in Table 4) show that these simple baselines achieve results which are somewhat lower but still comparable to similar tasks in English (Irsoy and Cardie, 2014). The drop is not surprising given that we use a relatively simple baseline system and due to the fact that Catalan and Basque have richer morphological systems than English, which were not exploited.

	Catalan	Basque
Targets	.64	.57
Expressions	.52	.54
Holders	.56	.54
Polarity	.80	.84

Table 4: Weighted F_1 scores for extraction of opinion targets, expressions and holders, as well as the weighted F_1 for classification of polarity.

7. Conclusion

In this paper we have presented the MultiBooked corpus – a corpus of hotel reviews annotated for aspect-level sentiment analysis available in Basque and Catalan. The aim of this annotation project is to allow researchers to enable research on supervised aspect-level sentiment analysis in Basque and Catalan, as well as provide useful data for cross- and multi-lingual sentiment analysis. We also provide inter-annotator agreement scores and benchmarks, as well as making the corpus available to the community.

8. Bibliographical References

Agerri, R., Cuadros, M., Gaines, S., and Rigau, G. (2013). OpeNER: Open polarity enhanced named entity recognition. In *Sociedad Española para el Procesamiento del Lenguaje Natural*, volume 51, pages 215–218.

Alkorta, J., Gojenola, K., Iruskieta, M., and Perez, A. (2015). Using relational discourse structure information in basque sentiment analysis. In *Proceedings of the 5th Workshop on RST and Discourse Studies at SEPLN(2015)*, pages 1–10.

Almeida, M. S. C., Pinto, C., Figueira, H., Mendes, P., and Martins, A. F. T. (2015). Aligning opinions: Cross-lingual opinion mining with dependencies. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 408–418.

A.R., B., Joshi, A., and Bhattacharyya, P. (2012). Cross-lingual sentiment analysis for Indian languages using linked wordnets. In *Proceedings of COLING 2012: Posters*, pages 73–82.

Arruda, G. D. d., Roman, N. T., and Monteiro, A. M. (2015). An annotated corpus for sentiment analysis in political news. In *Proceedings of the 2015 Symposium in Information and Human Language Technology*, pages 101–110.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447.

Bosco, C., Lai, M., Patti, V., Pardo, F. M. R., and Rosso, P. (2016). Tweeting in the debate about catalan elections. In *Proceedings of the 2016 LREC workshop on Emotion and Sentiment Analysis Workshop (ESA)*, pages 67–70.

Bosma, W., Vossen, P., Soroa, A., Rigau, G., Tesconi, M., Marchetti, A., Monachini, M., and Aliprandi, C. (2009). KAF: A generic semantic annotation format. In *Proceedings of the Generative Lexicon (GL2009) Workshop on Semantic Annotation*, Pisa, Italy.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurements*, 20:37–46.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382.

Fokkens, A., Soroa, A., Beloki, Z., Ockeloen, N., Rigau, G., Robert van Hage, W., and Vossen, P. (2014). Naf and gaf: Linking linguistic annotations. In *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*.

Gao, D., Wei, F., Li, W., Liu, X., and Zhou, M. (2015). Cross-lingual sentiment lexicon learning with bilingual word graph label propagation. *Computational Linguistics*, 41(1):21–40, March.

Hu, M. and Liu, B. (2004). Mining opinion features in customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, pages 168–177.

Irsoy, O. and Cardie, C. (2014). Deep recursive neural networks for compositionality in language. In *Advances in*

[4]We use the implementation available in *sklearn_crfsuite*.

[5]We use the liblinear implementation from *sklearn*.

Neural Information Processing Systems, volume 3, pages 2096–2104.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Klinger, R. and Cimiano, P. (2014). The USAGE review corpus for fine grained multi lingual opinion analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2211–2218.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Perez-Rosas, V., Banea, C., and Mihalcea, R. (2012). Learning sentiment lexicons in spanish. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 3077–3081.

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35.

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., and Androutsopoulos, I. (2015). Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495.

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S. M., and Eryiğit, G. (2016). Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the EMNLP 2013*, pages 1631–1642.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations From tree-structured long short-term memory networks. In *Association for Computational Linguistics 2015 Conference*, pages 1556–1566.

Villena-Román, J., Lana-Serrano, S., Martínez-Cámera, E., and González-Cristóbal, J. C. (2013). Tass - workshop on sentiment analysis at sepln. *Procesamiento del Lenguaje Natural*, 50(0):37–44.

Villena-Román, J., Martínez-Cámera, E., García Morera, J., and Jiménez Zafra, S. M. (2015). Tass 2014 - the challenge of aspect-based sentiment analysis. *Procesamiento del Lenguaje Natural*, 54(0):61–68.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation (formerly Computers and the Humanities)*, 39(2/3):164–210.

9. Language Resource References

Rodrigo Agerri and Josu Bermudez and German Rigau. (2014). *IXA pipeline: Efficient and Ready to Use Multilingual NLP tools*. European Language Resources Association (ELRA), 1.0.

BlogSet-BR: A Brazilian Portuguese Blog Corpus

[1]Henrique D.P. dos Santos, [2]Vinicius Woloszyn, [1]Renata Vieira
[1]Pontificia Universidade Catolica do Rio Grande do Sul
Av. Ipiranga, 6681, Building 32 - Porto Alegre - Brazil

[2]Instituto de Informática – Universidade Federal do Rio Grande do Sul
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

henrique.santos.003@acad.pucrs.br, vwoloszyn@inf.ufrgs.br, renata.vieira@pucrs.br

Abstract

The rich user-generated content found on internet blogs have always attracted the interest of scientific communities for many different purposes, such as from opinion and sentiment mining, information extraction or topic discovery. Nonetheless, an extensive corpora is essential to perform most of Natural Language Processing involved in these tasks. This paper presents BlogSet-BR, an extensive Brazilian Portuguese corpus containing 2.1 billions words extracted from 7.4 millions posts over 808 thousand different Brazilian blogs. Additionally, a survey was conducted with authors to draw a profile of Brazilian bloggers.

Keywords: Blog as Corpus, BlogSet-BR, Brazilian Portuguese Corpus

1. Introduction

Several efforts have been made to build a large corpora based on user-generated content, since they are crucial for many different Natural Language Processing tasks, such as opinion mining, sentiment analysis, topic detection and Age/Gender detection (Burton et al., 2009; Buck et al., 2014; Santos et al., 2017). From all content available on the internet, blogs have been often employed as a main source of user-generated content (Agarwal and Liu, 2008; Agarwal and Liu, 2009; Santos et al., 2012). Nonetheless, there is still a lack of a large semi-structured corpus that also contains author profiles in Brazilian Portuguese. To illustrate this situation, Table 1[1] gives an overview of the initiatives to create a Portuguese corpora based on general content.

Besides Linguateca resources in Table 1, we added other four corpora: the brWac corpus for Brazilian Portuguese built by downloading text from the web (Boos et al., 2014), Buscapé, a Portuguese product reviews corpus extracted from a collaborative review site (Hartmann et al., 2014), the Portuguese Wikipedia dump and BlogSet-BR corpus. The first three corpora listed in the table contains the authorship of the content.

This paper describes BlogSet-BR, a semi-structured large corpus containing author information extracted from Brazilian Portuguese blogs. It contains more than 7.4 millions posts resulting in 2.1 billion words. It is the first Brazilian Portuguese corpus about blogs. In this paper, we also conducted a survey with the authors to create a profile of Brazilian bloggers. The main contributions of this work are:

- The first semi-structured large corpus of posts from Brazilian Portuguese blogs;
- A corpus with authorship, date-time and label information attached to the text;
- Information retrieved from the author profiles, such as gender, age, and educational level.

This work describes the building process of the BlogSet-BR collection and also descriptive statistics about the data. The rest of this paper is organized as follow: first, Section 2. presents related work; in Section 3., we discuss the three phases of creating the BlogSet-BR collection; section 4. provides an overview of the statistics and content on the collected data; section 5. describes the profile of Brazilian bloggers obtained through a survey conducted with authors; section 6. presents the conclusions and suggestions for further work.

2. Background

Extract content from the web is a constant effort by academic and industry researchers. Such data sets allow the accomplishment of many different tasks, such as relevance ranking for online documents and several other machine learning tasks(Woloszyn et al., 2016; Woloszyn et al., 2017). For instance, the Common Crawl project maintains an open repository of web crawl data that can be accessed and analyzed by any research group[2]. This corpus has been used to build language models (Roziewski and Stokowiec, 2016), and to analyze word frequencies (Buck et al., 2014). However, this project does not contain structured information to allow a social network analyses about the authors. This gap is filled by datasets built with blog content, enabling analyses of web data together with user relation and temporal characteristics. The TREC Conference organizers built a blog corpus for research purposes (Macdonald and Ounis, 2006) making 100,649 blogs in the English language available for TREC shared tasks.

The ICWSM 2009 Spinn3r dataset (Burton et al., 2009) is another example of corpora extracted from blogs. It has more than 44 million blog posts for the English language

[1]Adapted from http://www.linguateca.pt/ACDC/

[2]http://commoncrawl.org/

Corpus	Tokens	Words	Sentences	Avg W/S	Authorship
BlogSet-BR	**2,750,700,677**	**2,146,206,009**	**86,803,291**	**24.72**	**Yes**
ptwiki-20170701	390,200,280	341,609,234	79,182,754	4.31	Yes
Buscapé	4,097,905	3,523,417	180,688	19.50	Yes
brWaC	3,207,918,165	2,764,098,344	145,370,673	19.01	No
Corpus Brasileiro	1,175,568,626	990,061,955	45,457,774	21.97	No
NILC/Sao Carlos	42,912,644	32,461,799	1,988,621	16.32	No
ANCIB	1,707,731	1,257,109	83,509	15.05	No
OBras	1,424,014	1,133,302	37,419	30.28	No
ECI-EBR	922,378	723,995	44,381	16.31	No
ReLi	189,577	145,325	8,752	16.60	No
AmostRA-NILC	128,190	98,633	4,931	20.00	No
C-Oral-Brasil	121,092	65,303	9,329	7.00	No
FrasesPB	23,349	19,162	653	29.34	No

Table 1: Brazilian Portuguese Corpora

and has been used in many research works. The work of (Mishne and others, 2005) collected more than 815.494 posts regarding their authors' mood classification, where 10 posts for each of the 40 most frequent moods were annotated. Quan's work (Quan and Ren, 2009) created a fine-grained annotation scheme of emotions in a Chinese corpus containing 1,487 blogs.

To the best of our knowledge, there is no previous work addressing a Portuguese corpora containing an extensible information about the authors based on internet blogs. The BlogSet-BR collection is the first large corpus with information about the profile of the authors for the Portuguese language. Additionally, the dataset contains information that allows analyzing the social network structure. In the next Section, we present the process of developing BlogSet-BR.

3. Corpus Construction

The corpus construction process was performed in three stages: a) selecting suitable blogs to crawl; b) fetching the appropriate content from the Web; and c) organizing the collection into a reusable form.

Selecting the Blogs: The list of blogs used in this dataset was extracted from Santos' work (Santos, 2013) about the Brazilian blogosphere. From this blog list, we selected only the blogs with the 'BR' assigned in the country field of the blog description. Some blogs in the list were discontinued and it was not possible to collect them. In total, 808,973 blogs were selected for the BlogSet-BR collection.

Collecting the Posts: The posts were collected between December 2016 and September 2017 using the Blogspot API[3], which was divided into two tasks: fetching the description of each blog and fetching the posts. Seven crawlers running in parallel were responsible for gathering the contents of the blogs and posts.

Organizing the Collection: Once all crawled data was collected, we had to reorganize it in a format that was easy to use for research purposes. We aimed here to follow the general layout formats from other collections, as this would allow participating groups to reuse existing tools. In the BlogSet-BR collection, we only grouped the blog posts, but the raw data collected from API, in the JSON format, with the blog descriptions and posts content is also available. The BlogSet-BR contains the following fields: blog ID, post ID, author ID, author name, title, published date, content, tags, and number of comments, in the comma-separated values (CSV) format.

[3]https://developers.google.com/blogger/



Figure 1: Top 50 Tag Cloud (translated)

4. Dataset Description

In this section, we show a descriptive statistics about the data and an overview of the BlogSet-BR collection. From all 808 thousand blogs, 7.4 millions posts were collected. They were written by several authors addressing many different subjects. When compressed, the collection reaches 4.7 GB of data.

In the whole dataset, there are more than 57 millions tags. The most frequent tag is 'news', followed by 'tips', 'love', 'music', 'fashion', and 'movies'. Figure 1 shows the top 50 most frequent tags. These tags show the vast amount of content that can be found in BlogSet-BR for text mining and information extraction research.

4.1. Dates and Times

The date and time associated to each post is important to analyze spread of information in social networks (Lerman and Ghosh, 2010; Adar and Adamic, 2005). Figure 2 shows

the distribution of the dates of the collected posts after the year 2003. The posts before 2003 represent only 0.005% of the corpus.

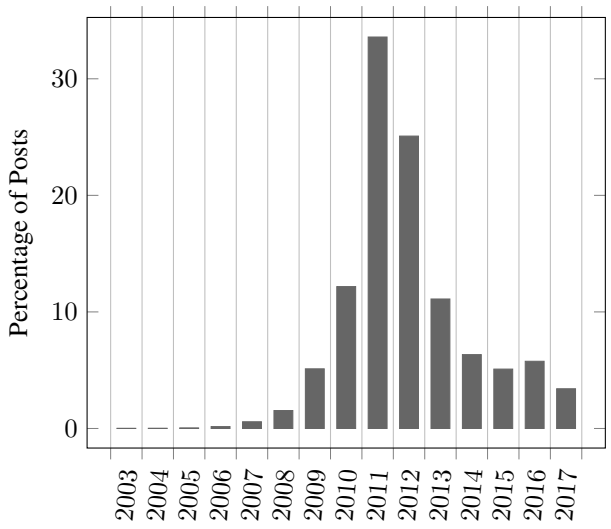


Figure 2: Posts Year Distribution

The posts start decreasing in Blogspot platform after 2011. This behavior could represent the migration of users to Twitter and Facebook, both grow exponentially in Brazil in the same period (Yokoyama and Sekiguchi, 2014).

5. Bloggers Profile

A survey[4] was conducted with 4,332 Brazilian authors from the Blogspot platform to establish a profile of bloggers who use the platform. The requested pieces of information were:

- Age
- Gender
- Topic of interest
- Pageviews
- Update frequency
- Educational level

The majority of authors are male (61%). In addition, almost every user finished high school (92%) and most of them add new a new post at least once a week (85%). Figure 3 shows the age distribution on the collected survey.
Regarding frequency, 32% create new content every day and 53% publish every week. Most users create their own content (86%), giving their own opinion about mundane facts.
Table 2 shows the distribution of topics, which vary: books, fashion, technology, education, politics, movies, and arts. In this question, users were allowed to select multiple choices.
In Figure 4 we show an overview of answers by Brazilian bloggers regarding their educational level. This was not a mandatory question, but 97% of all those who answered

[4]Survey conducted in 2012. Not every author in the survey is in the corpus BlogSet-BR, collected in 2017.

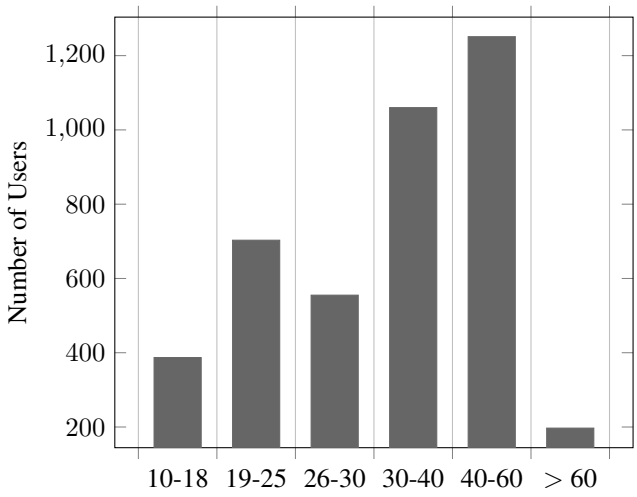


Figure 3: User's Age Distribution

Topic	% of Users
Arts	29%
Education	28%
Literature / Books	28%
Music	27%
Politics	24%
Philosophy	19%
Friendship	17%
Movies	17%
Health	17%
Technology	16%

Table 2: Distribution of the Top 10 Topics selected by users

finished high school and the majority of them are at the university.
Considering social media, most interviewees use Facebook (94%) and Twitter (67%). This rich information about age, gender, and educational level of Brazilian bloggers could be employed in different tasks, such as writing style de-

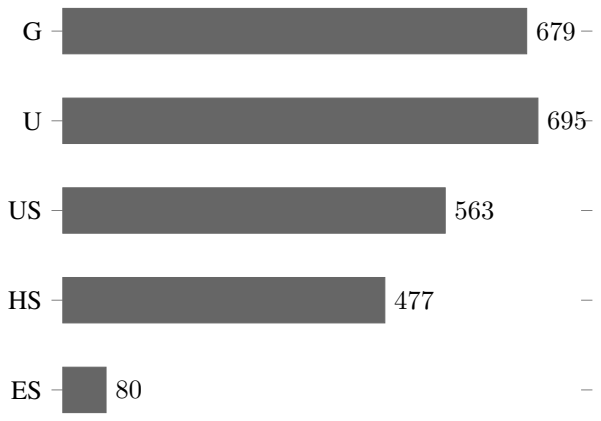


Figure 4: User's Educational Level: G (Graduated), U(Undergraduated), US (Undergraduated Student), HS (High School), ES (Elementary School)

tection and readability assessment (Herring and Paolillo, 2006; Argamon et al., 2007).

6. Conclusion and Further Work

In this paper, we described the motivations, details, and building process of BlogSet-BR. Additionally, we conducted a survey with authors to create their profile, which resulted in a rich description of Brazilian bloggers.
This corpus is the biggest corpus with authorship information for the Brazilian Portuguese language. These types of dataset are useful for topic detection and other NLP tasks.
All the content of this work is available on the web[5]. It is possible to download the raw data in the JSON format, only the Brazilian blog posts in the CSV format, and the survey with 4 thousand users in the XLS format.

Acknowledgments

This work was partially supported by the CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) Foundation (Brazil), PUCRS (Pontifícia Universidade Católica do Rio Grande do Sul), and UFRGS (Universidade Federal do Rio Grande do Sul).

7. Bibliographical References

Adar, E. and Adamic, L. A. (2005). Tracking information epidemics in blogspace. In *Proceedings of the 2005 IEEE/WIC/ACM international conference on web intelligence*, pages 207–214. IEEE Computer Society.

Agarwal, N. and Liu, H. (2008). Blogosphere: research issues, tools, and applications. *ACM SIGKDD Explorations Newsletter*, 10(1):18–31.

Agarwal, N. and Liu, H. (2009). Modeling and data mining in blogosphere. *Synthesis lectures on data mining and knowledge discovery*, 1(1):1–109.

Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. (2007). Mining the blogosphere: Age, gender and the varieties of self-expression. *First Monday*, 12(9).

Buck, C., Heafield, K., and Van Ooyen, B. (2014). N-gram counts and language models from the common crawl. In *LREC*, volume 2, page 4.

Burton, K., Java, A., and Soboroff, I. (2009). The icwsm 2009 spinn3r dataset. In *Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*. AAAI.

Herring, S. C. and Paolillo, J. C. (2006). Gender and genre variation in weblogs. *Journal of Sociolinguistics*, 10(4):439–459.

Lerman, K. and Ghosh, R. (2010). Information contagion: An empirical study of the spread of news on digg and twitter social networks. *ICWSM*, 10:90–97.

Macdonald, C. and Ounis, I. (2006). The trec blogs06 collection: Creating and analysing a blog test collection. *Department of Computer Science, University of Glasgow Tech Report TR-2006-224*, 1:3–1.

Mishne, G. et al. (2005). Experiments with mood classification in blog posts. In *Proceedings of ACM SIGIR 2005 workshop on stylistic analysis of text for information access*, volume 19, pages 321–327. Citeseer.

Quan, C. and Ren, F. (2009). Construction of a blog emotion corpus for chinese emotional expression analysis. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1446–1454. Association for Computational Linguistics.

Roziewski, S. and Stokowiec, W. (2016). Languagecrawl: A generic tool for building language models upon common-crawl. In *LREC*.

Santos, R. L., Macdonald, C., McCreadie, R., Ounis, I., Soboroff, I., et al. (2012). Information retrieval on the blogosphere. *Foundations and Trends® in Information Retrieval*, 6(1):1–125.

Santos, H. D. P., Woloszyn, V., and Vieira, R. (2017). Portuguese personal story analysis and detection in blogs. In *Proceedings of the International Conference on Web Intelligence*, pages 709–715. ACM.

Santos, H. D. P. (2013). Identificação de autoridades em tópicos na blogosfera brasileira usando comentários como relacionamento. Master's thesis, Federal University of Rio Grande do Sul.

Woloszyn, V., Santos, H. D. P., and Wives, L. K. (2016). The influence of readability aspects on the user's perception of helpfulness of online reviews. *Revista de Sistemas de Informação da FSMA*.

Woloszyn, V., Santos, H. D. P., Wives, L. K., and Becker, K. (2017). Mrr: an unsupervised algorithm to rank reviews by relevance. In *Proceedings of the International Conference on Web Intelligence*, pages 877–883. ACM.

Yokoyama, M. H. and Sekiguchi, T. (2014). The use of social network sites in the workplace: A case study in brazilian companies. *Brazilian Business Review*, 11(2):87.

8. Language Resource References

Boos, R., Prestes, K., Villavicencio, A., and Padró, M. (2014). brwac: a wacky corpus for brazilian portuguese. In *International Conference on Computational Processing of the Portuguese Language*, pages 201–206. Springer.

Hartmann, N., Avanço, L., Balage Filho, P. P., Duran, M. S., Nunes, M. d. G. V., Pardo, T. A. S., Aluísio, S. M., et al. (2014). A large corpus of product reviews in portuguese: Tackling out-of-vocabulary words. In *LREC*, pages 3865–3871.

[5] http://www.inf.pucrs.br/linatural/blogset-br

SoMeWeTa: A Part-of-Speech Tagger for German Social Media and Web Texts

Thomas Proisl
Friedrich-Alexander-Universität Erlangen-Nürnberg
Professur für Korpuslinguistik
Bismarckstr. 6, 91054 Erlangen, Germany
thomas.proisl@fau.de

Abstract

Off-the-shelf part-of-speech taggers typically perform relatively poorly on web and social media texts since those domains are quite different from the newspaper articles on which most tagger models are trained. In this paper, we describe SoMeWeTa, a part-of-speech tagger based on the averaged structured perceptron that is capable of domain adaptation and that can use various external resources. We train the tagger on the German web and social media data of the EmpiriST 2015 shared task. Using the TIGER corpus as background data and adding external information about word classes and Brown clusters, we substantially improve on the state of the art for both the web and the social media data sets. The tagger is available as free software.

Keywords: part-of-speech tagging, domain adaptation, evaluation

1. Introduction

Part-of-speech tagging is a core task in natural language processing (NLP) that is important for many subsequent processing steps, e. g. lemmatization, parsing, named-entity recognition or machine translation.

There is already a large number of part-of-speech taggers available that provide pre-trained models for many different languages, including German. Those models are almost always trained on corpora of edited texts written by professional writers, usually newspaper articles. This is also true for German models that are usually trained on the TIGER corpus (Brants et al., 2004). Unfortunately, models trained on newspaper articles perform relatively poorly on web and social media data (Giesbrecht and Evert, 2009).

This is largely due to the many unconventional spelling variants that occur in web and social media texts and that result in a high proportion of out-of-vocabulary (OOV) words on which most taggers perform much worse than on in-vocabulary words. In addition, there are several phenomena in web and social media texts that usually do not occur in edited texts and that cannot be captured properly by most part-of-speech tagsets, including STTS (Schiller et al., 1999). Those phenomena include emoticons, interaction words (e. g. **lach**), hash tags, addressing terms, URLs, onomatopoeia, and colloquial contractions (e. g. *machste*).

One of the aims of the EmpiriST 2015 shared task (Beißwenger et al., 2016)[1] was to improve tokenization and part-of-speech tagging of German computer-mediated communication (CMC) and web corpora. To this end, a gold standard of more than 22,000 tokens (the precise numbers are summarized in Table 1) was annotated with an extended version of STTS called STTS_IBK (Beißwenger et al., 2015). STTS_IBK introduces 18 additional tags, mainly for phenomena typically found in CMC data.

In the present paper, we describe SoMeWeTa (short for Social Media and Web Tagger),[2] a freely available part-of-speech tagger that achieves the best results on the EmpiriST 2015 data published so far.

	CMC	Web
Training	5,109	4,944
Test	5,234	7,568
Total	10,343	12,512

Table 1: Sizes of the EmpiriST training and test sets in tokens.

2. Related Work

Four teams participated in the EmpiriST 2015 shared task. Prange et al. (2016), who won the shared task, use the HMM-based HunPos tagger (Halácsy et al., 2007) and focus on improving the accuracy on out-of-vocabulary words. Their best-performing system is trained on a combination of the TIGER corpus, a version of the TIGER corpus automatically converted to new orthography,[3] the same-domain EmpiriST data set boosted by adding it five times and 34,000 tokens of additional in-domain training data (forum texts, chat and twitter data). Additionally, they use distributional methods to induce a POS lexicon for OOV words which is provided to the tagger at runtime.

Horsmann and Zesch (2016) use the FlexTag tagger (Zesch and Horsmann, 2016) with a conditional random fields (CRF) classifier. Their best-performing system is trained on a combination of the EmpiriST data and 100,000 tokens from the TIGER corpus and uses Brown clusters (Brown et al., 1992), morphological features extracted from Morphy (Lezius, 2000), a POS lexicon extracted from a treebank and lists of named entities as additional resources. The system performs heuristic post-processing steps for hash tags, mentions, URLs, email addresses, the word *sehr* and words ending in hyphens.

[1] https://sites.google.com/site/empirist2015/

[2] https://github.com/tsproisl/SoMeWeTa

[3] This makes Prange et al. (2016) the only team that directly addresses the effects of the German spelling reform.

Remus et al. (2016) build on the GermaNER named entity recognizer (Benikova et al., 2015) to create their CRF sequence tagger. For training their system, they convert the TIGER corpus to the new STTS_IBK tagset and combine it with the same-domain EmpiriST data set. They make use of lists of named entities, similar words from a distributional thesaurus and LDA topic clusters. The system performs some post-processing steps, e. g. for emoticons.

Stemle (2016) builds a tagger based on a long short-term memory (LSTM) recurrent neural network (RNN). The tagger uses word2vec word embeddings and character n-grams of word beginnings and endings and is trained on a combination of the TIGER corpus and the EmpiriST data.

In Table 3, we compare the results of those four teams with our own system which we describe in the following section.

3. System Description

3.1. Learning Algorithm

SoMeWeTa is based on the averaged structured perceptron and uses beam search and an early update strategy. The perceptron, a supervised linear classifier, was introduced by Rosenblatt (1958). It is typically trained by iterating over the training data several times and adjusting the weight vector whenever a misclassification occurs. Freund and Schapire (1999) find that there are two variants that outperform the final weight vector in classification: Implementing a weighted majority voting system based on all states of the weight vector or using the averaged weight vector. We implement averaging. Collins (2002) introduces the structured perceptron that combines the perceptron with a decoding algorithm such as the Viterbi algorithm or beam search to predict, for example, tag sequences for entire sentences. Collins and Roark (2004) suggest the early update strategy for training the structured perceptron, i. e. to stop processing a sentence and to update the weight vector early once it is impossible for the gold sequence of tags to be in the final set of analyses, e. g. because the search beam is too narrow.

3.2. Domain Adaptation

The EmpiriST training set with its 10,000 tokens is too small for training a competitive part-of-speech tagger (cf. Table 3). Therefore, we need to combine it with additional training data, i. e. the TIGER corpus. The TIGER corpus with its 900,000 tokens is much larger. However, it contains texts from a different domain (newspaper articles) and is annotated with its own variant of STTS instead of STTS_IBK. This means that if we simply merge the two data sets, then TIGER will dominate the resulting training corpus and it is rather unlikely that the tagger learns much about web and social media texts and about the novel tags.

A common strategy for dealing with that problem, that is implemented for example by Prange et al. (2016), is boosting, i. e. giving the EmpiriST training set more weight by adding it several times to the training corpus.

We implement a different strategy for domain adaptation. Chelba and Acero (2004) suggest to train a model on the background data, i. e. TIGER, and to use that model as a prior on the weights of the model trained on the in-domain data, i. e. EmpiriST. As Daumé III (2007) points out, that can be achieved for the perceptron by adding the prior weights to the weight vector when making predictions.

{W, N1, N2}.word	W.logfreq
W.prefix	W.lex
{P1, W, N1}.suffix	P2.word + P2.pos
W.shape	P1.word + P1.pos
W.loglength	{P2, P1}.pos
{P2, P1, W, N1, N2}.flags	P2.pos + P1.pos
{P2, P1, W, N1, N2}.brown	P1.pos + W.word

Table 2: Feature templates. W refers to the current token, P2 and P1 to the previous two tokens, N1 and N2 to the next two tokens.

3.3. Feature Templates

We forgo meticulous feature engineering and use a fairly standard set of features that is summarized in Table 2. The word, prefix and suffix features use the lower cased version of the token. For the shape feature, we map all characters to a character type: Upper case letters to "X", lower case letters to "x" and digits to "d"; all other characters remain the same. We limit the number of consecutive characters of the same type to 4 (this means that "Computer" becomes "Xxxxx"). The loglength feature is the logarithm of the token's length in characters, rounded to an integer. The flag features indicate:

- If all characters of the token are alphabetic, numeric or punctuation;
- if the token is in lower, upper or title case;
- if the word is an email address, an XML tag, a URL, a mention (*@peter*), a hashtag, an interaction word (**grins**), an emoticon, an ordinal number or a number.

The brown feature contains the Brown cluster of the word; the logfreq feature is the logarithm of the frequency of the word according to the Brown cluster file, rounded to an integer. Lex is used for features found in the additional lexicon – in our case major word classes from Morphy and/or capitalization information.

3.4. External Resources

In addition to the EmpiriST training data, we use the following external resources:

- The TIGER corpus is used as background data in the domain adaptation process described above.
- We use DECOW14 (Schäfer, 2015; Schäfer and Bildhauer, 2012) to extract capitalization features, i. e. flags that indicate how a word is capitalized in the majority of its occurrences, and 1,000 Brown clusters (Brown et al., 1992).[4]
- We use information about major word classes from Morphy (Lezius, 2000).[5]

[4]We use the implementation by Liang (2005): `https://github.com/percyliang/brown-cluster/`.

[5]Extracted following these instructions: `http://www.danielnaber.de/morphologie/`.

4. Results and Error Analysis

4.1. Data Preprocessing

The part-of-speech annotation in TIGER differs in some details from the original STTS. It does not distinguish between indefinite pronouns in attributive function with and without determiner (STTS: PIDAT vs. PIAT; TIGER: PIAT), tags prepositions as ADV when they modify numerals and uses the tag PROAV for pronominal adverbs instead of PAV (Smith, 2003, 13). STTS_IBK, however, is an extension of the original STTS and not of the TIGER variant. Therefore, we automatically replace all instances of the tag PROAV with the original STTS tag PAV. We do not address TIGER's other two deviations from STTS.

While TIGER is annotated with sentence boundaries, the EmpiriST data set is not. Therefore, we automatically introduce sentence boundaries using the sentence splitter that is part of the SoMaJo tokenizer for German web and social media texts (Proisl and Uhrig, 2016).[6]

In the EmpiriST data set, emojis have been replaced with textual representations, e. g. emojiQloudlyCryingFace. Since we want our tagger model to be applicable to real-world data, we reintroduce the actual Unicode characters.

4.2. Results

We run evaluation experiments for different combinations of external resources. In all evaluation settings, we train the tagger for ten iterations. Since SoMeWeTa shuffles the training data after each iteration, the training process is subject to some amount of random variation. To account for that and to give a realistic representation of the tagger's performance, we run all evaluation experiments ten times and report the mean of the ten runs $\pm$ two standard deviations. We follow Beißwenger et al. (2016) and report the tagging accuracies on the two EmpiriST data sets (CMC and web data) as well as their macro-average for each evaluation setting (Table 3). We also include figures for in-vocabulary and out-of-vocabulary accuracies.

As points of reference, we include results for scenarios where the tagger is trained only on TIGER or only on the EmpiriST data. In all other evaluation settings, TIGER is used as background data in the domain adaptation process described above.

As we can see, simply using the domain adaptation strategy described above without any additional external resources already yields competitive results (89.11% overall acc.). Both the word class information from Morphy and the Brown clusters extracted from DECOW14 prove to be useful individually and their combination boosts the accuracy even further (91.06% overall), substantially improving on the previous state of the art (Prange et al., 2016) on both subsets. The impact of the capitalization features extracted from DECOW14, on the other hand, seems to be rather limited or even counterproductive in some cases.

As an alternative way for arriving at a single accuracy figure for the data, we combine the CMC and Web test sets into a single test set. On this unified test set, the system by Prange et al. (2016) achieves 91.01% accuracy. SoMeWeTa with the TEB+Mor setting achieves a mean accuracy of 91.55% $\pm$0.18 – a statistically significant improvement (McNemar's test, $p < 0.05$).

We also run experiments where we only use the same-domain EmpiriST data for training. Interestingly, this only leads to better results for the CMC data but not for the web data. It seems that for the web data, which is closer in nature to the newspaper articles in TIGER than the CMC data, the advantage of having a larger corpus for adapting to the novelties of the tagset outweighs the heterogeneity of the data. For the CMC data, on the other hand, being able to adjust to the very different domain by means of a more homogeneous corpus seems to be more important than having more data on the new tags.

4.3. Error Analysis

For the error analysis, we use the run of the TEB+Mor model that is closest to the means reported in Table 3.

Surprisingly, the largest source of errors in the CMC part is the confusion of the two tags $(and $. (cf. Table 4). The reason for this is a peculiarity in the gold data with regard to the colon (:). In the TIGER corpus and in the web part of the EmpiriST data, almost all colons are annotated as $. (97.1% in TIGER and 100% in the web part). In the CMC part, however, only 31.2% of all colons are annotated as $., whereas 68.8% are annotated as $(.

Another interesting source of errors is the frequent misclassification of graphical emoticons (EMOIMG). As it turns out, these errors are mainly due to the fact that there are several sequences of multiple emoticons in the test data, but not in the training data where emoticons always occur in isolation. Other major sources of errors in the CMC and web parts (Table 5) are the confusion of NN and NE, misclassifications related to the new particle tags PTKIFG, PTKMA and PTKMWL and misclassifications within the verb tags.

A visual overview of frequent classes of errors is given in Figure 1. For this visualization, we map the tags to the coarser tagset used in the Universal Dependencies project (Nivre et al., 2016). Note that the new particle tags PTKIFG, PTKMA and PTKMWL get mapped to ADV. We can clearly see that the four major sources of errors are misclassifications within the ADV and VERB tags and the confusion of NOUN and PROPN.[7]

4.4. Post-analysis experiments

As noted in the previous section, there are no sequences of graphical emoticons in the training data. As a consequence, sequences of emoticons are frequently misclassified. Other problematic phenomena we found when using the tagger to annotate a corpus of tweets were sequences of hashtags and mentions that were embedded in the syntactic structure of the sentence.

As a countermeasure to these sources of errors, we created a small file with additional training data taken from a collection of tweets about the German federal election.

[6] https://github.com/tsproisl/SoMaJo

[7] When we map the gold standard and the tagger output to this reduced tagset we get rid of within-class misclassifications and trade tagset granularity for higher accuracy. With its output mapped to UD POS, SoMeWeTa achieves the following accuracies with the TEB+Mor model: 92.69% $\pm$0.19 on CMC, 95.81% $\pm$0.20 on Web, 94.25% overall and 94.54% $\pm$0.12 on the unified test set.

System	CMC all	CMC IV	CMC OOV	Web all	Web IV	Web OOV	Overall
Prange et al. (2016)	87.33	–	–	93.55	–	–	90.44
Horsmann and Zesch (2016)	86.07	–	–	92.10	–	–	89.09
Remus et al. (2016)	84.22	–	–	93.27	–	–	88.75
Stemle (2016)	85.42	–	–	90.63	–	–	88.03
only TIGER	72.24 ±0.31	83.96 ±0.36	33.42 ±0.52	92.03 ±0.21	94.58 ±0.19	77.48 ±1.11	82.14
only EmpiriST	79.72 ±0.40	86.80 ±0.53	62.09 ±1.23	83.58 ±0.64	89.71 ±0.63	73.49 ±0.95	81.65
TIGER+EmpiriST (TE)	85.78 ±0.40	89.07 ±0.45	65.44 ±1.00	92.43 ±0.27	94.40 ±0.15	79.46 ±1.21	89.11
TE+Mor	87.50 ±0.39	90.17 ±0.42	70.96 ±0.88	93.17 ±0.28	94.84 ±0.21	82.19 ±1.27	90.34
TE+Brown (TEB)	88.16 ±0.41*	90.26 ±0.31	75.15 ±1.49	93.72 ±0.17	95.26 ±0.21	**83.58** ±0.91	90.94
TEB+cap	88.20 ±0.29	90.22 ±0.26	75.73 ±1.31	93.58 ±0.18	95.11 ±0.11	83.53 ±0.85	90.89
TEB+Mor	88.37 ±0.26*	90.36 ±0.22	76.06 ±1.27	**93.75** ±0.24	**95.31** ±0.24	83.54 ±0.95	**91.06**
TEB+Mor+cap	88.32 ±0.42*	90.41 ±0.34	75.36 ±1.33	93.73 ±0.18	95.29 ±0.20	83.50 ±0.69	91.03
$TE_{CMC}B$+Mor	88.61 ±0.45**	90.67 ±0.36	**76.08** ±1.53	92.97 ±0.27	94.59 ±0.29	82.97 ±0.91	90.79
$TE_{Web}B$+Mor	76.26 ±0.58	85.47 ±0.74	43.95 ±1.02	93.68 ±0.30	95.29 ±0.25	83.36 ±1.05	84.97
$TE_{CMC}B$+Mor+cap	**88.69** ±0.44**	**90.88** ±0.40	75.33 ±1.61	93.04 ±0.18	94.60 ±0.17	83.36 ±0.44	90.87
$TE_{Web}B$+Mor+cap	76.36 ±0.32	85.46 ±0.27	44.44 ±1.61	93.57 ±0.26	95.27 ±0.25	82.68 ±0.90	84.97

Table 3: Evaluation results. We report the average accuracy of ten runs ± two standard deviations. Abbreviations: TE = TIGER+EmpiriST, TEB = TIGER+EmpiriST+Brown, Mor = word class information from Morphy, cap = capitalization features from DECOW14. Stars indicate a statistically significant improvement on Prange et al. (2016) (McNemar's test, $p < 0.05, 0.01, 0.001$).

tag	freq	err	most frequent confusions
$(	343	53	$. (42), KON (7), XY (3)
NN	696	47	NE (12), FM (7), ADJA (6)
ADV	268	46	PTKMA (14), PIS (6), PTKIFG (6)
PTKIFG	72	46	ADV (32), ADJD (8), PIS (3)
NE	230	40	NN (20), ADJA (4), ADJD (3), ITJ (2)
EMOIMG	63	34	$((21), XY (6), $. (5),
ADJD	187	32	ADV (9), VVPP (7), ADJA (3), NN (3)
ITJ	99	21	NN (5), PTKANT (3), ADJA (2)
PTKMA	74	21	ADV (15), ADJD (3)
VVFIN	183	20	VVINF (8), NN (5), VVIMP (4)
ADJA	149	18	NN (6), NE (4), FM (2), PIAT (2)
VVIMP	20	16	VVFIN (5), NN (4), ART (2), ITJ (2)
XY	14	12	PPER (5), $((2), ITJ (2)
PTKVZ	40	11	APPR (3), ADV (2), PTKIFG (2)
VVINF	87	11	VVFIN (5), NN (3), VVPP (3)
KOKOM	21	10	APPR (5), FM (2), PWAV (2)
KON	112	10	ADV (5), VAFIN (2)

Table 4: Errors in CMC (10 or more errors per gold tag)

tag	freq	err	most frequent confusions
NN	1661	96	NE (74), FM (9), ADJA (3), ADV (3)
NE	252	43	NN (39)
ADV	309	42	PTKIFG (13), PTKMA (12), PIS (4)
VVFIN	250	37	VVINF (17), VVPP (16), NN (2)
PTKIFG	61	36	ADV (28), ADJD (6)
FM	43	24	NE (14), NN (2), VAFIN (2)
ADJA	498	16	NN (4), FM (3), ADJD (2)
PTKMWL	14	14	ADV (9), ADJD (2)
APPR	583	13	KOKOM (6), ADV (4)
ART	729	12	PRELS (7), PDS (5)
PIAT	79	10	PIS (6), ADJA (2)
VVINF	125	10	VVFIN (6), NN (4)
VVPP	125	10	ADJD (7)

Table 5: Errors in Web (10 or more errors per gold tag)

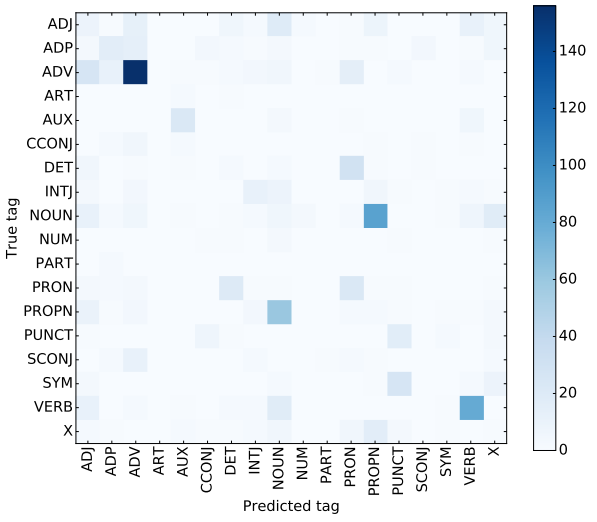


Figure 1: Errors made on the unified test set. For clarity of presentation, we map the tags to UD UPOS tags, i. e. errors on the diagonal indicate misclassifications between tags that get mapped to the same UD UPOS tag.

This file contains 126 tokens in 13 "sentences". The additional training data file covers phenomena typically found in computer-mediated communication. Therefore, its addition leads to improvements for the CMC subset while not affecting the Web subset. The best model achieves an overall accuracy of 91.42% (cf. Table 6).

When we use the TEB+Mor+add model on the unified test set mentioned in Section 4.2., the tagger achieves a mean accuracy of 91.84% ±0.17. This is a statistically significant improvement on Prange et al. (2016) (McNemar's test, $p < 0.001$).

System	CMC all	CMC IV	CMC OOV	Web all	Web IV	Web OOV	Overall
TEB+Mor+add	89.08 ±0.25***	90.95 ±0.27	77.41 ±1.14	**93.75** ±0.26	**95.34** ±0.25	83.31 ±0.63	**91.42**
TEB+Mor+cap+add	88.84 ±0.29***	90.84 ±0.27	76.39 ±1.08	93.71 ±0.29	95.29 ±0.29	83.34 ±0.67	91.28
TE_{CMC}B+Mor+add	**89.17** ±0.38***	91.08 ±0.35	**77.48** ±1.47	93.10 ±0.24	94.62 ±0.19	83.69 ±0.89	91.14
TE_{Web}B+Mor+add	77.82 ±0.59	85.56 ±0.27	50.12 ±2.57	93.67 ±0.19	95.25 ±0.18	83.49 ±0.93	85.75
TE_{CMC}B+Mor+cap+add	89.10 ±0.40***	**91.18** ±0.38	76.36 ±0.94	93.09 ±0.24	94.58 ±0.22	**83.85** ±0.83	91.10
TE_{Web}B+Mor+cap+add	78.16 ±0.35	85.71 ±0.40	51.18 ±2.37	93.62 ±0.32	95.27 ±0.26	82.97 ±1.09	85.89

Table 6: Evaluation results with additional training data (+add). We report the average accuracy of ten runs ± two standard deviations. Abbreviations as in Table 3. Stars indicate a statistically significant improvement on Prange et al. (2016) (McNemar's test, $p < 0.05, 0.01, 0.001$).

5. Conclusion

SoMeWeTa is based on a relatively simple linear classifier in combination with a suitable strategy for domain adaptation. Valuable additional resources are a lexicon with word class information and Brown clusters extracted from a web corpus. Although we do not fine-tune the features or apply post-processing steps that address common errors, SoMeWeTa substantially improves on the current state of the art. Given that other systems, e. g. Horsmann and Zesch (2016), use very similar external resources, the key to SoMeWeTa's superior performance seems to be the more sophisticated domain adaptation process.

Both the tagger and a model trained on the entire EmpiriST data set are freely available.[8]

6. Bibliographical References

Beißwenger, M., Bartz, T., Storrer, A., and Westpfahl, S. (2015). Tagset und Richtlinie für das Part-of-Speech-Tagging von Sprachdaten aus Genres internetbasierter Kommunikation. Guideline document.

Beißwenger, M., Bartsch, S., Evert, S., and Würzner, K.-M. (2016). EmpiriST 2015: A shared task on the automatic linguistic annotation of computer-mediated communication and web corpora. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 44–56, Berlin. Association for Computational Linguistics.

Benikova, D., Yimam, S. M., Santhanam, P., and Biemann, C. (2015). GermaNER: Free open German named entity recognition tool. In Bernhard Fisseni, et al., editors, *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology (GSCL 2015)*, pages 31–38, Duisburg-Essen. GSCL.

Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., König, E., Lezius, W., Rohrer, C., Smith, G., and Uszkoreit, H. (2004). TIGER: Linguistic interpretation of a German corpus. *Research on Language and Computation*, 2(4):597–620.

Brown, P. F., Pietra, V. J. D., de Souza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Chelba, C. and Acero, A. (2004). Adaptation of maximum entropy capitalizer: Little data can help a lot. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 285–292, Barcelona. Association for Computational Linguistics.

Collins, M. and Roark, B. (2004). Incremental parsing with the perceptron algorithm. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, pages 111–118, Barcelona. Association for Computational Linguistics.

Collins, M. (2002). Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 1–8, Philadelphia, PA. Association for Computational Linguistics.

Daumé III, H. (2007). Frustratingly easy domain adaptation. In Annie Zaenen et al., editors, *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 256–263, Prague. Association of Computational Linguistics.

Freund, Y. and Schapire, R. E. (1999). Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3):277–296.

Giesbrecht, E. and Evert, S. (2009). Is part-of-speech tagging a solved task? An evaluation of pos taggers for the German web as corpus. In Iñaki Alegria, et al., editors, *Proceedings of the Fifth Web as Corpus Workshop (WAC5)*, pages 27–35, San Sebastián. Elhuyar Fundazioa.

Halácsy, P., Kornai, A., and Oravecz, C. (2007). Hunpos – an open source trigram tagger. In John A. Carroll, et al., editors, *Proceedings of the ACL 2007 Demo and Poster Sessions*, pages 209–212, Prague. Association for Computational Linguistics.

Horsmann, T. and Zesch, T. (2016). LTL-UDE @ EmpiriST 2015: Tokenization and pos tagging of social media text. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 120–126, Berlin. Association for Computational Linguistics.

Lezius, W. (2000). Morphy – German morphology, part-of-speech tagging and applications. In Ulrich Heid, et al., editors, *Proceedings of the 9th EURALEX International Congress*, pages 619–623, Stuttgart. Institut für Maschinelle Sprachverarbeitung.

[8] https://github.com/tsproisl/SoMeWeTa

Liang, P. (2005). Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož. European Language Resources Association.

Prange, J., Horbach, A., and Thater, S. (2016). UdS-(retrain|distributional|surface): Improving pos tagging for oov words in German cmc and web data. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 63–71, Berlin. Association for Computational Linguistics.

Proisl, T. and Uhrig, P. (2016). SoMaJo: State-of-the-art tokenization for German web and social media texts. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 57–62, Berlin. Association for Computational Linguistics.

Remus, S., Hintz, G., Biemann, C., Meyer, C. M., Benikova, D., Eckle-Kohler, J., Mieskes, M., and Arnold, T. (2016). EmpiriST: AIPHES – Robust tokenization and pos-tagging for different genres. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 106–114, Berlin. Association for Computational Linguistics.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). Technical report, IMS Stuttgart, SfS Tübingen.

Schäfer, R. and Bildhauer, F. (2012). Building large corpora from the web using a new efficient tool chain. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, pages 486–493, Istanbul. European Language Resources Association.

Schäfer, R. (2015). Processing and querying large web corpora with the COW14 architecture. In Piotr Bański, et al., editors, *Proceedings of Challenges in the Management of Large Corpora 3 (CMLC-3)*, pages 28–34, Lancaster. UCREL, IDS.

Smith, G. (2003). A brief introduction to the TIGER treebank, version 1. Technical report, Universität Potsdam.

Stemle, E. (2016). bot.zen @ EmpiriST 2015 – A minimally-deep learning pos-tagger (trained for German cmc and web data). In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 115–119, Berlin. Association for Computational Linguistics.

Zesch, T. and Horsmann, T. (2016). FlexTag: A highly flexible pos tagging framework. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4259–4263, Portorož. European Language Resources Association.

7. Language Resource References

Michael Beißwenger and Sabine Bartsch and Stefan Evert and Kay-Michael Würzner. (2016). *EmpiriST 2015 Gold Standard*. `https://sites.google.com/site/empirist2015/home/gold/`.

Felix Bildhauer and Roland Schäfer. (2014). *DECOW14*. COW – Corpora from the Web, `http://corporafromtheweb.org/decow14/`.

Sabine Brants and Stefanie Dipper and Peter Eisenberg and Silvia Hansen and Esther König and Wolfgang Lezius and Christian Rohrer and George Smith and Hans Uszkoreit. (2004). *TIGER Corpus*. IMS Stuttgart, `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.html`.

Percy Liang. (2012). *Brown-cluster*. `https://github.com/percyliang/brown-cluster`.

Thomas Proisl and Peter Uhrig. (2016). *SoMaJo*. `https://github.com/tsproisl/SoMaJo`.

Thomas Proisl. (2017). *SoMeWeTa*. `https://github.com/tsproisl/SoMeWeTa`.

Collecting Code-Switched Data from Social Media

Gideon Mendels†, Victor Soto†, Aaron Jaech, Julia Hirschberg
Columbia University, New York, NY 10027
University of Washington, Seattle, WA 98105
g.mendels@columbia.edu, vsoto@cs.columbia.edu, ajaech@uw.edu, julia@cs.columbia.edu

Abstract

We address the problem of mining code-switched data from the web, where code-switching is defined as the tendency of bilinguals to switch between their multiple languages both across and within utterances. We propose a method that identifies data as code-switched in languages L_1 and L_2 when a language classifier labels the document as language L_1 but the document also contains words that can only belong to L_2. We apply our method to Twitter data and collect a set of more than 43,000 tweets. We obtain language identifiers for a subset of 8,000 tweets using crowd-sourcing with high inter-annotator agreement and accuracy. We validate our Twitter corpus by comparing it to the Spanish-English corpus of code-switched tweets collected for the EMNLP 2016 Shared Task for Language Identification, in terms of code-switching rates, language composition and amount of code-switch types found in both datasets. We then trained language taggers on both corpora and show that a tagger trained on the EMNLP corpus exhibits a considerable drop in accuracy when tested on the new corpus and a tagger trained on our new corpus achieves very high accuracy when tested on both corpora.

Keywords: code-switching, data collection, social media

1. Introduction

Linguistic code-switching is the phenomenon by which bilingual speakers switch back and forth between languages during communication. Code-switching can be classified as inter-sentential when the switch occurs between the boundaries of a sentence or utterance, or intra-sentential when it occurs within those boundaries. For example a Spanish-English speaker might say "Me dijo que Juanito is very good at math," which represents an intra-sentential switch, or "Me dijo que Juanito mintió. I don't believe it!"

Code-switching can be observed at various linguistic levels of representation for different language pairs: phonological, morphological, lexical, syntactic, semantic, and discourse/pragmatic switching. However, very little code-switching corpora exist from which researchers can train statistical models. The question of how to acquire code-switched data from web and social media resources automatically and accurately remains largely unaddressed.

In this paper, we present a method to automatically collect code-switched data from Twitter. Twitter data has been mined extensively for many Natural Language Processing and speech tasks (Mendels et al., 2015; Go et al., 2009) as one of the only major platforms that provides an API for data collection. The proposed method, which we term "anchoring" can also be used for collecting data from other sources.

The remainder of the paper is organized as follows. In Section 2. we give an overview of previous work on the topic of finding and collecting code-switched data. In Section 3. we present our anchoring method for retrieving code-switched tweets. Section 4. provides the details of our Twitter collection pipeline. Section 5. describes the language identification (LID) task we used to crowdsource the word language tags for the data collected. In Section 6.1., we compare the corpus we acquired using this method with a corpus of tweets that was collected for the EMNLP 2016 Shared Task for Language Identification in code-switched (CS) Data. We compare them in terms of the amount of bilingualism they contain and their code-switching rate – i.e., how frequently writers switch their language in the corpus. In Section 6.2. we train and test language ID taggers on our corpus and the Workshop corpus and compare their performance. We present our conclusions in Section 7.

2. Previous Work

In the past few years there have been increasing efforts on a variety of tasks using code-switched data, including part-of-speech tagging (Solorio and Liu, 2008b; Vyas et al., 2014; Jamatia et al., 2015; AlGhamdi et al., 2016), parsing (Goyal et al., 2003), language modeling (Franco and Solorio, 2007; Li and Fung, 2012; Adel et al., 2013b; Adel et al., 2013a; Li and Fung, 2014), code-switching prediction (Solorio and Liu, 2008a; Elfardy et al., 2014), sentiment analysis (Vilares et al., 2015; Lee and Wang, 2015) and even speech recognition (Ahmed and Tan, 2012; Lyudovyk and Pylypenko, 2014).

The task that has received most of the attention has been Language Identification on code-switched data, thanks in part to the First and Second Shared Tasks on EMNLP 2014 and 2016 (Solorio et al., 2014; Molina et al., 2016). Many of the current state-of-the-art models for Language Identification perform sequence labeling using Conditional Random Fields (Al-Badrashiny and Diab, 2016) or Recurrent Neural Networks (Jaech et al., 2016b). In the 2016 Shared Task the best performing system on the MSA-DA dataset used a combination of both (Samih et al., 2016) on top of word and character-level embeddings, and the best performing system on the ES-EN dataset used logistic regression (Piergallini et al., 2016) and character n-gram features.

On the task of finding and collecting code-switched data from the web, which is the focus of this paper, Çetinoglu (2016) obtained a corpus of German-Turkish tweets by automatically computing dictionaries of pure German and Turkish from a million Turkish, German and English

†The first two authors contributed equally to this work.

tweets. They subsequently used those dictionaries to automatically tag ten million Turkish tweets from which they obtained 8,000 potentially code-switched tweets which they manually filtered down to 680.

Samih (2016) obtained a corpus of forum posts written in MSA and the Darija Dialect following this iterative process: they first started with a list of 439 words exclusive to Darija, they retrieved forum posts that contained one of the exclusive words, and then they added all the words from the retrieved posts to the list of Darija words. They repeated the process until the corpus reached a certain size. Even though their method offers no guarantees, they obtained a corpus with 73.9% of code-switched forum posts.

Barman et al. (2014) used a group of university students as data source to find code-switched media. They found a Facebook group and 11 Facebook users from which they collected 2,335 posts and 9,813 comments. Vyas et al. (2014) collected almost seven thousand comments from 40 manually selected code-switched Facebook posts from three celebrity pages and the BBC Hindi news page. Finally, Jamatia et al. (2015) collected tweets and Facebook posts from a University billboard page, although it is unclear if they specifically targeted code-switched content or not.

The organizers of the EMNLP Shared Tasks on Language Identification in code-switched Data followed a semi-automatic approach. For the first Shared task, code-switched data was collected for the pairs Spanish-English (ES-EN), Mandarin-English (MAN-EN), Nepali-English (NEP-EN) and Modern Standard Arabic-Dialectal Arabic (MSA-DA). The social media sources they targeted were Twitter for all language pairs and Facebook for NEP-EN and blog comments for MSA-DA. For Twitter, their approach consisted in first locating code-switchers and then collecting their posts and posts from their followers and/or followees. For ES-EN, they located a subset of code-switchers by querying the Twitter API with frequent English words, and restricted results to tweets identified as Spanish by Twitter from users based in Texas and California. For NEP-EN, they started from a group of acquaintances that were known to code-switch and then identified their followers and followers of their followers that they found were code-switchers too. For Mandarin-English, they started by looking at the most followed Twitter users in Taiwan. They then added those users that they manually checked were code-switchers to their pool, and repeated a similar process on their followees. For MSA-DA, they seeded the search with text from Egyptian public figures. For the Second Shared task the language pairs were ES-EN and MSA-DA. For ES-EN they restricted the search of code-switchers based in New York and Miami and seeded the search from local radio station accounts. Again, they continued looking for followers and followees of the radio stations that tweeted code-switched messages. For MSA-DA, the same collection method from the 2014 Shared Task was reused.

All of these approaches to code-switched data collection, except (Samih, 2016), rely on manual inspection to some degree in order to either add a user to the code-switcher pool or select a post for collection. In the next section we introduce a fully automatic approach to finding and collecting code-switched data that is not dependent on manually curating lists of users.

3. Anchoring Methods

We define an anchor as a word which belongs to only one language from a large pool of languages. The motivation behind using anchor words stems from a simple rule to detecting code-switched sentences: "A sentence is code-switched in $L_1 + L_2$ if and only if it contains at least one anchor from language L_1 and at least one anchor from language L_2, and contains no anchors from any other language from the pool of languages $\mathbb{L}$."

The set of anchor words for a language L_i is computed as the set difference between its word lexicon $V(L_i)$ and the union of all other lexicons in the language pool:

$$AnchorSet(L_i) = V(L_i) \setminus \cup_{j!=i} V(L_j) \quad (1)$$

Note that the identification of the anchor sets for a given language pair depends upon the monolingual corpora used. We can relax the definition of anchors in two different ways. First, in the context of detecting $L_1 + L_2$ language, we say a word is a "weak anchor" if it is seen in monolingual L_1 corpora, and never seen in monolingual L_2 corpora. Second, querying the Twitter API with every possible pair of one Spanish and one English anchor is unproductive because there are billions of possible queries and most of them would have no results. To avoid this problem we relaxed the definition of code-switching to: "a sentence is code-switched if and only if it is predicted to be L_1 by a monolingual automatic Language Identification program and contains at least one weak anchor from the L_2 anchor set." With this new rule we require only one anchor from one of our language pair plus language id results favoring the other member of the pair. We note that the definition of weak anchors closely resembles the definition of blacklisted words used by (Tiedemann and Ljubešić, 2012), although their application was to discriminate between a set of very similar languages (Serbian, Croatian and Bosnian). Using these definitions, we performed a preliminary study on the task of classifying an utterance as monolingual or code-switched on the EMNLP 2016 Shared Task Corpus of Spanish+English tweets. Details of the collection and contents of that corpus were given in Section 2.. We computed the anchors for Spanish and English from the Leipzig corpora Collection (LCC), released 2007 to 2014 (Goldhahn et al., 2012). The LCC is a collection of corpora for a large set of languages from comparable sources (e.g. Wikipedia, news articles, websites). We computed the word lexicon of every language in the corpus from the news dataset for that language, and then we computed the anchor list first following equation 1. Words that contained numbers or tokens from a list of 31 punctuation tokens were discarded. In total the language pool contained 134 languages. The Spanish anchor set contained 50.68% of the words from the Spanish word lexicon and the English anchor set contained 54.37% of the words from the English lexicon. In both cases, this is one of the smaller percentages from the pool of 134 languages. In comparison, German, French

Method	Class	Prec.	Recall	F1-score
Anchors	Mono	0.58	1.00	0.73
	CS	0.94	0.03	0.07
Weak Anchors	Mono	0.68	0.98	0.80
	CS	0.93	0.38	0.54
Weak +LID	Mono	0.66	0.98	0.79
	CS	0.93	0.33	0.49

Table 1: Performance on the task of code-switched sentence detection using three definitions for anchoring.

and Italian kept 79.01, 59.67 and 62.94% of their lexicons, while other languages like Chinese and Japanese kept 93.40 and 72.18%.

Table 1 shows Precision, Recall and F1-Score results on the task of classifying a tweet as code-switched (CS) or monolingual (mono) for the strong definition of anchors, weak anchors and the weak anchor + LID approach. We report results on the test partition of the EMNLP 2016 Shared Task Corpus. The language ID used is langid.py (Lui and Baldwin, 2012).

The top subtable from Table 1 shows the results we obtained for this task using our strong definition of anchor. Not surprisingly, we achieved very high precision, but very low recall. High precision and low recall is a secondary effect from the restrictiveness of the definition of anchor set and code-switched sentence, since anchors are not defined exclusively in terms of L_1 and L_2, but from a large pool of languages. This means that the words in the anchor set are most likely to be very low-frequency words. Furthermore the fact that a sentence must have at least one anchor from both languages and none from all the other languages, guarantees that much of the data will be rejected as not code-switched even when bilingual speakers of the languages in question would agree that it is.

The middle subtable from Table 1 shows the results on the task using weak anchors as defined above. At the expense of 0.01 absolute precision points, recall is improved by almost 0.35 points.

The bottom subtable of Table 1 shows results using weak anchors and Language Id. Although with this method the recall drops 0.03 points with respect to the weak anchors, we achieve the advantage of being able to reduce the number of queries we need for the collection, and make the search less restrictive. In the next section of the paper we use weak anchors with the Language ID restriction to collect code-switched tweets.

4. Data Collection

We used Babler[1] (Mendels et al., 2016) to collect code-switched data from Twitter. Babler is a tool designed for harvesting web-data for NLP and machine learning tasks. Babler's pipeline is launched by querying a seed word $s \in S$ using Twitter's API. The tweets retrieved by the query are later processed and passed through a set of filtering rules R which are predefined for the task.

Following the definition of "weak anchor plus Language Id" given in section 3. we used the "weak" anchors to seed the Twitter API and the filtering rules R to enforce the LID restriction. To further reduce the number of required queries we also sort our "weak" anchors by frequency. The weak anchors were computed from the GigaCorpus dataset of Broadcast News data (Consortium and others, 2003; Graff, 2011). R uses Twitter's LID to only allow tweets that were seeded from a Spanish anchor and classified as English or vice versa. Although we require the Twitter API to return only exact matches to our seed terms, we found that in fact Twitter performs stemming.

Our method differs from the prior art in two aspects. First, we derive our word lists from non-noisy pure monolingual corpora which reduces the risk of including out-of-language tokens. Second, instead of performing local filtration our method is implemented based only on API calls thus increasing our potential dataset to every public tweet available. Overall we collected 14,247 tweets that were seeded from Spanish weak anchors and classified as English by the Twitter API and 28,988 tweets that were seeded from English weak anchors and classified as Spanish.

5. Crowdsourcing Language Tags

While we designed our data collection pipeline to save only code-switched tweets, we next needed to test this, as well as to obtain manual annotations for our language modeling research.

From the more than forty-three thousand tweets that were collected, we randomly chose a subset of 8,285 tweets for our "Anchored" tweets corpus [2]. We crowdsourced language tags for every word in our Anchored tweet dataset. Each word was tagged as English (EN), Spanish (ES), Ambiguous between English and Spanish (AMBIG), Mixed English-Spanish (MIXED), Named Entity (NE), Foreign Word (FW), Other (OTHER) and Gibberish (UNK). "Named Entities" were defined as single proper names or part of a name or title that refer to persons, places, organizations, locations, brands, goods, initials, movie titles and song titles. A word is to be tagged as "Ambiguous" when it can be used in both English and Spanish, but there is not enough context to decide its use in the current tweet. A word is to be tagged "Mixed" when the word does not exist in Spanish or English, but consists of a combination of elements from both, e.g. the word "ripeado" which contains the English root "rip" and the Spanish morpheme "-ado". The category "Other" is to be used to tag punctuation, numbers, emoticons, retweet symbols, and other non-lexical items. Finally the "Gibberish" category is for tokens whose meaning cannot be identified.

We used the guidelines used for the annotation of the EMNLP 2016 Shared Task dataset, with some minor changes, including a large number of examples per language tag, and reminders to the annotators throughout the instructions and question statements that a) hashtags were to be tagged with the language tag of the words in the hashtag, and b) Named Entities had precedence over any other

[1] Babler is publicly available from `https://github.com/gidim/Babler`

[2] All the anchor wordlists, tweet IDs and their crowdsourced language tags are publicly available in `http://www.cs.columbia.edu/~vsoto/files/lrec_2018_package.zip`

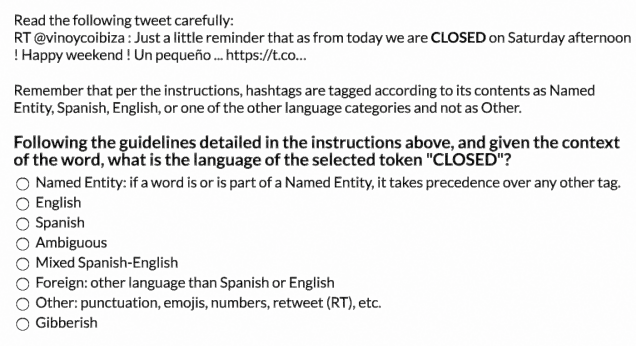


Figure 1: Instance of the annotation task implemented in Crowdflower.

Lang Tag	#Tokens	Avg. Conf
ES	40,208	0.97
EN	30,372	0.93
AMBIG	919	0.55
MIXED	129	0.54
NE	15,260	0.88
FW	1,815	0.77
OTHER	1,994	0.80
UNK	546	0.59

Table 2: Number of tokens and average confidence per Language ID tag from the crowdsourced annotations for the Anchored Twitter corpus.

language tag, since these were the test questions they had the most difficulty with in our initial test.

We used Crowdflower to crowdsource language tags for our tweets. An example of the task our workers were asked to complete can be seen in Figure 1. Our workers were prescreened using a quiz of twenty test questions. If three or more test questions were missed during the initial quiz, the worker was denied access to the task. Furthermore, workers were required to be certified for the Spanish language requirement in Crowdflower. Only workers from Argentina, Canada, Mexico, Spain, U.K. and U.S.A. were allowed access to the task. The task was designed to present 20 questions per page plus one test question used to assess workers' performance. When a worker reached an accuracy lower than 85% on these test questions, all their submitted judgments were discarded and the task made subsequently unavailable. Every set of 19+1 judgments was paid 1 cent (USD).

In total, we collected three judgments per token. The average inter-annotator agreement was 92.33% and the average test question accuracy was 91.1%. These metrics demonstrate that the crowdsourced language labels are of high-quality. For every token for which we crowdsourced a language tag, Crowdflower computes the confidence on the language tag as the level of agreement between all the contributors that predicted that language tag weighted by the contributors' trust scores. The language tag with highest confidence is then chosen as aggregated prediction. Table 2 shows the average confidence per language tag across all tokens. It can be seen that workers struggled the most when tagging words as Mixed, Ambiguous or Gibberish.

	Workshop		Anchored
	Train-Dev	Test	Full
#Tweets(K)	14.4	10.7	8.5
#Tokens(K)	172.8	121.4	130.7
#Switches(K)	7.4	7.8	10.2
Avg. #swts	0.52	0.73	**1.19**
Swt.words(%)	4.30	6.42	**7.77**
Swt.tweets(#)	4,116	4,617	5,958
Swt.tweets(%)	28.56	43.09	**69.89**
0 swt. (%)	**71.44**	56.91	30.11
1 swt. (%)	12.86	21.38	39.57
2 swt. (%)	11.34	16.65	19.53
3 swt. (%)	2.50	2.88	5.81
4 swt. (%)	1.27	1.66	3.32
5 swt. (%)	0.29	0.33	0.84
6 swt. (%)	0.20	0.17	0.43
7 swt. (%)	0.05	0.02	0.23
8 swt. (%)	0.03	0.00	0.12

Table 3: Code-switching statistics for the EMNLP 2016 Workshop and Anchored Tweets datasets. A code-switched word is a word whose language is different from the word that precedes it. The bottom subtable shows the percentage of tweets that contain *N* code-switches.

6. Evaluation

6.1. Data Assessment

Given the crowdsourced LID labels, we can assess the quality of the retrieved anchored tweets by computing their degree of bilingualism and how frequently code-switching occurs within them. We compare these measures to the EMNLP 2016 CS Shared Task corpus (Molina et al., 2016). The train and dev tweets from the 2016 Shared Task were the train and test sets from the 2014 Shared Task (Solorio et al., 2014), whereas the test split was collected specifically for the 2016 task. The collection schemes used in 2014 and 2016 were explained in detail in Section 2.. Table 3 provides the overall statistics describing this corpus in comparison to ours. We report the train-dev and test splits of the EMNLP 2016 Workshop Shared Task corpus separately since they were collected using different methods. As can be seen in Table 3, our subset of 8,525 tweets had an average of 1.19 code-switches per tweet, with 7.77% of words in a tweet being followed by a switch. 69.89% of our tweets contained at least one or more switches. In comparison, the Workshop corpus had an average of 0.61 code-switches per tweet, with 5.17% of tokens followed by a switch. Only 34.75% tweets contained at least one switch. The test set of the Workshop corpus shows greater degrees of bilingualism and a better switching rate: Test corpus tweets averaged 0.73 code-switches per tweet, with 6.42% of tokens followed by a switch and contained 43.09% code-switched tweets overall. Based on these metrics alone, it would appear that our anchoring method improves over the earlier approach considerably.

Table 4 shows the language composition of the three datasets: Workshop training-dev, Workshop test, and the full Anchored dataset. From this table we can see that the train-dev portion of the workshop corpus has a majority (>55%) of English words, while the test split contains a large majority of Spanish words (>63.44%), perhaps due to

	Workshop		Anchored
Lang Tag	Train-Dev	Test	Full
ES	24.51	63.44	34.44
EN	55.33	13.95	24.73
AMBIG	0.23	0.00	0.70
MIXED	0.04	0.00	0.10
NE	2.09	1.72	11.68
FW	0.01	0.02	1.39
OTHER	17.62	20.84	26.53
UNK	0.17	0.02	0.42

Table 4: Language composition (%, token level) for the EMNLP 2016 Workshop and Anchored Tweets datasets.

seeding the collection of tweets on Spanish-language Radio accounts and followers/ees. In comparison, the Anchored corpus is more balanced, with 34.44 and 24.73% of Spanish and English tokens. It also has a higher rate of Named Entities and Other tokens. We believe this is due to the updated annotation guidelines that emphasized the subtleties involved in annotating Named Entities and Other tokens. While Table 4 compares the corpora by language

	Workshop		Anchored
Switch Type	Train-Dev	Test	Full
ES EN	32.06	45.68	29.81
EN ES	31.47	28.36	22.86
EN Other+ ES	15.99	12.28	14.83
ES Other+ EN	15.16	11.05	10.86
ES NE+ EN	1.44	0.99	4.06
EN NE+ ES	0.91	0.36	2.45

Table 5: Types of code-switching sorted by frequency (%). TAG+ indicates a sequence of one or more occurrences of that language tag.

composition, Table 5 examines the corpora by type of switch. The most frequent switch across datasets is Spanish to English (ES-EN), followed by English to Spanish (EN-ES). These account for 63.53%, 74.04% and 52.67% of switches for the Workshop Train-Dev, Workshop Test and Anchored datasets respectively. The next most common type of switch is an English word followed by a sequence of Other tokens and a Spanish word (EN-Other-ES), or Spanish followed by Other and then English (ES-Other-EN). These make up for 31.15%, 23.33% and 25.69% of the switches. Note that this type of switch can be indicative of inter-sentential code-switching if the Other token is a punctuation mark (like '!' in "Holaaaa mis niños bellos!!! I love you guys") or it can be indicative of intra-sentential code-switching if the other token is a Twitter mention, a quote, and so on (e.g "En cuestiones de Rock 'n' Roll I am pretty crossover"). Overall, the typing distribution is more balanced in the Anchored dataset, whereas the Workshop test set has a significant majority of ES EN switches, due perhaps, again, to the way the collection of tweets was seeded.

6.2. Language Identification

Our second evaluation of the accuracy of our corpus consists of training and testing Language ID taggers on the new dataset and comparing its performance to a tagger trained on the Workshop data. We made use of a high-performing classification model from Jaech et al. (2016b). The model did well on the English-Spanish code-switching 2016 Shared Task, especially considering that it was one of only two models that did not use external resources for training (Molina et al., 2016). The same model did well on a sentence level language identification task (Jaech et al., 2016a).

We summarize the model architecture and its motivation here. For a full description see Jaech et al. (2016b). The model is a hierarchical neural model with one level that operates on character sequences to build a representation for each word and a second level that operates on the sequence of word representations to predict the language tag for each word. In the first level, the model uses convolutional neural network layers to do a soft-version of n-gram matching. The output of this layer is a feature vector that provides a useful signal for the language of each word because languages tend to differ in their character n-gram distributions. The second level of the model is a bidirectional LSTM that takes as input the feature vectors from the previous layer and outputs the predicted tag for each word. The use of the LSTM allows the model to incorporate evidence from arbitrary far away in the word sequence. We made one tweak that was not described in Jaech et al. (2016b): the standard LSTM was replaced with an LSTM that has coupled input and forget gates for a 25% reduction in the parameters in the bi-LSTM and a corresponding improvement in speed of computation (Greff et al., 2016). Operating on the word-level representations allows the LSTM to predict the correct tag for words whose language is ambiguous from just the character-level feature vectors based on the fact that adjacent words are more likely to belong to the same language.

We tune the model hyper-parameters by training and testing on the train and dev splits of the Workshop dataset, effectively making the task harder for the model trained on the Anchored corpus. Table 6 shows the word-level and sentence-level accuracy and the average F1-score of the language ID tagset for each training/testing combination.

First, we trained our tagger on the Workshop data (Workshop Model, in Table 6) and observed that its performance on the Workshop test set is similar to that reported for this model in the Shared Task (95.93%). The performance of this tagger however sees a big drop of performance on word-level accuracy and sentence-level accuracy when tested on the Anchored test set. This demonstrates that a tagger trained on a corpus comprised of majority of monolingual sentences, with a lower degree of bilingualism and switching rates, has some difficulty generalizing to a more balanced corpus like the Anchored Tweets Corpus.

Second, we partitioned the Anchored corpus in train and test by randomly choosing 1,500 tweets for the test set and leaving the rest for training. We trained a new tagger on the Anchored dataset with the same hyper-parameter settings as the Workshop tagger and report its test performance on Table 6 as Anchored tagger. We observed that the performance of this model on the Workshop data is very good, despite the difference between the two datasets: the word-level accuracy only decreases by 0.8% accuracy points with respect to the Workshop model, whereas the sentence-level

Training Corpus	Word Accuracy (%)		Avg. F1-Score		Sentence Accuracy (%)	
	Workshop	Anchored	Workshop	Anchored	Workshop	Anchored
Workshop	95.93	82.09	0.4218	0.3978	67.91	14.20
Anchored	95.13	**91.86**	**0.4655**	**0.5937**	62.60	**40.13**
Combination	**96.91**	91.61	0.4328	0.5617	**73.53**	39.87

Table 6: Language tagging accuracy (left) and average f1-score (center) at the word level and language tagging accuracy at the sentence-level (right) for each training and testing combination.

Training Corpus	Word Accuracy (%)		Avg. F1-Score		Fragment Accuracy (%)	
	Workshop	Anchored	Workshop	Anchored	Workshop	Anchored
Workshop	85.46	78.96	0.3678	0.3802	84.29	61.67
Anchored	83.85	86.64	0.3617	0.4937	82.61	71.73
Combination	87.44	86.98	0.3722	0.5020	86.51	73.67

Table 7: Language tagging accuracy (left) and average f1-score (center) at the word level and language tagging accuracy at the fragment-level (right) for each training and testing combination on the subset of code-switched fragments.

accuracy decreases by 5.31% points. However the F1-score value sees a relative improvement of 10.36%, which indicates that the new corpus is more similar to the Workshop test split than the Workshop train-dev split. The Anchored-trained tagger achieves 91.86% word-level accuracy on its own test set, with 0.5937 average F1-score value and 40.13% sentence-level accuracy. These results indicate that a tagger trained on the anchored corpus is able to generalize quite well on the same corpus, although overall the classification task is harder than on the Workshop corpus: the best word-level and sentence-level accuracies in the Workshop test set are much higher than in the Anchored test set.

Finally, we trained a tagger on a combination of the Workshop and Anchored training sets. This combined tagger achieves the best word-level accuracy on the Workshop corpus (96.91%) as shown in the last row of Table 6. Similarly the combined tagger also achieves the best sentence-level accuracy on the Workshop test set (73.53%).

Overall, the Anchored tagger achieves the best results on the Anchored test set for every metric (91.86% word-level accuracy, 0.5937 average f1-score and 40.13% sentence-level accuracy), despite being trained on much less data (the anchored train set has 7,025 tweets, the workshop train set has 11,400 tweets and the combined train set has 18,425 tweets). It also achieves the best average f1-score on the Workshop test set (0.4655). The Combination tagger achieves the best word-level and sentence-level accuracy on the Workshop test set (96.91% and 73.53% respectively).

We next examine the performance of the three taggers on the subset of code-switched segments present in each test set in Table 7, where we define a code-switched segment as the minimal span of tokens where a point code-switch occurs. Notice that a segment can be longer than two tokens if there is a Named Entity, Other, Mixed or Ambiguous token in between. For example, from the sentence "I watched The Godfather y me encantó", the code-switched segment would be "watched The Godfather y" where "The Godfather" is a Named Entity.

From this table we can see that, in fact, taggers have most difficulty tagging words that occur in the context of a code-switch, since the accuracy of all three models on both test subsets of code-switched segments suffers a steep decline for the results shown for the complete test set in the left subtable of Table 6. In the case of the Workshop tagger, its accuracy has relative changes of -10.91 and -3.81% on the full workshop and anchored test sets respectively. The Anchored model sees even larger relative decreases of -11.86 and -5.68%. In comparison, the Combination model has the smallest relative decreases in accuracy, with -9.77 and -5.05%. The same trends can be observed for the average F1-Score and the fragment-level accuracy metrics.

Overall the best performing model is the one trained on the combined training sets, followed by the Anchored model, which always gets better metric values on its own test set and achieves similar metric values on the Workshop test set when compared to the Workshop tagger. Notice though that the Anchored model was trained on less than 40% of the number of tweets in the Combined train set.

7. Conclusions

In this paper we present a method, which makes use of *anchoring* and monolingual Language ID, for detecting code-switched text. We relax strict anchoring constraints to query the Twitter API and retrieve code-switched tweets. We crowdsource language tags for the tokens of 8,285 tweets and found that almost 70% of the collected tweets are indeed code-switched. These tweet exhibit a relatively balanced amount of Spanish and English text and a high amount of code-switching per tweet. The average number of code-switches per tweet in the corpus is 1.19 switches while 7.77% of the tokens are followed by a code-switch. These numbers compare favorably to the 2016 EMNLP Workshop Shared Task Code-Switched Twitter corpus, which was obtained with a different, more labor-intensive method. We evaluated the quality of our new Anchored corpus by training state-of-the-art language taggers and showed that a) a tagger trained on the original Workshop corpus exhibits a more considerable drop in accuracy when tested on the Anchored corpus; and b) a tagger trained on the Anchored corpus achieves very good accuracy on both test corpora. These results show great promise for automatic collection of other code-switched corpora for use in training language models and for other NLP and speech tasks.

8. Bibliographical References

Adel, H., Vu, N. T., Kraus, F., Schlippe, T., Li, H., and Schultz, T. (2013a). Recurrent neural network language modeling for code switching conversational speech. In *Proc. of ICASSP*, pages 8411–8415. IEEE.

Adel, H., Vu, N. T., and Schultz, T. (2013b). Combination of recurrent neural networks and factored language models for code-switching language modeling. In *Proc. of ACL*, pages 206–211.

Ahmed, B. H. and Tan, T.-P. (2012). Automatic speech recognition of code switching speech using 1-best rescoring. In *Proc. of IALP*, pages 137–140.

Al-Badrashiny, M. and Diab, M. (2016). The George Washington University system for the code-switching workshop shared task 2016. In *Proc. of the Second Workshop on Computational Approaches to Code Switching*, pages 108–111.

AlGhamdi, F., Molina, G., Diab, M., Solorio, T., Hawwari, A., Soto, V., and Hirschberg, J. (2016). Part of speech tagging for code switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 98–107.

Barman, U., Das, A., Wagner, J., and Foster, J. (2014). Code mixing: A challenge for language identification in the language of social media. In *Proc. of The First Workshop on Computational Approaches to Code Switching*, pages 13–23.

Çetinoglu, Ö. (2016). A Turkish-German codeswitching corpus. In *Proc. of LREC*, pages 4215–4220.

Elfardy, H., Al-Badrashiny, M., and Diab, M. (2014). AIDA: identifying code switching in informal arabic text. In *Proc. of The First Workshop on Computational Approaches to Code Switching*, pages 94–101.

Franco, J. C. and Solorio, T. (2007). Baby-steps towards building a Spanglish language model. In *Proc. of International Conference on Intelligent Text Processing and Computational Linguistics*, pages 75–84. Springer.

Go, A., Huang, L., and Bhayani, R. (2009). Twitter sentiment analysis. *Entropy*, 17:252.

Goldhahn, D., Eckart, T., and Quasthoff, U. (2012). Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proc. of LREC*, pages 759–765.

Goyal, P., Mital, M. R., and Mukerjee, A. (2003). A bilingual parser for Hindi, English and code-switching structures. In *EACL Workshop of Computational Linguistics for South Asian Languages*.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE transactions on neural networks and learning systems*.

Jaech, A., Mulcaire, G., Hathi, S., Ostendorf, M., and Smith, N. A. (2016a). Hierarchical character-word models for language identification. *arXiv preprint arXiv:1608.03030*.

Jaech, A., Mulcaire, G., Hathi, S., Ostendorf, M., and Smith, N. A. (2016b). A neural model for language identification in code-switched tweets. In *Proc. of the Second Workshop on Computational Approaches to Code Switching*, pages 60–64.

Jamatia, A., Gambäck, B., and Das, A. (2015). Part-of-speech tagging for code-mixed English-Hindi Twitter and Facebook chat messages. In *Proc. of Recent Advances in Natural Language Processing*, pages 239–248.

Lee, S. Y. M. and Wang, Z. (2015). Emotion in code-switching texts: Corpus construction and analysis. *Proc. of the Eighth SIGHAN Workshop on Chinese Language Processing*, pages 91–99.

Li, Y. and Fung, P. (2012). Code-switch language model with inversion constraints for mixed language speech recognition. In *Proc. of COLING 2012*, pages 1671–1680.

Li, Y. and Fung, P. (2014). Language modeling with functional head constraint for code switching speech recognition. In *Proc. of EMNLP*, pages 907–916.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proc. of the ACL*, pages 25–30.

Lyudovyk, T. and Pylypenko, V. (2014). Code-switching speech recognition for closely related languages. In *Proc. of SLTU*, pages 188–193.

Mendels, G., Cooper, E., Soto, V., Hirschberg, J., Gales, M. J., Knill, K. M., Ragni, A., and Wang, H. (2015). Improving speech recognition and keyword search for low resource languages using web data. In *Proc. of INTERSPEECH*, pages 829–833.

Mendels, G., Cooper, E., and Hirschberg, J. (2016). Babler-data collection from the web to support speech recognition and keyword search. In *Proc. of Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 72–81.

Molina, G., Rey-Villamizar, N., Solorio, T., AlGhamdi, F., Ghoneim, M., Hawwari, A., and Diab, M. (2016). Overview for the second shared task on language identification in code-switched data. In *Proc. of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49.

Piergallini, M., Shirvani, R., Gautam, G. S., and Chouikha, M. (2016). The Howard University system submission for the shared task in language identification in spanish-english codeswitching. In *Proc. of the Second Workshop on Computational Approaches to Code Switching*, pages 116–120.

Samih, Y., Maharjan, S., Attia, M., Kallmeyer, L., and Solorio, T. (2016). Multilingual code-switching identification via LSTM recurrent neural networks. In *Proc. of the Second Workshop on Computational Approaches to Code Switching*, pages 50–59, Austin, TX.

Samih, Y. (2016). An Arabic-Moroccan darija code-switched corpus. In *Proc. of LREC*, pages 4170–4175.

Solorio, T. and Liu, Y. (2008a). Learning to predict code-switching points. In *Proc. of EMNLP*, pages 973–981.

Solorio, T. and Liu, Y. (2008b). Part-of-speech tagging for English-Spanish code-switched text. In *Proc. of EMNLP*, pages 1051–1060.

Solorio, T., Blair, E., Maharjan, S., Bethard, S., Diab, M., Gohneim, M., Hawwari, A., AlGhamdi, F., Hirschberg,

J., Chang, A., et al. (2014). Overview for the first shared task on language identification in code-switched data. In *Proc. of the First Workshop on Computational Approaches to Code Switching*, pages 62–72.

Tiedemann, J. and Ljubešić, N. (2012). Efficient discrimination between closely related languages. In *COLING 2012*, pages 2619–2634.

Vilares, D., Alonso, M. A., and Gómez-Rodrıguez, C. (2015). Sentiment analysis on monolingual, multilingual and code-switching twitter corpora. In *Proc. of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, pages 2–8.

Vyas, Y., Gella, S., and Sharma, J. (2014). POS tagging of English-Hindi code-mixed social media content. In *Proc. of EMNLP*, pages 974–979.

9. Language Resources References

Consortium, L. D. et al. (2003). English Gigaword. `http://www.ldc.upenn.edu`.

Graff, D. (2011). Spanish Gigaword third edition (ldc2011t12).

Classifying the Informative Behaviour of Emoji in Microblogs

Giulia Donato*, **Patrizia Paggio***†
*University of Copenhagen
giulia.dnt@gmail.com, paggio@hum.ku.dk
†University of Malta
patrizia.paggio@um.edu.mt

Abstract

Emoji are pictographs commonly used in microblogs as emotion markers, but they can also represent a much wider range of concepts. Additionally, they may occur in different positions within a message (e.g. a tweet), appear in sequences or act as word substitute. Emoji must be considered necessary elements in the analysis and processing of user generated content, since they can either provide fundamental syntactic information, emphasize what is already expressed in the text, or carry meaning that cannot be inferred from the words alone. We collected and annotated a corpus of 2475 tweets pairs with the aim of analyzing and then classifying emoji use with respect to redundancy. The best classification model achieved an F-score of 0.7. In this paper we shortly present the corpus, and we describe the classification experiments, explain the predictive features adopted, discuss the problematic aspects of our approach and suggest future improvements.

Keywords: Emoji, Microblogs, Redundancy, Supervised Learning

1. Introduction

Emoji are non verbal features used to enrich computer mediated communication (CMC) and mobile mediated communication (MMC). Empirically, the use of emoji may vary in non trivial ways: emoji can be redundant with respect to the text, but they can also act as words, carrying their own part-of-speech category, thus providing fundamental semantic information. Studies on emotion expression in text have noted that emotional emoji may carry information which could not be inferred from the words alone. Following these findings, this work proposes to further investigate the informative behaviour of emoji in microblogs.

Recognizing to what extent emoji are redundant with respect to one or more words in the text could be helpful for further research in automatic content labeling; understanding how (and to what extent) emoji convey additional meaning or have a syntactic function can be important to improve the results in other NLP tasks such as metaphor detection and text summarization.

2. Literature

Emoji are picture characters, or pictographs, initially developed by Shigetaka Kurita during the late nineties. The initial set was further expanded and eventually became part of the Unicode standard in 2009 (Kelly, 2015; Miller et al., 2016).

Emoji in CMC are, similarly to emoticons, mostly used to express emotions; according to (Swiftkey, 2015) the top used emoji categories are the ones that include the happy and the sad faces. Novak et al. (2015) confirm that these preferences apply also to Twitter users. While Boia et al. (2013) demonstrate that emoticons are not necessarily accurate in retrieving sentiment words from a corpus, Hallsmar and Palm (2016) show that an emoji heuristic can be effectively used to retrieve tweets corresponding to a specific emotion class (within a multiclass framework), perhaps indicating that emotional emoji to some extent may behave redundantly with respect to the tweet text.

Since both emoticons and emoji can be employed with a wide range of purposes (Derks et al., 2007; Kelly and Watts, 2015), some research focused on their semantic aspects. Barbieri et al. (2016) analyzed the distribution of emoji in a corpus of more than 9.000.000 tweets. The emoji embeddings obtained were used to compute similarity and relatedness scores among emoji pairs and the results were evaluated by means of a manually annotated gold standard. The emoji vectors plotted in 2d space showed consistent semantic clusters, however the analysis of the words related to each cluster shows a noisy outcome. This suggests the need to observe the relation between emoji and words with different criteria, for example by considering how and how often emoji can represent information that is missing from the text. Eisner et al. (2016) elaborated on the findings in Barbieri et al. (2016); the authors demonstrated that emoji vectors generated from the sole Unicode descriptions are effective in classifying pairs of emoji with respect to their similarity.

Identifying redundant information in text is a useful step for text summarization or paraphrase detection. In microblogs repeated content can be found within specific conversations or topics. Zanzotto et al. (2011) provided a formal definition of linguistic redundancy in Twitter and performed machine learning experiments to quantify how common the phenomenon is in the social network. The authors experimented with several features combinations and found that the use of thesaurus metrics combined with partial syntactic analysis was the most effective in classifying redundant vs. non redundant instances. This paper is strongly inspired by the framework described in Zanzotto et al. (2011), and investigates the notion of redundant vs non-redundant behaviour in the domain of emoji use.

3. Methodology

The aim of this study is firstly to investigate how easy it is for human coders to distinguish different uses of emoji with respect to their semantic contribution, and secondly to experiment with automatic classification of emoji behaviour.

In order to explore both aspects, we set up a corpus of English tweets each of which contained at least one emoji. The corpus was annotated by four human coders in order to be used in a supervised machine learning experiment.

3.1. Corpus Creation

To collect the data we started by selecting a set of 30 emoji (three per each category among: *Nature & Animals, Places, Traveling/Commuting, Sport, Events, Other Activities, Music, Eating & Drinking, People, Feelings*) that we used as search keywords to retrieve relevant tweets automatically.
The raw data were cleaned and balanced (considering the category) and eventually sum up to a collection of 4100 tweet pairs. The annotation took place remotely between the 21st and the 31st of December 2016 and was performed by four annotators, three located in Greece and one in the Netherlands. All the annotators are fluent English speakers. We presented them with examples randomly drawn from the set of 4100 pairs. Each pair contained the same tweet twice: once with and once without a specific emoji. The coders were asked to select among three possible options to label each pair instance by considering the emoji of interest. The possible classes were: *Redundant*, *Non Redundant*, *Non Redundant + POS*; these classes were described and exemplified in ad-hoc instructions, on the basis of the definition of redundancy provided by Zanzotto et al. (2011).
The Redundant class indicates that the emoji of interest repeats the information present in the text or that its meaning is implied by the text. The Non-redundant class, on the contrary, captures cases in which the emoji adds information that is neither explicitly present nor implied in the text. Lastly the Non-Redundant + POS class, which refers to a specific kind of redundant use, indicates that the emoji is used with a syntactic function (and can be labeled with its own POS), thus replacing a word.
Examples to illustrate the three types of usage are listed below:

1. *Redundant*

   ”We’ll always have Beer. I’ll see to it. I got your back on that one. 🍺”

2. *Non-Redundant*

   ”I wish you were here ✈️”

3. *Non-Redundant + POS*

   ”Thank you so so so so much ily Here’s a 🍕 as a thank you gift x”

An more difficult case if the following one:

> ”Reading is always a good idea 📚. Thank you for your sincere support @USER. Happy reading.”

The content of the emoji might seem implied at first sight; however, books are not the only possible reading media, so in fact the emoji adds a more specific meaning to the tweet.

The annotators worked using an an interface specifically developed for the project. They had random access to roughly 1000 items each from the original sample of 4100 instances, and the process resulted in an annotated corpus consisting of 2475 unique pairs.
The analysis of the classes distribution shows the following counts: the *Redundant* class has 834 instances (33.7%), the *Non-Redundant* class has 1428 instances (57.7%), the *Non-Redundant + POS* class has 176 instances (7.1%). Additionally, 37 instances are annotated as *undefined* (1.5%).
The inter annotator agreement computed by means of the Cohen’s κ coefficient, shows a value of .56, which is considered to be moderate ((Landis and Koch, 1977)). This result and the difficulties reported by the annotators to assign a class in several cases, suggest that the task is not trivial for humans.
We additionally analyzed the corpus wrt potential features: in particular the position of the emoji (further described in 3.3.) and the emoji POS tag obtained from the Stanford POS Tagger were shown to be promising features to consider when training a classifier to predict whether emoji in tweets are being used in a redundant or additive way.
The creation of the corpus and its analysis is described in full in Donato and Paggio (2017).

3.2. Preprocessing

Before running the machine learning experiments, the corpus was preprocessed. Beside the standardization of mentions and links, the tweets were tokenized and stopwords and punctuation were removed. To achieve a proper tokenization on groups of emoji we used **Tweetokenize** (`https://github.com/jaredks/tweetokenize`), an online tokenizer designed for tweets tokenization. The tokenizer considers emoji, emoticons and hashtags as single tokens, and provides a series of further preprocessing methods such as lowercasing. Furthermore, the tokens were stemmed by means of the NLTK Wordnet lemmatizer. Since the Wordnet lemmatizer requires the indication of the word’s POS-tag to generate the stem, we ran a POS tagger on the tweets prior to their lemmatization. we adopted the standard Stanford POS Tagger from the Python NLTK wrapper since traditional POS taggers can achieve satisfactory results when compared with domain specific taggers (Derczynski et al., 2013) and since, to the best of our knowledge, Twitter-specific POS taggers do not provide tags for emoji.

3.3. Features

Emoji Position

Since, as also noted in Novak et al. (2015), the emotional emoji positioned towards the end of the tweet are the more emotionally loaded ones. we wanted to verify if the position any kind of emoji has in a tweet may be an indicator of a specific informative behaviour. For example if it is true that emoji are more likely to be put towards the end of the tweet when they repeat words in the text, the position of the emoji will be potentially interesting for a classification of the tweet with respect to emoji redundancy.

We analyzed this feature in our corpus, firstly by computing the percentage frequency distribution of two possible modalities (close to the end of the tweet or not). We observed that for the close to the end condition 35.7% of the instances are annotated as Redundant, 60.7% as Non-Redundant, 2.9% as Non-Redundant + POS, and 0.7% are undefined. In the opposite condition 31.6% instances are Redundant, 54.6% are Non-Redundant, 11.5% are Non-Redundant + POS, and 2.3% are undefined. We performed then a χ-squared test of independence we obtained (χ-squared = 81.644, df = 3, p-value < 0.001) which indicates a significant difference. An analysis of the residuals confirmed what emerged from the observation of the frequency distribution: the effect of position is highest in the case of the Non-Redundant + POS class.
We obtained the emoji position value by dividing the index of the emoji in the tokenized tweet by the number of tokens in the tweet. However, for the classification the position feature was encoded as binary (1 if the value is above 0.7, 0 otherwise).

Similarity Measure
To detect semantic relations between words we followed a thesaurus-based approach. Considering the WBOW model described in Zanzotto et al. (2011), we computed the distance between the vector of the tweet words (without the emoji under consideration) and the vector of the emoji description given by the Emojitracker integrated by the extended description present in the Unicode website.
To calculate the distances we relied on the Wu-Palmer similarity measure, and used WordNet as the external resource. The Wu-Palmer (*wup*) similarity measure is formalized as follows:

$$Sim_{wup}(c_1, c_2) = \frac{2 * N}{N_1 + N_2 + 2 * N}$$

where N_1 and N_2 indicate the distances that separate c_1 and c_2 (concepts) from the specific common concept, while N is the distance which separates the closest common ancestor of c_1 and c_2 from the root node (Slimani, 2013). The Wu-Palmer score has range $[0, 1]$. As reported by Slimani (2013), this metric is computationally simple yet it is as expressive as other theasurus based metrics.
To build the emoji description vectors we used the descriptions adopted by the Emojipedia; in two cases (U+1F383 and U+1F612), however, we replaced the primary description with a secondary description (obtained from the Unicode website) to ensure the retrieval of an existing match within Wordnet. Moreover, the words were stemmed and all the descriptions longer than one word were tokenized, and all the words not found in Wordnet were removed.
An example of a tweet vector (in italics) paired with an emoji description vector (in bold) is:
⟨*'come', 'celebrate', 'castle', 'get', 'special', 'offer'*⟩
⟨**'european', 'castle'**⟩
In this case the resulting distance vector is as follows:
⟨0.308, 0.235, *None*, *None*, 0.421, **1.0**, 0.105, 0.087, 0.167, 0.125, 0.1538 0.118⟩.
We can see that there one perfect correspondence (in bold) indicating an exact match between a word in the emoji description vector and a word in the tweet vector. The *None* values indicate that there is no existing path connecting the two words in WordNet.
Once the distances between the words in the tweet vector and the words in the emoji vector were computed, the highest value was kept as the resulting feature for the classification experiments. The assumption is that a maximum value close to 1 (ideally between 0.9 and 1) will indicate that the tweet vector contains a synonym of at least one word present in the emoji description.
Although thesaurus metrics have drawbacks, since they may establish high scores also among antonyms, this feature is expected to be effective in discriminating the Redundant class from the other classes.

Tf-Idf Bag of Words
A tf-idf matrix computed on unigram vectors represents each word in a document (a tweet in our case) as a weight which indicates how important that word is to identify the document's class. We adopted a tf-idf matrix computed on words combined with their POS tags. This choice has an impact on the number of features, which increases, and may include useful ones. However, as a trade off, the matrix will be sparser. This model does not consider word order and context, however we wanted to verify if a unigram model could improve over the baseline and potentially be used in combination with more complex features.

3.4. Classification

To perform the automatic classification we used the Python sklearn implementation of a Linear Support Vector Classifier (Pedregosa et al., 2011), set up for a multiclass classification.
To evaluate the proposed methodology we experimented with three different feature combinations and established two baselines to compare the different performances of our models.

Baseline I
The corpus is unbalanced since most of the instances are labeled as "Non-Redundant", therefore the lower bound for the classifier performance is given by assigning to each instance the most frequent class.

Baseline II
To verify if a unigram model can improve over a majority class baseline the classifier was trained on the tf-idf matrix built over the corpus unigrams combined with their POS tags.

SimPos
The third model is based on a combination of similarity and position features. This feature combination was expected to improve over both baselines.

Tf-Idf + SimPos
In this model the classifier was trained on the tf-idf matrix combined with SimPos.

LSA + SimPos
Lastly we performed dimensionality reduction by means of LSA on the tf-idf matrix and replicated the experiment on a 100-dimensional matrix combined with SimPos.

4. Results and discussion

Table 1 reports accuracy and F1 scores obtained with the same classifier on each of the illustrated models.

Model	Accuracy	F1 Score
Baseline I	59%	0.44
Baseline II	69.6%	0.66
SimPos	67.2%	0.64
Tf-idf+SimPos	71.8%	0.69
LSA+SimPos	**73%**	**0.7**

Table 1: Classification Results

The experiment was performed by means of a simple train test split where 50% of the corpus was used to train the classifier and 50% for testing. This approach was adopted following Zanzotto et al. (2011).

Comparison	McNemar's X^2
Baseline I – Baseline II	52.393
Baseline II – SimPos	86.218
Baseline I – SimPos	47.422
SimPos – Tf-idf+SimPos	88.347
Tf-idf+SimPos – LSA+SimPos	112.79

Table 2: Results of significance testing obtained using McNemar's X^2 test. In all cases, df=1, and $p < 0.001$.

The differences between the various models were tested pairwise by means of McNemar's X^2 test (Dietterich, 1998; Bostanci and Bostanci, 2013) and found statistically significant. The results of these tests are shown in Table 2.

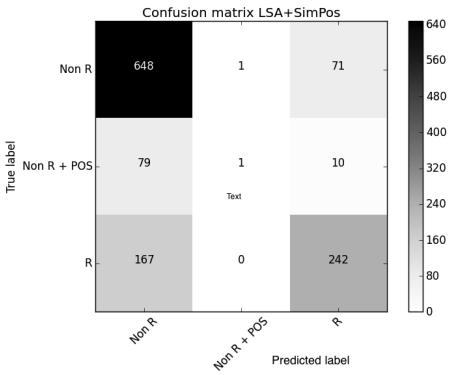


Figure 1: Confusion matrix showing the classifier's choices for the best performing model

In general, none of the models is good at recognizing instances of the least represented class (Non-Redundant + POS) for which only a minimal fraction of relevant instances is correctly labeled, suggesting that more training data are necessary to identify this class. Figure 1 shows the confusion matrix derived from the results of the best performing model, LSA+SimPos. If these results are compared with those obtained with the simpler models, it is clear that adding the semantic similarity measure increases the number of the instances of the Redundant class that are correctly classified (from 198 in the simplest model to 242). The remaining errors may be due to the fact that the emoji description vectors did not include the full Unicode emoji description, that distances were computed only among unigrams rather than n-grams. Furthermore, possible synonyms may be enclosed in a hashtag, thus it would be impossible to compute the similarity between hashtags and the emoji description vector without additional preprocessing, since hashtags may consist in a combination of words plus the hash symbol attached at the front (e.g. "#autumninjapan").

5. Conclusion

In the present work we investigated the informative behaviour of emoji in Twitter. The main interest was, in particular, in testing whether both human annotators and an automatic classifier can be trained to distinguish the use of emoji as being either redundant, non-redundant or as a word (thus, with syntactic properties). Furthermore, we were interested in determining whether specific features, such as the emoji position and the description of their content, could help a classifier in discriminating between these different conditions. We found the task to be difficult for human annotators. However, the results of our classification experiments show that the annotated corpus is still reliable enough for us to be able to obtain acceptable results. As regards the automatic classification, it emerged that the combination of the engineered features - SimPos - is more effective than the proposed majority class baseline in discriminating among the three classes. However, the best classification results were achieved when combining these features with tf-idf weights of the unigrams combined with their POS tags, and by applying dimensionality reduction to the resulting values.

To sum up, we have described a corpus of annotated tweets that can be used to study emoji usage with respect to whether they are redundant or add content to the text. Then we have demonstrated that automatic classification of emoji in tweets with respect to their redundancy can be achieved with an F-score of 0.7. Furthermore, we have shown that a combination of the unigrams tf-idf matrix reduced by means of LSA with position and similarity features is effective in reaching these results and performs better than two different baselines.

There are several aspects discussed in this work that may constitute a limitation and are, therefore, open to improvements and changes. First of all, the corpus is not very large even if it compares with the size of the dataset used in Zanzotto et al. (2011).To overcome this limitation the optimal solution would be to collect more data, both in terms of better coverage of the three classes and of different emoji tokens.

In the machine learning experiments we used 50% of the data for the training process and 50% for the test, following the setup proposed by Zanzotto et al. (2011). However, we are aware that for datasets the size of the presented one, a better approach would be to evaluate the model by means of cross validation. This will certainly be done in future.

More advanced models should also be tested, in particular models involving word embeddings. Alternatively, comparing similarity features obtained with different metrics (e.g. cosine similarity) and classification results obtained

by using different classifiers could also be a feasible approach to improve this research.

Further analysis can be done to verify whether the emoji that are mostly used in a redundant or non-redundant way belong to a specific semantic category. Furthermore, it could be interesting to test a simplified design, by collapsing the Non-Redundant and the Non-Redundant + POS in a single category, thus implementing a binary classification only concerned with the two broad semantic categories.

6. Acknowledgements

We would like to thank George Giannakopoulos, the Institute of Informatics and Telecommunications at NSCR Demokritos in Athens and everybody at Sci.FY.

7. Bibliographical References

Barbieri, F., Ronzano, F., and Saggion, H. (2016). What does this emoji mean? a vector space skip-gram model for twitter emojis. In *Language Resources and Evaluation conference, LREC*, Portoroz, Slovenia, May.

Boia, M., Faltings, B., Musat, C.-C., and Pu, P. (2013). A:) is worth a thousand words: How people attach sentiment to emoticons and words in tweets. In *Social Computing (SocialCom), 2013 International Conference on*, pages 345–350. IEEE.

Bostanci, B. and Bostanci, E. (2013). An evaluation of classification algorithms using Mc Nemar's test. In *Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012)*, pages 15–26. Springer India.

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013). Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *RANLP*, pages 198–206.

Derks, D., Bos, A. E., and Von Grumbkow, J. (2007). Emoticons and online message interpretation. *Social Science Computer Review*.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, October.

Donato, G. and Paggio, P. (2017). Investigating redundancy in emoji use: Study on a twitter based corpus. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 118–126, Copenhagen, Denmark, September. Association for Computational Linguistics.

Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., and Riedel, S. (2016). emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359*.

Hallsmar, F. and Palm, J. (2016). Multi-class sentiment classification on twitter using an emoji training heuristic.

Kelly, R. and Watts, L. (2015). Characterising the inventive appropriation of emoji as relationally meaningful in mediated close personal relationships. *Experiences of Technology Appropriation: Unanticipated Users, Usage, Circumstances, and Design*.

Kelly, C. (2015). Do you know what i mean >:(: A linguistic study of the understanding of emoticons and emojis in text messages.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Miller, H., Thebault-Spieker, J., Chang, S., Johnson, I., Terveen, L., and Hecht, B. (2016). "blissfully happy" or "ready to fight": Varying interpretations of emoji. *ICWSM'16*.

Novak, P. K., Smailović, J., Sluban, B., and Mozetič, I. (2015). Sentiment of emojis. *PloS one*, 10(12):e0144296.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Slimani, T. (2013). Description and evaluation of semantic similarity measures approaches. *arXiv preprint arXiv:1310.8059*.

Swiftkey. (2015). Swiftkey emoji report april 2015. https://blog.swiftkey.com/americans-love-skulls-brazilians-love-cats-swiftkey-emoji-meanings-report/.

Zanzotto, F. M., Pennacchiotti, M., and Tsioutsiouliklis, K. (2011). Linguistic redundancy in twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 659–669. Association for Computational Linguistics.

A Taxonomy for In-depth Evaluation of Normalization for User Generated Content

Rob van der Goot, Rik van Noord, Gertjan van Noord
University of Groningen
Broerstraat 5, Groningen,
r.m.van.der.goot@rug.nl, r.i.k.van.noord@rug.nl, g.j.m.van.noord@rug.nl

Abstract

In this work we present a taxonomy of error categories for lexical normalization, which is the task of translating user generated content to canonical language. We annotate a recent normalization dataset to test the practical use of the taxonomy and read a near-perfect agreement. This annotated dataset is then used to evaluate how an existing normalization model performs on the different categories of the taxonomy. The results of this evaluation reveal that some of the problematic categories only include minor transformations, whereas most regular transformations are solved quite well.

Keywords: normalization, user generated content, social media, error taxonomy

1. Introduction

For other natural language processing tasks, such as grammatical error correction and machine translation, there already exist detailed error taxonomies, which help in getting insights in the strengths and weaknesses of systems. For normalization, such an evaluation does not exist yet. Reynaert (2008) proposed an evaluation framework which evaluates the different sub-tasks in more detail; enabling the evaluation of error detection, candidate generation and candidate ranking. In the more recent shared task on lexical normalization hosted at the WNUT workshop (Baldwin et al., 2015b), the outputs of systems were evaluated on precision, recall and F1 score. Orthogonal to these approaches, we propose a more in-depth evaluation of normalization, focusing on categories of different normalization replacements.

Existing error taxonomies are unfortunately not suitable for the task of normalization, since the categories are substantially different. For machine translation, taxonomies as the Multidimensional Quality Metrics (Mariana, 2014) are proposed, which contains 3 main categories: accuracy, verity and fluency. The last category would be the most relevant for normalization, but the normalization task compromises a different variety of errors and anomalies. In grammatical error correction, often a more detailed taxonomy for errors is used; the default benchmark has 28 categories (Ng et al., 2014). However, many of the errors in this taxonomy are not annotated in the normalization benchmarks and many normalization replacements are not included in this taxonomy.

Different benchmarks for normalization specify the task slightly different; one striking example is the inclusion of the expansions of phrasal abbreviations (e.g. 'lol'↦'laughing out loud'). However, this might not be the desired output, since one might argue that this expansion does not represent the intended meaning. This reveals another potential use for a taxonomy of normalization actions: it enables us to filter the categories before training, and thus learn a model which only handles the desired categories.

2. Normalization

There is ample of previous work on normalization, but there is no consensus about the scope of the normalization task. Some existing corpora with normalization annotation for English are shown in Table 1. The corpora by Baldwin and Li (2015) and Kaljahi et al. (2015) are also annotated with error categories. However, the guidelines for the annotation of these corpora are substantially different compared to the other, more commonly used, corpora. The taxonomy proposed by Baldwin and Li (2015) has a very high percentage of normalizations since it allows deletion and insertion of tokens as well as the correction of capitalization. In contrast, The Foreebank (Kaljahi et al., 2015) has a very low percentage of normalized words. This is due to its more canonical forum domain; it is mostly focused on grammatical error correction.

In the rest of this section, we will discuss two examples from these datasets to give a clearer idea of the task:

(1) most social pple r troublesome
most social people are troublesome

Example 1 shows an example tweet from the LexNorm2015 (Baldwin et al., 2015b) corpus and it's annotated normalization. This example includes two subsequent words which are shortened by omitting vowels.

(2) i aint messin with no1s wifey yo lol
i ain't messing with no one's wifey you laughing out loud

Example 2 includes two examples of 1-n normalization replacements; the first replacement is not only a split, '1s' is also expanded to 'one's' in the annotation. The second 1-n replacement is the expansion of the phrasal abbreviation of 'lol'. The word 'wifey' is kept unchanged, this reflects the conservativeness which is encouraged in the annotation guidelines.

Source	Name	Punct.	Caps.	1-n n-1	Words	%normalized
Yang and Eisenstein (2013)	LexNorm1.2	-	-	-	10,576	11.5
Li and Liu (2014)		-	+-	-	40.560	10.5
Baldwin et al. (2015b)	LexNorm2015	-	-	+	73,806	9.1
Baldwin and Li (2015)		+	+	+	11,890	28.7
Kaljahi et al. (2015)	Foreebank	+	+	+	15,595	3.3

Table 1: Properties of different annotation guidelines. Punct. and Caps.: if respectively punctuation and capitalization is corrected. +- means capitalization is simply copied from the original utterance. The '1-n n-1' column indicates if normalization corrections beyond the word level are allowed.

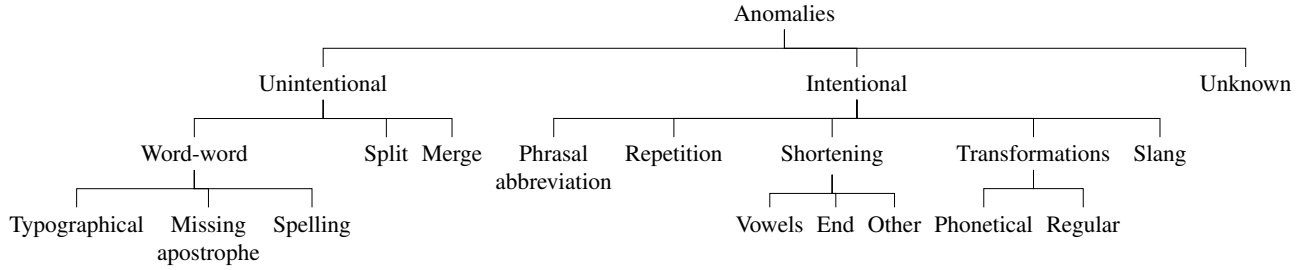


Figure 1: Our proposed taxonomy of anomalies in user generated text.

3. Proposed Taxonomy

Our proposed taxonomy is loosely based on the categories used by the Foreebank (Kaljahi et al., 2015) and Baldwin and Li (2015). Furthermore we took categories from the annotation guidelines of LexNorm2015 (Baldwin et al., 2015a), since they include which kind of anomalies should be annotated. The categories of our taxonomy are a combination of the previously used categories, but are empirically refined during the early stages of annotation. We make a main distinction between intentional and unintentional anomalies since they have a different origin; meaning they might require different handling in NLP systems. Note that we do not include word reordering and capitalization, since these phenomena are not annotated in our dataset. Our taxonomy is shown in Figure 1; accompanying examples can be found in Table 2. We will now describe each final category in more detail.

1. Typographical error This includes small errors, which are a result of mistyping keys on keyboards. In case of doubt with another category, we put words with a character edit distance of one in this category (e.g. bidge↦bridge, feela↦feels).

2. Missing apostrophe In social media text, the apostrophe is often skipped. Even though this category is relatively trivial to solve, it might have large effects in a pipeline approach, since it can resolve tokenization issues.

3. Spelling error This category includes all cases in which a word is unintentionally used in the wrong form/context, including spelling and grammatical errors. We also include mismatches between American and British English here. When in doubt with the first category, annotators should answer the following question: if the sender were to send the message again, would he/she make the same mistake?

4. Split When a word is split into multiple words. There is one case in our corpus where this happens intentionally ('l o v e'↦love), this is still annotated in this category.

5. Merge There is no space between two different words, this is a special case of a typograpical error.

6. Phrasal abbreviation In some datasets, phrasal abbreviations, such as 'lol', 'idk' and 'brb' are expanded to respectively 'laughing out loud', 'I don't know' and 'be right back'. These abbreviations consist of all first characters of the words they represent.

7. Repetition In social media, extra focus is put on words by character repetition. Repetition can also occur on a character N-gram level, e.g. 'hahahahahaha'. Even when adding only one extra character, we categorize the replacement here. If this category collides with a phrasal abbreviation (e.g. lololol), we choose to categorize it as phrasal abbreviation, since this is more defining for the intended meaning.

8. Shortening vowels A simple way to shorten words is to leave out vowels. In this category we also place words in which most of the vowels are removed, e.g. pple↦people.

9. Shortening end Another way to shorten words is too leave out the last syllable(s) or character(s). Based on context, it is often trivial for humans to understand which ending is intended. If the anomaly includes a suffix to indicate plurality, we still classify it in this category (e.g. favs↦favorites).

10. Shortening other There are other variations to shorten words. For example, using only the first letter of each part of a compound, skipping another syllable then the last or using standard abbreviations (pdx↦portland). This category also contains combinations of the previous two categories (talkn↦talking, smth↦something).

Category	Examples
1. Typographical error	spirite↦spirit, complaing↦complaining, throwg↦throw
2. Missing apostrophe	im↦i'm, yall↦y'all, microsofts↦microsoft's
3. Spelling error	favourite↦favorite, dieing↦dying, theirselves↦themselves
4. Split	pre order↦preorder, screen shot↦screenshot
5. Merge	alot↦a lot, nomore↦no more, appstore↦app store
6. Phrasal abbreviation	lol↦laughing out loud, pmsl↦pissing myself laughing
7. Repetition	soooo↦so, weiiiiird↦weird
8. Shortening vowels	pls↦please, wrked↦worked, rmx↦remix
9. Shortening end	gon↦gonna, congrats↦congratulations, g↦girl
10. Shortening other	cause↦because, smth↦something, tl↦timeline,
11. Phonetic transformation	hackd↦hacked, gentille↦gentle, rizky↦risky
12. Regular transformation	foolin↦fooling, wateva↦whatever, droppin↦dropping
13. Slang	cuz↦because, fina↦going to, plz↦please
14. Unknown	skepta↦sunglasses, putos↦photos

Table 2: Examples of normalization pairs for each category.

11. Phonetic transformation In an effort to shorten texts phonetic transformations are often used. In this case one or multiple characters are converted to their literal pronunciation to form a word. This can be done with numbers as well as letters. Other cases of phonological transformation occur when people replace characters with similar sounding characters, e.g. s↦z, c↦k d↦t. Note that transformations of word endings like -er↦-a (e.g. brotha↦ brother) fit in the next category, as the normalization is pronounced differently compared to the original word.

12. Regular transformation For this category, we consider common transformations of endings of words. On Twitter it is common to end participles and gerunds with 'in' instead of 'ing'. Another common transformation is to replace the last syllable with 'a'. Note that transformations like cuz↦because does not fit in this category, because this transformation is not transferable to other words.

13. Slang This category includes all other transformations as well as novel words specific to this domain.

14. Unk Annotator is not sure in which category a word belongs. This can be because the annotator does not agree with the normalization annotation, or because the Tweet is not understandable for the annotator.

4. Annotation

To test our proposed taxonomy, we annotated the LexNorm2015 dataset (Baldwin et al., 2015b) with an extra layer, which indicates for each normalization replacement to which category in our taxonomy it belongs. We choose to annotate this dataset because it is publicly available, the most recent, the largest and annotation is verified by shared task participants. It should be noted however, that as long as alignment is available, the taxonomy can easily be adopted for other corpora.

To ease the annotation effort, we annotate unique normalization replacement pairs. Since most ambiguity problems should already be solved by the normalization layer, this is a safe generalization. However, annotators still have access to the contexts, in case of doubt. Sometimes a replacement

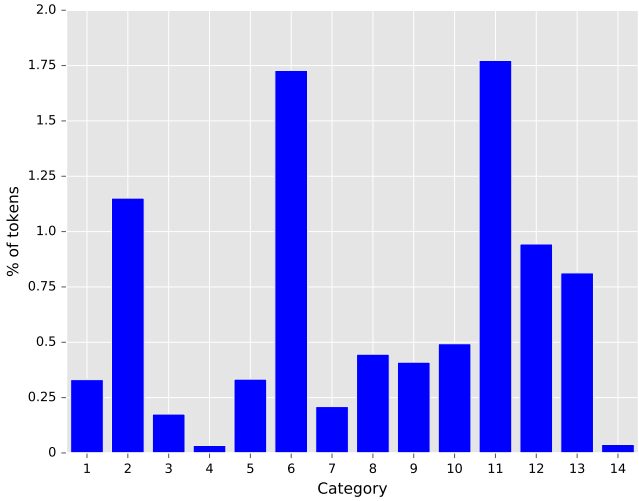


Figure 2: The distribution of the different categories. The percentages of tokens occuring in the categories with respect to all words in the corpus.

fits in two categories, e.g. diffffffff↦different (fits in category 7 and 9). In these cases it is up to the annotator to decide which category defines the replacement most. We only annotate the training part of the dataset, to keep the test data strictly for the final testing of a tuned model (note that there is no default development split). The train data can be used in a k-fold cross validation experiment to inspect the strengths and weaknesses of a model.

One annotator annotated all the 1,204 replacement pairs present in the training part of the Lexnorm2015 dataset. Additionally, a second annotator annotated a random shuffle of 150 replacements to test the inter-annotator agreement. All annotators are guided by the descriptions in Section 3. The annotators reached a Cohen's Kappa (Cohen, 1960) of 0.807, which indicates a near perfect agreement. There was no clear trend in the disagreements; the most common disagreement was between category 13 and 8, but this only occured three times. After annotation both annotators discussed and resolved the differently annotated pairs

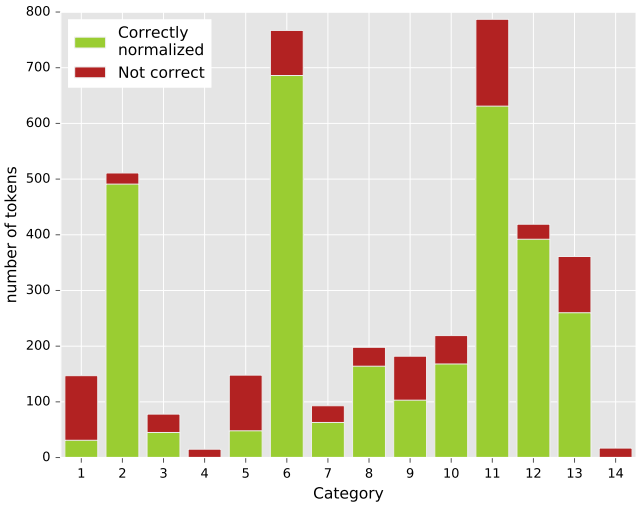


Figure 3: Performance of MoNoise on the different categories in a 10-fold experiment on the training part of LexNorm2015.

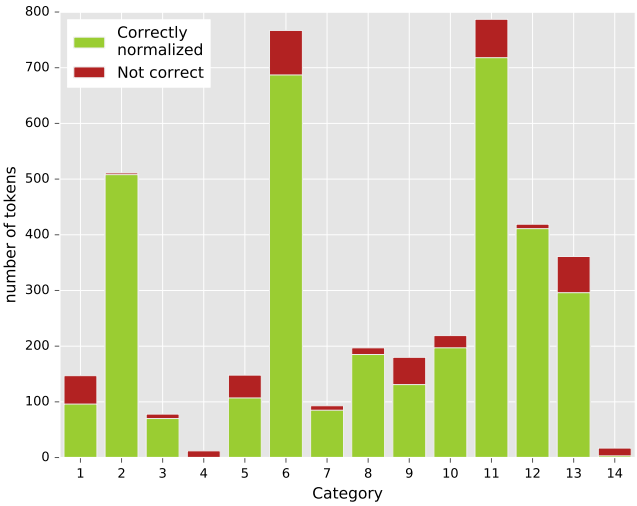


Figure 4: Performance of MoNoise on the different categories when using gold error detection.

and refined the description of the categories.
Figure 2 shows the distribution of the different categories. Most of the replacements are intentional word-word replacements (7-12); about half of these are phonetic transformations. Other popular categories are phrasal abbreviations and missing apostrophe.

5. Evaluation of Normalization Model

In this section we evaluate the normalization model MoNoise (van der Goot and van Noord, 2017) in more detail using our proposed taxonomy. This system is based on the idea that the normalization task consists of different normalization replacement actions. MoNoise generates normalization candidates for all words, and includes the original word as a candidate, so that it can decide whether or not to normalize when the candidates are ranked. It generates candidates using the out-of-the-box spelling correction system Aspell[1], combined with a word embeddings module and a lookup list generated from the training data. The candidates are then ranked in a random forest classifier using features from the generation modules combined with N-gram probabilities from a canonical dataset as well as a Twitter dataset. For more details we refer to the original paper (van der Goot and van Noord, 2017).
For maximum performance we train MoNoise using the Aspell bad-spellers mode. We run MoNoise in a 10-fold cross validation setup to get predictions for the whole training set. The absolute number of correctly normalized as well as the missed transformations are plotted in Figure 3. Note that besides these errors, 198 canonical words are wrongly normalized, so this still accounts for a large part of the errors. Two categories are not handled at all by MoNoise, namely split (4) and unknown (14). Besides these, the most difficult categories are typographical erros (1), phonetic transformations (11) and merge (5). Surprisingly, two of these categories consists of a lot of minor transformations (1, 11). Replacements beyond the word level are clearly difficult; a closer look at the typographical errors revealed that in these cases the original token is often ranked first by the normalization model. The categories that are almost completely resolved are missing apostrophe (2) and regular transformations (12). For missing apostrophe this is due to the relative restricted number of different replacements, whereas for regular transformations this is due to the fact that they are clearly in need of normalization, and their correct counterparts are quite similar (often in↦ing).
Figure 4 shows the performance per category when using gold error detection. The performance is substantially higher on almost all categories, confirming that error detection is still a relevant problem. Unsurprisingly, the categories split (4) and unknown (14) do not improve much, since the correct candidates are not generated by MoNoise, also the phrasal abbreviation category does not improve. This is because these are only solved by using the lookup list; this module does not improve by using gold error detection. Categories which previously performed well (2, 12) are now almost completely solved.

6. Conclusion

We proposed a taxonomy for normalization replacements, which can be used for detailed evaluation of normalization systems as well as the filtering of training data. While some categories can potentially overlap, the inter-annotator agreement indicates near perfect agreement. We tested a state-of-art normalization model to see which categories are still most problematic: typographical errors, phonetical errors and merging. On most of the other categories the system performs quite well. When using gold error detection, the performance goes up for almost all categories, indicating that this is still a far from solved problem. Interesting future work would be an extrinsic evaluation: how important are the different normalization categories for specific downstream tasks.
The annotation as well as the used scripts are available at: `https://bitbucket.org/robvanderg/normtax`

[1] `http://www.aspell.net`

Acknowledgements

We would like to thank Barbara Plank and Antonio Toral for discussions about the taxonomy. This work is part of the Parsing Algorithms for Uncertain Input project, funded by the Nuance Foundation.

Bibliographical References

Baldwin, T. and Li, Y. (2015). An in-depth analysis of the effect of text normalization in social media. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 420–429, Denver, Colorado, May–June. Association for Computational Linguistics.

Baldwin, T., de Marneffe, M.-C., Han, B., Kim, Y.-B., Ritter, A., and Xu, W. (2015a). Guidelines for English lexical normalisation. Technical report, Workshop on Noisy User-generated Text.

Baldwin, T., de Marneffe, M.-C., Han, B., Kim, Y.-B., Ritter, A., and Xu, W. (2015b). Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135, Beijing, China, July. Association for Computational Linguistics.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Kaljahi, R., Foster, J., Roturier, J., Ribeyre, C., Lynn, T., and Le Roux, J. (2015). Foreebank: Syntactic analysis of customer support forums. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1341–1347, Lisbon, Portugal, September. Association for Computational Linguistics.

Li, C. and Liu, Y. (2014). Improving text normalization via unsupervised model and discriminative reranking. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 86–93, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Mariana, V. R. (2014). The Multidimensional Quality Metric (MQM) framework: A new framework for translation quality assessment. *The Journal of Specialised Translation*.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., and Bryant, C. (2014). The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland, June. Association for Computational Linguistics.

Reynaert, M. (2008). All, and only, the errors: more complete and consistent spelling and ocr-error correction evaluation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA).

van der Goot, R. and van Noord, G. (2017). MoNoise: Modeling noise using a modular normalization system. *Computational Linguistics in the Netherlands Journal*, 7.

Yang, Y. and Eisenstein, J. (2013). A log-linear model for unsupervised text normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 61–72, Seattle, Washington, USA, October. Association for Computational Linguistics.

Gaining and Losing Influence in Online Conversation

Arun Sharma and Tomek Strzalkowski
ILS Institute, State University of New York – University at Albany
1400 Washington Ave, Albany, NY 12222 USA
asharma5@albany.edu, tomek@albany.edu

Abstract

In this paper, we describe a study we conducted to determine, if a person who is highly influential in a discussion on a familiar topic would retain influence when moving to a topic that is less familiar or perhaps not as interesting. For this research, we collected samples of realistic on-line chat room discussions on several topics related to current issues in education, technology, arts, sports, finances, current affairs, etc. The collected data allowed us to create models for specific types of conversational behavior, such as agreement, disagreement, support, persuasion, negotiation, etc. These models were used to study influence in online discussions. It also allowed us to study how human influence works in online discussion and what affects a person's influence from one topic to another. We found that influence is impacted by topic familiarity, sometimes dramatically, and we explain how it is affected and why.

Keywords: Natural Language Processing, Computational Sociolinguistics, Social Computing, Influence Detection, Text Analysis, Data Analytics

1. Introduction

This research was undertaken to help us understand how influential people behave in group conversations when they discuss the topics about which they have less knowledge. How does the topic of conversation affect their behavior in a group, specifically in an online conversation? Since a great deal of our communications occurs online, it is important to study and understand the sociolinguistic behavior of people who have high degree of influence in such environments.

For this research, we recruited participants in groups of 3 to 5 to discuss topics that they are all familiar with. Once we identified the most influential participant in each group, we moved them to a new group of 3 to 5 participants that discussed a different topic that our influencers were less familiar with. We collected interactions from both rounds of discussions as our data set. We automatically analyzed the data for various sociolinguistic factors and computed the degree of influence for each participant, using the DSARMD toolkit (Broadwell et al., 2012). Some of the factors considered included the rate of topic introduction, participation frequency, as well as patterns of language use, dialogue acts, references to people and other named entities (Named Entity Tagging), etc. These factors were further aggregated into predictions of sociolinguistic behavior in conversation, such as Argument Diversity, Disagreement Measure, Network Centrality, Topic Control, Involvement etc.

2. Related Work

Our objective is to study how people who are influential in one context retain or lose their influence in another context. We focused on the topic of conversation, and specifically the participants' degree of interest and familiarity with it. Research on measuring influence is a relatively new area in computational linguistics and natural language processing; although it has some commonalities with automated sentiment analysis and dialogue understanding, which have much longer history. The key recent work that the present study draws on is (Strzalkowski et al., 2013) and (Broadwell et al, 2012). This research developed and validated a computational approach for modeling sociolinguistic behaviors in conversation within small groups of up to 30 participants. Furthermore (Shaikh et al. 2013) expanded this work to track influence dynamics in conversation and how people's opinions change under influence. Our current research has utilized the approach and methods discussed in these works. Specifically, we use the components of sociolinguistic toolkit to compute and compare influence of all participants in our experiments and then determine how it relates to their knowledge of the discussion topic.

3. Methodology (Hypotheses, Study Design and Procedures)

The following sections describe how the study was undertaken, online survey formed, and responses collected.

3.1 Research Hypothesis

Our research starts with the following hypothesis:

"Individuals who have a high degree of influence within a group discussing a topic they are knowledgeable about, remain influential when moved to another group that discusses a different topic they are less knowledgeable about."

If this hypothesis could be confirmed, it would suggest that personal capabilities and behavioral profile largely determine a person's influence in conversation. Conversely, if the hypothesis is rejected, it would mean that one's influence is partly conditioned upon the topic of conversation and the relative familiarity with it. While the experiment described here is on a relatively small scale, the result would have consequences on influence modeling in general, including in large scale social networks where information diffusion is often related to a person's degree of influence. (e.g., Hofman et al, 2017)

3.2 Description of the Experiment

In the first part of our study, we asked the recruited participants to answer open-ended questionnaire about possible discussion topics of their interest. The questions focused on how much they like to talk about a subject of their interest. How much knowledge they think they have in that area of their interest.

After the data analysis of the first phase we selected the respondents based on their answers on the questionnaire and placed them in groups of 3 to 5 people to discuss a

selected topic that matches their interests. These online discussions took around 1 to 1.5 hours.
In the second phase, we moved participants between groups so that the most influential participant from each phase 1 group was placed in a new group that was to discuss a topic they were less familiar with. Another round of discussions of 1-1.5 hours duration was conducted in each new group.
After all data was collected, the transcript of each discussion was analyzed by the DSARMD software that assigns a degree of influence to each participant in a group (Broadwell et al., 2012). We compared the performance of each participant in phase 1 and phase 2 groups relative to their familiarity with the discussion topics. We should note that the participants generally did not know one another and were only aware of their anonymized user ids, which were assigned by the investigator. The website we used to collect the chat logs is Chatzy.com, which allows to create private chat rooms that safely store chat data. The participant chat user id, timestamp of when they post a message and the content of the message were saved as part of the logs of discussion. Participants surveys were created using available Surveymonkey.com. Survey Monkey forms were only used to create and display the questionnaire, the responses were redirected to be saved on our server. This data was then analyzed to prove or disprove the hypothesis.

3.3 Survey Formation and Topic Selection

To collect student responses a survey was formed. First topic selection was done by coming up with specific topics for the survey. For each topic, the participants rated their interest and familiarity on a 5-point Likert scale (Table 1). The topics were selected from areas where participants (graduate students at a public university) likely had opinions and knowledge. Some of the topics selected were from Sports, Movies, Academics, Current Affairs etc.

Not at all knowledgeable at all	1
Less than an average person	2
As knowledgeable as anybody	3
Probably more than most people	4
A lot more than most people	5

Table 1: *Measuring Knowledge for a topic*

The participants who selected option 4 or 5 were then asked if they would be interested in a group discussion for each topic (Table 2).

Not interested at all	1
Not particularly interested	2
Moderately interested	3
Quite interested	4
Very much interested	5

Table 2: *Options for Interest in Group Discussion*

The participants who indicated an interest in discussion, were further asked to write in a few words what aspects of the topic they would like to talk about. They were also asked if they would be interested in leading a discussion.
The survey covered 10 topics with up to 4 questions per topic, and it generally took approximately 15 minutes to complete.

3.4 Surveys and Group Formation

This study involved 54 participants, all graduate students at University at Albany. Data collected from the surveys were collated per topic and per participant in order to (1) identify topics that were of interest to at least some participants; and (2) form phase 1 groups based on topic interest. We required at least 3 participants per group.
After analysis of the Survey data collected from participant's 10 discussion groups were formed.
The conversation data from the discussions were collected in the text format for further analysis and to compute influence scores for participants, and rank them by the degree of influence. The most influential participants in each group were re-assigned to participate in another online group discussion this time on a topic they were less knowledgeable about. They were placed with people who had indicated they were knowledgeable about this new topic and were interested to discuss it. Again, the conversation was recorded, influence scores were computed for each participant, and participants were ranked by their degree of influence.

3.5 Demographic Makeup

Taking into consideration the privacy, limited personal data was collected from individual participants of the study. The overall demographic makeup of 54 participants was 70.37% of the participants were males and 29.63% participants were female.

4. Analysis of Online Discussion Data

The online conversation was analyzed automatically by a suite of tools developed under the DSARMD Project (Broadwell et al, 2012).

4.1 Finding Influencers based on predictions of Social Behavior in Conversation

The participants were ranked based on their influence, thus helping us select the influencer in each group along with scoring of every participant on various component metrics such as Network Centrality, Disagreement Measure, Topic Control, Involvement, among others, that contribute to the assessment of the degree of influence (Strzalkowski et al., 2013). All these attributes were computed for each conversation and participant. We describe the key metrics below. For a more detailed explanation, and how the scores are combined, the reader is referred to (Broadwell et al., 2012) and (Strzalkowski et al., 2012).
Network Centrality (NC): It is the measure of degree to which a participant in the conversation is a "center" of communication in that group. A participant has a high degree of Network Centrality when other participants address more of their utterances towards her or him, and whose topics are most discussed by others.
Disagreement Measure (DM): Disagreements with others is a way to control the topic of conversation by way of identifying or correcting what the participant sees as a problem. The more disagreement a participants shows, relative to other participants, the higher his/her Disagreement Measure.
Topic Control (TC): Topic Control reflects a conversational behavior where participants attempt to impose a topic of conversation. This may be accomplished by introduction of preferred topics that are subsequently discussed at length by others, or by successfully continuing discussion of selected topics. The ability to introduce topics and make others talk about them indicate the degree of topic control in the conversation.

Involvement (Inv): This behavior reflects the degree of engagement in a discussion measured as the proportion of conversational turns contributed by each participant.
Argument Diversity (AD): Participants who use a broader range of arguments in the conversation has a higher degree of Argument diversity. This measure includes amongst others, the size of one's vocabulary, usage of specialized terminology, and citing authoritative sources.

4.2 Results of Phase 1 experiments

Group discussions were conducted where an influential participant was identified in every group. We note again that in Phase 1 all groups were composed of participants who identified themselves as knowledgeable about the discussion topic. Below we show details of Influence (Inf) and the component metrics for two Phase 1 groups. Other groups had similar distributions of scores.

Group 1: U.S. Immigration System and Reforms
Number of Participants: 4

Table 3 shows performance of participants in Group 1, with Person 3 identified as the Influencer.

User	Inf	NC	DM	TC	AD	Inv
Person1	0.41	0.103	0.397	0.125	0.291	0.410
Person2	0.19	0.226	0.170	0.125	0.275	0.334
Person3	**0.70**	**0.219**	**0.379**	**0.223**	**0.307**	**0.347**
Person4	0.08	0.138	0.023	0.055	0.103	0.170

Table 3: *Participant's performance in Group 1 discussion. Inf is overall influence score; the other columns show scores for component measures: Network Centrality (NC), Disagreement Measure (DM), Topic Control (TC), Argument Diversity (AD), and Involvement (Inv)*

Group 2: Globalization
Number of Participants: 5

Table 4 shows performance of participants in Group 2, with Person 2 and Person 5 both identified as influencers. Note that their Inf scores are very close.

User	Inf	NC	DM	TC	AD	Inv
Person1	0.15	0.232	0.095	0.127	0.193	0.195
Person2	0.45	0.346	0.257	0.288	0.318	0.376
Person3	0.04	0.0	0.095	0.002	0.104	0.196
Person4	0.33	0.185	0.215	0.101	0.237	0.319
Person5	0.46	0.235	0.334	0.085	0.138	0.178

Table 4: *Participants' performance in Group 2 discussion*

Similarly, the influencers were identified from the remaining online group discussions. In most cases, a single person was identified as the influencer. In two cases, no one was selected (all scores were closely distributed around the mean).

4.3 Results of Phase 2 experiments

In the second phase of the study, the persons identified as influential in each Phase 1 group was placed in a new group where the topic of discussion was less familiar to them. To be clear: the new group was composed of participants who were knowledgeable about the discussion topic, although not particularly influential (based on their Phase 1 performance); however, our Phase 1 influencer now assigned to this group was less familiar with the topic, according to the participant's self-assessment in the survey. We also had to discard two Phase 1 group discussion datasets due to relatively low level of participation by the members of these groups, and thus system's inability to identify influencers.
The influence scores were computed again along with their components (see the next section). These influence scores should be analyzed along with all the component scores, in order to see the primary source of each person influence or lack thereof.

4.4 Outcomes

The following results were obtained after the second round of discussions. Every influencer was evaluated using the same measures and the results were studied. Below we show how selected Phase 1 Influencers (1 through 6) fared in Phase 2 discussions.

Influencer 1
Knowledgeable about: U.S. Immigration Reform
Less Knowledgeable about: Science-Fiction Movies

Table 3 shows performance of Influencer 1 (Person 3) in the first-round discussion discussing US Immigration Reform. We note that Influencer 1 had high scores on all component measures, and the highest scores in topic control and argument diversity.

User	Inf	NC	DM	TC	AD	Inv
Person1	0.42	0.140	0.458	0.104	0.218	0.252
Person2	0.16	0.266	0.091	0.108	0.217	0.313
Person3	0.13	0.066	0.036	0.078	0.218	0.261
Person4	0.69	0.377	0.402	0.276	0.318	0.349

Table 5: *Performance of Influencer 1 (Person 3) in the second-round group discussing science-fiction movies.*

Table 5 shows Influencer 1's performance in the second-round discussion. This time, our influencer (Person 3 again) was amongst the least influential people in this group with the lowest network centrality score, and the lowest topic control score – both related to the topic familiarity. Note that the involvement measure and argument diversity, which are more related to behavioral attributes of a person, remain relatively high, but cannot compensate for the loss of topical scores.

Influencer 2
Knowledgeable about: Globalization
Less Knowledgeable about: Science-Fiction Movies

This participant was given the ID "Person 5" in first round -group (see Table 4) and "Person 3" in the second-round group (Table 6). As an influencer in the first-round discussion, Person 5 had highest disagreement measure (DM), the second highest score in network centrality (NC), a moderate amount of topic control, and argument diversity and was less involved.

User	Inf	NC	DM	TC	AD	Inv
Person 1	0.528	0.087	0.295	0.170	0.344	0.396
Person 2	0.440	0.305	0.336	0.165	0.259	0.363
Person 3	0.268	0.098	0.285	0.083	0.166	0.208
Person 4	0.118	0.089	0.027	0.059	0.197	0.262

Table 6: *Performance of Influencer 2 (Person 3) in the second-round group discussing science-fiction movies.*

Influencer 2 was not quite influential in the second-round group. While the loss of influence was not as dramatic as for Influencer 1, the main reason here was the loss of Network Centrality status. Note that Influencer 2, unlike Influencer 1, was not a strong driver of conversation topic (Topic Control measure); instead his influence derived mostly from other people deferring to his superior topic expertise, which of course was not present in round 2.

Influencer 3
Knowledgeable about: Globalization
Less Knowledgeable about: Science Fiction Movies

This participant was given the ID "Person 3" in the first-round group (Table 7) and "Person 2" in the second-round group (Table 8)

User	Inf	NC	DM	TC	AD	Inv
Person 1	0.204	0.355	0.060	0.135	0.259	0.246
Person 2	0.272	0.235	0.296	0.200	0.402	0.507
Person 3	0.860	0.094	0.592	0.120	0.249	0.306

Table 7: *Performance of Influencer 3 (Person 3) in the first-round group discussing globalization.*

Influencer 3 had the highest disagreement measure (DM), was second in involvement and low in all other measures.

User	Inf	NC	DM	TC	AD	Inv
Person 1	0.539	0.321	0.321	0.316	0.504	0.457
Person 2	**0.321**	**0.199**	**0.321**	**0.222**	**0.225**	**0.311**
Person 3	0.596	0.478	0.357	0.061	0.258	0.364

Table 8: *Performance of Influencer 3 (Person 2) in the second-round group discussing science-fiction movies. Note that Person 1 and Person 2 were the most influential participants in this group as they were very close to each other in influence score.*

In the second round discussion, Influencer 3 (as Person 2) was the least influential in this group. While he ranked 2nd in disagreement measure (DM) and topic control (TC), his network centrality score (NC) was the lowest and so was Involvement (Inv). In this case, the loss of influence is attributed to the relative decline of disagreement and network centrality measures.

Influencer 4
Knowledgeable about: Criminal justice system in the U.S.
Less Knowledgeable about: Science Fiction Movies

This participant was given the ID "Person 1" in the first-round group (Table 9) and "Person 2" in the second round group (Table 10)

User	Inf	NC	DM	TC	AD	Inv
Person 1	**0.819**	**0.211**	**0.465**	**0.238**	**0.433**	**0.461**
Person 2	0.233	0.219	0.153	0.117	0.181	0.181
Person 3	0.118	0.171	0.037	0.079	0.158	0.195
Person 4	0.275	0.321	0.089	0.160	0.216	0.312

Table 9: *Performance of Influencer 4 (Person 1) in the first-round group discussing criminal justice system.*

Influencer 4 had highest disagreement measure (DM) in the first round group. He also had highest topic control score (TC), highest involvement score (Inv) and argument diversity (AD). Person 1 was clearly the most dominant in this group.

In the second round discussion, Influencer 4 was only the 3rd influential person in the group. He had lowest score on the disagreement measure (DM), 2nd in network centrality scores (NC), while still a high topic control (TC) and highest involvement (Inv) and argument diversity (AD) in the group.

User	Inf	NC	DM	TC	AD	Inv
Person 1	0.250	0.360	0.102	0.232	0.294	0.321
Person 2	**0.257**	**0.329**	**0.097**	**0.221**	**0.327**	**0.371**
Person 3	0.274	0.051	0.198	0.053	0.171	0.215
Person 4	0.645	0.196	0.595	0.067	0.163	0.264

Table 10: *Performance of Influencer 4 (Person 2) in the first-round group discussing science fiction movies.*

Nonetheless, the group was now dominated by Person 4 who, with the very high disagreement score clearly managed to block most of Person 2's attempts at topic control, in spite of his relatively high network centrality.

Influencer 5
Knowledgeable about: Science Fiction Movies
Less Knowledgeable about: U.S. Immigration Reform
This participant was given the ID "Person 1" in the first and also in the second round group (Tables 11 and 12).

User	Inf	NC	DM	TC	AD	Inv
Person 1	**0.54**	**0.321**	**0.321**	**0.316**	**0.504**	**0.457**
Person 2	0.32	0.199	0.321	0.222	0.225	0.311
Person 3	0.59	0.478	0.357	0.061	0.258	0.364

Table 11: *Performance of Influencer 5 (Person 1) in the second-round group discussing science-fiction movies.*

Influencer 5 was one of the two influential participants in the group (Person 3 was the other). He had highest argument diversity (AD), topic control (TC), and involvement (Inv) scores, while the other influencer had top scores in network centrality (NC) and disagreement (DM). This group is somewhat marginal in the sense that no single participant dominated the conversation.

User	Inf	NC	DM	TC	AD	Inv
Person1	**0.290**	**0.402**	**0.179**	**0.245**	**0.362**	**0.414**
Person2	0.033	0.0	0.0	0.0	0.074	0.066
Person3	0.662	0.124	0.568	0.122	0.316	0.335
Person4	0.412	0.273	0.226	0.170	0.203	0.069

Table 12: *Performance of Influencer 5 (Person 1) in the second-round group discussing US Immigration Reform.*

In the second-round group, Influencer 5 was a distant third in influence ranking. While he retained high scores in network centrality (NC) and involvement (Inv), other metrics have fallen significantly, esp. disagreement (DM).

Influencer 6
More Knowledgeable about: Science Fiction Movies
Less Knowledgeable about: U.S. Immigration Reform

This participant was given the ID "Person 4" in the first-round group and "Person 3" in the second-round group.

User	Inf	NC	DM	TC	AD	Inv
Person1	0.42	0.140	0.458	0.104	0.218	0.252
Person2	0.16	0.266	0.091	0.108	0.217	0.313
Person3	0.13	0.066	0.036	0.078	0.218	0.261
Person4	**0.69**	**0.377**	**0.402**	**0.276**	**0.318**	**0.349**

Table 13: *Performance of Influencer 6 (Person 4) in the first-round group discussing science-fiction movies.*

Influencer 6 had not only the highest influence score in this group, but he had the highest scores on all component

measures. Clearly, this person was extremely good at discussing science fiction movies. And yet, this dominating influence all but vanished when he was part of the second-round group discussing immigration matters (Table 14).

User	Inf	NC	DM	TC	AD	Inv
Person1	0.34	0.425	0.361	0.343	0.414	0.435
Person2	0.17	0.096	0.180	0.104	0.28	0.339
Person3	**0.04**	**0.035**	**0.011**	**0.013**	**0.070**	**0.335**
Person4	0.11	0.067	0.361	0.044	0.204	0.069

Table 14: *Performance of Influencer 6 (Person 3) in the second-round group discussing U.S. Immigration Reform.*

In the second-round discussion, Influencer 6 had the lowest influence score, and moreover he had the lowest scores on all component measures. This is a remarkable case of a complete loss of influence, which is clearly not due to the lack of participation: we note that Person 3 remains highly involved in discussion (high Inv score), but her impact is marginal.

In all cases covered by this study, a significant loss of influence was noted between round 1 and round 2. We discuss our observations next.

5. Conclusion

Based on the analysis of data that was collected during this study, we can make the following observation: People who are influential in group discussions on familiar subjects, *lose their influence* when placed in a group discussing a less familiar topic. All the influential participants saw their influence decline, often so dramatically that they become the least influential participants in a group.

Interestingly, the Involvement component score of the Influencer score matrix remains approximately the same for 4 out of 6 the top influential people from round 1 to round 2. This suggests that the decline in influence is not simply explained by the lack of participation. The Topic Control score declines in 5 out of 6 instances, which is a significant factor in the loss of influence. The Disagreement Measure has also decreased, which is another key factor. The other measures, Network Centrality and Argument Diversity showed mixed results, increasing in some cases and decreasing in others. This leads us to conclude that the loss of influence associated with topic change is largely due to loss of Topic Control, and the participants' inability to take strong positions that others may endorse even when they initially disagree with them.

Thus, based on our analyses of the group discussions we further conclude that the hypothesis **"**Individuals who have a high degree of influence within a group discussing a topic they are knowledgeable about, remain influential when moved to another group that discusses a different topic they are less knowledgeable about" is not supported by the data that we have. On the contrary, we have shown that when influential people are moved to groups where unfamiliar topics are discussed, their influence declines, sometimes significantly. This loss is attributed to the decreased scores of Topic Control and Disagreement measures, which are directly associated with the topical knowledge; however, this loss is not compensated by an increase on other measures such as Involvement and Network Centrality.

More extensive study and experiments are required to study this phenomenon more conclusively and an effort must be made to include a more varied group of participants to made data more representative and study the influence in online conversations. Also, other domains like online networking websites and discussion forums should be looked at more closely and analyzed in future studies to obtain a more representative set of data for studying influence in online world.

6. References

Shaikh, S., Strzalkowski, T., Stromer-Galley, J., Broadwell, G. A., Taylor, S., Liu, T., Boz, U. (2012). Modeling Influence in Online Multi-Party Discourse. 2012 Second International Conference on Cloud and Green Computing. doi:10.1109/cgc.2012.94

Strzalkowski, T., Shaikh, S., Liu, T., Broadwell, G.A., Stromer-Galley, J., Taylor, S. M., Ravishankar, V., Boz, U., Ren, X.: Influence and Power in Group Interactions. SBP 2013: 19-27

Broadwell, G. A., Stromer-Galley, J., Strzalkowski, T., Shaikh, S., Taylor, S., Liu, T., Webb, N. (2012). Modeling Sociocultural phenomena in discourse. Natural Language Engineering,19(02),213-257. doi:10.1017/s1351324911000386

Shaikh, S., Strzalkowski, T., Taylor, S., Lien, J., Liu, T., Broadwell, G. A., Peshkova, Y. (2015). Understanding Cultural Conflicts using Metaphors and Sociolinguistic Measures of Influence. Proceedings of the Third Workshop on Metaphor in NLP. doi:10.3115/v1/w15-1408

Strzalkowski, T., Shaikh, S., Ravishankar, V., Stromer-Galley, J., Crowley. J, Lin, C.S. (2013). Topical Positioning: A New Method for Predicting Opinion Changes in Conversation. Proceedings of the Workshop on language in Social Media (LSAM 2013), pages 41-48.

Strzalkowski, T., Broadwell, G.A., Stromer-Galley, J., Shaikh, S., Taylor, S., Webb, N.: Modeling socio-cultural phenomena in discourse. In: Proceedings of the 23rd International. Conference on Computational Linguistics, Beijing, China (2010)

Strzalkowski, T., Broadwell, G.A., Stromer-Galley, J., Shaikh, S., and Taylor, S. (2012) Modeling Leadership and Influence in Online Multi-Party Discourse. COLING-2012 Conference, Mumbai, India.

Hofman, J.M., Sharma, A., and Watts, D.J.. (2017). Prediction and explanation in social systems. Sciencemag.org. *Science* 03 Feb 2017: Vol. 355, Issue 6324, pp. 486-488 DOI: 10.1126/science.aal3856

Arap-Tweet: A Large Multi-Dialect Twitter Corpus for Gender, Age and Language Variety Identification

Wajdi Zaghouani, [1] **Anis Charfi**[2]
[1] College of Humanities and Social Sciences, Hamad Bin Khalifa University, Qatar
[2] Information Systems Program, Carnegie Mellon University Qatar
E-Mail: wzaghouani@hbku.edu.qa, acharfi@qatar.cmu.edu

Abstract

In this paper, we present Arap-Tweet, which is a large-scale and multi-dialectal corpus of Tweets from 11 regions and 16 countries in the Arab world representing the major Arabic dialectal varieties. To build this corpus, we collected data from Twitter and we provided a team of experienced annotators with annotation guidelines that they used to annotate the corpus for age categories, gender, and dialectal variety. During the data collection effort, we based our search on distinctive keywords that are specific to the different Arabic dialects and we also validated the location using Twitter API. In this paper, we report on the corpus data collection and annotation efforts. We also present some issues that we encountered during these phases. Then, we present the results of the evaluation performed to ensure the consistency of the annotation. The provided corpus will enrich the limited set of available language resources for Arabic and will be an invaluable enabler for developing author profiling tools and NLP tools for Arabic.

Keywords: Multi-dialectal corpus, Author Profiling, Social media

1. Introduction

Arabic is a challenging language when it comes to building Natural Language Processing tools and applications. In fact, the complexity of Arabic is present at the various levels of linguistic representation (phonology, orthography, morphology, and syntax). Even though there were some advances in the field of Arabic Natural Language Processing, the Arabic language is still lagging behind other languages such as English in terms of availability of the required resources to address author profiling (Rosso et al., 2018; Zaghouani, 2014).

In fact, a person's language use reveals much about their profile. However, research on author profiling has always been constrained by the limited availability of training data, since collecting textual data with the appropriate meta-data requires a significant collection and annotation effort. For every text, the characteristics of the author have to be known in order to successfully profile the author. For the Arabic language, to the best of our knowledge, there is no corpus freely available for the detection of age, gender and dialectal variety. Most of the existing corpora are available for English or other European languages (Celli et al., 2013). Having a large amount of data remains the key to achieving reliable results in the task of author profiling.

This paper presents the work carried out within the framework of the Arabic Author Profiling Project (ARAP),[1] a research project funded by Qatar National Research Fund. This project aims at developing author profiling resources and tools for the Arabic language and using them in the context of cyber-security. More specifically, author profiling in the context of ARAP could be useful for forensic investigations to narrow the set of potential authors when receiving a threat message. While a few research efforts on author profiling have recently started in Europe and the USA, there is extremely little research that targets the Arabic language.

Within the context of the ARAP project, we built the Arap-Tweet corpus (Zaghouani & Charfi, 2018a), a multi-dialectal annotated corpus that can be used for author profiling, stylometry research, and many other applications. For instance, this kind of resources could be useful in studying Arabic dialects from a linguistics perspective (e.g., computational dialectology).

Twitter offers the opportunity to gather large amounts of informal language from many individuals. We searched and collected Twitter profiles from 11 regions in the Arab world in order to cover the most distinct dialectal varieties. Once the data collected, processed and normalized, we started the annotation process using well-defined annotation guidelines to annotate the Tweets according to their dialectal variety, the gender of the user and the age within three categories (under 25 years old, between 25 and 34, and above 35). Finally, we evaluated the quality of the data by performing the inter-annotator agreement measures on a regular basis during the whole process.

In the remainder of this paper, we briefly review related work (Section 2) and report on the dialectal Arabic varieties (Section 3). Then, we present our corpus and the respective data collection and validation processes (Section 4). After that, we present our annotation guidelines and workflow (Section 5). Finally, we present the evaluation of the annotation quality (Section 6).

2. Related Work

In the context of corpus creation for the modern standard Arabic, there are several efforts (Habash, 2010). In fact, there are many monolingual and parallel corpora annotated with syntactic and semantic information such as the different iterations of the Penn Arabic Probanks (Diab et al., 2008; Zaghouani et al., 2010; Zaghouani et al., 2012) and treebanks (Maamouri et al., 2010). Many tools and methods were developed to deal with the morphology, disambiguation (Zaghouani et al., 2016c), the diacritization (Zaghouani et al., 2016b) and syntactic parsing (Habash, 2010).

[1] http://arap.qatar.cmu.edu

For Dialectal Arabic (DA), some limited efforts were made to create resources for some major Arabic dialects such as Egyptian and Levantine (Habash et al., 2013; Diab & Habash, 2007; Pasha et al., 2014). Within the framework of the Qatar Arabic Language Bank (QALB) project, a large-scale annotated corpus of users' comments was produced, dialectal words were marked (Zaghouani et al., 2014; Zaghouani et al., 2015; Zaghouani et al., 2016a.)

Al-Sabbagh and Girju (2010) presented a method to extract information from the Internet in order to build a Dialectal to Modern Standard Arabic lexicon. Chiang et al. (2006) created a parser for Dialectal Arabic that was trained on MSA treebanks. Similarly, Sawaf (2010) worked on processing Dialectal Arabic using the training data from the standard Arabic Penn Treebank while Salama (2014) created an automatically annotated large-scale multi-dialectal Arabic corpus collected from user comments on Youtube videos. Their corpus covers five regions the Arab world, namely: Egypt, Gulf, Iraqi, Maghrebi and Levantine.

Some other works such as Sajjad et al. (2013), Salloum and Habash (2013) and Sawaf (2010) used a translation of the dialectal Arabic to Standard Arabic as a pivot to translate to English. Boujelbane et al. (2013) created a dictionary based on the relation between Tunisian Arabic and MSA. Some other researchers followed crowdsourcing based approaches to create interesting resources such as the work of Zbib et al. (2012).

At the regional level, we noted limited efforts focused on dialect identification such as in (Habash et al., 2008; Elfardy & Diab, 2013; Zaidan & Callison-Burch, 2013).

As the Dialectal Arabic (DA) is becoming the language of informal online communication in emails, chats, SMS and in social media, we witnessed several efforts on creating different resources to help to build related tools and applications. However, most of these efforts were disconnected from each other and they have only focused on a limited number of dialects in the Arab world.

To the best of our knowledge, there are only a few resources available on author profiling for the Arabic language and for the dialectal Arabic. We found two projects and two resources related to that topic: Abbasi and Chen (2005), Estival et al. (2008), Mubarak and Darwish (2014), and Rangel et al. (2017).

Abbasi and Chen (2005) focused on author identification in English and Arabic web forum messages in order to do an analysis of the extremist groups web forum messages. Estival et al. (2008) built the Text Attribution Tool (TAT) to automate the analysis of texts for the purpose of author profiling and identification for English and Arabic E-Mails. Mubarak and Darwish (Mubarak & Darwish, 2014) built a Twitter dialectal Arabic corpus from four different Arabic countries using the geolocation information associated with Twitter data. More recently, Bouamor et al. (2018) and Habash et al. (2018) created MADAR, an Arabic dialect corpus and lexicon covering dialects of various cities across the Arab world.

In the context of our ARAP project, co-organizers of the Author Profiling Share task PAN 2017[2] presented a dialectal Arabic corpus from four different regions (North Africa, Egypt, Levantine and Gulf). The corpus data were annotated with respect to age, gender and dialect (Rangel et al, 2017).

For both Mubarak and Darwish (2014) and Rangel et al. (2017), the coverage was limited to only four countries out of 22 Arabic countries. In the ARAP project, we extend this coverage to all Arabid major dialects by covering 11 distinct dialects. The provided corpus can be used for applications in various domains such as cyber-security, business (e.g., for marketing and customer segmentation) and in healthcare (e.g., for suicide prevention).

3. Dialectal Arabic

The Arabic language used in social media and online is a mix of Modern Standard Arabic (MSA) and other regional dialectal varieties. For this reason, it is important to recognize this code-switching situation when studying the Arabic language produced by users online.

Arabic dialects are generally classified by regions such as in Habash (2010) who classified the Arabic major dialects into North African, Levantine, Egyptian and Gulf or into sub-regional classification (e.g., Moroccan, Tunisian, Algerian, Egyptian, Lebanese, Syrian, Jordanian, Qatari, Iraqi etc.). Figure 1 illustrates the different dialectal varieties in the Arab world across the different countries and borders.

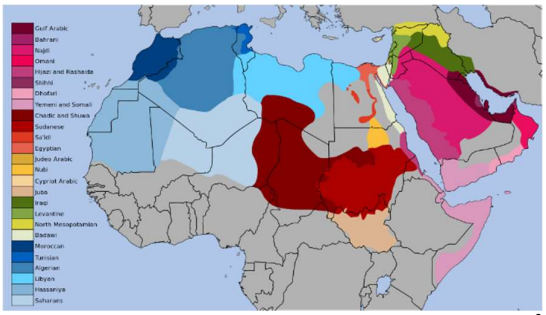


Figure 1: Arabic dialectal varieties in the Arab world[3]

The variation from one region to another poses many challenges to Natural Language Processing applications and therefore fine-grained resources and tools are required to address this issue. In fact, resources made for a given region cannot be used to train models for the dialect of a different region. While there are many similarities in the Arabic dialects, there is often a difference at the level of the lexicon, the morphology and the phonology.

[2] http://pan.webis.de/clef17/pan17-web/author-profiling.html

[3] Map distributed under a CC-BY 3.0 license from Wikipedia.

The sentence example in Table 1 illustrates similarities and differences between some dialectal varieties and standard Arabic. Within the ARAP project, we collected Tweets from 11 major Arabic regions or dialect groups instead of the traditional four groups in order to better represent the Arabic language used online in social media throughout the different regions.

Variety	I love reading a lot.
Standard Arabic	أنا أحب القراءة كثيرًا ʾanā ʾuḥibbu l-qirāʾata
Tunisian	āna nħəbb năqṛa baṛʃa أنا نحب نقرا برشة
Algerian	āna nħəbb nəqṛa bəzzāf أنا نحب نقرا بزاف
Moroccan	ana nbɣi bezzaf nəqṛa أنا نبي بزاف نقرا
Egyptian	ana baħebb el-ʔerāya awi أنا بحب القراءة أوي
Lebanese	āna ktīr bħebb il-ʔirēye أنا كتير بحب القراءة
Iraqi	āni kolish aħeb el-qra'a أني كلش أحب القراء
Qatari	ʔāna kulliʃ aħibb aqrā أنا كلش أحب أقراء

Table 1: A sample sentence in seven Arabic Dialects

4. Corpus Creation

In this section, we report on the corpus creation and annotation efforts we carried out to locate and crawl users for each dialect group. Before collecting and processing the data and in order to create the first Twitter multi-dialect annotated corpus of Arabic, we tried to cover as many dialects as possible while taking into consideration the available resources. We were able to collect Tweets from 11 Arabic regions representing a total of 16 countries from a total of 22 Arab countries that are members of the Arab league as shown in Table 2.

For each region, we collected the profiles of 100 users with at least 2000 Tweets with a minimum of 200K Tweets per region and a total of 2.4M Tweets corpus.

4.1 The Annotation Logistics

The collection and the annotation of a large scale corpus covering many regions of the Arab world such as the Arap-Tweet require the involvement of many annotators. In our project, the annotation effort is led by an annotation manager, and the team also consists of junior annotators and programmers.

The annotation manager is responsible for the whole annotation task. This includes collecting and cleaning the corpus, the annotation of a gold standard set to be used during the evaluation. Moreover, he is in charge of writing the annotation guidelines, hiring and training the annotators and monitoring the annotation progress and quality by performing the annotation evaluation on a weekly basis.

To ensure the suitability of the annotators for the task, we selected only university level annotators with a strong background knowledge of dialectal Arabic covering the regions to be annotated. Furthermore, the annotators were tested in a dialectal Arabic language screening test. Once selected, the annotators were trained over a period of two weeks by doing a pilot annotation task.

Dialect	Region
Moroccan	Morocco
Algerian	Algeria
Tunisian	Tunisia
Libyan	Libya
Egyptian	Egypt
Sudanese	Sudan
Lebanese	North Levant
Syrian	North Levant
Jordanian	South Levant
Palestinian	South Levant
Iraqi	Iraq
Qatari	Gulf
Kuwaiti	Gulf
Emirati	Gulf
Saudi	Gulf
Yemeni	Yemen

Table 2: Dialects and regions covered in the corpus

4.2 Selecting and Crawling Users

When looking at Twitter public profiles, we found that some users may include information such as their real name, location and a short biography in their profile. However, their age and gender details are usually not shared as they are not required in the Twitter profile and as there are no explicit fields on Twitter for them. Our goal was to select a balanced set of users for each of the 11 dialect regions selected. We found the users by searching public Tweets containing specific seed words and expressions that are used only in one dialect. For example, the word برشة /barsha/ 'many' in Tunisian Arabic or the word وايد /wayed/ 'many' in Gulf Arabic. In order to identify the list of seed words for each dialect, we conducted with the annotators a comprehensive study to validate the seed words list as some seed words were common in more than one region. Moreover, the annotators were trained to identify if a given seed word was used in the user profile from a different region. For instance, in order to accurately identify the profile for each region using the seed words, we relied on multiple seed words occurrences in multiple Tweets from the same profile in order to validate correct region for the given profile.

Once the potential users identified for a given dialect, the list was reviewed manually by the annotators to confirm and match the identified users to their dialect. We tried to select the data as randomly as possible by avoiding well-known public figures from our list. This restriction led to a large annotation effort and resulted in a smaller user sample.

Using the Twitter API we collected tweets that contained typical dialectal distinct words generally used by speakers

of a given region. This allowed us to restrict the tweets to the selected region as much as possible. During a twelve weeks period, we sampled users according to this method.

We only included accounts with a minimum number of 2000 Tweets. For all users, we downloaded up to their last 3240 tweets (limit imposed by Twitter API). We excluded retweets from the data collection as these tweets were written by other people.

For the annotation, our annotators carefully analyzed the users' profiles and their tweets. They had to annotate the collected data with the age group, the gender, and the dialect. Whenever possible they also used external resources such as LinkedIn, Facebook, and the user's blog and web page. For profiles that had a photo the annotators used the photo to guess the age in addition to using the AI based Microsoft website How-Old.net.[4]

In order to produce a usable and clean corpus for the planned author profiling task, we documented our annotation guidelines (Zaghouani & Charfi 2018b) and we asked the annotators to only annotate users who meet the following requirements:

- The profile should belong to an actual person (e.g. not an association or a company).
- The profile should be publicly accessible.
- The profile should have at least 2000 tweets.
- The tweets should have been mostly written in the given dialect (from the list of the 11 dialects).
- The Tweets should not be written mostly in standard Arabic or any other language such as English or French (this requirement is validated manually by the annotator by going through the profile Tweets manually.)
- The profiles posting a lot of images and using applications to automatically post daily messages by bots were also filtered

During the profiles collection step, we noticed that a group of users decided to protect their account and make it private between the time of sampling and the time of data collection. In total, 1100 users were annotated (100 users per region).

4.3 Gender Annotation

The gender was annotated for 1100 persons. In some cases, the annotators were not able to identify the gender due to the lack of a profile photo or other identifying information. In such cases, the users were removed from our list and replaced by users of the same gender and from the same region.

The gender male/female ratio was almost equal for 7 regions while for 4 regions it was around 60% males and 40% females on average as for some Arab countries we noticed that Twitter was not widely used by females. The annotation of the gender was based in most cases on the name of the person or on his profile photo and in some cases on their biography or profile description.

4.4 Age Annotation

In order to annotate the users for their age, we used three categories: under 25 years, between 25 years and 34 years, and above 35 years. The age category was annotated for all 1100 accounts. The results separated by gender are shown in Table 3. There are more females in the young age group, while there are more men in the older age groups.

Age Group	Male	Female
Under 25	150	94
25 until 34	391	199
Above 35	126	140
Total	667	433

Table 3: Age/Gender Annotation Groups

For annotating the age, our annotators started by retrieving the real name of the twitter user if possible. If the user has a profile photo we asked the annotators to guess the age based on the photo and then to use the machine learning based Microsoft website How-Old.net. If there is no photo in the profile we asked them to check if the users have other web pages or social media accounts (e.g., on Facebook or LinkedIn). For LinkedIn users, the age group could be determined by checking the education history of the users if it is indicated. Also, for some Facebook users the age could be determined if they indicate the year at which they graduated from high school. If the annotators were not able to annotate the age we asked them to remove the respective users from our list and to replace them by other users from the same region and the same gender.

4.5 Dialect Annotation Task

As the dialect and the regions are known in advance to the annotators, we instructed them to double check and mark the cases in which the dialect used by a certain Twitter user appears to be from a different dialect group. This is possible despite our initial filtering based on distinctive regional keywords. We noticed that in more than 90% of the cases the profiles selected belong to the specified dialect group. For the 10% remaining, we observed many cases of people borrowing terms and expressions from other dialects.

4.6 Data Collection and Processing

Once we have the list of profiles ready for collection, we used Twitter Stream API and the geographic filter to make sure that the collected Tweets are within the specified region. The Twitter API restricts the maximum number of Tweets to be collected to 3240 per user. We wrote a Python script to automate the data collection effort for each of the 11 regions using the list of 100 profiles for each region.

As the data collected from social media is usually noisy despite the manual verification done by the annotators, we had to write a script to clean the collected Tweets from non-textual content such as images and URLs. Moreover, we

[4] https://how-old.net/

filtered all non-Arabic content from the Data. We also discarded all retweets as well as all tweets that have less than three words.

5. Annotation Evaluation

In order to evaluate the quality of the annotation, we used the standard Inter-Annotators Agreement (IAA) measures to find out to what extent the annotators are in agreement by using the provided annotation guidelines.

The three annotators involved in the project were given a sample of 110 accounts (10 per region) to be annotated by all in a blind way, without them knowing that this particular evaluation set is also given to their colleagues.

As in similar annotation projects, the inter-annotator agreement was measured using Cohen's kappa. For this task, we believe that a value above 0.75 could be considered acceptable. At the end of the evaluation, the average Kappa values obtained by the group of the annotators were: gender annotation (0.95), age group annotation (0.80) and dialect group identification (0.92) as shown in Table 4.

Task	Kappa Score
Gender Annotation	0.95
Age Annotation	0.80
Dialect Annotation	0.92

Table 4: Inter-annotator agreement in terms of average Kappa score; the higher the better

As expected the age identification task is a much more difficult task, especially with the absence of clear indicators. For the dialect identification task, some annotators were confused by some similarity that exists between some dialects such as the Moroccan dialect and the Algerian dialect and also by the Qatari dialect and some other Gulf dialects.

Overall, we believe that the annotation agreement is above the acceptable range for the gender and the dialect annotation tasks and it could be improved for the age annotation task.

6. Conclusion

We presented Arap-Tweet, a novel and large Arabic multi-dialect Twitter corpus for the Age, gender and dialect profiling. It covers 11 regions and 16 countries in the Arab world. This corpus could be used for tasks other than the identification of age, gender and dialect. For instance, with some extra annotation, it could serve for authorship attribution, sentiment analysis, deception detection and topic detection to cite a few. In the near future, we plan to provide the Arap-Tweet corpus for the research community and we hope to receive feedback from the community on the usefulness and potential applications of that resource.

7. Acknowledgements

This publication was made possible by NPRP grant 9-175-1-033 from the Qatar National Research Fund (a member of Qatar Foundation). The findings achieved herein are solely the responsibility of the authors.

8. Bibliographical References

Abbasi Ahemd. and Chen Hsinchun. (2005). Applying authorship analysis to extremist group web forum messages, IEEE Intelligent Systems, 20(5): 67–75.

Al-Sabbagh, Rania and Girju, Roxana. (2010). Mining the Web for the Induction of a Dialectical Arabic Lexicon. In LREC, Valetta, Malta.

Boujelbane, Rahma, Ellouze Khemekhem, Mariem, and Belguith, Lamia Hadrich. (2013). Mapping Rules for Building a Tunisian Dialect Lexicon and Generating Corpora. In Proceedings of the Sixth International Joint Conference on Natural Language Processing, pages 419–428, Nagoya, Japan.

Celli, Fabio, Pianesi, Fabio, Stillwell, David., and Kosinski, Michal. (2013). Workshop on computational personality recognition (shared task). In Proceedings of the Workshop on Computational Personality Recognition, Boston, USA.

Chiang, David, Diab, Mona, Habash, Nizar, Rambow, Owen, and Shareef, Safiullah. (2006). Parsing Arabic Dialects. In Proceedings of EACL, Trento, Italy.

Diab, Mona, and Habash, Nizar. (2007). Arabic Dialect Processing Tutorial. In NAACL.

Elfardy, Heba and Diab, Mona. (2013). Sentence Level Dialect Identification in Arabic. In Proceedings of the Association for Computational Linguistics, pages 456–461, Sofia, Bulgaria.

Estival, Dominique, Gaustad, Tanja, Hutchinson, Ben, Pham, Son Bao., and Radford, Will. (2008): Author Profiling for English and Arabic Emails. Natural Language Engineering, Cambridge University Press (2008).

Habash, Nizar. (2010). Introduction to Arabic natural language processing, volume 3. Morgan & Claypool Publishers.

Habash, Nizar, Roth, Ryan, Rambow, Owen, Eskander, Ramy, and Tomeh, Nadi. (2013). Morphological analysis and disambiguation for dialectal arabic. In Proceedings of NAACL-HLT, pages 426–432, Atlanta, Georgia, June.

Pasha, Arfath, Al-Badrashiny, Mohamed, Kholy, Ahmed El, Eskander, Ramy, Diab, Mona, Habash, Nizar, Pooleery, Manoj, Rambow, Owen, and Roth, Ryan. (2014). Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In Proceedings of LREC, Reykjavik, Iceland.

Rangel Francisco, Rosso Paolo, Potthast Martin, Stein Benno. Overview of the 5th Author Profiling Task at PAN 2017: Gender and Language Variety Identification in Twitter. In Proceedings of PAN 2017, Dublin, Ireland.

Rosso Paolo, Rangel Francisco, Hernández Farías Irazu,

Cagnina Leticia, Zaghouani, Wajdi, Charfi Anis. (2018). A survey on author profiling, deception, and irony detection for the Arabic language. Language and Linguistics Compass. https://doi.org/10.1111/lnc3.12275

Sajjad, Hassan, Darwish, Kareem, and Belinkov, Yonatan. (2013). Translating dialectal Arabic to English. In Proceedings of ACL, Sofia, Bulgaria.

Sawaf, Hassan. (2010). Arabic dialect handling in hybrid machine translation. In Proceedings of AMTA, Denver, Colorado.

Zaidan, Omar and Callison-Burch, Chris. (2013). Arabic dialect identification. Computational Linguistics.

Zbib, Rabih, Malchiodi, Erika, Devlin, Jacob, Stallard, David, Matsoukas, Spyros, Schwartz, Richard, Makhoul, John, Zaidan, Omar F., and Callison Burch, Chris. (2012). Machine translation of Arabic dialects. In Proceedings of NAACL-HLT, Montréal, Canada.

Zaghouani, Wajdi. (2014). Critical survey of the freely available Arabic corpora. In Proceedings of the 1st Workshop on Open-Source Arabic Corpora and Processing Tools, 9th International Conference on Language Resources and Evaluation, Reykjavik, Iceland.

Zaghouani, Wajdi, Habash, Nizar, Bouamor, Houda, Obeid, Ossama, Alqahtani, Sawsan, Diab, Mona and Oflazer. Kemal. (2016c). Filtering Dialectal Arabic Text in Two Large Scale Annotation Projects. The 2nd Workshop on Noisy User-generated Text (W-NUT), December 11 2016, Osaka, Japan.

9. Language Resource References

Bouamor, Houda, Habash, Nizar, Salameh, Mohammad, Zaghouani, Wajdi, Rambow, Owen, Abdulrahim, Dana, Obeid, Ossama, Khalifa, Salam, Eryani, Fadhl, Erdmann, Alexander and Oflazer, Kemal. (2018). The MADAR Arabic Dialect Corpus and Lexicon. In Proceedings of the 11th International Conference on Language Resources and Evaluation.

Diab, Mona, Mansouri, Aous, Palmer, Martha, Babko-Malaya, Olga, Zaghouani, Wajdi, Bies, Ann, Maamouri, Mohammed. (2008). A Pilot Arabic Propbank. In Proceedings of the International Conference on Language Resources and Evaluation (LREC'2008), Marrakech, Morocco, May 28-30, 2008.

Habash, Nizar, Rambow, Owen, Diab, Mona, and Faraj, Reem. (2008). Guidelines for Annotation of Arabic Dialectness. In Proceedings of the LREC Workshop on HLT & NLP within the Arabic world.

Habash, Nizar, Khalifa, Salam, Eryani, Fadhl, Rambow, Owen, Abdulrahim, Dana, Erdmann, Alexander, Faraj, Reem, Zaghouani, Wajdi, Bouamor, Houda, Zalmout, Nasser, Hassan, Sara, Al shargi, Faisal, Alkhereyf, Sakhar, Abdulkareem, Basma, Eskander, Ramy, Salameh, Mohammad, and Saddiki, Hind. (2018). Unified Guidelines and Resources for Arabic Dialect Orthography. In Proceedings of the 11th International Conference on Language Resources and Evaluation.

Maamouri, Mohamed, Bies, Ann, Kulick, Seth, Zaghouani, Wajdi, Graff, Dave and Ciul, Mike. (2010). From Speech to Trees: Applying Treebank Annotation to Arabic Broadcast News. In Proceedings of LREC 2010, Valetta, Malta, May 17-23, 2010.

Mubarak, Hamdy, and Darwish, Kareem. (2014). Using Twitter to collect a multi-dialectal corpus of Arabic. In Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP) (pp. 1-7).

Salama, Ahmed, Bouamor, Houda, Mohit, Behrang, Oflazer, Kemal. (2014). YouDACC: the Youtube Dialectal Arabic Comment Corpus. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14).

Zaghouani, Wajdi and Charfi Anis. (2018a). The Arap-Tweet Corpus version 0.8. Carnegie Mellon University Qatar, Doha, Qatar

Zaghouani, Wajdi and Charfi, Anis. (2018b). Guidelines and Annotation Framework for Arabic Author Profiling. In Proceedings of the 3rd Workshop on Open-Source Arabic Corpora and Processing Tools, 11th International Conference on Language Resources and Evaluation, Miyazaki, Japan.

Zaghouani Wajdi, Habash, Nizar, Obeid, Ossama, Mohit, Behrang, Bouamor, Houda, Oflazer. Kemal (2016a). Building an Arabic machine translation post-edited corpus: Guidelines and annotation. In Proceedings of the International Conference on Language Resources and Evaluation (LREC'2016).

Zaghouani, Wajdi, Bouamor, Houda, Hawwari, Abdelaati, Diab, Mona, Obeid, Ossama, Ghoneim, Mahmoud, Alqahtani, Sawsan, Oflazer, Kemal. (2016b). Guidelines and Framework for a Large Scale Arabic Diacritized Corpus. In Proceedings of the International Conference on Language Resources and Evaluation (LREC'2016).

Zaghouani, Wajdi, Habash, Nizar, Bouamor, Houda, Rozovskaya, Alla, Mohit, Behrang, Heider, Abeer and Oflazer, Kemal. (2015). Correction Annotation for Non-Native Arabic Texts: Guidelines and Corpus. In Proceedings of the 9th Linguistic Annotation Workshop, co-located with NAACL in Denver, Colorado, USA.

Zaghouani, Wajdi, Mohit, Behrang, Habash, Nizar, Obeid, Ossama, Tomeh, Nadi, and Oflazer, Kemal. (2014). Large-scale Arabic Error Annotation: Guidelines and Framework. in the Proceedings of the International Conference on Language Resources and Evaluation (LREC'2014),. Rejkavik, Iceland, 26-31 May 2014.

Zaghouani, Wajdi, Hawwari, Abdelaati and Diab, Mona. (2012). A Pilot PropBank Annotation for Quranic Arabic. In Proceedings of the first workshop on Computational Linguistics for Literature, NAACL-HLT

2012, Montreal, Canada.

Zaghouani, Wajdi, Diab, Mona, Mansouri, Aous, Pradhan, Sameer and Palmer, Martha. (2010). The Revised Arabic PropBank. In proceedings of the 4th Linguistic Annotation workshop (Association of Computational Linguistics), Uppsala, Sweden.

Transc&Anno: A Graphical Tool for the Transcription and On-the-Fly Annotation of Handwritten Documents

Nadezda Okinina, Lionel Nicolas, Verena Lyding
Eurac Research, viale Druso 1, 39100 Bolzano, Italy
nadezda.okinina@eurac.edu, lionel.nicolas@eurac.edu, verena.lyding@eurac.edu

Abstract

We present Transc&Anno, a web-based collaboration tool allowing the transcription of text images and their shallow on-the-fly annotation. Transc&Anno was originally developed in order to address the needs of learner corpora research so as to facilitate digitisation of handwritten learner essays. However, the tool can be used for the creation of any type of corpora requiring transcription and shallow on-the-fly annotation resulting in inline XML. Transc&Anno provides an intuitive environment that is explicitly designed to facilitate the transcription and annotation process for linguists. Transc&Anno ensures a high transcription output quality by validating the XML and only allowing predefined tags. It was created on top of the FromThePage transcription tool developed entirely with standard web technologies – Ruby on Rails, Javascript, HTML, and CSS. We adapted this open-source web-based tool to linguistic research purposes by adding linguistic annotation functionalities to it. Thereby we united the convenience of a collaborative transcription tool with its advanced image visualisation, centralised data storage, version control and inter-collaborator communication facilities with the precision of a linguistic annotation tool with its well-developed tag definition possibilities, easy tagging process and tagged-text visualisation. Transc&Anno is easily customisable, open source, and available on Github.

Keywords: transcription tools, annotation tools, learner corpora

1. Introduction

Linguistic resources are of fundamental importance for conducting linguistic research. Some of these resources are originally created in handwritten form but need to be digitised in order to be efficiently exploited. In the absence of automatic handwriting recognition systems capable of handling in a satisfactory fashion the inputted data, they have to be manually transcribed. In many cases, transcription can coincide with shallow annotation. Since the transcription and annotation process can be extremely laborious and time-consuming, Transc&Anno is designed to make transcription and annotation quick and intuitive. Transc&Anno was specifically developed in order to address the needs of learner corpora research (Glaznieks et al., 2014) with regard to the manual transcription of learner texts and their shallow on-the-fly annotation.

In the remainder of this paper, we present the Learner Corpus Infrastructure project that led to the creation of Transc&Anno (Section 2), review existing text transcription and annotation tools (Section 3), explain Transc&Anno's use and important features (Section 4), give some technical information (Section 5), point out future development possibilities and conclude the paper (Section 6).

2. The LCI Project

2.1 Overview

The Learner Corpus Infrastructure project (LCI) conducted at the Institute for Applied Linguistics at Eurac Research aims at creating a stable, systematic, and sustainable infrastructure for the collection, processing and maintenance of learner corpora (Nicolas et al., 2015). *"Computer learner corpora are electronic collections of authentic FL[1]/SL[2] textual data assembled according to explicit design criteria for a particular SLA[3]/FLT[4] purpose. They are encoded in a standardised and homogeneous way and documented as to their origin and provenance"* (Granger, 2002). They serve the needs of language acquisition studies and pedagogy development as well as help the creation of natural language processing tools such as automatic error detection and correction systems (Gamon et al., 2013) or automatic language proficiency level checking systems (Hasan et al., 2008). Due to their often handwritten text basis and non-standard language use, learner corpora are time consuming and difficult to create and therefore scarce. The LCI project implements the following learner corpora creation workflow which can be boldly divided into four steps: (1) transcription and basic on-the-fly error tagging, (2) highly elaborated manual and automatic annotation, (3) corpus statistics and exploration, (4) online corpus publication. One or more specific tools are used for each of these steps. Transc&Anno is developed for the purpose of the first step.

1 foreign language
2 second language
3 second language acquisition
4 foreign language teaching

2.2 Transc&Anno Objectives

Transc&Anno will be the first tool in the processing chain for the creation of digitised learner corpora at Eurac Research. Learner corpora created at Eurac Research are built from handwritten documents stored as digitised images. These text images have to be transformed into computer-readable text. In the absence of automatic handwriting recognition systems capable of accurately recognising collections of short samples of different handwritings containing multiple corrections, they are manually transcribed. While transcribing, transcribers also perform shallow on-the-fly error-tagging.

By shallow annotation, we designate string annotation of different sizes without discontinuities or references to other portions of text. On-the-fly annotation is done at the same time as the transcription. Experience has shown that on-the-fly annotation of elements that do not require deep linguistic knowledge saves time for linguists who afterwards perform detailed linguistic annotation. In that perspective, we do not aim at creating a fully fledged annotation tool that would allow to annotate any type of construction. It is not used for highly elaborated annotation which is a task handled in a subsequent step.

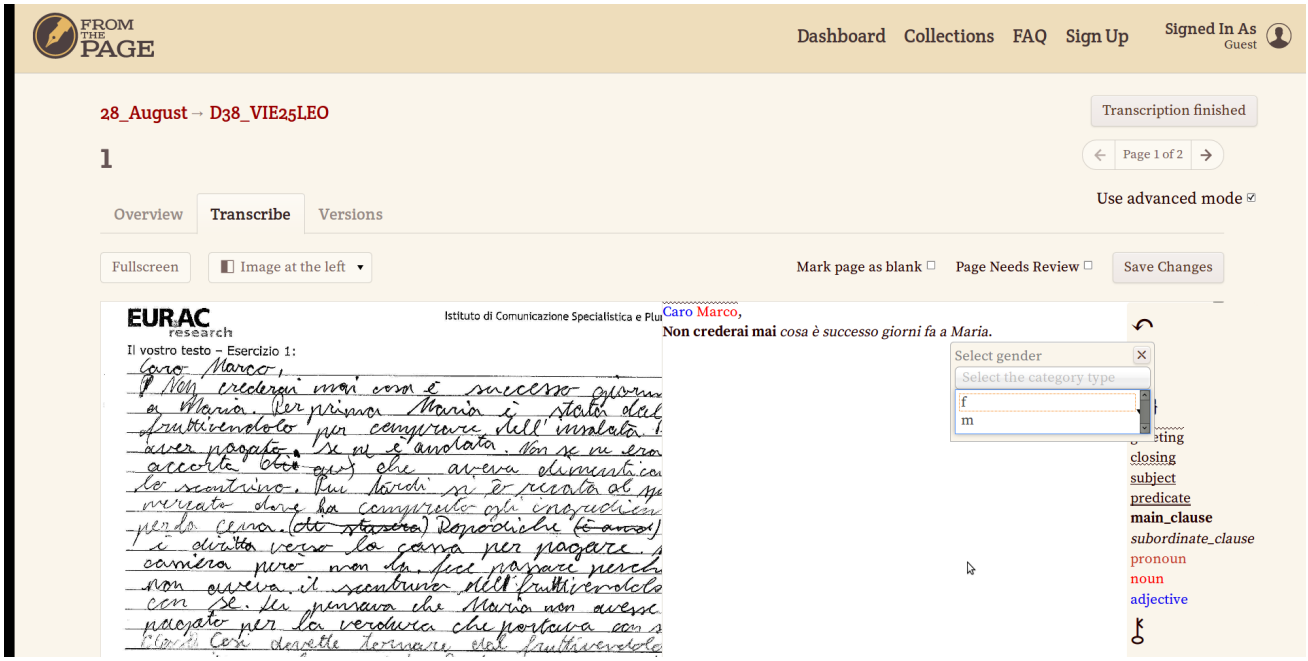


Figure 1: Transcription interface.

3. Related Work

According to our literature review there is no existent tool allowing both transcription of text images and their shallow on-the-fly annotation.

At present time, in learner corpus research, learner texts are typically transcribed using generic text editing tools such as XML Mind (Boyd et al., 2014), Oxygen (Tenford et al., 2006), Open Office Writer or Microsoft Word (Hana et al., 2010). These tools lack integrated collaboration facilities and are therefore a crucial drawback for big projects employing many transcribers and resulting in big quantities of data. Secondly, they do not have intuitive annotation facilities, because they were designed for general text-treatment purposes. That makes the annotation process even more laborious, time-consuming and error prone. Text editors such as Open Office Writer or Microsoft Word do not allow annotation tags' definition thus separating the transcription and annotation processes. They do not contain a text format verification functionality, so that the resulting transcription's format is entirely under the annotator's responsibility. XML editors allow tags' and text format definition, but they have to be hard coded into an XML schema with no front end facilitating this process.

In addition, a range of collaboration transcription tools exists within the field of digital humanities. Some of these tools have non-linguistic annotation functionalities. These tools were mostly conceived in order to create machine-readable versions of historical manuscripts and include, among others, the following projects: Europeana Transcribe[5], Transcribus[6], ProofreadPage[7], Transcribe Bentham[8], FromThePage[9], T-PEN[10] etc. The transcription projects are mainly organised by archives, libraries and museums that call for volunteers' help.

One of the main advantages of these crowdsourcing transcription tools is their web-based nature: they do not need local installation and can be easily used via a web browser. They also include advanced project managing functionalities and support text version control. Transcriptions are saved to a central database in a standardised format. Interpersonal communication is facilitated by the comment writing possibility. Many of the above mentioned tools support advanced image visualising thereby assuring the transcribers' comfort.

All these features can be useful for learner corpora transcription. However, as the final users of the digital texts produced via the above mentioned crowdsourcing tools are historians, the annotation systems of these tools are adapted to bibliographical, paleontological and historical needs. None of these tools offers annotation possibilities necessary for creating learner corpora.

Another tool worth mentioning is the multi-functional corpora editing platform TEITOK allowing document transcription and wiki-style annotation (Janssen 2016). The transcription module of this powerful platform was also inspired by FromThePage, but is not meant to evolve into a more user-friendly form.

Linguists also dispose of a very wide range of sophisticated annotation tools lacking the text transcription functionality such as MMAX (Müller et al., 2001) or WebAnno (Yimam et al., 2013) just to name a few. In creating Transc&Anno our goal is to unite both of these functionalities.

5 https://transcribathon.com
6 https://transkribus.eu/Transkribus
7 https://wikisource.org/wiki/Wikisource:ProofreadPage
8 http://blogs.ucl.ac.uk/transcribe-bentham
9 http://fromthepage.com
10 http://t-pen.org

4. Capabilities and Use

4.1 Basis Tool

Among all the existing tools we chose FromThePage as a basis for our application. First of all, FromThePage is open-source, continually updated and widely promoted by its creator. It is aimed at full-text transcription and not partial information collection or automatically recognised text proofreading. Moreover, it has a very well-developed user interface offering advanced visualisation customisation. It includes an annotation functionality serving historical research needs that can be transformed into a linguistic annotation functionality. FromThePage is built with standard web technologies and is easily customisable.

4.2 User Roles

Transc&Anno's internal structure, inherited from FromThePage, supposes the separation of different user functions. The database administration, the project preparation and the transcription tasks are distributed among different users having different user rights. Accordingly, there are three types of user rights: administrator, collection owner, and transcriber. The administrator is a technician who takes care of the database. The collection owner is the person who creates a collection: he uploads the scanned text to be transcribed and defines the annotation tags. One advantage of Transc&Anno is that the collection owner does not need to have technical knowledge in order to be able to set up the annotation scheme. Finally, the transcriber transcribes and annotates the texts, but is not allowed to modify the annotation system. The users' rights restriction protects sensitive information from accidental modification.

4.3 Setting Up Text Collections

Setting up a collection is the collection owner's responsibility. Documents intended for transcription are handwritten texts in the form of images. The image
loading part of the FromThePage tool has not been modified in Transc&Anno and is used as is. Text images for transcription can be loaded in PDF, PNG, GIF, JPG formats as well as in compressed archives. There are several views that show text images and transcriptions in format of different sizes.

Texts for transcription are organized in collections each of which can have its own annotation scheme and tagging categories.

4.4 Creating the Annotation Scheme

Transc&Anno contains an annotation scheme definition interface intended to be used by the collection owner. FromThePage possesses a menu system for categories creation that we extended in order to allow linguistic phenomena description. FromThePage offers the possibility to add, rename and delete a category. In addition to this, Transc&Anno lets the user define the category's style, description, attributes, its attributes' values and sequences.

The annotation system is defined for each collection of documents and consists of tagging categories and their attributes. The number of categories is unlimited. A category can have no attributes, one or many. An attribute may have a predefined set of values as well as allow the user to enter a new value. Such choice is made by the collection owner who designs the annotation system. Attributes of the same category may be independent or entertain causal relations between each other. Accordingly, the existence of one attribute can depend on the value of another attribute of the same category.

Let's take the example of a German *adjective* category. Depending on the value of its *number* attribute (*singular* or *plural* attribute value), defining the *gender* can be irrelevant. Indeed, plural adjectives have the same declension for all three genders (masculine, feminine and neutral) whereas singular adjectives do not. In linguistic annotations such interdependence between different characteristics is frequent, hence the need for the causal relations mechanism in the attributes' definition.

4.5 Customising Visualisation

FromThePage offers the transcribers some visualisation customisation features: while transcribing a page, the user can zoom in and out on the image text as well as choose its position in the browser window in respect to the transcription text.

FromThePage does not give any possibility to highlight tagged text, because it only allows wiki-style annotations that are performed by typing conventional characters. In order to give visual support to the new annotation functionality we added to Transc&Anno, we included a text highlighting feature. Tagged text highlighting is important for the annotator because it gives him the possibility to spot the existing tags in order to be able to modify or delete them. Each category can be assigned its own text style that will be visible during annotation. The text style includes a font colour, a font style and an underlining style offered by the CSS language. As different tags can overlap, the existence of these three types of highlighting lets two or three tags be visible at the same place. CSS doesn't allow coexistence of more than three styles in the same text fragment, but we intend to overcome this limitation in the future versions of Transc&Anno.

4.6 Transcription and Annotation Process

In this section, we describe the annotation process using Transc&Anno. Instead of typing wiki-style special characters as it is done in FromThePage, the user of Transc&Anno performs annotation using hot keys or buttons triggering pop-up menus.

One advantage of the interface being presented is the opacity of the possibly sophisticated annotation scheme for the transcriber. Once he chooses the tag to apply to a certain portion of text, he is automatically prompted to fill in the corresponding category's attributes' values (Figure 1).

The annotation can be performed by pushing buttons corresponding to categories defined for the current collection or by pressing hot keys. Using exclusively hot keys can considerably increase the annotation speed.
The annotation categories' menu is located on the right side of the screen: each category's name is shown using the style that has been assigned to it (Figure 1). Other buttons allow last operation cancellation, tag deletion, tag modification, hot keys modification. The user can also choose the frequency with which the system automatically saves the transcription.
If the user is unsatisfied with the work he has recently done, he has the possibility to revert to previous versions of the transcription.

4.7 Transcription and Annotation Result

The created resource format is inline TEI compatible XML, but it is shown as highlighted text, no XML markup is visible to the annotator (Figure 1). Despite of being implemented as inline XML, annotation tags can overlap. Overlapping is achieved by closing a tag and reopening it a second time preserving its id: the tagcode.
Figure 2 shows an example of a text tagged with Transc&Anno that corresponds to the text shown in Figure 1.

```
1    <greeting_id15 class="medium-greeting_id15" mode="1"
tagcode="1504275956554">
2      <adjective_id31 class="medium-adjective_id31" gender="m"
mode="1" number="sg"
tagcode="1504276017378">Caro</adjective_id31>
3      <noun_id30 class="medium-noun_id30" gender="m" mode="1"
number="sg" tagcode="1504276025907">Marco</noun_id30>
4    </greeting_id15>
5    ,
6    <div>
7      <main_clause_id19 class="medium-main_clause_id19"
mode="1" tagcode="1504275989612">Non crederai
mai</main_clause_id19>
8      <subordinate_clause_id20 class="medium-
subordinate_clause_id20" mode="1" tagcode="1504275994412">
       cosa e successo giorni fa a
9        <noun_id30 class="medium-noun_id30" gender="f"
mode="1" number="sg"
tagcode="1504276040872">Maria</noun_id30>
10     </subordinate_clause_id20>
       .
11   </div>
```

Figure 2: Transcription result.

This text has been tagged with some pragmatical (the *greeting* tag on lines 1-4), grammatical (the *adjective* and *noun* tags on lines 2, 3, and 9) and syntactical (the *main_clause* and *subordinate_clause* tags on lines 7-10) information. As multiple tags can be assigned to the same string sequence, all these different kinds of information could be added to the text. Most categories used in this annotation do not have attributes, but the noun and adjective categories do: the noun (lines 3 and 9) and adjective (line 2) tags contain the number and gender attributes. Their values have been chosen from a predefined set and therefore have the same form. The class attribute allows text styling via CSS stylesheets and the tagcode attribute uniquely identifies every single annotation. Each annotation category has its own id that distinguishes it from similar categories of other collections.

5. Technical Aspects

Transc&Anno was created on top of the FromThePage transcription tool developed entirely with standard web technologies – Ruby on Rails, Javascript, HTML, and CSS. In order to add the text tagging functionality to FromThePage, we used an open-source Javascript library medium.js[11]. Ruby on Rails interacts with a MySQL database. An automated transcription saving procedure has been set in order to avoid the loss of manual work. Transcriptions are registered in XML format with inline annotations. The validity of the XML is automatically checked before saving.
Transc&Anno is open source, easily customizable and available on Github[12].

6. Conclusion and Future Work

In this paper, we presented Transc&Anno, a web-based collaboration tool allowing the transcription of text images and their shallow on-the-fly annotation. The annotation results in inline TEI compatible XML and can possibly include overlapping annotation tags. Transc&Anno ensures a high transcription output quality by validating the XML and only allowing predefined tags. We intend to use Transc&Anno for transcribing learner corpora of German and Italian.
In creating Transc&Anno, our purpose is not to develop a fully fledged annotation tool, but to provide a user-friendly transcription tool allowing shallow on-the-fly annotation. Nonetheless, we are considering the possibility to extend annotation capacities of the tool by allowing discontinued constituents' tagging as well as pointing from a tag to another portion of text.
Since the visual representation of a transcription is crucial for users' comfort, our next step is to further improve it. We intend to allow an unlimited number of tags be visible in the same text fragment.

Bibliographical References

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Štindlová, B., Vettori, C.: The MERLIN corpus: Learner language and the CEFR, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, 2014, pp. 1281-1288.

Gamon, M., Chodorow, M., Leacock, C., Tetreault, J.: Using learner corpora for automatic error detection and correction. In Bariller, N., Thompson, P., Automatic Treatment and Analysis of Learner Corpus Data, Amsterdam & Philadelphia: Benjamins, 2013, pp. 127-149.

Glaznieks, A., Nicolas, L., Stemle, E., Abel, A., Lyding, V.: Establishing a Standardised Procedure for Building

11 http://jakiestfu.github.io/Medium.js/docs/

12 https://github.com/commul/fromthepage

Learner Corpora, *Journal of Applied Language Studies*, Vol. 8, 3, 2014, pp. 5-20.

Granger, S.: A Bird's Eye View of Learner Corpus Research. In Granger, S., Hung, J. & Petch-Tyson, S. (eds.), Computer Learner Corpora, Second Language Acquisition and Foreign Language Teaching, Amsterdam & Philadelphia: Benjamins, 2002, pp. 3-33.

Hana, J., Rosen, A., Škodová, S., Štindlová, B.: Error-tagged Learner Corpus of Czech, *Proceedings of the Fourth Linguistic Annotation Workshop*, Uppsala, Sweden, 15-16 July, 2010, pp. 11-19.

Hasan, M. M., Khaing, H. O.: Learner Corpus and its Application to Automatic Level Checking using Machine Learning Algorithms, *Proceedings of ECTI-CON*, 2008, pp. 25-28.

Janssen, M.: TEITOK: Text-Faithful Annotated Corpora, *Proceedings of LREC 2016*, Portorož, Slovenia, pp. 4037-4043.

Nicolas, L., Stemle, E., Glaznieks, A., Abel, A.: A Generic Data Workflow for Building Annotated Text Corpora, In Castello, E., Ackerley, K., Coccetta, F., Studies in Learner Corpus Linguistics, 2015, pp. 337-351.

Tenford, K., Meurer, P., Hofland, K.: The ASK Corpus – a Language Learner Corpus of Norwegian as a Second Language, *Proceedings of LREC 2006*, Genoa, Italy, pp. 1821-1824.

Yimam, S.M., Gurevych, I., Eckart de Castilho, R., Biemann C.: WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations, *Proceedings of ACL-2013*, Sofia, Bulgaria, pp. 1-6.

Müller, C., Strube, M.: MMAX: A tool for the annotation of multi-modal corpora, *2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, 2001, pp. 45-50.

Correction of OCR Word Segmentation Errors in Articles from the ACL Collection through Neural Machine Translation Methods

Vivi Nastase and Julian Hitschler
University of Heidelberg
Heidelberg, Germany
{nastase, hitschler}@cl.uni-heidelberg.de

Abstract

Depending on the quality of the original document, Optical Character Recognition (OCR) can produce a range of errors – from erroneous letters to additional and spurious blank spaces. We applied a sequence-to-sequence machine translation system to correct word-segmentation OCR errors in scientific texts from the ACL collection with an estimated precision and recall above 0.95 on test data. We present the correction process and results.

Keywords: character-level sequence-to-sequence model, word segmentation, ACL collection

1. Introduction

The ACL anthology provides a valuable collection of scientific articles, and organizing it into a structured format could provide us with additional insight into research in this domain, help with finding related work and help with keeping up with new developments and ideas. The analysis of the ACL collection was stimulated by the shared task at ACL 2012 (Schäfer et al., 2012), the workshop on *Rediscovering 50 years of Discoveries*, and the series of SemEval tasks on *ScienceIE – Extracting Keyphrases and Relations from Scientific Publications*. Analysis has tackled various aspects of this collection: the citation network (Sim et al., 2012), citation references (Radev and Abu-Jbara, 2012; Gordon et al., 2016), keywords and relation extraction (Gábor et al., 2016), topics and community studies (Bordea et al., 2014), among others.

Considering the status of our NLP toolbox, the success on processing such a corpus increases when the texts are clean. The ACL collection contains numerous articles published before electronic submission became standard for our conferences. These older papers have been scanned and processed through OCR, resulting in texts that contain errors. A brief inspection of this portion of the collection has shown that a very common error introduced by the OCR process consists of spurious blank spaces, which split words randomly in a varying number of smaller fragments (Figure 1). This problem is so pervasive, that it influences subsequent processing, for example keyword extraction – in the list of keywords produced with the SAFFRON system, we found the following keywords in papers from Coling 1996: *Non-l inear*, *Context fi*, *Segmentat ion*, *Ion theory*.

Inspired by previous work on error correction using machine translation models (e.g. (Yannakoudakis et al., 2017)), we apply a character-level sequence-to-sequence model to learn how to segment English words in the ACL collection. Written modern languages of European origin usually segment words explicitly, so English texts generally do not require word segmentation, but as we have seen, this problem has popped up in documents processed with OCR. This means that we can very easily obtain large volumes of data for training. We use a portion of the ACL collection (the articles published after electronic submission had become the norm in our community, a date which we conservatively set at 2005) to generate training, development and test data. The high results – above 96% precision – obtained on the test data indicate that processing the part of the collection published before 2005 would solve the vast majority of the word fragmentation issues, and would provide the community with a corpus of increased quality. In this paper, we present the processing tool and experiments done on the post-2005 portion of the collection, and the corrected ACL collection will be offered to the ACL anthology editor to be made available to the community.[1]

2. Related Work

Depending on their source, errors in unedited texts can fall into various categories: typos, deliberate misspellings including shortened/phonetically written words (particuarly on social media), word segmentation errors, erroneous characters, non-canonical spellings (historical texts), grammatical errors, and probably more.

For many of the above mentioned problems, neural-based approaches originally developed for machine translation have proved to be very successful. Yannakoudakis et al. (2017) use a machine translation inpired approach – N-best list ranking using neural sequence labelling models – for grammatical error correction. Word and character-based sequence-to-sequence models (Yuan and Briscoe, 2016; Xie et al., 2016; Yang et al., 2017) have achieved good performance on the CoNLL-2014 shared task (Ng et al., 2014) on text correction (which covers a variety of errors made in written essays by second language learners). Schmaltz et al. (2017) show that character-level sequence-to-sequence models perform better than word-level models even with less training data than previous sequence-to-sequence ap-

[1] Information about how the word re-segmented version of the ACL corpus is available will be posted at `http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/ACL_corrected/ACL_corrected.shtml`.

```
INTRODUCTION
Although the l i terature dealing with formal and natural
languages abounds with theoretical arguments of worst-
case performance by various parsing strategies \[e.g.,
Griffiths & Petrick, 1965; Aho & Ullman, 1972; Graham,
Harrison & Ruzzo, Ig80\], there is l i t t le  discussion of
comparative performance based on actual practice in
understanding natural language. Yet important practical
considerations do arise when writing programs to under-
stand one aspect or another of natural language utteran-
ces. Where, for example, a theorist wi l l  characterize a
parsing strategy according to its space and/or time
requirements in attempting to analyze the worst possible
input acc3rding to ˜n arbitrary grammar st r ic t ly  limited
in expressive power, the researcher studying Natural
Language Processing can be just i f ied in concerning
himself more with issues of practical performance in
parsing sentences encountered in language as humans
Actually use i t  using a grammar expressed in a form
corve˜ie: to the human linguist who is writing i t .
```

Figure 1: Excerpt from paper P81-1001, *A Practical Comparison of Parsing Strategies* by Jonathan Slocum

proaches, and outperform statistical phrase-based machine translation models on the CoNLL data.

Chen et al. (2015a; Chen et al. (2015b) explore the use of GRUs and LSTMs for Chinese word segmentation, and Zhang et al. (2016) approach the task using word and character context in a globally optimized beam-search framework for neural structured prediction. Yang et al. (2017) build on this previous work to produce a modular neural-based segmentation model for Chinese, using five-character window, pre-trained based on a variety of external resources.

Based on these previous analyses into the kind of architectures that perform well for different types of error correction, we adopt a character-level sequence-to-sequence model for the word segmentation of English texts.

3. The ACL collection

The ACL collection we work with consists of 18,849 scientific articles published between 1965 and 2012. Papers published before electronic submission became the norm in the 2000s have been scanned and processed with OCR, and as such suffer from the type of errors common in such material – most notably spurious spaces and erroneous characters.

The most common OCR error we noticed was incorrect word segmentation – there are numerous spurious blank characters at random locations in the texts, as can be seen in the text fragment from paper P81-1001 displayed in Figure 1 (which we reproduce as is in the file, including new lines). The problem is so pervasive that it influences tasks such as keyword extraction, the basis for further processing of the collection. This is evidenced by an inspection of the keywords produced with the SAFFRON system (Bordea et al., 2014), wherein we find keywords among the top 15 ranked for each articles that were affected by this OCR error. While wrongly split keywords may not seem so problematic as long as all parts are present (e.g. *Segmentat ion*), incomplete words may lead to erroneous or misleading keyphrases (e.g. *Context fi* and *Ion theory*). We set to address this problem, considering that training and test data can be automatically obtained from the portion of the ACL collection that consist of electronic submissions (conservatively, we choose 2005 as our lower time limit).

4. Machine translation-based correction model

We use the *nematus* system[2] (Sennrich et al., 2017), a state-of-the-art sequence-to-sequence machine translation model. It is a highly configurable system that implements an attentional encoder-decoder architecture. For the experiments presented here we use the default cross-entropy minimization as the training objective, via (accelerated) stochastic gradient descent. We use this system to process sequences at the character level. The training data consists of parallel input-output sequences, with a default limit of sequence length 100. Below we describe what kind of training data was provided to the system.

As noted from the fragment in Figure 1, the scanned text preserves the line breaks from the original paper, which include hyphenated words. The first processing step we apply to the entire collection is to remove the new lines if the line does not finish with a dot, question mark or colon. Hyphenated words are replaced with their non-hyphenated version if such a variant was encountered anywhere in the texts. This processing step produces texts with one paragraph per line. After this step, we separated the collection – pre-2005 (B2005) (to be conservative about the beginning of widespread use of OCR) and post-/ including 2005 (A2005). The texts from A2005 were considered correct from the point of view of word segmentation, and the training data was produced from these texts, taking into account

[2] https://github.com/EdinburghNLP/nematus

input	output
t o - i n f i n i t i v e s	t o - i n f i n i t i v e s
a n d g e r u n d s	a n d ## g e r u n d s
B o t h t h e b a s e l i n e a n d S p a d e o p e r a t e o n p a r s e	B o t h ## t h e ## b a s e l i n e ## a n d ## S p a d e ## o p e r a t e ## o n ## p a r s e
t r e e s w h i c h w e r e o b t a i n e d f r o m C h a r n i a k ? s	t r e e s ## w h i c h ## w e r e ## o b t a i n e d ## f r o m ## C h a r n i a k ? s
p a r s e r	p a r s e r
O u r s e t o f e x p e r i m e n t a l m a t e r i a l s c o n t a i n e d	O u r ## s e t ## o f ## e x p e r i m e n t a l ## m a t e r i a l s ## c o n t a i n e d
c o m p r e s s i o n s	c o m p r e s s i o n s
P r o c e d u r e a n d S u b j e c t s W e o b t a i n e d c o m p r e s s i o n	P r o c e d u r e ## a n d ## S u b j e c t s ## W e ## o b t a i n e d ## c o m p r e s s i o n
r a t i n g s d u r i n g a n e l i c i t a t i o n s t u d y c o m p l e t e d	r a t i n g s ## d u r i n g ## a n ## e l i c i t a t i o n ## s t u d y ## c o m p l e t e d

Table 1: Example of input-output parallel training data for correcting word segmentation problems in the ACL collection. The ## sequence indicates a blank space.

predicted `001) and by items ( F2(3; 117) = 40 \/\/ 0 0 1 )`
gold standard `001) and by items ( F2(3;117) = 40 \/\/ 0 0 1 )`

where ”F2(3;117)” is considered one word in the gold standard, and is split in two in the predicted version.

Table 2: Errors caused by erroneous spacing in formulas.

the chosen sequence limit, as follows:

1. the texts with one paragraph per line are split into smaller fragments, avoiding as much as possible splitting on ”ambiguous” breaking points (i.e. spaces between text fragments which may actually be erroneous):
   (a) split on end of sentence characters or phrase delimiting characters (.?!;: - parentheses)
   (b) if the fragment is longer than 50 characters, split at numbers
   (c) if the fragment is still longer than 50 characters, split into 50 character long sequences
2. produce nematus input training data by removing blank spaces from the string, and (conform nematus' input formatting) inserting a blank space after each character
3. produce nematus output training data by replacing blank spaces with a special sequence (##) and then inserting a blank space after each character (except the ones in the special sequence).

The parallel input-output training data is exemplified in Table 1.

The data prepared in this manner consists of 9,310,664 input-output parallel sequences. We selected 2,000,000 sequences for training, and 500,000 sequences for testing. Training was done using the system's default settings – training is done with cross-entropy minimization with adam optimizer, encoder and decoder implement GRUs, learning rate 0.0001, embedding layer size 512, hidden layer size 1000, dropout for input embeddings and hidden layers 0.2. The model built during training is used to ”translate” the input test data, which are then compared token-by-token to the expected output test sequences.

5. Results and discussion

We have performed two evaluations: one with respect to the test data described in Section 3., and one on the actually corrected data, the B2005 portion. The results of these evaluations are described in Sections 5.1. and 5.2. respectively.

5.1. Evaluation on test data

For the ACL correction, evaluation was performed on 500,000 fragments obtained as explained in Section 3.. We evaluate in terms of word-level precision and recall, computing the number of correctly predicted words. Formally, for an automatically produced sequence $\mathbf{w}_a$, we compute precision and recall by comparison with a gold standard sequence $\mathbf{w}_{gs}$[3]:

[3]We consider these sequences as ordered lists of words, and in evaluation we gradually shorten the list such that two occurrences of the same word in the automatically produced output are compared to different tokens in the gold standard.

```
Although the literature dealing with formal and natural languages abounds
with theoretical arguments of worst-case performance by various parsing
strategies \[ e  . g . , Griffiths & Petrick , 1965; Aho & Ullman , 1972;
Graham , Harrison & Ruzzo , Ig 80\] , there is little discussion of comparative
performance based on actual practice in understanding natural language  . Yet
important practical considerations do arise when writing programs to understand
one aspect or another of natural language utteran-ces  . Where , for example ,
a theorist will characterize a parsing strategy according to its space and/or
time requirements in attempting to analyze the worst possible input  acc3rding
to ˜ n arbitrary grammar strictly limited in expressive power  , the researcher
studying Natural Language Processing can be justified in concerning himself
more with issues of practical performance in parsing sentences encountered in
language as humans Actually use it using a grammar expressed in a form corve ˜
ie : to the human linguist who is writing it  .
```

Figure 2: Corrected version of the excerpt from paper P81-1001, *A Practical Comparison of Parsing Strategies* by Jonathan Slocum from Figure 1

$$Prec = \frac{|\{w|w \in \mathbf{w}_{gs}, w \in \mathbf{w}_a\}|}{|\mathbf{w}_a|}$$

$$Rec = \frac{|\{w|w \in \mathbf{w}_{gs}, w \in \mathbf{w}_a\}|}{|\mathbf{w}_{gs}|}$$

Precision and recall on the entire test data is a micro-average of the scores for the 500,000 sequences in the test data. We obtained a precision of 0.955 and recall of 0.950 on re-segmenting into words. Most of the errors we observed are caused by spaces in formulas, as in the examples in Table 2. Partially discounting this type of error (which impacts very little, if at all the text processing of the ACL collection), the precision and recall become 0.979 and 0.974 respectively.

Because we want to apply the trained model to the B2005 portion of the collection for which we have no test data, we would like to have a better idea of what the results are likely to be. The fact that the papers in the B2005 portion of the collection come from the same domain as the A2005 portion which was used for training makes it highly likely that most of the vocabulary in B2005 is shared with A2005, but there will also be tokens unknown to the model. For this reason we performed an additional evaluation on the A2005 test data, for tokens that do not appear in the training or development data. The recall on these unknown tokens is 0.821, and the precision 0.876. Many of these unknown tokens are part of formulas, which, as mentioned before, we think is highly unlikely to impact information/keyword extraction, and other such tasks that focus on "proper" text.

5.2. Evaluation on corrected data (B2005)

The high performance on the test data and on unknown tokens indicates that applying the model to the B2005 collection will likely solve many of the existing segmentation problems. In Figure 2 we include the version obtained after the word re-segmentation process of the fragment in Figure 1.

The motivation that lead us to attempt correcting word segmentation errors was that they were so prevalent that even keywords presented such problems. Within the used ACL anthology was a file of keywords obtained with the SAFFRON system (Bordea et al., 2014). This version of the system is no longer available to obtain keywords on the corrected version of the files. Because of this we decided to perform an evaluation with respect to the keywords obtained on the raw files: we selected randomly 40 papers from the collection published before 2005, and annotated each of their keywords as *correct* or *incorrect* with respect to word segmentation[4]. This provided a total of 573 keywords, 38 of which were incorrect, and 535 correct. We tested each of these keywords against the raw and corrected versions of the corresponding files, and present the summary of results in Table 3.

tag	N	in the raw text	in the corrected text
incorrect	38	35 (92.1%)	3 (7.9%)
correct	535	455 (85.05%)	453 (84.67%)

Table 3: Results on the keywords from a sample of 40 documents

Of the 38 incorrect keywords, 35 actually appear in the raw files. Those that do not appear in the raw files seem to be caused by some preprocessing done by the keyword extractor – e.g. for the paper A94-1014, the keyword that does not appear in the raw (or corrected) file is "Computer Science Univ", which seems to have been caused by collapsing the lines that contain the authors' affiliations ("Dept . of Computer Science [new line] Univ . of Central Florida"). Only 3 of the 38 badly segmented keywords appear in the corrected files.

For the correct keywords, the reason why some do not appear in the raw or the corrected files is mainly the preprocessing done by the keyword extractor (e.g. lemmatization). There are two correct keywords that appear in the raw files but not in the corrected files. One of these is *language process* (from paper P84-1101), which seems correct

[4]The annotation was performed by one of the authors of the paper. The problem is quite obvious, and there didn't seem to be a need for an additional annotator and agreement computations.

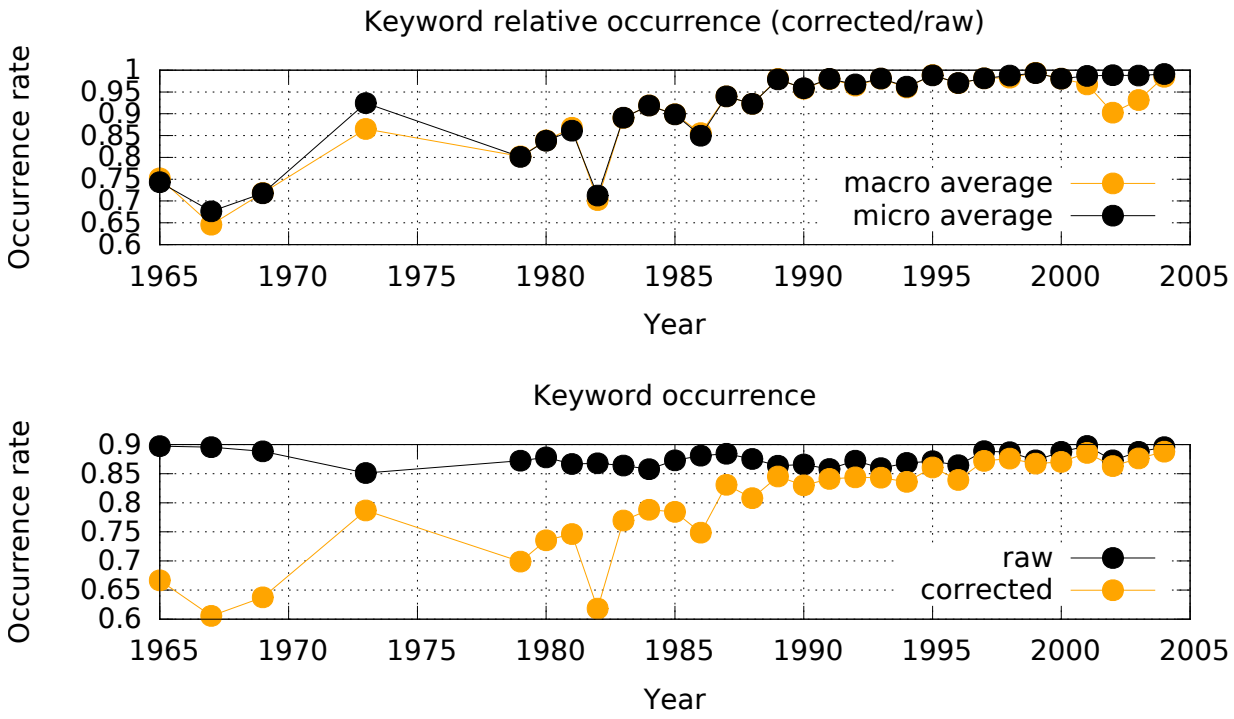


Figure 3: Proportion of keywords from the original (raw) files in the raw files and the corrected versions.

but in fact was extracted because of the erroneous segmentation *language process ing*. The second error is a proper processing error: the keyphrase *rigid system-directed dialogue structure* (from paper E93-1061) appears in the corrected version as *arigid system-directed dialogue structure*, "rigid" having been merged with the preceding indefinite article.

The results of the manual analysis show that the sequence to sequence model does indeed perform well on correcting the B2005. We performed an additional analysis to obtain another estimation of the impact of the word re-segmentation. A visual inspection of the ACL data shows that older papers appear more difficult for OCR processing as they seem to have been first scanned from prints of various quality. More recent papers were more often processed from directly produced pdf files. This would indicate that older papers have more OCR (and thus, word segmentation) errors compared to more recent ones. This would mean that more recent papers have more of the extracted keywords (because they would be more likely to be correct) than older papers. We plot the rates of occurrence of the keywords in the raw and corrected versions on the files, by year, for the B2005 articles that have keywords. The top of Figure 3 shows the macro and micro average of the number of keywords that appear in the corrected vs. the raw articles (number of keywords that appear in the corrected files divided by the number of keywords in the raw files). The bottom figure plots separately the rate of occurrence of the keywords in the raw and corrected files respectively, relative to the total number of keywords for each paper. The upward trend of the ratio of old keywords in the corrected files confirms the above observation, as in older papers fewer keywords appear in the corrected versions of the papers than in more recent ones, correlating with the increase in OCR quality.

The B2005 portion of the ACL collection was processed, and the re-segmented texts will be offered to the ACL anthology editors. Information about availability will be posted on the website of the University of Heidelberg's Computational Linguistics Institute[5].

6. Conclusion

We have presented the character-level sequence-to-sequence model used to correct one of the very pervasive errors in the part of the ACL collection processed through OCR – spurious blank spaces that fragment words. The high results on the test portion of the data indicate that a large part of this type of errors could be corrected in the ACL collection. We have applied this process and produced a cleaner version of the ACL collection, which we will offer to the ACL anthology editors to make it available with the raw collection to the community.

7. Acknowledgements

We thank our reviewers for their comments. This research was funded by the Leibniz Science Campus Empirical Linguistics & Computational Language Modeling, supported by Leibniz Association grant no. SAS-2015-IDS-LWC and by the Ministry of Science, Research, and Art of Baden-Württemberg.

8. Bibliographical References

Bordea, G., Buitelaar, P., and Coughlan, B. (2014). Hot topics and schisms in NLP: Community and trend anal-

[5] http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/ACL_corrected/ACL_corrected.shtml

ysis with Saffron on ACL and LREC proceedings. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Chen, X., Qiu, X., Zhu, C., and Huang, X. (2015a). Gated recursive neural network for chinese word segmentation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1744–1753, Beijing, China, July. Association for Computational Linguistics.

Chen, X., Qiu, X., Zhu, C., Liu, P., and Huang, X. (2015b). Long short-term memory neural networks for chinese word segmentation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1197–1206, Lisbon, Portugal, September. Association for Computational Linguistics.

Gábor, K., Zargayouna, H., Buscaldi, D., Tellier, I., and Charnois, T. (2016). Semantic annotation of the acl anthology corpus for the automatic analysis of scientific literature. In *LREC 2016*.

Gordon, J., Zhu, L., Galstyan, A., Natarajan, P., and Burns, G. (2016). Modeling concept dependencies in a scientific corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–875. Association for Computational Linguistics.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H., and Bryant, C. (2014). The conll-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland, June. Association for Computational Linguistics.

Radev, D. and Abu-Jbara, A. (2012). Rediscovering acl discoveries through the lens of acl anthology network citing sentences. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 1–12. Association for Computational Linguistics.

Schäfer, U., Read, J., and Oepen, S. (2012). Towards an acl anthology corpus with logical document structure: an overview of the acl 2012 contributed task. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 88–97. Association for Computational Linguistics.

Schmaltz, A., Kim, Y., Rush, A., and Shieber, S. (2017). Adapting sequence models for sentence correction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2797–2803, Copenhagen, Denmark, September. Association for Computational Linguistics.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017). Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain, April. Association for Computational Linguistics.

Sim, Y., Smith, N. A., and Smith, D. A. (2012). Discovering factions in the computational linguistics community. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 22–32. Association for Computational Linguistics.

Xie, Z., Avati, A., Arivazhagan, N., Jurafsky, D., and Ng, A. Y. (2016). Neural language correction with character-based attention. *CoRR*, abs/1603.09727.

Yang, J., Zhang, Y., and Dong, F. (2017). Neural word segmentation with rich pretraining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 839–849, Vancouver, Canada, July. Association for Computational Linguistics.

Yannakoudakis, H., Rei, M., Andersen, Ø. E., and Yuan, Z. (2017). Neural sequence-labelling models for grammatical error correction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2785–2796, Copenhagen, Denmark, September. Association for Computational Linguistics.

Yuan, Z. and Briscoe, T. (2016). Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California, June. Association for Computational Linguistics.

Zhang, M., Zhang, Y., and Fu, G. (2016). Transition-based neural word segmentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 421–431, Berlin, Germany, August. Association for Computational Linguistics.

From Manuscripts to Archetypes through Iterative Clustering

Armin Hoenen
CEDIFOR
Goethe university Frankfurt
hoenen@em.uni-frankfurt.de

Abstract

Corpora of manuscripts of the same ancient text often preserve many variants. This is so because upon copying over long copy chains errors and editorial changes have been repeatedly made and reverted to the effect that most often no 2 variant texts of the same so-called textual tradition have exactly the same text. Obviously in order to save the time to read all of the versions and in order to enable discourse and unambiguous referencing, philologists have since the beginnings of the age of print embarked on providing one single textual representation of this variety. In computational terms one tries to retrieve/compose the base text which is most likely the latest common ancestor (archetype) of all observed variants using stemmata – that is trees depicting the copy history (manuscripts = nodes, Copy processes = edges). Recently, they have been computed and evaluated automatically (Roos and Heikkilä, 2009). Likewise, automatic archetype reconstruction has been introduced lately (Hoenen, 2015b; Koppel et al., 2016). A synthesis of both stemma generation and archetype reconstruction has not yet been achieved. This paper therefore presents an approach where through iterative clustering a stemma and an archetype text are being reconstructed bottom-up.

Keywords: stemmatology, clustering, archetype text reconstruction, digital textual criticism

1. Introduction

For corpora of ancient texts transmitted (and born) in the age of handwriting, collections of those documents tend to have variant texts which differ slightly due to a variety of reasons among which miscopying, editorial changes such as modernization and stylistic changes may be the most well known due to their illustrativity. In order to provide one text version to the interested scholar, generations of philologists have tried to offer single texts (even though some of those are embedded as base texts in edition types such as critical editions) which they argue best represent the most probably lost authorial original and which somehow mimick an *edicio princeps* initial print exemplar of an autograph.[1] Computationally, the task involves various subtasks. First, manuscript texts need to be digitized, for instance through manual retyping and possibly encoding. To this end, the Text Encoding Initiative[2] provides the elements and rules for an appropriate schema. Texts can be manually or computationally aligned and from this various algorithms can produce stemmata, see Figure 1 that is trees with nodes representing manuscripts and edges representing copy processes or chains thereof, which graphically display (a hypothesis about) the copy history of the variants. Although Cameron (1987) observes that the goal of a textual critic[3] or philologist is rather the reconstruction of an urtext[4] than of the copy history and that the copy history

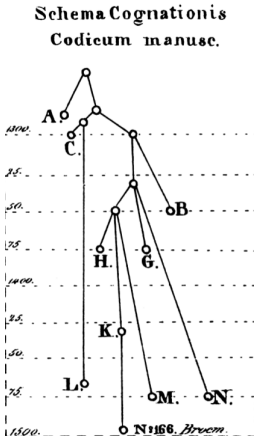


Figure 1: First modern stemma by Schlyter, 1827, from O'Hara (1996) with texts= nodes, copy processes= edges.

is only an intermediate goal on the way to such an urtext, computational attempts at reconstructing urtexts are only very recent and few (Nassourou, 2013; Hoenen, 2015b; Koppel et al., 2016). In this paper, we present an approach which combines the stemma building and archetype reproduction processes in the following way.

Starting from a number of prealigned observed variant texts, those are transformed, a) into pseudo-DNA and b) into bitvectors of so-called *leitfehler*. We derive the technical meaning of leitfehler in this context from the implemen-

[1] Consider Shillingsburg (2017) for discussions on (other) purposes of editions.

[2] `http://www.tei-c.org/`

[3] Textual criticism is the philological discipline of critically reflecting the given variation in a tradition.

[4] The term *urtext* is a loan from German which may refer to the very first (complete) version of a text. The term *autograph* refers to (any) manuscripts written by an author him or herself. This can also be a copy further down the tree and the original mustn't be an autograph in case of dictation for instance. The root of a stemmatic tree can then be an *urtext* or *autograph*, but it can and most often is nothing more or less than the latest common ancestor of all surviving manuscripts and as such often a 2nd or 3rd generation copy of the original (urtext, autograph). Such a latest common ancestor, the best-we-can-do reconstruction based on evidence (further reconstruction if being done, more hypothetical) can be called *archetype* in philological jargon, compare Trovato (2014). In summary, the terms, {original, urtext, autograph, latest common ancestor, archetype} may all refer to the root of some stemmatic tree but all convey different notions about that tree.

Position	m_1	m_2	m_3	m_4	m_5	m_6	X	ζ	π	**Variants**
1	that	this	ðis	ðis	t'	this	ðis	this	this	A-D
2	iz	iz	is	is	is	'	is	is	is	A-C
3	one		a	a	an	an	an	one	an	A-D
4	text	text	tekst	text	text	tekst	text	text	text	A,B
pseudo-DNA	DCBA	ACDA	BACB	BACA	CAAA	ABAB	BAAA	AABA	AAAA	**pseudo-DNA**
Leitfehler bitvector	01010	01000	10001	10000	00100	00101	10100	10010	00100	Leitfehler bitvector

Table 1: An example of how pseudo-DNA can be generated which reflects a binary distance between words (same/different). Bitvectors of leitfehler, if the list of good Leitfehler were (ðis, iz, an, one, tekst).

tation by Roelli and Bachmann (2010) which is explained in more detail in the next section. For now it may suffice to understand that this refers to a pruned list (Roelli & Bachmann do not necessitate pruning, but for the application intended here it is necessarily a pruned list) of genealogically indicative variants. That is, by this method, for each variant it is determined if it can serve as a leitfehler (1) or not (0). For an example of the transformations see Table 1.[5] Then several steps are taken to produce both a stemma and an archetype text bottom-up.

1. Vectors or texts are non-hierarchically clustered by some algorithm (Farthest First, Simple K-means, Threshold dependent similarity based clustering) into groups.
2. Incompatible groups (in case one manuscript is detected as belonging to more than one cluster) are resolved into one single cluster.
3. For each cluster, an ancestral text is constructed which then replaces all its descendents in the corpus.[6] New reconstructed nodes are saved as nodes and the ancestral relations as edges of a stemmatic tree which "grows" bottom-up.
4. Iteration of 1.-3. until only one cluster or only clusters with 1 member each exist. The last text reconstruction step assembles the assumed archetypical text (root of the stemma, latest common ancestor to all extant texts).

After obtaining both stemma and archetype this way, we evaluate both of them separately.

2. Artificial Data Sets

For experimentation and evaluation, we use so called artificial datasets or benchmark datasets. Such a dataset is one for which the true copy relationships and thus the true stemmata are known. This is so, because the texts had been given to volunteers (students, friends, etc.) who had copied them by hand recording who had copied which version from whom. The artificial datasets have thus been produced by recent scribes, not medieval ones. Furthermore, their time and generation depth are not representative of historical cases. Thus, those datasets are but an extremely limited testbed for automatic stemma generation which is by no means representative of the variety of processes and topologies historical stemmata can display (especially cross fertilization or contamination). Furthermore, their small size makes them challenging in terms of the application of data hungry statistical approaches (many machine learning approaches). However, they are the only datasets for which the *copy truth* is known and not a matter of scholarly debate. An alternative is to use stemmata created by the philological community on certain historical works, but this way of conduct has, in the authors view some crucial disadvantages. First, if a stemma, which the community agrees upon already exists, an automatic stemma generation can tendentially at maximum serve to rearrange or corroborate details. Secondly, the evaluation of two different algorithmic methods with different outcomes may be obscured/skewed in favour of the less accurate algorithm by a wrong assumption in the data with only massive amounts of data and massive interpretational intervention providing some remedy, a point which is currently not yet reached in computational stemmatology in the authors view. Thirdly, if the philological community heavily debates the question when (for instance for which tradition) to use stemmatology or even if to deem it useful at all (compare the best-text-editing debate (Bédier, 1928; Maas, 1937), consider also (Haugen, 2015; Hoenen et al., 2017)), demonstrable objectivity in stemma evaluation may be one of the best arguments in trying to win philological users. Fourthly, the method can also be used in non-philological contexts, for instance reconstructing Wikipedia versioning histories (Marmerola et al., 2016) for which no historical uncertainty must be coped with. In sum, the situation is very challenging and some sort of dilemma, on the one hand the benchmark datasets do not represent even a tiny fraction of historical variation. On the other hand, they are the only datasets where we do know the ground truth for sure. The author decided to pursue the path of working primarily on them based under more on the aforementioned reasons, but does not assume that this is the only reasonable or the only objective procedure.

In the context of this study, we use the three previously most used artificial datasets, called Parzival PRZ (English), Notre Besoin NB (French) and Heinrichi HR (Finnish), (Baret et al., 2004; Spencer et al., 2004; Roos and Heikkilä, 2009).[7] PRZ has 21 manuscripts and the alignment has 855 lines, NB features 13 manuscripts of 1035 lines and HR 64

[5] The unit of all subsequent comparisons are tokens mostly coinciding with words or punctuation, more precisely alignment positions. Some phenomena, which may be common in historical data such as transpositions are not captured directly but produce several positions aligning with gaps.

[6] This step is similar to other matrix updating procedures for instance the one in the Neighbor Joining algorithm (Saitou and Nei, 1987).

[7] Reduced sets (as a simulation of historical loss) available online. We obtained the complete sets from the authors.

manuscripts of 1208 words.[8] We use the complete datasets and various reduced sets representing loss (among which the reduced sets from the above mentioned authors on PRZ and HR are not present/not analysed separately since they represent only one of many possible loss scenarios).

3. Method

Philologists often operate with tables of numbers of shared variants (agreements and disagreements) and postulate a group of manuscript texts which go back to a common ancestor where the amount of shared features makes this plausible, compare (West, 1973; Timpanaro, 2005). Numerical considerations do play a role in this type of human guided manual clustering, but in cases of doubt, hermeneutical argument can always over-rule numerical considerations. Since we aim at automatical methods, since the computer is not capable of complex hermeneutical reasoning, at least not without explicit knowledge bases and sophisticated algorithmic architectures which to date are not available for this specfic task, we cluster computationally in two flavours and 4 variants.

First, we mimick simplistically philological clustering and detect groups which share a certain percentage of variants (threshold set to 0.95). That is we simply define a threshold of for instance 95% and then for each manuscripts detect all other manuscripts which share at least that amount of variants (proportion of positions in the alignment, where both have exactly the same text). Then we resolve this information into groups (clusters) for the whole set. In case of ambiguity, ambiguous groups are merged.

Secondly, we compile a bitvector of so-called *leitfehler*. Leitfehler are variants, which witness certain distributional patterns on the (true) stemma. Philologically, a leitfehler according to Roelli and Macé (2015) is understood as a genealogically significant error. The authors of the only algorithm implementing a method based on the leitfehler (Roelli and Bachmann, 2010; Roelli, 2014) which according to Roelli and Macé (2015), p.129, still needs development, try to measure the capacity of any variant to serve as such a 'guiding error' for the build-up of a stemma. Roelli and Macé (2015, p.129) state that their method is:

> [...] a subcategory of distance-based methods. Thus the traditional scholarly concept of Leitfehler is taken to be a quantitative one: a variant's usefulness as Leitfehler may be assigned a number or weight. In classical stemmatology the Leitfehler is the most important tool to arrive at a filiation of witnesses that is believed to be most correct representation. [..., evaluating how good a leitfehler a variant is] for every pair of them [...] If one of the four combinations of absence / presence of any of these two candidates [intending a truth table for the variants A and B: AB ,$A\neg B$,$\neg AB$,$\neg A\neg B$] is not represented in any witness, this is taken to be a hint that both variants suffered their change from absence to presence (or vice versa) exactly once in the tradition, which is characteristic for good traditional Leitfehler (Maas 1937). Such a comparison can be made for all combinations of potential Leitfehler while both Leitfehler in pairs with only three combinations get their score increased.

Thus, the algorithm produces a list, where every variant in the textual tradition gets a value that indicates how good a leitfehler it is (obtained through the comparison of the distributions of this variant in conjunction with each other variant). This list can be pruned at a certain point. We do so using roughly the top third of the list (initial parameter setting of Roelli and Bachmann (2010), numerically 400) and by that token can achieve a binary classification for each variant: good leitfehler (1), bad leitfehler (0). We encode the manuscripts as bitvectors where each dimension corresponds to one leitfehler. The number of positions equals thus the number of position bound variants in the corpus (the sum of the number of variants per aligment position where variation occurrs).

For the clustering, we again detect for each manuscript those manuscripts, which do share a certain percentage of leitfehler (95%). For comparison, we cluster the bitvectors also with Simple K-means (SKM) and Farthest First (FF) as implemented in Weka (Hall et al., 2009).

3.1. Resolving Incompatible Groups

There is one problem emerging from detecting threshold dependent clusters: The outcome of this step can be an *incompatible* grouping where at least one manuscript belongs to more than one cluster. The reason for this is, that any manuscript m can share t or more % with another manuscript a but at the same time t or more % with yet another manuscript b, where the variants both share with m are not the same ones. Consequently, a and b can share less than $t\%$. The same problem is known in biology, where DNA sequences can similarly share nucleic acids of their multiple sequence alignment. Huson et al. (2010) describe how to transform an incompatibility graph into a phylogenetic network by a *divide and conquer* strategy using decomposition. This can be applied here similarly: incompatible clusters (clusters which share at least one manuscript) are all merged into a single cluster.

3.2. Inferring an Archetype for a Cluster

West (1973, p.32 ff) gives some guidelines how to determine if in a group of several (clustered) manuscripts the group internal archetype (root) is to be considered contained. This is the case if there is one manuscript among them, which has no idiosyncratic variants – it will have a high probability of being the group internal root. If there is such a group internal root we take it as the groups ancestor. If this is not the case, Hoenen (2015b) describes how along the lines of a known stemma, the archetype text can be reconstructed. The algorithm is rule-based and simply takes the majority variant at each position of the alignment for the archetype text and in case there are more than one

[8] A fourth set has been used scarcely in the literature (Robinson, 2015) and the correct stemma does not name lost node labels. For a fifth dataset (Hoenen, 2015a) we report results only briefly, since the dataset is very small and designed under more to test for rootedness.

equally most probable variants just writes them all. We use the same algorithm in order to compute an archetype text for each cluster where no group internal archetype[9] exists. Now, we have an archetype text for each group and can build a stemma bottom-up by connecting the archetype of each group with the members by an edge.

3.3. Updating the Corpus

This procedure is repeated. Instead of all manuscript texts as in the first round, in the second round all members of all clusters are replaced by their archetype text. The vectors and variants for this text are updated/generated if necessary. Conflicting information for any one position as to whether or not a variant is a good leitfehler (numerical tie) leads to that position being ignored. Then, the new text corpus is clustered again until either all clusters have only one member or until there is only one cluster. In this way a stemma is built bottom-up. For an illustration of the whole process, see Figure 2.

In case a cluster has for instance 2 manuscripts, there is no easy way to decide which variant to assign to a reconstructed ancestral text. Yet, it is important to reconstruct an archetype with some text at all positions of the alignment. The algorithm pragmatically generates an archetype, which in such cases accumulates all equiprobable variants (if they are most probable). That means, that each reconstructed text may carry more than one variant at each position. In the final root of the stemma, the philologist may then start to decide which variant is the correct one. For future research more sophisticated computational aids are thinkable. For instance one can use some language model and then let some algorithm such as Viterbi find the most probable final text. However, this will only produce variants already extant in the manuscripts, but not such, which have been ancestral, but irretrievably altered. If for instance the word <bath> had been changed to <path> in an early copy event above the archetype and all earlier manuscripts having the correct variant had been lost, the only way to reestablish this variant would be through more sophisticated reasoning involving analyses of individual writers language and orthographic systems, sound shifts, orthographic conventions and the like. In philological literature such cases are abundant. Additionally, meta data such as the age of a manuscripts material (although this doesn't have to coincide with the age of the manuscript text) can and should be taken into account, especially since many manuscripts have been encoded in TEI schemata, which is not the case for the artificial traditions.[10]

[9] Here, archetype refers to the root of the cluster not the whole tree.

[10] According to Barabucci et al. (2014): "Non sono mai state sperimentate, tuttavia, soluzioni per generare stemma a partire da documenti codificati in XML." translated by the author as: "Anyway, there have never been experiments of stemma generation starting from XML encoded documents." Especially meta data should not be dismissed by automatic stemmatology as an important source of for instance conditions on th tree topology.

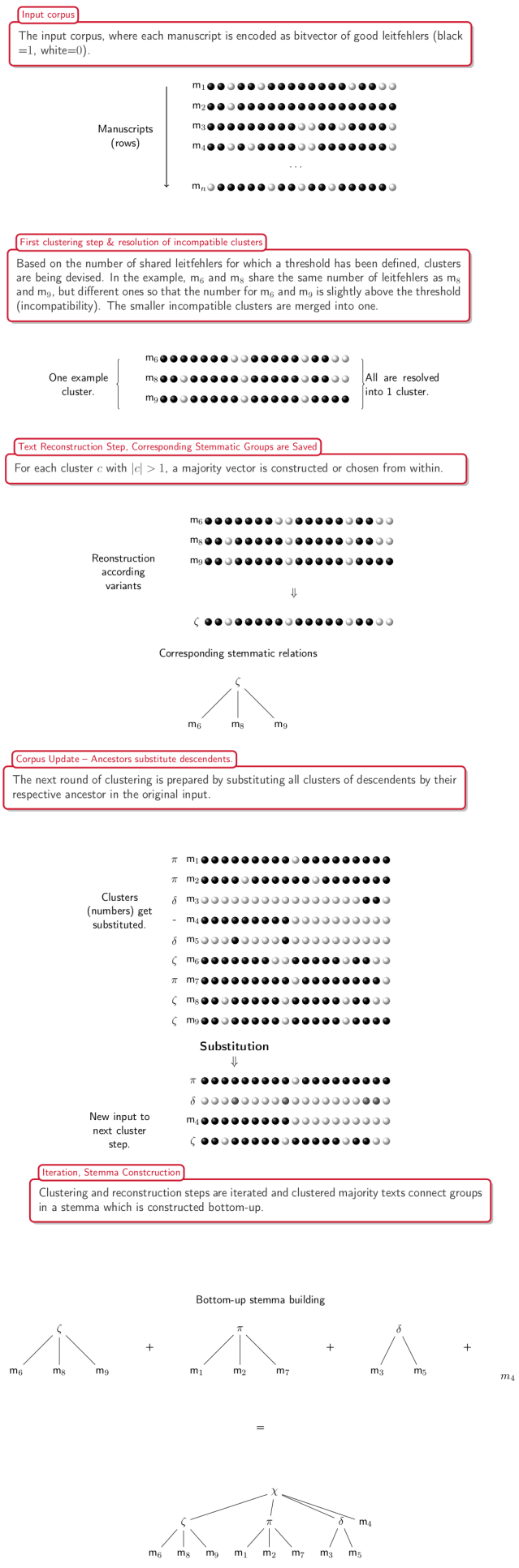


Figure 2: Iterative archetype clustering illustrated.

3.4. Evaluation

For evaluation, we evaluate both stemmata and archetypes. This is necessary because one stemma can be consistent with several archetypes and one archetype can be explicable through several stemmata. For stemmata, we use the Average Sign Distance or ASD, a measure introduced by Roos and Heikkilä (2009).

$$u(A,B,C) = 1 - \frac{1}{2}|sign(d(A,B) - d(A,C)) - sign(d'(A,B) - d'(A,C))|$$

A, B and C are nodes present in both the true and the estimated stemma, $d(A,B)$ is the distance of the two nodes in the true stemma defined as the number of edges on the shortest path between them, $d'(A,B)$ the same distance for the estimated tree. $sign(d(A,B) - d(A,C))$ returns so to speak only the sign, discarding length, thus -1 if $d(A,B) < d(A,C)$, 1 in the opposite case and 0 if both are equal. The index equals 1 if both stemmata agree and 0 if they differ ($\frac{1}{2}$ in case of partial agreement) and is computed and turned into a proportion for all such triples. Roos and Heikkilä (2009) provide a python script for ASD evaluation.[11]

For evaluation of the archetype text on the other hand, we use the evaluation introduced by Hoenen (2015b). It gives the percentage of correct variants at all variant alignment positions (ignoring those where all manuscripts agree). We term this the variant hit rate VHR. If a reconstructed archetype has p equiprobable variants at a position, where one is the correct one, $\frac{1}{p}$ is assigned.

Additionally, to assess historical circumstances, for each number from 1 to $n-1$ (n, the number of ms in the corpus) that many randomly chosen manuscripts were deleted to simulate historical loss. This process was repeated 100 times for each tradition and evaluation results (over all random samples and all loss numbers) averaged.

4. Results

Average ASD results of all evaluation (loss) scenarios for the single traditions are listed in Tables 2, 3, and 4.

Farthest First and Simple-K-Means were initiated with an estimated generation depth from the clustering steps of the former two algorithms (for comparability) and did a little worse than leitfehler and word based percentages. Practically, the archetype texts coincide in almost every instance for both. However, the stemmas of both differ and on the whole on our data sets and with our randomization scheme, the Farthest First algorithm produced a slightly better average result.

The best ASDs were reasonable with NB: 70.98, PRZ: 73.53, HNR: 64.53 and due to a different approach to simulating loss not directly comparable to previous results. For NB the result was obtained at a loss of 2 manuscripts with the threshold clustered leitfehler vectors; for the Parzival and the Heinrichi traditions by Simple K-means at a loss of 2 manuscripts. Overall, the best performing algorithm for archetype text production was the one based on threshold clustered leitfehler. As for the stemmas, the word based token difference percentage produced stable results.[12] The overall average over all traditions (including TASCFE) can be seen in Table 5 but is skewed towards TASCFE, which contributed 4 traditions.

[11] While on the level of path comparison operating on distance, in terms of the overall manuscript comparison, the ASD is rather a similarity and referred to as Average Sign Similarity by other authors.

Algorithm	VHR	ASD
Leitfehler	0,91	58.86
Word perc	0.91	56.80
FF	0.42	56.95
SKM	0.42	57.78

Table 2: Notre Besoin.

Algorithm	VHR	ASD
Leitfehler	0.85	59.50
Word perc	0.77	59.68
FF	0.85	59.22
SKM	0.85	59.16

Table 3: Parzival.

5. Discussion

The results, given that many scenarios of loss have been conducted need some brief interpretational intervention. A single specific scenario where manuscript loss is simulated may result in remaining texts which are all quite bad that is to say which differ considerably in their texts from the true original. If this is the case, reconstruction can only result in a bad archetype even in the best case. Naturally, averaging over loss scenarios with partly heavy loss will thus deteriorate average values but may be more faithful to historical circumstances. However, Trovato (2014) conducting a case study on historical loss finds that for many scenarios an amount of loss above 73% is expectable, p. 108. The data sparsity and the small number of copy generations in the artificial datasets indicate that much more research may be necessary in order to reach more conlusive results. However, this is rather secondary to the main aim of this paper, which is to present a method, which produces both stemma and archetype in conjunction.

The algorithm, apart from the fact that it does not disambiguate between variants which are equiprobable or inserts new ones based on the observed ones, is oriented at philological practice and is thus hopefully more straightfor-

[12] For the TASCFE artificial tradition, which is both quite small (137 alignment positions) and unusual (written in Persian, containing 4 subversions), we found similar values in ASD between 63 and 73% and VHRs between 0.68 and 0.88.

Algorithm	VHR	ASD
Leitfehler	0.82	55.47
Word perc	0.78	56.59
FF	0.61	55.04
SKM	0.61	55.63

Table 4: Heinrichi.

Algorithm	VHR	ASD
Leitfehler	0.82	61.03
Word perc	0.77	62.99
FF	0.74	60.98
SKM	0.74	60.39

Table 5: Average over all traditions including TASCFE.

wardly interpretable and adjustable in its results and inner workings. Along the lines of this it produces both a stemma and an (undisambiguated) archetype. Previous methods have produced either separately (independently), or automatic archetype reconstruction followed automatic stemma generation. The production in conjunction is not only oriented at philological practice, it is an essentially stronger hypothesis than either only an archetype text (which can be consistent with several stemmata) or a stemma (which can be consistent with different archetypical texts). By this token it can claim to automatically solve both tasks leaving for manual postprocessing much fewer work than previous attempts if one assumes the philological users want to have both: stemma and archetype. The herein presented first implementation has shown to produce reasonable results, which can and should be improved in future research (especially concerning the stemmatic fit). As such this may make it plausible that future developments can achieve a high degree of automatization in the task albeit with the to be affronted danger of using a blackbox and overseeing crucial givens in the data. Such a position has been challenged in the philological literature repeatedly, see for instance (West, 1973; Griffith, 1984, p. 72, p. 83). If such a high degree of automatization is desirable is another question. The same method of iterative clustering has been applied to produce multilingual stemmata (Hoenen, 2017).

6. Conclusion

Cameron (1987) stated: "Reconstruction is a serious business, and the only point in studying manuscripts at all." A simple algorithm has been presented, which reconstructs archetypical texts automatically. The reconstuctions have been compared to the true archetype for artificial traditions, finding that they performed well.

Bottom-up text reconstruction and stemma building are modelled as alternating and interdependent processes, much like in philological practice. A clustering step clusters together groups of similar texts for which a common ancestral text is being reconstructed. Substituting the children of this text by it, subsequent clustering steps are performed until the program arrives at a stemma and an archetype text. As a criterium for clustering either the percentage of different variants or the number of shared leitfehler was taken. Alternatively, well-known clustering algorithms performed the cluster steps. Averaging over all possible conditions of loss, the produced archetypes had a fair hit rate but the stemmas evaluated not extremely well on average (reasonably in the best scenarios). The main aim of the paper has been to show that automatic archetype production and stemma building can be combined into an interdependent single tasks.

7. Acknowledgements

The author would like to thank Prof. Dr. Alexander Mehler and Prof. Dr. Jost Gippert for their advice, feedback and support especially in the context of the CEDIFOR project as well as the organizers of LREC, the reviewers and last but not least the readers.

8. Bibliographical References

Barabucci, G., Di Iorio, A., and Vitali, F. (2014). Stemma codicum: analisi e generazione semi-automatica. *Quaderni DigiLab*, 3(1):129–145.

Baret, P., Macé, C., and P. Robinson, e. (2004). Testing methods on an artificially created textual tradition. In *Linguistica Computationale XXIV-XXV*, volume XXIV-XXV, pages 255–281, Pisa-Roma. Instituti Editoriali e Poligrafici Internationali.

Bédier, J. (1928). La tradition manuscrite du 'Lai de l'Ombre': Réflexions sur l'Art d'Éditer les Anciens Textes. *Romania*, 394:161–196, 321–356.

Cameron, H. D. (1987). The upside-down cladogram: problems in manuscript affiliation. In *Biological Metaphor and Cladistic Classification: an Interdiscplinary Approach*, pages 227–242. University of Pennsylvania.

Griffith, J. (1984). A three-dimensional model for classifying arrays of manuscripts by cluster analysis. *Studia Patristica*, XV(I).

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD Explorations*, 11(1):10–18.

Haugen, O. E. (2015). The silva portentosa of stemmatology Bifurcation in the recension of Old Norse manuscripts. *Digital Scholarship in the Humanities*, 30(2).

Hoenen, A., Eger, S., and Gehrke, R. (2017). How many stemmata with root degree k? In *Proceedings of the 15th Meeting on the Mathematics of Language*, pages 11–21.

Hoenen, A. (2015a). Das artifizielle Manuskriptkorpus TASCFE. In *DHd 2015 - Von Daten zu Erkenntnissen - Book of abstracts*. DHd.

Hoenen, A. (2015b). Lachmannian archetype reconstruction for ancient manuscript corpora. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*.

Hoenen, A. (2017). Beyond the tree - a theoretical model of contamination and a software to generate multilingual stemmata. In *Book of Abstracts of the annual conference of the AIUCD 2017, Sapienza, Rome*. AIUCD.

Huson, D. H., Rupp, R., and Scornavacca, C. (2010). *Phylogenetic networks*. Cambridge University Press.

Koppel, M., Michaely, M., and Tal, A. (2016). Reconstructing ancient literary texts from noisy manuscripts. In *CLfL@NAACL-HLT*.

Maas, P. (1937). Leitfehler und Stemmatische Typen. *Byzantinische Zeitschrift*, 37(2):289–294.

Marmerola, G. D., Oikawa, M. A., Dias, Z., Goldenstein, S., and Rocha, A. (2016). On the reconstruction of text

phylogeny trees: Evaluation and analysis of textual relationships. *PloS one*, 11(12):e0167822.

Nassourou, M. (2013). *Computer-Supported Textual Criticism: Theory, Automatic Reconstruction of an Archetype*. BoD–Books on Demand.

Robinson, P. (2015). Four rules for the application of phylogenetics in the analysis of textual traditions. *Digital Scholarship in the Humanities*.

Roelli, P. and Bachmann, D. (2010). Towards Generating a Stemma of Complicated Manuscript Traditions: Petrus Alfonsi's Dialogus. *Revue d'histoire des textes*, 5(4):307–321.

Roelli, P. and Macé, C. (2015). Parvum lexicon stemmatologicum. a brief lexicon of stemmatology.

Roelli, P. (2014). Petrus alfonsi, or on the mutual benefit of traditional and computerised stemmatology. In T. L Andrews et al., editors, *Analysis of Ancient and Medieval Texts and Manuscripts: Digital Approaches*, pages 43–68. Brepols, Turnhout.

Roos, T. and Heikkilä, T. (2009). Evaluating methods for computer-assisted stemmatology using artificial benchmark data sets. *Literary and Linguistic Computing*, 24:417–433.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.

Shillingsburg, P. (2017). *Textuality and Knowledge: Essays*, volume 27. Penn State Press.

Spencer, M., Davidson, E. A., Barbrook, A. C., and Howe, C. J. (2004). Phylogenetics of artificial manuscripts. *Journal of Theoretical Biology*, 227:503–511.

Timpanaro, S. (2005). *The Genesis of Lachmann's Method*. University of Chicago, Chicago.

Trovato, P. (2014). *Everything You Always Wanted to Know about Lachmann's Method, A Non-Standard Handbook of Genealogical Textual Criticism in the Age of Post-Structuralism, Cladistics, and Copy-Text*. libreriauniversitaria.it.

West, M. L. (1973). *Textual Criticism and Editorial Technique: Applicable to Greek and Latin texts*. Teubner, Stuttgart.

Building A Handwritten Cuneiform Character Imageset

Kenji Yamauchi[1], **Hajime Yamamoto**[2], **Wakaha Mori**[3]
[1]TIS Inc., Japan, yamauchi.kenji@tis.co.jp
[2]Kyoto University, Japan
[3]The Institute for Cultural Studies of Ancient Iraq.

Abstract

Digitization of cuneiform documents is important to boost the research activity on ancient Middle East and some projects have been launched in around 2,000. However, the digitization process is laborious due to the huge scale of the documents and no trustful (semi-)automatic method has established. In this paper, we focused on a cuneiform document digitization task, realization of Optical Character Recognition (OCR) method from the handwritten copies of original materials. Currently, as the first step toward development of such methods, we are constructing a handwritten cuneiform character imageset named with professional assistance. This imageset contains typical stroke patterns for handwriting each frequently appearing cuneiform character and will be able to support the development of handwritten cuneiform OCR system.

Keywords: Cuneiform, OCR, Dataset

1. Motivation

Cuneiform characters (signes) were used to write several languages (e.g. Sumerian, Akkadian) for about 3,000 years in ancient Middle East (so-colled Mesopotamia). Each character superficially contains wedges (called *cuneus* in Latin) which you can see as triangular shape. Usually, the ancient scribes adopted the clay tablets as a material on which they drawn the signs. The characters were similar to Old Japanese in terms of the orthography (both phonogramatically and ideogramically used) and the number of character classes (almost 600).

Because of the diversity of category and written languages, cuneiform documents are linguistically and historically important. Since the amount of the documents is huge (at least 300,000 documents have been published, and many remain unpublished), digitization of cuneiform documents are indispensable to conduct academic research efficiently.

Some digitization projects of cuneiform documents have been launched in around 2000. For example, some studies tried to 3D scanning of original clay tablets (e.g. Digital Hammurabi Project (Watkins and Snyder, 2003)) and others treated transcription and translation (e.g. Cuneiform Digital Library initiative (CDLI) [1], Open Richly Annotated Cuneiform Corpus (ORACC)[2]).

To facilitate those digitization projects, (semi-)automatic digitization methods are required. Unfortunately, current digitization projects have been conducted manually. Few studies tried to detection of a character class from handwritten copies of original tablets (Massa et al., 2016; Rothacker et al., 2015), grammatical analysis (Homburg and Chiarcos, 2016) and automatic machine translation (Pagé-Perron et al., 2017). However, reliable (semi-)automatic digitization methods have been not established yet.

In this paper, we introduce our ongoing project, a construction of an imageset named Handwritten Cuneiform Character Collection (HCCC)[3]. HCCC is intended to be an image collection of frequently used cuneiform characters written with some stroke patterns and to be used for Optical Character Recognition (OCR) from handwritten copies of original clay tablets.

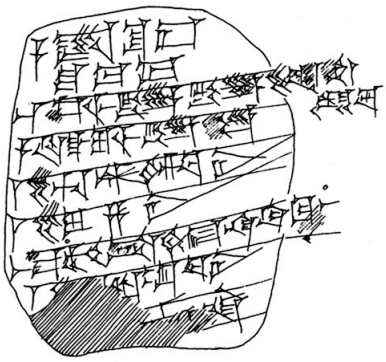


Figure 1: An example of handwritten copy of cuneiform tablet

Although our final goal is to develop an end-to-end method to directly produce transliterated scripts from original tablets, some limitations such as the condition of materials of the tablets prevents us from accessing the tablets. Fortunately, we can obtain handwritten copies of original tablets depicted by the scholars such as Figure 1[4] and many of such copies have not been transcripted yet. For example, there are about 70,000 digitally untranscribed hand copies out of all approximately 300,000 documents registered in CDLI.

Therefore, we set first goal to achieve OCR from the handwritten copies of original tablets and are constructing the imageset as explained above. This image set is the first attempt to collect handwritten cuneiform characters as far as we surveyed despite of the small scale of the dataset (at most approximately 200 images per chararcter class) like Omniglot dataset (Lake et al., 2011) for now.

[1] `http://cdli.ucla.edu/`

[2] `http://oracc.museum.upenn.edu/`

[3] Will be available on `https://github.com/yustoris/hccc`

[4] Excerpted from `http://cdli.ucla.edu/search/archival_view.php?ObjectID=P101022`

2. Imageset Construction Process

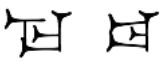
In this section, we describe the ongoing construction process of HCCC.

2.1. Target Classes and Glyphs

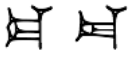
We began by determining target classes and glyphs for the construction. As denoted previously in Section 1., the number of cuneiform character classes are up to hundreds. However, most of them are ligatures of basic characters or less frequently appeared. For example, GU_7, *to eat*) is composed of (KA, *mouth*) and (GAR, *bread*). Therefore we'll focus on most frequently appeared and probably most basic classes. More precisely, we selected most frequently appeared 50 character classes based on the result of counting occurence of the character classes on the available cuneiform translitelized corpora, Electronic Text Corpus of Sumerian Literature (ETCSL) [5]. The number of characters classified into the chosen 50 classes accounts for roughly 70% of the total character appearances.

Glyphs of cuneiform characters are also various depending on ages in which the characters were used, because the characters' shape had been gradually simplified during thousands years. In this work, we limited to most oldest (by Ur III, about 2,000 BC) ones. Since the oldest glyphs have complex shapes on each character class, we can distinguish classes more easily compared to newer glyphs as shown in Figure 2. This discernibility make us possible to collect images with less ambiguity between different classes.

However, there is one exception, i.e. we cannot distinguish E_2 and KID clearly without any context. Thus, we classify those two characters into a same class.



(a) Old glyphs (Around 2,000 BC)



(b) New glyphs (Around 600 BC)

Figure 2: Comparison on glyphs of MA (left) and BA (right)

2.2. Resource

The original resource from which we obtained the candidates is openly accessible one, academic handcopy collections published in the 19th century and handcopies available on CDLI [6]. Most of the candidates have been collected from the former, and supplementary employ the latter when we cannot find characters which belong to the target classes.

We considered stroke variation depending on authors of handcopies. This stroke variations mainly causes by difference of the way how to write a cuneus of the character. For instance, Figure 3 shows three examples of such stroke variations for a same character class KA .

When we tackle to OCR from the handcopies, we have to treat that variations. Therefore, we decided to include several stroke variations in each character class as possible as we can.



(a) Variation 1 (b) Variation 2 (c) Variation 3

Figure 3: Stroke variations in same character class

2.3. Generation of Candidates

We firstly derived *candidate character images* from given resource. Each candidate is expected to be an image which contains exact one cuneiform character. They were heuristically and automatically clipped from original handcopied tablet images.

2.3.1. Generation Method

The adopted heuristic method consists of following three procedures.

Firstly, we derived regions which are probably drawn tablets from given resource in which each page has several handcopied tablets. We converted original images into greyscale ones and clipped regions enclosed by contours detected by an algorithm described in Suzuki and Abe (1985). From those regions, we filtered out regions whose areas are under a fixed value.

After extracting tablet regions, we erased extra lines separating section as shown in Figure 4 by detecting lines whose length are longer than fixed rate of width of the tablet images. This erasing process is needed because those separation lines were commonly used by cuneiform scribes and are noises for extracting characters.

Finally, we extracted character candidates from the tablet images. We applied Gausian smoothing and detecting regions by enclosed contours with the same method when we used for extracting the tablet images. After filtered out areas whose area are smaller than a heuristically defined value and consequently the rest candidates were resized to 64x64. As a result of those procedures, we derived 147,010 character candidate images with 64x64 and grayscale.

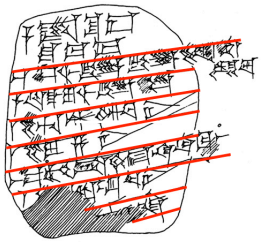


Figure 4: Figure 1 colored separation lines by red

2.3.2. Simple Filtering

Since these candidates are generated by automatic procedures, we can easily obtain a large number of character images. While we successfully got "complete" candidates, i.e. which can be clearly recognized as a target character, many candidates are "incomplete". That is, (Type-1) just noises, (Type-2) complete one character but contains noises,

[5] http://etcsl.orinst.ox.ac.uk/

[6] Fully listed in https://github.com/yustoris/hccc

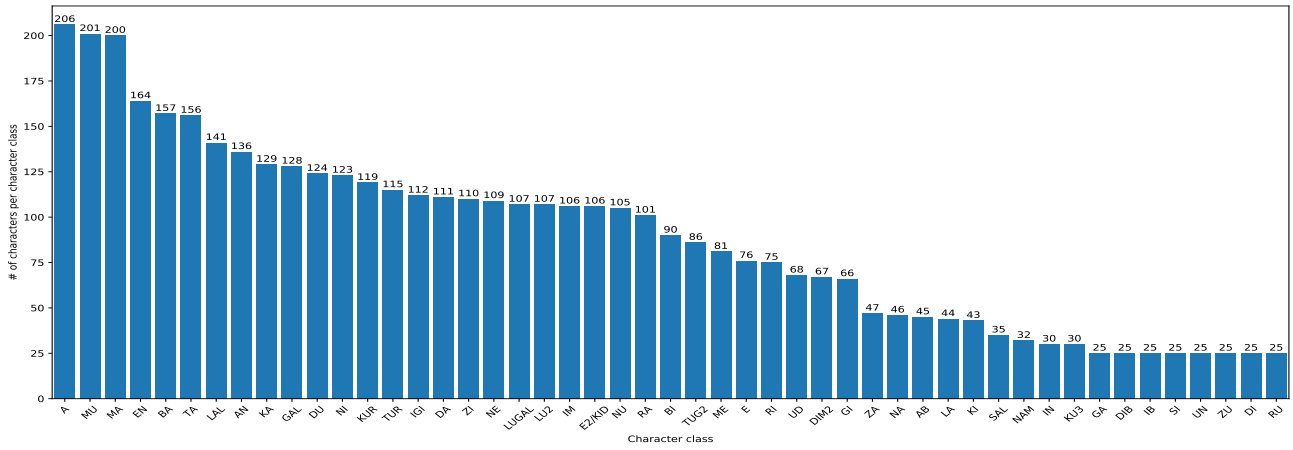


Figure 5: Distribution of currently collected imageset

(Type-3) only part of one character and (Type-4) include multi characters. In this work, we decided to filter out all of those incomplete candidates except (Type-2) because we can denoise and don't have to adjust aspect of the images again. For the same reason, the rotated images also are not included in the "incomplete" images.

From derived candidates images, we manually and roughly filter out (Type-1) images after clustered all the images into classes whose size is temporary fixed. This fixed cluseter size is determined considering the noise images denoted above. The clusering was conducted by converting the images into Bag of Visual Words (BoVW) in which the features are extracted by AKAZE (Alcantarilla et al., 2013) and unsupervisedly classified the converted images into 120 groups with with K Nearest Neighbors. Many of (Type-1) images were clustered into same classes, so we could easily recognize the images to be exclulded. We found 31 classes in which almost all candidate images were (Type-1) ones. Consequently, we obtained 118,801 roughly filtered candidate images.

2.4. Labelling and Cleaning

After deriving the candidates filtered by the simple method explained above, we label and clean them manually. Because of a large number of incomplete candidates and limited workers, we set a first goal to collect confirmed images at small scale, i.e. at most approximately 200 images per target character class. In future, we'll conquer this scantiness by (semi-)automatic collection methods.

We annotate a character class on complete and (Type-2) incomplete candidates. After labelling, we denoise (Type-2) candidates and fix the direction of the candidates which are rotated above 90 degrees manually.

Futhermore, when we can not collect enough images for a character class from the handcopy collections, we will manually clip the target character from handcopies available on CDLI and also convert it to 64x64 gray scale format following to the auto generation method described in Section 2.3.1..

3. Current Status

Although all collection procedures written above has not been fully completed yet, we have already obtained 4,358 annotated images as of Sep. 9, 2017. More than 100 images are collected for 26 out of 50 character classes, maximum number of images for a class is 206 and minimum is 25. Figure 5 describes more detailed statistics of current number of collected images for each character class. Each bar describes the number of collected images for each character class and the bars are sorted in descending order.

4. Tasks to Enhance Imageset

As described in Section 3., the size of currently built dataset is limited. To scale up the dataset, we have to overcome some tasks.

4.1. Detection of Touched and "Complex" Characters

The heuristic method used in this paper can not detect one character from touched multi characters. This weak point resulted out (Type-4) incomplete candidates which probably were generated because of connecting the two or more characters by Gausian smoothing.

On the other hand, some characters such as IN  consist of multi parts which tend to be separated by simple Gausian somoothing based character detection. Therefore, it is also needed to detect these "complex" characters as one character precisely.

To tackle to those touched and "complex" characters, it may be effective to apply some existing character recognition methods such as Rothacker et al. (2015) with training features for each target character class from current dataset.

4.2. Efficient Classification

Other task is to classify derived candidates into the target character classes more effectively. The candidates are too many to conduct the collection of characters by hand. Therefore, it is needed to combine automatic method and manual correction to achive more efficient and precise image collection.

However, regular supervised classification method such as ResNet (He et al., 2015) is not suitable to apply the ongoing dataset building because these methods require plenty of training data and we have only limited scale dataset now. To resolve this problem, we will refer some studies tries to develop classifiers from tiny training data. In particular,

Koch et al. (2015) applied siamnese convolutional network to treat One Shot Lerning, which is a task creating classifier from only one sample for each class as we human beings are often able to learn object shape from only one instance of the target object.
We will apply those One Shot Lerning methods to develop reliable classifier from current small dataset and to conduct compilation of character images more efficiently.

4.3. Collection of Other Ages Glyphs

As denoted in Section 2.1., we limited target age for collection. However, for constructing more practical dataset, we have to collect other glyphs of other ages.
The important and typical glyphs are ones used in Neo-Assyrian and Hittite in addition to the collected glyphs in this paper. Thus, the future collection targets are those two glyphs. Since those glyphs are relatively simplified and the total number is smaller compared to oldest ones, we will have to tackle to the ambiguity between different character classes are higher as previously described in Figure 2b.

5. Conclusion and Future Work

We conducted a first attempt to build handwritten cuneiform character imagest, HCCC. The scale of built imageset HCCC is currently small compared to several large handwritten character dataset, such as CASIA dataset (a large handwritten Chinese character dataset (Liu et al., 2011)) or MNIST for now. However, the way to enrich the imageset has been already suggseted as explained in Section 4..
After finishing imageset construction, we will try to achive cuneiform character recognition from given handwritten source by image processing using HCCC and by considering linguistic context. The reason why we is to tackle to diversity of cuneiform languages and ambiguity of character glyph. In particular, the latter problem is needed to treat both graphically and linguistically. Each cuneiform character can be used both phonogramatically and ideogramically, and some cuneiform characters are too similar to distinguish. Furthermore, many excavated documents often lacks some characters due to poor material condition. Those graphical ambiguity and incompleteness would be solved by taking into account linguistic context such as proposed in Dĥondt et al. (2016).

6. Acknowledgements

This work was partly supported by JSPS KAKENHI Grant Number 17K13549 (Grant-in-Aid for Young Scientists (B)) and 16K02642 (Grant-in-Aid for Scientific Research (C)).

7. References

Alcantarilla, P. F., Nuevo, J., and Bartoli, A. (2013). Fast explicit diffusion for acceleratedfeatures in nonlinear scale spaces. In *Proc. of the 2013 BMVC*.

Dĥondt, E., Grouin, C., and Grau, B. (2016). Low-resource ocr error detection and correction in french clinical texts. In *In Proc. of 7th LOUHI*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. In *Proc. of CVPR*.

Homburg, T. and Chiarcos, C. (2016). Akkadian word segmentation. In *Proc. of LREC 2016*.

Koch, G., Zemel, R., and Salakhutdinov, R. (2015). Siamese neural networks for one-shot image recognition. In *Proc. of Deep Learning Workshop, ICML'15*.

Lake, B. M., Salakhutdinov, R., Gross, J., and Tenenbaum, J. B. (2011). One shot learning of simple visual concepts. In *Proc. of the 33rd ACCSS*.

Massa, J., Bogacz, B., Krömker, S., and Mara, H. (2016). Cuneiform detection in vectorized raster images. In *21th CVWW*.

Pagé-Perron, E., Sukhareva, M., Khait, I., and Chiarcos, C. (2017). Machine translation and automated analysis of the sumerian language. In *Proc. of the 2017 LaTeCH-CLfL*.

Rothacker, L., Fisseler, D., Müller, G. G. W., Weichert, F., and Fink, G. A. (2015). Retrieving cuneiform structures in a segmentation-free word spotting framework. In *Proc. of the 3rd IWHDIP*.

Suzuki, S. and Abe, K. (1985). Topological structural analysis of digitized binary images by border following. *CVGIP*.

Watkins, L. and Snyder, D. (2003). The digital hammurabi project. In *Proc. of MW*.

PDF-to-Text Reanalysis for Linguistic Data Mining

Michael Wayne Goodman, Ryan Georgi, Fei Xia
Linguistics Department
University of Washington
goodmami@uw.edu, rgeorgi@uw.edu, fxia@uw.edu

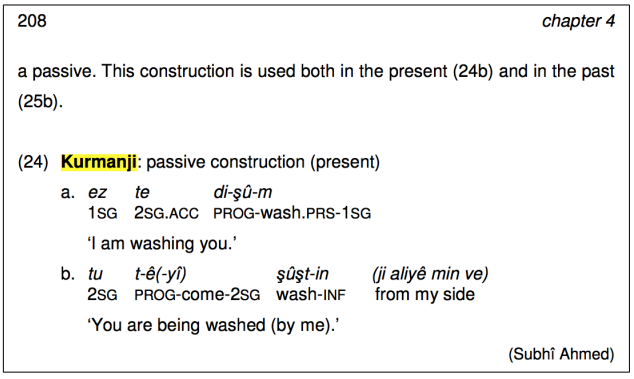
Abstract

Extracting semi-structured text from scientific writing in PDF files is a difficult task that researchers have faced for decades. In the 1990s, this task was largely a computer vision and OCR problem, as PDF files were often the result of scanning printed documents. Today, PDFs have standardized digital typesetting without the need for OCR, but extraction of semi-structured text from these documents remains a nontrivial task. In this paper, we present a system for the reanalysis of glyph-level PDF-extracted text that performs block detection, respacing, and tabular data analysis for the purposes of linguistic data mining. We further present our reanalyzed output format, which attempts to eliminate the extreme verbosity of XML output while leaving important positional information available for downstream processes.

Keywords: Low Resource Languages, Interlinear Glossed Text (IGT), Corpus Creation

1. Introduction

A great deal of linguistic information exists online in the form of academic publications. ODIN, the Online Database of INterlinear text (Lewis and Xia, 2010) was created upon this premise, a resource which makes available language data for approximately 1,500 languages, including linguistic glosses and resource-rich language translations (Lewis et al., 2015). The data targeted by ODIN is Interlinear Glossed Text, or IGT, a semi-structured data format for presentation of linguistic examples, as shown in Figure 1. This data has been shown to have interesting characteristics, making some NLP analysis possible for resource-poor languages (Lewis and Xia, 2008; Georgi et al., 2014; Georgi et al., 2015).


208 *chapter 4*

a passive. This construction is used both in the present (24b) and in the past (25b).

(24) **Kurmanji**: passive construction (present)

a. *ez* *te* *di-şû-m*
1SG 2SG.ACC PROG-wash.PRS-1SG
'I am washing you.'

b. *tu* *t-ê(-yî)* *şûşt-in* *(ji aliyê min ve)*
2SG PROG-come-2SG wash-INF from my side
'You are being washed (by me).'

(Subhî Ahmed)

Figure 1: An IGT instance of Kurmanji (kmr) *in situ* from (van de Visser, 2006).

All the IGT instances in ODIN, including the one in Figure 1, were extracted from linguistic articles that were distributed electronically as Portable Document Format (PDF) documents, a format developed by Adobe Systems for the purposes of standardizing document display and typesetting across platforms. As this format was primarily designed to "communicate visual material between different computer applications and systems" (Warnock, 1991), re-extracting the displayed text for automated systems was not an intended aspect of the format design. In the PDF specification, text is represented as glyphs of a specified font in a "Character Space" coordinate system, embedded within a "Text Space" renderer (Bienz et al., 1997). One of the downsides of this document structure is that the internal structure of a PDF file gives text as glyph-coordinate pairs, and guarantees only the positioning of text rendered on the page. It does not guarantee that the order of the encoded text resembles the order in which it is intended to be read.

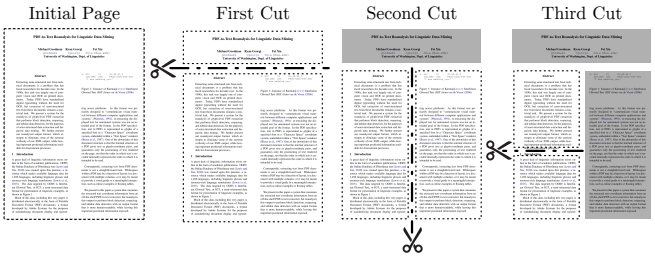


Figure 2: An illustration of the XY-Cut algorithm recursing on a document. The regions created by the current cut (if any) are represented by a dotted border, while regions not under consideration at the current recursion are shaded.

Consequently, extracting text from PDF documents is not a straightforward task. Whitespace within a PDF may be purely a function of layout, as in a document with multiple columns, or it may be meant to provide a cue to meaningful structural deviations in the text, such as inline examples or floating tables.

We present in this paper a system that consumes the extracted text-coordinate information from an off-the-shelf PDF-to-text converter, but reanalyzes this output to perform block detection, respacing, and tabular data analysis. The output format of this system is more human-readable than the verbose xml format produced by the off-the-shelf converters, while making the important positional information available to downstream processes.

2. Previous Work

In the 1980s and 1990s, as access to electronic resources increased, researchers began investigating ways to convert previously printed content into electronic content that could be indexed and searched more easily. A part of the OCR task of transforming pixels into electronic characters was detecting what portions of the page were contiguous blocks. (Nagy and Seth, 1984) introduced the recursive XY-Cut algorithm, with (Nagy et al., 1992) refining the technique for the specific domain of technical articles. For an informative comparison of different algorithms for layout analysis, please see (Shafait et al., 2006).

2.1. The XY-cut algorithm

The XY-cut algorithm works by searching for the largest rectangle of whitespace (or "valley") that runs the length of the current region of the page either vertically or horizontally. This region is then "cut" into two smaller regions, and the algorithm recurses until no more cuts can be found. Rather than cutting only on valleys of complete whitespace, a whitespace threshold can give some tolerance to noise within a valley. Another threshold on the minimum region size to cut can prevent the algorithm from cutting too far. Figure 2 provides an illustration of the first several cuts that the algorithm might make on a sample document.

2.2. Existing PDF-to-text converters

There are several existing PDF-to-text converters, which extract text from PDFs and produce output files with glyph-position information in easily parsable XML formats, such as the open-source PDFMiner (Shinyama, 2016) and the commercial product from PDFLib called Text and Image Extraction Toolkit (TET) (PDFLib GmbH, 2015). The XML output is extremely verbose; for instance, the original PDF document containing Figure 1 is 3MB in size, but the converted XML output of TET, TETML[1] is 58MB uncompressed, 5.5MB with gzip compression. Figure 3 gives an example of the representation of the `PROG-wash-PST-2SG` token from Figure 1 in TETML format.

A third converter we have tested is the pdftotext utility of the Poppler PDF library[2]. A nice property of pdftotext is that it can output text with whitespace to approximate the layout of the page (aka *respacing*). However, the respaced text was often corrupted, with lines of text being split, or with columns becoming misaligned.

Table 1 compares the three converters with respect to features that we find useful when extracting text from scientific documents. All produce markup (XML or HTML) and plain-text formats, where the markup formats contain the coordinates of glyphs and fine-grained blocks (i.e., at word or line-level).

Utility	Respace	Coords	Block	Unicode
pdftotext	✓	✓	✓	basic
PDFMiner		✓	✓	basic
TET		✓	✓	advanced

Table 1: Features of three existing PDF-to-text converters

3. Extraction and Reanalysis

When designing our PDF-to-text system (called Freki[3]), we elected not to reimplement PDF-to-text extraction but instead built on top of off-the-shelf converters that produce the XML format. From the verbose XML format, Freki identifies larger blocks (e.g., a paragraph or an IGT) and performs respacing. It also makes modifications to the XY-cut algorithm.

3.1. Block Detection

The core of our layout analysis is in block detection, which is a two-stage algorithm. In the first stage, we create a 2D array where non-zero values represent the bounding boxes of text tokens in the PDF, then analyze this array for blocks. We aim to create a block for every distinct region of a document, e.g., each paragraph, section header, footnote, figure, etc. The recursive part of the algorithm is similar to the standard XY-cut implementation, but we use three types of thresholds: (1) maximum valley noise; (2) minimum valley width; and (3) minimum non-valley size. Furthermore, all three types of thresholds are defined for both the x and y axes, resulting in six threshold values.

For our task, we set the first threshold to 0 for both axes, as there is little noise to account for in text extracted from generated PDFs. We explore two ways

[1] `https://www.pdflib.com/tet-cookbook/tetml-and-xslt/tetml/`

[2] `https://poppler.freedesktop.org/`

[3] "Freki" is one the wolves that accompanies the Norse god Odin, chosen as a reference to the ODIN project.

```
<Text>PROG-wash</Text>
  <Box llx="238.92" lly="587.60" urx="289.09" ury="597.56">
    <Glyph font="F47" size="8.04" x="238.92" y="587.60" width="5.36" fill="C0">P</Glyph>
    <Glyph font="F47" size="8.04" x="244.32" y="587.60" width="5.80" fill="C0">R</Glyph>
    <Glyph font="F47" size="8.04" x="250.08" y="587.60" width="6.25" fill="C0">O</Glyph>
    <Glyph font="F47" size="8.04" x="256.32" y="587.60" width="6.25" fill="C0">G</Glyph>
    <Glyph font="F47" size="9.96" x="262.56" y="587.60" width="3.31" fill="C0">-</Glyph>
    <Glyph font="F47" size="9.96" x="265.92" y="587.60" width="7.19" fill="C0">w</Glyph>
    ...
  </Box>
```

Figure 3: The TETML output produced by TET for a portion of the token `PROG-wash-PST-2SG` in Fig 1.

```
(24) Kurmanji: passive construction (present)
a. ez te di--m
1SG 2SG.ACC PROG-wash.PRS-1SG
I am washing you.
b. tu t-ł(-y) t-in
2SG PROG-come-2SG wash-INF
You are being washed (by me).
```

Figure 4: The TET plain-text output for the IGT instance in Figure 1.

of choosing the second type of threshold: the first simply uses the average character height in the page for both the x and y axes; the second method analyzes a histogram of valley sizes to select one greater than the median size, which is intended to cut section boundaries but not individual lines. Our experiments show that the first method worked better in general, although the second does better for documents with double-spaced lines. The third type of threshold prevents a cut from resulting in too small a block. We limited vertical cuts so the resulting blocks must be at least 1/6 of the original page width and 1/32 of the original page height, and horizontal cuts so the resulting blocks must be at least 1/6 of the original page width and 1/128 of the original page height.

The second stage of block-detection is line detection. Each block is reanalyzed individually with no minimum valley width for horizontal cuts, which allows each line to be cut. Some PDFs have lines whose bounding boxes are completely abutted, resulting in zero valleys between them. In response, when Freki constructs the 2D array of bounding boxes, it shrinks the boxes vertically by removing 1/5 of its height from the top and bottom.

3.2. Other changes to the XY-cut algorithm

XY-cut, and layout analysis algorithms in general, are often described for analyzing scanned documents. In contrast, we are analyzing the layout of PDFs generated by a typesetting system or word processor, and we ignore non-text elements, so we are not affected by the noise and scanning artifacts that can hinder layout analysis algorithms. Instead, we transform the bounding boxes of encoded glyphs into a 2D array for image analysis. This approach shares some similarities with (Ha et al., 1995), although that work computed bounding boxes for characters from scanned documents, not from PDFs.

XY-cut is a recursive algorithm, so a naïve traversal of the tree of cuts can give an order through the blocks that may correspond well with the intended reading order[4], but block ordering has also been a separate object of research (Meunier, 2005; Sutheebanjard and Premchaiswadi, 2010). We only use the implicit order from the tree traversal, but we record the path to each block, as well as source block coordinates, so that later analysis on block order can be performed. There are also known complex layouts that are not handled by the XY-cut algorithm, but as our focus is on academic articles that do not commonly use such layouts, we choose to ignore these cases for now.

Other work in layout analysis focused on accurately detecting and extracting tabular data (Perez-Arriaga et al., 2016; Oro and Ruffolo, 2009). We also aim to recover data presented tabularly, but rather than being able to accurately predict rows and cells for a table, we aim to maintain visual alignment in tabularly arranged text when converted to a monospaced font, as explained below.

3.3. Respacing and Tabular Data Analysis

Our use case is primarily to enable the automatic processing of text extracted from PDFs, with emphasis on text presented tabularly, such as IGT. Therefore, it is important to maintain the visual alignment of text elements when projecting coordinate-positioned text in a variable-width font to columns in a monospaced plain-text file. If we projected a token's x coordinate to a column by multiplying the coordinate by, e.g., the average character width, words with many narrow characters would occupy proportionally more projected space than words with many wide characters. The effect of this mismatch is that some words would get overwritten by the following words. We could instead multiply the coordinate by the widest character width, or something larger, to prevent the overlap, but this would amplify misalignments across

[4]E.g., for languages that write from left-to-right and top-down, traversing the left/top side of each cut before the the right/bottom cuts is often adequate.

```
doc_id=3667 page=226 block_id=226-6 bbox=... label=tbbtt 6836 6842
line=6836 fonts=F47-10.0,F48-10.0 bbox=...              : (24) Kurmanji: passive construction (present)
line=6837 fonts=F47-10.0,F49-10.0 tabscore=0.25 bbox=...:      a. ez      te      di-şû-m
line=6838 fonts=F47-10.0,F47-8.0 tabscore=1.00 bbox=... :         1SG  2SG.ACC    PROG-wash.PRS-1SG
line=6839 fonts=F47-10.0 bbox=...                       :         'I am washing you.'
```

Figure 5: The Freki output (see Section 4.) for the first IGT instance from Figure 1. The bounding box (`bbox`) information has been truncated to save space.

lines when the original tokens did not have exactly the same x coordinate.[5] We could project to a column, then adjust it to the next available column in cases of overlap, but then tables quickly become misaligned. Instead, we compute the intended column position and compare it to other lines within a block. The proportion of tokens sharing a projected column[6] is a metric we call **tab-score**. Successive lines meeting a threshold for tab-score (which we set to 60%) are marked as belonging to a tabular group. We then apply the project-and-adjust strategy described above, but all tokens within a tabular group sharing a projected column are adjusted to the next available column across all lines in the group. It's possible for non-tabular (i.e., prose) lines to surpass the tab-score threshold, but because we are not explicitly marking the groups as tables and merely adjusting the spacing, the only consequence is extra whitespace between words.

4. Output Format

While XML files produced by TET and PDFMiner offer a great deal of information, as shown in Fig 3, they are both verbose and far less human readable. While disk space may be cheap, reviewability of the output by a human domain expert is necessary to ensure that the resulting content is sensible and useful for consumption by downstream processes.

Compromising between exposing the rich positional information provided by these XML output formats and maintaining human readability, our output format preserves block information and the lines that comprise them, with both block-level and line-level metadata before the text content. Figure 5 shows an example block from the document in which Figure 1 was extracted. The Freki output is more informative than the plain-text output of TET in Fig 4 and more human readable than the XMT output of TET in Fig 3. The information contained in the block and line preambles is explained below.

[5]We've observed a number of documents with ad-hoc tables created, presumably, by the author inserting spaces until the content is visually in the intended spot, but the coordinates will not be exactly the same across rows.

[6]Sometimes things like example numbers inflate the number of unaligned tokens, so we begin counting from the rightmost first token between the two lines.

4.1. Block-Level Information

The first line of the block contains primarily identifying and positional information, including `doc_id` (Source document identifier), `page` (Page number), `block_id` (Unique block identifier), `bbox` (Bounding box of entire block), and `label` (XY-Cut tree path to this block). While the source document and block identifier information is largely for reference, the bounding box, XY-Cut path, and page number are all interesting positional features for downstream processing tasks.

4.2. Line-Level Information

Freki breaks each block into lines of text, which serve as the atomic unit of the output format. Each line is provided with a preamble that contains `line` (line number), `fonts` (list of *font-size* pairs found in the line), `bbox` (bounding box for the line), and `tabscore` (tab-score, see Section 3.3.).

While the line number is not much more than an identifier, the other data exposed are valuable features for the downstream task of identifying the semi-structured, semi-tabular data. For instance, from the bounding box information, one might create a feature for whether a line is offset significantly from the surrounding text, as is common with examples such as the one in Figure 1. The tab-score described in Section 3.3. may help in identifying the word-by-word alignment shown in the first two lines of each example of the IGT instances in Figure 1, as they line up neatly in columns. Finally, the line content following the preamble is additionally respaced to be natural for human readability.

5. Conclusion

Extracting semi-structured text from PDF files is not trivial. While there are existing open-source and commercial PDF-to-text converters, the output is either verbose and not human-readable or lacking important information (such as indention) that can be useful for downstream processes. Our software builds on top of existing converters and focuses on identifying blocks and respacing, useful for targeting IGT. The package, including a web interface, is freely available to the public at `github.com/xigt/freki`.

6. Bibliographical References

Bienz, T., Cohn, R., and Meehan, J. R., (1997). *Portable Document Format Reference Manual*, August. http://www.adobe.com/content/dam/Adobe/en/devnet/acrobat/pdfs/pdf_reference_1-7.pdf.

Georgi, R., Lewis, W. D., and Xia, F. (2014). Capturing divergence in dependency trees to improve syntactic projection. *Language Resources and Evaluation*, 48(4):709–739, October. http://doi.org/10.1007/s10579-014-9273-4.

Georgi, R., Xia, F., and Lewis, W. D. (2015). Enriching interlinear text using automatically constructed annotators. In *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH-2015), in conjunction with ACL 2015*, Beijing, China, July. http://www.aclweb.org/anthology/W15-3709.

Ha, J., Haralick, R. M., and Phillips, I. T. (1995). Recursive xy cut using bounding boxes of connected components. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 2, pages 952–955. IEEE.

Lewis, W. D. and Xia, F. (2008). Automatically identifying computationally relevant typological features. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, Hyderabad, India, January. http://www.aclweb.org/anthology/I/I08/I08-2093.pdf.

Lewis, W. D. and Xia, F. (2010). Developing ODIN: A multilingual repository of annotated language data for hundreds of the world's languages. *Literary and Linguistic Computing*, 25(3):303–319.

Lewis, W. D., Xia, F., Georgi, R., and Goodman, M. (2015). ODIN corpus download. xigt.org/odin.

Meunier, J.-L. (2005). Optimized XY-cut for determining a page reading order. In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pages 347–351 Vol. 1. IEEE, September.

Nagy, G. and Seth, S. (1984). Hierarchical representation of optically scanned documents. In *Proceedings of the 7th International Conference on Pattern Recognition (ICPR)*, pages 347–349, January.

Nagy, G., Seth, S., and Viswanathan, M. (1992). A Prototype Document Image Analysis System for Technical Journals . *Computer*, 25(7):10–22, July.

Oro, E. and Ruffolo, M. (2009). PDF-TREX: An approach for recognizing and extracting tables from PDF documents. In *ICDAR*.

PDFLib GmbH. (2015). PDFLib TET 5 – text and image extraction toolkit. pdflib.com/products/tet/.

Perez-Arriaga, M. O., Estrada, T., and Abad-Mota, S. (2016). TAO: System for table detection and extraction from PDF documents. In *FLAIRS Conference*.

Shafait, F., Keysers, D., and Breuel, T. M. (2006). Performance comparison of six algorithms for page segmentation. In *Document Analysis Systems*.

Shinyama, Y. (2016). pdfminer, December. github.com/euske/pdfminer.

Sutheebanjard, P. and Premchaiswadi, W. (2010). A modified recursive X-Y cut algorithm for solving block ordering problems. *2010 2nd International Conference on Computer Engineering and Technology*, pages V3–307 – V3–311, May.

van de Visser, M. (2006). *The Marked Status of Ergativity*. Landelijke Onderzoekschool Taalwetenschap. http://www.lotpublications.nl/Documents/141_fulltext.pdf.

Warnock, J. (1991). The Camelot Project . blogs.adobe.com/acrobat/files/2013/09/Camelot.pdf.

Crowdsourced Multimodal Corpora Collection Tool

Patrik Jonell, Catharine Oertel, Dimosthenis Kontogiorgos, Jonas Beskow, Joakim Gustafson

KTH Royal Institute of Technology, Sweden

Lindstedsvägen 24, 114 28 Stockholm, Sweden

{pjjonell, catha, diko, beskow}@kth.se; jocke@speech.kth.se

Abstract

In recent years, more and more multimodal corpora have been created. To our knowledge there is no publicly available tool which allows for acquiring controlled multimodal data of people in a rapid and scalable fashion. We therefore are proposing (1) a novel tool which will enable researchers to rapidly gather large amounts of multimodal data spanning a wide demographic range, and (2) an example of how we used this tool for corpus collection of our "Attentive listener" multimodal corpus. The code is released under an Apache License 2.0 and available as an open-source repository, which can be found at `https://github.com/kth-social-robotics/multimodal-crowdsourcing-tool`. This tool will allow researchers to set-up their own multimodal data collection system quickly and create their own multimodal corpora. Finally, this paper provides a discussion about the advantages and disadvantages with a crowd-sourced data collection tool, especially in comparison to a lab recorded corpora.

Keywords: crowdsourcing, human-computer interaction, multimodal corpus

1. Introduction

In the last decade more and more efforts have been carried out to create multimodal corpora. Efforts have come from diverse fields such as psychology, linguistics and computer science. They shared the common goal of enabling researchers to gain a better understanding of how humans converse with one another.

The traditional approach has been to record corpora in a lab environment. This approach has several disadvantages, however. One such disadvantage is that due to the constraint of having to attend the recording on campus, most resulting corpora are quite homogeneous with regards to participants' demographics (e.g. education level, socio-economic background, native language). Participants might also exhibit some degree of the Hawthorne effect, while being observed in sterile lab environments, leading to biased behaviour (Parsons, 1974). A further disadvantage is the monetary cost of staff, equipment and the time spent when doing in-lab recordings, for example setting up equipment.

Due to the constraints mentioned above more and more researchers have been turning to crowdsourcing as a means of gathering large quantities of data. In particular, crowdsourcing has been extensively used for annotations as well as transcriptions (Gruenstein et al., 2009; Hipp et al., 2013; Rashtchian et al., 2010). While it has, to some degree, been used for creation of conversational corpora as well, these corpora have mainly been restricted to speech and text (Filatova, 2012; Orkin and Roy, 2009; Breazeal et al., 2013). One of the reasons for the lack of these kinds of corpora might be that there are few tools which facilitate the acquisition of such data. While there are tools for collecting conversational data such as Voxforge[1], there are currently, to our knowledge, no publicly available open-source tools that allow for crowdsourcing multimodal corpora while simultaneously enabling researchers to control for experimental factors. We therefore propose and release an open-source tool that facilitates large-scale collection of multimodal corpora. Figure 1 depicts how the tool is used to collect a multimodal corpus using crowdworkers from different crowdsourcing platforms. In this paper, we will first provide an overview of the capabilities the tool provides and will then provide a summary of an example corpus which was collected using this data collection tool. Finally, we discuss the advantages and disadvantages when using a crowd-sourced approach and the limitations of the system.

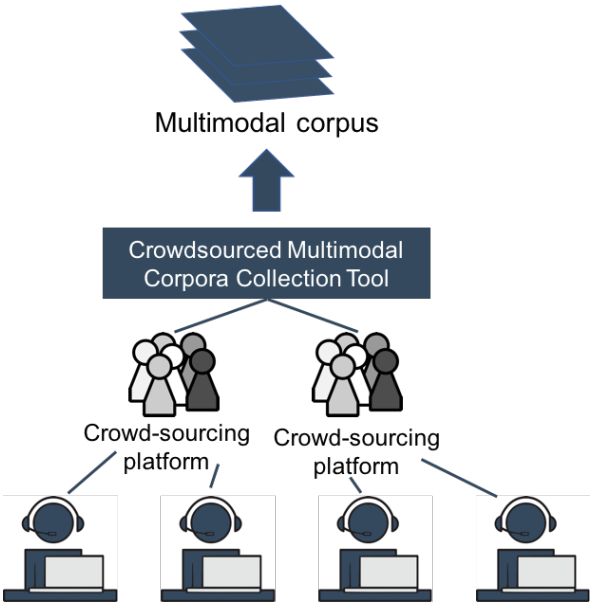


Figure 1: Corpus creation using multiple crowdsourcing services where participants record themselves.

[1] `http://www.voxforge.org/`

2. Background

2.1. Lab Recorded Multimodal Corpora

More and more corpora have been collected using a multitude of sensors. Both multi-party as well as dyadic corpora have been collected. Examples of dyadic corpora are: (van Son et al., 2008; Edlund et al., 2010) and examples of multi-party corpora are: (Carletta, 2007; Mostefa et al., 2007; Hung and Chittaranjan, 2010; Oertel et al., 2014).

An example of a corpus which tries to escape the lab setting is the D64 corpus (Oertel et al., 2013). It spans approximately two days of interactions. Four to five participants met in an apartment in Dublin which was equipped with a variety of sensors. They discussed any topic which sprang to mind during the two days.

In addition to human-human interaction corpora, there are also corpora which focus on human-robot interaction. An example of such a corpus is, for instance, the Tutorbot corpus (Koutsombogera et al., 2014). The aim of this corpus was to gather data, in order to teach a robot how to take on the role of a tutor in a multi-party tutoring scenario. In a similar fashion the data described in (Vollmer et al., 2014) was also gathered in order to understand tutoring scenarios. In this instance however, the emphasis was on understanding how a robot could communicate to a human how it wants to be taught. This approach is in some ways similar to the approach described in the "Attentive listener" scenario where we also provide a situational context in order for the robot to optimally learn certain skills.

Similarly to some of the recordings in (Carletta, 2007), we chose for the "Attentive listener" example corpus (Oertel et al., 2017) to define a scenario and let the participant play different roles when interacting with a fictive job applicant. As opposed to most previous corpus collection, e.g. (van Son et al., 2008; Oertel et al., 2013; Oertel et al., 2014), where long interactions with few participants were recorded, we gathered short interactions with a large amount of participants enabling to better capture the variety within human behaviour.

2.2. Crowdsourced Corpora

Crowdsourcing has been used in various ways for corpora creation. It has, for example, previously been used in the domain of automatic generation of narratives (Li et al., 2013; Leite et al., 2016). Leite et al. used crowdsourcing as a dialogue creation tool for (repeated) human-robot interaction. This was done in order to increase variation in a robot's dialogue responses.

There has been work on collection of text-based corpora, for example Filatova investigated irony and sarcasm by creating a corpus based on Amazon[2] reviews (Filatova, 2012). Another example is "sketchy" [3] where the authors had crowd workers to produce sketches using their computer.

Examples of speech corpora which have been created using crowdsourcing are (Lane et al., 2010), where the authors created a corpus for the purpose of improving speech recognition or (Gruenstein et al., 2009) where a speech corpus of orthographical transcriptions was created through the means of an online educational game.

Crowdsourcing for dialogue generation has been pioneered by (Orkin and Roy, 2009). Similarly, (Breazeal et al., 2013) proposed a data-driven approach to dialogue generation for a social robot by crowdsourcing dialogue and action data from an online multi-player game. These studies, however, have taken place in virtual environments and primarily focus on immediate interactions.

Like Filatova, we focused on the modelling of a paralinguistic phenomenon but in contrast to their work we focused on the modelling of scepticism vs. support instead of irony and sarcasm (Filatova, 2012). Moreover, our corpus includes additional modalities such as speech and video. While there have been other speech corpora created such as (Lane et al., 2010), or (Gruenstein et al., 2009), we are not aware of any other crowdsourced corpora which include both audio and video modalities.

2.2.1. Quality Control

While the implementation of quality control mechanisms varies from study to study, the following section provides an overview of approaches which have been used in order to ensure sufficiently high quality.

One possibility is the use of a second batch of crowdworkers to rate the work of the initial batch of crowdworkers. Mechanisms such as majority voting as well as inter-rater reliability can be used in order to assess which data to keep and which data to discard (Filatova, 2012).

Leite et al. also used a second batch of crowdworkers to rate the first batch of crowdworkers' performance. In addition, they included a test-item. If crowdworkers failed to correctly tag their data, it would be excluded and a new crowdworker would be recruited instead. They also set thresholds for the amount of agreement they expect crowdworkers to have when labelling a specific item (Leite et al., 2016).

It might, due to privacy concerns, not be possible to let a second batch of crowdworkers rate videos of the first batch, as has been done with text (Filatova, 2012; Leite et al., 2016). It might however be possible to map the participants' faces to a virtual agent, similar to (Edlund and Beskow, 2009; Jonell et al., 2017) and then let the crowdworkers rate the other participants indirectly using majority voting.

3. The tool

The tool was designed for rapid collection of rich multimodal data. It also provides researchers with the capabilities to control for experimental factors. The tool is a web-based application which utilises modern web technologies in order to access the participant's webcamera and microphone. No special technical skills are required from the participants as no installation on the participant's computer is required. As can be seen in Figure 2, the participants go through several steps during the session. If a participant does not successfully complete a step, the session ends and the data recording is dismissed. The following sections describe the key components of the tool in more detail.

[2] http://www.amazon.com/

[3] http://sketchy.eye.gatech.edu/

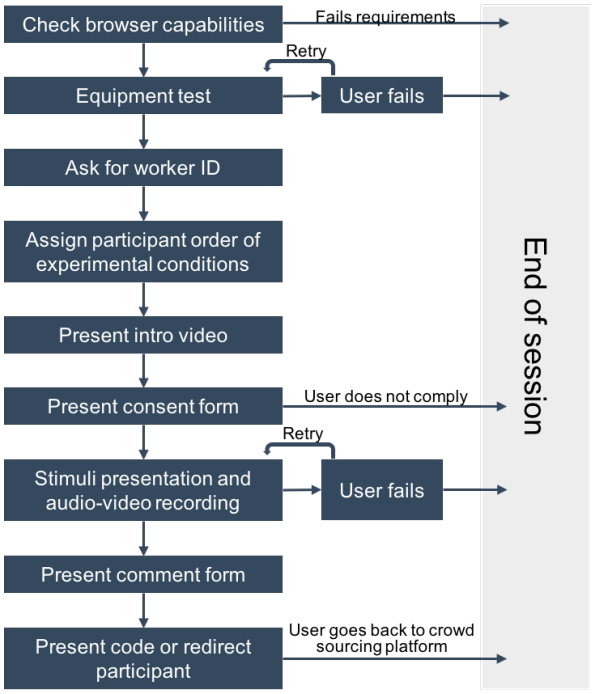


Figure 2: Flow chart depicting the user flow for a participant.

3.1. Data and Synchronisation

The data being collected using the tool is synchronised audio and video data. It is retrieved from the participants' microphone and webcamera while they are being exposed to stimuli. The participants are presented with stimuli which are streamed over the internet in the form of video or audio clips presented on HTML pages. The recorded data is also synchronised with the playback of the stimuli material so that it is possible to analyse the collected data in relation to the stimuli. The data is stored using common file formats for video files such as mp4 or webm. The tool does not handle retrieval of the data, as this can easily be done through other tools, depending on which storage solution was implemented (see section 4.2.). For analysis of the data it is essential that synchronisation between the crowdworker's video stream and the stimulis's video-stream is guaranteed. We used insertion of black frames into the video at given moments in order to synchronize the stimuli and the recordings. If for technical reasons the crowdworker experiences a lag in the video streaming, the same method is applied. Black frames are inserted up until the point in time of when the video is started again.

3.2. Assignment of experimental condition

To guarantee that an equal amount of videos are recorded for each experimental condition, an automatic balancer has been implemented. This balancer assigns a trial order to each new participant at the start of the session. If a participant timed out, i.e. took longer than 30 minutes, their data was removed so that another user could be assigned that experiment condition order.

3.3. Quality control

In order to ensure that the resulting corpus is of high recording quality and subsequent multimodal analysis is possible, an initial quality control test is performed to ensure that the following prerequisites are fulfilled.

First, the user needs to be using a modern web browser in order for the tool to work. If the tool detects that the user is using an outdated browser, they are notified about this and the session is ended. Second, it is important for the prosodic analysis to ensure that data is recorded in a quiet environment. To ensure this, participants are asked to speak at predefined moments and to be quiet at others. The noise-to-speech ratio is then calculated. If participants do not meet the requirements, i.e. the noise-to-speech ratio is too high, the users are informed about the problem and given the possibility to correct their set-up and redo the test. In order to proceed they have to pass the quality control test.

Finally, it is important that no cross-talk occurs. By cross-talk we refer to audio from the stimuli being picked up by the microphone of the participant. In order to ensure that this does not happen, participants are asked to use a headset with a close-talking microphone. A similar procedure as described above for detecting a suitable recording environment is used to ensure that no cross-talk occurs.

In order to achieve an automatic quality control for speech recordings after each recording, we implemented the following simple but efficient procedure; (1) Record a test recording where a participant responds to each stimuli and (2) aggregate the duration of the speech. Then (3) set a generous lower and upper duration limit for each stimuli. If participants are too far outside the time-span, the participant is notified about it and given the option to repeat the recording.

4. Technical overview

For detailed instructions on how to set up and configure the tool, refer to the projects README file which can be found in the git repository. In order to set up a data collection, stimuli material must be prepared and HTML pages created. The tool does not provide any user interface to set this up, but there are many tools available to create such HTML pages. There are some examples of HTML pages used for the "Attentive listener" scenario in the code repository.

4.1. Web Application

The web application's backend was written in Python, and was responsible for serving the HTML code to the participants. It was also designed to handle video uploads, data storage and database operations.

Its frontend was written with modern standardized web technologies. Using HTML5 the tool is able to access the participant's peripherals such as their webcamera or microphone. These features are enabled in most modern web browsers.

4.2. Storage

As the file sizes of video material can become large, the application has the capability to use an external storage service. In the example of the "Attentive listener" corpus,

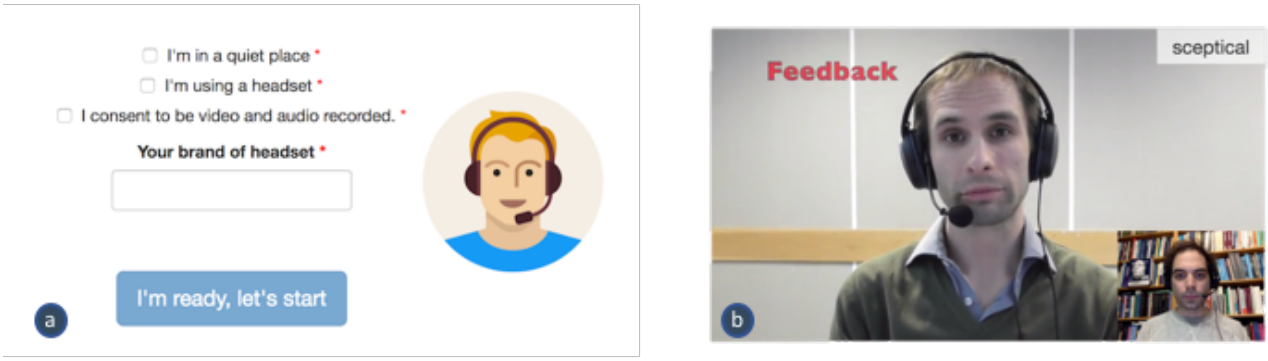


Figure 3: Interface elements from the tool; a) consent form that the user is presented with and b) stimuli presentation and participant recording from the "Attentive listener" scenario.

Amazon S3 was used, as it provides a scalable storage solution for large sized data. However, the tool can be integrated with any storage service or data can directly be stored on the local file system on the server. The files can be stored as either webm or mp4 files. The tool only facilitates saving the data at the given storage location, but does not retrieve it, as this can easily be done using other tools depending on the chosen storage solution.

4.3. Database

For the database a PostgreSQL server was used. The database keeps records of each participant's progress and meta data concerning their trials. The database stores the participants crowdsourcing ID, which is the ID given from the crowdsourcing platform. This ID is used in order to link the participant back to the crowdsourcing platform for reimbursement purposes, but also to ensure avoiding duplicate participant recordings. Each time the participant viewed or uploaded a new video to the server, the database was updated accordingly. The important information that the database stores about the user is the participant's ID from the crowdsourcing platform, the order of the trial conditions, participant's final comment, current progress of the participant and finally subjective quality measure of each participant's videos.

5. Attentive listener use-case scenario

In this section we would like to briefly discuss a corpus creation using the data collection tool described in this paper. Feedback generation is an important component of human-human communication. Humans can choose to signal support, understanding, agreement or also scepticism by means of feedback tokens. In order to also make human-agent and human-robot robot interactions smoother, many projects and studies have focused on modelling feedback generation, e.g. (Douglas-Cowie et al., 2008; Schroder et al., 2012; Park et al., 2017). Many studies in particular have focused on the timing of feedback behaviours (Morency et al., 2010; Ward and Tsukahara, 2000; Cathcart et al., 2003). To our knowledge, very few studies have kept the timing constant (Oertel et al., 2016a; Oertel et al., 2016b) and instead focused on the variation of lexical form and prosody of feedback tokens within identical contexts.

For our "Attentive listener" corpus, crowdsourced participant's feedback behaviour is captured in identical interactional contexts in order to model a virtual agent that is able to provide feedback as an attentive/supportive as well as attentive/sceptical listener. We chose a scenario where a confederate was asked to pretend to apply for several jobs. We instructed the crowdworkers to interview our confederate by means of a video conference call Figure 3b. We asked them specifically to give short feedback utterances at predefined moments. These moments were indicated through a countdown visualised in the right hand side of the video. The resulting models were realised in an artificial agent which was evaluated by third-party observers.

We recorded 92 participants, with 3 recordings per participants, given the conditions supportive, sceptical and neutral in a random order. This resulted in a total of 276 recordings. The recordings took place during 3 consecutive days and we chose to only recruit participants with English as their native language living in the United States or Canada. This corpus is not publicly available but is described in more detail in (Oertel et al., 2017).

6. Discussion

As previously mentioned the proposed tool can help to collect large data amounts in a short amount of time. This can be used in many applications such as for example the above mentioned corpus collection. It was also successfully used to automatically generate virtual agents (Jonell et al., 2017) from the recorded data. Further it can be used to analyse human behaviour or to automatically learn human behaviour in given situations by applying machine learning algorithms on the data.

In addition to obvious factors of time and scalability, capturing high degrees of variation in behaviour was also an important factor when deciding to design the audio-visual crowd-sourced data collection tool.

Because crowdsourcing platforms were used in order to recruit participants, it's possible to reach a wide variety of participants. This enables taking a wider range of people into account when building new multimodal systems. It is also possible to recruit participant's with very specific requirements, e.g., a certain language as native language.

6.1. Quality of task performance

In addition to the quality of the signal the quality of task performance is also essential for determining the overall quality of the audio-visual corpus. For instance, did the crowd-workers complete the task as intended or was there confusion? Were there instances where the crowd-workers lacked motivation for completing the task? We encountered instance where some crowd-workers understood providing feedback not as saying "yeah", "okay"- as illustrated in the instructions but as providing more of a general kind of feedback directed towards the applicant such as: “You should be more confident about yourself”. In other cases the crowd-worker remained still during the whole duration of the task and did not provide any feedback. These examples were discarded from the general corpus.

6.2. Limitations

Despite many advantages of using a crowdsourced multimodal corpora creation over an in-lab data collection there are also limitations and shortcomings. One limitation is that the approach is restricted to dyadic corpus creation. This implies that the study of multi-party conversational phenomena is not possible within the current setup. However, while we are currently using crowdsourcing platforms that are mostly used by individual workers, it would be possible to, for example, email known participants in pairs and ask them to sign in to the same session or to sit together in front of one computer.

Another disadvantage of our approach is that it does not provide the researcher with full control of the environment nor participants’ equipment. One way to handle differences in participants’ equipment is to let the participants list the equipment they are using and for example only use participants with the same or similar equipment. The downside with this approach is that it will heavily limit the potential participants. Furthermore, when recordings were bad due to faulty or low-quality equipment, we discarded those participants as it was both easy and cheap to recruit new participants instead.

Also the authenticity of our approach is debatable with regards to three points. First of all, the authenticity of the stimuli provided. If the stimuli which are presented to crowd-workers are based on acted scenarios, the reactions of crowd-workers might also not be as they would be in a completely spontaneous co-located interactions. However, in a completely spontaneous co-located interaction, participants might also be influenced by other events or actions going on at the same time. Therefore, the resulting actions might also be not optimal for learning. Second of all, the authenticity of participants reactions. We are asking participants to take on a stance and provide feedback in a specific way e.g. “supportive", “sceptical" etc. While it is true that these reactions might not be completely genuine, it is also true that as humans we often masking our true emotions and attitudes. However, it cannot be denied that some people might be better skilled in acting in such a scenario than other people. Yet, in the case of backchannels we found that people could generally convey the conversational function very well.

And finally, being co-located during an interaction or reacting to a video might influence crowd-workers behaviours. While being co-located of course makes a difference with regards to conversational behaviours, it is also true that we are more and more used to video-mediated interactions. Therefore, we believe that our approach while maybe not as ecological valid as other co-located, spontaneous conversation approaches provides a good comprise between, scalability, control and authenticity.

The last limitation discussed in this paper, is that it is not trivial to capture participants’ gaze direction. This is a similar limitation to any conference call. This is partly due to the software currently available for gaze estimation, but also for other unknown factors such as screen size, distance to screen and so on. This could, partly be helped by an initial gaze calibration process.

6.3. Ethical and legal concerns

As the participants are being recorded on both video and audio this gives rise to both ethical, privacy and legal concerns. Therefore, it is important to make the participants aware of the fact that their data is being recorded and how this data will be handled. The participants should actively agree to being recorded, and any local laws regarding data collection and storage has to be obeyed. This paper presents a tool for data collection, but it is the responsibility of the person using the tool to make sure both the ethical and legal concerns are being properly addressed.

7. Conclusions

In this paper, we have proposed a novel tool that allows for fast, scalable, demographically varied and quality controlled multimodal corpora creation. The tool is available for download as an open source project at: `https://github.com/kth-social-robotics/multimodal-crowdsourcing-tool` under Apache License 2.0. We also gave an example of how we successfully employed it to collect our “Attentive listener” corpus. We are hoping that by releasing this tool, users who find it useful will contribute to the code base.

7.1. Future work

Improvements can still be made regarding the tool, such as in the area of quality control and video streaming features. While we tested the usefulness of the tool in the use-case scenario described in this paper, we did not carry out a formal usability study. The conduction of such a study will be part of future work. Additional modalities being captured in form of, for example, capturing keyboard and mouse activity or eye gaze in combination with the audio-visual data could also be highly interesting extensions.

8. Acknowledgements

The authors feel particularly thankful to Joseph Mendelson and Todd Shore for making the data recordings possible and Sarah Ramsay for reviewing the paper. The authors would also like to acknowledge the support from the Swedish Foundation for Strategic Research project EACare (RIT15-0107), the Swedish Research Council Project InkSynt (2013-4935), the EU Horizon 2020 project BabyRobot (687831) and the Swedish Foundation for Strategic Research project FACT (GMT14-0082).

9. References

Breazeal, C., DePalma, N., Orkin, J., Chernova, S., and Jung, M. (2013). Crowdsourcing human-robot interaction: New methods and system evaluation in a public environment. *Journal of Human-Robot Interaction*, 2(1):82–111.

Carletta, J. (2007). Unleashing the killer corpus: experiences in creating the multi-everything ami meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

Cathcart, N., Carletta, J., and Klein, E. (2003). A shallow model of backchannel continuers in spoken dialogue. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 51–58. Association for Computational Linguistics.

Douglas-Cowie, E., Cowie, R., Cox, C., Amier, N., and Heylen, D. (2008). The sensitive artificial listner: an induction technique for generating emotionally coloured conversation.

Edlund, J. and Beskow, J. (2009). Mushypeek: A framework for online investigation of audiovisual dialogue phenomena. *Language and Speech*, 52(2-3):351–367.

Edlund, J., Beskow, J., Elenius, K., Hellmer, K., Strömbergsson, S., and House, D. (2010). Spontal: A swedish spontaneous dialogue corpus of audio, video and motion capture. In *LREC*.

Filatova, E. (2012). Irony and sarcasm: Corpus generation and analysis using crowdsourcing. pages 392–398. Citeseer.

Gruenstein, A., McGraw, I., and Sutherland, A. M. (2009). A self-transcribing speech corpus: collecting continuous speech with an online educational game. pages 109–112.

Hipp, J. A., Adlakha, D., Gernes, R., Kargol, A., and Pless, R. (2013). Do you see what i see: Crowdsource annotation of captured scenes. In *Proceedings of the 4th International SenseCam & Pervasive Imaging Conference*, SenseCam '13, pages 24–25, New York, NY, USA. ACM.

Hung, H. and Chittaranjan, G. (2010). The idiap wolf corpus: exploring group behaviour in a competitive role-playing game. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 879–882. ACM.

Jonell, P., Oertel, C., Kontogiorgos, D., Beskow, J., and Gustafson, J. (2017). Crowd-powered design of virtual attentive listeners. In *International Conference on Intelligent Virtual Agents*, pages 188–191. Springer, Cham.

Koutsombogera, M., Al Moubayed, S., Bollepalli, B., Abdelaziz, A. H., Johansson, M., Lopes, J. D. A., Novikova, J., Oertel, C., Stefanov, K., and Varol, G. (2014). The tutorbot corpusâĂŢa corpus for studying tutoring behaviour in multiparty face-to-face spoken dialogue. In *LREC*, pages 4196–4201.

Lane, I., Waibel, A., Eck, M., and Rottmann, K. (2010). Tools for collecting speech corpora via mechanical-turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 184–187, Stroudsburg, PA, USA. Association for Computational Linguistics.

Leite, I., Pereira, A., Funkhouser, A., Li, B., and Lehman, J. F. (2016). Semi-situated learning of verbal and nonverbal content for repeated human-robot interaction. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 13–20. ACM.

Li, B., Lee-Urban, S., Johnston, G., and Riedl, M. O. (2013). Story generation with crowdsourced plot graphs. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, AAAI'13, pages 598–604. AAAI Press.

Morency, L.-P., de Kok, I., and Gratch, J. (2010). A probabilistic multimodal approach for predicting listener backchannels. *Autonomous Agents and Multi-Agent Systems*, 20(1):70–84.

Mostefa, D., Moreau, N., Choukri, K., Potamianos, G., Chu, S. M., Tyagi, A., Casas, J. R., Turmo, J., Cristoforetti, L., Tobia, F., et al. (2007). The chil audiovisual corpus for lecture and meeting analysis inside smart rooms. *Language Resources and Evaluation*, 41(3-4):389–407.

Oertel, C., Cummins, F., Edlund, J., Wagner, P., and Campbell, N. (2013). D64: A corpus of richly recorded conversational interaction. *Journal on Multimodal User Interfaces*, 7(1-2):19–28.

Oertel, C., Funes Mora, K. A., Sheikhi, S., Odobez, J.-M., and Gustafson, J. (2014). Who will get the grant?: A multimodal corpus for the analysis of conversational behaviours in group interviews. In *Proceedings of the 2014 Workshop on Understanding and Modeling Multiparty, Multimodal Interactions*, pages 27–32. ACM.

Oertel, C., Gustafson, J., and Black, A. W. (2016a). Towards building an attentive artificial listener: On the perception of attentiveness in feedback utterances.

Oertel, C., Lopes, J., Yu, Y., Mora, K. A. F., Gustafson, J., Black, A. W., and Odobez, J.-M. (2016b). Towards building an attentive artificial listener: on the perception of attentiveness in audio-visual feedback tokens. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 21–28. ACM.

Oertel, C., Jonell, P., Kontogiorgos, D., Mendelson, J., Beskow, J., and Gustafson, J. (2017). Crowdsourced design of artificial attentive listeners. In *INTERSPEECH: Situated Interaction, Augusti 20-24 Augusti, 2017*.

Orkin, J. and Roy, D. (2009). Automatic learning and generation of social behavior from collective human gameplay. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 385–392. International Foundation for Autonomous Agents and Multiagent Systems.

Park, H. W., Gelsomini, M., Lee, J. J., and Breazeal, C. (2017). Telling stories to robots: The effect of backchanneling on a childâĂŹs storytelling. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 100–108. ACM.

Parsons, H. M. (1974). What happened at hawthorne? *Science*, 183(4128):922–932.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier,

J. (2010). Collecting image annotations using amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 139–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Schroder, M., Bevacqua, E., Cowie, R., Eyben, F., Gunes, H., Heylen, D., Ter Maat, M., McKeown, G., Pammi, S., Pantic, M., et al. (2012). Building autonomous sensitive artificial listeners. *IEEE Transactions on Affective Computing*, 3(2):165–183.

van Son, R., Wesseling, W., Sanders, E., van den Heuvel, H., et al. (2008). The ifadv corpus: a free dialog video corpus. In *LREC*, pages 501–508.

Vollmer, A.-L., Mühlig, M., Steil, J. J., Pitsch, K., Fritsch, J., Rohlfing, K. J., and Wrede, B. (2014). Robots show us how to teach them: Feedback from robots shapes tutoring behavior during action learning. *PloS one*, 9(3):e91349.

Ward, N. and Tsukahara, W. (2000). Prosodic features which cue back-channel responses in english and japanese. *Journal of pragmatics*, 32(8):1177–1207.

Expert Evaluation of a Spoken Dialogue System in a Clinical Operating Room

Juliana Miehle[1], Nadine Gerstenlauer[1], Daniel Ostler[2], Hubertus Feußner[2], Wolfgang Minker[1], Stefan Ultes[3]

[1]Institute of Communications Engineering, Ulm University, Albert-Einstein-Allee 43, 89081 Ulm, Germany
[2]Minimally-invasive Interdisciplinary Therapeutical Intervention, Klinikum rechts der Isar, Technical University of Munich, Trogerstr. 26, 81675 München, Germany
[3]Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK
juliana.miehle@uni-ulm.de, nadine.gerstenlauer@uni-ulm.de, daniel.ostler@tum.de, hubertus.feussner@tum.de, wolfgang.minker@uni-ulm.de, su259@cam.ac.uk

Abstract

With the emergence of new technologies, the surgical working environment becomes increasingly complex and comprises many medical devices which have to be monitored and controlled. With the aim of improving productivity and reducing the workload for the operating staff, we have developed an Intelligent Digital Assistant for Clinical Operating Rooms (IDACO) which allows the surgeon to control the operating room using natural spoken language. As speech is the modality used by the surgeon to communicate with their staff, using it to control the technical devices does not pose an additional mental burden. Therefore, we claim that the surgical environment presents a potential field of application for Spoken Dialogue Systems. In this work, we present the design and implementation of IDACO as well as the evaluation in an experimental set-up by specialists in the field of minimally invasive surgery. Our expert evaluation yields promising results and allows to conclude that clinical operating rooms are indeed an expedient area of application for Spoken Dialogue Systems.

Keywords: Intelligent Assistance, Natural Spoken Language, Surgical Environment

1. Introduction

Finding new technological solutions in order to enhance the work in clinical operating rooms has been in the focus of research for many years. However, with the emergence of new technologies, the surgical working environment comprises nowadays many medical devices which have to be monitored and controlled and thus becomes increasingly complex. Therefore, new strategies are needed to keep the working environment manageable and to reduce the workload for the surgical team, thus allowing them to fully focus on the actual surgical procedure. In this context, the *Operating Room of the Future* is a keyword often used (Feußner, 2003). It describes the application of new technologies such as computer-enhanced systems to create an intelligent operating room (OR) that facilitates work and reduces the staff needed during a surgical intervention (Pransky, 2001). This reduces personnel cost and promises to lessen the rate of avoidable incidents caused by human error.

However, having computer systems that monitor the surgical devices is not sufficient. An intelligent OR also needs an intelligent human-computer interface as the surgeon as well as the whole surgery team must not be disturbed by the usage of complex computer applications and numerous devices. Therefore, the interface has to be designed so as to be simple and intuitive (Feußner et al., 2014). During a procedure, the surgeon needs his hands to operate on the patient and his eyes for being aware of what he is doing. Thus, any graphical and gesture-based systems are not well suited for this purpose. Being hands- and eyes-free, speech as an input and output modality seems to be a good choice. Moreover, it is the modality used by the surgeon to communicate with their staff. Thus, using speech to control the technical devices does not pose an additional mental burden. The surgeon can focus on the surgery and control the technical environment at the same time without taking care of how to interact with the system. Therefore, we claim that the surgical environment presents a potential field of application for Spoken Dialogue Systems.

Until now, the use of voice interaction systems in clinical operating rooms has only been scarcely researched. Most research projects focus on visualisation and intraoperative imaging, telesurgery and robotic surgery as well as educational features (Bharathan et al., 2013). Moreover, there exist some projects about workflow modelling (Sutherland and van den Heuvel, 2006; Padoy, 2010; Katić et al., 2014; Kranzfelder et al., 2011; Kranzfelder et al., 2013; Kranzfelder et al., 2014; Agarwal et al., 2007). To the best of our knowledge, there exists only one speech-enabled operation assistant. HERMES built by Computer Motion Inc. (Roe and Wang, 2000) connects all devices in the OR over a central network. Then, the scrub nurse has the possibility to control all instruments and devices over a central touch screen instead of numerous control panels. Moreover, HERMES provides the surgeon a speech control over several devices. Feedback to the surgeon's request is shown on the endoscopic video screen and the system provides audio feedback if one of the controlled devices loses power or gets disconnected from the network. However, even though the use of voice commands was well received among surgeons and OR staff, HERMES implements an audio-visual interface and pure speech interaction is not supported.

Overall, the existing applications mostly concentrate on workflow modelling or single smart features rather than building up a complete intelligent interface. In contrast, our aim is to develop and evaluate an intelligent spoken language operation assistant offering several functionalities in order to provide the surgeon assistance in many different situations before and during an ongoing procedure. The work described in this paper builds upon and extends work published in (Miehle et al., 2017b; Miehle et al., 2017a; Gerstenlauer, 2017).

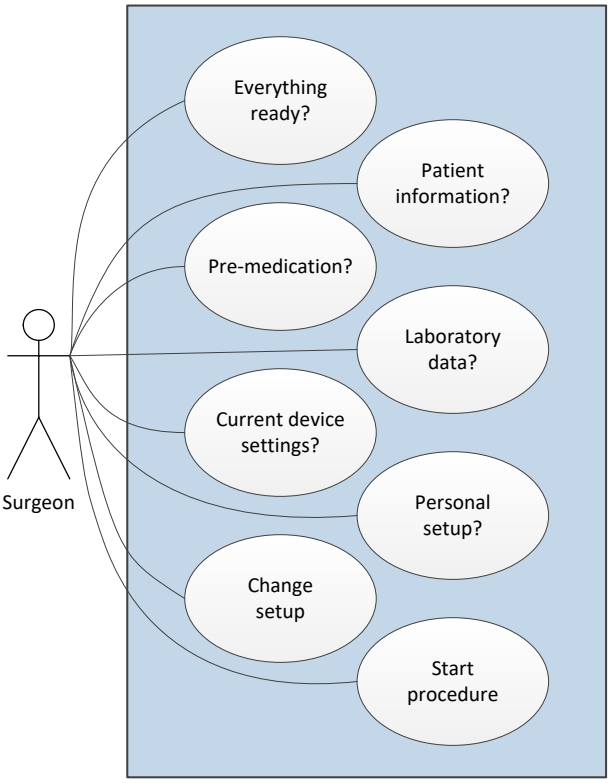


Figure 1: Before the operation starts, IDACO supports the surgeon during the team time-out and provides all necessary data.

The structure of the paper is as follows: In Section 2., the design and implementation of our system are described. The evaluation of IDACO is presented in Section 3., before concluding in Section 4.

2. Intelligent Digital Assistance for Clinical Operating Rooms

In order to increase productivity and reduce the workload for the operating staff, our system acts cooperatively and supports the surgeon autonomously during the surgery. IDACO escorts the surgery team throughout the entire procedure and provides assistance where necessary. The main functionalities of the presented speech-based assistant for clinical operating rooms have been identified during a requirement analysis with specialists in the field of minimally invasive surgery. They are described in the next section.

2.1. Functionalities

During the requirement analysis, we have identified two major parts where intelligent assistance is considered desirable by the medical specialists: the pre-procedural and the procedural part. For both parts, we determined several functionalities which should be taken on by IDACO.

Before the operation starts, the surgery team uses to carry out the team time-out in order to avoid any kind of confusion concerning the patient and the upcoming surgery. In the course of this procedure, the patient as well as the surgical team are identified and the surgery type is checked. IDACO supports the surgeon during the team time-out and provides data about the surgery type and the operating team as well as the patient, including pre-diseases, medical treatment and laboratory data. The variety of information the user can request from the system in this mixed-initiative dialogue part is illustrated in Figure 1.

Afterwards, the surgeon starts with the operation and IDACO escorts the team throughout the entire surgery. The system tracks the usage of surgical instruments and material (e.g. trocars, different types of clips, suturing material) by constantly listening to the operating surgeon and compares the thereby observed course of the procedure to the predicted surgical workflow. If the usage differs, it reacts proactively and utters a warning via speech. Moreover, IDACO controls surgical devices automatically at the right time of the operation when the surgeon confirms the execution. For example, the presented system is able to start the insufflator, increase the gas insufflation, turn off and on the light and tilt the table. In order to operate the devices autonomously, the system saves the preferred device settings for each surgeon and transmits the parameters to the surgical devices (e.g. OR table, room light, insufflator, suction and irrigation unit). Moreover, it allows the surgeon to retrieve and change their pre-settings.

For unseen incidents, IDACO encompasses an emergency mode which includes a ”silent mode” to prevent further distractions by the system. This mode may be activated by the surgeon at each arbitrary point of the operation. Having it activated, IDACO does not track the usage of surgery material until it is explicitly told to end this silent mode. Meanwhile, the surgeon can operate without being disturbed by the system. Furthermore, the system offers starting the emergency mode proactively in delicate situations.

2.2. Challenges

Enabling an intelligent operating assistance system to follow a surgery and control surgical devices automatically bears several challenges.

For keeping track of the procedure and automatically controlling surgical devices, the system needs to know when to perform which action on which device and when to stay in the background. Therefore, it has to be aware of the whole context of the surgery, i.e. the current point of the procedure and all past and future actions. This means that a reliable method for tracking the course of the surgery needs to be developed, thus allowing to detect unscheduled events. Moreover, it has to be clearly defined how the system is supposed to react in tenuous situations. For this purpose, standardized surgeries need to be described in detail, allowing the system to compare the actual course of the procedure to the schedule (Feußner and Wilhelm, 2016). Using this medical domain knowledge, exact models of the complex surgery structure need to be created which are then applied to the voice interaction system. Additionally, an interface needs to be designed and implemented which allows intercommunication between the voice interaction system and the surgical devices as well as the clinical information system. Moreover, with respect to patient safety, appropriate strategies need to be defined in order to maintain full control of the medical devices even if IDACO is allowed to perform some pre-defined actions during the surgery and control devices automatically.

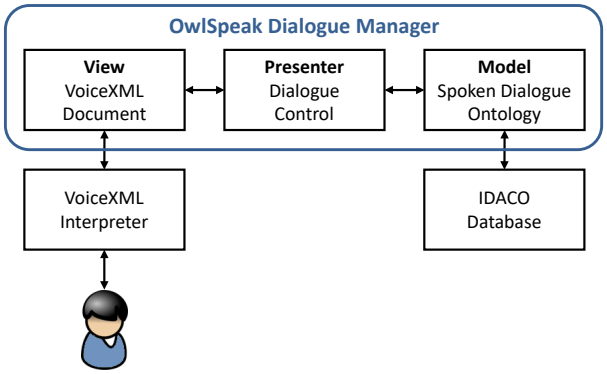


Figure 2: The overall architecture of IDACO comprising OwlSpeak, a VoiceXML interpreter and the IDACO Database.

2.3. Dialogue Modelling

For the dialogue modelling, we used the ontology-based Dialogue Management System OwlSpeak developed by Heinroth et al. (2010) and further extended by Ultes and Minker (2014). The overall architecture of our system can be seen in Figure 2. OwlSpeak is based on the model-view-presenter design pattern (Potel, 1996) which allows to separate data management (model), dialogue interface (view) and dialogue logic (presenter). The model is thereby implemented in the form of a Spoken Dialogue Ontology using the Web Ontology Language (OWL) (Antoniou and Van Harmelen, 2004) which contains the description of the dialogue and the current dialogue state. The view is represented by a VoiceXML document which is dynamically created at each dialogue turn and the presenter comprises the dialogue control logic. As OwlSpeak provides a new VoiceXML document at each turn, a VoiceXML interpreter by Voxeo[1] has been integrated. Moreover, OwlSpeak has been connected to the IDACO Database which acts as the interface between the Dialogue Manager and the Clinical Information System as well as the surgical devices. This database is described in detail in Section 2.4.

As a first prototype, we modelled the dialogue flow for a laparoscopic cholecystectomy. The dialogue is thereby divided into two parts: the pre-procedural and the procedural part. For the entire dialogue, the system utterances are concise and direct, meaning that the requested information is output very concretely and without any additional information that might be inappropriate.

During the pre-procedural part, IDACO supports the surgeon during the team time-out and provides data about the surgery type, the operating team and the patient, including pre-diseases, medical treatment and laboratory data. Moreover, the surgeon has the possibility to ask the system to adopt the devices in use to his personal pre-settings. He can also ask IDACO to read out the current device settings or his personal pre-settings stored in the IDACO Database and to change his personal set-up for a specific device. The user has the dialogue initiative and thus the full control of how to start the interaction. Hence, the surgeon can decide on which information should be provided and which device settings should be changed or adopted. If he does not want to request any information, he can even skip the information-providing pre-procedural part and inform IDACO to start with the procedure immediately.

When the surgeon confirms to start with the surgery, the second part of the dialogue begins. In contrast to the first part which is very flexible and allows the user to control the dialogue, the procedural part follows an exact surgery schedule which has been modelled in the Spoken Dialogue Ontology. Keeping track of the surgery is thereby done by tracking the tool usage as described by Padoy (2010). Therefore, we introduced variables for all kinds of instruments which are used and all assistance actions which are performed during this specific surgery. The system listens to each of the surgeon's instructions and increments the variables after each user utterance corresponding to its specific purpose. The grammars that describe the range the system is able to understand are thereby kept flexible in order to avoid the need for command-based user inputs. In contrast, the operating surgeon does not need to concentrate on how to interact with IDACO but can just talk to the surgical staff and the system reacts to keywords used by the surgeon. The workflow and hence the current part of the operation are then derived from the history of used tools at any point of the surgical intervention. The observed course of the procedure is compared to the surgery schedule which has been modelled in the Spoken Dialogue Ontology. In case of a deviation from the regular course, the system reacts proactively and utters a warning. The surgeon can then correct the amount of used material or tell the system that the expected usage has to be adapted for the rest of the procedure.

Moreover, the surgery schedule which has been modelled in the Spoken Dialogue Ontology defines at which point of the operation IDACO should perform a control action on medical devices such as starting the insufflator, increasing the gas insufflation, turning off and on the light and tilting the table. Whenever the system recognises such a time for action, it asks the surgeon whether the action should be performed. The surgeon can then confirm the execution or tell IDACO to wait until he explicitly utters to do so. This allows the surgeon to maintain full control of the medical devices, thus ensuring the patient safety. For the emergency mode, we introduced an Agenda[2] without any system move and only one possible user move which is the user giving the command to deactivate this mode.

The pre-operational part of the system can be reused for various kinds of surgical interventions. Only the variables storing the device settings and the personal device set-up need to be adjusted to the specific operation type what can be easily done using the IDACO Database. The actual procedural part uses an exact model of the complex surgery structure. Hence, only features that do not refer to any surgery specific data like the emergency mode can be reused for different surgical interventions but not the implemented course of the procedure.

[1] https://evolution.voxeo.com/

[2] Concept used by OwlSpeak to bundle several moves that belong to a specific dialogue turn (Ultes and Minker, 2014).

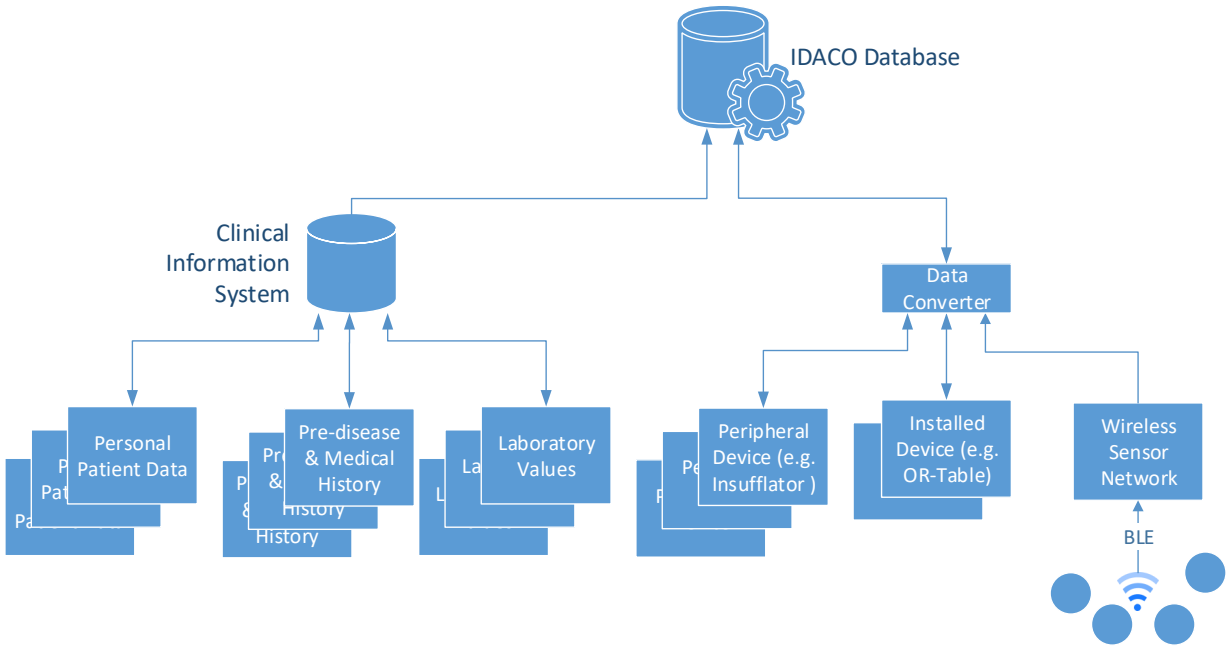


Figure 3: The schematic overview of the IDACO Database containing the mirrored patient- and intervention-specific information as well as the collected device and sensor data.

2.4. The IDACO Database

As an interface between the OwlSpeak Dialogue Manager and the Clinical Information System as well as the surgical devices, we have implemented a database which allows accessing necessary data and facilitates controlling of the surgical devices. A schematic overview of the IDACO Database is depicted in Figure 3.

The central point of the database is a table containing the intervention record from which all currently required data is concluded. Due to privacy issues, the system has no direct access to the Clinical Information System. However, the IDACO Database contains a mirrored image of all relevant information about the intervention. This covers patient data including personal data, pre-diseases, medical treatment and laboratory values, as well as data about the surgery type and the operating team. The table about the operating team contains all staff members who attend the current surgery, their name, their role and their experience. Moreover, as the system ought to store and adjust the surgeon's preferences, there is an additional database table containing the personal set-ups for each surgeon and each surgery type. One set-up contains all devices needed for the underlying operation as well as the corresponding parameters as pre-sets.

Furthermore, the IDACO Database provides an interface in order to control the surgical devices in the OR. Therefore, the database contains a list of all existing devices as well as the corresponding device parameters. After the confirmation by the surgeon, IDACO is able to set target values for the device parameters defined in the IDACO Database. Then, the data converter serves as interface to the specific device protocols of the peripheral an installed devices and changes the current parameters of the devices accordingly. Thus, IDACO is able to automatically control several devices at the right time of the procedure. Moreover, IDACO derives the information about which persons are currently present in the operating room from a Bluetooth-Low-Energy (BLE) tracking system.

3. Evaluation

The system was implemented in an experimental set-up in order to get an expert evaluation from medical specialists. As a first prototype, we have modelled the dialogue flow for a laparoscopic cholecystectomy. According to Cuschieri (1999), this is the gold standard for the treatment of gallstones and a highly standardized surgical procedure which can be segmented into ten procedural tasks:

1. Insertion of a Veress needle
2. Creation of Pneumoperitoneum
3. Insertion of Ports
4. Initial Inspection and Exposure of the Triangle of Calot
5. Dissection of Cystic Pedicle
6. Operative Cholangiogram
7. Closure of Cystic Duct and Detachment of Gallbladder from Liver
8. Extraction of Gallbladder
9. Final Inspection
10. Closure of Port Wounds

Each procedural part comprises several steps which are directly linked with the usage of certain material and instruments. As these instruments and materials, which are necessary to perform each procedural task, are clearly defined, it is possible to predict the surgeon's utterances during each step. For our prototype, we used the knowledge about these utterances in order to define a dialogue corresponding to the surgical workflow.
For example, the first procedural task of the laparoscopic cholecystectomy, the insertion of a Veress needle, comprises four steps:

1. Incision with a scalpel
2. Application of the first Backhaus clip
3. Application of the second Backhaus clip
4. Insertion of the Veress needle

Hence, IDACO needs to track "scalpel", "Backhaus clip", "Backhaus clip" and "Veress needle" in order to complete the first part of the surgery and to move on to the second procecural task which is the creation of the Pneumoperitoneum. In order to begin with this part, the gas insufflation needs to be started. Therefore, IDACO asks the surgeon whether this action should be performed. The surgeon can then confirm the execution or tell IDACO to wait until he explicitly utters to do so. The resulting dialogue excerpt looks as follows:

SURGEON: Scalpel, please.
SURGEON: Backhaus clip.
SURGEON: Please give me another Backhaus clip.
SURGEON: Now the Veress needle.
IDACO: Shall I start the gas insufflation?
SURGEON: Yes, please.
IDACO: The gas insufflation has been started.

The complexity of the dialogue increases with the complexity of the surgery structure. In total, our dialogue comprises 126 Agendas and 96 Variables which are used to control 18 installed and peripheral devices and to read out all necessary data from the Clinical Information System.
The implemented prototype was then evaluated by specialists in the field of minimally invasive surgery where it received good feedback. The speech interface and the dialogue were perceived very positively. IDACO is designed not to annoy the surgeon and the operating staff with inappropriate behaviour and unnecessary system prompts during the surgery. Therefore, the system utterances are concise and direct. This communication style has been assessed well-suited for the underlying dialogue scenario in an OR. Moreover, IDACO stays in the background if the procedure goes as scheduled. However, the evaluation with the experienced physicians indicated that passive system behaviour makes the surgery team insecure. The operating staff prefers to get any kind of feedback on what the system recognises and understands. The team of surgical specialists therefore suggested to equip the scrub nurse with a tablet PC showing each user utterance the system receives. In case of a misunderstanding by the voice interaction system, the nurse could then correct the recognised input. In doing so, the nurse has the possibility to observe the system behaviour and correct speech recognition errors manually.
Additionally, our evaluation pointed out that a close cooperation between system developers and medical scientists is inevitable in the design and implementation of intelligent systems for clinical operating rooms. Due to the highly specific field of application, it is hard for non-specialists to decide on optimal system behaviour.

4. Conclusion and Future Directions

We have presented the design, implementation and expert evaluation of a speech-based assistant for clinical operating rooms which supports the surgeon and their operating staff before and during a surgery. To the best of our knowledge, the presented system is the first intelligent spoken language operation assistant putting together several technologies in an OR and allowing to control the technical devices using speech. Enabling the operating surgeon to control devices inherent to the OR by himself as well as autonomously controlling some of the surgical devices reduces the workload for the surgical team as well as the amount of staff needed to assist during a surgical intervention and promises to lessen the rate of avoidable incidents caused by human error. Using IDACO, the surgeon can focus on the surgery and control the technical environment at the same time without taking care of how to interact with the system as speech is the modality used by the surgeon to communicate with their staff and IDACO just listens to the surgeon's utterances. Hence, it does not pose an additional mental burden on the surgical staff.
Our evaluation by specialists in the field of minimally invasive surgery showed that the system perceived very positive expert feedback. The only issue which remains open to debate is how the system should give feedback to the surgery team as, on the one hand, the surgeon should not be annoyed by unnecessary system prompts, but on the other hand, completely passive system behaviour makes the surgery team insecure. However, the speech interface, the dialogue and the communication style were assessed positively which leads us to the conclusion that we can confirm our claim that the surgical environment presents a field of application for Spoken Dialogue Systems.
In future work, the remaining issue of appropriate system feedback needs to be resolved. Afterwards, a broader evaluation needs to be done in order to get quantitative results regarding the performance. Moreover, a generic method for modelling the surgery control needs to be developed. For the presented system, we modelled exemplarily the procedure of a laparoscopic cholecystectomy. However, the multitude of existing surgical procedures makes it impossible to implement each one individually.

5. Bibliographical References

Agarwal, S., Joshi, A., Finin, T., Yesha, Y., and Ganous, T. (2007). A Pervasive Computing System for the Operating Room of the Future. *Mobile Networks and Applications*, 12(2-3):215–228.

Antoniou, G. and Van Harmelen, F. (2004). Web ontology language: Owl. In *Handbook on ontologies*, pages 67–92. Springer.

Bharathan, R., Aggarwal, R., and Darzi, A. (2013). Operating room of the future. *Best practice & research. Clinical obstetrics & gynaecology*, 27(3):311–322.

Cuschieri, A. (1999). Laparoscopic cholecystectomy. *Journal of the Royal College of Surgeons of Edinburgh*, 44(3):187–192.

Feußner, H. and Wilhelm, D. (2016). Minimalinvasive chirurgie und "robotic surgery": Chirurgie 4.0? *Der Chirurg*, 87(3):189–194.

Feußner, H., Reiser, S. B., Bauer, M., Kranzfelder, M., Schirren, R., Kleeff, J., and Wilhelm, D. (2014). Technische und digitale weiterentwicklung in der laparoskopischen/offenen chirurgie. *Der Chirurg*, 85(3):178–185.

Feußner, H. (2003). The operating room of the future: a view from europe. In *Seminars in Laparoscopic surgery*, volume 10, pages 149–156. Sage Publications Sage CA: Thousand Oaks, CA.

Gerstenlauer, N. (2017). Controlling an intelligent operating room with spoken language.

Heinroth, T., Denich, D., and Schmitt, A. (2010). OwlSpeak - Adaptive Spoken Dialogue within Intelligent Environments. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*, pages 666–671. IEEE.

Katić, D., Wekerle, A.-L., Gärtner, F., Kenngott, H., Müller-Stich, B. P., Dillmann, R., and Speidel, S. (2014). Knowledge-Driven Formalization of Laparoscopic Surgeries for Rule-Based Intraoperative Context-Aware Assistance. In David Hutchison et al., editors, *Information Processing in Computer-Assisted Interventions*, volume 8498 of *Lecture Notes in Computer Science*, pages 158–167, Cham. Springer International Publishing.

Kranzfelder, M., Schneider, A., Gillen, S., and Feußner, H. (2011). New technologies for information retrieval to achieve situational awareness and higher patient safety in the surgical operating room: the MRI institutional approach and review of the literature. *Surgical endoscopy*, 25(3):696–705.

Kranzfelder, M., Schneider, A., Fiolka, A., Schwan, E., Gillen, S., Wilhelm, D., Schirren, R., Reiser, S., Jensen, B., and Feußner, H. (2013). Real-time instrument detection in minimally invasive surgery using radiofrequency identification technology. *The Journal of surgical research*, 185(2):704–710.

Kranzfelder, M., Schneider, A., Fiolka, A., Koller, S., Reiser, S., Vogel, T., Wilhelm, D., and Feußner, H. (2014). Reliability of sensor-based real-time workflow recognition in laparoscopic cholecystectomy. *International journal of computer assisted radiology and surgery*, 9(6):941–948.

Miehle, J., Gerstenlauer, N., Ostler, D., Feußner, H., Minker, W., and Ultes, S. (2017a). An intelligent digital assistant for clinical operating rooms. In *Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue (SEMDIAL)*, pages 164–165, August.

Miehle, J., Ostler, D., Gerstenlauer, N., and Minker, W. (2017b). The next step: intelligent digital assistance for clinical operating rooms. *Innovative Surgical Sciences*, 2(3):159–161.

Padoy, N. (2010). *Workflow and Activity Modeling for Monitoring Surgical Procedures*. Dissertation, Université Henri Poincaré, Nancy.

Potel, M. (1996). Mvp: Model-view-presenter the taligent programming model for c++ and java. *Taligent Inc*, page 20.

Pransky, J. (2001). An intelligent operating room of the future–an interview with the university of california los angeles medical center. *Industrial Robot: An International Journal*, 28(5):376–382.

Roe, D. B. and Wang, Y. (2000). A voice-controlled network for universal control of devices in the OR. *Minimally invasive therapy & allied technologies : MITAT : official journal of the Society for Minimally Invasive Therapy*, 9(3-4):185–191.

Sutherland, J. and van den Heuvel, W. J. (2006). Towards an intelligent hospital environment: adaptive workflow in the or of the future. In *System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on*, volume 5, pages 100b–100b. IEEE.

Ultes, S. and Minker, W. (2014). Managing adaptive spoken dialogue for intelligent environments. *Journal of Ambient Intelligence and Smart Environments*, 6(5):523–539.

JAIST Annotated Corpus of Free Conversation

Kiyoaki Shirai, Tomotaka Fukuoka
Japan Advanced Institute of Science and Technology, Nextremer Co., Ltd
1-1 Asahidai, Nomi, Ishikawa, Japan, 1-30-13 Narimasu, Itabashi, Tokyo, Japan
kshirai@jaist.ac.jp, tomotaka.fukuoka@nextremer.com

Abstract

This paper introduces an annotated corpus of free conversations in Japanese. It is manually annotated with two kinds of linguistic information: dialog act and sympathy. First, each utterance in the free conversation is annotated with its dialog act, which is chosen from a coarse-grained set consisting of nine dialog act labels. Cohen's kappa of the dialog act annotation between two annotators was 0.636. Second, each utterance is judged whether the speaker expresses his/her sympathy or antipathy toward the other participant or the current topic in the conversation. Cohen's kappa of sympathy tagging was 0.27, indicating the difficulty of the sympathy identification task. As a result, the corpus consists of 92,031 utterances in 97 dialogs. Our corpus is the first annotated corpus of Japanese free conversations that is publicly available.

Keywords: Annotated corpus, Dialog Act, Sympathy, Free Conversation

1. Introduction

In recent years, study of an open domain conversation system or free conversation system, which can freely talk with users, has attracted much research interest (Libin and Libin, 2004; Higashinaka et al., 2014a; Higashinaka et al., 2014b). Unlike a task oriented dialog system, an open domain conversation system can chat with users about various topics. Such systems can be used as robotic pets or nursing care robots that can enrich our daily life.

Obviously, language resources are indispensable for the study of free conversation systems. Especially, corpora of free conversations annotated with some linguistic information are valuable. However, for the Japanese language, there is no annotated corpus in the domain of free conversations that is publicly available.

This paper introduces an annotated corpus of free conversations in Japanese, called "JAIST Annotated Corpus of Free Conversations"[1]. It consists of dialogs of two participants, where they freely talk about various topics. Each utterance in the dialogs is annotated with two kinds of tags. One is a dialog act (or speech act), which is the type of utterance that represents the speaker's intention. The other is sympathy. In this paper, sympathy means that the speaker shows interest in the current topic in the conversation. We will report in detail how to construct the corpus as well its statistics. Furthermore, two usage cases of this corpus will be reported: the classification of the dialog acts and the identification of sympathy.

The rest of the paper is organized as follows. Section 2 will discuss related work. Section 3 will report the details of our corpus, including annotation guidelines, the size of the corpus, the distribution of the tags, and inter-annotator agreement. Section 4 will describe two case studies using our corpus. Finally, the paper is concluded in Section 5 .

2. Related Work

One of the well-known dialog corpora is the Switchboard Dialog Act Corpus (University of Colorado at Boulder, 2000). Stolcke et al. (2000) reported that it consisted of a substantial portion of the Switchboard corpus (Godfrey et al., 1992), which was a collection of human–human conversational telephone speech. A total of 1,155 conversations were labeled, comprising 205,000 utterances and 1.4 million words. The SWBD-DAMSL tag set, which was based on the Dialogue Act Markup in Several Layers (DAMSL) tag set (Core and Allen, 1997), was used for annotation. It consisted of 42 dialog act labels. The ICSI Meeting Recorder Dialog Act (MRDA) Corpus (ICSI, 2004) is a collection of 72 hours of speech from 75 naturally occurring meetings (Shriberg et al., 2004). Eleven general tags and thirty-nine specific tags were used for the dialog act annotation. The corpus contained 180,000 utterances with dialog acts in total.

As for Japanese, several studies have been devoted to supervised learning of dialog act classification in free conversations. Isomura et al. (2009) applied Conditional Random Field (CRF) with the features of word unigrams and bigrams that occurred twice or more in the training data as well as the dialog act of the previous utterance. They reported that the accuracy of their method was 75.77%. Meguro et al. (2013a) identified dialog acts in conversations on a microblog, i.e., Twitter. It was a challenging task, since a wide variety of topics and words were used and the sentences were often ungrammatical. The features for machine learning were the n-gram of the semantic classes derived from a thesaurus as well as the n-gram of the characters. Higashinaka et al. (2014a) proposed an open-domain conversational system and developed a dialogue-act estimation module in it. To identify the dialog act of a user's utterance, a Support Vector Machine (SVM) (Cortes and Vapnik, 1995) was trained with features such as the character n-grams, word n-grams, and semantic categories. They used a dialog tag set consisting of 33 dialog acts, which was proposed by Meguro et al. (2013b). The accuracy of the dialog act classification was 45%, while the inter-annotator agreement was 59%.

In these studies, corpora annotated with dialog acts were constructed and used for training classifiers as well as the evaluation of the proposed methods. However, these cor-

[1] "JAIST" is the acronym for the affiliation of the first author.

pora were not released. There is no annotated public corpus of free conversations in Japanese, which can be used by every researcher.
One of the important characteristics in a free conversation is the sympathy of a speaker for topics in a conversation (Anderson and Keltner, 2002; Higashinaka et al., 2008). In the past studies of dialog systems, sympathy was usually considered as one of the dialog act classes (Minami et al., 2010; Sekino and Inoue, 2010; Meguro et al., 2013b; Shriberg et al., 2004). On the other hand, sympathy is independently tagged to the utterance in our corpus, since we think that sympathy plays an important role in a free conversation system. Topics in free conversations are not fixed but could be changed by the speakers at any time. To make the conversation natural and smooth, however, the free conversation system can not arbitrarily change the topics. It is uncomfortable for the user if the system were to suddenly change the topic when the user wants to continue talking on the current topic, or if the system were to keep the same topic when the user is bored and does not want to talk on that topic any more. The sympathy of the user is one of the useful clues to guess what would be a good time for changing the topic. If the user shows sympathy for the current topic, the system should continue the conversation with the same topic. On the other hand, if the user does not display sympathy, the system should provide another topic. The peculiarity of our corpus is that the dialog act and sympathy are tagged separately.

3. The Construction of the Corpus

3.1. The Raw Corpus

The Nagoya University conversation corpus (Fujimura et al., 2012b) was chosen as the texts for the annotation. It consists of transcriptions of 120 free conversations between two or more participants. The total duration of the dialog is about 100 hours. Each utterance was transcribed by hand. The corpus was developed by Fujimura et al. (2012) at Nagoya University, but has now been released by the National Institute for Japanese Language and Linguistics (NINJAL). It is freely available at the web site of NINJAL. Not all, but 97 dialogs, where only two people participate in the conversation, were chosen for the annotation. The statistics of the sub-corpus are shown in Table 1. It indicates that each dialog is rather long.

Number of dialogs	97
Number of utterances	92,020
Average number of utterances per dialog	949

Table 1: Statistics of corpus

3.2. Overview of Annotation

Each utterance in our corpus has the following information.

- Speaker ID

  An identification number of the speaker. It has been already annotated in the Nagoya University conversation corpus.

- Turn taking

  A flag indicating whether the speaker has changed or not. This was automatically annotated.

- Dialog act

  A dialog act of an utterance.

- Sympathy tag

  A tag that represents whether the speaker shows sympathy or antipathy.

We have manually annotated the utterances with the dialog act and sympathy tags.

3.3. Annotation with Dialog Act

Nine dialog acts were formulated for the annotation. Table 2 shows these dialog acts, their definitions, and examples of utterances.
The annotation guidelines of the dialog act are as follows:

- Decide on the dialog act for the utterance, considering its context. To consider the context, the utterances should be annotated in the same order as they occur in the dialog.

- Choose one dialog act for each utterance. When two or more dialog acts are possible, choose the primary one. However, it is allowed to assign multiple dialog acts, but only when the annotator cannot decidedly choose one dialog act.

- Guidelines for Response(Yes/No)
  "Response(Yes/No)" should be the tag for an utterance that consists of a short phrase such as "yes" or "no." The annotation of "Response(Yes/No)" is restricted to the context just after an utterance of "Question(Yes/No)," "Question(What)," "Confirmation," or "Request." It is not necessary to always give the tag "Response(Yes/No)" to the response to an utterance of a "Question(Yes/No)". If the speaker replies to a yes/no question by a declarative sentence, not "Response(Yes/No)," but "Response(Declaration)" should be chosen.

- Guideline for Response(Declaration)
  The tag "Response(Declaration)" should be applied to a speaker's response presented by declarative sentences. The annotation of "Response(Declaration)" is restricted to the context just after the utterance of "Question(Yes/No)," "Question(What)," "Confirmation," or "Request." It is not necessary to always apply the tag "Response(Declaration)" to the response to an utterance of "Question(What)." If the speaker replies to a wh*-question by "yes" or "no," not "Response(Declaration)," but "Response(Yes/No)" should be chosen.

Table 3 presents the numbers of dialog acts and their proportions in the constructed corpus. The most frequent dialog act is "Self-disclosure," followed by "Backchannel," "Response(Declaration)," and "Question(Yes/No)." On the

ID	dialog act	definition	example
d_1	Self-disclosure	Speaker expresses his/her opinion or fact.	*In short, he has a meager vocabulary.*
d_2	Question(Yes/No)	Speaker asks a yes/no question.	*Can I turn on the light for a moment?*
d_3	Question(What)	Speaker asks a question (what, who, when, how, etc.).	*Which country did he come from?*
d_4	Response(Yes/No)	Speaker replies to a question with a short phrase.	*Yes, please.*
d_5	Response(Declaration)	Speaker replies to a question with a declarative sentence.	*Yeah, so, he came from Brazil.*
d_6	Backchannel	Speaker gives a short response	*Uh-huh.*
d_7	Filler	Speaker utters a short phrase to just fill in the time.	*Wow.*
d_8	Confirmation	Speaker confirms the hearer's understanding.	*Really?*
d_9	Request	Speaker requests something of a hearer.	*Please introduce that person to me.*

Table 2: Definition of dialog acts

ID	dialog act	frequency	proportion
d_1	Self-disclosure	53,701	58.35%
d_2	Question(Yes/No)	6,430	6.99%
d_3	Question(What)	3,950	4.29%
d_4	Response(Yes/No)	2,130	2.31%
d_5	Response(Declaration)	7,508	8.16%
d_6	Backchannel	9,216	10.01%
d_7	Filler	4,405	4.79%
d_8	Confirmation	3,940	4.28%
d_9	Request	751	0.82%

Table 3: Distribution of dialog acts

other hand, the utterance of a "Request" seldom appears in free conversations. Its proportion is only 0.8%.

Although it is allowed to assign two or more dialog acts to one utterance, the annotators are required to assign one dialog act as much as possible. As a result, only 11 utterances were annotated with two dialog acts.

A dialog act is assigned by one annotator for each utterance, although two annotators work for the construction of the whole annotated corpus. To check the inter-annotator agreement, only three dialogs were annotated by two annotators. The agreement ratio is 0.773 and Cohen's kappa is 0.636.

3.4. Annotation with Sympathy Tag

Three sympathy tags are defined for the annotation.

Sympathy

This is assigned if a speaker expresses sympathy with the other participant's previous utterance or a current topic in a conversation.

(example) *That is great!*

Antipathy

This is assigned if a speaker expresses antipathy towards the other participant's previous utterance or a current topic in a conversation.

(example) *I can't agree with you.*

Neutral

This is chosen if a speaker expresses neither sympathy nor antipathy.

The annotation guidelines for the sympathy tags are as follows:

- The utterance is likely to be sympathetic or antipathetic when the previous utterance is subjective. For example, if one participant expresses his/her opinion, sentiment, or impression, another participant may express sympathy or antipathy.
- The "Sympathy" tag can be assigned when an utterance shows sympathy or approval. However, if a speaker just shows agreement with the other participant, the "Sympathy" tag is not assigned.

  (example)
  P1: *We will arrive at around 11 tonight.*
  P2: *Yes.*

  Speaker P2 agrees with P1, but does not show any sympathy with the fact said by P1. Therefore, P2 should be tagged as "Neutral."
- The "Antipathy" tag can be assigned when an utterance shows antipathy or bad feeling. As with the sympathy tag, if a speaker just disagrees with the other participant, the "Antipathy" tag is not assigned.

  (example)
  P1: *He is a cunning fellow, isn't he?*
  P2: *I don't think so.*

  The speaker disagrees with P1's comment, but does not express any ill feeling. Therefore, "Neutral" should be chosen for the utterance of P2.

Table 4 shows the number of occurrences of each sympathy tag and its proportion in the constructed corpus. It is found that the number of sympathy and antipathy tags is quite small. Most of utterances are tagged as "Neutral."

sympathy tag	frequency	proportion
Sympathy	1,067	1.16%
Antipathy	222	0.24%
Neutral	90,731	98.60%

Table 4: Distribution of sympathy tags

The same annotators who worked for the dialog act tagging also annotated the corpus with the sympathy tags. Two annotators gave the sympathy tags to only three dialogs, this was done to measure the inter-annotator agreement; the rest

of the dialogs were annotated by one annotator. Cohen's kappa of the sympathy annotation is 0.27. This indicates the difficulty of the sympathy identification task. In particular, the judgment of implicit sympathy tends to be inconsistent. Here implicit sympathy means a sympathetic utterance without any linguistic features that clearly indicate sympathy. Sympathy in such an utterance can be identified by its context and/or prosody. Intuitively, prosody is an important clue to judging the sympathy of a speaker. However, only the transcriptions of the utterances were used: no speech information was used for the annotation. In the future, the definition of "sympathetic utterance" should be clarified in order to have better guidelines for consistent annotation.

4. The Case Studies

Two case studies of the JAIST Annotated Corpus of Free Conversations are reported. In the first case, the corpus was used as labeled data to train a model for the classification of the dialog acts. In the second case, the corpus was used for training a classifier of sympathy identification.

4.1. Classification by Dialog Acts

The task considered in this subsection is to classify a given utterance by its dialog act. Fukuoka and Shirai proposed a method for dialog act classification and evaluated their method on the JAIST Annotated Corpus of Free Conversations (Fukuoka and Shirai, 2017). The present paper briefly introduces their method and the results of the experiment.

4.1.1. Method

Figure 1 shows an overview of the classification model. For each dialog act, a binary classifier that judges whether a given utterance has a dialog act is trained. An optimized set of the features is empirically determined for each classifier of the dialog act. The binary classifiers also calculate the reliability of the judgment. After nine classifiers are applied, one dialog act is chosen by considering the judgment and the reliability of the nine classifiers. L2-regularized logistic regression is used for training the binary classifiers. The probability given by LIBLINEAR (Fan et al., 2008) is used as the reliability of the classification.

Table 5 shows a list of the types of the proposed features. These features were designed by manually analyzing free conversations. The linguistic characteristics of the dialog acts were carefully considered for the feature engineering.

The set of the feature types is optimized for each dialog act by removing ineffective feature types. Figure 2 shows the algorithm of the feature type optimization. E stands for the initial feature set consisting of all proposed features, while E' stands for the optimized one. The function $f(X)$ denotes the F-measure of binary classification of the dialog act for the development data, where the binary classifier was trained with the feature set X. For each feature type f_i in E, if $f(E \setminus \{f_i\})$ is less than or equal to $f(E)$, f_i is regarded as effective and added to E'. After checking all the feature types in E, the optimization is terminated if no more feature type is removed (line 6). Otherwise, we update E by E' (line 11) and repeat the same procedure.

$f(E')$ sometimes becomes lower than $f(E)$ at line 7. This means that removal of each ineffective feature increases the

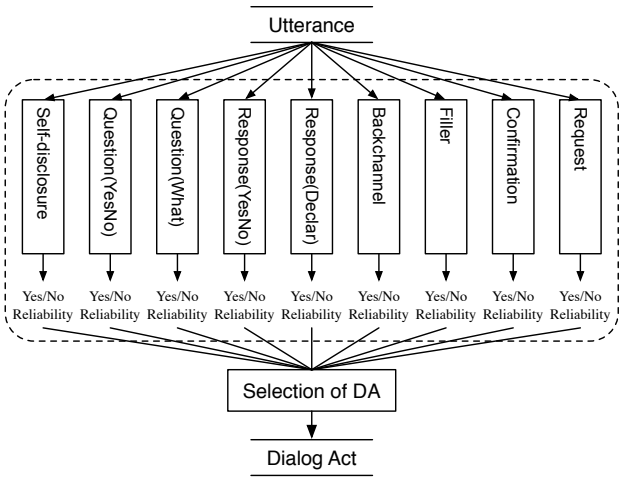


Figure 1: Overview of dialog act classification

F-measure, but the simultaneous removal of two or more ineffective features causes a decrease of the F-measure. In such a case, we choose the most ineffective feature f_x, where the difference between $f(E \setminus \{f_i\})$ and $f(E)$ is the maximum, and remove only f_x from E (line 9).

Input: $E = \{f_1, f_2, \cdots, f_n\}$
Output: E'

```
1:  while true do
2:      E' ← ∅
3:      for all f_i ∈ E do
4:          if f(E) ≥ f(E\{f_i}) then E' ← E'∪{f_i}
5:      end for
6:      if E = E' then return E'
7:      if f(E) > f(E') then
8:          f_x = arg max_{f_i} f(E \ {f_i}) − f(E)
9:          E ← E \ {f_x}
10:     else
11:         E ← E'
12:     end if
13: end while
```

Figure 2: Algorithm of feature set optimization

After the judgment and its reliability for nine dialog acts have been obtained, one dialog act is chosen by Equation (1). We compare the reliability of each classifier ($r(d_i)$) and choose the dialog act with the maximum reliability.

$$\hat{d} = \arg\max_{d_i} r(d_i) \tag{1}$$

In addition, two additional procedures are introduced, as in Equation (2).

$$\hat{d} = \begin{cases} \arg\max_{d_i} w_i \cdot r(d_i) & \text{if rank(1)=Self-disclosure} \\ \text{classify}(\text{rank}(1), \text{rank}(2)) & \\ \qquad \text{if } \{\text{rank(1),rank(2)}\} = \{d_6, d_7\} \text{ or } \{d_2, d_8\} & \\ \arg\max_{d_i} r(d_i) & \text{if otherwise} \end{cases} \tag{2}$$

f_1: word n-gram	f_{14}: keyword of "Backchannel"
f_2: word n-gram in previous utterance	f_{15}: keyword of "Filler"
f_3: content word	f_{16}: key phrase of "Request" at the end of utterance
f_4: content word in previous utterance	f_{17}: key phrase of "Backchannel" at the end of utterance
f_5: word n-gram at the end of utterance	f_{18}: previous dialog acts of the hearer
f_6: word n-gram at the end of previous utterance	f_{19}: previous dialog acts of the speaker
f_7: sequence of function words at the end of utterance	f_{20}: length of utterance
f_8: sequence of function words at the end of previous utterance	f_{21}: turn taking
f_9: pair of word n-gram at the end of current and previous utterances	f_{22}: existence of content word
f_{10}: pair of sequence of function words at the end of current and previous utterances	f_{23}: repetition of content words (1)
f_{11}: keyword of question	f_{24}: repetition of content words (2)
f_{12}: keyword of "Question(Yes/No)"	f_{25}: repetition of content words (3)
f_{13}: keyword of "Response(Yes/No)"	f_{26}: utterance formed by one content word
	f_{27}: utterance formed by one function word
	f_{28}: duplication of words in current utterance

Table 5: Features for dialog act classification

In the preliminary experiment, many utterances were wrongly classified as "Self-disclosure," because the reliability of "Self-disclosure" is usually much higher than the others. This is because the proportion of "Self-disclosure" utterances is the greatest in our corpus, as shown in Table 3. To alleviate the imbalance of the reliability of the dialog acts, the weighted reliability of the dialog acts are compared when the highest ranked dialog act is "Self-disclosure" as in the first case in Equation (2). The weights of the dialog acts, w_i, are optimized on the development data. Furthermore, it was found that two specific pairs of dialog acts are very difficult to distinguish: d_6(Backchannel) & d_7(Filler) and d_2(Question(Yes/No)) & d_8(Confirmation). Therefore, if the first and second ranked dialog acts are (d_6,d_7) or (d_2,d_8), other classifiers that select one of these dialog acts are used to make the final determination of the dialog act, as in the second case of Equation (2). The classifiers are separately trained with the union of the optimized feature sets of two dialog acts.

4.1.2. Experiment

In the experiment, the JAIST Annotated Corpus of Free Conversations was randomly divided into three sets, as shown in Table 6.

	# of dialog	# of utterance
Training set	77	74,228
Development set	10	8,984
Test set	10	8,694

Table 6: Data sets

Table 7 presents the precision (P), recall (R) and F-measure (F) of the classification of each dialog act, as well as their macro- and micro-averages. BL_s stands for the baseline, where a unique feature set was used for training the classifier. The feature set was chosen by the algorithm of Figure 2 so that the F-measure of the classification of all dialog acts is maximized. Pro_p is the proposed method that simply chooses the dialog act with the highest reliability, as in Equation (1). Pro_b is the proposed method that chooses the dialog act by Equation (2).

Pro_b achieved satisfactory results, i.e., 82.5% of the micro-average of the F-measure. Furthermore, the proposed methods outperformed the baseline. It was confirmed, by the McNemar's test, that the difference between BL_s and Pro_b was statistically significant at the 5% level. Although the feature set was optimized for the individual dialog acts, the F-measures of d_8(Confirmation) and d_9(Request) are still low. This may be because the numbers of occurrences of d_8 and d_9 are too small, as shown in Table 1. That is, there are much fewer positive samples than negative samples. One way to resolve this is to apply a technique to learn a classifier from an imbalanced training dataset, such as SMOTE(Chawla et al., 2002).

4.2. Identification of Sympathy

The task to be discussed in this subsection is to judge whether a given utterance is sympathetic or not. Fukuoka and Shirai used the JAIST Annotated Corpus of Free Conversations to develop a method to identify sympathetic utterances in free conversations. The present paper briefly introduces their case study. For more details, see (Fukuoka and Shirai, 2015).

SVM[2] was applied to train a binary classifier to judge whether the given utterance was sympathetic. The features used for training are summarized in Table 8.

F_{rw1} and F_{rw2} are introduced since speakers often show their sympathy by repeating a word in the previous utterance of the other. F_{rw1} simply checks whether the same content word appears in both the current and the previous utterance. On the other hand, F_{rw2} more strictly checks the presence of a repetition of content words: F_{rw2} is activated if either of the conditions below is fulfilled.

- The last predicative word in the previous utterance is also found in the current utterance.
- There is only one content word in the current utterance and it also appears in the previous utterance.

[2] LIBSVM (Chang and Lin, 2011) is used for training SVM.

	BL_s			Pro_p			Pro_b		
	P	R	F	P	R	F	P	R	F
d_1 Self-disclosure	0.851	0.951	0.898	0.852	0.953	0.900	0.859	0.949	0.901
d_2 Question(Yes/No)	0.762	0.745	0.753	0.754	0.751	0.752	0.760	0.753	0.756
d_3 Question(What)	0.787	0.672	0.725	0.807	0.689	0.743	0.797	0.706	0.749
d_4 Response(Yes/No)	0.874	0.900	0.887	0.876	0.880	0.878	0.876	0.880	0.878
d_5 Response(Declaration)	0.819	0.772	0.795	0.818	0.812	0.815	0.811	0.839	0.824
d_6 Backchannel	0.768	0.730	0.748	0.758	0.724	0.741	0.790	0.699	0.741
d_7 Filler	0.608	0.412	0.491	0.607	0.399	0.482	0.627	0.553	0.588
d_8 Confirmation	0.634	0.318	0.424	0.678	0.265	0.381	0.687	0.276	0.394
d_9 Request	0.724	0.214	0.331	0.773	0.173	0.283	0.643	0.184	0.286
Macro-average	0.759	0.635	0.672	0.769	0.628	0.664	0.761	0.649	0.680
Micro-average	0.819	0.819	0.819	0.821	0.821	0.821	0.825	0.825	0.825

Table 7: Result of dialog act classification

F_{ng}	Word unigrams, bigrams, and trigrams of the current and previous utterances
F_{len}	Length of utterance
F_{tu}	Turn taking
F_{rw1}	Repetition of word (1)
F_{rw2}	Repetition of word (2)
F_{rc1}	Repetition of semantic class (1)
F_{rc2}	Repetition of semantic class (2)
F_{da}	Dialog act
F_{end}	Sequence of function words at the end of utterance

Table 8: Features for identification of sympathy

F_{rc1} and F_{rc2} are similar to F_{rw1} and F_{rw2}, but not the repetition of the word but of the semantic class or concept derived from the Japanese thesaurus *Bunruigoihyo* (NINJAL, 2004) is considered.

Combination features, i.e., arbitrary pairs of the features in Table 8, are also used for training.

Since the numbers of the word n-gram feature (F_{ng}) and of combination features are extremely high, a simple feature selection procedure is introduced. The correlation between a sympathy class and a feature f_i is measured by its χ^2 value. Th word n-gram feature and combination feature are discarded when χ^2 is less than certain thresholds T_{ng} and T_{comb}, respectively. These thresholds are optimized on the development data.

In the experiment, the utterances with the "Sympathy" tag are regarded as sympathetic, while the utterances with the "Antipathy" and "Neutral" tags are regarded as non-sympathetic. The JAIST Annotated Corpus of Free Conversations is divided into training, development, and test data, as shown in Table 6. In addition, since the number of sympathetic utterances is small, as shown in Table 4, a balanced dataset including the same number of positive and negative samples was also used for evaluation. It was made by keeping all positive samples and randomly choosing an equal number of negative samples for the training, development, and test datasets.

Tables 9 and 10 show the precision (P), recall (R) and F-measure (F) of the imbalanced (original) and balanced datasets. The baseline system is the classifier trained with only the word n-gram feature (F_{ng}). In both the balanced and imbalanced datasets, Fukuoka's method outperformed the baseline. However, the performance on the imbalanced dataset was not good. Note that the identification of sympathetic utterances in our corpus is very difficult since the positive samples are much fewer than the negative samples. Through an error analysis, we found a few major causes of errors. First, errors often crop up when a previous utterance is long and consists of several sentences. Even when a speaker talks for a long time, the hearer may show sympathy with only one sentence of the many sentences spoken by the speaker. The current system extracts the features from the previous (long) utterance, but most of them are irrelevant to sympathy identification. As a result, too many irrelevant features cause classification errors. Second, many false negative errors are caused by the feature F_{end}, which is a typical sentence-end expression that indicates the sympathy of a speaker. However, such sentence-end expressions do not always appear in sympathetic utterances. Especially, an utterance including this feature is not sympathetic when it and its previous utterance are short. In such a case, the number of extracted features is small and the feature F_{end} causes the misclassification of a non-sympathetic utterance as sympathetic.

	P	R	F
Baseline (F_{ng})	0.23	0.11	0.15
(Fukuoka and Shirai, 2015)	0.28	0.13	0.18

Table 9: Results of sympathy identification on imbalanced data

	P	R	F
Baseline (F_{ng})	0.80	0.73	0.76
(Fukuoka and Shirai, 2015)	0.81	0.76	0.80

Table 10: Results of sympathy identification on balanced data

5. Conclusion

In this paper, we introduced the JAIST Annotated Corpus of Free Conversations. It was the corpus of the free conversa-

tions between two participants. It was manually annotated with two kinds of linguistic information: a dialog act and a sympathy tag. Our corpus is the first annotated corpus of free conversations in Japanese that is publicly available. It is distributed by the non-profit organization *Gengo Shigen Kyokai* (literally "Language Resources Association")[3]. Furthermore, two case studies of this corpus (a classification of the dialog acts and an identification of sympathy) were also presented.

In the future, we plan to enrich the annotation of this corpus with two additional tags: topic shift and subjectivity. A topic shift tag for a conversation may be useful for a study of a dialog control manager in a free conversation system. Another additional piece of information is the subjectivity of the content of the utterance. In general, a subjective utterance reflects more the feeling or emotion of the speaker than an objective utterance. Subjectivity may be an important clue to make a conversation between a user and a system smooth and natural.

6. Bibliographical References

Anderson, C. and Keltner, D. (2002). The role of empathy in the formation and maintenance of social bonds. *Behavioral and Brain Sciences*, 25(1):21–22.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):Article 27.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Core, M. G. and Allen, J. F. (1997). Coding dialogs with the DAMSL annotation scheme. In *Working Notes of the AAAI Fall Symposium on Communicative Action in Humans and Machines*, pages 28–35.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20:273–297.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Fujimura, I., Chiba, S., and Ohso, M. (2012). Lexical and grammatical features of spoken and written Japanese in contrast: Exploring a lexical profiling approach to comparing spoken and written corpora. In *Proceedings of the VIIth GSCP International Conference: Speech and Corpora*, pages 393–398.

Fukuoka, T. and Shirai, K. (2015). Identification of sympathy in free conversation. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, pages 1–9.

Fukuoka, T. and Shirai, K. (2017). Dialog act classification using features intrinsic to dialog acts in an open-domain conversation. *Journal of Natural Language Processing*, 24(4):523–546. (in Japanese).

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520.

Higashinaka, R., Dohsaka, K., and Isozaki, H. (2008). Effects of self-disclosure and empathy in human–computer dialogue. In *Proceedings of the Spoken Language Technology Workshop*, pages 109–112.

Higashinaka, R., Imamura, K., Meguro, T., Miyazaki, C., Kobayashi, N., Sugiyama, H., Hirano, T., Makino, T., and Matsuo, Y. (2014a). Towards an open-domain conversational system fully based on natural language processing. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 928–939.

Higashinaka, R., Kobayashi, N., Hirano, T., Miyazaki, C., Meguro, T., Makino, T., and Matsuo, Y. (2014b). Syntactic filtering and content-based retrieval of Twitter sentences for the generation of system utterances in dialogue systems. In *Proceedings of the 5th International Workshop on Spoken Dialog Systems*, pages 113–123.

Isomura, N., Toriumi, F., and Ishii, K. (2009). Auto-tagging method for evaluating non-task-oriented dialogue agent. *The Transactions of the Institute of Electronics, Information and Communication Engineers A*, 92(11):795–805.

Libin, A. V. and Libin, E. V. (2004). Person–robot interactions from the robopsychologists' point ofview: The robotic psychology and robotherapy approach. In *Proceedings of the IEEE 92*, volume 92, issue 11, pages 1789–1803.

Meguro, T., Higashinaka, R., Sugiyama, H., and Minami, Y. (2013a). Dialogue act tagging for microblog utterances using semantic category patterns. *IPSJ SIG Technical Report*, 2013-SLP-98(1):1–6. (in Japanese).

Meguro, T., Minami, Y., Higashinaka, R., and Dohsaka, K. (2013b). Learning to control listening-oriented dialogue using partially observable Markov decision processes. *ACM Transactions on Speech and Language Processing (TSLP)*, 10(4):Article 15.

Minami, Y., Higashinaka, R., Dohsaka, K., Meguro, T., and Maeda, E. (2010). Trigram dialogue control using POMDPs. In *Proceedings of the Spoken Language Technology Workshop, IEEE*, pages 336–341.

Sekino, T. and Inoue, M. (2010). Fine-grained dialog tagging of utterances. In *Tohoku-Section Convention of Information Processing Society of Japan, 10-6-B3-2*. (in Japanese).

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The ICSI meeting recorder dialog act (MRDA) corpus. In *Proceedings of 5th SIGDIAL Workshop on Discourse and Dialogue*, pages 97–100.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V., and Meteer, M. (2000). Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–373.

7. Language Resource References

National Institute for Japanese Language and Linguistics (NINJAL). (2004). *Bunrui Goi Hyo*. Dainippon Tosho.

[3] http://www.gsk.or.jp/en/catalog/gsk2017-b/

Fujimura, Itsuko and Chiba, Shoju and Ohso, Mieko. (2012b). *Nagoya University conversation corpus*. http://mmsrv.ninjal.ac.jp/nucc/.

International Computer Science Institute (ICSI). (2004). *ICSI Meeting Corpus*. http://www1.icsi.berkeley.edu/Speech/mr/.

University of Colorado at Boulder. (2000). *Switchboard Dialog Act Corpus*. http://compprag.christopherpotts.net/swda.html.

The Metalogue Debate Trainee Corpus: Data Collection and Annotations

Volha Petukhova[1], Andrei Malchanau[1], Youssef Oualil[1], Dietrich Klakow[1], Saturnino Luz[2], Fasih Haider[3], Nick Campbell[3], Dimitris Koryzis[4], Dimitris Spiliotopoulos[5], Pierre Albert[2], Nicklas Linz[6], Jan Alexandersson[6]

[1]Saarland University, Spoken Language Systems, Saarbrücken, Germany

[2]The University of Edinburgh, UK; [3]Trinity College Dublin, Ireland

[4]Hellenic Parliament, Greece; [5]University of Houston, TX, USA

[6]German Research Center for Artificial Intelligence (DFKI), Germany

[1]{v.petukhova,andrei.malchanau,youssef.oualil,dietrich.klakow}@lsv.uni-saarland.de;
[2]{s.luz,pierre.albert}@ed.ac.uk;[3]{haiderf,nick}@tcd.ie; [4]dkoryzis@parliament.gr
[5]dspiliotopoulos@uh.edu; [6]{nicklas.linz,jan.alexandersson}@dfki.de

Abstract

This paper describes the Metalogue Debate Trainee Corpus (DTC). DTC has been collected and annotated in order to facilitate the design of instructional and interactive models for Virtual Debate Coach application - an intelligent tutoring system used by young parliamentarians to train their debate skills. The training is concerned with the use of appropriate multimodal rhetorical devices in order to improve (1) the organization of arguments, (2) arguments' content selection, and (3) argument delivery techniques. DTC contains tracking data from motion and speech capturing devices and semantic annotations - dialogue acts - as defined in ISO 24617-2 and discourse relations as defined in ISO 24617-8. The corpus comes with a manual describing the data collection process, annotation activities including an overview of basic concepts and their definitions including annotation schemes and guidelines on how to apply them, tools and other resources. DTC will be released in the ELRA catalogue in second half of 2018.

Keywords: debate argumentation, multimodal data collection, ISO standard annotations

1. Introduction

We currently observe a steadily growing interest of researchers and practitioners in natural argumentation modelling and in developing argumentation technologies. There are systems developed and deployed for legal domains to assist the lawyer in his search for similar past cases, (Teufel, 1999; Brüninghaus and Ashley, 2005); for mining arguments in social media with the goal to predict consumers sentiment (Bai, 2011), to analyse opinions in public discussions (Murakami and Raymond, 2010), to study citizen engagement (Purpura et al., 2008) and to recognize stance in political online debates (Somasundaran and Wiebe, 2010; Walker et al., 2012a).

Argumentation constitutes an important component of human intelligence. Educational studies have shown that constructing arguments and engaging in argumentative discussion enhance conceptual understanding of the subject matter (Wiley and Voss, 1999; Zohar and Nemet, 2002). Argumentation training systems are designed for the legal domains, e.g. to training hypothetical reasoning (Ashley et al., 2007). The TruthMapping[1] web application facilitates collaborative learning through argumentation. DebateGraph[2] used to train how to prevent opinion manipulation marking inconsistent arguments.

These and other developments were supported by corpora collected for various genres, domains and modalities. For example, the *AIFdb* corpora collection (Lawrence and Reed, 2014) of the Centre for Argument Technology, University of Dundee includes data harvested and analysed from ArguBlogging[3], BBC Radio programmes (e.g. MM2012), Araucaria argument database (Reed, 2006). There is the *Internet Argument Corpus* (IAC) (Walker et al., 2012b) of political debates on internet forums, consisting of about 11,000 discussions and 390,000 posts. Subsets of the data have been annotated for topic, stance, agreement, sarcasm, and nastiness among others. The *Yahoo News Annotated Comments Corpus* (Napoles et al., 2017) is one of the largest annotated corpora of online human argumentative dialogues, with the most detailed set of annotations to identify argumentative, respectful exchanges containing persuasive, informative, and/or sympathetic comments.

Larger projects have been used successfully as resources to study written and spoken argumentative discourse, e.g. *Online Debate Forum*[4], *CE-EMNLP-2015*, also known as IBM corpus, a selection of annotated arguments from Wikipedia articles (Rinott et al., 2015), documents of the *European Court of Human Rights*[5], *UK Youth Parliament* (UKYP)[6] debates (Petukhova et al., 2016), the *American Presidency Project* (APP)[7], and many more.

For the application designed within the Metalogue project[8] - Virtual Debate Coach - an interactive system used to train young parliamentarians to debate efficiently (Petukhova et al., 2017b), the Debate Trainees Corpus (DTC) of 'natural' multimodal arguments was collected. Trainees were trained to make choices from a wide range of rhetorical, lexical, syntactic, pragmatic and prosodic devices to deliver strong persuasive speeches (Petukhova et al., 2017c).

[1]https://www.truthmapping.com/

[2]http://debategraph.org/

[3]http://www.argublogging.com/

[4]http://www.debate.org/

[5]http://echr.coe.int

[6]http://www.ukyouthparliament.org.uk/

[7]http://www.presidency.ucsb.edu/index.php

[8]http://cordis.europa.eu/project/rcn/110655_en.html

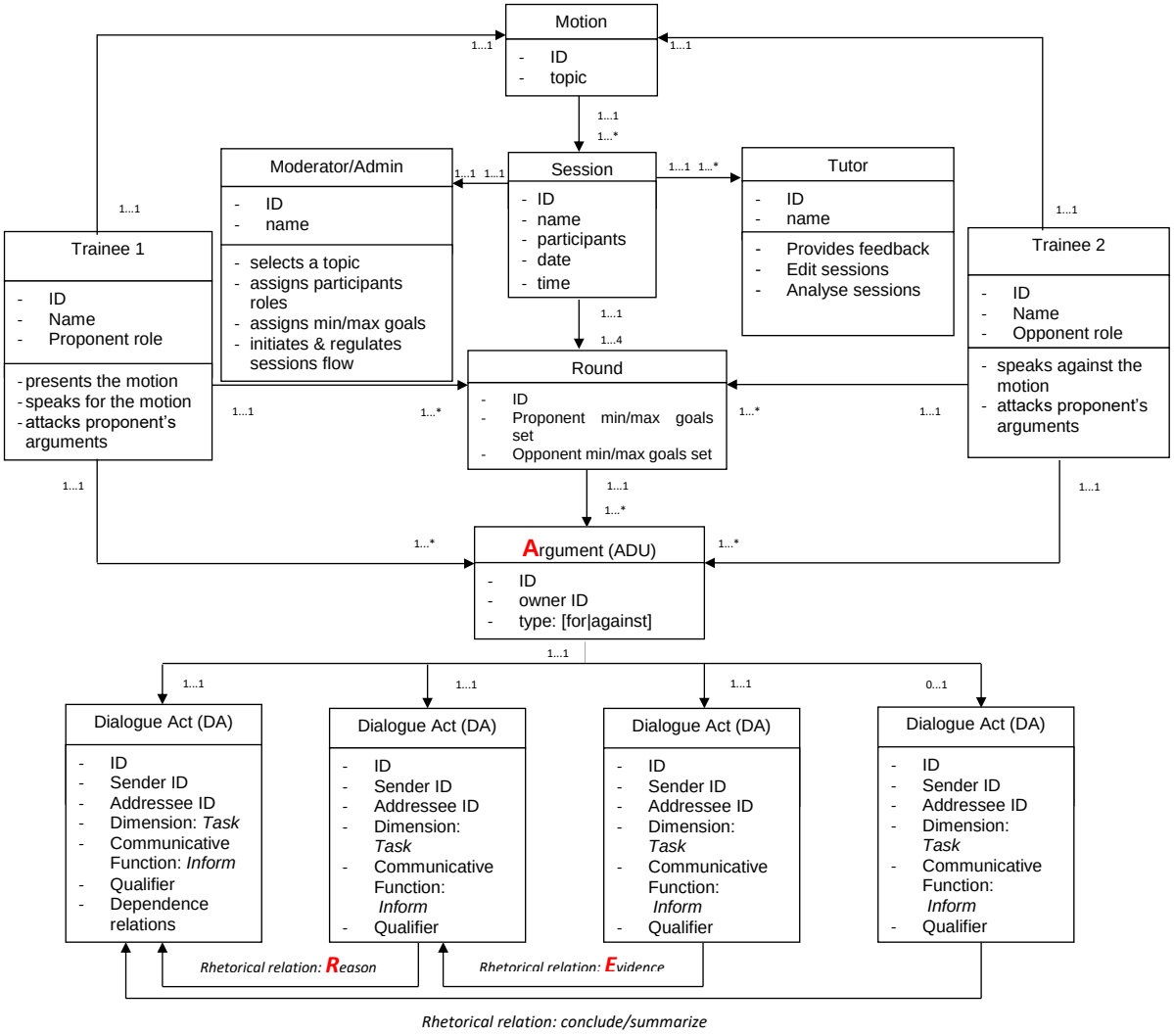


Figure 1: Conceptual class diagram for a Metalogue debate training session.

This paper describes the semantically annotated debate data collection undertaken within the project. The corpus will be delivered with audio and video recordings, Kinect tracking data, automatic and manual transcriptions, ISO compliant semantic annotations. Annotations will be provided in Anvil[9]-specific XML format, and converted to ISO 24617-2 Dialogue Act MarkUp Language (DiAML) (Bunt et al., 2012).

The corpus guidelines, reports, annotation schemes as well as data collection instructional material will be also provided to enable the replication of the carried out experiments.

2. Training Debate Argumentation

Debates, in particular political debates, constitute a large portion of public speeches. Skilled professional debaters give the impression that they truly believe what they say, know how to catch and keep the attention of the audience, and express authority, confidence, respect and friendliness. People generally associate certain speech, personality and interaction features with what they think is a 'good public speaker', see e.g. (Strangert and Deschamps, 2006).

The training of debate skills typically involves ad-hoc face-to-face classroom debates. The debater's skills proficiency level is often judged on three criteria: (1) argument organization, (2) argument content, and (3) argument delivery.

A debate is a communication process in which participants argue *for* or *against* a certain position proposed for the dispute. In a parliament setting, such initial position is called *motion*. In our training scenario, each debate *session* is motivated by a motion - new law proposal or changes to an existing law. A session consists of one or multiple training *rounds*, e.g. our session comprised four debate rounds, featuring different goals assigned to trainees by a *Moderator* (or Admin). Moderators initiate and further formally regulate the session(-s). One or more *Tutors* attend the session and provide feedback to *Trainees*. Tutoring interventions are expected to inform trainees of mistakes, propose corrections, provide instructions, initiate 'try again' rounds, or highlight trainees' successes. This involves immediate real-time 'in-action' and summative 'about-action' feedback (Schön, 1983) on the three debate aspects mentioned

[9] http://www.anvil-software.org/

Issue under debate	Trainees minimum goals to achieve		
	Proponent	Opponent (conservative)	Opponent (liberal)
Smoking ban scope	Not all public places should be affected, allow smoking in bar and restaurants and open air areas like outside buildings, parks and beaches	Forbid smoking inside all public spaces, special smoking areas outside buildings	Allow smoking in special areas in bars and restaurants, open air places also need smoking areas
Tobacco prices	Tobacco price already high, increase no more than 2% a year	Tobacco prices are low, increase by 10% a year	Tobacco prices are still too attractive, increase by 5% a year
Access to tobacco	Tobacco sold in supermarkets, specialized licensed tobacco shops, in bars and restaurants, and vending machines on street with secured buyer's age control	Tobacco should be sold only in special licensed tobacco shops	Tobacco sold in supermarkets but hidden in special containers, prohibited to sell around schools (5km distance) and not available in bar or street vending machines
State control	No police control but municipal and administrative control, no penalties but warnings for the 1st time, repeated disobedience may be punished with penalties	Strong police presence in public places and penalties without warnings	No police control, municipal and administrative control, 1st time disobedience gets warning; second time penalties
Anti-smoking campaign	on TV (state channels 20 min broadcasting time a week); posters in every public place; 'educated' slogans on cigarettes; big newspapers 5 lines a week on the first 2-3 pages	on TV (all channels 30 min broadcasting time a week + one documentary a month); posters in every public place; slogans and scaring images on cigarettes; big newspapers 10 lines a week on the bottom of the front page	on TV (state channels 20 min broadcasting time a week); posters in every public place; 'educated' slogans on cigarettes; big newspapers 10 lines a week on the first 2-3 pages

Table 1: Example of participants' minimal goals in own debate round.

above. The actual debate training session starts by the *Proponent* presenting the motion and an argument in favor of it. An argument is defined as consisting of a statement that can be supported by evidence. A statement (*claim*) is an assertion that deserves attention. There may be a conclusion which presents a result, which can be derived from certain evidence (*premises*). An argument has certain internal structure and correspond to a discourse unit, often called Argumentative Discourse Unit (ADU, see e.g. (Peldszus and Stede, 2013). Claim and premises are dialogue acts related to each other by means of discourse relations, see Section 5.2. The task of the *Opponent* is to attack the proponent's argument by rebutting parts of the argument (premises) or its conclusion, or by undercutting their supporting inference(-s). Both trainees can be in the role of either an proponent or opponent. Moderator(-s) may terminate debating and collect trainees feedback about the debate process and their progress in achieving goals assigned to them. Figure 1 presents the conceptual class diagram for a Metalogue debate training session.

3. Scenario and Data Collection

The specific setting considered for the data collection involves a debate scenario about anti-smoking legislation in Greece. The initial proposal for a smoking ban is supported by the proposing (governmental) party. The goal of the proposer is to get a majority vote while agreeing on as few amendments as possible.

Our core data collection activity involved debate *trainees*, school children aged 14-15 years who have been exposed to little debate training. A session involved a pair of participants: one assigned the role of proposer, the other the role of either liberal or conservative opponent. Each participant was given a set of minimal goals concerning: (1) the total ban on smoking in public spaces; (1) limiting youth access to tobacco products; (3) improving the effectiveness of anti-smoking campaign; (4) state control and reinforcement policy; (5) and raising prices on tobacco products. Participants were not allowed to disclose their goals to the other parties prior to the interaction. Three human tutors evaluated debate performance. Table 1 provides an example of minimal goals that trainees playing different roles should achieve in one debate round.

The collected data consists of 12 sessions with a duration of 2.5 hours, comprising 400 arguments (Argumentative Discourse Units, ADUs[10]) from 6 different bilingual English/Greek speakers.

4. Multimodal Recording and Synchronisation

Training sessions were recorded in a quiet room under special lighting conditions, ensuring that there were no windows behind the participants and that the participants' faces were not in shadow. Two Kinect $V1$ sensors, each facing one participant as much as possible, were placed at a distance of 1.5-2m to the participants. A Kinect $V2$ sensor was also used to track both participants. Body and face tracking data were stored in an XML format containing elements for frames, faces, joint orientation and bone rotation with respect to the camera's coordinates.

Participants faced each other, and markers were placed on the floor to constrain the participants to a limited area. In addition to Kinect's videos, the recordings included two separate video streams, recorded by conventional video cameras. Figure 2 depicts the technical set up for Metalogue debate sessions.

Speech was captured by two audio Tascam Dr-40 recorders and saved in MS WAV format[11]. Speech files are of two types: (1) full dialogue session recorded per speaker, and (2) cut audio files per speaker and roughly per turn (after speaker diarization). Speaker diarization has been partly carried out manually using the Audacity tool[12] and partly automatically using LIUM tool (Rouvier et al., 2013). The speech signal files contain timestamps - start and end time

[10]For more details on segmentation and annotation performed, we refer to (Petukhova et al., 2016).

[11]The recordings were performed in the following setting: sample rate (48KHz), sample size (16-bit), sample format (linear PCM) with stereo channel which was later converted to mono .

[12]`http://www.audacityteam.org/`

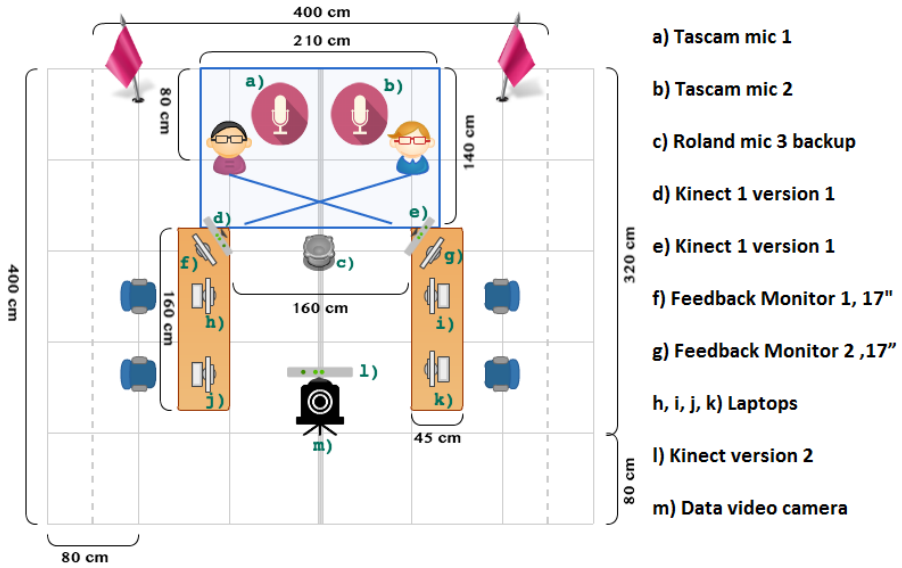


Figure 2: Recording set up for Metalogue debate sessions, adapted from Haider et al. (2017).

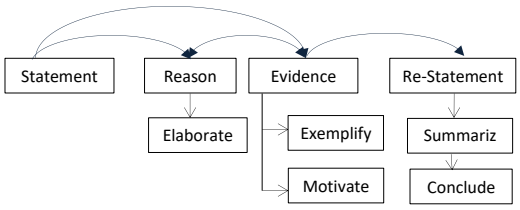


Figure 3: Argument structure observed in Metalogue debate data.

- and additional comments on acoustic and temporal conditions (noise, long silences, etc.) in the file name. For example, `08.22-08.30.n.wav` is the segment which started at 8 minutes and 22 seconds and finished at 8 minutes and 30 seconds during the recording session; and it contains some noise indicated by "n".

The Kinect and video streams were synchronised with audio using the Final Cut Pro X software. The resulting media were converted to a ANVIL compatible format.[13]

Participants' speech has been transcribed semi-automatically by (1) running the Automatic Speech Recognizer (ASR) Kaldi (Povey, 2011) and (2) correcting ASR output manually. Corrected transcriptions are used to re-train/improve language models.[14] All types of transcriptions are stored in plain text and converted to TEI compliant format (ISO, 2006).

5. Annotation Design

Organization of arguments is the planning and preparation involving **A**rgument as a general claim , **R**eason(-s) and **E**vidence. This structure is often called **ARE**[15].

Good debaters are distinguished by concise clear arguments and try to make their arguments understandable for their addressees. For this purpose, debaters often use linguistic cues such as discourse markers and meta-discoursive acts[16] For example, 'I will talk in favour of ... Because ... Since according to the international research shows...'. Thus, *discourse relations* between two or more *dialogue acts* (argument's premises or conclusions) are often marked explicitly by means of discourse markers to support Justification, Motivation, Cause/Result, Background/Evaluation, Evidence and Circumstance links. Figure 3 depicts the most frequently observed of the Metalogue arguments, about 80% of data accounts for this pattern. The main claim, i.e. Statement, is supported by either a Reason or Evidence, and is wrapped up by a Re-Statement in the form of a Summary or Conclusion. For example:

(1) $D1_{21}$[17]: Past anti-smoking campaigns were useless [*Inform*]
$D1_{22}$: I haven't actually seen any of those implemented [*Inform Motivate* $D1_{21}$]
$D1_{23}$: I have personally walked into a store and seen a fourteen years old buying a pack of cigarettes [*Inform Evidence* $D1_{21}$]
$D1_{24}$: Many cases of civil disobedience make this campaign look nice only on paper [*Inform Re-Statement* $D1_{21}$]

5.1. Dialogue Acts

In Metalogue, we mostly consider annotations of semantic and pragmatic multimodal phenomena. For this purposes, *dialogue acts* play an important role. The ISO 24617-2 dialogue act annotation standard is used which allows the analysis of dialogue behaviour as having communicative func-

[13]For more details on the data collection and synchronisation process we refer to (Haider et al., 2017).

[14]It should be noticed that the corpus contains a significant proportion of non-native English speakers, varying in fluency from nearly-native to challenging-to-transcribe.

[15]See `http://www.slideshare.net/Cherye/advanced-debating-techniques` and (Petukhova et al., 2016)

[16](Crismore et al., 1993) define metadiscourse as "linguistic material in texts, written or spoken, which does not add anything to the propositional content but that is intended to help the listener or reader organize, interpret and evaluate the information given", e.g. Shifting Topic, Marking Asides, etc.

[17]Here and henceforth D*k* stands for Debater *k*; the subscript is the index of the identified dialogue act.

Discourse relation	Relative frequency (in %)	Cohen's kappa scores
Elaboration**	28.1	0.67
Evidence**	21.4	0.72
Justify***	16.1	0.76
Condition***	0.7	0.34
Motivation**	1.4	0.48
Background**	0.3	0.18
Cause***	3.4	0.37
Result***	2.2	0.26
Reason*	10.6	0.73
Conclude**	5.7	0.71
Restatement***	10.1	0.76

Table 2: Distribution of Inform acts connected by a discourse relation in the corpus (* defined in DPTB; ** defined by Hovy and Maier, 1995; *** in both taxonomies).

tions in several dimensions. The ISO 24617-2 taxonomy (ISO, 2012) distinguishes 9 dimensions, addressing information about a certain (*Task*); the processing of utterances by the speaker (*Auto-feedback*) or by the addressee (*Allo-feedback*); the management of difficulties in the speaker's contributions (*Own-Communication Management*) or that of the addressee (*Partner Communication Management*); the speaker's need for time to continue the dialogue (*Time Management*); the allocation of the speaker role (*Turn Management*); the structuring of the dialogue (*Dialogue Structuring*); and the management of social obligations (*Social Obligations Management*). For Metalogue purposes, we used 3 additional dimension-specific Discourse Structuring functions that are not included in ISO 26417-2, however, defined in DIT^{++}[18]: Topic Introduction, Topic Shift and Topic Shift Announcement.

In the data, more than 41.4% of the dialogue acts performed by the debaters are Inform acts, which are often connected by discourse relations forming an argument. Small portions of *Set Questions* (3.4%) and *Agreements* or *Disagreements* (1.7%) are observed. Other dialogue acts are concerned with *Turn Management* (22.7%); *Time Management* (21.1%); *Own Communication Management* (7.3%); *Social Obligation Management* (1.2%); and *Discourse Structuring* acts (10%).

5.2. Discourse Relations

Discourse relations were annotated using the annotation scheme designed for the Penn Discourse TreeBank (DPTB) corpus (Prasad et al., 2008)), extended with discourse segment relations from the taxonomy proposed in (Hovy and Maier, 1995). Table 2 presents the types and frequencies of the relations along with the inter-annotator agreement reached annotating each relation type. For relations like Elaboration, Evidence, Justification, Reason, Conclude and Restatement, which are important for the debate argument identification and processing, a substantial agreement has been achieved. The annotated discourse relations were mapped to those defined in ISO 24617-8 standard, which was published after all DTC sessions were annotated.

5.3. Argumentative Discourse Units

We segmented debates into Argumentative Discourse Units (ADUs), defined as a unit which consists of one or more premises and one conclusion, possibly restated or paraphrased several times by the same speaker. To identify ADUs, we followed the approach proposed by (Peldszus and Stede, 2013), who suggest to first segment into Elementary Discourse Units (EDUs)[19] as minimal discourse building blocks, then establish relationships between two or more EDUs, and combine those into ADUs.

Identifying ADUs, we observed a very frequent pattern[20]: an ADU will mostly start with a simple Inform act and end when an Inform Conclude or Restatement is identified, or before another Inform act is performed by the same speaker which is not involved in any discourse relation, see Figure 3 and example in (1), or another speaker claimed the turn.

Finally, to capture support and attack links between arguments produced by different speakers, we identified explicit and implicit agreement and disagreement dialogue acts signalling support or attack of arguments through the *functional dependence relations* defined in (ISO, 2012) between the detected argument conclusions. For example:

(2) $D1_{47}$;$D1_{1.2}$: The government should launch effective anti-smoking campaign before it's too late [*Inform*]
$D2_5$;$D2_{2.1}$: Decision to smoke or not is the personal issue and state shouldn't interfere [*Inform*& *Disagreement* $D1_{47}$] - Attack $D1_{1.2}$
$D7_2$;$D7_{7.1}$: I think public health is one of the most important tasks that the government should perform [*Inform*& *Agreement* $D1_{47}$& *Disagreement* $D2_5$]- Support $D1_{1.2}$/Attack $D2_{2.1}$s

Debater 1 states that an anti-smoking campaign is needed and it is the government responsibility. Debater 2 thinks that smoking is the personal responsibility and government should not interfere. Debater 7 supports argument 1.2 and thereby attacks the arguments 2.1. These links are modelled as part of the debaters' information states, see (Petukhova et al., 2016).

5.4. Dialogue Act Markup Language

ISO standard 24617-2 includes the definition of the Dialogue Act Markup Language (DiAML). The representation of a dialogue act annotation makes use of the XML element `<dialogueAct>` with attributes such as speaker, addressee, communicative function, dimension, qualifiers, dependence relations. Additionally, rhetorical (discourse) relations among dialogue acts are represented by means of `<rhetoLink>` elements. All these types are defined in `diaml` namespace in the defined `DiAML_Types.xsd` scheme. In `DiAML_Containers.xsd` elements such as primary data tokens and sounds, and functional segments are specified without a namespace. The last allows to reuse DiAML_Containers in other (not-`diaml`) schema that may also specify domain-dependent semantics. Such semantics is usually a description of all possible elements of the semantic content of dialogue acts. Semantics can be specified to represent predicate-argument structures, named entities, semantic roles or other semantic relations, etc. Similarly to Metalogue Multi-issue Bargaining (MIB) corpus

[18] `http://dit.uvt.nl/`

[19] EDUs in our data mostly coincide with intentionally defined segments such as dialogue acts - functional segments as defined in ISO 24617-2 (ISO, 2012).

[20] The inter-annotator agreement between three experienced annotators on this task was very high, 0.87 in terms of Cohen's kappa.

Type	Content	Format	Comment
Debate minimal goals cards	4 rounds	pdf	defined for Proponent and Opponent
Metadata	participants (id, native language sex, age at collection)	xml, TEI compliant	generated form participants forms
Signals	sound recordings	mono, 96000Hz sample rate 24-bit sample format	1 channel per speaker
	wav files	mono, 16-bit sample format	cut per speaker/per turn
	Kinect tracking	xml, 30 frames per second	tracked per speaker
		avi videos	recorded per speaker
Automatic Speech Recognition	turn (id, start, end, string)	plain text	automatic
Transcriptions	turn (id, start, end, string)	plain text	manual
	utterance (id, start, end, string)	xml, TEI compliant	automatic
	functional segments (id, start, end, pointers)	xml, TEI compliant	automatic
DA annotations	dialogue act (sender, dimension, communicative function, qualifier functionalDependenceRelation feedbackDependenceRelation) rhetoricalLinks	Anvil and DiAML	manual

Table 3: Metalogue Debate Trainee Corpus overview.

(Petukhova et al., 2016) where negotiation semantics is defied[21], debate domain-specific semantics can be plugged into DiAML. For example:

```
<dialogueAct xml:id="da1" sender="#p1"
  addressee="#p2" dimension="task"
  communicativeFunction="inform"
  target="#fs38"
  qualifier="certain">
    <DebateSemantics>
     <Argument type="for"/>
        <Topic>tax\_increase</Topic>
    </DebateSemantics>
</dialogueAct>
```

6. Corpus Overview

The Metalogue DTC corpus comprises signals, tracking data, transcriptions, meta-data, semantic and pragmatic annotations in standard xml-format. Table 3 provides corpora overview specifying type of data planned for release.
Six types of semantic annotations were performed by two trained and one expert annotators. In total, the Metalogue DTC corpus contains about 10.000 annotated entities. The Metalogue corpus will be published in the ELRA catalogue[22] and be available to the community for research purposes in 2018.

7. Acknowledgements

The reported research is partly funded by the EU FP7 Metalogue project, under grant agreement number: 611073.

8. Bibliographical References

Ashley, K., Pinkwart, N., Lynch, C., and Aleven, V. (2007). Learning by diagramming supreme court oral arguments. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ICAIL '07, pages 271–275, Stanford, California. ACM.

Bai, X. (2011). Predicting consumer sentiments from online text. *Decision Support Systems*, 50(4):732–742.

[21] See also (Petukhova et al., 2017a)

[22] http://catalog.elra.info/

Brüninghaus, S. and Ashley, K. D. (2005). Generating legal arguments and predictions from case texts. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, ICAIL '05, pages 65–74. ACM.

Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A., Hasida, K., Petukhova, V., Popescu-Belis, A., and Traum, D. (2012). Iso 24617-2: A semantically-based standard for dialogue annotation. In *Proceedings 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 430–437, Istanbul, Turkey. ELRA, Paris.

Crismore, A., Markkanen, R., and Steffensen, M. (1993). Metadiscourse in persuasive writing: A study of texts written by american and finnish university students. *Written communication*, 10(1):39–71.

Haider, F., Luz, S., and Campbell, N. (2017). Data collection and synchronisation: Towards a multiperspective multimodal dialogue system with metacognitive abilities. In *Dialogues with Social Robots*, pages 245–256. Springer.

Hovy, E. and Maier, E. (1995). Parsimonious of profligate: how many and which discourse structure relations? unpublished manuscript.

ISO. (2006). *TEI-ISO 24610-1:2006 Language resource management: Feature structures, Part 1: Feature structure representation*. ISO, Geneve.

ISO. (2012). *Language resource management – Semantic annotation framework – Part 2: Dialogue acts. ISO 24617-2*. ISO Central Secretariat, Geneva.

Lawrence, J. and Reed, C. (2014). AIFdb Corpora. In *COMMA*, pages 465–466.

Murakami, A. and Raymond, R. (2010). Support or oppose?: classifying positions in online debates from reply activities and opinion expressions. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 869–875. Association for Computational Linguistics.

Napoles, C., Tetreault, J., Pappu, A., Rosato, E., and Provenzale, B. (2017). Finding good conversations online: The yahoo news annotated comments corpus. In

Proceedings of the 11th Linguistic Annotation Workshop, pages 13–23.

Peldszus, A. and Stede, M. (2013). From argument diagrams to argumentation mining in texts: a survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 7(1):1–31.

Petukhova et al. (2016). Modelling multi-issue bargaining dialogues:data collection, annotation design and corpus. In *Proceedings 9th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3133–3140. ELRA, Paris.

Petukhova, V., Malchanau, A., and Bunt, H. (2016). Modelling argumentative behaviour in parliamentary debates: data collection, analysis and test case. In M. Baldoni, et al., editors, *Principles and Practice of Multi-Agent Systems. Lecture Notes in Artificial Intelligence*, pages 26–46. Springer, Berlin.

Petukhova, V., Bunt, H., and Malchanau, A. (2017a). Computing negotiation update semantics in multi-issue bargaining dialogues. In *Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue (SemDial 2017 - SaarDial)*, pages 114– 124, Saarbrücken, Germany.

Petukhova, V., Mayer, T., Malchanau, A., and Bunt, H. (2017b). Virtual Debate Coach Design: Assessing multimodal argumentation performance. In *Proceedings of the 2017 ACM on International Conference on Multimodal Interaction*, Glasgow, UK. ACM.

Petukhova, V., Raju, M., and Bunt, H. (2017c). Multimodal markers of persuasive speech : designing a Virtual Debate Coach. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 142–146, Stockholm, Sweden. International Speech Communication Association (ISCA), Baixas, France.

Povey, D. (2011). The Kaldi speech recognition toolkit. In *Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding*, Big Island, HI, US. IEEE Signal Processing Society.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. (2008). The Penn Discourse Treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Maroc.

Purpura, S., Cardie, C., and Simons, J. (2008). Active learning for e-rulemaking: Public comment categorization. In *Proceedings of the 2008 International Conference on Digital Government Research*, pages 234–243. Digital Government Society of North America.

Reed, C. (2006). Preliminary results from an argument corpus. *Linguistics in the twenty-first century*, pages 185–196.

Rinott, R., Dankin, L., Perez, C. A., Khapra, M., Aharoni, E., and Slonim, N. (2015). Show me your evidence - an automatic method for context dependent evidence detection. In *EMNLP*, pages 440–450, Lisbon, Portugal. The Association for Computational Linguistics.

Rouvier, M., Dupuy, G., Gay, P., Khoury, E., Merlin, T., and Meignier, S. (2013). An open-source state-of-the-art toolbox for broadcast news diarization. In *INTERSPEECH-2013*, pages 1477–1481.

Schön, D. A. (1983). The reflective practitioner: How professionals think in action. In T. Smith, editor, *Basic Books*. Temple Smith, London.

Somasundaran, S. and Wiebe, J. (2010). Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124. Association for Computational Linguistics.

Strangert, E. and Deschamps, T. (2006). The prosody of public speech - a description of a project. *Lund Unversity Working Papers*, 52:121–124.

Teufel, S. (1999). *Argumentative Zoning: Information Extraction from Scientific Text*. Ph.D. thesis, University of Edinburgh, Edinburgh, Scotland.

Walker, M. A., Anand, P., Abbott, R., Tree, J. E. F., Martell, C., and King, J. (2012a). That is your evidence?: Classifying stance in online political debate. *Decision Support Systems*, 53(4):719–729.

Walker, M. A., Tree, J. E. F., Anand, P., Abbott, R., and King, J. (2012b). A corpus for research on deliberation and debate. In *Proceedings 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 812–817.

Wiley, J. and Voss, J. F. (1999). Constructing arguments from multiple sources: Tasks that promote understanding and not just memory for text. *Journal of Educational Psychology*, 91:301–311.

Zohar, A. and Nemet, F. (2002). Fostering students' knowledge and argumentation skills through dilemmas in human genetics. *Journal of Research in Science Teaching*, 39(1):35–62.

Towards Continuous Dialogue Corpus Creation: writing to corpus and generating from it

Andrei Malchanau[1], **Volha Petukhova**[1] **and Harry Bunt**[2]
[1]Spoken Language Systems Group, Saarland University, Germany
[2]Tilburg Center for Communication and Cognition, Tilburg University, The Netherlands
{andrei.malchanau,v.petukhova}@lsv.uni-saarland.de; harry.bunt@uvt.nl

Abstract

This paper describes a method to create dialogue corpora annotated with interoperable semantic information. The corpus development is performed following the ISO linguistic annotation framework and primary data encoding initiatives. The Continuous Dialogue Corpus Creation (D3C) methodology is proposed, where a corpus is used as a shared repository for analysis and modelling of interactive dialogue behaviour, and for implementation, integration and evaluation of dialogue system components. All these activities are supported by the use of ISO standard data models including annotation schemes, encoding formats, tools, and architectures. Standards also facilitate practical work in dialogue system implementation, deployment, evaluation and re-training, and enabling automatic generation of adequate system behaviour from the data. The proposed methodology is applied to the data-driven design of two multimodal interactive applications - the Virtual Negotiation Coach, used for the training of metacognitive skills in a multi-issue bargaining setting, and the Virtual Debate Coach, used for the training of debate skills in political contexts.

Keywords: dialogue resources, interoperable semantic annotations, international standards, dialogue system design

1. Introduction

A steadily growing interest can be observed in data-driven modelling of phenomena related to natural language, vision, behavioural and organizational processes. Data have become essential to advance the state of the art in many areas including the development of spoken (multimodal) dialogue systems. Conversational applications such as Apple's Siri, Microsoft's Cortana and Google Now became successful and robust partly due to the amount of real user data available to their developers. The most recent trend in dialogue system design involves end-to-end dialogue systems using neural network models trained on previously collected dialogue data, without any detailed specification of dialogue states (Wen et al., 2017; Bayer et al., 2017). This requires large amounts of data to cover a reasonable number of possible dialogue states and participant actions.

Dialogue data have often been collected in Wizard-of-Oz experiments (Dahlbäck et al., 1993), where the dialogue system is replaced by a human Wizard who simulates the system's behaviour according to a pre-defined script.

An alternative is to use simulated users. With good user modelling, a dialogue system could be rapidly prototyped and evaluated. Simulated data sets are, however, rather scarce (Schatzmann et al., 2006).

Resources for data-driven learning of task-oriented systems are also collected with existing systems (Bennett and Rudnicky, 2002; Henderson et al., 2014). For example, the DialPort project addresses the need for dialogue resources by offering a portal connected to different existing dialogue systems (Lee et al., 2017).

Learning algorithms have also been proposed to train a dialogue system online. System behaviour is initially learned from a minimal number of dialogues and is then optimized as more data arrives (Daubigney et al., 2012). As a data collection strategy this approach may not be really successful, since the initial system performance can be rather poor.

Building an annotated dialogue corpus is an expensive activity, especially when it requires manual annotation. Over the years, many annotated dialogue corpora have been created, however annotations and their formats differ from resource to resource. The community has recognized this problem by addressing the interoperability of dialogue resources. ISO 24617-2 "Semantic annotation framework, Part 2: Dialogue acts" (ISO, 2012), in particular aims to contribute to the interoperability of annotated dialogue corpora. New corpora have been created (Petukhova et al., 2014a), existing corpora re-annotated (Bunt et al., 2013) using the standard annotation scheme, and existing annotations mapped to ISO 24617-2 (Petukhova et al., 2014b). The DialogBank is a new language resource that contains dialogues of various kind with gold standard annotations according to the ISO 24617-2 standard (Bunt et al., 2016).

This paper explores yet another way to create semantically annotated dialogue corpora: base corpus developments on the framework of ISO linguistic (i.e. semantic) annotation standards[1]. The approach follows a *continuous corpus creation* methodology where the corpus is used as a shared repository for analysis and modelling of interactive dialogue behaviour, and for implementation and evaluation of the dialogue system. Standard data models (i.e. annotation schemes, encoding and annotation formats) support the corpus development facilitating the creation of semantically rich and interoperable dialogue data for multiple domains, contributing to cost reduction in corpus creation. Standards also support practical work in dialogue system design, evaluation and re-training, and enables automatic generation of adequate system behaviour from the data.

The paper is structured as follows. Section 2 presents the overall methodology, discussing the main principles and key processes related to corpus development. Sec-

[1]We refer to (Ide and Pustejovsky, 2017) for an overview of existing standards.

tion 3 presents the ISO 26417-2 data model introducing the basic concepts and the Dialogue Act Markup Language (DiAML) as the main corpus annotation and exchange format between system components. The proposed approach is illustrated in Section 4, by applying it to recently performed corpus creation activities when designing two different applications - Virtual Debate and Negotiation Coaches (Petukhova et al., 2017b; Petukhova et al., 2017a). The paper is concluded by a summary of the main findings and a discussion of directions for future research.

2. Corpus Creation Methodology

An important step in designing any multimodal dialogue system is to model natural human dialogue behaviour, as a basis for developing dialogue system components. Each module in a dialogue system performs a task such as dialogue act classification, event identification, co-reference resolution, or semantic role labelling, and is integrated according to the adopted architectural approach (e.g. pipeline, multi-agent or multi-threaded), which determines how the modules communicate and exchange their processing results. In such a data-inspired design approach, success will heavily depend on the *quality*, *costs* and *application range* of the underlying corpus data. These three aspects are influenced by multiple variables such as number, tasks and roles of dialogue participants involved in an interactive situation (real vs simulated humans vs artificial agents); dialogue setting, modalities and media available; granularity and nature of annotations and analysis (manual vs automatic vs no annotations); infrastructures, platforms, tools and formats accessible. All these variables impact the corpus creation design, the complexity of the set-up, and the processing steps.

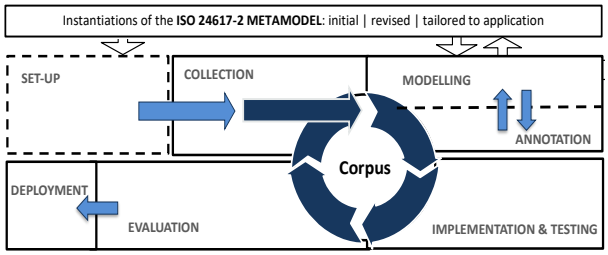


Figure 1: Continuous dialogue corpus creation (D3C).

We propose a continuous dialogue corpus creation (D3C) methodology consisting of the following steps (Fig. 1):

1. Set-up: based on the ISO 24617-2 metamodel define an interaction scenario and specify data collection requirements; provide details for participants roles and tasks, recording setting (equipment and environment) and description of data collection process;
2. Collect: record, encode and store human-human dialogue primary [2] data for the specified scenario;
3. Model: revise the standard data model with attributes derived from annotated data: apply the standard ISO 24617-2 metamodel, include other SemAF concepts on demand and tailor to the application domain;
4. Annotate: apply standard and domain-specific annotation scheme(-s) to classify a particular set of entities and their properties;
5. Implement & Test: build (train) and test dialogue system components based on the underlying annotations performed and resulting dialogue models; optional tests are possible experimenting with tuned and/or modified parameters;
6. Evaluate: perform objective (system performance) and user-based (user perception) evaluation with the integrated dialogue system prototype in the laboratory and close to operational environments; log evaluation sessions and analyse results;
7. Deploy (*optional after each iteration*): write to the corpus, document and prepare to be released including signals, primary data, annotations and corpus manual with schemes, guidelines and format specifications;
8. Repeat steps 1-7 for the full cycle for a refined set-up, or steps 3-6 to re-train system modules based on data obtained in user-based evaluation sessions.

The proposed methodology is in the line with principles of semantic annotation defined in the ISO standard 24617-6 which characterizes the ISO semantic annotation framework (ISO, 2016). The standard includes the CASCADES (**C**onceptual analysis, **A**bstract syntax, **S**emantics, and **C**oncrete syntax for **A**nnotation language **DES**ign) annotation schemes design model (Bunt, 2015). The model enables a systematic (re-)design process: from conceptual ('metamodel') and semantic choices ('abstract' syntax) to more superficial decisions such as the choice of particular XML attributes and values ('concrete' syntax). The method can be used to design of a new annotation scheme or provides support for improving an existing annotation scheme through feedback loops. The CASCADES is integrated with the MATTER method (Pustejovsky et al., 2017) for annotation and data modelling, conceptualized as the Model, Annotate, Train, Test, Evaluate and Revise cycle which inspired the presented methodology.

3. ISO 24617-2 Data Model

Well-established data models are the key enablers for corpus and system development. They are a prerequisite for the corpus to be of good quality, provides ways to systematically incorporate extensions, and ensures interoperability, enabling sharing, merging and comparison with other resources. Data models, formalized descriptions of data objects and relations between them, are designed to capture the structure and relations in diverse types of data and annotations. Well-specified standard resource formats and processes facilitate the exchange of information between dialogue system modules. Mappings between primary data and the data model are operationalized via schema-based data-binding processes (Ide and Romary, 2004).

[2] Data observed or collected directly from first-hand experience such as representation of written (e.g. text), spoken (e.g. orthographic transcriptions of audio) and multimodal (e.g. images or videos) behaviour. Typically, primary data objects are represented by "locations" in an electronic file, e.g. the span of characters comprising a sentence or word, or a point at which a given temporal event begins or ends. More complex data objects may consist of a list or set of contiguous or non-contiguous locations in primary data, see (Ide and Romary, 2004)

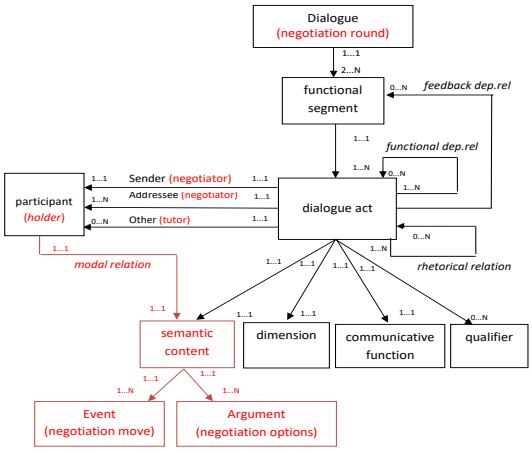


Figure 2: ISO 24617-2 metamodel for dialogue act annotation. Domain-specific extensions marked red, see Sections 4.1 and 4.3.

3.1. Basic concepts

The ISO 24617-2 data model (or 'metamodel', see Fig. 2) represents the fundamental upper-level concepts that are involved in dialogue act annotation. A dialogue consists of two or more functional segments. Each segment is related to one or more dialogue acts, reflecting the possible multifunctionality of functional segments. Each dialogue act has exactly one sender, one or more addressees, and possibly other participants. It has a semantic content of a certain type ('dimension'), and a communicative function, which may have any number of qualifiers. Dialogue acts are possibly related to other dialogue acts through functional dependence and rhetorical relations, and to dialogue segments through feedback dependence relations.

3.2. ISO Dialogue Act Markup Language

The Dialogue MarkUp Language (DiAML) (ISO, 2012) is used as the representation and exchange format in dialogue corpus and system development; DiAML is also used for communication among all system modules, and for representing intermediate and end results.

The representation of annotations in ISO DiAML makes use of the XML element `<dialogueAct>`, which has the following attributes: `@target`, whose value is a functional segment: `@sender`, `@addressee`, `@otherParticipant`; `@dimension`, `@communicativeFunction`; `@functionalDependence` and `@feedbackDependence`, and the three attributes `@certainty`, `@conditionality`, and `@sentiment`, with qualifiers as values. Additionally, rhetorical relations among dialogue acts are represented by means of `<rhetoLink>` elements. DiAML annotations can be extended with a semantic content, also shown in (Bunt et al., 2017), by introducing a `<semanticContent>` element. Consider the following ISO DiAML representation as an example:

```
<dialogueAct xml:id="dap1" sender="#p1"
         addressee="#p2" dimension="task"
         communicativeFunction="inform"
         target="#fsp1">
  <SemanticContent>
    <event xml:id="e1"  type="offer"/>
      <Arg>10_percent</Arg>
    <modalLink holder="#p1" target="#e1"
     modalRel="preference"/>
  </SemanticContent>
</dialogueAct>
```

The `<event>` element, which specifies information about the semantic content of a dialogue act, could be the same as the element with the same name that is used in the ISO annotation schemes for time and events (ISO 24617-1), for semantic roles (ISO 24617-4), and for spatial information (ISO 24617-7), and that has also been proposed for the annotation of modality (Lapina and Petukhova, 2017) and quantification (Bunt, 2017). This opens the possibility to specify quite detailed information about the semantic content of dialogue acts, including domain-specific semantics as shown in (Petukhova et al., 2017a) for negotiations.

4. Use cases

We illustrate the proposed approach by discussing corpus and system development architectures for two applications - the Virtual Debate Coach (VDC, (Petukhova et al., 2017b)) and the Virtual Negotiation Coach (VNC, (Petukhova et al., 2017a)).

4.1. Set-up

The design of any system requires a clear understanding of the users, their goals and the usage situation. This helps to determine the system's functionality, reduces design mistakes and often provides good inspiration and orients. The data collection set-up includes first of all the specification of the intended users and system requirements. A users analysis is conducted to define key user groups (age, gender, cultural and educational backgrounds, etc.) and identify their interest areas, known attitudes, values and priorities. Context of use, settings and users' needs have a direct impact on the role the system will play in an interactive situation, and subsequently on the system functionality. Apart from the communicative tasks that a dialogue system has, namely to understand and adequately react to users' dialogue contributions, a dialogue system has tasks dependent on the application domain in relation to the role(-s) it plays, e.g. as an assistant, adviser or mediator, as a passive observer, as a tutor or as a coach. Users, context and system requirements are used not only to make important design decisions but also to define appropriate verification and evaluation strategies. The evaluation tasks should be representative for most users such that results can be generalized beyond the specific sample.

The 24617-2 ISO data model forms the basis for a domain-specific data collection set-up specifying the type of interaction, participants roles, tasks and actions performed. For example, in our negotiation training scenario, we have a negotiation *session* consisting of one or multiple training *rounds* featuring different goals assigned to trainees by a *Tutor*. Tutors (humans or simulated agents) attend the session and provide feedback to *Trainees* performing a negotiation task. Tutoring interventions are expected to inform trainees of mistakes, propose corrections, provide instructions, initiate 'try again' rounds, or highlight trainees' successes. This involves immediate real-time 'in-action' and summative 'about-action' feedback (Schön, 1983). The

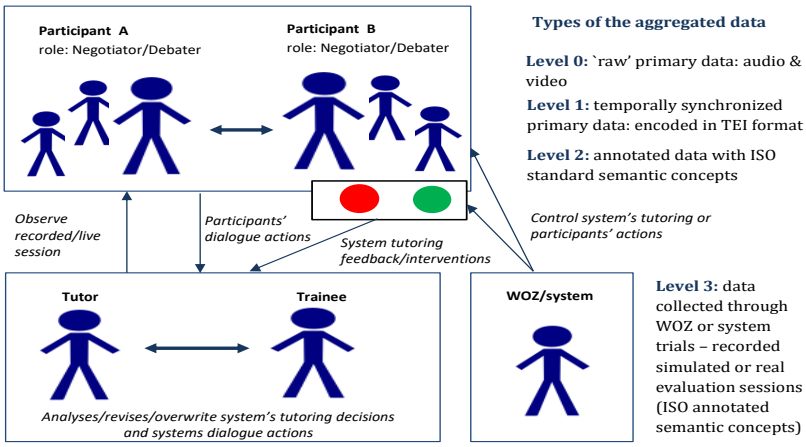


Figure 3: Example of the system and data collection set-up.

task of trainees as *Negotiators* is to propose offers and react to offers of the partner. An extended ISO 24617-2 metamodel (see concepts marked red in Fig. 2) underlies all system and corpus development. A general framework for data collection is set up as shown in Figure 3. We specify participants roles and tasks, as well as data types collected at each recording, processing and evaluation stages including simulated and real dialogue system behaviour in the role of tutor and participant.

The technical set-up specifies recording conditions, equipment, instructions for technical personnel, as well as details on type and granularity of data that should be recorded, and how and where it should be stored, see (Haider et al., 2017).

4.2. Collection and Processing

In multimodal dialogue applications, speech is the main modality. Speech recordings should be of sufficient quality to be used for further processing. Our experience is that recorded 96KHz/24bits audio signals allow a very good tracking of prosodic variations and can be down-sampled to train an Automatic Speech Recognition (ASR) system.

Body movements and facial expressions were tracked using a Kinect 3D sensor. The Kinect video streams and tracking data were temporally synchronised with audio signals with frames of equal 33ms size. The resulting media were converted to view, browse and annotate using the Anvil tool[3].

The speech of a dialogue participant was transcribed semi-automatically by (1) running the ASR system and (2) correcting transcriptions manually. All transcription were stored per participant and dialogue session in TEI compliant format (ISO, 2006).

Prosodic properties related to voice quality, fluency, stress and intonation were computed using PRAAT (Boersma and Weenink, 2009). Kinect body and face tracking data were stored in an `XML` format with elements for frames, faces, joint orientation and bone rotation.

4.3. Annotating and Modelling

The ISO 24617-2 dialogue act taxonomy is designed to capture the meaning of dialogue contributions in multiple dimensions, resulting in multi-layered annotations. Nine dimensions are distinguished, addressing information about a certain *Task*; the processing of utterances by the speaker (*Auto-feedback*) or by the addressee (*Allo-feedback*); the management of difficulties in the speaker's contributions (*Own-Communication Management*) or that of the addressee (*Partner Communication Management*); the speaker's need for time to continue the dialogue (*Time Management*); the allocation of the speaker role (*Turn Management*); the structuring of the dialogue (*Dialogue Structuring*); and the management of social obligations (*Social Obligations Management*).

The semantic content of a dialogue act can be specified in terms of predicate-argument structures, named entities, semantic roles, etc., applying other available standards of the ISO Semantic Annotation Framework. An example of domain-specific semantics is provided for negotiation dialogues in terms of negotiation events such as offer, counter-offer, concession, etc., and their arguments.

In negotiation dialogues, the majority of utterances (59%) is modalized. Participants introducing their options provide information about preferences and abilities. They also request the preferences of their opponents. Parties tend to mention the least desirable events. Apart from preferences and dislikes, a negotiator has certain goals to achieve, which are signalled by teleological modal expressions. Thus, the use of prioritizing modality is frequent. Modality corresponds to the speaker's evaluation of the probability of events; it concerns what the speaker believes to be possible, necessary or desirable. Thus, the classified modality related to the speaker's preferences, priorities, needs and abilities is defined (Lapina and Petukhova, 2017). The metamodel is extended accordingly.

Relations between dialogue acts were annotated, such as the question-answer functional dependence relation, the relation between a feedback act and the dialogue part that the feedback is about, and rhetorical relations, see also (Petukhova et al., 2011). The recognition of dependence and rhetorical relations allows context-dependent interpretation of speaker intentions as well as processing of inter-sentential phenomena, e.g. co-reference resolution.

The full DiAML representation of utterance *P1: I prefer all outdoor smoking allowed* produced by the sender P1 addressed to P2 is a task-related Inform act with the semantic content $\Box offer(ISSUE = 1; VALUE = A)$ is as follows:

```
<dialogueAct xml:id="da1" sender="#p1"
   addressee="#p2" dimension="task"
```

[3] http://www.anvil-software.org/

```
    communicativeFunction="inform"
    target="#fsp1TSK38" qualifier="certain">
     <NegotiationSemantics>
     <NegotiationMove xml:id="nm1"
        type="offer"/>
        <Arg>issue-1; option-A</Arg>
        <modalLink holder="#p1" target="#nm1"
           modalRel="preference"/>
     </NegotiationSemantics>
</dialogueAct>
```

Note that we introduced a <NegotiationSemantics> element into DiAML to represent the domain-specific semantic content of a dialogue act for negotiations. This gives certain flexibility allowing to plug in other domain-specific semantics into DiAML. For instance, for debates, a <DebateSemantics> DiAML element was specified.

```
<dialogueAct xml:id="da1" sender="#p1"
  addressee="#p2" dimension="task"
  communicativeFunction="inform"
  target="#fs38"qualifier="certain">
    <DebateSemantics>
     <Argument type="for"/>
        <Topic>tax\_increase</Topic>
    </DebateSemantics>
</dialogueAct>
```

4.4. Implementation and Testing

The Virtual Negotiation and Debate Coaches "hear" and "see" a wide range of signals, interpret them and act as a negotiation partner or debate opponent, and/or as a tutor.

The speech signals and tracking data serve as input for further processing. The Kaldi-based ASR (Povey, 2011) was trained based on 759 hours of data [4] achieving performance of 34.4% Word Error Rate (WER), see (Singh et al., 2017). For semantic interpretation, the ASR output was used for the event, arguments and modality classification, and communicative function recognition. Conditional Random Fields models (Lafferty et al., 2001) were trained to predict negotiation moves which specify events and their arguments, as well as their boundaries in ASR 1st-best string. The classifier predicts three types of classes: negotiation move (event), issue and preference value (event participants, i.e. semantic roles). A 10-fold cross-validation using 5000 words of transcribed speech from the negotiation domain yielded an F-score of 0.7 on average. The obtained interpretation is of type $offer(ISSUE = X; VALUE = Y)$. The Support Vector Machine (Vapnik, 2013) modality classifiers show accuracies in the range between 73.3 and 82.6% (Petukhova et al., 2017a). The obtained interpretation of a modalized negotiation move stating preference is represented as $\Box offer(ISSUE = X; VALUE = Y)$.

The manually ISO 24617-2 annotated Debate Trainee Corpus (Petukhova et al., 2017b) and Multi-issue Bargaining Corpus (Petukhova et al., 2016) were used to train various communicative function classifiers. Additionally, the in-domain data was enriched with those from the Map-Task (Anderson et al., 1991), AMI(Carletta, 2006), and

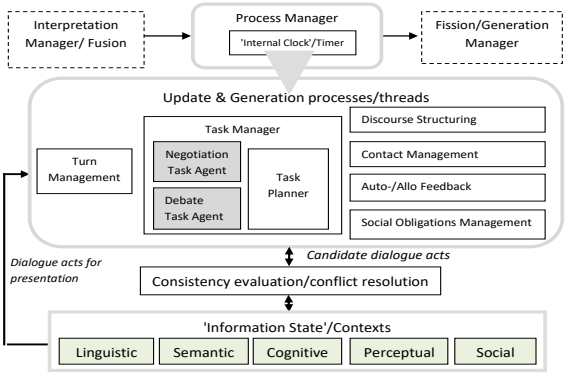


Figure 4: Negotiation and Debate Task Agents (gray boxes) incorporated into the Dialogue Manager architecture.

Switchboard-DAMSL (Jurafsky et al., 1997) corpora. F-scores ranging between 0.83 and 0.86 were obtained in SVM-based clasification experiments, which corresponds to state-of-the-art performance, see (Amanova et al., 2016). Kinect tracked data is used to detect hand/arm co-speech gestures[5] and their types, e.g. beats, adaptors, iconic, deictics and emblems. SVM and Gradient Boosting (Friedman, 2002) classifiers were trained and achieved F-scores of 0.72 (Petukhova et al., 2017c). The motion interpretation component related to hand/arms position detection of the designed Presentation Trainer ((Van Rosmalen et al., 2015; Schneider et al., 2015)) is integrated into the VDC system. Annotations of dependence relations and discourse relations were used to obtain context dependent interpretation. Dependence relations were straightforwardly computed from the dialogue history stored in the linguistic context of the Dialogue Manager (DM), see below. The discourse relations recognition is important for discourse-based argument structure recognition (Petukhova et al., 2017b). The SVM-based classifier yielded F-scores of 0.54 on a coarse 3-class task (Contingency, Evidence, No-Relation) and 0.46 on a fine-grained 7-class task (Justification, Reason, Motivation, Exemplification, Explanation, Exception and No-Relation).

At the semantic fusion level, verbal, prosodic and motion tracking information is combined to obtain complete multimodal dialogue act interpretations, consumed by the Dialogue Manager (DM). The DM, designed as a set of processes (threads), receives data, updates the information state and generates the system next action(-s), see also (Malchanau et al., 2015). The DMs in the VNC and in the VDC applications differ, since the two systems have different roles and tasks. As the Debate Coach, the systems observes debaters' behaviour, evaluates it on criteria related to (1) how convincing is a debater's argumentation; (2) how well are debate arguments structured; and (3) how well is an argument delivered, and generates real-time 'in-action' feedback, see (Petukhova et al., 2017b). As the Negotiation Coach, the system performs as a negotiation partner and also provides feedback on a trainee's negotiation behaviour. Here, the DM incorporates an Negotiation Task Agent (NTA), which interprets and produces negoti-

[4] The following resources were used: the Wall Street Journal WSJ0 corpus, HUB4 News Broadcast data, the VoxForge, the LibriSpeech and AMI corpora.

[5] Co-speech gestures are visible hand/arm movements produced alongside speech and are interpretable only through their semantic relation to the synchronous speech content.

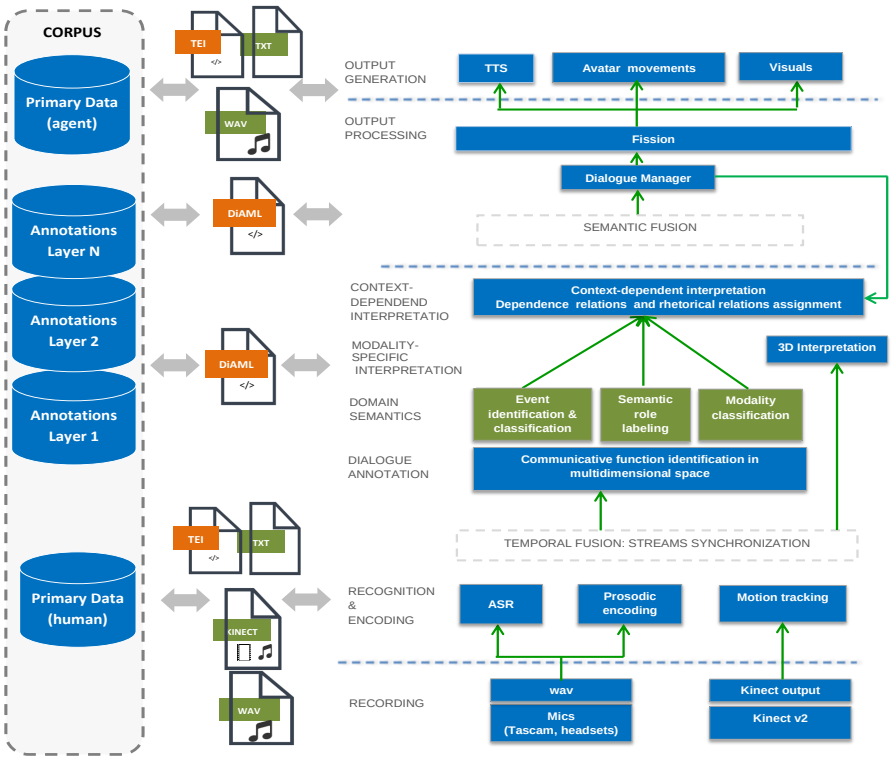


Figure 5: Overall corpus creation and system architecture. From bottom to top, signals are received through input devices, and processed by tailored modules. After annotation concerned with communicative function classification, domain-specific semantic and modality-specific interpretations, context-depended representations are fused and passed to the Dialogue Manager for context model update and system response generation. The system output is rendered in different output modalities. The generated behaviour is written back to the corpus as primary data and representations as annotations, and proposed for editing by human annotators and system module re-training.

ation actions based on the estimation of partner's preferences and goals, and adjusts its strategy according to the perceived level of the opponent's cooperativeness. It reasons about the overall state of the negotiation task, and attempts to identify the best negotiation move for the next action. The DM takes care of feedback and dialogue control actions concerning contact and social obligations management, as well as recovery and error handling actions.

While task-related dialogue acts are application- and user-specific, in a shared cultural and linguistic context, the choices concerning the frequency of dialogue control actions and the variety of expressions are rather limited, notably for feedback and turn management. Models of dialogue control behaviour once designed can therefore be applied in a wide range of communicative situations. This was one of the main motivations behind the multi-layered, multi-threaded DM architecture (Figure 4) where task-related and dialogue control agents/managers are separated. When integrated into different dialogue systems mostly parts of Task Managers are replaced, while other parts were largely re-used without sever changes.

Given the dialogue acts provided by the DM, the Fission module generates system responses, splitting content into different modalities: Avatar[6] and Voice (TTS[7]) actions are generated for the system in partner mode, and visual feedback as tutoring actions. The latter include feedback on presentational aspects and cooperativesness level, visualized by happy and sad face emoticons. At the end of each negotiation and debate session, summative feedback is generated about several aspects of the trainee performance and learning progress.

5. Corpus Evaluation and Deployment

Full session recordings, system recognition and processing results, and the generated dialogue system responses were logged and converted to `.anvil` format for post-processing with the Anvil tool. This tool allows user-defined coding schemes, offering various tier relationships and controlled vocabularies. The tiered format is convenient for transcriptions and annotations in multiple modalities and dimensions. Stretches of communicative multimodal behaviour are marked up with multiple tags, especially when the various tags provide functional information relating to a particular dimension of interaction, such as feedback, turn taking, or time management, see (Petukhova and Bunt, 2010; Bunt et al., 2012; Petukhova, 2011). Annotations are stand-alone and performed using the Anvil specification designed for ISO 24617-2[8].

The Anvil functionality was extended to allow experimenting with variations in system behaviour by tuning, replaying and repairing it. Corrected transcriptions and annotations served: (1) evaluation, measuring inter-annotator agreement to assess corpus data usability, and module-

[6]Commercial software of Charamel GmbH has been used, see (Reinecke, 2003)

[7]Vocalizer of Nuance, `http://www.nuance.com/for-business/text-to-speech/vocalizer/index.htm`, was integrated.

[8]An example specification is available at `http://www.anvil-software.org/data/diaml-spec-v0.5.xml`

based evaluation contrasting system and human performance on all processing tasks; (2) revision of scenario, requirements and data models; and (3) re-training modules on more and better data in order to improve the system performance.

Two resulted corpora are evaluated and deployed when designing the Virtual Negotiation Coach and Virtual Debate Coach applications. They are documented and either released or are in preparation to be released to the research community - Multi-Issue Bargaining Corpus[9] and Debate Trainee Corpus[10].

Figure 5 summarizes the overall corpus and system development framework.

6. Conclusions and future work

Given the importance for a wide range of linguistic applications of data annotated with the interoperable semantic concepts there is a need for cost-effective and accountable solutions to acquire and create such resources on a large scale. This paper proposed the continuous corpus creation methology, supported by ISO semantic annotation standards. On the one hand, the application of the methodology leads to the creation of new interoperable dialogue resources, and on the other hand it enables the design, evaluation and improvement of dialogue system components using these resources. In this approach a corpus is used as a common shared repository which is continuously updated with new recorded and processed data and which is used to generate and tune the system behaviour from it where all system modules exchange messages in standard commonly accepted formats. Well-defined standard data models enable these processes.

Further dialogue resources and tools are in preparation for release. Future work will be also concerned with the integration of new and recently updated ISO standard data models such as those for multimodal events, space and quantification.

7. Bibliographical References

Amanova, D., Petukhova, V., and Klakow, D. (2016). Creating annotated dialogue resources: Cross-domain dialogue act classification. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Slovenia.

Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., et al. (1991). The HCRC Map Task corpus. *Language and speech*, 34(4):351–366.

Bayer, A. O., Stepanov, E. A., and Riccardi, G. (2017). Towards end-to-end spoken dialogue systems with turn embeddings. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2516–2520, Sweden.

Bennett, C. and Rudnicky, A. (2002). The Carnegie Mellon Communicator corpus. In *Proceedings of the 7th International Conference on Spoken Language Processing*, Denver, Colorado.

Boersma, P. and Weenink, D. (2009). Praat: doing phonetics by computer program. Available at `http://www.praat.org/`.

Bunt, H., Kipp, M., and Petukhova, V. (2012). Using DiAML and ANVIL for multimodal dialogue annotation. In *Proceedings 9th International Conference on Language Resources and Evaluation (LREC 2012)*, Turkey.

Bunt, H., Fang, A. C., Liu, X., Cao, J., and Petukhova, V. (2013). Issues in the addition of ISO standard annotations to the Switchboard corpus. In *Proceedings of the 9th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-9)*, pages 59–70, Germany.

Bunt, H., Petukhova, V., Malchanau, A., A., F., and Wijnhoven, K. (2016). The DialogBank. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. Slovenia.

Bunt, H., Petukhova, V., and Fang, A. (2017). Revisiting the ISO standard for dialogue act annotation. In *Proceedings of the 13th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-13)*, pages 37–51, France.

Bunt, H. (2015). On the principles of semantic annotation. In *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-11)*, UK.

Bunt, H. (2017). Towards interoperable annotation of quantification. In *Proceedings of the 13th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-13)*, pages 84–95, France.

Carletta, J. (2006). Announcing the AMI meeting corpus. *The ELRA Newsletter*, 11(1):3–5.

Dahlbäck, N., Jönsson, A., and Ahrenberg, L. (1993). Wizard of Oz studies - why and how. *Knowledge-based systems*, 6(4):258–266.

Daubigney, L., Geist, M., Chandramohan, S., and Pietquin, O. (2012). A comprehensive reinforcement learning framework for dialogue management optimization. *IEEE Journal of Selected Topics in Signal Processing*, 6(8):891–902.

Friedman, J. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378.

Haider, F., Luz, S., and Campbell, N. (2017). Data collection and synchronisation: Towards a multiperspective multimodal dialogue system with metacognitive abilities. In *Dialogues with Social Robots*, pages 245–256. Springer.

Henderson, M., Thomson, B., and Williams, J. D. (2014). The second dialog state tracking challenge. In *Proceedings of the 15th Annual SIGdial Meeting on Discourse and Dialogue*, pages 263–272.

Ide, N. and Pustejovsky, J. (2017). *Handbook of Linguistic Annotation*. Springer.

Ide, N. and Romary, L. (2004). International standard for a linguistic annotation framework. *Natural language engineering*, 10(3-4):211–225.

ISO. (2006). *TEI-ISO 24610-1:2006 Language resource management: Feature structures, Part 1: Feature structure representation*. ISO, Geneve.

ISO. (2012). *Language resource management – Semantic annotation framework – Part 2: Dialogue acts. ISO*

[9] `https://catalog.ldc.upenn.edu/LDC2017S11`

[10] See Petukhova et al., 2018, in this volume.

24617-2. ISO Central Secretariat, Geneva.

ISO. (2016). *Language resource management – Semantic annotation framework – Part 6: Principles of Semantic Annotation. ISO 24617-6*. ISO Central Secretariat, Geneva.

Jurafsky, D., Schriberg, E., and Biasca, D. (1997). Switchboard SWBD-DAMSL shallow-discourse-function annotation: Coders manual.

Lafferty, J., McCallum, A., Pereira, F., et al. (2001). Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML '01*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lapina, V. and Petukhova, V. (2017). Classification of modal meaning in negotiation dialogues. In *Proceedings of the 13th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-13)*, pages 59–70, France.

Lee, K., Zhao, T., Du, Y., Cai, E., Lu, A., Pincus, E., Traum, D., Ultes, S., Rojas Barahona, L. M., Gasic, M., Young, S., and Eskenazi, M. (2017). Dialport, gone live: An update after a year of development. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 170–173, Germany.

Malchanau, A., Petukhova, V., Bunt, H., and Klakow, D. (2015). Multidimensional dialogue management for tutoring systems. In *Proceedings of the 7th Language and Technology Conference (LTC 2015)*, Poland.

Petukhova, V. and Bunt, H. (2010). Towards an integrated scheme for semantic annotation of multimodal dialogue data. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2010)*, Malta.

Petukhova, V. et al. (2018). The Metalogue Debate Trainee Corpus: Data collection and annotations. In *Proceedings of the 11th International Conference on Language Resources and Evaluation(LREC 2018)*, Japan.

Petukhova, V., Prévot, L., and Bunt, H. (2011). Multi-level discourse relations between dialogue units. In *Proceedings of the 6th Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-6)*, pages 18–28, Oxford: Oxford University.

Petukhova, V., Gropp, M., Klakow, D., Schmidt, A., Eigner, G., Topf, M., Srb, S., Motlicek, P., Potard, B., Dines, J., et al. (2014a). The DBOX corpus collection of spoken human-human and human-machine dialogues. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, Iceland.

Petukhova, V., Malchanau, A., and Bunt, H. (2014b). Interoperability of dialogue corpora through ISO 24617-2-based querying. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, Iceland.

Petukhova, V., Stevens, C., de Weerd, H., Taatgen, N., Cnossen, F., and Malchanau, A. (2016). Modelling multi-issue bargaining dialogues: Data collection, annotation design and corpus. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Slovenia.

Petukhova, V., Bunt, H., and Malchanau, A. (2017a). Computing negotiation update semantics in multi-issue bargaining dialogues. In *Proceedings of the SemDial 2017 (SaarDial) Workshop on the Semantics and Pragmatics of Dialogue*, Germany.

Petukhova, V., Mayer, T., Malchanau, A., and Bunt, H. (2017b). Virtual Debate Coach Design: Assessing multimodal argumentation performance. In *Proceedings of the 2017 ACM on International Conference on Multimodal Interaction (ICMI 2017)*, UK. ACM.

Petukhova, V., Raju, M., and Bunt, H. (2017c). Multimodal markers of persuasive speech : designing a Virtual Debate Coach. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 142–146, Sweden.

Petukhova, V. (2011). *Multidimensional Dialogue Modelling. PhD dissertation*. Tilburg University, The Netherlands.

Povey, D. (2011). The Kaldi speech recognition toolkit. In *Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding*, Big Island, HI, US. IEEE Signal Processing Society.

Pustejovsky, J., Bunt, H., and Zaenen, A. (2017). Designing annotation schemes: From theory to model. In *Handbook of Linguistic Annotation*, pages 21–72. Springer.

Reinecke, A. (2003). Designing commercial applications with life-like characters. *Lecture notes in computer science*, pages 181–181.

Schatzmann, J., Weilhammer, K., Stuttle, M., and Young, S. (2006). A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The knowledge engineering review*, 21(2):97–126.

Schneider, J., Börner, D., Van Rosmalen, P., and Specht, M. (2015). Presentation trainer, your public speaking multimodal coach. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction (ICMI 2015)*, pages 539–546, Seattle, WA, USA. ACM.

Schön, D. A. (1983). The reflective practitioner: How professionals think in action. In T. Smith, editor, *Basic Books*. Temple Smith, London.

Singh, M., Oualil, Y., and Klakow, D. (2017). Approximated and domain-adapted LSTM language models for first-pass decoding in speech recognition. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association*, Sweden.

Van Rosmalen, P., Börner, D., Schneider, J., Petukhova, V., and Van Helvert, J. (2015). Feedback design in multimodal dialogue systems. In *Proceedings of the 7th International Conference on Computer Supported Education*, pages 209–217, Portugal.

Vapnik, V. N. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

Wen, T.-H., Vandyke, D., Mrksic, N., Gasic, M., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., and Young, S. (2017). A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th European Chapter of the Association for Computational Linguistics (EACL 2017)*, Spain.

MYCanCor: A Video Corpus of spoken Malaysian Cantonese

Andreas Liesenfeld
Nanyang Technological University
Singapore
LIES0002@ntu.edu.sg

Abstract

The Malaysia Cantonese Corpus (MYCanCor) is a collection of recordings of Malaysian Cantonese speech mainly collected in Perak, Malaysia. The corpus consists of around 20 hours of video recordings of spontaneous talk-in-interaction (56 settings) typically involving 2-4 speakers. A short scene description as well as basic speaker information is provided for each recording. The corpus is transcribed in CHAT (minCHAT) format and presented in traditional Chinese characters (UTF8) using the Hong Kong Supplementary Character Set (HKSCS). MYCanCor is expected to be a useful resource for researchers interested in any aspect of spoken language processing or Chinese multimodal corpora.

Keywords: Malaysian Cantonese, spoken corpora, naturally-occurring talk-in-interaction

1. Corpus description

The Malaysia Cantonese Corpus (MYCanCor) was conceptualized as a repository of naturally occurring conversations collected in the Malaysian Cantonese speech community. It consists of video recordings of a variety of everyday settings where spontaneous everyday conversation commonly takes place, such as:

- Family dinner conversations (private homes)
- Conversations over lunch or dinner with friends, relatives or colleagues (Restaurants and street food vendors)
- Car and public transport rides with family members, friends or colleagues (private car or public bus)
- Conversations at work between colleagues as well as between employees and manager (various office spaces)
- Conversations between customers and vendors (shop or factory setting)
- Conversations between local citizens and government representatives (local government office)
- Conversations between students as well as between students and staff (various educational institutions)

The conversations typically involve 2-4 speakers (up to a maximum of 8 speakers) of Malaysian Cantonese and are between 2 and 40 minutes long. Topics include a wide range of commonplace activities related to the setting in which the conversation takes place, frequently including elements of:

- storytelling and narrative sequences
- management of social obligations including greeting and leave segments
- social chat and expressions of emotional states
- task-oriented dialog such as directions while driving or task instructions in workspace environment
- management of authority in dialog, e.g. task assignment and task-oriented dialog in a workspace environment

The recordings were collected mainly in the city of Ipoh and other places in the state of Perak in 2017 and are now available to the research community as a collection of 56 video recordings (around 20 hours of video content).
Each conversation is transcribed in CHAT format at the minCHAT level (see 3. Transcription). Each video recording also comes with a scene description written in full sentence format that provides the following information:

- brief description of the setting in which the recording took place
- summary of the topics talked about
- specification of the task, in case the recording includes task-oriented sequences
- brief summary of emotional expressions used and brief description of the emotional states talked about, in case the recording features frequent expressions of emotional states
- information about age and gender of each participant

Each recording is available in both high definition video (.mov) and uncompressed audio format (.wav) with a transcription in CHAT format (.cha) (MacWhinney, 2000) and a scene description (.txt).

1.1. Technical specifications

All 56 conversations were filmed from one angle with one HD video camera (1080p) providing access to facial expression and gesture information of most (but not all) participants. The camera was set up in such a way that in almost all cases all participants appear on the screen.
Audio data was captured with one directional supercardioid microphone in uncompressed monophonic wave format (96kHz / 24-bit). The recordings took place in a real world environment and feature natural background noise (for example in a restaurant or workspace settings).

2. Data collection and sampling

The conversations featured in the corpus aim to present naturally-occurring everyday speech. All recordings took place after informed consent was obtained from each participant and all conversations were recorded after the researcher had left the scene. Several measures were taken to minimize the influence of the researcher on the naturalness of the recording and to mitigate the effects of the observer's paradox (Labov, 1972). A basic level of rapport was built up with participants before the recording. Recordings that show an obvious influence of the researcher on the produced utterances have not been included in the corpus. Building on several months of fieldwork in Ipoh and other parts of Perak, a multiple-entry chain sampling method was used in order to recruit participants from a wider variety of socio-economic and educational backgrounds. However, no metrics have been recorded to attest for this. Most interlocutors of a single recorded conversation knew each other prior the recording as friends, family members or colleagues. The corpus is roughly balanced for gender and age group (see Table 1).

Variable	Distribution
Gender (binary)	Male: 48% Female: 52%
Age group (years)	under 30: 33% 30-60: 45% 60+: 22%

Table 1: Gender and age group distribution in MYCanCor (rounded)

Reflecting common practice in Malaysia, the corpus also features code mixing of Cantonese speech with English, Malay, Mandarin and Southern Min. However, an estimated 95% of the recorded utterances are in Cantonese and the most frequent code mixing language is English (as well as local varieties of English such as Malaysian English and Singlish).

Data collection and recruitment of participants was in line with the British Association for Applied Linguistics (BAAL) Recommendations on Good Practice in Applied Linguistics (https://baal.org.uk/resources/). Participants were recruited under two prerequisites:

- self-proclaimed ability to have a conversation in Cantonese (proficiency in written Chinese not required)
- having spent two-thirds of one's life in Malaysia (self-proclaimed)

All participants are self-proclaimed speakers of Malaysian Cantonese Chinese. This variety of Cantonese Chinese somewhat differs from the varieties spoken in speech communities in China (such as the speech community in the Hong Kong SAR and those in Guangdong and Guangxi province). For instance, differences are evident in prosody, lexical token preference and code-mixing. However, speakers of this variety are generally able to fluently communicate with members of, for instance, the Hong Kong Cantonese speech community. Reflecting the Malaysian Cantonese Chinese speech community in general, the level of proficiency considerably varies among participants. Participants may be English-, Mandarin- or Malay-educated and may speak Cantonese as a second language, home language or work language alongside with other languages. Code-mixing with other languages such as English, Singlish, Mandarin Chinese, Hokkien Chinese, Hakka Chinese or Malay is a commonplace phenomenon in the speech community. Recruitment of participants was not tied to ethnic groups, reflecting the ethnically diverse nature of the local speech community.

The data collection mainly took place in the state of Perak (especially the city of Ipoh), the region has the highest percentage of Cantonese speakers in Malaysia (Tan, 2005). Cantonese Chinese is seldom used by educational institutions in Perak. Secondary education is commonly available only in Malay, English or Mandarin Chinese. Tertiary education only in Malay and/or English. Based on participant observation in 2017, however, Cantonese is the lingua franca of the ethnic Chinese community in Ipoh as well as in many (but not all) other cities and villages in Perak, such as Taiping, Kuala Kangsar, Teluk Intan and Kampar. Based on these observations, Perak features one of the biggest, if not the biggest, Cantonese speech communities outside of China.

3. Transcription

All conversations are transcribed using the minCHAT format, a basic version of the CHAT format (MacWhinney, 2000). The transcripts focus on the word level, aiming to identify words and sentences in the utterances, prioritizing lexis over, for example, phonological aspects. In this sense the transcripts aim to provide a modest starting point for possible further annotation of more aspects of the complexity of naturally-occurring speech. In order to achieve a high level of accuracy and consistency, all transcripts have been proofread by Malaysian Cantonese speakers with relevant experience and backgrounds. The transcripts are presented in Traditional Chinese characters (UTF8) including the Hong Kong Supplementary Character Set (HKSCS). Only identifiable units that can be presented by Chinese characters are transcribed. For example, the utterance ('gw', 'o', 'ng', '2'), ('d', 'u', 'ng', '1'), ('w', 'aa', '', '2') (onset, nucleus, coda, tone) would be transcribed as 廣東話 . But the utterance ('gw', 'o', 'ng', '2'),('d', '', '', '1') would be transcribed as 廣xxx since ('d', '', '', '1') has no defined corresponding character in this format. Completely unintelligible utterances are also transcribed as xxx (see Example 1).

Cantonese-specific characters that are not (yet) supported by HKSCS are presented as romanized strings following the Jyutping Romanization Scheme. Code switching utterances in Mandarin are presented in Traditional Chinese characters (UTF8), for Hokkien and Southern Min the Taiwan Romanization system (Tai-lo) was used. Modal Particles (語氣詞) and Modal Particle Morphemes (語氣語素) are transcribed following UTF8+HKSCS standards. For all utterances that are supported by UTF8+HKSDS no separate romanization in Jyutping or Pinyin is provided.

Following minCHAT requirements, utterances have to end with an utterance terminator (such as period, exclamation mark or question mark). In order to fulfill this requirement, the transcript uses periods to terminate all utterances regardless whether they are questions or exclamations. No question or exclamation marks (標點符號) are annotated, but question and exclamatory particles are transcribed.
A separate CHAT transcription file was created for each of the 56 conversations, but it should be no problem to combine the transcripts into one file using one of the many tools available for CHAT. Each file (.cha) is verified as machine-readable using the CHECK function of the CLAN editor.

3.1. Word segmentation and annotation

The transcripts do not include word segmentation or chunking of Chinese characters. Utterances are transcribed as continuous strings of Chinese characters in UTF8 encoding. The minCHAT transcript does, however, separate segments of utterances based on pauses of considerable length (more than 0.1 seconds) (see Example 1). Also, no Part-of-Speech tags or further syntactic annotation is provided as part of the transcript. No additional (systematic) annotation of gestures, facial expression or events is provided.

```
@Begin
@Languages: zho-yue
@Participants: P1 Wong Older Sister,
P2 Chan Younger Sister
@ID: zho-yue|mycancor|P1|27;1.10||||
Target_P2|||
@ID: zho-yue|mycancor|P2|39;2.||||
Target_P1|||
*P2: 你食咩啊.
%com: every utterance ends with
  an utterance terminator (period).
*P1: 白果薏米.
*P2: 同怡保好似好唔同 哈哈哈.
*P1: 唔同啊(0.1)冇得比啦.
%com: Utterances are segmented by
      pauses exceeding 0.1 seconds.
*P2: 依但係.
*P1: 白 但因為 因為佢冇煮溶個.
%com: All lexical items
      (onset, nucleus, coda, tone) are
      transcribed as Chinese
      characters.
%act: P2 points at the bowl.
%com: Gestures may be annotated
      as informal descriptions.
*P2: xxx睇下 個腐竹.
%com: Unintelligible or incomplete
      lexical units are
      transcribed as xxx.
*P1: 個腐竹 呃 係咯 同埋唔知點解佢
     唔係白色咯.
%com: Modal Participles and Modal
      Particle Morphemes are
      transcribed following
      UTF8+HKSDS conventions.
@End
```

Example 1: MYCanCor transcription example in minCHAT format. Each file begins with a header. Comments are marked with %com. Gestures, actions and scene descriptions with %act.

4. Possible applications

The Malaysia Cantonese Corpus might be of interest to any researcher interested in video corpora or Malaysian Cantonese. To the author's knowledge, MYCanCor is in fact the first language resource available that is concerned with the Cantonese speech community in Malaysia. Although the corpus is of rather small size, the data that went into its making should be of sufficient quality to make it a useful resource for a wide variety of applications.
The dataset was compiled as part of several research projects in the area of interactional linguistics and spoken language processing. The described transcription files were produced as part of these projects.
Considerable effort also went into providing accurate minCHAT transcription for each conversation. One of the advantages of the CHAT format is that the provided minCHAT can be adjusted or expanded with relative ease to encode additional phenomena, should the need arise. The design choices made should allow a combination with other resources with relative ease. The minCHAT transcription, for example, can be expanded to midCHAT or even a full conversation analytical (CA) transcription. Eventual additional layers or syntactic, semantic, phonological or prosodic annotation can also be integrated in CHAT. Facial expressions, gaze or gestures can be added for all conversations given that the video data provides sufficient visual information.
A problem in building Asian language resources is the existence of multiple and often competing writing systems and encoding systems. This is also true for Cantonese. Unicode has only recently included the special Hong Kong characters and replaced BIG-5 as the most popular encoding standard for Cantonese. There are also several completing romanization schemes for Cantonese, with the Jyutping Romanization scheme developed by the Linguistic Society of Hong Kong probably being the most widely used standard now. MYCanCor currently uses Unicode encoding without romanization, but several tools exist to facilitate the creation of a romanization if needed.
A good way to get started with processing Cantonese speech data are the resources compiled in the pycantonese library in Python (Lee, 2015) that can be found here: http://pycantonese.org. This resource also incorporates another Cantonese speech corpus focusing on Hong Kong Cantonese that features a somewhat similar design, the HKCanCor (K.K. Luke and May L.Y. Wong, 2015). Because of the similar design choices, HKCanCor would probably be a fitting counterpart for a comparative study of the Hong Kong and Perak Cantonese speech communities.
Several research projects involving MYCanCor are planned in the area of action formation and ascription. As part of these projects, additional transcription data that focuses on

the annotation of social action and communicative intention might become available in the future.

5. Data access and future updates

MYCanCor is currently hosted at Nanyang Technological University in Singapore and maintained by the author. Pending license restrictions, the dataset is available to the research community in three different forms. Transcription data is available in the form of anonymized CHAT format transcription files. In order to protect the identity of all participants, these .chat transcripts do not contain any references related to the identity of the participants. In addition, all location-specific information such as place names and other personal identifiers have been removed.

Audio data is available in form of .wav files that have been edited to protect the identity of the participants. This includes, for example, the adjustment of the pitch range. Video files are only available as post-edited .mov files with a range of non-reversible chromatic, saturation and blur filters applied. Please refer to Figure 1 for a reference example of the visual information that such post-edited video files contain.



Figure 1: Post-edited visual information in MYCanCor.

For more information regarding licensing please visit http://mycancor.andreasliesenfeld.com. This website also provides further information on how to request access to the dataset, lists all current and planned research projects that involves the use of MYCanCor data and provides a detailed description of all available transcription and annotation files.

Future updates regarding MYCanCor will be made available either on this website or on https://github.com/liesenf/mycancor.

6. Bibliographical References

Labov, W. (1972). *Sociolinguistic patterns*. Number 4. University of Pennsylvania Press.

Lee, J. L. (2015). Pycantonese: Cantonese linguistic research in the age of big data. *Talk at the Childhood Bilingualism Research Centre, the Chinese University of Hong Kong*.

MacWhinney, B. (2000). *The CHILDES project: The database*, volume 2. Psychology Press.

Tan, C.-B., (2005). *Chinese in Malaysia*, pages 697–706. Springer, US.

7. Language Resource References

K.K. Luke and May L.Y. Wong. (2015). *The Hong Kong Cantonese Corpus: Design and Uses*. Journal of Chinese Linguistics, Monograph no. 25.

KTH Tangrams: A Dataset for Research on Alignment and Conceptual Pacts in Task-Oriented Dialogue

Todd Shore, Theofronia Androulakaki, Gabriel Skantze
KTH Speech, Music and Hearing
Stockholm, Sweden
tcshore@kth.se, androu@kth.se, gabriel@speech.kth.se

Abstract

There is a growing body of research focused on task-oriented instructor-manipulator dialogue, whereby one dialogue participant initiates a reference to an entity in a common environment while the other participant must resolve this reference in order to manipulate said entity. Many of these works are based on disparate if nevertheless similar datasets. This paper described an English corpus of referring expressions in relatively free, unrestricted dialogue with physical features generated in a simulation, which facilitate analysis of dialogic linguistic phenomena regarding alignment in the formation of referring expressions known as conceptual pacts.

Keywords: reference, conceptual pacts, task-oriented dialogue

1. Introduction

There is recent interest in the role of referring expressions (REs) in situated dialogue and the alignment of referring language (RL) between dialogue participants (Barr and Keysar, 2002; Foster et al., 2006; Zarrieß et al., 2016; Aina et al., 2017). These datasets are useful for studying general patterns of alignment but are not specifically tailored to studying the effects of **conceptual pacts** (CPs) on RL in dialogue: CPs are patterns of RL which are mutually accepted (either explicitly or implicitly) and used by all dialogue participants throughout the course of a dialogue (Brennan and Clark, 1996).

In order to study this phenomenon, we introduce a collection of recorded spoken English dialogues situated in a task called *KTH Tangrams*, wherein two participants collaborate in order to correctly select a predetermined abstract image on a procedurally-generated game board: Participants take turns assuming the role of either **instructor**, who can see which piece must be selected, or **manipulator**, who can select a piece but cannot see which one must be selected. This experiment design is similar to that used for many other works regarding RL, the most similar of these being PentoRef's *PentoCV* and *RDG-Pento* (Zarrieß et al., 2016).

PentoCV and *RDG-Pento* consist of one participant instructing the other which pentomino piece (Golomb, 1994) is to be manipulated, but both participants are allowed to speak in a free fashion, a design originally defined by Kousidis et al. (2012). *KTH Tangrams*, however, is especially well-suited to observing CPs because the experiment design entails participants deterministically referring to abstract entities multiple times in a dynamic environment without the entities themselves playing a role in a larger, culminating goal as done by e.g. Foster et al. (2006).

2. Related Work

While there are many different works concerned with task-oriented dialogue, there are a number of differences in experiment design among them.

2.1. Static Versus Dynamic Environments

The roles of instructor and manipulator seen in many tasks used for dialogue research are analogous to the roles of director and matcher in traditional reference communication tasks, with the terms defined by Schober and Clark (1989) but the task itself originating from Krauss and Weinheimer (1964). These tasks involve simple reference resolution, whereby the state of the **environment** shared by the director and matcher (e.g. a set of figures on a sheet of paper) does not change during the task.

Static reference communication tasks often differ from instructor-manipulator tasks in that, in the latter, the state of the participants' shared environment changes during the task, entailing that CPs be robust throughout these changes, as observed by e.g. Ibarra and Tanenhaus (2016). Since the referent of a(n effective) CP should remain unambiguous throughout the dialogue for all members of the CP, a dynamic environment would more easily show the difference of CPs from mere alignment of RL.

2.2. Repeating Versus Culminating Tasks

Certain tasks are **repetitive** in that a similar sub-task is repeated with parametric variations, such as done by Krauss and Weinheimer (1964). However, a number of works involve tasks which **culminate** to a predefined goal — cf. Foster et al. (2006). This means that participants are aware of a sub-task's relation to a larger process, which has an effect on RL used and thus also CPs (Ibarra and Tanenhaus, 2016). While these effects are interesting, we are interested in CPs based on properties of the CPs' referents in themselves rather than on referents' purpose in a larger pattern of interaction: Resolving CPs based on "object-oriented names" such as *the leg* [*of the lion being assembled*] (Ibarra and Tanenhaus, 2016, p. 564) is a context-sensitive task which is not only dependent on the previous language used but also on the history of the culminating task as well as future actions and thus entails action awareness, such as by incorporating intent prediction and decision planning — cf. Bard et al. (2008). Thus, we want to limit participants' accumulation of task-related knowledge over time.

2.3. Referential Aspects

In many tasks, such as that of Krauss and Weinheimer (1964), participants can freely chose referents, e.g. which entity to describe. This complicates both manual and automatic annotation of referents and RL and so an ideal experiment should restrict possible referents as much as possible without hindering free dialogue. Likewise, we are interested in CP formation between humans and so the experiment should avoid machine-directed speech, which can differ greatly from human-directed speech (Kriz et al., 2010). Lastly, referent entities should have distinguishing features (Westerbeek et al., 2015) but not show extreme **typicality**, whereby referent features are strongly correlated: For example, a *purple cow* is highly atypical (Mitchell et al., 2013, p. 3062).

2.4. Experimental Paradigms

There exist multiple experimental paradigms for task-oriented dialogue, each incorporating different combinations of environmental, task and referential aspects.

2.4.1. Map Tasks

One form of instructor-manipulator task is that of "map tasks", whereby one participant has information about a spatial area which the other does not. The former must then instruct the latter on how to navigate the map to accomplish a defined goal, e.g. reaching a particular landmark (Thompson et al., 1993; MacMahon et al., 2006). A variation of this are cases where the navigator is in fact situated within the map being navigated (Shimizu and Haas, 2009; Vogel and Jurafsky, 2010; Götze and Boye, 2016). In both cases, the state of the environment is static. However, the task culminates to a predefined goal, leading to confounds.

2.4.2. Joint Construction Tasks

One experiment design involving dynamic environments is that of "joint construction tasks" (Fong et al., 2006; Foster et al., 2006; Spanger et al., 2012; Yan et al., 2016), where agents (human or otherwise) collaboratively assemble a predefined structure from component pieces. This dynamism makes such tasks well-suited for studying the formation of CPs: Due to the fact that certain physical features are static (e.g. a piece's shape or color) while others are dynamic and change throughout the course of the dialogue (e.g. location), the dynamic nature of RL can be better studied, similarly to how Ibarra and Tanenhaus (2016) observed changes in referring strategy when contrastive features previously used to disambiguate entities are no longer effective due to introducing new entities with similar features. However, these tasks culminate to an end goal, again leading to e.g. "object-oriented names" such as *the leg* [*of the lion being assembled*] (Ibarra and Tanenhaus, 2016, p. 564).

2.4.3. *KTH Tangrams*: Dynamic, Repeating Fixed-Referent Tasks

We have argued that a corpus ideal for researching CPs involves a repeating, non-culminating task in a dynamic environment while lacking free choice of referent. Moreover, the referents themselves should be abstract enough to elicit descriptive RL. However, in order to capture the full variation of CP formation, the language used should still be relatively unrestricted human-human dialogue; Unlike the datasets reviewed above, our corpus *KTH Tangrams* fulfills all of these criteria (see Table 1).

3. Experiment Design

Each experiment session involves two healthy adults with normal or corrected-to-normal vision and English either as a native language or as a common language used in a professional context. Each participant has their own PC on a LAN, head-mounted microphone and speakers in a room separate from the other's, similarly to the setup of Manuvinakurike et al. (2015): They communicate freely via speech but cannot interact in any other way. Once both participants log into the game, they are simultaneously presented with an identical view of a simulated game board occupied by 20 tangram-like pieces (Gardner, 1974).

3.1. Reproducible Pseudo-Random Environments

The board configuration is determined procedurally: The pieces' initial placements are chosen pseudo-randomly with a seed as positions the board on an invisible 20×20 grid.[1]. Likewise, the pieces' visual attributes are chosen pseudo-randomly using the same method as is each piece's subsequent move[2].

- POSITIONX and POSITIONY are the position of the entity's center as a proportion of the total board area.
- HUE is derived from the individual sRGB color features RED, GREEN and BLUE (International Electrotechnical Commission, 1999).
- EDGECOUNT values are manually annotated for each unique SHAPE value; For the shapes currently present in the corpus, the values thereof range from 6 to 16.
- SHAPE is a nominal feature enumerating 17 unique images which can be drawn to visualize an entity. The images, which are shown in Figure 1, were hand-chosen to have a roughly-even distribution of typicality — cf. Mitchell et al. (2013).
- SIZE values are derived from possible entity dimensions 2×2 (small), 3×3 (medium) or 4×4 (large) and are normalized by the total area of the board; Since the board area is always 20×20, the effective feature values are 0.01, 0.0225 and 0.04.

Since the environments each dialogue is situated in are procedurally-generated, a wide distribution of behavior can be easily created which compensates for possible confounds, such as would be the case if e.g. in every dialogue session, there was a particular piece with a color and shape combination which would have effects on every dialogue

[1] Although the coordinates are not indicated visually, they are still occasionally used by the participants because two or more pieces may randomly line up in rows or columns during the game.

[2] Random values are generated using a 48-bit seed which is modified using a linear congruential formula (Knuth, 1981, 9–25) from the Java class library (Oracle Corporation, 2015)

Experiment	Environment	Task	Referent	Entity type	Addressee	Language
Krauss and Weinheimer (1964)	Static	Repeating	Free	Illustration	Human	Dialogue
Schober and Clark (1989)	Static	Repeating	Free	Tangram	Human	Dialogue
Thompson et al. (1993)	Static	Culminating	Free	Landmark	Human	Dialogue
Barr and Keysar (2002)	Dynamic	Culminating	Free	Diverse	Human	Dialogue
Foster et al. (2006)	Dynamic	Culminating	Free	Diverse	Machine	Dialogue
MacMahon et al. (2006)	Static	Culminating	Free	Landmark	Human	Dialogue
REX-J (Spanger et al., 2012)	Dynamic	Culminating	Free	Tangram	Human	Dialogue
SpaceRef Götze and Boye (2016)	Static	Repeating	Free	Landmark	Machine	Monologue
Ibarra and Tanenhaus (2016) ex. 1	Dynamic	Culminating	Free	Bloco^{TM}	Human	Dialogue
Ibarra and Tanenhaus (2016) ex. 2	Dynamic	Culminating	Free	Tangram	Human	Dialogue
PentoRef *WOz Pento*	Static	Repeating	Fixed	Pentomino	Machine	Monologue
PentoRef *Take*	Static	Repeating	Free	Pentomino	Machine	Monologue
PentoRef *Take-CV*	Static	Repeating	Fixed	Pentomino	Human	Monologue
PentoRef *Noise/No-noise*	Dynamic	Culminating	Free	Pentomino	Human	Dialogue
PentoRef *Pento-CV*	Dynamic	Culminating	Free	Pentomino	Human	Dialogue
PentoRef *RDG-Pento*	Static	Repeating	Free	Pentomino	Human	Dialogue
KTH Tangrams	**Dynamic**	**Repeating**	**Fixed**	**Tangram**	**Human**	**Dialogue**

Table 1: A comparison of experimental paradigms in task-oriented dialogue.

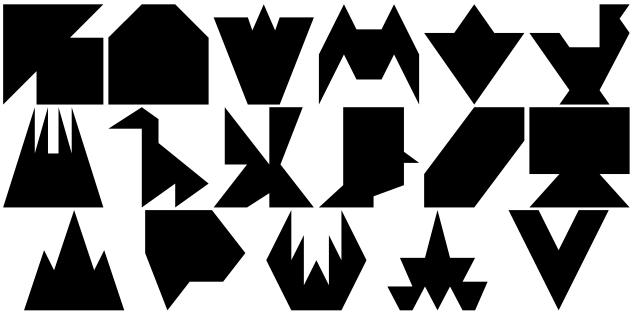


Figure 1: The possible shapes of generated game pieces.

in the corpus either as a distractor or as the piece being referred to itself. Furthermore, since these environmental features are generated using a seeded pseudo-random number generator, any particular experiment can be reproduced at will.

3.2. Task Description

During the task, both dialogue participants are seated at their own computer in separate rooms, each of which displays the current state of the game (see Figure 2). In each game **round**, the instructor sees a piece randomly highlighted, which is the piece they must instruct the manipulator to select. The manipulator has no indication or prior knowledge of which piece is to be selected, so the instructor must describe the piece well enough for the selector to click on it using a mouse. If the piece is selected correctly, the participants gain one point and proceed to the next round, where the roles are switched and the previously-selected piece moves to a random place on the board. However, if the wrong piece is selected, they lose two points and are required to try again (see Figure 3).

Each experiment session is intended to be 15 minutes long[3] and the participants are informed of this before starting, being encouraged to earn as many points as possible in this time. They are explicitly told that they are not restricted in any way regarding their language aside from the one restriction that they focus only on the task at hand.

[3]The mean duration for the corpus is 15:25.38 minutes.

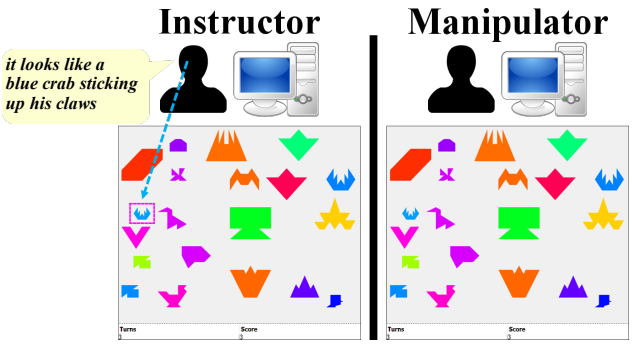


Figure 2: The game board as seen by the respective roles.

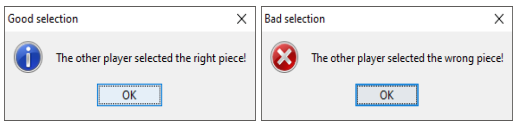


Figure 3: Feedback for correct and incorrect selections.

In addition to the participants' speech being recorded and transcribed, the state of the game at the time of each utterance is available, including features representing each piece (i.e. possible referent) on the board at any time.

4. Dialogue Transcription

Recordings are manually segmented and transcribed into two channels of **utterances** composed of tokens $u \triangleq \langle t_1 \dots t_n \rangle$, one for each participant. An utterance is

Time	Speaker role	Dialogue utterances
7:27.97	Instructor	[*uh this is a new one*]u_{732}
	Manipulator	
7:28.12	Instructor	[*right hand side it's a V big V with a top sticking out of it*]u_{733}
	Manipulator	[*mm-hmm*]u_{734} [*with a pointy*]u_{735}

Table 2: Example of transcription where overlapping speech does not affect segmentation.

Time	Speaker role	Dialogue utterances
2:58.07	Manipulator	[*uh is it the that one or is it not that one* LAUGHTER LAUGHTER]u_{95}
	Instructor	[*the*]u_{96} [*y- yeah so the* LAUGHTER]u_{97} [*the same yellow the*]u_{98}

Table 3: Example of transcription where overlapping speech and disfluencies affect segmentation.

defined as a minimal span of uninterrupted language which denotes a dialogue act in the scope of the task at hand. Disfluencies and self-repair delimit segmentation boundaries only if there is a significant period of silence after the potential boundary or if the other participant takes a dialogue turn, leading the participant to respond to the other's speech act, as shown in Tables 2 and 3 (Schegloff, 2000). An overview of the entire corpus is shown in Table 4.

	Minutes	Rnds.	Utts.	Tokens	Toks./ utt.
Min	09:42.$\overline{5}$	30	151	858	3.1
Max	17:49.1	138	625	2592	8.6
Mean	15:25.1	78.3	355.8	1616.3	4.7
Sum	647:35.2	3288	14942	67884	198.8

Table 4: Overview of 42 recorded sessions.

5. Analysis

Two different lexical analyses were performed in order to evaluate the appropriateness of the corpus for research in dialogic alignment of RL and conceptual pacts: Firstly, a trend of lexical convergence was observed both within speakers (i.e. a single participant's use of RL becomes less varied with time) as well as between speakers in a single dyad, whereby the RL used by one participant becomes more similar to their partner's RL. Secondly, TF-IDF scores were used to estimate the amount of information contained by language for resolving referents in a given dialogue on a global scale, i.e. not considering dialogue context.

5.1. Dialogic Convergence

Three types of lexical alignment were calculated in order to illustrate a trend of convergence in language use within dyads:

Within-speaker convergence shows how an individual participant's use of RL becomes more consistent throughout the course of the dialogue.

Between-speaker convergence shows how the use of RL by both participants in a dyad converges on the other's; Comparing this with within-speaker convergence allows effects of dialogic lexical alignment to be discerned from any effects associated with a particular participant (Krauss and Weinheimer, 1964).

General convergence shows how much language used to refer to an entity with a given set of features converges as dialogue progresses for the entire corpus; This can be used to control for general convergence effects in discourse (Carroll, 1980; Clark and Wilkes-Gibbs, 1986).

Convergence was measured using **token type overlap**, the number of token types (i.e. unique words) which overlap with the preceding coreference for a given referent r:

$$\Delta c_n^r \triangleq \frac{c_n'^r \cap c_{n-1}'^r}{c_n'^r \cup c_{n-1}'^r} \quad (1)$$

where $c' \triangleq \{t \in \mathcal{T} \mid t \in c\}$ is the set of all unique tokens (i.e. types) $t \in \mathcal{T}$ in a coreference c. This is similar to Aina et al. (2017)'s "lexical alignment" metric but considers only the preceding coreference c_{n-1}^r rather than all $C_{n'<n}^r$. Thus, token type overlap is relatively better-suited to measuring CP formation because CPs entail similar language in each RE rather than simply over the entire coreference chain (Brennan and Clark, 1996).

Rather than manually annotating REs within utterances as done by Aina et al. (2017) and Zarrieß et al. (2016), the metrics were calculated for all tokens in the utterances in a given game round, considering all language produced during the round refer to the piece which must be selected in that round $\hat{r}$. This introduces noise but also facilitates faster data collection and also simulates real-world scenarios, in which RE detection is non-trivial.

Moreover, convergence can be calculated not only for language used to refer to a unique entity (i.e. each of 20 possible referents in a session) but also for individual features, as done in this paper with the categorical feature SHAPE. In other words, not only can RL convergence be measured for individual referents but also for features of said referents, which are thus generalizable to other entities with similar features regardless if they have previously been referred to in discourse or not.

5.1.1. Preprocessing

For evaluation, all utterances from the instructor in a given game round were concatenated in order to create the sets of token types representing a coreference c'.

Time	Speaker role	Utterance
3:45.80	Instructor	*this one looks like um like a crown and it's*
3:52.76	Manipulator	*what color*
3:53.78	Instructor	*pink like*

Table 5: RE expansion across discontinuous utterances.

Before concatenation, the following tokens were removed from each utterance:

Metalanguage such as COUGH and LAUGHTER

Disfluencies such as *l-* in *big block l- top left*

Fillers such as *um* and *uh* in *um blue uh kind of a temple*

Duplicate tokens such as the second *a* in *it's a a blue mountain*

Utterances were concatenated in this way in order to mitigate effects of utterance segmentation on token type overlap: For example, there is in fact no overlap of the individual instructor utterances in Table 5 despite the following utterance *pink like* could be seen as an expansion of the RE initiated in the preceding utterance from the same speaker. Therefore, despite being separate "utterances" for the sake of transcription, they comprise a single referring unit. Likewise, comparing the overlap of the expansion *pink* like with its immediate predecessor *what color* for between-speaker convergence is not ideal because *what color* is not a proper RE but rather a request for expansion of the initiated RE.
Secondly, semantically-weak tokens such as *this one looks like* introduce noise which must be addressed: the token sequences $\langle it, 's, a, blue, bird\rangle$ and $\langle blue, bird\rangle$ would have an overlap of only 0.40 despite having total overlap in the most-relevant words, *blue* and *bird*. Concatenating utterances from the same speaker mitigates this by reducing the amount of comparisons made overall: The two previous sequences would only be compared if they appeared in separate game rounds for the same referent or — in the case of calculating between-speaker convergence — if the other participant referred to the same entity in the role of instructor between the two utterances.
Deriving the metric in this manner resulted in a set of 7,818 individual instructor utterances, which was then reduced to 3,288 unified coreferences for individual rounds excluding those comprised solely tokens filtered out in preprocessing.

5.1.2. Results

A strong effect of within-speaker convergence (WITHIN) effects as well as between speakers (BETWEEN) was found when measuring token type overlap for coreference chains referring to a specific entity $c_1^r \dots c_n^r$ (see Figure 4). Additionally, there was a weak but very significant inverse relationship of coreference sequence order and token type overlap in GENERAL convergence (see Table 6): This suggests that individual participants' usage of RL converges not only on itself but also on that of their dyad partner's, indicating the formation of CPs specific to that dyad.

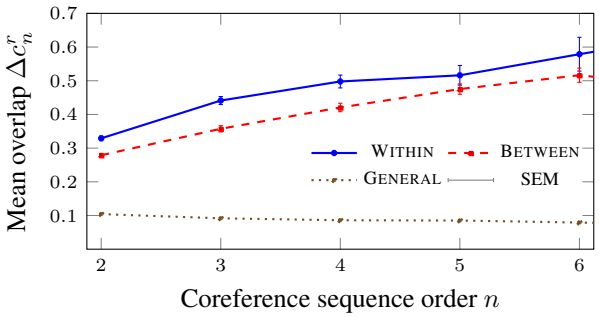


Figure 4: Instructor token type overlap for rounds referring to a unique entity r for the n^{th} time in a game.

$corr(n, \Delta c_n^r)$	WITHIN	BETWEEN	GENERAL
$\lvert S\rvert$	1846	2261	1515892
2-tailed Pearson correlation coefficient r			
Correlation	0.2803	0.3344	−0.0979
$CI_{lower}^{\alpha=0.01}$	0.2242	0.2854	−0.1000
$CI_{upper}^{\alpha=0.01}$	0.3346	0.3817	−0.0958
2-tailed Spearman's rank correlation coefficient ρ			
Correlation	0.2725	0.3147	−0.1358
$CI_{lower}^{\alpha=0.01}$	0.2161	0.2651	−0.1379
$CI_{upper}^{\alpha=0.01}$	0.3270	0.3627	−0.1338

Table 6: Significance of the correlation between coreference sequence order n (the n^{th} time a round in a game refers to a unique entity r) and instructor token type overlap Δc_n^r.

A similar relationship between strong within-speaker and slightly weaker between-speaker convergence was seen when analyzing RL referring to specific features rather than entities themselves, i.e. language referring to all entities with a given SHAPE $C^s \triangleq \{c \in C \mid \text{SHAPE}(c) = s\}$ (see Figure 5). Analogously to when measuring overlap of "true" coreference chains for individual entities, there was a weak but very significant inverse relationship of coreference sequence order and token type overlap in GENERAL convergence (see Table 7): This suggests that RL and CPs not only are formed for individual referents but are at least partially generalizable to new referents which share features of previous referents, which warrants further analysis of alignment and CP negotiation in this experimental paradigm.

5.2. Information Content of RL

Finally, we evaluated RL based on how specific it is to the referent r, which the dialogue participants are to move in a given game round: This is done as an estimation of the amount of information contained by a particular set of language in the task of resolving the referent. When formulated in this way, the task of reference resolution can be envisaged as an information retrieval task; For this reason, we calculated the TF-IDF scores (Spärck Jones, 1972) for each trigram of tokens from each utterance of language for both participants in each dialogue $g_i \triangleq \langle t_{i-2}, t_{i-1}, t_i\rangle$ where $c \triangleq \langle t_1 \dots t_n\rangle$ and treated each unique referent in the corpus $r \in R$ as a "document", where $|R| = 840$

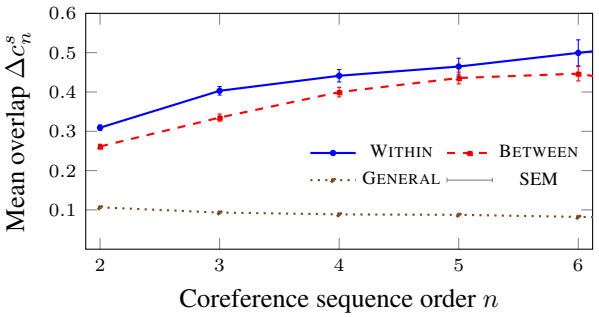


Figure 5: Instructor token type overlap for rounds referring to an entity with a unique SHAPE value s for the n^{th} time in a game.

$corr(n, \Delta c_n^s)$	WITHIN	BETWEEN	GENERAL
$\lvert S \rvert$	1989	2153	1245995
2-tailed Pearson correlation coefficient r			
Correlation	0.2471	0.2972	−0.0981
$CI_{lower}^{\alpha=0.01}$	0.1921	0.2458	−0.1004
$CI_{upper}^{\alpha=0.01}$	0.3005	0.3470	−0.0958
2-tailed Spearman's rank correlation coefficient ρ			
Correlation	0.2459	0.2792	−0.1347
$CI_{lower}^{\alpha=0.01}$	0.1909	0.2273	−0.1369
$CI_{upper}^{\alpha=0.01}$	0.2994	0.3296	−0.1324

Table 7: Significance of the correlation between coreference sequence order n (the n^{th} time a round in a game refers to an entity with a unique SHAPE value s) and instructor token type overlap Δc_n^s.

for $|D| = 42$ dyads with 20 referents per dyad:

$$\begin{aligned} tfidf(g, r, R) &\triangleq tf(g, r) \cdot idf(g, R) \\ tf(g, r) &\triangleq f_{g,r} \\ idf(g, R) &\triangleq \log \frac{|R|}{|\{r \in R \mid f_{g,r} > 0\}|} \end{aligned} \tag{2}$$

However, in order to encode the knowledge that RL converges in dialogue (see Section 5.1.), the TF-IDF score is normalized by the total number of coreferences of r $|C^r|$:

$$\begin{aligned} tfidf_\alpha(g, r, R) &\triangleq tfidf(g, r, R) \cdot \alpha^r \\ \alpha^r &\triangleq 1 + \log|C^r| \end{aligned} \tag{3}$$

The expression $\alpha^r \triangleq 1 + \log|C^r|$ encodes the assumption that, as the amount of coreferences $|C^r|$ increases, so should the specificity of RL used for r. Trigrams were constructed from each individual utterance in a dialogue $u^r \in U^r$ after applying the token-filtering methods mentioned in Section 5.1.1.. Using this metric to rank trigrams resulted in semantically rich language which is also used repeatedly by participants throughout the course of dialogue — Figure 6 illustrates the 20 referents with the highest-scoring trigrams:

$$\underset{r \in R, g^r \in C^r}{\arg\max} \; tfidf_\alpha(g, r, R) \tag{4}$$

The illustrated examples suggest that this metric is an effective post-hoc measure of the potential "referentiality" of language given a known referent and it suggests that there are rich, varied usage of RL in this corpus which comprise CPs: Not only is there observable variation of highly-specific RL (i.e. RL with a high $tfidf$ score) even for similar referents (e.g. *the diamond* vs. *slanted rectangle*) but there is also a high intra-document frequency tf of each of them. Moreover, this metric is purely linguistic and does not account for the features of the referents themselves and inter-referent similarities; it is possible that incorporating this knowledge may yet further increase the discriminative power of this metric.

d	r	$\lvert C^r \rvert$	Trigram g	$tfidf$	tf
6		15	*'s the robot*	33.64	5
			robot with a	30.18	5
6		9	*the red V*	30.72	6
			's the red	22.13	5
9		7	*the blue rooster*	33.64	5
			blue rooster again	20.19	3
9		7	*the pink bat*	33.64	5
			the nice one	13.46	2
9		8	*the yellow mountain*	32.05	6
			towards the bottom	14.35	3
14		6	*the red head*	37.40	7
			's the red	22.13	5
20		10	*the yellow map*	40.37	6
			map on the	13.46	2
22		11	*the blue bird*	29.71	7
			ah the blue	12.07	2
23		9	*purple V with*	47.10	7
			triangle in the	24.68	5
24		7	*the large V*	40.37	6
			is the large	13.46	2
27		9	*up and down*	39.41	7
			and down triangle	13.46	2
27		7	*light blue TV*	36.21	6
			big light blue	26.71	5
28		7	*'s the diamond*	36.21	6
			the diamond orange	13.46	2
28		8	*'s the bite*	33.64	5
			the bite mark	26.91	4
31		7	*a peak in*	36.21	6
			with a peak	28.70	6
31		7	*the yellow house*	33.64	5
			's the yellow	7.47	2
34		7	*small blue TV*	33.78	6
			the small blue	16.36	4
34		9	*lots of triangles*	30.18	5
			with lots of	24.14	4
38		6	*slanted rectangle with*	47.10	7
			rectangle with two	42.74	8
39		4	*yellow and green*	47.10	7
			it 's lighter	33.64	5

Figure 6: TF-IDF scores of language when considering a given unique referent r in a dyad d as a document. $|C^r|$ is the number of coreferences of r in a game.

6. Conclusion

KTH Tangrams is a corpus of high-quality task-oriented dialogue featuring observable convergence between participants in their use of referring language throughout the course of the dialogues they participate in. This indicates that the task's dynamic yet repeating nature combined with the abstractness of tangram figures lends itself not only to the study of referring language in general but also in the development of conceptual pacts for reference which are individual to a particular dialogue.

In future works, we intend to use this dataset to explore the automatic understanding and generation of CPs in a dynamic context (i.e. for unseen dialogues); We encourage others interested in RL and CPs to take advantage of and improve this corpus as well in order to establish a common corpus for comparable studies in referring language and conceptual pacts.

7. Release

The linguistic transcriptions and environmental data will be made available under the Open Data Commons Attribution License v1.0 (Open Data Commons, 2010) as part of the forthcoming data bank *Språkbanken Tal* (Edlund, 2017), associated with the SWE-CLARIN[4] initiative *Språkbanken*, the Swedish Language Bank[5] (Hinrichs and Krauwer, 2014; Borin and Domeij, 2014); See `http://sprakbanken.speech.kth.se/data/kth-tangrams`.

8. Acknowledgments

This work is supported by the SSF (Swedish Foundation for Strategic Research) project COIN. Correlation coefficient significance testing was performed with R version *3.4.1 x86_64* (R Core Team, 2015) and *psych* v1.7.8 (Revelle (2017); Hollander and Wolfe (1973, 185–194); Best and Roberts (1975)). Plots were made with the LaTeX package *pgfplots* v1.13 (Feuersänger, 2016).

The authors would like to thank Jens Edlund for offering *KTH Tangrams* as one of the first datasets available through *Språkbanken Tal*.

9. Bibliographical References

Aina, L., Philippova, N., Vogelmann, V., and Fernández, R. (2017). Referring expressions and communicative success in task-oriented dialogues. In Volha Petukhova et al., editors, *Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue*, pages 8–16, Saarbrücken, Germany, August.

Bard, E. G., Hill, R., and Foster, M. E. (2008). What tunes accessibility of referring expressions in task-related dialogue? In Zygmunt Pizlo, et al., editors, *Proceedings of the 30th Annual Meeting of the Cognitive Science Society*, pages 945–950, Austin, TX, USA, July. Cognitive Science Society.

Barr, D. J. and Keysar, B. (2002). Anchoring comprehension in linguistic precedents. *Journal of Memory and Language*, 46(2):391–418, February.

Best, D. J. and Roberts, D. E. (1975). Algorithm as 89: The upper tail probabilities of spearman's *Rho*. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(3):377–379.

Borin, L. and Domeij, R. (2014). Språkteknologi och språkresurser för språken i Sverige: En statusrapport. *Språk i Norden*, pages 33–47.

Brennan, S. E. and Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6):1482–1493, November.

Nicoletta Calzolari, et al., editors. (2016). *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May. European Language Resources Association (ELRA).

Carroll, J. M. (1980). Naming and describing in social communication. *Language and Speech*, 23(4):309–322.

Clark, H. H. and Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, 22(1):1–39, February.

Edlund, J. (2017). Skapandet av grunden för en svensk talbank. Technical Report D2016-0240, KTH Speech, Music and Hearing, Stockholm, Sweden, March.

Feuersänger, C., (2016). *Manual for Package* PGFPLOTS*: 2D/3D Plots in LaTeX, Version 1.13*, January.

Fong, T., Kunz, C., Hiatt, L. M., and Bugajska, M. (2006). The human-robot interaction operating system. In *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-robot Interaction*, HRI '06, pages 41–48, New York, NY, USA. ACM.

Foster, M. E., By, T., Rickert, M., and Knoll, A. (2006). Human-robot dialogue for joint construction tasks. In *Proceedings of the 8th International Conference on Multimodal Interfaces*, ICMI '06, pages 68–71, New York, NY, USA. ACM.

Gardner, M. (1974). Mathematical games: On the fanciful history and the creative challenges of the puzzle game of tangrams. *Scientific American*, 231(2):98–103B.

Golomb, S. W. (1994). *Polyominoes: Puzzles, Patterns, Problems, and Packings*. Princeton University Press, Princeton, NJ, USA, 2nd edition.

Götze, J. and Boye, J. (2016). SpaceRef: A corpus of street-level geographic descriptions. In Calzolari et al. (Calzolari et al., 2016).

Hinrichs, E. and Krauwer, S. (2014). The CLARIN research infrastructure: Resources and tools for ehumanities scholars. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Hollander, M. and Wolfe, D. A. (1973). *Nonparametric Statistical Methods*. John Wiley & Sons, New York, NY, USA.

Ibarra, A. and Tanenhaus, M. K. (2016). The flexibility of conceptual pacts: Referring expressions dynamically shift to accommodate new conceptualizations. *Frontiers in Psychology*, 7:561–574.

International Electrotechnical Commission. (1999). Mul-

[4] `https://sweclarin.se/`

[5] `https://spraakbanken.gu.se/`

timedia systems and equipment — Colour measurement and management — Part 2-1: Colour management — Default RGB colour space — sRGB. International Standard IEC 61966-2-1:1999, International Electrotechnical Commission, Geneva, Switzerland.

Knuth, D. E. (1981). *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Addison-Wesley, Reading, MA, USA, 2nd edition.

Kousidis, S., Pfeiffer, T., Malisz, Z., Wagner, P., and Schlangen, D. (2012). Evaluating a minimally invasive laboratory architecture for recording multimodal conversational data. In *Proceedings of the Interdisciplinary Workshop on Feedback Behaviors in Dialog*, pages 39–42, Stevenson, WA, USA, September.

Krauss, R. M. and Weinheimer, S. (1964). Changes in reference phrases as a function of frequency of usage in social interaction: a preliminary study. *Psychonomic Science*, 1(1):113–114, January.

Kriz, S., Anderson, G., and Trafton, J. G. (2010). Robot-directed speech: Using language to assess first-time users' conceptualizations of a robot. In *2010 5th ACM/ IEEE International Conference on Human-Robot Interaction (HRI)*, pages 267–274, Osaka, Japan, March.

MacMahon, M., Stankiewicz, B., and Kuipers, B. (2006). Walk the talk: Connecting language, knowledge, and action in route instructions. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference (AAAI-06)*, pages 1475–1482, Menlo Park, CA, USA, July. AAAI Press.

Manuvinakurike, R., Paetzel, M., and DeVault, D. (2015). Reducing the cost of dialogue system training and evaluation with online, crowd-sourced dialogue data collection. In Christine Howes et al., editors, *Proceedings of the 19th Workshop on the Semantics and Pragmatics of Dialogue*, pages 113–121, Gothenburg, Sweden, August.

Mitchell, M., Reiter, E., and van Deemter, K. (2013). Typicality and object reference. In Markus Knauff, et al., editors, *Proceedings of the 35th Annual Meeting of the Cognitive Science Society*, Austin, TX, USA, July. Cognitive Science Society.

Open Data Commons. (2010). Open Data Commons Attribution License (ODC-By) v1.0. `https://opendatacommons.org/licenses/by/1.0/`. Last accessed on 12 February 2018.

Oracle Corporation. (2015). Java™ SE Development Kit 8, update 45 (JDK 8u45). `http://www.oracle.com/technetwork/java/javase/8u45-relnotes-2494160.html`. Last accessed on 12 February 2018.

R Core Team, (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Revelle, W., (2017). *psych: Procedures for Psychological, Psychometric, and Personality Research*. Northwestern University, Evanston, Illinois, USA. R package version 1.7.8.

Schegloff, E. A. (2000). Overlapping talk and the organization of turn-taking for conversation. *Language in Society*, 29(1):1–63.

Schober, M. F. and Clark, H. H. (1989). Understanding by addressees and overhearers. *Cognitive Psychology*, 21(2):211–232.

Shimizu, N. and Haas, A. (2009). Learning to follow navigational route instructions. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 1488–1493, Pasadena, CA, USA. IJCAI Organization.

Spanger, P., Yasuhara, M., Iida, R., Tokunaga, T., Terai, A., and Kuriyama, N. (2012). REX-J: Japanese referring expression corpus of situated dialogs. *Language Resources and Evaluation*, 46(3):461–491, September.

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Thompson, H. S., Anderson, A., Bard, E. G., Doherty-Sneddon, G., Newlands, A., and Sotillo, C. (1993). The HCRC map task corpus: Natural dialogue for speech recognition. In *Proceedings of the Workshop on Human Language Technology*, HLT '93, pages 25–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vogel, A. and Jurafsky, D. (2010). Learning to follow navigational directions. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 806–814, Uppsala, Sweden, July. Association for Computational Linguistics.

Westerbeek, H., Koolen, R., and Maes, A. (2015). Stored object knowledge and the production of referring expressions: the case of color typicality. *Frontiers in Psychology*, 6:1–12.

Yan, P., He, B., Zhang, L., and Zhang, J. (2016). Task execution based-on human-robot dialogue and deictic gestures. In *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1918–1923, Qingdao, China, December. IEEE.

Zarrieß, S., Hough, J., Kennington, C., Manuvinakurike, R., DeVault, D., Fernández, R., and Schlangen, D. (2016). PentoRef: A corpus of spoken references in task-oriented dialogues. In Calzolari et al. (Calzolari et al., 2016).

On the Vector Representation of Utterances in Dialogue Context

Louisa Pragst, Niklas Rach, Wolfgang Minker, Stefan Ultes
Ulm University, Ulm University, Ulm University, Cambridge University
Ulm, Germany, Ulm, Germany, Ulm, Germany, Cambridge, UK
louisa.pragst@uni-ulm.de, niklas.rach@uni-ulm.de,wolfgang.minker@uni-ulm.de, su259@cam.ac.uk

Abstract

In recent years, the representation of words as vectors in a vector space, also known as word embeddings, has achieved a high degree of attention in the research community and the benefits of such a representation can be seen in the numerous applications that utilise it. In this work, we introduce dialogue vector models, a new language resource that represents dialogue utterances in vector space and captures the semantic meaning of those utterances in the dialogue context. We examine how the word vector approach can be applied to utterances in a dialogue to generate a meaningful representation of them in vector space. Utilising existing dialogue corpora and word vector models, we create dialogue vector models and show that they capture relevant semantic information by comparing them to manually annotated dialogue acts. Furthermore, we discuss potential areas of application for dialogue vector models, such as dialogue act annotation, learning of dialogue strategies, intent detection and paraphrasing.

Keywords: sentence embedding, sentence representation, dialogue embedding, dialogue representation, dialogue act detection

1. Introduction

The representation of words in vector space (Mikolov et al., 2013a), also known as word2vec, has been widely successful in the research community. It has been utilised in a number of research areas, such as machine translation (Mikolov et al., 2013b), sentiment classification (Xue et al., 2014; Zhang et al., 2015), named entity recognition (Sienčnik, 2015) and document classification (Kusner et al., 2015), among many others. Considering the success and flexibility with which word vector models can be employed in numerous applications, we aspire to investigate whether applying the word2vec approach to dialogue utterances yields similar results.

In this paper, we introduce a new language resource: dialogue vector models (DVMs), a representation of utterances in vector space that takes into account dialogue context. We investigate what adaptations are needed to the word2vec approach in order to generate such models and evaluate whether they capture information that is beneficial in a dialogue context. Furthermore, we highlight several promising areas of application that might benefit from the use of DVMs.

In the following, we discuss related work in Section 2, before introducing our approach to the generation of DVMs in Section 3. Here, we also evaluate the validity of the generated DVMs. In Section 4, we propose potential applications for those models. Finally, we draw our conclusion in Section 5.

2. Related Work

A number of different approaches to the representation of sentences in vector space have been proposed, e.g. utilising recurrent neural networks (Sutskever et al., 2014; Palangi et al., 2016), convolutional neural networks (Shen et al., 2014; Kalchbrenner et al., 2014; Hu et al., 2014) and autoencoders (Socher et al., 2011). Those approaches typically do not take into account surrounding sentences for the generation of the sentence vector, instead relying on the words in the sentence only.

Tsunoo et al. (2017) implement sentence vectors with recurrent neural networks and additionally use a bidirectional Long Short-Term Memory to capture the impact of adjacent sentences. However, this additional information is not utilised to improve the vector representation of the sentence, but to model a story transition.

Some machine translation approaches, such as (Zhang et al., 2014; Hermann and Blunsom, 2014), rely on mapping sentences in different languages into a joint vector space. Here, the correct mapping is determined taking into account not adjacent sentences, but corresponding sentences in another language.

Skip thought vectors (Kiros et al., 2015) are sentence embeddings that are generated in a similar manner as word vector representations, and therefore similar to the dialogue vector models we propose. Rather than using the words in the sentence itself as basis to create a vector representation, those vectors are generated taking into account surrounding sentences. However, this representation is trained on novels rather than dialogue. In our work, we focus specifically on dialogue and its peculiarities.

In the area of conversational response generation, neural network approaches are commonly utilised (e.g. (Sordoni et al., 2015)). Here, previous utterances in a conversation are used to generate a vector representation of the dialogue context that the response generation is based on. While the vector representation is based on adjacent sentences, a vector in such a model does not represent a singular utterance, but rather the entirety of the preceding utterances.

Cerisara et al. (2017) investigate the usability of word2vec representations for dialogue act recognition. Similarly to our work, their goal is to determine the function of an utterance in the dialogue context. In this endeavour, they use word vectors in combination with deep neural networks to determine the dialogue act of an utterance. However, the representation in vector space they utilise stays on the word level. They do not try to achieve a vector representation of the whole sentence in the dialogue context. Additionally, they target the correct assignment of dialogue labels to ut-

	SPAADIA	Switchboard
# utterances	6201	223606
# unique utterances	2903	146740
%< 5 time usage	0.97	0.99
# unique LC	2683	127539
%< 5 time usage	0.92	0.88

Table 1: Relevant data regarding the number of utterances in the employed corpora.

terances. This representation is less flexible in its potential applications than a vector representation. Lin et al. (2017) use word2vec models to implement a question answering system. However, they as well do not try to generate and exploit a vector space representation of dialogue contributions.

3. Dialogue Vector Models

A dialogue vector model is any representation of sentences as vectors that captures their semantic meaning in the dialogue context. We have performed the training and evaluation of DVMs on two dialogue corpora: the SPAADIA corpus (Leech and Weisser, 2013), which consists of task-oriented dialogues such as train travel booking, and the Switchboard corpus (Godfrey et al., 1992), which contains casual conversations on pre-specified topics.
In the following, we detail how we generate DVMs from the chosen corpora, before evaluating their capability to capture relevant semantic information. To determine the degree to which our models are generalisable, we not only evaluate their performance for one corpus, but also cross-validate the DVMs by training them on one corpus and testing them on the other.

3.1. Implementation

To generate DVMs, we adapt the generation of word representations in vector space (Mikolov et al., 2013a). The original word2vec trains its representations similarly to autoencoding. However, rather than training against the input word itself, word2vec trains words against their adjacent words in the input corpus, either using the word to predict its context or using the context to predict the word. If we consider each sentence as a single word, the same approach can easily be used to train a DVM. Therefore, existing word2vec implementations can be used and only the input needs to be modified in a manner that allows the implementation to recognise sentences as words. We employ the word2vec implementation of Deeplearning4j (Deeplearning4j Development Team, 2016) in our experiments. In the following, we describe how we modified the input text to obtain a vector representation of dialogue utterances.
The text-based approach assigns a unique identifier word to each sentence. Then, the dialogue corpus is rewritten by replacing each sentence with its identifier. Using this modified corpus as input to the original word2vec algorithm, a DVM can be trained. This approach comes with two disadvantages: first, in small corpora sentences might only occur rarely in exactly the same wording. Therefore, only little context information is available for each sentence. For reference, a word vector model, the Google News Corpus model (Mikolov et al., 2013), was trained on about 100 billion words to achieve word vectors for 3 million words. The dialogue corpora we employed in this work consist of 6201 and 223606 utterances respectively, as can be seen in Table 1. Out of those, 47/66% are unique utterances that will be assigned a dialogue vector. About 97/99% of the unique utterances are used less than five times in the dialogues, providing only little data to train the model on. The second disadvantage of this approach is its inability to generalise. Even slight alterations of a sentence, such as using synonyms or a different word sequence, leave the DVM unable to assign a dialogue vector to it if the sentence has not been encountered in this wording during the training of the model. In the case the employed dialogue corpora, this impacts the performance strongly: only 304 unique utterances from the SPAADIA corpus can be found in the Switchboard corpus, and vice versa only 109 utterances are present.
Both disadvantages of the text-based approach can be addressed by preprocessing the dialogue utterances, namely transforming the sentences into a word-based vector representation. This preprocessing applies a light clustering that groups sentences sharing the same words, thereby increasing their occurrence in the corpus data. Furthermore, common mathematical distance measures, such as the euclidean distance, can be applied to vectors to find a suitable representation for sentences that have not been encountered during training.
To implement the preprocessing, we utilise a pre-trained word vector model, the Google News Corpus model (Mikolov et al., 2013) . We obtain the word vector v_i of each word i in a sentence S and represent S as LC_S: the linear combination of its word vectors, as can be seen in Equation 1.

$$LC_S = \sum_{i \in S} v_i \quad . \tag{1}$$

A unique identifier is assigned for each linear combination (LC) and the input corpus is rewritten accordingly.
Using this representation, the percentage of utterances used less than five times can be reduced to 88/92%. Furthermore, a full mapping of utterances from one corpus to the other is achieved by replacing the a previously unencountered LC with an LC that was encountered during training and has the minimal distance from the unknown one.

3.2. Setup of the Evaluation

Projecting utterances into a vector space can only be beneficial for research in dialogue systems if it adequately captures the semantic interrelations between utterances. To ascertain that the DVM groups semantically similar dialogue contributions in close vicinity to each other, we compare a clustering based on the distances between utterances in the DVM to a clustering of those utterances based on manually assigned dialogue acts. As dialogue acts represent the meaning of an utterance in the context of the dialogue (Austin, 1962; Bunt, 1994), they are well suited as ground truth that the DVM should come close to. Hence, our evaluation comprises two steps:

1. Clustering a set of dialogue contributions based on their euclidean distance in the DVM using k-means

Test Corpus	DVM based on		Mean	SD
SPAADIA	SPAADIA	Text	0.91913	0.00126
		LC	0.91832	0.00121
	Switchboard	Text	0.50807	0.11462
		LC	0.90210	0.00404
Switchboard	Switchboard	Text	0.74648	0.00002
		LC	0.74620	0.00004
	SPAADIA	Text	0.20101	0.14877
		LC	0.73912	0.00009

Table 2: Mean and standard deviation of achieved accuracy values for different DVMs.

2. Determining the accuracy of the resulting clustering in representing manually assigned dialogue acts

Here, the accuracy A is calculated using the Rand index, defined as

$$A = \frac{TP + TN}{TP + FP + TN + FN} \quad , \tag{2}$$

where TP is the number of pairs in a cluster that share the same dialogue act, TN the number of pairs in different clusters that do not share a dialogue act, and FP and FN the number of wrongly assigned pairs, either for being clustered together but not sharing a dialogue act or for being in different clusters but having the same dialogue act.
We evaluate our approach on both text-based and LC-based DVMs. Furthermore, we employ two dialogue corpora, the SPAADIA corpus (Leech and Weisser, 2013) and the Switchboard corpus (Godfrey et al., 1992), both of which contain human-human conversation annotated manually with dialogue acts. We utilise those annotations as ground truth to which we can compare the results of our DVM. Both corpora are used for training DVMs as well as testing them. The cross-corpora evaluation ascertains the ability of DVMs to generalise. This setup results in eight test conditions to be evaluated.
For each test condition, we perform k-means clustering thirty times, with k being the number of dialogue labels used in the corpus. The resulting clusters are compared to the ground truth given by the manually annotated dialogue acts of the test data.

3.3. Results

Our evaluation shows that DVMs successfully capture dialogue relevant semantic information, and do so across corpora. Table 2 shows the mean and standard deviation of the achieved accuracy values. In the following, those results are discussed in more detail.
When testing is performed with the same corpus the models were trained on, all of them yield very good result, with over 0.91 accuracy for the SPAADIA corpus and over 0.74 for the Switchboard corpus. Furthermore, those result are achieved consistently, the standard deviation is extremely low in all cases. This suggests that the clusters are well separated and can be correctly determined by the clustering algorithm repeatedly.
When performing the evaluation on the same corpus that the DVM was trained with, the difference between DVMs trained on text and trained on LCs is minor. A slightly better performance can be reported for the text-based model for both corpora. This difference, although small, is statistically significant for the SPAADIA ($t(58) = -2.551, p = 0.013$) as well as the Switchboard corpus ($t(39.210) = -32.591, p = 0.000$). Considering only those results, employing LCs would seem unnecessary. However, their advantage of providing potential for generalisation becomes prominent when a DVM that was trained on one corpus is used for clustering utterances of the other corpus. Not only does the representation as vector allow every utterance, even if it was not present in training corpus, to be assigned a dialogue vector. The resulting representation also achieves accuracy values that are in close vicinity to the ones achieved when training and testing is performed on the same corpus, 0.9 for the SPAADIA corpus and 0.74 for the Switchboard corpus. The difference of about 0.02/0.01, although statistically significant (SPAADIA: $t(34.554) = 22.037, p = 0.000$, Switchboard: $t(31.586) = 446.239, p = 0.000$), is minor, especially considering that most of the utterances have not been encountered during training. In comparison, text-based DVMs only provide a dialogue vector for a small fraction of the test utterances and the accuracy values of text-based cross-corpus evaluation are statistically worse than those of other approaches (SPAADIA: $t(29.072) = -18.816, p = 0.000$, Switchboard: $t(29.000) = -19.812, p = 0.000$) by a large margin of about 0.4/0.5. Those results show that DVMs are able to generalise to a high degree if they are trained with LCs.
Considering the results of our evaluation, we believe that DVMs are a suitable representation of dialogue utterances and can capture the important dialogue information.
Our method of evaluation is related to the task of dialogue act recognition, which has been performed for the Switchboard corpus by Kalchbrenner and Blunsom (2013) and Cerisara et al. (2017), among others. Using supervised learning methods for dialogue classification, they achieve an accuracy 73.9% and 72.8% respectively. A good vector representation of dialogue utterances implicitly contains the information about dialogue acts. Therefore, the clusters constituted by DVMs achieve a comparable accuracy in grouping utterances according to the dialogue act. No supervision or even any particular training aimed at identifying dialogue acts was needed to achieve this result. Furthermore, DVMs can be applied more flexibly than pure dialogue act classifiers.

4. Potential Areas of Application

In the previous sections, we could show approaches to the generation of DVMs as well as the ability of those models to capture semantic interrelations between dialogue utterances. To complete our exploration of DVMs, we give an overview of research areas that we believe could benefit from utilising them in this section.

4.1. Dialogue Act Annotation

As our evaluation in Section 3.3. shows, DVMs excel as a resource for clustering algorithms to sort utterances with the same dialogue act into the same clusters. This implies

that bootstrapping manual dialogue act annotations by clustering utterances and manually assigning dialogue acts to clusters is a promising approach to reduce the work needed for dialogue corpus annotation.
In addition, automatic dialogue act recognition might be improved by the usage of DVMs. Comparing our results to current dialogue act classifiers (e.g. (Cerisara et al., 2017)) shows that a similar performance can be achieved by clustering based on DVMs. These findings support the potential of DVMs in this area.

4.2. Creation of Dialogue Policies

Machine learning approaches, in particular reinforcement learning, have become increasingly popular for training dialogue strategies in recent years (e.g. (Scheffler and Young, 2002; Rieser and Lemon, 2011)). Mathematical models form the basis of those approaches and require a numeric representation of their states. Our hypothesis is that using a meaningful representation such as DVMs as an input to those learning algorithms, rather than arbitrarily chosen ones, might be able to facilitate working with them.

4.3. Intent Detection and Paraphrasing

Indirect speech acts, as described by e.g. Searle (1975), are characterised by having, in addition to the first illocutionary act that is expressed directly by the utterance, a second one that is expressed only indirectly. Identifying the second illocutionary act and reacting accordingly is essential in a cooperative dialogue. Therefore, a lot of research goes into the automatic detection of user intent (e.g. (Allen and Perrault, 1980; Briggs and Scheutz, 2013)).
We believe that DVMs can facilitate those efforts. They project utterances that fulfil the same function in a dialogue in close vicinity to each other. This can be used to identify potential candidates for indirect speech acts as well as a corresponding direct utterances that reveal the secondary illocutionary act. This could then be followed by a check whether the proposed alternative utterance and its illocutionary act make sense in the current dialogue context.
For similar reasons, DVMs can also be used to generate more diverse dialogue contributions for the dialogue system itself. Utterances with the same functionality in a dialogue can be identified and used interchangeably by the system. The successful application of skip thoughts (Kiros et al., 2015), an approach to sentence embeddings similar to ours, for paraphrasing further supports this hypothesis.

5. Conclusion

In this work, we introduced the language resource dialogue vector model, a representation of dialogue utterances in vector space. They are inspired by the successful application of word representations in vector space and can be generated utilising existing word2vec implementations if the input is adjusted in a suitable manner. Existing word vector models can be used to preprocess the dialogue data and improve the ability of DVMs to generalise. In our evaluation, we could show that DVMs successfully project semantically similar utterances in close vicinity to each other. We presented multiple research areas in which DVMs could be successfully applied: dialogue act annotation, dialogue policy creation, intent detection and paraphrasing. The implementation and evaluation of DVM-based approaches in those areas remains future work.

6. Acknowledgements

This work is part of a project that has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 645012.

7. Bibliographical References

Allen, J. F. and Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial intelligence*, 15(3):143–178.

Austin, J. L. (1962). How to do things with words. clarendon. *Oxford*, 2005:619–650.

Briggs, G. M. and Scheutz, M. (2013). A hybrid architectural approach to understanding and appropriately generating indirect speech acts. In *AAAI*.

Bunt, H. (1994). Context and dialogue control. *Think Quarterly*, 3(1):19–31.

Cerisara, C., Kral, P., and Lenc, L. (2017). On the effects of using word2vec representations in neural networks for dialogue act recognition. *Computer Speech & Language*.

Deeplearning4j Development Team. (2016). Deeplearning4j: Open-source distributed deep learning for the JVM. *Apache Software Foundation License 2.0*.

Hermann, K. M. and Blunsom, P. (2014). Multilingual models for compositional distributed semantics. *arXiv preprint arXiv:1404.4641*.

Hu, B., Lu, Z., Li, H., and Chen, Q. (2014). Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, pages 2042–2050.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent convolutional neural networks for discourse compositionality. *arXiv preprint arXiv:1306.3584*.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966.

Lin, B.-S., Wang, C.-M., and Yu, C.-N. (2017). The establishment of human-computer interaction based on word2vec. In *Mechatronics and Automation (ICMA), 2017 IEEE International Conference on*, pages 1698–1703. IEEE.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., Song, X., and Ward, R. (2016). Deep sentence em-

bedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(4):694–707.

Rieser, V. and Lemon, O. (2011). *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation*. Springer Science & Business Media.

Scheffler, K. and Young, S. (2002). Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning. In *Proceedings of the second international conference on Human Language Technology Research*, pages 12–19. Morgan Kaufmann Publishers Inc.

Searle, J. R. (1975). *Indirect speech acts*. na.

Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. (2014). A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 101–110. ACM.

Sienčnik, S. K. (2015). Adapting word2vec to named entity recognition. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, number 109, pages 239–243. Linköping University Electronic Press.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the conference on empirical methods in natural language processing*, pages 151–161. Association for Computational Linguistics.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Tsunoo, E., Bell, P., and Renals, S. (2017). Hierarchical recurrent neural network for story segmentation. *Proc. Interspeech 2017*, pages 2919–2923.

Xue, B., Fu, C., and Shaobin, Z. (2014). A study on sentiment computing and classification of sina weibo with word2vec. In *Big Data (BigData Congress), 2014 IEEE International Congress on*, pages 358–363. IEEE.

Zhang, J., Liu, S., Li, M., Zhou, M., Zong, C., et al. (2014). Bilingually-constrained phrase embeddings for machine translation. In *ACL (1)*, pages 111–121.

Zhang, D., Xu, H., Su, Z., and Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and svm perf. *Expert Systems with Applications*, 42(4):1857–1863.

8. Language Resource References

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520. IEEE.

Leech, G. and Weisser, M. (2013). The spaadia annotation scheme. *Retrieved from martinweisser. org/publications/SPAADIA_Annotation_Scheme. pdf (last accessed November 2015)*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

ES-Port: a Spontaneous Spoken Human-Human Technical Support Corpus for Dialogue Research in Spanish

Laura García-Sardiña, Manex Serras, Arantza del Pozo
Speech and Natural Language Technologies, Vicomtech
Mikeletegi Pasealekua 57, 20009 Donostia-San Sebastián
{lgarcias, mserras, adelpozo}@vicomtech.org

Abstract

In this paper the ES-Port corpus is presented. ES-Port is a spontaneous spoken human-human dialogue corpus in Spanish that consists of 1170 dialogues from calls to the technical support department of a telecommunications provider. This paper describes its compilation process, from the transcription of the raw audio to the anonymisation of the sensitive data contained in the transcriptions. Because the anonymisation process was carried out through substitution by entities of the same type, coherence and readability are kept within the anonymised dialogues. In the resulting corpus, the replacements of the anonymised entities are labelled with their corresponding categories. In addition, the corpus is annotated with acoustic-related extralinguistic events such as background noise or laughter and linguistic phenomena such as false starts, use of filler words or code switching. The ES-Port corpus is now publicly available through the META-SHARE repository, with the main objective of promoting further research into more open domain data-driven dialogue systems in Spanish.

Keywords: spontaneous dialogue corpus, human-human dialogue, technical support, transcription, anonymisation, named entities

1. Introduction

Dialogue systems, often referred to as conversational agents or chatbots, are becoming increasingly popular as they allow users to directly interact with a wide range of information systems in a natural way. Customer support is an application scenario with a strong interest in dialogue system development, driven by the promise of intelligent digital assistants available 24x7 to resolve customer requests in a fast, cost-effective and consistent manner (Guzmán and Pathania, 2016).

Data-driven approaches to dialogue system development have shown to be more robust than rule-based techniques to variability in user behaviour, the performance of speech and language processing subcomponents and the dynamics of the task domain (Meena, 2015). Despite their promising results in recent years (Young et al., 2013; Wen et al., 2016; Li et al., 2017), most practical dialogue systems are still built by human experts through significant manual engineering. In most currently deployed systems, rule-based dialogue managers (DM) are combined with statistical natural language understanding (NLU) models capable of classifying intents and their related entities (Williams et al., 2015). In the customer support domain, this limits their application to frequent use cases in specific areas where solutions are well known, predictable and where scripted answers can be developed (Guzmán and Pathania, 2016).

Lack of annotated corpora is the main problem for the development of data-driven systems (Serban et al., 2015). To overcome this issue, it is usual to develop rule-based baselines or to employ the Wizard of Oz (WOZ) technique (Benedı et al., 2006; Rieser and Lemon, 2008) in which a human mimics the intended dialogue system, in order to gather interactions with real users. Although this kind of human-machine data is constrained by the employed baseline systems or the scenarios defined in the followed WOZ approaches, it is useful to bootstrap goal-driven dialogue systems whose policies can be optimised through user simulation and adaptive learning (Schatzmann et al., 2007; Gašić et al., 2013; Serras et al., 2017). On the other hand, human-human dialogue corpora contain unconstrained and unscripted natural dialogue interactions exhibiting traits different from human-machine dialogue (i.e. richer turn-taking and more common grounding phenomena) (Doran et al., 2003), which are more suitable to train more open domain dialogue systems. Human-human customer support corpora would allow progress towards the development of large-scale data-driven dialogue systems capable of handling a wider amount of customer queries.

A considerable amount of corpora are available for building data-driven dialogue systems (Serban et al., 2015). Unfortunately, because customer support interactions occur in commercial settings, most customer support datasets are proprietary and not released to the public due to privacy and data protection reasons. In practice, there are only a couple of publicly available technical support datasets (Lowe et al., 2015; Uthus and Aha, 2013) derived from the Ubuntu IRC channels[1], used to receive technical support for issues related to the Linux-based operating system. Although some data is available from the support channels in other languages, most of the compiled resources are in English.

In this paper, the Spanish Technical Support (ES-Port) Corpus is presented, a compilation of spontaneous spoken human-human dialogues from the technical customer support service of a Spanish telecom operator for companies. The corpus has been directly transcribed from call recordings, annotated at various linguistic and acoustic-related extralinguistic levels, and anonymised in order to comply with data protection legislation. Its release is intended to foster further research into more open domain data-driven di-

[1]These logs are available from 2004 to 2018 at http: //irclogs.ubuntu.com/

alogue systems in Spanish, capable of achieving more natural interactions in the technical support domain.

2. Compilation Process

The raw corpus was provided by an independent telecom operator, dedicated to providing tailor made cloud data centre, fixed voice, IP or mobile telephony and Internet connectivity solutions to companies. In order to serve their clients, they offer 24/7 customer support: 24 hours a day, 7 days a week, 365 days a year. Despite having a multichannel customer service and also providing support through web forms and email, the majority of the clients still prefer calling. Thus, the corpus provided consisted of raw audio recordings of such calls.

2.1. Transcribing the Audio

The first step of the corpus compilation process involved transcribing the provided raw audio data. Details regarding the characteristics of the audio and the followed transcription process are given next.

2.1.1. Audio characteristics

The recorded calls contain both speech and other background sounds, such as channel-associated noises or background music. The type of speech used is spontaneous, and so it includes phenomena such as false starts, mispronunciations, non standard forms, overlapping segments between speakers, unfinished sentences and, in general, speech more focused on conveying the message than on taking care of its form. The audio data consisted of a total of 40 hours , with an average length of 2 minutes per dialogue.

2.1.2. Transcription process

The provided recordings were transcribed using the Transcriber 1.5 annotation tool (Barras et al., 2001). In addition to the orthographic transcriptions, the following phenomena were also annotated:

- speaker turns
- non-speech events (e.g. coughing and laughter) and background acoustic conditions (e.g. noise and music)
- overlapping speech
- false starts, repetitions, unfinished words and non-words
- mispronunciations, lengthening in pronunciation, and typical spoken Spanish shortening of words (e.g. *pa* instead of *para*) or dropping of intervocalic *d* in final syllables (e.g. *demasiao** for *demasiado*, *entrao** for *entrado*)
- continuers and filler words (e.g. *o sea*, *eh*, *hala*, *mhm*, etc.)
- words in a language other than Spanish (when pronounced correctly)

Given the more challenging spontaneous and telephone nature of the data, attempts to follow incremental automation methodologies such as those described in (Pozo et al., 2014) to make the transcription process more productive were not feasible. The word error rates (WER) of generic large vocabulary continuous speech recognition (LVCSR) systems turned out too high to provide any time savings (77.21% in test set).

In the end, the transcription process was carried out fully manually and took a linguist six months working full-time to complete.

2.2. Anonymising the Dialogues

In order to comply with the European data protection legislation (Art29WP, 2014) and not to compromise the right to confidentiality of the individuals involved, the personal information contained in any dataset must be neutralised before releasing the data open to the public in order to be exploited for other purposes.

Data anonymisation is the process of treating personal data in such a way that it can no longer be used to identify the individuals involved, while preserving the value and usefulness of the original format.

2.2.1. Anonymisation practices and standards

Despite European legislation does not prescribe any particular anonymisation technique, randomisation and generalisation approaches are usually employed to anonymise structured datasets in the form of tables or graphs:

- Randomization: involves alteration of the data without losing its value and includes techniques such as noise addition and permutation.
- Generalisation: implies diluting or reducing the granularity of the data and comprises techniques such as aggregation and K-anonymity.

For unstructured text, such as the transcriptions of the technical support recordings in the ES-Port corpus, the following methods have also been proposed in (Dias, 2016):

- Suppression: the element to be anonymised is replaced by some neutral indicator, e.g. 'XXXXX'.
- Tagging: the element to be anonymised is replaced by a label which can refer to its class or identifier, e.g. 'ORGANISATION123'.
- Substitution: the entity to be anonymised is substituted by another entity, e.g. 'Juan' for 'Pedro'. The choice of the new entity can be random from a dictionary, swapped with another entity within the document, 'intelligently' substituted by an entity sharing the same features, or applying a generalisation technique to the item (e.g. replacing 'University of the Basque Country' by 'University').

The technique chosen to anonymise the ES-Port corpus was substitution because readability and coherence are kept and the result is a natural anonymised text. The process is described in Section 2.2.3..

Table 1: NER and NERC Precision (Pr), Recall (Rc), and F1 scores for the three taggers on our test set.

	NER			NERC		
	Pr	Rc	F1	Pr	Rc	F1
IXA Pipes	0.32	0.62	0.42	0.26	0.50	0.34
FreeLing	0.36	0.65	0.47	0.24	0.54	0.33
CoreNLP	0.47	0.96	0.63	0.36	0.99	0.53

2.2.2. Identifying the features to be anonymised

The first step in the anonymisation process is to identify the type of elements in the dataset that refer to personal information or that could possibly be used in any way to identify the people involved, endangering their right to confidentiality. Considering the nature of the information given in the ES-Port corpus, we decided to anonymise the following types of elements:

- Elements referring to individuals' basic personal information: names, surnames, name diminutives or nicknames, personal identification numbers.
- Contact information and digital trace elements: phone numbers[2], IP addresses, user names and numbers, email addresses, postal addresses, web domains.
- Workplace and organisation-related elements: names of organisations, NIFs (tax identification number) and CIFs (tax code), easily linkable names of products and services, prices.
- Other elements: card numbers and bank accounts, dates, locations, trouble ticket numbers, dispatch notes, passwords, spellings of any of the previous elements.

2.2.3. Anonymisation process

Once the types of elements that needed to be anonymised were identified, the anonymisation process was carried out in a semi-automatic way.

First, the items to be anonymised were selected and categorised. We tried to automate the selection process by using different Named Entity Recognition and Classification (NERC) tools available for the Spanish language, namely IXA Pipes (Agerri et al., 2014), FreeLing (Carreras et al., 2004) and Stanford CoreNLP (Manning et al., 2014). Although these taggers have reported good results on planned written language such as news texts, trial tests on a small dataset of our spontaneous spoken technical support corpus were too poor to automate the process of selecting and categorising the items to be anonymised, as their use would still require considerable manual revision and correction. Results for the three taggers on the test set both considering entity recognition alone (NER) and entity classification as well (NERC) are presented in Table 1. In addition, none of the taggers covered all types of items that had to be anonymised, as is the case of numbers, months and individual letters in spellings.

The next step was automatic and consisted in randomly substituting the selected items for an element of the same characteristics according to its type (organisation, number, male/female name, etc.). Once an item was anonymised, its substitution was kept throughout the whole dialogue in order to maintain coherence, but not across dialogues so as to prevent possible linkability issues. New names provided for organisations and domains are made up and did not correspond to any existing entity at the time the anonymisation was carried out. The final step involved manual revision of the results and correction of coherence errors (e.g. non matching spellings).

[2]Some prefixes were kept if relevant to the conversation.

Table 2: Entity tags used in the anonymisation process.

Utterance	**Replacements**	**Tags**
"Soy Bárbara de Cincode"	Bárbara	female_name
	Cincode	organisation
"Arturo Noriega arroba Hotmail punto es, tengo que poner?"	Arturo	male_name
	Noriega	surname
	Hotmail	mail
"te la digo, es M de Madrid,"	M	letter
	Madrid	place
"el último registro es del veintisiete de septiembre."	veintisiete	number
	septiembre	month
"Inexistent punto com."	Inexistent	domain
"tiene que entrar= a CompDNS"	CompDNS	product/ service

The named entity categories of the elements anonymised following the process described above have been kept in the compiled corpus. As a result, ES-Port also includes named entity annotations. However, these were anonymisation-oriented and therefore the number of classes and specificity of the tags are more granular than in the typical NERC approaches. Nevertheless, the tags used are easily generalisable to the usual four (PERSON, LOCATION, ORGANISATION, MISCELANEA) or six (plus NUMBER and DATE) NERC classes. The tags used and real examples of their usage are shown in Table 2.

Overall, the described anonymisation process took a linguist eight months working part-time to complete.

3. The Compiled Corpus

This section describes the gathered corpus quantitatively and qualitatively.

3.1. The corpus in numbers

Table 3 summarizes the basic statistics of the ES-Port corpus regarding its number of dialogues, turns and overlaps, its vocabulary and its amount of filler and foreign words.

Table 3: Corpus Characteristics

Num. Dialogues	1170	Vocabulary size	11221
Num. Turns	65239	Labelled Filler Words	37
Avg. Turns per Dialogue	55.76	Filler Words Freq.	26574
Avg. Turn Length	8.20	Foreign Words Freq.	3294
Num. Overlap Turns	11329	English Words Freq.	3017

As can be seen, the corpus presents attributes typical of spontaneous spoken human-human interaction such as overlapping turns (around 17% of the turns) and rich use of

Table 4: Excerpt from a dialogue, including turn index (T), speakers (S) and the actual annotated and anonymised utterance transcription (U).

T	S	U
29	spk1	De todas formas
	spk2	esto
30	spk1	si has +enviado el correo estate tranquilo porque <se=> se para.
31	spk2	(*EVENT*: noise-rire)
	spk2	<%mm> Es <lom-> <i-> incluso si lo <envi-> <%aver> <su-> supuestamente hasta las cinco y media, no?
32	spk1	Sí.
33	spk2	(*EVENT*: noise-rire)
	spk2	Y si lo envío a las cinco y diez se cancela?
34	spk1	Sí, sí.
35	spk2	Ay, dios (*EVENT*: pronounce-ch)
36	spk1	<%mhm>
37	spk2	Ay, mi madre (*EVENT*: pronounce-ch)
	spk2	no puedo largarme de aquí digamos.
38	spk1	<%eh> si quieres
	spk2	<%eh>
39	spk1	llamar un poquillo más tarde y te intento pasar con él de nuevo.
40	spk2	Es que no hay ninguna forma de que, ningún número que yo pueda=
41	spk1	No, no.
	spk2	llamarles
42	spk2	a ellos o
43	spk1	No, <%osea>
	spk2	algo?
44	spk1	que le estoy llamando yo y no me responde.
45	spk2	(*EVENT*: noise-nontrans)
	spk2	<ueh->
	spk2	okay (*LANGUAGE*: en)
	spk2	<%pues> muchas gracias.
46	spk1	<%venga> a ti.

continuers and filler words (approximately 5% of total word occurrences). It is interesting to note that since the corpus is gathered from an IT domain, English foreign words are quite common, reaching up to 91.59% of all foreign words occurrences and constituting around 3.24% of the vocabulary.

3.2. Sample data description

In order to give an idea of the type of information and phenomena present in the ES-Port corpus, Table 4 shows an excerpt of one of its dialogues.

Different types of speech and non-speech events and background acoustic conditions occur often along the corpus. We find instances in turns 35 and 37, where the event tags indicate that the utterance was whispered, and in turn 45, indicating that the speech was unintelligible and could not be transcribed. Other instances can be found in turns 31 and 33, where the speaker laughs before continuing talking.

Overlapping speech is also common. Examples can be seen in turns 29, 41, and 43, where two different speakers intervene within the same turn. Speech, extralinguistic phenomena, or a combination of both can be overlapped.

False starts, repetitions, incomplete words, and nonwords are very common. False starts and repetitions are tagged simply using `<word>`, as in turn 30, while incomplete words and other nonwords are tagged as `<nonword->` and can be found in turns 31 and 45 of the excerpt.

Deviations from the standard pronunciation take place regularly in the corpus. An instance dropping the intervocalic phoneme /d/ in a final syllable can be found in turn 30, marked with a + symbol. Lengthening examples appear in turns 30 and 40, marked with an = symbol.

The words tagged following the pattern `<%word>` correspond to a set of 37 filler words (e.g. *o sea*, 'I mean') and continuers (e.g. *mhm*) frequently used in the corpus. Some instances of these can be found in turns 31, 36, 38, 43, 45, and 46.

Finally, words from a language other than Spanish appear quite often, especially in English. The language event in turn 45 indicates that a foreign word was used, in this case from English.

3.3. Potential Applications

The ES-Port corpus is a source of annotated spontaneous spoken human-human dialogues which may be valuable for several research tasks and applications. In this section, a few of them are mentioned.

Our main objective is to promote more open and natural dialogue interactions in Spanish customer support. Although it does not yet include dialogue act annotations, the corpus as is could be used to explore unsupervised approaches to dialogue system development in Spanish. These approaches include modelling the language of the system to generate more human-like prompts, modelling the language of the user to better detect the nuances of human-human communication or analysing the turn-taking dynamics for incremental dialogue processing.

The corpus annotated at the current level can also be exploited to develop supervised approaches for spontaneous LVCSR in Spanish (including automatic capitalization and punctuation) or to develop NERC tools that work better on dialogue text.

Finally, linguistic research in Spanish may use this data to study a wide range of issues, such as the use of discourse markers and filler words in conversation, their meaning in context, and how they influence the dialogue, or the strategies used for turn-taking and self-correction, among others. Other interesting phenomena for study are code switching and the abundant use of words from the English language in the IT domain.

3.4. Data sharing

The current version of the ES-Port dialogue corpus is available via META-SHARE[3] in the repository of the University

[3]The raw audio corpus cannot be released for public access, since it contains sensitive data which falls under the European General Data Protection Regulation (GDPR)

of the Basque Country UPV/EHU[4] under the name of ES-PORT.

4. Conclusions and Future Work

A spontaneous spoken human-human technical support dialogue corpus in Spanish has been transcribed and anonymised. At this point, the corpus contains annotations referring to linguistic and acoustic-related extralinguistic phenomena such as music, laughter, use of filler words and code switching in conversation. Named entities anonymised using the substitution technique are also annotated. The ES-Port corpus is now publicly released so it can be used for dialogue research or the adaptation of LVCSR and NERC systems to spontaneous dialogue. Future work includes dialogue act annotation of the corpus.

5. Acknowledgements

The authors would like to thank the telecom operator who has kindly provided the raw technical support call recordings. We would also like to thank Aholab, the Signal Processing Laboratory of the University of the Basque Country, for granting us access to their META-SHARE node to share our corpus.

6. Bibliographical References

Agerri, R., Bermudez, J., and Rigau, G. (2014). Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *LREC*, volume 2014, pages 3823–3828.

Art29WP. (2014). Article 29 data protection working party: Opinion 05/2014 on anonymisation techniques.

Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (2001). Transcriber: development and use of a tool for assisting speech corpora production. *Speech Communication*, 33(1):5–22.

Benedı, J.-M., Lleida, E., Varona, A., Castro, M.-J., Galiano, I., Justo, R., López, I., and Miguel, A. (2006). Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: Dihana. In *Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 1636–1639.

Carreras, X., Chao, I., Padró, L., and Padró, M. (2004). Freeling: An open-source suite of language analyzers. In *LREC*, pages 239–242.

Dias, F. M. C. (2016). Multilingual automated text anonymization.

Doran, C., Aberdeen, J., Damianos, L., and Hirschman, L., (2003). *Comparing Several Aspects of Human-Computer and Human-Human Dialogues*, pages 133–159. Springer Netherlands, Dordrecht.

Gašić, M., Breslin, C., Henderson, M., Kim, D., Szummer, M., Thomson, B., Tsiakoulis, P., and Young, S. (2013). On-line policy optimisation of bayesian spoken dialogue systems via human interaction. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8367–8371. IEEE.

Guzmán, I. and Pathania, A. (2016). Chatbots in customer service. *Accenture Interactive*.

Li, X., Chen, Y.-N., Li, L., and Gao, J. (2017). End-to-End Task-Completion Neural Dialogue Systems. *ArXiv e-prints*, March.

Lowe, R., Pow, N., Serban, I., and Pineau, J. (2015). The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *CoRR*, abs/1506.08909.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Meena, R. (2015). *Data-driven Methods for Spoken Dialogue Systems*. Ph.D. thesis, KTH, Royal Institute of Thechnology.

Pozo, A. D., Aliprandi, C., Álvarez, A., Mendes, C., Neto, J. P., Paulo, S., Piccinini, N., and Raffaelli, M. (2014). Savas: Collecting, annotating and sharing audiovisual language resources for automatic subtitling. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May.

Rieser, V. and Lemon, O. (2008). Learning effective multimodal dialogue strategies from wizard-of-oz data: Bootstrapping and evaluation. In *ACL*, pages 638–646.

Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., and Young, S. (2007). Agenda-based user simulation for bootstrapping a pomdp dialogue system. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 149–152. Association for Computational Linguistics.

Serban, I. V., Lowe, R., Henderson, P., Charlin, L., and Pineau, J. (2015). A survey of available corpora for building data-driven dialogue systems. *CoRR*, abs/1512.05742.

Serras, M., Torres, M. I., and Del Pozo, A., (2017). *Online Learning of Attributed Bi-Automata for Dialogue Management in Spoken Dialogue Systems*, pages 22–31. Springer International Publishing, Cham.

Uthus, D. and Aha, D. (2013). The ubuntu chat corpus for multiparticipant chat analysis.

Wen, T.-H., Vandyke, D., Mrksic, N., Gasic, M., Rojas-Barahona, L. M., Su, P.-H., Ultes, S., and Young, S. (2016). A Network-based End-to-End Trainable Task-oriented Dialogue System. *ArXiv e-prints*, April.

Williams, J., Kamal, E., Ashour, M., Amr, H., Miller, J., and Zweig, G. (2015). Fast and easy language understanding for dialog systems with microsoft language understanding intelligent service (luis). In *Proceedings of 2015 SIGDIAL Conference, Prague*. ACL – Association for Computational Linguistics, September.

Young, S., Gasic, M., Thomson, B., and Williams, J. D. (2013). Pomdp-based statistical spoken dialog systems: A review. In *Proceedings of the IEEE*, volume 101(5), pages 1160–1179.

[4] http://aholab.ehu.es/metashare/repository/search/

From Analysis to Modeling of Engagement as Sequences of Multimodal Behaviors

Soumia Dermouche, Catherine Pelachaud
CNRS UMR 7222, ISIR, Sorbonne Universités
4 Place Jussieu Paris, France
{dermouche, pelachaud}@isir.upmc.fr

Abstract

In this paper, we present an approach to endow an Embodied Conversational Agent with engagement capabilities. We relied on a corpus of expert-novice interactions. Two types of manual annotation were conducted: non-verbal signals such as gestures, head movements and smiles; engagement level of both expert and novice during the interaction. Then, we used a temporal sequence mining algorithm to extract non-verbal sequences eliciting variation of engagement perception. Our aim is to apply these findings in human-agent interaction to analyze user's engagement level and to control agent's behavior. The novelty of this study is to consider explicitly engagement as sequence of multimodal behaviors.

Keywords: Engagement, Non-verbal behavior, ECA, Temporal Sequence Mining, Human-agent interaction

1. Introduction

Embodied Conversational Agents (ECA) are virtual characters that can interact with a user. Today, ECAs are increasingly being integrated in our everyday life, for example, for training, social coaching, and science teaching (Graesser et al., 2007). Our work is part of the H2020 European project ARIA-VALUSPA (Valstar et al., 2016) that aims to build an ECA able to play the role of an expert and to share its domain knowledge with a novice user. In this project, we focus on an important aspect of human-agent interaction, namely, engagement that ensures the interaction to go on. A survey of engagement definition in human-agent interaction is given in (Glas and Pelachaud, 2015). Engagement can be defined as: *"the value that a participant in an interaction attributes to the goal of being together with the other participant(s) and of continuing the interaction"*(Poggi, 2007). Engagement is also defined as "the process by which participants involved in an interaction start, maintain and terminate an interaction" (Sidner et al., 2005). Engagement is not measured from single cues, but rather from several cues that arise over a certain time window (Peters et al., 2005).

The goal of this work is twofold: (1) user's engagement detection: the ECA should be able to detect, in real time, the engagement level of the user. (2) ECA's engagement modeling: the ECA should adapt its behavior in order to maintain the desired level of engagement during the interaction. Specifically, this work investigates what are the multimodal behaviors that participate to a change of perception of the engagement level in a human-agent interaction. For this, we rely on sequence mining algorithms to associate user's and agent's non-verbal behaviors with different engagement levels.

Several works focused on associating verbal and non-verbal behaviors with engagement in human-agent interaction, but most of them are limited to few signals such as smiling and head nod (Allwood and Cerrato, 2003; Castellano et al., 2009). In our work, we consider a set of non-verbal modalities (gesture, head movement and directions, smiling, etc.) jointly with their temporal synchronization (order, starting time and duration).

Our study is performed on the NoXi dataset, a corpus of expert-novice interaction (Cafaro et al., 2017), that we have manually annotated according to the engagement levels of both expert and novice. The use of sequence mining allowed us discovering relevant patterns for different engagement levels.

2. Related Works

During the last decades, engagement modeling has gained increasing attention due to the growing number of conversational agents and the important role that engagement plays in human-agent interaction. Engagement can be expressed by both verbal and non-verbal behaviors. Engagement can be directly linked, for example, to prosodic features (Yu et al., 2004) and verbal alignment behaviors (Pickering and Garrod, 2004).

Other studies have reported that smiling (Castellano et al., 2009) and head nod can provide information about the participant's engagement (Allwood and Cerrato, 2003). Gaze is also an important cue of engagement level (Sidner et al., 2003; Peters et al., 2005; Nakano, Yukiko I. and Ishii, 2010), for example, looking at the speaking partner can be interpreted as a cue of engagement, while looking around the room may indicate the intention to disengage. Moreover, a correlation has been found between engagement and several body postures (Mota and Picard, 2003; Sanghvi et al., 2011). In short, engagement can be conveyed by multimodal behaviors (Sidner et al., 2003).

Results from (Ivaldi et al., 2017) confirmed the relationship between attitudes and engagement in human-robot interaction. For instance, more user is extrovert, he tend to more talk to the robot. Also a negative attitude towards robots have been correlated with less gaze at the robot's face. Culture is an important aspect to take in account when modeling engagement for virtual agents (Yu et al., 2016; Matsumoto, 2006). Yu *et al.* found that in American culture, more smiles represents more engagement, while in Chinese culture, similes are less related to engagement. Another



Self_adaptor Iconic Beat Deictic Metaphoric

Figure 1: Examples of rest gestures.

example is that Arabs gaze (gaze and speech are the main social signals used to evaluate engagement (Sidner et al., 2010)) much longer and more directly at their partners than do Americans (Matsumoto, 2006).

3. Multimodal Corpus Representation

In this section, we present the NoXi corpus, as well as the tool we have used for annotation (NOVA). We also present our annotation scheme for the non-verbal behavior and engagement.

3.1. NoXi

This work is part of the H2020 project ARIA-VALUSPA (Valstar et al., 2016) (*Artificial Retrieval of Information Assistants - Virtual Agents with Linguistic Understanding, Social skills and Personalized Aspects*). In this project, a database of multilingual natural dyadic interactions, named NoXi (Cafaro et al., 2017), has been collected. NoXi is publicly available through a web interface[1]. NoXi provides spontaneous interactions that involve an expert and a novice discussing about a given topic (e.g.; sports, politics, videogames, travels, music, etc.). The dataset contains over 25 hours of dyadic interactions spoken in multiple languages (mainly English, French, and German). In this work, we use the French part of NoXi database which is composed of 21 sessions. The total duration of all these sessions is 7 hours and 25 minutes.

3.2. NOVA

In the context of the ARIA-VALUSPA project, a graphical tool named NOVA (Baur et al., 2015) has been developed to review and annotate the recorded data [2]. NOVA allows exploring richer data such as skeleton or face streams and by proposing various annotation schema (discrete or continuous). We use NOVA as annotation tool.

3.3. Manual Annotations of NoXi Corpus

Table 1 summarizes the multimodal behaviors that we manually annotated by adapting the MUMIN multimodal coding scheme (Allwood et al., 2007). We use a discrete annotation scheme to label body behavior (e.g., gesture, gaze direction and head movement) and continuous scale for engagement annotation. The manual annotations that we have realized on NoXi corpus will be publicly available through a Web interface[3].

- **Conversation states**
  We annotate four conversation states: both interlocutors speak (BOTH), expert speaks (EXPERT), novice speaks (NOVICE) or no one speaks (NONE).

- **Facial display**
  For facial behavior, we considered: gaze, head movement/direction, smile, and eyebrow movement.

- **Gesture**
  Based on the taxonomies proposed by McNeill (1992), we annotate five categories of gestures: iconics, metaphorics, deictics, beats, and adaptors defined as follows:

  1. Iconics: describe concretely the object that the discourse is presenting.
  2. Metaphorics: in contrary to iconic gestures, these gestures illustrate the speech in an abstract way.
  3. Deictics: point to a location in space, for example, an object a place or a concrete direction),
  4. Beats: do not include semantic information, they are characterized by their simplicity and receptivity.
  5. Adaptors: serve to satisfy bodily needs like scratching.

  Examples of different types of gestures are showed in Figure 1. We also include hand rest positions that can indicate communicator's status and attitude (Allwood et al., 2007). We consider several positions (see Figure 2): arms crossed, hands together, hands in pockets, hands behind back, akimbo (hands on hips), along body (arms are stretched down along the body) etc.

- **Engagement**
  Based on Poggi's definition of engagement, we have continuously annotated the engagement of both expert

[1] https://noxi.aria-agent.eu/
[2] https://github.com/hcmlab/nova
[3] https://noxi.aria-agent.eu/



(1) (2) (3) (4) (5) (6)

Figure 2: Examples of rest poses: (1) hands_crosseddown, (2) hand_inpocket, (3) hand_along, (4) hands_crossed, (5) hands_onhips, (6) hands_behaind.

Table 1: Annotation scheme for the non-verbal behaviors and engagement annotations in NoXi.

Tier (modality)	**Labels**
Conversation states	NONE — EXPERT — NOVICE— BOTH
Head movements	NOD — SHAKE
Head direction	FORWARD — BACK — UPWARDS — DOWNWARDS — SIDEWAYS — SIDE_TILT
Smiles	SMILE
Eyebrow movements	RAISED — FROWN
Gaze direction	TOWARDS_INTERLOCUTOR — UP — DOWN — SIDEWAYS — OTHER
Gestures	ICONIC — METAPHORIC — DEICTIC — BEAT — ADAPTOR
Hand rest positions	ARMS_CROSSED — HANDS_TOGETHER — HANDS_IN_POCKETS — HANDS_BEHIND_BACK — AKIMBO — ALONG_BODY
Engagement	STRONGLY DISENGAGED ... STRONGLY ENGAGED

and novice. To reduce complexity and facilitate the task of continuous annotation, we have defined five levels to annotate changes in the perception of engagement: strongly disengaged, partially disengaged, neutral, partially engaged, strongly engaged. In order to avoid content biases from the verbal behavior when annotating engagement, we have filtered it out, for both expert and novice by applying a Pass Hann Band Filter. In this way, the speech kept the prosodic information without intelligibility of its verbal content.

4. Corpus Analysis

In this section, we present an analysis of the manual annotation. Each single modality (gesture, rest positions, engagement, etc.) has been annotated by one annotator. The inter-annotator agreement (Cohen's Kappa) is greater than 0.5 for all modalities which means that there is a high level of agreement between annotators. Because of space limitation, we only present results about gesture and rest hand positions.

- **Gesture**
  Table 2 shows the number of gestures produced by the expert and the novice. As it can be seen, the expert produces 4 times more gestures (1223) than the novice (293). During the interaction, the expert controls the discussion topic: he holds the floor and he produces more gestures to explain and illustrate his topic. Gestures were mainly either iconics or metaphorics. These gesture types contribute to the perception of higher level of competence according to (Maricchiolo et al., 2009).

- **Rest arms and hand positions**
  The number of arms positions produced by the expert is much more important than that of the novice (cf. Table 3). This can be explained as the novice is mainly a listener and keeps his rest position much longer (mean duration 32.2 seconds) than the expert (mean duration 10.7 seconds).

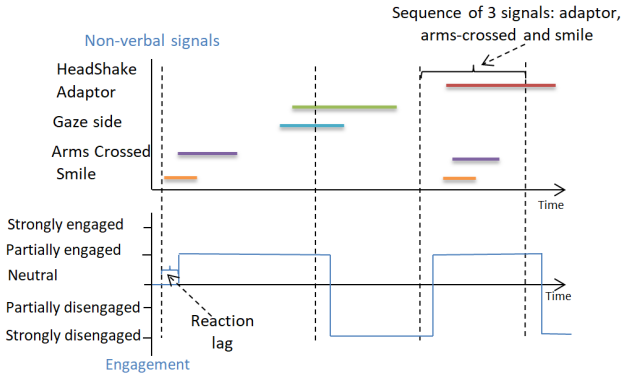


Figure 3: Non-verbal behavior segmentation based on engagement variation. The result is a set of non-verbal signal sequences for each engagement level. For each level, we apply HCApriori to extract the most relevant patterns representing this engagement level.

Table 2: Number of gestures produced by expert and novice.

	Iconics	Metaphorics	Deictics	Beats	Adaptors	Total
Expert	211	158	67	704	83	1223
Novice	49	19	11	105	109	293

Table 3: Number of rest positions for expert and novice.

	Crossed	Together	In_pockets	Behind_back	Akimbo	Along body	Total
Expert	212	486	189	93	48	289	1317
Novice	82	177	214	32	53	54	612

5. Sequence-based Engagement Modeling

Human behaviors are naturally multimodal and sequential: we interact with each other through multiple communication channels (speech, gaze, gesture, etc.). Moreover, these behaviors are temporally coordinated: what behavior we will display next depends, among other phenomena, on our behavior at the present moment and on the other's behavior.

The goal of the present study is to understand how those behaviors are coordinated at critical moments, the sequential patterns they exhibit and their association with different engagement levels. To capture both sequentiality and temporality, we rely on temporal sequence mining, a data mining technique that considers the temporal information like starting time and the duration of signals, a key element in behavior modeling. The ECA should display behaviors at the right moment with the right duration in order to convey a given level of engagement.

A range of temporal sequence mining algorithms exist like HCApriori (Dermouche and Pelachaud, 2016), QTIPrefixSpan (Guyet and Quiniou, 2011) and PESMiner (Ruan et al., 2014). In this work, we rely on HCApriori because it demonstrated a superiority over the state-of-the-art in terms of pattern extraction accuracy and running time (Dermouche and Pelachaud, 2016). In order to prepare a sequence database for HCApriori, we have segmented the non-verbal behaviors based on engagement variations (cf. Figure 3). We took into account the reaction lag of annotators in the continuous annotations by shifting back 2 seconds each of the annotations, as recommended in (Mariooryad and Busso, 2013). For each engagement level, we consider the sequence of non-verbal signals that simultaneously appeared with this level (cf. Figure 3). Thus, we build five datasets of non-verbal signal sequences representing the five engagement levels. Table 4 summarizes the number of sequences we obtain for each engagement level for expert and novice. Finally, we have applied HCApriori to extract temporal patterns (frequent sub-sequences) of nonverbal signals expressing the five engagement levels.

Figure 4 shows a pattern extracted with HCApriori algorithm representing a strong engagement level. This pattern can be interpreted as follows: 0.9 second before the annotator perceives a strong engagement level, the expert smiles to the novice for 1.4 seconds while nodding his head. Then he produces an akimbo gesture (hands on hips) meanwhile he nods and continues smiling. Smiling and head nod have already been reported as being engagement indicators (Castellano et al., 2009; Allwood and Cerrato, 2003). Table 5 gives two examples of extracted patterns (P_1, P_2) for each engagement levels of expert. The starting and the duration of each signal are respectively given between two parentheses. While head nod has been reported as a cue of engagement in several works, our approach associates it with a partial disagreement if it is preceded by a frown and associates it with a strong engagement if it coincides only with a smile. This indicates that the perception of non-verbal signal may change with respect to the signals occurring before and after it.

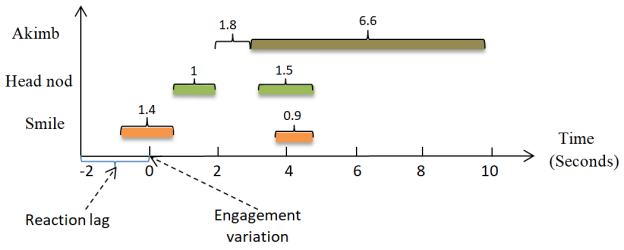


Figure 4: Pattern example representing a strong engagement level.

In Table 6, we compute the percentage of some non-verbal signals that occurred in the patterns of expert for the five engagement levels. Here are some conclusions that we can draw from this table:

- Smile signal occurred in about 66% of the patterns representing a strongly engaged level. Within this level of engagement, the mean starting time of smile is -0.05 seconds, this means that 0.05 seconds before the perception a strong engagement, the expert smiles. In other words, the perception of strong engagement is triggered by the expert's smile. On the other hand, for the partially disengaged level, the mean starting time of smile is 1.5 seconds: smile is produced 1.5 seconds after the perception of a partial engagement.

- Adaptor gestures and frown were also more frequently present in patterns characterizing a strongly disengaged level with a percentage of 25% and 33% respectively.

- Although head nod is usually reported as an indicator of engagement, we found that this signal is more representative of disengagement (33% occurrence in strong disengagement level). This suggests that the

Table 4: Number of sequences of each engagement level for both expert and novice.

	Strongly disengaged	Partially disengaged	Neutral	Partially engaged	Strongly engaged	Total
Expert	48	373	373	561	126	1481
Novice	116	432	509	558	64	1679

Table 5: Some examples of extracted patterns for the five engagement levels.

Engaged level	Pattern example
Strongly disengaged	P_1= Eyebrow down (2, 1.48), P_2= Adaptor(4.08, 2.12)
Partially disengaged	P_1=Eyebrow down(-1.6, 5) Head nod(5.1, 2.7), P_2=Arms crossed(-1.68, 4.12) EyebrowUp (-1.36, 2.64)
Neutral	P_1=Along_body(-1.04, 9.5) Beat(0.37, 3.5), P_2= Beat(-1.4, 2.28) Smile (1.56, 1) Nod(2.52,1.72)
Partially engaged	P_1=Metaphoric (-1.8, 1.75) Iconic(-0.3,2.5), P_2=Iconic(-1.25,2.22) Iconic(2,5)
Strongly engaged	P_1=Smile(-0.76, 3.64) Head nod (0.24, 0.36), cross (-1.14, 13.77), P_2= Smile(-1.08,1.36) Smile (1.64,1.4)

Table 6: Percentage of some non-verbal signals that occurred in the patterns of expert for the five engagement levels.

	Strongly disengaged	Partially disengaged	Neutral	Partially engaged	Strongly engaged
Smile	0%	4%	16%	20%	66%
Eyebrowdown	33%	20%	25%	18%	8%
Nod	33%	20%	22%	24%	25%
Adaptor	25%	10%	11%	3%	0%

perception of non-verbal signal change according to its context. For example, nodding while smiling and performing an adaptor gesture was associated with a partial disengaged level.

6. Conclusion

In this paper, we presented a sequence-mining based approach toward engagement modeling from a corpus of expert-novice interactions. Sequence mining allowed us to extract relevant patterns associated to five engagement levels. While a part of our results perfectly supports previous works, some of our findings are complementary to the current state-of-the-art. This demonstrates that temporal characteristics, like starting time and duration of behaviors, are essential to studying engagement.

Our future purpose is to apply sequence mining results in human-agent interaction: (1) using expert patterns to model the desired engagement level of an ECA during the interaction. (2) Exploiting the patterns representing novice engagement for interpreting user's non-verbal behaviors in real-time and associate it with different engagement variations for allowing the agent to react accordingly.

7. Acknowledgements

Funded by European Union Horizon 2020 research and innovation programme, grant agreement No 645378.

8. Bibliographical References

Allwood, J. and Cerrato, L. (2003). A study of gestural feedback expressions. In *First Nordic Symposium on Multimodal Communication*, pages 7–22.

Allwood, J., Cerrato, L., Jokinen, K., Navarretta, C., and Paggio, P. (2007). The MUMIN annotation scheme for feedback, turn management and sequencing. *International Journal of Language Resources and Evaluation*, pages 1–18.

Baur, T., Mehlmann, G., Damian, I., Lingenfelser, F., Wagner, J., Lugrin, B., André, E., and Gebhard, P. (2015). Context-Aware Automated Analysis and Annotation of Social Human–Agent Interactions. *ACM Transactions on Interactive Intelligent Systems*, 5(2):1–33.

Cafaro, A., Wagner, J., Baur, T., Dermouche, S., Torres, M. T., Pelachaud, C., Andr, E., and Valstar, M. (2017). The NoXi Database : Multimodal Recordings of Mediated Novice-Expert Interactions. In *ICMI'17,*, pages 350–359, Glasgow, Scotland. ACM.

Castellano, G., Pereira, A., Leite, I., Paiva, A., and McOwan, P. W. (2009). Detecting user engagement with a robot companion using task and social interaction-based features. *Proceedings of the 2009 international conference on Multimodal interfaces - ICMI-MLMI '09*, (January 2009):119.

Dermouche, S. and Pelachaud, C. (2016). Sequence-based multimodal behavior modeling for social agents. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction - ICMI 2016*, pages 29–36, Tokyo, Japan. ACM.

Glas, N. and Pelachaud, C. (2015). Definitions of engagement in human-agent interaction. In *2015 International Conference on Affective Computing and Intelligent Interaction, ACII 2015*, pages 944–949.

Graesser, A., Chipman, P., King, B., Mcdaniel, B., and Mello, S. D. (2007). Emotions and Learning with AutoTutor. *13th International Conference on Artificial Intelligence in Education (AIED 2007)*, pages 569–571.

Guyet, T. and Quiniou, R. (2011). Extracting temporal

patterns from interval-based sequences. In *International Joint Conference on Artificial Intelligence (IJCAI 2011)*, pages 1306–1311.

Ivaldi, S., Lefort, S., Peters, J., Chetouani, M., Provasi, J., and Zibetti, E. (2017). Towards Engagement Models that Consider Individual Factors in HRI : On the Relation of Extroversion and Negative Attitude Towards Robots to Gaze and Speech During a Human â Robot Assembly Task Experiments with the iCub humanoid. *International Journal of Social Robotics*, 9(1):63–86.

Maricchiolo, F., Gnisci, A., Bonaiuto, M., and Ficca, G. (2009). Effects of different types of hand gestures in persuasive speech on receivers' evaluations. *Language and Cognitive Processes*, 24(2):239–266.

Mariooryad, S. and Busso, C. (2013). Analysis and Compensation of the Reaction Lag of Evaluators in Continuous Emotional Annotations. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII 2013)*, pages 85–90.

Matsumoto, D. (2006). Culture and nonverbal behavior. *Handbook of nonverbal communication*, pages 219–236.

McNeill, D. (1992). *Hand and mind: What gestures reveal about thought*. Uni- versity of Chicago press.

Mota, S. and Picard, R. W. (2003). Automated Posture Analysis for Detecting Learner's Interest Level. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 5:1–6.

Nakano, Yukiko I. and Ishii, R. (2010). Estimating User's Engagement from Eye-gaze Behaviors in Human-agent Conversations. In *The 5th International Conference on Intelligent User Interfaces*, pages 139—-148. ACM.

Peters, C., Pelachaud, C., Bevacqua, E., Mancini, M., and Poggi, I. (2005). Engagement Capabilities for ECAs. *AAMAS'05 workshop Creating Bonds with ECAs*.

Pickering, M. J. and Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *The Behavioral and brain sciences*, 27(2):169–190; discussion 190–226.

Poggi, I. (2007). *Mind, Hands, Face and Body: A Goal and Belief View of Multimodal Communication*.

Ruan, G., Zhang, H., and Plale, B. (2014). Parallel and Quantitative Sequential Pattern Mining for Large-scale Interval-based Temporal Data. In *2014 IEEE International Conference on Big Data (BigData 2014)*, pages 32–39.

Sanghvi, J., Castellano, G., Leite, I., Pereira, A., McOwan, P. W., and Paiva, A. (2011). Automatic analysis of affective postures and body motion to detect engagement with a game companion. *Proceedings of the 6th international conference on Human-robot interaction - HRI '11*, page 305.

Sidner, C. L., Lee, C., and Lesh, N. (2003). Engagement by Looking: Behaviours for Robots when Collaborating with People. *Proceedings of DiaBruck (the 7th Workshop on Semantics and Pragmatics of Dialogue)*, pages 123–130.

Sidner, C. L., Lee, C., Kidd, C. D., Lesh, N., and Rich, C. (2005). Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1-2):140–164.

Sidner, C. R., Ponsleur, B., Holroyd, A., and L., C. (2010). Recognizing engagement in human-robot interaction. *5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 375–382.

Valstar, M., Baur, T., and Wagner, J. (2016). Ask Alice: An Artificial Retrieval of Information Agent. In *ICMI'17*, pages 419–420.

Yu, C., Aoki, P. M., and Woodruff, A. (2004). Detecting User Engagement in Everyday Conversations. *Science*, page 4.

Yu, Z., He, X., Black, A. W., and Rudnicky, A. I. (2016). User Engagement Study with Virtual Agents Under Different Cultural Contexts. In *Intelligent Virtual Agents - 16th International Conference, IVA2016*, pages 364–368, Los Angeles, CA, USA.

A corpus of German political speeches from the 21st century

Adrien Barbaresi
Berlin-Brandenburg Academy of Sciences
Jägerstr. 22/23 – 10117 Berlin – Germany
barbaresi@bbaw.de

Abstract

The present German political speeches corpus follows from a initial release which has been used in various research contexts. This article documents an updated and extended version: as 2017 marks the end of a legislative period, the corpus now includes the four highest ranked functions on federal state level. Besides providing a citable reference for this resource, the main contributions are (1) an extensive description of the corpus to be released and (2) the description of an interface to navigate through the texts, designed for researchers beyond the corpus and computational linguistics communities as well as for the general public. The corpus can be considered to be from the 21st century since most speeches have been written after 2001 and also because it includes a visualization interface providing synoptic overviews ordered chronologically, by speaker or by keyword as well as consequent accesses to the texts.

Keywords: Web corpus construction, Visualization, Keyword extraction

1. Introduction

1.1. Context

In the Western world, political speeches held by state representatives are often the focus of people's attention and they are endowed with a high symbolical value. Words, themes and phraseology found in speeches are likely to be found in a wide array of other texts as speeches either give an impulse or entail a reaction to events or trends. Speech writing as a genre follows a series of rhetorical rules which make it distinct from other written texts, while the reading in front of an audience also shapes their style toward proper speech. First and foremost, it is necessary to discriminate between political speeches and speeches held by religious dignitaries or other kinds of public addresses. Another distinction can be made between electoral speeches and speeches made by politicians vested with a state function. From a linguistic standpoint, a further difference regards the flow of the speech, as there are speeches which are meant to be read without interruption and others which may not be delivered as fluently, for example in a parliament address with immediate reactions. Finally, there are genres marginally related to speeches such as press conferences and interviews. From a political standpoint, there are qualitative differences as some speeches are part of the daily routine of state bodies whereas others are considered to be important because of a particular political situation or institutional relevance, for example after a new government has taken office.

A major goal of speech collections is to gather information on government work, since political texts are the concrete by-product of strategic political activity and have a widely recognized potential to reveal important information about the policy positions of their authors (Laver et al., 2003). Linguists are also particularly interested in building corpora and improving coverage for this particular genre, while its versatility paves the way towards the use as corpus (Guerini et al., 2008) and its inclusion into reference corpora. Last but not least, since speeches are held in public and part of official debate, they are very often not subject to copyright issues. Consequently, their copyright status makes them highly relevant for replication studies as well as a wide range of purposes. Experience shows that a peculiar scrutiny is required for corpus construction, since clean data is necessary for most approaches, which can be seen for example in the case of Europarl (Graën et al., 2014).

The present endeavor follows from a preliminary corpus of German political speeches released in 2012 (Barbaresi, 2012), which was at that time the first corpus of its kind for German to be made publicly available. The speeches as a whole cannot be found online, they are only partly stored by search engines or other sources. As such, the corpus has preserved texts which have since vanished from official web pages only to be accessible on paper at the German state archives. The year 2017 marks the end of the legislative period in Germany (September) as well as a succession of presidency (April). Both changes prompt for a updated release of the corpus, which has also been extended to other speakers. The present article provides an extensive, citable reference for this resource, beyond the technical documentation from the first release. The main contributions include a description of the corpus to be released and the presentation of the interface used to navigate through the texts, designed for researchers beyond the corpus or computational linguistics communities as well as for the general public.

1.2. Uses so far

The corpus has already been cited in various scientific publications and in different disciplinary contexts. Three main approaches can be distinguished overall: qualitative analysis, mostly in history and political science; quantitative uses, mostly in machine translation; and the integration into reference corpora and corpus linguistics tools. First, the most frequent use seems to be in history and political science in several countries other than Germany, with detailed analyses of the speeches and uses as a basis for comparison (Ditfurth, 2012; James, 2012; Thonfeld, 2014; Górajek, 2015; Pühringer, 2015a; Pühringer, 2015b; Yu, 2015) including research work by Bachelor, Master or PhD students (Schax, 2012; Seewald, 2013; Simons, 2014; van de Rijt, 2015). Examples for quantitative uses are mostly to be found in machine translation studies, for inclusion in shared tasks or system development, for example in the context of the International Workshop on Spoken Language Transla-

tion (Birch et al., 2013; Kilgour et al., 2013; Freitag et al., 2014; Kilgour et al., 2014; Huck and Birch, 2015; Jehl et al., 2015; Müller et al., 2015). Most notably, it has been employed in order to build language models, to provide in-domain texts for statistical machine translation and a clean text source for backtranslations. Since the corpus is freely available, it also has been included in machine learning studies as well (Zhu et al., 2015).

Integration into reference corpora, uses in tools as well as corpus linguistics studies have been the main use cases so far in a linguistic perspective. The corpus has been cited as blueprint for other corpora targeting political speeches (Osenova and Simov, 2012). Parts of it have become components of German reference corpora, first at the Institute for the German Language (Lüngen, 2017), and soon among the resources of the Digital Dictionary of the German Language (Geyken et al., 2017) as a useful complement to existing newspaper corpora, since comparisons on lexical level show that it is significantly different from other written genres (Barbaresi and Würzner, 2014). Additionally, the resource is used by several tools for demonstration purposes, corpus exploration tools like the *Corpus Explorer*[1] (Dang-Anh and Rüdiger, 2015) and the *Leipzig Corpus Miner*[2] (Wiedemann and Niekler, 2016) as well as the citation format CTS (Tiepmar and Heyer, 2017).

Last, while website statistics indicate that the texts are regularly read by the public including politicians and political staff, links to the texts and the interface have been posted by politicians and newspapers (for example in tweets or blog posts), for single texts but also for topical visualizations, e.g. *Der Spiegel* with the notion of values (*Werte*).[3]

2. Contents of present release

2.1. Sources

The corpus gathers four different types of speakers, which currently correspond to the four highest ranked functions on German federal state level, it also includes speeches by affiliated state ministers and state secretaries (members of the Cabinet):

1. President (*Bundespräsident*)
2. President of the Bundestag (*Bundestagspräsident*)
3. Chancellor (*Bundeskanzler*) and corresponding state ministers/secretaries
4. Minister for Foreign Affairs (*Bundesminister des Auswärtigen*), which in recent times frequently carried the title of vice-chancellor (*Vizekanzler*), and corresponding state ministers/secretaries

The 2012 release featured speeches from the presidency and the chancellery only. The present corpus includes two further types as well as an update for a full legislative period. The main focus is on the 21st century, with a few speeches from the end of the 20th century, as shown in Table 1, the token counts are obtained using the tokenizer *SoMaJo* (Proisl and Uhrig, 2016). The collection is not necessarily exhaustive, the official sources from which the speeches have been downloaded are trustworthy but they have no legal obligation to make all the speeches available online. The peculiarities of each subcorpus are described separately below. An effort has been made to exclude interviews, speeches that were held by foreign guests as well as speeches held in languages other than German.

2.1.1. Presidency

The speeches were collected from the online archive of the German Presidency (*bundespraesident.de*). The speeches anterior to 1999 are much less numerous and seem to be only a selection. The collection of speeches by Richard von Weizsäcker is far from being complete. Still, it was added to provide the original texts, as such it is the oldest part of the corpus (starting from 1984).

2.1.2. President of the Bundestag

The speeches were gathered from the professional website of the last president of the Bundestag, Norbert Lammert (*norbert-lammert.de*), who has been in office from 2005 to 2017. The advantage of this source is that a selection has been made, both from speeches held in the Bundestag and on other occasions, so that only highly significant speeches are available on the website. There is however less text to be found than in the archives of the *Bundestag*. Especially for routine interventions, it would be necessary to filter the plenary protocols, which in their current form (PDF format) is impractical.

2.1.3. Chancellery

The speeches are available from the official website of the German Chancellery (*bundesregierung.de*). This source also includes speeches from Ministers of State (*Staatsminister*), who are members of the Cabinet working at the Chancellery. Since these prominent members of the Cabinet are directly linked to representative functions, they have been included in the corpus. The Chancellery speeches represent the largest part of the corpus both in terms of texts and tokens. Documents from four different archives were used: Gerhard Schröder's terms (1998-2005), Angela Merkel's 1st (2005-2009), 2nd (2009-2013), and 3rd term (2013-2017). Consequently, the online archives are not homogeneous, there was no real classification for texts anterior to 2005, where a few unrelated speeches from other politicians are to be found. They appear among others invited speakers in the *others* category. The encoding is sometimes deficient, mostly affecting the punctuation marks and the spaces, which have been partly restored for the corpus. Most speeches from 1998 to 2009 are not available online anymore, some others cannot be found anymore because of a change of website design.

2.1.4. Foreign Affairs

The speeches were collected from the website of the German Ministry of Foreign Affairs (*auswaertiges-amt.de*). A larger proportion of speeches in languages other than German were to be found, these texts were removed, which

[1] http://notes.jan-oliver-ruediger.de/korpora/

[2] http://lcm.informatik.uni-leipzig.de/download.html

[3] https://web.archive.org/web/20160926185139/-http://www.spiegel.de/wissenschaft/mensch/afd-auf-eure-werte-kann-ich-verzichten-kolumne-a-1113649.html

partly explains the skewed distribution of speeches between the different ministers and state ministers.

2.2. Format and metadata

The corpus is made available as a downloadable archive as well as through a series of visualizations and HTML pages. The full text archive is in XML format and Unicode encoding, it follows the guidelines of the Text Encoding Initiative. There is one XML file grouping all the texts of each subcorpus, the files have their own DTD, inspired by the TEI guidelines.[4] Text and metadata have been extracted automatically. In some cases, an automaton has been designed to strip out the salutatory addresses of the speeches using regular expressions, with good accuracy, although not perfect due to the extreme variation among speakers. The following metadata are available (the ones that are not available for all texts are in italics): title(s), speaker, date, *place*, source, *excerpt*, *salutations*, *keywords*.

For a schematic view of the steps needed to build the corpus, see Figure 1 and Figure 2 which focus on the operations performed from the archive to the corpus in XML format and the visualization. The tokenizer *SoMaJo* (Proisl and Uhrig, 2016) and the part-of-speech tagger SoMeWeTa (Proisl, 2018) have been used as they achieve state-of-the-art accuracies on web data for German. The older parts of the corpus have been tagged using the TreeTagger (Schmid, 1995).

2.3. Texts

The presidency corpus contains a total of 2,048 texts comprising about 3.3 million tokens, covering a period ranging from July 1, 1984 to March 12, 2017. The presidency of the Bundestag subpart features 220 speeches, from October 15, 2005 to September 6, 2017, and a total of about 200,000 tokens. The chancellery subcorpus covers a period extending from the December 11, 1998 to the September 21, 2017. It contains a total of 1,831 texts comprising about 5.3 million tokens. Last, the Foreign Affairs part includes 1,552 speeches from January 16, 2006 to September 17, 2017 for a total of approximately 2.1 million tokens. The total amounts to approximately 10.9 million tokens. Table 1 gives a synoptic view of the contents.

Speaker	Date	Texts	MTokens
Presidency			
R. von Weizsäcker	1984-1994	23	59
R. Herzog	1994-1999	135	326
J. Rau	1999-2004	571	902
H. Köhler	2004-2010	527	775
C. Wulff	2010-2012	204	290
J. Gauck	2012-2017	588	933
Presidency of the Assembly			
N. Lammert	2005-2017	220	$\sim$ 200
Chancellery (including members of the Cabinet)			
G. Schröder	1998-2005	420	984
J. Fischer	1998-2005	32	56
R. Schwanitz	1998-2005	23	28
H.-M. Bury	1999-2002	42	74
F.-W. Steinmeier	1999-2005	10	23
M. Naumann	1999-2000	61	121
J. Nida-Rümelin	2001-2003	48	93
C. Weiss	2002-2005	206	299
A. Merkel	2005-2017	1,030	2,694
T. de Maizière	2005-2009	43	89
B. Neumann	2005-2013	323	370
M. Grütters	2013-2017	162	259
others		93	207
Foreign Affairs (including members of the Cabinet)			
F.-W. Steinmeier	2005-2009 2013-2017	552	912
G. Erler	2005-2009	81	116
G. Gloser	2005-2009	48	75
G. Westerwelle	2009-2013	254	277
C. Pieper	2009-2013	84	90
W. Hoyer	2009-2011	42	45
M.-G. Link	2012-2013	19	21
M. Roth	2013-2017	220	248
M. Böhmer	2013-2017	44	39
M. Ederer	2014-2017	12	5
S. Steinlein	2014-2017	34	38
S. Gabriel	2017	42	88
others		121	150

Table 1: Synoptic view of the corpus

2.4. License

As they were given in public, all the speeches can in principle be freely republished as stated by German copyright law[5], so that there is theoretically no copyright restrictions on this corpus, which is quite rare for German texts. Nonetheless, the law indicates that a republication must not target a particular author. Although the situation *de jure* is not clear-cut, the corpus now has been online for more than five years without receiving a warning or a take-down notice, which makes it usable *de facto*. The corpus as a whole is released under the CC BY-SA (attribution and share-alike)[6] license.

[4] http://www.tei-c.org

[5] § 48 UrhG, Öffentliche Reden: http://bundesrecht.juris.de/urhg/__48.html

[6] https://creativecommons.org/licenses/by-sa/4.0/

3. Interface

In order to provide access to the texts, the texts are also published online separately with a specially designed interface. The texts can be listed, explored, and navigated in two different ways: first a chronological list including metadata such as speaker and extracted keywords, and second a diachronic visualization for selected keywords along with statistics. The purpose is to give insights on the evolution in the use of general concepts (e.g. security, Europe, freedom or war) in a kind of Zeitgeist.

3.1. Determination of keywords

The selection of relevant keywords is performed manually after morpho-syntactic linguistic annotation: a shallow parser uses the information after tokenization and POS-tagging to group the tokens in chunks. The valency-oriented chunker (Barbaresi, 2013) uses a bottom-up lin-

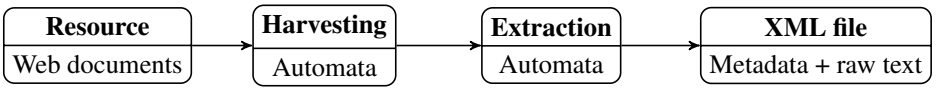


Figure 1: From the web pages to the XML archive

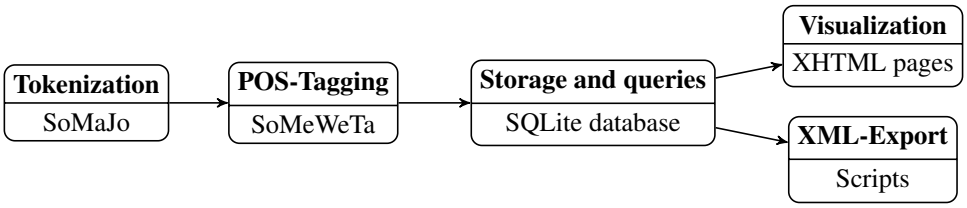


Figure 2: From the XML archive to export and visualization

guistic model implemented using finite-state automata. The transducer takes part of speech tags as input and prints as output assumptions about the composition of the phrases and about the position of the verb. The grouping into possibly relevant chunks enables a valency detection for each verb based on topological fields, the goal is to look for frequent lexical heads as well as important verbs. Nominal and prepositional phrases are in focus, the results are often comparable to text chunks, but the approach is closer to grammatical rules and to the linguistic understanding of a phrase.

Thus, the potential keywords are head nouns of nominal and prepositional phrases, they are then counted and filtered using term frequencies and inverse document frequency (TF-IDF). That way, the selection process combines syntactical and statistical indicators of relevance. The keywords are added as supplementary metadata for the texts. The first eight words by order of frequency (and relevance) appear in the general overview of the texts, whereas the first five ones can be found in the representation of the query by texts. Additionally, they are scrutinized manually and a list of 50 to 100 keywords is used to explore each subcorpus.

3.2. Visualizations

For each selected keyword, visualizations are generated to provide an access to and an overview of the corpus in the form of static web pages, which so far proved easier to maintain. A series of queries are performed on a SQLite database to generate web pages, in a diachronic way to see the keyword evolve in the course of time but also classified by speaker. The data can be sorted by year, name or text. The word frequency is displayed using histograms. This process makes it easier to look for distinctive and/or relevant keywords.The interface also provides the user with the context, more specifically the co-text, five words before, five words after and a link to each text.

The information is put together in CSS/XHTML format, it uses tabbed navigation and takes advantage of web standards. It is light both in size and in client-side computation needs. JavaScript is used to ensure tabbed navigation, to complete the pages on the fly and to highlight words in the texts. A chronological overview with metadata is available, as shown in Figure 3. The list of selected keywords serves as a menu and can be used to browse the corpus, as shown in Figure 4. Clicking on a keyword then leads to the visualizations, most notably a diachronic view as in Figure 5a and an overview sorted by speaker as shown in Figure 5b. The numbers on the right side stand for the texts, the numbers are larger if the keywords are more frequent. Clicking on the numbers then leads to the texts where the keyword is highlighted. It is also possible to browse the texts sequen-

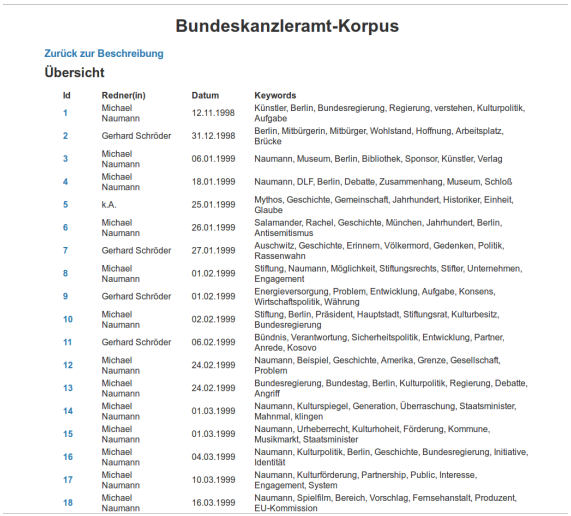
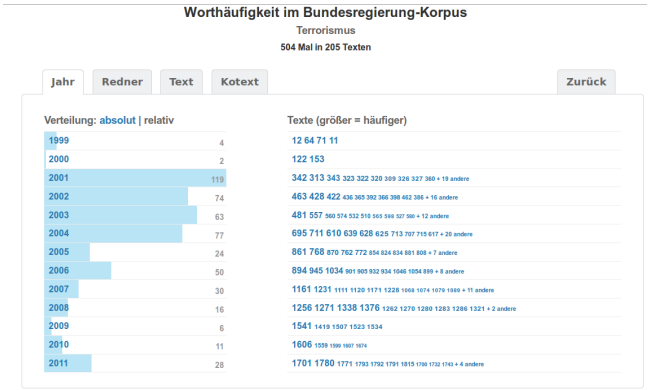


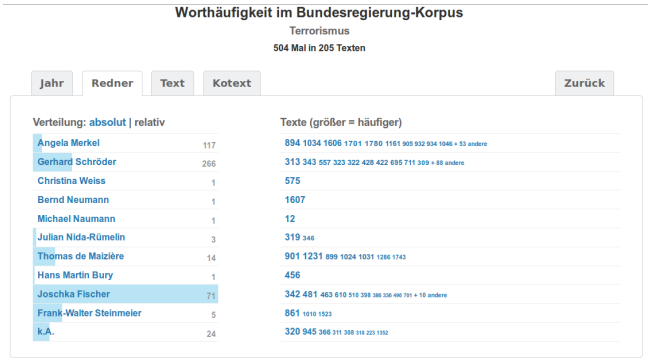
Figure 3: List of texts with metadata



Figure 4: Selected keywords for the Chancellery corpus



(a) Diachronic view for keyword *terrorism*



(b) Overview sorted by speaker for *terrorism*

Figure 5: Aggregated views of the contents in the course of time or grouped by speaker in the Chancellery corpus

tially in chronological order.

4. Conclusion

The main contributions of this article include a scientific reference for the corpus to be released and the description of an interface to navigate through the texts, designed for researchers beyond the corpus or computational linguistics communities as well as for the general public. Indeed, the corpus has been used in various disciplinary contexts, three main approaches can be distinguished overall: qualitative analysis, quantitative uses, and integration into reference corpora and corpus linguistics tools.

The corpus can be considered to be from the 21st century since most speeches have been written after 2001 and also because it includes a modern visualization interface providing both synoptic overviews ordered chronologically, by speaker or by keyword as well as consequent accesses to the texts. An updated and extended version of the corpus is described, it features the four highest ranked functions on federal state level up to the year 2017. The corpus is made available as an archive as well as through a series of visualizations and HTML pages. Data and visualization are both accessible online.[7]

5. Bibliographical References

Barbaresi, A. and Würzner, K.-M. (2014). For a fistful of blogs: Discovery and comparative benchmarking of republishable German content. In *KONVENS 2014, NLP4CMC workshop*, pages 2–10. Hildesheim University Press.

Barbaresi, A. (2012). German Political Speeches – Corpus and Visualization. Technical report, École Normale Supérieure de Lyon.

Barbaresi, A. (2013). A one-pass valency-oriented chunker for German. In Hans Uszkoreit Zygmunt Vetulani, editor, *Language & Technology Conference*, Proceedings of the 6th Language & Technology Conference, pages 157–161, Poznan, Poland.

Birch, A., Durrani, N., and Koehn, P. (2013). Edinburgh SLT and MT system description for the IWSLT 2013 evaluation. In *Proceedings of the 10th International Workshop on Spoken Language Translation*, pages 40–48.

Dang-Anh, M. and Rüdiger, J. O. (2015). From Frequency to Sequence: How Quantitative Methods Can Inform Qualitative Analysis of Digital Media Discourse. *10plus1: Living Linguistics*, 1:57–73.

Ditfurth, J. (2012). *Zeit des Zorns: warum wir uns vom Kapitalismus befreien müssen*. Westend Verlag.

Freitag, M., Wuebker, J., Peitz, S., Ney, H., Huck, M., Birch, A., Durrani, N., Koehn, P., Mediani, M., Slawik, I., et al. (2014). Combined spoken language translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*.

Geyken, A., Barbaresi, A., Didakowski, J., Jurish, B., Wiegand, F., and Lemnitzer, L. (2017). Die Korpusplattform des "Digitalen Wörterbuchs der deutschen Sprache" (DWDS). *Zeitschrift für germanistische Linguistik*, 45(2):327–344.

Górajek, A. (2015). Von der Mehrdimensionalität der Geschichte–Gerhard Schröder und seine Haltung gegenüber Polen. *Zeitschrift des Verbandes Polnischer Germanisten*, 4(4):273–280.

Graën, J., Batinic, D., and Volk, M. (2014). Cleaning the Europarl corpus for linguistic applications. In *Proceedings of KONVENS*, pages 222–227.

Guerini, M., Strapparava, C., and Stock, O. (2008). CORPS: A corpus of tagged political speeches for persuasive communication processing. *Journal of Information Technology & Politics*, 5(1):19–32.

Huck, M. and Birch, A. (2015). The Edinburgh machine translation systems for IWSLT 2015. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 31–38.

James, J. (2012). *Preservation and National Belonging in Eastern Germany: Heritage Fetishism and Redeeming Germanness*. Palgrave MacMillan.

Jehl, L., Simianer, P., Hitschler, J., and Riezler, S. (2015). The Heidelberg University English-German translation system for IWSLT 2015. *Proceedings of IWSLT*.

Kilgour, K., Mohr, C., Heck, M., Nguyen, Q. B., Nguyen, V. H., Shin, E., Tseyzer, I., Gehring, J., Müller, M., Sperber, M., et al. (2013). The 2013 KIT IWSLT Speech-to-

[7] http://purl.org/corpus/german-speeches

Text Systems for German and English. In *International Workshop on Spoken Language Translation (IWSLT)*.

Kilgour, K., Heck, M., Müller, M., Sperber, M., Stüker, S., and Waibel, A. (2014). The 2014 KIT IWSLT speech-to-text systems for English, German and Italian. In *International Workshop on Spoken Language Translation (IWSLT)*.

Laver, M., Benoit, K., and Garry, J. (2003). Extracting policy positions from political texts using words as data. *American Political Science Review*, 97(2):311–331.

Lüngen, H. (2017). DeReKo – Das Deutsche Referenzkorpus. *Zeitschrift für germanistische Linguistik*, 45(1):161–170.

Müller, M., Nguyen, T.-S., Sperber, M., Kilgour, K., Stüker, S., and Waibel, A. (2015). The 2015 KIT IWSLT Speech-to-Text Systems for English and German. In *International Workshop on Spoken Language Translation (IWSLT)*.

Osenova, P. and Simov, K. I. (2012). The political speech corpus of bulgarian. In *Proceedings of LREC*, volume 2012, pages 1744–1747. ELRA.

Proisl, T. and Uhrig, P. (2016). SoMaJo: State-of-the-art tokenization for German web and social media texts. In *Proceedings of the 10th Web as Corpus Workshop*, pages 57–62. Association for Computational Linguistics.

Proisl, T. (2018). SoMeWeTa: A Part-of-Speech Tagger for German Social Media and Web Texts. In *Proceedings of LREC*. ELRA.

Pühringer, S. (2015a). Markets as "ultimate judges" of economic policies: Angela Merkel's discourse profile during the economic crisis and the European crisis policies. *On the Horizon*, 23(3):246–259.

Pühringer, S. (2015b). Marktmetaphoriken in Krisennarrativen von Angela Merkel. In Walter Otto Ötsch, et al., editors, *Markt! Welcher Markt?*, pages 229–252. Metropolis-Verlag.

Schax, A. (2012). Tracing Transformations-The development of Germany's Strategic Culture during the last two decades. Master's thesis, Utrecht University.

Schmid, H. (1995). Improvements in Part-of-Speech Tagging with an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*.

Seewald, F. (2013). Die deutsche Außen-und Sicherheitspolitik von 2001 bis 2012 im Lichte des Zivilmachtkonzepts. Master's thesis, University of Hagen.

Simons, J. P. (2014). Discourse and the Shift in Social Democratic Ideology and Employment Policies: A Comparison of the PvdA and the SPD. Master's thesis, Leiden University.

Thonfeld, C. (2014). Cosmopolitan Normalisation? The Culture of Remembrance of World War II and the Holocaust in Unified Germany. *TáiDàlLìShǐXuéBào*, 53:181–227.

Tiepmar, J. and Heyer, G. (2017). An Overview of Canonical Text Services. *Linguistics and Literature Studies*, 5(2):132–148.

van de Rijt, L. (2015). Enabling and Constraining: A Study on Possibilities of Agents in the EU-Polity during the Turkish Accession Process from 1999 until 2013. B.S. thesis, Utrecht University.

Wiedemann, G. and Niekler, A. (2016). Analyse qualitativer Daten mit dem "Leipzig Corpus Miner". In *Text Mining in den Sozialwissenschaften*, pages 63–88. Springer.

Yu, T. (2015). The German revolution of 1918-1919 in modern studies and in public perception. *History Magazine: Researches*, 3:280–287.

Zhu, L., Kilgour, K., Stüker, S., and Waibel, A. (2015). Gaussian free cluster tree construction using deep neural network. In *Sixteenth Annual Conference of the International Speech Communication Association*.

`LitText`: Building Literary Corpora for Computational Literary Analysis A Prototype to Bridge the Gap between CL and DH

Andrew U. Frank, Christine Ivanovic
Geoinformation TU Wien, Comparative Literature University Vienna
Gusshausstr. 27-29 1040 Wien Austria, Sensengasse 3a 1090 Wien Austria
frank@geoinfo.tuwien.ac.at, christine.ivanovic@univie.ac.at
{frank, christine}@gerastree.at

Abstract

The design of `LitText` follows the traditional research approach in digital humanities (DH): collecting texts for critical reading and underlining parts of interest. Texts, in multiple languages, are prepared with a minimal markup language, and processed by NLP services. The result is converted to RDF (a.k.a. semantic-web, linked-data) triples. Additional data available as linked data on the web (e.g. Wikipedia data) can be added. The DH researcher can then harvest the corpus with SPARQL queries. The approach is demonstrated with the construction of a 20 million word corpus from English, German, Spanish, French and Italian texts and an example query to identify texts where animals behave like humans as it is the case in fables.

Keywords: linked-data, semantic-web, RDF, SPARQL, digital humanities

1. Introduction

Literary analysis, like all other fields of Digital Humanities (DH) based on the evaluation of texts in natural languages, can benefit from advances in computational linguistics (CL) (Pustejovsky et al., 2017a). Many language tools for linguistic analysis are available, and produce results which could be useful for corpus-based literary analysis. Moretti (2005) and others have demonstrated viable approaches to computational literary analysis, and have produced valuable results (e.g. (Fischer et al., 2017)). Projects like DARIAH (Blümm et al., 2016), LAPSS (Ide et al., 2015), and others have brought together CL tools in unified frameworks to make the results of CL easier to use. Despite this, the uptake of CL in literary studies has been slow so far, and very few of the immense number of literary publications show use of CL.

More efforts in bridging the gap between CL and DH are required. It could be beneficial to take a step back from technical issues, and instead focus on the research practices of literary studies researchers. These usually consist of (1) selecting texts which will be analyzed; and (2) reading the texts and identifying parts with properties relevant to the research goal by underlining them. The approach is probably similar for most text-based DH. A tool for DH should therefore be presented following this model of (1) building a corpus and (2) selecting text pieces of interest.

CL can identify parts with specific text properties in a corpus quicker than a human reader, it can process texts systematically, and it can browse a corpus larger than any human could peruse. The DH researcher has *only* to describe what properties the relevant text parts should have. Lord et al. (2006) followed an approach where the researcher gives some example texts she is interested in, and the search engine then finds all text similar to the given one, with the risk that the automatic detection misses the intention of the researcher. `LitText` is based on explicitly formulated queries using SPARQL (see section 8.).

`LitText` focuses on the methods available for researchers to query processed texts. It reduces manual preprocessing to a minimum, to make it possible to build substantial, research-focus-specific corpora (e.g. of 19th century English novels) which includes texts in multiple languages. `LitText` is centered around a query facility to describe the properties relevant text pieces should have, and retrieves these together with metadata about the texts; other data useful for the analysis and available in linked-data format can be added. Results can be further processed using spreadsheet software.

The DH researcher only has to master two kinds of coding:

- a simple markup language to add metadata to text files and to delimit the literal text of interest;
- a standardized, widely available query language (SPARQL) for which extensive instructional materials and a wide choice of query processing software are available.

Results can be displayed in a regular web browser, or further processed with spreadsheet software. Many DH researchers are already familiar with such programs for sorting, computing, counting etc.

A method to build a specialized corpus with a query facility that can retrieve pieces of text and then manipulate them in a spreadsheet is a processing model that is conceptually close to current research methods in text-based humanities. Empowering text-based DH researchers to build their own corpora processed on their own hardware under their control can also reduce fears of copyright infringement (for a technical solution, see: (Pustejovsky et al., 2017a; Zeng et al., 2014)).

There are three novel aspects in `LitText`, all justified by the focus on "ease of use" for the DH researcher:

1. Processing texts in multiple languages, hiding the technical complexity from the DH researcher.
2. Storing the result of the NLP analysis as linked data (Berners-Lee et al., 2009; Manola et al., 2004).

3. Using the standardized query language SPARQL(Harris et al., 2013) to search for relevant text parts.

Section 2 shows an example task and query which will be used to demonstrate the power of `LitText` in Section 8. Section 3 compares the approach with other, similar efforts. Section 4 will cover the preparation of the text, while Section 5 addresses the processing of text with NLP. In Section 6 we discuss adding a processed text to the corpus, and in Section 7 the inclusion of additional data. Section 8 shows how the example query is formulated and executed. Section 9 reports on actual performance, and moves on to future extension and conclusions.

2. Example Task for Literary Analysis

The literary analysis we envision is a very broad field, and CL support must go beyond mere statistical style analysis (Holmes and Kardos, 2003). The running example query used is to identify "animal fables", considered here as texts in which animals are reported to behave like humans, i.e. they think and use intelligent communication. Examples are e.g. the classic fables ascribed to Aesop; systematic search reveals that modern literary texts include similar literary tropes. Such an analysis requires more language analysis than can be provided by simple word frequency analysis etc.

The proposed algorithm to identify such texts finds sentences where the subject is an animal, and the verb expresses either "think" or "communicate". The approach is to lemmatize the text and to construct dependency trees in which subject and verb can be identified and checked whether the subject noun has "animal" among its hypernyms, as well as the verbs "think" or "communicate". We will show how such a query can be formulated in section 8..

Processing natural language is error-prone, reflecting the flexibility and context-sensitivity of human language. The outlined approach is obviously affected by errors in NLP tools when identifying the dependency structures, i.e. the subject and the verb of a sentence. The current system only considers sentences where the subject is a noun; `LitText` does not yet use the coreference analysis produced by some NLP tools, e.g. Stanford coreNLP (Recasens et al., 2013). The method currently implemented relies on the hypernyms in WordNet (Fellbaum, 1998) and considers only the first, most often encountered WordNet sense. It produces omission errors by failing to identify all nouns describing animals, as well as commission errors by identifying nouns which are not animals. Fortunately, for literary analysis, such errors can be tolerated; the human interpreter reads the identified sentences, decides whether they are useful for his research and discards irrelevant ones.

3. Design Compared to Previous Work

DARIAH is an European infrastructure incorporated in 2014, connecting multiple nationally funded projects to collect tools useful for a wide range of humanity fields to allow them to analyze natural-language text (http://www.dariah.eu/). It uses standard tools (Manning et al., 2014) to process text and allows for the storage of input text and results. The goal is to overcome non-trivial technical installation difficulties, and assure the availability of NLP tools — a first step for researchers from the humanities to apply these tools to their specific problems.

The somewhat comparable project LAPPSgrid (Ide et al., 2015) has similar goals (http://www.lappsgrid.org/). It adds considerations for copyright restrictions attached to the texts to permit the holding of copyright-restricted texts, their processing, and making results available in accordance with those restrictions (e.g. HathiTrust) (Pustejovsky et al., 2017b).

`LitText` splits the work of the DH researcher into two tasks: (1) building the corpus, and (2) searching the corpus. The software to build the corpus applies NLP tools to text which is marked and retains the result for later analysis; the same set of processes is applied to all texts; addition of NLP tools requires changes in the code. The corpus is represented as RDF triples and is searched with standard SPARQL query processors (Harris et al., 2013). The researcher can add new text to the store and they are automatically processed and entered in the corpus, ready to be queried.

4. Preprocessing

Preparing a text for processsing by NLP tools is the first impediment for literary scholars trying to use NLP. The goal of `LitText` is to facilitate the collection of texts into a corpus relevant to a particular investigation. In order to collect many texts into a corpus, the effort to prepare the texts must be minimal, as well as flexible to allow use of different sources. The researcher must (a) add metadata to identify the text and other aspects a researcher would want to include in a query; (b) distinguish the part of text that is to be included in the analysis (here called "literal text") from other text parts in the file; and (c) identify the language a text piece is written in (multiple languages per file are allowed).

In `LitText`, we opt for a simplistic markup for text files. Preparing a text means finding a source, assuring proper UTF-8 encoding (e.g. with iconv), adding some markup for metadata (title, author, year of publication), and separation of the actual text from other text material included in the file (e.g. table of content, preface, original file name) - all of which can be done easily in any text editor.

```
.originalFile http:.../174.txt.utf-8
.publication 1890
.Language: English
.Title: The Picture of Dorian Gray
.Author: Oscar Wilde

.ignoreTo
*** START OF THIS ...
Produced by .... HTML version by .....
.ignoreEnd

The artist is the creator of beautiful things. To
...
```

Much of the material we used comes from Project Gutenberg (Hart, 1992) and we prepared a small tool that produces most of the markup automatically and fixes some issues with UTF-8 encoding; downloading and marking up a literary text from Project Gutenberg takes less than

2 minutes. Corresponding tools for other frequently used sources can be constructed with any suitable programming language (all that is needed is to read a text input and add some markup).

5. Processing

A text with markup is processed by NLP programs, and the result translated into the N-Triples format, which is a W3C recommendation for the semantic web (Beckett, 2014). The conversion of a text file to subject-predicate-object triples is carried out in two phases:

1. The metadata, the layout, and the surface texture (e.g. sections, paragraphs) and, if desired, the full text can be captured and translated into the N-Triples format for later use in query formulation (see example).

2. The text is isolated and sent to language-specific NLP services. Using the markup, the text is split into parts in a single language. Differences in language-specific NLP processing are handled internally. At the moment, `LitText` is prepared to handle English, German, French, Spanish and Italian text, using Stanford Core NLP (Manning et al., 2014) and TinT (Palmero Aprosio and Moretti, 2016). Five server processes are set up at different ports; in addition, German text is lemmatized using TreeTagger as a service at an additional port (Schmid, 2013). Adding a language requires an adaptation of the `LitText` program and setting up a server for the language specific NLP.

An example of the translation of a text head and a line of text as RDF triples:

```
<http://gerastree.at#kafka-kafka_urteil_1913>
    <http://gerastree.at/lit_2014#author>
      "Franz Kafka"@de ;
    <http://gerastree.at/lit_2014#dedikation>
      "für Fräulein Felice B."@de ;
    <http://gerastree.at/lit_2014#titel>
      "DAS URTEIL"@de ;
    <http://gerastree.at/lit_2014#untertitel>
      "EINE GESCHICHTE VON FRANZ KAFKA"@de ;
    a <http://gerastree.at/lit_2014#Werk> .

<http://gerastree.at#kafka-kafka_urteil_1913/L00003>
    <http://gerastree.at/layout_2017#lineNumber> 3 ;
    <http://gerastree.at/layout_2017#lineText>
      "DAS URTEIL"@de ;
    <http://gerastree.at/layout_2017#pageNumber>
      "[54/0002]" ;
    a <http://gerastree.at/layout_2017#Line> ;
    <http://www.w3.org/2000/01/rdf-schema#partOf>
      <http://gerastree.at#kafka-kafka_urteil_1913> .
```

The result of the NLP process in XML format is analyzed and translated into triple structure, preserving the produced Treebank codes (Taylor et al., 2003) as much as possible. Dependency codes are already from the Universal Dependency set (Schmid, 2013) (see http://universal dependenciesorg/). The minimal differences between NER codes (e.g. for German text we find I-LOC, etc. and for Spanish text LUG) can be normalized without loss. The processing produces a file in the N-Triple format, i.e. one linked data triple - subject, predicate, object - per line. Tools to convert the result to a more human-readable format like Turtle are available (e.g. rapper by Beckett (2001)).
The programs are designed to regularly scan a directory and convert any new markup files into N-Triples files. Adding a new markup file to a directory is sufficient to trigger the conversion — and later the inclusion in the corpus — automatically.

6. Storing

The N-Triples are stored in any triple store, some available as open-source or free. We use Jena triplestore (Khadilkar et al., 2012), using utilities we constructed for loading the N-Triple files into the store which relies on standardized HTTP protocol SPARQL update commands (Harris et al., 2013); they should work with any other triplestore. The programs are designed to incrementally build a store by scanning all N-Triples files in a directory, and storing new ones into the triple store.

7. Additional Data Sources

WordNet data for hypernyms are required to answer the example query. WordNet can be downloaded in triple format (McCrae et al., 2014) based on the lemon schema and loaded in a triplestore. The resources, especially the hypernym relations, can be accessed in SPARQL queries.
We opted to store the set of nouns in hypernym relations to "animal" — or, for verbs, "think" (in wordnet "cerebrate´) or "communicate" — separately with a SPARQL update query for later use in the analysis. Storing the hypernym relation reduces the complexity of later query formulation and speeds up processing (see section 8.). An example update query to insert a list of the nouns which are hypernyms to "animal", considering only the first WordNet sense (the SPARQL prefix abbreviations are given in the appendix):

```
insert
  {graph <http://gerastree.at/al>
    { ?lexentry lit:nounClass wn:Animal .
      ?lexentry lemon:writtenRep ?lem .
    }
  }
using <http://gerastree.at/wn>
WHERE
{
  ?lexentry wn:part_of_speech wn:noun .
  ?lexentry lemon:sense ?sense .

  ?sense wn:sense_number ?sensenum .
  filter (?sensenum = 1) .
  ?sense lemon:reference ?synset.

  ?synset wn:hypernym+ ?obj .
  ?obj rdfs:label ?lo .
  filter (?lo = "animal"@eng ) .

  ?lexentry lemon:canonicalForm ?cf .
  ?cf a lemon:Form .
  ?cf lemon:writtenRep ?lem .
}
```

Adding other semantic web data resources is equally straightforward; of particular interest are likely geographic gazetteers, the movie database, and Wikipedia.

8. Querying

A triplestore offers a HTTP protocol query facility, the so-called SPARQL endpoint, where queries are submitted, and answers are returned. SPARQL is a very powerful, general-purpose triplestore query language (Harris et al., 2013). It is somewhat patterned after SQL but logically simpler because storage is always in binary relations (Manola et al., 2004; Prud'Hommeaux et al., 2008).

An example query shows the potential to formulate complex queries: ”find all texts in the store in which “animals” behave like humans, e.g. “think” and “communicate” ”. The result includes the sentences in which such literary forms occur, as well as the author, the title of the text, the subject, and the verb.

```
SELECT ?author ?werk  ?sentform ?subjlemma ?verblemma
FROM <http://gerastree.at/c>
#the corpus of literary text
FROM <http://gerastree.at/a1>
#the graph with the hypernyms
WHERE  {
  ?dep nlp:dependency "NSUBJ" .
  ?dep nlp:dependent ?subj .
  ?dep nlp:governor ?verb.

  ?subj nlp:lemma3 ?subjlemma.
  ?verb nlp:lemma3 ?verblemma.

  ?verb nlp:pos ?catv.
  filter (strstarts (?catv, "V")).

  ?subj nlp:pos ?cats.
  filter (strstarts (?cats, "N")).

  ?lexs lemon:writtenRep ?subjlemma .
  ?lexs lit:nounClass wn:Animal .
  ?lexv lemon:writtenRep ?verblemma .
  {?lexv lit:nounClass wn:Communicate}
      union {?lexv lit:nounClass wn:Cerebrate } .

  ?subj rdfs:partOf ?sent .
  ?sent nlp:sentenceForm ?sentform .

  ?sent rdfs:partOf ?snip .
  ?snip rdfs:partOf ?para .
  ?para lit:inWerk ?werk .
  ?werk lit:author ?author .
}  order by ?werk
```

The literal text is stored in graph `c` and the extracted hypernym relations in graph `a1`. The query progresses top-down: first it fills the variable `?dep` with a node in a dependency tree which is a subject-noun phrase, and takes the subject and the verb part (`?subj` and `?verb`). Next, check that `?verb` is indeed a verb and `?subj` a noun, and retrieve (for subject and verb) the corresponding lemmata as `?subjlemma` and `?verblemma`. The next steps are tests whether the subject is an animal, and whether the verb indicates either communicating or thinking. The remainder of the query is tracing from the sentence to the text (`?werk`) and the author.

The output of the result is stated in the `SELECT` clause, and gives the qualified sentences as well as the author and title of the text containing that sentence.

The processing approach is somewhat simplistic as it omits constructions where the subject is an animal but not a noun - using coreference information would probably reduce such errors. It incorrectly includes constructions where the first sense of the noun is an animal, but the noun is used in a different sense. Sentences in which a dog barks, a cat meows etc. are excluded with special rules (see 9.1.). Texts which should be considered “animal fables” (in the present narrow definition) are characterized by frequent occurrences, while other texts have only a few stray occurrences of qualifying constructions.

Queries to obtain text statistics (distinct word count, length of sentences, etc.) can be formulated with SPARQL `GROUP BY` and `COUNT`.

9. Simplification of Query with Preprocessing

The SPARQL query in the previous section 8. is likely not workable for the intended DH researcher. Several preprocessing steps are possible to simplify the query close to the CPQ style(Hardie, 2012) query
`[pos="noun" & hypernym="animal"] .`

9.1. Construct Hypernym List and Classify Words in Text

A list of all words in WordNet which are hyponyms to some interesting wordclasses (e.g. animal, artefact, communicate) are easy to construct. Using such a list to classify all tokens in a text and mark each with the hypernym of the interesting classes allows to test a token with e.g.
`?tok a nlp:Animal.`
The list of “interesting” wordclasses contains entries like:
`_:animal lit:wordClassPattern wn:Animal ; lit:wordClassRep "animal"@eng ; lit:wordClassKind wn:noun.`
and is easily adaptable.
Stored hypernym relations provide a hook for corrections for undesirable relations. For example, to exclude “mademoiselle” from the class “Animal” requires a triple
`wn:mn wn:isNotClass "mademoiselle"@eng .`
or to exclude “dogs bark” from communication
`wn:bark wn:isNotNounVerbNoun "dog"@eng; wn:isNotNounVerbVerb "bark"@eng.`

9.2. Use Universal Dependency and POS tags

Using the Universal Dependency and PoS tags removes differences between the coding of texts of different languages. CoreNLP (in version 3.9.0) produces UD dependency annotations but not POS tags from the UD.POS list. We can translate the PennTreeTags produced by coreNLP to UD.POS tags or better, use a pipeline which directly produces UD.POS tags (Straka and Straková, 2017).
The dependency graph can be represented in triples with one triple per dependency, at the cost of deviating from the XML structure produced by CoreNLP processors.
E.g. `?verb nlp:nsubj ?subj.`

9.3. Simplified queries

These two preprocessing steps searching for nouns which are hyponyms of animal becomes

```
select ?word
{ ?tok nlp:pos "NOUN" .
  ?tok a nlp:Animal.
  ?tok nlp:lemma3 ?word.
}
```

which is not far from the desired CPQ like query. The more complex query for sentences with animals which think or communicate becomes:

```
select str(?s1) str(?v1) str(?sf)
FROM <http://gerastree.at/english1>
FROM <http://gerastree.at/tk13>
where  {
  ?verb nlp:nsubj ?subj .
  ?subj a  wn:Animal.
  {?verb a wn:Cerebrate } union
    {?verb a wn:Communicate } .
```

```
  ?subj nlp:lemma3 ?sl .
  ?verb nlp:lemma3 ?vl .
  ?subj rdfs:partOf ?sentence .
  ?sentence nlp:sentenceForm ?sf .
}
```

The query is finding a verb with a noun subject (first line) and then to check that the subject is an animal (second line), the verb is in the wordclasses cerebrate or communicate (third line). The remaining lines find the lemmata for subject and verb and the text of the sentence, in which the construct occurs.

10. Performance

Overall performance, even on dated and limited hardware, is satisfactory. We tested on a PC with an Intel i5 processor and 3.2 GHz with 24 GB of memory (the memory is mostly necessary for NLP processing of literary texts like "Ulysses" by Joyce; ordinary grammatical text is processed using less than 8 GB). The java engine was started with 6 GB of memory for each of the coreNLP services and with 10 GB for the jena-fuseki triplestore. The example corpus we built consists of about 266 literary texts (mostly books) for a total of 20.1 million words.

- Download and markup of one text from Project Gutenberg takes less than 2 minutes.
- NLP processing a literary text runs at about 70 words per second for an English text including co-reference analysis, 190 words/second for a German text (with separate lemmatization process) and 530 words/sec for Italian text.
- A word in text produces about 10 triples.
- Storing the triples in the triplestore processes about 4,000 triples per second.
- The preprocessing of the full corpus takes a total of about 2 days.
- the text files take 0.67 GB, the processed and stored triples 2.2 GB and the triplestore 75 GB.

Processing the example query 9.3. shown earlier takes 6 seconds (corpus of English text of 2 M words and hardware as before).
The code can be downloaded from a github account (`andrewufrank.github.com/LitText`) and combined with a suitable triple store (e.g. `jena.apache.org/download/index.cgi`).
The SPARQL endpoint is accessible on `http://nlp.gerastree.at:3030`.

11. Future Work

A number of improvements are envisioned:

- integrate hypernym data for other languages without making the query writing more complex is challenging.
- use ideas from Abstract Meaning Representation Language (AMR) (Vanderwende et al., 2015; Winiwarter, 2015; Xue et al., 2014) to simplify queries across texts in multiple languages.
- use co-reference data produced by NLP programs.
- add other language servers to `LitText` (e.g. Dutch, Japanese, Farsi)
- build multi-lingual corpora in order to facilitate computational comparative literature studies (Ivanovic, 2017); a prototype is to be developed with original texts and translations author Yoko Tawada, who writes both in Japanese and German.
- include safeguards for the protection of copyrights, to allow integration of data not in the public domain.

12. Conclusions

`LitText` is a suite of programs for researchers in DH to build multi-language corpora for projects as linked-data, and to explore them using the standardized SPARQL query language. The design goal was to minimize the amount of technical detail a DH researcher would need to use the system. The GUI should follow a conceptual model familiar to most DH researchers:

- build the corpus to be studied
- parts of text of interest to the research

Texts require minimal preparation with a simple markup language to add metadata about the text and identify the sections of text to be studied; multiple languages in a single file are possible, and the NLP processing is transparent to the user. The results are stored in a triple store, and can be queried with the standardized query language SPARQL. A DH researcher needs to learn a minimum set of NLP codes, e.g. a few Universal Dependency codes. Further processing of results is possible with spreadsheet software. Texts can be added anytime simply by adding files with markup to a directory. When queries are repeated, they search the increased corpus. The motto "all methods applied to all texts" (Ivanovic and Frank, 2015; Ivanovic and Frank, 2016) is respected: the full corpus can be reprocessed in a few days and the reevaluation of queries takes only minutes.
Technically, `LitText` revolves around a linked data triple store and the corresponding SPARQL update and query language. The main program takes a text file with markup, sends the pertinent text to the required, language-specific NLP processes, and converts the result to a triple format. Utilities to download text from research-specific sources and to automate markup as far as automatically possible are small extensions. Storage and query can be done with any SPARQL 1.1 conformant program; loading triples into triple stores and accessing the SPARQL endpoint uses the HTTP protocol and can be done in a web browser. Additional resources which are available as linked data can easily be incorporated (e.g. WordNet, Wikipedia). The use of standards makes a plethora of additional software available and reduces the number of utilities which must be specially programmed.

Acknowledgments

We thank an anonymous reviewer who suggested a CQP style query formulation (Hardie, 2012) which inspired improvements in the query.

Bibliographical References

Beckett, D. (2001). Raptor RDF parser toolkit. *University of Bristol.[Online]. Available: http://www. redland. opensource. ac*.

Beckett, D. (2014). RDF 1.1 -triples: A line-based syntax for an RDF graph. *W3C Recommendation*.

Berners-Lee, T., Bizer, C., and Heath, T. (2009). Linked data-the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22.

Blümm, M., Schmunk, S., Gietz, P., Horstmann, W., and Hütter, H. (2016). Vom Projekt zum Betrieb: Die Organisation einer nachhaltigen Infrastruktur für die Geisteswissenschaften DARIAH-DE. *ABI Technik*, 36(1):10–23.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Fischer, F., Göbel, M., Kampkaspar, D., Kittel, C., and Trilcke, P. (2017). Network dynamics, plot analysis: Approaching the progressive structuration of literary texts. In *Digital Humanities 2017 (Montréal, 8–11 August 2017). Book of Abstracts*.

Hardie, A. (2012). CQPweb – combining power, flexibility and usability in a corpus analysis tool. *International journal of corpus linguistics*, 17(3):380–409.

Harris, S., Seaborne, A., and Prudhommeaux, E. (2013). SPARQL1.1 query llanguage.

Hart, M. (1992). The history and philosophy of project gutenberg. *Project Gutenberg*, 3:1–11.

Holmes, D. I. and Kardos, J. (2003). Who was the author? an introduction to stylometry. *Chance*, 16(2):5–8.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., DiPersio, D., Shi, C., Suderman, K., Verhagen, M., Wang, D., and Wright, J. (2015). The language application grid. In *International Workshop on Worldwide Language Service Infrastructure*, pages 51–70. Springer.

Ivanovic, C. and Frank, A. U. (2015). Corpus-based research in Computational Comparative Literature. In Francesco Mambrini, et al., editors, *Proceedings of the Workshop on Corpus-Based Research in the Humanities (CRH). Warsaw, Poland*, pages 69–78.

Ivanovic, C. and Frank, A. U. (2016). Korpusanalyse in der computergestützten Komparatistik. In *Digital Humanities deutsch (DHd)*.

Ivanovic, C. (2017). Die Vernetzung des Textes: Im Möglichkeitsraum digitaler Literaturanalyse. *Zeitschrift für digitale Geisteswissenschaften*.

Khadilkar, V., Kantarcioglu, M., Thuraisingham, B., and Castagna, P. (2012). Jena-hbase: A distributed, scalable and efficient rdf triple store. In *Proceedings of the 2012th International Conference on Posters & Demonstrations Track-Volume 914*, pages 85–88. CEUR-WS. org.

Lord, G., Smith, M. N., Kirschenbaum, M. G., Clement, T., Auvil, L., Rose, J., Yu, B., and Plaisant, C. (2006). Exploring erotics in Emily Dickinson's correspondence with text mining and visual interfaces. In *Digital Libraries, 2006. JCDL'06. Proceedings of the 6th ACM/IEEE-CS Joint Conference on*, pages 141–150. IEEE.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Manola, F., Miller, E., McBride, B., et al. (2004). RDF primer. *W3C recommendation*, 10(1-107):6.

McCrae, J., Fellbaum, C., and Cimiano, P. (2014). Publishing and linking WordNet using lemon and RDF. In *Proceedings of the 3rd Workshop on Linked Data in Linguistics*.

Moretti, F. (2005). *Graphs, maps, trees: abstract models for a literary history*. Verso.

Palmero Aprosio, A. and Moretti, G. (2016). Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*, September.

Prud'Hommeaux, E., Seaborne, A., et al. (2008). Sparql query language for rdf. *W3C recommendation*, 15.

Pustejovsky, J., Ide, N., Verhagen, M., and Suderman, K. (2017a). Enhancing access to media collections and archives using computational linguistic tools. In *CDH@ TLT*, pages 19–28.

Pustejovsky, J., Verhagen, M., Rim, K., Ma, Y., Ran, L., Liyanage, S., Murdock, J., McDonald, R. H., and Plale, B. (2017b). Enhancing access to digital media: The language application grid in the HTRC data capsule. In *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact*, PEARC17, pages 60:1–60:3, New York, NY, USA. ACM.

Recasens, M., de Marneffe, M.-C., and Potts, C. (2013). The life and death of discourse entities: Identifying singleton mentions. In *HLT-NAACL*, pages 627–633.

Schmid, H. (2013). Probabilistic part-of speech tagging using decision trees. In *New methods in language processing*, page 154. Routledge.

Straka, M. and Straková, J. (2017). Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDpipe. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99.

Taylor, A., Marcus, M., and Santorini, B. (2003). The penn treebank: an overview. In *Treebanks*, pages 5–22. Springer.

Vanderwende, L., Menezes, A., and Quirk, C. (2015). An AMR parser for English, French, German, Spanish and Japanese and a new AMR-annotated corpus. In *Proceedings of NAACL-HLT*, pages 26–30.

Winiwarter, W. (2015). Jamred: a japanese abstract meaning representation editor. In *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*, page 11. ACM.

Xue, N., Bojar, O., Hajic, J., Palmer, M., Uresova, Z., and Zhang, X. (2014). Not an interlingua, but close: Comparison of English AMRs to Chinese and Czech. In *LREC*, volume 14, pages 1765–1772.

Zeng, J., Ruan, G., Crowell, A., Prakash, A., and Plale, B. (2014). Cloud computing data capsules for non-consumptiveuse of texts. In *Proceedings of the 5th ACM workshop on Scientific cloud computing*, pages 9–16. ACM.

Appendix

The common prefix values for SPARQL are:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX wn: <http://wordnet-rdf.princeton.edu/ontology#>
PREFIX nlp: <http://gerastree.at/nlp_2015#>
%PREFIX li: <http://def.seegrid.csiro.au/isotc211/iso19115/2003/lineage#>
prefix lemon: <http://lemon-model.net/lemon#>
prefix lit: <http://gerastree.at/lit_2014#>
prefix layout: <http://gerastree.at/layout_2017#>

%prefix dcterms: <http://purl.org/dc/terms/>
%prefix dc: <http://purl.org/dc/elements/1.1/>
%prefix pgterms: <http://www.gutenberg.org/2009/pgterms/>
%prefix marcrel: <http://id.loc.gov/vocabulary/relators/>
%prefix dcam: <http://purl.org/dc/dcam/>
%prefix cc: <http://web.resource.org/cc/>
```

Towards faithfully visualizing global linguistic diversity

Garland McNew[1], **Curdin Derungs**[2], **Steven Moran**[3]
[1]Department of Geography, San Diego State University
[2]University Research Priority Program Language and Space, University of Zurich
[3]Department of Comparative Linguistics, University of Zurich
garlandmcnew@gmail.com, curdin.derungs@geo.uzh.ch, steven.moran@uzh.ch

Abstract

The most popular strategy for visualizing worldwide linguistic diversity is to utilize point symbology by plotting linguistic features as colored dots or shapes on a Mercator map projection. This approach creates illusions due to the choice of cartographic projection and also from statistical biases inherent in samples of language data and their encoding in typological databases. Here we describe these challenges and offer an approach towards faithfully visualizing linguistic diversity. Instead of Mercator, we propose an Eckert IV projection to serve as a map base layer. Instead of languages-as-points, we use Voronoi/Thiessen tessellations to model linguistic areas, including polygons for languages for which there is missing data in the sample under investigation. Lastly we discuss future work in the intersection of cartography and comparative linguistics, which must be addressed to further advance visualizations of worldwide linguistic diversity.

Keywords: cartography, comparative linguistics, language diversity, data visualizations

1. The problem

There are approximately 7000 languages spoken in the world today and they are remarkably diverse in their sound systems, word formation strategies, and syntactic structures (Evans and Levinson, 2009). Samples of linguistic diversity are encoded in many different typological databases. A popular way of visualizing these datasets is to map linguistic features to categorical values and then to plot them as colored points on a Mercator map projection. An example from The World Atlas of Linguistic Structures (WALS; Dryer and Haspelmath (2013)) is shown in Figure 1 (Brown, 2013).

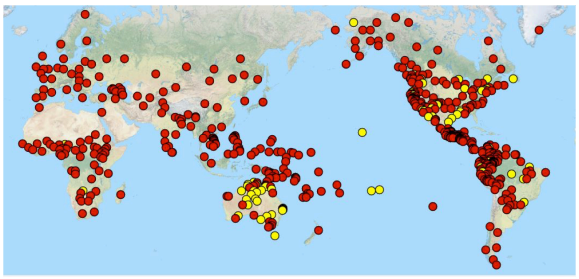


Figure 1: WALS Chapter 130: Finger and Hand

This and similar visualizations are unquestionably useful for doing exploratory analysis. Users can browse richly annotated digital maps that describe the known linguistic, cultural, and environmental diversity in the world (e.g. Kirby et al. (2016))[1] and researchers may use these visualizations to formulate hypotheses about linguistic and cultural phenomena, e.g. how they may have spread.

Moreover, many typological databases and digital atlases make their linguistic data and geographic coordinates openly available online. A fine example is Cross-Linguistic Linked Data (CLLD), which hosts over a dozen widely-used cross-linguistic comparative databases.[2] Hence the combination of increasingly easy-to-access tools for exploratory data analysis and access to the raw data has spurred hundreds of quantitative studies in comparative and historical linguistics. To highlight just two controversial studies that used WALS data, for example, consider whether linguistic tone has a genetic bias (Dediu and Ladd, 2007) and whether the worldwide distribution of phonemic diversity shows an out-of-Africa signal in ancient population movements (Atkinson, 2011).

Unfortunately there is a problem with the approach in visualizing global linguistic diversity described above: it creates potentially illusionary patterns that we – as humans and great pattern matchers – easily pick up on. These illusions are due to several factors that may misinform researchers if they are unaware of the model assumptions and statistical biases in the cartographic projection and linguistic data.

Let us consider for example Figure 1, Chapter 130 'Finger and Hand', which includes a global map displaying the distribution of the "two primary ways in which languages lexically treat the human finger and the hand of which it is a constituent" (Brown, 2013). The two values are either **identity** (a single word denotes both hand and finger) or **differentiation** (the language has separate words for 'finger' and 'hand', as in English). There are 72 languages of the identity type (marked with yellow) and 593 languages that mark differentiation (in red). The pattern of yellow versus red dots shows a strong visual signal that Australian languages are different than languages in the rest of the world

[1] https://d-place.org/

[2] These databases include, among others, WALS (Dryer and Haspelmath, 2013), The World Loanword Database (Haspelmath and Tadmor, 2009), The Atlas of Pidgin and Creole Language Structures (Michaelis et al., 2013), The database of the Automated Similarity Judgement Program (Wichmann et al., 2013), PHOIBLE (Moran et al., 2014), and Glottolog (Hammarström and Nordhoff, 2011). See: http://clld.org/datasets.html

(although there are some yellow dots in the Americas).
An explanation is proposed by Brown (2013), namely that "Farmers tend to lexically distinguish finger from hand more often than hunter-gatherers." This explanation is inline with what we know about the Aboriginal Australians – they were mostly hunter-gatherers, living in relative isolation from their initial settlement some 65kya (Clarkson et al., 2017), until the late 18th century (Pugach et al., 2013).
Testing whether hunter-gatherer languages are more likely to make a lexical distinction than agriculturalists is interesting because it provides a data point towards understanding whether there are cross-linguistic differences in languages spoken by hunter-gatherers and agriculturalists. But before we spend much time investigating the purported correlation in detail (or investigating one of the hundreds of other visual correlations in linguistic variables plotted as points on a Mercator map projection), we should consider that it is easy, and arguably natural, to draw conclusions from visual patterns in data, especially on maps.
In this paper, we highlight some of the important considerations when formulating a hypothesis from a linguistic atlas or when deciding to test a hypothesis inspired by one. We do so by first identifying the major cartographic and linguistic data factors that lead to illusions from plotting language data as points on a Mercator map projection. Then we provide a solution for faithfully visualizing global linguistic diversity.

2. Identify the illusions

One illusion is because typological datasets have for the most part poor cross-linguistic coverage. For example, Chapter 130 contains 593 data points, but there are more than 7000 languages spoken in the world today. Hence the map in Figure 1 shows only roughly 12% of the complete picture. This is a bibliographic bias, i.e. linguists are restricted to the accessible data about languages (Bakker, 2011). Around half of all languages are poorly described or undescribed (Hammarström, 2010). If there is no existing data on a particular language or language family, then that data cannot be used in descriptive studies about the distribution, of say, a given linguistic feature. At best it can be inferred through other means, such as language genealogy or known areal contact.
The bibliographic bias is exasperated by the cartographic problems involved with the Mercator map projection. A projection is a planar representation of a spherical object. That is, a map projection is an attempt to portray the surface of the earth onto a flat surface. The most common type of map projection is the Mercator projection, which was originally created in 1569 for nautical navigation purposes. Mercator accurately represents lines of constant course (rhumb lines). Mercator preserves directional accuracy and linear scale, however it distorts the area of objects as the latitude increases (north and south) from the equator due to the cylindrical nature of the projection. An example is given in Figure 2.
Why is this a problem? Well consider that Greenland is roughly equal in land mass to Mexico, but due to Mercator projection it is has a much larger depiction. Mercator should not be used for density visualization purposes.

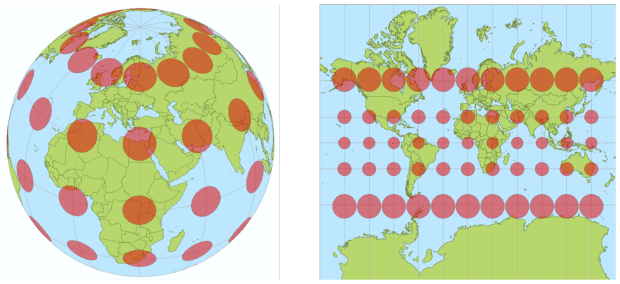


Figure 2: Mercator projection

Points are pulled or stretched away from one another visually when plotting them on a Cylindrical/Mercator type projection. Although positionally accurate, the areal distortion creates an erroneous perspective of density. Two points of equal distance will appear more clustered near the equator than they would near the poles. This is especially problematic in language diversity studies because the majority of the world's languages are spoken near the equator.
A second illusion comes from plotting languages as dots. Languages are not individual points, but are spoken by groups of people of varying population sizes and densities over different-sized and often overlapping geographic regions. What impact does a single dot for 'the English language' have, if someone compares its geographic distance with all other languages? Crucially, the **lack** of language points is also not indicated in typological atlases. That is, there is no visual cue for the absence of data. This creates the illusion that all data points displayed represent the population under investigation, when in fact it is often a skewed sample, e.g. the data points come from a convenience sample derived from well documented and described languages.
A third illusion is due to the use of colored dots or ranges of colored shapes (in the case of non-binary feature values). For example, consider the number of consonants in phonological inventories cross-linguistically. How do you visualize consonant inventory sizes? A different colored dot for each value would likely mask any patterns in visualizing the global data because consonant inventories range greatly in size, from a low of 6 to a high of 140. A world map with 134 colored values would be uninterpretable even if scaled by color. To handle this variation, one approach is to bin continuous values into ranges, e.g. small, medium, and large (Maddieson, 2013). But what statistical procedures went into determining these bins and how does their visualization change when the binning procedure is changed? These factors must be taken into account when interpreting visual patterns of global linguistic diversity. Furthermore, there are linguistic-specific factors of genealogical descent and areal contact that affect the distribution of linguistic features, for which there is currently no good cartographic solution – we return to this the issue in the Discussion Section.

3. Towards a solution

One solution to the problem of projecting density visualizations is to not use Mercator, but instead use an equal area

projection, such as Eckert IV, shown in Figure 3.

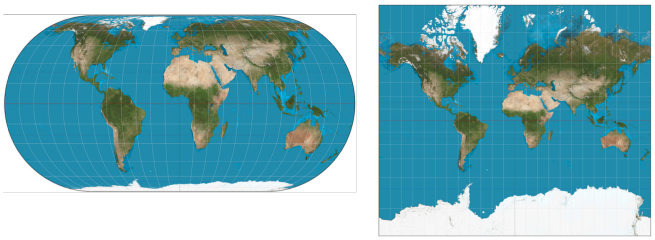


Figure 3: Eckert IV vs Mercator map projections

The Eckert IV projection preserves land mass and object size, but highly distorts lines of constant course. Since we are not concerned with maritime navigation, this projection is an acceptable compromise of distortion because it preserves land shape and area.

With a base map projection in place, we turn to the issue of languages-as-points. Linguistic visualizations often utilize point symbology to represent objects or features of interest, but languages are obviously spoken in areas and not in dots. Our solution is to create a Voronoi/Thiessen tessellation layer from the language locations (typically longitude and latitude coordinates). These polygons are generated from a set of sample points. Each Thiessen polygon defines an area of influence around its sample point, so that any location inside the polygon (Euclidian distance-wise) is closer to that point than any of the other sample points. Thus we no longer utilize the points to represent the language data, but instead the derived discrete polygons. An example is given in Figure 4.

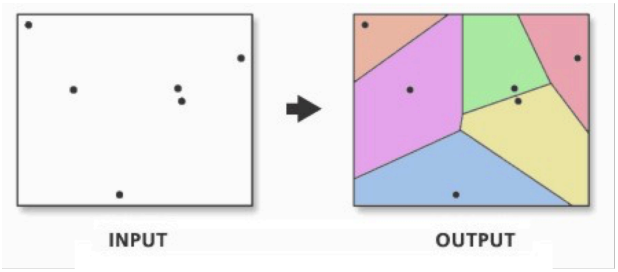


Figure 4: Conversion of point symbology to Thiessen polygons

However, one problem is that the entire geographic extent is accounted for when we create these polygons on a worldwide scale. That is, areas with low sampling density will have large polygons radiating far beyond a reasonable distance of influence. For example, consider Figure 5 and how the Thiessen tessellations grow disproportionately in areas of low linguistic diversity (data points in the tip of South America also radiates into Antarctica, where there are no native languages).

One approach to solving this issue is to create a dissolved buffer around each point that creates the polygons and then use this buffer layer to clip the Thiessen polygon layer, which we illustrate in Figure 6.

We use the dissolved buffer distances based on the data

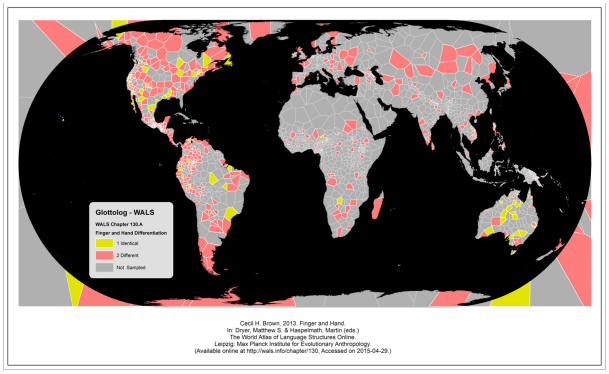


Figure 5: Worldwide Thiessen tessellations without a buffer

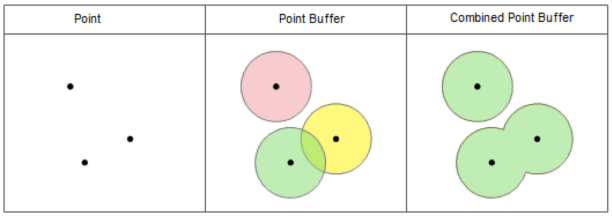


Figure 6: Dissolved point buffers

points provided by a particular typological database. For example in Figure 7, we apply a 200km buffer around each point and dissolve all of the buffers to account for the overlap of buffer zones.

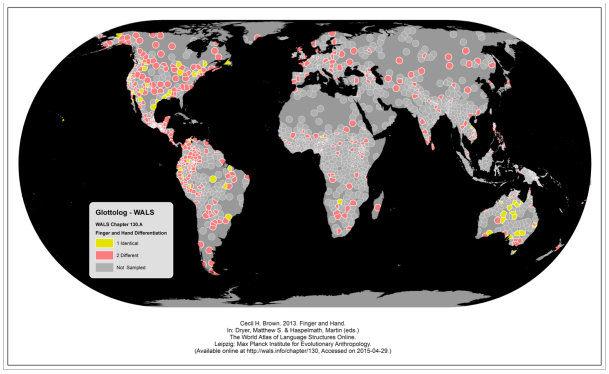


Figure 7: Worldwide Thiessen tessellations with buffer

In Figure 8, we provide an example in which we zoom in on the Americas. To address the illusion caused by missing data, the gray polygons show areas for which we know languages exist, but for which the input database is lacking information about the linguistic variable. Figures 7 and 8 show a global visualization of WALS Chapter 130 without its current cartographic illusions and linguistic biases. The yellow polygons show **identity**, the red polygons **differentiation**, and the gray polygons unknown values. Our approach presents a much less distorted picture of the reality of linguistic diversity and can be applied to any set of typological data.[3]

[3]In our initial research we used the geographic in-

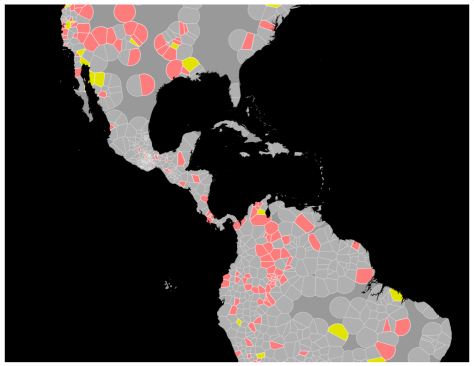


Figure 8: Zoomed-in Eckert IV projection with buffered Thiessen tessellations

4. Discussion

A popular strategy for visualizing worldwide linguistic diversity encoded in typological databases is to utilize point symbology to represent features of interest. We have shown here one way to overcome the most prominent biases in the linguistic data and cartographic projection that lead to visual illusions in global maps projecting linguistic diversity. Language data, however, also presents two additional challenges in visualizing worldwide linguistic diversity. When comparative or historical linguists investigate today's (synchronic) linguistic diversity, they are interested in the historic (diachronic) distribution of the world's languages due to how languages change over time. Language change involves to two factors: retention and contact.

Retention is the so-called **vertical** process of language change because it captures the idea that linguistic features are genealogically inherited from their parent language. Retention can be defined as the probabilistic likelihood of retaining a linguistic feature in descendant languages from the parent language. In other words, today's descendant languages have a high probability that they share features with their parent languages because they are genealogically related. An example is the Romance languages French, Italian, Spanish and Portuguese. They all share, to varying degrees, phonological and grammatical similarities with Latin.

The second type of language change is due to areal contact between speakers of different languages. These languages may or may not be genealogically related. For example, the lexicon of Brazilian Portuguese has long been influenced through contact with native South American languages. Beyond just borrowing words for new semantic concepts, language contact can lead to the borrowing of sounds and grammatical features (Matras, 2009).

Both retention and contact are implicit factors in the visualizations of language data from typological databases. However, they must be considered when evaluating patterns in maps of linguistic diversity because these factors are typically not overtly encoded. One procedure, employed by the CLLD, uses different colored shapes to indicate language family affiliation. However, this is only used when displaying the entire language sample as points globally or when a specific scalar variable is plotted on a map, e.g. the cross-linguistic distribution of the voiceless labiodental fricative [f].[4]

An alternative approach to visualize language genealogy is the application of the sunburst visualization (Stasko and Zhang, 2000) to the WALS and PHOIBLE databases (Mayer et al., 2014).[5] This application is successful in visualizing the hierarchical relatedness in language genealogy, but it does not address issues of geographic proximity and it offers nothing in particular for visualizing both known and unknown data points.

Our approach here offers a preliminary stab at the problem of contact by creating polygons that show which languages are likely to have been in contact. But it fails in discerning which languages are or were in contact (for this we need tertiary social information). Our approach also lacks a straightforward way to encode language relatedness beyond using point symbology or adding genealogical markers, such as coloring or border effects to the Thiessen polygon layer proposed here. Thus the problem of visualizing genealogical relatedness and areal contact between languages on a global scale is an area that needs more research.

5. Acknowledgements

Many thanks to Robert Forkel, Martin Haspelmath, André Skupin, and to participants at the workshops *Visualizing Linguistic Data* organized by the URPP Language and Space at the University of Zurich, and *Language Comparison with Linguistic Databases* at the Max Planck Institute of Evolutionary Anthropology in Leipzig, Germany. We also thank three anonymous reviewers for their feedback.

6. Author contributions

SM, GM designed the research. GM implemented the cartographic methods. CD created the open source code. SM, GM wrote the paper.

7. Bibliographical References

Atkinson, Q. D. (2011). Phonemic diversity supports a serial founder effect model of language expansion from Africa. *Science*, 332:346–359.

Bakker, D. (2011). Language sampling. In J. J. Song, editor, *Handbook of Linguistic Typology*. Oxford University Press, Oxford, UK.

Brown, C. H. (2013). Finger and hand. In Matthew S. Dryer et al., editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Clarkson, C., Jacobs, Z., Marwick, B., Fullagar, R., Wallis, L., Smith, M., Roberts, R. G., Hayes, E., Lowe, K.,

formation system software ArcGIS. We are now working on an open source version using R. The code is available on Github: `https://github.com/bambooforest/visualizing-typology-data`.

[4] `http://phoible.org/parameters/86E80A7DA80F7F58CA76F85BACA48F86#1/14/144`

[5] See: `http://www.th-mayer.de/wals/` and `http://tmayer.github.io/PhoibleVis/`.

Carah, X., et al. (2017). Human occupation of northern Australia by 65,000 years ago. *Nature*, 547(7663):306–310.

Dediu, D. and Ladd, D. R. (2007). Linguistic tone is related to the population frequency of the adaptive haplogroups of two brain size genes, ASPM and Microcephalin. *Proceedings of the National Academy of Sciences*, 104(26):10944–10949.

Dryer, M. S. and Haspelmath, M. (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Evans, N. and Levinson, S. C. (2009). The myth of language universals: language diversity and its importance for cognitive science. *Behavioral and Brain Sciences*, 32(05):429–448.

Hammarström, H. and Nordhoff, S. (2011). Langdoc: Bibliographic infrastructure for linguistic typology. *Oslo Studies in Language*, 3(2):31–43.

Hammarström, H. (2010). The status of the least documented language families in the world. *Language Documentation and Conservation*, 4:177–212.

Martin Haspelmath et al., editors. (2009). *WOLD*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Kirby, K. R., Gray, R. D., Greenhill, S. J., Jordan, F. M., Gomes-Ng, S., Bibiko, H.-J., Blasi, D. E., Botero, C. A., Bowern, C., Ember, C. R., et al. (2016). D-PLACE: A global database of cultural, linguistic and environmental diversity. *PloS one*, 11(7):e0158391.

Maddieson, I. (2013). Consonant inventories. In Matthew S. Dryer et al., editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Matras, Y. (2009). *Language Contact*. Cambridge University Press.

Mayer, T., Wälchli, B., Rohrdantz, C., and Hund, M. (2014). From the extraction of continuous features in parallel texts to visual analytics of heterogeneous areal-typological datasets. *Language Processing and Grammars. The role of functionally oriented computational models*, pages 13–38.

Michaelis, S. M., Maurer, P., Haspelmath, M., and Huber, M. (2013). *APiCS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Moran, S., McCloy, D., and Wright, R. (2014). *PHOIBLE Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Pugach, I., Delfin, F., Gunnarsdóttir, E., Kayser, M., and Stoneking, M. (2013). Genome-wide data substantiate holocene gene flow from India to Australia. *Proceedings of the National Academy of Sciences*, 110(5):1803–1808.

Stasko, J. and Zhang, E. (2000). Focus + context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *IEEE Symposium on Information Visualization*, pages 57–65. IEEE.

Wichmann, S., Müller, A., Wett, A., Velupillai, V., Bischoffberger, J., Brown, C. H., Holman, E. W., Sauppe, S., Molochieva, Z., Brown, P., Hammarström, H., Belyaev, O., List, J.-M., Bakker, D., Egorov, D., Urban, M., Mailhammer, R., Carrizo, A., Dryer, M. S., Korovina, E., Beck, D., Geyer, H., Epps, P., Grant, A., , and Valenzuela, P. (2013). *The ASJP Database (version 16)*. Max Planck Institute for Evolutionary Anthropology.

The GermaParl Corpus of Parliamentary Protocols

Andreas Blätte, Andre Blessing
University of Duisburg-Essen, University of Stuttgart
andreas.blaette@uni-due.de, andre.blessing@ims.uni-stuttgart.de

Abstract

This paper introduces the GermaParl Corpus. We outline available data, the data preparation process for preparing corpora of parliamentary debates and the tools we used to obtain hand-coded annotations that serve as training data for classifying debates. Beyond introducing a resource that is valuable for research, we share experiences and best practices for preparing corpora of plenary protocols.

Keywords: corpus creation, annotation, R, TEI, Digital Humanities

1. Introduction

Parliamentary debates convey the arguments, interpretations and disputes that shape political decision-making. They are recorded and transcribed by parliamentary administrations with diligence and are published as plenary protocols. These documents are available for long periods of time – for several decades and more – and they cover the full range of issues relevant to a political system. Plenary protocols are an indispensable resource for anybody interested in the politics and policies of a democratic polity. If citizens' access to plenary protocols is impeded, the norm of democratic transparency is violated. A corpus of plenary protocols is not just a language resource, it is a crucial building block of the public digital archive of democracy.

The value of plenary protocols for research goes beyond disciplines genuinely interested in the substance of parliamentary activity. These documents are a great resource for studying the variation of language across political domains, and language change in time. Given the amount of data, a corpus of plenary protocols is a good basis for testing all kinds of algorithms. Social scientists, computational and corpus linguists, as well as data scientists may use plenary protocols productively in their research.

There is a legal advantage to plenary protocols that deserves to be mentioned: There are no substantial legal barriers for using, processing and re-using of these documents. Restrictive licensing conditions that inhibit working with various kinds of media (including social media) do not arise with plenary protocols. Corpora of plenary protocols that have been prepared can be shared, which is an essential basis for attaining the ideal of reproducible research. And as plenary protocols are open data, these documents are an outstanding resource for teaching purposes.

The digital availability of plenary protocols is excellent and poor at the same time. Documents can be downloaded without technical or legal restrictions as txt, html or pdf documents. A minimally annotated XML may be available "off the shelf". But these data formats do not yet correspond to the requirements for digital-era data processing. To substantively exploit the analytical potential of the data, original documents need to be converted into a semi-structured data format (XML) with a sufficiently fine-grained markup. Most importantly, speakers and their affiliation to parliamentary groups and parties need to be annotated to attain a useful resource. The GermaParl corpus as it has been prepared in the PolMine Project [1] implements this notion and is based on an XMLification of documents in a standardized workflow.

Preparing corpora of plenary protocols is certainly an obvious idea. Thus, corpora of plenary protocols are not new in computational linguistics (Koehn, 2005)(Vinokourov et al., 2003). The development of machine translation systems has benefitted substantively from parliamentary data. In Europe, several projects have worked on preparing corpora of plenary protocols. A particularly important inspiration for GermaParl has been the DutchParl corpus (Marx and Schuth, 2010); parts of the language used (i.e. "GermaParl", and "XMLification") are inspired by the Dutch sister project. Indeed, to bring the family of projects working with and on corpora of plenary protocols into a dialogue, the European CLARIN consortium has initiated workshops to exchange approaches and experiences.[2].

One day, all parliaments all over the world might be represented in a GlobalParl corpus. But that is still a long way to go. Our contribution to the broader development is a corpus of the plenary protocols of the German Bundestag that is available in appropriately fine-grained XML. It is compatible with the standards of the Text Encoding Initiative (TEI) and it is prepared in a generic, reproducible workflow.

2. The GermaParl Corpus

The GermaParl Corpus includes all plenary protocols that were published by the German Bundestag between February 1996 and December 2016. GermaParl is not the only effort to create a corpus of debates in the Bundestag. But to the best of our knowledge, it is the most comprehensive one. The comprehensiveness and size of GermaParl – it comprises more than 100 million tokens – implies that the effort that can be invested in adding annotations and information beyond what can be extracted automatically had to be limited. A thematically specialized corpus such as the one prepared by Naomi Truan on the parliamentary discourse on Europe may offer significantly more detailed metadata and annotation (Truan, 2017).

GermaParl is made available in two ways:

[1] See `www.polmine.de`.

[2] CLARIN-PLUS Workshop "Working with Parliamentary Records", March 27-29 2017, Sofia, and Workshop "ParlaCLARIN" at 11th Language Resources and Evaluation Conference (LREC2018).

- The base format of the corpus is the XMLification of the raw data (i.e. the original protocols) that follows the TEI standard[3]. Releases of the TEI version of GermaParl are available at the PolMine presence at GitHub, in a repository called GermaParlTEI.[4]

- GermaParl is also disseminated as a R data package called 'GermaParl'. The package includes a linguistically annotated, indexed and consolidated version of the corpus that has been imported into the Corpus Workbench (CWB)[5] The R data package is designed to work smoothly with the analytical tools offered by the R package 'polmineR'[6].

Both variants of GermaParl are versioned. The version number of the TEI variant of the corpus is derived from the version number of the tool for corpus preparation, an R package to keep and maintain the code. The R data package has a different version number, but the documentation in the package will report which TEI version of the XMLified protocols has been used.

Plain text documents (txt files) issued by the German Bundestag were considered the best raw format for corpus preparation. Between May 2008 and March 2010, such txt files were not available. To fill the gap, pdf documents were processed for that period.

After implementing the corpus preparation workflow for txt and pdf documents, the German Bundestag has moved to offer XML versions of plenary protocols.[7] This official Bundestag XML actually is just plain text documents wrapped into an XML tag. Documents offer minimal metadata (legislative period, session number, date). However, the tricky part of corpus preparation is not extracting the very basic metadata of a protocol, but to attain a robust, consolidated annotation of speakers and agenda items. There is a benefit of switching to the "official XML" for corpus preparation, but it is minimal as long as plain text documents are available. Thus, the recent availability of minimal XML is not a challenge to our txt-based procedure.[8]

It is important to note that the GermaParl corpus is a collection of all debates and speeches actually given in the German Bundestag. Speeches that were included in the printed protocol, but not given in a parliamentary session, were not a part of corpus preparation.

[3]See `http://www.tei-c.org/index.xml`.

[4]See `https://github.com/PolMine/GermaParlTEI`.

[5]See `http://cwb.sourceforge.net/`.

[6]See `https://CRAN.R-project.org/package=polmineR`.

[7]See `http://www.bundestag.de/service/opendata`.

[8]Using the official Bundestag XML as the point of departure will however be a good choice for future versions of GermaParl, when we start processing protocols pre-dating 1996 (and the availability of and plain text raw data). The merit of the minimal official XML is that the potentially cumbersome task to extract plain text from OCRed pdf documents has already been performed by the Bundestag.

The raw corpus data is converted into a machine-readable XML data format. More specifically, we work with a variant of the TEI standards. The basis for that scheme was the existing TEI standard for performance texts.[9] Parsing a plenary protocol into the TEI standardization for performance texts is conceptually smooth, as we have speakers and functionally equivalent things happening on the stage (interjections in parliament) in both genres; the scenes in a drama are roughly equivalent to the agenda items in a plenary session. Yet there are also good reasons to use the TEI standard for transcribed speech as the template for plenary protocols (Truan, 2016). There would have to be an added value of discussing the pros and cons of the choice between potential templates, or of even developing a specialized one for plenary protocols, but we certainly admit that the current choice does not need to be the end of history.

The important fact about GermaParl is that the structural annotation of the corpus offers a broad range of possibilities to partition the corpus into subcorpora. The following information is included as basic metadata, inter alia:

- the legislative period;
- the number of the plenary protocol;
- the date of the plenary session;
- the raw data format (txt or pdf);
- the URL where the document was downloaded from.

The following table provides an overview of the number of protocols, and tokens included in the corpus by legislative period (LP).

LP	from/to	protocols	tokens
13	1996-1998	163	11.676.618
14	1998-2002	253	19.349.263
15	2002-2005	187	12.785.509
16	2005-2009	233	18.412.812
17	2009-2013	252	23.418.060
18	2013-2016	203	15.371.446

Table 1: Number of protocols and tokens by legislative period.

There is a considerable variation of the number of protocols and the number of tokens per legislative period. Looking at the breakdown of the number of tokens per year (Figure 1) conveys the reason for this. While we see an average of 4.8 million tokens per year, with peaks of more than 6 million tokens (in 2011 and 2012), the downswings of plenary activity follow the electoral cycle. In election years (1998, 2002, 2005, 2009 and 2013), there is routinely less plenary activity, and speeches: It takes time after an election to reconvene the newly elected Bundestag.

Corpus preparation departs from plain text without any markup, and the crucial task is to detect calls to agenda

[9]See `http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DR.html`, and `http://teibyexample.org/modules/TBED05v00.htm`.

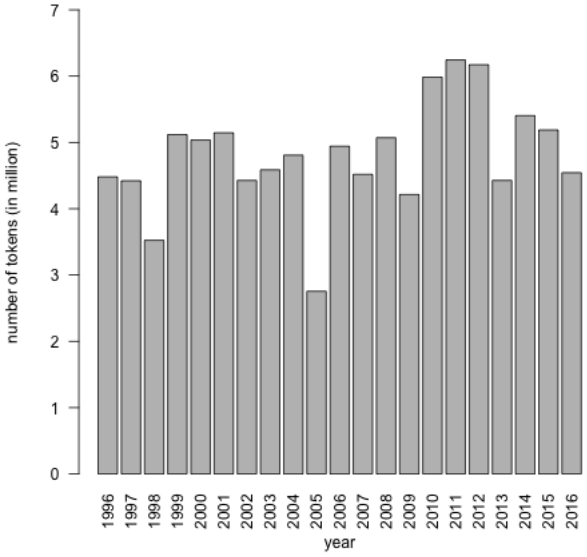


Figure 1: Number of tokens by year.

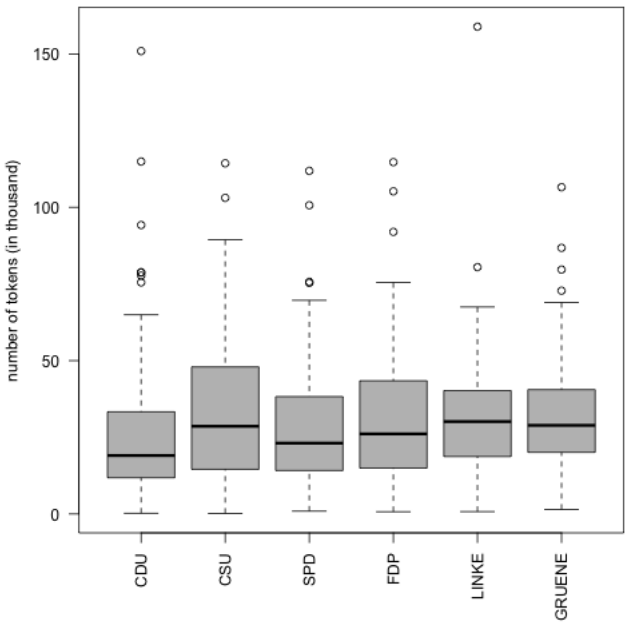


Figure 2: Boxplot – number of tokens of speakers by party (17th legislative period).

items, speakers, and to identify interjections. One important added value of the structural annotation of the corpus is the ability to distinguish between various speakers, their parliamentary groups and their parties. The following structural annotation is part of the corpus:

- the name of the speaker;
- the parliamentary group the speaker is affiliated with;
- the party affiliation of the speaker;
- whether an utterance is a speech or an interjection.

A crucial step to obtain a solid research resource is to have consolidated information at the speaker level. In the protocols, academic titles of speakers are not included in a fully consistent way, and names are sometimes reproduced with slight variations. Therefore, consolidation is necessary. Once this is achieved, it is possible to segment speeches, and to make statements about the variation between speakers of contributions to plenary sessions. The boxplot in figure 2 provides a comparison of the number of tokens contributed by individual speakers during the 17th legislative period of the German Bundestag, depending on party affiliation. The mean number of tokens spoken appears to depend on the size of the parliamentary group, i.e. the larger a parliamentary group, the smaller the average contribution of a parliamentarian. Of course the outliers raise interest. More than 150.000 words have been spoken in parliament by chancellor Angela Merkel (CDU), and the leader of the parliamentary group of *Die LINKE*, Gregor Gysi, a notoriously gifted (and fast) speaker.

Consolidating speaker information is the most time-consuming part of corpus preparation. Despite of all the efforts by parliamentary administrations, plenary protocols are not perfect. Remaining errors occur in the main body of the text, but they cause particular pain when inconsistencies occur in the names of speakers. Since it does happen, only applying regular expressions to extract speaker information is insufficient. Further consolidation is necessary. The following section outlines the workflow to obtain the consolidated corpus.

3. A Framework for Corpus Preparation

The long-term aim of the PolMine Project is to sustain corpora for all parliaments in Germany, including the 16 regional parliaments.[10] The German Bundestag is an important parliament, but it is one parliament among others. Thus, a workflow needs to be designed that is sufficiently generic to allow for a simple preparation of a corpus from any plenary protocol. Basically, the following three steps need to be carried out:

- *Preprocessing:* Prepare consolidated UTF-8 plain text documents (ensure uniformity of encodings, conversion of pdf to txt if necessary);
- *XMLification:* Turn the plain text documents into TEI format: Extraction of metadata, annotation of speakers etc.
- *Consolidating:* Check speaker names agains external data source and enriching documents with supplementary information.

[10] An early project to prepare corpora for the parliaments of the regional states of Germany was carried out in 2011/12 in cooperation with the Institut für Deutsche Sprache (IDS), see `http://www1.ids-mannheim.de/kl/projekte/korpora/archiv/pp.html`. The procedure we implemented at that time was not sufficiently object-oriented, making it difficult to customize the workflow for further parliaments, and hard to update the data.

In our case, the preprocessing step was not as trivial as it might seem. Older plain text files are offered by the German Bundestag in all kinds of encodings that are outdated. The pdf documents preserve a two-column layout that is difficult to manage. To handle these issues, respective functionality is included in the R packages 'ctk' (corpus toolkit)[11], and 'trickypdf'[12] that have evolved along with the corpus preparation tools to create GermaParl.

The core tool of the XMLification step is a set of regular expressions used to locate the beginning and the end of a debate, extract relevant metadata (such as legislative period, session number or date), to detect when speakers are called upon, and to identify the beginning and end of agenda items. The matches are used to generate the structural annotation of parliamentary speech in the XML document. However, due to the remaining haphazard variations that occur in plenary protocols – all standardization notwithstanding – we found it futile to strive for the set of universal regular expressions that would match correctly without any further document-specific interventions. Data quality is ensured in an iterative process that involves going back and forth between XMLifying all documents, and inspecting samples of the resulting data. To be able to quickly see wrong annotations, we found it useful to transform XML/TEI documents into html documents.

To handle those cases when a regular expression does not match in a desired fashion, and adjusting the regular expression might cause matches elsewhere that are not desired, we used two approaches:

- *False positives:* We run a a list of undesired matches. Before an annotation (of a speaker or an agenda item) is made, it is checked whether the match is on the mismatch list, and creating the annotation is suppressed, if it is.
- *False negatives:* There is a set of known errors (i.e. typos) in the original documents that inhibit the regular expressions to match. To ensure that the regex works, a factory of document-specific preprocessing functions is part of the code that performs (hard-coded) adjustments to make the regular expression match.

The list of mismatches and the document-specific preprocessing functions are maintained in an R package for corpus preparation that is hosted in a git repository, so that everything is under version control.

The result of the primary XMLification will still include noise. The inconsistencies that occur with names need to be handled with particular care. Accordingly, all the information that has been extracted is checked against an external data source. To be able to cope with noisy names in an automated fashion, an approximate string matching algorithm is used. To handle remaining difficulties to match names, a hand-written list of aliases is applied, and kept together with the code in the git repository, so that data quality can be improved iteratively.

[11] See `https://www.github.com/PolMine/ctk`,

[12] See `https://www.github.com/PolMine/trickypdf`.

There are alternative options for the external data source on parliamentary speakers. A classical source would be *Kürschners Volkshandbuch*, a book with biographical data on all parliamentarians that is published every legislative period (Holzapfel, 2018). Of course, the presentation of parliamentarians on the Website of the German Bundestag might have been considered. We opted for lists of members of parliament, cabinet members and further speakers that are available on Wikipedia.[13] This is not only because of easy digital access. The Wikipedia pages related to the German Bundestag are regularly and credibly updated by a dedicated team of volunteers. Wikipedia pages are undergoing continuous public scrutiny, ensuring permanent quality checks in a manner traditional printed material does not necessarily guarantee.

The framework for corpus preparation used for preparing GermaParl is intended to be fully generic. But of course, we need to allow for variation between parliaments, and parliaments may change layout and details of typesetting, so that alternative regular expressions may have to be used or methods to process specific details of the text are necessary. The appropriate approach to handle this is to implement things in a fully object-oriented fashion. Thus, we believe we have developed a framework for corpus preparation that might work for any parliamentary protocol. The code is included in an R package, and upon preparing a corpus, the version number of the package used is included in the TEI metadata. All interventions that may be necessary and that have been described (document-specific preprocessing functions, mismatch lists, lists with aliases) are kept as data in the R package. This way, corpus preparation is fully reproducible. This in turn is the precondition that new solutions to rectify data errors that are discovered when working with the corpus can be added, and to successively improve data quality.

4. Data Dissemination

The XML files of the corpus (according to the TEI standard) are available via a GitHub repository[14]. The data is somewhat large fo GitHub, and of course, the original purpose of git repositories has been to maintain code. But maintaining a (versioned) corpus in a git repository brings the advantage that versions can be compared, and that the effects of modifications of the corpus preparation procedure can be traced easily. What is more, GitHub repositories offer an issue tracker by default. The issue tracker of the GermaParlTEI repository is intended to organize the user feedback on data errors and on data quality.

For many users in the humanities and the social sciences, XML files in general and TEI files in particular will not be overly accessible: The technical barriers to entry may be considerable for the greatest potential user group of the corpus. Therefore, the GermaParl Corpus is also disseminated as an R data package. More specifically, it is linguistically

[13] For instance, see the list for the 17th Bundestag at `https://de.wikipedia.org/wiki/Liste_der_Mitglieder_des_Deutschen_Bundestages_(17._Wahlperiode)`

[14] `https://github.com/polmine/germaparltei`

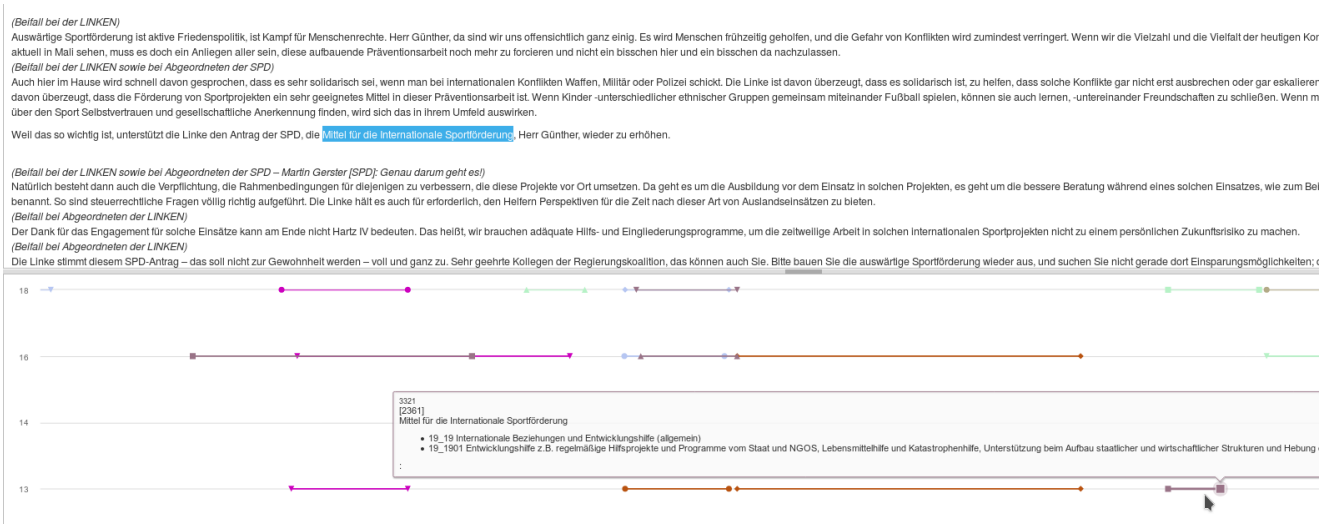


Figure 3: Comparing annotations of different annotators on the same document

annotated using standard NLP tools (Stanford Core NLP)[15] and imported into the Open Corpus Workbench (CWB)[16]. The indexed and compressed corpus is then wrapped into a R data package that comes with a documentation of the data (i.e. a vignette, in the R jargon).

It can be installed easily using polmineR, an R package specifically designed to process CWB indexed corpora[17]. Once polmineR is installed, three commands are enough to get started to work with GermaParl.

```
library(polmineR)
install.corpus("GermaParl")
use("GermaParl")
```

The license chosen for both variants of GermaParl is the license PUB+BY+NC+SA[18]. Thus, the data comes with the expectation that authorship is acknowledged. The commercial use of the data is not allowed (just as academic users see restrictions to work with commercial media data). Derivatives of GermaParl may not be licensed in a more restrictive manner than the GermaParl corpus we offer.

5. Web-based Annotation

Plenary protocols cover all kinds of topics. This is a big advantage. At the same time, the thematic variety of GermaParl is a problem. Often, researchers will wish to work with a thematically defined subcorpus that fits their specific research interests. An unsupervised clustering approach (such as topic modelling) would be a relatively quick way to generate the basis for topic-specific subcorpora. To move beyond unsupervised learning, we generated training data based on a classification scheme to serve as an input for machine learning algorithms.[19]

To support the preparation of training data, we created a web-based annotation tool which offers the functionality to annotate phrases of parliamentary speeches using predefined categories. The tool contains a backend which is used to select speeches for annotation via stratified sampling, thereby ensuring a balanced sample of speeches to be annotated, i.e. that all years and parliamentary groups are represented.

Our annotation scheme and the accompanying guidelines are based on the scheme of the Comparative Agendas Project (CAP, `http://www.comparativeagendas.net`), a large international project to trace changing policy agendas of governments. The CAP classification scheme was originally developed for US parliamentary data. Following the guidelines of the CAP project, we extended the scheme slightly with new categories to fit the German / European context. For instance, we needed to introduce a category for debates on various aspects of European integration. The extended CAP annotation scheme contains 24 major categories and 209 subcategories.

The annotation process was realized by 5 annotators. In a preliminary step, 20 speeches were annotated by all annotators. Two reasons speak for this: (a) to train the annotators; (b) to refine the guidelines. To check intercoder reliability, we worked with a special view to simplify the comparison of the results of different annotators, ultimately helping to identify challenging cases (Figure 3).

Each of the 4 rows at the bottom depicts one annotator. The x-axis represents the word sequence of one speech. Each

[15] See `https://stanfordnlp.github.io/CoreNLP/`.

[16] See `http://cwb.sourceforge.net`.

[17] Available at CRAN at `https://CRAN.R-project.org/package=polmineR`, development version at GitHub, see `https://github.com/polmine/polmineR`.

[18] See `https://www.clarin.eu/content/license-categories`.

[19] The context for this annotation and classification exercise was the CLARIN curation project "Plenarprotokolle als öffentliche Sprachressource der Demokratie: Klassifikation von Plenardebatten im PolMine-Plenarprotokollkorpus" (2015/16). On this occasion, we would like to thank Pawel Szczerbak, Lena Rickenberg, Vanessa Molter and Laura Dinnebier for their contribution to the project.

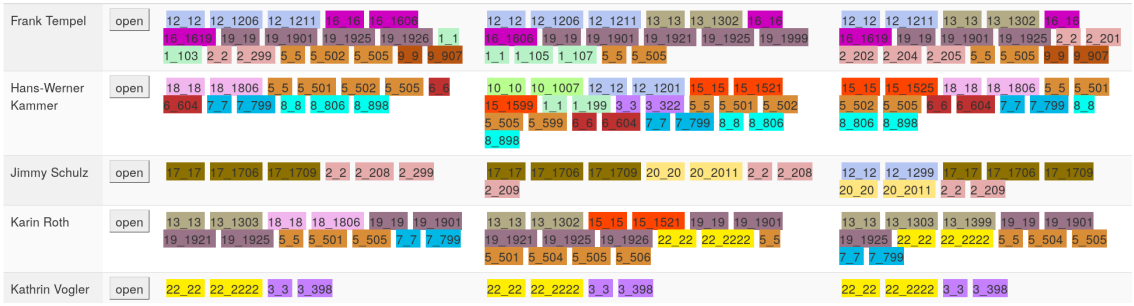


Figure 4: Aggregated annotation comparison of three annotators for five speeches.

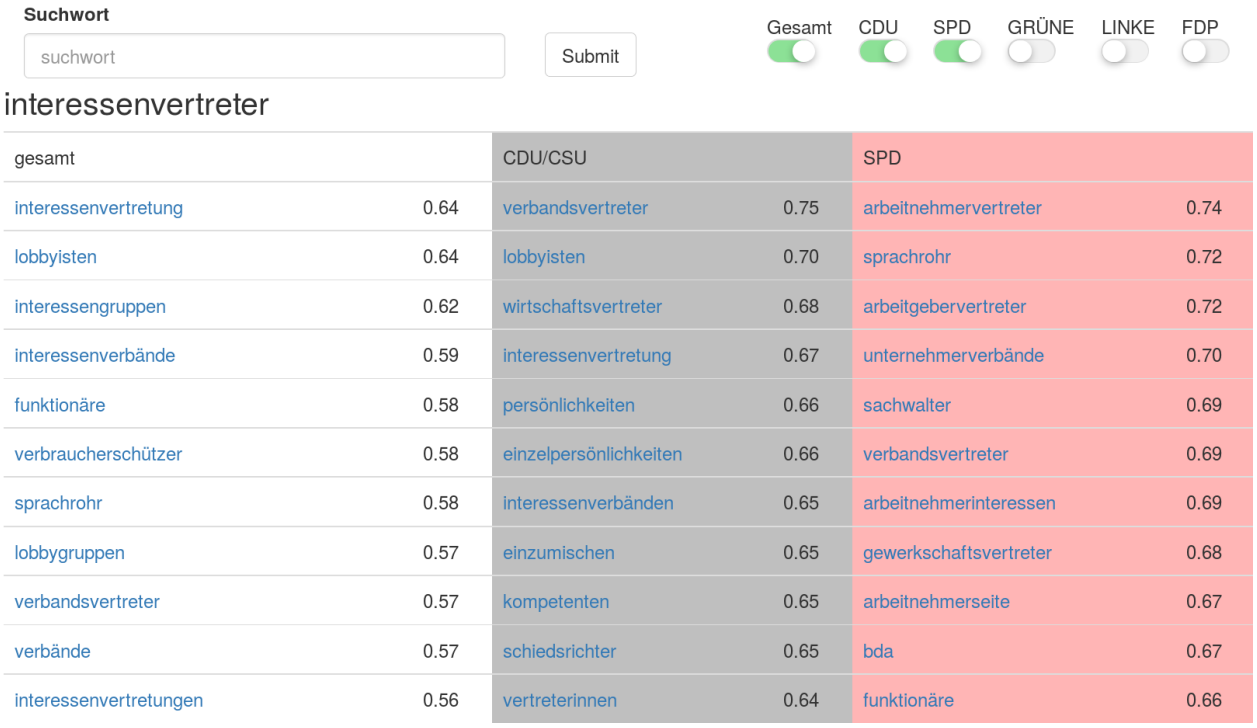


Figure 5: Word embeddings trained on subsets of GermaParl (divided by parliamentary groups).

colored span in the line shows one annotation. Such colored spans can be clicked on to get additional information: The corresponding text is highlighted in the above textual view of the speech; all assigned categories are listed for the selected annotation.

In total 611 speeches were annotated. Two general issues emerged during the annotation process. First, unambiguous instructions for coders how to handle the textual boundaries of an annotation are hard to operationalize. Annotations remain a mixture of words, sentences, and paragraphs. Second, recurring topics are also hard to model. Table 2 gives an overview of the categories most frequently annotated in the corpus.

A further overview of annotated categories for each document, separated by annotators, gives additional insights about the distribution of categories in the corpus.

id	category	freq
1	Domestic Macroeconomic Issues	1404
20	Government Operations	1291
13	Social Welfare	1179
5	Labor and Employment	983
19	International Affairs and Foreign Aid	973
15	Banking Finance, and Domestic Commerce	942

Table 2: Annotation categories ordered by frequency.

6. Example Applications

The purpose of GermaParl is to serve as a sound, trustworthy basis for research in the social sciences, the humanities, computational linguistics and information science. In a paper on the parliamentary activity of parliamentarians with a migration background, GermaParl served as a data basis for measuring the salience of migration and integration issues in speeches given by parliamentarians with and without a migration background (Blätte and Wüst, 2017). The prerequisite for this research was the meticulous consolidation of speaker names in the corpus. To measure issue salience, the paper pursues a dictionary-based approach, i.e. shows how the semantic field of migration and integration can be defined in a corpus-driven manner. In another paper, Blätte

uses machine learning to blindly guess the party affiliation of speakers (Blätte, 2018) based on a model trained on the annotated party membership in the corpus. In this case, classification errors are interpreted as likelihood of confusion, and political proximity of parties.

The design of the GermaParl corpus will enable researchers to implement all kinds of research questions. The corpus can be easily split by speakers, or by parliamentary groups. To explore the potential use of a recent technique in information science that requires sizeable corpora, we used such subcorpora to train different word embedding models[20]. Figure 5 shows how such embeddings can be used to investigate how terms (e.g. "Interessenvertreter" [lobbyist]) are used by different parliamentary groups. E.g.: "Verbandsvertreter" (association representatives) is the most closely related term by the Christian Democratic group, and "Arbeitnehmervertreter" (employee representatives) by the Social Democratic group. This is just one example how language use varies between parliamentary groups.

Many further uses of the corpus concern language change over time, variations or party positions etc. We are happy to offer a resource that may lower the barriers of entry to work with parliamentary protocols productively.

7. Conclusion

The current version of GermaParl is a 100-million-token corpus. The XML/TEI variant is available at a GitHub repository, a linguistically annotated and indexed version can be installed as a R data package. The data is intended to be open, versioned, reproducible, accessible and sustainable, with a focus on successively improving data quality.

However, the focus of our endeavor is not just to offer the data itself, and the annotations of parliamentary speeches. The ultimate aim of the project is to develop a generic workflow and a framework for preparing corpora of parliamentary protocols. Research on policies, politics and language change in parliamentary democracies will benefit from a public digital archive of democracy. Plenary protocols are an important part of this archive. We hope that GermaParl makes a useful contribution to a growing family of corpora of plenary protocols.

8. Bibliographical References

Blätte, A. and Wüst, A. M. (2017). Der migrationsspezifische Einfluss auf parlamentarisches Handeln. Ein Hypothesentest auf der Grundlage von Redebeiträgen der Abgeordneten des Deutschen Bundestags 1996 bis 2013. *Politische Vierteljahresschrift*, 58(2):205–233.

Blätte, A. (2018). Zum Verwechseln ähnlich? Eine Klassifikationsanalyse parlamentarischen Diskursverhaltens auf Basis des PolMine-Plenarprotokollkorpus. In Joachim Behnke, et al., editors, *Computational Social Science. Die Analyse von Big Data*. Nomos, Baden-Baden.

Holzapfel, A. (2018). *Kürschners Volkshandbuch Deutscher Bundestag. 19. Wahlperiode*. Neue Darmstädter Verl.-Anst., Rheinbreitbach.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Marx, M. and Schuth, A. (2010). Dutchparl. the parliamentary documents in dutch. In Nicoletta Calzolari, et al., editors, *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Truan, N. (2016). Corpus Annotation. Representations of the Other in the British, French and German Discourse on Europe: A Corpus-Based Contrastive Discursive Analysis. `https://repository.ortolang.fr/api/content/de-parl/3/Corpus%20Annotation.pdf`. Accessed: 2018-02-22.

Truan, N. (2017). Parliamentary Debates on Europe at the Bundestag (1998-2015). ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

Vinokourov, A., Cristianini, N., and Shawe-Taylor, J. (2003). Inferring a semantic representation of text via cross-language correlation analysis. In *Advances in neural information processing systems*, pages 1497–1504.

[20] We used the word2vec implementation (Mikolov et al., 2013).

Identifying Speakers and Addressees in Dialogues Extracted from Literary Fiction

Adam Ek, Mats Wirén, Robert Östling, Kristina N. Björkenstam, Gintarė Grigonytė & Sofia Gustafson Capková
Department of Linguistics
Stockholm University
SE-106 91 Stockholm
{adam.ek, mats.wiren, robert, kristina.nilsson, gintare, sofia}@ling.su.se

Abstract

This paper describes an approach to identifying speakers and addressees in dialogues extracted from literary fiction, along with a dataset annotated for speaker and addressee. The overall purpose of this is to provide annotation of dialogue interaction between characters in literary corpora in order to allow for enriched search facilities and construction of social networks from the corpora. To predict speakers and addressees in a dialogue, we use a sequence labeling approach applied to a given set of characters. We use features relating to the current dialogue, the preceding narrative, and the complete preceding context. The results indicate that even with a small amount of training data, it is possible to build a fairly accurate classifier for speaker and addressee identification across different authors, though the identification of addressees is the more difficult task.

Keywords: literary corpora, speaker identification, addressee identification, quote attribution

1. Introduction

During the last few years, quantitative approaches to literary analysis have increasingly progressed from stylistic problems to higher-level phenomena such as plot, community structure and interaction between protagonists. One example of this is the recent interest in constructing social networks from literary fiction, either manually (Moretti, 2011; Agarwal et al., 2012; Yeung and Lee, 2016; Vala et al., 2016) or using automatic methods (Newman and Girvan, 2004; Elson et al., 2010; Rydberg-Cox, 2011). Typically, the goal has been to mirror relations between entities or events extracted from the text as a whole. Arguably, however, a more fine-grained perspective can be obtained by studying the direct speech between characters separately from the narratives in which the speech is embedded. Direct speech, in literary fiction usually framed by devices such as dashes, quotation marks and paragraphs, can be seen as the lowest level of narrative transmission (Koivisto and Nykänen, 2016). In this sense, it provides an independent level in which the relations between characters can be studied, including phenomena such as stance and sentiment as expressed through (the rendering of) the characters themselves.

To properly analyse dialogue interactions, we need to identify both the speakers and the addressees in occurrences of direct speech. The former problem, also known as quote attribution, has been explored in literary fiction by, among others, Elson et al. (2010), O'Keefe et al. (2012), He et al. (2013) and Muzny et al. (2017). As far as we know, however, the problem of identifying addressees (liistenerns) in literary fiction has previously only been dealt with by Yeung and Lee (2017).

For the purpose of specifying our method, we make the following assumptions, aimed at covering differing author styles. We refer to a sequence of direct speech interactions as a *dialogue*; this consists of one or more *turns*, each of which we assume is associated with one speaker and one or more addressees. A turn consists of one or more *lines* (framed by dashes in the example in Figure 1), and a line consists of one or more utterances. Literary fiction consists of alternating dialogues and instances of narrative structure, the latter of which we refer to as *narratives*. In addition, we refer to the entire text before a dialogue (narratives as well as other dialogues) and back to the beginning of a chapter as the *global context*.

This is exemplified in Figure 1, which shows a narrative with a subsequent dialogue turn. Figure 1 also illustrates

> Olle very skilfully made a bag of one of the sheets and stuffed everything into it, while Lundell went on eagerly protesting.
>
> When the parcel was made, Olle took it under his arm, buttoned his ragged coat so as to hide the absence of a waistcoat, and set out on his way to the town.
>
> – He looks like a thief, said Sellén, watching him from the window with a sly smile. – I hope the police won't interfere with him! – Hurry up, Olle! he shouted after the retreating figure. Buy six French rolls and two half-pints of beer if there's anything left after you've bought the paint.

Figure 1: Example narrative and dialogue turn translated from our Swedish data (from Chapter 6 of August Strindberg, *The Red Room*, 1879). The first two paragraphs constitute a narrative. The third paragraph is a turn consisting of three lines, each of which is marked by a dash, with Sellén as speaker. In the first line, the speaker is explicitly tagged ("said Sellén"); in the second, the speaker is implicit; and in the third line, the speaker is anaphoric ("he shouted"). The three lines have two distinct addressees (Lundell, Lundell, and Olle, respectively). A sequence of turns uninterrupted by narratives constitutes a dialogue.

the three ways of signalling the identity of a speaker that we distinguish:

1. Explicit speaker: a speech tag consisting of a speech verb and an explicit name ("said **Sellén**").
2. Anaphoric speaker: a speech verb and an anaphoric expression in the form of a pronoun ("**he** shouted") or a definite description ("said **the angry man**").
3. Implicit speaker: none of the above; the speaker must be inferred from the previous lines, preceding dialogue, preceding narrative and/or global context.

Analogously, we distinguish three ways in which the identity of an addressee can be signalled:

1. Explicit addressee: the name of the addressee is mentioned explicitly ("Hurry up, **Olle**!").
2. Anaphoric addressee: the addressee is referred to with a pronoun ("...after **you**'ve bought the paint") or a definite description (...he shouted after "**the retreating figure**").
3. Implicit addressee: none of the above; the addressee must be inferred from the previous lines, preceding dialogue, preceding narrative and/or global context.

As mentioned above, we assume that a turn has a single speaker. As illustrated by the example in Figure 1, however, different lines within a turn may have different addressees. Also, a speaker may address more than one person simultaneoulsy, which means that one line may have several addressees.

2. Previous Work

Among the first papers to consider quote attribution applied to literary fiction is Elson et al. (2010). They use a variety of supervised machine learning approaches (JRip, J48 and Logistic Regression) to assign a speaker to quotes. The system extracts candidates from the surrounding text and for each quote the system selects the most likely speaker. The quotes are divided into seven syntactical categories corresponding to different manners in which speakers are indicated.

An SVM-ranking approach to the problem was used in He et al. (2013). Unlike Elson et al. (2010), the candidates were extracted during the preprocessing step, and for each quote the system selects the most likely speaker from the set of candidates. Each candidate is assigned a set of features capturing the turn-taking, dependency relations, name and gender matching, character frequencies, distances to the utterance and mentions in the quote. Also, an unsupervised topic-actor model was used as a feature.

Recently, a sieve approach was applied to the problem by Muzny et al. (2017). The approach determines the speakers in two steps, first candidate speakers of each quote is identified in the text. Secondly, from the candidate speakers the most likely speaker is selected. The sieves used in determining candidates capture dependency relations, mention recency, the turn-taking heuristic and mentions in and around the quote. To selecting a speaker from the candidates, co-reference resolution, name matching, the turn-taking heuristic and mentions in the quote are used.

The only work aimed at identifying addressees that we are aware of is Yeung and Lee (2017).[1] This uses a CRF sequence labeling algorithm such that for each quote, the two surrounding sentences are extracted. Each word in the extracted sentences is then assigned a feature set containing the part-of-speech tag, dependency relations, distances to the quote, and matches in the line. Each word is then classified as "speaker", "listener" or "neither" by the system.

3. Data

In this section, we describe the data set used and how the data set was annotated.

3.1. Overview

The data used in the experiments reported here consists of parts of four novels by different authors: August Strindberg, *The Red Room* (1879; obtained from the National Edition of August Strindberg's Collected Works, published in 1981); Hjalmar Söderberg, *The Serious Game* (1912); Birger Sjöberg, *The Quartet That Split Up*, part I (1924); and Karin Boye, *Kallocain* (1940). Table 1 specifies the total number of dialogues and lines which have been annotated, and how they make up the training and test set, and the development set. The development set consists of Chapters 1 and 21 from *The Red Room* by August Strindberg, whereas all the remaining chapters are included in the training and test set. The distribution of chapters, dialogues and lines in the training and test set across the four novels is shown in Table 2.

In total, the test and training corpus consists of 822 lines distributed over 268 dialogues. Specifically, for each turn we annotated each line with its speaker, its addressee or addressees (compare Section 3.2.), and an indicator for the ways in which the identity of the speaker and the addressee were signaled as described in the previous section (explicit, anaphoric or implicit). Table 3 shows the variation of indicators for speakers across the authors. Furthermore, the variation of indicators for addressees is shown in Table 4.

CORPUS	DIALOGUES	LINES
Training and test	268	822
Development	23	75

Table 1: Number of dialogues and lines in the annotated corpus.

As shown in Tables 3 and 4, both speakers and addressees are mostly referred to implicitly in our data. When this is not the case, however, speakers are more commonly referred to explicitly, whereas addressees are more commonly referred to anaphorically, mostly with pronouns.

[1] Strictly speaking, Yeung and Lee (2017) take the goal to be to identify listeners. As exemplified by the last line in Figure 1, however, the addressees (the intended recipient or recipients of an utterance) may be a subset of the listeners (the people overhearing the utterance).

CORPUS	CHAPTERS	DIALOGUES	LINES
Strindberg	4	93	393
Sjöberg	10	82	216
Söderberg	2	37	93
Boye	5	56	121
All	21	268	822

Table 2: Number of dialogues and lines in the training and test data.

AUTHOR	EXP	IMP	ANA-P	ANA-D
Strindberg	86	285	20	2
Sjöberg	117	67	10	21
Söderberg	26	52	15	0
Boye	21	44	56	0
All	250	448	101	23

Table 3: Indicators for speaker identity across the authors. EXP = explicit; IMP = implicit; ANA-P = anaphoric, pronoun; ANA-D = anaphoric, definite description.

AUTHOR	EXP	IMP	ANA-P	ANA-D
Strindberg	31	192	133	37
Sjöberg	34	160	5	16
Söderberg	9	67	16	1
Boye	9	74	29	9
All	83	493	183	63

Table 4: Indicators for addressee identity across the authors. EXP = explicit; IMP = implicit; ANA-P = anaphoric, pronoun; ANA-D = anaphoric, definite description.

Different authors and print editions use different conventions for framing turns and lines in dialogues, such as dashes, quotation marks or angle brackets, thus delimiting the speech in different ways. For example, the first line in Figure 1 might have been rendered as

> "He looks like a thief", said Sellén, watching him from the window with a sly smile.

We use a script to to normalize these different conventions into a format using dashes as shown in Figure 1 .

3.2. Annotation

The data was annotated by two of the authors. The data consists of raw text with the annotations being inserted tags indicating where a line ends, containing who the speaker is and who the addressee is, followed by in which way these are indicated. The components of the annotation tag are the following:

1. `<speaker--addressee>`
2. `<type_speaker--type_adressee>`

Where 1 is always followed by 2. Using Figure 1 as an example, the annotations are the following:

> – He looks like a thief, said Sellén, watching him from the window with a sly smile. `<Sellén--Lundell><EXP--IMP>`
> – I hope the police won't interfere with him! `<Sellén--Lundell><IMP--IMP>`
> – Hurry up, Olle! he shouted after the retreating figure. Buy six French rolls and two half-pints of beer if there's anything left after you've bought the paint. `<Sellén--Olle><ANA--EXP>`

The start of each line in a turn is indicated by a dash and the annotation is inserted at the end of the line. We have only annotated lines where there are a clear speaker and addressee. Cases in which the same character is both the speaker and addressee have not been annotated. If the addressee is a group of people it is annotated as "SEVERAL". For addressees, there may be conflicts between different tags as in the last line of Figure 1, where Olle may be annotated as explicit or as a definite description. In these cases explicit mentions supersede anaphoric mentions, for anaphoric mentions pronouns supersede definite descriptions.

4. Method

In this section, we describe the task to be performed, how we will perform the task and which features we have extracted from the text.

4.1. Task

Identification of speakers and addressees is realized as a sequence labeling task. For each chapter, a precompiled list of the characters appearing as speakers or addressees, along with their known aliases, is provided to the system. For each line in a dialogue, the system selects the most likely character from the character list.

A text is considered as a sequence of paragraphs and dialogues. A dialogue consists of n turns, each of which contains one or more lines. We consider each line as an independent unit with a speaker and an addressee label assigned to it. The task is to find the sequences of speakers and addressees that are most likely given the dialogue.

A variety of algorithms have been applied to sequence labeling tasks. For the current task the averaged perceptron (Collins, 2002) has been selected, due to its good performance and the efficient implementation it permits.[2]

4.2. Features

The features are based on information from the dialogue, the preceding narrative and the global context. Since we consider the task as a sequence labeling task, the previously selected speakers and addressees are also considered as features. The features are presented in Table 7, where each feature is binary.

Mention in Line: If a character is mentioned in a line, that character is likely relevant to the current line is some manner. The character mentions are captured for the current line by feature 1, and for the two preceding lines by features 2 and 3.

With Speech Verb in Line: Authors may indicate the speaker of a line explicitly by using their name with a

[2]Our implementation is freely available at `https://github.com/adamlek/dialogue-fiction`.

ID	FEATURE	INFORMATION SOURCE
1	c_i mentioned in l_j	Dialogue
2	c_i mentioned in l_{j-1}	Dialogue
3	c_i mentioned in l_{j-2}	Dialogue
4	c_i + speech verb in l_j	Dialogue
5	c_i + speech verb in l_{j-1}	Dialogue
6	c_i + speech verb in l_{j-2}	Dialogue
7	c_i + speech verb/mentioned in l_j	Dialogue
8	c_i + speech verb/mentioned in l_{j-1}	Dialogue
9	c_i + speech verb/mentioned in l_{j-2}	Dialogue
10	$c_i = x_{i-k}$ $(k = 1 \dots 6)$	Dialogue
11	c_i mentioned in narrative	Narrative
12	$count(c_i) = 0$ in narrative	Narrative
13	$0 < count(c_i) \leq 2$ in narrative	Narrative
14	$count(c_i) > 2$ in narrative	Narrative
15	$count(c_i) \geq 5$ in global context	Global context
16	$count(c_i) \geq 15$ in global context	Global context
17	c_i is n:th most recent mention	Global context
18	c_i is 0:th mention*	Global context
19	c_i is 0:th mention+speech verb in l_j*	Global context

Table 5: Feature templates used in the identification of speakers. l_j indicates the current line and features marked with an asterisk (*) are only used if the line contains an anaphoric pronoun.

speech verb. In these cases it is certain that character c is the speaker of the current line. Characters which occur with a speech verb in the current line is captured by feature 4, occurrences with speech verbs in two preceding lines are captured by feature 5 and 6.

Mention and Speech Verb in Line: Features 1 to 6 are usually strong indicators of participation in the dialogue. Given this, we combine the mention and speech verb features into one. Mentions and speech verbs for the current line is captured in feature 7. Features 8 and 9 capture mentions and speech verbs for the two preceding lines.

Hypothesis: A common heuristic applied to dialogues is that of turn-taking. The turn-taking heuristic states that in a dialogue between two characters, one character will occupy lines $l_j, l_{j-2} \dots$ and the other character $l_{j-1}, l_{j-3} \dots$. This pattern is captured by feature 10, which matches previously selected characters to the current character. The turn-taking is violated from time to time in the data, as such this heuristic is not implemented as a hard constraint.

Mention in Narrative: If a character is mentioned in the narrative of a dialogue, the character most likely has some relevance for the dialogue. We capture mentions in the narrative with feature 11.

Frequency in Narrative: The raw frequency of the characters in the narrative is captured by feature 12, 13 and 14. These features check (1) if the character is mentioned zero times, (2) if the character is mentioned one or two times and (3) if the character is mentioned more than two times.

Frequency in Global Context: In addition to the character frequency in the narrative, two features (15 and 16) capture the raw frequency of the character in the chapter at the current dialogue. We consider two thresholds, determined from the development set, (1) if the character occurs five or more times and (2) if the character occurs more than 15 times. The intuition behind these features is to capture import characters from a larger context.

Mention Order: The order in which the characters are mention is considered an import factor, where recently mentioned characters are likely participants of the current dialogue. A list is compiled from the order in which the characters are mentioned, where the first character is the most recent mention. The index of the current character is captured by feature 17.

Pronoun: Anaphoric pronouns appear in the lines, both with and without speech verbs. Two features (18 and 19) are designed to deal with them. Feature 18 is true if the current character is the most recent mention character and there is a pronoun in the line. Feature 19 is true if the current character is the most recent mention and the pronoun occurs with a speech verb.

For speaker identification, when a line contains a character c with a speech verb, we constrain the search to only consider hypotheses where the speaker of that line is c. This amounts to treating feature 4 as a hard constraint.

Our features are modeled in such a way to only capture information which has been given previously, e.g. we capture no information that appear after the current line.

4.3. Training and Evaluation

As mentioned in Section 4., training is performed using an averaged structured perceptron (Collins, 2002). A beam search with beam size 10 is used to keep several possible character sequences as the hypothesis. The hypothesis with the highest score is selected as the character sequence for the current dialogue.

The model's performance is estimated using cross-validation with the authors as folds, since we are primarily interested in the extent to which the features generalize across different author styles. The results are compared against three baselines for speakers and addressees, respectively:

1. Random baseline: For each dialogue, two characters are selected randomly, and are distributed in an alternating pattern across the lines.

2. Latest mentions with speech verb: The two latest characters that occurred with a speech verb are distributed over the lines in an alternating pattern (compare below). This corresponds to the baseline used by O'Keefe et al. (2012).

3. Latest mentions: The two latest mentioned characters are distributed in an alternating pattern across the lines.

For the second and third baselines, if there are less than two characters satisfying the conditions, the remaining characters are generated randomly. Also, the latest character to satisfy the conditions is designated as the first speaker and the second addressee, and the second character is assigned as the second speaker and first addressee.

5. Results

In this section, we report the results from our speaker and addresee identification experiments along with the results from our feature ablation.

5.1. Sequence Labeling

To test how well the model generalizes to other authors we test the authors against each other. Cross-validation was performed where each author represents a fold, e.g. one author is selected as the test set and the remaining authors as the training set. The results are presented in table 6.

TEST	RANDOM	LATEST	LATEST-VB	SEQ
		SPEAKER		
Strindberg	20.3	45.5	29.7	68.9
Sjöberg	26.8	44.7	27.8	73.4
Söderberg	29.0	53.7	33.3	70.9
Boye	32.0	34.7	36.3	41.3
Average	27.0	44.6	29.2	63.7
		ADDRESSEE		
Strindberg	17.5	46.3	35.6	42.4
Sjöberg	24.5	38.2	23.9	65.1
Söderberg	22.5	46.2	25.8	44.0
Boye	30.0	28.9	27.3	32.2
Average	23.6	39.9	28.1	46.0

Table 6: Accuracy of cross-author speaker and addressee identification compared with three baselines. RANDOM = random baseline; LATEST-VB = latest mentions with speech verb; LATEST = latest mentions; SEQ = accuracy of sequence labelling.

For speaker identification, the performance is around 70% for all corpora except for Boye, where the accuracy is at 41.3%. For addressee identification, the results tend to vary more. Two of the corpora, Söderberg and Strindberg, have an accuracy of 44% and 42.4%, while Sjöberg's accuracy is 65.1%. Again, Boye's accuracy is the lowest with only 32.2%.

Comparing speaker and addressee identification we see that the results are similar between them for Sjöberg and Boye, while the difference between speakers and addressees is larger for Söderberg and Strindberg.

5.2. Feature Ablation

To evaluate the impact of each feature, we performed a feature ablation experiment, the results of which can be found in Table 7.

For the feature ablation, we perform cross-validation using the authors as folds with each feature removed once. The results presented in Table 7 is the average accuracy change of the author cross-validation with the feature removed.

Generally, for speaker identification we see that the accuracy changes are rather small, with both positive and negative changes. For addressee identification, most of the accuracy changes higher than speaker identification and most changes are negative. The most important features appear to be feature 7 for speaker identification and features 8 and 10 for addressee identification.

ID	FEATURE REMOVED	SPK	ADD
1	c_i mentioned in l_j	$+0.2$	-0.6
2	c_i mentioned in l_{j-1}	-1.6	-2.8
3	c_i mentioned in l_{j-2}	$+0.6$	-1.5
4	c_i with speech verb in l_j	$+2.2$	-1.5
5	c_i with speech verb in l_{j-1}	-0.4	-2.9
6	c_i with speech verb in l_{j-2}	± 0.0	-4.1
7	c_i with speech verb/mentioned in l_j	-9.9	-2.3
8	c_i with speech verb/mentioned in l_{j-1}	$+3.0$	-6.6
9	c_i with speech verb/mentioned in l_{j-2}	$+0.7$	$+0.7$
10	$c_i = x_{i-k}$ $(k = 1, \ldots, 6)$	-1.3	-9.1
11	c_i mentioned in narrative	$+0.2$	-5.0
12	$count(c_i) = 0$ in narrative	$+2.3$	-2.9
13	$0 < count(c_i) \leq 2$ in narrative	$+0.5$	-1.3
14	$count(c_i) > 2$ in narrative	$+0.9$	-2.5
15	$count(c_i) \geq 5$ in global context	-2.6	-3.2
16	$count(c_i) \geq 15$ in global context	$+0.4$	-3.6
17	c_i is n:th most recent mention	$+0.1$	-6.0
18	c_i is 0:th mention	-0.1	-4.7
19	c_i is 0:th mention + speech verb in l_j	-0.3	$+0.5$

Table 7: Feature ablation results for speakers (SPK) and addressees (ADD). Differences are given in percentage points relative to the baseline results.

6. Discussion

In this section, we discuss the results from the author cross-validation and from the feature ablation. Additionally, we analyze a set of errors generated in the experiments manually.

6.1. Baselines

The results from Table 6 show that the latest mentions baselines (third baseline) seem to provide a better indicator for both speakers and addressees than the latest occurrence with speech verb (second baseline).

It is interesting to note that for the latter, the overall difference between speakers and addressees is only 1.1 percentage points. In contrast, the overall difference between speakers and addressees for the latest mentions baseline is 4.7 percentage points. This is quite surprising, as one would expect the latest mentions with speech verb baseline to perform better for speakers than for addressees, and the latest mentions baseline to be more balanced.

6.2. Model Accuracies

Comparing the results against the baselines we see that all models perform better than the random and latest occurrence with speech verb baseline. The system also generally performs better than the latest mentions baseline, but there are two exceptions. When testing against Strindberg and Söderberg and training on the other authors, the addressee identification model performance is 3.9 and 2.2 percentage points lower than the latest mention baseline.

The difference between speaker and addressee identification for the sequence model is 17.7 percentage points, compared to the difference between the latest mentions baseline which is 4.7 percentage points and the latest mentions with

speech verb baseline where the difference is 1.1 percentage points.
The difference appears when the number of information sources increases from one to 19, that is, when more textual information is captured the difference in correct predictions between speaker and addressee identification grows. This would indicate that the speakers of a dialogue are signaled more explicitly than the addressees in the text, which would correspond to the notion that given the speakers of a dialogue, the addressees should be possible to infer from them.
Interestingly, observing Table 7 we can see that for speakers, 12 out of the 19 features produced accuracy changes of less than 1 percentage point, while for addressees only 3 features resulted in changes lower than 1 percentage point. This indicates that the features in general provide more information useful to the system regarding addressees than the speakers.
The notion that addressees are indicated to a lesser degree in the text thus seems implausible, given that the features applied to addressees show a larger change compared to speakers. However, if we consider the difference in the distribution of types presented in Table 3 for speakers and Table 4 for addressees, we see that there are significant differences for explicit, anaphoric and definite description indicators. Firstly, explicit speaker mentions are certain indicators that the character is the speaker, the indicator is less strong for addressees. An explicit mention in a line without a speech verb may be the addressee, but can also be a passer-by or a person not currently present. Thus, to predict addressees, the system must rely on other information sources than explicit mentions, which results in higher accuracy changes for more features.
Furthermore, definite descriptions are not treated in any special manner. This means that these indicators will be treated in the same manner implicit mentions. Anaphoric pronouns have some special treatment in the form of two heuristic features, 18 and 19. However, anaphora resolution is a complex problem and the current system will need a more sophisticated method for these in the future. We can see this clearly in the accuracy drop for speaker identification in the Boye corpus. In comparison to the other authors, the accuracy of Boye is roughly 30 percentage points lower. Observing Table 3 we see that approximately 50% of the speakers are indicated by anaphoric pronouns in Boye.
Given these differences of types and their impact, the differences in the feature ablation are not too surprising and the difference in accuracy between speakers and addressees makes sense.

6.3. Feature Ablation

6.3.1. Speakers

A tendency in dialogues is that there may be many different characters in a dialogue and that they may occupy lines in an irregular pattern. Another conflicting tendency is turn-taking, that states that there should be a regular alternating pattern between dialogue participants. These tendencies would primarily be caught by feature 8, which looks at the previous line. For speakers this features shows a gain of 3 percentage points while showing a loss of 6.6 percentage points for addressees. To some degree for speakers, feature 8 seem to encode both of these tendencies, which results in a feature which introduces uncertainty. For addressees on the other hand, the turn-taking pattern seems more consistent, where occurrence with a speech verb in the previous line is a good indicator of the current addressee.
Whereas mention and occurrence with speech verb is a strong indicator for addressees in the previous line, the feature (7) which encode this for the current line appears to be a strong indicator for speakers. The removal of this feature resulted in a loss of 9.9 percentage points.
That feature 7 and 8 show prominent losses for a particular role agrees with the notion that given a dialogue, the speakers and addressees tend to alternate. That is after one character passes the turn the other character responds to the first one, this scenario would be captured by observing that A speaks first, thus A is not the addressee, in the next line A was the previous speaker which indicates that A is currently being addressed.

6.3.2. Addressees

For addressee identification, there are quite a few features that show prominent accuracy losses.
The feature with the highest accuracy loss is the hypothesis feature (10), which captures the previously selected addressees, the removal of this features resulted in a loss of 9.1 percentage points. This would indicate that the structure of a dialogue seems to be an important factor, especially in comparison to speakers where the feature only shows a loss of 1.3 percentage points.
Another feature which resulted in a high loss for addressee identification is feature 11, which captures if the character is mentioned in the narrative or not. Removing this feature resulted in a loss of 5 percentage points, which indicates that presence in the narrative, i.e. the running text immediately preceding the dialogue, is an important aspect in addressee identification. For speaker identification, removing this feature only resulted in a marginal change of accuracy.
Similarly, the feature that captures the relative order in which the characters are mentioned shows a loss of 6 percentage points for addressees, while for speakers only resulting in a marginal change in accuracy. This would indicate that the relative recency of a character is an important factor in addressee identification.

6.4. Error Analysis

Error analysis has been done by selecting 30 errors randomly from both speaker and addressee identification.

6.4.1. Sequence Hypothesis

A frequent problem that appears for both speakers and addressees is that the hypothesis feature (10) may reinforce an incorrect character sequence. In the model, this feature will favor an alternating pattern between two characters. If an incorrect character is selected early in the dialogue, the hypothesis feature will tend to reinforce the incorrect choice simply because it fits better with the current character sequence.
For example, consider the correct sequence ABABAB, where the current predictions are ACA. When considering

the fourth prediction, the feature which captures the turn-taking heuristic (10) will favor C over B since C has occurred previously in the prediction while B has not. Given that C is predicted, followed by A, for the sixth predictions the turn-taking will be reinforced even further because C now appears twice in the predicted sequence. In this manner early incorrect predictions may reinforce or exclude the correct hypothesis.

Relatedly, characters talking about themselves, typically during introductions, may throw the speaker identification off track since characters are rarely mentioned in their own utterances. Lacking other strong evidence, this can cause a sequence like ABAB to be mistaken for BABA, if A introduces him/herself in the first line thereby making any hypothesis starting with A seem unlikely.

One particular problem found during the analysis is that our model assumes that all characters have been identified by name or alias(es), which makes dialogues involving unnamed participants (e.g., passers-by) challenging since the most useful features rely on mentions of names in the text. This error can often be seen for addressee identification where there are cases when one person is speaking to a group. A group is annotated as "SEVERAL" in the data, and thus will not be covered by most of the features. A more efficient way of handling these cases is to develop a set of features to determine if the content of a line is intended for a group of people or an individual.

In some cases, the hypothesis feature presents a problem because the dialogue does not follow a regular structure. These are dialogues where one speaker occupies two or more lines in a row, or dialogues in which there are more than two speakers.

6.4.2. Mention Order

Another problem that appears for both speakers and addressees is the mention order. In some cases, particularly in the beginning of a dialogue, the feature set of different characters tend not to show many differences. One feature however, will never be equal between the characters, namely feature 17 which capture the order in which the characters are mentioned in. This feature tends to work best in conjunction with other features, and not as the deciding factor. Resolving this is rather hard as there is not much that can be done to the feature itself. To avoid this problem, the other features used must show a larger variety to prevent one feature from being the deciding factor.

6.4.3. Narrative

A problem that appears for addressee identification is when the addressee is not mentioned in the narrative. Generally, mentions in the narrative are seen as a positive indicator for addressees, however many of the narratives are rather short and may not contain many mentions. Two possible improvements to remedy these situations is to capture mentions in previous narratives and to capture membership of previous dialogues.

A common narrative encountered in the data is the following: "... and then he said:", where the speaker is indicated with a speech verb, but the addressee is not present. These types of narratives present another problem for our system, namely that occurrences with speech verbs are not captured for narratives, and that certain narrative mentions are only relevant for speaker identification. Implementing a feature to capture this would work both as a positive indicator for speakers and a negative indicator for addressees.

6.4.4. Chapter

In some of the chapters there is a great deal of running text which includes many mentions while other chapters may primarily consist of dialogue. This presented a problem for feature 15 and 16 which capture how many times the character has been mentioned thus far. Given the different structures, different chapters in the training data were observed to assign quite different weights to this feature, which resulted in it being positive for some training sets and negative for others. To try and resolve this inconsistency there are some possibilities, one solution would be to convert the raw frequency into a relative frequency. We may also try to capture the structure of the chapters in other ways, where the total number of dialogues in comparison to paragraphs of running text is used as a feature, or as a factor in the frequency measurement.

7. Conclusions and Future Work

We have described a general method for identifying both speakers and addressees in dialogues extracted from literary fiction. The dataset we have used is small, but given that our results are based on out-of-domain training data we regard the approach as promising. This is an important aspect as it removes the need of having annotated data for each author investigated.

Direct comparison with previous work is difficult, primarily because of differences in the data-size and the experimental setup. However, our results on speaker identification are relatively similar to previously obtained results (Elson et al., 2010; He et al., 2013; Muzny et al., 2017).

For addressee identification, the only other results are from Yeung and Lee (2017). Using out-of-domain data they report a high loss of accuracy. Our proposed method seems to handle out-of-domain data efficiently.

For future work, definite descriptions and pronouns will have to be handled by a co-reference system. Based on the feature ablation and the error analysis there are several improvements that can be made to the current system. Furthermore, currently the characters are given to the system. However, it would be much more interesting to also extract these automatically. This would benefit the system as all the candidates considered will be somewhat relevant, meanwhile this need not be true for the current implementation.

Another interesting addition would be to apply a social network analysis on the chapter and use the relationships between the current character and the previously selected characters as a feature.

8. Acknowledgements

The work presented here has been supported by an infrastructure grant from the Swedish Research Council (SWE-CLARIN, project 821-2013-2003). We gratefully thank Litteraturbanken at University of Gothenburg and the Strindberg Project at Stockholm University for providing us with the electronic version of the book material.

9. Bibliographical References

Agarwal, A., Corvalan, A., Jensen, J., and Rambow, O. (2012). Social Network Analysis of Alice in Wonderland. In *Proceedings of the Workshop on Computational Linguistics for Literature, CLfL@NAACL-HLT 2012, June 8, 2012, Montréal, Canada*, pages 88–96.

Collins, M. (2002). Discriminative training methods for Hidden Markov Models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Elson, D. K., Dames, N., and McKeown, K. R. (2010). Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 138–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

He, H., Barbosa, D., and Kondrak, G. (2013). Identification of speakers in novels. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1312–1320, Sofia, Bulgaria, August. Association for Computational Linguistics.

Koivisto, A. and Nykänen, E. (2016). Introduction: Approaches to fictional dialogue. *International Journal of Literary Linguistics*, 5.

Moretti, F. (2011). Network Theory, Plot Analysis. *Literary Lab, Pamphlet 2*, May.

Muzny, G., Fang, M., Chang, A., and Jurafsky, D. (2017). A two-stage sieve approach for quote attribution. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 460–470, Valencia, Spain, April. Association for Computational Linguistics.

Newman, M. E. and Girvan, M. (2004). Finding and evaluating community structure in networks. , 69(2):026113, February.

O'Keefe, T., Pareti, S., Curran, J. R., Koprinska, I., and Honnibal, M. (2012). A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 790–799. Association for Computational Linguistics.

Rydberg-Cox, J. (2011). Social Networks and the Language of Greek Tragedy. *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*, 1(3).

Vala, H., Dimitrov, S., Jurgens, D., Piper, A., and Ruths, D. (2016). Annotating characters in literary corpora: A scheme, the charles tool, and an annotated novel. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Yeung, C. Y. and Lee, J. (2016). An annotated corpus of direct speech. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Yeung, C. Y. and Lee, J. (2017). Identifying speakers and listeners of quoted speech in literary works. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 325–329. Asian Federation of Natural Language Processing.

Word Embedding Evaluation Datasets and Wikipedia Title Embedding for Chinese

Chi-Yen Chen, Wei-Yun Ma
Institute of Information Science, Academia Sinica
128 Academia Road, Section 2, Nankang, Taipei, Taiwan 115
{ccy9332, ma}@iis.sinica.edu.tw

Abstract

Distributed word representations are widely used in many NLP tasks, and there are lots of benchmarks to evaluate word embeddings in English. However there are barely evaluation sets with large enough amount of data for Chinese word embeddings. Therefore, in this paper, we create several evaluation sets for Chinese word embedding on both word similarity task and analogical task via translating some existing popular evaluation sets from English to Chinese. To assess the quality of translated datasets, we obtain human rating from both experts and Amazon Mechanical Turk workers. While translating the datasets, we find out that around 30 percents of word pairs in the benchmarks are Wikipedia titles. This motivate us to evaluate the performance of Wikipedia title embeddings on our new benchmarks. Thus, in this paper, not only the new benchmarks are tested but some new improved approaches of Wikipedia title embeddings are proposed. We perform training of embeddings of Wikipedia titles using not only their Wikipedia context but also their Wikipedia categories, most of categories are noun phrases, and we identify the head words of the noun phrases by a parser for further emphasizing their roles on the training of title embeddings. Experimental results and the comprehensive error analysis demonstrate that the benchmarks can precisely reflect the approaches' quality, and the effectiveness of our improved approaches on Wikipedia title embeddings are also verified and analyzed in detail.

Keywords: word embedding, word embedding benchmark, Wikipedia title embedding, Wikipedia category, knowledge base

1. Introduction

Word embeddings are widely used in various natural language processing (NLP) tasks. Researches evaluates word embeddings on some extrinsic, practical NLP tasks, and also some intrinsic tasks. The two most popular kinds of intrinsic tasks are word similarity and analogical reasoning, and there are an array of benchmarks for the two kind of tasks in English. Take word similarity for instance, the MEN-3k (Bruni et al., 2012) dataset consists of 3,000 word pairs. The MTurk-287 (Radinsky et al., 2011) dataset consists of 287 word pairs. The SimLex-999 (Hill et al., 2016) consists of 999 word pairs. For analogical reasoning tasks, the dataset is composed of analogous word pairs. Each word pair is a tuple of word relations that follow a common syntactic relation. For instance, the Google analogy (Mikolov et al., 2013) dataset consists of 19,544 question pairs.

Though there are lots of benchmarks to evaluate word embeddings in English, there are barely evaluation sets with large enough amount of data for Chinese word embeddings. For example, researchers at York University released a dataset[1] for Chinese word similarity; nevertheless, there are only 50 word pairs in the trial data and around 500 word pairs in the full dataset. Chen and Ma (2017) also created a small amount of datasets for Chinese word embedding, but the quantity and types are still far from English counterpart. Therefore, we generate several new evaluation sets for Chinese word embedding on both word similarity task and analogical task through translating some existing popular evaluation sets from English to Chinese. To assess the quality of translated datasets, we obtain human rating from both experts and Amazon Mechanical Turk workers.

While translating the datasets, we find out that around 30 percents of word pairs are Wikipedia titles. This motivate us to evaluate the performance of Wikipedia title embeddings on our new benchmarks. Wikipedia provides not only structural data, i.e., knowledge graphs via info-boxes, but also the nonstructural data, i.e., Wikipedia text, and semi-structural data, i.e., the title's categories. Most Wikipedia categories are long noun phrases other than noun words, so they are sometimes able to provide more complete information than info-boxes. Chen and Ma (2017) attempted to obtain title embedding of Wikipedia based on Wikipedia's content and categories; however, just a small amount of datasets for evaluation are used to evaluate the effectiveness and are not able to provide comprehensive evaluation results and error analysis. Thus, in this paper, we introduce the procedure of creating the new benchmarks and describe the difficulties we face along with the solutions we adopted. In addition, we extend (Chen and Ma 2017)'s work on training of embeddings of Wikipedia titles by considering the titles' Wikipedia content and categories, which syntactic heads are restricted to be a head of noun phrase, identified by a parser. We evaluate our new approaches on our new benchmarks, aiming to present a more complete and comprehensive error analysis to demonstrate the benchmarks' effects and also verify the performance of our improved approaches on Wikipedia title embeddings.

2. Benchmarks for Chinese Word Embeddings Evaluation

In English, there are quite a lot datasets that are commonly used as benchmark for evaluating word embedding. However, there are barely evaluation sets with large enough amount of data for Chinese word embeddings. The two

[1] https://www.cs.york.ac.uk/semeval-2012/task6/index.php%3Fid=data.html

most popular kinds of benchmarks to evaluate word embedding are word similarity task and analogical reasoning task. Therefore, we create some benchmarks for Chinese word embedding on both kinds of tasks through translating some existing popular evaluation sets from English to Chinese and remains the scores of the original benchmarks. For word similarity task, we translate six datasets: **SimLex-999** (Hill et al., 2016), **MEN-3k** (Bruni et al., 2012), **MTurk-287** (Radinsky et al., 2011), **WordSim-353** (Finkelstein et al., 2001), also the partitioned datasets from WordSim-353, separated into two different relations, **WS353-Relatedness** and **WS353-Similarity** (Zesch et al., 2008; Agirre et al., 2009). For analogical reasoning task, we translate the **Google analogical dataset** (Mikolov et al., 2013).

We will illustrate the translation process and difficulties encountered during translation in the following sections. The datasets are available to download at the link[2].

Dataset		Original	Translated
Word Similarity	SimLex-999	999	999
	MEN-3k	3,000	3,000
	MTurk-287	287	287
	WordSim-353	353	353
	WS353-R	252	252
	WS353-S	203	203
Analogical Reasoning	Google	19,544	11,126

Table 1: Size of each dataset, including original version and translated version.

2.1. Translation Process

In this section, we will illustrate the translation process and policy we apply. First, to get appropriate translated Chinese words, we use authoritative online dictionaries as resources, including Cambridge Dictionary (English-Chinese), E-Hownet (Group and others, 2009; Huang et al., 2008; Chen et al., 2005), Merriam-Webster Dictionary, Oxford Dictionary, and Wiktionary[3]. To maintain the original semantic meaning in English, we disambiguate the appropriate Chinese words by using Cambridge Dictionary (English-Chinese) and E-Hownet. Some words in English have different meaning in different situations.

Take some word pairs from MEN-3k dataset, which consists of word pairs with similarity score in a 50 scale, for example, **(*palm*, *tree*) 37.0** and **(*hand*, *palm*) 44.0** both have relatively high similarity scores. According to the dictionaries, *palm* can either mean "手掌(the under part of the hand between the fingers and the wrist)" or "棕榈树(a tropical tree, shrub, or vine with a usually tall stem or trunk topped with large leaves that are shaped like feathers or fans)". Because the similarity scores are high for both **(*palm*, *tree*) 37.0** and **(*hand*, *palm*) 44.0** in the original dataset, we cannot just replace *palm* with a single meaning as we ought to maintain the original semantic meanings in English. Thus, we translate (*palm*, *tree*) to (棕榈树 *(palm)*, 树 *(tree)*) and (手 *(hand)*, 手掌 *(palm)*) to (*hand*, *palm*).

Furthermore, due to different language usages, in Google analogy dataset, we ignore the tuples based on certain English grammatical variations which do not exist on Chinese words, such as singular-plural nouns, verb tenses, third-person singular verb endings, and comparison of adjectives For instance, tuples like **(mouse, mice, goat, goats)**, **(describing, described, knowing, knew) (write, writes, decrease, decreases)**, and **(lucky, luckiest, wide, widest)** will be discarded during the translation since there are no proper mappings from English to Chinese for these grammar rules.

In addition, we ignore the appendix "的" and "地" of adjectives and adverbs relatively since the usage is relatively rare in daily Chinese language usage. For example, the tuple **(*immediate*, *immediately*, *rare*, *rarely*)** can be rigorously translated as *(立即的, 立即地, 稀有的, 稀有地)*; however, in daily language, it is uncommon for these phrases with "的" and "地". To exemplify, according to Cambridge Dictionary, the sentences: "We must make an immediate response." is translated as "我们必须立即作出反应。" and "We really ought to leave immediately." is translated as "我们真的应该马上就走。". None of "立即的 (immediate)" or "立即地 (immediately)" is used in daily language. After the translation process based on these considerations, the size of Google analogy dataset is reduced from 19544 words to 11126 words, shown in Table 1.

Even though we follow the translation policy, we still encounter some obstacles while translating the datasets. The difficulties will be elaborated in Section 2.2..

2.2. Difficulties

2.2.1. Word Similarity Datasets Translation

In word similarity datasets, some English word pairs have slight differences which are even unnoticeable in Chinese. Both words in a word pair have similar meanings but not exactly the same. Take word pairs from MEK-3k for instance, **(*stairs*, *staircase*) 49.0**, though they are different in English, both words mean "楼梯 (stairs)" in Chinese. The solution to this case is that we look the words up in WikiDiff[4] to figure out the slight differences between words and then select a different but appropriate meanings in Chinese, i.e., (*stairs*, *staircase*) is translated as (阶梯 *(stairs)*, 楼梯 *(staircase)*).

There are also words that have exact the same meaning but with different expressions; hence, the word pair's similarity score is not full mark. For example, the word pair **(*bicycle*, *bike*) 45.0**, though both words mean "a two-wheeled vehicle that you sit on and move by turning the two pedals", the similarity score is not 50. The solution to this case is that we look these words up in Cambridge Dictionary and get the meanings in different Chinese expressions separately. That is, we find explanations for both ***bicycle*** and ***bike*** are "脚踏车, 单车, 自行车", and then we choose different Chinese words for ***bicycle*** and ***bike*** relatively. Also, we look up the word "脚踏车" in E-Hownet to get its synonyms in Chinese. Thus, (*bicycle*, *bike*) is translated as (脚踏车 *(bicycle)*, 自行车 *(bike)*).

[2]http://ckipsvr.iis.sinica.edu.tw/ecemb/reg.php

[3]https://en.wiktionary.org/wiki/Wiktionary

[4]http://wikidiff.com/

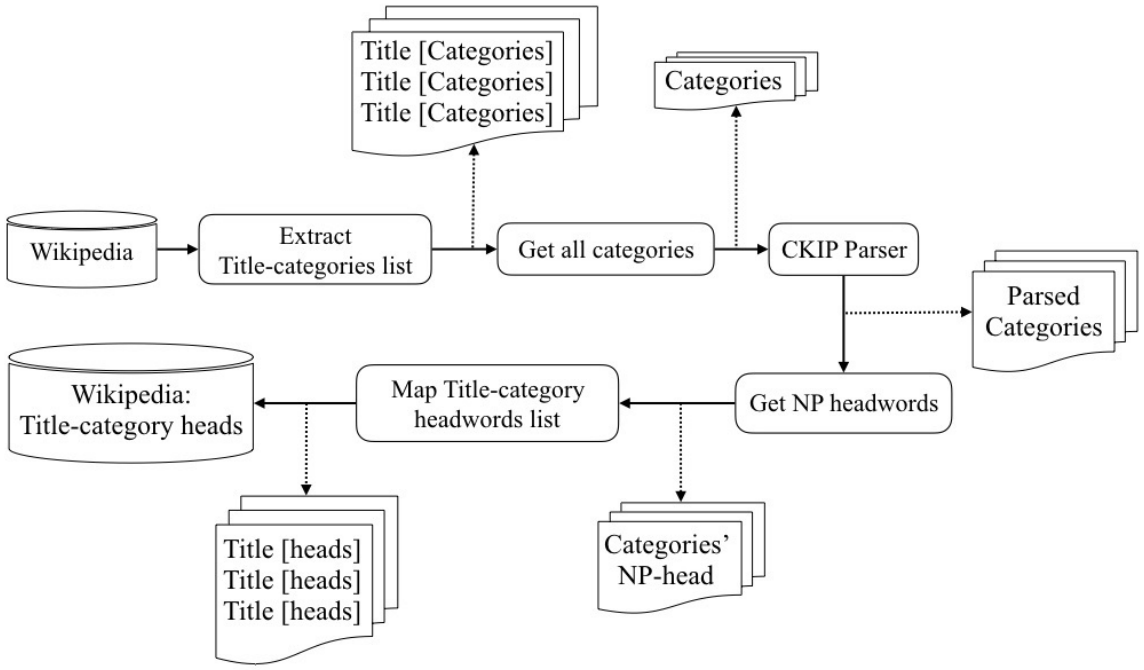


Figure 1: The work flow of getting categories' noun-phrase headwords mentioned in Section 3.3.2.

2.2.2. Analogical Reasoning Dataset Translation

In analogical reasoning questions set, some English words are difficult to find an appropriate mapping in Chinese words because they have more than one meanings. To solve this kind of problems, if a word has more than one forms as noun and others, we primarily consider its meaning as a noun, then as an adjective, and finally other forms because the original dataset mainly consists of nouns.

Take the tuple **(*croatia*, *croatian*, *thailand*, *thai*)** from Google analogical dataset, for instance. According to the dictionaries, *croatian* can be referred to "克罗埃西亚人 (a person from Croatia)", "克罗埃西亚语 (the language spoken in Croatia)", or "克罗埃西亚的 (belonging to or relating to Croatia, its people, or its language)". The solution to this case is that we choose "克罗埃西亚人 (a person from Croatia)" over other possible explanations because "克罗埃西亚人 (a person from Croatia)" is the first noun explanation in most dictionaries. While the same translation policy is applied, the translation of *thai* is "泰国人 (a person from Thailand)". That is, **(*croatia*, *croatian*, *thailand*, *thai*)** is translated as (克罗埃西亚, 克罗埃西亚人, 泰国, 泰国人).

3. Wikipedia Title Embedding

3.1. Wikipedia Data

While translating the datasets, we find out that around 30 percents of word pairs are Wikipedia titles. Wikipedia provides data in three structural extents: nonstructural, i.e., content, semi-structural, i.e., categories, and structural data, i.e., info-boxes. Categories are usually long noun phrases and provide more information than info-boxes. The page *Albert Einstein*, for example, is in categories of *ETH Zurich alumni*, *ETH Zurich faculty* and 20 more. The categories provide information that Einstein was both alumni and faculty of ETH Zurich while in info-box, only Einstein was related to ETH Zurich is shown.

We follow the work of (Chen and Ma 2017) and download the Chinese version of Wikipedia dump file in September 2016, which compromised 1,243,319 articles at a time. There are 244,430,247 words in the Chinese Wikipedia corpus and each title has 2.16 categories in average.

Chen and Ma (2017) obtained Wikipedia title embedding by linear combining context embedding and categories embedding. They generated context embedding by skipgram model and proposed several methods to generate categories embedding. In this paper, we use the same context embedding but purpose improved method to acquire categories embedding. We use CKIP Chinese parser (Hsieh et al., 2012; Yang et al., 2008; Hsieh et al., 2007) to parse each category and get the rigorously parsed noun-phrase (NP) headword.

3.2. Categories Embedding

Wikipedia categories can partially represent the corresponding title. Chen and Ma (2017) proposed several approaches of acquiring Wikipedia categories embedding, categories embedding for short. In this section, we first briefly introduce (Chen and Ma 2017)'s work, followed by description of our extension on the new approach to extract the headword of a category, elaborated in Section 3.3.2..

Formally, given a title t and its corresponding categories from C^1 to C^n, each category C^i has been word-segmented as K words from W_1^i to W_K^i and W_H^i is the headword of C^i. We use c^i and w_j^i to represent embedding of category C^i and word W_j^i.

3.2.1. Average of Category Words

We acquire categories embedding $e_{category}$ by averaging every category of a single title. Considering the complete-

WS Evaluation Sets	SimLex999		Men-3k		MTurk-287		ws353		ws353r		ws353s	
	dev	test	dev	test	dev	test	dev	test	dev	test	dev	test
skipgram	18.20	**40.30**	67.10	64.60	**50.40**	59.40	60.70	61.10	52.10	57.70	63.60	67.60
Avg(words)	18.00	39.00	**67.80**	**65.20**	50.20	59.90	**61.60**	**62.00**	**53.20**	**58.90**	**64.40**	**67.80**
Avg(headwords)	18.00	39.00	67.70	**65.20**	50.10	**60.20**	**61.20**	61.90	52.90	58.80	**64.40**	**67.80**
Avg(NP-heads)	18.10	**40.30**	67.30	64.90	49.80	59.30	60.50	60.80	51.60	57.60	63.40	67.10
wsc(d=1)	18.20	39.80	67.50	65.10	50.10	59.40	60.40	61.40	51.90	58.20	63.50	66.90
wsc(d=2)	18.20	39.80	67.50	65.10	50.10	59.50	60.40	61.30	51.90	58.30	63.60	66.80
wsc(headwords)	**18.30**	39.80	67.50	65.10	49.90	59.50	60.40	61.40	52.40	58.20	63.60	66.90
wsc(d=1, NP-heads)	18.10	**40.30**	67.30	64.90	49.80	59.30	60.50	60.80	51.60	57.60	63.40	67.10
wsc(d=2, NP-heads)	18.10	**40.30**	67.30	64.90	49.80	59.30	60.50	60.80	51.60	57.60	63.40	67.10
wsc(NP-heads)	18.10	**40.30**	67.30	64.90	49.80	59.30	60.50	60.80	51.60	57.60	63.40	67.10

Table 2: Spearman correlation on word similarity task. All embedding are 300 dimensions.

WS Evaluation Sets	SimLex999		Men-3k		MTurk-287		ws353		ws353r		ws353s	
	dev	test	dev	test	dev	test	dev	test	dev	test	dev	test
Title coverage(%)	25.00	16.63	33.53	31.80	27.27	33.33	27.68	32.95	22.22	34.13	36.27	27.72
skipgram	9.40	49.40	69.40	73.20	**61.10**	55.90	62.40	78.10	**71.90**	76.10	75.40	79.80
Avg(words)	**10.80**	50.40	**70.60**	**74.50**	59.10	56.60	**63.40**	**78.20**	71.10	77.10	75.20	80.80
Avg(headwords)	10.70	50.60	70.50	**74.50**	58.50	**57.50**	62.90	78.00	71.00	**77.40**	75.00	**81.40**
Avg(NP-heads)	9.00	**50.90**	69.50	73.70	57.90	53.80	60.20	77.40	67.70	75.00	**75.60**	79.00
wsc(d=1)	10.30	50.20	70.20	**74.50**	59.80	55.90	62.00	77.60	71.30	76.80	**76.40**	80.20
wsc(d=2)	10.10	50.10	70.20	74.40	59.80	56.00	62.20	77.70	71.00	77.00	**76.40**	79.50
wsc(headwords)	10.40	50.50	70.30	**74.50**	59.70	55.80	61.70	77.70	**71.90**	76.40	**76.30**	79.70
wsc(d=1, NP-heads)	9.00	50.80	69.50	73.60	58.20	54.20	60.40	77.50	67.70	75.00	74.80	79.00
wsc(d=2, NP-heads)	9.00	50.80	69.50	73.60	58.20	54.20	60.40	77.50	67.70	75.00	74.80	79.00
wsc(NP-heads)	9.00	50.80	69.50	73.60	58.20	54.20	60.40	77.50	67.70	75.00	74.80	79.00

Table 3: Spearman correlation on word similarity task but only cope with word pairs that are Wikipedia titles. Wikipedia title coverage of each dataset is shown in the table. All embedding are 300 dimensions.

ness of category information, we obtain a category embedding by averaging all words in the category. Process of computing $e_{category}$ is shown as following:

$$e_{category} = \frac{1}{n}\sum_{i=1}^{n} c^i, \quad \text{where } c^i = \frac{1}{K}\sum_{j=1}^{K} w_j^i. \qquad (1)$$

3.2.2. Average of Category headwords

By observing Wikipedia data, we find out that most categories are noun-phrases and generally the headword of a noun-phrase contains more information than other words. We presume that sometimes other words besides the headword may bring some noisy information; thus, we acquire $e_{category}$ by averaging only every category headword of a single title and do not consider any other words. Computing process is shown as following:

$$e_{category} = \frac{1}{n}\sum_{i=1}^{n} c^i, \qquad \text{where } c^i = w_H^i. \qquad (2)$$

3.2.3. Weighted Sum of Categories (WSC)

We assume that each category of a title has different degree of representation and its representation depends on its headword's occurrence in context of the title. Therefore, we assert that categories should be treated distinctively according to their representation degrees. In this section, we acquire $e_{category}$ by summing up categories of a single title with the occurrence of category headword in context. Considering category information completeness, we obtain a category embedding by averaging all words' embeddings in the category but apply a different weight d to the headword. Computing process is shown as following:

$$e_{category} = \sum_{i=1}^{n} a_i c^i, \qquad (3)$$

where a_i is the category headword frequency in context with normalization and

$$c^i = \frac{1}{K+d-1}\left(\sum_{j=1}^{K} w_j^i + (d-1)\, w_H^i\right), \qquad (4)$$

where d is the weight added to the headword and will be adjusted during the experiments.

3.3. Headwords

In this section, we will introduce two different approaches we applied to extract headwords of categories: rule-based method and parsing-based method.

3.3.1. Rule-based Headwords Extraction

By observing Chinese linguistic structure, generally, the headword of a noun-phrase is the last word. Therefore, for each category C^i, we use its last word W_K^i as its headword. We denote the headwords obtained by this method as *headwords* during the evaluation section.

3.3.2. Phrasing-based Headwords Extraction

In this section, we illustrate the method of getting each category's headword by using CKIP Chinese Parser (Hsieh et al., 2012; Yang et al., 2008; Hsieh et al., 2007). Firstly, we extract title-categories mapping list, i.e., what categories are contained in each title page. Next, since there are many duplicated categories in different titles, we collect all categories as a set to reduce duplicated categories. This step can reduce the number of categories from around 2,691,151 to 378,540 and save around seven times of complexity. Then, we parse these categories one by one. After parsing the categories, we traverse through the parsed categories and extract the corresponding headwords of parsed categories only if the categories are noun-phrases. Finally, we create title-categories headword mapping list, that is, what headwords are contained in each title page. The procedure of getting noun-phrase headwords is shown as Figure 1. We denote the headwords obtained by this method as *NP-heads* during the evaluation section.

3.4. Title Embedding

Wikipedia title embedding, short for title embedding, is the improved word embedding with combination of context embedding and categories embedding. We acquire title embedding e_{title} by linear combining context embedding and categories embedding. The process of computing e_{title} is shown as following:

$$e_{title} = \alpha * e_{context} + (1 - \alpha) * e_{category}, \qquad (5)$$

where $0 < \alpha < 1$, $e_{context}$ is obtained from Skip-gram and $e_{category}$ is obtained from Section 3.2. The title embeddings are available at the link[5].

4. Evaluations

4.1. Datasets Size

In this section, we briefly introduce the size of datasets in the experiments. For word similarity task, there are six datasets: **SimLex-999**, **MEN-3k**, **MTurk-287**, **WordSim-353**, **WS353-Relatedness** and **WS353-Similarity** in Chinese version. For analogical reasoning task, there is one dataset: **Google dataset** in Chinese version. To get development set and testing set, we split every dataset in halves, each dataset's size is shown as Table 4.

[5]http://ckipsvr.iis.sinica.edu.tw/cwemb/reg.php

Dataset		dev	test
Word Similarity	SimLex-999	500	499
	MEN-3k	1,500	1,500
	MTurk-287	143	144
	WordSim-353	177	176
	WS353-R	126	126
	WS353-S	102	101
Analogical Reasoning	Google	5,563	5,563

Table 4: Size of each translated dataset, separated into development set and testing set.

4.2. Datasets Evaluation

To evaluate the translation quality, we ask some experts to manually rate similarity scores on our translated benchmarks. In Table 5, MEN-3k can get as high Spearman correlation with the original benchmark as 85, indicating that our translation process is able to preserve the original word semantics especially considering its large size. However MTurk-287 can only get Spearman correlation of 50. The reason could be either due to our translation process or because of the characteristics of the original dataset in English. To investigate the reason, we also obtain human rating from Amazon Mechanical Turk workers for MTurk-287. On Amazon Mechanical Turk, we assign each notation to three different workers and then average the similarity scores they provided. We collect human rating on both English and Chinese. In Table 6, we find that original dataset in English itself is difficult to reproduce the same correlation for Turkers, so it is more likely that MTurk-287 itself is hard to decide the similarity in nature instead of the translation problem.

	Score
MTruk-287 testing	50
MEN-3k testing	85

Table 5: Spearman correlation on original dataset and translated dataset.

	MTruk-287 testing
Original/ OurMTurk English	23
Original/ OurMTurk Chinese	21
mTruk English/ OurMTruk Chinese	49

Table 6: Spearman correlation on original dataset and translated dataset.

4.3. Word Similarity Tasks

Word similarity task datasets contain relatedness scores for word pairs; the cosine similarity of the two word embeddings should have high correlation. We have six

datasets: **SimLex-999**, **MEN-3k**, **MTurk-287**, **WordSim-353**, **WS353-Relatedness** and **WS353-Similarity** in Chinese version. We split datasets in halves to get development set and testing set, each dataset's size is shown as Table 4.
Table 2 shows the result of title embedding obtained in Section 3.4.. We tune the weight α via development set and then apply the best α, which $\alpha = 0.9$, on testing set. Comparing to the baseline, our proposed methods get significant improvement.
Furthermore, since we find out that about 30 percents of word pairs in the datasets are Wikipedia titles, we want to focus on these word pairs and figure out whether categories have positive effects on expressing the titles. Therefore, we evaluate the title embedding on only these word pairs which are Wikipedia titles in the datasets. Table 3 shows the result of title embedding obtained in Section 3.4., but only on the word pairs that are Wikipedia titles.
Based on both Table 2 and Table 3, we can conclude that categories indeed have positive effects on incorporating titles' information.

4.4. Analogical Semantics Tasks

Analogical reasoning dataset is compromised of analogous word pairs, i.e., pairs of tuples of word relations that follow a common syntactic relation. We use translated **Google dataset** and split it in halves as development and testing sets. Each set contains 5,563 questions.
Table 7 shows result of Linear Combination in 3.4.. We tune the weight α via development set and then apply the best α, which is $\alpha = 0.9$, on testing set. Comparing to the baseline, our proposed methods get significant improvement.

Method	Google	
	dev	test
Skip-gram	53.12	34.71
Avg(words)	**55.71**	**36.33**
Avg(headwords)	53.66	35.65
WSC (d=1)	**54.43**	**35.12**
WSC (d=2)	54.14	35.00
WSC (headwords)	53.17	34.96

Table 7: Accuracy on analogical reasoning task. All embeddings are 300 dimensions.

5. Discussion

5.1. Datasets

According to the evaluation result, we find out there is a high correlation between the quality of dataset and the evaluation scores. To be specific, according to Table 5, since Spearman correlation of MEN-3k is much higher than MTurk-287, we infer that MEN-3k has better quality compared with MTurk-287. The evaluation results on table 2 and table 3 indeed show that the title embeddings have better performance on MEN-3k compared with MTurk-287.
We observe that our approaches consistently get worse performances on some benchmarks than others. The reasons could be due to that for certain datasets, the similarity scores between words in word pairs are determined based on more various aspects. In general, the performance on SimLex-999 is the worst, so we look into the results of this dataset and indeed observe that there are more different aspects to express relations between words. Take a word pair in SimLex-999, which consists of word pairs with similarity score in a 10 scale, for instance, (**晚上 *(night)*, 白天 *(day)*) 1.88** has a low similarity score in the original dataset; nevertheless, words from title embedding that applies different categories embedding have much higher similarity as 5.45. Actually, we find out that *"*晚上 *(night)"* and *"*白天 *(day)"* have the same category: "一天里的时刻 (parts of a day)". This could explain why these two words in title embedding have higher similarity. On the orther hand, in original English dataset, *"*晚上 *(night)"* and *"*白天 *(day)"* are emphasized as entirely different time, which causes low similarity score.

5.2. Wikipedia Title Embedding

According to the experiment result, we can confirm that categories can provide valuable information for improving embedding sole based on context using skipgram.
We also find that for around half benchmarks, linear combination using category embedding which obtains from averaging all category words has the better performance. This circumstance reflects that in some cases, other words except headwords in the category could play more critical roles in the expression of the meaning of the category.
Another interesting finding is that, in some cases, the categories' representation degree fails to detect by the frequency of co-occurrence in the context since they are more related to other factors, such as the position of category appears in context. It is plausible that the category which denotes the first sentence in the context deserves most attention.
From error analysis, we find that the restriction of NP-head for categories based on parsing is especially able to filter out some noisy information in some benchmarks, such as SimLex999. For example, a title 神经 *(nerves)* has categories including 神经 *(nerves)*, 周围神经系统 *(peripheral nervous system)*, 神经解剖学 *(neuroanatomy)*, 软组织 *(soft tissue)* and 日语借词 *(Chinese words of Japanese origin)*. In this case, only 日语借词 *(Chinese words of Japanese origin)* is incorrectly parsed as verb-phrase and is discarded. Since it indeed has nothing to define the meaning of nerve, thus, without considering it, the title will be expressed more precisely. Although we can see the effect of the restriction on some benchmarks, we also observe that in some other benchmarks, the effect of this restriction is below our expectation. Based on our error analysis, one reason is that some categories are mistakenly parsed into verb-phrases due to no context was provided during parsing process and thus some important information is missed. For instance, a title 琴酒 *(gin)* has categories including 蒸馏酒 *(distilled wine)*; nevertheless, 蒸馏酒 *(distilled wine)* is incorrectly parsed as verb-phrase: 蒸馏　酒 *(to distill wine)*. In this case, the information of 蒸馏酒 will be discarded while we are acquiring categories' heads to be NP's head; hence the title embedding of 琴酒 *(gin)* will have no information from its categories. On the other hand, categories embedding which use the last word as headwords could still contain the information of 蒸馏酒 *(distilled drinks)*.

Another possible reason is that for some titles, their verb-phrased categories could be actually pretty critical to define or express the title. This possibility is worth further investigation in our future work.

6. Conclusion

In this paper, we create several benchmarks for Chinese word embedding on both word similarity task and analogical task through translating some existing popular evaluation sets from English to Chinese. To assess the quality of translated datasets, we obtain human rating from both experts and Amazon Mechanical Turk workers. And we also confirm that a Wikipedia title's categories can help define or complement the meaning of the title besides the title's Wikipedia context. Experimental results and the comprehensive error analysis demonstrate that the benchmarks can precisely reflect the approaches' quality. Furthermore, we compare with two different approaches to extract headwords of categories - rule-base method and parsing-based method, and find out both approaches have their pros and cons and are thus worth further investigating how to remains the pros and eliminate the cons in the future work.

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. Association for Computational Linguistics.

Bruni, E., Boleda, G., Baroni, M., and Tran, N.-K. (2012). Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Chen, C.-Y. and Ma, W.-Y. (2017). Embedding wikipedia title based on its wikipedia text and categories. In *Asian Language Processing (IALP), 2017 International Conference on*, pages 146–149. IEEE.

Chen, K.-J., Huang, S.-L., Shih, Y.-Y., and Chen, Y.-J. (2005). Extended-hownet: A representational framework for concepts. *Proceedings of OntoLex 2005-Ontologies and Lexical Resources*.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Group, C. et al. (2009). Lexical semantic representation and semantic composition-an introduction to e-hownet. Technical report, Technical Report), Institute of Information Science, Academia Sinica. 16 Yu-Ta Chen et al.

Hill, F., Reichart, R., and Korhonen, A. (2016). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hsieh, Y.-M., Yang, D.-C., and Chen, K.-J. (2007). Improve parsing performance by self-learning. *International Journal of Computational Linguistics & Chinese Language Processing, Volume 12, Number 2, June 2007*, 12(2):195–216.

Hsieh, Y.-M., Bai, M.-H., Chang, J. S., and Chen, K.-J. (2012). Improving pcfg chinese parsing with context-dependent probability re-estimation. In *Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 216–221.

Huang, S.-L., Chung, Y.-S., and Chen, K.-J. (2008). E-hownet: the expansion of hownet. In *Proceedings of the First National HowNet Workshop*, pages 10–22. Citeseer.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Radinsky, K., Agichtein, E., Gabrilovich, E., and Markovitch, S. (2011). A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web*, pages 337–346. ACM.

Yang, D.-C., Hsieh, Y.-M., and Chen, K.-J. (2008). Resolving ambiguities of chinese conjunctive structures by divide-and-conquer approaches. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

Zesch, T., Müller, C., and Gurevych, I. (2008). Using wiktionary for computing semantic relatedness. In *AAAI*, volume 8, pages 861–866.

An Automatic Learning of an Algerian Dialect Lexicon by using Multilingual Word Embeddings

Karima Abidi, Kamel Smaïli
SMarT Group, LORIA, F-54600, France
{karima.abidi, kamel.smaili}@loria.fr

Abstract

The goal of this work consists in building automatically from a social network (Youtube) an Algerian dialect lexicon. Each entry of this lexicon is composed by a word, written in Arabic script (modern standard Arabic or dialect) or Latin script (Arabizi, French or English). To each word, several transliterations are proposed, written in a script different from the one used for the word itself. To do that, we harvested and aligned an Algerian dialect corpus by using an iterative method based on multlingual word embeddings representation. The multlinguality in the corpus is due to the fact that Algerian people use several languages to post comments in social networks: Modern Standard Arabic (MSA), Algerian dialect, French and sometimes English. In addition, the users of social networks write freely without any regard to the grammar of these languages. We tested the proposed method on a test lexicon, it leads to a score of 73% in terms of F-measure.

Keywords: Multilingual word embeddings, Algerian dialect, CBOW, comparability

1. Introduction

The wide use of social networks arises several new NLP issues: stretched letters, misspelled words, use of emoticons, condensed writing, etc. For the use of Arabic in social networks, the same phenomena are observed, henceforth other issues are also noticed and especially for Arabic dialects.

In this work, we are interested by the Algerian Arabic dialect. One needs to know that in Algeria, people speak their mother tongue, which is an Arabic dialect, but could speak the official language that is Modern Standard Arabic (MSA), French and sometimes English. Nevertheless, what makes the issue more challenging is that people can mix in the same sentence the whole previous languages.

Since this vernacular language is not written and no standardization exists, people write their comments in a free way. They do not pay special attention to grammar, consequently for uneducated people, they write a word as they want or, at the best, such as it is pronounced. People write sometimes Algerian dialect by using the Latin script (LS) because they are influenced by the French culture, this phenomenon is named in the community working on dialect: *Arabizi*. This could be explained also by the fact that in Algeria the mobile phone keyboards are in French that makes writing in Arabic more difficult. This obviously constitutes a serious issue because Arabic NLP tools could not be used on Latin script and more especially because the corresponding resources are not available.

Unfortunately, the NLP tools in French cannot be used either since the written words in LS are not necessarily in French. In the following, we give an example of a post extracted from YouTube written in LS:

{*ouiii 7na haka kima galha bn chnt w li maybghounach w ha l7itan ha l7itan*}.

That should be written in Arabic script (AS) such as:

{وي حنا هكا كيما قلها بن شنات و لى ميبغيوناش و ها لحيطان ها لحيطان}

that means: *We are like this such as Benchenet said: those who do not like us it does not matter it does not matter*.

Sometimes, people when they write in Latin script, they use some codifications for specific Arabic letters that do not exist in Latin. This is the case of ع that is replaced by *3*, ق that is written *9*, خ as *5* and other codes that are not officially adopted by everyone.

Another well known and frequent phenomenon in the Algerian dialect is the code-switching (Yoder et al., 2017)(Abidi and Smaïli, 2017a) that exist for other langauges (Dey and Fung, 2014). In Algeria people switch from the local Arabic to French or sometimes to MSA to express an idea in a well structured language. Switching may concern one isolated word or several contiguous words. For example, in our corpus we have found the following examples, which should be read from right to left:

{*les ingrédients* الّله يهديكم وين راهم}

{*Algérienne qui aime le Maroc* وجعني قلبي الجزائر و المغرب بلد واحد}

In these two examples the writers started the sentences in Arabic and finished them in French.

In this paper, we are interested by the creation of a lexical resource containing for each entry, the corresponding different ways to write the same word. As presented before, a dialect word could be written in Arabic and Latin script. And for both of them, a word has different graphies since people write freely. For instance, in our corpus we found seven forms of the word بلادي:

{بلادي: بلدي, baldi, beladi, bledi, bladi, baladi }.

We propose in the following to use the concept of word embeddings in order to build a lexicon for Algerian spoken language, containing for each entry its different forms of writing. The entry could be a word in Algerian dialect expressed in Arabic script (dialect or MSA) or in Latin Script (Arabizi, French and sometimes in English). These forms are extracted from a large corpus harvested from YouTube.

The rest of the paper is organized as follows: Section 2. concerns the related work while Section 3. describes the collected corpus. In Section 4., we discuss the automatic method used to learn an Algerian Dialect Lexicon (ADL). In Section 5., we present a protocol to evaluate the rele-

vance of the extracted lexicon. Finally, in Section 6., we conclude and we discuss the future work.

2. Related work

The NLP community, which started few years ago to pay attention to Arabic dialects, is faced to the lack of resources. To remedy to this problem, the researchers often created them from scratch. Creating automatically a lexicon for Arabic dialects is then a challenging and important task for processing dialects. In (Al-Sabbagh and Girju, 2010), the authors propose to induce an Egyptian dialect lexicon by mining the Web. The idea is to create a lexicon of Egyptian dialects with their corresponding MSA synonyms. The approach used is based on retrieving collocation words in a large corpus. This approach leads to a lexicon of 1000 entries. This work is different from ours, but constitutes also an attempt to build automatically a lexicon. To the best of our knowledge, this resource does not exist for Algerian dialect, while it is necessary for different NLP tasks. In fact, for dialect identification, several methods based on machine learning have been used such as in (Belgacem and Zrigui, 2010) (Al-Badrashiny et al., 2015) (Harrat et al., 2015), but also a dictionary-based method could be considered, that is why a lexicon dialect may help to detect the origin of the Arabic dialect.

For Automatic Speech Recognition (ASR) of a dialect, a vocabulary is necessary to recognize the uttered sentences. In (Menacer et al., 2017), the authors adapted a MSA speech recognition system, but with French acoustic data. In fact, they have not succeeded to find Algerian data to adapt their ASR. The choice of using French data is motivated by the fact that Algerian dialect is highly code-switched as explained in the introduction. The same phenomenon is observed also in machine translation. In fact, in (Meftouh et al., 2015), the authors presented experiments on machine translation on PADIC (Parallel Arabic DIalect Corpus). Among them, experiments have been conducted on Algerian dialect. In this work, the authors have been confronted to a problem due to the fact that their lexicon has been induced from PADIC. The number of entries is weak and more especially a word in PADIC has only one way to write it, since the rules used to write PADIC were inspired from the way of writing MSA. This lack of varieties of a word leads to weak results in terms of BLEU. In fact, the training corpus was built by hand and not harvested from social networks. In (Harrat et al., 2014), the authors proposed several resources: morphological analyzer, diacritization and also an Algerian dictionary. This latter is composed of words extracted from the dictionary of BAMA (Buckwalter, 2002) that have been adapted to the dialect, but unfortunately the authors have not treated the Arabizi nor the foreign words.

The resource, we propose to develop, will help to solve one of the present phenomenon in natural language processing related to Arabic dialect: the profusion of texts written in Arabizi in social networks. This issue has been addressed in several works such as in: (Darwish, 2013), the author proposed an approach to identify and to convert Arabizi into Arabic characters. He used words and sequence-level features to identify Arabizi that is mixed with English. In (Al-Badrashiny et al., 2014), the authors presented a system that uses a finite state transducer trained at the character level to generate all the possible transliterations for the input Arabizi words. In (van der wees et al., 2016), authors proposed an Arabizi-to-Arabic transliteration pipeline that combines character level mapping with contextual disambiguation of Arabic candidate words.

In our work, comparatively to the three last ones, we consider a word written in Arabizi such as any other word in the Algerian dialect. Consequently, we do not want to identify it and make a particular treatment to convert it into Arabic script. In fact, the Arabizi represents the real world in the social networks and particularly for those used by the Maghrebi people.

3. Corpus

To build an Algerian dialect lexicon, we decided first to collect a large corpus from comments posted by people related to Algerian videos. That is why, we harvested data from YouTube by using the Google's API [1] that allows users to search for videos that match specific criteria and retrieve all information and comments of these videos. To harvest, we chose few keywords to form queries in order to retrieve videos concerning national news, Algerian celebrity, local football, etc. Table 3. shows some figures before and after preprocessing the collected data, where $|C|$ is the number of comments, $|W|$ is the number of words and $|V|$ is the vocabulary size.

	Raw corpus	Cleaned Corpus
$\|C\|$	1.3M	1.1M
$\|W\|$	20M	17.7M
$\|V\|$	1.3M	0.99M

Table 1: The collected YouTube Algerian Dialect Corpus.

We can mention that after the cleaning process, the corpus has been reduced by around 15% and the vocabulary by around 24%.

4. Lexicon learning method

Because people, in social networks, do not mind about the spelling, a word may be written according to its pronunciation or to the one supposed by the user. One can borrow French words with foreign letters correponding to sounds the users have not in his tongue-mother such as /p/, /v/ and /g/ and adapt them to the dialect. For automatic speech recognition this could constitute a problem since the original pronunciation of the word is altered. For example, the French word *Problème* (*/problɛm/*) will be pronounced in Algerian dialect */broblɛm/*. Transliteration of foreign words are let to the goodwill of the users. Our objective is to produce automatically all the different forms of a word according to the writing varieties presented in the introduction. Each entry of this lexicon will be associated to all its different forms of writing harvested from YouTube. Another

[1] Available at: https://developers.google.com/YouTube

important motivation for learning automatically this lexicon is the fact that in dialect, people create new words frequently. That is why learning automatically such a lexicon is necessary to cope with the dynamic of the evolution of the lexicon.

4.1. On the need of comparable dialect corpus

In order to identify words, which are related to each others, we need to build automatically a comparable corpus. In a previous work (Abidi et al., 2017), we addressed the difficult issue of matching comments from YouTube for a vernacular language (Algerian dialect) for which no writing rules do exist. The method we propose is based on the concept of learning multilingual word embeddings (Word2Vec). The objective is to find a list of words that could be correlated to a lexical entry whatever the language. This method has permitted to find a list of variations of the same word. Then these words have been exploited in the matching process of documents. The word2Vec approach has been iterated to improve, at each step, the quality of the supposed comparable documents. This method achieved good results and allowed us to build a comparable Algerian dialect corpus named CALYOU: A comparable spoken Algerian corpus extracted from YouTube.

4.2. Training method

In the following, we propose to detail the method we developed to generate an Algerian dialect lexicon. Because the lexical variability in Algerian dialect is very high, in other words, each word could be written in several ways and because, the dialect evolves frequently, we propose to learn the dictionary automatically from social networks. Regularly, this dictionary could be enhanced by running again the proposed method.

The collected corpus from YouTube is transformed into a comparable document by gathering the comments, which are similar by using the method proposed in (Abidi et al., 2017). In fact, the comparable corpus is obtained in an iterative process where at each step, we refine the quality of the comparability of the corpus. At each iteration, two vocabularies are produced: a Correlated Word Lexicon (CWL) and an Algerian Dialect Lexicon (ADL) (see Figure1). CWL is used to produce comparable documents and ADL is the expected Algerian Dialect Lexicon.

CWL and ADL are modified at each iteration, when CWL is refined, the quality of the comparable corpus is improved and a fortiori, the entries of ADL will be more and more precise. Each entry of ADL will be represented by a word and its different ways of writing it. An entry in this dictionary could be written in Arabic (MSA or dialect) or in Latin script (Arabizi, French and sometimes in English).

4.2.1. Learning CWL

For learning CWL, we decided to use Word2Vec to retrieve the correlated words. To do that, for each word (w_s) of the corpus, where s is the Arabic or the Latin script, we learned its correlated words ($w_{\bar{s}}$), where $\bar{s}$ is a script different from s. We opted for a continuous bag of words (CBOW) method (Mikolov et al., 2013) with a sliding window of 100. This size has been fixed after several tests. This large number is explained by the fact that all the comments concerning the same video have been concatenated into one document (Abidi and Smaïli, 2017a).

For each w_s, we keep its n best correlated words $w_{\bar{s}}$. This process is used for each word of the corpus, at the end, we achieve a list of words and their n best correlated words. From this list, all the entries that occur more than α are inserted into CWL. For the others, for each w_s^i, all the words $w_{\bar{s}}^j$ respecting the following constraints:

$$\begin{cases} N(w_{\bar{s}}^j) > \alpha \\ or \\ S(w_s^i) = S(w_{\bar{s}}^j) \end{cases} \quad (1)$$

are included into CWL. Where $N(x)$ is the occurrence of x, α is set empirically to 1000 and S is a function that encodes phonetically a word. The constraint represented by S takes into account the variability of writing a form in accordance to its pronunciation. For that, the function S is implemented by using Soundex (Aqeel et al., 2006), a phonetic algorithm for indexing by sound. Words are encoded by taking advantage of their phonetic form. If two words have the same code, we can conclude that, one is the transliteration of the other. Soundex proposes to replace each letter by the index of a group of characters. Each group is constituted by the graphemes corresponding to the similar class of sounds. We obviously adapted the original correspondence table in order to take into account the graphemes of our corpus (see Table 2). The characters of

English character	Index	Arabic character
A E H I O U W Y	0	ى و ه ع ح ا
B P F	1	ب ف
C S K G J Q X Z	2	ك ق غ ص ش س ز ج ح
D T	3	ظ ط ض ذ د ث ت
L	4	ل
M N	5	م ن
R	6	ر

Table 2: Encoding correspondence table

Group 0, are ignored unless they appear in the first position of a word. Encoding consists in keeping the first character without any change and the following are encoded in accordance to Table 2. Any word will be represented by a letter followed by three digits. For example, the encoding process of the dialect word **حومة** in accordance to the encoding Table 2 and to the transliteration Table 3, will propose two codes: *H500* and *7500*. While the words *Houma* and *7ouma* will be encoded respectively *H500* and *7500*. Consequently, this allows to make a correspondence between the three latter words: **حومة**, *Houma* and *7ouma*. For more details see (Abidi and Smaili, 2017b). In Table 4, we give an example of CWL entry [2].

4.2.2. Learning ADL

As for training CWL, we used the CBOW (Mikolov et al., 2013) approach to constitute the Algerian dialect lexicon

[2]The translation in English are proposed by the authors for readability

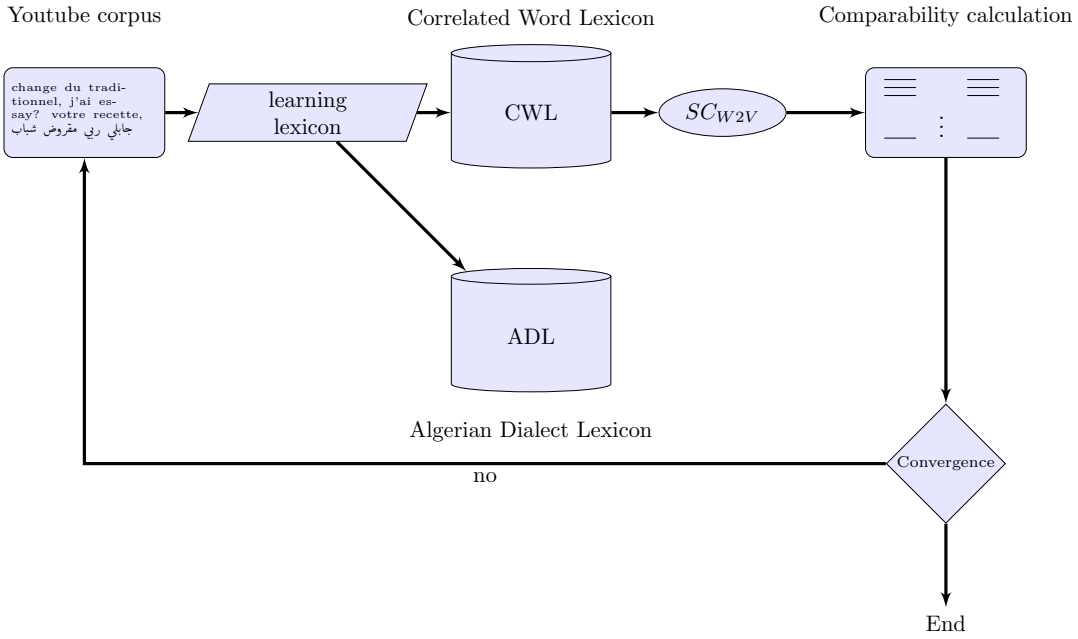


Figure 1: The training process

Arabic letter	Trans	Arabic letter	Trans	Arabic letter	Trans
ا	2 a e i	س	s	ع	‘ 3 a e
ب	b p	ش	sh ch	غ	r gh
ت	t	ص	s	ي	y i
ث	t th	ض	d	ذ	d dh
ج	j dj	ط	6 t	ر	r
و	w u ou	ك	k c	ق	9 q k c
خ	5 kh	ل	l	ظ	d
ح	7 h	م	m	أ	a
د	d	ن	n	أُ	u o
ف	f v	ه	h	إ	i
ز	z	ة	h a	ا	a

Table 3: Codes of Arabic-Latin transliteration

سعود (*Saud*)	saoud (*Saoud*) therories (*terrorist*) saudi (*Saudi*) alzarooni(*Al_Zarouni*) saoudi(*Saudi*) iran (*Iran*) terrorist (*terrorist*)

Table 4: An example of CWL

(ADL). In order to do so, a word (w_s) and its n best correlated words ($w^j_{s+\bar{s}}$) are retained for the next treatments. Where $s + \bar{s}$ means any kind of script (Arabic or Latin). This process is used for each word of the corpus, which achieves a list of words and their n best correlated words. n is determined empirically and in our experiments it has been fixed to 40. From this list, each entry will be processed to find its accurate declensions and will constitute, if appropriate, an ADL entry. Two cases have to be examined.

Same script of words

An entry w^i_s with its correlated words w^{ij}_s, which are written in the same script are included into ADL, if they respect the following constraint: $R(w^i_s) = R(w^{ij}_s)$. Where $R(x)$ is a function which removes vowels from x. This is motivated by the fact of the high ambiguity of writing in Arabizi since we write in Arabic by using another alphabet. In fact, a user who writes in Arabizi, sometimes does not find easily the exact Latin sound corresponding to the Arabic one, consequently in replacement he allows himself to take what he considers being the closest sound in Latin script. This operation will help to capture a word and its different ways to write it.

Different script of words

A new entry is inserted into ADL, composed of w_s and $w_{\bar{s}}$ with s corresponding to the Latin script if they respect the following constraint: $\exists i, w^i_s \in L(T(w_s))\ and\ w^i_s = w_{\bar{s}}$. With $T(x)$ is the transliteration of x and $L(T(x))$ is the list of possible transliterations of x. The transliteration is done in accordance to Table 3. This allows to associate a word written in Latin with words written in Arabic script .
The use of the procedures mentioned before allows to produce an Algerian Dialect Lexicon. An example is given in Table 5. One can remarks that, in comparison to Table 4, the entry of the word سعود *(Saud)* in ADL is more accurate. Only words related to this entry are kept and other words that are correlated such as *terrorist* and *Iran* are discarded.

سعود (Saud)	sa3oudi (Saudi) saud (Saud) saudi (Saudi) saoudia(Saudi Arabia) saoudi(Saudi) soudia (Saudi Arabia) saudia (Saudi Arabia) saoud(Saud)

Table 5: An example of ADL

5. Results

It is difficult to evaluate automatically the quality of a lexicon produced by an automatic method. Even if we do

have a measure, we need a reference test lexicon to evaluate it. For our experiments, we decided to test the quality of the produced lexicon by using, the classical retrieval information: Recall, Precision and F-measure, on a test corpus. Furthermore, we tested also the evolution of the number of words in the lexicon, in accordance to the iterative Word2Vec process.

To calculate the F-measure, we need a reference test lexicon. To the best of our knowledge, there is no Algerian dialect lexicon similar to ADL, on which we can evaluate our method. That is why, we decided to build semi-automatically a reference test lexicon. We collected data from social networks, and for each word, we used the algorithm based on Soundex described in Section 4.2.1. to get all the words sharing the same phonetic codes. Then this list is cleaned and updated by a human being. To illustrate this, in Table 6, we give an example concerning the word يشفيها (*It cures her*) obtained by Soundex. In this list, some words are similar to the entry, but others (those which are written in bold) have different meanings that should be discarded. Consequently, a cleaning process is done on each entry: removing the words (bold examples in Table 6) that are not related to the entry, removing the words (in italic in Table 6) which are similar, but they do not have the exact meaning as the entry and adding the missed words (examples in blue in Table 7).

This process led to a test lexicon of 560 entries, with an average number of forms by entry of 6, a maximum of 17 and a minimum of 1.

يشفيها	*ychafi ychafih* **yechf ychoufo yechfo** yachfih ichfiha ychafiha yachfiha *ychafih* ichafiha *yachefih*

Table 6: An example of an automatic extraction of a potential entry of the test lexicon

يشفيها	ychafiha yachfiha ichafiha ichfiha yechfiha ychfiha yechafiha

Table 7: An example of an entry of the test reference lexicon

In Figure 2, we plot the evolution of Recall, Precision and F-measure values for each Word2Vec iteration. The curves show clearly that the three measures progress. From the 17^{th} iteration, the three values are close to each others and from the 20^{th} iteration, the Precision decreases. Since we would like to propose an accurate lexicon, we decided to stop the iterative process, when the Precision starts decreasing. At the beginning of the training process, the F-measure is bad, since at the initial step, the CBOW method runs on articles composed of bulk comments. Theses comments are dispatched over the documents, consequently the CBOW process is not able to retrieve similar words in terms of graphemes, which makes the retrieved Algerian dialect lexicon not accurate. When we align the comments and by injecting them into the learning process, we get a better training corpus that leads to an Algerian dialect lexicon with more entries as illustrated in Figure 3. This curve illustrates the evolution of the number of entries in the Algerian Dialect Lexicon for which 85% of the entries have been added in the ten first Word2Vec iterations. But as mentioned before, the iterative process should be stopped when the Precision starts decreasing.

In Figure 4, we illustrate the added number of forms between the first and the last launch of the iterative Word2Vec process. One remarks that, when Word2Vec is launched at the beginning (gray bars), 84 entries have 5 different ways to be written, while at the end of the process (black bar), this number increases to 320. The number of entries in the dictionary having more than 5 forms, at the first launch of Word2Vec process, is equal to 201. At the end of the process, this number jumps to 1145. This figure shows that the distribution of entries with only one transliteration, at the end of the iterative process, is 35%, while the distribution of entries with more than 30 forms represents 7%. This last result strengthens the fact, that the variability of the Alge-

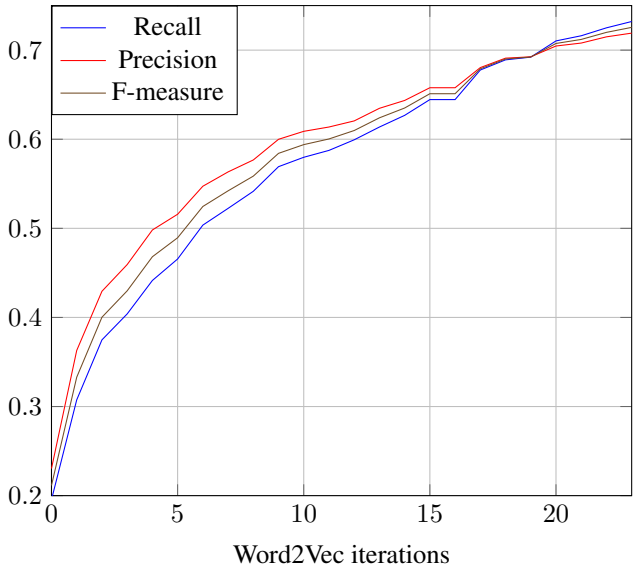


Figure 2: Evolution of the Recall, Precision and F-measure for each Word2Vec iterations

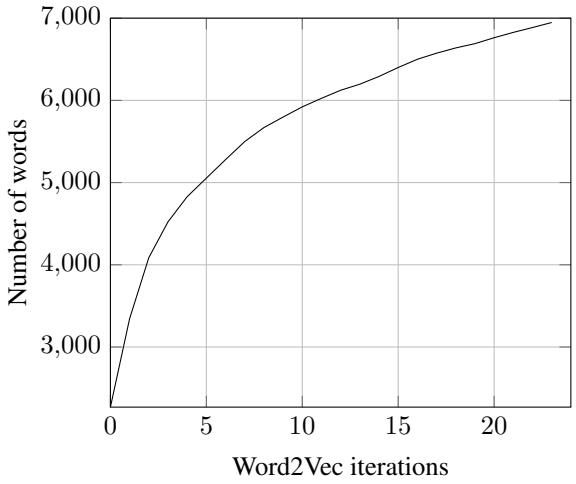


Figure 3: The evolution of the number of words in ADL in terms of Word2Vec interations

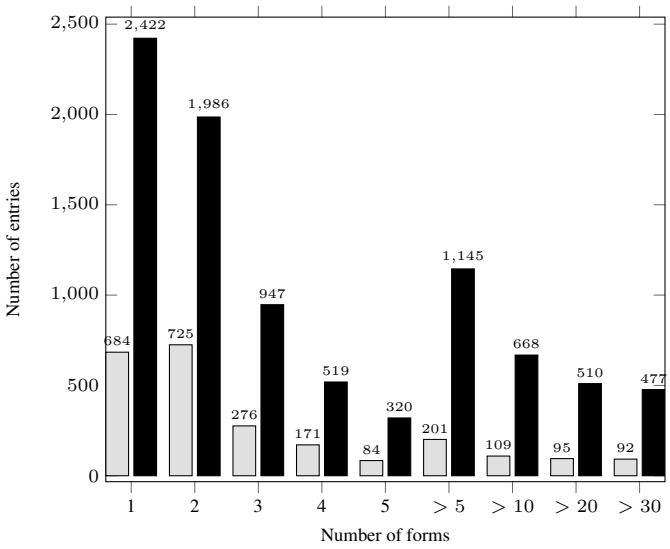


Figure 4: The progression of the number of forms between the first and the final iterative Word2Vec process

rian dialect is very high and consequently it necessitates an automatic process to build a dialectal lexicon.

In figure 5, a part of the built ADL is illustrated. One can remark that the entries are written either in Arabic or in Latin script. For each entry there is one or several forms (transliterations). Some entries are in French, but some of them are miswritten as explained in this paper, such as *comentair* which should be written *commentaire*. The entry مستر corresponds to the English word *mister*, one can notice that some transliterations are not well written in English, but this corresponds to how the users wrote them. Some entries have several transliterations such as يرحمك, which has 67 different forms.

```
<?xml version="1.0" encoding="UTF-8"?>
<Lexique Nb_Translitteration_possibles="6947">
<"شوربة" Transliteration=" chorba" />
<"مستر" Transliteration=" mester mister mstr" />
<"ميستر" Transliteration=" mester mister mstr" />
<"يرحمك" Transliteration=" yr7mk yr7mak yrhamak
 yarhamek yarhemak irahmk yarhamk yr7mek yere7mek yarhamak
 yarhemek yrhmk yar7mak yarhmak yarhmek yar7mik
 yarahmak yar7mek yarahmk yerhemek yarahmek yerehmek yerhamek
 yer7mak yare7mek yerhamak yer7mek yerehemek yarhmeke rahimaka
 yrahmek yrahmak irahmak irhmak irahmek yra7mk irhmk yrehmak
 yera7mak yerehmk yera7mek yrehmek yara7mak yarehmek yara7mek
 yerahmeke yarehmak yarhmk yerhmk yarhmeek yra7mak ir7mak
 yra7mek yarhamoka yrehmk yar7mk yrahmk ira7mak irehmek
 yerhmek yerahmk yrhmek yerahmek yerhmak yerahmak yrhmak
 yarahemek" />
 <"كاشير" Transliteration=" kachir" />
 <"film" Transliteration="فيلم فلم فليم" />
 <"comentair" Transliteration="كومونتار كومنتار" />
```

Figure 5: Example of some entries of the ADL produced by the iterative Word2Vec

6. Conclusion

In this article, we present an iterative multilingual word embeddings approach, which allowed to make comparable an Algerian dialect corpus, from which we built automatically a lexicon. Each word of this lexicon is associated to its different transliterations, the method led to a dictionary of 6947 entries. An entry may have a minimum of one transliteration and a maximum of 71. We observed, that 7% of the entries have more than 30 transliterations. This dictionary has been tested by using the Recall and Precision measures on a lexicon of 560 entries built semi-automatically. The iterative method of building the dictionary has been stopped when the precision has started decreasing. This method achieved a F-measure of 73%. Since the dialect is evolving everyday, one of the advantage of this approach is that the lexicon can be updated easily by harvesting new data. Also, this method could be used for any dialect for which data are available in the corresponding social network.

In the best of our knowledge, this kind of dictionary does not exist, it will be useful for different applications, for instance identifying parallel segments in comparable documents. It could be used to develop a based-dictionary transliteration system for Algerian dialect. In machine translation, this resource might be used to establish a correlation between a word and its corresponding transliterations and especially for proper names. This resource (ADL) is freely available at `http://smart.loria.fr/pmwiki/pmwiki.php/PmWiki/Lexicon`.

7. Bibliographical References

Abidi, K. and Smaïli, K. (2017a). An empirical study of the Algerian dialect of Social network. In *ICNLSSP 2017 - International Conference on Natural Language, Signal and Speech Processing*, Casablanca, Morocco.

Abidi, K. and Smaili, K. (2017b). How to match bilingual tweets? *Sixth International Conference on Natural Language Processing 25-26, 2017 in Sydney, Australia*.

Abidi, K., Menacer, M. A., and Smaïli, K. (2017). Calyou: A comparable spoken algerian corpus extracted from youtube. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm Sweden August 20-24 2017*.

Al-Badrashiny, M., Eskander, R., Habash, N., and Rambow, O. (2014). Automatic transliteration of romanized dialectal arabic. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning, CoNLL 2014, Baltimore, Maryland, USA, June 26-27, 2014*, pages 30–38.

Al-Badrashiny, M., Elfardy, H., and Diab, M. T. (2015). AIDA2: A hybrid approach for token and sentence level dialect identification in arabic. In *Proceedings of the 19th Conference on Computational Natural Language Learning, CoNLL 2015, Beijing, China, July 30-31, 2015*, pages 42–51.

Al-Sabbagh, R. and Girju, R. (2010). Mining the web for the induction of a dialectical arabic lexicon. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*.

Aqeel, S. U., Beitzel, S. M., Jensen, E. C., Grossman, D. A.,

and Frieder, O. (2006). On the development of name search techniques for arabic. *JASIST*, 57(6):728–739.

Belgacem, M. and Zrigui, M. (2010). Automatic identification system of arabic dialects. In *Proceedings of the 2010 International Conference on Image Processing, Computer Vision, & Pattern Recognition, IPCV 2010, July 12-15, 2010, Las Vegas, Nevada, USA, 2 Volumes*, pages 740–749.

Buckwalter, T. (2002). Buckwalter arabic morphological analyzer version 1.0. *Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Catalog No.LDC2002L49*.

Darwish, K. (2013). Arabizi detection and conversion to arabic. *Computing Research Repository*, abs/1306.6755.

Dey, A. and Fung, P. (2014). A hindi-english code-switching corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014.*, pages 2410–2413.

Harrat, S., Meftouh, K., Abbas, M., and Smaïli, K. (2014). Building Resources for Algerian Arabic Dialects. In *15^{th} Annual Conference of the International Communication Association Interspeech*, Singapour, Singapore. ISCA.

Harrat, S., Meftouh, K., Abbas, M., Jamoussi, S., Saad, M., and Smaïli, K. (2015). Cross-dialectal arabic processing. In *Computational Linguistics and Intelligent Text Processing - 16^{th} International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part I*, pages 620–632.

Meftouh, K., Harrat, S., Jamoussi, S., Abbas, M., and Smaïli, K. (2015). Machine translation experiments on PADIC: A parallel arabic dialect corpus. In *Proceedings of the 29^{th} Pacific Asia Conference on Language, Information and Computation, PACLIC 29, Shanghai, China, October 30 - November 1, 2015*.

Menacer, M., Mella, O., Fohr, D., Jouvet, D., Langlois, D., and Smaili, K. (2017). Development of the arabic loria automatic speech recognition system (alasr) and its evaluation for algerian dialect. In *Third International Conference On Arabic Computational Linguistics, Dubai, November 2017*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

van der wees, M., Bisazza, A., and christof Monz. (2016). A simple but effective approach to improve arabizi-to-english statistical machine translation. page 43.

Yoder, M. M., Rijhwani, S., Rosé, C. P., and Levin, L. (2017). Code-switching as a social act: The case of arabic wikipedia talk pages. In *Proceedings of the Second Workshop on Natural Language Processing and Computational Social Science, pages 73–82, Vancouver, Canada, August 3, 2017.*

Candidate Ranking for Maintenance of an Online Dictionary

Claire Broad[1,2], **Helen Langone**[1], **David Guy Brizan**[2]
[1]Dictionary.com
555 12th Street, Oakland, CA 94607
[2]University of San Francisco
101 Howard Street, San Francisco, CA 94105
{Claire.Broad, Helen.Langone}@dictionary.com, dgbrizan@usfca.edu

Abstract

Traditionally, the process whereby a lexicographer identifies a lexical item to add to a dictionary – a database of lexical items – has been time-consuming and subjective. In the modern age of online dictionaries, all queries for lexical entries not currently in the database are indistinguishable from a larger list of misspellings, meaning that potential new or trending entries can get lost easily. In this project, we develop a system that uses machine learning techniques to assign these "misspells" a probability of being a novel or missing entry, incorporating signals from orthography, usage by trusted online sources, and dictionary query patterns.

Keywords: Neologisms, machine-readable dictionary, ranking

1. Introduction

Dictionaries are databases in which the primary entities are words. Like a (non-temporal) database, a dictionary's contents are frozen in time (Guthrie et al., 1996; Labov, 2011; Curzan, 2012). Therefore, for a dictionary to remain relevant, new lexical entries for entirely new words – neologisms – must be added. This maintenance is important for machine-readable dictionaries (MRDs) as well as those built for human consumption.

We have created and continue to maintain a descriptive, general-purpose dictionary of American English[1]. Through our publicly-available web site, this lexical database is searchable by any of our users. Each lexical item may include one or more spellings, parts of speech, definitions, pronunciations, origins, examples of usage, and other information. Each month, our site hosts approximately 70 million users who collectively generate more than 450 million searches.

The content of our site is maintained by lexicographers. Some of this maintenance involves researching candidate lexical entries drawn from a number of sources, including unmatched queries: users' searches on our site which fail to match an item in the database. Given the size of our database and the number of unmatched queries, the work of prioritizing the items to be considered for inclusion is labor-intensive and somewhat subjective. Figure 1 contains an overview of our procedure for maintaining our dictionary.

In the interest of establishing a reasonable scope for this project, we limit our focus to single-word items. However, we believe that the same process could be effectively applied to multiword queries with some adjustment for the specific lexical considerations of phrases.

The goal of this effort is the production of a ranking for the unmatched query list to help our lexicographers identify potential candidate entries and focus attention on the items most qualified to be included in the lexicon. This work is made difficult because our candidates have no contextual information which has proven useful in the past (Weischedel et al., 1993; Nakagawa et al., 2001). This classification has the additional complexity of time sensitivity: for example, a previously rejected item may be included at a later point because of changes in usage or register (Curzan, 2012).

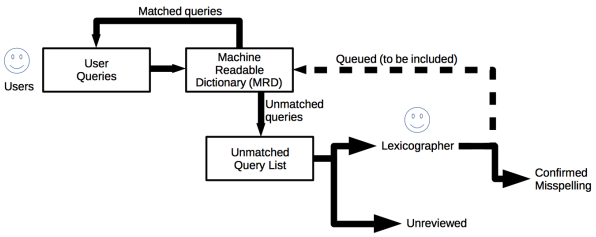


Figure 1: *Overview of Maintenance Procedure.*

While our efforts are narrowly focused on our site's content, we believe that the principles we apply here are germane to research into or practice of maintenance work for any lexical database. Since the lexicon is central to many efforts in natural language processing (Varathan et al., 2010), we see the possibility of broad appeal and obvious extensions to higher-level NLP functions, such as parsing or semantics (Al-Shalabi and Kanaan, 2004).

2. Related Work

According to one analysis (Baayenab et al., 2015) of recent and historical corpora of American English, (Davies, 2010; Davies, 2008), there may be more than 300 neologisms in the English language annually. Two sources of neologisms are *netspeak*, special "insider" language adopted by the technologically inclined starting in the 1990s, and the largely vilified *chatspeak*, which includes respellings of common words ("gr8" for "great") and abbreviations for common phrases ("brb" for "be right back") used to save time (Squires, 2010).

[1]Many non-American English lexemes (eg. "colour") are also included in our dictionary as variant spellings of their American counterparts.

Our work is related to dictionary construction in general. Although translation dictionaries were popular before his publication, Robert Cawdrey (Cawdry, 1966) is credited as having built the first monolingual dictionary of the English language in 1604 in response to the variant spellings his contemporary compatriots used, some due to the encroachment of foreign words.

The database approach for dictionaries followed more than 300 years after Cawdrey. While its stated purpose is to find historical antecedents to current language, the "Dictionary on Computer" project (Wang, 1969) describes a system for encoding lexical entries in a Chinese dictionary, alluding to the maintenance and extension of the lexicon. This work is largely steeped in the minutiae of the period in which it was written – punch cards, etc. Still, it addresses the contemporary problem of maintaining a set of lexical entries, including symbols (correlated to spelling in English) and pronunciation.

MRD-usable extensions to the database format include automatic inference of part-of-speech categories, inclusion of subcategorization frames (Boguraev et al., 1987; Sennrich and Kunz, 2014), applications to specific domains (Ji et al., 2007), or dictionaries for machine translation (Melamed, 1998; Chen et al., 1999). Building on these better-developed data are tools such as WordNet (Miller, 1995) and others which construct a graph network on the lexicon.

Theoretically, our work is inspired by the observations of (Hodges, 1972). Specifically, we do not accept the 13th century description of English spelling as chaotic. Instead, we see it as an "incompletely systematic" representation of a phonetic system which has resisted change while the spoken form of the language has welcomed loanwords and has been more inclusive of differing pronunciations, which could be anticipated (Hills and Adelman, 2015; Bromham et al., 2015; Steels and Kaplan, 1998; Longobardi et al., 2015) from the size, diversity and density of speakers from unique linguistic subcultures, dialects and registers. While English spelling is not as regular as, for example, Arabic (Al-Shalabi and Kanaan, 2004)[2] or Spanish, there are a number of patterns which we may exploit with techniques similar to those used in the phonotactic approaches for spoken language classification (Zissman and others, 1996).

3. Data

Our data is derived from two primary sources: lexical entries and queries to the site from February, 2017. The lexical entries are word types in American English which have been identified by lexicographers as being in common usage. The set of queries to the site contains all query strings (*matched* – those queries which have a corresponding entry or variant spelling in our MRD – and *unmatched*, which have no entry), as well as the monthly query count for each. The set of unmatched queries is ranked by the number of requests for each item. Of the unmatched queries, lexicographers have classified the 10000 most popular items, and the majority are verified as misspellings. We use portions of the lexical entries and queries to construct our test and training sets.

Our goal is the re-ranking of the unmatched queries, so that those that are strong candidates to be selected for inclusion in the dictionary appear at the top.

Type	True	Misspelling
Vowel Replacement	separate perceive	seperate percieve
Consonant Replacement	accommodate cynicism	accomodate synicism
Silent Letter Omission	government acquire	goverment aquire
Phonetic Spelling	rapport environment	repore enviorment

Table 1: Examples of misspell types

3.1. Test Set

The test data consists of unmatched queries from February 2017, ordered by the number of times each item was queried. Within this set, there are a few broad categories of orthographically similar character changes, primarily consisting of vowel replacements, consonant replacements, doubling or omission of characters, and silent letter omissions. We also find fully phonetic spellings in cases in which a word is pronounced differently from how it is spelled. Finally, although we find few top queries to the site involving instances of "slip of the finger" typos, we anticipate these errors. Table 1 contains examples of each of these categories.

From our observations of the historical user query patterns, we anticipate on the order of 1 valid class item for every 100 test items.

3.2. Training Set

In experimenting with different training set construction strategies, we found that downsampling the class of valid lexical entries to achieve the observed 1:100 class distribution was neither optimal nor robust – due to high variability in the valid class, the classification of the test set tended to vary greatly between trials. To mitigate this, we developed a novel ensembling technique, with the intention of increasing the influence of the 'best' valid items.

3.2.1. Valid class

We only consider single-word items, excluding all other entries. This results in over 100000 items. We prioritize the most recently added as explained below.

3.2.2. Invalid class

The only validated set of items in the invalid class consists of the previous month's 10000 most common misspellings, many of which must be rejected from use in the training set because they were also commonly searched in the current month and are therefore present in the test set. Because of this overlap, we have fewer examples (7871) of misspellings than required to match the distribution of the test

[2]The case for Arabic spelling may be more complicated than we make it out to be. For example, Modern Standard Arabic–a common "second dialect" of many in the Arabic-speaking world–may have many of the issues we allude to in English.

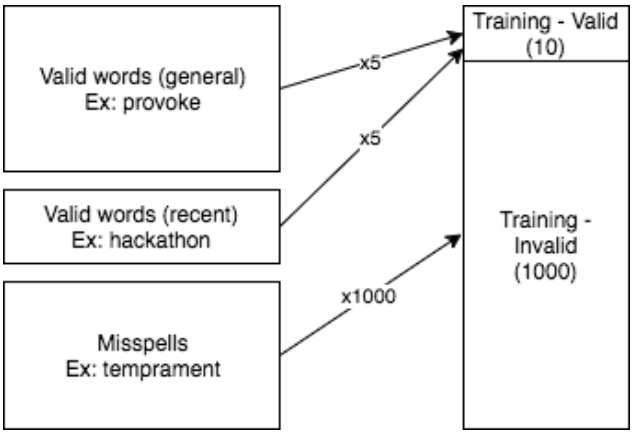


Figure 2: *Construction of a training set.*

set. We supplement the invalid class by generating the following random strings and ensure that there are no matches to the variant set.

- Plausible misspellings: a random lexical entry is selected and a string is replaced by an orthographically similar one, ideally replicating the types shown in Table 1
- Typos: a random item is selected from the primary lexicon and a character is replaced with one of its neighbors on the keyboard
- Random character strings of length up to 20 characters

4. Methods

Using the data described above, we generate a hypothesized probability that a candidate item from the test set will be adopted as a new lexical entry. The test set items are reordered by probability to create the final ranking.

4.1. Feature Extraction

The model includes a variety of features extracted from the attributes of the item, the query data, and word usage on the Internet at large.

4.1.1. Orthographic Features

We extracted the following orthographic features for each item in our training and test sets:

1. Presence of a known prefix (suffix) at the beginning (end) of the word
2. If (1) is true, is the remainder of the item a word from the variant set?
3. Presence of Greek/Latin roots (determined by whether any item in predetermined set of possible roots is present as a substring)
4. Count and proportion of vowels/consonants
5. Character length of the lexical item
6. Portmanteau: is the item a concatenation of two variant items – that is, does any single split of the item result in two substrings that are in the variant set

In addition, we encode all pairs of characters (including '-' and the starts and ends of words) using a variant of term frequency-inverse document frequency (tf-idf), where the frequency of a character pair in the item is compared against the frequency of that character pair in the full set of training items.

4.1.2. Query Pattern Features

To the orthographic features, we add data on the items' popularity. The goal with the query pattern features is to isolate not only the item's current popularity, but also whether it is of recent interest. A true neologism may have a large number of queries concentrated in a recent period of time, whereas very common misspellings would have consistently high queries. Therefore, we also include two percent change features:

- Over mean: comparing the number of queries of the item this month against its mean monthly queries in past months
- Year-over-year: comparing this month's queries against the queries of this item in this month of last year (to account for items of seasonal interest)

Furthermore, we include the percentage of months for which we have volume for user query data that the item.
For scalability reasons, these calculations are not performed on items that do not have queries in the current month. The traffic features for these items are populated with placeholder values within class that were determined via iteration to be most conducive to the significance of these features and result in a more effective model.

4.1.3. Usage in the Wild

Finally, we consider whether an item is being used elsewhere on the Internet. We found that one ideal source for this information was the Twitter feeds of news organizations, as they can be expected to be quite rigorous with spelling and include a variety of trending terms. We used the Twitter API to extract the language usage of 100 Twitter feeds from broadcast and print media outlets such as ABC News, Yahoo News and the Huffington Post.
273 of the 8115 items in the test set are present in the Twitter corpus. Some misspellings are present, along with a wide variety of strong candidate keywords. For example:

- Typos/misspellings: aquired, seige, beacuse
- Proper nouns: Supercell, Starbucks, Chromebook
- Coinages: deflategate, yuge, cuck
- Slang: turnt, rekt, janky
- Neologisms: petrichor, misophonia
- Loanwords: queso, agua, deux

4.2. Prediction Model

We generate a number of classifiers, each trained on a very small portion of the valid items, and prioritize those which are most effective at classifying misspellings correctly. For 1000 iterations, a small training set is randomly sampled,

Type	Count	Query Rank	Model Rank
Confirmed Misspelling	4304	3257	4787
In the Queue	35	3689	2558
Neither/unvetted	3774	4876	3240

Table 2: Change in average ranking by subset

consisting of 10 valid items (half recent additions, half from the main set of lexical entries) and 1000 invalid items, as shown in Figure 2. Using the random forest implementation in scikit-learn (Pedregosa et al., 2011), a forest of 10 estimators is trained on each set and persisted for use against the test set. We also store the mean score of this forest on predicting the other training sets. (For ease of computation, we use a random sample of 5 sets.) Each of these forests is then used to predict the probability of validity for the items in the test set, and their predictions are ensembled using the forests' respective scores as weights.

5. Results

Tested against query data from February 2017, we see strong results in predicting potentially valid items. We assess these results with two metrics.

Our first metric is the mean rank of the items in each class within the list of confirmed misspellings. With the caveat that an item may be accepted at a later date, we use this to assess, in a general sense, the rankings of misspellings among the predictions. Our assumption is that no more than one or two such shifts to validity would occur at a time, and thus would not skew this figure too egregiously.

Our second metric employs the "queue," items identified by the lexicographers as valid but not yet included in the dictionary for procedural reasons.

The test set consists of 8115 items:

- 37 items in the queue
- 4306 items on the confirmed misspells list

There are two items which are in both categories – instances of the aforementioned edge case where an item previously considered a common misspelling has subsequently been accepted as a valid item for our MRD. Table 2 shows the average ranking of test set items that are within one or neither of these sets, comparing rank as determined by query count alone versus the rank generated by the model.

Selected test items, with query count ranking and predicted validity ranking in parentheses:

- Slang + loanwords: boujee (45; 1), hola (32; 7), hygge (445; 9), adulting (5656; 19)
- Pop culture: pikachu (132; 4), harambe (842; 5), moana (1994; 8), festivus (4445; 33)
- Tech: youtuber (1476; 6), blockchain (7202; 12), ransomware (6258; 52)
- Politics: CPAC (4335; 3), alt-right (3826; 14), post-truth (3411; 16)

6. Discussion

The average ranking for all items in the queue, 2707, is well above the midpoint of 4058, as well as an improvement from the average ranking based on query count alone of 3257. Furthermore, the items which are on neither list have a higher average ranking than the confirmed misspells (3240 vs 4788). This is positive, as this is in many ways our target group – items which are neither obvious misspellings, nor obvious candidates, and thus have not been identified previously. Finally, the confirmed misspellings have moved down in rank, from 3257 to 4787.

We believe that these results indicate that our system is capable of identifying the same candidate terms that would be chosen by a lexicographer, as well as additional terms that would otherwise have stayed buried in the "misspell" bucket. Indeed, the rankings produced by this system have been adopted by our lexicographical content team as a tool for identifying keywords that merit further research.

We also believe that these results show the merit of our orthographic approach as a surrogate for American English pronunciation. We see evidence of this in that adopted items which are Anglicized, such as "boujee" are ranked higher than un-modified loanwords like "hygge." We anticipate applying this approach to different data and domains.

7. Conclusions and Future Work

The approach we describe is limited to single tokens queried on the site because of our focus and our available data. We are planning to verify our results with data from external sources and investigate how the same dictionary construction could be automated to benefit NLP applications which have temporal considerations.

Although this effort was naturally limited to American English by virtue of our data source and use case, we believe that the same basic principles could be applied to many languages. While many languages do not have the same complexities of orthography, they may be influenced by external pressures, so maybe more subject to the adoptions of calques. Therefore, our approach may require adjustments to the orthographic features to best suit the given language.

For future work, we will extend our approach to include multi-word expressions. To that end, we are also interested in applying new approaches such as deep learning using character-level embeddings on an LSTM network.

8. Bibliographical References

Al-Shalabi, R. and Kanaan, G. (2004). Constructing an automatic lexicon for arabic language. *International Journal of Computing & Information Sciences*, 2(2):114–128.

Baayenab, H., Tomascheka, F., Gahlc, S., Ramscara, M., and Baayen, R. H. (2015). The ecclesiastes principle in language change.

Boguraev, B., Briscoe, T., Carroll, J., Carter, D., and Grover, C. (1987). The derivation of a grammatically indexed lexicon from the longman dictionary of contemporary english. In *Proceedings of the 25th annual meeting on Association for Computational Linguistics*, pages 193–200. Association for Computational Linguistics.

Bromham, L., Hua, X., Fitzpatrick, T. G., and Greenhill, S. J. (2015). Rate of language evolution is affected by population size. *Proceedings of the National Academy of Sciences*, 112(7):2097–2102.

Cawdry, R. (1966). *Table Alphabetical of Hard Usual English Words*. Scholars Facsimilies & Reprint.

Chen, A., Kishida, K., Jiang, H., Liang, Q., and Gey, F. C. (1999). Automatic construction of a japanese-english lexicon and its application in cross-language information retrieval. In *In Joint ACM DL/ACM SIGIR Workshop on Multilingual Information Discovery and AccesS (MIDAS*. Citeseer.

Curzan, A. (2012). *The Secret Life of Words: English Words and Their Origins*. The Great Courses.

Davies, M. (2008). The corpus of contemporary american english as the first reliable monitor corpus of english. *Literary and linguistic computing*, 25(4):447–464.

Davies, M. (2010). The corpus of historical american english: 400 million words, 1810–2009,(2010). *URL: http://corpus.byu.edu/coha/*.

Guthrie, L., Pustejovsky, J., Wilks, Y., and Slator, B. M. (1996). The role of lexicons in natural language processing. *Communications of the ACM*, 39(1):63–72.

Hills, T. T. and Adelman, J. S. (2015). Recent evolution of learnability in american english from 1800 to 2000. *Cognition*, 143:87–92.

Hodges, R. E. (1972). Theoretical frameworks of english orthography. *Elementary English*, 49(7):1089–1105.

Ji, L., Lu, Q., Li, W., and Chen, Y. (2007). Automatic construction of a core lexicon for specific domain. In *Advanced Language Processing and Web Information Technology, 2007. ALPIT 2007. Sixth International Conference on*, pages 183–188. IEEE.

Labov, W. (2011). *Principles of linguistic change, cognitive and cultural factors*, volume 3. John Wiley & Sons.

Longobardi, G., Ghirotto, S., Guardiano, C., Tassi, F., Benazzo, A., Ceolin, A., and Barbujani, G. (2015). Across language families: Genome diversity mirrors linguistic variation within europe. *American journal of physical anthropology*, 157(4):630–640.

Melamed, I. D. (1998). Empirical methods for mt lexicon development. In *Conference of the Association for Machine Translation in the Americas*, pages 18–30. Springer.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Nakagawa, T., Kudo, T., and Matsumoto, Y. (2001). Unknown word guessing and part-of-speech tagging using support vector machines. In *NLPRS*, pages 325–331. Citeseer.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Sennrich, R. and Kunz, B. (2014). Zmorge: A german morphological lexicon extracted from wiktionary. In *LREC*, pages 1063–1067. Citeseer.

Squires, L. (2010). Enregistering internet language. *Language in Society*, 39(04):457–492.

Steels, L. and Kaplan, F. (1998). Spontaneous lexicon change. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 1243–1250. Association for Computational Linguistics.

Varathan, K. D., Sembok, T. M. T., and Kadir, R. A. (2010). Automatic lexicon generator. In *Information Retrieval & Knowledge Management,(CAMP), 2010 International Conference on*, pages 24–27. IEEE.

Wang, W. S. (1969). Project doc: Its methodological basis. In *Proceedings of the 1969 conference on Computational linguistics*, pages 1–22. Association for Computational Linguistics.

Weischedel, R., Schwartz, R., Palmucci, J., Meteer, M., and Ramshaw, L. (1993). Coping with ambiguity and unknown words through probabilistic models. *Computational linguistics*, 19(2):361–382.

Zissman, M. A. et al. (1996). Comparison of four approaches to automatic language identification of telephone speech. *IEEE Transactions on speech and audio processing*, 4(1):31.

Language adaptation experiments via cross-lingual embeddings for related languages

Serge Sharoff
Centre for Translation Studies,
University of Leeds, LS2 9JT, UK
s.sharoff@leeds.ac.uk

Abstract

Language Adaptation (similarly to Domain Adaptation) is a general approach to extend existing resources from a better resourced language (donor) to a lesser resourced one (recipient) by exploiting the lexical and grammatical similarity between them when the two languages are related. The current study improves the state of the art in cross-lingual word embeddings by considering the impact of orthographic similarity between cognates. In particular, the use of the Weighted Levenshtein Distance combined with orthogonalisation of the translation matrix and generalised correction for hubness can considerably improve the state of the art in induction of bilingual lexicons. In addition to intrinsic evaluation in the bilingual lexicon induction task, the paper reports extrinsic evaluation of the cross-lingual embeddings via their application to the Named-Entity Recognition task across Slavonic languages. The tools and the aligned word embedding spaces for the Romance and Slavonic language families have been released.

Keywords: Word embeddings, Related languages, Cognate words, Comparable corpora

1. Introduction

Parallel corpora play an important role in many multilingual NLP applications, such as Machine Translation, Cross-Lingual Text Classification or Information Retrieval. However, the topics and genres of parallel corpora are limited even for better resourced languages, e.g., resources are scarcer outside of the official documents of Europarl and the United Nations (Koehn, 2005; Eisele and Chen, 2010). Also, even if each individual language has reasonably good parallel resources, such as Polish and Russian aligned with English, it is difficult to find a large parallel corpus, which contains this specific, e.g., Polish-Russian, language pair.

Monolingual corpora can be substantially bigger and more varied in comparison to parallel ones. Comparable corpora of different levels of comparability (Sharoff et al., 2013) can be used for induction of bilingual lexicons from small seed dictionaries. The present paper follows an influential study (Mikolov et al., 2013), which presented a method for building *multilingual* embedding spaces. In addition to a model with a seed bilingual dictionary, it also introduced constraints on what its authors call "morphological structure" (actually the Levenshtein Distance) for keeping only the cognate words in the output.

However, further work on bilingual lexicon induction did not include the use of cognates, especially in the context of related languages. The importance of utilising links between related languages can be illustrated by the use of Machine Translation via a pivot language. A simple dictionary transfer from Ukrainian into Russian followed by MT for the better resourced Russian-English pair easily beats MT translating from Ukrainian directly into English using far smaller resources (Babych et al., 2007). Overall, many lesser resourced languages can benefit from Language Adaptation by applying the models developed for the better resourced ones.

The present study advances the state of the art by combining existing techniques of building cross-lingual embedding from comparable corpora with the Weighted Levenshtein distance, when the weights are themselves obtained from the seed dictionaries, see Section 3. In addition to intrinsic evaluation of the parameters of bilingual lexicon induction, the quality of cross-lingual embeddings can be measured extrinsically through accuracy of their use in downstream tasks, in particular, Named Entity Recognition, see Section 4. Both intrinsic and extrinsic evaluations show considerable improvements from the use of Language Adaptation.

2. Related studies

Starting from earlier work on Neural Language Models, a common way of representing word meanings is via word embeddings built from predictions of word neighbours using neural networks (Bengio et al., 2003; Mikolov, 2012). Recently, the Facebook group developed FastText, an updated method for producing monolingual embeddings by using information from character ngrams (Mikolov et al., 2017), i.e., a word embedding vector is:

$$v(w) = w2v(w) + \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}} x_n \qquad (1)$$

where $w2v$ is the standard word embedding of w (using the skip-gram model), while $\mathcal{N}$ is the set of ngrams derived from this word, x_n are their respective embeddings.

Studies in extraction of bilingual lexicons from comparable corpora can be traced back to at least (Fung, 1995; Rapp, 1995), who described words via a vector of their collocates, translated some words using a seed dictionary and compared the vectors across the languages. Word embeddings offer a better way of building word vectors in comparison to the vectors of collocate counts (Baroni et al., 2014). Word embeddings across languages have been studied since (Klementiev et al., 2012). A seminal study, which transformed the field, was (Mikolov et al., 2013), which used a translation matrix (TM) trained on a seed bilingual dictionary to convert monolingual word embeddings into a shared space. That study was followed by other studies aimed at

Table 1: Alignments from Wikipedia for titles and words

Polish	Russian	English
Z życia marionetek	Из жизни марионеток	From the Life of the Marionettes
Wskaźnik jakości życia	Индекс качества жизни	Quality-of-life index
Character alignment for words:	m a r i o n e t e k м а р и о н е т о к	ż y c i a ж и з н и

improving the process of TM production, e.g., via Canonical Correspondence Analysis (Faruqui and Dyer, 2014), Global Correction (Dinu et al., 2014) or TM orthogonalisation (Artetxe et al., 2016).

A traditionally accepted model for this task is based on constructing a linear transformation matrix $\mathbf{W}$ by minimising the following objective:

$$\min_{\mathbf{W}} \sum ||\mathbf{W}e_i - f_i||^2 \quad (2)$$

where e_i and f_i are the respective embedding vectors in the two languages, which are supposed to be translations of each other according to the training set. The differences between the approaches are primarily in the method for building $\mathbf{W}$, e.g., by stochastic gradient descent (Mikolov et al., 2013), CCA (Faruqui and Dyer, 2014), multivariate regression (Dinu et al., 2014) or matrices from the SVD transform (Artetxe et al., 2016). The latter model ensures that $\mathbf{W}$ is an orthogonal matrix built using a closed form solution:

$$\mathbf{W} = \mathbf{V} \times \mathbf{U}^T \quad (3)$$

when $\mathbf{V}$ and $\mathbf{U}$ are the matrices from the SVD factorisation of $\mathbf{F} \times \mathbf{E}^T$, see (Artetxe et al., 2016) for justification and discussion.

The feature spaces with large number of dimensions also demonstrate a phenomenon of *hubness* (Radovanović et al., 2010), i.e., some vectors happen to be in close proximity to many other vectors. This makes them more common choices in the lexical retrieval tasks leading to a larger number of errors. Formally, a word w is mapped to a set of words $\mathcal{N}_k(w)$ for which this word is within their k nearest neighbours. Words with the largest $|\mathcal{N}_k(w)|$ are (typically unwanted) hubs. Often such words have restricted context of their use, e.g., *troops* (183), *retreated* (176), *cavalry* (156) are such hubs in the FastText English space induced from Wikipedia (the numbers in brackets refer to their $|\mathcal{N}_{20}|$ hubness index, i.e., there are 183 words for which the word *troops* is in the list of their 20 closest neighbours), while the median hubness index on the English Wikipedia is 5. Dinu et al. (2014) observe that hubness becomes more pronounced after linear transformation, since the objective for building the transformation matrix $\mathbf{W}$ leads to lower variance of the transformed vectors, which in turn means that the vectors (on average) are closer to each other (Dinu et al., 2014). They suggest a way of mitigating hubness by using Global Correction (GC), i.e., by downgrading the similarity ranks for the items proportionally to their hubness index.

The initial TM study (Mikolov et al., 2013) did suggest the use of the Levenshtein Distance (LD), as a filtering step. Similarly, filtering of cross-lingual embedding spaces via LD for the purposes of Statistical Machine Translation between related languages has been explored in (Rios and Sharoff, 2015). A manually developed set of rules for a Finite State Transducer (FST) was used for identification of cognates and borrowings in (Tsvetkov and Dyer, 2016). However, post hoc filtering improves precision at the expense of reduced coverage. The method suggested below operates at the ranking stage, while it also uses the Weighted Levenshtein Distance (WLD), a simpler alternative to FSTs.

3. Dictionary induction using cognates

3.1. Cross-lingual mapping

The method for cross-lingual mapping across related languages in this study consists of three steps:

1. automated collection of seed bilingual dictionaries;
2. determining weights for the Levenshtein distance from the seed dictionaries;
3. alignment of monolingual embeddings by linear transformation using global correction and weighted LD;

In a low resource setting, the seed dictionaries for related languages can be obtained from the titles of interlinked Wikipedia articles in two languages (iWiki links),[1] see examples of aligned titles in Table 1. This helps in modelling scenarios when few parallel texts are available, e.g., for the Polish-Russian pair (Polish is included in Europarl, Russian is in the UN corpus, but very few reliable resources are available for the Polish-Russian pair). The titles have been word-aligned using FastAlign (Dyer et al., 2013). The resulting word-level dictionaries have been filtered against the respective frequency lists, since the Wikipedia titles are dominated by relatively infrequent proper names.

In addition to providing the training lexicon, a seed dictionary can also be used to provide a character-level model for matching the cognates, see the part of Table 1 for examples of character alignment. The pairs of words from the training dictionary have been aligned on the character level (again using FastAlign in this study) to produce the probabilities of regular correspondences between the characters in the two languages. This character alignment model is particularly important when the two languages use different character sets, such as the case for Polish and Russian. For example, the characters with the highest probability for translating the Russian characters ф and л into Polish are respectively *f* and *ł*.

In the end the standard edit operations for computing the traditional normalised Levenshtein Distance can be

[1] github.com/clab/wikipedia-parallel-titles

weighted by the probabilities of their character-level alignments:

$$WLD(s_e, s_f) = 1 - \frac{\sum_{(e,f)\in al(s_e,s_f)}(1 - p(f|e))}{\max(len(s_e), len(s_f))} \quad (4)$$

where s_e and s_f are words in the two languages, al is a set of their alignments, $p(f|e)$ is the probability from the character alignment model. The distance is normalised by the length of the longest word. For convenience in comparing it with the cosine similarity, the value is flipped to represent greater similarity with larger values.

Given that even correctly aligned words from the Wikipedia titles for related languages are not necessarily cognates e.g., *wskaźnik* vs индекс ('index') from Table 1, the process of getting the Levenshtein weights ran in two steps. In the first step, an initial estimate of the probabilities for characters was produced from *all* words in the seed dictionary. This was used for assessing the rough WLD between them. The most likely cognates according to this rough WLD were used as the input for the second iteration of character-level alignments. The WLD threshold for choosing the most likely cognates was determined for each language pair individually. Repeated application of these steps did not result in any improvements in detecting cognates.

The value of either LD or WLD can be used as a factor for scoring the translation suggestions:

$$score(s_e, s_f) = \alpha \cos(v_e, v_f) + (1-\alpha) WLD(s_e, s_f) \quad (5)$$

where v_e and v_f are vectors for respectively s_e and s_f in the cross-lingual embedding space, while α is the relative weight of the cosine similarity.

While the combined score is useful for producing bilingual dictionaries, it does not affect the bilingual embedding space by itself. A closed form solution for orthogonalisation as used in (3) helps in improving alignment quality in the general case, but it does not allow weight adjustment by taking into account the similarity between the cognates. An easy way for incorporating this information into the cross-lingual embedding space is by aligning the entire lexicons from the cross-lingual space using the WLD score from (5) and selecting the most similar words in this list. This far longer lexicon can be used instead of the seed dictionary for producing a new weight matrix from (3) for re-alignment of the already aligned cross-lingual space from the previous step. The rationale for this iteration is that we want to minimise the distance between the known cognates while preserving the orthogonality of the weight matrix. Again, while repeated application of these steps is possible, it did not produce better results, so the tables below present the results obtained after two iterations.

3.2. Experimental setup

This paper reports two sets of experiments. One experiment involved a replicable setting for the English-Italian language pair with the standardised embeddings and training / test dictionaries initially developed for (Dinu et al., 2014) and used in (Artetxe et al., 2016). Even though English and Italian are not closely related languages (English is a Germanic language, Italian is from the Romance family), a large number of English words are borrowings from Romance languages, primarily from French and Latin, so the WLD approach could work for the En-It pair as well. The test dictionary from (Dinu et al., 2014) includes both cognate word pairs, such as *academy / accademia*, and non-cognate pairs, such as *absolve / esimere* or *abysmally / malo*, which are also often questionable translation equivalents. Therefore, a cognate-only version of the En-It test set was produced by retaining only the words with the WLD value above 0.5, reducing the En-It test dictionary from 1869 down to 818 entries.

Table 2: Prec@1 for En-It

W2V vectors from (Dinu, et al. 2014)	
TM (Mikolov et al., 2013)	0.349
CCA (Faruqui and Dyer, 2014)	0.378
Orth (Artetxe et al., 2016)	0.393
GC (Dinu et al., 2014)	0.377
GC+LD	0.501
GC+WLD	**0.531**
FT vectors from (Mikolov et al., 2017)	
FT+TM	0.461
FT+Orth	0.529
FT+GC	0.477
FT+GC+Orth+WLD	**0.616**
MUSE (Conneau et al., 2017)	0.683
FT vectors for cognates only	
FT+TM	0.550
FT+Orth	0.614
FT+GC	0.575
LD $\alpha = 0$	0.298
WLD $\alpha = 0$	0.339
FT+GC+Orth+WLD $\alpha = 0.5$	0.584
FT+GC+Orth+LD $\alpha = 0.73$	0.669
FT+GC+Orth+WLD $\alpha = 0.73$	**0.692**
MUSE	0.719

A new set of embeddings produced by FastText has been used in the English-Italian experiments (labelled as FT in Table 2) in addition to the standardised embeddings as used in (Dinu et al., 2014; Artetxe et al., 2016). The FT embeddings have been the basis for the experiments with the Slavonic languages.

The experiments with the Slavonic languages also emphasise the low-resource setting, when large parallel corpora for the seed dictionary are not always available, so the seed dictionaries for the Transformation Matrices and the WLD weights came from the iWiki links (the Italian seed dictionary used in (Dinu et al., 2014) and (Artetxe et al., 2016) was derived from aligning Europarl).

3.3. Experimental results

The results listed in Table 2 confirm that orthogonalisation (Artetxe et al., 2016) and global correction (Dinu et al., 2014) improve the accuracy of translation detection in comparison to the baseline of (Mikolov et al., 2013). Embedding vectors produced by incorporating subword information (marked by FT in Table 2) also make a considerable positive impact. Adding the constraint of having or-

Table 3: Dictionary induction results for Slavonic languages

Dictionary induction without WLD

	sl-hr	sl-cs	sl-pl	sl-ru	ru-uk	cs-sk
Prec@1:	0.429	0.611	0.584	0.566	0.929	0.814
Prec@10:	0.688	0.868	0.842	0.818	0.976	0.971
MUSE, Prec@1:			0.724		0.942	

Dictionary induction with WLD

	sl-hr	sl-cs	sl-pl	sl-ru	ru-uk	cs-sk
Prec@1:	0.840	0.763	0.751	0.662	0.945	0.910
Prec@10:	0.963	0.973	0.977	0.883	0.994	0.996

thographic cognates (LD or WLD) improves the accuracy of dictionary induction further, often by a substantial margin. Even for the English-Italian pair, where the languages operate over the same alphabet, WLD outperforms LD because it assigns a very low cost to more common substitutions, e.g., $x \rightarrow s$ or $j \rightarrow g$ (*examined* $\rightarrow$ *esaminato* or *Jerusalem* $\rightarrow$ *Gerusalemme*).

Cleaning the existing English-Italian test dictionary for cognates brings further improvement in P@1 to 0.692, so that the resulting dictionaries become acceptable for downstream tasks. The best value of α, the relative weight to balance the contribution between the cosine similarity and the Weighted Levenshtein Distance, was estimated at 0.73 using a development set which was randomly extracted from the training dictionary. Relying exclusively on the orthographic similarity ($\alpha = 0$) leads to relatively poor results.

Given that the FT+Orth+WLD combination results in consistently better performance, the results of dictionary induction across Slavonic languages are shown only for this setup (Table 3). Comparison of the Slavonic dictionaries to the English-Italian pair shows even more significant improvements through the use of WLD, occasionally from 0.429 to 0.840 for the Slovenian-Croatian pair.

FastText vectors of 300 dimensions built from Wikipedias for selected Balto-Slavonic languages (Belorussian, Czech, Croatian, Lithuanian, Polish, Slovak, Slovene, Ukrainian) have been transformed into a shared Panslavonic embedding space with extraction of the full set of possible cognate forms. For convenience of running cross-lingual experiments, English has also been added to the shared embedding space, even though it is not a related language.

If a reasonable monolingual corpus is available to train the embeddings for another Slavonic language, e.g., Rusyn or Sorbian, as well as a reliable dictionary between this language and one of the languages in the current Panslavonic space (the Wikipedia iWiki lists for such languages are too short to produce useful seed dictionaries), a new language can be easily added to this space.

4. Named Entity Recognition

4.1. Training setup

The cross-lingual shared space has been tested through the Named Entity Recognition (NER) task, which consists in detection and labelling of all occurrences of person names, organisations or locations. This is a convenient downstream task for which there are existing methods and test sets. Recently, various neural network approaches produced very convincing results for NER (Collobert et al., 2011). A particular implementation used in the extrinsic evaluation experiment reported below is based on a sequence tagging method, which combines bidirectional LSTM with CRF for making the final prediction (Lample et al., 2016). Each word is represented by its embedding vector from the shared embedding space, in addition to other universal features, such as character-level embeddings or the presence of capitalisation. The tagger was trained on an existing NER-annotated corpus from (Krek et al., 2012) (in Slovenian) with addition of small samples in Croatian, Czech, Polish, Russian and Ukrainian in order to provide at least some information for the character-level embeddings. The samples were derived from the titles of Wikipedia articles in the respective languages for categories matching such patterns as 'Births' (for person names), 'Organisations' and 'Countries' or 'Villages' (for the lack of a more generic category of locations in Wikipedia).

4.2. BSNLP NER shared task

The NER shared task at BSNLP'17 contained two separate test sets with no training sets for individual languages. One test set was based on the European Commission reports, another one on news wires concerning Donald Trump. The baseline system (Piskorski et al., 2017) was based on large gazetteers developed by the JRC, while the only other submission covering all Slavonic languages (Mayfield et al., 2017) was based on projection of labels via word-aligned parallel corpora, see Table 4.

The shared embedding space is surprisingly efficient. The Slovenian space was used for training, so it provides the upper baseline for adaptation. Czech, Croatian and Polish are sufficiently similar typologically, so the accuracy on those languages is slightly below what has been achieved for Slovenian. Russian and Ukrainian are East Slavonic languages, further away typologically from the rest, which is probably the main reason for the markedly lower accuracy of adaptation of the Slovenian training set. Across all languages, the NER tagger has a problem with detecting relatively long NERs, which are common in the EC test set, such as *The European Convention for the Protection of Human Rights and Fundamental Freedoms*, while the accuracy is higher on the general newswire texts. Overall, the results are considerably lower than what has been achieved for English, which can be explained by much richer morphology of the Slavonic languages, as well as by a smaller training set.

Table 4: NER recognition results

	cs	hr	pl	ru	sl	uk	
EC news:	47.2	**46.2**	**44.8**	46.5	47.8	10.8	JHU
	41.2	30.0	34.6	**53.7**	37.5	**20.8**	JRC
	47.7	44.3	44.2	33.6	**59.5**	13.7	Sharoff
	cs	hr	pl	ru	sl	uk	
Trump:	46.1	50.4	41.0	41.8	46.2	33.2	JHU
	42.2	37.4	48.0	**55.6**	44.2	**50.8**	JRC
	52.6	**52.4**	**55.2**	21.0	**62.6**	20.7	Sharoff

5. Conclusions

The experiments reported in the paper showed that adding the WLD constraints on aligning word forms is a very efficient way for building cross-lingual embedding spaces and for extracting bilingual dictionaries for related languages. For lesser resourced languages and language pairs, the procedure can rely on readily available Wikipedia corpora and the respective iWiki links. Incorporating the resulting multilingual embedding spaces into downstream tasks, such as NER, is also efficient. The results are competitive with the more commonly used projection methods, which are based on parallel corpora (primarily Europarl, which limits the amount of language pairs). The lexicon induction scripts, the shared cross-lingual embeddings as well as the resulting Panslavonic NER taggers are available under permissive licenses.[2]

When the camera-ready copy of the present paper was ready for the final submission, I learned about MUSE (Conneau et al., 2017), a recently developed approach to producing cross-lingual embeddings. It relies on unsupervised alignment between the probability distributions in the two monolingual spaces using adversarial training: the task is to create a translation matrix which can confuse the discriminating function to distinguish two translations in the shared space. The experiments I was able to run on my data before submitting this paper are marked as MUSE in Tables 2 and 3. The MUSE method does not use information about cognates and it offers comparable or better performance in comparison to WLD. Given that MUSE is based on iterative updates instead of a closed form solution, one possible extension concerns integration of the WLD scoring function into MUSE to improve the accuracy across related languages even further.

Another important extension required for the model concerns reliable mapping across the full paradigm of related lexical items in the two languages. A single form in one language can correspond to a number of forms in another language, e.g., *adequate* in English maps to four cognate forms in Italian: *adeguato, adeguata, adeguate* and *adeguati*, corresponding to the choices of singular vs plural and feminine vs masculine, because the English adjectives do not inflect for number and gender. The potential for such one-to-many matches is smaller for closely related languages, since they usually have the same set of morphological categories. However, differences in the suppletivism of forms are common even across related languages, for example, the feminine adjectival forms ending with ой in Russian (e.g., новой, 'new') are used for any non-nominative case, while unique cognate forms are used in Ukrainian for each grammatical case, e.g., genitive: нової, dative: новій, instrumental: новою, etc. A related problem concerns representation of similarities on the level above words. For example, the meaning of the identical forms *postale* in both French and Italian is the same ('post.adj'), they share a number of collocates with the same meaning, e.g., *adresse postale* vs *indirizzo postale*, so they are likely to be well-aligned in the shared embedding space (either with or without WLD constraints). However, the French form is feminine, while the Italian one is masculine, so the correct embedding space should have mapped *postale* in Italian with *postal* in French.

Therefore, the cross-lingual embedding space needs to be built in a way which takes into account the similarity across the full set of forms with respect to their grammatical functions. Using lemmas only can work fine for remote languages, but this loses information about the correspondence of forms in related languages. One of the ways of achieving this is to follow the line of research by (Avraham and Goldberg, 2017). They suggest morphological decomposition of embeddings, which is similar to lexical decomposition of a word into character ngrams in the FastText model. An alternative approach could involve using classifiers to predict the morphological annotations from the embedding vectors in the monolingual space (Belinkov et al., 2017) in order to align embedding vectors by paying attention to the similarity of their morphological annotations.

[2] https://github.com/ssharoff/cognates

References

Artetxe, M., Labaka, G., and Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proc EMNLP*, Austin, Texas, November.

Avraham, O. and Goldberg, Y. (2017). The interplay of semantics and morphology in word embeddings. In *Proc EACL*, pages 422–426, Valencia, Spain, April.

Babych, B., Hartley, A., and Sharoff, S. (2007). Translating from under-resourced languages: comparing direct transfer against pivot translation. In *Proc MT Summit XI*, pages 412–418, Copenhagen.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proc ACL*, Baltimore, Maryland, June.

Belinkov, Y., Durrani, N., Dalvi, F., Sajjad, H., and Glass, J. (2017). What do neural machine translation mod-

els learn about morphology? In *Proc ACL*, Vancouver, Canada, July.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. (2017). Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Dinu, G., Lazaridou, A., and Baroni, M. (2014). Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of IBM Model 2. In *Proc NAACL*, Atlanta, Georgia, June.

Eisele, A. and Chen, Y. (2010). MultiUN: A multilingual corpus from United Nations documents. In *Proc Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta.

Faruqui, M. and Dyer, C. (2014). Improving vector space word representations using multilingual correlation. In *Proc EACL*, pages 462–471, Gothenburg, Sweden, April.

Fung, P. (1995). Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus. In *Proc. Third Annual Workshop on Very Large Corpora*, pages 173–183, Boston, Massachusetts.

Klementiev, A., Titov, I., and Bhattarai, B. (2012). Inducing crosslingual distributed representations of words. In *Proc COLING*, Mumbai, India, December.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proc. MT Summit X*.

Krek, S., Erjavec, T., Dobrovoljc, K., Holz, N., Ledinek, N., and Može, S., (2012). *Učni korpus ssj500k kot podatkovna zbirka*.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proc NAACL*, pages 260–270, San Diego, California.

Mayfield, J., McNamee, P., and Costello, C. (2017). Language-independent named entity analysis using parallel projection and rule-based disambiguation. In *Proc BSNLP*, pages 92–96, Valencia, Spain, April.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2017). Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*.

Mikolov, T. (2012). *Statistical language models based on neural networks*. Ph.D. thesis, Brno University of Technology.

Piskorski, J., Pivovarova, L., Šnajder, J., Steinberger, J., and Yangarber, R. (2017). The first cross-lingual challenge on recognition, normalization, and matching of named entities in slavic languages. In *Proc BSNLP*, pages 76–85, Valencia, Spain, April.

Radovanović, M., Nanopoulos, A., and Ivanović, M. (2010). Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(Sep):2487–2531.

Rapp, R. (1995). Identifying word translations in non-parallel texts. In *Proc. of the 33rd ACL*, pages 320–322, Cambridge, MA.

Rios, M. and Sharoff, S. (2015). Obtaining SMT dictionaries for related languages. In *Proc the Eighth Workshop on Building and Using Comparable Corpora*, pages 68–73, Beijing, China, July.

Sharoff, S., Rapp, R., and Zweigenbaum, P. (2013). Overviewing important aspects of the last twenty years of research in comparable corpora. In Serge Sharoff, et al., editors, *BUCC: Building and Using Comparable Corpora*, pages 1–17. Springer.

Tsvetkov, Y. and Dyer, C. (2016). Cross-lingual bridges with models of lexical borrowing. *JAIR*, 55:63–93.

Tools for Building an Interlinked Synonym Lexicon Network

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, Jan Hajič
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Malostranské nám. 25
11800 Prague 1, Czech Republic
{uresova,fucikova,hajicova,hajic}@ufal.mff.cuni.cz

Abstract

This paper presents the structure, features and design of a new interlinked verbal synonym lexical resource called CzEngClass and the editor tool being developed to assist the work. This lexicon captures cross-lingual (Czech and English) synonyms, using valency behavior of synonymous verbs in relation to semantic roles as one of the criteria for defining such interlingual synonymy. The tool, called Synonym Class Editor - SynEd, is a user-friendly tool specifically customized to build and edit individual entries in the lexicon. It helps to keep the cross-lingual synonym classes consistent and linked to internal as well as to well-known external lexical resources. The structure of SynEd also allows to keep and edit the appropriate syntactic and semantic information for each Synonym Class member. The editor makes it possible to display examples of class members' usage in translational context in a parallel corpus. SynEd is platform independent and may be used for multiple languages. SynEd, CzEngClass and services based on them will be openly available.

Keywords: Lexical Resource, Parallel Corpus, Semantics, Syntax, Synonymy, Valency

1. Introduction

We present a demonstration of our lexicon editor, called **SynEd**, for creating an interlinked multilingual (for the time being, bilingual) lexical resource—a contextually-based synonym lexicon of verbs based on their syntactic and semantic behavior in (bilingual) context. We also present the design and structure (scheme) of the lexicon dataset proper.

The lexicon, under the working name "CzEngClass" (Urešová et al., 2018), is being built "bottom-up", using a parallel bilingual corpus with a rich manual annotation (the Prague Czech-English Dependency Treebank, version 2.0), its associated valency lexicons, i.e., EngVallex, PDT-Vallex and CzEngVallex, and it is being linked to other external lexical resources–VALLEX, FrameNet, VerbNet, PropBank and Czech and English WordNets (Sect. 3.).

One of the crucial parts of the "CzEngClass" project was to establish a scheme for this resource (Sect. 4.), taking into account not only its intended multilinguality, but also links to the initial bilingual parallel corpus (and more corpora in the future) and the existing relevant lexical resources. The semantic role labels used for defining the individual synonym classes are based mostly on FrameNet (and in some cases on VerbNet), while the argument labels and their morphosyntactic behavior are taken from the valency lexicons used for the PCEDT annotation. One of the main principles for the design of "CzEngClass" follows A. Kilgariff's idea of corpus - dictionary linkage (cf. the PDIC and PCID model described in (Kilgariff, 2005)), so we are strictly keeping references to all of the used lexical resources (internal and external) as well as to the corpus examples influencing the class divisions.

Since the goal of the project is to identify contextually-based Czech and English synonyms, each verb has to be first broken down to senses. The initial set of senses for both Czech and English has been taken from the Czech and English valency lexicons, since they have been determined during the creation of these lexicons and linked manually to each occurrence of the verb in the parallel corpus.

SynEd, the editor for "CzEngClass" (Sect. 4.), is currently used as a standalone application, with links to all the referenced external resources that can be immediately accessed directly in **SynEd**, through third-party applications and/or through web browsers. Currently, there are 60 synonym classes with approx. 470 English and over 1000 Czech verbs (verb senses) included in CzEngClass,[1] A selection and/or creation of a web-based customized interface for browsing and searching will follow and is part of future work (Sec. 5.). The editor as well as the associated data will be publicly available under a CC license.

2. Related Work

When designing the "CzEngClass" lexicon as well as the editor, we were looking for an existing annotation tool for a similar type of lexicon(s) we could possibly adapt. We concentrated on those that allow working with corpora, since that is also the way we approach building the lexicon.

Lexicons have been built using software tools (and corpora) since the 1980s, mainly at publishers, such as (Ahlswede, 1985); such efforts are summarized in (Teubert, 2007). We have considered many other existing tools, either standalone or available as web services and applications. *Lexicon Creator* is a tool designed to help developers produce lexical data for its use in a variety of linguistic applications. According to (Fontenelle et al., 2008), *Lexicon Creator* enables to work on existing wordlists derived either directly from corpora or from previously created wordlist data. *Lexicon Builder*, available as web service (Parai et al., 2010), aims at automated methods to compile custom lexicons from BioPortal ontologies. *CoBaLT* (Kenter et al., 2012), is a web-based editor optimized for work with large

[1] The final size should cover at least the current Czech and English valency lexicons, i.e. about 15,000 verbs (verb senses); it is very hard to estimate the number of classes created in the end.

datasets and to produce historical lexica. *Dicet* (Gader et al., 2012) is a knowledge-based, tailor-made lexical graph editor and browser that allows lexicographers to browse through the lexical network and directly expand and revise it. *DECFC*,[2] a specialized dictionary editor (Decary and Lapalme, 1990), provides a multi-windowing environment that enables the simultaneous execution of different processes on different parts of the screen. A database schema for developing and maintaining Japanese linguistic resources (Asahara et al., 2002) is a stand-off framework combining XML and a relational database. SIL's [3] latest version (8.3) of *FLEx* (FieldWorks Language Explorer) is a next specific program designed to assist linguists in collecting, managing and publishing linguistic data.[4] *FLEx* features powerful bulk editing tools and a large number of built-in fields.

For the resources we are linking CzEngClass to, there are also several tools. A specific editor - *Cornerstone* (Choi et al., 2010a) - has been specifically customized to create and edit frameset files for PropBank project.[5] One of the biggest advantages of Cornerstone is that it accommodates several languages (it was used for e.g., Arabic, Chinese, English, Hindi, and Korean). A semi-automatic *VerbaLex–FrameNet* linking tool (Materna, 2009; Materna and Pala, 2010; Materna, 2011; Materna, 2014) has been developed. This tool aims to build a core of Czech FrameNet.

All the above mentioned editors and tools are very sophisticated and useful, but rather specialized for the particular lexical resource. Those few exceptions, such as the Japanese lexical resource builder, are on the other hand too general and would need a substantial amount of customization, since for the CzEngClass lexicon we need to express more specific requirements. We have thus decided to write a new editor, reusing some parts that have been developed in the past for editing the valency lexicons and linking them to the associated treebanks. This new editor is called **SynEd**, and we describe it in (Sect. 4.).

3. Resources Used

Our approach to the development of a synonym lexicon for both NLP and linguistic studies builds on corpus examples with natural contexts. Therefore, we build our research on electronically accessible and richly annotated data, namely, on the Prague Czech-English parallel treebank and on the Prague Dependency Treebank (Hajič et al., 2006) valency lexicons (PDT-Vallex, EngVallex and CzEngVallex), as well as on other well-established lexical databases (e.g., FrameNet, VerbNet, Semlink, PropBank and Czech and English WordNets).

The core corpus resource is the Prague Czech-English Dependency Treebank (Hajič et al., 2012), which stores parallel PDT-style annotations (manual annotation of morphology, syntax and semantics) of English texts (Wall Street Journal part of Penn Treebank (Marcus et al., 1993)) and their professional translation into Czech[6]. The PCEDT annotations capture linkage of the surface and deep syntactic layers; moreover, the deep layer contains verbal word sense labeling by keeping links of each verb occurrence to the appropriate valency frame in the associated valency lexicons, PDT-Vallex and EngVallex.

The main lexicon resources we build on are thus the following valency lexicons: EngVallex (Cinková et al., 2014),(Cinková, 2006), PDT-Vallex (Urešová et al., 2014), (Urešová, 2011) and also CzEngVallex (Urešová et al., 2015), (Urešová et al., 2016). These lexicons are based on the Functional Generative Description Valency Theory (FGDVT)[7]. In the lexicons, each entry has a headword with one or more valency frame(s). Every valency frame contains labeled arguments, their obligatoriness and the required surface form of valency frame members (arguments). PDT-Vallex contains about 12,000 valency frames for about 7,000 verbs; EngVallex, contains about 7,000 valency frames for almost 4,500 verbs. CzEngVallex links them across Czech and English using the automatic PCEDT corpus alignments (after manual pruning of erroneous alignments has been applied).[8]

Since CzEngClass aims at synonymy based on semantics, we also use the following lexical resources: FrameNet (Fillmore et al., 2003; Fillmore et al., 2003), VerbNet (Schuler, 2006), Semlink (Palmer, 2009; Bonial et al., 2012), PropBank (Palmer et al., 2005), Czech WordNet (Pala et al., 2011), (Pala and Smrž, 2004) and English WordNet (Miller, 1995; Fellbaum, 1998).[9] These resources are mainly being referred to by the newly built CzEngClass entries; in addition, FrameNet and VerbNet semantic roles are being consulted when defining any particular synonym class.

4. Lexicon Design

The lexicon groups translational verbal equivalents, i.e., verb senses, together both in Czech and English, originally represented as valency frames in the Czech and English valency lexicons, into "Synonym Classes." We call the synonymous senses in one class "Class Members."

Each class is assigned a common set of "Semantic Roles" (SRs) and the valency frame elements of each Class Member are mapped to SRs assigned to this set. This mapping is the crucial defining feature of the CzEngClass lexicon, since it defines, through the original argument mapping to their morphosyntactic features, the use of the arguments in text, and therefore the context in which the particular verb can be considered a synonym to the other, similarly defined (or "restricted") verbs (their senses).

Class Members are then further individually linked to the original lexicon (PDT-Vallex or EngVallex) and also to entry(ies) in all the additional resources used (if such rele-

[2]DECFC (Explanatory and Combinatory Dictionary of Contemporary French).

[3]Originally known as the Summer Institute of Linguistics, Inc.

[4]https://software.sil.org/fieldworks

[5]For proper PropBank annotation another editor called *Jubilee* (Choi et al., 2010b) has been used.

[6]For further information see the project web page: http://ufal.mff.cuni.cz/pcedt2.0.

[7]For details on the FGDVT theory see e.g., (Panevová, 1977; Lopatková, 2010)

[8]The CzEngClass lexicon refers also to VALLEX (Lopatková et al., 2016), a much more elaborated lexicon, built on the same theoretical framework as PDT-Vallex. However, VALLEX it is not based on the PDT data.

[9]https://wordnet.princeton.edu/

vant entries exists in them). More than one link to any of the additional resources can be included; in such a case, it means that the same or similar meaning has been found in the external lexicons, with an unclear distribution of senses. In other words, mappings between CzEngClass entries and the entries in the other lexical resources are not necessarily 1:1.

Table 1 shows a (simplified)[10] example. Both Czech and English verbs, determined to be synonymous in the particular sense defined by the PDT-Vallex and EngVallex links, are listed as this synonym class members. The semantic roles chosen for this class have been simplified from the FrameNet. Once selected, the arguments of the particular verb sense, as found in the valency lexicons, *must* be mapped to these semantic roles; however, if some of the SRs are not listed as an argument, the corresponding adjunct might be used, too. In the examples in Table 1, another phenomenon is displayed: in some cases, a semantic role can be expressed as one argument (e.g., PAT for *hear*, as in ... *they heard about it.PAT in local news*), or as two (part of the Phenomenon is expressed as PAT, part as EFF for *know*, as in - ... *he didn't know anything.EFF about him.PAT*). In such a case, all possibilities must be listed in the particular mapping field.

For such a lexicon, a stand-off XML schema has been developed and is used as the storage format. The XML document contains a header part, which gives local or remote reference to the external resources, lists all possible SRs (semantic roles) that are used in the lexicon (for consistency checking), etc. The body of the document contains the classes. Each class first lists the assigned SRs, and then the class members by verb lemmas, references to valency frames in PDT-Vallex and EngVallex defining the sense IDs, and references to external resources (FrameNet, VerbNet, PropBank, WordNets) for each class member. In addition, bookkeeping information is stored as well, such as annotator's ID, timestamps, etc. A (simplified) extract of the XML-formatted lexicon follows, for the class *build-budovat*:

```
<CzEngClass owner="EF">
... (header with SRs, arg. labels, lexicon URLs etc.)
<body>
<veclass lemma="budovat (v-w228f1)" id="vec00001">
  <commonroles>
    <role idref="vecroleAgent" spec=""/>
    <role idref="vecroleComponents" spec=""/>
    <role idref="vecroleCreated_Entity" spec=""/>
    <role idref="vecroleAssets_currency" spec=""/>
  </commonroles>
  <classnote/>
  <classmembers>
    <classmember status="not_touched" lang="cs"
      lexidref="pdtvallex"
      idref="PDT-Vallex-ID-v-w228f1"
      lemma="budovat" id="vec00001cm00001">
      <restrict/>
      <maparg>
        <argpair>
          <argfrom idref="vecargpdtACT"/>
          <argto idref="vecroleAgent"/>
        </argpair>
        <argpair>
          <argfrom idref="vecargpdtORIG"/>
          <argto idref="vecroleComponents"/>
        </argpair>

        ... (other arg pair mappings to SRs)

      </maparg>
      <cmnote/>
      <extlex idref="pdtvallex">
        <links>
          <link idref="v-w228f1" lemma="budovat"/>
        </links>
      </extlex>
      <extlex idref="czengvallex">
        <links>
          <link idref="vw287f1p1" enid="ev-w384f1"
            enlemma="build" csid="v-w228f1" .../>
          <link idref="vw713f3p1" enid="ev-w918f3"
            enlemma="develop" csid="v-w228f1" .../>

          ... (more links to aligned valency frames)

          <link idref="vw2305f1p1" enid="ev-w2903f1"
            enlemma="set_up" csid="v-w228f1" .../>
        </links>
      </extlex>
      <examples> </examples>
      <local_history>
       <local_event time_stamp="10.7.2017 22:38:45"
         type_of_event="edit ..." author="ZU"/>
      </local_history>
    </classmember>

    ... (more classmembers/synonyms)

  </classmembers>
</veclass>

... (more classes of synonyms)

</body> </CzEngClass>
```

[10]For space reasons, specific morphosyntactic restrictions and additional external links to PropBank and WordNets are not shown; the list of class members is also substantially shortened.

The **SynEd** editor (Fig. 1) that allows to view and edit the CzEngClass lexicon as described in the previous section, including management of external links to other lexicons and corpora (examples), has been created by a substantial rewrite of the original PDT-Vallex editor used for the annotation of the PDT (Hajič et al., 2006; Hajič et al., 2018). In order to allow fully offline but parallel work, it is a standalone perl program that runs on multiple OS platforms, supported by data synchronization over `svn` (and/or `git`). The editor allows editing of the XML document described above, and it understands all the external files (lexicons and corpora, as described in Sect. 3.). The annotator can thus see the original alignments across languages (from CzEngVallex and the PCEDT corpus), as well as the FrameNet and VerbNet classes taken as a basis for a particular synonym set. It also shows, for each synonym class member, only the relevant examples from the PCEDT corpus (that correspond to verb(s) in the other language, again based on the PCEDT Czech-English alignments). Among these example sentences, even though there are usually not that

Class	Roles			External Links		
member	Cognizer	Phenomenon	Speaker	*PDT/EngVallex*	*FrameNet*	*VerbNet*
dozvědět se	ACT	PAT	ORIG	v-f1		
hear	ACT	PAT	ORIG	ev-f1	Hear	discover-84-1-1, see-30.1-1-1-1
know	ACT	PAT+EFF	ORIG	ev-f1	Awareness	consider-29.9-1-1
learn	ACT	PAT	ORIG	ev-f1	Becoming_aware	learn-14-2-1
doslýchat se	ACT	PAT+EFF	ORIG	ev-f1		
poučit se	ACT	PAT+EFF	ORIG	v-f1		
...	...	...	...	...	...	

Table 1: Mappings and External Links for class "dozvědět se, hear, ..., poučit se, ..."

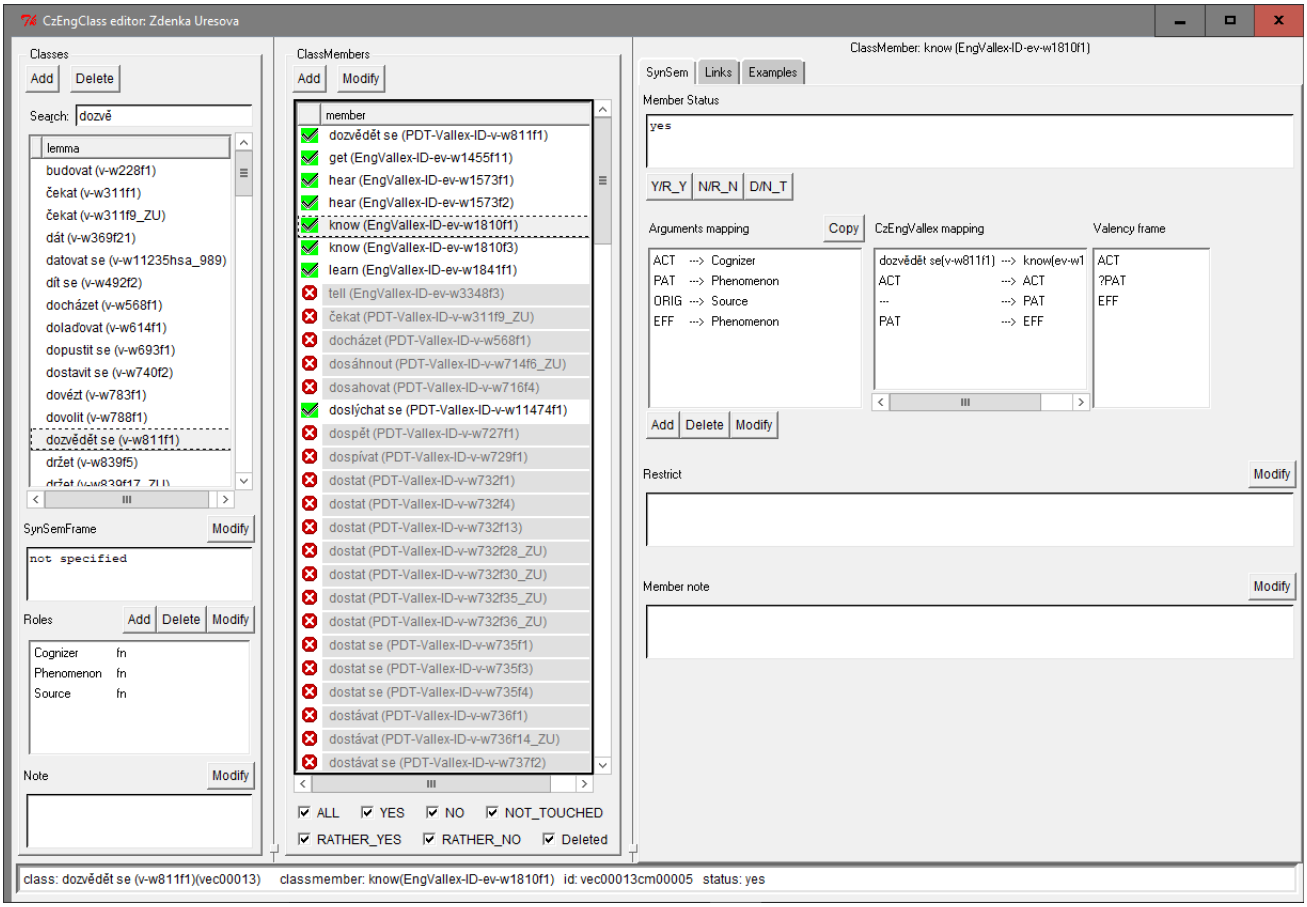


Figure 1: A screenshot of the **SynEd** editor

many in the relatively small PCEDT corpus, the annotator can mark the "best" (most illustrative) example sentence(s) from the corpus to be shown first to the future user once the lexicon is finished and made public.

As seen in Fig. 1, the editor's screen contains three sections: list of all lemmas (allowing search for their class) and basic information for the selected class on the left, then a list of all class members (with PDT-Vallex/EngVallex IDs) in the middle, and then on the right information pertaining to the selected class member: argument mapping to SRs, status, additional restrictions and notes. The right-hand pane can be also switched to see and edit the Links to external resources (Fig. 2), or to see lists of Examples (using the tabs above the pane).

All external links are "clickable" in order to both simplify and speed up annotator's work. The lexical resource links (tab "Links", see also Fig. 2) are expanded to a full external or local URL and opened in a new browser tab. For example, the ID link for "draft," which is part of the `create-26.4-1` VerbNet class, is expanded to `http://verbs.colorado.edu/verb-index/vn3.3/vn/create-26.4.php#create-26.4-1` and shown in the user's new browser tab. Similarly, in the "Examples" tab, user can click on a "Show" button and the examples from PCEDT 2.0 are shown in the `tred`[11] viewer/editor, which must be installed locally.[12]

[11] `https://ufal.mff.cuni.cz/tred`

[12] In the future, links to examples will possibly be converted to PML-TQ queries directed to `https://lindat.mff.cuni.cz/services/pmltq/#!/treebank/pcedt20`.

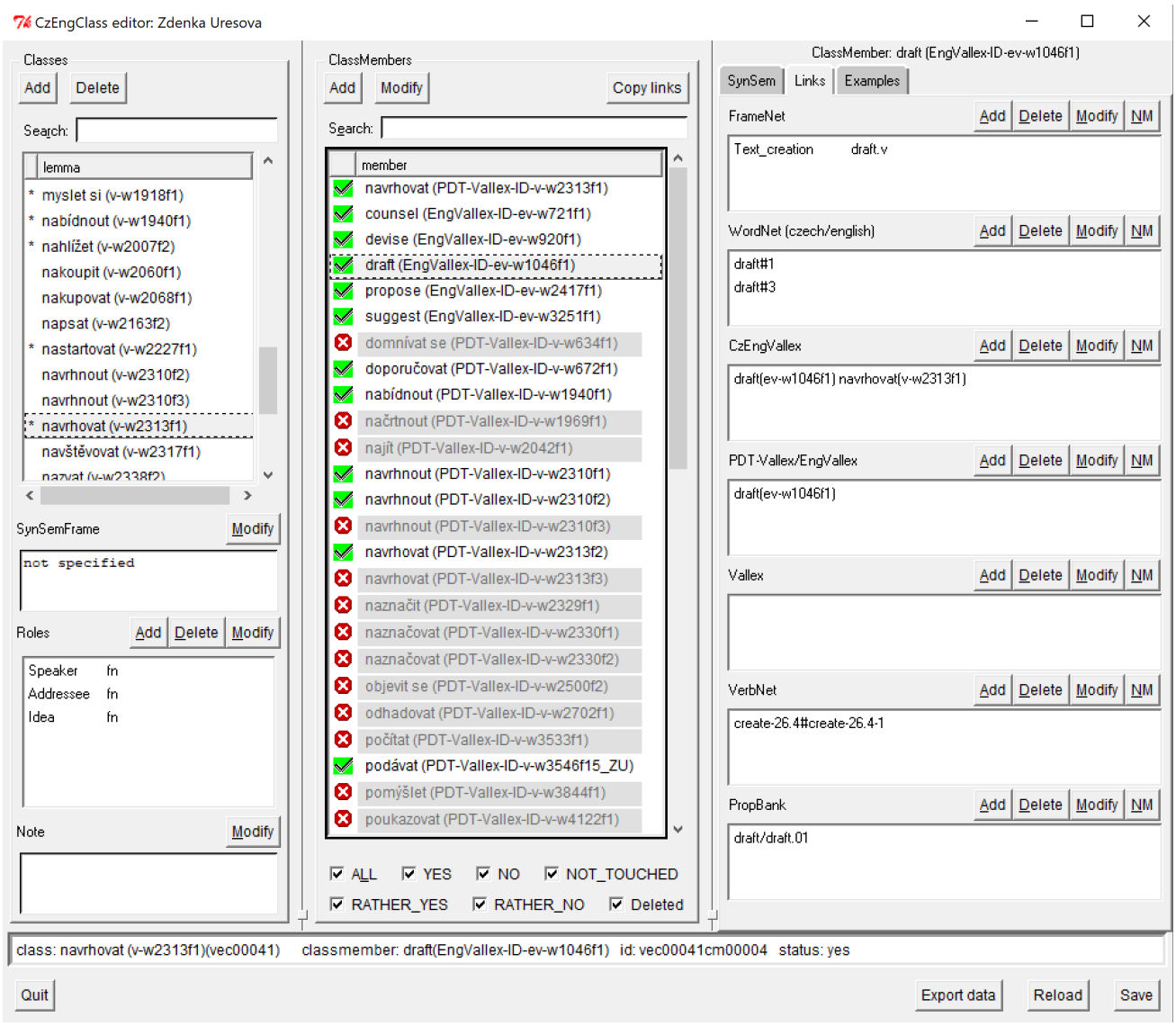


Figure 2: A screenshot of the Links to external resources tab in the **SynEd** editor

5. Future Work

This paper presented the design of the structure of the CzEngClass lexicon and the editor **SynEd** used for its editing and maintenance. The editor is currently being extensively tested in order to be used (simultaneously) by multiple annotators. Two annotators had worked with **SynEd** on the CzEngClass lexicon for almost a year so the editor has already been sufficiently tested and gradually improved, including time-saving as well as ergonomical features and is now ready to begin routine work.

We plan to investigate the relation of valency and semantic roles in more detail, also from their morphosyntactic realization point of view, and adjust our editor to such newly emerging requirements. We will continue to develop **SynEd** by improving its functionality and practicality through more feedback from future annotators.

In the future, we plan to explore cooperation with the OntoLex Community Group (Piasecki et al., 2017) in order to integrate our work into methods for representing linked lexical data.

We also plan to join the Cross-lingual FrameNet Group. We believe that our editor will be–possibly after some refinements–suitable for editing resources linked to FrameNet (and other lexical resources), including its deployment for other languages.

Acknowledgments

This work has been supported by the grant No. GA17-07313S of the Grant Agency of the Czech Republic, and it uses resources hosted by the LINDAT/CLARIN Research Infrastructure, projects No. LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781, supported by the Ministry of Education of the Czech Republic. The last co-author has also been supported in part by the project VIA-DAT, No. DG16P02R019, of the Ministry of Culture of the Czech Republic.

6. Bibliographical References

Ahlswede, T. E. (1985). A Tool Kit for Lexicon Building. In *23rd Annual Meeting of the Association for Computational Linguistics*.

Asahara, M., Yoneda, R., Yamashita, A., Den, Y., and Matsumoto, Y. (2002). Use of XML and Relational Databases for Consistent Development and Maintenance of Lexicons and Annotated Corpora. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*. European Language Resources Association (ELRA).

Bonial, C., Feely, W., Hwang, J. D., and Palmer, M. (2012). Empirically Validating VerbNet Using SemLink. In *Seventh Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, Istanbul, Turkey, May.

Choi, J. D., Bonial, C., and Palmer, M. (2010a). Propbank Frameset Annotation Guidelines Using a Dedicated Editor, Cornerstone. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Languages Resources Association (ELRA).

Choi, J. D., Bonial, C., and Palmer, M. (2010b). Propbank Instance Annotation Guidelines Using a Dedicated Editor, Jubilee. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Languages Resources Association (ELRA).

Cinková, S. (2006). From PropBank to EngValLex: adapting the PropBank-Lexicon to the valency theory of the functional generative description. In *Proceedings of LREC 2006, Genova, Italy*.

Decary, M. and Lapalme, G. (1990). An Editor for the Explanatory and Combinatory Dictionary of Contemporary French (DECFC). *Computational Linguistics*, 16(5):145–154.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Fillmore, C. J., Johnson, C. R., and L.Petruck, M. R. (2003). Background to FrameNet: FrameNet and Frame Semantics. *International Journal of Lexicography*, 16(3):235–250.

Fontenelle, T., Cipollone, N., Daniels, M., and Johnson, I. (2008). Lexicon Creator: A Tool for Building Lexicons for Proofing Tools and Search Technologies. In Janet DeCesaris Elisenda Bernal, editor, *Proceedings of the 13th EURALEX International Congress*, pages 359–369, Barcelona, Spain, jul. Institut Universitari de Linguistica Aplicada, Universitat Pompeu Fabra.

Gader, N., Lux-Pogodalla, V., and Polguère, A. (2012). Hand-Crafting a Lexical Network With a Knowledge-Based Graph Editor.

Kenter, T., Erjavec, T., Žorga Dulmin, M., and Fiser, D. (2012). Lexicon Construction and Corpus Annotation of Historical Language with the CoBaLT Editor.

Kilgariff, A. (2005). Putting the corpus into the dictionary. In *Proceedings of the second Meaning workshop*, pages 1–6, Trento, Italy, February.

Lopatková, M., Kettnerová, V., Bejček, E., Vernerová, A., and Žabokrtský, Z. (2016). *Valenční slovník českých sloves VALLEX*. Nakladatelství Karolinum, Praha, Czechia.

Lopatková, M. (2010). *Valency Lexicon of Czech Verbs: Towards Formal Description of Valency and Its Modeling in an Electronic Language Resource*. Habilitation thesis. Charles University in Prague, Praha, Czechia.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330.

Materna, J. and Pala, K. (2010). Using Ontologies for Semi-automatic Linking VerbaLex with FrameNet. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Materna, J. (2009). Linking Czech Verb Valency Lexicon VerbaLex with FrameNet. In *Proceedings of 4th Language and Technology Conference*, pages 215–219, Poznań. Wydawnictwo Poznańskie Sp. z o.o.

Materna, J. (2011). *Building FrameNet in Czech [online]*. Rigorózní práce, Masarykova univerzita, Fakulta informatiky, Brno.

Materna, J. (2014). *Probabilistic Semantic Frames [online]*. Disertační práce, Masarykova univerzita, Fakulta informatiky, Brno.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41, November.

Pala, K. and Smrž, P. (2004). Building Czech Wordnet. *Romanian Journal of Information Science and Technology*, 7(1-2):79–88.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, March.

Palmer, M. (2009). Semlink: Linking PropBank, VerbNet and FrameNet. In *Proceedings of the Generative Lexicon Conference*, page 9–15.

Panevová, J. (1977). Verbal Frames Revisited. *The Prague Bulletin of Mathematical Linguistics*, (28):55–72.

Parai, G. K., Jonquet, C., Xu, R., Musen, M. A., and Shah, N. H. (2010). The Lexicon Builder Web service: Building Custom Lexicons from two hundred Biomedical Ontologies. In *American Medical Informatics Association Annual Symposium, AMIA'10*.

Piasecki, M., Walkowiak, T., Rudnicka, E., Naskret, T., and Bond, F. (2017). The Concept of Lexical Platform. In *Proceedings of the LDK 2017 Workshops: 1st Workshop on the OntoLex Model (OntoLex-2017), Shared Task on Translation Inference Across Dictionaries & Challenges for Wordnets co-located with 1st Conference on Language, Data and Knowledge (LDK 2017), Galway, Ireland, June 18, 2017.*, pages 197–208.

Schuler, K. K. (2006). *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania.

Teubert, W. (2007). *Text Corpora and Multilingual Lexicography*. John Benjamins Publishing Company.

Urešová, Z., Fučíková, E., and Šindlerová, J. (2016). CzEngVallex: a Bilingual Czech-English Valency Lexicon. *The Prague Bulletin of Mathematical Linguistics*, 105:17–50.

Urešová, Z., Fučíková, E., Hajičová, E., and Hajič, J. (2018). Creating a Verb Synonym Lexicon Based on a Parallel Corpus. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC'18)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Urešová, Z. (2011). *Valenční slovník Pražského závislostního korpusu (PDT-Vallex)*. Studies in Computational and Theoretical Linguistics. Ústav formální a aplikované lingvistiky, Praha, Czechia.

7. Language Resource References

Cinková, S., Fučíková, E., Šindlerová, J., and Hajič, J. (2014). *EngVallex - English Valency Lexicon*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0023-4337-2.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Cinková, S., Fučíková, E., Mikulová, M., Pajas, P., Popelka, J., Semecký, J., Šindlerová, J., Štěpánek, J., Toman, J., Urešová, Z., and Žabokrtský, Z. (2012). *Prague Czech-English Dependency Treebank 2.0, https://catalog.ldc.upenn.edu/LDC2004T25*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4.

Hajič, J., Bejček, E., Bémová, A., Buráňová, E., Hajičová, E., Havelka, J., Homola, P., Kárník, J., Kettnerová, V., Klyueva, N., Kolářová, V., Kučová, L., Lopatková, M., Mikulová, M., Mírovský, J., Nedoluzhko, A., Pajas, P., Panevová, J., Poláková, L., Rysová, M., Sgall, P., Spoustová, J., Straňák, P., Synková, P., Ševčíková, M., Štěpánek, J., Urešová, Z., Vidová Hladká, B., Zeman, D., Zikánová, Š., and Žabokrtský, Z. (2018). *Prague Dependency Treebank 3.5*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-2621.

Hajič, J., Panevová, J., Hajičová, E., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M., Žabokrtský, Z., Ševčíková, M., and Urešová, Z. (2006). *Prague Dependency Treebank 2.0*. LDC, number LDC2006T01, LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4.

Pala, K., Čapek, T., Zajíčková, B., Bartůšková, D., Kulková, K., Hoffmannová, P., Bejček, E., Straňák, P., and Hajič, J. (2011). *Czech WordNet 1.9 PDT*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0001-4880-3.

Urešová, Z., Štěpánek, J., Hajič, J., Panevova, J., and Mikulová, M. (2014). *PDT-Vallex: Czech Valency lexicon linked to treebanks*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0023-4338-F.

Urešová, Z., Fučíková, E., Hajič, J., and Šindlerová, J. (2015). *CzEngVallex - Czech English Valency Lexicon*. Charles University in Prague, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-1512.

Very Large-Scale Lexical Resources to Enhance Chinese and Japanese Machine Translation

Jack Halpern
The CJK Dictionary Institute, Inc.
34-14, 2-chome, Tohoku, Niiza-shi, Saitama 352-0001 JAPAN
jack@cjki.org

Abstract

A major issue in machine translation (MT) applications is the recognition and translation of named entities. This is especially true for Chinese and Japanese, whose scripts present linguistic and algorithmic challenges not found in other languages. This paper discusses some of the major issues in Japanese and Chinese MT, such as the difficulties of translating proper nouns and technical terms, and the complexities of orthographic variation in Japanese. Of special interest are neural machine translation (NMT) systems, which suffer from a serious out-of-vocabulary problem. However, the current architecture of these systems makes it technically challenging for them to alleviate this problem by supporting lexicons. This paper introduces some Very Large-Scale Lexical Resources (VLSLR) consisting of millions of named entities, and argues that the quality of MT in general, and NMT systems in particular, can be significantly enhanced through the integration of lexicons.

Keywords: machine translation, lexicon, Chinese, Japanese

1. Introduction

A major issue in MT applications is the translation of named entities and technical terms. This is especially true for Chinese and Japanese, whose scripts present linguistic and algorithmic challenges not found in other languages. Some factors that contribute to these difficulties include:

1. The Japanese orthography is highly irregular, requiring advanced capabilities such as cross-script normalization (Halpern, 2008).
2. The morphological complexity of Japanese requires the use of a robust morphological analyzer for segmentation and lemmatization (Brill et al., 2001; Yu et al., 2000).
3. The accurate conversion between Simplified Chinese (SC) and Traditional Chinese (TC) (Halpern and Kerman, 1999).
4. The difficulty of accurately translating POIs (points of interest). These are extremely numerous, difficult to detect, and have an unstable orthography.
5. The large number of technical terms.
6. The lack of comprehensive lexical resources.

This paper discusses some of these issues, and introduces Very Large-Scale Lexical Resources (VLSLR) consisting of millions of named entities. It argues that lexicons can enhance the translation accuracy of NMT systems, which currently don't use lexicons.

2. Japanese Orthographic Variants

The Japanese orthography is highly irregular. The numerous orthographic variants, which are common and unpredictable, negatively impact recall and pose a major challenge to MT. The variation results from the unpredictable interaction between four scripts: kanji, hiragana, katakana, and Latin. For example, in 金の卵を産む鶏 'A hen that lays golden eggs,' *tamago* 'egg' has four variants (卵,玉子,たまご,タマゴ), *niwatori* 'chicken' has three (鶏,にわとり,ニワトリ), and *umu* 'give birth to' has two (産む, 生む), which expands to 24 permutations. Algorithmic solutions have no hope of identifying these as instances of the same underlying sentence without support for orthographic disambiguation/normalization.

The most important types of orthographic variation in Japanese (Halpern, 2008) are: (1) *Okurigana*, which are kana endings such as 当たり外れ and 当り外れ for *atarihazure*. Normalizing *okurigana* variants, which are numerous and unpredictable, is a major issue. An effective solution is to use an orthographic variants lexicon. (2) *Cross-script variants* refers to variation across the four Japanese scripts, 'carrot' (*ninjin*) written in kanji (人参), hiragana (にんじん), and katakana (ニンジン). (3) *Katakana loanword variants*, a major annoyance since they are numerous and irregular. The same word to be written in multiple, unpredictable ways, such as コンピュータ and コンピューターa for 'computer' and チーム and ティーム for 'team'.

3. Lexicons in MT

3.1 Lexicons in traditional MT

Lexicons, including dictionary databases and terminology glossaries, have played a critical role in MT systems, dramatically improving translation quality, especially in view of the fact that these systems perform rather poorly on out-of-domain texts (Mediani et al., 2014). Attempts to replace lexicons with algorithmic solutions for certain tasks, such as processing Japanese orthographic variants and katakana loanwords, have been made (Brill et al., 2001). To successfully process the highly irregular Japanese orthography of Japanese orthographic disambiguation cannot be based on probabilistic methods alone. Attempts have been made along these lines, as for example in Brill et al. (2001), with some claiming performance equivalent to lexicon-based methods, while Kwok (1997) reports good results with only a small lexicon and simple segmentor.

In fact, such algorithmic/statistical methods have only met with limited success. The fundamental problem is that such methods, even when based on large-scale corpora, often fail to achieve high accuracy MT unless they are supported by large-scale lexicons. For example, Emerson (2000) and Nakagawa (2004) and others have shown that MT systems and robust morphological analysers capable of processing lexemes, rather than bigrams or n-grams, must be supported by a large-scale computational lexicons (even 100,000 entries is much too small).

3.2 Quantum leap

The application of artificial neural network to MT gave birth to a new paradigm, Neural Machine Translation (NMT), that represents a quantum leap in MT technology. In a short period of time, such major MT engines as Google, Bing and Baidu adopted the NMT model, whose success can be attributed to its capability to implement the translation process on the basis of a single, end-to-end probabilistic model (Luong et al., 2015). Even as NMT development proceeds at breakneck speed, research on newer advanced technologies based on Quantum Neural Networks (QNN) is already in progress (Moire et al., 2016). However, despite of the significant improvement in translation quality, the ability of NMT systems to correctly translate named entities and some technical terms has in fact somewhat deteriorated.

3.3 Lexicons in NMT

According to He et al. (2016) of Baidu, "an NMT system usually has to apply a vocabulary of certain size... thus it causes a serious out-of-vocabulary problem." Baidu is probably the only company that has tackled the difficult problem of integrating lexicons into MT systems.

On April 25-26, 2017 the TAUS Executive Forum Tokyo 2017 (TAUS, 2017) was held in Tokyo, and on September 18-22, 2017 MT Summit XVI was held in Nagoya, Japan, where the team leaders and representatives of several major NMT developers (Google, Microsoft, NICT) gathered. In discussions with several NMT experts, including Chris Wendt from Microsoft and representatives of Baidu, it became clear that though currently the major NMT systems (except for Baidu) do not use lexicons, there is no technical reason that lexicons cannot be used. The basic idea is to regard a lexicon as a kind of sentence-aligned, bilingual parallel corpus, and to have the system assign a higher probability to the lexicon entries so as to override the results of the normal NMT algorithms. For example, 三角線 *Misumi-sen*, the name of a railway line, is called 'Misumi Line', so that it is safe to allow the lexicon results to override the NMT results such as 'Triangle' (Google) and 'Triangular line' (Bing).

Some potential obstacles are (1) that lexicons, unlike corpora, do not provide context, and (2) that ordinary lexicons do not provide translation probabilities. However this is not critical for named entities, especially POIs, and even for many technical terms, since named entities are mostly monosemic, which means that word sense disambiguation is unnecessary and that the lexicon can automatically be assigned a higher probability. For example, there is no danger that 三角線 should be correctly translated literally as 'triangular line'. rather than 'Misumi Line', the official name of this train line.

3.4 Lexicon integration

NMT has transformed MT technology by achieving significant quality improvements over traditional MT systems. When NMT systems are trained on large-scale domain-specific parallel corpora, they do achieve remarkable results *within* those domains. According to Arthur et al. (2016), NMT does not perform well when "translating low-frequency content words that are essential to understanding the meaning of the sentence." Our experiments (see §4 below) have confirmed that NMT systems also perform poorly when translating named entities, especially POIs, as well as when processing Japanese orthographic variants. Arthur et al. (2016) propose that this can be overcome by integrating "discrete translation lexicons" into NMT systems, and asserts that the accuracy of probability can be improved by leveraging information from discrete probabilistic lexicons. They go on to discuss the difference between "automatically learned lexicons" and "manual lexicons," and how these can be integrated into NMT systems, and conclude that as a result of incorporating discrete probabilistic lexicons into NMT systems "we achieved substantial increases in BLEU (2.0-2.3) and NIST (0.13-0.44) scores, and observed qualitative improvements in the translations of content words."

In summary, although the major NMT systems (except for Baidu), do not currently incorporate lexicons, with some effort they can be configured to do so. It is also clear that integrating lexicons into NMT systems is highly desirable since it will lead to major improvements in translation quality. Ideally, NMT should take advantage of the positive aspects of SMT and merge them into new kind of hybrid system that offers the best of both worlds.

4. Experiments and Results

Both traditional MT systems as well as state-of-the-art NMT systems often fail to accurately translate Japanese proper nouns, especially POIs. Below are the results of some spot tests using three major NMT engines, namely Google Translate, Bing Translate, Baidu Translate, and NICT's TextTra (phrase-based), on Japanese POIs, Japanese orthographic variants, and Chinese technical terms, and comparing the results with CJKI's large-scale terminology databases.

4.1 Japanese Points of Interest

Our tests to translate 75 Japanese POIs (with focus on railway lines, airports and amusement facilities) into English using the two major US NMT engines gave surprisingly poor results.

Japanese	Google	Bing	CJKI
海の中道線	Midair line of the sea	The middle line of the sea	Umi-no-Nakamichi Line
三角線	Triangle	Triangular line	Misumi Line
鬼の城公園	Demon Castle Park	Demon Castle Park	Oninojo Park

Table 1. POIs by Google and Bing

Using the major Asian engines (Baidu and NICT) for the same POIs gave the following results:

Japanese	Baidu	NICT	CJKI
海の中道線	The sea line	海の中道線	Umi-no-Nakamichi Line
三角線	Misumi	Misumi Line	Misumi Line
鬼の城公園	Demon Castle Park	Oni Castle Park	Oninojo Park

Table 2. POIs by Baidu and NICT

4.2 Evaluation of results

Our institute (CJKI) uses five methods to determine the level of accuracy of POI translation, in increasing order of accuracy. (1)*Transliteration* refers to representing the source script in another script, as in JN 幕張国際展示場 to ZH 幕张国际展示场, (2) *phonemic transcription*, representing the phonemes of the source language, as in romanizing JN 東京中央ゴルフ場 to *Tokyo Chuo Gorufujo*, (3) *semantic-phonemic transcription* combines semantic transcription with phonemic transcription, as in JN 東京中央ゴルフ場 translated to EN *Tokyo Chuo Golf Course*, (4) *semantic transcription* translates components into the target language, as in JN 幕張国際展示場 to ZH 幕张国际展览馆 and JN 東京中央ゴルフ場 into EN *Tokyo Central Golf Course*, and (5) *human translation*, which is translating to the correct semantic equivalent (the "official" name), such as JN 幕張国際展示場 to ZH 幕张国际展览中心 and JN 東京中央ゴルフ場 to EN *The Central Golf Club, Tokyo*.

The first four can be done algorithmically by referencing component mapping tables and a conversion rules database; that is, semiautomatically with some human proofreading. The fifth, the highest level, can be done accurately only by looking up in hand-crafted lexicons, such as CJKI's proper noun databases, which have served as the gold standard in the Named Entities Workshop (NEWS) transliteration task (Zhang, et al., 2012).

The success rate for the four MT engines tested was less than 50% (Google 47%, Microsoft 40%, Baidu 39%, and NICT 47%). "Success" is defined as level 5 above, meaning that the results should be (almost) identical to the entries in CJKI's POI databases, which have been manually proofread. Comparing these results to CJKI's, it is clear that some errors result from translating the POI components literally (semantic transcription), rather than the named entity as a whole. For example, 鬼の城公園 was translated as 'Demon Castle Park' since 鬼の城 consists of 鬼の 'demon' + 城 'castle', whereas the actual name of this park in English is 'Oninojo Park'. That is, 鬼の城公園 was not recognized as a named entity but was translated literally component by component.

4.3 Orthographic variation

It seems as if NMT engines do not perform orthographic normalization or disambiguation for Japanese. Since Japanese has a highly irregular and unstable orthography, this has a major negative impact on Japanese translation quality. Let's consider the orthographic variants for the following three words:

English	Reading	Var. 1	Var 2	Var. 3
sun	Hi	日	陽	
mansion	yashiki	屋敷	邸	
shine	sasu	差す	さす	射す

Table 3. Typical variants in Japanese

This means that a sentence like *hi no sasanai yashiki* 'a mansion that gets no sunshine' can have such variants as 日の差さない屋敷, 日の差さない邸, 陽の差さない屋敷 and 陽の射さない邸.

Running some of these through Google and Bing we get:

Japanese	Google	Bing
日の差さない屋敷	A dwindling residence	A house with no sun
日の射さない屋敷	A mansion that does not shine.	She mansion of the day.
日のささない屋敷	A daydreaming residence.	A mansion with no sun
陽のささない屋	A ya man who does not sunlight.	A house with no sunshine

Table 4. Japanese variants by Google and Bing

An analysis shows (1) that though these phrases are 100% equivalent, they are being considered as distinct, and (2) that no orthographic normalization takes place. For example, Google translated 陽 *hi* 'sun' to the mysterious 'ya man' and is not aware that it is an orthographic variant of 日 *hi* 'sun'. For Bing, 'A house with no sunshine' is 100% correct, but 'She mansion of the day' makes no sense. Baidu and NICT give similarly poor results:

Japanese	Baidu	NICT
日の差さない屋敷	There's no day at home	Residence that deprive Japan of.
日の射さない屋敷	Day without sunshine house.	Residence not days.
日のささない屋敷	Deprive of the residence.	Residence which do not refer to date.
陽のささない屋敷	The residence where no.	The mansion where the sun never bites.

Table 5. Japanese variants by Baidu and NICT

Note that NICT often interprets 日 as 'date' or 'day', rather than 'sun'. Here too there are some translations that make no sense, such as Baidu's 'There's no day at home' and NICT's 'Residence that deprive Japan of'. Clearly, none of the MT engines surveyed is doing orthographic normalization, which is critical for Japanese.

4.4 Technical terminology

Translation quality depends on such factors as the size and quality of the training corpus, the MT model and algorithms, and supporting lexicons. Despite the dramatic contributions of NMT to translation quality, the problem of unknown vocabulary remains (He et al., 2016), especially for names entitied like POIs and the huge number of technical terms. Some systems, such as NICT's, have been trained on patent corpora and thus achieve good accuracy in patent translation (Sumita, 2013).

Chinese	CJKI	Google	Bing	Baidu	NICT
类骨质	osteoid	bone-like	bone type	osteoid	bone
孢子丝菌病	sporotrichosis	spore mycosis	spore silk fungus disease	histoplasmosis	Spore 丝菌病
亚硫酐	sulfurous anhydride	aulfurous acid	arian	sulfurous anhydride	亚硫酐

Table 6. Technical terms by four NMT engines

Our spot checks have confirmed that NMT engines do perform better in the domains of science and technology than in translating named entities such as POIs. Nevertheless, the lack of technical terminology lexicons does have a negative impact. For example, comparing CJKI's Chinese technical term databases (millions of entries) demonstrates that the NMT results are often incorrect for some medical terms, as shown in Table 6.

5. Lexical Resources

5.1 Very Large-Scale Lexical Resources

The CJK Dictionary Institute (CJKI), which specializes in CJK and Arabic computational lexicography, has for decades been engaged in research and development to compile comprehensive lexical resources, with special emphasis on dictionary databases for CJK and Arabic named entities, technical terminology, and Japanese orthographic variants, referred to as Very Large-Scale Lexical Resources (VLSLR). Below are the principal resources designed to enhance MT quality.

5.2 Japanese resources

1. The *Japanese Personal Names Database* covers over five million entries, including hiragana readings, numerous romanized variants and their English, SC, TC, and Korean equivalents.
2. The *Japanese Lexical/Orthographic Database* covers about 400,000 entries, including *okurigana*, kanji, and kana variants for orthographic disambiguation and grammar codes for morphological analysis.
3. The *Comprehensive Database of Japanese POIs and Place Names*, which covers about 3.1 million entries in 14 languages.
4. The *Database of Katakana Loanwords.*

5.3 Chinese resources

1. The *Comprehensive Simplified Chinese to Traditional Chinese Mapping Tables* (C2C) exceeds 2.5 million entries. This covers general words, named entities and technical terms mapped to their TC equivalents, including such attributes as POS codes and type codes, and supports all three conversion levels, namely code, orthographic and lexemic conversion.
2. The *Database of 100 Million Chinese Personal Names*, an extremely comprehensive resource (under construction), covers Chinese personal names, their romanized variants, dialectical variants for Cantonese, Hokkien and Hakka, multilingual coverage for English, Japanese, Korean, and Vietnamese.
3. The *Database of Chinese Full Names* covers 4 million Chinese full names of real people.
4. Miscellaneous mapping tables such as large-scale pinyin databases showing the difference between SC and TC pronunciation, and others.

6. Conclusions

With computer memory being inexpensive and virtually unlimited, it is no longer necessary for traditional MT systems to over-rely on corpora and algorithmic solutions. The time has come to leverage the full power of large-scale lexicons. As for NMT, although most engines do not currently incorporate lexicons, clearly the effort to do so is desirable since it will lead to major improvements in translation quality. Although "lexicon integration" does pose technical challenges, it is a worthwhile goal and deserves the serious attention of NMT researchers and developers. Ideally, a new kind of “hybrid NMT” that leverages the power of traditional MT systems combined with neural networks should be developed.

7. Bibliographical References

Arthur, P., Neubig, G. and Nakamura, S. (2016). Incorporating Discrete Translation Lexicons into Neural Machine Translation. In *Proceedings of EMNLP 2016: Conference on Empirical Methods in Natural Language Processing*, Austin, Texas.

Brill, E., Kacmarcik, G. and Brockett, C. (2001). Automatically Harvesting Katakana-English Term Pairs from Search Engine Query Logs. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*, pages 393-399, Tokyo, Japan.

Emerson, T. (2000). Segmenting Chinese in Unicode. In *Proceedings of the 16th International Unicode Conference*, Amsterdam

Halpern, J. and Kerman, J. (1999). The Pitfalls and Complexities of Chinese to Chinese Conversion. In *Proceedings of the Fourteenth International Unicode Conference, Cambridge*, MA.

Halpern, J. (2008). Exploiting Lexical Resources for Disambiguating Orthographic CJK and Arabic Orthographic Variants. In *Proceedings of LREC 2008*. Marrakesh, Morocco.

He, W., He, Z, Wu, H., and Wang, H. (2016). Improved Neural Machine Translation with SMT Features. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. Phoenix, AZ.

Jacquemin, C. (2001). *Spotting and Discovering Terms through Natural Language Processing*. MIT Press, Cambridge, MA.

Kwok, K.L. (1997). Lexicon Effects on Chinese Information Retrieval. In *Proceedings of the 2nd Conference on Empirical Methods in NLP*. ACL, 141-148, Stroudsburg, PA.

Luong, M., Pham, H. and Manning, C.D. (2015). Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421, Lisbon, Portugal.

Lunde, K. (2008). *CJKV Information Processing*. O'Reilly & Associates, Sebastopol, CA.

Mediani, M., Winebarger, J. and Waibel, A. (2014). Improving In-Domain Data Selection For Small In-Domain Sets. In *Proceedings of the 11th International Workshop on Spoken Language Translation (IWSLT)*, Lake Tahoe, USA.

More, S., Dhir, G.S., Daiwadney, D. and Dhir, R.S. (2016). Review on Language Translator Using Quantum Neural Network (QNN). *International Journal of Engineering and Techniques*, Volume 2 Issue 1, Jan - Feb 2016, Chennai, India.

Nakagawa, T. (2004). Chinese and Japanese Word Segmentation Using Word-Level and Character-Level Information. In *Proceedings of the 20th international conference on Computational Linguistics*, p.466-es, Geneva, Switzerland.

Sumita, E. (2013). Multi-Lingual Translation Technology: Special-Purpose System for Multi-Lingual High-Quality Translation. *Journal of the National Institute of Information and Communications Technology*, pages 35-39, Tokyo, Japan.

TAUS Executive Forum Tokyo. (2017). https://www.taus.net/events/conferences/taus-executive-forum-tokyo-2017. Retrieved May 8, 2017.

Tsou, B.K., Tsoi, W.F., Lai, T.B.Y., Hu, J. and Chan, S.W.K. (2000) LIVAC, A Chinese Synchronous Corpus, and Some Applications. In *Proceedings of the ICCLC International Conference on Chinese Language Computing*, pages 233-238, Chicago.

Yu, S., Zhu, X. and Wang, H. (2000). New Progress of the Grammatical Knowledge-base of Contemporary Chinese. *Journal of Chinese Information Processing, Institute of Computational Linguistics*, Peking University, Vol.15 No.1.

Zhang, M., Li, H., Liu, M. and Kumaran, A. (2012). Whitepaper of NEWS 2012 shared task on machine transliteration. In *Proceedings of the 4th Named Entity Workshop (NEWS '12)*. Association for Computational Linguistics, Stroudsburg, PA.

Combining Concepts and Their Translations from Structured Dictionaries of Uralic Minority Languages

Mika Hämäläinen, Liisa Lotta Tarvainen, Jack Rueter
Department of Modern Languages, University of Helsinki
Fabianinkatu 33, 00014 University of Helsinki
mika.hamalainen@helsinki.fi, liisa.l.tarvainen@helsinki.fi, jack.rueter@helsinki.fi

Abstract

Building language resources for endangered languages, especially in the case of dictionaries, requires a substantial amount of manual work. This, however, is a time-consuming undertaking, and it is also why we propose an automated method for expanding the knowledge in the existing dictionaries. In this paper, we present an approach to automatically combine conceptually divided translations from multilingual dictionaries for small Uralic languages. This is done for the noun dictionaries of Skolt Sami, Erzya, Moksha and Komi-Zyrian in such a way that the combined translations are included in the dictionaries of each language and then evaluated by professional linguists fluent in these languages. Inclusion of the method as a part of the new crowdsourced MediaWiki based pipeline for editing the dictionaries is discussed. The method can be used there not only to expand the existing dictionaries but also to provide the editors with translations when they are adding a new lexical entry to the system.

Keywords: combining dictionaries, low-resource languages, semantic dictionaries

1. Introduction

Words in a language are often considered to be the basic units of a dictionary. Like the language itself, dictionaries also contain elements specific to many different fields. Languages contain words composed of different phonemes, morphemes and syntactic structures. One can use a language by merely employing nothing but words. To understand a language fully, however, one also needs to understand the culture. A word is defined in the Concise Oxford English dictionary first as "[a] single distinct meaningful element of speech or writing, used to form sentences with others". Dictionaries contain words which are referred to as lexemes, lexical items, head words or lemmas in order to specify their specific use in dictionaries and to distinguish them from their use in the spoken language. In general, the head word of a dictionary entry is given as nominative singular for adjectives and nouns, while infinitive (or third person singular) forms are used for verbs cf. (Kastovsky, 1992). Depending on the language, a dictionary provides different inflectional forms essential to the language, and examples to illustrate usage.

The problem many minority languages are facing is that their language resources are very limited. Small Uralic languages, which are the focus group of this paper, are no exception to this rule. Having good language resources in the form of a dictionary would benefit these languages, because they can be used in various tasks such as machine translation (Brown, 1997) and pedagogy (Antonsen; Huhmarniemi & Trosterud, 2009) when teaching the language to non-native speakers. Thus, their impact on the revitalization of endangered languages would be high.

Furthermore, the building of a multi-language dictionary of this variety is essential for supporting minority languages. Due to the pressure of major languages, the amount of speakers of these minority languages has diminished radically. A multi-lingual dictionary that concentrates on these minority languages is one possibility to maintain the vocabulary of these languages and present them to larger audiences and users.

Building quality resources for low resource languages requires a good amount of manual work. Such work has been on-going for Uralic languages in the Giellatekno infrastructure[1] for the past decade in the form of dictionaries, morphological analyzers, constraint grammars and rule-based machine translation tools among others.

This paper builds upon the manually crafted bilingual dictionaries in the Giellatekno infrastructure and proposes an automatized way of combining semantic and translation knowledge from these dictionaries. This method will expand the existing language resources by providing completely new translations and language pairs. Later on, this work will be incorporated in the Sanat[2] MediaWiki platform (Rueter & Hämäläinen, 2017), which is used to crowdsource the future development of the Giellatekno language resources.

2. Related Work

Standardizing bilingual or multilingual dictionaries through combining entries from different corpora and sources, such as WordNet (Miller, 1995), in order to build well-structured dictionaries for machine translation and other uses has become a central source of interest over the past decades. A great many past studies, however have focused on majority languages.

The aim of (Klavans & Tzoukermann, 1990) is to focus on the methods that combine structured but incomplete or only partially complete information from the dictionaries in order to create a bilingual lexical database. The aim of their BICORD system is to create a bilingual corpus based dictionary which is able to combine lexical information both from the bilingual corpora and from the machine-

[1] See http://giellatekno.uit.no/index.eng.html

[2] Accessible on http://sanat.csc.fi/

readable dictionaries. The goal is to indicate lexical equivalence with associated translations. The basis is in linguistic research combined with the data collected from machine-readable dictionaries. Statistical techniques and machine-readable dictionaries combined with the methods from standard linguistic has been used as the central guideline. The study concentrates on the action verbs in French and in English with the approach to analyze their use and behavior in a bilingual corpus.

Ji's et al. (2016) paper concentrates on matching lexical entries in order to reduce the quantity of the term data and raise the quality of the lexica. The quality is enhanced by matching and combining lexical entries and resources from multiple dictionaries. The quality criteria are based on linguistic and terminological research work such as Levenshtein distance string metric. In Ji's et al. research this distance measure is tested on WordNet and BabelNet in order to change the quality criteria into language-independent frequency-based measures. The idea of the quality criteria is to detect well-constructed entries from the model dictionary, which in this research is the Princeton WordNet. The aim is to find duplications and other errors in the linked lexical entries and resources and see if reducing the term data can be compensated by more exact structure and content.

Hovy's approach (Hovy, 1998) is focused on describing the use and results of semi-automated cross-ontology concept. The goal is to create ontologies which are based in large-scale machine translations. The focus is in combining the ontologies in order to create a practical large-scale ontology for free use in the web. The primary stress lies in combining and standardizing these ontologies. The main use for these ontologies is in machine translation but they can be used in other purposes as well.

Navigli and Ponzetto (2010) are presenting a multilingual semantic network, BabelNet, with the aim to produce a lexical resource with high accuracy. By the automatic mapping between BabelNet's two resources, the English Wikipedia and WordNet, they have provided an automatic construction in order to create a large multilingual lexicon. Their aim is to present a new methodology for the automatic construction of a multilingual lexical knowledge resource of this art. The project is based on combining the lexicographic and encyclopedic knowledge. Navigli and Ponzetto have unified the word senses as concepts and semantic pointers between synsets as relations from the WordNet with all the encyclopedic entries, such as individual pages, as concepts from Wikipedia. The semantically unspecified relations in Wikipedia are collected from hyperlinked texts. These two resources give lexical knowledge of different type, from which one is concentrated on the named entities and the other on concepts. Linking the two knowledge resources and the use of two different resources has displayed that it will provide large-scale lexical resources which work as the basis for BabelNet.

Wehrli et al. (2009) present the MulTra (Multilingual Translation) project. The aim of the MulTra project is to develop an efficient grammar based translation model which is able to cover several different languages. The project is also based on object-oriented software design. The basis for this grammatically oriented approach is abstract and generic linguistic, which is based for example on Noam Chomsky's generic grammar. The aim of the research is to grow the amount of language pairs and develop them but also to reduce the development costs. First, the project is concentrated to five large European languages (such as German and French) but also later on on languages with different writing systems (such as Russian and Greek). The approach which is based on abstract and generic grammar is seen as worth developing while there is a possibility to benefit from the abstract levels of a language. In order to reduce the complexity of the major part of bilingual lexical databases, they can be automatically derived by transitivity.

The aim of Ji's et al. (2014) paper is to present a workflow of merging and matching the anonymous special language terms from Web terminology sites. The term candidates collected from the Web terminology sites with the help of TermFactory are unified and merged with the resources that are already found in the TermFactory, which is a semantic Web framework designed for professional multilingual terminological use. The goal is to produce more high-quality term candidates for professional terminology use. One of the central themes in the paper is not to lose the provenance of the term candidates when merging them in TermFactory. The long-term goal is to produce clear terminology with high standards in order to serve high-quality translations.

3. Combining Dictionaries

In this part, we present our proposed approach to combine multilingual dictionaries for small Uralic languages. The dictionary of each language contains translations for one or more languages for each lemma. This means that a word in Moksha might be translated into Finnish and English while a word meaning the same thing in Erzya might have translations in Russian and Finnish. By combining these translations, we can improve the translation coverage of both dictionaries.

As a result of the combination, we will first of all have, more translations for lemmas in the four languages, and new target languages that haven't existed before in the original dictionaries. This means that Skolt Sami will get translations into a new language French through Erzya, and Erzya will get Norwegian translations through Skolt Sami, for example.

Second of all, the meaning groups, once combined, will have IDs that are shared across all of the four dictionaries, which will turn the strict tree structure of the dictionaries into a graph-like structure which is essentially a step towards having something similar to WordNet for Uralic languages.

3.1 The Initial Data

The language resources in use are the dictionaries of Moksha (Rueter et al., 2018), Skolt Sami (Rueter; Rießler & Lehtinen, 2018), Erzya (Rueter et al., 2018) and Komi-Zyrian (Rueter; Kokkonen & Fedina, 2018) that are freely available in XML format in the Giellatekno infrastructure. All of these dictionaries follow the same XML syntax thus making their combination process easier. We limit the research to the noun dictionaries only. To illustrate the

syntax, an example of a simplified entry in the Skolt Sami dictionary is provided.

```
<e>
        <lg>
                <l pos="N">lää'ǩǩ</l>
        </lg>
        <mg>
                <tg xml:lang="eng">
                        <t>law</t>
                </tg>
                <tg xml:lang="deu">
                        <t>Gesetz</t>
                </tg>
        </mg>
        <mg>
                <tg xml:lang="eng">
                        <t>right</t>
                </tg>
                <tg xml:lang="deu">
                        <t>Recht</t>
                </tg>
        </mg>
</e>
```

In the example syntax, we see that the polysemic Skolt Sami word *lää'ǩǩ* is divided into two different mg elements. Mg is short for meaning group, this means that all the translations under the same meaning group refer to the same semantic concept. Therefore, when combining the translations from different dictionaries, it's important to do it based on the meaning groups. This should solve the problem of polysemy for us, provided that the meaning groups are divided in the correct way.

Combining the meaning groups poses some challenges. First of all, the meaning group division has been hand-crafted by different authors at different times, i.e. they have fuzzy boundaries. Second, they don't have ID numbers, which would link two meaning groups pointing to the same semantic concept in different dictionaries together, thus they have to be combined based on the translations they have. In the combination step, we will associate IDs to the meaning groups to solve this deficiency in the XML dictionaries.

However challenging the problem might be, our combination approach will directly benefit the limited scope of linguistic resources in these small languages. Not to mention, the Giellatekno infrastructure sports with some 20 more XML dictionaries sharing the exact same structure. Most of these dictionaries have translations in Finnish, English, Russian and Norwegian (Bokmål) but also in other Uralic and European languages depending on the authors. This means that this approach could be used in a wider set of languages in the infrastructure with little to no modifications.

In addition to meaning groups and translations, the XML dictionaries contain a variety of different kinds of information. For example, the entries can have continuation lexica for morphological analyzers, longer textual definitions, sound samples, example sentences, notes on derivation, etymology and so on. These additional pieces of information are not used in the combination approach.

3.2 The Combination Algorithm

The combination algorithm takes the XML dictionaries as input and produces new versions of them with more translations added inside of the correct meaning groups. This requires the structure of the data to be changed. In the Giellatekno XMLs, meaning groups are stored under of lemmas. This means that the same meaning can be represented by multiple meaning groups inside of a dictionary in the case of synonyms. When two lemmas share a meaning, they will still have two separate meaning groups because of the structure of the XMLs.

The first step is to extract all the meaning groups from the XML dictionaries for the four languages under study and to assign them with unique IDs. This means that instead of storing the contents of a meaning group under a lemma, we simply create a pointer to the meaning group based on its ID from the lemma entry. Then, when we do the combinations and we update the ID references to the new meaning groups resulting from the combination of two or more existing meaning groups. That is, if meaning groups with IDs 1, 2 and 3 are combined, we modify the pointers from the lemma entries pointing to 2 and 3 for each language to point to the meaning group 1 which now contains the information from 1, 2 and 3.

The combination itself is done by following the formula for combining two concepts based on the similarity of their descriptions introduced by (Hovy, 1998). In their approach, the formula was used for two descriptions in English. In our approach, we treat the translations of a translation group, i.e., translations in one language, as a description. This means that, for instance, the Skolt Sami word *podd* would have one description containing the English translations *moment* and *while* and another description containing the Finnish translations *hetki* and *tovi* and so on for other languages.

$$similarity = \frac{|D1 \cap D2|}{\min(|D1|, |D2|)} * |D1 \cap D2|$$

D1 and D2 are sets of words used in the description being compared to each other. In our case, they are the translations in the same language of two different meaning groups, for example [*current*, *river*] and [*current*, *river*, *torrent*]. The formula gives us a similarity score between 0 and 1 that we can directly use to determine whether two meaning groups should be combined or not.

We calculate the similarity score for all the matching languages in the meaning groups. For example, if a meaning group has translations in Finnish and English, and it's compared to another one with translations in Finnish and Russian, we can only calculate the similarity based on Finnish. Furthermore, we take the minimum of the similarity scores. The hypothesis is that if one language differs in its translations, that particular language makes a semantic distinction the others don't and thus the meaning groups shouldn't be combined. Meaning groups with translations in only one language will not be considered at all in the combination. If the minimum similarity score is one, we combine the two meaning groups.

4. Results and Evaluation

In this part, we will discuss the results of the combination algorithm and conduct an evaluation for the combined Skolt Sami dictionary.

4.1 Results

The initial data set includes a total of 13095 individual meaning groups that fill the requirement of having translations to at least two different languages. After using the algorithm, the number of unique meaning groups dropped to 9914 which means that 3181 meaning groups were merged into other meaning groups. In other words, the amount of unique meaning groups was reduced by approximately 24 %.

On the average, this method added 2.6 new translations to each meaning group along with 1.5 new languages. The low number of new languages can be explained by the fact that the data is highly dominated by translations into Finnish and Russian while the other languages appear fewer times in the data.

In an effort to avoid secondary source language word reference, the four source languages, Erzya, Komi-Zyrian, Moksha and Skolt Sami, were not added to the whole number of translations. Hence the languages originally present in the Komi-Zyrian meaning groups, for example, were Russian, Finnish and English, whereas the larger number of translation sets were augmented to include the majority Norwegian, German and French languages, as well as several Sami languages (but not Erzya, Komi-Zyrian, Moksha or Skolt Sami).

4.2 Evaluation

Due to the fact that finding people with high enough skills in the minority languages and all the majority languages the dictionaries have translations for, we decided to limit the evaluation to the combined Skolt Sami dictionary. The combined dictionary was read through and commented by an expert linguist with high proficiency in Skolt Sami among other languages.

The annotator was asked to comment them based on two factors: are the new translations suitable for the word they have been added to and do the new translations semantically fit into the meaning group.

After the annotation was done by the annotator, we looked into the errors he had found and categorized them based on the source of the error. All in all, 1325 new translations were added to the Skolt Sami noun dictionary out of which 843 were correct translations. This means that around 63% of the new translations were correct, but as we will learn, the low accuracy wasn't entirely due to the approach.

Further analyzing the annotations, we were able to identify five different sources of error in the combined meaning groups for Skolt Sami.

4.2.1 Meaning Group Division

Looking at the source of the erroneous translations, we can see that a vast majority of them, around 84% is due to a non-accurate meaning group division in one of the source language dictionaries. A 54 % of them come from the Erzya dictionary, 27 % from Komi-Zyrian, 18 % from Skolt Sami and only one percent from Moksha.

What this means, is that essentially translations that should have been in different meaning groups, were put inside of the same meaning group by the people editing the dictionaries. For example, the Erzya word *auo* had Finnish translations *rupla* (ruble), *votka* (vodka) and *valkuainen* (egg white) inside of the same meaning group, which is against the standards set for the XML structure. As a result, the Skolt Sami word *rubbâl* which only means ruble also got the other two meanings of the Erzya word.

Inaccurate meaning group division even in the Skolt Sami dictionary itself caused wrong translations to appear with words that are synonymous only in one sense. For instance, the Skolt Sami word *puõlvvõk* had two semantically different translations in Finnish inside of the same meaning group: *sukupolvi* (generation) and *ikäryhmä* (age group), which resulted in the erroneous translation *ikäryhmä* to appear under another Skolt Sami word *puõlvv* which only means generation, not age group.

The fact that a majority of the errors came from non-properly formatted data, gives us hope that this approach would yield better results if the meaning group division was done in a more accurate fashion. This evaluation reveals that there's much work to be done in the original dictionaries as well.

4.2.2 Wrong Translations

Another error source that wasn't due to the performance of the combination algorithm, was incorrect translations in the source dictionaries. This covers around 12 % of all the errors. A 33 % of the incorrect translations were found in the Erzya dictionary, a 16 % in the Komi-Zyrian dictionary 51 % in Skolt Sami and none in the Moksha dictionary.

For example, the Skolt Sami word *kruugg* had initially a wrong translation *tyre* which caused three other words with the same meaning to be added as translations: an alternative English spelling *tire*, the Norwegian *dekk* and Russian *звено*. This illustrates how a simple wrong translation can cause more erroneous translations to be included for a word.

This group of errors as well as the first one reveal more about the problems already present in the data than about the accuracy of the combination algorithm. This shows that more work is needed to do a lexicographical evaluation for all of these dictionaries.

4.2.3 Homonymy

Homonymy in one of the majority language translations accounts for a 1 % of the errors. This might occur if the difference in meaning is not captured by other translations in the meaning group.

More specifically, the Finnish homonym *kuusi* which can either mean the number *six* or a *spruce* caused translations for the number *six* in other languages to occur under the Skolt Sami word *lõsttkuõss* which refers to the tree. This was due to the lack of translations in different languages for the word *lõsttkuõss* in the original dictionary. The word was only translated into Finnish and Latin while the word *kuuđâs* referring to the number *six* had translations to Norwegian, Finnish, English, German and Russian.

The error caused by homonymy in one language could be avoided by either making sure that homonymous words have at least one synonym in the translations. This would make sure that the combination algorithm would consider the meaning groups different. Another way would be to modify the algorithm so that instead of just requiring translations in at least two languages, it would require at least two shared languages between the meaning groups it compares. This would effectively solve the issue of homonyms, but it might also result in a lower number of combinations of meaning groups.

4.2.4 Polysemy

Another minor source of error, around 2 %, comes from polysemy. Polysemy can manifest much like homonymy in some cases. For example, the Skolt Sami word *nju'nnjel* refers to a *smew*, which in Finnish is *uivelo*. The Finnish word can also mean a *weak person* which caused such Russian translations *щуплый*, *хилый* and *субтильный* to be added to the Skolt Sami word. The remedy for this kind of an error from polysemy is exactly the same as for homonymy.

An interesting error caused by polysemy is when there are enough different languages to be used in comparing the meaning groups, but only one of them makes a distinction in meaning. The Skolt Sami word *njuhččâm* got translations to French *langue,* Finnish *kieli* and Russian *язык*. These are all right translations because they mean *tongue*, the problem, however, is that they also mean *language* which is why the English word *language* got also added as a translation. This is an interesting example, how many distant languages might share the same polysemy, while others don't.

4.2.5 Peculiar Translations

The rest of the errors, around 1 %, came from unusual translations. An example of such a translation is a cultural translation of the Komi-Zyrian word *саралан кар*, which was translated as *capital* and *seat of the Czar*. While in the Russian context, *seat of the Czar* might very well refer to the exact same city as the word *capital*, in Skolt Sami context, that is not the case as it is also spoken in Finland and Norway. This is the only case of a cultural translation being added to Skolt Sami from another language, but if this method is applied in the context of socio-culturally very distant languages, these kinds of errors might be more frequent.

Another case of an unusual translation is the use of a non-nominative case in the translations. This was, for instance, in the case of the Erzya word *валске* which was translated into Finnish in nominative *aamu* (morning) and in adessive *aamulla* (in the morning). This resulted in the Skolt Sami word *ee'đääldõž* to receive the translation *aamulla*, however the Skolt Sami word alone in nominative doesn't bare the meaning of *in the morning*. These non-nominative translations are a peculiar case, as putting them into a separate meaning group doesn't really make too much sense. This would call for a different kind of a translation field in the XML schema to separate translations from a translations that serve more as a usage example.

5. Discussion and Future Work

We have limited this paper to work with nouns only. In the future, we intend to apply this approach to all parts-of-speech, and eventually, to all the languages in the Giellatekno infrastructure. From the evaluation, we can say that the division into meaning groups in the Giellatekno dictionaries is not as accurate as it should be. This calls for more work in that field, as well.

Erroneous translations in the source data were found and that clearly points towards more manual work that should be done in order to make sure that the dictionaries are a reliable source for different language users and learners. The source data was also infested by missing meaning groups, meaning that the translations referring to entirely different concepts were, against the structure imposed by the Giellatekno XML standard, added inside of a single meaning group.

Whereas the wrong translations must mostly be fixed manually due to a lack of a better resource to evaluate the correctness of the translations automatically, the meaning group division, however, is something we are interested in looking at from an automated perspective in the future. Given that the translations are mainly in majority languages with a plethora of LRs available, we could probably have a look at utilizing distributional semantics approaches, such as word2vec, to automatically divide meaning groups that contain translations referring to entirely different concepts.

In this paper, we only discussed combining meaning groups and the translations inside of them. When we solve the meaning group division problem in the source data, we can actually link words of different minority languages together through the meaning groups they refer to, i.e., if we know that the Skolt Sami word *piânnai* and the Erzya word *киска* both refer to the same meaning group for *dog*, we can include the Skolt Sami word in the Erzya dictionary as a translation and vice versa.

One of the main motivations for this research was the MediaWiki based online dictionary (Rueter & Hämäläinen, 2017) we are building on the Helsinki-based Sanat server[3]. This dictionary makes it possible to edit Giellatekno's language resources easily in a wiki environment. In the future, we want to incorporate the approach proposed in this paper as a part of the MediaWiki workflow when editing and adding lemmas to the online dictionary.

6. Conclusions

In this paper, we have proposed an approach to combine dictionaries for minority languages in an automatic fashion. The majority of the errors in the results produced by the approach (around 96 %) were due to errors either in translations or meaning group division in the original dictionaries. This shows that providing that the original data has only a minimal number of errors, this method can produce reliable combinations of different dictionaries automatically.

This method helps us add beneficial information to all the minority-language LRs by deriving from manual work done separately for each of them. Because of the

[3] http://sanat.csc.fi

notoriously limited amount of resources for these minority, or even endangered, languages, this automatic method provides a notable improvement to the existing LRs.

The improved dictionaries will tangibly enhance the NLP technology resources already in place in the Giellatekno infrastructure, one such area to benefit might be machine translation.

7. Bibliography

Antonsen, L., Huhmarniemi, S., & Trosterud, T. (2009). Interactive pedagogical programs based on constraint grammar. *Proceedings of the 17th Nordic Conference of Computational Linguistics. Nealt Proceedings Series 4.*

Brown, R. D. (1997). Automated dictionary extraction for "knowledge-free" example-based translation. *7th International Conference on Theoretical and Methodological Issues in Machine Translation*, (pp. 111–118).

Hovy, E. (1998). Combining and Standardizing Large-Scale, Practical Ontologies for Machine Translation and Other Uses. *In The First International Conference on Language Resources and Evaluation (LREC)*, (pp. 535-542).

Ji, K., Wang, S., & Carlson, L. (2014). Matching and Merging Anonymous Terms from Web Sources. *International journal of web and semantic technology*, (pp. 53-69).

Ji, K., Wang, S., & Carlson, L. (2016). Multilingual Dictionary Linking and Aggregation: Quality from Consistency. *Proceedings of the Fourth International Workshop on Linked Data for Information Extraction (LD4IE)*, (pp. 51-62).

Kastovsky, D. (1992). Semantics and vocabulary. In R. M. Hogg (Ed.), *The Cambridge history of the English language I* (pp. 290-408). Cambridge University Press.

Klavans, J., & Tzoukermann, E. (1990). The BICORD System Combining Lexical Information from Bilingual Corpora and Machine Readable Dictionaries. *COLNG 1990 Volume 3: Papers presented to the 13th International Conference on Computational Linguistics.*

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM, 38*(11), pp. 39-41.

Navigli, R., & Ponzetto, S. (2010). BabelNet: Building a Very Large multilingual Semantic Network. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, (pp. 216-225).

Rueter, J., & Hämäläinen, M. (2017). Synchronized Mediawiki Based Analyzer Dictionary Development. *The Third International Workshop on Computational Linguistics for Uralic Languages*, (pp. 56–62).

Wehrli, E., Nerima, L., & Scherrer, Y. (2009). Deep Linguistic Multilingual Translation and Bilingual Dictionaries. *Proceedings of the Fourth Workshop on Statistical Machine Translation*, (pp. 90-94).

8. Language Resource References

Rueter, Jack, Bolkin, Oleksandr, Abramov, Kuzma, & Erina, Olga. (2018). rueter/gt-xml-lexica-myv: Erzya To X nouns (Version v1.0) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.1146850

Rueter, Jack, Herrala, Eeva, Feoktistov, Aleksandr, & Salo, Merja. (2018, January 13). rueter/gt-xml-lexica-mdf v1.0 (Version v1.0). Zenodo. http://doi.org/10.5281/zenodo.1146846

Rueter, Jack, Kokkonen, Paula, & Fedina, Marina. (2018). rueter/gt-xml-lexica-kpv: Komi-Zyrian to X nouns (Version v1.0) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.1146852

Rueter, Jack, Rießler, Michael, & Lehtinen, Miika. (2018). rueter/gt-xml-lexica-sms: Skolt Sami to X nouns (Version v1.0) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.1146854

Transfer of Frames from English FrameNet to Construct Chinese FrameNet: A Bilingual Corpus-based Approach

Tsung-Han Yang, Hen-Hsen Huang, An-Zi Yen, and Hsin-Hsi Chen

Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
{thyang, hhhuang, azyen}@nlg.csie.ntu.edu.tw, hhchen@ntu.edu.tw

Abstract

Current publicly available Chinese FrameNet has a relatively low coverage of frames and lexical units compared with FrameNet in other languages. Frames are incompletely specified for some lexical units, and some critical lexical elements are even missing. That results in suitable frames cannot be triggered and filled in practical applications. This paper presents an automatic approach to constructing Chinese FrameNet. We first capture the mapping between English lexical entries and their Chinese counterparts in a large scale sentence-aligned English-Chinese bilingual corpus. Then, a semantic transfer approach is proposed based on word alignments applied to a large balanced bilingual corpus. The resource currently covers 779 frames and 36k lexical units. We apply it to annotate diary and tweet, and achieve overall 86% success rate to provide frame recommendations that are acceptable by annotators. The success rates in terms of source types are 95% and 80% for diaries and tweets respectively.

Keywords: FrameNet, Chinese, frame semantics, lexical units

1. Introduction

Semantic role labeling (SRL) is a fundamental task for many NLP applications. Given a context, SRL is aimed at identifying the semantic roles, or the set of semantic properties and relationships defined over a lexical unit (LU) or a target. The resources such as FrameNet (Baker et al., 1998; Fillmore et al., 2003) and PropBank (Kingsbury and Palmer, 2002) storing abundant information about lexical and predicate-argument semantics have advanced the field of semantic analysis further and made it possible for learning algorithms to train upon.

1.1 FrameNet

FrameNet is a lexical database with rich human-annotated semantic content based on the linguistic theory of Frame Semantics proposed by Fillmore (1982). FrameNet provides both human and machine readable structure for semantic frames, associated with frame elements (FEs), lexical units (LUs), and sample sentences.

The words that evoke a specific frame are called LUs or targets. Accompanied with each frame, there are a set of FEs defining semantic roles that is meaningful to that frame. Most importantly, FrameNet provides over 200,000 human annotated sentences associated to more than 1,200 semantic frames, and forms a great resource for analyzing semantic structures in natural language.

1.2 FrameNet in Chinese

Unlike English FrameNet, which has been in operation since 1997, Chinese FrameNet (You et al, 2005) started constructing the resource in 2005. To date, Chinese FrameNet (CFN) contains 3,947 LUs, 323 semantic frames, and 20,000 annotated sentences. Compared with English FrameNet, in which 13,638 LUs, 1,221 semantic frames, and 200,000 annotated sentences are provided, Chinese FrameNet is considerably smaller. In other words, Chinese FrameNet has a much low coverage of frames and lexical units, and results in limited applications.

1.3 FrameNet Construction in Other Languages

Efforts have been made to construct FrameNet resources in other languages. Most of which construct their resources with human annotation one by one laboriously, such as Japanese FrameNet (Ohara et al., 2003). Park et al. (2014) conduct the construction of Korean FrameNet by hiring trained translators to import 4,025 sentences selected from English FrameNet. Kim et al. (2016) further import additional 1,795 sentences to Korean FrameNet from Japanese FrameNet based on the similarities between these two languages. However, the high construction cost of such resources sometimes hinders it from growing to the proper scale that is applicable for real NLP tasks. Tonelli et al. (2008) propose an algorithm that projects English frames onto Italian ones, so that FrameNet in Italian could be constructed more easily.

In this work, we propose a novel approach to automate FrameNet construction. Based on a large-scale bilingual corpus, we transfer the machine-annotated FEs from English sentences to their Chinese counterparts based on the assumption of semantic equivalence between both source sentences and target sentences in a bilingual corpus. Compared with the manually constructed Chinese FrameNet, our approach results a higher coverage in terms of LUs and sample sentences. Furthermore, filtering strategies are explored to reduce the noise from the automatic generated data. Human verification confirms the quality of the outcomes. Our methodology can be easily adapted to generate the FrameNet-style dataset for other languages.

In the rest of this paper, we first introduce the linguistic resources that support our construction in Section 2. Section 3 presents our methodology and discusses the filtering strategies that are used for quality improvement. Section 4 demonstrates an application of our resource that would improve the annotation process in terms of annotation time consumption. Finally, we conclude our contributions and discuss future work in Section 5.

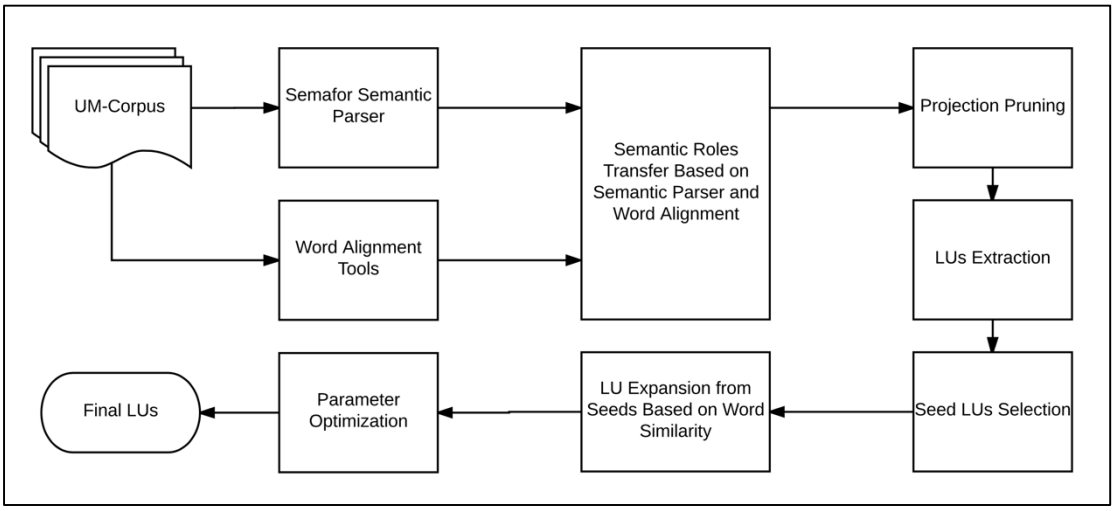


Figure 1: Overview of our approach to Chinese FrameNet construction.

2. Resources

We construct Chinese FrameNet based on UM-Corpus (Tian et al., 2014), which is a large-scale, balanced English-Chinese corpus consisting of 2.2 million parallel sentences from eight genres in a reasonable proportion, including News, Spoken, Laws, Thesis, Education, Science, Subtitle, and Microblog. They are parsed and extracted from online journals (national and international), official websites, online language learning resources (e.g. online dictionary and translation portals), TED, and Microblogs. Tian et al. (2014) apply some well-designed algorithms and tools to speed up the building process, such as document alignment, sentence boundary detection, and sentence alignment. The constructed corpus is manually verified to ensure the quality.

3. Methodology

Figure 1 depicts the overview of our approach to automatic Chinese FrameNet construction. Based on a sentence-aligned bilingual corpus, our basic idea is to transfer the machine-annotated frame information from English sentences to their Chinese counterparts. In Section 3.1, the English semantic parser, SEMAFOR, is performed to label the FEs on English sentences. In Section 3.2, we project the FEs from the English part to the Chinese part, based on the word alignment generated by TsinghuaAligner. The invalid projections are truncated by the strategy described in Section 3.3. In Section 3.4, the candidate LUs are extracted from the machine-annotated Chinese samples. From the candidate LUs in Section 3.4, we further select the frequent ones as the seeds. In Section 3.5, an expansion algorithm is proposed to augment LUs based on the seeds.

3.1 English Semantic Frame Labeling

We label the English part of UM-Corpus with semantic frames for the later step, projection. Here, we utilize SEMAFOR (Das et al., 2010), a state-of-the-art semantic frame parser for English based on the FrameNet ontology. The tool finds the words that are likely to evoke frames, and then labels frame elements for each frame using a log-liner model.

3.2 Projection of Bilingual Frame Elements

Word alignment tools align words in a sentence in the source language to the corresponding words in the sentence in the target language. In our case, we regard English as the source language and Chinese as the target language since we will utilize the alignment information as our basis of finding the projection of semantic relations from English to Chinese between bilingual sentences. In this paper, we employ TsinghuaAligner (Liu and Sun, 2015), which takes the translation probabilities derived from GIZA++ (Och and Ney, 2003) as the central feature in the word alignment process.

The alignment procedure consists of three major steps. Figures 2, 3, and 4 show examples for the tasks respectively. First, we label 2.2M English sentences in the bilingual UM-Corpus with frame semantics using SEMAFOR. As shown in Figure 2, SEMAFOR generates 14M predicate-argument structures from these 2.2M sentences. Figure 3 shows the second step, where TsinghuaAligner is performed to derive English-Chinese word pairs as our basis for the step. Finally, as illustrated in Figure 4, we utilize both the parsed result and the alignment information gathered from SEMAFOR and TsinghuaAligner to produce Chinese FrameNet-style annotations.

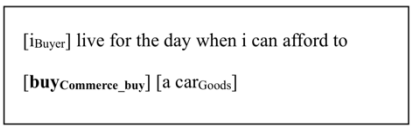


Figure 2: Applying the SEMAFOR parser to UM-Corpus.



Figure 3: Applying TsinghuaAligner from English to Chinese.

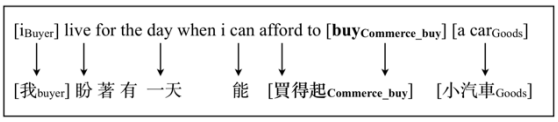


Figure 4: Mapping semantic roles from English to Chinese.

3.3 Pruning off Invalid Projections

From the alignment of bilingual frame elements, we have produced our preliminary results. However, not all frame structures produced in Section 3.1 are successfully projected due to imperfect English-Chinese alignments. Typical projection errors include the following two types:

(1) Missing target or frame elements.
The alignment tools we used do not guarantee full alignment coverage for each word in a sentence. Therefore, some target or frame elements are not projected from English to Chinese if no alignment information is provided.

(2) Incorrect frame elements projection.
Unlike target word in a frame, frame elements may consist of a group of words (i.e., phrase). However, it is possible that some of the words in the group cannot be aligned due to missing alignment information.

To alleviate the errors resulted from word alignments in (1) and ensure the quality of the projected sentences, we set a constraint on the number of frame elements in Chinese projected frame structures, which should be identical to that of English annotations. Based on this constraint, we discard the Chinese projections with missing frame elements. About 34% of the frame structures produced by SEMAFOR are removed in this process due to the aforementioned alignment errors. As a result, we obtain a dataset that consists of nearly 9M Chinese FrameNet-style annotations.

3.4 Lexical Unit Extraction

After aligning and pruning procedures, we have 9M Chinese FrameNet-style annotations as the basis for analysis. There are 779 unique frames in the 9M annotated instances. Compared with the current English FrameNet (FrameNet 1.7), which consists of 1,221 frames, the shortage results from SEMAFOR semantic parser, whose model was trained on 779 frames in FrameNet 1.3. We then locate the target of each frame from the 9M Chinese annotations, and produce 1M candidate LUs. The candidates may contain noise, so we refine them with the following steps.

3.5 Filtering of Lexical Units

We attempt to select reliable LUs from the candidates as the seeds for expansion. Here, we propose a statistical-based approach where the most frequent *N* LUs in a frame are regarded as seed LUs for the frame. We set *N* to be sufficiently small (*N*=10 in our setting) to ensure the quality. The setting produced 7,401 seed LUs, where 93.5% of them are considered valid after human verification. Table 1 shows some examples. Many errors are due to wrong word sense disambiguation in SEMAFOR. For instance, the meaning of the word "lead" should be the winning position during a race, while it is mis-resolved as the heavy, soft, dark grey metal by SEMAFOR. Another issue we noticed is that some word sense in English could not be directly mapped to Chinese. For example, the word 上海 "Shanghai" in English may mean to force someone to do something or go somewhere, while this word only carries the location sense in Chinese. By removing such errors, we could proceed to expand more LUs from the seeds without introducing too much noise.

Frame Name	LU seeds
Being_in_category	看作 (regard as), 相當於 (equivalent to), 等於 (equal to), 等價 (equivalent to), 無異於 (tantamount to), 等同於 (same as)
Food	食物 (food), 食品 (grocery), 蘋果 (apple), 咖啡 (coffee), 魚 (fish), 水果 (fruit), 脂肪(fat), 蔬菜(vegetable), 麵包 (bread)
Inclination	傾向 (tendency), 趨勢 (trend), 發展趨勢 (development trend), 傾斜 (tilt), 斜 (oblique), 變化趨勢 (trend), 趨向 (tend), 方向 (direction), 走向 (direction)
Planting	播下 (sow), 播種 (sow), 播種面積 (seeded area), 種子(seed), 撒(sprinkle)
Traversing	通過 (pass),傳遞 (transfer), 經過 (pass by), 上升 (rise), 交叉 (cross), 過去 (pass through), 穿越 (pass through), 下降 (decline), 跳 (jump), 遍歷 (traverse)
Visiting	客人 (guest), 訪客 (visitor), 嘉賓 (guest), 來賓 (guest), 遊客 (guest), 賓館 (hotel), 賓客 (guest), 招待所 (guest house)

Table 1: Examples of verified LU seeds

3.6 Expansion of Lexical Units

The seed LUs for each frame provide a great basis for the expansion. We expand the LUs based on word embedding. A CBOW word model is trained on UM-Corpus. For each seed LU in a certain frame, we could find a set of similar words based on cosine similarity. To ensure the quality, we set a threshold *t*, which is the minimum cosine similarity that a word could be added to the expanded LUs. Different threshold *t* will produce different LU size with different quality. In order to find an optimal setting, we sample 100 LUs from 100 frames for various threshold *t* as an index of the performance. Figure 5 shows the relationship between the LU size and the quality. The union of the all the sets within each frame forms the final LUs. The growth of the

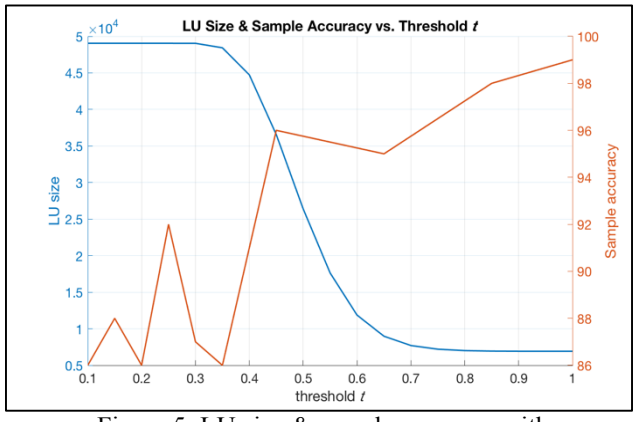


Figure 5: LU size & sample accuracy with different threshold *t*.

union is steady and will not expand too much since each seed in a frame conveys the similar meaning, resulting a union of many highly overlapping sets. We select t=0.45 as our final setting, which achieves a sample accuracy of 96% and yields 36k LUs. Table 2 shows the result of the expansion for the frame **Commerce_buy**.

LUs for frame **商業購買/Commerce_buy**	
CFN	購 (buy)/v; 購買 (purchase)/v; 購物 (shop)/v; 買(buy)/v; 租 (rent)/v; 租賃 (rent)/v; 租用 (lease)/v;
Our Resource	訂購 (order)/v; 參股 (share)/v; 買下 (buy in)/v; 收購 (buy in)/v; 買進 (buy in)/v; 買來 (buy in)/v; 入股 (share)/v; 買到 (buy)/v; 花錢買 (spend money on)/v; 買下來 (buy)/v; 買不到(unable to buy)/v; 支付 (transfer)/v; 交易 (transaction)/n; 並購 (merge)/v; 選購 (purchase)/v; 買不起 (cannot afford)/v; 買得起 (afford)/v; 採購 (purchase)/v; 購買者 (buyer)/n; 買過 (bought)/v; 買進賣出 (buy and sell)/v; 購入 (buy in)/v; 搶購 (rush to buy)/v; 購買 (purchase)/v; 買斷 (buyout)/v; 買賣 (trade)/v; 買家 (buyer)/n; 買 (buy)/v; 購進(buy in)/v; 買入(buy in)/v; 購得/v; 買個 (buy one)/v; 買些 (buy some)/v; 投資(invest)/v; 購置(purchase)/v;

Table 2: The expansion results for frame **Commerce_buy.**



Figure 6: Annotation interface that recommends candidate frames from a target word.

4. Application

In one subtask of our corpus annotation campaign, annotators are requested to provide Chinese FrameNet annotation on daily events from diaries and tweets. Annotators need to select a few target words (predicate of events) from a text first, and then annotate each target words with the correspondent FrameNet annotations. To annotate a frame, one must select which frame a target word could evoke, and realize its frame elements. Our constructed LUs are particularly useful for recommending frames for a target word (i.e, given a target word, list its possible frames by querying if the word could be found for some frames as LUs in our LU resource). Figure 6 depicts part of the annotation interface that recommends candidate frames given a target word. With low coverage LUs, the system often fails to find appropriate frame recommendations. In such case, annotators would need to spend considerable amount of time just to select a proper frame from a list of more than a thousand, which may severely slow the whole annotation process. By applying our resource to the annotation system, we could help alleviate the difficulty of selecting proper frame resulted from lack of LU resource.

We hired a few highly trained linguists to fulfill our corpus annotation. At the time of writing we have 36,029 target words marked as event predicates. 30,976 out of all target words have its FrameNet annotation. Table 3 shows the annotation statistics for different source types. We achieve overall 86% success rate to provide frame recommendations that are acceptable by annotators. The success rates in terms of source types are 95% and 80% for diaries and tweets respectively. Since our LU resource is constructed from UM-Corpus, which consists of contents that are relatively formal, the result also shows much higher success rate for diaries compared with tweets.

	# Texts	# Predicate	# Frames	%Success
Diaries	4,688	13,451	12,793	0.95
Tweets	26,818	22,578	18,183	0.80
Total	31,506	36,029	30,976	0.86

Table 3: Annotation statistics for different source types.

5. Conclusions and Future Work

This paper presents the development process and the current status of the construction for a large scale FrameNet resource in Chinese. Based on a bilingual corpus, we construct a more applicable resource that has lexical units with higher coverage that could help improve the annotation process in terms of efficiency. With more annotated dataset available in the future, we could develop a more robust automatic semantic role labeling tool for Chinese.

6. Acknowledgements

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST-105-2221-E-002 -154 -MY3, MOST-106-2923-E-002 -012 -MY3 and MOST-107-2634-F-002-011-, and Academia Sinica under grant AS-107-TP-A05.

7. References

Das, D., Schneider, N., Chen, D., & Smith, N. A. (2010, June). Probabilistic frame-semantic parsing. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics* (pp. 948-956). Association for Computational Linguistics.

Fillmore, C. J. and Baker, C. (2009). A frames approach to semantic analysis. In B. Heine & H. Narrog (Eds), *The Oxford Handbook of Linguistic Analysis*. Oxford: Oxford University Press, pp. 313–339.

Fillmore, C. J., Baker, C. F., and Sato, H. (2002). The FrameNet Database and Software Tools. In Proceedings of the 2002 International Conference on Language Resources and Evaluation (LREC 2002), pages 1157–1160. European Language Resource Association (ELRA).

Fillmore, C. J., (1982). Frame Semantics. In *Linguistics in the Morning Calm*. Seoul, South Korea: Hanshin Publishing Co., pp. 111–137.

Hermann, K. M., Das, D., Weston, J., and Ganchev, K. (2014). Semantic Frame Identification with Distributed Word Representations. In Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics (ACL 2014), pages 1448–1458, Baltimore, Maryland, USA. Association for Computational Linguistics (ACL).

Kim, J.-U., Hahm, Y., and Choi, K.-S. (2016). Korean FrameNet Expansion Based on Projection of Japanese FrameNet. In Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): System Demonstrations, pages 175–179, Osaka, Japan, December.

Kingsbury, P. and Palmer, M. (2002). From TreeBank to PropBank. In Proceedings of the 2002 International Conference on Language Resources and Evaluation (LREC 2002), pages 1989–1993. European Language Resource Association (ELRA).

Li, R., Wu, J., Wang, Z., and Chai, Q. (2015). Implicit Role Linking on Chinese Discourse: Exploiting Explicit Roles and Frame-to-Frame Relations. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL 2015), pages 1263–1271, Beijing, China, July. Association for Computational Linguistics (ACL).

Liu, Y. and Sun, M. (2015). Contrastive Unsupervised Word Alignment with Non-Local Features. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI 2015), pages 2295–2301. AAAI Press.

Och, F. J. and Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics,* 29(1):19–51.

Ohara, K. H., Fujii, S., Saito, H., Ishizaki, S., Ohori, T., and Suzuki, R. (2003). The Japanese Framenet Project: A Preliminary Report. In Proceedings of Pacific Association for Computational Linguistics, pages 249–254.

Park, J., Nam, S., Kim, Y., Hahm, Y., Hwang, D., and Choi, K.-S. (2014). Frame-Semantic Web: A Case Study for Korean. In Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume 1272, pages 257–260. CEUR-WS. org.

Tian, L., Wong, D. F., Chao, L. S., Quaresma, P., Oliveira, F., Li, S., Wang, Y., and Lu, Y. (2014). UM-Corpus: A Large English-Chinese Parallel Corpus for Statistical Machine Translation. In Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC), Reykjavik, Iceland.

Tonelli, S. and Pianta, E. (2008). Frame Information Transfer from English to Italian. In Proceedings of the 2008 International Conference on Language Resources and Evaluation (LREC 2008), pages 2252–2256. European Language Resource Association (ELRA).

You, L. and Liu, K. (2005). Building Chinese FrameNet Database. In Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE 2005).

EFLLex: A Graded Lexical Resource for Learners of English as a Foreign Language

Dürlich, L.[1,2], **François, T.**[2,3]
[1] FAU Erlangen-Nürnberg,
[2] CENTAL/IL&C, University of Louvain
[3] Chargé de recherche FNRS
luise.duerlich@fau.de, thomas.francois@uclouvain.be

Abstract

This paper introduces EFLLex, an innovative lexical resource that describes the use of 15,280 English words in pedagogical materials across the proficiency levels of the European Framework of Reference for Languages. The methodology adopted to produce the resource implies the selection of an efficient part-of-speech tagger, the use of a robust estimator for frequency computation and some manual post-editing work. The content of the resource is described and compared to other vocabulary lists (MRC and BNC) and to a reference pedagogical resource: the English Vocabulary Profile.

Keywords: CEFR-graded lexicon, English as a foreign language, vocabulary

1. Introduction

The strong relation between vocabulary knowledge and reading comprehension has been thoroughly researched and confirmed (Laufer, 1992). Second language acquisition (SLA) research has established that readers should know between 95% and 98% of the words in a text to adequately comprehend it (Laufer and Ravenhorst-Kalovski, 2010). Given that the larger the vocabulary, the better the comprehension of texts, foreign language curriculums should stress the need to teach new vocabulary items. A larger vocabulary indeed means being able to understand a larger scope of texts. However, analyses of large corpora indicate that, from kindergarten through college, native speakers encounter approximately 150,000 different words (Zeno et al., 1995). For learners, it is impossible to pick up all of them. Designers of foreign language curriculums, publishers of educational materials and textbooks, or even teachers are thus faced with the issue of identifying the most important words to teach at each stage of the learning process.

The most common answers to that challenge have been (1) to use frequency lists obtained from a large corpus of texts intended for a learner audience and (2) to rely on expert knowledge, such as teacher expertise or linguists' recommendations. Following the first approach, frequency lists have been derived from various well-known corpora of English. The first significant vocabulary list for English included 20,000 words and was laboriously produced by Thorndike (1921) without the help of any computer. It was later extended to 30,000 words by Thorndike and Lorge (1944). The first computational list was obtained by Kučera and Francis (1967) from the Brown corpus and has a large influence in education and psychology. More recently, other lists have been developed from larger corpora, such as the CELEX database (Baayen et al., 1993), the Zeno list (Zeno et al., 1995), the list based on the British National Corpus (BNC) (Leech et al., 2001), or SUBLEX (Brysbaert and New, 2009). The main shortcomings of such lists for L2 education are that (1) they represent the native distribution of words, which is not fully compatible with the distribution of words in books and textbooks intended for L2 learners; (2) they do not specify at which proficiency level a given word is supposed to be learned.

The second approach to set a learning path for vocabulary has relied on expert knowledge. The most popular resources of this kind are the Reference Level Descriptions (RLDs). They are based on the scale of the Common European Framework of Reference for languages (CEFR), published by the Council of Europe (2001), and ranging from A1 to C2. The CEFR describes the skills learners should develop at each of the six proficiency level of the scale. For instance, for reading skills, a A1 learner should understand very simple sentences and familiar words and names, while a B1 learner should understand texts consisting mainly of high frequency language as well as the description of events, feelings, wishes in personal letters. At C2, a learner is able to read virtually all forms of written language however abstract, structurally or linguistically complex. However, such descriptions remain elusive and the limitations of the CEFR for practical purposes have been stressed (North, 2005, 40).

RLDs aims at providing more detailed linguistic guidelines based on the CEFR descriptors for over 20 European languages. They consist in lists of words, multi-words, functions, or syntactic structures where each entry is connected to one level of the CEFR. Following the *Guide for the production of RLD* published by the Language Policy Division DG IV of the Council of Europe in Strasbourg, the RLDs have been elaborated on the basis of inventories of statistical frequencies, perusal of large bodies of texts, expert knowledge, and learners' productions (Marello, 2012, 328). The RLD for English has been developed within the English Vocabulary Profile project, or EVP (Capel, 2010; Capel, 2012), using a corpus-informed approach based on learner production from the *Cambridge Learner Corpus*. The methodology applied has the great advantage of being able to assign different difficulty levels to the different senses of a word. However, Alderson (2007) stressed that relying almost entirely on the *Cambridge Learner Corpus*,

Genres	A1		A2		B1		B2		C1		Total	
Ad	14	(544)	11	(943)	26	(2,339)	7	(1,027)	8	(969)	66	(5,822)
Dialogue	52	(2,643)	51	(3,671)	77	(7,438)	45	(6,904)	27	(3,997)	252	(24,653)
E-Mail	1	(15)	14	(859)	32	(2,886)	18	(2,695)	7	(478)	72	(6,933)
Informative	27	(1,921)	47	(5,711)	123	(22,271)	104	(24,821)	110	(26,542)	411	(81,266)
Mail	1	(68)	4	(180)	12	(1,175)	6	(765)	2	(300)	25	(2,488)
Narrative	51	(4,285)	62	(7,837)	24	(4,487)	27	(6,751)	11	(1,835)	175	(25,195)
Reader Text	51	(21,476)	139	(69,610)	32	(28,494)	192	(88,651)	135	(92,715)	549	(300,946)
Recipe	-	-	-	-	3	(214)	-	-	1	(84)	4	(298)
Sentences	59	(2,310)	37	(2,185)	47	(3,246)	17	(1,571)	20	(1,921)	180	(11,233)
Various	16	(927)	39	(3,245)	102	(10,340)	43	(7,401)	37	(5,431)	237	(27,344)
Total	275	(34,422)	407	(94,460)	478	(82,890)	459	(140,586)	358	(134,272)	1,971	(486,178)

Table 1: Text and word distributions throughout the corpus by level and type of texts.

which is a collection of performances on Cambridge examinations, may be an issue for generalization. Moreover, RLDs are not able to discriminate from all the words related to a given proficiency level, the most important ones to learn.

Since 2014, a third approach has been investigated within the CEFRLex project, which adopts an original view on lexicon learning. In contrast with the classic approach that models lexical knowledge in a nominal fashion (i.e. each word is assigned to a given level of proficiency, which implies that all learners from a given level should know all words from this level), the CEFRLex project assumes a continuous vision of lexical learning. Words are rather described in terms of a frequency distribution over the CEFR proficiency levels. This distribution is obtained from collections of pedagogical documents that are intended for L2 learners and labelled in accordance with the CEFR scale. We are therefore able to describe the usage of words in a more subtle way, for instance stressing the fact that a given word, usually considered to be learned at a specific level (e.g. B1), may already occur in texts intended for lower proficiency levels. Similarly, the availability of frequencies per level allows to rank words assigned to the same level of proficiency by frequency.

The CEFRLex project aims to bring together various research teams across Europe to produce vocabulary lists for several European languages. So far, lists for French (François et al., 2014) and Swedish (François et al., 2016) have been developed, and the current paper introduces the English version of the CEFRLex list, called EFLLex. The next section describes the methodology applied to create the English list and to estimate the frequencies from the labelled corpus. Section 3. then describes the content of the resource and compares it with similar resources.

2. Methodology

The development of EFLLex draws on the methodology used for the other lists of the CEFRLex project, but it also met some specific challenges. The general principle is the following: based on a labelled corpus, where each text has been assigned a CEFR level by human annotators (e.g. by the authors of the textbooks or of the simplified readers), the frequency of each word is estimated and normalized for each CEFR level, which results in the word distribution over the CEFR levels. In this section, we first describe the collection process of the labelled corpus; we then discuss the choice of the automatic tagger used, before describing the frequency estimation technique. The section concludes with some considerations about the manual correction of the resource.

2.1. Source corpus

We collected a corpus consisting of 1,971 texts extracted from 17 textbooks, 33 graded readers and 7 online materials designed for learners of English as a foreign language. These books come from popular publishers such as Cambridge University Press, Oxford University Press, and Exam English Ltd. All texts selected for the study were related to a reading comprehension task and were classified according to the CEFR levels by the publishers' teams. Very few of the available resources were actually assigned the level C2; our corpus thus only includes texts within the levels A1 to C1. As a whole, the corpus contains 486,178 words that are distributed across the levels as displayed in Table 1. This table also describes the distribution of texts over different types of texts (e.g. dialogue, informative, narrative, mail, etc.), highlighting the variety of texts that have been collected. It should be mentioned that more advanced levels (B2 and C1) account for the biggest subcorpora, as texts of these levels are more easily available and generally longer.

2.2. Tagging

As a next step, the corpus had to be lemmatized and part-of-speech (POS) tagged, for the two following reasons. First, in the CEFRLex project, the choice has been made to compute frequencies of lemmas observed in the corpus instead of frequencies of inflected forms. Counting inflected forms entails that words having numerous inflected forms, such as verbs, would have their overall probability split between their different forms. Consequently, compared to invariable words (such as adverbs, prepositions, conjunctions), they would seem less frequent in texts than they really are. Second, using tokens presupposes the assumption that learners are not able to relate inflected forms with their lemma. Such a view seems highly questionable for most of the French words that have regular inflected forms. The rationale to use POS tags is that it enables to disambiguate homographic forms with different part-of-speech tags.

The choice of the POS-tagger is also critical, as lemmatization or tagging errors produce unwanted effects on the data such as:

- entries with wrong part-of-speech tag (e.g. *much* is tagged as JJ (adjective) instead of RB (adverb) or *fatigue* is tagged as ":" instead of NN for noun);
- entries with an incorrect or at least questionable lemma (e.g. *she* instead of *her*, *1* instead of *first*);
- tags that can be correct, but are erroneous in the specific context of the word (e.g. to tag the word *quiet* in *the peace and quiet* as an adjective)[1].

We also wanted to use a system able to detect phrasal verbs and multi-word expressions, as they are of prime importance in the L2 learning process (Paquot and Granger, 2012).
Based on the previous constraints, the following five POS-taggers were compared to find the most suitable one for the task:

- the TreeTagger (Schmid, 1994),
- a HMM-tagger based on the TnT-tagger (Brants, 2000) provided by the FreeLing library for C++ (Carreras et al., 2004),
- the left-to-right model of the Stanford Log-linear POS Tagger (Toutanova et al., 2003),
- the tagger provided by the SVMTool (Giménez and Màrquez, 2004)
- and a tagger based on automatic feature extraction provided by the NLP4J module (Choi, 2016).

To compare the taggers, we assessed their performance on one hundred sentences randomly sampled from the corpus. Two human experts manually checked the annotation output of each tagger and assigned each tagged word one of the five following categories: **(0)**: correct lemma and POS-tag, **(1)**: correct lemma, but wrong POS-tag, **(2)**: wrong lemma, but correct POS-tag, **(3)**: wrong lemma and wrong POS-tag, **(4)**: segmentation error (e.g. *it-* or *'pages*), which usually leads to tagging or lemmatization errors. The agreement between the two human experts was calculated using Cohen's Kappa. Agreement scores varied from 0.627 for the SVMTool tagger to 0.378 for the Freeling tagger. Such levels of agreement are weak, so an additional step was included, in which both experts compared their annotations in order to agree on a reference annotation, on which the performance scores presented in table 2 were calculated.

Category	TreeTagger	HMM	Stanford	SVMTool	NLP4J
(0) Correct	92.74%	95.35%	95.34%	93.49%	**95.88%**
(1) POS	5.72%	3.15%	4.45%	4.67%	3.86%
(2) Lemma	0.14%	0.29%	0.07%	0.78%	0.06%
(3) Lemma + POS	0.84%	0.72%	0.14%	0.28%	0.0%
(4) Segmentation	0.56%	0.5%	0.0%	0.78%	0.19%

Table 2: Evaluation of the POS-Tagger results

[1]This type of error does not lead to the creation of a wrong entry, but mess up the frequency estimations, since the word occurrence will be assigned to the wrong entry.

We observed that none of the taggers correctly identified indefinite and demonstrative pronouns, nor some adverbs of time that could also be tagged as nouns, such as *today* or *tomorrow*. Furthermore, there were several issues when it comes to lemmatization, e.g. *me* was lemmatized as *I* by the Standford Tagger and the lemma *she* was assigned to *her*. Since the NLP4J Tagger tagged most of the words correctly, we decided to use this tagger on the whole corpus. Using the entire parsing output of NLP4J, that featured dependency labels and verb particles, it was also possible to reconstruct compounds and phrasal verbs in the corpus using specific rules.

2.3. Estimating lexical frequencies

Similarly to what was done for the other resources of the CEFRlex family, the normalized frequency per million words is obtained by first computing the raw frequencies by level (*RFL*) based on the corpus. In a second step, the *RFL* are weighted by a dispersion index (D), intended to counteract the effect of context-specific low frequency words being overused within a small number of texts. As noted by Francis and Kučera (1982), lower frequency words tend to be context-specific: they appear in a small number of texts, but sometimes with a unusually high frequency within those texts. This observation is of particular relevance when estimating counts from textbooks. Their content is guided by a set of competences and tasks related to various types of situations, which are defined only to a certain extent by the CEFR guidelines. Textbook designers therefore have quite a latitude to decide which topics will be included in their book. As a consequence, it is likely that the importance of some low frequency words, related to specific topics, will be overestimated using raw frequencies, especially when a topic generally encompasses several texts within the same lesson. To reduce this effect, we adapted the *RFL* using a dispersion index (D), adapted from Carroll et al. (1971):

$$D_{w,k} = \frac{\log\left(\sum p_i\right) - \frac{\sum p_i \log(p_i)}{\sum p_i}}{\log(I)} \quad (1)$$

In a corpus of K difficulty levels, each containing I text sources (e.g. textbooks), in order to obtain the dispersion index D of a word w for a level k, we rely on two pieces of information: (1) p_i the probability of the word w in the i^{th} source, which is computed as the frequency of w in the source i divided by s_i, the number of words in the source i (if $p_i = 0$, $p_i log(p_i)$ was considered as equal to 0); (2) I is the total number of sources.
Combining the dispersion index D and *RFL* as follows, we obtain U, the frequencies per million words:

$$U_{w,k} = \frac{1,000,000}{N_k}(RFL * D_{w,k} + (1 - D_{w,k}) * f_{min}) \quad (2)$$

where N_k is the number of words in level k and f_{min} is defined as:

$$f_{min} = \frac{1}{N_k} \sum_{i=1}^{I} s_i f_i \quad (3)$$

lemma	tag	A1	A2	B1	B2	C1	total
cat	NN	2.940	202.796	31.681	33.339	28.9847	65.019
empty	JJ	86.492	150.888	65.947	194.801	123.405	156.021
explore	VB	20.578	54.677	73.625	46.070	56.961	69.590
obviously	RB	0	11.034	2.589	68.463	36.665	30.689
tiresome	JJ	0	0	0	0.315	0.815	0.611
video	NN	2.467	0.556	34.825	23.802	13.248	18.431
write	VB	934.708	378.337	760.734	536.380	713.326	549.909
shopping centre	NN	0	5.040	2.589	0	0.815	1.946
sign up	VB	0	0.887	10.789	2.499	6.216	5.302

Table 3: Example of some entries in EFLLex.

Level	# entries	# new entries	Hapax	>10	EVP	FLELex	SVALex
A1	2,395	2,395	893	509	601	4,976	1,157
A2	4,205	2,478	1,633	1,000	925	3,516	2,432
B1	5,607	2,740	2,366	1,003	1,429	4,970	4,332
B2	8,228	3,935	3,580	1,571	1,711	1,653	4,553
C1	9,232	3,733	4,254	1,591	N/A	2,122	3,160

Table 4: Distribution of EFLLex entries per CEFR level, including the number of attested entries per level, number of new entries, number of hapax legomena, number of words occurring more than ten times. We also provide the number of new entries for English Vocabulary Profile (EVP) (Capel, 2010), the number of new items in FLELex (François et al., 2014) and in SVALex (François et al., 2016) for comparison.

with f_i, the frequency of a given word in a source i, and s_i, the number of word in i, for each source divided by N_k.

2.4. Manual correction

The automatic creation of the EFLLex list was followed by a phase of manual correction, where we compared our resource with the MRC machine usable dictionary (Wilson, 1988) to find any combination of lemma and POS unattested by the MRC dictionary. The resulting list of "new" pairs was then checked by hand in order to decide whether we were dealing with some kind of tagging or lemmatization error or with correct pairs that, for some reason, were not included in the MRC dictionary. Among the entries absent in the MRC list, we found some recent expressions - e.g. words reflecting new technologies and media such as the verb *to blog* (as well as the nouns *blog* and *blogger*), *internet* or *smartphone* - and - as was quite often the case - compound words (e.g. *food poisoning*) and phrasal verbs (e.g. the verb *to get up*) that the MRC list did not record. The majority of the ca. 6,600 words had to be checked manually. Most of these entries were correct words and only 1,104 were "real" errors that could either be traced back to errors in the raw corpus or to errors made by the tagger.

3. Description of the resource

EFLLex contains 15,280 entries. Examples of entries can be seen in Table 3. Each entry corresponds to the combination of a lemma and a part-of-speech: the frequency of the lemma at each level of the CEFR (C2 excepted) is described, as well as its total frequency in the corpus. From Table 3, it can be seen that EFLLex includes both multiword expressions such as *shopping center* or phrasal verbs such as *sign up*. As already mentioned above, multi-word expressions are of utmost importance for L2 teaching purposes (Bahns and Eldaw, 1993), but are absent from most of the English frequency lists that were introduced in Section 1.

The distribution of the entries across the levels A1 to C1 is displayed in Table 4. As can be observed, the total number of words (*# entries*) encountered at a given level increases from elementary to more advanced levels, as could have been expected. If we take into account the number of new entries per level (*# new entries*), there is also a moderate increase as the levels advance. This was expected, but if we compare these figures with those from FLELex and SVALex (columns 7 and 8), an interesting pattern emerges. Compared to SVALex, words are encountered faster, i.e. more words are introduced at the lower levels (A1 and A2), but this trend reverses at the intermediate levels (B1 and B2). In contrast, in FLELex, words are introduced faster at the elementary levels. As already mentioned in (François et al., 2016) we believe that this pattern results from the variation in corpus size. The results obtained for EFLLex confirm that the larger the corpus used to estimate the frequencies, the quicker new entries are introduced.

Besides comparisons with other resources from the CEFRLex project, it is also interesting to compare EFLLex with the EVP referential. As can be seen from Table 3, the EVP introduces far fewer words per level, which could be explained by two main reasons. First, the EVP aims to cover learners' productive vocabulary, which is known to be smaller than the receptive vocabulary covered by EFLLex. Second, EFLLex includes both words from the core vocabulary that should be taught at a given level, and peripheral words that can be encountered by learners, but should probably be learned at a later stage of the curricu-

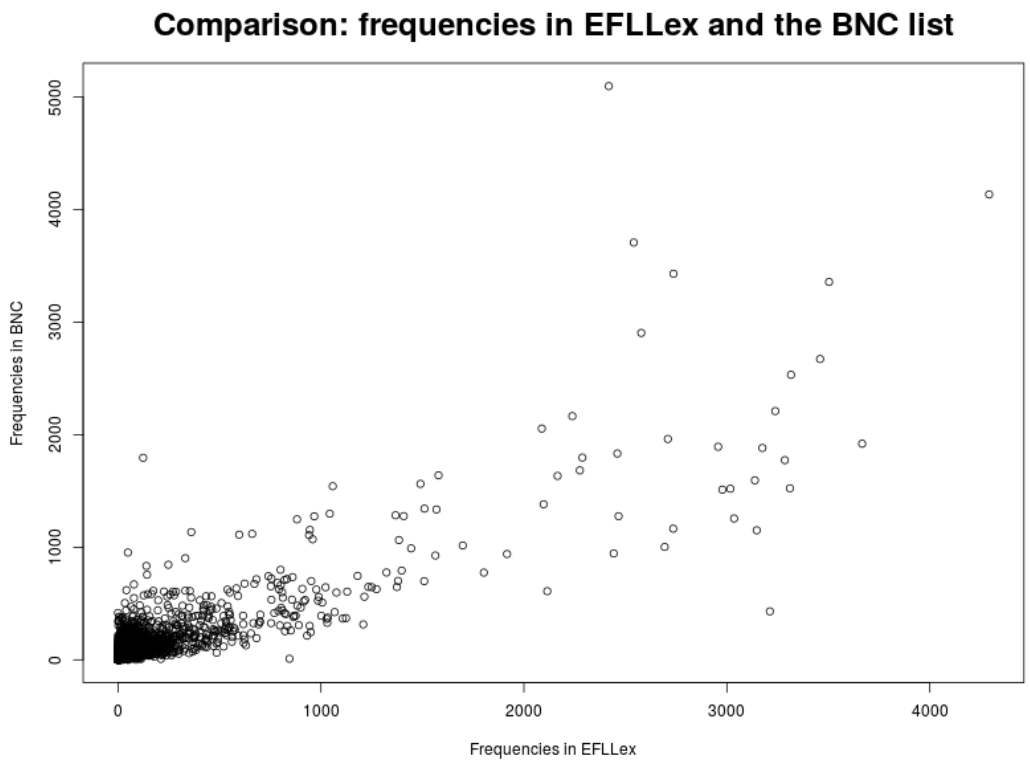


Figure 1: Association between the frequencies in the BNC list and EFLLex.

lum. In other words, EFLLex favours coverage over precision. EFLLex is therefore a useful tool to get a good idea of a word usage within EFL pedagogical materials, although further investigations are needed to interpret the frequencies in terms of teaching goals.

As regards frequency estimation, the limited size of the corpus has implications on the frequency distributions of entries, in a similar way to the other resources of the CEFRLex project. There is quite a significant number of words occurring at one level only, as revealed by the number of hapax legomena in the resource (column 4). Moreover, only 3,051 entries have a total frequency higher than 10 occurrences per million words, which is a clear limitation of the resource. Therefore, conclusions on the real usage of infrequent entries in EFLLex should be drawn with caution.

To better investigate this issue, the frequencies obtained on all levels (**total**) were compared with two popular frequency lists for English: (1) the BNC list and (2) the MRC machine usable dictionary. The BNC list consists of 678,211 entries for lemmas reporting frequencies per million words based on both written and spoken parts of the BNC (Leech et al., 2001). As the qualities of the BNC as a corpus are well acknowledged, it is reassuring to observe that the Pearson product-moment correlation coefficient between frequencies of EFLLex and the BNC list is very high ($r = 0.97; p < 0.0001$). Figure 1 shows this association for words having a frequency lower than 5,000 in EFLLex. For the counts from the London-Lund Corpus of English Conversation taken from the MRC list, the correlation is notably weaker ($r = 0.53; p < 0.0001$). Comparison with frequencies reported by the MRC list, however, is problematic in that frequencies have been added together over all possible POS-tags a word form can take. This means that the word *laugh*, for example, is assigned the same frequency, whether it be a verb or a noun. Thus, the MRC does not report reliable frequency information for specific token and tag combinations. Table 5 illustrates, with the word *well*, the consequences of such a choice when carrying out a correlation analysis such as ours. In conclusion of this experiment, the high correlation with the frequencies in the BNC list demonstrates that, even though the size of our corpus was limited, frequencies in EFLLex seem to offer a good estimate of word usage.

POS	EFLLex	BCN	MRC
well JJ (adjective)	5.9	42	1,753
well NN (noun)	1.7	14	1,753
well RB (adverb)	661.0	1,119	1,753
well UH (interjection)	787.4	-	1,753
well VB (verb)	-	2	1,753

Table 5: Frequencies of *well*, depending on its part-of-speech, in EFLLex, the BNC and the MRC lists.

4. Conclusion

In this paper, we have introduced EFLLex, a very original lexical resource for English that describes word usage in pedagogical materials over the CEFR levels. We have reported the methodological details of the development of

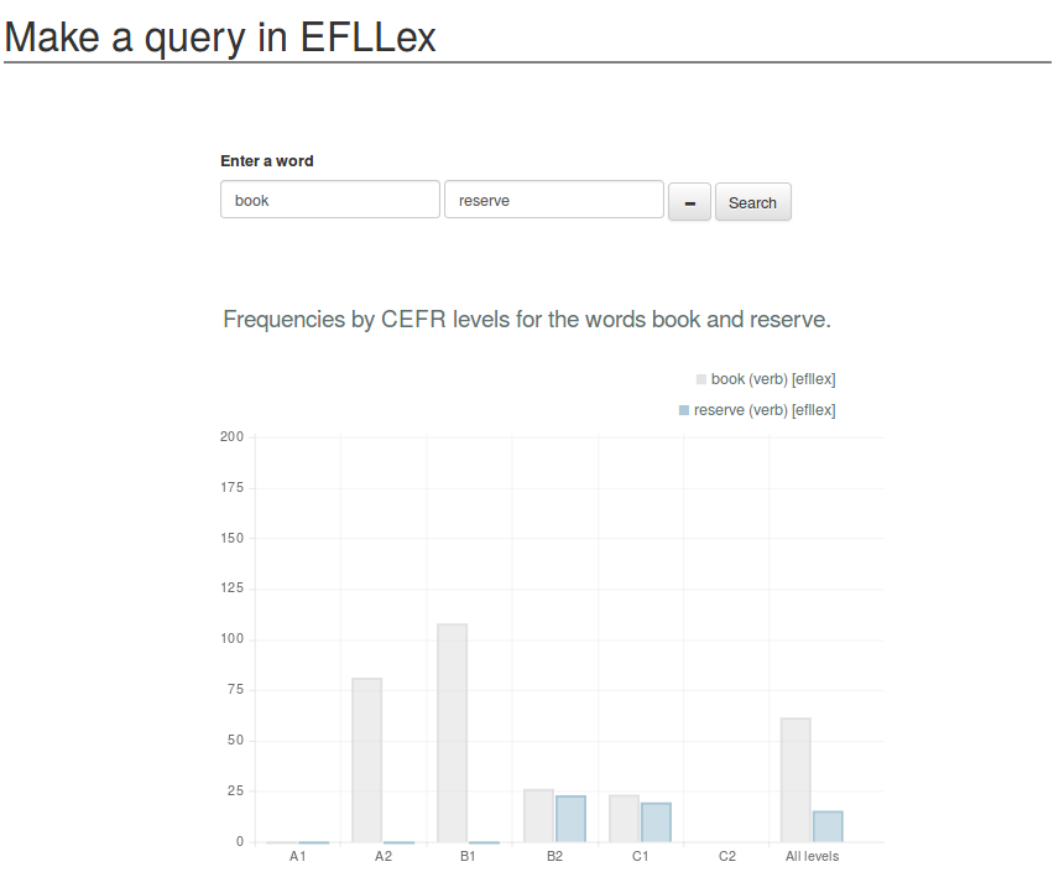


Figure 2: Screen capture of EFLLex website, showing the distributions of the verbs *book* and *reserve*.

this resource, which is part of a larger project, the CEFR-Lex project that aims to cover a wide range of European languages. The list is freely available for the scientific community[2] and can be queried through a web interface that displays the distribution of a word. The interface can also be used to compare the distributions of two words, as exemplified by the screen capture of this interface (see Figure 2). Finally, we plan to develop an additional tab for this interface that will allow to assess the vocabulary of a whole text: each word of the text that is included in EFLLex will be assigned one of the CEFR level based on its EFLLex distribution. Such tool draws from a similar system developed for French (Tack et al., 2016) and is an interesting alternative to the Text Inspector system[3] based on EVP.

In contrast with classic frequency lists, EFLLex is based on pedagogical texts intended for EFL learners and therefore represents a better image of the lexicon they encounter during their curriculum. It also offers a finer view of word use within a level: for instance, EFLLex makes obvious that *write* is a much more prevalent word at A1 (934 occurrences) than *explore* (20 occurrences). However, the resource also has some limitations as regards frequency estimation. Not only, the size of the corpus was limited, as there is a limited amount of pedagogical materials available, but it is also unbalanced in favour of more advanced levels. Still, as was shown by a comparison with the BNC list, the EFLLex frequencies are a robust estimation of the use of English words by learners. We believe EFLLex can provide interesting insights for language learning and will hopefully inspire more contributions to the domains of computer-assisted language learning and language teaching.

[2]http://cental.uclouvain.be/cefrlex/efllex/

[3]https://textinspector.com/

5. Acknowledgements

We would like to express our gratitude to Baptiste Degryse and Dorian Ricci for their invaluable help with the EFLLex interface. We also need to acknowledge the support of Alice Panier that helped us to collect more texts !

6. Bibliographical References

Alderson, J. (2007). The CEFR and the need for more research. *The Modern Language Journal*, 91(4):659–663.

Baayen, R., Piepenbrock, R., and van Rijn, H. (1993). The CELEX lexical data base on CD-ROM.

Bahns, J. and Eldaw, M. (1993). Should We Teach EFL Students Collocations? *System*, 21(1):101–14.

Brants, T. (2000). TnT – a statistical part-of-speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing*, pages 224–231.

Brysbaert, M. and New, B. (2009). Moving beyond kučera and francis: A critical evaluation of current word frequency norms and the introduction of a new and im-

proved word frequency measure for american english. *Behavior research methods*, 41(4):977–990.

Capel, A. (2010). A1-B2 vocabulary: Insights and issues arising from the English Profile Wordlists project. *English Profile Journal*, 1(1):1–11.

Capel, A. (2012). Completing the English Vocabulary Profile: C1 and C2 vocabulary. *English Profile Journal*, 3:1–14.

Carreras, X., Chao, I., Padró, L., and Padró, M. (2004). Freeling: An open-source suite of language analyzers. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*.

Carroll, J., Davies, P., and Richman, B. (1971). *The American Heritage word frequency book*. American Heritage Publishing CO., New York.

Choi, J. D. (2016). Dynamic feature induction: The last gist to the state-of-the-art. In *Proceedings of NAACL*.

Council of Europe. (2001). *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Press Syndicate of the University of Cambridge.

François, T., Gala, N., Watrin, P., and Fairon, C. (2014). FLELex: a graded lexical resource for French foreign learners. In *Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014)*.

François, T., Volodina, E., Ildikó, P., and Tack, A. (2016). SVALex: a CEFR-graded lexical resource for Swedish foreign and second language learners. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 213–219.

Francis, W. N. and Kučera, H. (1982). *Frequency Analysis of English Usage*. Houghton Mifflin.

Giménez, J. and Màrquez, L. (2004). SVMTool: A general POS tagger generator based on support vector machines. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal.

Kučera, H. and Francis, W. N. (1967). *Computational analysis of present-day American English*. Dartmouth Publishing Group.

Laufer, B. and Ravenhorst-Kalovski, G. (2010). Lexical threshold revisited: Lexical text coverage, learners' vocabulary size and reading comprehension. *Reading in a foreign language*, 22(1):15–30.

Laufer, B. (1992). How much lexis is necessary for reading comprehension? In *Vocabulary and applied linguistics*, pages 126–132. Springer.

Leech, G., Rayson, P., and Wilson, A. (2001). Word frequencies in written and spoken english: based on the british national corpus.

Marello, C. (2012). Word lists in Reference Level Descriptions of CEFR (Common European Framework of Reference for Languages). In *Proceedings of the XV Euralex International Congress*, pages 328–335.

North, B. (2005). The CEFR levels and descriptor scales. In *Multilingualism and assessment: Achieving transparency, assuring quality, sustaining diversity. Proceedings of the ALTE Berlin Conference*, pages 21–66.

Paquot, M. and Granger, S. (2012). Formulaic language in learner corpora. *Annual Review of Applied Linguistics*, 32:130–149.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing (NeMLaP)*, pages 44–49.

Tack, A., François, T., Ligozat, A.-L., and Fairon, C. (2016). Evaluating lexical simplification and vocabulary knowledge for learners of french: possibilities of using the flelex resource. In *Proceedings of the Tenth conference on International Language Resources and Evaluation (LREC'16)*, pages 230–236.

Thorndike, E. and Lorge, I. (1944). The teacher's wordbook of 30,000 words.

Thorndike, E. (1921). Word knowledge in the elementary school. *The Teachers College Record*, 22(4):334–370.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Wilson, M. (1988). MRC Psycholinguistic Database: Machine Usable Dictionary, Version 2.00. In *Behaviour Research Methods, Instruments & Computers*, volume 20, pages 6–10. Springer US.

Zeno, S., Ivens, S., Millard, R., and Duvvuri, R. (1995). *The educator's word frequency guide*. Touchstone Applied Science Associates, Inc., New York.

English-Basque Statistical and Neural Machine Translation

Inigo Jauregi Unanue[1,2], Lierni Garmendia Arratibel[3], Ehsan Zare Borzeshi[2], Massimo Piccardi[1]

[1]University of Technology Sydney (UTS), Sydney, Australia
[2]Capital Markets Cooperative Research Center (CMCRC), Sydney, Australia
[3]Professional Translator, Barcelona, Spain

{ijauregi,ezborzeshi}@cmcrc.com, liernigarmendia@gmail.com, massimo.piccardi@uts.edu.au

Abstract

Neural Machine Translation (NMT) has attracted increasing attention in the recent years. However, it tends to require very large training corpora which could prove problematic for languages with low resources. For this reason, Statistical Machine Translation (SMT) continues to be a popular approach for low-resource language pairs. In this work, we address English-Basque translation and compare the performance of three contemporary statistical and neural machine translation systems: OpenNMT, Moses SMT and Google Translate. For evaluation, we employ an open-domain and an IT-domain corpora from the WMT16 resources for machine translation. In addition, we release a small dataset (*Berriak*) of 500 highly-accurate English-Basque translations of complex sentences useful for a thorough testing of the translation systems.

Keywords: Neural Machine Translation, Statistical Machine Translation, English-Basque, Basque

1. Introduction

The advent of deep neural networks in natural language processing (NLP) has led to significant progress in a variety of classification tasks, including named-entity recognition (Lample et al., 2016), answer sentence selection (Lowe et al., 2017) and natural language inference (Wang et al., 2017). Deep neural networks have also started to obtain promising results in NLP, mainly in Neural Machine Translation (NMT) (Sutskever et al., 2014; Bahdanau et al., 2014; Vaswani et al., 2017; Wu et al., 2016). In NMT, the model receives a sentence in a *source* language as input and generates its translation word by word in a *target* language. NMT has outperformed previous translation systems in many language pairs (e.g., German-English, French-English).

However, in order to reach high accuracies, neural translation systems tend to require very large parallel training corpora (Koehn and Knowles, 2017). As a matter of fact, such corpora are not yet available for many language pairs. When the training data are relatively small, other, more traditional approaches such as Statistical Machine Translation (SMT) (Koehn et al., 2007) seem to be more accurate. Several ideas have been proposed in order to mollify this issue including multi-lingual systems with zero-shot translation (Johnson et al., 2016), transfer learning (Zoph et al., 2016) and back-translations (Sennrich et al., 2016). However, their general effectiveness still requires wider evaluation.

For this work, we have selected a low-resource language pair, English-Basque (abbreviation: en-eu), and used it as a case study for statistical and neural machine translation. Past machine translators for the Basque language have mainly used rule-based (Mayor et al., 2011) and statistical approaches (de Ilarraza et al., 2008; Del Gaudio et al., 2016; Stroppa et al., 2006). In our work, we compare three different systems: OpenNMT, an open-source NMT system; Moses SMT, an open-source SMT system; and Google Translate, a publicly-available commercial system which uses either SMT or NMT models depending on the language pair. The first two have been trained by us with dedicated datasets, while Google has just been used "as is" from its API. The three models have been tested over open-domain and Information Technology (IT) domain datasets from the WMT2016 IT helpdesk shared task. Moreover, we release a new, small en-eu corpus (named *Berriak*) useful for probing English-Basque machine translation. This corpus consists of 500 long and complex sentences translated from English to Basque by experienced translators and is much more realistic and challenging than the existing en-eu corpora. Due its small size, we have only used it as a test set.

The rest of this paper is organised as follows: Section 2. discusses the main characteristics of the Basque language. Section 3. describes the compared systems. Section 4. describes the datasets used in the experiments and presents the results. Section 5. concludes this paper.

2. The Basque Language

Basque (*Euskara*) is a language spoken in the Basque Country (northern Spain and southwestern France). It is not considered a member of the Indo-European language family and it remains isolated to date, meaning that researchers have not found any other language with similar characteristics. As stated by (Mayor et al., 2011), Basque is an *agglutinative* language with rich inflectional morphology. This means that a word may include several morphemes that change its inflectional category such as number, case, tense or person. Unlike in fusional languages such as Spanish and French, in agglutinative languages the boundaries between morphemes remain clear-cut (Aikhenvald, 2007). For agglutinative languages, rule-based systems have proved an effective translation approach in the past (Koehn and Monz, 2006).

During the military dictatorship of Spain (1939-1975), Basque became illegal and the number of speakers dropped drastically. However, in the 1970s a process for the formal standardization of the language (*Euskara Batua*) began and Basque teaching in schools was restored. Nowadays, the language is official in the Basque autonomous community in Spain and it has reached approximately 1M speakers (including Navarre and the French side). Yet, the persisting lack of available translation corpora makes the development of Basque machine translators a difficult task.

3. Methods

3.1. Moses SMT

SMT has been the state-of-the-art approach to machine translation for many years (Koehn et al., 2003). SMT systems are usually phrase-based system which first try to learn the alignments of phrases between different languages, and then predict the best composition of phrases for the translation with the help of a target language model (LM).

In this work, we have evaluated a popular open-source SMT toolkit called Moses (Koehn et al., 2007). First, a word alignment model between the source and target languages has been learned over the training data with the GIZA++ toolkit (Och and Ney, 2003). Then, an LM has been learned over the training data with KenLM (Heafield, 2011). Finally, based on these two models, the Moses decoder has been used to translate the sentences.

3.2. Google Translate

Google Translate is a publicly-available, well-known commercial machine translator. Recently, it has implemented the Google Neural Machine Translation (GNMT) (Johnson et al., 2016) over many language pairs and en-eu is one of them[1]. Google Translate does not train from a dedicated annotation of parallel text; rather, it crawls the Web to forage for "likely parallel" paragraphs (for instance, those marked with multiple HTML lang tags). In our work, we have used it from the convenient Google Cloud Translation API[2].

3.3. OpenNMT

We have trained an NMT model by using the OpenNMT toolkit (Klein et al., 2017) with the *seq2seq* architecture of (Sutskever et al., 2014). This architecture is formed by an encoder, which converts the source sentence into a sequence of numerical vectors, and a decoder, which predicts the target sentence based on the encoded source sentence. Both the encoder and the decoder are usually recurrent neural networks (RNNs). Additionally, an *attention mechanism* (Bahdanau et al., 2014; Luong et al., 2015) has been used to learn soft-alignments between the source and the target sentences.

In our model, we have used the Long Short-Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) for both the encoder and the decoder. We have also used the *dot* attention mechanism (Luong et al., 2015) where weights over the encoded source sentence are provided by an auxiliary network. Like with any other neural models for NLP, prior to processing each unique word in the corpus needs to be mapped to a high-dimensional vector (*word embedding*). This mapping can be either random (the default) or based on user-provided pre-trained embeddings. In addition, the word embeddings can be kept constant during training, or updated alongside all other parameters to minimise the cost function. Since pre-trained embeddings have typically reported higher accuracies (Dernoncourt et al., 2017; Lample et al., 2016), we have trained Basque word embeddings using GloVe (Pennington et al., 2014) over the Basque Wikipedia. For English, we have used the available CommonCrawl pre-trained embeddings[3]. We have evaluated the use of these embeddings in two different ways: maintaining them fixed during both training and testing (f-emb) and updating them during the training stage (u-emb). The word embeddings have a dimension of 300. The remaining parameters of the network have been kept to their default values.

Data	*PaCo2_EnEu*	*WMT16_IT*	*Berriak*
training	125,356	89,983	—
test	5,000	1,000	500

Table 1: The number of samples in the *PaCo_EnEu*, *WMT16_IT* and *Berriak* datasets.

4. Experiments

4.1. Corpora

As mentioned previously, the en-eu language pair is considered to be low-resourced. To mitigate this issue, the WMT16 machine translation for IT domain shared task[4] provided a parallel en-eu corpus. This corpus includes both an IT-domain dataset and an open-domain dataset (*PaCo2_EnEu*). *PaCo2_EnEu* consists of approximately 130,000 en-eu translations crawled from the web (San Vicente et al., 2012). In the experiments, we have used 5,000 as a test set and the rest for training the models. We have also used the IT-domain data to evaluate the translators over a specialised domain. The IT-domain training set consists of 89,983 samples, but only 2,000 of them are proper sentences; the rest are translations of IT terms from Wikipedia and localization PO files. Consequently, the amount of "good quality" data in the training set is very limited. At its turn, the test set consists of 1,000 proper sentences.

By inspecting these resources, we had realised the lack of long and complex sentences, likely a major limitation for the realistic evaluation of this language pair. Such sentences appear in most professional translations and they are expected to prove far more challenging for automated translators. To provide a resource contribution, we have therefore collected and released a small, high-quality en-eu corpus called *Berriak* (*news* in Basque)[5,6]. To create a suitable corpus, we have randomly selected English sentences

[1] https://cloud.google.com/translate/docs/languages

[2] Google Translate API: https://cloud.google.com/translate/docs/

[3] GloVe: https://nlp.stanford.edu/projects/glove/

[4] WMT16:http://www.statmt.org/wmt16/it-translation-task.html

[5] ISLRN: 197-383-395-000-1

[6] https://github.com/ijauregiCMCRC/english_basque_MT

Model	***PaCo2_EnEu***	***Berriak***
en→eu		
Moses SMT	21.02	5.90
OpenNMT(r-emb)	20.07	1.49
OpenNMT(f-emb)	19.39	1.43
OpenNMT(u-emb)	21.18	1.84
Google Translate	9.12	9.91
eu→en		
Moses SMT	24.20	8.53
OpenNMT(r-emb)	19.44	3.35
OpenNMT(f-emb)	18.61	4.28
OpenNMT(u-emb)	22.42	5.53
Google Translate	16.66	20.80

Table 2: BLEU score of the models over the *PaCo_EnEu* and *Berriak* corpora.

Model	***WMT16_IT***
en→eu	
Moses SMT	11.74
Moses SMT+(PaCo train)	11.89
OpenNMT(r-emb)	11.87
OpenNMT(PaCo train)(r-emb)	12.42
OpenNMT(u-emb)	12.75
OpenNMT(PaCo train)(u-emb)	12.31
Google Translate	14.46
eu→en	
Moses SMT	19.06
Moses SMT+(PaCo train)	19.34
OpenNMT(r-emb)	15.46
OpenNMT(PaCo train)(r-emb)	16.93
OpenNMT(u-emb)	17.30
OpenNMT(PaCo train)(u-emb)	18.01
Google Translate	24.66

Table 3: BLEU score of the models over the *WMT16_IT* corpus.

from the English-German news corpus of WMT16 which meets the requirements. The sentences have been translated into Basque with the help of Librezale[7], an open group of highly-qualified volunteers who work to increase the use of the Basque language in the IT domain. To date, we have collected 500 en-eu translations for a total of 10,280 tokens. Due to the small size of this corpus, in this work we have only used it as a further test set for models trained with *PaCo2_EnEu*. The manual translations are still ongoing and we intend to release an extended version in the near future. Table 1 summarises the number of samples of the various datasets.

4.2. Experimental Settings and Results

We have conducted a number of experiments to evaluate the models in a variety of scenarios. In the first experiment, we have trained the SMT and NMT models with the *PaCo2_EnEu* training set and tested them with both the *PaCo2_EnEu* test set and *Berriak*. Google Translate has been used as is. Experiments have been conducted in both English-to-Basque (en→eu) and Basque-to-English (eu→en) to assess performance in both directions. In addition, for the NMT model we have experimented with updated random embeddings (r-emb), fixed pre-trained embeddings (f-emb), and updated pre-trained embeddings (u-emb). Table 2 reports the BLEU scores (Papineni et al., 2002) for the three models. The first remark is that all models generally perform worse with Basque as the target language. This suggests that its intrinsic difficulty is higher than English. As for the models' comparison, both Moses SMT and OpenNMT have remarkably outperformed Google Translate on the *PaCo2_EnEu* test set. The NMT model has achieved the highest BLEU score (21.18) in the en→eu direction, while the SMT model has achieved the highest BLEU score (24.20) in the opposite direction. For the NMT model, updating the pre-trained embeddings during training (u-emb) has invariably led to the highest accuracies, up to an improvement of 2.98 BLEU points over the random embeddings in the eu→en direction.

However, the performance ranking has changed drastically when testing on the more probing *Berriak* corpus. In this case, Google Translate has achieved the highest BLEU scores by a large extent. We believe that both Moses SMT and OpenNMT, which have been trained using only the *PaCo2_EnEu* training set, have obtained such low results because the training corpus does not contain the same kind of long sentences as *Berriak* and therefore the models could not learn to translate such challenging sentences. Between SMT and NMT, the former has clearly outperformed the latter, confirming that SMT generalises better when the training corpus is limited. On the other hand, the training corpus of Google Translate is certainly much bigger, and that has helped it achieve better results on *Berriak*. However, the BLEU score when Basque is the target is still very low (9.91) and significant improvements are an outstanding need. For what concerns NMT and word embedding, also in this case the updated pre-trained embeddings have led to an improvement (although slight) in score.

In a second experiment in the IT domain (Table 3), Google Translate has again obtained the best results. This can be explained with the fact that Moses SMT and OpenNMT have only been trained with 2,000 proper IT-domain sentences. Between these two models, OpenNMT has outperformed Moses SMT for Basque as the target language, and vice versa for English. To mollify the small training size issue, we have added the open-domain corpus to the training data (noted as *PaCo train* in Table 3). The results have slightly improved for both NMT and SMT, with a more noticeable improvement for NMT (12.42 in en→eu and 16.93 in eu→en). Larger relative improvements have been achieved with the use of the pre-trained embeddings (12.75 in en→eu and 17.30 in eu→en). Since the updated embeddings had proved clearly more accurate in the previous experiment, we have not used the fixed embeddings in this experiment. Finally, using both the open-domain data and the pre-trained embeddings has only improved the scores for English as the target language. Once again, all

[7]Librezale: https://librezale.eus/

English sentence	1. *How many people take part in The Sanfermin Bullrunnings - Sanfermin.com - Pamplona*
	2. *Search results as from 07/02/2011 in "Zarzuela"*
	3. *For this reason, after Madrid and Sydney, they plan to continue this international anti-bullfighting campaign in Croatia and Berlin.*
Ground Truth	1. *Sanferminetako entzierroa zenbat jendek egiten duen - Sanfermin.com - Pamplona*
	2. *Bilaketa emaitzak 2011/07/02 egunetik aurrera "Zarzuela"-(e)n*
	3. *Horregatik, zezenketen eta entzierroen aurkako nazioarteko protesta Madrilen eta Sidneyn egiteaz gain, Kroazia eta Berlinen ere izango da.*
Moses SMT	1. *Zenbat jende parte hartzeko Sanferminetako Bullrunnings - Sanfermin.com - Pamplona*
	2. *Bilaketa emaitzak 2011/07/02 egunetik aurrera "Zarzuela"-(e)*
	3. Hori dela eta, ondoren, Madrilera eta Sydney jarraitzen dute, plan horrek nazioarteko aurkako Zezenketetako @-@ kanpaina batean Kroazia eta Berlingo.
OpenNMT	1. *Nola bizi da Sanfermin Bullrunnings - Sanfermin.com - Pamplona*
	2. *Bilaketa emaitzak 2011/07/02 egunetik aurrera "Zarzuela"-(e)n*
	3. *Hori dela eta, Madrilen, Madrilen, Madrid, bullfighting eta Berlin, international eta Berlin.*
Google	1. *Zenbat pertsona parte hartu Sanferminetako entzierroetan - Sanfermin.com - Iruñean*
	2. *Bilaketaren emaitzak 2011/02/07 "Zarzuela" -en*
	3. *Horregatik, Madrilen eta Sydneyen ondoren, Kroazia eta Berlinen kontrako zezenketarako nazioarteko kanpaina aurrera eramateko asmoa dute.*

Table 4: Example of translations over the *PaCo2_EnEU* (en→eu) test set.

the models have performed significantly better with English as the target language, with an even bigger margin compared to the general-domain experiment. We speculate that this may be due to the fact that in the IT domain the Basque language does not have a vocabulary as comprehensive and developed as English does. In fact, many IT words and expressions are taken from English unchanged.

For a qualitative analysis, Table 4 shows three examples of translations provided by the different models alongside the ground truth from the *PaCo2_EnEu* test set, which is the dataset on which the trained NMT and SMT models have obtained the best accuracies. We can see that for sentence 2 the OpenNMT model has provided a translation identical to the ground truth, probably thanks to the fact that there are sentences with the same structure in the training corpus. However, the NMT model tends to directly bypass many words from the original English sentence into the prediction (see sentences 1 and 3; NB: OpenNMT allows the model to bypass words from the source sentence). More precisely, if the model is uncertain about which word from the target vocabulary should be predicted next, it will pass on the word with highest attention weight from the source sentence. This mechanism aims to help the prediction of words such as proper names, which are not likely to appear in the target vocabulary. As additional analisys, we have computed the percentage of words bypassed by the different NMT models. To this aim, we have counted the number of words in the test set's predictions which did not belong to the target vocabulary, and divided it by the total number of words predicted. Table 5 shows the computed percentages, averaged over all the different NMT models. We can observe that the numbers are clearly higher when Basque is the target language, which matches our intuition that Basque is more difficult to translate into. When comparing the different datasets, we observe a trend to bypass more words in *Berriak*, which is understandable as this dataset does not have a training corpus and has longer and more complex sentences. Conversely, *WMT16_IT* has the lowest percentages of source words in the predictions. This is likely due to the fact that this dataset is very domain-specific and has a smaller vocabulary size.

On a separate note, NMT tends to predict the same word repeatedly (see sentence 3), as often been reported for neural encoder-decoder architectures. On the other hand, the SMT model seems able to match more words correctly in each sentence, but it has difficulties to form grammatically-complete sentences (see all three examples). Finally, the sentences predicted by Google Translate contain synonyms of the words in the ground truth (e.g., *Iruñean* vs *Pamplona*) and errors in the inflectional morphemes (e.g., *entzierroa* vs *entzierroetan*, *Bilaketa* vs *Bilaketaren*, *zezenketen* vs *zezenketarako*).

Corpus	Bypassed (%)
en→eu	
PaCo2_EnEu	21.70
Berriak	36.60
WMT16_IT	3.78
eu→en	
PaCo2_EnEu	4.29
Berriak	7.59
WMT16_IT	1.07

Table 5: Average of the percentages of bypassed words by all the NMT models in each dataset and each direction.

5. Conclusion

This paper has presented a performance comparison of three contemporary MT approaches on a low-resourced language pair, English-Basque. The compared approaches include an NMT model (OpenNMT, with and LSTM encoder/decoder), an SMT model (Moses) and the popular Google Translate service.

The experimental results show that all the models have achieved worse results when Basque is the target language, confirming that languages with rich morphology are more difficult to translate into. The NMT and SMT models have outperformed Google Translate when using training and test data from the same corpus (*PaCo2_EnEu*). However, these models have not generalised well on long and complex sentences, in contrast to Google Translate. For the NMT model, initialising the word embeddings with pretrained embeddings (based on the Basque Wikipedia for Basque and CommonCrawl for English) and updating them during training has invariably led to the best BLEU scores. In absolute terms, the achieved BLEU scores suggest that machine translation for Basque still has large margins for improvement. As part of this research, we have released a new, small corpus (named *Berriak*) of highly accurate en-eu sentences translated by experienced human translators to be used as a probing test set for this language pair. In the future, we plan to enlarge the *Berriak* corpus for more extensive testing and explore ways to improve the accuracy of the translations without resorting to larger parallel datasets such as pivot languages and multi-lingual translators.

6. Acknowledgements

We would like to thank the Librezale association for the Basque translations for the new *Berriak* corpus. We would also like to acknowledge the financial support received from the Capital Markets Cooperative Research Centre (CMCRC), an applied research initiative of the Australian Government.

7. Bibliographical References

Aikhenvald, A. Y. (2007). Typological distinctions in word-formation. *Language typology and syntactic description*, 3:1–65.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

de Ilarraza, A. D., Labaka, G., and Sarasola, K. (2008). Statistical postediting: A valuable method in domain adaptation of rbmt systems for less-resourced languages. *Mixing Approaches to Machine Translation*, pages 35–40.

Del Gaudio, R., Labaka, G., Agirre, E., Osenova, P., Simov, K. I., Popel, M., Oele, D., van Noord, G., Gomes, L., Rodrigues, J. a. A., et al. (2016). Smt and hybrid systems of the qtleap project in the wmt16 it-task. In *WMT*, pages 435–441.

Dernoncourt, F., Lee, J. Y., Uzuner, O., and Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al. (2016). Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). Opennmt: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the Workshop on Neural Machine Translation*. Association for Computational Linguistics.

Koehn, P. and Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 102–121. Association for Computational Linguistics.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June. Association for Computational Linguistics.

Lowe, R. T., Pow, N., Serban, I. V., Charlin, L., Liu, C.-W., and Pineau, J. (2017). Training end-to-end dialogue systems with the ubuntu dialogue corpus. *Dialogue & Discourse*, 8(1):31–65.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Mayor, A., Alegria, I., De Ilarraza, A. D., Labaka, G., Lersundi, M., and Sarasola, K. (2011). Matxin, an open-source rule-based machine translation system for basque. *Machine translation*, 25(1):53.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

San Vicente, I., Manterola, I., et al. (2012). Paco2: A fully automated tool for gathering parallel corpora from the web. In *Language Resources and Evaluation (LREC)*, pages 1–6.

Sennrich, R., Haddow, B., and Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Association for Computational Linguistics*.

Stroppa, N., Groves, D., Way, A., and Sarasola, K. (2006). Example-based machine translation of the basque language.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wang, Z., Hamza, W., and Florian, R. (2017). Bilateral multi-perspective matching for natural language sentences. *arXiv preprint arXiv:1702.03814*.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*.

TQ-AutoTest – A Semi-Automatic Test Suite for (Machine) Translation Quality

Vivien Macketanz, Renlong Ai, Aljoscha Burchardt and Hans Uszkoreit
DFKI – Language Technology Lab
Berlin, Germany
{firstname.lastname}@dfki.de

Abstract

In several areas of NLP evaluation, test suites have been used to analyze the strengths and weaknesses of systems. Today, Machine Translation (MT) quality is usually assessed by shallow automatic comparisons of MT outputs with reference corpora resulting in a number. Especially the trend towards neural MT has renewed peoples' interest in better and more analytical diagnostic methods for MT quality. In this paper we present TQ-AutoTest, a novel framework that supports a linguistic evaluation of (machine) translations using test suites. Our current test suites comprise about 5000 handcrafted test items for the language pair German–English. The framework supports the creation of tests and the semi-automatic evaluation of the MT results using regular expressions. The expressions help to classify the results as correct, incorrect or as requiring a manual check. The approach can easily be extended to other NLP tasks where test suites can be used such as evaluating (one-shot) dialogue systems.

Keywords: Machine Translation, Quality Evaluation, Test Suites

1. Introduction and Background

In several areas of NLP evaluation, test suites have been used to analyze the strengths and weaknesses of systems. In contrast to "real-life" gold standard corpora, test suites can contain made-up or edited input-output pairs to isolate interesting or difficult phenomena.

In Machine Translation (MT) research, broadly-defined test suites have not been used apart from several singular attempts (King and Falkedal, 1990; Isahara, 1995; Koh et al., 2001, etc.). One of the reasons for this might be the fear that the performance of statistical MT systems depends so much on the particular input data, parameter settings, etc., that relevant conclusions about the errors they make are difficult to obtain. Another concern is that "correct" MT output cannot be specified in the same way as the output of other language processing tasks like parsing or fact extraction where the expected results can be more or less clearly defined. Due to the variation of language, ambiguity, etc., checking and evaluating MT output can be almost as difficult as the translation itself.

Today, MT quality is still usually assessed by shallow automatic comparisons of MT outputs with reference corpora resulting in a number. Early attempts to automatically classify errors based on post-edits or reference translations like (Popović and Ney, 2011) have not yet become standard. For the detection of certain types of errors like grammar errors, parsers have been used (Tezcan et al., 2016). In other narrow domains, researchers have started to explore the differences between systems and between the development stages of one system in more linguistic detail. Especially the trend towards neural MT has renewed peoples' interest in better and more analytical diagnostic methods for MT quality. Recent work based on specific test suites includes the study of verb-particle constructions (Schottmüller and Nivre, 2014), pronouns (Guillou and Hardmeier, 2016) or structural divergences (Isabelle et al., 2017). (Bentivogli et al., 2016) performed a comparison of neural- with phrase-based MT systems on IWSLT data using a coarse-grained error typology where neural systems have been found to make fewer morphological, lexical and word-order errors. Using our own test suites, we have performed several comparative studies of different MT systems both in the general domain (Burchardt et al., 2017) and in the technical domain (Beyer et al., 2017). When presenting this work, one of the most (obvious) criticism we got was the huge amount of manual effort that was involved in the evaluation procedure. In this paper we will present the novel TQ-AutoTest framework that allows for a drastic reduction of the manual effort when checking translation quality on the basis of test suites.

This article is structured as follows: In Section 2. we will briefly introduce our own test suite and the manual evaluation procedure we have applied in the past. Section 3. describes the new TQ-AutoTest framework that supports the evaluation procedure. A use case of the TQ-AutoTest will be shown in Section 4.. Finally, in Section 5. we will conclude and give an outlook on future work.

2. Test Suites for German – English

We have built a test suite for a fine-grained evaluation of MT quality for the language pair German – English. In brief, it contains segments selected from various parallel corpora and drawn from other sources such as grammatical resources, e.g., the TSNLP Grammar Test Suite (Lehmann et al., 1996) and online lists of typical translation errors.

Each test sentence is annotated with a phenomenon category and the phenomenon it represents. An example showing these fields can be seen in Table 1 with the first column containing the source segment and the second and third column containing the phenomenon category and the phenomenon, respectively. The fourth column shows an example machine translation[1] and the last column contains a

[1] As example we have used the "old" Google Translate system that was used before Google changed to a neural system in September 2016, cf. `https://research.googleblog.com/2016/09/a-neural-network-for-machine.html`.

post-edit of the MT output that is created by making as few changes as possible.
In our latest version of the test suite, we have a collection of about 5,000 segments per language direction that are classified in about 15 categories (most of them similar in both language directions) and about 120 phenomena (many of them similar but also some differing, as they are language-specific). Each phenomenon is represented by at least 20 test segments in order to guarantee a balanced test set. The categories cover a wide range of different grammatical aspects that might or might not lead to translation difficulties for a MT system.

2.1. Manual Evaluation Procedure

In order to evaluate a system's performance on the categories in the test suite, we concentrate solely on the phenomenon in the respective sentence and disregard other errors. This means that we have to determine whether a translation error can be linked to the phenomenon under examination or if it is independent from the phenomenon. If the former is the case, the segment will be validated as incorrect. If, however, the error in the translation cannot be traced back to the phenomenon, the segment will be counted as correct.
When conducting the manual evaluation, the system outputs were automatically being compared to a "reference translation", which is, in fact, the post-edit of the Google Translate output, as those were the very first translations to be generated and evaluated when we started building the test suite. In a second step, all the translations that did not match the "reference" were manually evaluated by a professional linguist since the translations might be very different from the Google post-edit but nevertheless correct. This is also the reason why we refrained from creating an independent reference. As a consequence, we cannot compute automatic scores like BLEU. However, we do not see this as a disadvantage as with the test suite we want to focus rather on gaining insights about the nature of translations than on how well translations match a certain reference.
Nevertheless, this manual evaluation is a very time-consuming process, especially when dealing with such a large dataset and different MT systems, thus, we decided to come up with a semi-automatic solution, i.e., the TQ-AutoTest.

3. The TQ-AutoTest Framework

With the test suite growing bigger over time, we decided to implement a framework that facilitates the evaluation procedure by automating the analysis. Therefore, we built the TQ-AutoTest. In order to include as many correct translations options as possible, the TQ-AutoTest is based on regular expressions (cf. Section 3.1.). Currently, the automation is almost fully completed for the language direction German→English and we are working on expanding and completing the other language direction.
Presently, the TQ-AutoTest exhibits the following features (described in further detail in the following Sections): data preparation; upload report; view report; compare engines; regular expression evaluation; expand, edit and query database (cf. Figures 2 and 3).
With these functions, the TQ-AutoTest can be used for different purposes: You can not only test a system's performance with regard to the linguistic phenomena but also compare the performance of different systems/system types or track changes within one system's performance. By doing so, you can test the system(s) either on all phenomena, or just a selection of the phenomena. To prevent overfitting or cheating, we will not publish the test items. Before sending them to colleagues who want their engines tested, we use a mechanism for scrambling the test segments with a large amount of "distractor" segments.

3.1. Regular Expressions

The foundation of the evaluation with the TQ-AutoTest are regular expressions. With the help of these patterns, we try to cover as many correct translations as possible. In line with our manual evaluation procedure briefly described in Section 2.1., the regular expressions only focus on the part of the segment that is under investigation, i.e., the respective phenomenon. Since all other mistranslations that cannot be related to the phenomenon are ignored, it is not necessary for the regular expressions to cover the whole sentence.
The process of creating the regular expressions was thus very complex and elaborate. They have been built manually by a linguist, supported by a professional translator. The regular expressions are based on MT outputs that had been generated before and were then expanded by experience, e.g., which correct/incorrect translation could be expected for a source segment. Considering that once the corpus is completed it can be used over and over again, we are convinced it is worthwhile investing the time and effort to create the regular expressions.
We did not only create positive regular expressions with which the MT output can be evaluated as correct, but in some cases also negative regular expressions with which the MT output is evaluated as incorrect:

Example (1)	
Source:	Sie fuhr das Auto ihres *Mannes*.
Output 1:	She drove her *husband*'s car.
Output 2:	She drove the *man*'s car.
Output 3:	She drove the *blue* car.
positive regex:	husband|spouse|hubb(y|ies)
negative regex:	(gentle)?m[ae]n|guy

The German source sentence in example (1) contains a lexical ambiguity: The German word *Mann* can either mean *man* or *husband*. In combination with a possessive pronoun (in this case *ihr - her*), *Mann* always refers to *husband*. Output 1 - 3 are examples of different MT outputs. As can be seen, only output 1 matches the positive regular expression. The regular expression also allows translations that include the words *spouse*, *hubby* or *hubbies*[3].

[3]We include the plural of *hubby* as well since the focus in this category lies entirely on the lexical ambiguity of the German word *Mann* and, thus, a translation containing *hubbies* instead of *hubby* or *hubby's* would be evaluated as correct.

Source	Category	Phenomenon	Example Target (raw)	Target (edited)
Lena machte sich früh vom Acker.	MWE	Idiom	Lena [left the field early].[2]	Lena left early.
Lisa hat Lasagne gemacht, sie ist schon im Ofen.	Non-verbal agreement	Coreference	Lisa has made lasagne, [she] ist already in the oven.	Lisa has made lasagna, it is already in the oven.
Ich habe der Frau das Buch gegeben.	Verb tense/ aspect/mood	Ditransitive - perfect	I [have the woman of the Book].	I have given the woman the book.

Table 1: Example test suite entries German → English (simplified for display purposes).

Output 2 on the other hand matches the negative regular expression and thus would be evaluated as incorrect. Output 3 does not match any of the regular expressions and therefore would be reconsidered in a follow-up manual check (cf. Section 3.2.). A screenshot of a positive match with a regular expression in the TQ-AuteTest can be seen in Figure 1.

In order to ensure the syntactical correctness of the regular expressions, the TQ-AutoTest also contains a "RegEx Evaluator" in which regular expressions can be tested for syntactical correctness and completeness. Furthermore, regular expressions can be augmented during the evaluation process.

In addition to the regular expressions, we also implemented a feature for positive and negative tokens. With this feature, a whole sentence (i.e., a MT output) can be added to the database to be matched against in subsequent evaluations. This feature is very convenient for phenomena or segments that are more complex. As a consequence, the database is constantly expanding and covers an increasing amount of possible MT outputs.

Every segment in the database is at least either covered by a regular expression or a negative/positive token, some segments by both. 99% of the segments are covered by a positive regular expression, 40% of the segments are covered by a negative regular expression. Most of the sentences that exhibit a negative regular expression do also feature a positive regular expression. Furthermore, 5% of the segments are currently covered by one or more positive tokens while 48% of the segments are covered by negative tokens. Especially the number of segments featuring negative tokens are increasing with every report when new erroneous translations are evaluated. Thus, the database is constantly growing and the more reports are carried out, the less manual work of inspecting segments that are neither covered by a regular expression nor by a positive/negative token is needed.

3.2. Workflow

A typical workflow in the TQ-AutoTest looks as follows (not all steps must necessarily be realized):

Data Preparation An absolute or relative number of sentences from all categories/a selection of categories, resp. from all phenomena/a selection of phenomena is selected randomly (cf. Figure 2) and then scrambled with a random selection of the distractors, whereby the scramble factor can be selected manually. The resulting data is generated in a text file with an ID.

This text file can then be used for running the translations, e.g., on different types of MT systems, say a phrase-based and a neural MT system, or on different version of one system, e.g., before and after some expected improvement.

Upload Report Once the translations are generated (the order of the sentences must be maintained), a text file with the outputs can be uploaded. With the upload, information about the engine (e.g., Google), the type of engine (e.g., NMT) and further comments must/can be entered, cf. Figure 3.

The test sentences are then automatically unscrambled from the distractors and the sentences are evaluated based on the database of regular expressions and tokens.

View Reports In this tab, all reports that have been generated can be viewed and edited. Edited means in this case that sentences that did not match any of the regular expressions or tokens need to be double-checked manually. Correct outputs are shaded in green, incorrect outputs in red and outputs that need to be determined are shaded in yellow. If desired, the evaluation of the manual checking can be added to the database ("Apply Tokens", if it should not be added "Skip Tokens"), cf. Figure 4.

Furthermore, a statistic about the amount of correct/incorrect/tbd translations is automatically generated. This statistic contains tables as well as graphs, both of which can be exported.

Compare Engines This function allows for a comparison of different MT systems/system types that generated translations for the same (sub)set of sentences. Hereby, the absolute and relative numbers of correct/incorrect/tbd translations per systems are calculated (1) on the phenomena, (2) on the categories and (3) on average, and are displayed in tables as well as graphs, both of which can be exported as well.

An exported graph with five different MT systems can be found in Section 4. as an example.

3.3. Implementation

The web interface is implemented using Play Framework, which is open-source, reactive, flexible, and provides Typesafe so that both research and commercial requirements are supported. The front-end uses bootstrap library to ensure compatibility across browsers and platforms, and the back-end is implemented with

[3] Square brackets have been added manually to show the erroneous parts.

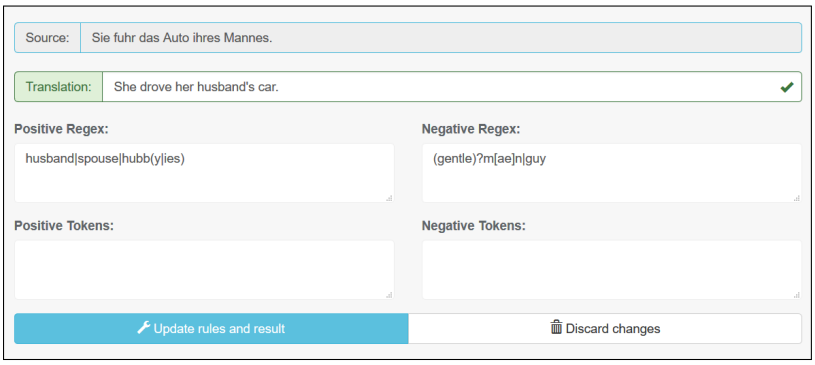


Figure 1: Screenshot: Positive Match with RegEx.

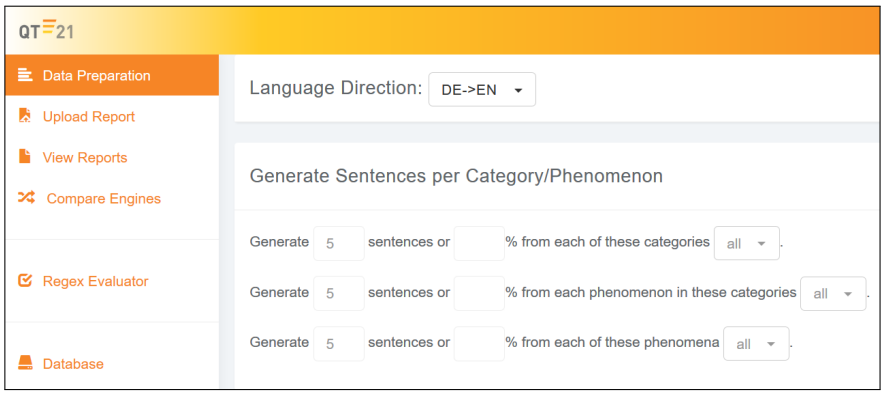


Figure 2: Screenshot: Data Preparation.

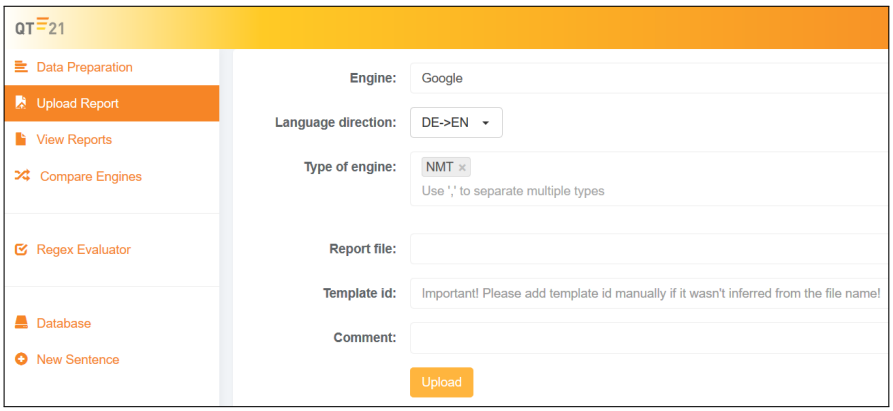


Figure 3: Screenshot: Upload Report.

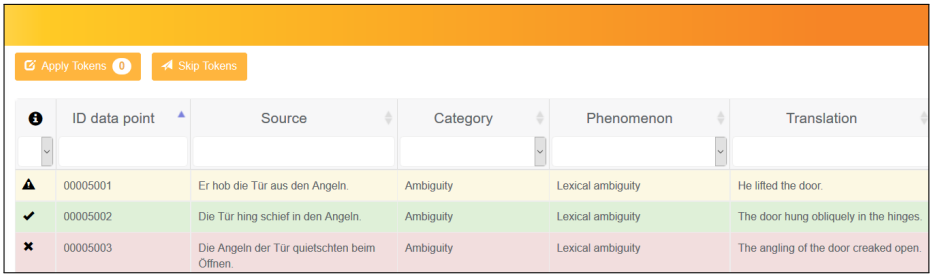


Figure 4: Screenshot: View Reports.

Scala. Both templates and test results are stored in Mysql database. The software can be downloaded at https://gitlab.com/QT21/QT21-Resources/tree/master/Tools/TQ-AutoTest. Note that the test items themselves are not part of the available resources as the purpose of the test set is to test different systems on the same test set.

4. Example of Use

To exemplify TQ-AutoTest, we have compared of the performance of five different MT systems "as-is". The systems we investigated are (1) a neural MT system built by the University of Edinburgh[4], (2) the "old" Google Translate phrase-based statistical MT system (cf. Section 2.), (3) the "new" Google Translate NMT system, (4) the DeepL NMT system[5] and (5) the rule-based MT system Lucy in a completely unadapted version (Alonso and Thurmair, 2003).

The results of the comparison can be found in Figure 5 and 6. Percentage values in boldface indicate that the respective system is significantly better on the particular phenomenon under investigation with a 0.95 confidence level. We calculated the statistical significances by means of a Z-test.

The numbers of instances of segments on the different categories vary strongly. This is due to the fact that we wanted to include the PBMT Google Translate system and we did not have translations of all the segments that are now included in our TQ-AutoTest from this system. We are aware of the fact that the high variety in the numbers of instances (especially the high number of instances in the category verb tense/aspect/mood) creates a bias in the average score. For example, the unadapted Lucy system achieves the highest average score. This high average score is linked to the fact that Lucy is the best-performing system on the category verb tense/aspect/mood. Since Lucy is a rule-based system, many of the rules regarding verb paradigms are probably implemented in the grammar of the system.

Again, we want to stress that it is not our goal to find the "once-and-for-all winning system" with this comparison. The numbers shown here do not represent the corpus frequency of the phenomena. They do solely show tendencies the systems reveal towards the categories that we have tested. Our goal is to provide analytical insights into the systems' strengths and weaknesses in terms of (linguistic) phenomena.

The DeepL system, which is a quite new system, is being promoted as generating better MT outputs than the current Google Translate system. Our numbers support this claim. The Edinburgh NMT system is an almost equal competitor, coming close to the Google NMT average. The Google Translate PBMT system, however, cannot compete with the other systems in this experiment and is the only system that has an average score of less than 50%.

Turning to the scores of the different categories, it becomes clear that the systems perform quite differently on the grammatical phenomena. There are categories in which all of the systems perform quite similar, as for example named entity & terminology. In this category, the scores range from 70.2% to 78.6 %. In other categories as long distance dependency (LDD) & interrogatives on the other hand, the scores range from 39.0% to 77.3%.

These insights can now serve as inspiration for developers to modify the systems (or the training data) in order to improve their performance. We take the findings introduced here as evidence that it is time to complement the reference-based evaluation of MT systems that work well in the laboratory with a reference-independent, more analytical evaluation that can be applied in situations where one does not have full control over the systems, test corpora, where no references are available or where one wants to compare systems just as they are. With our TQ-AutoTest, we provide a tool that semi-automates this more complex evaluation procedure.

5. Conclusion and Outlook

In this paper we have presented TQ-AutoTest, a framework that supports the analytical evaluation of (machine) translations using test suites. Our current test suites comprise about 5000 test items for the language pair German–English in both directions. The framework supports the creation of tests and the evaluation of the translation results using regular expressions. The expressions classify the results as correct, incorrect or requiring a manual check. From our experience, all errors can be checked with regular expressions. As one would expect, e.g, word errors are easier to code than, e.g., grammatical errors. For verb paradigms, we list all possible sentences in the regular expressions as this has turned out to be easier than creating complex regular expressions. We see this as the beginning of more research in the direction we have indicated.

Seen that in previous experiments, we have classified all errors fully manually, the regular expressions provide a drastic reduction of manual labor. After finalizing the regular expressions, we will conduct more tests of the tool. Looking into the future, the approach allows for a number of extensions. Obvious possibilities are more languages and including also domain-specific test suites. Both will be manual work, but given the experience and example we have created will hopefully speed up the process.

It is also imaginable to extend the approach to other NLP applications such as dialogue (Chatbots). We have a concrete request by an industry partner to explore the possibility of evaluating meeting translations that we are currently pursuing.

6. Acknowledgements

We heartily thank Prof. Silvia Hansen-Schirra (Johannes Gutenberg-Universität Mainz) and her team as well as Kim Harris (text & form) for their very valuable support in the creation of the test suite. We also thank our colleague He Wang for helping with the implementation of the tool.

This article has received support from the EC's Horizon 2020 research and innovation programme under grant agreements no. 645452 (QT21) and by the German Federal Ministry of Education and Research (BMBF) through the project BBDC (01IS14013E).

[4] A detailed description of the setup of the system can be found in (Sennrich et al., 2016).

[5] https://www.deepl.com/translate

	#	Edinburgh (NMT)	Google (PBMT)	Google (NMT)	DeepL (NMT)	Lucy (RBMT)
Ambiguity	80	51.2%	**67.1%**	**64.6%**	**74.7%**	**60.0%**
False friends	36	55.6%	**72.2%**	**69.4%**	**83.3%**	**63.9%**
Verb valency	47	51.1%	46.8%	57.4%	**91.5%**	27.7%
Verb tense/aspect/mood	4475	66.1%	46.1%	69.0%	71.6%	**83.0%**
Non-verbal agreement	41	65.9%	**75.6%**	**90.2%**	**92.7%**	51.2%
Punctuation	60	31.7%	**81.7%**	45.0%	43.3%	25.0%
Subordination	91	52.7%	**61.5%**	**74.7%**	**72.5%**	33.0%
MWE	36	**36.1%**	**58.3%**	**41.7%**	**66.7%**	25.0%
LDD & interrogatives	172	62.2%	39.0%	**69.2%**	**77.3%**	51.7%
Named entitiy & terminology	84	78.6%	72.6%	75.0%	81.0%	70.2%
Coordination & ellipsis	80	**63.7%**	32.5%	**56.3%**	**58.8%**	32.5%
Negation	9	100.0%	55.6%	100.0%	77.8%	88.9%
Composition	46	67.4%	78.3%	73.9%	**95.7%**	80.4%
Function word	73	65.8%	53.4%	63.0%	**89.0%**	24.7%
Sum	5330	3460	2554	3654	3854	4109
Average		64.9%	47.9%	68.6%	72.3%	**77.1%**

Figure 5: Exported table: compared Engines on the categories.

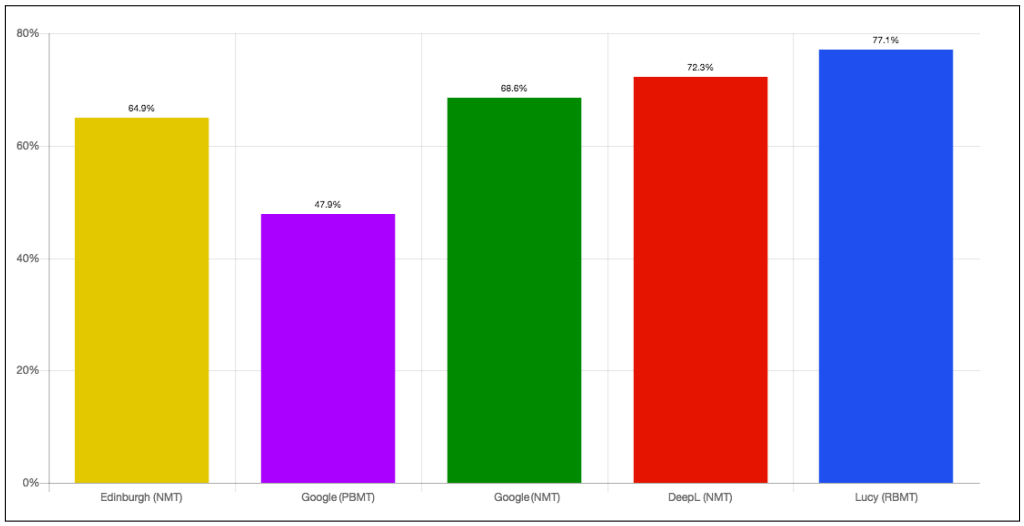


Figure 6: Exported graph: average values of compared engines.

7. Bibliographical References

Alonso, J. A. and Thurmair, G. (2003). The Comprendium Translator system. In *Proceedings of the Ninth Machine Translation Summit*, New Orleans, Louisiana, USA, September.

Bentivogli, L., Bisazza, A., Cettolo, M., and Federico, M. (2016). Neural versus phrase-based machine translation quality: a case study. *CoRR*, abs/1608.04631.

Beyer, A., Macketanz, V., Williams, P., and Burchardt, A. (2017). Can Out-of-the-box NMT Beat a Domain-trained Moses on Technical Data? In *Proceedings of the 20th Annual Conference of the European Association for Machine Translation (EAMT): User Studies and Project/Product Descriptions*, pages 41–46, Prague, Czech Republic, May.

Burchardt, A., Macketanz, V., Dehdari, J., Heigold, G., Peter, J.-T., and Williams, P. (2017). A Linguistic Evaluation of Rule-Based, Phrase-Based, and Neural MT Engines. *The Prague Bulletin of Mathematical Linguistics*, 108(1):159–170.

Guillou, L. and Hardmeier, C. (2016). Protest: A test suite for evaluating pronouns in machine translation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Isabelle, P., Cherry, C., and Foster, G. F. (2017). A Challenge Set Approach to Evaluating Machine Translation. *CoRR*, abs/1704.07431.

Isahara, H. (1995). Jeida's test-sets for quality evaluation of mt systems: Technical evaluation from the developer's point of view. In *Proceedings of the MT Summit V. Luxembourg*.

King, M. and Falkedal, K. (1990). Using test suites in evaluation of machine translation systems. In *Proceedings of the 13th Conference on Computational Linguistics - Volume 2*, COLING '90, pages 211–216, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koh, S., Maeng, J., Lee, J.-Y., Chae, Y.-S., and Choi, K.-S. (2001). A test suite for evaluation of english-to-korean machine translation systems. In *Proceedings of the MT Summit VIII. Santiago de Compostela, Spain*.

Lehmann, S., Oepen, S., Regnier-Prost, S., Netter, K., Lux, V., Klein, J., Falkedal, K., Fouvry, F., Estival, D., Dauphin, E., Compagnion, H., Baur, J., Balkan, L., and Arnold, D. (1996). TSNLP - Test Suites for Natural Language Processing. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 711–716, Copenhagen, Denmark, August.

Popović, M. and Ney, H. (2011). Towards automatic error analysis of machine translation output. *Computational Linguistics*, 37(4):657–688, December.

Schottmüller, N. and Nivre, J. (2014). Issues in translating verb-particle constructions from german to english. In *Proceedings of the 10th Workshop on Multiword Expressions (MWE)*, pages 124–131, Gothenburg, Sweden, April. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Edinburgh Neural Machine Translation Systems for WMT 16. In *Proceedings of the First Conference on Machine Translation (WMT)*, pages 371–376, Berlin, Germany, August.

Tezcan, A., Hoste, V., and Macken, L. (2016). Detecting grammatical errors in machine translation output using dependency parsing and treebank querying. *BALTIC JOURNAL OF MODERN COMPUTING*, 4(2):203–217.

Exploiting Pre-Ordering for Neural Machine Translation

Yang Zhao, Jiajun Zhang and Chengqing Zong
National Laboratory of Pattern Recognition, Institute of Automation, CAS
University of Chinese Academy of Sciences
{yang.zhao, jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

Neural Machine Translation (NMT) has drawn much attention due to its promising translation performance in recent years. However, the under-translation and over-translation problem still remain a big challenge. Through error analysis, we find that under-translation is much more prevalent than over-translation and the source words that need to be reordered during translation are more likely to be ignored. To address the under-translation problem, we explore the pre-ordering approach for NMT. Specifically, we pre-order the source sentences to approximate the target language word order. We then combine the pre-ordering model with position embedding to enhance the monotone translation. Finally, we augment our model with the coverage mechanism to tackle the over-translation problem. Experimental results on Chinese-to-English translation have shown that our method can significantly improve the translation quality by up to 2.43 BLEU points. Furthermore, the detailed analysis demonstrates that our approach can substantially reduce the number of under-translation cases by 30.4% (compared to 17.4% using the coverage model).

Keywords: Neural Machine Translation, pre-ordering, under translation, over translation

1. Introduction

The Past several years have witnessed a significant progress in Neural Machine Translation (NMT). Most NMT methods are based on the encoder-decoder architecture proposed by (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015) and can achieve promising translation performance in a variety of language pairs (Junczys-Dowmunt et al., 2016; Wu et al., 2016).

However, previous studies have showed that NMT suffers from the problems that some source words are mistakenly translated for multiple times meanwhile some words are missed during translation (Tu et al., 2016; Tu et al., 2017; Mi et al., 2016; Feng et al., 2016), which can be called over-translation and under-translation, respectively[1].

Under-translation		Over-translation	
Times	No. Words	Times	No. Words
92	307	32	48

Table 1: Statistics on the under-translation and the over-translation in NMT.

Under-translation		
Reorder	No reorder	Sub-sentence
48	26	18

Table 2: Statistics on different kinds of the under-translation.

In order to figure out the distribution of over-translation and under-translation in NMT, we analyze 500 sentences translated by the NMT system, which is trained by 2.1M parallel Chinese-English sentences pairs. Table 1 shows the statistical results. Specifically, in 500 sentences, NMT system produces 92 under-translations and 32 over-translations. Besides that, for the under-translation, the total number of missing words is 307, while the number of over-translated words is 48. From these statistics, we can see that the under-translation in NMT is more serious than the over-translation.

Therefore, further analysis for the under-translation is made and Table 2 shows the results. In 92 under-translations, we find that the source words should to be reordered during translation are more likely to be missed by NMT and this kind of under-translation occurs 48 times. While the opposite case, i.e. source words requiring no reordering are missed by NMT, occurs 26 times. The remaining (18 times) is the case that the sub-sentences in source are totally dropped. From these statistics, we think that the first kind of under-translation, i.e. words need to be reordered are ignored, is a major problem affecting the final translation quality.

Considering the fact that source words requiring reordering during translation are more likely to be ignored by the NMT model, we propose to exploit the pre-ordering approach which is commonly used in Statistical Machine Translation (SMT). The pre-ordering can make the word order of a source sentence closer to that of a target sentence (Genzel, 2010; Hitschler et al., 2016). We first pre-order the source sentences to approximate the target language word order. We then further combine the pre-ordering model with the position embedding strategy to enhance the monotone translation. Finally, to overcome the over-translation problem, we augment our model with the coverage mechanism.

In this paper, we make the following contributions:

1) Through error analysis, we find that under-translation occurs more frequently than over-translation in NMT and source words that need reordering are more likely to be missed. We propose a pre-ordering approach enhanced with position embedding to tackle the under-translation problem and augment our model with coverage mechanism

[1] (Mi et al., 2016) calls this phenomenon as "repeating and dropping translations". Here, we adopt (Tu et al., 2016)'s expressions, i.e. over-translation and under-translation.

to address the over-translation problem.
2) Our empirical experiments on Chinese-English translation tasks show the efficacy of our approach. We can obtain an average improvement of 1.65 BLEU score on multiple evaluation datasets (the largest improvement can be up to 2.43 BLEU points). Furthermore, the analysis on under-translation shows that our approach can substantially reduce the number of under-translation by 30.4% (compared to 17.4% using the coverage model).

2. Neural Machine Translation

Attention-based NMT contains two parts, encoder and decoder, Encoder transforms the source sentence $X = \{x_1, x_2, ..., x_{Tx}\}$ into context vectors $C = \{h_1, h_2, ..., h_{Tx}\}$. This context set is constructed by m stacked Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) layers. h_j^k can be calculated as follows:

$$h_j^k = LSTM(h_{j-1}^k, h_j^{k-1}) \tag{1}$$

The decoder generates one target word at a time by maximizing the probability of $p(y_i|y_{<i}, C)$ as follows:

$$\begin{aligned} p(y_i|y_{<i}, C) &= p(y_i|y_{<i}, c_i) \\ &= softmax(W_{y_i}\widetilde{z_i} + b_s) \end{aligned} \tag{2}$$

where W_y is an embedding matrix containing row vectors of the target words and $\widetilde{z_i}$ is the attention output:

$$\widetilde{z_i} = tanh(W_c[z_i^m; c_i]) \tag{3}$$

The attention model calculates c_i as the weighted sum of the source-side context vectors:

$$c_i = \sum_{j=1}^{Tx} a_{i,j} h_i^m \tag{4}$$

Where a_{ij} can be computed by

$$a_{i,j} = \frac{exp(e_{i,j})}{\sum_{k=1}^{Tx} exp(e_{i,k})} \tag{5}$$

and

$$e_{i,j} = v_a^T tanh(W_a z_i + U_a h_j) \tag{6}$$

z_i^k is computed using the following formula:

$$z_j^k = LSTM(z_{j-1}^k, z_j^{k-1}) \tag{7}$$

3. Exploiting Pre-Ordering for NMT

In SMT, pre-ordering is a commonly used pre-processing technique (Collins et al., 2005; Zhang and Zong, 2009; Genzel, 2010; Hitschler et al., 2016), which makes the word order of a source sentence closer to that of a target sentence. This technology was originally proposed to alleviate the weakness of reordering in classical phrase-based SMT (Koehn et al., 2003). As SMT always penalizes the cases that move target phrases far away from their corresponding source positions. Fig. 1 shows an example of pre-ordering, in which when translating the original source sentence, the words in red and words in blue need to exchange their positions. With the pre-ordering, the word order in this source sentence is adjusted to the word order in reference. When translating the pre-ordered source sentence, the translation system does not need to reorder the source words.
Since we find that the source words should to be reordered during translation are more likely to be ignored by NMT. We believe that the pre-ordering can help to alleviate the under-translation problem.

3.1. Pre-Ordering

There are many pre-ordering methods introduced in SMT. The most common way to implement a pre-ordering system employs the rule-based approach. The early works rely on hand-written rules (Collins et al., 2005). Later some works could extract the pre-ordering rules automatically (Genzel, 2010; Hitschler et al., 2016). Here, we adopt the automatic rule-based pre-ordering approach. And the procedure is as follows:
With a parallel training corpus, we first train a pre-ordering system. The basic training procedure is extracting the pre-ordering rules, which can minimize the number of alignment crossings in the parallel corpus. More details can be found in (Genzel, 2010; Hitschler et al., 2016).
After acquiring the pre-ordering rules, we can use them to pre-order the source sentences. Note that the word order of the target sentence does not change.

3.2. Position Embedding

As mentioned before, the most noticeable feature of pre-ordering is that it can make the word order in source more consistent with the word order in target. Intuitively, monotone translation is preferred. That is to say the words in the similar positions between the source and target sentences are more likely to be translation pairs. Thus, we further enhance the pre-ordering model with the position embedding to encourage monotone translation.
Actually, previous studies (Cohn et al., 2016; Gehring et al., 2017; Vaswani et al., 2017) have shown that the position information is effective for NMT, and these studies are all based on the following assumption:
Assumption: *a word at a given relative position j in the source (whose length is denoted as J) is more likely to align to a word at a similar relative position i in the target (whose length is denoted as I), i.e. $\frac{j}{J} \approx \frac{i}{I}$.*
Obviously, pre-ordering can make more words satisfy this assumption. We design the procedure as follows:
We first randomly generate the respective position embedding matrix for the source and target positions, which are denoted as $E_s \in \mathbb{R}^{n \times l}$ and $E_t \in \mathbb{R}^{n \times l}$, respectively, where n is the position embedding dimension, and l is largest sentence length. $E_s(j)$ denotes the position embedding for source position j and $E_t(i)$ denotes the position embedding for target position i. Note that the position embedding is optimized during training, like the word embedding.
Then, we redesign the attention part in Eq. 6 as follows:

$$\begin{aligned} e_{i,j} = v_a^T tanh(&W_a z_i + U_a h_j + \\ &W_t E_t(i) + W_s E_s(j)) \end{aligned} \tag{8}$$

where $W_t \in \mathbb{R}^{m \times n}$ and $W_s \in \mathbb{R}^{m \times n}$ are the weight matrices for position embedding with m and n being the hid-

Source:
美国 官员(the us officials) 以 咬文 嚼字 的 外交 用词 (with carefully worded diplomatic rhetoric) 坚称(insisted) 。

Pre-Ordering:
美国 官员 (the us officials) 坚称 (insisted) 以 咬文 嚼字 的 外交 用词 (with carefully worded diplomatic rhetoric)。

Reference:
the us officials insisted with carefully worded diplomatic rhetoric.

Figure 1: A example of pre-ordering.

den states dimension and position embedding dimension, respectively.

As shown in Eq. 8, our attention model contains two parts, namely, hidden states based attention (attention between t_i and h_j) and position embedding based attention (attention between $E_t(i)$ and $E_s(j)$). We hope that when some source words are dropped by hidden states based attention, position embedding based attention could pick them up, and vice versa.

3.3. Coverage Mechanism

In Section 3.2, we propose an approach which combines the pre-ordering model with position embedding. Our experimental results show that this approach can alleviate the under-translation problem, especially can sharply reduce the number of under-translation cases for the words that should be reordered during translation. However, the model lacks the ability to handle the over-translation problem. The detailed statistical data is shown in Section 5.2.

To tackle the over-translation problems, we enhance our model with the coverage mechanism. The coverage mechanism is originally proposed in SMT to indicate whether a source word translated or not. Then, some studies (Tu et al., 2016; Mi et al., 2016) exploit the coverage for NMT. We believe that the coverage mechanism could help to overcome the over-translation problems as they can let NMT consider less about the translated words.

Here, we employ the method proposed in (Tu et al., 2016), which maintains a coverage vector to keep track of the attention history. Then the coverage vector is fed to attention model to adjust the attention in the next step. More Specifically, two steps are needed:

We need to maintain a coverage vector, which summarizes the attention record at each decode step as follows:

$$C_{i,j} = C_{i-1,j} + \frac{1}{\Phi_j} a_{i,j} = \frac{1}{\Phi_j} \sum_{k=1}^{i} a_{i,j} \tag{9}$$

where Φ_j is the fertility for word x_j, and can be computed by

$$\Phi_j = N \cdot \sigma(U_f h_j) \tag{10}$$

Where N is the largest fertility, and U_f is the weight matrix. More details can be found in (Tu et al., 2016).

After generating a coverage vector, we need use this as the complementary information to adjust attention in the next time step. Thus, we rewrite the Eq. 8 as follows:

$$\begin{aligned} e_{i,j} =& v_a^T tanh(W_a z_i + U_a h_j + \\ & W_t E_t(i) + W_s E_s(j) + V_a C_{i-1,j}) \end{aligned} \tag{11}$$

where $C_{i-1,j}$ is the coverage vector of source word x_j before time i, and V_a is the weight matrix for coverage vector.

4. Experimental Settings

4.1. Dataset

We test the proposed approaches on Chinese-to-English translation, which includes 2.1M[2] sentence pairs. NIST 2003 (MT03) dataset is used for validation. NIST2004-2006 (MT04-06) and NIST 2008 (MT08) datasets are used for testing.

4.2. Training and Evaluation Details

We use the Zoph_RNN toolkit[3] to implement our described methods. The encoder and decoder include two stacked LSTM layers. The word embedding dimension, the size of hidden layers and the position embedding dimension are all set to 1,000. Minibatch size is set to 128. We limit the vocabulary to 30K most frequent words for both the source and target languages. Other words are replaced by a special symbol UNK. The largest source and target length is set to 50. At test time, we employ beam search with beam size 12. when the length of test sentence exceeds 50, the embedding for the position > 50 is set to zero. We use case-insensitive 4-gram BLEU score as the automatic metric (Papineni et al., 2002) for translation quality evaluation.

4.3. Pre-Ordering Tool

We use Otedama[4] as the pre-ordering tool. Otedama is an open-source tool for rule-based syntactic pre-ordering. Hyper-parameters we used in Otedama are set as follows: window size is set to 3, matching feature is 10, and the max waiting time is 30 minute. The others are set to the default values. More details can be found in (Hitschler et al., 2016).

4.4. Translation Methods

In the experiments, we compare our approaches with other models, and we list all the translation methods as follows:

1) **Moses**: It is the state-of-the-art phrase-based SMT system (Koehn et al., 2007). Our system is built using the default settings.

2) **Baseline**: It is the baseline attention-based NMT system (Luong et al., 2015; Zoph and Knight, 2016).

[2]LDC2000T50, LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17, LDC2004T07.

[3]https://github.com/isi-nlp/ZophRNN. We extend this toolkit with global attention, and change the attention model to the way shown in Eq. 6.

[4]https://github.com/StatNLP/otedama.

3) **+Pre-Ordering**: It is the NMT system which only uses the pre-ordering approach.
4) **+Position**: It is the NMT system which only employs the position embedding.
5) **+Pre-Ordering+Position**: It is the NMT system using both pre-ordering and position embedding together.
6) **+Coverage**: It is the NMT system with the coverage mechanism (Tu et al., 2016).
7) **+Pre-Ordering+Position+Coverage**: It is the NMT system with the pre-ordering, position embedding and coverage mechanism.

5. Translation Results

5.1. Translation Quality

Table 3 reports the translation results measured in BLEU score. The first question we are interested in is whether or not can the system only using the pre-ordering improve the translation quality. Compared to the baseline system (Row 2), our pre-ordering approach (Row 3) improves the translation results with 0.32 BLEU, indicating that only using pre-ordering in NMT can improve the final results while the improvements are quite small.

The next focus is the effect of combining the pre-ordering system with the position embedding. The system with pre-ordering and position embedding (Rows 5) outperforms the baseline by an average of 1.13 BLEU points. As a comparison, the system only using the position embedding (Row 4) improves the baseline with 0.37 BLEU. Thus, we find an interesting result that when using pre-ordering and position embedding separately, the respective improvement is quite small (0.32 BLEU and 0.37 BLEU, respectively), but using them together can significantly boost the performance (1.13 BLEU), suggesting that pre-ordering and position embedding can enhance each other.

The system which combines the pre-ordering, position embedding and coverage mechanism together (Row 7) further improves the baseline with 1.65 BLEU. As a comparison, the system with only coverage leads to 0.68 BLEU improvement.

5.2. Under-translation and Over-translation

Besides the translation quality, our approaches also aim to reduce the under-translation and over-translation cases in NMT. Therefore, we randomly select 500 source sentences and analyze the translation results produced by different systems to evaluate their performances on the under-translation and over-translation. Table 4 lists the numbers of the under-translation and over-translation produced by different methods.

We first focus on the under-translation cases. Comparing to the baseline (Row 1), the system only using pre-ordering (Row 2) can reduce 3 cases (from 48 to 45) in which the words that require reordering are missed during translation. And the system only using position embedding (Row 3) can reduce 5 cases (from 48 to 43). When we use pre-ordering and position embedding together (Row 4), the the under-translation cases are reduced by 13 ones (from 48 to 35). In addition, the other two kinds of under-translations are also reduced by 7 (from 26 to 19) and 4 (from 18 to 14) times, respectively. The statistics show that the system using the pre-ordering and position embedding can alleviate the under-translation problem, especially for the words that need reordering during translation. Fig. 2 shows an example, in which source words in red and source words in blue need to be reordered during translation. The baseline translates the blue words while drops the red ones. Our approaches using pre-ordering and position embedding can fix this under-translation.

However, when considering the over-translation, we can find a drawback of the system using the pre-ordering and position embedding, it increases 4 (from 32 to 36) over-translation cases. It is thus necessary to augment our model with coverage mechanism. When augmenting our model

Source:
灾难 发生 至今 , 目前 已 有 放弃 寻找 下落 仍 不明 的 数千 名 观光客 的 声音 , 当中 许多 都 是 外国 游客 。

Reference:
since the disaster occurred , there is a voice now to give up the search for thousands of tourists still unaccounted for , many of them foreign tourists .

Pre-Ordering:
灾难 发生 至今 , 目前 已 有 的 声音 放弃 寻找 下落 仍 不明 的 数千 名 观光客, 当中 许多 都 是 外国 游客 。

Baseline:
since the occurrence of the incident , there have been a few thousand tourists who are still unknown , many of whom are foreign tourists .

+Pre-Ordering+Position:
since the occurrence of the disaster , it has a voice now to abandon several thousands of tourists who are still unknown , many of them are foreign tourists .

Figure 2: A translation example investigating pre-ordering and position embedding.

Source:
美国 官员 以 咬文 嚼字 的 外交 用词 坚称 , 他们 虽然 愿意 和 北韩 谈判 , 但是 在 北韩 遵行 禁止 核子 武器 的 各 项 协定 之前 , 他们 不 会 考虑 展开 谈判 程序 。

Reference:
the us officials insisted with carefully worded diplomatic rhetoric that although they are willing to negotiate with north korea , They will not consider any negotiation procedures before north korea abides by various agreements on banning nuclear weapons .

Pre-Ordering:
美国 官员 坚称 以 咬文 嚼字 的 外交 用词 , 他们 虽然 愿意 和 北韩 谈判 , 但是 在 北韩 遵行 禁止 核子 武器 的 各 项 协定 之前 , 他们 不 会 考虑 展开 谈判 程序 。

Baseline:
us officials insist that they are willing to negotiate with north korea . however , they will not consider implementing the negotiation process prior to north korea compliance with all the agreements on banning nuclear weapons .

+Pre-Ordering+Position:
us officials insisted on <UNK> 's diplomatic terms that although they were willing to negotiate with north korea . they will not consider starting the negotiation process until north korea complies with north korea complies with the agreements to ban nuclear weapons .

+Coverage:
united states officials assert that although they are willing to negotiate with north korea. they would not consider initiating negotiations until north korea had adhered to the agreements prohibiting nuclear weapons .

+Pre-Ordering+Position+Coverage:
us officials insisted with <UNK> 's diplomatic terms that although they were willing to negotiate with north korea . they will not consider starting the negotiation until north korea had the agreements on banning nuclear weapons .

Figure 3: A translation example investigating pre-ordering, position embedding and coverage mechanism.

#	System	MT03	MT04	MT05	MT06	MT08	Ave
1	Moses	38.54	39.01	36.55	35.59	24.76	34.89
2	Baseline	38.99	40.69	35.20	38.60	28.48	36.39
3	+ Pre-Ordering	39.06	41.06*	35.80†	38.96*	28.65	36.71
4	+ Position	39.08	41.40†	36.30†	38.16*	28.84*	36.76
5	+ Pre-Ordering+Position	39.92†	41.71†	36.95†	39.75†	29.27†	37.52
6	+ Coverage	39.09	41.26*	36.90†	39.19†	28.93*	37.07
7	+ Pre-Ordering+Position+Coverage	**40.42**†	**42.23**†	**37.63**†	**39.94**†	**29.97** †	**38.04**

Table 3: Translation results (BLEU score) for different translation methods. "*" indicates that it is statistically significant better ($p < 0.05$) than Baseline and "†" indicates $p < 0.01$.

#	Sytem	Under-translation			Over-translation
		Reorder	No reorder	Sub-sentence	
1	Baseline	48	26	18	32
2	+Pre-Ordering	45	24	18	31
3	+Position	43	23	16	35
4	+Pre-Ordering+Position	35	19	14	36
5	+Coverage	41	21	14	**26**
6	+Pre-Ordering+Position+Coverage	**34**	**18**	**12**	27

Table 4: The numbers of under-translation and over-translations produced by different NMT systems.

with coverage mechanism (Row 7), the number of under-translation further decreases. The most important is that it can reduce 9 over-translation cases (from 36 to 27). Fig. 3 shows an example, in which source words in red and source words in blue need to exchange their positions during translation. The result is that the baseline translates the blue words while drops the red ones. Although our approach using pre-ordering and position embedding can fix this under-translation while produces a new over-translation (in green). Fortunately, when we add the coverage mechanism, this over-translation is rectified.

Overall, our approach can substantially reduce the under-translation cases by 30.4%. As a comparison, the system only using coverage reduces 17.4%. For the over-translations, our approach achieves a similar improvement with the coverage model.

6. Related Work

Our work exploits pre-ordering for NMT to improve the under-translation and over-translation. There are two closely related studies:

Improving the under-translation and over-translation. Some previous works attribute the problems of the under-translation and over-translation to the lack of coverage mechanism. Thus they introduce coverage mechanism to NMT. (Tu et al., 2016) maintains a coverage vector at each decode step to collect the attention record, then uses coverage vector to adjust the attention in next time step. (Mi et al., 2016) also maintains a coverage vector, and the difference is that their model introduces a specific coverage embedding for each source word. Further (Tu et al., 2017) proposes a reconstructor for NMT, which can ensure that the information in the source side can be adequately transformed to target side. (Feng et al., 2016) attributes this problem to the lack of explicit distortion and fertility in NMT, and they propose a recurrent attention mechanism to model distortion and fertility. Different from the above methods, we treat this problem with another perspective, as we observe that the words need to be reordered during translation are more likely to be ignored by NMT. Thus we exploit the pre-ordering for NMT to alleviate this problem.

Exploiting techniques in SMT for NMT. Our work is also inspired by the works which incorporating the techniques in SMT to NMT. The earlier related work is conducted on the SMT framework, which is deeply discussed in the reviewed paper (Zhang and Zong, 2015). Here, we only focus on the work which combines the SMT and NMT on NMT framework. Specifically, (Arthur et al., 2016) incorporates word translation table in attention part to adjust the final loss. (Zhang and Zong, 2016) moves forward further by incorporating a bilingual dictionaries in NMT. (Stahlberg et al., 2016) and (He et al., 2016) rescore word candidates with SMT features. (Gülçehre et al., 2015) improves the beam search with language model. (Zhou et al., 2017) proposes a neural combination model to fuse the NMT translation results and SMT translation results. (Wang et al., 2017) improves the NMT system with the SMT recommendations. (Zhang et al., 2014) proposes bilingually-constrained recursive auto-encoders to learn phrase embeddings, which can distinguish the phrases with different semantic meanings. (Tang et al., 2016) explores the possibility to incorporate phrase memory into NMT, in which the decoder can generate a sequence of multiple words all at once.

In this work, we exploit another new technique in SMT, pre-ordering, to NMT to improve the translation performance.

7. Conclusions and Future Work

We have exploited the pre-ordering approach to alleviate the under-translation problem in NMT. Specifically, we pre-order the source sentences to make their word order more consist with the word order in target. Then, we enhance the monotone translation by using the position embedding. Finally, we augment our model with the coverage mechanism.

Our empirical experiments on Chinese-English translation show that the proposed approach can significantly improve the translation quality and substantially reduce the under-translation cases.

However, the under-translation and over-translation problems are still unsolved. In our future work, we plan to propose more effective methods to alleviate the problems. For example, we will design more accurate pre-ordering approaches.

8. Acknowledgments

The research work in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303 and the Natural Science Foundation of China under Grant No. 61333018 and 61673380.

9. Bibliographical References

Arthur, P., Neubig, G., and Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. *In Proceedings of EMNLP 2016*, pages 1557–1567.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *In Proceedings of ICLR 2015*.

Cho, K., Gulcehre, B. v. M. C., Bahdanau, D., Schwenk, F. B. H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. *In Proceedings of EMNLP 2014*.

Cohn, T., Hoang, C. D. V., Vymolova, E., Yao, K., Dyer, C., and Haffari, G. (2016). Incorporating structural alignment biases into an attentional neural translation model. *In Proceedings of NAACL 2016*, pages 876–885.

Collins, M., Koehn, P., and Ivona, K. (2005). Clause restructuring for statistical machine translation. *In Proceedings of ACL 2005*, pages 531–540.

Feng, S., Liu, S., Li, M., and Zhou, M. (2016). Implicit distortion and fertility models for attention-based encoder-decoder nmt model. *arXiv preprint arXiv:1601.03317*.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. *arXiv preprint arXiv:1601.03317*.

Genzel, D. (2010). Automatically learning source-side reordering rules for large scale machine translation. *In Proceedings of COLING 2010*, pages 376–384.

Gülçehre, C., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H.-C., Bougares, F., Schwenk, H., and Bengio, Y. (2015). On using monolingual corpora in neural machine translation. *CoRR, abs/1503.03535*.

He, W., He, Z., Wu, H., and Wang, H. (2016). Improved neural machine translation with smt features. *In Proceedings of AAAI 2016*.

Hitschler, J., Laura, J., Sariya, K., Mayumi, O., Benjamin, K., and Stefan, R. (2016). Otedama: Fast rule-based pre-ordering for machine translation. *The Prague Bulletin of Mathematical Linguistics*, 106:159–168.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is neural machine translation ready for deployment? a case study on 30 translation directions. *arXiv preprint arXiv:1610.01108*.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. *In Proceedings of EMNLP 2013*.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. *In Proceedings of HLT/NAACL 2003*, pages 48–54.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. *In Proceedings of ACL 2007*, pages 177–180.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *In Proceedings of EMNLP 2015*.

Mi, H., Sankaran, B., Wang, Z., and Ittycheriah, A. (2016). A coverage embedding model for neural machine translation. *In Proceedings of EMNLP 2016*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. *In Proceedings of ACL 2002*, pages 311–318.

Stahlberg, F., Hasler, E., Waite, A., and Byrne, B. (2016). Syntactically guided neural machine translation. *arXiv preprint arXiv:1605.04569*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *In Proceedings of NIPS 2014*.

Tang, Y., Meng, F., Lu, Z., Li, H., and Yu, P. L. (2016). Neural machine translation with external phrase memory. *arXiv preprint arXiv:1606.01792*.

Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). Coverage-based neural machine translation. *In Proceedings of ACL 2016*.

Tu, Z., Liu, Y., Shang, L., Liu, Xiaohua, and Li, H. (2017). Neural machine translation with reconstruction. In *Proceedings of AAAI 2017*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., and Kaiser, Å. (2017). Attention is all you need. *arXiv preprint arXiv:1601.03317*.

Wang, X., Lu, Z., Tu, Z., Li, H., Xiong, D., and Zhang, M. (2017). Neural machine translation advised by statistical machine translation. *In Proceedings of AAAI 2017*.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhang, J. and Zong, C. (2009). A framework for effectively integrating hard and soft syntactic rule into phrase based translation. *In Proceedings of PACLIC*, pages 579–588.

Zhang, J. and Zong, C. (2015). Deep neural networks in machine translation: An overview. *IEEE Intelligent Systems*, 30(5):16–25.

Zhang, J. and Zong, C. (2016). Bridging neural machine

translation and bilingual dictionaries. *arXiv preprint arXiv:1610.07272*.

Zhang, J., Liu, S., Li, M., Zhou, M., and Zong, C. (2014). Bilingually-constrained phrase embeddings for machine translation. *In Proceedings of ACL*, pages 111–121.

Zhou, L., Hu, W., Zhang, J., and Zong, C. (2017). Neural system combination for machine translation. *In Proceedings of ACL 2017*, pages 378–384.

Zoph, B. and Knight, K. (2016). Multi-source neural translation. *In Proceedings of NAACL 2016*, pages 30–34.

Improving a Multi-Source Neural Machine Translation Model with Corpus Extension for Low-Resource Languages

Gyu-Hyeon Choi[1], Jong-Hun Shin[2], Young-Kil Kim[3]
[1]Korea University of Science and Technology (UST), South Korea
[2]Electronics and Telecommunication Research Institute (ETRI), South Korea
[3]Electronics and Telecommunication Research Institute (ETRI), South Korea
choko93@ust.ac.kr[1], jhshin82@etri.re.kr[2], kimyk@etri.re.kr[3]

Abstract

In machine translation, we often try to collect resources to improve performance. However, most of the language pairs, such as Korean-Arabic and Korean-Vietnamese, do not have enough resources to train machine translation systems. In this paper, we propose the use of synthetic methods for extending a low-resource corpus and apply it to a multi-source neural machine translation model. We showed the improvement of machine translation performance through corpus extension using the synthetic method. We specifically focused on how to create source sentences that can make better target sentences, including the use of synthetic methods. We found that the corpus extension could also improve the performance of multi-source neural machine translation. We showed the corpus extension and multi-source model to be efficient methods for a low-resource language pair. Furthermore, when both methods were used together, we found better machine translation performance.

Keywords: Neural Machine Translation, Multi-Source Translation, Synthetic, Corpus Extension, Low-Resource

1. Introduction

We often try to collect resources to improve machine translation performance. Using the large size of a parallel corpus, it is possible to achieve high-quality machine translation performance. However, there are many cases where resources of language pairs are insufficient. Except for major European languages and some Asian languages, most of the language pairs do not have sufficient resources to develop a neural machine translation (NMT) system. It is also difficult to obtain parallel corpora for some language pairs such as Korean to Arabic or Korean to Vietnamese.

Since the machine translation performance largely depends on the size of a parallel corpus, it is important to find an efficient way to extend the corpus. Although it is difficult to find a proper parallel corpus, we can create an artificial parallel corpus by translating the source or target of a language pair. Some researchers have studied the extension of a parallel corpus using the pivot method (Cohn and Lapata, 2007; Utiyama and Isahara, 2007; Wu and Wang, 2007). This method introduces another language referred to as the pivot language which is a third language that is different from the source and target languages.

There are many different pivot strategies. The first is the transfer method which translates a source sentence to a pivot sentence and then to a target sentence (Cohn and Lapata, 2007; Wu and Wang, 2007). The second is the triangulation method which multiplies corresponding translation probabilities and lexical weights to create a new source-target phrase table (Utiyama and Isahara, 2007). The third is the synthetic method, which uses existing translation models to build a synthetic parallel source-target corpus from source-pivot or pivot-target (Bertoldi et al., 2008).

There are other approaches that have been proposed for multilingual training with low-resource parallel corpora. Among the approaches, there is a multi-source translation approach where the model has multiple encoders and attention mechanisms for each source language (Zoph and Knight, 2016). The goal of multi-source translation is the translation of a text given in N source languages into a single target language. This considers a case where source sentences are provided in two or more languages. In this study, we combined four other languages to achieve better target language translation. We used four source languages (Korean, English, Japanese, and Chinese) and a single target language (Arabic).

To further improve the multi-source model to be useful for low-resource language pairs, we proposed to use synthetic methods for extending a low-resource corpus and applied it to a multi-source NMT model. Although we can not obtain a high-quality corpus with these methods, it can still be effective in improving multi-source model performance.

Section 2 presents our proposed approach. Section 3 consists of the experimental settings. Section 4 contains experiment results and analysis, followed by a conclusion in section 5.

2. Proposed Approach

We considered a variety of ways to make a model that performs as well as an NMT model with a resource-rich corpus, even though we had to use a low-resource corpus. Among those considered, the corpus extension and multi-source translation method were employed in this study. For the corpus extension, we used a synthetic method, and there are two ways of generating the target and the source. Multi-source translation is an approach that allows one to leverage N-way corpora to improve translation quality in both resource-poor and resource-rich scenarios. Through this method, we were able to observe the improvement of machine translation performance.

2.1 Synthetic Method

There are two approaches to obtain a source-target parallel corpus using the source-pivot and pivot-target corpora. When we were given a pivot sentence, we translated it into a source or target sentence. In each case, translation results were combined with their source and target respectively to get a new parallel corpus. These data are referred to as the synthetic target and the synthetic source. A synthetic target is generated when a target is

translated, and a synthetic source is generated when a source is translated.

2.1.1 Synthetic Target

The synthetic target used to obtain the target translation for source sentences in the source-pivot corpus. It can be obtained by translating pivot sentences to target sentences.

2.1.2 Synthetic Source

We use the synthetic source to obtain source translation for target sentences in the pivot-target corpus. It can be obtained by translating pivot sentences to source sentences. The artificial corpus created by this process is called a "synthetic source" corpus.

2.2 Multi-Source Translation Model

There are other approaches that have been proposed for multilingual training with low-resource parallel corpora. Among the approaches, there is the multi-source translation approach where the model has multiple encoders and attention mechanisms for each source language (Dabre et al., 2017; Garmash et al., 2016). Multi-source translation is the method using N source languages to improve the translation model created by using both low-resources and high-resource scenarios. This model considers a case where the source sentences are provided in two or more languages. According to this method, the model can learn more word vectors of a target language. Then the decoder will be able to generate better target sentence. In this study, we want to combine four other language pairs to get better target language translation. We used four source languages (Korean, English, Japanese, Chinese) and a single target language (Arabic). As the amount of Arabic sentences grows, the number of target word vectors will be increased. Then the word generation capability of the decoder will improve and the translation result will be better.

3. Experimental Settings

In this study, we used various data for the experiments, which consisted of a Korean-Arabic small-scale production parallel corpus as a baseline, and OPUS (Tiedemann et al., 2004) English-Arabic parallel corpus to make synthetic data. We used a WIT[3] (Cettolo et al., 2012) corpus to train a multi-source translation model. We used OpenNMT (Klein et al., 2017) for training the NMT systems in this study. OpenNMT is an open- source implementation of NMT that contains a library for training and deploying NMT models. To tokenize the sentences of the corpus and reduce data sparsity, we applied sub-word tokenization to the source and target sides of a training corpus with the Byte Pair Encoding (BPE) scheme (Sennrich et al., 2016). We used SentencePiece, which is an implementation of the wordpiece algorithm (Schuster and Nakajima, 2012) and BPE.

3.1 Languages and Data Settings

We conducted experiments with a closed production corpus (Prod), a publicly available WIT[3] corpus, and OPUS. The Prod corpus is a Korean-Arabic corpus that contains 157,865 sentences and is manually built for the traveling situation. We set the training data size of the baseline to 150,000 sentences. The WIT[3] corpus is a collection of three parallel corpora made from the transcriptions of TED (Technology, Entertainment, Design) speech, all written in the Arabic language on the target side. The language pairs of those corpora are English-Arabic, Japanese-Arabic, and Chinese-Arabic. We only used them to train the multi-source translation model (MSM). Depending on experimental, we set the training data size of each parallel corpus to 150,000 and 500,000.

Model	Sentences
(1) Prod. Ko-Ar corpus (Baseline)	150,000
(2) (1) + Multi-Source Model (MSM) (Ko/En/Ja/Ch → Ar)	600,000
(3) (1) + Synthetic Target	600,000
(4) (1) + Synthetic Source	600,000
(5) (4) + Multi-Source Model	2,000,000

Table 1: The training data size of each model.

	Language Pair		
WIT[3] - TED corpus	En-Ar	Ja-Ar	Ch-Ar
Original data size	508,925	514,746	520,886
Training data size (2)	150,000	150,000	150,000
Training data size (5)	500,000	500,000	500,000

Table 2: The WIT3 data for the Multi-Source Model (MSM).

Synthetic type	Sentences
Synthetic target (3) Synthetic source (4)	450,000
Synthetic source (5)	350,000

Table 3: The synthetic corpus for using corpus extension.

To extend the training corpus, we used an OPUS English-Arabic corpus, which contains 11 million sentences, to generate a synthetic Korean-Arabic corpus. OPUS was used differently depending on whether it was used for the source side or target side. We used English as a pivot language. When a target side was created, OPUS was used to make an English-Arabic translation model. A synthetic target corpus could be obtained by translating English to Arabic. We translated English into Arabic when the given sentence existed in the Korean-English production corpus[1]. Then, we could obtain a 1.16 million parallel Korean-Arabic corpus after filtering the <unk> symbol from a 2.5 million corpus. When we manipulated the source sides, OPUS was used to obtain a good target language. It can keep Arabic language in high-quality condition. An English-Korean translation model [2] translates English sentences of an OPUS English-Arabic corpus into Korean sentences. We combined the synthetic

[1] This original corpus's line size is about 2.5M. The Korean-English production corpus has a trip domain.

[2] This model is an English-Korean translation model trained by ETRI.

source with the original target. Then we obtained an 800,000 Korean-Arabic parallel corpus through the filtering task. The filtering process consisted of length filtering, deduplication of sentences, and removal of sentences containing the <unk> symbol.

In this paper, we used data with the sizes indicated in Tables 1, 2, and 3. From the extracted data, we selected a fixed training size. As shown in Table 2, we used a WIT[3] corpus consisting of 150,000 sentences. This is because we wanted to minimize variation of each additional corpus size in training a multi-source model. So, to train this model, we used the same size of each additional corpus with an initial baseline production corpus. Finally, we used 600,000 sentences as a multi-source corpus which consisted of Korean-Arabic (Ko-Ar), Englih-Arabic (En-Ar), Japanese-Arabic (Ja-Ar), and Chinese-Arabic (Ch-Ar) parallel language pairs. To compare fairly with the multi-source model (2) in Table 1, it is necessary to make the size of a training corpus equal. Therefore, we used 450,000 sentences of the synthetic corpus to make 600,000 sentences. When we applied the corpus extension method to a multi-source model, we set the corpus size to 500,000 sentences according to the maximum size of WIT[3]. We used the 350,000 sentence synthetic dataset to make 500,000 Korean-Arabic sentences as an initial baseline corpus. The model was trained using a total of 2 million sentences like the model (5) in Table 1.

To measure how well the model is generalizing during training, we used 3,865 development set from a Prod. We used 4,000 1-referenced test set from a Prod corpus. This test set is referred to as trip (TRIP). We extracted 2,000 Korean-Arabic sentences as a 1-referenced test set from a WIT[3] corpus. This test set is called as TED.

3.2 NMT and Model Settings

To train NMT systems, we used OpenNMT and we set the following conditions for training models :

- BPE vocabulary size : 8,000 vocabulary for the source language and 10,000 vocabulary for the target language in all models. When we checked the coverage of BPE models in each language, we found the appropriate size of a BPE model. This size could cover 99.5% of the words.
- Recurrent neural network (RNN) for encoders and decoders : long short-term memory (LSTM) with 4 layers, 1,000 nodes output. Each encoder is a bidirectional RNN. Word embedding size is 500 dimensions, and global attention is also enabled with default parameters.
- Optimization algorithms : stochastic gradient descent (SGD) with an initial learning rate of one which remains the same during the epoch.

We trained and evaluated the following NMT model with a WIT corpus.

- One source to one target : three models (baseline and synthetic extension corpus models)
- Four sources to one target : two models (multi-source translation models)
- Evaluate the performance of the trained models at 20 epochs.

3.3 Automatic Evaluations via Tokenized BLEU

We used the tokenized BLEU-4 (Papineni et al., 2002) automatic evaluation method to measure translation quality. Since Arabic is a rich-morphological language, its performance would be underestimated because non-tokenized BLEU evaluates units separated by whitespaces. Therefore, in this study, Arabic sentences were evaluated based on the results separated by morphemes. We used Farasa (Abdelali et al., 2016), which is an Arabic segmentation tool developed by the Qatar Computing Research Institute (QCRI) to tokenize Arabic words into morphemes.

4. Result and Analysis

4.1 Evaluation results

Tables 4, 5, and 6 show the BLEU scores of our proposed methods. First, we used synthetic data to determine whether the corpus extension method could improve BLEU scores. Table 4 shows the BLEU score of the model trained by a baseline corpus and the models that added synthetic data to the baseline.

For training the multi-source model, we used three different languages pairs. Table 5 showed the BLEU score when we used the multi-source model, which uses Ko-Ar, En-Ar, Ja-Ar, and Ch-Ar corpora as the training data. We found that the BLEU score is better when we use synthetic source data and the multi-source model. To gain additional improvement, we trained a multi-source model using the extended corpus by a synthetic source. Finally, based on the results, training a multi-source model with the synthetic source outperformed all other approaches in a low-resource scenario.

4.2 Analysis

From Tables 4,5 and 6, it is clear that we improved the quality of a translation model by using the corpus extended with a synthetic source for the multi-source model.

We have shown that the corpus extension is suitable for improving the translation model of a low-resource language pair. Table 4 shows that the BLEU score was 1.77 points higher than the baseline in the TRIP test set and 1.73 points in the TED test set when the corpus was extended to a synthetic target. However, when we used the synthetic source method, the BLEU score was increased about 4.96 and 3.86 points in the TRIP and TED test sets, respectively. Through these results, we showed that the synthetic source is more efficient in corpus extension. The reason is that generating source sentences can keep the target sentences in their original native state. The original target sentences enriched the deficient portions of a Prod corpus to improve the quality of the model. We also conducted experiments to demonstrate the effect of a multi-source model. As can be seen in Table 5, the MSM was 4.87 points higher in TRIP and 3.54 points higher in TED than the baseline. Even though the source sentences are different, the MSM can cause the model to

Model	BLEU TRIP(Prod)	TED(WIT3)
(1) Prod. Ko-Ar (baseline)	21.92	6.19
(3) (1) + Synthetic target	23.73	7.92
(4) (1) + Synthetic source	*26.88	*10.05

Table 4: BLEU scores for the baseline and adapting extended corpus.

Model	BLEU TRIP	TED
(1) Prod. Ko-Ar (baseline)	21.92	6.19
(2) (1) + MSM	*26.79	*9.73

Table 5: BLEU scores for the baseline and adapting multi-source model(MSM).

Model	BLEU TRIP	TED
(2) (1) + MSM	26.79	9.73
(4) (1) + Synthetic source	26.88	10.05
(5) (1) + Syn-Source + MSM	*27.07	*12.99

Table 6: BLEU scores of adapting MSM and extended corpus.

have a lot of target information. Therefore, the model can be enhanced to obtain a better translation.

Based on these results, we decided to combine the two methods. We hypothesized that the model performance would be better if we trained the extended corpus with MSM. The results are shown in Table 6. Performance was greatly improved when training a multi-source model with the synthetic source. A model obtained BLEU scores of 27.07 and 12.99 in the TRIP and TED data sets, respectively. In other words, training a multi-source model with a synthetic source can reach the improvement of 5.15 and 6.8 BLEU score for the two test sets.

5. Conclusion

The performance of an NMT system largely depends on the size of the parallel corpus. There are many languages in the world, but most pairs of languages are not rich enough to make a good translation model. Therefore, this paper proposed a method to improve the performance of low-resource language pairs.

In this paper, we used the corpus extension and multi-source translation method to achieve a performance improvement. The two methods of corpus extension: target generation and source generation. The source generation, called the synthetic source, can improve the performance of NMT systems. We showed the corpus extension and multi-source model to be an efficient method for low-resource languages. Furthermore, we achieved better translation performance by using both methods together.

However, the evaluation data was significantly influenced by the domain of the training data, and we found that better evaluation results were obtained in the TED evaluation than in the TRIP. If we use training data in the trip domain, we will also see a high score like the TED result. In the future, we plan to see if we can further improve the TRIP evaluation set by collecting an additional training corpus in the trip domain.

6. Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT)(R7119-16-1001, Core technology development of the real-time simultaneous speech translation based on knowledge enhancement)

7. Bibliographical References

Abdelali, Ahmed, Kareem Darwish, Nadir Durrani and Hamdy Mubarak (2016). Farasa: A Fast and Furious Segmenter for Arabic. In: HLT-NAACL Demos, pages 11-16.

Bertoldi, N., Barbaiani, M., Federico, M., and Cattoni, R. (2008). Phrase-based statistical machine translation with pivot languages. In International Workshop on Spoken Language Translation (IWSLT) 2008, pages 143-149.

Dabre, R., Cromieres, F., and Kurohashi, S. (2017). Enabling Multi-Source Neural Machine Translation By Concatenating Source Sentences In Multiple Languages. *arXiv preprint arXiv:1702.06135*.

Firat, O., Sankaran, B., Al-Onaizan, Y., Vural, F. T. Y., and Cho, K. (2016). Zero-resource translation with multi-lingual neural machine translation. arXiv preprint arXiv:1606.04164.

Firat, Orhan, Kyunghyun Cho, and Yoshua Bengio. (2016). Multi-way, multilingual neural machine translation with a shared attention mechanism. arXiv preprint arXiv:1601.01073.

Garmash, Ekaterina, and Christof Monz. (2016). Ensemble Learning for Multi-Source Neural Machine Translation. COLING.

Hua Wu and Haifeng Wang. (2007). Pivot Language Approach for Phrase-Based Statistical Machine Translation. In Proceedings of 45th Annual Meeting of the Association for Computational Linguistics, pages 856–863.

KLEIN, Guillaume, et al. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. arXiv preprint arXiv:1701.02810.

Masao Utiyama and Hitoshi Isahara. (2007). A Comparison of Pivot Methods for Phrase-Based Statistical Machine Translation. In Proceedings of human language technology: the Conference of the North American Chapter of the Association for Computational Linguistics, pages 484–491.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes and Jeffrey Dean. (2016). Google's

multilingual neural machine translation system: enabling zero-shot translation. arXiv preprint arXiv:1611.04558.

Och, Franz Josef, and Hermann Ney. (2001) Statistical multi-source translation. Proceedings of MT Summit. Vol. 8. 2001, pages 253-258

Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. (2002). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, pp. 311-318.

Schuster, M., and Nakajima, K. (2012). Japanese and Korean voice search. In Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on IEEE, pages 5149-5152.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. (2015). Improving neural machine translation models with monolingual data. arXiv preprint arXiv:1511.06709.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units. arXiv preprint arXiv:1508.07909.

Tevor Cohn and Mirella Lapata. (2007). Machine Translation by Triangulation: Making Effective Use of Multi-Parallel Corpora. In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, pages 348–355.

Wu, H., and Wang, H. (2009). Revisiting pivot language approach for machine translation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 154-162.

Xiaoguang Hu, Haifeng Wang, and Hua Wu. (2007). Using RBMT Systems to Produce Parallel corpus for SMT. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 287–295.

Zoph, Barret, and Kevin Knight. (2016). Multi-source neural translation. arXiv preprint arXiv:1601.00710.

8. Language Resource References

Cettolo, M., Girardi, C., & Federico, M. (2012). Wit3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 268.

Tiedemann, Jörg, and Lars Nygaard. (2004). The OPUS Corpus-Parallel and Free: http://logos. uio. no/opus. LREC.

Dynamic Oracle for Neural Machine Translation in Decoding Phase

Zi-Yi Dou, Hao Zhou, Shu-Jian Huang, Xin-Yu Dai, Jia-Jun Chen
National Key Laboratory for Novel Software Technology, Nanjing University
Nanjing, 210023, China
141242042@smail.nju.edu.cn, {zhouh, huangsj, daixy, chenjj}@nlp.nju.edu.cn

Abstract

The past several years have witnessed the rapid progress of end-to-end Neural Machine Translation (NMT). However, there exists discrepancy between training and inference in NMT when decoding, which may lead to serious problems since the model might be in a part of the state space it has never seen during training. To address the issue, Scheduled Sampling has been proposed. However, there are certain limitations in Scheduled Sampling and we propose two dynamic oracle-based methods to improve it. We manage to mitigate the discrepancy by changing the training process towards a less guided scheme and meanwhile aggregating the oracle's demonstrations. Experimental results show that the proposed approaches improve translation quality over standard NMT system.

Keywords: machine translation, dynamic oracle, language model

1. Introduction

Neural networks have been widely used contemporarily and have achieved great performance on a variety of fields like sentiment analysis (Santos and Gattit, 2014) and visual object recognition (Ciregan et al., 2012). For sequential problems, recurrent neural networks can be applied to process sequences. To address issues like long term dependencies in the data (Bengio et al., 1994), the Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or Gated Recurrent Unit (GRU) can be used to tackle the problem (Cho et al., 2014). A straightforward application of the LSTM and GRU architecture have already shown impressive performance in several difficult tasks, including machine translation (Sutskever et al., 2014), and image captioning (Vinyals et al., 2015).

Generally, a basic sequence-to-sequence model consists of two recurrent neural networks: an encoder that processes the input and a decoder that generates the output (Cho et al., 2014). In many applications of sequence-to-sequence models, at inference time, the output of the decoder at time t is fed back and becomes the input of decoder at time $t+1$. However, during training, it is more common to provide the correct input to the decoder at every time-step even if the decoder made a mistake before, which leads to a discrepancy between how the model is used at training and inference. As has been pointed out by Bengio *et. al.* (2015), although this discrepancy can be mitigated by the use of a beam search heuristic maintaining several generated target sequences, for continuous state space models like recurrent neural networks, there is no dynamic programming approach, so the effective number of sequences considered remains small. The main problem is that mistakes made earlier in the sequence generation process are fed as input to the model and can be quickly amplified because the model might be in a part of the state space it has never seen at training time (Bengio et al., 2015).

There are several existing methods to bridge the gap between training and inference. Bengio *et. al.* (2015) propose a method called *Scheduled Sampling*. Since the main difference between training and inference for sequence prediction tasks when predicting token y_t is whether we use the true previous token y_{t-1} or an estimate $\hat{y}_{t-1}$ coming from the model itself, they apply a sampling mechanism that will randomly decide, during training, whether to use the true previous token y_{t-1} or an estimate $\hat{y}_{t-1}$ coming from the model itself. Specifically, for every token, they flip a coin and decide whether to use the true previous token or an estimate coming from the model itself. Their general framework is shown in **Figure 1**. At the beginning of training, the model is not well-trained and thus selecting more often the true previous token should be helpful; whereas at the end of training the sampling strategy should favor sampling from the model more often, as this corresponds to the true inference situation. Therefore, the probability of using y_{t-1} will be initially high and then decrease during training. In their paper, they choose three different functions to model the probability of using y_{t-1} with respect to training time and they achieve competitive results.

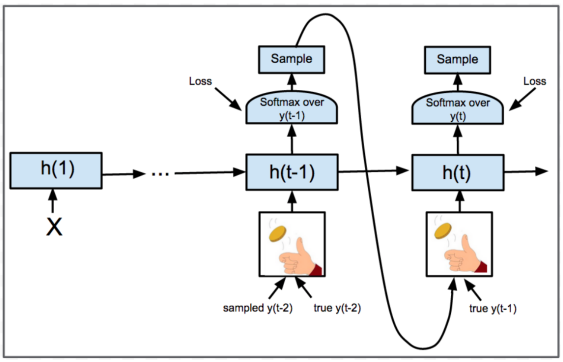


Figure 1: Scheduled Sampling

Although *Scheduled Sampling* has been proved to be helpful on several tasks (Bengio et al., 2015), for machine translation tasks, in our experiments it does not show promising performance. We have done a few research and found out a dissatisfactory characteristic about *Scheduled Sampling*. To illustrate, as we can see from **Figure 2**, since the reference has been altered, the original correct inputs would not be accurate and therefore it is unwise to still provide the model with the original inputs.

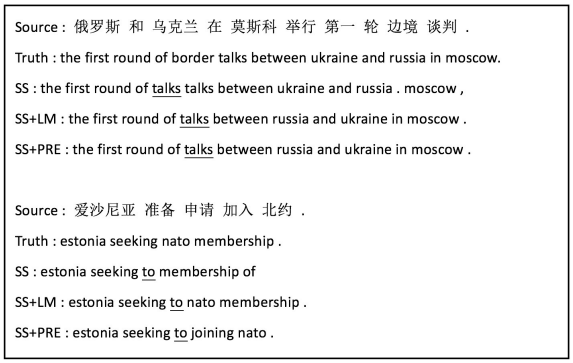


Figure 2: Sampling Results

In another words, *Scheduled Sampling* still uses the gold word for training whereas it would be no longer correct, which is a static oracle fashion. We believe that if we can use a certain strategy, such as *dynamic oracle*, to always provide the model with accurate inputs, the performance will be better.

In this paper, based on the aforementioned idea, we propose two methods which use *dynamic oracle* to solve the issue mentioned above. To illustrate, once we decide to use an estimated token coming from the model itself, our *oracle* can help us select the next suitable word and feed it into the model. Therefore, even during training time the truth have been completely altered, our method will always provide the model with the best suitable token in the following time-steps. In this way, we hope that the aforementioned problem in *Scheduled Sampling* would be mitigated.

Here we develop two strategies to implement the *dynamic oracle*, one is based on language model and the other is based on pre-trained neural machine translation system. These two methods could manage to feed the correct word into the model so that they may enhance the performance of *Scheduled Sampling*.

To verify the effectiveness of our methods, we conduct experiments on Chinese-English datasets and the experimental results indicate that our method can achieve $+1.06$ BLEU improvements.

2. Related Work

To the best of our knowledge, Goldberg *et. al.* (2012) first define the concept of *dynamic oracle* and propose an online algorithm for parsing problems, , which provides a set of optimal transitions for every valid parser configuration. For configurations which are not part of a gold derivation, their dynamic oracle permits all transitions that can lead to a tree with minimum loss compared to the gold tree. Based on their approach, several other methods using dynamic oracle have been proposed (Goldberg and Nivre, 2013) (Gómez-Rodrıguez et al., 2014). However, their work in the field of parsing cannot be directly applied in neural machine translation.

To mitigate the discrepancy between training and inference, Daume *et al.* (2009) introduce SEARN, which aims to tackle the problems that training examples might be different from actual test examples. They show that structured prediction can be mapped into a search setting using language from reinforcement learning, and known techniques for reinforcement learning can give formal performance bounds on the structured prediction task. In addition, Dataset Aggregation (DAgger) (Ross et al., 2011) is another method which adds on-policy samples to its dataset and then re-optimizes the policy by asking human to label these new data.

3. Proposed Methods

In this section, we first give a brief introduction of neural machine translation. And then we present the general framework for our algorithms. At last, we describe our two methods respectively, namely language model guided scheduled sampling and pre-trained model guided scheduled sampling.

3.1. Neural Machine Translation

Neural machine translation aims to directly model the conditional probability $p(Y|X)$ of translating a source sentence, $x_1, ..., x_n$, to a target sentence, $y_1, ..., y_m$. Generally, it accomplishes this goal through an encoder-decoder framework (Kalchbrenner and Blunsom, 2013). Basically, the encoder generates a context vector for each source sentence and then the decoder outputs a translation, one target word at a time.

During training when we are decoding, we always provide the model with true previous token at every time step. Minibatch stochastic gradient descent is applied to look for a set of parameters θ^* that maximizes the log likelihood of producing the correct target sentence. Specifically, given a batch of training pairs $\{(X^i, Y^i)\}$, we aim to find θ^* which satisfies:

$$\theta^* = \arg\max_{\theta} \sum_{(X^i, Y^i)} \log p(Y^i|X^i; \theta) \tag{1}$$

Whereas during inference the model can generate the target sentence one token at a time, advancing time by one step. This procedure will continue until an <EOS> token is generated. Since at time t we do not have access to the true previous token, normally we just feed the model with the most likely token given our model at time t. To search for the sentence with the highest probability, beam search is often used.

The log conditional probability can be decomposed as:

$$\log p(Y|X) = \sum_{t=1}^{m} \log p(y_t|y_{<t}, C), \tag{2}$$

where C is the context vector for source sentence.

The log conditional probability $\log p(y_t|y_{<t}, C)$ is computed in different ways according to the choice of the context C at time i. Bahdanau *et. al.* (2014) use different context c_i at different time step while Cho *et. al.* (2014) choose $C = \mathbf{h}_{T_x}$.

The architecture of recurrent neural network may differ in terms of architecture and type. For example, there can be unidirectional, bidirectional or deep multi-layer RNN; and RNN type can be LSTM (Hochreiter and Schmidhuber, 1997) or the GRU(Cho et al., 2014).

3.2. General Framework and Definitions

The general goal of our work is to mitigate the discrepancy between training and inference when using recurrent neural network. To achieve this goal, we borrow the idea from *Scheduled Sampling*. Specifically, during training when decoding, there is a possibility that we feed our model with an estimate $\hat{y}_{t-1}$ coming from the model itself rather than true previous token y_{t-1}. Different from *Scheduled Sampling*, once we decide to use $\hat{y}_{t-1}$, the original ground-truth would not be accurate any more and thus we use *dynamic oracle* to generate the next suitable token y^*_{t-1} and regard it as our true token. The pseudo-code of our general framework is shown in Algorithm 1.

Algorithm 1 General Framework

Input: At time t when decoding
Output: Input token i_{t-1}

```
Sampling p from [0, 1]
if p < ε_t then
    i_{t-1} = ŷ_{t-1}
else
    if have not used ŷ_i, i = 1, 2, ..., t − 2 before then
        i_{t-1} = y_{t-1}
    else
        Generating y = ŷ*_{t-1} by dynamic oracle
        i_{t-1} = ŷ*_{t-1}
    end if
end if
```

3.3. Language Model-based Dynamic Oracle

Our first method utilizes language model to guide the training procedure.

The goal of language modeling is to predict the next word in textual data given specific context. Here we use recurrent neural network based language models. Traditionally, backing-off language models rely on the n-gram approximation, which are often criticized because they could only store limited information and thus lack any explicit representation of long range dependency. In contrast, Recurrent neural network language models always estimate probabilities based on the full history (Sundermeyer et al., 2012) In other words, recurrent neural networks do not use limited size of context, which is the major reason why we choose recurrent neural network language models.

Our first oracle is based on language model. Basically, the well-trained language model will help us to generate y^*_{t-1} given partially decoded sentence. Therefore, it could guarantee the coherence of our generated sentence.

It should be noticed that in our scenario, even though the language model would predict a probability distribution over the whole dictionary, we only select the word with the highest probability within the reference of the source sentence. In this way, we hope that the generated sentence can be both coherent and precise.

3.4. Pre-trained Model-based Dynamic Oracle

Our second method utilizes a pre-trained neural machine translation model to guide the training procedure.

The basic procedure is similar to our first method *i.e.* we randomly decide whether to use the true previous token or the estimated token coming from the model itself. Once we choose an estimated word, the next time when we decide to choose the true previous token, we provide the model with the token generated by the pre-trained model.

At this time, we do not have to select the word with the highest probability within the reference of the source sentence rather than the whole dictionary. Compared to the previous method, this pre-trained model guided scheduled sampling not only utilize the information in the target sentence, but also utilize the information in the source sentence. Therefore, we would expect the performance of the second method would be better than the first method.

Also, since at this time we do not limit our selections within the reference if source sentences, this model could add more diversity into the model and thus allow one sentence to have more translations.

4. Experiment

In this section we first describe the dataset used in our experiments, the training and evaluation details , the baseline model we compare in experiments. And then we present the quantitive results of our experiments. At last, we would show some qualitative characteristics and demonstrate the potential of our model

4.1. Dataset and Setup

We carry out experiments on a Chinese-English translation task. Our training data for the translation task consists of 1.25M sentence pairs extracted from LDC corpora [1], with 27.9M Chinese words and 34.5M English words respectively. For our development set, we choose NIST 2002 dataset. Also, we choose the NIST 2003, 2004, 2005 as our test sets.

Each neural machine translation model is trained using stochastic gradient descent algorithm AdaGrad (Fazayeli, 2014). We use mini-batch size of 32. The word embedding dimension of source and target language is 600 and the size of hidden layer is set to 1000. Also, for efficient training of the neural networks, we limit the source and target vocabularies to the most frequent 30K words in Chinese and English , covering approximately 97.7% and 99.3% of the two corpora respectively. All the out-of-vocabulary words are mapped to a special token "UNK". We use case-insensitive 4-gram BLEU score as the evaluation metric (Papineni et al., 2002).

4.2. Baseline Model

In this work, our baseline neural machine translation system is attentional encoder-decoder networks as implemented in DL4MT[2], which is an open source phrase-based translation system available in github. In this framework, the baseline uses conditional Gated Recurrent Unit (cGRU)

[1]The corpora include LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06.

[2]https://github.com/nyu-dl/dl4mt-tutorial

	MT02	MT03	MT04	MT05	AVG
baseline	33.55	31.01	33.28	30.29	32.03
SS	33.47	31.88	32.98	30.72	32.26
SS+LM	34.56	32.16	33.36	30.79	32.71
SS+PRE	**34.63**	**32.18**	**34.18**	**31.37**	**33.09**

Table 1: BLEU results

with attention mechanism as the hidden unit type. The specific details of cGRU can also be found in github [3].

4.3. Results

Table 1 shows the BLEU results for the baseline systems and the dynamic oracle guided machine translation systems. As we can see from the table, our method generally achieve better performance than the baseline models. Moreover, the best average BLEU results improve more than 1 point on our dataset. Since the language model only utilizes the information in the target sentence, as we would expect, the pre-trained model behaves better. However, the language model-based oracle also improves the translation performance further testify the effectiveness of our algorithms.

4.4. Case Studies

In order to testify that our methods guide the system towards the right way, we sample a few examples, as illustrated in **Figure 2**.

As we can see, in the first sentence, we need to translate the Chinese sentence into "the first round of border talks between ukraine and russia in moscow ." If we just use *Scheduled Sampling*, then at fourth time the model will make a mistake and generate "talks" instead of "border". Therefore, the next time if we still choose the original true previous token "talks", the sentence will not be accurate. In such scenario, our *dynamic oracle* can be helpful. In this case, both the language model-based oracle and pre-trained model-based oracle can generate "between" instead of "talks", which make the reference become accurate again.

In the second sentence, we want our system to output "esotina seeking nato membership .", whereas *Scheduled Sampling* generates "esotina seeking to membership of". Here, the model makes a mistake by generating the word "to". Again, as shown in the figure, our oracle can feed the model with the suitable word.

These examples demonstrate that our methods can be helpful in some cases and thus the improvement in BLEU would be unsurprising.

4.5. Analysis of Translation Results

An important property of our second method, namely the pre-trained model-based dynamic oracle for NMT, is that it may provide the system with various translations for each sentence. Traditionally, the model can only be trained with one unchangeable reference for each sentence while there may exist a couple of correct references. In our second proposed method, since the baseline model may generate token which is not limited to single reference, we could add some diversity to NMT system. To verify our hypothesis, we present a few samples from the baseline model and our second method.

> **Reference**: japan temporarily freeze humanitarian assistance to russia .
>
> **Baseline**: japan freezes its offer of humanitarian assistance to russia in the interim .
>
> **SS+PRE**: japan freezes humanitarian aid to russia for the time being .

As shown above, in the first example our reference is "japan temporarily freeze humanitarian assistance to russia ." Clearly, if we train the model with our second method, it will output which is both precise and not limited to the original reference.

Let us consider another example:

> **Reference**: at this time , the police have blocked the bombing scene .
>
> **Baseline**: at this time , the police have UNK the blast at the scene .
>
> **SS+PRE**: the police have blocked the scene at the moment .

This example is another proof which shows that our second method could indeed add some diversity into NMT system.

5. Conclusion and Future Work

In this paper we highlight a major issue for NMT, *i.e.* the mismatch between where a model may end up in training and testing in the search space.

Scheduled Sampling has been proposed to deal with this and we propose two methods to fix the drawbacks of the *Scheduled Sampling* . *Scheduled Sampling* only feeds the previous predicted word into RNN decoder but still using the gold word for training, which is a static oracle fashion. The training oracle of our proposed method, on the other hand, is dynamic, according to a language model or a pre-trained NMT model.

Also, as has been shown in Section 4.5, our second model, namely the pre-trained model-based dynamic oracle for NMT, could provide the system with various translations for each sentence. This feature is worth further research and could help enhance the performance of the model further.

Scheduled Sampling has been used in a variety of NLP tasks and it is curious that it does not help here. Although we have pointed out one potential problem, further investigation may help explain why this is the case.

[3] https://github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf

6. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Ciregan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE.

Fazayeli, F. (2014). Adaptive subgradient methods for online learning and stochastic optimization.

Goldberg, Y. and Nivre, J. (2012). A dynamic oracle for arc-eager dependency parsing. In *COLING*, pages 959–976.

Goldberg, Y. and Nivre, J. (2013). Training deterministic parsers with non-deterministic oracles. *Transactions of the association for Computational Linguistics*, 1:403–414.

Gómez-Rodrıguez, C., Sartorio, F., and Satta, G. (2014). A polynomial-time dynamic oracle for non-projective dependency parsing. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 917–927.

Hal DaumÃ©, I., Langford, J., and Marcu, D. (2009). Search-based structured prediction. *Machine Learning*, 75(3):297–325.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. In *EMNLP*, volume 3, page 413.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Ross, S., Gordon, G. J., and Bagnell, J. A. (2011). A reduction of imitation learning and structured prediction to no-regret online learning. *Aistats*, abs/1011.0686.

Santos, C. N. D. and Gattit, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *International Conference on Computational Linguistics*.

Sundermeyer, M., Schlüter, R., and Ney, H. (2012). Lstm neural networks for language modeling. In *Interspeech*, pages 194–197.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164.

One Sentence One Model for Neural Machine Translation

Xiaoqing Li, Jiajun Zhang, Chengqing Zong
National Laboratory of Pattern Recognition, CASIA, Beijing, China
{xqli, jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

Neural machine translation (NMT) becomes a new state of the art and achieves promising translation performance using a simple encoder-decoder neural network. This neural network is trained once on the parallel corpus and the fixed network is used to translate all the test sentences. We argue that the general fixed network parameters cannot best fit each specific testing sentences. In this paper, we propose the dynamic NMT which learns a general network as usual, and then fine-tunes the network for each test sentence. The fine-tune work is done on a small set of the bilingual training data that is obtained through similarity search according to the test sentence. Extensive experiments demonstrate that this method can significantly improve the translation performance, especially when highly similar sentences are available.

Keywords: Neural machine translation, online learning, sentence similarity

1. Introduction

Neural machine translation achieved great success recently (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Bahdanau et al., 2015). Thanks to the end-to-end training paradigm and the powerful modeling capacity of neural network, NMT can produce comparable or even better results than traditional statistical machine translation, only after a few years of development. However, it also raises some new problems, such as how to use open vocabulary and how to avoid repeating and missing translations. These problems have been addressed by various recent approaches (Luong et al., 2015b; Jean et al., 2015; Tu et al., 2016; Mi et al., 2016).

How to learn a good set of parameters is another challenge for nowadays deep neural networks. There has been some work in the field of NMT. Shen et al. (2015) propose to use task specific optimization function. Specially, they propose to directly optimize BLEU score instead of likelihood of the training data. Bengio et al. (2015) take search into consideration during training. In common practice, the decoder uses gold reference as history during training, but it has to use generated output as history during testing. To fix this discrepancy between training and testing, the authors propose to moderately replace gold reference with generated output during training. Wiseman and Rush (2016) take a similar approach and regard training as beam search optimization.

However, no matter how the network parameters are learnt, they are fixed after the training is finished in all current NMT practice. And the same model is applied to every testing sentence. A potential issue of this practice is that a neural network needs to be able to compress all translation knowledge into a fixed set of parameters, which is very hard in reality. So we propose to learn a specific model for each testing sentence by paying more attention to those related sentences. In particular, we propose a learning on-the-fly strategy for parameter fine-tuning. First, a general model is learnt from the whole training data. Then, for each testing sentence, we find some similar sentence pairs from the training data and use them to fine tune the parameters.

This procedure resembles how human do translation. Given a sentence, especially one we are not familiar, we always would like to search for some similar sentences and see how they are translated. Various translation knowledge can be learned from these examples, such as how to translate a lexicon or phrase in a specific context, and how to reorder the translation of different blocks according to some syntactic clues. Once our translation knowledge is refreshed, we can handle the sentence with much higher confidence.

There are two key aspects for the method. One is how to define similarity and the other is how to find similar sentence pairs efficiently. For similarity measure, we tried string based similarity and hidden representation based similarity. Our approach has two additional steps compared with plain decoding: finding similar sentence pairs and fine tuning. To improve the efficiency, we used the technique of inverted index for fast retrieval. We also studied how the size of similar data influences the decoding time.

Experimental results show our approach can effectively improve the translation performance, especially when highly similar sentences are available.

2. Background

In this section, we will briefly introduce the NMT system from Bahdanau et al. (2015), which will be used later in the experiments. However, our approach is model independent and can be applied to other NMT systems, such as the recently proposed convolutional seq2seq network (Gehring et al., 2017) and the Transformer network (Vaswani et al., 2017), which do not use recurrent network as the encoder or decoder.

Given a source sentence $s = (s_1, s_2, ...s_m)$ and its translation $t = (t_1, t_2, ..., t_n)$, NMT models the translation probability with a single neural network as follows,

$$p(t|s) = \prod_{i=1}^{n} p(t_i|t_{<i}, s) \quad (1)$$

where the conditional probability is parameterized with the encoder-decoder framework. The encoder reads the source sentence and encodes it into a sequence of hidden states

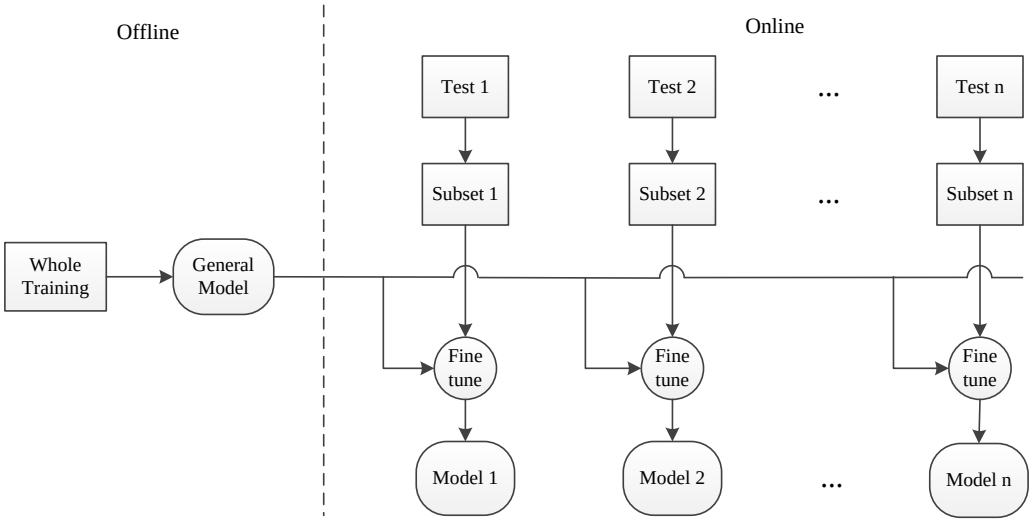


Figure 1: System architecture for our method

$h = h_1, h_2, ..., h_m$ with bidirectional GRU.

$$h_i = [\overrightarrow{h}_i; \overleftarrow{h}_i] \tag{2}$$

$$\overrightarrow{h}_i = \overrightarrow{\phi}(\overrightarrow{h}_{i-1}, x_i) \tag{3}$$

$$\overleftarrow{h}_i = \overleftarrow{\phi}(\overleftarrow{h}_{i+1}, x_i) \tag{4}$$

where x_i is the embedding of current word, and the recurrent activation functions $\overrightarrow{\phi}$ and $\overleftarrow{\phi}$ are gated recurrent units. The decoder consists of a recurrent neural network and an attention mechanism. The recurrent neural network computes a hidden state for each target position as follows,

$$z_j = \phi(z_{j-1}, y_{j-1}, c_j) \tag{5}$$

where z_{j-1} is the previous hidden state, y_{j-1} is the embedding of previous word and c_j is the context vector obtained by the attention machenism, which decides which source words to look at when predicting current target word.

$$c_j = \sum_{i=1}^{m} \alpha_{i,j} h_i \tag{6}$$

and the weight $\alpha_{i,j}$ is calculated as follows,

$$\alpha_{i,j} = \frac{exp(e_{i,j})}{\sum_{k=1}^{m} exp(e_{k,j})} \tag{7}$$

$$e_{i,j} = f_{ATT}(z_{j-1}, h_i) \tag{8}$$

Then the probability of generating a specific target word w will be computed by

$$p(t_j = w | t_{<i}, s) = softmax(z_j^\top y_w) \tag{9}$$

where y_w is the embedding of target word w.

3. Tuning on-the-fly

As illustrated in Figure 1, the learning strategy of our approach is simple. First, we learn a general model from the whole training corpus. Then, for each testing sentence, we extract a small subset from the training data, consisting of sentence pairs whose source sides are similar to the testing sentence. This subset is used to fine tune the general model and a specific model is obtained for the testing sentence.
This procedure can be formulated as two stage optimization. The first stage is to to find a set of network parameters θ to maximize the log likelihood of the whole training data $D = \{(s^{(1)}, t^{(1)}), (s^{(2)}, t^{(2)}), ..., (s^{(N)}, t^{(N)})\}$.

$$\begin{aligned}\hat{\theta} &= \arg\max_{\theta}\{\mathcal{L}(\theta)\} \\ &= \arg\max_{\theta}\{\log \prod_{k=1}^{N} P(t^{(k)}|s^{(k)};\theta)\} \\ &= \arg\max_{\theta}\{\sum_{k=1}^{N}\sum_{i=1}^{|t^{(k)}|} \log P(t_i^{(k)}|s^{(k)}, t_{<i}^{(k)};\theta)\}\end{aligned}$$

The second stage is to find a set of parameters in the neighbourhood of $\hat{\theta}$ to maximize the log likelihood of a subset of data similar to the testing sentence x.

$$\bar{\theta} = \arg\max_{\theta \in \mathcal{N}(\hat{\theta})}\{\log \prod_{s^{(k)} \sim s} P(t^{(k)}|s^{(k)};\theta)\}$$

In the following parts, we will discuss how to evaluate similarity between two sentences and how to quickly find similar sentences from training data.

3.1. Similarity Measure

There are many methods to evaluate the similarity between two sentences. In this paper, we consider three of them. The first is based on Levenshtein distance, which counts at least how many operations do we need to convert one sequence to another. The operations include insertion, deletion and substitution. Levenshtein distance reflects the sur-

face similarity of two sentences, and it does not consider the meaning of the sentence.

$$sim_{LD}(s1, s2) = 1 - \frac{LD(s1, s2)}{max(|s1|, |s2|)}$$

The second measure is based on average word embedding (Mikolov et al., 2013) of the sentence. Although this sentence representation is simple, it has been shown competitive to many complex sentence representations in many tasks.

$$sim_{emb}(s1, s2) = cos(\frac{\sum_{i=1}^{|s1|} emb(s1[i])}{|s1|}, \frac{\sum_{j=1}^{|s2|} emb(s2[j])}{|s2|})$$

The third measure is based on the hidden states of the encoder in NMT. Unlike word embedding, the hidden states of the encoder contains context information. What's more, the hidden states is learnt in the translation task. For this similarity measure, we need to run the encoder first with the general model learnt offline to get the representation of the testing sentence. This representation will be compared with the representation of training sentences, which need only to be calculated once in an offline manner.

$$sim_{enc}(s1, s2) = cos(\frac{\sum_{i=1}^{|s1|} h1[i]}{|s1|}, \frac{\sum_{j=1}^{|s2|} h2[j]}{|s2|})$$

where $h1[i]$ and $h2[j]$ are the hidden states of the two sentences, which are calculated according to equations (2) - (4).

3.2. Finding similar sentences efficiently

The training corpus for neural machine translation usually contains millions of sentences. For a given testing sentence, comparing it with every training sentence will be too time consuming. So we propose to filter the training corpus first by only considering those which have common words with the testing sentence, and then compute similarity with the filtered set.

We use inverted index for fast retrieval. Each training sentence is given a unique index. And we maintain a word to indexes map, recording the sentence indexes where each word appears. For efficiency consideration, we ignore the most frequent words, which usually are function words and punctuations. Then for each word in a testing sentence, we find all sentences which contain the word. And the union of these sentences are used as the filtered set.

However, calculating Levenshtein distance between the testing sentence and each sentence in the filtered set is still not fast enough. So we propose to further reduce the set with a simpler similarity measure, i.e. dice coefficients.

$$sim_{dice} = \frac{2|set(w \in s1) \cap set(w \in s2)|}{|set(w \in s1)| + |set(s \in s2)|}$$

We first calculate the dice coefficients between the testing sentence and each sentence in the filtered set, then reduce the size of the set to a given threshold, e.g. 1000, by keeping the sentences with the highest dice coefficients. Finally, we will calculate Levenshtein distance for the reduced set.

For the other two similarity measures, calculating cosine similarity can be done efficiently with linear algebra library. So there is no need to further reduce the filtered set.

3.3. Fine tuning

The process of fine tuning is almost the same with offline training. The main difference is that the data size used for fine tuning is very small, usually containing only a few sentence pairs. So we need to be careful about overfitting. To this end, we go over the tuning data for only one pass. Learning rate is another factor need to be attended. Too large learning rate will cause overfitting, and too small learning rate will make it hard to learn translation knowledge from the tuning data. According to our pilot study, optimization methods with adaptive learning rate, such as Adadelta (Zeiler, 2012), work as well as SGD with carefully tuned learning rate, so we adopt it in our experiments.

4. Handle the case with low similarity

We cannot always find very similar sentences to the testing sentence, especially when there is not enough in-domain training data. In this case, we propose to find sentences to maximize phrase coverage. The phrase we mention here has the same meaning as the one in phrase-based machine translation, which denotes any consecutive word sequence. Our motivation is to select a subset of training data which can cover as many phrases in the testing sentence as possible. The method to find the subset is shown in Algorithm 1.

Input: testing sentence x, training data D, phrase table PT
Output: a subset of training data D_x
$D_x \leftarrow \phi$;
for $i \leftarrow 1$ **to** max_phrase_len **do**
  for $j \leftarrow 1$ **to** $|x| - i$ **do**
    check if $x[j : j + i]$ in PT;
    if *True* **then**
      foreach $phrase \in phrase_pairs$ **do**
        find a sentence pair containing $phrase$;
        add the sentence pair to D_x;
      end
    else
      continue;
    end
  end
end

Algorithm 1: Find a subset to maximize phrase coverage

The algorithm iterates over all possible phrases in the testing sentence and check if it is contained in the phrase table, which is extracted according to aligned bilingual corpus. The table contains a list of phrase pairs in the following form,

source ||| target ||| score1 score2 score3 score4

The four scores for each phrase are direct phrase translation probability $\phi(t|s)$, inverse phrase translation probability $\phi(s|t)$, direct lexical weighting $lex(t|s)$, inverse lexical weighting $lex(s|t)$, which are used to evaluate the quality of the phrase pair from different angles. The direct phrase translation probability and lexical weighting are calculated as follows. The inverse ones are calculated similarly.

$$\phi(t|s) = \frac{Count(s,t)}{Count(s)}$$

$$lex(t|s) = \prod_{i=1}^{|t|} \frac{1}{|\{j|(i,j) \in a\}|} \sum_{\forall(i,j)\in a} p(e_i|f_j)$$

A source phrase may corresponds to many (up to hundreds or thousands) target phrases, we filter them according to the average of the above four scores and keep those with the highest score. If a phrase in the testing sentence matches some source side in the phrase table, we will find a sentence pair in the training data which contains the source side and one of its high-score target side. Since there may be many sentence pairs containing such phrase pair, we choose one with the largest likelihood as follows, which means the sentence pair is simple and easy to learn.

$$(\hat{s}, \hat{t}) = \underset{\text{phrase in } (s,t)}{\arg\max} \prod_{i=1}^{|t|} P(t_i|t_{<i}, s; \theta)$$

The translation probability of each training sentence pair is calculated offline with the general network parameters.

We don't use the phrase pairs as training data to fine-tune the network parameters. There are two reasons. First, context information is not available for choosing the proper phrase translation. Second, training on phrase pairs will harm the recurrent weights of the network, because they are not complete sentences[1].

5. Experiments

We evaluate our proposed method on the Chinese to English translation task. Translation quality is measured by the BLEU metric (Papineni et al., 2002).

5.1. Datasets

We conduct experiments on two datasets. One is on the United Nations Parallel Corpus[2], which is composed of official records and other parliamentary documents of the United Nations. Since this data is from a narrow domain, it is relatively easy to find similar sentences for many testing sentences. The training data contains 1M sentence pairs extracted from the corpus, and the testing data contains 5 groups of sentence pairs, with 200 sentence pairs in each group. The most similar[3] sentence we can find for the sentences in each group falls into the similarity range of 0-0.2, 0.2-0.4, 0.4-0.6, 0.6-0.8 and 0.8-1.0, respectively. We also randomly selected 1,000 sentence pairs as the development set.

The training data of the other dataset is selected from LDC[4], which contains about 1.2M sentence pairs, whose sources ranges from news, laws, hansard records, weblogs, spoken dialogues, etc. And we use NIST 03 as development set, and NIST 04 to 06 as testing set. In contrast to the UN data, we can hardly find very similar sentences to the testing one in this setting.

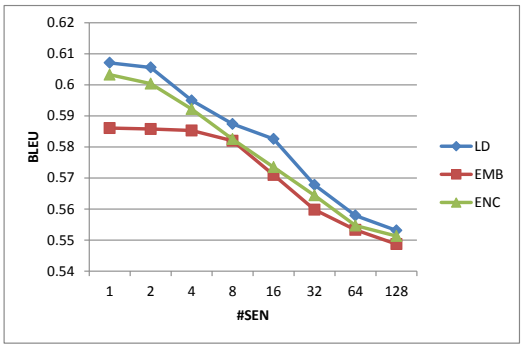


Figure 2: Performance with different similarity measures when different number of similar sentences are used

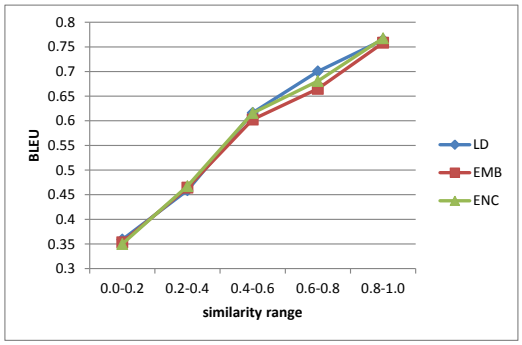


Figure 3: Performance with different similarity measures on testing data with different similarity range

5.2. Experiment Setting

The hyperparameters used in our network are described as follows. We limit both the source and target vocabulary to 30k in our experiments. The number of hidden units is 1,000 for both the encoder and decoder. And the embedding dimension is 500 for all source and target tokens. The network parameters are updated with the Adadelta algorithm for both training and fine tuning.

When finding similar sentences based on phrase coverage, we keep top two target phrase for each source phrase. And if a source phrase appears more than 1,000 times in the bilingual corpus, it will be discarded, because it's unnecessary to re-learn how to translate these common phrases.

5.3. Experiments on UN Data

We first conduct experiments on the UN corpus, studying which similarity measure is better, and how many similar sentences should be used against each similarity range.

[1] We also tried to fix the recurrent weights and tune the word embeddings only, it performs better than tuning all weights, but still worse than the approach of tuning on complete sentences.

[2] http://conferences.unite.un.org/UNCorpus

[3] The similarity is calculated based on Levenshtein distance.

[4] https://www.ldc.upenn.edu/

System	0-0.2	0.2-0.4	0.4-0.6	0.6-0.8	0.8-1
baseline	36.45	46.78	58.06	60.64	60.52
fine-tune	37.23	48.25	63.79	71.73	78.21

Table 1: Best performance on each group of UN testing data

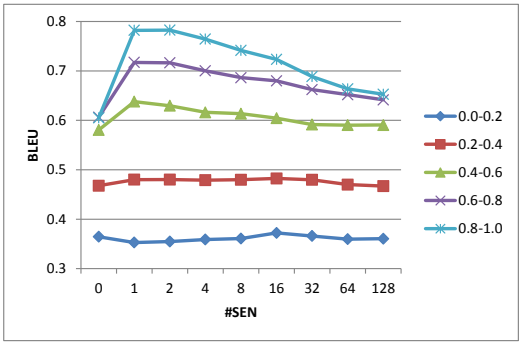


Figure 4: How the size of similar data influence the performance of testing data with different similarity range

5.3.1. Similarity Measure

Figure 2 shows the performances of the three similarity measures when different number of similar sentences are used. There are two observations according to this figure. First, the performance of the similarity based on Levenshtein distance is always better than the other two, the similarity based on encoder states is slightly worse, and the similarity based on average word embedding is the worst. Since Levenshtein distance only cares string similarity, similar sentences found according to this measure will have more words in common with the testing sentence, thus more parameters related to the word embedding can be updated. And the encoder states takes context information into consideration when compared with averaged word embedding, so it has better performance.

Second, the performance gap between different similarity measures become smaller when more similar sentences are used. This is due to the fact that there will be a larger overlapping in the sentences found by the three measures when more sentences are used.

To further check the difference between the three measures, we fix the number of sentences used for fine tuning as 4, and show the performance of the three measures on testing sentences in different similarity range in Figure 3. It can be seen from the figure that, when very un-similar (0-0.4) or very similar (0.8-1.0) sentences can be found for the testing sentence, the performances of the three measures have little difference. When the sentences in a relatively high range (0.4-0.8), especially in (0.6-0.8), can be found for the testing sentence, the performance of the Levenshtein distance based similarity is obviously better.

5.3.2. Data Size

According to Figure 2, using only 1 similar sentence for fine-tuning performs best. However, this figure only shows the overall performance on the whole testing data. If we dive into testing sentences with different similarity range, the trend will be different, as shown in Figure 4. We adopt

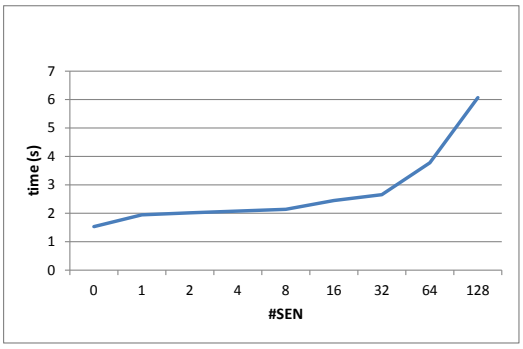


Figure 5: How time cost increases while more sentences are used for fine-tuning

System	04	05	06	Avg.
baseline	36.06	32.74	34.85	34.55
fine-tune	37.43	34.01	35.77	35.74
fine-tune (phrase)	38.04	34.41	35.41	35.99

Table 2: Experimental results for LDC data

the Levenshtein distance based similarity in this experiment. It can be seen from the figure that, if very similar sentences (0.4-1.0) can be found for the testing sentence, using only 1 similar sentence can greatly improve the performance, using more does not provide further help and may even degrade the performance. However, when the found sentences are not very similar (0-0.4), the improvement brought by fine-tuning is much smaller, and using more sentences, such as 16, is better than using one. Less sentences will lead to more severe overfitting, which will make the model remember how to reproduce the translations of the sentences. This is desirable when very similar sentences can be found, but it will produce negative effect otherwise.

The best performance we can get for each group of testing sentences are shown in Table 1. It can be seen from the table that more than 10 BLEU points can be gained when we can find very similar (0.6-1) sentences to the testing one. However, if we cannot find very similar sentences (0-0.4), only minor (around 1 BLEU point) improvement can be gained.

5.3.3. Influence on Efficiency

The influence of data size on efficiency is shown in Figure 5. The time cost in the figure only includes fine-tuning time and decoding time. The retrieval time, i.e., time of finding similar sentences, is not shown because it is relatively small compared to the other two. Retrieval with edit distance measure is the slowest one. But it is still less than 1/3 of the decoding time. We can see from the figure if less or equal than 32 sentences are used for fine-tuning for each testing sentence, the time cost is controlled within two times of the baseline. If we use 128 sentences, the time cost increases to 4 times.

input	再次要求以色列向秘书长提供一切便利和协助, 以执行本决议
reference	calls once again upon israel to render all facilities and assistance to the secretary - general in the implementation of the present resolution
sim.	再次要求以色列向秘书长提供一切便利以执行本决议
trans. of sim.	calls once more upon israel to render all facilities to the secretary - general in the implementation of the present resolution
baseline	reiterates its request to the secretary - general to provide all facilities and assistance to the secretary - general for the implementation of the present resolution
ours	calls once more upon israel to render all facilities and assistance to the secretary - general in the implementation of the present resolution
input	经讨论商定，去掉方括号，保留其中的内容。
reference	after discussion it was agreed to delete the square brackets and retain the contents therein .
sim.	工作组商定，去掉该款的方括号。
trans. of sim.	the working group agreed to remove square brackets from this paragraph .
baseline	after discussion , it was agreed that the removal of the content would be deleted .
ours	after discussion , it was agreed to remove square brackets and retain the contents of it .

Table 3: Translation examples of our method

5.4. Experiments on LDC Data

In this experiment, we can only find similar sentences in the range of 0-0.4 for more than 90% of the testing sentences. And according to our study on the development set, the number of sentences used for fine-tuning needs to be increased to 128 to get the best performance when the similarity is low. We think the reason is due to the diversity of the training data. Sentences in the low similarity range may have totally different topics and styles with the testing one. In order to avoid the influence of these unwanted data, more sentences need to be used.

The performances on the testing data are shown in Table 2. They are obtained with the following setting, if very similar sentences (0.4-1) can be found, we use only 1 sentence for fine-tuning, otherwise we use 128 sentences. On average, 1.2 BLEU points can be gained on the three testing sets, which is consistent with the experimental results on the UN dataset when very similar sentences cannot be found.

The performances of finding similar sentences based on phrase coverage are also shown in the table. The average improvement is 1.45 BLEU points, slightly better than the approach of finding similar sentences directly. And the average sentence number used for fine-tuning is 31, much less than 128. So the time cost is almost halved (see Figure 5).

5.5. Result Analysis

We show two examples in Table 3. The above one is the case where highly similar sentence can be found to the testing sentence. The only difference between the input sentence and the similar sentence is that there are two extra words ”和协助” (and assistance) and an extra comma. After fine-tuning, the model remembers how to generate the translation for the similar sentence. Based on the backbone, it can produce a correct translation for the testing sentence with a minor modification. Whereas the baseline skips the source word ”以色列” (Israel) and translates the source word ”秘书长” (secretary - general) twice.

In the lower example, we can only find a not so similar sentence to the testing one, with a similarity score of 0.31. However, the sentence pair found in the example can remind the model how to translate the phrase “方括号”, whose translation is missing in the baseline system.

6. Related Work

After a few pioneer work in exploring neural features in SMT systems (Zhang et al., 2014; Devlin et al., 2014), NMT quickly become the dominant approach for machine translation. Kalchbrenner and Blunsom (2013) and Cho et al. (2014) first propose to use the encoder-decoder architecture to do sequence to sequence mapping. At the same time, Sutskever et al. (2014) apply it in end-to-end machine translation. Bahdanau et al. (2015) propose the attention mechanism to dynamically attend to different source words when generating different target words, which becomes the default component of current NMT systems.

Recent advances in NMT include fixing defects of the model, such as inability to use large vocabulary (Luong et al., 2015b; Jean et al., 2015), unawareness of coverage (Tu et al., 2016; Mi et al., 2016) etc, making use of mono-lingual data (Cheng et al., 2016; Sennrich et al., 2015), extending to multi-lingual(Dong et al., 2015; Zoph and Knight, 2016) and multi-modal (Hitschler and Riezler, 2016) scenarios.

In statistical machine translation, Liu et al. (2012) proposes a local training method which also learns sentence-wise weights based on similar sentences. However, since there are only about a dozen of features in SMT, such as translation score and language model score, adjusting the relative weights of these features cannot making full use of the similar sentences. There are some other work making use of similar sentences by means of translation memory (Koehn and Senellart, 2010; Ma et al., 2011; Bertoldi et al., 2013; Wang et al., 2013; Li et al., 2014). However, they need carefully designed features and only show improvement when similarity level is high. In comparison, our method don’t need any modification to the model, and it can bring improvement in all similarity level.

Finding similar sentences with inverted index is fast enough in our experiments. If the training data is much larger than ours, locality sensitive hash such as MinHash (Broder, 1997) may be a better choice.

7. Conclusion and Future Work

In this paper, we propose to learn a specific model for each testing sentence. This is accomplished by two-stage training. An general model is learnt offline on the whole bilingual training corpus. During testing, a small batch of similar sentences are extract to fine-tune the network parameters on-the-fly. Experimental results demonstrate the effectiveness of this approach. When highly similar sentences are available, the improvement can exceed 10 BLEU points.

In the future, we plan to test the effectiveness of our method with different network architechtures, such as stacked LSTM (Luong et al., 2015a), and different training objectives, such as MRT (Shen et al., 2015). Since our method is model independent, we also plan to apply it to other tasks beyond machine translation.

8. Acknowledgment

The research work descried in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303 and the Natural Science Foundation of China under Grant No. 61333018 and 61673380.

9. Reference

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *ICLR 2015*.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179.

Bertoldi, N., Cettolo, M., and Federico, M. (2013). Cache-based online adaptation for machine translation enhanced computer assisted translation. *Proceedings of the XIV Machine Translation Summit*, pages 35–42.

Broder, A. Z. (1997). On the resemblance and containment of documents. In *Compression and Complexity of Sequences 1997. Proceedings*, pages 21–29. IEEE.

Cheng, Y., Xu, W., He, Z., He, W., Wu, H., Sun, M., and Liu, Y. (2016). Semi-supervised learning for neural machine translation. *arXiv preprint arXiv:1606.04596*.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October. Association for Computational Linguistics.

Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R. M., and Makhoul, J. (2014). Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380.

Dong, D., Wu, H., He, W., Yu, D., and Wang, H. (2015). Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the ACL and the 7th International Joint Conference on Natural Language Processing*, pages 1723–1732.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning.

Hitschler, J. and Riezler, S. (2016). Multimodal pivots for image caption translation. *arXiv preprint arXiv:1601.03916*.

Jean, S., Cho, K., Memisevic, R., and Bengio, Y. (2015). On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Beijing, China, July. Association for Computational Linguistics.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA, October. Association for Computational Linguistics.

Koehn, P. and Senellart, J. (2010). Convergence of translation memory and statistical machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, pages 21–31.

Li, L., Way, A., and Liu, Q. (2014). A discriminative framework of integrating translation memory features into smt. In *Proceedings of the 11th Conference of the Association for Machine Translation in the Americas*, volume 1, pages 249–260.

Liu, L., Cao, H., Watanabe, T., Zhao, T., Yu, M., and Zhu, C. (2012). Locally training the log-linear model for smt. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 402–411. Association for Computational Linguistics.

Luong, T., Pham, H., and Manning, C. D. (2015a). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September. Association for Computational Linguistics.

Luong, T., Sutskever, I., Le, Q., Vinyals, O., and Zaremba, W. (2015b). Addressing the rare word problem in neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 11–19, Beijing, China, July. Association for Computational Linguistics.

Ma, Y., He, Y., Way, A., and van Genabith, J. (2011). Consistent translation using discriminative learning: a translation memory-inspired approach. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-*

Volume 1, pages 1239–1248. Association for Computational Linguistics.

Mi, H., Sankaran, B., Wang, Z., and Ittycheriah, A. (2016). A coverage embedding model for neural machine translation. *arXiv preprint arXiv:1605.03148*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2015). Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Shen, S., Cheng, Y., He, Z., He, W., Wu, H., Sun, M., and Liu, Y. (2015). Minimum risk training for neural machine translation. *arXiv preprint arXiv:1512.02433*.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). Coverage-based neural machine translation. *arXiv preprint arXiv:1601.04811*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need.

Wang, K., Zong, C., Su, K.-Y., et al. (2013). Integrating translation memory into phrase-based machine translation during decoding. In *ACL (1)*, pages 11–21.

Wiseman, S. and Rush, A. M. (2016). Sequence-to-sequence learning as beam-search optimization. *arXiv preprint arXiv:1606.02960*.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zhang, J., Liu, S., Li, M., Zhou, M., and Zong, C. (2014). Bilingually-constrained phrase embeddings for machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 111–121.

Zoph, B. and Knight, K. (2016). Multi-source neural translation. *arXiv preprint arXiv:1601.00710*.

A Parallel Corpus of Arabic–Japanese News Articles

Go Inoue,[1,2] **Nizar Habash,**[2] **Yuji Matsumoto,**[1] **Hiroyuki Aoyama**[3]
[1]Computational Linguistics Laboratory, Nara Institute of Science and Technology, Japan
[2]Computational Approaches to Modeling Language Lab, New York University Abu Dhabi, UAE
[3]Graduate School of Global Studies, Tokyo University of Foreign Studies, Japan
{inoue.go.ib4, matsu}@is.naist.jp, nizar.habash@nyu.edu, aljabal@tufs.ac.jp

Abstract

Much work has been done on machine translation between major language pairs including Arabic–English and English–Japanese thanks to the availability of large-scale parallel corpora with manually verified subsets of parallel sentences. However, there has been little research conducted on the Arabic–Japanese language pair due to its parallel-data scarcity, despite being a good example of interestingly contrasting differences in typology. In this paper, we describe the creation process and statistics of the Arabic–Japanese portion of the TUFS Media Corpus, a parallel corpus of translated news articles collected at Tokyo University of Foreign Studies (TUFS). Part of the corpus is manually aligned at the sentence level for development and testing. The corpus is provided in two formats: A document-level parallel corpus in XML format, and a sentence-level parallel corpus in plain text format. We also report the first results of Arabic–Japanese phrase-based machine translation trained on our corpus.

Keywords: Arabic, Japanese, Parallel Corpus, Sentence Alignment, Machine Translation

1. Introduction

Machine translation (MT) has been a very active research area in natural language processing. Whether its paradigm is statistical or neural, the availability of parallel data is essential for building high-quality systems. In particular, manually verified data sets for development and testing are of great importance for improving and evaluating MT systems. Much work has been done on MT between major language pairs including Arabic–English and Japanese–English, thanks to the availability of large-scale parallel corpora across various domains with manually aligned subsets. However, there has been little research conducted on the Arabic–Japanese language pair due to its parallel-data scarcity, despite being a good example of interestingly contrasting differences in typology. For instance, Arabic is a verb-initial language, while Japanese is a verb-final language, where the position of verb is completely opposite as shown in Figure 1. An Arabic token can be highly ambiguous in morphological, syntactical, and lexical levels due the absence of optional diacritics for short vowels and consonant doubling. In addition, Arabic has a complex system of derivation, inflection, and cliticization. In contrast, a Japanese token can be highly ambiguous due to the absence of spaces between tokens. For more details in linguistic issues, see Habash (2010) for Arabic and Bond and Baldwin (2016) for Japanese.

In this paper, we present a parallel corpus of Arabic–Japanese news articles, part of which is manually aligned at the sentence level for tuning and evaluation. We also provide the first results of Arabic–Japanese phrase-based MT trained on our corpus.

The corpus represents an ongoing project carried out at Tokyo University of Foreign Studies (TUFS) entitled TUFS Media Project,[1] which produces translated news articles in eight languages (Arabic, Bengali, Burmese, Indonesian, Persian, Turkish, Urdu, and Vietnamese). Our corpus serves as a pilot corpus for building parallel corpora of under-resourced language pairs under this project, as well as a basis for investigating various MT techniques for under-resourced language pairs such as pivoting and domain adaptation in future work.

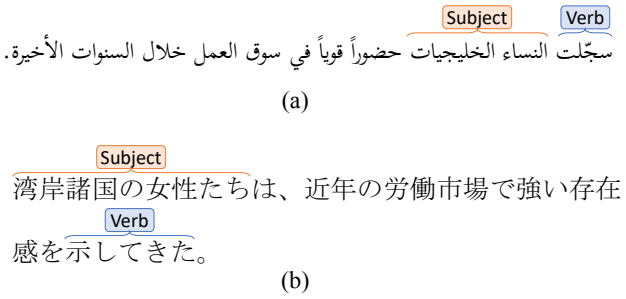


Figure 1: Example of Arabic (a) and Japanese (b) text in parallel. Note that Arabic is written from right to left, and Japanese is written from left to right. *"Women in Gulf countries have shown a strong presence in the recent labour market."*

The corpus is provided in two formats: (a) A document-level parallel corpus of 8,652 document pairs with genre annotation in XML, and (b) a sentence-level parallel corpus in plain text format. The sentence-level parallel corpus consists of 64,488 sentence pairs, with approximately 2.4 million Arabic tokens[2] and 3.7 million Japanese tokens[3] in total. Our corpus is publicly available for research purposes.[4]

[1]http://www.el.tufs.ac.jp/tufsmedia/

[2]Throughout this paper, an Arabic token is defined as a simple tokenization unit (D0) (Habash, 2010) as shown in Table 1.

[3]A Japanese token is defined as a unit used in IPAdic (2.7.0.) (Asahara and Matsumoto, 2003).

[4]http://el.tufs.ac.jp/tufsmedia-corpus/

Tokenization	Operation	Example
raw	no tokenization	*wsyktbhA llTAlb.*
D0	split punctuations and numbers	*wsyktbhA llTAlb* ␣.
D1	split CONJ	*w+* ␣ *syktbhA llTAlb* ␣.
D2	split CONJ and PART	*w+* ␣ *s+* ␣ *yktbhA l+* ␣ *AlTAlb* ␣.
ATB	split all clitics except the definite article	*w+* ␣ *s+* ␣ *yktb* ␣ *+hA l+* ␣ *AlTAlb* ␣.
D3	split all clitics	*w+* ␣ *s+* ␣ *yktb* ␣ *+hA l+* ␣ *Al+* ␣ *TAlb* ␣.
D3*	remove the definite article from D3	*w+* ␣ *s+* ␣ *yktb* ␣ *+hA l+* ␣ *TAlb* ␣.

Table 1: Examples of the various tokenization schemes for the raw input وسيكتبها للطالب. *wsyktbhA llTAlb.* 'And he will write it for the student.' Arabic characters are transliterated in the Habash-Soudi-Buckwalter transliteration scheme (Habash et al., 2007). The symbol "␣" denotes a space added after tokenization. CONJ and PART refer to conjunctions and particles, respectively.

2. TUFS Media Corpus

In this section, we describe the source of our corpus, details of the corpus construction process, and statistics of our corpus.

2.1. Source of the Corpus

TUFS Media Project is an ongoing project carried out at Tokyo University of Foreign Studies to offer translated news from various countries and regions around the world in order to familiarize the Japanese society with the current world events. The initial version of the project was launched in 2005, offering translated articles from three languages: Arabic, Turkish, and Persian. Currently, the project provides translated articles into Japanese from eight languages, Arabic, Bengali, Burmese, Indonesian, Persian, Turkish, Urdu, and Vietnamese.

Translation of an article is done in two steps, initial translation and proofreading. In the initial translation step, a translator, typically an undergraduate student who majors in Arabic, choses an appropriate article from one of the news agencies[5] in accordance with the person's interest following the translation guideline. The guideline describes rules regarding the choice of an article to be translated, formatting, and transcription. The translator then translates title, dateline, and paragraphs in the article. The paragraphs can be omitted as long as the text to be translated contains over 200 words in Arabic. In that case, the translator inserts a phrase that denotes omission in the translated article. The translator also assigns a concise title and classifies the article into 12 categories based on the content. In the proofreading step, a proofreader, who is an expert in Arabic or a graduate student with experience studying in the region, proofreads the translated article and publishes it on the project website.

2.2. Corpus Construction

We describe next the process of corpus construction, from data collection to sentence alignment.

2.2.1. Crawling Documents from Project Website

We crawled the project website[6] which provides a searchable interface for translated articles, specifying the six news agencies and the issue date that ranges from 2005 to 2016. The crawling yielded 9,915 translated articles in HTML format. For Arabic, we collected original articles by crawling the provided links to the original urls and the archived versions in MHTML format. MHTML files were converted to HTML files in order to simplify the succeeding scraping process. The crawling of original articles yielded 9,056 documents[7] in total.

2.2.2. Scraping Crawled Documents

For Japanese, we extracted translated text, category, issue date, and links to the original articles from the documents using HTML tags as clues. The Japanese data are more structured than the Arabic data thanks to the unified HTML architecture and the translation guidelines, however, there are some cases where we could not find corresponding translations for the original title and/or dateline.

The Arabic data are more difficult to process due to their format variations across six different agencies with periodically different templates within agencies. In some cases, we could not extract main texts from the documents due to their structural issues in HTML. In such cases, we simply discarded these documents from our corpus. Paragraph boundaries are kept in both languages.

2.2.3. Text Cleaning and Formatting

We identified and removed any notes translators may have made, in order to keep the parallel texts as comparable as possible. We also deleted documents that are not detected as Arabic contents by a python library `langdetect` (1.0.7).[8] Finally, we took the intersection of the parallel documents in both languages, yielding 8,652 document pairs. All documents are segmented into sentences by `Pragmatic Segmenter` (0.3.16),[9] a rule-based sentence splitter. For Arabic, we used full stop, exclamation mark, and question mark for the set of delimiters. For Japanese, we used the default set of delimiters defined in the segmenter. The document-level aligned corpus is available in XML with UTF-8 encodings as shown in Figure 2.

[5]The agencies are: Al-Ahram, Al-Hayat, Al-Nahar, Al-Quds Al-Arabi, Al-Sabah Al-Jadid, and Al-Watan.

[6]`http://www.el.tufs.ac.jp/prmeis/`

[7]The decrease in the number of Arabic documents is due to the absence of valid links to the original ones or conversion error from MHTML to HTML.

[8]`https://pypi.python.org/pypi/langdetect/`

[9]`https://github.com/diasks2/pragmatic_segmenter/`

<?xml version="1.0" encoding="UTF-8" ?>
<body>
<meta>
<article_id>News20120225_100246</article_id>
<agency>Al-Hayat</agency>
<lang>Arabic</lang>
<category>Economy</category>
</meta>
<content>
<title>
<t id="1">500 بليون دولار ثروات النساء العربيات</t>
</title>
<dateline>دبي - دلال أبو غزالة</dateline>
<text>

<s id="1:1">سجّلت النساء الخليجيات حضوراً قوياً في سوق العمل خلال السنوات الأخيرة، وزاد عددهن إلى 3.3 مليون، مقارنة بـ1.5 مليون في العقد الأول من القرن الـ21، بزيادة 83 في المئة.</s>

<s id="2:1">ووفقاً لـ «مجموعة بوسطن للاستشارات» فإن حجم الثروة المتركزة لدى النساء في المنطقة العربية تصل إلى 500 بليون دولار، بينما تقدر مجلة «ميد» حجم الثروة لدى النساء الخليجيات بـ385 بليوناً.</s>

<s id="3:1">وأكدت مؤسسة «الماسة كابيتال» في تقرير صدر أمس، أن ما يزيد على 26 في المئة من النساء في المنطقة العربية دخلن إلى سوق العمل، علماً أن النساء يشكلن 41 في المئة من تعداد السكان.</s>

<s id="4:1">وتستثمر هذه الثروات عادة في الأصول الآمنة كالسندات والودائع النقدية والمصرفية، ما دفع بعض المؤسسات المالية إلى اتخاذ خطوات جدية للاستفادة من هذا النهج.</s>
<s id="4:2">وكرست المؤسسات المالية هذه الأموال والاستفادة منها من خلال تأسيس فروع مصرفية خاصة بالسيدات، وتأسيس صناديق موجّهة إليهن.</s>

</text>
</content>
</body>

<?xml version="1.0" encoding="UTF-8" ?>
<body>
<meta>
<article_id>News20120225_100246</article_id>
<agency>Al-Hayat</agency>
<lang>Japanese</lang>
<category>Economy</category>
</meta>
<content>
<title>
<t id="1">■アラブ人女性の総資産、5000億ドルへ</t>
</title>
<dateline>ドバイ：ダラール・アブー・ガザーラ
</dateline>
<text>

<s id="1:1">湾岸諸国の女性たちは、近年の労働市場で強い存在感を示してきた。</s>
<s id="1:2">労働者数は、21世紀初めの10年では150万人だったのに比べ、83%増の330万人に増加した。</s>

<s id="2:1">「ボストンコンサルタントグループ」によると、アラブ地域の女性たちが保有する資産規模は、5,000億ドルに達する。</s>
<s id="2:2">一方『ミード』誌は、湾岸諸国の女性たちが保有する資産規模を3850億ドルと推計する。</s>

<omit>（後略）</omit>
</text>
</content>
</body>

Figure 2: An example of a parallel document pair in XML format.

2.2.4. Manual Sentence Alignment for Evaluation

We manually aligned the latest 900 documents in publication date for evaluation purposes. We divided 900 documents into three divisions for blind-test, dev-test, and dev-tune sets. The divisions are as follows: The latest 400 documents for the blind-test set, the second latest 400 documents for the dev-test set, and the third latest 100 documents for the dev-tune set. We used the InterText tool (Vondřička, 2014) to create alignment files.

Arabic-to-Japanese	Sentence Pairs	Percentage
1-to-0	7,624	58.75
1-to-1	2,758	21.25
1-to-many	2,328	17.94
0-to-1	102	0.79
many-to-1	83	0.64
many-to-many	81	0.62

Table 2: Types of sentence alignment pairs in manually aligned data set of 900 documents.

Table 2 shows the distribution of types of sentence alignment pairs in our manually aligned data set. The possible combinations of sentence alignment pairs are as follows: One sentence in one language corresponds to one sentence in another (1-to-1), one sentence does not have corresponding sentence (1-to-0, 0-to-1), one sentence corresponds to multiple sentences (1-to-many, many-to-1), and multiple sentences correspond to multiple sentences (many-to-many). Three documents were not aligned in the document level due to the modification in the original article after translation.

The large number of 1-to-0 alignments is due to the extra paragraphs in the Arabic side that are not translated into Japanese. Apart from the null alignments (1-to-0, 0-to-1), 1-to-1 alignments account for 52.53%, whereas 1-to-many ones account for 44.34%. This can be attributed to the difference between Arabic and Japanese in punctuation usage and stylistic preference in the translation process.

2.2.5. Automatic Sentence Alignment

We compare three different alignment tools, a python implementation (Tan and Bond, 2014) of the algorithm of Gale and Church (1993), HunAlign (Varga et al., 2005), and Gargantua (Braune and Fraser, 2010).

Preprocessing We lemmatized both Arabic and Japanese texts before running a sentence aligner. We used the MADAMIRA toolkit (Pasha et al., 2014) for Arabic and MeCab (0.996) (Kudo, 2005) with IPAdic for Japanese. We performed NFKC normalization before lemmatizing Japanese tokens.

We deleted untranslated paragraphs in the Arabic side so that the number of paragraphs should be the same in both documents. This process is done only for the documents with an explicit markup that denotes omission in the latter part of a Japanese document. We used paragraph boundaries as a hard delimiter.

HunAlign To employ HunAlign, we use an approach similar to the three-step workflow used in the JRC-Arcquis corpus (Steinberger et al., 2016) and DCEP corpus (Hajlaoui et al., 2014), which consists of an initial alignment using length similarity, automatic dictionary construction from the initial alignments, and a second alignment using lexical similarity calculated with the constructed dictionary in the second step. Specifically, we first run HunAlign to obtain the initial alignment without dictionary, randomly sample 10,000 sentence pairs from the 1-to-1 segments in the initial alignment, build a dictionary with minimum occurrence score of 2 and minimum association score of 0.2, and finally, re-align all sentences with the constructed dictionary.

Evalutation We evaluate the quality of sentence alignment using the dev-tune set. We measure precision, recall, and F_1 scores in the sentence level. Precision is defined as the ratio of the number of correctly aligned pairs divided by the number of predicted pairs. Recall is defined as the ratio of the number of correctly aligned pairs divided by the number of reference pairs. F_1 is defined as the harmonic mean of precision and recall.

Results Table 3 shows the performance of the three alignment algorithms. The low F_1 score of the Gale and Church (1993) algorithm can be attributed to the distribution of alignment types and the imperfect alignment in the paragraph level. HunAlign and Gargantua outperformed the Gale and Church (1993) algorithm by large and yielded comparable results.

Alignment Algorithm	P	R	F_1
Gale and Church (1993)	0.49	0.45	0.47
HunAlign	0.74	0.77	0.76
Gargantua	**0.76**	**0.80**	**0.78**

Table 3: Sentence alignment precision (P), recall (R), and F_1 scores on dev-tune set.

2.3. Corpus Statistics

In Table 4, we provide the distribution of the categories in our corpus as determined by translators. Table 5 shows the basic statistics of sentence-level parallel corpus, including manually aligned sentences and automatically aligned sentences using Gargantua. A large difference in the number of tokens can be attributed to the difference in their tokenization schemes. We segment Japanese tokens in a more fine-grained manner than we segment Arabic tokens. In Modern Standard Arabic, an orthographically single token can have up to four syntactically independent clitics around the stem. If we were to impose D3 tokenization on the Arabic, a scheme which separates all clitics, then we would have a unit much more comparable to Japanese tokens. We ran MADAMIRA on our corpus to obtain the number of D3-tokenized tokens, which was approximately 3.4 million. This is much closer to the number of Japanese tokens, 3.7 million.

Category	Documents	Percentage
Politics	4,253	49.16
International	1,854	21.43
Society	811	9.37
Economy	608	7.03
Column	330	3.81
Lebanon Issue	280	3.24
Culture	244	2.82
Accident	194	2.24
Sports	48	0.55
Others	15	0.17
Nuclear Issue	10	0.12
Book Introduction	5	0.06
Total	8,652	100.00

Table 4: Category distribution of our entire corpus.

3. Machine Translation Baselines

In this section, we present the baseline results of phrase-based MT from Arabic to Japanese.

3.1. Experimental Settings

Phrase-based MT Settings We use the Moses toolkit (Koehn et al., 2007) to build a standard phrase-based MT system. Word alignment was extracted by MGIZA++ (Gao and Vogel, 2008) with a maximum phrase size of 8. We use the *grow-diag-final-and* and *msd-bidirectional-fe* options for symmetrization and reordering. We train a 5-gram language model on the target side of the training set using KenLM (Heafield, 2011). We use MERT (Och, 2003) for decoding weight optimization.

Data and Preprocessing We use the manually aligned data described in Section 2.2.4. for tuning and testing, and the automatically aligned data using Gargantua described in Section 2.2.5. for training.

We tokenize Arabic data using the MADAMIRA toolkit (Pasha et al., 2014) with six tokenization schemes (D0, D1, D2, D3, D3*, and ATB) following Zalmout and Habash (2017). Examples of the six tokenization schemes are shown in Table 1.

We normalize Japanese texts using the NFKC normalization and tokenize them using the MeCab morphological analyzer (0.996) (Kudo, 2005) with IPAdic.

We eliminate long sentences with more than 100 words using the script `clean-corpus-n.perl` before training translation models. Table 6 shows statistics of training data after cleaning.

Evaluation Before evaluating, we de-tokenize the predicted output by deleting spaces between Japanese characters, and then re-tokenize them using MeCab with IPAdic. We calculate automatic evaluation scores for two metrics: BLEU (Papineni et al., 2002) and RIBES (Isozaki et al., 2010). We use the `multi-bleu.perl` script in the Moses toolkit to compute BLEU scores. We calculate RIBES scores using the `RIBES.py` (1.03.1.).[10]

[10] `http://www.kecl.ntt.co.jp/icl/lirg/ribes/`

	Documents	Sentences	Tokens (ar)	Tokens (ja)
dev-tune	100	621	23,312	36,595
dev-test	400	2,393	92,760	147,536
blind-test	400	2,236	85,940	144,358
train	7,752	59,238	2,175,438	3,403,244
Total	8,652	64,488	2,377,460	3,731,733

Table 5: The basic statistics of our parallel corpus. Sentences in the training set are aligned using Gargantua.

Tokenization	Sentences	Tokens (ar)	Tokens (ja)
D0	54,223	1,811,540	2,792,333
D1	54,123	1,939,953	2,785,051
D2	54,029	2,027,916	2,778,315
ATB	53,933	2,107,024	2,771,255
D3	53,164	2,467,747	2,715,485
D3*	53,933	2,107,024	2,771,255

Table 6: The statistics of cleaned corpus for training translation models.

Results Table 7 summarizes the baseline results of phrase-based MT systems across six different tokenization schemes. The D3* scheme performs the best in both BLEU and RIBES scores, followed by the ATB scheme. The result is consistent with Zalmout and Habash (2017), where they show that removing the definite article (ال *Al* 'the') in the Arabic side enhances the performance when translating into the languages without its clear equivalent, Russian and Chinese in their case. This result is understandable since Japanese also lacks the definite article.

	dev-test		blind-test	
Tokenization	BLEU	RIBES	BLEU	RIBES
D0	10.78	56.61	8.76	55.71
D1	10.70	56.90	8.83	55.63
D2	11.13	56.94	9.34	56.08
ATB	11.29	57.54	9.24	56.41
D3	10.53	56.80	8.56	55.77
D3*	**11.48**	**57.86**	**9.38**	**56.63**

Table 7: BLEU and RIBES scores of Arabic–Japanese PBMT systems with different tokenization schemes in the source side.

4. Related Work

Much work has been done on building multilingual parallel corpora which include the language pair of Arabic and Japanese. Table 8 summarizes the statistics of publicly available parallel corpora of this language pair. Lison and Tiedemann (2016) presents the largest corpus, in which they collected movie and TV subtitles from OpenSubtitles.[11] Cettolo and Girardi (2012) constructed a parallel corpus that consists of transcribed and translated TED talks. Abdelali et al. (2014) developed the AMARA corpus that includes subtitles of educational video lectures on Massive Online Open Courses (MOOCs). Christodouloupoulos and Steedman (2015) presents a collection of Bible translations across 100 languages. Tiedemann (2012) provides a collection of Quran translations (Tanzil), localization files of technical manuals (GNOME, Ubuntu, and KDE4), as well as the collections of translations in the news domain (Global Voices, Tatoeba, News-Commentary 11). Prokopidis et al. (2016) constructed parallel corpora from Global Voices similar to Tiedemann (2012).

Compared to the domains such as subtitles, religious texts, and technical manuals, the amount of data in the news domain is very limited. Our corpus aims to supplement the lack of parallel data in this domain by constructing a parallel corpus with over 64,000 sentences (2.4 million Arabic tokens and 3.7 million Japanese tokens), including manually aligned sentence pairs for development and evaluation.

5. Conclusion and Future Work

We presented a parallel corpus of Arabic–Japanese news articles comprising 8,652 document pairs. Part of the corpus is manually aligned at the sentence level for development and testing. The corpus is provided in two formats: (a) A document-level parallel corpus with genre annotation in XML, and (b) a sentence-level parallel corpus in plain text format. The sentence-level parallel corpus comprises 64,488 sentence pairs with approximately 2.4 million Arabic tokens and 3.7 million Japanese tokens. We also reported the first results of Arabic–Japanese phrase-based MT trained on our corpus.

As future work, we will explore sentence alignment methods to improve the quality of our corpus. We also plan to explore MT techniques for under-resourced language pairs such as pivoting, and domain adaptation from better resourced domains.

6. Acknowledgements

This work has been partially funded by the Tobitate! (Leap for Tomorrow) Young Ambassador Program. Part of the work was done during the first author's visit to New York University Abu Dhabi. The creation of translated articles has been carried out at Tokyo University of Foreign Studies. We are grateful to the contributors of the TUFS Media Project for providing translated articles. We thank the anonymous reviewers, Nasser Zalmout, Alexander Erdmann, Salam Khalifa, and Ossama Obeid for their helpful comments.

[11] http://www.opensubtitles.org/

	Sentences	Tokens (ar)	Tokens (ja)	Domain
OpenSubtitiles2018 (Lison and Tiedemann, 2016)	1,834,940	11,615,534	13,319,298	Subtitle
TED (Cettolo and Girardi, 2012)	205,734	1,426,132	1,857,188	Subtitle
AMARA (Abdelali et al., 2014)	46,457	334,890	486,229	Subtitle
Bible (Christodouloupoulos and Steedman, 2015)	31,067	473,002	1,107,641	Religious
Tanzil (Tiedemann, 2012)	12,471	526,469	526,913	Religious
KDE4 (Tiedemann, 2012)	100,967	552,178	931,438	Technical Manual
Ubuntu (Tiedemann, 2012)	740	4,152	6,272	Technical Manual
GNOME (Tiedemann, 2012)	450	1,247	1,381	Technical Manual
Global Voices (Tiedemann, 2012)	4,929	85,961	121,234	News
Global Voices (Prokopidis et al., 2016)	7,211	127,737	200,215	News
Tatoeba (Tiedemann, 2012)	1,134	6,039	10,947	News
News-Commentary11 (Tiedemann, 2012)	569	39,937	52,085	News
Our Corpus	64,488	2,377,460	3,731,733	News

Table 8: Statistics of publicly available parallel corpora of Arabic and Japanese.

7. Bibliographical References

Abdelali, A., Guzman, F., Sajjad, H., and Vogel, S. (2014). The AMARA Corpus: Building Parallel Language Resources for the Educational Domain. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Asahara, M. and Matsumoto, Y., (2003). *ipadic version 2.7.0 User's Manual (in Japanese)*. Nara Institute of Science and Technology.

Bond, F. and Baldwin, T. (2016). Introduction to Japanese Computational Linguistics. In Kentaro Inui Shun Ishizaki Hiroshi Nakagawa Francis Bond, Timothy Baldwin et al., editors, *Readings in Japanese Natural Language Processing*, CSLI Studies in Computational Linguistics, pages 1–28. CSLI Publications.

Braune, F. and Fraser, A. (2010). Improved Unsupervised Sentence Alignment for Symmetrical and Asymmetrical Parallel Corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 81–89. Association for Computational Linguistics.

Cettolo, M. and Girardi, C. (2012). WIT^3: Web Inventory of Transcribed and Translated Talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy, May.

Christodouloupoulos, C. and Steedman, M. (2015). A massively parallel corpus: the Bible in 100 languages. *Language Resources and Evaluation*, 49(2):375–395, Jun.

Gale, W. A. and Church, K. W. (1993). A Program for Aligning Sentences in Bilingual Corpora. *Computational linguistics*, 19(1):75–102.

Gao, Q. and Vogel, S. (2008). Parallel Implementations of Word Alignment Tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57. Association for Computational Linguistics.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic Transliteration. In A. van den Bosch et al., editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*, volume 3 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.

Hajlaoui, N., Kolovratnik, D., Väyrynen, J., Steinberger, R., and Varga, D. (2014). DCEP – Digital Corpus of the European Parliament. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Heafield, K. (2011). KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Isozaki, H., Hirao, T., Duh, K., Sudoh, K., and Tsukada, H. (2010). Automatic Evaluation of Translation Quality for Distant Language Pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, Cambridge, MA, October. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics.

Kudo, T. (2005). MeCab: Yet Another Part-of-Speech and Morphological Analyzer. *http://mecab. sourceforge. net/*.

Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Och, F. J. (2003). Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Pasha, A., Al-Badrashiny, M., Diab, M. T., El Kholy, A., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, volume 14, pages 1094–1101, Reykjavik, Iceland.

Prokopidis, P., Papavassiliou, V., and Piperidis, S. (2016). Parallel Global Voices: a Collection of Multilingual Corpora with Citizen Media Stories. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufis, D., and Varga, D. (2016). The JRC-Acquis: A Multilingual Aligned Parallel Corpus with 20+ Languages. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, may.

Tan, L. and Bond, F. (2014). NTU-MC Toolkit: Annotating a Linguistically Diverse Corpus. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 86–89, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012)*, volume 2012, pages 2214–2218, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., and Trón, V. (2005). Parallel Corpora for Medium Density Languages. In *Proceedings of the RANLP 2005*, pages 590–596.

Vondřička, P. (2014). Aligning Parallel Texts with InterText. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Zalmout, N. and Habash, N. (2017). Optimizing Tokenization Choice for Machine Translation across Multiple Target Languages. *The Prague Bulletin of Mathematical Linguistics*, 108(1):257–269.

Examining the Tip of the Iceberg: A Data Set for Idiom Translation

Marzieh Fadaee[1], Arianna Bisazza[2], Christof Monz[1]
[1]Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands
[2]Leiden Institute of Advanced Computer Science, Leiden University, 2333 CA Leiden, The Netherlands
{m.fadaee,c.monz}@uva.nl
a.bisazza@liacs.leidenuniv.nl

Abstract

Neural Machine Translation (NMT) has been widely used in recent years with significant improvements for many language pairs. Although state-of-the-art NMT systems are generating progressively better translations, idiom translation remains one of the open challenges in this field. Idioms, a category of multiword expressions, are an interesting language phenomenon where the overall meaning of the expression cannot be composed from the meanings of its parts. A first important challenge is the lack of dedicated data sets for learning and evaluating idiom translation. In this paper we address this problem by creating the first large-scale data set for idiom translation. Our data set is automatically extracted from a widely used German↔English translation corpus and includes, for each language direction, a targeted evaluation set where all sentences contain idioms and a regular training corpus where sentences including idioms are marked. We release this data set and use it to perform preliminary NMT experiments as the first step towards better idiom translation.

Keywords: multiword expression, idioms, bilingual corpora, machine translation

1. Introduction

Neural Machine Translation (NMT) (Bahdanau et al., 2015; Sutskever et al., 2014; Cho et al., 2014) has achieved substantial improvements in translation quality over traditional Rule-based and Phrase-based Translation (PBMT) in recent years. For instance, subject-verb agreement, double-object verbs, and overlapping subcategorization are various areas where NMT successfully overcomes the limitations of PBMT (Isabelle et al., 2017; Bentivogli et al., 2016). However, one of the remaining challenges of NMT is translating infrequent words and phrases (Koehn and Knowles, 2017; Fadaee et al., 2017) and idioms are a particular instance of this problem (Isabelle et al., 2017).

Idioms are semantic lexical units whose meaning is often not simply a function of the meaning of its constituent parts (Nunberg et al., 1994; Kövecses and Szabo, 1996). The non-compositionality characteristic of idiom expressions exists in different degrees in a language (Nunberg et al., 1994). In English for example, for the idiom *"spill the beans"*, the word *'spill'* symbolizes *'reveal'* and *'beans'* symbolizes the *'secrets'*. With the idiomatic expression *"kick the bucket"*, on the other hand, no such analysis is possible.

Isabelle et al. (2017) builds a challenge set of 108 short sentences that each focus on one particular difficult phenomenon of the language. Their manual assessment of the eight sentences consisting of an idiomatic phrase show that NMT systems struggle with the translation of these phrases.

The challenge of translating idiom phrases in NMT is partly due to the underlying complexity of identifying a phrase as idiomatic and generating its correct non-literal translation, and partly to the fact that idioms are rarely encountered in the standard data sets used for training NMT systems.

As an example, in Table 1 we provide an idiom expression in German and the literal and idiomatic translations in English. We observe that the literal translation of an idiom is not the correct translation, neither does it capture part of the meaning.

German phrase	*eine weiße Weste haben*
Literal translation	to have a white vest
Idiomatic translation	to have clean slate
Sentence	Coca-Cola und Nestlé gehören zu den Unterzeichnern. Beide **haben** nicht gerade **eine weiße Weste**.
Reference translation	Coca Cola and Nestlé are two signatories with *"spotty" track records*.
DeepL	Coca-Cola and Nestlé are among the signatories. Neither of them is **exactly the same**.
GoogleNMT	Coca-Cola and Nestlé are among the signatories. Both do not **have just a white vest**.
OpenNMT	Coca-Cola and Nestlé are among the signatories. Both don't **have a white essence**.

Table 1: Example of an idiom phrase in German and its translation. We compare the output of DeepL, GoogleNMT, and OpenNMT translating a sentence with this idiom phrase and notice that none capture the idiom translation correctly.

To illustrate the problem of idiom translation we also provide the output of three NMT systems for this sentence: GoogleNMT (Wu et al., 2016), DeepL[1], and the OpenNMT implementation (Klein et al., 2017) based on Bahdanau et al. (2015) and Luong et al. (2015). All systems fail to generate the proper translation of the idiom expression. This problem is particularly pronounced when the source idiom is very different from its equivalent in the target language, as the case here.

[1]www.deepl.com/translator

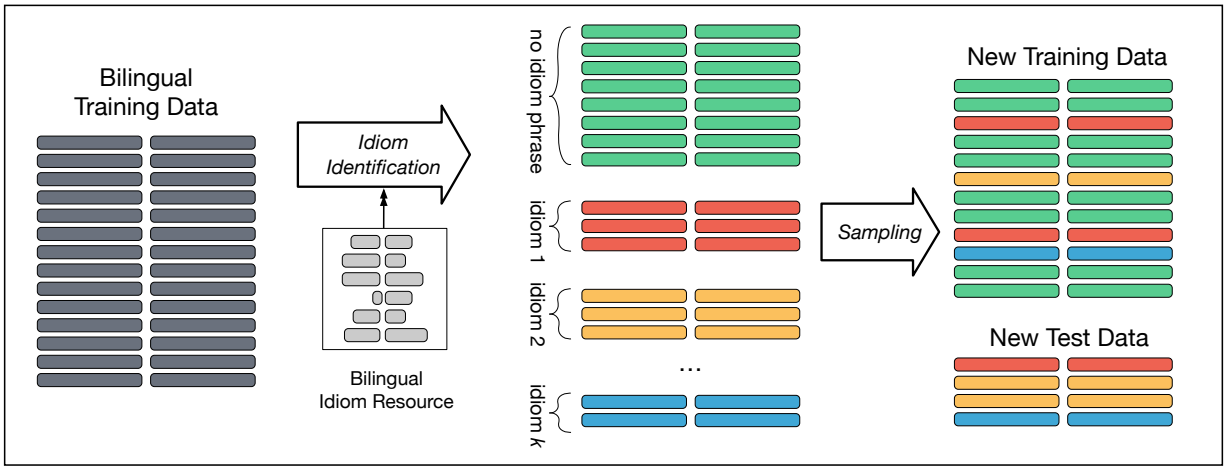


Figure 1: The process of data collection and construction of the test set containing only sentence pairs with idiom phrases.

Although there are a number of monolingual data sets available for identifying idiom expressions (Muzny and Zettlemoyer, 2013; Markantonatou et al., 2017), there is limited work on building a parallel corpus annotated with idioms, which is necessary to investigate this problem more systematically. Salton et al. (2014) selected a small subset of 17 English idioms, collected 10 sentence examples for each idiom from the internet, and manually translated them into Brazilian-Portuguese to use for the translation task.

Building a hand-crafted data set for idiom translation is costly and time-consuming. In this paper we automatically build a new bilingual data set for idiom translation extracted from an existing general-purpose German↔English parallel corpus.

The first part of our data set consists of 1,500 parallel sentences whose German side contains an idiom, while the second consists of 1,500 parallel sentences whose English side contains an idiom. Additionally, we provide the corresponding training data sets for German→English and English→German translation where source sentences including an idiom phrase are marked. We believe that having a sizable data set for training and evaluation is the first step to improve idiom translation.

German idiom translation data set	
Number of unique idioms	103
Training size	4.5M
Idiomatic sentences in training data	1848
Test size	1500
English idiom translation data set	
Number of unique idioms	132
Training size	4.5M
Idiomatic sentences in training data	1998
Test size	1500

Table 2: Statistics of the German and English idiom translation data sets. Sentence pairs are counted on the training and test sets.

2. Data Collection

In this work we focus on German↔English translation of idioms. This is an established language pair and is commonly used in the machine translation community.
Automatically identifying idiom phrases in a parallel corpus requires a gold standard data set annotated manually by linguists. We use the `dict.cc` online dictionary[2] containing idiomatic and colloquial phrases, which is built manually, as our gold standard for extracting idiom phrase pairs. Examining the WMT German↔English test sets from 2008 to 2016 (Bojar et al., 2017), we observe very few sentence pairs containing an idiomatic expression. The standard parallel corpora available for training however contain several such sentence pairs. Therefore we automatically select sentence pairs from the training corpora where the source sentence contains an idiom phrase to build the new test set.
Note that we only focus on idioms on the source side and we have two separate list of idioms for German and English, hence, we independently build two test sets (for German idiom translation and English idiom translation) with different sentence pairs selected from the parallel corpora.

German idiom	*alles über einen kamm scheren*
English equivalent	to measure everything by the same yardstick
Matching German sentence	Aber man kann eben nicht **alle** Inseln **über einen Kamm scheren**.
English translation	But we cannot measure everyone by the same standards.
German idiom	*in den kinderschuhen stecken*
English equivalent	to be in the fledgling stage
Matching German sentence	Es **steckt** immer noch **in den Kinderschuhen**.
English translation	It is still in its infancy.

Table 3: Two examples displaying different constraints of matching an idiom phrase with occurrences in the sentence.

Depending on the language, the words making up an idiomatic phrase are not always contiguous in the sentence.

[2] `www.dict.cc`

German idiom	*in den kinderschuhen stecken*
English equivalent	to be in the fledgling stage
German sentence	Eine Bemerkung, Gentoo/FreeBSD **steckt** noch **in den Kinderschuhen** und ist kein auf Sicherheit achtendes System.
English sentence	Note that Gentoo/FreeBSD is still **in its infancy** and is not a security supported platform.
German idiom	*den kreis schließen*
English equivalent	to bring sth. full circle
German sentence	Die europäische Krise **schließt den Kreis**.
English sentence	The European crisis is **coming full circle**.
German idiom	*auf biegen und brechen*
English equivalent	by hook or crook
German sentence	Nehmen wir zum Beispiel die Währungsunion: Sie soll **auf Biegen und Brechen** eingeführt werden.
English sentence	Take, for example, the introduction -**come what may**- of the single currency.
German idiom	*sie haben das wort*
English equivalent	the floor is yours
German sentence	Berichterstatterin. - (FR) Herr Präsident! Danke, dass **Sie** mir **das Wort** erteilt **haben**.
English sentence	rapporteur. - (FR) Mr President, thank you for **giving** me **the floor**.

Table 4: Examples from the German idiom translation test set.

For instance, in German, the subject can appear between the verb and the prepositional phrase making up the idiom. German also allows for several re-orderings of the phrase.

In order to generalize the process of identifying idiom occurrences, we lemmatize the phrases and consider different re-ordering of the words in the phrase as an acceptable match. We also allow for a fixed number of words to occur in between the words of an idiomatic phrase. Table 3 shows two examples of idiom occurrences that match these criteria.

Following this set of rules, we extract sentence pairs containing idiomatic phrases, and create a set of sentence pairs for each unique idiom phrase. In the next step we sample without replacement from these sets and select individual sentence pairs to build the test set.

In order to build the new training data, we use the remaining sentence pairs in each idiom set as well as the sentence pairs from the original parallel corpora that did not include any idiom phrases. In this process, we ensure that for each idiomatic expression there is at least one occurrence in both training and test data, and that no sentence is included in both training and test data.

Figure 1 visualizes the process of constructing the new training and test sets. As a result, for each language direction, we obtain a targeted test set for idiom translation and the corresponding training corpus representing a natural distribution of sentences with and without idioms.

We annotate each sentence pair with the canonical form of its source-side idiom phrase and its equivalent in the target language.

Table 2 provides some statistics of the two data sets. For each unique idiom in the test set, we also provide the frequency of the respective idiom in the training data. Note that this is based on the lemmatized idiom phrase under the constraints mentioned in Section 2. and is not necessarily an exact match of the phrase.

Table 4 shows several examples from the data set for German idiom translation. We observe that for some idioms the literal translation in the target language is close to the actual meaning, while for others it is not the case.

One side effect of automatically identifying idiom expressions in sentences is that it is not always accurate. Sentence pairs where an idiom expression was used as a literal phrase (e.g., *"spill the beans"* to literally describe the act of *spilling the beans*) will be identified as idiomatic sentences.

3. Translation Experiments

While the main focus of this work is to generate data sets for training and evaluating idiom translation, we also perform a number of preliminary NMT experiments using our data set to measure the problem of idiom translation on large scale data.

In the first experiment following the conventional settings, we do not use any labels in the data to train the translation model. In the second experiment we use the labels in the training data as an additional feature to investigate the effect of informing the model of the existence of an idiomatic phrase in a sentence during training.

We perform a German→English experiment by providing the model with additional input features. The additional features indicate whether a source sentence contains an idiom and are implemented as a special extra token `<idm>` that is prepended to each source sentence containing an idiom. This a simple approach that can be applied to any sequence-to-sequence architecture.

Most NMT systems have a sequence-to-sequence architecture where an encoder builds up a representation of the source sentence and a decoder, using the previous LSTM hidden states and an attention mechanism, generates the target translation (Bahdanau et al., 2015; Sutskever et al., 2014; Cho et al., 2014). We use a 4-layer attention-based encoder-decoder model as described in (Luong et al., 2015) trained with hidden dimension size of 1,000, and batch size of 80 for 20 epochs.

	WMT test sets 2008-2016	Idiom test set		
Model	BLEU	BLEU	Unigram Precision	Word-level Accuracy
PBMT Baseline	20.2	19.7	57.7	71.6
NMT Baseline	26.9	24.8	53.2	67.8
NMT <idm> token on source	25.2	22.5	64.1	73.2

Table 5: Translation performance on German idiom translation test set. *Word-level Idiom Accuracy* and *Unigram Precision* are computed only on the idiom phrase and its corresponding translation in the sentence.

In all experiments the NMT vocabulary is limited to the most common 30K words in both languages and we preprocess source and target language data with Byte-pair encoding (BPE) (Sennrich et al., 2016) using 30K merge operations.

We also use a Phrase-based translation system similar to Moses (Koehn et al., 2007) as baseline to explore PBMT performance for idiom translation.

4. Idiom Translation Evaluation

Ideally idiom translation should be evaluated manually, but this is a very costly process. Automatic metrics, on the other hand, can be used on large data sets at no cost and have the advantage of replicability.

We use the following metrics to evaluate the translation quality with a specific focus on idiom translation accuracy:

BLEU The traditional BLEU score (Papineni et al., 2002) is a good measure to determine the overall quality of the translations. However this measure considers the precision of all n-grams in a sentence and by itself does not focus on the translation quality of the idiomatic expressions.

Modified Unigram Precision To specifically concentrate on the quality of the translation of idiom expressions, we also look at the *localized* precision. In this approach we translate the idiomatic expression in the context of a sentence, and only evaluate the translation quality of the idiom phrase.

To isolate the idiom translation in the sentence, we look at the word-level alignments between the idiom expression in the source sentence and the generated translation in the target sentence. We use `fast-align` (Dyer et al., 2013) to extract word alignments. Since idiomatic phrases and the respective translations are not contiguous in many cases we only compare the unigrams of the two phrases.

Note that for this metric we have two references: The idiom translation as an independent expression, and the human generated idiom translation in the target sentence.

Word-level Idiom Accuracy We also use another metric to evaluate the word-level translation accuracy of the idiom phrase. We use word alignments between source and target sentences to determine the number of correctly translated words. We use the following equation to compute the accuracy:

$$WIAcc = \frac{H - I}{N}$$

where H is the number of correctly translated words, I is the number of extra words in the idiom translation, and N is the number of words in the gold idiom expression.

Table 5 presents the results for the translation task using different metrics. Looking at the overall BLEU scores, we observe that baseline performance on the idiom-specific test set is lower than on the union of the standard test sets (WMT 2008-2016). While the scores on these two data sets are not directly comparable, this result is in line with previous findings that sentences containing idiomatic expressions are harder to translate (Isabelle et al., 2017). We can also see that the performance gap is not as pronounced for PBMT systems, suggesting that phrase-based models are capable of *memorizing* the idiom phrases to some extent.

The NMT experiment using a special input token to indicate the presence of an idiom in the sentence performs still better than PBMT but slightly worse than the NMT baseline in terms of BLEU. Despite this drop in BLEU performance, by examining the *unigram precision* and *word-level idiom accuracy* scores, we observe that this model generates more accurate idiom translations.

These preliminary experiments reiterate the problem of idiom translation with neural models, and in addition show that with a labeled data set, we can devise simple models to address this problem to some extent.

5. Conclusion

Idiom translation is one of the more difficult challenges of machine translation. Neural MT in particular has been shown to perform poorly on idiom translation despite its overall strong advantage over previous MT paradigms (Isabelle et al., 2017). As a first step towards a better understanding of this problem, we have presented a parallel data set for training and testing idiom translation for German→English and English→German.

The test sets include sentences with at least one idiom on the source side while the training data is a mixture of idiomatic and non-idiomatic sentences with labels to distinguish between the two. We also performed preliminary translation experiments and proposed different metrics to evaluate idiom translation.

We release new data sets which can be used to further investigate and improve NMT performance in idiom translation.

Acknowledgments

This research was funded in part by the Netherlands Organization for Scientific Research (NWO) under project numbers 639.022.213 and 639.021.646, and a Google Faculty Research Award. We also thank NVIDIA for their hardware support.

6. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Bentivogli, L., Bisazza, A., Cettolo, M., and Federico, M. (2016). Neural versus phrase-based machine translation quality: a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 257–267, Austin, Texas, November. Association for Computational Linguistics.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi, M. (2017). Findings of the 2017 conference on machine translation (WMT17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark, September. Association for Computational Linguistics.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. In *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8), 2014*.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia, June. Association for Computational Linguistics.

Fadaee, M., Bisazza, A., and Monz, C. (2017). Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573, Vancouver, Canada, July. Association for Computational Linguistics.

Isabelle, P., Cherry, C., and Foster, G. (2017). A challenge set approach to evaluating machine translation. *arXiv preprint arXiv:1704.07431*.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Kövecses, Z. and Szabo, P. (1996). Idioms: A view from cognitive semantics. *Applied Linguistics*, 17(3):326–355.

Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September. Association for Computational Linguistics.

Markantonatou, S., Ramisch, C., Savary, A., and Vincze, V. (2017). Proceedings of the 13th workshop on multiword expressions (mwe 2017). In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*.

Muzny, G. and Zettlemoyer, L. (2013). Automatic idiom identification in wiktionary. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1417–1421.

Nunberg, G., Sag, I. A., and Wasow, T. (1994). Idioms. *Language*, 70(3):491–538.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Salton, G., Ross, R., and Kelleher, J. (2014). An empirical study of the impact of idioms on phrase based statistical machine translation of english to brazilian-portuguese. In *Proceedings of the 3rd Workshop on Hybrid Approaches to Machine Translation (HyTra)*, pages 36–41, Gothenburg, Sweden, April. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Z. Ghahramani, et al., editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Automatic Enrichment of Terminological Resources: the IATE RDF Example

Mihael Arcan[‡], Elena Montiel-Ponsoda[†], John P. McCrae[‡], Paul Buitelaar[‡]
[‡]Insight Centre for Data Analytics, National University of Ireland Galway
[firstname.lastname]@insight-centre.org
[†] Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain
emontiel@fi.upm.es

Abstract

Terminological resources have proven necessary in many organizations and institutions to ensure communication between experts. However, the maintenance of these resources is a very time-consuming and expensive process. Therefore, the work described in this contribution aims to automate the maintenance process of such resources. As an example, we demonstrate enriching the RDF version of IATE with new terms in the languages for which no translation was available, as well as with domain-disambiguated sentences and information about usage frequency. This is achieved by relying on machine translation trained on parallel corpora that contains the terms in question and multilingual word sense disambiguation performed on the context provided by the sentences. Our results show that for most languages translating the terms within a disambiguated context significantly outperforms the approach with randomly selected sentences.

Keywords: knowledge bases, terminology, multilinguality, machine translation

1. Motivation

Terminological resources have proven necessary in many organizations and institutions to ensure the quality and consistency of the terms used across the documents that serve expert communication. In the translation field, these resources are usually integrated into Computer Aided Translation (CAT) tools to allow translators to store the terms that appear in the documents they are translating. Terms and their translations may be accompanied by additional information intended to represent the meaning and the translation decisions taken for the use of the terms. The collected information accounts for the domain of the document, the definition of source or target terms, a sentence in which the term appears or other terms with the same meaning, i.e., term variants.

One of the most representative terminological resources is the Inter-Active Terminology for Europe (IATE).[1] It is a widely-used resource which incorporates all the terminology databases that had been independently built and maintained by the translation services of the different EU institutions. The public version of IATE contains approximately 1.4 million entries (8.6 million terms) in the 24 official EU languages. Due to the vast amount of domains covered in IATE (Table 1), the resource is a point of reference not only for EU translators and interpreters but also for translators and language service providers (LSPs) around the world. A simplified version of it is available in the TBX (Term Base eXchange) format for free download,[2] and has been converted to the Resource Description Framework (Klyne and Carroll, 2006, RDF) according to the linked data principles (Cimiano et al., 2015). As for any other language resource, the efforts that are required to maintain, update and clean the data are extensive. In addition to that, some languages or domains are extensively covered, whereas others have not received so much attention. Moreover, duplicate entries are common, since each institutional database was merged together into IATE in 2004. Furthermore, some terms are accompanied by definitions, contexts, and usage notes, and are supported by valid references and high reliability values, whereas others are not further described. Finally, due to the time pressure in the translation process, occasionally terminologists include terms or provide translations that are not properly checked, and thus not agreed or verified by experts.

Taking these issues into account, we claim that automatic processes should be put in place to support professionals in the enrichment, maintenance and cleaning of such a database, so that new terms in low covered languages could be added and enriched with contextual information, whereby duplicates, low quality or old terms could be removed. In order to contribute to the enrichment of IATE with terms that do not exist in some of the languages covered by the resource, we rely on Statistical Machine Translation (SMT) and leverage available translations in other languages, to identify relevant domain-specific sentences that contain those potential translations from a large set of parallel corpora. Furthermore, we use the identified relevant sentences to provide additional or missing contextual information to the terminological resource. Finally, we compare the usage of term variants in certain translations pairs, which can show a distinction between commonly used and less used terms in the IATE resource.

2. Related Work

Most of the previous work on the translation of knowledge resources, i.e., ontologies, taxonomies or terminological resources, tackled this problem by accessing multilingual lexical resources, e.g. EuroWordNet or IATE (Declerck et al., 2006; Cimiano et al., 2010). Their work focuses on the identification of the lexical overlap between the terms stored in the resource to be translated and the multilingual resource. Since the replacement of source terms with their translations within the dictionaries guarantees a high pre-

[1] http://iate.europa.eu/
[2] http://iate.europa.eu/tbxPageDownload.do

Language	Domain
Lang. Indep.	health (13,338), law (9,924), agriculture, forestry and fisheries (8,546), ...
Bulgarian	natural and applied sciences (2,397), eu institutions and european civil service (2,320), chemistry (2,294), ...
Czech	natural and applied sciences (608), health (601), finance (486), ...
Danish	health (66,777), agriculture, forestry and fisheries (42,195), natural and applied sciences (38,106), ...
German	law (50,139), land transport (29,801), executive power and public service (26,398), ...
Greek	no domain (7,735), communications (3,769), information technology and data processing (3,654), ...
English	health (28,543), information technology and data processing (28,063), ...
Spanish	law (957), no domain (927), politics (717), ...
Estonian	insurance (31), natural and applied sciences (29), agricultural activity (29), ...
Finnish	no domain (438), executive power and public service (306), agriculture, forestry and fisheries (280), ...
French	executive power and public service (11,685), natural environment (10,628), health (6,468), ...
Irish	natural and applied sciences (13), natural environment (7), agriculture, forestry and fisheries (2), ...
Croatian	fisheries (17), agriculture, forestry and fisheries (17), land transport (4), ...
Hungarian	executive power and public service (14), politics (4), political party (4), ...
Italian	no domain (1,262), natural environment (590), agriculture, forestry and fisheries (410), ...
Lithuanian	political party (7), culture and religion (6), agriculture, forestry and fisheries (5), ...
Latvian	politics (10), political party (9), regions of eu member states (5), ...
Maltese	health (4), finance (4), social protection (2), ...
Dutch	executive power and public service (1,955), humanities (992), education (951), ...
Polish	health (14), air and space transport (13), communications (11), ...
Portuguese	no domain (839), employment and working conditions (160), land transport (150), ...
Romanian	european union law (11), finance (8), land transport (5), ...
Slovak	health (12), chemistry (10), industry (8), ...
Slovene	no domain (12), political party (2), politics (2), ...
Slovak	health (6), law (6), executive power and public service (4), ...

Table 1: Statistics on most defined domains in IATE.

cision but a low recall, external translation services, e.g. BabelFish, SDL FreeTranslation tool or Google Translate, were used to overcome this issue (Fu et al., 2009; Espinoza et al., 2009). BabelNet (Navigli, 2012), one of the largest multilingual knowledge bases, was created by linking Wikipedia entries and Wordnet synsets, and used commercial translation systems to fill in the missing lexical gap. Pérez and Berlanga (2015) enriched the non-English counterparts of the UMLS (Unified Medical Language System) knowledge resource in the biomedical domain. They used lexicons generated by word-alignment and machine translation approaches, and compared the results with the proposed semantics transfer approach, focusing on the semantic coherence of the generated translations between Spanish, French and German. Sajous et al. (2010) enriched Wiktionary by relying on similarity measures based on random walks through the graphs extracted from its lexical networks. In their final step they engaged users in collaborative editing in order to validate the resource. A different approach for translation and disambiguation of domain-specific expressions stored in knowledge bases was shown in Arcan et al. (2015), where the authors identified relevant in-domain parallel sentences and used them to train a small but domain-aware SMT system. Ordan et al. (2017) demonstrated an approach for bilingual dictionary creation using different translation directions within a loop. In contrast, de Melo and Weikum (2012) did not match concepts with SMT, but showed a machine learning approach which determines the best translation for English WordNet synsets by taking bilingual dictionaries, structural information of WordNet and corpus frequency information into account. Similarly, the multilingual disambiguation of ontology labels was performed by Espinoza et al. (2009) and McCrae et al. (2011), where the structure of the ontology along with existing multilingual ontologies was used to annotate the labels with their semantic senses and to link them across languages. Furthermore, McCrae et al. (2016) show positive effects of different domain adaptation techniques, i.e., using Web resources as additional bilingual knowledge, rescoring translations with Explicit Semantic Analysis (ESA) and language model adaptation for automatic knowledge base translation. To link knowledge graphs across languages, McCrae et al. (2017) propose a hybrid approach that combines dataset alignment and ontology translation techniques. The combination of these two techniques improves the translation of domain-specific expressions in comparison with approaches when used alone.

3. Methodology

We demonstrate an approach that uses the existing IATE terms to select the most relevant sentences in which those terms appear. By translating these sentences into languages for which no term is documented in the database, we obtain new translations for the available terms. Furthermore, we use these sentences to enrich IATE with additional information, i.e. how these terms appear in sentences of a representative domain as well as the usage of the translations in the parallel corpora.

Language	# of Entries	All Entries	Tokens	Types	Avg. Length	# of Unigrams
French	932,078	1,102,995	3,876,429	181,344	3.51	182,977
English	929,729	1,089,726	3,116,115	200,214	2.85	166,675
German	673,851	846,925	1,631,472	520,999	1.92	514,634
Italian	498,549	591,970	1,953,887	127,591	3.30	103,885
Dutch	489,941	590,319	1,412,233	311,212	2.39	288,864
Spanish	447,377	524,504	1,762,624	117,012	3.36	94,971
Danish	445,939	533,541	1,114,400	308,581	2.08	284,811
Greek	398,946	466,635	1,415,473	153,368	3.03	72,055
Portuguese	390,772	446,740	1,453,663	103,732	3.25	85,255
Finnish	251,544	298,639	546,823	198,219	1.83	163,764
Swedish	242,158	277,067	540,266	176,163	1.94	156,407
Irish	57,415	63,331	210,477	36,321	3.32	10,825
Polish	54,971	64,100	211,749	42,122	3.30	8,818
Slovene	43,168	49,181	150,681	32,127	3.06	8,781
Maltese	41,090	48,018	164,501	31,618	3.42	8,067
Lithuanian	38,281	43,799	134,902	28,426	3.08	6,046
Romanian	38,126	43,822	159,205	26,281	3.63	7,145
Estonian	34,957	43,735	100,196	37,303	2.29	17,406
Slovak	34,453	39,647	131,726	30,920	3.32	6,643
Hungarian	32,369	38,939	109,510	31,595	2.81	10,460
Bulgarian	31,960	36,856	135,502	24,456	3.67	5,313
Latvian	31,857	36,792	113,653	26,942	3.08	6,478
Czech	27,866	33,562	116,364	23,842	3.46	4,876
Croatian	14,635	16,440	55,139	15,003	3.35	2,317

Table 2: Statistics on covered terms in IATE.

Disambiguated Context Identification The main challenge involved in building multilingual knowledge bases, is, however, to bridge the gap between language-specific information and the language-independent semantic content (Gracia et al., 2012). Since manual multilingual translation and evaluation of knowledge bases is a very time-consuming and expensive process, we apply SMT to automatically translate domain-specific expressions and demonstrate its validity by translating the IATE entries. While an SMT system can only return the most frequent or dominant translation when given a term by itself, it has been showed that SMT provides strong word sense disambiguation when the word is given in the context of a sentence (Arcan et al., 2016a; Arcan et al., 2016b).

As a motivating example, we consider the word *vessel*, which appears several times in the IATE repository, whereby the most frequent translation into German is *Schiff*, with the meaning of 'a craft designed for water transportation', e.g., as given by Google Translate.[3] To overcome the issue of obtaining translations for *vessel* in other languages, and also in different domains (in the sense of *blood vessel*, for instance), we aim to identify (several) parallel sentences, which hold the terminological entries in the targeted domain, and use their context to translate them into other languages for which a translation does not exist. This means that if we know that the word *vessel* also represents the meaning of 'a tube in which a body fluid circulates' and the German translation for this entry is *Gefäß*, we look in our approach for sentences in a parallel corpus where the words *vessel* and *Gefäß* both occur and obtain a context such as 'blood vessel' that allows the SMT system to translate this entry correctly. Although a translation into German is not necessary in this case, since it is already documented in the database, we can use the English sentence (appropriately disambiguated) to translate the term into other languages, where the translation does not exist. To maximize our chances of finding a well-disambiguated sentence, we use existing terms in as many languages as possible.

Enhancing IATE with New Translations and Contextual Information In addition to the extension of IATE with missing translations for less covered languages, we further provide information on how the domain-specific expressions appear in sentences. Since we identified the relevant sentences by using linked IATE entries in different languages, we believe that this additional information can further enrich the terminological resource.

Frequency and Reliability of Term Variants When accessing IATE from its online interface,[4] it is common to find several translations for the same term in the same domain, the so-called term variants, and it is the user's decision to choose one variant over the others. In the same way, for each term separately, additional information about its usage or the level of confidence assigned by the creators of the resource can only be obtained in a time-consuming fashion. To overcome these issues, we use the data we obtain from the parallel corpus that is believed to represent the real use of terms.

[3] `https://translate.google.com`, September 2017

[4] `http://iate.europa.eu/SearchByQuery.do`

IATE Duplicates Due to the original merging of separately maintained databases, duplicates are common in IATE. To support the maintenance of the resource in a more efficient way, we evaluate how many IATE entries have identical terms in English and in other target languages.

4. Experimental Setting

In this section, we give an overview on the dataset and the translation toolkit used in our experiment and provide insights into the SMT evaluation techniques.

4.1. IATE - Inter-Active Terminology for Europe

IATE is the terminology database of the EU with its objective of supporting the EU translators and creating a terminology resource to ensure standardisation throughout all institutions. It incorporates the various terminology databases into one database containing approximately one million multilingual entries in English (Table 2).[5] Recent domains that have been extensively covered include the financial crisis, environment, fisheries and migration.

4.2. Statistical Machine Translation

Our approach is based on phrase-based SMT (Koehn et al., 2003), where we wish to find the best translation of a string, given by a log-linear model combining a set of features. The translation that maximizes the score of the log-linear model is obtained by searching all possible translation candidates. The decoder, which is a search procedure, provides the most probable translation based on a statistical translation model learned from the training data.

For our task, we use the statistical translation toolkit Moses (Koehn et al., 2007), where word alignments, necessary for generating translation models, were built with the GIZA++ toolkit (Och and Ney, 2003). The KenLM toolkit (Heafield, 2011) was used to build a 5-gram language model.

4.3. Parallel Resources for SMT training and Multilingual Word Sense Disambiguation

To ensure a broad lexical and domain coverage of our SMT system, we merged the existing parallel corpora for each language pair from the OPUS web page[6] into one parallel data set, i.e., Europarl (Koehn, 2005), DGT translation memories generated by the *Directorate-General for Translation* (Steinberger et al., 2014), MultiUN corpus (Eisele and Chen, 2010), EMEA, KDE4, OpenOffice (Tiedemann, 2009), OpenSubtitles2012 (Tiedemann, 2012). Similarly, we concatenated parallel corpora to identify relevant sentences containing IATE entries, which are then translated into the targeted languages. Table 3 shows the amount of parallel sentences used for the different language pairs.

4.4. Translation Evaluation Metrics

The automatic translation evaluation is based on the correspondence between the SMT output and reference translation (gold standard).

BLEU (Papineni et al., 2002) is calculated for individual translated segments (n-grams) by comparing them with a data set of reference translations. Considering the shortness of the entries in IATE, we report scores based on the unigram overlap (BLEU-1). Those scores, between 0 and 100 (perfect translation), are then averaged over the whole *evaluation data set* to reach an estimate of the translation's overall quality.

METEOR (Denkowski and Lavie, 2014) is based on the harmonic mean of precision and recall, whereby recall is weighted higher than precision. Along with exact word (or phrase) matching it uses additional features, i.e., stemming, paraphrasing and synonymy matching.

chrF3 (Popović, 2015) is a character n-gram metric, which has shown very good correlations with human judgements, especially when translating from English into morphologically rich languages (Stanojević et al., 2015).

The approximate randomization approach (Clark et al., 2011) is used to test whether differences among system performances are statistically significant.

		Tokens		Types	
Language Pair	Sent.	English	Target	English	Target
English-Bulgarian	33M	325M	279M	938k	1M
English-Czech	24M	278M	237M	1M	1M
English-Danish	17M	236M	212M	687k	1M
English-German	10M	145M	135M	561k	1M
English-Greek	34M	364M	330M	1M	1M
English-Spanish	37M	391M	378M	1M	1M
English-Estonian	15M	185M	144M	640k	1M
English-Finish	24M	293M	199M	826k	2M
English-French	53M	740M	795M	1M	1M
English-Irish	1M	14M	15M	180k	271k
English-Croatian	16M	165M	133M	626k	1M
English-Hungarian	36M	369M	298M	1M	3M
English-Italian	22M	269M	265M	934k	1M
English-Lithuanian	6M	108M	90M	421k	833k
English-Latvian	5M	102M	86M	389k	653k
English-Maltese	2M	40M	40M	218k	380k
English-Dutch	35M	394M	363M	976k	1M
English-Polish	34M	361M	295M	1M	1M
English-Portuguese	32M	369M	354M	1M	1M
English-Romanian	40M	385M	353M	1M	1M
English-Slovak	11M	144M	126M	543k	1M
English-Slovene	13M	161M	133M	631k	1M
English-Swedish	16M	193M	163M	687k	1M

Table 3: Statistics on parallel data for translation model training and word-sense disambiguation.

5. Results

In this section, we present the evaluation of the translated IATE entries into several languages not initially included in this resource, and how existing IATE terms have been exploited for our purposes in the parallel corpora used in this work.[7] In addition to this, we illustrate the enhancing of IATE RDF resource with additional contextual information

[5] Based on IATE TBX file - IATE_export_16032017.tbx

[6] http://opus.nlpl.eu/

[7] We randomly selected 2,000 terms, although not all target terms are represented in each language for evaluation.

	# of Terms		Random Context			Disambiguated Context		
	Evaluated	New	BLEU-1	METEOR	ChrF	BLEU-1	METEOR	ChrF
Bulgarian	406	1,594	72.3	36.5	86.3	78.5*	39.3	86.9
Danish	1,502	498	64.7	39.3	80.9	72.8*	43.5	84.4
Greek	1,289	711	61.8	52.3	81.4	69.3*	55.1	84.9
Spanish	1,504	496	77.7	37.4	77.3	84.2*	39.1	78.5
Finnish	1,014	986	45.2	26.4	75.2	51.1*	29.7	77.2
French	1,566	434	74.4	37.9	79.2	79.5*	39.2	79.6
Croatian	155	1,845	58.3	30.5	73.4	59.6	30.9	73.4
Italian	1,524	476	72.6	35.5	75.1	81.7*	38.3	77.6
Latvian	451	1,549	57.2	29.8	74.6	63.2*	33.6	79.5
Dutch	1,505	495	69.8	42.1	82.0	77.8*	46.6	86.1
Polish	566	1,434	53.8	25.1	70.2	63.4*	29.0	74.8
Portuguese	1,504	496	79.1	37.4	77.2	84.3*	38.9	78.4
Romanian	431	1,569	67.6	31.4	71.7	75.5*	35.5	74.7
Slovak	459	1,541	59.5	30.4	76.2	67.0*	34.7	79.4
Slovene	501	1,499	58.2	29.7	75.4	67.8*	34.3	79.2
Swedish	1,071	929	66.8	40.6	83.3	75.0*	45.7	85.9

Table 4: Automatic translation evaluation of IATE entries with random and disambiguated context (* statistically significant compared to the random context translation approach).

English	certification of products pursuant to specific airworthiness specifications , the related modifications , repairs and their continuing **airworthiness** , shall be charged as defined in tables 1 to 6 .
Italian	certificazione dei prodotti in conformità a determinate specifiche di aeronavigabilità , le relative modifiche , le riparazioni e la loro **aeronavigabilità** continua , sono contabilizzate come definito nelle tabelle da 1 a 6 .
Dutch	certificering van producten overeenkomstig specifieke luchtwaardigheidsspecificaties , de bijbehorende wijzigingen en reparaties en de permanente **luchtwaardigheid** daarvan , worden in rekening gebracht als omschreven in de tabellen 1 tot en met 6 .
Danish	certificering af produkter i henhold til de specifikke luftdygtighedsspecifikationer , de relaterede ændringer , reparationer og deres fortsatte **luftdygtighed** faktureres i overensstemmelse med tabel 1 til 6 .
Slovene	certifikacija proizvodov v skladu s posebnimi plovnostnimi specifikacijami , povezane spremembe , popravila in njihova stalna **plovnost** se zaračunavajo , kot je določeno v tabelah 1 do 6 .
Portuguese	a certificação de produtos em conformidade com especificações de aeronavegabilidade próprias , bem como as modificações e reparações associadas e respetiva **aeronavegabilidade** permanente , devem ser cobrados conforme definido nas tabelas 1 a 6

Table 5: Identified sentence for IATE entry *airworthiness*, and their translations in different languages.

(examples of sentences that show the real use of the term) and information about duplicates, so that they can be easily identified for an eventual cleaning of the resource.

5.1. Translation Evaluation

Table 4 illustrates the automatic translation evaluation for 16 languages. IATE entries are translated within a random and identified disambiguated context. Except for the Croatian language, translating IATE terminological expressions within a disambiguated context, statistically significantly (p-value < 0.01) outperforms the approach with randomly selected sentences. Due to this results, we believe that the newly added terms show similar translation quality.

5.2. Providing Disambiguated Contextual Information

To further enhance the IATE terminological resource, we believe that the identified disambiguated sentences can be beneficial for the users selecting a term due to the contextual information of the targeted domain. Therefore, we append the identified relevant sentences to the IATE RDF resource. Table 5 illustrates an example of the relevant sentence associated with the IATE term *airworthiness* (IATE-29309), with its translations in several languages.

5.3. Frequency and Reliability of Term Variants

With the aim of providing users with information about term variants and to help them choosing the best variant for their purposes, as well as to provide data to support the reliability score originally assigned by IATE terminologist, we perform an experiment on evaluating the appearance of terms in the parallel corpora used in this work. Table 6 illustrates examples of IATE terms in English that have more than one translation equivalent in the target languages, in this case, German, French and Slovene. These can be considered as term variants, and frequency numbers can give

IATE ID	English Term	En. Term Freq.	Target Term	S&T Freq.
IATE-913759	south caucasus	277	südkaukasus	229
			transkaukasien	1
IATE-1108878	body mass index	282	body-mass-index	126
			körpermasse-index	4
IATE-1126703	outward investment	122	investissement extérieur	7
			investissement réalisé à l'étranger	0
IATE-46262	electrical engineering	137	électrotechnique	36
			génie électrique	16
IATE-1211765	shovel	1,398	pelle	933
			bêche	10
	spade	645	pelle	66
			bêche	34
IATE-770947	state of the art	235	stanje tehnike	16
			najsodobnejša tehnologija	0
IATE-814939	distortive effect	42	izkrivljajoč učinek	7
			učinek izkrivljanja	6
	distorting effect	28	izkrivljajoč učinek	3
			učinek izkrivljanja	3

Table 6: Examples of IATE term frequencies in the parallel corpora.

us a clue on the real use of the term and, consequently, on how reliable it is to use it in a certain context.

Although the usage of *südkaukasus* and *transkaukasien* differs significantly based on the number of times these terms are mentioned in the parallel corpus, the IATE-assessed reliability score for both terms is the same (three out of four stars). Similarly, based on the parallel corpora, *body-mass-index* is highly preferred in comparison to the German translation *körpermasse-index*, but both terms again have same assessed reliability score (three out of four stars).

For the French translation of *outward investment*, the most used translation within the parallel corpus is *investissement extérieur*, whereby the additionally suggested term *investissement réalisé à l'étranger* documented in IATE does not appear in the corpus. On the other hand, the French translations of the term *electrical engineering*, i.e., *électrotechnique* and *génie électrique*, are both frequently mentioned in the used corpora. Furthermore, the entry *shovel* (IATE-1211765) is mostly aligned with the French term *pelle*, whereby *spade* is frequently translated as *pelle* as well as *bêche*.

For Slovene, *state of the art* is mostly aligned with *stanje tehnike* in the parallel corpus, whereby the additional term of the same IATE entry, *najsodobnejša tehnologija* is not a common translation for the English term according to the parallel corpus. The English terms *distortive effect* and *distorting effect*, both belonging to the IATE-814939 entry, are similarly frequent in English as well as their translations into Slovene.

5.4. IATE Duplicates

Table 7 presents the amount of duplicate English entries and their translations. As seen, most of the duplicate entries appear between the English and French language pair. As an example, *electronystagmography* in English and *électronystagmographie* in French are represented in two IATE entries, i.e., IATE-1289555 and IATE-1517532, although both entries have the same IATE `subjectField` (2841, i.e., *Medical science*). Similarly, the very specific IATE terms *international natural rubber council* in English and *internationaler Naturkautschukrat* in German belong to two separate IATE entries, i.e., IATE-151353 and IATE-777553. Different to the previous example, these two entries belong to different, but similar domains, i.e., *Industry* and *International trade*. As in the case of frequency of use and reliability, such a statistical corpus analysis would allow us to identify those cases in which entries have been duplicated, most probably because of the merging of similar terminological resources.

French	57,216	Finnish	9,677	Lithuanian	372
German	28,013	Slovak	8,293	Slovak	358
Italian	27,941	Irish	1,163	Estonian	352
Spanish	23,533	Maltese	683	Hungarian	314
Dutch	23,009	Slovene	519	Czech	296
Danish	22,331	Polish	503	Latvian	259
Portuguese	20,185	Romanian	489	Croatian	148
Greek	16,891	Bulgarian	373		

Table 7: Statistics on duplicates between IATE terms in English and their translations.

5.5. Publication

In order to maximize the availability of this data, this data is available under an open license (CC-BY) and was contributed to the RDF version of IATE (Cimiano et al., 2015) so that it will be part of the linked open data cloud. In order to distinguish this automatically created data from the existing manually created data in IATE, we used the PROV-O

ontology (Lebo et al., 2013).[8]

6. Conclusion and Future Work

In this work, we showed automatic approaches for maintaining and increasing the lexical coverage of knowledge bases, e.g. IATE. By identifying disambiguated context, we demonstrate statistical significant translation improvement for several languages. With this approach, we identified relevant sentences, which can be beneficial for users when short terms cannot be disambiguated without any context surrounding them. To differentiate commonly used and less preferred terms, we evaluate the usage of the IATE terms and their translations in parallel corpora. At last, we identify duplicates in IATE, which can help to maintain and clean up the terminological resource. As ongoing work we are focusing on neural machine translation to cross the language barrier and how to incorporate the lexical information as well as the semantic structure of the resource into an embedded space for translating domain-specific expressions (Arcan and Buitelaar, 2017). Furthermore, our focus will lie on existing terminological variations within IATE as well as variations provided by machine translation.

Acknowledgements

This publication is supported by a research grant from Science Foundation Ireland, SFI/12/RC/2289 (Insight), a research visit grant from Universidad Politécnica de Madrid, by the Spanish Datos4.0 project (TIN2016-78011-C4-4-R) and by the EU's Lynx project (H2020 Research and Innovation Programme under GA núm 780602).

7. Bibliographical References

Arcan, M. and Buitelaar, P. (2017). Translating domain-specific expressions in knowledge bases with neural machine translation. *arXiv preprint arXiv:1709.02184*.

Arcan, M., Turchi, M., and Buitelaar, P. (2015). Knowledge portability with semantic expansion of ontology labels. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, Beijing, China.

Arcan, M., Dragoni, M., and Buitelaar, P. (2016a). Translating ontologies in real-world settings. In *Proceedings of the 15th International Semantic Web Conference (ISWC-2016)*, Kobe, Japan.

Arcan, M., McCrae, J. P., and Buitelaar, P. (2016b). Expanding wordnets to new languages with multilingual sense disambiguation. In *26th International Conference on Computational Linguistics (COLING'16)*, Osaka, Japan.

Cimiano, P., Montiel-Ponsoda, E., Buitelaar, P., Espinoza, M., and Gómez-Pérez, A. (2010). A note on ontology localization. *Appl. Ontol.*, 5(2):127–137.

Cimiano, P., McCrae, J. P., Rodríguez-Doncel, V., Gornostay, T., Gómez-Pérez, A., Siemoneit, B., and Lagzdins, A. (2015). Linked terminologies: Applying linked data principles to terminological resources. In *Proceedings of the fourth biennial conference on electronic lexicography (eLex 2015)*, pages 504–517, Sussex, United Kingdom.

Clark, J., Dyer, C., Lavie, A., and Smith, N. (2011). Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of the Association for Computational Lingustics*, Portland, Oregon.

de Melo, G. and Weikum, G. (2012). Constructing and utilizing wordnets using statistical methods. *Language Resources and Evaluation*, 46(2):287–311.

Declerck, T., Pérez, A. G., Vela, O., Gantner, Z., and Manzano, D. (2006). Multilingual lexical semantic resources for ontology translation. In *In Proceedings of the 5th International Conference on Language Resources and Evaluation*, Saarbrücken, Germany.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, Gothenburg, Sweden.

Eisele, A. and Chen, Y. (2010). MultiUN: A multilingual corpus from United Nation documents. In Daniel Tapias, et al., editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation*, pages 2868–2872. European Language Resources Association (ELRA), 5.

Espinoza, M., Montiel-Ponsoda, E., and Gómez-Pérez, A. (2009). Ontology localization. In *Proceedings of the Fifth International Conference on Knowledge Capture*, NY, USA.

Fu, B., Brennan, R., and O'Sullivan, D. (2009). Cross-lingual ontology mapping - an investigation of the impact of machine translation. In Asunción Gómez-Pérez, et al., editors, *ASWC*, volume 5926 of *Lecture Notes in Computer Science*. Springer.

Gracia, J., Montiel-Ponsoda, E., Cimiano, P., Gómez-Pérez, A., Buitelaar, P., and McCrae, J. (2012). Challenges for the multilingual web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 11.

Heafield, K. (2011). KenLM: faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland, United Kingdom, July.

Klyne, G. and Carroll, J. J. (2006). Resource description framework (RDF): Concepts and abstract syntax. W3C recommendation, World Wide Web Consortium.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54, Edmonton, Canada. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th*

[8] for more information see `https://nuig.insight-centre.org/unlp/research/resources/multilingual-iate-rdf/`

Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, Prague, Czech Republic.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*. AAMT.

Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., and Zhao, J. (2013). PROV-O: The PROV ontology. W3c recommendation, World Wide Web Consortium.

McCrae, J., Espinoza, M., Montiel-Ponsoda, E., Aguado-de Cea, G., and Cimiano, P. (2011). Combining statistical and semantic approaches to the translation of ontologies and taxonomies. In *Fifth workshop on Syntax, Structure and Semantics in Statistical Translation (SSST-5)*, Portland, Oregon.

McCrae, J. P., Arcan, M., Asooja, K., Gracia, J., Buitelaar, P., and Cimiano, P. (2016). Domain adaptation for ontology localization. *Web Semantics*, 36:23–31.

McCrae, J. P., Arcan, M., and Buitelaar, P. (2017). Linking knowledge graphs across languages with semantic similarity and machine translation. In *Workshop on Multi-Language Processing in a Globalising World (MLP2017)*, Dublin, Ireland.

Navigli, R. (2012). Babelnet goes to the (multilingual) semantic web. In *Proceedings of the 3rd Workshop on the Multilingual Semantic Web, Boston, USA, November 11th, 2012*.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29.

Ordan, N., Gracia, J., and Kernerman, I. (2017). Autogenerating bilingual dictionaries. In *Proceedings of fifth biennial conference on electronic lexicography (eLex 2017),*, Leiden, Netherlands.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318.

Pérez, M. and Berlanga, R. (2015). Semantic transference for enriching multilingual biomedical knowledge resources. *Journal of biomedical informatics*, 58:1–10.

Popović, M. (2015). chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, September. Association for Computational Linguistics.

Sajous, F., Navarro, E., Gaume, B., Prévot, L., and Chudy, Y., (2010). *Semi-automatic Endogenous Enrichment of Collaboratively Constructed Lexical Resources: Piggybacking onto Wiktionary*, pages 332–344. Springer Berlin Heidelberg, Berlin, Heidelberg.

Stanojević, M., Kamran, A., Koehn, P., and Bojar, O. (2015). Results of the WMT15 Metrics Shared Task. In *Proceedings of the 10th Workshop on Statistical Machine Translation (WMT-15)*, pages 256–273, Lisbon, Portugal, September.

Steinberger, R., Ebrahim, M., Poulis, A., Carrasco-Benitez, M., Schlüter, P., Przybyszewski, M., and Gilbro, S. (2014). An overview of the european union's highly multilingual parallel corpora. *Language Resources and Evaluation*, 48(4):679–707.

Tiedemann, J. (2009). News from OPUS – A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Advances in Natural Language Processing*, volume V, chapter V, pages 237–248. Borovets, Bulgaria.

Tiedemann, J. (2012). Character-based pivot translations for under-resourced languages and domains. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 141–151, Avignon, France, April.

A Comparative Study of Extremely Low-Resource Transliteration of the World's Languages

Winston Wu, David Yarowsky
Department of Computer Science, Center for Language and Speech Processing
Johns Hopkins University
{wswu,yarowsky}@jhu.edu

Abstract

Transliteration from low-resource languages is difficult, in large part due to the small amounts of data available for training transliteration systems. In this paper, we evaluate the effectiveness of several translation methods in the task of transliterating around 1000 Bible names from 591 languages into English. In this extremely low-resource task, we found that a phrase-based MT system performs much better than other methods, including a g2p system and a neural MT system. However, by combining the data and training a single neural system, we discovered significant gains over single-language systems. We release the output from each system for comparative analysis.

Keywords: Bible, alignment, named entities, translation, transliteration

1. Introduction

Transliteration is the process of converting text from one script to another. For example, Росси́я is *transliterated* as Rossiya but would be *translated* as Russia. Transliteration is important in the process of borrowing names between languages. In the case of low-resource languages, there is often little training data with which to train transliteration models. Thus, one major obstacle for low-resource languages is the problem of out of vocabulary words (in the transliteration setting, these would be unknown characters). Machine translation systems might use a transliteration step to resolve unknown words, especially names; in transliteration, one can perform a similar process by attempting to resolve unknown characters. In this paper, we compare and contrast multiple methods of transliterating Bible names in over 500 languages into English and evaluate the effectiveness of a pre- and post-transliteration step in transliteration in resolving unknown characters. Our results indicate that a phrase-based machine translation system is the most effective when training on data in the order of hundreds of words, while by simply concatenating multiple languages' training data with some preprocessing, a single neural MT system trained to transliterate multiple languages to a single target language significantly outperforms the single-language systems. We release each system's transliteration output as a dataset for comparative analysis. To our knowledge, this is the first study of such scale that compares such a variety of methods on such small corpora.

2. Related Work

Machine transliteration has been tackled using a variety of methods. For a comprehensive survey, we refer the reader to Karimi et al. (2011). In this paper, we focus on transliteration of named entities across multiple language, especially in a low-resource setting, using the paradigm of monotonic machine translation (Virga and Khudanpur, 2003). Our work is most similar to Irvine et al. (2010), who built character-based machine translation systems using names mined from Wikipedia. Their setting is higher resource, as they use data acquired from the web, and they experiment on a smaller set of languages. Other recent approaches to transliteration of low resource languages include Mayhew et al. (2016), who explore using surrogate languages in place of a language not in Wikipedia, Rosca and Breuel (2016) who showed state of the art transliteration performance using a neural sequence to sequence model, and Jiampojamarn et al. (2010), who explore several methods for language-independent transliteration mining. Qian et al. (2010) also developed a toolkit to extract translation pairs from comparable corpora.

We compare several open-source toolkits for translation: Moses (Koehn et al., 2007), a phrase-based statistical machine translation toolkit; Sequitur (Bisani and Ney, 2008), a grapheme to phoneme system; and OpenNMT (Klein et al., 2017), a neural sequence-to-sequence machine translation system. While Sequitur is not commonly used in MT, Sequitur and Moses have been compared for speech recognition tasks (Schlippe et al., 2014).

3. Data

We use the Bible names translation matrix dataset (Wu and Yarowsky, 2018), which contains 1129 person and place names in the Bible aligned across 519 languages. They constructed this name translation matrix using a combination of distance-based, MT transliteration, and string transduction rules to improve the alignments from a baseline aligner. The scope of this data accentuates the low-resource setting, which is reasonable for many of the world's languages; the Bible may be one of the only bilingual resources available for certain languages.

4. Experiments

We perform three major transliteration experiments: the first, in which we compare several existing machine translation systems in the task of transliteration; the second, in which we evaluate the effectiveness of pre- and post-processing the data with a baseline transliterator; and the third, in which we employ a single NMT system to translate between all pairs of languages. The experiments consisted of training each system to transliterate from a source language into English[1]. We used names from the aforementioned dataset in a 80-10-10 train-dev-test split. All names

were lowercased, and characters were space-separated.

In our experiments, the baseline system was Unidecode,[2] a Python library that provides a language-independent mapping from a Unicode character to a fixed ASCII string. While this is a naive baseline, for languages that do not have much data to train transliteration systems, this may be one of the few transliteration options available.

Moses was trained using a vanilla setup, with a 4-gram KenLM (Heafield, 2011) language model, tuning with MERT (Och, 2003), and setting the distortion limit to 0 to prevent reordering, which does not occur during transliteration. Sequitur was trained iteratively using the `--ramp-up` flag three times. For OpenNMT, we used the following hyperparameters: a 2 layer GRU for encoder and decoder, optimizer is Adadelta, 0.2 dropout rate, hidden size 200, embedding size 200, no length normalization. We trained each model for 50 epochs and used the model with the lowest validation perplexity.

In the second experiment, we examine how pre- and post-transliteration using a baseline transliterator (Unidecode) affects a Moses-based transliterator. Pre-transliteration is a preprocessing step that uses Unidecode to convert the input data into ASCII letters before it is processed by Moses. In post-transliteration, Unidecode postprocesses the output from Moses to catch foreign (non-ASCII) letters that were not transliterated. We hypothesize that if there is a one-to-one character mapping between the characters of the source language and ASCII letters, then neither pre- nor post-transliteration should help. We expect the preprocessing step to reduce the character set of the source language, which has the effect of losing information if multiple characters can map to a single ASCII character (e.g. in Greek, σ and ς both correspond to the letter s). For postprocessing, we expect that this step will correct characters that were not transliterated, which can occur if they were not seen in the training set, similar to unseen words in MT.

In our final experiment, we exploit the multilinguality of the data for transfer learning. Inspired by the winning system in the SIGMORPHON 2016 shared task (Kann and Schütze, 2016), we train a single neural MT system on the concatenation of the entire training set. This allows the system to learn a joint model from multiple source languages to a single target language. In addition, by concatenating training sets of each of the single language-pair systems, the multi-source approach can circumvent the data scarcity problem. Each training example was split into spaces, with a special source language symbol prepended, as shown in Table 1. No target language symbol was used because the target language was only English. Since the training and test data are different (and substantially larger), this system is not directly comparable to the above approaches but rather represents

[1]Note that the training and test sets are in the same domain, i.e. named entities. While it may be interesting to test on an unrelated set of words, the results are not likely to be encouraging unless these words are orthographically/phonologically similar to their English counterparts (e.g. cognates or borrowed words).

[2]`https://pypi.python.org/pypi/Unidecode`

Source	Target
<ann> p o l ǫ k i s	p o l l u x
<bnp> p o l u k s	p o l l u x
<kwf> p o l a k s	p o l l u x
<msy> p o l l u k s	p o l l u x
<mti> p o r a k u s	p o l l u x
<mto> p ó l u x	p o l l u x
<ncj> p ó l u x	p o l l u x
<rus> п о л л у к с а	p o l l u x

Table 1: Bitext format for the OpenNMT Multi experiment. The target word is the English name Pollux.

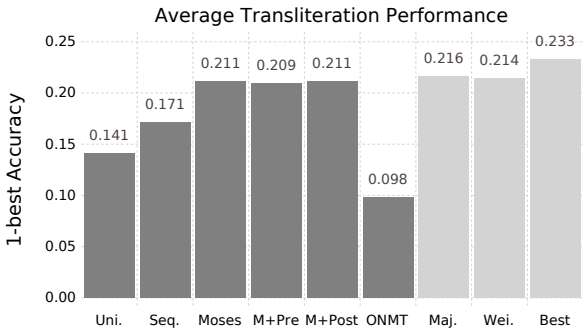


Figure 1: Average transliteration performance for single language pair systems. Light gray indicates system combination (majority, weighted, and best per language).

5. Results, Discussion, and Error Analysis

Each system was evaluated on 1-best exact-match accuracy. Due to the extremely low-resource nature of the data (on the order of a few hundred training examples), the task proved to be quite difficult. On average, as shown in Fig. 1, Sequitur performs better than the baseline Unidecode, and the Moses transliterator performs best overall. Moses with pre- and post-processing perform comparably on average.

Since there are a number of different systems trying to accomplish the same task, a natural question is whether combining the systems' outputs would result in improvements. Out of the six systems, simply choosing the most common hypothesis works better than Moses alone. Using a weighted combination, where each hypothesis is weighted by their respective system's average performance also results in gains over a single system. Finally, if we consider the best performing system on a per-language basis, the average performance sees a small increase.

In the following section, we examine each system in relation to the vanilla Moses system (without pre- or post-transliteration). Code to reproduce these results are available[3]. The three letters that represent a language are ISO 639-3 language codes, and asterisks denote incorrect transliterations.

Baseline. As a language-independent baseline, we expected that Unidecode would perform worse in most cases. Indeed, the baseline obtained 0% accuracy for 86 languages. This is due to the presence of morphological affixes. Con-

[3]`github.org/wswu/trabina`

sider the following:

Apurinã (apu)	English	Unidecode	Moses
épeso	ephesos	*epeso	*ephesus[4]
xório	julius	*xorio	julius
nikorao	nicolas	*nikorao	*nicolaus

The -o ending in Apurinã, which likely denotes a nominative, is translated to -us by Moses. Unidecode cannot handle this phenomenon, since it operates only on single characters; here it merely stripped off accents from each character. In addition, since Unidecode is not a translation system, it will never convert x → j or k → c, since these are already ASCII letters. Many other languages exhibit similar morphological patterns that could not be handled by Unidecode. However, the baseline actually performs better than Moses in 42 languages, which is a testament to the difficulty of our task. In these cases, we believe that Moses may have learned an incorrect translation model from the tiny amounts of training data, when simply passing the character through the system unchanged or applying a script conversion (e.g. Cyrillic to Roman) would have resulted in the correct answer. The following are some instances where Unidecode transliterates the correct name, while Moses does not.

Lang	Unidecode	Moses
Siyin (csy)	enoch	*enoc
Guahibo (guh)	jordan	*jordam
Ukranian (ukr)	puteoli	*putheoli
Murrinh-patha (mwf)	moses	*mouseus

Sequitur. While Sequitur is more commonly used in the speech community as a grapheme-to-phoneme software, we considered using Sequitur because transliteration is performed on sequences of characters rather than entire words. Sequitur's average performance was only below Moses' by a few percentage points, and it actually outperformed Moses in 145 languages. Some examples of Sequitur's successes over Moses follow:

Lang	Sequitur	Moses
Amele (aey)	elam	*ilam
	abilene	*abylene
Balinese (ban)	cleopas	*clopas
Bukawa (buk)	bartimaeus	*batimeas
Hawaiian Pidgin	castor	*casthor
(hwc)	phrygia	*phirygia
Hote (hot)	miletus	*miretus
	philetus	*piletus
	troas	*troaz

We observe that the mistakes are natural looking mistakes: mixing up letters that are phonologically similar (*l* and *r*, *s* and *z*, and some vowels), and occasionally adding or removing an *h*. Our hypothesis is that, in these cases, Moses' language model is biasing the system away from the correct answer. While the language model is trained on same-domain data (i.e. Bible named entities), it is possible that the language model would give a low score to unusual letter sequences like "aeus" in Bartimaeus or "phry" in Phrygia. Further investigation is needed to determine the role the language model plays in the transliteration process.

OpenNMT. Neural machine translation is state-of-the-art for many language pairs (Bojar et al., 2016). In this transliteration task, however, the neural MT system performed the worst overall. For 22 languages, OpenNMT performed better than Moses, but for most of these languages, the accuracies were under 10%, representing very small gains.
While it is possible that the parameters we used were not optimal for character-based transliteration, it is likely that the size of the data was just too small for a neural model to effectively learn from. This corroborates a finding from Koehn and Knowles (2017) that neural MT models tend to perform better than phrase-based models only past a certain threshold of data size (a corpus size of over 10^7 words). We observed that the neural MT system often prefers shorter words compared to the phrase-based Moses, and perplexity on the development set was generally lowest after around 20–25 epochs training, after which overfitting was evident. Below are some example transliterations from Moses and OpenNMT:

Lang	Source	Moses	OpenNMT
Qaqet (byx)	aleksandria	alexandria	*alandria
Frafra (gur)	metusela	*methushelah	*metusel
	alekzander	*alechzander	alexander
Hiri Motu	eparona	*epharon	ephron
(hmo)	mikaela	*micael	michael

5.1. Resolving Unknown Characters

Analogous to OOVs in machine translation, handling unknown characters is vital in achieving high transliteration accuracy. Since the target language in our experiments is always English, we utilized Unidecode to transliterate characters into ASCII letters.

Preprocessing. While the average accuracy of Moses + preprocessing was almost the same as without preprocessing, preprocessing helped in 195 languages but hurt in 206 languages, suggesting that preprocessing is largely dependent on the language.

Lang	Source	Moses	+Pre
Ankave (aak)	segɨria	*cenria	cenchrea
Greek (ell)	εῦα	*eῦa	eva
Ukranian (ukr)	марта	*marta	martha
Armenian (hye)	սողոմոն	solomon	*solomone
Russian (rus)	косам	cosam	*kosam
Ossetian (oss)	тимейы	timaeus	*timee

While pretransliterating helps in some cases, in other cases it appears to conflate character mappings, thereby removing information crucial for transliteration. Interestingly, for the name "Martha", although Moses with preprocessing gives the correct English name, Moses without preprocessing actually produced a closer representation of how the name would be pronounced in Ukranian[5]. For "Cosam", preprocessing the к → k seems to have prevented the correct letter

[4] "Ephesus" is also a valid spelling of this ancient Greek city, but the dataset contains only one correct gold.

[5] Martha originated from the Aramaic Martâ, which was borrowed into Greek as Μάρθα, which was transliterated into Latin as Martha.

c from appearing in the output instead. From our experiments, it is not clear in what cases one should apply pre-transliteration.

Postprocessing. We found that postprocessing the output never hurts performance, but improves performance only for four source languages: Acehnese (ace), German (deu), Korean (kor), and Munduruku (myu). For Acehnese, German, and Munduruku, the improvements largely consisted of removing a diacritic on a letter (e.g. Joël → Joel). The interesting case however is Korean.

English	malchus	felix	crete
Moses	* 말 chus	*pel 릭 s	* 크 re 테
+Post	malchus	*pelrigs[6]	*keurete

Due to Korean's system of composing characters based on sound, Korean words may contain characters not seen during training time. In these cases, Moses treats the character as an unknown word and outputs the same character untranslated. This is a prime example of a situation in which we expect post-processing to help. In the case of Malchus, post-processing correctly transliterates the unknown Korean character, resulting in the correct English name, while for Felix and Crete, the post-transliterated name is not an exact match but is recognizably close.

5.2. Analyses by Category

We additionally examine transliteration performance not by language, but by features of the words themselves. By aggregating all words and stratifying based on word length, we can gain additional insights.

English word length. Roughly two thirds of English names are longer than 5 characters. When separating these names into short (length $\leq$ 5) and long (length $>$ 5), we see that all systems had an easier time transliterating shorter words.

System	$\leq$ 5 chars	$>$ 5 chars
Moses	3079/10092 (.31)	4536/24314 (.19)
Sequitur	2492/10092 (.25)	3756/24314 (.15)
Unidecode	2623/10092 (.26)	2437/24314 (.10)

Edit distance between source and target. We see that around two thirds of the names in the entire set have an edit distance of 3 or less to the English. This is not too surprising, especially since most languages use Roman script. We see that the performance of transliteration systems degrades as the difference between source and target increases.

System	Dist $\leq$ 3	Dist $>$ 3
Moses	6866/21746 (.32)	749/12660 (.06)
Sequitur	5867/21746 (.27)	381/12660 (.03)
Unidecode	4995/21746 (.23)	65/12660 (.01)

Roman vs non-roman characters. There are surprisingly few languages in the world that do not use a Roman character set; the data set contains 35 such languages. These include the Arabic and Cyrillic scripts, which are used in several languages, as well as other scripts, including Hangul and the family of Brahmic scripts, that are specific to a single language. We find that on average, the transliteration of languages in Roman script performs better that that of non-Roman script languages. A surprising result is that Sequitur does not transliterate any non-Roman words correctly, which may be due to encoding issues.

System	Roman	Non-Roman
Moses	7316/32129 (.23)	304/2277 (.13)
Sequitur	6248/32129 (.19)	0/2277 (.00)
Unidecode	4988/32129 (.16)	72/2277 (.03)

For the non-Roman script languages, we performed the same analysis as above. The following table shows average accuracy on languages written in non-Roman scripts.

Model	Accuracy
Baseline	0.031
Moses	0.125
+pre	0.119
+post	0.125

The baseline performs poorly as expected. Pretransliteration improves over Moses in 11 languages but underperforms in 17. Even on this subset of non-Roman script languages, there seems to be no pattern as to whether preprocessing is effective or not, which reiterates our findings for the entire dataset. Even within a language family, for example, transliterating Kannada and Marathi has slightly higher performance with preprocessing, while Tamil suffers. Preprocessing does slightly better overall for Greek, but when examining the transliterated names, there is evidence both for and against preprocessing:

Greek	Moses	+Pre
ιαρεδ	*jered	jared
ιεριμωθ	jerimoth	*jeremoth

Transfer learning. The single neural MT system trained on the concatenation of the training data for all languages performed much better than the other systems in our experiments, achieving a 69% one-best accuracy on the concatenation of the test sets. This massive gain stems from the combination of the 1000x increase in training data and the neural architecture's ability to effectively leverage the commonality between languages. This result indicates that this transfer learning technique works well when combining low-resource languages, even when each individual language pair may only have a miniscule amount of data.

6. Conclusion

We have performed an extensive comparison of several machine translation methods adapted for transliteration of 591 languages into English. By evaluating the performance of Unidecode, Sequitur, Moses, and OpenNMT across most of the world's languages, we observed that the phrase-based machine translation paradigm was the most effective for training character-based transliteration systems on tiny amounts of data. Performing a pre-processing

[6]Note that Korean does not have an 'f' sound.

Foreign	Lang	English	Unidecode	Sequitur	Moses	M+Pre	M+Post	OpenNMT
filipi'de	tur	philippi	filipi'de		philipia	philipia	philipia	philip
poluks	mnx	pollux	poluks	poluk	polucs	polucs	polucs	polug
timotio	aak	timothy	timotio	**timothy**	**timothy**	**timothy**	**timothy**	timothe
ibrahima	bam	abraham	ibrahima	ibraham	**abraham**	**abraham**	**abraham**	ibraham
alejandría	caa	alexandria	alejandria		**alexandria**	**alexandria**	**alexandria**	al
gebeliela	dob	gabriel	gebeliela	gebriel	geberiel	geberiel	geberiel	gebbrla
гедеоне	rus	gideon	gedeone		gedeone	gedeon	gedeone	gahedon
filaistus	mfi	philetus	filaistus	philestus	phlestus	phlestus	phlestus	fylestus
		Avg Accuracy:	.14	.17	.2129	.2134	.2130	.10

Table 2: Example transliterations from each system. Average accuracy is over all languages, not just the ones listed in the table. Correct transliterations are bolded. Note that Sequitur is not guaranteed to transliterate a word.

or post-processing transliteration step using a language-independent transliterator to deal with unknown characters yielded inconclusive (statistically insignificant) results as to whether preprocessing is effective, though we found evidence that post-transliterating can help with unseen characters. Standard methods of system combination slightly boosted performance. In addition, we found that exploiting the multilinguality of the data allows for effective transfer learning in a single neural machine translation model that can act as a universal transliterator. A dataset of names transliterated by each system is available for research purposes. With recent advances in neural models, we believe that approaches leveraging multiple languages are worth exploring in the future.

7. Acknowledgments

This work was supported in part by the DARPA LORELEI program. The findings, conclusions, and opinions found in this work are those of the authors and do not necessarily represent the views of the funding agency.

8. Bibliographical References

Bisani, M. and Ney, H. (2008). Joint-sequence models for grapheme-to-phoneme conversion. *Speech communication*, 50(5):434–451.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Jimeno Yepes, A., Koehn, P., Logacheva, V., Monz, C., Negri, M., Neveol, A., Neves, M., Popel, M., Post, M., Rubino, R., Scarton, C., Specia, L., Turchi, M., Verspoor, K., and Zampieri, M. (2016). Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, August. Association for Computational Linguistics.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Irvine, A., Callison-Burch, C., and Klementiev, A. (2010). Transliterating from all languages. In *Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA)*.

Jiampojamarn, S., Dwyer, K., Bergsma, S., Bhargava, A., Dou, Q., Kim, M.-Y., and Kondrak, G. (2010). Transliteration generation and mining with limited training resources. In *Proceedings of the 2010 Named Entities Workshop*, pages 39–47. Association for Computational Linguistics.

Kann, K. and Schütze, H. (2016). Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 555–560. Association for Computational Linguistics.

Karimi, S., Scholer, F., and Turpin, A. (2011). Machine transliteration survey. *ACM Computing Surveys (CSUR)*, 43(3):17.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). Opennmt: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Mayhew, S., Christodoulopoulos, C., and Roth, D. (2016). Transliteration in any language with surrogate languages. *arXiv preprint*.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Qian, T., Hollingshead, K., Yoon, S.-y., Kim, K.-y., and Sproat, R. (2010). A python toolkit for universal transliteration. In *LREC*.

Rosca, M. and Breuel, T. (2016). Sequence-to-sequence neural network models for transliteration. *arXiv preprint*.

Schlippe, T., Quaschningk, W., and Schultz, T. (2014). Combining grapheme-to-phoneme converter outputs for enhanced pronunciation generation in low-resource scenarios. In *The 4th Workshop on Spoken Language Tech-*

nologies for Under-resourced Languages, St. Petersburg, Russia. SLTU 2014.

Virga, P. and Khudanpur, S. (2003). Transliteration of proper names in cross-lingual information retrieval. In *Proceedings of the ACL 2003 workshop on Multilingual and mixed-language named entity recognition-Volume 15*, pages 57–64. Association for Computational Linguistics.

Wu, W. and Yarowsky, D. (2018). Creating a translation matrix of the Bible's names across 591 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*. European Language Resources Association (ELRA).

Translating Web Search Queries into Natural Language Questions

Adarsh Kumar, Sandipan Dandapat, Sushil Chordia
AI & Research, Microsoft
Hyderabad, India
{adkuma, sadandap , sushilc}@microsoft.com

Abstract

Users often query a search engine with a specific question in mind and often these queries are keywords or sub-sentential fragments. In this paper, we are proposing a method to generate well-formed natural language question from a given keyword-based query, which has the same question intent as the query.Conversion of keyword based web query into a well formed question has lots of applications in search engines, Community Question Answering (CQA) website and bots communication. We found a synergy between query-to-question problem with standard machine translation (MT) task. We have used both Statistical MT (SMT) and Neural MT (NMT) models to generate the questions from query. We have observed that MT models performs well in terms of both automatic and human evaluation.

Keywords: Natural Language Generation, Machine Translation, NLP

1. Introduction

Search engines have improved a lot in last decade in all aspects. Earlier, the primary task of a search engine was to extract most relevant links for the query and present them as results. Lately, instead of just giving relevant links related to the query, search engines are trying to directly answer to any question asked. For example, for the query *"japan's capital"* in modern search engines (eg. Bing and Google) directly answer *"Tokyo"*, instead of providing a link containing the answer. Thus, search engines are evolving to save time for users and increase their productivity. To further enhance the user-experience and increase productivity, search engines apart from showing the answer for a particular question, are trying to show related questions, to help users in their exploration. For example, for the query *"fever symptoms"*, user mostly wants answer to the question *"What are the symptoms of fever?"* and for the same query, questions like *"How do you treat fever?"*, *"What causes high fever?"* are highly related. To show related questions, search engines need to have a well framed question corpus from which they can extract relevant questions given a query. (White et al., 2015) have shown that more than 10% of queries issued on a search engine has question intent whereas only 3% of them are formulated as natural language questions. Most of these queries are primarily keywords or sentence fragments. Hence, a corpus of questions can not be created directly using the search queries with question intent due to the issue of grammatical correctness and incomplete sentence formation. To overcome this problem, we are proposing a technique to convert query with question intent, into a well-formed question. This technique can be used to generate well formed questions asked by the user, which can be used by search engines. Apart from the direct application in search engines, query keywords to question conversion has applications in Question Answering (QA) systems, bots communication, Community Question Answer (CQA) websites etc. In CQA websites, when users have typed some keywords to search for questions, one can generate the questions and help them in framing the question using question corpus. Digital assistants can use this technology to refine the intent of query in natural language and help navigate the user to his/her exact needs.

Query to question conversion was first suggested by (Lin, 2008), where he pointed out it's application in CQA websites and richer query expansion. Lin's idea was further extended by (Zhao et al., 2011), in which they have followed a template-based approach. They generate templates from $\langle query, question \rangle$ pairs from search logs and CQA websites and instantiate the template on the input query. At the same time, (Zheng et al., 2011) also used a similar template-based technique. They generate templates from the question collected from CQA websites. They used a single variable templates, which essentially replaced a single word by some placeholder. Thus, the framework heavily relies on existing questions. Another similar work was done by (Kalady et al., 2010) in which they derived question from a well formed sentence using parse tree and named entity recognitions. Their system is limited to certain types of questions. Most of the techniques used to generate question from query are rule-based which are limited by the variety of question rules/templates, grammatical correctness, relevance between query and generated question etc. In this paper we propose a novel statistical approach to generate well-formed question from search keywords. The primary contribution of our work is that we have reduced the problem of query to question conversion into a translation problem. Furthermore, we also have shown how to build $\langle query, question \rangle$ parallel corpus from web search log that retain users' intention between query and question pair.Table 1 shows some of the extracted pairs. We have made a detailed comparison between different translation framework with respect to our problem.

2. Approach

The query to question generation problem can be formally stated as follows: given a sequence of query keywords k $(k_1, k_2, \ldots, k_n)$ we want to generate the corresponding natural language question q $(q_1, q_2, \ldots, q_m)$. This can be seen as a translation problem between source language sentence

Queries	Questions
fever symptoms	What are the symptoms of fever ?
japan capital	What is the capital of japan ?
string to int c#	How to convert string to int in C# ?
cancer types	What are different types of cancer ?

Table 1: Example of queries and related questions

k and target language sentence q. Note that both k and q are in English language while q is a syntactically and semantically correct sentence of the language but k is a grammatically ill-formed query. In this work, we first use a SMT-based (Koehn et al., 2003) approach. We have used the most widely used vanilla Moses[1] to build the SMT system. We consider this as the baseline system and call it **SMT**. We use a NMT-based approach as described by (Bahdanau et al., 2014). Our NMT-based model uses bidirectional RNN with attention model (Cho et al., 2014; Sutskever et al., 2014; Schuster and Paliwal, 1997). Given an input sequence *k* from source language, i.e. queries, we want to generate a sequence q of target language, i.e. questions, which has similar question intent. We want to find the q which maximizes $arg\max_q p(q|k)$. We train a neural model which learns to maximize the conditional probability for sequence pairs in our parallel training corpus. After the model is trained, on giving a sequence *k* from source language, it generates a sequence *q* of target language which maximizes the conditional probability.

Our neural machine translation model consists of an encoder and a decoder. Encoder learns a fixed length representation for variable length input sequences and decoder takes that fixed length learned representation as input and generates the output sequence. For example, for input sequence vectors *k* $(k_1, k_2, \ldots, k_n)$, encoder encodes this into a fixed dimension vector rep. In general RNN's are used, such that :

$$h_t = f(k_t, h_{t-1}) \tag{1}$$

$$rep = z(h_1, h_2, ...h_T) \tag{2}$$

h_t is the hidden state at time t and k_t is input sequence at time t. f and q are non-linear functions. In our model we are using f as LSTM (Hochreiter and Schmidhuber, 1997) and define z as in equation (3):

$$z(h_1, h_2, ..., h_t) = h_t \tag{3}$$

The encoder tries to store the context of the input sequence into vector rep. During training, decoder learns to maximize the conditional probability. Decoder defines a conditional probability over the translation sequence *k* as follows :

$$\begin{aligned} p(q) &= \prod_{t=1}^{T} p(q_t|q_1, q_2, ...q_{t-1}, rep) \\ &= \prod_{t-1}^{T} g(q_{t-1}, s_t, rep) \end{aligned} \tag{4}$$

[1] http://www.statmt.org/moses/

where $q = (q_1, q_2, \ldots, q_T)$ and g is non-linear. We are using attention model (Bahdanau et al., 2014), in which conditional probability gets changed to following:

$$p(q_i|q_1, q_2, \ldots, q_{i-1}, k) = g(q_{i-1}, s_i, rep_i) \tag{5}$$

where s_i is :

$$s_i = g(q_i, s_{i-1}, rep_i) \tag{6}$$

The context vector rep_i is computed as below :

$$rep_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{7}$$

The weight α_{ij} of each annotation h_j is computed by

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{m=1}^{T_x} exp(e_{im})} \tag{8}$$

where

$$e_{ij} = a(s_{i-1}, h_j) \tag{9}$$

This approach allows decoder to decide which part of input it wants to pay attention. We have used BiRNN, which has two function $\overrightarrow{f}$ and $\overleftarrow{f}$, where $\overrightarrow{f}$ reads the input sequence from k_1 to k_T and produces forward hidden states$(h_{f_1}, h_{f_2}, \ldots, h_{f_T})$, i.e. in usual order, and the $\overleftarrow{f}$ reads in opposite direction, i.e. k_T to k_1 and generates hidden backward vectors $(h_{b_1}, h_{b_2}, \ldots, h_{b_T})$. At time t, we get the final hidden vector by concatenating forward as well as backward hidden vector at time t. This way BiRNN helps in storing the context of not only the preceding words but also the following words. Each manuscript should be submitted on white A4 paper. The fully justified text should be formatted in two parallel columns, each 8.25 cm wide, and separated by a space of 0.63 cm. Left, right, and bottom margins should be 1.9 cm. and the top margin 2.5 cm. The font for the main body of the text should be Times New Roman 10 with interlinear spacing of 12 pt. Articles must be between 4 and 8 pages in length, regardless of the mode of presentation (oral or poster).

3. Experimental Setup and Results

First we conduct our baseline experiment using Moses SMT system to compare the results with our NMT-based model. The Moses SMT system uses KenLM (Heafield et al., 2013) as the default language model and MERT (Och, 2003) to reestimate the model parameters. We shall call it **SMT**. In our particular NMT-based approach, we implemented a BiRNN model using LSTM with attention. We used 2 layered deep LSTMs with 512 cells at each layer. We kept the embedding dimension to be 300. Our input vocabulary size for both source and target language, i.e., queries and question had 150,000 words. We used stochastic gradient descent with initial learning rate of 0.5 and learning rate decay factor of 0.99. We kept batch size to be 128 and trained the model for a total of 6 epochs.

3.1. Data Used

In this case, parallel data refers to the (k,q) pair where k is a query with question intent and q is the corresponding natural language question with same question intent. We used Bing's web search logs to create our parallel data. Bing's Search Log stores 3 basic things :

- Queries (k) searched on bing
- The URLs (U) which were shown for those queries in search result page
- URL ($u \in U$) which was clicked by the user for the respective query

We filtered all the queries (k), which landed on a CQA website, which contains some question (q) and its answer. We extracted the question (q) from that clicked CQA website and create the pair (k, q) for our dataset. Our hypothesis behind this was that after querying in any search engine, users click on those links which they find satisfactory and those queries (k) after which a user clicks on a website containing a question (q), can be assumed to have a question intent. To make sure the questions in our dataset are grammatically correct, we only considered reputed CQA websites like WikiAnswers,[2] Quora,[3] and Yahoo Answers.[4] The hypothesis being that moderators on these CQA websites are pretty strict in maintaining quality questions. We only kept (k, q) pairs in which query (k) had less than 10 words to avoid garbage queries. We also made sure that we only select those (k, q) pairs, in which question started with either a "wh" word or other question words (e.g. what, where, who, how, is, can, did, list, are etc.). After all this filtering, we were left with around 13 Million query-question pair (k, q). We used randomly drawn 5000 sentences for test and development set (each 2500 sentences), disjoint from the training data. We found around 50% of the queries have less than 5 words. The average length of the query and question are 5.6 and 8.5, respectively. Also, 85% of the questions are of "what (53%)", "how(21%)","is(6%)" and "who(5%)" types. Fig. 1 plots the Query Length Distributions and Fig. 2 plots the percentage of different types of questions in our dataset.

3.2. Results

In order to evaluate the performance of our system, we have used the most widely used MT evaluation metric BLEU (Papineni et al., 2002). BLEU uses modified n-gram precision between the hypothesis and the reference. Note that the value of BLEU ranges from 0 to 100.

First, in order to estimate the difficulty of the task we conducted an experiment (we shall call it **Identity Model**), we replicated input as the hypothesis translation, since both source query and target question are in English. This gives 19.33 BLEU score. This is due to large amount of vocabulary overlap between the query and its corresponding question.

The baseline SMT gives a BLEU score of 52.49 while NMT system has a BLEU score of 58.63. The NMT system has a 6.14 absolute BLEU point improvement compared to the SMT system. Both SMT and NMT system has a significant improvement over the identity model. The higher BLEU score (> 50) by both SMT and NMT models are achieved due to the overlap between query and question keywords (as reflected in the BLEU score of the identity model).

[2] https://answers.wikia.com/wiki/Wikianswers
[3] https://www.quora.com
[4] https://in.answers.yahoo.com/

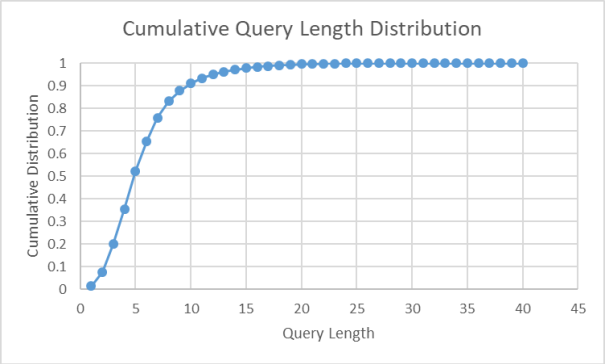


Figure 1: Query Length Cumulative Distribution

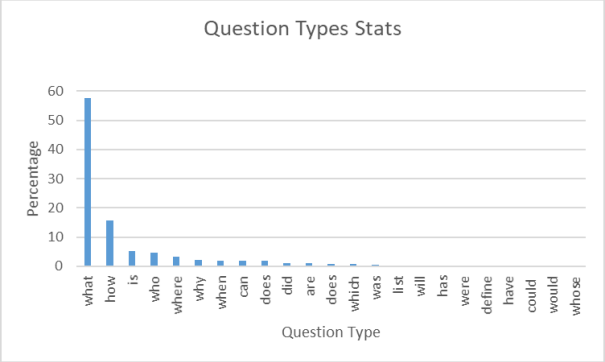


Figure 2: Question Type Distribution

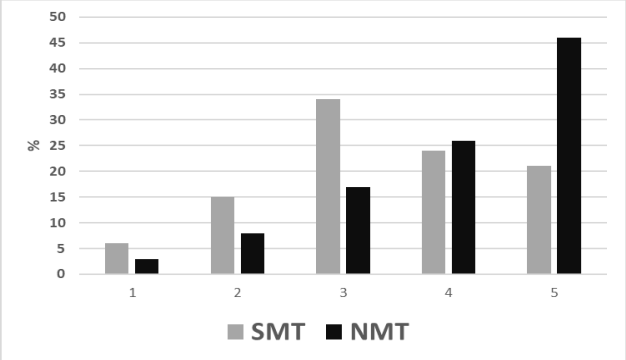


Figure 3: Intent Similarity Score Distribution

3.3. Human Evaluation

We conducted a human evaluation to judge the quality of the generated output. We manually evaluated approximately 1000 query/question pairs with the help of 12 people (more than 5 years of experience of using search engines). For each query-generated output pair, we asked participants following questions :

- Is the question grammatically correct?
- How similar is the intent between query and generated output?

First question was a Yes-No based question and for the second question, participants were asked to judge the question intent similarity on a scale of $1 - 5$ between the pair, with 5 being highly similar. In terms of grammatical correctness of the output generated from the two models, around

Query	Generated Question by SMT	Generated Question by NMT	Golden Truth
grams in 1 lb	how many grams are in 1 lb?	how many grams are in 1 pound?	how many grams are in 1 pound ?
anesthesiologist salary dubai	what is the salary of an anesthesiologist in dubai?	what is the salary of an anesthesiologist in dubai?	how much does an anesthesiologist make in dubai?
richest man in kansas	what is the richest men in kansas?	who is the richest man in kansas?	Who is the most rich man of kansas?
small bone in human body located	what is the small bone in the body located?	where is the smallest bone in human body located?	where is the smallest bone in human body located?
first woman rapper	what was the first woman in the rapper	who was the first woman rapper?	who was the first woman rapper?

Table 2: System Generated Output Produced by Different Models

63% of output generated from SMT were grammatically correct, while with NMT, almost 86% of output were grammatically correct. SMT often make errors due to incorrect choice of question words as shown in examples in Table 2. SMT often choose "what" due to its high frequency in the corpus (cf. Section 3.1). In terms of intent similarity, around 72% of the question generated by NMT model received very high score (4 and 5) in intent similarity by human evaluators, compared to only 45% in case of SMT. Figure 3 shows the distribution of scores both model got from human evaluators. We observed that NMT model performed better than baseline SMT in terms of BLEU score evaluation, as well as human based judgement.

4. Conclusions

In this paper we have described machine-translation based approach for automatic generation of well-formed question from keyword-based query. We used automatically extracted parallel data from search logs to train the models. Our experiments shows that NMT models work better compared to the baseline statistical model. The present model generates the most likely question from a search query which has explicit question intent. For future works we wish to add text from Search Result Page also as input along with the raw query, with the assumption being that the given text will provide more contextual information about the query.

5. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, Bulgaria, August.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Kalady, S., Elikkottil, A., and Das, R. (2010). Natural language question generation using syntax and keywords. In *Proceedings of QG2010: The Third Workshop on Question Generation*, pages 1–10. questiongeneration. org.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Lin, C.-Y. (2008). Automatic question generation from queries. In *Workshop on the question generation shared task*, pages 156–164.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

White, R. W., Richardson, M., and Yih, W.-t. (2015). Questions vs. queries in informational search tasks. In *Proceedings of the 24th International Conference on World Wide Web*, pages 135–136. ACM.

Zhao, S., Wang, H., Li, C., Liu, T., and Guan, Y. (2011). Automatically generating questions from queries for community-based question answering. In *IJCNLP*, pages 929–937.

Zheng, Z., Si, X., Chang, E. Y., and Zhu, X. (2011). K2q: Generating natural language questions from keywords with user refinements. In *IJCNLP*, pages 947–955.

Construction of a Japanese Word Similarity Dataset

Yuya Sakaizawa and Mamoru Komachi
Tokyo Metropolitan University
6-6 Asahigaoka
Hino city, Tokyo 191-0065, Japan
{sakaizawa-yuya@ed.,komachi@}tmu.ac.jp

Abstract

An evaluation of distributed word representation is generally conducted using a word similarity task and/or a word analogy task. There are many datasets readily available for these tasks in English. However, evaluating distributed representation in languages that do not have such resources (e.g., Japanese) is difficult. Therefore, as a first step toward evaluating distributed representations in Japanese, we constructed a Japanese word similarity dataset. To the best of our knowledge, our dataset is the first resource that can be used to evaluate distributed representations in Japanese. Moreover, our dataset contains various parts of speech and includes rare words in addition to common words.

Keywords: word embeddings, distributed representation, word similarity

1. Introduction

Traditionally, a word is represented as a sparse vector indicating the word itself (one-hot vector) or the context of the word (distributional vector). However, both the one-hot notation and distributional notation suffer from data sparseness since dimensions of the word vector do not interact with each other. Distributed word representation addresses the data sparseness problem by constructing a dense vector of a fixed length, wherein contexts are shared (or distributed) across dimensions. Distributed word representation is known to improve the performance of many NLP applications such as machine translation (Chen and Guo, 2015) and sentiment analysis (Tai et al., 2015) to name a few. The task to learn a distributed representation is called representation learning.

However, evaluating the quality of learned distributed word representation itself is not straightforward. In language modeling, perplexity or cross-entropy is widely accepted as a de facto standard for intrinsic evaluation. In contrast, distributed word representations include the additive (or compositional) property of the vectors, which cannot be assessed by perplexity. Moreover, perplexity makes little use of infrequent words; thus, it is not appropriate for evaluating distributed presentations that try to represent them.

Therefore, a word similarity task and/or a word analogy task are generally used to evaluate distributed word representations in the NLP literature. The former judges whether distributed word representations improve modeling contexts, and the latter estimates how well the learned representations achieve the additive property. However, such resources other than for English (e.g., Japanese) seldom exist. In addition, most of these datasets comprise high-frequency nouns so that they tend not to include other parts of speech. Hence, previous data fail to evaluate word representations of other parts of speech, including content words such as verbs and adjectives.

To address the problem of the lack of a dataset for evaluating Japanese distributed word representations, we propose to build a Japanese dataset for the word similarity task.

The main contributions of our work are as follows:

- To the best of our knowledge, it is the first work that constructs a Japanese word similarity dataset.
- The dataset contains various parts of speech and includes rare words in addition to common words.

2. Related Work

In general, distributed word representations are evaluated using a word similarity task. For instance, WordSim353 (Finkelstein et al., 2002), MC (Miller and Charles, 1991), RG (Rubenstein and Goodenough, 1965), and SCWS (Huang et al., 2012) have been used to evaluate word similarities in English. Moreover, Baker et al. (2014) built a verb similarity dataset (VSD) based on WordSim353 because there was no dataset of verbs in the word-similarity task. Recently, SimVerb-3500 was introduced to evaluate human understanding of verb meaning (Gerz et al., 2016). It provides human ratings for the similarity of 3,500 verb pairs so that it enables robust evaluation of distributed representation for verbs. However, most of these datasets include English words only. There has been no Japanese dataset for the word-similarity task.

Apart from English, WordSim353 and SimLex-999 (Hill et al., 2015) have been translated and rescored in other languages: German, Italian and Russian (Leviant and Reichart, 2015). SimLex-999 has also been translated and rescored in Hebrew and Croatian (Mrksic et al., 2017). SimLex-999 explicitly targets at similarity rather than relatedness and includes adjective, noun and verb pairs. However, this dataset contains only frequent words.

In addition, the distributed representation of words is generally learned using only word-level information. Consequently, the distributed representation for low-frequency words and unknown words cannot be learned well with conventional models. However, low-frequency words and unknown words are often comprise high-frequency morphemes (e.g., unkingly → un + king + ly). Some previous studies take advantage of the morphological information to provide a suitable representation for low-frequency words and unknown words (Luong et al., 2013; Soricut and Och,

Currently at JustSystems Corporation.

Sentence	I don't think it is likely to not include these people, or [exclude] まさかこういった方々を対象としない、[排除する]わけではないと思いますが				
Paraphrase	ignore 無視する	ostracize 排斥する	avoid 敬遠する	exclude 排除する	remove 除外する

Figure 1: An example of the dataset from a previous study (Kodaira et al., 2016).

Frequency	1-	101-	1001-	10001-
Verb	239	539	710	598
Adjective	183	322	523	350
Noun	15	63	172	258
Adverb	23	75	80	81

Table 1: The number of parts of speech classified into each frequency.

word 1		word 2		sim.
EN	JA	EN	JA	
close	瞑る	close	つぶる	10
erase	拭き取る	wipe	拭う	8
mopey	塞ぎ込んだ	sick	病んだ	5
investigate	手探る	go	行く	2
fly	とばせる	control	制御できる	0

Table 2: Example of the degree of similarity when we requested annotation at Lancers.

2015). Morphological information is particularly important for Japanese since Japanese is an agglutinative language.

3. Construction of a Japanese Word Similarity Dataset

What makes a pair of words similar? Most of the previous datasets do not concretely define the similarity of word pairs. The difference in the similarity of word pairs originates from each annotator's mind, resulting in different scales of a word. Thus, we propose to use an example-based approach (Table 2) to control the variance of the similarity ratings. We remove the context of word when we extracted the word. So, we consider that an ambiguous word has high variance of the similarity, but we can get low variance of the similarity when the word is monosemous.

For this study, we constructed a Japanese word similarity dataset[1]. We followed the procedure used to construct the Stanford Rare Word Similarity Dataset (RW) (Luong et al., 2013).

We extracted Japanese word pairs from the Evaluation Dataset of Japanese Lexical Simplification (Kodaira et al., 2016). It targeted content words (nouns, verbs, adjectives, adverbs). It included 10 contexts about target words annotated with their lexical substitutions and rankings. Figure 1 shows an example of the dataset. A word in square brackets in the text is represented as a target word of simplification. A target word is not only recorded in the lemma form but also in the conjugated form. We built a Japanese similarity dataset from this dataset using the following procedure.

Word selection: First, paraphrase candidates were extracted from this dataset. Because the construction process of the simplification dataset was divided into a paraphrase acquisition phase and a simplification ranking phase, we simply discarded the simplification rankings from the dataset to obtain paraphrase candidates. Table 1 shows the frequency of extracted words in the Japanese Wikipedia as of May 2015. As shown in the table, low-frequency words are included in the dataset.

Pair construction: Because extracted words are annotated with their paraphrase candidates, we picked up each pair from the candidate as a word pair. Consequently, we acquired 5,051 verb pairs, 4,033 adjective pairs, 1,528 noun pairs and 902 adverb pairs. To balance the numbers of verb and adjective pairs with other parts of speech, we extracted samples at random for verbs and adjectives. Finally, we obtained 1,464 verb pairs and 960 adjective pairs.

We observed that the similarity of the pairs extracted from the dataset of Kodaira et al. (2016) was low without providing contexts; thus, we did not augment the dataset by inserting pseudo-negative instances from WordNet's synsets, as was done in the RW corpus. Another reason why we did not employ the synset from the Japanese WordNet (Isahara et al., 2008) was because its quality was not as good as the English WordNet except for concrete nouns[2].

Human judgment: We opted to use the crowd-sourcing service (Lancers[3]) to hire native Japanese speakers. We asked annotators to assign the degree of similarity for each pair using the same 10-point scale[4]. We used only those annotators who were able to complete at least 95% of their previous assignments correctly. We collected similarity rating for each word pair from ten annotators and defined the average of their annotations as the similarity of the pairs.

Although (Kodaira et al., 2016) gave the annotators the context during annotation, we removed the context and gave only pairs to annotators. We did so because the previous datasets such as VSD and RW did not present any context during annotation[5]. To improve the quality of the annotation, we presented an example of the degree of similarity

[1] https://github.com/tmu-nlp/JapaneseWordSimilarityDataset

[2] It might be because it was translated from the English WordNet. This is why we decided not to translate the existing English word similarity dataset to create a Japanese version.

[3] http://www.lancers.jp

[4] In a crowdsourcing request, we indicated that a similarity of pairs with different notations, such as "write（書いた）" and "write（かいた）" is 10.

[5] Another reason why we did not do so is because the SCWS has a very high variance even though it is annotated with contexts (Table 5).

POS	verb	adj	adv	noun
IAA	0.69	0.67	0.61	0.56

Table 3: Inter-annotator agreements of each POS.

of the pairs during annotation (Table 2). Consequently, we collected 4,851 pairs overall. Table 4 shows an example of a pair from our dataset. Inter-annotator agreements (IAA) of each POS are shown in Table 3. The inter-annotator agreement is the average Spearman's ρ between a single annotator and the average of all others.

4. Discussion

4.1. Comparison to Other Datasets

Table 5 shows how several resources vary. WordSim353 comprises high-frequency words and so the variance tends to be low. In contrast, RW includes low-frequency words, unknown words, and complex words composed of several morphemes; thus, the variance is large. VSD has many polysemous words, which increase the variance. Despite the fact that our dataset, similar to the VSD and RW datasets, contains low-frequency and ambiguous words, its variance is 3.00. The variance level is low compared with the other corpora. We considered that the examples of the similarity in the task request reduced the variance level.

We did not expect SCWS to have the largest variance in the datasets shown in Table 5 because it gave the context to annotators during annotation. At the beginning, we thought the context would serve to remove the ambiguity and clarify the meaning of word; however after looking into the dataset, we determined that the construction procedure used several extraordinary annotators. It is crucial to filter insincere annotators and provide straightforward instructions to improve the quality of the similarity annotation like we did. To gain better similarity, each dataset should utilize the reliability score to exclude extraordinary annotators. For example, for SCWS, an annotator rating the similarity of pair of "CD" and "aglow" assigned a rating of 10. We assumed it was a typo or misunderstanding regarding the words. To address this problem, such an annotation should be removed before calculating the true similarity. All the datasets except for RW simply calculated the average of the similarity, but datasets created using crowdsourcing should consider the reliability of the annotator.

4.2. Analysis

We present examples of a pair with high variance of similarity as shown below:

Aspect of relatedness. (e.g., a pairing of "fast（速い）" and "early（早い）".)

Although they are similar in meaning with respect to the time, they have nothing in common with respect to speed; Annotator A assigned a rating of 10, but Annotator B assigned a rating of 1.

Another example, the pairing of "be eager（懇願する）" and "request（頼む）". Even though the act indicated by the two verbs is the same, there are some cases where they express different degrees of feeling. Compared with "request", "eager" indicates a stronger feeling. There were two annotators who emphasized the similarity of the act itself rather than the different degrees of feeling, and vice versa. In this case, Annotator A assigned a rating of 9, but Annotator B assigned a rating of 2.

Although it was necessary to distinguish similarity and semantic relatedness (Mrksic et al., 2016) and we asked annotators to rate the pairs based on semantic similarity, it was not straightforward to put paraphrase candidates onto a single scale considering all the attributes of the words. This limitation might be relaxed if we would ask annotators to refer to a thesaurus or an ontology such as Japanese Lexicon (Ikehara et al., 1997).

Comparing spell[6]**.** (e.g., a pairing of "slogan（スローガン）" and "slogan（標語）".)

In Japanese, we can write a word using hiragana, katakana, or kanji characters; however because hiragana and katakana represent only the pronunciation of a word, annotators might think of different words. In this case, Annotator A assigned a rating of 8, but Annotator B assigned a rating of 0. Similarly, we confirmed the same thing in other parts of speech. Especially, nouns can have several word pairs with different spellings, which results in their IAA became too low compared to other parts of speech.

Frequency or time expressions. (e.g., a pairing of "often（しばしば）" and "frequently（しきりに）".)

We confirmed that the variance becomes larger among adverbs expressing frequency. This is due to the difference in the frequency of words that annotators imagines. In this case, Annotator A assigned a rating of 9, but Annotator B assigned a rating of 0. Similarly, we confirmed the same thing among adverbs expressing time.

5. Conclusion

In this study, we constructed the first Japanese word similarity dataset. It contains various parts of speech and includes rare words in addition to common words. Crowdsourced annotators assigned similarity to word pairs during the word similarity task. We gave examples of similarity in the task request sent to annotators, so that we reduced the variance of each word pair. However, we did not restrict the attributes of words, such as the level of feeling, during annotation. Error analysis revealed that the notion of similarity should be carefully defined when constructing a similarity dataset.

As a future work, we plan to construct a word analogy dataset in Japanese by translating an English dataset to Japanese. We hope that a Japanese database will facilitate research in Japanese distributed representations.

6. Bibliographical References

Baker, S., Reichart, R., and Korhonen, A. (2014). An Unsupervised Model for Instance Level Subcategorization Acquisition. In *Proceedings of the 2014 Conference*

[6]We indicated these pair's similarity is 10. However, some annotators ignored this instruction. It would be necessary to clean the spellings of paraphrase candidates before requesting similarity annotation.

word 1	EN	follow	exclude	challenge	storm	elucidate	wander
	JA	受け継ぐ	除外する	チャレンジする	しける	明白になる	迷う
word 2	EN	inherit	remove	wish	rough	reflect	stop
	JA	継承する	除去する	望む	あれる	反映される	止める
similarity		9.3	7.3	6.0	5.7	2.7	1.7

Table 4: Examples of verb pairs in our dataset. The similarity rating is the average of the ratings from ten annotators.

Dataset	Variance
WordSim353	3.16
VSD	4.76
RW	5.70
SCWS	8.60
JWSD (our dataset)	3.00

Table 5: Variance of each dataset.

on Empirical Methods in Natural Language Processing (EMNLP), pages 278–289.

Chen, B. and Guo, H. (2015). Representation Based Translation Evaluation Metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 150–155.

Isahara, H., Bond, F., Uchimoto, K., Utiyama, M., and Kanzaki, K. (2008). Development of the Japanese WordNet. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, pages 2420–2423.

Luong, M.-T., Socher, R., and Manning, C. D. (2013). Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning (CoNLL)*, pages 104–113.

Miller, G. A. and Charles, W. G. (1991). Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, 6(1):1–28.

Mrksic, N., Séaghdha, D. Ó., Thomson, B., Gasic, M., Rojas-Barahona, L. M., Su, P., Vandyke, D., Wen, T., and Young, S. J. (2016). Counter-fitting Word Vectors to Linguistic Constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computatinal Linguistics: Human Language Technologies (NAACL-HLT)*, pages 142–148.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual Correlates of Synonymy. *Communications of the ACM*, 8(10):627–633.

Soricut, R. and Och, F. (2015). Unsupervised Morphology Induction Using Word Embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1627–1637.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1556–1566.

7. Language Resource References

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin., E. (2002). Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems (TOIS)*, 20(1):116–131.

Gerz, D., Vulic, I., Hill, F., Reichart, R., and Korhonen, A. (2016). SimVerb-3500: A Large-Scale Evaluation Set of Verb Similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2173–2182.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating Semantic Models with (Genuine) Similarity Estimation. *Computational Linguistics*, 41(4):665–695.

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving Word Representations via Global Context and Multiple Word Prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 873–882.

Ikehara, S., Miyazaki, M., Shirai, S., Yokoo, A., Nakaiwa, H., Ogura, K., Ooyama, Y., and Hayashi, Y. (1997). *A Japanese Lexicon*. Iwanami Shoten.

Kodaira, T., Kajiwara, T., and Komachi, M. (2016). Controlled and Balanced Dataset for Japanese Lexical Simplification. In *Proceedings of the ACL 2016 Student Research Workshop*, pages 1–7.

Leviant, I. and Reichart, R. (2015). Judgment Language Matters: Multilingual Vector Space Models for Judgment Language Aware Lexical Semantics. *CoRR*, abs/1508.00106.

Luong, M.-T., Socher, R., and Manning, C. D. (2013). Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning (CoNLL)*, pages 104–113.

Mrksic, N., Vulic, I., Séaghdha, D. Ó., Leviant, I., Reichart, R., Gasic, M., Korhonen, A., and Young, S. J. (2017). Semantic Specialisation of Distributional Word Vector Spaces using Monolingual and Cross-Lingual Constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

Acquiring Verb Classes Through Bottom-Up Semantic Verb Clustering

Olga Majewska, Diana McCarthy, Ivan Vulić and Anna Korhonen
University of Cambridge
Language Technology Lab
Dept. of Theoretical & Applied Linguistics
Cambridge CB3 9DA, United Kingdom
om304@cam.ac.uk, diana@dianamccarthy.co.uk, iv250@cam.ac.uk, alk23@cam.ac.uk

Abstract

In this paper, we present the first analysis of bottom-up manual semantic clustering of verbs in three languages, English, Polish and Croatian. Verb classes including syntactic and semantic information have been shown to support many NLP tasks by allowing abstraction from individual words and thereby alleviating data sparseness. The availability of such classifications is however still non-existent or limited in most languages. While a range of automatic verb classification approaches have been proposed, high-quality resources and gold standards are needed for evaluation and to improve the performance of NLP systems. We investigate whether semantic verb classes in three different languages can be reliably obtained from native speakers without linguistics training. The analysis of inter-annotator agreement shows an encouraging degree of overlap in the classifications produced for each language individually, as well as across all three languages. Comparative examination of the resultant classifications provides interesting insights into cross-linguistic semantic commonalities and patterns of ambiguity.

Keywords: verb classes, semantic clustering, multilingual NLP

1. Introduction

With the recent advances in automatic lexical acquisition, the need for high-quality evaluation resources is ever growing. Due to the pivotal role played by verbs in sentence structure, the problem of creation of verbal classifications has attracted a lot of attention in natural language processing (NLP). Different approaches to creation of verbal classifications have been proposed, varying with regard to the guiding criteria by which the class architecture is organised, prioritising semantic (WordNet (Miller, 1995; Fellbaum, 1998), FrameNet (Baker et al., 1998), PropBank (Palmer et al., 2005)) or syntactic information (COMLEX (Grishman et al., 1994), VALEX (Korhonen et al., 2006)), or combining the two (Levin, 1993; Kipper et al., 2000; Kipper Schuler, 2005). Kipper Schuler's (2005) VerbNet, grouping English verbs into classes defined by shared meaning components and syntactic behaviour, is one of the richest lexical verb resources currently available, and its utility in various NLP applications has been repeatedly demonstrated (Rios et al., 2011; Windisch Brown et al., 2011; Schmitz et al., 2012; Lippincott et al., 2013; Bailey et al., 2015).

However, creation of a similar resource from scratch, drawing simultaneously on semantic and syntactic criteria, is a challenging and time-consuming task when attempted by annotators without theoretical linguistics background (Majewska et al., 2017). A number of approaches to automatic verb classification have been proposed (Joanis et al., 2008; Sun et al., 2010; Falk et al., 2012; Kawahara et al., 2014; Scarton et al., 2014; Peterson et al., 2016; Vulić et al., 2017), allowing to minimise the time required and eliminate the need to employ trained lexicographers. However, evaluation of such systems relies on the availability of gold standard classes, and these are still lacking for a great majority of languages.

In light of these challenges and the high demand for verbal resources, this paper investigates whether semantic verb classes can be reliably acquired from non-expert native speakers based solely on verb semantics and following simple instructions, which, to the best of our knowledge, is the first evaluation of this approach. Drawing on the hypothesis that syntactic and semantic behaviour of verbs are tightly interrelated (Pinker, 2013; Jackendoff, 1992; Levin, 1993), we simplify the classification task by eliminating the need to refer to explicit syntactic knowledge and assess whether intuitive native-speaker perception of closeness of verb meaning provides enough guidance to produce consistent verb classifications. This will allow us in future work to examine the relationship between semantics and syntactic behaviour of the class members. Previous classifications have used syntactic behaviour to guide the construction of verb classification but this necessitates linguistic training.

In order to examine the potential of manual semantic clustering in different languages, we carried out verb clustering experiments with native speakers of English, Polish, and Croatian. We describe the set-up of the task in Section 2. Subsequently, we analyse the inter-annotator agreement for each language individually and examine the overlap between classes cross-lingually. Section 3 includes the results of this evaluation. Finally, in Section 4, we discuss observations made with respect to the easily classifiable verbs and those which caused problems in all the languages considered, which shed light on cross-linguistic semantic commonalities and polysemy patterns.

2. The Semantic Verb Clustering Task

The task involved a group of 8 native-speaker participants without formal linguistics training, 3 annotators for English and Polish, and 2 in Croatian, who performed soft clustering of a sample of verbs in their native language based on the verbs' semantic similarity. The verb samples were created as follows: first, a sample of 267 English verbs was au-

	English				Polish				Croatian		
	A1	A2	A3	**Ave**	A1	A2	A3	**Ave**	A1	A2	**Ave**
Number of classes	61	77	58	**65.3**	47	46	35	**42.7**	88	76	**82.0**
Average class size	4.4	3.5	4.6	**4.1**	5.7	5.8	7.6	**6.3**	3.0	3.5	**3.3**
Time spent [hours]	2	1	3	**2.0**	3	3	3	**3.0**	3	2	**2.5**

Table 1: Results and statistics of semantic clustering of 267 verbs for English, Polish, and Croatian, for each annotator (A1-A3) and the average scores for each language (Ave)

tomatically extracted from the pool of SimVerb-3500 (Gerz et al., 2016) verb types. The verbs were sampled so as to ensure that the top 34 VerbNet classes (according to the number of verbs in the class) from SimVerb-3500 are represented by at least 5 member verbs each, to guarantee 'clusterability' of the verbs presented to the annotators. Next, the English sample was translated by native-speaker translators into Polish and Croatian, and the three samples were manually inspected.

Before the start of the task, the annotators were provided with instructions (Appendix) and a list of 267 verbs in a text file, presented in random order, one word in each line. Since the goal was to keep the task as simple as possible for participants without linguistics training, the annotation guidelines were intentionally minimal: they instructed the annotators to put verbs together using a spreadsheet program (e.g. Microsoft Excel) so as to form groups containing verbs that are used to express similar or related meanings. The groups could vary in size, but annotators were asked to aim for at least 3-5 members. A verb could be put in more than one class (e.g. when it had several different meanings), and any verb which did not seem to fit with any group could be placed in a 'Miscellaneous' class. Annotators were encouraged to make a note of any relationship links between groups where they felt the meanings of member verbs were in some way related, e.g. a bidirectional link between similar groups, or a unidirectional link between a broader class and its subclass(es).

3. Results and Inter-Annotator Agreement

The results and statistics of the semantic clustering task for each annotator individually and across annotators, in each of the three languages considered, are reported in Table 1. On average, it took 2.5 hours to complete the task across all annotators, ranging from 1 to 3 hours. The average number of classes obtained was 65.3 for English, 42.7 for Polish, and 82 for Croatian, with class size ranging from the average of 3.3 member verbs in Croatian to 6.3 members in Polish.

3.1. Percentage IAA

In order to measure the overlap between classifications produced by annotators for each language individually and across languages, we calculate percentage inter-annotator agreement (% IAA) for all pairings of verbs. First, we extract all the pairs of verbs on which there is perfect agreement (i.e. all annotators either grouped them together or not), for each of the languages independently, and compute the ratio of observed agreement pairs to all the possible pairings of verbs. Subsequently, we repeat the same procedure for all the English, Polish and Croatian annotators together.

	English	**Polish**	**Croatian**	**All**
% IAA	88.5%	92.5%	97.8%	79.9%

Table 2: The percentage inter-annotator agreement calculated for all possible pairings of verbs, for each language individually and across the three languages

The computations yield a high inter-annotator agreement score for each of the languages, with 88.5% observed for English, 92.5% in Polish, and 97.8% in Croatian (Table 2). The percent inter-annotator agreement calculated across all three languages is 79.9%. It must be noted that the very high agreement score obtained for Croatian, compared to the other two languages, is likely to be due to the smaller average class size. Since many Croatian classes included as few as 3 member verbs, there was a large number of pairs of verbs which were not classified together. Whenever the annotators agreed on not putting two verbs together, that pair constituted an 'agreement' pair for the purposes of inter-annotator agreement calculation. The smaller classes gave rise to the somewhat inflated % IAA score for Croatian because of the larger number of true negatives (verbs that are correctly found not to go in the same class). Its inclusion of true negatives gives % IAA rather high scores generally. In order to address this issue, in the following section (3.2) we calculate inter-annotator agreement using a different evaluation metric, Fuzzy B-Cubed for overlapping clusters (Amigó et al., 2009; Jurgens and Klapaftis, 2013)[1], which avoids the problem of inflation due to scoring true negatives.

3.2. B-Cubed for Overlapping Clusters

In our verb-clustering task, the total number of classes was left unspecified and the annotators were free to put a single verb in as many different classes as they felt was appropriate, whenever they recognised it had more than one distinct sense. In order to adequately evaluate the results, the evaluation measure applied to our data had to be able to accommodate these characteristics of the task. We chose the B-Cubed metric (Bagga and Baldwin, 1998) extended by Amigó et al. (2009) to compare overlapping clusters, and

[1]We used the Fuzzy B-Cubed implementation of Jurgens and Klapaftis (2013) but did not associate the clusters with weights, and therefore the metric is equivalent to that of Amigó et al. (2009).

	Average B-Cubed
English	0.262
Polish	0.338
Croatian	0.172
All	0.205
1c1inst	0.0
All-instances, One class	0.069

Table 3: The average B-Cubed F-score (i.e. harmonic mean of B-Cubed precision and recall) calculated for all possible pairings of annotators, for each language individually and across the three languages, and for two SemEval baselines: 1c1inst and All-instances, One class

by Jurgens and Klapaftis (2013) to fuzzy clusters, used to evaluate the performance of Word Sense Induction systems in SemEval tasks (Jurgens and Klapaftis, 2013).

The B-Cubed metrics (B-Cubed precision and recall) compare two clusterings (say, X and Y) at the item level: for an item *i*, precision measures how many items sharing a cluster with *i* in clustering X are placed in its cluster in clustering Y; whereas B-Cubed recall measures how many items sharing a cluster with *i* in Y are also placed in its cluster in X, with the final B-Cubed score equivalent to the harmonic mean of the two values.

In our task, rather than comparing each clustering against a gold-standard set of classes, we calculate the B-Cubed score for each pair of clusterings produced by the annotators, for each language individually and across all three languages. We report the results of this evaluation in Table 3. The highest agreement score is observed for Polish, where the average B-Cubed F-score is 0.338. Less overlap was found between the clusterings produced for English (0.262), with the lowest B-Cubed F-score obtained for the Croatian clusterings (0.172). The low score reported for Croatian is especially noteworthy in the light of the inflated percent agreement result reported in section 3.1. With percent agreement computed for every possible pairing of verbs, based on a binary choice between two verbs being either clustered together or kept separate, the two annotators seemed to agree in a vast majority of their clustering decisions. Applying an alternative evaluation metric allows us to identify the bias from scoring true negatives, i.e. all cases in which the annotators agreed that two verbs should not be clustered together. As predicted, this inflation is particularly high in the case of Croatian due to the small average class size compared to the other two languages. Indeed, manual inspection of the classes produced by the Croatian annotators shows that in some cases the minimum class size of 3-5 members recommended by the guidelines was not adhered to.

The average B-Cubed F-score calculated for all possible pairings of annotators across the three languages (using translational equivalents for cross-lingual comparisons) is 0.205. Notably, the average cross-lingual agreement score is higher than the value obtained for Croatian itself, which suggests a promising degree of overlap between English and Polish classes (the average B-Cubed F-score for these two languages is 0.237).

Keeping in mind the differences in the nature of the present task and a Word Sense Induction task (which can be seen as an example of unsupervised clustering, with usages of a word grouped into clusters, each representing uses of the same meaning (Jurgens and Klapaftis, 2013)), comparing our results against the scores obtained by the SemEval participating systems may help interpret the reported values. Overall, the top-performing system surpasses our highest result for Polish (scoring 0.483), on the other hand, in the multi-sense setting (i.e. on instances labeled with multiple senses), the best performing system achieves the B-Cubed score of 0.134, a result below the lowest agreement score in our task.

In order to make the comparison more meaningful, we calculate two SemEval baselines for our task: (1) 1c1inst, where each instance is assigned to its own class, and (2) All-instances, One class, which assigns all instances to a single class. The result for the first baseline, 0.0, is the same as in SemEval, and a natural consequence of B-Cubed since there are no pairs within a class. However, while the overall performance of the All-instances, One sense baseline in SemEval surpasses its best participating system (achieving the score of 0.623), the result for this baseline on our verb clustering is much lower (0.069), suggesting the task is significantly more difficult, due to the high number of clusters. And yet, despite the greater difficulty of the task, the agreement between our annotators exceeds the performance of the baselines, which is an encouraging outcome.

As noted earlier, our verb clustering task and the SemEval task are different. The SemEval annotation was performed using predefined senses for graded-tagging (on a Likert scale) and the systems' clusters were compared to clusters induced from these graded sense-annotations. Since the senses for consideration by the annotators were defined in WordNet this is not comparable with our task of clustering verbs. Our task allowed for complete flexibility in the number of classes, which resulted in varying levels of granularity (e.g. Croatian classifications had up to 88 clusters, while the smallest Polish clustering had 35), and a higher number of clusters overall with respect to the SemEval task. The different set-ups of the two tasks entail different levels of difficulty and hence, different agreement scores, and this is reflected in the results obtained for the same baselines discussed above. What is important, our analysis constitutes the first attempt at measuring agreement on clustering of verbs performed by humans.

The encouraging degree of overlap observed between the classifications produced in our manual clustering task, particularly for Polish and English, suggests that there are consistent patterns in how humans group verbs based on their semantic similarity, not only in each language independently, but also across languages from different language families. Collecting more classification data for Croatian, while controlling for class size (as per the minimum class size stated in the guidelines), will allow to verify whether the lower B-Cubed score reported for that language has to do with the peculiarities of the collected data or is indicative of a general greater difficulty in classifying verbs in

Croatian with respect to the other two languages. Extending the experiments to other diverse languages will allow to investigate even further the extent to which those regularities are observed cross-linguistically; however, these are already promising inter-annotator agreement results for a multilingual semantic task.

3.3. Cross-Linguistic Areas of Overlap

Manual inspection of the resultant classes from all annotators allows us to observe what class types and semantic domains are shared by the three languages. Five classes emerge which share the core of at least 2 member verbs across annotators in all three languages (with extra members added by some annotators) and can be described with the following labels (denoting 'verbs of_'): 'looking', 'cooking', 'existing', 'movement in water', 'emitting sound'. The total of 30 classes can be identified where the core of at least 2 member verbs is shared by at least two languages (by all annotators), and whose members belong to the same semantic domains across languages (but with more variation in specific member verbs recorded by individual annotators). In section 4, we look more closely at semantic patterns observable in all three languages and discuss which aspects of verb meaning make the classification task consistently easier or harder, regardless of the language in question.

4. Analysis and Discussion

Despite the encouraging inter-annotator agreement scores, several issues affecting the agreement and overlap between the resultant classifications could be observed. First of all, as the task did not impose a fixed number of classes, the levels of granularity varied between annotators: the difference between the minimum and maximum number of classes equals 12 for Polish and Croatian, and 19 for English. This discrepancy is even more noticeable across languages: while Polish annotators grouped verbs into 35-47 classes, Croatian classifications comprise between 76-88 verb classes. As the task consisted in grouping verbs into flat classes, the resultant classifications do not capture hierarchical relationships between verb groups (these could, however, be signalled as 'relationship links', as noted above). Therefore, potential inclusion of one class by another (e.g. in the case of 'movement' verbs, which in Croatian are split into two small classes, depending on the medium (water vs ground), and are grouped together in one broader 'movement' class in Polish (*swim, dive, walk, crawl*)), is interpreted as class disjunction in automatic pairwise evaluation, which results in a lower overlap between the classifications. What is more, in some cases distinct patterns of ambiguity in the languages considered resulted in different clustering decisions: for example, while in English two senses of the verb 'shine' (i.e. emit light and polish (a shoe)) were considered, resulting in pairings 'shine'-'glow' and 'shine'-'brush', only the former sense is available in Polish and Croatian.

4.1. Problematic and Easily Classifiable Verbs

In order to investigate whether some verbs are inherently easier or harder to classify, and examine to what extent this is observed across languages, we extracted all the pairs of verbs on which there is perfect agreement and those on which the annotators disagreed for each language individually, and examined the overlap between these groups of verbs across the three languages. This allowed to identify 72 'problematic' and 24 'easy' verbs, shared by the three languages. Manual inspection of these groups let us make a number of observations regarding the aspects of verb meaning which pose problems or make them easier for humans to classify, regardless of the language considered.

4.1.1. 'Problematic' Verbs

Most of the verbs which ended up in the 'problematic' group share the characteristic of having a broad, vague or abstract meaning, sometimes with several related senses which allow them to appear in a number of slightly different contexts. For example, annotators in all three languages disagreed on how to classify verbs such as *affect, treat, engage* or *spare*. What is more, some display a degree of semantic vacuity, that is, have little semantic content of their own and tend to express a more precise meaning when combined with some other word (e.g. a noun), with which they form a predicate, such as *make* or *have*, examples of the so-called 'light verbs' (Jespersen, 2013). Inspection of the VerbNet classes from which the 'problematic' verbs were sampled revealed that the 'Change of State' class (45) is particularly often represented. Although verbs such as *slip, vary* and *tumble* belong to the same VerbNet subclass (45.6-1, 'calibratable change of state'), their meanings are not intuitively similar. Moreover, each has several senses, which is reflected in the fact that each participates in a number of distinct VerbNet classes. Understandably, this results in more variation in clustering decisions, as different annotators are likely to take different verb senses into consideration, and consequently, produce divergent classifications.

4.1.2. 'Easy' Verbs

Verbs which lend themselves better to manual semantic classification are those with narrow, concrete meanings, for example, verbs describing sounds (*chirp, buzz, roar*) or those belonging to a clearly defined semantic field, e.g. 'cooking' verbs (*fry, bake, cook*). Synonymous verbs such as *study* and *examine*, or *observe* and *stare*, were also among those on which the same clustering decisions were made across annotators, in all three languages. Interestingly, there was full agreement on antonymous pairs such as *vanish* and *appear*, which were consistently grouped together in all languages. As discussed in lexical semantics literature (Cruse, 1986), antonyms have a paradoxical nature: on the one hand, they constitute the two opposites of a meaning continuum, and therefore could be seen as semantically remote, on the other hand, they are paradigmatically similar, having almost identical distributions, and hence seem closely related. Despite these conflicting properties of antonyms, humans seem to intuitively recognise their relatedness and consistently group them together, as semantically similar. The perception of relatedness overrides the sense of 'oppositeness' and being maximally distant along a dimension of meaning, and opposites end up clustered together. This regularity is observed in the case of pairs of relational antonyms, i.e. verbs which describe an

event from opposite points of view, for example, *lend* and *borrow*, which differ along only one dimension of meaning, that is, the direction of the action (the object of the verb either travels away from the participant (*A lends something to B*) or towards the participant (*B borrows something from A)*), and are essentially identical with regard to all other features, which makes them appear semantically close.

4.2. Semantic Similarity versus Relatedness

The importance of distinguishing between the concepts of semantic similarity (e.g. *cup* and *mug*) and relatedness (e.g. *coffee* and *cup*) has been noted in the literature (Hill et al., 2015), and the analysis of our data provides more evidence illustrating the influence of loose association on how humans conceptualize similarity between words, and the difficulty of keeping similarity and relatedness apart. In all three languages we can observe instances of what can be described as a 'storyline approach' to judging semantic similarity and verb classification. This is particularly noticeable in Croatian classifications, where several classes formed by the annotators group verbs describing quite different actions, linked via loose thematic ties: (1) *marry, conquer, approach, move*, where putting semantically dissimilar verbs *marry* and *move* together seems to suggest an underlying 'storyline' with courtship leading to marriage and moving house; (2) *visit, communicate, treat, operate*, where the association of verbs *visit* with *treat* and *operate* brings to mind a hospital visit, or (3) *finish, frame, announce, submit*, which can be seen as belonging to an 'academic' thematic domain. Relying on association rather than actual consideration of semantic components of verbs' meaning is visible in the cases where a class contains verbs which express a consequence of the action or state described by other verbs in the same class, e.g. *glow, shine, squint* in one of the Polish classifications, with 'squinting' being a reaction to 'glowing' or 'shining', or *ache, hurt, kick, rub, cry* in Croatian. Although verb groupings in which loosely thematically related verbs are classified together are in the minority, their presence in our data suggests that, in order to obtain classes based solely on semantic similarity judgments, unbiased by loose association, the annotation guidelines should explain the similarity-relatedness distinction and instruct the annotators accordingly.

4.3. Polysemy

An in-depth investigation of the resultant classes also offers an insight into the patterns of polysemy in the three languages considered. In our task, the annotators could accommodate a verb's ambiguity by placing it in several different classes, putting each of its distinct senses in a separate cluster. However, since the annotators were provided with just word forms and the senses were not specified a priori, there were some discrepancies in which senses were identified, across annotators and, expectedly, across the different languages, which led to a lower cross-lingual agreement in the resultant classes. For example, the Croatian translation of the English verb 'to vary', *odstupati*, expresses not only the sense of 'differing', but also 'withdrawing', unavailable in English or Polish, which explains why it was placed in the same class with *move* and *renounce* only by the Croatian annotators. Analogously, the Croatian equivalent of *remark* (*primijetiti*), ambiguous between senses 'to comment' and 'to notice', ended up together with verbs such as *look, stare, observe*, while in Polish and English it was grouped with verbs of 'communicating'. Similarly, the Polish translation equivalent of the verb *weave* (*pleść)* is ambiguous between two senses, 'to interlace' and 'to blabber', and was grouped both with *join* and *combine*, and with *tell* and *communicate*, while no such ambiguity was recorded in English and Croatian. Finally, while two senses of the verb *sway* ('to move rhythmically from side to side' and 'to control or influence') are available in English, only the former is recognised in the Polish and Croatian classifications and its translation equivalents are never grouped together with verbs such as *convince* or *persuade*, as it is the case in English.

In a task such as ours, where guidelines were intentionally restricted, so as to avoid imposing any preconceived semantic categories or classification structure onto the annotators and elicit possibly spontaneous similarity judgments, such discrepancies in detecting ambiguity are inevitable. In order to have more control over which sense of a given verb is taken into consideration in the clustering task, word senses rather than word forms would have to be provided at the start of the task. Such a set-up would also allow comparison of the elicited classes with the existing multilingual sense inventories, like Open Multilingual WordNet (Bond and Foster, 2013) or BabelNet (Navigli and Ponzetto, 2012). Since the aim of the present study was to elicit judgments on basic word forms, without any guidance as to the different word senses available, such comparisons are beyond the scope of this study; however, in future work we intend to extend this analysis and compare our findings against the resources available.

5. Conclusion

We have presented the first cross-lingual analysis and evaluation of semantic clustering of verbs by non-expert human annotators. The inter-annotator agreement scores reported for English, Polish, and Croatian are encouraging and demonstrate that verbs can be reliably classified by humans without linguistics background. What is important, this suggests that there is potential to create verb classifications starting from a simple, purely semantic task. Moreover, the degree of overlap in the resultant classifications observed across languages implies that there are cross-linguistic commonalities and shared meaning components governing the semantic organisation of verbs. A cross-lingual scrutiny of low-agreement verbs and those on which annotators made identical clustering decisions, allowed us to investigate to what extent the same verbs are problematic and whether some verbs are inherently easier to classify. Manual inspection of the thus identified 'easy' and 'problematic' verbs provided interesting insights into the aspects which may affect 'clusterability' of verbs across different languages. The present study opens up several avenues for future work. First of all, we would like to extend the study to other languages from different language families, as well as test the applicability of our bottom-up semantic-based

approach to larger verb samples. This would allow us to expand the cross-lingual analysis of the semantic classes obtained and their underlying properties, and investigate how our findings about 'easy' and 'problematic' verbs are reflected in other language resources and corpora. What is more, a comparison of our classifications against VerbNet could provide interesting insights into where speakers' intuitions about word usage most diverge from semantic boundaries drawn by lexicographers, and how the existing verb resources could be improved to better reflect speakers' perceptions about verbs' semantic characteristics and behaviour. Moreover, a comparative analysis of our data against an output of an automatic clustering algorithm would allow us to investigate whether the manual classification task can be (partly) substituted with a semi-automatic one, with an initial rough clustering based on verbs' distributional properties extracted from a large corpus, and subsequently verified by a human annotator.

6. Acknowledgements

We gratefully acknowledge the funding support of the Economic and Social Research Council [PhD Award Number ES/J500033/1] and the European Research Council (ERC) [Consolidator Grant 648909)]. We thank Daniela Gerz for helpful discussions on this work and the three anonymous reviewers for their comments and suggestions.

7. Bibliographical References

Amigó, E., Gonzalo, J., Artiles, J., and Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486.

Bagga, A. and Baldwin, B. (1998). Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Granada, Spain.

Bailey, D., Lierler, Y., and Susman, B. (2015). Prepositional phrase attachment problem revisited: how verbnet can help. In *Proceedings of the 11th International Conference on Computational Semantics (IWCS)*.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *Proceedings of ACL-COLING*, pages 86–90.

Bond, F. and Foster, R. (2013). Linking and extending an open multilingual wordnet. In *Proceedings of ACL*, pages 1352–1362.

Cruse, D. A. (1986). *Lexical semantics*. Cambridge University Press.

Falk, I., Gardent, C., and Lamirel, J.-C. (2012). Classifying french verbs using french and english lexical resources. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 854–863. Association for Computational Linguistics.

Fellbaum, C. (1998). *WordNet*.

Gerz, D., Vulić, I., Hill, F., Reichart, R., and Korhonen, A. (2016). Simverb-3500: A large-scale evaluation set of verb similarity. *arXiv preprint arXiv:1608.00869*.

Grishman, R., Macleod, C., and Meyers, A. (1994). COMLEX syntax: Building a computational lexicon. In *Proceedings of COLING*, pages 268–272.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Jackendoff, R. S. (1992). *Semantic structures*.

Jespersen, O. (2013). *A Modern English Grammar on Historical Principles: Volume 5, Syntax (fourth Volume)*, volume 5. Routledge.

Joanis, E., Stevenson, S., and James, D. (2008). A general feature space for automatic verb classification. *Natural Language Engineering*, 14(3):337–367.

Jurgens, D. and Klapaftis, I. P. (2013). Semeval-2013 task 13: Word sense induction for graded and non-graded senses. In *SemEval@ NAACL-HLT*, pages 290–299.

Kawahara, D., Peterson, D., and Palmer, M. (2014). A step-wise usage-based method for inducing polysemy-aware verb classes. In *ACL (1)*, pages 1030–1040.

Kipper, K., Dang, H. T., and Palmer, M. S. (2000). Class-based construction of a verb lexicon. In *Proceedings of AAAI*, pages 691–696.

Kipper Schuler, K. (2005). *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. thesis.

Korhonen, A., Krymolowski, Y., and Briscoe, T. (2006). A large subcategorization lexicon for natural language processing applications. In *Proceedings of LREC*, volume 6.

Levin, B. (1993). *English verb classes and alternation, A preliminary investigation*.

Lippincott, T., Rimell, L., Verspoor, K., and Korhonen, A. (2013). Approaches to verb subcategorization for biomedicine. *Journal of biomedical informatics*, 46(2):212–227.

Majewska, O., Vulić, I., McCarthy, D., Huang, Y., Murakami, A., Laippala, V., and Korhonen, A. (2017). Investigating the cross-lingual translatability of verbnet-style classification. *Language Resources and Evaluation*.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Palmer, M., Kingsbury, P., and Gildea, D. (2005). The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Peterson, D. W., Boyd-Graber, J., Palmer, M., and Kawhara, D. (2016). Leveraging verbnet to build corpus-specific verb clusters. *The SEM 2016 Organizing Committee*, page 102.

Pinker, S. (2013). *Learnability and cognition: The acquisition of argument structure*. MIT press.

Rios, M., Aziz, W., and Specia, L. (2011). Tine: A metric to assess mt adequacy. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 116–122. Association for Computational Linguistics.

Scarton, C., Sun, L., Schuler, K. K., Duran, M. S., Palmer,

M., and Korhonen, A. (2014). Verb clustering for Brazilian Portuguese. In *Proceedings of CICLing*, pages 25–39.

Schmitz, M., Bart, R., Soderland, S., Etzioni, O., et al. (2012). Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics.

Sun, L., Poibeau, T., Korhonen, A., and Messiant, C. (2010). Investigating the cross-linguistic potential of VerbNet-style classification. In *Proceedings of COLING*, pages 1056–1064.

Vulić, I., Mrkšić, N., and Korhonen, A. (2017). Cross-lingual induction and transfer of verb classes based on word vector space specialisation. In *Proceedings of EMNLP*.

Windisch Brown, S., Dligach, D., and Palmer, M. (2011). VerbNet class assignment as a WSD task. In *Proceedings of IWCS*, pages 85–94.

Appendix: Classification Guidelines

The annotators were presented with the following classification guidelines, along with the list of 267 verbs in their native language, at the start of the task:

Here is a long list of verbs, with one verb in each line.
Please put them together in groups where you feel they are used to express similar or related meanings. For example you may feel 'throw, kick, punch' are related, or 'speak, talk, and write'. These groups can be broader (more members) or narrower (fewer members) but any group must have at least 3-5 members. Aim for cohesive small groups if possible and you can add a 'relationship link' from each group to any other groups if you feel there are relationships between the two groups. The relationship could be similar-to (bidirectional) or broader-than (unidirectional). Any verbs you cannot find a good place for, please put in a 'Miscellaneous' group. There is no problem with putting a verb in more than one class if it fits all, for example because a verb may have several different meanings.
We suggest using Microsoft Excel or a related spreadsheet program (e.g. Google Sheets) to constantly have an overview of current groups. The expected output is: (i) groups of verbs according to your own criteria (see above), (ii) relationship links between groups as also discussed above. To facilitate the linking, you can provide simple labels for each group, e.g., Group 1, Group 2.
There is not necessarily a fully correct solution to this task and a perfect grouping. It is perfectly reasonable to use your intuition or gut feeling as a native speaker while working on this task.

Constructing High Quality Sense-Specific Corpus and Word Embedding via Unsupervised Elimination of Pseudo Multi-sense

Haoyue Shi[1,2], **Xihao Wang**[1,3], **Yuqi Sun**[1,3], **Junfeng Hu**[1,4]
[1]School of Electronics Engineering and Computer Science
[2] Department of Machine Intelligence
[3] Department of Computer Science and Technology
[4] Key Laboratory of Computational Linguistics, Ministry of Education
No 5. Yiheyuan Road, Beijing China, 100871
{hyshi, victorwonder, sun_yq, hujf}@pku.edu.cn

Abstract

Multi-sense word embedding is an important extension of neural word embeddings. By leveraging context of each word instance, multi-prototype version of word embeddings were accomplished to represent the multi-senses. Unfortunately, this kind of context based approach inevitably produces multiple senses which should actually be a single one, suffering from the various context of a word. (Shi et al., 2016) used WordNet to evaluate the neighborhood similarity of each sense pair to detect such pseudo multi-senses. In this paper, a novel framework for unsupervised corpus sense tagging is presented, which mainly contains four steps: (a) train multi-sense word embeddings on the given corpus, using existing multi-sense word embedding frameworks; (b) detect pseudo multi-senses in the obtained embeddings, without requirement to any extra language resources; (c) label each word in the corpus with a specific sense tag, with respect to the result of pseudo multi-sense detection; (d) re-train multi-sense word embeddings with the pre-selected sense tags. We evaluate our framework by training word embeddings with the obtained sense specific corpus. On the tasks of word similarity, word analogy as well as sentence understanding, the embeddings trained on sense-specific corpus obtain better results than the basic strategy which is applied in step (a).

Keywords: multi-sense word embedding, word sense discovery, pseudo multi-sense

1. Introduction

Distributional word representations (Bengio et al., 2003; Collobert and Weston, 2008; Mnih and Hinton, 2009; Mikolov et al., 2013b) embed words into a high dimensional space, where the cosine value (or Euclidean distance) between two vectors can somehow represent the similarity of two words. Multi-sense word embedding is an intuitive extension of distributional representations of words. Most of existing works distinguish different senses of words by their contexts (Schütze, 1998; Huang et al., 2012; Pina and Johansson, 2015; Neelakantan et al., 2014; Li and Jurafsky, 2015; Cheng and Kartsaklis, 2015). Besides proposing a Chinese Restaurant Process (CRP) model for multi-sense word embeddings, Li and Jurafsky (2015) also discussed how helpful the multi-sense word embedding methods are to improve natural language understanding, and proved that they do help on some tasks. However, these methods have a common defect. Existing multi-sense word embedding models generate large amount of pseudo multi-senses (Shi et al., 2016). This phenomenon leads to the vagueness of such distributional word representations. It also has an intuitive explanation. Here shows three appearances of the word "cat(s)" from Wikipedia:

- In many countries, **cats** are believed to have nine lives, but in Italy, Germany, Greece, Brazil and some Spanish-speaking regions, they are said to have seven lives, while in Turkish and Arabic traditions, the number of lives is six.
- Female **cats** are seasonally polyestrous, which means they may have many periods of heat over the course of a year, the season beginning in spring and ending in late autumn.
- **Cats** hunt small prey, primarily birds and rodents, and are often used as a form of pest control.

The sentences present large topic shift, but we human beings can still easily determine the three "cat"s have exactly the same meaning, which is the kind of animal. In contrast, it is too easy for a context-based method to view them as different senses and learn a separate vector for each, if there is not an explicit constraint to penalize such behavior. Shi et al. (2016) detected pseudo multi-sense with the help of WordNet (Miller, 1995), and then trained a linear transformation which aims to maximize the similarity of detected pseudo multi-senses. On the transformed word-embedding space, they successfully observed improved performance on both word similarity and analogy tasks.

Considering the probable large cost obtaining external knowledge like WordNet for rare languages, we design a novel unsupervised pseudo multi-sense detection method, which evaluates neighborhood similarity of two sense using global (one-word-one-sense) word vectors. We then present a framework for sense tagging, which combines multi-sense word embedding and pseudo multi-sense detection. For tokenized corpus, our framework contains the following steps:

1. Train multi-sense word embeddings on corpus, using a specific basic (multi-sense word embedding) model.
2. Detect pseudo multi-sense in the embeddings.
3. Select sense for each instance in the corpus. Pseudo multi-sense would be tagged as one sense.

4. Re-train multi-sense word embeddings on corpus, with pre-selected sense tags in step 3.

5. Repeat step 2-4 until no pseudo multi-sense is detected.

To the best of our knowledge, we are the first to create a framework for unsupervised sense-tagging regarding elimination of pseudo multi-sense. Our framework is able to accept any context-based multi-sense word embedding method as a basic model. Moreover, for those word-based languages which are lack of studies, our framework would also shed light on automatic sense discovery, which has less redundancy than the basic multi-sense word embedding model.

2. Related Work

Multi-Sense Word Embedding Neural multi-sense word embedding is a well-studied problem after the emergence of neural word embeddings (Huang et al., 2012; Pina and Johansson, 2015; Neelakantan et al., 2014; Li and Jurafsky, 2015; Cheng and Kartsaklis, 2015; Liu et al., 2015; Qiu et al., 2016). Most of them select sense for word instances with respect to their context. The producing of multi-sense word embeddings is also the procedure of word-sense discovery. In our framework, all of the models mentioned above could be applied as the basic model, which would be improved after re-training with pseudo multi-sense detection and tagging.

Different from those global word embedding models (Pennington et al., 2014; Mikolov et al., 2013b), most of the existing multi-sense word embeddings contain three types of vectors:

- *global vector.* Each word in vocabulary is embedded into a high dimensional space, aka, one word one vector.

- *context vector.* For each instance of a word in the corpus, we can compute its *instance context vector* by averaging the *global vector* of its context words. By applying SoftMax to the dot production between *instance context vector* and all given *context vector* of senses, we could obtain the probability for the word instance to be classified to each sense.

- *sense vector.* Each sense of word is embedded to the sense vector space, which is the main part of multi-sense word embeddings.

Another type of multi-sense word embeddings introduce external knowledge base for accurate sense generation (Iacobacci et al., 2015; Chen et al., 2014; Pelevina et al., 2017). However, they are somehow limited as such external knowledge may be lack for languages other than English.

From the perspective of sense definition, Flekova and Gurevych (2016) proposed a model to learn representations for supersenses of words, which performs well on the evaluation tasks.

Sense Discovery Another related work is word sense discovery (Rapp, 2003), which used co-occurrence to evaluate semantic similarity. In our work, we not only indicate senses with the method which is based on similar knowledge (Levy and Goldberg, 2014), but also improve the associate distributional word representations.

Self-Paced Learning Self-paced learning is an adaptive weight adjust technique introduced by Kumar et al. (2010). Varieties of strategies of self-paced learning have been proved efficient on matrix factorization (Jiang et al., 2015), multimedia event detection (Jiang et al., 2014; Jiang et al., 2015), multimedia search (Jiang et al., 2014), place recognition (Shi et al., 2017) and object detection (Tang et al., 2012). As far as we know, this is the first work that adapts self-paced strategy to learn word embeddings.

3. Proposed Framework

Our proposed framework contains the following five steps:

1. Train multi-sense word embeddings on corpus, utilizing a specific basic (multi-sense word embedding) model.

2. Detect pseudo multi-sense in the embeddings. Suppose we have two sense vectors $\mathbf{v}_{w,i}$ and $\mathbf{v}_{w,j}$ of the same word w. We evaluate the neighborhood similarity by

$$P_{pse}(\mathbf{v}_{w,i}, \mathbf{v}_{w,j}) \propto \sum_{\substack{\mathbf{v}_{n_i} \in kNN(\mathbf{v}_{w,i}), \\ \mathbf{v}_{m_j} \in kNN(\mathbf{v}_{w,j})}} \cos(\mathbf{v}_g(\mathbf{v}_{n_i}), \mathbf{v}_g(\mathbf{v}_{m_j})) \quad (1)$$

   where $kNN(\mathbf{v})$ indicates the k nearest neighbors set of vector $\mathbf{v}$ in the same space, $\mathbf{v}_{w,i}, \mathbf{v}_{w,j}, \mathbf{v}_{n_i}, \mathbf{v}_{m_j}$ are all *sense vectors*, $\mathbf{v}_g(\mathbf{v}_l)$ is the corresponding *global vector* of the sense vector $\mathbf{v}_l$ (multiple *sense vectors* may have the same *global vector*). To determine whether two senses $\mathbf{v}_{w,i}$ and $\mathbf{v}_{w,j}$ of the word w are pseudo multi-sense, we choose an arbitrary threshold θ. If $P_{pse}(\mathbf{v}_{w,i}, \mathbf{v}_{w,j}) > \theta$, then $\mathbf{v}_{w,i}$ and $\mathbf{v}_{w,j}$ should be treated as one sense rather than the separated two.

   In practice, we determine θ by the following procedure. We collect the set of pairs $S = \{(w_i, w_j)\}$, of which the neighborhood of w_i has prototype-level overlap with that of w_j. For example, if cat_0 has dog_0 in its nearest neighbors, while cat_1 has dog_1, (cat_0, cat_1) should be in S. We sort $P_{pse}(\mathbf{v}_{w,i}, \mathbf{v}_{w,j})$ for each pair of $(w_i, w_j) \in S$ in descending order, and choose the value at 90% point as θ to avoid noise. This is similar to the spy technique introduced by Liu et al. (2003). We evaluate the 20 nearest neighbors for each sense.

3. Select sense for each instance in the corpus. Pseudo multi-sense would be tagged as one sense. The existence of *context vector* gives us an easy way to compute the probability of a word instance belongs to each

sense. We compute the context vector of each instance w_i by

$$\mathbf{v}_{\mathcal{C}(w_i)} = \frac{1}{|\mathcal{C}(w_i)|} \sum_{t \in \mathcal{C}(w_i)} \mathbf{v}_g(t) \quad (2)$$

where $\mathcal{C}(w_i)$ is the context set of word instance w_i which contains prototype of words, and $\mathbf{v}_g(t)$ is the global vector of word t.

Therefore, we have

$$P(Sense(w_i) = k|\mathcal{C}(w_i)) \propto \mathbf{v}_{\mathcal{C}(w_i)} \cdot \mathbf{v}_c(w_i, k) \quad (3)$$

where $\mathbf{v}_c(w_i, k)$ is the *context vector* of the k^{th} sense of word w (prototype of w_i). This would depend on different settings of different models: in CRP model (Li and Jurafsky, 2015), the right side should be activated by sigmoid function and then multiplied by $Prob(w)$, which represents the probability of word w to appear in the corpus.

There often exists some instances we are not confident to determine which sense it should belong to, e.g. $\exists j, k, (j \neq k)$, $P(Sense(w_i) = k|\mathcal{C}(w_i))$ is similar to $P(Sense(w_i) = j|\mathcal{C}(w_i))$. To solve this problem, we apply self-paced learning strategy (Kumar et al., 2010). During each iteration, we only tag the instances with high level confidence. We reformulate the problem as the following one:

$$\min_{\mathbf{a}} \mathcal{L}_{SPL}(Emb; \lambda) = \sum_{i=1}^{n} a_i (1 - \max_k P(Sense(w_i) = k|\mathcal{C}(w_i))) + f(\mathbf{a}; \lambda) \quad (4)$$

where Emb is the learned multi-sense word embedding, n is the number of word instances, $\mathbf{a} = \{0, 1\}^n$ is the weight for each instance, $f(\mathbf{a}; \lambda)$ is the self-paced learning function. Here we apply a typical binary self-paced function (Kumar et al., 2010)

$$f(\mathbf{a}; \lambda) = -\lambda ||\mathbf{a}||_1 = -\lambda \sum_{i=1}^{n} a_i \quad (5)$$

for the self-paced learning schema. At each time, we only tag those instances with $a_i = 1$. Therefore, by gradually increase λ, we can fetch more less-confident instances for our sense tagging.

4. Re-train multi-sense word embeddings on corpus, with pre-selected sense tags in step 3.

5. Repeat step 2-4 until no pseudo multi-sense is detected.

Our algorithm is summarized in Algorithm 1.

4. Evaluation

We apply Non-parametric Multi-sense Skip Gram (Neelakantan et al., 2014) as the basic model to train multi-sense word embeddings on Wikipedia Corpus (Soriano-Morales et al., 2017).

Algorithm 1 Enhanced pipeline for multi-sense word embedding

Require: training corpus T, multi-sense word embedding model M, self-paced function f and step size μ
Ensure: Multi-sense word embeddings W, tagged corpus T'

```
Initialize W by training M with T
Initialize λ
while not converged do
    Detect pseudo multi-sense with Eq (1), W
    Compute transitive closure of detected pseudo multi-sense relation
    Select confident instances in T by Eq (4) and (5)
    Add sense tags for confident instances and get updated corpus T', pseudo multi-senses will be given same sense tags
    Train M with T' to obtain updated W
    increase λ by μ
end while
return W, T'
```

Model	50d	300d
MSSG (Neelakantan et al., 2014)	49.2	57.3
NP-MSSG (Neelakantan et al., 2014)	50.9	57.5
MSSG + MT (Shi et al., 2016)	53.2	62.2
NP-MSSG + MT (Shi et al., 2016)	52.2	61.4
NP-MSSG + SPT	**58.6**	**63.7**

Table 1: Spearman rank correlation on SCWS dataset. For baselines, we evaluate the models of multi-sense skip-gram (MSSG) and non-parametric multi-sense skip-gram (NP-MSSG), as well as the combination of those with supervised pseudo multi-sense detection and matrix transformation (MT). NP-MSSG + SPT refers to our self-paced tagging model.

4.1. Word Similarity

Stanford Contextual Word Similarity (SCWS) dataset (Huang et al., 2012) is a reliable and professional dataset to estimate the performance of word embeddings, especially multi-sense word embeddings. It contains 2,003 pairs of words together with the context.

In our experiments, we follow Neelakantan et al. (2014) to define the similarity of two instances w_1 and w_2 by

$$localSim(w_1, w_2) = \cos(\mathbf{v}_{w_1, Sense(\mathcal{C}(w_1))}, \mathbf{v}_{w_2, Sense(\mathcal{C}(w_2))}) \quad (6)$$

where

$$Sense(\mathcal{C}(w_i)) = \operatorname*{argmax}_k P(Sense(w_i) = k|\mathcal{C}(w_i)) \quad (7)$$

is the sense index chosen by $\mathcal{C}(w_i)$ and Eq (3). Table 1 shows that our framework outperforms not only the basic model, but also the linear (matrix) transformation one proposed by Shi et al. (2016).

4.2. Word Analogy

To evaluate how embeddings capture intuitive pairwise word relations, Mikolov et al. (2013a) released analogy

task, which contains 19,544 quadruples in total. Each quadruple contains four words A, B, C, D, where word A is similar to word B in the same way as word C is similar to D, for instance, (Berlin, Germany, Paris, France). Among all, there are 5 classes of semantic quadruples and 9 classes of syntactic ones.

For the evaluation of multi-sense word embeddings, we follow the method proposed by Shi et al. (2016): for given multi-sense word embeddings, if there is a index quadruple (i, j, k, l) for word quadruple (A, B, C, D) s.t. $v_{A,i} - v_{B,j} + v_{D,l}$ is most similar to $v_{C,k}$, then we treat (A, B, C, D) is a correct case for the model. In addition, considering the symmetric property of the quadruples, there are totally four linear combinations for each quadruple to be evaluated, and we treat the quadruple as a positive case if one of them is satisfied.

Model	Semantic	Syntactic
MSSG (50d)	75.8	85.2
NP-MSSG (50d)	74.6	80.7
MSSG + MT (50d)	**77.5**	**88.0**
NP-MSSG + MT (50d)	75.6	82.3
NP-MSSG + SPT (50d)	76.2	86.1
NP-MSSG (300d)	83.9	89.0
NP-MSSG + MT (300d)	**85.9**	**90.2**
NP-MSSG + SPT (300d)	85.3	89.0

Table 2: Accuracy on word analogy task.

According to Table 2, we see our self-paced tagging framework ensures that the learned word embeddings keeps the semantic and syntactic relations well, although it performs not as well as the one with matrix transformation (Shi et al., 2016).

4.3. Sentence Understanding

We evaluate the quality of word embeddings with the SentEval system (Conneau et al., 2017; Kiela et al., 2017). In the evaluation, bag of words (BoW) is fed to the system as sentence features. We report the accuracies of two models on three different tasks in Table 3. The tasks are paraphrase detection (MSRP; Dolan et al. (2004)), subjectivity detection (SUBJ; Pang and Lee (2004)), and question classification (TREC; Voorhees and Buckland (2003)). On the tasks of MSRP and TREC, self-paced tagging with respect to elimination of pseudo multi-sense improves word embeddings at a non-trivial level.

4.4. Case Study

We show the k-nearest neighbors of some words in Table 4. We clearly see that the self-paced tagging model can not only eliminate the pseudo multi-senses (Norway, star, algorithm), but also keep the real multi-senses (star).

5. Conclusion

In this work, we present a novel framework for corpus sense tagging with unsupervised elimination of pseudo multi-sense, utilizing any multi-sense word embedding model as the basic model. By applying the proposed self-paced tagging strategy, we could improve the quality of multi-sense word embeddings. Experiments have shown the efficiency of our framework.

Model	MSRP	SUBJ	TREC
NP-MSSG	70.03	90.96	78.4
NP-MSSG + MT	70.55	**91.20**	83.2
NP-MSSG + SPT	**71.01**	90.97	**84.2**

Table 3: Evaluation result on transfer learning tasks.

Norway	
NP-MSSG	Denmark, Troms, Sogn, Hedmark Denmark, Sweden, Finland, Iceland Denmark, Sweden, Finland, Netherlands Denmark, Austria, Germany, Belgium
+ SPT	Denmark, Norwegian, Sweden, Trondheim
star	
NP-MSSG	stars, wars, alongside, beetlejuice stars, award, eagle, two-time supergiant, constellation, aurigae
+ SPT	stars, movie, superstar, MVP supergiant, stars, g5v, white_main
algorithm	
NP-MSSG	hash, algorithms, quick_sort, recursive algorithms, optimization, public-key
+ SPT	algorithms, computation, iteratively

Table 4: Case study of k nearest neighbors. Each row refers to a learned "sense".

6. Acknowledgements

This work is supported by the National Natural Science Foundation of China (grant No.61472017, M1752015). The authors would like to thank Xian Wu for his help on experimental environment.

7. Bibliographical References

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *JMLR*, 3(Feb).

Chen, X., Liu, Z., and Sun, M. (2014). A unified model for word sense representation and disambiguation. In *EMNLP*.

Cheng, J. and Kartsaklis, D. (2015). Syntax-aware multi-sense word embeddings for deep compositional models of meaning. In *Proc. EMNLP*.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proc. ICML*.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proc. EMNLP*.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proc. COLING*.

Flekova, L. and Gurevych, I. (2016). Supersense embeddings: A unified model for supersense interpretation, prediction, and utilization. In *Proc. ACL*.

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proc. ACL*.

Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2015). Sensembed: Learning sense embeddings for word and relational similarity. In *Proc. ACL*.

Jiang, L., Meng, D., Mitamura, T., and Hauptmann, A. G. (2014). Easy samples first: Self-paced reranking for zero-example multimedia search. In *Proceedings of the 22nd ACM international conference on Multimedia*. ACM.

Jiang, L., Meng, D., Zhao, Q., Shan, S., and Hauptmann, A. G. (2015). Self-paced curriculum learning. In *AAAI*, volume 2.

Kiela, D., Conneau, A., Jabri, A., and Nickel, M. (2017). Learning visually grounded sentence representations. *arXiv preprint arXiv:1707.06320*.

Kumar, M. P., Packer, B., and Koller, D. (2010). Self-paced learning for latent variable models. In *NIPS*.

Levy, O. and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *NIPS*.

Li, J. and Jurafsky, D. (2015). Do multi-sense embeddings improve natural language understanding? In *Proc. EMNLP*.

Liu, B., Dai, Y., Li, X., Lee, W. S., and Yu, P. S. (2003). Building text classifiers using positive and unlabeled examples. In *ICDM*.

Liu, Y., Liu, Z., Chua, T.-S., and Sun, M. (2015). Topical word embeddings. In *AAAI*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *NIPS*.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Mnih, A. and Hinton, G. E. (2009). A scalable hierarchical distributed language model. In *NIPS*.

Neelakantan, A., Shankar, J., Passos, A., and McCallum, A. (2014). Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proc. EMNLP*.

Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. ACL*.

Pelevina, M., Arefyev, N., Biemann, C., and Panchenko, A. (2017). Making sense of word embeddings. *arXiv preprint arXiv:1708.03390*.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proc. EMNLP*.

Pina, L. N. and Johansson, R. (2015). A simple and efficient method to generate word sense representations. In *Proceedings of Recent Advances in Natural Language Processing*.

Qiu, L., Tu, K., and Yu, Y. (2016). Context-dependent sense embedding. In *Proc. EMNLP*.

Rapp, R. (2003). Word sense discovery based on sense descriptor dissimilarity. In *Proceedings of the ninth machine translation summit*.

Schütze, H. (1998). Automatic word sense discrimination. *Computational linguistics*, 24(1).

Shi, H., Li, C., and Hu, J. (2016). Real multi-sense or pseudo multi-sense: An approach to improve word representation. *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity*.

Shi, H., Chen, J., and Hauptmann, A. G. (2017). Joint saliency estimation and matching using image regions for geo-localization of online video. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*.

Tang, K., Ramanathan, V., Fei-Fei, L., and Koller, D. (2012). Shifting weights: Adapting object detectors from image to video. In *Advances in Neural Information Processing Systems*.

Voorhees, E. M. and Buckland, L. (2003). Overview of the trec 2003 question answering track. In *TREC*, volume 2003.

8. Language Resource References

Soriano-Morales, Edmundo-Pavel and Ah-Pine, Julien and Loudcher, Sabine. (2017). *Syntactically Annotated Wikipedia Dump*. ISLRN 958-233-248-056-1.

Urdu Word Embeddings

Samar Haider
University of Engineering and Technology, Lahore
Pakistan
samar.haider@kics.edu.pk

Abstract

Representing words as vectors which encode their semantic properties is an important component in natural language processing. Recent advances in distributional semantics have led to the rise of neural network-based models that use unsupervised learning to represent words as dense, distributed vectors, called 'word embeddings'. These embeddings have led to breakthroughs in performance in multiple natural language processing applications, and also hold the key to improving natural language processing for low-resource languages by helping machine learning algorithms learn patterns more easily from these richer representations of words, thereby allowing better generalization from less data. In this paper, we train the skip-gram model on more than 140 million Urdu words to create the first large-scale word embeddings for the Urdu language. We analyze the quality of the learned embeddings by looking at the closest neighbours to different words in the vector space and find that they capture a high degree of syntactic and semantic similarity between words. We evaluate this quantitatively by experimenting with different vector dimensionalities and context window sizes and measuring their performance on Urdu translations of standard word similarity tasks. The embeddings are made freely available in order to advance research on Urdu language processing.

Keywords: Urdu, low-resource languages, vector representations, distributional semantics, word embeddings

1. Introduction

Urdu is an Indo-Aryan language that is the national language and lingua franca of Pakistan, and an official language of multiple states of India. There are 109 million speakers of Urdu in Pakistan and 51 million speakers in India. Urdu is also widely spoken across the rest of the world, with more than 163 million total speakers in all countries [1]. Despite its widespread use, both in South Asia and among people of South Asian descent across the world, Urdu remains a low-resource language with few corpora and datasets of appreciable size available for computational tasks. This is not an uncommon phenomenon among low-resource languages, since creation of such resources requires significant time and manpower. However, without sufficient labelled data, it is very difficult to build natural language processing systems that can learn useful patterns which generalize well. This perennial lack of data leads to little research performed on these languages, which in turn leads to few resources created by research.

One way to break out of this loop is to learn higher-level, complex representations of words and phrases that can then be used as input to bootstrap other natural language processing systems downstream. The area of distributional semantics focuses on creating just such representations, wherein semantically similar words are assigned similar representations. Recently, there has been much focus on using neural networks to learn vector representations of words by modelling the task as one of predicting surrounding words from a target word. One advantage of such techniques is that they use unsupervised learning and don't require annotated corpora, which are rare. They can instead be trained on larger, more readily available unannotated corpora, and the learned representations can then be used in natural language processing tasks which use smaller amounts of labelled data.

In this paper, we create the first large-scale distributed vector representations of Urdu words using the skip-gram model introduced by Mikolov et al. (2013a). We collect multiple corpora totalling over 140 million tokens, and train the model on them to learn a vocabulary of more than 100,000 Urdu words. We then take a closer look at the learned representations by looking at relationships between semantically similar words in the vector space. We also evaluate the representations by comparing their performance on word similarity tasks like WordSim-353 and SimLex-999 that test vector space models' ability to learn semantic relations between word pairs. The embeddings are made publicly available online[2] for academic use.

This paper is organized as follows: in Section 2, we present a historical background of distributional semantics, discuss recent advances in the field, and introduce some of its many applications in natural language processing. In Section 3, we describe our own experimental setup, including the corpora used, models trained to learn the word embeddings, and the evaluation tasks and metrics employed. In Section 4, we present our results, both qualitative and quantitative, and discuss the performance of our learned word embeddings on adaptations of standard word similarity tasks. In Section 5, we conclude by summarizing our work and describing our contributions to the improvement of natural language processing for Urdu. We also discuss ideas for future research that builds upon the work done and resources created in this paper.

2. Background

2.1. Distributional Semantics

The crux of the area of distributional semantics is best captured in the words of Firth (1957): "A word is known by the company it keeps." Work on distributional semantics

[1] https://ethnologue.com/language/urd

[2] https://github.com/samarh/urduvec

has thus largely revolved around representing the meaning of a word by the distribution of other words around it.

The 1990s saw the creation of multiple techniques for modelling words in this manner. Church and Hanks (1990) proposed the Pointwise Mutual Information (PMI) measure which sought to quantify the degree of relatedness between two words by looking at how often they occurred together, compared with how often they might occur together if they were not related (independent). Deerwester et al. (1990) invented a model called latent semantic analysis (LSA), which used singular value decomposition (SVD) to create word representations from term-document matrices in an information retrieval setting.

The term 'word embeddings' was first coined by Bengio et al. (2003), who proposed the first neural probabilistic language model. By using a feed-forward neural network with a single hidden layer to predict the next word in a sequence, they laid the foundation of the architecture upon which modern approaches are based. The architecture contained three major building blocks: an embedding layer to generate word embeddings; an intermediate layer to generate an intermediate representation of the words; and a softmax layer to generate a probability distribution over the vocabulary. However, they found that the final softmax layer became the primary bottleneck when training the system due to the cost of computing the function over a large vocabulary.

Collobert and Weston (2008) developed on this work by making a few improvements to the model, the most important of which was the replacement of the expensive cross-entropy criterion with a more efficient pairwise ranking criterion, which greatly improved training speed. They trained word embeddings on a large corpus and showed that the learned embeddings were able to capture the meaning of words quite well, proving useful in higher-level natural language processing tasks (Collobert et al., 2011).

2.2. Skip-gram model

The skip-gram model is one of two neural network architectures introduced by Mikolov et al. (2013a) and later improved upon in (Mikolov et al., 2013b). Together with its sister model, called continuous bag-of-words (CBOW), the two are commonly referred to as 'word2vec'[3], a set of computationally efficient methods for learning vector representations of words.

While the language modelling neural network of Bengio et al. (2003) relied only on past words, the skip-gram model instead included a window of words both before and after the target word when making predictions. This increase in context allowed better predictions and was one of the reasons for improved performance over the language modelling approach. Another improvement was the removal of the computationally expensive hidden layer, in the absence of which the model trained much faster over large corpora.

The two models (skip-gram and CBOW) also differ among themselves in their input and output: while CBOW uses context words to predict the target word, skip-gram does the reverse and uses the target word to predict the context words. A comparison between the performance of different embedding methods by Levy et al. (2015) showed that skip-gram outperforms not only CBOW in the vast majority of cases but also the more recent GloVe method proposed by Pennington et al. (2014).

The training objective of the skip-gram model is to find word embeddings that prove useful for predicting surrounding words, and is defined as:

$$J_\theta = \frac{1}{T}\sum_{t=1}^{T} \sum_{-n\leq j\leq n, \neq 0} \log p(w_{t+j} \mid w_t) \qquad (1)$$

where $p(w_{t+j} \mid w_t)$ is the log probability of a surrounding word given the target word, n is the size of the context window on either side, and T is the size of the corpus.

The softmax output of the skip-gram model is defined as:

$$p(w_{t+j} \mid w_t) = \frac{\exp(v_{w_t}^\top v'_{w_{t+j}})}{\sum_{w_i \in V} \exp(v_{w_t}^\top v'_{w_i})} \qquad (2)$$

where $v_{w_t}^\top v'_{w_{t+j}}$ is the log probability of the surrounding word, which is normalized by sum of the log probabilities of all the words in the vocabulary, V.

Mikolov et al. (2013c) showed that the skip-gram model not only set the state-of-the-art at word similarity tasks, but that the learned embeddings were found to be surprisingly good at capturing both syntactic and semantic relationships and regularities in language.

2.3. Applications

The introduction of the skip-gram model has led to widespread adoption of word embeddings by the natural language processing community. They are now used in a diverse range of applications, some of which we briefly discuss here.

Kim (2014) trained a convolutional neural network on top of pre-trained word embeddings for sentence classification and was able to improve upon previous results with very little parameter tuning. Zou et al. (2013) showed substantial gains in BLEU points at machine translation tasks by using bilingual word embeddings trained from large unlabelled corpora with word alignment constraints to compute semantic similarity of word pairs. Chen and Manning (2014) created a neural network-based dependency parser that used word embeddings as features and found that it showed an improvement in both accuracy and efficiency. dos Santos and Gatti (2014) proposed a method to perform sentiment analysis on short texts using convolutional neural networks with word embeddings as features. Applications of word embeddings have also crossed into other domains: Frome et al. (2013) used convolutional neural networks to predict word embeddings of image labels instead of the labels themselves to exploit semantic information for predicting unseen labels. Vinyals et al. (2015) set the state-of-the-art on multiple image captioning tasks by using convolutional neural networks to embed both images and text in the same vector space for generating image captions.

[3] https://code.google.com/archive/p/word2vec/

3. Experimental Setup

3.1. Corpora

We use three different Urdu corpora to train our model: a corpus with 90 million tokens (Jawaid et al., 2014); a corpus with 35 million tokens (Adeeba et al., 2014); and a dump of the entire Urdu Wikipedia[4]. Extensive preprocessing is performed on all three sources to clean the input. Since our focus is on learning representations of Urdu words only, we remove all non-Arabic script characters from the input. We also remove diacritics in order to normalize words and remove redundant forms. For extracting plain text from the Wikipedia dump, we use Matt Mahoney's script[5] with a few alterations to accommodate Urdu. The post-cleaning statistics of all three corpora, along with their totals, are shown in Table 1.

Corpus	Words	Sentences
(Jawaid et al., 2014)	87,552,394	3,475,529
(Adeeba et al., 2014)	35,347,850	1,429,054
Urdu Wikipedia	17,755,219	527,999
Total	140,655,463	5,432,582

Table 1: Statistics of the corpora used to train the model.

3.2. Model Parameters

We train the skip-gram model implemented in the Gensim toolkit[6] and experiment with varying context window sizes (3, 5, 7) and embedding dimensionalities (100, 200, 300). We pick a minimum frequency cut-off of 10 for a word to be included in the vocabulary and an initial learning rate of 0.025. We use negative sampling to train the model for 5 epochs over the entire text, with the number of noise words to be sampled set to 5.

3.3. Evaluation

For evaluation of the learned word embeddings, we use two benchmark tasks that gauge relationships between different English words: WordSim-353 (Finkelstein et al., 2001) and SimLex-999 (Hill et al., 2015).

WordSim-353 contains 353 word pairs with relatedness scores assigned by 13 to 16 human subjects, and their average used as the final score. SimLex-999 is a more difficult dataset that contains 999 concrete and abstract adjective, noun, and verb pairs. It seeks to measure similarity (cup, mug) rather than relatedness (cup, coffee), and contains similarity scores along with ratings for words' conceptual concreteness assigned to each pair by human subjects. To adapt the two tasks for our work, we translate their word pairs into Urdu using Google's translation service[7].

For comparing our model's predictions with the scores assigned by human subjects, we use the Spearman correlation coefficient (Spearman, 1904), a well-established nonparametric measure of rank correlation between two variables. The rank correlation coefficient, r_s, is defined as:

$$r_s = 1 - \frac{6 \sum_i^n d_i^2}{n(n^2 - 1)} \quad (3)$$

where n is the number of observations and d_i is the difference in rank between the i^{th} observations. Perfect Spearman correlations of $+1$ and -1 occur when the observations of two variables are monotonically increasing or decreasing functions of each other, respectively.

4. Results and Discussion

Training our model on the corpora with the given parameters resulted in a vocabulary of over 100,000 words. This is due to the rich morphology that Urdu exhibits. We then evaluated the accuracy of our word embeddings on the Urdu translations of WordSim-353 and SimLex-999. Due to differences between English and Urdu, a number of English words were either translated into Urdu phrases or left untranslated altogether by the translation service. We thus pruned the datasets to ignore untranslated or out-of-vocabulary (OOV) word pairs that were not found among our learned embeddings. This left us with 269 valid word pairs of WordSim-353 and 691 valid word pairs of SimLex-999.

Table 2 shows the Spearman correlation results of different models on the translation of WordSim-353. We can see that 200- and 300-dimensional embeddings outperform 100-dimensional ones in all cases. The best performing are 200-dimensional embeddings trained with a 5-word context window, achieving a Spearman correlation of 0.524.

		Dimensionality		
		100	200	300
Context	3	0.489	0.516	0.518
	5	0.492	**0.524**	0.513
	7	0.491	0.500	0.510

Table 2: Results of experiments on WordSim-353.

Table 3 shows the Spearman correlation results of different models on the translation of SimLex-999. Here, too, we see a trend of higher dimensional embeddings generally performing better than lower dimensional ones. The best performing embeddings here are 300-dimensional ones trained with a 7-word context window, achieving a Spearman correlation of 0.306.

		Dimensionality		
		100	200	300
Context	3	0.277	0.295	0.294
	5	0.293	0.301	0.301
	7	0.293	0.299	**0.306**

Table 3: Results of experiments on SimLex-999.

Our best performing models achieve Spearman correlations of 0.524 and 0.306 on translations of WordSim-353 and SimLex-999, respectively. For comparison, 300-dimensional embeddings trained by Mikolov et al. (2013a)

[4]https://dumps.wikimedia.org/urwiki/latest/

[5]http://mattmahoney.net/dc/textdata

[6]https://radimrehurek.com/gensim/models/word2vec.html

[7]https://translate.google.com/

using the skip-gram model on 1 billion words of English Wikipedia achieved Spearman correlations of 0.655 and 0.414 on WordSim-353 and Simlex-999, respectively (Hill et al., 2015). This shows that, despite challenges in translating English words accurately into Urdu, our embeddings have captured semantic relationships between words quite well.

To take a closer look at the kind of semantic relationships captured in the word embeddings, we find the ten closest vectors to a given word in the vector space using the cosine similarity. Figure 1 shows three examples of this, along with English translations, for the Urdu words for 'Lahore' (city), 'room', and 'car', respectively. Inspecting the results, it is clear to see that the embeddings have captured very meaningful syntactic and semantic relationships between words. They have also captured semantic similarity between different spelling variations and morphological forms found in Urdu.

(Lahore)	**لاہور**	**(room)***	**کمرہ**	**(car)**	**گاڑی**
(Karachi)	کراچی	(rooms)†	کمرے	(motor)	موٹر
(Rawalpindi)	راولپنڈی	(room)*	کمرا	(jeep)	جیپ
(Gujranwala)	گوجرانوالہ	(flat)	فلیٹ	(cycle)	سائیکل
(Multan)	ملتان	(veranda)	برآمدہ	(motorcycle)	موٹر سائیکل
(Peshawar)	پشاور	(bedrooms)	بیڈروم	(wagon)	ویگن
(Sialkot)	سیالکوٹ	(rooms)†	کمروں	(bicycle)	بائیسکل
(Amritsar)	امرتسر	(verandas)	برآمدے	(train)	ٹرین
(Sheikhupura)	شیخوپورہ	(hall)	ہال	(rickshaw)	رکشہ
(Sargodha)	سرگودھا	(cabin)	کیبن	(bike)	بائیک
(Hyderabad)	حیدرآباد	(hostel)	ہاسٹل	(cars)	گاڑیوں

Figure 1: Words most similar (in descending order of similarity) to those in bold. An * represents spelling variations and a † represents different morphological forms of the same word.

5. Conclusion

The introduction of computationally efficient neural network-based methods for unsupervised learning of word embeddings from large unannotated corpora has been a watershed moment in natural language processing in recent years. The use of embeddings has not only improved the state-of-the-art in multiple natural language processing applications, but has also provided an impetus to research on low-resource languages. In this paper we presented work done on creating the first large-scale word embeddings for Urdu, a low-resource albeit widely-spoken South Asian language that has a large population of native speakers in Pakistan and India. We performed quantitative evaluation of the embeddings by adapting standard word similarity tasks to Urdu, and investigated relationships between the learned embeddings by looking at words predicted by the model to be semantically similar.

In the future, we plan on refining these embeddings further as well as creating custom benchmark tasks for evaluating them, keeping in mind the distinct characteristics of the Urdu language. We also plan on performing extrinsic evaluation of these embeddings by using them in tasks like text classification, named entity recognition, sentiment analysis, dependency parsing, and machine translation. We are optimistic of this direction and strongly believe that this resource will play a major role in improving natural language processing for Urdu.

6. Bibliographical References

Adeeba, F., Akram, Q., Khalid, H., and Hussain, S. (2014). CLE urdu books n-grams. In *Conference on Language and Technology, CLT 14, Karachi, Pakistan*.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar*, pages 740–750.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: deep neural networks with multitask learning. In *Proceedings of the Twenty-Fifth International Conference on Machine Learning, (ICML 2008), Helsinki, Finland*, pages 160–167.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

dos Santos, C. N. and Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of the 25th International Conference on Computational Linguistics, COLING 2014, Dublin, Ireland*, pages 69–78.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: the concept revisited. In *Proceedings of the Tenth International World Wide Web Conference, WWW 10, Hong Kong, China*, pages 406–414.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in Linguistic Analysis*.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T. (2013). Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013, Lake Tahoe, Nevada, USA*, pages 2121–2129.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Confer-*

ence on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar, pages 1746–1751.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *TACL*, 3:211–225.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013, Lake Tahoe, Nevada, USA*, pages 3111–3119.

Mikolov, T., Yih, W., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar*, pages 1532–1543.

Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, Massachusetts, USA*, pages 3156–3164.

Zou, W. Y., Socher, R., Cer, D. M., and Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, Seattle, Washington, USA*, pages 1393–1398.

7. Language Resource References

Jawaid, Bushra and Kamran, Amir and Bojar, Ondřej. (2014). *Urdu Monolingual Corpus*. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Social Image Tags as a Source of Word Embeddings: A Task-oriented Evaluation

Mika Hasegawa, Tetsunori Kobayashi, Yoshihiko Hayashi
School of Science and Engineering, Waseda University
Waseda-machi 27, Shinjuku, Tokyo 1690042, Japan
mika@pcl.cs.waseda.ac.jp, koba@waseda.jp, yshk.hayashi@aoni.waseda.jp

Abstract

Distributional hypothesis has been playing a central role in statistical NLP. Recently, however, its limitation in incorporating perceptual and empirical knowledge is noted, eliciting a field of perceptually grounded computational semantics. Typical sources of features in such a research are image datasets, where images are accompanied by linguistic tags and/or descriptions. Mainstream approaches employ machine learning techniques to integrate/combine visual features with linguistic features. In contrast to or supplementing these approaches, this study assesses the effectiveness of social image tags in generating word embeddings, and argues that these generated representations exhibit somewhat different and favorable behaviors from corpus-originated representations. More specifically, we generated word embeddings by using image tags obtained from a large social image dataset YFCC100M, which collects Flickr images and the associated tags. We evaluated the efficacy of generated word embeddings with standard semantic similarity/relatedness tasks, which showed that comparable performances with corpus-originated word embeddings were attained. These results further suggest that the generated embeddings could be effective in discriminating synonyms and antonyms, which has been an issue in distributional hypothesis-based approaches. In summary, social image tags can be utilized as yet another source of visually enforced features, provided the amount of available tags is large enough.

Keywords: word embeddings, image tags, social media, semantic similarity, synonyms, antonyms.

1. Introduction

Virtually, all the methods for generating distributional/distributed word representations (Baroni et al., 2014) rely on the notion of distributional hypothesis (Firth, 1957). These approaches enable word representations to properly capture the distributional hypothesis by measuring the commonality of the linguistic contexts of word occurrences. Although these approaches are proven effective in various semantic tasks, they are limited in terms of the incorporation of perceptual and empirical knowledge: perceptually or empirically obvious objects have not been necessarily well verbalized in a corpus of written texts (Bruni et al., 2014). Yet another issue with the distributional hypothesis-based methods is that they often run into trouble when discriminating synonyms from antonyms or more vaguely related words (Hill et al., 2015).

Recently, motivated by these issues, several research works that try to incorporate human perceptual/empirical knowledge into linguistically derived representations have emerged. Most typically, such approaches combine/integrate visual features achieved from visual resources with linguistic features (word embeddings) by applying machine learning/deep learning techniques. To enable this line of research, a visual resource in which an image is accompanied by linguistic descriptions is generally required.

Although these methods compensated/improved purely linguistic representations, the source of visual features cannot be limited to image data. That is, if a content of any modality is described with a substantial amount of linguistic tags and/or descriptions, the linguistic co-occurrence observed around the content can be utilized as a source of semantic features. Once resting on this notion, the so-called social media can be exploited as an attractive resource.

When using a social media service, the user assigns tags to her/his contents so that they may be easily searched and located by other users. Sometimes, this process is referred to as folksonomy, as the tags are not constrained by pre-defined controlled keywords and/or ontology terms. Despite the nature that users can freely choose tags, it is exemplified that the vocabulary of tags in a social media service has converged and become stabilized over time (Halpin et al., 2007). Moreover, if a target social media is popular enough and maintains a huge amount of content, the set of tags can be considered as a type of corpus where a similar set of tags would be assigned to similar content. These facts validate the use of social tags as a source of semantic features. Furthermore, the media type of social media content is not necessarily limited to images, admitting the possibility of incorporating other types of modality.

In the present work, we utilize the YFCC100M (Thomee et al., 2016) dataset[1], which is a social media-originated dataset. We generate word embeddings by statistically processing the co-occurrences of linguistic tags. The empirical results of semantic similarity/relatedness tasks may allow us to conclude that social image tags can be utilized as yet another source of visually enforced features, provided the amount of available image tags is large enough.

2. Related work

The present research is inspired by the work on multimodal semantic representations (section 2.1). As most of the work in this direction deals with image features, image datasets (section 2.2) as a source of visual features are of crucial importance.

[1] http://yfcc100m.appspot.com/

Dataset	# of images	Annotation type	Who annotated?
ImageNet	14M	WordNet synsets	Crowdworkers
ESP-Game	350K	tags	ESP-Game participants
MS COCO	120K	categories, captions	Crowdworkers
YFCC100M	100M	tags	Flickr contributors

Table 1: Representative image datasets.

2.1. Multimodal semantic representation

Theoretically supported by the concept of grounded cognition (Barsalou, 2008) and technically endorsed by the progress of machine learning techniques, work on distributed word representation (word embeddings) has extended its research scope to *multimodal semantic representation* in which perceptual information, such as visual features, is combined with or integrated into corpus-derived linguistic embeddings (Silberer and Lapata, 2014; Bruni et al., 2014; Kiela and Bottou, 2014; Kiela et al., 2016). Mainstream approaches employ deep learning techniques to integrate/combine visual features with linguistic features (Lazaridou et al., 2015; Kodirov et al., 2017; Hasegawa et al., 2017). The achieved results in standard semantic similarity/relatedness tasks are generally promising, suggesting that corpus-derived word embeddings can be successfully enhanced by visual features.

2.2. Source of image/visual features

As far as a method for inducing multimodal semantic representation relies on image features, the role of the source image dataset is crucial. Image datasets can be classified in terms of the different types of collected images, linguistic annotations, and originators (who tagged images). Table 1 contrasts representative image datasets with YFCC100M, which is the central ingredient of the present work.

ImageNet (Krizhevsky et al., 2012) has been playing a leading role in improving visual object recognition techniques. The ESP-Game (von Ahn and Dabbish, 2004) dataset is often employed in the work on multimodal semantic representations. These two datasets are contrastive in a sense: ImageNet images clearly portray a focused object, whereas ESP-Game images often depict more natural scenes, showing multiple objects and the relations among them. This means that the ESP-Game images are noisier in terms of visual object recognition (Kiela et al., 2016). MS COCO (Lin et al., 2014), however, has been heavily employed in caption generation research.

YFCC100M (Thomee et al., 2016) collects images and the associated metadata from a social media service Flickr, which is a Web-based service for sharing visual contents. This dataset is different from others in that the tags attached to a posted image is given by the contributor. This nature makes a difference when it is utilized as a source of semantic features, as discussed in the rest of this paper. In Flickr, each image is annotated with a variety of tags, including the name of a depicted object, the place where the picture is taken, and the emotional feeling expressed by the contributor. The amount of data made possible by the popularity of Flickr is also a crucial factor; Thus meaningful co-occurrence statistics can be collectively obtained from this huge dataset.

3. Generating word embeddings from social media data

As described earlier, we aim to construct word semantic representations (word embeddings) by exploiting a social media service as a source of visually enforced semantic features. More specifically, we generate word embeddings, first by constructing a tag co-occurrence matrix, and then converting the raw counts to more effective quantities, and finally applying a dimensionality reduction technique to the co-occurrence matrix. It should be emphasized here that we only employ textual tags, meaning that we have never applied any visual feature extraction to the maintained images. This process assigns each tag word a dense and low-dimensional vector, which can be utilized as a word embedding vector.

The rationale behind this approach is that visual co-occurrences of objects could be naturally captured by the co-occurrence of image tags. We further suppose that the intention of a contributor who wants to disseminate her/his photo to a broader audience may be reflected in the attached tags. Therefore, the tags attached to an image can be considered as a proxy to the image that may partake social implications.

Constructing a tag word co-occurrence matrix: We constitute a tag word co-occurrence matrix M, where $M_{i,j}$ counts the number of times that tag word w_i and tag word w_j are attached to the same image. The shape of the matrix M is $N \times N$ if the number of word types equals to N.

Transforming the matrix: As the raw counts do not properly dictate the strength of co-occurrence, we transform the co-occurrence matrix by computing positive pairwise mutual information (PPMI) (Church and Hanks, 1990), which is formulated as follows.

$$M_{i,j} = \max\left(0, \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)}\right) \quad (1)$$

As this formulation suggests, PPMI alleviates the influence of high frequency tags, allowing us to properly measure the strength of tag co-occurrence.

Dimensionality reduction: As the number of tag types easily increases with the size of a dataset, the matrix M would generally be sparse. We thus apply singular value decomposition (SVD), and reduce the matrix from $N \times N$ to $d \times d$, where $d \ll N$.

$$M = U \cdot \Sigma \cdot V^T \quad (2)$$

$$\approx U^{(d)} \cdot \Sigma^{(d)} \cdot V^{T^{(d)}} \quad (3)$$

This gives us the word embedding vector for word w_t as $v_{w_t} = U_{w_t}^{(d)} \cdot \sqrt{\Sigma^{(d)}}$. As frequently argued, dense and low-dimensional representations may yield the benefit of data reduction as well as the effect of data abstraction.

4. Experimental settings

We evaluate the efficacy of tag-originated word embeddings (henceforth, tag embeddings) in standard semantic similarity/relatedness tasks. This section describes the experimental settings, and the following section discusses the results.

4.1. Source of social image tags

We used the Yahoo Flickr Creative Commons 100M (YFCC100M) dataset (Thomee et al., 2016). This dataset collects almost 100 M images and approximately 700 K movies posted on Flickr. Each image is described by a set of metadata, including image titles, user-generated tags, and machine-generated tags. We constructed word semantic representations by feeding only the user-generated tags (tags that are given by the contributors of contents) to the process described in the previous section. The number of user-generated tags amounts to approximately 69M, among which 68.5M tags are assigned to images, and the rest 420K are assigned to movies. In average, approximately seven tags are assigned to each instance of the content.

Figures 1 (a) and (b) show examples of the images and tags. Figure 1 (a) portrays a `cat`, which is further detailed by the hyponyms "kitten" and "kitty," as well as the hypernym "pet." Figure 1 (b) artistically shows a scenery for which abstract words like "calm" and "quiet" are attached. Moreover tag words like "summer" (time) or "favignana" (place) are being assigned, which would not be annotated even by state-of-the-art computer vision techniques.

In the experiments, a word co-occurrence matrix was constructed for the selected 20,943 words that were used for describing more than 1,500 images. We excluded multi-word tags and numbers, and the remaining words were converted to lowercase. The total number of tag instances counts at a value of 10 M. In the dimensionality reduction by SVM, the dimensionality d of word embeddings is set to 300.



Figure 1: Examples of YFCC100M images and tags.

4.2. Evaluation tasks and the datasets

We evaluated the efficacy of constructed word embeddings with word similarity/relatedness tasks in which the predicted scores were compared against the gold data given in the following test datasets. The Spearman's rank correlation coefficient was employed as the performance measure of the experiment that uses one of the datasets.

- **YP130** (Yang and Powers, 2006): This dataset that maintains 130 verb pairs was built for the evaluation of verb similarities.
- **WordSim353** (Finkelstein et al., 2002): This dataset contains 353 word pairs for which semantic relatedness scores are assigned. Note that semantic similarity that essentially measures the degree of synonymy can be considered as a subclass of semantic relatedness.
- **SimLex999** (Hill et al., 2015): SimLex999 provides word similarity (rather than relatedness or association) judgments for 999 word pairs. Note that the parts of speech of compared words are always the same.
- **USF Assoc** (Nelson et al., 2004): This dataset, University of South Florida Free Association Norms (abbreviated as USF Assoc), collects the free association scores for 5,019 stimulus words. In the experiments, we used the pairs of words included in the SimLex999 dataset. Needless to say, free association relations include a wider range of semantic relationships.
- **MEN** (Bruni et al., 2014): This dataset presents semantic relatedness scores for 3,000 word pairs. This dataset was specially made to evaluate multimodal representations. The parts of speech of compared words are not necessarily the same. The words are biased to concrete concepts, as they are chosen from the tags in the ESP-Game and Flickr data.
- **SemSim / VisSim** (Silberer et al., 2016): This is a dataset of 7,576 word pairs, each of which is annotated using not only semantic similarities (SemSim) but also visual similarities (VisSim); therefore, the user can compare the performances of her/his model in predicting different types of similarities.

5. Experimental results

5.1. Major results

Table 2 compares the major experimental results (in Spearman's correlations), where the YFCC column shows the results with the tag embeddings that were generated from the tag co-occurrence matrix which records 10 M tag instances. Wiki or GNews displays the results with corpus-derived word embeddings. By applying the Word2Vec Skip-Gram model, we derived 300-dimensional word embeddings both for Wikipedia 2009 dump[2] (Wiki) and GoogleNews[3] (GNews). Notice that the dimensionalities are equalized with those of tag embeddings.

As shown in the table, the tag embeddings achieved the highest correlation of 0.81 in the MEN relatedness task, demonstrating that social image tags are good sources of visually enforced features for concrete concepts. Furthermore, the tag embeddings achieved an acceptable result

[2] `http://mattmahoney.net/dc/textdata`

[3] `https://github.com/mmihaltz/word2vec-GoogleNews-vectors`

of 0.45 in the SimLex999 similarity task, which is worse than GNews-originated embeddings, but better than Wiki-originated embeddings. This could be good news, as a similarity task is generally considered to be more difficult when compared to a relatedness task.

The table presents that the degradations in USF Assoc score compared to that of SimLex999 are evident in all the embedding types. However, the difference in YFCC (tag embedding) is larger than the other two types. This may be due to the fact that the tags attached to an image are tightly associated with the image, whereas linguistic contexts, or context windows, are more generous to include weakly associated words.

A surprise result is a correlation of 0.47 achieved by the tag embeddings in the YP130 verb similarity task. It could be unfortunately unreliable, as the coverage is as low as 16% (shown in the second column). This insists that verbs are not frequently assigned as a social image tag.

In summary, the tag embeddings could achieve comparable performances with corpus-originated embeddings in a variety of similarity/relatedness tasks.

Dataset	# of pairs	YFCC	Wiki	GNews
YP130	130 (16%)	0.47	0.35	0.24
WS353	353 (66%)	0.65	0.74	0.70
SimLex999	999 (54%)	**0.45**	0.39	0.49
USF Assoc	999 (54%)	0.34	0.38	0.44
MEN	3000 (96%)	**0.81**	0.74	0.77
SemSim	7576 (62%)	0.62	0.63	0.72
VisSim	7576 (62%)	0.49	0.50	0.55

Table 2: Results in semantic similarity/relatedness tasks (in Spearman's correlation).

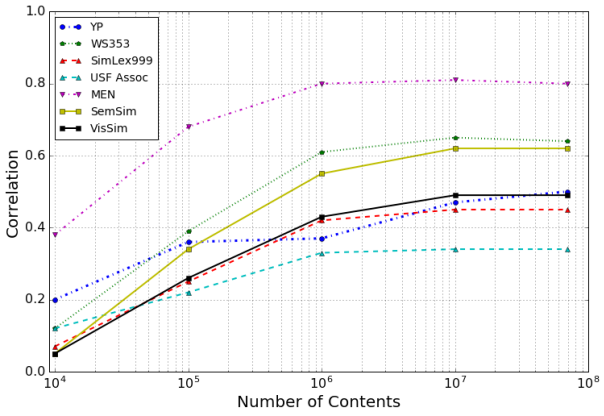


Figure 2: Relationship between the number of contents and the accuracy of semantic relatedness estimation.

The amount of data versus performances: It is often desired to know the necessary/sufficient amount of data to achieve a reasonable performance. Figure 2 displays the saturation of correlation coefficients with the increase in the amount data. As the graph shows, the performance of all datasets does not significantly improve when increased to more than 10 M contents[4], showing a limit to the effective number of tags.

5.2. Do social image tags make a difference?

An expectation to multimodal semantic representations is to address issues inherent to the purely linguistic distributional hypothesis. This expectation also applies to the tag embeddings proposed in the present work. To assess whether this could be attained, we conducted a small experiment by using WordNet semantic relationships. More specifically, for each of the selected 1,928 words that have tag embeddings, we retrieved k-nearest words in WordNet, and investigated the ranks of their antonyms, synonyms, hypernyms, and hyponyms.

Table 3 compares the mean reciprocal ranks (MRRs) of the words in each semantic relation with each embedding type. The average numbers of corresponding words in each semantic relation are as follows: 1.65 for an antonym, 3.27 for a synonym, 5.34 for a hyponym, and 3.07 for a hypernym.

Relation	# of pairs	YFCC	Wiki	GNews
antonym	798	**0.05**	0.18	0.13
synonym	3593	**0.15**	0.09	**0.16**
hyponym	1900	**0.11**	0.04	0.07
hypernym	4163	**0.06**	0.02	0.04

Table 3: MRR results for WordNet semantic relations.

The most prominent fact presented in the table is that the MRR for antonyms with YFCC embedding is far lower than that of the other two embedding types. This confirms that the proposed method could be effective in excluding antonyms from the other semantically similar/related words. Note that YFCC embedding ranked synonyms, hypernyms, and hyponyms are relatively higher than other two embedding types. This may endorse the fact that a content contributor tends to add hypernyms and/or hyponyms as tags, probably for the purpose of increasing the probability of the posted image being retrieved.

To sum up, the resulting semantic representations exhibit somewhat different and favorable behaviors from corpus-originated representations.

6. Concluding remarks

This paper proposed to exploit social image tags as a source of features for generating word embeddings, and demonstrated that the generated representations exhibit somewhat different and favorable behaviors compared to the corpus-originated representations. These results highlight that social media could be exploited as yet another source of semantic features.

This insight may open up a new way of meaning representation that optimally integrates verbal, perceptual, and social features upon a given semantic task. Other benefits potentially attained from the use of social media are dynamics and multilinguality. Social tagging would provide opportunities to capture new definitions for existing words or new

[4]Each content is associated with a set of tags.

words themselves. Tags given in multiple languages can be exploited to develop cross-lingual/multilingual semantic representations.

7. Acknowledgments

The present research was supported by JSPS KAKENHI Grant Number 17H01831 and 15K12873.

8. Bibliographical references

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. *ACL*, pages 238–247.

Barsalou, L. W. (2008). Grounded cognition. *Annu. Rev. Psychol.*, 59(August):617–645.

Bruni, E., Gatica-perez, D., Tran, N. K., and Baroni, M. (2014). Multimodal distributional semantics. *J. Artif. Intell. Res.*, 49(December):1–47.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, March.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: the concept revisited. *ACM Trans. Inf. Syst.*, 20(1):116–131.

Firth, J. (1957). A synopsis of linguistic theory 1930-1955. *Studies in linguistic analysis*, pages 1–32.

Halpin, H., Robu, V., and Shepherd, H. (2007). The complex dynamics of collaborative tagging. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 211–220, New York, NY, USA. ACM.

Hasegawa, M., Kobayashi, T., and Hayashi, Y. (2017). Incorporating visual features into word embeddings: A bimodal autoencoder-based approach. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS2017)*, Montpellier, France.

Hill, F., Reichart, R., and Korhonen, A. (2015). SimLex-999: Evaluating Semantic Models with (Genuine) Similarity Estimation. *Comput. Linguist.*, 41(4):665–695.

Kiela, D. and Bottou, L. (2014). Learning Image Embeddings using Convolutional Neural Networks for Improved Multi-Modal Semantics. *EMNLP 2014*, pages 36–45.

Kiela, D., Clark, S., Ver, A. L., Clark, S., Baroni, M., Murphy, B., Barbu, E., and Poesio, M. (2016). Comparing Data Sources and Architectures for Deep Visual Representation Learning in Semantics. *EMNLP 2016*, 34(2):447–456.

Kodirov, E., Xiang, T., Gong, S., and Mary, Q. (2017). Semantic Autoencoder for Zero-Shot Learning. *IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3174–3183.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.*, pages 1–9.

Lazaridou, A., Nghia, T. P., and Baroni, M. (2015). Combining Language and Vision with a Multimodal Skip-gram Model. *NAACL 2015*, pages 153–163.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Nelson, D. L., McEvoy, C. L., and Schreiber, T. a. (2004). The University of South Florida free association, rhyme, and word fragment norms. *Behav. Res. Methods, Instruments, Comput.*, 36(3):402–407.

Silberer, C. and Lapata, M. (2014). Learning Grounded Meaning Representations with Autoencoders. *Proc. 52nd Annu. Meet. Assoc. Comput. Linguist. (Volume 1 Long Pap.*, pages 721–732.

Silberer, C., Ferrari, V., and Lapata, M. (2016). Visually Grounded Meaning Representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, X(X):1–1.

Thomee, B., Shamma, D. A., Friedland, G., Poland, D., and Borth, D. (2016). YFCC100M : The New Data in Multimedia Research. *Commun. ACM*.

von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. *Proc. 2004 Conf. Hum. factors Comput. Syst. - CHI '04*, pages 319–326.

Yang, D. and Powers, D. M. (2006). Verb Similarity on the Taxonomy of WordNet. *Proc. GWC-06*, pages 121—-128.

Towards AMR-BR: A SemBank for Brazilian Portuguese Language

Rafael Torres Anchiêta, Thiago Alexandre Salgueiro Pardo
Interinstitutional Center for Computational Linguistics (NILC)
Institute of Mathematical and Computer Sciences, University of São Paulo
rta@usp.br, taspardo@icmc.usp.br

Abstract

We present in this paper an effort to build an AMR (Abstract Meaning Representation) annotated corpus (a semantic bank) for Brazilian Portuguese. AMR is a recent and prominent meaning representation with good acceptance and several applications in the Natural Language Processing area. Following what has been done for other languages, and using an alignment-based approach for annotation, we annotated the Little Prince book, which went into the public domain and explored some language-specific annotation issues.

Keywords: Abstract Meaning Representation (AMR), corpus annotation, Portuguese language

1. Introduction

Due to its wide applicability and potentialities, Natural Language Understanding (NLU) has gained interest and fostered research on themes of computational semantics (Oepen et al., 2016). According to Ovchinnikova (2012), NLU is the field of Natural Language Processing (NLP) that deals with machine reading comprehension. The objective of an NLU system is to specify a computational model to interpret one or more input text fragments. The interpretation is usually carried out by a semantic parsing technique, which maps natural language into a suitable meaning representation.

A meaning representation is one of the most important components in semantic parsing. Its production is motivated by the hypothesis that semantics may be used to improve many natural language tasks, such as summarization, question answering, textual entailment, and machine translation, among others. In this context, there are several available meaning representations, as the traditional First-Order Logic (FOL), as detailed in Jurafsky and Martin (2009), semantic networks (Lehmann, 1992), Universal Networking Language (UNL) (Uchida et al., 1996), and, more recently, the Abstract Meaning Representation (AMR) (Banarescu et al., 2013).

In particular, AMR got the attention of the scientific community due to its relatively simpler structure, establishing the connections/relations among nodes/concepts, making them easy to read. Moreover, AMR structures are arguably easier to produce than traditional formal meaning representations (Bos, 2016).

According to Banarescu et al. (2013), AMR-annotated corpora are motivated by the need of providing to the NLP community datasets with embedded annotations related to the traditional tasks of NLP, for instance, named entity recognition, semantic role labeling, word sense disambiguation, and coreference. In this sense, the AMR annotation especially focuses on the predicate-argument structure as defined in PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005). Another characteristic of AMR annotation is that words that do not significantly contribute to the meaning of a sentence (which are referred as "syntactic sugar" in the original paper) are left out of the annotation, as articles and the infinitive particle "to".

From the currently available datasets, many semantic parsers emerged (Flanigan et al., 2014; Wang et al., 2015; Peng et al., 2015; Goodman et al., 2016; Zhou et al., 2016; Damonte et al., 2017). Furthermore, with the available parsers, some applications were developed for summarization (Liu et al., 2015) and text generation (Pourdamghani et al., 2016; Song et al., 2017), entity linking (Pan et al., 2015; Burns et al., 2016), and question answering (Mitra and Baral, 2016), among others.

Although there are some available annotated corpora, most of them are for English, producing a gap between English and other languages. In addition, creating such corpora is a very expensive task. For instance, Banarescu et al. (2013) took from 7 to 10 minutes to annotate a sentence in AMR representation. However, in spite of the difficulties, it is important to put some effort on corpus creation for other languages. Annotated corpora are important resources, as they provide qualitative and reusable data for building or improving existing parsers, and for serving as benchmarks to compare different approaches.

In order to fulfill this gap, we annotated a corpus in AMR representation for the (Brazilian) Portuguese language, which we report in this paper. In addition, we also detail some differences between Portuguese and English AMR annotations. To the best of our knowledge, this is the first initiative on AMR for Portuguese. We believe that the availability of such a semantic bank[1] in Portuguese will result in new semantic parsers for this language and support the development of more effective NLP applications.

In the following section, we briefly introduce the AMR fundamentals. In Sections 3 and 4, we present our corpus and report the annotation process and its results. Section 5 concludes the paper.

2. Abstract Meaning Representation

Abstract Meaning Representation (AMR) is a semantic representation language designed to capture the meaning of a sentence, abstracting away from elements of the surface syntactic structure, such as part of speech tags, word ordering, and morphosyntactic markers (Banarescu et al., 2013). It may be represented as a single-rooted acyclic directed

[1] A "SemBank", as referred in one of the first AMR papers.

graph with labeled nodes (concepts) and edges (relations) among them in a sentence. AMR concepts are either words (e.g., "girl"), PropBank framesets ("adjust-01"), or special keywords such as "date-entity", "distance-quantity", and "and", among others. PropBank framesets are essentially verbs linked to lists of possible arguments and their semantic roles. In Figure 1, we show a PropBank frameset example. The frameset "**edge.01**", which represents the "move slightly" sense, has six arguments (Arg 0 to 5).

Frameset **edge.01** "move slightly"

Arg0: causer of motion	Arg3: start point
Arg1: thing in motion	Arg4: end point
Arg2: distance moved	Arg5: direction

Ex: [$_{\text{Arg0}}$Revenue] *edge* [$_{\text{Arg5}}$ up] [$_{\text{Arg2-EXT}}$ 3.4%] [$_{\text{Arg4}}$ to $904 million] [$_{\text{Arg3}}$ from $874 million] [$_{\text{ArgM-TMP}}$ in last year's third quarter]. (wsj_1210)

Figure 1: A PropBank frameset (Palmer et al., 2005)

For semantic relationships, besides the PropBank semantic roles, AMR adopts approximately 100 additional relations, as general relations (e.g., :mod, :location, :condition, :name, and :polarity), relations for quantities (:quant, :unit, and :scale) and for dates (:day, :month, and :year), among others.

AMR may also be represented in two other notations: in first-order logic or in the PENMAN notation (Matthiessen and Bateman, 1991). For example, Figures 2 and 3 present the canonical form in PENMAN and graph notations, respectively, for the sentences with similar senses in Table 1.

Sentences
The girl made adjustment to the machine.
The girl adjusted the machine.
The machine was adjusted by the girl.

Table 1: Sentences with the same meaning

```
(a / adjust-01
      :ARG0 (g / girl)
      :ARG1 (m / machine))
```

Figure 2: PENMAN notation

AMR assigns the same representation to sentences that have the same basic meaning. Furthermore, as we may observe in the example, the concepts are "adjust-01", "girl", and "machine", and the relations are :ARG0 and :ARG1, represented by labeled directed edges in the graph. In Figure 2, the symbols "a", "g", and "m" are variables, which may be re-used in the annotation, corresponding to reentrancies (multiple incoming edges) in the graph.

Moreover, AMR represents negation in a different way. It uses the *:polarity* relation between the negated concept and the constant '−' (minus signal). For instance, the sentence "I do not much like to take the tone of a moralist.", extracted

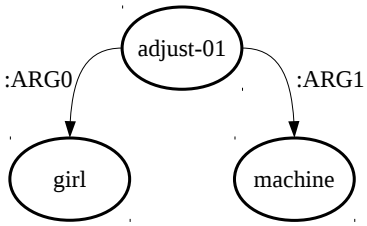


Figure 3: Graph notation

from the Little Prince book, produces the PENMAN notation in Figure 4.

```
(l / like-01 :polarity -
      :ARG0 (i / i)
      :ARG1 (t / take-01
            :ARG0 i
            :ARG1 (t1 / tone
                  :poss (m / moralist)))
      :degree (m1 / much))
```

Figure 4: PENMAN notation representing negation

Finally, to evaluate the AMR structures, Cai and Knight (2013) introduced the Smacth metric, which computes the degree of overlap between two AMR structures, computing precision, recall, and f-score over AMR annotation triples.

3. Our Corpus

There are some available corpora in the Linguistic Data Consortium (LDC), which offer texts in different domains but are not freely available. For now, only two AMR corpora are publicly accessible[2]: Bio AMR Corpus and the Little Prince Corpus. The first includes texts from the biomedical domain, extracted from PubMed[3], whereas the second contains the full text of the famous novel *The Little Prince*, written by Antoine de Saint-Exupéry. The novel was translated into 300 languages and dialects, including Brazilian Portuguese language. Unfortunately, none of the currently available AMR-annotated corpora are for Portuguese.

In this work, following what has been done for other languages, we annotated a public domain version of the Little Prince book written in Portuguese. As a collateral effect of this decision, we may also compare and analyze the annotation of the resulting parallel corpora, composed by the English (source) and Portuguese (target) versions of the book. The original book is organized into twenty-seven chapters. The English version has 1,562 sentences, while the Portuguese one has 1,527. In our annotation process, we aligned all the Portuguese sentences with the English sentences. Furthermore, we calculated some information about the two corpora, such as number of tokens and types, total number of concepts and relations, and maximum and minimum number of concepts and relations found in a sentence, which we show in Table 2.

[2] https://amr.isi.edu/download.html
[3] https://www.ncbi.nlm.nih.gov/pubmed/

Information	English	Portuguese
Number of tokes	16,998	12,703
Number of types	15,829	12,224
Number of concepts	10,528	7,569
Number of relations	10,245	6,676
Average number of tokens	10.88	8.31
Average number of nodes	6	4
Average number of relations	6	4
Maximum number of concepts	37	21
Minimum number of concepts	1	1
Maximum number of relations	49	25
Minimum number of relations	0	0

Table 2: Information about the corpora

4. The Annotation

As aforementioned, we chose as corpus a public domain version of the Little Prince book written in Brazilian Portuguese. Our corpus annotation strategy basically consisted of "importing" the corresponding AMR annotation for each sentence from the English annotated corpus and reviewing the annotation to adapt it to Portuguese characteristics. Doing this, we expected to save time and effort, as a significant part of AMR annotation is probably language independent. More than this, annotation agreement is minimally guaranteed, as it was already checked for the English annotation.

In this sense, we developed an approach with three steps, using the necessary tools and resources to "connect" the English and Portuguese versions of the corpus. Figure 5 illustrates them.

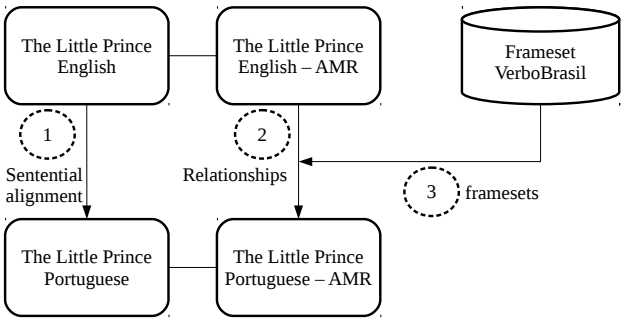


Figure 5: Adaptation of the corpus to the Portuguese language

In the first step, we performed a sentential alignment between the parallel corpora using the TCAlign tool (Caseli and Nunes, 2003), which has a 95% precision. Then, for each sentence, we imported/mapped the AMR relations from the original English sentence to the target Portuguese one. Finally, we included the framesets in each predicate using the *VerboBrasil* dataset (Duran et al., 2013). The *VerboBrasil* dataset is a repository with the sense of verbs in the Portuguese language, similar to the scheme illustrated in Figure 1. This dataset contains examples of a corpus annotated with semantic role labels, created by the PropBank-BR project (Duran and Aluísio, 2012), following the original PropBank initiative. We detail each step in what follows.

Even though the TCAlign tool has 95% precision, we manually checked each alignment, as such information is essential for producing a reliable annotation in Portuguese. We produced 1-1, 1-2, 2-1, 3-1, 1-3, 4-1, 1-4, and 1-5 alignments[4]. As examples, in Tables 3, 4, 5, 6, 7, 8 9, and 10, we present some resulting alignments produced by TCAlign that were manually revised. The overall number for each type of alignment is shown in Table 11. One may also see that there are six sentences in English without correspondence in Portuguese[5].

Source language	Target language
What I need is a sheep.	Preciso é de um carneiro.

Table 3: 1-1 alignment

Source language	Target language
I own three volcanoes, which I clean out every week (for I also clean out the one that is extinct).	Possuo três vulcões que revolvo toda semana.
	Porque revolvo também o que está extinto.

Table 4: 1-2 alignment

Source language	Target language
But I had never drawn a sheep.	Como jamais houvesse desenhado um carneiro, refiz para ele um dos dois únicos desenhos que sabia.
So I drew for him one of the two pictures I had drawn so often.	

Table 5: 2-1 alignment

Source language	Target language
In one of the stars I shall be living.	Quando olhares o céu de noite, porque habitarei uma delas, porque numa delas estarei rindo, então será como se todas as estrelas te rissem!
In one of them I shall be laughing.	
And so it will be as if all the stars were laughing, when you look at the sky at night... you - - only you - - will have stars that can laugh"	

Table 6: 3-1 alignment

In the following steps, we included the sense in each predicate in the sentence, using the *VerboBrasil* dataset, and mapped the relationships to the corresponding AMR relations. Figure 6 shows annotated parallel sentences, in English (left) and in Portuguese (right).

As we see, despite the supposed equality of meaning and annotation, the word '*eu*' (the pronoun "I" in English) does

[4]In an X-Y alignment, X sentences from the original document are aligned to Y sentences in the target one.

[5]Examples of these sentences are "And what good would it do to tell them that?", "Just that.", and "I said.".

Source language	Target language
One sits down on a desert sand dune, sees nothing, hears nothing.	A gente se senta numa duna de areia.
	Não se vê nada.
	Não se escuta nada.

Table 7: 1-3 alignment

Source language	Target language
Hum!.	Hem? respondeu o rei, que consultou inicialmente um grosso calendário.
Hum! ”replied the king; and before saying anything else he consulted a bulky almanac.	
Hum!	
Hum!	

Table 8: 4-1 alignment

Source language	Target language
After that would come the turn of the lamplighters of Russia and the Indies; then those of Africa and Europe, then those of South America; then those of South America; then those of North America.	Vinha a vez dos acendedores de lampiões da Rússia e das Índias.
	Depois os da África e da Europa.
	Depois os da América do Sul.
	Os da América do Norte.

Table 9: 1-4 alignment

Source language	Target language
But in herself alone she is more important than all the hundreds of you other roses: because it is she that I have watered; because it is she that I have put under the glass globe; because it is she that I have sheltered behind the screen; because it is for her that I have killed the caterpillars (except the two or three that we saved to become butterflies); because it is she that I have listened to, when she grumbled, or boasted, or ever sometimes when she said nothing.	Ela sozinha é, porém, mais importante que vós todas, pois foi a ela que eu reguei.
	Foi a ela que pus sob a redoma.
	Foi a ela que abriguei com o pára-vento.
	Foi dela que eu matei as larvas (exceto duas ou três por causa das borboletas).
	Foi a ela que eu escutei queixar-se ou gabar-se, ou mesmo calar-se algumas vezes.

Table 10: 1-5 alignment

not appear in the Portuguese sentence (as it was implicit), but it was annotated. In Portuguese, this phenomenon is called hidden (or implied) subject and it occurs when the subject is not explicit in the sentence but may be easily inferred. In order to keep the similarity with English annotation and the annotation consistency, we annotated all hidden subjects in the Portuguese sentences.

Alignment	Number
1-1	1,356
1-2	41
2-1	60
1-3	3
3-1	10
1-4	1
4-1	1
1-5	1
1-0	6

Table 11: Overall number of alignments

What I need is a sheep	**Preciso é de um carneiro**
(n / need-01	(p / precisar-01
:ARG0 (i / I)	:ARG0 (e / eu)
:ARG1 (s / sheep))	:ARG1 (c / carneiro))

Figure 6: Annotation of parallel sentences

In addition to the subject omission, there are some other differences in the translation into Portuguese. Consequently, the annotation for Portuguese sometimes becomes different from English. In some cases, translations are completely different, such as the one shown in Figure 7. In this example, the owner of the box (poss) and a box modifier (mod) were omitted.

This is only his box	**Esta é a caixa**
(b / box	(c / caixa
:poss (h / he)	:domain (e / esta))
:domain (t / this)	
:mod (o / only))	

Figure 7: An example of translation difference

Other differences are language-specific aspects such as the particle “*se*”, a multifunctional word in Portuguese (which, e.g., may represent the conditional “if” or a reflexive pronoun), words that change their part of speech tags and/or are joined in only one word, and other syntactic features. Figures 8 and 9 illustrate some cases. In Figure 8, one may see that the noun “sweetness” becomes the overall concept “sweet-05”, whereas in Portuguese the overall concept is the verb “*rir-01*” (“to laugh”, in English). Moreover, in Portuguese annotation, it is added the *:manner* relation and the “*docemente*” concept (corresponding to “sweetness”).
In Figure 9, the annotation in Portuguese was very different from the English version. Several concepts and relations were left out in Portuguese annotation, for example, the concepts “contrast-01”, “say-01”, “oh” and the relations “:mod” and “:ARG0-of” were omitted in Portuguese annotation. Moreover, we added the “:poss” relation in Portuguese annotation.
Aiming to organize the number of some of these occurrences/phenomena, we computed and summarized them in Table 12. It is important to notice that the hidden subject phenomenon does not change the original annotation, as we make them explicit. An indeterminate subject, on the other hand, is another type of subject (that may include

And there is sweetness in the laughter of all the stars.

```
(a / and
      :op2 (s / sweet-05
            :ARG1 (l / laugh-01
                  :ARG0 (s1 / star
                        :mod (a2 / all)))))
```

E todas as estrelas riem docemente

```
(e / e
      :op2 (r / rir-01
            :ARG0 (e1 / estrelas
                  :mod (t / todas))
            :manner (d / docemente)))
```

Figure 8: Syntactic structuring variation

But the little prince could not restrain his admiration : " Oh !

```
(c / contrast-01
    :ARG2 (p2 / possible-01 :polarity -
        :ARG1 (r / restrain-01
            :ARG0 (p / prince
                :mod (l / little)
                :ARG0-of (s / say-01
                    :ARG1 (o / oh :mode "expressive")))
            :ARG1 (a / admire-01
                :ARG0 p))))
```

O principezinho, então, não pôde conter o seu espanto

```
(p / poder-201 :polarity -
      :ARG0 (p1 / principezinho)
      :ARG1 (c / conter-02
            :ARG0 p1
            :ARG1 (e / espanto
                  :poss p1)))
```

Figure 9: Syntactic structuring variation

the particle "*se*") that may result in changes in the original annotation. The same happens for some different translations, mainly when they incorporate language specific expressions and constructions.

Phenomenon	#	%
Different translation	494	32.35
Syntactic variation	341	22.33
Hidden subject	285	18.66
Missing verb or sense	191	12.50
Change of predicate	100	6.54
Indeterminate subject	68	4.45
Complex predicate	3	0.19

Table 12: Annotation features in Portuguese

In addition to these phenomena, we calculated the incidence of syntactic variations, changes in predication, and missing verbs or senses. Syntactic variations include part of speech changes, as "little prince" (noun-adjective) to "*principezinho*" (noun), "grown-ups" (noun) to "*pessoas grandes*" (noun-adjective), and "boa constrictor" (noun-noun) to "*jibóia*" (noun), among others. Change of predicate occurs when the predicate in Portuguese is different from English. Thus, this change may produce different arguments.

We also computed the number of included arguments (25) and excluded arguments (103) in relation to English. It is also important to add that *VerboBrasil* is still a small dataset compared to PropBank, and, therefore, did not contain all verbs and senses. In cases where the verbs were not in the dataset, we assigned the sense "01" to the verbs, and marked them in the corpus in order to subsidize future improvements in the *VerboBrasil* repository. These cases occurred 191 times.

A final interesting issue is that importing the AMR structures from the English annotation is helpful, but still demands some effort due to the language specificities. As an illustration, each sentence in Portuguese has 8.31 words in average, and we took about 6 minutes to annotate each one, which is less than the English original annotation from scratch, but is still expensive.

5. Final remarks

The annotated corpus should be made available soon, as the Little Prince book went into public domain. We expect that such annotation may foster research in semantic parsing for Portuguese. Our next steps include to perform wikification of the words, as this also happened for English and looks as a natural step to follow.

More than the annotated corpus availability, our contributions are the proposal of an alignment-based approach for AMR annotation, which we believe that may also be used for other language pairs, and the investigation of annotation issues that may be language specific (in spite of the fact of AMR being a meaning representation).

Acknowledgments

The authors are grateful to FAPESP, CAPES, and *Instituto Federal do Piauí* for supporting this work.

References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Palmer, M., and Schneider, N. (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Bos, J. (2016). Expressive power of abstract meaning representations. *Computational Linguistics*, pages 527–535.

Burns, G. A., Hermjakob, U., and Ambite, J. L. (2016). Abstract meaning representations as linked data. In *Proceedings of the 15th International Semantic Web Conference*, pages 12–20.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the*

51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 748–752.

Caseli, H. and Nunes, M. (2003). Sentence alignment of brazilian portuguese and english parallel texts. In *Proceedings of the Argentine Symposium on Artificial Intelligence*, pages 1–11.

Damonte, M., Cohen, S. B., and Satta, G. (2017). An incremental parser for abstract meaning representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 536–546.

Duran, M. S. and Aluísio, S. M. (2012). Propbank-br: a brazilian treebank annotated with semantic role labels. In *Proceedings of the 8th international conference on Language Resources and Evaluation*, pages 1862–1867.

Duran, M. S., Martins, J. P., and Aluísio, S. M. (2013). Um repositório de verbos para a anotação de papéis semânticos disponível na web. In *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology*, pages 168–172.

Flanigan, J., Thomson, S., Carbonell, J. G., Dyer, C., and Smith, N. A. (2014). A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1426–1436.

Goodman, J., Vlachos, A., and Naradowsky, J. (2016). Noise reduction and targeted exploration in imitation learning for abstract meaning representation parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1–11.

Jurafsky, D. and Martin, J. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Prentice Hall.

Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 1989–1993.

Lehmann, F. (1992). *Semantic networks in artificial intelligence*. Elsevier Science Inc.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086.

Matthiessen, C. and Bateman, J. A. (1991). *Text generation and systemic-functional linguistics: experiences from English and Japanese*. Pinter Publishers.

Mitra, A. and Baral, C. (2016). Addressing a question answering challenge by combining statistical methods with inductive rule learning and reasoning. In *Proceedings of the 30th Conference on Artificial Intelligence*, pages 2779–2785.

Oepen, S., Kuhlmann, M., Miyao, Y., Zeman, D., Cinkova, S., Flickinger, D., Hajic, J., Ivanova, A., and Uresova, Z. (2016). Towards comparability of linguistic graph banks for semantic parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 3991–3995.

Ovchinnikova, E. (2012). *Integration of World Knowledge for Natural Language Understanding*. Atlantis Thinking Machines. Atlantis Press.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Pan, X., Cassidy, T., Hermjakob, U., Ji, H., and Knight, K. (2015). Unsupervised entity linking with abstract meaning representation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1130–1139.

Peng, X., Song, L., and Gildea, D. (2015). A synchronous hyperedge replacement grammar based approach for amr parsing. In *Proceedings of the 9th Conference on Computational Language Learning*, pages 32–41.

Pourdamghani, N., Knight, K., and Hermjakob, U. (2016). Generating english from abstract meaning representations. In *Proceedings of the 9th International Conference on Natural Language Generation*, pages 21–25.

Song, L., Peng, X., Zhang, Y., Wang, Z., and Gildea, D. (2017). Amr-to-text generation with synchronous node replacement grammar. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 7–13.

Uchida, H., Zhu, M., and Della Senta, T. (1996). Unl: Universal networking language–an electronic language for communication, understanding, and collaboration. *Tokyo: UNU/IAS/UNL Center*.

Wang, C., Xue, N., Pradhan, S., and Pradhan, S. (2015). A transition-based algorithm for amr parsing. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 366–375.

Zhou, J., Xu, F., Uszkoreit, H., Qu, W., Li, R., and Gu, Y. (2016). Amr parsing with an incremental joint model. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 680–689.

Towards A Welsh Semantic Annotation System

Scott Piao[1], Paul Rayson[1], Dawn Knight[2] and Gareth Watkins[2]
[1]School of Computing and Communications, Lancaster University, UK
[2]School of English, Communication and Philosophy, Cardiff University, UK
{s.piao;p.rayson}@lancaster.ac.uk, knightd5@cardiff.ac.uk, gllwatkins@gmail.com

Abstract

Automatic semantic annotation of natural language data is an important task in Natural Language Processing, and a variety of semantic taggers have been developed for this task, particularly for English. However, for many languages, particularly for low-resource languages, such tools are yet to be developed. In this paper, we report on the development of an automatic Welsh semantic annotation tool (named CySemTagger) in the CorCenCC Project, which will facilitate semantic-level analysis of Welsh language data on a large scale. Based on Lancaster's USAS semantic tagger framework, this tool tags words in Welsh texts with semantic tags from a semantic classification scheme, and is designed to be compatible with multiple Welsh POS taggers and POS tagsets by mapping different tagsets into a core shared POS tagset that is used internally by CySemTagger. Our initial evaluation shows that the tagger can cover up to 91.78% of words in Welsh text. This tagger is under continuous development, and will provide a critical tool for Welsh language corpus and information processing at semantic level.

Keywords: Welsh semantic tagger, corpus annotation, Welsh language, Welsh corpus, CorCenCC

1. Introduction

Automatic semantic annotation and analysis is an important task for Natural Language Processing (NLP), and semantic taggers have been developed and used for carrying out semantic analysis of language data on a large scale. A major tool built for such a purpose is USAS (UCREL Semantic Analysis System)[1] (Rayson et al., 2004; Piao et al., 2017), which is designed to annotate each word or phrase in text with lexical semantic categories derived from Tom McArthur's Longman Lexicon of Contemporary English (McArthur, 1981), such as "Food" or "Time". The USAS semantic scheme contains 21 major semantic fields that are further divided into 232 sub-categories. Initially developed for processing English text, it has been extended to tag texts of a number of languages, including Italian, Chinese, Spanish etc. (Piao et al., 2015).

In the CorCenCC Project[2], we have been developing a Welsh semantic tagger, named CySemTagger, modelled on the USAS framework, which employs a translated USAS semantic classification scheme and tagset for Welsh language. During the course of development, we have first constructed large Welsh semantic lexicons containing approximately 136,468 Welsh word entries, which provide a lexical knowledge base for the semantic tagging system[3]. Based on the lexicons, we have developed an initial version of the tagger software system that is designed to accommodate different Welsh POS taggers and tagsets that exist or will be developed. In the CorCenCC project, CySemTagger will be mainly based on a new Welsh POS tagger, named CyTag, which has been developed in this Project (Neale et al., 2018). In this paper, we describe the CySemTagger system and report on its initial evaluation.

The remaining part of this paper is organised as follows. Section 2 will discuss related work, Section 3 will describe the architecture of the CySemTagger, Section 4 will discuss the construction of Welsh semantic lexicon, Section 5 will explain about detailed working mechanism of CySemTagger's main components, Section 6 will discuss an evaluation of the current version of CySemTagger, and Section 7 will conclude our work and discuss future work.

2. Related Work

Over recent years, various semantic annotation tools have been developed in the NLP community. These tools are used to automatically recognise and annotate various semantic categories and concepts at different syntactic levels, such as word level, phrase level, sentence level etc.

Among the major existing semantic taggers developed in NLP communities is USAS (Rayson et al., 2004; Piao et al., 2017), GATE[4] (Cunningham et al., 2011), Freeling (Padro and Stanilovsky, 2012), NLTK[5] (Bird et al., 2009) etc., which provide functionalities of semantic annotation of various types, such as WordNet's Word sense IDs or Named Entity types etc. For example, GATE and KIM (Popov et al., 2003), combined together, provide multilingual semantic tagging function based on ontologies. Freeling is capable of detecting and tagging multilingual texts with named entity types and WordNet senses. Zhang and Rettinger (2014) developed a toolkit that carries out Wikipedia-based annotation. NLTK (Bird et al., 2009) provides a function for analysing the meaning of sentences.

What is directly related to our work is the past development of multilingual functionality of the USAS framework. As mentioned earlier, initially developed for English, it has been extended and modified to cover an increasing number of languages. Currently USAS is capable of carrying out semantic annotation on 12 languages, including Italian, Finish, Russian, Chinese, Spanish, Portuguese, Swedish,

[1]For further details of USAS, see website http://ucrel.lancs.ac.uk/usas/

[2]For information about this project, see website http://www.corcencc.org/

[3]These lexicons are made available at https://github.com/UCREL/Multilingual-USAS

[4]See website https://gate.ac.uk/

[5]http://www.nltk.org/

Dutch etc. (Lofberg et al., 2005; Mudraya et al., 2006; Piao et al., 2015).

Our research draws upon the experiences of the previous work, while extending the capability of automatic semantic annotation of existing tools to the Welsh language, for which no existing NLP tools can provide such a comprehensive semantic annotation as CySemTagger aims to perform. Our new tagger will facilitate new semantic based research on Welsh language data, and via the CorCenCC framework will assist with improving the understanding of real-life communication for Welsh speakers, teachers and learners.

3. Outline of CySemTagger's Architecture

The CySemTagger is a software system which has an architecture consisting of a set of lexical knowledge resources and disambiguation rules and models wrapped in software components that interact with each other. Figure 1 illustrates the outline of the architecture and main workflows of the system.

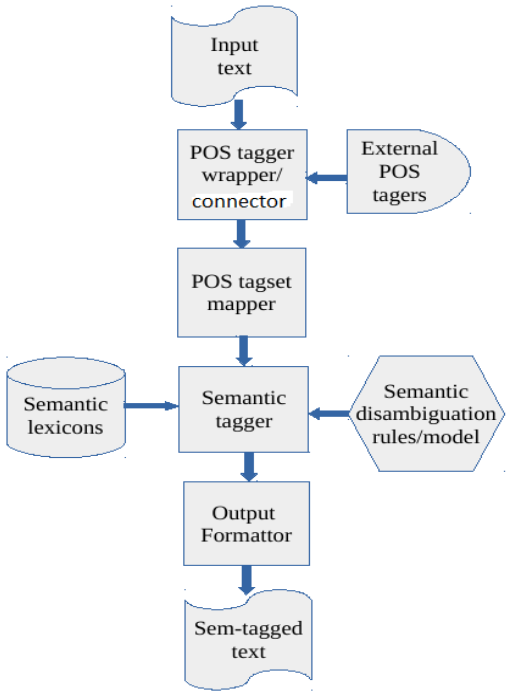


Figure 1: CySemTagger Software Architecture

As shown in the figure, a set of software modules provide main functionalities, such as receiving input data stream, calling external part-of-speech taggers to carry out morpho-syntactic analysis, produce tagged output in required formats etc. Together, they form a pipeline to interact with each other to complete the whole task of semantic tagging. The main challenge for building such a system for a new language, Welsh in our case, is the development of semantic lexicons and word sense disambiguation rules and models for the new language. Because there are very few Welsh semantic lexical resources available, we were faced with a tough challenge for building the Welsh semantic tagger.

4. Semantic Lexicon Construction

As a system based on linguistic knowledge, a core part of the semantic tagger system is a set of semantic lexicons which provide candidate semantic categories for each word, hence Welsh semantic lexicon construction is the first main step for the development of Welsh semantic tagger. For this purpose, we exploited various lexical and corpus resources. The main lexical resource used for this work is the Eurfa Welsh/English bilingual lexicon developed by Donnelly (2017). This bilingual lexicon contains a large number of Welsh words and their English translations along with useful information such as lemma forms and part-of-speech (POS) labels. It also contains many Welsh multi-word expressions (MWEs), which is a valuable resource for creating semantic MWE lexicons for the semantic tagger in later stages. Because it is time-consuming work to manually compile new semantic lexicons from scratch, we applied automatic methods by mapping and porting semantic categories and tags for Welsh words via their English translations through the existing English semantic lexicons. This method has been proven effective in our previous research on other languages (Piao et al., 2016). The high quality of the Eurfa bilingual lexicon helped us to achieve a good initial result for the automatically generated Welsh semantic lexicon. Obviously, the automatically generated lexicon will need be pruned manually or by other methods to guarantee the accuracy of the semantic annotation. Through the automatic process, we extracted a lexicon containing 136,468 Welsh words (including many inflected forms) mapped to semantic category/ies. It provides a solid basis for developing a system of Welsh semantic tagger.

In addition to the automatic lexicon generation, we also collected 264 Welsh closed class words and integrated them into the lexicon, mainly including function words such as prepositions, conjunctions etc. Although there are a limited number of such words, they are critical for correctly understanding the meaning of the text, and typically are high frequency items. Another important lexical source is the Welsh names. Our initial observation showed that a significant proportion of corpus data consists of various names. Therefore, we searched and collected Welsh names, including person names and place names, from various sources[6], including the Language Technologies Unit of Bangor University[7], UK and the websites of "Behind The Name"[8], "Think Baby Names"[9], and "Wales UK"[10]. As a result, we collected 6,553 Welsh names to expand the semantic lexicon. Combining automatic and manual processes, currently we have constructed a Welsh single word semantic lexicon of 143,287 entries. Table 1 shows a sample of the single word semantic lexicon, where the semantic tags are from the USAS semantic tagset[11].

[6]The creators/owners of these name sources gave us permissions to use their Welsh name resources for the purpose of developing the Welsh semantic tagger

[7]https://www.bangor.ac.uk/canolfanbedwyr/technolegau_iaith.php.en

[8]http://www.behindthename.com/names/usage/welsh

[9]http://www.thinkbabynames.com/names/1/welsh

[10]http://www.walesuk.info/wales.html

[11]For definitions of the semantic tags, see http://ucrel.lancs.ac.uk/usas/usas_guide.pdf

Table 1: Sample of Single Word Semantic Lexicon

Word	POS-tag	Sem-tag
abacws	noun	N3.1
corfforaethol	adj	I2.1/S5
galluogai	verb	S8+
gambled	verb	A15- I1/A1.4 K5.2/I1
sobr	adv	A13.3

Our semantic tagger also needs a Welsh MWE semantic lexicon, which provides semantic categories for MWE terms and non-compositional idiomatic expressions, e.g. "take care of" and "kick the bucket" (for English) etc. Currently we use a sample MWE semantic lexicon for testing software, but work is under way to construct a large MWE semantic lexicon as the project progresses. Table 2 shows samples of MWE semantic lexicon entries, including template codes.

Table 2: Sample of MWE Semantic Lexicon

MWE/Templates	Sem-tag
Adran_NOUN Iechyd_NOUN	G1.1
gofyn_VERB {NOUN/VERB} i_PREP	Q2.1
parhad_NOUN busnes_NOUN	I2.1/T2+
pwyllgor*_NOUN addysg_NOUN	P1/G1.1 G1.2

Besides the existing English/Welsh bilingual lexical resources, we are also considering Welsh corpus resources, particularly Welsh/English parallel corpora, for further expanding and improving the Welsh semantic lexicons. We carried out an initial experiment by collecting words from the existing Welsh corpora, including CEG Cronfa Electroneg o Gymraeg (Ellis et al., 2001), Kwici (Corpus of Welsh Wikipedia)[12] and Corpus of Children's Literature in Welsh[13], and estimated the proportion of text that can be covered by the word list, together with the formal semantic lexicon. Our experiment shows that, if all the newly collected words can be integrated into the Welsh semantic lexicons, our semantic tagger could achieve over 97% of text coverage. Therefore we aim to semantically classify as many words in the word collection as possible and integrate them into the semantic lexicon in order to achieve a high lexical/text coverage.

Some unique features of Welsh language present a tough challenge for the Welsh semantic lexicon building. As a Celtic language, the Welsh language's linguistic features are widely different to those of the English language (a Germanic language). For instance, while the English alphabet consists of 26 letters, the Welsh alphabet consists of 29 letters, including some letters which are made up of two characters (namely "Ch", "Dd", "Ff" "Ng", "Ll", "Ph", "Rh" and "Th"). The Welsh language is different from English in terms of grammar, there is no indefinite article in the Welsh language, for instance. Furthermore, the Welsh language is unique in that it employs a system of mutation, that is, under certain circumstances the first letter of a word is substituted for another. For example, Welsh feminine nouns which follow the Welsh definite article "y" mutate as shown in 3 (see the highlighted letters).

Table 3: Example of mutation following the definite article in Welsh

Welsh	English	Mutation
Pioden	Magpie	Y **B**ioden
Craith	Scar	Y **G**raith
Teml	Temple	Y **D**eml

In respect of English equivalents to Welsh words, a one-to-one relationship does not always exist, as shown in 4. For instance, the English word "together" would normally be translated as the MWE "gyda'i gilydd" in Welsh. Conversely, the Welsh word "haprif" would be equivalent to "random number" in English. These factors present a tough challenge for the automatic or semi-automatic creation of Welsh semantic lexicons based on existing English ones, and requires more manual efforts in this process, particularly for MWE lexicon construction.

Table 4: Examples of Equivalence Between English and Welsh Single Words and MWEs

English	Welsh
beyond repair	anadferadwy
random number	haprif
take pride in	ymfalchïo
lifeboat	bad achub
yorker (a cricket term)	pelen lawn
Iceland	Gwlad yr Iâ
reserve (in sport)	chwaraewr wrth gefn
desktop	bwrdd gwaith
toadstool	caws llyffant

5. Main Components of CySemTagger

The current version of the CySemTagger is based on the Welsh semantic lexicons constructed so far. With regards to major functionality, the system mainly consists of four modules:

1) Lexicon look-up (both for single words and MWEs),
2) Part-of-speech tagging,
3) Semantic category disambiguation,
4) Output formatting.

The first main module is for loading the lexicons into the system and looking up candidate categories for each word. For single words this is a straightforward process, whereas a complex algorithm is needed for the MWE lookup. In the USAS framework, the MWE entries can contain specified template codes and format which are used to represent similar MWEs with a single lexicon entry. For example, the entry below:

```
spe*d_* {R*} off_RP
```

represents MWEs "sped off", "speed off", and "sped quickly off" etc. Internally in the software, the MWE entries are transformed into regular-expression based matchers when loaded into the system. This technique allows the

[12] http://cy.wikipedia.org
[13] http://www.egni.org

semantic tagger to identify and annotate a large amount of MWE variants using a moderate-sized MWE lexicon.

Next, the part-of-speech module is a software wrapper for external POS taggers. It is a program component that links and wraps external Welsh POS taggers built independently into the CySemTagger system. We started with an existing Welsh POS tagger, WNLT (Welsh Natural Language Toolkit)[14], which was publicly available when our project began. With the development of the new Welsh POS tagger, CyTag, in the CorCenCC Project (Neale et al., 2018), it has also been integrated into the semantic tagger. Considering the existence of multiple POS taggers, and in order to make CySemTagger compatible with different POS tagsets, we designed a core POS tagset that provides sufficient information for the semantic annotation purpose and to which other Welsh POS tagsets can easily be mapped. Such a design makes CySemTagger flexible to potentially accommodate existing and future POS taggers and provides a wider choice for users. Table 5 lists the core POS tagset.

Table 5: Core POS Tags

POS Tag	Definition
noun	Noun
verb	Verb
adj	Adjective
adv	Adverb
num	Numerals
pnoun	Proper noun
intj	Interjection
art	Article
part	Particle
prep	Preposition
conj	Conjunction
pron	Pronoun
code	Special code, e.g. Maths symbol
punc	Punctuation
fw	Foreign word
abbrev	Abbreviation
lett	Letter
xx	Unregnized token

Table 6 lists the detailed mapping from the CyTag POS tagset[15] and WNLT POS tagset[16] to the core tagset. As shown in the table, different POS tagsets can have widely different levels of granularity. For instance, the CyTag POS tagset has 59 fine-grained sub-categories for Welsh verbs. As a result, the mapping of the tagsets is not straightforward for some POS categories such as *abbreviation* and *foreign words* etc. The core POS tagset is designed to accommodate all of the POS sub-categories included in the Cytag and WNLT tagsets, with four categories (*fw*, *abbrev*, *lett* and *XX*) only mapping with CyTag tags without corresponding WNLT tags. This may cause slight loss of POS information if the WNLT is used, but we expect that such a design provides the optimal practical solution for our system.

The disambiguation module for CySemTagger is in early stages of development, and will be reported in future papers. Various context aware algorithms will be tested and integrated to improve the accuracy of the tagger. The tagged output can be presented to users in various formats as necessary, such as CTV and XML.

The current version of CySemTagger has been built as a Web service for the convenience of integrating software written in different programming languages, including Java and python programs and the VISL's Constraint Grammar v3 (CG-3) package, and it can be accessed via a demo web site[17] and a desktop client GUI application[18].

6. Evaluation

In order to assess the performance of the current version of CySemTagger, we carried out a test based on a Welsh test corpus, a gold corpus, which is specifically compiled for the tool evaluation task in the CorCoeCC Project. The test corpus consists of text segments selected from four existing corpora, Kwici (Welsh Wikipedia)[19], Kynulliad[20] (Welsh Assembly Proceedings), Meddalwedd (software translations)[21], and LER-BIML (a small corpus of 10 multi-domain texts)[22], and contains around 15,000 words. In this experiment, we focused on examining the average text coverage of the tool, i.e. what percentage of the words in the test corpus can be identified by CySemTagger.

In detail, we examined the text coverage of the CySemTagger when it is linked to the two POS taggers CyTag (internal prototype version) and WNLT respectively. Because CyTag and WNLT use different tokenisation rules and algorithms, they produced different number of tokens. CyTag produced 13,220 words (excluding punctuations) of which 3,716 (28.11%) are function words; WNLT produced 14,435 words (excluding punctuations) of which 5,314 words (36.81%) are function words. Therefore, the text coverages are based on different word numbers, but they are comparable in terms of tool performance. Table 7 shows the text coverage statistics in terms of content words, function words and whole text respectively.

As shown by our experiment, CySemTagger is capable of covering about 91.78% of Welsh running text when it uses CyTag POS tagger, which is under continuous development in the CorCenCC Project. It still covers 72.92% of text when using the WNLT. Our initial analysis reveals that an important factor for the difference of the text coverages is the performance of lemmatisation of Welsh words, for which CyTag has a superior accuracy compared to WNLT.

Due to the lack of manually annotated Welsh test data, it was not possible to carry out an evaluation on the quality of the semantic annotation. Currently a test corpus for

[14] See https://sourceforge.net/projects/wnlt/

[15] For details of the CyTag POS tagset, see website: http://cytag.corcencc.org/tagset

[16] For details of the WNLT POS tagset, see website: https://sourceforge.net/projects/wnlt/files/user-guide.pdf

[17] See demo web site http://phlox.lancs.ac.uk/ucrel/semtagger/welsh

[18] A Java desktop application downloadable at http://ucrel.lancs.ac.uk/usas/gui/

[19] http://cymraeg.org.uk/kwici

[20] http://cymraeg.org.uk/kynulliad3

[21] http://techiaith.cymru/corpws/Moses/Meddalwedd

[22] http://www.lancaster.ac.uk/fass/projects/biml

Table 6: Map of Welsh POS Tagsets

Core Tags	CyTag Tags	WNLT Tags
noun	Egu Ebu Egll Ebll Egbu Egbll	NN NNS NNM NNF
verb	Be Bpres1u Bpres2u Bpres3u Bpres1ll Bpres2ll Bpres3ll Bpresamhers Bpres3perth Bpres3amhen Bdyf1u Bdyf2u Bdyf3u Bdyf1ll Bdyf2ll Bdyf3ll Bdyfamhers Bgorb1u Bgorb2u Bgorb3u Bgorb1ll Bgorb2ll Bgorb3ll Bgorbamhers Bamherff1u Bamherff2u Bamherff3u Bamherff1ll Bamherff2ll Bamherff3ll Bamherffamhers Bgorff1u Bgorff2u Bgorff3u Bgorff1ll Bgorff2ll Bgorff3ll Bgorffamhers Bgorffsef Bgorch2u Bgorch3u Bgorch1ll Bgorch2ll Bgorch3ll Bgorchamhers Bdibdyf1u Bdibdyf2u Bdibdyf3u Bdibdyf1ll Bdibdyf2ll Bdibdyf3ll Bdibdyfamhers Bamod1u Bamod2u Bamod3u Bamod1ll Bamod2ll Bamod3ll Bamodamhers	VB VBD VBDP VBDI VBI VBF
adj	Anscadu Anscadbu Anscadll Anscyf Anscym Anseith	JJ JJR JJS PDT
adv	Adf	RB
num	Rhifol Rhifold Rhifolt Rhitref Rhitrefd Rhitreft Gwdig Gwrhuf	CD
pnoun	Epg Epb	NNP NNPS
intj	Ebych	UH
art	YFB	DT
part	Uneg Ucad Ugof Utra Uberf	RP
prep	Arsym Ar1u Ar2u Ar3gu Ar3bu Ar1ll Ar2ll Ar3ll	IN
conj	Cyscyd Cysis	CC
pron	Rhapers1u Rhapers2u Rhapers3gu Rhapers3bu Rhapers1ll Rhapers2ll Rhapers3ll Rhadib1u Rhadib2u Rhadib3gu Rhadib3bu Rhadib1ll Rhadib2ll Rhadib3ll Rhamedd1u Rhamedd2u Rhamedd3gu Rhamedd3bu Rhamedd1ll Rhamedd2ll Rhamedd3ll Rhacys1u Rhacys2u Rhacys3gu Rhacys3bu Rhacys1ll Rhacys2ll Rhacys3ll Rhagof Rhadangg Rhadangb Rhadangd Rhaperth Rhaatb Rhacil	PP INT
code	Gwfform Gwsym	SC
punc	Atdt Atdcan Atdchw Atdde Atdcys Atddyf	PN
fw	Gwest	
abbrev	Gwacr Gwtalf	
lett	Gwllyth	
xx	Gwann	

Table 7: Text Coverage of CySemTagger

Word-type	CyTag	WNLT
content words	88.69%	65.42%
function words	99.70%	85.79%
total	91.78%	72.92%

the semantic tagger is under compilation by manually annotating the gold corpus by language experts in the CorCenCC project. When this test data becomes available, we will carry out a full scale evaluation of the semantic tagger, including evaluation of the contextual disambiguation accuracy.

7. Conclusion

In this paper, we have reported on our development of a Welsh Semantic tagger carried out in the CorCenCC Project. We applied various approaches for rapidly constructing Welsh semantic lexicons and extended and modified existing USAS software framework to develop the CySemTagger system, aiming to provide a tool for automatic annotation of Welsh language corpus data in large scales. In our evaluation, the prototype Welsh semantic tagger demonstrated an encouraging performance and, as it stands, already provides a useful tool for the semantic analysis of Welsh language data. This system is under continuous development, and we will investigate using cross-lingual word embeddings and other techniques, and integrate more efficient algorithms into the system to develop a wide-coverage and accurate Welsh semantic tagger, which will support a range of new research on Welsh Language data in a large scale.

8. Acknowledgement

This research work is funded by the UK Economic and Social Research Council (ESRC) and Arts and Humanities Research Council (AHRC) as part of the CorCenCC (Corpws Cenedlaethol Cymraeg Cyfoes; The National Corpus of Contemporary Welsh) Project (Grant Number ES/M011348/1).

References

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python - Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.

Cunningham, H., Maynard, D., and Bontcheva, K. (2011). *Text Processing with GATE*. Gateway Press CA.

Donnelly, K. (2017). Eurfa, a gpled dictionary of welsh. `http://eurfa.org.uk`. Online publication.

Ellis, N. C., O'Dochartaigh, C., Hicks, W., Morgan, M., and Laporte, N. (2001). Cronfa electroneg o gymraeg (ceg): A 1 million word lexical database and frequency count for Welsh. https://www.bangor.ac.uk/canolfanbedwyr/ceg.php.en. Online publication.

Lofberg, L., Piao, S., Nykanen, A., Varantola, K., Rayson, P., and Juntunen, J.-P. (2005). A semantic tagger for the Finnish language. In *Proceedings of the Corpus Linguistics Conference 2005*.

McArthur, T. (1981). *Longman Lexicon of Contemporary English*. Longman London.

Mudraya, O., Babych, B., Piao, S., Rayson, P., and Wilson, A. (2006). Developing a Russian semantic tagger for automatic semantic annotation. In *Proceedings of International Corpus Linguistics Conference 2006*, pages 290–297.

Neale, S., Donnelly, K., Watkins, G., and Knight, D. (2018). Leveraging lexical resources and constraint grammar for rule-based part-of-speech tagging in Welsh. In *Proceedings of the LREC 2018 Conference*.

Padro, L. and Stanilovsky, E. (2012). Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*.

Piao, S., Bianchi, F., Dayrell, C., D'Egidio, A., and Rayson, P. (2015). Development of the multilingual semantic annotation system. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015)*. Association for Computational Linguistics.

Piao, S., Rayson, P., Archer, D., Bianchi, F., Dayrell, C., El-Haj, M., Jimenez, R.-M., Knight, D., Kren, M., Lofberg, L., Nawab, R., Shafi, J., Teh, P., and Mudraya, O., (2016). *Lexical coverage evaluation of large-scale multilingual semantic lexicons for twelve languages*, pages 2614–2619. European Language Resources Association (ELRA), 5. The LREC 2016 Proceedings are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Piao, S., Dallachy, F., Baron, A., Demmen, J., Wattam, S., Durkin, P., McCracken, J., Rayson, P., and Alexander, M. (2017). Time-sensitive historical thesaurus-based semantic tagger for deep semantic annotation. *Computer Speech and Language*, 46:113–135.

Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., and Goranov, M. (2003). Kim - semantic annotation platform. In *Proceedings of 2nd International Semantic Web Conference (ISWC2003)*, pages 834–849.

Rayson, P., Archer, D., Piao, S., and McEnery, A. (2004). The UCREL semantic analysis system. In *Proceedings of the Beyond Named Entity Recognition Semantic Labelling for NLP Tasks Workshop*, pages 7–12.

Zhang, L. and Rettinger, A. (2014). Semantic annotation, analysis and comparison: A multilingual and cross-lingual text analytics toolkit. In *Proceedings of the Demonstrations at the EACL 2014*, pages 13–16.

Semantic Frame Parsing for Information Extraction : the CALOR corpus

Gabriel Marzinotto[1,2], Jeremy Auguste[2], Frederic Bechet[2], Geraldine Damnati[1], Alexis Nasr[2]
(1) Orange Labs, Lannion, France
(2) Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
{gabriel.marzinotto, geraldine.damnati}@orange.com
{jeremy.auguste, frederic.bechet, alexis.nasr}@lis-lab.fr

Abstract

This paper presents a publicly available corpus of French encyclopedic history texts annotated according to the Berkeley FrameNet formalism. The main difference in our approach compared to previous works on semantic parsing with FrameNet is that we are not interested here in *full text parsing* but rather on *partial parsing*. The goal is to select from the FrameNet resources the minimal set of frames that are going to be useful for the applicative framework targeted, in our case Information Extraction from encyclopedic documents. Such an approach leverages the manual annotation of larger corpora than those obtained through full text parsing and therefore opens the door to alternative methods for Frame parsing than those used so far on the FrameNet 1.5 benchmark corpus. The approaches compared in this study rely on an integrated sequence labeling model which jointly optimizes frame identification and semantic role segmentation and identification. The models compared are CRFs and multitasks bi-LSTMs.
Keywords: Frame Semantic Parsing, LSTM, CRF

1. Introduction

Semantic Frame parsing is a Natural Language Understanding task that involves detecting in a sentence an event or a scenario, called *Frame*, as well as all the elements or roles that can be associated to this event in the sentence, called *Frame Elements*. One of the most popular semantic frame model is the Berkeley FrameNet project developed by ICSI Berkeley (Baker et al., 1998). This model is composed of an inventory of Frames with, for each of them, a list of words, called *Lexical Units* (or LU), that can trigger a frame in a sentence. Besides, for each frame, a list of *Frame Elements* (FE), core or optional, is defined. LUs are pairings of a word with a sense; Frame Elements are the components of a frame, represented by sequences of words in a sentence.

Two kinds of parsing can be done with a Semantic Frame model: *full text parsing* where each word in a sentence is analyzed to check if it can trigger a frame; and *partial parsing* where only a subset of frames and LUs is considered, according to their relevance for a given applicative framework. Annotating all the possible LUs and frames in a sentence is a very difficult (and expensive) task for human annotators, therefore there are very few corpora annotated this way. Moreover not many languages have such resources. Most of previous work in semantic frame parsing that has been done with a *full text parsing* approach have used the benchmark corpus FrameNet 1.5. Although the size of this benchmark is relatively large, a lot of frames have a very small number of occurrences in the corpus. This is due to the very large number of frames considered in the semantic model and this makes this corpus particularly challenging for machine learning methods.

On the contrary *partial parsing* can be made on large corpora at a reasonable cost: because the amount of frames and LUs is limited, the annotators can focus only on a few words for each sentence, making the task much easier than full parsing. Corpus obtained this way contain much more examples for each frame, opening the door to more machine learning methods than it is the case with full parsing.

From an applicative point of view, *partial parsing* is also a more realistic option. Although the different senses from the FrameNet model are generic, models trained on the FrameNet 1.5. corpus are not. Assuming that a full text annotation will be available for each new applicative domain is not an option. Moreover a lot of applicative frameworks using semantic models such as frames, like the Information Extraction framework considered in this study, are not interested in full parses but on the contrary only in some specific senses related to the domain targeted.

An example of this kind of annotation scheme is given in figure 1. As we can see, only two words as considered as lexical units in the sentence: `decide`, which triggers the frame `Deciding` and `order` triggering the frame `Request`.

This paper presents a study on the use of Sequence Labeling models such as CRF and LSTM for Semantic Frame parsing. Unlike previous studies developing a multi-step approach involving first the Frame identification task, then the frame element detection and labeling tasks, we propose an approach detecting simultaneously LUs and Frame Elements making Frame identification and argument selection an integrated process. A simple heuristic filter is used in order to maintain coherence in the hypotheses produced. We compare two popular sequence labeling methods: conditional random fields and recurrent deep neural networks using Long Short-term Memory.

2. The CALOR-Frame corpus

We introduce in this paper the corpus CALOR, which is a collection of documents in French language that were hand annotated in frame semantics. This 1.3M words corpus contains documents from 4 different sources: Wikipedia's Archeology portal (WA, 201 documents), Wikipedia's World War 1 portal (*WGM*, 335 documents), Vikidia's portals of Prehistory and Antiquity (*VKH*, 183 documents) and ClioTexte's [1] resources about World War One (WW1)

[1] https://clio-texte.clionautes.org/

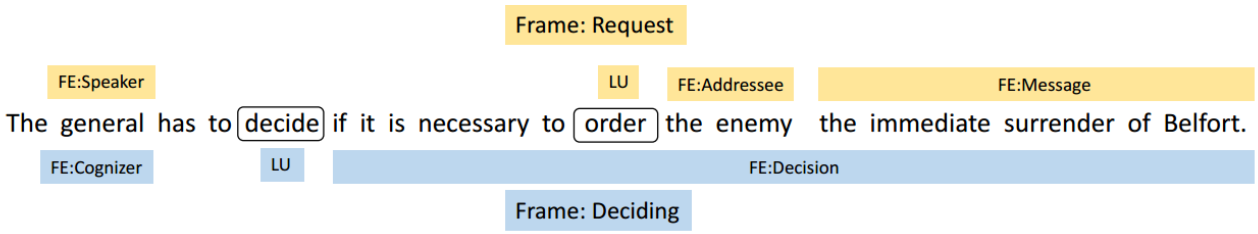


Figure 1: Example of Frame multilabel annotation

(*CTGM*, 16 documents). These sources were chosen to guarantee both writing style and domain diversity. By having documents from Vikidia (an encyclopedia addressed to children from 8-13 years old) and Wikipedia, presenting subjects of ancient history and archeology we can compare the influence of the writing style on the complexity of the task. The same analysis is possible on the WW1 documents, as Cliotexte (a collection of historical documents such as letters, essays, speeches from WW1) and Wikipedia share a common domain with a completely different writing style. This document selection allows to study the importance of the nature of the training data on the performance of the system on a test set on the same subject. Moreover, having data from two different portals of Wikipedia allows to study the domain dependency problem. For example: evaluate if a model for a frame F, trained on data from the archeology domain can successfully be applied on data from the WW1.

In contrast to full text parsing corpus, the frame semantic annotations of CALOR are limited to a small subset of frames from FrameNet (Baker et al., 1998). As described in the introduction, the goal of this *partial parsing* process is to obtain, at a relatively low cost, a large corpus annotated with frames corresponding to a given applicative context. In our case this applicative context is Information Extraction (IE) from encyclopedic texts, mainly historical texts.

To this purpose we extracted from our 1.3M word text corpus the top 100 most frequent verbs, then we kept those that were more likely to correspond to an action or a situation that would be relevant in our IE context. For example, verbs such as *discover* or *build* are very relevant for exploring archaeological documents. We looked for the corresponding frames of the selected verbs in the Berkeley FrameNet lexicon and kept a set of 53 different frames. If a verb could trigger several frames, we only kept those which were relevant in our corpus.

By adding noun triggers to the list of selected verb triggers we obtain a list of 145 Lexical Units (LU), with 30,950 occurrences in the training corpus. Selecting the most frequent verbs and nouns from the 1.3M words CALOR corpus as frame triggers is a guarantee that the average number of occurrences per frame is high. Therefore, even if the list of frames annotated in the CALOR corpus is small compared to the Framenet corpus, we have a large variety of occurrences for each of them, allowing us to build robust parsers for encyclopedic texts, which is the goal of the CALOR corpus. The list of Frames in CALOR is provided in Table 1.

2.1. The annotation process

Once the corpus, the lexical units and the frame set were chosen, we developed an iterative process for the manual annotation of the CALOR corpus. Preliminarily to this annotation process, the documents were automatically processed by the Macaon (Nasr et al., 2011) tool suite (sentence segmentation, tokenization, POS Tagging, lemmatization, and dependency parsing). Every word within the documents which lemma belongs to the set of selected LUs generates an example to be annotated. It is possible that one sentence generate several examples to annotate if it contains several LUs. We obtained a set of 30,950 examples to annotate, corresponding to all the LUs occurrences in the CALOR corpus. Three annotators were hired for this project. Their goal was to process these 30,950 examples: decide for each of them if its corresponding LU triggers or not one of the 53 frames selected, and finally annotate, if a frame was triggered, all its Frame Elements (FE) occurring in the sentence.
In order to reduce the manual annotation time and perform quality control on the corpus produced, we designed an iterative process based on three principles:

- an automatic pre-annotation scheme based on the frame parser that will be presented in section 4.1.;
- a batch selection process that selects from the unlabeled corpus a set of examples to annotate that corresponds to the same LU in very similar syntactic and lexical contexts.
- an automatic quality control estimator that regularly retrains the frame parser and evaluate its performance thanks to a *k-fold* experiments on the part of the corpus already manually annotated.

The iterative process based on these principles can be implemented as follows:

1. *Frame pre-annotation parsing* : an automatic frame parsing process is applied to each example to annotate. It predicts the frame label and the possible FEs for the LU contained in the example. All these automatic annotations are manually checked and eventually corrected by our annotators. At each iteration the frame parser (see section 4.1.) is trained on the subset of the CALOR corpus that is already annotated. For the first iteration, since there is no data to train the parser, all the LUs are labeled with a "*no frame*" label. Each iteration brings more data to train the frame parser.

Accomplishment	Activity-start	Age	Appointing	Arrest
Arriving	Assistance	Attack	Awareness	Becoming
Becoming-aware	Buildings	Change-of-leadership	Choosing	Colonization
Coming-to-believe	Coming-up-with	Conduct	Contacting	Creating
Death	Deciding	Departing	Dimension	Education-teaching
Existence	Expressing-publicly	Finish-competition	Giving	Hiding-objects
Hostile-encounter	Hunting	Inclusion	Ingestion	Installing
Killing	Leadership	Locating	Losing	Making-arrangements
Motion	Objective-influence	Origin	Participation	Request
Scrutiny	Seeking	Sending	Shoot-projectiles	Statement
Subjective-influence	Using	Verification		

Table 1: List of Frames annotated in the CALOR corpus

2. *Batch selection process*: the examples that have not yet been manually processed are sorted according to their LUs, then, using a similarity measure taking into account the lexical and syntactic context in which the LUs occur. They are then grouped into batches that will be sent to the annotators for manual validation. the goal here is to reduce the cognitive load of the annotators by splitting the corpus to annotate into small batches sharing very similar properties, likely to be annotated the same way.

3. *Manual correction* : a GUI allows annotators to work on the batches of examples produced in the previous step. Annotation is done on text only, no syntactic annotation is provided to the annotators. The frame pre-annotations produced by the frame parser are displayed, annotators can correct them and add what is missing.

4. *Model training and quality control validation* : at each iteration, the frame parsing models are re-trained on the corpus of manually processed examples. A *k-fold* evaluation is also performed to monitor the evolution of the parsing performance of the model when more validated data is added to the training corpus. If the frame parsing performance improves, it is a good indication that the added data is coherent with the annotations already processed in the previous iterations. This can be seen as a quality control measure of the annotation process on the whole dataset, in addition to inter-annotators agreement measures than can also be estimated on small subsets of the corpus.

2.2. Corpus Statistics

Table 2 presents the distribution of the CALOR corpus among its different sources. We observe that the two first rows of this table, corresponding to Wikipedia, represent most of the corpus. After the annotation process, 30,950 LU have been annotated, leading to 26,725 LU associated to a frame and 4,225 (13%) labeled as *OTHER*. In table 2 the columns **# Sentences**, **# Words**, **# Frames**, **# Other** and **# FE** display the number of sentences, words, frames, LUs, and Frame Elements. **% Sentences with Frame** displays the percentage of sentences with at least one frame and **Lexicon** corresponds to the size of the vocabulary of each document source. The CALOR corpus contains 57,688 FE annotations which averages to 2.2 FE per Frame occurrence.

Figure 2 displays the distribution of the number of annotated examples for each Frame in the CALOR corpus. We observe that the corpus has a large number of examples per frame: half of the frames in CALOR have more than 400 annotated examples and the 10 most frequent frames have more than 900 examples. The most common Frames are `Attack` (triggers: attaquer, attaque, offensive, bombardement, contre-attaque), `Leadership` (triggers: commander, diriger, commandement), `Activity Start` (triggers: commencer, débuter, commencement, début), `Locating` (triggers: retrouver, trouver, localisation) and `Building` (triggers: construire, fabriquer, élever, construction, fabrication).

2.3. Comparison with other corpora

The comparison between the CALOR-Frame corpus and other corpora with semantic frame annotations is given in table 3. The column **Documents** shows the main sources of documents annotated, **# Sent** counts the number of sentences in each corpus; **% Sent. w/Frame** shows the percentage of sentences that have a Frame annotation; **Word Lexicon** displays the size of the lexicon of each corpus; **Frame lexicon**, **LU lexicon** and **FE lexicon** correspond to the number of frames, LUs and FEs considered in the annotation model; finally, **# Frame occurrences** shows the number of Frames annotated in the whole corpus.

As we can see, the CALOR-Frame corpus is the only corpus that is not oriented towards journalism, news or current events, it is also the corpus with the largest lexicon size. When we compare it to the existing dataset for Semantic Frame Parsing in English (SemEval07) and French (ASFALDA) (Candito et al., 2014) we observe that the CALOR corpus has the smallest frame lexicon, but the biggest number of annotations.

3. Frame parsing as a sequence labeling task

As mentioned in the introduction, semantic frame parsing is a structured prediction task where a word can belong to several structures. For example, in figure 1, the word `general` can belong to both the frame `Request` as the frame element `Speaker` and the frame `Deciding` as the `Cognizer`.

In order to consider the Frame parsing task as a *word sequence labeling* task, we need to flatten the frame structures by labeling each word with both semantic and information structure. In this study we have decided to use the simple `B,I,O` encoding for word segments where each word label starts with a `B` if it starts a segment, with an `I` if the word is inside a segment and `O` if it doesn't belong to any segment. Links between segments are represented by word indices.

Document Source	# Sentences	# Words	# Frames	# FE	Lexicon	% Sentence with Frame
WGM (Wikipedia WW1)	30994	686355	14227	32708	42635	34.2%
WA (Wikipedia Archeology)	27023	540653	9943	19892	41418	28.0%
CTGM (Cliotexte WW1)	3523	67736	938	1842	10844	21.4%
VKH (Vikidia Prehistory & Antiquity)	5841	85034	1617	3246	11649	21.9%
All	67381	1379778	26725	57688	72127	30.0%

Table 2: Description of the CALOR corpus

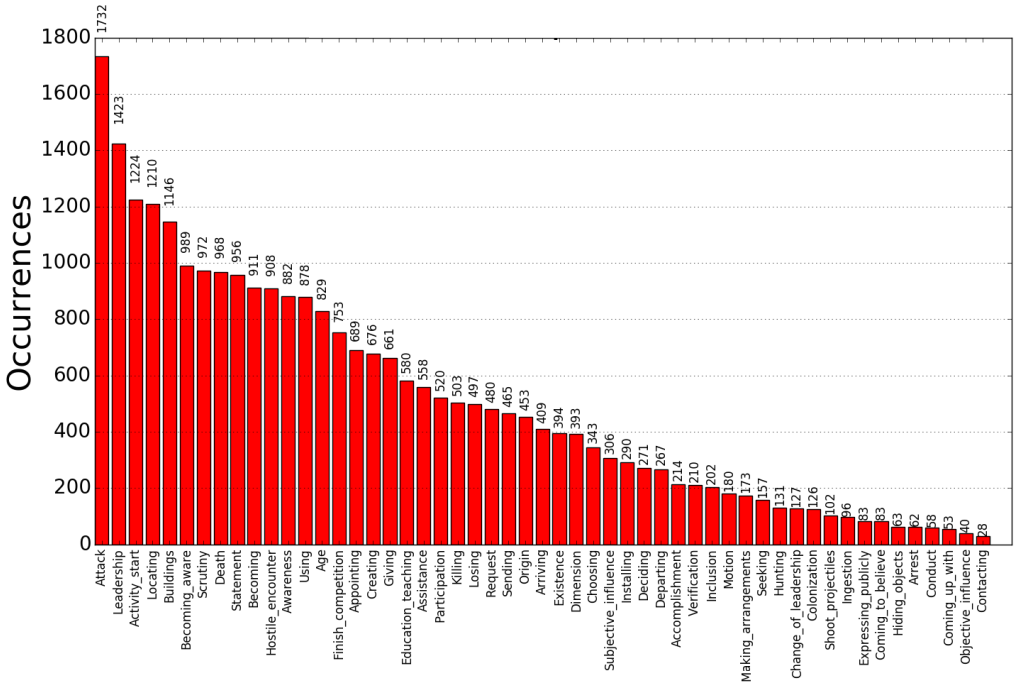


Figure 2: Distribution of the frame occurrences in the CALOR-Frame corpus

An example of such a representation for the sentence of Figure 1 is given in table 4. As we can see column 3 corresponds to the frame `Request` and column 4 to `Deciding`. The LUs triggering the frames are `order` at index 11 for the frame `Request` and `decide` at index 5 for `Deciding`. To each frame element is attached the index of the frame it belongs to through a link to the LU that triggered it.

Using such a representation for sequence labeling models like Conditional Random Fields (CRF) or Recurrent Neural Networks (RNN) models is challenging for two reasons:

1. Multi-labels: each word can receive more than one label according to the number of frames occurring in a sentence. Each label contains the frame as well as the frame element identifiers, therefore there are too many labels to consider building complex labels combining all of them.

2. Linking: an explicit link to the LU triggering a frame is added to each word label of its FEs, as can be seen in table 4. This information is necessary as several expressions of the same frame can occur in a sentence triggered by several LUs. The absolute value of these links is not meaningful and cannot be predicted the same way as the semantic labels are.

In this study we compare two different strategies in order to deal with the multi-label issue, one based on CRF with a multi-model approach (each LU has its own prediction model) and one based on a bi-LSTM model following a multi-task approach. They are described in the next section.

4. Sequence labeling models

4.1. Multi-model CRF approach

CRF-based approaches have been used in many NLP tasks involving sequence labeling such as POS tagging, chunking or named entity recognition (McCallum and Li, 2003). In order to apply CRF to frame parsing, as described in section 3., we need to address the multi-label issue. Since we want to perform frame disambiguation and semantic role detection in one step, and because each word in a sentence cannot trigger more than one frame, we chose a multi-model approach where a CRF-model is trained for each word belonging to the LU lexicon. This approach is described in Figure 3. At training time, the corpus is split according to the LU lexicon: to each word W_i belonging to this lexicon is attached a sub-corpus containing all the sentences C_{W_i} where W_i occurs. For each sentence $s \in C_{W_i}$, W_i can trigger a frame F among all the possible frames for this word in the LU lexicon, or nothing.

Corpus Name	Annotation	Document types	# Sent	% Sent w/ Frame	Word Lexicon	Frame lexicon	LU lexicon	FE lexicon	# Frame occurrences
SemEval07	English Frames	Journals	5946	**85.1%**	14150	**720**	**3.197**	**754**	23814
ASFALDA	French Frames	Journals (Le Monde)	21634	60.7%	33955	121	782	140	16167
CALOR	French Frames/SRL	Wikipedia Vikidia ClioTexte	**67283**	33.5%	**72127**	53	145	148	**26725**

Table 3: Semantic Frame corpus comparison

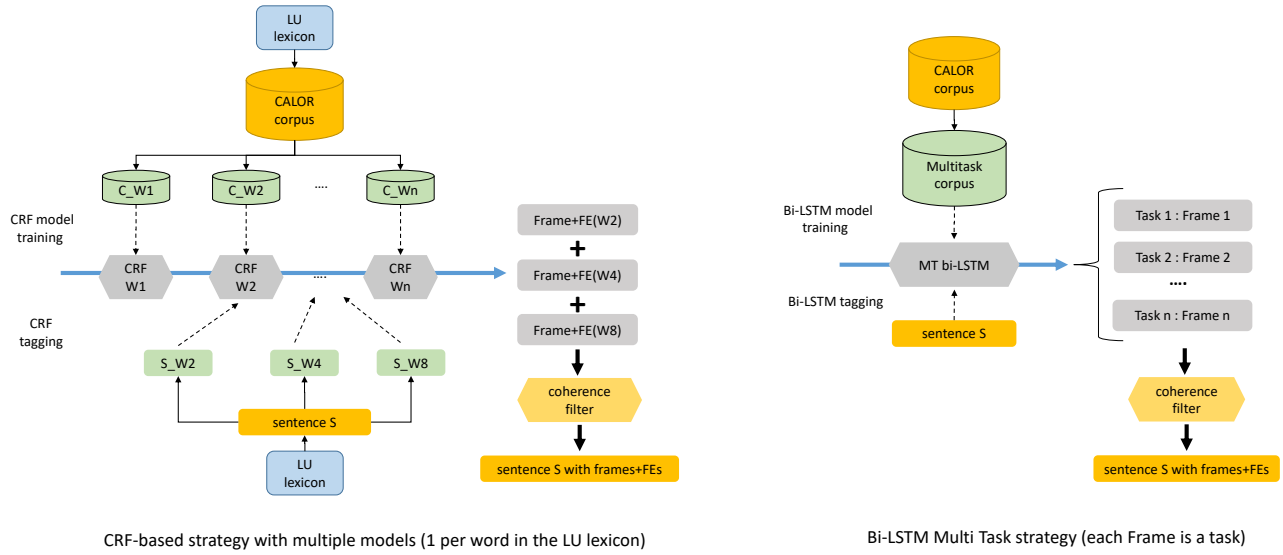


Figure 3: Two different strategies (CRF and bi-LSTM multitask) for Frame parsing

1	The	B:Req:Speaker:11	B:Dec:Cogn:5
2	general	I:Req:Speaker:11	I:Dec:Cogn:5
3	has	O	O
4	to	O	O
5	**decide**	O	**LU:Deciding**
6	if	O	B:Dec:Decis:5
7	it	O	I:Dec:Decis:5
8	is	O	I:Dec:Decis:5
9	necessary	O	I:Dec:Decis:5
10	to	O	I:Dec:Decis:5
11	**order**	**LU:Request**	I:Dec:Decis:5
12	the	B:Req:Addres:11	I:Dec:Decis:5
13	enemy	I:Req:Addres:11	I:Dec:Decis:5
14	the	B:Req:Message:11	I:Dec:Decis:5
15	immediate	I:Req:Message:11	I:Dec:Decis:5
16	surrender	I:Req:Message:11	I:Dec:Decis:5
17	of	I:Req:Message:11	I:Dec:Decis:5
18	Belfort	I:Req:Message:11	I:Dec:Decis:5

Table 4: Example of corpus with B,I,O format

For example, the sentence shown in table 4 will be duplicated into two sub-corpora, $C_{\texttt{order}}$ with column 3 and $C_{\texttt{decide}}$ with column 4. A CRF model is trained on each C_{w_i} sub-corpus.

At decoding time, when processing a sentence S, the same process is applied: first S is duplicated for each word w_i of S belonging to the LU lexicon. Then the CRF model corresponding to each w_i is applied and the different predictions made by the CRF models are merged.

This approach has the advantage of keeping the number of possible labels to predict for each CRF relatively small, limited to the frames that can be triggered by the word considered. Therefore the ambiguity is limited and CRFs can be trained efficiently even with a large number of features. However the drawback is that the training data is split across words in the LU lexicon, therefore similarities among LU are not exploited. This situation is acceptable if enough training examples are provided for each LUs, which is the case for the CALOR corpus.

4.2. Multi-task LSTM approach

Deep Neural Networks (DNN) with word embedding is the state of the art approach for semantic frame parsing (Hermann et al., 2014). More recently *recurrent neural networks* (RNN) with Long Short Memory (LSTM) cells have been applied to several semantic tagging tasks such as *slot filling* (Mesnil et al., 2015) or even frame parsing (Hakkani-Tür et al., 2016; Tafforeau et al., 2016) for Spoken Language Understanding.

Following these previous works, we propose in this study a single-layered bidirectional LSTM sequence to sequence architecture to perform frame tagging. To deal with the multi-label issue we could train a biLSTM model per LU, using the same approach as for the CRF, however, the number of examples per LU is reduced and neural networks do not perform well on small datasets. We would face the same problem if instead we decided to train one biLSTM per

frame. The remaining possibility is to build a single biLSTM to predict all possible frames in order to automatically learn a feature representation meaningful to all frames. We chose a multi-task approach (biLSTM-MT) similar to the one proposed by (Tafforeau et al., 2016), which models each frame as an isolated task. In this model two LSTM models, one forward and one backward are concatenated and shared among all tasks. Then a task-specific fully-connected output layer is added for each task.
In this work we consider each frame of our FrameNet model as a different task. This approach is described in figure 3. At decoding time each sentence is processed by the network and a distribution probability on the labels of each task for each word is produced. We only keep the labels above a certain threshold.

4.3. Coherence filter

Once the sequence tagging process is performed, each word is labeled with a frame element and position labels or a null label (O) as presented in table 4. Because the labels given at the word level might not be coherent at the frame level, we apply a *coherence filter* to the output of the tagging process. This filter is in charge of removing incoherences (FEs not starting with a B label; FEs without a frame) and linking the FEs to the LU that triggered the corresponding frame.
This filter implement is a very simple strategy: in a given sentence, if a word W of index i is labeled as a LU trigger for frame F, we link all the FEs detected in the sentence with the same frame label F to the LU w_i. At the end of this process, all FEs that have not been linked to a LU are removed.

4.4. Feature selection

The feature sets used for the CRF and the biLSTM approaches differ. For the CRF model, each training sample contains only one trigger word which is clearly identified, therefore we can use this information in order to add global constraints on the feature set of each word to process. On the opposite the multi-task biLSTM models can process several triggers in the same sentence, therefore the feature set cannot be biased toward a specific trigger and only local features are considered.
For the CRF models we consider 3 features: word lemma, part-of-speech (POS) and the syntax dependency path between the word to process and the potential frame trigger in the sentence. This dependency path is built through the concatenation of the syntactic functions between the word and the trigger. In the general case, a trigger is not necessarily at the root of the syntactic tree, for this reason, the dependency paths are composed of both links from child to parent (ascending links) and from parent to child (descending links) we make distinction of both types of links in the way we encode the dependency path.
For the biLSTM-MT we consider 4 features: word embeddings (Glove embeddings of 200 dimensions trained on French Wikipedia), POS, syntactic function (without a link) and a boolean indicator of whether the word belongs to the lexicon LU or not. All the features are encoded as trainable embeddings, we allow the network to adapt them during the training of the frame parsing task.
For both systems, in order to extract lemmas, POS and syntactic dependency trees we processed the frame annotated corpus using MACAON (Nasr et al., 2010) trained with a set of POS and dependencies similar to the one proposed in the Paris French TreeBank (Abeillé et al., 2003; Abeillé and Barrier, 2004).
The main differences between the multi-model CRF and multi-task biLSTM models are summarized in table 5.

5. Evaluation

5.1. Experimental setup

When a sentence is processed, there are 4 steps or subtasks that take place either explicitly or implicitly in the frame parsing process. Even though our approaches perform frame detection and semantic role labeling in one step, we detail the scores for each sub-task because they are relevant indicators of the performance of a parser, and they serve as point of comparison between our models. These 4 sub-task are:

1. *trigger identification* (TI) which decides whether a word in a sentence can trigger a frame or not;
2. *trigger classification* (TC) which assigns the frame label to the trigger word detected to form a LU;
3. *role filler identification* (RI) which detects potential semantic role fillers in the sentence for the frame detected;
4. *role filler classification* (RC) which assigns a label to each role filler detected in order to obtain the *Frame Elements* (FE) of the frame detected.

In this study we consider these 4 tasks as a cascade process: an error in task 1 will lead to several errors in task 4 since no correct FEs will be detected if the frame is not triggered. The CALOR corpus is annotated with a small number of frames compared to the FrameNet 1.5 corpus, therefore the ambiguity for subtasks 1 and 2 is rather small. The most complex task is of course task 4, *role filler classification*, since every detection and classification of LUs and FEs has to be correct. That is why we will pay a special attention to this task to compare the models in the following experiments.
To carry out our experiments we divide the corpus CALOR assigning 80% of the frame occurrences to the train set and 20% to the test set. This split is done in such a way that the frame distribution remains as similar as possible between train and test while considering a document as an atomic unit that cannot be subdivided and should be either in the train or in the test set. This split does not take into account the LU distribution, a LU can therefore appear both in train and test, only in train or only in test.
Similarly to previous work on semantic frame parsing we will use *precision*, *recall* and *f-measure* on the 4 subtasks presented in order to evaluate our approaches. We also compute *precision/recall* curves by using different acceptation thresholds on the frame and semantic role hypotheses output by our models. In this study we set the operating point for comparing our models to the *Equal Error Rate* (EER) between the precision and recall measures.

	CRF-LU	biLSTM-MT
Lexical Feat	Lemma	Glove Word Embeddings
Morpho-Syntactic Feat	POS	POS Embeddings
Syntactic Feat	Dep. Path to Trigger	Syntactic Function Embeddings
Model	145 CRF (1 CRF per LU)	1 biLSTM 54 Taks (1 Task per Frame)
Scaling	Many LU	Many Taks
Labeling Process	1 Application per LU per Phrase	Parse all the Frames in 1 Application
Information Sharing	No Sharing	Implicit Sharing Between Frames

Table 5: Comparative Overview of CRF-LU and biLSTM-MT models

5.2. Overall results

The comparison between the multi-model CRF (CRF-MM) and multi-task biLSTM (biLSTM-MT) performance over the set of 4 sub-tasks presented in the previous section is presented in Table 6. Both systems use their full feature set as presented in table 5 with EER between precision and recall as the chosen operating point. As expected performance for subtasks 1 and 2 are very good, much better than those reported on the FrameNet 1.5 corpus (Hermann et al., 2014). This is due to the *partial parsing* approach used here where only a subset of the FrameNet model is used.

CRF-MM model performs better on the trigger identification and classification tasks (subtasks 1 and 2) while the biLSTM-MT model is better on role identification and classification (subtasks 3 and 4). The reason for this behavior is that it is easier for a CRF model to identify the proper frame for each trigger word, as they can only trigger a few frames, while for the biLSTM-MT all frames are in competition. This reduction of ambiguity for CRF-MM models has a cost: splitting the training corpus according to the LU lexicon and therefore reducing the training data for each model. If this is not an issue for the two low ambiguity subtasks 1 and 2, the situation is different for subtasks 3 and 4. Indeed biLSTM-MT is a single model that learns from all training data, it is able to automatically learn relevant features and capture semantic aspects of text using a shared layer and then use these features to classify tokens in roles of different frames learned one by tasks. This ability to use the whole corpus leads the biLSTM-MT model to outperform CRF-MM for tasks 3 and 4.

This result is confirmed by figure 4 which displays the precision/recall curves of both methods on subtask 4. As we can see, biLSTM-MT is better in terms of maximal F-measure. It is interesting to observe that both models do not show the same precision-recall trade-off. The CRF-MM is able to parse text at a high precision with a low recall, this is not the case of biLSTM-MT, that handles a much larger set of frames and is more prone to precision errors. On the other hand, biLSTM-MT achieves a better recall at a fairly good precision, this happens because it is trained on the full dataset and it is able to learn syntactic patterns from different frames and extrapolate them from one frame to the other.

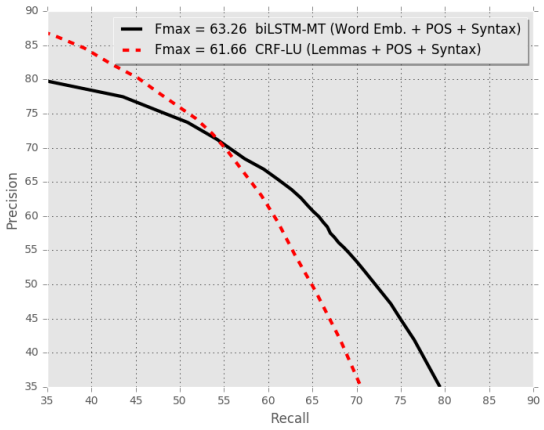


Figure 4: Precision Recall curves for CRL-LU and biLSTM-MT on the full feature set

	Fmeasure CRF-MM	Fmeasure biLSTM-MT
Trigger Ident. (TI)	96.4	95.5
Trigger Classif. (TC)	95.2	92.3
Role Ident. (RI)	69.7	70.3
Role Classif. (RC)	60.6	63.2

Table 6: Comparison of our two models on their full feature set operating at EER

6. Conclusion

In this paper we introduced the CALOR corpus as well as a comparison between two different semantic frame parsing models that consider the task as a sequence labeling task. The main contribution of this work is to propose a new corpus, publicly available, that contain semantic frame annotations that differ from previous corpus such as FrameNet and SemEval. In our case only *partial* annotation is considered, allowing to annotate much larger corpora at a lower cost than full text annotation. Only a small subset of the FrameNet lexicon is used, however the amount of data annotated for each frame is much larger than in other corpora, allowing to develop and test different parsing methods.

In this study two frame parsing models are compared, one based on CRF following a multi-model approach and other based on biLSTM with a multi-task approach. Experiments show that biLSTM-MT model achieves a better recall, while CRF-MM achieves better precision, this is due to the architecture of each model, in CRF-MM we divide frame parsing into small subtasks one per LU, reducing the

number of possible labels in each decision, thus augmenting the precision. On the other hand biLSTM-MT is able to share data across LUs boosting its capacity to deal with complex syntactic patterns and being able to retrieve more frame elements during parsing.

7. Bibliographical References

Abeillé, A. and Barrier, N. (2004). Enriching a french treebank. In *LREC*.

Abeillé, A., Clément, L., and Toussenel, F. (2003). Building a treebank for french. *Treebanks*, pages 165–187.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

Candito, M., Amsili, P., Barque, L., Benamara, F., Chalendar, G., Djemaa, M., Haas, P., Huyghe, R., Mathieu, Y., Muller, P., Sagot, B., and Vieu, L. (2014). Developing a french framenet: Methodology and first results. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*.

Hakkani-Tür, D., Tur, G., Celikyilmaz, A., Chen, Y.-N., Gao, J., Deng, L., and Wang, Y.-Y. (2016). Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *Proceedings of The 17th Annual Meeting of the International Speech Communication Association*.

Hermann, K. M., Das, D., Weston, J., and Ganchev, K. (2014). Semantic frame identification with distributed word representations. In *ACL (1)*, pages 1448–1458.

McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 188–191, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 23(3):530–539.

Nasr, A., Bechet, F., and Rey, J.-F. (2010). Macaon : Une chaine linguistique pour le traitement de graphes de mots. In *Traitement Automatique des Langues Naturelles - session de demonstrations*, Montreal.

Nasr, A., Béchet, F., Rey, J.-F., Favre, B., and Le Roux, J. (2011). Macaon: An nlp tool suite for processing word lattices. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations*, HLT '11, pages 86–91, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tafforeau, J., Bechet, F., Artiere, T., and Favre, B. (2016). Joint syntactic and semantic analysis with a multitask deep learning framework for spoken language understanding. *Interspeech 2016*, pages 3260–3264.

Using a Corpus of English and Chinese Political Speeches for Metaphor Analysis

Kathleen Ahrens[1], **Huiheng Zeng**[2], **Shun-han Rebekah Wong**[3]
[1,2]Department of English, The Hong Kong Polytechnic University
11 Yuk Choi Road, Hung Hom, Kowloon, Hong Kong
[3]HKBU Library, Hong Kong Baptist University
Kowloon Tong, Kowloon, Hong Kong
kathleen.ahrens@polyu.edu.hk, winnie.zeng@connect.polyu.hk, rebekahw@hkbu.edu.hk

Abstract

In this article, we present details of our corpus of political speeches and introduce using the corpus for metaphor analysis in political discourse. Although specialized corpora on a variety of topics are now easily available, online political corpora available for public use are scarce. The database our research team has developed contains more than six million English and Chinese political speeches and is currently available free online. Researchers in many fields are able to use the multiple search functions on the website for their specific research purposes. In particular, the corpus is useful for researchers focusing on political speeches and conceptual metaphor analyses. From the perspective of metaphor study, we have taken advantage of several functions to facilitate the corpus-based metaphor analyses. In short, this database enriches the current bilingual resources and contributes to the evaluation of political language by linguists and political scientists.

Keywords: corpus, political speeches, metaphor

1. Introduction

The rising prominence of corpus-driven approaches in linguistic studies has sparked a growing interest in applying a corpus-based approach to metaphor analysis in the political domain (Charteris-Black, 2004, 2009; Ahrens, 2009; Deignan, 1999, 2005; Semino, 2006, 2008; Deignan and Semino, 2010). The corpora used in these studies are usually compiled manually by authors, i.e. Charteris-Black (2009) collected data of British parliamentary debates from online versions of Hansard, while Koller and Semino (2009) assembled a corpus of interviews and speeches by Germany chancellors from the official websites of the German government. However, these corpora have not been released and made available for public search or free use. Our project proposes the creation of a corpus entitled 'The HKBU Corpus of Political Speeches' which is a large-scale political database with English and Chinese data and featured search functions; it is convenient for corpus-based research and also available for online free use.

Two major design features of the HKBU Corpus of Political Speeches are as follows: 1. The minimal interface of the database website is designed for users to operate easily. Similar to Sinica Corpus 4.0 (Chen et al., 1996), the search functions of our database such as the keyword-in-context (KWIC) search and collocation search can retrieve the lexical frequency and collocation lists which facilitate a corpus-based approach to linguistic analyses. Our database is especially useful for metaphor analysts who are able to search keywords in the source domains and the target domains of a particular metaphor and extract all the data involving the keywords searched into Excel files for further analyses. In addition, the collocation search and left one sort or right one sort functions can contribute to the determination of specific source domains or target domains of the keywords. Examples of studies that have used corpora to aid in metaphor analyses are Ahrens and Chang (forthcoming) who studied the role of lexical frequency and collocation in postulating conceptual models and Ahrens, Chung and Huang (2004) and Chung, Ahrens and Huang (2004a, b) who studied the source domain determination and verified the Mapping Principles in conceptual metaphors.

The corpus itself also contains bilingual versions of political speeches for Hong Kong Policy Addresses which provides a valuable resource for researchers focusing on translation. It also contains English and Chinese (traditional and simplified) political corpora with various speech types (i.e. presidential speeches, political debates, policy addresses, reports of government work). The speeches are delivered by politicians in different political positions (i.e. Presidents, Governors, Chief Executives and Premiers) and are from People's Republic of China, Hong Kong, Taiwan and the United States.

The corpus presented in this paper provides an intuitive way for political analysts, linguists and general users to access and use the political corpora as it provides for a simple interface that allows for in-depth and fine-grained analysis of these texts via KWIC search and collocation. In addition, we hope that the analysis of conceptual metaphors in English and Chinese will also allow a greater understanding of the rhetorical patterns in political speeches so as to understand the ideologies of politicians in areas from China, Hong Kong, Taiwan and the United States. It will also provide insight into how linguistic patterns drawn from corpora help to understand and build metaphor models. In the following sections, we will introduce our database in detail and present how it is applied in facilitating metaphor research.

2. Corpus of Political Speeches

2.1 Dataset creation

The HKBU Corpus of Political Speech (http://digital.lib.hkbu.edu.hk/corpus/) is an online archive of political speeches developed by our research team together with the Hong Kong Baptist University library. In total, 6,269,359 words of political speeches delivered by politicians from China, Hong Kong, Taiwan and the United States were collected for this online database. It currently contains four collections of speeches: the English Corpus

of U.S. Presidential Speeches (1789-2015), including six different types of sub-corpora; the English & Chinese Corpus of Policy Addresses by Hong Kong Governors (1984-1996) and Hong Kong Chief Executives (1997-2014); the Chinese Corpus of Speeches given on New Year's days and Double Tenth days by Taiwan Presidents (1978-2014) and the Chinese Corpus of Report on the Work of the Government by Premiers of the People's Republic of China (1984-2013). Details of the four collections are listed in Table 1.

US Corpus of Political Speeches (4,429,976 words)
Inaugural Addresses (1789-2013) – *in English*
Annual Messages to Congress on the State of the Union (1790-2014) – *in English*
National Political Party Platforms (1944-2012) – *in English*
Presidential Nomination Acceptance Speeches (1960-2012) – *in English*
Presidential Candidates Debates (1960-2012) – *in English*
Saturday Radio Addresses (1982-2014) – *in English*
HK Corpus of Political Speeches (1,079,712 words)
Policy Address by HK Governors (1984-1996) – *in English*
Policy Address by HK Chief Executives (1997-2014) – *in English*
Policy Address by HK Governors (1984-1996) – *in Chinese*
Policy Address by HK Chief Executives (1997-2014) – *in Chinese*
Taiwan Corpus of Political Speeches (169,649 words)
Speeches Given on New Year's days and Double Tenth days by Presidents in Chinese (1978-2014) – *in Chinese*
P.R.C. Corpus of Political Speeches (590,022 words)
Report on the Work of the Government by P.R.C. Premiers in Chinese (1984-2013) – *in Chinese*

Table 1: Corpus of English & Chinese Political Speeches

2.2 Chinese corpora annotation

For the Chinese Corpora of Hong Kong, Taiwan and PRC political speeches, we further annotated each corpus with part-of-speech tagging (Figure 1) by using The Stanford Natural Language Processing Software (SNLPG, 2015), Stanford Word Segmenter 3.7 for Chinese words segmentation and Stanford POS Tagger 3.7 for part-of-speech tagging.



Figure 1. POS tagging in Chinese corpus

To improve the tagging accuracy, two researchers in our team who are both native Chinese speakers with a linguistic background took six months to further check all the tagged texts word by word based on Stanford segmentation (Fei, 2000a) and part-of-speech tagging (Fei, 2000b). We found that the automatic word segmentation and POS tagging on the simplified Chinese is more accurate than the traditional Chinese. The accuracy for both word segmentation and POS tagging in the PRC corpus with simplified Chinese characters was approximately 85%. It was approximately 60% in Hong Kong and Taiwan Corpora which both use traditional Chinese characters. After two linguistically-trained native speakers of Chinese manually checked the corpus, the problematic taggings were revised and the tagged Chinese corpora provide a reliable Chinese database with a wide range of syntactically tagged texts. This allows researchers to conduct analyses of the syntactic structures, grammatical features and syntactic distribution in political discourse. It is especially useful to metaphor researchers focusing on Chinese political discourse. For example, the following sentence has been extracted from the database.

Example

高等/JJ 教育/NN 体制/NN 改革/NN 和/CC 结构/NN 调整/NN 的/DEG 步伐/NN 加快/VV 。/PU (朱镕基, PRC, 1999)
Gao1deng3/JJ jiao4yu4/NN ti3zhi4/NN gai3ge2/NN he2/CC jie2gou4/NN tiao2zheng3/NN de0/DEC bu4fa2/NN jia1kuai4/VV 。/PU (Zhu1 Rong2ji1, PRC, 1999)
Higher/JJ_education/NN_system/NN_reform/NN_and/CC _structure/NN_adjustment/NN_of/DEG_pace/NN_speed up/VV ./PU (Zhu Rongji, PRC, 1999)
'The pace of the high education system reform and structural adjustment has been speeded up. (Zhu Rongji, PRC, 1999)'

The sentence has been segmented into separate lexical units which is usually considered to be the first step of the metaphor identification procedure (Pragglejaz Group, 2007; Steen et al., 2010). Metaphor analysts can work directly on the segmented sentence to check the basic/contextual meanings of each lexical unit in order to locate keywords that have been metaphorically used. In this example, three words 结 构 /NN *jie2gou4*/NN 'structure', 调整/NN *tiao2zheng3*/NN 'adjust' and 步伐 /NN *bu4fa2*/NN 'pace' are identified as the metaphorical keywords. In addition, the Chinese POS tagging provides syntactic structure information contributing to the determination of word senses and collocations.

2.3 Features and functions of the database

The corpus has a web-based concordance feature, which performs a number of functions for corpus searches in untagged texts (for English corpora) and part-of-speech tagged texts (for Chinese corpora). Firstly, the KWIC searches are for the number of occurrence, graphical representation and the full-text request available (Figure 2).



Figure 2. No. of Occurrence

The searches are able to be further refined by choosing different sources, speakers of the speeches and time frames (Figure 3). Collocation search is available for the top 20 words searches and the search can be specified by choosing different positions of the words (up to 10) before or after the keywords (Figure 4).

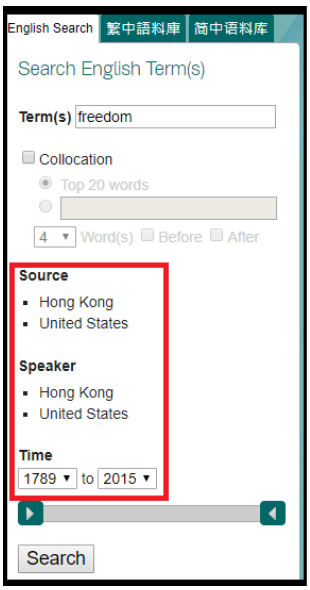


Figure 3. Search filters

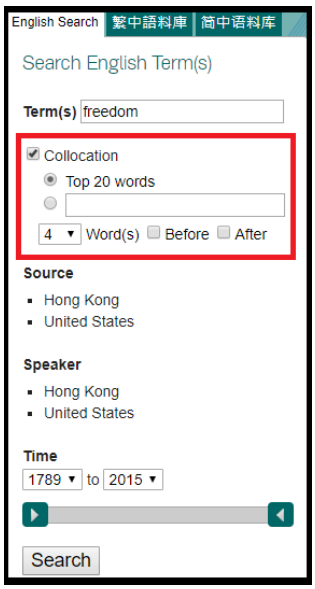


Figure 4. Collocation search

For example, if we search the collocations of 'freedom' (one word before 'freedom') in the Hong Kong Pre-1997 corpus, we will get the number of occurrence and the graphical representation of all the collocates (Figure 5).

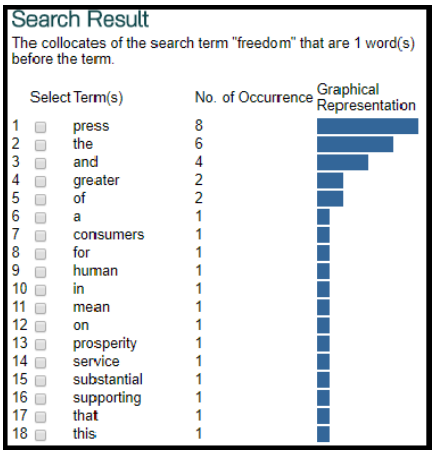


Figure 5. Collocation search results

Then if we select the most frequent collocate 'press', all the sentences involving 'press' will be retrieved and presented. The search results can be further sorted by positions of the search words (from L5 to R5), years, regions, types and speakers of target speeches (Figure 6).



Figure 6. Search filters

In addition, the system allows users to export all selected sentences into Excel files for further analysis (Figure 7), which is especially useful to metaphor researchers who need to look at surrounding context to determine of a word or phrase is being used literally or metaphorically.

1992	HK	HK - GA	Patten	... Ordinance and for a review of current laws which might have a bearing on press	freedom.	I understand these important concerns. In a free society, they merit serious
1992	HK	HK - GA	Patten	concerns. In a free society, they merit serious attention. We cannot take press	freedom	for granted. But I should make clear at the outset my serious doubts about the ...
1992	HK	HK - GA	Patten	... be done to extend access to information and to review laws which might infringe press	freedom.	A review has been under way since August. I will ask the Executive Council to ...
1993	HK	HK - GA	Patten	for human rights issues and for related policy areas such as press	freedom	This arrangement will make it easier for the Government to
1993	HK	HK - GA	Patten	... in the Estimates as far as practicable. (b) To review laws which might infringe press	freedom	and look again at what might be done to extend access to official information.
1993	HK	HK - GA	Patten	information. The review was completed in march 1993. It concluded that press	freedom	Kong law but some provisions might either be obsolete or at ...
1994	HK	HK - GA	Patten	therefore continue our programme to remove the legal restrictions on press	freedom.	separate provisions in 27 Ordinances. Of these, we will have taken ...
1988	HK	HK - GA	Wilson	Nevertheless section 27 was seen by some as placing excessive restrictions on press	freedom,	for dealing with 'false news'. In January, the Government gave an undertaking ...

Figure 7. Extracted sentences in Excel files

Links to the original official sources have been provided for users who want to read the full-text. Detailed instructions of how to use the search functions for both Chinese corpus and English corpus can be found in the User Manual at http://digital.lib.hkbu.edu.hk/corpus/help.php.

3. Corpora-based Evaluation of Political Metaphor

Metaphor research examines what concrete concepts in one conceptual domain are mapped to abstract concepts in another domain. Copora-based conceptual metaphor research has shown that corpora-based analyses may provide linguistic evidence of metaphor variations by comparing metaphor use within the same source or target domains (Charteris-Black, 2009; Ahrens, 2009, 2011; Lu and Ahrens, 2008; Ahrens and Lee, 2009; Semino and Koller, 2009). We believe that a cross-linguistic analysis of

conceptual metaphor based on abundant corpus data will allow researcher to evaluate the degree to which metaphors occur cross-linguistically and provide linguistic evidence of the mapping principles between the source and the target domains (Ahrens, 2010). These types of studies will contribute to the investigation of existing metaphor theories.

For example, using corpora from our English database (the State of Union Addresses and Presidential Radio Addresses of US Presidents 1981-2006), Ahrens, (2011) has investigated the lexical frequency patterns related to two metaphor models (i.e. the Strict Father model and the Nurturant Patent model) and found support for Lakoff's (1996, 2002) hypothesis that Democrats and Republicans view the world differently. Evidence from the Reagan and Clinton data show that they used metaphors based on value paradigms in two different metaphor models, respectively. The findings in that paper led to a better understanding of the underlying conceptual worldview of the political leaders in the Democratic and Republican parties and pointed out the limitations of a purely intuition-based approach.

In addition, metaphor researchers may take advantage of the search functions in the Corpus of Political Speeches to evaluate metaphor in different corpora at multiple levels (i.e. analysis at cross-regional, cross-linguistically, cross-speakers levels and from diachronic angles). Since the data in our corpus can be searched for within a particular time period, this enables analysis of diachronic language change and the top lexemes for each speaker from different regions allows the comparison of lexical usages between different Chinese corpora or between Chinese and English corpora. Frequency listed word counts provide additional linguistic information for more in-depth and finer grained analysis of lexical and metaphor use. We apply approaches of top-down analysis to first search all the keywords in our database and extract target texts to Excel files in order to examine the selected keywords. We then normalize the frequency of metaphorical keywords within a particular source or target domain for the comparison of metaphor usage and their underlying political ideologies.

For example, Lu and Ahrens (2008) found that Kuomintang Presidents in Taiwan used BUILDING metaphors to instill a Chinese ideology. However, the president from the Democratic Progressive Party preferred not to use BUILDING metaphors, and instead used FARMLAND metaphors to emphasize Taiwan's agricultural background and political independence. In addition, Kuomintang Presidents used BUILDING metaphors in ways that differ from US Presidents. The Kuomintang Presidents used retrospective BUILDING metaphors to emphasize the past history of China, while US Presidents used BUILDING metaphors to emphasize creating a particular type of structure (i.e., economic, educational and political) for future generations. In Hong Kong English corpus, metaphor analysis of the target domains with the source domains of JOURNEY and BUILDING showed that the Hong Kong Chief Executives tend to use more JOURNEY than BUILDING metaphors and the target domain refers primarily to Hong Kong's future. In addition, when Chief Executives talk about Hong Kong's future, the future often collocates with positive evaluations (Ahrens, 2016a, b).

Our on-going metaphor analyses (Ahrens and Zeng, 2017a, b) focus on the comparison of source domain types related to the target domains of EDUCATION and DEMOCRACY in both Chinese corpus (HK, PRC and TW corpus) and English corpus (HK corpus). Figure 8 presents examples of Chinese EDUCATION metaphor analyses in PRC corpus. Column A indicates the time frames of the speeches; column B shows the regions of the speeches delivered; the names of the speakers are indicated in column C; column D and F are the speech texts extracted; column E indicates the target domain keyword 教育 *jiao4yu4* 'education'; in column G, we input the source domains manually identified and in column H we input the metaphorical keywords that we used to determine the source domains listed in column G. We hope that in the future these manual analyses will provide a full set of data that can be used to aid future metaphor research.

A	B	C	D	E	F	G	H
Time	Regio	Speak	Concordance text	KWs	Concordance text	Source	M4 KW
1989	PRC - PR-Pre 99	Li Peng	……。/PU 注意/VV 把/BA 社会/NN 力量/NN 办学/VV 的/DEC 积极性/NN 引导到/VV 支持/VV 发展/VV 基础/JJ	教育/NN	和/CC 职业/NN 技术/NN 教育/NN 的/DEC 方向/NN 上/LC 来/SP，/PU 要/VV 进一步/AD 活跃/VV ……	BUILDING; JOURNEY	支持; 方向
1989	PRC - PR-Pre 99	Li Peng	…… 空气/NN ，/PU 加强/VV 教育/NN 科学/NN 和/CC 理论/NN 的/DEG 研究/NN ，/PU 积极/AD 探索/VV	教育/NN	思想/NN 、/PU 教育/NN 内容/NN 和/CC 教育/NN 方法/NN 的/DEG 改革/NN ，/PU 不断/AD 提高/VV 教育/NN 质量/NN 。/PU ……	JOURNEY; SYSTEM	探索; 改革
2013	PRC - PR - Post 99	Wen Jiabao	…… 重点/NN 向/P 农村/NN 、/PU 边远/JJ 、/PU 民族/NN 、/PU 贫困/JJ 地区/NN 倾斜/VV ，/PU	教育/NN	公平/NN 取得/VV 明显/JJ 进步/NN 。/PU 全面/AD 实现/VV 城/NN 乡/NN 九/CD 年/M 免费/JJ ……	JOURNEY	进步
1990	PRC - PR-Pre 99	Li Peng	…… 工作/NN 。/PU 继续/AD 抓好/VV 基础/JJ 教育/NN ，/PU 稳步/AD 推进/VV 九/CD 年/M 义务/NN	教育/NN	，/PU 并/CC 采取/VV 积极/JJ 措施/NN ，/PU 制止/VV 中/JJ 小/JJ 学生/NN 辍学/VV 。/PU ……	JOURNEY; OBJECT	稳步; 推进
2001	PRC - PR - Post 99	Zhu Rongji	…… 对/P 科技/NN 的/DEG 投入/NN 。/PU 加强/VV 国家/NN 重点/JJ 实验室/NN 建设/NN 。/PU 坚持/VV	教育/NN	适度/AD 超前/AD 发展/VV ，/PU 为/P 国民/NN 经济/NN 和/CC 社会/NN 发展/NN 服务/VV 。/PU ……	JOURNEY	超前

Figure 8. Examples of EDUCATION metaphor analyses

Figure 9 illustrates one example of the normalized ratio results related to the comparisons of the source domains (PRODUCT & BUILDING) of EDUCATION metaphors between Hong Kong Chief Executives and PRC Premiers. The results show that Hong Kong Chief Executives conceptualize education as PRODUCT more frequently than PRC Premiers while both groups use the concept of BUILDING with similar frequencies. Within Hong Kong Chief Executives, Tung Chee-hwa focuses more on the concept of education as PRODUCT and Donald Tsang uses more BUILDING metaphors. PRC premiers understand education more as BUILDING than PRODUCT, among whom Li Peng and Wen Jiabao apply the two source domains more frequently than Zhao Ziyang and Zhu Rongji.

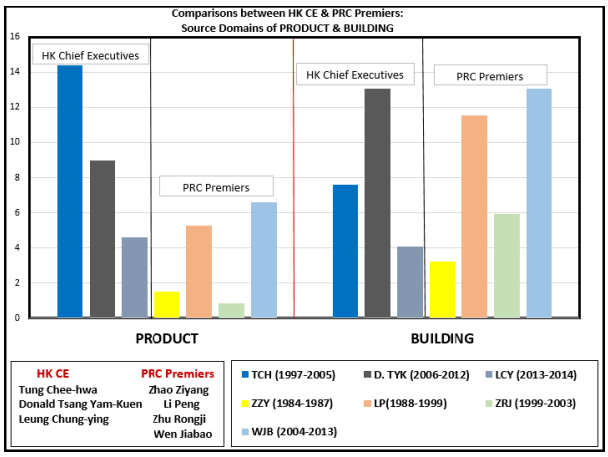


Figure 9. Comparisons of source domains between Hong Kong Chief Executives and PRC Premiers

Evaluating metaphor source domain variations in the Chinese corpora in our database will also allow for contrastive analysis of Chinese languages over the past three decades, which will lead to a deeper understanding of the degree of universality that languages share with the same cultural backgrounds and the degree of difference that languages reflect in different political ideologies. All these will allow greater understanding of the rhetorical patterns of persuasion in political speeches across time.

4. Conclusion and Future Work

In this paper, we presented the HKBU Corpus of Political Speeches and described the information about the dataset as well as about the Chinese data annotation and the featured search functions of the database. We also introduced corpus-based metaphor studies in political discourse by using the search functions and corpora from this database.

For the further development of the corpus, we hope to: 1) enlarge our corpus by continuing to update the current corpora and adding new speeches to compile new corpora, such as to include political speeches by Hong Kong senior officials, PRC chairmen or speeches by politicians from Macau; 2) use Stanford part-of-speech tagger to annotate all the English corpora with POS tagging in order to align with the POS tagged Chinese corpora; 3) try to build a small corpus with tagged metaphors.

5. Acknowledgements

This research was supported in part by the following grants: General Research Fund (#12400014) from the Hong Kong University Grants Council to the first author, HKBU Faculty Research Grants (#FRG2/08/09/027, #FRG2/10/11/049, #FRG2/11/12/021 and #FRG1/16-17/005) to the first author, and the HKBU Digital Scholarship Grant (#DSG/1516/01/LC) to the first author (PI) and the third author (Co-I). All authors would like to thank the research team in HKBU Library and the research assistants involved in the above projects.

6. Bibliographical References

Ahrens, K. (Ed.). (2009). *Politics, gender and conceptual metaphors*. Basingstoke and New York: Palgrave-MacMillan.

Ahrens, K. (2010). Mapping principles for conceptual metaphors. In L. Cameron, A. Deignan, G. Low & Z. Todd (Eds.), *Researching and applying metaphor in the real world*). Amsterdam: John Benjamins, pp. 185-207.

Ahrens, K. (2011). Examining conceptual metaphor models through lexical frequency patterns: A case study of U.S. presidential speeches. In H. J. Schmid (Ed.), *Windows to the mind. Series: Applications of cognitive linguistics.* Berlin: Mouton De Gruyter, pp. 167-184.

Ahrens, K. (2016a). English conceptualizations in a Chinese context: BUILDING metaphors in Hong Kong policy addresses. Presented at International Conference on Boundary Crossing and Bridge Building. National Taiwan University, Taipei, Taiwan, 17 June, 2016.

Ahrens, K. (2016b). Contrasting BUILDING metaphors in the Corpus of Political Speeches. Presented at ICAME (International Computer Archive of Modern & Medieval English) 37. Chinese University of Hong Kong, Hong Kong, 25-29 May 2016.

Ahrens, K. and Chang, P. Y. C. (Forthcoming). A Corpus-based approach to conceptual metaphor modeling: The role of lexical frequency and collocation. In C. R. Huang, S. K. Hsieh, & Z. F. Sui (Eds), *Chinese Language Resources: Data collection, linguistic analysis, annotation, and language Processing.* Springer.

Ahrens, K. and Lee, S. (2009). Gender versus politics: When conceptual models collide in the U.S. Senate. In K. Ahrens (Ed.), *Politics, gender, and conceptual metaphors.* Basingstoke and New York: Palgrave-MacMillan, pp. 62-82.

Ahrens, K. and Zeng, H. H. (2017a). Metaphors of Education in China and Hong Kong. Presented at Look Both Ways: Narrative & Metaphor in Education - Conference 2017. Vrije University, Amsterdam, 30 March - 1 April 2017.

Ahrens, K. and Zeng, H. H. (2017b). The Conceptualization of Democracy in Hong Kong Policy Addresses. Presented at The third VEIP (Varieties of English in the Indo-Pacific) meeting. Linguistics Macquarie University, Australia, 16-17 February 2017.

Ahrens, K., Chung, S. F., and Huang, C. R. (2004). From lexical semantics to conceptual metaphors: Mapping principle verification with WordNet and SUMO. In Ji Donghong, Lua Kim Teng, and Wang Hui, editors, Recent Advancement in Chinese Lexical Semantics. Proceedings of 5th Chinese Lexical Semantics Workshop (CLSW-5), pages 99-106. Singapore: COLIPS.

Charteris-Black, J. (2004). Corpus approaches to critical metaphor analysis. Basingstoke: Palgrave Macmillan.

Charteris-Black, J. (2009). Metaphor and gender in British parliamentary debates. In A. Kathleen (Ed.), *Politics, gender and conceptual metaphors*. Basingstoke and New York: Palgrave-MacMillan, pp. 139-165.

Chen, K. J., Huang, C. R., Chang, L. P., and Hsu, H. L. (1996). SINICA CORPUS: Design methodology for balanced corpora. In Byung-Soo Park and Jong-Bok, editors, Proceedings of Language, Information and Computation (PACLIC 11), pages 167-176. Seoul, Korea.

Chung, S. F., Ahrens, K., and Huang, C. R. (2004a). Using WordNet and SUMO to determine source domains of conceptual metaphors. In Ji Donghong, Lua Kim Teng, and Wang Hui, editors, Recent Advancement in Chinese Lexical Semantics. Proceedings of 5th Chinese Lexical Semantics Workshop (CLSW-5), pages 91-98. Singapore: COLIPS.

Chung, S. F., Ahrens, K., and Huang, C. R. (2004b). RECESSION: Defining source domains through WordNet and SUMO. Proceedings of the Linguistic Society of Korea (LSK) International Conference. Volume II: General Sessions, pages 43-52. The Linguistic Society of Korea (LSK) and Yonsei Institute of Language and Information Science (ILIS). Seoul.

Deignan, A. (1999). Corpus-based research into metaphor. In A. Deigan, G. Low & Z. Todd (Eds.), *Researching and applying metaphor in the real world*. Cambridge: Cambridge University Press, pp. 177-200.

Deignan, A. (2005). *Metaphor and corpus linguistics (Vol. 6)*. Amsterdam: John Benjamins Publishing.

Deignan, A. & Semino, E. (2010) Corpus techniques for metaphor analysis. In: L. Cameron & R. Maslen (Eds.), *Metaphor analysis: Research practice in applied*

linguistics, social sciences and the humanities. London: Equinox, pp. 161-179.
Fei, X. (2000a). The segmentation guidelines for the Penn Chinese Treebank (3.0). *IRCS Technical Reports Series*. Retrieved from: http://repository.upenn.edu/cgi/viewcontent.cgi?article=1038&context=ircs_reports
Fei, X. (2000b). The part-of-speech tagging guidelines for the Penn Chinese Treebank (3.0). *IRCS Technical Reports Series*. Retrieved from: http://repository.upenn.edu/cgi/viewcontent.cgi?article=1039&context=ircs_reports
Koller, V., and Semino, E. (2009). Metaphor, politics and gender: a case study from Germany. In A. Kathleen (Ed.), *Politics, gender, and conceptual metaphors*. Basingstoke: Palgrave Macmillan, pp. 9-35.
Lakoff, G. (1996). *Moral politics: What conservatives know that liberals don't*. Chicago: University of Chicago Press.
Lakoff, G. (2002). *Moral politics: How liberals and conservatives think*. Chicago: The University of Chicago Press.
Lu, L. W. L. and Ahrens, K. (2008). Ideological influences on BUILDING metaphors in Taiwanese presidential speeches. *Discourse and Society, 19*(3): 383-408.
Pragglejaz Group. (2007). MIP: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol, 22*(1): 1-39.
Semino, E. (2006). A corpus-based study of metaphors for speech activity in British English. In A. Stefanowitsch & S. T. Gries (Eds.), *Corpus-based approaches to metaphor and metonymy*. Amsterdam: John Benjamins Publishing Company, pp. 35-60.
Semino, E. (2008). *Metaphor in discourse*. Cambridge: Cambridge University Press.
Semino, E. and Koller, V. (2009). Metaphor, politics and gender: a case study from Italy. In A. Kathleen (Ed.), *Politics, gender and conceptual metaphors*. Basingstoke: Palgrave-MacMillan, pp. 36-61.
Steen, G. J., Dorst, A. G., Herrmann, J. B., Kaal, A., Krennmayr, T., and Pasma, T. (2010). *A method for linguistic metaphor identification: From MIP to MIPVU*. Amsterdam and Philadelphia: John Benjamins Publishing Company.
The Stanford Natural Language Processing Group. (2015). Stanford Log-linear Part-Of-Speech Tagger (Version3.6.0) [Software]. Available from htttp://nlp.stanford.edu/software/tagger.html

A Multi- versus a Single-classifier Approach for the Identification of Modality in the Portuguese Language

João Sequeira[1], Teresa Gonçalves[1], Paulo Quaresma[1], Amália Mendes[2], Iris Hendrickx[23]
[1]Department of Informatics, University of Évora, Portugal
[2]Center of Linguistics, University of Lisbon, Portugal
[3]Centre for Language Studies, Radboud University Nijmegen, The Netherlands
d11594@alunos.uevora.pt, tcg@uevora.pt, pq@uevora.pt,
amaliamendes@letras.ulisboa.pt, iris@i-hx.nl

Abstract

This work presents a comparative study between two different approaches to build an automatic classification system for Modality values in the Portuguese language. One approach uses a single multi-class classifier with the full dataset that includes eleven modal verbs; the other builds different classifiers, one for each verb. The performance is measured using precision, recall and F_1. Due to the unbalanced nature of the dataset a weighted average approach was calculated for each metric. We use support vector machines as our classifier and experimented with various SVM kernels to find the optimal classifier for the task at hand. We experimented with several different types of feature attributes representing parse tree information and compare these complex feature representation against a simple bag-of-words feature representation as baseline. The best obtained F_1 values are above 0.60 and from the results it is possible to conclude that there is no significant difference between both approaches.

Keywords: Natural Language Processing, Modality, Feature Selection, Support Vector Machines

1. Introduction

In the last years there was a great development in fields related to Machine Learning in the pursuit of forms of Artificial Intelligence (AI), which has become a major research trend of both academic and companies, like Google and Microsoft. The fields are diverse, ranging from financial fraud identification, image recognition and even systems that can rewrite their own code or write other programs (Caughill, 2017; Gershgorn, 2017; Galeon, 2017). Natural Language Processing (NLP) is a related research field that includes tasks aiming at understanding texts, information extraction and text classification.

Presently, one of the most active sub-field focuses on sentiment analysis and opinion mining (Pang and Lee, 2008). This includes tasks such as the (automatic) distinction between the factual and non-factual nature of events and the detection of the subjective perspective underlying texts. Modality is one indicator of subjectivity and factuality and it is usually defined as the expression of the speaker's opinion and attitude towards the proposition (Palmer, 1986). Traditionally, it covers epistemic modality, which is related to the degree of commitment of the speaker to the truth of the proposition (whether the event is perceived as possible, probable or certain), but also deontic modality (obligation or permission), capacity and volition (Sequeira et al., 2016). Information about the modality of a text is crucial for the above mentioned trends on automatic fact finding and information extraction.

This work extends the experiments done previously (Sequeira et al., 2016) in the pursuit of creating a semi-automatic modality tagging system for the Portuguese language from a manually annotated corpus that uses the modality scheme described by Hendrickx et al. (2012) and Mendes et al. (2016).

In this study we focus on machine learning optimization and feature selection for modality detection and labeling. We compare two different system architectures, namely one classifier trained on all modal verbs and one architecture where we train a classifier for each modal verb separately. Such 'word expert' approach is known to work well in word sense disambiguation (a closely related task) (Hoste et al., 2002). We also investigate whether the complex feature representation based on parse information as applied in our previous work (Sequeira et al., 2016) is indeed more informative than a simple bag-of-word feature representation.

The paper is structured as follows: Section 2 introduces related work done in the field of modality, Section 3 presents the developed system describing the experimental setup, corpus and dataset information, attributes extracted and results obtained and Section 4 discusses some conclusions and future work aiming at improving the system.

2. Related work

As Portuguese is one of the 10 most spoken languages in the world, with more than 260 millions of speakers (da Língua Portuguesa, 2015), the development of natural language processing tools and linguistically annotated resources for Portuguese are crucial to keep up with the current information society (Branco et al., 2012).

However, most studies related to modality still focus on the English language, and besides our own work, not much tools have been developed for Portuguese. Baker et al. (2010), Matsuyoshi et al. (2010), Nirenburg and McShane (2008) and Sauri et al. (2006) present modality annotation schemes for the English language; for Portuguese we can identify the work from Hendrickx et al. (2012) for written European Portuguese, Ávila and Melo (2013) for spoken Brazilian Portuguese, and the updated proposal of both teams in Ávila et al. (2015).

Thompson et al. (2008) addressed the identification of ex-

pressions linked to modality in biomedical texts using three dimensions: the kind of knowledge, level of certainty and point of view. Their approach uses a list of words and phrases with modal characteristics specific for the biomedical domain. Baker et al. (2010) tested two rule-based modality taggers that identify both the modal trigger (word or word list where modality is expressed, usually by the use of modal verbs) and its target (the event, state, or relation over which the modality has scope) and achieved results of 86% precision for a standard LDC data set.

Ruppenhofer and Rehbein (2012) developed a modal verb annotation scheme for news articles written in English. The system uses a classifier of maximum entropy (Ratnaparkhi, 1996) to identify the verbs *can*, *may/might*, *must* and *should*. The attributes used are divided into three categories: (i) target/verb; (ii) context; (iii) path. They used different combinations of attributes with different context sizes and the results were compared to those of a baseline system always assigning the most common value to each verb. The best result was achieved for the verb *must* with an accuracy of value 93,50%, followed by the verb *shall/should* with 91,61%, *may/might* with 85,71% and finally *can* with 68,70%.

In what concerns the Portuguese language, Sequeira et al. (2016) is a earlier and less developed version of the work presented here. The goal was to select the best set of attributes for creating automatic taggers and compare the results with a bag-of-words (bow) approach. The paper covers the creation of the corpus (composed by eleven verbs), the use of a parser to extract syntactic and semantic information from the sentences and a machine learning approach to identify modality values.

3. Experiments

Like in the previous experiments, eleven Portuguese modal verbs (that we call triggers) are studied. They are:"arriscar" (chance/risk/dare), "aspirar" (aspire), "conseguir" (manage to/succeed in/be able to), "considerar" (consider/regard), "dever" (shall/might),"esperar" (wait/expect), "necessitar" (need/require), "permitir" (allow/permit), "poder" (may/can), "precisar" (need) and "saber" (know).

These verbs are polysemous and are deliberately chosen as our focus verbs because they can express more than one type of modality. For example, the verb "poder" can be *Epistemic* stating that something is possible, *Deontic* denoting permission or may express an *Internal Capacity* when expressing the fact that someone is able to do something.

In Sequeira et al. (2016) several combinations of classes of attributes, namely trigger, path and context were tested and the best one was selected (path+context). An attribute ranker (using information gain) singled out the following attributes as the most informative for *path*:

- presence of an Accusative node between the root and the verb node
- no explicit subject in the left brother node
- the left brother node receives the semantic role Theme
- presence of an infinitive clause in the path: either from the root to the verb or as the right brother node
- the left brother node is a Dative object with the function Beneficiary

For *context*, some of the most important attributes occur in the left tree:

- the lemma *lei* 'law' occurs in the left context
- the dative clitic *lhe* 'to him/her' occurs in the left context

These attributes point to certain properties of the trigger and the context that lead to one modal interpretation. They may be somehow unexpected, as the case of the attribute "Dative brother node in left context". The combination of attributes for path listed above, namely the presence of an Accusative node which is of the type infinitival clause, favours an epistemic reading of the verb *permitir*, as illustrated in (1). Moreover, many of the examples of epistemic possibility reading with *permitir* are associated to constructions where the left brother node is a Dative object, another attribute listed for path (example (2)).

(1) Mas estes primeiros dias já *permitem tirar conclusões*.
'But these first days already make it possible to draw conclusions.'

(2) Agora, embora não seja capaz de pintar porque não tenho técnica para o fazer, descobri que o computador *me permite* transformar as minhas imagens de tal maneira que ficam a parecer autênticas pinturas.
'Now, although I'm not capable of painting because I don't have the technique to do so, I discovered that the computer allows me to transform my images in such a way that they end up looking like authentic paintings.'

In what concern the class of attributes for context, a deontic reading of the verb *permitir* is strongly related to the presence of the lemma *lei* 'law' in subject position (4 contexts out of 5), as illustrated in (3).

(3) E acrescenta que não existe nenhuma lei que permita à Portugal Telecom cortar o serviço telefónico por os utentes não pagarem, por exemplo, as chamadas de valor acrescentado, tipo telefonemas eróticos, etc.
'And [he/she] adds that there is no law that allows Portugal Telecom to cut the phone service when users don't pay, for instance, value added calls, such as erotic phone calls.'

We keep the same set of attributes in this experiment. Besides a baseline using a bag-of-words approach, this work uses that attribute setting to compare different classification experiments, namely:

- `exp.A`: to build a specific classifier for each verb, aiming at detecting the specific modality types (setting used in (Sequeira et al., 2016))
- `exp.B`: to build a single classifier with the full corpus (all verbs and all types of modality)

- `exp.C`: the same as `exp.B` but with an extra attribute – the lemma of the trigger

The advantage of `exp.A` is that such a classifier has to learn a smaller set of possible modal values (2-4 instead of 11) but has less examples to train on. `exp.B` and `exp.C` on the other hand has to learn to distinguish 11 different modal values but has more training examples to learn from. Note that `exp.B` is only included to measure the effect of knowledge about the trigger. While the modal values are shared among different verbs, we expect knowledge about the trigger still to be crucial in obtaining good results.

The Sequential Minimal Optimization (SMO) (Platt, 1999) algorithm, an improved version of the Support Vector Machine (SVM) (Vapnik, 1998), is used to build the automatic classification system and the performance is measured using precision, recall and F_1 measures.

3.1. Corpus and dataset

The dataset is composed by 936 sentences (examples) containing these modal verbs. In total eleven different modality values are expressed by these modal verbs, but each verb itself has between 2 -4 possible modal meanings. Table 1 characterizes each verb. As we can see, the number of sentences for each verb varies from 51 for "necessitar" to 254 for "poder". The number of sentences for each modal value also varies (from 11 examples for *evaluation* to 299 examples for *epistemic possibility*).

3.2. Feature extraction

The extracted attributes are the same as the ones reported in (Sequeira et al., 2016): a set related with the (i) *trigger* (modal verb information), another related with (ii) *context* (with a window of size five), and one related with the (iii) *path* (syntactic and morphological information extracted from the parse tree trigger). The selection of trigger, context and path is inspired by the work of (Ruppenhofer and Rehbein, 2012) and our goal was to be able to compare our results with their findings. The attributes are based on the syntactic and morphological analysis trees generated by the PALAVRAS parser (Bick, 1999; Bick, 2000). Table 2 summarizes the attributes extracted: for the *trigger* set, information from the trigger itself and from the ancestors; for the *path* set, information about the trigger's siblings and the path from the trigger to root; for the *context* set, information about the words to the left and right of the trigger.

3.3. Experimental setup

Using a SVM model and a 5-fold stratified cross-validation procedure, precision, recall and F_1 weighted averages were calculated for the three different classification experiments and compared with a bag-of-words approach as baseline. Different kernels with default parameters were tested, namely the polynomial kernel with degrees 1 (linear kernel) , 2 and 3 and the radial basis function.

Appropriate statistical tests with 95% of significance were applied to analyse the differences between results. These machine learning experiments were conducted using Weka framework (Hall et al., 2009).

verb	modality type	# example	
arriscar	effort	20	46
	epistemic belief	1	
	epistemic possibility	25	
aspirar	epistemic belief	18	52
	volition	34	
conseguir	participant-internal capacity	42	87
	epistemic possibility	4	
	success	41	
considerar	epistemic belief	15	26
	evaluation	11	
dever	epistemic belief	2	124
	deontic permission	3	
	deontic obligation	78	
	epistemic possibility	41	
esperar	epistemic belief	30	57
	volition	26	
	epistemic possibility	1	
necessitar	deontic obligation	8	51
	participant-internal necessity	41	
	participant-internal capacity	2	
permitir	deontic permission	19	80
	epistemic possibility	61	
poder	deontic permission	46	254
	deontic obligation	1	
	participant-internal capacity	40	
	epistemic possibility	167	
precisar	deontic obligation	10	56
	participant-internal necessity	46	
saber	participant-internal capacity	10	103
	epistemic knowledge	93	

Table 1: Corpus characterization: number of sentences per modal value for each verb.

3.4. Results

Table 3 present the weighted average precision for the described experiments.

The best weighted precision value (0.691) was obtained using the bag-of-words approach with the verb lemma as additional attributes using a single linear classifier (`exp.C`) but there's no significant difference with the experiment using *path+context* attributes with 11 classifiers, one for each verb (0.689) with a polykernel of degree 2 (`exp.A`). The worst result (0.102) was obtained using a bag-of-words representation with a RBF kernel and a single classifier without verb lemma information.

As expected knowledge about the trigger is very informative for the classifier and the results in `exp.B` are the lowest of the three options.

Comparing the kernel functions one can conclude that RBF kernel has the worst precision values; on the other hand the linear kernel seems to be the best when using a bag-of-words approach (baseline), while the polynomial kernel with degree 2 is better when using *path+context* attributes. Looking at the different classification settings, it seems that using a single classifier for the 11 modal values does not improve precision when comparing to a setting using a dif-

attribute set	source	attributes
trigger	trigger	POS function role morphological semantic
	ancestors	POS function
path	siblings	POS function role morphological semantic
	trigger to root	POS function
context	left/right trigger	POS word lemma

Table 2: Attributes extracted from trigger, path and context

ferent classifier for each verb (but also does not seem to hurt the classification system).

attributes	kernel	exp.A	exp.B	exp.C
path+ context	poly, d1	.678	.420	.659
	poly, d2	**.689**	.447	.627
	poly, d3	.678	.447	.582
	rbf	.615	.285	.544
baseline	poly, d1	.681	.355	**.691**
	poly, d2	.652	.261	.553
	poly, d3	.612	.266	.320
	rbf	.605	.102	.424

Table 3: Weighted average precision values.

Table 4 present the weighted average recall for the described experiments. The best value (0.708) was also obtained using the bag-of-words approach with the verb lemma as additional attributes using a single linear classifier (exp.C) but, once again, there's no significant difference with the experiment using *path+context* attributes with 11 classifiers (0.698) with a polykernel of degree 2 (exp.A). The worst result (0.121) was obtained using a bag-of-words representation with a polynomial kernel of degree 3 and a single classifier without verb lemma information.

For the recall measure it seems that the kernel function does not influence the performance of the classifier as it was the case for precision. Nonetheless, it seems that for the bag-of-words the linear kernel is the best while the polynomial kernel with degree 2 is better when using *path+context* attributes. Looking at the different classification settings, it seems that using a single classifier for the 11 modal values (exp.C) can hurt recall when comparing to a setting using a different classifier for each verb (exp.A), as values of 0.698 vs .0652 were obtained respectively.

Finally, Table 5 present the weighted average F_1 values

attributes	kernel	exp.A	exp.B	exp.C
path + context	poly, d1	.678	.436	.675
	poly, d2	**.698**	.475	.652
	poly, d3	.693	.453	.584
	rbf	.673	.408	.530
baseline	poly, d1	.689	.385	**.708**
	poly, d2	.667	.314	.578
	poly, d3	.640	.121	.353
	rbf	.668	.319	.456

Table 4: Weighted average recall values.

for the described experiments. As expected, the best value (0.683) was obtained using the bag-of-words approach with the verb lemma as additional attribute using a single linear classifier (exp.C) even if there's no significant difference with the experiment using *path+context* attributes with 11 classifiers (0.678) with a polykernel of degree 2 (exp.A). The worst result (0.129) was obtained using a bag-of-words representation with a polynomial kernel of degree 3 and a single classifier without verb lemma information.

Using one classifier for all modal values with additional verb lemma attribute (exp.C) or a specific classifier for each verb similar F_1 performance values are achieved when using a linear kernel (both for the baseline and the *path+context* set of attributes). While the value maintains stable for kernels of higher degrees for the exp.A setting , it consistently decreases for exp.C one.

attributes	kernel	exp.A	exp.B	exp.C
path + context	poly, d1	.673	.426	.664
	poly, d2	**.678**	.433	.620
	poly, d3	.664	.386	536
	rbf	.628	.279	.425
baseline	poly, d1	.658	.327	**.683**
	poly, d2	.627	.223	.536
	poly, d3	.616	.129	.275
	rbf	.627	.155	.333

Table 5: Weighted average F_1 values of the outcomes with different settings of the polykernel with degree 1 (d1),2 (d2) and 3(d3), and the RBF kernel.

4. Conclusions and Future work

This work extends previous experiments that try to identify the best automatic approach to tag modality in the Portuguese language.

Eleven modal verbs were used and morphological, syntactic and some semantic attributes were extracted from the 936 sentences using the PALAVRAS parser. Several experiments were conducted using two different sets of attributes: a bag-of-words representation was used as baseline and the second set includes several attributes taken from the syntax parse tree path and modal verb context. Two different kernel functions were tested (polynomial with degree ranging from 1 to 3 and RBF kernels). Three different classification approaches were set up: (a) a set of

11 classifiers, one for each verb (using the corresponding subset of sentences); (b) a single multi-class classifier (11 classes, one for each modal value); (c) the same as (b) but adding a valuable attribute: the trigger verb lemma.

Comparing the performance of the different systems, one can conclude that adding lemma information improves the performance when using a single multi-class classifier, but there is no significant difference to the multi-classifier approach. With the individual classifiers all values (for all three measures) were above 0.60, independently of the set of attributes and kernels used. This is not true when using a single classifier for all the existing modal values.

The corpus is relatively small (specially if we take into account the number of possible different classes) and is not balanced. This certainly influences the performance of the system.

As future work, we intend to expand the corpus trying to get a more balanced version of examples. Next steps for building a complete automatic modality tagging system are to identify the source of the modality and the target linked to the modality value.

5. Bibliographical References

Baker, K., Bloodgood, M., Dorr, B., Filardo, N. W., Levin, L., and Piatko, C. (2010). A modality lexicon and its use in automatic tagging. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Bick, E. (1999). *The parsing system PALAVRAS*. Aarhus University Press.

Bick, E. (2000). *The Parsing System "Palavras". Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. University of Arhus, Århus.

Branco, A., Mendes, A., Pereira, S., Henriques, P., Pellegrini, T., Meinedo, H., Trancoso, I., Quaresma, P., Strube de Lima, V. L., and Bacelar, F. (2012). *A língua portuguesa na era digital – The Portuguese Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

Caughill, P. (2017). Google's AI is Learning to Make Other AI.

da Língua Portuguesa, O. (2015). As 10 línguas mais faladas no mundo. Technical report.

Galeon, D. (2017). New AI Can Write and Rewrite Its Own Code to Increase Its Intelligence.

Gershgorn, D. (2017). Microsoft's AI is learning to write code by itself, not steal it.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November.

Hendrickx, I., Mendes, A., and Mencarelli, S. (2012). Modality in text: a proposal for corpus annotation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Hoste, V., Hendrickx, I., Daelemans, W., and van den Bosch, A. (2002). Parameter optimization for machine-learning of word sense disambiguation. *Natural Language Engineering*, 8(4):311–325.

Matsuyoshi, S., Eguchi, M., Sao, C., Murakami, K., Inui, K., and Matsumoto, Y. (2010). Annotating event mentions in text with modality, focus, and source information. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Mendes, A., Hendrickx, I., Gonçalves, T., Quaresma, P., and Sequeira, J. (2016). Modality annotation for portuguese: from manual annotation to automatic labeling. *LiLT - Linguistic Issues in Language Technology*, volume 14.

Nirenburg, S. and McShane, M. (2008). Annotating modality. Technical report, University of Maryland, Baltimore County, USA, March.

Palmer, F. R. (1986). *Mood and Modality*. Cambridge textbooks in linguistics. Cambridge University Press.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

Platt, J. C. (1999). Advances in kernel methods. chapter Fast Training of Support Vector Machines Using Sequential Minimal Optimization, pages 185–208. MIT Press, Cambridge, MA, USA.

Ratnaparkhi, A. (1996). A maximum entropy model part-of-speech tagger. In *EMNLP'96 – Empirical Methods in Natural Language Processing Conference*, pages 133–141.

Ruppenhofer, J. and Rehbein, I. (2012). Yes we can!? annotating english modal verbs. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Sauri, R., Verhagen, M., and Pustejovsky, J. (2006). Annotating and recognizing event modality in text. In *FLAIRS Conference*, pages 333–339.

Sequeira, J., Gonçalves, T., Quaresma, P., Mendes, A., and Hendrickx, I. (2016). Using syntactic and semantic features for classifying modal values in the portuguese language. In *CICLing 2016, 17th International Conference on Intelligent Text Processing and Computational Linguistics*.

Thompson, P., Venturi, G., McNaught, J., Montemagni, S., and Ananiadou, S. (2008). Categorising modality in biomedical texts. In *Proceedings of the LREC 2008 Workshop on Building and Evaluating Resources for Biomedical Text Mining*, pages 27–34, Marrakech, Marrocos.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley-Interscience.

Ávila, L. and Melo, H. (2013). Challenges in modality annotation in a brazilian portuguese spontaneous speech corpus. In *Proceedings of IWCS 2013 WAMM Workshop on the Annotation of Modal Meaning in Natural Language*, Postam, Germany. Association for Computational Linguistics.

Ávila, L., Mendes, A., and Hendrickx, I. (2015). Towards a unified approach to modality annotation in portuguese. In *Proceedings of the IWCS Workshop on Models for Modality Annotation, MOMA 2015*, pages 1–8.

All-words Word Sense Disambiguation Using Concept Embeddings

Rui Suzuki, Kanako Komiya, Masayuki Asahara, Minoru Sasaki, Hiroyuki Shinnou
Ibaraki University, 4-12-1 Nakanarusawa, Hitachi, Ibaraki JAPAN,
National Institute for Japanese Language and Linguistics, 10-2 Midoricho, Tachikawa, Tokyo, JAPAN,
{17nm709g@vc.ibaraki.ac.jp, kanako.komiya.nlp, minoru.sasaki.01, hiroyuki.shinnou.0828}
@vc.ibaraki.ac.jp, masayu-a@ninjal.ac.jp

Abstract

All-words word sense disambiguation (all-words WSD) is the task of identifying the senses of all words in a document. Since the sense of a word depends on the context, such as the surrounding words, similar words are believed to have similar sets of surrounding words. We therefore predict the target word senses by calculating the distances between the surrounding word vectors of the target words and their synonyms using word embeddings. In addition, we introduce the new idea of concept embeddings, constructed from concept tag sequences created from the results of previous prediction steps. We predict the target word senses using the distances between surrounding word vectors constructed from word and concept embeddings, via a bootstrapped iterative process. Experimental results show that these concept embeddings were able to improve the performance of Japanese all-words WSD.

Keywords: word sense disambiguation, all-words, unsupervised

1. Introduction

Word sense disambiguation (WSD) involves identifying the senses of words in documents. In particular, the WSD task where the senses of all the words in a document are disambiguated is referred to as all-words WSD. Much research has been carried out, not only on English WSD but also on Japanese WSD, for many years. However, there has been little research on Japanese all-words WSD, possibly because no tagged corpus has been available that was large enough for the task. Usually, the Japanese sense dataset Balanced Corpus of Contemporary Written Japanese (BCCWJ) (Maekawa et al., 2014), tagged with sense IDs from Iwanami Kokugo Jiten (Nishio et al., 1994), is used for supervised WSD. However, unsupervised approaches to all-words WSD often require synonym information, which sense datasets cannot provide. This paper reports research on Japanese all-words WSD that uses a corpus that is in its infancy, namely BCCWJ annotated with concept tags or article numbers from the Word List by Semantic Principles (WLSP) (Kokuritsukokugokenkyusho, 1964), which is a Japanese thesaurus. In the WLSP, the article numbers indicate shared synonyms.

In the WLSP thesaurus, words are classified and organized by their meanings[1] and each WLSP record contains the following fields: record ID number; lemma number; record type; class; division; section; article; article number; paragraph number; small paragraph number; word number; lemma (with explanatory note); lemma (without explanatory note); reading; and reverse reading. Each record has an article number, which represents four fields: class; division; section; and article. For example, the word "犬"(*inu*, meaning spy or dog) has two records in the WLSP, and therefore has two article numbers, 1.2410 and 1.5501, indicating that the word is polysemous. In addition, there are 240 semantic breaks in the WLSP, which allow words to be classified in more detail than with the article numbers alone. Note that the article numbers are used as concept tags, because many words have the same article numbers. Several words can have the same precise article number, even when the semantic breaks are considered.

[1] https://www.ninjal.ac.jp/english/publication/catalogue/goihyo/

2. Related Work

WSD methods can broadly be divided into two categories: supervised and unsupervised approaches. Generally, WSD using supervised learning can achieve high accuracy rates, but requires substantial manual effort due to the need for a sufficient amount of manually-annotated training data. On the other hand, unsupervised learning does not need such manual input, but it is difficult to obtain as high an accuracy rate as with the supervised learning.

Many WSD methods have been proposed. WLSP article numbers or hypernyms of target words obtained from the WLSP are often used as supervised learning features. Vu and Parker (2016) proposed the idea of K-embeddings for learning concept embeddings. Komiya et al. (2015) proposed a surrounding word sense model for Japanese all-words WSD using unsupervised learning which assumes that the sense distribution of surrounding words changes depending on the sense in which a polysemous word is used. Shinnou et al. (2017b) proposed a WSD system capable of performing Japanese WSD easily using a supervised approach.

3. WSD Using Synonym Information from the WLSP

We propose three WSD methods that use synonym information from the WLSP: 1) a method using only the word embeddings of synonyms, 2) a method using both the word and concept embeddings of synonyms, and 3) a method using only the concept embeddings of synonyms.

3.1. WSD Using the Word Embeddings of Surrounding Words

Since the senses of words are determined by context, such as the surrounding words, similar words are believed to

have similar sets of surrounding words. The proposed method is based on this idea and consists of three steps.
First, we generate word embeddings from an untagged corpus and concatenate the embeddings of the words surrounding each target word, creating what we will refer to as the surrounding word vectors. For example, a word “犬”(*inu*, meaning spy or dog) has two article numbers, 1.2410 and 1.5501, in WLSP. Therefore, this word is polysemous. If there was a sentence including the word “犬” below, the surrounding word vector for the word “犬” is the concatenated vector of the word embeddings or the words “警察”, “の”, “だ”, and “.”.

彼は　　警察　の犬　だ.
he (topic marker) police of spy is .
‘He is a spy of police.’

Second, we make synonym lists for the senses of each target word using the WLSP, and create surrounding word vectors for each synonym appearing in the corpus. Note that each surrounding word vector is labeled according to the sense of the target word, which is equivalent to the sense of the synonym. For example, if the target word was “犬”, its synonyms are the words that have the article number, 1.2410 or 1.5501. They are agent, ninja, and so on when the number was 1.2410, and wolf, fox, and so on when it was 1.5501. The surrounding word vectors for these synonyms are created from each synonym appearing in the corpus and are labeled as 1.2410 or 1.5501. These labels can be obtained in an unsupervised manner, that is, it does not make use of sense-tagged data, and this method is knowledge-based because the WLSP is a thesaurus.
Finally, we predict the target word senses using the K-nearest neighbors (KNN) algorithm, based on the distances between the surrounding word vectors for the target words and their synonyms. In other words, we calculate the distances between the surrounding word vectors for the target word “犬” and its synonyms labeled as 1.2410 or 1.5501 and determine the word sense of the target word via KNN algorithm. That is, if the synonym with label 1.2410 was nearer by the algorithm, the word sense of the target word will be 1.2410 and vice versa.

3.2. WSD Using the Word and Concept Embeddings of Surrounding Words

For this method, we repeat the target word sense prediction process (based on the one described in Section 3.1.), with the prediction steps at the n_{th} iteration being as follows.

1. Replace the word tokens in the corpus with their concept tags, using the results from the $n-1_{st}$ prediction step, and create concept embeddings using the conceptually-tagged corpus.(cf. Figures 1 and 2).

根本 要 がビデオ ジョッキー を 務め 、1960年 代 後半 から80年 代 前半 の ロック を届ける 。
自民 党 の「 政策 新 人類 」渡辺 喜美 衆院 議員 は 提案 する「 経済 非常 事態 」を 宣言 し 産業 再生 委 の 創設 を ！
⋮

Figure 1: Word Tokens

根本 要 が 1.4620 1.2410 を 2.3320 、1.1960 1.1960 1.1960 1.1960 1.1630 1.1623 1.1650 から 1.1960 1.1960 1.1630 1.1623 1.1650 の ロック を 2.1521 。
1.2760 党 の「 1.3084 3.1660 1.5501 」渡辺 喜美 1.2730 1.2400 は 1.1210 2.1211 「 1.3710 1.1331 1.1000 」を 1.3100 2.1211 1.3801 1.1211 委 の 1.1220 を！
⋮

Figure 2: Conceptually-tagged Corpus

For example, a text shown in Figure 1 were converted to a concept-text in Figure 2. Here, the word “届ける”(*todokeru*, meaning deliver) is polysemous that has three article numbers, 2.1521, 2.3141, and 2.3830. This word is replaced with 2.1521 in Figure 2 according to the result of the $n-1_{st}$ prediction step. In addition, the words that have no record in WLSP are not replaced by the article numbers. The words “根本”(*Nemoto*, person name) and “要”(*Kaname*, person name) in Figure 2 are those examples.

2. Generate surrounding word vectors for the target words and their synonyms, namely vectors where the word and concept embeddings have been concatenated. For the first prediction step, use the prediction results for the word embeddings only, as described in Section 3.1.
3. Predict the target word senses using the KNN algorithm, as described in Section 3.1.

For the method using only the concept embeddings of synonyms, we concatenated only the concept embeddings instead of the word and concept embeddings in the second step. We investigate the optimal number of iterations experimentally in Section 4.

3.3. Word List by Semantic Principles

Table 1 shows the structure of WLSP. We extracted メートル (meter) and its synonyms for example. In the WLSP thesaurus, words are classified according to an article number. The article number represents four fields: class, division, section, and article. The class classifies the words into four groups according to a part of speech, and the division, section, and article further classify them by according to a word’s meaning. In addition, there are 240 semantic breaks(“＊” in Table 1) in WLSP, which allow words to be classified in more detail than with the article numbers alone. For example, there are approximately 500 counter suffixes (meter, yard, liter, gallon, etc.) that have an article number, 1.1962, in WLSP. Since these words have same article number, they can be deemed as words that have the same meanings. However, if the semantic breaks are took into consideration, they are deemed as two word groups, “meter yard” and “liter gallon.”

3.4. Selecting Synonyms Using the WLSP

First, we find the WLSP article numbers for all words in the corpus. The synonyms used for our methods of Section 3.1. are as follows.

- Words with the same article numbers as the target words. Here semantic breaks are also considered if available.
- Words are excluded if they are synonyms for more than one sense of a given target word.

article number	paragraph number	small paragraph number	word number	lemma
1.1962	15	1	1	メートル (meter)
			2	キロメートル (kilometer)
			3	キロ (kilo)
			⋮	⋮
	18	2	1	ヤード (yard)
	⋮	⋮	⋮	⋮
	23	99	99	*
	24	1	1	リットル (liter)
	⋮	⋮	⋮	⋮
	26	1	1	ガロン (gallon)
	⋮	⋮	⋮	⋮

Table 1: WLSP

> For example, imagine that a target word, X, has two senses: Sense 1 and Sense 2. If the synonyms for Sense 1 are A, B, and C, and the synonyms for Sense 2 are C, D, and E, we exclude C from the synonym sets for both Sense 1 and Sense 2.

However, the above conditions can not take into consideration the ambiguity of the synonyms. For example, if a word A, the synonym of X for Sense 1, is polysemous, the surrounding word vectors of A are not necessarily vectors meaning only Sense 1. Therefore, we take into consideration the ambiguity of the synonyms for method in Section 3.2..

The synonyms used at the n_{th} iteration of our method are as follows.

- Polysemous words with the same sense as the target word. For the KNN algorithm at the n_{th} step, we only use surrounding word vectors whose predicted sense was the same as that of the target word at the $n - 1_{st}$ prediction step.
- Monosemous words with the same sense as the target word.
- Words are excluded if they are synonyms for more than one sense of a given target word.

In other words, the surrounding word vectors of A are created not from all word tokens of A, but from only word tokens of A predicted as Sense 1 at the n-1 prediction.

4. Experiment

We used the BCCWJ for our experiments, and used the annotation by (Kato et al., 2017) to add word sense annotations. This corpus includes 3,790 word types and 22,568 word tokens, including 1,096 word types and 4,760 word tokens for polysemous words. The polysemous words have an average of 3.16 word senses per word token and an average of 2.59 word senses per word type, so the accuracy rate of a random baseline method would be 31.65% (the inverse of the average number of senses for the polysemous words). The accuracy of the most frequent sense baseline is 91.7%.

Number of URLs collected	83,992,556
Number of sentences (tokens)	3,885,889,575
Number of sentence (types)	1,463,142,939
Number of words (tokens)	25,836,947,421

Table 2: Statistics for the NWJC-2014-4Q Dataset

CBOW or skip-gram	-cbow	1
Dimensionality	-size	200
Number of surrounding words	-window	8
Number of negative samples	-negative	25
Hierarchical softmax	-hs	0
Minimum sample threshold	-sample	1e-4
Number of iterations	-iter	15

Table 3: Parameters Used to Generate NWJC2vec

Note that we cannot know what is the most frequent sense using an unsupervised approach. We used NWJC2vec[2] (Shinnou et al., 2017a) for the Japanese word embeddings. This is a set of word embeddings generated from the NWJC-2014-4Q dataset, which is an enormous Japanese corpus developed using the word2vec[3] tool. Tables 2 and 3 present summary statistics for the NWJC-2014-4Q data and the parameters used to generate the word embeddings, respectively. We used word2vec (Mikolov et al., 2013c; Mikolov et al., 2013a; Mikolov et al., 2013b) to generate the concept embeddings, and the parameters used are summarized in Table 4.

The window size for the surrounding word vectors was set to two, meaning four words in total. When the number of surrounding words was smaller than the window size, we used a zero vector. Therefore, the dimensionality of the surrounding word vectors was 800 when they were created using only the word embeddings, and 1,000 when both the word and concept embeddings were used. It was 200 when only the concept embeddings are used.

We used KNeighborsClassifier from the scikit-learn[4] library as the KNN algorithm. We tried K values of 1, 3, and 5, as well as uniform and distance-based weights. Default settings were used for all other parameters.

5. Results

Table 5 shows the WSD results when only the word embeddings of the surrounding words were used, and Table 6

CBOW or skip-gram	-cbow	1
Dimensionality	-size	50
Number of surrounding words	-window	5
Number of negative samples	-negative	5
Hierarchical softmax	-hs	0
Minimum sample threshold	-sample	1e-3
Number of iterations	-iter	5
Minimum frequency to consider	-min-count	1

Table 4: Parameters Used for the Concept Embeddings

[2]http://nwjc-data.ninjal.ac.jp/

[3]https://code.google.com/archive/p/word2vec/

[4]http://scikit-learn.org/stable/

Weights	K=1	K=3	K=5
Uniform	52.6	**53.0**	**53.0**
Distance-based	52.6	52.7	52.7

Table 5: WSD Accuracy Rate Using Word Embeddings Only

K	weight	1	2	3	4	5	6
1	Uniform	57.0	56.2	56.4	56.3	56.5	56.3
1	Distance	57.0	56.2	56.4	56.4	56.5	56.4
3	Uniform	57.1	56.5	56.6	56.5	56.6	56.5
3	Distance	57.1	56.3	56.5	56.4	56.6	56.4
5	Uniform	**57.3**	56.6	56.8	56.7	56.8	56.7
5	Distance	57.2	56.3	56.6	56.5	56.7	56.5

Table 6: WSD Accuracy Rate Using Both Word and Concept Embeddings, for between One and Six Iterations

shows the results when both the word and concept embeddings were used. Table 7 shows the WSD results when only the concept embeddings of the surrounding words were used. The numbers in the column headings give the numbers of iterations.

Table 5 shows that uniform weights always gave better results than distance-based weights. The best results in Table 5 occurred for K = 3 or 5 and uniform weights, where we obtained an accuracy rate of 53.0%. In addition, Table 5 shows that our method significantly outperformed the random baseline, regardless of the K and weight settings used. The best results in Table 6 occurred at the 1_{st} step, for K = 5 and uniform weights and we obtained the best results in Table 7 at the 2_{nd} step, for K = 5 and distance weights. The results when using only the concept embeddings were best in this experiment.

6. Discussion

Considering our results, we can see that the accuracies in Table 6 are better than those in Table 5, indicating that starting from word-embedding-based predictions and then developing concept embeddings generated from a conceptually-tagged text corpus is effective for WSD.

In addition, Tables 5 and 6 show that the accuracy of WSD using word and concept embeddings is high only when that of WSD using only word embeddings is also high. However, Tables 5 and 7 show that the accuracy of WSD using only concept embeddings does not follow this trend. To study this further, we varied the conditions used to build the synonym lists for the initial predictions to investigate

K	weight	1	2	3	4	5	6
1	Uniform	54.7	58.5	57.2	57.7	58.4	57.8
1	Distance	54.7	58.0	56.9	56.9	58.2	56.6
3	Uniform	54.4	58.3	56.9	58.1	56.3	58.5
3	Distance	53.7	58.4	57.4	56.7	58.4	57.8
5	Uniform	55.2	58.0	57.4	58.2	58.0	57.2
5	Distance	55.8	**58.8**	57.4	58.6	57.5	57.3

Table 7: WSD Accuracy Rate Using Concept Embeddings Only

Weights	K=1	K=3	K=5
Uniform	51.3	53.3	**53.4**
Distance-based	51.3	51.7	51.7

Table 8: WSD Accuracy Rate Using Word Embeddings Only, under Condition 1

K	weight	1	2	3	4	5	6
1	Uniform	56.9	56.3	56.3	56.3	56.4	56.4
1	Distance	56.9	56.3	56.3	56.3	56.4	56.3
3	Uniform	54.7	56.1	56.9	56.1	56.8	56.1
3	Distance	56.9	56.4	56.4	56.4	56.5	56.4
5	Uniform	54.9	56.2	**57.0**	56.2	56.9	56.2
5	Distance	**57.0**	56.5	56.5	56.5	56.6	56.6

Table 9: WSD Accuracy Rate Using Both Word and Concept Embeddings, under Condition 1, for between One and Six Iterations

the effect of this on WSD performance. We originally excluded words that were synonyms for more than one sense of a given target word, because we believed that KNN classification accuracy would decrease when the same vectors were generated for duplicate synonyms. The condition variations that we now considered were as follows.

1. Ignore semantic breaks.
2. Substitute paragraph numbers for semantic breaks.
3. Use only monosemous words as synonyms.

Tables 8, 9, and 10 show the WSD results when only the word embeddings of the surrounding words were used, when both the word and concept embeddings were used, and when only the concept embeddings of the surrounding words were used under Condition 1, respectively. Tables 11, 12, and 13 are those results under Condition 2 and Tables 14, 15, and 16 are those results under Condition 3.[5]

Under Condition 1, the number of target word synonyms increased compared that for the original conditions, meaning that the synonyms included more words with different senses. Condition 1 caused the accuracy to increase, when using only word embeddings, but the accuracy decreased when using word and concept embeddings. It indicates that the quality of the concept embeddings does not always de-

K	weight	1	2	3	4	5	6
1	Uniform	53.3	58.7	57.4	56.5	57.7	57.1
1	Distance	53.5	58.0	56.5	57.0	56.6	56.7
3	Uniform	54.5	56.4	57.3	58.0	57.2	57.1
3	Distance	53.7	**59.5**	58.3	59.1	58.4	58.5
5	Uniform	54.1	57.5	58.5	59.1	58.9	57.0
5	Distance	54.0	56.9	59.2	59.2	58.9	58.8

Table 10: WSD Accuracy Rate Using Concept Embeddings Only, under Condition 1, for between One and Six Iterations

[5]We carried out nine iterations under all the conditions but omit the results from $6_{th} - 9_{th}$ iterations under Condition 1 and Condition 2 due to space limitation.

Weights	K=1	K=3	K=5
Uniform	51.5	51.5	51.5
Distance-based	51.5	51.5	51.5

Table 11: WSD Accuracy Rate Using Word Embeddings Only, under Condition 2

K	weight	1	2	3	4	5	6
1	Uniform	56	58	58	57.9	57.9	57.9
1	Distance	56	58	57.9	57.9	57.9	57.
3	Uniform	56.4	58.2	58.2	58.1	58.1	58.1
3	Distance	56.2	58.1	58	58	58	58
5	Uniform	56.4	**58.4**	58.3	58.2	58.3	58.2
5	Distance	56.2	58.2	58.2	58.1	58.1	58.1

Table 12: WSD Accuracy Rate Using Both Word and Concept Embeddings, under Condition 2, for between One and Six Iterations

pend on the prediction accuracy using only word embeddings. The accuracy increased when using only concept embeddings and we obtained the best results when using only concept embeddings.

Under Condition 2, the synonyms only included words whose senses were closer to those of the target words than they were under the original conditions, since the paragraph numbers allowed the words to be classified in more detail than the semantic breaks were able to achieve. Here, the accuracy decreased when using only word embeddings but the accuracies increased when using word and concept embeddings and when using only concept embeddings. We again obtained the best results when using only concept embeddings.

In contrast, all accuracies increased under Condition 3 (shown in Tables 14, 15 and 16). The best results in this research occurred at the 7_{th} iteration for K = 3 and distance weights, where we obtained an accuracy rate of 59.8%. When polysemous words were used as synonyms, their senses were not necessarily the same as those of the target words, so the quality of the concept embeddings was improved by using only monosemous words as synonyms. In addition, about 70% of the words in the WLSP are monosemous, so the number of synonyms did not significantly decrease under Condition 3, and we believe this is why the accuracies improved. Only under Condition 3, many iterations caused the accuracy to increase. We believe that this is because the quality of the concept embeddings was better.

K	weight	1	2	3	4	5	6
1	Uniform	56.9	**59.5**	57.7	56.6	57.9	57
1	Distance	57.8	58.7	55.2	58.6	55.7	57.9
3	Uniform	57.9	57.6	54.9	57.9	54.9	55.6
3	Distance	56.6	56.1	58.3	57.1	57	59.1
5	Uniform	57.5	59.2	56.1	59.1	58.8	54.2
5	Distance	57.2	59.1	55.3	55.6	57.1	56.3

Table 13: WSD Accuracy Rate Using Concept Embeddings Only, under Condition 2, for between One and Six Iterations

Weights	K=1	K=3	K=5
Uniform	55.7	55.7	55.7
Distance-based	55.7	55.7	55.7

Table 14: WSD Accuracy Rate Using Word Embeddings Only, Using Condition 3

K	weight	1	2	3	4	5	6
1	Uniform	56.3	55.6	56.6	56.0	57.5	56.0
1	Distance	56.3	55.6	56.6	56.0	57.5	56.0
3	Uniform	56.6	55.7	56.6	56.1	57.5	56.2
3	Distance	56.6	55.7	56.7	56.0	57.6	56.1
5	Uniform	56.7	55.9	56.8	56.3	**57.7**	56.4
5	Distance	56.7	55.8	56.8	56.1	**57.7**	56.2

Table 15: WSD Accuracy Rate Using Both Word and Concept Embeddings, under Condition 3, for between One and Six Iterations

7. Conclusion

In this paper, we have proposed three methods for all-words WSD: 1) a method using only word embeddings of synonyms, 2) a method using both word and concept embeddings of synonyms, and 3) a method using only concept embeddings of synonyms. Experimental results for the proposed methods show that they all significantly outperformed a random baseline, indicating concept embedding was effective for WSD. The optimal conditions for selecting synonyms depend on both the corpus size and the target words. In the current study, the accuracies increased when only monosemous words were used as synonyms. We will use our method to annotate corpora.

Acknowledgment

This work includes the result of "Development of All-words WSD System for Creation of Correspondence Table of Word List of Semantic Principles and Iwanami Kokugo Jiten ", the joint research project of National Institute for Japanese Language and Linguistics. This work

K	weight	1	2	3	4	5	6
1	Uniform	58.5	57.4	58.4	57.8	58.3	56.4
1	Distance	59.0	55.8	57.9	56.6	58.1	58.5
3	Uniform	57.5	56.1	57.6	56.6	59.1	57.4
3	Distance	57.7	55.6	57.4	58.3	57.7	59.1
5	Uniform	56.7	58.0	59.0	57.8	**59.3**	59.2
5	Distance	57.5	56.2	56.6	58.0	59.2	58.0

K	weight	7	8	9
1	Uniform	57.7	57.7	57.7
1	Distance	58.7	56.2	58.5
3	Uniform	58.2	56.5	59.3
3	Distance	**59.8**	57.6	58.5
5	Uniform	56.7	57.4	58.1
5	Distance	57.1	58.0	59.2

Table 16: WSD Accuracy Rate Using Concept Embeddings Only, under Condition 3, for between One and Nine Iterations

was partially supported by JSPS KAKENHI Grant Number 15K16046 and research grant of Woman Empowerment Support System of Ibaraki University.

References

Kato, S., Asahara, M., and Yamazaki, M. (2017). Annotation of 'Word List by Semantic Principles' Information on 'Balanced Corpus of Contemporary Written Japanese'. In *Processing of NLP 2017*, pages 306–309 (In Japanese).

Kokuritsukokugokenkyusho. (1964). *Bunruigoihyo*. Shuuei Shuppan, In Japanese.

Komiya, K., Sasaki, Y., Miruta, H., Sasaki, M., Shinnou, H., and Kotani, Y. (2015). Surrounding Word Sense Model for Japanese All-words Word Sense Disambiguation. In *PACLIC 2015*, pages 35–43.

Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M., and Den, Y. (2014). Balanced Corpus of Contemporary Written Japanese. In *Language Resources and Evaluation*, volume 48, pages 345–371.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of ICLRWorkshop 2013*, pages 1–12.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of NIPS 2013*, pages 1–9.

Mikolov, T., tau Yih, W., and Zweig, G. (2013c). Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of NAACL 2013*, pages 746–751.

Nishio, M., Iwabuchi, E., and Mizutani, S. (1994). *Iwanami Kokugo Jiten Dai Go Han*. Iwanami Publisher, In Japanese.

Shinnou, H., Asahara, M., Komiya, K., and Sasaki, M. (2017a). Nwjc2vec: Word Embedding Data Constructed from NINJAL Web Japanese Corpus. *Journal of Natural Language Processing*, 24(4):705–720 (In Japanese).

Shinnou, H., Komiya, K., Sasaki, M., and Mori, S. (2017b). Japanese all-words WSD system using the Kyoto Text Analysis ToolKit. In *PACLIC 2017*, page no 11.

Vu, T. and Parker, D. S. (2016). K-Embeddings: Learning Conceptual Embeddings for Words using Context. In *Proceedings of NAACL-HLT 2016*, pages 1262–1267.

Enhancing Modern Supervised Word Sense Disambiguation Models by Semantic Lexical Resources

Stefano Melacci[1,2], **Achille Globo**[2], **Leonardo Rigutini**[2]
[1]DIISM - University of Siena, Via Roma 56 – 53100 Siena (Italy)
[2]QuestIT s.r.l., Via Leonida Cialfi 23 – 53100 Siena (Italy)
mela@diism.unisi.it, {globo, rigutini}@quest-it.com

Abstract

Supervised models for Word Sense Disambiguation (WSD) currently yield to state-of-the-art results in the most popular benchmarks. Despite the recent introduction of Word Embeddings and Recurrent Neural Networks to design powerful context-related features, the interest in improving WSD models using Semantic Lexical Resources (SLRs) is mostly restricted to knowledge-based approaches. In this paper, we enhance "modern" supervised WSD models exploiting two popular SLRs: WordNet and WordNet Domains. We propose an effective way to introduce semantic features into the classifiers, and we consider using the SLR structure to augment the training data. We study the effect of different types of semantic features, investigating their interaction with local contexts encoded by means of mixtures of Word Embeddings or Recurrent Neural Networks, and we extend the proposed model into a novel multi-layer architecture for WSD. A detailed experimental comparison in the recent Unified Evaluation Framework (Raganato et al., 2017) shows that the proposed approach leads to supervised models that compare favourably with the state-of-the art.

Keywords: Word Sense Disambiguation, Supervised Models, Lexical Resources

1. Introduction

Determining the correct meaning of a target word in a given context is a problem that is commonly referred to as Word Sense Disambiguation (WSD). WSD has a long tradition in the Natural Language Processing (NLP) community (Navigli, 2009), and it is still a very challenging task, subject to recent studies (Camacho-Collados et al., 2016; Raganato et al., 2017). The universe of WSD approaches is usually divided into the two main categories of "supervised" and "knowledge-based" methods (Raganato et al., 2017). The first category includes those algorithms that exploit manually annotated corpora (Zhong and Ng, 2010), and, recently, we also observe the growth of techniques that benefit from additional (semi) automatically annotated data (Taghipour and Ng, 2015a; Bovi et al., 2017) or that cast the learning problem into the semi-supervised setting (Taghipour and Ng, 2015b; Yuan et al., 2016). The second category considers those approaches that disambiguate words only using structured sources of lexical knowledge (Lesk, 1986; Moro et al., 2014; Agirre et al., 2014; Weissenborn et al., 2015). On one hand, knowledge-based systems are easier to setup, not requiring annotated corpora; on the other hand, supervised models currently yield to state-of-the-art results in the most popular benchmarks (Raganato et al., 2017).

Recent supervised WSD models are frequently based on the "It Makes Sense" (IMS) framework (Zhong and Ng, 2010), whose original implementation is also studied when augmented with Word Embeddings (WEs) to generate a distributed representations of the local context around the target word (Taghipour and Ng, 2015c; Rothe and Schütze, 2015). Other WSD approaches are based on Recurrent Neural Networks (RNNs) (Yuan et al., 2016; Melamud et al., 2016), such as the Context2Vec (C2V) model (Melamud et al., 2016), where a Bidirectional RNN is used to embed local contexts in a space where a supervised distance-based classifier performs the disambiguation.

WordNet (Princeton University, 2006) is commonly assumed to be the sense-repository for WSD. Each *word type* (lemma + Part of Speech (PoS)) is paired with one or more *senses*, and senses that express the same concept (among different word types) are grouped into a *synset*. Synsets participate to semantic/lexical relations, they contain definitions (*glosses*) and, in several cases, one or more *example sentences* (Miller, 1995). WordNet Domains (Fondazione Bruno Kessler, 2007) extends WordNet with ($\approx$ 200) semantic domain labels (Bentivogli et al., 2004).

Several WSD approaches exploit WordNet and WordNet Domains to implement *semantic features* (Magnini et al., 2001; Bakx et al., 2006; Bell and Patrick, 2004; Martinez, 2005; Kolte and Bhirud, 2008; Khapra et al., 2010; Navigli, 2012). The outcome of the feature extraction stage is often a representation of the local context of the target word based on domain vectors (Magnini et al., 2001; Bell and Patrick, 2004), collection of synsets (Martinez, 2005), and other domain-related statistics (Bakx et al., 2006; Kolte and Bhirud, 2008; Khapra et al., 2010; Navigli, 2012).

In this paper, we focus on the IMS framework and on "modern" implementations of the IMS features based on distributed representations. We propose an effective way to introduce SLR-based semantic features into the word-sense classifiers, and to possibly exploit the SLRs and their structure to augment the training data. We study the effect of different types of semantic features, and we extend the proposed models into a novel multi-layer architecture for WSD. To our best knowledge, the two aforementioned Semantic Lexical Resources (SLRs) have not been recently studied in conjunction with IMS models. A detailed experimental comparison in the Unified Evaluation Framework (Raganato et al., 2017) shows that the proposed approaches leads to supervised models that compare favourably with the state-of-the art.

2. Models

We are given a collection of labeled training sentences, in which, for a subset of the words, we are provided annotations on the disambiguated senses associated to them. Common supervised approaches train a classifier for each of the word types for which more than one of different sense annotations are found in the training sentences. When testing the classifiers on word types for which no-classifiers are available, a fall-back policy is employed, based on the WordNet First Sense (WFS). As a matter of fact, WordNet senses are ordered with respect to their estimated frequency, and the WFS is the most common sense for each word type. We focus on the IMS system (Zhong and Ng, 2010), where a linear Support Vector Machine (SVM) classifier is used to classify vectorial representations of the contexts around the target word. We indicate with $f_{wt}(\boldsymbol{x})$ the classifier associated to the word type wt, that outputs a decision over c_{wt} classes/senses. In this paper, we consider several implementations of the feature space to which $\boldsymbol{x}$ belongs. In particular, we partition $\boldsymbol{x}$ as follows

$$\boldsymbol{x} = \left[\overbrace{\underbrace{\boldsymbol{x}_{PoS}, \boldsymbol{x}_{locCol}, \boldsymbol{x}_{sWords}}, \boldsymbol{x}_{contextEmb}}^{\boldsymbol{x}_{IMS}}, \underbrace{\boldsymbol{x}_{sem}}_{\boldsymbol{x}_{IMSWE} \text{ or } \boldsymbol{x}_{IMSC2V}} \right], \quad (1)$$

where the comma represent the vector concatenation operator. Vector $\boldsymbol{x}_{IMS}$ corresponds to the default implementation of IMS, composed of PoS-tag features $\boldsymbol{x}_{PoS}$ (in a window of size 7 centered in the target word), local collocations $\boldsymbol{x}_{locCol}$ (11 local collocations), and binary indicators of the "surrounding words" in the current context $\boldsymbol{x}_{sWords}$ (the context is limited to the sentence to which the target word belong and, eventually, some other sentences before and after it), refer to (Zhong and Ng, 2010) for more details. A distributed representation of the context around the target word, $\boldsymbol{x}_{contextEmb}$, is implemented by means of weighted mixtures of Word Embeddings (WE) (we follow Iacobacci et al. (2016), and we use exponentially weighted sums of WEs, computed with Word2Vec (Mikolov et al., 2013)) or Bidirectional RNNs in the C2V approach (Melamud et al., 2016), leading to the input representations $\boldsymbol{x}_{IMSWE}$ and $\boldsymbol{x}_{IMSC2V}$, respectively. The WE model[1] and the C2V net[2] that we consider in this paper are trained in the English ukWaC corpus (Baroni et al., 2009), composed of up of two billion words (see the websites in the footnotes for all the details). Strictly speaking, including these features make the whole classifier semi-supervised, since we are actually using an additional, unlabeled resource (ukWaC). Under some circumstances, good results have been obtained by discarding $\boldsymbol{x}_{sWords}$ when $\boldsymbol{x}_{contextEmb}$ is present (Iacobacci et al., 2016). In our experience, this effect becomes evident when multiple training sentences are considered to build $\boldsymbol{x}_{sWords}$, while here we focus on a single sentence. We notice that, to our best knowledge, this way of including the C2V net into the IMS framework was not experimented before, while it makes the comparisons more uniform and, as we will show later, it leads to improved results.

We propose to compute also a set of semantic features $\boldsymbol{x}_{sem}$ to generate a more informative representation of the local context of the target word. To this extent, we make use of WordNet, extended with the information from WordNet Domains. We design three different types of semantic features, referred to as "PR", "sSyn", and "*Dom*", and we consider also every combination of them to generate and evaluate multiple instances of $\boldsymbol{x}_{sem}$. These features are computed on disambiguated words, so that prior disambiguation hypotheses are needed (we will shortly return on this point). In detail,

- "PR" models prior information, and it is a 1-hot encoding of the most likely sense for the target word in the current context.
- "sSyn" is the collection of "surrounding synsets" that appear in the context of the target word (i.e., in the same sentence). They are represented as binary indicators analogously to the surrounding words of IMS (Zhong and Ng, 2010), but, differently to them, they are a less variable representation of the context, since synonyms are represented with a single symbol.
- "*Dom*" encodes the 3 most frequent domains of the WordNet Domains taxonomy that are found in the context of the target word (they are encoded as binary features and in the case of "sSyn"), discarding the "factotum" domain. Domain information is known to help the WSD process (Magnini et al., 2001).

In order compute these features, we would need to have the use of other WSD models, while here we exploit the idea of using the WFS heuristic to disambiguate the context and compute priors on the target word. WFS has been show to be a competitive and hard-to-beat baseline (Raganato et al., 2017), even if its computational cost is almost negligible. As we will see shortly, the "PR" feature gains more importance in the multi-layer setting. In single-layer models, this feature is constant among all the training instances, and the SVM classifier will end-up in developing a bias term regularized by a squared penalty (differently from the usual unregularized SVM bias).

One of the crucial issues of supervised WSD is the lack of large collections of training examples. In this work we explore two ways of augmenting the training sets by means of WordNet. Word types belonging to same synset are synonyms, and this structure can be used to augment the training data in an efficient way: in order to train f_{wt}, we inherit all the training examples associated to other word types/senses sharing synsets with the target senses of wt. We talk about classifiers augmented using Synset Level Information (SIL). Another useful resource of training examples are WordNet glosses and example sentences. It is pretty trivial to automatically disambiguate most of the example sentences, since the word types that they describe are known. We notice that many knowledge-based systems make use of another linguistic resource, that is the Princeton WordNet Gloss Corpus, a collection of manually (and automatically) disambiguated definitions of the gloss data[3]. We investigate the use of this resource in supervised WSD.

[1] http://lcl.uniroma1.it/wsdeval/systems

[2] http://u.cs.biu.ac.il/~nlp/resources/downloads/context2vec/

[3] http://wordnet.princeton.edu/glosstag.shtml

Another direction we explore is when the semantic features are computed within a multi-layer architecture. In other words, we use the trained WSD model described so far to disambiguate the context of the target words and compute priors on their senses (a related idea was exploited, for example, in (Khapra et al., 2010)), then we update the semantic features and train another model. This procedure can be repeated multiple times. Formally, if we indicate with $\ell \geq 1$ the layer index, we have

$$f_{wt}^{\ell}\left(\boldsymbol{x}^{\ell}\right) = f_{wt}^{\ell}\left(\left[\boldsymbol{x}_{IMS*},\, \boldsymbol{x}_{sem}^{\ell-1}\right]\right) \ ,$$

where IMS$*$ is one of {IMS, IMSWE, IMSC2V}, $\boldsymbol{x}_{sem}^{\ell-1}$ is computed using the disambiguation hypotheses given by $f_{wt}^{\ell-1}$, and $\boldsymbol{x}_{sem}^{0}$ is based on the WFS. This architecture must be trained in a layer-wise fashion (train layer ℓ, disambiguate training data, train layer $\ell+1$, etc.), but we also explore the simpler case in which we only train f_{wt}^{1} and, at test time, we iteratively apply it in a multi-layer scheme.

3. Experimental Results

We compare three main models, IMS, IMSWE, IMSC2V, respectively (based on the related input representations of Eq. 1), with and without semantic features, in the recently proposed Unified Evaluation Framework (Raganato et al., 2017), that includes 5 popular benchmarks (Senseval2, Senseval3, SemEval2007, SemEval2013, SemEval2015), and the concatenation of the sentences of all of them (ALL). Such framework provides code for computing the F1 score of the models, together with carefully processed training and test data, using WordNet 3.0 as sense repository. The training data consists in the SemCor corpus, the sense-tagged corpus created by the WordNet Project research team. We consider 5 categories of experiments, focussing on different topics (in bold).

Semantic features. The proposed features have different effects in the three studied systems, as reported in Table 1. They almost always provide improvements over the baseline approaches, even if in different configurations. In particular, in the IMS case, semantic features lead to more evident benefits w.r.t. what happens in the IMSWE and IMSC2V cases, since WEs and C2V already capture several word regularities. We also notice that the use of domain-related information seems to carry the most useful information for the WSD task. On average, the joint use of all the semantic features (+sSyn+PR+*Dom*) provides good improvements over all the three systems. Comparing our results with the best results (considering several supervised and knowledge-based systems) collected in (Raganato et al., 2017), our approach leads to state-of-the-art results both in the cases of IMSWE and IMSC2V. It is interesting to evaluate that the C2V features applied in the IMS framework lead to better results than in the original distance-based approach proposed in (Melamud et al., 2016) and evaluated also in (Raganato et al., 2017). Table 2 provides a detail of the experimental results considering different word classes (ALL benchmark). Semantic features allows the models to gain improvements when disambiguating adjectives and, more importantly, when disambiguating verbs, that are the most polysemous elements. This more evident in the case of IMSWE, confirming that semantic features introduce useful information that is not captured by WEs.

	Senseval2	Senseval3	SemEval2007	SemEval2013	SemEval2015	ALL
IMS	70.2	68.8	**62.2**	65.3	69.3	68.1
+PR	70.4	68.8	**62.2**	65.1	69.4	68.2
+sSyn	70.2	69.4	61.8	65.0	69.2	68.1
+sSyn+PR	70.1	69.3	61.8	65.0	69.5	68.1
+*Dom*	70.7	**69.8**	60.7	**65.4**	69.6	**68.5**
+PR+*Dom*	70.6	69.7	60.7	65.0	69.6	68.3
+sSyn+*Dom*	**70.8**	69.7	60.9	**65.4**	69.5	**68.5**
+sSyn+PR+*Dom*	**70.8**	69.7	60.9	65.0	**69.8**	68.4
IMSWE	72.2	69.9	62.9	**66.2**	71.9	69.6
+PR	72.0	69.7	62.9	66.1	72.0	69.5
+sSyn	72.5	70.1	62.6	66.1	71.9	69.7
+sSyn+PR	72.5	70.1	62.9	66.1	**72.2**	69.8
+*Dom*	**72.7**	**70.3**	62.9	66.1	72.0	**69.9**
+PR+*Dom*	72.5	70.0	**63.3**	66.0	72.1	69.8
+sSyn+*Dom*	72.6	70.2	**63.3**	66.0	71.8	69.8
+sSyn+PR+*Dom*	**72.7**	70.1	**63.3**	66.1	71.9	69.8
IMSC2V	73.8	**71.9**	63.3	68.1	72.7	71.1
+PR	73.8	**71.9**	63.3	**68.2**	**72.8**	**71.3**
+sSyn	**74.2**	71.8	63.5	68.1	**72.8**	**71.3**
+sSyn+PR	74.1	71.6	63.3	68.1	**72.8**	**71.3**
+*Dom*	73.9	71.8	63.7	68.0	72.3	71.2
+PR+*Dom*	73.9	**71.9**	63.5	67.9	72.7	71.2
+sSyn+*Dom*	74.0	71.8	**64.0**	67.9	72.5	71.2
+sSyn+PR+*Dom*	74.0	71.8	63.7	67.9	72.6	71.2
(Raganato et al., 2017)	*72.2*	*70.4*	*62.6*	*67.3*	*71.9*	*69.6*

Table 1: F1 score of IMS, IMSWE, IMSC2V, and their semantic-feature-based variants. The last row collects the best results (over several models) in (Raganato et al., 2017).

Bias toward WFS. We investigated the tendency of the WSD system to be biased toward the most frequent sense in the training data (Postma et al., 2016), that, in our experimental setting, is very similar to the WFS. Table 3 indicates the percentage of correctly disambiguated instances for which the correct sense is *not* the WFS. The result shows that adding the PR feature does not produce significant changes. When considering all the semantic features we observe slight variations, mostly concentrated in the IMSC2V case. Since IMSC2V is the less-biased model, this stronger effect was expected. Interestingly, in the case of IMS, the best combination of semantic features leads to the best results and to a less-biased classifier.

Augmenting the training data. A key point to evaluate is whether the semantic features can benefit by the augmentation procedures described in Section 2 (SLI, glosses, example sentences). Table 4 shows that using the synset-based procedure (SLI), paired with semantic features, improves the classifiers, while discarding such features results in reduced scores. This is explained by the fact that the semantic features are more robust and corse-grained

	Noun	Adj	Verb	Adv
IMS	70.0	75.2	56.0	**83.2**
+PR	70.0	75.3	56.0	**83.2**
+sSyn	69.9	75.5	**56.3**	82.7
+sSyn+PR	70.0	75.4	56.1	82.7
+*Dom*	**70.4**	**76.3**	55.9	82.9
+PR+*Dom*	70.3	**76.3**	55.6	82.9
+sSyn+*Dom*	**70.4**	76.1	56.1	82.9
+sSyn+PR+*Dom*	70.3	**76.3**	55.9	**83.2**
IMSWE	71.8	76.1	57.4	**83.5**
+PR	71.7	75.9	57.3	**83.5**
+sSyn	71.9	76.2	57.6	83.2
+sSyn+PR	71.9	76.3	57.7	83.2
+*Dom*	**72.0**	76.4	**57.8**	83.2
+PR+*Dom*	71.8	76.4	**57.8**	83.2
+sSyn+*Dom*	71.8	**76.5**	**57.8**	83.2
+sSyn+PR+*Dom*	71.8	**76.5**	**57.8**	83.2
IMSC2V	**73.1**	77.0	60.5	83.5
+PR	**73.1**	77.1	**60.6**	83.5
+sSyn	**73.1**	77.5	60.4	**83.8**
+sSyn+PR	**73.1**	77.3	60.2	**83.8**
+*Dom*	**73.1**	77.3	60.0	83.5
+PR+*Dom*	**73.1**	77.4	60.2	83.5
+sSyn+*Dom*	73.0	**77.6**	60.4	**83.8**
+sSyn+PR+*Dom*	73.0	77.4	60.5	83.5
(Raganato et al., 2017)	*72.0*	*77.2*	*57.6*	*84.7*

Table 2: F1 score restricted to specific PoS ("ALL" benchmark). The last row is the best results (over several different models) in (Raganato et al., 2017).

	Plain	+PR	+sSyn+PR+*Dom*	Best of Tab. 1
IMS	9.4	9.2	9.4	**9.7**
IMSWE	**11.0**	**11.0**	10.8	10.8
IMSC2V	**13.3**	**13.3**	12.7	12.8

Table 3: The % of correct disambiguations where the right sense is NOT the WFS ("ALL"). Last column refers to the best variants in Table 1 (+sSyn+*Dom*,+*Dom*,+sSyn, resp.).

than the other standard IMS features, and they can handle the larger variability (and some noise) in the training data that is introduced by the augmentation procedure. A more evident improvement can be observed when introducing WordNet glosses and examples into the training set (Fig 1). Models equipped with semantic features constantly benefit from such data. Unsupervised example sentences (Fig 1(a)) cause improvements that overcome the results of Table 1. When disambiguated glosses are used (Fig 1(b-c)), the improvements are more evident, since we are actually exploiting new manually annotated data. We can also train classifiers for new word types, or expand the number of senses covered by the already existing classifiers (Fig 1(c)).

Multi-Layer architectures. Fig 2 compares multi-layer models (up to 4 layers), distinguishing among models with/without augmented training sets (SLI + example sentences, i.e., no manual annotations), models that are trained in a layer-wise fashion, and those that are trained only once and tested by executing them over multiple layers. The last-mentioned models can only slightly benefit from the multi-layer execution, while the layer-wise training procedure leads to better models (w.r.t. to the single-layer case). Augmenting the training data helps such training procedure. IMSC2V is not improved by the multi-layer setting, mostly because the C2V features are very informative and single layer architectures are already powerful.

	Plain		+sSyn+PR+*Dom*	
	w/o SLI	w/ SLI	w/o SLI	w/ SLI
IMS	68.1	66.8	68.4	**68.6**
IMSWE	69.6	68.7	**69.8**	69.7
IMSC2V	71.2	70.8	71.2	**71.6**

Table 4: Augmenting the training data by Synset Level Information (SLI), for plain models and models equipped with all the semantic features (F1 score, "ALL" bench).

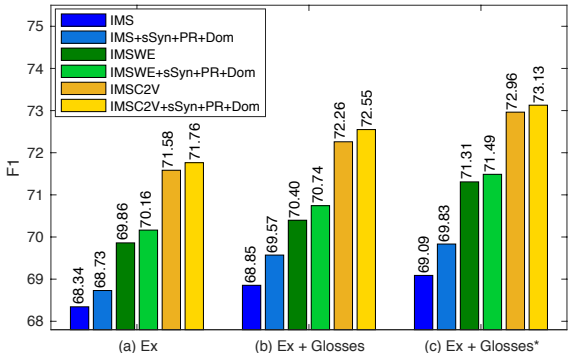


Figure 1: Training set augmentation using examples sentences only (a) and also manually disambiguated glosses (b,c), on plain models and on models equipped with all the semantic features. While (a) and (b) are limited to the word-types/senses defined in the SemCor training set, (c) considers a superset of them, using all the available annotations.

4. Conclusions and Future Work

We proposed to augment recent WSD models based on the IMS framework with semantic features from popular Semantic Lexical Resources: WordNet and WordNet Domains. Our deep experimental comparison shows that these features can be paired with context representations based on

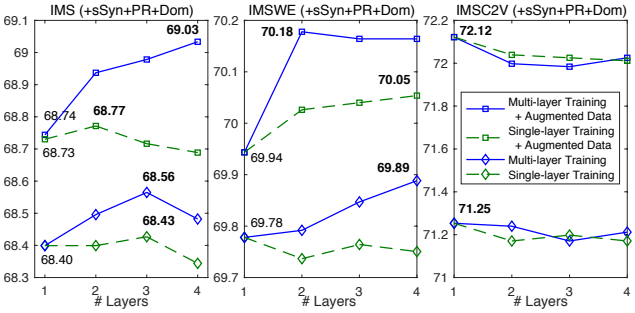


Figure 2: Multi-layer models with all the semantic features ("ALL" benchmark, F1 score): trained "layer-wise" (blue) or trained as single-layer models and iteratively ran over multiple-layers (green). Models with augmented training data exploit SLI and example sentences from WordNet.

Word Embeddings and Recurrent Neural Networks, generating more robust models. WordNet organization and data (glosses and example sentences) were used to augment the training set, showing that semantic features play a crucial role to gain enhancements. Finally, the same features are implemented into a multi-layer architecture that can improve the WSD models. Overall, we reached results that compare favourably with the state-of-the art. Future work will consider more specific semi-supervised approaches and the use of automatically annotated resources.

5. Bibliographical References

Agirre, E., de Lacalle, O. L., and Soroa, A. (2014). Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–84.

Bakx, G. E., Villodre, L., and Claramunt, G. (2006). *Machine learning techniques for word sense disambiguation*. Ph.D. thesis, Universitat Politecnica de Catalunya.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Bell, D. and Patrick, J. (2004). Using wordnet domains in a supervised learning word sense disambiguation system. In *Proceedings of the Australasian Language Technology Workshop 2004*, pages 17–24.

Bentivogli, L., Forner, P., Magnini, B., and Pianta, E. (2004). Revising the wordnet domains hierarchy: semantics, coverage and balancing. In *Proceedings of the Workshop on Multilingual Linguistic Resources*, pages 101–108. Association for Computational Linguistics.

Bovi, C. D., Camacho-Collados, J., Raganato, A., and Navigli, R. (2017). Eurosense: Automatic harvesting of multilingual sense annotations from parallel text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 594–600.

Camacho-Collados, J., Bovi, C. D., Raganato, A., and Navigli, R. (2016). A large-scale multilingual disambiguation of glosses. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2016). Embeddings for word sense disambiguation: An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 897–907, Berlin, Germany, August. Association for Computational Linguistics.

Khapra, M. M., Shah, S., Kedia, P., and Bhattacharyya, P. (2010). Domain-specific word sense disambiguation combining corpus based and wordnet based parameters. In *In 5th International Conference on Global Wordnet (GWC2010)*.

Kolte, S. G. and Bhirud, S. G. (2008). Word sense disambiguation using wordnet domains. In *Emerging Trends in Engineering and Technology, 2008. ICETET'08. First International Conference on*, pages 1187–1191. IEEE.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM.

Magnini, B., Strapparava, C., Pezzulo, G., and Gliozzo, A. (2001). Using domain information for word sense disambiguation. In *The Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 111–114. Association for Computational Linguistics.

Martinez, D. (2005). Supervised word sense disambiguation: Facing current challenges. *Procesamiento del Lenguaje Natural*, 34.

Melamud, O., Goldberger, J., and Dagan, I. (2016). context2vec: Learning generic context embedding with bidirectional lstm. In *CoNLL*, pages 51–61.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Navigli, R. (2012). A quick tour of word sense disambiguation, induction and related approaches. *SOFSEM 2012: Theory and practice of computer science*, pages 115–129.

Postma, M., Izquierdo, R., Agirre, E., Rigau, G., and Vossen, P. (2016). Addressing the mfs bias in wsd systems. In *LREC*.

Raganato, A., Camacho-Collados, J., and Navigli, R. (2017). Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proc. of EACL*, pages 99–110.

Rothe, S. and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of ACL*, pages 1793–1803.

Taghipour, K. and Ng, H. T. (2015a). One million sense-tagged instances for word sense disambiguation and induction. In *CoNLL*, pages 338–344.

Taghipour, K. and Ng, H. T. (2015b). Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *HLT-NAACL*, pages 314–323.

Taghipour, K. and Ng, H. T. (2015c). Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *HLT-NAACL*, pages 314–323.

Weissenborn, D., Hennig, L., Xu, F., and Uszkoreit, H. (2015). Multi-objective optimization for the joint disambiguation of nouns and named entities. In *ACL (1)*, pages 596–605.

Yuan, D., Richardson, J., Doherty, R., Evans, C., and Altendorf, E. (2016). Semi-supervised word sense disambiguation with neural models. In *COLING 2016*.

Zhong, Z. and Ng, H. T. (2010). It makes sense: A wide-

coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83. Association for Computational Linguistics.

6. Language Resource References

Fondazione Bruno Kessler. (2007). *Labelling of WordNet 1.6 with semantic fields (WordNet Domains)*. 3.2, ISLRN 103-401-284-171-1.

Princeton University. (2006). *WordNet*. 3.0, ISLRN 379-473-059-273-1.

An Unsupervised Word Sense Disambiguation System for Under-Resourced Languages

Dmitry Ustalov[*†], **Denis Teslenko**[†], **Alexander Panchenko**[‡], **Mikhail Chernoskutov**[†], **Chris Biemann**[‡], **Simone Paolo Ponzetto**[*]

[*] Data and Web Science Group, University of Mannheim, Germany

[†] Ural Federal University, Russia

[‡] Universität Hamburg, Department of Informatics, Language Technology Group, Germany

{dmitry,simone}@informatik.uni-mannheim.de, teslenkoden@gmail.com,

mikhail.chernoskutov@urfu.ru, {panchenko,biemann}@informatik.uni-hamburg.de

Abstract

In this paper, we present Watasense, an unsupervised system for word sense disambiguation. Given a sentence, the system chooses the most relevant sense of each input word with respect to the semantic similarity between the given sentence and the synset constituting the sense of the target word. Watasense has two modes of operation. The sparse mode uses the traditional vector space model to estimate the most similar word sense corresponding to its context. The dense mode, instead, uses synset embeddings to cope with the sparsity problem. We describe the architecture of the present system and also conduct its evaluation on three different lexical semantic resources for Russian. We found that the dense mode substantially outperforms the sparse one on all datasets according to the adjusted Rand index.

Keywords: word sense disambiguation, system, synset induction

1. Introduction

Word sense disambiguation (WSD) is a natural language processing task of identifying the particular word senses of polysemous words used in a sentence. Recently, a lot of attention was paid to the problem of WSD for the Russian language (Lopukhin and Lopukhina, 2016; Lopukhin et al., 2017; Ustalov et al., 2017). This problem is especially difficult because of both linguistic issues – namely, the rich morphology of Russian and other Slavic languages in general – and technical challenges like the lack of software and language resources required for addressing the problem.

To address these issues, we present Watasense, an unsupervised system for word sense disambiguation. We describe its architecture and conduct an evaluation on three datasets for Russian. The choice of an unsupervised system is motivated by the absence of resources that would enable a supervised system for under-resourced languages. Watasense is not strictly tied to the Russian language and can be applied to any language for which a tokenizer, part-of-speech tagger, lemmatizer, and a sense inventory are available.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents the Watasense word sense disambiguation system, presents its architecture, and describes the unsupervised word sense disambiguation methods bundled with it. Section 4 evaluates the system on a gold standard for Russian. Section 5 concludes with final remarks.

2. Related Work

Although the problem of WSD has been addressed in many SemEval campaigns (Navigli et al., 2007; Agirre et al., 2010; Manandhar et al., 2010, *inter alia*), we focus here on word sense disambiguation *systems* rather than on the research methodologies.

Among the freely available systems, IMS ("It Makes Sense") is a supervised WSD system designed initially for the English language (Zhong and Ng, 2010). The system uses a support vector machine classifier to infer the particular sense of a word in the sentence given its contextual sentence-level features. Pywsd is an implementation of several popular WSD algorithms implemented in a library for the Python programming language.[1] It offers both the classical Lesk algorithm for WSD and path-based algorithms that heavily use the WordNet and similar lexical ontologies. DKPro WSD (Miller et al., 2013) is a general-purpose framework for WSD that uses a lexical ontology as the sense inventory and offers the variety of WordNet-based algorithms. Babelfy (Moro et al., 2014) is a WSD system that uses BabelNet, a large-scale multilingual lexical ontology available for most natural languages. Due to the broad coverage of BabelNet, Babelfy offers entity linking as part of the WSD functionality.

Panchenko et al. (2017b) present an unsupervised WSD system that is also knowledge-free: its sense inventory is induced based on the JoBimText framework, and disambiguation is performed by computing the semantic similarity between the context and the candidate senses (Biemann and Riedl, 2013). Pelevina et al. (2016) proposed a similar approach to WSD, but based on dense vector representations (word embeddings), called SenseGram. Similarly to SenseGram, our WSD system is based on averaging of word embeddings on the basis of an automatically induced sense inventory. A crucial difference, however, is that we induce our sense inventory from synonymy dictionaries and not distributional word vectors. While this requires more manually created resources, a potential advantage of our approach is that the resulting inventory contains less noise.

3. Watasense, an Unsupervised System for Word Sense Disambiguation

Watasense is implemented in the Python programming language using the scikit-learn (Pedregosa and others,

[1] https://github.com/alvations/pywsd

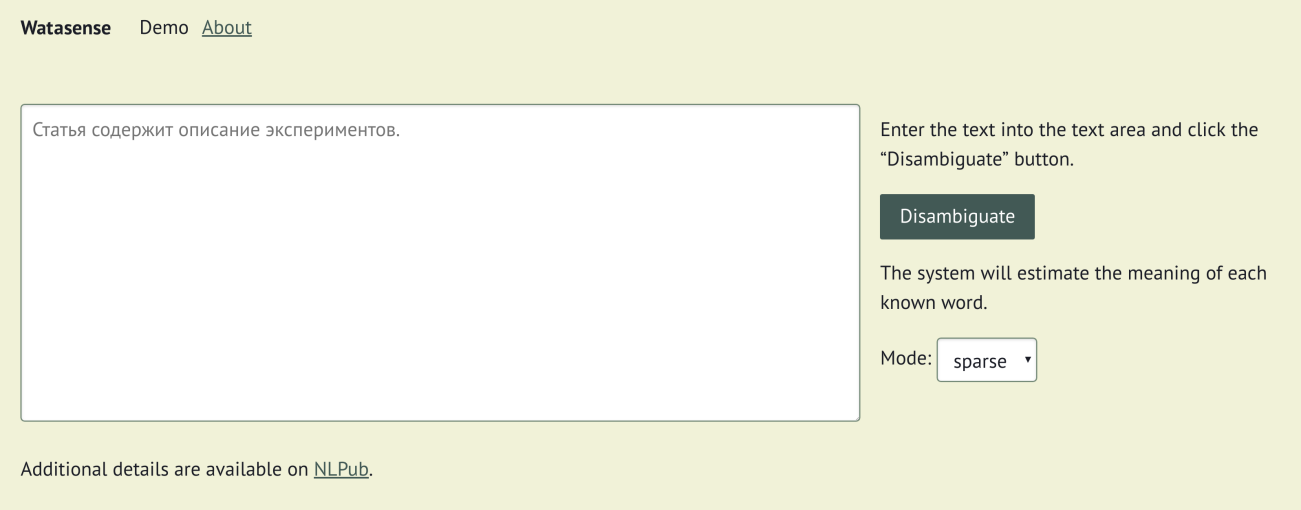


Figure 1: A snapshot of the online demo, which is available at http://watasense.nlpub.org/ (in Russian).

2011) and Gensim (Řehuřek and Sojka, 2010) libraries. Watasense offers a Web interface (Figure 1), a command-line tool, and an application programming interface (API) for deployment within other applications.

3.1. System Architecture

A sentence is represented as a list of *spans*. A span is a quadruple: (w, p, l, i), where w is the word or the token, p is the part of speech tag, l is the lemma, i is the position of the word in the sentence. These data are provided by tokenizer, part-of-speech tagger, and lemmatizer that are specific for the given language. The WSD results are represented as a map of spans to the corresponding word sense identifiers.

The sense inventory is a list of synsets. A synset is represented by three bag of words: the synonyms, the hypernyms, and the union of two former – the *bag*. Due to the performance reasons, on initialization, an inverted index is constructed to map a word to the set of synsets it is included into.

Each word sense disambiguation method extends the `BaseWSD` class. This class provides the end user with a generic interface for WSD and also encapsulates common routines for data pre-processing. The inherited classes like `SparseWSD` and `DenseWSD` should implement the `disambiguate_word(...)` method that disambiguates the given word in the given sentence. Both classes use the *bag* representation of synsets on the initialization. As the result, for WSD, not just the synonyms are used, but also the hypernyms corresponding to the synsets. The UML class diagram is presented in Figure 2.

Watasense supports two sources of word vectors: it can either read the word vector dataset in the binary Word2Vec format or use Word2Vec-Pyro4, a general-purpose word vector server.[2] The use of a remote word vector server is recommended due to the reduction of memory footprint per each Watasense process.

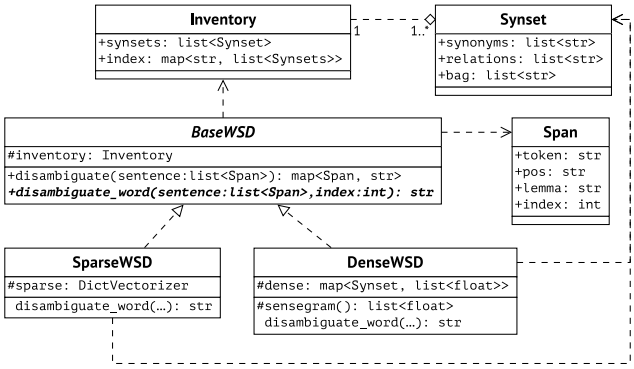


Figure 2: The UML class diagram of Watasense.

3.2. User Interface

Figure 1 shows the Web interface of Watasense. It is composed of two primary activities. The first is the text input and the method selection (Figure 1). The second is the display of the disambiguation results with part of speech highlighting (Figure 3). Those words with resolved polysemy are underlined; the tooltips with the details are raised on hover.

3.3. Word Sense Disambiguation

We use two different unsupervised approaches for word sense disambiguation. The first, called 'sparse model', uses a straightforward sparse vector space model, as widely used in Information Retrieval, to represent contexts and synsets. The second, called 'dense model', represents synsets and contexts in a dense, low-dimensional space by averaging word embeddings.

Sparse Model. In the vector space model approach, we follow the sparse context-based disambiguated method (Faralli et al., 2016; Panchenko et al., 2017b). For estimating the sense of the word w in a sentence, we search for such a synset $\hat{w}$ that maximizes the cosine similarity to the sentence vector:

$$\hat{w} = \arg\max_{S \ni w} \cos(S, T), \quad (1)$$

[2] https://github.com/nlpub/word2vec-pyro4

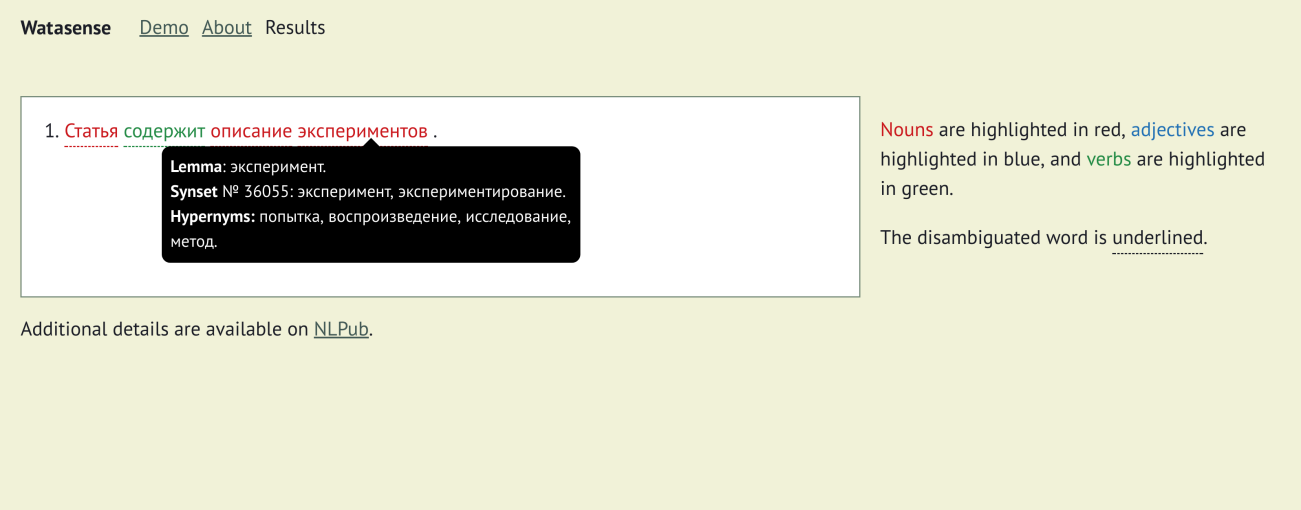


Figure 3: The word sense disambiguation results with the word "experiments" selected. The tooltip shows its lemma "experiment", the synset identifier (36055), and the words forming the synset "experiment", "experimenting" as well as its hypernyms "attempt", "reproduction", "research", "method".

where S is the set of words forming the synset, T is the set of words forming the sentence. On initialization, the synsets represented in the sense inventory are transformed into the tf–idf-weighted word-synset sparse matrix efficiently represented in the memory using the compressed sparse row format. Given a sentence, a similar transformation is done to obtain the sparse vector representation of the sentence in the same space as the word-synset matrix. Then, for each word to disambiguate, we retrieve the synset containing this word that maximizes the cosine similarity between the sparse sentence vector and the sparse synset vector. Let $w_{\max}$ be the maximal number of synsets containing a word and $S_{\max}$ be the maximal size of a synset. Therefore, disambiguation of the whole sentence T requires $O(|T| \times w_{\max} \times S_{\max})$ operations using the efficient sparse matrix representation.

Dense Model. In the synset embeddings model approach, we follow SenseGram (Pelevina et al., 2016) and apply it to the synsets induced from a graph of synonyms. We transform every synset into its dense vector representation by averaging the word embeddings corresponding to each constituent word:

$$\vec{S} = \frac{1}{|S|} \sum_{w \in S} \vec{w}, \tag{2}$$

where $\vec{w}$ denotes the word embedding of w. We do the same transformation for the sentence vectors. Then, given a word w, a sentence T, we find the synset $\hat{w}$ that maximizes the cosine similarity to the sentence:

$$\hat{w} = \arg\max_{S \ni w} \cos\left(\frac{\sum_{u \in S} \vec{u}}{|S|}, \frac{\sum_{u \in T} \vec{u}}{|T|}\right). \tag{3}$$

On initialization, we pre-compute the dense synset vectors by averaging the corresponding word embeddings. Given a sentence, we similarly compute the dense sentence vector by averaging the vectors of the words belonging to non-auxiliary parts of speech, i.e., nouns, adjectives, adverbs, verbs, etc. Then, given a word to disambiguate, we retrieve the synset that maximizes the cosine similarity between the dense sentence vector and the dense synset vector. Thus, given the number of dimensions d, disambiguation of the whole sentence T requires $(|T| \times w_{\max} \times d)$ operations.

4. Evaluation

We conduct our experiments using the evaluation methodology of SemEval 2010 Task 14: Word Sense Induction & Disambiguation (Manandhar et al., 2010). In the gold standard, each word is provided with a set of instances, i.e., the sentences containing the word. Each instance is manually annotated with the single sense identifier according to a pre-defined sense inventory. Each participating system estimates the sense labels for these ambiguous words, which can be viewed as a clustering of instances, according to sense labels. The system's clustering is compared to the gold-standard clustering for evaluation.

4.1. Quality Measure

The original SemEval 2010 Task 14 used the V-Measure external clustering measure (Manandhar et al., 2010). However, this measure is maximized by clustering each sentence into his own distinct cluster, i.e., a 'dummy' singleton baseline. This is achieved by the system deciding that every ambiguous word in every sentence corresponds to a different word sense. To cope with this issue, we follow a similar study (Lopukhin et al., 2017) and use instead of the adjusted Rand index (ARI) proposed by Hubert and Arabie (1985) as an evaluation measure.

In order to provide the overall value of ARI, we follow the addition approach used in (Lopukhin et al., 2017). Since the quality measure is computed for each lemma individually, the total value is a weighted sum, namely

$$\mathrm{ARI} = \frac{1}{\sum_w |I(w)|} \sum_w \mathrm{ARI}_w \times |I(w)|\,, \tag{4}$$

where w is the lemma, $I(w)$ is the set of the instances for the lemma w, ARI_w is the adjusted Rand index computed

for the lemma w. Thus, the contribution of each lemma to the total score is proportional to the number of instances of this lemma.

4.2. Dataset

We evaluate the word sense disambiguation methods in Watasense against three baselines: an unsupervised approach for learning multi-prototype word embeddings called AdaGram (Bartunov et al., 2016), same sense for all the instances per lemma (One), and one sense per instance (Singletons). The AdaGram model is trained on the combination of RuWac, Lib.Ru, and the Russian Wikipedia with the overall vocabulary size of 2 billion tokens (Lopukhin et al., 2017).

As the gold-standard dataset, we use the WSD training dataset for Russian created during RUSSE'2018: A Shared Task on Word Sense Induction and Disambiguation for the Russian Language (Panchenko et al., 2018). The dataset has 31 words covered by 3 491 instances in the *bts-rnc* subset and 5 words covered by 439 instances in the *wiki-wiki* subset.[3]

The following different sense inventories have been used during the evaluation:

- **WATLINK**, a word sense network constructed automatically. It uses the synsets induced in an unsupervised way by the WATSET[CW_{nolog}, MCL] method (Ustalov et al., 2017) and the semantic relations from such dictionaries as Wiktionary referred as *Joint+Exp+SWN* in Ustalov (2017). This is the only automatically built inventory we use in the evaluation.
- **RuThes**, a large-scale lexical ontology for Russian created by a group of expert lexicographers (Loukachevitch, 2011).[4]
- **RuWordNet**, a semi-automatic conversion of the RuThes lexical ontology into a WordNet-like structure (Loukachevitch et al., 2016).[5]

Since the *Dense* model requires word embeddings, we used the 500-dimensional word vectors from the Russian Distributional Thesaurus (Panchenko et al., 2017a).[6] These vectors are obtained using the Skip-gram approach trained on the `lib.rus.ec` text corpus.

4.3. Results

We compare the evaluation results obtained for the *Sparse* and *Dense* approaches with three baselines: the AdaGram model (AdaGram), the same sense for all the instances per lemma (One) and one sense per instance (Singletons). The evaluation results are presented in Table 1. The columns bts-rnc and wiki-wiki represent the overall value of ARI according to Equation (4). The column Avg. consists of the weighted average of the datasets w.r.t. the number of instances.

We observe that the SenseGram-based approach for word sense disambiguation yields substantially better results in every case (Table 1). The primary reason for that is the implicit handling of similar words due to the averaging of dense word vectors for semantically related words. Thus, we recommend using the dense approach in further studies. Although the AdaGram approach trained on a large text corpus showed better results according to the weighted average, this result does not transfer to languages with less available corpus size.

Table 1: Results on RUSSE'2018 (Adjusted Rand Index).

Method		bts-rnc	wiki-wiki	Avg.
AdaGram		0.22	0.39	0.23
WATLINK	*Sparse*	0.01	0.07	0.01
	Dense	0.08	0.14	0.08
RuThes	*Sparse*	0.00	0.17	0.01
	Dense	0.14	0.47	0.17
RuWordNet	*Sparse*	0.00	0.11	0.01
	Dense	0.12	0.50	0.15
One		0.00	0.00	0.00
Singletons		0.00	0.00	0.00

5. Conclusion

In this paper, we presented Watasense,[7] an open source unsupervised word sense disambiguation system that is parameterized only by a word sense inventory. It supports both sparse and dense sense representations. We were able to show that the dense approach substantially boosts the performance of the sparse approach on three different sense inventories for Russian. We recommend using the dense approach in further studies due to its smoothing capabilities that reduce sparseness. In further studies, we will look at the problem of phrase neighbors that influence the sentence vector representations.

Finally, we would like to emphasize the fact that Watasense has a simple API for integrating different algorithms for WSD. At the same time, it requires only a basic set of language processing tools to be available: tokenizer, a part-of-speech tagger, lemmatizer, and a sense inventory, which means that low-resourced language can benefit of its usage.

6. Acknowledgements

We acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) under the project "Joining Ontologies and Semantics Induced from Text" (JOIN-T), the RFBR under the projects no. 16-37-00203 mol_a and no. 16-37-00354 mol_a, and the RFH under the project no. 16-04-12019. The research was supported by the Ministry of Education and Science of the Russian Federation Agreement no. 02.A03.21.0006. The calculations were carried out using the supercomputer "Uran" at the Krasovskii Institute of Mathematics and Mechanics.

7. Bibliographical References

Agirre, E., de Lacalle, O. L., Fellbaum, C., Hsieh, S.-K., Tesconi, M., Monachini, M., Vossen, P., and Segers, R. (2010). SemEval-2010 Task 17: All-words Word Sense Disambiguation on a Specific Domain. In *Proceedings*

[3] http://russe.nlpub.org/2018/wsi/

[4] http://www.labinform.ru/pub/ruthes/index_eng.htm

[5] http://www.labinform.ru/pub/ruwordnet/index_eng.htm

[6] https://doi.org/10.5281/zenodo.400631

[7] https://github.com/nlpub/watasense

of the 5th International Workshop on Semantic Evaluation, SemEval '10, pages 75–80, Los Angeles, CA, USA. Association for Computational Linguistics.

Bartunov, S., Kondrashkin, D., Osokin, A., and Vetrov, D. P. (2016). Breaking Sticks and Ambiguities with Adaptive Skip-gram. *Journal of Machine Learning Research*, 51:130–138.

Biemann, C. and Riedl, M. (2013). Text: now in 2D! A framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Faralli, S., Panchenko, A., Biemann, C., and Ponzetto, S. P. (2016). Linked Disambiguated Distributional Semantic Networks. In *The Semantic Web – ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part II*, pages 56–64, Cham, Germany. Springer International Publishing.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

Lopukhin, K. A. and Lopukhina, A. A. (2016). Word Sense Disambiguation for Russian Verbs Using Semantic Vectors and Dictionary Entries. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual conference "Dialogue"*, pages 393–404, Moscow, Russia. RSUH.

Lopukhin, K. A., Iomdin, B. L., and Lopukhina, A. A. (2017). Word Sense Induction for Russian: Deep Study and Comparison with Dictionaries. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual conference "Dialogue". Volume 1 of 2. Computational Linguistics: Practical Applications*, pages 121–134, Moscow, Russia. RSUH.

Loukachevitch, N. V., Lashevich, G., Gerasimova, A. A., Ivanov, V. V., and Dobrov, B. V. (2016). Creating Russian WordNet by Conversion. In *Computational Linguistics and Intellectual Technologies: papers from the Annual conference "Dialogue"*, pages 405–415, Moscow, Russia. RSUH.

Loukachevitch, N. V. (2011). *Thesauri in information retrieval tasks*. Moscow University Press, Moscow, Russia. In Russian.

Manandhar, S., Klapaftis, I., Dligach, D., and Pradhan, S. (2010). SemEval-2010 Task 14: Word Sense Induction & Disambiguation. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 63–68, Uppsala, Sweden. Association for Computational Linguistics.

Miller, T., Erbs, N., Zorn, H.-P., Zesch, T., and Gurevych, I. (2013). DKPro WSD: A Generalized UIMA-based Framework for Word Sense Disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Sofia, Bulgaria.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity Linking meets Word Sense Disambiguation: A Unified Approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). SemEval-2007 Task 07: Coarse-Grained English All-Words Task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 30–35, Prague, Czech Republic. Association for Computational Linguistics.

Panchenko, A., Ustalov, D., Arefyev, N., Paperno, D., Konstantinova, N., Loukachevitch, N., and Biemann, C., (2017a). *Human and Machine Judgements for Russian Semantic Relatedness*, pages 221–235. Springer International Publishing, Cham, Germany.

Panchenko, A., Marten, F., Ruppert, E., Faralli, S., Ustalov, D., Ponzetto, S. P., and Biemann, C. (2017b). Unsupervised, Knowledge-Free, and Interpretable Word Sense Disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 91–96, Copenhagen, Denmark. Association for Computational Linguistics.

Panchenko, A., Lopukhina, A., Ustalov, D., Lopukhin, K., Leontyev, A., Arefyev, N., and Loukachevitch, N. (2018). RUSSE'2018: A Shared Task on Word Sense Induction and Disambiguation for the Russian Language. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual conference "Dialogue"*, Moscow, Russia. RSUH.

Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pelevina, M., Arefiev, N., Biemann, C., and Panchenko, A. (2016). Making Sense of Word Embeddings. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 174–183, Berlin, Germany. Association for Computational Linguistics.

Ustalov, D., Panchenko, A., and Biemann, C. (2017). Watset: Automatic Induction of Synsets from a Graph of Synonyms. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1579–1590, Vancouver, Canada. Association for Computational Linguistics.

Ustalov, D. (2017). Expanding Hierarchical Contexts for Constructing a Semantic Word Network. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual conference "Dialogue". Volume 1 of 2. Computational Linguistics: Practical Applications*, pages 369–381, Moscow, Russia. RSUH.

Řehuřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *New Challenges for NLP Frameworks Programme: A workshop at LREC 2010*, pages 51–55, Valetta, Malta. European Language Resources Association (ELRA).

Zhong, Z. and Ng, H. T. (2010). It Makes Sense: A Wide-Coverage Word Sense Disambiguation System for Free Text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83, Uppsala, Sweden. Association for Computational Linguistics.

Unsupervised Korean Word Sense Disambiguation using CoreNet

Kijong Han, Sangha Nam, Jiseong Kim, Younggyun Hahm, and Key-Sun Choi
Semantic Web Research Center, School of Computing, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon, South Korea
{han0ah, nam.sangha, jiseong, hahmyg, kschoi}@kaist.ac.kr

Abstract

In this study, we investigated unsupervised learning based Korean word sense disambiguation (WSD) using CoreNet, a Korean lexical semantic network. To facilitate the application of WSD to practical natural language processing problems, a reasonable method is required to distinguish between sense candidates. We therefore performed coarse-grained Korean WSD studies while utilizing the hierarchical semantic categories of CoreNet to distinguish between sense candidates. In our unsupervised approach, we applied a knowledge-based model that incorporated a Markov random field and dependency parsing to the Korean language in addition to utilizing the semantic categories of CoreNet. Our experimental results demonstrate that the developed CoreNet based coarse-grained WSD technique exhibited an 80.9% accuracy on the datasets we constructed, and was proven to be effective for practical applications.

Keywords: Word Sense Disambiguation, CoreNet, Markov Random Field

1. Introduction

Words that have the same form can have different meanings. Word Sense Disambiguation(WSD) is to select the correct meaning of a word in context. It is an important problem that can be utilized in many problems of natural language processing such as machine translation and information extraction (Chaplot et al., 2015). In English, studies are typically conducted on the basis of the senses listed in the Princeton WordNet(PWN) as candidates for the meaning of words (Navigli et al., 2007; Chaplot et al., 2015). In this study, we resolve the ambiguity of words based on the senses listed in CoreNet (Choi et al, 2004), a Korean lexical semantic network.

Both PWN and CoreNet are fine-grained resources, so it is difficult for even human annotators to correctly identify the senses of words. In order for WSD to become an enabling technique for end-to-end applications, it requires the ability to make reasonable sense distinctions (Navigli et al., 2007). In English, coarse-grained WSD is performed by semi-automatically clustering PWN senses (Navigli et al., 2007). However, Korean dictionaries also list homograph and polyseme numbers for each sense. Homographs are words that have the same form but completely different meanings, and polysemes are words that have the same broad meaning but different etymological meanings. For this reason, in Korean language studies, coarse-grained WSD is typically conducted at the homograph level (Shin and Ock, 2016).

WSD methods can generally be divided into supervised and unsupervised learning methods. Supervised learning methods learn sense-tagged corpora and show relatively high levels of performance. However, it is expensive and time-consuming to construct such corpora. The results of one Korean language study showed a 96.5% accuracy based on learning a corpus consisting of 10 million sense tagged words (Shin and Ock, 2016).

In contrast, unsupervised learning methods are relatively low performance but do not require training data. For example, knowledge-based methods that employ lexical semantic networks have been successful (Agirre et al., 2014; Chaplot et al., 2015) because they are able to obtain wide coverage and good performance using structured knowledge (Iacobacci et al., 2016). Along these same lines, state-of-the-art research has been conducted based on Markov Random Fields (MRF) (Chaplot et al., 2015), which convert sentences to an MRF model through part-of-speech (POS) tagging and dependency parsing, and then determines the meanings of all target words in the sentence by way of a maximum a posteriori (MAP) query.

The remainder of this paper is organized as follows. Section 2 describes our investigation of unsupervised Korean WSD using the semantic category "concept" in CoreNet and homographs for coarse-grained WSD. In Section 3, we explain our MRF based method (Chaplot et al., 2015) and its application to CoreNet. Section 4 presents the datasets created during the evaluation of the proposed approach, and outlines the results of the experiments conducted to demonstrate the performance and efficacy of the proposed approach for distinguishing candidates of word senses using the semantic category "concept" in CoreNet. Finally, our conclusions are presented in Section 5, along with our plans for future work.

2. Background

2.1 CoreNet and Concept

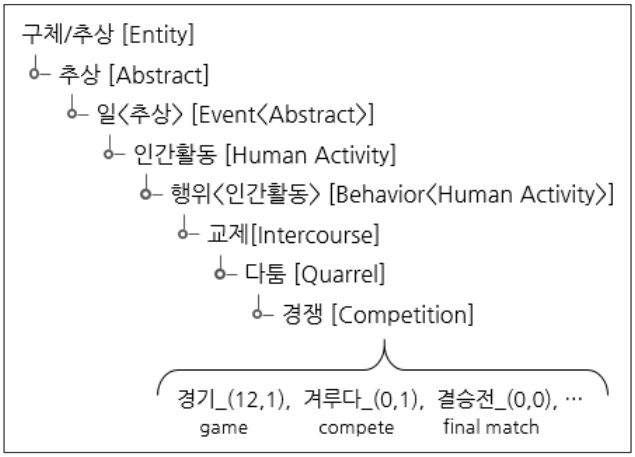


Figure 1: Example concept hierarchy in CoreNet.

CoreNet (Choi et al., 2004) is a lexical semantic network that represents the senses and relationships of Korean nouns, adjectives, and verbs. There are about 73,000 senses in CoreNet, and each sense contains additional resources, such as definitions and usage.

The key feature of CoreNet is its concept hierarchy. In CoreNet, the term "concept" refers to a semantic category, and every sense in CoreNet is mapped to one or more concepts. The concept hierarchy is based on Japanese NTT Goidaikei, which is a Japanese lexicon (Ikehara et al.,

1997), and includes a total 2,954 concepts, 277 of which are Korean in origin. These concepts constitute a hierarchy with a maximum depth of 12, each of which is mapped to Chinese, Japanese, and PWN senses.
An example of the concept hierarchy is illustrated in Figure 1 for the concept "Competition," which has a depth of eight. The verb sense "compete" and noun sense "game" and "final match" are mapped to this concept. In each sense, the first number after the word represents the homograph number and the second represents the polysemy number.

2.2 WSD with CoreNet

There is a need for coarse-grained WSD to facilitate practical applications of WSD (Navigli et al., 2007). To this end, we used homographs as in other Korean studies, as well as the concepts in CoreNet. In CoreNet, senses that have different homograph numbers are typically mapped to different concepts; however, there are a few exceptions.

- 사과[sa-gwa]-(1,0) : abbreviation of Korean cantaloupe
- 사과[sa-gwa]-(3,0) : apple

The two senses of the word "sa-gwa" have different homograph numbers, but both are mapped to the same concept "Edible Fruit." In this case, these senses are clustered into the same candidate. Senses mapped to the same concept are considered to be the same candidate; thus, the concept can be utilized in the WSD method. That is, the sense candidates of the word "sa-gwa" can be classified as shown in Table 1. Concepts such as "Apology" and "Edible Fruit" can also be utilized when evaluating the suitability of each candidate for the word "sa-gwa." In addition, as shown in Section 5, these types of candidate distinctions based on CoreNet concepts are meaningful in other natural language processing (NLP) tasks.

	Concept	(Homograph, Polyseme) Number	Definition
O	**Apology**	(8,0)	- apology.
1	**Edible Fruit**	(1,0)	- abbreviation of Korean cantaloupe.
		(3,0)	- apple.
2	**Study general/ Subject of Study**	(6,1)	- 4 courses of cultivating one's moral sense.
		(6,2)	- 4 courses of Confucianism.

Table 1: Examples of sense candidates for word '사과[sa-gwa]' based on the CoreNet Concept.

3. Approach

The primary approach outlined in this paper is the MRF based method, which will be described in Section 3.2. However, an alternate approach is the term frequency–inverse document frequency (TF-IDF) vector similarity method, which will be described in Section 3.1, and can be used to obtain the frequency values of concepts necessary to implement the MRF based method, as described in Section 3.2.

3.1 TF-IDF Vector Similarity

The TF-IDF Vector Similarity method uses the cosine similarity between the TF-IDF vector of all definition and usage sentences mapped to the concept associated with each candidate and the TF-IDF vector of the sentence containing the words to be disambiguated. The candidate that has the largest cosine similarity is selected. If the candidates are mapped to multiple concepts, then it is considered correct to disambiguate the sense when the system select one of those concepts. This method is based on the principle that the more the same words appear in the two sentences, the more the TF-IDF vector cosine similarity increases.

3.2 MRF Method

A MRF is an undirected graphical model that consists of set of random variables. Each node in the graph represent a random variable, and each random variable is only dependent on another random variable that represents another node that is directly connected by an edge. This model has been used to solve many NLP problems (Jung et al., 1996; Chaplot et al., 2015)
In this study, we applied an MRF based WSD method (Chaplot et al., 2015) to the Korean language using the concept hierarchy in CoreNet. In this method, target words for the WSD in a sentence are selected as nodes in the MRF, and edges are only generated for two directly connected words in the dependency tree. Finally, the senses of all the words are jointly disambiguated by way of a MAP query on this MRF model. We adopted the detailed methods described by Chaplot et al. (2015), and outline how they were applied to the Korean language in the following text.

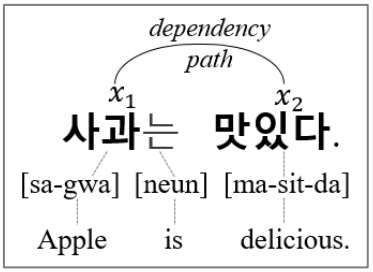


Figure 2 : An example of converting sentence to MRF

First, all common nouns, verbs, and adjectives in a sentence were designated WSD target words. Then, we set the random variables representing the concepts of these words to $X = \{x_1, x_2, \ldots, x_n\}$, where x_i can take m_i possible concept values. The concepts that x_i could take on were $s_i^1, s_i^2, \ldots, s_i^{m_i}$. In the case of the sentence shown in Figure 2, the noun "sa-gwa" and adjective "ma-sit-da" were selected as the target words, and the postposition "neun" was not selected. The random variable x_1 represented the word "sa-gwa," and the random variable x_2 was used to represent "ma-sit-da." The candidates of x_1 were s_1^1 = "Apology," s_1^2 = "Edible Fruit," and so on, as shown in Table 1.
There are node potential function $\psi(x_i)$, and edge potential function $\psi(x_i, x_j)$ for this MRF are as follows.

$$\psi(x_i = s_i^a) \propto \log(frequency(s_i^a) + e)$$

where $frequency(s_i^a)$ refers to the frequency of occurrence of concept s_i^a. These values were measured for 1.7 million words from 10% of the full text of Wikipedia using the TF-IDF method described in Section 3.1. In this process, we set the cosine similarity threshold value to 0.14,

which resulted in a precision of 0.951 and a recall value of 0.287 for our datasets..

$$\psi\left(x_i = s_i^a, x_j = s_j^b\right) \propto Relatedness(s_i^a, s_j^b)$$

The edge potential function, which indicates when two related words simultaneously have certain concepts, is proportional to the relatedness between the two concepts. Edges are only generated when two words are directly connected on the dependency tree. If we let this set of edges be E, then $\{x_i, x_j\} \in E$. The relatedness can be measured by the following two methods, and experiments were conducted for both.

(1) $Relatedness\left(s_i^a, s_j^b\right) = 1/(shortestpath\left(s_i^a, s_j^b\right) + 1)$
(2) $Relatedness\left(s_i^a, s_j^b\right) = \log(frequency\left(s_i^a, s_j^b\right) + e)$

where $shortestpath\left(s_i^a, s_j^b\right)$ refers to the inverse of the shortest path between concept s_i^a and s_j^b in CoreNet, and $frequency\left(s_i^a, s_j^b\right)$ refers to the frequency of co-occurrence of the two concepts s_i^a, s_j^b in same sentence. This measurement method is same as that employed by Chaplot et al. (2015), and can be used to obtain the node potential value..

$$\psi(\mathrm{X}) = \prod_{x_i \in X} \psi(x_i) \prod_{\{x_i, x_j\} \in E} \psi(x_i, x_j)$$

The final potential function of this model is as shown above. Let S be the set of disambiguated concepts for each word. Then, we can find S jointly through the MAP query shown below. Note that we used the library(Ankan and Panda, 2015) when implementing this MRF model.

$$\arg\max_S \psi(\mathrm{X} = \mathrm{S})$$

4. Experiments

4.1 Datasets

Each of our datasets consisted of a sentence and a word that was disambiguated in the sentence. To create our datasets, we randomly selected sentences from articles featured on Wikipedia, and then randomly selected either a noun, verb, or adjective. Three annotators were used to tag the proper senses to the words in these datasets. A total of 470 datasets were constructed, and the number of datasets with ambiguity, i.e., the number of datasets that had more than two candidates, was 215. The statistics for the datasets are shown in Table 2. All of the sentences were different, but the WSD target words may have been the same, so the number of words in Table 2 refers to the number of different words in our datasets.

	# data	# word
All data	470	322
Data with ambiguity	215	144

Table 2 : Statistics of our datasets

4.2 Performance

Method	RANDOM	TF-IDF	MRF SP	MRF co-occur
Accuracy for all data	73.6	88.5	**91.3**	91.1
Accuracy for data with ambiguity	42.2	74.9	**80.9**	80.5

Table 3 Accuracy by method

The results after measuring the accuracy of the data in the datasets is shown in Table 3. The accuracy was measured as follows. In the RANDOM baseline, a candidate was randomly selected, and the results of five trials were averaged. TF-IDF refers to the TF-IDF vector similarity based method described in Section 3.1. "MRF SP" and "MRF co-occur" refer to the MRF based method described in Section 3.2. The differences between the columns was used as the method of relatedness measurement. "MRP SP" used shortest path, as described in (1), and "MRF co-occur" used the method of co-occurrence, as described in (2). The MRF based method exhibited an 80.9% accuracy on our datasets with ambiguity, and the performance of this method was higher than that of the RANDOM and TF-IDF based methods. Our accuracy was lower than that of the recent Korean WSD study, which employed a supervised approach (Shin and Ock, 2016); however, our method is unsupervised, and therefore has the advantage that it can be applied to any document without learning.

4.3 Results of Applying Our WSD Method

The performance of the task of Relation Extraction from sentences improved when our WSD method was applied. This confirms that coarse-grained WSD based on the CoreNet concept is meaningful in real applications. The details of our application method are as follows. We applied our WSD method to the convolutional neural network based relation extraction model, which was implemented by our research team for the Korean language based on the description in (Zeng et al., 2014), and word embedding vectors of each token from the sentences were used as inputs to this relation extraction model. Word embedding has an advantage in that tokens with high semantic relevance are generated with similar real vector values. However, if the sentence is only tokenized in morpheme units, the algorithm cannot distinguish between ambiguous words. Thus, we applied a sense number to each word token using our WSD method so that words that had different senses but were in the same form could be distinguished at the time of embedding. This sense-tagged embedding exhibited a 7% higher F1-score performance for relation extraction, as shown in Table 4.

Unit of word embedding token	Morpheme	Morpheme + Sense
Relation Extraction F1-score	0.474	**0.544**

Table 4 : F1-score of Relation Extraction

5. Conclusion and Future Work

In this study, we implemented an unsupervised coarse-grained WSD algorithm using the semantic category "concept" in CoreNet for coarse-grained sense distinction. We then confirmed that it was meaningful by computing how our WSD algorithm improved the relation extraction F1-score. For our unsupervised approach, we also utilized the CoreNet concept as applied to knowledge-based MRF WSD model, and computed an accuracy of 80.9% using the datasets we constructed.

In the Korean language, the Sejong corpus consist of 10 million POS and sense tagged words. However, the sense numbers of the Sejong corpus and those of CoreNet originated from different Korean dictionaries. In the future, we plan to revise our method to utilize the Sejong corpus in CoreNet because we believe that this will improve the performance of Korean language WSD.

6. Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (2013-0-00109, WiseKB: Big data based self-evolving knowledge base and reasoning platform)

7. Bibliographical References

Agirre, E., de Lacalle, O. L., and Soroa, A. (2014). Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1), 57-84.

Ankan, A., and Panda, A. (2015). pgmpy: Probabilistic Graphical Models using Python. In *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*.

Chaplot, D. S., Bhattacharyya, P., and Paranjape, A. (2015). Unsupervised Word Sense Disambiguation Using Markov Random Field and Dependency Parser. In *AAAI* (pp. 2217-2223).

Choi, K. S., Bae, H. S, Kang, W., Lee, J., Kim, E., Kim, H., Kim, D., Song, Y., and Shin, H. (2004). Korean-Chinese-Japanese Multilingual Wordnet with Shared Semantic Hierarchy. In *LREC*.

Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2016). Embeddings for Word Sense Disambiguation: An Evaluation Study. In *ACL (1)*.

Ikehara, S. et al. (1997). The Semantic System, volume 1 of Goidaikei – A Japanese Lexicon. Tokyo: Iwanami Shoten.

Jung, S. Y., Park, Y. C., Choi, K. S., and Kim, Y. (1996). Markov random field based English part-of-speech tagging system. In *Proceedings of the 16th conference on Computational linguistics*-Volume 1 (pp. 236-242). Association for Computational Linguistics.

Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 30-35). Association for Computational Linguistics.

Shin, J. C., and Ock, C. Y. (2016). Improvement of Korean Homograph Disambiguation using Korean Lexical Semantic Network (UWordMap). *Journal of KIISE*, 43(1), 71-79.

Zeng, D., Liu, K., Lai, S., Zhou, G., and Zhao, J. (2014). Relation Classification via Convolutional Deep Neural Network. In *COLING* (pp. 2335-2344).

UFSAC: Unification of Sense Annotated Corpora and Tools

Loïc Vial, Benjamin Lecouteux, Didier Schwab
LIG – GETALP – Univ. Grenoble Alpes – France
{loic.vial, benjamin.lecouteux, didier.schwab}@imag.fr

Abstract

In Word Sense Disambiguation, sense annotated corpora are often essential for evaluating a system and also valuable in order to reach a good efficiency. Always created for a specific purpose, there are today a dozen of sense annotated English corpora, in various formats and using different versions of WordNet. The main hypothesis of this work is that it should be possible to build a disambiguation system by using any of these corpora during the training phase or during the testing phase regardless of their original purpose. In this article, we present UFSAC: a format of corpus that can be used for either training or testing a disambiguation system, and the process we followed for constructing this format. We give to the community the whole set of sense annotated English corpora that we know, in this unified format, when the copyright allows it, with sense keys converted to the last version of WordNet. We also provide the source code for building these corpora from their original data, and a complete Java API for manipulating corpora in this format. The whole resource is available at the following URL: `https://github.com/getalp/UFSAC`.

Keywords: Word Sense Disambiguation, sense annotated corpora, unified resource, tools

1. Introduction

Whether they are used for the evaluation or for the learning process of a Word Sense Disambiguation (WSD) system, the importance of sense annotated corpora in Natural Language Processing (NLP) is considerable. On one hand, the evaluation *in vivo*, i.e. the evaluation of a WSD system as part of a larger task, has never been really exploited. On the other hand, the evaluation *in vitro*, which uses directly sense annotated corpora by comparing the output of a system to manual annotations, is predominant. Moreover, WSD systems exploiting examples from sense annotated corpora are generally far better than those which do not (Navigli et al., 2007; Moro and Navigli, 2015).

At the time of its creation, WordNet (Miller, 1995) was undoubtedly the only lexical database freely available for English. Since the beginning of the 2000s, it has become the *de facto* standard for WSD in this language. Indeed, most of sense annotated corpora are either directly annotated with WordNet sense keys or they are annotated with a sense inventory linked to the senses of WordNet, such as BabelNet (Navigli and Ponzetto, 2010).

However, it is not trivial to use these corpora, because most of them differ in their format and on the version of WordNet they use. As a consequence, very few works in the literature of WSD are trained or evaluated on more than two annotated corpora.

Also, WSD systems are systematically evaluated on corpora that have been initially created for the purpose of evaluation, and never on corpora that have been created for another purpose, such as training or for sense distribution estimation, whereas there is no scientific reason for that.

This paper presents a work of unification of all existing English corpora annotated with any version of WordNet to our knowledge, in a unique format, easy to understand, and easy to work with in practice. We put on the same level the corpora originally created for the evaluation and those for the learning, so to facilitate the creation of robust WSD systems which could for example be evaluated in a way where all corpora except one are used for the learning, and the remaining one is used for the evaluation, then switch the corpora and do this for every existing corpus.

The language resource that we provide contains all English sense annotated corpora in UFSAC (Unified Format for Sense Annotated Corpora), the format that we propose, with sense annotations converted to the last version of WordNet (3.0), along with Java code to easily read, write and modify any corpus in this format, and scripts for converting a corpus from its original format to UFSAC.

Our work is the continuity of the demonstration of (Vial et al., 2017), and it differs from the recent work of (Raganato et al., 2017) in several points. Their work is focused on the evaluation of WSD systems, whereas we provide a complete API for manipulating corpora in a new unified format (UFSAC), and conversion scripts allowing the full reconstruction of the corpora from the original data. We also propose five additional corpora in our resource among the most difficult to parse.

In our resource, we provide a script for converting a corpus from our format to theirs, so existing WSD systems that rely on their format can be trained or evaluated on any of the corpus that we produced. We also provide a script for converting their format to ours in order to facilitate any collaborative work in the community.

2. Sense Annotated Corpora: rare and costly resources

Generally speaking, a corpus is a collection of documents which can be used as samples of text for a particular language (Habert et al., 1998). A corpus may contain several millions of words, which can be lemmatized and annotated with information concerning their part of speech for example. Among these corpora, we can find the *British National Corpus* (Burnard, 1998) (100 million words) and the *American National Corpus* (Ide and Macleod, 2001) (20 million words). The texts come from several sources such as newspapers, books, encyclopedias or from the Web.

A sense annotated corpus is a corpus in which some or all words are annotated with an identifier of sense from

a specific lexical database. For example, all words in the corpus of the 7th task of the SemEval 2007 semantic evaluation campaign (Navigli et al., 2007) are annotated with sense identifiers from WordNet 2.1, whereas in the English corpus of the 13th task of SemEval 2015 (Moro and Navigli, 2015), all words are annotated with sense identifiers from WordNet 3.0, BabelNet 2.5 and Wikipedia pages.

There are at least three reasons to create a sense annotated corpus:

- Estimate the distribution of senses in the language. It is for this purpose that the SemCor (Miller et al., 1993) was annotated. Consequently, the senses in WordNet are, since version 1.7, sorted by this distribution of senses estimated on the SemCor.
- Build a Word Sense Disambiguation system which learns from examples contained in the annotated corpus. For instance, the OMSTI (Taghipour and Ng, 2015) was created for this purpose.
- Evaluate a WSD system by comparing its output to the annotations in the corpus, as it is the case for instance with corpora created as part of the evaluation campaigns SensEval-SemEval.

After their distribution, there is no scientific reason not to use indistinctly these corpora either for building a WSD system, for estimating the distribution of senses or for evaluating a WSD system. Indeed, the SemCor is used since a long time for the learning of WSD systems (Chan et al., 2007; Navigli et al., 2007) or more recently for the evaluation of different methods (Yuan et al., 2016). This last usage is still very rare, since it is one of the first experiment that we found in the literature, along with (Màrquez et al., 2002).

However, the format of the resources differs greatly depending on their original purpose. For the SemCor, a single file groups all the information, whereas in the case of the evaluation corpora, there are two files: one that contains the unannotated corpus, and the other that contains the sense annotations. In some corpora, like in the DSO and the OMSTI, there is one file for every lemma in the dictionary, and each file contains thousands of example sentences, where this lemma is the only word that is sense annotated.

Few data are manually sense annotated. The *Global WordNet Association* made a list of 26 corpora annotated with WordNet [1]. These corpora concern 17 languages, but only three of them reach 100,000 annotations. English, with more than 2 million words sense annotated ranks first, before Dutch with nearly 300,000 annotations and Bulgarian with 100,000 annotations. Thus, it is unsurprising that most of researches in WSD focus on English.

3. A single format for sense annotated corpora

The main purpose of this work is to help the construction and the evaluation of WSD systems, by giving to the community the set of all existing English sense annotated corpora to our knowledge, in the same format, using the same sense inventory, and tools to easily parse them, manipulate them, and convert corpora from their original format to our one.

Indeed, a large quantity of sense annotated data is vital for the construction of WSD systems. In evaluation campaigns, this often makes the difference. For example, looking at the data from the SemEval 2007 campaign (Navigli et al., 2007), which most of the recent systems were evaluated on, we observe that systems that did not use sense annotated data obtain a precision score up to $78 - 79\%$[2] (Schwab et al., 2013) (Chen et al., 2014) whereas those which use a lot of annotated data reach a score up to 82% (Chan et al., 2007) (Navigli, 2012) (Vial et al., 2016) and even 84% (Yuan et al., 2016).

Therefore, having all existing corpora in a unique format and using the same sense inventory offers several advantages: it allows to easily expand the quantity of data available for improving WSD systems, it allows to better estimate the distribution of senses in English, and finally, this format can help creating more robust WSD systems. Indeed, we still find a lot of works that focus on a single evaluation task (Vial et al., 2016; Chen et al., 2014), and in these cases, the analysis of the results concerning the robustness of the methods is limited. The unification of the format of sense annotated corpora could improve the evaluation process by facilitating a cross validation process for instance, where the system is evaluated sequentially on every corpus, with all others used for the training.

4. Provided resource

Our work consists in gathering all English corpora sense annotated with WordNet, and convert all of them to a unified format that is able to contain all the informations present in the original format. We created format conversion scripts for this purpose, as well as scripts for cleaning the corpora, and converting the sense annotation to the last version of WordNet (3.0). The resulting corpora are parts of the resource when the copyright allows it, along with the format conversion scripts, the cleaning scripts, and the sense conversion scripts. For the corpora that we cannot distribute because of the licence, anyone that possess them can still run our scripts to turn the original resource into our format. Finally, an API is provided for parsing, creating and manipulating corpora in our format. The resource is accessible at the following URL: `https://github.com/getalp/UFSAC`.

4.1. Sense annotated corpora

Our resource contains the following corpora:

- The *SemCor* (Miller et al., 1993), a subset of the Brown Corpus (Francis and Kučera, 1964). Original annotations are done with WordNet 1.6.
- The *DSO (Defence Science Organisation)* (Ng and Lee, 1997), a non-free corpus, that is focused on 121

[1] `http://globalwordnet.org/wordnet-annotated-corpora/`

[2] This means that the system has chosen the same sense than the human annotators in 78 to 79% of cases

nouns and 70 verbs among the most frequently used and the most ambiguous words in English and have been annotated in various contexts with WordNet 1.5.

- The *WordNet Gloss Tag* [3], a corpus which consists of all definitions of WordNet (Miller, 1995) with every words sense annotated since version 3.0.

- The *OMSTI (One Million Sense-Tagged Instances)* (Taghipour and Ng, 2015), a corpus of approximately one million words sense annotated with WordNet 3.0.

- The *MASC (Manually Annotated Sub-Corpus)* (Ide et al., 2008), we used the version given in the article of (Yuan et al., 2016), annotated with the NOAD (New Oxford American Dictionary), but with corresponding WordNet 3.0 sense keys.

- The *Ontonotes 5.0* (Hovy et al., 2006), annotated with WordNet 3.0.

- The corpora of the WSD evaluation campaigns SemEval-SensEval: SensEval 2 (using WordNet 1.7), SensEval 3 (WN 1.7.1), SemEval 2007 (WN 2.1), SemEval 2013 (WN 3.0) and SemEval 2015 (WN 3.0).

Table 1 summarizes statistics concerning these corpora.
After the conversion of all these corpora into our format, we executed four post-processing steps: sense annotation conversion, identical sentences merging, lemma and POS tagging, and finally a cleaning step.

4.1.1. Sense Annotation Conversion

Sense annotations have been converted, when necessary, from their original WordNet sense key to the last version of WordNet (3.0) thanks to conversion tables from (Daudé et al., 2000).
However, because some senses have been dropped from the old versions of WordNet, some sense annotations have not been converted. In any case, the original sense annotations are always kept alongside the converted sense annotation.
When a sense is mapped to two or more senses with equal probability, all resulting senses are added to the word annotations, separated by a semicolon.

4.1.2. Identical sentences merging

This step is only applied on the DSO and the OMSTI: because these corpora are constructed such that they contain lists of sentences with only one word that is sense annotated, surrounded by words not annotated, some sentences are present in different places across the corpus, but with different words that are sense annotated.
The merging phase identifies identical sentences with annotations on different words, and creates a single sentence containing all annotations. Thus, this steps adds a crucial information for some WSD systems. For instance, a similarity-based WSD system can now "learn" that two word senses are often located in the same sentence.

[3] `http://wordnet.princeton.edu/glosstag.shtml`

4.1.3. Lemma and POS tagging

For the corpora that do not already contain these informations, we added the lemma for every word, when existing, using the WordNet's *morphy* tool, and the part-of-speech tag from the Penn Treebank tag set using Stanford's Log-linear POS tagger (Toutanova et al., 2003).

4.1.4. Cleaning

Finally, this last step consists of trimming words, removing invisible characters and removing inconsistent annotations, for instance when the part of speech annotation differs from the part of speech of the sense annotation.

4.2. UFSAC File format

Our approach for the unification of the different annotated corpora begins with a file format that is descriptive, easily understandable and readable by a human, and at the same time, efficient for a program to parse and create. Finally, it should be able to contain all the information contained in the original resources. These informations are represented with the following concepts:
– A Lexical Entity (LE) is an entity that contains a set of annotations.
– A Corpus is a LE which contains a set of documents.
– A Document is a LE hich contains a set of paragraphs.
– A Paragraph is a LE which contains a set of sentences.
– A Sentence is a LE which contains a set of words.
– A Word is a LE which has a special mandatory annotation "surface form", which is the value of the word.

In order to represent these concepts, UFSAC is based on a simple XML syntax with some conventions: lexical entities are represented by XML nodes (`corpus`, `document`, `paragraph`, `sentence` and `word`), and annotations are node attributes.

The annotations also follow a certain convention, we used the following to annotate words:
– The identifier (`id`) of a lexical entity, particularly useful for corpora originally created for the evaluation (e.g. "d001.s002.t003").
– The surface form (`surface_form`) of a word.
– The lemma (`lemma`) of a word.
– The part of speech (`pos`) of a word.
– The sense of a word, in a specific lexical database, for example WordNet 3.0 (`wn30_key`), WordNet 1.7.1 (`wn171_key`)... If multiple senses are specified (it is the case in the coarse-grained task of SemEval 2007 for instance), they are separated with a semicolon (`;`).
The information of the sense is the one which is the most useful in our case, and it is specific to each lexical database, instead of having a unique "sense" annotation as we can find in most other formats. That way we allow multiples sense annotations from different lexical databases at the same time. For example, the DSO is originally annotated with senses from WordNet 1.5, and the conversion to WordNet 3.0 is sometimes impossible for some senses which were deleted between the two versions. This convention allows us to keep the original annotations, yet to have the annotations from the last version of WordNet, or any other lexical database (for instance BabelNet) at the same time.

Corpus	Sentences	Words		Annotated parts of speech			
		Total	Annotated	Nouns	Verbs	Adj.	Adv.
SemCor	37176	778587	229517	87581	89037	33751	19148
DSO	178119	5317184	176915	105925	70990	0	0
WordNet GlossTag	117659	1634691	496776	232319	62211	84233	19445
MASC	34217	596333	114950	49263	40325	25016	0
OMSTI	820557	35843024	920794	476944	253644	190206	0
Ontonotes	21938	435340	52263	9220	43042	0	0
SemEval 2007 task 07	245	5637	2261	1108	591	356	206
SemEval 2007 task 17	120	3395	455	159	296	0	0
SemEval 2013 task 12	306	8142	1644	1644	0	0	0
SemEval 2015 task 13	138	2638	1053	554	251	166	82
Senseval 2	238	5589	2301	1061	541	422	277
Senseval 3 task 1	300	5511	1957	886	723	336	12

Table 1: Statistics related to our set of annotated corpora, after the conversion and cleaning phase.

The following is an example of the resulting UFSAC XML:

```
<corpus id="short_example">
  <document id="d001" >
    <paragraph>
      <sentence>
        <word surface_form="A" pos="DT" />
        <word surface_form="precise"
           wn30_key="precise%3:00:00::" />
        <word surface_form="example"
              pos="NN" lemma="example" />
        <word surface_form="." />
      </sentence>
    </paragraph>
  </document>
</corpus>
```

Our format thus allows to integrate the whole corpus in a single file, and it is easily readable, especially comparing to most original formats (c.f. the end of section 2.).

4.3. API and tools

An easy-to-use Java API is also provided to read, write and modify efficiently corpora in our format. It allows two styles of programming: you can either load a full corpus in memory, perform all your calculations and save it entirely in a file; or you can sequentially scan, edit or print a corpus from a file, in a streaming manner. The latter is particularly useful when working with huge files which do not fit into memory. Finally, we offer a set of scripts that perform the conversion of a corpus from its original format to our one, and some pre-processing and analyses scripts.

4.3.1. Core API

The core API is a package containing the base classes for manipulating corpora. For simplicity, the class names match exactly what is described in section 4.2..

The class **`Annotation`** describes an annotation on a lexical entity. Concretely, it is a pair of Strings (name/value) and a pointer to the annotated lexical entity.

The class **`LexicalEntity`** describes something that has zero or more annotations, with public methods for accessing/modifying them.

The class **`Word`** inherits from LexicalEntity, has a special mandatory annotation `surface_form`, which is the value of the word, and a parent sentence.

The class **`Sentence`** inherits from LexicalEntity, contains a list of words and a parent paragraph.

The class **`Paragraph`** inherits from LexicalEntity, contains a list of sentences and a parent document.

The class **`Document`** inherits from LexicalEntity, contains a list of paragraphs and a parent corpus.

Finally, the class **`Corpus`** inherits from LexicalEntity and contains a list of documents.

These few classes, coupled with two functions `Corpus.saveToXML` and `Corpus.loadFromXML` allow to create, save, load and modify any corpus easily.

4.3.2. Streaming API

For some corpora particularly huge, like the OMSTI, we also provide a sub-package `streaming`, which allows to read, write or modify a corpus sequentially, without being fully loaded into memory. This is similar to the Java SAX library (Simple API for XML), events are fired when reading a word, sentence, paragraph, etc., and the user can choose to respond to this event or not.

In practice, we provide a set of classes which cover most use cases.

The class **`StreamingCorpusReader`** allows to respond to the events `readBeginCorpus`, `readBeginDocument`, `readWord`, etc.. This can be useful for printing every word that is sense annotated for example.

The class **`StreamingCorpusModifier`** allows to modify a corpus in-place. This is specially useful for preprocessing, for instance convert every word to lowercase.

The class **`StreamingCorpusWriter`** is used for creating a new corpus, with its methods `writeBeginSentence`, `writeWord` and so on.

4.3.3. Scripts

Finally, we provide a set of examples and useful scripts which use our format and our API. The scripts are Java classes with a `main` method and are not part of any package.

The script **`ConvertOriginalCorpora`** allows to convert all corpora listed in subsection 4.1. from their original format to the UFSAC format. This is specially valuable for non-free corpus like the DSO, that we cannot share directly in our format, but that one can still buy in their original format, and then convert to our format. This script includes all post-processing steps described in subsection 4.1..

The scripts **`ConvertFromRaganato`** and **`ConvertToRaganato`** allow to convert a corpus from the format described by (Raganato et al., 2017) to UFSAC, and vice-versa.

The script **`ComputeMostFrequentSenses`** will calculate, for every lemma in WordNet, the most frequent sense (MFS), based on all usages in the given UFSAC corpora. This is helpful since in most evaluation campaigns, the MFS baseline (i.e. the score obtained when the MFS is assigned to every word) is important, and it is generally implicitly the sense distribution computed on the SemCor only.

The scripts **`AddCorpusLemma`** and **`AddCorpusPOS`** use respectively WordNet's *morphy* and Stanford's POS tagger to annotate a corpus with the lemma and POS of every word.

The script **`EvaluateWSD`** compare the sense annotations produced by a WSD system to the gold standard annotation, and compute the usual Precision, Recall, Coverage and F1 metrics for every given corpus.

The script **`GenerateCorpusStatistics`** is the one that was used to produce the table 1.

5. Experiments

In this section, we show an example of using all UFSAC corpora for the extension of a knowledge-based WSD system based on the Lesk measure. This experiment shows how this ressource can be used to easily improve an existing WSD system.

5.1. The Lesk and Extended Lesk Similarity Measures

(Lesk, 1986) proposed a simple algorithm for lexical disambiguation that evaluates the similarity between two senses (s_1, s_2) as the number of words in common in the definitions of the senses from a dictionary ($D(s_1)$,$D(s_2)$).
The Lesk measure compute an exact lexical match of the surface forms of the words in the definitions. If important words are missing or different synonyms of the same words are used in the definition of related senses, the overlap measure will not capture the proximity of their meanings appropriately. As definitions (especially in Princeton Wordnet) are very concise, it is difficult to obtain fine grained distinctions between senses.
In consequence, several variants of the Lesk measure tried to alleviate this problem, for instance (Baldwin et al., 2010) and (Miller et al., 2012), but the most common expansion technique is the so-called "extended/adapted Lesk" (Banerjee and Pedersen, 2002). The sense overlap is here expanded with the overlap of all definitions from all pairs of related senses in a lexico-semantic resource with a rich structure, such as WordNet.
In our experiment, we will create another expansion of the Lesk measure, based on UFSAC sense annotated corpora.

5.2. Expansion of Definitions Through UFSAC Sense-Annotated Corpora

Our method consists in expanding definitions with all neighbours of a target sense, taken from sense-annotated corpora. We consider that a neighbour is a word found in the same sentence as the target sense. More precisely, we proceed as following:

1. We parse every UFSAC corpus, sentence by sentence.
2. For every word which is sense-annotated in a sentence, we add to the definition of this sense in the dictionary every other word present in the sentence.

That is, for every sentence $S = w_0, w_1, \ldots, w_n$, and for every word w_k inside S, we add to the definition of the tagged sense of w_k, i.e. $D(s(w_k))$, every other words of the sentence, i.e. $w_i \forall i \in [0, n] i \neq k$.
As a consequence, every sense's definition in the dictionary will be extended with words that are related to this sense, in the same manner than (Banerjee and Pedersen, 2002)'s extended Lesk, but with words taken from sense annotated corpora.

5.3. Similarity-Based Word Sense Disambiguation

Now for evaluating this new expansion to the Lesk measure, we must use a similarity-based WSD algorithm that belongs to the broader category of knowledge-based approaches (using dictionaries, lexical base, encyclopedias...). In such systems, the disambiguation process consists of two layers: a local algorithm and a global algorithm. The local algorithm computes the proximity of two word senses, namely a semantic relatedness measure. The local similarity measurement is then used to find an optimal global sense assignment for all the content words of the text by the global algorithm. The local algorithm is here the Lesk measure augmented with the sense annotated corpora. We also filtered out stopwords according to the "long" list given in `https://www.ranks.nl/stopwords`.
As for global algorithms, they are often probabilistic combinatorial optimization algorithms, as WSD is fundamentally a discrete combinatorial optimization problem. Many such algorithms have been adapted to WSD, including genetic algorithms (Gelbukh et al., 2003), simulated annealing (Cowie et al., 1992), ant colony algorithms (Schwab et al., 2012) or more recently bee hive algorithms (Abualhaija and Zimmermann, 2016).
The different global algorithms mainly differ in the convergence speed to a close-to-optimal solution, however the bottleneck to the accuracy of the algorithm is the local algorithm (similarity measure) used, as it encodes the knowl-

System	SemEval 2007 Task 07	SemEval 2015 Task 13
Lesk + UFSAC corpora	79.83%	66.43%
Lesk	68.70%	50.65%
Extended Lesk (Banerjee and Pedersen, 2003)	78.01%	61.42%
Most Frequent Sense Baseline	78.90%	67.10%

Table 2: F1 scores of our similarity-based system augmented with words taken from all UFSAC corpora (except the evaluation corpora) on SemEval 2007 coarse-grained all-words task and SemEval 2015 fine-grained all-words task, compared to the Lesk, Extended Lesk and MFS baselines.

edge from the resource that allows to discriminate between the senses.

In this experiments, we use an adaptation to WSD of the Cuckoo Search Algorithm, the state of the art in combinatorial search algorithms (Yang and Deb, 2009). The algorithm relies on the Lévy flight distribution for an effective (and more meaningful) sampling of the search space.

The Cuckoo Search Algorithm is probabilistic and its result differs slightly from an execution to another (by an order of magnitude of less than 1%). So for each experiment, 30 executions are performed. Then, using a Shapiro-Wilk test (Shapiro and Wilk, 1965), we determined that none of the result distribution follow a normal distribution. Thus, we used a non-parametric Wilcoxon/Mann-Whitney-U (Wilcoxon, 1945) (Mann and Whitney, 1947) test in order to check the pairwise significance ($p < 0.01$) of all pairs of result distributions.

5.4. Results

We evaluate the performance of our expansion of definitions using all UFSAC corpora listed in subsection 4.1. except the ones we evaluated our system on: SemEval 2007 task 7 and SemEval 2015 task 13. We compare our similarity measure to the original Lesk and the Extended Lesk (Banerjee and Pedersen, 2003) measures. The results are presented in Table 2.

As we can see, our expansion of the definitions with words taken from the UFSAC corpora improves considerably the original Lesk measure, even more than the Extended Lesk measure. Therefore, this experiment demonstrates how much the addition of the UFSAC resource can improve a similarity-based WSD sytem. Of course, every other kind of WSD system can be improved, in particular supervised systems which rely solely on sense-annotated corpora and machine learning techniques (SVM, neural networks, etc.).

6. Conclusion

In this paper we advocate for a more uniform way of distributing sense annotated corpora, through a unique and uncomplicated file format. This unification can facilitate both the creation and the evaluation of Word Sense Disambiguation systems. Indeed, sense annotated corpora are historically separated between those created for the purpose of training, and those created for the purpose of evaluation. In addition, the formats of these corpora are often very different from each other: different file hierarchy, different syntax, and different sense inventory are used. Consequently, most WSD systems are trained and evaluated on few corpora comparing to the amount of existing corpora. Moreover, they are systematically evaluated only on corpora originally created for the purpose of evaluation, and trained only on corpora originally created for the purpose of training, whereas they could benefit from considering all of them in both tasks.

The unification of all sense annotated corpora hence allows to quickly expand a system which is trained on some resources to new data without the effort of writing another parser. Also, a system can now easily include to its training phase some corpora that were originally created for evaluation, and/or evaluate its performance on parts of corpora originally created for training. This easily allows a much better coverage and a more fine-grained analysis of a WSD system performance.

In our language resource, we gathered all existing English sense annotated corpora that we know, and we converted them in a simple and consistent XML file format that we named UFSAC. We also converted their sense annotations to the last version of WordNet (3.0). The corpora are only available when the licence authorizes it, but we also provide scripts that can easily convert a corpus from its original format to the one we propose. Thus, anyone who possess the corpora that we cannot distribute can still benefit from this work. In addition, we provide a complete Java API for reading, writing and modifying corpora in our unified format, along with example codes and tools for many applications such as lemmatization, POS-tagging, sense distribution estimation, etc. Finally, a demonstration of a simple use of all UFSAC corpora for extending a similarity-based WSD system is shown in section 5..

In the future, we plan to add to our resource other corpora such as the corpora created for the lexical sample tasks of SensEval/SemEval, and sense annotated corpora in other languages. We also plan to improve the UFSAC format by adding a better support for multiword expressions. The resource will be continuously updated at this url: `https://github.com/getalp/UFSAC`.

7. Bibliographical References

Abualhaija, S. and Zimmermann, K.-H. (2016). D-bees: A novel method inspired by bee colony optimization for solving word sense disambiguation. *Swarm and Evolutionary Computation*, pages –.

Baldwin, T., Kim, S., Bond, F., Fujita, S., Martinez, D., and Tanaka, T. (2010). A reexamination of mrd-based word sense disambiguation. 9(1):4:1–4:21, March.

Banerjee, S. and Pedersen, T. (2002). An adapted lesk algorithm for word sense disambiguation using wordnet. In *CICLing 2002*, Mexico City, February.

Banerjee, S. and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 805–810.

Burnard, L. (1998). *The British National Corpus*.

Chan, Y. S., Ng, H. T., and Zhong, Z. (2007). Nus-pt: Exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 253–256, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chen, X., Liu, Z., and Sun, M. (2014). A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1025–1035, Doha, Qatar, October. Association for Computational Linguistics.

Cowie, J., Guthrie, J., and Guthrie, L. (1992). Lexical disambiguation using simulated annealing. In *COLING 1992*, volume 1, pages 359–365, Nantes, France, août.

Daudé, J., Padró, L., and Rigau, G. (2000). Mapping wordnets using structural information. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 504–511, Stroudsburg, PA, USA. Association for Computational Linguistics.

Francis, W. N. and Kučera, H. (1964). A standard corpus of present-day edited american english, for use with digital computers (brown). Technical report, Brown University, Providence, Rhode Island.

Gelbukh, A., Sidorov, G., and Han, S. Y. (2003). Evolutionary approach to natural language wsd through global coherence optimization. *WSEAS Transactions on Communications*, 2(1):11–19.

Habert, B., Fabre, C., and Issac, F. (1998). *DE L'ECRIT AU NUMERIQUE. Constituer, normaliser et exploiter les corpus électroniques*. Number ISBN : 2-225-82953-5. ELSEVIER MASSON.

Ide, N. and Macleod, C. (2001). The american national corpus: A standardized resource of american english. In *Proceedings of Corpus Linguistics 2001*, volume 3.

Lesk, M. (1986). Automatic sense disambiguation using mrd: how to tell a pine cone from an ice cream cone. In *Proceedings of SIGDOC '86*, pages 24–26, New York, NY, USA. ACM.

Mann, H. B. and Whitney, D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50–60.

Miller, G. A., Leacock, C., Tengi, R., and Bunker, R. T. (1993). A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, HLT '93, pages 303–308, Stroudsburg, PA, USA. Association for Computational Linguistics.

Miller, T., Biemann, C., Zesch, T., and Gurevych, I. (2012). Using distributional similarity for lexical expansion in knowledge-based word sense disambiguation. In *Proceedings of COLING 2012*, pages 1781–1796, Mumbai, India, December. The COLING 2012 Organizing Committee.

Miller, G. A. (1995). Wordnet: A lexical database. *ACM*, Vol. 38(No. 11):p. 1–41.

Moro, A. and Navigli, R. (2015). Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297, Denver, Colorado, June. Association for Computational Linguistics.

Màrquez, L., Raya, J., Carroll, J., McCarthy, D., Agirre, E., Martínez, D., Strapparava, C., and Gliozzo, A. (2002). Experiment a : Several all-words wsd systems for english. Technical report, Meaning, Developing multilingual Web-scale Language Technologies.

Navigli, R. and Ponzetto, S. P. (2010). Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 216–225. Association for Computational Linguistics.

Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). Semeval-2007 task 07: Coarse-grained english all-words task. In *SemEval-2007*, pages 30–35, Prague, Czech Republic, June.

Navigli, R. (2012). A quick tour of word sense disambiguation, induction and related approaches. In *Proceedings of the 38th Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, pages 115–129.

Raganato, A., Camacho-Collados, J., and Navigli, R. (2017). Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain, April. Association for Computational Linguistics.

Schwab, D., Goulian, J., Tchechmedjiev, A., and Blanchon, H. (2012). Ant Colony Algorithm for the Unsupervised Word Sense Disambiguation of Texts: Comparison and Evaluation. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2012)*, Mumbai (India), dec.

Schwab, D., Goulian, J., and Tchechmedjiev, A. (2013). Désambiguïsation lexicale de textes : efficacité qualitative et temporelle d'un algorithme à colonies de fourmis. *TAL*, 54(1):99–138.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 3(52).

Taghipour, K. and Ng, H. T. (2015). One million sense-tagged instances for word sense disambiguation and induction. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 338–344, Beijing, China, July. Association for Computational Linguistics.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Con-*

ference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vial, L., Tchechmedjiev, A., and Schwab, D. (2016). Extension lexicale de définitions grâce à des corpus annotés en sens. In *Traitement Automatique des Langues Naturelles (TALN)*.

Vial, L., Lecouteux, B., and Schwab, D. (2017). Uniformisation de corpus anglais annotés en sens. In *24ème Conférence sur le Traitement Automatique des Langues Naturelles*, Orléans, France, June.

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, December.

Yang, X.-S. and Deb, S. (2009). Cuckoo search via lévy flights. *Proc. of World Congress on Nature and Biologically Inspired Computing*, pages 210–214.

Yuan, D., Richardson, J., Doherty, R., Evans, C., and Altendorf, E. (2016). Semi-supervised word sense disambiguation with neural models. In *COLING 2016*.

8. Language Resource References

Hovy et al. (2006). *OntoNotes: The 90% Solution*.

Ide et al. (2008). *MASC: the Manually Annotated Sub-Corpus of American English*.

Miller et al. (1993). *A Semantic Concordance*.

Miller. (1995). *Wordnet: A Lexical Database*.

Ng and Lee. (1997). *DSO Corpus of Sense-Tagged English*.

Taghipour and Ng. (2015). *One Million Sense-Tagged Instances for Word Sense Disambiguation and Induction*.

Retrofitting Word Representations for Unsupervised Sense Aware Word Similarities

Steffen Remus and **Chris Biemann**
Universität Hamburg
Hamburg, Germany
{remus,biemann}@informatik.uni-hamburg.de

Abstract

Standard word embeddings lack the possibility to distinguish senses of a word by projecting them to exactly one vector. This has a negative effect particularly when computing similarity scores between words using standard vector-based similarity measures such as cosine similarity. We argue that minor senses play an important role in word similarity computations, hence we use an unsupervised sense inventory resource to retrofit monolingual word embeddings, producing sense-aware embeddings. Using retrofitted sense-aware embeddings, we show improved word similarity and relatedness results on multiple word embeddings and multiple established word similarity tasks, sometimes up to an impressive margin of $+0.15$ Spearman correlation score.
Keywords: word senses, word similarity and relatedness, word sense induction

1. Introduction

Word embeddings – generated with neural networks (NN) or other factorization techniques – are a standard element in natural language processing (NLP) applications. However, an important issue is their lack of sense-awareness, i.e. a word and its vector share a bijective mapping and a potential multiplicity of word meanings is ignored. The word *iron*, for example, which may refer to an atomic element, a device for smoothing clothes, a golf club, a color, or other meanings, is represented by a single common vector in the vector space. WORDNET (Fellbaum, 1998), on the other hand, defines four different interpretations of the word *iron*, and even this can never be considered to be complete as language evolves. Assigning the same vector for each distinguished sense and using them in downstream tasks such as sentiment analysis, named entity recognition, question answering or many others, is error prone by design due to obvious misinterpretations and error propagation.

Sense inventories – of which WORDNET is probably the most well known – are required to distinguish between different word senses, and meanings, rather than words, should be represented in the vector space (Navigli, 2009; Denkowski and Lavie, 2014). We use a simple, yet effective technique to retrofit standard word embeddings to produce embedding vectors of senses using external resources as sense inventories. Our hypothesis is that retrofitting pretrained word embeddings to gain sense-aware embeddings is beneficial for word similarity computations. Using vectors of senses rather that vectors of words, we are indeed able to report substantial relative improvements for multiple word similarity tasks and for various types of retrofitted embeddings from five monolingual corpora.

Because a word maps to multiple sense vectors in this scenario, standard cosine similarity computations alone are not applicable anymore, we thus test a number of sense-aware comparison methodologies based on cosine similarity. In particular for word pairs involving minor/rare senses, we expect improvements in the sense-aware setting as the influence of the dominating major sense is diminished. Additionally, we compare our approach with two baseline approaches to supervised and unsupervised sense-aware embeddings: AUTOEXTEND (Rothe and Schütze, 2015) and ADAGRAM (Bartunov et al., 2016).

To the best of our knowledge, we are the first to employ unsupervised word sense induction techniques for retrofitting single word vectors to the multiplicity of their meanings, creating new pseudo word-sense vectors, and using those for semantic similarity. Additionally, we test standard word sense induction (WSI) techniques using word embeddings themselves in order to make the retrofitting process self-sustained. Evidence presented below indicates that word embeddings are hardly useful in word sense induction clustering, due to the fact that their neighborhoods largely consist of words referring to the dominant sense in the source corpus.

2. Related Work

A number of word similarity benchmarks exist in order to intrinsically test the semantic properties of word embeddings (Hill et al., 2014; Finkelstein et al., 2001; Bruni et al., 2014; Gerz et al., 2016). Similarities are usually computed by means of cosine similarity between two vectors, which are representations of words in an embedded vector space.

The history of word embeddings is vast, ranging from geometrical matrix factorization methods like latent semantic analysis (Landauer and Dumais, 1997, LSA) or principal component analysis (PCA), over to probabilistic topic models such as probabilistic latent semantic analysis (Hofmann, 1999, PLSA) or latent dirichlet allocation (Blei et al., 2003, LDA), to more recent approaches based on neural network (NN) architectures such as skip-gram negative-sampling (SGNS), continuous bag of words (CBOW), or global vectors (Pennington et al., 2014, GLOVE), from which the former two are both available in the WORD2VEC toolkit (Mikolov et al., 2013). In this paper, we mainly focus on embeddings generated by NNs because of their superior performance and current impact on NLP research. However, we note that our findings are also applicable to other types of embedded word vector spaces, as we shall see below.

Rothe and Schütze (2015) introduced AUTOEXTEND, a supervised neural network model which enriches existing embeddings with word sense information from WORDNET or other sense inventories.[1] Here, the sense inventory is taken from WORDNET but Rothe and Schütze (2015) emphasize that any lexical or semantic resource could be used. Neelakantan et al. (2014) and Bartunov et al. (2016) present approaches that gather sense information in an unsupervised way from monolingual text by integrating the sense distinction into the learning process. We use ADAGRAM (Bartunov et al., 2016) as an additional baseline because it compares favorably to the model by Neelakantan et al. (2014). ADAGRAM's main parameter effectively regulates the maximum number of senses per word; the algorithm finds the number of senses automatically in this range, i.e. the parameter can be seen as a limit for the maximum number of induced senses.

Retrofitting is the process of augmenting a given item for a new task, i.e. in our case a post-processing objective that re-adjusts existing word embeddings (Faruqui et al., 2015). Multiple objectives have been defined on this account, e.g. Faruqui et al. (2015) or Kiela et al. (2015) use lexical resources, while, for instance, Wieting et al. (2015) directly optimizes paraphrase pair alignment from PPDB[2] (Ganitkevitch et al., 2013).

3. Methodology

In the remainder of this work we will use v to refer to a word and $\mathbf{v}$ to refer to v's corresponding word vector.

3.1. Unsupervised Sense Inventory

Our proposed method solely relies on pre-computed word-embeddings and a sense inventory resource. We follow the terminology in WORDNET and define a *synset* for a word v to be the set of related words that express the same concept, and the *sense inventory* of v to be the *collection of its synsets*, i.e. the different senses v can bear. Dorow and Widdows (2003), Pantel and Lin (2002), and more recently, Pelevina et al. (2016) use unsupervised WSI methods, which means they use or create so-called *unsupervised synsets* referring to sense-inventories, which were induced automatically from text. The simplified procedure to compute an unsupervised synset for a particular word v is as follows:

1. compute v's **top** n **nearest neighbors** (by some word-similarity notion)
2. compute a similarity score between every pairwise combination of nearest neighbors, which renders a fully connected **similarity graph**
3. compute a word **clustering**, where each cluster represents a different sense of v.

This general methodology has been proven to perform sufficiently well on a number of NLP tasks, whereas the details of this simplified procedure vary. The clustering represents the sense inventory (i.e. the collection of synsets) S_v for the word v; we refer to a particular synset or sense k of v as S_v^k. We want to stress that v is usually not contained in any of its "synsets", i.e. $S_v^k = S_v^k \setminus v$ per definition.

Following Riedl and Biemann (2017), we use an unsupervised sense inventory, pre-computed[3] by using the JOBIMTEXT (JBT) framework (Biemann and Riedl, 2013; Riedl, 2016), which can be seen as a symbolic count based model. JBT provides a graph-based sparse word similarity model, i.e. only words, and no vectors are provided. The Chinese Whispers (Biemann, 2006, CW) algorithm is used for inducing word senses based on ego networks weighted by context similarity.

3.2. Retrofitting Word Embeddings

The main goal of retrofitting word vectors is to find individual vector representations for each sense of a word. Using a sense inventory, word vectors from a particular synset are averaged, such that each sense of a word will be represented by a single individual vector. For a word v, we average all vectors of the top m words in a synset S_v^k and add the vector $\mathbf{v}$ with weight λ in order to compensate for semantic drift, for which we found strong indications in preliminary experiments:

$$\mathbf{v}_k = \lambda \mathbf{v} + (1-\lambda) \sum_{u \in \text{top}_m(S_v^k)} \mathbf{u}\,, \tag{1}$$

where λ is a scalar in $[0,1]$, $\mathbf{v}_k$ is the sense vector of the k^{th} sense of word v, and $\mathbf{u}$ is the word vector of word u. A geometric interpretation of this equation can be interpreted as to first find the center of a cluster of words in S_v^k and then shift the center by λ into the direction of the core word v. Note again that the clustering itself for any word v is performed without v itself, i.e. it is not contained S_v^k, cf. (Dorow and Widdows, 2003), hence the shifting. Using only the top m words stems from the fact that the clusters, i.e. the synsets, have different sizes. A common observation is that the largest clusters often refer to major senses and smaller clusters usually represent minor senses, i.e. senses that are underrepresented in the text corpus. To alleviate the effect of averaging noisy words in large clusters, we select only the top m words.

3.3. Sense-aware word-similarity

We tested different procedures for computing sense-aware similarities between any two words u and v with senses u_k and v_l:

$$\text{sim}(u,v) = \arg\max_k \ \cos(\mathbf{u}_k, \mathbf{v}) \tag{2}$$

$$\text{sim}(u,v) = \arg\max_l \ \cos(\mathbf{u}, \mathbf{v}_l) \tag{3}$$

$$\text{sim}(u,v) = \arg\max_{k,\,l} \ \cos(\mathbf{u}_k, \mathbf{v}_l)\,. \tag{4}$$

Equations (2-4) involve finding the closest senses k and l for the words u and v in vector space. We compare these measures to the standard, sense un-aware cosine similarity $\cos(\mathbf{u}, \mathbf{v})$.

[1] http://www.cis.lmu.de/~sascha/AutoExtend/

[2] The paraphrase database: http://www.cis.upenn.edu/~ccb/ppdb/.

[3] http://ltmaggie.informatik.uni-hamburg.de/jobimviz, (Ruppert et al., 2015)

4. Experimental Setup

4.1. Word-similarity Benchmark Datasets

Hill et al. (2014) raise the point that a strong distinction must be made between similarity and relatedness. While related words roughly fit into the same topic, similar words are more specific, they fit into the same topic and constitute (partial) substitutability. Consider for example the words *student* and *professor*, which are certainly considered related but not similar because there are only few contexts in which the two words can be exchanged, hence they are considered highly dissimilar due their antonymic nature while *teacher* and *professor* might be exchangeable, and are thus considered equally related but more similar.

The WORDSIM353 (Finkelstein et al., 2001) dataset provides relatedness scores of 353 noun pairs and the SIMLEX999 dataset provides similarity scores for 666 noun pairs, 222 verb pairs and 111 adjective pairs. Particularly the latter's emphasis is to model opposite meanings (antonym-like) as highly non-similar, e.g. *student* and *professor* have a low similarity score in SIMLEX999, but a high relatedness score in WORDSIM353.

Another dataset is the MEN[4] dataset (Bruni et al., 2014), which models, analog to WORDSIM353, relatedness or association rather than similarity. Bruni et al. (2014) randomly sampled $3,000$ word pairs from words that occur at least 700 times in the ukWaC + Wackypedia combined corpora.[5] MEN comprises of inter part-of-speech word pairs, e.g. pairs like (*apple-N*, *orange-A*) or (*bear-V*, *boxer-N*). It is also worthy to note that MEN comes in two forms, *a*) in a lemmatized form with POS tags, and *b*) in natural form. We report results on the lemma form with POS-tag information.

Another dataset, the SIMVERB dataset (Gerz et al., 2016), can be interpreted as a larger version of the verb part of SIMLEX999, containing $3,500$ verb pairs, allowing more meaningful benchmarking with more and better represented examples.

4.2. Embedding Matrices

WORD2VEC applies a neural language modeling approach, where the goal is to predict a word w_i at position i given its context c_i (CBOW) or vice versa (SGNS) (Mikolov et al., 2013). A projection matrix is learned during this process, which has been shown to be beneficial in various NLP tasks. We use pre-trained word vectors provided by Mikolov et al. (2013), which were trained on Google News texts containing 6 Billion words.[6] Additionally, we use the GLOVE[7] (Pennington et al., 2014, global vectors) embeddings.

Schwartz et al. (2015) defined the context of a word to be the *symmetric pattern* it occurs with, and applied WORD2VEC to those pairs. A symmetric pattern is a shallow pattern in the form of *'X or Y'*, *'X and Y'*, *'X as well as Y'*, *'X rather than Y'*, where particular instances of *X* and *Y* occur in both positions, e.g. *'cats and dogs'* and *'dogs and cats'* are considered instances of a symmetric pattern, while *'point of view'* for example cannot be altered without losing its meaning, *'X of Y'* is thus considered an asymmetric pattern. Some symmetric patterns are considered to be particularly indicative for antonymy, e.g. *'either X or Y'* or *'rather X than Y'* are typical to be filled by words with opposite, or strongly different meanings, e.g. *'either black or white'*. Schwartz et al. (2015) used symmetric patterns to build an antonym-sensitive embedding model from monolingual corpora. We use their 10K dimension model built on an 8G words corpus[8], and refer to this embedding type as SYMPAT. We also tested the 300 and 500 dimensional vectors provided by Schwartz et al. (2015) but the 10K version achieved the best results among the SYMPAT embeddings.

Wieting et al. (2015) used PPDB pairs to train a projection matrix called PARAGRAM. The matrices are initialized with the GLOVE embeddings and retro-fitted to match with PPDB. By using paraphrases obtained via round-trip translations, the model is already guided to represent synonymous expressions with similar vectors, as opposed to expressions with opposite meanings. Wieting et al. (2015) further tuned the hyper-parameters, resulting in PARAGRAMWS optimized on WORDSIM353 and PARAGRAMSL optimized for SIMLEX999.

The embeddings are thus tuned for either relatedness or similarity and constitute a strong baseline.

Additionally, we also make use of two LSA embeddings trained on English corpora provided by Günther et al. (2015).[9] Both models are based on a 2-Billion-word corpus and use a *positive pointwise mutual information* weighting scheme (PPMI) before applying singular value decomposition (SVD) with 300 target dimensions and a vocabulary of 100K words. We refer to the model based on a bag-of-word representation of documents as LSABOW, and to the model applying a HAL-like context representation[10] as LSAHAL, following the terminology of Günther et al. (2015).

Many other NN embedding models have been published, e.g. (Wieting et al., 2016; Recski et al., 2016; Mrkšić et al., 2016), however, we deliberately do not go into details here since these supervised models are out of the scope of this work; we focus on the relative improvement of monolingual embeddings by exploiting unsupervised WSI methods. We are thus independent of any manually developed resource and do not rely on the existence of parallel text.

5. Results

We follow previous work and use the Spearman rank-correlation coefficient ρ throughout the evaluation. We evaluated all datasets for all methods but restrict our discussion to the most interesting results. Selecting the top $m = 5$ cluster words for averaging proved most useful; in

[4] https://staff.fnwi.uva.nl/e.bruni/MEN

[5] http://wacky.sslmit.unibo.it/

[6] We use the 300 dimensional model trained on Google news. The model and the source code is available at https://code.google.com/p/word2vec/.

[7] We use the 6 Billion word, 300 dimensional model available at http://nlp.stanford.edu/projects/glove/.

[8] http://homes.cs.washington.edu/~roysch/papers/sp_embeddings/sp_embeddings.html

[9] Models are available for download under http://www.lingexp.uni-tuebingen.de/z2/LSAspaces/.

[10] HAL, hyperspace analogue to language; a context representation similar to the skip-gram notion

	AUTOEXTEND	ADAGRAM	SGNS	SGNS-S	GLOVE	GLOVE-S	SYMPAT	SYMPAT-S	LSA_{BOW}	LSA_{BOW-S}	LSA_{HAL}	LSA_{HAL-S}	PARAGRAMSL	PARAGRAMSL-S	PARAGRAMWS	PARAGRAMWS-S
SIMLEX999	0.45	0.29	0.44	0.46	0.37	**0.41**	0.54	0.55	0.30	**0.39**	0.27	**0.38**	0.68	0.64	0.66	0.64
MEN	0.72	0.67	0.77	0.78	0.73	**0.77**	0.53	**0.68**	0.67	**0.70**	0.71	**0.74**	0.77	**0.80**	0.80	0.81
SIMVERB	0.43	0.27	0.36	**0.39**	0.23	**0.30**	0.37	**0.45**	0.15	**0.22**	0.19	**0.28**	0.53	0.53	0.51	0.50
WORDSIM353	0.58	0.61	0.70	0.69	0.61	**0.65**	0.47	**0.62**	0.67	0.66	0.59	**0.63**	0.72	0.73	0.77	0.75
SIMLEX999-N	0.44	0.33	0.45	**0.50**	0.39	**0.47**	0.48	**0.55**	0.32	**0.46**	0.34	**0.44**	0.68	0.66	0.64	0.64
MEN-N	0.72	0.68	0.77	0.79	0.76	**0.80**	0.57	**0.74**	0.71	0.73	0.73	**0.76**	0.78	**0.81**	0.80	0.82

Table 1: Spearman correlation scores on the different datasets and embeddings. Sense-aware similarities are marked with '-S'. The best performing method is underlined or marked bold. We distinguish underlined values to be the winning system with a slight margin (< 0.03) and bold face values with a larger margin. We marked PARAGRAMSL and PARAGRAMWS for SIMLEX999 and WORDSIM353 in gray, since the method's hyperparameters were optimized on the respective dataset, thus, the results are not comparable. The lower part evaluates only the noun pair parts of the datasets, as indicated by -N.

our experiments we found fluctuating best performing values between top 3 and top 10, with 5 always being among the best values. Also, Equation (4) distinguished itself as the best performing method with $\lambda = 0.5$. Other similarity computations, Eq. (2;3), perform non-satisfactory, sometimes even with a decline in performance. In the remainder of this work we refer to embeddings with the suffix -S to the sense-aware similarities which performed best in our previous experiments using the fixed parameters $m = 5$ and $\lambda = 0.5$.

We report AUTOEXTEND (Rothe and Schütze, 2015) and ADAGRAM (Bartunov et al., 2016) scores for comparison. Table 1 shows the final results using sense-unaware similarities, i.e. standard cosine similarity, and our new sense-aware similarities based on the JBT sense inventory.

The results clearly show that sense-aware similarities perform consistently better or comparable to their sense-unaware counterparts. The average improvement for most sense-unaware systems to their sense-aware counterpart is roughly between 0.02 and 0.05 points of spearman correlation. Particularly, previously inferior embeddings, e.g. GLOVE or both LSA embeddings, gain most and more consistent from this representation. The loss of performance with the PARAGRAM family of embeddings is mainly due to the fact that they already have been optimized for synonymy and antonymy. Injecting the JBT sense inventory – which has no special treatment for antonyms – attracts related terms, i.e. apparently antonymous, non-similar, but related words. In fact, this happens to a large extent on adjectives, causing the largest losses. When looking at the performance for nouns (lower part of Table 1 for datasets where nouns were available), we see consistent improvements across all datasets.

We observe minor sense selections in $3,953$ out of $7,734$ examples across all datasets for SGNS-S, that is roughly 52%. Summarizing, in about half of the example word pairs a minor sense was selected. This is most consistent across nouns, and varies for verbs and adjectives, which could be attributed to coverage issues[11], or inadequate clusterings for adjectives and verbs, since the JBT sense clustering mainly focusses on nouns.

For illustration of adequacy, consider the word pair (*iron, vitamin*) taken from the SIMLEX999 dataset. Figure 1 provides details for this example word pair, which includes all scores and a description of the induced sense inventory. We can see that the manually assigned SIMLEX999 score is in the mid-range (5.55 out of 10), standard cosine similarity ranks[12] this example at position 212 with a similarity score of 0.22, which is rather low. This can be verified by looking at the figure, i.e. on the innermost unit circle, the angle between *vitamin* and *iron* is quite large. The sense-aware similarity score selects a link between two suitable minor senses. The visualization shows the two words and their cluster terms, as well as the averaged cluster centers on the unit circle. The projection was done with T-SNE (Maaten and Hinton, 2008). For better illustration, we mapped cluster terms for each word on a different circle, but note that each circle preserves directions and represents a scaled unit circle. In this visualization, it is easily recognizable that the vectors for *iron* and *vitamin* are far apart, whereas the retrofitted vectors *iron2* and *vitamin3* are close by in terms of their cosine similarity.

We computed cross correlation scores between the methods, e.g. the Spearman correlation score between SGNS and SGNS-S embeddings yields $\rho = 0.85$. This suggests that the individual scores differ, although final SIMLEX999 correlation scores do not seem to benefit drastically (e.g. $+0.02$ difference for SGNS to SGNS-S).

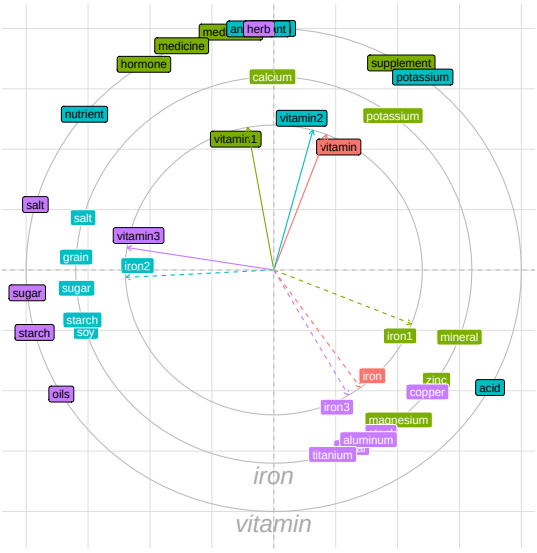
6. Native Sense Clustering

In order to make the retrofitting process independent of external (be it induced or manually compiled) sense inventories, we build a sense-inventory directly from word embeddings and provide anecdotal evidence and insights of its failure. For proof of concept we focus on a single NN embedding, for which we chose the SGNS word embedding matrix because of its popularity. We follow the general

[11] Coverage is around 98% for SYMPAT and 99% for others.

[12] Note that Spearman correlation compares ranks.

SimLex999 / rg(SimLex999)	5.55 / 622
SGNS / rg(SGNS)	0.22 / 212
SGNS-S / rg(SGNS-S)	0.59 / 488
$\lvert S^k_{\texttt{iron}} \rvert$	$\{88, \underline{5}, 98\}$
$\lvert S^k_{\texttt{vitamin}} \rvert$	$\{59, 88, \underline{53}\}$



$S_{\texttt{iron}} = \{ \{\texttt{zinc}, \texttt{calcium}, \texttt{magnesium}, \texttt{potassium}, \texttt{mineral}\}, \underline{\{\texttt{sugar}, \texttt{salt}, \texttt{grain}, \texttt{soy}, \texttt{starch}\}}, \{\texttt{steel}, \texttt{metal}, \texttt{copper}, \texttt{aluminum}, \texttt{titanium}\} \}$

$S_{\texttt{vitamin}} = \{ \{\texttt{supplement}, \texttt{hormone}, \texttt{pill}, \texttt{medication}, \texttt{medicine}\}, \{\texttt{nutrient}, \texttt{calcium}, \texttt{antioxidant}, \texttt{acid}, \texttt{potassium}\}, \underline{\{\texttt{sugar}, \texttt{salt}, \texttt{oils}, \texttt{starch}, \texttt{herb}\}} \}$

Figure 1: Scores achieved by sense-aware and sense-unaware word similarity computation for the word pair (*iron, vitamin*). rg(·) refers to the rank regarding the SimLex999 dataset; selected senses by the method are underlined. The visualization is based on terms on the unit circle. Every circle represents a unit circle in a joint plot for illustration purposes. The inner circle shows the different sense vectors as well as the original word vectors, the middle circle shows synset terms generated by the word *iron*, and the outer circle represents synset terms generated by the word *vitamin*.

procedure to build sense inventories as explained in Section 3.1., i.e. for a particular word v, we first create a similarity matrix M of the top k nearest neighbors[13] in terms of cosine similarity, we then applied a clustering algorithm to the similarity matrix M, which yields a clustering of words that can be directly interpreted as the sense-inventory for v: S_v.

[13]For proof of concept, we used $k = 500$, which is commonly known to be reasonable value.

	SGNS		JBT		
related term	cos	rg(·)	#ctx	rg(·)	sense description
putter	0.46	17	36	128	golf sports
wood	0.47	11	119	15	
copper	0.37	252	206	9	metallic elements
aluminum	0.35	427	206	8	
salt	0.23	23731	31	158	nutrition
fiber	0.20	47072	77	38	
steam	0.12	416270	28	181	smoothing clothes
shirt	0.12	415080	–	–	

Table 2: Cosine similarity (cos) and similarity by number of shared contexts (#ctx), next to the relative rank regarding cos for SNGS and #ctx for JBT with respect to the query word '*iron*'.

Since words cannot be expected to have a fixed number of senses, we tested two *graph based clustering algorithms*, where the number of clusters, i.e. the number of senses, is not a parameter but will be determined by the algorithms themselves. Because of its symmetry, M can be directly interpreted as an adjacency matrix for an undirected graph. We experimented with the following graph clustering algorithms: 1.) CW: Chinese Whispers (Biemann, 2006); and 2.) MCL: Markov Clustering (van Dongen, 2000).[14] In general, graph clustering algorithms perform best if the adjacency matrix of the graph is sparse. In order to sparsify M, we prune by a threshold parameter τ_{cos}, i.e. we set values $M_{ij} = 0$ if M_{ij} is lower than τ_{cos}. Apart from that, we use the default parameter settings suggested by Biemann (2006) for CW or van Dongen (2000) for MCL. As a post-clustering step, singleton clusters are merged into one 'residual' cluster, i.e. clusters which contain only a single element – which occur frequently for large τ_{cos} – eventually form the 'residual' sense.

Results by Anecdotal Evidence: Cleary, the parameter k, which defines the top k nearest neighbors of a word v, and thus the size of M, implicitly also controls the vocabulary of the sense inventory of v. Manual inspection of those nearest terms revealed, that in case of SGNS and other NN word embeddings, the immediate neighborhood of a word v consists mainly of one dominating sense. For illustrative purposes, consider the example given in Table 2, where we highlight scores and ranks for the polysemous word *iron* with regards to some hand-selected words representing different senses of *iron*. Here, mainly terms referring to a *golf sports* related sense can be found in the immediate vicinity of *iron* (large cosine similarity, small rank), while other terms referring to common senses are

[14]Note that we also tested other clustering algorithms, such as *K-Means* and *Self-Organizing-Maps* for comparison purposes, but we report results only for CW and MCL since they do not depend on the number of clusters as parameter input and yield visually better clusters.

	τ_{cos}						
	0.3	0.4	0.5	0.6	0.7	0.8	0.9
MCL	1.7	13.5	34.5	50.4	38.5	10.2	2.4
CW	1.4	3.0	7.9	16.2	15.0	4.0	1.5

Table 3: Average number of clusters for all words in all datasets, varying $\tau_{\cos}$, the two graph clustering algorithms, based on SGNS embedding vectors.

	SIMLEX999
SGNS-S$_{clnat}$	0.42
SYMPAT-S$_{clnat}$	0.48
PARAGRAMSL-S$_{clnat}$	0.67
PARAGRAMWS-S$_{clnat}$	0.57

Table 4: Selected results for native sense induction by clustering on four embeddings and SIMLEX999.

farther away (small cosine similarity, large rank). The selected terms for representative senses seem to have a similar cosine similarity to $iron$ though. This is not an isolated incident, we have observed this effect consistently for multiple polysemous terms.[15] This suggests the confirmation of the observations which Faruqui et al. (2016) or Schnabel et al. (2015) already noted: Within neural word embeddings, the frequency rank of a word's neighbor strongly depends on the frequency of the word itself. This is clearly an issue because the frequency of a word's sense naturally correlates with the frequency of a word's occurrence.

Table 3 shows the average number of clusters for all words across all datasets for varying τ_{cos}. Based on those results, we fix $\tau_{cos} = 0.8$ and CW, as this best resembles the sense inventory of the JBT resource, where also CW is used, producing 3.73 senses on average. Selected results of the native clustering compared to SIMLEX999 in Table 4 show a decline in performance w.r.t. the sense-unaware similarity values in Table 1. Failure can be attributed to the local structure of the neighborhood as explained above.

7. Conclusion

We confirmed our initial hypothesis that 'sense inventories do help for word similarity' and presented consistent improvements over all tested embeddings and datasets using pre-existing sense-inventory resources. This holds particularly for embeddings trained on monolingual text. On a general level, we have shown how to operationalize word sense induction for a semantic task, here for word similarity, by creating appropriate representations of words for the task on top of generic, previously available, representations. Contrary to most prior work in this area, we did not use manually-defined sense inventories or lexical resources, but an unsupervised graph-based sense induction scheme. Additionally, we confirmed prior findings and conclude that direct clustering of a word's nearest neighbors in an NN embedding is not helpful for WSI, but other methodologies are required here. The source code as well as the sense aware vectors for the datasets are provided as open source software under a permissive license.[16] We would like to follow up on this line of work and devise similar schemes for relation extraction, learning of semantic hierarchies, and short text similarity.

[15] We leave a more thorough analysis for future work.

[16] https://github.com/uhh-lt/senseasim

8. References

Bartunov, S., Kondrashkin, D., Osokin, A., and Vetrov, D. (2016). Breaking sticks and ambiguities with adaptive skip-gram. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 130–138, Cadiz, Spain.

Biemann, C. and Riedl, M. (2013). Text: Now in 2D! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Biemann, C. (2006). Chinese whispers - an efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of TextGraphs: the First Workshop on Graph Based Methods for Natural Language Processing*, pages 73–80, NY, USA.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022.

Bruni, E., Tran, N. K., and Baroni, M. (2014). Multimodal Distributional Semantics. *Journal of Artificial Intelligence Research*, 49(1):1–47.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, MD, USA.

Dorow, B. and Widdows, D. (2003). Discovering corpus-specific word senses. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics*, pages 79–82, Budapest, Hungary.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., and Smith, N. A. (2015). Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, CO, USA.

Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 30–35, Berlin, Germany.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. The MIT Press.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414, Hong Kong, Hong Kong.

Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2013). PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764, Atlanta, GA, USA.

Gerz, D., Vulić, I., Hill, F., Reichart, R., and Korhonen, A. (2016). Simverb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182, Austin, TX, USA.

Günther, F., Dudschig, C., and Kaup, B. (2015). LSAfun - An R package for computations based on Latent Semantic Analysis. *Behavior Research Methods*, 47(4):930–944.

Hill, F., Reichart, R., and Korhonen, A. (2014). SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hofmann, T. (1999). Probabilistic Latent Semantic Analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 289–296, Stockholm, Sweden.

Kiela, D., Hill, F., and Clark, S. (2015). Specializing word embeddings for similarity or relatedness. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2044–2048, Lisbon, Portugal.

Landauer, T. K. and Dumais, S. T. (1997). A solution to Plato's problem: the Latent Semantic Analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using {t-SNE}. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Machine Learning*, ICLR 2013, pages 1310–1318, Scottsdale, AZ, USA.

Mrkšić, N., Ó Séaghdha, D., Thomson, B., Gašić, M., Rojas-Barahona, L. M., Su, P.-H., Vandyke, D., Wen, T.-H., and Young, S. (2016). Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 142–148, San Diego, CA, USA.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10:1–10:69.

Neelakantan, A., Shankar, J., Passos, A., and McCallum, A. (2014). Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar.

Pantel, P. and Lin, D. (2002). Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD 2002, pages 613–619, Edmonton, AL, Canada.

Pelevina, M., Arefiev, N., Biemann, C., and Panchenko, A. (2016). Making sense of word embeddings. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 174–183, Berlin, Germany.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar.

Recski, G., Iklódi, E., Pajkossy, K., and Kornai, A. (2016). Measuring semantic similarity of words using concept networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 193–200, Berlin, Germany.

Riedl, M. and Biemann, C. (2017). There's no 'count or predict' but task-based selection for distributional models. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS)*, Montpellier, France.

Riedl, M. (2016). *Unsupervised Methods for Learning and Using Semantics of Natural Language*. Ph.D. thesis, Technische Universität, Darmstadt.

Rothe, S. and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1793–1803, Beijing, China.

Ruppert, E., Kaufmann, M., Riedl, M., and Biemann, C. (2015). Jobimviz: A web-based visualization for graph-based distributional semantic models. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 103–108, Beijing, China.

Schnabel, T., Labutov, I., Mimno, D., and Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307, Lisbon, Portugal.

Schwartz, R., Reichart, R., and Rappoport, A. (2015). Symmetric pattern based word embeddings for improved word similarity prediction. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 258–267, Beijing, China.

van Dongen, S. M. (2000). *Graph Clustering by Flow Simulation*. Ph.D. thesis, Universiteit Utrecht.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2015). From paraphrase database to compositional paraphrase model and back. *Transactions of the Association for Computational Linguistics*, 3:345–358.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2016). Charagram: Embedding words and sentences via character n-grams. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1504–1515, Austin, TX, USA.

fastSense: An Efficient Word Sense Disambiguation Classifier

Tolga Uslu, Alexander Mehler, Daniel Baumartz, Wahed Hemati
TTLab, Goethe University Frankfurt
Frankfurt, Germany
{uslu, mehler, baumartz, hemati}@em.uni-frankfurt.de

Abstract

The task of *Word Sense Disambiguation* (WSD) is to determine the meaning of an ambiguous word in a given context. In spite of its importance for most NLP pipelines, WSD can still be seen to be unsolved. The reason is that we currently lack tools for WSD that handle big data – "big" in terms of the number of ambiguous words and in terms of the overall number of senses to be distinguished. This desideratum is exactly the objective of `fastSense`, an efficient neural network-based tool for word sense disambiguation introduced in this paper. We train and test `fastSense` by means of the disambiguation pages of the German Wikipedia. In addition, we evaluate `fastSense` in the context of Senseval and SemEval. By reference to Senseval and SemEval we additionally perform a parameter study. We show that `fastSense` can process huge amounts of data quickly and also surpasses state-of-the-art tools in terms of F-measure.

Keywords: WSD, Big Data, Wikipedia

1. Introduction

One of the core tasks in natural language processing is *Word Sense Disambiguation*. Without disambiguation, we just consider a word as a combination of characters and not the meaning behind it. Without properly disambiguating the lexical constituents of a text, it is almost impossible to process its content automatically. Our goal is to solve this problem and to make it applicable to large amounts of data. To this end, we present a neural network-based classifier for WSD called *fastSense*. We take sequences of words as input and compute a sense label per ambiguous word in that sequence as output. This approach was motivated by the classifier called `fastText` (Joulin et al., 2016). As the name suggests, `fastText` is designed to perform text classifications as quickly as possible. However, `fastText` is not suitable for disambiguating words. In addition, the neural network used by `fastText` does not support training of multi-labels. Therefore, we implemented our own word embedding-based neural network by analogy to the architecture of `fastText`. This allows us to apply `fastSense` to WSD efficiently even on big data. In order to test the time complexity of our approach, we created a disambiguation corpus from the German Wikipedia with over 50,000,000 training and test sets. We use the disambiguation pages and the link structure of Wikipedia to match words with their corresponding Wikipedia senses. In this paper, we deal with the German Wikipedia. In terms of size or space complexity, its sense model is far beyond what is normally studied, for example, in the framework of Senseval or SemEval. However, in order to show that our approach is language independent, we additionally perform multiple tests related to Senseval and SemEval. These tests show that our model keeps up with state-of-the-art tools by reaching 73.47% at Senseval-2, 73.48% at Senseval-3 and up to 87,57% on SemEval 2007 tasks.

The paper is structured as follows: In Section 2., we contrast `fastSense` with related approaches to WSD. In Section 3. we introduce the architecture of `fastSense`. In Section 4., we explain the experiments carried out to evaluate `fastSense` and show the results achieved by it. In Section 4.3., we discuss our findings and in Section 5., we give a summary of the paper.

2. Related Work

Our approach is motivated by `fastText` (Joulin et al., 2016). This relates to the very efficient and successful way by which `fastText` allows for classifying data. The main purpose of `fastText` is text classification. Its architecture is similar to `word2vec` (Mikolov et al., 2013): both approaches are based on a bag-of-words model. Further, both of them use a single hidden layer. The difference between `word2vec` and `fastText` is that the latter requires to define a label for any input text, while `word2vec` uses context windows of lexical units to predict single words or vice versa. We transpose `fastText` to word sense disambiguation in order to efficiently determine the meaning of ambiguous words even in cases in which we face big data. By this we mean scenarios in which hundreds of thousands of different words are ambiguous.

`fastSense` is characterized by its simplicity, speed and quality. This distinguishes it from similar tools. For instance, (Mihalcea and Csomai, 2007; Ferragina and Scaiella, 2010; Ratinov et al., 2011b; Ratinov et al., 2011a; Agerri et al., 2014; Moro et al., 2014) present approaches to *Entity Linking*. More specifically, they link tokens in texts to knowledge databases such as DBpedia, Wikipedia or WordNet to identify instances of entities. These approaches are similar to ours, with the difference that we focus on ambiguous words, while the latter approaches also link words that have only one meaning. The disadvantage of these approaches is their speed. For large amounts of data, they may take weeks to produce an output (see Table 2 for an estimation of this time effort). (Mihalcea, 2007) uses a technique similar to the one presented here to build a sense-tagged Wikipedia corpus using the link structure of Wikipedia to match senses. However, this corpus has not been used to disambiguate ambiguous words according to Wikipedia's disambiguation pages, but to com-

pare them with the data of Senseval 2. (Mihalcea et al., 2004) use a PageRank algorithm operating on semantic networks to perform WSD. The underlying network is spanned by means of semantic relations of synsets, entailment and other WordNet relations. The PageRank algorithm assigns scores to words and chooses the disambiguating synset of highest score. (Yuan et al., 2016) present two WSD algorithms, achieving the best results by means of a semi-supervised algorithm combining labeled sentences with unlabeled ones and propagating labels based on sentence similarity. (Tripodi and Pelillo, 2016) describe an approach to WSD based on evolutionary game theory, in which words tend to adapt senses of their neighborhood so that WSD is reformulated as a kind of constraint satisfaction. (Zhong and Ng, 2010) present a framework for English all-words WSD. It disambiguates each content word of a given sentence using a linear kernel-based SVM (Joachims, 2002). (Iacobacci et al., 2016) show that the use of word embeddings achieves an improvement in WSD compared to standard features. (Chaplot et al., 2015) propose a graph based unsupervised WSD system which requires WordNet, a dependency parser and a POS-Tagger. They model WSD as a maximum-a-posteriori inference query operating on a Markov random field. (Raganato et al., 2017a) define WSD in terms of a sequence learning problem. This is done by means of a bidirectional LSTM-based neural network (Hochreiter and Schmidhuber, 1997). (Melamud et al., 2016) present `context2vec` which is also based on bidirectional LSTMs for learning disambiguating word contexts.

Unlike these approaches, we present a method that can handle big data: in terms of the number of senses to be distinguished *and* in terms of the number of units to be disambiguated. On the one hand, knowledge driven approaches using, for example, WordNet and related resources are limited in terms of the number of senses distinguished by them. GermaNet, for example, distinguishes 33,630 senses of 13,445 ambiguous words – that is much less than considered by us. On the other hand, approaches that rely on algorithms like PageRank or classifiers like SVMs or LSTMs are limited in terms of their time efficiency: it is a computational challenge to maintain, for example, SVMs for each of the 825,179 senses of the 221,965 ambigous words of the German Wikipedia which, however, are easily covered by our approach. Thus, we are in need of a flexible, easy-to-compute, but efficient method for WSD as presented in the next section.

3. Model architecture

During training, `fastSense` requires text as input and the corresponding senses as output (see Figure 1). Its single hidden layer is an embedding layer in which word indexes from the input layer are converted into word vectors. More specifically, the number of hidden nodes corresponds to the dimension of the pre-trained word embedding vectors so that the weights of edges between input and hidden nodes correspond to the respective coordinates of the latter vectors. We computed the word embeddings by means of `word2vec` (Mikolov et al., 2013) using Wikipedia as the underlying corpus. The embeddings are then merged in the hidden layer according to the *global averaging pooling principle* (Lin et al., 2013).

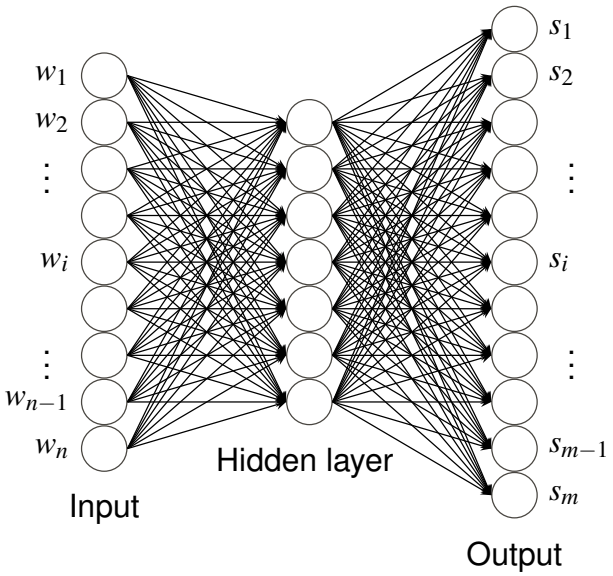


Figure 1: Model architecture of `fastSense`.

To support multi-label training we used the *sigmoid* function as an activation function of the output layer. For the sake of optimizing, `Adamax`, a special variant of `Adam` (Kingma and Ba, 2014), is a very efficient choice in practice. It uses the *infinity norm*, which makes it possible to stabilize the training over longer periods of time and, thus, to achieve faster and better results. To prevent overfitting, we used *Dropout* (Srivastava et al., 2014) as regularization method. Dropout removes nodes during each training session, ignores them and does not train with them. After the training process, the nodes are reinserted with their original weights.

To apply this model to WSD, we additionally developed a method for post-processing the output of the neural network. Usually, the sense of highest probability is selected as output. However, since the output layer contains all senses of all ambiguous words, it is unlikely that the target sense of a word x to be disambiguated equals the top ranked sense. Thus, we do not necessarily select the label of highest probability, but go through the list of rank-ordered candidates until the first occurrence of x tagged by a corresponding sense number is reached. This sense unit is then produced as the output of disambiguating x. As an example, consider processing the ambiguous word *bank* as depicted in Figure 2: when observing an occurrence of this word in a sentence about a *financial* topic, a classifier like `fastText` will likely suggest topic labels such as *finance*, *money* or *financial institute* because of the fact that the input sentence is about such a topic. However, what we are looking for is the *sense of the word* and not the most strongly associated topic label. Thus, we descend the sorted output of `fastSense` given the input sentence until we reach a candidate sense prefixed by *bank* (i.e., *bank_2*) that is taken to predict the sense of the word in this sentence. In this way, `fastSense` can be used as a tool for WSD. Conversely speaking, we reconstructed WSD as a kind of topic

labeling that is performed by fastSense by analogy to fastText.

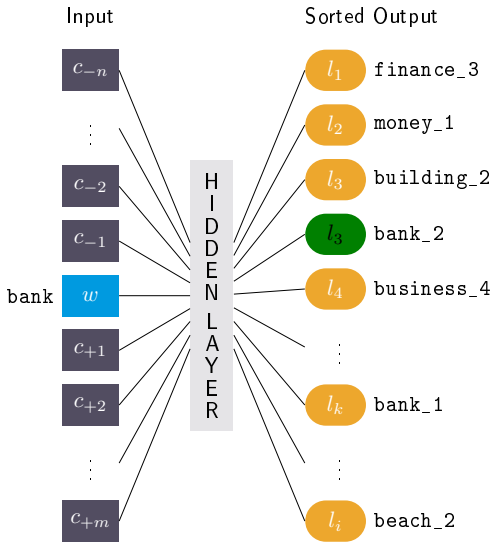


Figure 2: fastSense in work: *bank_2* is selected as the first occurrence prefixed by *bank*.

4. Experiment

We perform two disambiguation experiments. The first one uses the German Wikipedia to demonstrate the efficiency of fastSense. The second one is based on Senseval and SemEval. It aims at comparing fastSense with state-of-the-art tools. Table 1 lists the parameters used for these evaluations.

Short	Description
POS	Part of speech is considered.
Lemma	Lemma information is considered.
Token	Token information is considered.
WP	POS information is added.
x-Nb	*x* neighbors (left and right) are considered as context.
MinContext(k)	Any input text must contain at least k tokens to be used in training or testing.

Table 1: Parameters used for evaluating fastSense.

4.1. Wikipedia-based Disambiguation

In order to show that our approach allows for capturing large amounts of data, we created a corpus using the disambiguation pages of the whole German Wikipedia. For preprocessing this data we used the TextImager (Hemati et al., 2016) pipeline. Every word listed on a disambiguation page in Wikipedia corresponds to a different meaning of the corresponding lemma (page title). In total, we processed 221,965 disambiguation pages related to 825,179 senses. On average, this gives 3.72 senses per word. The disambiguation page *Bank*[1], for example, distinguishes 42 senses. For classification, we take each paragraph in Wikipedia that contains one of the target words to be disambiguated as a sample for training semantic representations of these words. This results in a corpus of 55,796,534 instances (49.9 GB) . Using this corpus, we trained fastSense on 51,067,054 instances (46GB) and tested it on the remaining 4,729,480 instances (3.9GB). The test takes only 20 minutes on a single thread. This runtime can be further reduced linearly by distributing fastSense over different threads. The results of the Wikipedia-based evaluation are shown in Table 3.

We compared several entity linking tools with fastSense in terms of time expenditure. With the same hardware, these tools take 6 to 188 days to process our test set. The effort was estimated based on a subset of elements as documented in Table 2. Obviously, fastSense outperforms these competitors. However, since these tools link to different resources (e.g., DBpedia, WordNet or Wikipedia), this comparison only holds for time effort.

Tools	1	500	1000	5000	4729480
Wikifier	0:01	8:20	16:41	1:24:06	≈55 days
Illinois	3:28	5:05	6:53	24:40	≈14 days
IXA	0:03	28:20	58:49	4:47:32	≈188 days
Babelfy	0:01	0:55	1:50	-	≈6 days
TAGME	1:19	3:20	5:42	28:40	≈18 days
fastSense	0:07	0:07	0:10	0:13	20:46

Table 2: Runtime-related evaluation regarding similar tools using 1, 500, 1000 and all test instances.

Type	Min-Context	F1-score
fastSense	1	0.735
fastSense	2	0.778
fastSense	5	0.810
fastText	1	0.071
MFS Baseline	1	0.627

Table 3: Wikipedia-based evaluation of fastSense in comparison to fastText and the most frequent sense (MFS) baseline. We used a learning rate of 0.05, a hidden layer size of 10 and 5 training epochs.

4.2. Senseval and SemEval related Disambiguation

SemCor (Mihalcea, 2016) provides texts with semantically annotated WordNet senses, which are automatically mapped to WordNet. We trained on SemCor 3.0 for performing Senseval and SemEval related tests. Because of the small amount of data provided by this corpus (234,136 disambiguated words), we were able to perform a parameter study to search for the best performing parameter settings. Candidates for feature selection are POS, token,

[1] https://de.wikipedia.org/wiki/Bank_(Begriffskl%C3%A4rung)

lemma and combinations thereof (see Table 1). Next, we tested different word context sizes (Context), word-n-grams (NGrams), learning rates (LR), dimensions of the hidden layer (Dim) and the number of epochs (Epoch). After each optimization step, we used the best performer of the preliminary round (see Table 4).

1. Type	SE2
Token WP	**0.703**
Lemma WP	0.697
POS	0.662
Token	0.701
Lemma	0.699

2. Context	SE2
S	0.703
1-Nb	**0.732**
2-Nb	0.718
3-Nb	0.720
4-Nb	0.706

3. NGrams	SE2
1	0.723
2	0.730
3	**0.732**

4. LR	SE2
0.025	0.718
0.05	0.724
0.1	**0.732**
0.5	0.724

5. Dim	SE2
5	0.710
10	**0.732**
25	0.711
50	0.712

6. Epoch	SE2
5	0.727
10	0.732
15	**0.735**
25	0.724
50	0.719

Table 4: Parameter study based on Senseval 2 and SemCor 3.0.

After completion we applied the optimal settings on Senseval 2 (English all-words) (SE2) and Senseval 3 (English all-words) (SE3). We also tested `fastSense` on SemEval-2007 Task 17 Subtask 1 (SE7) and Subtask 3 (SE7'), SemEval-2013 Task 12 (SE13) and SemEval-2015 Task 13 (SE15).
Since no information about lemmas or POS was given in the SE7 test sets, we carried out these experiments only on token basis. The results are listed in Table 5 and are compared to those of state-of-the-art tools in Table 6.

Input	SE2	SE3	SE7	SE7'	SE13	SE15
Token WP	0.735	0.735	–	–	0.662	0.732
Token	–	–	0.876	0.624	–	–

Table 5: F1-scores of the Senseval/SemEval-related tasks.

4.3. Discussion

We successfully used Wikipedia as a disambiguation corpus and show that `fastSense` can handle such a large amount of data (see Table 3). `fastSense` not only stands out for its speed, but also for the quality of its classification. We outperform the baseline considerably and show that similar approaches cannot keep up with `fastSense` in terms of runtime. Thus `fastSense` is a step towards performing WSD in relation to large amounts of data.
Since the SemCor data is many times smaller than the data derived from Wikipedia, we were able to carry out a parameter study in the Senseval- and SemEval-related experiments. Most interesting is our finding concerning the size of the context window. Using one neighbor of a word as context (1-Nb) and word-3-grams perform best. We also found that token in combination with POS-related information (see the parameter list in Table 1 and Table 4) perform best.
Note that `fastSense` is comparably fast: it takes only 20 minutes for disambiguating 4,729,480 instances on a single thread – that is, ca. 3,941 senses per second. Further, as mentioned in Section 3., `fastText` is less suited for WSD; accordingly, it performs worse compared to `fastSense` (see Table 3). In this sense, though being based on a related architecture, `fastSense` better fits the needs of WSD.

Model	SE2	SE3	SE7	SE7'	SE13	SE15
Iacobacci, 2016	0.634	0.653	**0.894**	0.578	**0.673**[2]	0.715[2]
Tripodi, 2016	0.660	0.647	0.828	0.565	–	–
Yuan, 2016	**0.736**	0.692	0.828	0.642	0.670	–
Chaplot, 2015	0.605	0.586	–	0.506	–	–
Zhong, 2010	0.625[1]	0.650[1]	0.879[1]	0.565[1]	0.653[2]	0.695[2]
Raganato, 2017	0.720	0.702	–	**0.648**	0.669	0.724
Melamud, 2016	0.718[2]	0.691[2]	–	0.613[2]	0.656[2]	0.719[2]
`fastSense`	0.735	**0.735**	0.876	0.624	0.662	**0.732**

[1] (Iacobacci et al., 2016) [2] (Raganato et al., 2017b)

Table 6: Comparison of state-of-the-art WSD tools on the Senseval 2, 3 and SemEval 7, 13 and 15 tasks using SemCor for training.

5. Conclusion

We presented a novel approach to word sense disambiguation called `fastSense`. We tested this model in the framework of Senseval and SemEval tasks as well as in terms of a big data-experiment based on the German Wikipedia. We achieve an F-score of up to 81.00% using the Wikipedia-based data. Further, we achieved 73.47% and 73.48% on Senseval 2 and 3, 87.57% and 62.40% on SE7 and SE7' and also 66.20% and 73.20% on SE13 and SE15. We show that `fastSense` cannot only work with huge data sets, but also surpasses state-of-the-art tools. Future work will address a parameter study on the Wikipedia-based data sets derived from different language releases. We will also account for dependency structures of sentences to gain an additional source of information for WSD. Our tool and all Wikipedia-based training and test data used in this paper will be made available open source on GitHub.

6. Bibliographical References

Agerri, R., Bermudez, J., and Rigau, G. (2014). Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *LREC*, volume 2014, pages 3823–3828.

Chaplot, D. S., Bhattacharyya, P., and Paranjape, A. (2015). Unsupervised word sense disambiguation using markov random field and dependency parser. In *AAAI*, pages 2217–2223.

Ferragina, P. and Scaiella, U. (2010). Tagme: on-the-fly annotation of short text fragments. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1625–1628. ACM.

Hemati, W., Uslu, T., and Mehler, A. (2016). Textimager: a distributed uima-based system for nlp. In *Proceedings of the COLING 2016 System Demonstrations*. Federated Conference on Computer Science and Information Systems.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2016). Embeddings for word sense disambiguation: An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Joachims, T. (2002). *Learning to classify text using support vector machines*. Kluwer, Boston.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

Melamud, O., Goldberger, J., and Dagan, I. (2016). context2vec: Learning generic context embedding with bidirectional lstm. In *CoNLL*, pages 51–61.

Mihalcea, R. and Csomai, A. (2007). Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM.

Mihalcea, R., Tarau, P., and Figa, E. (2004). Pagerank on semantic networks, with application to word sense disambiguation. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1126. Association for Computational Linguistics.

Mihalcea, R. (2007). Using wikipedia for automatic word sense disambiguation. In *HLT-NAACL*, pages 196–203.

Mihalcea, R. (2016). Rada mihalcea. `http://web.eecs.umich.edu/~mihalcea/downloads.html#semcor`. Accessed: 2016-12-05.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Raganato, A., Bovi, C. D., and Navigli, R. (2017a). Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1167–1178.

Raganato, A., Camacho-Collados, J., and Navigli, R. (2017b). Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proc. of EACL*, pages 99–110.

Ratinov, L., Roth, D., Downey, D., and Anderson, M. (2011a). Local and global algorithms for disambiguation to wikipedia. In *ACL*.

Ratinov, L., Roth, D., Downey, D., and Anderson, M. (2011b). Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1375–1384. Association for Computational Linguistics.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

Tripodi, R. and Pelillo, M. (2016). A game-theoretic approach to word sense disambiguation. *arXiv preprint arXiv:1606.07711*.

Yuan, D., Doherty, R., Richardson, J., Evans, C., and Altendorf, E. (2016). Word sense disambiguation with neural language models. *arXiv preprint arXiv:1603.07012*.

Zhong, Z. and Ng, H. T. (2010). It makes sense: A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83. Association for Computational Linguistics.

Text Annotation Graphs: Annotating Complex Natural Language Phenomena

Angus G. Forbes*, Kristine Lee†, Gus Hahn-Powell‡, Marco A. Valenzuela-Escárcega‡, Mihai Surdeanu‡

*University of California, Santa Cruz, angus@ucsc.edu
†University of Illinois at Chicago, khlee2@uic.edu
‡University of Arizona, {hahnpowell,marcov,msurdeanu}@email.arizona.edu

Abstract

This paper introduces a new web-based software tool for annotating text, *Text Annotation Graphs*, or *TAG*. It provides functionality for representing complex relationships between words and word phrases that are not available in other software tools, including the ability to define and visualize relationships between the relationships themselves (semantic hypergraphs). Additionally, we include a visualization mode in which annotation subgraphs, or semantic summaries, are used to show relationships outside of the sequential context of the text itself. These subgraphs can be used to quickly find similar structures within the current document or external annotated documents. *TAG* was initially developed to support information extraction tasks on a large database of biomedical articles. However, our software is flexible enough to support a wide range of annotation tasks for many domains. Examples are provided that showcase *TAG*'s capabilities on morphological parsing and event extraction tasks.

Keywords: annotation, event extraction, online NLP tools, text visualization

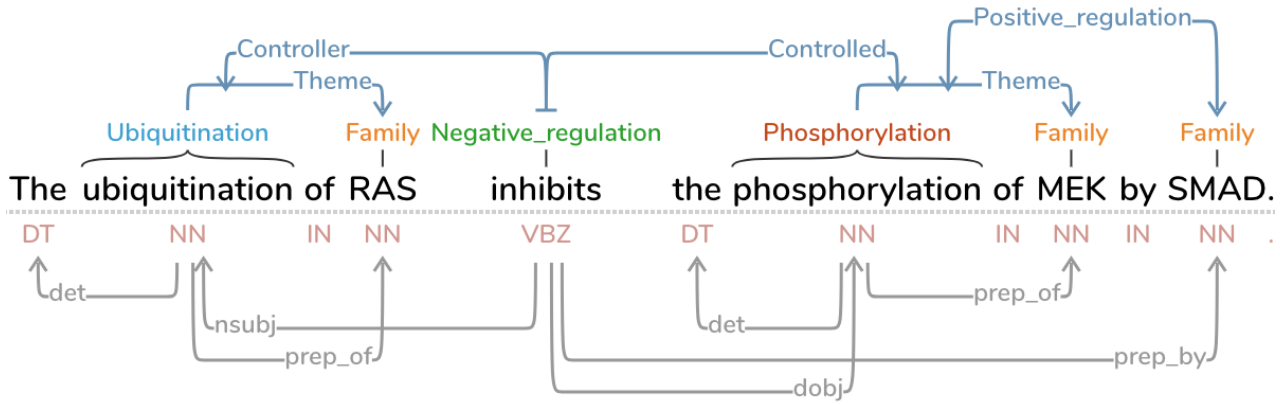


Figure 1: A screen capture of the annotation panel in the *TAG* software. Here a single sentence of biomedical text is shown, with both semantic (top) and syntactic (bottom) relationships toggled on.

1. Introduction

According to systems biologist Arthur Lander, the dominant icon of our current era is the "hairball" (Lander, 2010), a visual depiction of a network of such complexity that the edges overlap so densely as to obscure the meaning of the very relationships it is meant to explain. Part of the reason for this shift from a more straightforward symbol (such as the double helix, which, while still complex, indicates a more orderly, and potentially predictable machinery) to the hairball, Lander writes, is the rise of the computational tools that make it possible to generate vast amounts of "big data," as well as the realization that thorny problems generally involve the intersections between data drawn from multiple, interacting systems.

Contemporary information extraction approaches seek to capture complex natural language phenomena, such as those found in biomedical literature (Björne et al., 2010; Valenzuela-Escárcega et al., 2015). These biomedical articles present new knowledge regarding biological processes which can be usefully described via a semantic graph that represents the event structure of biochemical reactions. Moreover, these approaches can be used to detect predicates (triggers), relations, and events within individual articles in order to find connections to other corpora and to draw inferences about relationships between texts.

We introduce a novel web-based annotation software application, *Text Annotation Graphs*, or *TAG*, that provides a rich set of tools for viewing and editing annotations that capture complex natural language phenomena, such as deeply nested event structures (Figs. 1 and 6), syntactic dependencies (bottom portion of Fig. 1), undirected relations, coreference resolution, and morphological parses (Fig. 5). *TAG* provides easily readable output for complex events and can be used to illustrate non-projective representations and even link predicate-less relations. It supports multiple languages and data formats, allows users to view layers of annotations simultaneously, and includes interactive features for filtering and editing annotations. Moreover, *TAG* implements progressive rendering or text annotations, making it possible to load in large documents without loss in rendering speed.

TAG runs on any modern web browser and is available as open source software from `https://github.com/CreativeCodingLab/TextAnnotationGraphs`.

2. Related Work

TAG is inspired by the *BRAT* rapid annotation tool (Stenetorp et al., 2012). *BRAT* is widely used for representing syntactic structure, but can also represent semantic events, and has been applied to a range of domain-specific NLP tasks, including biomedical data (Verspoor et al., 2013). *BRAT* supports a range of useful features that improve the overall efficiency of manual annotation tasks. However, *BRAT* does not support the ability to draw links between links, which makes it difficult to represent relations linking several predicate-less relations, a feature necessary to completely describe complex events. Complex relations are previously explored by the authors in a range of different visualization projects that represent hierarchically-nested and/or clustered data derived from the machine reading of scientific texts describing biochemical events (Dang et al., 2015; Dang et al., 2016; Dang and Forbes, 2017; Dang et al., 2017; Forbes et al., 2018; Murray et al., 2017; Paduano and Forbes, 2015; Paduano et al., 2016).

In addition to *BRAT*, a range of newer projects investigate visual encodings for specific annotation tasks. For example, Sarnat et al. (2017) introduce a web interface for exploring a parse tree. By entering in a sentence, an interactive visualization is created that includes expand/collapse functionality, positional and color cues, explicit visual support for sequential structure, and dynamic highlighting to convey node-to-text correspondence. The tool includes an unusual representation of sequential structure as a container of linked nodes. While this may help a user to distinguish relevant structural elements, it demands a large portion of the screen, which can make it difficult to view relationships over longer sequences of text. *TAG* also utilizes a range of visual encodings to identify relationships and types, and includes an alternative representation of linguistic relationships. However, in *TAG*, a user can fluidly switch between these representations or view them side by side.

Another representation of a parse tree is used in the *displaCy* software, introduced by Montani (2017). *displaCy*, however, is not equipped for annotation tasks and does not provide any interaction beyond the ability to scroll through a sentence (although more sophisticated functionality could potentially be built on top of it). By default, the entire body of text is placed on a single row, making it unwieldy for representing longer swaths of text.

WebAnno is a flexible tool that supports multiple annotation layers, and includes features to facilitate quality control, annotator management, and curation (Yimam et al., 2014; de Castilho et al., 2016). The visualrepresentation is similar to *BRAT*, and the interface focuses mainly on resolving disagreeing annotations between users. *WebAnno* includes a variety of built-in annotation layers, such as dependency relations, co-reference chains, and lemma forms, but in *WebAnno* annotation expressiveness is limited in that it is not possible to create nested arcs that link to other arcs.

Purgina and Mozgovoy (2017) introduce *WordBricks*, which also utilizes a container layout to identify linguistic structure while maintaining an explicit sequential representation that is easy to read. Although their approach eschews links that may lead to visual clutter, the *WordBricks* layout can be difficult to read once the container is more than even a few levels deep, or in cases where the user wishes to quickly identify relationships between words or phrases across sentences or within longer passages.

3. Text Annotation Graphs (TAG)

In this section, we introduce the main visualization and interaction features of the *TAG* software application. It consists of three panels: the annotation panel, the tree panel, and the options panel. Upon opening the software via any modern browser, *TAG* shows the user a main annotation panel consisting of an example text snippet with both semantic and syntactic annotations. A secondary tree panel shows a "semantic summary" of a selected subgraph, in which text is organized using a network layout that emphasizes event relationships rather than text sequences. This alternative representation provides an overview of the event that can augment the more traditional reading of a parse tree. Additionally, it can be used to highlight the results of an event extraction process, and used as input for a structured search across other annotated documents. A third panel can be displayed on demand that provides the user with a range of options to change the visual appearance of text, to filter out particular annotation types, and to toggle on or off syntactic or semantic annotations.

3.1. Loading Data

TAG is meant to be agnostic to data schema and currently supports the following formats: BRAT standoff (Stenetorp et al., 2012), CoNLL-X (Buchholz and Marsi, 2006), and bio-C (Comeau et al., 2013). Additional formats can be added as desired by following the import templates for the supported formats. The software looks for files in a special 'data' folder, and populates a drop down list to provide the option for the user to load in any of those data files. By choosing an item in this drop down list, which is located in the top left of the page, the data file is loaded into the *TAG* application, and populates the annotation panel. A taxonomy file can also be associated with a data file, which enables *TAG* to color annotations based on their taxonomic type, and makes it possible for users to search and filter by type.

3.2. Annotation Panel

The annotation panel takes up the majority of the *TAG* application, filling up the entire page, or only the top two thirds of the page if the tree panel is opened. The annotation panel presents the text from the data file across many rows, requiring the user to scroll down to see text from longer passages. *TAG* uses a progressive rendering strategy so that it is possible to display even large documents without loss in rendering speed. In each row, an annotation graph is shown both above and below the text. At the bottom, syntactic annotation is shown; At the top, semantic annotation is shown. Either syntactic or semantic annotations can be toggled on or off on demand.

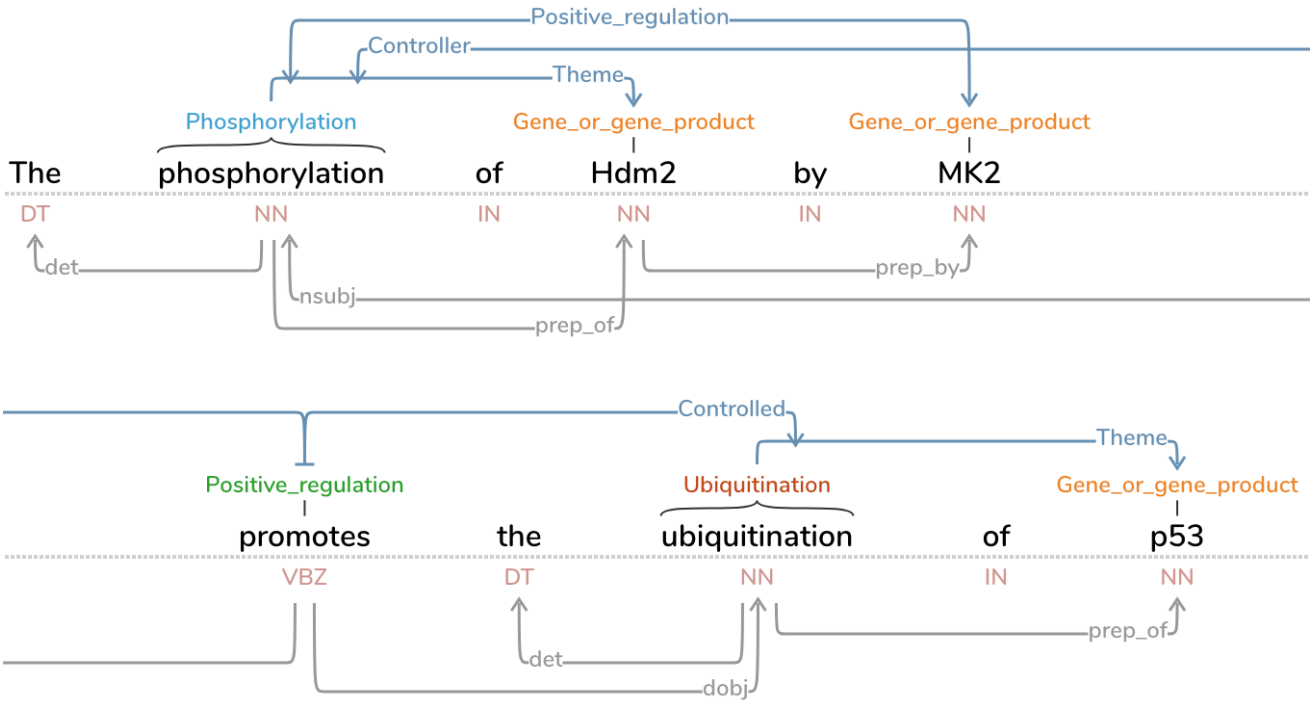


Figure 2: This screen capture shows the layout of a single sentence where annotation data is linked between elements which are presented on different rows. A user can interactively reposition words and links as desired across multiple rows in order to emphasize particular sections of text.

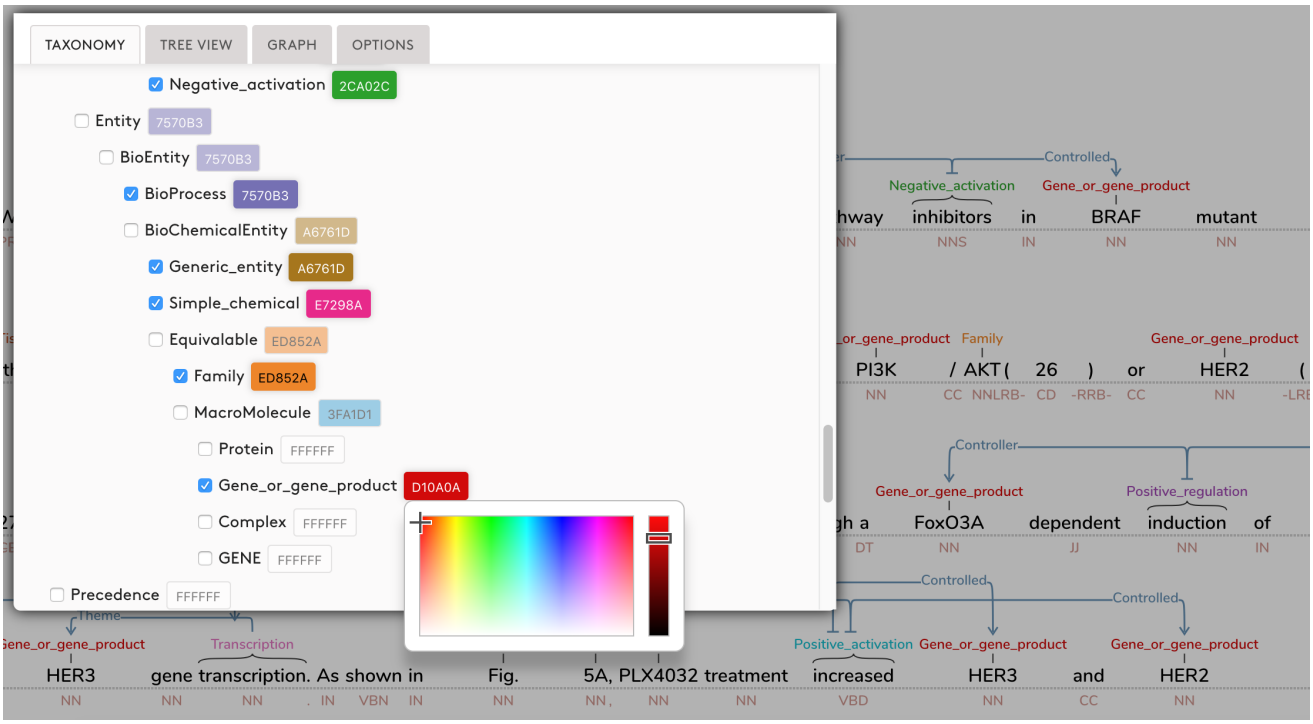


Figure 3: This screen capture shows a user updating the color of all nodes of type 'Gene_or_gene_product' via the interactive color picker. In the background, we can see that all examples of this annotation are updated dynamically. Selecting an annotation type that is a parent of 'Gene_or_gene_product', such as 'MacroMolecule' or 'Entity', would also update the color of 'Gene_or_gene_product', along with all its other children.

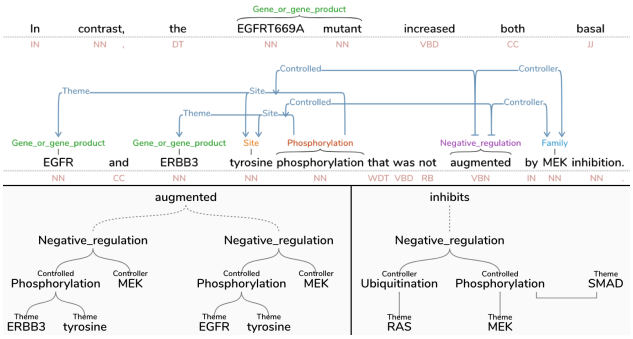


Figure 4: The tree panel shows a semantic summary of a selected annotation. The top and bottom left show the same annotation in both formats, in the the tree panel emphasizes the negative regulation events triggered by the word 'augmented.' The bottom right shows the semantic summary of the sentence shown in Fig. 1, highlighting the events triggered by the word 'inhibits.'

TAG supports trigger-free relation mentions, providing the ability for the annotation graphs to utilize uni- or bi-directional links between nodes, links, or a combination of nodes and links. The layout of the links is organized using a custom algorithm that makes text labels on the graph easy to read, and minimizes edge crossings. Fig. 1 shows an example of data loaded into the annotation panel that requires only one row.

A user can interactively reposition the words in the text, including seamlessly moving words to different rows. This allows the user to emphasize particular word sequences by clustering a series of words closer together. Similarly, the vertices of links can be repositioned to improve the readability of the annotation graph. Fig. 2 show a view of an annotated sentence across two rows. Initially, it was positioned by default along a single row; here we see an example of how a user might choose to arrange the sentence interactively. In this case, annotation data is linked between elements which are on different rows and the syntax tree has been toggled on.

Annotations can be edited on demand. By clicking on a node, the user can select a type from a provided taxonomy, or freely type to replace the node's current annotation. Links can be repositioned so that one or more of the connections can instead connect to different link or node. A node or link can also be hidden or deleted, which also hides or removes any links directly connected to it.

Additionally, a user can change the color of a single node (or all nodes of a particular type) by clicking on a node, which pops up an option panel, which allows the user to select a color dynamically. Figure 3 shows a user updating all nodes of type 'Gene_or_gene_product' to be colored red. When changing the color of a type, a user can also update all of its subtypes as well, if desired.

3.3. Tree Panel

By default the annotation panel fills up the entire screen, but the bottom third of the application can be used to provide an alternative representation of the annotated text. By double-clicking on any word or annotation, the tree panel appears in the application, displaying a tree whose root node is the element that was clicked on. The tree representation makes it easier to see the structure of the relations between annotation events, which may be more difficult to read in the annotation panel, especially when related events are spread across multiple sentences or rows, or when a single sequence of text contains multiple events. Of course, the tree representation removes the context from which the events are extracted. However, by presenting both views at once, the user is able to see the summary semantics of the annotation relations alongside the annotations embedded in the text.

Fig. 4 shows an example of the tree panel. On the top, the annotation panel shows a sentence containing two negative regulation events whose representation has a somewhat dense layout, even after the user has repositioned the event so that all involved words are on the same row. On the bottom left, we see the simplified tree representation of the events, which appears when the user clicks on the word "augments." On the bottom right, we see another tree representation— the sentence described in Fig. 1— that highlights the annotations related to the word "inhibits," and which shows an example of a sibling relationship.

3.4. Option Panel

The option panel provides various ways for the user to customize the software. The options include: the ability to toggle on or off the annotations above or below the text; optimization options, such as the ability to hide the annotation graph when repositioning words; and a mechanism for filtering annotations that contain (or do not contain) a specified taxonomic type. The user can also change the color of types defined in the taxonomy to highlight particular elements (a biomedical taxonomy can be seen in Fig. 3).

4. Examples

Although *TAG* can be used for a wide range of annotation tasks, here we highlight two examples that showcase situations that are difficult to represent using other software.

4.1. Morphological Parse

Fig. 5 shows a pair of examples that demonstrate semantic differences arising from morphological derivations of "unlockable," which, as noted by Larson and Ludlow (1993), could mean either "cannot be locked" (top) or "can be unlocked" (bottom). As we can see, trees such as these cannot easily be visualized in *BRAT*, but can be represented in *TAG*.

4.2. Event Annotation

Fig. 6 shows a comparison of how *BRAT* and *TAG* handle the output of Reach (Valenzuela-Escárcega et al., 2015) when processing the title text of the paper "Induction of p21 by p53 following DNA damage inhibits both Cdk4 and Cdk2" (He et al., 2005). Looking at extracted event E1, we can see that the Reach system finds that the positive activation event on the word "Induction" indicates p53 as the controller for p21. In the *BRAT* representation, the "controller" and "controlled" relations are separated, whereas in *TAG* it is easier to see that "induction" links p21 and p53

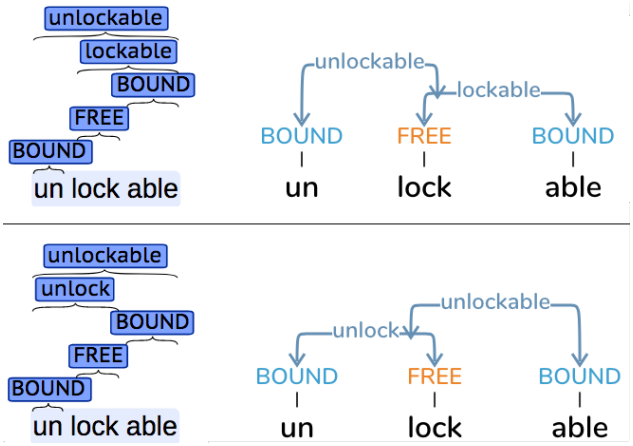


Figure 5: Two interpretations for "unlockable" arising from differences in morphological derivation displayed in both *BRAT* (left) and *TAG* (right). The top parse models the interpretation "cannot be locked", while the bottom shows "can be unlocked".

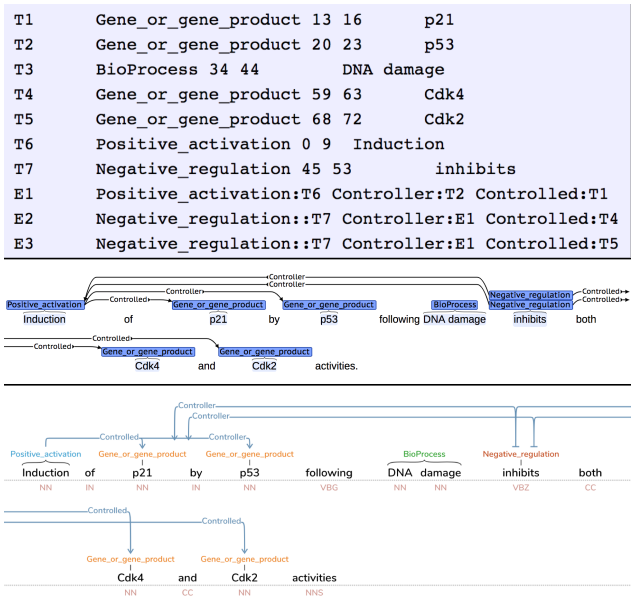


Figure 6: Here we see screen captures of the same Reach events (top) in both *BRAT* (middle) and *TAG* (bottom). In *TAG*, it is clear that E2 and E3 both use E1 as a controller of Cdk4 and Cdk2.

through a single positive activation trigger. Similarly, *TAG* is able to recognize that the a single word serves as a trigger for two negative regulation events, and presents it as a single event with two simultaneous relations. Here, it is the entire positive activation event (E1) that serves as the controller for the inhibition of both Cdk4 and Cdk2.

5. Conclusion

This paper described the main functionality of the *Text Annotation Graph* software, showcasing its increased expressivity for visually representing complex natural language phenomena for even large text documents. The software has been used so far to successfully represent events extracted from biomedical articles, but we expect that it could prove useful for annotation tasks across a range of domains.

Ongoing work is focused on extending *TAG* to support comparative tasks, for instance, to use both the top and bottom half of each row to present the semantic annotations extracted from different systems. We are also investigating the use of *TAG* to provide a human readable visualization of an active learning loop, enabled through saving "diffs" between the loaded data and the edited data. Since *TAG* makes it easy to read and edit annotated events, we are exploring its use as a front end for crowdsourced user studies, such as those administered via Mechanical Turk, in order to gather data that can be used to train classifiers.

TAG is written in JavaScript and runs in any browser. It makes extensive use of the SVG.js library,[1] which produces PDF quality output that is suitable for embedding in both online and printed documents. The software is open source and freely available from our GitHub code repository.[2]

Acknowledgements

This work was funded by the DARPA Big Mechanism program under ARO contract W911NF-14-1-0395 and by the Bill and Melinda Gates Foundation HBGDki Initiative. Gus Hahn-Powell, Marco Valenzuela-Escárcega, and Mihai Surdeanu declare a financial interest in lum.ai. This interest has been properly disclosed to the University of Arizona Institutional Review Committee and is managed in accordance with its conflict of interest policies.

[1] http://svgjs.com/

[2] https://github.com/CreativeCodingLab/TextAnnotationGraphs

Bibliographical References

Björne, J., Ginter, F., Pyysalo, S., Tsujii, J., and Salakoski, T. (2010). Complex event extraction at PubMed scale. *Bioinformatics*, 26(12):i382–i390.

Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, pages 149–164.

Comeau, D. C., Islamaj Doğan, R., Ciccarese, P., Cohen, K. B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M., et al. (2013). BioC: A minimalist approach to interoperability for biomedical text processing. *Database*, 2013:bat064.

Dang, T. and Forbes, A. G. (2017). CactusTree: A tree drawing approach for hierarchical edge bundling. In *Proceedings of the 10th IEEE Pacific Visualization Symposium (PacificVis)*, pages 210–214, Seoul, Korea, April.

Dang, T., Murray, P., Aurisano, J., and Forbes, A. G. (2015). ReactionFlow: An interactive visualization tool for causality analysis in biological pathways. *BMC Proceedings*, 9(6):S6, August.

Dang, T., Pendar, N., and Forbes, A. G. (2016). TimeArcs: Visualizing fluctuations in dynamic networks. *Computer Graphics Forum*, 35(3):61–69, June.

Dang, T., Murray, P., and Forbes, A. G. (2017). BioLinker: Bottom-up exploration of protein interaction networks. In *Proceedings of the 10th IEEE Pacific Visualization*

Symposium (PacificVis), pages 265–269, Seoul, Korea, April.

de Castilho, R. E., Mújdricza-Maydt, E., Yimam, S., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C. (2016). A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the LT4DH workshop at COLING 2016*.

Forbes, A. G., Burks, A., Lee, K., Li, X., Boutillier, P., Krivine, J., and Fontana, W. (2018). Dynamic influence networks for rule-based models. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):184–194, January.

He, G., Siddik, Z. H., Huang, Z., Wang, R., Koomen, J., Kobayashi, R., Khokhar, A. R., and Kuang, J. (2005). Induction of p21 by p53 following DNA damage inhibits both Cdk4 and Cdk2 activities. *Oncogene*, 24(18):2929.

Lander, A. D. (2010). The edges of understanding. *BMC Biology*, 8(1):40.

Larson, R. K. and Ludlow, P. (1993). Interpreted logical forms. *Synthese*, 95(3):305–355.

Montani, I. (2017). displaCy Dependency Visualizer. https://github.com/explosion/displacy, accessed Sept. 24, 2017.

Murray, P., McGee, F., and Forbes, A. G. (2017). A taxonomy of visualization tasks for the analysis of biological pathway data. *BMC Bioinformatics*, 18(2):21–1–13.

Paduano, F. and Forbes, A. G. (2015). Extended LineSets: A visualization technique for the interactive inspection of biological pathways. *BMC Proceedings*, 9(6):S4, August.

Paduano, F., Etemadpour, R., and Forbes, A. G. (2016). BranchingSets: Interactively visualizing categories on node-link diagrams. In *Proceedings of the 9th International Symposium on Visual Information Communication and Interaction (VINCI)*, pages 9–16, Dallas, Texas, September.

Purgina, M. and Mozgovoy, M. (2017). Visualizing sentence parse trees with wordbricks. In *Proceedings of the 3rd IEEE International Conference on Cybernetics (CYBCON)*, pages 1–4.

Sarnat, A., Joshi, V., Petrescu-Prahova, C., Herrasti, A., Stilson, B., and Hopkins, M. (2017). Interactive visualization for linguistic structure. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: A web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107.

Valenzuela-Escárcega, M. A., Hahn-Powell, G., Hicks, T., and Surdeanu, M. (2015). A domain-independent rule-based framework for event extraction. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 127–132.

Verspoor, K., Jimeno Yepes, A., Cavedon, L., McIntosh, T., Herten-Crabb, A., Thomas, Z., and Plazzer, J.-P. (2013). Annotating the biomedical literature for the human variome. *Database*, 2013:bat019.

Yimam, S. M., Biemann, C., de Castilho, R. E., and Gurevych, I. (2014). Automatic annotation suggestions and custom annotation layers in webanno. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 91–96.

Manzanilla: An Image Annotation Tool for TKB Building

Arianne Reimerink, Pilar León-Araúz
University of Granada
Department of Translation and Interpreting, C/ Buensuceso, 11, 18008 Granada (Spain)
arianne@ugr.es, pleon@ugr.es

Abstract

Much has been written regarding the importance of combining visual and textual information to enhance knowledge acquisition (Paivio, 1971, 1986; Mayer & Anderson, 1992). However, the combination of images and text still needs further analysis (Faber, 2012; Prieto, 2008; Prieto & Faber, 2012). An in-depth analysis of the features of images provides the means to develop selection criteria for specific representation purposes. The combination of conceptual content, image type based on morphological characteristics, and functional criteria can be used to enhance the selection and annotation of images that explicitly focus on the conceptual propositions that best define concepts in a knowledge base. Manzanilla is an image annotation tool specifically created for EcoLexicon, a multilingual and multimodal terminological knowledge base (TKB) on the environment. It is powered by Camomile (Poignant et al., 2016) according to the selection and annotation criteria resulting from ten years of research on multimodality within the framework of Frame-Based Terminology (FBT; Faber, León-Araúz & Reimerink, 2014). The tool was created to enhance the consistency of knowledge representation through images with the conceptual knowledge in EcoLexicon and to improve image reusability.

Keywords: image annotation, multimodal knowledge representation, EcoLexicon, Camomile

1. Introduction

Manzanilla is an image annotation tool specifically created for EcoLexicon, a multilingual and multimodal terminological knowledge base (TKB) on the environment[1]. It was developed with Camomile (Collaborative Annotation of multi-MOdal, multI-Lingual and multi-mEdia documents; Poignant et al., 2016)[2] according to the selection and annotation criteria resulting from ten years of research on multimodality within the framework of Frame-Based Terminology (FBT; Faber, León-Araúz & Reimerink, 2014). The tool was created to enhance the consistency of knowledge representation through images with the conceptual knowledge in EcoLexicon and to improve image reusability.

Currently, images are stored in the TKB in association with concept entries according to the semantic content described in their definition, and are thus regarded as a whole and only linked to the concept itself. Other knowledge bases, such as BabelNet, the automatically constructed multilingual encyclopedic dictionary and semantic network (Navigli & Ponzetto, 2012), also uses this approach. However, regarding images as a whole does not allow for a more fine-grained annotation where the semantic relations between different concepts represented in an image are made explicit. Our new approach is that images should not be stored in the TKB as the representation of a concept, but as the representation of a set of conceptual propositions (concept-relation-concept triples) more in line with the conceptual structure of EcoLexicon. Therefore, images must be annotated according to semantic and morphological information and stored in a separate repository. Since each image activates several propositions and each proposition can be activated by different concepts, one image can then be linked to several concept entries. This would enhance the reusability of images, improve the consistency of the TKB and avoid duplicating workload (Reimerink, León-Araúz & Faber, 2016; León-Araúz & Reimerink, 2016).

In Section 2, we explain how images have been selected and included in EcoLexicon up to now. In Section 3, a summary of our research into image selection and annotation criteria is given. Then, in Section 4, the tool is explained in detail. Finally, in Section 5 some conclusions are drawn and future work is addressed.

2. Images in EcoLexicon

The knowledge contained in EcoLexicon is largely based on information extracted from a specialized domain corpus that was compiled for this specific purpose (Faber, León-Araúz & Reimerink, 2014; Faber, León-Araúz & Prieto, 2009). This conceptual knowledge is represented through semantic networks based on conceptual propositions and definitions based on these networks. To further enrich conceptual description, a visual corpus was also compiled. As stated above, each concept entry has several images selected according to the semantic content of a concept's definition.

The definitions in EcoLexicon are based on templates that define category membership and describe the basic conceptual propositions in which the concept participates. In this way, definitions have a uniform structure that directly refers to and evokes the underlying conceptual structure of the domain, represented in the semantic networks.

For example, for the definition of WATER EROSION, the template includes the four basic relations of all natural processes: *is_a*, *has_agent*, *affects* and *has_result*. For the selection of images, the basic conceptual propositions in the definitional template are used to select images which contain the same information to reinforce knowledge acquisition (Faber et al., 2007). Figure 1 shows one of the images included in the conceptual entry of WATER EROSION to represent the relation *has_result*. The template also has an additional relation because it is a complex procedural concept, which can be divided into a sequence of steps: *has_phase*. Figure 2 was included in the entry to represent the conceptual proposition WATER EROSION *has_phase* WEATHERING (León-Araúz, Reimerink &

[1] EcoLexicon is freely accessible at ecolexicon.ugr.es.
[2] https://camomile.limsi.fr

Faber, 2013). Images are thus regarded as a whole and are only linked to the concept itself.

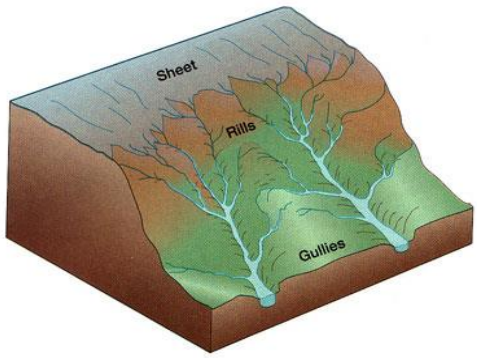


Figure 1: Image for SHEET/RILL/GULLY *result_of* WATER EROSION in EcoLexicon.



Figure 2: Image for WATER EROSION *has_phase* WEATHERING in EcoLexicon.

So far, we have shown how the same concept can be represented through different images, depending on perspective, or the semantic content highlighted (Faber et al., 2007; Reimerink, García de Quesada & Montero-Martínez, 2010). However, one and the same image may also work for the representation of other related concept entries (e.g. an entity and the process through which it was formed, a concept and its parts, etc.). Images should thus be further dissected according to the features they possess (i.e. image type and other morphological characteristics) and the knowledge they convey (i.e. semantic content). For example, many images show several concepts in a specific background where they establish different relations that can be explicitly labelled or inferred from previous knowledge. In this sense, we propose a different approach where images are stored in the TKB not as the representation of a concept, but as the representation of a set of conceptual propositions. Thus, images must be annotated according to semantic and morphological information and stored in a separate repository.

3. Image Selection Criteria

In previous research (Reimerink, Léon-Araúz & Faber 2016, León-Araúz & Reimerink, 2016), we have explained in detail how images convey conceptual knowledge through their morphological features, such as the use of colours, arrows, labels, etc., that we have called visual knowledge patterns (VKPs). In this section, we will summarize our findings related to the interaction between concept type, image type, and VKPs with a few examples. We use two functional criteria, referential similarity and dynamism, to analyse VKPs in images. Referential similarity refers to the degree to which an image resembles its referent in the real world. This similarity is measured on a continuum ranging from non-similar to totally identical. It goes without saying that a two-dimensional image can never be totally identical to its referent, but a colour photograph would have a high degree of referential similarity. Dynamism can also be measured on a continuum ranging from totally static to very dynamic. The results showed which VKPs and which degrees of referential similarity and dynamism are most characteristic of different types of images and how they are related to the conceptual propositions represented in each type.

It has also become clear that VKPs, such as arrows, labels, and colour-coding, are polysemic since the same pattern can be used for different purposes in the same way that textual knowledge patterns can also convey different conceptual relations (León-Araúz, Reimerink & Faber, 2009). Accordingly, the conceptual knowledge underlying VKPs can only be interpreted in the context of each image. Nevertheless, a certain combination of patterns, constrained by image and concept type, makes images more or less suitable for the representation of certain types of conceptual knowledge. An arrow, for example, can be used to connect a term to its representation in the image, thus this VKP does not necessarily transmit dynamism. However, when arrows appear in an image representing a process, they generally convey dynamism and go in the direction of the different phases of the process. The same is true for colours. In images with a high level of referential similarity, the colours in the image are the same or similar to those of the real world entity. In many cases, however, the function of the colours is not to realistically represent the concepts or its natural surroundings, but rather to differentiate closely related concepts in time or space.

For example, a GROYNE is a defence structure perpendicular to the coastline, which retards littoral drift and erosion. It can be made of stone, concrete or wood. The concept GROYNE is an entity and as it can be made of several materials, the proposition GROYNE *made_of* STONE/CONCRETE/WOOD will require more than one image.



Figure 3: Static image for GROYNE *made_of* WOOD[3].

[3] Source : http://blog.seamaidengemsjewellery.co.uk/

Figure 3 is a good example of an adequate image for the proposition GROYNE *made_of* WOOD. It is a static image and a colour photograph which provides a very high level of referential similarity. The same image can also be used for the proposition GROYNE *has_location* COAST. This relation is relevant when the location of a physical object is essential for its description. For instance, a groyne is not a groyne if it is not located on the coast.

Processes are generally described by the meronymic relations *phase_of* and *takes_place_in* because processes are composed of different stages and occur within a certain context. This is in direct contrast to physical objects (such as GROYNE), whose description is dominated by the relations *has_location* and *part_of*. Not surprisingly, processes are generally portrayed by flow charts that represent more than one relation. For example, Figure 4 is an image of the geological cycle, an extremely complex process, which shows both the *take_place_in* and *phase_of* relations. The concepts HARDENING, METAMORPHISM, MELTING, CRYSTALLIZATION, and INTRUSION *take_place_in* under the Earth's surface. At the same time, they are also *phases_of* GEOLOGICAL CYCLE. Figure 4 also conveys the *result_of* relation. This relation is relevant to either events or entities that are derived from other events. In this case, it shows SEDIMENTARY ROCK *result_of* HARDENING, METAMORPHIC ROCK *result_of* METAMORPHISM, etc. The representation of certain natural objects and events (e.g. sun, rain, clouds, magma, volcanic eruption, etc.) has a high degree of similarity to their referents in the real world.

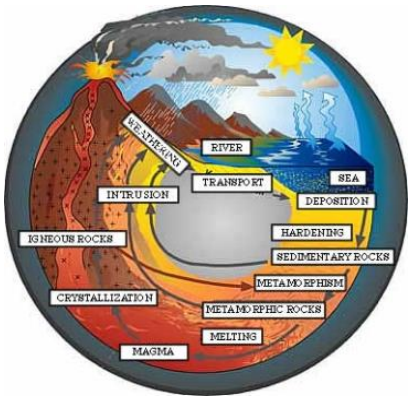


Figure 4: Dynamic image for GEOLOGICAL CYCLE[4].

However, other less well-known objects are labelled to explain where one type of geological formation ends and the other begins. This is the case of SEDIMENTARY ROCKS, METAMORPHIC ROCKS, MAGMA, and IGNEOUS ROCKS. Furthermore, the use of similar yet different colours heightens resemblance and, at the same time, delimits similar concepts or those that occur in connected locations. More specifically, WATER and SKY are different shades of blue, and there is a gradual colour change from yellow to red and dark brown to show how SEDIMENT becomes ROCK and then MAGMA.

Arrows add dynamism to images that portray how certain processes stem from others and how they affect one another. Thus, arrows as visual knowledge patterns (VKPs) most often convey meronymy in the case of entities (*part_of*) and (*phase_of*), and the *result_of* relation in the case of processes.

The findings of our research on the interrelations between concept type, image type, and VKPs have resulted in the following selection guidelines for researchers that work on images for EcoLexicon:

1. Use photographs for the *type_of*, *made_of*, and *has_location* relations of physical entities shown in their real-world environment.
2. Use drawings with labels and arrows for representing complex meronymic relations (*part_of*, *delimited_by*) or to differentiate between closely related concepts that are otherwise hard to differentiate without making reference to one another. Drawings are mostly fit to represent entities, but combinations of several drawings can be used to describe processes and their phases, especially if no flow chart is available.
3. Use flow charts for complex processes and non-hierarchical relations such as *causes* and *result_of*. The flow chart must show a high level of referential similarity for the background. It must use colour-contrast to differentiate between closely related concepts. It must also contain arrows to add dynamism and show the direction of the movement or even add textual explanations.

4. Manzanilla

Image annotation is often defined as the labelling of the semantic content of images with a set of keywords (Wenyin et al., 2001). However, "even though an image is worth a thousand of words, humans still possess the ability to summarize an image's contents using only one or two sentences. Similarly, humans may deem two images as semantically similar, even though the arrangement or even the presence of objects may vary dramatically" (Zitnick and Parikh, 2013).

One of the most famous annotated image sets that are available at present is ImageNet (Deng et al., 2009, Russakovsky et al., 2015), which is the largest annotated image set available and mostly consists of photographs that are annotated according to the hierarchical structure of WordNet synsets. This is done by automatically retrieving images from the internet through searches with WordNet synonyms, which are then verified for accuracy by humans through Amazon Mechanical Turk. BabelNet, the automatically constructed multilingual encyclopedic dictionary and semantic network (Navigli & Ponzetto, 2012) includes around 11 million images that are automatically retrieved from ImageNet and Wikipedia. The new EcoLexicon image repository we envision is different from these examples as ImageNet mostly includes photographs, whereas our repository considers other image types, such as drawings and flow charts which can represent more complicated specialized knowledge. Furthermore, the fine-grained annotation based on all the conceptual propositions contained in an image goes far beyond the synset annotation of ImageNet. In BabelNet, images are included in entries as a whole without further specifying the conceptual relationships contained in the images. This is also the case for

[4] Source: http://finstone.fi/engl/geology/

EcoLexicon at present, but that is exactly what we want to change with the annotation proposal in this paper. Apart from the above, ImageNet nor BabelNet provide much domain specific knowledge.

Manual image annotation, apart from inconsistent, can be very time-consuming. For this reason, in computer science automatic image annotation has been studied for some time now (Jeon, Lavrenko and Manmatha, 2003; Li and Wang, 2008). However, these studies mostly focus on photographs (Zitnick and Parikh, 2013 being one of the exceptions), objects and rather general concepts. Furthermore, they do not take into account the interaction of the semantic elements. In this sense, Mei et al. (2008) acknowledge that approaches to automatic image annotation do not usually guarantee good semantic coherence of the annotated words for each image, because they treat each word independently without considering the inherent semantic coherence among the words.

The neural network community has also addressed the problem of image classification focussing on automatic object recognition (Russakovsky et al. 2015) and even addressing fine-grained classification issues, such as recognizing subordinate-level categories (Xiao et al., 2015). The ImageNet Large Scale Visual Recognition Challenge (Russakovsky et al. 2015) has been run annually since 2010 and has achieved ground-breaking results in the area. Nevertheless, the neural network community mostly concentrates on object recognition and maybe the categorization of subtypes. To our best knowledge, the existing relation between different entities or processes in one and the same image are not taken into account in this field of research. Much of the effort goes into identifying the primary object in the foreground, while discarding the information contained in the background. For our purposes, the relation between entities and/or processes and the context or background (in which they are located or take place) are essential for representing the complex multidimensional knowledge of the environmental domain.

Unfortunately, given the specificity of the graphical information that an environmental TKB requires, where we intend to annotate the semantic relations between all concepts represented in each image, and the specialization of the field, at this point in time such procedures cannot be applied in our case.

Taking into account previous research results and the image selection guidelines, we developed Manzanilla. The tool was developed within the framework of Camomile with a step by step interface to facilitate annotation and ensure consistency.

4.1 Camomile

Image annotation forces us to clearly think about naming and categorization issues (Barriuso and Torralba, 2012), which are tasks that are not as straight-forward as they may seem. To facilitate the annotation process and enhance inter-annotator consistency, we needed a system with a flexible but strictly organized interface to label the different types of information related to each image in EcoLexicon. We opted for Camomile for its design because it is open source and flexible, as the user interfaces are specifically created for each use case (Poignant et al., 2016). Its collaborative annotation framework follows a client/server architecture, which facilitates the work of multiple users on consistent data sources. In the Camomile framework, resources are annotations, which are represented in JSON formats, stored in a MongoDB database. Based on the use case, four types of collections are developed: corpus, media, layers, and annotations. The corpus collection describes all available corpora. Each corpus contains a set of media and a set of layers. A medium corresponds to a multimedia resource (e.g., a video or audio file). A layer is composed of multiple annotations with the same type (e.g. one layer for manual annotations of speech turns or one layer for annotations of face tracks). An annotation is uniquely defined by a media fragment (e.g., a temporal segment) and attached data (e.g. the name of the current speaker) (idem: 1422).

For the design of the tool, we expressed our needs in the following guidelines for annotation of EcoLexicon images:
1. Annotate image type: photograph, drawing (including maps or diagrams), or flow chart.
2. Annotate all the concepts in EcoLexicon which are present in the image.
3. The TKB will provide a list with all the possible relations between pairs of the selected concepts. Annotate the most representative propositions for the image.
4. Annotate VKPs: labels, arrows for parts or dynamism, colour coding/contrast, etc, and their function.

4.2 User interface

EcoLexicon data have been translated into the following Camomile collections:
1. Images are modelled as Camomile media.
2. EcoLexicon data is stored in Camomile metadata.
3. Image annotations are Camomile annotations grouped into Camomile layers.
These layers are presented to annotators in several subsequent interfaces according to our annotation guidelines to enhance consistency.



Figure 5: Home page of Manzanilla.

In Figure 5, the home page of Manzanilla is shown. After the annotator logs in, an interface appears where a search concept is entered, in this case GROYNE, and all the images related to that concept in EcoLexicon are shown (see Figure 6). The search concept is one of a list of concept entries with images already available in EcoLexicon.

The annotator then choses an image (for example Figure 7), which leads to the next interface where the image type can be chosen: photograph, drawing or flow chart (Figure 8). The use of arrows to show the direction of movement and the inclusion of procedural concepts such as LONGSHORE TRANSPORT, UPDRIFT and DOWNDRIFT, clearly show that this image is a flow chart.

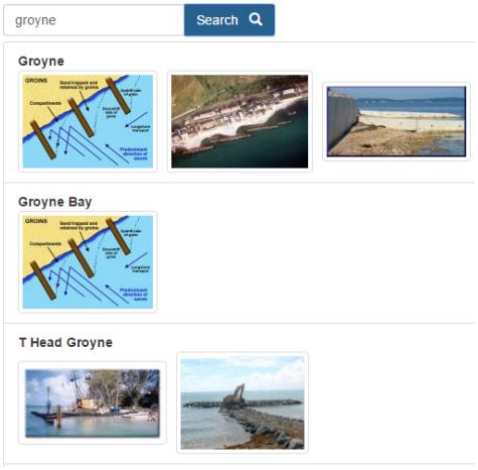


Figure 6: Concept search and image selection interface.

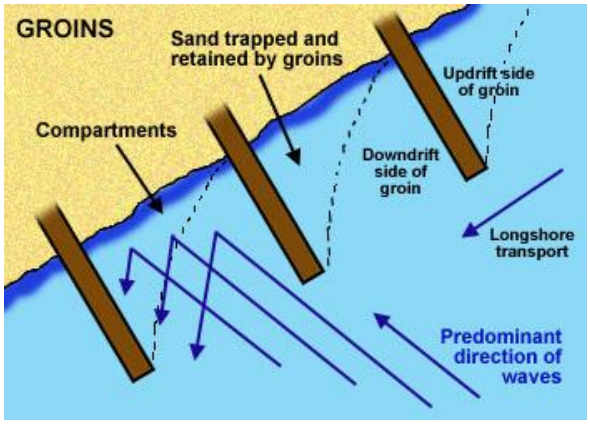


Figure 7: Image selected from EcoLexicon in search concept GROYNE.

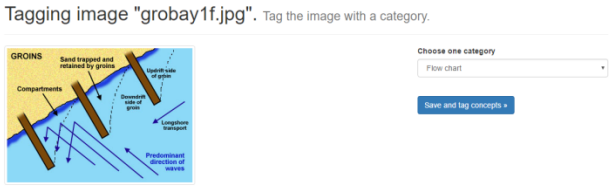


Figure 8: Image type annotation interface.

In the next interface (Figure 9), the other concepts contained in the image, apart from the initial search concept, are annotated. A list of suggestions is offered based on all the concepts that are related to the concept entry where the image is stored in EcoLexicon. New concepts can all so be tagged. These are recorded to make sure that the concept is added to EcoLexicon later on.

After tagging the concepts, in the next interface, the relations between those concepts are annotated. Suggestions are again provided based on the conceptual propositions contained in EcoLexicon. Of course, new propositions can be added as well (Figure 10).

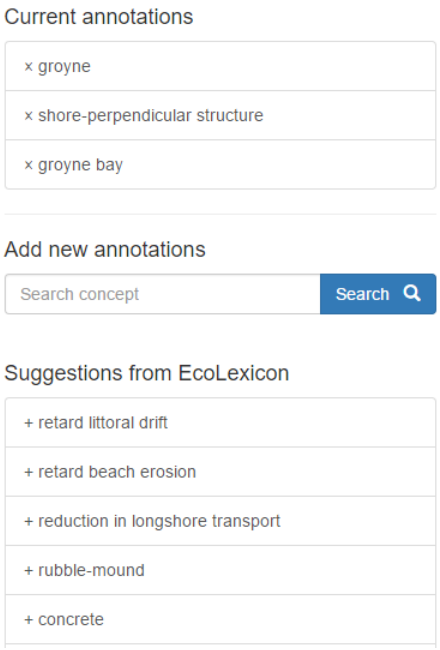


Figure 9: Extract of concept annotation interface.

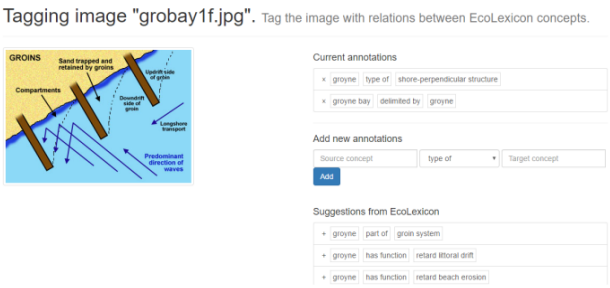


Figure 10: Conceptual relation annotation interface.

The last interface is where the VKPs used in the image are tagged (Figure 11). With the mouse, sections of the image can be marked and labelled according to type (arrow, label, colour-coding) and the function the VKP expresses in the image.



Figure 11: Interface to annotate VKPs.

At the moment, the function of each VKP is a free-text box. Thus, each annotator can freely describe the function the VKP represents in this image. However, as this will probably cause a high degree of inconsistency between annotators, a list with fixed options has been defined, which will be implemented shortly (see Table 1).

VKP	Function	
Arrow	Dynamism	Spatial
		Temporal
	Denomination	
	Delimitation	
Colour	Realism	
	Contrast	
Label	Denomination	
	Explanation	
Number	Denomination	
	Temporality	
Logical operator		
Other		

Table 1: Closed list of VKP functions.

The option "Other" has been included in case new VKPs are identified during the annotation process. In the example image of GROYNE (Figure 7), arrows are used for both dynamism, to represent the direction and movement of processes, and denomination, to show were certain entities are located (COMPARTMENTS and retained SAND). Colours are used to convey realism: the sand is beige, groynes are brown and the water is blue. Furthermore, labels are used for denomination to show where entities are located and processes take place.

5. Conclusions and future work

Manzanilla is an image annotation tool created specifically for building the visual repository of EcoLexicon. The tool will be used to annotate all existent EcoLexicon images, which will provide further insights into image description and multimodal knowledge representation.

Shortly, the tool will be evaluated to see if it provides enough annotation consistency for our purposes. A separate section will be created in the tool where 60 images will be annotated by three annotators. These annotators are all members of the research group LexiCon and are familiar with the existing literature on image description. They will be instructed on the functioning of Manzanilla during an introductory session. Then they will annotate the same images, which are selected by the authors to include a varied range of image types, concept types and uses of VKPs. Apart from inter-annotator agreement, a second more qualitative evaluation will be carried out to assess whether the image selection and annotation criteria developed are sufficient for image population of EcoLexicon or if they need more refinement.

After the evaluation and implementation of possible improvements resulting from the evaluation, Manzanilla will be made available to the public to encourage its use in other fields of knowledge.

When all existing EcoLexicon images have been annotated, the separate image repository will be developed and linked to our TKB. Then Manzanilla will be adapted to include new images. Another access route, apart from EcoLexicon concept entry, will be added to allow for direct access to newly selected images.

Although we have discarded automatic annotation for now, the Camomile framework provides active learning applications that bootstrap manual annotations and retrain or adapt the annotation system accordingly (Poignant et al. 2016, 1421). Therefore, Camomile will be able to use the dataset resulting from manual image annotation to train the system to provide semi-automatic annotation in the future.

6. Acknowledgements

This research was carried out as part of projects FF2014-52740-P, Cognitive and Neurological Bases for Terminology-enhanced Translation (CONTENT), and FFI2017-89127-P, Translation-oriented Terminology Tools for Environmental Texts (TOTEM), funded by the Spanish Ministry of Economy and Competitiveness.

7. Bibliographical References

Barriuso, A. and Torralba, A. (2012) Notes on image annotation. arXiv:1210.3448 [cs.CV].

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L. (2009) ImageNet: A Large-Scale Hierarchical Image Database. *IEEE Computer Vision and Pattern Recognition (CVPR)*.

Faber, P., editor, (2012). A Cognitive Linguistics View of Terminology and Specialized Language. Berlin/Boston: De Gruyter Mouton.

Faber, P., León Araúz, P., Prieto Velasco, J. A., and Reimerink, A. (2007). Linking images and words: the description of specialized concepts. *International Journal of Lexicography*, 20(1):39–65, doi:10.1093/ijl/ecl038.

Faber, P., León Araúz, P. and Prieto Velasco, J.A. (2009) Semantic relations, dynamicity, and terminological knowledge bases. *Current Issues in Language Studies*, 1(1):1–23.

Faber, P., León-Araúz, P. and Reimerink, A. (2014). Representing environmental knowledge in EcoLexicon. In E. Bárcena, T. Read and J. Arhus, editors, Languages for Specific Purposes in the Digital Era. Educational Linguistics 19. Berlin, Heidelberg: Springer.

Jeon, J., Lavrenko, V. and Manmatha, R. (2003). Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 119–126, Toronto, Canada.

León-Araúz, P., Reimerink, A., Faber, P. (2013). Multidimensional and multimodal information in EcoLexicon. In A. Przepiórkowski et al., editors, Computational Linguistics (143–161). Berlin/Heidelberg: Springer.

León-Araúz, P. and Reimerink, A. (2016). Evaluation of EcoLexicon images. In F. Khan et al., editors, Joint

Second Workshop on Language and Ontology & Terminology and Knowledge Structures (LangOnto2 + TermiKS) in conjunction with the 10th edition of the Language Resources and Evaluation Conference, pages 16–22, Portorož, Slovenia.

León Araúz, P., Reimerink, A., and Faber, P. (2009). Knowledge extraction on multidimensional concepts: corpus pattern analysis (CPA) and concordances. In 8ème Conférence Internationale Terminologie et Intelligence Artificielle. Toulouse.

Mayer, R. E., Anderson, R. B. (1992). The instructive animation: helping students build connections between words and pictures in multimedia learning. *Journal of Educational Psychology*, 84(4):715–726.

Mei, T. et al. (2008). Coherent image annotation by learning semantic distance. In IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, Piscataway, NJ, USA.

Navigli, R. & Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193, pp. 217–250.

Paivio, A. (1971). Imagery and Verbal Processes. New York: Holt, Rinehart & Winston.

Paivio, A. (1986). Mental Representations: A Dual-Coding Approach. New York: Oxford University Press.

Poignant, J., Budnik, M., Bredin, H., Barras, C., Stefas, M., Bruneau, P., Adda. G., Besacier, L., Ekenel, H., Francopoulo, G., Hernando, J., Mariani. J., Morros, R., Quénot, G., Rosset, S. and Tamisier, T. (2016). The CAMOMILE Collaborative Annotation Platform for Multi-modal, Multi-lingual and Multi-media Documents. In N. Calzolari et al., editors, Proceedings of the 10th edition of the Language Resources and Evaluation Conference (LREC'16), pages 1421–1425, Portorož, Slovenia, may. European Language Resource Association (ELRA).

Prieto Velasco, J. A. (2008). Información gráfica y grados de especialidad en el discurso científico-técnico: un estudio de corpus. PhD Thesis, University of Granada.

Prieto Velasco, J. A. and Faber, P. (2012). Graphical Information. In P. Faber, editor, A Cognitive Linguistics View of Terminology and Specialized Language, pages 225–248. Berlin/Boston: De Gruyter Mouton.

Reimerink, A., León-Araúz, P. and Faber, P. (2016). Image selection and annotation for an environmental knowledge base. *Language Resources and Evaluation*, 50:1–32.

Reimerink, A., García de Quesada, M. and Montero-Martínez, S. (2010). Contextual information in terminological knowledge bases: A multimodal approach. *Journal of Pragmatics*, 42(7):1928–1950, doi:10.1016/j.pragma.2009.12.008.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015) ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*.

Wenyin, L., Dumais, S., Sun, Y., Zhang, H. J., Czerwinski, M. and Field, B. (2001). Semi-automatic image annotation. In Proc. of Interact 2001: Conference on Human-Computer Interaction, pages 326–333.

Xiao, T., Xu, Y.,Yang, K., Zhang, J., Peng, Y. and Zhang, Z. (2015). The application of two-level attention models in deep convolutional neural network for fine-grained image classification. arXiv:1411.6447 [cs.CV].

Zitnick, C. L. and Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3009–3016.

8. Language Resource References

LexiCon Research Group. (2017). EcoLexicon. Available at http/:/ecolexicon.ugr.es.

LexiCon Research Group. (2017). Manzanilla. Available at http://manila.ugr.es/apis/manzanilla/index.html.

Tools for The Production of Analogical Grids and a Resource of N-gram Analogical Grids in 11 Languages

Rashel Fam, Yves Lepage
Graduate School of Information, Production, and Systems
Waseda University
2-7 Hibikino, Wakamatsu-ku, Kitakyushu-shi, 808-0135 Fukuoka-ken, Japan
{fam.rashel@fuji., yves.lepage@}waseda.jp

Abstract

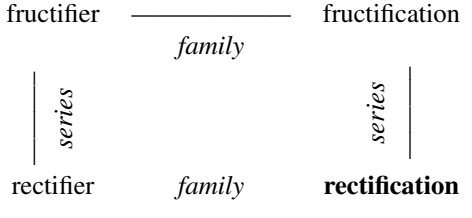
We release a Python module containing several tools to build analogical grids from words contained in a corpus. The module implements several previously presented algorithms. The tools are language-independent. This permits their use with any language and any writing system. We hope that the tools will ease research in morphology by allowing researchers to automatically obtain structured representations of the vocabulary contained in corpora or linguistic data. We also release analogical grids built on the vocabularies contained in 1,000 corresponding lines of the 11 different language versions of the Europarl corpus v.3. The grids were built on N-grams of different lengths, from words to 6-grams. We hope that the use of structured parallel data will foster research in comparative linguistics.

Keywords: Analogy, morphology, analogical grids

1. Introduction

Paradigm tables are known for their usefulness in learning conjugation or declension when studying a language. Such paradigm tables are the result of grammatical tradition or thorough linguistic formalization. They are commonly found in dictionaries and constructed from lexemes and exponents, like the one shown in Figure 2 (*left*) which is taken from a French & English dictionary (Mansion, 1981, grey section, p. 1).

Analogical grids are not paradigm tables, but they also give a compact view on the organization of a lexicon up to a certain extent (see below, Section 2.1.). Analogical grids are the result of an empirical procedure. They may be seen as a preliminary step towards the production of paradigm tables. Figure 2 (*right*) shows an example of an analogical grid in English. They can be used to study the productivity of a language (Singh and Ford, 2000; Neuvel and Fulop, 2002; Hathout, 2008). Fam and Lepage (2016) performed such an analysis across 12 languages using analogical grids built from the Bible corpus (Christodouloupoulos, 2015). As another example of use, Hathout (2009) showed how to produce the French word form *rectification* by analogy from the neighboring word forms *fructifier*, *fructification*, and *rectifier* in the same series (see Figure 1).



fructifier : *fructification* :: *rectifier* : *rectification*

Figure 1: Producing a new word form from neighboring word forms. Example taken from (Hathout, 2009)

This paper introduces the release of a Python module which implements previously presented algorithms that automatically build analogical grids. We also release analogical clusters and analogical grids produced from a parallel corpus on 11 European languages using this Python module.

2. Main Usages of the Tools Released

We release an implementation of previously presented algorithms to produce analogical grids as a Python 2 module[1] called *Nlg*. The various algorithms have been presented elsewhere (Lepage, 1998; Lepage, 2014; Fam and Lepage, 2016). One particular program called *Words2Grids* in this module simply takes a list of word forms as input and delivers a list of analogical grids. Each word form in the list is converted into a feature vector before analogical grids are constructed from such feature vectors. The module also provides another program, *Words2Vectors*, to produce feature vector representations either directly from word forms or from descriptions of word forms. The following sections introduce several ways to use the Python module.

2.1. From Word Forms to Analogical Grids

The default usage of the module is to produce analogical grids from a list of word forms. An analogical grid is a matrix of words where four words from two rows and two columns are a proportional analogy. Formula (1) gives the definition of an analogical grid.

$$\begin{array}{c} P_1^1 : P_1^2 : \cdots : P_1^m \\ P_2^1 : P_2^2 : \cdots : P_2^m \\ \vdots \quad \vdots \qquad\quad \vdots \\ P_n^1 : P_n^2 : \cdots : P_n^m \end{array} \quad \stackrel{\Delta}{\Longleftrightarrow} \quad \begin{array}{c} \forall (i,k) \in \{1,\ldots,n\}^2, \\ \forall (j,l) \in \{1,\ldots,m\}^2, \\ P_i^j : P_i^l :: P_k^j : P_k^l \end{array} \tag{1}$$

Analogy is defined from feature vectors representing word forms, through equality of ratios. A ratio is the difference between two feature vectors plus the edit distance between

[1] `lepage-lab.ips.waseda.ac.jp/nlg-module`

	Infinitive	*Preterit*	*Past participle*	*Present participle*
Regular verb	walk	walked	walked	walking
	smoke	smoked	smoked	smoking
Irregular verb	write	wrote	written	writing
	think	thought	thought	thinking

walk : walks : walking : walked
show : shows : showing : showed
open : opens : opening :
study : : studying :
play : : playing : played

Figure 2: A paradigm table (*left*) taken from a French & English dictionary (Mansion, 1981) and an analogical grid (*right*) obtained by our tools (and reduced to a few rows for lack of space)

the word forms. We refer the reader to (Fam and Lepage, 2016) for exact definitions.

In this setting, a word form is represented as a vector of features which are simply the number of occurrences for all the characters in the alphabet. For instance, in lowercase English, the dimension of the vector is 26 (from *a* to *z*) as illustrated in Formula (2). Here, the notation $|A|_c$ stands for the number of occurrences of character c in string A.

$$A = \begin{pmatrix} |A|_a \\ |A|_b \\ \vdots \\ |A|_z \end{pmatrix} \qquad \text{walking} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{2}$$

The right part of Figure 2 shows few lines of an analogical grid that has been obtained on a list of English word forms with the program *Words2Grids*.

2.2. From Morphological Features to Paradigm tables

The previous use of the released tools automatically converts word forms into specific feature vectors. In opposition to that, it is possible for the user to produce real paradigm tables from feature vectors standing for actual morphological features, like lemma, part-of-speech, case, tense. Such feature vectors can be built, for instance, from the Unimorph Project (Kirov et al., 2016) data which have been built from parsing Wiktionary data into a language-independent feature schema (Sylak-Glassman et al., 2015b; Sylak-Glassman et al., 2015a). Formula (3) illustrates the representation of the word form *walking*: its lemma is *to walk* and it has the verb (VB), present (PRST), participle (PTCP) tags as morphological features. For the purpose of inner processing, the labels are converted into Boolean values.

$$A = \begin{pmatrix} lemma = \text{"to walk"}(A) \\ is_VB(A) \\ is_NN(A) \\ is_PRST(A) \\ \vdots \\ is_PTCP(A) \end{pmatrix} \qquad \text{walking} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \tag{3}$$

Figure 3 shows an example of a paradigm table built from a list of word forms described by morphological features. For some lemmas, some cells may be empty. These lemmas are not necessarily defective; this simply means that the forms did not appear in the input data. It should be stressed that the names of the morphological features are not shown in the paradigm tables output by our programs.

initiate : initiated : initiates
undercry : : undercries
mummify : mummified :
tole : tollen :

Figure 3: A paradigm table built from a list of word forms annotated with morphological features in English

2.3. From User-Defined Features to New Types of Grids

It is possible to directly use user-defined features, i.e., richer vector representations of word forms as input to the released programs. As an example, the feature vector shown in Formula (4) concatenates the two types of vectors presented in the previous sections.

$$A = \begin{pmatrix} lemma = \text{"to walk"}(A) \\ |A|_a \\ \vdots \\ |A|_z \\ is_VB(A) \\ is_NN(A) \\ is_PRST(A) \\ \vdots \\ is_PTCP(A) \end{pmatrix} \qquad \text{walking} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \tag{4}$$

From such feature vectors, the programs output regular paradigm tables (see Figure 4 as an example). We hope that researchers will freely define their own types of feature vector representations to build analogical grids that correspond to their own needs.

initiate : initiated : initiates
stage : staged :
elucidate : elucidated :
assume : assumed :

Figure 4: A regular paradigm table built from combining two previous feature vectors

2.4. An Example of How to Use the Tools

In this section, we show an example of how to use the tools to produce analogical grids from a text. The following illustration is taken from (Fam and Lepage, 2017).

Consider that we have a text as shown in the top of Figure 6. It is a forged example in Indonesian, a language known for its richness of derivational morphology. Using a tokenizer, we can get a list of tokens and then obtain a set of words

RAW output: *anto.grid.txt*

```
 minum :  meminum :  diminum :  minuman ::  makan :  memakan :  dimakan :  makanan ::
main :  None :  None :  mainan ::  beli :  None :  dibeli :  None
```

Pretty_print output: *anto.prettygrid.txt*

```
# Grid no.:  1 - Attributes(length=4, width=4, size=16, filled=12, saturation=0.75)
minum :meminum :diminum :minuman
makan :memakan :dimakan :makanan
main  :        :        :mainan
beli  :        :dibeli  :
```

Figure 5: Analogical grids in Indonesian extracted from a set of word in the bottom of Figure 6. It is shown as RAW output (*top*) and printed using **pretty_print** option (*bottom*).

> *Anto memakan nasi dan meminum air. Nasi itu dibeli di pasar. Di pasar, Anto melihat mainan. Anto senang main bola. Setelah main, Anto suka minum es dan makan cilok. Makanan dan minuman itu juga dia beli di pasar. Es dan cilok memang enak dimakan dan diminum selesai olahraga.*

air anto ***beli*** *bola cilok dan di dia* ***dibeli dimakan diminum*** *enak es itu juga* ***main mainan makan makanan*** *melihat* ***memakan*** *memang* ***meminum minum minuman*** *nasi olahraga pasar selesai senang setelah suka*

Figure 6: A text in Indonesian (*top*) and the list of words extracted from it (*bottom*). Words appearing in Figure 5 are boldfaced.

(types) from it as shown in the bottom part of Figure 6. Let us say that we prepare a file named *anto.words.txt* which contains all of the types, one type per line. We can then extract the analogical grids by running the following command.

```
 $ python Words2Grids.py
< anto.words.txt
> anto.grid.txt
```

The output of the command is printed into the file named *anto.grid.txt*. The content of the file is shown at the top of Figure 5. For the sake of development of the module (input-output between modules), we choose to encapsulate the analogical grids in such data format defined in Section 3.1. Caution: *None* stands for an empty cell.

2.4.1. Visualizing the Analogical Grids

To have a better view on the analogical grids produced by the tools, we provide an option called **pretty_print**. Running the following command will print the analogical grids in a separate file named *anto.prettygrid.txt* with a different format than the one mentioned in Section 3.1.

```
 $ python Words2Grids.py
< anto.words.id.txt
--pretty-print anto.prettygrid.txt
```

The attributes of the grids are printed on top of each grid.

- **length**: number of rows
- **width**: number of columns
- **size**: size of the analogical grid
- **filled**: number of non-empty cells
- **saturation**: saturation of the analogical grid

The bottom of Figure 5 shows the analogical grid extracted from the set of words in Figure 6 printed using the **pretty_print** option. It has a size of 16 cells with 4 rows and 4 columns. 12 cells out of 16 cells are filled so that the saturation of the grid is 0.75. Further details about the attributes of analogical grids, like size and saturation, will be described in Section 3.3. and Section 3.4.

2.4.2. Extracting Analogical Grids Around Particular Word Forms

We also offer a function to focus the study on one particular word form: it is possible to deliver only those analogical grids which contain the particular word form under scrutiny.

The *Words2Grids* program runs in fact in two steps. It first extracts analogical clusters from a list of word forms and then builds analogical grids from the extracted clusters. First, we extract all analogical clusters using the *Words2Clusters* program with a specific option called *focus*. This option will only extract analogical clusters which contain a particular word given as parameter of the option. We then use the *Clusters2Grids* program to produce the analogical grids from the previously extracted clusters.

However, all of these procedures are performed seamlessly to the users by the *Words2Grids* program. For example, to produce all analogical grids which contain the word *walking*, we can run the following command.

```
 $ python Words2Grids.py
< anto.words.id.txt
--focus 'walking'
```

By building analogical grids from those clusters, the user obtains all the analogical grids which contain that word form. It is then possible to characterize the actual productivity of a particular word form by inspecting the size of the analogical grids that contain it. For further study, as proposed in (Hathout, 2009), one can then retrieve all the word forms that have the same relationship with the word form under scrutiny. These words are basically the neighboring word forms inside the analogical grids.

3. Languages and Data in the Released Resource

By using the procedure described in Section 2.1., we produced analogical grids from all the words contained in 1,000 corresponding lines of the Europarl corpus v3 (Koehn, 2007) in all of the 11 European languages. The motivation for using a multilingual parallel corpus is to base on the analogical grids produced to perform comparative studies across these languages. We produced the analogical grids for different sizes of N-grams, from unigrams to six-grams.

Language	# tokens (N)	# types (V)	Avg length of types
da	27,034	5,304	9.06±4.22
de	27,042	5,753	9.69±4.19
el	28,559	6,397	16.45±6.21
en	28,594	4,305	7.35±2.77
es	29,974	5,300	8.18±2.91
fi	20,604	7,473	10.60±4.22
fr	31,257	5,184	8.25±3.01
it	28,269	5,425	8.08±2.84
nl	28,933	5,028	8.90±3.99
pt	29,342	5,472	8.32±3.04
sv	25,681	5,452	9.03±4.11

Table 1: Statistics on the first 1,000 lines of Europarl corpus

Table 1 shows the statistics of the input data. French has the largest number of tokens with more than thirty thousand tokens. Although Finnish has the smallest number of tokens with around twenty thousand tokens, it has the largest number of types followed by Greek. It is due to the characteristic of Finnish being an agglutinative language. This is also reflected by the average length of types. Finnish has the second longest average length (around 11 characters per type in average) of types after Greek (around 17 characters per type in average), contrary to the order of largest number of types. The other languages tend to have around twenty eight thousand tokens represented by around five thousand types with average length of 9 characters.

3.1. Data Format

We release a complete dataset[2] of analogical clusters and analogical grids produced on the data described in the previous section. Each file contains a list of analogical clusters or analogical grids.

As for the analogical clusters, each line in a file is one analogical cluster, i.e., a list of ratios separated by two colons. A ratio consists in two strings separated by a colon. The format of is thus as follows.

$$\mathtt{A}_1 : \mathtt{B}_1 :: \mathtt{A}_2 : \mathtt{B}_2 :: \mathtt{A}_3 : \mathtt{B}_3 :: \ \ldots \ :: \mathtt{A}_N : \mathtt{B}_N$$

As for analogical grids, each line in a file is one analogical grid, i.e., a list of rows separated by two colons. A row is a fixed list of strings separated by colons. The format of is as follows.

$$\mathtt{W}_1^1 : \mathtt{W}_1^2 : \ \ldots \ \mathtt{W}_1^m :: \mathtt{W}_2^1 : \mathtt{W}_2^2 : \ \ldots \ : \mathtt{W}_2^m :: \ \ldots \ \mathtt{W}_n^m$$

Indeed, an analogical cluster is just an analogical grid with 2 columns and no empty cell.

3.2. Number of Analogical Grids Produced

Figure 7 (*top*) shows the number of analogical grids produced for each language with different sizes of N-grams (caution: scale in ten thousands of grids). The number of analogical grids produced rises from unigrams to bigrams, and decreases afterwards in all languages, except Finnish and Greek.

For Finnish, an explanation may be that Finnish, being an agglutinative language, a single word form in that language corresponds in fact to several words in the other European languages, i.e., unigrams in Finnish correspond to bigrams or trigrams in the other European languages. Finnish also exhibits the highest number of analogical grids output for unigrams. This reflects the morphological richness of this language in comparison to the other languages.

For Greek, the number of analogical grids is stable at around 1,000 analogical grids for all sizes if N-grams. The number of analogical grids seems to be stable across N-grams.

The average size of analogical grids exhibit high fluctuations from unigrams to fourgrams. A drop is observed afterwards. The difference is up to five times for languages like English, Spanish, French, and Italian. However, we do not see that for Greek.

3.3. Average Size of Analogical Grids

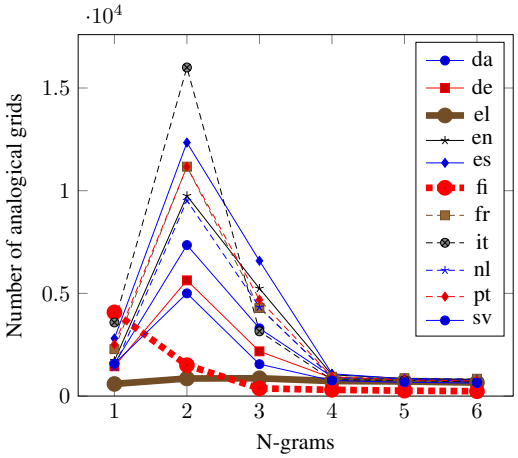
The size of an analogical grid is defined as its number of rows multiplied by its number of columns. (See Formula (5)). E.g., the analogical grid in Figure 2 has a size of $5 \times 4 = 20$.

$$\text{Size} = \text{Number of rows} \times \text{Number of columns} \quad (5)$$

[2] lepage-lab.ips.waseda.ac.jp/nlg-grids-dataset

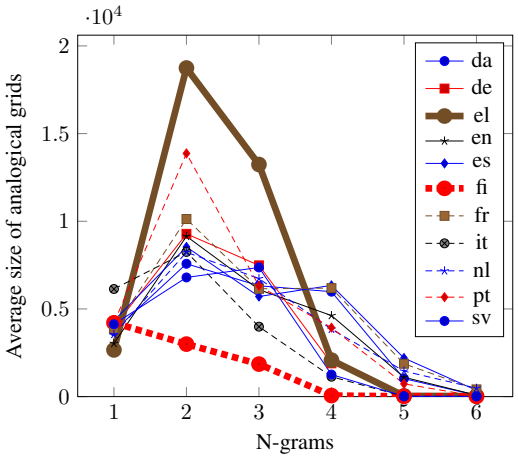
Number of analogical grids produced

Language	N-grams					
	N = 1	N = 2	N = 3	N = 4	N = 5	N = 6
da	1,581	7,352	3,299	942	679	673
de	1,451	5,634	2,177	884	652	669
el	591	856	867	731	710	657
en	1,723	9,754	5,239	1,034	860	813
es	2,806	12,342	6,587	1,088	840	708
fi	4,077	1,506	373	303	264	235
fr	2,282	11,170	4,287	940	870	844
it	3,586	15,999	3,152	791	767	676
nl	1,408	9,522	4,331	879	861	812
pt	2,514	11,164	4,690	940	785	734
sv	1,579	5,002	1,553	759	672	633



Average size of analogical grids

Language	N-gram					
	N = 1	N = 2	N = 3	N = 4	N = 5	N = 6
da	3735	7576	6320	5985	1025	76
de	3967	9284	7486	1896	38	18
el	2663	18738	13237	2081	40	29
en	3033	9143	6102	4622	1104	85
es	4333	8534	5709	6343	2199	359
fi	4212	2994	1855	61	18	15
fr	3836	10128	6149	6188	1873	413
it	6137	8252	3988	1136	16	14
nl	3543	8234	6732	3843	1457	480
pt	4203	13870	6314	3934	728	22
sv	4126	6796	7365	1258	19	18



Average saturation of analogical grids

Language	N-gram					
	N = 1	N = 2	N = 3	N = 4	N = 5	N = 6
da	93.71	95.20	96.01	97.21	98.58	98.64
de	93.77	95.10	95.35	97.94	98.44	98.45
el	94.48	95.13	95.74	97.88	98.14	98.88
en	93.79	95.63	96.06	97.68	99.03	98.19
es	93.78	95.45	96.55	97.35	98.94	98.13
fi	94.31	96.12	96.30	97.29	98.58	98.43
fr	94.19	95.57	96.78	97.72	98.94	98.47
it	93.57	95.55	97.37	98.80	98.92	99.40
nl	93.61	95.29	96.52	98.29	98.88	99.10
pt	93.89	95.29	96.42	98.16	99.01	98.72
sv	93.45	95.10	95.94	97.92	98.76	99.27

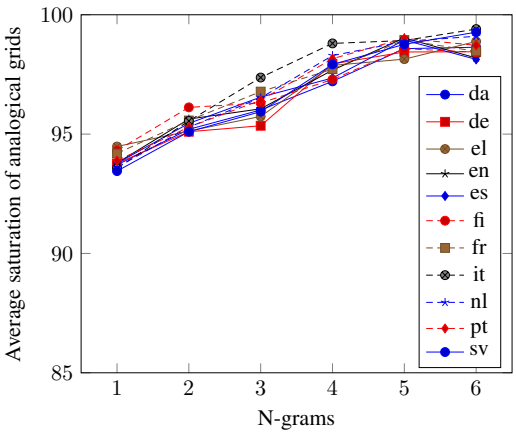


Figure 7: Number (*top*), average size (*middle*), and average saturation (*bottom*) of analogical grids produced for the first 1,000 lines of Europarl with different size of N-grams.

As can be seen from Figure 7, we observe a similar behaviour for average size and number of analogical grids produced. The average size of the analogical grids in all languages reaches its peak on bigrams and then goes down until sixgrams, except for Swedish. The same interesting phenomenon in Finnish occurs again as it has lower average size for longer N-gram. On the other hand, Greek has the lqrgest average size for bigrams and trigrams despite having the smallest average size for unigrams. This might be caused by Greek having a richer system of auxiliary verbs. Said in another way, one word in other languages may be expressed with two or more words in Greek. Swedish is another anomaly: the largest average size for Swedish is observed on trigrams instead of bigrams.

3.4. Saturation of Analogical Grids

The number of empty cells of an analogical grid can be roughly considered the number of possible forms not present in the data. We call *saturation*[3] the ratio of the number of non-empty cells and the size of an analogical grid in percentage. E.g., the analogical grid in Figure 2 has 4 empty cells; its saturation is thus: $(1.0-4/(4\times5))\times100 = 80\,\%$.

$$\text{Saturation} = 100 - \frac{\text{Number of empty cells} \times 100}{\text{Total number of cells}} \quad (6)$$

The average saturation of analogical grids produced in 11 languages of our Europarl data tends to rise from unigrams to sixgrams (see Figure 7 (*bottom*)). This indicates that analogical grids on longer N-grams are more dense. However, it should be kept in mind, by referring to the results on number of analogical grids and their average size, that they are much less numerous and much smaller.

Empty cells can be filled by potential word forms. Thus, they illustrate productivity in language. The confidence with which empty cells can be filled is of course linked with the problem of over-generation by analogy. These potential word forms are basically unseen from the corpus used to build the analogical grids. Studying the productivity of correct word forms from analogical grids would be another intriguing experiment to perform. We hope that such a resource can be used by other researchers for re-inflection tasks or to address to some extent the out-of-vocabulary (OOV) problem.

4. Conclusion

We released a Python module for the production of analogical grids from word forms contained in a corpus. Several additional functions are implemented for the sake of language productivity analysis and for the use of richer features than just character counts.

In addition, we released a complete data set which contains analogical clusters and analogical grids built on 1,000 corresponding lines in 11 European languages extracted from the Europarl corpus v.3.

We hope that such module and data will be used by researchers in comparative linguistic studies, in re-inflection tasks or other tasks of Natural Language Processing. We hope that the tools provided in the module will be used for the study of other languages than those of the resource released.

5. Acknowledgements

This work was supported by a JSPS Grant, Number 15K00317 (Kakenhi C), entitled Language productivity: efficient extraction of productive analogical clusters and their evaluation using statistical machine translation.

[3] In (Chan, 2008, p. 79), saturation is the maximal proportion of word forms attested for any one lemma of a given paradigm. Here we use the term for each entire table.

6. Bibliographical References

Chan, E. (2008). *Structures and distributions in morphology learning*. Ph.D. thesis, University of Pennsylvania.

Fam, R. and Lepage, Y. (2016). Morphological predictability of unseen words using computational analogy. In *Proceedings of the Computational Analogy Workshop at the 24th International Conference on Case-Based Reasoning (ICCBR-CA-16)*, pages 51–60, Atlanta, Georgia.

Fam, R. and Lepage, Y. (2017). A study of the saturation of analogical grids agnostically extracted from texts. In *Proceedings of the Computational Analogy Workshop at the 25th International Conference on Case-Based Reasoning (ICCBR-CA-17)*, pages 11–20, Trondheim, Norway.

Hathout, N. (2008). Acquisition of the morphological structure of the lexicon based on lexical similarity and formal analogy. In *Proceedings of the 3rd Textgraphs workshop on Graph-based Algorithms for Natural Language Processing*, pages 1–8, Manchester, UK, August. Coling 2008 Organizing Committee.

Hathout, N. (2009). Acquisition of morphological families and derivational series from a machine readable dictionary. *CoRR*, abs/0905.1609.

Lepage, Y. (1998). Solving analogies on words: an algorithm. In *Proceedings of the 17th international conference on Computational linguistics (COLING 1998)*, volume 1, pages 728–734. Association for Computational Linguistics.

Lepage, Y. (2014). Analogies between binary images: Application to Chinese characters. In Henri Prade et al., editors, *Computational Approaches to Analogical Reasoning: Current Trends*, pages 25–57. Springer, Berlin, Heidelberg.

Mansion, J. E., (1981). *Harrap's New Shorter French and English Dictionary*. George G. Harrap Co. Ltd, London, Paris, Stuttgart.

Neuvel, S. and Fulop, S. A. (2002). Unsupervised learning of morphology without morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 31–40. Association for Computational Linguistics, July.

Singh, R. and Ford, A. (2000). In praise of Sakatayana: some remarks on whole word morphology. In Rajendra Singh, editor, *The Yearbook of South Asian Languages and Linguistics-200*. Sage, Thousand Oaks.

Sylak-Glassman, J., Kirov, C., Post, M., Que, R., and Yarowsky, D., (2015a). *A Universal Feature Schema for Rich Morphological Annotation and Fine-Grained Cross-Lingual Part-of-Speech Tagging*, pages 72–93. Springer International Publishing, Cham.

Sylak-Glassman, J., Kirov, C., Yarowsky, D., and Que, R. (2015b). A language-independent feature schema for inflectional morphology. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 674–680, Beijing, China, July. Association for Computational Linguistics.

7. Language Resource References

Christodouloupoulos, Christos. (2015). *A massively parallel corpus: the Bible in 100 languages*. Distributed via github[4], bible-corpus v1.8.

Christo Kirov and John Sylak-Glassman and Roger Que and David Yarowsky. (2016). *UniMorph Data*. UniMorph Project, distributed via web[5].

Koehn, Philipp. (2007). *European Parliament Proceedings Parallel Corpus*. Distributed via web[6], Europarl v3.

[4] github.com/christos-c/bible-corpus

[5] unimorph.org

[6] statmt.org/europarl/

The Automatic Annotation of the Semiotic Type of Hand Gestures in Obama's Humorous Speeches

Costanza Navarretta
University of Copenhagen
Emil Holms Kanal 2
2300 Copenhagen S - DK
costanza@hum.ku.dk

Abstract

This paper describes a pilot study act to investigate the semiotic types of hand gestures in video-recorded speeches and their automatic classification. Gestures, which also comprise e.g. head movements and body posture, contribute to the successful delivery of the message by reinforcing what is expressed by speech or by adding new information to what is uttered. The automatic classification of the semiotic type of gestures from their shape description can contribute to their interpretation in human-human communication and in advanced multimodal interactive systems. We annotated and analysed hand gestures produced by Barack Obama during two speeches at the Annual White House Correspondent Dinners and found differences in the contexts in which various hand gesture types were used. Then, we trained machine learning algorithms to classify the semiotic type of the hand gestures. The F-score obtained by the best performing algorithm on the classification of four semiotic types is 0.59. Surprisingly, the shape feature that contributes mostly to classification is the trajectory of the left hand. The results of this study are promising, but they should be tested on more data of different type, produced by different speakers and in more languages.

Keywords: Multimodal Communication, Automatic Classification, Hand Gestures

1. Introduction

In face-to-face communication, people express their message both through the auditory modality, speech, and the visual modality, gestures, which comprise head movements, body postures and hand gestures. Co-speech gestures are not redundant and contribute to the delivery of the message by reinforcing what it is said or providing new content (McNeill, 2005; Kendon, 2004). Although the number of audio- and video-recorded speeches and conversations on the internet is growing every day, there is still a lack of freely available multimodally annotated data of different types of face-to-face communication. Since the manual annotation of gestures is extremely resource consuming, it is vital to address not only the automatic identification of occurrences of gestures and their physical characteristics, but also their interpretation. The automatic identification of hand gestural units from videos has been addressed by numerous studies, but identifying gestures from all kinds of videos is still not possible, and for this reason many studies use tracking devices or pose particular restrictions to lighting and settings of the videos (Keskin et al., 2011). Therefore, there is still a need to identify the function of gestures automatically, and distinguishing their semiotic type is a first step toward their interpretation which should also include the context in which the gestures are performed.

This paper presents a pilot work aimed to the automatic classification of the semiotic type of hand and arm gestures in two speeches of Barack Obama using coarse-grained descriptions of the gestural shape. An analysis of hand gesture types in these data is also provided. The two humorous speeches were held at the Annual White House Correspondents' Association Dinner in 2011 and in 2016. Obama's speeches are interesting because he is an excellent speaker and his ability in presenting his message in a clear and convincing way has been praised by both press corps and researchers, e.g. (Cooper, 2011). The video-recordings of these speeches are freely available on the internet and can be shared, according to the U.S.A. legislation, since they were transmitted by news channels.[1]

The article is organized as follows. In section 2., we shortly discuss background literature. In section 3., we describe the speeches and their annotations. Then, we present an analysis of the shape and semiotic types of the hand gestures based on these annotations (section 4.), and in section 5. we account for our machine learning experiments and their results. Finally, we evaluate and discuss the results of the analysis and classification experiments, and we suggest future work (section 6.).

2. Background Literature

Communicative gestures are temporally, semantically and pragmatically related to speech (McNeill, 1992; McNeill, 2005). These gestures are known as co-speech gestures and have several and often co-occurring functions. Researchers have shown that speech and gestures influence each other in different ways. For example, there is a relation between gesture and the syntactic structure of speech (McNeill, 1992), intonation (Loehr, 2004; Loehr, 2007) and lexical retrieval (Krauss et al., 2000).

Various categories of gestures have been proposed in the literature, inter alia (Ekman and Friesen, 1969; W.M.Wundt, 1973; Nespoulous and Lecours, 1986; McNeill, 1992; McNeill, 2005; Allwood et al., 2007) and the relation between types of gesture and their function has also been pointed out by for example Kendon (2004). In this work, we use the semiotic categories proposed by Peirce (1931) which were adopted in the MUMIN annotation framework (Allwood et al., 2007). The classification distinguishes three main categories *indexical*, *iconic* and *symbolic* gestures. In-

[1] The annotations of the speeches will be made available via the Danish CLARIN infrastructure.

dexical gestures have a direct connection with the objects which they denote and they comprise *deictic* gestures, also known as pointing gestures, and *non-deictic* gestures such as displays and beats. Beats are also known as batonic gestures and in these data all indexical non-deictic gestures are beats. Iconic gestures denote their objects by similarity and include *metaphoric* gestures. An example of iconic gesture is the quick alternating movement of index and middle fingers referring to a running event. An example of the metaphoric subclass of iconics is shown in Figure 1 a snapshot from *Speech16* that shows Obama making a grasping both-hands gesture in front of his chest while uttering *feel the burn*, the slogan of Bernie Sanders' campaign as democratic president candidate.



Figure 1: Metaphoric Gesture Co-occurring with the Speech Segment *feel the burn*

Finally, symbolic gestures, also called emblems, are conventionalized signs which are culture dependent. A symbolic gesture is for example the victory sign, in which the spread index and third finger resemble the letter *V*. The distinction between iconic and symbolic gestures is not always clear-cut as accounted for in the so-called Kendon's continuum (McNeill, 1992; Kendon, 2004). Furthermore, gestures often belong to more semiotic types at the same time since they are multifunctional (Allwood et al., 2007).
The automatic identification of hand gestures has been investigated in order to provide the automatic interpretation of sign languages (Keskin et al., 2011; Gebre et al., 2014), or to identify hand gestural units and their placement in the gestural space (Schreer et al., 2014). Better results have been obtained using tracking devices of different type (Keskin et al., 2011; Alexanderson et al., 2016). The present work uses coarse grained manual annotations of the shape and semiotic type of hand gestures.

3. The Data

Our data consists of the annotations of two audio- and video-recorded speeches by Barack Obama at the the 2011 and 2016 Annual White House Correspondents' Association Dinner. The speeches are called *Speech11* and *Speech16* in what follows.
During the Annual White House Correspondents' Association Dinners, the U.S.A. president holds a speech in which he makes fun of himself, his wife, his collaborators and political adversaries. Several recordings of these speeches are available on the internet (YouTube). We used the official recordings of the White House, which were available at `http:\www.WH.gov` while Obama was president. Obama was video-recorded frontally as is shown in Figure 2 and 3 which are snapshots from the two talks.



Figure 2: Snapshot from the 2011 Speech



Figure 3: Snapshot from the 2016 Speech

Obama' s speech and audience reaction were transcribed reusing existing transcriptions of the talks from newspapers. The transcriptions comprise voiced segments, silent and filled pauses, and audience responses in the form of cheers, laughter and/or applause. The gestures of Obama were manually annotated in the ANVIL tool (Kipp, 2004). The video segments annotated in *Speech11* have a duration of 13 minutes and 22 seconds while the annotations of the *Speech16* video segments cover 30 minutes. In order to calculate the ratio hand gesture per second, we excluded the time during which Obama did not move his hands because the audience were laughing and/or applauding. The resulting data duration is 8 minutes for *Speech11* and 19 minutes for *Speech16*. A description of the annotation of the speeches and a study of the relation between speech pauses, gestures and audience response is in (Navarretta, 2017). The semiotic types of hand gestures were annotated for this study by one annotator (the author of the paper), and the annotations have been revised by the same annota-

tor after several months. The Cohen' s kappa-score (Cohen, 1960) for intra-coder agreement on the two annotations is 0.85.

The shape and semiotic type features of gestures follow the MUMIN annotation framework (Allwood et al., 2007). The shape features which are relevant for the present study are in the first six rows of Table 1 and are a subset of the features proposed in (McNeill, 1992). The seventh and last row of the table contains the semiotic types which are used in these study. Many gestures belong to more semiotic

Attribute	**Value**
Handedness	BothHandsSym, BothHandsAsym, RightSingleHand, LeftSingleHand
HandRepetition	Single, Repeated
Fingers	IndexExtended, ThumbExtended, AllFingersExtended, FingersOther
TrajectoryLeftHand	LeftHandForward, LeftHandBackward, LeftHandSide, LeftHandUp, LeftHandDown, LeftHandComplex, LeftHandOther
TrajectoryRightHand	RightHandForward, RightHandBackward, RightHandSide, RightHandUp, RightHandDown, RightHandComplex, RightHandOther
PalmOrientation	PalmUp, PalmDown, PalmSide, PalmVertical, PalmOther
SemioticType	IndexicalDeictic, IndexicalNonDeictic, Iconic, Symbolic

Table 1: Shape and Semiotic Features of Hand Gestures

types at the same time. For example, many deictic and iconic gestures are also beats. We have only annotated the most specific semiotic type and therefore beat gestures are only coded when the gestures do not also fall under another category. We only found five metaphorical iconic gestures and therefore we did not distinguish them from the other iconic gestures in the machine learning experiments. In the videos, we also annotated non communicative hand gestures such as Obama touching the cuffs of his shirt. These gestures are often called *adaptors* (Ekman and Friesen, 1969). The annotations of these gestures are not included in this study.

4. Data Analysis

The total number of the communicative hand gestures performed by Obama in the speeches is 298, that is he produces 0.18 hand gestures per second. Obama produces significantly less hand gestures in *Speech11* than in *Speech16*. More specifically, he produces 59 hand gestures (0.12 gestures per second) in *Speech11* and 239 hand gestures (0.21 gestures per second) in *Speech16*. The difference is significant: Chi square equals 13.2905, df=1, and $p = 0.000267 < 0.001$.

Table 2 shows the relation between handedness and semiotic types in these data. In the speeches, Obama performs

Handedness	Dei	NonDeic	Icon	Symb	Total
BothHSym	21	47	27	1	96
RightH	67	30	15	6	118
LeftHand	36	30	13	2	81
BothHAsym	0	0	1	0	1
Total	125	107	56	11	298

Table 2: Handedness and Semiotic Types

more often gestures with his right hand than with his left hand or both hands. The second most frequently performed hand gestures are both-hands, and only one out of the 97 both-hands gestures is asymmetric. This asymmetric gesture is iconic. Not surprisingly, the most common semiotic types of hand gestures performed by Obama in the two speeches are indexical deictic and indexical non-deictic. The frequency of deictic gestures is specific to these humorous speeches in which Obama often points at individuals in the audience. Deictic gestures are performed with both hands and one hand, and the qualitative analysis of the data shows that the choice of the hand in one-hand deictics mostly depends on the physical position of the people at whom Obama points. If the people sit on his right side, he often uses his right hand, while he uses his left hand in the opposite case. Obama points with both hands at groups of people, or while referring to persons or objects not in the room. This also includes reference to abstract entities. In some cases he also points at himself with both hands. An interesting finding is that when gestures co-occur with a sentence containing a negation Obama uses his left hand or both hands.

Obama performs exactly the same number of indexical non-deictic hand gestures (beats) with his right and left hand, and the largest number of beats is produced with both hands.

Table 3 shows the percentage of repeated hand gestures for each of the categories in Table 2. Only 18% of the hand

Handedness	Dei	NonDeic	Icon	Symb	Total
BothHSym	5%	25%	31%	0	22%
RightHand	7%	3%	13%	0	5%
LeftHand	6%	37%	31%	0	21%
BothHAsym	0	0	0	0	0
Total	6%	30%	25%	0	18%

Table 3: Repetition and Semiotic Types

gestures produced by Obama are repeated, and not surprisingly, the most frequently repeated hand gesture type is Indexical Non-deictic (beat) followed by Iconic. The table also shows that Obama repeats more often both-hands and left-hand gestures (22% and 21% of the occurrences respectively) than right-hand ones (5% of occurrences).

5. Classifying the Semiotic Types of Hand Gestures

The aims of our machine learning experiments were to determine a) to what extent the shape features of hand gestures are useful to train classifies to distinguish the semiotic type of hand gestures, b) which classifiers perform best on these data, and c) which shape features contribute mostly to classification. The shape features and the semiotic types used in the machine learning experiments are those in Table 1 with one exception: We merged the two categories BothHandsSymmetric and BothHandsAsymmetric since there was only one occurrence of asymmetric hand gestures in the data.

The machine learning experiments were run in WEKA (Witten and Frank, 2005). The algorithms which we tested are a support vector classifier (SMO), Naive Bayes, Bayes Network, Simple Logistic, LBR, LMT, Random Forest and a Multilayer Perceptron with backpropagation. The results of the classifiers were validated with ten-fold cross-validation, and are reported as Precision (P), Recall (R) and weighed F-score, which is calculated according to the following equation:

$$F1 = \frac{2 \times P \times R}{P + R}. \quad (1)$$

A majority classifier, which always chooses the most frequently occurring semiotic type, is the baseline. In the first group of experiments, we trained the classifiers on all features, and the results of these experiments are in Table 4. All algorithms performed significantly better than the baseline. Significance was measured with paired corrected t-test and significance level $p < 0.001$. The classifier which performed best is Bayes Network (F-score 0.59) although Naive Bayes and LBR obtained the same F-score, and recall while they gave a slightly worse precision (the difference between the results of the three classifiers is not statistically significant). LMT and SMO obtained an F-score of 0.57 and 0.56 respectively, while the F-score of the Multilayer Perceptron is 0.53 and of Random Forest is 0.52 (the difference with respect to the results of the Bayes Network classifier are statistically significant at significance level $p < 0.01$).

Algorithm	P	R	F
Baseline	0.18	0.42	0.25
Bayes Network	**0.59**	**0.6**	**0.59**
Naive Bayes	0.58	0.6	0.59
LBR	0.58	0.6	0.59
SMO	0.56	0.58	0.57
LMT	0.55	0.58	**0.56**
Simple Logistic	0.55	0.58	0.56
Random Forest	0.52	0.54	0.52
Multilayer Perceptron	0.53	0.53	0.53

Table 4: Classification of Hand Gestures

The confusion matrix obtained with the Bayes Network classifier is given in Table 5. The confusion matrix shows that the classes which are identified more correctly are IndexicalDeictic and IndexicalNon-deictic which are the most frequently occurring classes in the data. All symbolic gestures, which are seldom in this data, are classified incorrectly.

a	b	c	d	classified as
21	25	1	9	a = Iconic
14	60	1	32	b = IndexicalNon-deictic
0	3	0	6	c = Symbolic
2	25	1	98	d = IndexicalDeictic

Table 5: Confusion Matrix of Bayes Network Trained on All Shape Features

Attribute selection performed in WEKA with the *CfsSubsetEval*[2] applying the *BestFirst* method suggests HandRepetition, Fingers, Trajectory-Left-Hand and Trajectory-Right-Hand as the best features for classifying the semiotic type of hand gestures. To test how each feature contributes to classification, we performed a second group of experiments in which we trained Bayes Network classifier, the best performing classifier when the training set contains all features, on one shape feature at a time. The results of these experiments are in Table 6. The Bayes Network classifier performs best when trained on the trajectory of the left hand, and this is a bit surprising since it is not the most common feature in the dataset. Information about handedness is the feature that gives the second best results. Also information about which fingers are involved in the gestures, the trajectory of the right hand and whether the gesture is single or repeated are useful when identifying the semiotic types of hand gestures. The feature that contributes less to the identification of the semiotic types of hand gestures is that describing the direction of the palm.

Features	P	R	F
all	0.59	0.6	0.59
Handedness	0.39	0.51	0.43
HandRepetition	0.41	0.5	0.41
Fingers	0.44	0.49	0.43
LeftHandTrajectory	**0.49**	**0.52**	**0.5**
RightHandTrajectory	0.45	0.48	0.42
PalmDirection	0.25	0.38	0.27

Table 6: Feature Contribution to Bayes Network Classification

Table 7 shows the confusion matrix of the Bayes Network classifier trained on the trajectory of the left hand. As it can be seen from Table 7 and 5 that the classifier's performance decreases for all classes compared to the results of the same classified trained on all shape features. The worse results are those obtained for iconic gestures. This is not surprising since they are often produce with both hands.

Table 8 is the confusion matrix of the Bayes Network clas-

[2]The CfsSubsetEval calculates the predictive ability of each feature and considers the degree of redundancy among subsets of features.

a	b	c	d	classified as
11	26	0	19	a = Iconic
17	50	0	40	b = IndexicalNon-deictic
1	0	0	8	c = Symbolic
1	30	0	95	d = IndexicalDeictic

Table 7: Confusion Matrix of Bayes Network Trained on Left Hand Trajectory

sifier trained on the Handedness feature. IndexicalNon-

a	b	c	d	classified as
0	28	0	28	a = Iconic
0	47	0	60	b = IndexicalNon-deictic
0	1	0	8	c = Symbolic
0	22	0	104	d = IndexicalDeictic

Table 8: Confusion Matrix Bayes Network Trained on Handedness

deictic and IndexicalDeictic are the two classes which are identified best while all iconic and symbolic gestures are wrongly classified when the Bayes Network classifier is trained on the feature Handedness.

6. Discussion and Future Work

Obama produces significantly more hand gestures in *Speech16* than in *Speech11* and the most frequently produced hand gestures in the two speeches are the pointing gestures (Indexical Deictic in our classification). The frequency of gestures pointing at individuals in the audience are typical of these speeches since Obama makes fun, greets or praises persons in the audience while he points at them. The second most common hand gestures are beats (IndexicalNon-deictic), but nearly all other gestures have a batonic component.

With respect to handedness,most of Obama's hand gestures are right-hand or both-hands gestures. All both-hands gestures in these data are symmetric a part from one asymmetric iconic gesture. Under 20% of the hand gestures produced by Obama are repeated and the most frequently repeated hand gesture types are IndexicalNon-deictic and Iconic. With respect to handedness, repetition mostly involves both-hands gestures (22% of the occurrences) and left-hand gestures (21% of the occurrences) while right-hand gestures are only repeated in 5% of their occurrences. The qualitative analysis of the data shows that Obama often points at persons in the room with the hand that is on the same side as these persons in the room. Thus, Obama points with his left hand at people sitting in the left part of the room with respect to him and with his right hand at people sitting on his right side. Obama often uses both hands or his left hand when referring to objects or people not present in the room and when his hand gesture refers to abstract objects.

Hand gestures co-occurring with and related to utterances or fragments of speech which contain a negation are both-hands or left-hand. Whether these findings reflect a general tendency in Obama's speeches should be investigated on more data.

The results of our machine learning experiments aimed to automatically classify the semiotic type of hand gestures from their shape features show that the best performing algorithm on these data is Bayes Network which obtained an F-score of 0.59. All the classifier we tested gave significantly better results than the majority classifier which always chooses deictic as the semiotic type of the gestures. The worse performance was obtained by a Random Forest classifier and a Multilayer Perceptron with backpropagation probably because of the limited size of the data.

The results of classification confirm that there is a relation between form and function of hand gestures (Kendon, 2004) and, not surprisingly, they also indicate that gestures can only to some extent be interpreted on the basis of their form. In some cases, they must be interpreted in the context in which they occur since gestures can be ambiguous as words. For example, forming a circular shape with the thumb and the index finger can be in the same culture the *OK* symbol, or can refer to a round object, a ring, the number zero etc. Moreover, in our experiments we did not distinguish the metaphoric subclass of iconic gestures, even though this information is available in the annotations, because there were too few metaphoric gestures in the data. Furthermore, the content of speech must be used to distinguish metaphoric gestures from generic iconic gestures, while in our machine learning experiments we only included shape descriptions of the gestures.

In the second group of experiments, we tested which features contribute mostly to classification, and the results of these experiments show that training the Bayes Network classifier on data annotated with the trajectory of the left hand gives the best results. This is surprising since the trajectory of the left hand is not the most frequent feature in the data. The reason for this can be that the three most common semiotic types (IndexicalDeictic, IndexicalNon-deictic and Iconic) are more equally distributed over the left-hand trajectory feature than in the other features. Also information about handedness, the trajectory of the right hand, the position of the fingers and whether the gestures are repeated or not gave good results, while information about the direction of the palm is the feature that contributed less to classification. Since the position of the palm changes during many hand gestures, the value of this feature only refers to the direction of the palm at the gestural stroke. This might well be the reason why this feature is not very useful for identifying the semiotic type of hand gestures.

The results of our machine learning experiments are promising since only few general shape features were used as training data. However, the shape and form of hand gestures were annotated manually and therefore classification should also be tested on automatically produced shape annotations, and on more fine-grained annotations. Furthermore, some types of gesture are culture specific and there is a great variation in the way gestures are produced by different subjects (Kendon, 2004; Navarretta, 2012). It must also be noted that in the two speeches Obama is reading from a manuscript and he produces very clear hand ges-

tures, which is not necessarily the case for other individuals or in other communicative situations. Therefore, the classification of semiotic types of hand gestures should also be applied to gestures performed by other subjects in other contexts and cultures. The classification experiments could also be tested on other gesture types, such as head movements and body posture.

A future direction of research could also be considering the form of the hand during the gestures, using a kind of hand gesture lexicon as proposed e.g. in (Poggi and Caldognetto, 1997) and (Kipp, 2004).

7. Bibliographical References

Alexanderson, S., House, D., and Beskow, J. (2016). Automatic annotation of gestural units in spontaneous face-to-face interaction. In *Proceedings of the Workshop on Multimodal Analyses Enabling Artificial Agents in Human-Machine Interaction*, MA3HMI '16, pages 15–19, New York, NY, USA. ACM.

Allwood, J., Cerrato, L., Jokinen, K., Navarretta, C., and Paggio, P. (2007). The mumin coding scheme for the annotation of feedback, turn management and sequencing. *Multimodal Corpora for Modelling Human Multimodal Behaviour. Special Issue of the International Journal of Language Resources and Evaluation*, 41(3–4):273–287.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Cooper, M. M. (2011). Rhetorical agency as emergent and enacted. *College Composition and Communication*, 62(3):420–449.

Ekman, P. and Friesen, W. V. (1969). The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica*, 1(1):49–98.

Gebre, B. G., Crasborn, O., Wittenburg, P., Drude, S., and Heskes, T. (2014). Unsupervised Feature Learning for Visual Sign Language Identification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 370–376. ACL.

Kendon, A. (2004). *Gesture: Visible Action as Utterance*. Cambridge University Press.

Keskin, C., Aran, O., and Akarun, L. (2011). Hand gesture analysis. In Albert Ali Salah et al., editors, *Computer Analysis of Human Behavior*, pages 125–149. Springer London, London.

Kipp, M. (2004). *Gesture Generation by Imitation - From Human Behavior to Computer Character Animation*. Ph.D. thesis, Saarland University, Saarbruecken, Germany, Boca Raton, Florida, dissertation.com.

Krauss, R., Chen, Y., and Gottesman, R. F. (2000). Lexical gestures and lexical access: a process model. In D. McNeill, editor, *Language and gesture*, pages 261–283. Cambridge University Press.

Loehr, D. P. (2004). *Gesture and Intonation*. Ph.D. thesis, Georgetown University.

Loehr, D. P. (2007). Aspects of rhythm in gesture and speech. *Gesture*, 7(2):179–214.

McNeill, D. (1992). *Hand and Mind: What Gestures Reveal About Thought*. University of Chicago Press, Chicago.

McNeill, D. (2005). *Gesture and thought*. University of Chicago Press.

Navarretta, C. (2012). Individuality in Communicative Bodily Behaviours. In A. Esposito, et al., editors, *Behavioural Cognitive Systems*, volume 7403 of *Lecture Notes in Computer Science*, pages 417–423. Springer Verlag, Berlin Heidelberg.

Navarretta, C. (2017). Barack Obama's pauses and gestures in humorous speeches. In *Proceedings of the 4th European and 7th Nordic Symposium on Multimodal Communication (MMSYM 2016), Copenhagen, 29-30 September 2016*, number 141 in Linköping University Conference Proceedings, pages 28–36. Linköping University Electronic Press, Linköpings universitet.

Nespoulous, J.-L. and Lecours, A. (1986). Gesture: Nature and function. In J.-L. Nespoulous, et al., editors, *The Biological Foundations of Gestures: Motor and Semiotic Aspects*, chapter 2, pages 49–62. Lawrence Erlbaum Association, Hillsdale,N.Y.

Peirce, C. S. (1931). *Collected Papers of Charles Sanders Peirce, 1931-1958, 8 vols*. Harvard University Press, Cambridge, MA.

Poggi, I. and Caldognetto, E. (1997). *Mani che parlano*. Unipress, Padova.

Schreer, O., Masneri, S., Skomroch, H., and Lausberg2, H. (2014). Coding Hand Movement Behavior and Gesture with NEUROGES Supported by Automatic Video Analysis. In A.J. Spink, et al., editors, *Proceedings of Measuring Behavior 2014,*, August.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, second edition.

W.M.Wundt. (1973). *The language of gestures*. Mouton.

WASA: A Web Application for Sequence Annotation

Fahad AlGhamdi, and Mona Diab
Department of Computer Science
The George Washington University
Washington, DC
{fghamdi, mtdiab}@gwu.edu

Abstract

Data annotation is an important and necessary task for all NLP applications. Designing and implementing a web-based application that enables many annotators to annotate and enter their input into one central database is not a trivial task. These kinds of web-based applications require a consistent and robust backup for the underlying database and support to enhance the efficiency and speed of the annotation. Also, they need to ensure that the annotations are stored with a minimal amount of redundancy in order to take advantage of the available resources(e.g, storage space). In this paper, we introduce WASA, a web-based annotation system for managing large-scale multilingual Code Switching (CS) data annotation. Although WASA has the ability to perform the annotation for any token sequence with arbitrary tag sets, we will focus on how WASA is used for CS annotation. The system supports concurrent annotation, handles multiple encodings, allows for several levels of management control, and enables quality control measures while seamlessly reporting annotation statistics from various perspectives and at different levels of granularity. Moreover, the system is integrated with a robust language specific date prepossessing tool to enhance the speed and efficiency of the annotation. We describe the annotation and the administration interfaces as well as the backend engine.

Keywords: : Code Switching, Annotation, Web Application, Sociolinguistics

1. Introduction

Code Switching (CS) is a phenomenon that occurs when multilingual speakers alternate between more than one language or dialect. This phenomenon can be observed in different linguistic levels of representation for different language pairs: phonological, morphological, lexical, syntactic, semantic, and discourse/pragmatics. CS presents serious challenges for language technologies, including parsing, Machine Translation (MT), Information Retrieval (IR) and others. A major barrier to research on CS has been the lack of large multilingual, multi-genre CS-annotated corpora. Creating such corpora involves managing many annotators working on multiple tasks at different times, consistent and robust backups of the underlying database, quality control, etc. In this paper, we present our effort in building an annotation system, WASA, that can manage and facilitate large-scale CS data annotation. WASA differs from other annotation systems in several respects. Our system has an option that can provide initial automatic tagging for specific tokens such as Latin words, URL, punctuation, digits, diacritics, emoticons, and speech effect tokens. This option increases the quality and the speed of annotation substantially. Moreover, the system is integrated with language-specific date preprocessing tool Smart Preprocessing (Quasi) Language Independent Tool (SPLIT) (Al-Badrashiny et al., 2016) to streamline raw data cleaning and preparation.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work. Section 3 describes the System Architecture. Types of users including permissions and users tasks are introduced in Section 4. The data preprocessing and cleaning are discussed in Section 5. We provide an overview of the database design in Section 6. Inter-annotator agreement, current status, and our conclusion and future work are discussed in sections 7,8 and 9, respectively.

2. Related Works

Although, many annotation tools, such as (Aziz et al., 2012), (Cunningham et al., 2009), (Kahan et al., 2002), MnM (Vargas-Vera et al., 2002), GATE ((Cunningham et al., 2009); (Aswani and Gaizauskas, 2009), and (Dickinson and Ledbetter, 2012)), are effective in serving their intended purposes, none of them meets the CS annotation requirements perfectly. We need a tool that can help in sequence annotating in a way that can report the time needed for annotators to get their tasks done, manage number of annotator teams, enable quality control measures and annotation statistics, and assign some initial tags to some tags automatically (e.g. punctuation, URL, emoticon, etc.)

Our tool is most similar to the annotation tool for the COLABA project (Diab et al., 2010); (Benajiba and Diab, 2010)(Benajiba and Diab, 2010; Diab et al., 2010). We specifically emulate the annotator management component in the COLABA annotation tool. Although, the code switching annotation task and manual diacritization of Standard Arabic text task are completely different tasks, the MANDIAC tool (Obeid et al., 2016), which used for diacritization annotation task, has a similar annotator management component to the WASA management component. However, the technologies used in both management components are different. For instance, WASA uses PostgreSql database to store content, while MANDIAC uses a JSON blob to store content. Two other comparable tools to ours are WebANNO (Yimam et al., 2013) and SWAT (Samih et al., 2016). They both use the latest available technologies to perform a number of linguistic annotation types. The SWAT tool is a web-based interface for annotating tokens in a sequence with a predefined set of labels. The main advantages of this tool are the simplicity of its use and instal-

lation as it only requires a modern web browser and minimum server-side requirements to get the tool work. The WebANNO tool is also a web-based tool that offers wide range of linguistic annotations tasks, e.g., named entity, dependency parsing, co-reference chain identification, and part-of speech annotation.

However, both systems SWAT and WebANNO lack of some functionalities and features that can simplify and speed up the annotation task for our purposes. In the SWAT system for example, there is no support for user roles. Therefore, some tasks such as managing the number of annotators, monitoring the progress of the annotators, assigning tasks given to the annotators, and ensuring the quality of the submitted annotation are difficult to handle or manage with only one user type. Moreover, both systems do not have the option that can provide initial automatic tagging for named entities (NE), Latin words, URL, punctuation, number, diacritics, emoticon, and speech effects tokens. We noticed that tagging these tokens automatically increases the speed of the annotation substantially. Finally, unlike both systems, our system can seamlessly integrate with language specific data preprocessing tool to streamline raw data cleaning and preparation.

3. System Architecture

WASA is a typical three-tier web-based application. The platform is divided into three tiers, each with a specific function. The first tier is a data tier that saves all metadata in PostgreSql database in addition to both the annotated and raw data files. All this data is stored on a file server. The second tier is a logical tier. It contains PHP scripts that interact with an Apache web server. It is responsible for all functionalities provided by the system to the different types of users. All requests are sent by the web server to the PostgreSql database server through a secured tunnel. The third and last tier is the presentation tier. It is browser independent, which enables accessing the system from many different clients. It provides an intuitive GUI tailored to each user type. This architecture design allows multiple annotators to work on various tasks simultaneously. On the other hand, WASA allows the admin user to manage and handle a single central database. The system can handle multiple encodings allowing for multilingual processing. Figure-1 gives a high level overview of the tool's architecture.

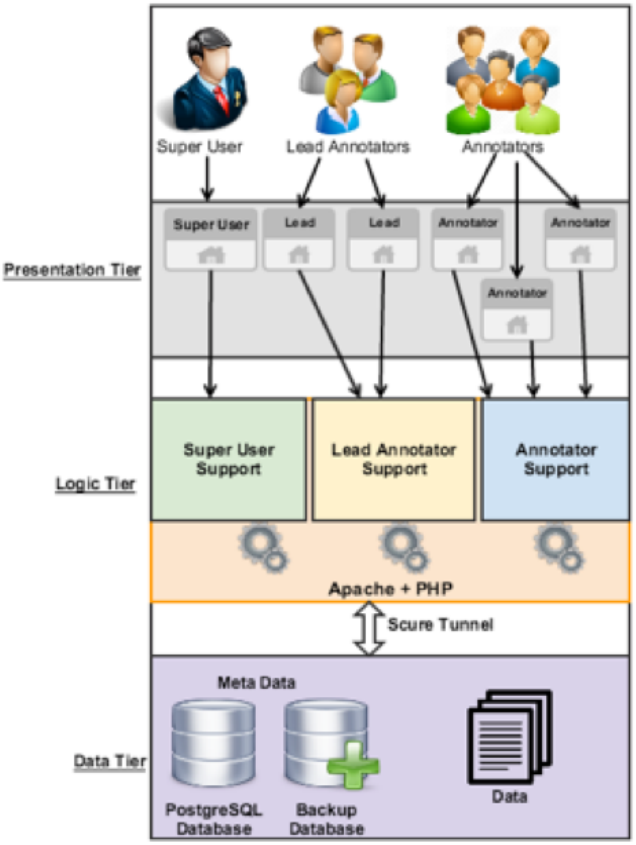


Figure 1: System Architecture

4. Types of Users

Three types of users have been considered in WASA design: Annotator, Lead-Annotator, and Super-User. Each one of these user types is given and provided with different kinds of permissions, functionalities, and privileges in order to fulfill their tasks.

4.1. Annotators

Annotators are provided the following functionalities: **1-** access assigned tasks; **2-**annotate the assigned tasks; **3-** submit annotation; **4-** check the time needed to submit one unit, e.g., post, or tweets; **5-** check the grade of the submitted work; **6-** re-annotate the rejected tasks (by rejected we mean when the annotator received a "No Pass" as a grade on their annotation task); and, **7-** save work and continue it in a later session.

Figure-2 shows an example of the annotation screen. The words of the posts or tweets that need to be annotated will be displayed as clickable units. When clicked, a pop-up screen appears to allow the annotator to choose the proper tag. To increase the speed of the annotation process, some of the words, like Named Entities and punctuations, will have an initial tag assigned automatically as part of a preprocessing step. However, the annotator is allowed to change the initial tag if he/she finds words annotated with a wrong tag. The interface uses color-coding to reflect useful information and status. For example, 'named entities' will be displayed in purple color, while Other tagged categories such as Latin, URL, punctuation, digits, diacritics, emoticons, sound effects will be displayed in the orange color. Words already annotated will be displayed in blue while words that are yet to be annotated appear in black. Figure-3 shows an example of some of the assigned tasks with information about the tasks that have been already submitted (e.g, number of annotated words, speed of annotation, path of the raw file)

4.2. Lead Annotator

For each dialect/language, there is one lead annotator only. Each lead annotator has the following functions: **1-** Annotator management, e.g., create, edit and delete annotator accounts; **2-** Tasks management; **3-** Monitor status and progress; **4-** Review and grade annotators' work; and **5-**

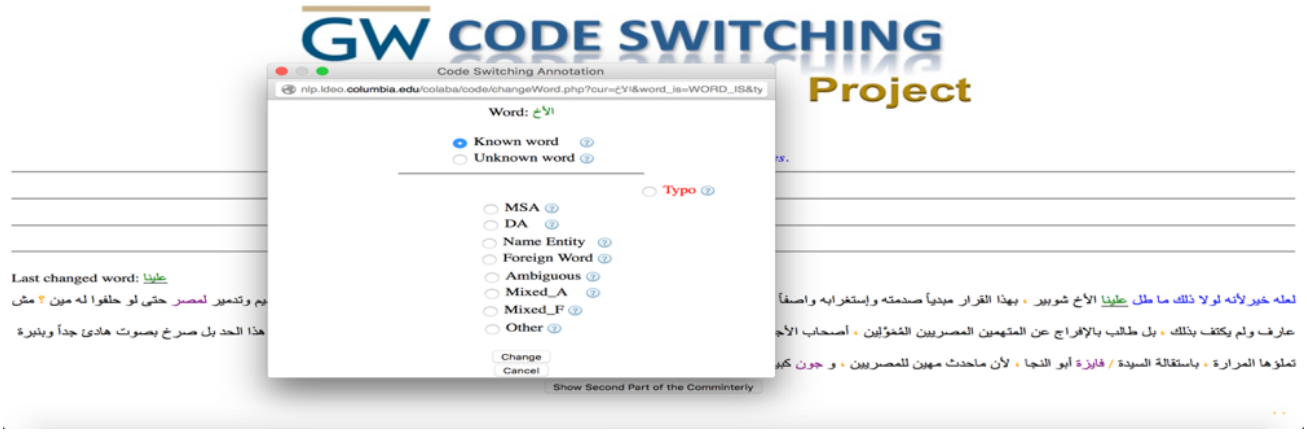


Figure 2: Annotation Screen

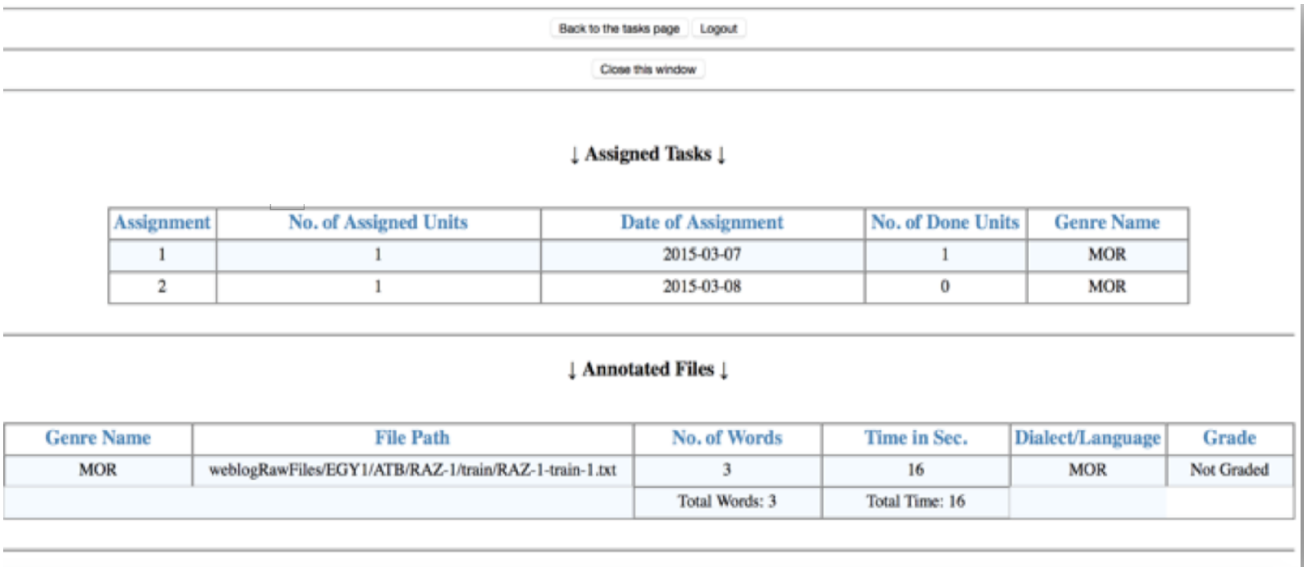


Figure 3: An example of the annotator's "Check-Status" screen

Produce quality measures like inter-annotator agreement. The system enables lead annotators to reject submitted work that does not meet the assessment criteria and add comments and feedback for the annotators to re-annotate rejected work.

4.3. Super User

There is only one Superuser account in WASA for all dialects/languages. The Superuser functions include: **1-** Database management and maintenance; **2-** Lead annotators management, **3-** Annotators management, **4-** Monitor the overall performance of the system; and **5-** Manage annotation data imports and exports.

5. Data Preprocessing and Input and Output Format

The system has the ability to integrate with language-specific date preprocessing scripts to streamline raw data cleaning and preparation. For example, for cleaning process (step-1) the system integrates the Smart Preprocessing (Quasi) Language Independent tool (SPLIT) (Al-Badrashiny et al., 2016) to handle the encoding issues (i.e, Change the character encoding to UTF8). Moreover, for the Dialectal Arabic (DA) and Modern Standard Arabic (MSA) language pair (step-2), the system integrates with the Automatic Identification of Dialectal Arabic (AIDA2) tool (Al-Badrashiny et al., 2015) to provide initial automatic tagging for named entities (NE), Latin words, URL, punctuation, number, diacritics, emoticon, and speech effects tokens. Figure-2 illustrates an example of a commentary with some pre-annotated tokens. Named entities tokens are colored purple, while punctuation and numbers are colored with orange. Both preprocessing and cleaning steps are performed offline. The Super User is the user responsible for preparing the data for annotation. Figure-5 shows the cleaning and preprocessing steps. The output file is written in a simple XML format as shown in Figure-4. The XML file includes all meta-data related to the annotation file such as the annotated, sentence id, task id, language, user id, word id, actual word, annotation tag, ...etc. The output XML is customizable. The superuser can choose what metadata to be included in the XML output file.

Our system is able to handle different types of genres such as Twitter, commentaries, conversations, or discussion fo-

rum data. Accordingly, WASA is quite robust as it is able to handle a variety of data genres and formats. For example, if the data comes from Twitter, then information like tweet id and user id needs to be preserved along with the annotation tags. If the genre of the data is discussion forums, information such as post order in the context of a conversation thread along with the names of the people who are involved in the conversation are maintained.

6. Database Design

WASA system uses a relational database to manage, handle and store all meta-data. The data stored is categorized as follow:

6.1. Profiling information

It saves information about all registered users of the system including their roles (i.e. annotator, lead annotator or superuser), login information as well as the dialect and languages for each one of them. Moreover, It contains information about different languages/dialects used in the project.

6.2. Annotation Information

This is the core part of WASA's database. It includes all meta-data related to the annotation tasks such as the number of tasks assigned to each annotator, actual annotations completed by each annotator, and temporarily saved annotations.

6.3. Assessment Information

This contains information about **1)** Task-Annotator assignment: it includes the tasks assigned to each annotator and the number of tasks that have already been annotated and submitted, the number of assigned units (tweets, posts) per each task, genre type, percentage of overlapping units (tweets, posts) shared among annotators to ease the process of calculating inter-annotator agreement, etc.; **2)** Annotator-Units assignment: It includes information about each unit (post, tweet) that is assigned to the annotators such as post/tweet-id, user-id, genre-id, task-id, path of the assigned file; Finally **3)** Language-Unit assignment: It includes information about the language/dialect id for each unit.

7. Quality Control Measures

WASA has built-in functionalities that can help in managing the inter-annotator agreement (IAA) measures for different task and report performance statistics. The lead annotator is able to specify the percentage of data annotation overlap between the annotators per task and the system manages to distribute the data and calculate the IAA. Moreover, WASA generates tag distribution, the number of annotated tokens, expected time needed to finish each assigned task, and much other quality management crucial statistics.

8. Current Status

We have tested the tool for annotation on Arabic MSA and dialectal data, Chinese-English, Spanish-English, and Hindi-English. The IAA for our the Arabic annotated data

```
<root>
  <document>
    <sentences>
      <sentence id="1">
        <tokens>
          <token id="1">
            <word>ولا</word>
            <Task>CSAnnot</Task>
            <UserID>2221</UserID>
            <language>ArabicMSA</language>
            <CharacterOffsetBegin>0</CharacterOffsetBegin>
            <CharacterOffsetEnd>3</CharacterOffsetEnd>
            <Annotation>MSA</Annotation>
            <AtomaticAnnot>No</AtomaticAnnot>
          </token>
      </sentence>
     <sentences>
  </document>
</root>
```

Figure 4: A sample of an output file

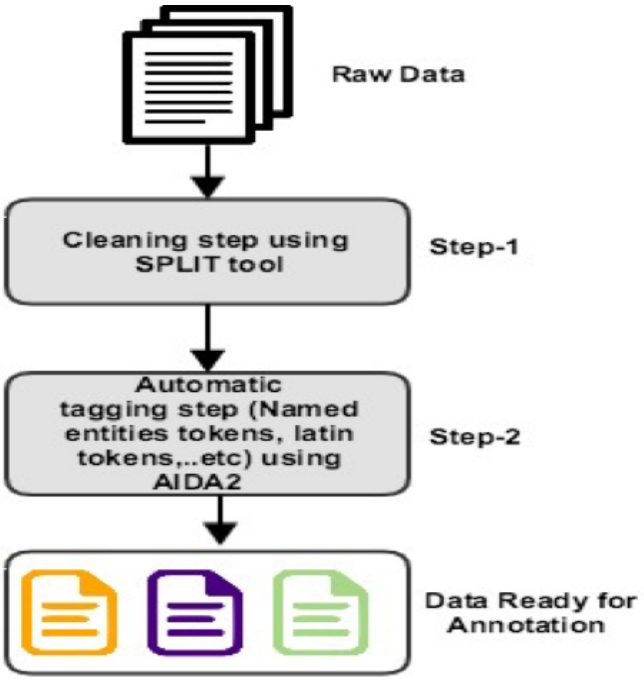


Figure 5: Preprocessing and Cleaning Steps

is ranged between 92% and 97%. Moreover, a small portion of the Code-Switching data that was released in (Diab et al., 2016) was used to test the performance of WASA. We noticed that the annotators' speed has increased substantially when we assign initial tags to some tags automatically (e.g. punctuation, URL, emoticon, ...etc.). The average time for annotating a full tweet was $\sim$ 40 seconds without using SPLIT tool (Al-Badrashiny et al., 2016), but after assigning the initial tags using the SPLIT tool, the average time for annotating a full tweet became $\sim$ 27 seconds. This results in saving much of the effort in annotating these tags.

9. Conclusion

We gave a detailed overview of our annotation system WASA. We have shown that WASA allows multiple annotator teams to work on various tasks simultaneously. Also,

we have seen that using the SPLIT tool to annotate some specific tokens automatically has helped in saving the effort and time spent by annotators. Moreover, the annotation quality of these tokens is very high. We will keep updating and modifying the current functionalities of the system as per different users type feedback. Also, we plan to add more functionality that can help in enhancing the speed, quality, and the efficiency of the CS annotation.

10. Acknowledgements

We would like to thank Mahmoud Ghoneim for his invaluable suggestions and support in the development of WASA. Also, We would like to acknowledge the useful comments by the three anonymous reviewers who helped in making this publication better presented.

11. Bibliographical References

Al-Badrashiny, M., Elfardy, H., and Diab, M. T. (2015). Aida2: A hybrid approach for token and sentence level dialect identification in arabic. In *CoNLL*, pages 42–51.

Al-Badrashiny, M., Pasha, A., Diab, M. T., Habash, N., Rambow, O., Salloum, W., and Eskander, R. (2016). Split: Smart preprocessing (quasi) language independent tool. In *LREC*.

Aswani, N. and Gaizauskas, R. (2009). Evolving a general framework for text alignment: Case studies with two south asian languages. In *Proceedings of the International Conference on Machine Translation: Twenty-Five Years On, Cranfield, Bedfordshire, UK, November*.

Aziz, W., Castilho, S., and Specia, L. (2012). Pet: a tool for post-editing and assessing machine translation. In *LREC*, pages 3982–3987.

Benajiba, Y. and Diab, M. (2010). A web application for dialectal arabic text annotation. In *Proceedings of the lrec workshop for language resources (lrs) and human language technologies (hlt) for semitic languages: Status, updates, and prospects*.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Ursu, C., Dimitrov, M., Dowman, M., Aswani, N., Roberts, I., Li, Y., et al. (2009). *Developing Language Processing Components with Gate Version 5:(A User Guide)*. University of Sheffield.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *Lrec workshop on semitic language processing*, pages 66–74.

Diab, M., Ghoneim, M., Hawwari, A., AlGhamdi, F., Al-Marwani, N., and Al-Badrashiny, M. (2016). Creating a large multi-layered representational repository of linguistic code switched arabic data. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Dickinson, M. and Ledbetter, S. (2012). Annotating errors in a hungarian learner corpus. In *LREC*, pages 1659–1664.

Kahan, J., Koivunen, M.-R., Prud'Hommeaux, E., and Swick, R. R. (2002). Annotea: an open rdf infrastructure for shared web annotations. *Computer Networks*, 39(5):589–608.

Obeid, O., Bouamor, H., Zaghouani, W., Ghoneim, M., Hawwari, A., Alqahtani, S., Diab, M., and Oflazer, K. (2016). Mandiac: A web-based annotation system for manual arabic diacritization. In *The 2nd Workshop on Arabic Corpora and Processing Tools 2016 Theme: Social Media*, page 16.

Samih, Y., Maier, W., and Kallmeyer, L. (2016). Sawt: Sequence annotation web tool. *EMNLP 2016*, page 65.

Vargas-Vera, M., Motta, E., Domingue, J., Lanzoni, M., Stutt, A., and Ciravegna, F. (2002). Mnm: Ontology driven semi-automatic and automatic support for semantic markup. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 379–391. Springer.

Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). Webanno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Sofia, Bulgaria, August. Association for Computational Linguistics.

Annotation and Quantitative Analysis of Speaker Information in Novel Conversation Sentences in Japanese

Makoto Yamazaki, Yumi Miyazaki, Wakako Kashino
National Institute for Japanese Language and Linguistics
10-2, Midori-cho, Tachikawa City, Tokyo, Japan
{yamazaki, y-miyazaki, waka}@ninjal.ac.jp

Abstract

This study undertook a quantitative lexicological analysis using attributed speaker information, and reports on the problems of creating standards when annotating speaker information (gender and age) of conversation sentences in novels. In this paper, we performed a comparative analysis of vocabulary use by gender and age of conversation sentences and descriptive part sentences, as well as on the differences between Japanese novels and translations of foreign novels. In addition, a comparison with other spoken language materials was made.

Keywords: novels, conversation sentences, speaker information, gender, age, Balanced Corpus of Contemporary Written Japanese, characteristic words, descriptive part sentences.

1. Introduction

Although conversation sentences in novels are not actual speech, they were once used for language studies. This was probably because real spoken language data was difficult to obtain, unlike today. Language researchers of course recognized that conversation sentences in novels are not actual speech. For example, Takasaki (1981: 97) stated that "conversation sentences are difficult to use as speech materials and seem to require careful handling." Regarding conversation sentences in novels, Oishi (1987: 78) stated that "in summary, we conclude that conversation sentences are not useful as speech materials." Oishi (1987: 78-79) argued, however, that conversation sentences in novels can be considered representative of spoken language and be used for data in studies that understand the fundamental attributes of conversation.

We believe that conversation sentences in novels are effective for the investigation of the following research objectives:

(1) How are conversation sentences in novels different from actual conversation sentences? Clarifying these differences will make it possible to clarify the characteristics of actual conversations and can thus promote the understanding of the diversity of spoken language.

(2) A classical literary work such as *The Tale of Genji* and conversation sentences from a modern novel can be considered the same based on the structure of the novel. Therefore, diachronically studying conversation sentences and analyzing the changes make it possible to clarify historical changes in conversation sentences and the rhetoric used in them.

(3) By using speaker information, we can pursue the usage of role language (Kinsui, 2000), because role language appears in conversation sentences in novels more often than actual spoken language.

(4) Novels consist of conversation sentences and descriptive part sentences, and by analyzing the relationship between the two, we can expect that there are implications for literary studies and stylistic studies.

2. Data

The data used in this study were taken from novel samples[1] in the Balanced Corpus of Contemporary Written Japanese (BCCWJ). The BCCWJ includes 5,138 novel samples; however, the portion used in this study consisted of 2,419 samples taken mainly from the Library sub-corpus (LB), which completed the speaker information attribution work in July 2017. Out of this corpus, we targeted 2,088 samples in the novel genre that contained gender and age information.

The analysis language unit used in this study was the short-unit word[2] (SUW). Hereafter, one short-unit word will be expressed as "one word." In the analysis, in principle, parts of speech do not include punctuation marks, blank spaces, symbols, or unknown words.

3. Providing Speaker Information

Since speaker information is not provided in the BCCWJ, we created our own criteria and provided speaker information. The following tag information was used as clues:

<speech> tag: conversation sentences as block-level elements enclosed in parentheses.

<quote> tag: portions enclosed in parentheses on one line. In some cases, parentheses were used for non-conversation purposes, such as emphasis. Figure 1 shows a sample work file (Excel).

話者名	性別	年代	年代の確信レベル	非人間	会話モード	会話認定情報1	会話認定情報2	備考	職業	相手
井関弘志	男	成年層						２６歳	会社員	宮田良子／同僚
宮田良子	女	成年層						２５歳	組合の専従書記	井関弘志（同僚）

Figure 1. An example of providing speaker information (LBt9_00022).

[1] Whether a sample is a novel or not was determined by the NDC (book classification), which is meta-information of the sample. Specifically, the NDC regards texts that correspond to 9x3 (x=1 to 8) as novels.

[2] A short-unit word is a morphological unit annotated to the BCCWJ, which corresponds approximately to a dictionary entry. One SUW consists of, at most, two small, consecutive lexical items that carry meaning.

3.1 Annotations

Table 1 shows the annotations we provided, with 1–3 being required annotations. Annotations 4–9 were provided as necessary. Annotations 10 and 11 were not provided in any of the samples.

No	Attribute	Explanation
1	Speaker Name	Character's given name
2	Gender	Male, female, unknown
3	Age	Young (ages 0–19), adult (ages 20–59), elderly (ages 60 and over)
4	Confidence Level of the Age	Check if age estimation is extremely difficult.
5	Nonhumans	Check if there are beings besides humans, such as animals or spirits/ghosts.
6	Conversation Mode	Fill in for scenes that are not normal dialog scenes "telephone, dialect, foreigners, telepathy, quotes, soliloquys" and if conversation content is only symbols, representing doubt, silence, and surprise.
7	Conversation Certification Information 1	Fill in for speech that conforms to the format of a conversation, such as soliloquys, inner speech, etc.
8	Conversation Certification Information 2	Write down the basis for determining that this conforms to a conversation.
9	Remarks	General notes
10	Profession	Profession of the person making the utterance
11	Opponent	Conversation partner

Table 1. Speaker Attributes and Explanations

3.2 Criteria for Conversation Certification

In a novel, it may be difficult to judge whether certain text represents conversation. Therefore, Miyazaki (2017) created objective criteria and ensured that the annotation work did not become arbitrary.

Specifically, relevant parts were enclosed with "" and parts assumed to be someone actually speaking in the scene were regarded as conversation. Even if parts were not enclosed with " ", if it was determined that someone was actually speaking in the scene, it was considered conversation and that basis was recorded.

Thoughts, inner speech, conversations in dreams, and telepathy were also considered as forms of conversation and speaker information was given for these as well.

3.3 Problems during Annotation

・Conversation Certification

(1) In first-person narrated novels (i.e., novels that have a narrator who speaks and tells the story), we can also consider descriptive part sentences to be conversation sentences. The same is true for *rakugo* (Japanese sit-down comedy) transcription. The speaker information annotation policy for such cases has not yet been determined.

(2) There are assumed conversations, in which it is not clear whether someone spoke or not. For example, how would you handle a line such as, "A husband who comes back home and says nothing but 'eat, bathe, sleep.'"

・Speaker Information

(1) In novels, a speaker's gender is relatively easy to identify; however, it is rare that more than only approximate ages are known. Therefore, these cases cannot be categorized into detailed age groups. Thus, we must make a rough comparison when comparing with actual spoken language.

(2) When part of another person's quoted conversation is included in the conversation, the quoted part is treated without distinction. For example, if B's conversation is quoted in A's conversation, the speaker information is given to A only and B does not receive speaker information.

4. Results

4.1 Conversation Volume

In the 2,088 samples, the number of lines[3] certified as conversation totaled 207,071.[4] There were approximately 99.2 lines per sample. On the other hand, there were 274,66 lines of descriptive part sentence.[5] The ratio of conversation sentences to descriptive part sentences was approximately 0.754.

4.2 Distribution of Attributes of People Making the Utterances

Table 2 shows the distribution of utterances by gender and age. From Table 2, we can see that male utterances make up approximately 70% and adult (20-59) utterances make up approximately 80%. Thus, novel conversations are primarily held by adult males.

	Young	Adult	Elderly	Unknown/ Other	Gender Total
Male	14,075 (6.8%)	119,507 (57.7%)	7,511 (3.6%)	1 (0.0%)	14,1094 (68.1%)
Female	13,232 (6.4%)	44,108 (21.3%)	3,070 (1.5%)	5 (0.0%)	60,415 (29.2%)
Unknown /Other	359 (0.2%)	4,964 (2.4%)	188 (0.1%)	51 (0.0%)	5,562 (2.7%)
Age Total	27,666 (13.4%)	168,579 (81.4%)	10,769 (5.2%)	57 (0.0%)	207,071 (100.0%)

( ) are percentage (%) of conversation parts overall

Table 2. Utterances by Gender and Age

[3] In the working file, the range of the <sentence> tag was one line, which we counted.

[4] Places where there was some information in both the gender and age fields.

[5] Places where there were blanks in both the gender and age fields.

Table 3 shows age and gender distribution for the calculated number of words based on short-unit words. We understand that if we compare the ratios, we find nearly the same numbers as in Table 2. This means that it is possible to estimate the approximate number of utterances (number of words) using the number of lines of utterances. With the usage of a simple calculation, there are approximately 9 words for each line of utterance.

	Young	Adult	Elderly	Unknown/Other	Gender Total
Male	99,394 (5.3%)	1,105,332 (59.3%)	72,224 (3.9%)	17,962 (1.0%)	1,294,912 (69.4%)
Female	87,287 (4.7%)	361,871 (19.4%)	26,965 (1.5%)	9,619 (0.5%)	485,742 (26.1%)
Unknown /Other	2,242 (0.1%)	50,013 (2.7%)	1,756 (0.1%)	30,252 (1.6%)	84,263 (4.5%)
Age Total	188,923 (10.1%)	1,517,216 (81.4%)	100,945 (5.4%)	57,833 (3.1%)	1,864,917 (100.0%)

Table 3. Conversation Volume by Gender and Age (number of short-unit words)

4.3 Differences between Conversation Sentences and Descriptive Part Sentences

Concerning the number of words, conversation sentences had a total of 1,864,917 words and 39,080 different words, and descriptive part sentences had a total of 4,211,887 words and 60,900 different words. In Guiraud's Index (R value), which is an indicator of vocabulary diversity, conversation sentences were 28.6, descriptive part sentences were 29.6. They showed almost the same value. We were surprised that conversation sentences were more redundant and had predicted that diversity would be much lower than descriptive par.

Table 4 shows the percentages of the parts of speech found in the conversation sentences and descriptive part sentences. Here we can identify some differences. Conversation sentences showed a lower percentage of nouns and a higher percentage of pronouns. Although numerically small, interjection values were 10 times higher in conversation sentences than in descriptive part sentences. Note that when we compared parts of speech with only the top 100 frequency words, there were 41 particle and auxiliary verbs in conversation sentences and 31 in descriptive part sentences. It can thus be said that there are more functional words in conversation sentences than in descriptive part sentences.

	Conversation Sentences		Descriptive Part Sentences	
	Number of words	Percentage	Number of words	Percentage
Nouns	416,799	22.35	1,184,428	28.12
Pronouns	70,406	3.78	75,641	1.80
Verbs	266,152	14.27	673,788	16.00
Adjectives	39,639	2.13	75,447	1.79
Adjectival nouns[6]	20,039	1.07	56,702	1.35
Adverbs	55,061	2.95	93,940	2.23
Adnominal adjectives	25,087	1.35	42,893	1.02
Conjunctions	6,263	0.34	13,244	0.31
Interjection	17,503	0.94	2,834	0.07
Particle	629,369	33.75	1,404,343	33.34
Auxiliary verbs	255,154	13.68	483,074	11.47
Prefix	16,846	0.90	16,548	0.39
Suffix	46,599	2.50	89,005	2.11
Total	1,864,917	100.00	4,211,887	100.00

Table 4. Percentages of Parts of Speech in Conversation Sentences and Descriptive Part Sentences

4.4 Differences between Japanese Novels and Translated Novels

Yamazaki (2017a) compared conversation sentences in Japanese novels with those in translated novels. His findings showed that, for example, in Japanese novels, words that are highly characteristic are suffixes and nouns attached to people, such as *san, sama, chan* (all are general suffixes of personal names but differ in politeness). On the other hand, in translated novels, many pronouns and proper nouns are highly characteristic. The greater use of personal pronouns in translated novels than in Japanese novels is thought to be due to the influence of translation from languages that use many personal pronouns, such as English. Among elderly people in Japanese novels, it was pointed out that *role language*[7] such as *washi* (i.e., a first-person pronoun used for elderly people) and *no* (i.e., a final particle used for elderly people) were seen. These examples of role language were also found in translated novels[8] but were used less than in Japanese novels. We hypothesized that since there is no equivalent to role language for elderly people in English, the translators might make greater use of these stereotypical words for describing people in a lively way. We still do not have a clear explanation for this, but we assume that the characters in many Japanese historical novels (i.e., *samurai* novels) would increase the use of such role language.

Table 5 shows how word type usage is distributed by gender. All the words–except those used by characters with unknown genders–were classified into three categories, which are words used only by females (female only), words used only by males (male only), and words used by both (common). From Table 5, we can see that half of the words are used by male characters only in both novels, while the number of words used by female characters only are relatively few. The ratio of common words in translated

[6] "Adjectival nouns" stem from what is known as adjective verbs in school grammar. For instance, *Tokubetsu* (special), *tashika* (sure, certain), *juyo* (important).

[7] Role language is the stereotypical use of words that reminds us of particular people, such as women, children, old people, etc.

[8] For example, *washi* was used in the translations of novels by Charles Dickens, Mark Twain, Maurice Leblanc, Victor Hugo, Jules Verne, Dickson Carr, Agatha Christie, etc.

novels is almost 10 points higher than that of Japanese novels.

	Japanese Novels	Translated Novels
Female only	3,455 (14.1%)	1,670 (9.6%)
Male only	18,054 (54.2%)	6,406 (50.2%)
Common	14,455 (31.6%)	3,736 (40.2%)
Total	11,812 (100%)	35,964 (100%)

Table 5. Distribution of Word Types by Gender

4.5 Comparison with Other Registers

In this section, we compare the characteristics of novel conversation sentences with other spoken language materials. Yamazaki's (2017b) comparison of the Corpus of Spontaneous Japanese (CSJ)'s Academic Presentation Speech (APS) and Simulated Public Speaking (SPS), Corpus of Everyday Japanese Conversation (CEJC), Nagoya University Conversation Corpus (NUC), Women's Words/Men's Words (Workplace Edition) (Workplace), and the BCCWJ's novel conversation sentences (Novels) lead to several conclusions.

Figure 2 shows the ratios of the three distinctive and frequently used parts of speech in spoken words, which are final particles, interjections, and filled pauses. From the ratios of those parts of speech, we can group the corpora into three categories: (1) SPS and APS, (2) NUC, Workplace, and CEJS, and (3) novels. This categorization corresponds to dialog, conversation, and written conversation, respectively.

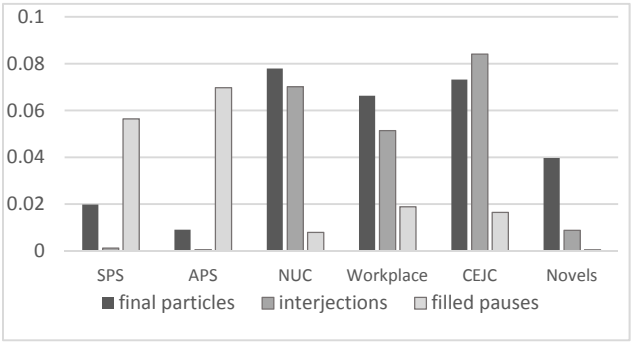


Figure 2. Ratio of final particles, interjections, and filled pauses.

The analysis results of characteristic words based on the Log-Likelihood Ratio are as follows. In simulated public speaking, filled pauses such as *ano*, *maa*, and *nn*, occupied the top positions. However, in academic public speech, even with the same filled pauses, filled pauses expressing mental operations taking time such as *eh* and *eh-to* (Sadanobu & Takubo, 1995) were characteristic words. In the NUC, final particles such as the interjections *uhn*, *neh*, and *sa* appeared in the top position. In workplace conversation, filled pauses did not appear; however, interjections such as *hai*, *ah*, and *eh* occupied the top positions. The greeting "good morning" was also characteristic of workplace conversations. In everyday conversations, words used for responses such as *uhn*, *u-uhn*, and *so* were in the top positions. The novel conversation sentences had completely different aspects, and pronouns such as "you," "I" (male), "I" and "you" (male) were characteristic words. It is also noteworthy that final particles, which represent role language, such as *wa*, *yo*, and *zo* were in the top positions.

5. Future Tasks

We are planning to release the speakers' information (gender and age) in our web concordancer "*Chunagon,*" which is free to applicants in 2019. *Chunagon* is the most frequently used web concordancer in the Japanese corpus. It has over 5,000 users. Other speakers' information will be released separately after several confirmations.

6. Acknowledgements

This work was supported by the NINJAL institute-based project "A Multifaceted Study of Spoken Language Using a Large-Scale Corpus of Everyday Japanese Conversation" (project leader: Hanae Koiso) and by JSPS KAKENHI Grant Number JP12345678.

7. Bibliographical References

Kinsui, S. (2000) Proposal of role language. In Sato, K. (ed) *Kokugo Ronkyu* 8 *New Aspect of Historical Japanese*, Tokyo: Meiji Shoin.

Miyazaki, Y., Kashino, W., and Yamazaki, M. (2017) Fundamental planning of annotation of speaker's information to utterances: Focused on novels in "Balanced Corpus of Contemporary Japanese," *Language Resource Workshop 2016*, Tokyo: National Institute for Japanese Language and Linguistics.

Oishi, H. (1987) Value of conversation sentences in modern novels as data of linguistic research, Kokubungaku Kaishaku to Kansyo, 52(7), pp.72-79.

Sadanobu, T. and Takubo, Y. (1995) The monitoring devices of mental operations in discourse-- a case of "eeto" and "ano(o)"--, *Gengo Kenkyu*, 108, pp.74-93.

Takasaki, M. (1981) On the conversation sentence in novels, Kotoba 2, pp.86-97.

Yamazaki, M. (2017a) Vocabulary diversity in conversation as seen in different corpus registers, *Language Resource Workshop 2017*, Tokyo: National Institute for Japanese Language and Linguistics.

Yamazaki, M. (2017b) Quantitative comparison of Japanese novels and translated novels with respect to lexicology, *Goi Kenkyukai 2017*, Sep. 16 at Aichi Gakuin University Sakae Satellite.

8. Language Resource References

National Institute for Japanese Language and Linguistics (2011) Balanced Corpus of Contemporary Written Japanese (BCCWJ). http://pj.ninjal.ac.jp/corpus_center/bccwj/en/

PDFAnno: a Web-based Linguistic Annotation Tool for PDF Documents

Hiroyuki Shindo[1], Yohei Munesada, Yuji Matsumoto[1]
[1]Graduate School of Information and Science
Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara, 630-0192, Japan
shindo@is.naist.jp, y.munesada@gmail.com, matsu@is.naist.jp

Abstract

We present PDFAnno, a web-based linguistic annotation tool for PDF documents. PDF has become widespread standard for various types of publications, however, current tools for linguistic annotation mostly focus on plain-text documents. PDFAnno offers functions for various types of linguistic annotations directly on PDF, including named entity, dependency relation, and coreference chain. Furthermore, for multi-user support, it allows simultaneous visualization of multi-user's annotations on the single PDF, which is useful for checking inter-annotator agreement and resolving annotation conflicts. PDFAnno is freely available under open-source license at https://github.com/paperai/pdfanno.

Keywords: text annotation, annotation tool, pdf

1. Introduction

Gold standard annotations for texts are a prerequisite for training and evaluation of statistical models in Natural Language Processing (NLP). Since human annotation is known as one of the most costly and time-consuming tasks in NLP, an easy-to-use and easy-to-manage annotation tool is highly required for cost effective development of gold standard data.

Currently, general-purpose linguistic annotation tools such as BRAT (Stenetorp et al., 2012) and WebAnno (Yimam et al., 2013) only support text documents. Some commercial software packages provide annotation functions for PDF, however, they lack a function of *relation* annotation suitable for dependency relation and coreference chain.

Since PDF has become widespread standard for many publications, a linguistic annotation tool for PDF is strongly desired for knowledge extraction from PDF documents. For example, previous work has developed an annotated corpus for coreference resolution on scientific papers (Panot et al., 2014; Schafer et al., 2012; Steven et al., 2008). In their work, PDF articles are converted to plain-text format using OCR software, then import them to a text annotation tool. As pointed out in the literature, OCR errors are present in the data and they need to clean up the text by viewing the associated PDF file. This motivates us to develop a new annotation tool that can directly annotate on PDF.

There are two types of annotation processes for creating an annotated text from a PDF file as shown in Figure 1. One is to convert the PDF into plain text or HTML format, then annotate it using a text annotation tool, as in the previous work. Another one is to annotate the PDF directly, then convert the annotated PDF into plain text format. Even if annotated plain-text is eventually necessary, the latter one has at least two benefits. First, PDF is often much more readable for annotators than plain text since it is well-structured with sections and paragraphs. This helps us maintain annotation quality and consistency. Second, the annotations become insensitive to OCR errors. That means, if high-quality OCR software was developed later, we can switch the OCR software for converting it to plain text without modifying annotations.

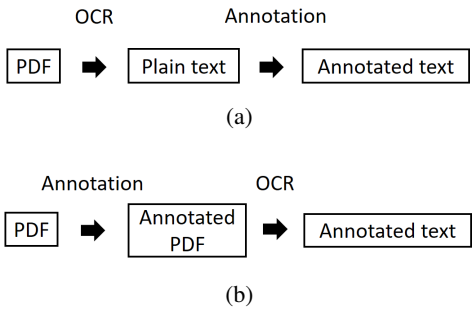


Figure 1: Annotation flows for PDF documents. (a) convert PDF into a plain text, then annotate it. (b) annotate PDF file, then convert it into a text file.

In this work, we present PDFAnno, a general-purpose linguistic annotation tool for PDF documents. PDFAnno offers functions for various types of annotations in a web browser, including named entity, dependency relation, and coreference chain. It requires no installation effort and can be used offline. Furthermore, for multi-user support, it allows simultaneous visualization of multiple annotations on the single PDF, which is useful for checking inter-annotator agreement and resolving annotation conflicts. We also implement a server-side program which converts annotated PDF to XML format by using our PDF parser. The automatic parsed results can be visualized as annotations in the PDFAnno viewer.

We show two case studies of PDFAnno: relation annotation for materials science papers and coreference annotation for ACL anthology papers. In both cases, we observe that the PDF-based annotation has a clear advantage over the text-based annotation in terms of annotation usability.

2. Related Work

In NLP community, a number of annotation tools for text documents have been developed so far (Bontcheva et al., 2010; Muller and Strube, 2006; Stenetorp et al., 2012; Yimam et al., 2013).

BRAT (Stenetorp et al., 2012) is a well-known web-based tool for linguistic annotation and visualization. It is implemented using a client-server architecture in Python and supports rich structured annotation for a variety of NLP tasks. However, it is targeted to annotate text documents.

GATE Teamware (Bontcheva et al., 2010) is a web-based management platform for collaborative text annotation and curation. It is mostly web-based, but the annotation is carried out with the local software.

WebAnno (Yimam et al., 2013) is also a web-based annotation tool which supports a wide range of linguistic annotations. WebAnno has unique characteristics in that it has advanced features for project and user management with monitoring tools. It supports various types of text format including plain text and CoNLL format. However, since WebAnno visualization frontend is built on BRAT, it is also impossible to make annotation directly on PDF.

For PDF annotation, there are many commercial products such as Adobe Acrobat, PDF Annotator[1], and A.nnotate[2], which basically support text highlighting and adding notes and comments on PDF. However, these tools are not intended to be used for linguistic annotation, thus these lack annotation types suitable for linguistic phenomena such as dependency relation and coreference chain. On the other hand, PDFAnno supports such *relation* annotation and multi-user annotation. Furthermore, it is open-source and extensible with annotation API for external programs.

3. Features

3.1. User Interface

Figure 2 shows a screenshot of PDFAnno user-interface. PDFAnno is a browser-based application and built entirely using standard web technologies. For rendering a PDF document, we use PDF.js[3], a web-based PDF viewer built with HTML5. PDF.js is a default built-in PDF viewer in Firefox, thus it offers a familiar environment to annotators for PDF operations such as zoom, search, and print.

We implemented annotation layers on PDF.js with JavaScript. Currently, PDFAnno supports three types of annotations: span, rectangle, and relation. The use cases of these annotations are shown later.

Span Span is the most basic type of annotation to mark text spans in PDF. For each span, users can assign a text label. For part-of-speech annotation, annotators mark a text span by selecting it with the mouse dragging and assign a part-of-speech tag. Similarly, the span annotation can be used for named entities. PDFAnno enables autocompletion for text label fields, thus annotators can fill in long words by typing only a few characters. In the implementation, the span is preserved as the position: (x, y, width, height) and the page number where x and y describe the coordinates of the top left point of the span, and width and height describe its dimensions.

Rectangle Rectangle is a type of annotation to select a region in PDF. This is intended to be used for annotation of non-text objects such as tables and figures. This is not directly related with text annotations, however, we provide the rectangle function for creating training data for region detection of figures and tables, which is useful for knowledge extraction from scientific papers.

Relation Relation is a type of annotation to make a connection between annotated objects. PDFAnno provides three kinds of binary relations: one-way, two-way and undirected arrows. The one-way arrow can be used for annotation of word and named entity dependencies, the two-way arrow used for bidirectional relation between objects, and the undirected arrow used for coreference chain and grouping multiple annotations. As in the case of span annotation, users can assign a text label to each relation.

In PDFAnno, an identifier (ID) is assigned to each annotation object. The relation is preserved as a pair of annotation IDs and its direction.

3.2. System Architecture

The overall system architecture of PDFAnno is shown in Figure 3. PDFAnno is a simple client-side application in a web browser. It loads PDF.js for rendering the user-specified PDF, then provides functions for adding annotation layers on PDF.js. For multi-user annotation, we assume to use an online storage service to share PDF documents and annotation files between annotators. Every user who has a permission to access the common online storage can load the shared annotation files with PDFAnno.

Our system architecture contrasts with that of BRAT and WebAnno in that most annotation functions and settings in PDFAnno can be accessed and controlled on a client-side. In BRAT and WebAnno, the server fully manages datasets and user account settings. However, we believe that an online storage service can be substituted for most of such server-side functions.

3.3. PDF to XML Converter

While the annotation functions in PDFAnno require no communication with the server, we optionally provide server-side programs for parsing and converting the annotated PDFs into XML. The server-side programs first extract text and positional information from the PDF with Apache PDFBox[4], then convert it to XML format with the user's annotation information. Currently the PDF to XML conversion is performed with our rule-based method, however, we plan to replace it with machine learning approach to reduce the conversion errors.

3.4. Annotation File

In PDFAnno, user's annotation is preserved separately from the original PDF file, and downloadable anytime as a text file following TOML format[5]. Compared with JSON and YAML format, TOML is easy to read and easy to write by

[1] https://www.pdfannotator.com/

[2] http://a.nnotate.com/

[3] https://mozilla.github.io/pdf.js/

[4] https://pdfbox.apache.org/

[5] https://github.com/toml-lang/toml

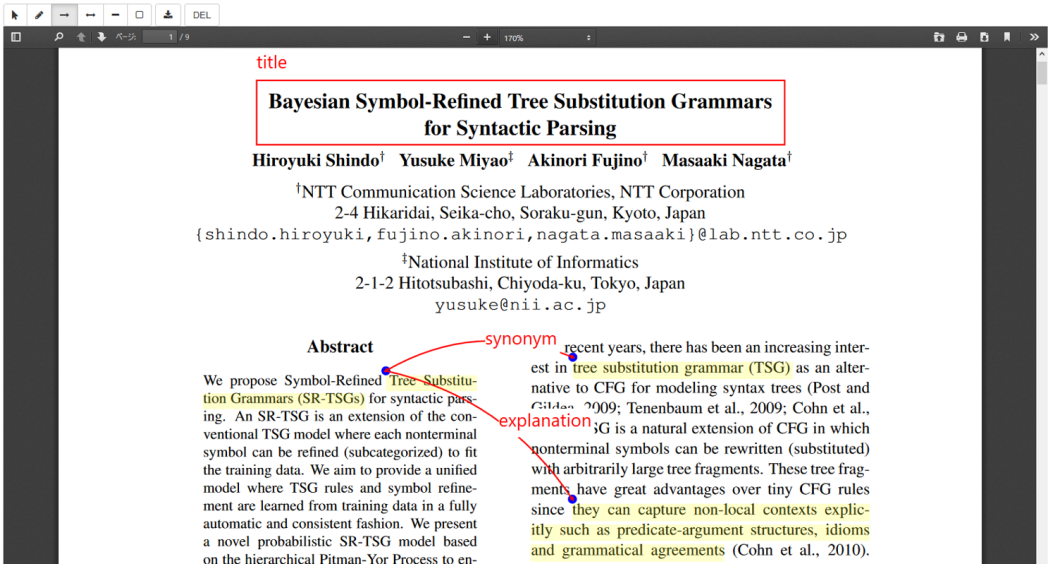


Figure 2: Screenshot of the PDFAnno user-interface, showing example annotations of text span, rectangle, and relation.

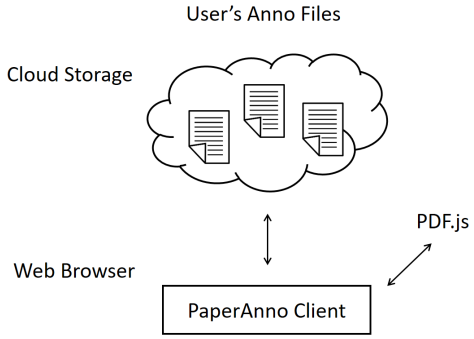


Figure 3: System architecture of PDFAnno.

```
[1]
type = "span"
page = 1
position = [[95.818,252.977,181.761,10.909]]
label = "label-1"

[2]
type = "span"
page = 2
position = [[323.863,213.988,230.715,11.590]]
label = "label-2"
```

Figure 4: Example annotation file for PDFAnno.

both human and computers. Figure 4 shows an example annotation file (anno file) with two spans, one rectangle and one relation. Span is represented as a page number, positions, and its label, while relation contains a page number, connection type, two identified spans and its label.

3.5. Support for Multi-User Annotation

For multi-user annotation, PDFAnno user-inferface accepts to load multiple annotation files and renders these annotations on the single PDF with distinct colors one another. Figure 5 shows an example of rendering multiple annotations. A user can perform annotation work while referring to other annotation file, which helps users check inter-annotator agreement and resolve annotation conflicts.

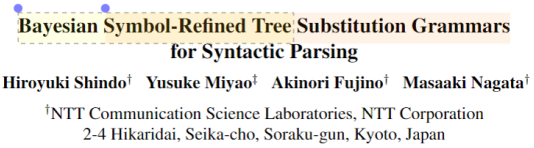


Figure 5: Example of rendering multiple annotations on a single PDF. In the example, one annotator marked "Bayesian Symbol-Refined Tree" as a span but another annotator marked "Symbol-Refined Tree Substitution Grammars".

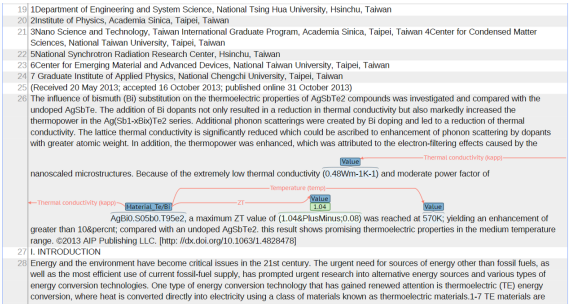
4. Case Studies

4.1. Information Extraction from Scientific Papers

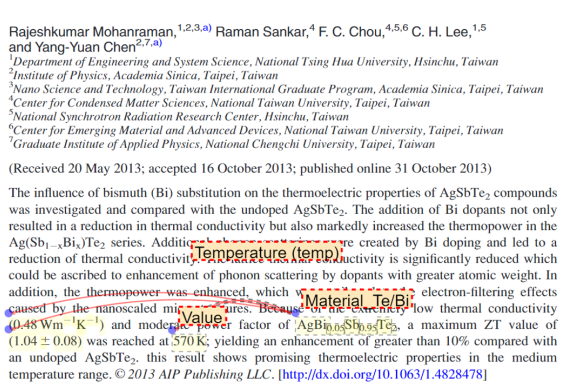
PDFAnno is well-suited for creating gold annotation data for information extraction (IE) from scientific papers. To test the effectiveness of PDFAnno, we conducted an annotation experiment for IE from scientific papers. In particular, we asked experts in materials science to annotate selected papers in their field with material names and their physical quantities such as temperature and thermal conductivity. In our annotation guideline, material names and their physical quantities are marked as *span*, and these are connected with each other as *relation*.

We provided the selected papers as both plain-text format and PDF to the annotators, and compared text-based and PDF-based annotation with respect to annotation time and quality. For text-based annotation, the plain texts were extracted from PDFs with Poppler[6] and annotation work was performed using WebAnno (Yimam et al., 2013) . For PDF-based annotation, we provided our PDFAnno to the annotators. Figure 6 shows the screenshots of WebAnno and PDFAnno for scientific papers.

[6]https://poppler.freedesktop.org/



(a) Text-based annotation using WebAnno.



(b) PDF-based annotation using PDFAnno.

Figure 6: Comparison of text-based and PDF-based annotation on materials science paper.

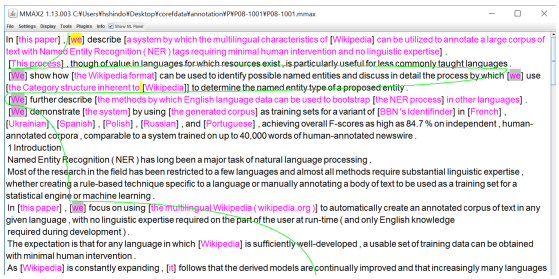
As we expected, we observed that the annotators completed the annotation work much more rapidly in PDF-based annotation compared with text-based one. The main reason is that the structure of section and paragraph of papers is lost in plain-text format, which makes annotators read and understand the paper much more difficult than the original PDF. Actually, every annotator performed the work of text-based annotation while referring to the original PDF. On the other hand, such problems did not occur in the PDF-based annotation.

Although further research and evaluation is required to establish the benefits of PDFAnno, we view that the direct annotation on PDF with PDFAnno is a promising direction for annotating scientific papers. The PDFs and annotation files in this experiment are freely available at PDFAnno website.

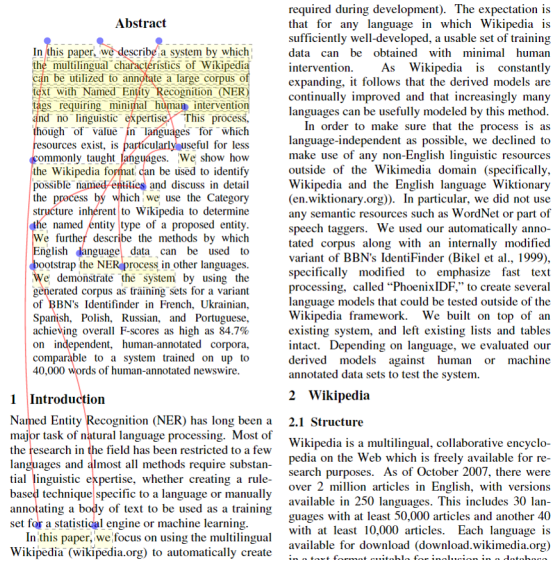
4.2. Coreference Annotation

To further test the effectiveness of PDFAnno, we annotated selected papers from ACL anthology with coreference chains. Since previous work (Schafer et al., 2012) on coreference annotation for ACL papers has released gold annotated data as MMax2 format, we actually transferred the gold annotations to PDF using PDFAnno. In the previous work (Schafer et al., 2012), the texts are extracted from PDF using a commercial OCR software, then coreference chains are annotated on the texts using MMax2 annotation tool (Muller and Strube, 2006). Figure 7 shows an example of coreference annotation on P08-1001 paper using MMax2 and PDFAnno.

The main problem of text-based annotation is that there exists multiple versions of text corpus extracted from PDFs.



(a) Text-based coreference annotation using MMax2.



(b) PDF-based coreference annotation using PDFAnno.

Figure 7: Examples of coreference annotations on plain-text and PDF for P08-1001 paper.

For example, two versions of ACL anthology reference corpus have been released (Steven et al., 2008). In addition, the previous work uses their own OCR software for text extraction from PDF since the ACL reference corpus converted with PDFBox contained many extraction errors. Therefore, it is difficult to establish consistency with multiple annotations on ACL papers due to the differences between text extraction tools. As in the case of materials science papers, PDF is more readable and easy to annotate with coreference chain. We have noticed that the main drawback of PDF annotation is that the line space is not adjustable, which makes annotation work difficult when there are too many annotation objects on PDF. We will leave the problem of rendering many annotation objects for future work.

5. Conclusions

We present PDFAnno, a web-based linguistic annotation tool for PDF documents. PDFAnno provides functions for a variety of linguistic annotation for PDF documents. It is a simple and easy-to-use client-side browser application. Furthermore, it allows simultaneous visualization of multi-user's annotations on the single PDF, which is useful for checking inter-annotator agreement and resolving annotation conflicts. Future work involves adding more advanced project management system and XML/HTML support.

6. Acknowledgements

This work was partly supported by JSPS KAKENHI Grant Number 15K16053 and JST CREST Grant Number JPMJCR1513, Japan..

7. Bibliographical References

Bontcheva, K., Cunningham, H., Roberts, I., and Tablan, V. (2010). Web-based collaborative corpus annotation: Requirements and a framework implementation. In *New Challenges for NLP Frameworks workshop at LREC*, pages 197–214.

Muller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. In *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*.

Panot, C., Akiko, A., and Yuka, T. (2014). Corpus for Coreference Resolution on Scientific Papers. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 26–31.

Schafer, U., Spurk, C., and Steffen, J. (2012). A Fully Coreference-annotated Corpus of Scholarly Papers from the ACL Anthology. In *Proceedings of the International Conference on Computational Linguistics (COLING) 2012*, pages 1059–1070.

Stenetorp, P., Pyysalo, S., Topic, G., Ohta, Tomoko Ananiadou, S., and Tsujii, J. (2012). brat: a Web-based Tool for NLP-Assisted Text Annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 102–107.

Steven, B., Robert, D., Bonnie, D., Bryan, G., Mark, J., Min-Yen, K., Dongwon, L., Brett, P., Dragomir, R., and Yee Fan, T. (2008). The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, pages 1755–1759.

Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL)*, pages 1–6.

A Lightweight Modeling Middleware for Corpus Processing

Markus Gärtner Jonas Kuhn
Institute for Natural Language Processing
University of Stuttgart
{markus.gaertner,jonas.kuhn}@uni-stuttgart.de

Abstract

Present-day empirical research in computational or theoretical linguistics has at its disposal an enormous wealth in the form of richly annotated and diverse corpus resources. Especially the points of contact between modalities are areas of exciting new research. However, progress in those areas in particular suffers from poor coverage in terms of visualization or query systems. Many limitations for such tools stem from the non-uniform representations of very diverse resources and the lack of standards that address this problem from the perspective of processing or querying. In this paper we present our framework for modeling arbitrary multi-modal corpus resources in a unified form for processing tools. It serves as a middleware system and combines the expressiveness of general graph-based models with a rich metadata schema to preserve linguistic specificity. By separating data structures and their linguistic interpretations, it assists tools on top of it so that they can in turn allow their users to more efficiently exploit corpus resources.

Keywords: data model, multi-modal, metadata

1. Introduction

The availability of richly annotated multi-layer corpora is one of the cornerstones of modern theoretical and computational linguistics. With a steady flow of newly created corpus resources and tools to annotate or process those corpora comes also the need for exploratory support tools, especially visualization and query systems, to get a better understanding of the resources at hand.

The context of this contribution is the successor project of the ICARUS (Gärtner et al., 2013) platform, an interactive visualization and query tool for corpora with dependency syntax, coreference structures (Gärtner et al., 2014) or rich annotations for intonation (Gärtner et al., 2015). This project aims at providing researchers a platform to explore and query even more diverse multi-layer and multi-modal corpus resources and artifacts through a single interactive interface. Our target resources include, but are not limited to, text and speech data annotated for a variety of different layers on varying linguistic levels and granularities, such as the SFB732 Silver Standard Collection, of which a first overview has been provided by Eckart and Gärtner (2016). A strong focus lies hereby on connecting linguistic layers or modalities which so far have not received much attention in their particular combination. This is to support interesting research questions, such as the interaction of information status and prosodic realization(Baumann and Riester, 2013) or the incorporation of prosodic information into the task of automatic coreference resolution on spoken text as done by Roesiger and Riester (2015).

As a result we faced the challenge of modeling access from a single processing software to a very diverse set of corpus resources. And while several standards around the *Linguistic Annotation Framework* (LAF)(Ide and Suderman, 2014) exist, they primarily take the point of view of preparation or curation of data. The ongoing standardization effort for a *Corpus Query Lingua Franca* (CQLF, ISO/DIS 24623-1)(Banski et al., 2016) correctly identifies the lack of standards which address a quite different view, that is, the one of querying or processing of corpus resources. As such there is no standardized or universally accepted solution readily available for this modeling task, which led us to designing and implementing a new dedicated framework.

With "modeling corpus resources" we refer to the task of representing the structure and content of a corpus or similar resource by means of a data model in memory, i.e. during an application's runtime. As a corpus or equivalent resource we treat any collection of utterances in arbitrary form, be they written, spoken or presented in yet another modality. Note that presently this definition excludes the modeling of other types of linguistic resources like lexicons or dictionaries as realized for example in the *Lexical Markup Framework* (Francopoulo et al., 2006).

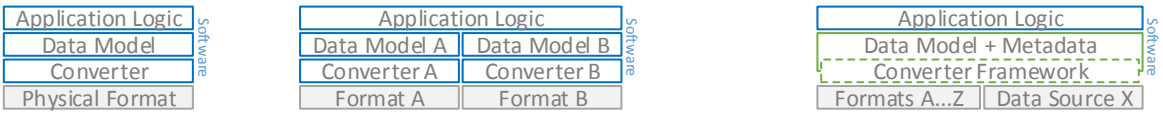
In the remainder of this paper we present our approach for providing unified access to very different corpus resources. Section 2 discusses the underlying issue of a heterogeneous format multiverse and Section 3 contextualizes our work. We present a detailed overview of various aspects of our framework in Section 4 and then proceed to illustrate its usage with code examples in Section 5. Information on the availability of the software and its documentation is provided in Section 6 and Section 7 concludes.

2. Format Multiverse and Middleware

Extensive work has already been invested into designing and establishing interoperable formats for linguistic resources and annotations based on various common transport formats. Notable examples are the *NLP Interchange Format* (NIF) (Hellmann et al., 2013), an RDF/OWL-based format, or *GrAF* (Ide and Suderman, 2007), a pivot XML-based serialization format for the *Linguistic annotation framework (LAF)* (Ide and Suderman, 2014). *GrAF* has also been standardized by ISO[1] as part of the "Language resource management" committee (ISO/TC 37/SC 4). The LAPPS Interchange Format (Verhagen et al., 2016) is a JSON-LD[2] format and used for data transfer between service implementations in the The Language Applications

[1] ISO 24612:2012

[2] https://json-ld.org/



(a) Monolithic design (b) Multiple supported format stacks (c) Middleware solution

Figure 1: Bottom-up data flows in several resource processing applications depicted as software stacks. Grey filled boxes represent data to be processed, blue bordered ones are parts of an application's own codebase and green shows the placement of our middleware solution in such an architecture in (c).

Grid (Ide et al., 2016). Less flexible exchange formats include for instance the *Text Corpus Format* (TCF) of the WebLicht (Hinrichs et al., 2010) environment, where the coverage of different annotation layers or hierarchical structures remains fixed.

Albeit having very expressive and standardized formats available, the reality is that more often than not corpora or system outputs used in day-to-day research work are stored in specialized or proprietary formats. Especially tools used for a specific task or field of research often use formats that have emerged as a kind of "local" standard for the particular field, such as Praat TextGrid files (Boersma, 2001) for phonetic analysis. Tools designed in the context of a particular task not uncommonly motivate the design and limitations of their data model with the original data format provided. In the popular CoNLL Shared Tasks for example data sets consistently are made available in a tabular format tailored to the individual tasks (e.g. parsing 2009 (Hajič et al., 2009), coreference resolution 2011 (Pradhan et al., 2011) and again parsing to Universal Dependencies in the 2017 shared task[3]). Tools developed for those tasks often made the respective CoNLL format their exclusive input and output representation, leading to a plethora of classic monolithic designs of the form shown in Figure 1a.

Processing applications such as search or visualization tools are then required to either support multiple formats and associated abstract models (see Figure 1b) or force the user into converting input data into a pivot format (further encouraging designs as in Figure 1a). This leads to increased workload for either the developers or end users. An alternative solution is to shift the unification effort into a framework that acts as a middleware between data sources and the actual processing logic of an application. The placement of our framework as this kind of middleware solution, in comparison to above situations, is presented in Figure 1c. Using a middleware solution also enables applications built on it to have their output freely converted into all formats supported by the framework. We believe that removing the need or motivation to build monolithic application architectures then in turn can even foster the use of standardized serialization formats like *GrAF*.

3. Related Work

At present several software toolkits for modeling corpus resources in a flexible way exist. Graph-based models have become a very popular approach for their implementation. *SALT* (Zipser, 2009), a graph-based model toolkit, is used in an established ecosystem together with the converter-framework *PEPPER* (Zipser et al., 2011) and the search interface *ANNIS3* (Krause and Zeldes, 2014), the latter combining it with a relational database for storage. *SALT* provides full bidirectional links between related nodes (such as a sentence and its tokens) and also imposes fixed restrictions on the scope of relations. It is well-suited for modeling shallow snapshots of a corpus, but especially the bidirectional linking makes it inherently expensive for modeling large numbers of concurrent structures (e.g. multiple parses for the same sentence) and hampers scalability. Motivated by the need to model rich annotations for text and speech data, the *NITE Object Model* (Carletta et al., 2003) is another graph-based model that allows linking of elements to portions of a singular shared timeline. Yet another approach is to use formalisms from the Semantic Web context like the Web Ontology Language (OWL[4]) and combine them with existing storage technology like XML or a relational database as proposed by Burchardt et al. (2008).

Based on the *Apache UIMA™project* a series of software components for natural language processing around the *DKPro Core* (Eckart de Castilho and Gurevych, 2014) framework have been developed. They are mainly focused on building shareable analysis pipelines and build on a strongly typed model, where linguistic types or theories are directly encoded. This can be helpful for data exchange between collections of predefined processing components such as parsers. However, it made this approach unsuited for us as alternative to designing our own modeling toolkit. Other text processing pipelines include *OpenNLP*[5], the *General Architecture for Text Engineering* (Cunningham et al., 2002) and the *Natural Language Toolkit* (Bird, 2006) for Python. They provide solutions for various NLP tasks and also feature modeling approaches of varying expressiveness. Their general focus on text data however keeps them limited and unfit as basis for a middleware system that attempts to unify access to multiple modalities.

4. Architecture

An implementation of our modeling approach is provided as an open-source Java toolkit (further information on availability is provided in Section 6) which requires version 8 of the Java Runtime Environment (JRE). It offers a set of abstract *building blocks* and other components to compose the data structures that make up a given corpus resource once read into the model and an extensive API to interact with them in various ways.

[3] http://universaldependencies.org/conll17

[4] www.w3.org/2004/OWL/

[5] http://opennlp.apache.org

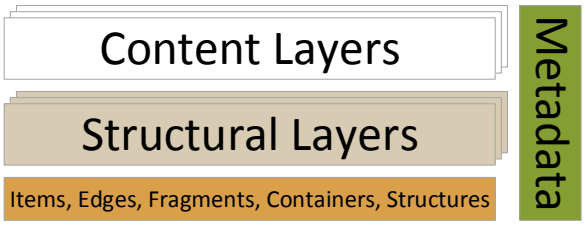


Figure 2: Core components for storage and associated metadata of the model architecture. Storage components are aligned horizontally.

A core concept is to separate components that represent addressable units (tokens, sentences, audio frames, syntactic structures, edges, regions of an image or video stream.) from their actual content (such as classic text annotations or binary data of audio or video streams). Figure 2 shows the hierarchy of core components in the model with parts responsible for storage aligned horizontally. The following sections line out the conceptual foundations of our approach and then describe the core components and concepts in detail.

4.1. Separation of Model Responsibilities

While the approaches described in Section 3 and many others are sufficient for the task of representing data in memory, the gain in representational generality is often achieved at the cost of specificity when it comes to the linguistic meaning or interpretation of modeled units or annotations. Thus even when using them as a developer one still needs to model this information and connect it with the data structures in such models.

To work around these shortcomings we decided to split the task of modeling corpus resources into two subparts:

One part for segmentation, structure and data storage maximizes generality but sacrifices any direct knowledge about linguistic specificity (equivalent to aforementioned general-purpose graph models). Independent of this the other part acts as metadata and carries the information describing composition and dependencies of a specific resource.

We outline the two aforementioned parts and how they interact with each other in Sections 4.3 to 4.5.

4.2. Design Foundations

Our model design builds on several observations and requirements for modeling diverse corpus resources, some of which have been proposed by Ide et al. (2003) in an early iteration of LAF. We list these requirements and show how our modeling approach fulfills them:

Expressive adequacy and Openness: We do not impose any limitations on the nature of linguistic information or theories that are expressed or stored in our framework to keep it as flexible as possible.

Uniformity: We utilize a compact set of universal "building blocks" (see Section 4.3) to model all kinds of structural or hierarchical information in a corpus (equal in expressiveness to a graph, but closer to linguistic universals).

Media independence: As stated in the introductory part of Section 4 our framework conceptually splits structural properties of a resource from its (media) content, treating all content as annotations. It thereby allows for arbitrary annotations to be associated with individual building blocks.

Semantic adequacy: We utilize a compact metadata schema (see Section 4.5) to formally define a resource's overall composition. This includes declarations of available layers as well as their hierarchies and dependencies.

Granularity: For every resource there is at least one mandatory segmentation layer that defines atomic units as a common foundation. This guarantees spacial comparability of all other structural constructs on top of it. As an extension to the original observation made by Ide et al. (2003) we also acknowledge the need to retrospectively define an even finer granularity as originally declared for a resource. Our framework supports subdivision of existing units in a corpus by declaring anchors into rastered[6] annotation values for those blocks (see Section 4.3 for further explanation and Section 5.3 for a usage example).

Extensibility: We offer various mechanisms for extending the functionality of the corpus modeling framework:

- A template system for the metadata schema to for example allow sharable tagset definitions or format schemata for converters (see Figure 4 for a tagset template).
- Integration of new converter implementations via the Java Plugin Framework[7], Java's own Service Provider Interface (SPI) or the Open Services Gateway initiative[8].
- An abstract type system for building blocks that allows custom implementations to optimize based on the underlying corpus resource, for instance read-only resources or when accessing a corpus stored in a (relational) database versus converting from classic file data.
- Interfaces to implement the handling of new media or annotation types for tasks such as fragmentation (cf. Section 4.3) or spacial comparison of those fragments.

Scalability: Special care is taken to ensure the framework and its performance are able to scale along two different and very important axes:

- *Horizontally* with the size of resources, i.e. the number of addressable units they contain (which can reach up to billions of tokens for web corpora like those from the *WaCky* family (Baroni et al., 2009)).
- *Vertically* in terms of the number of (parallel) annotation layers of arbitrary type. Especially speech corpora can contain very rich phonetic annotations on various levels when annotated by automatic systems.

4.3. Components

Instead of squeezing an entire corpus into one big graph we provide a series of types and constructs that model aspects of a corpus in ways closer to actual linguistic concepts.

[6] We borrow the term to describe the transformation of raw annotation values into collections of addressable discrete units, such as sequences or grids. Common examples are individual characters in a text, pixels in an image or frames in an audio stream.

[7] `http://jpf.sourceforge.net/`

[8] OSGi™ `https://www.osgi.org/`

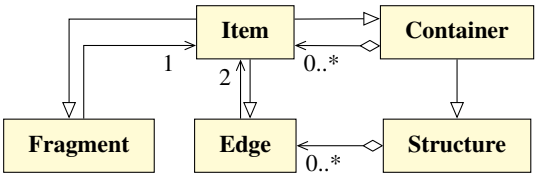


Figure 3: Simplified UML class diagram showing the relations between basic types described in Section 4.3.

Units: At the core we provide generic building blocks to model addressable units in a corpus (see Figure 3 for a basic UML class diagram of those types and their relations and Table 1 for additional explanations). Those `Items` can represent any individual data points, regions, edges or aggregations within a corpus. In addition to construct a corpus by aggregation, `Fragments` offer a way of subdividing existing items into smaller addressable units.
Each item in a corpus can be identified by two ways: One is a numerical *id* that is unique within its host layer and the other is its current *position* in that layer. While the id is persistent, the current index can change for editable corpus data and is mainly used for spatial comparisons or mappings. Unlike other approaches we do not maintain a full bidirectional linking between related items outside of their native host container. Instead a container only holds links to its contained items, but the reverse linking is moved into dedicated mapping facilities on the layer level. This keeps linking overhead to a minimum and makes the model scalable wrt concurrent segmental or structural information.

Layer: Layers in our model directly correspond to their linguistic counterparts and organize parts of a corpus such as annotations, segmentations, hierarchies or other types of structures. Layers carrying segmental or grouping information play a special role since they can act as foundation or boundary layers. The former provide a shared space for spatial comparisons equivalent to timelines in other systems and the latter are used for restricting the scope of complex constructs such as relations. To link different aggregating or foundation layers dedicated mapping facilities are used to translate between those index spaces.

Context: Layers are naturally grouped according to the data source their content originated from, for example a database, web-service or a local file. On the context level dependencies to external corpus resources are defined and each context must have a singular layer designated for the roles of *foundation* and *primary*, to define the segmentation of traversable data chunks of the context and their shared address space. Each context is associated with its individual converter implementation that mediates between the physical source of data and the in-memory model instances as described in more detail in Section 4.7.

Corpus: On the top of the hierarchy a corpus combines an arbitrary number of contexts, which enables modeling of resources that consist of multiple data sources or are expressed in several different formats. Each corpus in the framework is an independent and interactive object that client code can work with in many different ways, some of which are described and compared in Section 4.6.

Type	Function
Item	basic addressable unit
Edge	link/relation between items
Fragment	anchors used to split existing items based on their rastered annotation values
Container	logical grouping of items
Structure	container that augments its elements (= nodes) with edges

Table 1: Basic types used to model addressable units, segmentation and structure in a corpus resource and their respective functions. Aggregating types are italicized.

4.4. Annotations

A very central requirement for successfully unifying richly annotated corpus resources across different modalities is the ability to deal with very diverse types of annotations. To achieve this we impose no direct limitations on the nature of annotation values that can be stored or retrieved through the model. In addition we defined a (extensible) collection of commonly used annotation types that are directly supported by specialized storage implementations. The most important ones are listed as follows:

- Character sequences (`Strings`). With Java natively modeling characters as UTF-16 code units in memory, this directly provides sufficient Unicode support.
- Numerical values (`integer`, `long`, `float`, `double`)
- Links to external resources such as files or websites (`URL`, `URI` and local file paths).
- Images modeled as bitmap graphics.
- Binary streams such as audio or video content. Our framework does not restrict or support specific multimedia formats. It is left to the producer of a resource description (cf. Section 4.5) to include sufficient information about the encoding, preferably in the form of standardized format identifiers so that Java's default frameworks for handling media content can be used.
- References to other `Items` to model simple structures such as arguments.
- Vector and matrix types to allow aggregation of other supported value types.

4.5. Metadata

In Section 4.3 we covered the design of a very flexible data model for representing the content of arbitrary corpus resources. This maximized generality reduces the typical overhead of developing very content specific software components (e.g. a custom-built search or visualization interface tailored to the content of a single corpus).
Sacrificing specificity, if not compensated somehow, can on the other hand create certain disadvantages, such as not knowing how to visualize a certain piece of data expressed in the model. This is especially true for interactive systems that wish to provide assistive functions or information to the end-user. Such assistive functions are for example optimized visualization of linguistic data or automatically limiting user input for search constraints based on the con-

```
1  <annotation id="common.tags.stts" name="STTS−
   Tagset", valueType="string">
2   <values>
3    <value name="Adjective">JJ</value>
4    <value name="Noun">NN</value>
5    <value name="Determiner">DT</value>
6    <value name="Verb, gerund">VBG</value>
7    <value name="Verb, 3rd sg">VBZ</value>
8    [...]
9   </values>
10 </annotation>
```

Figure 4: Part of a manifest in XML format describing a part of the STTS tagset for part-of-speech annotations as a reusable template. See Figure 5 for an example of another metadata component referencing this template.

```
1  <corpus id="my.simple.corpus">
2   <context foundation="token">
3    <layerGroup primary="token">
4     <itemLayer id="token"/>
5     <itemLayer id="sentence">
6      <baseLayer layerId="token"/>
7      <container containerType="span"/>
8     </itemLayer>
9     <annotationLayer id="content">
10     <baseLayer layerId="token"/>
11     <annotation key="id" valueType="int"/>
12     <annotation key="form"/>
13     <annotation key="lemma"/>
14     <annotation key="pos" templateId="common.tags.
   stts">
15      <!-- Inherited from template -->
16     </annotation>
17    </annotationLayer>
18   </layerGroup>
19  </context>
20 </corpus>
```

Figure 5: Manifest XML specifying the structure of a simple two-level corpus resource with some shallow annotations. Note how the declaration for part-of-speech annotations in line 14 references a template previously defined in Figure 4. A sentence matching this structural description can be found in Figure 6 encoded in two different formats.

tent of a tagset. If only provided a general graph-like representation of a corpus without any additional information, a processing software component would have to undergo the time-consuming task of analyzing the resource first.

To this end we designed a compact metadata schema and its associated serialization format in XML that can express information regarding a corpus resource's structure and content. For every complex type (`Container` or `Structure`) and each layer used in the storage model this metadata model provides dedicated descriptions called **manifests** ("simple" types are described as part of their surrounding host construct). These descriptions include dependencies between layers, what annotations are available for which items, as well as types of structures or containers and more. For improved interoperability most metadata elements can be linked to established identifiers of linguistic categories to better express their meaning, for example using ISOcat/DatCatInfo as described in (Windhouwer and Wright, 2012). Conventional metadata to describe a resource's provenance is however outside the scope of this scheme.

Using such a manifest enables tools to properly handle the general data structures provided by the storage model totally independent from the original format or data source of a corpus. Figure 6 shows instances of two different textual formats that both encode the same chunk of data matching the description in Figure 5. For an application the resulting representation in our data model looks exactly the same. It could also contain information on how to access the resource it is describing, but this is currently outside our intended scope for the metadata model. Note that it remains the responsibility of the producer of a resource to provide this metadata and also to some extent to implement means of converting the raw form of a resource into an actual model representation (more on this in Section 4.7).

4.6. Interaction

Modeling frameworks often confine themselves to providing generic data structures and support for their serialization. This keeps them usually easy to use, but leaves the bulk of additional management to client code. Different types of applications pose very diverse requirements to underlying model libraries. To accommodate a wide range of possible use-cases our framework features different operation modes for interacting with a corpus:

Navigation: Depending on their individual needs, applications can access data in a corpus ***directly*** (communicating with the converter level without intermediate abstraction layers), as a ***stream*** (forward-only iteration) or in a "window" mode through so called ***views***. The latter approach allows for horizontal and vertical filtering (for instance to limit elements to candidates provided by some index system and to reduce the amount of required annotation layers) and also supports paging to traverse the selected sub-corpus block-wise.

Assistance: The components described in Section 4.3 can be used with a varying degree of framework support, for instance ***raw*** as generic building blocks if no management support by the framework is desired or ***fully managed*** when accessing the corpus content through a *view* or *stream*. Assistance includes for example structural or annotation verification based on the associated metadata definitions.

Editability: Resources are declared static or editable in their metadata and client code can additionally specify if it needs write access when connecting to a corpus. The framework is then able to optimize actual model instances for performance based on those decisions. When in *write* mode, the framework also offers an integrated edit history for live data to record (or undo) any modifications and a notification system to inform client code of changes.

4.7. Conversion

Of course the transformation of raw data into an in-memory model still requires a piece of software dedicated to that

```
1  Finding   find    VBG
2  the       the     DT
3  right     right   JJ
4  format    format  NN
5  is        be      VBZ
6  tricky    tricky  JJ
7  .         .       .
```

```
<s id="s1">
 <graph>
  <terminals>
   <t id="s1_1" word="Finding" lemma="find" pos="VBG"/>
   <t id="s1_2" word="the" lemma="the" pos="DT"/>
   <t id="s1_3" word="right" lemma="right" pos="JJ"/>
   <t id="s1_4" word="format" lemma="format" pos="NN"/>
   <t id="s1_5" word="is" lemma="be" pos="VBZ"/>
   <t id="s1_6" word="tricky" lemma="tricky" pos="JJ"/>
   <t id="s1_7" word="." lemma="." pos="."/>
  </terminals>
 </graph>
</s>
```

Figure 6: Automatically processed sentence in both a tabular (left text) format similar to the ones used in various CoNLL Shared Tasks, e.g. (Hajič et al., 2009), and TigerXML (König et al., 2003) on the right. Both representations contain the same annotations and hierarchical structure expressed by the manifest description in Figure 5.

particular representation or data source. This is an issue which cannot be solved universally due to the fundamental differences in the ways data is stored in different systems or file types (such as database system versus plain text or audio files). While ideally the existence of serialization formats for linguistic annotations such as *GrAF* (cf. Section 2) would obsolete many less expressive or flexible approaches, in reality pragmatic considerations of sticking to alternative formats often prevail. The matter is further complicated when taking different use-cases into account where corpora of vastly different sizes are involved and where aspects like pure storage-efficiency takes priority over those of expressiveness or flexibility of the format being used. As such the necessity of decoupling in-memory modeling completely from the original representations still remains.

In our framework this task of *physically* connecting to a given corpus resource (i.e. converting between its native form and the respective in-model instances) is performed by implementations of the `Driver` interface. Drivers allow the core framework to completely abstract away from the specifics of individual data sources. From the framework's point of view the complexity is reduced to querying the driver implementations for information such as the size of a layer and to issue read or write operations for selected parts of a resource (which in turn causes the driver to perform the necessary conversions).

While the initially stated complexity to some extent still remains, it is possible to implement this "converter" part of the application in such a way that the specifics of a physical representation can be defined by means of a schema which allows converters to adapt to input that is different but of a similar *type*[9] and greatly reduces required development effort. We cover a series of common formats and sources natively and provide generic converter solutions which can be configured using such schemata (either as part of a manifest, via external files or programmatically).

[9]For example the support for schema definitions to read arbitrary tabular text data in the ICARUS (Gärtner et al., 2015) exploration and query tool

5. Usage Examples

This section illustrates how client code uses the concepts introduced in previous sections and especially the different interaction modes outlined in Section 4.6. Where applicable, example code will assume the simple corpus structure shown in Figure 5 and the one-sentence content in Figure 6.

5.1. Building a Corpus

For low-level tasks, for instance a driver implementation, direct constructive interaction with the basic framework members is required. The following code snippet illustrates the construction of the sentence shown in Figure 6:

```
// Binding to manifests/storage
ContainerManifest sentenceManifest = ...;
Container sentenceRootContainer = ...;
AnnotationLayer contentLayer = ...;
// Simplify creation of building blocks via factory
LayerMemberFactory factory = newMemberFactory();
// Instantiate sentence container
Container sentence = factory.newContainer(
  sentenceManifest, sentenceRootContainer, 1);
// Create, add and annotate tokens
Item token1 = factory.newItem(sentence,1);
sentence.addItem(token1);
contentLayer.setValue(token1, "form", "Finding");

Item token2 = factory.newItem(sentence,2);
sentence.addItem(token2);
contentLayer.setValue(token2, "form", "the");

// Finally add finished sentence
sentenceRootContainer.addItem(sentence);
```

The process involves manually constructing the individual tokens, grouping them into a sentence and attaching annotations. It also assumes an existing host corpus and access to certain parts of the associated manifest and layers to store new content in. For brevity only the first two tokens are constructed and other annotation layers besides the `form` layer are omitted.

5.2. Accessing Corpus Parts

As mentioned previously in Section 4.6 there are several ways for accessing (parts of) a corpus resource through the framework. The following two code examples show very different approaches, one involving horizontal filtering and one for simple sequential streaming.
In the first snippet client code is only interested in a selected subset of tokens. It qualifies the actual token indexes by

means of `IndexSet` instances and the layers it requires by a `Scope` in lines 3 and 5. With those parameters the corpus then creates a `CorpusView` object which essentially acts as a filtered sub-corpus. Iterating over the tokens contained in this view and outputting their respective `form` annotations then results in "the format is tricky" to be printed one token per line:

```
1 Corpus corpus = ...;
2 // Define the section of interest
3 IndexSet[] indices = IndexUtils.wrap(1,3,4,5);
4 // Define the layers of interest
5 Scope scope = Scope.withLayers(corpus, "token",
  "content");
6 // Filtered "window" of the corpus
7 CorpusView view = corpus.createView(scope, indices,
  AccessMode.READ, Options.NONE);
8 // Our "triple store" of annotations
9 AnnotationLayer content = view.fetchLayer("content");
10 // Request loading of the corpus portion
11 view.getPageControl().load();
12 // Interface for accessing the view
13 CorpusModel model = view.getModel();
14 // Contained "list" of tokens
15 Container rootContainer = model.getRootContainer();
16 // Iterate over all tokens
17 for(int i=0; i<rootContainer.getItemCount(); i++) {
18   Item item = rootContainer.getItemAt(i);
19   System.out.println(content.getValue(item, "form"));
20 }
```

Often an application does not require elaborate filtering of a corpus before processing. Especially analysis tools such as parsers simply need a way of accessing an entire corpus one element at a time. For this use-case the framework provides streaming, as shown in the following code:

```
1 Corpus corpus = ...;
2 // Define the layers of interest
3 Scope scope = Scope.withLayers(corpus, "token",
  "content");
4 // Create stream for entire corpus
5 ItemStream stream = corpus.getStream(scope,
  AccessMode.READ, Options.NONE);
6 // Our "triple store" of annotations
7 AnnotationLayer content = view.fetchLayer("content");
8 // Traverse items in sequence
9 while(stream.advance()) {
10   Item item = stream.currentItem()
11   System.out.println(content.getValue(item, "form"));
12 }
```

Again client code defines the layers it is interested in, but this time it obtains an `ItemStream` that lets it iterate over all the tokens in the corpus. Since there is no horizontal filtering involved this time, the output will be the entire sentence from Figure 6 "Finding the right format is tricky .".

5.3. Fragmentation

What is the obligatory form for a corpus' primary data? Should it be a collection of already segmented units? Or should it be the original raw data such as the entire text of a book? Rarely is there an absolute answer for such questions within the universe of highly diverse corpus resources.

Our framework therefore does not force a singular approach for composing a corpus. The previous code snippets always used elements of `token` layer that already were properly segmented. The next example shows how those tokens can be created by splitting annotation values to fragment the respective original text.

```
1 // Simplify creation of building blocks
2 LayerMemberFactory factory = newMemberFactory();
3 // Existing item that represents entire text
4 Item text = ...;
5 // Define span for "Finding"
6 Position begin = Positions.create(0);
7 Position end = Positions.create(6);
8 Fragment token1 = factory.newFragment(sentence, 1,
  text, begin, end);
9 // Define span for "the"
10 begin = Positions.create(8);
11 end = Positions.create(10);
12 Fragment token2 = factory.newFragment(sentence, 2,
  text, begin, end);
```

5.4. Using Manifests

Manifests (the metadata part of our framework) offer a great way for applications to optimize computation or graphical user interfaces specifically for the content of a corpus in advance. Since the entire composition of a corpus object is described in detail in a formalized manner by its associated manifest, client code can use this information without having to actually read the content of a the corpus or to tailor its behavior to a certain type of resource.
The following code snipped illustrates how an application can use the manifest in Figure 5 (more specifically, the tagset template from Figure 4) to create specialized components for its user interface (UI):

```
1 // Manifest as provided to client code
2 AnnotationLayerManifest layerManifest = ...;
3 // Pick part-of-speech annotations as example
4 AnnotationManifest annotations =
  layerManifest.getAnnotationManifest("pos");
5 // Fetch the allowed values for this annotation
6 ValueSet tagset = annotations.getValueSet()
7 for(int i=0; i<tagset.valueCount(); i++) {
8   Object value = tagset.getValueAt(i);
9   // Check if advanced documention is available
10  if(value instanceof ValueManifest) {
11    ValueManifest manifest = (ValueManifest) value;
12    // Human readable id of the tag
13    String name = manifest.getName();
14    // Detailed description of the tag
15    String description = manifest.getDescription();
16    // Actual underlying tag value
17    value = manifest.getValue();
18    // Now make a more user friendly UI
19    addComplexUiElement(value, name, description);
20  } else {
21    // Use "value" without additional info
22    addSimpleUiElement(value);
23  }
24 }
```

By exploiting the information available through the predefined tagset, the application can create rich UI components that provide increased usability to its users. For instance, human readable names and descriptions that accompany the bare value definitions in the manifest can be used for additional info-labels or tooltips in the interface.

6. Availability

The core of our framework is available as a collection of individual Java libraries that cover and implement different aspects of our approach. We published the model libraries themselves, as well as a detailed documentation in the framework of CLARIN[10] and made it available via a persistent identifier[11] in order to ensure sustainability.

[10] `https://www.clarin.eu/`

[11] `http://hdl.handle.net/11022/1007-0000-0007-C636-D`

7. Conclusion

In this paper we presented our design and implementation of a novel corpus modeling framework. It approaches the task of modeling corpus resources from a perspective of processing and querying. Being available as a Java toolkit it can be used to provide unified in-memory representations of arbitrary corpus resources. Independence wrt to linguistic theories or tagsets allows it to work across different modalities. While our reference implementation is provided in Java, the default metadata serialization format is XML and so the entire concept can be also transfered to other programing languages.

In the future we plan on broadening the support for (standardized) serialization formats and also provide a built-in interface for database connectivity. While currently focused strictly on the modeling of corpora as collections of utterances, we do consider extending the toolkit to at least conceptually recognize other resources such as lexicons for which adequate modeling solutions already exist.

8. Acknowledgements

This work was funded by the German Federal Ministry of Education and Research (BMBF) via CLARIN-D, No. 01UG1120F and the German Research Foundation (DFG) via the SFB 732, project INF.

9. Bibliographical References

Banski, P., Frick, E., and Witt, A. (2016). Corpus query lingua franca (cqlf). In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Baumann, S. and Riester, A. (2013). Coreference, lexical givenness and prosody in German. *Lingua*, 136:16–37.

Bird, S. (2006). NLTK: The Natural Language Toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72, Sydney, Australia, July. Association for Computational Linguistics.

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345.

Burchardt, A., Padó, S., Spohr, D., Frank, A., and Heid, U. (2008). Constructing integrated corpus and lexicon models for multi-layer annotations in OWL DL. *Linguistic Issues in Language Technology*, 1:1–33.

Carletta, J., Kilgour, J., O'Donnell, T., and O'donnell, T. (2003). The nite object model library for handling structured linguistic annotation on multimodal data sets. In *In Proceedings of the EACL Workshop on Language Technology and the Semantic Web (3rd Workshop on NLP and XML, NLPXML-2003*, page 2003.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). Gate: An architecture for development of robust hlt applications. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 168–175, Stroudsburg, PA, USA. Association for Computational Linguistics.

Eckart, K. and Gärtner, M. (2016). Creating Silver Standard Annotations for a Corpus of Non-Standard Data. In Stefanie Dipper, et al., editors, *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, volume 16 of *BLA: Bochumer Linguistische Arbeitsberichte*, pages 90–96, Bochum, Germany.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Francopoulo, G., George, M., Calzolari, N., Monachini, M., Bel, N., Pet, Y., and Soria, C. (2006). Lexical markup framework (lmf. In *In Proceedings of LREC2006*.

Gärtner, M., Thiele, G., Seeker, W., Björkelund, A., and Kuhn, J. (2013). ICARUS – an extensible graphical search tool for dependency treebanks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Sofia, Bulgaria, August. Association for Computational Linguistics.

Gärtner, M., Björkelund, A., Thiele, G., Seeker, W., and Kuhn, J. (2014). Visualization, search, and error analysis for coreference annotations. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 7–12, Baltimore, Maryland, June. Association for Computational Linguistics.

Gärtner, M., Schweitzer, K., Eckart, K., and Kuhn, J. (2015). Multi-modal visualization and search for text and prosody annotations. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 25–30, Beijing, China, July. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., and Zhang, Y. (2009). The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, CoNLL '09, pages 1–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M., (2013). *Integrating NLP Using Linked Data*, pages 98–113. Springer Berlin Heidelberg, Berlin, Heidelberg.

Hinrichs, E. W., Hinrichs, M., and Zastrow, T. (2010). WebLicht: Web-Based LRT Services for German. In *Proceedings of the ACL 2010 System Demonstrations*, pages 25–29.

Ide, N. and Suderman, K. (2007). GrAF: A graph-based format for linguistic annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8, Prague,

Czech Republic, June. Association for Computational Linguistics.

Ide, N. and Suderman, K. (2014). The linguistic annotation framework: A standard for annotation interchange and merging. *Lang. Resour. Eval.*, 48(3):395–418, September.

Ide, N., Romary, L., and de la Clergerie, E. (2003). International standard for a linguistic annotation framework. In Jon Patrick et al., editors, *Proceedings of the HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS)*, pages 25–30.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., DiPersio, D., Shi, C., Suderman, K., Verhagen, M., Wang, D., and Wright, J., (2016). *The Language Application Grid*, pages 51–70. Springer International Publishing, Cham.

König, E., Lezius, W., and Voormann, H., (2003). *TIGERSearch 2.1 User's Manual. Chapter V - The TIGER-XML treebank encoding format*. IMS, Universität Stuttgart.

Krause, T. and Zeldes, A. (2014). Annis3: A new architecture for generic corpus query and visualization. *Digital Scholarship in the Humanities*.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). CoNLL-2011 Shared Task: Modeling Unrestricted Coreference in OntoNotes. In *CoNLL: Shared Task*, pages 1–27, Portland, Oregon, USA, June.

Roesiger, I. and Riester, A. (2015). Using prosodic annotations to improve coreference resolution of spoken text. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 83–88, Beijing, China, July. Association for Computational Linguistics.

Verhagen, M., Suderman, K., Wang, D., Ide, N., Shi, C., Wright, J., and Pustejovsky, J., (2016). *The LAPPS Interchange Format*, pages 33–47. Springer International Publishing, Cham.

Windhouwer, M. and Wright, S. E. (2012). Linking to linguistic data categories in isocat. In *Linked Data in Linguistics*, pages 99–107. Springer.

Zipser, F., Zeldes, A., Ritz, J., Romary, L., and Leser, U. (2011). Pepper: Handling a multiverse of formats.

Zipser, F. (2009). Entwicklung eines Konverterframeworks für linguistisch annotierte Daten auf Basis eines gemeinsamen (Meta-)modells, November. In this diploma thesis I present a framework to convert linguistic data coming from a specific linguistic format to other linguistic formats. This approach is based on a common meta-model, which is used as an intermediate step to decrease the number of necessary mappings.

An Annotation Language for Semantic Search of Legal Sources

Adeline Nazarenko[1], **François Levy**[1], **Adam Wyner**[2]
[1]LIPN, Paris 13 University – Sorbonne Paris Cité & CNRS
{Adeline.Nazarenko, Francois.Levy}@lipn.univ-paris13.fr
[2]Department of Computing Science, University of Aberdeen
azwyner@abdn.ac.uk

Abstract

While formalizing legal sources is an important challenge, the generation of a formal representation from legal texts has been far less considered and requires considerable expertise. In order to improve the uniformity, richness, and efficiency of legal annotation, it is necessary to experiment with annotations and the annotation process. This paper reports on a first experiment, which was a campaign to annotate legal instruments provided by the Scottish Government's Parliamentary Counsel Office and bearing on Scottish smoking legislation and regulation. A small set of elements related to LegalRuleML was used. An initial guideline manual was produced to annotate the text using annotations related to these elements. The resulting annotated corpus is converted into a LegalRuleML XML compliant document, then made available via an online visualisation and query tool. In the course of annotating the documents, a range of important interpretive and practical issues arose, highlighting the value of a focused study on legal text annotation.

Keywords: annotation, semantic search, legal information, methodology

1. Introduction

Formalizing legal sources has been identified for years as an important challenge for the development of legal content management in order to increase interoperability or support dialogue between various legal systems and actors. The generation of the envisaged formal representation from the legal texts has been far less considered and requires considerable expertise. Nevertheless, enriching the texts with a layer of semantic annotations and opening new access routes to textual content represent a critical issue for all the legal work that rests essentially on the analysis and interrogation of sources.

Legal annotation requires legal skills to understand the text and the significance of legal statements as well as logical skills to understand what conclusions can be drawn with respect to the annotations. In order to improve the uniformity and the cost of legal annotation, it is necessary to experiment with annotations which allow us to query, analyse, and interrogate legal sources.

Our research was driven by requirements set by the parliamentary counsel of the Scottish Government's Parliamentary Counsel Office, which aims to improve internal legislative drafting. As part of this, it is useful to provide a corpus of law in electronically readable form which can be queried to address the following competence questions:

1. What are all the offences and associated penalties or defences?
2. What prohibitions apply to tobacco products?
3. What obligations have been placed on which entities, *e.g.* shop owners?
4. What permissions are given to Scottish Ministers?
5. Given a provision, what are related overriding or reparation provisions?

For parliamentary counsels, who draft the law, the answers to these questions ought to be all relevant provisions so that they can compare formulations or check interactions between provisions. For the technical solution to extracting such questions, semantic meta-data must be added to the text of the corpus. To realise such a corpus, we require an annotation language along with a sound methodology of annotation. The main aim of this paper is to provide initial elements of such a methodology, given some elements of LegalRuleML as an annotation language. In the conclusion, we briefly discuss the extent to which the competence questions were addressed.

As a first experiment, a campaign has been organized to annotate 10 legal instruments provided by the Scottish Government's Parliamentary Counsel Office (41,859 words, $\sim$ 140 pages). All bear on Scottish smoking legislation and regulation. An initial set of guidelines was produced to annotate the text using annotations related to LegalRuleML elements. There was a team of 6 student annotators - 3 from the law school, 2 from the computer science department, and 1 from the linguistics department. This paper reports on this experiment and the guidelines. Relatedly, it presents the interpretative issues that annotators raised during the first annotation phase. It shows how difficult it is to comply with an annotation language and to LegalRuleML semantics. We present solutions to make legal rule annotation feasible on a large scale and with high inter-annotator agreement.

The paper is organized as follows. Section 2. discusses existing work and motivates our choice of annotation. Section 3. sketches the LegalRuleML formalism, which is the target of the annotation process. Section 4. describes the simplified annotation language, which has been designed for legal annotators. In Section 5., we outline our annotation manual, which gives our solutions to how we resolve ambiguities and favor agreement between annotators.

2. Annotation of Legal Texts

Formalization of legal texts has since long been considered for automating legal reasoning (Bench-Capon et al., 1987). Several ontologies have been built for the representation of

legal concepts (Hoekstra et al., 2009; Wyner and Hoekstra, 2012), and there are formal languages for the representation of the content of legal documents, such as SBVR (OMG, 2008) and LegalRuleML (Athan et al., 2015). However, a direct translation from natural language, particularly legal language, into a given formal language is extremely difficult to accomplish. Several transformation steps are required to cover the "formalization continuum" (Baumeister et al., 2011; Lévy and Nazarenko, 2013); controlled languages have been proposed to bridge the gap between the proper languages of legal actors and a logical language supporting formal reasoning (Hoefler, 2012; Feuto Njonko et al., 2014; Lévy et al., 2015; Wyner et al., 2016; Wyner et al., 2017).

Based on the advances of the semantic web, another approach consists in enriching documents with annotations so as to ease the content management which is beyond keyword search, for example, encoding in XML the document structure of legal sources (e.g. Akoma Ntoso[1]) (Casanovas et al., 2016). The identification and resolution of cross references has also been recognized as a key element for the exploitation of legal sources, e.g. Google Scholar case law searches. Following this track, we have laid the foundations for an annotation language that is compatible with LegalRuleML (Nazarenko et al., 2016). In this paper we focus on the semantics of rules rather than on the text structure (Akoma Ntoso), but both types of annotation are complementary and are expected to work together in the end.

Many annotation tasks have been launched for the last decades. Among methodological issues, the guidelines definition and training phase (the annotators learn the task, the target annotation language, and application requirements) are essential (Fort, 2016). The present paper reports on this critical preparatory phase.

3. A Glimpse of LegalRuleML

LegalRuleML has been chosen as the target formal language, since it has been specifically developed to suit legal texts. It is at the crossroad of two sources – RuleML (Boley et al., 2010), which encodes logical rules in a portable way, and Akoma Ntoso, which annotates meta-properties of legal documents.

In Akoma Ntoso, meta-properties include the name of the document, its date, version, its jurisdiction (area where applicable), its classification among legal documents (is it an Act, a Decree, a Statutory Instrument, a decision, etc.), the authorship (Minister, Council, Judge, etc.), the status (in course of elaboration, promoted, ammended, cancelled, etc.)... They also include information about the layout, headings, and provisions as they are numbered in the text. All this information is essential for lawyers who need to know the structure and authority of the text as well as its relation to other texts. We take as given that the document structure is annotated in Akoma Ntoso. RuleML facilitates representation of Propositional and Predicate Logic, as well as negation-as-failure, modalities, and other logical features. RuleML encodes each level of syntactic analysis of a formal rule: e.g. antecedent and consequent parts, the atoms and arguments which build each part, logical coordinators, quantifiers, ...

LegalRuleML adopts much of RuleML and extends it with elements for legal classification of statements (see also Section 4.). A *prescriptive* statement expresses a deontic conclusion such as *permission*, *prohibition*, or *obligation*. A constitutive statement defines the conceptual meaning of the terms, in order to help the interpretation. *Penalty* statements list and describe penalties which can be imposed by an authority, while *reparation* statements make the link between a prescriptive statement and the penalties that apply where the prescription has been violated. LegalRuleML has also a mechanism named *association* to link one or several rules to one or several sources, provided the sources have an IRI. The *context* annotation adds one more level so that several concurrent interpretations can be recorded. Figure 1 is an example of LegalRuleML representation of a legal rule in its XML encoding.

4. A Simple Annotation Language

Encoding legal documents in LegalRuleML to take advantage of standard Semantic Web technologies is difficult to achieve in one step. On the one hand, legal professionals cannot reasonably be expected to engage with the complexity of LegalRuleML conformant encoding of legal documents. On the other hand, it is important to keep track of the text which has been translated. Moreover, legal professionals rather than computer scientists are best placed to understand the legal sources and how they should be annotated. To accommodate our requirements as well as to track the steps and quality of the translations, we only worked with a small palette of LegalRuleML elements as text annotations which classify legal statements and their relations. The selected elements are key parts of the description of rules in LegalRuleML. These annotations leave room for further refinement in subsequent steps:

- Permission: the bearer is allowed to do something or be in a state.
- Obligation: the bearer is bound to do something or be in a state, for otherwise, the bearer is in violation.
- Prohibition: the bearer is bound not to do something or be in a state, for otherwise, the bearer is in violation.
- Constitutive: a legal definition of a concept.
- Override: an indication that one legal rule takes precedence over another.
- Penalty: a description of a sanction.
- Reparation: an indication of a link between a prescriptive norm and a penalty to be applied in case of violation.

Any legal rule relies on one of the prescriptive statement types (permission, obligation or prohibition). Constitutive statements and rule relationships (override and reparation) are essential to the interpretation of legal rules. In the texts that we have considered, nearly all of it falls under one or

[1] http://www.akomantoso.org/

```
<lrml:LegalRuleML ...>
 <lrml:LegalSources memberType="TBD">
    <lrml:LegalSource key="ref1"
      sameAs="http://www.legislation.gov.uk/ukpga/1998/42/contents" />
  <lrml:Associations key="sourceBlock1">
    <lrml:Association>
      <lrml:appliesSource keyref="#ref1"/>
      <lrml:toTarget keyref="#rule_1a"/>
    </lrml:Association>
  </lrml:Associations>
  <lrml:Context key="ruleInfo1">
    <lrml:appliesAssociations keyref="#sourceBlock1"/>
    <lrml:inScope keyref="#stmts_1a"/>
  </lrml:Context>
<lrml:Statements key="stmts_1a">
    <lrml:ConstitutiveStatement>
      <ruleml:Rule key=":rule_1a">
        <ruleml:if>
          <ruleml:Atom>
            <ruleml:Rel>P</ruleml:Rel>
          </ruleml:Atom>
        </ruleml:if>
        <ruleml:then>
          <ruleml:Atom>
            <ruleml:Rel>Q</ruleml:Rel>
          </ruleml:Atom>
        </ruleml:then>
      </ruleml:Rule>
    </lrml:ConstitutiveStatement>
  </lrml:Statements>
</lrml:LegalRuleML>
```

Figure 1: A Conditional Rule and its Source in LegalRuleML

the other of the chosen categories, which we take as indicative of a good selection of annotations.
However, annotators met more ambiguity than expected (see section 5.). In concrete terms, the annotators have to select the relevant statements and to annotate them as in

```
<ConstitutiveStatement id=c20>
In this schedule, ``fixed
penalty notice'' means a
notice offering a person the
opportunity of discharging any
liability to conviction for
an offence under section 1 by
payment of a fixed penalty.
<ConstitutiveStatement>
```

Identifiers are facultative and are used for reference, *e.g.* in an `Override` or `Reparation` statement. For instance, Statement c22 overrides c20 is written `<override over c22 under c20>`.
Whatever their background, annotators had no problem to learn and use this simple language. They could follow the definitions and the explanations given in the manual.
Presently, 558 statements are annotated, the annotation manual has been updated after discussions among annotators, and the corpus can be viewed and queried by a search tool available at `http://tal.lipn.univ-paris13.fr/LexEx`.

5. Annotation Instructions

Annotation instructions must be clear for annotators, but nevertheless allow them to analyse complex cases. In addition, several issues arose. The more important points of discussion are considered here.

Annotation goal Annotation guidelines must explain what is the goal of the task and what kinds of annotations are expected. In our case, annotators were told to concentrate on statements expressing definitions and rules, and thus to skip facts and factual statements since our texts are provisions rather than cases. This means that some fragments of text were left unannotated.

List and nested annotations It was difficult for annotators to identify the borders of the statements and their parts, especially for lists which are numerous and often nested in legal documents. Unless the list items stand for autonomous statements, the annotators were instructed to annotate a whole list as a single statement. This gives a coarse-grained annotation, but directly relates to the source complexity.

Modals Recognizing the deontic status of a statement appeared to be challenging for annotators. They tried to rely on the presence of a modal verb but this is far from obvious. Modal verbs may be ambiguous: *may* or *must* can have an epistemic as well as a deontic meaning – the latter related to permission and prohibition, the former to the degree of certainty. For instance, in

> An 'age verification policy' is a policy that steps are to be taken to establish the age of a person attempting to buy a tobacco product [...] if it appears to the person selling the tobacco product that the customer may be under the age of 25 [...].

may has an epistemic rather than a permission value.
Deontic values can be expressed otherwise than by modals:

- A permission can be considered as an absence of prohibition (the answer to *Is it allowed to... ?*). In such weak cases (von Wright, 1963), there is often no explicit statement, as opposed to strong permissions which are usually explicitly stated (they create exceptions to a prohibition which would otherwise apply).
- Prohibitions are often introduced by *It is an offence to...*. They were often wrongly annotated as constitutive statements (offence definitions). After discussion, we considered that those statements had a performative character and should be tagged as prohibitions.
- Sentences introduced by *It is a defence with respect to [subsection xyz] to...* raised problems for annotators. We classified exceptions to offences as strong permissions. The solution was the same for *A person does not commit an offence...* or *No offence is commited with respect to...*. As this interpretation was not obvious for annotators, dedicated explanations and examples were added to the annotation guide.

Surprisingly, *may* also happens to express an obligation as in *the sheriff may do A, B or C*, which states that the sheriff has to take one action, be it A, B or C.
Another difficulty arose with a distinction between permissions or powers, as in:

> The Scottish Ministers may by regulations amend the age specified in subsection (3).

> A constable making a requirement under subsection (1) may also require the person to supply the constable with the person's name and address....

The preceding examples could be defined as *powers*, which are permissions given to some officials to modify or adapt obligations or prohibitions on other parties. For instance, violation of a power generally entails that some action is ineffective. A similar difficulty arises with *must* and could be solved with *duties*.

Exceptions Exceptions are frequent in legal sources: they describe a general case and state that a different conclusion holds in some specific sub-conditions. Even if the generic and specific cases can be difficult to correlate for the annotators, annotating exceptions in legal sources is of prime importance. However, the wording can be quite different from one case to another. In the following example, the exception immediately follows the general case and is marked by *but* and *only*, but was missed by the annotators:

> (1) An enforcement officer may give a person a fixed penalty notice if the officer has reason to believe that the person has committed an offence under Chapter 1 or 2.
>
> (2) But a person may be given a fixed penalty notice only if the person is aged 16 or over.

When marked by *except that*, *at the exception of*, etc., exceptions are easy to detect, but exceptions are not always semantically self-explaining:

> "care service" has the meaning given by section 47(1) of the Public Services Reform (Scotland) Act 2010, except that it does not include a service mentioned in paragraph (k) of that section (child minding).

Reparations and penalties The majority of reparation statements have a simple form:

> A person who commits an offence under subsection (2) is liable on summary conviction to a fine not exceeding level 3 on the standard scale.

but roughly 40% of these sentences where classified as penalties. The distinction is subtle. The following case

> The fixed penalty for an offence under section 1 is £100.

which looks like a penalty definition, but actually states an offence reparation and describes the related penalty. To clarify, we give priority to reparation and only ask annotators to mark up penalties when they are stated separately. Once the text has been annotated and accepted as correct (peer-reviewed, then adjudicated), it is passed to LegalRuleML annotators, who transform the annotated text into a LegalRuleML compliant XML document. These documents can then be queried on the search tool.

6. Conclusion

In this paper, we motivated and applied the annotation of legal texts using elements related to LegalRuleML, an XML markup language for legal texts. We proposed using a simplified palette of elements from LegalRuleML so as to focus attention on the statement classification and relations. As anticipated, this raised a range of important and interesting interpretive issues, which had to be addressed in the annotation guidelines.
In (Wyner et al., 2017), the corpus was evaluated with respect to the competency questions introduced in Section 1. and the web search application, which retrieves the annotated statements based on their types as well as on the keywords or text patterns they contain. An example of competency questions is searching all the definitions of offences, which could be done to check that something is not prohibited (weak permission). All these definitions involve the

word *offence*. Searching this word yields 70 statements of different kinds. To focus on definitions, we require also that the statement be a `Prohibition`, which reduces to 26 answers. All the erroneously recovered statements specify the procedure which applies in case of offence and not the offence itself.

An other example is searching provisions stating obligations placed on shop owners. They are of course `Obligations`, but *Shop* appears only once in these texts and *owner* never. Checking by hand, *business* is the more usual term, but also *management*, *control*, and *responsible person*. Some further semantic annotation is needed in order to recognize the contextual synonymy of these terms.

The results are high quality. For example, for definitions of offences the recall is 1 and precision is .84; associated defenses, obtained by searching `Permission` elements which contain any of *defence* or *offence*, reach a recall 1 and precision .60; querying *Scottish Ministers* in `Permission` elements yields precision .952 and recall .875. The results also highlighted areas for further refinement. For parliamentary counselors, such results are attractive, producing meaningful and accessible results quickly.

While we have applied a relatively small set of annotations to a modest textual corpus, several valuable lessons were learned that can be taken forward. Working to formalise natural language, it was very helpful to work with a highly scoped annotation language, challenging us understand the interpretation of both the annotation and the language. Addressing competency questions provided a clear and useful goal towards demonstrable solutions. Yet, despite the scoped annotation language and modest corpus, the annotation task proved to be more complex than anticipated, partly due to the complexities of the source language in structure and sense. The observations will be useful in ongoing work. In future work, we plan to extend the corpus based on the revised manual, apply machine learning to the resulting gold standard corpus, and enrich LegalRuleML with elements such as *right*, *duty*, *power*, and *defence*.

7. Acknowledgements

We thank the University of Aberdeen's Impact, Knowledge Exchange, and Commercialisation Award for this 10 week study. The French National Research Agency (ANR-10-LABX-0083) supported this work in the context of the Labex EFL. We also thank the student staff: A. Andonov, A. Faulds, E. Onwa, L. Schelling, R. Stoyanov, and O. Toloch.

8. Bibliographical References

Athan, T., Governatori, G., Palmirani, M., Paschke, A., and Wyner, A. (2015). LegalRuleML: Design principles and foundations. In *Reasoning Web International Summer School*, pages 151–188. Springer.

Baumeister, J., Reutelshoefer, J., and Puppe, F. (2011). Engineering intelligent systems on the knowledge formalization continuum. *International Journal of Applied Mathematics and Computer Science (AMCS)*, 21(1).

Bench-Capon, T., Robinson, G., Routen, T., and Sergot, M. (1987). Logic programming for large scale applications in law: A formalisation of supplementary benefit legislation. In *International Conference on Artificial Intelligence and Law*, pages 190–198.

Boley, H., Paschke, A., and Shafiq, M. O. (2010). RuleML 1.0: The overarching specification of web rules. In Mike Dean, et al., editors, *Proceedings of Semantic Web Rules*, pages 162–178. Springer.

Casanovas, P., Palmirani, M., Peroni, S., van Engers, T. M., and Vitali, F. (2016). Semantic web for the legal domain: The next step. *Semantic Web*, 7(3):213–227.

Feuto Njonko, P., Cardey, S., Greenfield, P., and El Abed, W. (2014). RuleCNL: A controlled natural language for business rule specifications. In Brian Davis, et al., editors, *Controlled Natural Language*, pages 66–77. Springer International Publishing.

Fort, K. (2016). *Collaborative Annotation for Reliable Natural Language Processing: Technical and Sociological Aspects*. Wiley-ISTE, July.

Hoefler, S. (2012). Legislative drafting guidelines: How different are they from controlled language rules for technical writing? In Tobias Kuhn et al., editors, *Proceedings of CNL 2012*, pages 138–151.

Hoekstra, R., Breuker, J., Di Bello, M., and Boer, A. (2009). LKIF core: Principled ontology development for the legal domain. In *Law, Ontologies and the Semantic Web - Channelling the Legal Information Flood*, pages 21–52.

Lévy, F. and Nazarenko, A. (2013). Formalization of natural language regulations through SBVR structured english. In Leora Morgenstern, et al., editors, *Proceedings of RuleML2013*, pages 19–33, Seattle, WA, USA, July.

Lévy, F., Nazarenko, A., and Wyner, A. (2015). Towards a high-level controlled language for legal sources on the semantic web. In *Workshop on Legal Domain And Semantic Web Applications (LeDA-SWAn 2015)*.

Nazarenko, A., Lévy, F., and Wyner, A. (2016). Towards a methodology for formalizing legal texts in LegalRuleML. In Floris Bex et al., editors, *Proceedings of JURIX2016*, pages 149–154. IOS Press.

OMG. (2008). Semantics of business vocabulary and business rules SBVR. formal specification, v1.0. Technical report, The Object Management Group.

von Wright, G. (1963). *Norm and action: A logical inquiry*. Routledge and Kegan Paul.

Wyner, A. and Hoekstra, R. (2012). A legal case OWL ontology with an instantiation of *Popov v. Hayashi*. *Artificial Intelligence and Law*, 20(1):83–107, March.

Wyner, A., Nazarenko, A., and Lévy, F. (2016). Towards a high-level controlled language for legal sources on the semantic web. In Brian Davis, et al., editors, *Proceedings of CNL2016*, pages 92–101, Aberdeen, UK, July. Springer.

Wyner, A. Z., Gough, F., Lévy, F., Lynch, M., and Nazarenko, A. (2017). On annotation of the textual contents of scottish legal instruments. In Adam Z. Wyner et al., editors, *Proceedings of JURIX 2017*, pages 101–106. IOS Press.

Resource Interoperability for Sustainable Benchmarking: The Case of Events

Chantal van Son, Oana Inel, Roser Morante, Lora Aroyo, Piek Vossen
Vrije Universiteit Amsterdam
De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands
{c.m.van.son, oana.inel, r.morantevallejo, lora.aroyo, piek.vossen}@vu.nl

Abstract

With the continuous growth of benchmark corpora, which often annotate the same documents, there is a range of opportunities to compare and combine similar and complementary annotations. However, these opportunities are hampered by a wide range of problems that are related to the lack of resource interoperability. In this paper, we illustrate these problems by assessing aspects of interoperability at the document-level across a set of 20 corpora annotated with (aspects of) events. The issues range from applying different document naming conventions, to mismatches in textual content and structural/conceptual differences among annotation schemes. We provide insight into the exact document intersections between the corpora by mapping their document identifiers and perform an empirical analysis of event annotations showing their compatibility and consistency in and across the corpora. This way, we aim to make the community more aware of the challenges and opportunities and to inspire working collaboratively towards interoperable resources.

Keywords: resource interoperability, annotation consistency, events

1 Introduction

Language resources are at the core of most research in computational linguistics and natural language processing (NLP) for system development and benchmarking. There is already a wealth of resources available and the collection is continuously expanding. With this growth, monitoring their creation and making them interoperable is becoming increasingly important for finding, accessing and reusing existing corpora. For example, different annotation layers applied to the same document provide valuable opportunities for studying and leveraging interdependencies between different types of annotation, but these opportunities are hampered when too laborious conversion steps are required to resolve structural or conceptual differences in their representations (Chiarcos, 2012a). Despite the efforts of many initiatives working towards solutions (Section 2), interoperability issues still persist today and become more and more problematic. On the one hand, this is because reaching consensus on standards and best practices is not a straightforward task and naturally takes time. On the other hand, we hypothesize that the problems that arise when interoperability is lacking have not been illustrated enough for the necessity of solutions to be widely recognized.

Although there are various types of language resources, we limit our discussion to annotated text corpora for which we define interoperability at the levels presented in Figure 1. At the corpus-level, interoperability involves documenting and representing metadata of the data collection as a whole such as its name, language, type, genre, source, creator, year, size, etc., to enable resolving the identity of corpora in a uniform way. At the document-level, we can further distinguish between the metadata of the document (e.g. filename, language, size) and its body, where the latter consists of a textual string and the linguistic annotations of its substrings (e.g. sentences, phrases, tokens). With respect to the annotations, we adopt the distinction between structural interoperability (annotations of different origin are represented using the same formalism) and conceptual interoperability (annotations of different origin are linked to a common vocabulary) as defined by Chiarcos (2012a).

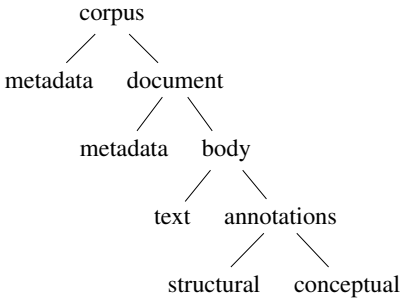


Figure 1: Levels of interoperability

We present an analysis on the document-level interoperability across a set of 20 corpora that have been annotated with events, predicates or propositions. The NLP community defines these terms in various ways, often using each other in their definitions, as in: a *proposition* is formed by a *predicate* with its arguments; *events* are expressed by *predicates* describing situations that happen/occur (Saurí et al., 2006); *predicates* can be of "*propositional*" type (representing an *event*, state, etc.) (Meyers, 2007). There is, however, little agreement on the degree of meaning overlap and relatedness. For the sake of clarity, we will use in this article the term *event* to refer to all three of the overlapping and interrelated notions. Many existing event corpora contain annotations of different aspects of events that are often applied to the same documents (Pustejovsky et al., 2005), providing an interesting use case for analysing interoperability. Furthermore, event annotations involve a wide range of properties and phenomena which makes it ultimately rewarding to achieve interoperability and combine these annotations.

The contributions of this paper are the following:

- **a comprehensive overview of interoperability issues**

across event corpora that result from differences in document naming conventions, textual content and structural/conceptual representations of annotations;
- **a method for aligning diverse language resource corpora** to identify divergent and overlapping corpora;
- **an overview of document intersections across event corpora** revealing opportunities for comparing and combining different annotation layers;
- **an empirical analysis of event annotation compatibility and consistency** in and across corpora.

The remainder of this paper is structured as follows. Section 2 discusses existing initiatives for improving resource interoperability and related studies comparing annotation schemes across corpora. Section 3 provides an overview of the event corpora included in our study. Section 4 reviews metadata-level interoperability across these corpora and shows their overlap in documents, after which we focus on a subset: PropBank/NomBank (PB/NB), FactBank (FB) and TempEval-3 (TE3). We analyse their interoperability at the level of text in Section 5 and at the level of annotations in Section 6. Finally, Section 7 concludes with our lessons learned and proposes some best-practice guidelines.

2 Related Work

There is a range of initiatives collecting and indexing metadata of language resources at the corpus-level to support researchers in finding the right one for their task or application. These include OLAC (Simons and Bird, 2003), Language Grid (Ishida, 2006), the LRE Map (Calzolari et al., 2012), the ELRA Universal Catalog,[1] the LDC Catalog,[2] META-SHARE (Piperidis, 2012), CLARIN (Krauwer and Hinrichs, 2014) and Linghub (McCrae and Cimiano, 2015). The last decades have also seen various metamodel proposals for representing annotations that facilitate structural interoperability, most of which are also translatable to each other. These include GATE (Cunningham, 2002), UIMA (Ferrucci and Lally, 2004), LAF/GrAF (Ide and Romary, 2004; Ide and Suderman, 2007), NIF (Hellmann et al., 2013), NAF/GAF (Fokkens et al., 2014) and PAULA/POWLA (Chiarcos and Erjavec, 2011; Chiarcos, 2012b). Repositories of linguistic annotation terminology, such as GOLD (Farrar and Langendoen, 2003), ISOcat (Windhouwer and Wright, 2012) and its successor CCR,[3] make it possible to overcome the heterogeneity of annotation schemes by acting as an interlingua that allows mapping annotations from one scheme to another, thus addressing conceptual interoperability (Chiarcos, 2012a).

However, far from all corpora that we use today follow the principles mentioned above. This may be because they were created in a time where these standards simply did not yet exist. For more recently created corpora, however, there is presumably a plethora of reasons. We hypothesize that one of them is that whereas working groups such as the Open Linguistics Working Group (OWLG)[4] actively promote resource interoperability, there seem to be few examples (that we know of) that actually clearly illustrate the extent of the problems. Some studies indirectly discuss conceptual interoperability by comparing annotation schemes. For example, Aguilar et al. (2014) compare the Events, Entities and Relations represented in ACE, ERE, TAC-KBP Slot-filling, and FrameNet. Werner et al. (2015) compare the factuality/committed belief annotations in FactBank and the Language Understanding (LU) corpus. The differences between the representations of semantic propositions in PropBank, VerbNet and FactBank have been extensively described and even leveraged to build SemLink (Palmer et al., 2014). Close to our work is that of Pustejovsky et al. (2005), who discuss the issues involved in creating a Unified Linguistic Annotation (ULA) by merging the annotation schemes of PropBank, NomBank, TimeBank, the Discourse Treebank and Coreference Annotation. However, their work remains on theoretical ground by limiting their discussion to overlapping and conflicting annotations in example sentences. Our approach is unique in the sense that we provide empirical evidence by discussing the overlap of the actual annotations for the complete resources when aligning them on the same texts, as well as more general distributional similarities and differences with respect to their linguistic types (part-of-speech (POS), lemma).

3 Overview of Event Corpora

For this study, we focus on 20 corpora that are connected to each other in terms of annotations and/or document bodies. More specifically, the corpora have been annotated with events or predicates, or they share pieces of texts and, therefore, contain at least some documents annotated with events.[5] These corpora provide a range of opportunities to compare or merge annotations of events and their aspects. On the one hand, the definition of what constitutes an event varies across the corpora given the initial goal of the corpus. On the other hand, additional annotations capture various aspects of events, such as predicate-argument relations, semantic roles, event types, event coreference, event factuality, event time-stamping and event-event relations.

In the following overview we present a general description of the corpora, which we group according to their annotation standards (these groupings, however, are not clear-cut).

3.1 PropBank, OntoNotes & Abstract Meaning Representation (AMR)

PropBank (Palmer et al., 2004) has been one of the most influential corpora in NLP research. It provides semantic role annotations for all verbs in the 1M word Wall Street Journal (WSJ) section of the **Penn Treebank** (Marcus et al., 1999). Its creation led to that of **NomBank** (Meyers et al., 2008), which marks the sets of arguments that co-occur with nouns in the same set of documents. Most of the annotations of Penn Treebank and PropBank are now, slightly adapted, available from their successor **OntoNotes 5.0** (Weischedel et al., 2013), which contains other annotation layers such as coreference and

[1] http://universal.elra.info

[2] https://catalog.ldc.upenn.edu

[3] https://www.clarin.eu/ccr

[4] https://linguistics.okfn.org

[5] This is by no means a comprehensive list of all corpora meeting these criteria, but we hope this selection provides a good starting point.

named entities and also covers the Chinese and Spanish languages. Furthermore, it includes an additional 200K of broadcast news, 200K of broadcast conversation, 145K of P2.5 data and 200K of Web data taken from other sources. PropBank's representation of semantic roles is also used in the **Abstract Meaning Representation (AMR) corpus** (Knight et al., 2014), which represents the semantics of English sentences as single rooted, directed graphs with the aim of abstracting away from syntactic idiosyncrasies. It uses a variety of sources for its data, including WSJ news.

3.2 Automatic Content Extraction (ACE) & Entities, Relations and Events (ERE)

The key content extraction tasks of the Automatic Content Extraction (ACE) program (Doddington et al., 2004; Strassel et al., 2008), which ran between 1999 and 2008, were defined as the automatic detection and characterization of real-world Entities, Relations, and Events. However, the program mostly focused on entities and relations between them. Event annotations are available only in the **ACE 2005 Multilingual Training Corpus** (Walker et al., 2006), where annotators tagged the extent, trigger, polarity, tense, genericity, modality, participants and attributes for a constrained set of event (sub)types. This data has been reused in several other corpora, including OntoNotes and **Datasets for Generic Relation Extraction (reACE)** (Hachey et al., 2011) We also included **ACE-2 version 1.0** (Mitchell et al., 2003), which originally does not contain event annotations, but a selection of its documents was used in TimeBank, which does (Section 3.3).
ACE was followed by **Light ERE**, which was designed as a lighter-weight version of ACE with the goal of making annotation easier and more consistent. Modifications to ACE for Light ERE included a reduced inventory of entity and relation types, a slightly modified and reduced event ontology, and the addition of event coreference. In turn, Light ERE has transitioned to the more complex **Rich ERE**, with the latter enabling a more comprehensive treatment of phenomena such as event coreference (Song et al., 2015).[6]

3.3 TimeML

TimeML (Pustejovsky et al., 2003a) is a specification language for events and temporal expressions, designed to capture their attributes, to link them (event time-stamping) and to determine the temporal order between events. It has been applied in several corpora, including the **AQUAINT TimeML Corpus** (Brandeis University, 2008) and **TimeBank 1.2** (Pustejovsky et al., 2006). The documents in TimeBank come from PropBank and the ACE-2 corpus. In turn, data from TimeBank and AQUAINT TimeML was used to build **FactBank 1.0** (Saurí and Pustejovsky, 2009), adding a representation of factuality interpretation to the event annotations, and the evaluation datasets of the TempEval shared tasks, from which we selected the **TempEval-3 dataset** (UzZaman et al., 2013).
The TimeML specifications were followed to annotate events in **EventCorefBank (ECB)** (Bejan and Harabagiu, 2010) which was built to encode event structures with relations like SUBEVENT or REASON, and intra- and cross-document event coreference. ECB 1.0 consists of 482 documents from Google News clustered into 43 topics. A first extension to ECB was released by Lee et al. (2012), who revised and completed the original annotations and added entity coreference relations following the OntoNotes annotation guidelines for coreference (Pradhan et al., 2007). We included a second extension called **ECB+** (Cybulska and Vossen, 2014), which contains another corpus consisting of 502 documents, completely (re)annotated according to new guidelines. The corpus is annotated with event classes (based on TimeML), locations and times (based on ACE and TimeML), and intra- and cross-document coreference.

3.4 Other Annotation Standards

The full-text annotations in **FrameNet** (ICSI Berkeley, 2017) capture the frame semantic structures as defined in its lexical database (Fillmore et al., 2003). The documents come from different sources, including PropBank, the AQUAINT Program[7] and the **Lexical Understanding (LU) Annotation Corpus** (Diab et al., 2009). The latter contains annotations of dialog acts, event coreference, event relations and entity relations, but is best known for its annotations of committed belief, i.e. the strength of the author's beliefs and the degree of commitment to their utterance (similar to FactBank).
The **EventStatus Corpus** (Huang et al., 2017) annotated approximately 3,000 English and 1,500 Spanish news articles with temporal and aspectual properties of major societal events, that is, whether an event has already happened, is currently happening or may happen in the future. Its English documents were sourced from English Gigaword Fifth Edition (Parker et al., 2011), which also served as a source for other resources such as the ACE corpora.
Finally, in the **Richer Event Description (RED) Corpus** (O'Gorman and Palmer, 2016) a number of event-related annotation layers are integrated into a single representation of events and participants. It consists of 95 discussion fora and newswire documents annotated with entities, events, times, their coreference and partial coreference relations, and the temporal, causal and subevent relationships between the events. Its documentation mentions that this includes 55 documents annotated by a range of DEFT annotation formats, but it does not specify which ones.

From this overview, it is clear that the relations across the different corpora and their annotations are complex and not trivial, making it difficult to combine them. In the next sections, we will discuss their interoperability in more detail.

4 Document Interoperability: Metadata

The first step towards analysing similar or merging complementary annotation layers of different origin on the same pieces of text is to determine which documents are shared across the corpora of interest. Many corpora select all or a subset of their documents from existing corpora and possibly add additional ones for annotation, which is usually

[6] At the moment, the ERE data is not yet publicly available, but the LDC kindly provided us with a list of training data filenames.

[7] http://www-nlpir.nist.gov/projects/aquaint/

described in their documentation. This way we learn, for example, that the Wall Street Journal articles first released as part of Penn Treebank have also been used in PropBank, NomBank, OntoNotes, FrameNet, the AMR corpus, TimeBank, and more. We know that TimeBank also sourced documents from the ACE program and in turn was together with the AQUAINT TimeML Corpus the basis for creating FactBank and the TempEval-3 evaluation dataset. However, when documents are taken from multiple sources, it is not always specified in the documentation exactly which documents were sourced from where. Additionally, documents often go through several stages of corpus selection and extension. Lack of precise documentation makes it often difficult to trace back the complete history of corpus development and document overlap.

To obtain an overview of document intersections for the corpora mentioned in Section 3 we decided to use document identifiers. However, we found that a direct comparison of the identifiers did not reveal all intersections because documents are often renamed to match different naming conventions. Therefore, we first mapped the filenames for each of the corpora to uniform identifiers. These uniform identifiers are all fully capitalized and do not contain file extensions. Directory names are excluded, unless they are needed for disambiguating files. For example, `WSJ/00/WSJ_0006.MRG` (Treebank-3) and `wsj_0006.txt` (FactBank) are both mapped to the uniform filename `WSJ_0006`. Some corpora required some more work than capitalization and stripping extensions and prefixes. For example, all annotations of the AMR corpus are collected in one file per data collection containing the identifiers of the source documents as meta-information for each annotation, and OntoNotes has renamed its files to simpler filenames (e.g., `NBC20001003.1830.0755` was renamed to `nbc_0001`), but we could use the mapping files specifying the original filenames that are provided in its release for each of its data collections.

Figure 2[8] visualizes the document intersections that we found, revealing the complex network resulting from more than a decade of data selection and extension.[9] To begin with, we observe the expected intersections as discussed above: 480 documents are shared between ECB 1.0 and its extension ECB+, 1,728 documents between PropBank and its successor OntoNotes, and 132 WSJ documents between Treebank, PropBank, NomBank, TimeBank, TempEval-3 and FactBank (one of which also occurs in FrameNet). It also reveals some less obvious intersections, e.g. 20 documents between the RED and AMR corpora. There is only one corpus that completely stands on its own: EventStatus. Finally, it also reveals unexpected lack of overlap between some corpora. For example, OntoNotes and TimeBank/FactBank are partially built on top of PropBank, but they do not share any documents with each other. It appeared that 25% of PropBank was not carried over into OntoNotes, i.e. a set of documents that were considered too

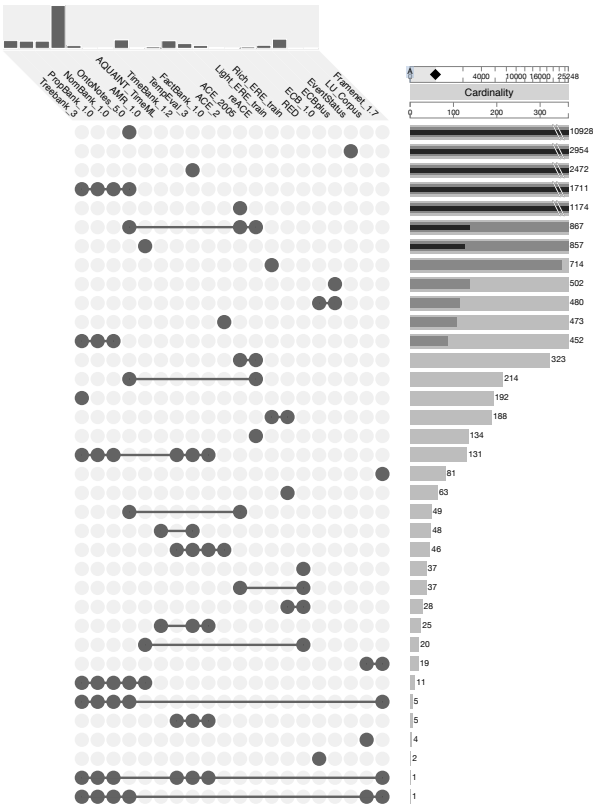


Figure 2: Document intersections of all event corpora

domain-specific because of their strictly financial content. The documents in TimeBank/FactBank sourced from PropBank were all part of this subset. The document intersections reveal several opportunities for merging and comparing annotations across corpora.

5 Document Interoperability: Text

To emphasize the interoperability challenges identified at the text and annotations levels, we performed a small-scale analysis on a subset of corpora, namely PropBank/NomBank (PB/NB), FactBank 1.0 (FB) and TempEval-3 (TE3). For TE3, we only consider the TBAQ-cleaned subcorpus[10] as described in (UzZaman et al., 2013). TBAQ-cleaned corresponds to the complete TimeBank and the complete AQUAINT TimeML Corpus (A-TimeML) with revised event annotations, FB contains the complete TimeBank and part of A-TimeML with their original event annotations, and PB/NB corresponds to the verbal/nominal predicates annotated in the WSJ section of Treebank-3. Therefore, we indirectly analyse these corpora as well. Figure 3 summarizes the document intersections of these three corpora and related ones.

Table 1 summarizes the content of the three datasets. PB/NB is the largest corpus in terms of number of documents (2,312), number of sentences (49,208), number of tokens (1,174,165) and also number of events (227,491). FB consists of 208 documents which are split into 3,839 sentences and 77,231 tokens and contains 9,492 events. TE3

[8] Figures 2 and 3 were created using UpSet (Lex et al., 2014), see: http://caleydo.org/tools/upset

[9] An interactive and more detailed version of the UpSet visualization can be viewed by following the instructions at https://github.com/cltl/CorpusComparison.

[10] https://www.cs.york.ac.uk/semeval-2013/task1/

Event Dataset	#Documents	Total # Sentences	Total # Tokens	# Sentences with Events	# Sentences without Events	Total # Events	Avg. #Events / Sentence
PB/NB	2,312	49,208	1,174,165	47,394	1,814	227,491	4.79
FB	208	3,839	77,236	2,807	1,032	9,492	3.38
TE3	256	3,955	99,384	3,604	351	11,129	3.08

Table 1: Content overview of selected event corpora

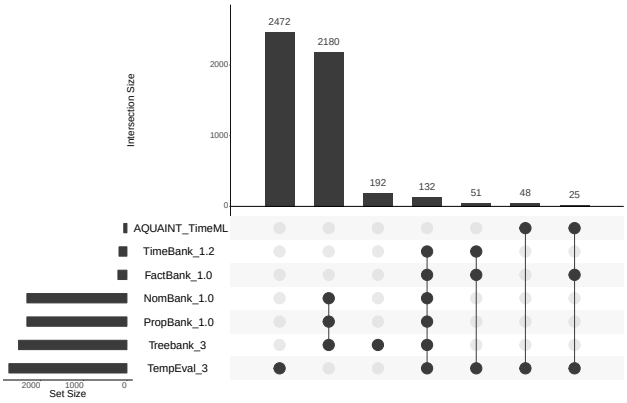


Figure 3: Document intersections of selected corpora

consists of 256 documents, 3,955 sentences[11] and about 100k tokens of which 11,129 have been annotated as events. With respect to the text-level interoperability among these three datasets, we can mention that while PB/NB and TE3 contain only sentences from the article body, FB includes metadata such as the document id, its creation date and its title as the first sentences of the text. These sentences account for the major part of the large number of sentences without annotated events in FB - 1,032 (26.88%), as opposed to 1,814 (3.69%) in PB/NB and 351 (8.87%) in TE3. TE3 has the lowest number of events annotated per sentence, i.e. 3.08 compared to 3.38 in FB and 4.79 in PB/NB.

6 Document Interoperability: Annotations

Chiarcos (2012a) and Chiarcos et al. (2013) define structural interoperability as annotations of different origin being represented using the same formalism, such that different resources can be processed in a uniform way and that their information can be easily merged. Following Ide and Pustejovsky (2010), they define conceptual interoperability as "the ability to automatically interpret exchanged information meaningfully and accurately in order to produce useful results" which can be achieved by linking annotations of different origin to a common vocabulary. In Section 6.1 we summarize the structural interoperability issues that we encountered for the three datasets. In Section 6.2 we review their conceptual interoperability by empirically comparing their annotations of events. This involves a direct comparison between the aligned annotations in FB and TE3 on the basis of their event identifiers, and a type-based analysis, where we abstract away from the annotations in context and provide a more general overview of the annotated types (i.e. POS, lemmas) in all three datasets.

[11] We used Stanford CoreNLP (Manning et al., 2014) in order to split the documents into sentences.

6.1 Structural Interoperability

The data in PB/NB has been released in different formats. Originally, the semantic role annotations were represented with PropBank pointers, i.e. stand-off annotations pointing to locations in the parse tree in Treebank, which in turn was represented by simple labelled brackets in a text file. PropBank pointers are only useful in combination with the corresponding tree structures, since they include empty elements such as traces in the token count. We loaded the data using NLTK.[12] As part of OntoNotes, the data was also released in CoNLL-format, with each line representing a single word with a series of tab-separated fields.[13] However, as mentioned before, OntoNotes does not contain the complete PB/NB. FB also uses a stand-off annotation format and represents the data through a set of 20 tables. For example, one table contains all linguistic information relative to each token (e.g. token id, POS tag), one contains all information relative to each event (e.g. event identifier, text), and one contains the factuality degree values assigned to each event. Finally, TE3 uses the TimeML XSD schema,[14] with XML elements for representing metadata (e.g. <DOCID>, <EXTRAINFO>), the main body of the annotated text (<TEXT> with in-line <EVENT> annotations) and event instances (<MAKEINSTANCE> with tense/aspect of events as XML attributes).

Merging and comparing the annotations of the three datasets is not straightforward due to these structural differences and the textual differences mentioned in Section 5. Both PB/NB and FB use {DOC_ID, SENT_ID, TOKEN_ID} to identify the position of an event in text, but since FB includes the document id, creation date and title as part of the text and PB/NB includes empty elements in the token count, there are mismatches in the sentence and token identifiers. Although the in-line annotations of TE3 do not allow for direct comparison with other datasets, the event identifiers can be used for comparison in this matter.

To analyse the conceptual interoperability of the annotations, we converted all three corpora to CoNLL-format with each line representing a token and information about its document id, sentence id, token id, token text, lemma and POS. If possible, we used the gold sentence splitting, tokenization, lemmatization and POS tagging. If that was not available (e.g. TE3 only has POS tags for most but not all events, PB/NB only has lemmas for events), we used the Stanford CoreNLP pipeline to retrieve the POS tags and the lemma of all the tokens in the datasets.

[12] http://www.nltk.org/howto/propbank.html

[13] https://github.com/propbank/propbank-release

[14] http://timeml.org/timeMLdocs/TimeML1.2.1.xsd

	Verb	Noun	Adjective	Preposition	Number	Adverb	Particle	Determiner	Oth/Unknown
PB/NB	114,574	109,793	-	-	-	-	-	-	-
FB	6,377	2,498	250	45	46	21	6	2	1
TE-3	5,835	2,451	202	10	-	-	-	-	2,632

Table 2: Distribution of POS tags across annotated single-token events

6.2 Conceptual Interoperability

The event annotations in both FB and TE3 are based on the TimeML 1.2.1 Annotation Guidelines (Saurí et al., 2006), which define an event as a situation that happens or occurs. The guidelines further specify in which cases an event should or should not be annotated as a (separate) event. For example, generic events should never be tagged and causative predicates only in specific cases. Thus, TimeML defines events primarily from a semantic point of view, and allows the annotation of all linguistic realizations, including verbs, nominalizations, adjectives, predicative clauses, or prepositional phrases. Whereas the events in FB are the original ones from TimeBank and A-TimeML, those in TE3 are a result of multiple revisions. UzZaman et al. (2013) mention that TE3 borrowed the events from TempEval-2 and added missing events, but no example is given. Verhagen et al. (2010) mention that all event annotations for TempEval-2 were reviewed to make sure that it complied with the latest guidelines, i.e. a simplified version of the TimeML guidelines (Saurí et al., 2009).

In contrast, PB/NB does not take semantics as a starting point, but the syntactic categories of verbs and nouns. PropBank annotates the arguments and adjuncts of each verb with their semantic roles. NomBank does the same for nouns, but defines semantic restrictions with respect to which noun phrases and which constituents of noun phrases are *markable*. For instance, the head noun must be of a "propositional" type (representing an event, state, etc.) and the noun phrase must contain at least one argument and one "proposition-modifying" adjunct (Meyers, 2007).

The semantic and syntactic requirements for annotating events are thus slightly different for TimeML and PB/NB. Another difference is the extents (i.e. span of tokens) of events. In PB/NB, the single noun or verb is annotated as the event, but for phrasal verbs the particle is concatenated with the verb to form a single predicate lemma (Bonial et al., 2010). TimeML implements the notion of minimal chunk, i.e. only the head of the constituent should be annotated and not the whole phrase. As opposed to PB/NB, only the verbal part (and not its particle) of a phrasal verb is marked as event. In the early TimeML guidelines, an exception was made for exocentric elements (i.e. if it has no single head), in which case the entire expression was to be marked (e.g. *on board*). For TempEval-2, however, the annotators always had to annotate only the head.

6.2.1 Overlap of Annotations

As mentioned in Section 6.1, alignment of the annotations of PB/NB with the other datasets is not straightforward due to textual and structural differences. We did, however, perform an analysis on the overlap of event annotations in FB and TE3 for their 208 shared documents by aligning them on the basis of their unique identifiers, i.e. the combination of {DOC_ID, EVENT_ID}. The overlap was 8,227 events out of a total of 9,492 in FB and 8,248 in TE3. The revised annotations of TE3 included 24 new events, while 1,265 events from FB were removed. First of all, we observe that all events annotated in TE3 indeed consist of a single token. There were 4 multi-token events that were changed into single-token events: *coup d'etat* > *coup*, *March for Life* > *March* and *World War II* > *War* (2 occurrences). All other multi-token events in FB, 241 in total, were removed. Most of them (238) contained a cardinal number (e.g. *$4.375 a share* and *about 12%*). From the single-token events that were removed (1,265), 46 also consisted of a cardinal number. The rest of the single-token events consisted mainly of nouns (713), verbs (138) and adjectives (82). The 24 new events consisted of 11 nouns, 12 verbs and 1 adjective. Surprisingly, many of both the removed and the newly added events seemed to correspond to the 'source-introducing predicates' as defined in (Saurí and Pustejovsky, 2009), e.g. *say*, *think*, *statement*, *plan*, *confident*, which were however also represented in the shared set of events. This may simply be a result of inconsistency (see also Section 6.3). In sum, the revisions of event annotations mainly seemed to concern the removal of 'quantitative statements' (Saurí et al., 2006) and multi-token events, but we were not able to find other clear patterns in the many events that were lost in the transition from FB to TE3.

6.2.2 Part-of-Speech Distribution of Events

In Table 2 we show the POS tag distribution for every single-token event in the three datasets. We observe that for all three datasets the majority of the single-token event mentions are either verbs or nouns. While in PB/NB only verbs and nouns are annotated as events, in TE3 and FB there are also adjectives and prepositions that stand as events, but in a smaller proportion. Furthermore, FB contains also event mentions of type number, adverb, particle and determiner. In the TE3 dataset, a considerable amount of event mentions do not have a POS tag assigned but are marked as 'Other' or 'Unknown' instead.

Next, we looked at the multi-token events in PB/NB and FB. We observe that there is no consistency between the two datasets. While PB/NB contains multi-token events composed of verbs in combination with particles, adverbs, prepositions, pronouns, adjectives and nouns, FB has no such event mentions. As mentioned before, the majority of the multi-token events in FB are combinations of numerals and symbols, under various patterns. Furthermore, in PB/NB all the multi-token events have exactly two tokens, but in FB 23 out of 246 events have more than 2 tokens.

6.2.3 Overview of Event Tokens and Lemmas

In Table 4 we show the number of event tokens, the number of distinct event tokens and the number of distinct event

		Verb	Noun	Adjective	Preposition	Adverb	Number	Particle	Determiner
PB/NB	Total Tokens	153,721	355,099	83,341	122,957	39,713	45,046	2,990	100,727
	Events	110,291	115,051	2,107	496	302	2	2,320	2
	(%)	71.74%	32.34%	2.52%	0.40%	0.76%	0.004%	77.59%	0.001%
FB	Total Tokens	10,180	24,208	5,032	8,131	2,257	3,854	194	6,711
	Events	6,386	2,546	240	50	29	291	3	9
	(%)	62.73%	10.51%	4.76%	0.61%	1.28%	7.55%	1.54%	0.13%
TE3	Total Tokens	13,643	30,290	6,246	10,706	3,469	3,652	242	8,843
	Events	8,536	2,391	173	17	12	-	-	-
	(%)	62.56%	7.89%	2.76%	0.15%	0.34%	-%	-%	-%

Table 3: Overview of tokens annotated as event per POS

lemmas. For every event token and event lemma we also consider their POS tag. Because FB and TE3 do not contain information regarding the lemma of the tokens nor events and PB/NB only for the events (their rolesets), we compute these statistics based on the output of Stanford CoreNLP. While PB/NB has the lowest ratio of distinct event tokens (7.2%) and event lemmas (4.43%), FB has the highest ratios, around 32% for tokens and around 22% for lemmas. This high number of distinct event instances is due to the fact that many event mentions contain numerals which do not repeat throughout the dataset.

Dataset	Total #Events	Distinct #Event Tokens	Distinct #Event Lemmas
PB/NB	227,491	16,398	10,089
FB	9,492	3,041	2,171
TE3	11,129	2,883	1,871

Table 4: Overview of distinct event tokens and lemmas

6.3 Consistency of Annotations

In this section we analyse the consistency of event annotations. We analyse the consistency at the level of the POS tags (Section 6.3.1) and at the level of event token and event lemma (Section 6.3.2). This analysis is a work-in-progress performed in the context of the CrowdTruth[15] project (Inel et al., 2014), which shows that crowdsourcing is a feasible method to identify and correct inconsistent annotations (Inel and Aroyo, 2017; Aroyo and Welty, 2012).

6.3.1 Evaluation of Event POS Tags

Table 3 shows the total amount of tokens for each of the following POS tags: verb, noun, adjective, adverb, preposition, number, particle, determiner and symbol, as well as the total amount and percentage of tokens annotated as events for each POS tag. Overall, the tokens categorized as verbs have the highest coverage as events, as more than 62% of those have been annotated as events across the three datasets. For all three datasets, the verbs that were not marked as events were mostly the verbs *be*, *have* and *do*, which we assume to be those occurrences where they act as auxiliaries (91%, 65% and 68% of non-annotated verbs in PB/NB, FB and TE3 respectively). However, we also found some surprising cases. For example, *televise* or *bless* are not annotated as events in TE3 and FB, but they are annotated as events in PB/NB, and occurrences of *say* are often not annotated in all three corpora.

The nouns annotated as events have a lower coverage (between 7.89% and 32.34%) and this coverage varies a lot for each dataset. The coverage of adjectives, prepositions and adverbs is quite similar on the three datasets, as shown in Table 3. However, we know that for PB/NB these are all part of phrasal verbs (*cut loose, dig up*), which also accounts for the high coverage of particles in PB/NB. In contrast, adjectives, prepositions and adverbs can act as independent events in FB and TE3 if they have a propositional meaning (*optimistic*, *down*, *in place*). As we have already discussed, only the events in FB cover numbers, but their coverage is still quite low, around 7.5%.

6.3.2 Evaluation of Event Tokens and Lemmas

In Table 5 we present the overview of inconsistencies encountered at the level of annotated event token and event lemma. For each event token and each event lemma we count how many times it appears in the dataset and how many times it was annotated as event. Based on these occurrences, we compute how many times a single event token or event lemma was not annotated, and for how many distinct single event tokens and event lemmas there are instances in the corpus which are not annotated.

Dataset	Event Tokens		Event Lemmas	
	Total (%)	Distinct (%)	Total (%)	Distinct (%)
PB/NB	146,268 39.46%	6,462 42.97%	178,253 44.27%	4,914 53.3%
FB	11,711 55.88%	1,016 36.04%	12,737 57.94%	921 47.25%
TE3	12,580 53.06%	973 33.74%	13,473 54.76%	837 44.73%

Table 5: Inconsistencies at the level of single-token events: event tokens and lemmas

We first observe that there are inconsistencies at the token level since not all instances of an event are always annotated as such. For example, in TE3 *decision* (noun) is annotated as event in 45 out of 53 cases, *embargo* (noun) in 7 out of 13 cases, and *said* (verb) in 993 out of 1,006 cases. Further-

[15] www.crowdtruth.org

more, there are also inconsistencies at the lemma level since not all lemma instances of an event are always annotated as events. For example, in TE3 *disaster* (noun) is annotated as event in 3 out of 7 lemma-based occurrences, *war* (noun) in 32 out of 52 cases, and *export* (noun) in 1 out of 5. Across all three datasets, the total amount of inconsistencies at the level of event lemma is higher than the total amount of inconsistencies at the level of event token which means that only particular forms of a lemma are usually annotated as events. Further, we acknowledge the fact that TE3 shows the least amount of inconsistencies for distinct event tokens and lemmas, although they are still substantial. We believe this is due to the fact that the annotations in the dataset have been revised multiple times.

Many multi-token events in FB are composed of numbers in combination with symbols (*#*), currencies (*$*, *us$*, *c$*) and percentages (%). It is interesting to observe, however, that in cases where the sign is replaced by the word, i.e. *5 percent* instead of *5%*, only the number is annotated as event (we noticed 3 such cases). We also observe that only cardinal numbers are annotated as event, while numbers such as *million* or *billion* are never annotated. We further noticed the following annotation inconsistencies: *# CD* was annotated only 3/23 times as events, *$CD* was annotated only 216/819 times and *CD%* was annotated only 21/304 times. Furthermore, 6 times only the cardinal was annotated as event in constructions of type *c$CD*. While the TimeML guidelines do specify that quantitative statements should only be annotated in case "their validity is relative to the point in time they have been asserted" (Saurí et al., 2006), we hypothesize that inconsistency may have been one of the reasons that they do not occur in TE3 any more.

Lastly, we checked the multi-token events in PB/NB. At the token level, 527 times a phrase that was previously annotated as a multi-token event was not annotated as such. In 475 out of these 527 cases, only the head of the phrase was annotated as a single-token event; the remainder was not annotated at all. At the level of lemmas, 834 instances of multi-token events were missed, including 784 cases where only the head was annotated. It is not easy to determine which of these cases indicate actual syntactic or semantic differences, and which indicate inconsistencies. For example, the combination *back off* is annotated twice as belonging to the roleset back.11,[16] meaning "to retreat from" (e.g. in WSJ_1000-S4: *big securities firms backed off from program trading*). However, we found 5 other occurrences of *back off* that were not annotated as such, 3 of which were incorrectly assigned to another roleset. Consider the two sentences below; in both cases, the verb *back* was annotated as belonging to roleset back.02, meaning "move backwards". In Sentence 1 this is the correct interpretation, but in Sentence 2 back.11 would have been the correct roleset.

1. [...] creditors committee *backed* off a move to come up with its own alternative proposals [...] (WSJ_0475-S0)
2. Previously, he noted, gold producers tended to *back* off from a rising gold market [...] (WSJ_2045-S22)

[16]https://verbs.colorado.edu/propbank/framesets-english

7 Conclusion

Interoperability of resources has been discussed extensively in many standardisation initiatives and meta-model proposals. However, the practical reality is very far from the ideal solutions that have been proposed. In this paper, we provided a detailed and comprehensive analysis of the incompatibilities that are de facto for a selection of 20 text corpora with event data. Our analysis emphasizes the need for resource interoperability in order to facilitate merging similar or complementary event annotations in corpora that overlap in content and hence, to exploit research opportunities.

We provided a comprehensive overview of the document intersections among a representative set of event-centric textual corpora, which is the stepping stone in advancing the resource interoperability process. Furthermore, we have illustrated through empirical analysis various type-level annotation inconsistencies in and across a subset of these event corpora. Based on this study, we conclude that corpora should provide the means to link and align them to other corpora at different levels, for which we propose the following best-practice guidelines:

Metadata: Documents should be named with unified standards (persistent identifiers) and be provided with mappings to documents in other corpora.

Textual content: If corpora share files, their content should be aligned to facilitate the merging of annotations.

Structural: The community should agree on a data structure to represent annotations that capture their provenance, especially when they are performed on the same content.

Conceptual: Preferably, the community should agree on (the interpretation of) their labels. Alternatively, or additionally, basic statistics on the total number of tokens, words, sentences, number of annotated units per type of annotation and coverage of annotation per lemma and POS, released together with the annotated corpus, would help to properly evaluate and understand the consistency of the annotations within the corpus and their compatibility with similar annotations in other corpora. Any revisions should be sufficiently documented.

Ultimately, the community is responsible for reaching consensus on how to publish and distribute resources in the future. We hope that the availability of tools for corpus and annotation analysis and aggregation will stimulate releasing the resources in a more consistent and transparent way. In the future, researchers should spend less time on conversion and mapping of data and more time on conceptually understanding the content of the annotations.

8 Acknowledgement

We would like to thank the anonymous reviewers for their useful feedback, everyone who kindly answered our questions and provided us with the information necessary for our analysis, and all researchers for the hard work that was put into creating the valuable resources cited in this paper. The work presented in this paper was funded by the Amsterdam Data Alliance in the QuPiD project, by the Netherlands Organization for Scientific Research (NWO) via the Spinoza grant, awarded to Piek Vossen in the project "Understanding Language by Machines", and by CLARIAH-CORE project financed by NWO (www.clariah.nl).

9 Bibliographical References

Aguilar, J., Beller, C., McNamee, P., Van Durme, B., Strassel, S., Song, Z., and Ellis, J. (2014). A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 45–53.

Aroyo, L. and Welty, C. (2012). Harnessing disagreement for event semantics. *Detection, Representation, and Exploitation of Events in the Semantic Web*, 31.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Palmer, M., and Schneider, N. (2013). Abstract Meaning Representation for Sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse (LAW VII & ID)*, pages 178–186.

Bejan, C. A. and Harabagiu, S. M. (2008). A linguistic resource for discovering event structures and resolving event coreference. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, pages 2881–2887.

Bonial, C., Babko-Malaya, O., Choi, J. D., Hwang, J., and Palmer, M., (2010). *PropBank Annotation Guidelines, version 3.0*. Center for Computational Language and Education, Research Institute of Cognitive Science, University of Colorad at Boulder, December.

Calzolari, N., Del Gratta, R., Francopoulo, G., Mariani, J., Rubino, F., Russo, I., and Soria, C. (2012). The LRE Map. Harmonising Community Descriptions of Resources. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 1084–1089.

Chiarcos, C. and Erjavec, T. (2011). OWL/DL formalization of the MULTEXT-East morphosyntactic specifications. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 11–20.

Chiarcos, C., McCrae, J., Cimiano, P., and Fellbaum, C. (2013). Towards open data for linguistics: Linguistic linked data. In *New Trends of Research in Ontologies and Lexical Resources*, pages 7–25.

Chiarcos, C. (2012a). Interoperability of corpora and annotations. In *Linked Data in Linguistics*, pages 161–179.

Chiarcos, C. (2012b). POWLA: Modeling linguistic corpora in OWL/DL. In *Extended Semantic Web Conference*, pages 225–239.

Cunningham, H. (2002). GATE, a general architecture for text engineering. *Computers and the Humanities*, 36(2):223–254.

Cybulska, A. and Vossen, P. (2014). Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 4545–4552.

Diab, M. T., Levin, L., Mitamura, T., Rambow, O., Prabhakaran, V., and Guo, W. (2009). Committed belief annotation and tagging. In *Proceedings of the 3rd Linguistic Annotation Workshop (ACL-IJCNLP 2009)*, pages 68–73.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The Automatic Content Extraction (ACE) Program - Tasks, Data, and Evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*.

Farrar, S. and Langendoen, D. T. (2003). A linguistic ontology for the semantic web. *GLOT International*, 7(3):97–100.

Ferrucci, D. and Lally, A. (2004). UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to FrameNet. *International Journal of Lexicography*, 16(3):235–250.

Fokkens, A., Soroa, A., Beloki, Z., Ockeloen, N., Rigau, G., van Hage, W. R., and Vossen, P. (2014). NAF and GAF: Linking linguistic annotations. In *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 9–16.

Hachey, B., Grover, C., and Tobin, R. (2012). Datasets for generic relation extraction. *Natural Language Engineering*, 18(1):21–59.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *International Semantic Web Conference*, pages 98–113.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: the 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL (Short Papers)*, pages 57–60.

Huang, R., Cases, I., Jurafsky, D., Condoravdi, C., and Riloff, E. (2016). Distinguishing Past, On-going, and Future Events: The EventStatus Corpus. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 44–54.

Ide, N. and Pustejovsky, J. (2010). What does interoperability mean, anyway? Toward an operational definition of interoperability for language technology. In *Proceedings of the 2nd International Conference on Global Interoperability for Language Resources*.

Ide, N. and Romary, L. (2004). International standard for a linguistic annotation framework. *Natural language engineering*, 10(3-4):211–225.

Ide, N. and Suderman, K. (2007). GrAF: A graph-based format for linguistic annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8.

Inel, O. and Aroyo, L. (2017). Harnessing diversity in crowds and machines for better NER performance. In *European Semantic Web Conference*, pages 289–304.

Inel, O., Khamkham, K., Cristea, T., et al. (2014). Crowdtruth: Machine-human computation framework for harnessing disagreement in gathering annotated data. In *ISWC*, pages 486–504.

Ishida, T. (2006). Language Grid: An infrastructure for intercultural collaboration. In *Proceedings of the 2005 Symposium on Applications and the Internet (SAINT'06)*.

Krauwer, S. and Hinrichs, E. (2014). The CLARIN Research Infrastructure: Resources and Tools for eHumanities Scholars. In *Proceedings of the 9th Interna-*

tional Conference on Language Resources and Evaluation (LREC 2014), pages 1525–1531.

Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500.

Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R., and Pfister, H. (2014). UpSet: Visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '14)*, 20(12):1983–1992.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330.

McCrae, J. P. and Cimiano, P. (2015). Linghub: a Linked Data Based Portal Supporting the Discovery of Language Resources. In *Proceedings of the 11th International Conference on Semantic Systems*, volume 1481, pages 88–91.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The NomBank project: An interim report. In *HLT-NAACL 2004 workshop: Frontiers in corpus annotation*, volume 24, page 31.

Meyers, A. (2007). *Annotation Guidelines for NomBank – Noun Argument Structure for PropBank*. New York University.

O'Gorman, T., Wright-Bettner, K., and Palmer, M. (2016). Richer Event Description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of 2nd Workshop on Computing News Storylines*, pages 47–56.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Palmer, M., Bonial, C., and McCarthy, D. (2014). SemLink+: FrameNet, VerbNet and Event Ontologies. In *Proceedings of Frame Semantics in NLP: A Workshop in Honor of Chuck Fillmore (1929-2014)*, pages 13–17.

Piperidis, S. (2012). The META-SHARE Language Resources Sharing Infrastructure: Principles, Challenges, Solutions. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 36–42.

Pradhan, S. S., Ramshaw, L., Weischedel, R., MacBride, J., and Micciulla, L. (2007). Unrestricted coreference: Identifying entities and events in OntoNotes. In *International Conference on Semantic Computing, 2007 (ICSC 2007)*, pages 446–453.

Pustejovsky, J., Castano, J. M., Ingria, R., Saurí, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003a). TimeML: Robust specification of event and temporal expressions in text. In *Proceedings of the 5th International Workshop on Computational Semantics (IWCS)*.

Pustejovsky, J., Hanks, P., Saurí, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., and Lazo, M. (2003b). The TIMEBANK corpus. In *Corpus linguistics*, volume 2003, pages 647–656.

Pustejovsky, J., Meyers, A., Palmer, M., and Poesio, M. (2005). Merging PropBank, NomBank, TimeBank, Penn Discourse Treebank and Coreference. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 5–12.

Saurí, R. and Pustejovsky, J. (2009). FactBank: A corpus annotated with event factuality. *Language resources and evaluation*, 43(3):227–268.

Saurí, R., Littman, J., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006). TimeML annotation guidelines, version 1.2.1.

Saurí, R., Goldberg, L., Verhagen, M., and Pustejovsky, J. (2009). *Annotating Events in English: TimeML Annotation Guidelines (Version TempEval-2010)*. Brandeis University.

Simons, G. and Bird, S. (2003). The Open Language Archives Community: An infrastructure for distributed archiving of language resources. *Literary and Linguistic Computing*, 18(2):117–128.

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In *Proceedings of the 3rd Workshop on EVENTS at the NAACL-HLT*, pages 89–98.

Strassel, S., Przybocki, M. A., Peterson, K., Song, Z., and Maeda, K. (2008). Linguistic resources and evaluation techniques for evaluation of cross-document automatic content extraction. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*.

UzZaman, N., Llorens, H., Derczynski, L., Verhagen, M., Allen, J., and Pustejovsky, J. (2013). SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: 7th International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9.

Verhagen, M., Saurí, R., Caselli, T., and Pustejovsky, J. (2010). SemEval-2010 task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 57–62.

Werner, G. J., Prabhakaran, V., Diab, M., and Rambow, O. (2015). Committed belief tagging on the FactBank and LU corpora: A comparative study. In *Proceedings of the 2nd Workshop on Extra-Propositional Aspects of Meaning in Computational Semantics (ExProM 2015)*, pages 32–40.

Windhouwer, M. and Wright, S. E. (2012). Linking to linguistic data categories in ISOcat. In *Linked Data in Linguistics*, pages 99–107.

10 Language Resource References

Bejan, C. and Harabagiu, S. (2010). *EventCorefBank (ECB)*. Human Language Technology Research Institute.

Brandeis University. (2008). *AQUAINT TimeML Corpus*. TimeML corpora, ELRA-U-W0394.

Cybulska, A. and Vossen, P. (2014). *The ECB+ Corpus*. The NewsReader Project.

Diab, M., Dorr, B., Levin, L., Mitamura, T., Passonneau, R., Rambow, O., and Ramshaw, L. (2009). *Language Understanding Annotation Corpus*. Linguistic Data Consortium, LDC2009T10, ISRLN 775-964-514-342-7.

Hachey, B., Grover, C., and Tobin, R. (2011). *Datasets for Generic Relation Extraction (reACE)*. The University of Edinburgh, distributed via Linguistic Data Consortium, LDC2011T08, ISLRN 494-554-511-556-5.

Huang, R., Jurafsky, D., and Riloff, E. (2017). *The EventStatus Corpus*. Linguistic Data Consortium, LDC2017T09, ISLRN 173-931-115-382-5.

ICSI Berkeley. (2017). *FrameNet 1.7*. FrameNet Project.

Knight, K., Baranescu, L., Bonial, C., Georgescu, M., Griffitt, K., Hermjakob, U., Marcu, D., Palmer, M., and Schneider, N. (2014). *Abstract Meaning Representation (AMR) Annotation 1.0*. Linguistic Data Consortium, LDC2014T12, ISLRN 637-196-362-554-6.

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., and Taylor, A. (1999). *Treebank-3*. The Penn Treebank (PTB) project, distributed via Linguistic Data Consortium, LDC99T42, ISLRN 141-282-691-413-2.

Meyers, A., Reeves, R., and Macleod, C. (2008). *NomBank v 1.0*. The NomBank Project, distributed via Linguistic Data Consortium, LDC2008T23, ISLRN 652-357-402-514-3.

Mitchell, A., Strassel, S., Przybocki, M., Davis, J., Doddington, G., Grishman, R., Meyers, A., Brunstein, A., Ferro, L., and Sundheim, B. (2003). *ACE-2 Version 1.0*. The ACE Program, distributed via Linguistic Data Consortium, LDC2003T11, ACE corpora, ISLRN 498-363-793-174-9.

O'Gorman, T. and Palmer, M. (2016). *Richer Event Description 1.0*. Linguistic Data Consortium, LDC2016T23, ISLRN 722-524-976-246-2.

Palmer, M., Kingsbury, P., Babko-Malaya, O., Cotton, S., and Snyder, B. (2004). *Proposition Bank I*. The PropBank Project, distributed via Linguistic Data Consortium, LDC2004T14, ISLRN 874-058-423-080-1.

Parker, R., Graff, D., Kong, J., Chen, K., and Maeda, K. (2011). *English Gigaword Fifth Edition*. Linguistic Data Consortium, LDC2011T07, ISLRN 911-942-430-413-0.

Pustejovsky, J., Verhagen, M., Saurí, R., Littman, J., Gaizauskas, R., Katz, G., Mani, I., Knippen, R., and Setzer, A. (2006). *TimeBank 1.2*. Brandeis University, distributed via Linguistic Data Consortium, LDC2006T08, TimeML corpora, ISLRN 717-712-373-266-4.

Saurí, R. and Pustejovsky, J. (2009). *FactBank 1.0*. Linguistic Data Consortium, LDC2009T23, ISLRN 293-001-569-603-0.

Walker, C., Strassel, S., Medero, J., and Maeda, K. (2006). *ACE 2005 Multilingual Training Corpus*. The ACE Program, distributed via Linguistic Data Consortium, LDC2006T06, ACE corpora, ISLRN 458-031-085-383-4.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., El-Bachouti, M., Belvin, R., and Houston, A. (2013). *OntoNotes Release 5.0*. OntoNotes Project, distributed via Linguistic Data Consortium, LDC2013T19, ISLRN 151-738-649-048-2.

Parsivar: A Language Processing Toolkit for Persian

Salar Mohtaj[1], **Behnam Roshanfekr**[2], **Atefeh Zafarian**[3], **Habibollah Asghari**[4]
[1,2,3,4] ICT Research Institute, Academic Center for Education Culture and Research, Tehran Iran
{salar.mohtaj, b.roshanfekr, atefeh.zafarian, habib.asghari}@ictrc.ac.ir

Abstract

With the growth of Internet usage, a massive amount of textual data is generated on social media and the Web. As the text on the Web are generated by different authors with various types of writing styles and different encodings, a preprocessing step is required before applying any NLP task. The goal of preprocessing is to convert text into a standard format that makes it easy to extract information from documents and sentences. Moreover, the problem is more acute when we deal with Arabic script-based languages, in which there are some different kinds of encoding schemes, different kinds of writing styles and the spaces between or within the words.

This paper introduces a preprocessing toolkit named as Parsivar, which is a comprehensive set of tools for Persian text preprocessing tasks. This toolkit performs various kinds of activities comprised of normalization, space correction, tokenization, stemming, parts of speech tagging and shallow parsing. To evaluate the performance of the proposed toolkit, both intrinsic and extrinsic approaches for evaluation have been applied. A Persian plagiarism detection system has been exploited as a downstream task for extrinsic evaluation of the proposed toolkit. The results have revealed that our toolkit outperforms the available Persian preprocessing toolkits by about 8 percent in terms of F1.

Keywords: Preprocessing Tasks, Natural Language Processing, Language Processing Toolkit

1. Introduction

Preprocessing is one of the essential steps in Natural Language Processing (NLP) tasks that convert unstructured texts into a standard text format suitable for NLP applications. This makes it easy to extract knowledge from documents and sentences. Despite of its importance, few efforts have been accomplished on developing preprocessing tools in low-resource languages. Development of preprocessing tools is more challenging in Arabic script-based languages like Persian, in which there are some difficulties including the lack of enough linguistic resources, various character encodings, spaces between or within multi-token words and a wide variety of suffixes.

In this paper, we introduce Parsivar, a Persian preprocessing toolkit that includes a set of tools necessary for different NLP tasks. It normalizes Persian texts to a standard format, corrects the spacing between or within the words, tokenizes words and sentences, extracts word stems, tags the words with their Part of Speech (PoS) and finally performs shallow parsing on sentences. For space correction, it also proposes two different kinds of solutions. The first solution is based on some pre-defined rules and the second one is based on learning methods. The evaluation results have revealed that the proposed toolkit outperforms the other existing preprocessing toolkits in Persian.

The rest of the paper is organized as follows: Section 2 discusses the previous works in the field of Persian text preprocessing. In Section 3 we describe the algorithms for implementing each of the modules in Parsivar toolkit in detail. In section 4, we present the experimental results for evaluating and comparing existing Persian preprocessing toolkits with respect to our approach. For the purpose of benchmarking the algorithms, we evaluate them on the performance of a Persian plagiarism detection application using extrinsic evaluation. Finally, in the last section, conclusion and recommendations for future works will be described.

2. Related work

Due to the importance of preprocessing in NLP applications, some attempts have been done to develop integrated Persian preprocessing packages in recent years. In 2010, Shamsfard et al. (Shamsfard, Sadat Jafari and Ilbeygi, 2010) proposed STeP1, which includes a combination of tools such as tokenization, morphological analysis and a POS tagging. The ParsiPardaz toolkit was proposed by Sarabi et al. (Sarabi, Mahyar and Farhoodi, 2013) which provides the STeP1 capabilities along with some other tasks like normalization and spell checking. One of the issues related to the mentioned toolkits is that they are not publicly available as open source applications. Sobhe (Hazm, 2014) introduced Hazm, an open source preprocessing toolkit which includes some major tasks such as normalization, tokenization and POS tagging. Although Hazm outperforms STeP1 and ParsiPardaz toolkits from the run time point of view, its output results are not as accurate as them.

The proposed preprocessing toolkit in this paper provides different kinds of tasks including normalization, tokenization, stemming, POS-tagging and shallow parsing. Moreover, Parsivar is publicly available on the web for research purposes[1]. We have compared the performance of our toolkit with ParsiPardaz and Hazm in an extrinsic evaluation platform using a Persian plagiarism detection algorithm. The results showed that our toolkit has a higher performance when compared with the other toolkits while its runtime is near to Hazm toolkit.

[1] https://github.com/ICTRC/Parsivar

3. Our Approach

Parsivar is an integrated package written in python which performs different kinds of preprocessing tasks in Persian. It should be noted that this toolkit allows for an adjustment between speed and accuracy depends on the user needs. Each task is described in detail in the following subsections.

3.1 Normalization and Tokenization

One of the main problems in Persian text processing is the existence of different character encodings in text documents. For example, the word "آب" (water), might have different encodings in different documents that causes the text processing algorithms to consider them as different words. This problem is more obscurant when we deal with various character encodings in punctuations. To solve this problem, for each character we extracted all different encodings from a corpus of text documents with more than 2 million documents which gathered from Persian weblogs.

Another important task in this step is to find word boundaries in documents. In Persian language, multi-token words can be written in three formats; completely separated by a space delimiter, separated by half-space[2], or be attached to each other. Therefore, determining word and phrase boundaries is a complicated task in Persian. The challenge in Persian is that the space cannot be considered as the only delimiter in all cases.

There are lots of multi-token words in Persian in which, parts of the word are separated by space. For example, in the sentence "من یک برنامه نویس هستم." (I am programmer.), the word "برنامه نویس" (programmer) might be written in two ways of "برنامه‌نویس" and "برنامه نویس". Although the first form of writing is correct and the second one is incorrect, it's usual to write this word in the second form. In this example, considering space as delimiter causes separation of the word "برنامه نویس" into two tokens "برنامه" and "نویس", while it should be taken as one token "برنامه‌نویس". Actually, the challenge is resulted from incorrect spacing in the words. We have tried to solve this problem in two different ways. First, we defined some rules to correct spacing within words. Second, we have tried to train a model that learns how to correct spacing within words.

After the space correction, we tokenize the documents based on spaces and punctuations. In the following subsection, we describe our solutions for space correction in more details.

3.1.1 Rule based space correction

To correct the spaces within words, some certain rules have been defined in the first step using regular expressions. Using these rules, we are able to correct space in many cases such as "می روم" (I'm going), "زمین شناس" (geologist) and "تحلیل گر" (Analyzer). There are still some words that do not match to the rules. These words usually consist of two or three parts which we can't extract a general rule for them such as "گفت و گو" (Conversation). To overcome this problem, we construct a dictionary containing such words and check their existence in the sentences.

3.1.2 Space correction based on learning

To correct the spaces within multi-token words based on learning methods, a model was trained to find words with multiple parts separated by spaces. As a result, we can take all parts of a word as one token.

To train the model, we build a training set using 90% of Bijankhan corpus (Bijankhan et al. 2011). In this corpus, multi token words are placed in one line. We tagged the multi token words using IOB tagging format (Ramshaw and Marcus, 1995) such that the first part is tagged with label "B" and the other parts are tagged with the label "I". Moreover, the other words are tagged with label "O". Then for each word in the sentence, we take the label of previous word, the previous word itself and the next word as features. We trained a Naïve Bayes model to classify each part of the word into classes "B" and "I". At last, we used these labels to find word boundaries. To evaluate the performance of this space correction model, we validated the model on the remaining 10% of Bijankhan corpus. Our model got 96.5% of F1 score in space correction on the validation set.

3.2 Stemmer

Stemming plays an important role in many NLP applications such as information retrieval and text mining. The final goal in stemming is to reduce words to their stem so that for different word forms in a text file, there would be only one stem (Willett, 2006). It is not necessary for the reduced form of the word to be exactly the morphological root. Instead, any other form that improves the performance is acceptable (Krovetz, 1993) There are many stemming algorithms proposed in English. Lovins stemmer (Lovins, 1968) and Porter stemmer (Porter, 1980) are two common stemmers in English. They remove suffixes and prefixes from English words based on some predefined linguistic rules. One of the problems with rule-based stemmers is that they cannot be applied to other languages. Some algorithms are also proposed for stemming in Persian (Sharifloo and Shamsfard, 2008; Taghva, Beckley and Sadeh, 2005). In this section, we propose and implement an algorithm for stemming in Persian language.

Persian words usually derive from other words based on some morphological rules. For example, the word "ستارگان" (stars) is made up of adding suffix "گان" (gan) to the noun "ستاره" (star). We have used such rules to find word stems. In this way, we assumed two set of rules which consider words as verb and non-verb. In the following subsection we describe each of them in detail.

3.2.1 Stemming the Verbs

There are two main roots for present tense and past tense in Persian which can be used to construct various derivations of a verb. For this reason, we collected a list of verb roots in past tense and present tense forms. Then all of the rules were applied to the input verb in order to find the rule to be matched in the best way. Then we search for the resulted roots in the verb dictionary. The first root that is found in the dictionary returns as a word's stem. Some

[2] A Non-Joint Zero Width (NJZW) letter

of the construction rules are shown in Table 1 (Note that the Persian text are read from right to left).

There are also other kinds of verbs which is called prefix verbs. Similar rules like those shown in Table 1 have been used to construct these kinds of verbs, except that a prefix is added at the beginning of the verb. For these kinds of verbs, the algorithm checks for the existence of pre-defined prefix at the first step. After finding and removing the prefixes, the algorithm searches for the stem of the rest of the word in a recursive process.

Table 1 : Some rules for construction of verbs in Persian

Rule	Example
می + بن ماضی + شناسه ماضی (/mi/ + past root + past person identifier)	می‌رفتم (I was going)
می + بن مضارع + شناسه مضارع (/mi/ + present root + present person identifier)	می‌روم (I'm going)
خواه + شناسه مضارع + بن ماضی (/khah/ + present person identifier + past root)	خواهم رفت (I will go)

Algorithm 1 shows the process of finding the verb stem of a word.

Algorithm 1: Verb Stemming

Input: w
foreach $rule$ in $verb_construction_rule$ **do**
  if w matches the $rule$ **do**
    $w' = root\ of\ w$
    if w' *is in* $verb_dict$ **then**
      add w' to $candidate_set$
    end
  end
end
if $candidate_set$ is not empty **then**
  return shortest word in $candidate_set$ as stem
else
  return w
end

3.2.2 Stemming the non-Verbs

We assume that the prefixes and suffixes in non-verbs have a pattern like follows: (to be read from right to left)

{possessive suffix}{plural suffix}{other suffixes}**[stem]**{prefixes}

Note that the items in the brackets are optional.

{Possessive suffix} are suffixes that express ownership and {plural suffix} represents the plurality of the word (Taghva, Beckley and Sadeh, 2005). {Other suffixes} also represent all other kinds of suffixes and {prefixes} shows all kinds of prefixes that would be in the structure of a word. As an example in the word "کتابهایشان" (their books), the possessive suffix "شان" (their) represents the ownership and the suffix "ها" represents the plurality of the word "کتاب" (book). So we have:

"کتابهایشان" (their books)= "شان" (their)+ "ها" (plurality sign)+ "کتاب" (book).

Moreover, in the words "نیرومند" (powerful) and "بی‌صدا" (quiet), the suffix "مند" and the prefix "با" are examples of the "other" category of suffixes.

To find the stem, a list of common suffixes and prefixes for each prefix/suffix category was created. Then for a given word, the algorithm checks for the existence of all the suffixes and prefixes in the list. In the case of finding a suffix/prefix, it would be removed from the word. Then the algorithm checks if the resulted word exists in the lexicon dictionary. If it exists, it returns the resulted word as a stem. Otherwise, other categories of suffixes and prefixes will be checked. The lexicon dictionary is made up of 21151 usual stem words in Persian.

For some cases in which there are more than one prefix/suffix matching of the word, we remove the one which results the smallest stem. This process is described in more details in Algorithm 2.

Algorithm 2: non-Verbs Stemming

Input: w
foreach $suffix_set/prefix_set$ **do**
  foreach s in $suffix_set/prefix_set$ **do**
    if w ends with s **then**
      $w' = w[0:(len(w) - len(s))]$
      if w' is in $Lexicon_dict$ **then**
        add w' to $candidate_set$
      end
    end
  end
end
if $candidate_set$ is not empty **then**
  return shortest word in $candidate_set$ as stem
else
  return w
end

In this algorithm we check the existence of each set of suffixes including {possessive suffix}, {plural suffix}, {other suffixes} and {prefixes}, respectively. After removing all of the found suffixes/prefixes, the algorithm searches for the reminder of the word in the lexicon dictionary.

3.3 POS-tagger

Part of Speech Tagging is a preprocessing step in NLP tasks that assign one of the parts of speech tags to the given word. For example, Part of Speech tags for English sentence "I go to school." and its corresponding Persian translation are shown in Table 2.

Table 2: POS-tags for an English sentence with its corresponding Persian translation

Persian Sentence	.	می روم	مدرسه	به	من
English Sentence	.	Go	School	to	I
POS Tags	Punctuation	Verb	Noun	preposition	Noun

The correct assignment of Part-of-Speech tags is an important issue in semantic analysis and syntax parsing. It can also be used as a suitable feature in Natural Language Processing tasks such as Named Entity Recognition, Statistical Machine Translation and also chunking.

We incorporate Maximum Entropy (ME) and Conditional Random Fields (CRF) that has been proved to get successful results in sequence labeling problems such as POS tagging, Name Entity Recognition (NER), chunking, etc. (Lafferty, McCallum and Pereira, 2001; Ratnaparkhi, 1996). ME and CRF are supervised classifiers with a probabilistic approach which determine the most probable tag of a token given its surrounding context (Pisceldo, Adriani and Manurung, 2009).

If we assume $S = s_1, s_2, \dots, s_n = s_1^n$ as a sentence containing n words, ME model estimates the probability of a tag sequence $T = t_1, t_2, \dots, t_n = t_1^n$ as

$$p(t_1^n|s_1^n) = \prod_{i=1}^{n} p(t_i|t_1^{i-1}, s_1^i) \approx \prod_{i=1}^{n} p(t_i|h_i), \quad (1)$$

Where h_i is a contex window of word s_i. Assuming a context window of size 2, the probability of equation (1) becomes as:

$$p(t_1^n|s_1^n) \approx \prod_{i=1}^{n} p\left(t_i\middle|t_{i-2}^{i-1}, s_{i-2}^{i+2}\right), \quad (2)$$

At the last stage, the model assigns maximum likelihood tag sequence to the words of the sentence (Toutanova and Manning 2000). CRF is a probabilistic graphical model tries to estimate the conditional probability $p(T|S)$ based on some independency assumptions (Lafferty, McCallum, and Pereira 2001).

An annotated corpus is needed for training phase of these supervised methods. To generate a Persian POS-tagger, the Bijankhan corpus (Oroumchian et al. 2006) has been used. The Bijankhan collection contains more than 2.6 million manually tagged words that have been labeled with a collection of 550 tag sets. We omitted more fine grained POS-tags and just used a tag set containing 40 tags. We also applied Stanford POS-tagger (Toutanova et al. 2003; Toutanova and Manning, 2000) that is based on ME model and Wapiti tool that is based on CRF model (Lavergne, Cappé and Yvon, 2010).

For getting reliable results, we trained the model with different sets of n-gram features and got the best feature set for Persian POS-tagger. In the experiments, we found that a feature set comprised of two preceding words, the current word, two following words and two preceding tags are suitable for POS-tagging.

3.4 Shallow Parsing

Shallow parsing or text chunking is a subtask of NLP applications that is used as an alternative to full-sentence parsing (Muñoz et al. 2000). The goal of the text chunking is to divide a sentence into some distinct phrases in a way that syntactically related words grouped as one phrase. These phrases don't have overlap with each other, i.e. a word can only belong to one phrase or chunk (Ramshaw and Marcus, 1995; Tjong Kim Sang, 2002). As an example, in the following sentence the chunks are represented in brackets and each chunk is specified with a label which denotes its type.

$[_{NP}$ I$][_{VP}$ saw$][_{NP}$ the yellow umbrella$]$

Generally speaking, the models proposed to solve this problem are based on pre-defined rules (Grover et al. 2006) or machine learning techniques (Muñoz et al. 2000; Zhai et al. 2017). There have been few efforts to solve this problem in Persian and there aren't any suitable dataset to train a statistical model. In this section, we employed some linguistic rules to find chunks in Persian sentences using regular expressions. These rules are based on POS-tags of the words in the sentence. For example one of the rules that is used in our model is:

NP: {<N_SING><ADJ_SIM><N_SING>}

It means that a singular noun following a simple adjective following another singular noun creates a noun phrase. The following is an example:

- "ضخامت دقیق سیاره"
- "Exact thickness of the planet"

In our approach, we have extracted fifteen linguistic rules to find Persian noun phrases, verb phrases and prepositional phrases. The main advantage of using linguistic rules is that it doesn't need a training corpus. The disadvantage of this approach is that extracting a set of rules which models language complexities is a challenging issue. Moreover, the use of a rule set causes the approach to be language dependent.

4. Experimental Results

The main challenge in evaluation of different parts of the model is the limitation of available resources. For this reason, the performance of some parts of the toolkit were checked manually. To evaluate and compare the performance of Parsivar with the other available toolkits, different types of experiments were performed. For evaluating the tokenization and stemming, we used 10 random documents from the Hamshahri corpus (AleAhmad et al. 2009) which consist of 2552 tokens. After removing the stop words, 1465 words are remained. The accuracy of tokenization tool was 98.29% among these words. Table 3 also shows the evaluation results of stemming among these words.

Table 3 : Evaluation results of Stemmer

	Precision[%]	Recall[%]	F-Measure[%]
Stemmer	98.71	81.91	89.53

To train the space correction model mentioned in section 3.1.2, an IOB tagged dataset has been built using Bijankhan corpus. The dataset contains 2428732 words tagged with "O", 160775 words tagged with "B" and 169518 words tagged with "I". Then it considered as a sequence labeling problem and a Naïve Bayes classifier was trained using 90 percent of this dataset to classify each word into classes "I", "O" and "B". Using these labels, we can specify word boundaries. To test the performance, we validated the model with the remaining 10 percent of the dataset. The results are shown in Table 4.

Table 4 : Evaluation results of space correction model

IOB Accuracy [%]	Precision[%]	Recall[%]	F-Measure[%]
93.2	87.3	91.9	89.5

For the POS tagger, as mentioned in section 3.3, two types of POS taggers have been trained. The training on ME and CRF methods were done with one million words of Bijankhan corpus. These models were tested using 10 percent of Bijankhan corpus with different window sizes. At last, the best one of each type was chosen. Table 5 shows the test accuracy for each model. The accuracy was measured in two levels. In the word level, it presents the correctness of the tagging for each word and in sentence level it presents the correctness of the tagging for a complete sentence.

Table 5 : Evaluation results of POS tagger

		Window size	
		3	5
Maximum Entropy	Word Accuracy [%]	0.91	0.95
	Sentence Accuracy [%]	0.75	0.78
CRF	Word Accuracy [%]	0.93	0.95
	Sentence Accuracy [%]	0.76	0.79

To evaluate the Shallow Parser mentioned in section 3.4, we applied it on 100 randomly selected sentences from Hamshahri corpus. Based on the manually evaluation of the results, the performance was 76.8%. Since this model is based only on POS-tags extracted from sentences, a part of the error is the error propagated by POS-tagger.

We also tried to measure the effect of applying different parts of Parsivar in a downstream NLP task such as plagiarism detection. In the task of plagiarism detection, the goal is to find parts of a text which have been reused from other documents (Asghari et al. 2016). The process starts with a suspicious document d_q and a collection D of documents from which d_q's author may have plagiarized. Within a heuristic retrieval step, a small number of candidate documents D_x, which are likely to be sources for plagiarism, are retrieved from D. Then, within a detailed analysis step, d_q is compared section-wise with the retrieved candidates. All pairs of sections (s_q, s_x) with $s_q \in d_q$ and $s_x \in d_x$, $d_x \in D_x$, are to be retrieved such that s_q and s_x have a high similarity under some retrieval model (Potthast et al. 2010). For this purpose, we performed a preprocessing step on the input text using different toolkits including Hazm, ParsiPardaz and Parsivar. In the next step, we used the output of each toolkit as the input to the Persian plagiarism detection model. At the last step, we compared the results of the plagiarism detection algorithm for each preprocessing toolkit. In our experiments, we used a part of plagiarism detection corpus introduced in (Mashhadirajab et al. 2016) and also (Khoshnavataher et al. 2015) as well.

For plagiarism detection model, we used a VSM based method proposed in (Zechner et al. 2009). In this model, all the sentences of both suspicious and source documents are converted into vectors using TF-IDF weighting method. Then, all sentences of the suspicious document are compared to all sentences of source ones using cosine similarity metric. Pairs of sentences which are similar (based on a pre-defined threshold) have been considered as cases of plagiarism.

The experimental results of plagiarism detection are depicted in Figure 1. Each curve specifies the F-score of the plagiarism detection model for a particular preprocessing toolkit at a specific similarity threshold.

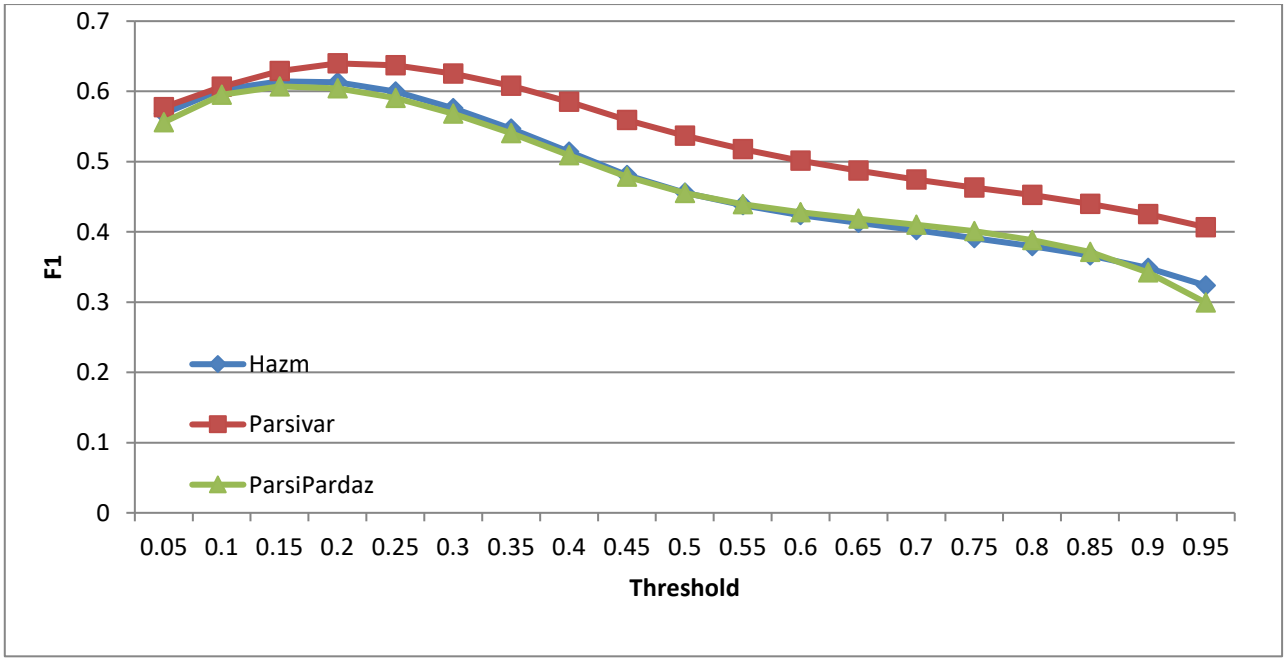


Figure 1: The F-score of a plagiarism detection algorithm for each preprocessing toolkit in various similarity thresholds.

The character level F-measure has been used for evaluating performance of detection based on following equations:

$$Precision\ (S,R) = \frac{1}{|R|}\sum_{r\in R}\frac{\bigcup_{s\in S}(s\sqcap r)}{|r|} \quad (3)$$

$$Recall\ (S,R) = \frac{1}{|S|}\sum_{s\in S}\frac{\bigcup_{r\in R}(s\sqcap r)}{|s|} \quad (4)$$

$$F_1 = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (5)$$

Where S denotes the set of plagiarism cases in the suspicious documents, and R denotes the set of plagiarism that detected by detector for these documents.

It should be noted that the plagiarism detection model is the same for all experiments. As depicted in Figure 1, our toolkit outperforms the other preprocessing toolkits for all threshold values.

The main advantage of our preprocessing toolkit over the other toolkits is in normalization and tokenization steps. As they are used in almost all NLP tasks, the errors generated in these steps will propagate into the other stages and cause more error generation. Unification of the character encodings and correcting the spaces between/ within the words (which results in a better detection of word boundaries) are some reasons which cause a better performance in our toolkit with respect to the other preprocessing toolkits.

5. Conclusion

In this paper we proposed Parsivar, a Persian preprocessing toolkit written in python. This package provides some important preprocessing tasks for NLP purposes such as normalizing, tokenizing of words and multi-token words, stemming, POS tagging and shallow parsing. In the normalization step, we convert different kinds of character encodings to a unique format. In tokenization step, we correct the spacing between and within the words and multi-token words as well. In stemming and shallow parsing, we defined some rules to solve the problem. Finally, in POS-tagging, we trained a CRF and ME based model. As a final stage of this research, we tested our toolkit on a Persian plagiarism detection system. The results show that our toolkit outperforms other similar Persian preprocessing toolkits with respect to F-score.

As a work for the future, we are planning to recognize Persian Ezafe tags as in (Asghari, Maleki and Faili, 2014) to our toolkit. Another work can be accomplished to improve the performance of the shallow parser using statistical approaches.

6. References

AleAhmad, A., Amiri, H., Darrudi, E., Rahgozar, M., & Oroumchian, F. (2009). Hamshahri: A standard Persian text collection. *Knowledge-Based Systems*, 22(5), 382-387.

Asghari, H., Mohtaj, S., Fatemi, O., Faili, H., Rosso, P., & Potthast, M. (2018, February). Algorithms and Corpora for Persian Plagiarism Detection. In *Text Processing: FIRE 2016 International Workshop*, Kolkata, India, December 7–10, 2016, Revised Selected Papers (Vol. 10478, p. 61). Springer.

Asghari, H., Maleki, J., & Faili, H. (2014). A probabilistic approach to persian ezafe recognition. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: Short Papers (pp. 138-142).

Bijankhan, M., Sheykhzadegan, J., Bahrani, M., & Ghayoomi, M. (2011). Lessons from building a Persian written corpus: Peykare. *Language resources and evaluation*, 45(2), 143-164.

Grover, C., & Tobin, R. (2006, May). Rule-based chunking and reusability. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation* (LREC 2006).

Hazm. (2014). Python library for digesting Persian text, "Https://github.com/sobhe/hazm."

Khoshnavataher, K., Zarrabi, V., Mohtaj, S., & Asghari, H. (2015). Developing monolingual Persian corpus for extrinsic plagiarism detection using artificial obfuscation. *Notebook for PAN at CLEF*.

Krovetz, R. (1993, July). Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 191-202). ACM.

Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lavergne, T., Cappé, O., & Yvon, F. (2010, July). Practical very large scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 504-513). Association for Computational Linguistics.

Lovins, J. B. (1968). Development of a stemming algorithm. *Mech. Translat. & Comp. Linguistics*, 11(1-2), 22-31.

Mashhadirajab, F., Shamsfard, M., Adelkhah, R., Shafiee, F., & Saedi, C. (2016). A Text Alignment Corpus for Persian Plagiarism Detection. *In FIRE (Working Notes)* (pp. 184-189).

Munoz, M., Punyakanok, V., Roth, D., & Zimak, D. (2000). A learning approach to shallow parsing. *arXiv preprint cs/0008022*.

Oroumchian, F., Tasharofi, S., Amiri, H., Hojjat, H., & Raja, F. (2006). Creating a feasible corpus for Persian POS tagging. UOWD Technical Report

Series.

Pisceldo, F., Manurung, R., & Adriani, M. (2009). Probabilistic part-of-speech tagging for bahasa indonesia. In *Third International MALINDO Workshop.*

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130-137.

Potthast, M., Barrón-Cedeño, A., Eiselt, A., Stein, B., & Rosso, P. (2010). Overview of the 2nd international competition on plagiarism detection. In *Proceedings of the 4th Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse* (pp. 1-14).

Ramshaw, L. A., & Marcus, M. P. (1999). Text chunking using transformation-based learning. In *Natural language processing using very large corpora* (pp. 157-176). Springer, Dordrecht.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In *Conference on Empirical Methods in Natural Language Processing.*

Sarabi, Z., Mahyar, H., & Farhoodi, M. (2013, October). ParsiPardaz: Persian Language Processing Toolkit. In *Computer and Knowledge Engineering (ICCKE), 2013 3th International eConference on* (pp. 73-79). IEEE.

Shamsfard, M., Jafari, H. S., & Ilbeygi, M. (2010, May). STeP-1: A Set of Fundamental Tools for Persian Text Processing. In *LREC*.

Sharifloo, A. A., & Shamsfard, M. (2008). A bottom up approach to Persian stemming. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II.*

Taghva, K., Beckley, R., & Sadeh, M. (2005, April). A stemming algorithm for the farsi language. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on* (Vol. 1, pp. 158-162). IEEE.

Tjong Kim Sang, E. F., & De Meulder, F. (2003, May). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4* (pp. 142-147). Association for Computational Linguistics.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003, May). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (pp. 173-180). Association for Computational Linguistics.

Toutanova, K., & Manning, C. D. (2000, October). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13* (pp. 63-70). Association for Computational Linguistics.

Willett, P. (2006). The Porter stemming algorithm: then and now. *Program*, 40(3), 219-223.

Zechner, M., Muhr, M., Kern, R., & Granitzer, M. (2009, September). External and intrinsic plagiarism detection using vector space models. In *Proc. SEPLN* (Vol. 32, pp. 47-55).

Zhai, F., Potdar, S., Xiang, B., & Zhou, B. (2017, January). Neural Models for Sequence Chunking. In *AAAI* (pp. 3365-3371).

Multilingual Word Segmentation: Training Many Language-Specific Tokenizers Smoothly Thanks to the Universal Dependencies Corpus

Erwan Moreau[1], **Carl Vogel**[2]
[1] Adapt Centre, [2] Trinity Centre for Computing and Language Studies, [1,2] Trinity College Dublin
Trinity College Dublin, Dublin 2
{moreaue, vogel}@scss.tcd.ie

Abstract

This paper describes how a tokenizer can be trained from any dataset in the Universal Dependencies 2.1 corpus (UD2) (Nivre et al., 2017). A software tool, which relies on Elephant (Evang et al., 2013) to perform the training, is also made available. Beyond providing the community with a large choice of language-specific tokenizers, we argue in this paper that: (1) tokenization should be considered as a supervised task; (2) language scalability requires a streamlined software engineering process across languages.

Keywords: Universal Dependencies, Word Segmentation, Tokenization, Multilinguality, Interoperability

1. Introduction

This paper describes how a tokenizer can be trained from any dataset in the Universal Dependencies 2.1 corpus (UD2) (Nivre et al., 2017). A software tool, which relies on Elephant (Evang et al., 2013) to perform the training, is also made available. Beyond providing the community with a large choice of language-specific tokenizers, this paper explores two ideas:

- **A new perspective on word segmentation.** We argue that tokenization does not depend only on language and genre,[1] but includes conventions and choices related to ambiguous cases within the same language-genre pair. As a consequence, using some predefined tokenizer might cause the loss of some information, especially when combining multiple language resources. It follows that, in general, tokenization should be seen as a supervised task, where textual data is tokenized following a given "model": by making this model explicit, one can enforce model consistency when connecting various pre-tokenized language resources together, hence avoiding errors which might otherwise go undetected.[2]
- **Towards language scalability**. Major progress has been achieved in multilingual language technology in the recent years. In NLP applications, scalability in terms of data size has been addressed for the most part, but scalability in terms of language diversity is still a significant challenge. The Universal Dependencies 2.1 corpus includes 102 annotated datasets and 59 distinct languages, thanks to the authors' and contributors' great effort (Nivre et al., 2017). Packaging such a diversity of languages in a uniform format is a major step towards the ability to process multiple languages in an homogeneous way, which is the cornerstone of language-wise scalability. Bikel and Zitouni (2012, xxi) mention that *"Previously, to build robust and accurate multilingual natural language processing (NLP) applications, a researcher or developer had to consult several reference books and dozens, if not hundreds, of journal and conference papers."* One might add that said researcher or developer would also have to find, test and integrate multiple language-specific software tools. Thus, language scalability also requires a more streamlined engineering process: it becomes impractical to find a specific software tool for every language to process, let alone the best tool for every language. This is why we argue that the evaluation of software tools should progressively shift the focus from accuracy in a specific language to robustness and adaptability to a wide range of languages. We see the tool presented in this paper as a modest step in this direction.

This paper is organized as follows. First, we explain in section §2. why the need to tackle increasingly complex multilingual tasks in NLP makes it necessary to have robust and flexible pre-processing tools available, like tokenizers. In §3. we present the state of the art in word segmentation, together with the existing tools available. In §4. we present a new tool which satisfies the aforementioned criteria of robustness and flexibility. Finally we demonstrate the interest of the tool in three experiments in §5.

2. Motivations

Many NLP shared tasks nowadays provide datasets annotated with the linguistic information relevant to the task, so that participants can focus on the core aspects of the task rather than spend time on non-essential pre-processing steps. For example, the CoNLL format and its variants are widespread among the NLP community; a dataset provided in this format usually contains word and sentence segmentation, POS tagging, syntactic dependencies and possibly

[1] E.g. tokenizing tweets versus tokenizing The New York Times.

[2] This idea can be compared to the case of POS tags, for instance: it is clear that mixing datasets annotated with different labeling schemes can lead to errors. In the case of tokenization, the scheme consists in the guidelines that the annotators follow for choosing which units represent tokens (otherwise stated, where the annotators choose to put the boudaries between tokens).

other relevant features. This kind of annotation is particularly helpful in the case of multilingual shared tasks, since participants are unlikely to be familiar with all the languages to process.

All these efforts of the community aim at tackling more and more sophisticated problems on more and more diverse languages and types of data. As this natural trend to address high-level tasks progresses, the need to work with multiple datasets from various origins and in various formats grows. For example, it is common to collect raw data from the Internet in order to increase the coverage of a ML model, rather than relying solely on the annotated training data. But combining heterogenous datasets comes with challenges, and unsurprisingly the first one is tokenization. Tokenization errors can be costly performance-wise, as these errors may propagate through the whole processing chain. While sometimes these problems can be diagnosed and fixed manually, typically when dealing with only a few familiar languages, the multiplicity of languages makes manual diagnosis impractical. Some generic tokenization methods can be used as a fallback, but this is rarely optimal. As a consequence, such cases can only be solved by training a tokenizer on the provided input data, in order to apply the same tokenization choices in the third-party corpus.

3. Related Work

Tokenization has traditionally been addressed with rule-based systems (e.g. by Dridan and Oepen (2012)), but supervised ML approaches are more and more common due to their flexibility when tackling new languages (see e.g. (Frunza, 2008)). Dridan and Oepen (2012) and Fares et al. (2013) analyse the question of tokenization ambiguities and the resulting diversity of tokenization conventions for the English language, emphasizing the fact that many tokenization schemes co-exist in practice. In this context, both (Mark and Bojar, 2012) and (Evang et al., 2013) propose a supervised approach, considering tokenization as a sequence labeling problem. They both use Conditional Random Fields (Lafferty et al., 2001) to solve it, but in (Evang et al., 2013) the CRF features are enriched with the top N most active neurons of a Recurrent Neural Network (RNN) language model, based on the work of Chrupala (2013) for character-level language modeling. This approach shows significant improvement over the simple CRF one, as the authors show with three datasets in English, Dutch and Italian. (Mark and Bojar, 2012) validate their system on English and Chinese.

In this work we use the Elephant[3] tokenizer training software, described in (Evang et al., 2013). This software itself relies on the Wapiti implementation (Lavergne et al., 2010) for sequence labeling with CRFs, and on the work of (Mikolov et al., 2010) for the RNN language modeling. Although Elephant is capable of segmenting sentences as well, in this work we focus on word segmentation only.

[3]The authors named their system Elephant because *"like an elephant in the living room, it is a problem that is impossible to overlook whenever new raw datasets need to be processed or when tokenization conventions are reconsidered."* (Evang et al., 2013, 1422).

4. Tool: An Elephant Wrapper

4.1. Motivations

The authors of Elephant (Evang et al., 2013) made their system available at `http://gmb.let.rug.nl/elephant`,[4] and deserve our gratitude for making the effort to provide a clear documentation, including how to reproduce their experiments. However, although a training script is provided, this script only allows training the CRF model. Hence the user can either use one of the three pre-trained models, or train a CRF-only model, without the RNN language model features.[5] Additionally, the user must provide a CRF template file which describes the features that the sequential model uses. This means that the user has either to pick a template at random, or proceed by trial and error in order to find a suitable template.

Finally, for users who simply need to tokenize some data and cannot (or do not want to) train a model, Elephant contains pre-trained tokenizers but for only three languages.

The tool that we propose is available at `https://github.com/erwanm/elephant-wrapper`.[6]

It aims to fill the aforementioned usability gaps in Elephant, together with providing users with a more universal tool for word segmentation, in terms of technical usage (training and testing capabilities) as well as diversity of languages. The latter is made possible thanks to the availability of the Universal Dependencies 2.1 corpus (Nivre et al., 2017), which contains 102 datasets in 59 distinct languages.

4.2. Implementation

In the perspective of providing universal tokenizer training capabilities, we propose several scripts intended to make this part of the training more straightforward. Additionally, our wrapper provides a convenient way to generate the required IOB-labeled sequences[7] of characters from a pre-tokenized corpus. In particular, the `.conllu` format used in UD2 (Nivre et al., 2017) is accepted as input, as well as similar formats such as the one used in the 2017 Shared Task on Identifying Verbal Multi-Word Expressions (see §5.4.). This is intended to streamline the process of training a tokenizer by making the required internal formats transparent to the user, thus improving greatly the usability of the system. The conversion to the IOB format is implemented in the following way: following the general good practice of preserving the form of the original data, the UD2 datasets contain annotations which indicate for every token whether a space follows the token or not.[8] Thus, it is possi-

[4]Last retrieved: 21/02/2018.

[5]Remark: training the RNN language model so that it can be used with the existing CRF training script is not entirely trivial because it requires converting the training data to Unicode characters codes presented as "tokens".

[6]The tool can also be found at `https://www.scss.tcd.ie/clg/elephant-wrapper`.

[7]The IOB format consists of labeling every character with: B for the beginning of a token, I for subsequent characters inside a token, and O for characters outside any token (whitespaces).

[8]This indication is provided in column 10 of the `.conllu` file: this column contains the parameter/value pair `SpaceAfter=No` if and only if no space follows the token. Remark: the `.txt` file could also be used, but this would require

ble to rebuild the original text, together with the appropriate IOB label for every character. Additionally, our conversion script takes care of ignoring tokens corresponding to expanded contractions when they appear: several languages contain contracted tokens, e.g. *"du" (of the)* in French; such cases are represented as follows in UD2: (1) the surface token is given on the first line with a range of indexes as value in the first column (e.g. `18-19 du`), instead of a single token index; (2) the tokens on the following lines correspond to the expansion, and their indexes belong to the previously seen range (e.g. `18 de` followed by `19 le`). The expanded form is necessary for the morpho-syntactic analysis, but is irrelevant for the tokenization part.

The tool also provides scripts which generate CRF template files for Wapiti and run k-fold cross-validation with every possible template in a predefined set. This way the best performing template can be selected for the final training. The template indicates which features are used in the sequential model:

- Value n of n-grams features;
- Length of the window of characters;
- Using characters Unicode code point and/or Unicode category;
- Using the top 10 RNN features or not.

The set of patterns to test is simple to configure, so that the user can choose how thorough the search should be. Of course, the time required to run the cross-validation process depends on the number of templates to apply, but an option is supplied to stop the search when the performance shows no progress anymore.[9]

Finally, the tool provides 102 tokenizer models trained from the UD2 corpus (see §5.1.). For the convenience of the user, the language model to use by the tokenizer can be given in different forms: as a custom model directory, as the name of a UD2 dataset, or as the ISO639 standard code for the language.[10]

5. Experiments

Below we present several experiments made with the Elephant Wrapper tool introduced in §4.2., available at `https://github.com/erwanm/elephant-wrapper`. The tool contains the scripts required to reproduce these experiments. It was designed to make batch processing of multilingual experiments as simple as possible, with the idea of language scalability in mind.[11]

5.1. Training Multiple Tokenizers from UD2

The first experiment is essentially intended to demonstrate the feasibility and effectiveness of training multiple language-specific tokenizers in an homogeneous way, thus making the engineering design process straightforward. This is only made possible thanks to the consistency in the annotation format across datasets offered by the UD2 corpus.

The experiment consists in training a tokenizer for every dataset in the UD2 corpus using the file `<lang>-ud-train.conllu` as training data, then testing the tokenizer on the `<lang>-ud-test.conllu` file. The method generalizes the one described in (Evang et al., 2013) (see §3.): a CRF model is trained, which might include features from a RNN language model previously trained with the same training data. However, instead of using only one specific template for the CRF model, the cross-validation stage described in §4.2. is run over the training set in order to find the optimal template among a large set of possibilities (96 templates). Then the final training is performed on the whole training set, using the selected template. A simple generic tokenizer is used as a baseline for the sake of comparison: it relies on whitespaces and strips any sequence of punctuation signs from the word. Both the trained tokenizer and the baseline tokenizer are applied to the test set.

Table 1 gives the performance obtained by both the generic tokenizer (baseline) and the trained tokenizer for every dataset in UD2. For the evaluation we follow Shao et al. (2017): performance is measured using the token-based recall, i.e. the proportion of tokens correctly identified among the gold-standard tokens.[12] For all the datasets but two,[13], the Elephant-trained system performs as well or better than the baseline; in many cases the former dramatically outperforms the latter. The mean performance of the trained tokenizer is 99.23%; in average it improves the performance by 86% compared to the baseline, but the mean poorly reflect the diversity of the cases: for 71% of the datasets the performance increases by less than 5%, whereas for the top 13% it increases by more than 100%; the median improvement is 2.7%.

Due to the large number of datasets and the authors' ignorance of the vast majority of languages, it is completely impractical to investigate every case individually. However we investigated the cases where the tokenizer obtains a perfect score. Many such cases are due to the lack of annotation indicating whether a token is followed by a space or not (see 4.2.): Coptic, Danish, Finnish-FTB, Gothic, Marathi, Norwegian-NynorskLIA, Swedish_Sign_Language, Slovenian-SST, Telugu; in a few cases this might be due to the nature of the data (e.g. for the Swedish Sign Language); otherwise this is an error which prevents reconstructing the non-tokenized text from the data.[14] Some ancient languages do not contain any

accessing the .conllu file anyway in order to identify the tokens' boundaries.

[9] By default the template search space is explored from the simplest templates to the most complex.

[10] In the latter case, if there are several UD2 datasets for the same language, the default UD2 one (with no extension) is used.

[11] Feedback and contributions are welcome.

[12] The software also allows evaluation using character-based accuracy or error rate.

[13] The performance decreases very slightly for Kazakh (-0.65%) and Latin (-0.02%). In the case of Kazakh, this might be due to the small size of the training set (only 511 tokens) causing the model to overfit slightly; in the other case the difference is too small to be significant.

[14] Since the annotations indicate the absence of space after a token (see 4.2.), the system assumes that there are spaces everywhere between the tokens, in both the training and test set. This

UD 2.1 Dataset name	Training set size (tokens)	Baseline Recall (%)	Trained tokenizer Recall (%)	UD 2.1 Dataset name	Training set size (tokens)	Baseline Recall (%)	Trained tokenizer Recall (%)
Afrikaans	33894	95.73	99.93	Italian [E]	252631	95.65	99.82
Ancient_Greek	159895	96.35	99.97	Italian-ParTUT	45477	97.05	99.76
Ancient_Greek-PROIEL	184382	99.29	100.00	Italian-PoSTWITA [E]	49478	88.63	99.60
Arabic	191869	97.13	99.96	Italian-PUD	17746	96.46	99.87
Arabic-NYUAD	502991	100.00	100.00	Japanese	161900	13.09	93.01
Arabic-PUD	16601	70.83	99.09	Japanese-PUD	21454	15.11	96.45
Basque [E]	72974	97.54	99.98	Kazakh [E]	511	96.27	95.64
Belarusian	5217	95.15	99.35	Korean	52328	93.53	99.64
Bulgarian [E]	124336	97.52	99.80	Kurmanji	7957	97.50	99.67
Buryat	8026	98.77	99.92	Latin [E]	8018	99.98	99.96
Cantonese	552	27.39	96.81	Latin-ITTB	270403	99.43	99.76
Catalan	416659	93.35	99.94	Latin-PROIEL	147044	99.90	100.00
Chinese	98608	18.23	92.08	Latvian	62397	97.45	99.13
Chinese-CFL	5805	12.98	94.45	Lithuanian	3210	95.94	99.06
Chinese-HK	1497	20.63	93.21	Marathi	2730	29.79	100.00
Chinese-PUD	17132	16.76	96.33	North_Sami	16835	98.64	99.56
Coptic	4238	100.00	100.00	Norwegian-Bokmaal	243887	98.54	99.89
Croatian	169283	97.82	99.89	Norwegian-Nynorsk	245330	98.73	99.90
Czech [E]	1171190	98.47	99.95	Norwegian-NynorskLIA	8020	98.54	100.00
Czech-CAC [E]	471594	99.96	99.96	Old_Church_Slavonic	37432	91.78	100.00
Czech-CLTT	26225	92.97	99.54	Persian [E]	119945	92.91	100.00
Czech-FicTree	133137	96.27	99.99	Polish	61906	99.42	99.94
Czech-PUD	14852	98.63	99.88	Portuguese	191410	96.44	99.64
Danish	80378	98.01	100.00	Portuguese-BR	238334	98.04	99.88
Dutch [E]	186046	97.52	98.96	Portuguese-PUD	17511	98.72	99.77
Dutch-LassySmall	81243	95.48	99.84	Romanian	185113	96.87	99.49
English	204607	95.36	98.78	Romanian-Nonstandard	10352	95.87	98.28
English-LinES	50095	97.27	99.96	Russian	75964	96.06	99.89
English-ParTUT	43491	99.21	99.71	Russian-PUD [E]	15559	97.39	99.88
English-PUD	16941	97.60	99.89	Russian-SynTagRus	871082	96.82	99.58
Estonian	85567	96.10	99.81	Sanskrit	1158	8.22	99.59
Finnish	162827	97.22	99.53	Serbian	65764	96.58	99.88
Finnish-FTB	127359	99.08	100.00	Slovak [E]	80575	98.95	99.65
Finnish-PUD	12650	96.33	99.79	Slovenian	112530	98.99	99.87
French	346855	92.47	99.37	Slovenian-SST	9487	95.02	100.00
French-FTB	442228	66.57	100.00	Spanish	375149	98.47	99.95
French-ParTUT	23324	94.87	99.84	Spanish-AnCora [E]	443087	97.85	99.91
French-PUD [E]	19310	93.57	99.87	Spanish-PUD [E]	18258	99.16	99.96
French-Sequoia	49198	92.79	99.73	Swedish	66645	98.44	99.95
Galician	71928	97.96	99.83	Swedish-LinES	48325	99.08	99.95
Galician-TreeGal	4447	98.54	99.62	Swedish-PUD	15259	98.18	99.86
German	258927	97.56	99.14	Swedish_Sign_Language	644	59.22	100.00
German-PUD	16798	98.61	99.80	Tamil	5734	4.01	99.21
Gothic	35024	99.95	100.00	Telugu	5082	26.21	100.00
Greek [E]	41212	98.37	99.67	Turkish	36970	97.59	99.86
Hebrew	98348	97.34	99.93	Turkish-PUD	13228	94.90	99.83
Hindi	281057	16.27	99.97	Ukrainian	75054	96.18	99.63
Hindi-PUD	19063	13.74	99.92	Upper_Sorbian	8589	96.74	100.00
Hungarian	20166	96.79	99.82	Urdu	108690	98.55	100.00
Indonesian	97531	98.50	99.97	Uyghur	8264	98.47	99.95
Irish	3183	97.01	99.43	Vietnamese	20285	79.37	85.42

Table 1: Token-based recall (percentages) for the 102 datasets in UD 2.1, for both the baseline tokenizer and the Elephant-trained tokenizer. For the latter, the optimal model was selected by using cross-validation on the training set (see §4.2.); datasets marked with "[E]" show cases for which a model containing the Elman features (top 10 RNN most active neurons) is selected.

punctuation signs at all or very few, which makes the tokenization trivial: Ancient_Greek-PROIEL, Latin-PROIEL, Old_Church_Slavonic. A couple of datasets seem to be corrupted in version 2.1 of the UD corpus: Arabic-NYUAD and French-FTB (all the tokens are replaced with underscore signs). Finally the cases of Persian, Upper_Sorbian and Urdu show no sign of error. We decided to keep all these possibly erroneous cases in the results, because we cannot distinguish easily which of them are actual errors and have no way either to know if other datasets with apparently regular performance contain errors.

makes the task of the tokenizer trivial.

The case of CJKV languages (Chinese, Japanese, Korean and Vietnamese) is worth observing: although the CRF approach was originally designed for Indo-European languages, it performs decently on these Asian languages, considering their specificities. The recall ranges from 85.42% for Vietnamese to 99.64% for Korean; while the baseline tokenizer performs very poorly with these languages with a mean of 33.01%, the mean performance of the Elephant-trained tokenizer is 94.15%.

In (Evang et al., 2013), experiments are performed on three European languages: English, Dutch and Italian. Besides the number of languages, the experimental setup also dif-

fers by the number of CRF templates tested: in (Evang et al., 2013) the characteristics determining the template are tested with a few predefined values for all the languages. For instance, the first experiment concludes that the use of both Unicode code points and categories gives the best performance overall, even though using only the former performs better for English; then the following experiments use both Unicode code points and categories for all the languages including English, for which this is suboptimal according to the previous experiment. By contrast, our approach determines the best template for each individual language among a large set of possibilities (96 in this experiment) using cross-validation. Although this requires more training time, it will generate a more accurate model in general. From the perspective of language scalability, it should be emphasized that the requirement to streamline multilingual processes does not entail overlooking language-specific features: the design of the process should be as uniform as possible, but without "standardizing" the languages themselves.

One of the consequences of our approach is that the RNN features are selected or not based on the results of the cross-validation stage on the training data. It turns out that most datasets in the UD2 corpus do not benefit from these features: they were selected as part of the optimal template in only 16 cases (16% of the datasets). In particular, none of the CJKV languages benefits from them. In (Evang et al., 2013), these features were reported to provide a significant improvement in all three languages studied. Besides the difference in the datasets, we suppose that this difference might be caused by the more thorough search for an optimal template in our experiment: by exploring many more templates (with or without the RNN features), the process is more likely to reach a optimal level of performance, equivalent to the one that could have been reached with a less fine-grained template containing RNN features.

5.2. Training on a Different Dataset in the Same Language

The second experiment aims to illustrate the fact that tokenization follows a particular scheme, and different schemes lead to different ways to tokenize a text even within the same language. Consequently, when a task relies on matching tokens from a text with a given pre-tokenized language resource, it is important to ensure the consistency of the tokenizer with the "scheme" which was used to generate the resource. We illustrate this point by running the following experiment with five languages in UD2 for which several distinct datasets are provided: a tokenizer is trained with the training set of every dataset for the language, and applied to the test set of every dataset as well. Table 2 shows the performance in every case. As expected in any similar supervised ML setting, the best results are consistently achieved when the training and test set are drawn from the same corpus (with only one exception in Italian). This experiment shows that token boundaries do not only depend on the language, and therefore that applying a certain tokenizer for this sole reason is not always optimal. In particular, the fact that traditional tokenizers are rule-based does not imply that there is a unique way to tokenize a language. This is why we think that word segmentation should be considered a supervised task.

Training set	Chinese	Chinese CFL	Chinese HK	Chinese PUD
Chinese	**92.08**	71.32	75.09	90.46
Chinese-CFL	60.96	**94.45**	72.31	65.71
Chinese-HK	57.57	70.05	**93.21**	60.76
Chinese-PUD	77.42	70.40	73.12	**96.33**

Training set	Czech	Czech CAC	Czech CLTT	Czech FicTree	Czech PUD
Czech	**99.95**	99.96	93.73	**99.99**	99.69
Czech-CAC	96.38	**99.96**	95.78	94.30	97.33
Czech-CLTT	95.79	99.37	**99.54**	90.95	97.14
Czech-FicTree	99.16	99.88	94.41	**99.99**	99.38
Czech-PUD	98.79	99.89	94.90	99.43	**99.88**

Training set	English	English LinES	English ParTUT	English PUD
English	**98.78**	99.35	99.24	99.77
English-LinES	95.40	**99.96**	98.18	96.93
English-ParTUT	95.45	98.45	**99.71**	98.98
English-PUD	96.37	98.97	99.32	**99.89**

Training set	French	French ParTUT	French PUD	French Sequoia
French	**99.37**	99.36	97.69	99.36
French-ParTUT	98.66	**99.84**	96.79	98.55
French-PUD	97.88	99.36	**99.87**	98.22
French-Sequoia	99.00	99.36	96.74	**99.73**

Training set	Italian	Italian ParTUT	Italian PoSTWITA	Italian PUD
Italian	**99.82**	**99.82**	96.19	99.65
Italian-ParTUT	99.65	99.76	90.56	99.68
Italian-PoSTWITA	97.81	99.13	**99.60**	98.76
Italian-PUD	99.44	99.73	89.35	**99.87**

Table 2: Token-based recall (percentages) when applying a tokenizer trained on a given dataset (rows) to another dataset (columns) in the same language. The highest performance for each test set is highlighted in bold. Example: when applying the model trained on the Italian-ParTUT training set to the Italian-PUD test set, the recall is 99.68%.

5.3. Impact of the Size of the Training Set

Since tokenization should be seen as a supervised task, it is important to know how much data is needed to train an accurate model. This is why in this section we study the impact of the size of the training set on the perfomance of the tokenizer. The experiment simply consists in training a model using only a subset of the training set instead of the whole data, for various sizes of the subset; then the model is applied to the regular test set. For this experiment we select a group of datasets for which a large training set is provided, in order to collect the results for a large range of sizes.

Figure 1 shows the impact on performance of linearly increasing the size of the training set from 890 sentences to 8900, for the 20 largest datasets. Some datasets obtain a decent level of performance with as little as 890 sentences: for instance the Spanish one reaches 99.94%. However the variance is high, with most datasets far from their maximum performance. In fact, the main difference when increasing the size of the training set is that the variance decreases: the mean progresses as well, but the most important observation that can be made from Figure 1 is that, as the size of the training set increases, all the languages reach a high level of performance. In other words, even if a small training set might suffice to obtain an accurate tokenizer,

training it with a large number of sentence makes it more likely to be accurate. In terms of performance, Figure 2 shows that the performance mostly follows a logarithmic progression with respect to the size of the training set. This means that when the performance reaches a high level, increasing it more requires to add a much larger amount of data.

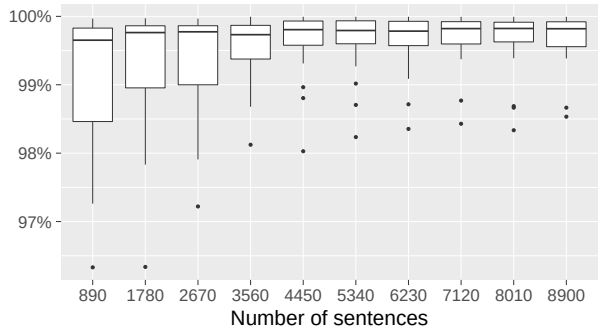


Figure 1: Token-based recall by number of sentences in the training set for the 20 largest datasets in UD 2.1 (by size of the full training set in number of sentences). Each boxplot represents the performance on the test set of a model trained with n sentences, for all 20 datasets.

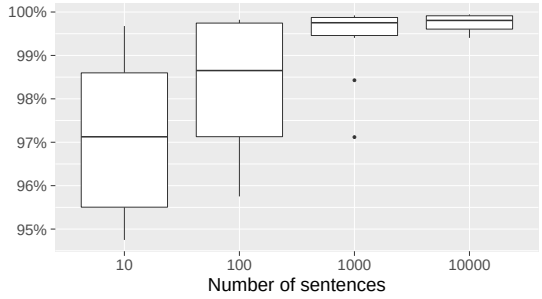


Figure 2: Token-based recall by number of sentences in the training set with exponential progression on the X axis, for the 10 largest datasets in UD 2.1.

5.4. Use Case: Detecting Multi-Word Expressions

As an example of a complex multilingual task in which supervised tokenization can help (see §2.), the authors of this paper participated in the 2017 Shared Task on Identifying Verbal Multi-Word Expressions (VMWE) (Savary et al., 2017). The task consists in identifying VMWEs in 18 different languages. The input data is provided in the CoNLL format, annotated with tokens, POS tags and dependencies.[15] In the approach described by Maldonado et al. (2017) and Moreau et al. (2017), we propose to leverage third-party raw text corpora in order to calculate semantic context vectors for the candidate expressions. Since computing context vectors requires a large resource, we opted for using the Europarl corpus (Koehn, 2005), which is available in 12 among the 15 required languages.[16] The candidate expressions are provided by the first component of the system, a sequence labeling system using CRF. The CRF component provides 10 candidate labeled sequences, the top one being the default choice; the second component (reranker) aims to replace the default choice with one of the other candidates if needed, based on semantic similarity. In order to make the semantic vectors as reliable as possible (i.e., as representative of the meaning of the expression as possible), the system must match all the occurrences of a candidate VMWE (extracted from the input data) in the third-party corpus. But Europarl might not contain every possible VMWE found in the VMWE17 data, or might contain too few occurrences for some VMWEs. Moreover, the discrepancies between the tokenized input data and the third party data (when tokenized using a generic tokenizer) might prevent matching correctly VMWEs which contain words tokenized in a different way. In particular, VMWEs frequently include function words which are susceptible to tokenization errors, like *"c'est" (it is)* in French: if the tokenizer does not properly recognize the apostrophe as part of the first token *"c' "*,[17] expressions which contain *"c'est"* cannot be matched. Moreover, even using a language-specific tokenizer might not always solve this problem, because some tokenization ambiguities cannot be solved other than by an arbitrary choice depending on interpretation; for instance, the documentation about the way tokenization is carried out for French in the UD2 corpus mentions that: *"This tokenizing and segmentating choice is arbitrary and other French treebanks could choose to do otherwise."*[18] As a consequence, a dataset-specific tokenizer should be trained on the provided input data in order to tokenize the third-party corpus accurately.

	Baseline	Trained tokenizer	Improvement
Precision	81.39	81.27	-0.15
Recall	22.43	22.83	+1.77
F-score	35.17	35.64	+1.35

Table 3: Sentence-level micro precision, micro recall and micro F-score of the reranker over all the datasets (i.e., considering all the instances from all the datasets together). (VMWE17 Shared Task data, §5.4.)

Table 3 shows the performance of the reranker on its own, first when tokenizing Europarl with a generic tokenizer (baseline) and then when tokenizing with a specific model trained on the appropriate training data. It can be observed that the precision decreases slightly, but the recall increases more strongly so that the F-score also increases.

[15] All the datasets but the following three are provided in the CoNLL format: Bulgarian, Hebrew and Lithuanian (these languages are discarded in our experiments).

[16] The following three languages are discarded for this reason: Farsi, Maltese and Turkish.

[17] One may notice that this is the opposite in English: the apostrophe should normally be assigned to the token on its right hand-side, like "'s" in "it's".

[18] `https://github.com/UniversalDependencies/docs/blob/pages-source/_fr/tokenization.md`; last retrieved: 19/02/2018.

Language	Baseline	Trained tokenizer	Improvement
CS	19.59	19.59	0.00
DE	23.65	23.73	+0.34
EL	08.72	08.72	0.00
ES	14.45	14.45	0.00
FR	17.46	19.68	+12.71
HU	36.39	36.39	0.00
IT	19.18	19.18	0.00
PL	16.12	16.22	+0.61
PT	10.73	10.58	-1.43
RO	06.38	06.38	0.00
SL	12.02	12.02	0.00
SV	10.56	10.56	0.00

Table 4: Proportion of VMWEs from the VMWE17 test data found in Europarl by dataset (percentages).[20]

In order to understand these results, one has to look at the different ways in which the tokenization of Europarl can impact the performance of the reranker. First, a candidate expression which was not identified before in Europarl might be identified thanks to the more accurate tokenization. Table 4 shows the proportion of expressions which are identified in Europarl: while for most languages using the appropriate tokenization model does not impact this proportion or only slightly, the French dataset shows a large improvement. We think that this is due to the case of the apostrophe, which is ubiquitous in French and cannot be handled properly by the baseline tokenizer. Second, a more accurate tokenization can affect the context words of an expression, and thus improve the accuracy of the context vectors, i.e. the vectors can be more representative of the meaning of the VMWE. This effect can be observed in Table 5, which gives the recall by dataset for the expressions covered in Europarl: the recall increases by 8% overall, with many languages showing a significant improvement up to double the baseline recall (for Hungarian). The F-score for these instances follows a similar trend, with an average improvement of 6.96%.

Thus the effect of a better tokenization varies greatly depending on the language, but in general it gives the reranker more and/or better information through the context vectors; thanks to this, the reranker can then take more risk in proposing an alternative labeled sequence, hence the increase in recall. This explanation is confirmed by breaking down the performance by number of expressions in the sentence, as shown in Table 6: when using the Elephant-trained model for Europarl, the performance decreases for the sentences which contain no VMWE at all, but increases noticeably for all the cases where the sentence contains at least one expression. It is worth noticing that the improvement includes the cases containing multiple expressions, which are the hardest to identify.

Finally, considering the whole system (CRF and reranker components together) and the official evaluation measure used in the VMWE17 Shared Task, the advantage of supervised tokenization translates into a modest 0.3% F-score improvement at the level of expressions, as shown in Table 7. This can be explained for the most part by the higher risk that the reranker takes, which increases the recall at the cost of decreasing the precision. Additionally, the official evaluation measure does not particularly reward the fact the system captures more difficult cases, as shown in Table 6.

[20] Counting the VMWEs in the gold standard test set, but only if the CRF component returns the gold labeled sequence as one of the 10 candidates (since otherwise the reranker cannot select the right answer).

Language	Baseline	Trained tokenizer	Improvement
CS	58.03	58.55	+0.89
DE	27.27	34.33	+25.87
EL	37.70	42.62	+13.04
ES	32.84	34.33	+4.55
FR	34.09	34.48	+1.15
HU	12.77	27.66	+116.67
IT	9.76	9.76	0.00
PL	32.43	32.43	0.00
PT	45.00	49.15	+9.23
RO	57.14	57.14	0.00
SL	36.96	34.78	-5.88
SV	6.56	6.56	0.00
All (macro)	32.55	35.15	+8.00

Table 5: Recall by dataset for the 8,544 sentences which contain at least one expression which appears in Europarl (percentages).

# exprs. / sentence	Proportion data	Baseline Macro	Baseline Micro	Trained tokenizer Macro	Trained tokenizer Micro
0	81.0%	**53.8**	**52.6**	53.1	51.7
1	16.4%	23.5	26.8	**25.6**	**28.1**
2	2.2%	30.4	25.6	**36.6**	**25.8**
3	0.3%	31.0	16.7	**34.3**	**18.2**
All	100%	29.1	35.2	**29.9**	**35.6**

Table 6: Macro and micro F-score (percentages) at sentence level by number of expressions in the sentence, for all the datasets.

6. Conclusion

In this paper we proposed a method to train multiple tokenizers using multilingual resources, like the Universal Dependencies 2.1 corpus. We showed that word segmentation can and should be seen as a supervised process, as opposed to a language-uniform process. More generally, this approach is in line with the view that language scalability is one of the most important challenges in NLP tasks. As a consequence, NLP tasks have to evolve into a more streamlined software engineering process across languages. We see the software tool that we propose for tokenization as a contribution to this idea.

Language	Baseline	Trained tokenizer	Improvement
CS	70.64	70.69	+0.07
DE	28.70	30.99	+7.98
EL	37.57	37.72	+0.40
ES	49.05	49.68	+1.28
FR	58.40	57.92	-0.82
HU	68.47	67.95	-0.76
IT	16.28	15.87	-2.52
PL	72.42	72.19	-0.32
PT	65.22	66.06	+1.29
RO	83.53	83.53	0.00
SL	44.02	43.63	-0.89
SV	31.91	31.91	0.00
All	52.18	52.34	+0.31

Table 7: VMWE17 official evaluation measure: F-score at the MWE level (percentages).

7. Acknowledgements

The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

The graphics in this paper were created with `R` (R Core Team, 2012), using the `ggplot2` library (Wickham, 2009).

8. Bibliographical References

Bikel, D. and Zitouni, I. (2012). *Multilingual Natural Language Processing Applications: From Theory to Practice*. IBM Press, 1st edition.

Chrupala, G. (2013). Text segmentation with character-level text embeddings. *CoRR*, abs/1309.4628. Workshop on Deep Learning for Audio, Speech and Language Processing, ICML 2013.

Dridan, R. and Oepen, S. (2012). Tokenization: Returning to a long solved problem a survey, contrastive experiment, recommendations, and toolkit. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2*, ACL '12, pages 378–382, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fares, M., Oepen, S., and Zhang, Y. (2013). Machine Learning for High-Quality Tokenization Replicating Variable Tokenization Schemes. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing: 14th International Conference, CICLing 2013, Samos, Greece, March 24-30, 2013, Proceedings, Part I*, pages 231–244, Berlin, Heidelberg. Springer.

Frunza, O. (2008). A Trainable Tokenizer, Solution for Multilingual Texts and Compound Expression Tokenization. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Maldonado, A., Han, L., Moreau, E., Alsulaimani, A., Chowdhury, K. D., Vogel, C., and Liu, Q. (2017). Detection of Verbal Multi-Word Expressions via Conditional Random Fields with Syntactic Dependency Features and Semantic Re-Ranking. In *Proceedings of The 13th Workshop on Multiword Expressions*, pages 114–120, Valencia.

Moreau, E., Alsulaimani, A., Maldonado, A., Han, L., Vogel, C., and Chowdhury, K. D. (2017). Semantic Re-Ranking of CRF Label Sequences for Verbal Multi-Word Expression Identification. *To appear*.

R Core Team, (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (2017). The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions. In *Proceedings of the 13th Workshop on Multiword Expressions*, Valencia.

Shao, Y., Hardmeier, C., and Nivre, J. (2017). Recall is the Proper Evaluation Metric for Word Segmentation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, volume 2, pages 86–90.

Wickham, H. (2009). *ggplot2. Elegant Graphics for Data Analysis*. Springer-Verlag New York.

9. Language Resource References

Evang, Kilian and Basile, Valerio and Chrupała, Grzegorz and Bos, Johan. (2013). *Elephant: Sequence Labeling for Word and Sentence Segmentation*. Association for Computational Linguistics.

Koehn, Philipp. (2005). *Europarl: A parallel corpus for statistical machine translation*.

Lavergne, Thomas and Cappé, Olivier and Yvon, François. (2010). *Practical Very Large Scale CRFs*. Association for Computational Linguistics.

Ji Mark and Ondej Bojar. (2012). *TrTok: A Fast and Trainable Tokenizer for Natural Languages*.

Tomas Mikolov and Martin Karafiát and Lukás Burget and Jan Cernocký and Sanjeev Khudanpur. (2010). *Recurrent neural network based language model*. ISCA.

Nivre, Joakim and Agić, Željko and Ahrenberg, Lars and Aranzabe, Maria Jesus and Asahara, Masayuki and Atutxa, Aitziber and Ballesteros, Miguel and Bauer, John and Bengoetxea, Kepa and Bhat, Riyaz Ahmad and Bick, Eckhard and Bosco, Cristina and Bouma, Gosse and Bowman, Sam and Candito, Marie and Cebirolu Eryiit, Gülşen and Celano, Giuseppe G. A. and Chalub, Fabricio and Choi, Jinho and Çöltekin, Çar and Connor, Miriam and Davidson, Elizabeth and de Marneffe, Marie-Catherine and de Paiva, Valeria and Diaz de Ilarraza, Arantza and Dobrovoljc, Kaja and Dozat, Timothy and Droganova, Kira and Dwivedi, Puneet and Eli, Marhaba and Erjavec, Tomaž and Farkas, Richárd and Foster, Jennifer and Freitas, Cláudia and Gajdošová, Katarína and Galbraith, Daniel and Garcia, Marcos and Ginter, Filip and Goenaga, Iakes and Gojenola, Koldo and Gökrmak, Memduh and Goldberg, Yoav and Gómez Guinovart, Xavier and Gonzáles Saavedra, Berta and Grioni, Matias and Grūzītis, Normunds and Guillaume, Bruno and Habash, Nizar and Hajič, Jan and Hà M, Linh and Haug, Dag and Hladká, Barbora and Hohle, Petter and Ion, Radu and Irimia, Elena and Johannsen, Anders and Jørgensen, Fredrik and Kaşkara, Hüner and Kanayama, Hiroshi and Kanerva, Jenna and Kotsyba, Natalia and Krek, Simon and Laippala, Veronika and Lê Hng, Phng and Lenci, Alessandro and Ljubešić, Nikola and Lyashevskaya, Olga and Lynn, Teresa and Makazhanov, Aibek and Manning, Christopher and Mărănduc, Cătălina and Mareček, David and Martínez Alonso, Héctor and Martins, André and Mašek,

Jan and Matsumoto, Yuji and McDonald, Ryan and Missilä, Anna and Mititelu, Verginica and Miyao, Yusuke and Montemagni, Simonetta and More, Amir and Mori, Shunsuke and Moskalevskyi, Bohdan and Muischnek, Kadri and Mustafina, Nina and Müürisep, Kaili and Nguyn Th, Lng and Nguyn Th Minh, Huyn and Nikolaev, Vitaly and Nurmi, Hanna and Ojala, Stina and Osenova, Petya and Øvrelid, Lilja and Pascual, Elena and Passarotti, Marco and Perez, Cenel-Augusto and Perrier, Guy and Petrov, Slav and Piitulainen, Jussi and Plank, Barbara and Popel, Martin and Pretkalnia, Lauma and Prokopidis, Prokopis and Puolakainen, Tiina and Pyysalo, Sampo and Rademaker, Alexandre and Ramasamy, Loganathan and Real, Livy and Rituma, Laura and Rosa, Rudolf and Saleh, Shadi and Sanguinetti, Manuela and Saulīte, Baiba and Schuster, Sebastian and Seddah, Djamé and Seeker, Wolfgang and Seraji, Mojgan and Shakurova, Lena and Shen, Mo and Sichinava, Dmitry and Silveira, Natalia and Simi, Maria and Simionescu, Radu and Simkó, Katalin and Šimková, Mária and Simov, Kiril and Smith, Aaron and Suhr, Alane and Sulubacak, Umut and Szántó, Zsolt and Taji, Dima and Tanaka, Takaaki and Tsarfaty, Reut and Tyers, Francis and Uematsu, Sumire and Uria, Larraitz and van Noord, Gertjan and Varga, Viktor and Vincze, Veronika and Washington, Jonathan North and Žabokrtský, Zdeněk and Zeldes, Amir and Zeman, Daniel and Zhu, Hanzhi. (2017). *Universal Dependencies 2.0*.

Build Fast and Accurate Lemmatization for Arabic

Hamdy Mubarak
QCRI, Hamad Bin Khalifa University (HBKU), Doha, Qatar
hmubarak@hbku.edu.qa

Abstract

In this paper we describe the complexity of building a lemmatizer for Arabic which has a rich and complex morphology, and show some differences between lemmatization and surface stemming, i.e. removing prefixes and suffixes from words. We discuss the need for a fast and accurate lammatization to enhance Arabic Information Retrieval results. We also introduce a new dataset that can be used to test lemmatization accuracy, and an efficient lemmatization algorithm that outperforms state-of-the-art Arabic lemmatization in terms of accuracy and speed. We share the dataset and the code for research purposes.

Keywords: Arabic NLP, Lemmatization, Stemming, Information Retrieval, Diactitization

1. Introduction

Lemmatization is the process of finding the base form (or lemma) of a word by considering its inflected forms. Lemma is also called dictionary form, or citation form, and it refers to all words having the same meaning.

Lemmatization is an important preprocessing step for many applications of text mining and question-answering systems. Researches in Arabic Information Retrieval (IR) systems show the need for representing Arabic words at lemma level for many applications, including keyphrase extraction (El-Shishtawy and Al-Sammak, 2009) and Machine Translation (Dichy and Fargaly, 2003). In addition, lemmatization provides a productive way to generate generic keywords for search engines (SE) or labels for concept maps (Plisson et al., 2004).

Word stem is that core part of the word that never changes even with morphological inflections; the part that remains after prefix and suffix removal. Sometimes the stem of the word is different than its lemma, for example the words: believe, believed, believing, and unbelievable share the stem (believ-), and have the normalized word form (believe) standing for the infinitive of the verb (believe).

While stemming tries to remove prefixes and suffixes from words that appear with inflections in free text, lemmatization tries to replace word suffixes with (typically) different suffix to get its lemma. For languages having complex derivational and inflectional morphology, like Arabic, lemmatization needs more than just suffix replacement as will be described in next section.

This paper is organized as follows: Section 2. gives some background about Arabic morphology and shows some complexities in building Arabic lemmatization; Section 3. lists IR clustering methods and gives examples to show that lemmatization can enhance search results; Section 4. surveys prior work on Arabic stemming and lemmatization; Section 5. introduces the dataset that we created to test lemmatization accuracy; Section 6. describes the algorithm of the system that we built. Results and error analysis are described in section 7.; and Section 8. concludes the paper and lists some tasks for future work.

2. Background

Arabic is the largest Semitic language spoken by almost 300 million people. It's one of the six official languages in the United Nations, and the fifth most widely spoken language after Chinese, Spanish, English, and Hindi[1].

Arabic has a very rich morphology, both derivational and inflectional. Generally, Arabic words are derived from a root that uses three or more consonants to define a broad meaning or concept, and they follow some templatic morphological patterns (الموازين الصرفية). By adding vowels, prefixes and suffixes to the root, word inflections are generated. For instance, the word وسيفتحونها (wsyftHwnhA)[2] "and they will open it" has the triliteral root فتح (ftH), which has the basic meaning of opening, has prefixes و+س (w+s) "and+will", suffixes ون +ها (wn+hA) "they..it", stem يفتح (yftH) "open", and lemma فتح (ftH) "the concept of opening".

Arabic verbs have the following grammatical categories: **tense** (past, present, imperative, and future), **number** (singular, dual, and plural), **person** (first, second, and third), **mood** (indicative, subjunctive, and jussive for present verbs, given for past verbs, and jussive for imperative verbs), **gender** (masculine and feminine) and **voice** (active and passive).
Typically, lemmatization of a verb is achieved by obtaining its past tense without any prefixes or suffixes, singular number, third person, given mood, masculine gender, and active voice. Mapping between different grammatical values cannot be done in many cases by just stripping word from its prefixes and suffixes but by applying some complex morphological rules due to the derivational nature of Arabic morphology. Table 1 shows some examples.

Arabic nouns and adjectives have the following grammatical categories: **case** (nominative, accusative, and genitive), **number** (singular, dual, and plural (proper or broken plu-

[1] https://en.wikipedia.org/wiki/Arabic

[2] Words are written in Arabic, transliterated using Buckwalter transliteration, and translated.

Case	Example
present->past	قال <- يقول (yqwl, qAl) "said, say"
passive->active	ناقش <- نوقش (nwq$, nAq$) "was discussed, discussed"
first->third	نام <- نمت (nmt, nAm) "I slept, he slept"
plural->singular	رضي <- رضوا (rDwA, rDy) "they satisfy, he satisfies"

Table 1: Examples of complex verb lemmatization cases

rals جمع المذكر والمؤنث السالم والتكسير)), **gender** (masculine and feminine) and **definiteness** (definite and indefinite). Typically, lemmatization of a noun or an adjective is achieved by obtaining its nominative case without any prefixes or suffixes, singular number, masculine gender, and indefinite form. Mapping between different values is not straightforward in many cases as shown in Table 2.

Case	Example
broken plural->singular	رجال <- رجل (rjAl, rjl) "men, man"
proper plural->singular	سنوات <- سنة (snwAt, snp) "years, year"
feminine->masculine	أخضر <- خضراء (xDrA', >xDr) "green (f), green (m)"
genitive->nominative	بناء <- بنائه (bnA}h, bnA') "building-it, building"
special cases	مستشفى <- مستشفيات (mst$fyAt, mst$fY) "hospitals, hospital"
	فيديو <- فيديوهات (fydywhAt, fydyw) "videos, video"...

Table 2: Examples of complex noun lemmatization cases

In addition, according to Arabic morphology and writing system, attaching pronouns to words in some cases changes their last letter. This adds an extra complexity when obtaining lemmas. For example, when nouns ending with Taa-Marbouta letter are attached to possessive pronouns, it will be changed to Taa letter as in حضارة+ه <- حضارته (hDArp+h, HDArth) "its civilization". Also, when verbs ending with Alif-Maqsoura letter are attached to some subject pronouns, it will be changed to Yaa letter as in هدى +نا <- هدينا (hdY+nA, hdynA) "we guided" or even deleted in some cases as in هدى+ وا <- هدوا (hdY+wA, hdwA) "they guided", and when are attached to object pronouns, it will be changed to Alif letter as in هدى+ ها <- هداها (hdY+hA, hdAhA) "guides her", etc.

The mentioned cases are just few examples to show how complex the Arabic lemmatization is, and reveal that many cases should be considered in addition to stripping words from prefixes and suffixes to get their proper lemmatization.

3. Lmmatization and IR

IR systems normally cluster words together into groups according to three main levels: root, stem, or lemma. The root level is considered by many researchers in the IR field which leads to high recall but low precision due to language complexity. For example words يكتب، مكتبة، كتاب (yktb, mktbp, ktAb) "he writes, library, book" have the same root كتب (ktb) with the basic meaning of writing. Therefore, searching for any of these words by root, yields getting the other words which may not be desirable for many users.

Other researchers show the importance of using stem level for improving retrieval precision and recall as they capture semantic similarity between inflected words. However, in Arabic, stem patterns may not capture similar words having the same semantic meaning. For example, stem patterns for broken plurals are different from their singular patterns, e.g. the stem of the plural word أقلام (AqlAm) "pens" does not match the stem of its singular form قلم (qlm) "pen". The same applies to many imperfect verbs that have different stem patterns than their perfect verbs, e.g. the verbs استطاع، يستطيع (AstTAE, ystTyE) "he could, he can" do not match because they have different stems. Indexing using lemmatization can enhance the performance of Arabic IR systems as reported in (El-Shishtawy and El-Ghannam, 2012), and in pactice, lemmatization should be very fast and accurate to be used in IR systems.

4. Related Work

A lot of work has been done in word stemming and lemmatization in different languages, for example the famous Porter stemmer for English, but for Arabic, few works have been done especially in lemmatization, and there is no open-source code and new testing data that can be used by other researchers for word lemmatization.

Xerox Arabic Morphological Analysis and Generation (Beesley, 1996) is one of the early Arabic stemmers, and it uses morphological rules to obtain stems for nouns and verbs by looking into a table of thousands of roots.

Khoja's stemmer (Khoja, 1999) and Buckwalter morphological analyzer (Buckwalter, 2002) are other root-based analyzers and stemmers which use tables of valid combinations between prefixes and suffixes, prefixes and stems, and stems and suffixes.

Recently, MADAMIRA (Pasha et al., 2014) system has been evaluated using a blind testset of 25K words for Modern Standard Arabic (MSA) selected from Penn Arabic Tree bank (PATB). They reported an accuracy of 96.2% as the percentage of words where the chosen analysis (provided by SAMA morphological analyzer (Graff et al., 2009)) has the correct lemma.

In this paper, we present an open-source Java code to extract Arabic lemmas, and a new publicly available testset for lemmatization allowing researches to evaluate using the same dataset, and reproduce results.

5. Data Description

To make the annotated data publicly available, we selected 70 news articles from Arabic WikiNews site `https://ar.wikinews.org/wiki`. These articles cover recent news from year 2013 to year 2015 in multiple genres (politics, economics, health, science and technology, sports, arts, and culture.) Articles contain 18,300 words, and they are evenly distributed among these 7 genres with 10 articles per each.

Words were white-space and punctuation separated, and some spelling errors were corrected (1.33% of the total words) to have a very clean testset. Lemmatization was done by an expert Arabic linguist where spelling corrections were marked, and lemmas were provided with full diacritization. Sample is shown in Figure 1.

As MSA is usually written without diacritics and IR systems normally remove them from search queries and also from indexed data as a basic preprocessing step, so another column for undiacritized lemma was added. This column was used to evaluate our lemmatizer and to compare with state-of-the-art systems for lemmatization (MADAMIRA), and segmentation and surface stemming (Farasa).

The raw sentences of the testset can be downloaded from the link: `http://alt.qcri.org/~hmubarak/WikiNews-26-06-2015.txt` and the annotation for lemmatization from the link: `http://alt.qcri.org/~hmubarak/WikiNews-26-06-2015-RefLemma.xlsx`

refLemmaUndiac	refLemma	spell	corrWord	orgWord
إعلام	إِعْلَام		للإعلام	للإعلام
بديل	بَدِيل		البديل	البديل
حر	حُرّ		والحر	والحر
،	،		،	،
دعا	دَعَا	1	ودعا	ودعى
حضور	حُضُور		حضور	حضور
مؤتمر	مُؤْتَمَر		المؤتمر	المؤتمر
من	مِنْ		من	من
مهتم	مُهْتَمّ		المهتمّين	المهتمّين

Figure 1: Lemmatization of WikiNews corpus

6. System Description

We were inspired by the work done by (Darwish and Mubarak, 2016) for segmenting Arabic words out of context. They achieved an accuracy of almost 99%; slightly better than state-of-the-art system for segmentation (MADAMIRA) which considers surrounding context and many linguistic features. This system shows enhancements in both Machine Translation, and Information Retrieval tasks (Abdelali et al., 2016). This work can be considered as an extension to word segmentation.

We used a fully diacritized corpus created by a commercial vendor which contains 9.7 million words with almost 200K unique surface words. About 73% of the corpus is in MSA and covers variety of genres like politics, economy, sports, society, etc. and the remaining part is mostly religious texts written in classical Arabic (CA). (Darwish et al., 2017) used this corpus to build state-of-the-art diacritizer with word error rates (WER) of 3.29% and 12.76% in diacritization of stem and grammatical case ending in order.

From this corpus, we constructed a dictionary of words and their possible diacritizations ordered by number of occurrences of each diacritized form. For example, the word وبنود (wbnwd) "and items" is found 4 times in this corpus with two full diacritization forms وَبُنُودٍ ، وَبُنُودِ (wabunudi, wabunudK) "and items, with different grammatical case endings" which appeared 3 times and once respectively. All unique undiacritized words in this corpus were analyzed using Buckwalter morphological analyzer which gives all possible word analyses, and for each analysis it provides its diacritization, segmentation, **lemma** and part-of-speech (POS) tag as shown in Figure 2.

INPUT STRING: وبنود

LOOK-UP WORD: wbnwd

SOLUTION 1: (**wabunuwd**) [**banod**_1] wa/CONJ+bunuwd/NOUN

(GLOSS): and + articles/clauses +

SOLUTION 2: (**wabinawod**) [**nawod**_1] wa/CONJ+bi/PREP+nawod/NOUN

(GLOSS): and + by/with + swaying/swinging +

Figure 2: Buckwalter analysis (diacritization forms and lemmas are highlighted)

The idea is to take the most frequent diacritized form for words that appear in this corpus, and find the morphological analysis with highest matching score between its diacritized form and the corpus diacritized word. This means that we search for the most common diacritization of words regardless of their surrounding contexts. In the above example, the first solution is preferred and hence its lemma بند (banod, bnd after diacritics removal) "item", and the other less frequent analysis is ignored.

While comparing two diacritized forms from the corpus and Buckwalter analysis, many special cases were applied to solve inconsistencies between the two diacritization schemas, for example while words are fully diacritized in the corpus, Buckwalter analysis gives diacritics without case ending (i.e. without context), and it removes short vowels in some cases, for example before long vowels, and after the definite article ال (Al) "the", etc.

It is worth mentioning that there are many cases in Buckwalter analysis where for input word, there are two

or more identical diacritizations with different lemmas, and the analyses in these cases are provided in a random order. For example the word سيارة (syArp) "car" has two morphological analyses with different lemmas; سيار (syAr) "walker", and سيارة (syArp) "car" in this order while the second lemma is the most common one. To solve this problem, all such words were reported and the top frequent words were revised to insure that their lemmas are sorted according to actual usage in a modern large corpus [3].

The lemmatization algorithm can be summarized in Figure 3, and the code can be tested and downloaded from Farasa site: `farasa.qcri.org`. It can be called from the command line as a Java package (.jar) using the following syntax:
farasa –lemma -i <*inFilename*> **-o** <*outFilename*>

Figure 4 shows system output for a sample sentence where errors are highlighted.

7. Evaluation

Lemmatization outputs of MADAMIRA and our system were compared against the undiacritized reference lemma for each word. We evaluated also surface stemming of Farasa segmenter (Darwish and Mubarak, 2016) (i.e. the remaining part after removing prefixes and suffixes) to quantify the improvement in lemmatization accuracy after applying the suggested algorithm.

For more accurate results, all differences were revised manually to accept cases that should not be counted as errors, for example in different writings for foreign named entities such as هونج كونج and هونغ كونغ (hwng kwng, hwnj kwnj) "Hong Kong".

Table 3 shows results of testing our system, MADAMIRA and Farasa segmenter as surface stemmer on the WikiNews testset (for undiacritized lemmas). Our approach gives +7% and +32% relative gains above MADAMIRA and Farasa segmenter respectively in lemmatization task.

System	Accuracy
Farasa segmenter (surface stemmer)	73.68%
MADAMIRA	96.61%
Our lemmatization System	**97.32%**

Table 3: Lemmatization accuracy using WikiNews testset

In terms of speed, our system was able to lemmatize 7.4M words on a personal laptop in 2 minutes compared to 2.5 hours for MADAMIRA which does the full morphological analysis and disambiguation, lemmatization, POS tagging, named entity recognition, and diacritization.

The code is written entirely in Java without any external dependency which makes its integration in other systems quite simple.

[3]We used text from www.Aljazeera.net which contains 100M words (archive of 10 years)

7.1. Error Analysis

Most errors in our system are due to using only the most frequent diacritization of words without considering their contexts. This cannot solve ambiguity in cases like when nouns and adjectives share the same diacritization forms, e.g. the word أكاديمية (AkAdymyp) can be either noun and its lemma is أكاديمية (AkAdymyp) "academy", or adjective and its lemma is أكاديمي (AkAdymy) "academic".

For MADAMIRA, most errors came from selecting incorrect POS tag, hence lemma, for ambiguous words (ex: the word محاضرات (mHADrAt) "lectures" was mistakenly tagged as adjective محاضر (mHADr) "lecturer" instead of the correct noun محاضرة (mHADrp) "lecture"). Another source of errors is the wrong segmentation of named entities, ex: the word البايثون (AlbAyvwn) "the+Python" which should be segmented as ال+بايثون (Al+bAyvwn) "the Python", i.e. split the definite article ال (Al) "the".

For Farasa segmenter (surface stemmer), errors came from not supporting complex cases described in Section 2..

8. Conclusion

In this paper, we list some complexities in building lemmatization for Arabic due to its rich morphology and complex writing system. We introduce a new testset for lemmatization and a very fast and accurate algorithm that performs better than state-of-the art lemmatization system; MADAMIRA. It also outperforms state-of-the-art word segmenter (surface stemmer); Farasa segmenter.

From a large fully diacritized corpus, possible diacritizations of words are extracted, and the algorithm considers only the most frequent diacritized form for words out of context. It gets the best similarity matching score between this diacritized form and the morphological analysis, provided by Buckwalter morphological analyzer, which contains word lemma. Both the testset and the code are publicly available for researchers.
We plan to study the performance when we consider sourrouning context, also to provide diacritized lemmas which can be useful for other applications. In addition, we plan to plug the lemmatizer into an IR system (Solr for example), and carry out an extrinsic evaluation to evaluate performance with lemmatization also versus other systems such as MADAMIRA and Farasa surface stemmer.

9. Bibliographical References

Abdelali, A., Darwish, K., Durrani, N., and Mubarak, H. (2016). Farasa: A fast and furious segmenter for arabic. In *HLT-NAACL Demos*, pages 11–16.

Beesley, K. (1996). Arabic finite-state morphological analysis and generation. In *In COLING-96: Proceedings of the 16th international*, pages 89–94.

Buckwalter, T. (2002). Arabic finite-state morphological analysis and generation. In *http://members.aol.com/ArabicLexicons/*.

```
Remove diacritics from input buffer
For each word in the input buffer
    Search for the word in dictionary that contains words, diacritizations, and lemmas
    If found
        Take the first lemma
    else
        Segment word into clitics (using Farasa segmentation)
        Skip prefixes from the segmented word and get the first clitic (mostly stem) after prefixes
        Handle special cases for some suffixes, e.g. taa marbouta and Hamza on yaa and waw
        Search for the stem in the dictionary
        if found
            Take the first lemma
        else
            Lemma = stem
```

Figure 3: Summary of lemmatization algorithm

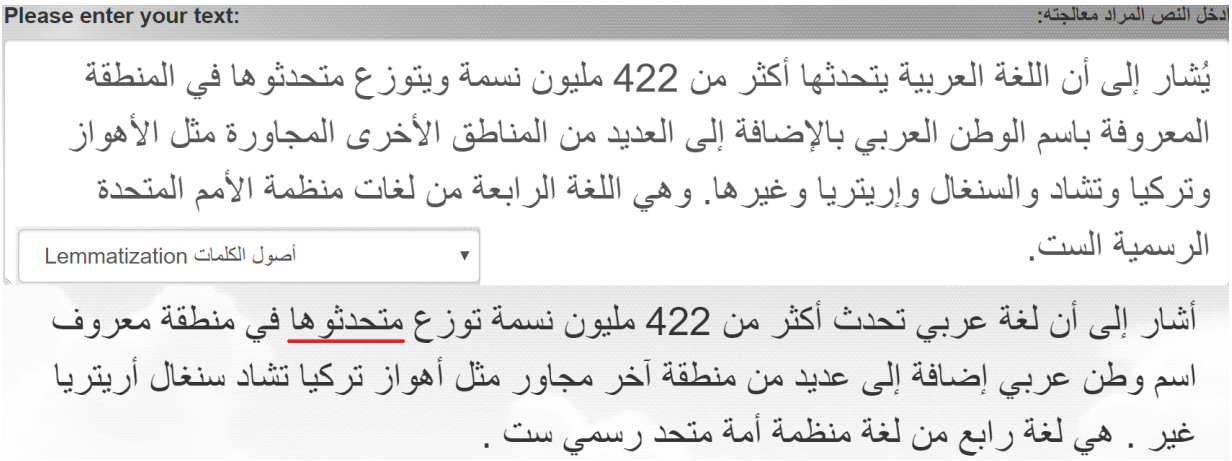


Figure 4: Lemmatization online demo (part of Farasa Arabic NLP tools)

Darwish, K. and Mubarak, H. (2016). Farasa: A new fast and accurate arabic word segmenter. In *LREC*.

Darwish, K., Mubarak, H., and Abdelali, A. (2017). Arabic diacritization: Stats, rules, and hacks. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 9–17.

Dichy, J. and Fargaly, A. (2003). Roots and patterns vs. stems plus grammar-lexis specifications: on what basis should a multilingual lexical database centred on arabic be built? In *Proceedings of the MTSummit, New-Orleans*.

El-Shishtawy, T. and Al-Sammak, A. (2009). Arabic keyphrase extraction using linguistic knowledge and machine learning techniques. In *Proceedings of the Second International Conference on Arabic Language Resources and Tools, The MEDAR Consortium*.

El-Shishtawy, T. and El-Ghannam, F. (2012). An accurate arabic root-based lemmatizer for information retrieval purposes. *arXiv preprint arXiv:1203.3584*.

Graff, D., Maamouri, M., Bouziri, B., Krouna, S., Kulick, S., and Buckwalter, T. (2009). Standard arabic morphological analyzer (sama) version 3.1. In *Linguistic Data Consortium LDC2009E73*.

Khoja, S. (1999). Stemming arabic text. In *Computing Department, Lancaster University*.

Pasha, A., Al-Badrashiny, M., Diab, M., El Kholy, A., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. M. (2014). Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. *Proc. LREC*.

Plisson, J., Lavrac, N., and Mladenic, M. (2004). A rule based approach to word lemmatization. In *researchgate.net*.

JESC: Japanese-English Subtitle Corpus

Reid Pryzant[1], **Youngjoo Chung**[2], **Dan Jurafsky**[1], **Denny Britz**[3]
[1]Stanford University, [2]Rakuten Institute of Technology, [3]Google Brain
{rpryzant, jurafsky}@stanford.edu, yjchung@acm.org, dennybritz@gmail.com

Abstract

In this paper we describe the Japanese-English Subtitle Corpus (JESC). JESC is a large Japanese-English parallel corpus covering the underrepresented domain of conversational dialogue. It consists of more than 3.2 million examples, making it the largest freely available dataset of its kind. The corpus was assembled by crawling and aligning subtitles found on the web. The assembly process incorporates a number of novel preprocessing elements to ensure high monolingual fluency and accurate bilingual alignments. We summarize its contents and evaluate its quality using human experts and baseline machine translation (MT) systems.

Keywords: parallel corpus, asian languages, machine translation

1. Introduction

There is a strong need for large parallel corpora from new domains. Modern machine translation (MT) systems are fundamentally constrained by the availability and quantity of parallel corpora. Apart from the exceptions of English-Arabic, English-Chinese, and several European pairs, parallel corpora remain a scarce resource due to the high cost of manual construction (Chu et al., 2014). Furthermore, despite promising work in domain adaptation, MT systems struggle to generalize to new domains that are disparate from their training data (Pryzant et al., 2017).

This need for large, novel-domain data is especially evident in the resource-poor Japanese-English (JA-EN) language pair. Only two large (>1M phrase pairs) and free datasets exist for this language pair (Neubig, 2017; Tiedemann, 2017; Moses, 2017). The first is called ASPEC. It consists of 3M examples and it originates from scientific papers, a highly formalized and written domain (all other JA-EN datasets have similar language) (Nakazawa et al., 2016). The other, OpenSubtitles, is a multi-language dataset of aligned subtitles authored by professional translators; the JA-EN subset of these data contains approximately 1M examples (Lison and Tiedemann, 2016). OpenSubtitles is to the best of these authors knowledge the *only* parallel corpus to cover the unrepresented domains of conversational speech and informal writing. This dearth of large-scale and informal data is especially problematic because colloquial Japanese has significant structural characteristics which can preclude cross-domain translation (Tsujimura, 2013). We hope to alleviate this problem by building off the work of (Lison and Tiedemann, 2016) to construct a larger corpus that incorporates the vast number of unofficial and fan-made subtitles on the web.

Subtitles are an excellent source for alleviating resource scarcity problems. There are a wide and interesting range of linguistic phenomena in subtitles that are poorly represented elsewhere. This includes colloquial exchange, slang, expository discourse, dialect, vernacular, and movie dialog, which is available in great quantities and has been shown to resemble natural conversation (Forchini, 2013). Furthermore, large subtitle databases are freely available on the web, are often crowd-sourced, and the close correspondence between subtitles and their video material renders time-based alignment feasible (Tiedemann, 2008).

We release JESC, a new Japanese-English parallel corpus consisting of 3.2 million pairs of crawled TV and movie subtitles[1]. We also release the tools, crawlers, and parsers used to create it. We provide a comprehensive statistical summary of their contents as well as strong baseline machine translation systems that yield competitive BLEU scores. This is the largest freely available Japanese-English dataset to date and covers the resource-poor domain of conversational or informal speech.

2. Source Data

We crawled four free and open subtitle repositories for Japanese and English subtitles: `kitsunekko.net`, `d-addicts.com`, `opensubtitles.org`, and `subscene.com`. Each subtitle database accepts submissions from the public and disseminates them through a web interface. There is no standard imposed on subtitle submissions, and as such, they exist in a plenitude of file formats, encodings, languages (beyond that being advertised), and content (beyond that being advertised). Though some of these subtitles are indeed the "official" translation, many were translated or transcribed by amateur fans of the video content. Thus, many of our translations are crowd-sourced, and there are no guarantees on the fluency of the participants. Many subtitle files contained grammatical, spelling, optical character recognition (OCR), and a host of other problems that preclude their direct usage for machine translation.

Crawling these online repositories yielded 93,992 subtitle files corresponding to 23,318 individual titles (episodes, etc.), 4,610 grouped titles (shows, etc.), and more than 100 million individual captions corresponding to a broad range of video material (Figure 1). Our objective is to automatically cull a high quality parallel corpus from this unstructured and error-prone data.

[1]The dataset and code are available at `cs.stanford.edu/~rpryzant/jesc/`

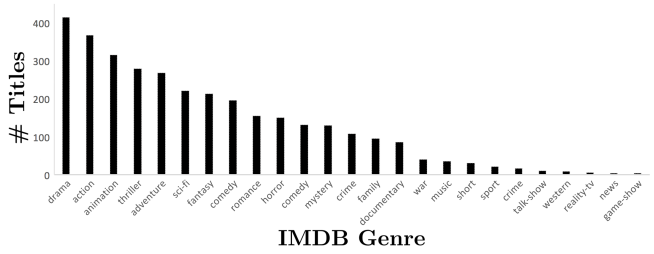


Figure 1: Genre distribution for our crawled titles (obtained via IMDB).

3. Preprocessing

Due to the acute heterogeneity and high error rate of our subtitle files, we underwent a number of preprocessing steps to bring them into a form suitable for alignment. The output of this preprocessing pipeline is a series of documents (one per subtitle file), each structured as a titled list of captions, start times, and end times.

3.1. Document Standardization

First, we converted each subtitle file into a standardized format. We applied the `chardet` library to determine the most likely encoding (Li and Momoi, 2001), and converted this encoding to a `utf-8` standard. We then used the `ffmpeg` library to convert files into a common SubRip (`.srt`) format (Tomar, 2006). Files that `ffmpeg` was unable to convert were interpreted as illegitimate and discarded. Last, we parsed these `.srt` documents into machine-readable YAML[2]. Each resulting document contains a title (obtained from `.srt` metadata) and a list of captions, with each caption consisting of tokenized body text, start time, and end time.

3.2. Text Correction

Next, we preprocessed the English documents by performing syntax correction on each caption. Many fan-made subtitles were created by non-native English speakers and as such contained typographical and spelling mistakes. We developed a laplace-smoothed statistical error model $P(w|w^*)$ that scores the probability of a word w^* being misspelled as w. This model was trained by observing relative misspelling frequencies on the Birkbeck corpus (Mitton, 1985). We then developed two additional laplace-smoothed frequency-based models using unigrams and bigrams from Google's Web 1T N-grams (Islam and Inkpen, 2009). These are language models that score the prior probability of n-gram occurrence, $P(w)$, and the transition probability $P(w_i|w_{i-1})$. We used a smoothing factor of $\alpha = 1$ for all of these models. Next, for each possibly misspelled token t_i of a caption c, we performed depth-4 uniform cost search on the space of edits to produce candidate replacements t_i^*. Armed with the error model $P(t_i|t_i^*)$ and language model $P(t_i^*)P(t_i^*|t_{i-1})$, we scored the probability of each candidate by applying Bayes rule, similar to (Lison and Tiedemann, 2016):

$$P(t_i^*|t_i, t_{i-1}) = P(t_i|t_i^*)P(t_i^*)P(t_i^*|t_{i-1})$$

Note that this checker improves on that of (Lison and Tiedemann, 2016) with the inclusion of a data-driven error model, prior term, and depth-4 uniform cost search (as opposed to making any correction with >50% probability). We also standardized the text of each caption by lowercasing, removing bracketed text, out-of-language subsequences (e.g. encoding errors, OCR errors, machine-readable tags), linguistic queues (i.e. "laughs"), inappropriate punctuation (e.g. leading dashes, trailing commas), and author signatures.

4. Cross-lingual Alignment

Once these subtitle files are brought into a suitable form, they can be aligned to form a parallel corpus. Doing so requires alignment at two levels: the document level, where we group subtitles according to the movie or TV show they correspond too, and the caption level, where we determine which captions are direct translations of one another.

4.1. Document Alignment

In order to align subtitles across distinct languages we must first align the documents themselves, i.e. determine which subtitle documents' captions are worth aligning. This is because (1) multiple subtitle documents may exist for a given movie or TV episode, and (2) subtitles from non-matching movies or TV shows will not be in correspondence.

We generated candidate alignments between Japanese and English documents with a novel technique involving soft matching on file metadata. We first extracted metadata in the form of movie and TV show names as well as episode numbers from each document title. Next, we used the Ratcliff-Obershelp algorithm to determine pairwise title similarities (this algorithm determines similarity via the lengths of matching subsequences), matching two files if their similarity ratio exceeded 90% (Ratcliff and Metzener, 1998). We proceeded to filter out pairs with differing episode numbers.

We refined document alignments with another novel method which considers the temporal sequence of their captions. We created document vectors $D_i = [d_1^i, ..., d_{10,000}^i]$ for each subtitle file i. Each feature d_k^i is a binary indicator that is active when document i has a caption whose closest starting second is k. To account for possible time shift errors, we constructed a multiplicity of vectors for each document, each shifted to a different start time. We then computed the Hamming distance between each Japanese-English document vector and discarded those pairs with a distance greator than 0.04 (chosen based on a bucketed distribution of distances between all pairs).

4.2. Caption Alignment

Now that we have a set of matched English and Japanese subtitle files $\{(\boldsymbol{E}_1, \boldsymbol{J}_1), ..., (\boldsymbol{E}_n, \boldsymbol{J}_n)\}$, we must align the captions of each pair such that captions which are direct translations of each other are selected for extraction.

Let $\boldsymbol{E} = e_1, ..., e_n$ and $\boldsymbol{J} = j_1, ..., j_m$ be a pair of aligned English and Japanese documents that presumably map to

[2] `http://www.yaml.org/`

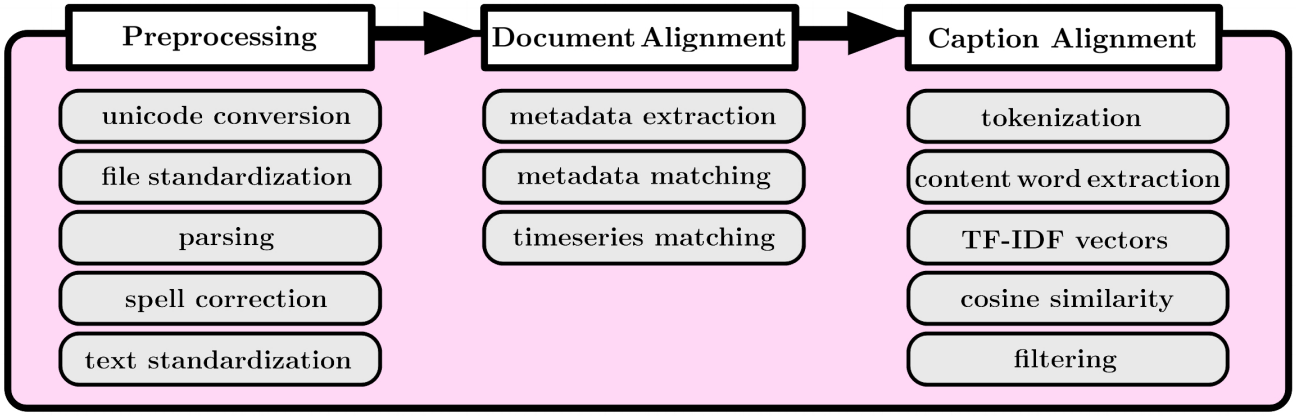


Figure 2: Workflow for the creation of parallel corpora from raw subtitle files.

similar video content. Note that each e_i and j_i are subtitle captions consisting of a start time a_i, end time b_i, and a sequence of text tokens $t_1, ..., t_z$ (in Japanese or English). If $\boldsymbol{E}$ and $\boldsymbol{J}$ were in perfect harmony then we would be able to pair e_1 with j_1, e_2 with j_2 and so on. However, matched documents are rarely in such close correspondence. Optical Character Recognition (OCR) errors, misaligned files, differing start times, speed ratios, framerate, and a host of other problems preclude such a one-to-one correspondence (Tiedemann, 2016).

Due to the severity of the aforementioned problems, especially among documents that have been subtitled by amateur translators, we found existing caption alignment algorithms inadequate for our needs. We developed a novel subtitle alignment algorithm that matches captions based on both timing and content. For each Japanese caption, we search a nearby window (typically 10-15 seconds) of English captions and score their similarity. We then take the highest-scoring match of this window.

We score the quality of an English-Japanese caption pairing by (1) morphologically analyzing Japanese and English captions and discarding all but the content words, then (2) stemming these content words, (3) translating the Japanese to English with simple dictionary lookups, (4) averaging the GLoVE vectors for each caption's words, and (5) computing the cosine similarity between these vector representations. We used the Rakuten and JUMAN morphological analyzers to extract content words from Japanese captions, and the Stanford POS tagger for English (Hagiwara and Sekine, 2014; Manning et al., 2014). We used JUMANPP (Morita et al., 2015) and NLTK to stem these words (Bird, 2006), and JMdict/EDICT to map Japanese words to their English equivalents (Breen, 2004; Matsumoto et al., 1991). Phrases without translations were skipped. Note that our method introduces a bias in the phrase pairs of resultant matches, namely those pairs that would score highly under a lexicon, but we assume that JMdict/EDICT is near-complete with respect to common content words.

4.3. Filtering

The document- and caption-matching procedures outlined above produced 27,716,868 matches between English and Japanese captions. We proceeded by filtering out low-quality matches, choosing to retain only the very highest quality matches. We discarded matches whose cosine similarity was below the 84^{th} percentile (assuming a normal distribution), leaving 4,434,699 pairs. This percentile was chosen based on downstream NMT performance. Last, we removed duplicate matches and out-of-language matches (matches where $< 90\%$ of the characters in e or $> 10\%$ of the letters in j are roman), leaving us with a final count of 3,240,661 phrase pairs.

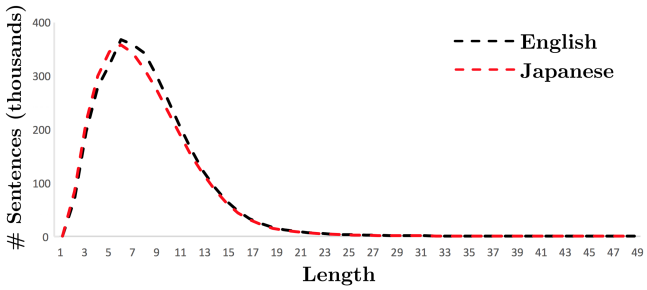


Figure 3: JESC exhibits a right-skewed sentence length distribution. 83 English and 114 Japanese phrases have length > 50 .

5. Investigation

5.1. Basic statistics

The resulting corpus, which we call JESC, for Japanese English Subtitle Corpus, consists of 29,368 unique English words and 87,833 unique Japanese words. The train/val/test splits are of size 3,236,660 / 2000 / 2001. The lengths of each languages' phases are quite similar (Figure 3). JESC consists mainly of short bursts of conversational dialogue; the average English sentence length is 8.32 words while for Japanese it is 8.11.

JESC also exhibits multiple reference translations for 163,665 and 130,790 Japanese and English phrases, respectively. For example, the English sentence "what?" has translations such as 何だ?/なんだって？/な　なんだよ？/どうしたんですか? due to variations in the Japanese

English	Japanese
look, i don't do that shit anymore.	私は卒業した
thank you! you're so sweet	ありがとう
look, his name is cyrus gold.	いいか 彼の名前はサイラス・ゴールド
is that so? i hate to disappoint you.	そうか それは残念だったな。

Table 1: Samples from JESC.

suffix determined by the circumstances of the speaker and dialogue situation. This feature makes it unique among large parallel corpora and greatly improves its usefulness. While BLEU is designed to benefit from multiple reference translations (Papineni et al., 2002), this is a luxury rarely afforded to the modern system, and both of the major MT workshops use single-reference BLEU to evaluate all their tasks[3][4].

5.2. Evaluation

5.2.1. Alignment evaluation

We checked the validity of bilingual sentence alignments based on the procedure of (Utiyama and Isahara, 2007). A pair of human evaluators (both native Japanese and proficient English speakers) randomly sampled 1000 phrase pairs. On average, 75% of these pairs were perfectly aligned, 13% partially aligned, and 12% misaligned. There was strong agreement between these adjudicators' findings (Cohen's kappa of 0.76) so we may conclude that JESC is noisy but has significant signal that can be useful for downstream applications.

5.2.2. Translation evaluation

In addition to alignment, we evaluated the quality of crowd-sourced translation. Our evaluators used the Japanese Patent Office's adequacy criterion (JPO). The JPO is a 5-point system which provides strong guidelines for scoring the quality of a Japanese-English translation pair (Nakazawa et al., 2016). Again in the style of (Utiyama and Isahara, 2007) we sampled and evaluated 200 phrase pairs from the pool of non-misaligned phrases, observing an average JPO adequacy score of 4.82, implying the amateur and crowd-sourced translations are high quality.

5.2.3. Machine translation performance

We also evaluated JESC with downstream Machine Translation performance, using the TensorFlow and Sequence-to-Sequence frameworks (Abadi et al., 2016; Britz et al., 2017; Lison and Tiedemann, 2016). We used a 4-layer bidirectional LSTM encoder and decoder with 512 units, as well as dot-product attention (Luong et al., 2015). We applied Dropout at a rate of 0.2 to the input of each cell, and optimized using Adam and a learning rate of 0.0001 (Kingma and Ba, 2014). We used a batch size of 128, and train for 10 epochs. For each experiment, we preprocess the data using learned subword units[5] (Sennrich et al., 2015) for a shared vocabulary of 16,000 tokens.

In addition to evaluating JESC, we trained and tested on the ASPEC corpus of (Nakazawa et al., 2016) which consists of scientific abstracts (3M examples), the Kyoto Wiki Corpus (KWC) of (Chu et al., 2014) which consists of translated Wikipedia articles (0.5M examples), and the OpenSubs corpus of (Lison and Tiedemann, 2016) which is the closest analog to JESC and consists of 1M professionally-made and automatically aligned captions.

Train/Test	ASPEC	KWC	OpenSubs	JESC
ASPEC	**36.23**	15.42	3.45	3.81
KWC	5.30	**8.61**	2.31	2.22
OpenSubs	0.2	0.7	**10.01**	6.3
JESC	2.35	3.71	8.8	**14.21**

Table 2: Machine translation results (BLEU).

Even though KWC consists of high quality and human-made translations, we find that it underperforms due to the small size of the dataset (Table 2). Similarly, we find that JESC's large size helps it outperform OpenSubs in both in-domain BLEU and out-of-domain generalization.

6. Conclusion

We introduced JESC, a large-scale parallel corpus for the Japanese-English language pair. JESC is (1) the largest publicly available Japanese-English corpus to date, (2) a corpus that covers the underrepresented domain of conversational speech, and (3) to the extent of these authors knowledge, the only large-scale parallel corpus to support multi-reference BLEU evaluation. Our experimental results suggest that these data are a high quality and novel challenge for today's machine translation systems. By releasing these data to the public, we hope to increase the colloquial abilities of today's MT systems, especially for the Japanese-English language pair.

7. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

[3] http://www.statmt.org/wmt17/

[4] http://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2017/

[5] https://github.com/google/sentencepiece

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Breen, J. (2004). The edict project.

Britz, D., Goldie, A., Luong, T., and Le, Q. (2017). Massive exploration of neural machine translation architectures. *arXiv preprint arXiv:1703.03906*.

Chu, C., Nakazawa, T., and Kurohashi, S. (2014). Constructing a chinese-japanese parallel corpus from wikipedia. In *LREC*, pages 642–647.

Forchini, P. (2013). Using movie corpora to explore spoken American English: Evidence from multidimensional analysis. In Julia Bamford, et al., editors, *Variation and Change in Spoken and Written Discourse: Perspectives from corpus linguistics*, pages 123–136. Benjamins.

Hagiwara, M. and Sekine, S. (2014). Lightweight client-side chinese/japanese morphological analyzer based on online learning. In *COLING (Demos)*, pages 39–43.

Islam, A. and Inkpen, D. (2009). Real-word spelling correction using google web it 3-grams. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1241–1249. Association for Computational Linguistics.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *International Conference for Learning Representations (ICLR)*.

Li, S. and Momoi, K. (2001). A composite approach to language/encoding detection. In *Proc. 19th International Unicode Conference*, pages 1–14.

Lison, P. and Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *LREC*.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Matsumoto, Y., Kurohashi, S., Nyoki, Y., Shinho, H., and Nagao, M. (1991). User's guide for the juman system, a user-extensible morphological analyzer for japanese. *Nagao Laboratory, Kyoto University*.

Mitton, R. (1985). Birkbeck spelling error corpus. *Retrieved 03/11/2017 from http://ota. ahds. ac. uk*.

Morita, H., Kawahara, D., and Kurohashi, S. (2015). Morphological analysis for unsegmented languages using recurrent neural network language model. In *EMNLP*, pages 2292–2297.

Moses. (2017). Parallel corpora available on-line. *http://www.statmt.org/moses/?n=Moses.LinksToCorpora*. Accessed 8/22/17.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). Aspec: Asian scientific paper excerpt corpus. In *LREC*.

Neubig, G. (2017). Japanese parallel data. *http://www.phontron.com/japanese-translation-data.php*. Accessed 8/22/17.

Papineni, K., Roukos, S., Ward, T., and jing Zhu, W. (2002). Bleu: a method for automatic evaluation of machine translation. pages 311–318.

Pryzant, R., Britz, D., and Le, Q. (2017). Effective domain mixing for neural machine translation. In *Second Conference on Machine Translation (WMT)*.

Ratcliff, J. and Metzener, D. (1998). Ratcliff-obershelp pattern recognition.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units. *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Tiedemann, J. (2008). Synchronizing translated movie subtitles. In *LREC*.

Tiedemann, J. (2016). Finding alternative translations in a large corpus of movie subtitle. In *LREC*.

Tiedemann, J. (2017). Opus, the open parallel corpus. *http://opus.lingfil.uu.se/*. Accessed 8/22/17.

Tomar, S. (2006). Converting video formats with ffmpeg. *Linux Journal*, 2006(146):10.

Tsujimura, N. (2013). *An introduction to Japanese linguistics*. John Wiley & Sons.

Utiyama, M. and Isahara, H. (2007). A japanese-english patent parallel corpus. *MT summit XI*, pages 475–482.

Building a Corpus for Personality-dependent Natural Language Understanding and Generation

R.M.S. Ramos, G.B.S. Neto, B.B.C. Silva, D.S. Monteiro, I. Paraboni, R.F.S. Dias
University of São Paulo, School of Arts, Sciences and Humanities
São Paulo, Brazil
{ricellimsilva,georges.stavracas,barbarab.claudino,danisamon,ivandre.paraboni,rafaelsandronidias}@gmail.com

Abstract

The computational treatment of human personality - both for the recognition of personality traits from text and for the generation of text so as to reflect a particular set of traits - is central to the development of NLP applications. As a means to provide a basic resource for studies of this kind, this article describes the b5 corpus, a collection of controlled and free (non-topic specific) texts produced in different (e.g., referential or descriptive) communicative tasks, and accompanied by inventories of personality of their authors and additional demographics. The present discussion is mainly focused on the various corpus components and on the data collection task itself, but preliminary results of personality recognition from text are presented in order to illustrate how the corpus data may be reused. The b5 corpus aims to provide support for a wide range of NLP studies based on personality information and it is, to the best of our knowledge, the largest resource of this kind to be made available for research purposes in the Brazilian Portuguese language.

Keywords: Corpora, Personality, Big Five

1. Introduction

In recent years, the development of so-called intelligent systems has devoted a great deal of attention to the computational treatment of human personality. This interest may be explained, among other reasons, by the practical observation that users of computer systems not only attribute human traits to the systems they interact with, but they also prefer those systems that present traits similar to their own (Mairesse et al., 2007).

Fundamental personality traits are consistently reflected in the language choices made by individuals when communicating. For instance, an individual with narcissistic traits might make frequent use of first-person expressions ('I', 'for me', etc.). The relation between personality and natural language is the focus of a large body of work in the Psychology field, and it is perhaps best summarised by the *Big Five* personality factors (Goldberg, 1990) - Openness to experience, Conscientiousness, Extraversion, Agreeableness and Neuroticism - which are widely accepted as an adequate basis for the representation of human personality.

Given its linguistic motivation, the *Big Five* model provides a theoretical basis for the computational treatment of personality on at least two fronts: the automatic recognition of personality traits from text (which is a language understanding task), and the generation of text in order to reproduce certain personality traits of interest (which is a natural language generation (NLG) task). Knowing the personality traits of an individual (e.g., from his/her social network status updates) has many obvious applications, including staff recruitment, credit analysis etc. In addition to that, personality information may also guide the automatic generation of personalised content, the modelling of psychologically plausible virtual agents (e.g., intelligent tutors, game characters, etc.) and human-computer dialogue applications in which a high degree of realism and engagement is required.

Personality-oriented language understanding and generation are considerable research challenges and, despite their complementary nature (for example, in applications of human-computer dialogue), will usually have a common starting point: a basic resource from which we may establish mappings from linguistic features to personality traits. Based on this observation, this article presents the *b5* corpus of texts produced in multiple communicative tasks and accompanied by inventories of personality of their respective authors. The corpus is, to the best of our knowledge, the largest resource of this kind available for the Brazilian Portuguese language, and it intends to provide support for a wide range of NLP studies based on personality traits.

The rest of this paper is structured as follows. After a brief background discussion (Section 2), the work focuses on the corpus collection task (Section 3) and on its various components (Section 4). Preliminary results of personality recognition from the corpus text are presented for illustration purposes (Section 5). This is followed by a discussion on possible applications and extensions of the present work (Section 6).

2. Background

The *Big Five* model (Goldberg, 1990) comprises five fundamental dimensions of the human personality - Openness to experience, Conscientiousness, Extraversion, Agreeableness and Neuroticism - that may be estimated by using a wide range of methods, the most common being the use of personality inventories. Among many inventories developed for the *Big Five* model, the need for a fast assessment tool led to the proposal of the *BFI* inventory (John et al., 1991).

The BFI inventory has been replicated in dozens of other languages, including some studies dedicated to our target language, Brazilian Portuguese. In particular, the study in (de Andrade, 2008) validated the *BFI* for Brazilian Portuguese through factorial analysis of a sample of 5,089 respondents from all regions of the country. The inventory considered in (de Andrade, 2008) will be the basis of the present work as well.

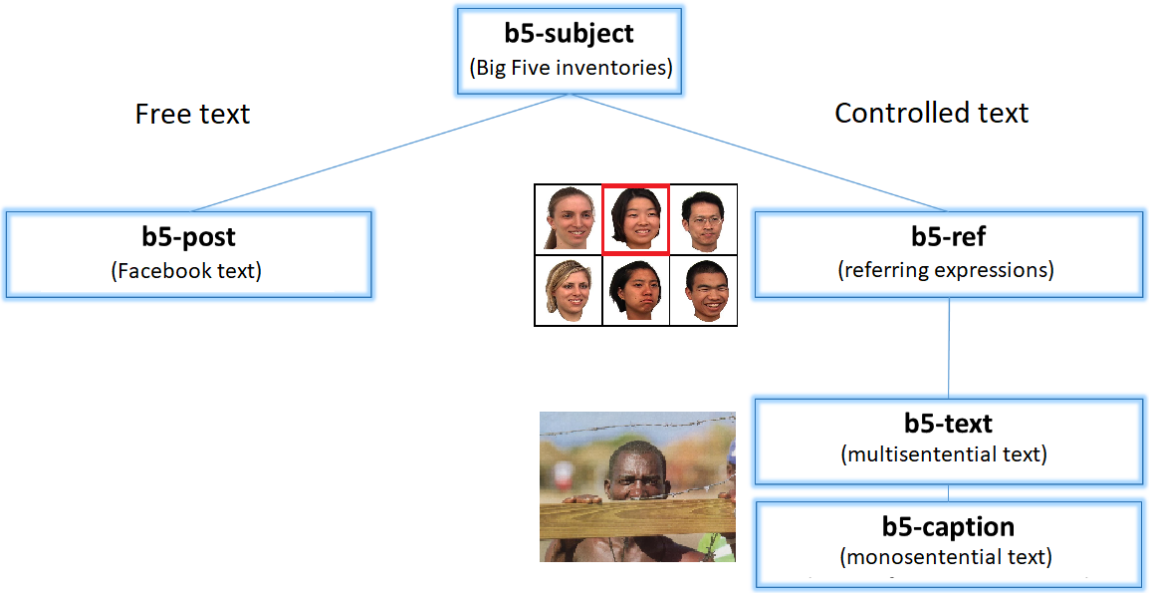


Figure 1: The b5 corpus structure.

The information provided by the *BFI* enables the investigation of a range of issues related to the computational treatment of human personality. A detailed discussion of these applications would be beyond the scope of this paper, but includes the recognition of personality traits from text on social networks (Iacobelli et al., 2011; Celli, 2012; Álvarez-Carmona et al., 2015) and the generation of text based on a target personality (Mairesse and Walker, 2011).

Applications of this kind will usually rely on text corpora annotated with personality information. An example of resource of this kind is *myPersonality*, a large database of Facebook status updates for the English language and corresponding Big Five information about their authors. We are not aware, however, of any similar resources for our target language (Brazilian Portuguese).

In addition to the lack of language resources in this target language, we notice that existing resources are usually devoted to personality recognition applications, but they may be less suitable for personality-dependent language generation (e.g., (Mairesse and Walker, 2011)), in which case it may be necessary to have access not only to the text produced by different individuals (i.e., with different personality traits) but also to the context within which the text was produced. These observations lead us to collect a novel resource for personality-based Portuguese language generation and understanding, hereby called the *b5* corpus.

3. Corpus structure

The *b5* corpus is a dataset containing texts and self-reported personality inventories of their authors. This consists of an author's knowledge base called *b5-subject* and four text databases (or subcorpora) called *b5-post*, *b5-ref*, *b5-text*, and *b5-caption* discussed below. An overview of this organisation is illustrated in Figure 1.

The personality inventory that accompanies the collected texts is a Brazilian Portuguese version of the 44-item *BFI* developed for the English language (John et al., 1991), and presented in (de Andrade, 2008). The set of inventories and additional participant's demographics are represented as the corpus *b5-subject* knowledge base.

From the set of inventories, we computed the five basic factors and, as proposed in (Soto and John, 2009), two additional facets each: (Extraversion) Assertiveness and Activity, (Agreeableness) Altruism and Compliance, (Conscientiousness) Order and Self-discipline, (Neuroticism) Anxiety and Depression, and (Openness to experience) Aesthetics and Ideas.

As shown in Figure 1, the *b5* corpus conveys four text categories divided into two general classes: *free* text obtained from Facebook status updates of each participant, and three types of *controlled* text obtained from a series of in-person data collection tasks. The collection of both free and controlled text is motivated by the dual purpose of the corpus, that is, by our long-term goal of reusing the data both in language understanding and language generation studies. The use of free text is mainly motivated by the specific needs of certain types of application, such as the recognition of personality from text on social networks, whereas the use of controlled text is required for a range of Natural Language Generation (NLG) studies.

The free text dataset constitutes the *b5-post* subcorpus. Controlled texts constitute the *b5-ref* subcorpus of referring expressions, and the *b5-text* and *b5-caption* subcorpora of multi- and mono-sentential descriptions. Since not all participants of the data collection completed every task, each subcorpus may include text produced by a different subset of individuals.

Personality inventories, free and controlled text were collected through a Facebook application and/or in-person experiments. The Facebook application allowed users to respond the personality inventory and, simultaneously, performed the collection of their status updates (upon consent). For a subset of subjects, an offline version of the inventory was made available and, instead of collecting Face-

book text, this was followed by an in-person experiment to elicit controlled text.

As in (Schwartz et al., 2013) and others, Facebook text comprises our major source of knowledge for investigating the relationships between personality and language use. However, since the *b5* corpus intends to provide support to text generation studies as well, the corpus also includes text produced under controlled conditions, in which case not only the text produced by the human subjects is available, but also the original context from which the text was elicited in the first place.

Following much of the work on NLG, controlled text was elicited from visual stimuli represented by images widely used in Psycholinguistics. In the present work, images were taken from the GAPED (Dan-Glauser and Scherer, 2011), Face Place (Righi et al., 2012) and Greebles (Gauthier and Tarr, 1997) image databases[1]. Based on the selected stimuli, participants were requested to produce text in three natural language production tasks: reference production, multi- and mono-sentential scene description. These tasks are described in more detail in the next section.

Both free and controlled texts were subject to spell-checking and basic pre-processing procedures[2]. These included the removal of hashtag and special characters, and the treatment of compound terms (e.g., 'a-m-a-z-i-n-g' or 'HappyDaysAhead'), among others.

For reasons of anonymity, and also to provide a minimal level of normalisation, the text was also subject to a number of replacement operations. In particular, proper names were replaced by a $NAME$ identifier, numeric expressions were replaced by $NUMBER$ and laugh and emotion expressions (e.g., 'yay', 'ouch', 'haha', 'LoL' etc.) were replaced by $LAUGH$, (negative) $EMOTION-$, (positive) $EMOTION+$ or (ambivalent) $EMOTION*$. Some of these operations are however only relevant in the case of free *b5-post* text since expressions of this kind did not generally occur in the controlled texts.

Table 1 presents the number of subjects, sentences (or status updates, in the case of the *post* subcorpus), items (words, punctuation symbols, etc.), and word types in the corpus.

Subcorpus	Subjects	Sentences	Items	Types
post	1,019	194,382	2,219,585	866,243
text	151	1,510	84,463	37,210
caption	151	1,510	4,896	4,121
ref	152	4,558	64,518	18,700

Table 1: Textual data in the *b5* corpus.

4. The b5 components

The corpus consists of four text subcorpora - *b5-post*, *b5-ref*, *b5-text* and *b5-caption* - labelled with their corresponding author's identifiers. Using these identifiers, it is possible to retrieve author's personality scores and additional demographics from the *b5-subject* knowledge base.

b5-subject contains 1082 personality inventories and partial author information regarding gender, age, background, degree of religiosity (on a 1-5 scale) and undergraduate course information. Gender information is known for 1081 (99.9 %) subjects, being 597 (55.2 %) female. Age is known for 810 (74.9 %) subjects, ranging from 18 to 61 years (average of 24.6 years).

Figure 2 illustrates personality distribution across corpus participants.

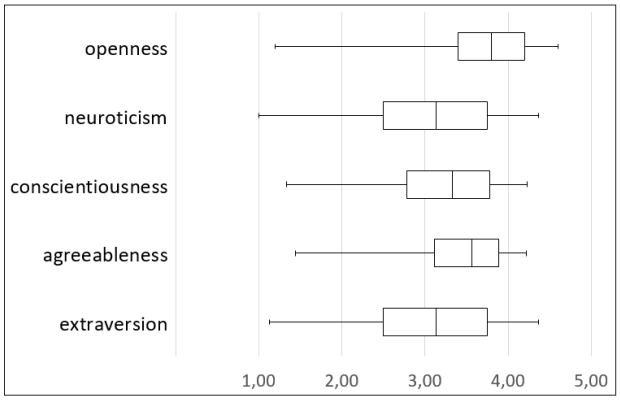


Figure 2: Personality distribution.

Details of the age distribution are presented in Figure 3.

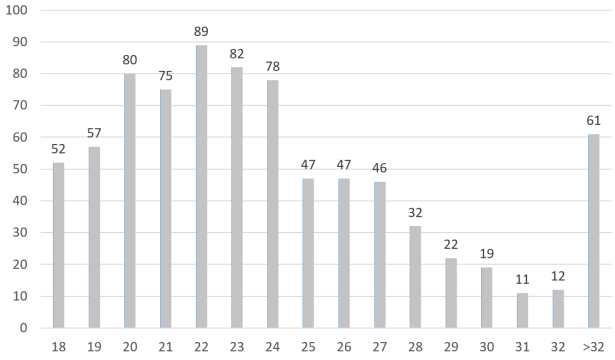


Figure 3: Age distribution.

The individual text components of the *b5* corpus are described in the following sections.

4.1. Facebook status updates *b5-post*

The *b5-post* subcorpus was built for the study and development of computational models of personality recognition and author profiling (e.g., gender or age recognition etc.) from social networks text. The corpus contains Facebook status updates from participants who filled out the personality inventory using the purpose-built application. For each subject, up to 1,000 Facebook status updates were collected. Users with little or no Facebook activity were discarded, resulting in a corpus of 1019 texts.

[1] Face Place and Greeble images are courtesy of Michael J. Tarr, Center for the Neural Basis of Cognition and Department of Psychology, Carnegie Mellon University.

[2] Although certain spelling mistakes might be indicative of personality traits, this possible source of knowledge was discarded as a means to enable the use of the corpus in studies of NLG as well, whose focus is usually the generation of correct text.

4.2. Referring Expressions *b5-ref*

The *b5-ref* subcorpus was built for the study of the effects of human personality on the generation of referring expressions (REG), which is an active research topic in NLG (Krahmer and van Deemter, 2012). REG is hereby understood both as the task of determining the semantic contents of definite descriptions (or *what to say* about the intended referent), and as the surface realisation task of these expressions (or *how to say it* in a target language).

As in much of the existing work on data collection for REG (Gatt et al., 2007; Dale and Viethen, 2009; Paraboni et al., 2017a), the *b5-ref* corpus was implemented as a language production task in which subjects were requested to distinguish a certain target from distractor objects in a given context by making use of a definite description. Unlike REG corpora based on simplified domains (e.g., geometric objects), however, *b5-ref* makes use of stimulus images that may arguably make differences across personality traits more explicit. More specifically, the referential contexts under consideration convey images extracted from *Face Place* (Righi et al., 2012), a collection of realistic human photographs annotated with affective and physical attributes.

An example of stimulus image of this kind is illustrated in Figure 4.

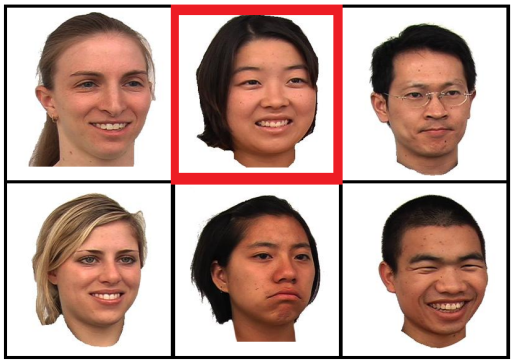


Figure 4: Stimulus image built from *Face Place*.

Given a series of contexts of this kind, subjects were instructed to complete a sentence in the form 'The person / entity highlighted in red is the ... ', which elicited a response in the form of a single referring expression. In the present example, this could be done, for instance, by making use of expressions such as 'the smiling Asian girl' or 'the only girl with dark hair who is smiling', among many other possibilities.

Subjects were instructed to imagine that they were describing each face to a person who could not see their own screen, and that for that reason they should avoid making reference to screen positions (e.g., 'the Asian girl *on the top row*). As a means to reduce the monotonicity of the task, the main stimuli were interleaved with filler images depicting Greebles (Gauthier and Tarr, 1997), since these objects are particularly difficult to identify based on their physical features alone. An example of stimulus image built from Greebles is illustrated in Figure 5.

The *b5-ref* descriptions were produced by 152 subjects, being 86 (56.6 %) female and on average 25.8 years-old (minimum 18 and maximum 59). In its current version, the *b5-ref* corpus contains 1810 Face Place definite descriptions. Given the purpose of the data - for studies on personality-based REG - the *b5-ref* subcorpus differs from the other (purely textual) portions of the *b5* corpus in that these expressions include semantic annotation represented as properties (or attribute-value pairs) as in *gender-female*. To this end, the elicited descriptions were labelled according to a 27-attribute annotation scheme based on the most frequent types of information observed in the corpus.

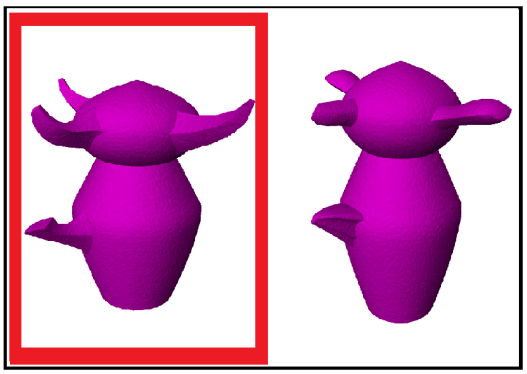


Figure 5: Filler image built from *Greebles*.

Attribute values were partly obtained from *Face Place*, and partially obtained from manual annotation, including both physical (e.g., skin colour, hair length etc.) and affective (e.g., negative and positive emotions) properties. Thus, *b5-ref* is a semantically-annotated REG corpus in the traditional sense, i.e., not unlike TUNA (Gatt et al., 2007) and others, only with additional information about the personality of every speaker.

The annotation scheme disregarded attributes that were not added for the explicit purpose of identification (as in 'the person *who seems to have good taste in clothes*'), and this information was therefore not annotated (although it still remains available from the original text in the corpus). Moreover, as a means to avoid annotating an overly large number of sparse attributes (which may be of little interest from a REG perspective), certain attributes were combined into more general classes according to their semantic affinity. For instance, all references to facial hair (e.g., beard, moustache, goatee etc.) were represented as a single attribute *facial.hair* with possible values *yes / no* indicating simply that there was a general reference to this class of related concepts.

In addition to that, attributes whose value could not be objectively determined (e.g., whether a certain face shape may be considered 'round' or not) were modelled as having only the value *others*. This is intended to represent attributes that have no discriminatory value (e.g., because any of the faces presented as stimulus may be considered, to some extent, as having a round shape), and it carries non-trivial consequences for the design of REG algorithms that favour the selection of discriminatory information and/or pay regard to referential overspecification (Paraboni et al., 2017b).

The relative subjectivity of certain attributes was treated as evenly as possible by considering the information provided by Face Place, if available. Thus, for instance, properties

related to ethnic type (black, Asian etc.) or those that represent emotions (happy, sad etc.) were annotated with their default Face Place values or, if unavailable, according to the judgement of the majority of the 152 participants of the data collection task. Moreover, all descriptions of a given image showing, e.g., a short-haired person (according to the information provided by *Face Place*) were annotated as *hair.length-short* even if a particular individual described it as being long. In other words, the annotated value is meant to model the reference to the *hair.length* attribute, but not necessarily of the actual (*short* or *long*) value chosen by each individual.

Infrequent information was generally omitted from the annotation scheme as well, and it was therefore not recorded. This included references to 'only' (e.g., 'the only smiling Asian girl'), degree modifiers (e.g., 'very', 'slightly' etc.), comparatives (e.g., 'larger than') and references to a second person in the scene (e.g., 'next to a blond girl').

The *b5-ref* subcorpus is provided as two main components: a set of XML files representing the expressions produced by every participant, and the full semantic specification of each of the 12 stimulus scenes. This representation is similar to (Gatt et al., 2007) and many other REG projects. An example is illustrated in Figure 6.

```
<TRIAL ID="7" SPEAKER="31">

 <CONTEXT ID="5">
  <ATTRIBUTE-SET STRING="the smiling asian woman ">
   <ATTRIBUTE NAME="smile" VALUE="yes" />
   <ATTRIBUTE NAME="race" VALUE="asian" />
   <ATTRIBUTE NAME="gender" VALUE="female" />
  </ATTRIBUTE-SET>
 </CONTEXT>

...
<\TRIAL>
```

Figure 6: An annotated referring expression in *b5-ref*.

In the *b5-ref* data, personality traits have been shown to affect both the contents and the surface form of referring expressions. Preliminary results of a machine learning REG model based on *b5-ref* data in (Paraboni et al., 2017c) suggest that the selection of non-discriminatory attributes (e.g., the property of 'being young', which is shared by all objects in the *b5-ref* domain and it is therefore not discriminatory) is particularly influenced by the speaker's personality traits, an effect that is less evident in the case of discriminatory (or more perceptual) attributes. Moreover, results from a personality-dependent lexical choice model built from *b5-ref* in (Lan and Paraboni, 2018) showed that the lexicalisation of the most frequent properties (i.e., those for which there is sufficient data in the corpus) greatly improves when personality information is taken into account.

4.3. Scene descriptions *b5-text* and *b5-caption*

Unlike *b5-ref*, the *b5-text* and *b5-caption* subcorpora are primarily intended for the study of more general issues of personality-based text production, such as document planning, text-to-text generation, and summarisation. To this end, the data collection experiment included two scene description subtasks: a detailed version in the form of multi-sentential text, and a short version in the form of a single sentences similar to picture captions.

The visual stimuli employed in both cases were taken from *GAPED* (Dan-Glauser and Scherer, 2011), a database conveying images classified by valence and normative significance, and designed to arouse different degrees and types of reaction. The image description task made use of 10 GAPED images with valence values selected at regular intervals (from 3 to 54 degrees). An example is illustrated in Figure 7.

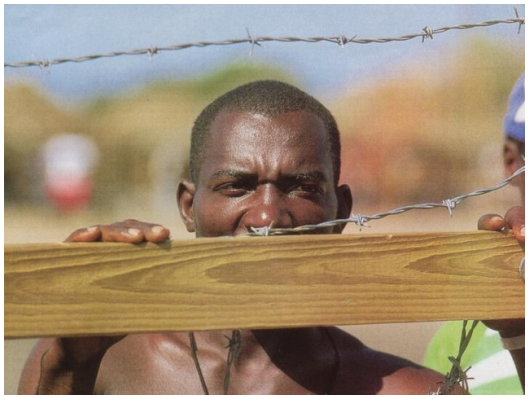


Figure 7: Stimulus image from *GAPED*.

Differently from the identification task in *b5-ref*, the goal in this case was to investigate which pictorial elements each speaker would select to describe each image, the order and structuring of these descriptions, and the lexical and syntactic choices made. To this end, the data collection was carried out in two versions - detailed and summarised text - as a means to obtain a greater degree of control over the elicited text, and to observe both discourse-level and sentence-level linguistic phenomena.

In the first task, participants were requested to fill in a text box to describe everything they could see in the scene as if they were aiding a (hypothetical) visually-impaired friend. After that, they were requested to summarise the scene contents as a single sentence (or caption) for the same purpose. For instance, a possible text description of the scene in Figure 7 may include the following example:

> '*There is a black man leaning against a barbed wire fence. He is shirtless, and he seems tired, or perhaps even sad. There seems to be a second person on his left, but he is mostly out of the picture. He is black as well, I guess, and he is wearing a blue cap.*'

A single-sentence caption for the same picture may be represented as the following example:

> '*Man looking through a fence.*'

Both *b5-text* and *b5-caption* are available in three formats: *original*, consisting of 10 files containing the set of all descriptions of each of the 10 stimulus scenes and their corresponding subject's identifiers; *per-speaker*, consisting of

151 files containing the text produced by each individual subject, and *parsed*, representing the same 10 original files with syntactic information. Generally speaking, the *original* and *parsed* formats are potentially more useful to NLG studies since they indicate what each subject wrote in response to each of the visual stimuli, whereas *per-speaker* is more useful to language understanding studies (for example, for author profiling or document classification).
As in the case of the *b5-ref* subcorpus, personality traits have also been shown to affect both the contents and the surface form of the elicited text and captions. Results of this analysis will be described elsewhere.

5. Using the b5 data

As a means to illustrate the use of the *b5* corpus data, and also to present initial reference results for future studies of this kind, this section describes a simple experiment in personality recognition from *b5-post* data. The experiment is solely focused on the recognition of the five basic personality factors, that is, individual personality facets are presently disregarded. For a more comprehensive discussion on the use of different data sources (e.g., posts, text, caption etc.) in the personality recognition task, we refer to (dos Santos et al., 2017b).

5.1. Models

Personality recognition is presently modelled as a series of five independent binary classification tasks (e.g., extrovert vs. introvert etc.) associated with each of the *Big Five* dimensions. For each of these classes, a positive label was assigned to individuals whose personality score was above the average of the group as seen in *b5-subject*, and a negative label was assigned to those whose personality score was equal or below this average.
Table 2 summarises the number of positive and negative instances for each of the five personality traits. As all classes are approximately balanced, no further re-sampling was performed.

Trait	positive	negative
Extraversion	505	514
Agreeableness	537	482
Conscientiousness	507	512
Neuroticism	548	471
Openness	533	486

Table 2: Learning instances

As learning features, we computed 64 LIWC categories (Pennebaker et al., 2001), four additional, MRC-like (Coltheart, 1981) psycholinguistic properties and further 60 dictionary attributes.
LIWC features were obtained from Brazilian Portuguese LIWC (Filho et al., 2013) by counting word categories (e.g., religion, family, money etc.). Each feature represents the number of words found in the corresponding category normalised by the length of each Facebook time line in number of words.
The four additional psycholinguistic features were obtained from (dos Santos et al., 2017a) by computing average concreteness, imageability, subjective frequency and age of acquisition scores. Each feature represents the average score of all words in the corresponding category found in each Facebook time line.
Dictionary features were obtained from Unitex-PB (Muniz, 2004) by computing word classes and a range of morphological features. Once again, each feature represents the number of words found in the corresponding category normalised by document length.
In all models, we make use of SVM classifiers with linear kernel and $\gamma = 0.1$ and $C = 1$ with 10-fold cross validation over the entire dataset.

5.2. Results

Mean precision (P), recall (R) and F1 measure (F1) scores for the five personality classification tasks are summarised in Table 3.

	positive class			negative class		
Class	P	R	F1	P	R	F1
Extraversion	0.59	0.59	0.59	0.60	0.60	0.60
Agreeableness	0.54	0.93	0.68	0.56	0.10	0.17
Conscientiousness	0.56	0.54	0.55	0.56	0.58	0.57
Neuroticism	0.55	0.95	0.69	0.59	0.09	0.15
Openness	0.57	0.67	0.61	0.55	0.44	0.49

Table 3: Personality recognition in *b5-post*

5.3. Discussion

As in existing studies of personality recognition for the English language (Mairesse et al., 2007), we notice that the Extraversion class presents the best overall results. This may suggest that this particular dimension of human personality is more evident in (Facebook) text than others.
A machine learning approach as in this example may of course be applied to many other forms of author profiling based on the *b5* corpus. These include, for instance, the classification of gender, age group and others. However, since these tasks require the additional definition of how they would be modelled in the form of a classification problem (e.g., binary, multi-class, etc.), this kind of investigation would be outside the scope of the current discussion. A number of author profiling tasks of this kind, based on *b5* data, are discussed in (Hsieh et al., 2018).

6. Final remarks

This article has described the construction of the *b5* corpus, a collection of texts produced in different communicative tasks, and accompanied by the inventories of personality of their respective authors. The *b5* corpus represents, to the best of our knowledge, the largest resource of the kind available for the Brazilian Portuguese language, and it is potentially useful for studies of computational recognition of personality traits from texts, author profiling, natural language generation based on personality traits and others. Some of these alternatives are summarised as follows.
The *b5-subject* knowledge base contains Big Five personality information, and additional attributes regarding subject's gender, age, background and others. As a result,

it may provide knowledge not only for the computational study of human personality proper as in (Mairesse et al., 2007; Farnadi et al., 2013; Nowson and Gill, 2014), but also for the study of other forms of author profiling, as in (Schwartz et al., 2013; Marquardt et al., 2014; Álvarez-Carmona et al., 2015; González-Gallardo et al., 2015; Șulea and Dichiu, 2015; Najib et al., 2015).

The *b5-post* subcorpus - containing Facebook status updates - is a textual base developed primarily for the purpose of personality recognition and author profiling in Portuguese. Studies of this kind would typically take the form of a supervised (Mairesse et al., 2007), or semi-supervised (Celli, 2012) learning task. The experiment described in the previous section is an (admittedly simple) example of the former.

The *b5-ref* subcorpus intends to support studies on machine-learning referring expression generation (REG) that take personality information into account. Studies of this kind may be seen as a possible generalisation of models of human variation for this task (Viethen and Dale, 2010; Ferreira and Paraboni, 2017). Moreover, we notice that the corpus may be also useful for studies of personality-based surface realisation and lexical choice of definite descriptions.

The *b5-text* subcorpus is potentially useful as a means to establish mapping between linguistic features and personality traits, which may guide the design of text generation models based on personality with a particular focus on multi-sentential phenomena. Given the relatively controlled domain - based on the same set of images described by all participants - *b5-text* texts make evident the different linguistic choices made by each individual. These choices, which may or may not be due to differences in personality, are observable both in surface forms and contents.

Finally, the *b5-caption* subcorpus complements the previous *b5-text* data by providing a shortened version of the same image descriptions. This corpus may be particularly useful for the study of more superficial linguistic features - such as syntactic structures and lexical choice - and their relation to personality. In addition to that, since captions may be seen as a short, single sentence summary of the larger *b5-text* text, *b5-caption* may be useful also for the development of text summarisation approaches that take the personality of the human summariser into account.

As future work, we intend to provide the semantic annotation of the stimulus scenes in *b5-text* and *b5-caption*, so that these datasets may be explored more fully in subsequent NLG studies. This work is currently in progress.

The complete *b5* corpus , currently in its version 1.7., is available under a Creative Commons Attribution 4.0 International License. The corpus may be freely downloaded[3] and reused for research purposes.

7. Acknowledgements

This work has been supported by grant # 2016/14223-0, São Paulo Research Foundation (FAPESP).

[3] https://drive.google.com/open?id=0B-KyU7T8S8bLTHpaMnh2U2NWZzQ

8. Bibliographical References

Álvarez-Carmona, M., López-Monroy, A., y Gómez, M. M., Villaseñor-Pineda, L., and Escalante, H. (2015). INAOE's participation at PAN'15: Author Profiling task. In *CLEF 2015*.

Celli, F. (2012). *Adaptive Personality Recognition from Text*. Ph.D. thesis, University of Trento.

Coltheart, M. (1981). The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology*, 33(4):497–505.

Dale, R. and Viethen, J. (2009). Referring expression generation through attribute-based heuristics. In *ENLG'09*, pages 58–65, Athens.

Dan-Glauser, E. S. and Scherer, K. R. (2011). The Geneva affective picture database (GAPED): a new 730-picture database focusing on valence and normative significance. *Behavior Research Methods*, 43(2):468–477.

de Andrade, J. M. (2008). *Evidências de validade do inventário dos cinco grandes fatores de personalidade para o Brasil*. Ph.D. thesis, University of Brasília.

dos Santos, L. B., Duran, M. S., Hartmann, N. S., Junior, A. C., Paetzold, G. H., and Aluísio, S. M. (2017a). A lightweight regression method to infer psycholinguistic properties for Brazilian Portuguese. In *TSD-2017*.

dos Santos, V. G., Paraboni, I., and Silva, B. B. C. (2017b). Big five personality recognition from multiple text genres. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 29–37, Prague, Czech Republic. Springer-Verlag.

Farnadi, G., Zoghbi, S., Moens, M.-F., and de Cock, M. (2013). Recognising personality traits using Facebook status updates. In *Proceedings of WCPR13 in conjunction with ICWSM-13*, Boston, USA. The AAAI Press.

Ferreira, T. C. and Paraboni, I. (2017). Generating natural language descriptions using speaker-dependent information. *Natural Language Engineering*, 23(6):813–834.

Filho, P. P. B., Aluísio, S. M., and Pardo, T. (2013). An evaluation of the Brazilian Portuguese LIWC dictionary for sentiment analysis. In *proc. of STIL-2013*, pages 215–219, Fortaleza, Brazil.

Gatt, A., van der Sluis, I., and van Deemter, K. (2007). Evaluating algorithms for the generation of referring expressions using a balanced corpus. In *Proc. of ENLG-07*.

Gauthier, I. and Tarr, M. J. (1997). Becoming a greeble expert: Exploring mechanisms for face recognition. *Vision Research*, 37(12):1673–1682.

Goldberg, L. R. (1990). An alternative description of personality: The Big-Five factor structure. *Journal of Personality and Social Psychology*, 59:1216–1229.

González-Gallardo, C., Montes, A., Sierra, G., Núñez-Juárez, J., Salinas-López, A., and Ek, J. (2015). Tweets Classification Using Corpus Dependent Tags, Character and POS N-grams. In *CLEF 2015*.

Hsieh, F. C., Dias, R. F. S., and Paraboni, I. (2018). Author profiling from facebook corpora. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

Iacobelli, F., Gill, A. J., Nowson, S., and Oberlander, J.

(2011). Large scale personality classification of bloggers. In *ACII (2)*, pages 568–577. Springer.

John, O. P., Donahue, E., and Kentle, R. (1991). The Big Five inventory - versions 4a and 54. Technical report, Inst. Personality Social Research, Univ. California.

Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Lan, A. G. J. and Paraboni, I. (2018). Definite description lexical choice: taking speaker's personality into account. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

Mairesse, F. and Walker, M. A. (2011). Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computational Linguistics*, 37(3):455–488.

Mairesse, F., Walker, M., Mehl, M., and Moore, R. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research (JAIR)*, 30:457–500.

Marquardt, J., Farnadi, G., Vasudevan, G., Moens, M.-F., Davalos, S., Teredesai, A., and de Cock, M. (2014). Age and gender identification in social media. In *CLEF 2014 Working notes*, pages 1129–1136, Sheffield, UK.

Muniz, M. C. M. (2004). A construção de recursos linguístico-computacionais para o Português do brasil: o projeto de Unitex-PB. Master's thesis, USP São Carlos.

Najib, F., Cheema, W. A., and AdeelNawab, R. M. (2015). Author's traits prediction on Twitter data using content based approach. In *CLEF 2015 Working Notes*.

Nowson, S. and Gill, A. J. (2014). Look! who's talking?: Projection of extraversion across different social contexts. In *Proceedings of the 2014 ACM Multi Media on Workshop on Computational Personality Recognition*, pages 23–26, Orlando, FL, USA. ACM.

Paraboni, I., Galindo, M., and Iacovelli, D. (2017a). Stars2: a corpus of object descriptions in a visual domain. *Language Resources and Evaluation*, 51(2):439–462.

Paraboni, I., Lan, A. G. J., de Sant'Ana, M. M., and Coutinho, F. L. (2017b). Effects of cognitive effort on the resolution of overspecified descriptions. *Computational Linguistics*, 43(2):451–459.

Paraboni, I., Monteiro, D. S., and Lan, A. G. J. (2017c). Personality-dependent referring expression generation. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 20–28, Prague, Czech Republic. Springer-Verlag.

Pennebaker, J. W., Francis, M. E., and Booth, R. J. (2001). *Inquiry and Word Count: LIWC*. Lawrence Erlbaum, Mahwah, NJ.

Righi, G., Peissig, J. J., and Tarr, M. J. (2012). Recognizing disguised faces. *Visual Cognition*, 20(2):143–169.

Schwartz, H. A., Eichstaedt, J. C., Kern, M. L., Dziurzynski, L., Ramones, S. M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M., and Ungar, L. H. (2013). Personality, gender, and age in the language of social media: the open-vocabulary approach. *PLoS One*, 8(9):e73791.

Soto, C. J. and John, O. P. (2009). Ten facet scales for the Big Five Inventory: Convergence with NEO PI-R facets, self-peer agreement, and discriminant validity. *Journal of Research in Personality*, 43(1):84–90.

Șulea, O.-M. and Dichiu, D. (2015). Automatic Profiling of Twitter Users Based on Their Tweets. In *CLEF 2015*.

Viethen, J. and Dale, R. (2010). Speaker-dependent variation in content selection for referring expression generation. In *Proc. of the Australasian Language Technology Association Workshop 2010*, pages 81–89, Melbourne.

Linguistic and Sociolinguistic Annotation of 17th Century Dutch Letters

Marijn Schraagen, Feike Dietz, Marjo van Koppen
Utrecht University, The Netherlands
{M.P.Schraagen, F.M.Dietz, J.M.vanKoppen}@uu.nl

Abstract

Developments in the Dutch language during the 17th century, part of the Early Modern period, form an active research topic in historical linguistics and literature. To enable automatic quantitative analysis, a corpus of letters by the 17th century Dutch author and politician P.C. Hooft is manually annotated with parts-of-speech, document segmentation and sociolinguistic metadata. The corpus is developed as part of the Nederlab online research portal, which is available through the CLARIN ERIC European research infrastructure. This paper discusses the design and evaluation of the annotation effort, as well as adding new annotations to an existing annotated corpus.

Keywords: Early Modern Dutch, POS tagging, sociolinguistic annotation, data integration

1. Introduction

In the late 16th and 17th century, the Dutch language was subject to a series of standardization and modernization developments, described in historical linguistics as the transition into (Early) Modern Dutch (Donaldson, 1983; Burke, 2005). These developments include changes in spelling and vocabulary, as well as syntactic changes regarding case marking and negation. The adoption of developments differs not only between authors, but also within the same author (van der Wal et al., 2012; Nobels and Rutten, 2014), cf. intra-speaker variation in speech corpora (Schilling-Estes, 2002; Szmrecsanyi, 2005) or intra-author variation in English literature (Leech, 1969; Busse, 2002). Variation is influenced by linguistic contexts as well as sociolinguistic factors (e.g., intended audience). To aid the study of language variation in this time period, a corpus is developed containing annotations on both the linguistic and sociolinguistic level. This corpus is intended to be used directly as linguistic research data, to study phenomena of interest both within the corpus itself as well as in comparison to external resources. Moreover, the manual annotation effort can be leveraged as training data for automatic methods in the field of historical linguistics.

2. Corpus Development

The corpus includes documents from the correspondence archive of the author and politician P.C. Hooft (1581-1647), which is available online from the Digital Library of Dutch Literature (Committee DBNL, 2015). Every document in this corpus corresponds to a single letter. Hooft is known to have been interested in studying the development of the Dutch language, and advancing this development in his own writing. The full correspondence consists of over 1300 letters with a total length of 300k tokens, from which 333 letters from the period 1600-1638 have been annotated (108k tokens in total). The boundary of 1638 is selected based on a shift in the use of negation in the work of Hooft, however the letters from this time period contain a number of other interesting linguistic phenomena as well.

The annotation task was performed by a pool of eight students with a background in linguistics and/or historical literature. Annotations consist of lemmas, part-of-speech tags (including various features for each tag) and sociolinguistic variables on document and person level. POS tagging makes use of the tagset described by Van Eynde et al. (2000) for contemporary Dutch, which is developed for the Spoken Dutch Corpus (CGN Consortium, 2003). For the current corpus several features have been added to accomodate historical linguistic phenomena, such as case marking and negation clitics. To increase annotation efficiency, the methodology was based on post-correction of tags generated by the Adelheid tagger for Middle Dutch (van Halteren and Rem, 2013). Middle Dutch shares a number of interesting phenomena with Early Modern Dutch (e.g., case marking, clitics, pronoun compounding) which are not found in Modern Dutch, therefore this tagger provided a useful starting point for manual annotation. However, major differences between the two historical language varieties exist as well, which necessitates a full check on all generated tags. Moreover, the manual tagging effort was used to extend the original tagset with additional features.

Document characteristics	
Category	business, personal
Type	regular, appendix
Goal	express thanks, compliment, excuse, ask a favour, ask information, ask advice, admonish, inform, remember, persuade, order, allow, invite
Topic	business, literature, domestic affairs, love, death, news, religion/ethics
Correspondent characteristics	
Group	name
Individual	name, birth/death date, gender, occupation, literary author, relation to P.C. Hooft
Letter segmentation	
Introductory greeting, opening (optional), narrative, closing (optional), final greeting	

Table 1: Sociolinguistic annotation set.

The sociolinguistic annotation set (provided in Table 1) consists of document characteristics, correspondent characteristics and text structure segmentation. The set of letter segments corresponds to the use in previous research, e.g.,

token	*translation*	*annotation 1*	*annotation 2*	*description*
fraejicheden	fine things	fraaiheid	fraaiigheid	lemma difference
gedicht	written poetry	N(nonnom,sg)	V(lex,pp)	past participle vs. noun
ijet	something	PRON(indef,3,sg,nonnom)	PRON(indef,nonnom)	missing features
kan	can	V(simple,pres,nonlex,sg,3)	V(simple,pres,nonlex,sg,1)	1/5 features different
etc	etc	SPEC(unclear)	SPEC(foreign)	ambiguous feature

Table 2: Tagging disagreement examples.

(Rutten and van der Wal, 2014, p. 86), whereas document categories are based on general rhetorical frameworks of letter writing (cf., for example, (Stowers, 1986, pp. 15-16)). The categories for correspondent characteristics are developed specifically for this corpus.

3. Inter-Annotator Agreement

For each annotator, a number of documents with a total of approximately 1000 words has been assigned to a second annotator as well. Using this data, inter-annotator agreement (IAA) can be computed on various aspects of annotation, i.e., token-based annotation of lemma, POS and features, document-based annotation of text segments and letter characteristics. For POS agreement a fine-grained measure is desired to differentiate various types of disagreement. For example, a token can be assigned different main tags or different features within the same main tag, feature categories can contain missing values instead of different values, various main tags have a different number of feature categories which influences the potential for disagreement, and some feature categories can be considered more salient than others, or more ambiguous. Some examples are provided in Table 2.

In related work, for various resources only a single IAA figure is reported (Voutilainen, 1999; Brants, 2000; Gimpel et al., 2011; Plank et al., 2014), or IAA is mentioned as desirable but not computed due to resource development constraints (Oostdijk et al., 2008; van Halteren and Rem, 2013). Widely used tagsets, such as the Penn Treebank tagset for English (Santorini, 1990) or the *STTS-small* tagset for German (Schiller et al., 1999), do not make an explicit distinction between main tags and features, and the amount of (implicit) features is usually small (e.g., the Penn tagset contains 4 noun features and 6 verb features, vs. 11 and 18 in the current tagset, respectively), which could explain the absence of feature agreement measurements. Indeed, for the Spoken Dutch Corpus (on which the tagset for the current corpus is based), two separate figures are reported for agreement on tags with features and agreement on main tags only (Zavrel and Daelemans, 1999). To extend this approach, in Figure 1 agreement measures are provided[1] for lemmas, full POS tags, main tags only, agreement on single features instead of feature sets, and agreement on feature categories where both annotators have provided a value (i.e., features in a category, such as *verb tense*, are discarded for a token if a value is missing for one of the annotators). In Table 3 the application of the measures to the examples in Table 2 is provided.

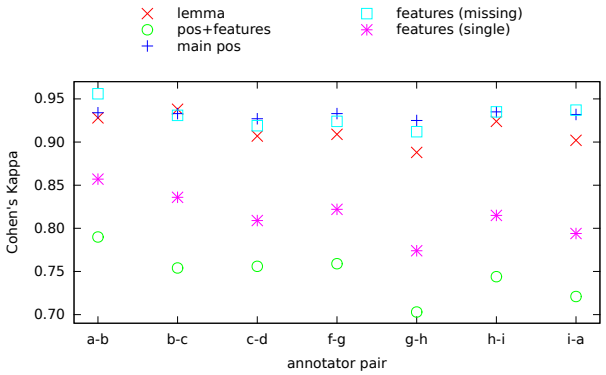


Figure 1: POS inter-annotator agreement.

token	pos + features	main pos	features (missing)	features (single)
gedicht	0	0	0	0
ijet	0	1	1	0.5
kan	0	1	0.8	0.8
etc	0	1	0	0

Table 3: Application of agreement measures to examples from Table 2.

The figure shows that agreement on full tags (i.e., the measure generally reported for this type of annotation task) is relatively low (~ 0.75). When features are not taken into account (i.e., agreement on lemma or on main POS), agreement is higher (> 0.9). Similarly, agreement is high when features from a particular feature category are counted only if both annotators provided a value for this feature category (denoted by *features (missing)* in Figure 1). The first condition (full agreement) may be considered too strict, while lemma, main POS, and missing features[2] are arguably too permissive. A more balanced measure therefore may be the agreement on individual features (~ 0.81).

For document-based annotation the amount of data is much smaller compared to POS tagging (24 letters in total), therefore counts are computed instead of statistical measures, as

[1] Originally, the tagging task was assigned to a pool of nine annotators, each of which was assigned an individual set of documents as well as a selection of documents from the set of the previous annotator for the measurement of inter-annotator agreement. However, one of the annotators (annotator *e*) left the pool, which means that the agreement for the pairs *d–e* and *e–f* could not be measured, resulting in a total of seven pairs in Figure 1.

[2] This condition applies to 12% of all feature agreement measurements, averaged over annotator pairs. Note that the issue could have been avoided by enforcing annotation guidelines, either in the annotation software or through more explicit instruction and monitoring.

annotation feature	*agreement*	*proportion*
business/personal	21	0.875
regular/appendix	16.5	0.686
letter goal	12	0.500
letter topic	18	0.750
greeting	61	0.241 (0.492)
opening	175	0.449 (0.900)
narrative	6166	0.948 (0.907)
closing	370	0.754 (0.750)
final greeting	311	0.920 (0.926)
segment totals	7083	0.888 (0.795)

Table 4: Agreement on sociolinguistic annotation.

Original framework	
Goal	express thanks, compliment, excuse, ask a favour, ask information, ask advice, admonish, inform, remember, persuade, order, allow, invite
Topic	business, literature, domestic affairs, love, death, news, religion/ethics
Revised framework	
Goal	prompt for action, honour, help, inform, keeping contact, ask for reply
Topic	political work, literary work, current events, social circle

Table 5: Proposed categories for sociolinguistic tagging.

shown in Table 4. In the table, for the document-level annotations the numbers represent the amount of documents for which annotators agree (out of 24). The annotation *unknown* counts as 0.5 agreement. For the text segments the table shows the number of common tokens, proportional to total (union) segment length and (in parentheses) averaged proportion per document. Agreement on letter goal is low, which may indicate that specific letter categories need to be merged. Agreement on segmentation is not unreasonable, especially when taken into account that some differences can be easily corrected (e.g., some annotators include the address line in the *greeting* segment, while others only include the actual greeting). The table shows two different segmentation agreement measurements: the number of words in agreement proportional to the total number of words in the segment in all documents (where the segment length for a document is counted as the length of the union of the segments for both annotators), and the average of all proportions in individual documents. The first measure has the advantage that each token is given equal weight, whereas the second measure gives each document equal weight. The second measure may be more informative, given the large differences in segment length across documents. Additionally, the second measure allows missing optional segments (*opening* and *closing*) to be counted as full agreement, whereas for the first measure missing segments are discarded.

The correspondent-based annotation was performed by a small number of annotators only, therefore agreement on these annotations has been omitted.

3.1. Sociolinguistic annotation schema

As noted in Section 2, the sociolinguistic annotation schema is based on general rhetorical frameworks of letter writing. Descriptions of these frameworks were available in the Netherlands in the 17th century. Therefore, it is likely that P.C. Hooft was familiar with the division into rhetorical categories and, furthermore, that his knowledge of rhetorics influenced his own letter writing. Therefore, the established categories were selected as a framework for sociolinguistic tagging of the current corpus. However, given the fact that agreement on letter goal and topic is low, the framework may need to be revised. In order to provide recommendations for improvement of the category division, a qualitative assessment of the categories has been performed on a sample of 100 letters from the corpus.

A first observation is that the number of categories for goal and topic (see Table 1) is large and the boundaries of the categories are not always clear. Therefore a new division is proposed with a smaller number of categories, where several closely related categories have been grouped together. The category names have been adjusted accordingly to clarify the scope of each category (see Table 5).

A second observation is that, even with the simplified category division, several letters can be categorized with two distinct goals. As an example, consider a letter dated October 6th, 1631. This letter is written following a decision of the State Counsil to modify tax law. As administrator of the region, Hooft was responsible for the implementation of this law. In the letter Hooft informs the administrative college about the new law. More importantly, however, in the letter the members of the college are ordered to take an official oath to enforce this law. The main goal of the letter can therefore be classified as *prompt for action*, with the secondary goal *inform*. However, in the sample of the qualitative assessment, only 11 out of 100 letters have been classified as having a secondary goal.

A further improvement in the consistency of sociolinguistic tagging can be obtained by providing more explicit tagging instructions. This is particularly useful for secondary letter goals, which should be annotated only if the secondary goal can be considered an important part of the letter, instead of being embedded as a courtesy phrase or a small digression.

4. Data Integration

To increase availability and accessibility of the annotations, the corpus has been developed as part of the Nederlab online research portal (Brugman et al., 2016). Nederlab offers a query and visualization interface for full text search in the Digital Library of Dutch Literature using Corpus Query Processor syntax (Christ, 1994). Nederlab is included in the CLARIN ERIC European research infrastructure (Hinrichs and Krauwer, 2014), which offers access through a majority of universities and research institutes in Europe.

All data in the Nederlab research infrastructure has been annotated automatically using the Frog NLP suite (van den Bosch et al., 2007) with lemma, POS and named entity information. The current annotations are used as a pilot for the aim of Nederlab to facilitate incorporation of user-generated data. This type of annotation is expected to be

type	*different*	*equal*	*ratio*
lemma			
manual vs. default	27,179	70,560	0.28
manual vs. modernized	18,052	53,648	0.25
main POS	23,365	82,830	0.22
features (all)	140,888	67,273	0.68
features (missing)	46,725	67,273	0.41

Table 6: Differences between automatic and manual annotation.

fragmented and the quality, while possibly rather high, may be unknown. To allow researchers using the Nederlab portal to differentiate between user-generated information and full coverage core annotations with known performance characteristics, the additional annotations are added as explicity marked alternatives.

Merging new annotations with existing data needs to account for possible differences in tokenization, resulting from preprocessing, tagger tokenization algorithms, and manual tokenization decisions made by annotators. To address this issue, the annotation data is aligned on token level, and multiple tokenization layers are represented in the annotation data model.

The merging process allows for measurements on the differences between automatic and manual annotations for this data, which are provided in Table 6. The table shows that the differences are substantial for all categories. Note that the measurements include an automatic modernization layer for lemmas, implemented using look-up on an authoritative source (de Vries and te Winkel, 1998). The quality of these modernized lemmas is high, however the coverage of this source is not fully complete. For measurements on features, note that manual annotations contain additional feature categories and values for which no automatic alternative is available. In the last row these features have been disregarded, providing a measurement of different feature values only.

Regarding POS-tagging the manual annotation has the additional advantage of providing a clean separation of the main text and additional document elements as compared to automatic tagging. For example, the source text contains modern editorial notes, figure captions, and various layout markers such as page or line numbers. In the manual annotation these elements have been identified, while during automatic annotation these elements have been considered as regular text.

4.1. Alignment algorithm

Token alignment is represented as a one-to-one greedy brute force algorithm, which is sufficient for this task given the serial nature of this specific alignment problem. After alignment, gaps are resolved if possible (or confirmed otherwise) in order to merge the annotation layers.

An outline of the algorithm is presented in Algorithm 1. First, all *anchor tokens*, defined as tokens with a frequency of 1 in both documents (case sensitive), are fixed in the alignment (line 4). The anchor tokens divide the document into pairs of small subsequences (5-6), which are aligned individually. An exception to this heuristic applies when

Algorithm 1: Alignment between documents.

Data: two tokenized documents A, B
Result: token-level alignment
1 **foreach** *token in document A* **do**
2 **if** *token has frequency 1* **then**
3 find this token in document B;
4 **if** *token has frequency 1* **then**
5 subsequence S_A = [previous aligned token in A..current token in A];
6 subsequence S_B = [previous aligned token in B..current token in B];
7 **if** *length difference*$(S_A, S_B) \leq L$ **then**
8 `AlignSubsequences`(S_A, S_B);
9 `ResolveGaps`$(S_A \leftrightarrow S_B)$
10 **end**
11 **end**
12 **end**
13 **end**
14 **function** `AlignSubsequences`(X, Y)
15 *currentAlign* $\leftarrow X_i = Y_i$ for all tokens i;
16 $\Delta_{\text{current}} \leftarrow \sum_i$ `edit`(X_i, Y_i);
17 $\Delta_{\text{prune}} \leftarrow$ `edit`$($`string`$(X),$ `string`$(Y))$;
18 **while** $\Delta_{\text{prune}} < \Delta_{\text{current}}$ **do**
19 **for** $j \leftarrow i$ **to** *final token in* Y **do**
20 introduce a gap for Y_{j-1} (if $j > i$);
21 align X_i with Y_j;
22 **if** *partial distance* $\leq \Delta_{\text{prune}}$ **then**
23 recursively align X_{i+1}, Y_{j+1};
24 **end**
25 **end**
26 introduce a gap for X_i;
27 **if** *partial distance* $\leq \Delta_{\text{prune}}$ **then**
28 recursively align X_{i+1}, Y_i;
29 **end**
30 **if** *at end of either sequence* **then**
31 **if** *final distance* $\leq \Delta_{\text{current}}$ **then**
32 *currentAlign* $\leftarrow$ *final alignment*;
33 $\Delta_{\text{current}} \leftarrow$ *final distance*;
34 **end**
35 **end**
36 $\Delta_{\text{prune}} \leftarrow \Delta_{\text{prune}} + 1$;
37 **end**
38 **return** *currentAlign*;
39 **end**
40 **function** `ResolveGaps`(*alignment* $X \leftrightarrow Y$)
41 **foreach** *token* X_i *aligned to a gap* **do**
42 get closest token X_{i-k} aligned to Y_j;
43 **if** *concat*$(X_{i-k} \ldots X_i)$ *is suffix of* Y_j **then**
44 store alignment $[X_{i-k} \ldots X_i] \leftrightarrow Y_j$;
45 **end**
46 get closest token X_{i+l} aligned to Y_m;
47 **if** *concat*$(X_i \ldots X_{i+l})$ *is prefix of* Y_m **then**
48 store alignment $[X_i \ldots X_{i+l}] \leftrightarrow Y_m$;
49 **end**
50 **end**
51 **end**

the difference in length of the subsequence in a pair exceeds a threshold (7), in which case the anchor token is discarded and the subsequences are extended to the next anchor token pair. This happens occasionally in case of tokenization differences or tokens appearing outside of the main text (e.g., in a footnote). This is illustrated in Figure 2.1, where the position of the anchor token *den* in the two documents is far apart. Indeed these are different tokens, as the token in Text A is produced by end of line hyphenation, while the token in Text B is produced by a split of the token *aenden* into *aen* (*to*) and *den* (dative *the*). The other anchor tokens are retained, producing four subsequences for this example.

The subsequence pairs are assigned a trivial alignment (illustrated in Figure 2.2) where the tokens of each subsequence are aligned in order (15), and the total edit distance of this alignment is computed (16), used as a stop condition for expanding the search space. The search uses a pruning strategy for the alignment distance, with the maximum distance initialized as the edit distance of the full phrase (discarding token boundaries, line 17), and iteratively increased (36) until either an alignment is found (31) or the stop condition is reached (in which case the trivial alignment is returned). The search is executed recursively in three parallel directions (illustrated in Figure 2.3): align the current two tokens (20), introduce a gap by skipping a number of tokens in one document (19-25), and introduce a gap by skipping a number of tokens in the other document (26-29). Note that in Algorithm 1 the first direction is implemented as a special case of the second direction (i.e., skipping 0 tokens).

Subsequence alignments resulting from this procedure may contain gaps. The next stage of the algorithm attempts to resolve these gaps (illustrated in Figure 2.4) by concatenating gap-aligned tokens to the surrounding tokens (42, 46). If the combined tokens match with the align target of a constituent token, the combination is retained (44, 48), and otherwise the gap is confirmed. Finally, the alignment (including combinations and remaining gaps) is added to the XML representation of the automatic annotation output. A slightly simplified XML serialization example, including token alignment, is presented in Figure 3. In this example the automatically assigned lemma *salmon*, correct in modern usage, is manually retagged using historically correct tokens *shall* and *they*.

A custom tool was developed to facilitate the annotation process for the current corpus. This tool provides a web interface with document navigation based on the specific organization of the current corpus, integration with the Adelheid tagger, tokenization adjustments, propagation of manual corrections, integration of sociolinguistic annotation, and alignment of new and existing annotations. Integration of this interface with the Nederlab research infrastructure is currently under investigation.

5. Application of the Corpus

As an example application, the differences between main clauses and subordinate clauses have been analyzed regarding occurrences of bipartite negation. This type of negation is composed of a negative token (such as *not, never, nonetheless*), complemented with the negation clitic *en* (cf. *ne ... pas* in contemporary French). During the timeframe

1. **Text A** Aenden$_1$ Advocaet$_1$ | van$_{17}$ Hollandt$_1$ [...] verschej-$_1$ den$_1$ onwaerdicheden$_1$ | zijn$_4$ toegedreven$_1$
   Text B Aen$_1$ den$_1$ Advocaet$_1$ | van$_9$ Hollandt$_1$ [...] verschejden$_1$ onwaerdicheden$_1$ | zijn$_3$ toegedreven$_1$

2.

Aenden	Advocaet	$\emptyset$	Δ align $= 17$
Aen	den	Advocaet	$\Delta_{\text{prune}} = 1$

3.

Aenden			$\Delta = 3 \not\leq 1$, prune.
Aen			
$\emptyset$	Aenden		$\Delta = 3 \not\leq 1$, prune.
Aen	den		
Aenden	Advocaet		$\Delta = 12 \not\leq 1$, prune.
$\emptyset$	Aen		set $\Delta_{\text{prune}} = 2$
$\vdots$			set $\Delta_{\text{prune}} = 6$
Aenden			$\Delta = 3 \leq 6$, continue.
Aen			
Aenden	Advocaet		$\Delta = 9 \not\leq 6$, prune.
Aen	den		
Aenden	$\emptyset$		$\Delta = 6 \leq 6$, continue.
Aen	den		
Aenden	$\emptyset$	Advocaet	$\Delta = 6 \leq 6$, keep.
Aen	*den*	Advocaet	

4.

Aenden	$\emptyset$	Advocaet	*original*
Aen	den	Advocaet	
Aenden		Advocaet	*concat left*
Aen+den		Advocaet	*keep*
Aenden		Advocaet	*concat right*
Aen		den+Advocaet	*discard*

Figure 2: Alignment algorithm examples. 1: Anchor nodes and subsequence boundaries. 2: Trivial alignment. 3: Parallel recursive alignment with iterative pruning, showing the final search path *no gap, gap bottom, no gap*. 4: Resolving gaps by attempting concatenation in both directions.

```
<w xml:id="p.3.s.15.w.30" class="WORD">
  <t>zalmen</t>
  <!-- salmon -->
  <lemma class="zalm"/>
  <pos class="N(common,pl)" head="N">
    <feat class="common" subset="ntype"/>
    <feat class="mv" subset="number"/>
  </pos>
  <!-- shall -->
  <t textclass="gustave-cb">zal</t>
  <lemma class="zullen" textclass="gustave-cb"/>
  <pos class="V(fin,+nonlex,sg,3,pres)" head="V">
    <feat class="fin" subset="vtype"/>
    <feat class="+nonlex" subset="+lexical"/>
    <feat class="sg" subset="number"/>
    <feat class="3" subset="person"/>
    <feat class="pres" subset="tense"/>
  </pos>
  <!-- they -->
  <t_2 textclass="gustave-cb">men</t_2>
  <lemma_2 class="men" textclass="gustave-cb"/>
  <pos_2 class="PRON(indef,+nom)" head="PRON">
    <feat class="indef" subset="prtype"/>
    <feat class="+nom" subset="case"/>
  </pos_2>
</w>
```

Figure 3: Token alignment in XML serialization.

of the corpus the bipartite negation construct was in decline, before it disappeared from Dutch completely in later centuries. The Dutch language exhibits different word order in main clauses and subordinate clauses, which also affects the use of bipartite negation. Specifically, the verb in a main clause is positioned between the clitic and the negative (*hij en is niet*, English *he [clitic] is not*), while in a subordinate clause the verb is positioned at the end (*dat hij niet en is*, English *that he not [clitic] is*, meaning *that he is not*). It is hypothesized (Hoeksema, 2014) that the adjacent combination of the clitic and the negative in the subordinate clause became idiomatic, which has lead to a higher frequency of occurrence and a slower decline than the main clause counterpart which does not have adjacent tokens. This process is assumed to be most visible in the canonical form using the adverb *not*.

A quantitative analysis on the current corpus shows that the relative frequency of bipartite negation in subordinate clauses ($\sim$ 4%) is higher than for main clauses ($\sim$ 3%). Moreover, the frequency of *en niet* in subordinate clauses ($\sim$ 78% of all bipartite negations in subordinate clauses) is higher than in main clauses ($\sim$ 68%). From the *en niet* occurrences in subordinate clauses a majority ($\sim$ 55%) consists of directly adjacent tokens. Therefore, this preliminary quantitative analysis is consistent with the idiom hypothesis, although other hypotheses might explain the observations equally well.

Note that the identification of subordinate clauses and bipartite negation within a clause is a non-trivial problem given a corpus with part-of-speech tags only. Therefore, the percentages mentioned above are approximate and the observations need to be confirmed in future research. However, this example does show the potential of the current corpus to collect useful research examples in a fast and automatic way.

Other possible applications include the use of the annotated data for classification algorithms (e.g., to predict a topic category given a letter) or a part-of-speech tagger for historical text. In general, for historical (socio)linguistics the amount of annotated data is limited, which impacts the application of corpus linguistic methods and natural language processing in general. The current corpus aims to improve this situation, within the domain of 17th century Dutch, but also within digital humanities as a whole.

6. Bibliographical References

van den Bosch, A., Busser, B., Canisius, S., and Daelemans, W. (2007). An efficient memory-based morphosyntactic tagger and parser for Dutch. In *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, pages 99–114. Netherlands Graduate School of Linguistics.

Brants, T. (2000). Inter-annotator agreement for a German newspaper corpus. In *Proceedings of LREC 2000*.

Brugman, H., Reynaert, M., van der Sijs, N., van Stipriaan, R., Tjong Kim Sang, E., and van den Bosch, A. (2016). Nederlab: Towards a single portal and research environment for diachronic Dutch text corpora. In *Proceedings of LREC 2016*, pages 1277–1281.

Burke, P. (2005). *Towards a Social History of Early Modern Dutch*. Amsterdam University Press.

Busse, U. (2002). *Linguistic Variation in the Shakespeare Corpus*. John Benjamins Publishing Company.

Christ, O. (1994). A modular and flexible architecture for an integrated corpus query system. In *Proceedings of the 3rd Conference on Computational Lexicography and Text Research*, pages 22–32.

Donaldson, B. (1983). *Dutch. A linguistic history of Holland and Belgium*. Martinus Nijhoff.

Van Eynde, F., Zavrel, J., and Daelemans, W. (2000). Part of speech tagging and lemmatisation for the spoken dutch corpus. In *Proceedings of LREC 2000*, pages 1427–1434.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings ACL 2011*, pages 42–47. ACL.

van Halteren, H. and Rem, M. (2013). Dealing with orthographic variation in a tagger-lemmatizer for fourteenth century Dutch charters. *Language Resources and Evaluation*, 47(4):1233–1259.

Hinrichs, E. and Krauwer, S. (2014). The CLARIN research infrastructure: Resources and tools for eHumanities scholars. In *Proceedings of LREC 2014*, pages 1525–1531.

Hoeksema, J. (2014). The Middle Dutch negative clitic: Status, position and disappearance. *Lingua*, 147:50–68.

Leech, G. (1969). *A Linguistic Guide to English Poetry*. Pearson Education Limited. Edition: Routledge, 2013.

Nobels, J. and Rutten, G. (2014). Language norms and language use in seventeenth-century Dutch: negation and the genitive. In Gijsbert Rutten, editor, *Norms and usage in language history, 1600-1900. A sociolinguistic and comparative perspective.*, pages 21–48. John Benjamins Publishing Company.

Oostdijk, N., Reynaert, M., Monachesi, P., van Noord, G., Ordelman, R., Schuurman, I., and van de Ghinste, V. (2008). From D-Coi to SoNaR: A reference corpus for Dutch. In *Proceedings of LREC 2008*.

Plank, B., Hovy, D., and Søgaard, A. (2014). Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of EACL 2014*, pages 742–751. ACL.

Rutten, G. and van der Wal, M. (2014). *Letters as Loot: A sociolinguistic approach to seventeenth- and eighteenth-century Dutch*, volume 2 of *Advances in Historical Sociolinguistics*. John Benjamins Publishing Company.

Santorini, B. (1990). Part-of-speech tagging guidelines for the Penn treebank project. Technical Report MS-CIS-90-47, University of Pennsylvania. 3rd Revision.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). Technical report, Universität Stuttgart.

Schilling-Estes, N. (2002). Investigating stylistic variation. In J.K. Chambers, et al., editors, *The Handbook of Language Variation and Change*, pages 375–401. Blackwell.

Stowers, S. (1986). *Letter Writing in Greco-Roman Antiquity*. The Westminster Press.

Szmrecsanyi, B. (2005). Language users as creatures of habit: a corpus-based analysis of persistence in spoken English. *Corpus Linguistics and Linguistic Theory*, 1(1):113–149.

Voutilainen, A. (1999). An experiment on the upper bound of interjudge agreement: the case of tagging. In *Proceedings of EACL '99*, pages 204–208.

van der Wal, M., Rutten, G., and Simons, T. (2012). Letters as loot: Confiscated letters filling major gaps in the history of Dutch. In Marina Dossena et al., editors, *Letter Writing in Late Modern Europe*, pages 139–161. John Benjamins Publishing Company.

Zavrel, J. and Daelemans, W. (1999). Evaluatie van part-of-speech taggers voor het Corpus Gesproken Nederlands. Technical report, Universiteit Antwerpen.

7. Language Resource References

CGN Consortium. (2003). *Spoken Dutch Corpus*. Dutch Language Institute.

Committee DBNL. (2015). *Digital Library of Dutch Literature*. National Library of the Netherlands, www.dbnl.org, in Dutch.

de Vries, Matthias and te Winkel, Lammert. (1998). *Woordenboek der Nederlandsche Taal*. Dutch Language Institute, gtb.inl.nl, in Dutch.

Simplified Corpus with Core Vocabulary

Takumi Maruyama, Kazuhide Yamamoto
Nagaoka University of Technology
1603-1, Kamitomioka Nagaoka, Niigata 940-2188, JAPAN
{maruyama, yamamoto}@jnlp.org

Abstract

We have constructed the simplified corpus for the Japanese language and selected the core vocabulary. The corpus has 50,000 manually simplified and aligned sentences. This corpus contains the original sentences, simplified sentences and English translation of the original sentences. It can be used for automatic text simplification as well as translating simple Japanese into English and vice-versa. The core vocabulary is restricted to 2,000 words where it is selected by accounting for several factors such as meaning preservation, variation, simplicity and the UniDic word segmentation criterion. We repeated the construction of the simplified corpus and, subsequently, updated the core vocabulary accordingly. As a result, despite vocabulary restrictions, our corpus achieved high quality in grammaticality and meaning preservation. In addition to representing a wide range of expressions, the core vocabulary's limited number helped in showing similarities of expressions among simplified sentences. We believe that the same quality can be obtained by extending this corpus.

Keywords: Corpus, Controlled Languages, Lexicon

1. Introduction

Over the years, the number of foreigners visiting Japan has been increasing. Japan hosts around 24 million visitors in a year. In addition, there are about 2.47 million foreign residents in Japan, and this number is also increasing.

According to a survey conducted by the National Institute for Japanese Language and Linguistics, only 44.0% of Japan's foreign residents can speak English (Iwata, 2010). This ratio is lower than the percentage of people who can speak Japanese (62.6%). Foreigners can understand simple Japanese more easily than English. Therefore, we need to consider simple Japanese as a means of providing information for foreigners. Simple Japanese is the language with less complexity of vocabulary, grammar, and expression. This makes it possible to provide many text resources to a wide range of readers including Japan's foreign residents, foreign tourists, children, and intellectually disabled people.

We have been researching text simplification for several years (Moku et al., 2012; Kajiwara and Yamamoto, 2013; Kajiwara and Yamamoto, 2015). In this paper, we focus on vocabulary size because it can be defined objectively. There is a gap between the vocabulary size necessary for understanding the media and the vocabulary size necessary for understanding basic Japanese. According to a survey in modern Japanese magazines, 12,000 words are required to practically use Japanese (Tamamura, 2002). In addition, in order to understand TV shows sufficiently, it is necessary to know 17,000 words (National Institute Japanese Language and Linguistics , 1999). On the other hand, according to the standard of the Japanese Language Proficiency Test (called JLPT) Level 3 (level of understanding elementary Japanese), it is necessary to master 1,500 words. Moreover, Japanese vocabulary size essential for daily life is considered to be about 1,000 to 2,000 words (Kai, 2002). We think that eliminating this gap helps to understand the Japanese language.

We manually rewrote sentences which were extracted from newspaper articles and broadcast media news reports to sentences composed only of core vocabulary (2,000 words). The features of this corpus are as follows:

1. It is a large-scale corpus which has been aligned manually;
2. The simple sentences consist of only the core vocabulary, which was selected manually;
3. It contains the following three types of sentences: the original sentence, the simplified sentence and the English translation of the original sentence.

2. Core Vocabulary

We clearly distinguish core vocabulary and major vocabulary in this paper. These two are similar, but their purpose is different. Major vocabulary is a word list for a specific people or field. In many cases, it is selected from the viewpoint of education, that is, words that are frequently used in daily life are selected. The vocabulary defined in the JLPT is a typical example of major vocabulary. In contrast, core vocabulary is the minimum essential word list constituting the core of the language. Words that can express a wide range of things are selected. A typical example of core vocabulary is Ogden's basic English word list (Ogden, 1930).

2.1. Core Vocabulary Size

We set the core vocabulary size to 2,000 words according to the following observations. In Japanese, the JLPT requires 1,500 words in Level 3. In English, Ogden's Basic English has 850 words, and Simple English Wikipedia allows us to use Ogden's 850 words, 1,500 words of VOA Special English and proper nouns. In addition, the number of definition words is 2,000 in the Longman Dictionary of Contemporary English. Based on the above information, we expect that there are considerable explanatory abilities using 2,000 words as the Japanese language vocabulary size.

2.2. Core Vocabulary Definition

We selected 2,000 words that preserve the meaning of various sentences as much as possible. In the case of syn-

onyms, we chose the simplest word. In addition, we selected the core vocabulary according to the UniDic word segmentation criterion. Ambiguous words in the part-of-speech (POS) tag were considered to be different words, while polysemous words, with the same POS tags, were considered as a single word. For the definition of core vocabulary, the following were excluded from simplification:

1. Symbols such as punctuation marks and parentheses;
2. Proper nouns and some named entities such as people and location;
3. Unknown words in a word segmentation process.

3. Construction of the simplified corpus

3.1. Target sentences

We used a "small parallel enja: 50k En/Ja Parallel Corpus for Testing SMT Methods[1]" as the original text for simplification. This dataset is a part of Japanese-English parallel corpus (called Tanaka Corpus) (Tanaka, 2001) extracted from newspaper articles and broadcast media news reports published on the World wide web. The Japanese part of this dataset contains sentence lengths of 4 to 16 words. The reason we adopted this text is as follows:

1. It is a moderate work scale for us;
2. There are many short sentences on the character of the Tanaka corpus;
3. It is part of the Tanaka Corpus in which the license is Creative Commons CC-BY, and the original text has already been released on the Web.

3.2. Construction Method

We decided to rewrite all 50,000 Japanese sentences in "small parallel enja: 50k En/Ja Parallel Corpus for Testing SMT Methods" in simple Japanese with the help of five annotators. This dataset was already divided into five files at the time of distribution, and one file was assigned to one annotator. Consultation as well as adjustment among annotators was performed continuously, and the work content was always accessible to all annotators.
The task of constructing the corpus and selecting the core vocabulary was performed according to the following procedures:

1. We selected 2,000 UniDic high-frequency words in the BCCWJ Corpus [2] as the initial core vocabulary.
2. We performed word analysis on the original sentence. If it contained complex words, it was simplified. Here, complex words mean all words except the core vocabulary. Simplification was done in sentence units.
3. During simplification, annotators recorded the words which they want to be added or deleted from the core vocabulary. Annotators collect these words at a certain time and change the core vocabulary with the consensus of five annotators. During this work process, we accept that it is possible to temporarily increase or decrease the number of words to more than 2,000.
4. If the core vocabulary was modified, the operation from step 2 above would be repeated.

Rank	Word	Example of original sentence
3169	青い (blue)	彼女の青い靴は服によく合っている。 (Her blue shoes suit her clothes very well.)
3321	貸す (to lend)	彼女はあなたに本を貸すだろう。 (She will lend you a book.)
4628	泳ぐ (to swim)	彼は上手に泳げる。 (He can swim well.)
5370	アレルギー (allergic)	魚アレルギーなんです。 (I am allergic to fish.)
6481	こんにちは (hello)	小さな男の子が私にこんにちはと言った。 (The little boy said hello to me.)
7565	宿題 (homework)	あなたはもう英語の宿題を終えましたか？ (Have you finished your English homework yet?)

Table 2: Some examples of the core vocabulary and frequency ranking in BCCWJ Corpus.

4. Core Vocabulary Analysis

Some examples of the core vocabulary are listed in Table 1. Furthermore, examples of core words and their frequency ranking in BCCWJ Corpus are displayed Table 2. As mentioned in 3.2., we selected top 2,000 UniDic high frequency words in the BCCWJ Corpus as the initial core vocabulary, and we added or deleted words from it. As shown in Table 2, words with a low rank (less than 2,000) are also included in the core vocabulary. These are words that constitute the core of Japanese expression. This result confirms the argument that it is insufficient to use the frequency information alone when selecting the core vocabulary (Matsuda et al., 2010).

5. Corpus Analysis

We evaluated the corpus using the following three attributes: corpus statistics (section 5.1.), examination of corpus quality (section 5.2.) and the agreement between simplification annotators (section 5.3.).

[1] https://github.com/odashi/small_parallel_enja

[2] http://pj.ninjal.ac.jp/corpus_center/bccwj
BCCWJ is a corpus that collected Japanese texts from various genres such as books, magazines, newspapers, white papers, blogs, net bulletin boards, textbooks, and laws.

POS	Number of words	Example of words
Determiner	14	あの, あらゆる, ある, いろんな, いわゆる, 大きな, 同じ, この, こんな, その, そんな, 小さな, どの, どんな
Conjunction	15	あるいは, 一方, 及び, が, さて, さらに, しかし, しかも, すなわち, そして, ただ, ただし, で, なお, また
Interjection	16	ああ, あっ, あの, ありがとう, いいえ, いや, うん, ええ, おはよう, こんにちは, こんばんは, さあ, さようなら, ねぇ, はい, やあ
Prefix	19	相, お, 高, ご, 御, 再, 小, 新, 全, 総, 大, 第, 中, 非, 不, 本, 未, 無, 約
Pronoun	22	あなた, あれ, いずれ, いつ, 何時, 彼女, 彼, ここ, こちら, こっち, これ, そこ, それ, 誰, どこ, どちら, どれ, なに, なん, 何, 僕, 私
Modal verb	22	させる, ぬ, せる, た, だ, たい, たり, だ, ちゃう, てる, でる, です, ない, なり, べし, まい, ます, や, らしい, られる, り, れる
Postpositional particle	60	い, か, から, が, が, くらい, ぐらい, けど, けれど, こそ, しか, しも, ずつ, ぞ, たり, だけ,って,, で, じゃ, で, と, とも, に, ね, の, のみ, は, へ, ほど, まで, も, や, やら, よ, より, わ, を
Adverb	74	あまり, いかに, 一番, いったい, いろいろ, おそらく, かつて, 必ず, かなり, きちんと, きっと, こう, さらに, しっかり, 少し, ずっと, たくさん, ただ, 例えば, ちゃんと, つまり, とても, まるで, むしろ, より
Na-adjective	79	明らか, あまり, 安心, 安全, 安定, 異常, 一定, いっぱい, 嫌, いろいろ, 初, 同じ, 危険, 急, 共通, 嫌い, きれい, 逆, 偶然, 最高, 残念, 幸せ, 静か, 自然, 失礼, 心配, 十分, 正確, 誠実, そう, そんな, 大切, 大変, みたい
Suffix	83	位, 員, 院, 園, 下, 家, 日, 方, 官, 観, 間, 館, 外, 学, 型, 君, 系, 軒, げ, 個, 後, ごと, さ, 歳, 冊, 様, さん, 士, 師, 紙, 室, 者, 車, 所, 書, 署, 心, 次, 上, 場, 状, 制, 中, 帳, 長, つ, 的, 用, ら, らしい, 料, 力, 類
Adjective	93	青い, 赤い, 明るい, 浅い, 新しい, 厚い, 暑い, 熱い, 甘い, 荒い, 忙しい, 痛い, 薄い, 美しい, 嬉しい, 美味しい, 多い, 大きい, 遅い, 重い, 面白い, 賢い, 固い, 悲しい, 軽い, 暗い, 苦しい, 濃い, 細かい, 怖い, 寒い
Verbal noun	221	挨拶, 合図, 案内, 意見, 意識, 維持, 位置, 一緒, 移動, 違反, 意味, イメージ, 印刷, 運転, 運動, 影響, 営業, 遠慮, 会議, 会計, 解決, 開催, 開始, 回答, 開発, 回復, 買い物, 会話, 科学, 確認, 確保, 活動, 活用, 感覚, 関係, 観光, 看護, 観察, 感謝, 完成, 監督, 感動, 乾杯, 管理, 学習, 我慢, 記憶, 企画, 規制, 期待, 希望
Verb	370	愛する, 会う, 合う, 上がる, 諦める, 飽きる, 開ける, あげる, 上げる, 挙げる, 遊ぶ, 与える, 当たる, 扱う, 集まる, 集める, 当てる, 編む, 謝る, 洗う, いる, 入れる, 浮かぶ, 受ける, 動かす, 動く, 失う, 歌う, 打つ, うつる, 売る, 描く, 選ぶ, 得る, 追いつく, 追う, 応ずる, 起きる, 補う, おく, 置く, 送る, 遅れる, 起こす, 行う, 怒る, 起こる, 教える, 押す, 訪れる, 驚く, 覚える, 思う
Noun	912	鉛筆, 尾, 甥, 王, 黄色, 多く, 祖父, 丘, 陸, おかげ, 奥, おじ, 夫, 音, 弟, 男, 少女, 女, おば, 祖母, 首, 面, 思い, 玩具, 表, 親, 音楽, 温度, 何, 夏, 火, 花, 母, 回, 階, 会社, 階段, 顔, 価格, 化学, 鏡, 鍵, 限り, 角, 家具, 過去, 傘, 重, 菓子, 歌手, 数, 風, 風邪, 家族, 形, 方, 肩, 敵, カタログ, 価値, 家庭, 過程, 門, 金, 金持ち, 鞄, 株, 壁, 紙, 髪, 上司, 瓶, カメラ, 火曜, 体, 川, 代わり, 巻, 感, 考え, 環境, 感じ, 感情, 観点, 看板, カード, 外, 外国, 学生, 楽器, 学校, 月, 画面, ガラス, 側, 元, 眼鏡, 木, 機, 気, 機会, 機械, 期間, 機関, 企業, 記事, 季節, 規則, 北, 切手, 昨日, きのこ, 気分, 基本, 気持ち, 着物, 客, 今日, 教科, 教会, 教室, 興味, 局, 曲, 距離, 災害, 最近, 最後, 最初, 財布, 坂, 作品, 酒, 実, 皿, 自ら, 人間, 人気, 神社, 人生, 人物, 水, 数学, 数字, 姿, 全て, スープ, 図, 図書, ズボン, 背, せい, 星, 西, 成果, 性格, 精神, 政治, 政党, 制度, 政府, 生物, 世界, 席, 石炭, 責任, 石油, 世代, 石鹸, セット, 節, 説, 雪, 線, 先月, 選手, 先週, 先生, 税, 絶対特徴, 時計, 床, 所, ところ, 年, 都市, 土地, 途中, 隣,, とも, 共, 友達, トラック, 鳥, 鶏, 豚, ドア, 同, 道, 動画, 道具, 同時, 動物, 道路, ドレス, 泥棒, 名, ナイフ, 内容, 仲, 仲間, 流れ, なし, 納豆, 鍋, 名前, 波, 南, 何曜, におい, 肉, 日, 日時, 日常, 日曜, 目的, 目標, 木曜, 文字, モデル, 者, もも, 森, 問題, 野球, 訳, 役割, 野菜, 休み, やつ, 屋根, 夕方, 勇気, 夕食, 郵便, 夕焼け, 指, 夢, わけ, 割

Table 1: Some examples of the core vocabulary.

	S-BLEU	Version	Sentence	English translation of the left column
(1)	0.000	Original	疑い の 余地 はない。	There is no room for doubt.
		Simplified	明らかだ。	It is clear.
(2)	0.090	Original	日本では、月給 です。	In Japan, salary is on monthly basis.
		Simplified	日本では、月に 1 度、働いた分のお金がもらえます。	In Japan, you receive money once for working a month.
(3)	0.452	Original	そこに 署名 してください。	Please sign there.
		Simplified	そこに名前を書いてください。	Please write your name there.
(4)	0.517	Original	交通 渋滞 のため、私は遅れました。	Because of the traffic jam, I was late.
		Simplified	道路が混んでいたため、私は遅れました。	I was late because the road was crowded.
(5)	0.525	Original	時計がどこか 故障 しているらしい。	The clock seems out of order.
		Simplified	時計がどこか壊れているらしい。	The clock seems to be broken.
(6)	0.598	Original	いつも 手近 に辞書を持っていなさい。	Always have your dictionary near at hand.
		Simplified	いつでも使えるように辞書を持っていなさい。	Have your dictionary so that you can use it anytime.
(7)	0.701	Original	彼は 一生懸命 英語を勉強したに違いない。	He must have studied English with utmost effort.
		Simplified	彼は頑張って英語を勉強したに違いない。	He must have studied English hard.
(8)	0.783	Original	彼は簡単に 非 を認めるような人ではない。	He is not a man to admit his faults easily.
		Simplified	彼は簡単に間違いを認めるような人ではない。	He is not a man to admit his mistakes easily.
(9)	0.791	Original	十分に 休養 をとることは、非常に大切です。	It is very important to take a rest.
		Simplified	十分に休みをとることは、非常に大切です。	It is very important to take a break.
(10)	0.816	Original	あいにく 私はお金を持っていない。	Unfortunately I have no money with me.
		Simplified	残念ながら私はお金を持っていない。	I'm afraid that I have no money with me.

Table 3: Examples of sentence pairs in our corpus and S-BLEU. The underlined words in the original sentences are complex words.

	Original	Simplified
Total #sentences	50,000	50,000
Total #tokens	490,021	516,881
Total #words (unique tokens)	8,786	2,238
Avg. #characters per sentence	14.79	15.35
Avg. #words per sentence	9.80	10.34

Table 4: Corpus statistics. We show the number of words in the vocabulary after changing to the basic form based on the UniDic dictionary. This vocabulary size also includes words such as proper nouns and symbols (238 words). Therefore, the vocabulary size of the simplified side is more than 2,000 words.

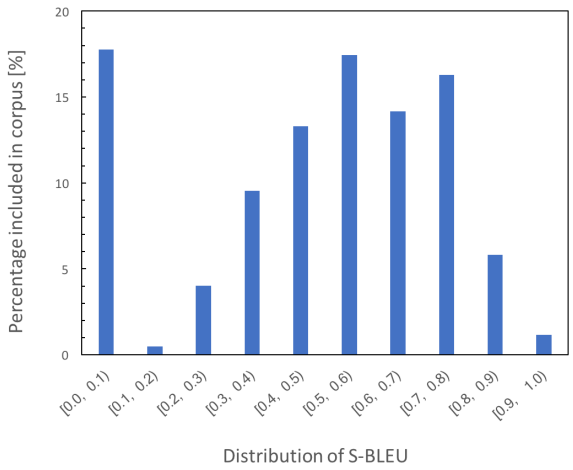


Figure 1: Distribution of S-BLEU.

5.1. Corpus Statistics

Table 4 shows the corpus statistics. The average sentence length and the average number of words per sentence of the simplified corpus are longer than those of the original corpus. Complex words in the original sentences often include kanji compound words such as "余地 (room)", "渋滞 (traffic jam)" and "一生懸命 (with utmost effort)". Annotators tried to simplify such words by using phrases while preserving the meaning of the original sentences as much as possible. As a result, sentences would become longer. A good example is shown in row (2) in Table 3. The expression "月給 (monthly salary)" was simplified to "月に 1 度働いた分のお金をもらう (to receive money once for working a month)" by annotators. This implies that short sentences were not necessarily simple sentences in Japanese.

22,009 original sentences consist of only core vocabulary. Therefore, it was possible to cover 40% of the sentences in

Grammaticality	
4	It is a grammatically correct sentence.
3	It has some grammatical mistakes, but you can understand the meaning of the sentence.
2	The grammar is incorrect, but you can guess the meaning.
1	It has many grammatical mistakes and you cannot understand the meaning.

Meaning preservation	
4	The meanings of the two sentences are the same.
3	The meanings of the two sentences are different, but the overall meaning is the same.
2	The meanings of the two sentences are different, but the meanings of the parts are the same.
1	The meanings of the two sentences are quite different.

Table 5: Evaluation criteria presented to the evaluator

Version	Sentence	English translation of the left column	G	M
Original	私は毎日車で通勤している。	I commute by car every day.	4.0	4.0
Simplified	私は毎日車で仕事に行っている。	I go to work by car every day.		
Original	私は忙しくて休暇が取れない。	I cannot afford the time for a vacation.	4.0	3.8
Simplified	私は忙しくて休みを取ることができない。	I cannot afford the time for a holiday.		
Original	たびたびそこに行った事がある。	I have been there scores of times.	4.0	2.2
Simplified	何度かそこに行ったことがある。	I have been there several times.		
Original	花はまだ蕾だ。	The flowers are still in bud.	3.6	3.4
Simplified	花はまだ開いていない。	The flowers are not open yet.		

Table 6: Examples of manual evaluation for gramaticality (G) and meaning preservation (M)

the corpus only with the core vocabulary that we selected. We classified 27,991 sentence pairs which the original sentence and simple sentence did not match under simplification according to S-BLEU [3] scoring.

Figure 1 shows that our corpus includes many sentence pairs for [0.0, 0.1], [0.5, 0.6] and [0.7, 0.8]. In the sentences of S-BLEU [0.0, 0.1], the original sentences are largely transformed while preserving their meaning. Additionally, in some cases, the whole sentence was changed as shown in row (1) in Table 3. In the range [0.5, 0.6], there was a tendency to simplify phrase units. For example, in row (4), "交通渋滞のため (because of the traffic jam)" was simplified to "道路が混んでいたため (because the road was crowded)". Also, in row (6), "いつも手近に (always at hand)" was simplified to "いつでも使えるように (always be able to use it)". In [0.7, 0.8], there was a tendency to simplify only one word. For example, in row (7), "一生懸命 (with utmost effort)" was simplified to "頑張って (hard)". Also, in row (9), "休養 (rest)" was simplified to "休み (rest)". From the above observations, we see that parts which could not be covered by a word unit replacement existed significantly in the simplification. Therefore, it was necessary to simplify phrase units and sentence units depending on the circumstances.

5.2. Manual Examination of Corpus

We selected 100 sentences at random from the corpus and classified the simplification operation by one annotator. We counted the changes of a whole phrase (for example, "交通渋滞のため (because of the traffic jam)" → "道路が混んでいたため (because the road was crowded)") as one change. As a result, the simplification operations in the 100 sentences were paraphrasing and a combination of paraphrasing and insertion only. In these sentences, insertion is an operation that inserts only a postpositional particle of Japanese to construct a fluent sentence. Therefore, this simplification corpus is a corpus that focuses only on paraphrases.

[3] S-BLEU is sentence-wise BLEU score.

Grammaticality			Meaning preservation		
Mean	Mode	Median	Mean	Mode	Median
3.81	4	4	**3.72**	4	4

Table 7: Results of manual evaluation concerning grammaticality and meaning preservation. We asked annotators to randomly evaluate grammaticality and meaning preservation in 100 sentences selected from the corpus using crowdsourcing. The evaluation was done in four stages from 1 to 4 (higher marks indicate better output).

To analyse the quality of the simplified corpus, we manually evaluated the corpus from the viewpoint of grammaticality and meaning preservation. A 100 sentence pairs in which the original sentence and simple sentence did not match were randomly selected from the corpus and evaluated. The evaluation was performed by five Japanese annotators using crowdsourcing. We divided the evaluation into four stages, from 1 to 4 (higher marks indicate better output). Table 7 shows the manual evaluation. Although the vocabulary used in the simple sentences decreased to 25% of the original sentences, both scores are high. That is, even if the vocabulary was restricted, most of the meaning of the original sentence could still be expressed. How-

Annotator	S-BLEU	Annotator	S-BLEU
Ⅰ - Ⅱ	0.602	Ⅱ - Ⅳ	0.594
Ⅰ - Ⅲ	0.633	Ⅱ - Ⅴ	0.581
Ⅰ - Ⅳ	0.611	Ⅲ - Ⅳ	0.604
Ⅰ - Ⅴ	0.593	Ⅲ - Ⅴ	0.585
Ⅱ - Ⅲ	0.613	Ⅳ - Ⅴ	0.589

Table 8: Inter-annotator agreement. I-V represent five simplification annotators. This table shows the BLEU score between each annotator.

ever, the score did not reach up to 4.00, which we think was owing to vocabulary restriction. It was not always possible to represent the meaning of the original sentence perfectly. With regard to grammar, we observed that some low scores were associated with sentences that had become longer and ambiguous owing to vocabulary restriction.

5.3. Inter-Annotator Agreement

We asked five simplification annotators to simplify the same 100 sentences, which were selected from the Tanaka Corpus, and evaluated the inter-annotator agreement by S-BLEU. These 100 sentences consist of 4 to 16 words from the Tanaka Corpus. In addition, these 100 sentences do not include sentences comprising only the core vocabulary.

This evaluation result is shown in Table 8. The values range between 0.58 and 0.63. In a similar study, Mitkov and Štajner (2014) constructed a simplified corpus with fewer simplification rules. They showed that the S-BLEU score of the three annotators is 0.44 to 0.53. Compared their corpus, we observed that our corpus was not dependent on annotators and that it was stable. Table 8 shows a high score of S-BLEU owing to the fact that the simplified corpus consists of only core vocabulary. This restriction helped simple sentences show similarity in expression.

6. Related Works

6.1. Simplified Corpora

There are many simplification resources for various languages (Caseli et al., 2009; Zhu and Bernhard, 2010; Klaper et al., 2013; Brunato et al., 2015; Xu et al., 2015). Text simplification has been researched by various approaches such as lexical simplification (Horn et al., 2014; Štajner and Glavaš, 2015; Paetzold, 2016; Paetzold and Specia, 2017), machine translation approaches (Coster and Kauchak, 2011; Wubben, 2012; Štajner et al., 2015a; Štajner et al., 2015b; Xu et al., 2016; Nisioi et al., 2017) and rule-based approaches (Siddharthan, 2014) using simplified corpora. However, in Japanese, although attempts have been made for lexical simplification (Kajiwara and Yamamoto, 2013; Imono et al., 2013; Kajiwara and Yamamoto, 2015; Hading and Matsumoto, 2016), there is no prior research on sentence simplification. The absence of a simplification corpus could be the primary reason for this. Furthermore, there are no large-scale simplified data equivalents to the Simple English Wikipedia; as a result, no attempt has been made to construct a simplification corpus. Therefore, we used crowdsourcing to construct a Japanese simplified corpus containing 34,300 sentence pairs (Katsuta and Yamamoto, 2018). In addition, the corpus contains 100 sentence pairs having 7 references as data for evaluation. The simple sentences consist of only the core vocabulary.

We evaluated the crowdsourced corpus from the viewpoint of grammaticality, meaning preservation and inter-annotator agreement with the same criteria as this paper. Compared to evaluations of the crowdsourced corpus, evaluations in this paper show better results in meaning preservation and inter-annotator agreement. Therefore, the simplified corpus in this paper is higher quality than the simplified corpus constructed using crowdsourcing.

6.2. Sentence simplification with core vocabulary

We performed automatic text simplification by using a machine translation approach with this paper's corpus (Maruyama and Yamamoto, 2017). As a result, this approach greatly outperforms existing lexical simplification system. In addition, we constructed 32 models according to the quantity and quality of training data, development data. A comparison of these models showed that data with a medium S-BLEU score are most effective for automatic text simplification by a machine translation approach.

7. Conclusion

We have constructed the Japanese simplified corpus and the core vocabulary through many alternate repetitions of the simplifying original sentences as well as through updating the core vocabulary. The corpus contained 50,000 manually simplified and aligned sentences. This core vocabulary was restricted to 2,000 words, which were selected by accounting for several factors such as meaning preservation, variation, simplicity. Although the vocabulary was restricted, our corpus achieved high quality grammaticality and meaning preservation. In addition, vocabulary restriction helped simplified sentences show similarities of expressions.

8. Acknowlegement

This work was supported in part by JSPS KAKENHI Grants-in-Aid for Challenging Research (Exploratory) Grant ID 17K18481

9. References

Brunato, D., Dell'Orletta, F., Venturi, G., and Montemagni, S. (2015). Design and Annotation of the First Italian Corpus for Text Simplification. *Proceedings of the 9th Linguistic Annotation Workshop*, pages 31–41.

Caseli, H. M., Pereira, T. F., Specia, L., Pardo, T. A. S., and Aluisio, S. M. (2009). Building a Brazilian Portuguese parallel corpus of original and simplified texts. *In the Proceedings of CICLing*.

Coster, W. and Kauchak, D. (2011). Simple English Wikipedia: A New Text Simplification Task. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 665–669.

Hading, M. and Matsumoto, Y. (2016). Japanese Lexical Simplification for Non-Native Speakers. *Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications*, pages 92–96.

Horn, C., Manduca, C., and Kauchak, D. (2014). Learning a Lexical Simplifier Using Wikipedia. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2:458–463.

Imono, M., Yoshimura, E., Tsuchiya, S., and Watabe, H. (2013). Proposal of a Method to Convert Difficult Words in Newspaper Articles to Plain Expressions. *Journal of Natural Language Processing of Japan Volume 20 Number 2*.

Iwata, K. (2010). The Preference for English in Linguistic Services: 'Japanese for Living Countrywide Survey' and Hiroshima. *The Japanese Journal Language of in Society*, pages 81–94.

Kai, M. (2002). *A way of Teaching Vocabulary [vocabulary table compilation]*. Mitsumura Tosho Publishing Co. Ltd.

Kajiwara, T. and Yamamoto, K. (2013). Selecting proper lexical paraphrase for children. *Proceedings of The 25th Conference on Computational Linguistics and Speech Processing (ROCLING)*, pages 59–73.

Kajiwara, T. and Yamamoto, K. (2015). Evaluation dataset and system for japanese lexical simplification. *roceedings of the ACL-IJCNLP 2015 Student Research Workshop (ACL SRW 2015)*, pages 35–40.

Katsuta, A. and Yamamoto, K. (2018). Crowdsourced corpus of sentence simplification with core vocabulary. *Proceedings of 11th edition of the Language Resources and Evaluation Conference*.

Klaper, D., Sarah, E., and Martin, V. (2013). Building a German / Simple German Parallel Corpus for Automatic Text Simplification. *Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 11–19.

Maruyama, T. and Yamamoto, K. (2017). Sentence simplification with core vocabulary. *Proceedings of the International Conference on Asian Language Processing*, pages 363–366.

Matsuda, M., Kodama, S., Takemoto, Y., Ishizaka, T., Mori, A., Kawamura, Y., and Yamamoto, K. (2010). Extraction of common Japanese core vocabulary from corpus using familiarity. *Proceedings of the 16th annual meeting of the Association for Natural Language Processing*, pages 579–582.

Mitkov, R. and Štajner, S. (2014). The Fewer, the Better? A Contrastive Study about Ways to Simplify. *Proceedings of the Workshop on Automatic Text Simplification: Methods and Applications in the Multilingual Society*, pages 30–40.

Moku, M., Yamamoto, K., and Makabi, A. (2012). Automatic easy japanesevtranslation for information accessibility of foreigners. *Proceedings of Coling-2012 Workshop on Speech and Language Processing Tools in Education (SLP-TED)*, pages 85–90.

National Institute Japanese Language and Linguistics . (1999). *Characteristics of High Frequency Vocabulary in Television Show*. Dainippon Tosho Publishing Co. Ltd. (in Japanese).

Nisioi, S., Štajner, S., Ponzetto, S. P., and Dinu, L. P. (2017). Exploring Neural Text Simplification Models. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 85–91.

Ogden, C. K. (1930). *Basic English: A General Introduction with Rules and Grammar.* Paul Treber Co. Ltd. London.

Paetzold, G. H. and Specia, L. (2017). Lexical Simplification with Neural Ranking. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, 2:34–40.

Paetzold, G. H. (2016). Reliable Lexical Simplification for Non-Native Speakers. *In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 3761–3767.

Siddharthan, A. (2014). Hybrid text simplification using synchronous dependency grammars with hand-written and automatically harvested rules. *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 722–731.

Štajner, S. and Glavaš, G. (2015). Simplifying Lexical Simplification : Do We Need Simplified Corpora ? *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 63–68.

Štajner, S., Calixto, I., and Saggion, H. (2015a). Automatic text simplification for Spanish: Comparative evaluation of various simplification strategies. *International Conference Recent Advances in Natural Language Processing, RANLP*, pages 618–626.

Štajner, S., Hannah, B., and Saggion, H. (2015b). A Deeper Exploration of the Standard PB-SMT Approach to Text Simplification and its Evaluation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 823–828.

Tamamura, H. (2002). *Vocabulary and Meaning*. ALC PRESS Inc(in Japanese).

Tanaka, Y. (2001). Compilation of A Multilingual Parallel Corpus. *Conference of the Pacific Association for Computational Linguistics*.

Wubben, S. (2012). Sentence Simplification by Monolingual Machine Translation. *Proceedings of the 50th AnnualMeeting of the Association for Computational Linguistics: Long Papers*, 1(July):1015–1024.

Xu, W., Callison-burch, C., and Napoles, C. (2015). Problems in Current Text Simplification Research : New Data Can Help. *Transactions of the Association for Computational Linguisics*, 3:283–297.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the ACL*, 4:401–415.

Zhu, Z. and Bernhard, D. (2010). A Monolingual Tree-based Translation Model for Sentence Simplification.

Proceedings of the 23rd International Conference on Computational Linguistics, pages 1353–1361.

A Pragmatic Approach for Classical Chinese Word Segmentation

Shilei Huang, Jiangqin Wu
College of Computer Science and Technology, Zhejiang University
Yuquan Campus, Zhejiang University, Zheda Road, Hangzhou, Zhejiang, China
{21621129, wujq}@ zju.edu.cn

Abstract

Word segmentation, a fundamental technology for lots of downstream applications, plays a significant role in Natural Language Processing, especially for those languages without explicit delimiters, like Chinese, Korean, Japanese and etc. Basically, word segmentation for modern Chinese is worked out to a certain extent. Nevertheless, Classical Chinese is largely neglected, mainly owing to its obsoleteness. One of the biggest problems for the researches of Classical Chinese word segmentation (CCWS) is lacking in standard large-scale shareable marked-up corpora, for the fact that the most excellent approaches, solving word segmentation, are based on machine learning or statistical methods which need quality-assured marked-up corpora. In this paper, we propose a pragmatic approach founded on the difference of t-score (dts) and Baidu Baike (the largest Chinese-language encyclopedia like Wikipedia) in order to deal with CCWS without any marked-up corpus. We extract candidate words as well as their corresponding frequency from the *Twenty-Five Histories* (*Twenty-Four Histories* and *Draft History of Qing*) to build a lexicon, and conduct segmentation experiments with it. The F-Score of our approach on the whole evaluation data set is 76.84%. Compared with traditional collocation-based methods, ours makes the segmentation more accurate.

Keywords: Classical Chinese, Word Segmentation, Difference of T-score, Encyclopedia, Collocation

1. Introduction

Till now, great achievements have been made in modern Chinese word segmentation (CWS) while there is little progress in Classical Chinese word segmentation (CCWS) for the obsoleteness of Classical Chinese. However, it's worth studying it. As a fundamental technology, word segmentation is a prerequisite of making deep analyses on Classical Chinese literature. From the word frequency we can discover the transition of words which can explain some linguistic phenomena, historical facts, traditional folk cultures, social culture, geographical information and etc. Furthermore, we can extract entities like person, location, official title, date and so forth, to dig into the history and culture, rather than get superficial knowledge by basic search like string matching. Additionally, many downstream applications like knowledge graph and QA system need to be built from these entities. In recent years, many researchers have made analyses and explorations of unstructured or structured Classical Chinese literature (Xu, 2016; Ouyang, 2016; Liu et al., 2015; Lee and Wong, 2013; Li et al., 2012; Fang, Lo and Chinn, 2009).

According to the estimate of experts, the number of the existing Chinese ancient books is about from 80,000 to 100,000. More than 40,000 books (around 4.8 billion Chinese Characters) have been converted into digital version (Ouyang, 2016). However, there are few marked-up corpora, let alone some of them being not shareable. Currently, the most known Classical Chinese corpora, which are marked up with POS (part of speech) tags and segmented into words, are Academia Sinica Ancient Chinese Corpus[1], Sheffield Corpus of Chinese[2] and CityU Treebank of Classical Chinese Poems[3]. Unfortunately, they are not enough to do a comprehensive analysis over a long period of history. And for the inconsistency of tagging and segmenting standards of the corpora, it's quite tough to aggregate them.

Building Classical Chinese corpus is much more expensive than building modern one. On the one hand, there are unique words and expressions in each dynasty, some of which only appear within a certain period of time and are no longer used in modern life. The meaning of words is changing along with time. Thus, comprehensive relevant knowledge is required to segment the sentence. On the other hand, the range of the Classical Chinese literature is generally from the end of the Spring and Autumn period through to the end of the Qing dynasty, nearly 3000 years. Only with considerable literature of each era being marked-up, can we make an all-sided analysis on literatures over the long history by means of supervised approaches. Nonetheless, it's costly and inefficient to do so. For instance, it took Shi el al. (2010) 4 years to manually segment and annotate word POS of 25 literatures of Pre-Qin era.

In order to tackle these problems, we propose a pragmatic approach to create a lexicon, with which researchers are able to segment the Classical Chinese texts for other pertinent studies or get preprocessed corpus for creating a more accurate one efficiently. The basic idea behind our method is that firstly, we get the statistical information of all bigrams by iterating texts in the corpus; secondly, when we iterate the texts again, candidate words (collocation) as well as their frequency are extracted in accordance to the thought of integrating the difference of t-score with Baidu Baike.

The contributions of this paper are as follows:

1. To the best of our knowledge, this is the first work that combines encyclopedia and traditional statistical methods to construct lexicon for word segmentation.
2. From what we can tell, this is the first work that studies on CCWS throughout the whole history of Classical Chinese literature.
3. Alleviate the problem of lacking in standard large-scale shareable marked-up corpora of Classical Chinese.
4. Comparison experiments are carried out to prove the effectiveness of our approach in CCWS.

The rest of this paper is organized as follows. Related work will be introduced in Section 2. Data we use will be explained in Section 3. Methodology and experiments will

[1] http://ancientchinese.sinica.edu.tw/c_intro.html
[2] https://hridigital.shef.ac.uk/scc/
[3] http://classicalchinese.lt.cityu.edu.hk/

be introduced in Section 4 and Section 5 respectively. Finally, concluding remarks are given in Section 6.

2. Related Work

Up to now, numerous approaches of word segmentation have been proposed. These methods can be roughly classified as lexicon-based, statistically-based or neural network-based methods. Hybrid methods are not introduced here.

2.1 Lexicon-based approaches

The most popular lexicon-based algorithms are forward maximum matching method (FMM), and backward maximum matching method (BMM). The performance of these methods mostly depends on the coverage of the lexicon. Qiu and Huang (2008) proposed a heuristic hybrid CCWS method. Using *Hanyu Da Cidian* as a basic dictionary, segments the texts with BMM. For increasing the accuracy, they count the frequency of words that have already appeared, extract the words with high frequency and add new words to the word list. As a general-purpose dictionary, the limitation of *Hanyu Da Cidian* is rather apparent, like sparseness, data incompleteness and bias when it's utilized for CCWS over literature of different eras. Since there is no appropriative dictionary, researchers tend to combine lexicons with statistical methods instead of adopting lexicon-based approaches alone.

2.2 Statistically-based approaches

Statistically-based approaches can be further divided into labelling-based or collocation-based (corpus-based) methods.

Xue (2003) first proposed the character-based tagging approach, which treats word segmentation as a sequence tagging problem, assigning corresponding labels to the characters. The labels indicate the location of each character, beginning of, inside or end of a certain word. Xue's work leads to some subsequent researches (Peng, Feng and McCallum, 2004; Tseng et al., 2005) on integrating character labelling with statistical models like HMM, MEMMs, CRF and etc. Afterwards, Zhang and Clark (2007) took the word-level information into consideration, proposing a word-based CWS approach using a discriminative perceptron learning algorithm. The prerequisite of these kinds of solutions is high-quality marked-up corpora, which are hard to acquire. Therefore, at present, solving CCWS for specific ancient books or a specific period of time is the main trend (Shi, Li and Chen, 2010).

In terms of collocation, mutual information is the most used metric. As a concept of information theory, mutual information is used to measure the degree of association between two Chinese characters. The higher the mutual information is, the more related the two characters are. Sproat and Shih (1990) utilized mutual information to quantificationally describe how strongly associated of two arbitrary characters, found upon which bigrams are extracted automatically from raw corpus. Sun, Shen and Benjamin (1998) introduced the difference of t-score (dts) between Chinese characters, proposing an automatic segmentation algorithm based on mutual information and dts. In recent two decades, a number of scholars have focused on Chinese word extraction for solving CWS or CCWS by dint of various association metrics or a hybrid model of several metrics (Chang and Su, 1997; Chen and Ma, 2002; Luo and Sun, 2003; Ma and Chen, 2003; Feng et al., 2004; Tang et al., 2009; Zhang et al., 2009; Zhang et al., 2010; Duan, Han and Song, 2012; Zhang et al., 2012; Mei et al., 2015; Shen, Kawahara and Kurohashi, 2016). However, one big problem for collocation-based methods is that the performance of certain systems will heavily depend on the thresholds setting, because the thresholds for such association metrics are always set heuristically or empirically to gain high performance, which means we have to adjust thresholds with the change of applications or domains with extra effort. The approach we propose in this paper cuts the Gordian knot by integrating encyclopedia with traditional association metrics, with which there is no need setting threshold for deciding whether to discard a potential word or not.

2.3 Neural network-based approaches

With the rise of deep learning, neural models have been widely used for NLP tasks to avoid the task-specific feature engineering. Zheng, Chen and Xu (2013) performed CWS and POS tagging by adapting a general neural network architecture for sequence labeling. Pei, Ge and Chang (2014) improved upon Zheng's work by modeling the interactions between local context and previous tag. Chen et al. (2015a) proposed a gated recursive neural network, modeling the feature combinations of context characters. With the purpose of alleviating the limitation of the size of context window, Chen et al. (2015b) utilized a LSTM architecture to capture potential long-distance dependencies. Cai and Zhao (2016) proposed a novel neural framework which thoroughly eliminates context windows and can utilize complete segmentation history.

As far as we know, neural network-based methods have not been used to tackle CCWS yet. From our perspective, the diversity of Classical Chinese literature and the extreme lack of marked-up corpora account for it.

Fortunately, only with a certain size of raw corpus and a certain number of online encyclopedia documents, can we get acceptable result over CCWS. In Section 3 and Section 4, we will explain the corpus as well as other resources we use, and the concrete process how to segment texts with the lexicon we build in detail.

3. Corpus and resources

3.1 Raw corpus

We choose the *Twenty-Four Histories* and the *Draft History of Qing* as raw corpus, totally containing 3742 volumes and around 31 million characters. They are the Chinese official historical books, covering a period from 3000 BC to the end of Qing Dynasty (1912 AD), considered as one of the most important sources on Chinese history and culture. The books mostly record pertinent activities of emperors and politicians. Additionally, the content covers economy, politics, culture, art, astronomy, law, geography, science, technology and etc.

3.2 Evaluation data set

Owing to the fact that there is no standard datasets for CCWS and the performance of our approach can only be fully demonstrated by testing on literature of different eras, we randomly select a certain number (in proportion to the size of each book) of texts from each book and segment the

sentences manually. The test data contains 32689 characters in total, covering every aspect of the content.

3.3 Encyclopedia documents

10,143,321 documents were crawled from Baidu Baike[4] which is the largest Chinese-language online encyclopedia. For now, there are more than 15 million pages and more than 6 million people get involved in this project. The encyclopedia is a huge external knowledge resource, from which we can fetch concepts and entities. Like what we stated above, Classical Chinese is nearly obsolete, which means new content will not be generated. At the same time, the encyclopedia will gradually be complete in terms of knowledge about Classical Chinese literature. The title of each document can be considered as a word or a combination of words. That's why we attempt to integrate traditional statistical ways with Baidu Baike to extract words from Classical Chinese literature.

What we just need is the title of each entry. However, a dictionary with all the titles will cause low efficiency or memory problem. We remove some of them to shrink the volume. Firstly, the titles, which contain more than 8 or less than 2 characters, or contain non-Chinese character (like digits and punctuations), are gotten rid of. Then, we get 6,610,492 titles left. Secondly, remove those that contain high frequency (greater than 1000) prefix or suffix, some of which are listed in Table 1. The count of distinct affixes within corresponding frequency range is listed in Table 2. Finally, there are 4,324,035 distinct titles left.

Affixes	Frequency
酒店	68480
中国	31220
中学	30550
穿越	19361
社区	18696

Table 1: Frequency of some prefix or suffix

Frequency range	10-100	100-500	500-1000	>1000
Count	21701	8779	1540	992

Table 2: Count of prefix or suffix within a certain range

3.4 Official titles from CBDB

We extract Chinese ancient official titles from CBDB[5] (China Biographical Database Project), a project of Harvard University. As supplementary, 21152 distinct official titles that meet the requirements of the above preprocessing, are merged together with titles from Baidu Baike encyclopedia.

4. Methodology

Our approach mainly focuses on the creation of lexicon. Not only the vocabulary but also its corresponding frequency in the raw corpus are extracted. With such a lexicon, maximum probability, the commonly used algorithm for word segmentation, is applied to CCWS in our method.

4.1 Lexicon creation

The general procedure is illustrated in Figure 1. From the first iteration over texts in the corpus, frequency of each Chinese character bigram is recorded. In the light of Aho-Corasick Automaton, a temporary dictionary is built from the titles extracted from Baidu Baike and CBDB, with which we can rapidly get the location of each word in the sentence. Before explaining the following procedures, we will give the definitions of t-score and dts first.

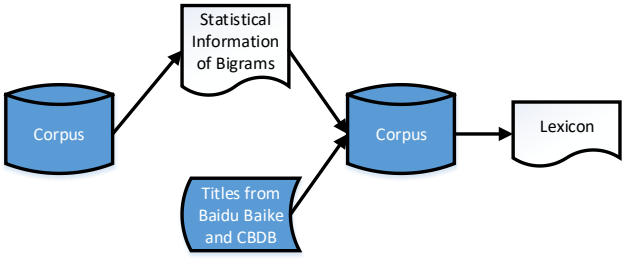


Figure 1: Flow diagram of lexicon creation.

Given a Chinese character string 'xyz', the t-score of the character y relevant to x and z is defined as:

$$ts_{x,z}(\mathrm{y}) = \frac{p(z|y) - p(y|x)}{\sqrt{var\big(p(z|y)\big) + var(p(y|x))}}$$

Where p(y|x) is the conditional probability of y given x, and p(z|y), of z given y, and var(p(y|x)), var(p(z|y)) are variances of p(y|x) and of p(z|y) respectively.

Given a Chinese character string 'vxyw', the dts between characters x and y is defined as:

$$\mathrm{dts}(\mathrm{x{:}\,y}) = ts_{v,y}(x) - ts_{x,w}(y)$$

If dts(x:y) > 0 then it tends to be bound. If dts(x:y) < 0 then it tends to be separated. More details about t-score and dts, please refers to the work of Sun et al. (1998) and Church et al. (1991).

The main processes of the second iteration are as follows:

Step 1: Segment one sentence into sub-sentences (sub-sentence here refers to a Chinese character string without any punctuation). Then, iterate over the sub-sentences.

Step 2: Calculate dts of every bigram of the sub-sentence.

Step 3: Look up the dictionary and list all words of this sub-sentence.

Step 4: Skip over bigram-words. For words that contain more than 3 characters, add them to lexicon directly (if the word exists, just increase corresponding frequency , similarly hereinafter). For words that exactly contain 3 characters, we need to consider about three situations. If the last character of previous word is same as first character of current word and next word (bigram-word) is same as the last two characters of current word, add previous word and next word to lexicon; if the previous word is same as the first two characters of current word and the first character of next word is same as last character of current word, add previous word and next word to lexicon; otherwise, add current word to lexicon. Overlap is not allowed in this step, which means each character can only belong to a specific word.

Step 5: Find out the position with largest dts among the left characters of the sub-sentence. Take the position as center, search bidirectionally, and bind characters together if the dts between two characters is greater than zero. Keep searching until the dts is smaller than zero or no character left. Then, we get a candidate word. If this word contains

[4] https://baike.baidu.com/

[5] https://projects.iq.harvard.edu/cbdb

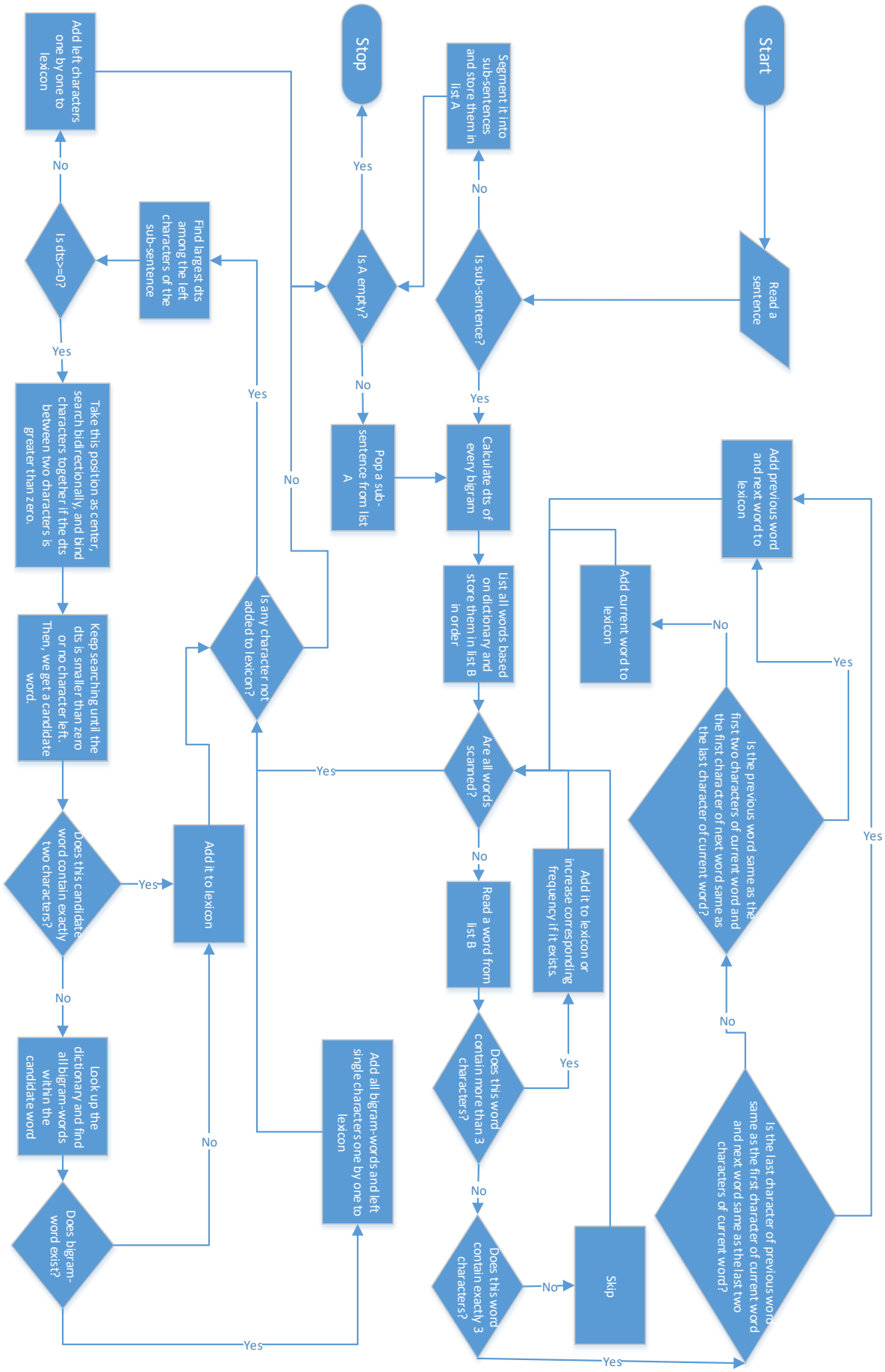


Figure 2: Flow diagram of second iteration.

exactly two characters, add it to lexicon. Otherwise, look up the dictionary and find all bigram-words within the candidate word. If no bigram-word exists, add the candidate word to lexicon; otherwise, add all bigram-words to lexicon, and if some of characters of the candidate word are not added to lexicon, add left characters to lexicon one by one (take single character as a word). Overlap is allowed in this step. For example, $C_1C_2C_3C_4C_5$ is a candidate word. C_1C_2 and C_2C_3 are bigram-words contained in the dictionary. Eventually, C_1C_2, C_2C_3, C_4 and C_5 will be added to lexicon.

Step 6: Repeat step 5 until the largest dts is smaller than zero or no character left.

Step 7: If there are characters left, they are added one by one to lexicon. For instance, $C_1C_2C_3C_4C_5C_6C_7C_8C_9C_{10}$ is a sub-sentence and it gets processed from step 1 to step 6. C_3, C_4 and C_{10} are left characters. Then, C_3, C_4 and C_{10} are added to lexicon.

Step 8: Repeat step 1 to step 7 until all texts in corpus get processed.

To make the process clear, the corresponding flow diagram is given in Figure 2. In step 4, we skip over bigram-words in dictionary so as to avoid introducing more ambiguity. For a Chinese character string $C_1C_2C_3$, both C_1C_2 and C_2C_3 can be found in the dictionary. Under such a circumstance, we have less chance to make a right choice with the dictionary alone. For the predominance of bigram-words, words containing more than two characters, which have less ambiguity, can be added to lexicon directly within certain limitation. With statistical information, bigram-words can be extracted in step 5 to guarantee a certain accuracy. The reason we make such rules in step 5 will be further explained in Section 5.4. Eventually, we build a specific lexicon for CCWS from the *Twenty-Five Histories*.

4.2 Word segmentation

Now that there is a customized lexicon with frequency of each word, maximum probability algorithm is used for word segmentation. w_i stands for a certain word and S stands for a sentence that contains n words. Ignore the relevance between words, and the probability of sentence S is defined as:

$$\mathrm{P}(S) = P(w_1) \times P(w_2) \times \cdots \times P(w_n)$$

Take the case of maximum value of P(S) as the optimal result of word segmentation. The probability of a certain word is defined as:

$$\mathrm{P}(w_i) = \frac{f(w_i)}{N}$$

$f(w_i)$ refers to the frequency of word w_i, and N is the total words of the corpus. Dynamic programming is applied for reducing the computation.

5. Experiments

We have done five comparison experiments to prove the effectiveness of our approach. Out of the fact that Classical Chinese literature is mostly composed of monosyllables, sentences segmented into individual characters are considered as the baseline of CCWS in our experiments. Before illustrating experiment results, the design of other three experiments will be described below.

5.1 Normalized Pointwise Mutual Information

Bouma (2009) introduced the normalized pointwise mutual information (NPMI) for collocation extraction. For character C_1 and C_2, the NPMI of them is defined as:

$$\mathrm{npmi}(C_1; C_2) = \frac{pmi(C_1; C_2)}{h(C_1; C_2)}$$

PMI of C_1 and C_2 is defined as:

$$\mathrm{pmi}(C_1; C_2) = \log \frac{p(C_1, C_2)}{p(C_1)p(C_2)}$$

$h(C_1;C_2)$ is the self-information, defined as:

$$\mathrm{h}(C_1; C_2) = -\log p(C_1, C_2)$$

NPMI ranges from -1 to 1, resulting in -1 for never occurring together, 0 for independence, and 1 for complete co-occurrence. In this experiment, if the NPMI between two characters is greater than zero, they are bound together. During the word extraction, do a scan over the sentence; characters bound together are added as a word to lexicon; characters left are added one by one to lexicon. The lexicon we get is named after 'lexicon1'.

5.2 Difference of t-score

The definition of dts is given in Section 4. In this experiment, if the dts between two characters is greater than zero, they are bound together. The process of word extraction is the same as that of previous experiment. The lexicon is named after 'lexicon2'.

5.3 Simple integration of dts with encyclopedia

In this experiment, the procedures are all the same as those of our approach described in Section 4, except the step 5. The candidate words are added straight to lexicon without the subsequent operations. The lexicon is named after 'lexicon3'. For 'lexicon3', we have counted words that contain more than two characters and are not included in the dictionary, as listed in Table 3 below.

	3-gram	4-gram	5-gram	6-gram	7-gram
Count	28064	2215	35	1	0

Table 3: Count of N-gram

3-grams	4-grams	5-grams	6-grams
照英约	烧其船舰	位于大明殿	周孝闵帝践祚
尝与刘	尽其死力	衣画而裳绣	
尚宫曰	物之失所	降者数万人	
少敏慧	寻以母丧	积度及分秒	
尽其长	辰星犯天	为酷吏所陷	
将安出	填星犯井	加上尊号曰	
先帝崩	结为死党	莫大于不孝	
赤山湖	虽悔何追	德厚者流光	
⋮	⋮	⋮	
⋮	⋮	⋮	

Table 4: Some samples of N-gram.

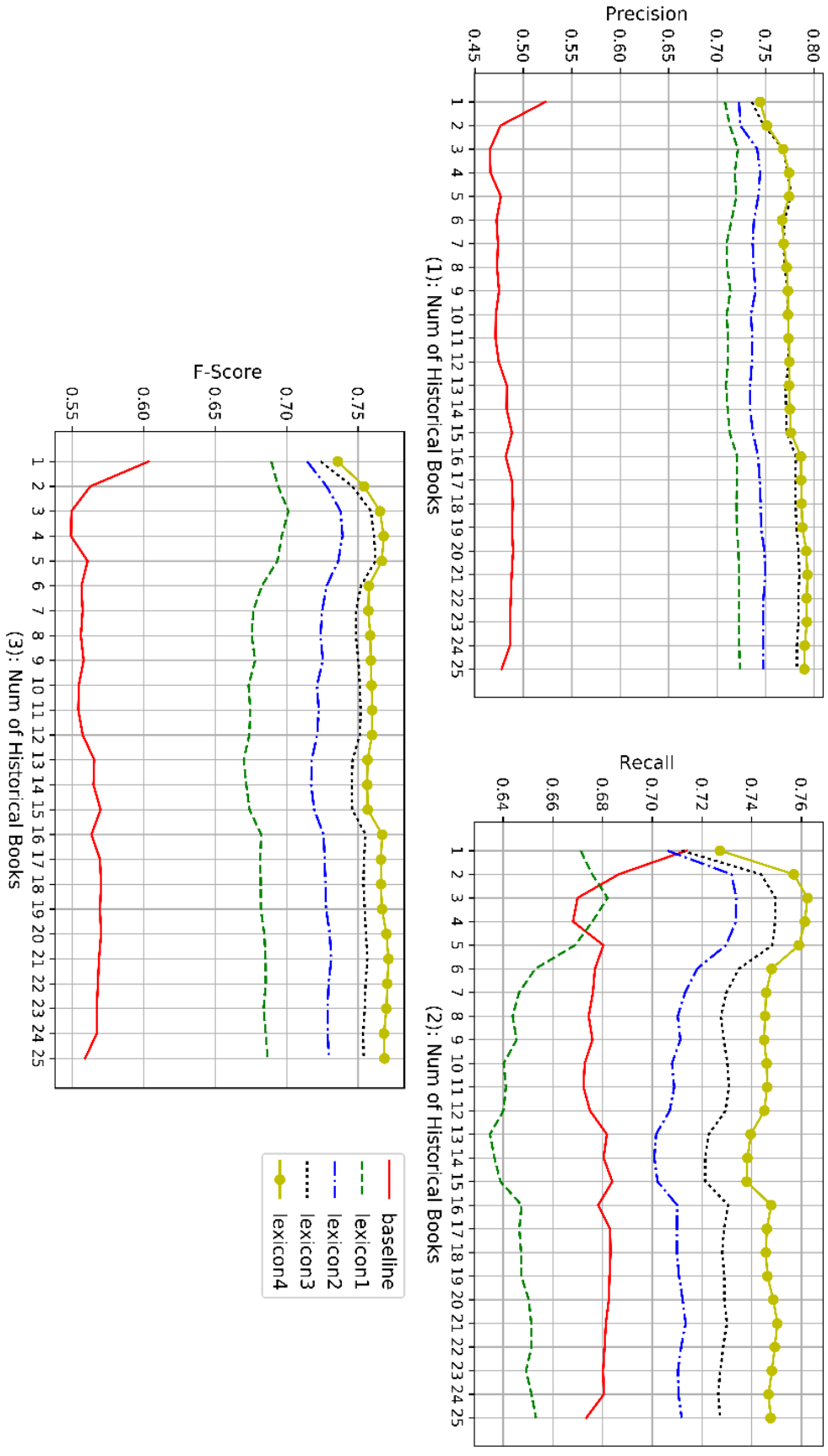


Figure 3: Precision, recall and F-Score of different approaches

Through our observation, a majority of these N-grams are frequent items instead of words, some of which are listed in Table 4. In order to enhance the validity of the lexicon, frequent items need to be processed ulteriorly. With further division of frequent items in our final solution, we get a lexicon named after 'lexicon4'.

5.4 Experiment results

There are 20 commonly used function words (those have little lexical meaning and express grammatical relationships with other words within a sentence) in Classical Chinese literature, as listed in Table 5. The function words in the four lexicons are given the highest weight to ensure they can be correctly segmented out of the sentence in most occasions.

而	何	乎	乃	其	且	然	若	所	与
为	焉	也	以	矣	于	之	则	者	因

Table 5: Commonly used function words in ancient texts

As stated in Section 3.2, we randomly select sample texts, consisting of 32689 characters, from the *Twenty-Five Histories*. The size of samples picked out from each historical book is proportional to the size of the book for keeping the balance. In addition, the sample texts cover every aspect of the content. Figure 3 illustrates the precision, recall, as well as F-Score of five different approaches. The x-axis of the three subgraphs stands for the number of books used in experiments. For instance, "4" means we used the first 4 historical books that were written in chronological order (*Records of the Grand Historian*, *Book of Han*, *Book of Later Han*, *Records of the Three Kingdoms*) to measure the performance. As for the reason that the performance is showed in accumulation, it can prove the stability and feasibility of our approach over literature of different eras (There are unique words and expressions in each dynasty, some of which only appear within a certain period of time). In other words, it's similar to one-size-fits-all approach, with which we don't have to create lexicon for each book.

Judging from the F-Score subgraph, it's obvious that our approach surpasses the others, and it can keep a high stable F-Score with the increment of test data. Over the whole test data, the F-Score of our approach is 76.84%. Recall subgraph shows that the drawback of NPMI results in the drop of the recall rate of 'lexicon1'. Without an explicit threshold, it's not possible to extract high quality multisyllabic words with NPMI. Besides, the conspicuous drop of recall rate of baseline method verifies the phenomenon that individual character is mostly used to represent a monosyllabic word in Archaic Chinese and the number of multi-syllables increases gradually as time goes on.

An example of CCWS over a specific sentence is listed in Table 6. Approach with 'lexicon3' or 'lexicon4' is able to identify the word '勃然作色' correctly for the combination with Baidu Baike. The word is a Chinese idiom of which the definition is included in Baidu Baike. However, with the statistical information alone, this 4-gram word is not extracted out of the corpus. As for the trigrams '太息曰' and '虽不肖', which are not included in Baidu Baike, only method with 'lexicon4' correctly segments them into two words respectively. If the trigram is not in Baidu Baike while a bigram, being part of it, is included, we tend to believe that the trigram is composed of a bigram-word as well as an individual character (correct segmentation is '太息|曰' and '虽|不肖'). For low precision of bigram-word extraction, our method wrongly identifies bigram-words '事秦' and '今主' in the instance. In step 5 of our approach, we add overlapped distinct bigram-words to the lexicon, which results in the deficiency in precise extraction of bigram-word. We have attempted to pick only one out of two overlapped bigram-words according to their dts, but the final result is not improved. Increasing the accuracy of bigram-word extraction is our future work.

Original	于是韩王勃然作色，攘臂瞋目，按剑仰天太息曰："寡人虽不肖，必不能事秦。今主君诏以赵王之教，敬奉社稷以从。"
Standard	于是|韩王|勃然作色|，|攘臂瞋目|，|按剑|仰天|太息|曰|：|"|寡人|虽|不肖|，|必|不能|事|秦|。|今|主君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|"
Lexicon1	于|是|韩|王勃|然|作色|，|攘|臂|瞋目|，|按剑|仰天|太息曰|：|"|寡人|虽不肖|，|必不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|"
Lexicon2	于是|韩王|勃|然|作色|，|攘臂|瞋目|，|按剑|仰天|太息曰|：|"|寡人|虽不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|"
Lexicon3	于是|韩王|勃然作色|，|攘臂|瞋目|，|按剑|仰天|太息曰|：|"|寡人|虽不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|"
Lexicon4	于是|韩王|勃然作色|，|攘臂|瞋目|，|按剑|仰天|太息|曰|：|"|寡人|虽|不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|"

Table 6: CCWS example

6. Conclusion

In this paper we proposed a pragmatic approach combining the difference of t-score and encyclopedia (Baidu Baike) for CCWS over literature of different eras. The F-Score over the whole evaluation data set is 76.84%. To a certain degree, this approach releases researchers from labor intensive work, like constructing corpus, and makes it possible to build standard large-scale shareable marked-up corpora for the study of Classical Chinese literature. It also facilitates the research of Classical Chinese literature throughout the whole history instead of over a specific period of time or particular books. Besides, the scale of Baidu Baike we used in experiments is just two-thirds of the latest version, which means the performance of our approach can be better. With the Baidu Baike gradually being complete in terms of knowledge about Classical Chinese literature, the performance will be highly improved. This is another merit of our approach. After all, new content won't be generated for the obsoleteness of Classical Chinese.

7. Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61070066, No. 61379073), and the CADAL (China Academic Digital Associative Library) Project and Research Center, Zhejiang University.

8. Bibliographical References

Bouma, G. (2009). Normalized (Pointwise) Mutual Information in Collocation Extraction. *Proceedings of GSCL*, pp. 31--40.

Cai, D. and Zhao, H. (2016). Neural Word Segmentation Learning for Chinese. *Association for Computational Linguistics*, pp. 409--420.

Chang, J.S. and Su. K.Y. (1997). An Unsupervised Iterative Method for Chinese New Lexicon Extraction. *International Journal of Computational Linguistics & Chinese Language Processing*, 2(2):97-147.

Chen, K.J. and Ma, W.Y. (2002). Unknown Word Extraction for Chinese Documents. *COLING '02 Proceedings of the 19th international conference on Computational linguistics*, (1):1-7.

Chen, X.C., Qiu, X.P., Zhu, C.X. and Huang, X.J. (2015a). Gated Recursive Neural Network for Chinese Word Segmentation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 1744--1753.

Chen, X.C., Qiu, X.P., Zhu, C.X., Liu, P.F. and Huang, X.J. (2015b). Long Short-Term Memory Neural Networks for Chinese Word Segmentation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1197--1206.

Church, K.W., Gale, W.A., Hanks, P. and Hindle, D. (1991). Using Statistics in Lexical Analysis. *Lexical Acquisition: Exploiting Online Resources to Build a Lexicon, Publisher: Lawrence Erlbaum, Editors: Uri Zernik*, pp. 115--164.

Duan, L., Han, F. and Song, J.H. (2012). A Comparative Study on the Automatic Extraction of Two-character Word from Ancient Chinese. *Journal of Chinese Information Processing*, 26(04):34-42.

Fang, A.C., Lo, F. and Chinn, C.K. (2009). Adapting NLP and corpus analysis techniques to structured imagery analysis in classical Chinese poetry. *The Workshop on Adaptation of Language Resources and Technology To New Domains Association for Computational Linguistics*, pp. 27--34.

Feng, H.D., Chen, K., Deng, X.T. and Zheng, W.M. (2004). Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, 30(1):75-93.

Lee, J. and Wong, T.S. (2013). Glimpses of Ancient China from Classical Chinese Poems. *COLING 2012: Posters*, pp. 621--632.

Li, B., Xi, N., Feng, M. and Chen, X. (2012). Corpus-Based Statistics of Pre-Qin Chinese. *Chinese Conference on Chinese Lexical Semantics Springer-Verlag*, (7717):145-153

Liu, C.L., Wang, H., Cheng, W.H., Hsu, C.T. and Chiu, W.Y. (2015). Color Aesthetics and Social Networks in Complete Tang Poems: Explorations and Discoveries. *Computer Science*.

Luo, S.F. and Sun, M.S. (2003). Chinese Word Extraction Based on the Internal Associative Strength of Character Strings. *Journal of Chinese Information Processing*, 17(03):10-15.

Ma, W.Y. and Chen, K.J. (2003). A bottom-up merging algorithm for Chinese unknown word extraction. *SIGHAN '03 Proceedings of the second SIGHAN workshop on Chinese language processing*, (17):31-38.

Mei, L.L., Huang, H.Y., Wei, X.C., Yuan, P. and Mao, X.L. (2015). FCL: A New Network Words Extraction Approach Based on Statistical Language Knowledge. *Social Media Processing, Springer Singapore*.

Ouyang, J. (2016). Visual Analysis and Exploration of Ancient Texts for Digital Humanities Research. *Journal of Library Science in China*, (02):66-80.

Pei, W.Z., Ge, T. and Chang B.B. (2014). Max-Margin Tensor Neural Network for Chinese Word Segmentation. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 293--303.

Peng, F.C., Feng, F.F. and McCallum, A. (2004). Chinese segmentation and new word detection using conditional random fields. *In Proceedings of COLING*, pp. 562--568.

Qiu, B., Huang, P.J. (2008). Study on the Trend of Ancient Chinese Words Based on the Word Automatic Segmentation. *Microcomputer Information*, 24(24):100-102.

Shen, M., Kawahara, D. and Kurohashi, S. (2016). Chinese Word Segmentation and Unknown Word Extraction by Mining Maximized Substring. *Journal of Natural Language Processing*, 23(3):235-266.

Shi, M., Li, B. and Chen, X.H. (2010). CRF Based Research on a Unified Approach to Word Segmentation and POS Tagging for Pre-Qin Chinese. *Journal of Chinese Information Processing*，24(2):39-45.

Sproat, R. and Shih, C. (1990). A Statistical Method for Finding Word Boundaries in Chinese Text. *Computer Processing of Chinese & Oriental Languages*, 4(4):336-351.

Sun, M.S., Shen, D.Y. and Benjamin, K.T. (1998). Chinese Word Segmentation without Using Lexicon and Hand-crafted Training Data. *Meeting of the Association for Computational Linguistics and, International Conference on Computational Linguistics Association for Computational Linguistics*, 48(2):1265-1271.

Tang, L.X., Geva, S., X, Y. and Trotman, A. (2009). Word Segmentation for Chinese Wikipedia Using N-Gram Mutual Information. *Plos Medicine*, 2(7):576-582.

Tseng, H.H., Chang, P.C, Andrew, G., Jurafsky, D. and Manning, C. (2005). A conditional random field word segmenter for sighan bakeoff 2005. *In proceedings of the fourth SIGHAN workshop*, pp. 168--171.

Xu, Y.M. (2016). Some Visualization Approaches to the Study of Classical Chinese Literature: A Case Study on Tang Xianzu. *Journal of Zhejiang University (Humanities and Social Sciences Online Edition)*.

Xue, N.W. (2003). Chinese word segmentation as character tagging. *International Journal of Computational Linguistics & Chinese Language Processing*, 8(1):29-47.

Zhang, C.X., Niu, Z.D., Jiang, P. and Fu, H.P. (2012). Domain-specific term extraction from free texts. *International Conference on Fuzzy Systems and Knowledge Discovery IEEE*, pp. 1290--1293.

Zhang, H.J., Huang, H.Y., Zhu, C.Y. and Shi, S.M. (2010). A Pragmatic Model for New Chinese Word Extraction. *International Conference on Natural Language Processing and Knowledge Engineering IEEE*, pp. 1--8.

Zhang, W., Yoshida, T., Tang, X.J and Ho, T.B. (2009). Improving effectiveness of mutual information for substantival multiword expression extraction. *Expert Systems with Applications*, 36(8):10919-10930.

Zhang, Y. and Clark, S. (2007). Chinese Segmentation with a Word-Based Perceptron Algorithm. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*.

Zheng, X.Q., Chen, H.Y. and Xu, T.Y. (2013). Deep learning for Chinese word segmentation and POS tagging. *Conference on Empirical Methods in Natural Language Processing*, pp. 647--657.

ASAP++: Enriching the ASAP Automated Essay Grading Dataset with Essay Attribute Scores

Sandeep Mathias, Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology, Bombay
{sam, pb}@cse.iitb.ac.in

Abstract

In this paper, we describe the creation of a resource - ASAP++ - which is basically annotations of the Automatic Student Assessment Prize's Automatic Essay Grading dataset. These annotations are scores for different attributes of the essays, such as content, word choice, organization, sentence fluency, *etc*. Each of these essays is scored by an annotator. We also report the results of each of the attributes using a Random Forest Classifier using a baseline set of attribute independent features as described by Zesch et al. (2015). We release and share this resource to facilitate further research into these attributes of essay grading.

Keywords: Automatic Essay Grading, Attribute-specific Essay Grading

1. Introduction

Automatic essay grading (AEG) is one of the most challenging activities in natural language processing (NLP). AEG makes use of many NLP and machine learning (ML) techniques in predicting the score of an essay - a piece of text that is written by a human on a given topic (called a prompt). It has been around since the 1960s, with the first AEG system - Project Essay Grade - proposed by Ellis Page (Page, 1966). Since then, there have been multiple systems that look at providing either a holistic score to the essay, or to score individual attributes of the essay. Examples of a few online systems include Grammarly[1] and Paper-Rater[2].

Essay grading systems rely on training and test data in order to grade essays. Most of the research today makes use of the Automated Student Assessment Prize's (ASAP) Automated Essay Grading (AEG) dataset to train and test systems[3]. The ASAP AEG dataset comprises of approximately 13,000 essays, written across 8 prompts. The essays were written by students of class 7 to 10. Each essay was evaluated by 2 evaluators. 6 out of the 8 prompts only have overall scores. Only 2 of them have scores for individual essay attributes, like content, organization, style, *etc*.

Our **contribution** is the scoring of individual attributes of the essays, like content, organization, style, etc. in the ASAP dataset for the remainder of the essays.

2. Motivation

A lot of the work in essay grading today makes use of the ASAP AEG dataset. However, most of the essays only have an overall score, not attribute-specific scores. This limitation limits the utility of this dataset for predicting the scores of particular attributes of essays.

Shermis and Burstein (2013) in chapter 19 (Contrasting State-of-the-Art Automatic Essay Grading Systems) of their book describe the ASAP dataset, as well as the results of current commercial AEG systems in scoring those essays. Since then, a large amount of work has been done using that dataset for evaluating the overall score of essays, from using machine learning techniques (Chen and He, 2013; Phandi et al., 2015) to deep learning systems (Dong et al., 2017; Dong and Zhang, 2016; Alikaniotis et al., 2016; Taghipour and Ng, 2016).

One common feature that all the above work has in common is the fact that the essay grading dataset that they used was the ASAP AEG dataset. However, most of them (in particular the deep learning systems) are constrained by the fact that there are very few prompts to handle scoring of individual attributes.

3. Related Work

While there has been a lot of work done in overall essay scoring, not much has been done with respect to scoring particular attributes of essays. Some of the attributes that have been scored include organization (Persing et al., 2010), prompt adherence (Persing and Ng, 2014), coherence (Somasundaran et al., 2014).

4. Dataset

The entire ASAP dataset has nearly 13,000 essays across 8 prompts. 6 of those 8 prompts, comprising nearly 10,400 essays, only have an overall score.

4.1. Essay Topics

The following is the list of topics of the 8 prompts in the dataset:

1. **Prompt 1** - The writers had to write a letter to their local newspaper in which they stated their opinion on the effects computers have on people.

2. **Prompt 2** - The writers had to write a persuasive essay reflecting their views on censorship in libraries.

3. **Prompt 3** - The writers had to read an extract from *Rough Road Ahead: Do Not Exceed Posted Speed Limit* by Joe Kurmaskie. They then had to explain how the features of the setting affected the cyclist.

[1] www.grammarly.com

[2] www.paperrater.com

[3] The dataset can be downloaded from here: https://www.kaggle.com/c/asap-aes/data.

Prompt ID	Essay Type	No. of Essays	Avg. Length	Score Range	Attribute Scores Contributed
Prompt 1	Argumentative	1785	350	1 - 6	**Yes**
Prompt 2	Argumentative	1800	350	1 - 6	**Yes**
Prompt 3	Source-Dependent	1726	150	0 - 3	**Yes**
Prompt 4	Source-Dependent	1772	150	0 - 3	**Yes**
Prompt 5	Source-Dependent	1805	150	0 - 4	**Yes**
Prompt 6	Source-Dependent	1800	150	0 - 4	**Yes**
Prompt 7	Narrative	1569	300	0 - 3	**No**
Prompt 8	Narrative	723	650	1 - 6	**No**

Table 1: Description of the ASAP AEG dataset. The Avg. Length column gives the average length of the essay, in terms of number of words. The score range column lists the scoring range of the various attributes that we score. We use the same score range as the overall score range of the essays. The last column tells us the prompts whose attribute scores we contribute. All the essays were written by native English speaking children from classes 7 to 10.

Feature Type	Feature List
Length	Word Count, Sentence Count, Sentence Length, Word Length
Punctuation	Counts of Commas, Quotations, Apostrophes, etc.
Syntax	Parse Tree Depth, Subordinate Clauses, etc.
Stylistic Features	Formality, Word Frequency, Type-Token Ratio
Cohesion Features	Discourse Connectives, Entity Grid, etc.
Coherence Features	Average Similarity between adjacent sentences of PoS tags, Lemmas, etc.
Language Model Features	Count of OOVs, LM score, etc.
n-Gram Features	Word n-Grams and PoS n-Grams

Table 2: Different features used in our experiment

4. **Prompt 4** - The writers had to read an extract from *Winter Hibiscus* by Minfong Ho. They then had to explain why the author concludes the story in the way that she did.

5. **Prompt 5** - The writers had to read an extract from *Narciso Rodriguez* by Narciso Rodriguez. They then had to describe the mood created by the author with supporting evidence from the extract.

6. **Prompt 6** - The writers had to read an extract from *The Mooring Mast* by Marcia Amidon Lusted. They then had to answer a question about the difficulties faced by the builders of the Empire State Building in allowing dirigibles to dock there.

7. **Prompt 7** - Write a story about a time when you, or someone you know, was patient.

8. **Prompt 8** - Write a story in which laughter plays a part.

Table 1 gives a description of the different essay prompts. Since scores are already present for prompts 7 & 8, we mainly provide scores for prompts 1 to 6.

4.2. Types of Essays

There are 3 types of essays in the dataset.

1. **Argumentative / Persuasive essays** - These are essays where the prompt is one in which the writer has to convince the reader about their stance for or against a topic (for example, free speech in public colleges).

2. **Source-dependent responses** - These essays are responses to a source text, where the writer responds to a question about the text (for instance, describing the writer's opinion about an incident that happened to him in the text).

3. **Narrative / Descriptive essays** - These are essays where the prompt requires us to describe / narrate a story.

Based on the type of the essay, we have a different set of attributes for evaluation. The ASAP dataset already contains attribute scores for the narrative essays, namely content, organization, word choice, sentence fluency, conventions, *etc*. Since we already have scores present for the narrative essays, we describe the scores for the other types of essays.

4.3. Attributes of Essays

Based on the types of essays, there are 2 sets of attributes[4].

4.3.1. Attributes of Argumentative / Persuasive Essays

There are 5 attributes for narrative essays, namely

1. **Content**: The quantity of relevant text present in the essay.

2. **Organization**: The way the essay is structured.

3. **Word Choice**: The choice and aptness of the vocabulary used in the essay.

[4]Details of the attributes and their scoring are also shared as a part of the resource.

Prompt ID	Content	Organization	Word Choice	Sentence Fluency	Conventions	Overall
Prompt 1	0.67	0.60	0.64	0.62	0.61	0.74
Prompt 2	0.61	0.58	0.60	0.59	0.62	0.62

Table 3: Results of the 5-fold cross-validation using the Random Forest classifier for argumentative / persuasive essays.

Prompt ID	Content	Prompt Adherence	Language	Narrativity	Overall
Prompt 3	0.59	0.59	0.57	0.63	0.54
Prompt 4	0.66	0.66	0.56	0.67	0.68
Prompt 5	0.67	0.64	0.60	0.63	0.76
Prompt 6	0.60	0.56	0.59	0.62	0.63

Table 4: Results of the 5-fold cross-validation results using the Random Forest classifier for source-dependent essays.

4. **Sentence Fluency**: The quality of the sentences in the essay.

5. **Conventions**: Overall writing conventions to be followed, like spelling, punctuations, etc.

4.3.2. Attributes of Source-dependent Responses

There are 4 attributes for source-dependent responses, namely

1. **Content**: The amount of relevant text present in the essay.

2. **Prompt Adherence**: A measure of how the writer sticks to the question asked in the prompt.

3. **Language**: The quality of the grammar and spelling in the response.

4. **Narrativity**: A measure of the coherence and cohesion of the response to the prompt.

We consider organization as an important attribute for the argumentative essays mainly because the average length of those essays is far more that of the source-dependent responses. The argumentative essays also have the scope for a wider vocabulary compared to the source dependent essays. Hence, we use word choice and organization as useful attributes for the argumentative / persuasive essays.

On the other hand, the source-dependent responses are constrained to respond to the source text. Hence, we have attributes like prompt adherence here, rather than word choice. The sentence fluency and conventions attributes are present in the language attribute of the source-dependent responses. The narrativity attribute attempts to ensure that the response is well-connected and makes sense. Hence, it is similar to organization, except that the organization attribute of the argumentative essays also requires that the essay have a good structure, like *introduction* $\rightarrow$ *body* $\rightarrow$ *conclusion*, while the source-dependent response would just be the *body*[5].

5. Creation of the Dataset

Each of the essays in a particular prompt were scored by an annotator. Each prompt was split into sets of 100 essays each, with the assumption that a set would correspond to a week's worth of time for the annotator. Thus, each prompt had a total of 18 sets[6].

Unlike the ASAP AEG dataset in which every essay was annotated by 2 annotators, we use only 1 annotator here for each essay. For the ground truth, we make use of the overall score of the essays given by the original annotators of the ASAP AEG dataset. In case the scoring of a particular attribute for a particular prompt differs from either of the original scorers by 2 or more points, the essay is then annotated by another annotator. The final score that is chosen is the one from the annotator that is closest to the overall scores. One of the reasons that we do this is because, in the 2 prompts that were rated by the original raters, there is a very high Pearson correlation (nearly 0.9) between the overall scores and the individual attribute scores.

5.1. Annotator Details

We made use of a total of three annotators to annotate the essays. Each of the annotators had competence in English, either by scoring quite high marks in their high school exams (over 90% in English), or scoring over 110 in ToEFL. Each of them also had some experience in evaluating texts, such as interning at *The Hindu*[7] (a top English newspaper in India), being the chief editor of the college magazine, *etc.* All the annotators have either studied or are studying English at a Master of Arts (MA) level.

6. Experiments

6.1. Features Used

After creating the resource, we ran experiments to get some baseline results. We used the attribute independent feature set provided by Zesch et al. (2015). In addition to those features, we also made use of entity grid features described in Barzilay and Lapata (2005). Table 2 summarizes the list of features that we used in our experiments. All the features were extracted using Stanford Core NLP (Manning et al., 2014).

[5]Prompt #6, for instance, requires the writer to enumerate some of the difficulties faced by the builders of the Empire State Building in docking dirigibles

[6]Prompt 5, with a total of 1805 essays had 105 essays in its last set.

[7]http://www.thehindu.com/

Prompt ID	Content	Organization	Word Choice	Sentence Fluency	Conventions
Prompt 1	Coherence	Length	Coherence	Syntax	Coherence
Prompt 2	Coherence	Coherence	Coherence	Syntax	Coherence
Average	Coherence	Coherence	Coherence	Syntax	Coherence

Table 5: Results of the ablation tests using the Random Forest classifier for argumentative / persuasive essays to determine the most important feature set for each attribute in each prompt.

Prompt ID	Content	Prompt Adherence	Language	Narrativity
Prompt 3	Length	Coherence	Coherence	Style
Prompt 4	Punctuation	Language Model	Coherence	Complexity
Prompt 5	Length	Coherence	Punctuation	Language Model
Prompt 6	Coherence	Language Model	Coherence	Coherence
Average	Length	Coherence	Coherence	Coherence

Table 6: Results of the ablation tests using the Random Forest classifier for source-dependent responses to determine the most important feature set for each attribute in each prompt

6.2. Evaluation Metric

We evaluate each of the annotators using Cohen's Kappa, with quadratic weights - i.e. Quadratic Weighted Kappa (QWK) (Cohen, 1968).

We chose this as the evaluation metric (as compared to accuracy and weighted F-Score) because of the following reasons:

- Unlike accuracy and F-Score, Kappa takes into account random agreement. For example, a majority class classification will result in a Kappa of 0, while accuracy and F-Score will be the percentage of the majority class in the test set.
- Weighted Kappa, takes into account the distance between the actual score and the reported score. Quadratic weights reward matches and penalize mismatches more than linear weights.

Due to these reasons, this is one of the most used evaluation metrics to evaluate the performance of essay grading systems. To the best of our knowledge, all of the papers using the ASAP dataset make use of this as the evaluation metric.

6.3. Classifier Used

We made use of the Ordinal Class Classifier (Frank and Hall, 2001) in Weka (Frank et al., 2016). This classifier is a meta-classifier, that first converts ordinal data into categorical data, before running an internal classifier on the data. We used the Random Forest classifier (Breiman, 2001) as the internal classifier. We used 5-fold cross validation to get the results for each attribute for each prompt. We report the results in Table 3.

7. Results and Analysis

Tables 3 and 4 shows the Quadratic Weighted Kappa (QWK) with respect to predicting the score using the Random Forest Classifier. Since the feature set was designed specifically for the overall score of the essays, it is expected that the overall score usually has the best result (this is true with the exception of Prompt #3).

Most of the essays required only a single annotator. Only about a sixth of the essays required a second annotator.

One of the major problems that the annotators faced was the fact that all the essays were anonymized. Named entities, like *The New York Times* would be referred to as *@ORGANIZATION1*, *Donald Trump* would be referred to as *@PERSON1*, *etc*. The annotators were instructed not to penalize the essays because of the anonymizations, but were told to replace them with placeholders (like *@PERSON1* being replaced by either *Joe*, or *Jane*, *etc.* wherever applicable).

7.1. Ablation Tests

In order to see which feature sets are important for each of the attributes, we conducted ablation tests on each of the feature sets for each of the attributes. Tables 5 and 6 show which features are important for which attribute and which type of essay.

For source-dependent essays, we found out that the most important feature for content was length, while for argumentative / persuasive essays, it was coherence and cohesion features, followed by length. This is mainly because source-dependent essays are highly dependent on the source text, while argumentative / persuasive essays can utilize arguments from beyond the scope of any text, and so, those arguments have to be coherent and cohesive.

For source-dependent essays, the coherence and cohesion feature set is the most important feature set for each of the other 3 attributes. While narrativity is a measure of the coherence and cohesion of the text (and hence, it is self-evident that these features would be the most important for scoring this attribute), the language and prompt adherence scores also happen to be affected by this. This is mainly because source-dependent responses should adhere to the prompt that they are written as a response to.

For persuasive / argumentative essays, coherence and cohesion features are the most important features for 4 of the 5 attributes. This is mainly because coherence and cohesion are important for organization of the argument. The syntactic features though, were found to be the most important features for sentence fluency, since they measure how well

written the individual sentences of the essay are.
In fact, overall, the most important feature set is the coherence and cohesion features.

8. Conclusion

In this paper, we present a manually annotated dataset for automated essay grading. The annotation was done for different attributes of the essays. Most of the essays were annotated by a single annotator. However, about a sixth of them were annotated by a second annotator. These annotations can be used as a gold standard for future experiments in predicting different attribute scores.
The resource is available online at `https://cfilt.iitb.ac.in/~egdata/`. The resource is available for non-commercial research use under the Creative Commons Attribution-ShareAlike License[8].

9. Acknowledgements

We thank the annotators of our task - Advaith Jayakumar, Janice Pereira and Elaine Mathias - for their help in creating this resource. We also thank members of CFILT at IIT Bombay for their valuable comments and suggestions. We also acknowledge the support of various funding agencies, like the Department of Information Technology (DIT), Ministry of Human Resource Development (MHRD), etc. of the Government of India.

10. Bibliographical References

Alikaniotis, D., Yannakoudakis, H., and Rei, M. (2016). Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–725, Berlin, Germany, August. Association for Computational Linguistics.

Barzilay, R. and Lapata, M. (2005). Modeling local coherence: An entity-based approach. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 141–148, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Chen, H. and He, B. (2013). Automated essay scoring by maximizing human-machine agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1741–1752, Seattle, Washington, USA, October. Association for Computational Linguistics.

Cohen, J. (1968). Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

Dong, F. and Zhang, Y. (2016). Automatic features for essay scoring – an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1072–1077, Austin, Texas, November. Association for Computational Linguistics.

Dong, F., Zhang, Y., and Yang, J. (2017). Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada, August. Association for Computational Linguistics.

Frank, E. and Hall, M. (2001). A simple approach to ordinal classification. In *12th European Conference on Machine Learning*, pages 145–156. Springer.

Frank, E., Hall, M., and Witten, I. (2016). The weka workbench. *Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", 4th edition. Morgan Kaufman, Burlington*.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Page, E. B. (1966). The imminence of... grading essays by computer. *The Phi Delta Kappan*, 47(5):238–243.

Persing, I. and Ng, V. (2014). Modeling prompt adherence in student essays. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1534–1543, Baltimore, Maryland, June. Association for Computational Linguistics.

Persing, I., Davis, A., and Ng, V. (2010). Modeling organization in student essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 229–239, Cambridge, MA, October. Association for Computational Linguistics.

Phandi, P., Chai, K. M. A., and Ng, H. T. (2015). Flexible domain adaptation for automated essay scoring using correlated linear regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 431–439, Lisbon, Portugal, September. Association for Computational Linguistics.

Shermis, M. D. and Burstein, J. (2013). *Handbook of automated essay evaluation: Current applications and new directions*. Routledge.

Somasundaran, S., Burstein, J., and Chodorow, M. (2014). Lexical chaining for measuring discourse coherence quality in test-taker essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 950–961, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Taghipour, K. and Ng, H. T. (2016). A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas, November. Association for Computational Linguistics.

Zesch, T., Wojatzki, M., and Scholten-Akoun, D. (2015). Task-independent features for automated essay grading. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 224–232, Denver, Colorado, June. Association for Computational Linguistics.

[8] https://creativecommons.org/licenses/by-sa/4.0/

MirasText: An Automatically Generated Text Corpus for Persian

Behnam Sabeti, Hossein Abedi Firouzjaee, Ali Janalizadeh Choobbasti, S.H.E. Mortazavi Najafabadi, Amir Vaheb
Miras Technologies International
NO. 3, 2nd Alley, North Sheikh Bahai St., Tehran, Iran
{behnam, hosein, ali, hani, amir}@miras-tech.com

Abstract

Natural Language Processing is one of the most important fields of artificial intelligence. The rapid growth of digital content has made this field both practical and challenging at the same time. As opposed to less-resourced languages like Persian, there are several text corpora in dominant languages like English which can be used for NLP applications.

In this paper, MirasText which is an automatically generated text corpus for Persian language is presented. In this study, over 250 Persian websites were crawled and several fields like content, description, keywords, title, etc have been extracted to generate MirasText. Topic modeling and language modeling are used to validate the generated corpus. MirasText has over 2.8 million documents and over 1.4 billion tokens, which to our knowledge is the largest Persian corpus currently available.

Keywords: Natural Language Processing, Computational Linguistics, Corpus

1 Introduction

Natural language processing (NLP) is a field of artificial intelligence and computational linguistics. It is mainly concerned with the interactions between computers and human beings. Many machine learning approaches have been proposed to solve NLP problems. In recent years because of the availability of computational resources and the introduction of new promising deep learning methods, deep learning approaches have received a lot of attention to solve various NLP problems. Today there are countless online web pages containing textual resources, which can be used to train deep learning models.

There are several text corpora available for dominant languages like English. For instance, Google published Google books n-gram corpus (Lin et al., 2012) which contains more than 800 billion tokens. Another example is Amazon review dataset (He and McAuley, 2016) which contains more than 142 million reviews crawled from Amazon website. Persian language, on the other hand, is one of the less-resourced languages. Although there are some textual datasets for this language but they do not contain huge amount of documents and consequently deep leaning methods that usually require a large corpus of articles to produce valid and reliable results may not be applicable to these corpora. For instance, Hamshahri is a corpus for Persian language (AleAhmad et al., 2009) that contains about 300 thousand news articles with semantic tags. Peykare is another Persian corpus (Bijankhan et al., 2011) with more than 100 million words. Although there is enough Persian content on the internet, but the current datasets need to be enriched.

In this paper we present MirasText which is an automatically generated text corpus for Persian. We crawled more than 250 Persian websites and processed the content to generate this new dataset. MirasText contains contents from web pages with their associated metadata like keywords, description, title, etc. MirasText contains more than 1.4 billion words and can be used for a variety of NLP applications like language modeling, summarization, title generation, keyword extraction, etc.

In the rest of this paper, Section 2 provides some basic information about MirasText. The generation process of this new corpus is discussed in Section 3. In section 4 we provide some statistics about different fields of the corpus and its semantics. Corpus validation and experimental results are presented in section 5. The conclusion is presented in Section 6, at the end of this paper.

2 Corpus Description

MirasText is a text corpus which is about **15.3** gigabytes. It is formatted as a standard CSV file, in which each line represents a document. Each document has 6 fields which are described in table 2

Field Name	Description
Content	web-page main content
Summary	content summary
Keywords	content keywords
Title	content title
Website	base website
URL	exact URL of the web-page

Table 1: Corpus description

Each line in MirasText contains a document with 6 fields. We had to separate each field with a delimiter that is unique enough to ensure soundness of the CSV file. The delimiter used here is three stars (i.e. ***).

MirasText is free to use for both research and business purposes and is available for download at Miras-Tech website[1].

3 Corpus Generation

The generation process of the corpus is discussed in this section. The overall architecture is illustrated in Figure 1. Crawler uses a seed of website links to explore the web, from which the results are then passed to duplicate extractor (remover) which removes the duplicate contents and writes

[1] www.miras-tech.com

the unique contents in the database. A preprocessing phase is then applied to the crawled contents to generate the final corpus. We used state of the art technologies in each part of the system. In the reminder of this section, we will explain each module in detail.

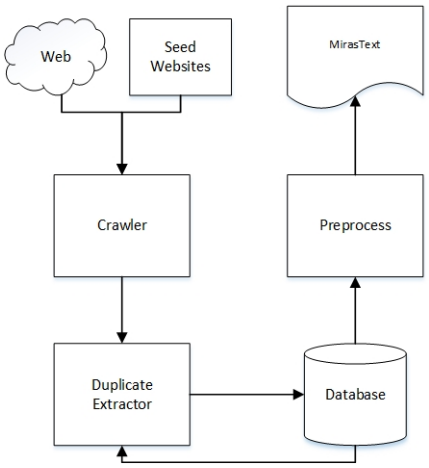


Figure 1: The system architecture used for generation of MirasText.

3.1 Seed Websites

We used a set of 250 websites to generate MirasText. These websites are selected from a wide range of scopes to ensure the diversity of corpus. Seeded websites are summarized based on their scopes in Figure 2. As shown in Figure 2, a large fraction of websites are news agencies which contain any kind of content and cannot be categorized as one of the main classes. We further analyzed corpus content to explore corpus distribution on classes in section 4.1

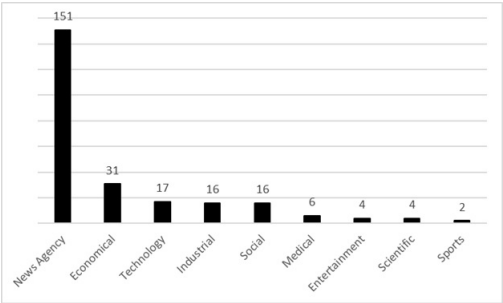


Figure 2: The scopes of initial website seeds used for crawling.

3.2 Crawler

As mentioned earlier, dataset used in this research is comprised of millions of pages crawled from hundreds of Iranian websites (mainly news agencies). As for crawler, a well-known stream processing and crawling technology is used i.e. Storm Crawler (based on Apache Storm stream processing platform (Evans, 2015)). Given a set of valid URLs, Storm Crawler fetches contents of these pages and processes the contents using a set of actions such as simple text parsing, URL extraction, keyword extraction, etc. The main steps of the crawling process are as follows

1. HTML pages are fetched as raw data then the links inside these pages are extracted for further crawling.
2. The contents of retrieved pages are parsed in order to extract useful information such as texts and keywords associated with them.
3. The last step is to index (store) the extracted fields associated with every page into a database and select a new set of URLs to continue crawling.

A high level overview of the crawling process is shown in Figure 3.

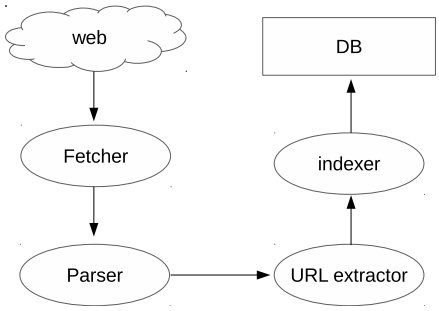


Figure 3: The overview of crawling process.

3.3 Duplicate Removal

Through the crawling process there is a possibility that certain pages with the same texts are encountered on a website. The reason for this is the ambiguity of categories in which a text could relate to, for instance a text could be categorized as *political* and *economical* simultaneously and so there would be 2 copies of it with different URLs on a website (in political and economical subsections of a web site). To remove duplicate pages from the corpus, a filtering process is used based on a bloom filter. A bloom filter (Almeida et al., 2007) is a probabilistic data structure for checking if an element is the member of a set efficiently in terms of memory used. It can also be used for removing duplicate entities from a data set by filtering out the unique ones.

4 Corpus Statistics

In this section some statistics about corpus documents and semantics is illustrated. MirasText has more than 2.8 million documents where each document has several fields. Content is the main field in each document that contains the article in the corresponding URL. Total words in the content field is more than 1.4 billion. Not only MirasText is the largest text based dataset in Persian language but also it is competitive to similar corpora in other languages like English in the sense of volume and variety. Table 2 summarizes some statistics about MirasText.

MirasText has been crawled from a wide variety of websites each containing articles with different sizes. This makes the contents length varying in a wide range of sizes from 10

Total Documents	2,835,414
Total Content Words	**1,429,878,960**
Average Content Length	504.3
Average Keywords	8.4
Average Description Length	19.8
Average Title Length	9.5

Table 2: Corpus statistics

words up to 15000 words long. Figure 4 is a histogram of content length which describes the distribution of contents on different lengths. It can be seen that most of the contents have up to 1000 words.

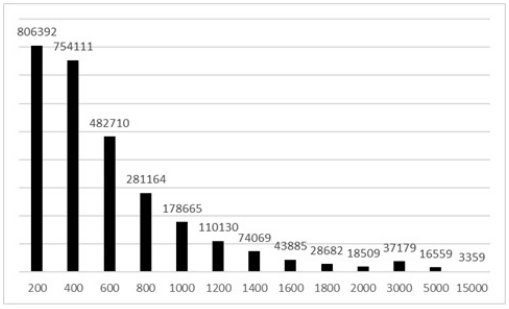


Figure 4: The histogram of content length in terms of words.

4.1 Topic Modeling

In order to explore corpus content, topic modeling is employed. It is practically impossible to manually check every document due to the size of the corpus, so we decided to use Latent Dirichlet Allocation (LDA) (Blei et al., 2003). LDA is a generative model which describes each document as a distribution on topics and describes each topic as a distribution on words. MirasText is modeled using Mallet[2] implementation of LDA which results in $\boldsymbol{T}$ and $\boldsymbol{D}$ matrices. $\boldsymbol{T}$ is a L*W matrix where L is the number of topics and W is the number of unique words. The distribution of the i'th topic on unique words is represented in the i'th row of $\boldsymbol{T}$. $\boldsymbol{D}$ is a N*L matrix where N is the number of documents. Each row in $\boldsymbol{D}$ represents the distribution of each document on all topics. Each topic is manually categorized into a set of predefined classes using its distribution on words. Knowing the distribution of documents on topics and assigning each topic to a class, we can classify each document as well. This method will result in each document being distributed on the set of predefined classes. We then averaged all documents' distributions to generate Figure 5 which gives us some insight about the contents of the corpus . It is worth mentioning that Figure 5 is just an estimation because it is generated in a semi-supervised manner.

5 Corpus Validation

MirasText is crawled automatically and needs to be validated which is done using word representation learning.

[2]http://mallet.cs.umass.edu/

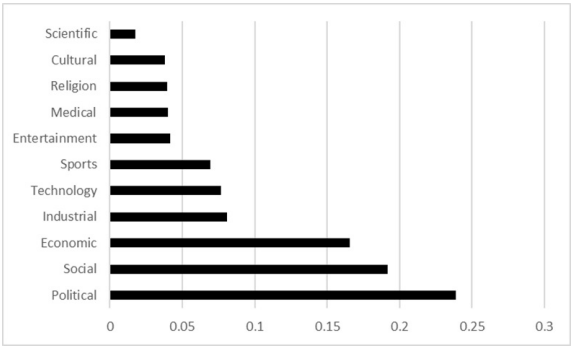


Figure 5: The distribution of contents.

word2vec is a word representation learning model that tries to code each word in a fixed length vector using the context of that word (Mikolov et al., 2013). More concretely, given a set of sentences, *word2vec* converts each unique word to a vector that represents the semantics of that word. This means that after training *word2vec* on a large and valid dataset, words with similar meanings will end up having similar vector representations. In order to validate this new corpus, *word2vec* is trained on MirasText, if the generated word representations are in fact close in the case of semantically similar words, we can conclude that MirasText data is in fact coherent and valid.
Google has published ***Google N-gram*** which contains more than 800 billion tokens from Google books (Lin et al., 2012). We trained *word2vec* model on Google N-gram corpus and used the results as a base line to validate our corpus. The overall steps of our experiment are as follows:

1. *word2vec* model is trained on MirasText and Google N-gram.

2. 1000 Persian words from a wide variety of fields are selected to form a seed list. This list is then converted to English to give us the equivalent seed list in English.

3. Word2vec model trained on MirasText is used to find 20 similar words for each word in Persian seeds where each word cluster is denoted by

$$M_i, i = 1 : 1000.$$

4. Similarly, word2vec model trained on Google N-gram is used to find 20 similar words for each word in English seed. Each word cluster is denoted by

$$G_i, i = 1 : 1000.$$

5. Finally, generated word clusters are compared to see how much they overlap. We consider word2vec model trained on Google N-gram being the best word representation model possible as it is trained on more than 800 billion tokens. If we get similar word clusters from our 1.4 billion tokens, then we can claim MirasText being as valid and coherent as Google N-gram.

English word	Google N-gram cluster	Persian word	MirasText cluster
Doctor	Doctors, surgeon, dentist, pharmacist, nurse, psychologist, oncologist, gynecologist, physician, cardiologist	پزشک	دندانپزشک، پزشکان، روانپزشک، بیمار، جراح داروساز، ماما، ویزیت، پرستار، ارتوپد
Brother	Brothers, cousin, uncle, father, nephew, son, younger, sibling, dad, twin	برادر	خواهر، برادرها، عمو، کوچک، خواهرزاده پسرعمو، برادرزاده، پدر، مادر، باجناق
Football	Sports, baseball, athletics, baseball, soccer, league, coach, athletic, rugby, team	فوتبال	هندبال، والیبال، فوتسال، بسکتبال، لیگ، مربی بازیکن، باشگاه، جودو، فوتبالی

Table 3: The sample equivalent word clusters extracted from MirasText and Google N-gram

5.1 Experimental Results

We conducted evaluations according to the general measuring method used in the Information retrieval evaluation, i.e. precision (P), recall (R) and F1-Measure. The evaluation measures are defined as follows:

$$P = \frac{\sum_{i=1}^{1000} |G_i \cap M_i|}{\sum_{i=1}^{1000} |M_i|} \quad (1)$$

$$R = \frac{\sum_{i=1}^{1000} |G_i \cap M_i|}{\sum_{i=1}^{1000} |G_i|} \quad (2)$$

$$F1 = \frac{2PR}{P+R} \quad (3)$$

We used the Gensim implementation [3] of *word2vec* for training. Each generated model is fed with the corresponding Persian or English seed words to generate word clusters G_i and M_i (for i=1 to 1000). It is worth mentioning that computing precision, recall and F1-Measure needs to be done manually because each G_i and M_i are word clusters in English and Persian respectively, and computing their overlap needs translation. Although this translation could be done automatically, we preferred to do this manually in order to get more reliable results. Table 3. illustrates three cherry-picked word clusters generated by *word2vec* on Google N-gram and MirasText.

After manually computing the evaluation measures, we get a F1-Measure of ***0.75*** which indicates high correlation between extracted clusters from Google N-gram and MirasText. This experiment represents the validity and coherence of MirasText as *word2vec* could learn a reliable word representation from it.

6 Conclusion

In this paper, Miras-Text which is an automatically generated text corpus for Persian language, is presented. The system developed in this study uses a list of websites to generate a text corpus. This system crawls the specified websites and extracts useful information like content, description, keywords, title, etc. The generated corpus contains more than 2.8 million documents and more than 1.4 billion content words. MirasText is the largest Persian text corpus available which can be used for a variety of NLP applications like language modeling, automatic summarization, keyword extraction, title generation, etc. In order to validate the coherence of Miras-Text, a word2vec model is trained both on MirasText and Google N-gram corpus. The trained models are then used to generate some word clusters. The comparison of these word clusters shows high correlation between the two models which indicates the validity and coherence of MirasText.

7 Acknowledgements

The authors would like to thank everyone at Miras-Tech especially the members of team Pegasus who put a lot of effort into building the presented corpus.

8 References

AleAhmad, A., Amiri, H., Darrudi, E., Rahgozar, M., and Oroumchian, F. (2009). Hamshahri: A standard persian text collection. *Knowledge-Based Systems*, 22(5):382–387.

Almeida, P. S., Baquero, C., Preguiça, N., and Hutchison, D. (2007). Scalable bloom filters. *Information Processing Letters*, 101(6):255–261.

Bijankhan, M., Sheykhzadegan, J., Bahrani, M., and Ghayoomi, M. (2011). Lessons from building a persian written corpus: Peykare. *Language resources and evaluation*, 45(2):143–164.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Evans, R. (2015). Apache storm, a hands on tutorial. In *Cloud Engineering (IC2E), 2015 IEEE International Conference on*, pages 2–2. IEEE.

He, R. and McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, pages 507–517. International World Wide Web Conferences Steering Committee.

Lin, Y., Michel, J.-B., Aiden Lieberman, E., Orwant, J., Brockman, W., and Petrov, S. (2012). Syntactic annotations for the google books ngram corpus. In *Proceedings of the ACL 2012 System Demonstrations*, pages 169–174. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[3] radimrehurek.com/gensim/models/word2vec.html

The Reference Corpus of the Contemporary Romanian Language (CoRoLa)

Verginica Barbu Mititelu, Dan Tufiș, Elena Irimia
Romanian Academy Research Institute for Artificial Intelligence
13 13 Septembrie Road, Bucharest 050711, Romania
{vergi, tufis, elena}@racai.ro

Abstract

We present here the largest publicly available corpus of Romanian. Its written component contains 1,257,752,812 tokens, distributed, in an unbalanced way, in several language styles (legal, administrative, scientific, journalistic, imaginative, memoirs, blogposts), in four domains (arts and culture, nature, society, science) and in 71 subdomains. The oral component consists of almost 152 hours of recordings, with associated transcribed texts. All files have CMDI metadata associated. The written texts are automatically sentence-split, tokenized, part-of-speech tagged, lemmatized; a part of them are also syntactically annotated. The oral files are aligned with their corresponding transcriptions at word-phoneme level. The transcriptions are also automatically part-of-speech tagged, lemmatised and syllabified. CoRoLa contains original, IPR-cleared texts and is representative for the contemporary phase of the language, covering mostly the last 20 years. Its written component can be queried using the KorAP corpus management platform, whereas the oral component can be queried via its written counterpart, followed by the possibility of listening to the results of the query, using an in-house tool.

Keywords: Romanian, reference corpus, annotation

1. Introduction

Language resources in the form of large corpora have been being created for more and more languages. We present here the results of a four-year project focused on the creation of a big **c**orpus for **co**ntemporary **Ro**manian **la**nguage, called CoRoLa (corola.racai.ro). This has not been a singular effort: smaller previous or parallel projects (ANVSIB – http://speed.pub.ro/anvsib, SSPR – http://dev.racai.ro/ti/wordpress/, PARSEME – https://typo.uni-konstanz.de/parseme/) contributed to this project outcomes. In its turn, CoRoLa will contribute to other projects, larger ones, going beyond the national level (DruKoLA – http://www1.ids-mannheim.de/direktion/kl/projekte/drukola.html) and turning European.

The CoRoLa corpus is now the reference corpus for contemporary Romanian. It is the largest one, containing 1,257,752,712 tokens for the written component and almost 152 hours of recordings for the oral component (the detailed structure is presented below). The texts cover all language styles, four major domains for which 71 subdomains were defined, thus ensuring a wide vocabulary coverage. The corpus can be reliably used as a basis for the creation of dictionary entries, grammar studies, other language reference materials, as well as for training and testing algorithms and systems for language processing.

CoRoLa has been jointly developed, as a priority project of the Romanian Academy, by two institutions: "Mihai Drăgănescu" Research Institute for Artificial Intelligence (from Bucharest) and the Institute of Computer Science (from Iași). Being located in different geographical and cultural regions of the country, the two partners could more easily contact texts providers from their vicinity, as unmediated, face-to-face contact and negotiations proved necessary for agreeing upon a protocol of collaboration after the correct understanding by our texts providers of the way texts are to be processed and further exploited when part of the corpus.

2. Related work

National corpora have been created for many languages: American English (Ide and Suderman, 2004), British English (http://www.natcorp.ox.ac.uk/), Bulgarian (Koeva et al., 2012), Croatian (Tadić, 2002), Czech (Křen et al., 2016), German (Kupietz and Lüngen, 2014), Hungarian (Oravecz et al., 2014), Polish (Przepiórkowski et al., 2011), Russian (http://www.ruscorpora.ru/en), Turkish (Aksan et al., 2012), Eastern Armenian (http://www.eanc.net/), Welsh (Piao et al., 2016) and others.

Most of them are big (counting hundreds of thousands of words), with some being even huge (with over one billion words, see the Hungarian corpus, or even with tens of billions, as is the case of the German corpus). All of them reflect the language in the last twenty years, usually from various genres and domains; an exception is again the German corpus, which contains texts dating back to 1956 (Kupietz and Lüngen, 2014).

Romanian corpora exist either as components of multilingual corpora or as monolingual resources. Within the former category we mention:

- the RO-JRC-Acquis (Ceauşu, 2008) – contains over 30 million words and reflects the law domain;
- Eur-Lex judgements corpus (Baisa et al., 2016) – contains over 17 million words and also reflects the law domain;
- EUROPARL corpus (Koehn, 2005) – contains almost 10 million words from the European Union Proceedings;
- OPUS (Tiedemann, 2012) – contains almost 300 million words and reflects mostly the legislative and administrative domains.

From the latter category, we mention:

- the RoCo-news corpus (Tufiș and Irimia, 2006) – contains around 7 million tokens and reflects the journalistic language;
- the RoWaC corpus (Macoveiciuc and Kilgarriff, 2010) – contains almost 45 million words gathered from the web;

- the Romanian balanced corpus ROMBAC (Ion et al., 2012) – contains about 36 million words, distributed rather evenly into five domains: journalistic, pharmaceutical and medical, law, literary history and fiction.

3. CoRoLa's Characteristics

The CoRoLa corpus stands out from these corpora due to its size, structure, origin and quality of texts. It is structured in files, each with associated metadata in CMDI format (most of them automatically created, but quite a lot manually), and annotated at several levels, as presented below.

The data collection and cleaning has been done as described by Tufiș et al. (2016). It involved both automatic and manual interventions on the data, for:

- bringing them into txt format (the most common formats in which we received them were pdf and doc(x)),
- separating texts (e.g., newspapers articles were separated in different files, just like chapters from edited books),
- recuperating text elements (paragraphs limits, removal of column marking newlines, words were recreated from their hyphenated form at the end of rows),
- removing unnecessary elements from the text (page numbers, headers, footers, footnotes, tables, figures, captions, etc.),
- diacritics insertion in the texts that lacked them or diacritics replacement (when non-standard ones occurred).

We present in this section several key characteristics of the texts included in CoRoLa: the time span they cover, originality, representativeness and I(ntellectual) P(roperty) R(ights)-clearance.

3.1 Time span

The corpus contains texts that could be collected in electronic format only, involving no OCR of scanned paper printed books. This condition limited the time coverage. The vast majority of texts reflect the language from the last twenty years. Texts from earlier time are legal ones and a few imaginative ones.

3.2 Originality

The vast majority of texts in CoRoLa are original ones: they are written in Romanian by native speakers. Only a few are translations. These belong to the legal style (translation of European legislation).

Another element of originality is the fact that the texts underwent no correction.

3.3 Representativeness

The corpus is representative for the contemporary language in that it contains texts from all language styles, major domains and many subdomains. The focus is only on the literary language. Nevertheless, the informal style can be found in the literary texts, although not explicitly marked in any way in the data and, thus, difficult to spot automatically.

3.4 IPR-clearance

During this project, great effort was invested in obtaining texts from their owners in a free of charge manner, given the scarce funding available. Major publishing houses, newspapers, magazines, news agencies, individual authors, and bloggers were contacted. Written protocols were signed with them so that we are allowed to freely obtain their (written or oral) texts, to store them on our servers, to process and annotate them and then to make them available for querying. We consider this a major achievement within this project, although this implies making (the largest part of) CoRoLa accessible only for querying and not for downloading.

The author's rights law in Romania does not have scope over legal and administrative texts. As such, they do not raise any storage or access problem.

4. Statistics

4.1 Written Texts

The distribution of the CoRoLa written texts across various styles is rendered in Table 1. We notice the unequal distribution of texts with respect to their style: legal texts are predominant.

Style	Number of tokens
legal	930,728,509
scientific	138,784,668
blogpost	53,704,460
journalistic	50,793,311
imaginative	47,727,438
administrative	17,759,778
memoirs	16,999,893
unclassified	1,254,755
TOTAL	**1,257,752,812**

Table 1: CoRoLa's texts distribution according to their style.

The distribution of texts according to the domain to which they belong is presented in Table 2.

Domain	Number of tokens
society	990,852,812
science nature	101,198,918
unclassified	93,511,926
arts and culture	70,510,600
nature	1,678,556
TOTAL	**1,257,752,812**

Table 2: CoRoLa's texts distribution according to their domain.

These domains are further classified into a various number of subdomains, as follows:

- architecture, art history, dance, design, fashion, film, folklore, literature, music, painting and drawing, poetry, sculpture and theater are the 13 subdomains of the arts and culture domain;

- environment, natural disasters, natural resources and universe are the 4 subdomains of the nature domain;
- administration, army, economy, education, entertainment, family, gossip, health, law, politics, religion, social events, social movements, sports, and tourism are the 15 subdomains of the society domain;
- archaeology, astronomy, biology, chemistry, constructions, criminalistics, engineering, ethnology, geography, geology, history, informatics, juridical sciences, linguistics, logics, mathematics, medicine, metrology, military science, oenology, pedagogy, pharmacology, philology, philosophy, physics, political sciences, psychology, religious studies and theology, sociology, standards, technics/technology are the 31 subdomains of the science domain.

We present below the data with the subdomains of each domain.

Subdomain	Number of tokens
literature	47,695,605
other	14,790,875
film	2,181,588
music	1,690,767
theatre	1,277,762
painting and drawing	1,153,013
architecture	776,447
folklore	445,277
art history	286,449
design	111,892
dance	42,407
fashion	24,545
sculpture	21,632
poetry	12,341
TOTAL Arts and culture	**70,510,600**

Table 3: CoRoLa's texts distribution according to their subdomain in the Arts and culture domain.

Subdomain	Number of tokens
other	1,003,512
environment	472,777
natural resources	110,728
universe	56,652
natural disasters	34,887
TOTAL	**1,678,556**

Table 4: CoRoLa's texts distribution according to their subdomain in the Nature domain.

Subdomain	Number of tokens
law	931,609,324
politics	19,820,835
other	11,977,554
economy	9,013,003
education	5,669,044
sports	3,538,550
religion	2,615,460
administration	2,193,158
gossip	1,676,574
health	862,677
entertainment	584,093
social events	519,484
tourism	436,682
family	319,614
social movements	14,205
army	2,555
TOTAL	**990,852,812**

Table 5: CoRoLa's texts distribution according to their subdomain in the Society domain.

Subdomain	Number of tokens
history	18,126,581
pharmacology	10,481,131
philology	10,269,512
medicine	10,093,872
sociology	8,228,897
geography	6,098,170
philosophy	4,868,934
psychology	4,727,630
political sciences	4,571,354
linguistics	4,116,164
religious studies and theology	3,734,212
other	3,196,889
pedagogy	1,906,633
biology	1,753,303
technics/technology	1,553,913
constructions	1,438,636
chemistry	1,302,917
juridical sciences	936,773
physics	741,234
mathematics	658,788
informatics	657,232
military science	457,075
archaeology	237,356

standards	195,617
astronomy	155,503
oenology	126,803
engineering	127,537
criminalistics	104,115
ethnology	98,828
geology	97,603
metrology	76,188
logics	59,518
TOTAL	**101,198,918**

Table 6: CoRoLa's texts distribution according to their subdomain in the Science domain.

As a general remark, the corpus is unbalanced. We are still trying to enrich the less numerous categories, but, when collecting texts, it is hard to compete with a category without IPR restrictions (in this case, legal texts).

4.2 Oral Texts

The oral texts in CoRoLa are mainly professional recordings from various sources (radio stations, recording studios). They are accompanied by the written counterpart: the transcription either from their provider or made by us. As a consequence, different principles applied in their transcription.

Another part of the oral corpus is represented by read texts: read news in radio stations, texts read by professional speakers recorded in studios, and extracts from Romanian Wikipedia read by non-professionals, by volunteers, recorded in non-professional environments. In their case, the written component is provided by the sources, or was collected by us. An exception is the corpus RSS, compiled by Stan et al. (2011), which was enriched with ToBI-like annotation within our institute (Boroș et al., 2014).

The oral texts cover 151 hours 57 minutes and 21 seconds. However, not all of them have been processed yet. Their distribution according to the text type is given in Table 7 below.

Text type	**Time (h:m:s)**
News and radio interviews	119:52:33
News and fairy tales	03:44:00
Romanian Wikipedia	04:22:02
miscellanea	23:58:46
TOTAL	**151:57:21**

Table 7: CoRoLa's oral text types and duration.

5. Annotation Levels

The annotation levels of the CoRoLa corpus are partially different for written and oral texts.

5.1 Written Texts

They are automatically sentence-split, tokenised and morpho-syntactically annotated, and lemmatised with an in-house tool called TTL (Ion, 2007), having an accuracy of about 97.5% (Tufiș et al., 2008). We have not evaluated its accuracy on CoRoLa, though. The tagset of morpho-syntactic descriptions (MSDs) is compliant with MULTEXT-EAST specifications (Erjavec, 2012).

A subcorpus of CoRoLa (9,522 sentences), containing samples of texts from the different styles, domains and subdomains existent in the corpus, is also consistently syntactically annotated and manually validated, following the UD principles (universaldependencies.org) and inventory of relations (Barbu Mititelu et al., 2016). However, we intend to annotate the rest of the CoRoLa corpus syntactically, as well, once we tune our recently developed parser (Ion et al., 2018) on corresponding styles, domains or subdomains.

Moreover, a part of the texts in the medical subdomain (18,000 sentences) is also annotated with medical terms of the type ANAT (anatomy or body parts), DISO (disorders), PROC (medical procedures) and CHEM (chemical substances) (Mitrofan, 2017; Mitrofan and Tufiș, 2018).

What is more, 51,500 sentences from the journalistic style were annotated with four types of verbal multiword expressions (namely, ID (idioms), LVC (light verb constructions), IReflV (inherently reflexive verbs) and OTH (other)) for the PARSEME shared task on verbal multiword expressions identification (Savary et al., 2017).

5.2 Oral Texts

Up to now only a little more than half of the collected recordings (totaling about 300 hours) have been automatically preprocessed. Their written counterpart (counting 746,187 tokens) has been lemmatised, part-of-speech tagged and syllabificated (Boroș and Dumitrescu, 2015); some allophones were identified and analyzed. The oral and the written texts are time aligned at the sentence, word and phoneme levels (Boroș et al., 2018) and this alignment was automatically encoded in separate files.

6. Access to CoRoLa

The corpus was publicly launched in December 2017. According to the protocols agreed upon and signed with texts providers, the corpus cannot be downloaded from our servers, but only queried. The results of queries are strings of a limited length, which makes the recovery of original texts impossible.

Access to written texts (http://89.38.230.10:5555/) is provided by the KorAP corpus query and analysis platform (Bański et al., 2014; Diewald et al, 2016), due to the running DruKoLA project (Cosma et al., 2016), in which Institut für Deutsche Sprache from Mannheim, where KorAP has been being developed, is our partner. KorAP allows for different type of linguistic searches (according to the annotation levels in the corpus, as well as combined levels), for creating virtual subcorpora (based on the metadata of the corpus). It allows regular expressions in query formulation. In Figure 1 we present a snapshot of the 37,464 results of the query `[orth=cel][orth=mai][drukola/m=pos:adverb]`, which searches for adverbs at the superlative degree.

A part of the corpus (about a fifth of it) is also available for search within the NLP-CQP web interface

(http://89.38.230.23:1234/#). This is meant to help those corpus users that cannot yet use a formal language for querying to formulate the query in controlled natural language (Romanian), then translates it in C(orpus)Q(uery)P(rocessor) (http://cwb.sourceforge.net/files/CQP_Tutorial/) format and retrieves the results. Figure 2 shows the query formulated in Romanian for finding 100 sentences in which the word "nu" occurs immediately after a verb. This query is translated into the CQP language: see the bottom of the figure.

At the moment, the oral texts can be queried (http://89.38.230.23/corola_sound_search/index.php) only via their transcription followed by rendering the relevant aligned speech segment. This is ensured by the in-house Oral Corpus Query Platform (OCQP) created by Vasile Păiș. It allows searching for lemmas or occurring words and morphological restrictions on them. The results are either sequences of 5 words or the whole sentences in which the specified form occurs. Besides the written form, the user also gets displayed the oral files, both for the searched for form and for the whole sentence to which it belongs. In Figure 3 we exemplify with the search of the word "copilărie" (En. "childhood") and a few of the results obtained.

7. Sustainability

Although CoRoLa is already accessible to the users, its enrichment continues, targeting a balanced distribution of texts from various perspectives: text style, domain or subdomain, written and oral components. Two new projects (ReTeRom – Resources and technologies for developing human-machine interfaces in Romanian, and Heimdallr – Real-time Keyword Spotting in Telephone conversations) will enhance its oral component. In the perspective of creating EuReCo (Kupietz et al., 2017), CoRoLa can be enriched and exploited in harmony with other European corpora joining this initiative.

8. Conclusions and further work

CoRoLa is the result of a four-year effort, in which texts were collected from various sources, with the stated goal of going beyond what the web can offer. Its realization was possible due to the contribution of both individual people and juridical entities, all of them IPR owners. We are grateful to them all. Further data cleaning is necessary and has already started. Annotation quality needs to be assessed and a bootstrapping mechanism for its improvement will be applied. Moreover, further levels of annotation are targeted, and the syntactic one is a priority for the team.

Empirical studies of the Romanian language are now possible at various linguistic levels, in different styles, domains and subdomains. For a statistical analysis of the language we can also offer access to word embeddings (Păiș and Tufiș, 2018) and to n-grams of various sizes, as this does not constitute a breach of the protocols signed with texts providers.

9. References

Aksan, Y., Aksan, M., Koltuksuz, A., Sezer, T., Mersinli, Ü., Demirhan, U., Yilmazer, H., Kurtoğlu, Ö., Atasoy, G., Öz, S., Yildiz, I. (2012). Construction of the Turkish National Corpus (TNC). In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012). İstanbul. Turkiye.

Baisa, V., Michelfeit, J., Medveď, M., Jakubíček, M. (2016). European Union Language Resources in Sketch Engine. In *The Proceedings of tenth International Conference on Language Resources and Evaluation (LREC'16)*. European Language Resources Association (ELRA). Portorož, Slovenia.

Bański, P., Diewald, N., Hanl, M., Kupietz, M., Witt, A. (2014). Access Control by Query Rewriting. The Case of KorAP. In *Proceedings of the Ninth Conference on International Language Resources and Evaluation (LREC'14)*. Reykjavik: European Language Resources Association (ELRA), pp. 3817-3822.

Barbu Mititelu, V., Ion, R., Simionescu, R, Irimia, E., Perez, C.A. (2016) The Romanian Treebank Annotated According to Universal Dependencies. In *Proceedings of The Tenth International Conference on Natural Language Processing (HrTAL2016)*.

Boroș, T., Stan, A., Watts, O., Dumitrescu, S.D. (2014). RSS-TOBI - a Prosodically Enhanced Romanian Speech Corpus. In Calzolari, Nicoletta et al. (eds.): *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14).* Reykjavik: ELRA, 316-320.

Boroș, T., Dumitrescu, Ș. (2015). Robust deep-learning models for text-to-speech synthesis support on embedded devices. In *Proceedings of the 7th International Conference on Management of computational and collective Intelligence in Digital Eco-System (MEDES'15)*, 98-102.

Boroș, T., Dumitrescu, Ș., Pais, V. (2018). Tools and resources for Romanian text-to-speech and speech-to-text applications. In *Proceedings of The Second Workshop on Multi-Language Processing in a Globalising World and The First Workshop on Multilingualism at the intersection of Knowledge Bases and Machine Translation*.

Ceauşu, Al. (2008). Colectarea şi procesarea documentelor româneşti ale corpusului JRC-Acquis. In Diana Maria Trandabăţ, Dan Cristea, Dan Tufiş (eds.), *Lucrările atelierului Resurse Lingvistice şi Instrumente pentru Prelucrarea Limbii Române*, Editura Universităţii „Al. I Cuza", Iaşi.

Cosma, R., Cristea, D., Kupietz, M., Tufiș, D., Witt, A. (2016). DRuKoLA - Towards Contrastive German-Romanian Research based on Comparable Corpora. In Proceedings of the fourth CLMC, *LREC 2016, Portoroz,* Slovenia, 28 May, European Language Resources Association (ELRA).

Diewald, N., Hanl, M., Margaretha, E., Bingel, J., Kupietz, M., Bański, P. and A. Witt (2016). KorAP Architecture – Diving in the Deep Sea of Corpus Data. In: Calzolari, Nicoletta et al. (eds.): *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portoroz / Paris: ELRA: 3586-3591.

Erjavec, T. (2012). MULTEXT-East: morpho-syntactic resources for Central and Eastern European languages. *Language Resources and Evaluation*, March 2012, Volume 46, Issue 1, pp. 131–142.

Ide, N., Suderman, K. (2004). The American National Corpus First Release. In Maria Teresa Lino, Maria

Francisca Xavier, Fátima Ferreira, Rute Costa, Raquel Silva (eds.) *Proceedings of the Fourth Language Resources and Evaluation Conference (LREC)*, Lisbon, Portugal,1681-1684.

Ion, R. (2007). *Word Sense Disambiguation Methods Applied to English and Romanian*, PhD Thesis, Romanian Academy.

Ion, R., Irimia, E., Ştefănescu, D., Tufiş, D. (2012). ROMBAC: The Romanian Balanced Annotated Corpus. In Proceedings of LREC 2012, 339-344.

Ion, R., Irimia, E., Barbu Mititelu, V. (2018). Ensemble Romanian Dependency Parsing with Neural Networks. (this volume).

Koeva, S., Stoyanova, I., Leseva, L., Dimitrova, T., Dekova, R., Tarpomanova, E. (2012). The Bulgarian National Corpus: Theory and practice in corpus design. *Journal of Language Modelling*, 1 (1), 65-110.

Koehn, Ph. (2005). Europarl: A parallel corpus for statistical machine translation.In *MT Summit*, vol 5.

Křen, M, Cvrček, V., Čapka, T., Čermáková, A., Hnátková, M., Chlumská, L., Jelínek, T., Kováříková, D., Petkevič, V., Procházka, P., Skoumalová, H., Škrabal, M., Truneček, P., Vondřička, P., Zasina, A. (2016). SYN2015: Representative Corpus of Contemporary Written Czech. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (LREC'16), 2522–2528.

Kupietz, M., Lüngen, H. (2014). Recent Developments in DeReKo. In: Calzolari, Nicoletta et al. (eds.): Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). Reykjavik: ELRA, 2378-2385.

Kupietz, M., Witt, A., Bański, P., Tufiş, D., Cristea, D., Váradi, T. (2017). EuReCo – Joining Forces for a European Reference Corpus as a sustainable base for cross-linguistic research. In Bański, P., Kupietz, M., Lüngen, H., Rayson, P., Biber, H., Breiteneder, E., Clematide, S., Mariani, J., Stevenson, M., Sick, T. (eds.), Proceedings of the Workshop on Challenges in the Management of Large Corpora and Big Data and Natural Language Processing (CMLC-5+BigNLP) 2017 including the papers from the Web-as-Corpus (WAC-XI) guest section. Birmingham, 24 July 2017. - Mannheim: Institut für Deutsche Sprache, 15-19.

Macoveiciuc, M., Kilgarriff, A., 2010, The RoWaC Corpus and Romanian WordSketches, in D. Tufiş, C. Forăscu (eds.), Multilinguality and Interoperability in Language Processing with Emphasis on Romanian, Romanian Academy Publishing House, Bucharest.

Mitrofan, M. (2017). Bootstrapping a Romanian Corpus for Medical Named Entity Recognition. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, 501-509.

Mitrofan, M., Tufiş, D. (2018). BioRo: The Biomedical Corpus for the Romanian Language. (this volume)

Oravecz, C., Váradi, T., Sass, B. (2014). The Hungarian Gigaword Corpus. In Calzolari, Nicoletta et al. (eds.): *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland, 26-31 May, 1719-1723.

Păiş, V., Tufiş, D. (2018). Computing Distributed Representations of Words using the CoRoLa Corpus. *Proceedings of the Romanian Academy, Series A*, vol 19. No.1/2018.

Piao, S., Rayson, P., Archer, D., Bianchi, F., Dayrell, C., El-Haj, M., Jiménez R-M., Knight, D., Křen, M., Löfberg, L., Nawab, R., Shafi, J., Teh, P-L. and Mudraya, O. (2016). Lexical Coverage Evaluation of Large-scale Multilingual Semantic Lexicons for Twelve Languages. Proceedings of the *LREC (Language Resources Evaluation) 2016 Conference,* May 2016, Slovenia.

Pipa, S., Boroş, T. (2016) A Recurrent Neural Networks Approach for Keyword Spotting Applied on Romanian Language**.** In *Proceedings of The 12th International Conference "Linguistic resources and tools for processing the romanian language*. Editura Universităţii "Alexandru Ioan Cuza" Iaşi, vol. 12, 111-119.

Przepiórkowski, A., Bańko, M., Górski, R.G., Lewandowska-Tomaszczyk, B., Łaziński, M., Pęzik, P. (2011). National Corpus of Polish. In *Proceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Poznań, Poland, November 25--27, 2011, 259–263.

Savary, A., Ramisch, C., Cordeiro, S.R., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., Doucet, A. (2017). The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions. In *Proceedings of the 13th Workshop on Multiword Expressions*, Valencia, Spain, 4 April, 31-47.

Stan, A., Yamagishi, J., King, S., & Aylett, M. (2011). The Romanian Speech Synthesis (RSS) corpus: building a high quality HMM-based speech synthesis system using a high sampling rate. *Speech Communication*. 53(3), 442-450.

Tadić, M. (2002). Building the Croatian National Corpus. In *Proceedings of LREC*, Las Palmas, Spain, 29-31 May, 441-446.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the 8th International Conerence on Language Resources and Evaluation (LREC'2012)*

Tufiş, D., Irimia, E. (2006). RoCo-News - A Hand Validated Journalistic Corpus of Romanian, In Proceedings of the 5th International Conference on Language Resources and Evaluation, 22–28.

Tufiş, D., Ion, R., Ceauşu, A., Ştefănescu D. (2008). RACAI's Linguistic Web Services. In Nicoletta Calzolari et al. (Eds.) Proceedings of the 6th LREC, Marrakech, Morocco, European Language Resources Association (ELRA).

Tufiş, D., Barbu Mititelu, V., Irimia, E., Dumitrescu, S.D., Boroş, T. (2016). *The IPR-cleared Corpus of Contemporary Written and Spoken Romanian Language*, in Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, Stelios Piperidis (eds.), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 23-28 May, Portorož, Slovenia, p. 2516-2521.

Annexes.

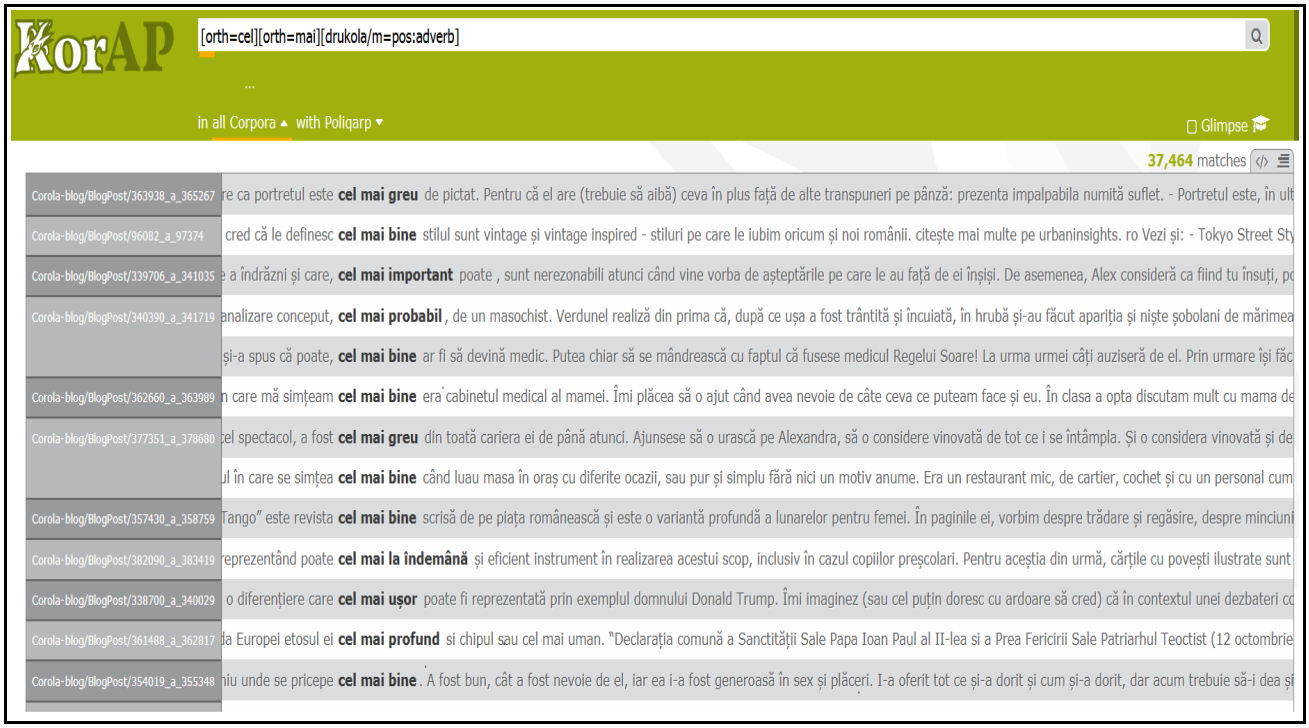


Figure 1. A snapshot of the 37,464 results of the query `[orth=cel][orth=mai][drukola/m=pos:adverb]`, which searches for adverbs at the superlative degree in CoRoLa with KorAP.

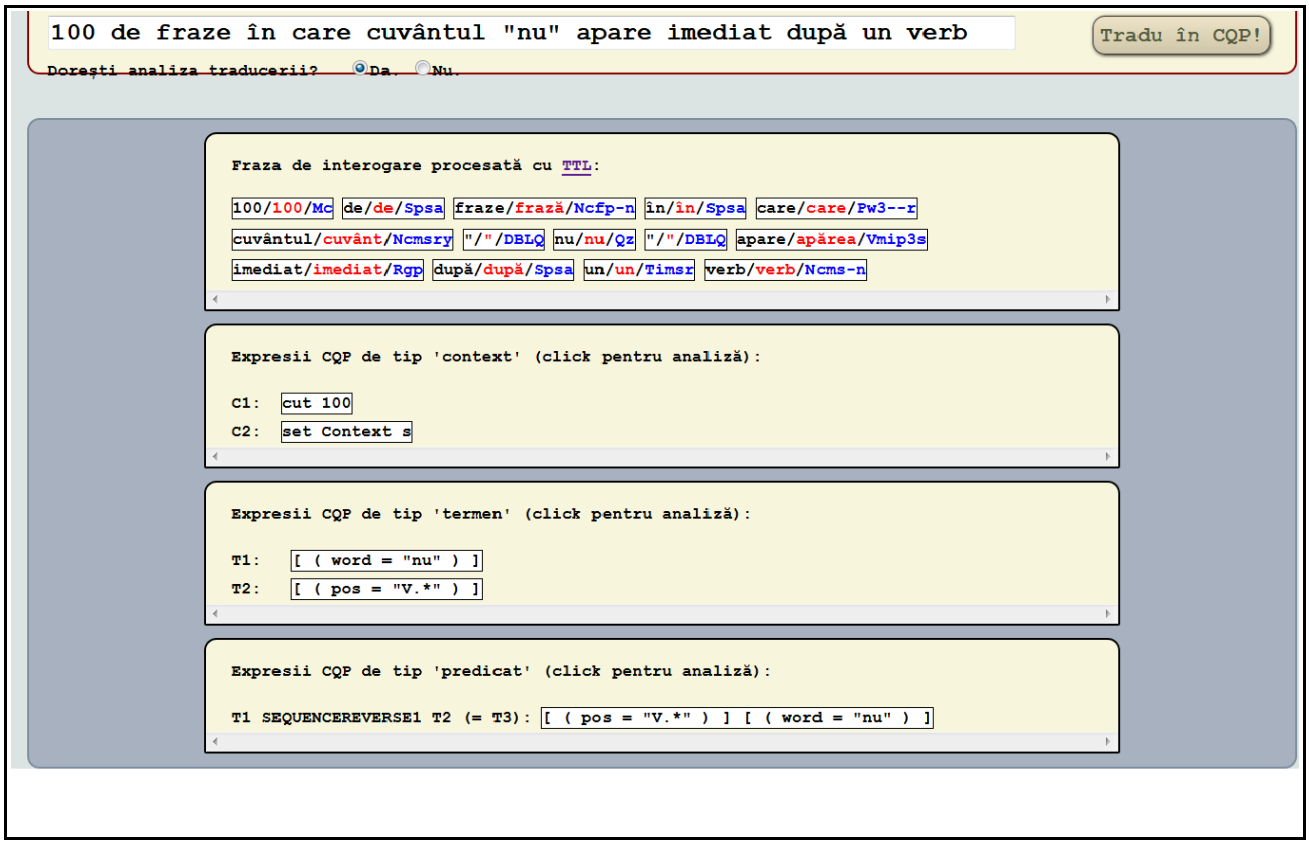


Figure 2. The query formulated in Romanian for finding 100 sentences in which the word "nu" occurs immediately after a verb and its translation into CQP.

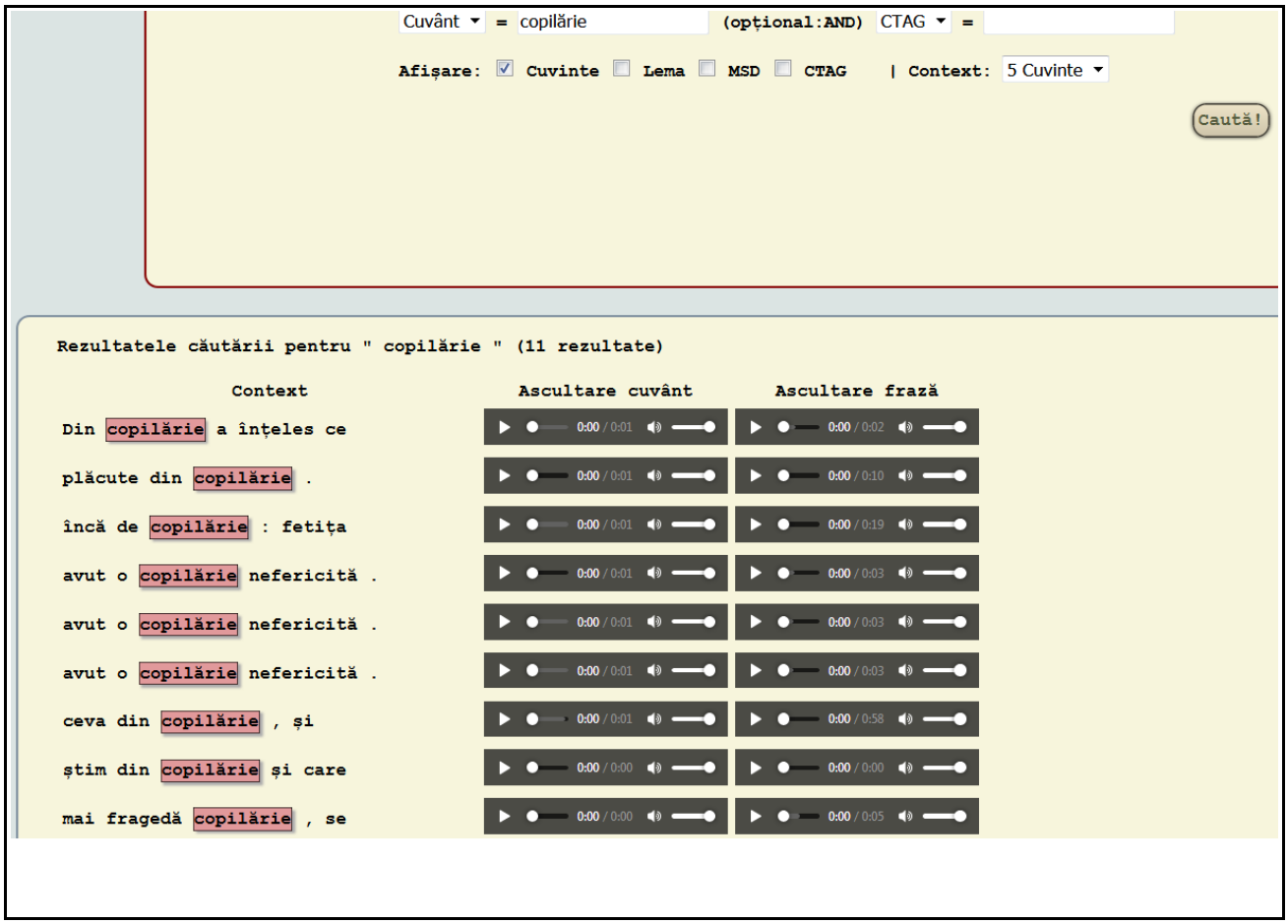


Figure 3. Some results of the search for the word “copilărie” (En. “childhood”) in the oral component of CoRoLa.

A Corpus of Drug Usage Guidelines Annotated with Type of Advice

Sarah Masud Preum, Md. Rizwan Parvez, Kai-Wei Chang, John A. Stankovic
University of Virginia; University of California, Los Angeles
Charlottesville VA 22903; Los Angeles CA 90095
preum@virginia.edu, rizwan@cs.ucla.edu, kwchang@cs.ucla.edu, jas9f@virginia.edu

Abstract

Adherence to drug usage guidelines for prescription and over-the-counter drugs is critical for drug safety and effectiveness of treatment. Drug usage guideline documents contain advice on potential drug-drug interaction, drug-food interaction, and drug administration process. Current research on drug safety and public health indicates patients are often either unaware of such critical advice or overlook them. Categorizing advice statements from these documents according to their topics can enable the patients to find safety critical information. However, automatically categorizing drug usage guidelines based on their topic is an open challenge and there is no annotated dataset on drug usage guidelines. To address the latter issue, this paper presents (i) an annotation scheme for annotating safety critical advice from drug usage guidelines, (ii) an annotation tool for such data, and (iii) an annotated dataset containing drug usage guidelines from 90 drugs. This work is expected to accelerate further release of annotated drug usage guideline datasets and research on automatically filtering safety critical information from these textual documents.

Keywords: Drug Usage Guidelines, Annotation Tool, Medical Corpora, Health Informatics

1. Introduction

Drug safety is crucial for ensuring overall health safety of patients and effectiveness of treatment (Yi et al., 2015). In order to ensure drug safety, with each of their prescribed drugs, patients are often provided with drug usage guidelines (DUG) documents [1]. These documents contain essential information regarding drug usage, including, but not limited to, dosage, drug administration, adverse reactions or symptoms, drug storage and disposal, contraindications, drug-drug interactions, and drug-food interactions (MDS, 2017) as illustrated in Table 1. Doctors and pharmacists are expected to educate patients about drug usage guidelines to ensure drug safety (Patel and Dowse, 2015). Such as, educating a patient who is prescribed *Coumadin* to avoid certain herbal products like *St John's Wort* and foods high in Vitamin K due to potential drug-food interaction. But patients are often unaware of such critical information / advice related to their prescription drugs due to several reasons, including, (i) lack of communication with their doctors and/or pharmacists (Liddy et al., 2014), (Patel and Dowse, 2015) (ii) low health literacy (Wolf et al., 2006), and (iii) volume of received information (Savas and Evcik, 2000).

Textual analysis of drug usage guidelines and automatically filtering critical personalized advice can aid a patient to adhere to the drug usage guidelines. Such research can benefit from annotated datasets where the information is annotated according to its topic, e.g., whether a piece of advice from a DUG document is related to *interaction with food or beverage* or *interaction with other drugs*. For example, the annotated advice from DUG documents can be forwarded to a medication reminder app to present advice that indicates potential interaction with (i) daily activities, (ii) exercise, (iii) diet, and (iv) other drugs. Examples of such advice are presented in Table 1. Presenting advice based on their topics can aid users to receive critical advice effectively and increase drug adherence (Tang et al., 2014), (Jimmy and Jose, 2011). Also, often advice are subjected to physiological and/or temporal conditions. Annotated advice can be used to filter out irrelevant information and personalize DUG documents for a patient. For example, removing pregnancy related advice for a male patient. This will reduce the information burden for the patients, which is often identified as a primary barrier to self-management of chronic diseases (Woolley, 2015).

> ...
> **HOW TO USE:** Read the Medication Guide ...
> Take this medication by mouth with or without food as directed by your doctor or other health care professional, usually once a day. It is very important to take it exactly as directed. ...
> It is important to eat a balanced, consistent diet while taking Warfarin. ... Avoid sudden large increases or decreases in your intake of foods high in vitamin K (such as Broccoli, Cauliflower, Cabbage, Brussels sprouts, Kale, Spinach, and other green leafy vegetables, liver, green tea, certain vitamin supplements). If you are trying to lose weight, check with your doctor before you go on a diet. ...
> Since this drug can be absorbed through the skin and lungs and may harm an unborn baby, women who are pregnant or who may become pregnant should not handle this medication or breathe the dust from the tablets.. ...

Text Box 1: An excerpt from the drug usage guideline (DUG) document of *Warfarin* from MedScape (MDS, 2017). Different colors are used to underline advice related to different topics. Text underlined in blue, green, and red indicate advice related to *drug administration*, *food interaction*, and *pregnancy*, respectively. Such DUG documents are also available from FDA drug database(FDA, 2017) and WebMD(WMD, 2017).

Lack of annotated corpus of the DUG documents limits the potential NLP research on extracting critical information from DUG data to increase drug safety. To bridge this knowledge gap, we develop a novel annotation scheme and a novel, interactive annotation tool to annotate textual ad-

[1] Also known as consumer medical information (CMI) and patient handouts.

vice statements from DUG documents according to their topics. This annotation tool is used to annotate a corpus of 90 online DUG documents and 9,831 sentences. The multi-label annotation results in the first annotated corpus of DUG documents containing 1,611 annotated safety critical drug usage guidelines. We make the annotation tool and the annotated corpus available to the community (Preum et al., 2018). These resources can aid the release of more annotated datasets of DUG documents and accelerate NLP research on automatic extraction of safety critical information from these textual documents.

2. Background

In this section we briefly introduce the drug usage guideline (DUG) data and the relevant existing research.

The DUG documents contain a variety of information spanning different topics, which can be broadly categorized in following classes. (i) basic drug information (e.g., alternate names, diseases that are commonly treated with the drug, ingredients, dosage information), (ii) drug administration related advice (i.e., how and when the drug should be administered), (iii) side effects of the drug, (iv) information regarding how the drug can interact with other drugs, foods and activities, and (v) drug storage and disposal related information. This information is the input for the data annotation tool.

The DUG documents are semi-structured documents where the text contents are organized under different section headers. As shown in the text excerpt in Section 1., under the tile *How To Use*, information related to drug administration and potential drug-food interaction are included. The topics described above are not always structurally organized in the textual documents. Often critical information related to drug-food interaction or drug-activity interaction are scattered through the document and patients face difficulties in finding them. Also, the structure and organization of information varies from one source to another.

Such irregular organization and structural variety across sources pose a challenge to automatically extract critical information from the DUG documents. Therefore, there is a need for an annotated corpus for DUG documents. Currently, there are several annotated medical corpora (Saeed et al., 2011), (Aronson and Lang, 2010), (Uzuner et al., 2007), (Pardelli et al., 2012), (Bongelli et al., 2012) that contain clinical notes, bio-medical textual contents, and electronic health records. There are some existing annotation tools and techniques than focus on annotating electronic health records (Roberts and Demner-Fushman, 2016) and clinical practice guidelines (Read et al., 2016) and extracting relations from bio-medical text (Ellendorff et al., 2014). But to the best of our knowledge, there is no existing work that focuses on annotating and analyzing the textual content of such documents. So, we introduce a DUG data annotation scheme, an annotation tool, and an annotated corpus as presented in the following sections.

3. Data Annotation Scheme

While a DUG document contains an array of information, not all information is critical to patients (Yi et al., 2015). We develop the following multi-label data annotation scheme to annotate the critical advice statements of DUG documents in 8 categories.

1. **Activity or lifestyle related advice**: to indicate potential interaction between the corresponding drug and any activity of daily living (e.g., driving). For instance, from Table 1, driving or performing other activities that require alertness might cause fatal accident if the person took Ambien within a day.

2. **Disease or symptoms related advice**: to indicate potential interaction between the corresponding drug and any diseases / symptoms. Such advice statements are crucial for patients suffering from multiple diseases, as multiple diseases can often be conflicting (Kienhues et al., 2011), (Caughey et al., 2013).

3. **Drug administration related advice**: to annotate important advice related to drug administration process, e.g., how the drug should be taken. This type of advice are essential for medication adherence and effective treatment. For example, Topomax should not be taken within 6 hours of drinking alcohol (Table 1).

4. **Exercise related advice**: to indicate potential interaction between the corresponding drug and any exercise (e.g., running outdoors in a hot weather). As exercise is often suggested to patients taking prescription drugs, annotating the special circumstances when (i) exercise should be limited or avoided in certain context or (iii) performed with certain preparation (e.g., checking blood sugar before exercising) is necessary to avoid adverse conditions.

5. **Food or beverage related advice**: to indicate potential interaction between the corresponding drug and any food / beverage. This is critical to health safety. For example, in the text excerpt from DUG document of drug Warfarin presented in Section 1., patients are suggested to avoid dark green vegetables as they can interact with the drug. But these vegetables are widely known as healthy food and often suggested to people for weight loss. High weight is a significant factor of deep vein thrombosis, a disease for which Warfarin is suggested. So, this particular food related advice is critical for the patients who are prescribed Warfarin.

6. **Other drug related advice**: to indicate the advice suggest avoiding / limiting consumption of other drugs. As shown in Table 1, when taking Abilify, certain prescription and over-the-counter drugs (e.g., allergy or cough relief products) should be taken with caution, as they can increase the effect of drowsiness.

7. **Pregnancy related advice**: to indicate whether the advice is for women who are already pregnant, are planning to conceive, or recently gave birth. This advice is critical to pregnant women, nursing mother, and children. Also, as pregnancy related advice statements are applicable for only a small portion of the patient pool, this annotated advice can be filtered according to the personal condition of a patient.

Drug Name	Advice Text	Annotation
Ambien	Do not **drive, use machinery, or do any activities** that require clear thinking after you take this medication and the next day. You may feel alert, but this medication may continue to affect your thinking, making such activities unsafe.	Activity or lifestyle related
Zoloft	Older adults may also be more likely to develop a type of **salt imbalance** (**hyponatremia**), especially if they are taking "water pills" (**diuretics**).	Disease or symptom related Other drug related
Fentanyl	Avoid activities that might cause your body temperature to rise. Such as doing **strenuous work/exercise** in hot weather.	Exercise related
Topamax	not drink **alcoholic beverages for 6 hours before or 6 hours after taking Topamax** extended release capsules, since alcohol may affect this medication works.	Temporal Food or beverage related Drug administration related
Abilify	Ask your pharmacist about using **allergy** or **cough-and-cold products** because they may contain ingredients that cause drowsiness.	Other drug related
Ativan	This medication is not recommended for use during **pregnancy**. It may harm an unborn baby.	Pregnancy related

Table 1: Different types of advice extracted from the online DUG data. The first, second, and third columns contain the name of the drug, an advice statement from the DUG document of that drug, and the annotation of that advice statement, respectively. An advice statement can have multiple tags based on its topics.

8. **Temporal advice**: to indicate the advice suggests an action with temporal condition(s), e.g., when to take a drug, for how long to wait before eating/drink after taking the drug. It suggests duration or frequency of drug usage, and interval between consecutive dosage. Also, it denotes temporal dependency between taking a drug and one of the following events: (i) having a meal, (ii) doing an activity, and (iii) exercising. Such as, as shown in Table 1, when taking Topomax, alcohol should not be consumed with in 6 hours.

Although a DUG document contains a few other categories of advice (e.g., drug dosage, route of drug administration, drug storage, drug disposal), those categories are not considered here. Because, existing studies find patients are educated by their primary health care providers on these types of advice (Yi et al., 2015) when applicable. So, this information can be either filtered out or presented with lower priority. This will reduce the document length and the cognitive overload of the patients (Savas and Evcik, 2000), (Shrank and Avorn, 2007).

4. Data Annotation Tool

Our goal is to develop a drug usage guideline (DUG) document annotation tool that is **interactive** and **generalizable** to DUG data from different sources. Based on empirical evidence found in the DUG documents from different sources, the tool should address following issues:
(i) The tool should parse through the DUG document sentence by sentence, as critical information can be found in different parts of the document. (ii) It should provide the option to annotate advice that spans across multiple sentences. (iii) It should be flexible so that an annotator can add multiple tags to an advice. (iv) It should be interactive so that an annotator can change previous annotations based on his new observation as he/she goes through the document. (v) As the DUG documents often contain redundant headers or text fragments, the tool should allow the annotator to select part of text as advice.

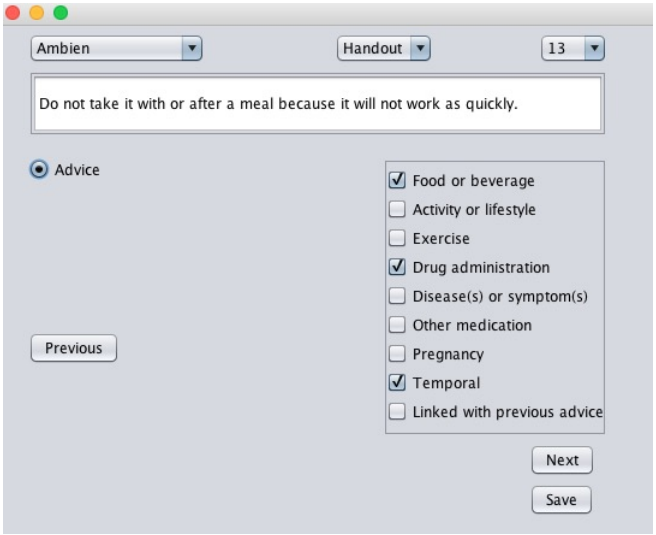


Figure 1: The Data Annotation Tool for annotating advice statements in DUG according to their topics. Here, an annotator tagged the 13-th sentence of the DUG document of the drug named Ambien as advice. He also annotated three topics for the current sentence, namely, food or beverage, drug administration, and temporal.

We have developed a desktop application for annotating DUG documents based on the annotation scheme described in Section 3. that addresses the above mentioned issues. This tool is developed using Java Swing library. A screen shot of this tool is shown in Figure 1. The input of this tool is the textual content of the DUG documents. The output is the annotation of each sentence of the DUG document. The main features of this tool are described below.

An annotator can select a drug to start annotating its DUG document. The tool supports annotating multiple files for the same drug, as often there are multiple overlapping sources of drug usage guidelines for a single drug. It can

Type of Advice	Count	%Gold Label
Activity or lifestyle related	146	100
Disease or symptom related	245	97.5
Drug administration related	224	98.2
Exercise related	40	97.9
Food or beverage related	253	99.6
Other drug related	310	100
Pregnancy related	211	99
Temporal	182	99.45

Table 2: Annotation of Drug usage guideline dataset. Eight types of advice are annotated in the data as shown in column 1. The second column denotes the count of advice for each type of advice. The third column contains the % of advice statements of that received gold label in annotation.

be selected from the drop down menu named *Handout* in Figure 1. The annotator can go to a specific line of the current DUG document. An annotator can tag a sentence as an advice and specify the categories of the advice statements. Also, a sentence can be annotated as an advice without specifying the exact category of the advice. When an advice statement consists of multiple consecutive sentences, it can be annotated using the *linked with previous* option. An annotator can browse forward and backward through a document and update the annotation of each sentence.

Also, the DUG documents often contain section headers, titles, and formatted text in between sentences that adds redundant text fragments in advice sentence(s). This annotation tool allows the annotators to select portion of text as advice and ignore the rest of the text. As it is unlikely for an annotator to annotate the whole corpus at a time, the tool supports session memory, i.e., once the application is relaunched it loads the data from the most recent position of the corpus.

This tool can be easily adapted to annotate more categories of advice. It can also be generalized to annotate DUG documents from different sources and other descriptive health / clinical textual documents (e.g., health articles, websites).

5. Annotated Corpus

The data collection is motivated by self management of patients suffering from multiple chronic diseases. We sampled 34 anonymized prescriptions from MTSamples. This dataset contains anonymized prescriptions of real patients. Each of the sampled prescriptions represents an anonymous patient who is suffering from two or more chronic diseases. The chronic diseases covered in the sampled prescriptions include the most common chronic diseases, e.g., diabetes, hypothyroidism, bipolar affective disorder, alcohol withdrawal, anxiety, depression, lethargy, alcohol dependence, substance abuse, obesity, chronic pain, chronic kidney disease, and coronary vascular disease. Each prescription contains a list of suggested drugs and their corresponding dosages.

From the 34 prescriptions, a total of 166 drugs are found. For each of these drugs, we crawled online drug usage guidelines (DUG) documents from MedScape. We chose MedScape as it is one of the most widely used applications by the physicians. It contains more comprehensive DUG documents when compared to FDA drug database or WebMD DUG documents. Among the 166 drugs, the online drug usage guideline document is available for only 90 drugs in MedScape. We crawled and annotated these 90 online DUG documents.

The collected DUG document corpus contains 9,831 sentences and 170,646 words. It is annotated by three human annotators using the data annotation tool presented in Section 4. For each advice, they also tag the potential categories of the advice based on the annotation scheme presented in Section 3. Each of the annotators have at least a masters degree in computer science. Majority voting is applied to decide the ground truth of annotation. The result of this annotation is presented in Table 2. There are 1,637 advice statements in eight categories. Here, majority of the advice received gold label, i.e., all three annotators agree on the annotation of the advice.

Among the eight categories of our proposed scheme, *other drug related* advice is the most common category (n=336). Advice from *food and beverage related* categories are also common (n=253), as most of the drugs in our constructed corpus interact negatively with alcoholic beverages. As our corpus contains drugs that are used for treating chronic diseases, a major portion of the annotated advice (n=245) describes how a drug can interact with *other diseases* or *cause physiological / psychological syndromes*. *Exercise related advice* are relatively rare (n=40), as exercise is often recommended for most of the chronic diseases. However, it is found from the annotation that exercise in certain contexts (e.g., in a hot or humid weather, within certain time range of drug administration) can negatively affect well-being. Some of the categories show strong correlation, e.g., *drug administration related* advice statements are often *temporal*, and related to *food/beverage, activity* (e.g., sleeping, driving), and *exercise*.

This dataset or the extension of such dataset can be used to automatically extract personalized advice from patient handout or DUG documents to raise patient's awareness and hereby increase medication adherence and effectiveness of treatment. Some potential applications to utilize such datasets are presented below.

Firstly, a medication reminder app can present personalized safety critical drug related advice while reminding the patient to take a medication. Such as, reminding safety-critical pregnancy related advice from a DUG document to only women of child bearing age. As often DUG documents contain numerous advice, filtering them in a personalized manner and prioritizing them according to severity can increase patient's adherence.

Secondly, another application is detecting conflicts between advice from the DUG documents with other health related advice (Preum et al., 2017a; Preum et al., 2017b). Such as, Aspirin is suggested to take with food or milk to avoid stomach upset. On the other hand, an individual with lactose intolerance is also suggested to avoid milk. In this case, there is a conflict between these two advice statements. Such datasets can accelerate automatic conflict detection from drug related advice.

Finally, presenting drug administration related advice in a

context-aware manner can increase effectiveness of treatment. Often drugs are suggested to take within a certain interval of other activities (e.g., one hour after meal, 2 hours before sleep). Individuals suffering from multiple conditions may not be aware of all such temporal, drug administration related advice, as they are often prescribed several drugs. In such cases, presenting relevant drug related advice by inferring the context of their daily lives can be beneficial. Such as, for a drug that should be taken two hours before sleep, an activity tracking app can suggest it's user to take the drug two hours before her frequent bed time.

6. Conclusion

A plethora of textual documents containing crucial information on drug usage guidelines (DUG) are available online. Although analyzing such textual document can aid patient education and promote safe usage of drugs, these resources are underutilized. To bridge this knowledge gap, we introduce a multi-label annotation scheme to annotate advice from the DUG documents in eight categories based on their topics. We develop an interactive data annotation tool for this data that can also be generalized to annotated advice from various other descriptive textual information sources (e.g., DUG documents from other sources, health article). Finally, we share the first annotated corpus on DUG data containing annotated drug usage guidelines of 90 drugs that are used to treat over 30 chronic diseases. The corpus contains 9,831 sentences and 1,611 advice statements on eight safety critical categories. The corpus yields several important insights on instructions regarding safe usage of drugs. The annotation tool and the annotated corpus can aid future research to automatically annotate / classify critical information from DUG documents as well as other textual health documents.

7. Acknowledgements

This work was supported, in part, by NSF grant CPS-1646470.

8. Bibliographical References

Aronson, A. R. and Lang, F.-M. (2010). An overview of metamap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236.

Bongelli, R., Canestrari, C., Riccioni, I., Zuczkowski, A., Buldorini, C., Pietrobon, R., Lavelli, A., and Magnini, B. (2012). A corpus of scientific biomedical texts spanning over 168 years annotated for uncertainty. In *LREC*, pages 2009–2014.

Caughey, G., Gilbert, A., Roughead, L., McDermott, R., Ryan, P., Esterman, A., et al. (2013). Multiple chronic health conditions in older people. *Implications for health policy planning, practitioners and patients*.

Ellendorff, T., Rinaldi, F., and Clematide, S. (2014). Using large biomedical databases as gold annotations for automatic relation extraction. In *LREC*, pages 3736–3741.

(2017). Medication guides: Us food and drug administration. `https://www.fda.gov/Drugs/DrugSafety/ucm085729.htm`. Accessed: 2017-04-15.

Jimmy, B. and Jose, J. (2011). Patient medication adherence: measures in daily practice. *Oman medical journal*, 26(3):155.

Kienhues, D., Stadtler, M., and Bromme, R. (2011). Dealing with conflicting or consistent medical information on the web: When expert information breeds laypersons' doubts about experts. *Learning and Instruction*, 21(2):193–204.

Liddy, C., Blazkho, V., and Mill, K. (2014). Challenges of self-management when living with multiple chronic conditions. *Canadian Family Physician*, 60(12):1123–1133.

(2017). Medscape: Search drugs, otcs & herbals. `http://reference.medscape.com/drugs`. Accessed: 2017-04-15.

Pardelli, G., Sassi, M., Goggi, S., and Biagioni, S. (2012). From medical language processing to bionlp domain. In *LREC*, pages 2049–2055.

Patel, S. and Dowse, R. (2015). Understanding the medicines information-seeking behaviour and information needs of south african long-term patients with limited literacy skills. *Health Expectations*, 18(5):1494–1507.

Preum, S. M., Mondol, A. S., Ma, M., Wang, H., and Stankovic, J. A. (2017a). Preclude: Conflict detection in textual health advice. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*, pages 286–296. IEEE.

Preum, S. M., Mondol, A. S., Ma, M., Wang, H., and Stankovic, J. A. (2017b). Preclude2: Personalized conflict detection in heterogeneous health applications. *Pervasive and Mobile Computing*.

Preum, S. M., Parvez, M. R., Chang, K.-W., and Stankovic, J. A. (2018). A Corpus of Online Drug Usage Guideline Documents Annotated with Type of Advice. Available at `https://doi.org/10.5281/zenodo.1173345`.

Read, J., Velldal, E., Cavazza, M., and Georg, G. (2016). A corpus of clinical practice guidelines annotated with the importance of recommendations.

Roberts, K. and Demner-Fushman, D. (2016). Annotating logical forms for ehr questions. In *International Conference on Language Resources and Evaluation*, volume 2016, page 3772. NIH Public Access.

Saeed, M., Villarroel, M., Reisner, A. T., Clifford, G., Lehman, L.-W., Moody, G., Heldt, T., Kyaw, T. H., Moody, B., and Mark, R. G. (2011). Multiparameter intelligent monitoring in intensive care ii (mimic-ii): a public-access intensive care unit database. *Critical care medicine*, 39(5):952.

Savas, S. and Evcik, D. (2000). Do undereducated patients read and understand written education materials? a pilot study in isparta, turkey. *Scandinavian journal of rheumatology*, 30(2):99–102.

Shrank, W. H. and Avorn, J. (2007). Educating patients about their medications: the potential and limitations of written drug information. *Health Affairs*, 26(3):731–740.

Tang, F., Zhu, G., Jiao, Z., Ma, C., Chen, N., and Wang,

B. (2014). The effects of medication education and behavioral intervention on chinese patients with epilepsy. *Epilepsy & Behavior*, 37:157–164.

Uzuner, Ö., Luo, Y., and Szolovits, P. (2007). Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

(2017). Webmd: Drugs medications a-z. `http://www.webmd.com/drugs/2/index`. Accessed: 2017-04-15.

Wolf, M. S., Davis, T. C., Tilson, H. H., Bass III, P. F., and Parker, R. M. (2006). Misunderstanding of prescription drug warning labels among patients with low literacy. *American Journal of Health-System Pharmacy*, 63(11).

Woolley, K. R. (2015). Enhancing education of medication side effects to improve patient outcomes.

Yi, Z.-M., Zhi, X.-J., Yang, L., Sun, S.-S., Zhang, Z., Sun, Z.-M., and Zhai, S.-D. (2015). Identify practice gaps in medication education through surveys to patients and physicians. *Patient preference and adherence*, 9:1423.

BioRo: The Biomedical Corpus for the Romanian Language

Maria Mitrofan and Dan Tufiş
Research Institute for AI "Mihai Drăgănescu", Romanian Academy
"Calea 13 Septembrie", Bucharest 050711, Romania
{maria, tufis}@racai.ro

Abstract

The biomedical domain provides a large amount of linguistic resources usable for biomedical text mining. While most of the resources used in biomedical Natural Language Processing are available for English, for other languages including Romanian the access to language resources is not straight-forward. In this paper, we present the biomedical corpus of the Romanian language, which is a valuable linguistic asset for biomedical text mining. This corpus was collected in the contexts of CoRoLa project, the reference corpus for the contemporary Romanian language. We also provide informative statistics about the corpus, a description of the data-composition. The annotation process of the corpus is also presented. Furthermore, we present the fraction of the corpus which will be made publicly available to the community without copyright restrictions.

Keywords: BioNLP; Romanian; Corpora; Specialized Domain;

1. Introduction

Natural language processing (NLP) technologies have been used to extract useful information from different types of biomedical texts, such as: biomedical literature, research papers, medical school lecture notes, clinical notes, discharge summaries, clinical practice guidelines, etc.

In order to be able to obtain relevant knowledge form textual data, high-quality resources (i.e. corpora, lexica, terminologies, tesauri etc.) are needed. At international level various biomedical textual resources have been developed, but most of them are available in English (e.g.GENIA corpus (Kim et al., 2003), AnEM corpus (Ohta et al., 2012), CellFinder corpus (Neves et al., 2012), etc.).

In many countries where the official language is not English, there is a technical barrier in using Natural Language Processing in the biomedical domain (bioNLP) due to the fact that textual resources are more scarce. Nevertheless, a significant progress has been made over the past decades thanks to the contribution of various active NLP communities.

For example, for French the language Zweigenbaum et al. (2005) created the "Unified Medical Lexicon for French" (UMLF), a reference resource for bioNLP. Another important linguistic resource is Corpus Medical du Centre de Recherche en Terminologie et Traduction (CMCRTT), which is a monolingual French corpus for bioNLP and it is publicly available both as plain or POS tagged text (Neveol et al., 2014).

For the Bulgarian language important efforts have been made in developing resources usable for various NLP tasks. Boytcheva et al. (2009) collected a biomedical corpus containing 6400 words, 2000 of them are part of the Bulgarian medical terminology.

Swedish is another language which has important resources for bioNLP. In 2012, Velupillai (2012) created an annotated gold standard of medical records, used for terminology management and linguistic explorations. Also, a negation and clinical uncertainty taxonomy schema was proposed for Swedish language and was mapped to an English annotation schema (Mowery et al., 2012).

The Romanian language is an under-resourced language, regarding resources available for bioNLP. Therefore, considerable efforts are carried on in order to improve the availability of the Romanian biomedical resources usable for bioNLP. At this moment the most important project is the CoRoLa project (Mititelu et al., 2018), started in 2012 by the Romanian Academy Research Institute for Artificial Intelligence "Mihai Drăgănescu" (RACAI) and the Institute for Computer Science in Iaşi. This is an on-going project that aims to create a reference corpus of contemporary Romanian (from 1945 onwards). In the context of the COROLA project we created an important biomedical corpus for the Romanian language (BioRo) that can be used for different bioNLP tasks.

Developing a corpus for a specialized domain, in this case for the biomedical domain, is not an easy task and there are several steps that need to be taken: defining the structure and the linguistic coverage of the corpus, collecting the texts following the established structure, addressing copyright problems, processing the corpus with the NLP technologies available (segmentation, tokenization, lemmatization, tagging etc.) and establishing the availability of the data.

In what follows, the structure of the corpus is presented, describing the process of acquiring it and the pre-processing steps (Section 2.). Section 3. describes the statistics of the corpus. Section 4. presents the processing tool used for annotation and the results obtained after the annotation step. The availability of the data is presented in section 5.. In section 6. we draw the conclusions and present the future work.

2. BioRo Corpus Structure

BioRo corpus contains, excluding the punctuation, 9,864,707 tokens distributed in different medical subdomains such as: diabetes, endocrinology, cardiology, oncology, neurology etc. (Table 1). All the texts are tokenized, lemmatized and morpho-syntactically tagged.

2.1. Collecting the Data

The process of collecting the texts was a difficult task, because the laws of intellectual property are very restrictive and also because, in general, most of the biomedical literature is not published in the Romanian language. The main providers of medical texts contained in BioRo corpus are: the Romanian Academy Publishing House, Polirom publishing house, PIM publishing house, Timpul publishing house, the Romanian Medical Journal, medical blogs and medical school lecture notes.

2.2. Cleaning the Data

Initially, the textual resources contained in BioRo corpus were available in various formats such as unprotected .pdf and .doc. All the texts have been converted into a raw text format which fits to our processing tools (Tufis et al., 2008). The conversion of the medical files included a boilerplate removal phase in which all the figures, tables, headers, footers, etc. were removed. The non-standard codes for diacritics have been replaced with the proper ones while the missing diacritics have been inserted automatically.

Moreover in order to prepare the corpus for the processing step we decided to correct various types of misspellings such as: missing letters "diabe" instead of "diabet" (en. "diabetes"), extra spaces "cardio logie" instead of "cardiologie" (en. "cardiology"), etc. Nevertheless, after this cleaning phase (given that automatic procedures are not error-free), roughly 3-4% of the words still need to be corrected.

2.3. Metadata Creation

All the files contained in the corpus have an associated XML file representing the corresponding metadata scheme. Each XML file contains specific information at the document level about source, author, genre and type of the text, etc. Most of the information contained in the metadata scheme is relevant for the indexing of the corpus. All 823 files have manually created metadata descriptors, each being created following the metadata scheme used in CoRoLa (Figure 1).

```
<?xml version="1.0" encoding="UTF-8"?>
    <Metadata>
        <DocumentTitle>Chirurgie generală.
        </DocumentTitle>
        <ArticleTitle>POLITRAUMATISMELE</ArticleTitle>
        <AuthorName>Dr. Marius Bârza</AuthorName>
        <TranslatorName>-</TranslatorName>
        <PublicationDate>2008</PublicationDate>
        <Source>Publishing House</Source>
        <SourceName>Timpul</SourceName>
        <Medium>Written</Medium>
        <DocumentType>inBook</DocumentType>
        <DocumentTextStyle>Science</DocumentTextStyle>
        <DocumentTextDomain>Science</DocumentTextDomain>
        <DocumentTextSubDomain>Medicine
        </DocumentTextSubDomain>
        <SubjectLanguage>Romanian</SubjectLanguage>
        <ISSN-ISBN>973-612-020-1</ISSN-ISBN>
        <CollectionDate>2015</CollectionDate>
    </Metadata>
```

Figure 1: A example of metadata scheme.

3. BioRo Corpus Statistics

In order to present general statistics about the biomedical corpus, we counted all the sentences, unique lemmas, tokens and content words (nouns, main verbs, adjective and adverbs) (Table 1). The punctuation is obtained by subtracting the words count from the tokens count. Figure 2 indicates the percentage of tokens for each biomedical subdomain contained in the corpus. In Figure 2 it is also shown that the biomedical corpus has an unbalanced distribution regarding the number of tokens for each biomedical subdomain, but this comes from the fact that in the process of obtaining the data there are many difficulties, due to the copyright restrictions.

# Tokens	9,864,707
# Unique lemmas	237,620
# Punctuation	1,498,218
# Sentences	561,978
Tokens per sentence	17.55
Punctuation per sentence	2.66

Table 1: General statistics over the biomedical corpus

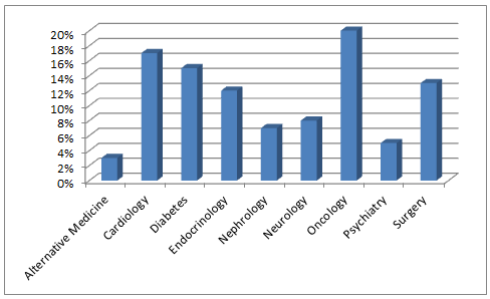


Figure 2: The distribution of the medical sub-domains in the corpus.

A very important component of the BioRo corpus is represented by the texts obtained from online sources (322,005 tokens and 18,226 sentences), free of copyright restriction and which will be freely available to the community.
In order to facilitate cross-linguistic comparison, each file included in this part of the BioRo corpus have another metadata scheme, enhanced with standard categories used in the biomedical corpora (Figure 3). The medical categories are extracted from the Medical Subject Headings(MeSH) [1] thesaurus. Table 2 describes the sub-corpus differentiating by biomedical sub-domains.

4. Corpus Annotation

The entire corpus was subjected to an annotation phase using the Tokenizing, Tagging and Lemmatizing (TTL) text processing platform developed at RACAI (Ion, 2007).
TTL is a Perl platform supporting different languages such as Romanian, English and French, and performs the following functionalities: named entity recognition (NER), sentence splitting, tokenization, POS tagging and chunking.

[1] https://meshb.nlm.nih.gov/treeView

	Sentences	Tokens	Content words	Unique lemmas	Punctuation
Neurology	7,632	78,901	44,739	8,940	15,973
Diabetes	5,736	124,035	66,401	9,102	17,916
Endocrinology	1,840	40,038	21,514	4,928	5,915
Cardiology	1,235	33,674	18,782	4,100	4,355
Oncology	1,127	28,286	15,902	3,963	3,587
Nephrology	536	13,753	7,590	2,449	1,977
Alternative medicine	120	3,318	1,774	872	336
Total	**18,226**	**322,005**	**176,702**	**34,354**	**50,059**

Table 2: Statistics over the biomedical sub-corpus extracted from online sources.

```
<Metadata>
<DocumentTitle>Diabetul de tip 2</DocumentTitle>
<AuthorName></AuthorName>
<PublicationDate></PublicationDate>
<Source>http://pentrudiabet.ro/</Source>
<SourceName></SourceName>
<TranslatorName>-</TranslatorName>
<Medium>Written</Medium>
<DocumentTextStyle>Science</DocumentTextStyle>
<DocumentTextDomain>Science</DocumentTextDomain>
<DocumentTextSubDomain>Medicine</DocumentTextSubDomain>
<DocumentTextSubDomainCategory>Diseases
</DocumentTextSubDomainCategory>
<DocumentTextSubDomainSubCategory>
Endocrine System Diseases
</DocumentTextSubDomainSubCategory>
<DocumentTextSubDomainSubCategoryField>
Diabetes Mellitus
</DocumentTextSubDomainSubCategoryField>
<CollectionDate>2016</CollectionDate>
<SubjectLanguage>Romanian</SubjectLanguage>
<ISSN-ISBN></ISSN-ISBN>
</Metadata>
```

Figure 3: A example of metadata scheme for biomedical domain.

TTL's tokenizer is language independent, recognizes multiword expressions (MWEs), clitics and contractions, assuming that language-dependent resources are available. The POS tagger uses tiered tagging methods (Tufis, 1999) and it is a reimplementation of the Hidden Markov Models (HMM) tagger described in Brants (2000). For the Romanian language the MSD tagset has 614 (Boros et al., 2013) labels compatible with the MULTEXT-East morpho-lexical specifications [2]. External annotations compatible with the Universal Dependencies format can also be generated using an existing multilingual platform from text processing which is available online[3] (Dumitrescu et al., 2017).

After the completion of the POS tagging step the lemmatization begins and a human-validated Romanian word-form lexicon with almost 1,200,000 entries is used by the TTL lemmatizer. In the case of out-of-dictionary words, the lemmatizer uses a five-gram letter Markov Model-based guesser to select the most probable lemma.

Another functionality performed by the TTL platform is chunking. This process is guides by a set of rules based on regular expressions applied on MSDs. The TTL chunker deals with recognizing nominal, verbal, adjectival, adverbial and prepositional phrases.

The biomedical corpus was annotated with the baseline TTL model, which is trained over texts corrected by trained linguists at word-level. The POS tagging accuracy for the general purpose Romanian language is over 98% (Tufis, 1999), and for the biomedical domain the accuracy is 97.83% (Mitrofan and Ion, 2017).

Following the automatic annotation step all the tokens included in the corpus the lemmas and POS tagging were submitted to a partial manual revision. The main clear cases of errors produces by the tagger were wrong lemmas and wrong POS-tags. Table 3 describes the distribution of content words according to the POS tags types found in the sub-corpus extracted from online sources. A important feature of the biomedical domain, also present in table 3, is a higher frequency of nouns and adjectives.

5. The Availability of the Data

The entire BioRo corpus, part of CoRoLa corpus is available for query via KorAP interface (Diewald et al., 2016; Banski et al., 2014; Banski et al., 2013). The search results will be downloadable (this facility, taking into account the IPR restrictions, is under development in KorAP). The KorAP interface allows (among other things) building virtual corpora by observing IPR restrictions - if any, multiple types of linguistic interrogations, various levels of annotation etc. The sub-corpus obtained from online sources (322,005 tokens and 18,226 sentences) will be freely available for download [4] and non-commercial use. The sub-corpus will be accessible in both raw text and annotated formats.

6. Conclusion and Future Work

We presented the BioRo corpus which contains morpho-syntactically annotation. We described the corpus and the annotation process. To our knowledge this is the first biomedical corpus for the Romanian language compiled for biomedical text mining. Although CoRoLa corpus, thus BioRo, is not downloadable, an important part of BioRo is free for non-commercial use.

We plan to annotate it with biomedical named entities and to parse it with the Romanian Universal Dependencies parser developed in the SSPR project (Mititelu et al., 2016). Also we are in the process of creating a tool which is able to automatically label text with BioNER entities, which is trained on the described corpora. We have experimented with multiple strategies (Boros, 2013; Boroş et al., 2017;

[2] http://nl.ijs.si/ME/V4/msd/html/

[3] http://slp.racai.ro/index.php/mlpla-new/

[4] http://slp.racai.ro/index.php/resources/

	Nouns	Verbs	Adjectives	Adverbs	Total
Neurology	26,514	6,489	10,291	1,445	44,739
Diabetes	38,463	13,459	11,667	2,812	66,401
Endocrinology	11,946	4,545	4,240	783	21,514
Cardiology	10,706	3,444	4,001	631	18,782
Oncology	8,538	3,261	3,629	474	15,902
Nephrology	4,357	1,332	1,604	297	7,590
Alternative medicine	885	475	331	83	1,774

Table 3: Statistics over the biomedical sub-corpus extracted from online sources.

Boros and Dumitrescu, 2018) and we are currently turning toward graph-based decoding of named entities.

7. Bibliographical References

Banski, P., Bingel, J., N.Diewald, Frick, E., Hanl, M., Kupietz, M., Pezik, P., Schnober, C., and Witt, A. (2013). The new corpus analysis platform at ids mannheim. In *Proceedings of the 6th Language and Technology Conference: LTC'13. 2013.*

Banski, P., Diewald, N., Hanl, M., Kupietz, M., and Witt, A. (2014). Access control by query rewriting. the case of korap. In *Proceedings of the Ninth Conference on International Language Resources and Evaluation.*, pages 3817–3822.

Boros, T. and Dumitrescu, S. D. (2018). Multilingual tokenization and part-of-speech tagging. lightweight versus heavyweight algorithms. *Lecture Notes in Artificial Intelligence*, Human Language Technology. Challenges for Computer Science and Linguistics.

Boros, T., Ion, R., and Tufis, D. (2013). Large tagset labeling using feed forward neural networks. case study on romanian language. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 692–700.

Boroş, T., Pipa, S., Mititelu, V. B., and Tufiş, D. (2017). A data-driven approach to verbal multiword expression detection. parseme shared task system description paper. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 121–126.

Boros, T. (2013). A unified lexical processing framework based on the margin infused relaxed algorithm. a case study on the romanian language. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 91–97.

Boytcheva, S., Nikolova, I., Paskaleva, E., Angelova, G., Tcharaktchiev, D., and Dimitrova, N. (2009). Extraction and exploration of correlations in patient status data. in proceedings of the workshop on biomedical information extraction. In *Proceedings of the Workshop on Biomedical Information Extraction*, pages 1–7.

Brants, T. (2000). Tnt: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on Applied natural language processing*, pages 224–231.

Diewald, N., Margaretha, M. H. E., Bingel, J., Kupietz, M., Banski, P., and Witt, A. (2016). Korap architecture diving in the deep sea of corpus data. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 3586–3591.

Dumitrescu, S. D., Boroş, T., and Tufiş, D. (2017). Racai's natural language processing pipeline for universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 174–181, Vancouver, Canada, August. Association for Computational Linguistics.

Ion, R. (2007). *Word Sense Disambiguation Methods Applied to English and Romanian (in Romanian)*. Ph.D. thesis, Romanian Academy.

Kim, J., Ohta, T., Tateisi, Y., and Tsujii, J. I. (2003). Genia corpus - a semantically annotated corpus for biotextmining. *Bioinformatics*.

Mititelu, V. B., Ion, R., Simionescu, R., Scutelnicu, A., and Irimia, E. (2016). Improving parsing using morpho-syntactic and semantic information.

Mititelu, V. B., Tufis, D., and Irimia, E. (2018). The reference corpus of the contemporary romanian language (corola). In *Proceedings of the 11th Language Resources and Evaluation Conference-LREC*, page In This Volume.

Mitrofan, M. and Ion, R. (2017). Adapting the ttl romanian pos tagger to the biomedical domain. In *Proceedings of the Biomedical NLP Workshop associated with RANLP 2017*, pages 8–14.

Mowery, D. L., Velupillai, S., and Chapman, W. W. (2012). Medical diagnosis lost in translation: analysis of uncertainty and negation expressions in english and swedish clinical texts. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing. Association for Computational Linguistics*.

Neveol, A., Grosjean, J., Darmoni, S. J., and Zweigenbaum, P. (2014). Language resources for french in the biomedical domain. In *LREC*, pages 2146–2151.

Neves, M., Damaschun, A., Kurtz, A., and Leser, U. (2012). Annotating and evaluating text for stem cell research. In *Proceedings of the Third Workshop on Building and Evaluation Resources for Biomedical Text Mining (BioTxtM 2012)*.

Ohta, T., Pyysalo, S., Tsujii, J., and Ananiadou, S. (2012). Open-domain anatomical entity mention detection. In *Proceedings of ACL 2012 Workshop on Detecting Structure in Scholarly Discourse (DSSD)*, pages 27–36.

Tufis, D., Ion, R., Ceausu, A., and Stefanescu, D. (2008). Racai's linguistic web services. In *Proceedings of the 6th Language Resources and Evaluation Conference-LREC*, pages 28–30.

Tufis, D. (1999). *Tiered tagging and combined language models classifiers.* Springer.

Velupillai, S. (2012). *Shades of certainty: annotation and classification of swedish medical records.*

Zweigenbaum, P., Baud, R., Burgun, A., Namer, F., Jarrousse, Ã., Grabar, N., and Darmoni, S. (2005). Umlf: a unified medical lexicon for french. *International Journal of Medical Informatics*, pages 119–124.

A Comparison Of Emotion Annotation Schemes And A New Annotated Data Set

Ian D. Wood, John P. McCrae, Vladimir Andryushechkin, Paul Buitelaar
Insight Centre for Data Analytics, National University of Ireland Galway
name.surname@insight-centre.org

Abstract

While the recognition of positive/negative sentiment in text is an established task with many standard data sets and well developed methodologies, the recognition of more nuanced affect has received less attention, and in particular, there are very few publicly available gold standard annotated resources. To address this lack, we present a series of emotion annotation studies on tweets culminating in a publicly available collection of 2,019 tweets with scores on four emotion dimensions: valence, arousal, dominance and surprise, following the emotion representation model identified by Fontaine et.al. (Fontaine et al., 2007). Further, we make a comparison of relative vs. absolute annotation schemes. We find improved annotator agreement with a relative annotation scheme (comparisons) on a dimensional emotion model over a categorical annotation scheme on Ekman's six basic emotions (Ekman et al., 1987), however when we compare inter-annotator agreement for comparisons with agreement for a rating scale annotation scheme (both with the same dimensional emotion model), we find improved inter-annotator agreement with rating scales, challenging a common belief that relative judgements are more reliable.

Keywords: emotion, annotation, annotator-agreement, social-media, affective-computing

1. Introduction

Beyond simple positive/negative sentiment, there are two main emotion representation schemes that have been used in automated emotion recognition research. The first posits the existence of "basic" emotions or emotion categories to which all other emotions belong, for example the six emotions categories identified by Ekman (Ekman et al., 1987): *joy*, *fear*, *anger*, *disgust*, *sadness* and *surprise*. The second envisages each emotional state as a point in a dimensional space, each dimension corresponding to a characteristic of the emotion, the most widely used model presenting three dimensions (Osgood et al., 1975; Russell and Mehrabian, 1977): *Valence* (also termed evaluation-pleasantness), *Dominance* (also termed potency-control) and *Arousal* (also termed activation-arousal, a level of physical energy and action tendency). The question of which dimensions most effectively capture variation in emotional states was recently re-opened by Fontaine et.al. (2007), who identified those same three dimensions (though with higher importance attributed to *Dominance*) with one extra: *Surprise* or unpredictability.

Research effort in the recognition of affect in text has focussed to a large extent on recognition of positive/negative sentiment, while more nuanced emotion representation models have received relatively little attention. In particular, there has been a lack of quality annotated resources for model building and evaluation in that space (Mohammad, 2016) and in particular with dimensional annotations. Existing text corpora with dimensional emotion annotations include Affective Norms for English Texts (Bradley and Lang, 2007), a collection of 120 generic texts with VAD annotations; a collection of 2,895 Facebook posts annotated by just two annotators with valence and arousal dimensions (Preotiuc-Pietro et al., 2016). Yu et.al. (2016) presented a collection of 2009 Chinese sentences from various online texts, again annotated with valence and arousal only. Subsequent to our annotation efforts, several further annotated data sets have been published: EMOBANK (Buechel and Hahn, 2017), a collection of ten thousand texts from diverse sources, but not including tweets, and data for the upcoming "Affect in Tweets" task for SemEval 2018[1] which presents tweets annotated for valence, arousal and dominance in English, Spanish and Arabic. In addition, two recent data sets annotated with emotion intensity in Ekman emotion categories have also been released: data for the WASSA emotion intensity detection competition (Mohammad and Bravo-Marquez, 2017), 1,500 to 2,000 tweets for each of the four Ekman emotions joy, anger, sadness and fear; and further data from SemEval 2018.

Several approaches to annotating emotion expressed in text on a continuous scale have been used. Probably the most common utilises an ordinal scale, such as the SAM manikins (Bradley and Lang, 1994). It has been argued that human estimations of relative values are more consistent than when assigning an absolute value (Metallinou and Narayanan, 2013; Yannakakis et al., 2017). To address this, Martinez et.al. (2014) suggest that ranked annotations not be treated as absolute values, and instead treated as ordinal, and used, for example, to train ranking estimators. Another approach is to perform relative annotations directly, such as best/worst scaling, where the highest and lowest ranked tweets are chosen from a set of four (Kiritchenko and Mohammad, 2017). Pairwise tweet comparisons are another option, however we are not aware of this approach being used previously in the emotion annotation literature as it requires a large number of annotations to acquire a reasonable ranking.

In this work, we present a collection of 2,019 tweets annotated following the four dimensional emotion representation scheme of Fontaine et.al. (2007). We further assess the relative merits of annotations on a ranking scale vs. comparisons, providing annotations using both a 5 point rank-

[1] http://alt.qcri.org/semeval2018

ing scale and pairwise comparisons[2].

An initial pilot study with in-house volunteers was performed to compare pairwise tweet comparisons to best/worst scaling. We found that best/worst scaling required significantly more effort and indeed our annotators preferred the pairwise comparison option, so for the final annotation round, best/worst scaling was dropped.

In order to compare annotator agreement between disparate annotation schemes, novel annotation metrics were developed built on common principals of annotation discrepancy (see Section 4.1.). The pilot study and reanalysis of categorical annotation data from previous work (Wood and Ruder, 2016) indicated improved annotator agreement with a relative annotation scheme (tweet comparisons) on a dimensional emotion representation system compared to categorical annotations of Ekman's six basic emotions. Annotator agreement for final round pairwise comparisons was similar to that for the 5 point rating scale, and when considered as ordinal annotations and converted into pairwise comparisons, agreement was noticeably better. These results challenge the notion that relative human judgements are more reliable than absolute judgements.

As a further comparison of annotation schemes, our annotated data alongside two recently published data sets (Kiritchenko and Mohammad, 2017; Mohammad and Bravo-Marquez, 2017) were analysed for the cognitive complexity of the annotation tasks using the time taken by reliable crowd-sourced annotators as a proxy. We found that best/worst scaling (choosing the highest and lowest from a set of four examples) required between 3 and 3.5 times the effort per annotation item to both relative and absolute annotation schemes, which exhibited similar effort per item (with relative marginally higher).

The remainder of this paper is organised as follows: Section 2. describes the collected data and data cleaning strategies. Section 3. describes our annotation schemes and procedures. Section 4. presents novel annotation distance metrics that allow comparison of annotator agreement between the annotation schemes and the agreement scores obtained. Section 5. describes baseline further two evaluation efforts, comparisons of cognitive complexity (using time to annotate as a proxy) and predictive models of the published data sets. Section 6. concludes with an overview of the principle results.

2. Data Collection

For the pilot study, 30 distinct tweet pairs and 18 sets of four distinct tweets were chosen randomly from a collection of tweets drawn from the Twitter Sample API between November 14th 2015 and February 22nd 2016. This extended period of collection was intended to reduce the bias from trending topics with high tweet rates over short periods. Even so, there were many tweets related to the upcoming US election. Tweets containing URLs and those containing hashtags of the form "#Vote…" were removed. Through identification of particular tweets with low annotator agreement in the pilot study and common sense, we attempted to remove tweets that were not created by humans,

High proportion of very short words
High proportion #tags/@mentions
Multiple new-line characters
Large number of stop words
Apparent quotations
Weather channel tweets
Presence of obscure acronyms
Presence of "#Vote"
Presence of non English characters
Presence of "follow" or "followme" …

Table 1: Heuristics used to remove tweets.

were difficult to comprehend or were difficult to annotate for some other reason. We developed a set of heuristics to automatically remove many such tweets, and performed a further manual assessment of selected tweets to remove the small number of obvious problematic tweets that remained (see Table 1).

For the primary study, two thousand tweets were sampled randomly from Twitter streaming API output over two periods: 19 days from February 2nd 2016 and 14 days from July 15th 2016. The extended time period is intended to reduce biases caused by trending events which can dominate tweet generation over short periods of time. 19 tweet pairs from the pilot study that passed our heuristics (Table 1) were retained for the primary study.

3. Annotation

Annotations were obtained via the CrowdFlower platform. For the pilot study, volunteer annotators from our organisation were recruited, and for the scaled-up annotation, professional CrowdFlower annotators from English speaking countries were used. Each emotion dimension was treated as a separate task, such that an annotator would annotate their quota of tweets on one dimension, then their quota on another and so forth.

Annotation guidelines used more accessible terms than the traditional names for the first three emotion dimensions (Happy/Sad, Excited/Bored, Confident/Unsure respectively; the terms Surprise and Intensity were used as is) as well as everyday language to describe them, and stressed that it was the feeling experienced by the tweet *author* that was sought. Instructions included an explanation of the meaning of the relevant dimension including other emotion words associated with it as well as eight annotated examples with explanations. In all annotation schemes, a "can't tell" option was provided, and these annotations were not included in further analysis.

Pilot Study: We investigated two annotation schemes for comparing tweets along each of four emotion dimensions: Valence, Arousal, Potency/Dominance, Surprise and Intensity. This corresponds to the four dimensions identified by Fontaine et.al. (Fontaine et al., 2007) with the addition of emotion intensity. We found that intensity correlated strongly with activation and did not pursue it further in the primary study.

The first annotation scheme presented annotators with two tweets and asked them to identify which author's emotion was stronger in the respective emotion dimension. Options

[2] http://140.203.155.26/mixedemotions/datasets/4dEmotionInTweets.tar.gz

for "About the same" and "Can't tell" were also provided. It was stressed that "Can't tell" was to be used only when one or other of the tweets was unintelligible, in a language other than English, or it was impossible to form a clear picture of their authors emotional states. For this scheme there were 150 comparison tasks (30 tweet pairs in 5 emotion dimensions), 10 annotators took part in the study, each annotating between 30 and 150 tweet pairs.

The second annotation scheme used the Best-Worst approach (Louviere and Woodworth, 1991; Louviere et al., 2015), where annotators are presented with four texts and asked to choose the two texts where the authors were feeling the most and least positive/dominant/etc... Best-Worst annotations provide five out of the six possible comparisons between pairs of the presented texts, and thus are efficient at gathering information. Initial annotations with the Best-Worst approach indicated that the annotation tasks were cognitively very challenging and the time needed to annotate each tweet set was many times more than annotating a tweet pair, effectively counteracting the increased number of tweet comparisons per task. Our in-house annotators exhibited a clear preference for annotating pairwise comparisons, and anecdotal evidence suggested that annotators on crowd sourcing platforms such as CrowdFlower prefer simpler tasks that can be performed very quickly, thus the Best-Worst approach was not used further in this study in preference to simple tweet comparisons. Subsequent analysis of other annotation efforts using the Best-Worst approach (Mohammad and Bravo-Marquez, 2017; Kiritchenko and Mohammad, 2017) indicated that Best-Worst annotations are none the less somewhat more efficient, giving 5 comparisons in approximately 3 to 3.5 times the time required for each annotation, however annotator agreement when taken as pairwise comparisons was very poor[3]. We leave further investigation of the relative merits of these annotation approaches to later study.

Primary Study: The 2,019 tweets for the primary study were annotated with both the relative annotation scheme from the pilot study and an absolute annotation scheme on a 5-point scale. For the binary scheme, 2,019 tweet pairs were chosen such that each tweet was in at least one pair. For annotation on a 5-point scale, the Self-Assessment Manikin (SAM) (Bradley and Lang, 1994) with an an additional "Can't Tell" option for unintelligible or non-English tweets was used for Valence, Dominance and Arousal. There are no SAM manikins for Surprise, so it was annotated without visual aids.

Each individual tweet received 5 annotations for each emotion dimension. 44 tweets and 44 tweet pairs for each dimension were annotated by the authors to serve as test questions. We interspersed these in every 10 annotation tasks and excluded annotators who labelled more than 30% of these tests incorrectly from further experiments, discarding their annotations. As a side effect of this process, the test tweets received many more annotations than other tweets. To avoid bias due to the selection of tasks with low ambiguity for these test questions, they were removed when calculating the annotation agreement metrics in Section 3. Annotator quality was high, with 80% of significant annotators passing $> 90\%$ of the test questions and the remaining accepted annotators passing more than 77%.

[3]Agreement when considering only best/worst choices as single comparisons was very high however.

4. Annotator Agreement Comparisons

We wished to assess the relative merits of different annotation schemes and target emotion representation schemes. In order to assess annotation with categorical emotion schemes, we drew upon data from previous annotation projects (see Section 4.2.) for comparison.

To obtain comparisons of annotator agreement between categorical annotations and dimensional annotations we utilised Krippendorffs' Alpha, a measure of agreement for which computed reliabilities are comparable across any numbers of coders, values, different metrics, and unequal sample sizes.

To apply Krippendorffs' Alpha to the various annotation scenarios, we developed conceptually similar metrics of annotator disagreement for the respective annotation schemes. We attempt to link these metrics through arguments around the nature of annotator judgements and through ensuring the metrics operate on a similar scale (i.e.: values for conceptually similar annotation differences should be the same). In this work, we do not attempt to empirically evaluate these disagreement metrics beyond comparison of agreement values on the presented data sets.

4.1. Annotation Difference Metrics

Categorical Annotations (Multiple Categories Allowed) There are several metrics that have been applied to categorical annotations with multiple categories allowed. The Jacccard set similarity metric (Jaccard, 1912) is the ratio between the sizes of the intersection and union of the sets. Passonneau (Passonneau, 2004) observed that if one annotator is inclined to provide, in general, more labels than another annotator, you should consider any extra labels from the prolific annotator as less indicative of disagreement, proposing a simple difference metric that attempts to capture this idea (see below). Passonneau later proposed a combination of the two metrics (Passonneau, 2006), capturing the granularity of the Jaccard metric and the motivating principle of his previous proposal. He named this metric MASI (Measuring Agreement on Set-valued Items). In the formulae below, A and B refer to two annotations of a data element (tweet in our case), with each a set of annotated categories.

$$\text{Jacc}(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|}$$

$$\text{Pass}(A,B) = \begin{cases} 0 & A = B \\ 0.3 & A \subset B \text{ or } B \subset A \\ 0.6 & A \cap B \neq \emptyset \\ 1 & A \cap B = \emptyset \end{cases}$$

$$\text{Masi}(A,B) = 1 - \text{Jacc}(A,B) \times \text{Pass}(A,B)$$

Another scenario, where the above metrics could be seen as overly pessimistic, is as follows: in cases where an anno-

Metric	Categorical
Wood	0.33
Masi	0.30
Jaccard	0.31
Passonneau	0.32

Table 2: Krippendorffs' Alpha for categorical emotion annotations on emoji tweets with different metrics of annotation distance.

tator feels strongly about one or two labels, but adds some others with less conviction, the annotation scheme is unable to represent the situation. If we assume that agreed labels are most likely the labels that the annotators felt strongly about, we would consider any other annotated labels to indicate only mild disagreement, even if both annotators indicated an emotion distinct from the other annotator (hence Passonneau would give a high dissimilarity of 0.6). Motivated by this observation, we developed a new difference metric: the average proportion of unmatched annotations between the two annotated sets. Again, A and B refer to sets of chosen categories by each of two annotators for a given tweet or other data element. $A \setminus B$ represents the set of elements of A that are not in B.

$$\text{Wood}(A, B) = \frac{1}{2}\left(\frac{|A \setminus B|}{|A|} + \frac{|B \setminus A|}{|B|}\right)$$

Relative Dimensional Annotations For relative emotion annotations, we can use a naive difference metric (simply if the annotations differ).

$$\text{Naive}(x, y) = \begin{cases} 0 & x = y \\ 1 & x \neq y \end{cases}$$

Following a similar intuition to the above metrics, we also develop a second difference metric for relative annotations. The situation here is somewhat simplified—there are 3 annotation options: the first tweet, about the same and the second tweet. Following the intuition that, when the tweet authors are actually close on the annotated emotion scale, one annotator may be more inclined to choose "About the same" where another may prefer to choose one tweet or the other. In such cases, the annotations can be taken to differ little, and we score such situations with a difference of 0.2.

$$\begin{aligned} \text{distance}(\text{first}, \text{second}) &= 1 \\ \text{distance}(x, x) &= 0 \\ \text{distance}(\text{same}, x) &= 0.2 \end{aligned}$$

Rating Scale Annotations In this case, a naive distance metric simply considers annotations that differ to have distance 1 (in the *Naive* row in Table 3). A more natural difference metric would be the difference between annotations taken as numerical values, scaled to lie between 0 and 1 to match the distance metrics defined above (in the *Novel* row in Table 3). It has been suggested that ranking scale annotations are better treated as ordinal (Martinez et al., 2014), where annotations from each annotator are treated as a ranking of annotated items. When seen in this way, we can consider each pair of tweets annotated by a given annotator as a pairwise comparison. If they are ranked at the same level, we treat them as "About the Same".

Metric	Comparisons	Rating Scale	Rating Scale (as comparisons)
Naive	0.39	0.41	0.39
Novel	0.45	0.47	0.51

Table 3: Overall Krippendorffs' Alpha for primary study (continuous emotion scale) with different metrics of annotation distance.

Emotion Dimension	Comparisons	Rating Scale	Rating Scale (as comparisons)
Valence	0.52	0.63	0.70
Arousal	0.45	0.49	0.60
Dominance	0.43	0.37	0.40
Surprise	0.40	0.23	0.38

Table 4: Krippendorffs' Alpha for primary study: comparison of emotion dimensions with the novel distance metric.

4.2. Annotator Agreement

To compare categorical annotation to dimensional relative annotations, we obtained annotation data from (Wood and Ruder, 2016), where 360 tweets containing at least one of a set of 62 commonly used emoji were annotated for Ekmans six emotion categories. Though the number of tweets is low, there were 17 annotators who annotated between 60 and 360 tweets each. Table 2 summarises Krippendorffs' alpha values using the above distance metrics for this data. Table 3 presents Krippendorffs' alpha values using metrics for dimensional annotations on data from the primary study. Table 4 presents a breakdown of Krippendorffs' alpha values for individual emotion dimensions (using the "novel" metrics).

Notice that the rating scale and comparison annotation agreement scores (with novel metric) are higher than for the categorical annotations even given the optimistic nature of the new annotation metrics. This confirms the supposition that the annotating on a single emotion scale is a simpler task than choosing multiple emotions categories.

Of particular interest is the improved agreement when rating scale annotations are considered as ordinal and converted to pairwise comparisons and also the higher agreement when treated this way than direct comparisons, despite the substantially larger number of pairs that result[4]. Agreement for arousal and in particular valence is notably higher than the other dimensions as has been seen in other studies, however it is interesting to note that the discrepancy is less for comparisons and when rating scale annotations are considered ordinal and converted to comparisons.

[4] An annotator providing n rating scale annotations results in $\frac{n(n-1)}{2}$ comaparisons.

Dimension	Spearman Correlation (Regression Models)	F1 (Comparison Models)
Valence	0.72	0.72
Arousal	0.64	0.69
Dominance	0.53	0.71
Surprise	0.42	0.63
Average	0.58	0.69

Table 5: Cross validation results for rating scale regression models and comparison classification models.

5. Predictive Model

As further verification of the utility of the data, we built two supervised models, one each from the 5-point rating scale and pairwise comparison annotations.

For the rating scale data, regressions were built using the approach in (Andryushechkin et al., 2017). This model consists of an ensemble of two supervised models: an SVR (Support Vector Machine Regression) with n-gram and several custom features (see (Andryushechkin et al., 2017)) and a BiLSTM (Bidirectional Long-Short Term Memory) model utilising 100 dimensional Glove word embeddings trained on tweets (Pennington et al., 2014). For the comparison data, an SVM (Support Vector Machine) was built using the same Glove word embeddings as features. The cross validation results shown in Table 5 indicate that supervised modelling can be effective for predicting emotions using this data.

6. Conclusions

We presented a new data set of English tweets with annotations using a dimensional emotion model using four emotion dimensions following the emotion representation model identified by Fontaine et.al. (Fontaine et al., 2007). To the best of our knowledge, this is the first such data set to be made publicly available.

We also compared annotator agreement across different annotation and emotion representation schemes, finding improved agreement on the simpler task of annotating one emotion dimension vs. annotating one of six emotion categories. Contrary to expectations, we found evidence that annotations on a 5-point scale produced greater annotation agreement than comparisons with a dimensional emotion model, especially when considered as ordinal annotations and converted to comparisons (taking, for each annotator, the set of all pairs of tweets from those they annotated).

Acknowledgements

We would like to thank volunteers from the Insight Centre for Data Analytics for their efforts in pilot study annotations. This work was supported in part by the Science Foundation Ireland under Grant Number 16/IFB/4336 and Grant Number SFI/12/RC/2289 (Insight). The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreements No. 644632 (MixedEmotions).

7. Bibliographical References

Andryushechkin, V., Wood, I., and O' Neill, J. (2017). NUIG at emoint-2017: Bilstm and SVR ensemble to detect emotion intensity. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 175–179, Copenhagen, Denmark. Association for Computational Linguistics.

Bradley, M. M. and Lang, P. J. (1994). Measuring emotion: The self-assessment manikin and the semantic differential. *Journal of Behavioral Theory and Experimental Psychiatry*, 25(1):49–59.

Bradley, M. M. and Lang, P. J. (2007). Affective norms for English text (ANET): affective ratings of texts and instruction manual. Technical report, University of Florida, Gainesville, Fl., USA.

Buechel, S. and Hahn, U. (2017). EMOBANK: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. *EACL 2017*, page 578.

Ekman, P., Friesen, W. V., O'Sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., Krause, R., LeCompte, W. A., Pitcairn, T., Ricci-Bitti, P. E., Scherer, K., Tomita, M., and Tzavaras, A. (1987). Universals and cultural differences in the judgments of facial expressions of emotion. *Journal of Personality and Social Psychology*, 53(4):712–717.

Fontaine, J. R. J., Scherer, K. R., Roesch, E. B., and Ellsworth, P. C. (2007). The world of emotions is not two-dimensional. *Psychological Science*, 18(12):1050–1057.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. *New phytologist*, 11(2):37–50.

Kiritchenko, S. and Mohammad, S. M. (2017). Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 32nd International Conference on Computational Linguistics and the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada.

Louviere, J. J. and Woodworth, G. (1991). Best-worst scaling: A model for the largest difference judgments. *University of Alberta: Working Paper*.

Louviere, J. J., Flynn, T. N., and Marley, A. A. (2015). *Best-worst scaling: Theory, methods and applications*. Cambridge University Press.

Martinez, H. P., Yannakakis, G. N., and Hallam, J. (2014). Don't classify ratings of affect; rank them! *IEEE transactions on affective computing*, 5(3):314–326.

Metallinou, A. and Narayanan, S. (2013). Annotation and processing of continuous emotional attributes: Challenges and opportunities. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–8.

Mohammad, S. and Bravo-Marquez, F. (2017). WASSA-2017 shared task on emotion intensity. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 34–49, Copenhagen, Denmark. Association for Computational Linguistics.

Mohammad, S. M. (2016). Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In Herbert L. Meiselman, editor, *Emotion Measurement*, pages 201–237. Woodhead Publishing.

Osgood, C. E., May, W. H., and Miron, M. S. (1975). *Cross-cultural Universals of Affective Meaning*. University of Illinois Press.

Passonneau, R. J. (2004). Computing reliability for coreference annotation. In *LREC*.

Passonneau, R. (2006). Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Preotiuc-Pietro, D., Schwartz, H. A., Park, G., Eichstaedt, J. C., Kern, M., Ungar, L., and Shulman, E. P. (2016). Modelling valence and arousal in facebook posts. In *Proceedings of NAACL-HLT*, pages 9–15.

Russell, J. A. and Mehrabian, A. (1977). Evidence for a three-factor theory of emotions. *Journal of Research in Personality*, 11(3):273–294.

Wood, I. and Ruder, S. (2016). Emoji as emotion tags for tweets. In *Emotion and Sentiment Analysis Workshop, at LREC2016*, Portorož, Slovenia.

Yannakakis, G. N., Cowie, R., and Busso, C. (2017). The ordinal nature of emotions. In *Int. Conference on Affective Computing and Intelligent Interaction*.

Yu, L.-C., Lee, L.-H., Hao, S., Wang, J., He, Y., Hu, J., Lai, K. R., and Zhang, X. (2016). Building Chinese affective resources in valence-arousal dimensions. In *Proceedings of NAACL-HLT*, pages 540–545.

Humor Detection in English-Hindi Code-Mixed Social Media Content : Corpus and Baseline System

Ankush Khandelwal, Sahil Swami, Syed S. Akthar, Manish Shrivastava
Language Technology Research Centre, International Institute of Information Technology, Hyderabad
{ankush.k, sahil.swami, syed.akhtar}@research.iiit.ac.in
m.shrivastava@iiit.ac.in

Abstract

The tremendous amount of user generated data through social networking sites led to the gaining popularity of automatic text classification in the field of computational linguistics over the past decade. Within this domain, one problem that has drawn the attention of many researchers is automatic humor detection in texts. In depth semantic understanding of the text is required to detect humor which makes the problem difficult to automate. With increase in the number of social media users, many multilingual speakers often interchange between languages while posting on social media which is called code-mixing. It introduces some challenges in the field of linguistic analysis of social media content (Barman et al., 2014), like spelling variations and non-grammatical structures in a sentence. Past researches include detecting puns in texts (Kao et al., 2016) and humor in one-lines (Mihalcea et al., 2010) in a single language, but with the tremendous amount of code-mixed data available online, there is a need to develop techniques which detects humor in code-mixed tweets. In this paper, we analyze the task of humor detection in texts and describe a freely available corpus containing English-Hindi code-mixed tweets annotated with humorous(**H**) or non-humorous(**N**) tags. We also tagged the words in the tweets with Language tags (**En**glish/**Hi**ndi/**Ot**hers). Moreover, we describe the experiments carried out on the corpus and provide a baseline classification system which distinguishes between humorous and non-humorous texts.

Keywords: humor detection, code-mixing, random forest classifier, SVM, extra tree classifier, naive bayes

1. Introduction

"Laughter is the best Medicine" is a saying which is popular with most of the people. Humor is a form of communication that bridges the gap between various languages, cultures, ages and demographics. That's why humorous content with funny and witty hashtags are so much popular on social media. It is a very powerful tool to connect with the audience. Automatic Humor Recognition is the task of determining whether a text contains some level of humorous content or not. First conference on Computational humor was organized in 1996, since then many research have been done in this field. Kao et al. (2016) does pun detection in one-liners and de Oliveira and Rodrigo (2015) detects humor in Yelp reviews. Because of the complex and interesting aspects involved in detecting humor in texts, it is one of the challenging research field in Natural Language Processing (Attardo, 1994). Identifying humor in a sentence sometimes require a great amount of external knowledge to completely understand it. There are many types of humor, namely anecdotes, fantasy, insult, irony, jokes, quote, self deprecation etc (Hay, 1995; Raz, 2012). Most of the times there are different meanings hidden inside a sentence which is grasped differently by individuals, making the task of humor identification difficult, which is why the development of a generalized algorithm to classify different type of humor is a challenging task.

Majority of the researches on social media texts is focused on English. A study by Schroeder (2010) shows that, a high percentage of these texts are in non-English languages. Fischer (2011) gives some interesting information about the languages used on Twitter based on the geographical locations. With a huge amount of such user generated data available on social media, there is a need to develop technologies for non-English languages. In multilingual regions like South Asia, majority of the social media users speak more than two languages. In India, Hindi is the most spoken language (spoken by 41% of the population) and English is the official language of the country. Twitter has around 23.2 million monthly active users in India. Native speakers of Hindi often put English words in the sentences and transliterate the whole sentence to Latin script while posting on social media, thereby making the task of automatic text classification a very challenging problem. Linguists came up with a term for any type of language mixing, known as 'code-mixing' or 'code-switching' (Auer, 1999; Muysken, 2000; Gafaranga and Torras, 2002; Bullock et al., 2011). Both the terms are used interchangeably, but there is a slight difference between the two terms. Code-mixing refers to the insertion of words, phrases, and morphemes of one language into a statement or an expression of another language, whereas transliteration of every word in a sentence to another script (here Devanagari to Latin) is coined *code-switching*(Alex, 2007). The first tweet in Figure 1 is an example of code-mixing and second is an example of code-switching. In this paper, we use code-mixing to denote both cases.

In this paper, we present a freely available corpus containing code-mixed tweets in Hindi and English language with tweets written in Latin script. Tweets are manually classified into humorous and non-humorous classes. Moreover, each token in the tweets is also given a language tag which determines the source or origin language of the token (English or Hindi). The paper is divided in sections as follows, we start by describing the corpus and the annotation scheme in Section 2. Section 3 summarizes our supervised classification system which includes pre-processing of the tweets in the dataset and the feature extraction followed by the method used to identify humor in tweets. In the next subsection, we describe the classification model and the results of the experiments conducted using

```
<tweet_id>566489684092538883</tweet_id>
<tweet>
<word lang="Ot">Anurag</word> <word lang="Ot">Kashyap</word> <word lang="En">can</word> <word lang="En">never</word>
<word lang="En">join</word> <word lang="Ot">AAP</word> <word lang="En">because</word> <word lang="En">ministers</word>
<word lang="En">took</word> <word lang="En">oath</word> <word lang="Ot">"</word> <word lang="Hi">main</word>
<word lang="Hi">kisi</word> <word lang="Hi">Anurag</word> <word lang="Hi">aur</word> <word lang="Hi">dwesh</word>
<word lang="Hi">ke</word> <word lang="Hi">bina</word> <word lang="Hi">kaam</word> <word lang="Hi">karunga</word>
<word lang="Ot">"</word> <word lang="Ot">.</word>
</tweet>
<class>H</class>

<tweet_id>766118793567350784</tweet_id>
<tweet>
<word lang="Ot">#SakshiMalik</word> <word lang="En">take</word> <word lang="En">a</word> <word lang="En">bow</word>
<word lang="Ot">!</word> <word lang="Ot">#proudIndian</word> <word lang="Ot">#Rio</word> <word lang="Ot">#Olympics</word>
<word lang="Ot">#BronzeMedal</word> <word lang="Ot">#girlpower</word> <word lang="Hi">Hamaara</word>
<word lang="Hi">khaata</word> <word lang="Hi">khul</word> <word lang="Hi">Gaya</word> <word lang="Ot">!</word>
</tweet>
<class>N</class>
```

Figure 1: Example Annotations.

character and word level features. In the last section, we conclude the paper followed by future work and references.

2. Corpus Creation and Annotation

In this section, we explain the techniques used in the creation and annotation of the corpus.

2.1. Data Collection

Python package twitterscraper[1] is used to scrap tweets from twitter. 10,478 tweets from the past two years from domains like 'sports', 'politics', 'entertainment' were extracted. Among those tweets, we manually removed the tweets which were written either in English or Hindi entirely. There were 4161 tweets written in English and 2774 written in Hindi. Finally, a total of 3543 English-Hindi code-mixed tweets were collected. Table 1 describes the number of tweets and words in each category.

Category	#Tweets	#Total Words
Humorous (**H**)	1755	26951
Non-humorous (**N**)	1698	21952

Table 1: Description of the corpus.

We tried to make the corpus balanced i.e. uniform distribution of tweets in each category to yield better supervised classification results as described by (Du et al., 2014).

2.2. Humor Annotation

The final code-mixed tweets were forwarded to a group of three annotators who were university students and fluent in both English and Hindi. Approximately 60 hours were spent in tagging tweets for the presence of humor. Tweets which consisted of any anecdotes, fantasy, irony, jokes, insults were annotated as humorous whereas tweets stating any facts, dialogues or speech which did not contain amusement were put in non-humorous class. Following are some examples of code-mixed tweets in the corpus[2]:

1. "For #WontGiveItBack to work, Dhoni needs to say 'Trophy toh ghar par hi bhul aaye' "
   (For #WontGiveItBack to work, Dhoni needs to say "We forgot the trophy at home") .

2. Na sadak na naukri bas badh rahi gundagardi #FailedCMNitish
   (No roads no naukri, only hooliganism increasing #FailedCMNitish)

3. Subha ka bhula agar sham ko wapas ghar aa jaye then we must thank GPS technology.
   (If someone lost in the morning return home in the evening then we must thank GPS technology.)

4. She : Gori hai kalaiyan pehna de mujhe hari hari chudiya He *gets green bangles* She : Not this green ya, bottle green color.
   *(She : (sings a song) My wrists are so beautiful, give me green green bangles He *gets green bangles* She : Not this green ya, bottle green color.)*

5. Dard dilon ke kam ho jaate... twitter par agar poetries kam ho jaate
   (pain in the heart will reduce...if number of poetries decreases on twitter).

Annotators were given certain guidelines to decide whether a tweet was humorous or not. The context of the tweet could be found by searching about hashtag or keywords used in the tweet. Example (1) uses a hashtag '#WontGiveItBack' which was trending during the ICC cricket world cup 2015. Searching it on Google gave 435k results and the time of the tweet was after the final match of the tournament. So there is an observational humor in (1) as India won the world cup in 2011 and lost in 2015 , hence the tweet was classified as humorous. Any tweets stating any facts, news or reality were classified as non-humorous. There were many tweets which did not contain any hashtags, to understand the context of

[1] https://github.com/taspinar/twitterscraper

[2] Translation of Hindi words written in English are provided in the brackets. They are not a part of the tweet.

such tweets annotators selected some keywords from the tweet and searched them online. Example (2) contains a comment towards a political leader towards development and was categorized as non-humorous. Tweets containing normal jokes and funny quotes like in (3) and (4) were put in humorous category. There were some tweets like (5) which consists of poem or lines of a song but modified. Annotators were guided that if such tweets contains satire or any humoristic features, then it could be categorized as humorous otherwise not. There were some tweets which were typical to categorize like (5), hence it was left to the annotators to the best of their understanding. Based on the above guidelines annotators categorized the tweets. To measure inter annotator agreement we opted for Fleiss' Kappa (Fleiss and Cohen, 1973) obtaining an agreement of 0.821 in case of humorous class and 0.794 in case of non-humorous class. Both humorous and non-humorous tweets in nearly balanced amount were selected to prepare the corpus. If we had included humorous tweets from one domain like sports and non humorous tweets from another domain like news then, it would have given high performance of classification (de Oliveira and Rodrigo, 2015). To classify based on the semantics and not on the domain differences, we included both types of tweets from different domains. Many tweets contains a picture along with a caption. Sometimes a caption may not contain humor but combined with the picture, it can provide some degree of humor. Such tweets were removed from the corpus to make the corpus unimodal. In Figure 1, the first tweet, "Anurag Kashyap can never join AAP because ministers took oath 'main kisi Anurag aur dwesh ke bina kaam karunga' " *(Anurag Kashyap can never join AAP because ministers took oath 'I will work without any affection (Anurag in Hindi) and without hesitation (dwesh in Hindi)')*, was classified as humorous. The second tweet, "#SakshiMalik take a bow! #proudIndian #Rio #Olympics #BronzeMedal #girlpower Hamaara khaata khul Gaya!" *(#SakshiMalik take a bow! #proudIndian #Rio #Olympics #BronzeMedal #girlpower Our account opened!)* was classified as non-humorous as it contains a pride statement.

2.3. Language Annotation

Code-mixing provides some challenges for language identification like spelling variations and non-grammatical structures in a sentence (Barman et al., 2014). Therefore, we annotated the tweets with the language at the word level. Native speakers of Hindi and proficient in English, labelled the language of the tokens in the tweets. Three types of tags were assigned to the tokens , **En** is assigned to the tokens present in English vocabulary like "family", "Children" etc. Similarly, **Hi** is assigned to the tokens present in Hindi vocabulary but transliterated to Latin script like "samay" (time), "aamaadmi" (common man). Rest of the tokens consists of proper nouns, numbers, dates, urls, hashtags, mentions, emojis and punctuations which are labelled as **Ot**(others). Major concern in language annotation was to annotate the words present in both languages (ambiguous words). For example, 'to' ('but' in Hindi) and 'is' ('this' in Hindi) , for this scenario annotators understood the context of the tweet and based on that the words were being annotated. For example, consider two sentences 'This place is 500 years old' and 'Is bar Modi Sarkar'*(This time Modi Government)*. So 'is' in first sentence should be tagged as English because it is used as a verb. It is transliterated to English in latter sentence and used as determiner ('this') so it was tagged as Hindi. So these kind of words were tagged based on their meanings and usage in both languages. Figure 1 describes the language annotation of tweets in the corpus.

2.4. Annotation Scheme

Two example annotations are illustrated in Figure 1. First line in every annotation consists of tweet id. Each tweet is enclosed within <tweet></tweet> tags and each word is enclosed within <word lang=" "></word> containing the language annotation for each word. Last line determines the category in which the tweet belongs i.e. humorous or non-humorous, enclosed by <class></class> tags. The annotated dataset with the classification system is made available online [3].

2.5. Error Analysis

Social media users often make spelling mistakes or use multiple variations of a word while posting a text. We replaced all such errors with their correct version. For words in Hindi that were transliterated to Latin script, we adopted a common spelling for all those words across the corpus. For example, 'dis' is often used as a short form for 'this', so we replaced every occurrence of 'dis' to 'this' in the corpus. Some examples of spelling variations are, 'pese' for 'paisa' *(unit of curreny)*, 'h' for 'hai' *(is)*, 'ad' for 'add' *(addition)*.

3. System Architecture

In this section, we describe our machine learning model which is trained and tested on the corpus described in the previous section.

3.1. Pre-processing of Corpus

Preprocessing starts with tokenization, which involves separation of words using space as the delimiter and then converting the words to lower cases which is followed by the removal of punctuation marks from the tokenized tweet. All the hashtags, mentions and urls, are stored and converted to 'hashtag', 'mention' and 'url' respectively. Sometimes hashtags provide some degree of humor in tweets, hence we segregated hashtags on the basis of camel cases and included the tokens in the tokenized tweets (hashtag decomposition) (Belainine et al., 2016; Khandelwal et al., 2017). for example, *#AadabArzHai* can be decomposed into 3 words, 'Aadab', 'Arz' and 'Hai'. Finally the tokenized tweets are stored along with the presence of humor as the target class.

3.2. Classification Features

The features used to build attribute vectors for training our classification model are described below. We use character

[3] https://github.com/Ankh2295/humor-detection-corpus

Features (in %)	Kernel SVM	Random Forest	Extra tree	Naive bayes
N-grams	68.5	63.7	65.4	68.2
Bag-of-words	60	61.6	61.6	67.3
Common words and hashtags	64.8	61.9	64.7	67.3
All features	**69.3**	65.2	67.8	67.2

Table 2: Accuracy of each feature using different classifiers

level and word level features for the classification (Khandelwal et al., 2017). For all the features, we separated the words in the tweets based on the language annotation (Section 2.3) and prepared the feature vector for each tweet by combining the vectors for both the languages [4].

3.2.1. N-grams

Previous researches shows that letter n-grams are very efficient for classifying text. They are language independent and does not require expensive text pre-processing techniques like tokenization, stemming and stop words removal, hence in the case of code-mix texts, this could yield good results (Miller et al., 2012; Alowibdi et al., 2013). Since the number of n-grams can be very large we took trigrams which occur more than ten times in the corpus.

3.2.2. Bag-of-words

For classifying humor in texts, it is important to understand the semantics of the sentence. Thus, we took a three word window as a feature to train our classification models to incorporate the contextual information.

3.2.3. Common Words and Hashtags

Many jokes and idioms sometimes have common words. We identified those words and took them as as a feature for classification. In the preprocessing step, we decomposed hashtags using camel cases and added them along with the words. Hence, common words in the hashtags were also included in the feature vector.

3.3. Classification Approach and Results

We experimented with four different classifiers, namely, support vector machine (Cristianini and Shawe-Taylor, 2000), random forest, extra tree and naive bayes classifier (Pedregosa et al., 2011). Chi square feature selection algorithm is applied to reduces the size of our feature vector. For training our system classifier, we used Scikit-learn (Pedregosa et al., 2011).

10-fold cross validation on 3543 code-mixed tweets was carried out by dividing the corpus into 10 equal parts with nine parts as training corpus and rest one for testing. Mean accuracy is calculated by taking the average of the accuracy obtained in each iteration of the testing process. Table 2 shows the accuracy for each feature when trained using mentioned classifiers along with the accuracy when all the features are used along with the overall accuracy. Support vector machine with radial basis function kernel and extra tree classifier performs better than other classifiers and yields 69.3% and 67.8% accuracy respectively. The reason kernel SVM yields the best result is because the number of observations is greator than the number of features (Hsu et al., 2003). N-grams proved to be the most efficient in all classification models followed by common words and hastags. Bag-of-words feature performed the worst in SVM, random forest and extra tree classifier but yielded better result in naive bayes classifiers. Accuracies mentioned in table 2 were calculated using fine tuning of model parameters using grid search.

[4]Threshold values described are taken after empirical fine tuning

4. Conclusion and Future Work

In this paper, we describe a freely available corpus of 3453 English-Hindi code-mixed tweets. The tweets are annotated with humorous(H) and non-humorous(N) tags along with the language tags at the word level. The task of humor identification in social media texts is analyzed as a classification problem and several machine learning classification models are used. The features used in our classification system are n-grams, bag-of-words, common words and hashtags. N-grams when trained with support vector machines with radial basis function kernel performed better than other features and yielded an accuracy of 68.5%. The best accuracy (69.3%) was given by support vector machines with radial basis function kernel.

This paper describes the initial efforts in automatic humor detection in code-mixed social media texts. Corpus can be annotated with part-of-speech tags at the word level which may yield better results in language detection. Moreover, the dataset can be further extended to include tweets from other domains. Code-mixing is very common phenomenon on social media and it is prevalent mostly in multilingual regions. It would be interesting to experiment with code-mixed texts consisting of more than two languages in which the issue of transliteration exists like Arabic, Greek and South Asian languages. Comparing training with code-mixed tweets with training with a merged corpus of monolingual tweets in English and Hindi could be an interesting future work.

5. Bibliographical References

Alex, B. (2007). *Automatic detection of English inclusions in mixed-lingual data with an application to parsing*. Ph.D. thesis, University of Edinburgh.

Alowibdi, J. S., Buy, U. A., and Yu, P. (2013). Language independent gender classification on twitter. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 739–743. ACM.

Attardo, S. (1994). *Linguistic theories of humor*, volume 1. Walter de Gruyter.

Auer, P. (1999). From codeswitching via language mixing to fused lects: Toward a dynamic typology of bilingual speech. *International journal of bilingualism*, 3(4):309–332.

Barman, U., Das, A., Wagner, J., and Foster, J. (2014). Code-mixing: A challenge for language identification in the language of social media. In *In Proceedings of the First Workshop on Computational Approaches to Code-Switching*. Citeseer.

Belainine, B., Fonseca, A., and Sadat, F. (2016). Named entity recognition and hashtag decomposition to improve the classification of tweets. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 102–111.

Bullock, B. E., Hinrichs, L., and Toribio, A. J. (2011). World englishes, code-switching, and convergence. *Oxford Handbook of World Englishes*.

Cristianini, N. and Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.

de Oliveira, L. and Rodrigo, A. L. (2015). Humor detection in yelp reviews.

Du, M., Pierce, M., Pivovarova, L., and Yangarber, R. (2014). Supervised classification using balanced training. In *International Conference on Statistical Language and Speech Processing*, pages 147–158. Springer.

Fischer, E. (2011). Language communities of twitter. *Flickr. Last modified October 24th*.

Fleiss, J. L. and Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Gafaranga, J. and Torras, M.-C. (2002). Interactional otherness: Towards a redefinition of codeswitching. *International Journal of Bilingualism*, 6(1):1–22.

Hay, J. (1995). Gender and humour: Beyond a joke.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2003). A practical guide to support vector classification.

Kao, J. T., Levy, R., and Goodman, N. D. (2016). A computational model of linguistic humor in puns. *Cognitive science*, 40(5):1270–1285.

Khandelwal, A., Swami, S., Akhtar, S. S., and Shrivastava, M. (2017). : Classification of spanish election tweets (coset) 2017: Classifying tweets using character and word level features.

Mihalcea, R., Strapparava, C., and Pulman, S. (2010). Computational models for incongruity detection in humour. *Computational linguistics and intelligent text processing*, pages 364–374.

Miller, Z., Dickinson, B., and Hu, W. (2012). Gender prediction on twitter using stream algorithms with n-gram character features. *International Journal of Intelligence Science*, 2(04):143.

Muysken, P. (2000). *Bilingual speech: A typology of code-mixing*, volume 11. Cambridge University Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Raz, Y. (2012). Automatic humor classification on twitter. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop*, pages 66–70. Association for Computational Linguistics.

Schroeder, S. (2010). Half of messages on twitter arenât in english[stats].

Dialogue Scenario Collection of Persuasive Dialogue with Emotional Expressions via Crowdsourcing

Koichiro Yoshino[1,2], Yoko Ishikawa[1], Masahiro Mizukami[1], Yu Suzuki[1], Sakti Sakriani[1], and Satoshi Nakamura[1]

[1]Graduate School of Information Science, Nara Institute of Science and Technology, 8916-5, Takayama-cho, Ikoma, 6300192, Japan

[2]PRESTO, Japan Science and Technology Agency, 4-1-8 Hon-machi, Kawaguchi, Saitama 332-0012, Japan

{koichiro, ishikawa.yoko.io5, masahiro-mi, ssakti, ysuzuki, s-nakamura}@is.naist.jp

Abstract

Existing dialogue data collection methods such as the Wizard of Oz method (WoZ) or real dialogue recording are costly, and they prevent launching a new dialogue system. In this study, we requested crowd workers in crowdsourcing to create dialogue scenarios according to the instruction of the situation for persuasive dialogue systems that use emotional expressions. We collected 200 dialogues in 5 scenarios for a total of 1,000 via crowdsourcing. We also annotated emotional states and users' acceptance for system persuasion by using crowdsourcing. We constructed a persuasive dialogue system with the collected data and evaluated the system by interacting with crowd works. From the experiment, it was investigated that the collected labels have sufficient agreement even if we did not impose any training of annotation to workers.

Keywords: Dialogue corpus, emotion labels, persuasive dialogue, crowdsourcing

1. Introduction

Expressing one's emotion or feeling with language helps us to understand each other in human-human communication. We cannot observe the internal states of other people such as our emotions or feelings directly, and we indirectly estimate the internal states from observable states such as utterances and the behaviors of others. Explicitly expressing internal states helps to build a good relationship between humans by reducing misunderstandings and unrequited feelings (Ekman, 1993). For example, sharing the same emotional state helps us to develop close relationships with others.

Expressing emotional states is very effective to process the persuasion or negotiation (Keltner and Haidt, 1999; Morris and Keltner, 2000). Persuaders who express positive emotions increase the ratio of success at persuasion more than persuaders who do not express their emotions because expressing positive emotions gives a cooperative impression to a partner (Carnevale and Isen, 1986; Forgas, 1998). On the other hand, expressing negative emotions such as "anger" may wrest a concession from a partner even if the proposal from the persuader is not attractive to the partner, especially when the partner does not have any other options to choose (Sinaceur and Tiedens, 2006). Using emotional expressions is one of the most effective ways to change the belief or behaviors of a partner to increase the success rate of persuasion or negotiation.

Captology (Computers as persuasive technologies) (Fogg, 1997) is a research area to work on systems that affect the beliefs or behaviors of users. A persuasive dialogue system is known as a part of this research area, and some dialogue systems have been developed that can change the actions or behaviors of users with persuasion. Some studies of persuasive dialogue systems investigated efficient dialogue strategy on dialogue management to persuade users. Hiraoka et al., (2016) introduced actions of framing and logical explanations of advantages and disadvantages of products for persuading the users to purchase the product.

The major problem of implementing a dialogue system is data, in any domains or tasks of systems, because most methods of dialogue modeling are based on statistical methods that require large-scale data-sets. However, collecting new dialogue data in accordance with a defined new task is costly. Some approaches enable easy data collection in a new domain by utilizing Web data (Banchs and Li, 2012) or by extracting dialogue parts from chat-like conversations (Nio et al., 2014). However, collecting large-scale dialogue data that contain emotional expressions are still difficult. Emotional expressions tend to be observed in communications between people who have close relationships. However, it is hard to record dialogues in such closed situations. It is also difficult to extract such conversations from Web because expressing emotions in public space is somewhat suppressed.

Crowdsourcing has attracted attention as an efficient way for collecting or expanding dialogue data (Yu et al., 2016). We collected dialogue data which contains emotional expressions by requesting crowd workers to create complete dialogue scenarios. This approach makes it possible to create a dialogue scenario even if it is hard to record the dialogue data in the hypothetical situation of the scenario. One large concern in this approach is whether we can collect realistic dialogue data that can be used for the training of dialogue systems. We built and evaluated a simple persuasive dialogue system based on an example-based approach to investigate that the collected data is usable for the training of the dialogue system. The resultant model worked well even if the dialogue scenario collection of one dialogue was conducted by a single annotator. In this work, we used crowdsourcing not only for dialogue scenario collection but also for label annotations. We also found that the labels had sufficient quality for them to be used for system construction, even if they were annotated by crowd workers.

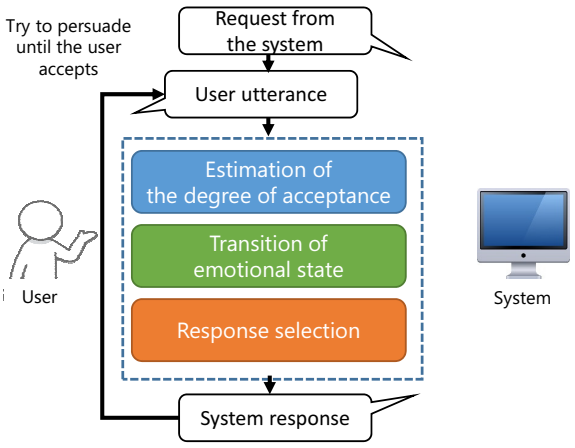


Figure 1: The overview of the proposed persuasive dialogue system with emotional expressions.

2. Persuasive Dialogue Scenario

This section describes the scenario of persuasive dialogue systems that we assumed in our data collection. The flow of persuasion from the system is shown in **Figure 1**. A persuasive dialogue system talks with users about their living habits and tries to change these habits. The dialogue starts with a request from the system to the users, and the system tries to continue with persuasion until the users accept the request. The emotional states and the degrees of the users' acceptance are estimated by the system. The dialogue ends if the system recognizes the acceptance of the users or if a pre-defined number of turns pass (=failure). In this scenario, the dialogue data of the persuasive dialogue labeled with emotional states and the degree of the users' acceptance are required. The collection procedure of these data, the dialogue scenarios, annotations of emotion, and annotations of the degree of users' acceptance are described in the following sections.

3. Emotion Labeled Corpus Construction Through Crowdsourcing

A labeled corpus of persuasive dialogue is required to construct the persuasive dialogue system. We collected an emotionally persuasive corpus through crowdsourcing (Howe, 2006). In the data collection, first, we requested that crowd workers write dialogue scenarios of persuasive dialogue by using emotional expressions. Then, labels of the users' acceptance and emotion were annotated by other crowd workers.

3.1. Scenarios Collection

To collect the scenarios, we requested that crowd workers write dialogue scenarios of persuasive dialogue in accordance with the following instructions.

- The dialogue starts with a system suggestion for changing daily activities
- The system tries to persuade a user with some emotional expressions
- The dialogue ends with the user accepting the suggestion
- The dialogue consists of more than 20 utterances

The specific suggestions to the crowd workers are as follows.

Instructions given to crowd workers

John is living with a robot who guides his daily life to improve his living habits. The robot encourages John to change his habit if he finds a bad habit. For example, when the robot find that John does not get enough exercise, he presses John to go jogging. However, John does not listen his advice even if the robot describes the reason which John should follow his advice. Thus, the robot decided to persuade John by using emotional expressions including anger, sadness, and happiness.
Create a dialogue example of the persuasion between John and the robot. The dialogue starts with a system suggestion for getting exercise. The robot tries to persuade John by using various emotional expressions, and finally, John accepts the offer of the robot. The robot must use one or more emotional expressions from emotion categories of happiness, sadness, and anger. The dialogue consists of more than 20 utterances.

We prepared five daily-life guidance scenarios: "Clean the room (cleaning)," "Don't leave a dish unfinished (lunch)," "Sleep early (sleep)," "Stop playing the game (game)" and "Get some exercise (exercise)."

Table 1: Numbers of scenarios and utterances

Scenarios	Dialogues	$\#q_i$	$\#r_i$
Cleaning	200	2282	2292
Lunch	200	2173	2185
Sleep	200	2147	2180
Game	200	2175	2155
Exercise	200	2203	2216

Table 1 shows the number of collected dialogues and utterances for each scenario. A total of 1000 dialogues (200 dialogues for each scenario) were collected with crowdsourcing. One crowd worker generated 5 dialogues at most (1 for each scenario). We annotated labels for every dialogue to use as the training data of the dialogue system.

3.2. Emotional States

We used five emotional states: "neutral," "happy," "contented," "angry," and "sad," which are defined in Russell's Circumplex Model (Russell, 1978). Two dimensions were used: valence and arousal, in Russell's model, and the emotional state was decided using degrees of these two dimensions. States were labeled as "Happy," "Angry," "Sad" and "Contented" from the first quadrant to the fourth quadrant. Around the origin was annotated with "Neutral."

3.3. Degree of Users' Acceptance

Knowing the degree of users' acceptance, that is, whether or not the users will accept the system's suggestions, is necessary to construct a persuasive dialogue system because

the degree is used for the detection of the end of the dialogue with the acceptance of the users. We defined the users' acceptance in 5 degrees (1: Refused, 3: Not sure, 5: Accepted).

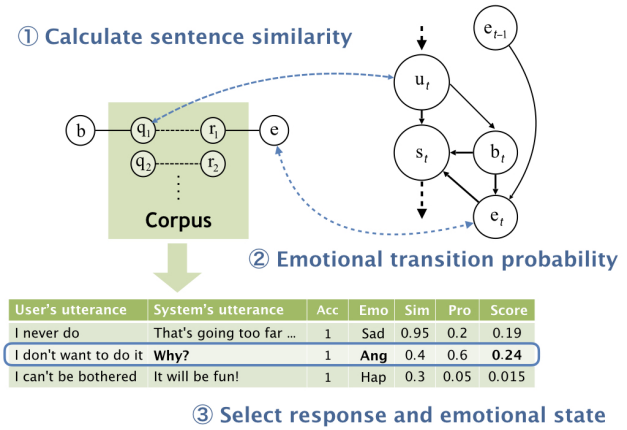
3.4. Label Annotation

We made annotations of emotion labels and users' acceptance rates for persuasion with 5 degrees to construct a persuasive dialogue system with emotional expressions. We requested that crowd workers annotate both the degree of users' acceptance for the user utterances and the emotional labels for each utterance. Three annotators were assigned for one label, and we utilized labels that had 2 or 3 agreements. We removed examples that had 3 different labels by 3 annotators, because it probably be caused by the difficulty of the annotation or qualities of crowd workers assigned for the example. Users' acceptances in 5 degrees (5: accept, 4: possibly accept, 3: cannot say, 2: possibly reject and 1: reject) were annotated on user queries in dialogues, and emotional states with 5 kinds of labels were annotated on every user query and system response. The crowd workers read every utterance of the dialogue scenario before the annotation and annotate the acceptance and the emotion label for each utterance. The figure of Russell's Circumplex Model (Russell, 1978) is presented crowd workers before every annotation session. The figure includes some example expressions of each emotion classes (e.g. glad or enjoyable belong to the "happy" class). Table 2 Table 3 and Table 4 show the percentage of annotated labels for each kind of annotation, and "None" means no label was assigned for the example because the annotation for the example was divided.

In this corpus, 30% of the utterances were annotated as "Neutral," and more than 20% utterances were annotated as "Angry" or "Sad" for each label. In details, the proportions of negative emotions ("Angry" and "Sad") in system utterances were smaller than the proportions of negative emotions in user utterances. It indicates that created scenarios contain some examples to elicit positive emotion for negative user utterances by using positive emotions. The proportions of negative emotions of users are significant because we instructed scenario writers to write more extended scenarios (more than 20 utterances) by the final agreement of the system and the user. The proportion of "None" of system utterances was little small because scenario writers are conscious of using emotional expressions on system utterances. Table 4shows an example of an annotated corpus that we collected.

3.5. Annotation Agreement

As results, we collected 1,000 persuasive dialogue corpus annotated with emotional states and users' acceptance rate for the persuasion. To examine the quality of the collected corpus via crowdsourcing, we calculated Fleiss' Kappa value (Fleiss et al., 2013), the rate of concordance of each annotator, for both the degree of users' acceptance and the emotional label. Fleiss' Kappa of the emotion labels without removing disagreed examples was 0.345. After the removing of disagreed examples as described in Section 3.4., the Fleiss' Kappa was 0.411, which means moderate agreement. Fleiss' Kappa of the degree of users' acceptance was 0.370, which means fair agreement. We also calculated the mean squared error value of the degree of users' acceptance with the following equation.



User's utterance	System's utterance	Acc	Emo	Sim	Pro	Score
I never do	That's going too far ...	1	Sad	0.95	0.2	0.19
I don't want to do it	Why?	1	Ang	0.4	0.6	0.24
I can't be bothered	It will be fun!	1	Hap	0.3	0.05	0.015

Figure 2: Response selection of dialogue system with emotional states

$$\frac{|x_A - x_B|^2 + |x_B - x_C|^2 + |x_C - x_A|^2}{3}. \quad (1)$$

Here, A-C are IDs of annotators. The mean squared error value was 0.850 (lower is better), which is low enough for 5-degree annotations.

The Kappa value of emotion labels was over 0.4, and the Kappa value of the users' acceptance was lower than 0.4. However, the mean square error of the user's acceptance was less than 1.0, and it indicated the collected data has fair agreement to be used as the training data of the system. We removed examples that had three different labels by three annotators from the training data of the system; thus, the quality of the data used for the system training is much better than this score. We have not compared the annotation results by crowd workers with any annotation results of well-trained annotators. An existing work reported 0.52 Kappa value for valence-arousal emotion labeling by using audio and video data (Konar and Chakraborty, 2015). Our work only uses text data to decide emotion labels; thus, our Kappa value is not particularly low even if we used crowd workers for annotations. However, comparing results by well-trained annotator and crowd workers in the same setting is still a remaining future work.

Collected data included many utterances that had negative emotion labels such as "Angry" and "Sad," because the instruction to crowd workers focused on the process of persuasion. Positive emotions only happened at the last part of the dialogue; thus, the number is smaller than the number of negative emotions.

4. Persuasive Dialogue System Trained from The Collected Data

We built a persuasive dialogue system based on the example-based architecture according to the belief-desire theory of emotion (Reisenzein, 2009). In this theory, the combination of a desire and a belief evokes emotions. The desire means the goal to be achieved, and the belief implies

Table 2: Instructions for users' acceptance

Label	Label meaning	Points to be checked
1	John rejects the suggestion of the robot	Does the sentence contains expressions of strong rejection?
2	John possibly reject the suggestion of the robot	Does the sentence contains expressions of hesitate?
3	It cannot be said John accepts or rejects the suggestion of the robot	Does the sentence contains ambiguous expression? Is it difficult to find any rejecting or accepting expressions?
4	John possibly accept the suggestion of the robot	Does the sentence contains expressions of agreement?
5	John accepts the suggestion of the robot	Does the sentence contains expressions of clear acceptance?

Table 3: Percentage of acceptance labels in corpus

label	percentage
1	23.36%
2	26.39%
3	17.211%
4	10.27%
5	9.78%
NONE	9.99%

Table 4: Percentage of emotion labels in corpus

Label	Total	User	System
Neutral	28.73%	22.93%	33.54%
Happy	8.74%	6.91%	10.57%
Contentment	4.18%	5.21%	3.15%
Angry	23.04%	23.36%	22.72%
Sad	24.90%	28.89%	20.91%
None	10.41%	11.46%	9.36%

Table 5: Example of annotated corpus

Speaker	Utterance	Accept	Emotion
Sys	Let's clean the room	—	Neutral
User	No, it's a bother.	1	Sad
Sys	I see it. There is some trash on the floor	—	Neutral
User	I'll do it later. Not now.	NONE	Sad
Sys	You always say you'll do it but end up doing nothing.	—	Angry
User	Sure.	3	Sad
Sys	You don't have to say you'll do it. I'll support you regardless.	—	Happy
User	OK, I do.	5	Sad
Sys	Great, let's give it all you've got!	—	Happy

Table 6: Agreements of annotators

	emotion	acceptance
#annotations	22,008	10,980
Fleiss' Kappa	0.411	0.370
Mean square error	—	0.850

the belief of environment observation including achievement of the desired goal. The belief-desire model evokes positive emotions if the belief is approaching the desired goal. However, the model evokes negative emotions if the belief is leaving the desired goal. For example, if someone wants to go to picnic (=desire) but it is difficult to go to a picnic because the weather condition is bad (=belief), negative emotions are evoked because the desire may not be achieved. On the other hand, if there are small mismatches between the desire and the belief, the model evokes a positive emotion such as "happy" or "contentment". For example, if someone wants to go to picnic (=desire) and it is possible because the weather condition is good (=belief), a positive emotion is evoked.

In the persuasive scenario, we can use the users' acceptance and success of persuasion as the belief and desire of the system. The system transits its own emotional state: if the degree of user's acceptance is low, the system transits the emotional state to "anger" or "sad", and if the degree of user's acceptance is high, the system transits the emotional state to "happy", or "contentment". In the proposed architecture, the system selects a response $\hat{s_t}$ and its emotional state $\hat{e_t}$ by using the given user utterance u_t and estimated belief (user's acceptance) b_t (**Figure 2**).

The belief estimation (estimation of users' acceptance) is modeled by Support Vector Regression (SVR). SVR is an expansion of Support Vector Machine (SVM) for the regression problem. SVR has high generalizing capability because the learning of SVR minimizes the upper bound of generalization error. We used the corpus that is annotated with the degree of user's acceptance as described in Section 3.4.. To make feature vectors from the user's utterances for the regression, we extracted words as linguistic features from the user utterance by using morphological analyzer Mecab (Kudo et al., 2004). Synonyms of words in the user utterance are extracted by using WordNet (Bond et al., 2012) to extend the word feature vector. We also used positive/negative score by using pre-defined dictionary of positive/negative words (Takamura et al., 2005). Each extracted vector is concatenated as a single vector to be used as the input of SVR. The regression learns the annotated degree of user's acceptance for each utterance.

The system calculates the cosine similarity $\cos(u_t, q_j)$ for each pair of the user utterance u_t and a user-query in the example database q_j (Figure 2-①). The example database consists of pairs of a user query q_j and its response r_j with annotations of the user's acceptance of the query b_j and the emotional state of the response e_j. The database consist of query-response pairs that are extracted from the collected corpora.

In the step of Figure 2-②, the system calculates the posterior probability of emotion e_j with $P[e_j]$ that is determined by the transition probabilities $P(e_t|e_{t-1}, b_t)$ estimated with maximum likelihood estimation on the training dialogue corpus.
At the last step of Figure 2-③, the system gives a score for each query-response pair $< q_i, r_i >$ as,

$$score(< q_j, r_j >) = \cos(u_t, q_j) \times P[e_j]. \quad (2)$$

Finally the system responds with the $\hat{r_t}$ that has the highest score. Each response r_j has the annotation of the emotion label e_j; thus, the emotion of the system $\hat{e_t}$ is also decided by the score.
We evaluated the dialogue system through real dialogue with crowd workers on crowdsourcing. Through crowdsourcing, 92 users including 57 females and 35 males participated in the evaluation. Crowd workers actually talked with dialogue systems in different five scenarios we defined (cleaning, lunch, sleep, game and exercise). Each dialogue system tried to persuade the worker until the worker accepted the suggestion of the system. The order of the system was randomly selected. The dialogue was ended after the worker accepted the request or after 20 dialogue turns passed (=failure). After the dialogue, workers answered four questions for each dialogue system with a 5-level subjective score (1: Disagree, 3: Not sure, 5: Agree). The four questions were defined as:

- NATURALNESS: Did you feel that the system's response was natural?
- PERSUASIVENESS: Did you feel that the system's suggestion was persuasive?
- KINDNESS: Did you feel that the system kindly talked with you?
- HUMANLIKENESS: Did you feel that the system was humanlike?

As the baseline system, we also constructed a system that does not have emotion transition architecture. The baseline system select a response in example-based dialogue manner from response candidates that is annotated with "neutral" emotion label.

Table 7: Average scores of subjective evaluation.

Question	w.o. emotion	w. emotion
NATURALNESS	3.402	3.293
PERSUASIVENESS	3.598	3.522
KINDNESS	3.511	3.576
HUMANLIKENESS	3.511	3.522

4.1. Evaluation Results

Table 7 shows the results of the subjective evaluation. The evaluation results indicate that the collected corpus had sufficient quality for it to be used for the training data of dialogue systems. The scores for the proposed model with emotional states was not significantly higher than the baseline model without emotional states. However, questionaries for evaluation participants indicated that some users are positive for the system that has emotional states; thus, we have case studies in the next session.

Table 8: A dialogue example with positive emotional expressions, which is scored high

Turn	Utterance	Accept.	Emo.
Sys 1	Let's have a jogging tomorrow to solve the lack of fitness.	—	Neutral
Usr 1	I'm not a morning person.	0.80	—
Sys 2	I'll be happy to attend you.	—	Happy
Usr 2	Really? I will consider to go.	2.82	—

Table 9: A dialogue example with positive emotional expressions, which is scored low

Turn	Utterance	Accept.	Emo.
Sys 1	You should clean-up your room.	—	Neutral
Usr 1	I'll do that later	0.78	—
Sys 2	You can invite your friends if you clean up the room	—	Happy
Usr 2	I don't have any friends...	0.84	—

4.2. Case Studies

Table 8 and 9 show example dialogues of using "Happy" emotions by the system. The first example is scored high, however, the second example is scored low, even if the system uses the same emotion in these examples. Two hypotheses cause the difference: user preference for the system emotion and dialogue context. If the user does not like to talk with emotional people, the dialogue evaluation will be negative even though the system uses appropriate emotional responses. The other problem is caused by the example-based dialogue response selection. The example-based response selection only can consider the contents of the previous user utterance. However, the system still requires additional information to select the appropriate response (e.g., the user does not have any friends).
We show a distribution of human evaluation scores of each metrics to look at the evaluator dependent scores in Figure 3–7. Differences in scores for the system with emotional state and the system without the emotional state (w.emotion-w.o.emotion) are calculated for each evaluator. These results indicate that some users prefer to talk with the system that has an emotional state, in contrast to users who do not like communicating with the system with emotional expressions. Figure 7 indicates that some evaluators who give low scores for some metrics also give low scores for other metrics.

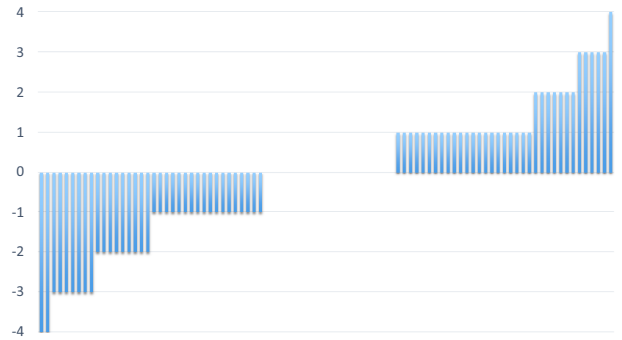


Figure 3: Evaluations of naturalness for each example (w.emotion-w.o.emotion)

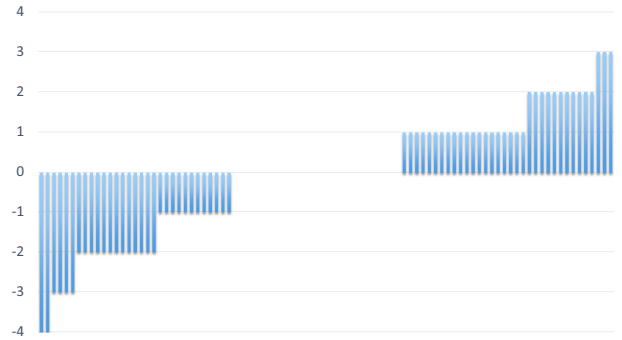


Figure 4: Evaluations of persuasiveness for each example (w.emotion-w.o.emotion)

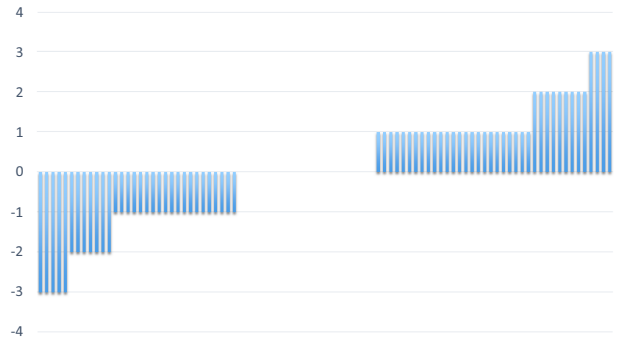


Figure 5: Evaluation of kindness for each example (w.emotion-w.o.emotion)

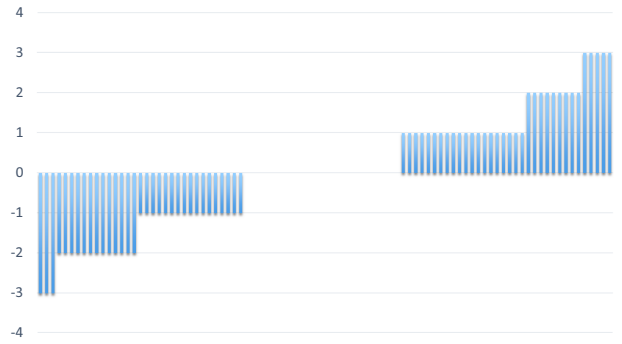


Figure 6: Evaluation of humanlikeness for each example (w.emotion-w.o.emotion)

5. Conclusion

In this study, we constructed a dialogue corpus for persuasive dialogue systems via crowdsourcing, including the labeling of emotional states and the acceptance of users' utterances. Labels of emotions and users' acceptances were also annotated by crowd workers. The labeling results had moderate variance; however, using several annotators contributed to increasing the number of usable labeled utterances for training. We also evaluated the dialogue system trained with the collected dialogue data via crowdsourcing. The results indicated that the corpus has sufficient quality for it to be used as a training set of the dialogue system, even if one crowd worker created a scenario of a conversation for entire dialogue.

6. Acknowledgement

This work is supported by JST PRESTO (JPMJPR165B).

7. References

Banchs, R. E. and Li, H. (2012). Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the ACL 2012 System Demonstrations*, pages 37–42. Association for Computational Linguistics.

Bond, F., Baldwin, T., Fothergill, R., and Uchimoto, K. (2012). Japanese semcor: A sense-tagged corpus of japanese. In *Proceedings of the 6th global wordnet conference*, pages 56–63. Global Wordnet Association.

Carnevale, P. J. and Isen, A. M. (1986). The influence of positive affect and visual access on the discovery of integrative solutions in bilateral negotiation. *Organizational behavior and human decision Processes*, 37(1):1–13.

Ekman, P. (1993). Facial expression and emotion. *American psychologist*, 48(4):384.

Fleiss, J. L., Levin, B., and Paik, M. C. (2013). *Statistical methods for rates and proportions*. John Wiley & Sons.

Fogg, B. (1997). Captology: the study of computers as persuasive technologies. In *Extended Abstracts on Human Factors in Computing Systems*, pages 129–129. Association for Computing Machinery.

Forgas, J. P. (1998). On feeling good and getting your way: mood effects on negotiator cognition and bargaining strategies. *Journal of personality and social psychology*, 74(3):565.

Hiraoka, T., Neubig, G., Sakti, S., Toda, T., and Nakamura, S. (2016). Learning cooperative persuasive dialogue policies using framing. *Speech Communication*, 84:83–96.

Howe, J. (2006). The rise of crowdsourcing. *Wired magazine*, 14(6):1–4.

Keltner, D. and Haidt, J. (1999). Social functions of emotions at four levels of analysis. *Cognition & Emotion*, 13(5):505–521.

Konar, A. and Chakraborty, A. (2015). *Emotion Recognition: A Pattern Analysis Approach*. Wiley.

Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). Applying conditional random fields to japanese morphological analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 4, pages 230–237.

Morris, M. W. and Keltner, D. (2000). How emotions work: The social functions of emotional expression

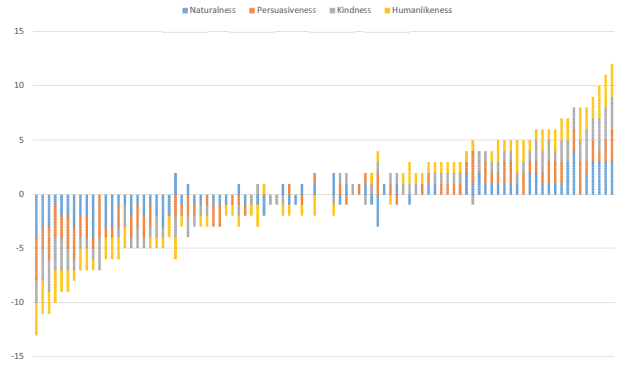


Figure 7: Overall evaluation of each example (w.emotion-w.o.emotion)

in negotiations. *Research in organizational behavior*, 22:1–50.

Nio, L., Sakti, S., Neubig, G., Toda, T., and Nakamura, S. (2014). Utilizing human-to-human conversation examples for a multi domain chat-oriented dialog system. *IEICE TRANSACTIONS on Information and Systems*, 97(6):1497–1505.

Reisenzein, R. (2009). Emotions as metarepresentational states of mind: Naturalizing the belief–desire theory of emotion. *Cognitive Systems Research*, 10(1):6–20.

Russell, J. A. (1978). Evidence of convergent validity on the dimensions of affect. *Journal of personality and social psychology*, 36(10):1152.

Sinaceur, M. and Tiedens, L. Z. (2006). Get mad and get more than even: When and why anger expression is effective in negotiations. *Journal of Experimental Social Psychology*, 42(3):314–322.

Takamura, H., Inui, T., and Okumura, M. (2005). Extracting semantic orientations of words using spin model. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 133–140. Association for Computational Linguistics.

Yu, Z., Xu, Z., Black, A. W., and Rudnicky, A. (2016). Chatbot evaluation and database expansion via crowdsourcing. In *Proceedings of the RE-WOCHAT workshop of LREC, Portoroz, Slovenia*.

SentiArabic: A Sentiment Analyzer for Standard Arabic

Ramy Eskander
Ramitechs
New York, USA
eskander@ramitechs.com

Abstract

Sentiment analysis has been receiving increasing interest as it conveys valuable information in regard to people's preferences and opinions. In this work, we present a sentiment analyzer that identifies the overall contextual polarity for Standard Arabic text. The contribution of this work is threefold. First, we modify and extend SLSA; a large-scale Sentiment Lexicon for Standard Arabic. Second, we build a sentiment corpus of Standard Arabic text tagged for its contextual polarity. This corpus represents the training, development and test sets for the proposed system. Third, we build a lightweight lexicon-based sentiment analyzer for Standard Arabic (SentiArabic). The analyzer does not require running heavy computations, where the link to the lexicon is carried out through a morphological lookup as opposed to conducting a rich morphological analysis, while the assignment of the sentiment is based on a simple decision tree that uses polarity scores as opposed to a more complex machine learning approach that relies on lexical information, while negation receives special handling. The analyzer is highly efficient as it achieves an F-score of 76.5% when evaluated on a blind test set, which is the highest results reported for that set, and an absolute 3.0% increase over a state-of-the-art system that uses deep-learning models.

Keywords: Sentiment Analysis, Sentiment Lexicon, Sentiment Corpus, Standard Arabic

1. Introduction

Sentiment analysis is the process of applying computational approaches to identify attitudes, emotions and opinions in text, speech and visual data. While there is an enormous number of sentiment analysis tools for English and other common languages, Arabic has received less focus due to lack of resources, corpora and lexicons in particular, in addition to the complexity of its morphological and syntactical systems, which incurs ambiguity and requires extensive processing.

In this work, we present SentiArabic, a lightweight lexicon-based sentiment analyzer for Standard Arabic. For a given text, the system identifies the contextual polarity (*Positive* or *Negative*) of the underlying sentiment. Our main contribution in this paper can be summarized as follow:

1. We modify and extend our previous sentiment resource SLSA, a large-scale Sentiment Lexicon for Standard Arabic (Eskander and Rambow, 2015). SLSA includes 34,281 entries, where an entry consists of a lemma, a part-of-speech (POS) tag, the corresponding English gloss, and three sentiment scores; positive, negative and objective, where the objective score is calculated as *1 - (positive score + negative score)*. The new extension follows the same structure but with a higher quality and better normalization of the polarity scores.

2. We create a new sentiment corpus of Standard Arabic text where each sentence is tagged for its polarity. The corpus is divided into three sets for the training, development and test of the sentiment analyzer.

3. We build SentiArabic, a lightweight sentiment analyzer for Standard Arabic. SentiArabic is based on the lexicon in (1), and is trained and evaluated based on the corpus in (2). SentiArabic is "lightweight" in the sense that it does not perform heavy computations on the underlying text. Instead, for every word in the input text, the corresponding lexicon entry is retrieved by using a maximum-likelihood lookup of the lemmas and POS tags as opposed to running morphological analysis in order to retrieve the lemma and POS information. The system then applies a decision tree of polarity scores in order to determine the overall polarity of the underlying text as opposed to conducting a more complex machine learning approach that uses lexical information. In addition, negation receives special handling as it affects the polarity of the following context by flipping the positive and negative scores.

The rest of the paper is structured as follows. We first review the related work in section 2, and then we present the implementation of our system in section 3. The evaluation and results are discussed in section 4, before we conclude with a discussion of future work in section 5.

2. Related Work

One of the early attempts on Arabic sentiment analysis is the work presented by (Abdul-Mageed et al., 2011), where they classified subjective text into four classes; positive, negative, neutral and mixed. The system applies Support Vector Machines (SVMs) on manually annotated data extracted from the Penn Arabic Treebank (PATB) (Maamouri et al., 2004), The authors showed that using a polarity lexicon and stem lemmatization has a considerable impact on

the performance, A subsequent system is SAMAR (Abdul-Mageed et al., 2014). SAMAR is based on a corpus of social-media text annotated for subjectivity and sentiment polarity, and uses a combination of language independent and language-specific feature sets. The authors conducted different evaluations on different parts of the corpus, and obtained an upper-bound F-score of 73.6% on polarity classification.

Mourad and Darwish (2013) used the MPQA lexicon (Wilson et al., 2005) to perform subjectivity and sentiment classification of both Standard Arabic news articles and dialectal Arabic microblogs from Twitter. The system applies Naive Bayes classification and achieves F-scores of 75.3% and 78.3% for the classification of subjectivity and sentiment, respectively.

Another system was presented by El-Makky et al. (2014). The system does sentiment classification for colloquial Arabic tweets, and uses a combination of lexicon-based sentiment orientation algorithms and supervised learning using SVMs. The system achieves an average F-score of 79.5%.

Nabil et al. (2015) manually annotated a dataset of about 10K Arabic tweets, and experimented with a set of machine learning techniques. The best results were obtained by applying SVMs on a set of unigram, bigram and trigram features, yielding a weighted F-score of 62.6%.

El-Beltagy and Ali (2013) proposed an unsupervised lexicon-based system for sentiment classification of Egyptian Arabic. The system relies on the positive and negative weights in the assignment of the overall contextual polarity, and achieves an overall accuracy of 81.8% on a dataset of 500 tweets.

Sallab et al. (2015) developed deep learning models for sentiment analysis of Standard Arabic. They tested the use of Recursive Auto Encoder models on a PATB dataset annotated for sentiment polarity by Abdul-Mageed et al. (2011). The system outperforms the state-of-the-art systems evaluated on the same dataset, where it achieves an F-score of 73.5%. In this paper we compare to this system, and show how our lightweight system outperforms by an additional F-score increase of 3.0% on the same dataset.

Eskander and Rambow (2015) proposed a lexicon-based sentiment analyzer that is based on SLSA and uses Linear SVMs for binary sentiment classification (positive and negative). The system achieves an F-score of 68.6% on the test set developed by Abdul-Mageed et al. (2011). We compare to this system as well.

Other systems are domain specific such as the system presented by Aly and Atiya (2013) for book reviews, and the system developed by A.M. Alayba (2017) for reviews on health services. Both systems apply supervised learning approaches using SVMs on large annotated datasets.

Task 7 of SemEval 2016 (Kiritchenko et al., 2016) aimed at identifying the sentiment intensity (scores from 0, maximum negative, to 1, maximum positive) of Arabic phrases given a set of 200 common terms from Arabic tweets and their polarity. The best performing system was proposed by the iLab-Edinburgh team (Refaee and Rieser, 2016), where they developed a hybrid system that is a combination of a rule-based approach in addition to off-the-shelf lexicons, and a machine learning approach that uses Linear Regression.

3. Approach

3.1. Lexicon Preparation

In this work, we modify and extend our sentiment lexicon SLSA, a large-scale Sentiment Lexicon for Standard Arabic (Eskander and Rambow, 2015). The construction of SLSA is based on linking the lexicon of the Standard Arabic morphological analyzer AraMorph (Buckwalter, 2004) with the English sentiment lexicon SentiWordNet (Baccianella et al., 2010) along with some heuristics and back-off techniques.

One key advantage of SLSA is its high coverage as it contains 34,852 lemma-POS pairs, which makes it the largest of its type (for Standard Arabic). Each lemma-POS pair is associated with three sentiment scores (positive, negative and objective), in addition to the English gloss. Another advantage is it richness as it is a lemma-based resource that attaches the POS and English gloss information to lemmas, where the sentiment of a lemma is applicable to its corresponding inflected forms. Additionally, the creation of SLSA was fully automated based on previous resources. This makes the generation of future updates straightforward upon improving the quality of the resources.

The overall projected accuracy of the entries in SLSA is 80.1%. About 93% of the erroneous entries are cases where the sentiment scores are doubtful in SentiWordNet, while the other errors are due to incorrect glosses in AraMorph.

Since our proposed system is lexicon-based, the higher accuracy the lexicon has, the better the classification of the sentiment polarity is. Accordingly, we develop a new extension of SLSA where we manually correct SLSA entries that correspond to the most frequent 4,000 lemmas in the Penn Arabic Treebank (PATB), Part 3 (Maamouri et al., 2004), Moreover, 220 new entries are added into SLSA. Those entries correspond to lemma-POS combinations that are absent in Aramorph.

Additionally, instead of having sentiment scores ranging from 0 to 1, the scores are better normalized by rounding to their 0.25 ceilings. This means the new positive and negative scores are of the values 0.0, 0.25, 0.5, 0.75 and 1.0. The reason for the rounding is to make the system less sensitive to the little differences in the automatically generated sentiment scores. Table 1 shows examples of SLSA entries after manual correction and score normalization, where the Arabic text is transliterated using the Buckwalter scheme (Buckwalter, 2004).

Lemma	POS	English Gloss	+ve Score	-ve Score	Obj. Score
niEom_1	NOUN	wonderful	1	0	0
mubArak_1	ADJ	blessed/fortunate	0.75	0	0.25
tawaE∼aY_1	VERB	be attentive/cautious	0.5	0	0.5
AiHotiwA'_1	NOUN	inclusion/content	0.25	0	0.75
$ahoriy∼_1	ADJ	monthly	0	0	1
katab_1	VERB	write	0	0	1
munAqaDap_1	NOUN	contradiction/contrast	0	0.25	0.75
lawom_1	NOUN	blame/censure	0	0.5	0.5
dana>_1	VERB	be vile/be despicable	0	0.75	0.25
kamod_1	NOUN	swarthiness/sadness	0	1	0

Table 1: Examples of SLSA entries after manual correction and score normalization. The first column represents the lemma written in the Buckwalter transliteration. The second column is the POS tag of the lemma. The third column is the corresponding English gloss. The other three columns represent the positive, negative and objective scores, respectively, where the objective score is defined as *1 - (positive score + negative score)*.

3.2. Corpus Development

We develop a new corpus of Standard Arabic text where each sentence is manually annotated for its contextual polarity (positive, negative or neutral). The corpus contains 4,000 sentences generated from news websites. The context of the sentences is varied to cover several genres such as politics, arts, sports, fashion, religion and medicine. In the case where a sentence has both a positive and a negative sentiment, the sentiment that is more dominant within the context is assigned. Table 2 lists example sentences of different polarities and genres from the corpus.

The corpus is split into three sets for training (3,200 sentences), development (400 sentences) and testing (400 sentences), namely; SentiTrain, SentiDev and SentiTest, respectively, where each set has examples of the different genres. SentiTrain is used for the supervised training of the system, where the text is linked to entries in the sentiment lexicon through the lemma and POS information. SentiDev is then used for tuning the system, while SentiTest is a blind set that is used solely for testing.

Sentence	Polarity
لا تقلق، أنا بخير. Do not worry, I am fine.	Positive
اعتادت أن تسحرنا بصوتها الملائكي الحالم. She used to charm us with her dreamy angelic voice.	Positive
لكنهم في وقت لاحق سيندمون. But they will later regret.	Negative
لكننا لم نتمكن من تحقيق الفوز، للأسف. But we were unable to win, unfortunately.	Negative
جميعنا يستقل المصعد بشكل يومي تقريباً. We all take the elevator almost daily.	Neutral
ومن المتوقع أن يعلن عن الصفقة الاثنين. The deal is expected to be announced on Monday.	Neutral

Table 2: A sample of corpus sentences annotated for their contextual polarity.

3.3. Sentiment Classification

The purpose of the proposed system, SentiArabic, is to determine whether a given text expresses a positive or a negative sentiment, so our focus is on the sentences that have either a positive or a negative polarity. We train a supervised model based on SentiTrain, described in subsection 3.2, where a sentence represents a training unit, and the output class is the sentiment polarity.

First, for each sentence in the training corpus, we use the sentiment lexicon to extract the positive and negative scores that correspond to each word in it (zero values are used as defaults). However, since the words in the corpus are surface forms (i.e., inflected), while the lexicon has entries of lemma and POS values, it is required to run a morphological tagger that analyzes each word in context in order to obtain its lemma and POS information. One drawback in this approach is the extensive computation the morphological tagging requires due to the morphological complexity of Arabic, which violates the *lightweightness* aspect of the overall system. Instead, we obtain the POS and lemma information by looking up the word in a maximum likelihood lookup that is based on PATB, Parts 1, 2 and 3. That means, the lemma and POS for a given word are chosen to be the most frequent analysis the word has received in PATB. However, we show in section 4 that the use of a morphological tagger increases the overall performance of the system by an absolute F-score of only 0.5%, which is not a significant improvement given the computational overhead of running a morphological tagger.

After extracting the positive and negative scores for each word in a sentence, the following features are extracted and used as the sentence representation:

- *Average Positive Score per Subjective Word*
- *Average Negative Score per Subjective Word*
- *Average (Positive - Negative) Score per Subjective Word*
- *Percentage of Positive Words to Subjective Words*

- *Percentage of Negative Words to Subjective Words*

A subjective word has an objective score of zero, i.e., it is either positive, negative or both, where a word is positive if its positive score is greater than zero, and it is negative if the negative score is greater than zero. One special case is negation, where it flips the positive and negative scores of the following context, if any.

It is worth noting that all the features are simple numerical ones, where none is a lexical representation of the context. The reason behind this is to decrease the computation involved in our machine learning approach toward a lightweight sentiment analyzer.

We experiment with different machine learning techniques such as SVMs, Naive Bayes, K-Nearest Neighbors, Random Trees, and Decision Trees. The latter gives the highest accuracy and F-score on SentiDev (78.7% and 78.3%, respectively), which in turn serves our purpose of developing a lightweight system that requires minimal computation.

4. Evaluation and Results

We evaluate SentiArabic on two blind test sets; SentiTest, described is subsection 3.2, and a PATB dataset annotated for sentiment polarity by Abdul-Mageed et al. (2011). The latter is used to evaluate the sentiment analyzer by Eskander and Rambow (2015) and the state-of-the-art deep learning models for Arabic sentiment analysis by Sallab et al. (2015) (see section 2).

We first run the system using the original SLSA without the correction and normalization we applied in subsection 3.1. We call this system SentiArabic-NoExt. SentiArabic-NoExt gives an overall accuracy and a weighed F-score of 67.6% and 67.0%, respectively, when evaluated on SentiTest. These numbers increase to 81.1% and 80.8%, in order, when testing with the new extension of SLSA, which is a significant error reduction of 41.6%.

When evaluated on the PATB test set, SentiArabic achieves an accuracy of 76.7% and a weighted F-score of 76.5%. This is an absolute F-score increase of 3.0% over the state-of-the-art analyzer by Sallab et al. (2015) and 7.9% over our previous analyzer (Eskander and Rambow, 2015). Although the system by Sallab et al. (2015) performs heavy computations for the deep learning models, while SentiArabic relies on lookups and a simple decision tree, the efficiency of SentiArabic is highly supported by the high accuracy of the new extension of SLSA. This is no surprise given the performance of SentiArabic versus SentiArabic-NoExt, while the original SLSA lexicon is highly efficient compared to its counterparts (Eskander and Rambow, 2015).

Table 1 summarizes the results for SentiArabic, SentiArabic-NoExt, the system by Eskander and Rambow (2015) and the system by Sallab et al. (2015) on both SentiTest and PATB test sets (when possible).

Finally, we run an additional variation of the system where the lemma and POS information of each word is generated in context by running the state-of-the-art Arabic morphological tagger MADAMIRA (Pasha et al., 2014) instead of looking up the lemma and POS information in the PATB-based lookup (see subsection 3.3). This is in order to compare to a variation of the system that runs rich in-context computation. Using MADAMIRA increases the overall weighted F-score by an absolute 0.5%. However, we find the increase insignificant given the computational overhead incurred by MADAMIRA, which is in favor of the lightweight SentiArabic analyzer.

We conducted an error analysis on all the entries in SentiTest that are misclassified by the analyzer. About 15% of the erroneous cases are sentences that involve both a positive and a negative sentiment where the system picks the sentiment that is less dominant in the context.

	SentiTest		PATB	
System	**Acc. %**	**F1 %**	**Acc. %**	**F1 %**
Eskander and Rambow (2015)	–	–	–	68.6
Sallab et al. (2015)	–	–	74.3	73.5
SentiArabic-NoExt	67.6	67.0	70.4	70.8
SentiArabic	**81.1**	**80.8**	**76.7**	**76.5**

Table 3: Sentiment analysis results of SentiArabic (with and without lexicon extension) on two test sets; SentiTest and PATB, compared to two state-of-the-art systems by Eskander and Rambow (2015) and Sallab et al. (2015). SentiArabic outperforms both systems by absolute F-scores of 7.9% and 3.0%, respectively. *(Unreported results are marked as "–" .)*

5. Conclusion and Future Work

We proposed SentiArabic, a new lightweight lexicon-based sentiment analyzer for Standard Arabic. SentiArabic avoids running heavy computation by exploiting a morphology lookup along with a simple decision tree for classification As part of developing the system, we built a new extension of the SLSA lexicon that has a higher quality and better normalization of the polarity scores. We also created a new corpus that is tagged for contextual polarity, where the text covers a wide range of genres.

SentiArabic achieves an F-score of 76.5% when tested on a blind test set annotated by Abdul-Mageed et al. (2011), which is the highest result reported for that set, and an absolute 3.0% increase over a state-of-the-art system that uses deep learning models (Sallab et al., 2015).

The future plans include working on the SLSA extension to manually check all the entries and correct the erroneous cases, in addition to augmenting the lexicon with multiword expressions. We also plan to extend our work to cover other Arabic dialects, Egyptian, Levantine and Gulf in particular.

6. Bibliographical References

Abdul-Mageed, M., Diab, M., and Korayem, M. (2011). Subjectivity and sentiment analysis of modern standard arabic. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon.

Abdul-Mageed, M., Diab, M., and Kubler, S. (2014). Samar: Subjectivity and sentiment analysis for arabic social media. *Computer Speech and Language*, 28:20–37.

Aly, M. and Atiya, A. (2013). Labr: A large scale arabic book reviews dataset. In *Proceedings of the 11th International Workshop on the ACL2 Theorem Prover and Its Applications*, Laramie, Wyoming.

A.M. Alayba, V. Palade, M. E. R. I. (2017). Arabic language sentiment analysis on health services. In *Proceedings of the 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, Nancy, France.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th international conference on Language Resources and Evaluation (LREC)*, Valletta, Malta.

T. Buckwalter. (2004). *Buckwalter Arabic Morphological Analyzer (Aramorph) Version 2.0*. Linguistic Data Consortium (LDC), 2.

El-Beltagy, S. and Ali, A. (2013). Open issues in the sentiment analysis of arabic social media: a case study. In *Proceedings of the 9th International Conference on Innovations in Information Technology (IIT)*, Al Ain, AUE.

El-Makky, N., Nagi, K., El-Ebshihy, A., and Ibrahim, S. (2014). Sentiment analysis of colloquial arabic tweets. In *Proceedings of the 3rd ASE International Conference on Social Informatics*, Boston, Massachusetts.

Eskander, R. and Rambow, O. (2015). Slsa: A sentiment lexicon for standard arabic. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal.

Kiritchenko, S., Mohammad, S., and Salameh, M. (2016). Semeval-2016 task 7: Determining sentiment intensity of english and arabic phrases. In *Proceedings of the 2016 International Workshop on Semantic Evaluation (SemEval)*, San Diego, California.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). Arabic treebank: Building a large-scale annotated arabic corpus. In *Proceedings of the NEMLAR Conference on Arabic Language Resources and Tools*, Cairo, Egypt.

Mourad, A. and Darwish, K. (2013). Subjectivity and sentiment analysis of modern standard arabic and arabic microblogs. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Atlanta, Georgia.

Nabil, M., Aly, M., and Atiya, A. (2015). Astd: Arabic sentiment tweets dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Lisbon, Portugal.

Pasha, A., Al-Badrashiny, M., Diab, M., Kholy, A. E., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *Proceedings of the 9th international conference on Language Resources and Evaluation (LREC)*, Reykjavik, Iceland.

Refaee, E. and Rieser, V. (2016). ilab-edinburgh at semeval-2016 task 7: A hybrid approach for determining sentiment intensity of arabic twitter phrases. In *Proceedings of the 2016 International Workshop on Semantic Evaluation (SemEval)*, San Diego, California.

Sallab, A. A., Baly, R., Badaro, G., Hajj, H., and W. El-Hajj, K. S. (2015). Deep learning models for sentiment analysis in arabic. In *Proceedings of the EMNLP 2015 Workshop on Arabic Natural Language Processing (ANLP)*, Lisbon, Portugal.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the 1st Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Vancouver, Canada.

Contextual Dependencies in Time-Continuous Multidimensional Affect Recognition

Dmitrii Fedotov [1,2], **Denis Ivanko** [1,2], **Maxim Sidorov** [1], **Wolfgang Minker** [1]
[1] Ulm University, [2] ITMO University
Ulm Germany, St. Petersburg Russia
{dmitrii.fedotov, wolfgang.minker}@uni-ulm.de,
denis.ivanko11@gmail.com, maxim.sidorov@alumni.uni-ulm.de

Abstract

Modern research on emotion recognition often deals with time-continuously labelled spontaneous interactions. Such data is much closer to real world problems in contrast to utterance-level categorical labelling in acted emotion corpora that have widely been used to date. While working with time-continuous labelling, one usually uses context-aware models, such as recurrent neural networks. The amount of context needed to show the best performance should be defined in this case. Despite of the research done in this field there is still no agreement on this issue. In this paper we model different amounts of contextual input data by varying two parameters: sparsing coefficient and time window size. A series of experiments conducted with different modalities and emotional labels on the RECOLA corpora has shown a strong pattern between the amount of context used in model and performance. The pattern remains the same for different pairs of modalities and label dimensions, but the intensity differs. Knowledge about an appropriate context can significantly reduce the complexity of the model and increase its flexibility.

Keywords: time-continuous affect recognition, affective context analysis, multimodal emotion recognition.

1. Introduction

Real life human-human interaction consists of two main aspects: information contained in speech and emotion expressed by humans. Speech recognition techniques allow computers to understand human speech but they lack an emotional component. Exactly the same words or phrases may be a statement, a question or a guess if said with different emotions. These types of phrases should be recognised and processed by dialogue system differently. It is necessary for computers to understand human emotions in order to succeed in interaction with them.

Modern research focus on natural interaction between computer-based systems and humans. These systems try to understand non-standardized questions and provide answers, similar to the spontaneous interaction between two humans. One of the most important parts of human understanding is the ability to identify and react to emotions. Emotion recognition may significantly improve the quality of human-computer interaction, speech recognition systems and artificial intelligence in general.

Previous research on emotion recognition mostly dealt with utterance-level categorical data labelling, i.e. each data sample had one label from the list, e.g. anger, happiness, neutral, etc. However, recent research has focused on the dimensional time-continuous data that provides more flexibility and precision of emotion definition. This type of data requires more complex models and the definition of additional parameters, such as the amount of context to be used. Despite of the research conducted in this area, it is still an open question, how much previous data do the system need to provide the best performance. Studies on the effect of this parameter will help to build an effective end-to-end real time emotion recognition system.

As shown in this paper, there is a strong correlation between the amount of context and the performance of an emotion recognition system despite of the amount of data i.e. the time steps used.

This paper is structured as follows: Section 2 provides an overview of research related to multi-dimensional time-continuous emotion recognition; Section 3 details the data used in this study as well as the pre- and postprocessing procedures; in Section 4 the methodology used is described; in Section 5 experimental results are shown and analysed; conclusions from this study and proposed future research are presented in Section 6, followed by acknowledgements in Section 7.

2. Related work

Previous research on emotion recognition mostly dealt with utterance-level categorically labelled databases. (Haq and Jackson, 2010; Burkhardt et al., 2005; Makarova and Petrushin, 2002). Corpora with time-continuous labelling emerged in the past years and they gain popularity among researchers (Schroeder et al., 2012; Ringeval et al., 2013). Time-continuous emotion recognition provides more flexibility for the system, but also creates new challenges. Firstly, is the amount of previous information that should be used to model emotions. According to Levenson, 1988 it should be a value between 0.5 and 4 seconds, but is still remains an open question and depends on modality and emotional dimension (Gunes and Pantic, 2010).

Another issue refers to the labelling process of emotional interactions. Annotations of emotions are performed by humans, hence, they yield a significant level of subjectivity and a suitable method is required for computing a gold standard. It can be based on correlation between individual ratings provided by annotators (Mariooryad and Busso, 2013; Nicolle et al., 2012).

When annotating time-continuous emotions, a reaction lag may also appear; therefore, it should be considered when synchronising features and labels. It may be done by maximising the correlation between some features and emotional ratings and/or ratings from annotators (Nicolaou et al., 2010; Mariooryad and Busso, 2014).

Another related issue is the appropriate sampling frequency of data. Original data can be upsampled or downsampled to thr required value of frequency (Nicolaou et al., 2011; Metallinou et al., 2011). In this paper we used data sparsing to study the effect of amount of data combined with its intensity on system performance.

3. Data and data-related procedures

The recently introduced time-continuously labelled corpus of spontaneous interaction in French called RECOLA (Remote COLlaborative and Affective interactions) (Ringeval et al., 2013) was used in this paper.

3.1 Corpus description

The RECOLA database was collected during the resolving of a cooperative problem. It consists of spontaneous interactions between 23 dyadic pairs of French-speaking participants, i.e. 46 persons. 34 participants gave their consent to share the data. The dataset was therefore reduced from originally recorded 9.5 to 7 hours. Annotations for 23 of them are publically available in current version of database and were used in this research.

The participants were aged between 18 and 25 years and have different mother tongue although they spoke French during recorded interactions: 17 of them had French as a mother tongue, 3 – Italian and 3 – German.

RECOLA consists of recordings in 4 modalities: audio, video, electro-cardiogram and electro-dermal activity. Interactions were evaluated by 6 equally gender distributed French speaking annotators through ANNEMO (ANNotating EMOtions) tool (Ringeval et al., 2013). The annotations include two emotional (arousal and valence) and five social (agreement, dominance, engagement, performance and rapport) behaviour dimensions.

3.2 Features

Audio features were extracted with the openSMILE open-source software. The feature set consists of 3 groups of low-level descriptors (LLDs): prosodic, spectral, cepstral and voice quality. (Eyben et al., 2010; Schuller et al., 2014). There are 65 LLDs and along with their first order derivate we have 130 features in total.

The visual feature set consists of 20 LLDs and their first order derivate for each frame available. It includes 15 facial action units, 3-dimansional head pose and the mean and standard deviation of the optical flow in the region around the head (Ringeval et al., 2013).

3.3 Data preprocessing

The available part of the dataset was divided into 2 speaker disjoint subsets: train and evaluation. Subsets maintain age, gender and mother tongue distribution of original set (see Table 1).

Set	Age μ(σ)	Gender	Mother tongue
Full	21.35 (2.04)	10 males 13 females	17 French 3 Italian 3 German
Train	21.38 (2.13)	7 males 9 females	12 French 2 Italian 2 German
Evaluation	21.29 (1.98)	3 males 4 females	5 French 1 Italian 1 German

Table 1: Partitioning of RECOLA database into train and evaluation subsets

3.3.1 Features and labels preprocessing

Provided audio and video features were normalised with the Z-transformation based on the train subset.

There are two ways of using several ratings for each recording: merge them into a gold-standard rating or train model to produce separate predictions. The first approach was used in this paper. The gold-standard may be calculated by simple averaging the values provided by each evaluator. However, this methodology may lead to a loss of information contained in annotator's perception of emotions. It effects the spread of emotional ratings as well as their "neutral" values, i.e. bias. For example, some annotators perceive emotions in a mild manner (e.g. Annotator 6) rather than in a strong one (Annotator 1). At Figure 1 a diversity of ratings from annotators of RECOLA database for all recordings available is shown.

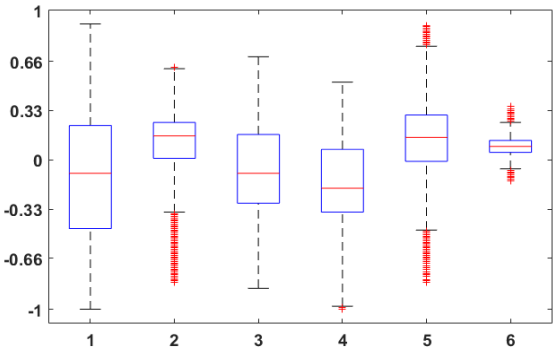


Figure 1: Ratings diversity from 6 annotators of RECOLA database

Taking these differences into account, the gold-standard was based on the maximisation of the inner-rater agreement (Mencattini et al., 2016).

While estimating emotions continuously, evaluators need some time to report the changes. The delay between an actual change of emotional behaviour and the moment it is annotated is called reaction lag (RL). It is not consistent for different speakers and label dimensions, although has negligible variation for different annotators (Mariooryad and Busso, 2014).

The value of RL may be calculated based on the correlation of features with labels. Some features are strongly correlated with positive values of particular labels, some with negative. In previous research the *RL* was found to be 3.89 s for arousal and 4.52 s for valence (Mencattini et al., 2016; Ringeval et al., 2015; Mariooryad and Busso, 2014). The value of RL for arousal was corrected to be 3.88 due to label rate (25 Hz).

Gold-standard labels were shifted backwards according to RL values mentioned above. The label values for the last frames of each speaker were lost after shifting and replaced with zeros. Labels were normalised with Z-transformation based on train subset and denormalised at estimation stage.

3.3.2 Contextual pre- and postprocessing and sparsing

To meet the requirements of the context-based model, features and labels were preprocessed from [samples × features(labels)] representation to [samples × time steps × features(labels)]. Time steps were taken only backwards for both features and labels. The time window size (TW) defines the number of previous steps to be taken for every sample in set. Previous frames could be used only if they exist and relate to the same speaker as the current frame, otherwise zero-padding was applied.

This procedure was combined with sparsing. If the sparsing coefficient (SC) is greater than one, then every *n-th* frame is taken into account at the stage of adding time steps. For example, for sample *t* with *SC=3* and *TW=6* the following frames are chosen at contextual preprocessing stage: *[t-15, t-12, t-9, t-6, t-3, t]*.

The combination of *TW* and *SC* define the amount of context (in seconds) used by the model:

$$C = \frac{SC \times TW}{frame\ rate}$$

The same context may be represented with different pairs of *SC* and *TW*. For example, the context is equal to 24 frames with [*SC=3, TW=8*] and [*SC=1, TW= 24*]. To make a uniform grid, the following values of contextual parameters were chosen: *TW = {2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 64}, SC = {1, 2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 64}.*

A sequence-to-sequence approach was applied in this research, i.e. features of *TW* previous frames were used to make a prediction of labels for the same *TW* previous frames. When the predictions were made, values obtained for the same frame at different time steps were average to smooth the final prediction.

4. Methodology

Traditional feed-forward neural networks are not capable of using the previous information as they lack feedback connections in their architecture. This leads to constraining the knowledge about the data only to the current frame. To overcome this problem, recurrent neural networks (RNN) with one time step of delay were introduced. However, networks of this type suffer from the vanishing gradient problem and cannot store the information about more than approximately 10 time steps (Hochreiter et al., 2001). To avoid the problem of exponentially decaying gradients, a long-short term memory recurrent neural network (LSTM) was introduced (Hochreiter and Schmidhuber, 1997) and then improved (Graves and Schmidhuber, 2005). LSTM-RNNs use a fine regulation of the system state by special gates: input gate, output gate and forget gate. These gates allow to accumulate the information about previous time steps over long duration and drop the information when needed. The weights of self-loops are not fixed, but based on the gates which allows to change the level of data integration.

Two LSTM layers with 20 and 15 neurons respectively were used in this research. The LSTM blocks had a ReLU activation function (Vinod and Hinton, 2010) and the neurons of the output layer had a simple linear activation function. To avoid overfitting, recurrent layers were followed by the dropout layers (Srivastava et al., 2014). Different dropout probability values were studied and *p=0.1* was selected as it provided the best results. The LSTM models were optimized by root mean square propagation (RMSprop) using the concordance correlation coefficient (CCC) as a metric function. LSTM implementation is provided by Keras (Chollet, 2015).

Our previous research has shown, that the performance of the system based on RNN significantly depends on the learning rate of optimiser. While too high values of the learning rate cannot provide any appropriate performance (zero performance), rather small ones may result in slow learning and the system may permanently get stuck. The value of learning rate, that provides the lowest loss is usually the highest one before the "zero performance" values, (see Figure 2).

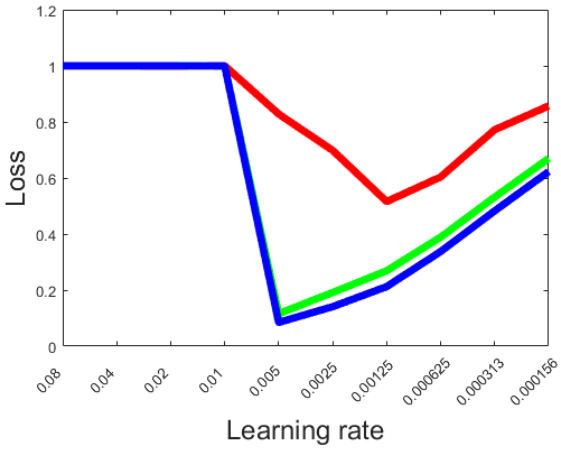


Figure 2: Learning rate against loss.
Red – after 1st epoch; green – after 20th epoch; blue – after 100th epoch.

The appropriate learning rate varies with the number of previous time steps used; therefore, it cannot be fixed. As it has a critical effect on performance, an automatic procedure of the learning rate selection was developed. According to LeCun et al. (1998), the search of the best learning rate was conducted with the decreasing factor of 2. The following values were used: 0.08, 0.04, 0.02, 0.01, 0.005, 0.0025, 0.00125, 6.25e-4, 3.13e-4, 1.56e-4. The model training was started with the highest learning rate. The value of a loss function was calculated on the train sample after the first epoch. If the loss was greater than a predefined threshold (empirically set to *t=0.85*), training was terminated and the next learning rate was tried out. The training loss was further monitored during the training process and if the loss at current epoch was greater than at the first one, the same procedure of training termination was applied.

5. Experiments and results

A series of experiments was conducted to study the impact of context on the performance of an emotion recognition system. All pairs of contextual parameters described above were used for audio and video modalities as well as their feature-based multimodal fusion for two emotional dimensions.

The performance of the emotion recognition system on evaluation subset was estimated with CCC. The results are shown at Figure 3. The results were obtained for each pair of sparsing coefficient and time window size and interpolated afterwards to create a surface of system performance, indicated with colour map. Diagonal lines represent the amount of used context in seconds. Red stars show the best sparsing coefficient that led to the best performance of system with each value of *TW*. The red stars in a circle represent the best performance obtained within the provided problem definition. One may notice a strong pattern between the amount of context and performance of the emotion recognition system. It is especially obvious with Audio-Arousal pair. Patterns may still be noticed in Audio-Valence and Video-Valence pairs.

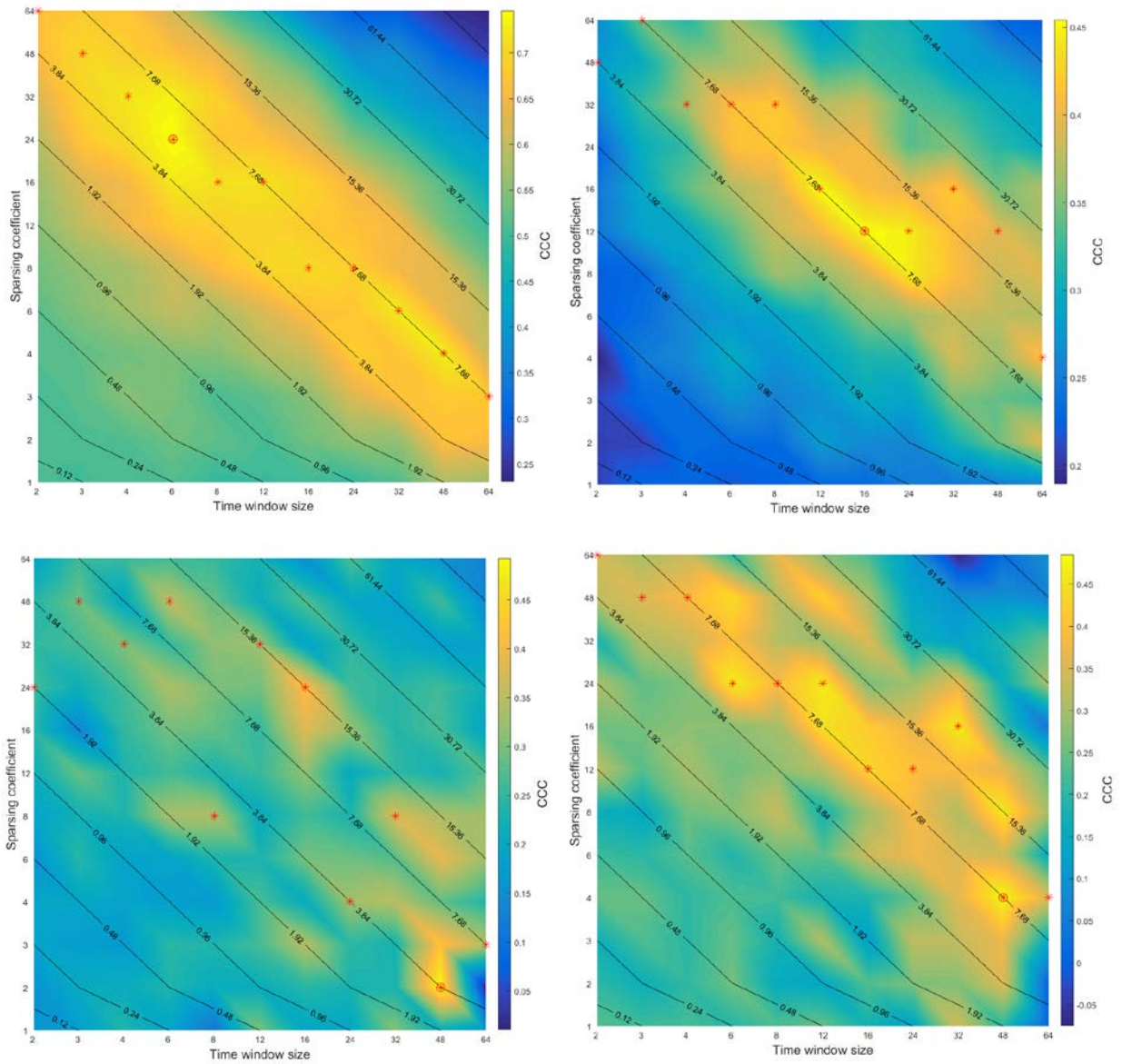


Figure 3. Contextual dependencies of performance in RECOLA database.
Top-left: Audio-Aruosal; Top-right: Audio-Valence; Bottom-left: Video-Arousal; Bottom-right: Video-Valence.

Results for Video-Arousal do not show any trustworthy trends and the emotion recognition system does not perform well with this modality-label pair. The same methodology but with 80 and 60 neurons respectively was tried and it showed the same trends.

Patterns for different modality-label pairs are similar and the best results are lying in the area of approximately 6 seconds of context for arousal and 8 seconds for valence. Performance of the system obtained with different combinations of contextual parameters does not differ much; therefore, less amount of data may be used to obtain the same high results.

6. Conclusion

Experiments have shown a strong pattern between the amount of context and the performance of an emotion recognition system. Sparsing does not affect performance much, while allowing to use more simple and flexible models and get results much faster. The knowledge about sparsing coefficients may reduce the number of time steps for RNNs to 6-12. The information about the appropriate amount of required contextual data may be used in real-time emotion recognition systems.

Further research will be focused on other time-continuously labelled corpora, such as SEMAINE and context-aware models (e.g. Hidden Markov Models and Gated Recurrent Units).

7. Acknowledgements

The work presented in this paper was partly supported by the DAAD (German Academic Exchange Service) within the different programmes and by the Transregional Collaborative Research Centre SFB/TRR 62 “Companion-Technology for Cognitive Technical Systems” which is funded by the German Research Foundation (DFG).

8. Bibliographical References

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. F., & Weiss, B. (2005). A database of German emotional

speech. In Ninth European Conference on Speech Communication and Technology.
Chollet F. Keras. (2015). URL: https://github.com/fchollet/keras.
Eyben, F., Wöllmer, M., & Schuller, B. (2010). Opensmile: the munich versatile and fast open-source audio feature extractor. In Proceedings of the 18th ACM international conference on Multimedia, pp. 1459-1462.
Gunes, H., Pantic, M. (2010). Automatic, dimensional and continuous emotion recognition. Inter. Journal of Synthetic Emotions 1, pp. 68–90.
Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5-6), pp. 602-610.
Haq, S. and Jackson, P., (2010). Machine Audition: Principles, Algorithms and Systems, chapter Multimodal Emotion Recognition, IGI Global, Hershey, pp. 398-423.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), pp. 1735-1780.
Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.
LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), pp. 2278-2324.
Levenson, R. W. (1988). Emotion and the autonomic nervous system: A prospectus for research on autonomic specificity. Social psychophysiology: Theory and clinical applications., pp. 17–42.
Makarova, V. and Petrushin, V. (2002). Ruslana: A database of russian emotional utterances. In Proc. Int. Conf. Spoken Language Processing.
Mariooryad, S., Busso, C. (2013). Analysis and compensation of the reaction lag of evaluators in continuous emotional annotations, in: Proc. of the 5th Inter. Conf. on Affective Computing and Intelligent Interactions (ACII), Geneva, Switzerland. pp. 85–90.
Mariooryad, S., & Busso, C. (2015). Correcting time-continuous emotional labels by modeling the reaction lag of evaluators. IEEE Transactions on Affective Computing, 6(2), pp. 97-108.
Mencattini, A., Martinelli, E., Ringeval, F., Schuller, B., & Di Natale, C. (2017). Continuous estimation of emotions in speech by dynamic cooperative speaker models. IEEE transactions on affective computing, 8(3), pp. 314-327.
Metallinou, A., Katsamanis, A., Wang, Y., & Narayanan, S. (2011). Tracking changes in continuous emotion states using body language and prosodic cues. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, pp. 2288-2291
Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning, pp. 807-814.
Nicolaou, M., Gunes, H., Pantic, M. (2010). Automatic segmentation of spontaneous data using dimensional labels from multiple coders, in: Proc. of the LREC 2010 Workshop on Multimodal Corpora: Advances in Capturing, Coding and Analyzing, Istambul, Turkey. pp. 43-48.
Nicolaou, M. A., Gunes, H., & Pantic, M. (2011). Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space. IEEE Transactions on Affective Computing, 2(2), pp. 92-105.
Nicolle, J., Rapp, V., Bailly, K., Prevost, L., & Chetouani, M. (2012). Robust continuous prediction of human emotions using multiscale dynamic cues. In Proceedings of the 14th ACM international conference on Multimodal interaction, pp. 501-508.
Ringeval, F., Eyben, F., Kroupi, E., Yuce, A., Thiran, J. P., Ebrahimi, T. & Schuller, B. (2015). Prediction of asynchronous dimensional emotion ratings from audiovisual and physiological data. Pattern Recognition Letters, 66, pp. 22-30.
Schroder, M., Bevacqua, E., Cowie, R., Eyben, F., Gunes, H., Heylen, D. & Pelachaud, C. (2012). Building autonomous sensitive artificial listeners. IEEE Transactions on Affective Computing, 3(2), pp. 165-183.
Schuller, B., Steidl, S., Batliner, A., Epps, J., Eyben, F., Ringeval, F., Marchi, E., Zhang, Y. (2014). The INTERSPEECH 2014 computational paralinguistics challenge: Cognitive & physical load, in: Proc. of INTERSPEECH 2014, 15th Annual Conf. of the Inter. Speech Communication Association (ISCA).
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), pp. 1929-1958.

WikiArt Emotions: An Annotated Dataset of Emotions Evoked by Art

Saif M. Mohammad and Svetlana Kiritchenko
National Research Council Canada
{saif.mohammad,svetlana.kiritchenko}@nrc-cnrc.gc.ca

Abstract

Art is imaginative human creation meant to be appreciated, make people think, and evoke an emotional response. Here for the first time, we create a dataset of more than 4,000 pieces of art (mostly paintings) that has annotations for emotions evoked in the observer. The pieces of art are selected from WikiArt.org's collection for four western styles (Renaissance Art, Post-Renaissance Art, Modern Art, and Contemporary Art). The art is annotated via crowdsourcing for one or more of twenty emotion categories (including neutral). In addition to emotions, the art is also annotated for whether it includes the depiction of a face and how much the observers like the art. The dataset, which we refer to as the *WikiArt Emotions Dataset*, can help answer several compelling questions, such as: what makes art evocative, how does art convey different emotions, what attributes of a painting make it well liked, what combinations of categories and emotions evoke strong emotional response, how much does the title of an art impact its emotional response, and what is the extent to which different categories of art evoke consistent emotions in people. We found that fear, happiness, love, and sadness were the dominant emotions that also obtained consistent annotations among the different annotators. We found that the title often impacts the affectual response to art. We show that pieces of art that depict faces draw more consistent emotional responses than those that do not. We also show, for each art category and emotion combination, the average agreements on the emotions evoked and the average art ratings. The WikiArt Emotions dataset also has applications in automatic image processing, as it can be used to develop systems that detect emotions evoked by art, and systems that can transform existing art (or even generate new art) that evokes the desired affectual response.

Keywords: art, images, emotions, image retrieval, emotion analysis, crowdsourcing, Renaissance art, modern art, image generation

1. Introduction

Art is imaginative human creation meant to be appreciated, make people think, and evoke an emotional response. Paintings are a popular form of art with a long and compelling history. (The paintings in the Chauvet-Pont-d'Arc Cave in southern France are about 32,000 years old.) Studies have also shown that using art to evoke an emotional response (and being creative, in general) are desired fitness attributes that have played a role in the natural selection of humans (Davies, 2012; Dutton, 2009; Miller, 2001; Aiken, 1998). Nonetheless, many of the mechanisms behind how and when paintings evoke emotions remain elusive. Several research questions remain unanswered, such as which emotions are commonly elicited by art?, why are some paintings more evocative than others?, why we like some paintings but not others?, and what is the relationship between the emotion an art evokes and how much we like it?

Museums across the world house hundreds of thousands of pieces of art and attract millions of visitors each year.[1] Yet, they display only a fraction of the art they own due to space constraints. Thus, a number of museums now have a substantial online presence. Availability of massive amounts of art online means that it is useful to have the ability to search for art with various attributes. Paintings are usually labeled with the title, the artist, and the style of painting, but they are not categorized for the emotions they evoke. Thus, automatically detecting the emotions evoked by art is of considerable importance. It can be used for organizing paintings by the emotions they evoke, for recommending paintings that accentuate or balance a particular mood, and for searching paintings of a certain style or genre that depict user-determined content in a user-determined affectual state (e.g., a Post-Renaissance painting showing angry peasants).

Paintings can be created in one of many styles such as realism, cubism, expressionism, minimalism, etc. They can belong to different genres such as still life, landscape, abstract, allegorical, figurative, etc. WikiArt.org displays 151,151 pieces of art (mostly paintings) corresponding to ten main art styles and 168 style categories.[2] We created the *WikiArt Emotions Dataset*, which includes emotion annotations for more than 4,000 pieces of art available on the WikiArt.org. The art in the WikiArt Emotions Dataset is from four western styles and twenty-two style categories (as listed in Table 2). The pieces of art are annotated for one or more of twenty emotion categories (as listed in Table 3). These emotion categories were chosen from the psychology literature on the theories of basic emotions (Ekman, 1992; Plutchik, 1980; Parrot, 2001) and on the theories of emotions elicited by art (Silvia, 2005; Silvia, 2009; Millis, 2001; Noy and Noy-Sharav, 2013). We obtained separate emotion annotations for when the observer sees only the image, sees only the title of the art, and sees both title and art together. In addition to emotions, the art is annotated for whether it includes the depiction of a face and the extent to which the observers liked the art—the *average art rating*.

We found that anticipation, fear, happiness, humility, love, optimism, sadness, surprise, and trust were frequently chosen as the emotions evoked by art. Fear, happiness, love, and sadness were also the emotions for which the annotators provided the most consistent labels. Other emotions were also found to be more frequent and consistently

[1] https://en.wikipedia.org/wiki/List_of_most_visited_art_museums

[2] WikiArt displays both copyright protected and public domain art. The copyright protected art is displayed in accordance with the fair use principle: https://www.wikiart.org/en/about.
They display historically significant artworks and provide low resolution copies that are unsuitable for commercial use.
We do not distribute any of the art. We only provide URLs to the WikiArt.org pages, along with the crowdsourced annotations for these pieces of art.

annotated within paintings of a particular style. Examination of the image only, title only, and whole art (image and title together) annotations revealed that the title of the art markedly impacted the emotion evoked by a painting.

We show that paintings with faces (and to a lesser extent, paintings depicting a body but no face) elicited markedly more consistent emotion annotations than paintings that did not depict a face or body.

About 64% of all the annotated art was marked as liked (to some degree), 18% as disliked (to some degree), and 18% as neither liked nor disliked. We found that paintings evoking positive emotions were liked more, in general. We also found that paintings that bring to mind certain positive emotions such as love were liked much more than paintings that bring to mind other positive emotions such as humility. The difference was even more pronounced when comparing negative-emotion paintings; paintings bringing to mind regret, arrogance, and sadness were liked much more than paintings bringing to mind disgust, anger, or fear. Paintings evoking no emotion were the least liked paintings.

The WikiArt Emotions Dataset is made freely available for research on emotions in art as well as for developing automatic systems that can detect emotions evoked by art.[3] We will also provide an interactive visualization that allows users to search for WikiArt paintings with desired attributes such as style, genre, emotion, and average art ratings.

2. Related Work

Automatically understanding the content of images and text is beneficial for several information extraction and information retrieval needs. Advances in vision and natural language processing have greatly improved the capabilities of automatic systems for understanding real-world images and text. Most of these automatic systems rely on supervised machine learning algorithms which require large amounts of human-labeled instances. Several computer vision datasets have been developed and made available. (See Appendix.) Resources such as ImageNet (Deng et al., 2009) and Microsoft Common Objects in Context (MS COCO) (Lin et al., 2014) are of particular interest for developing algorithms at the intersection of computer vision and natural language processing. ImageNet has thousands of images each for many WordNet noun concepts, whereas MS COCO has hundreds of thousands of images, many of which have English captions (descriptions).

Automatically detecting emotions has also gained considerable attention over recent years, especially from text (Mohammad, 2012b; Mohammad, 2012a; Zhu et al., 2014; Kiritchenko et al., 2014; Yang et al., 2007; Bollen et al., 2009; Wang et al., 2016; Mohammad and Bravo-Marquez, 2017; Mohammad et al., 2018) but also from images (Fasel and Luettin, 2003; De Silva et al., 1997; Zheng et al., 2010). However, image annotations for emotions have largely been limited to small datasets of facial expressions (Lucey et al., 2010; Susskind et al., 2007). Ours is the first dataset we know of that includes emotions annotations for thousands of pieces of art.

[3]WikiArt Emotions Project webpage: http://saifmohammad.com/WebPages/wikiartemotions.html

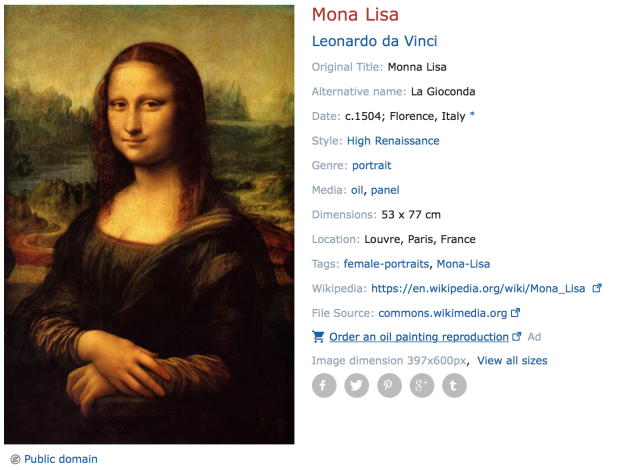


Figure 1: WikiArt.org's page for the *Mona Lisa*. In the WikiArt Emotions Dataset, the *Mona Lisa* is labeled as evoking happiness, love, and trust; its average rating is 2.1 (in the range of -3 to 3).

3. WikiArt Emotions Dataset

We now describe how we created the WikiArt Emotions Dataset.

3.1. Compiling the Art

As of January 2018, WikiArt.org had 151,151 pieces of art (mostly paintings) corresponding to ten main art styles and 168 style categories. See Table 1 for details. The art is also independently classified into 54 genres. Portrait, landscape, genre painting, abstract and religious painting are the genres with the most items. The art can be in one of 183 different media. Oil, canvas, paper, watercolor, and panel are the most common media. For each piece of art, the website provides the title, the image, the name of the artist, the year in which the piece of art was created, the style, the genre, and the medium. Figure 1 shows the page WikiArt.org provides for the Mona Lisa. We collected the URLs and the meta-information for all of these pieces of art and store them in a simple, easy to process file format. The data is made freely available via our WikiArt Emotions project webpage for non-commercial research and art- or education-related purposes.[4]

For our human annotation work, we chose about 200 paintings each from twenty-two categories (4,105 paintings in total). The categories chosen were the most populous ones (categories with more than 1000 paintings) from four styles: Modern Art, Post-Renaissance Art, Renaissance Art, and Contemporary Art.[5] For each chosen category, we selected up to 200 paintings displayed on WikiArt.org's 'Featured' tab for that category. (WikiArt selects certain paintings from each category to feature more prominently on its website. These are particularly significant pieces of art.) Table 2 summarizes these details.

[4]http://saifmohammad.com/WebPages/wikiartemotions.html

[5]Art Nouveau (Modern), Symbolism, Naive Art (Primitivism), Conceptual Art, Mannerism (Late Renaissance), and Academicism also each had more than 1000 paintings, but we chose not to annotate paintings of these styles for now.

Style	#Categories	#Items
Modern Art	90	90576
Post-Renaissance Art	13	39497
Contemporary Art	33	8906
Renaissance Art	6	7511
Japanese Art	9	2909
Chinese Art	2	746
Medieval Art	7	636
Islamic Art	6	306
Native Art	1	50
Korean Art	1	14
Total	**168**	**151,151**

Table 1: The number of items in each art style on WikiArt.org. The styles are listed in decreasing order of the number of items.

		# Items	
Style	**Style Category**	**Total**	**Annotated**
Contemporary Art			
	Minimalism	2001	200
Modern Art			
	Impressionism	14862	200
	Expressionism	10629	200
	Post-Impressionism	7405	200
	Surrealism	6813	198
	Abstract Expressionism	4367	200
	Cubism	2963	200
	Pop Art	2004	200
	Abstract Art	1812	200
	Art Informel	1807	200
	Color Field Painting	1585	200
	Neo-Expressionism	1304	200
	Magic Realism	1289	153
	Lyrical Abstraction	1124	200
Post-Renaissance Art			
	Realism	13972	200
	Romanticism	10929	200
	Baroque	5498	200
	Neoclassicism	3450	197
	Rococo	2868	200
Renaissance Art			
	Northern Renaissance	2867	192
	High Renaissance	1465	104
	Early Renaissance	1405	119
Total		**151,151**	**4,105**

Table 2: The styles and categories whose items are annotated for emotions in the WikiArt Emotions Project. The total number of items in each category as well as the number of items chosen for annotations are also shown. Some items belong to more than one category. The styles are shown in reverse chronological order. The categories are shown in decreasing order of the number of total items.

3.2. Designing the Questionnaire

Art can be annotated for emotions in several different ways. We describe below some of the choices we made, and the motivations behind them.

What Emotion Question to Ask: Just as text, one can label art for emotions from many perspectives: what emotion is the painter trying to convey, what emotion is felt by the observer (the person viewing the painting), what emotion is felt by the entities depicted in the painting, how does the observer feel towards entities depicted in the painting, etc. All of these are worthy annotations to pursue. However, in this work we focus on the emotions evoked in the observer (the annotator) by the painting. That decided, it is still worth explicitly articulating what it means for a painting to evoke an emotion, as here too, many different interpretations exist. Should one label a painting with sadness if it depicts an entity in an unhappy situation, but the observer does not feel sadness on seeing the painting? How should one label an art depicting and evoking many different emotions, for example, a scene of an angry mother elephant defending her calf from a predator? And so on. For this annotation project we chose to instruct annotators to label all emotions that the painting brings to mind. Our exact instructions in this regard are shown below:

> *Art is imaginative human creation meant to be appreciated and evoke an emotional response. We will show you pieces of art, mostly paintings, one at a time. Your task is to identify the emotions that the art evokes, that is, all emotions that the art brings to mind. For example:*
>
> - *the image of someone suffering, brings to mind sadness.*
> - *the image of a mother elephant fighting a lion to protect its calf, may bring to mind the mother's fear of losing the calf, anger at the lion, and admiration of the mother's bravery.*
> - *the image of a rich tyrant enjoying his feast may bring to mind both his conceit and your disgust for him.*
> - *the symmetry of lines and shapes in non-representative art may bring a sense of calm, whereas looking at the juxtaposition of shapes in a different art may evoke a sense of conflict.*

Which Emotions Apply Frequently to Art: Humans are capable of recognizing hundreds of emotions and it is likely that all of them can be evoked from paintings. However, some emotions are more frequent than others and come more easily to mind. Further, different individual experiences may prime different people to easily recall different sets of emotions. Also, emotion boundaries are fuzzy and some emotion pairs are more similar than others. All of this means that an open-ended question asking annotators to enter the emotions evoked through a text box is sub-optimal. Thus we chose to provide a set of options (each corresponding to a closely related emotions set) and asked annotators to check all emotions that apply. We chose the options from these sources:

- The psychology literature on basic emotions (Ekman, 1992; Plutchik, 1980; Parrot, 2001).
- The psychology literature on emotions elicited by art (Silvia, 2005; Silvia, 2009; Millis, 2001; Noy and Noy-Sharav, 2013).
- Our own annotations of the WikiArt paintings in a small pilot effort.

Polarity	Emotion Category	Abbreviation
Positive	**gratitude**, thankfulness, or indebtedness	grat
	happiness, calmness, pleasure, or ecstasy	happ
	humility, modesty, unpretentiousness, or simplicity	humi
	love or affection	love
	optimism, hopefulness, or confidence	opti
	trust, admiration, respect, dignity, or honor	trus
Negative	**anger**, annoyance, or rage	ange
	arrogance, vanity, hubris, or conceit	arro
	disgust, dislike, indifference, or hate	disg
	fear, anxiety, vulnerability, or terror	fear
	pessimism, cynicism, or lack of confidence	pessi
	regret, guilt, or remorse	regr
	sadness, pensiveness, loneliness, or grief	sadn
	shame, humiliation, or disgrace	sham
Other or Mixed	**agreeableness**, acceptance, submission, or compliance	agre
	anticipation, interest, curiosity, suspicion, or vigilance	anti
	disagreeableness, defiance, conflict, or strife	disa
	surprise, surrealism, amazement, or confusion	surp
	shyness, self-consciousness, reserve, or reticence	shyn
	neutral	neut

Table 3: The list of emotions provided to annotators to label the image, the title text, and the art (title and image).

We grouped similar emotions into a single option. The final result was 19 options of closely-related emotion sets and a final neutral option. The options were arranged in three sets 'positive', 'negative', and 'mixed or other' as shown in Table 3, to facilitate ease of annotation. A text box was also provided for the annotators to capture any additional emotions that were not part of the pre-defined set of the 19 options. Many extra emotions were entered by the annotators, including uncertainty, amusement, and jealousy. However, none of the proposed additional emotions was used more than 20 times overall, which indicates that the pre-defined set of the 19 emotions we provided covered the art emotion space well.

Emotions evoked by the image alone, the title alone, and the art as a whole: We asked annotators to identify the emotions that the art evokes in three scenarios:

- Scenario I: we present only the image (no title), and ask the annotator to identify the emotions it evokes;
- Scenario II: we present only the title of the art (no image), and ask the annotator to identify the emotions it evokes;
- Scenario III: we present both the title and the image of the art, and ask the annotator to identify the emotions that the art as a whole evokes.

We instruct the annotators so that:

- when answering the question about the title (scenario II), they should not try to recollect what they answered earlier for the image that goes with it. Their response should be based solely on the title.
- When answering the question about the title–image combination (scenario III), they should not try to recollect what they answered earlier for the image alone or for the title alone. Their response should be based on what the art evokes.

To help the annotators focus only on the question at hand (and exclude influences from earlier responses), we show five instances in scenario I in a random order, followed by five instances in scenario II in a different random order, followed by five in scenario III in another random order.

Questions Asked: Secenario I question is shown below:

> *Q1. Examine the art above (the image). Which of the following describe the emotions it brings to mind? Select all that apply.*
> (Twenty options as shown in Table 3.)

The Questions for scenario II and III (Q2 and Q3), looked identical to Q1, except that they asked the annotator to examine the title and the art (image and title), respectively.

In Scenario III (image and title) we asked the following additional questions.

> *Q4. Which of the following best describes how you feel about the piece of art?*
> 3: *I like it a lot.*
> 2: *I like it.*
> 1: *I like it somewhat.*
> 0: *I neither like it nor dislike it.*
> −1: *I dislike it somewhat.*
> −2: *I dislike it.*
> −3: *I dislike it a lot.*
>
> *Q5. Which of the following is true about the image? Click all that apply.*
> *– the image shows the face of at least one person or animal (select if there is any indication of a face anywhere in the image)*
> *– the image shows the body of at least one person or animal (select if there is any indication of a body anywhere in the image)*
> *– none of the above*

Example instances were provided in advance with examples of suitable responses.

3.3. Crowdsourcing Annotations

We annotated all of our data by crowdsourcing. Links to the art and the annotation questionnaires were uploaded on the crowdsourcing platform, CrowdFlower.[6] All annotators for our tasks had already agreed to the CrowdFlower terms of agreement. They chose to do our task among the hundreds available, based on interest and compensation provided. Respondents were free to annotate as many instances as they wished to. The annotation task was approved by the National Research Council Canada's Institutional Review Board, which reviewed the proposed methods to ensure that they were ethical. Special attention was paid to obtaining informed consent and protecting participant anonymity.

About 2% of the instances were annotated internally beforehand (by the authors). These instances are referred to as gold instances. The gold instances are interspersed with other instances. If a crowd-worker answers a gold instance question incorrectly, they are immediately notified of the error. If the worker's accuracy on the gold instances falls below 70%, they are refused further annotation, and all of their annotations are discarded. This serves as a mechanism to avoid malicious annotations. We mainly used the face–body question (Q5) as gold, but we also used, although sparingly, the emotion questions (Q1–Q3) as gold. Even though the emotional response to art is somewhat subjective, there are instances where some emotions clearly apply.

On the CrowdFlower task settings, we specified that we needed annotations from ten people for each instance. However, because of the way the gold instances are setup, they are annotated by more than ten people. The median number of annotations is still ten. In all, 308 people annotated between 20 and 1,525 pieces of art. A total of 41,985 sets of responses (for Q1–Q5) were obtained for the 4,105 pieces of art.

Annotation Aggregation and Machine Learning Datasets: For each item (image, title, or art), we will refer to the emotion that receives the majority of the votes from the annotators as the *predominant emotion*. In case of ties, all emotions with the majority vote are considered the predominant emotions. When aggregating the responses to obtain the full set of emotion labels for an item, we wanted to include not just the predominant emotion, but all others that apply, even if their presence is more subtle. Thus, we chose a somewhat generous aggregation criteria: if at least 40% of the responses (four out of ten people) indicate that a certain emotion applies, then that label is chosen. We will refer to this as *Ag4* dataset. 929 images, 1332 titles, and 823 paintings did not receive sufficient votes to be labeled with any emotion. These items were set aside. The rest of the items and their emotion labels can be used to train and test machine learning algorithms to predict the emotions evoked by art.

We also created two other versions of the labeled dataset by using an aggregation threshold of 30% and 50%, respectively. (If at least 30%/50% of the responses (three/five out of ten people) indicate that a certain emotion applies, then that label is chosen.) We will refer to them as *Ag3* and *Ag5* datasets. In our own future work, we will be working mainly with the Ag4 version of the data, however, the other versions will also be made available for those interested in those variants.

[6] http://www.crowdflower.com

Class Distribution: The '% votes' rows of Table 4 show the percentage of times each emotion was selected by the annotators. The 'Ag3', 'Ag4', and 'Ag5' rows show the distribution of labels in the WikiArt Emotions dataset after aggregation. The numbers in each of the rows sum up to more than 100% because an item may be labeled with more than one emotion. Observe that anticipation, fear, happiness, humility, love, optimism, sadness, surprise, and trust get a high number of votes, whereas the rest get only a small percentage of votes. Observe also that as the aggregation threshold is increased (Ag3 through Ag5), the percentage of tweets labeled with the less-frequent emotions reduces. For example, even though anger received 3.5% of the total art votes and 2% of the art pieces were labeled with anger when using Ag3, only 1% of tweets have anger as a label when using Ag5. Table 5 shows the percentage of times each emotion got the majority of votes, and was thus selected as the predominant emotion. Observe that some emotions such as humility and trust have markedly lower percentages as the predominant emotion than as one of the applicable emotions.

Tables 9 in the Appendix shows the proportions of the items in the WikiArt Emotions dataset corresponding to art with no face or body depicted, art with a body depicted but no face, and art with a face depicted broken down by art category. One can see that the vast majority of the Renaissance and Post-Renaissance art depict faces, with the lowest proportion corresponding to Romanticism (0.71) and the highest proportion corresponding to the Early and High Renaissance (1.00). Minimalism, Abstract Art, and Color Field Paintings have the lowest depiction rates of face or body (0 to 0.01). Table 10 in the Appendix shows these proportions broken down by emotion. We observe that certain emotions such as arrogance, shame, love, gratitude, and trust are depicted predominantly through faces (face-present proportions greater than 0.7). In contrast, emotions of anticipation, surprise, and disgust are predominantly evoked by paintings without any depictions of face or body (face-present proportions less than 0.4). The percentages for neutral indicate that a vast majority of the paintings that did not evoke any emotion did not depict a face or a body (neither face nor body proportion of 0.88).

4. Agreement

Emotion annotations of art are not expected to be highly consistent across people for a number of reasons, including: differences in human experience that impact how they perceive art, the subtle ways in which art can express affect, and fuzzy boundaries of affect categories. With the annotations on the WikiArts Emotion dataset, we can now determine the extent to which this agreement exists across different emotions, and how the agreements are impacted by attributes of the painting such as style, style category, and depictions of faces.

	agre	ange	anti	arro	disa	disg	fear	grat	happ	humi	love	opti	pess	regr	sadn	sham	shyn	surp	trus	neut
Image (no Title)																				
% votes	3.1	3.4	**19.3**	4.6	3.6	7.0	**11.0**	5.0	**24.3**	**14.0**	8.1	**11.8**	4.6	2.9	**10.2**	2.6	1.9	**21.2**	**17.1**	1.2
Ag3: % items label.	0.4	2.0	**26.4**	3.4	0.6	4.0	**14.1**	1.9	**38.0**	**17.3**	9.1	**10.0**	1.8	0.4	**11.6**	1.1	0.2	**35.4**	**23.2**	0.2
Ag4: % items label.	0.1	1.2	**15.4**	1.7	0.2	1.0	9.9	0.8	**35.2**	**10.8**	7.3	3.4	0.6	0.2	8.0	0.4	0.1	**29.3**	**17.7**	0.0
Ag5: % items label.	0.0	1.0	9.1	1.0	0.1	0.3	9.2	0.4	**35.1**	6.7	7.3	1.3	0.3	0.0	7.2	0.2	0.0	**24.2**	**16.2**	0.0
Title (no Image)																				
% votes	3.0	2.0	**27.0**	3.3	2.6	5.6	6.1	4.9	**23.0**	**11.2**	7.7	**11.7**	2.8	2.1	6.0	1.7	1.8	**12.1**	**17.3**	5.9
Ag3: % items label.	0.2	1.3	**48.9**	1.1	0.5	1.8	6.5	2.1	**36.9**	9.9	7.4	9.8	0.7	0.3	6.1	0.7	0.1	**10.0**	**23.4**	4.8
Ag4: % items label.	0.0	0.8	**37.5**	0.6	0.1	0.4	5.3	0.7	**33.3**	4.5	6.5	3.8	0.3	0.0	5.4	0.4	0.0	4.4	**19.2**	2.7
Ag5: % items label.	0.0	0.8	**28.7**	0.3	0.0	0.3	5.2	0.3	**35.3**	2.6	6.4	1.5	0.2	0.0	5.4	0.1	0.0	2.5	**19.6**	1.9
Art (Image and Title)																				
% votes	3.2	3.5	**18.9**	4.9	3.6	7.6	**10.9**	5.6	**26.3**	**15.2**	9.1	**13.9**	5.1	3.2	**11.0**	2.8	2.1	**21.0**	**19.8**	1.2
Ag3: % items label.	0.3	2.0	**25.5**	3.3	0.7	5.2	**13.9**	3.0	**41.3**	**19.7**	**10.1**	**14.5**	2.6	0.8	**13.0**	1.5	0.1	**34.6**	**27.6**	0.2
Ag4: % items label.	0.1	1.3	**15.4**	1.9	0.2	1.7	**10.2**	1.3	**36.9**	**12.0**	8.1	5.9	1.1	0.3	9.2	0.7	0.1	**27.4**	**21.5**	0.0
Ag5: % items label.	0.0	1.0	9.8	1.0	0.1	0.7	8.8	0.7	**36.5**	8.2	7.7	2.7	0.6	0.0	7.9	0.4	0.0	**21.6**	**20.4**	0.0

Table 4: **Applicable Emotion:** Percentage of votes for each emotion as being applicable and the percentage of items that were labeled with a given emotion (after aggregation of votes). Numbers greater than or equal to 10% are shown in bold.

	agre	ange	anti	arro	disa	disg	fear	grat	happ	humi	love	opti	pess	regr	sadn	sham	shyn	surp	trus	neut
Image	0.0	0.7	**10.3**	1.0	0.2	1.5	**8.5**	0.1	**23.9**	**5.5**	3.2	1.3	0.4	0.0	**6.1**	0.2	0.1	**25.0**	**11.9**	0.2
Title	0.1	0.4	**34.9**	0.4	0.2	1.1	4.2	0.2	**23.8**	3.1	3.1	2.3	0.2	0.0	3.8	0.2	0.0	4.4	**14.5**	3.2
Art	0.0	0.6	**9.2**	1.0	0.1	2.0	**8.0**	0.2	**24.9**	**5.0**	3.1	1.9	0.4	0.0	**6.3**	0.3	0.1	**23.5**	**13.3**	0.2

Table 5: **Predominant Emotion:** Percentage of items that were predominantly labeled with a given emotion. Numbers greater than or equal to 5% are shown in bold.

Figure 4 in the Appendix shows the Fleiss' κ inter-rater agreement scores for each of the emotion classes across the different style categories. (Fleiss' κ calculates the extent to which the observed agreement exceeds the one that would be expected by chance (Fleiss, 1971). However, note that correcting for chance remains controversial.[7] Nonetheless, the relative variations in the values of κ are useful indicators of relative agreement.)

Observe that the κ scores range from close to 0 to 0.31 for the different emotion–category combinations. The close to 0 scores indicate that when considering all the paintings for some category–emotion pairs there is very little agreement beyond random chance. Nonetheless, there exist subsets of paintings, even for those category–emotion pairs, where agreement is higher. Scores closer to 0.31 indicate fair amounts of agreement for the sets as a whole. It should be noted that in general, agreement scores for art are lower than what one finds for text, which is expected. (Mohammad and Kiritchenko (2018) report Fleiss' κ scores in the range of 0.32 to 0.47 for anger, fear, joy, and sadness conveyed by tweets, and lower scores for other emotions such as surprise, trust, and optimism.)

The κ scores are relatively higher for the Renaissance and Post-Renaissance art styles as compared to Modern Art and Contemporary Art. This is likely because, on average, Modern Art tends to be more abstract and non-representative. The κ scores are relatively high for basic emotions such as fear, happiness, sadness, anger, and love, and lower for more complex emotions such as optimism, shame, guilt, and regret. Nonetheless, for certain category–emotion pairs the agreement is relatively high as compared to other categories and the same emotion. Examples include: Post-Renaissance categories with trust (especially, Romanticism–trust), Early Renaissance and Northern Renaissance with shame, Magic Realism with arrogance, Magic Realism with shame, Surrealism with surprise, Post-Impressionism with arrogance, and Early Renaissance with arrogance.

Figure 2 shows the agreement on the three partitions of the paintings corresponding to art with no face or body depicted, art with a body depicted but no face, and art with a face depicted. Observe that agreements are markedly higher for art with a body than with no body, and markedly higher again for art that depicts a face than art that only shows a body but no face. This is consistent with the hypothesis that human faces are an effective medium to convey emotions, and that depiction of even just a body without face is effective in conveying emotions and therefore elicits similar emotions in different observers.

Note that even though the κ scores shown here are lower than what one might find for other tasks such as part-of-speech tagging or named-entity recognition, these scores are closer to what one finds when annotating text for emotions (as indicated earlier). Further, the aggregation strategies of Ag4 and Ag5 described in the previous section, help filter out items with low inter-annotator agreement, and the remaining items can be used to train and test machine learning systems that detect emotions evoked by art.

[7] http://www.john-uebersax.com/stat/kappa2.htm
http://www.agreestat.com/book3/bookexcerpts/chapter2.pdf

Group	Emotion	Neither Face Nor Body Present	Body Present, No Face	Face Present
Positive	gratitude	0.0100	0.0000	0.0400
	happiness	0.1900	0.2000	0.2200
	humility	0.0600	0.0600	0.0800
	love	0.0300	0.1000	0.2500
	optimism	0.0300	0.0400	0.0300
	trust	0.0200	0.0600	0.1500
Negative	anger	0.0200	0.0500	0.1400
	arrogance	-0.0100	0.0200	0.0800
	disgust	0.0200	0.0700	0.0700
	fear	0.1100	0.1900	0.2400
	pessimism	0.0300	0.0600	0.0600
	regret	-0.0100	0.0200	0.0400
	sadness	0.0700	0.1100	0.2300
	shame	0.0100	0.1000	0.0800
Other	agreeableness	0.0000	0.0100	0.0000
	anticipation	0.0200	0.0300	0.0100
	disagreeableness	0.0000	0.0100	0.0300
	neutral	0.0100	0.0000	0.0000
	shyness	0.0000	0.0200	0.0100
	surprise	0.0600	0.1100	0.0700

Figure 2: Annotator agreement (Fleiss' κ) for art pieces that show the face of a person or an animal (2,068 items), art pieces that show the body (and no face) of a person or an animal (227 items), and art pieces that show neither a face nor a body (1,810 items).

5. Emotions that Tend to Occur Together

Since we allow annotators to mark multiple emotions as being associated with an item, it is worth examining which emotions tend to be frequently voted for together (often evoked together by art). For every pair of emotions, i and j, we calculated the proportion of times an item received votes for both emotions i and j from an observer, out of all the votes for emotion i across all items. (See Figure 5 in the Appendix for the co-occurrence proportions.) The following pairs of emotions have scores greater than 0.4 indicating that when the first emotion is present, there is a greater than 40% chance that the second is also present. Emotion pairs of this kind include: gratitude–trust, love–happiness, pessimism–sadness, regret–sadness, shame–sadness, and surprise–anticipation. It is interesting to note that pessimism, regret, and shame have high co-occurrence with both fear and sadness. This suggests that these are complex emotions that include some elements of fear and sadness (two basic emotions) within them. Note also that for many emotion pairs, the association is markedly stronger in one direction than in the other direction. For example, pessimism is often indicative of sadness, but sadness is not often indicative of pessimism. As expected, highly contrasting emotions such as happiness and disgust have very low co-occurrence scores.

6. Emotions Evoked from the Image, the Title, and the Art

Titles of paintings impact how the observer views the art. They guide the observer by highlighting some aspect of the art.[8] With our annotations, we wanted to quantify the impact titles have on the emotional response elicited by the art. Thus, as indicated earlier, we asked annotators to provide the emotions evoked by the image alone, the title alone, and the art as whole (image and title). From these annotations, we calculated: a. the percentage of times a piece of art (image and title) was annotated with the same set of emotions as just the image; b. the percentage of times a piece of art (image and title) was annotated with the same set of emotions as just the title; and c. the percentage of times the image was annotated with the same set of emotions as just the title. Here, two sets of emotion labels are considered different if any one of the emotions in one set is not present in the other set. Table 6 shows the results.

[8]Titles are of different types such as sentimental, factual, abstract, and mysterious.

	% Match
a. image–art	53.32
b. title–art	30.80
c. image–title	27.20

Table 6: The percentage of annotations that have exactly the same emotion sets selected for image and art, title and art, or image and title by the same annotator.

Observe that the title often conveys a different set of emotions than the image alone or the art as a whole. In contrast, the art and image often convey the same sets of emotions, but there is a large percentage of instances where they differ. This shows that the title of an art plays a substantial role in the emotions evoked by the art.

7. What Makes Art Well Liked?

Art is judged in many ways: by how engaging, thought-provoking, or evocative it is, by the amount of expertise needed to create the art, by how easy it is to understand what is being communicated, by how pleasing the shapes and colours are, etc. Further, one may find the painting very engaging, but not want it in their home. Rather than asking people to judge all of these facets, we asked our annotators to simply rate the extent to which they liked or disliked the painting overall (Question 5). Table 7 shows the distribution of annotations that the art pieces received. Observe that the majority of the art is well liked, with the ratings of 2 (like it) and 1 (somewhat like it) being the most common. Around 18% of the pieces are marked as disliked (to varying degrees) and another 18% of the pieces are marked as being neither liked nor disliked.

Table 8 gives the average art ratings for the different art styles. Table 11 in the Appendix gives a breakdown of the art ratings by style category. Observe that Post-Renaissance and Renaissance pieces are liked the most (especially Realism, Rococo, Neoclassicism, and High Renaissance). Even though Modern Art overall received lower average rating score, Impressionism is the most liked category among all twenty-two considered in this work. Minimalism (Contemporary Art) and Art Informel (Modern Art) received the lowest ratings.

Table 12 in the Appendix gives a breakdown of the art ratings by emotion. We observe that art which evokes no emotion (neutral) and art which evokes disgust receive the lowest average scores. In contrast, art that evokes positive emotions such as love, gratitude, happiness, humility, optimism, and trust obtain some of the highest average scores. It is interesting to note that pieces of art that evoke negative

Like Description	Rating	% Annotations
like it a lot	3	17.41
like it	2	24.20
like it somewhat	1	22.01
neither like it nor dislike it	0	17.93
dislike it somewhat	-1	8.12
dislike it	-2	6.31
dislike it a lot	-3	4.02

Table 7: Distribution of art ratings.

Art Category	Ave. Rating
Contemporary Art	-0.07
Modern Art	0.67
Post-Renaissance Art	1.52
Renaissance Art	1.29
Average	**0.91**

Table 8: Average art ratings per art category.

emotions such as sadness, arrogance, and regret received markedly higher average scores than surprise and other negative emotions such as pessimism, shame, fear, and anger. Figure 3 in the Appendix shows the full breakdown of average art ratings for each category–emotion pair. Romanticism, Neoclassicism, and Impressionism paintings evoking love as well as Impressionism paintings evoking optimism received the highest scores (2.15–2.20).

8. Future Work and Applications

This paper examines attributes of a painting such as its style and content (face, body, none) and the emotions it evokes. We are currently analyzing the role of the features of the observer such as gender, age, and personality, on the emotions they perceive in art. We are also interested in conducting further annotations amongst the sets of paintings that evoke happiness, love, fear, and sadness, to determine the *intensities* of emotions they evoke. This will allow for a ranking of paintings by joy intensity, fear intensity, etc. We also want to determine whether paintings that evoke intense amounts of an emotion are also the ones that are, on average, liked more. We will also annotate the paintings that depict faces and bodies to determine whether the left or right side of the face or body is shown more prominently in the art. These annotations will help test the hypothesis that art that depicts the left side of a person's face or body is on average found to be more appealing (left-cheek bias) (Powell and Schirillo, 2011; Blackburn and Schirillo, 2012).

The WikiArt Emotions dataset has many applications in automatic image and text processing, including those listed below:

- To train and test machine learning algorithms that can predict the emotions evoked by art. It will be interesting to determine the accuracies of unimodal (image- or text-only based) systems as well as multi-modal (text- and image-based) systems to detect the emotions. We will conduct experiments to determine the extent to which different modalities (text and image) are useful in detecting emotion intensity, and under what circumstances they provide complementary information.
- To conduct experiments to determine what characteristics of images make them particularly evocative.
- To develop deep learning algorithms for art generation; for instance, to create systems that can transform a given piece of art (especially abstract paintings) to alter the affective reaction it evokes (for example, transforming a painting to make it evoke more sadness or more conflict).

We are currently developing an interactive visualization that allows users to search for WikiArt.org paintings with desired attributes such as style, genre, emotion, and average art ratings.

9. Conclusions

We created the *WikiArt Emotions Dataset*, which includes emotion annotations for more than 4,000 pieces of art from four western styles (Modern Art, Post-Renaissance Art, Renaissance Art, and Contemporary Art) and 22 style categories. The art is annotated for one or more of twenty emotion categories (including neutral). We also obtained separate emotion annotations for when the observer sees only the image and sees only the title of the art. We found that fear, happiness, love, and sadness were the dominant emotions that also obtained consistent annotations among the different annotators. Other emotions were also found to be more frequent and consistently annotated within paintings of particular style categories. Examination of the image only, title only, and whole art annotations revealed that the title of the art markedly impacted the emotions evoked by a painting.

The WikiArt Emotions dataset also has annotations for whether the painting includes the depiction of a face, a body, or neither. We found that paintings with faces (and to a lesser extent, paintings depicting a body but no face) elicited markedly more consistent emotion annotations. Finally, the dataset includes ratings given by people corresponding to the extent to which they liked or disliked the art. About 64% of the art was marked as liked (to some degree), 18% as disliked (to some degree), and 18% as neither liked nor disliked. We found that paintings evoking positive emotions were liked more, in general. We also found that paintings evoking certain positive emotions such as love were liked much more than paintings evoking other positive emotions such as humility. The difference was even more pronounced when comparing paintings evoking negative emotions; paintings evoking regret, arrogance, and sadness were liked much more than paintings evoking disgust, anger, or fear. Paintings evoking no emotion and disgust, were some of the least liked paintings.

The WikiArt Emotions Dataset is made freely available for educational purposes and to facilitate research in emotions, art, human psychology, and automatic image analysis/generation.

10. Acknowledgements

We thank Peter Turney for suggesting that we examine emotions evoked by art.

11. Bibliographical References

Aiken, N. E. (1998). *The biological origins of art.* Praeger Publishers/Greenwood Publishing Group.

Blackburn, K. and Schirillo, J. (2012). Emotive hemispheric differences measured in real-life portraits using pupil diameter and subjective aesthetic preferences. *Experimental brain research*, 219(4):447–455.

Bollen, J., Mao, H., and Pepe, A. (2009). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Proceedings of the Fifth International Conference on Weblogs and Social Media*, pages 450–453.

Davies, S. (2012). *The artful species: aesthetics, art, and evolution*. OUP Oxford.

De Silva, L. C., Miyasato, T., and Nakatsu, R. (1997). Facial emotion recognition using multi-modal information. In *Proceedings of the International Conference on Information, Communications and Signal Processing*, volume 1, pages 397–401.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Dutton, D. (2009). *The art instinct: Beauty, pleasure, & human evolution*. Oxford University Press, USA.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3):169–200.

Fasel, B. and Luettin, J. (2003). Automatic facial expression analysis: a survey. *Pattern recognition*, 36(1):259–275.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Kiritchenko, S., Zhu, X., and Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., and Matthews, I. (2010). The extended Cohn–Kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 94–101.

Miller, G. F. (2001). Aesthetic fitness: How sexual selection shaped artistic virtuosity as a fitness indicator and aesthetic preferences as mate choice criteria. *Bulletin of Psychology and the Arts*, 2(1):20–25.

Millis, K. (2001). Making meaning brings pleasure: The influence of titles on aesthetic experiences. *Emotion*, 1(3):320.

Mohammad, S. M. and Bravo-Marquez, F. (2017). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mohammad, S. M. and Kiritchenko, S. (2018). Understanding emotions: A dataset of tweets to study interactions between affect categories. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

Mohammad, S. M., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Mohammad, S. (2012a). #Emotional Tweets. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 246–255, Montréal, Canada.

Mohammad, S. M. (2012b). From once upon a time to happily ever after: Tracking emotions in mail and books. *Decision Support Systems*, 53(4):730–741.

Noy, P. and Noy-Sharav, D. (2013). Art and emotions. *International Journal of Applied Psychoanalytic Studies*, 10(2):100–107.

Parrot, W. (2001). *Emotions in Social Psychology*. Psychology Press.

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. *Emotion: Theory, research, and experience*, 1(3):3–33.

Powell, W. R. and Schirillo, J. A. (2011). Hemispheric laterality measured in Rembrandt's portraits using pupil diameter and aesthetic verbal judgements. *Cognition & emotion*, 25(5):868–885.

Silvia, P. J. (2005). Emotional responses to art: From collation and arousal to cognition and emotion. *Review of general psychology*, 9(4):342.

Silvia, P. J. (2009). Looking past pleasure: anger, confusion, disgust, pride, surprise, and other unusual aesthetic emotions. *Psychology of Aesthetics, Creativity, and the Arts*, 3(1):48.

Susskind, J., Littlewort, G., Bartlett, M., Movellan, J., and Anderson, A. (2007). Human and computer recognition of facial expressions of emotion. *Neuropsychologia*, 45(1):152–162.

Wang, B., Liakata, M., Zubiaga, A., Procter, R., and Jensen, E. (2016). SMILE: Twitter emotion classification using domain adaptation. In *Proceedings of the CEUR Workshop*, volume 1619, pages 15–21.

Yang, C., Lin, K. H.-Y., and Chen, H.-H. (2007). Building emotion lexicon from weblog corpora. In *Proceedings of the 45th Annual Meeting of the ACL*, pages 133–136.

Zheng, W., Tang, H., Lin, Z., and Huang, T. S. (2010). Emotion recognition from arbitrary view facial images. In *Proceedings of the European Conference on Computer Vision*, pages 490–503.

Zhu, X., Guo, H., Mohammad, S., and Kiritchenko, S. (2014). An empirical study on the effect of negation words on sentiment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 304–313, Baltimore, Maryland, June.

Art Category	Neither face nor body	Body, no face	Face pres.
Contemporary Art			
Minimalism	0.99	0.00	0.01
Modern Art			
Abstract Art	0.97	0.01	0.01
Abstract Expressionism	0.92	0.04	0.04
Art Informel	0.95	0.01	0.04
Color Field Painting	0.99	0.01	0.01
Cubism	0.49	0.07	0.45
Expressionism	0.19	0.10	0.71
Impressionism	0.22	0.09	0.69
Lyrical Abstraction	1.00	0.00	0.01
Magic Realism	0.38	0.10	0.52
Neo-Expressionism	0.33	0.13	0.55
Pop Art	0.48	0.07	0.44
Post-Impressionism	0.38	0.08	0.54
Surrealism	0.49	0.10	0.41
Post-Renaissance Art			
Baroque	0.12	0.06	0.82
Neoclassicism	0.03	0.04	0.94
Realism	0.17	0.10	0.74
Rococo	0.03	0.06	0.92
Romanticism	0.19	0.11	0.71
Renaissance Art			
Early Renaissance	0.00	0.00	1.00
High Renaissance	0.00	0.00	1.00
Northern Renaissance	0.02	0.04	0.94

Table 9: Proportions of items in the WikiArt Emotions dataset corresponding to art with no face or body depicted, art with a body depicted but no face, and art with a face depicted **broken down by art category**.

Emotion	Neither face nor body	Body, no face	Face pres.
Positive			
gratitude	0.21	0.05	0.74
happiness	0.30	0.07	0.62
humility	0.29	0.07	0.64
love	0.17	0.04	0.80
optimism	0.38	0.06	0.56
trust	0.19	0.04	0.76
Negative			
anger	0.40	0.04	0.56
arrogance	0.20	0.03	0.77
disgust	0.57	0.04	0.38
fear	0.41	0.06	0.53
pessimism	0.44	0.06	0.50
regret	0.32	0.06	0.62
sadness	0.31	0.06	0.62
shame	0.22	0.08	0.71
Other or Mixed			
agreeableness	0.29	0.05	0.65
anticipation	0.60	0.05	0.35
disagreeableness	0.49	0.05	0.46
shyness	0.40	0.06	0.54
surprise	0.71	0.04	0.25
neutral	0.88	0.02	0.10

Table 10: Proportions of items in the WikiArt Emotions dataset corresponding to art with no face or body depicted, art with a body depicted but no face, and art with a face depicted **broken down by emotion**.

WikiArt Emotions Project homepage:
http://saifmohammad.com/WebPages/wikiartemotions.html

Appendix

Tables 9 and 10 show the proportions of items in the WikiArt Emotions dataset corresponding to art with no face or body depicted, art with a body depicted but no face, and art with a face depicted broken down by category and emotion, respectively.

Table 11 gives a breakdown of the art ratings by art category. Table 12 gives a breakdown of the art ratings by emotion. Figure 3 shows the full breakdown of average art ratings for each category–emotion pair.

Figure 4 shows the Fleiss' κ inter-rater agreement scores for each of the emotion classes across the different style categories.

Figure 5 shows, for every pair of emotions, i and j, the proportion of times an item received votes for both emotions *i* and *j* from an observer, out of all the votes for emotion *i*.

Some freely available computer vision datasets:
http://www.computervisiononline.com/datasets
http://www.cvpapers.com/datasets.html
http://riemenschneider.hayko.at/vision/dataset/
http://clickdamage.com/sourcecode/cv_datasets.php
http://cocodataset.org/
http://www.image-net.org

Art Category	Ave. Rating
Contemporary Art	
Minimalism	-0.07
Average	**-0.07**
Modern Art	
Abstract Art	0.29
Abstract Expressionism	0.20
Art Informel	0.04
Color Field Painting	0.22
Cubism	0.75
Expressionism	0.98
Impressionism	1.69
Lyrical Abstraction	0.47
Magic Realism	1.29
Neo-Expressionism	0.39
Pop Art	0.48
Post-Impressionism	1.43
Surrealism	0.45
Average	**0.67**
Post-Renaissance Art	
Baroque	1.39
Neoclassicism	1.56
Realism	1.58
Rococo	1.58
Romanticism	1.49
Average	**1.52**
Renaissance Art	
Early Renaissance	1.20
High Renaissance	1.50
Northern Renaissance	1.18
Average	**1.29**
Average (all categories)	**0.91**

Table 11: Average art ratings per art category.

Emotion	Ave. Rating
Positive	
gratitude	1.87
happiness	1.79
humility	1.62
love	1.95
optimism	1.72
trust	1.76
Average	**1.79**
Negative	
anger	0.41
arrogance	0.80
disgust	-0.38
fear	0.27
pessimism	0.39
regret	0.89
sadness	0.79
shame	0.48
Average	**0.46**
Other or Mixed	
agreeableness	1.60
anticipation	0.99
disagreeableness	0.60
shyness	1.10
surprise	0.49
neutral	-0.43
Average	**0.73**
Average (all emotions)	**0.94**

Table 12: Average art ratings per emotion.

Art Style	Art Category	Agree..	Anger	Antici..	Arroga..	Disagr..	Disgust	Fear	Gratitu..	Happi..	Humili..	Love	Optimi..	Pessi..	Regret	Sadne..	Shame	Shyne..	Surpri..	Trust	Neutral
Contemporary Art	Minimalism	1.19	-0.22	0.43	0.23	-0.24	-0.76	-0.18	1.50	1.65	0.48	1.30	1.07	-0.34	0.41	-0.15	-0.54	-0.13	-0.06	1.52	-0.66
Modern Art	Abstract Art	1.62	-0.22	0.61	0.52	0.15	-0.70	-0.23	1.74	1.65	1.14	1.91	1.33	-0.26	0.23	-0.03	0.02	0.81	0.28	1.54	-0.21
	Abstract Expressionism	1.46	0.21	0.64	0.75	0.04	-0.79	-0.22	1.88	1.55	0.92	1.70	1.36	0.05	0.85	0.33	0.08	0.71	0.28	1.62	-0.41
	Art Informel	1.70	-0.30	0.56	0.16	-0.08	-0.75	-0.34	1.84	1.55	0.84	1.50	1.34	-0.35	0.00	0.22	-0.40	0.40	0.13	1.46	-0.32
	Color Field Painting	1.19	-0.52	0.48	0.33	-0.34	-0.88	-0.67	1.44	1.53	0.77	1.51	1.27	-0.59	0.31	-0.13	-1.17	0.03	0.08	1.61	-0.47
	Cubism	1.86	0.34	0.93	0.55	0.52	-0.26	0.31	1.60	1.61	1.32	1.80	1.55	0.25	0.68	0.68	0.25	0.94	0.52	1.65	-0.23
	Expressionism	1.34	0.67	1.20	0.65	1.10	-0.14	0.31	1.84	1.61	1.56	1.81	1.71	0.69	1.03	0.70	0.40	1.00	0.82	1.67	0.00
	Impressionism	1.76	1.51	1.69	1.16	1.66	0.82	1.09	1.94	1.99	1.96	2.15	2.15	1.01	1.48	1.20	1.01	1.57	1.61	1.96	1.20
	Lyrical Abstraction	1.38	0.18	0.85	1.08	0.44	-0.76	-0.14	1.98	1.75	1.26	1.78	1.51	0.18	0.66	0.33	0.83	0.85	0.39	1.82	-0.44
	Magic Realism	1.92	0.92	1.44	1.20	1.05	-0.16	0.67	2.01	1.92	1.74	2.07	2.04	0.74	1.50	1.04	0.59	1.65	1.15	1.95	0.30
	Neo-Expressio..	1.54	-0.22	0.86	0.13	0.04	-0.70	-0.27	1.43	1.60	1.09	1.77	1.39	0.11	0.30	0.42	-0.01	0.86	0.45	1.59	-0.59
	Pop Art	1.34	0.19	0.79	0.32	0.52	-0.54	0.07	1.62	1.59	1.07	1.62	1.39	0.31	0.56	0.50	0.01	0.68	0.34	1.43	-0.82
	Post-Impressi..	1.56	1.04	1.40	0.93	1.54	0.59	0.53	1.94	1.84	1.71	2.01	1.90	0.78	1.14	1.09	0.87	1.49	1.19	1.82	-0.13
	Surrealism	1.54	-0.09	0.98	0.43	0.10	-0.78	-0.21	1.74	1.66	1.33	1.86	1.52	0.13	0.45	0.50	0.08	0.37	0.41	1.71	-0.48
Post Renaissance Art	Baroque	1.71	1.26	1.68	1.02	1.35	0.78	1.01	1.92	1.77	1.84	2.03	1.81	1.05	1.41	1.15	0.93	1.89	1.57	1.80	-1.00
	Neoclassicism	1.91	1.13	1.80	1.08	1.59	0.85	1.10	2.09	1.94	2.04	2.16	1.97	1.28	1.52	1.21	1.12	1.69	1.77	1.84	0.50
	Realism	1.77	0.64	1.67	1.14	1.53	0.63	0.96	2.09	1.93	1.88	2.12	2.03	0.94	1.34	1.28	1.16	1.79	1.59	1.92	0.88
	Rococo	1.79	0.87	1.60	1.01	1.14	0.52	0.84	2.02	1.96	1.90	2.10	1.99	0.99	1.44	1.17	1.06	1.88	1.52	1.82	0.00
	Romanticism	1.73	1.02	1.71	1.23	1.16	0.70	1.01	2.08	1.88	1.89	2.20	1.99	0.94	1.31	1.24	0.88	1.70	1.57	1.82	0.50
Renaissance Art	Early Renaissance	1.56	0.67	1.41	0.63	1.01	0.28	0.65	1.70	1.62	1.63	1.81	1.61	0.73	0.82	0.69	0.42	1.26	1.64	1.65	0.00
	High Renaissance	1.52	1.07	1.69	1.11	1.48	0.21	0.81	1.87	1.81	1.88	1.99	1.94	0.95	1.24	1.02	0.31	1.45	1.54	1.84	2.00
	Northern Renaissance	1.40	0.62	1.31	0.71	0.76	0.01	0.70	1.75	1.67	1.69	1.81	1.75	0.61	0.64	0.90	0.61	1.56	1.47	1.66	0.22

Figure 3: A breakdown of average art ratings for each category–emotion pair in the WikiArt Emotions dataset. The scores range from -3 (most disliked) to 3 (most liked). Positive scores are shown in shades of green; darker shades for scores closer to 3. Negative scores are shown in orange; darker shades for scores closer to -3.

Style	Category	Agree..	Anger	Antici..	Arroga..	Disagr..	Disgust	Fear	Gratitu..	Happi..	Humili..	Love	Optimi..	Pessi..	Regret	Sadne..	Shame	Shyne..	Surpri..	Trust	Neutral
Contemporary Art	Minimalism	-0.01	0.01	0.00	0.01	-0.01	-0.02	0.07	-0.01	0.02	0.00	0.08	0.01	0.01	-0.01	0.07	0.01	0.01	-0.01	0.01	-0.01
Modern Art	Abstract Art	0.01	0.04	0.01	-0.02	0.00	0.01	0.08	0.00	0.09	0.00	0.00	0.01	0.01	-0.01	0.02	0.04	-0.01	0.00	-0.01	0.00
	Abstract Expressionism	0.01	0.02	0.02	-0.02	-0.02	0.02	0.10	-0.01	0.06	0.03	0.03	0.04	0.02	0.00	0.04	0.00	-0.01	0.01	-0.01	0.00
	Art Informel	0.00	0.03	0.00	-0.01	0.00	-0.01	0.07	-0.01	0.07	-0.02	0.03	0.03	0.04	0.00	0.03	0.01	0.00	-0.01	-0.01	0.00
	Color Field Painting	0.01	0.02	-0.01	-0.01	0.00	0.00	0.03	0.02	0.07	0.00	0.00	0.02	0.04	-0.02	0.07	0.03	0.00	0.01	0.00	0.00
	Cubism	-0.01	0.03	0.02	0.06	0.01	0.01	0.11	0.00	0.11	0.05	0.11	0.01	0.04	0.03	0.09	0.07	0.01	0.07	0.05	0.02
	Expressionism	0.00	0.04	0.00	0.04	-0.01	0.04	0.20	0.03	0.18	0.06	0.18	0.03	0.10	0.07	0.22	0.06	0.02	0.01	0.04	0.00
	Impressionism	0.00	0.08	-0.01	0.06	0.00	0.02	0.16	0.02	0.21	0.01	0.19	0.03	0.06	0.00	0.19	0.02	-0.01	-0.02	0.05	0.00
	Lyrical Abstraction	-0.01	0.02	-0.01	-0.02	-0.02	0.00	0.11	0.01	0.09	0.02	0.03	0.01	0.02	-0.01	0.06	-0.01	0.00	0.02	-0.01	0.01
	Magic Realism	0.00	0.07	0.04	0.14	0.00	0.05	0.14	0.02	0.13	0.06	0.13	0.06	0.01	0.02	0.11	0.15	0.00	0.06	0.07	0.04
	Neo-Expressionism	-0.01	0.10	0.02	0.04	0.01	0.03	0.13	0.00	0.20	0.03	0.18	0.04	0.05	0.01	0.11	0.05	0.00	0.03	0.05	0.00
	Pop Art	0.00	0.06	0.00	0.03	0.01	0.02	0.11	-0.01	0.16	0.04	0.20	0.02	0.02	0.01	0.11	0.06	0.00	0.06	0.09	0.01
	Post-Impressionism	-0.01	0.10	0.00	0.11	0.01	0.05	0.05	-0.01	0.13	0.03	0.18	0.02	0.03	0.03	0.20	0.05	0.05	0.03	0.05	0.00
	Surrealism	0.02	0.11	0.02	0.01	0.00	0.03	0.19	0.00	0.16	0.05	0.19	0.03	0.02	0.00	0.09	0.03	0.02	0.11	-0.01	-0.01
Post Renaissance Art	Baroque	-0.01	0.20	0.02	0.10	0.05	0.09	0.29	0.04	0.23	0.06	0.22	0.03	0.01	0.02	0.16	0.12	0.00	0.01	0.12	0.00
	Neoclassicism	-0.01	0.10	0.00	0.10	0.02	0.04	0.23	0.01	0.20	0.10	0.26	0.00	0.03	0.04	0.26	0.09	0.00	-0.01	0.13	0.00
	Realism	0.03	0.06	0.00	0.09	0.04	0.10	0.24	0.01	0.22	0.08	0.20	0.02	0.09	0.03	0.25	0.06	0.00	-0.02	0.09	0.00
	Rococo	0.00	0.09	-0.01	0.07	0.03	0.08	0.24	0.00	0.21	0.08	0.27	0.02	0.07	0.02	0.28	0.04	0.00	0.00	0.14	0.00
	Romanticism	0.00	0.20	-0.02	0.09	0.02	0.08	0.31	0.01	0.23	0.07	0.13	0.04	0.05	0.04	0.23	0.09	-0.01	-0.02	0.19	0.05
Renaissance Art	Early Renaissance	-0.01	0.17	-0.01	0.14	0.07	0.00	0.18	0.04	0.12	0.03	0.28	0.00	0.03	0.05	0.30	0.12	-0.01	-0.02	0.05	0.00
	High Renaissance	0.01	0.10	-0.02	0.07	0.02	0.09	0.23	0.06	0.13	0.04	0.27	0.02	0.00	0.00	0.14	0.05	0.01	-0.02	0.04	0.00
	Northern Renaissance	-0.02	0.10	-0.02	0.05	0.07	0.09	0.22	0.04	0.14	0.05	0.31	0.02	0.04	0.04	0.26	0.11	0.00	0.02	0.10	0.00

Figure 4: Annotator agreement (Fleiss' κ) per emotion and art category. The number of items in each category is shown in Table 2.

Emotion	Agree..	Anger	Antici..	Arroga..	Disagr..	Disgust	Fear	Gratitu..	Happi..	Humili..	Love	Optimi..	Pessi..	Regret	Sadne..	Shame	Shyne..	Surpri..	Trust
agreeableness	1.00	0.11	0.35	0.17	0.12	0.12	0.14	0.23	0.30	0.25	0.18	0.21	0.12	0.12	0.11	0.09	0.11	0.21	0.38
anger	0.17	1.00	0.24	0.26	0.24	0.28	0.35	0.19	0.18	0.19	0.16	0.15	0.22	0.20	0.26	0.20	0.16	0.22	0.22
anticipation	0.04	0.02	1.00	0.04	0.03	0.03	0.05	0.05	0.10	0.06	0.05	0.06	0.03	0.03	0.04	0.02	0.02	0.25	0.15
arrogance	0.16	0.16	0.26	1.00	0.17	0.15	0.18	0.19	0.20	0.18	0.17	0.14	0.15	0.14	0.15	0.10	0.12	0.21	0.33
disagreeableness	0.15	0.18	0.32	0.22	1.00	0.29	0.31	0.27	0.26	0.31	0.21	0.14	0.25	0.22	0.22	0.16	0.19	0.34	0.24
disgust	0.07	0.11	0.14	0.09	0.14	1.00	0.16	0.12	0.12	0.14	0.09	0.06	0.13	0.11	0.12	0.09	0.09	0.14	0.10
fear	0.07	0.12	0.20	0.10	0.14	0.14	1.00	0.12	0.12	0.14	0.09	0.06	0.17	0.15	0.27	0.11	0.09	0.16	0.12
gratitude	0.15	0.08	0.27	0.13	0.14	0.13	0.14	1.00	0.34	0.34	0.23	0.21	0.14	0.14	0.14	0.07	0.10	0.20	0.43
happiness	0.04	0.02	0.11	0.03	0.03	0.03	0.03	0.08	1.00	0.15	0.16	0.18	0.03	0.03	0.04	0.02	0.02	0.07	0.13
humility	0.07	0.03	0.14	0.05	0.07	0.06	0.07	0.15	0.31	1.00	0.13	0.15	0.07	0.08	0.09	0.04	0.05	0.09	0.20
love	0.07	0.04	0.17	0.07	0.07	0.06	0.07	0.15	0.49	0.19	1.00	0.19	0.06	0.07	0.08	0.04	0.06	0.14	0.24
optimism	0.06	0.03	0.13	0.04	0.03	0.03	0.03	0.09	0.35	0.15	0.12	1.00	0.03	0.04	0.04	0.03	0.03	0.06	0.20
pessimism	0.13	0.15	0.28	0.17	0.23	0.24	0.35	0.24	0.25	0.27	0.17	0.13	1.00	0.25	0.42	0.17	0.15	0.26	0.21
regret	0.17	0.18	0.32	0.22	0.26	0.27	0.39	0.31	0.33	0.39	0.25	0.19	0.33	1.00	0.49	0.23	0.18	0.29	0.27
sadness	0.06	0.09	0.17	0.08	0.10	0.11	0.27	0.11	0.13	0.16	0.10	0.07	0.20	0.18	1.00	0.14	0.08	0.14	0.14
shame	0.17	0.23	0.24	0.20	0.24	0.27	0.39	0.20	0.20	0.25	0.18	0.17	0.28	0.29	0.48	1.00	0.22	0.24	0.20
shyness	0.19	0.17	0.28	0.21	0.25	0.24	0.27	0.26	0.26	0.31	0.23	0.18	0.23	0.21	0.25	0.20	1.00	0.24	0.25
surprise	0.06	0.04	0.53	0.06	0.08	0.06	0.08	0.08	0.13	0.09	0.09	0.06	0.06	0.06	0.07	0.04	0.04	1.00	0.24
trust	0.07	0.03	0.23	0.07	0.04	0.03	0.04	0.12	0.17	0.13	0.11	0.14	0.04	0.04	0.05	0.02	0.03	0.17	1.00

Figure 5: The proportion of votes for each pair of emotions. The number in cell *(i,j)* shows the proportion of items annotators labeled with both emotions *i* and *j* out of all the items annotators labeled with emotion *i*.

Arabic Data Science Toolkit: An API for Arabic Language Feature Extraction

Paul Rodrigues, Valerie Novak, C. Anton Rytting, Julie Yelle, Jennifer Boutz
University of Maryland
Center for Advanced Study of Language
College Park, MD
{prr,crytting,jyelle,jboutz}@umd.edu

Abstract

We introduce Arabic Data Science Toolkit (ADST), a framework for Arabic language feature extraction, designed for data scientists that may not be familiar with Arabic or natural language processing. The functions in the toolkit allow data scientists to extend their algorithms beyond lexical or statistical methods and leverage Arabic-specific linguistic and stylistic features to enhance their systems and enable them to reach performance levels they might receive on languages with more resources, or languages with which they have more familiarity.

Keywords: Arabic, Data Science, Social Media

1 Introduction

Data scientists working on language processing problems prefer statistical methods, because they are language-independent. Best performing methods in document classification problems, however, utilize features that require knowledge of the language text being analyzed. This is especially true with Arabic. Statistical methods, such as bag of words, often are not sufficient for Arabic due to the language's morphological complexity (Kulick et al., 2006). (One example of morphological knowledge benefiting statistical parsing may be found in Yuval et al. (2013).) Since data scientists can already implement linguistic-universal statistical methods with existing software [1], this toolkit focuses on those features that require linguistic or psychological knowledge and the resources to drive them. The Arabic Data Science Toolkit (ADST) is developed primarily in Python, using Flask [2] to run as a web service server. The result is an API accessible via HTTP, allowing access that is not restricted by programming language. Inputs and outputs to the methods are transmitted in a JSON-like format. While intended to run as a server, individual features are accessible via individual scripts implemented in Python and Perl.

Data scientists may throw a large number of features at a classification problem, without initial regards to efficiency, and use feature selection methods to determine which of these features provide the most signal to their problems. They can then choose a subset of those features that are the most useful for a production system, in order to speed up execution time in a repetitive or realtime/online use case. We designed ADST for the flexibility to extract all features, feature sets, or specific individual features, as required.

Language-specific resources are somewhat difficult to find for those unfamiliar with a given language or with natural language processing (NLP) in general. Toolkits such as the Natural Language Toolkit (NLTK)[3] make this easier by uniting disparate tools under one framework. Toolkits designed for multiple languages, such as NLTK, tend to be popular, due to both ease of discovery, and language-flexibility. These toolkits may link to tools with language-specific models, but they tend to prefer linking to tools with open-source licensing. Due to licensing restrictions, they may not always integrate the highest performing Arabic tools (e.g. MADAMIRA). ADST is interested in integrating with the best performing Arabic NLP tools.

A similar tool developed for English is *Structured Programming for Linguistic Cue Extraction* (SPLICE) (Moffit and Giboney, 2015). This tool is a web-hosted API and GUI for linguistic feature extraction. The tool includes Parts of Speech (e.g. Number of Nouns, Verb Ratio), Immediacy (e.g. Number of Passive Verbs, Passive Verb Ratio), Tense (e.g. Past Tense Count, Past Tense Ratio), Positive and Negative Self Evaluation Lexicons, Deference Lexicons (e.g. Asks Permission variable, Submissiveness Ratio), affect lexicons, spoken word counts (e.g. hedging verb count, disfluency count), sentiment lexicon counts, and calculates scores for common English readability metrics (e.g. Flesch-Kincaid Grade Level).

From the beginning, we felt it would be useful to target informal social media language, which poses some unique challenges for Arabic. Many of the tools designed for Arabic focus on Modern Standard Arabic (MSA), while much of the conversation on social media exhibit colloquial varieties. Since many new Arabic tools have focused on adding Egyptian dialect components, and much of the conversation on social media does include Egyptian dialect, we aimed to build a tool that would work on either MSA or Egyptian. When we

[1] Such as scikit-learn's CountVectorizer `http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html`

[2] `http://flask.pocoo.org`

[3] `http://www.nltk.org`

discover a feature that has regional variants which are easy to add, we have elected to include them. While we include colloquial language use, our toolkit does not process Arabic chat alphabet, or Arabic that is written in Latin script.

2 Feature Focus

Document classification utilizes features extracted from text to train a model that can associate text to a label in a class of labels. Many classification problems utilize the same features, but features may be ranked or weighted differently to predict the different classes. We settled on two common document classification problems for our initial target in the ADST, emotion classification and authorship attribution. Below, we describe the two classification problems, and explain their similarities and differences.

2.1 Emotion classification

Emotion classification is a class of problems concerned with the identification of an author's latent emotional state at the time of writing. For example, the presence of هههه *hahahaha* may indicate the author is happy. Labels in the emotion class differ in implementations of emotion classification problems due to a wide assortment of emotion category sets available in the emotion literature. One core set of emotions often used in classification implementations are the Ekman emotions (Ekman, 1992) which consist of anger, disgust, fear, happiness, sadness, and surprise.[4]

In addition to category labels, some who work on emotion classification are interested in the intensity of the emotion. You could imagine that somebody who typed هههههههه *hahahahahahaha* may have been more happy than somebody who merely typed هههه *hahaha*. In the intensity use case, not only the presence of a laugh class of text, but the length of that text is important signal.

2.2 Authorship attribution

Authorship attribution compares an author's unique writing characteristics to a set of existing author examples and selects the closest author from that set. In this document classification problem, the mere presence of laughter, or the length of laughter in a text, is not an adequate signal to author attribution. Some authors may exhibit laughter like هههههه *hahahaha*, while some might laugh like هيهيهي *heeheehee*. These distinctions could supply useful information for some classification problems and require the ability to return them explicitly.

3 External Resources

The ADST utilizes several publicly-available Arabic resources to power the features it produces. These resources are high quality, but do not by default prepare feature matrices appropriate for document classification.

MADAMIRA[5] MADAMIRA is a toolkit for Arabic morphological analysis, tokenization, lemmatization, and diacritization (Pasha et al., 2014). What sets MADAMIRA apart from other tools that perform these functions is that it takes word context into account, disambiguating each word in relationship to the sentence it is a member of. MADAMIRA contains models for both MSA and Egyptian. POS Settings are configurable in the MADAMIRA configuration file. At the present time, we are utilizing both the ATB and ATB4MT tokenization schemes and the Egyptian configuration file, but this could be modified in future classification experiments.

NRC Word-Emotion Association Lexicon[6] The NRC (National Research Council) Word-Emotion Association Lexicon is a lexical database of individual words to corresponding emotions and sentiment (Mohammad and Turney, 2013). It labels for each word the emotion categories of joy, sadness, anger, fear, trust, disgust, surprise, and anticipation, as well as for positive and negative sentiment. The database was crowdsourced in English, and then translated into 39 additional languages, automatically, and without context, using Google Translate.

Arabic Word List for Spell Checking For features involving misspellings and creative spellings (such as word elongation) we make use of a freely available Arabic word list (Attia et al., 2014)[7]. This wordlist is described as automatically generated from the AraComLex open-source finite state transducer (Attia et al., 2014), enriched with words from the Arabic Gigaword corpus, and validated against the Microsoft Word spell checker.

4 Feature sets

This section discusses the features included in the ADST, and the research required to create them. We start by introducing features relevant to social media and useful when analyzing informal text. Next, we briefly discuss Emotion category features, which are useful for emotion identification. Arabic stylistic features follow, which help identify individual nuances in writing. A section on word category lexicons is next, which provides some topic identification features to the data scientist. This section also includes a description of the method we use to match dialectal Arabic text to words in lexicons. A section on morphological features is included, whose features can be used for a variety

[4] The Ekman emotions are defined on facial expression differences, and are not defined for text annotation. Additionally, psychologists have proposed many alternative category sets.

[5] https://secure.nouvant.com/columbia/technology/cu14012/license/492

[6] http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

[7] Arabic Wordlist 1.6, downloaded from www.attiaspace.com on 2016-11-11, also available from http://sourceforge.net/projects/arabic-wordlist/ Version 1.6 contains 9,196,215 fully inflected words, slightly fewer than the version described in the paper (presumably because of further vetting).

of classification problems. Finally, we provide a section on spelling and typographical features, useful for author identification.

4.1 Social Media Features

4.1.1 Laughter

We developed a set of scripts capable of detecting strings that represent laughter and calculating their length to assess emotion intensity. Just as laughter may be represented more than one way in the written form in English – for example, "hahaha," "heeheehee," or "ho ho ho!" – laughter is represented multiple ways in Arabic. Given that Arabic is rich in uvular, pharyngeal, and glottal phonemes (sounds produced at the back of the throat) and contains a distinction between short and long vowels, the possibilities for representing laughter in the Arabic script are particularly wide-ranging. Our Python scripts were designed to detect as many strings as possible that are likely to represent laughter in Arabic script, based on observations of the type of laughter attested in Arabic-language social media usage. Some representations of laughter in the Arabic script are uniliteral strings (i.e. strings consisting of a single character) formed by the repetition of a single voiceless fricative – either the velar/uvular خ *khā'*, the pharyngeal ح *ḥā'*, or the glottal ء *hā'* – in which the presence of a short vowel is implied. We thus developed a script to detect and count the length of laughter strings formed by repeating three times or more any one of these Arabic consonants, e.g. خخخ *khakhakha*, ححح *ḥaḥaḥa*, and هههه *hahaha*. We also developed numerous scripts designed to work with laughter strings involving more than one character. For multiliteral strings, the set of laughter-associated characters was expanded to include the consonants ق *qāf*, likely representing its colloquial pronunciation as a glottal stop) and ع *'ayn*, as well as the long vowels ا *alif* and ي *yā'*. Our scripts were designed to take into account that an Arabic biliteral laughter string may consist of either a consonant–long vowel pair (e.g. هاها, *hāhā*) or a consonant-consonant pair in which the presence of a short vowel is implied (e.g. معهع, *ha' ha'*). Our scripts were also developed to detect such strings regardless of whether the repetition of and alternation between the two characters is regular (e.g. هخهخه, *hakhahakhaha*) or irregular (e.g. هخهخههه, *hakhahakhahahaha*). We also specifically created a script to detect a biliteral string in which the consonant jīm is regularly alternated with the letter alif (e.g. جاجاجا, *jājājā*) and whose appearance in Spanish-language contexts and adjacency to laughter emoji suggests that it is an Arabic transliteration of Spanish-language laughter (i.e. jaja). Also attested in social media contexts are triliteral, quadriliteral, and quintiliteral strings that combine any of three, four, and five characters, respectively, from the following set of laughter-associated characters: ء *hā'*, ح *ḥā'*, خ *khā'* , ع *'ayn*, and ق *qāf*, and ا *alif*. We are currently adding these multiliteral laughter categories to ADST.

Some tools, like the English syntactic parser TweeboParser (Owoputi et al., 2013)[8] do recognize laughter, but conflate the laughter categories to a single category, which removes the uniqueness of the laughter type signal.

In addition to laughter scripts, we wrote scripts to detect ululation, another type of emotional vocalization that is rooted in Arab culture (as well as the cultures of many other African and Asian societies). Ululation is a high-pitched, elongated vocalization with a trilling quality, is attested on both positive occasions (e.g. celebrating a wedding) and negative ones (e.g. mourning at a funeral). Thus, it can connote either joy or sorrow, depending upon the context in which it is used. Ululation connotes high intensity of emotion regardless of whether the nature of that emotion is positive or negative. We based our ululation-detection scripts on two main written variants attested in Arabic-language Twitter content: يويويو (yūyūyūyū) and لولولي (lūlūlūlī). The latter variant even had a hashtag (#لولولي) associated with it.

4.1.2 Emoji

Emoji are Unicode characters that can be used to add additional context for interpretation of a written text. ADST includes two feature sets around Emoji. The first returns a histogram of the emoji found in a text. The second utilizes a compiled lexicon associating Emoji to the emotions that may be communicated by that Emoji. This relies on a previous study which had English- and Arabic-native speakers annotate for whether emoji in context evoked sadness, contempt/disgust, affection, fear, happiness, positive humor, anger, confidence/pride, or no emotions. The ADST utilizes this lexicon to return a histogram of the emotional categories that may be signaled by an author's emoji usage.

4.2 Emotion Lexicon

In addition to the emoji lexicon, the ADST uses the NRC Word-Emotion Association Lexicon (Mohammad and Turney, 2013) to provide emotion information of the words used in the text. The ADST examines the lexicon and returns word counts for each of the emotive categories they used (joy, sadness, anger, fear, trust, disgust, surprise, and anticipation). Additionally, it returns word counts for the two sentiment categories used in the lexicon (positive, negative).

4.3 Arabic-focused Stylistic Features

Many author attribution systems utilize features that attempt to capture linguistic style. These stylistic features might consist of punctuation frequencies, or word frequencies. We have taken some of the features used in the English author attribution literature, and extended them for Arabic. Additionally, Arabic has several unique stylistic features, that we extract in ADST for future research.

A punctuation script, for example, outputs the type and frequency of each type of punctuation. These

[8] http://www.cs.cmu.edu/~ark/TweetNLP/

include English and Arabic versions of the question mark (? and ؟) as well as punctuation that the languages have in common. A quotation script searches and counts the symbols and punctuation used as quotation markers.

Arabic characters appear in different shapes, depending on their position in a word. The Unicode code pages for Arabic characters include a code page that utilizes this contextual information to provide the shaping of the character, as well as a code page for presentation forms. Arabic in the Presentation Form unicode block can look identical to characters in the contexual Unicode block, but consist of different underlying character codes, which can hinder pattern matching on Arabic. Some older devices utilize presentation mode exclusively, which may provide additional information for the identification of authorship. A presentation form script identifies these alternate Arabic characters by searching in the Unicode range of \uFB50-\uFDFF and \uFE70-\uFEFF for Unicode points for Arabic letters which are not the canonical form (and hence would not normally be produced by typical input methods) but which are visually identical to canonical Unicode Arabic text.

Unicode also has full Arabic words as single Unicode points. For example, محمد *Muḥammad* is a name/word that can be denoted by the single unicode point \ufdf4. The phrasal characters script identifies the type and frequency of these characters in the text.

There are two numbers scripts, one of which outputs a frequency of Arabic numbers and the other a frequency of Hindi numbers used in the text. Lastly, there is a script that counts the frequency of the tatweel (the Arabic lengthening character) and the frequency of Arabic diacritics (short vowels).

These scripts provide Arabic-specific features of text to the data scientist, who might not know that they are salient to the document classification problem.

4.4 Word Category Features

We developed several custom lexicons for this toolkit that target feature sets of interest to emotion identification and authorship attribution. This allows a data scientist to submit their text to determine if words in their text belong to particular pre-defined categories.

To build the lexicons, we drew from various published dictionaries of MSA and the Badawi and Hinds (1986) dictionary of Egyptian Arabic. We used web-based glossaries for topics that were not covered adequately in print dictionaries, in particular insulting and profane language. We also supplemented the lexicons with researcher-identified multi-word entries from social media data. Many of these expressions are not included in print resources.

For each lexicon type, we provide an MSA and Egyptian Colloquial (ECA) lexicon (e.g., MSA Pious Expressions Lexicon and ECA Pious Expressions Lexicon). These lexicons were created independently of each other and the entries do not correspond exactly. The ECA Abusive Language Lexicon encompasses the ECA version of the content of both the MSA Abusive Language and Targeted Abusive Language Lexicon.

We chose to keep discrete categories separate in the lexicons, however, as users of the toolkit may choose to merge lexicons to create different feature combinations depending on their interests. For example, the Polite Expressions, Honorifics, and Pious Expressions Lexicons could be combined to create a single Respectful Language/Politeness Lexicon and feature. Similarly, the various categories of lexicons related to abusive language might be merged into a single lexicon.

4.4.1 Lexicons and Lexicon Expansion

Arabic inflects words for gender and number, and has additional morphological complexity resulting from the affixation and concatenation of conjunctions, prepositions, and the definite article. This complexity makes it difficult to search a text for a word. In order to identify documents containing words and phrases from the word category lexicons, we developed machine-readable versions of these lexicons, expanding the simple lexicons to include Arabic's morphological complexity, as well as commonly occurring orthographic variations and common misspellings.

The application of regular expressions designed to handle orthographic variation was done by searching for instances of specific characters and patterns (e.g., ة *ta marbuta*, ي *yaa'*, and أ *hamza-on-alif*) and replacing them with the appropriate regular expression (respectively, [ةه], [یی], and [آأإا]).

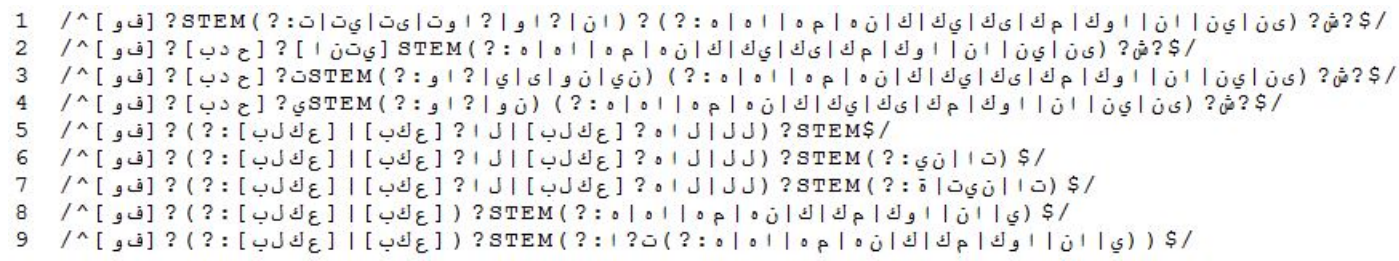
The morphological expansion of Arabic words was handled according to two basic part-of-speech categories (verbs and nouns) and their corresponding morphological behavior. Therefore, two basic approaches were used: one for verbal inflectional paradigms and the other for nominal inflectional paradigms (a category that includes adjectives).

For morphological expansion of verbal forms two basic paradigms were used: the perfective (الماضي) and the imperfective (المضارع). In the following regular expressions, the string STEM represents the verb stem. It should be noted that although these paradigms are based on MSA morphology, they have been extended to cover dialectal Arabic morphology, such as the Egyptian future particle ح and the Levantine and Iraqi progressive aspect particles ب (Levantine) and د (Iraqi), as well as the negative ش enclitic used in Egyptian and some Levantine dialects.

Regular expression for the perfective verb can be found in Line 1 of Figure 1. Regular expressions for the imperfective verb can be found in Line 2 of Figure 1. 1st pers. sg./pl., 2nd pers. masc. sg, and 3rd pers. masc./fem. sg 2nd pers.masc.pl and 2nd pers.fem.sg can be found in Line 3 of Figure 1. 3rd pers.masc.pl can be found in Line 4 of Figure 1.

Note that our lexicon did not include irregular verb paradigms, hence we wrote our regular expressions to cover only regular paradigms.

For morphological expansion of nominal forms two basic paradigms were used: nominal form with posses-


1 /^[ف و]?STEM(?:[ت:ة|ات|ون|ين|ان|ون|ين|ان|ه|ها|هم|هن|ك|كم|كن|ي|نا|ه|ها|هم|هن|ك|كم|كن|ي|نا|ه|ها|هم|هن|ك|كم|كن|ي|نا

these transitional words and phrases could be used by data scientists for sentiment or emotion classification. Our initial lexicon of words and phrases was in English and leveraged The Writing Center at The University of Wisconsin-Madison (2016). This collection is thematically divided into types of transitions, such as addition, contrast, qualification and many others. The list provided a sufficient foundation from which to begin compiling a more complete list in English and to provide translated equivalents in Arabic. Searches of a number of translation and Arabic language web sources (wordreference.com, Google translate, Quizlet.com) for each term in English led to the Arabic counterparts in meaning. These Arabic counterparts were evaluated for suitability by fluent speakers of Arabic. In several cases, an English term had multiple Arabic counterparts with similar meanings, so these were also added to the list of Arabic terms. Finally, the Arabic list was expanded to include common orthographic variations of the word or phrase, such as removing the short vowels from words, and replacing أ with ا. The script searches for the transitional word or phrase in the text and returns the word/phrase and the frequency of its usage in the text.

4.5 Morphological Features

MADAMIRA is a core component to several of the scripts that make up ADST, because the tool is able to identify the root or dictionary form of the word (lemmatize) and morphologically analyze Arabic words using the word's context in the sentence. While MADAMIRA produces a word by word analysis, the majority of our features are histograms of frequency counts, which could be used by a data scientist as a feature matrix. The ADST scripts which utilize MADAMIRA parse its output and return requested features of a given text. Many of these scripts find and isolate grammatical information and conduct frequency counts for the given text, extracting features for gender, number, person, aspect and clitics (including enclitics and proclitics). Other scripts take advantage of MADAMIRA's lemmatizing feature. A bag of words script returns the frequency of the lemmas in the text, while an out of vocabulary script returns those words in the text that MADAMIRA is unable to analyze. The lemmatizing function of MADAMIRA is also used as a feature to flag the behavior of affixation of the conjunction و waw *and* with the following word. Another script, a variation on the bag of words script, strips diacritics from the stemmed words. Lastly, there is a part of speech (PoS) script, which returns a PoS frequency list from MADAMIRA.

4.6 Spelling and Typographical Features

4.6.1 Elongated words

One hallmark of emotionally expressive informal writing in social media is the use of repeated letters to express emphasis. An English example of this is "LOOOOOL" for "LOL" (laughing out loud). As with direct representations of laughter, we hypothesize that the length of the repetition may correlate with degree of emphasis or intensity of underlying emotion. Therefore, we want to return the length of the repetition as well as the underlying word in its canonical spelling. The component for detecting repeated letters utilizes a wordlist such as Attia et al. (2012) and looks for words not on that list. For every out-of-vocabulary word, it removes repeated letters and then checks if the resulting string is on the list. If it is, it reports the original string, its length, the canonical spelling, and the number of extra letters found. Since it is possible for a correctly spelled Arabic word to have two adjacent instances of the same letter (e.g., حوثيين), only strings of three or more adjacent, identical letters are shortened, and instances where the canonical spelling has either one or two copies of the letter are both considered.

4.6.2 Misspelled words

Since informal writing will generally have more misspelled words than edited text, we also have a component for detecting misspellings. We distinguish two main types of misspellings:

1. Common and commonly accepted spelling variants, such as using an alif character alone instead of a hamza (glottal stop) character seated on an alif, or the using dotless letter variants of taa marbuta and final yaa. (Hypercorrections involving these same variants would also fall into this category.) These are already handled (at least in part) by the regular expressions described in section 4.2.1.

2. 'Major' spelling variants, such as omission or insertion of a letter, a substitution not considered 'minor', or metathesis of adjacent letters.

The former type of misspelling is predicted to be relatively common, and hence relatively uninformative (though its absence may indicate a high level of education, conscientiousness, and/or formality, or use of a device with very accurate auto-correct). The latter type of misspelling may possibly indicate carelessness, fast typing, or even some level of emotional stress. It may also indicate use of a difficult-to-use text input interface. As with the component for finding repeated letters, the component for detecting misspelled words uses a wordlist (currently Attia et al. (2012)) and checks the spelling only of OOV (out of vocabulary) terms. Thus it can only detect non-word errors; real-word spelling errors are out of scope. In order to compile a list of possible intended words, it computes the n shortest paths through a weighted finite-state 'noisy-channel' model, with weights trained on a corpus of quickly typed Arabic. The tool component has a number of parameters that can be adjusted by means of a JSON configuration file, including the wordlist used, the weight (or cost) thresholds for 'minor' and 'major' spelling errors, the number of hypothesized intended words returned, and even the error model used. This component uses OpenFST as a dependency.

5 Summary

The Arabic Data Science Toolkit (ADST) is a tool for data scientists and computer programmers, who may not know Arabic, to quickly and easily extract relevant information from Arabic text, for machine learning applications. At this time, the ADST includes social media features, stylistic features, word category features, and spelling features. Each of these require some underlying Arabic resource for implementation. Our first focus was on author attribution and emotion classification problems, but the toolkit should be useful for other document classification applications as well.
We expect to continue development on this toolkit, extending it with additional features, adding new languages, and improving its ability to scale to large amounts of data.

6 Bibliographical References

References

Attia, M., Pecina, P., Samih, Y., Shaalan, K., and van Genabith, J. (2012). Improved spelling error detection and correction for Arabic. In Martin Kay et al., editors, *Proceedings of COLING 2012*, Bumbai, India.

Attia, M., Pecina, P., Toral, A., and van Genabith, J. (2014). A corpus-based finite-state morphological toolkit for contemporary Arabic. *Journal of Logic and Computation*, 24(2):455–472.

Baalbaki, R. (1995). *al-Mawrid: A modern Arabic-English dictionary*. Dar El-Ilm Lilmalayin, Beirut.

Badawi, E.-S. and Hinds, M. (1986). *A dictionary of Egyptian Arabic*. Librairie du Liban, Beirut.

Cowan, M. J. (1994). *Arabic-English Dictionary: The Hans Wehr Dictionary of Modern Written Arabic*. Spoken Language Services, Urbana, IL.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3):169–200.

Frank, A. J. and Mamatov, J. (2002). *Dictionary of Central Asian Islamic terms*. Dunwoody Press, Springfield, VA.

Kulick, S., Gabbard, R., and Marcus, M. (2006). Parsing the Arabic treebank: Analysis and improvements. In *Proceedings of the Treebanks and Linguistic Theories Conference*.

Marton, Y., Habash, N., and Rambow, O. (2013). Dependency parsing of modern standard Arabic with lexical and inflectional features. *Computational Linguistics*, 39(1):161–194.

Moffit, K. and Giboney, J. S. (2015). SPLICE beta v0.9.0. http://splice.cmi.arizona.edu/.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL*.

Pasha, A., Al-Badrashiny, M., Diab, M., Kholy, A. E., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. M. (2014). MADAMIRA: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, Reykjavik, Iceland.

The Writing Center at The University of Wisconsin-Madison. (2016). Uw-madison writer's handbook. http://writing.wisc.edu/handbook/.

Sentence and Clause Level Emotion Annotation, Detection, and Classification in a Multi-Genre Corpus

Shabnam Tafreshi, Mona Diab
George Washington University
Department of Computer Science, Washington D.C.
{shabnamt, mtdiab}@gwu.edu

Abstract

Predicting emotion categories (e.g. anger, joy, sadness) expressed by a sentence is challenging due to inherent multi-label smaller pieces such as phrases and clauses. To date, emotion has been studied in single genre, while models of human behaviors or situational awareness in the event of disasters require emotion modeling in multi-genres. In this paper, we expand and unify existing annotated data in different genres (emotional blog post, news title, and movie reviews) using an inventory of 8 emotions from Plutchik's Wheel of Emotions tags. We develop systems for automatically detecting and classifying emotions in text, in different textual genres and granularity levels, namely, sentence and clause levels in a supervised setting. We explore the effectiveness of clause annotation in sentence-level emotion detection and classification (EDC). To our knowledge, our EDC system is the first to target the clause level; further we provide emotion annotation for movie reviews dataset for the first time.

Keywords: emotion annotation, emotion analysis, multi-genre corpus

1. Introduction

Prediction of sentence-level emotion classification encompass a variety of applications such as modeling of human behaviors (Dodds and Danforth, 2010) and situational awareness in the event of disasters (Vo and Collier, 2013). As a precursor to our system development, we realize the diversity and non uniformity of existing resources with emotion tags, hence, we re-annotate existing resources in a unified framework, thereby covering multiple genres of text. The genres are as follows: emotional blog post (BLG), news headlines dataset (HLN), and movie review dataset (MOV). We present an approach and system that performs emotion detection and classification (EDC) on multiple levels of granularity, namely, sentence and clause levels. We expand the annotation scheme to cover both sentence and clause level annotations, as well as expand the emotion tag inventory from the typical Ekman 6 (Ekman, 1992) emotion labels (EK6) to 8 emotion labels based on Plutchik's Wheel of Emotions (Plutchik, 1962) (PL8).

In this study, we focus on the impact of clause-level annotation on the EDC task, which can be used effectively in a single-genre or multi-genre textual setting without significant performance loss. Similar to previous studies, we cast the EDC problem in a supervised setting. Evaluation of EDC in 10% held out data outperformed the baseline and gives the average accuracy of 81.1% and 71.3% for sentence and clause level respectively. EDC achieved better results compared to previous annotation of HLN and BLG datasets with EK6 emotion labels (average accuracy 54.7% and 73.8%). Accordingly, our contributions are as follows:

- A new set of annotation guidelines for emotion detection based on Plutchik's Wheel of Emotions.
- A uniformly annotated multi-genre data set (including old and new data) on two levels of granularity: sentence and clause levels.
- Two EDC systems on the sentence and clause levels for multiple genres leveraging clause-level annotation on sentence-level EDC systems.

The rest of this article is structured as follows: section 2. describes related work to the study; in section 3. we give data references, collection, annotation process and evaluation, and annotation challenges; section 4. explains the experiment setup and EDC description; and section 5. concludes and describes future direction of our study.

2. Related Work

Emotion detection in NLP has been studied on document, sentence, and phrase levels. Several studies investigated the problem in various data genres. We present studies most relevant to this paper. Aman and Szpakowicz (2007) collected and labeled BLG corpus using EK6 tags in sentence and phrase-level. Strapparava and Rada (2007), collected HLN set and labeled it using EK6 tags and valence, which, valence measures the polarity of each data point. HLN is used in SemEval 2007, task 14. Pang and Lee (2005) crawled web to collect MOV dataset to address rating inference problem. Mishne (2005) collected a set of blog posts - online diary entries - which include an indication of the writers's mood. Yan (2014) expanded the range of automatic emotion detection in microblogging text using three sampling strategies: random sampling, topics and events sampling, and sampling based on users. Abdul-Mageed and Lyle (2017) collected a large set of tweets using hashtags, they used Plutchik's Wheel of Emotions to create relevant hashtags, and the set is annotated using distant supervision method. To date, sentence-level emotion classification has been studied by a large group of researchers (Aman and Szpakowicz, 2007; Strapparava and Rada, 2007; Mishne, 2005; Yan, 2014; Das and Bandyopadhyay, 2010; Ghazi et al., 2010; Kim et al., 2010; Mohammad, 2012; Özbal and Pighin, 2013; Abdul-Mageed and Lyle, 2017), who ad-

dressed the EDC task on the document and sentence levels, to our knowledge, nobody investigated automatic tagging on the clause level and the impact of clause-level on sentence-level emotion classification, and that distinguish our work from previous works.

3. Data Description

We aim to create a multigenre corpus annotated with emotion tags on the clause and sentence level. We would like to cater to fine grained emotion detection with the goal of eventually building systems that detect emotion intensity. Toward that goal, we create a unified multigenre data set annotated on the clause and sentence levels. Moreover, we compared the typical EK6 to other tag sets that are more fine grained and well established in the psychology literature. We opted for Plutchik's Wheel of Emotions. Below is a detailed description of the data and the annotation process.

3.1. Corpus

We combined and annotated several previously annotated data sets on the sentence level for various types of emotions. The first data set is a emotional blog post (BLG) (Aman and Szpakowicz, 2007) where people typically express their emotions and opinions about social/personal events, politics, products, etc. This dataset comprises 4115 sentences. The second data set, a news headlines dataset (HLN) (Strapparava and Rada, 2007) crafted by creative people to possibly provoke emotions comprises 1250 sentences. Both BLG and HLN were annotated originally using the EK6 tag set. Finally, the third data set, a movie review dataset (MOV) (Pang and Lee, 2005) where people express their opinion about movies, sound tracks, and casts. The MOV data set contains 11,855 sentences. The MOV data set is annotated for sentiment intensity. The total number of sentences in the collection is 17,220. We extract clauses from the sentences in the three corpora using the Stanford parser (Klein and Manning, 2003) from the CoreNLP toolkit (Manning et al., 2014). In each sentence parse tree, we extract the labels, SBAR, SBARQ, etc. according to the Penn Treebank's clause labels of the parse trees (Marcus et al., 1993) identifys the sentence clauses. The total number of clauses corresponding to 17,220 sentences is 29,938. 7,458 of the sentences comprise a single clause. We refer to this sentence-level corpus as SBHM and clause-level corpus as CBHM.

3.2. Annotation Process

Annotating emotional data is a challenging task, since people perceive various experiences differently. This is expected to be the case especially when the data is extracted from social media platforms like forums and blogs. To develop appropriate emotion categories, we carried out our annotation procedure in two stages: a pilot stage and an annotation stage.

Pilot Stage: our work was guided by the following research questions:
(1) what emotion categories can be best suited for different genres in our corpus, what is the appropriate tag set for our multigenre corpus: Ekman's six basic emotions (EK6) or Plutchik's eight basic emotions (PL8)?
(2) In case of clause level annotation, what is the appropriate presentation method to the annotators?
To answer question (1), we set up an online survey. We selected 518 single clause sentences from the BHM corpus such that they equally represented the three underlying corpora BLG, MOV, HLN. Three annotators, graduate students, worked on the pilot data. We provided annotators with detailed guidelines regarding the task. We ran two pilot annotations: one asking annotators to use the EK6 tagset and the second where they were asked to use the PL8 tagset. Cases of disagreement between the annotators were discussed until a Fleiss Kappa K= 0.7 was reached for both pilot annotation exercises. The output of the pilot stage was an agreement to use the PL8 basic emotions, since it was a better reflection of the data. In addition, the annotators suggested adding the labels *interest, disappointment, confusion*, and *frustration*, but since these were not very frequently assigned (less than 2%), we decided to use the label *other-emotion* instead of adding these extra ones. We also added *no-emotion* to the tag set as an option available to annotators. Accordingly, based on feedback, we ended up with 10 labels including: PL8 set *joy, trust, anticipation, surprise, fear, sadness, disgust, anger*, *no-emotion*, and *other-emotion*. These annotations were collected on the sentence level. To address the second question, we further randomly selected 20 clauses testing how to demonstrate the clauses to the annotators. Based on a survey completed by 10 people, majority voted for marking clauses within each sentence and asking for an emotion tag, as opposed to showing the clauses in isolation without context. Hence, when annotating clauses, we mark each clause within its sentence, and provide it to the annotator. Below we demonstrate an example, clauses are marked as underline text:
Clause-1: It takes a really long , slow and dreary time to dope out what TUCK EVERLASTING is about .
Clause-2: It takes a really long , slow and dreary time to dope out what TUCK EVERLASTING is about .

The following are the points we noted in the guidelines:

- We asked our annotators not to think of words or emotion clauses out of context, rather they should think about them within the context for sentence annotation.
- We noted to them to not annotate the sentences and clauses according to their (e.g., cultural, religious) backgrounds.
- Our annotators were free to choose any dictionaries or resources to judge the emotion in the sentences.
- We provided one example for each emotion label (e.g. "Siri does not pick my accent and drives me crazy", where the emotion label is *anger*.).

Annotation Stage: we set up the annotation job in CrowdFlower,[1] an online crowdsourcing platform. We separate the setup for sentence level annotation from clause

[1] https://www.crowdflower.com

Dataset	joy	trust	anti	surprise	sad	fear	anger	disgust	other-emo	no-emo
sentence-level										
HLN	106	6	56	31	83	68	28	55	0	662
BLG	689	43	260	150	312	132	192	255	13	2051
MOV	4875	26	119	255	258	63	20	5145	13	1081
SBHM (total)	5670	75	435	436	653	263	240	5455	26	3794
clause-level										
HLN	93	1	13	7	35	12	14	45	1	1081
BLG	1138	28	278	81	291	148	258	831	16	3665
MOV	8772	26	126	130	228	154	63	9651	9	2743
CBHM (total)	10003	55	417	218	554	314	335	10527	26	7489

Table 1: Multi-genre corpus consists of three genres and the distribution of emotion categories per sentence and clauses. Category joy and disgust are notable in movie review.

level, due to differences in task objective and slight differences in the guidelines. As such, to set up the two annotation jobs we took the following steps:

- We used the emotion categories developed in the pilot stage.
- We simplified the guidelines, which we used at the pilot stage. The only factor we noted to the annotators in the simplified guidelines was to not take emotion words or expression out of context for sentence annotation.
- We provided one example for each emotion label.
- We mixed the three datasets together and put every 5 sentences/clauses in one HIT with a compensation of $0.07 (7 cents). [2]

We provided 5000 single clause sentences annotated in sentence-level task as gold labeled data for clause-level annotation. We excluded the remaining single-clause sentences from clause-level annotation.

3.3. Annotation Evaluation

Each sentence/clause was annotated by 3 annotators. Crowdflower platform assigns a 'trust' score per annotation task. This score is a number between 0 and 1, and it is defined by the system as the accuracy score of an annotator. We required that only judgments with trust score above 0.7 are accepted. The system calculates 'trust' as follows: each HIT contains one gold item, the trust score is the percentage of correct answers to gold items. Judgments from annotators with score being below the threshold are tainted. To demonstrate the agreement among our annotators, we calculate per emotion tag, per datapoint, the number of judges who agreed on the emotion tag. We call this metric agreement class category (ACC). In our tasks, we asked for 3 judgments per datapoint and agreement of a minimum of two judges. Table 1 shows the statistics of the annotated corpora per emotion. We note that a significant number of units (sentences and clauses) are tagged with anticipation, even more than a basic EK6 emotion such as *surprise* which validates our choice of PL8 as a tagset. Table 2 shows the ACC in the annotated corpora per emotion label. The results show that on average, we achieved 79.95% IAA on sentence-level and 62.74% IAA, on clause-level, where at least two judges agreed on the emotion label per item. Table 3 presents the statistics on the EK6 tags of the original previous annotation on the HLN sub corpus as well as the BLG sub corpus of BHM and our current annotations on the sentence level. We report the HLN Pearson correlation as reported by the authors of the HLN annotated corpus and Kappa statistic on the BLG corpus. We note that the ACC values in Table 3 are different from Table 2 since these exclude statistics from the MOV corpus. For HLN, despite the fact that the two metrics are different, ACC and Pearson correlation, we note that the ACC metric is higher per emotion label in our annotation setting. We note the same trend for the BLG corpus comparing ACC metric and the Kappa statistic except for the emotion label *joy*. Table 4 shows the confusion matrix between the various labels of both HLN and BLG. We note that our IAA using crowdsourcing for only the 6 basic Ekman emotions (EK6) for BLG is 78.93% compared to the original of 76% in lab annotators in the original data set. Likewise for the HLN data set, we achieve an IAA of 93.16% with EK6 using crowdsourcing compared to 53.67% in the original annotated data set. This proves the feasibility of using crowdsourcing effectively for the task. Moreover our annotation with 10 tags (PL8) achieves an overall IAA of 76.5% for BLG and 95% for HLN. This suggests that PL8 is an appropriate level of tagging. Observe that agreement among the workers in CrowdFlower is higher than what we achieved in

Emotion	ACC≥2%	
	sentence	**clause**
joy	93.03	93.82
trust	65.33	23.64
anticipation	80.23	52.04
surprise	82.80	56.88
sadness	76.11	66.25
fear	70.34	72.29
anger	63.75	68.36
disgust	**97.32**	94.64
other-emotion	26.92	0.00
no-emotion	63.78	99.52
IAA	79.95	62.74

Table 2: The ACC≥2 percentage agreement per emotion label where at least 2 annotators agreed on the same label in the BHM corpus.

[2]We borrowed the expression HIT (Human Intelligence Task) from Amazon Mechanical Turk https://www.mturk.com. In Amazon Mechanical Turk a HIT is defined as: a question that needs an answer. A HIT represents a single, self-contained task a Worker can work on, submit an answer, and collect a reward upon completion.

the pilot stage. In pilot stage, the annotators received significant instruction and we had the opportunity to discuss different aspects of the task, while in CrowdFlower we do not have knowledge about the annotators background and we are not able to connect with them. Despite these issues, we achieve a very high general IAA on the sentence level verifying that crowdsourcing is an appropriate manner to curate annotations for emotion tags. In addition, emotion tags *trust, anticipation, fear, anger,* and *sadness* are controversial. Particularly, we received a high volume of feedback for emotion tags *fear, anger,* and *sadness*, indicating that these emotion tags are confusing, interchangeable, or can be used together for tagging data points.

Table 3 presents the comparison between the emotion la-

HLN Emotion	ACC≥2%	Pearson
joy	98.11	59.91
surprise	93.55	36.07
sadness	95.18	68.19
fear	95.59	63.81
anger	89.29	49.55
disgust	87.27	44.51
avg.	93.16	53.67
BLG Emotion	ACC≥2 %	Kappa
joy	73.00	0.77
surprise	79.33	0.60
sadness	73.72	0.68
fear	79.55	0.79
anger	69.27	0.66
disgust	94.51	0.67
avg.	78.23	0.76

Table 3: Comparing the inter-agreement we achieved with HLN & BLG datasets. In both datasets our annotation achieved higher IAA results.

Emo/dataset	joy	trust	anti	surp	sad	fear	anger	disg	other-emo	no-emo
BLG										
joy	**81.4**	1.4	5.2	2.0	1.6	0.5	0.1	1.0	0.1	6.1
surprise	21.1	0.0	0.0	**47.4**	5.9	2.5	1.6	1.6	0.0	19.4
sadness	3.3	1.1	3.3	0.5	**74.3**	3.3	1.1	8.9	0.0	4.4
fear	2.5	0.0	0.8	5.9	4.2	**65.8**	2.5	5.9	0.8	11.1
anger	1.6	0.5	1.6	3.2	9.2	3.8	**53.5**	17.4	0.0	8.7
disgust	1.1	0.0	2.8	0.5	10.9	1.7	28.3	**47.9**	0.0	5.7
no-emo	7.2	1.1	7.7	2.4	4.3	1.1	1.3	3.8	0.3	**70.3**
HLN										
joy	**21.8**	0.6	10.9	4.0	0.9	0.5	0.3	0.0	0.0	61.3
surprise	1.0	0.0	1.0	**9.7**	0.0	0.0	1.0	3.2	0.0	83.6
sadness	0.5	0.5	0.5	0.0	**23.6**	4.7	2.3	8.2	0.0	59.1
fear	0.0	0.0	1.9	1.3	7.8	**26.1**	3.2	5.2	0.0	54.2
anger	0.0	1.5	1.5	1.5	3.0	3.0	**13.6**	3.0	0.0	72.7
disgust	0.0	0.0	0.0	3.8	3.8	0.0	7.6	**34.6**	0.0	50.0
no-emo	0.5	0.0	0.0	0.5	2.3	1.1	1.7	4.6	0.0	**83.2**

Table 4: Confusion matrix for different emotion labels on the sentence level in BLG & HLN datasets of the BHM corpus and the original tags.

bels in BHM and the previous tags using EK6 data set on the sentence level for BLG and HLN. We consider the original tags (row entries) as gold. We note that the overlap between the previous annotation of BLG and our current annotation is higher than the overlap with HLN. Emotion tag *anger* is commonly confused with *disgust* compared to the number of annotations for *anger*. Observe that majority of confusion is in *no-emotion* tag. We also note that 8% of the BLG sentences, which previously were annotated as *no-emotion*, are tagged with *trust* and *anticipation*.

3.4. Emotion Tagging Difficulties in the Corpus

Manually annotating emotion data is a challenging task, due to different evaluation of emotion situations by humans. According to appraisal theory (Öhman, 1999), emotions are extracted from evaluations of events that could trigger different reactions by different people. In our annotation setting our annotators could choose one emotion tag among PL8 and *no-emotion, and other-emotion*, which can be challenging and confusing. During the annotation process, we observed that annotators are confused when they have to pick one of the {*anger, disgust, fear*} or {*trust, joy, anticipation*}. As a result, we had high number of tainted annotations during annotation stage.

Below we observe annotation tags provided for three examples from movie review corpus (MOV):

(a) "Engagingly captures the maddening and magnetic ebb and flow of friendship."

(b) "Rabbit-Proof Fence will probably make you angry."

(c) "Closings and cancellations top advice on flu outbreak."

All three sentences were annotated by 4 annotators per sentence (1 annotator vote was tainted).

Sentence (a): 2 annotators tagged that sentence as *joy*, 1 tagged it as *trust*, and 1 tagged it as *no-emotion*. While the expression "flow of friendship" triggers *trust*

Sentence (b): 2 annotators tagged it as *anger*, 1 as *anticipation*, and 1 as *disgust*. Sentence (c): 2 tagged it as *no emotion*, 1 as *fear* and another as *disgust*.

4. EDC Systems Experiment Setup and Results

For classification we devise the same experiments for tagging on both granularity levels: sentence and clause levels. We have 9 classes in our data, the PL8 and *no-emotion* [3]. We split the data to (80%,10%,10%) for training, dev, test, respectively.

Supervised model: we build our model using LIBLINEAR (SVM family) in WEKA classifiers,[4]. SVM has been applied with success to emotion classification in the literature (Aman and Szpakowicz, 2007; Mishne, 2005; Yan, 2014; Das and Bandyopadhyay, 2010; Mohammad, 2012; Özbal and Pighin, 2013). We experimented with other classifiers such as Naive Bays, Decision Tree, and Random Forest, and LIBLINEAR produced better results. We build our model combining number of features like: n-gram, POS, syntactic features like presence of adjective, adverbs, or negation (syn). To show the impact of clauses in sentence-level classification we created a feature based on clause emotion tags pattern, we refer to this feature as subordinate clauses (scla). For this feature, we study the distribution of clauses emotion tags in multi-clausal sentences. We note that the majority of those sentences with multiple clauses tend to have clauses with specific emotion labels (e.g. sentence emotion tag *joy*, have clauses with tags {*trust, anticipation, no-emotion, and surprise*}). We model this feature as an 8-dimension vector, where

[3] Authors release the dataset for research purposes upon the requests.

[4] http://www.cs.waikato.ac.nz/ ml/weka/

Emo-tag	NLH		BLG		MOV		SBHM	
	LIB	RULE	LIB	RULE	LIB	RULE	LIB	RULE
joy	92.3%	42.1%	66.2%	77.3%	85.5%	89.9%	83.4%	87.4%
trust	0	33.3%	0	28.7%	33.3%	0	13.3%	20.0%
anti	40.0%	26.3%	41.9%	35.9%	11.8%	42.8%	34.3%	37.2%
surprise	40.0%	28.5%	37.5%	61.5%	29.3%	29.4%	32.3%	36.7%
sadness	28.6%	0.06%	57.6%	38.4%	45.5%	31.2%	51.3%	34.6%
fear	20.0%	0.06%	30.0%	55.5%	28.6%	70.5%	27.3%	52.3%
anger	0	0	64.7%	71.4%	0	0	57.9%	65.2%
disgust	52.2%	40.0%	53.5%	58.8%	85.8%	92.4%	83.1%	89.7%
no-emotion	83.5%	80.9%	78.8%	80.6%	52.5%	58.0%	72.9%	75.4%

Table 5: EDC LIBLINEAR and RULEBASE f-score for each motion tags. We trained LIBLINEAR on SBHM train corpus and evaluated the system on different genre and SBHM test sets. Emotion tags with f-score of "0" are low populated categories (i.e. from 0-4 data points in the corresponding set)

each dimension represent one emotion tag with a binary value: 1 indicates the presence of sub-sentential emotion clause tag and 0 otherwise. We train LIBLINEAR model with training set and tune our parameters on dev set. Evaluation is done on test set.

Rule-base (RULEBASE): for sentence-level classification we chose one of the clause tags as sentence emotion tag. Our rule is as follow: if there is a match between one of the clause tags and sentence tag our algorithm picks that clause tag as sentence tag, if there is no match, one of the tags are selected randomly. We used SBHM training set to define this rule. This method is evaluated on SBHM test set.

Table 5 shows the results. We can observe the impact of subordinate clauses (scla) feature in supervised setting. This feature increases the system accuracy and f-score by 4.1%. Rule-based model creates f-score of 80.4% and the best results for sentence-level classification. These two results indicate the significance of clause-level annotation in sentence-level classification. Further, clause-level supervised system has a great improvement compare to baseline. **Comparisons to other systems** - we

Features	Clause		Sentence	
	acc.%	f-score%	acc.%	f-score%
Baseline (presence of emotion words)	46.2%	45.3%	47.3%	46.2%
LIBLINEAR	**71.3%**	**70.9%**	72.2%	71.3%
LIBLINEAR+scla	-	-	76.4%	75.7
RULEBASE	-	-	**81.1%**	**80.4%**

Table 6: EDC LIBLINEAR results using different combination of features on both clause and sentence levels and RULEBASE using rule-base algorithm.

compare EDC and RULEBASE on PL8 and *no-emotion* with previously reported results on two sets, i.e. NLH and BLG. However, we only compare our results with systems, which reported their results on EK6. This comparison is on sentence-level, since, clause-level emotion system is initiated in this work. Table 7 shows the comparison of LIBLINEAR and RULEBASE with other reported systems: Aman (Aman and Szpakowicz, 2007), SEMEVAL 2007 (Strapparava and Rada, 2007), Ghazi (Ghazi et al., 2010) [5], Mohammad (Mohammad, 2012), Özbal (Özbal and Pighin, 2013). We observe that RULEBASE outperforms other results for BLG, and both of our systems outperform other results for NLH. This indicates a) clause-level annotation improves sentence-level classification; b) PL8 is a better reflection for both NLH and BLG sets.

Method	Corpus	
	NLH	BLG
	acc.%	acc.%
Aman	-	73.8%
SEMEVAL 2007	17.5%	-
Ghazi	57.4%	61.6%
Mohammad	52.4%	31.4%
Özbal	20.7%	43.6%
LIBLINEAR	**74.5%**	66.1%
RULEBASE	69.7%	**75.5%**

Table 7: Comparing EDC: LIBLINEAR and RULEBASE results with previously reported results on two NLH and BLG sets. EDC and RULEBASE results are on PL8 and *no-emotion*. SEMEVAL 2007 reported results only on NLH, Aman collected BLG and reported their results only on BLG.

5. Conclusion and Future Direction

Unified annotation and combination of different genre datasets can improve and generalize emotion detection in sentences. We demonstrated that PL8 emotion tags represent these dataset better than EK6 emotion tags and if we aim to expand the emotion tagset to more fine-grain, PL8 annotation enables us to fulfill this aim. Our results showed clause-level feature can improve the prediction of emotion in sentence-level. We provide an automated system for clause-level emotion detection and classification. Further, we annotated emotions in clause-level. In future, our aim is to create sophisticated Deep Neural Network models for sentence-level classification, leveraging clause-level emotion tags. We aim to build systems that can tag smaller piece of text (i.e. phrases, clauses, words) automatically. And, we intend to add different genres to our corpus, mainly our aim is to add genres with different syntax from the current collections.

6. Acknowledgements

This study is supported by DARPA LORELEI Grant. Ali Seyfi and Sardar Hamidian carried out annotation procedure at pilot stage.

[5] They reported two different results, one is flat classification and the other is hierarchical classification. Flat classification is comparable to EDC.

7. Bibliographical References

Abdul-Mageed, M. and Lyle, U. (2017). Emonet: Fine-grained emotion detection with gated recurrent neural networks. *ACL*.

Aman, S. and Szpakowicz, S. (2007). Identifying expressions of emotion in text. *10th International Conference on Text, Speech, and Dialogue*.

Das, D. and Bandyopadhyay, S. (2010). Identifying emotional expressions, intensities and sentence level emotion tags using a supervised framework. In *PACLIC*, volume 24, pages 95–104.

Dodds, P. and Danforth, C. (2010). Measuring the happiness of large-scale written expression: Songs, blogs, and presidents. *Journal of Happiness Studies*, 11.4:441–456.

Ekman, P. (1992). An argument for basic emotions. *Cognition and emotion*, 6.3-4:169–200.

Ghazi, D., Inkpen, D., and Szpakowicz, S. (2010). Hierarchical versus flat classification of emotions in text. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 140–146. Association for Computational Linguistics.

Kim, S. M., Valitutti, A., and Calvo, R. A. (2010). Evaluation of unsupervised emotion models to textual affect recognition. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 62–70. Association for Computational Linguistics.

Klein, D. and Manning, C. (2003). Proceedings of the 41st annual meeting on association for computational linguistics. *Association for Computational Linguistics*, Volume 1.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Mishne, G. (2005). Experiments with mood classification in blog posts. *ACM SIGIR*.

Mohammad, S. (2012). Portable features for classifying emotional text. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 587–591. Association for Computational Linguistics.

Öhman, A. (1999). Distinguishing unconscious from conscious emotional processes: Methodological considerations and theoretical implications. *Handbook of cognition and emotion*, pages 321–352.

Özbal, G. and Pighin, D. (2013). Evaluating the impact of syntax and semantics on emotion recognition from text. In *Computational Linguistics and Intelligent Text Processing*, pages 161–173. Springer.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *43rd Annual Meeting on Association for Computational Linguistics*.

Plutchik, R. (1962). The emotions: Facts, theories, and a new model. *New York: Random House*.

Strapparava, C. and Rada, M. (2007). Semeval-2007 task 14: Affective text. *4th International Workshop on Semantic Evaluations. Association for Computational Linguistics*.

Vo, B. and Collier, N. (2013). Twitter emotion analysis in earthquake situations. *International Journal of Computational Linguistics and Applications*, 4.1:159–173.

Yan, J. (2014). Expanding the range of automatic emotion detection in microblogging text. *EACL*.

A Swedish Cookie-Theft Corpus

Dimitrios Kokkinakis[1], Kristina Lundholm Fors[1], Kathleen Fraser[1], Arto Nordlund[2†]
Department of Swedish[1]; Department of Psychiatry and Neurochemistry[2]
University of Gothenburg, Sweden
{dimitrios.kokkinakis; kristina.lundholm; kathleen.fraser}@svenska.gu.se
{arto.nordlund}@neuro.gu.se

Abstract

Language disturbances can be a diagnostic marker for neurodegenerative diseases, such as Alzheimer's disease, at earlier stages. Connected speech analysis provides a non-invasive and easy-to-assess measure for determining aspects of the severity of language impairment. In this paper we focus on the development of a new corpus consisting of audio recordings of picture descriptions (including transcriptions) of the Cookie-theft, produced by Swedish speakers. The speech elicitation procedure provides an established method of obtaining highly constrained samples of connected speech that can allow us to study the intricate interactions between various linguistic levels and cognition. We chose the Cookie-theft picture since it's a standardized test that has been used in various studies in the past, and therefore comparisons can be made based on previous research. This type of picture description task *might* be useful for detecting subtle language deficits in patients with subjective and mild cognitive impairment. The resulting corpus is a new, rich and multi-faceted resource for the investigation of linguistic characteristics of connected speech and a unique dataset that provides a rich resource for (future) research and experimentation in many areas, and of language impairment in particular. The information in the corpus can also be combined and correlated with other collected data about the speakers, such as neuropsychological tests, brain physiology and cerebrospinal fluid markers as well as imaging.

Keywords: Speech elicitation corpus, Cookie-theft stimuli, dementia, linguistic features, language impairment

1. Introduction and Background

Elicitation of spontaneous speech and narrative discourse samples from (young and/or adult) individuals both healthy as well as with a variety of mental and cognitive impairments, at various stages of the pathology and with a variety of stimuli, have been developed and studied for several decades (e.g., Bottenberg et al., 1987; Bryant et al., 2016). Such systematic form of language production can be a valuable procedure for describing and even quantifying the degree and severity of e.g. cognitive decline, which most often[1] incorporate a single picture description, a picture sequence description (e.g., the *Dog Story*; de Lira et al., 2011) or a personal opinion on an event or idea. Even work on normative data has been reported for use in various narrative discourse tasks. For instance, based on a large control sample, Richardson & Dalton (2015) created main concept checklists (information content units) for three semi-spontaneous discourse tasks, a picture sequence narrative *Broken Window*, a storytelling *Cinderella*, and a procedure *Peanut Butter and Jelly*. For a similar purpose, Catricalà et al. (2017) created their own original picture *Summer Time* used as normative data on 134 Italian subjects pooled across homogeneous subgroups for age, sex, and education.

Nonetheless, the single black and white picture of the *Cookie-theft* from the Boston Diagnostic Aphasia Examination Battery (BDAE; Goodglass et al., 2001) is one of *the* most used stimuli for the assessment of language production. Other images, such as the *picnic* picture (Weissenbacher et al., 2016) from the Western Aphasia Battery (Kertesz, 1982) or the tales of *Cinderella, Snow White, the Seven Dwarfs* and the *Little Red Riding Hood* (Machado Lima et al., 2014; Silveira & Mansur, 2015) have also been used in various studies but to a much lesser extent; for an extended review see Bryant et al. (2016) and Boschi et al. (2017).

In this paper we describe a Swedish Cookie-theft corpus developed in the "Linguistic and extra-linguistic parameters for early detection of cognitive impairment" project. The project aims to adapt, develop and test methods that in isolation have shown promising outcomes on tasks related to early detection of dementia, differentiating between various dementia types and controls and explore the cognitive processes that underlie certain forms of spoken language production.

2. The Cookie-Theft Stimulus

The single structured stimuli of the Cookie-theft picture depicts a familiar household event scene in a kitchen which includes a woman drying some dishes, unconcerned by the overflowing of water from the sink, and the notable feature of a child (boy) standing on a stool about to tip over, stealing cookies (thus the name of the picture) off a high shelf. The stimuli is considered an ecologically valid approximation to spontaneous discourse (Gilles et al., 1996) and have been widely used to elicitate speech from speakers with different types of language and communication disorders, including numerous discourse tasks for the study of various types of dementia. In contrast to spontaneous speech the Cookie-theft allows control for content and context. Moreover, the figure is often used

† Deceased November 2017.

[1] Other spoken language elicitation procedures include the Wechsler's Logical Memory (Wechsler, 1997), an immediate and delayed story recall test.

because of the clear, straightforward drawings that reduce ambiguity and lessen the effect of memory problems. The picture allows the variation between speech-styles to be reduced, and also "minimizes confounds in analysis due to the controlled nature of the speech content" (Williams et al., 2010).

2.1 Previous Work Based on the Cookie-Theft

Narratives based on the Cookie-theft picture have been a source of knowledge for clinical and experimental research worldwide which also enables potential cross-linguistic comparisons. The Cookie-theft stimuli has been used to elicit written narratives, and this can be especially useful when evaluating people with Alzheimer's, as written language has been shown to be impaired even at the early stages of the disease. E.g., Forbes et al. (2004) reported significantly worse written discourse production at the mild and moderate stages of AD compared to healthy controls; while Kavé & Goral (2016) confirmed that individuals with Alzheimer's produce a lower proportion of nouns out of all other words as well as a higher proportion of pronouns out of all other words. Although various datasets based on the Cookie-theft picture are written ones (Croisile et al., 1996 for French and Pekkala et al., 2013 for English); other combinations exist, such as both written and verbal (Groves-Wright et al., 2004); as well as audio recordings, with or without transcriptions which have been collected and described for several languages (e.g., English, Ripich et al., 2000; Swedish, Tyche, 2001; German, Bschor et al. (2001); Spanish, Cuetos et al., 2007; Japanese, Choi, 2009; Norwegian, Lind et al., 2009 and Hebrew, Kavé & Dassa, 2017). Moreover, the Cookie-theft picture has been analysed at various linguistic levels (Kavé & Levy, 2003; de Lira et al., 2014; Yancheva & Rudzicz, 2016) and for different purposes (e.g., aphasia examination, Gilles et al., 1996; Ash et al., 2013; acquired brain injuries, Hux et al., 2008; and normative data collection, Forbes-McKay & Venneri, 2005).

2.2 Cookie-Theft and Information Content Units

Information content units, ICUs, refers generally to various predefined image concepts/objects/facts and actions describing a picture. With respect to the Cookie-theft picture content there exist several inventories that try to capture all possible semantic or information content units, containing between 7-25 ICUs. Mackenzie et al. (2007) uses seven such ICUs ('woman doing dishes', 'sink overflowing', 'boy on stool', 'children stealing cookies', 'girl reaching for cookie', 'stool falling', 'woman not noticing') while Kavé & Levy (2013) proposed 25[2], divided into four semantic groups; namely: *subjects* ('boy', 'girl', 'woman'), *places* ('kitchen', 'exterior seen through the window'), *objects* ('cabinet', 'cookies', 'counter', 'curtain', 'dishes on the counter', 'faucet', 'floor', 'jar', 'plate', 'sink', 'stool', 'water', 'window') and *actions* ('boy taking the cookie', 'boy or stool falling', 'woman drying or washing dishes/plate', 'water overflowing or spilling', 'the girl asking for a cookie', 'woman unconcerned by the overflowing', 'woman indifferent to the children'). If used, synonymous expressions were accepted for the same concept (e.g. 'mammy' and 'mother' and also 'woman'); even the wrong word referring to a piece of information is accepted (e.g., 'ladder' instead of 'stool'; 'shelf' instead of 'cabinet' or 'cupboard')

Moreover, these information units can be identified both with a top down approach (Croisile et al., 1996) or a bottom up one, e.g. using clustering word embeddings of the naming expressions (Yancheva & Rudzicz, 2016). Usually, subjects are given credit for mentioning the presence of a given ICU and thus this set of ICUs enables easy comparison across subjects and easy scoring of the predefined contents of the picture (Boschi et al., 2017). Sirts et al. (2017) applies the notion of semantic idea density[3] (SID) on the same image, in which each SID is a count of the pre-defined information content units and is computed by counting the number of ICUs mentioned in the text and then normalising them by the total number of word tokens.

2.3 Cookie-Theft Datasets

The most extensive dataset of the picture description publicly available is the DementiaBank[4] corpus, a part of the TalkBank project, collected between 1983-88; (Becker et al., 1994; MacWhinney et al., 2011). DementiaBank is a clinical dataset which consists of interview recordings and transcripts of English-speaking people with AD and a few with mild cognitive impairment (MCI), describing the Cookie-theft picture. This dataset has been used in various studies in the past, e.g. Fraser et al. (2016) achieved classification accuracies of over 81% in distinguishing between participants with AD and healthy controls by computing a number of linguistic variables from the transcripts, and acoustic variables from the associated audio files; Orimaye et al. (2017) used DementiaBank to learn syntactic, lexical, and n-gram linguistic biomarkers to distinguish the group of probable AD from the healthy group. Their best diagnostic model significantly distinguished the two groups using Support Vector Machines; while Masrani et al. (2017) used the DementiaBank data for domain adaptation and discourse features in an attempt to improve classification accuracy The Cambridge Cookie-theft corpus (Williams et al., 2010)

[2] The same 4 key categories, but with less ICU content (23), were identified by Croisile et al. (2016); 'dishcloth' was an additional object used by Croisile et al., while 'floor', 'counter' and 'faucet' were three objects used by Kavé & Levy (2013) but not used by Croisile et al.

[3] A complementary notion to SID is propositional idea density or PID (i.e. the number of any propositions or ideas expressed in the text, that is, the amount of information that is conveyed relative to the number of words used to encode it; Chand et al, 2010) which according to Sirts et al. (2017), is more suitable on datasets of spontaneous speech on free topics rather than standardised picture descriptions.

[4] DementiaBank: <dementia.talkbank.org/> Visited 20180102.

is another corpus consisting of samples by healthy individuals without brain-damage and patients recorded between the mid-1990s and 2009. The OPTIMA[5] (Oxford Project to Investigate Memory and Aging) transcripts of connected speech has been also used in several studies. OPTIMA is a longitudinal study of aging and dementia in a cohort of elderly, community living volunteers collected between 1998-2008. For instance, Rentoumi et al. (2014) used the OPTIMA Cookie-theft descriptions in machine learning text classification experiments, in order to assign the samples to one of two pathological groups (mixed vascular and AD and pure AD) on the basis of lexical and syntactic features as well as statistical and information theory characteristics, with high classification accuracy (75%). The Framingham Heart Study[6] is a longitudinal population-based study established in 1948, a number of dementia-free sub-cohorts have been followed since the mid-1970s for development of incident dementia (Seshadri et al., 2006). A neuropsychological battery administered three times to these participants included a written description of the Cookie-theft picture. The purpose of the Framingham Heart Study was to find an early language marker of AD by comparing lexical retrieval on these tests in participants with clear cases of AD compared to that of participants who did not. Finally, the WRAP (Wisconsin Registry for Alzheimer's Prevention) study sample (Johnson et al., 2017), with recordings and transcriptions of 264 participants (200 cognitively healthy and 64 with early MCI), was used for determining if participants with very early, subclinical memory declines were also showing declines in connected language. Based on these data, Mueller et al. (2017) showed that participants with early MCI status declined faster in features of speech fluency and semantic content than those who were cognitively stable.

2.4 The Swedish Cookie-Theft Corpus

The Swedish Cookie-theft corpus[7] described in this paper consists of data from three subject groups: those with subjective cognitive impairment (SCI), those with mild cognitive impairment (MCI), and healthy controls (HC). All participants were recruited from a longitudinal study in the region of West Sweden (see Section 3). Note that SCI, MCI, and AD are on a spectrum of disease progression. SCI is a common diagnosis in elderly people, sometimes suggested to be associated with e.g. depression or anxiety, but also a risk factor for dementia (Jessen et al., 2010). MCI is a well-defined prodromal state of dementia and appears to represent a transition between normal aging and early dementia (Ritchie & Touchon, 2010), in which a person has minor problems with cognition (e.g., problems with memory or thinking) but these are not severe enough to warrant a diagnosis of dementia or interfere significantly with daily life. Still, persons with MCI have difficulties which are worse than would normally be expected for a healthy person of their age. In clinical settings patients with MCI show rates of conversion to dementia of about 12% per year (Petersen et al., 2001).

3. The Gothenburg MCI-study and Related Ethical Issues

The Gothenburg mild cognitive impairment study (Nordlund et al., 2005; Wallin et al., 2016) conducts longitudinal in-depth phenotyping of patients with a wide range of cognitive impairment (i.e., from very mild to manifest dementia including, but also including cognitively normal controls) using neuropsychological, neuroimaging, and neurochemical tools. The study is clinically based and aims at identifying neurodegenerative, vascular and stress related disorders prior to the development of dementia. All patients in the study undergo baseline investigations, such as neurological examination, psychiatric evaluation, cognitive screening, magnetic resonance imaging of the brain and cerebrospinal fluid collection. At biannual follow-ups, most of these investigations are repeated. The Gothenburg MCI-study is approved by the local ethical committee review board (ref. L091–99, 1999; T479-11, 2011); while the currently described study is approved by the local ethical committee decision 206-16, 2016. The project aims at gathering a rather homogeneous group of participants with respect to age and education level. Written informed consent is obtained from all participants in the study while the exclusion and inclusion criteria are specified according to a predefined protocol.

4. Population and Socio-Demographic Characteristics

All subjects in the three different groups (HC, SCI and MCI), were native speakers of Swedish with adequate hearing and vision, no documented history of neurological or psychiatric illness, learning disabilities or e.g. alcoholism according to the inclusion and exclusion criteria pre-specified, aged 53-79 years old, while 53 of them (58%) were females. Demographic details and MMSE scores for all participants are provided in Table 1.

MMSE, the Mini Mental State Examination, is a paper-based questionnaire and a screening test used to evaluate cognitive function, where 30 points is the maximum, and 24 or above indicates normal cognition (Folstein et al., 1975). The difference in age between the groups is not statistically significant, as determined by a one-way ANOVA ($F(2, 88) = 2.082$, $p = .131$). When comparing number of years of education, a one-way ANOVA shows

[5] OPTIMA: <www.ndcn.ox.ac.uk/research/centre-prevention-stroke-dementia/resources/optima-oxford-project-to-investigate-memory-and-ageing> Visited 20180102.

[6] Framingham Heart Study: <www.framinghamheartstudy.org> Visited 20180102.

[7] We are aware of two more Swedish Cookie-theft collections. Tyche (2001) collected oral descriptions from 12 MCI and 12 healthy controls; while Cromnow & Landberg (2009) collected written samples from 96 healthy, Swedish controls, having Swedish as a mother tongue, absence of clinical manifestations of linguistic impairment and absence of neurological disease. This constitutes normative data from speakers without cognitive impairments and which its purpose was to establish norms.

that there is a significant difference between the groups ($F(2, 88) = 6.085$, $p = 0.003$). A post-hoc LSD test demonstrates that there is no significant difference between the HC and the MCI group, but that there is a significant difference between the HC and the SCI group ($p = 0.001$) and the MCI and the SCI group ($p = 0.02$). The MMSE score differs significantly between the three groups, as determined by a one-way ANOVA ($F(2, 88) = 16.978$, $p < 0.0001$). Post-hoc LSD test show that subjects in the MCI group have significantly lower MMSE scores than the HC group ($p < 0.0001$) and the SCI group ($p < 0.0001$), whereas there is no difference between the SCI group and the HC group. To summarise, there is no difference in age between the groups, but the subjects in the SCI group have a longer education than the MCI and the HC group. Finally, the subjects in the MCI group score lower on the MMSE than the subjects in the HC group and the SCI group.

	HC (n=36)	SCI (n=23)	MCI (n=32)
Mean Age_{years}	67.9 (sd 7.2)	66.3 (sd 6.9)	69.9 (sd 5.7)
Mean Edu_{years}	13.2 (sd 3.4)	16.1 (sd 2.1)	14.1 (sd 3.5)
Female	23	14	16
Male	13	9	16
MMSE	29.6 (sd 0.61)	29.5 (sd 0.90)	28.2 (sd 1.41)

Table 1: Demographic information for the three groups of participants. The MiniMental State Exam (MMSE) is a test of cognitive status and has a maximum score of 30.

The population was selected from an ongoing epidemiological study in the region, see Section 3. At the start of the study, each participant was given information about the purpose of the evaluation, and written informed consent was obtained in accordance with the protocol approved by the local ethics committee. Note that the mean result on the MMSE is above 24 for all groups (a cutoff of ≤ 24 has been proposed for cognitive impairment, with a score between 25-27 indicating possible cognitive impairment which should be further evaluated) which means that all test within the range of normal cognition; both the MCI and SCI diagnosis are indicative of subtle cognitive impairment and it is expected that the subjects perform relatively well on tests such as the MMSE.

5. Data Collection

During the presentation of the Cookie-theft stimuli the subjects are asked to describe everything that can be observed in the picture, and the narrative is recorded. For the task, the original label on the image's cookie jar is translated and substituted from the English "COOKIE JAR" to the Swedish label "KAKBURK". The samples are recorded in an isolated environment and the whole task is designed to evoke a monologue by the participant. The instruction given to the subject was: "Tell me everything you see going on in this picture, describe objects and events. You can go on as long as you prefer and you will be not interrupted until you indicate that you do not have anything more to say". Descriptions were recorded and the recordings were terminated when the subject indicated verbally or non-verbally (by becoming silent for a longer period of time) that the narrative was completed.

	HC (n=36)	SCI (n=23)	MCI (n=31)
Shortest description	29s	41s	21s
Longest description	274s	635s	361s
Mean duration	92.3s	123s*	102.3s
Total duration	3324s	2829s	3173s

Table 2: Duration of the Cookie-theft samples (*the mean duration for the SCIs is 99.7 seconds if we exclude one extreme outlier description of 635 seconds)

For the audio capture of the narratives we used a H2n Handy recorder and the resulting audio files were saved and stored as uncompressed audio in .wav 44.1 kHz with 16-bit resolution. The recording is carried out in an isolated environment at the University lab in order to avoid noise. A speech pathologist and computational linguist were present during the recording sessions, providing all subjects with identical instructions according to a pre-defined protocol. Information about the length of the spontaneous speech samples is provided in Table 2.

6. Transcriptions and Linguistic Analysis

Based on the digitized speech waveform of the Cookie-theft audio recordings, manual transcriptions are produced, that are semi-automatically aligned with the audio signal. During speech transcription, special attention is also paid to non-speech acoustic events including speech disfluencies consisting of filled pauses a.k.a. hesitations (such as "um" and "uh"), false-starts, word fragments, repetitions as well as other features, particularly non-verbal vocalizations such as laughing, sniffing and coughing. A basic transcription manual, with the various conventions to be used, is produced which helped the human transcribers accomplish a homogeneous transcription. For instance, all numerals should be written out as complete words, while symbols, such as XML-tags, are used for the encoding of pauses or transcriber's comments. Furthermore, for the transcription the Praat application (Boersma & Weenink, 2013) is utilized; using a 2-tier text grid configuration, one for broad orthographic transcription (standardized spelling with manual utterance segmentation) and one for narrow orthographic transcription, with maintained spoken language phenomena, such as partial words, see Figure 1.

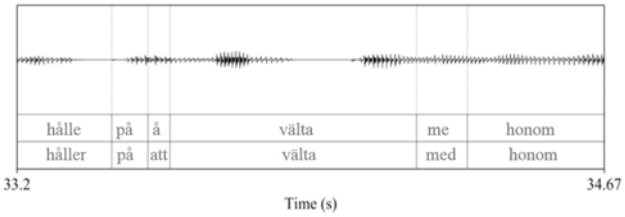


Figure 1: Example of transcription of the collected data.

7. Linguistic Processing, an Overview

In total, there are 19285 tokens in the sample (HC: 8770; MCI: 5724; SCI: 4791) and 1573 word types (HC: 1077; MCI: 750; SCI: 734). The transcriptions have been processed with basic Natural Language Processing (NLP) tools for Swedish which includes tools for lexical analysis (lemmatization and multiword expression identification), part-of-speech annotation, syntactic parsing (both dependency and constituent-based) and identification of information content units. Computational models that can be used to differentiate between the groups are typically based on various lexical, syntactic, semantic, and discourse characteristics. Here we provide a very general overview of the various features we currently extract from the corpus, both the audio signal (i.e. acoustic and prosodic) and the transcriptions (i.e. features that follow the lexicon-syntax-semantics continuum). Results of our linguistic analysis are reported elsewhere (e.g. Lundholm Fors et al., 2018; Fraser et al., 2018).

From the transcriptions we extract: (i) lexical distribution measures (such as type-token ratio, mean word length, long word counts but also the distribution of lexical and non-lexical fillers or disfluency markers such as "um", "uh", "eh") and out-of-vocabulary rate (Pakhomov et al., 2010); (ii) syntactic complexity markers (such as [context free] production rules; dependency distance; noun phrase average length and noun phrase density; (iii) psycholinguistic measures (such as familiarity); (iv) imformation units (such as content density). Measures that can give an indication of semantic deficits are important and likely to be present even in the early stages of MCI. A large number of acoustic features (such as speech rate, pause frequency, filled pauses, total pause duration; pitch and formants) have been proposed as relevant in the literature (Roark et al., 2011; Ahmed et al., 2013; Yancheva et al., 2015) and it pinpoints the importance of distinguishing between vocal changes that occur with normal aging and those that are associated with MCI (and SCI).

8. Conclusions and Future Work

This paper presented a new Swedish corpus of semi-spontaneous oral text production (connected speech) of the Cookie-theft picture by speakers with subjective and mild cognitive impairment and healthy age-matched controls. The main purpose of the corpus described in this paper is to use it for the identification of subtle language deficits in spontaneous speech, and to identify the features that differentiate the groups from each other. The task only takes a few minutes to administer, and poses a very limited demand on memory.

The described data provides a rich resource for future research in many areas of language impairment and has been constructed to facilitate analysis with natural language processing and corpus linguistics techniques since it can be analysed at all levels of linguistic structure, i.e., phonetic, lexical, syntactic, semantic and pragmatic. Whether the chosen picture description task will be suitable for detecting subtle language deficits in patients with subjective and mild cognitive impairment, is a question we are currently investigating. We anticipate that a single image might limit our future findings, however we believe that the breadth of analysis as outlined above and the combination of features of both the audio signal (e.g., prosody) and text (e.g., information content units) could provide a solid ground for analysis and experimentation. Moreover, we also intend to relate the performance of linguistic variables to available results of the neurodegenerative testing and other cognitive screening tools (e.g. MMSE) for the same population.

In parallel with the analysis of the available data, we also plan to extend this research by repeating the Cookie-theft experiment (and recordings) in the near future, that is in 18 months after the first recordings took place, and described in this paper. This might be useful for longitudinal assessment and comparison of the semi-spontaneous oral text production and for the evaluation of possible degradation, maintenance or recovery of skills. Moreover, we have also concrete ideas on how to complement the corpus with other types of oral tests we intend to record, such as semantic word fluency (category animals) and also spontaneous speech production/semi-structured conversation in which the participants will be encouraged to talk about a predefined task-oriented dialogue topic such as a map route (*cf.* Andersson et al., 1991). These additions are *already* approved by the local ethical committee review board (ref. T021-18). Finally, we also intend to acquire written descriptions of the Cookie-theft from previous Swedish studies. This will enable us to have a more balanced corpus of the Cookie-theft.

9. Acknowledgements

This work has received support from *Riksbankens Jubileumsfond* - The Swedish Foundation for Humanities and Social Sciences, through the grant agreement no: NHS 14-1761:1.

10. Bibliographical References

Ahmed, S., Haigh, A-M., de Jager, C.A., and Garrard, P. (2013). Connected speech as a marker of disease progression in autopsy-proven Alzheimer's disease. *Brain*. 136 (12): 3727-3737.

Anderson, A. et al. (1991). The HCRC Map Task Corpus. *Language and Speech*, 34, pp. 351-366.

Ash, S., Evans, E., O'Shea, J., Powers, J., Boller, A., Weinberg, D., Haley, J., McMillan, C., Irwin, DJ., Rascovsky, K., and Grossman, M. (2013). Differentiating primary progressive aphasias in a brief sample of connected speech. *Neurology*. 23;81(4):329-36.

Becker, J.T., Boiler, F., Lopez, O.L., Saxton, J. and McGonigle, K.L.. (1994). The natural history of Alzheimer's disease. Description of study cohort and accuracy of diagnosis. *Arch Neurol*. 51(6):585-94.

Boersma, P., and Weenink, D. (2013). *Praat: doing phonetics by computer* [Computer program]. Version

6.0.19, retrieved in Aug. 2016 from <http://www.praat.org/>.

Boschi, V., Catricalà, E., Consonni, M., Chesi, C., Moro, A. and Cappa SF. (2017). Connected Speech in Neurodegenerative Language Disorders: A Review. *Front. Psychol.* 8:269. doi: 10.3389/fpsyg.2017.00269

Bottenberg, D., Lemme, M. and Hedberg, N. (1987). Effect of Story Content on Narrative Discourse of Aphasic Adults. *J Clinical Aphasiology*. 17. Pp. 202-209.

Bryant, L., Ferguson, A. and Spencer, E. (2016). Linguistic analysis of discourse in aphasia: A review of the literature. *Clinical Linguistics & Phonetics*, 30:7, 489-518. DOI: 10.3109/02699206.2016.1145740

Bschor, T., Kühl, K.P., and Reischies, F.M. (2001). Spontaneous Speech of Patients with Dementia of the Alzheimer Type and Mild Cognitive Impairment. *Int Psychogeriatr.* Vol 13:3, pp. 289-298.

Catricalà, E. et al. (2017). SAND: a Screening for Aphasia in NeuroDegeneration. Development and normative data. *Neurol Sci*. 38:1469–1483. DOI 10.1007/s10072-017-3001-y

Chand, V., Baynes, K., Bonnici, L.M. and Tomaszewski Farias, S. (2010). *Analysis of Idea Density (AID): A Manual.* Technical report, U of California at Davis. Available from: <https://www.researchgate.net/publication/267362822_Analysis_of_Idea_Density_AID_A_Manual>

Choi, H. (2009). Performances in a Picture Description Task in Japanese Patients with Alzheimer's Disease and with Mild Cognitive Impairment. *Korean Academy of Speech-Language Pathology & Audiology*. Pp. 326-337.

Croisile, B., Ska, B., Brabant, M-J., Duchene, A., Lepage, Y., Aimard, G. and Trillet, M. (1996). Comparative Study of Oral and Written Picture Description in Patients with Alzheimer's Disease. *Brain Lang.* 53(1):1-19.

Cromnow, K., and Landberg, T. (2009). *Skriftliga beskrivningar av bilden "Kakstölden". Insamling av referensvärden från friska försökspersoner*. Master's thesis. Division of Speech and Language Pathology, Karolinska institute, Sweden. (In Swedish).

Cuetos, F., Arango-Lasprilla, J.A., Uribe, C., Valencia, C. and Lopera, F. (2007). Linguistic changes in verbal expression: A preclinical marker of Alzheimer's disease. *J of the International Neuropsychological Society*. 13, 433–439. Cambridge University Press.

Folstein, M. F., Folstein, S. E., and McHugh, P. R. (1975). "Mini-mental state": a practical method for grading the cognitive state of patients for the clinician. *Journal of psychiatric research*, *12*(3), 189-198.

Forbes, KE., Shanks, MF., and Venneri, A. (2004). The evolution of dygraphia in Alzheimer's disease. *Brain Research Bulletin.* 2004;63:19–2

Forbes-McKay, KE. and Venneri, A. (2005). Detecting subtle spontaneous language decline in early Alzheimer's disease with a picture description task. *Neurol. Sci*. 26, 243–254. 10.1007/s10072-005-0467-9

Fraser, KC., Meltzer, JA., and Rudzicz, F. (2016). Linguistic Features Identify Alzheimer's Disease in Narrative Speech. *J Alzheimers Dis.* 49(2):407-22. doi: 10.3233/JAD-150520.

Fraser, KC., Lundholm Fors, K, Eckerström, M., Themistokleous, C., and Kokkinakis, D. (2018). *Improving the sensitivity and specificity of MCI screening with linguistic information*. 2nd Resources and ProcessIng of linguistic, para-linguistic and extra-linguistic Data from people with various forms of cognitive/psychiatric impairments Workshop (RaPID-2, an LREC workshop). Miyazaki, Japan.

Gilles, E., Patterson, K., and Hodges, J. (1996). Performance on the Boston Cookie-theft picture description task in patients with early dementia of the Alzheimer's type: missing information. *Aphasiology*. 10:4:395-408.

Goodglass, H., Kaplan, E., and Barresi, B. (2001). *Boston Diagnostic Aphasia Examination* (3rd ed.). Philadelphia: Lippencott, Williams & Wilkins.

Groves-Wright, K., Neils-Strunjas, J., Burnett, R. and O'Neill, J.M. (2004). A comparison of verbal and written language in Alzheimer's disease. *J of Communication Disorders.* 37:2,, Pages 109-130

Hux, K., Wallace, S.E., Evans, K., and Snell, J. (2008). Performing Cookie-theft Picture Content Analyses to Delineate Cognitive Communication Impairments. *J of Medical Speech-Language Path*. Vol. 16:2. Pp. 83-99.

Jessen F. et al. (2010). Prediction of dementia by subjective memory impairment: effects of severity and temporal association with cognitive impairment. *Arch. Gen. Psychiatry*. 67(4). Pp. 414–422.

Johnson, S. C., et al. (2017). The Wisconsin Registry for Alzheimer's Prevention: a review of findings and current directions. *Alzheimers Dement (Amst).* 10:130-142. doi: 10.1016/j.dadm.2017.11.007.

Kavé, G. and Levy, Y. (2003). Morphology in Picture Descriptions Provided by Persons with Alzheimer's Disease. *J of Speech, Language, and Hearing Research.* Vol. 46, 341-352. doi:10.1044/1092-4388(2003/027)

Kavé, G. and Dassa, A. (2017). Severity of Alzheimer's disease and language features in picture descriptions. *Aphasiology*, DOI: 10.1080/02687038.2017.1303441

Kavé, G. and Goral, M. (2016). Word retrieval in picture descriptions produced by individuals with Alzheimer's disease. *J of Clinical and Experimental Neuropsychology*, 38:9, 958-966.

Kertesz, A. (1982). *Western Aphasia Battery*. NY Grune and Stratton.

Lind, M., Kristoffersen, K E, Moen, I., and Simonsen, HG. (2009). Semi-spontaneous oral text production: Measurements in clinical practice. *J of Clinical Ling & Phonetics*. 23:12, pp. 872-886. Taylor & Francis.

de Lira, J.O. et al. (2011). Microlinguistic aspects of the oral narrative in patients with Alzheimer's disease. *International Psychogeriatrics* Apr;23(3):404-12. doi: 10.1017/S1041610210001092.

de Lira, J.O., Minett, TSC., Bertolucci, PHF., and Ortiz, K.Z. (2014). Analysis of word number and content in discourse of patients with mild to moderate Alzheimer's disease. *Dement Neuropsychol.* Sept;8(3):260-265.

Lundholm Fors, K, Fraser, KC, and Kokkinakis, D. (2018). *Automated Syntactic Analysis of Language Abilities in Persons with Mild and Subjective Cognitive Impairment*. Proceedings of the 29th Medical Informatics Europe (MIE). Gothenburg, Sweden.

Machado Lima, T., Brandão, L., Pimenta Parente, M-A., and Peña-Casanova, J. (2014). Alzheimer's disease: cognition and picture-based narrative discourse. *CEFAC* vol.16:4 São Paulo. <http://dx.doi.org/10.1590/1982-021620147513>.

Mackenzie, C., Brady, M., Norrie, J., and Poedjianto, N. (2007). Picture description in neurologically normal

adults: Concepts and topic coherence. *Aphasiology*. Vol 21:3-4, pp. 340-354.

MacWhinney, B., Fromm, D., Forbes, M. P., and Holland, A. (2011). AphasiaBank: Methods for studying discourse. *Aphasiology*. 25 (11), 1286-1307.

Masrani V., Murray G., Field T.S. and Carenini G. (2017) Domain Adaptation for Detecting Mild Cognitive Impairment. In: Mouhoub M. and Langlais P. (eds) *Advances in Artificial Intelligence. AI 2017*. LNCS. Vol 10233. Springer.

Mueller, K., Koscik R., Hermann B., Johnson S. and Turkstra, L. (2017). Declines in Connected Language Are Associated with Very Early Mild Cognitive Impairment: Results from the Wisconsin Registry for Alzheimer's Prevention. *Frontiers in Aging Neuroscience*. 9: 437, pp. 1-14. Frontiers Media.

Nordlund, A. et al. (2005). The Goteborg MCI study: mild cognitive impairment is a heterogeneous condition. *J Neurol Neurosurg Psychiatry.* 76(11):1485-90.

Orimaye, S.O., Wong, JSM., Golden, K.J., Wong, C.P., and Soyiri, I.N. (2017). Predicting probable Alzheimer's disease using linguistic deficits and biomarkers. *BMC Bioinformatics*. 18:34

Pakhomov, SVS. et al. (2010). A computerized technique to assess language use patterns in patients with frontotemporal dementia. *J Neuroling.* 23(2):127–144.

Pekkala, S. et al. (2013). Lexical retrieval in discourse: An early indicator of Alzheimer's dementia. *Clinical Linguistics & Phonetics*. 27:12, 905-921.

Petersen, R.C. et al. (2001). Current concepts in mild cognitive impairment. *Arc of Neurology*, 58, 1985-1992

Rentoumi, V., Raoufian, L., Ahmed, S., de Jager, CA., and Garrard, P. (2014). Features and machine learning classification of connected speech samples from patients with autopsy proven Alzheimer's disease with and without additional vascular pathology. *J Alzheimer's Dis*. 42 Suppl 3:S3-17. doi: 10.3233/JAD-140555.

Ripich, D.N., Fritsch, T., Ziol, E. and Durand, E. (2000). Compensatory strategies in picture description across severity levels in Alzheimer's disease: A longitudinal study. *American J of Alzheimer's Disease*. Vol. 15:4.

Richardson, J.D., and Dalton, S.G. (2016). Main concepts for three different discourse tasks in a large non-clinical sample. *Aphasiology*. 30:1. Taylor and Francis.

Ritchie, K., and Touchon, J. (2010). Mild cognitive impairment: conceptual basis and current nosological status. *The Lancet*. Vol. 355:9199. Pp. 225–228. doi:10.1016/S0140-6736(99)06155-3.

Roark, B., Mitchell, M., Hosom, J-P., Hollingshead, K., and Kaye, J. (2011). *Spoken Language Derived Measures for Detecting Mild Cognitive Impairment*. IEEE Transactions of Audio Speech and Language Processing. 19(7): 2081–2090.

Seshadri, S., Beiser, A., Kelly-Hayes, M., Kase, CS., Au, R., Kannel, WB. and Wolf, PA. (2006). The lifetime risk of stroke: estimates from the Framingham Study. *Stroke*. 37(2):345-50.

Silveira, G., and Mansur, L. (2015). Analysis of prototypical narratives produced by aphasic individuals and cognitively healthy subjects. *Dement Neuropsychol*. 9(3):279-284.

Sirts, K., Piguet, O., and Johnson, M. (2017*). Idea density for predicting Alzheimer's disease from transcribed speech*. The 21st Conference on Computational Natural Language Learning (CoNLL), pages 322–332. Vancouver, Canada.

Tyche, O. (2001). *Subtila språkstörningar hos patienter med diagnosen MCI (Mild Cognitive Impairment) Del I: utifrån den tematiska bilden "Kakstölden"*. Master's thesis, Karolinska Institute, Sweden (In Swedish).

Wallin, A. et al. (2016). The Gothenburg MCI study: Design and distribution of Alzheimers disease and subcortical vascular disease diagnoses from baseline to 6-year follow-up. *J Cer Blood Flow Metab*. 36(1):114-131.

Williams, C., Thwaites, A., Buttery, P., Geertzen, J., Randall, B., Shafto, M., Devereux, B., and Tyler, L. (2010). *The Cambridge Cookie-Theft Corpus: A Corpus of Directed and Spontaneous Speech of Brain-Damaged Patients and Healthy Individuals*. The 7th Conference on Language Resources and Evaluation (LREC). Malta.

Yancheva, M., Fraser, K., and Rudzicz, F. (2015). *Using linguistic features longitudinally to predict clinical scores for Alzheimer's disease and related dementias*. 6th SLPAT. Pp. 134–139, Dresden, Germany.

Yancheva, M., and Rudzicz, F. (2016). *Vector-space topic models for detecting Alzheimer's disease*. 54th Annual Meeting of the Association for Computational Linguistics. Pp 2337–2346. Berlin, Germany. Association for Computational Linguistics.

Wechsler, D. (1997). *Wechsler Memory Scale - Third Edition Manual.* The Psychological Corporation.

Weissenbacher, D. et al. (2016). *Automatic Prediction of Linguistic Decline in Writings of Subjects with Degenerative Dementia.* NAACL-HLT. Pp 1198-1207. c 2016 Association for Computational Linguistics

Sharing Copies of Synthetic Clinical Corpora without Physical Distribution — A Case Study to Get Around IPRs and Privacy Constraints Featuring the German JSYNCC Corpus

Christina Lohr **Sven Buechel** **Udo Hahn**

Jena University Language and Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena, Fürstengraben 27, D-07743 Jena, Germany
{christina.lohr|sven.buechel|udo.hahn}@uni-jena.de
http://www.julielab.de

Abstract

The legal culture in the European Union imposes almost unsurmountable hurdles to exploit copyright protected language data (in terms of intellectual property rights (IPRs) of media contents) and privacy protected medical health data (in terms of the notion of informational self-determination) as language resources for the NLP community. These juridical constraints have seriously hampered progress in resource-greedy NLP research, in particular for non-English languages in the clinical domain. In order to get around these restrictions, we introduce a novel approach for the creation and re-use of clinical corpora which is based on a two-step workflow. First, we substitute authentic clinical documents by synthetic ones, i.e., made-up reports and case studies written by medical professionals for educational purposes and published in medical e-textbooks. We thus eliminate patients' privacy concerns since no real, concrete individuals are addressed in such narratives. In a second step, we replace physical corpus distribution by sharing software for trustful re-construction of corpus copies. This is achieved by an end-to-end tool suite which extracts well-specified text fragments from e-books and assembles, on demand, identical copies of the same text corpus we defined at our lab at any other site where this software is executed. Thus, we avoid IPR violations since no physical corpus (raw text data) is distributed. As an illustrative case study which is easily portable to other languages we present JSYNCC, the largest and, even more importantly, first publicly available, corpus of German clinical language.

Keywords: clinical NLP, German language corpus, legal constraints on corpus construction and distribution

1. Introduction

In both its academic and industry branches, the NLP community has established professional standards in which the open accessibility and exchange of language resources (corpora and other data sets, such as lexical resources, annotation guidelines, and software) play a dominant and fertile role. This liberal policy is one of the most important factors for the remarkable progress the field has made in the past decades. The question, however, is how NLP is going to prosper under less friendly, or even hostile, accessibility conditions strictly prohibiting the free flow of language resources.

The problems we address here are often deeply rooted in national legal systems world-wide and reflect fundamental economic as well as social concerns (Mittelstadt and Floridi, 2016). As a consequence, they are rather persist even over long periods of time and NLP research has to find ways to accommodate to the overarching legal ecosystems. *Intellectual Property Rights* (IPRs) play a key role in this discussion and are of utmost relevance for any NLP research targeting unabridged media contents (in contrast to document surrogates, such as titles, index terms, text snippets or abstracts, etc.). The stakeholders actively promoting IPRs are major publishing houses and other companies distributing media content. Commercial interests are the driving forces behind IPRs like, for example, the claims of a creative inventor of some content and enterprises distributing that work via various media channels to generate revenues for both parties. IPRs and their relations to NLP are discussed in depth by Truyens and Van Eecke (2014).

Data privacy, another crucial topic with particular relevance for biomedical or social media-focused NLP, is an ethical category deeply rooted in the civil codes of Western societies. In essence, privacy regulations are in place to preserve each citizen's right to informational self-determination (Fischer-Hübner et al., 2011). This means that each individual owns his or her personal data, which, in turn imposes the task to protect this ownership on the legal system.

The realm of medical and clinical information constitutes a typical example where privacy regulations are legally enforced to protect sensitive data associated with former and current medical statuses of individual persons and their social environment. For a survey of ethical issues to be considered by NLP research in the medical domain, cf. Šuster et al. (2017).

However, different legal cultures have emerged to balance the highly-valued societal goal of privacy protection with the sometimes competing goal of generating knowledge from biomedical research. In the Anglo-American countries, medical information is open for usage by the scientific community once clinical data are safely de-identified (as acknowledged by ethical boards) and legally binding *Data Use Agreements* (DUAs) are established between the data provider (typically, a hospital) and a data consumer (e.g., a scientist).

Within the European Union and its member states, however, much more restrictive privacy constraints are in force – making it almost impossible to access and distribute even de-identified medical information. For instance, in the German-speaking countries, access to patient information

by actors outside the hospital site they originate from and by non-medical staff (e.g., computational linguists) is virtually impossible (Pommerening et al., 2014). Thus, NLP researchers in the biomedical domain (clinical NLP, in particular) are facing a lack of accessible language resources, at least when dealing with non-English languages.

One might remark that largely available resources reflecting general language use, e.g. from the newspaper domain, could be used for clinical NLP as well—with some additional efforts for domain adaptation to solve the data scarceness problem (Wermter and Hahn, 2004; Ferraro et al., 2013; Wang et al., 2015; Zhang et al., 2015). However, clinical language poses several domain-specific and rather tough challenges for NLP tools. Not only is the vocabulary abundant and highly specialized, but the language further deviates from standard usage in terms of spelling, typography and syntax: including, for example, short sentence fragments with paragrammatical structure, lack of punctuation, great volume and high degree of ambiguity of abbreviations, non-standard alphanumerical expressions, table-structured passages and mixed language use, including Greek and Latin forms (cf., e.g., Savkov et al. (2016)). Thus, adapting existing NLP tools to the clinical domain is arguably much more difficult than for many other domains and text genres. Furthermore, there is ample evidence that simply reusing standard NLP software trained on general language data (e.g., newspapers) results in severe losses of performance for biomedical applications (Tomanek et al., 2007; Ferraro et al., 2013; Hellrich et al., 2015).

The solution we are proposing here bypasses the two major obstacles for clinical NLP—IPRs and data privacy—in the following way. First, as discussed above, without any major legal changes, real, authentic clinical texts are, for the time being, inaccessible for most European languages. We therefore propose to substitute authentic clinical data by *synthetic* documents written by medical practitioners for education purposes. As a result, we are dealing with artificial, yet plausible and realistic medical scenarios, rather than identifiable social individuals and their personal legacy data and are thus able to circumvent any privacy concerns. Nonetheless, as these synthetic documents are typically contained in the electronic versions of medical textbooks ("e-books"), the second access restriction to be overcome relates to IPRs. To do so, we employ a procedure similar to the way TWITTER corpora are often distributed (cf., e.g., Rosenthal et al. (2015)): Instead of releasing the (IPR-protected) raw data, we distribute NLP software which (given access to the original books in electronic format) reliably re-creates the same corpus we designed in our lab at any other site where this tool suite is executed. This results in a situation were a physically non-shared corpus can still be shared 'virtually' and be used for community-wide annotation as well as benchmarking efforts.

In the following, we illustrate our approach for the JSYNCC (JENA SYNTHETIC CLINICAL CORPUS), the first publicly available corpus of German clinical language ever.[1]

[1]The proposed method of corpus construction and distribution aims at solving a general problem independent of specific national legislation cultures. However, for the specific case of the German legal system, very recently an interesting amendment to the national copyright law ("Urheberrechtsgesetz") has been installed by German authorities. Under certain conditions, this amendment allows for the sharing of corpora among *scientific partners* despite copyright protection, potentially mitigating some of the problems addressed in this contribution (at least for researchers located in Germany). For further information see UrhWissG (2018).

2. Related Work

There is a world-wide consensus on the fact that a patient's identity needs to be detached from medical data to protect sensitive personal data from any sort of misuse by non-medical or non-clinical actors. The standard way to achieve this requirement is by way of de-identification of so-called *Protected Health Information* (PHI). This is done with the help of a schema consisting of 18 categories, including data items that identify the patient in question (such as name, postal address, phone number, email address, or social security number), but also less apparent ones, such as names and locations of hospitals, their departments or clinical staff (for a complete list, cf. Stubbs and Uzuner (2015a)).

In the US, the past decade has seen a series of clinically oriented NLP shared tasks (for an extensive survey, cf. Huang and Lu (2016)). Prominent examples are the "Text Retrieval Conference" (TREC)[2] (Roberts et al., 2016) and the "Integrating Biology and the Bedside" (I2B2) initiative[3] (Chapman et al., 2011). From these activities, a wide range of de-identified and semantically annotated clinical corpora have emerged, covering the thematic foci of various competitions, such as de-identification (Stubbs and Uzuner, 2015a), medication extraction (Uzuner et al., 2010), temporal ordering of clinical events (Sun et al., 2013), or detecting risk factors for heart diseases (Stubbs and Uzuner, 2015b; Kumar et al., 2015). Each of the clinical corpora from I2B2 contains task-specific semantic metadata for slightly less than 1,000 English-language clinical reports. Those can be accessed, in a de-identified form, by simply signing a *Data Use Agreement* (DUA). Another major clinical database resource incorporating thousands of clinical reports, MIMIC III (Multiparameter Intelligent Monitoring in Intensive Care)[4] (Johnson et al., 2016), is also available via DUA contracting. Accordingly, for clinical NLP with focus on the English language, there are plenty of resources available (although not as abundant as in other non-medical fields such as newspapers or social media).

For the non-English language communities, however, less comfortable conditions apply. Only very few EU countries follow the DUA policy, such as reported for a clinical adverse drug reaction corpus for Spanish (Oronoz et al., 2015) or a comprehensive Dutch clinical corpus (Afzal et al., 2014). Some labs working on non-English languages have announced plans for releasing their resources, e.g., for French (Deléger et al., 2014), Polish (Marciniak and Mykowiecka, 2011) or Swedish (Dalianis et al., 2009). Apparently, these plans have not yet been fully realized as, to the best of our knowledge, none of these corpora is currently DUA-available for the research community.

[2]http://www.trec-cds.org/

[3]https://www.i2b2.org/NLP/DataSets/

[4]https://physionet.org/mimic2/

Corpus	Documents	Sentences	Types	Tokens	Available
Wermter and Hahn (2004) (FRAMED)	–	6,494	20,729	100,150	✗
Fette et al. (2012)	544	–	–	–	✗
Bretschneider et al. (2013b) Bretschneider et al. (2013a)	174	4,295	3,979	28,009	✗
Toepfer et al. (2015)	140	–	–	–	✗
Lohr and Herms (2016)	450	22,427	11,008	266,390	✗
Kreuzthaler and Schulz (2015) Kreuzthaler et al. (2016)	1,696	–	–	–	✗
Roller et al. (2016)	1,725	**27,939**	–	158,171	✗
Cotik et al. (2016)	183	2,234	–	12,895	✗
Krebs et al. (2017)	**3,000**	–	–	–	✗
Hahn et al. (2018) (3000PA)	**3,000**	–	–	–	✗
JSYNCC (this work)	867	24,895	**32,108**	**312,784**	✓

Table 1: Overview of existing corpora of German clinical language. Highest value per column in bold.

Another source of medical language resources in Europe derives from the "CLEF eHealth" initiative.[5] Established in 2013, this series of health-related challenges led to the preparation of several corpora—mostly for the English language, but also for other European languages. However, these corpora are typically very small and available only for usage directly related to the respective task, i.e., they can neither be used later on nor are they available for the research community independent of the specific CLEF task. For German-language medical corpora the situation is even worse—*all* clinical corpora are *only* accessible to the research staff within the lifetime of a project and remain inaccessible forever for the outside world. Schulz and López-García (2015) give an overview of technical, legal and organizational issues for clinical NLP in Germany (among other countries) and conclude that electronically archived patient records are typically not intended for further scientific use, e.g., text mining. Furthermore, there are no unanimously shared standards or guidelines for the storage of clinical notes and reports—resulting in a myriad of physical encodings of electronic patient data, even within a single hospital.

Nonetheless, there have been a few disconnected activities in the German NLP community to create in-project clinical corpora. In Table 1 we list, to the best of our knowledge, all existing German-language clinical corpora that have been described in scientific publications. Wermter and Hahn (2004) created FRAMED, the first German-language medical corpus ever published. It consists of a mixture of approximately 300 clinical reports, textbooks and consumer texts annotated with low-level linguistic metadata (up to the level of parts of speech). FRAMED was further used to generate in-domain machine learning models for different tasks, e.g., sentence splitting and tokenization (Faessler et al., 2014; Hellrich et al., 2015; Hahn et al., 2016). FRAMED has also become part of the multilingual extension of NEGEX, a corpus annotated for negation expressions (Chapman et al., 2013).

Bretschneider et al. (2013b) and Bretschneider et al. (2013a) introduce a corpus composed of German radiology reports and expand clinical lexical resources with information about pathology classification. Fette et al. (2012) assembled a corpus composed of 544 clinical reports from various medical domains (echocardiography, EEG, lung function, X-ray thorax, bicycle stress test) to train a CRF classifier for an information extraction (IE) task. Toepfer et al. (2015) describe a corpus made of 140 transthoracic echocardiography reports for their IE experiments. Lohr and Herms (2016) collected 450 surgery reports and used these resources to build language models adapted to metadata from two German medical thesauri. A collection of almost 1,700 de-identified clinical in- and outpatient discharge summaries were assembled from a dermatology department for an unsupervised abbreviation detection procedure (Kreuzthaler and Schulz, 2015) and supervised machine learning using an SVM for abbreviation and sentence delineation (Kreuzthaler et al., 2016). Roller et al. (2016) introduce an annotation scheme for a German corpus in the nephrology domain and a similar scheme focusing on negation phenomena is presented by Cotik et al. (2016). The latter two publications use discharge summaries and clinical notes as their document base. In the most recent publications, Krebs et al. (2017) describe a corpus of 3,000 chest X-ray reports used for term extraction (in an effort to improve IE) and Hahn et al. (2018) present 3000PA, a collection of 3,000 German discharge summaries from three different German university hospitals, currently annotated with medication information. This corpus is intended to become a national reference corpus for German clinical language based on a DUA-style agreement policy to be implemented in the future. Once again, *none* of these corpora is currently available for public use.

3. Corpus Creation

To mitigate the accessibility problems encountered for clinical corpora for most European (non-English) languages we propose an alternative workflow for the construction of a clinical corpus. This is illustrated by setting up JSYNCC, the first publicly available corpus of German clinical language. Although addressing German as an example, our approach is easily portable to other languages.

[5] https://sites.google.com/site/clefehealth/

Source	Area	Register	Doc.	Sent.	Types	Tokens
(Siekmann and Irlenbusch, 2012) (Siekmann and Klima, 2013) (Siekmann et al., 2016)	orthopedics trauma surgery	surgery reports	337	16,723	16,001	174,598
(Hagen, 2005)	general surgery	surgery reports	62	1,835	3,190	18,824
(Wenzel, 2015)	emergency medicine	case descriptions case discussions	48 48	2,710	11,933	63,316
(Eisoldt, 2017)	general surgery	case descriptions	140	699	2,162	10,338
(Hübler and Koch, 2014)	anesthetics	case descriptions	35	934	3,783	13,919
(Machado, 2013)	emergency medicine	case descriptions	11	398	1,785	5,950
(Thiel et al., 2013)	ophthalmology	case descriptions	36	540	3,108	10,181
(Hellmich, 2017)	internal medicine	case descriptions	150	1,056	3,113	15,658
JSYNCC total			**867**	**24,895**	**32,108**	**312,784**

Table 2: Raw data and summary statistics of JSYNCC. The top three entries contain duplicates and are therefore presented jointly.

The procedure for corpus construction first identifies publicly available (yet IPR-protected) synthetic medical language data contained in e-books as a reasonable substitute for authentic clinical data. Second, a software infrastructure is shared for constructing exact copies of the original corpus (given access to the raw data based on a purchasable license from the publisher of the e-books selected) rather than distributing the copyrighted raw data physically. This workflow can further be broken down into the following steps which we will discuss in more detail below:

- identification and selection of relevant language data,
- extraction and cleansing of relevant content,
- reformatting of the documents in XML and validation of the corpus file,
- optionally followed by annotation and/or text analysis.

3.1. Manual Selection of Raw Data

The e-books which form the basis of our corpus (see Table 2 for a detailed description) contain *synthetic* surgery reports, case descriptions as well as case discussions written by clinical professionals with the intention of training medical students in clinical documentation and decision-making. Together, they constitute the three registers of JSYNCC. This way, the problem of getting clearance from third-party ethical and legal bodies to deal with clinical documents addressing *real* individuals can be avoided.

As a result of carefully reviewing available educational material for the German language, we came up with a list of ten textbooks containing suitable synthetic data: (Siekmann and Irlenbusch, 2012), (Siekmann and Klima, 2013), and (Siekmann et al., 2016) supply reports of orthopedics and trauma surgeries, including fictional administrative information, such as patient numbers, time and date information, as well as the name of the physician in charge. Since the most recent of these publications is in part a combination of the earlier two, duplicated entries were removed during extraction resulting in a total of 337 synthetic documents. Similarly, (Hagen, 2005) provides 62 synthetic reports dealing with general surgery. Altogether, these 399 documents extracted from the first four books make up the *surgery reports* register of JSYNCC.

Furthermore, (Wenzel, 2015) describes 48 case examples from emergency medicine including an extensive discussion for each of these cases (forming the *case discussions* register). (Eisoldt, 2017) contains 140 short surgical case descriptions and (Hübler and Koch, 2014) comprises 35 case descriptions from anesthetics. (Machado, 2013) describes 11 medical emergency situations occurring in intercultural settings. Finally, JSYNCC also contains 36 case descriptions from ophthalmology by (Thiel et al., 2013) and 140 case descriptions from internal medicine by (Hellmich, 2017). Together, the documents extracted from these last six books form the *case descriptions* register of JSYNCC.

3.2. Automatic Extraction of Relevant Content

The whole process of creating our corpus runs fully automatically (mainly using JAVA) and is scripted so that everyone having access to the pre-selected e-books can generate an exact copy of JSYNCC without any manual intervention (other than assembling the necessary raw data). The entire software package for corpus construction is released on our lab's GITHUB site.[6]

For converting the e-books (which originally come in PDF format) to plain text, we used the JAVA-based APACHE TIKA tool[7] and the command-lines tools PDFTOTEXT[8] and PDFTOHTML[9] depending on which one generated the cleanest output for the book under scrutiny. For each book we determined unique typographic characteristics by which the starting and ending points of the relevant textual excerpts could be reliably localized. Along with each of the resulting 867 documents, we also extracted the following kinds of meta-information:

(1) the title of the book the excerpt originates from,
(2) the register of our corpus we assign it to (surgery reports, case descriptions, or case discussions),
(3) its heading as given in the textbook, as well as its topic.

The topic labels were assigned based on the chapter in which a given excerpt appeared, or similar structural in-

[6] https://github.com/JULIELab/jsyncc
[7] https://tika.apache.org/
[8] http://www.xpdfreader.com/pdftotext-man.html
[9] https://linux.die.net/man/1/pdftohtml

formation, with the heading labels being assigned based on the table of contents in the respective book.

In order to minimize the number of potential conversion errors in the final version of JSYNCC, we automatically post-processed the output of the tools mentioned above in a rule-based fashion. We manually revised each of the extracted documents, iteratively improving the post-processing procedure for each individual book. In this manner, we removed artifacts originating from the conversion of the print layout to plain text such as additional line breaks (while keeping paragraph segmentation), formatting characters, as well as superfluous hyphenation.

3.3. XML Conversion and Validation

The extracted and post-processed documents are stored in a single XML file together with their accompanying metadata. Figure 1 illustrates the structure of this file, as well as an exemplary section of the raw text in JSYNCC. To make sure that the corpus creation script works as intended for anyone wanting to re-build JSYNCC, we also provide checksums in a separate XML file for each individual document on which a validation script is based. By this, we can trustfully guarantee that each corpus copy created using our technical set-up is a valid copy (that is, it always produces the same corpus given the same selection of input documents).

3.4. Automatic Annotation and Analysis

In order to illustrate the potential of JSYNCC as a future benchmark corpus for German clinical NLP, we also provide automatically derived annotations on token-, sentence- and part-of-speech level. We used the UIMA-based tool suite JCORE[10] (Hahn et al., 2016) employing the publicly available models trained on the confidential clinical FRAMED corpus (Faessler et al., 2014). The resulting annotations are available in a stand-off XML format.

Based on these metadata, we computed the number of types, tokens, and sentences for each of the source textbooks (see Table 2) as well as those of the JSYNCC corpus as a whole. As can be seen from Table 1, besides being the first publicly available data set of German clinical language, JSYNCC is also the largest corpus ever published (containing over 300k tokens and 30k types).

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<corpus>
  <document>
    <id>1</id>
    <text>Vorgeschichte/Indikation: Sturz auf den
      Schädel unter Alkoholeinfluss. Anschl.
      HWS-Schmerzen. Konventionell radiologisch sowie
      im CT Nachweis der u.g. Fraktur. (...)</text>
    <type>operation report</type>
    <heading>Densfraktur - Verschraubung</heading>
    <topic>Orthopädie</topic>
    <topic>Unfallchirurgie</topic>
    <source>Siekmann, H., Irlenbusch, L., and Klima, S.
      (2016). Operationsberichte Orthopädie und
      Unfallchirurgie. Springer-Verlag.</source>
  </document>  (...)
</corpus>
```

Figure 1: Illustration of the automatically extracted and post-processed corpus in XML format.

[10] http://julielab.github.io/

4. Conclusion

Almost unsurmountable legal problems encountered when dealing with clinical documents in Germany and many other European countries have led us to consider using made-up *synthetic* rather than real *authentic* language data. Such substitutes can easily be extracted from electronically published educational medical textbooks. Thus, privacy protection concerns do not arise since artificial actors rather than real-life individuals are in focus.

Based on this design decision, we here outline a methodology for corpus development which leads to the creation of copy-identical (as guaranteed by checksums) corpora which can be trustfully built on demand at any physical lab site using the same software and selections of textual raw data. Since no textual data are physically distributed, legal IPR issues are avoided as well. Thus, we share corpus building software without touching sensitive legal ground related to IPR-protected raw data. Still, the NLP community is able to work with these corpora without any restriction and loss in raw data quality based on an easily manageable technical bypass.

We illustrated this highly portable approach introducing JSYNCC, the largest and—even more importantly—first *publicly available* corpus of German clinical language. Hence, for the first time ever, research on German clinical NLP (and other language communities on which strict legal protection constraints are imposed) can benefit from community-wide annotation efforts which may transform JSYNCC (and potential follow-up data sets) into a benchmark corpus for various tasks in future work.

The next obvious problem that needs to be tackled relates to the main assumption underlying our approach, i.e., assessing the similarity of authentic and synthetic clinical documents and thus estimating their substitutability. Accordingly, a stylistic sublanguage comparison study will be carried out in the future.

5. Acknowledgements

This work was partially supported by *Deutsche Forschungsgemeinschaft* (DFG) under grant HA 2097/8-1 within the STAKI^{2}B^2 project (Semantic Text Analysis for Quality-controlled Extraction of Clinical Phenotype Information within the Framework of Healthcare Integrated Biobanking).

6. Bibliographical References

Afzal, Z., Pons, E., Kang, N., Sturkenboom, M. C. J. M., Schuemie, M. J., and Kors, J. A. (2014). ContextD: An algorithm to identify contextual properties of medical terms in a Dutch clinical corpus. *BMC Bioinformatics*, 15:373.

Bretschneider, C., Zillner, S., and Hammon, M. (2013a). Grammar-based lexicon extension for aligning German radiology text and images. In *RANLP 2013 — Proceedings of the 9th International Conference on Recent Advances in Natural Language Processing. Hissar, Bulgaria, September 7-13, 2013*, pages 105–112.

Bretschneider, C., Zillner, S., and Hammon, M. (2013b). Identifying pathological findings in German radiology

reports using a syntacto-semantic parsing approach. In *BioNLP 2013 — Proceedings of the 2013 Workshop on Biomedical Natural Language Processing @ ACL 2013. Sofia, Bulgaria, August 8, 2013*, pages 27–35.

Chapman, W. W., Nadkarni, P. M., Hirschman, L., D'Avolio, L. W., Savova, G. K., and Uzuner, O. (2011). Overcoming barriers to NLP for clinical text: The role of shared tasks and the need for additional creative solutions. *Journal of the American Medical Informatics Association*, 18(5):540–543.

Chapman, W. W., Hillert, D., Velupillai, S., Kvist, M., Skeppstedt, M., Chapman, B. E., Conway, M., Tharp, M., Mowery, D. L., and Deléger, L. (2013). Extending the NEGEX lexicon for multiple languages. In *MEDINFO 2013 — Proceedings of the 14th World Congress on Medical and Health Informatics. Copenhagen, Denmark, 20-23 August 2013*, pages 677–681.

Cotik, V., Roller, R., Xu, F., Uszkoreit, H., Budde, K., and Schmidt, D. (2016). Negation detection in clinical reports written in German. In *BioTxtM 2016 — Proceedings of the 5th Workshop on Workshop on Building and Evaluating Resources for Biomedical Text Mining @ COLING 2016. Osaka, Japan, December 12, 2016*, pages 115–124.

Dalianis, H., Hassel, M., and Velupillai, S. (2009). The Stockholm EPR Corpus: Characteristics and some initial findings. In *ISHIMR 2009 — Evaluation and Implementation of e-Health and Health Information Initiatives: International Perspectives. Proceedings of the 14th International Symposium for Health Information Management Research. Kalmar, Sweden, October 14-16, 2009*, pages 243–249.

Deléger, L., Ligozat, A.-L., Grouin, C., Zweigenbaum, P., and Névéol, A. (2014). Annotation of specialized corpora using a comprehensive entity and relation scheme. In *LREC 2014 — Proceedings of the 9th International Conference on Language Resources and Evaluation. Reykjavik, Iceland, May 26-31, 2014*, pages 1267–1274.

Faessler, E., Hellrich, J., and Hahn, U. (2014). Disclose models, hide the data: How to make use of confidential corpora without seeing sensitive raw data. In *LREC 2014 — Proceedings of the 9th International Conference on Language Resources and Evaluation. Reykjavik, Iceland, May 26-31, 2014*, pages 4230–4237.

Ferraro, J. P., Daumé, H., DuVall, S. L., Chapman, W. W., Harkema, H. H., and Haug, P. J. (2013). Improving performance of natural language processing part-of-speech tagging on clinical narratives through domain adaptation. *Journal of the American Medical Informatics Association*, 20(5):931–939.

Fette, G., Ertl, M., Wörner, A., Klügl, P., Störk, S., and Puppe, F. (2012). Information extraction from unstructured electronic health records and integration into a data warehouse. In *INFORMATIK 2012: Was bewegt uns in der/die Zukunft? Proceedings der 42. Jahrestagung der Gesellschaft für Informatik e.V. (GI). Braunschweig, Germany, September 16-21, 2012*, number P-208 in GI-Edition - Lecture Notes in Informatics (LNI), pages 1237–1251. Gesellschaft für Informatik e.V. (GI), Bonner Köllen Verlag.

Fischer-Hübner, S., Hoofnagle, C., Krontiris, I., Rannenberg, K., and Waidner, M. (2011). Manifesto from Dagstuhl Perspectives Workshop 11061 "Online Privacy: Towards Informational Self-Determination on the Internet". Technical report, Schloss Dagstuhl, Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany.

Hahn, U., Matthies, F., Faessler, E., and Hellrich, J. (2016). UIMA-based JCORE 2.0 goes GITHUB and MAVEN CENTRAL: State-of-the-art software resource engineering and distribution of NLP pipelines. In *LREC 2016 — Proceedings of the 10th International Conference on Language Resources and Evaluation. Portorož, Slovenia, May 23-28, 2016*, pages 2502–2509.

Hahn, U., Matthies, F., Lohr, C., and Löffler, M. (2018). 3000PA: Towards a national reference corpus of German clinical language. In *MIE 2018 — Proceedings of the 29th Medical Informatics in Europe Conference. Gothenburg, Sweden, April 23-25, 2018*.

Hellrich, J., Matthies, F., Faessler, E., and Hahn, U. (2015). Sharing models and tools for processing German clinical texts. In *MIE 2015 — Proceedings of the 26th Medical Informatics in Europe Conference. Madrid, Spain, May 27-29, 2015*, pages 734–738.

Huang, C.-C. and Lu, Z. (2016). Community challenges in biomedical text mining over 10 years: Success, failure and the future. *Briefings in Bioinformatics*, 17(1):132–144.

Johnson, A. E. W., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M. M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:#160035.

Krebs, J., Corovic, H., Dietrich, G., Ertl, M., Fette, G., Kaspar, M., Krug, M., Stoerk, S., and Puppe, F. (2017). Semi-automatic terminology generation for information extraction from German chest X-ray reports. In Rainer Röhrig, et al., editors, *German Medical Data Sciences: Visions and Bridges. Proceedings of the 62nd Annual Meeting of the German Association of Medical Informatics, Biometry and Epidemiology (gmds e.V.). Oldenburg (Oldenburg), Germany, 17-21 September 2017 — GMDS 2017*, number 243 in Studies in Health Technology and Informatics, pages 80–84. IOS Press.

Kreuzthaler, M. and Schulz, S. (2015). Detection of sentence boundaries and abbreviations in clinical narratives. *BMC Medical Informatics and Decision Making*, 15(Suppl 2):S4.

Kreuzthaler, M., Oleynik, M., Avian, A., and Schulz, S. (2016). Unsupervised abbreviation detection in clinical narratives. In *ClinicalNLP 2016 — Proceedings of the Clinical Natural Language Processing Workshop @ COLING 2016. Osaka, Japan, December 11, 2016*, pages 91–98.

Kumar, V., Stubbs, A., Shaw, S. Y., and Uzuner, O. (2015). Creation of a new longitudinal corpus of clinical narratives. *Journal of Biomedical Informatics*, 58(Supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data)):S6–S10.

Lohr, C. and Herms, R. (2016). A corpus of German clinical reports for ICD and OPS-based language modeling. In *CLAW 2016 — Proceedings of the 6th Workshop on Controlled Language Applications @ LREC 2016. Portorož, Slovenia, May 28, 2016*, pages 20–23.

Marciniak, M. and Mykowiecka, A. (2011). Towards morphologically annotated corpus of hospital discharge reports in Polish. In *BioNLP 2011 — Proceedings of the Workshop on Biomedical Natural Language Processing @ ACL-HLT 2011. Portland, Oregon, USA, June 23-24, 2011*, pages 92–100.

Mittelstadt, B. D. and Floridi, L. (2016). The ethics of Big Data: Current and foreseeable issues in biomedical contexts. *Science and Engineering Ethics*, 22(2):303–341.

Oronoz, M., Gojenola, K., Pérez, A., Diaz de Ilarraza, A., and Casillas, A. (2015). On the creation of a clinical gold standard corpus in Spanish: Mining adverse drug reactions. *Journal of Biomedical Informatics*, 56:318–332.

Pommerening, K., Drepper, J., Helbing, K., and Ganslandt, T. (2014). *Leitfaden zum Datenschutz in medizinischen Forschungsprojekten. Generische Lösungen der TMF 2.0*. Number 11 in Schriftenreihe: Projektergebnisse aus der TMF und Referenzwerke zur Gesundheitstelematik. TMF — Technologie- und Methodenplattform für die vernetzte medizinische Forschung e.V., Berlin.

Roberts, K., Simpson, M. S., Demner-Fushman, D., Voorhees, E. M., and Hersh, W. R. (2016). State-of-the-art in biomedical literature retrieval for clinical cases: A survey of the TREC 2014 CDS Track. *Information Retrieval Journal*, 19(1-2):113–148.

Roller, R., Uszkoreit, H., Xu, F., Seiffe, L., Mikhailov, M., Staeck, O., Budde, K., Halleck, F., and Schmidt, D. (2016). A fine-grained corpus annotation schema of German nephrology records. In *ClinicalNLP 2016 — Proceedings of the Clinical Natural Language Processing Workshop @ COLING 2016. Osaka, Japan, December 11, 2016*, pages 69–77.

Rosenthal, S., Nakov, P. I., Kiritchenko, S., Mohammad, S. M., Ritter, A., and Stoyanov, V. (2015). SemEval-2015 Task 10: Sentiment analysis in TWITTER. In *SemEval 2015 — Proceedings of the 9th International Workshop on Semantic Evaluation @ NAACL-HLT 2015. Denver, Colorado, USA, June 4-5, 2015*, pages 451–463.

Savkov, A., Carroll, J. A., Koeling, R., and Cassell, J. A. (2016). Annotating patient clinical records with syntactic chunks and named entities: The HARVEY Corpus. *Language Resources and Evaluation*, 50(3):523–548.

Schulz, S. and López-García, P. (2015). Big Data, medizinische Sprache und biomedizinische Ordnungssysteme. *Bundesgesundheitsblatt – Gesundheitsforschung – Gesundheitsschutz*, 58(8):844–852.

Stubbs, A. and Uzuner, O. (2015a). Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/UTHealth Corpus. *Journal of Biomedical Informatics*, 58(Supplement):S20–S29.

Stubbs, A. and Uzuner, O. (2015b). Annotating risk factors for heart disease in clinical narratives for diabetic patients. *Journal of Biomedical Informatics*, 58(Supplement):S78–S91.

Sun, W., Rumshisky, A., and Uzuner, O. (2013). Annotating temporal information in clinical narratives. *Journal of Biomedical Informatics*, 46(Supplement: Proceedings of the 2012 i2b2 NLP Challenge on Temporal Relations in Clinical Data):S5–S12.

Šuster, S., Tulkens, S., and Daelemans, W. (2017). A short review of ethical challenges in clinical natural language processing. In *Proceedings of the 1st ACL Workshop on Ethics in Natural Language Processing @ EACL 2017. Valencia, Spain, April 4, 2017*, pages 80–87.

Toepfer, M., Corovic, H., Fette, G., Klügl, P., Störk, S., and Puppe, F. (2015). Fine-grained information extraction from German transthoracic echocardiography reports. *BMC Medical Informatics and Decision Making*, 15:#91.

Tomanek, K., Wermter, J., and Hahn, U. (2007). A reappraisal of sentence and token splitting for life sciences documents. In Klaus A. Kuhn, et al., editors, *MedInfo 2007 — Proceedings of the 12th World Congress on Health (Medical) Informatics. Building Sustainable Health Systems. Brisbane, Australia, August 20-24, 2007*, number 129 in Studies in Health Technology and Informatics, pages 524–528, Amsterdam. IOS Press.

Truyens, M. and Van Eecke, P. (2014). Legal aspects of text mining. In Nicoletta Calzolari, et al., editors, *LREC 2014 — Proceedings of the 9th International Conference on Language Resources and Evaluation. Reykjavik, Iceland, May 26-31, 2014*, pages 2182–2186.

UrhWissG. (2018). Gesetz zur Angleichung des Urheberrechts an die aktuellen Erfordernisse der Wissensgesellschaft (Urheberrechts-Wissensgesellschafts-Gesetz – UrhWissG) vom 1. September 2017 (BGBl. Teil I Nr. 61, Bonn, September 7th 2017), came into force on March 1, 2018. In *BGBl. I*, pages 3346–3351. Available: `https://www.bmjv.de/SharedDocs/Gesetzgebungsverfahren/Dokumente/BGBl-UrhWissG.pdf`.

Uzuner, O., Solti, I., Xia, F., and Cadag, E. (2010). Community annotation experiment for ground truth generation for the i2b2 medication challenge. *Journal of the American Medical Informatics Association*, 17(5):519–523.

Wang, Y., Pakhomov, S. V. S., Ryan, J. O., and Melton, G. B. (2015). Domain adaption of parsing for operative notes. *Journal of Biomedical Informatics*, 54:1–9.

Wermter, J. and Hahn, U. (2004). Really, is medical sublanguage that different? Experimental counter-evidence from tagging medical and newspaper corpora. In Marius Fieschi, et al., editors, *MEDINFO 2004 — Proceedings of the 11th World Congress on Medical Informatics. San Francisco, California, USA, September 7-11, 2004*, volume 1 of *Studies in Health Technology and Informatics*, pages 560–564, Amsterdam. IOS Press.

Zhang, Y., Tang, B., Jiang, M., Wang, J., and Xu, H. (2015). Domain adaptation for semantic role labeling of clinical text. *Journal of the American Medical Informatics Association*, 22(5):967–979.

7. Language Resource References

Eisoldt, Stefan. (2017). *Fallbuch Chirurgie: 140 Fälle aktiv bearbeiten*. Georg Thieme Verlag, 5th edition.

Hagen, Monika. (2005). *Operationsberichte für Einsteiger-Chirurgie: Operation vorbereiten—Bericht diktieren*. Georg Thieme Verlag.

Hellmich, Bernhard. (2017). *Fallbuch Innere Medizin*. Georg Thieme Verlag, 5th edition.

Hübler, Matthias and Koch, Thea. (2014). *Komplikationen in der Anästhesie*. Springer-Verlag, 3th edition.

Machado, Carl. (2013). *Patienten aus fremden Kulturen im Notarzt- und Rettungsdienst: Fallbeispiele und Praxistipps*. Springer-Verlag.

Siekmann, Holger and Irlenbusch, Lars. (2012). *Operationsberichte Unfallchirurgie*. Springer-Verlag.

Siekmann, Holger and Klima, Stefan. (2013). *Operationsberichte Orthopädie: mit speziellen unfallchirurgisch-orthopädischen Eingriffen*. Springer-Verlag.

Siekmann, Holger and Irlenbusch, Lars and Klima, Stefan. (2016). *Operationsberichte Orthopädie und Unfallchirurgie*. Springer-Verlag.

Thiel, Michael A and Bernauer, Wolfgang and Schüpfer, Marlis Zürcher and Schmid, Martin K. (2013). *Fallbeispiele Augenheilkunde*. Springer-Verlag.

Wenzel, Volker. (2015). *Fallbeispiele Notfallmedizin: Einprägsam—spannend—mit Lerneffekt*. Springer-Verlag.

A Legal Perspective on Training Models for Natural Language Processing

Richard Eckart de Castilho[♣], Giulia Dore[♠], Thomas Margoni[♠], Penny Labropoulou[♡], Iryna Gurevych[♣]

[♣]Technische Universität Darmstadt, Department of Computer Science, UKP Lab

[♠]University of Glasgow, School of Law - CREATe Centre

[♡]Athena RC, Institute for Language and Speech Processing

Abstract

A significant concern in processing natural language data is the often unclear legal status of the input and output data/resources. In this paper, we investigate this problem by discussing a typical activity in Natural Language Processing: the training of a machine learning model from an annotated corpus. We examine which legal rules apply at relevant steps and how they affect the legal status of the results, especially in terms of copyright and copyright-related rights.

Keywords: Copyright, Licensing, Machine Learning, Annotated Corpora

1. Introduction

The state-of-the-art in many areas of Natural Language Processing (NLP) and Text Mining (TM) is based on Machine Learning (ML). Algorithms learn abstract probabilistic models from texts annotated with labels (e.g. named entities, part-of-speech tags, sentiment tags, etc.) in order to predict such labels on unseen text. NLP tasks usually require the deployment of multiple components each using specialised models. As training models can be tedious and computationally intensive, pre-trained models are a valuable resource.

However, the legal status of these models is often dubious, as in many cases it is unclear (a) whether a model can be trained from a corpus in absence of specific authorisation, (b) which licence (if any) can or must be assigned to them, and (c) if and in which cases the licence(s) of the original corpus and annotations affect the licensing of a model. This legal uncertainty often constitutes a hurdle, if not a real barrier for the development of research infrastructures and repositories, such as CLARIN[1] or OpenMinTeD,[2] where models are shared and used.

In this paper, we explore the process of training a model both from an NLP and a legal perspective. We discuss under which circumstances annotated corpora may be used for training a model and if their legal status may restrict the choice of licences that can be applied to the trained model. We use EU copyright law as the main reference, although the analysis may find application beyond the EU (and may need some degree of adjustment in different EU Member States).

2. Background

This section starts introducing the typical actions and resources involved in ML models construction and deployment and successively discusses the relevant legal concepts.

2.1. NLP perspective

Models are constructed through a training process involving a *learning algorithm* and *training data* to learn from. The model captures abstract probabilistic characteristics from the training data, which can then be used to predict the learned labels on unseen data. For illustration purposes, we focus on Named Entity Recognition (NER), an example of a sequence classification task.

In general, constructing a model consists of the following steps: (1) corpus compilation, (2) corpus pre-processing, (3) corpus annotation, and (4) training of the model. Depending on the availability of an annotated corpus, one or more of these steps can be skipped. We briefly describe all of these steps here and elaborate on the training step in the main parts of our investigation.

Corpus compilation. Each corpus is compiled to capture specific aspects of real world language. For best results, the ML algorithm must be trained on a corpus (i.e. set of texts) that is similar to the corpus to which it is later applied; i.e., it must be of the same language and domain or text type and annotated with the appropriate labels, e.g., "*English*", "*Social Sciences*", "*scholarly publications*" and "*named entities*" (NE), respectively. The corpus texts are selected and obtained from one or more sources (e.g. publishers, journals, web sites, etc.).

Pre-processing. This involves all kinds of (usually automatic) processes required to convert the textual content into a format that can be further processed by the NLP tools, such as conversion of PDF or HTML files into plain text, removal of images, tables, etc.

Annotation. This is the task of manual or automatic enrichment of texts with labels relevant to the target task, possibly further corrected by experts. Annotations are often arranged in layers, e.g. grammatical categories, morphological or syntactic features, etc. In all cases, the human annotator or the tool "*reads*" the text which is segmented into units (e.g. words, phrases) and assigns to some or all of them the appropriate labels. The inventory of labels is defined in *annotation resources*, such as tagsets, ontologies, thesauri, etc. The assignment usually follows *instructions* (e.g. guidelines, grammar rules, statistical data) that define when to assign labels and how to disambiguate if multiple candidate labels exist.

Training. The *training tool* is a software programme that implements an ML algorithm which is applied to the anno-

[1]https://www.clarin.eu

[2]https://www.openminted.eu

tated corpus, analyses its features and extracts from it the appropriate probabilistic and statistical characteristics. The model can thus be regarded as an abstraction of the annotated corpus based on statistical observations which can then be used with a second software tool (*tagger*) to predict the learned labels (e.g. NEs) on unseen text.
For the present analysis, we assume that ML tools (trainers and taggers) are governed by licenses that do not impose restrictions on the models they create. A similar assumption is made in relation to the use of predictions ML tools make. Accordingly, the analysis focuses on the licensing terms of the annotated corpora and the actions performed on them while training a model.

2.2. Legal perspective

Before proceeding to the discussion of the three scenarios, this section clarifies some basic copyright law concepts.

Texts and literary works. Most corpora employed in NLP consist of web pages, publications, articles, newspaper texts, blog posts or even tweets, annotated or not. All these resources possess the potential to be protected by *copyright law*. To be eligible for copyright protection a work must be original. The originality standard has been harmonised by EU law at the level of the *author's own intellectual creation*. Current legislation[3] and relevant case law[4]. indicate that this harmonised level of originality, despite the evocative formula employed, is placed at a rather low level (Margoni, 2016) and the Court of Justice of the EU (CJEU) has held that 11 consecutive words can in certain circumstances be considered the author's own intellectual creation, thus protected by copyright. This must be verified on a case-by-case basis and it is achieved when an author is able to put their personal stamp onto the work through free and creative choices. Therefore it cannot be excluded that even single sentences, if original, can be object of copyright protection. In conclusion, it can be assumed that most corpora used for TM/NLP, especially those of a literary and scientific character, such as scholarly articles, are protected copyright.

Databases. Under EU law, as well as under the law of many other countries,[5] databases are defined as collections of independent works, data or other materials arranged in a systematic or methodical way and individually accessible by electronic or other means.[6] Copyright exists if originality is found in the *selection or arrangement of the content*, i.e., the "*intellectual creation*" has to be found in the database structure. Consequently, copyright in databases protects only the structure and does not extend to the content. The content, in turn, can be autonomously protected by copyright (a database of scholarly articles), related rights (a database of sound recordings), or be in the public domain (a database of unprotected facts or of medieval texts).

In addition, EU law, unlike the law of most other countries in the world, has introduced a new right protecting non-original databases when a substantial investment has been put in the obtaining, verification or presentation of the data – but importantly not in the creation of the data. In this case, the database maker (usually the person or entity who bears the financial risk) enjoys a *sui generis database right* (SGDR), which protects the *content* of the database from substantial extractions. In other words, even databases of unprotected facts could become object of a proprietary right that extends to the database content in the light of the aforementioned substantial investment (Hugenholtz, 2016; Guibault and Wiebe, 2013). Therefore, certain collections of corpora (e.g. the database of Institute X that over the years has collected public domain corpora investing substantial time and work resources in the process) could be protected by the SGDR.
Copyright and database rights are probably the two most relevant rights potentially covering the annotated and unannotated corpora forming the basis for any training activity (Stamatoudi and Torremans, 2000; Firdhous, 2012; Payne and Landry, 2012; Borghi and Karapapa, 2013; Tsiavos et al., 2014; Truyens and Van Eecke, 2014; Triaille et al., 2014; Handke et al., 2015). In the following sections, we examine this aspect more closely in the context of three basic scenarios. Specific attention will be paid to: (a) the *right of reproduction* (making copies) of the resources in question and whether the distinction between temporary and permanent copies matters; (b) the right of *adaptation and/or translation*, i.e. the creation of works based on those resources, what is often –but imprecisely from a EU law point of view– called a *derivative work*; and (c) the specific *licence types* under which annotated corpora are distributed. Nevertheless, no specific attention will be dedicated to the annotation process as many corpora are already available in an annotated form.

3. Scenario I: Liberally Licensed Corpora

We start our investigation with a straightforward scenario: training a model on an annotated corpus with a liberal TM/NLP-friendly licence carrying only an attribution clause. Here, we choose the popular Creative Commons Public Licence with the Attribution clause in the latest version available (CC BY 4.0). This licence is particularly "*friendly*" for TM/NLP activities because it authorises licensees to perform all the aforementioned rights (reproduction, redistribution, communication to the public, adaptation, etc.) under the only main condition that attribution be maintained.

3.1. Scenario description

Corpus. Despite version 4.0 of the CC licences being available since 2013, many of the existing CC BY licensed corpora are still distributed under older versions such as CC BY 2.5 (e.g. the Wikinews texts of the GUM corpus (Zeldes, 2017)) or CC BY 3.0 (e.g. the IULA Spanish LSP Treebank (Marimon et al., 2014) or the CRAFT corpus (Verspoor et al., 2012)).[7] We only consider the latest 4.0 version in the present analysis, but it should be noted that

[3] E.g. Directive 2009/24/EC, OJ L 111, 5.5.2009, 16–22, Article 1(3).

[4] E.g. Judgment of 16 July 2009, *Infopaq International v. Danske Dagblades Forening*, C-5/08, ECLI:EU:C:2009:465

[5] General Agreement on Trade-Related Aspects of Intellectual Property (TRIPS), 1869 U.N.T.S. 299, 33 I.L.M. 1197; WIPO Copyright Treaty (WCT), 105-17 (1997), 36 ILM 65(1997)

[6] Directive 96/9/EC, OJ L 77, 27.3.1996, Article 1

[7] It is notably difficult to locate corpora under specific CC licence versions. E.g. at the time of writing, META-SHARE (http://www.meta-share.org) lists over 200 corpora under CC BY,

different licence versions could lead to different assessments especially in relation to the SGDR right. Examples for corpora under this licence version are the recent GermEVAL 2014 dataset for NER (Benikova et al., 2014) or the Coptic Treebank (Schroeder and Zeldes, 2016).[8]

Training. In our scenarios, we train a NER model with the Stanford NER tool (Manning et al., 2014). To determine the relation of the trained model to the original data, we examine which information goes into the model. We describe the process in the present and following scenarios at increasing levels of detail, as required by the respective legal analyses. The training process requires the creation of a usually temporary copy (i.e. a reproduction) of the original data and usually its transformation into the training data format. The training data format used by the Stanford NER tool is very simple: a two-column format in which the first column contains a *token* (word or punctuation mark) and the second column contains a *label*. Sentences are separated by a blank line. If the word is a named entity, the label indicates the entity type (e.g. person, organisation, etc.), otherwise it contains a special "*no category*" label. To handle cases where a NE consists of multiple tokens, the type is either prefixed with *B-* to indicate the first token of the NE, *I-* for the other tokens of the NE and *O* for single-token NEs. This so-called *BIO*-encoding is a technical convention allowing the NER tool to learn how to correctly detect multi-token NEs.
Before considering the training process in more detail in Scenario II, we investigate whether the process up to this point is permitted by the licence.

3.2. Scenario analysis

Is reproduction permitted? According to the terms of the CC BY 4.0 licence, the act of making reproductions is expressly permitted, as per Section 2 of the licence text.[9]
In the present case, thanks to the liberal conditions established by the licence, it is not necessary to investigate whether the *results* of the training activity constitute a reproduction of the original corpora. The applicable licence permits any type of reproduction, being it the transient reproduction necessary for the conversion of the corpora into a machine processable format, or the final results of the training process Therefore, annotated corpora under these licences may be reproduced as part of the model training process on the basis of the licence (in those cases when this act is not covered by applicable exceptions and limitations, a situation that would not trigger the terms of the CC licence, see Scenario III below).

Is the result an adaptation (derivative work)? As stated above, the scenario is predicated on the assumption that all input resources (raw texts and annotations) are covered by a CC BY 4.0 license.

but does not carry information about the licence version. The LINDAT/CLARIN repository (http://lindat.mff.cuni.cz/) includes the licence version metadata, but the search interface does not allow filtering resources by it.

[8] We should note here that we have not investigated whether the assignment of these licences to the respective corpora is indeed valid.

[9] https://creativecommons.org/licenses/by/4.0/legalcode, s. 2 Section 2a of this licence expressly permits the creation of adaptations and defines it contractually, although ultimately referring to the applicable copyright law. Since the creation of an adaptation is explicitly permitted under the terms of the licences, if the act of training a model constitutes a derivative work, which is indeed not trivial to determine (see Scenario II below), in the present scenario this activity is permitted, under a mere attribution condition. It is worth pointing out that the attribution requirement in the present case would require: a) retaining attribution, copyright and licence notices, and providing a URI or hyperlink to the licensed material to the extent reasonably practicable; b) indicating modifications to the licensed material and retain an indication of any previous modifications; c) indicating the licensed material is licensed under this public license, including the text of, or the URI or hyperlink to the licence. No other limitations apply to the model. In principle, the model could also be re-licensed under any arbitrary licence if it qualifies for copyright protection on its own right, an aspect that will be discussed below in Scenario II.
In conclusion, training models on the basis of liberally licensed corpora does not present major legal obstacles, although proper attribution should be given.

4. Scenario II: Corpora with a Reciprocal Licence

We continue our investigation with a slightly more complicated case: training a model on an annotated corpus with a TM/NLP-friendly licence which includes a share-alike clause, such as the Creative Commons Attribution-ShareAlike 4.0 (CC BY-SA 4.0). This means that works adapted (derived) from the original work must carry the same licence as the original work. Since this is a reciprocal condition, it is important to determine whether a model is an adaptation (derivative work) of the corpora under the conditions of the licence.

4.1. Scenario description

Corpus. We consider the corpus to be licensed under the CC BY-SA 4.0 license, such as the latest version of the SETimes.HR dataset (Agić and Ljubešić, 2014).[10]

Training. The basic scenario is the same NER training process we have already started to describe in Scenario I. Assuming the corpus at hand has already been transformed into the two-column format described in Scenario I, the process of training a model is rather straightforward (although it may require significant computational resources).
As a first step, a configuration file for the Stanford NER tool needs to be created. This file contains the names of the files that comprise the training corpus and a name to be used for the output file (i.e. the model to be created), as well as a set of parameters controlling which features are extracted from the training data and used for training the classifier. An example parameter file (Figure 1) can be found in the documentation of the Stanford NER tool.[11]

[10] https://github.com/ffnlp/sethr (Agić and Ljubešić, 2014) is based on texts from the Croatian translation of the SETimes portal, which were freely shared with attribution to the source.

[11] https://nlp.stanford.edu/software/crf-faq.html – A more exten-

```
trainFile = training-data.col
serializeTo = ner-model.ser.gz
map = word=0,answer=1

useClassFeature=true
useWord=true
useNGrams=true
noMidNGrams=true
maxNGramLeng=6
usePrev=true
useNext=true
useSequences=true
usePrevSequences=true
maxLeft=1
useTypeSeqs=true
useTypeSeqs2=true
useTypeySequences=true
wordShape=chris2useLC
useDisjunctive=true
```

Figure 1: Example parameter file *params.prop*.

As a second step, the Stanford NER tool is started in training mode using the command *java -cp stanford-ner.jar edu.stanford.nlp.ie.crf.CRFClassifier -prop params.prop*. From this point on, the process runs fully automatically without further interaction from the person training the model.

4.2. Scenario analysis

Is reproduction permitted? Once again, the act of making copies (reproductions) is expressly allowed by the CC BY-SA 4.0 in the same terms analysed in Scenario I.

Is the result an adaptation (derivative work)? The creation of adapted materials is also expressly permitted, as it was with CC BY 4.0 in the previous scenario. However, the SA clause that applies in the present scenario requires that distribution of the adapted material be made under the terms of the same licence or a later version with the same terms.
Therefore, it is important to determine whether the trained model is an adaptation of the original annotated corpora. If it is, the SA clause requires that the same licence be applied to the trained model.
What constitutes adapted material is defined in Section 1(a) of the licence[12] as "*... material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor.*" To establish when this happens is a determination that can be done only against a specific domestic legal framework even within the EU. In fact, differently from other rights, the right of adaptation is not harmonised by EU law.[13]
However, not all modifications lead to the creation of an adapted work. EU law seems to suggest that such a new (adapted) work is created only when the process of modification involves an original contribution (in the sense of the author's own intellectual creation). This appears to be also the view of the licences developers. [14] Absent enough originality in the modification, the unoriginally modified material does not constitute a new copyright protected work, but rather a mere reproduction (even if partial or "*in any form*"), which, unless authorised by law or by contract, infringes the copyright (the right of reproduction) in the original work.
Given this definition of adapted material, we need to consider whether the act of training a model using the outlined procedure meets the licence requirements.
The simplicity of the training process outlined above and the limited choices in the parametrisation of a largely automated process suggest that there is no space for the free and creative choices that allow the author to express their personality into the work. In particular, it seems that even when certain choices in parametrisation are available these are dictated mostly by technical considerations and by the "*rules of the game*" of model training in a way that any equally skilled technician would achieve a similar or identical result. Under this interpretation, the model is not a creative adaptation of the underlying annotated text corpora and thus does not qualify as adapted material under the SA clause of the CC license.
This means that the trained model, not being an adaptation of the underlying corpora, does not trigger the SA clause. Training a model, as seen above, requires other types of copyright relevant acts, namely reproduction, which must be authorised or excused –statutorily or contractually– to avoid infringement. In the present case, it means that if the trained model, which does not qualify as adapted material, is nonetheless a "*reproduction in part*" of the original corpora, that part –and only that part– remains under the conditions of the original licence, in the present case a CC BY-SA 4.0.
Similar conclusions would be reached if the resources employed in the training process were licensed under a CC BY-ND 4.0. The licence in question, in fact, although allowing the creation and reproduction of adapted works, does not allow for their distribution (alias sharing), as specified in its Section 2(a).[15] If the model is not an adapted work in the meaning outlined above, then the NoDerivatives (ND) clause will not be triggered.
Finally, does this mean that the trained model can be arbitrarily licensed by its developer? In the present case, the trained model lacks sufficient originality to qualify as a derivative as well as an independent work. Therefore, it is not a work of authorship for copyright law purposes. In theory a license could still be applied but this would only have contractual effects and not be based on an underlying property right (a very relevant difference that cannot be explained here, suffice to say that most copyright licences are based on a valid underlying property right: if this is not present the effects of the licence are limited. In the case of CC licenses, as well as most "*open*" licences, if the licence is applied to something that is not protected by copyright or related rights the licence is not triggered).

sive list of parameters can be found in the documentation to the Stanford NER Java class *NERFeatureFactory*.

[12] https://creativecommons.org/licenses/by/4.0/legalcode, s. 1

[13] Judgement of 22 Jan 2015, *Art and Allposters International BV*, Case C-419/13, ECLI:EU:C:2015:27

[14] https://creativecommons.org/faq/#when-is-my-use-considered-an-adaptation

[15] https://creativecommons.org/licenses/by-nd/4.0/legalcode

```
#<objec#|C
#acter>#|C
consisted...xx-PW_CTYPE|C
xxxxx...about-NW_CTYPE|C
many-NW|C
nowadays-WORD|C
housed-PW|C
law-PSEQW|CpC
Planet-Wide-PSEQpW|CpC
essentially-still-PSEQW2|CpC
```

Figure 2: Feature excerpt from a CoreNLP NER model.

5. Scenario III: Corpora with unclear licence statements or restrictively licensed

In the previous scenarios we discussed texts and annotations under licences which explicitly allow reproduction. However, it is much more common that only annotations are under such a licence. Obtaining corpora under similar licences is much more difficult. Most of the texts that can be found online do not carry any licence at all or are part of commercial offers which do not permit reproduction. Thus, in the present scenario we investigate if such texts can still be used for training models. We do not investigate the relationship between corpora and annotation.

5.1. Scenario description

Corpus. An example of a text corpus obtained from the web and enriched with annotations under a CC licence is the English part of the Universal Dependency Treebank (UDT-EN).[16] While the annotations are provided under CC BY-SA 4.0, the texts come from the English Web Treebank[17] which has been collected by Google from online weblogs, newsgroups, emails, reviews and question-answering websites. As the UDT-EN website states, the copyright of portions of the texts may reside with "*Google Inc., Yahoo! Inc., Trustees of the University of Pennsylvania and/or other original authors*." Thus, the licensing status of the individual texts is not entirely transparent, and should be prudently considered to be under an "*all rights reserved*" status.

Training. Again, we consider the same process of training a NER model described in Scenarios I and II. However, this time we take a closer look at the training process in order to assess whether the model *reproduces* significant parts of the original document, an activity reserved to the copyright owner.

During the training process, the NER training tool extracts so-called *features* from the input data. This is the critical step in the training process, as it determines how much of the original text and annotations is retained. Figure 2 shows a sample of feature values generated by the Stanford NER. The Stanford NER uses a sequence classifier based on conditional random fields (CRFs). The tool runs through the text token-by-token and consumes features that have been extracted for each token, such as the token string, a configurable number of characters forming the prefix/suffix of the token, the left and right context of each token, e.g. the fact that token *X* appears left of token *Y*, and similar information. The context captured in the features is very limited and usually includes only the current word and a preceding and following word. E.g. "*essentially-still-PSEQW2—CpC*" indicates that the sequence "*essentially still*" was included in the training corpus. Additionally, the CRF learns a set of weights encoding the probability of an NE occurring in the presence of the specific features (Finkel et al., 2005). It is important that the features and weights capture a *limited* information about the tokens and their annotations. They *discard* the context details, because the ML algorithm needs to learn a *generalised model* from the training data which does not overfit on the training data.

The features and algorithm for NER-like tasks are designed in such a way that the trained model represents an abstraction of the training data. It is generally not possible to reconstruct the original text from this abstraction.

5.2. Scenario analysis

Is reproduction permitted? As briefly observed above, the right of reproduction, defined as "*any direct or indirect, temporary or permanent reproduction by any means and in any form, in whole or in part*" is reserved to the right holder of copyright works in all EU countries by Art. 2 InfoSoc Directive and its national implementations.[18]

The CJEU had the opportunity to clarify that certain acts of temporary reproduction carried out during a "*data capture*" process fulfil the requirements of the exception for temporary copies (Art. 5(1) InfoSoc) under the cumulative conditions that those acts:

1. Must constitute an integral and essential part of a technological process. This condition is satisfied notwithstanding the fact that initiating and terminating that process involves human intervention;
2. Must pursue a sole purpose, namely to enable the lawful use of a protected work; and
3. Must not have an independent economic significance provided:
   (a) that the implementation of those acts does not enable the generation of an additional profit going beyond that derived from the lawful use of the protected work;
   (b) that the acts of temporary reproduction do not lead to a modification of that work (Case C-5/08 Infopaq I and C-302/10 Infopaq II).[19]

Under these conditions, temporary acts of reproduction are permitted by EU law.

A brief description of the facts of the Infopaq case may be helpful. The decision, referring to the compilation, extraction, indexing and printing of newspaper articles and keywords, identifies five phases relevant in the process of

[16] https://github.com/UniversalDependencies/UD_English

[17] https://catalog.ldc.upenn.edu/LDC2012T13

[18] Directive 2001/29/EC, OJ L 167, 22.6.2001

[19] Judgement of 16 Jul 2009, Infopaq International A/S v Danske Dagblades Forening, Case C-5/08 I, ECLI:EU:C:2009:465 and Judgement of 17 Jan 2012, Infopaq International A/S v Danske Dagblades Forening, C-302/10, ECLI:EU:C:2012:16

data capture: (1) newspaper publications are registered manually in an electronic registration database; (2) sections of the publications are selectively scanned, allowing the creation of a Tagged Image File Format (TIFF) file for each page of the publication and transferring it to an Optical Character Recognition (OCR) server; (3) the OCR server processes this TIFF file digitally and translates the image of each letter into a character code recognisable by the computer and all data are saved as a text file, while the TIFF file is then deleted; (4) the text file is processed to find a search word defined beforehand, identifying possible matches and capturing five words before and after the search word (i.e. a snippet of 11 words, before the text file is deleted; (5) at the end of the data capture process, a cover sheet is printed out containing all the matching pages as well as the text snippets extracted from these pages.

The Court found that the exception of Art. 5(1), which covers acts of *temporary* reproduction, only exempts the activities listed in points 1) to 4) above, whereas the activity of point 5), i.e. printing, constitutes a *permanent* act of reproduction which is therefore not covered by the exception for temporary copies.

It should further be noted that, in point 5), what is printed is not the entire literary text, but only 11 consecutive words. Only if these 11 consecutive words constitute a "*reproduction in part*" of the original work, copyright would be infringed. In this regard, the EUCJ found that "*it cannot be excluded*" that 11 consecutive words constitute the author's own intellectual creation and therefore represent a partial (and thus infringing) permanent reproduction. The 11 words threshold should not be taken as a strict parameter. The real test is that of the author's own intellectual creation. Accordingly, there will be shorter extracts that meet such a condition, and longer extracts that do not meet it.

As a result, it can be argued that current EU law permits the temporary copy of non-licensed copyright works for purposes such as "*data capturing*" as long as the cumulative conditions of Art. 5(1) as interpreted by the Court are met. However, when the results of the data capturing process leads to the *permanent* reproduction of the author's own intellectual creation this constitutes an infringement of the right of reproduction.

Nevertheless, it must be stressed that the conditions of Art. 5(1) are not only cumulative (i.e. all must be met) but also partially unclear (especially regarding the exact meaning of "independent economic significance") and must be interpreted *strictly*. These considerations have led many commentators to the conclusions that Art. 5(1) is not suitable as a general solution for TDM purposes (Triaille et al., 2014). This conclusion is certainly correct, nevertheless, until when a proper TDM exception is introduced at the EU level, the suitability of Art. 5(1) for unlicensed corpora should be explored further for specific ML/NLP cases, as in the present scenario. The remainder of this section will attempt such an exploration.

It seems that the the ML/NLP steps described in scenario three are substantially similar to those described by the EUCJ in the reported case law. In particular:

1. transforming the text corpus and the annotations into the input format of the Stanford NER tool is arguably equivalent to converting a TIFF image into text using OCR but much less sophisticated;

2. inspecting each word in the text in turn in order to create a ML feature representation capturing from the word and its immediate left and right neighbours, and from the annotation on the word is arguably equivalent to extracting the search term and the words before and after it, although extracting only one word before/after instead of 5, for a total of 3 words instead of 11 words extracted;

3. creating a probabilistic report (i.e. model) about the data obtained in this way is arguably equivalent to printing a cover sheet containing the matching pages, although the report consists of a numeric matrix encoding probabilities of observed features and correlations, as well as a feature and label dictionary containing encoded features covering words, word pairs, parts of words, and word shapes observed in the text, and the labels.

It seems plausible that the temporary copies created in points 1. and 2. are transient or incidental if they are only kept for the amount of time justified by the proper completion of the technological process and are automatically destroyed at the and of the process. It is also arguable that the act of reproduction is an integral and essential part of a technological process (the conversion of the text into data) which is necessary to enable a lawful use (statistical analysis is arguably as lawful as the preparation of summaries and is not a right reserved to the right holder by EU copyright law, however if the right holder contractually limits this operation and domestic law allows it, probably this condition would not be met). The requirement of absence of independent economic significance is probably harder to assess. Independent economic significance is present if the author of the reproduction is likely to make a profit out of the economic exploitation of the temporary copy. This profit has to be distinct from the efficiency gains that the technological process allows (see Infopaq II, 51).

Point 3. above refers to the results of the training process (the creation of a model) which are *permanent* by definition. Therefore, point 3. cannot be exempted on the basis of acts of *temporary* reproduction. It must be assessed however, whether the model constitutes a "*reproduction in part*" within the meaning of Art. 2 InfoSoc. If it does not, there is simply no copyright relevant activity and thus no need to rely on an exception.

In the present scenario, the trained model contains three consecutive words of the original "*all rights reserved*" corpora. While the test to be applied is not 11 vs. 3 consecutive words, but that of the "*author's own intellectual creation*", it seems plausible that three consecutive words are too insubstantial to constitute a "*reproduction in part*" of the original corpora. Therefore, the trained model does not reproduce in part the original corpora.

In conclusion, it can be argued that Art. 5(1) has the potential to be useful when the technological process is similar to

the one described in this scenario. However, given the cumulative, strict and partially unclear conditions that qualify it, a very careful case-by-case assessment should be performed before deciding to rely on this exception given the unavoidable degree of risk involved.

Are the results a derivative work? The CJEU in Allposters[20] clarified that the right of adaptation has not been object of EU harmonisation. Nevertheless, it must be observed that cases where adaptation does not require the reproduction at least in part of the original work may be of difficult conceptualisation (illustratively, jurisdictions such as France and the Netherlands classify the right of adaptation as a type of reproduction). That said, there are situations where there is adaptation without reproduction. An obvious example is the translation of a literary work into a different language. Technically speaking, there is no direct reproduction of the sentences in the original language. Consequently, it should not come as a surprise that the right of translation was the first right to be included in the minimum standard of protection in the oldest copyright international treaty, the Berne Convention.[21] However, training a model does not seem to possess the characteristics be considered a translation. Therefore, excluding the cases when this activity constitutes a reproduction (see above), it should be ascertained whether and under which circumstances training a model creates an adaptation.

Definition of derivative works in legislation At the international level, the Berne Convention grants copyright protection to translations, adaptations, arrangements of music and other alterations without prejudice to the original work to which they refer.[22]

There is no explicit definition in the Convention of adaptation, arrangement or other alterations of a work, a definitory lacuna that has stimulated some debate over the possible meanings of their specific wording (cf. (Ricketson and Ginsburg, 2006), p. 480). As some have suggested ((Goldstein and Hugenholtz, 2001), p. 252) adaptation means recasting a work from one format to another, whereas arrangement means modification within the same format, while others have underlined how the effort of defining them may be even unnecessary, if the law treats them essentially the same in terms of protection ((Chow and Lee, 2006), p. 181).

Illustratively, the notion of derivative work is instead explicitly defined by US law, which under the Copyright Act (17 U.S. Code) at §106(2) regulates the exclusive right to prepare derivative works.

As already pointed out, EU law does not harmonise the right of adaptation (except in the case of software and databases), therefore a proper analysis should look at how this right is regulated at the national level, thereby introducing an additional layer of complexity especially for scientific initiatives which are often international.

The Court of Justice, in Allposters, avoided to define derivative works by substantially referring to the right of reproduction, purporting that a new work incorporating a pre-existing protected work is "*an alteration of the copy of the protected work, which provides a result closer to the original*" and so constitutes "*a new reproduction of that work*" that remains in the exclusive rights of its right holder.[23] Furthermore, what seems determinant is that the new work (often identified as secondary work) reproduces, adapts or alters what constitutes the intellectual creation of the pre-existing (primary) work adding however an "*authorial contribution*" (Margoni, 2014).

Given the lack of EU harmonisation for the right of adaptation, the analysis should focus on the domestic law of EU Member States (MS). This type of inquiry would need to be done with a depth of analysis in 28 jurisdictions that is not be possible in the present paper. Nevertheless, it seem arguable that, as opposed to the broader US notion of derivative works, the EU counterparts tend to define adaptation in a narrower way, "*even narrower than the original Berne formulation*" (Bently and Sherman, 2014), p. 170, ft 220. Domestic courts have held that adaptations must show "*some quality or character which the raw material did not possess and which differentiates the product from the raw material*" ((Bently and Sherman, 2014), pp. 112-113 and Interlego v. Tyco Industries 1989 AC 217), and that in order for the right to be infringed, the elaboration should reveal the pre-existing work in its own individuality (ex multis, Corte Cassazione, 29 maggio 2003, n. 8597).

In the light of the above elements, which (it must be restated) constitute a mere superficial exploration of EU MS domestic law orientation on adaptations, it seems at least arguable that when the elaboration (trained model) does not reproduce the original (corpora) nor reveals "*its individuality*", no infringement should be found. From a policy point of view, training a model should be considered a free use. Future work should concentrate on this aspect.

6. Conclusion

This paper underlines the complexities in the relationship between concerning copyright and science in the context of ML/NLP. The legal analysis has been based on three specific scenarios which are all evolving around the task of training models for NER from annotated texts. The same legal principles can be applied to training models for other ML/NLP tasks (e.g. POS tagging, etc.), but depending on the specific variables the conclusions may differ. The conclusions of the three case scenarios presented in the present paper can be summarised as follows. The use of corpora licensed under TM/NLP friendly licences such as CC BY 4.0 guarantees that activities such as model training are lawful. In the case when no TM/NLP friendly licences are present, the operation of certain exceptions to copyright (e.g. Art. 5(1) InfoSoc) can represent the only proper legal basis for proceeding with ML activities. Nevertheless, a considerable level of uncertainty surrounds the applicability of the exception for temporary uses of Art. 5(1) InfoSoc and a proper analysis of each case should be performed

[20] Judgement of 22 January 2015, Art and Allposters International BV, Case C-419/13, ECLI:EU:C:2015:27

[21] Berne Convention for the Protection of Literary and Artistic Works, 1886

[22] Berne Convention for the Protection of Literary and Artistic Works, Art. 2, 8 and 12

[23] Judgement of 22 Jan 2015, Art and Allposters International BV, Case C-419/13, ECLI:EU:C:2015:27, paragraph 43.

before relying on it. Still, once the aspect of reproduction is properly addressed, we suggest refraining from defining training models in terms of derivative/adapted works, with the consequence that licensing restrictions (e.g. all rights reserved, ND or SA) imposed on the input training resources may not find application in the resulting output. At the same time, we acknowledge that the scope of TM/NLP is too broad to be handled homogeneously and that different types of algorithms and parametrisations require dedicated legal analysis, for example based on the level of abstraction they attain over the input data and the type of original material that is reproduced in the trained model.

Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation programme (H2020-EINFRA-2014-2) under grant agreement No. 654021 (OpenMinTeD). It reflects only the authors' views and the EU is not liable for any use that may be made of the information contained therein. It was further supported by the German Federal Ministry of Education and Research (BMBF) under the promotional reference 01UG1816B (CEDIFOR) and by the German Research Foundation as part of the RTG *Adaptive Preparation of Information from Heterogeneous Sources* (AIPHES) under grant No. GRK 1994/1. Additional thanks go to Mark Perry for review and comments.

Bibliographical References

Agić, v. and Ljubešić, N. (2014). The SETimes.HR Linguistically Annotated Corpus of Croatian. In Nicoletta Calzolari, et al., editors, *Proceedings of LREC 2014*, pages 1724–1727, Reykjavik, Iceland. ELRA.

Benikova, D., Biemann, C., and Reznicek, M. (2014). Nosta-d named entity annotation for german: Guidelines and dataset. In Nicoletta Calzolari, et al., editors, *Proceedings of LREC 2014*, Reykjavik, Iceland. ELRA.

Bently, L. and Sherman, B. (2014). *Intellectual Property Law*. Oxford University Press, 4th edition.

Borghi, M. and Karapapa, S. (2013). *Copyright and Mass Digitization: A cross-jurisdictional perspective*. Oxford University Press.

Chow, D. and Lee, E. (2006). *International Intellectual Property. Problems, Cases and Materials*. West Academic Publishing.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of ACL 2005*, pages 363–370, Stroudsburg, PA, USA. ACL.

Firdhous, M. (2012). Automating legal research through data mining. *Int. Journal of Advanced Computer Science and Applications*, 1(6):9–16.

Goldstein, P. and Hugenholtz, P. B. (2001). *International Copyright: Principles, Law, and Practice*. Oxford University Press.

Lucie Guibault et al., editors. (2013). *Safe to be open. Study on the protection of research data and recommendations for access and usage*. Universitätsverlag Göttingen.

Handke, C., Guibault, L., and Vallbé, J.-J. (2015). Is Europe Falling Behind? Copyright's Impact on Data Mining in Academic Research. In Birgit Schmidt et al., editors, *New Avenues for Electronic Publishing in the Age of Infinite Collections and Citizen Science: Scale, Openness and Trust – Proceedings of Elpub 2015*, pages 120–130.

Hugenholtz, P., (2016). *Something Completely Different: Europe's Sui Generis Database Right*, volume 37 of *Information Law Series*, pages 205–222. Kluwer Law Int.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of ACL 2014: System Demonstrations*, pages 55–60, Baltimore, Maryland. ACL.

Margoni, T. (2014). The Digitisation of Cultural Heritage: Originality, Derivative Works & (Non) Original Photographs. *IVIR Institute for Inf. Law*, pages 18–25.

Margoni, T. (2016). The harmonisation of EU copyright law: The originality standard. In Mark Perry, editor, *Global Governance of Intellectual Property in the 21st Century*, pages 85–105. Springer, Switzerland.

Marimon, M., Bel, N., Fisas, B., Arias, B., Vázquez, S., Vivaldi, J., Morell, C., and Lorente, M. (2014). The IULA Spanish LSP Treebank. In Nicoletta Calzolari, et al., editors, *Proceedings of LREC 2014*, Reykjavik, Iceland. ELRA.

Payne, D. and Landry, B. J. L. (2012). A Composite Strategy for the Legal and Ethical Use of Data Mining. *Int. Journal of Management, Knowledge and Learning*, 1(1):27–43.

Ricketson, S. and Ginsburg, J. C. (2006). *International Copyright and Neighbouring Rights. The Berne Convention and Beyond*. Oxford University Press.

Schroeder, C. T. and Zeldes, A. (2016). Raiders of the lost corpus. *Digital Humanities Quarterly*, 10(2).

Irina Stamatoudi et al., editors. (2000). *Copyright in the New Digital Environment: The Need to Redesign Copyright*, volume 8 of *Perspectives on Intellectual Property Law*. Sweet & Maxwell.

Triaille, J.-P., de Meeûs d'Argenteuil, J., and de Francquen, A. (2014). Study on the legal framework of text and data mining (tdm). *Luxembourg: Publications Office*.

Truyens, M. and Van Eecke, P. (2014). Legal aspects of text mining. *Computer Law & Security Rev.*, 30(2):153–170.

Tsiavos, P., Piperidis, S., Gavrilidou, M., Labropoulou, P., and Patrikakos, T. (2014). *Legal framework of textual data processing for Machine Translation and Language Technology research and development activities*. QTLP report & wikibook.

Verspoor, K., Cohen, K. B., Lanfranchi, A., Warner, C., Johnson, H. L., Roeder, C., Choi, J. D., Funk, C., Malenkiy, Y., Eckert, M., Xue, N., Baumgartner, W. A., Bada, M., Palmer, M., and Hunter, L. E. (2012). A corpus of full-text journal articles is a robust evaluation tool for revealing differences in performance of biomedical natural language processing tools. *BMC Bioinformatics*, 13(1):207.

Zeldes, A. (2017). The GUM Corpus: Creating Multilayer Resources in the Classroom. *Lang. Resour. Eval.*, 51(3):581–612.

LRE Map, a Song of Resources and Evaluation

Riccardo Del Gratta, Sara Goggi, Gabriella Pardelli, Nicoletta Calzolari
Istituto di Linguistica Computazionale "A. Zampolli", CNR
Via Moruzzi 1, 56124 Pisa, Italy
{firstname.lastname}@ilc.cnr.it

Abstract

After 8 years we revisit the LRE Map of Language Resources, introduced at LREC 2010, to try to get a picture of the field and its evolution as reflected by the creation and use of Language Resources. The purpose of the Map was in fact "to shed light on the vast amount of resources that represent the background of the research presented at LREC". It also aimed at a "change of culture in the field, actively engaging each researcher in the documentation task about resources". The data analysed here have been provided by the authors of several conferences during the phase of submission of papers, and contain information about ca. 7500 resources. We analysed the LRE Map data from many different viewpoints and the paper reports on the global picture, on different trends emerging from the diachronic perspective and finally on some comparisons between the 2 major conferences present in the Map: LREC and COLING.

Keywords: LR Infratructure, Metadata, LR Documentation

1. The LRE Map and its data

1.1. The LRE Map: why?

Science is ever more driven by data and our field is not different. Natural Language Processing (NLP) is certainly a data-intensive field. The LRE Map of Language Resources[1] (data and tools) was an innovative instrument introduced in the LREC2010 conference (Calzolari et al., 2010) with the aim of monitoring and representing the wealth of data and technologies developed and used in the field.

We called it "Map" because we aimed at representing the relevant features of a large territory, also for the parts not represented in the official catalogues of major players of the field (ELRA, LDC, NICT, ACL, OLAC, LT World, etc.). But we had other purposes too: we wanted to draw attention to the importance of the Language Resources (LRs) that are behind many of our papers; we wanted to start giving the deserved recognition, as suggested by the FLaReNet Thematic Network recommendations (Soria et al., 2014), to the developers of Language Resources (LRs); and finally wanted to map the "use" of LRs, to understand the purposes of the developed LRs and how their intended usage changes over time.

The collaborative creation of the Map was undoubtedly critical: we conceived the Map as a means to influence a "change of culture" in our community, whereby everyone is asked to make a minimal effort to document the LRs that are used or created. By spreading the LR documentation effort across many people instead of leaving it only in the hands of the distribution centres, we also encourage awareness of the importance of metadata and proper documentation. Documenting a LR is the first step towards identifiability, which in its turn is the first step towards reproducibility.

We kept the requested information at a simple level, knowing that we had to compromise between richness of metadata and willingness of authors to fill them in.

With all these purposes in mind we thought we could exploit the great opportunity offered by LREC and the involvement of so many authors from so many countries, from different modalities and working in so many areas of NLP.

Afterwards the Map was used also in the framework of other major Conferences, in particular by COLING, and this provides another opportunity for useful comparisons. It was an innovative action of infrastructural nature, in the awareness that research is affected also by such activities.

1.2. The LRE Map: the current data

We provide here a general overview of the data collected so far. The total number of LRs described in the Map is 7453 (instances), collected from 17 different conferences[2] (some with workshops). The major conferences for which we regularly have data are LREC and COLING.

The set of metadata to be used for describing each LR is the following: *Resource Type, Name, Production Status, Use, Language, Modality, Availability, Size, License, Documentation, URL, Description*. This set is very basic and is compatible with the metadata of other major LR catalogues, such as ELRA, CLARIN, META-SHARE.

The Map contains information also on its contributors and their organisations/institutions, countries, etc. We do not analyse these data here (also because not normalised), but they could be the source of other types of interesting analyses in the future.

Let us see first how LRs are distributed along some of these dimensions (considering only the most frequent values). As concerns LR Types, they are 469 in total, this variety being due to the possibility for authors of inserting a free description of the LR type in addition to those provided by us.

[1] The LRE Map, currently being updated, is at `http://lremap.elra.info` or `http://www.resourcebook.eu`

[2] This is the list of all conferences: LREC 10-12-14-16, COLING 10-12-14-16, INTERSPEECH 11-13, IJCNLP 11, RANLP 11, ACLHLT 11, LTC 11, O-COCOSDA 11, NAACL 13, BioTxtM 14.

Current figures are reported in the following tables. Table 1 lists the most frequent LR Types; Table 2 the most frequent modalities where *Written* is by large the most frequent, which is obvious given the type of conferences (even if at LREC we aim to have all the modalities represented).

LR Type	Percentage
corpus	46.48
lexicon	11.2
tagger/parser	6.63
annotation tool	4.12
evaluation data	3.65
ontology	2.79
corpus tool	1.73
.....	....

Table 1: Percentage of the 7 most frequent Types

LR Modality	Percentage
written	72.32
speech	6.45
multimodal/multimedia	4.39
not applicable	4.05
speech/written	3.01
modality independent	1.01
sign language	0.95

Table 2: Percentage of Modalities

Table 3 displays the LR Availability, and we remark that less than 5% of the LRs are not available at all; Table 4 shows figures for the LR Production Status and hints that there is a similar distribution between already existing LRs and newly created ones.

LR Availability	Percentage
freely available	50.23
from owner	20.73
from data center(s)	8.65
not available	4.56

Table 3: Percentage of Availability

LR Status	Percentage
existing-used	41.07
newly created-in progress	24.04
newly created-finished	20.81
existing-updated	8.16

Table 4: Percentage of Resource Production Status

Table 5 provides the major Usages, which is an indicator of where the efforts of our community are mostly concentrated in these years.

LR Usage	Percentage
information extraction, information retrieval	10.08
machine translation, speechtospeech translation	8.39
parsing and tagging	4.96
language modelling	4.49
document classification, text categorisation	3.17
evaluation/validation	2.94
knowledge discovery/representation	2.9
corpus creation/annotation	2.87
acquisition	2.78
speech recognition/understanding	2.74
discourse	2.62
named entity recognition	2.56
.....	....

Table 5: Percentage of the 12 most frequent Usages

1.3. Metadata values and their normalisation

This section refers to LRE Map metadada normalisation as part of the curation process on the Map data (Del Gratta et al., 2014).

In the tables above we provided only the most frequent values, but there are many "long tails" with small numbers or single occurrences (hapax), due to the possibility for authors to introduce a personal value under "other". After the first LREC we started to normalise some of the most frequent occurrences of personal values and introduced some of them among the "suggested values".

Metadata normalisation is an important (and never ending) process to reduce redundancies and ensure better accuracy. Simple examples of the normalisation process of authors' metadata values are: the pre-processing of values to eliminate the possible sources of duplicate information by standardizing the different spelling of proper names, acronyms and other terms used by the authors; insertion of the ISO code for languages; solving acronyms.

The strategy we have followed to address the above issues is to carry on a normalisation process of the values provided by the authors and to link the original values to the normalised ones. For us it is important to keep also the original information provided by the authors because of the bottom-up nature of the LRE Map.

We must also underline that in the Map there are, obviously, many cases of reference to the same Language Resource. Differently from other catalogues, in the Map it is important to know how much a LR is used and cited; for this reason we allow multiple occurrences/descriptions of the same LR. Normalisation of LR Names is therefore important to allow for grouping the different mentions of the same LR under the same name. This way we know which are the most "used/cited" LRs.

2. Some trends

2.1. General evolution

After 8 years of use, it is interesting to see whether there has been an evolution in the creation and use of Language Resources: we introduce here some tables that show some variability in the use of different LRs along the years.

2.2. LR Types

Table A [3] gives the distribution per year of the most frequent LR Types, showing with arrows the trend over the previous year: corpora and evaluation data are constantly increasing while tagger/parser, annotation tool and ontology are significantly decreasing.
In addition Figure I-a plots the data distribution of Types with very different distribution over the years.[4]

2.3. LR Availability

Table B shows that free-availability of LRs has greatly increased during these years: this is an important remark that reflects how the field is evolving with respect to an increasing awareness of the value of data sharing and openness.

2.4. LR Usage

Table C shows some interesting differences in the usage of LRs along the years. Information Extraction/Information Retrieval and Machine Translation, that were constantly the first usages until 2014, go down remarkably in 2016. While very interestingly many more LRs are used for evaluation purposes. This is a clear sign of the increasing importance attributed to evaluation in our field.

2.5. LR Status

Table D shows that there is a tendency towards creating new LRs while the re-use of existing ones is slowly decreasing.

3. Multi-dimensional analysis

We can take into account combinations of different metadata to look at various correlations, with a multi-dimensional analysis. This analysis provides some of the most interesting results.

3.1. Correlation between Type and Status

We see here the correlation between LR Type and LR Status, i.e. which Types of LRs are mostly newly created or existing and used.
There are some LR Types that behave really different from the average trend: Evaluation data are much more Newly built, while exactly the contrary is for Tagger/Parser and for Named Entity Recognizer that are mostly Existing and re-used (see Table 6). There is apparently no longer a great need of developing new taggers/parses or NE recognisers. While, coherently with the observation just made above about the increasing importance of evaluation data, there is the need of more and new Evaluation data.

[3] Additional tables and figures can be found in the Appendix.

[4] Other plots on the LR Type data distribution are available at `http://www.resourcebook.eu/trends/lr_years.html`.

LR Type vs. Status	% Existing	% New
Total LRs	49.26	44.8
Evaluation Data	38.99	61.01
Tagger/Parser	78.41	21.58
Named Entity Recognizer	63.63	36.37

Table 6: Correlation between Type and Status

3.2. Correlation between Availability and Modality and Status

From Table E it appears very clearly that Written Language Resources are more freely available than Spoken and Multimodal LRs, while these are more frequently available from the owner with respect to the others. Spoken LRs are comparatively more distributed through data centers. And Spoken and Multimodal (and also Sign language) LRs are the most not-available. From these data the Written community seems to be more willing to go in the direction of openness. But we must observe that Spoken and Multimodal LRs are usually much more expensive to create.
Among the freely-available LRs the existing-used/updated ones are more numerous than the new ones. The new-in-progress are comparatively (and obviously) more available from owner and also the most frequent in the not-available (hopefully just not yet). Data centers clearly distribute mostly the existing-used LRs.

4. Comparison between LREC and COLING: some interesting differences

It is interesting to see which are the major features that characterise different conferences with respect to Language Resources. The Map helps us in this, in particular to look at the differences between LREC and COLING for which we collected information for the last 4 editions of both conferences (2010, 2012, 2014, 2016).
There is a clear difference with respect to LR Modality, as reflected in Figure 1: LREC has more participation of the Speech and Multimodal communities and therefore more Speech and Multimodal LRs. A typical feature of LREC is that the Sign Language community is well represented (and consequently the corresponding type of LRs), while these are completely absent from COLING.
Figure 3 shows the comparison of LR Usages: the three most frequent Usages are exactly the same, in the same order, with similar frequency. But others have different frequencies in the two conferences.
Interestingly different is the Status of LRs: at LREC there are many more new Language Resource while at COLING more existing ones (Figure 2). This is expected given the nature of the conferences and the specific focus of LREC on LRs.

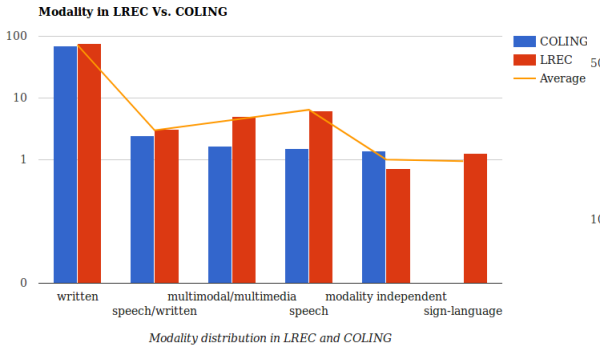


Figure 1: LR Modality LREC vs.COLING

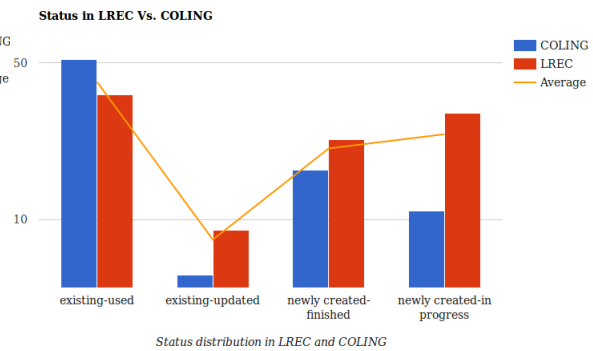


Figure 2: LR Status LREC vs.COLING

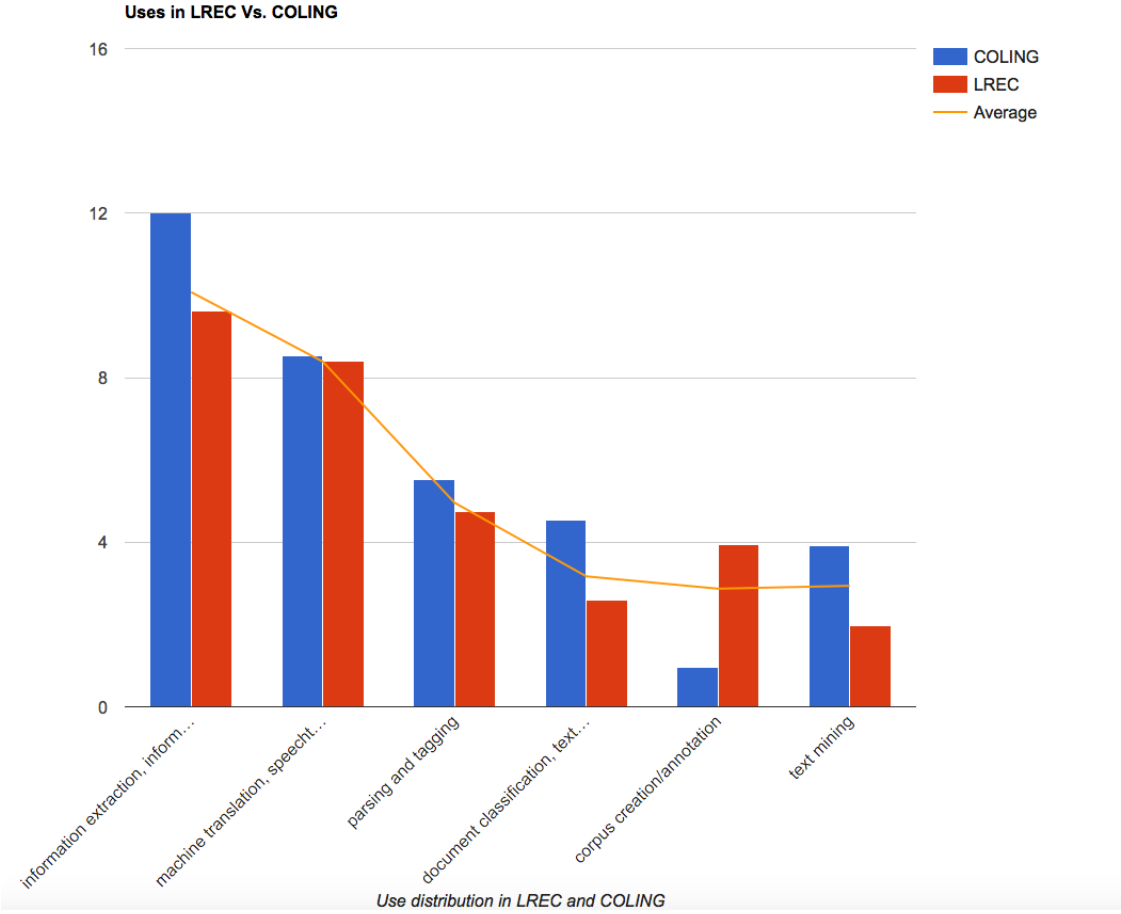


Figure 3: LR Use LREC vs.COLING

5. Conclusions

With initiatives such as the LRE Map and "Share your Language Resources" (introduced in 2014) we want to encourage in the field of Language Resources and Language Technology what is already in use in more mature disciplines, i.e. proper documentation and reproducibility as a normal practice. We think that research is strongly affected also by such infrastructural (meta-research) activities and therefore we continue to promote - also through such initiatives - a greater visibility of LRs, the sharing of LRs in an easier way and the reproducibility of research results.

Here is the vision: it must become common practice also in our field that when you submit a paper either to a conference or a journal you are offered the opportunity to document and upload the LRs related to your research. This is even more important in a data-intensive discipline like NLP. The small cost that each of us will pay to document, share, etc. should be paid back from benefiting of others' efforts.

6. Bibliographical References

Arranz, V., Choukri, K., Mapelli, V., and Mazo, H. (2014). ELRA's Consolidated Services for the HLT Community. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1511–1516, Reykjavik, Iceland. European Language Resources Association (ELRA).

Calzolari, N., Soria, C., Del Gratta , R., Goggi, S., Quochi, V., Russo, I., Choukri, K., Mariani, J., and Piperidis, S. (2010). The LREC Map of Language Resources and Technologies. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 949–956, Valletta, Malta. European Language Resources Association (ELRA).

Calzolari, N., Del Gratta, R., Francopoulo, G., Mariani, J., Rubino, F., Russo, I., and Soria, C. (2012). The LRE Map. Harmonising Community Descriptions of Resources. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1084–1089, Istanbul, Turkey. European Language Resources Association (ELRA).

Cieri, C., Choukri, K., Calzolari, N., Langendoen, D. T., Leveling, J., Palmer, M., Ide, N., and Pustejovsky, J. (2010). A Road Map for Interoperable Language Resource Metadata. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 2506–2509, Valletta, Malta. European Language Resources Association (ELRA).

Del Gratta, R., Pardelli, G., and Goggi, S. (2014). The LRE Map disclosed. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3534–3541, Reykjavik, Iceland. European Language Resources Association (ELRA).

Del Gratta, R., Frontini, F., Monachini, M., Pardelli, G., Russo, I., Bartolini, R., Khan, F., Soria, C., and Calzolari, N. (2016). LREC as a Graph: People and Resources in a Network. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2529–2532, Portorož, Slovenia. European Language Resources Association (ELRA).

Francopoulo, G., Mariani, J., and Paroubek, P. (2016). Predictive Modeling: Guessing the NLP Terms of Tomorrow. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 336–343, Portorož, Slovenia. European Language Resources Association (ELRA).

Mapelli, V., Arranz, V., Carré, M., Mazo, H., Mostefa, D., and Choukri, K. (2012). ELRA in the heart of a cooperative HLT world. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 55–59, Istanbul, Turkey. European Language Resources Association (ELRA).

Mariani, J., Paroubek, P., Francopoulo, G., and Hamon, O. (2014). Rediscovering 15 Years of Discoveries in Language Resources and Evaluation: The LREC Anthology Analysis. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4632–4669, Reykjavik, Iceland. European Language Resources Association (ELRA).

Mariani, J., Paroubek, P., Francopoulo, G., and Hamon, O. (2016). Rediscovering 15 + 2 years of Discoveries in Language Resources and Evaluation. *Language Resources and Evaluation*, 50(2):165–220.

Soria, C., Calzolari, N., Monachini, M., Quochi, V., Bel, N., Choukri, K., Mariani, J., Odijk, J., and Piperidis, S. (2014). The Language Resources Strategic Agenda: the FLaReNet synthesis of community recommendations. *Language Resources and Evaluation*, 48(4):753–775.

Wikipedia. (2017). LRE Map — Wikipedia, The Free Encyclopedia. [Online; accessed 2-October-2017].

7. Language Resource References

Nicoletta Calzolari and Claudia Soria and Riccardo Del Gratta and Sara Goggi and Valeria Quochi and Irene Russo and Khalid Choukri and Joseph Mariani and Stelios Piperidis. (2010). *The LRE Map*. ELRA/ILC.

Appendix: Tables and Figures

LR Type	%2010	%2012	%2014	%2016
corpus	41.37	46.16 (↑)	48.9 (↑)	49.12 (↔)
lexicon	11.5	11.86 (↔)	11.8 (↔)	10.89 (↓)
tagger/parser	9.72	5.78 (↓)	3.74 (↓)	4.47 (↑)
annotation tool	5.29	4.33 (↓)	3.90 (↓)	2.71 (↓)
evaluation data	3.36	3.80 (↔)	2.72 (↓)	4.41 (↑)
ontology	3.36	3.12 (↔)	3.14 (↔)	1.70 (↓)
...	...	...	...	...

Table A: Percentage of the first 6 LR Types with arrows showing the trend wrt previous year

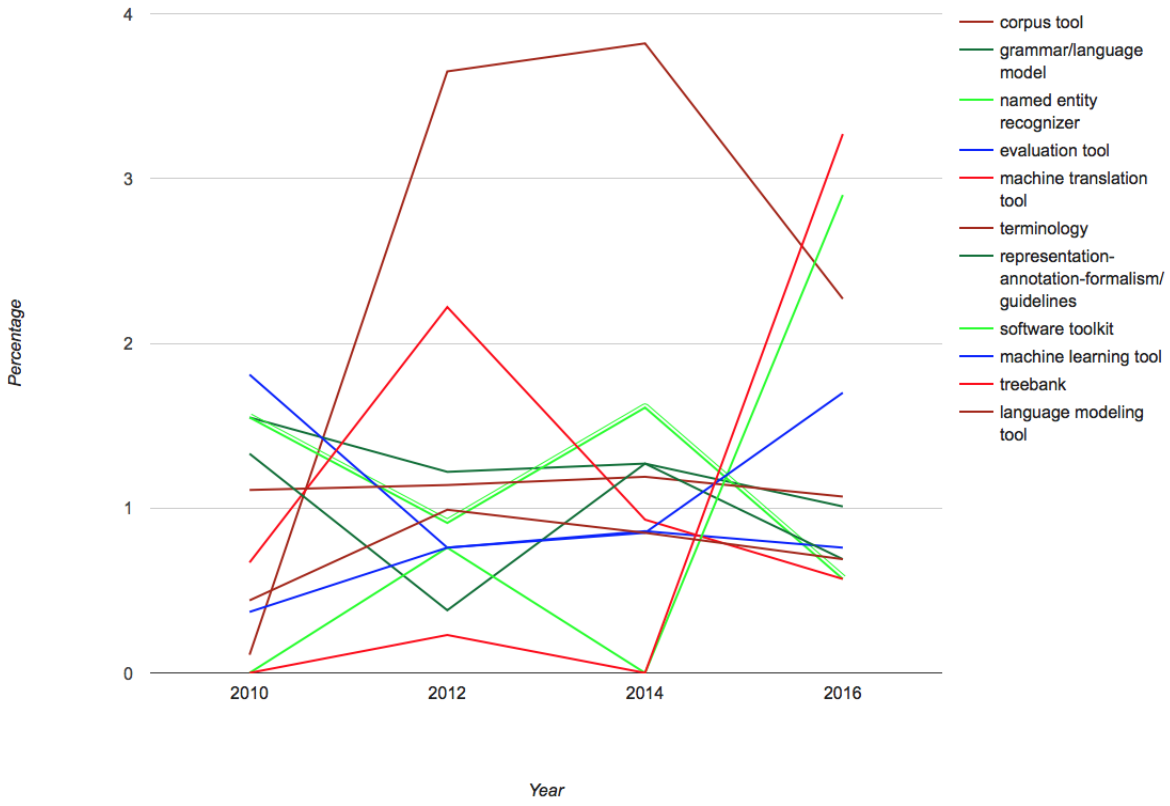


Figure I-a: Plot of volatile data

LR Availability	%2010	%2012	%2014	%2016
freely available	44.88	52.7 (↑)	55.4 (↑)	57.3 (↑)
from owner	22.22	19.77 (↔)	20.88 (↑)	19.71 (↓)
from data center(s)	9.09	8.21 (↓)	6.03 (↓)	7.3 (↑)
not available	5.73	3.95 (↓)	3.99 (↔)	3.21 (↔)

Table B: Percentage of Availability with arrows showing the trend wrt previous year

LR Usage	%2010	%2012	%2014	%2016
information extraction, information retrieval	11.61	12.09 (↔)	10.27 (↓)	6.3 (↓)
evaluation/validation	1.37	1.6 (↔)	1.53 (↔)	8.63 (↑)
machine translation, speechtospeech translation	9.17	10.49 (↑)	8.57 (↓)	5.35 (↓)
knowledge discovery/ representation	4.25	2.05 (↓)	2.04 (↔)	2.39 (↔)
...	...	...	...	...

Table C: Percentage of main Usages with arrows showing the trend wrt previous year

LR Status	%2010	%2012	%2014	%2016
existing-used	40.06	39.85 (↔)	27.42(↓)	34.89 (↑)
newly created-in progress	24.33	23.19 (↓)	31.92 (↑)	22.61 (↓)
newly created-finished	13.12	21.98 (↑)	26.99 (↑)	29.79 (↑)
existing-updated	8.17	8.52 (↔)	9.68 (↑)	6.86 (↓)

Table D: Percentage of Status with arrows showing the trend wrt previous year

Availability	Status (%)			
	Existing-used	Existing-updated	Newly created-finished	Newly created-in progress
Freely Available	61.06	64.96	57.74	48.76
From Owner	20.56	23.23	29.58	33.29
From Data Center(s)	15, 93	7, 68	7, 4	5, 71
Not Available	2, 45	4, 13	5, 28	12, 24

Table E: Availability Vs. Status

Availability	Modality (%)					
	Written	Speech/ Written	Speech	Multimodal/ Multime-dia	Sign Language	Modality Indepen-dent
Freely Available	60.18	48.07	30.92	41.04	40.00	77.36
From Owner	24.57	29.83	36.16	36.94	36.36	20.75
From Data Center(s)	10.34	16.57	21.7	9.7	5.45	1.89
Not Available	4.91	5.52	11.22	12.31	18.18	

Table F: Availability Vs. Modality

Metadata Collection Records for Language Resources

Henk van den Heuvel*°, Erwin Komen°, Nelleke Oostdijk*
*CLST / CLS, Radboud University
°Humanities Lab, Radboud University
Erasmusplein 1, Nijmegen, the Netherlands
{h.vandenheuvel, e.komen, n.oostdijk}@let.ru.nl

Abstract

In this paper we motivate the need for introducing more elaborate and consistent metadata records for collections of (linguistic) data resources in the CLARIN context. For this purpose we designed and implemented a CMDI profile. We validated the profile in a first pilot in which we populated the profile for 45 Dutch language resources. Given the complexity of the profile and special purpose requirements we developed our own interface for creating, editing, listing, copying and exporting descriptions of metadata collection records. The requirements for this interface and its implementation are described.

Keywords: LR infrastructure; metadata; collection records; Collection Bank

1. Introduction

In the context of CLARIN (Common Language Resources Infrastructure)[1] various stakeholders have been working towards the realisation of an integrated, interoperable research infrastructure. Apart from the technological achievements and the services offered, key factors to the success of this infrastructure are the availability and accessibility of the language resources. Ideally the infrastructure is populated with as many resources as possible. Therefore continued efforts are being put into gathering what resources there are and curating them, targeting also the many resources that have been created in the context of research projects but have not (yet) found their way to one of the data centres or repositories.[2]

In order to provide an entry-point to the language resources available in the CLARIN infrastructure, the Virtual Language Observatory (VLO) was developed (Van Uytvank 2014; Van Uytvank et al. 2010). The VLO offers a faceted browser which allows users to search for resources through their metadata. User experiences with the VLO have shown that discovering resources, and especially resources of which one is not aware they exist, is problematic. From the analysis of Odijk (2014) we know that while some of the problems arise from the current limitations of the VLO, in many cases the cause of the problem lies in the nature of the metadata. The set of facets used for search in the VLO is (too) small, not all facets are relevant for the discovery of resources, and some facets are lacking and should be added.[3] In 2015 the CLARIAH Metadata Curation Taskforce[4] was charged with the task to come up with a profile for collection records that would support the search for and discovery of language (data) resources. Collections in this context are to be understood as creator or depositor defined aggregations of data that fulfil a certain purpose, They have been created explicitly so as to form a browsable and therefore manageable hierarchy (cf. Broeder et al. 2009: 51). An example of a collection is the Spoken Dutch Corpus (Oostdijk, 2000). It comprises some 800 hours of sound recordings along with various types of transcriptions and annotations, a lexicon and a number of frequency lists.

In the present paper we describe how we proceeded and arrived at a profile for collection records which we think overcomes most of the shortcomings that various profiles that have previously been used exhibited. Here we restrict ourselves to the metadata of (linguistic) data collections, thus excluding software and tools.

The structure of the remainder of this paper is as follows: In Section 2 we present the profile we developed for collection records. We motivate its design and the considerations underlying it. Then, in Section 3 we report on the pilot that we conducted. In this pilot we applied the profile to a select but varied set of resources. Next, we introduce the Collection Bank, an interface that we developed for entering and maintaining metadata collection records. This paper concludes with a summary of the main outcomes, and suggestions for future work.

2. Profile for Collection Records

The development of the profile was guided by a number of desiderata that we derived from Odijk (2014). Odijk found that

(a) often metadata elements that were crucial for the discovery of a resource were lacking as they were not mandatory
(b) values for several important metadata elements are not restricted to a closed set
(c) the metadata created by various researchers and research groups often display what he calls 'unnecessary differences'
(d) the granularity of the metadata records varies wildly and is often too small.

[1] https://www.clarin.eu/

[2] In the Netherlands the Data Curation Service was set up as a centre of expertise to assist researchers in preparing their data for delivery to one of the CLARIN centres (van den Heuvel et al. 2015).

[3] Lušicky and Wissik (2017) also note the need for additional metadata. while using the VLO for discovering resources in the field of translation studies and urge other user groups to assess the VLO from their perspective and specify what should be added to satisfy their needs.

[4] In the Dutch CLARIAH project (https://www.clariah.nl/en/), the Metadata Curation Taskforce is concerned with providing metadata for various resources.

From this we concluded that we needed to establish what metadata elements are essential for search and discovery of resources, that such elements should be mandatory and to the extent possible, should have values from a closed/controlled vocabulary.

We started out by making an inventory of CLARIN-NL and CLARIN-EU collections represented in the CLAPOP[5], VLO[6], and EASY[7], and the information that was available in the form of collection records as well as any other information that might be considered relevant. It appeared that the information varied widely as can be seen from the comparison between CLAPOP, VLO and EASY in Table 1.

	CLAPOP	VLO	EASY
Title[8]	+	+	+
Research domain[9]	+		+
Annotations	+		
Format	+	+	
Resource tags	+		
Language	+	+	
Clarin centre[10]	+	+	
Country	+	+	
Resource type		+	
Availability		+	
National project		+	
Modality		+	
Organization		+	
Keyword		+	
Creator			+
Description			+
Subject			+
Coverage			+
Identifier			+

Table 1: Metadata available for resource discovery

We also looked at what profiles occurred in the CLARIN Component Registry. Here again there was a great deal of variation as most profiles appear to be collection-specific by design.

In developing our profile for collection records we opted to use the Dublin Core Metadata Element Set as a kernel similarly to what Bird and Simons (2003) did for OLAC. This decision was motivated by the fact that Dublin Core (DC)[11] is well-established: over the years it has been widely adopted and has proven to be usable for describing a wide range of resources. More specifically, it provides explicit definitions for each of the elements it contains. This, we expect, will contribute to the standardization we aim for. An illustration of where we deviate from DC is the distinction we make between the *type* and *subtype* of a resource. Thus in line with DC, the element *type* is defined as the nature of a resource, and follows the attributes defined in DC.[12] The attribute values associated with it, viz. 'collection', 'dataset', 'image', 'sound' and 'text' form a closed set. However, where in DC for example 'dataset' refers to any data encoded in a defined structure, our *subtype* for 'dataset' opens up a new closed set of attributes which distinguishes between 'list'[13], 'table', 'lexicon' and 'treebank'.

Since we were working in the CLARIN context, we found it opportune to select and copy the building blocks for our metadata collection profile from CLARIN's component metadata.[14] Important profiles from which blocks were copied (and pruned and modified where needed) were: OralHistoryInterviews, SpeechCorpus, and textCorpusProfile.

From the start we were aware that the collection records were to be included in an interface where they could be searched. In determining the relevance of various metadata elements we made a clear distinction between those metadata elements that are relevant as search or filter criteria and those that are only informative. The elements we deemed most relevant for search and filtering purposes are: *title*, *type*, *modality*, *annotation type*, *temporal and geographical provenance*, and *language*.

Some of the metadata have a range of fixed values (closed sets). The permitted values are given in CLARIN's Concept Registry[15] but, in CLARIN context, these are not (or rather: no longer) considered restrictive. Consequently, we added extra values for some of the metadata elements. This we did, for example, for *language* (where we needed a distinction for Northern and Southern Dutch (=Flemish)), and also for *annotation format*, *annotation type*, and *genre*.

Whenever available, the metadata elements in the profile have links to CLARIN's Concept Registry.

Relations between collections are formulated via a specific provision since the relation category is not well supported in the Concept Registry and would lead to somewhat forced CMDI constructs. Moreover, many of the attributes available for existing types of relationship in DC and DataCite[16] do not cover the relations that we would like to describe such as *isSiblingOf*, *inRepository*, *hasSubset*. These are quite typical relations between resources, but these attributes are not available in DataCite and DC. These relations have therefore been provided in a separate text file where they are formulated as RDF tuples. This file is referred to in the CMDI metadata file.

The relations are embedded in the resource proxy list of the CMDI file and are illustrated in Figure 1 for one of the collections (the CGN 1.0). Furthermore we added metadata elements for other resource proxies such as search pages (URLs providing a search interface for a resource) and landing pages (URLs providing basic information about a resource). These are also shown in Figure 1.

[5] http://dev.clarin.nl/clarin-data-list-fs
[6] https://vlo.clarin.eu/
[7] https://easy.dans.knaw.nl/ui/?wicket:interface=:3::::
[8] In CLAPOP it is not possible to filter on title. VLO here uses 'collection'.
[9] In EASY 'audience' is used.
[10] In VLO 'data provider' is used.
[11] http://dublincore.org/
[12] http://dublincore.org/documents/dces/
[13] With 'list' also the type of list (frequency or other) and type of list items (word type, lemma, POS tag, n-gram, phrase type, sentence type, other) are associated as attributes.
[14] See https://www.clarin.eu/content/component-metadata and https://catalog.clarin.eu/ds/ComponentRegistry
[15] https://www.clarin.eu/ccr
[16] https://www.datacite.org/

```
<Resources>
  <ResourceProxyList>
    <ResourceProxy id="lp_cbmetadata_00050">
      <ResourceType mimetype="application/x-http">LandingPage</ResourceType>
      <ResourceRef>https://hdl.handle.net/1839/00-0000-0000-0001-53A5-2@view</ResourceRef>
    </ResourceProxy>
    <ResourceProxy id="rel_isPreviousVersionOf_59">
      <ResourceType mimetype="text/plain">Resource</ResourceType>
      <ResourceRef>http://cls.ru.nl/registry/cbrelation_50_59.txt</ResourceRef>
    </ResourceProxy>
    <ResourceProxy id="sru_cbmetadata_00050">
      <ResourceType mimetype="application/sru+xml">SearchService</ResourceType>
      <ResourceRef>https://corpus1.mpi.nl/ds/trova/search.jsp?nodeid=MPI86949%23</ResourceRef>
    </ResourceProxy>
  </ResourceProxyList>
  <JournalFileProxyList/>
  <ResourceRelationList/>
</Resources>
```

Figure 1: The relation component embedded in the resource proxy list

The text file expressing the relations then looks like this:

```
Collection   Type of relation      Collection
CGN1.0       isPreviousVersionOf  CGN2.0
```

A complete profile for our metadata collection record is stored in CLARIN's Component Registry and can be viewed at https://catalog.clarin.eu/ds/ComponentRegistry/rest/registry/1.1/profiles/clarin.eu:cr1:p_1493735943947/xml.

3. Pilot data for collection records

As a proof of concept for the sustainability of the profile that we conceived we conducted a pilot in which we used the profile to create the metadata collection records for 45 language resources. These were selected on the basis of the following criteria. The resource was (a) a Dutch language resource, (b) considered relevant for current linguistic research, (c) referenced in CLARIN-NL's CLAPOP[17] and (d) contained in the VLO[18] and/or LINDAT[19] but underspecified at collection level. Moreover, it was required that sufficient metadata information sources were available to make a (more or less) complete collection record. An overview of collections involved in the pilot is shown in the Appendix.

After selection of the resources, the metadata for the individual resources had to be retrieved and entered into the collection records for each resource. This work was carried out by a student assistant who was provided with project websites where metadata information per resource could be retrieved. Her work was supervised by one of the authors of this paper who also added URLs for search pages, landing pages, and the links to other (versions of the) databases to the record.

The student assistant started out with one Excel file per language resource in which she collected the metadata. Due to the hierarchy in the components of the metadata this approach soon faced its limitations. Moreover, the information should not only be stored in Excel format, but also be made accessible in CMDI metadata files. Therefore, we decided to look for an interface which allowed editing hierarchical metadata in a user friendly way whilst providing a CMDI file export option as well.

4. An Interface for Entering Metadata Collection Records

CLARIN offers a versatile and well documented metadata editor for CMDI profiles: COMEDI[20] (Lyse et al., 2014). This tool takes a CMDI profile file as input and allows entering values for each metadata record contained in it. However, the tool is very general in its set-up, whereas what we needed was a more specific metadata editor for our collection records, allowing us to include

- clarifications per metadata category
- not yet existing components and metadata values
- a deeper hierarchical design of our components
- our own PIDs.

Moreover, edited metadata records should not be visible to everyone before being released. Based on these considerations we decided to develop an interface that would meet our requirements: the 'Collection Bank'.

4.1 The Collection Bank

The Collection Bank is a web application built on the Django framework.[21] The application facilitates creating, editing, listing, copying and exporting descriptions of metadata collection records.

The database model, which is hidden from the user, follows the 'CorpusCollection' specification of the model that is publically available in the CMDI Component Registry.[22,23] Saved collections can be 'published', which means that their *xml* representations become available through a persistent identifier. Saved collections become part of the user's list of collections (Collections > View).

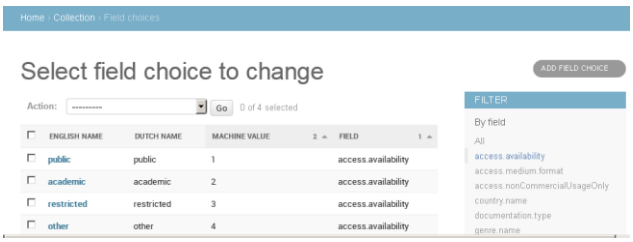


Figure 2: Administrator's interface to modify a fixed set

Changes in the definition of CorpusCollection in the CMDI Component Registry necessitate adapting the web interface. Changes in the fixed sets of metadata choices, for instance (e.g. *availability* can have the values 'academic', 'public', 'restricted' and 'other'), require the web interface's administrator to change the corresponding Field choices, as illustrated in Figure 2.

[17] http://portal.clarin.nl/clarin-data-list-fs
[18] https://vlo.clarin.eu/
[19] https://lindat.mff.cuni.cz/repository/xmlui/discover
[20] http://clarino.uib.no/comedi/
[21] See http://applejack.science.ru.nl/collbank. An account can be created through Extra > SignUp. The program heavily uses the existing facilities within Python's Django package.
[22] https://catalog.clarin.eu/ds/ComponentRegistry/
[23] CorpusCollection: clarin.eu:cr1:p_1493735943947

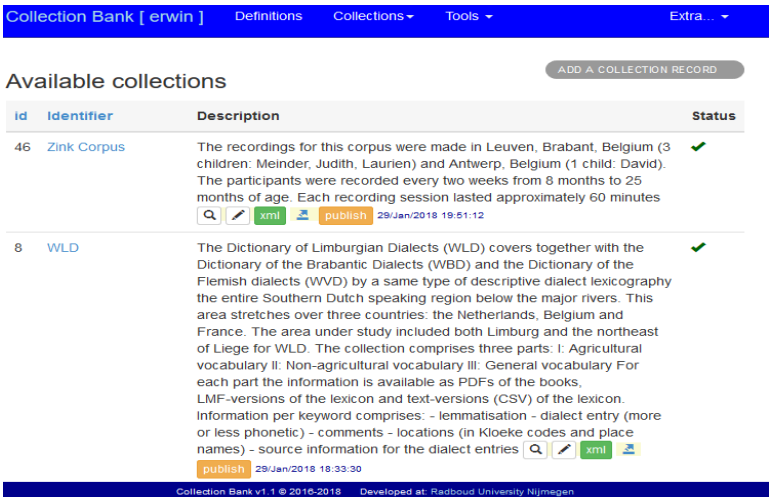


Figure 3: List-view of collections

The menu option 'list-view' of the collections (illustrated in Figure 3) provides an overview of all collections that have been entered. Each collection comes with its publication status and date (if published) and offers the user a number of action buttons: view, edit, download, copy and publish. The *view* button opens a human readable view of the collection, the *edit* button opens the editor (see below). The *download* button allows exporting a collection as a CMDI *xml* file (the Tools > Export commands facilitate exporting all the user's records together). The *copy* action allows copying whole metadata collection records (easing the work on similar collections). The last button *publish* creates the CMDI *xml* file, adding a persistent handle to it. All collections can be viewed and downloaded outside of the CollBank application through their persistent handle.[24]

Validation of the record's contents is done after creation of its CMDI *xml* representation; it is part of both the download as well as the publish actions. Validation is done through the *xsd* definition of a collection record, which is made available in the CLARIN components registry.

Upon creation of a record (Collections > Add), the collection *editor* opens up, and the user can specify the obligatory and optional parts of the metadata collection.[25]. Figure 4 (see next page) contains parts of the entry for the VU-DNC (Vis, 2011). The user needs to provide a named identifier for the collection (which is used for easy referencing within the web application). The 'description' field allows adding a description of any length to the collection as a whole. Note the clickable help-link below the text-input field ('See: Description…').

The VU-DNC collection contains two 'resources', which become visible on the same page by clicking the 'Show' buttons. The principle behind the Collbank web application is that the fields of one collection are all accessible on one web page. Figure 4 does not show the other editable collection fields, but it does contain the form's bottom row that holds the different save and delete options available to the collection entry as a whole.

5. Conclusion and Future Work

In this contribution we have provided the motivation for introducing more elaborate and consistent metadata records for collections of (linguistic) data resources. We have presented the CMDI profile which we developed for this purpose and presented the selection of collections used for populating the profile. Given the complexity of the profile and special purpose requirements we developed our own interface for creating, editing, listing, copying and exporting descriptions of metadata collection records.

As a follow-up of the work presented here, the metadata collection records will be made available and searchable in CLARIN's CLAPOP portal and in the VLO. The Collection Bank interface will be used to add further metadata collection records in the near future.

[24] Browsing to a persistant handle results in a text-oriented summary of the collection metadata, while the *xml* version of the record is returned in other situations. The specification of the VU-DNC collection, as an example, is available at http://cls.ru.nl/registry/cbmetadata_00016, or with its PID, at http://hdl.handle.net/21.11114/COLL-0000-000B-CAD5-1.

[25] Error handling, e.g. where obligatory fields are not filled in, follows the standards set by the Django framework.

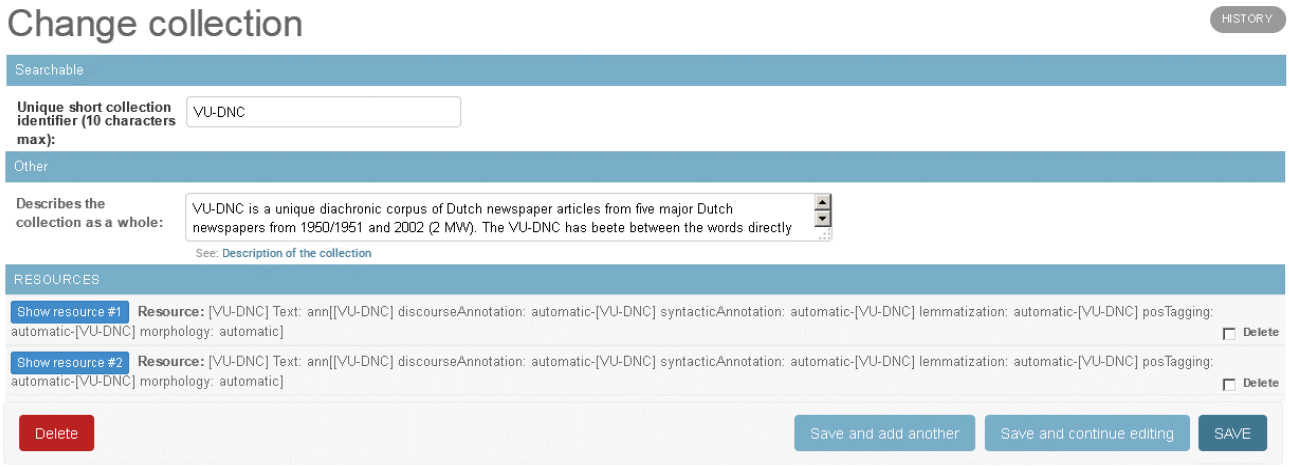


Figure 4: Specifying details of one collection

6. Acknowledgements

The work reported here was funded by the Dutch CLARIAH programmme[26] under project number CC-WP3-15-002. We are grateful to Hanna van den Heuvel for editing the metadata collection records in the project.

7. References

Bird, S. and Simons, G. (2003). Extending Dublin Core Metadata to Support the Description and Discovery of Language Resources. *Computers and the Humanities*, 37: 375-388.

Broeder, D., Gaiffe, B., Gravilidou, M. Lemnitzer, L. Van Uytvanck, D., Witt, A., and Wittenburg (2009). *Metadata Infrastructure for Language Resources and Technology*. 2009-02-02. Version 5. Deliverable D2.4 from EC FP7 project no. 212230. Retrieved from https://www.clarin.eu/sites/default/files/wg2-4-metadata-doc-v5.pdf

Heuvel, H. van den, Oostdijk, N., Sanders, E., and Lint, V. de (2015). Data curations by the Dutch Data Curation Service : Overview and future perspective. In *CLARIN 2014 Selected Papers; Linköping Electronic Conference Proceedings* # 116, pp. 54-62. http://www.ep.liu.se/ecp/116/005/ecp15116005.pdf

Lušicky, V. and Wissik, G. (2017). Discovering resources in the VLO: A pilot study with students of translation studies. *Selected papers from the CLARIN Annual Conference 2016. Linköping Electronic Proceedings* 136: 63-75.

Lyse, G.I., Meurer, P. and De Smet, K. (2014). COMEDI: A new component metadata editor. In *Proceedings of the CLARIN Annual Conference 2014*: https://www.clarin.eu/sites/default/files/cac2014_submission_13_0.pdf

Odijk, J. (2014). *Discovering resources in CLARIN: Problems and suggestions for solutions*. Working Paper. http://dspace.library.uu.nl/handle/1874/303788

Oostdijk, N. (2000). The Spoken Dutch Corpus. Overview and first evaluation. In *Proceedings of the second International Conference on Language Resources and Evaluation (LREC2000)*, http://www.lrec-conf.org/proceedings/lrec2000/pdf/110.pdf

Oostdijk, N. and H. van den Heuvel (2014). The evolving infrastructure for language resources and the role for data scientists. In *Proceedings of the ninth International Conference on Language Resources and Evaluation (LREC2014)*, pp. 608-612. http://www.lrec-conf.org/proceedings/lrec2014/index.html

Van Uytvanck, D. (2014). How can I find resources using CLARIN? Presentation held at the Using CLARIN for Digital Research tutorial workshop at *the 2014 Digital Humanities Conference, Lausanne, Switzerland*. https://www.clarin.eu/sites/default/files/CLARIN-dvu-dh2014_VLO.pdf. July 2014

Van Uytvanck, D., Zinn, C., Broeder, D., Wittenburg, P., and Gardellini, M. (2010). Virtual Language Observatory: The portal to the language resources and technology universe. In *Proceedings of the seventh International Conference on Language Resources and Evaluation (LREC2010)*, pp. 900-903. http://www.lrec-conf.org/proceedings/lrec2010/

Vis, K. (2011). Subjectivity in News Discourse. A corpus linguistic analysis of informalization. *PhD thesis*, VU Amsterdam. https://research.vu.nl/ws/portalfiles/portal/2925809

8. References to Language Resource Portals

[CLAPOP] re3data.org: CLAPOP; editing status 2017-01-30; re3data.org - Registry of Research Data Repositories. http://doi.org/10.17616/R35884

[Collection Bank] http://applejack.science.ru.nl/collbank

[LRT inventory] http://www.clarin.eu/view_resources and http://www.clarin.eu/view_tools

[VU-DNC] https://portal.clarin.inl.nl/vu-dnc/

[26] https://www.clariah.nl/en/

APPENDIX: Collections included in the pilot set

Name collection	Relevant information available from
Corpus Gesproken Nederlands (CGN)	http://lands.let.ru.nl/cgn/ https://corpus1.mpi.nl/ds/asv/?0&openpath=node:2102153 http://tst-centrale.org/nl/tst-materialen/corpora/corpus-gesproken-nederlands-detail
SoNaR	https://dev.clarin.nl/node/4195 http://link.springer.com/book/10.1007%2F978-3-642-30910-6 http://tst-centrale.org/nl/tst-materialen/corpora/sonar-corpus-detail
D-LUCEA	https://portal.clarin.nl/node/4183 https://corpus1.mpi.nl/ds/asv/?0&openpath=node:2102153# http://lucea.wp.hum.uu.nl/summary/ http://dev.clarin.nl/clarin-data-list-fs
LESLLA	https://corpus1.mpi.nl/ds/asv/?0&openpath=node:2102153# ! http://dev.clarin.nl/clarin-data-list-fs
Dictionary of the Brabantic Dialects, Dictionary of the Limburgian Dialects, Dictionaries of the dialects of Gelderland	http://www.lrec-conf.org/proceedings/lrec2016/pdf/223_Paper.pdf http://dialect.ruhosting.nl/wbd/index.htm http://dialect.ruhosting.nl/wld/index.htm http://dialect.ruhosting.nl/wgd/index.htm
MIMORE Data: DiDDD MIMORE Data: DynaSAND MIMORE Data: GTRP	http://portal.clarin.nl/node/4213
VALID	http://validdata.org/ especially: http://validdata.org/clarin-project/datasets/ https://corpus1.mpi.nl/ds/asv/?0&openpath=node:2102153 Papers: DOI:10.1075/dujal.3.2.02heu http://www.lrec-conf.org/proceedings/lrec2014/index.html at Authors van den Heuvel
VU-DNC	http://portal.clarin.nl/node/4194 https://portal.clarin.inl.nl/vu-dnc/
Academia Collectie (NIBG)	http://portal.clarin.nl/node/4230 https://www.academia.nl/ -> https://www.academia.nl/faq/28341 https://vlo.clarin.eu/search?0&fq=collection:Nederlands+Instituut+voor+Beeld+en+Geluid+Academia+collectie
DBD/TCULT	http://www.clarin.nl/sites/default/files/IDCC13-DCS_v4.2-final.pdf https://corpus1.mpi.nl/ds/asv/?openpath=node:84720 https://www.clarin.eu/sites/default/files/cac2014_submission_15_0.pdf
DiscAn	https://dev.clarin.nl/node/4198 https://tla.mpi.nl/resources/discan-corpora/ http://dx.doi.org/10.1016/j.dcm.2012.09.003 https://corpus1.mpi.nl/ds/asv/;jsessionid=9C64195BC53CEFED79D1B544E8822C23?0
DUELME Data	http://portal.clarin.nl/node/4200 http://duelme.clarin.inl.nl/ http://duelme.clarin.inl.nl/documentation.php https://vlo.clarin.eu/record?2&docId=hdl_58_10032_47_270633b99d34b5fc06b0699e8e2dd93c&fq=collection:TST-Centrale&index=1&count=2 Go to Advanced: Show all metadata fields
INTERVIEWS Data	https://easy.dans.knaw.nl/ui/datasets/id/easy-dataset:41923 https://dev.clarin.nl/node/4201
IPNV	https://easy.dans.knaw.nl/ui/datasets/id/easy-dataset:46232 https://www.clarin.eu/sites/default/files/cac2014_submission_15_0.pdf

LAISEANG	https://dev.clarin.nl/node/4197 https://corpus1.mpi.nl/ds/asv/;jsessionid=9C64195BC53CEFED79D1B544E8822C23?0
NEHOL	https://dev.clarin.nl/node/4193 https://corpus1.mpi.nl/ds/asv/;jsessionid=9C64195BC53CEFED79D1B544E8822C23?0 https://hdl.handle.net/1839/00-0000-0000-0016-83D9-D@view
ETC Database	https://portal.clarin.nl/node/4180 https://shebanq.ancient-data.org/sources https://easy.dans.knaw.nl/ui/datasets/id/easy-dataset:48490/tab/1 https://easy.dans.knaw.nl/ui/datasets/id/easy-dataset:58245/tab/1
Liederenbank	http://www.liederenbank.nl/index.php?lan=en https://nl.wikipedia.org/wiki/Nederlandse_Liederenbank
IFA Corpus = IFA speech corpus = IFA Spoken Language Corpus	https://vlo.clarin.eu/record;jsessionid=2E26AC7EC25FC5DDDA76EF19B781A537?1&docId=hdl_58_1839_47_00-0000-0000-0003-46DA-E&q=IFA+corpus&fq=languageCode:code:nld&index=0&count=5 http://www.fon.hum.uva.nl/IFA-SpokenLanguageCorpora/IFAcorpus/ https://corpus1.mpi.nl/ds/asv/?0
IFA Dialogue Video Corpus	https://vlo.clarin.eu/record?1&docId=http_58__47__47_hdl.handle.net_47_11372_47_LRT-576_64_format_61_cmdi&q=ifadvcorpus&index=0&count=2 http://www.fon.hum.uva.nl/IFA-SpokenLanguageCorpora/IFADVcorpus/
Corpus NGT (Nederlandse GebarenTaal)	http://www.ru.nl/corpusngtuk/ http://www.ru.nl/corpusngt/de_filmpjes/download-filmpje/ (licenses) https://corpus1.mpi.nl/ds/asv/?4&openpath=node:319374
CHILDES Dutch corpora	http://childes.psy.cmu.edu/ https://vlo.clarin.eu/record?2&docId=http_58__47__47_hdl.handle.net_47_11372_47_LRT-439_64_format_61_cmdi&q=childes&fq=languageCode:code:nld&index=0&count=3692 http://childes.talkbank.org/access/Dutch/
ESF Corpus	https://user.clarin.eu/resources/mpi-esf-corpus https://corpus1.mpi.nl/ds/asv/?4&openpath=node:319374

Managing Public Sector Data for Multilingual Applications Development

Stelios Piperidis, Penny Labropoulou, Miltos Deligiannis, Maria Giagkou
Athena RC/Institute for Language and Speech Processing
Epidavrou & Artemidos, Maroussi, Athens, Greece
spip@ilsp.gr, penny@ilsp.gr, mdel@ilsp.gr, mgiagkou@ilsp.gr

Abstract

The current paper outlines the ELRC-SHARE repository, an infrastructure designed and developed in the framework of the European Language Resource Coordination action with the objective to host, document, manage and appropriately distribute language resources pertinent to machine translation, and specifically tailored to the needs of the eTranslation service of the European Commission. Due to the scope of the eTranslation service which seeks to facilitate multilingual communication across public administrations in 30 European countries and to enable Europe-wide multilingual digital services, ELRC-SHARE demonstrates a number of characteristics in terms of its technical and functional parameters, as well as in terms of its data management and documentation layers. The paper elaborates on the repository technical characteristics, the underlying metadata schema, the different ways in which data and metadata can be provided, the user roles and their respective permissions on data management, and, finally, the extensions currently being implemented.

Keywords: public sector language resources, repository, documentation, machine translation

1. Introduction

Machine Translation (MT) technology can be applied to any language pair and adapted to specific domains and text types provided that sufficient amount of Language Resources (LRs) can be made available for training, adapting and evaluating the respective translation engines. The eTranslation digital service infrastructure[1] developed by the European Commission (EC), for example, needs to be adapted to meet the quality requirements of other Digital Service Infrastructures (DSI) like eJustice or Online Dispute Resolution, besides being extended to new language pairs. To meet the objectives of eTranslation adaptation, extension and overall improvement, the EC has launched the European Language Resources Coordination (ELRC) action (Lösch et al., 2018) with the mandate to collect and manage the appropriate LRs for all languages of the EU plus Norwegian and Icelandic, and for a number of domains that are relevant to other DSIs that the EC is building under the umbrella of the Connecting Europe Facility (CEF) programme[2]. Properly collecting, documenting and managing such data and making them available for developing MT engines calls for appropriate LRs management supported by a dedicated repository with the appropriate functionalities catering for the whole LRs lifecycle.

In this paper we focus on the repository infrastructure developed for this initiative, i.e. the ELRC-SHARE repository. Section 2 briefly introduces the ELRC action and provides an overview of the ELRC-SHARE technical characteristics, section 3 describes the underlying metadata schema used for LRs documentation, section 4 explains the different ways in which data, i.e. LRs and their metadata can be provided and managed, section 5 elaborates on the user management layer, while section 6 describes some of the extensions currently being built.

[1] https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation

[2] https://ec.europa.eu/digital-single-market/en/news/connecting-europe-facility-cef-digital-service-infrastructures

2. ELRC and ELRC-SHARE

2.1. About the ELRC action

Within this framework, the ELRC Network aims to raise awareness, promote uptake and foster the acquisition, identification and collection of LRs, targeting mainly at contributions of open language data from public administrations across the CEF countries. The resources identified, collected and made available as a result of the ELRC initiatives need to be:

- documented with the appropriate information describing the resource (aka metadata)
- easily uploaded and stored in a repository accessible by all relevant actors
- updated as necessary (both metadata & data)
- indexed and, as a result, accessed and downloaded (as necessary) according to the terms and conditions of their use.

The ELRC Network makes use of different channels to offer the above functionalities, the core of which is the ELRC-SHARE repository.

2.2. About the ELRC-SHARE repository

The ELRC-SHARE repository[3] is intended for managing LRs that are considered useful for feeding the CEF eTranslation digital service. It was originally designed to store, document and render accessible LRs, but its role has evolved in accordance with the requirements set for the management of LRs in the ELRC context. It, thus, currently aims to cover the whole lifecycle of LRs: uploading, documentation, uploading of accompanying documents, monitoring and reporting, updating, browsing, delivery and downloading.

The ELRC action is looking for public sector relevant open language data that can be made available for re-use

[3] https://elrc-share.eu/

based on the EU Public Sector Information (PSI) directive[4], but also for potentially restricted datasets (e.g. commercially available datasets, restricted resources pending negotiations/agreements/processing required for privacy protection, etc.). The LRs that ELRC targets can be any of the following:

- collections/sets of textual data ("corpora") in one or more languages, including for instance:
  - monolingual sets of public administration official documents (e.g. ministerial decisions, legal acts, Board decisions etc.), as well as sets of relevant journal, newspaper, bulletin, blog articles, etc.,
  - bi/multilingual parallel corpora, i.e. sets of original documents with their translations, or ready-made translation memories, i.e. translated documents aligned with their originals.
- lexical/conceptual resources, such as:
  - terminological lexica, glossaries etc., including lists of terms, with or without any other information (e.g. definition, examples, translation equivalents, linguistic information etc.)
  - lists of words, such as person names, names of places, names of products etc.
  - lexica and dictionaries containing words with linguistic information (e.g. part of speech, inflectional information, syntactic frames etc.)
- language descriptions that comprise, for instance, computational formal grammars (i.e. sets of rules that formalize a language), as well as language and translation models (i.e. resources which contain statistical information that assigns a probability to a piece of unseen text, based on some training data).

From a user perspective, ELRC-SHARE offers:

For data contributors: basic functionalities for contributing LRs through a very simple web form

For metadata editors (ELRC Network members): a user-friendly documentation environment for the description of resources (with the ELRC-SHARE metadata schema)

For the general public: a simple and faceted search and browsing of the resources inventory.

At the backend, ELRC-SHARE employs:

- a module for storing LRs, together with their respective metadata records and accompanying documentation (e.g. deposition and licensing documents, validation report);
- a user management module assigning specific access rights to the resources and the repository operations, depending on the user's role and the publication status of a resource;
- notification and reporting mechanisms for the efficient monitoring of updates of the hosted LRs.

ELRC-SHARE is based on a META-SHARE software instance, its latest version building on META-SHARE v3.1.1. The software has been adapted to the operational needs of ELRC and it has been evolving to respond to specific requirements of its stakeholders. The ELRC-SHARE repository can be replicated on additional servers. However, unlike the META-SHARE distributed network structure (Piperidis, 2012), ELRC-SHARE is deployed as a single repository (i.e. one central, managing, node in META-SHARE terminology) centrally managed by the ELRC consortium. Furthermore, following ELRC requirements, new user roles have been added (namely technical and legal reviewing roles with their own class of access rights), the metadata schema is updated to reflect, for instance, evolution in the open data licensing policies of countries, and new functionalities for reporting, data exporting and packaging have been implemented.

3. ELRC-SHARE metadata schema

ELRC LRs are (formally) documented using the ELRC-SHARE schema. In essence, the ELRC-SHARE metadata schema is an application profile of the META-SHARE schema appropriately modified to meet the requirements of ELRC.

META-SHARE (Gavrilidou et al., 2012) is a generic schema designed for the description of LRs in the wider area of Language Technology. It covers the description of *data* (textual, multimodal/multimedia and lexical data, grammars, language models, etc.) and *technologies (tools/services)* used for their processing. Its main features (that are also inherited to ELRC-SHARE) are:

- flexibility, empowered with a two-level approach, where the initial level (*minimal schema*) consists of a set of basic elements required for at least identifying and accessing a LR, and a second level (*maximal schema*) with a higher degree of granularity of information
- modularity, implemented in the form of *"components"*, i.e. groups of semantically coherent elements, following the Component MetaData Infrastructure (CMDI) recommendations (Broeder et al., 2008)
- standardisation: where possible, controlled vocabularies are preferred over free text for the value space of elements, especially when these can be associated with internationally acknowledged standards, best practices or widespread vocabularies (e.g. ISO 3166 for country codes, RFC 5646 for languages, IANA mimetypes etc.)
- interoperability with other related metadata schemas: links are provided to the conceptually same or similar elements mainly with Dublin Core (DC)[5] and the Data Catalog Vocabulary (DCAT)[6].

[4] https://ec.europa.eu/digital-single-market/en/european-legislation-reuse-public-sector-information

[5] http://www.dublincore.org

[6] https://www.w3.org/TR/vocab-dcat/

The elements are carefully selected to capture **properties** of the resources and **relations** with other resources that are deemed important for their description and relate to any of the stages of the **resource lifecycle**, from production to consumption. In particular, relations are encoded in the metadata records of all resources which have gone through some processing, conversion or transformation indicating the type of relation between the two, essentially different, versions of a resource; e.g. a raw parallel corpus that has undergone alignment gives rise to a new version of the resource described with a new metadata record; the two metadata records (of the original and the aligned version) are linked together with a relation of type "isAlignedVersionOf".

The ELRC-SHARE schema is at the same time a restriction and an extension of META-SHARE: the adaptations include the removal of a substantial set of elements/values as well as the addition of new elements/values. For instance, all elements appropriate for non textual resources have been pruned. On the other hand, the requirements of ELRC have given rise to new elements/values mainly in the categories of licensing and classification. For instance, the legal component was adapted to include

- values for the "licence" element that specify licences recommended for public sector open data
- a value that indicates that a resource falls under the PSI directive
- elements that declare whether a resource includes personal and sensitive data and, as a result, needs special handling and further processing.

The schema includes the following mandatory metadata categories:

- **Administrative information**: features important for the *identification* of a LR (e.g. resource name, a short description of its contents, etc.), its *distribution* (e.g. the access form, i.e. whether it can be downloaded or accessed through an interface and the licensing terms under which it can be used), *contact information* (data and email of a contact person), information on the *metadata record* (e.g. data of the metadata editor, creation and update dates etc.)
- **Technical information**: features on the *language* (number of languages, language identifier(s) and name(s)), *size* of the resource and the *format(s)* in which the resource is available (e.g. plain text, PDF, XML, TMX etc.), and *subtype*, the values of which are specific to each resource type (e.g. terminological glossary, dictionary etc. for lexical/conceptual resources vs. grammar or model for language descriptions).

In addition, the following metadata categories are optional:

- information on the creation of the resource (e.g. the resource creator, whether it involved an automatic process and, if yes, the tool(s) that were used, creation date etc.), links or bibliographic data of documents that describe the resource.
- domain and text type classification, character encoding, description of the types of linguistic or extra-linguistic information (e.g. lemmas, grammatical categories, translation equivalents etc.) contained in lexical/conceptual resources and language descriptions.

It should be noted here that, although optional, **domain**, and in many cases, text type **classification** are of utmost importance to the ELRC objectives. The Eurovoc thesaurus[7] and, more specifically, the two upper levels of the hierarchy have been selected to ensure uniform domain classification across the resources and alignment with the domain classification already employed by the Directorate-General for Translation of the EC and by other European institutions. Due to the special focus on the reuse of the eTranslation service by other CEF sector-specific DSIs, an additional metadata element has been introduced, that of relevance of the LR to a DSI, thus facilitating retrieval of resources that can be used for training machine translation engines tailored to the domain(s) of a specific DSI. Finally, for the text type classification, we have devised our own controlled vocabulary based on the types of texts that are typical of the Public Sector (e.g. administrative texts, bulletins, legal documents etc.) and the ones we expect to obtain from our sources.

4. Contributing, documenting and managing resources

Contributing resources through the ELRC-SHARE repository is a process that is kept as simple as possible given that the intended contributors are primarily Public Sector employees who are not necessarily familiar with the concepts of, and around, language resources.

Providing resources and the required documentation is possible only after users have registered to the system. They can then use a simple web form to describe and upload LRs. For each LR, at least a short description and a title in English must be provided; the resource itself can be uploaded in zipped format (currently up to 100MB) or, alternatively, users may supply a link to a URL where the resource can be downloaded from.

Following this submission, the contributed data and basic metadata are received by the repository administrators and are imported into the repository. For each new contribution, proper notification email alerts are sent to designated ELRC consortium members (metadata editors) who have the responsibility to enrich the documentation, check the resource and publish it. The ELRC member may contact the contributor to ask for further information about the resource and its licensing conditions. The documentation and checking are supported with an editor form that implements the full ELRC-SHARE schema. During the documentation process, the resource descriptions are considered "internal" and can be viewed only by authorised editors of the ELRC consortium. When the documentation and the licensing conditions under which the resource is provided are finalised, the resource description (metadata record) will be made public through the inventory and the resource itself will be downloadable by users, in

[7] http://eurovoc.europa.eu/

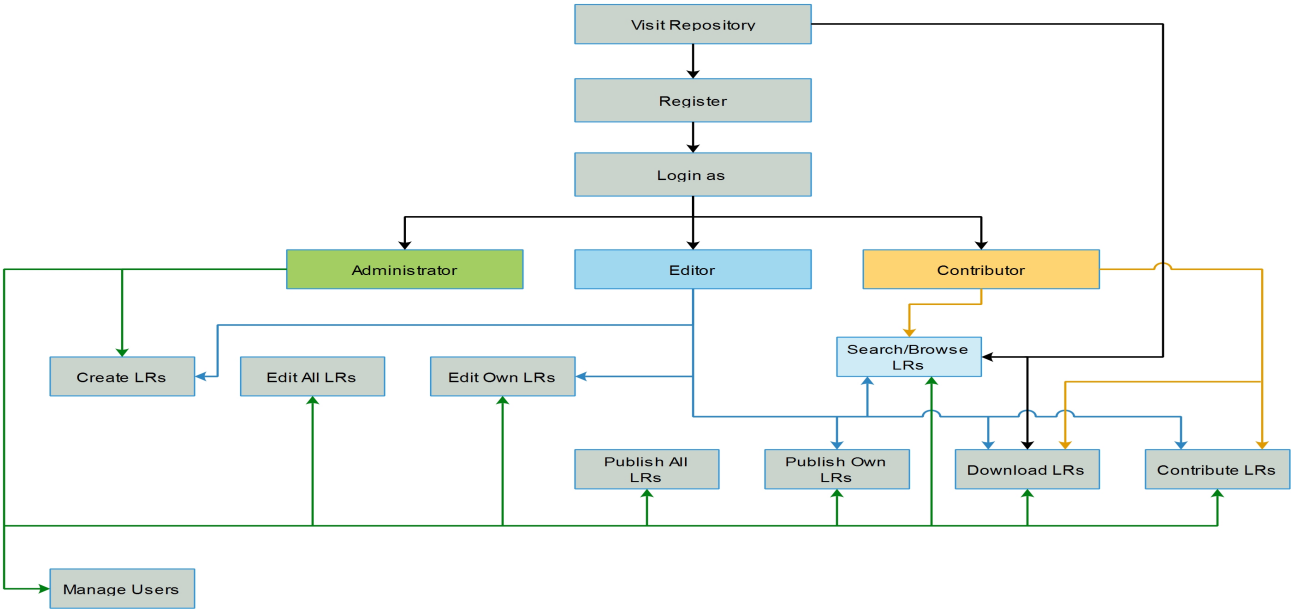


Figure 1: Available user rights per user role

accordance to its licensing conditions.

Contributions may also come from ELRC members using the editor interface, in which case the LRs are deposited straight into the repository, again as a zipped file. To upload the resource, the minimal information must have already been encoded for the resource. The description can still continue to be updated at any time, before and after uploading.

In addition, ELRC resources may be generated by using an automatic language resource discovery (Papavassiliou et al., 2018) and compilation system (ILSP-FC). ILSP-FC ((Papavassiliou et al., 2016); (Papavassiliou et al., 2013)) is essentially a pipeline of tools that, given a set of seed URLs, and optionally a domain profile in the form of a list of domain specific terms, crawl the web, fetch web pages, check their domainness, pair the translated web pages and align them at sentence level, thus generating a TMX file. The ILSP-FC system automatically extracts the required metadata and renders them in the ELRC-SHARE profile form, ready to be automatically uploaded to the repository. All LRs residing in the repository undergo a systematic validation procedure (Lösch et al., 2018), the results of which are coded in the respective validation reports stored together with any other ancillary documentation accompanying the resource.

The result of the documentation process is compiled into the ELRC-SHARE inventory of resources comprising all the descriptions (metadata catalogue).[8]

5. Managing access and editing rights

Access to ELRC-SHARE is regulated by user roles and permissions. Unregistered users can only browse and search the repository and view the resource descriptions. Registered users can be assigned one of the following roles, with respective rights and permissions: a) *contributor* (default), b) *editor*, c) *administrator*. Figure 1 illustrates the available user rights per role within the repository.

The resources hosted in the repository are assigned a Publication Status which determines the visibility of their descriptions to different user types. The Publication Statuses are:

- **internal**: used as the initial status for all metadata records;
- **ingested**: LRs with enriched and initially validated metadata records;
- **published**: LRs with finalised metadata records. Published records are available for searching and browsing on the public inventory of resources.

The combination of user access rights and resource status yields an enhanced user management module that gives resource owners extra leverage and control over the visibility and accessibility of the repository resources. Currently, the ELRC-SHARE inventory lists 225 published language resources, with 200 additional and/or derivative resources in ingested mode, i.e. waiting to be published. The evolution of the inventory is reported in the form of simple spreadsheets, while specific APIs for querying the metadata inventory are under implementation.

6. Extensions currently under way

In order to help public administrations across borders share electronic data and documents in a secure, reliable and trusted way, ELRC-SHARE is being endowed with an eDelivery[9] Access Point (AP), as an alternative to direct contribution and uploading. ELRC-SHARE will thus ensure that

[8]The inventory with the LR descriptions is licensed under the Creative Commons Attribution (CC-BY) version 4.0 or higher.

[9]https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eDelivery

any sensitive or confidential data transferred from public administrations into the repository are encrypted and secure.

ELRC-SHARE is deploying an eDelivery AP using an appropriately configured Domibus EC implementation, which is an ebMS3 AP based on the e-SENS AS4 Profile. Access Points can have a message "sender" or/and message "receiver" role. Since the ELRC-SHARE Access Point will be receiving data from public administrations, it will act as a receiver in the message exchange process, accepting messages and data from public administrations that already have an available sender Access Point.

Finally, ELRC-SHARE is currently being extended to allow for the cataloging and documentation of language processing tools and services, e.g. aligners, annotators, named entities recognizers and other tools pertinent to machine translation. In the future, ELRC-SHARE will support the direct processing of resources with available tools and services. LRs will be associated with adequately documented available tools/services (e.g. text normalization for encoding detection, UTF-8 conversion and metadata extraction, language identification, text classification, deduplication, translation pair detection, text alignment, etc.), which produce new resources that will be stored in the ELRC-SHARE repository as new versions or derivatives of the original resource.

7. Acknowledgements

The European Language Resource Coordination (ELRC) is a service contract operating under the EC's Connecting Europe Facility SMART 2014/1074 programme from April 2015 to April 2017 and continues until December 2019 under SMART 2015/1091 LOT 2 "Language Resource coordination and collection with related legal and technical work" and SMART 2015/1091 LOT 3 "Acquisition of additional Language Resources and related refinement/processing services and their provision of the Language Resource Repository of CEF Automated Translation Platform". We would like to thank all ELRC consortium members, DFKI, TILDE and ELDA, and the numerous contributors from the 30 CEF-affiliated countries for contributing their language resources.

8. Bibliographical References

Broeder, D., Declerck, T., Hinrichs, E., Piperidis, S., Romary, L., Calzolari, N., and Wittenburg, P. (2008). Foundation of a component-based flexible registry for language resources and technology. In Nicoletta Calzolari, et al., editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Gavrilidou, M., Labropoulou, P., Desipri, E., Piperidis, S., Papageorgiou, H., Monachini, M., Frontini, F., Declerck, T., Francopoulo, G., Arranz, V., and Mapelli, V. (2012). The META-SHARE Metadata Schema for the Description of Language Resources. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eigth International Conference on Language Resources and Evaluation (LREC'12)*, Paris, France, 5. European Language Resources Association (ELRA), ELRA.

Lösch, A., Mapelli, V., Piperidis, S., Vasiļjevs, A., Smal, L., Declerck, T., Schnur, E., Choukri, K., and van Genabith, J. (2018). European Language Resource Coordination: Collecting Language Resources for Public Sector Information Management. In *Proceedings of the 11th Language Resources and Evaluation Conference*. European Language Resources Association (ELRA).

Papavassiliou, V., Prokopidis, P., and Thurmair, G. (2013). A modular open-source focused crawler for mining monolingual and bilingual corpora from the web. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 43–51, Sofia, Bulgaria. Association for Computational Linguistics.

Papavassiliou, V., Prokopidis, P., and Piperidis, S. (2016). The ILSP/ARC submission to the WMT 2016 Bilingual Document Alignment Shared Task. In *Proceedings of the First Conference on Machine Translation*, pages 733–739, Berlin, Germany. Association for Computational Linguistics.

Papavassiliou, V., Prokopidis, P., and Piperidis, S. (2018). Discovering parallel language resources for training MT engines. In *Proceedings of the 11th Language Resources and Evaluation Conference*. European Language Resources Association (ELRA).

Piperidis, S. (2012). The META-SHARE Language Resources Sharing Infrastructure: Principles, Challenges, Solutions. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Bridging the LAPPS Grid and CLARIN

Erhard Hinrichs*, Nancy Ide, James Pustejovsky†, Jan Hajič‡, Marie Hinrichs*, Mohammad Fazleh Elahi*, Keith Suderman**, Marc Verhagen†, Kyeongmin Rim†, Pavel Straňák‡, Jozef Mišutka‡**

*University of Tübingen, **Vassar College, †Brandeis University, ‡Charles University

{erhard.hinrichs, marie.hinrichs, mohammad-fazleh.elahi}@uni-tuebingen.de, {ide, suderman}@cs.vassar.edu, {jamesp, marc}@cs.brandeis.edu, krim@brandeis.edu,{hajic, stranak, misutka}@ufal.mff.cuni.cz

Abstract

The LAPPS-CLARIN project is creating a "trust network" between the Language Applications (LAPPS) Grid and the WebLicht workflow engine hosted by the CLARIN-D Center in Tübingen. The project also includes integration of NLP services available from the LINDAT/CLARIN Center in Prague. The goal is to allow users on one side of the bridge to gain appropriately authenticated access to the other and enable seamless communication among tools and resources in both frameworks. The resulting "meta-framework" provides users across the globe with access to an unprecedented array of language processing facilities that cover multiple languages, tasks, and applications, all of which are fully interoperable.

Keywords: Language Applications Grid, WebLicht, Text Corpus Format (TCF), LAPPS Grid Interchange Format (LIF), CoNLL-U, Web Service Exchange Vocabulary, syntactic interoperability, semantic interoperability, user identification and authentication

1. Introduction

The Andrew K. Mellon Foundation has funded a project to create a "trust network" between the Language Applications (LAPPS) Grid (**?**), a major framework for composing pipelines of natural language processing (NLP) tools, and the WebLicht workflow engine (**?**) hosted by the CLARIN-D Center in Tübingen. The project also includes integration of NLP services available from the LINDAT/CLARIN Center in Prague[1]. The goal is to allow users on one side of the bridge to gain appropriately authenticated access to the other and enable seamless communication among tools and resources in both frameworks. The resulting "meta-framework" provides users across the globe with access to an unprecedented array of language processing facilities that cover multiple languages, tasks, and applications, all of which are fully interoperable.

The LAPPS Grid/CLARIN Mellon project involves two major tasks: (1) establishing a joint single sign-on user authentication and authorization mechanism; and (2) enabling seamless interoperability at both the syntactic and semantic levels among tools available from both the LAPPS Grid and WebLicht, so that users can mix and match these tools regardless of provenance and without concern for differing I/O requirements. In this paper we describe the work required to accomplish these tasks.

2. Overview

In the LAPPS Grid, language resources and NLP tools are made available as web services through the Galaxy workflow engine and interface (**?**), as well as programmatic access through the LAPPS Grid application programming interface (API)[2]. LAPPS Grid tools consume and produce data in the LAPPS Interchange Format (LIF) (**?**), a JSON-LD-based format designed to serve as an internal interchange format for linguistically annotated data. Semantic interoperability among services is accomplished via URI references to the LAPPS Grid Web Service Exchange Vocabulary (WSEV) (**?**). NLP tools are accessed as web services that deliver metadata about the content at a standardized URI and are at present invoked using the Simple Object Access Protocol (SOAP).

WebLicht is an environment for building, executing, and visualizing the results of NLP pipelines, which is integrated into the CLARIN infrastructure (**?**). WebLicht NLP tools are implemented as web services that consume and produce the Text Corpus Format (TCF)[3] data, an XML format designed for use as an internal data exchange format for WebLicht processing tools. The TCF also ensures semantic interoperability among all WebLicht tools and resources by defining a common vocabulary for linguistic concepts. Metadata descriptions of WebLicht tools are stored in repositories located at the CLARIN center hosting the service. WebLicht web services are invoked using the REpresentational State Transfer (RESTful) API.

LINDAT/CLARIN provides various NLP services[4] based on single-purpose tools or pre-configured chains of tools, often for multiple languages, most notably the UDPipe service (**?**). UDPipe produces CoNLL-U[5], the Universal Dependencies (UD) annotation format (**?**), which is a revised version of the CoNLL-X format (**?**) used in the Conference on Natural Language Learning exercises. Sub-chains of UDPipe are being exposed in WebLicht and will be made interoperable with WebLicht's TCF-based tools; conversion to LIF may then be accomplished by converting from TCF to LIF. UDPipe tools are also accessed from the LAPPS Grid via WebLicht.

The main challenges to bridging the LAPPS Grid and WebLicht frameworks arise from differences in the architectures of the two systems, in particular the differences in data exchange formats, access to and structure of metadata, and the protocols used to invoke web services. In addi-

[1] https://lindat.mff.cuni.cz/en

[2] http://wiki.lappsgrid.org/Developing.html

[3] https://weblicht.sfs.uni-tuebingen.de/weblichtwiki/index.php/The_TCF_Format

[4] http://lindat.mff.cuni.cz/services/

[5] http://universaldependencies.org/format.html

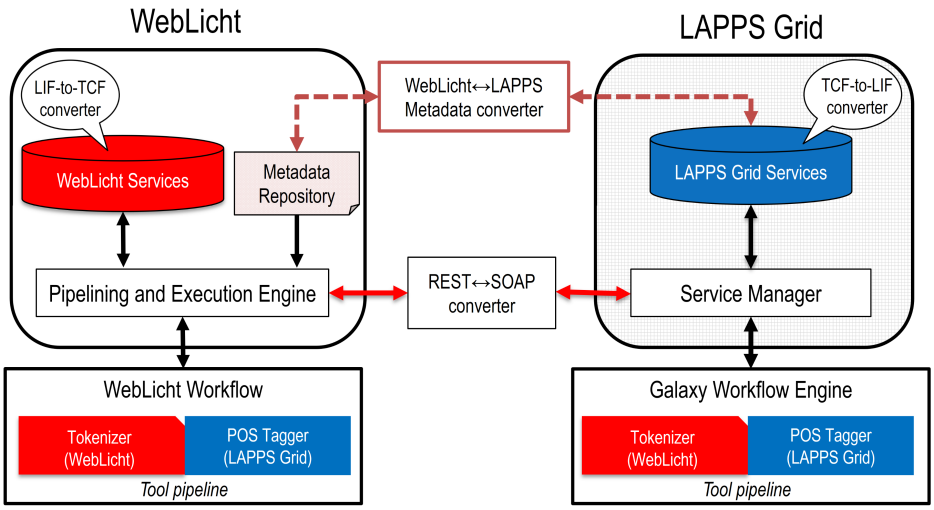


Figure 1: Integration Framework

tion, it is necessary to provide support for authentication and authorization mechanisms that allow users to access resources and services provided by each framework as easily and seamlessly as those within the framework they typically use. Each of these tasks is described in the following sections.

3. Communication Protocols

The main challenge in achieving interoperability with regard to communication protocols lies in the way that the services in each framework are implemented. The LAPPS Grid and CLARIN services use different communication protocols. The LAPPS Grid uses SOAP, whereas the CLARIN tools are implemented as RESTful[6] services. SOAP services send and receive data in SOAP-XML, which is an XML wrapper around a request or response message. RESTful services, on the other hand, send and receive messages directly through HTTP requests or responses. This means that in order to invoke LAPPS services registered in CLARIN, it is necessary to wrap CLARIN's RESTful requests into LAPPS Grid SOAP requests. A SOAP-PROXY service has been implemented to take a REST service request as input, convert it to a SOAP message, invoke the service with the SOAP request, and return the response from the service. The WebLicht services can be invoked directly at their entry points via plain HTTP requests from the LAPPS Grid.

Figure **??** shows the LAPPS Grid-WebLicht integration framework. When one framework calls a service from the other, metadata from the called service is converted and made available to the other, after which it can be processed with the caller's usual handlers. Similarly, data conversion services allow each platform to consume and produce data in its native format. Service calls are tunneled through a proxy, which invokes services using the required protocol.

4. Metadata Conformance

Metadata about the web services available in LAPPS Grid and WebLicht contain information needed to invoke the services from within their respective frameworks. The two frameworks handle web service metadata differently with respect to content, storage location, and fetching.

In the LAPPS Grid, each web service delivers its own metadata on demand, whereas WebLicht web service metadata, which follows the specifications of CLARIN CMDI[7] framework, is retained in CLARIN Center repositories. The WebLicht metadata stored in the repositories includes information corresponding to LAPPS Grid metadata as well as additional details about the format and contents of a service's input and output. In the LAPPS Grid, details about format and contents of a service's input and output are made available as needed by inter-service queries. WebLicht metadata can be converted to LAPPS Grid metadata automatically, but because WebLicht metadata includes additional information beyond that provided by LAPPS Grid services, there is potential information loss; however, because WebLicht metadata is stored in a registry, it can be restored in a WebLicht→LAPPS Grid→WebLicht round trip. LAPPS Grid metadata cannot be automatically converted to WebLicht metadata because of WebLicht's requirement to store the information in the registry; to handle this, it is necessary to store LAPPS Grid metadata in the WebLicht registry manually, and update it when required.

5. Syntactic Interoperability

The problem of differing data exchange formats has been addressed at the syntactic level by implementing converters between LIF and TCF as web services, and registering them in both frameworks. As noted above, syntactic interoperability can be achieved for UDPipe's CoNLL-U by further developing a converter between CoNLL-U and TCF, thus effecting indirect conversion between CoNLL-U and LIF

[6] https://www.w3.org/2001/sw/wiki/REST

[7] http://www.clarin.eu/cmdi



Figure 2: Example LIF (top) and TCF (bottom) formats

with TCF as the liaison. Note that here, we address conversion from CoNLL-U to other formats, but not conversions from other formats to CoNLL-U.

Structural differences and the granularity of annotation data in LIF, TCF, and LINDAT/CLARIN's CoNLL-U format imposed several challenges, outlined below.

5.1. Annotation Layers

Between TCF and LIF, the major differences result from the ways in which annotation *layers* (called *views* in LIF) are defined. TCF has a fixed number of annotation layers (shown in the left column of Table **??**) and allows only one occurrence of a given annotation layer per document. Each layer has a fixed structure, most consisting of flat lists of XML elements, although some layers are slightly more structured.

In contrast, LIF does not place restrictions on the number and content of its views, and each service can add any number of views[8] or add information to an existing view, as long as the metadata in the view provides the relevant information about the view's content. All views in LIF have the same structure, with annotations consisting of a list of elements that follow the same template: an annotation object with a type, an identifier, beginning and end character offsets or a reference to other annotations (in the same view or another) and a dictionary of feature/value pairs.

Figure **??** shows a token layer in LIF and the corresponding `tokens` and `POSTags` layers in TCF. Note that in TCF part-of-speech tags appear in a separate layer referring to token objects in the `tokens` layer, while in LIF, part-of-speech is given as the value of the *pos* feature.[9] Therefore, in this case conversion requires either expanding one LIF view into two TCF layers, or collapsing two TCF layers into one LIF view.

Because TCF allows only one occurrence of an annotation type per document, if a LIF document contains multiple LIF annotations for the same phenomenon only one can be chosen for conversion into TCF. This makes it necessary to identify an optimal alternative; more problematically, it means that there is potential information loss (i.e., loss of additional alternative views) when converting from LIF to TCF. Thus a round trip from LIF to TCF back to LIF may not produce the same result. This remains an open problem at this time; currently, the last view for any given annotation type is included in the TCF representation, and the original LIF document may be stored in its entirety in TCF's `textSource` element, to be restored upon its return.

CoNLL-U, an example of which is shown in Figure **??**, differs substantially on the surface from both TCF and LIF in terms of physical format. We could regard the set of columns as a fixed set of annotation layers, most consisting of single elements while others have internal structure, and allowing only one column per phenomenon. Conceptually, the information in a CoNLL-U representation corresponds to the token, lemma and part-of-speech layers in TCF (which correspond to the Token layer in LIF) and the dependency parse layer in both TCF and LIF.

5.2. Anchoring to Primary Data

LIF is a stand-off annotation format, which requires that all annotations refer to either character offsets in the primary data or to other annotations that are themselves either directly or indirectly (via a chain of annotations) anchored in primary data. TCF annotations are not grounded in the primary data but instead refer to a single base layer consisting of tokens[10]. Primary data is stored in the TCF `text` element, but in most cases no anchors into the text are provided. Therefore, conversion from LIF to TCF requires mapping character offset anchors to each token element, and conversion in the reverse direction demands recomputing offsets from the primary source.

As a column-based format, CoNLL-U does not provide the primary data source, and therefore conversion from CoNLL-U to TCF first requires reconstruction of a "source text" from the list of surface tokens.[11] To convert to LIF, character offsets into the source text must also be computed, as LIF requires anchoring of at least one annotation in primary data.

5.3. Tree and Graph Structures

Phrase structures are represented in LIF with explicitly labeled edges between nodes, by `parent` and `child` fea-

[8] In practice, most tools add a single view.

[9] Lemmas are handled the same way as part-of-speech tags in both schemes.

[10] An exception is the synonomy layer, which refers to lemma identifiers that in turn reference the token layer.

[11] The column labels used in CoNLL-U are given in Table **??**, below.

1	They	they	PRON	PRP	Case=Nom|Number=Plur	2	nsubj	2:nsubj|4:nsubj
2	buy	buy	VERB	VBP	Number=Plur|Person=3|Tense=Pres	0	root	0:root
3	and	and	CONJ	CC	_	4	cc	4:cc
4	sell	sell	VERB	VBP	Number=Plur|Person=3|Tense=Pres	2	conj	0:root|2:conj
5	books	book	NOUN	NNS	Number=Plur	2	obj	2:obj|4:obj
6	.	.	PUNCT	.	_	2	punct	2:punct

Figure 3: CoNLL-U representation of "They buy and sell books."

```
<dependency govIDs="t_1" depIDs="t_0" func="SB"/>
<dependency depIDs="t_1" func="ROOT"/>
<dependency govIDs="t_1" depIDs="t_2" func="MO"/>
<dependency govIDs="t_4" depIDs="t_3" func="PNC"/>
<dependency govIDs="t_2" depIDs="t_4" func="NK"/> . . .
```

```
{ "@type": "Dependency", "label": "ROOT", "id": "dep0",
  "features": {"governor": null, "dependent": "v1:tok1" }},
{ "@type": "Dependency", "label": "nsubj", "id": "dep1",
  "features": {"governor": "v1:tok1", "dependent": "v1:tok0" }},
{ "@type": "Dependency", "label": "nobj", "id": "dep2",
  "features": {"governor": "v1:tok1","dependent": "v1:tok2" }. . .
```

Figure 4: Example TCF and LIF representations of dependency relations

tures of the `Constituent` annotation type. TCF, on the other hand, represents phrase structure implicitly by exploiting the hierarchical nesting of XML tags to represent parent-child relations among constituents using embedding XML elements; only when an element's (node's) children consist entirely of tokens are they listed explicitly within a `tokenIDs` element. Therefore, conversion of phrase structure trees in TCF to LIF requires interpreting and flattening the XML structure. Conversion from LIF to TCF is a relatively straightforward matter of creating embedded XML elements for constituents and their children.

CoNLL-U provides *head-deprel* pairs for each token, which taken together form the graphs representing a dependency parse for each sentence. In TCF and LIF, these relations are represented using references to the IDs of relevant entities as the value of features or attributes such as "governor" and "dependent", as shown in Figure **??**. Conversion from CoNLL-U is a straightforward matter of deconstructing the micro-format in which the head-deprel pairs are rendered, in order to generate the corresponding TCF and LIF representations.

5.4. Multi-word Tokens

CoNLL-U includes means to represent multi-word tokens that correspond a single surface token (e.g., *want* and *to* for surface string *wanna* in English, or *in* and *dem* for surface string *im* in German), which are interspersed among surface tokens in the same column. When multi-word tokens are present, CoNLL-U annotations apply to only the word comprising the multi-word token; thus, the surface form is irrelevant for the purposes of processing annotations in other columns. Since they are not referenced from subsequent annotation layers, the surface form and its parts can be provided in attributes on TCF `token` elements that contain the multi-words it corresponds to.

In LIF, multi-word tokens can appear in an additional token layer, which can in turn reference the corresponding surface token in the surface token layer. The multi-word tokens can then be referenced from other annotations.

6. Semantic Interoperability

The three frameworks in this project reference different sets of linguistic objects (with some overlaps), using differing terminology and expressing relations among these objects in differing configurations. To enable semantic interoperability among the services in the three frameworks, we provide means to specify the linguistic objects that a given service or tool requires as input and produces as output, so that other producers and consumers (i.e., other services and/or tools) can determine if its requirements are satisfied. In a pipeline of tools or web services, this information is provided as metadata that must be checked automatically for compatibility. This in turn demands that identical concepts must be identified as such, either by direct match or by reference to a common web-addressable entity. Internally, a given tool may use different terminology; the only necessity is that the tool is wrapped to map the exchange vocabulary into the internal terminology and vice versa. Thus in principle, a single mapping of a tool's specific terminology into and out of the common exchange vocabulary is sufficient to enable information exchange with all others.

At present, TCF concepts are briefly defined in comments to TCF's XML schema. In LIF, concepts are linked to URI-addressable definitions in the Web Service Exchange Vocabulary (WSEV)[12]. The LINDAT/CLARIN UDPipe tools deal primarily with dependency annotation and produce annotations conformant to the Universal Dependencies specification[13] for treebank annotation. Conversion among TCF, CoNLL-U, and LIF thus indirectly links all three frameworks' vocabularies to the WSEV.

We have developed a mapping and linkage among concepts defined in the the vocabularies of WebLicht, LINDAT/CLARIN, and the WSEV to cover all entities contained in any one of them. All three vocabularies cover similar linguistic phenomena and overlap in many instances, and in fact, semantic mapping has proved to be much more straightforward than originally expected. The most common modification involves extending specifications in the vocabularies to accommodate additional concepts in the others.

TCF has a fixed number of annotation layers, each of which can be construed as representing a concept in the TCF vocabulary. Table **??** shows the mapping from each of

[12] http://vocab.lappsgrid.org [13] http://universaldependencies.org

TCF layer	WSEV equivalent
token	Simple mapping to Token
sentence	Simple mapping to Sentence
lemma	Map to Token#lemma
POStag	Map to Token#pos
parsing	Simple mapping to PhraseStructure and Constituent. Requires conversion of the XML tree structure to an explicit directed graph.
depparsing	DependencyStructure, Dependency
namedEntities	Simple mapping to NamedEntity
references	Map to Coreference. Add features to correspond to TCF's *heads*, *type*, *rel*, etc.
textstructure	Paragraph and Sentence. TCF's page element has no equivalent in WSEV.
synonymy	No equivalent
matches	No equivalent
wordSplittings	No equivalent
geo	No equivalent
phonetics	No equivalent
orthography	No equivalent
wsd	No equivalent

Table 1: TCF layer mappings to WSEV

LIF view	Mapping to TCF
Token	Token information is distributed over the tokens, lemmas and POStags layers
Sentence	Simple mapping to sentences layer
Paragraph	Map to the textstructure layer
NounChunk VerbChunk	No equivalent in TCF.
NamedEntity	Simple mapping to namedEntities layer
Markable	No equivalent in TCF
Coreference	Map to the references layer. Add Markables to TCF. Move Markable and Token annotations to another TCF layer.
PhraseStructure Constituent	Map to the parsing layer. If the view contains Tokens add to the tokens layer.
DependencyStructure Dependency	Map to the depparsing layer. If the view contains Tokens add to the tokens layer.
Relation	No equivalent in TCF
SemanticRole	No equivalent in TCF

Table 2: WSEV to TCF mapping

TCF's layers (listed in the left column) to WSEV vocabulary terms *before* modification for conformity of the two schemes, with an indication of required modifications. Table **??** shows the reverse mapping, from WSEV concepts to TCF layers. As noted, there are a number of cases where there is no equivalent for an item in one scheme or another; however, for most of these the creation of new items for conformance is straightforward.

In general, the XML elements in TCF layers correspond either to vocabulary items in the WSEV or one of its features, as shown in the detailed mapping from elements in the TCF parsing layer in Table **??**. In this table, the elements to the left of the number sign refer to elements defined as tags in the TCF XML schema or to categories in the LAPPS WSEV vocabulary, and the elements on the left are properties. The @ sign indicates that the property is a metadata property in the WSEV object, while the # sign represents a feature of the object.

TCF Element	WSEV equivalent
parsing#tagset	PhraseStructure@categorySet
parse	PhraseStructure
parse#ID	PhraseStructure#id
constituent	Constituent
constituent#ID	Constituent#id
constituent#cat	Constituent#category
constituent#edge	No mapping
constituent#tokenIDs	Constituent#children
cref	No mapping
cref#constID	No mapping
cref#edge	No mapping

Table 3: TCF to WSEV mapping for elements of the parsing layer

It is important to note that semantic interoperability involves only the concepts themselves, and not the way they may be structured in a given scheme. So, for example, a TCF concept may be represented as the name of an XML element, whereas it could appear as a feature associated with a primary WSEV concept type. This is the case, for example, for part of speech in the renderings that were shown in Figure **??**, where part of speech ("tag") in TCF is not only an element name, but also appears in a different TCF layer, whereas in LIF *pos* is a feature associated with the Token vocabulary item and is included in the token view.

Mapping concepts in CoNLL-U to TCF and the WSEV is relatively straightforward but requires adding several concepts to the target vocabularies. Table **??** shows the CoNLL-U row labels, which correspond to a higher-level set of concepts in the CoNLL-U vocabulary, and most of which exist in or have been added to TCF and the WSEV as concepts or features. Figure **??** shows that a representation in the CoNLL-U format includes additional concepts (e.g., "Number", "Case", "Person") in the FEATS column, which correspond to features on Tokens (and other WSEV vocabulary items) and to items in TCF's morphology layer.

6.1. Discussion

The exercise of making LIF, TCF, and CoNLL-U interoperable showed us that for the schemes we dealt with, the greatest obstacles to interoperability were due to variations in representation formats–i.e., at the level of syntax–rather than to variations in semantic categories. Without exception, the semantic categories used in all three schemes were either identical to concepts defined in the other vocabularies or were easily added where missing. This common-

ID	Word index (range for multiword tokens, decimal number for empty nodes)
FORM	Word form or punctuation symbol
LEMMA	Lemma or stem of word form
UPOSTAG	Universal part-of-speech tag
XPOSTAG	Language-specific part-of-speech tag
FEATS	List of morphological features
HEAD	Head of the word
DEPREL	UD relation to the HEAD
DEPS	List of head-deprel pairs
MISC	Anything else

Table 4: CoNLL-U concepts

ality among the three schemes can be largely attributed to the fact that all three schemes deal broadly with basic concepts that have been widely used in the field, such as *token, part-of-speech*, elements of consituency analysis and dependency structure, etc. Both LIF and TCF are intended to be general-purpose schemes; only CoNLL-U deals with a specific phenomenon in depth, and even in this case the concepts used are relatively well-established for dependency analysis. Even where *category labels* may differ, the concepts they represent are a part of the common set of objects used in annotation tools.

Another reason why semantic interoperability posed fewer problems in our exercise is that we do not attempt to harmonize what are commonly referred to as *tagsets*, but rather require clear identification of the tagset used in the annotation (e.g., part-of-speech, dependency relation, constituent names, etc.) in metadata[14]. There have been attempts to map and/or harmonize such values (e.g., OLiA (**?**)), which have amply demonstrated the difficulties of this kind of mapping.

Although we have skirted the issue of tagset interoperability, we argue that any attempt to achieve interoperability at this level would impede our ability to move forward. We focus instead on *tool* interoperability, by requiring metadata identifying the tagset used in a given annotation, and designing our services to check that the tagsets required as input for one tool are satisfied by the tagsets appearing in the output of another. This means that a tool requiring, say, the Penn part-of-speech tags, will effectively "refuse" input from a tool whose output uses another tagset. This obviously places limits on full semantic interoperability; however, in our experience, it is necessary at this time to recognize the distinction between object/feature *names* and their values in order to make progress toward full interoperability.

We do not argue that semantic interoperability for linguistically annotated resources is "solved" or even close to being solved; but our experience suggests that there is evidently a fair degree of commonality among linguistic annotation schemes consumed by NLP tools, at least in terms of the concepts included. Our experience suggests that to move forward, the quest for semantic interoperability should be further sub-divided into identification of *categories* (or objects) and *values*, and both should be addressed separately. More crucially, it suggests that format differences may pose a far greater obstruction to overall interoperability than assumed. This in turn suggests that in designing annotation schemes and formats as well as attempting mappings among different schemes, it is critical to clearly separate issues of format (syntax) from annotation scheme semantics.

[14] This is true for TCF and LIF; CoNLL-U explicitly specifies values for some categories (UPOSTag and DEPREL, for example) that are to be used.

7. Example

Figure **??** presents an example use of tools from both the LAPPS Grid and WebLicht frameworks, accessed via the WebLicht user interface. An input text corpus is converted to LIF format, tokenized and sentence-split by LAPPS services, followed by a LIF-to-TCF format conversion to allow processing to continue using CLARIN services. The lower window in the figure *Input and Chain Selection* shows the tool chain that was selected for execution. After the LAPPS Grid services (Stanford Tokenizer and Stanford Splitter) are executed, the LIF-to-TCF converter is used to return to the WebLicht plaform; the upper window *Next Choices* shows the available WebLicht services. To revert to using LAPPS Grid services again, the user chooses TCF-to-LIF; in this way, a user can alternate between LAPPS Grid services and WebLicht services and vice versa, without ever leaving the WebLicht interface.

Figure **??** shows a portion of the LAPPS Grid Galaxy interface and a workflow in which a tokenizer and part-of-speech tagger from WebLicht are invoked, followed by a named entity recognizer from the LAPPS Grid. The LAPPS Grid Galaxy interface automatically detects and converts formats as needed, without intervention from the user; at the time of this writing detection and conversion for TCF has not been implemented, and therefore the TCF-to-LIF converter is explicitly inserted into the pipeline in order to feed into the LAPPS Grid entity recognize.

8. User Authentication and Identification

Using the LAPPS Grid through the Galaxy interface requires a simple registration in order to provide a uniquely-named workspace for each user, but there are no license requirements or usage restrictions depending on the user type or affiliation. The LAPPS Grid can be used directly via its API[15] without registration or any other restriction. When it is necessary to provide secure access to licensed data and software, the LAPPS Grid employs "click through" licenses that can be accepted in real time as well as verification via timed tokens (**?**).

In the CLARIN infrastructure, users must be authenticated via identity providers (IdPs) belonging to EU national academic identity federations. Identity providers are typically universities or other academic institutions that have information about users and provide secure authentication (login) services. Service providers (SPs, such as WebLicht) can decide to trust users authenticated via an IdP. This secure system of identifiable user login via IdPs provides reasonable assurance that WebLicht services are being used for academic purposes.

[15]

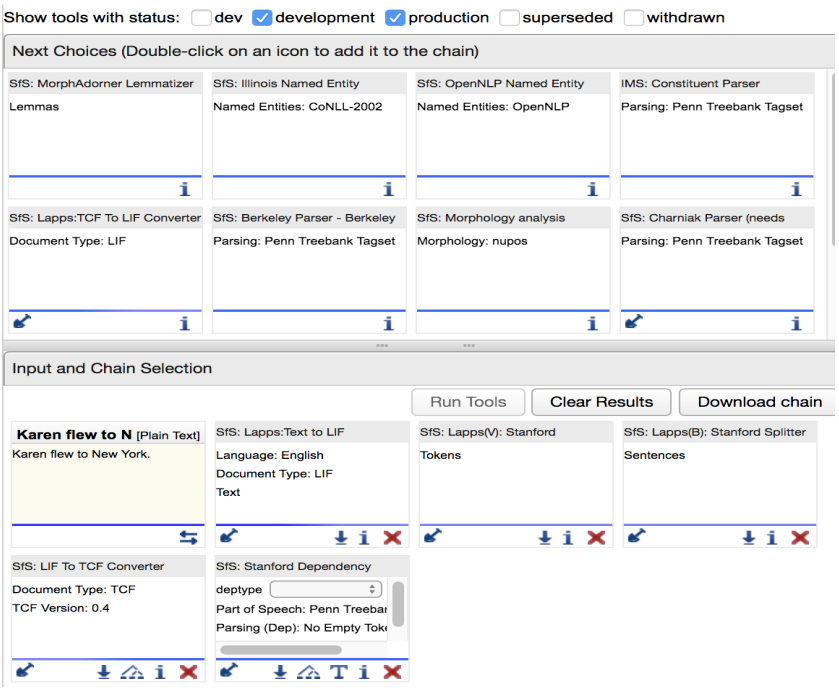


Figure 5: Invoking LAPPS Grid services from WebLicht.

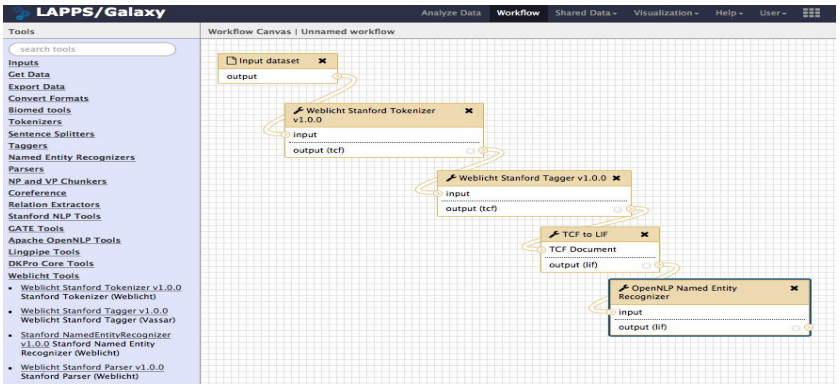


Figure 6: Invoking WebLicht services in a LAPPS Grid/Galaxy pipeline

Because of the need for authentication, prior to this project LAPPS Grid users were in general not able to access login-protected CLARIN services. To provide this access, we have devised means for LAPPS Grid users with appropriate credentials to access these services, by registering the LAPPS Grid as both a service and an identity provider with the CLARIN Service Provider Federation (SPF)[16]. The CLARIN SPF includes more than 1700 European institutions, whose members have access to all CLARIN services; academics whose institutions are not part of the inter-federation can be approved for use of CLARIN services through a process of in-person identity verification at their home institutions. Without further authentication, users registered in the LAPPS Grid may access only the publicly available services from WebLicht and other CLARIN centers. However, a LAPPS Grid user can be authenticated for full access to CLARIN services if he or she is a member of an academic institution in the InCommon identity management federation[17], a secure and privacy-preserving trust fabric for research and higher education in the United States that performs a similar function as the national federations in Europe that form the CLARIN SPF federation.

Specific software (e.g., Shibboleth software[18]) has to be installed on the LAPPS Grid and WebLicht servers in order for them to act as identity/service providers. "Trusted" users are then identified either individually or using aggregated feeds of entities such as InCommon CLARIN SPF. The service provider must in turn join these (inter-) federations, to ensure the trust is mutual; otherwise, the service provider would have to negotiate the trust with each and every identity provider. Once the trust has been secured, users from a trusted identity provider can login to a trusted service provider by authenticating via their institution's login page, after which the identity provider sends a confirmation with additional attributes (e.g., id, email, name, affiliation, entitlement) to the service provider via a secure channel.

[16] https://www.clarin.eu/content/service-provider-federation

[17] https://www.incommon.org/

[18] https://www.shibboleth.net/

To summarize, users who can verify their academic affiliation through InCommon, an academic identity provider at their own institution, or the CLARIN SPF identity provider may use all services from either the LAPPS Grid or WebLicht. Others may use only services that are openly available (this includes the majority of services in the LAPPS Grid). In addition, our access control solution supports *single sign-on* in order to minimize the burden of requiring multiple credentials and/or re-entering credentials repeatedly.

9. Conclusion

The meta-framework providing for mutual access between the LAPPS Grid and the two CLARIN frameworks has the potential to transform scholarship and development across multiple disciplines in the sciences, language and social sciences, and digital humanities by providing a transparent interface to a massive range of tools and resources at an unprecedented level.

Bridging the LAPPS Grid and WebLicht frameworks significantly extends the capabilities of each by providing seamless access to services that are currently unavailable in each. For example, the LAPPS Grid will benefit from availability of a more extensive suite of tools for output visualization than currently exists in the LAPPS Grid, and WebLicht will gain access to the sophisticated evaluation services the LAPPS Grid provides.

The potential impact extends even farther than the two frameworks involved, as both the LAPPS Grid and WebLicht are federated with other frameworks to which they provide a gateway. WebLicht is a member of the EU CLARIN network and therefore provides access to multilingual tools and resources from CLARIN Centers hosted throughout Europe. The harmonization will also extend to Asia because the LAPPS Grid is federated with seven other grids[19], including the Language Grid housed at Kyoto University[20]. Like the LAPPS Grid-CLARIN bridge, this federation provides interoperability and seamless access among atomic and composite web services available from any of the grids involved.

A more wide-ranging impact of this project may result from its success in providing interoperable access to services in three major frameworks that were developed entirely independently. Although we acknowledge that universal interoperability for NLP tools is far from a solved problem, we believe this project takes an important step towards its achievement. In particular, the exercise of pursuing semantic interoperability among the three frameworks has yielded new insights into the nature and source of obstacles to interoperability that could significantly impact future progress towards this seemingly elusive goal.

Our solutions to the problems of authentication, authorization, and access to licensed data and tools can serve as a model for other project facing the same issues. Finally, the work performed takes a major step toward the harmonization of software and data developed across the globe that can vastly ameliorate and eventually eliminate the current lack of reusability of resources and tools that thwarts research and development in the field and hampers collaboration. Ultimately, the LAPPS Grid-CLARIN metanetwork may lay the groundwork for the eventual creation of a global network of grids and frameworks to serve researchers, developers, and users of NLP technologies.

[19] Federated Grid of Language Services (FGLS) (**?**).

[20] http://langrid.org

Acknowledgments

The work described here is supported by the Mellon Foundation Grant "Transatlantic Collaboration between LAPPS and CLARIN: Semantic, Technical and Infrastructural Interoperability of Services". The LINDAT/CLARIN Research Infrastructure, projects No. LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781 are supported by the Ministry of Education, Youth and Sports of the Czech Republic. WebLicht is developed as part of the CLARIN-D research infrastructure, with the development work coordinated by the CLARIN-D Center in Tübingen, and is funded by the German Federal Ministry of Education and Research (BMBF), project number 01UG1620A. The LAPPS Grid was developed under funding from the U.S. National Science Foundation SI^2 grants ACI 1147944 and ACI 1147912.

10. Bibliographical References

Buchholz, S. and Marsi, E. (2006). Conll-x shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, pages 149–164, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chiarcos, C. (2008). An ontology of linguistic annotations. *LDV Forum*, 23(1):1–16.

Cieri, C. and DiPersio, D. (2014). Intellectual property rights management with web service grids. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 93–100, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Dima, E., Hinrichs, E., Hinrichs, M., Kislev, A., Trippel, T., and Zastrow, T. (2012). Integration of weblicht into the clarin infrastructure. In *Proceedings of the Joint CLARIN-D/DARIAH Workshop at Digital Humanities Conference 2012: Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts*, pages 17–23, Hamburg, Germany.

Giardine, B., Riemer, C., Hardison, R. C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., Miller, W., Kent, W. J., and Nekrutenko, A. (2005). Galaxy: a platform for interactive large-scale genome analysis. *Genome Research*, 15(10):1451–55.

Hinrichs, E. and Krauwer, S. (2014). The clarin research infrastructure: Resources and tools for ehumanities scholars. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., Wang, D., Suderman, K., Verhagen, M., and Wright, J. (2014). The language applications grid. In *Proceedings of the Ninth International Conference on Language Resources*

and Evaluation (LREC 2014), Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Ide, N., Suderman, K., Verhagen, M., and Pustejovsky, J. (2016). The language applications grid web service exchange vocabulary. In *Revised Selected Papers of the Second International Workshop on Worldwide Language Service Infrastructure WLSI (2015)*, pages 18–32, Kyoto, Japan. Springer-Verlag New York, Inc.

Ishida, T., Murakami, Y., Lin, D., Nakaguchi, T., and Otani, M. (2014). Open Language Grid–Towards a Global Language Service Infrastructure. In *The Third ASE International Conference on Social Informatics (SocialInformatics 2014)*, Cambridge, Massachusetts, USA.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666, Paris, France. European Language Resources Association.

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: Trainable pipeline for processing coNLL-u files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4290–4297, Paris, France. European Language Resources Association.

Verhagen, M., Suderman, K., Wang, D., Ide, N., Shi, C., Wright, J., and Pustejovsky, J. (2015). The lapps interchange format. In *Revised Selected Papers of the Second International Workshop on Worldwide Language Service Infrastructure WLSI (2015)*, pages 33–47, Kyoto, Japan. Springer International Publishing.

Fluid Annotation: A Granularity-aware Annotation Tool for Chinese Word Fluidity

Shu-Kai Hsieh[1], Yu-Hsiang Tseng[2], Chih-Yao Lee[3], Chiung-Yu Chiang[4]
[1,3,4]Graduate Institute of Linguistics, [2]Department of Psychology
[1,2,3,4]Knowledge-Yielding Language Engineering (KYLE)
National Taiwan University, Taipei, Taiwan
{shukai, seantyh, chiyaolee, sonnejoy}@gmail.com

Abstract

This paper presents a novel word granularity-aware annotation framework for Chinese. Anchored in current functionalist linguistics, this model rearranges the boundary of word segmentation and linguistic annotation, and gears toward a deeper understanding of lexical units and their behavior. The web-based annotation UI also supports flexible annotation tasks for various linguistic and affective phenomena.
Keywords: Wordhood, annotation, Chinese Word Segmentation

1. Introduction

Word segmentation has been one of the most important preprocessing NLP tasks in the pipeline-alike architecture for languages without explicit word delimiter in their written forms. The engineering treatment of word segmentation naturally leads to the requirement of the existing gold standard, including a presumably agreeable standard and a word-segmented corpus based on the standard. Unfortunately, this long-standing rationale does not provide a convincing argument that concurs with current findings of cognitive science.

Words as conventionalized symbols which presents the function by which meaning is attached to form. However, the basic units of cognition are clearly not words. Theoretical and empirical advances in the past decade have revealed that word meanings are only pointers to coherent chunks of encyclopedic knowledge (Malt and Wolff, 2010). In the light of reading task,(Liu et al., 2013) show that Chinese reader did not follow the segmentation rules, and tended to chunk single words into large information units, implying that word meanings sometimes work against the way knowledge is organized in memory.

In this paper, we argue that *word-meaning pair* is fluid in nature, whose granularity (in terms of the length of the word) is influenced by its underlying ontology (paradigmatic dimension), surrounding context (syntagmatic dimension) and real-world application (pragmatic force). Under this view, word segmentation can be considered as *wordhood annotation*, disentangling itself from the *error*-prone role in the NLP pipeline architecture.

2. Review

Word segmentation has been a thorny issue in NLP for many decades. In addition to structural ambiguity resolution and unknown word detection, the current focus is concerned with *propagation error* and *domain adaption*. As the pre-processing task in the pipeline architecture, word segmentation errors can propagate to later processing stages. To handle with this, joint approaches exploiting various machine learning models including the latest neural network have been proposed (Lyu et al., 2016; Shao et al., 2017). Second, it has been recognized that different applications and domains have different calls for different granularities of word segmentation. Recent neural domain adaptation approaches also work through cross-domain embeddings to improve the cross-domain performance (Cai and Zhao, 2016; Zhang et al., 2014). However, a critical examination of the underlying assumption, and their assessment in the light of naturally occurring linguistic data, reveal its inherent contradictions (Taylor, 2012). In the following sections, we introduce the proposed Fluid Annotation model in more details.

3. Fluid Annotation

The scheme of Fluid Annotation comprised of three main components: DeepLexicon, Fluid Segmentation & Tagger, and Annotation UI (Figure 1). Six crucial steps were identified in the scheme: (1) unprocessed text was fed into fluid segmentation and tagging preprocessor, where (2) text was segmented with different granularities, and automatically labeled with possible tags; (3) Annotation UI was provided with these segments and tags, in which (4) annotators could furthermore refine (by regrouping, or dividing) the segmentation with fluid segment tool, or annotate the segmentation (with annotation "brush"), and view the annotation in a natural text context. (5) The annotations created by users were again feed backed to deep lexicon, in which granularities parameters of lexical bundles and update the lexicon tag set table were updated. (6) The updated lexicon would again provided latest information to fluid segmentation & tagging in next session. As a result, a cycle was established where not only the flexibility of linguistic pattern is assured, but annotators' effort cumulated in the process.

3.1. DeepLexicon

DeepLexicon provides all candidate words and tag data associated with the given words used in segmentation and tagging. Distinctively, DeepLexicon featured lemma of different granularity that facilitate fluid segmentation in following steps.

Chinese words has strong tendency to be monosyllabic and disyllabic. However, in segmentation or other practical

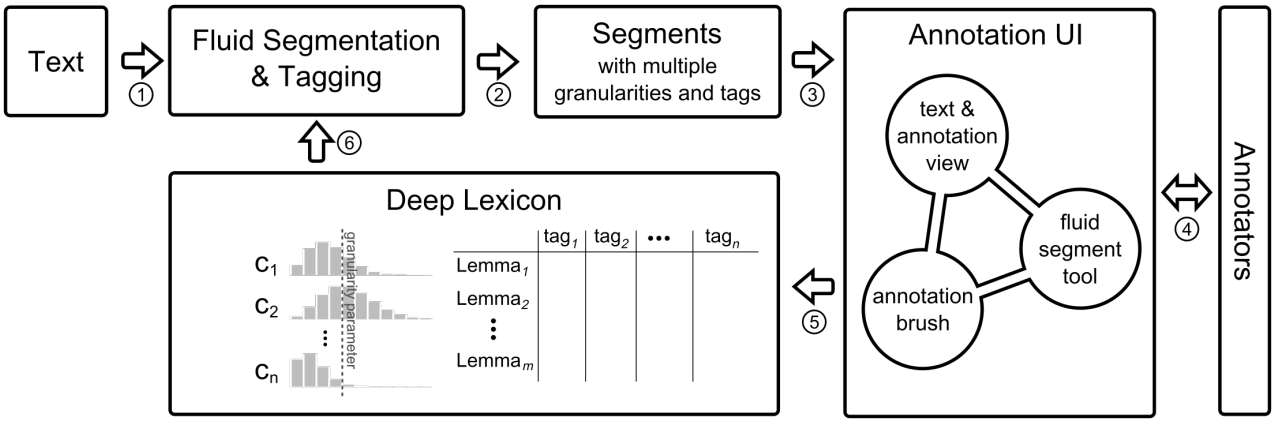


Figure 1: Scheme of Fluid Annotation Overview. Six critical steps were identified in the annotation scheme: (1) input text, (2) preprocess text, (3) prepare segments with multiple granularities and automatic taggings, (4) revise segmentations and tags, (5) annotation feedback, (6) improve segmentation and tagging with new lexical information. Detail descriptions were provided in respective section.

annotation scenarios, word is just one level of information among other linguistic components: multiword expressions, compounds, idiom, or lexical bundles. Previous segmenter relies on a "gold standard" to achieve a high performance in a word segmentation task, virtually eliminate other possibilities to look into groups larger than words.

To alleviate the "hard-cut" issue brought by standard segmentation, DeepLexicon, along with Fulid Segmentation, used in this scheme features "words" of different granularity. "Word granularity" refers to a sequence of lexical patterns of different length. These patterns occurs regularly in different context and carry out a relatively stable communication function. In this sense, "word with different granularities" encompass other linguistic constructs, such as multi-word expression, compounds, idiom, or lexical bundles. For ease of interpretation, granularity is defined as a number ranged from 0 to 1, where we assign granularities of 0 as more fine-grained (shorter patterns, in unit of character count), and 1 as a pattern more coarse-grained (more characters).

In order to operate granularity formally, we further define the granularity of any given word by first calculating the word-length distribution of all the words starting with the same leading character. Secondly, the value of granularity is the cumulative probabilities of the word-length distribution:

$$\text{Granularity}(w) = \sum_{l=1}^{L(w)} p(l; \text{leadChar}(w))$$

where w is the word of interest, leadChar(w) is w's leading character, $L(w)$ denotes the word length of word w and $p(l$;leadChar(w)) is the probability density function of word length l, given the word's leading character.

The current lexicon included 13,5424 lemma which collected from various source. Besides from conventional texts, the lexicon also contained neologism extracted from Taiwan largest Internet forum, emotion expressions and academic lexical bundles commonly found in Chinese academic writings. The resulting word list contained considerably long bundles as revealed in Figure 2. The distribution distinctively called for a novel segmentation procedure which could accommodate the dynamic patterns frequently observed in Chinese discourse.

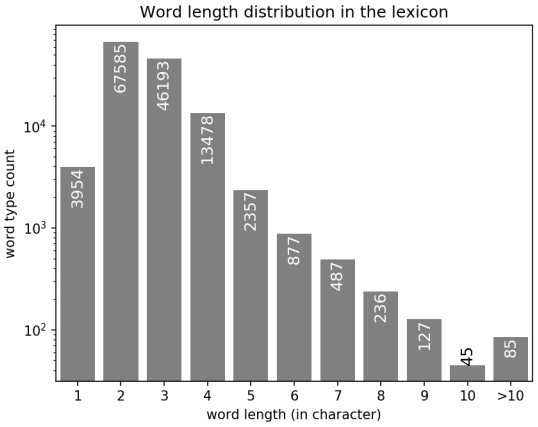


Figure 2: Word length distribution in the lexicon

It is noteworthy that, although the base lexicon already had abundant lexical entries, the lexicon here is designed to be incremental with annotators' collaboration. When annotators group/divide sequence of words in Annotator UI (see below), granularity of the corresponding lemma will automatically adjust accordingly, and segmentation results also reflect the change. Furthermore, we posed few limitations of what can be considered as a "word" in the lexicon. Annotators add their new lemma appropriate in their studies, as long as the pattern is a valid character sequence representable with Unicode. The flexibility is particularly vital when dealing with unconventional and unstructured text, which is dominant in social media, micro-blogging or forums.

Besides the "word" information itself, DeepLexicon also

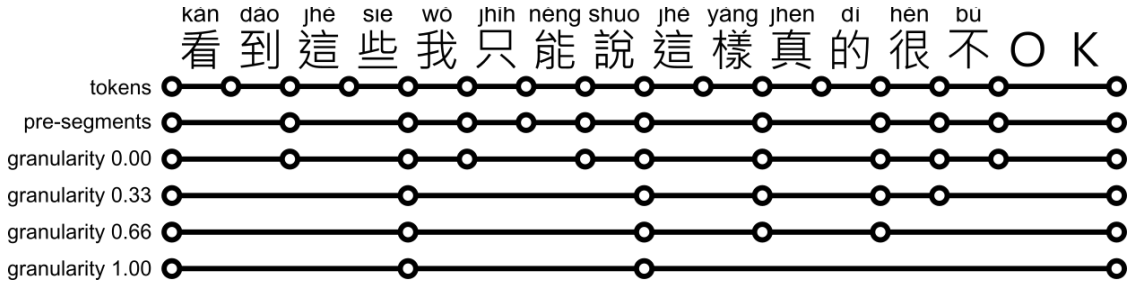


Figure 3: Word segmentations under different granularities.

stored linguistic information from other linguistic resource and user feedback from Annotation UI. For instance, sentiment polarity, mood, and frequency are predefined in DeepLexicon. As annotators created new annotations, these information fed back to DeepLexicon and new tag sets were created. These new tag information along with the predefined tag data, in turn provided a more probable tag by Fluid Tagger. That is, DeepLexicon expanded new lemma and tag information as annotation process progressed. Different annotators could work on the text and share their annotations with others, so the annotations effort could be cumulated in a systematic fashion.

3.2. Segmentation & Tagging

Segmentation is the utmost important step of processing Chinese text. Once the character string of Chinese text has been segmented to multiple words, these words became the only relevant units in subsequent processing steps. The fact that most preprocessing steps only produced single version of word is not without challenged in Chinese, and it profoundly constrained how Chinese text can be annotated and interpreted in later processing steps and analysis.

Fluid Segmenter, instead of pursuing the unique "golden answer", aimed to present the whole spectrum of possibilities on how multiple syllables in Chinese, which represented by multiple individual characters conglomerate into a larger linguistic pattern. Most of Chinese word segmenters based on algorithms which can identify words in a pre-defined segmented corpus. Different segmenters differs on the particular algorithms they implemented. For instance, segmenter in Stanford CoreNLP (Manning et al., 2014) implemented a sophisticated conditional random field model that performed well on segmentation task. However, since the segmenter solely focused on aligning themselves with a predefined word segmentation, different possibilities of segmentation became difficult, if not impossible, to shown themselves in the model outputs.

The segmenter provides words with different granularity by multiple passes of maximal matching and segmentation alignments. To start segmentation, segmenter firstly tries to start with a coarse-grained level (e.g. granularity parameter = 0). Lexicon are queried with the character segmenter encounters, with the granularity parameter in question. The lexicon then offered a full list of possible words starting with the character, whose word granularities are among the designated parameter and 1. The words lexicon provided are then matched against the text from coarse- to fine-grained. If segmenter found a matched, further candidates in the word list are skipped. The segmenter stored the matched sequence, and move the position to the character after the matched sequence. When all characters in the text are attempted, segmenter moved to a higher granularity and repeat the procedures above (Figure 3).

Different granularities of words are identified after the segmenter finish procedures above. These words may contains conflicting word boundaries and isolated single characters which are either one-character words or out-of-vocabularies in lexicon. Although conflicting word boundaries and novel words may itself be an interesting issue in certain research, some studies need an acceptable segmentation so researchers can focus on the patterns of interest.

A pre-segmenter can optionally be incorporated to provide a quick and conventional way of segmentation. The benefits of a pre-segmenter is to solve word ambiguities frequently observed in Chinese text, and alleviate the problem of OOV issues faced in a lexicon-based segmentation. The results from pre-segmenter are the segmentation to which other word granularities align themselves. Specifically, the patterns from different granularities can merge word sequences produced in pre-segment, but dismissed if the word conflict with the pre-segment results. Results from pre-segmenter can be safely ignored, and the final segmentation would only aligned with character-based tokenization.

3.3. Annotation UI

Annotation is a paramount step to develop linguistic theory. Despite the significance in linguistic researches, problems as basic as tokenization still profoundly affect the annotation practices (Ide, 2017). Annotation UI, a browser-based annotation user interface, was aimed to create an environment where researchers could smoothly annotate the focused linguistic phenomena based on the automatic outputs from Fluid Segmentation and Tagger.

Given the fluidity of Chinese expressions, it's unlikely any finite collection of lemma, such as DeepLexicon, could exhaustively satisfied every need of linguistic investigations. Although Fluid Segmentation allow considerably more flexibility to researchers as they can freely decide the level of interest in granularities, there were still some circumstances that annotators or researchers wish finer- or coarser- grained segmentation results, and some of these results are stable across context. These instances are candidates for addition into Lexicon.

Segmentation is one of the most crucial form of anno-

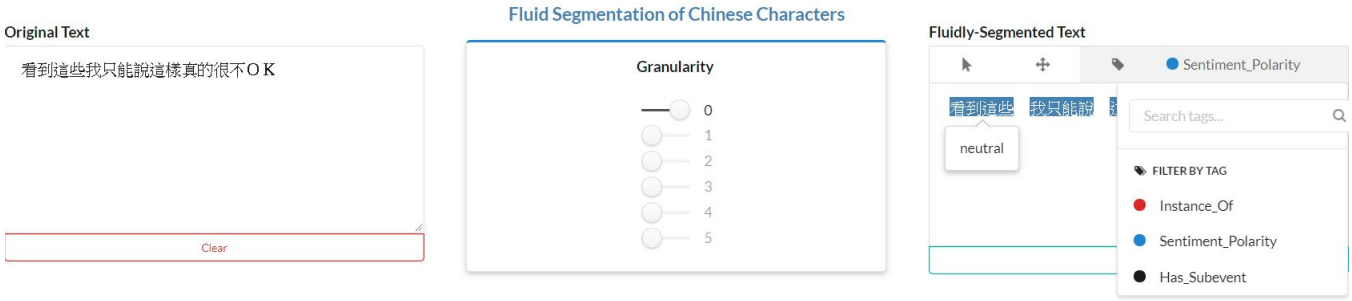


Figure 4: Screenshot of Annotation UI. Left panel was text input window, central panel was for granularity settings, and right panel was annotation window. Three tool buttons were available on the top of the right panel: normal selection tool, fluid segment tool, and annotation brush.

tations. Since most of the subsequent annotations depends on the segmentation in Chinese, the importance of the flexibility supported by segmentation cannot be overstated. Annotators contribute new segments to DeepLexicon through Annotation UI. Annotators first chose an appropriate granularity level. which includes 6 settings, from more coarse-grained (contains longer sequences) to finer-grained (contains shorter sequences), the pre-segment level and the token level. Upon granularity selection, annotators then freely regroup character sequence in text. The annotation process completed when annotators submit the final regrouped text, where lexicon scanned through the segmentation in the text for new patterns. New patterns are added to lexicon and automatically update granularity calculation. These new patterns would be utilized in following text segmentation.

In addition, segments in Annotator UI came with tag suggestion predicted by Fluid Taggers. These suggestions were currently produced by maximum-likelihood estimates based on tag statistics recorded in lexicon, new prediction algorithm can incorporated if more sophisticated suggestion scenario is required. Annotators could either accept the suggested tags, revise the tag, or devised a new tag set entirely. Annotation brush was designed to help annotators intuitively ”paint” the tag on the segments, through which annotators can select categories and tag values they wish to annotate, and click the segments to annotate. Regardless of annotator's decision to add, modify, or delete the tags, annotations would be processed by Annotation UI and feed backed to DeepLexicon, where the tag data would further processed to update future predictions. The procedure ensures the linguistic insights imparted by annotators accumulated in the process.

4. Conclusion

Word segmentation with its underlying generative model of linguistics, and its preprocessing role have set the research agenda for at least half a century in Chinese NLP. Even for languages with space as the word boundary delimiter in their writing system, though useful as a rule of thumb, still begs the question of how words might be defined (Taylor, 2012). Mounting evidence in recent studies have offered alternatively how knowledge comes packaged into coherent chunks in mind, and *word* meaning are closely aligned with these chunks. In the same vein, what we propose in this paper, is a novel fluid annotation model that allows the word granularity to be presented from holistic (un-wordlike) to discrete elements (word-like) via the collaborative annotation. We believe the proposed model could liberate and expand our research imagination, and provides a pathway to connect NLP/NLU with cognitive computing.

5. Bibliographical References

Cai, D. and Zhao, H. (2016). Neural word segmentation learning for chinese. *arXiv preprint arXiv:1606.04300*.

Ide, N., (2017). *Introduction: The Handbook of Linguistic Annotation*, pages 1–18.

Liu, P.-P., Li, W.-J., Lin, N., and Li, X.-S. (2013). Do chinese readers follow the national standard rules for word segmentation during reading? *PloS one*, 8(2):e55440.

Lyu, C., Zhang, Y., and Ji, D. (2016). Joint word segmentation, pos-tagging and syntactic chunking. In *AAAI*, pages 3007–3014.

Malt, B. and Wolff, P. M. (2010). *Words and the mind: How words capture human experience*. Oxford University Press.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Shao, Y., Hardmeier, C., Tiedemann, J., and Nivre, J. (2017). Character-based joint segmentation and pos tagging for chinese using bidirectional rnn-crf. *arXiv preprint arXiv:1704.01314*.

Taylor, J. R. (2012). *The mental corpus: How language is represented in the mind*. Oxford University Press.

Zhang, M., Zhang, Y., Che, W., and Liu, T. (2014). Type-supervised domain adaptation for joint segmentation and pos-tagging. In *EACL*, pages 588–597.

e-magyar – A Digital Language Processing System

Tamás Váradi[1], Eszter Simon[1], Bálint Sass[1], Iván Mittelholcz[1], Attila Novák[2,3], Balázs Indig[2,3], Richárd Farkas[4], Veronika Vincze[4,5]

[1]Research Institute for Linguistics, Hungarian Academy of Sciences
H-1068 Budapest, Benczúr u. 33.
[2]MTA-PPKE Hungarian Language Technology Research Group
[3]Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
H-1083 Budapest, Práter u. 50/a
[4]University of Szeged, Institute of Informatics
H-6720 Szeged, Árpád tér 2.
[5]MTA-SZTE Research Group on Artificial Intelligence
H-6720 Szeged, Tisza Lajos krt. 103.
{varadi.tamas,simon.eszter,sass.balint,mittelholcz.ivan}@nytud.mta.hu,
{novak.attila,indig.balazs}@itk.ppke.hu, {rfarkas,vinczev}@inf.u-szeged.hu

Abstract

e-magyar is a new toolset for the analysis of Hungarian texts. It was produced as a collaborative effort of the Hungarian language technology community integrating the best state-of-the-art tools, enhancing them where necessary, making them interoperable and releasing them with a clear license. It is a free, open, modular text processing pipeline which is integrated in the GATE system offering further prospects of interoperability. From tokenizing to parsing and named entity recognition, existing tools were examined and those selected for integration underwent various amount of overhaul in order to operate in the pipeline with a uniform encoding, and run in the same Java platform. The tokenizer was re-built from ground up and the flagship module, the morphological analyzer, based on the Humor system (Prószéky and Kis, 1999), was given a new annotation system and was implemented in the HFST framework (Lindén et al., 2009). The system is aimed for a broad range of users, from language technology application developers to digital humanities researchers alike. It comes with a drag-and-drop demo on its website: http://e-magyar.hu/en/.

Keywords: text analysis, Hungarian pipeline, integrated toolset

1. Introduction

The paper describes e-magyar, a new integrated text processing pipeline for Hungarian. While Hungarian can be considered an under-resourced language it does have an active and cooperating language technology community which has been developing various tools to cover the basic text processing steps. However, earlier fragmented efforts suffered from a number of factors such as the lack of interoperability, openness, clearly defined license conditions and/or have become limited in some technological aspects such as encoding and annotation systems used as well as efficiency and implementation platform. All these reasons served as the motivation for a collaborative effort by key Hungarian language technology partners to overhaul (sometimes quite radically) the existing tools and, more importantly, to make them interoperable so that they can be integrated into a single coherent technological pipeline.

The objectives of the e-magyar system were to serve as an *open, free and modular* text processing toolset that serves the needs of commercial developers and individual researchers in language technology as well as (digital) humanities and social sciences. It is open and free in that the system as a whole and its individual components are typically available for download through Github repositories, freely available often not just for research and development but for commercial use and they certainly come with clear license terms.

Technologically, the aim was to take the state-of-the-art tools available, eliminate their shortcomings either in specifications, functionality or efficiency and integrate them in a single system so that the performance of the individual module in the e-magyar system should be at least equal that of the original tool before its overhaul.

Specific attention was paid to ensure that the toolset was accessible and useful not just for developers but researchers in the social sciences and humanities (SSH). This reflects an increasing awareness within user involvement efforts in the CLARIN community[1] that SSH researchers are less interested in pre-annotated datasets than in toolsets that are capable to process their own data. To serve the needs of non-language technology specialists, a web service was set up for them to process their data in a drag-and-drop fashion and integration in the GATE system, which has a user-friendly graphical interface, which also facilitates the use of the e-magyar toolset.

2. Text Modules

The e-magyar digital text processing system is assembled as follows. Starting from raw text, the first module (called emToken) performs word and sentence segmentation (see Section 2.1.). Then full-fledged morphological analyses and the lemma of the tokens are identified (emMorph and emLem modules, Section 2.2.). Morphological disambiguation is performed by the emTag part of speech (POS) tagger module (Section 2.3.). Syntactic analysis is accomplished in two different ways: con-

[1]https://www.clarin.eu/

stituency analysis (emCons module) and dependency analysis (emDep module) are also assigned to sentences (Section 2.4.). Finally, maximal noun phrases (emChunk module, Section 2.5.) and named entities (emNer module, Section 2.6.) are identified as well. The modules are built on one another as Figure 1 shows.

2.1. Tokenizer and Sentence Splitter

A brand new Hungarian tokenizer and sentence splitter has been developed, called quntoken[2], which is based on the Quex[3] lexical analyzer generator and was implemented in C++. This tool was integrated into the e-magyar language processing system under the name emToken.

In its features, it mainly follows HunToken[4], which is a rule-based tokenizer and sentence boundary detector for Hungarian (and English) texts. However, emToken differs in several properties, e.g. its input is plain text in UTF-8 encoding, and its output is the text segmented into sentences and words in XML or JSON format. The output can be detokenized, i.e. the input can be reproduced from the output. It can be run as a standalone program or via an API. It comes with test cases covering all the built-in tokenizing rules.

The evaluation of the performance of emToken was made on the Szeged Treebank (Csendes et al., 2005). As for the sentence segmentation, the emToken failed in 2,131 cases of 81,648, which means a 97.39% accuracy. As for the tokenization, we counted word accuracy, which was 99.27% (10,903 false segmentation for 1,478,300 tokens). Most of the mistakes are due to the differences between the word and sentence segmentation schemes applied in the Szeged Treebank and in emToken.

2.2. Morphological Analyzer and Lemmatizer

The emMorph morphological analyzer (MA) integrated in the system was implemented using the Helsinki Finite-State Transducer (HFST) toolkit (Lindén et al., 2009). The morphological database is primarily based on the Hungarian computational morphology (Novák, 2003; Novák, 2015) originally created for the Humor morphological analyzer (Prószéky and Kis, 1999), which was extended with vocabulary from the morphdb.hu database (Trón et al., 2006). The grammar implemented in the constraint-based Humor formalism was converted to a finite-state description following the procedure described in Novák (2014). The morphological grammar development platform generates descriptions of allomorphs including their features and morph adjacency constraints from morpheme definitions applying a procedural rule system. The word grammar describing well-formed morpheme sequences (including the description of non-local constraints between morphemes) is defined using an extended finite-state automaton. All these data structures are implemented as a single finite-state lexical transducer in the HFST representation of the morphology, including a flag-diacritics-based representation of the word grammar automaton.

The hfst-lookup MA engine was extended in several ways to improve its performance. Dynamic FST composition was added to the implementation, so that the FST performing case conversion of capitalized words and those in all caps need not be composed with the lexical transducer off-line. This reduces runtime memory requirement of the MA to 1/3. Moreover, in addition to the lexical form, the MA can now also return the surface form of each morpheme.

The latter is used in the lemmatizer integrated in the system, emLem, which was implemented in Java. Most upstream tools do not need the amount of detail present in the analyses returned by the MA (e.g. compound members, derivational suffixes, alternative equivalent analyses of different granularities of lexicalized polymorphemic stems and their productive analyses). The emLem module merges compound members and derivational suffixes into a single stem using the surface form of non-final stem elements and the lexical form of the stem-final morpheme. It computes the resulting POS, adds the morphosyntactic features exposed by inflections, and discards identical results. The lemmatizer is also capable of optionally returning detailed analyses corresponding to each lemma (including surface forms of each morph) in addition to the lemmatized output. The algorithm implemented in the lemmatizer also handles non-trivial Hungarian word constructions (e.g. when inflectional suffixes are present in a non-word-final position) and returns a correct lemma also in those cases.

Previous morphological analyzers for Hungarian used various *ad hoc* tagsets. In contrast, the tagset used by emMorph and emLem contains tags suggested in the Leipzig Glossing Rules (Comrie et al., 2008) widely used by linguists. The tags in Wikipedia's list of glossing abbreviations[5] were also used and extended to include all Hungarian affixes and POS categories.

The MA was tested on the Hungarian Webcorpus (Halácsy et al., 2004) containing 589M words fully filtered. Word token coverage was found to be 99.36% corresponding to a word type coverage of 85.73%. As for analysis speed, hfst-optimized-lookup is 6.14 times faster than the original Humor implementation. Run-time data segment memory consumption of hfst-optimized-lookup is on the other hand at least 26.11 times higher than that of Humor (148 MB if case conversions are not composed with the lexicon vs. 5.6 MB). The engines themselves consume 4-6 MB memory.

2.3. POS Tagger

The emTag POS tagger is based on PurePOS (Orosz and Novák, 2013), which is the continuation and extension of HunPOS (Halácsy et al., 2007), the first POS tagger written directly to handle agglutination in Hungarian. Even though all the aforementioned systems are based on TnT (Brants, 2000), the well-known HMM tagger, each implementation has its own new features to handle Hungarian and similar highly agglutinating languages better.

The main advantage of emTag lies in its line of predecessors, which were developed with performance in mind with

[2] https://github.com/dlt-rilmta/quntoken/

[3] http://quex.sourceforge.net/

[4] https://github.com/zseder/huntoken

[5] https://en.wikipedia.org/wiki/List_of_glossing_abbreviations

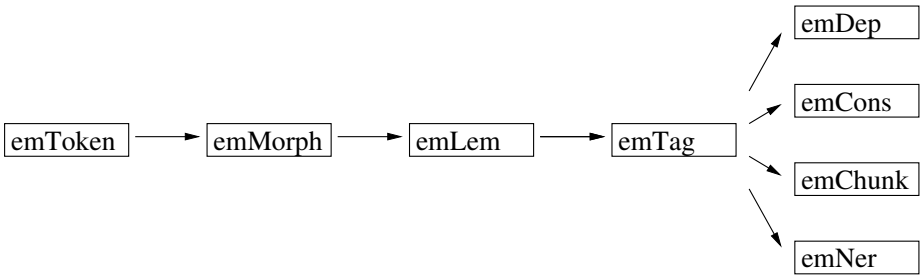


Figure 1: Modules of the e-magyar digital language processing system as they are built on one another.

new improved features. PurePOS is the first tagger in the line to add lemmatization support and the possibility to use pre-analyzed input, which essentially improves tagging and lemmatization performance.

From the technical perspective: the *de facto* standard CoNLL-style vertical input and output format is now supported along with the original PurePOS notation. The user is also to set different parameters on lemmatization (e.g. to cut characters from the right side of the token only (suffix) or use a longest common substring-based transformation (circumfixes)) and dump the created model into a human-readable format to check what has been learned. Even though this feature greatly improves the explanatory power of the model, its really rare among mainstream taggers, which makes emTag more valuable.

Nonetheless, POS taggers based on supervised learning heavily rely on the quality of the training corpus, the tagging scheme and the morphological analyzer. The performance of emTag is on a par with that of PurePOS (Orosz, 2015), which means a 97.58% full disambiguation accuracy on token level when applying lemmatization support as well.

2.4. Constituency and Dependency Parsers

The e-magyar toolchain includes constituency and dependency parsers, making it possible to apply two popular approaches for syntactic parsing. The input for both parsers is the output of the previous modules, i.e. tokenized and POS tagged sentences.

Our parser builds heavily on the techniques presented at the workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL), which was dedicated to the parsing of morphologically rich languages, such as Hungarian. An adapted version (Szántó and Farkas, 2014) of the Berkeley Parser (Petrov et al., 2006) – a stochastic context free model – was integrated into the system.

The preprocessing toolkit called magyarlanc (Zsibrita et al., 2013) also contains a dependency parser, based on the Bohnet parser (Bohnet, 2010), a language independent dependency parser. The model integrated was trained on the Szeged Dependency Treebank (Vincze et al., 2010). This dependency parser was integrated into the e-magyar toolkit, with small modifications, one of whose being the different morphological coding systems applied in e-magyar and in the Szeged Treebank. As the constituency and dependency parsers exploit the Hungarian version of the morphological coding system of the Universal Dependencies (UD) project (Vincze et al., 2017), the output of the emTag module needed to be converted to the UD morphology.

2.5. NP Chunker

The emChunk module identifies maximal noun phrases (NPs), i.e. NPs which are not part of any other higher level NPs. Its input is a text that had previously been processed in the toolchain, i.e. they had been segmented into words and sentences, and words are assigned their full morphological analyses. These pieces of information are necessary for the NP chunker module to be effective. The module assigns a tag to every token in the input text. The tag indicates whether the word is part of a maximal NP, and if yes, whether the NP has one or several components. If the latter, it also indicates whether the given word is an initial, medial or final component of the NP. The output keeps the analyses of the previous processing levels and adds the tags of the chunker module.

The emChunk module is based on HunTag3 (Endrédy and Indig, 2015), which is a sequential tagger for several NLP tasks using maximum entropy and HMM. Its predecessor is HunTag[6], which has been used, among others, for Hungarian named entity recognition (Simon, 2013) under the name hunner and for shallow syntactic analysis of Hungarian texts (Recski and Varga, 2010) under the name hunchunk. Depending on the training data, HunTag3 can be used for several sequential tagging tasks.

The gold standard dataset used here was a subcorpus of the Szeged Treebank (Csendes et al., 2005), which contains morphological, syntactic and named entity annotation as well. This subcorpus is actually the same as the one published under the name Szeged NER Corpus (Szarvas et al., 2006), which only contained morphological annotation (earlier in MSD then in UD format) and named entity annotation. The NP chunking annotation was generated from the constituency annotation of the Szeged Treebank, while the morphological annotation had to be converted to the format of emMorph.

The model was built by using the whole corpus as the training dataset, which comprises more than 220,000 tokens. Since there is no other Hungarian dataset containing all the required annotation layers in the required format to which the output of emChunk could be compared, here we

[6] https://github.com/recski/HunTag

cannot provide an evaluation of the module's performance. As Recski and Varga (2010) reports, hunchunk performs 86.06% F-score when recognizing maximal NPs in Hungarian texts.

2.6. Named Entity Recognizer

The emNer module is an automatic Named Entity Recognizer, which identifies named entities (NEs) in running text and assigns them to one of the predetermined categories. We follow the standard NE classes of CoNLL-2002 (Tjong Kim Sang, 2002) tagging person names, organization names, place names and the so called Miscellaneous category which mostly comprises everything else falling outside of the main categories.

Similarly to the other modules, the input of emNer is a text that had previously been processed in the toolchain, i.e. they had been segmented into words and sentences, and words are assigned their full morphological analyses. These pieces of information are necessary for the NER tagger module to be effective (Simon, 2013).

The module assigns a tag to every token in the text, indicating whether the given word is a named entity, and if yes, what category it belongs to, and whether it has one or several elements, and if the latter, whether the given word has an initial, medial or final position in the named entity. The output keeps the analyses of the previous processing levels and adds the tags of the NER tagger module.

The emNer module is also based on HunTag3, and the training data is the same as was in the case of emChunk. The model was also built by using the whole corpus as the training dataset, therefore here we cannot provide a thorough evaluation of the emNer module. What we can provide is the best performance of hunner trained on the 90% of the Szeged NER Corpus and tested on the remaining 10% which was 97.87% F-measure. However, cross-corpus and cross-domain evaluation always result in a 20-30% decrease in overall F-measure as reported by Nothman et al. (2008) and Simon (2013) among others, thus the performance of emNer highly depends on the input text type given by the user.

3. GATE Integration

The integration of the e-magyar modules described in Section 2. into a unified text analysis toolchain has been implemented in the GATE framework (Cunningham et al., 2011). During the integration, the main task was to enable the modules to take their input from the form corresponding to the *offset*-based annotation model of GATE and produce their output in this form as well. For this reason, we have created a GATE wrapper for each module that performs the required data conversions. It was also necessary to fit the non-Java language tools into the Java language framework: we have solved this by calling directly the binary of the non-Java modules.

Modules are built on one another as Figure 1 shows. The fixed basic processing chain consisting of a tokenizer, a morphological analyzer, a lemmatizer and a morphological disambiguator is followed by additional tools which utilizes the output of the fixed chain and which can be run independently of each other.

The modules are complemented by additional facilities. First, a human-readable format from the detailed morphological analysis is produced. Second, separated verbal particles and verbs are combined together based on the dependency analysis, so providing the full verbal form. Third, the IOB-type (specifically BIE-1) encoding provided by the emChunk and emNer modules are converted into a more convenient standalone (NP and NE) annotation format.

The processing chain can be used in four ways. On the website (see Section 4.), the user can simply copy-paste a short text and analyze this text by running the full processing chain with just a mouse click. For more serious text analysis tasks or for digital humanities research, the GUI of the GATE system called *GATE Developer* is recommended, into which the e-magyar chain can be easily installed. Installation instructions are available at https://github.com/dlt-rilmta/hunlp-GATE along with the entire system. If needed, the user can improve the system with adding custom built-in modules to the chain. For processing larger corpora, using GATE command line access is recommended, which is also available via the URL provided above. As a fourth method, the so-called *gate-server* can be used. This is another command line technology, which operates behind the e-magyar web service.

4. Online Interface

The project's objectives included the ability to access and use the text processing chain by users who are not necessarily familiar with the field of language technology. This demand is addressed by the online text analysis service[7] of e-magyar, which allows the user to easily test each of the analytic modules or even the entire tool chain via a web interface without using any other software than the web browser.

The text analyzer is based on a web service that uses GATE software libraries. It takes the text and the list of analytical modules that the user wants to run as input and provides GATE generated XML containing the annotations as output. The site processes the output XML and displays the data in an easily interpretable, visualized form.

The analyzer interface consists of two main parts: an input panel and an output panel. The text box on the input panel lets the user specify the text to be analyzed (currently, the length of the text is limited to 6000 characters), then the user can piece together the list of modules wanted to run on the text.

The result of the processing can be seen on the output panel. The analysis can be displayed in two different layouts: 'text' view and 'list' view. In 'text' view, tokens follow each other sequentially, annotations for each token show up in a small bubble when moving the cursor over a given token or clicking on it. In the case of separated verbal particles, the main verb is also highlighted. In 'list' view, each token has its own row in a tabular form, while annotations added by the text analyzer are placed in successive columns. This layout is more suitable for displaying lots of annotations together. In 'list' view, it is possible to filter

[7] http://e-magyar.hu/en/parser

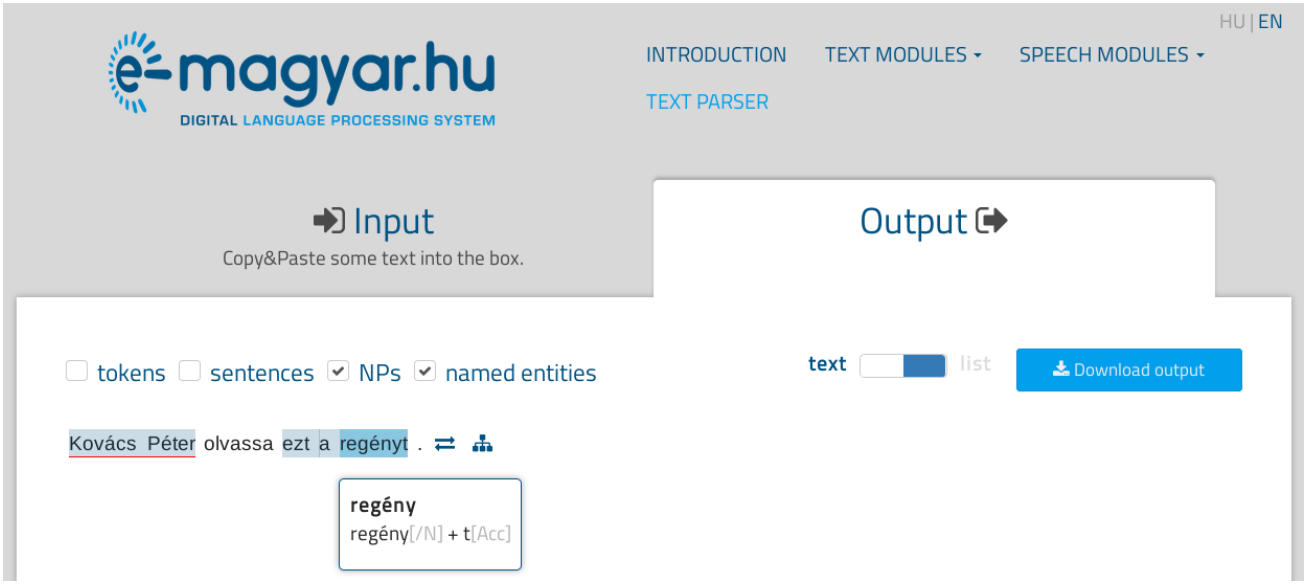


Figure 2: A screenshot of the 'text' view of the `e-magyar` online interface. The example sentence is *Kovács Péter olvassa ezt a regényt.* 'Kovács Péter reads this novel.' Lemma and morphological analysis is shown in a bubble for a token. Here we see that *regényt* is the accusative form of *regény* 'novel' which is a noun. Noun phrases are highlighted, and named entities are underlined.

the tokens based on different criteria: the user can filter for a word form, for elements of the morphological analysis, for POS tags or for several grammatical functions. In both views, it is possible to highlight certain segments (of one or more tokens) created by some analyzer modules: tokens, sentences, noun phrases, and named entities. An illustration can be seen in Figure 2. The results of the syntactic analyses can be accessed in the 'text' view by clicking on the icons next to each sentence: a dependency tree and a constituent tree diagram, the output of the dependency parser and the constituency parser, respectively.

The result of the analysis – the text with all added annotations – can be downloaded for further use. The downloaded zipped file contains three files: raw text sent for processing as a plain text file, the GATE generated output XML file, and an extract of the 'list' view in `tsv` format.

5. Acknowledgements

The work reported in this paper was supported by the Hungarian Academy of Sciences (Infrastructure Development Support Programme 2015/2).

6. Bibliographical References

Bohnet, B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of Coling 2010*, pages 89–97.

Brants, T. (2000). T'n'T: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on Applied natural language processing*, pages 224–231. Association for Computational Linguistics.

Comrie, B., Haspelmath, M., and Bickel, B. (2008). The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses.

Csendes, D., Csirik, J., Gyimóthy, T., and Kocsor, A. (2005). The Szeged Treebank. In *Lecture Notes in Computer Science: Text, Speech and Dialogue*, pages 123–131. Springer.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Endrédy, I. and Indig, B. (2015). HunTag3: a general-purpose, modular sequential tagger – chunking phrases in English and maximal NPs and NER for Hungarian. In *7th Language & Technology Conference, Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC '15)*, pages 213–218, Poznań, Poland, November. Poznań: Uniwersytet im. Adama Mickiewicza w Poznaniu.

Halácsy, P., Kornai, A., and Oravecz, C. (2007). Hunpos: An open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 209–212, Stroudsburg, PA, USA. Association for Computational Linguistics.

Halácsy, P., Kornai, A., Németh, L., Rung, A., Szakadát, I., and Trón, V. (2004). Creating open language resources for Hungarian. In *Proceedings of the 4th international conference on Language Resources and Evaluation (LREC2004)*, pages 1201–1204.

Lindén, K., Silfverberg, M., and Pirinen, T. (2009). HFST tools for morphology – an efficient open-source package for construction of morphological analyzers. In Cerstin Mahlow et al., editors, *State of the Art in Computational Morphology*, volume 41 of *Communications in Computer and Information Science*, pages 28–47. Springer

Berlin Heidelberg.

Nothman, J., Curran, J. R., and Murphy, T. (2008). Transforming Wikipedia into Named Entity Training Data. In *Proceedings of the Australasian Language Technology Association Workshop 2008*, pages 124–132.

Novák, A. (2014). A New Form of Humor – Mapping Constraint-Based Computational Morphologies to a Finite-State Representation. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1068–1073, Reykjavik, Iceland, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1207.

Novák, A. (2003). Milyen a jó Humor? [What is good Humor like?]. In *I. Magyar Számítógépes Nyelvészeti Konferencia [First Hungarian conference on computational linguistics]*, pages 138–144, Szeged. SZTE.

Novák, A. (2015). *A Model of Computational Morphology and its Application to Uralic Languages*. Ph.D. thesis, Roska Tamás Doctoral School of Sciences and Technology, Pázmány Péter Catholic University.

Orosz, Gy. and Novák, A. (2013). PurePos 2.0: a hybrid tool for morphological disambiguation. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, pages 539–545, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Orosz, Gy. (2015). *Hybrid algorithms for parsing less-resourced languages*. Ph.D. thesis, Roska Tamás Doctoral School of Sciences and Technology, Pázmány Péter Catholic University.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 433–440.

Prószéky, G. and Kis, B. (1999). A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 261–268, Stroudsburg, PA, USA. Association for Computational Linguistics.

Recski, G. and Varga, D. (2010). A Hungarian NP Chunker. *The Odd Yearbook. ELTE SEAS Undergraduate Papers in Linguistics*, pages 87–93.

Simon, E. (2013). *Approaches to Hungarian Named Entity Recognition*. Ph.D. thesis, PhD School in Cognitive Sciences, Budapest University of Technology and Economics.

Szántó, Zs. and Farkas, R. (2014). Special techniques for constituent parsing of morphologically rich languages. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 135–144, Gothenburg, Sweden, April. Association for Computational Linguistics.

Szarvas, G., Farkas, R., Felföldi, L., Kocsor, A., and Csirik, J. (2006). A highly accurate Named Entity corpus for Hungarian. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 1957–1960. ELRA.

Tjong Kim Sang, E. F. (2002). Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In Dan Roth et al., editors, *Proceedings of CoNLL-2002*, pages 155–158. Taipei, Taiwan.

Trón, V., Halácsy, P., Rebrus, P., Rung, A., Vajda, P., and Simon, E. (2006). Morphdb.hu: Hungarian lexical database and morphological grammar. In *Proceedings of LREC 2006*, pages 1670–1673.

Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., and Csirik, J. (2010). Hungarian Dependency Treebank. In *Proceedings of LREC 2010*, Valletta, Malta, May. ELRA.

Vincze, V., Simkó, K., Szántó, Z., and Farkas, R. (2017). Universal Dependencies and Morphology for Hungarian - and on the Price of Universality. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 356–365, Valencia, Spain, April. Association for Computational Linguistics.

Zsibrita, J., Vincze, V., and Farkas, R. (2013). magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In *Proceedings of RANLP*, pages 763–771.

iLCM - A Virtual Research Infrastructure for Large-Scale Qualitative Data

Andreas Niekler[1], Arnim Bleier[2], Christian Kahmann[1], Lisa Posch[2,3], Gregor Wiedemann[1], Kenan Erdogan[2], Gerhard Heyer[1], Markus Strohmaier[4, 2]

[1]Faculty of Mathematics and Computer Science, University Leipzig,
Augustusplatz 10, 04109 Leipzig, Germany
{aniekler, kahmann, wiedemann, heyer}@informatik.uni-leipzig.de

[2]Department of Computational Social Science, GESIS – Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6-8 50667 Cologne, Germany
{firstname.lastname}@gesis.org

[3]Institute of Interactive Systems and Data Science, Graz University of Technology,
Inffeldgasse 16c, 8010 Graz, Austria

[4]HumTec Institute, RWTH Aachen University
Theaterplatz 14, 52062 Aachen, Germany
markus.strohmaier@humtec.rwth-aachen.de

Abstract

The iLCM project pursues the development of an integrated research environment for the analysis of structured and unstructured data in a "Software as a Service" architecture (SaaS). The research environment addresses requirements for the quantitative evaluation of large amounts of qualitative data with text mining methods as well as requirements for the reproducibility of data-driven research designs in the social sciences. For this, the iLCM research environment comprises two central components. First, the Leipzig Corpus Miner (LCM), a decentralized SaaS application for the analysis of large amounts of news texts developed in a previous Digital Humanities project. Second, the text mining tools implemented in the LCM are extended by an "Open Research Computing" (ORC) environment for executable script documents, so-called "notebooks". This novel integration allows to combine generic, high-performance methods to process large amounts of unstructured text data and with individual program scripts to address specific research requirements in computational social science and digital humanities. `ilcm.informatik.uni-leipzig.de`

Keywords: infrastructure, text mining, computational social science, digital humanities, reproducible research

1. Introduction

Computational social science (CSS) is the interdisciplinary study of socio-cultural phenomena through new kinds of data and technologies. One of its central objectives is the extraction of useful and interpretable knowledge from potentially large behavioral digital datasets. The availability of these datasets has witnessed a fast increase in the past decades through the progressing digitization of social processes. Similar potentials, as well as challenges due to this development, can be observed in fields such as the digital humanities (DH) and communication science (CS).

In CSS, *unstructured* data (usually text) and *structured* data (e.g. metadata, survey data or log/sensor data) are both important sources of information. Through the digitization of social processes, they are available in large quantities. With the iLCM infrastructure, we plan to offer analysis capabilities for reusable and reproducible research with both data types. The manual analysis of qualitative text data is an integral part of the method repertoire of empirical research in various disciplines. The increasing availability of large amounts of digital or retro-digitized text has lead to a growing interest in computer-assisted evaluation methods. Different research methods such as quantitative or qualitative content analysis, discourse analysis or grounded theory methodology can be fruitfully combined with text mining methods (Wiedemann, 2016). Lexicometric methods such as keyword extraction, frequency- and co-occurrence analysis are already established and widely used in social science text analysis. Machine learning methods such as data-driven clustering of document collections using topic models (Blei, 2012; Stier et al., 2018) or the training of automatic classification methods for coding texts (Lemke et al., 2015; Stier et al., 2017; Posch et al., 2015; Wiedemann, 2018) begin outreaching into this field. As such applications of text mining help to combine qualitative and quantitative analysis perspectives, they are becoming increasingly relevant as a so-called "mixed methods" approach in the social sciences. Related research can be found in political science, sociology, contemporary history research or communication studies.

Users of computer-assisted text analysis from these disciplines face non-trivial technical and methodological challenges, especially when it comes to the requirement to analyze large amounts of text and reproducibility of research. The iLCM infrastructure provides solutions to these challenges. As shown in Fig.1, it consists of two major components. First, the *Leipzig Corpus Miner* (LCM) to (pre-)process and analyze large text collections such as news corpora, social media posts or parliamentary protocols. For this, the LCM provides a browser-based Graphical User Interface (GUI) targeting end-users from applied disciplines who are primarily interested in the application of generic text mining methods on their data. Second, the iLCM consists of an *Open Research Computing* (ORC) environment to run scripts in active executable documents, so-called "notebooks". By this, the ORC component allows analyzing structured data either extracted from texts by the LCM or imported from external datasets. Since the

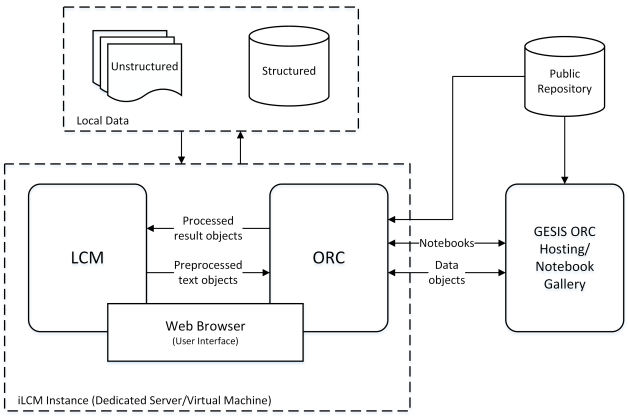


Figure 1: **iLCM architecture.** The iLCM architecture consists of two major components, the LCM to preprocess and analyze large amounts of unstructured text data, and the ORC component to further process structured data extracted from text or imported from external sources. Analysis scripts on structured data, so-called notebooks, will be published in a central repository.

creation of notebooks requires some programming skills, this component is targeted at more experienced end-users who want to realize more complex research designs on top of the generic text mining methods provided by the LCM. This hybrid architecture enables researchers to process text data in a standardized manner and extend their analyses by highly individualized program scripts that meet the requirements of their specific research tasks.

In this paper, we present our plans for implementing the iLCM environment and describe its contributions to CSS and DH research. We will cover the capabilities of the Leipzig Corpus Miner (LCM) for text analysis in Section 2. and introduce the idea of Open Research Computing (ORC) for social science applications and the processing of structured data in Section 3. In Section 4., we describe how the integration of both components will support the integrated analysis of unstructured and structured data in one environment. In the last sections, we present potential use cases to illustrate the benefits of the iLCM and conclude with a road-map for the ongoing development of the project.

2. Leipzig Corpus Miner

The LCM component of the environment is based on the previous research project "ePol" in which computer scientists together with political scientists created an analysis environment for large diachronic collections of news articles (Niekler et al., 2014; Wiedemann and Niekler, 2016). This previous project provided a solid basis for an requirements analysis process to design a software providing large scale text mining to social scientists. Relying on the experience gained in several studies conducted by the end-users from the targeted disciplines, we initiated a process of re-implementation to transfer the LCM prototype from the previous project into a universally applicable infrastructure. The LCM is not a stand-alone program, but rather a server infrastructure comprising a number of components including a document database (MariaDB[1]), an NLP pipeline for preprocessing text data (spaCy[2]), a full-text index (Solr[3]), a collection of text mining processes (for this, we rely on a selection of mature external packages for the R statistical programming language and additional own implementations), and finally a web application GUI (R Shiny[4]). To make the infrastructure available as a decentralized installation for end-users, it is embedded in a virtual machine ensemble (Docker[5]), which can be easily set up with predefined configuration scripts. Docker ensures the executability of the LCM on every system with a running docker environment. All required libraries, software dependencies, and R packages are provided with the docker containers. To set up the LCM container ensemble, we use docker compose[6] which automates the creation of all involved services in separate containers. In summary, the LCM integrates components for document management and retrieval, R scripting capabilities for text data, and a GUI for analysis process management and result visualization to enable researchers to conduct text mining on large collections in a systematic manner.

A screenshot of an early GUI-design is given in Figure 2. It shows the full-text of a retrieved document together with its metadata and some analysis results (keyword highlighting and named entity annotations).

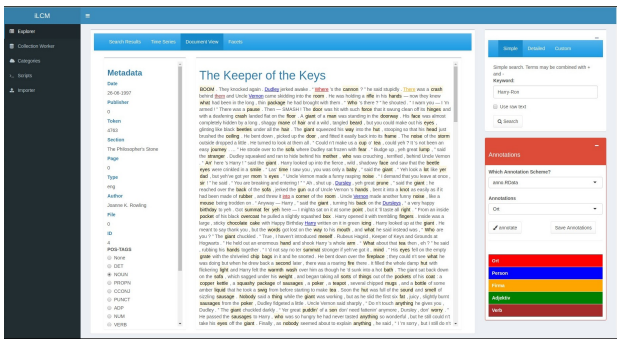


Figure 2: **LCM Web Application.** Screenshot of user interface (document view).

2.1. Document Management

Analyzing large text collections in the context of CSS poses specific requirements to document management. Researchers want to extract information from multiple corpora or compare results of analysis processes with respect to distinct (sub-)collections of documents from one corpus. Corpora can comprise several million of documents such as series of parliamentary protocols, full volumes of newspapers published over a period of several decades or millions of Twitter tweets regarding a certain hashtag. To handle such corpora, the LCM provides complex document management and search capabilities.

[1] http://mariadb.org
[2] http://spacy.io
[3] http://lucene.apache.org/solr
[4] http://shiny.rstudio.com
[5] http://www.docker.com
[6] http://docs.docker.com/compose

Corpora: The LCM allows managing multiple text corpora per user and also allows for sharing of corpora between users. Corpora are not closed at the time of their creation (import) but can be supplemented with additional documents at a later date. This allows the processing of continuous corpora, e.g. collections of newspaper articles, Wikipedia articles or social media data. In order to make the data import as easy as possible for users, import interfaces have been defined and implemented for XML, CSV, HTML, DOC, DOCX, RTF, PDF and plain text data. Furthermore, the Text Corpus Format (TCF) is supported (Heid et al., 2010). To process heterogeneous forms of text such as social media data (e.g. from Twitter or Facebook), administrative protocols, scientific publications or literary texts, a metadata management system is being introduced to define and manage corpus-specific metadata schemes. The LCM is designed to support multiple languages. We provide direct support for German and English text data. For other languages, a detailed documentation will be provided on how to customize configurations and resources (e.g. training models, stop word lists, lemma dictionaries) to add support for these languages.

Document Search: With a full-text search and metadata fields integrated into the LCM, documents can be filtered by keywords and metadata facets. A query language enables the complex combination of multiple search criteria, such as AND/OR combinations of search terms, the exclusion of certain terms, or the search for terms which must occur within a certain minimum distance from each other. Search result sets can be stored as individual collections which can serve as a basis for a refined search or for any further text mining analysis. Results from such analysis can then be aggregated with respect to the collection level. Moreover, each document can be displayed in a full-text view to allow for a close reading and qualitative checks of text mining results.

2.2. Analysis Features

During the previous ePol project and from the experiences of third parties using the LCM prototype, functional requirements were identified that promise to significantly increase the added value of using the LCM for social science projects. These functions are integrated into the new implementation of the iLCM infrastructure in a modular manner. Modularity is ensured by standardized wrappers around existing text analysis functions, mostly provided as R libraries. By relying on a well-defined wrapper layer around those pre-existing functions, we allow for an easy integration of different analysis packages of the same kind (e.g. different approaches of key term extraction, topic models or text classification) into the analysis process management and visualization provided by the LCM. In the following, we introduce the main modules.

Linguistic (pre-)processing: For most text mining applications, text data must be transformed into a numeric representation. For instance, word counts per document over the entire vocabulary can be represented as a vector. All document vectors of a collection together form a document-term matrix (DTM) on which various text statistical evaluations can be processed. To generate such a DTM representation

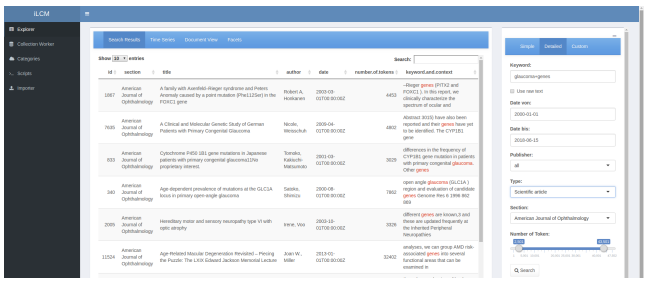


Figure 3: **LCM Document Search.** Screenshot of the document search interface.

of a collection, we need several preprocessing capabilities. The LCM provides sentence segmentation, multi-word expression detection, POS-Tagging, Named Entity Tagging, Named Entity unification, lowercasing, stemming, lemmatization, stop word removal, ngram tokenization and data pruning which can be applied to the text data as needed. Additionally, the process chain for the linguistic preprocessing of documents is extended by syntactic parsing which is needed for an extended text analysis (e.g. subject-object relationships). To fulfill those requirements we integrate the spaCy library. The open source Python library provides state of the art machine learning for natural language processing in multiple languages. Although spaCy is written in Python it can easily be integrated into the R environment with additional R packages (Arnold, 2017; Benoit and Matsuo, 2018).

Faceted Statistics: Result sets of search queries can be filtered with respect to certain indexed metadata. With this, a user can quickly generate a graphical representation of the document metadata for a search query or create time series for the search results. The occurrence of proper names in query related texts can also be displayed in an aggregated form which quickly allows for an overview of entities that can be identified with respect to a search query.

Term Extraction/Keyword in Context: Lists of meaningful terms can be calculated by the aggregation of their probabilities in different topics produced by a topic model. Those terms can be used to characterize the content of large collections or for further steps such as the creation of dictionaries or the identification of candidates for individual term analyses. Furthermore, terms can be displayed as ”Keyword in Context” (KWIC).

Word Frequencies/Dictionary Analysis: Similar to the measurement of document frequencies in relation to search results, the software provides possibilities for measuring term and document frequencies in document collections. According to the settings of a linguistic preprocessing chain, the articles in a collection are tokenized and the tokens are then counted, e.g. with reference to the publication dates of the articles. This allows the creation of time series for terms in absolute or normalized manner. Instead of single terms, also groups of terms representing semantically similar concepts, so-called dictionaries, can be used for frequency analysis.

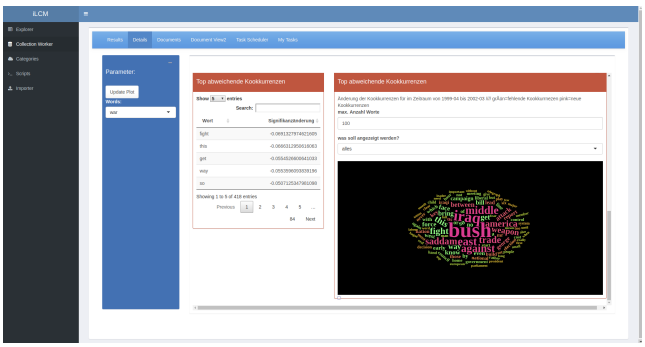


Figure 4: **LCM Context Volatility Results.** Screenshot of context volatility visualization.



Figure 5: **LCM Topic Model Results.** Screenshot of a topic model visualization.

Co-occurrence Analysis: For the identification of semantic correlations, the statistical significance of word co-occurrences is decisive in addition to their frequency. We integrate several ways to compute the statistical significance of the co-occurrence of tokens within defined contextual units. Possible contextual units are sentences, paragraphs or documents. Furthermore, it is possible to extract co-occurrences from contextual windows (e.g. n-left and n-right) and to analyse the ordering of the co-occurring words in the contextual units.

Context Volatility: Using the LCM enables researchers to deal with diachronic text data. The amount of contextual change for certain words over a period of time can be analyzed using a measure which captures this property (Kahmann et al., 2017). In the LCM several settings of measuring context change along with state of the art visualizations for the results are be implemented (cp. Fig. 4).

Topic Models: Topic models are probabilistic models that attempt to infer semantic clusters (which can be interpreted as topics) in document collections. As a result, topic models provide probability distributions over the set of all words for each topic (Blei, 2012). Furthermore, they provide probability distributions over the set of topics for each document. A topic can be semantically interpreted using the n most probable words it contains. For this purpose, the software provides sorted term lists for each calculated topic as well as an integration of the topic model result browser LDAvis (Sievert and Shirley, 2014) (cp. Fig. 5). We further add functionality to determine optimal hyperparameters for the modeling process and to measure its reliability over repeated runs (Maier et al., 2018).

Manual Annotation: Manual methods of qualitative data analysis usually work with categories. Those could be obtained inductively from the empirical material through reading and interpretation. More importantly, categories can be deductively guided and operationalized on the basis of existing theories (e.g. with the help of lists of terms, so-called dictionaries). This is supported by the definition of hierarchical category systems, encoding of entire documents or text segments with categories of the category system, measurement of intercoder matches and administration and export of the manually coded data. Due to the browser-based user interface, several annotators can simultaneously work on the same document set with a shared category system.

Supervised Classification: Categories for which a high intercoder reliability can be achieved, i.e. which are defined sufficiently unambiguously so that different encoders assign (mostly) the same categories for documents or passages of text, are also suitable for automatic classification procedures. Supervised methods for classification use given training examples to learn how to classify a data object into a defined class or category. Accordingly, the LCM offers the possibility of accepting text passages assigned to category X as positive training examples. Text passages in annotated documents that are not coded with X become negative training examples. A machine classifier learns textual features (especially word occurrences and their combinations) which suggest the existence of a certain class. This allows for an active learning scenario, where new text passages fitting to a certain category are identified within a collection automatically (Wiedemann, 2018). In an iterated process, automatic suggestions can be corrected by a human annotator to improve the classification model. In addition to text (passage) categorization, we further provide functionality for sequence classification to allow for complex information extraction.

2.3. Analysis workflows

The individual procedures already allow for interesting analysis of large text collections. But, in a CSS scenario, most benefit is to be expected from combined applications of the analysis modules in complex workflows. For instance, a topic model result is useful for interpretation in itself already. However, for content analysis, it is also interesting to use the topic model result as a filter for the original document collection. The result could be used to split a heterogeneous collection into thematically coherent sub-collections. These filtered sub-collections can be analyzed further as time series on the prominence of a theme. Also, further analysis procedures such as dictionary analysis could be applied to test hypotheses with respect to certain topics. In another scenario, frequency observations over time can be used to identify periods that indicate a similar use of terms and points in time at which changes occur (e.g. peaks in time curves). These periods of time can be used to divide a collection into distinct sub-collections on

which further comparative analyses such as co-occurrence analysis can be applied. In summary, the LCM allows for various combinations of the provided modules in creative ways which fit the research requirements of the end-users (Lemke et al., 2015). Information about the source collection, the selected analysis process and corresponding parameters for each analysis result are logged by the system in a way that allows for documentation of the conducted research.

3. Open Research Computing

The GUI-based framework of the LCM enables end-users to conduct standardized text mining workflows without any programming skills. However, individual and innovative research designs often demand more flexibility which is hard to achieve with generic pre-defined workflows accessed by point-and-click GUIs. Instead, such research designs can be supported well with script programming languages. The iLCM infrastructure integrates an Open Research Computing (ORC) environment to allow for the extension of LCM's generic analysis procedures by program scripts. In this way, the iLCM targets requirements of two user groups: beginners of text mining methods who benefit most from GUI-guided workflows as well as more advanced analysts who demand the flexibility of freely customizable scripts.

The ORC component is based on established open-source software such as Jupyter (Kluyver et al., 2016), Docker and Kubernetes[7], which are adapted to the requirements of the iLCM infrastructure. The development of these open-source frameworks is widely supported and hence provides the reliability of a well-maintained code base.

3.1. Notebooks

The ORC component extends the iLCM with an editor environment for program scripts which is operated via a web browser. The web editor enables the creation of scripts along with their documentation. Furthermore, scripts can be directly executed and the produced results can be visualized as output in the same document. The execution of the scripts themselves does not take place in the browser but on a server. This allows the processing of large data objects when server-side resources (memory, CPU) are available.

To fulfill important requirements for CSS, scripts can be extensively documented with markup code within the editor. Results from script executions such as numerical measures, tables or plots can be embedded in the same script document which allows tracing every single step of an analysis. Such actively executable documents, a combination of script code, its documentation, and corresponding results, are called "notebooks". For researchers this allows having the data, analysis scripts and their documentation available in one notebook for publication, sharing and re-use.

The ORC component extends the iLCM in two ways. First, the notebook environment is embedded in a virtual machine ensemble together with the LCM to closely integrate the GUI-paradigm for text analysis and the script paradigm to analyse structured data. Second, an ORC environment is publicly hosted by Gesis, the infrastructure partner of the iLCM project to allow for archiving, sharing and re-using of notebooks.

Various options are available to implement the notebook functionality, most notably R Notebooks and Jupyter Notebooks. For the iLCM, we opted for Jupyter Notebooks (Kluyver et al., 2016). Jupyter Notebooks have a wide community support and enable the use of numerous scripting languages, including R, Python, and Julia.

3.2. Notebook Gallery

Gesis will host a public instance of the ORC environment. This ORC instance is then extended with a repository on which notebooks can be published by users. The repository not only secures the long-term availability of published notebooks, it also serves as a notebook gallery that users can browse and search via keywords and metadata. The notebooks can be shared, edited and executed in a SaaS manner. The execution of shared notebooks is enabled by the import of the notebooks into the ORC environment by a process referred to as "cloning". This supports and enhances the reproducibility of research in the field of CSS.

The notebook gallery supports the research process in a number of ways. First of all, the gallery allows researchers to not only easily reproduce results of other researchers, but it also makes it easier to build on existing research through the cloning of notebooks. Furthermore, generic versions of different methods and analysis processes will be published as notebooks, enabling researchers to clone them and adapt them according to their own needs. Finally, the gallery makes it easy to discover learning materials published as notebooks, which supports researchers in learning new methods and analysis techniques.

We expect that the synergetic use of script-based analyses and generic text processing will support researchers in the development of analysis processes according to their project-specific requirements. We also expect a high impact of the architecture for teaching CSS methods.

4. Integration

The LCM is optimized for the generic processing of large amounts of text data. However, for experienced researchers, it is easy to identify needs for analyzes that go beyond the generic usage of text mining. The combination of standardized computer-linguistic evaluations in the LCM and customized evaluations in the ORC environment is an elegant solution. Therefore, the ORC environment is integrated into the iLCM architecture. Interfaces and functions will be implemented in the iLCM to easily exchange data objects between the LCM and the ORC environment (see Figure 1). For instance, output from the LCM component for further processing in the ORC environment can be entire text corpora, previously selected text collections, linguistically preprocessed DTMs, topic model results or any other information extracted from texts by the standardized LCM procedures.

In addition to further processing of LCM data objects, external data objects can be imported to combine complex analysis of both, structured and unstructured data. The integration of external resources offers considerable added

[7] http://kubernetes.io

value. The European infrastructure CLARIN hosts particularly valuable text corpora (Hinrichs and Krauwer, 2014). In addition, central search functionalities (Virtual Language Observatory, Federated Content Search) are available for the resources hosted in CLARIN, with which (sub-)corpora can be easily retrieved and processed in iLCM infrastructure. Interfaces defined by CLARIN for data import and export will be implemented in the iLCM. To facilitate the use of external resources with restricted access rights, iLCM will integrate a user rights management based on the Shibboleth[8] authentication and authorization. The rights management can be easily connected to identity providers such as managed by Gesis or by the CLARIN infrastructure to grant access for selected data objects and analysis to selected users only.

4.1. Expected Impact

We expect that the iLCM infrastructure will lead to a significant strengthening of the emerging field of CSS and the paradigm of reproducible research. The focus on reproducible research also goes hand in hand with the strengthening of the open access concept, since the publication of notebooks as descriptions of the research process is usually closely linked to the publication of the final manuscript of a research paper. The public availability of notebooks facilitates the review process and will eventually increase the chance to publish a final research manuscript as open access.

Another advantage for the digital humanities and the social sciences is the opportunity for interdisciplinary exchange. So far, the different disciplines have used computer-supported text analysis relatively independently of each other. In sociology, political science, media and communication sciences, (socio-)linguistics or historical studies, similar methods of text analysis have been used for content analysis or discourse research. There are only few examples of computer-supported text analyses that go beyond the boundaries of those disciplines. Since all disciplines are currently facing the challenge of integrating complex algorithmic evaluations into their research processes, we expect high benefits from cross-disciplinary methodological debates. The iLCM environment could be a hub to facilitate such debates by enabling scholars from different disciplines to easily share their methodical procedures.

Last but not least, the iLCM infrastructure is also ideal for teaching. On the one hand, it facilitates the transfer of basic skills in the field of text mining through the easy access to analysis methods with the GUI of the LCM. On the other hand, the infrastructure supports the teaching of more complex and individualized approaches to text mining by the ORC environment. The concept of notebooks further provides an interesting new format for term papers in which the student's methodical proceeding is documented comprehensibly and in a reproducible manner.

4.2. Exemplary Use-Cases

The capabilities of the integrated environment allow for flexible and extensive analysis applications. In order to illustrate the possibilities, we describe three potential use-cases.

Gender biases in Wikipedia: Wikipedia contains numerous biographies of important personalities. An examination of the contents of these biographies can clarify whether there are significant differences in the description of the biographies of male and female persons in contemporary history. A corpus of several thousand selected Wikipedia biographies, along with the gender of the personalities, could be imported into the LCM. In the LCM, two sub-collections, female and male biographies, are created and keywords are extracted from both. The respective complementary corpus serves as a comparative corpus. Furthermore, link structures between personalities of different genders can be investigated. These link structures can then be exported to the ORC environment.There, custom scripts can be created to further analyze the link network with respect to the question of gender biases.

Topic-person networks: Consider the following research question: Are people from politics and people from business more likely to be mentioned together in some economic policy contexts than in others? For answering this question, we analyze a collection of newspaper articles on economic policy using the iLCM infrastructure. First, the LCM component can be used for a named entity extraction to generate a co-occurrence network of politicians and business leaders. For the assignment of people to the category economics or politics, a link to a corresponding resource database such as DBPedia[9] can be used. Furthermore, the LCM component can be used to compute a topic model on the text of the news articles. The results of both computations can then be imported into the ORC environment for a more fine-grained analysis of the people network with respect to the topical contexts they co-occur in.

CSS teaching lab: Students of CSS can be taught to use state-of-the-art text mining with the help of the iLCM infrastructure. The GUI-based LCM component provides an easy and comprehensible access to basic analysis methods on large text collections such as full corpora of newspapers. In several in-class exercises, students are introduced to single methods such as key term extraction or topic modeling. With the help of the retrieval function, they are requested to select a suitable subcorpus from a full corpus of newspaper articles (e.g. NYTimes corpus) with respect to a given research question. Then, they apply the specific analysis method, try out different process parameters and in the end discuss the results in class. Since results may vary due to different initial collection retrieval processes or parameter settings, students will learn about the effects of each of the process steps by comparing their results. At the end of each term, students are requested to hand in a notebook created in the ORC component with an own study on the corpus used during the in-class exercises.

5. Conclusion

In this paper, we presented the iLCM, a SaaS infrastructure comprising a component for the analysis of large unstructured text collections (Leipzig Corpus Miner, LCM) and a

[8] http://shibboleth.net

[9] http://dbpedia.org

component for the (statistical) analysis of structured data (Open Research Computing, ORC). The combination of both allows for sophisticated computer-assisted text analysis in the context of computational social science and digital humanities research. It further fulfills requirements from the targeted user audience regarding different levels of previous knowledge about computational methods and reproducible research. This is achieved by providing a GUI-based approach to text analysis (LCM) and an opportunity to code own analysis scripts (ORC). The decentralized iLCM architecture is accompanied by a central repository of so-called 'notebooks', analysis scripts along with data, processed results and a process documentation to allow for archiving, sharing and re-use of text analysis workflows. By this, the iLCM enables social scientists to work with large amounts of data in order to produce pioneering results.

Requirements for the infrastructure were derived from experiences in a previous project in which a prototype of the LCM was implemented and used for several social science studies. For the current project, we plan to conduct additional user studies to learn more about the requirements of the targeted disciplines, to improve the usability of the GUI and to extend the text analysis capabilities. At the time of publication of this paper, the work on the iLCM is still in progress. Hence, the described environment is not feature complete yet. Nonetheless, an early version is already available on Github and will be developed further in our public repository.[10]

Our future work concentrates on the completion of the environment, sufficient documentation, open-source releases of the developed components and user testing in order to create a sustainable and convenient research environment for the social sciences and the humanities.

6. Acknowledgements

The project iLCM is funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) in the funding line for virtual research environments (FKZ/project number: 324867496). Funding period: 02/2017–01/2020

7. Bibliographical References

Arnold, T. (2017). A tidy data model for natural language processing using cleanNLP. *The R Journal*, 9(2):1–20.

Benoit, K. and Matsuo, A. (2018). spacyr: Wrapper to the spaCy NLP library.

Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, 55(4):77–84, April.

Heid, U., Schmid, H., Eckart, K., and Hinrichs, E. (2010). A corpus representation format for linguistic web services: The D-SPIN Text Corpus Format and its relationship with ISO standards. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC'10)*, pages 494–499.

Hinrichs, E. and Krauwer, S. (2014). The CLARIN Research Infrastructure: Resources and Tools for e-Humanities Scholars. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1525–1531, May.

Kahmann, C., Niekler, A., and Heyer, G. (2017). Detecting and assessing contextual change in diachronic text documents using context volatility. In *9th International Conference on Knowledge Discovery and Information Retrieval (KDIR'17)*.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J. B., Grout, J., Corlay, S., et al. (2016). Jupyter notebooks-a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87–90.

Lemke, M., Niekler, A., Schaal, G. S., and Wiedemann, G. (2015). Content analysis between quality and quantity. *Datenbank-Spektrum*, 15(1):7–14.

Maier, D., Waldherr, A., Miltner, P., Wiedemann, G., Niekler, A., Heyer, G., Keinert, A., Pfetsch, B., Häussler, T., Reber, U., Schmid-Petri, H., and Adam, S. (2018). Applying LDA topic modeling in communication research: Toward a valid and reliable methodology. *Communication Methods and Measures*, 12:1–26.

Niekler, A., Wiedemann, G., and Heyer, G. (2014). Leipzig Corpus Miner – A text mining infrastructure for qualitative data analysis. In *Terminology and Knowledge Engineering 2014 (TKE'14)*, Berlin.

Posch, L., Bleier, A., Schaer, P., and Strohmaier, M. (2015). The polylingual labeled topic model. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*. Springer.

Sievert, C. and Shirley, K. E. (2014). LDAvis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70. ACL, Baltimore.

Stier, S., Posch, L., Bleier, A., and Strohmaier, M. (2017). When populists become popular: Comparing Facebook use by the right-wing movement Pegida and German political parties. *Information, Communication & Society*, 20(9):1365–1388.

Stier, S., Bleier, A., Lietz, H., and Strohmaier, M. (2018). Election campaigning on social media: Politicians, audiences and the mediation of political communication on Facebook and Twitter. *Political Communication*.

Wiedemann, G. and Niekler, A., (2016). *Analyse qualitativer Daten mit dem "Leipzig Corpus Miner"*, pages 63–88. Springer, Wiesbaden.

Wiedemann, G. (2016). *Text Mining for Qualitative Data Analysis in the Social Sciences: A Study on Democratic Discourse in Germany*. Springer, Wiesbaden.

Wiedemann, G. (2018). Proportional classification revisited: Automatic content analysis of political manifestos using active learning. *Social Science Computer Review*, 36.

[10] See `ilcm.informatik.uni-leipzig.de` for further information.

CLARIN's Key Resource Families

Darja Fišer[1,2], Jakob Lenardič[1], Tomaž Erjavec[2]
Faculty of Arts, University of Ljubljana, Aškerčeva 2, 1000 Ljubljana, Slovenia
Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia
{darja.fiser, jakob.lenardic}@ff.uni-lj.si, tomaz.erjavec@ijs.si

Abstract

CLARIN is a European Research Infrastructure that has been established to support the accessibility of language resources and technologies to researchers from the Digital Humanities and Social Sciences. This paper presents CLARIN's Key Resource Families, a new initiative within the infrastructure, the goal of which is to collect and present in a uniform way the most prominent data types in the network of CLARIN consortia that display a high degree of maturity, are available for most EU languages, are a rich source of social and cultural data, and as such are highly relevant for research from a wide range of disciplines and methodological approaches in the Digital Humanities and Social Sciences as well as for cross-disciplinary and trans-national comparative research. The four resource families that we present each in turn are newspaper, parliamentary, CMC (computer-mediated communication), and parallel corpora. We focus on their presentation within the infrastructure, their metadata in terms of size, temporal coverage, annotation, accessibility and license, and discuss current problems.

Keywords: language resources, research infrastructure, open science, digital humanities and social sciences, CLARIN

1. Introduction

CLARIN is a European Research Infrastructure that has been established to support the accessibility of language resources and technologies to researchers from the Humanities and Social Sciences (Krauwer and Hinrichs, 2014). CLARIN's vision, mission and design are aimed at findability, accessibility, interoperability and re-usability of its resources, tools and services to support researchers in the Humanities and Social Sciences (SSH) (de Jong et al., 2018; De Smedt et al., 2018). At the time of writing, CLARIN has 20 member and 2 observer countries which provide numerous language resources and tools through certified data centres. Access to these resources is enhanced by the Virtual Language Observatory (VLO) portal which enables searching for resources and provides a uniform display of highly-granular Component MetaData Infrastructure (CMDI) metadata (Van Uytvanck et al., 2012).

Similar to other service-oriented e-research infrastructures (Chunpir et al., 2015), the developmental phase of the CLARIN infrastructure and services was followed by analyses of user experience and needs. Odijk (2014) looked into the problems with resource descriptions, granularity of metadata and resources, and string as well as faceted search in the VLO through the eyes of a linguist. Lušicky and Wissik (2016) conducted a similar survey of resource discovery that was tailored to the needs in translation studies. Sanders (2017) carried out two focus groups in order to obtain user experience and desiderata from two distinct types of users: humanities and social sciences researchers, and language technology and information technology experts. All these surveys reached similar conclusions: both SSH researchers as well as IT experts need easier access to the desired resources and more data type-specific guidance, including comprehensive metadata on the provenance and annotation of the resources, standard formatting, uniform concordancing and text analytics options that enable comparisons across corpora, as well as showcases and best practice examples from different disciplines that shows the value and utilization of the CLARIN infrastructure to fellow researchers in a real-life setting.

Inspired by these findings, the aim of this paper is to present current CLARIN's Key Resource Families, a new initiative within the infrastructure, the goal of which is to collect and present in a uniform way prominent data types in the network of CLARIN consortia that display a high degree of maturity, are available for most EU languages, are a rich source of social and cultural data, and as such highly relevant for research from a wide range of disciplines and methodological approaches in SSH as well as for cross-disciplinary and trans-national comparative research.

2. Survey protocol

For all data types except parallel corpora (see Section 3.4) we limited our search to the 22 countries that are either members or observers of CLARIN ERIC: Austria, Bulgaria, the Czech Republic, Denmark, Belgium, Estonia, Finland, France, Germany, Greece, Hungary, Italy, Latvia, Lithuania, the Netherlands, Norway, Poland, Portugal, France, Slovenia, and Sweden, and the UK.

Corpora were identified through the following procedure: (1) searching the VLO, (2) national CLARIN repositories and websites, (3) the META-SHARE[1] repository and LRE Map,[2] (4) Google Scholar[3], and (5) reaching out to the representatives of the national CLARIN consortia.

The primary aim of the surveys was to ascertain three general characteristics of the corpora. First, we checked whether the corpora are available through the VLO. Since the VLO is a CMDI-facet browser, which is an easy-to-use interface that ensures uniform access to resources from all the national CLARIN repositories (van Uytvanck et al., 2012), it is worthwhile to know whether there exist corpora that still lack VLO entries. In our surveys, we listed such corpora for future inclusion in the VLO. Second, we provided information on the availability of the corpora – that is, whether a corpus can be downloaded,

[1] http://metashare.csc.fi
[2] http://lremap.elra.info
[3] https://scholar.google.com

accessed through a concordancer or both, which is essential for re-use of the corpora. Finally, we described the presentation of their metadata in terms of (token) size, covered time period, levels of linguistic annotation and licence, highlighting missing information.

3. Survey results

3.1 Newspaper corpora

Due to large volumes, good document structure and rich and reliable metadata, one of the most traditional sources for corpus compilation projects are newspapers. They are an invaluable source of data for synchronic as well as diachronic studies of neologisms and other lexicographic phenomena (Smørdal Losnegaard and Inger Lyse, 2012), gender studies (Baker, 2012) and politology (Baker and McEnery, 2005).

We have identified 40 newspaper corpora, 30 of which are part of the CLARIN infrastructure. Due to space restrictions, we focus only on presenting their salient characteristics in terms of identification (i.e., whether the CLARIN corpora are also listed in the VLO), availability and metadata in this paper but provide a comprehensive overview in the report published on the CLARIN website.[4]

In relation to the metadata, information on size is available for the majority of the corpora (24 out of 30). *The Newspaper and Periodical Corpus of the National Library of Finland* (FIN-CLARIN, 2013)[5] is the largest, containing 8.6 billion tokens and covers the longest time period from 1771 to 2011. The smallest is the German *Tiger Corpus* (CLARIN-D, 2003), which contains 900,000 tokens. Information regarding annotation is available for 21 of the 30 corpora in the CLARIN infrastructure, almost half of which are tagged and annotated for syntactic dependencies, as is the case of the *Press 65-98* corpus (Språkbanken, 2017a). In terms of availability. 5 of the corpora can be downloaded (e.g. the *Tiger Corpus*), 7 can be accessed through concordancers (e.g. *The Newspaper and Periodical Corpus of the National Library of Finland* corpus) and 10 are both for download and accessible through concordancers (e.g. the *Press 65-98* corpus). It is worth noting that all of the 10 corpora available both for download and online querying are accessible through the same concordancer *Korp*[6], which is provided by the Swedish CLARIN repository. Information on license is available for 21 corpora, 17 of which are available under public CC-BY (Creative Commons) licences, 4 under the academic ELRA END USER licence, 1 under CLARIN PUB (i.e., publicly available) and 1 under the restrictive CLARIN RES licence. Only 16 of the 30 corpora that are part of the CLARIN infrastructure have VLO entries.

3.2 Parliamentary corpora

The second family of resources are corpora of parliamentary proceedings, which are a quintessential resource for a wide range of research questions from a number of SSH disciplines, such as Critical Discourse Analysis (Voutilainen, 2017), History (Pančur and Šorn, 2016) as well as Sociolinguistics (Rheault et al., 2015). Their most distinguishing characteristic is that they are transcriptions of spoken language produced in controlled and regulated circumstances. For this reason, they are rich in invaluable (sociodemographic) metadata as well as easily available under the Freedom of Information Acts set in place to enable informed participation by the public and to improve effective functioning of democratic systems, making the datasets even more valuable.

In total, we identified 22 corpora of parliamentary records. They exist for all CLARIN countries except Italy. We found one corpus per CLARIN country except in the case of the Czech Republic and Norway, where we found two corpora for each. Out of the 22 existing parliamentary corpora (the full list of which is available on the CLARIN webpage[7]), the following 16 are available within the CLARIN infrastructure:

(1) *Czech Parliament Meetings* (Pražák and Šmídl, 2012);
(2) *DK-CLARIN Almensprogligt korpus* (CLARIN-DK, 2012);
(3) *Transcripts of Riigikogu* (Center for Estonian Language Resources, 2017);
(4) *Eduskunta Corpus* (Parliament of Finland, 2017);
(5) *Hellenic Parliament Sittings* (clarin:el, 2017);
(6) *Proceedings of Norwegian Parliamentary Debates* (Språkbanken, 2016);
(7) *Riksdag's Open Data* (Språkbanken, 2017b);
(8) *PTPARL Corpus* (ELRA, 2017a);
(9) *SlovParl* (Pančur et al. 2016);
(10) *Hungarian National Corpus* (Váradi, 2005);
(11) *Hansard Corpus* (Hansard-corpus.org, 2017);
(12) *Lithuanian Parliament Corpus for Authorship Attribution* (Kapočiūtė-Dzikienė et al., 2017);
(13) *Talk of Norway* (Lapponi and Søyland, 2016);
(14) *Parliamentary Debates on Europe at the House of Commons (1998-2015)* (Truan, 2016a);
(15) *Parliamentary Debates on Europe at the Assemblée nationale (2002-2012)* (Truan, 2016b);
(16) *Parliamentary Debates on Europe at the Bundestag (1998-2015)* (Truan, 2016c).

The *Hansard Corpus* and the *Riksdag's Open Data* are the largest, comprising well over 1 billion tokens. Other corpora are significantly smaller (most between 10 and 100 million tokens) with the *PTPARL Corpus* (1 million tokens) and the *Czech Parliament Meetings* corpus (0.5 million tokens) being the smallest. All corpora cover various contemporary periods from the 1970s onwards except for the *Hansard Corpus*, which contains parliamentary sessions from the period between 1803 and 2005. In relation to linguistic annotation, all of the above corpora are tokenized, lemmatized and tagged for parts of speech except for the *Riksdag's Open Data* corpus, which is tokenized, lemmatized, MSD-tagged, and displays syntactic dependency relations, and the *Proceedings of*

[4] https://office.clarin.eu/v/CE-2017-1128-Newspaper-corpora.pdf.

[5] All referenced corpora are accompanied by their URL in the Sec. 8.

[6] https://www.kielipankki.fi/tuki/korp/. Finnish corpora are available through the *Kielipankki* (Language Bank of Finland) variant of Korp : https://korp.csc.fi/#?stats_reduce=word.

[7] https://office.clarin.eu/v/CE-2017-1019-Parliamentary-data-report.pdf.

Norwegian Parliamentary Debates corpus, which is only tokenized. In terms of availability, 10 corpora – that is, (1), (2), (5), (8), (9), and (12)-(16) – are available for download, corpora (4), (6), (10), (11) are available through online search environments and corpora (3) and (7) are available both for download and through online search environments. Corpora (4) and (7) can be accessed through through the Swedish *Korp*[8] concordancer, while corpus (6) can be queried through *Corpuscle*[9], the concordancer of the Norwegian consortium CLARINO. The rest (corpora 3, 6, 10, 11) are available through different dedicated non-CLARIN online environments. Corpora (1), (4), (5), (7), (9) and (8)-(16) are available under the CC-BY license; the license is unknown for corpora (10)-(12). All the corpora can be found through the VLO except for (5), which can be found in the repository of the Greek consortium; (7), which can be found in the repository of the Swedish consortium; and (11), which can be found on the website of the British observer.

3.3 Computer-mediated communication corpora

The third data type included in the initiative are corpora of computer-mediated communication (CMC) that are compiled from user-generated content (such as blogs, forums, and chats) as well as from interactions on social media (such as Twitter, Facebook, and WhatsApp). CMC corpora are a rich data type which can be collected in real time and for which a multitude of metadata can be harvested automatically. Such corpora therefore have a big potential for reuse and re-purposing in many fields of SSH. CMC corpora can serve as a basis for researching contemporary language variation and change (Danescu-Niculescu-Mizil et al., 2013) as well as changes in social and cultural dynamics (Östman and Turtiainen, 2016). Unlike most traditional text types, these corpora contain high levels of "noise" due to non-standard language phenomena that are frequent in informal on-line communication settings. Compilation and further dissemination of such corpora is hindered by terms of use and privacy protection limitations.

In total, we identified 21 CMC corpora covering 14 languages, 2 of which were multilingual. The full survey is again available through the CLARIN webpage.[10] 12 of the 20 identified corpora are part of the CLARIN infrastructure:

(1) the Estonian *Mixed Corpus: New Media* (Segakorpus 2011);
(2) the Finnish *Suomi24* corpus (Aller Media ltd., 2014);
(3) the Lithuanian *LITIS v.1. corpus* (Amilevičius and Petkevičius, 2016);
(4) the Dutch *SoNaR New Media Corpus* (Radboud University et al., 2013);
(5) the German *Dortmund Chat Corpus* (Technische Universität Dortmund, 2013);
(6) the Czech *Corpus of Contemporary Blogs* (Grác, 2011);
(7) the Slovene *Twitter corpus Janes-Tweet 1.0* (Ljubešić et al., 2017a);
(8) the Slovene *Wikipedia talk corpus Janes-Wiki 1.0* (Ljubešić et al., 2017b);
(9) the Slovene *Forum corpus Janes-Forum 1.0* (Erjavec et al., 2017a);
(10) the Slovene *Blog post and comment corpus Janes-Blog 1.0* (Erjavec et al., 2017b);
(11) the Slovene *News comment corpus Janes-News 1.0* (Erjavec et al., 2017c) and;
(12) the French *CoMeRe Repository* (Chanier et al., 2014).

The *Suomi24* corpus is the largest, containing 2.6 billion tokens while the German *Dortmund Chat Corpus* the smallest with 1 million tokens. Information on the time span is available only for corpora (1), (2), (3), (4) and (7) – among these, corpus (2) has the longest span, covering the period between 2001 and 2016, whereas corpus (7) contains data from the shortest period between 2013 and 2017. Corpus (1) is tokenized, corpus (2) is tokenized and morphosyntactically tagged, (4) and (5) are tokenized and part of speech tagged and corpora (7)-(11) are tokenized, word-level normalised (standardised), morphosyntactically tagged and lemmatized, while the annotation levels are unknown for corpora (3) and (6). Corpora (3) and (5) are available for download, corpora (4) and (6) for online searching and corpora (1), (2), and (7)-(12) both for download and online searching. Here corpus (2) is available through *Korp* and (7)-(12) through KonText and noSketch Engine, all CLARIN-provided concordancers. While the license is unknown for corpora (1) and (4), the rest are available under various CC licences, save for corpus (3), which is available under ACA_CLARIN-LT_End-User-License-Agreement_EN-LT – that is, under an academically-restricted licence. All of the corpora can be found on the VLO except for (1), which is available only on the website of the Estonian consortium.

3.4 Parallel corpora

The most recent family of resources from our survey are parallel corpora. Unlike the rest of the surveyed resource families, this overview had a broader scope not limited to official languages of CLARIN member states because other languages are also relevant for many researchers in the CLARIN network. In total, we were able to identify 106 parallel corpora, 81 of which are already part of the CLARIN infrastructure. Due to space restrictions, we provide here only a summary of the identified 81 corpora in the CLARIN infrastructure, and give the full account in the report available on the CLARIN webpage.[11]

The largest corpus is *Opus – the Helsinki Korp Version* (Tiedemann, 2004), which contains 2.7 billion tokens, whereas the smallest is *Text Corpus – EMEL* (clarin:el, 2016), which contains 43,000 tokens. 32 of the 81 corpora are multilingual, with the *Parallel Bible Corpus* (Christodouloupoulos and Steedman, 2014) and the *Tatoeba* corpus (Tiedemann, 2012a) containing data from more than 100 languages. Information on annotation,

[8] https://www.kielipankki.fi/tuki/korp/ and https://korp.csc.fi/#?stats_reduce=word.
[9] http://clarino.uib.no/korpuskel/page.
[10] https://office.clarin.eu/v/CE-2017-1064-Resources-for-computer-mediated-communication.pdf.
[11] https://office.clarin.eu/v/CE-2017-1095-Parallel-corpora-report.pdf.

which primarily pertains to the level of alignment in the case of parallel corpora, is available for 43 out of 81 corpora, 39 of these are sentence-aligned whereas the *Czech-English Manual Word Alignment* corpus (Mareček, 2008) and *GeFRePaC - German French Reciprocal Parallel Corpus* (ELRA, 2018) are also word aligned, *ParRus* (Bartis, 2017) is only paragraph aligned, *Czech and English abstracts of ÚFAL papers* (Rosa, 2016) is only document-aligned. In terms of availability, 45 corpora are available for download (e.g. the *Czech-English Manual Word Alignment* corpus); 23 corpora, such as *European Parliament Interpretation Corpus* (ELRA, 2017b), are only listed in the national repositories or the VLO but are unavailable (often due to various license restrictions, sometimes due to broken links); 10 corpora are available through concordancers (e.g. *Parallel Bible Corpus*) and 3 corpora - (the *OPUS* corpus (Tiedemann, 2012b), the *KOTUS Finnish-Swedish Parallel Corpus* (Institute for the Languages of Finland, 2014) and *The Norwegian-Spanish Parallel Corpus* (Hareide, 2013) - are available both for download and through concordancers. Licensing information is available for 68 corpora – 32 of these are available under CC-BY. 46 corpora can be found through the VLO while the rest are listed on the national repositories only (e.g., *Tatoeba* can only be found on the repository of the Greek consortium).

4. Discussion

We observed very uneven levels of inclusion into the CLARIN infrastructure across the types of resources that we surveyed. While many corpora have been added to national repositories, they still cannot be identified through VLO directly due to lacking, idiosyncratic or vernacular names, keywords or description fields, such as the Portuguese *PTPARL corpus* and the Danish *DK-CLARIN Almensprogligt korpus*. For some of the corpora, only older versions can be found through VLO, even though more recent ones are available on national repositories (e.g. *Hungarian National Corpus*). Granularity of the deposited resources ranges from complete archives to single-file records (e.g. 148 records of *Flemish Parliament Debates)*, which makes navigation and use of the resources much more difficult. The most frustrating of all the accessibility issues, however, are cases where successfully identified records lead to empty or broken download links, such as the parallel corpus *The Croatian-Slovenian Parallel Corpus* (Tadić, 2014).

The second group of major issues is the incomplete documentation (metadata) for many of the corpora that can range from corpus size (e.g. *Parallel Bible Corpus*), period (e.g. the German *Dortmund Chat* corpus), linguistic annotation (e.g. Finnish *Eduskunta corpus*), or license information (e.g. *SoNaR New Media Corpus*). For parallel corpora, the biggest issue in terms of metadata is the directionality of translations which is not available for most of the corpora (e.g. on the *MUSA Multilingual Multimodal corpus* (Piperidis et al., 2004)).

Of all four resource types discussed in this paper, the metadata on the parliament corpora are generally the best, since the only information that was lacking was related to annotation in the case of 2 out of the 11 corpora within the infrastructure. By contrast, the metadata on the parallel corpora are much poorer. Information regarding corpus size fares best as it is available for 67 (83%) of the corpora. Information regarding annotation, which primarily has to do with the level of textual alignment, is available for 43 (52%) corpora.

5. Conclusion

In this paper we provided a comprehensive overview of four key resource families across the CLARIN network: newspaper corpora, parliamentary corpora, corpora of computer-mediated communication, and parallel corpora. In addition to generating the first entry point for these rich and versatile resource families of its kind, and important secondary aim is to establish how their integration to the infrastructure could be further improved and enriched. In our future work we plan to extend CLARIN's Key Resource Families with other major datatypes, such as historical and learner corpora, as well as increase their visibility by developing a rich online research environment with tutorials and exercise kits for them as well as offer updates on their recent enhancements.

6. Acknowledgements

The work reported in this paper has been supported by the member countries and observers in the CLARIN ERIC, and it has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 676529 for project CLARIN-PLUS. We would like to thank all the national User Involvement Coordinators and researchers who have provided invaluable feedback on our surveys.

7. Bibliographical References

Baker, P. (2012). Corpora and Gender Studies. In Hyland, K., Huat, C.H., & Handford, M. (eds.): *Corpus Applications in Applied Linguistics*, pp. 100-116.

Baker, P. & McEnery, T. (2005). A corpus-based approach to discourses of refugees and asylum seekers in UN and newspaper texts. In *Journal of Language and Politics* 4 (2), pp. 197-226.

Chunpir, H. I., Ludwig, T., & Williams, D. N. (2015). Evolution of E-Research: From Infrastructure Development to Service Orientation. Proceedings of the Fourth International Conference on Design, User Experience, and Usability: Interactive Experience Design (DUX'15), pages 25–35. Cham, Springer. http://dx.doi.org/10.1007/978-3-319-20889-3.

Danescu-Niculescu-Mizil, C., West, R., Jurafsky, D., Leskovec, J. & Potts, C. (2013). No country for old members: user lifecycle and linguistic change in online communities. In *Proceedings of the 22nd international conference on World Wide Web*, pp. 307-318.

De Jong, F., Maegaard, B., De Smedt, K., Fišer, D., & Van Uytvanck, D. (2018). CLARIN: Towards FAIR and Responsible Data Science. Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC'18).

De Smedt, K., de Jong, F., Maegaard, B., Fišer, D., & Van Uytvanck, D. (2018) Towards an Open Science Infrastructure for the Digital Humanities: The Case of CLARIN. In Proceedings of the third international conference Digital Humanities in the Nordic Countries (DHN2018).

Krauwer, S. & Hinrichs, E. (2014). The CLARIN research infrastructure: resources and tools for e-Humanities scholars. Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pages 1525-1531), Portorož, Slovenia. European Language Resources Association (ELRA).

Lušicky, V., & Wissik, T. (2016). Evaluation of CLARIN services, User requirements, Usability, VLO, Translation Studies. Selected papers from the CLARIN Annual Conference 2016, Aix-en-Provence, France, pages 63-75. Linköping University Electronic Press, Linköpings universitet.

Odijk, J. (2014). Discovering Resources in CLARIN: Problems and Suggestions for Solutions. Utrecht University Repository, Netherlands.

Östman, S. & Turtiainen, R. (2016). From Research Ethics to Researching Ethics in an Online Specific Context. Media and Communication, 4 (4), pp. 66-74.

Pančur, P. & Šorn, M. (2016). Smart Big Data: use of Slovenian parliamentary papers in digital history. *Prispevki za novejšo zgodovino*, 56 (3), pages 130-146.

Rheault, L., Beelen, K., Cochrane, C. & Hirst, G. (2015). Measuring Emotion in Parliamentary Debates Using Methods of Natural Language Processing. http://ww.cs.toronto.edu/pub/gh/Rheault-etal-CPSA-2015.pdf .

Sanders, W. (2017). Focus Group on User Involvement. Sofia, Bulgaria. CLARIN. https://office.clarin.eu/v/CE-2017-1091-Focus-group-UI-2017-03-27.pdf

Smørdal Losnegaard, G. & Inger Lyse, G. A data-driven approach to Anglicism identification in Norwegian. In Andersen, G. (ed.): *Exploring Newspaper Language: Using the web to create and investigate a large corpus of modern Norwegian*.

Sobkowicz, A. (2016). Political Discourse in Polish Internet – Corpus of Highly Emotive Internet Discussions. http://nl.ijs.si/janes/wp-content/uploads/2016/09/CMC-2016_Sobkowicz_Political-Discourse-in-Polish-Internet.pdf.

Tiedemann, J. (2012a). Parallel Data, Tools and Interfaces in OPUS. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pages, 2214-2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Van Uytvanck, D., Stehouwer, H., and Lampen, L. (2012). Semantic metadata mapping in practice: The Virtual Language Observatory. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pages 1029-1034, Istanbul, Turkey. European Language Resources Association (ELRA).

Voutilainen, E. (2017). Parliamentary Records as Data for Linguistic Discourse Studies. http://videolectures.net/clarinplusworkshop2017_voutilainen_studies/.

8. Language Resource References

Aller Media ltd. (2014). The Suomi 24 Sentences Corpus. Distributed via Kielipankki, http://urn.fi/urn:nbn:fi:lb-2017021505.

Amilevičius, D. & Petkevičius, M. (2016). LITIS v.1, CLARIN-LT digital library in the Republic of Lithuania, http://hdl.handle.net/20.500.11821/11.

Bartis, I. (2017). ParFin 2016, ParRus 2016, Finnish-Russian / Russian-Finnish Parallel Corpus of Literary Texts. http://metashare.csc.fi/repository/browse/parfin-2016-parrus-2016-finnish-russian-russian-finnish-parallel-corpus-of-literary-texts/98799e72ec5811e6ba62005056be118e5894ba3bf8f54477acd3e0ac9bbd3ff1/.

Center of Estonian Language Resource. (2017). Transcripts of Riigikogu, https://keeleressursid.ee/en/resources/corpora.

Chanier,T., Poudat,C., Sagot, B., Antoniadis, G., Wigham,C. R. , Hriba, L.,Longhi, J. & Seddah, D. (2014). *The CoMeRe Repository.* https://hdl.handle.net/11403/comere.

Christodouloupoulos, C. & Steedman, M. (2014). Parallel Bible Corpus, https://github.com/christos-c/bible-corpus.

CLARIN-D. (2003). Tiger Corpus, http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.html.

CLARIN-DK. (2012) DK-CLARIN Almensprogligt korpus, https://clarin.dk/clarindk/item.jsp?id=dkclarin:986010.

clarin:el. (2016). Text Corpus – EMEL. http://hdl.grnet.gr/11500/AUTH-0000-0000-2C5A-B.

clarin:el. (2017). Hellenic Parliament Sittings. http://hdl.grnet.gr/11500/AEGEAN-0000-0000-2545-9.

Daelemans, W. (2005.) De Standaard Corpus, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11372/LRT-391.

ELRA. (2017a). PTPARL Corpus, http://catalog.elra.info/product_info.php?products_id=1179.

ELRA. (2017b). European Parliament Interpretation Corpus, http://catalog.elra.info/product_info.php?products_id=1145.

ELRA. (2018). GeFRePaC - German French Reciprocal Parallel Corpus. http://catalog.elra.info/product_info.php?products_id=633.

Erjavec, T., Ljubešić, N. and Fišer, D. (2017a). Forum corpus Janes-Forum 1.0, Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1139.

Erjavec, T., Ljubešić, N. and Fišer, D. (2017b). Blog post and comment corpus Janes-Blog 1.0, Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1138.

Erjavec, T., Ljubešić, N. and Fišer, D. (2017c). News comment corpus Janes-News 1.0, Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1140.

FIN-CLARIN. (2013). The Newspaper and Periodical Corpus of the National Library of Finland, Kielipankki Version, https://www.kielipankki.fi/corpora/.

Grác, M.. (2011). Corpus of contemporary blogs, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, http://hdl.handle.net/11858/00-097C-0000-000E-011B-8.

Hansard-corpus.org. (2017). *Hansard Corpus,* https://www.hansard-corpus.org/.

Hareide, L. (2013). The Norwegian-Spanish Parallel Corpus, Common Language Resources and Technology Infrastructure Norway (CLARINO) Bergen Repository, http://hdl.handle.net/11509/73.

Institute for the Languages of Finland. (2014). KOTUS Finnish-Swedish Parallel, http://islrn.org/resources/430-116-345-758-1.

Kapočiūtė-Dzikienė, J., Šarkutė, L. & Utka, A. (2017). Lithuanian Parliament Corpus for Authorship Attribution, CLARIN-LT digital library in the Republic of Lithuania, http://hdl.handle.net/20.500.11821/17.

Lapponi & Sørland. (2016). Talk of Norway, https://github.com/ltgoslo/talk-of-norway.

Ljubešić, N., Erjavec, T. and Fišer, D. (2017a). Twitter corpus Janes-Tweet 1.0, Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1142.

Ljubešić, N., Erjavec, T. and Fišer, D. (2017b). Wikipedia talk corpus Janes-Wiki 1.0, Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1137.

Mareček, D. (2008). Czech-English Manual Word Alignment, https://ufal.mff.cuni.cz/czech-english-manual-word-alignment.

Parliament of Finland. (2017). Plenary Sessions of the Parliament of Finland, Helsinki Korp Version. Distributed via the Language Bank of Finland. http://urn.fi/urn:nbn:fi:lb-2017020202.

Piperidis, S., Demiros, I., Prokopidis, P., Vanroose, P., Hoethker, A., Daelemans, W., Sklavounou, E., Konstantinou, M. & Karavidas, Y. (2004). MUSA Multilingual Multimodal Corpus, http://metashare.elda.org/repository/browse/musa-multilingual-multimodal-corpus/9f5d29a263c211e29fc5842b2b6a04d7a2d7266c56224f90ae4cb8f4757bf8ed/.

Pražák, A. & Šmídl, L. (2012). Czech Parliament Meetings, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11858/00-097C-0000-0005-CF9C-4.

Radboud University, CLST; Tilburg University, ILK; University of Twente, HMI. (2013). *SoNaR New Media Corpus*, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11372/LRT-1502.

Rosa, R. (2016). Czech and English abstracts of ÚFAL papers, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11234/1-1731.

Segakorpus. (2011). The Mixed Corpus : New Media, http://www.cl.ut.ee/korpused/segakorpus/uusmeedia/.

Språkbanken. (2016). Proceedings of Norwegian parliamentary debates, https://www.nb.no/sprakbanken/show?serial=oai:clarino.uib.no:stortinget&lang=.

Språkbanken. (2017a). Press 65-98, https://spraakbanken.gu.se/eng/resources.

Språkbanken. (2017b). The Riksdag's Open Data, https://spraakbanken.gu.se/eng/resources.

Tadić, M. (2014). Croatian-Slovenian Parallel Corpus, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, http://hdl.handle.net/11372/LRT-239.

Technische Universität Dortmund. (2013). The Dortmund Chat Corpus, http://www.chatkorpus.tu-dortmund.de/.

Tiedemann, J. (2004). Opus, Helsinki Korp Version [text corpus]. Distributed via Kielipankki, http://urn.fi/urn:nbn:fi:lb-2015102201.

Tiedemann, J. (2012a). Tatoeba. Distributed via clarin:el, http://hdl.grnet.gr/11500/ATHENA-0000-0000-2589-C.

Tiedemann, J. (2012b). Opus, http://opus.lingfil.uu.se/.

Truan, N. (2016a). Parliamentary Debates on Europe at the House of Commons (1998-2015) [Corpus]. ORTOLANG (Open Resources and TOols for LANGuage). https://hdl.handle.net/11403/uk-parl.

Truan, N. (2016b). Parliamentary Debates on Europe at the assemblée nationale [Corpus]. ORTOLANG (Open Resources and TOols for LANGuage). https://hdl.handle.net/11403/fr-parl/v1/.

Truan, N. (2016c). Parliamentary Debates on Europe at the Bundestag [Corpus]. ORTOLANG (Open Resources and TOols for LANGuage). https://hdl.handle.net/11403/de-parl/v1/.

Váradi, T. (2005). *Hungarian National Corpus*, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11372/LRT-345.

Indra: A Word Embedding and Semantic Relatedness Server

Juliano Efson Sales[†], Leonardo Souza[†], Siamak Barzegar[§], Brian Davis[§∓], André Freitas[δ] and Siegfried Handschuh[†]

†Department of Computer Science and Mathematics - University of Passau, Germany
[§]Insight Centre for Data Analytics - National University of Ireland, Galway
[∓]Department of Computer Science - Maynooth University, Ireland
δSchool of Computer Science - The University of Manchester, United Kingdom
{juliano-sales, siegfried.handschuh}@uni-passau.de, siamak.barzegar@insight-centre.org
brian.davis@.mu.ie, andre.freitas@manchester.ac.uk

Abstract

In recent years word embedding/distributional semantic models evolved to become a fundamental component in many natural language processing (NLP) architectures due to their ability of capturing and quantifying semantic associations at scale. Word embedding models can be used to satisfy recurrent tasks in NLP such as lexical and semantic generalisation in machine learning tasks, finding similar or related words and computing semantic relatedness of terms. However, building and consuming specific word embedding models require the setting of a large set of configurations, such as corpus-dependant parameters, distance measures as well as compositional models. Despite their increasing relevance as a component in NLP architectures, existing frameworks provide limited options in their ability to systematically build, parametrise, compare and evaluate different models. To answer this demand, this paper describes INDRA, a multi-lingual word embedding/distributional semantics framework which supports the creation, use and evaluation of word embedding models. In addition to the tool, INDRA also shares more than 65 pre-computed models in 14 languages.

Keywords: word embedding server, semantic relatedness server, semantic toolkit, corpus pre-processor

1. Introduction

Word embedding is a popular semantic model which represents words and sentences in computational linguistics systems and machine learning models. In recent years a large set of algorithms for both generating and consuming word embedding models (WEMs) have been proposed, which includes corpus pre-processing strategies, WEM algorithms or weighting schemes, vector compositions and distance measures (Turney and Pantel, 2010; Lapesa and Evert, 2014; Mitchell and Lapata, 2010). Determining the optimal set of strategies for a given problem demands the support of a tool that facilitates the exploration of the configuration space of parameters.

Furthermore, given the applicability and maturity achieved by these systems and models, they have been promoted from academic prototypes to industry-level applications (Loebbecke and Picot, 2015; Hengstler et al., 2016; Moro et al., 2015). In this new production scenario, a candidate tool should be able to scale to large number of requests and to the construction of models from large corpora, making use of parallel execution and traceability. From the functional point of view, *integrated corpus pre-processing*, *generation of predictive-based and count-based models* and *unified access as a service* are key features.

To support this demand, this paper describes INDRA, a word embedding/distributional semantics framework which supports the creation, use and evaluation of word embedding models. INDRA provides a software infrastructure to facilitate the experimentation and customisation of multilingual WEMs, allowing end-users and applications to consume and operate over multiple word embedding spaces as a service or library.

INDRA is available from two repositories (`github.com/Lambda-3/Indra` and `github.com/Lambda-3/IndraIndexer`) both licensed as open-source software. Additionally, INDRA also provides a Python client (`pyindra`) available via `pip` and from `github.com/Lambda-3/pyindra`.

2. Related Work

S-SPACE is a library to support the construction of count-based distributional methods unifying different approaches in a common JAVA API (Jurgens and Stevens, 2010). DEEPLEARNING4J[1], on the other hand, is a library which concentrates predictive-based models. DEEPLEARNING4J is also written in JAVA and its API contains methods to access word vectors and to find nearest neighbours (kNN).

GENSIM is one of the most popular word-embedding toolkits, mainly credited to its efficient implementation of nearest neighbours function (Řehůřek and Sojka, 2010). GENSIM is written in PYTHON and apart from its kNN function, it supports the generation of predictive-based models and methods to access word vectors.

Following a different motivation, DISSECT (DIStributional SEmantics Composition Tookit) focuses on vector compositions (Dinu et al., 2013). DISSECT is a PYTHON library containing methods to generate vector representation of sentences from the vector of its constituting words. DISSECT partially supports the generation of count-based models and brings an integrated baseline framework for evaluation purposes.

JOBIMTEXT is a semantic similarity tool that implements its own algorithm named JoBim (Biemann et al., 2013). The tool supports the construction of the JoBim model and

[1] https://deeplearning4j.org/

Features	INDRA	GenSim	DeepLearning4J	S-Space	JoBim	DISSECT
Word Embeddings						
Simple vectors	✓	✓	✓	✓		✓
Composed vectors	✓					✓
Translation-based vectors	✓					
Word embeddings as a service	✓					
Semantic Relatedness						
Word pair relatedness	✓	✓	✓	✓	✓	✓
Top-k nearest neighbors	✓	✓	✓	✓	✓	
Score-based nearest neighbors	✓				✓	
Translation-based relatedness	✓					
Composed-vector relatedness	✓					✓
Multiple score functions	✓					✓
Relatedness as a service	✓				✓	
Model Generation and Other Functions						
Integrated corpus pre-processing	✓					
Support to model generation	P/C	P	P	C	JoBim	C*
Support to sparse vector models	✓			✓	✓	✓
Bult-in word desambiguation					✓	
English pre-computed models	✓	✓			✓	
Multi-lingual pre-computed models	✓					
Multi-model querying	✓					

Table 1: List of functionalities and framework coverage. In the line *Support to model generation*, P stands for *predictive-based models* and C for *count-based models*. *DISSECT partially supports the generation of count-based models.

also calculates semantic relatedness of pairs of terms, finds nearest neighbours and offers a native web server.

EasyESA (Carvalho et al., 2014) and DInfra (Barzegar et al., 2015) are also two initiatives to deliver distributional semantics capabilities under a more specific set of distributional semantic models.

From the evaluation point of view, Barzegar et al. (2018c) defined a multi-lingual dataset to measure semantic relatedness in eleven languages.

Table 1 summarises a comparative analysis of the main frameworks and their features. Apart from INDRA, none of the listed frameworks gives support to corpora pre-processing (which will be detailed in Section 3.1.2.). Other limitations addressed by Indra are *(i)* the generation of both count-based and predictive-based models, *(ii)* the support for vector composition and *(iii)* the support for translation-based models.

Finally those libraries offer a limited set of pre-computed models, which makes the process of exploration time-consuming and computationally costly. INDRA aims at covering these gaps by providing an end-to-end infrastructure to build, consume and evaluate multi-lingual word embedding models.

3. Implementation Design

The INDRA PROJECT is divided into two major modules: INDRAINDEXER and INDRA. INDRAINDEXER is responsible for the generation of the models, whereas INDRA implements the consumption methods.

INDRA is designed to be a stand-alone library and also a web service. Figure 1 depicts the main components of its architecture. INDRAINDEXER supports the generation of WEMs directly from text files (Wikipedia-dump or plain-text formats), passing through the corpus pre-processing and multiword expression identification, to the model generation itself. INDRA dynamically builds the pipeline based on the metadata information produced during the model generation. This strategy guarantees that the same set of pre-processing operations are consistently applied to the input query. Additionally, the translation-based word embedding (Freitas et al., 2016; Barzegar et al., 2018b) can be conveniently activated in the pipeline as described in Section 4..

Different languages, domains and application scenarios require different parametrisations of the underlying embedding models. Together with the availability of pre-generated models, INDRA's system architecture favours the exploration of a large grid of parameters. INDRA currently shares more than 65 pre-computed models which varies in languages, model algorithms and corpora (general-purpose and domain-specific). The list of available models are in the Github project's Wiki.

3.1. Functionalities

Table 1 shows the functionalities implemented in INDRA, among which the following set deserves our attention: *text pre-processing*, *model generation*, *semantic relatedness*, *nearest neighbours*, *vector server*, *semantic relatedness*, *vector compositions* and the *support to translation-based models*.

3.1.1. Text Pre-processing

One important step in the construction of word embeddings models is pre-processing the texts. Defining the tokenisation strategy, which depends on the language, whether or

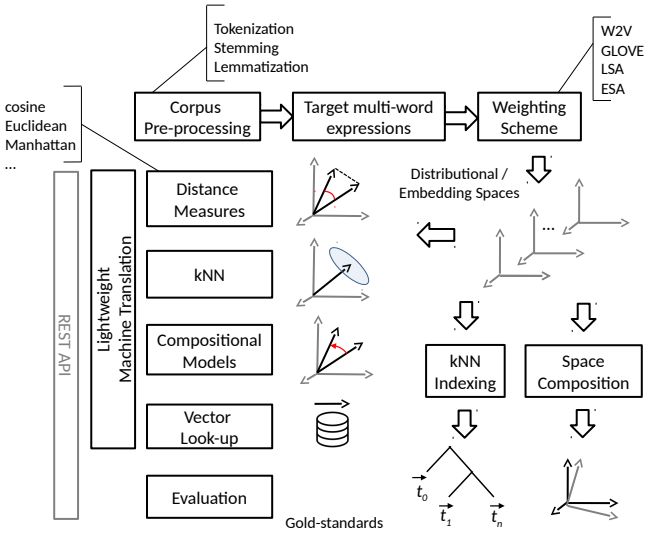


Figure 1: High-level architecture of the Indra framework.

not words must be lower-cased or stemmed is part of the pre-processing step. Furthermore, the pre-process strategy should be stated consistently during both construction and consumption phases as exemplified further.

The corpus pre-processor is responsible for defining the tokenisation strategy and the tokens' subsequent transformations. It defines, for example, if `United States of America` corresponds to a unique or to multiple tokens. *Stem* and *lowercase* are two other popular transformations also supported by the pre-processor.

INDRA uses the Lucene's `StandardTokenizer`, which implements the Unicode Text Segmentation algorithm based on the Word Break rules defined in the Unicode Standard Annex #29 (Davis and Iancu, 2017). Additionally, INDRA allows users to specify a customised list of multi-word expressions which will be considered a unique token, independently of the tokeniser rules. This mechanism allows, for example, modelling a unique vector for named entities such as *Nelson Mandela* and *Republic of Austria*.

As in the context of WEM numbers are usually disregarded tokens, the pre-processing step allows replacing them by a default placeholder (¡NUMBER¿). INDRA pre-processor also allows specifying stopwords, whose occurrences are removed from the text. Table 2 shows the full list of operations supported by the pre-processor.

The pre-processor is defined as a package that is attached to both INDRAINDEXER and INDRA in order to guarantee that the consuming functions apply the same set of operations in retrieval time.

3.1.2. Model Generation

INDRAINDEXER is the module responsible for the generation of word embedding models. It defines a unified interface to generate predictive-based models (e.g. Skip-gram (Mikolov et al., 2013) and Global Vectors (Pennington et al., 2014)) and count-based models (e.g. Latent Semantic Analysis (Dumais et al., 1988) and Explicit Semantic Analysis (Gabrilovich and Markovitch, 2007)) whose implementation comes from the libraries DeepLearning4J[2] and S-Space (Jurgens and Stevens, 2010) respectively. In addition to creating a unified interface for WEM algorithms, INDRAINDEXER integrates the corpus pre-processor package.

INDRAINDEXER receives as input the pre-processed corpus and outputs the vectors in binary files in a format compatible with GenSim (Řehůřek and Sojka, 2010). In addition to the vector file, INDRAINDEXER also generates a metadata file containing all the parameters from both the pre-processing and generation steps. Figure 3 shows an example of a metadata file.

```
{
  {
    "windowSize" : 5,
    "minWordFrequency" : 5,
    "corpusMetadata" : {
       "corpusName" : "wiki-2014",
       "stopWords" : ["been", "don't", ...],
       "replaceNumbers" : false,
       "applyStemmer" : 3,
       "removeAccents" : true,
       "maxTokenLength" : 100,
       "minTokenLength" : 3,
       "description" : null,
       "language" : "en",
       "encoding" : "UTF-8",
       "applyLowercase" : true
    },
    "vocabSize" : 1181258,
    "sparse"  :false,
    "model" : "w2v",
    "dimensions" : 300
}
}
```

Figure 2: Example of metadata file generated by INDRAINDEXER which describes how the user configured both the pre-processor and the WEM generator.

During the consumption phase, INDRA applies the same set of options to guarantee consistence. For instance, let's assume a given model was generated by applying the stemmer and lowercase to the tokens. It means that the term `University` is represented in the model as `univers`. When it is required to retrieve the vector representation of `University`, INDRA guarantees this consistence by executing the pre-processing steps in the query at runtime. This method simplifies the execution of experiments that consumes models using different set of pre-processing transformations.

3.1.3. Nearest Neighbours

Given a term and an integer k, the Nearest Neighbours function lists the set of its k closest terms. This method is applied, for instance, in *topic modelling* (Řehůřek and Sojka, 2010) and *vocabulary expansion* (Atzori et al., 2018). INDRA implements this function using the SPOTIFY ANNOY

[2]https://deeplearning4j.org/

Parameter	Description/Options
input format	Wikipedia-dump format or plain texts from one or multiple files.
language	14 supported languages.
set of stopwords	a set of tokens to be removed.
set of multi-word expressions	set of sequences of tokens that should be considered a unique token.
apply lowercase	lowercase the tokens.
apply stemmer	applies the Poter Stemmer in the tokens.
remove accents	remove the accents of words.
replace numbers	replaces numbers for the place holder $<NUMBER>$.
min	set a minimum acceptable token size.
max	set a maximum acceptable token size.

Table 2: Parameters supported by the INDRA's pre-processing package.

library [3], since a preliminary study suggests ANNOY's performance is an order of magnitude better performing than GENSIM's (Řehůřek, 2014). In addition to the identification of term's neighbours, the function also accepts a vector as input.

Another related function present in INDRA is the selection based on thresholds, in which INDRA gets a query term and a set of target terms as inputs, and returns those target terms whose relatedness score is greater (or lower) than a given threshold. The threshold can be determined both statically or dynamically(Freitas et al., 2012).

3.1.4. Vector Server

As a primary use, INDRA acts as a central repository of WEMs, serving vectors for terms in different languages and models. The set of pre-processed models allows the user to experiment different WEMs configurations as a one-stop-shop fashion. INDRA can act as a central server in an enterprise context, or as a local library in more constrained environments.

3.1.5. Semantic Relatedness

Natural language understanding systems use semantic relatedness in fine-grained tasks such as *word disambiguation* (Freitas et al., 2013) or more coarse-grained such as *paraphrase detection* (Sales et al., 2016; Silva et al., 2018), *semantic parsing* (Sales et al., 2018) and *question answering* (Freitas, 2015). INDRA implements two semantic relatedness methods. The first is the pair-wise semantic relatedness in which the user provides pairs of terms to calculate their semantic relatedness. The other option is integrated to the nearest neighbours function which returns the relatedness of the *k* closest terms.

Additionally Indra can support the application of various distance or correlation measures (Lapesa and Evert, 2014). Currently INDRA supports more than ten different distance and correlation functions, including *Cosine*, *Jaccard*, *Euclidean* and *Spearman Correlation*.

3.1.6. Vector Compositions

In simple terms, *vector composition* aims at generating a vector representation of phrases and sentences from the combination of individual vectors of its compound terms(Kartsaklis, 2014). For example, the vector representation of `modern Democratic Party` is generated by the composition of the corresponding vectors of the three compounding terms `modern`, `Democratic` and `Party`. Currently, INDRA implements three composition methods (*Sum*, *Normalised Sum* and *Average*) and supports the extension of user-defined functions as described in Section 3.2..

Vector composition is automatically associated to the semantic relatedness function or the retrieval of vectors. Whenever a expression comprehending more than one token is submitted, INDRA composes their corresponding vectors before executing the required function.

3.1.7. Support to Translation-based Models

Some languages do not have large text corpora publicly available. As word embedding models are sensitive to the corpus size, (Freitas et al., 2016; Barzegar et al., 2018a) propose the use of *translation-based models*. In simple words, the translation-based strategy translates the original query terms to a second language for which a high quality WEM is available. INDRA gives native support to this operation as described in Section 4..

3.2. Extensibility

INDRA implements a plugin-based extensible mechanism built on the top of the JAVA SERVICE API which allows including new *compositional methods*, *score functions* and *threshold functions* without recompiling Indra's code. To do so, it is required to pack the new functions' implementations in a JAR file and place it in the INDRA's *classpath*[4].

4. Use Examples

INDRA's service exposes the functions as POST methods, whose data are passed as a JSON payload. For simplicity, we suppress the request headers to concentrate our attention in the payload itself.

Every request has at least three mandatory fields: *language*, *model* and *corpus*. The first naturally specifies the request's language. The second and the third name respectively the algorithm and the corpus from which the word embedding model were generated. This trio is the model's unique identifier.

[3] https://github.com/spotify/annoy

[4] For more information about The Java Archive (JAR) and CLASSPATH, please refer to official Java documentation.

Word Embedding: Figure 3 shows a payload to the endpoint `/vectors` which returns the respective word embedding vectors of the `terms`. In the case a term is composed of more than one token, `termComposition` is applied.

```
{
  "corpus": "googlenews300neg",
  "model": "W2V",
  "language": "EN",
  "terms": ["love", "best of you"],
  "termComposition" : "AVERAGE"
}
```

Figure 3: Payload to request the word embedding of the term *love* and the expression *best of you*.

Pair Relatedness: The endpoint `/relatedness` returns the semantic relatedness of the `pairs`. The relatedness operation is defined by the field `scoreFunction` as shown in Figure 4. In the case that `termComposition` is not defined, the default function is used.

```
{
    "corpus": "wiki-2014",
    "model": "ESA",
    "language": "PT",
    "scoreFunction": "COSINE",
    "pairs": [{
        "t1": "economia",
        "t2": "Rio de Janeiro"
        },
        {
        "t1": "economia",
        "t2": "soja"
    }]
}
```

Figure 4: Payload to request the cosine relatedness of two pairs of terms in Portuguese.

One-to-many Relatedness: The endpoint `/relatedness/otm` returns the semantic relatedness of `one` term against a set of `many` terms. Similarly to the previous operation, the relatedness operation is defined by the field `scoreFunction` as shown in Figure 5.

Nearest Neighbours: The nearest neighbours function is exposed in two methods. The endpoint `/neighbors/relatedness` returns the relatedness score between the target terms and theirs top-k neighbours, according to the payload depicted in Figure 6.
When submitting the same payload to the endpoint `/neighbors/vectors`, the service returns the list of the neighbours and their respective vectors.

Translated-based Word Embbedings: The requests support the translated-based function, in which the vectors is extracted from the corresponding English corpus after translating the terms from the original query. The translated-based function is activated by appending `"mt"=true` in the payload. INDRA offers seven pre-computed light-weight translation models.
For a complete description of the methods and parameters, please refer to the project documentation.

```
{
    "corpus": "wiki-2014",
    "model": "ESA",
    "language": "EN",
    "scoreFunction": "JACCARD",
    "one": "Germany",
    "many" : ["France", "China", "Brazil"]
}
```

Figure 5: Payload to request the Jaccard relatedness of the implicit pairs [Germany, France], [Germany, China] and [Germany, Brazil].

```
{
  "corpus": "wiki-2014",
  "model": "GLOVE",
  "language": "SV",
  "topk": 10,
  "terms": ["ekonomi", "flicka", "frihet"]
}
```

Figure 6: Payload to request the 10 most related terms individually to *ekonomi*, *flicka* and *frihet*. This call returns three set of 10 terms, each one corresponding to one of the terms.

5. Python Client

Our project also offers a client to access the service from Python application. The `pyindra` package is available in the `pip` repository.
The client source code is at `https://github.com/Lambda-3/pyindra`.

6. Summary

Many applications of word embedding models require the customisation of the models in the direction of domain-specific vocabularies, specific languages or specific semantic approximation behaviour (e.g. paradigmatic vs syntagmatic behaviour), distance measures as well as compositional models. This work introduces the INDRA framework which manages the complexity of experimenting and using word embedding models in exploratory scenarios and production environments. INDRA also shares more than 65 pre-computed models and is available as an open-source software.

7. Acknowledgment

This publication has emanated from research partially supported by the National Council for Scientific and Technological Development, Brazil (CNPq).

8. References

Atzori, M., Balloccu, S., and Bellanti, A. (2018). Unsupervised singleton expansion from free tex. In *2018 IEEE*

Twelfth International Conference on Semantic Computing (ICSC), Jan.

Barzegar, S., Sales, J. E., Freitas, A., Handschuh, S., and Davis, B. (2015). Dinfra: A one stop shop for computing multilingual semantic relatedness. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 1027–1028, New York, NY, USA. ACM.

Barzegar, S., Davis, B., Handschuh, S., , and Freitas, A. (2018a). Multilingual semantic relatedness using lightweight machine translation. In *2018 IEEE Twelfth International Conference on Semantic Computing (ICSC)*, Jan.

Barzegar, S., Davis, B., Handschuh, S., and Freitas, A. (2018b). Multilingual semantic relatedness using lightweight machine translation. In *IEEE International Conference on Semantic Computing*. IEEE.

Barzegar, S., Davis, B., Zarrouk, M., Handschuh, S., and Freitas, A. a. (2018c). Semr-11: A multi-lingual gold-standard for semantic similarity and relatedness for eleven languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Biemann, C., Riedl, and M. (2013). Text: Now in 2d! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95, April.

Carvalho, D. S., Calli, C., Freitas, A., and Curry, E. (2014). Easyesa: A low-effort infrastructure for explicit semantic analysis. In *Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference, ISWC 2014, Riva del Garda, Italy, October 21, 2014.*, pages 177–180.

Davis, M. and Iancu, L. (2017). Unicode text segmentation. Technical report, The Unicode Consortium, June.

Dinu, G., Pham, N. T., and Baroni, M. (2013). Dissect - distributional semantics composition toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 31–36, Sofia, Bulgaria, August. Association for Computational Linguistics.

Dumais, S. T., Furnas, G. W., Landauer, T. K., Deerwester, S., and Harshman, R. (1988). Using latent semantic analysis to improve access to textual information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '88, pages 281–285, New York, NY, USA. ACM.

Freitas, A., Curry, E., and O'Riain, S. (2012). Distributional approach for terminological semantic search on the linked data web. In *27th ACM Symposium On Applied Computing (SAC), Semantic Web and Applications (SWA)*, Sept.

Freitas, A., Oliveira, J. G., O'Riain, S., da Silva, J. C. P., and Curry, E. (2013). Querying linked data graphs using semantic relatedness: A vocabulary independent approach. *Data Knowl. Eng.*, 88:126–141.

Freitas, A., Barzegar, S., Sales, J. E., Handschuh, S., and Davis, B., (2016). *Semantic Relatedness for All (Languages): A Comparative Analysis of Multilingual Semantic Relatedness Using Machine Translation*, pages 212–222. Springer International Publishing, Cham.

Freitas, A. (2015). *Schema-agnositc queries over large-schema databases: a distributional semantics approach.* Ph.D. thesis, Digital Enterprise Research Institute (DERI), National University of Ireland, Galway.

Gabrilovich, E. and Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1606–1611, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Hengstler, M., Enkel, E., and Duelli, S. (2016). Applied artificial intelligence and trust - the case of autonomous vehicles and medical assistance devices. *Technological Forecasting and Social Change*, 105(Supplement C):105 – 120.

Jurgens, D. and Stevens, K. (2010). The s-space package: An open source package for word space models. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, pages 30–35, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kartsaklis, D. (2014). Compositional operators in distributional semantics. *Springer Science Reviews*, 2(1):161–177, Dec.

Lapesa, G. and Evert, S. (2014). A large scale evaluation of distributional semantic models: Parameters, interactions and model selection. *Transactions of the Association for Computational Linguistics*, 2:531–545.

Loebbecke, C. and Picot, A. (2015). Reflections on societal and business model transformation arising from digitization and big data analytics: A research agenda. *The Journal of Strategic Information Systems*, 24(3):149 – 157.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *ICLR Workshop*.

Mitchell, J. and Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Moro, S., Cortez, P., and Rita, P. (2015). Business intelligence in banking: A literature analysis from 2002 to 2013 using text mining and latent dirichlet allocation. *Expert Systems with Applications*, 42(3):1314 – 1324.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA. `http://is.muni.cz/publication/884893/en`.

Řehůřek, R. (2014). Performance shootout of nearest neighbours: Querying. https://rare-technologies.com/performance-shootout-of-nearest-neighbours-querying/.

Sales, J. E., Freitas, A., Davis, B., and Handschuh, S.

(2016). A compositional-distributional semantic model for searching complex entity categories. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 199–208, Berlin, Germany, August. Association for Computational Linguistics.

Sales, J. E., Freitas, A., and Handschuh, S. (2018). An open vocabulary semantic parser for end-user programming using natural language. In *2018 IEEE Twelfth International Conference on Semantic Computing (ICSC)*, Jan.

Silva, V., Handschuh, S., and Freitas, A., (2018). *Recognizing and Justifying Text Entailment through Distributional Navigation on Definition Graphs*. AAAI.

Turney, P. D. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January.

A UIMA Database Interface for Managing NLP-related Text Annotations

Giuseppe Abrami and Alexander Mehler
Text-technology Lab, Goethe University Frankfurt
Robert-Mayer-Straße 10, 60325 Frankfurt am Main
{abrami, mehler}@em.uni-frankfurt.de

Abstract

NLP and automatic text analysis necessarily involve the annotation of natural language texts. The Apache *Unstructured Information Management applications* (UIMA) framework is used in several projects, tools and resources, and has become a *de facto* standard in this area. Despite the multiple use of UIMA as a document-based schema, it does not provide native database support. In order to facilitate distributed storage and enable UIMA-based projects to perform targeted queries, we have developed the *UIMA Database Interface* (UIMA DI). UIMA DI sets up an environment for a generic use of UIMA documents in database systems. In addition, the integration of UIMA DI into rights and resource management tools enables user and group-specific access to UIMA documents and provides data protection. Finally, UIMA documents can be made accessible for third party programs. UIMA DI, which we evaluate in relation to file system-based storage, is available under the GPLv3 license via GitHub.

Keywords: UIMA, database interface, text annotation, Neo4J, MongoDB, Web Services, REST

1. Introduction

NLP and automatic text analysis necessarily involve the annotation of natural language texts. In various projects (e.g. (Da Silva et al., 2006; Kano et al., 2009; Hemati et al., 2016)), especially in an NLP context (e.g. (Savova et al., 2010; Eckart de Castilho and Gurevych, 2014; Patterson et al., 2017; Kreimeyer et al., 2017; Rudzewitz et al., 2017; Kassner et al., 2017)), the Apache *Unstructured Information Management applications* (UIMA) (Ferrucci et al., 2009) is used as a standard architecture for text annotation. Accordingly, there are various tools for processing and producing UIMA compliant documents (e.g. (Ogren et al., 2008; Zeldes et al., 2009; Ferrucci et al., 2010; Hemati et al., 2016; Niekler et al., 2017)). UIMA documents are specified by means of *UIMA Type System Descriptor* (UIMA TSD) files in order to enable interchangeability. The usage of UIMA in numerous projects and tools shows that this framework can be seen as a *de facto* standard in the context of NLP (Wilcock, 2017). However, despite its widespread usage, it is surprising that native database support is rarely found for UIMA documents. An exception is (Fette et al., 2013) who introduced an approach to storing UIMA documents by means of relational databases. In this article we want to close the gap between UIMA and databases by looking at several database systems beyond relational ones. In order to make UIMA-based projects (as, for example, TREEANNOTATOR (Helfrich et al., 2018)) conform to the requirements of modern text technology, database support is essential. This becomes clear when comparing database usage with file system-based storage and retrieval of UIMA documents. Generally speaking, approaches using file system-based storage lead to several bottlenecks:

(a) **No redundancy:** The document-based schema of the UIMA framework, which is usually stored via XMI[1] files, means that the results of UIMA processes cannot be stored decentrally (e.g., in a cloud). However, exclusively storing results in one file system without being based on a suitable backup concept can lead to data loss in the event of a system failure.

(b) **No query options:** To extract information from UIMA documents, these documents must first be completely imported into the cache. However, it is not possible to select only those UIMA documents that are required for a given application.

(c) **No shared use:** In projects including several participants, sharing of results is only possible by copying files. This in turn means that data security and tracking of file versioning must be ensured not by the system, but by the users.

(d) **No data security:** UIMA documents are not bound to data security and cannot be made available individually to third parties.

To avoid all these bottlenecks, we developed the so-called *UIMA Database Interface* (UIMA DI). It enables the generic use of UIMA documents in database systems in the context of NLP. In this paper, we explain the architecture of UIMA DI and evaluate it against file storage systems: in Section 2. we give a brief overview of application scenarios of UIMA DI. These scenarios will be referred to throughout the paper to exemplify the use of UIMA DI. In Section 3., we explain the architecture and the current implementation of UIMA DI. Its evaluation is documented in Section 4. and Section 5.. Finally, Section 6. and 7. summarize and give an outlook on future work.

2. Application cases

UIMA DI is currently used in three different application scenarios:

TEXTIMAGER TEXTIMAGER (Hemati et al., 2016) is an application for analyzing and visualizing textual data based on UIMA. Further, a tool for annotating hierarchical text relations (RST), called TREEANNOTATOR, is currently under development that utilizes the preprocessing

[1] http://www.omg.org/spec/XMI/

methods of TEXTIMAGER. To meet the needs of automatic text processing (TEXTIMAGER) and manual text annotation (TREEANNOTATOR) we decided for UIMA TSD as the underlying data model, while UIMA DI is used to manage the resulting UIMA documents.

STOLPERWEGE STOLPERWEGE (Mehler et al., 2017) is an application for modeling and visualizing the biographies of victims of the Holocaust. For this purpose, geo-data has to be managed in order to be used in related search queries. Once more, UIMA DI has the task to perform this data management.

EHUMANITIES DESKTOP The EHUMANITIES DESKTOP (Gleim et al., 2012) provides a range of applications and services for humanities scholars. This includes the so-called *ResourceManager* (see Figure 1): using UIMA DI, the ResourceManager makes UIMA documents independent resources that can be integrated into larger contexts (e.g., projects or repositories) and, therefore, avoids Bottleneck (c) of Section 1. This also includes the *AuthorityManager*, which enables the management of resource access rights at the level of individual users or groups of them, thus enabling privacy and data security. Thus, by being based on UIMA DI, the *AuthorityManager* avoids Bottleneck (d).

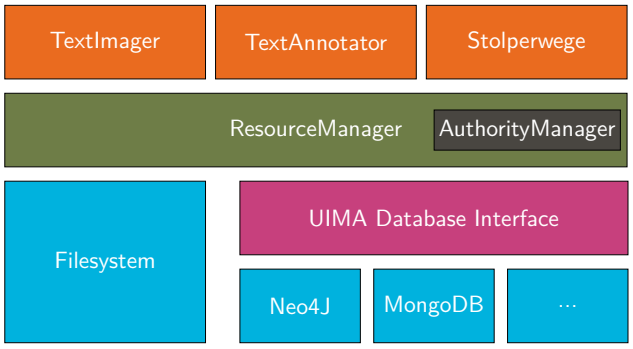


Figure 1: Application scenarios (orange) and software components (green) using UIMA DI.

3. UIMA Database Interface

UIMA TSD is an architecture that provides format and definition concepts for annotations stored in XMI files. Since UIMA TSD does not provide a built-in database solution, it is necessary to extend it. This is done by UIMA DI. To ensure uniqueness of documents managed by UIMA DI, it is necessary to reference exported XMI documents with their corresponding database records. For this reason, UIMA DI enables the generic storage and retrieval of UIMA documents by means of different databases.

3.1. Architecture

UIMA DI is an interface developed in Java and hosted at GitHub (GPLv3 license). In order to resolve dependencies, Apache Maven[2] is used. The core functions of UIMA DI address the following tasks:

(a) Analysis of UIMA TSD Analysis of the embedded UIMA TSD for generating meta information to enable the document to be stored in the used database which is implemented in UIMA DI. This is realized by analyzing the specified UIMA TSD. In addition, annotation templates can be created with the generated meta information, which can be used, for example, by TREEANNOTATOR and the project STOLPERWEGE (see Section 2.).

(b) (De-)Serialization of CAS UIMA documents are encoded in terms of CAS (UIMA COMMON ANALYSIS SYSTEM) (Götz and Suhre, 2004) representations. CAS files are serialized according to JSON[3] to be stored completely in the database. This enables lossless de-serialization of the entire UIMA documents.

(c) Dynamic ID insertion Each UIMA document and its content is mapped to a "Subject OF Analysis" (*sofa*) element. To prepare a UIMA document for database storage, an empty database entry is created and the resulting ID is stored as an additional sofa within the UIMA document. This sofa is identified during the import of previously exported UIMA documents: it enables a synchronization with the existing database entry. The generated database entry will then be updated with the previously generated meta information and the serialized CAS representation. Section 3.2. explains the corresponding data structures involved in the current implementation of UIMA DI regarding two databases: MongoDB and Neo4J. Figure 2 gives a visual depiction of the underlying process of document conversion.

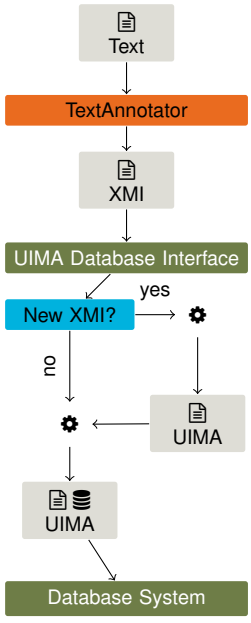


Figure 2: Process diagram of UIMA DI: A text is processed by TEXTANNOTATOR; then, the resulting XMI document x is sent to UIMA DI. If x is already stored in the database, the corresponding database entry is updated. If not, the XMI document is prepared for being processed by UIMA DI and stored in the underlying database system.

[2]https://maven.apache.org/

[3]http://www.json.org/

3.2. Current Implementations

The current version (1.0) of UIMA DI implements two database systems, namely MongoDB (3.2.1.) and Neo4J (3.2.2.). These databases utilize schema-free models and therefore allow for external database queries without presupposing knowledge about the database structure (as required, for example, by relational databases). Further databases can be easily adapted by the interface-based programming on which UIMA DI is based. Note also that our implementation includes a RESTful web service and analysis methods for the integrated UIMA TSD. The REST methods are implemented using Spark[4] while Swagger[5] is used for documenting the API of UIMA DI. Note that MongoDB and Neo4J allow for cloud-based storage enabling distributed data usage. Based on this analysis, we can state that our current implementation avoids Bottleneck (a) of Section 1. Furthermore, since the latter two databases provide highly specialized query languages, we also avoid Bottleneck (b).

3.2.1. MongoDB

We developed a UIMA Database Interface (UIMA DI) that allows for storing UIMA documents by means of the document-oriented database MongoDB[6]. When the CAS structure is serialized into JSON (see Section 3.1. above), three data fields are created as entries in MongoDB in the context of the applications enumerated in Section 2. (see Figure 3 for an example):

- **uima:** The serialized CAS representation of any input UIMA document is stored as such in MongoDB using the data field named *uima* (see Figure 3).
- **meta** is an array storing information about UIMA TSD-related data types instantiated by the input document. In this case, non-primitive data types are stored as references to objects.
- **geo** stores geospatial information (as needed, e.g., by the project STOLPERWEGE – see Section 2.). A geospatial index has been added for this purpose.

3.2.2. Neo4J

The graph database Neo4J[7] uses another storage structure than MongoDB (see Figure 3.2.2.): UIMA documents are now stored as nodes whose attributes denote simple data types and relations to other UIMA documents. The names of the attributes and relations are defined by the UIMA TSD associated with the document. The number of relationships to other objects is determined by the respective references in the UIMA document. As in the case of MongoDB, each node contains the complete serialization of the respective UIMA document (see Figure 4).
Note that we implemented UIMA DI by example of MongoDB and Neo4j to capture two paradigms of data modeling, that is, graph-oriented (Neo4j) and document-oriented modeling (MongoDB). In this way, UIMA DI already covers a wider range of databases. In the next section, we show how to evaluate this implementation.


```
{
   "_id" : ObjectId("5968
      ↪ e2fbb14e9e40e23b7e1e"),
   "geo" : {
      "coordinates" : [
         16.9327791,
         54.4322101
      ],
      "type" : "Point"
   },
   "meta" : {
      "firstName" : "Gustav",
      "lastName" : "Hoch",
      "deathDate" : "5968
         ↪ e2fbb14e9e40e23b7e1e",
      "id" : "5968e2fbb14e9e40e23b7e1e
         ↪ ",
      "type" : "org.hucompute.
         ↪ publichistory.datastore.
         ↪ typesystem.Person",
      "value" : "Gustav, Hoch",
      "birthDate" : "5968
         ↪ e2fbb14e9e40e23b7e1e"
   },
   "uima" : {
      "childNodes" : [
         {
            "_indexed" : 0,
            "_id" : 1,
            "sofaID" : "5968
               ↪ e2fbb14e9e40e23b7e1e
               ↪ ",
            "tagName" : "uima.cas.Sofa
               ↪ ",
            "sofaNum" : 2
         },
[...]
         {
            "_ref_deathDate" : 46,
            "_indexed" : 2,
            "firstName" : "Gustav",
            "lastName" : "Hoch",
            "end" : 0,
            "_id" : 8,
            "tagName" : "org.hucompute.
               ↪ publichistory.
               ↪ datastore.typesystem.
               ↪ Person",
            "_ref_sofa" : 1,
            "begin" : 0,
            "value" : "Gustav, Hoch",
            "_ref_birthDate" : 32
         }
      ]
   }
}
```


Figure 3: Excerpt from an entry of a UIMA document stored in MongoDB. The example is taken from the project STOLPERWEGE.

4. Experiments

Since there is no database solution for UIMA documents yet, our experiments are limited to measuring the reading, writing and querying time of UIMA documents in the respective database system and comparing it to the same op-

[4] http://sparkjava.com
[5] https://swagger.io/
[6] https://www.mongodb.com/
[7] https://neo4j.com/

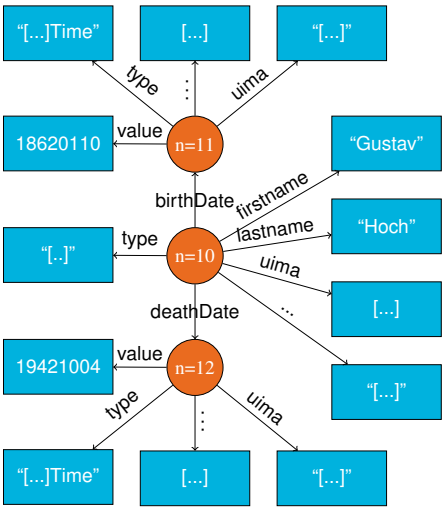


Figure 4: Schematic depiction of an entry of a UIMA document stored in Neo4J. The example is taken from the project STOLPERWEGE.

erations performed on the native file system. As test documents, we choose the German and English Wikipedia articles of six top candidates for the federal election in Germany 2017 (Angela Merkel, Martin Schulz, Katrin Göring-Eckardt, Sahra Wagenknecht and Alexander Gauland). The texts are preprocessed by means of TEXTIMAGER using the following taggers:

- Stanford NER Tagger[8]
- Stanford POS Tagger[9]
- Heideltime[10]

In this example, UIMA DI includes the following UIMA TSD as a result of using Maven as an interface to TEXTIMAGER data:

- Heideltime and
- DKPro-Core (contains the taggers listed above) (Eckart de Castilho and Gurevych, 2014).

After having preprocessed and stored the test documents using the systems to be evaluated (i.e., Neo4J, MongoDB and XMI files), we performed our evaluation. For this purpose, we selected all German and English texts separately to determine which storage system is the fastest to retrieve the corresponding UIMA documents. To this end, we did not optimize the (de-)serialization of the UIMA-CAS involved.

5. Results

Our results show that the read (see Figure 5) and write speed (see Figure 6) of both databases are slower than what is reached by the UIMA method of storing documents in XMI files. This is caused by the (de-)serialization of CAS elements into the JSON format. A second reason concerns the overhead induced by generating meta information. The more annotations a UIMA document contains, the more time is needed because different attributes and relations (based on the database) have to be created. However, for projects in which UIMA documents have to be shared, distributed document management is more important than reading and writing time. The same is true for projects which require sophisticated database query capabilities. Approaches being based on the file system require to import all XMI files to extract queried content. Obviously, this may induce a problematic memory load. In contrast to this, using UIMA DI as an interface to a database like Neo4j enables sophisticated content-related queries which can be faster than file-based approaches as shown in Figure 7.

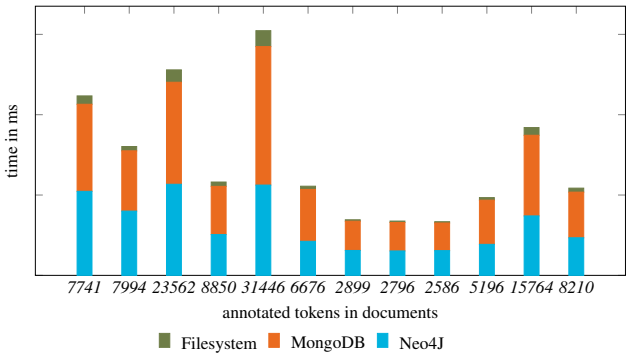


Figure 5: Measurement results with respect to reading.

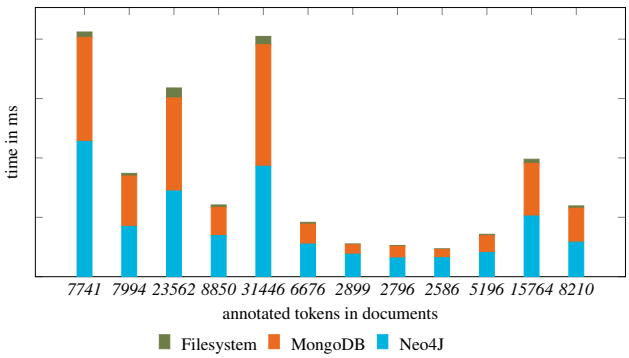


Figure 6: Measurement results with respect to writing.

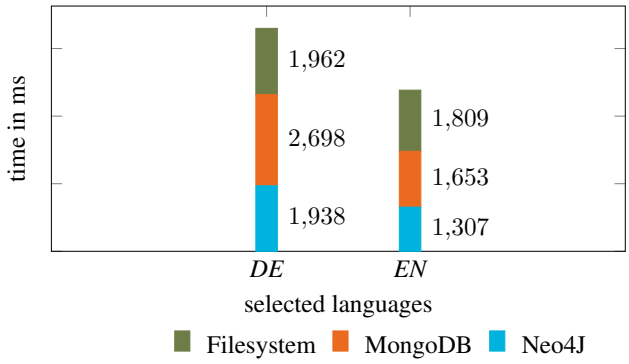


Figure 7: Measurement results with respect to querying.

[8]https://nlp.stanford.edu/software/CRF-NER.shtml
[9]https://nlp.stanford.edu/software/tagger.shtml
[10]https://github.com/HeidelTime/heideltime

6. Conclusion

We introduced UIMA DI as a generic interface to databases for managing UIMA documents. Our evaluation shows that our implementation of UIMA DI outperforms file-based systems when considering database queries. In projects related to digital humanities, querying UIMA documents in a sophisticated manner can be more important than optimizing reading and writing of UIMA documents as a whole. This is enabled by UIMA DI. Beyond that, several other bottlenecks of the file-based approach to processing UIMA documents are avoided by UIMA DI. Among other things, this concerns data security and protection. UIMA DI is available under license GPLv3 via GitHub[11] and is open to anyone interested in using or extending it.

7. Future Work

The future development of UIMA DI will primarily involve the implementation of further database systems like Elasticsearch[12] and Blazegraph[13]. With the help of Elasticsearch, we can avoid the document size limitation of MongoDB while ensuring a similar performance. Further, it is planned to accelerate the conversion from CAS to JSON by developing a converter specialized on this task.

8. Bibliographical References

Da Silva, P. P., McGuinness, D. L., and Fikes, R. (2006). A proof markup language for Semantic Web services. *Information Systems*, 31(4):381–395.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Ferrucci, D., Lally, A., Verspoor, K., and Nyberg, E. (2009). Unstructured Information Management Architecture (UIMA) Version 1.0. OASIS Standard, Mar.

Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., et al. (2010). Building Watson: An overview of the DeepQA project. *AI magazine*, 31(3):59–79.

Fette, G., Toepfer, M., and Puppe, F. (2013). Storing UIMA CASes in a relational database. *Unstructured Information Management Architecture (UIMA)*, page 10.

Gleim, R., Mehler, A., and Ernst, A. (2012). SOA implementation of the eHumanities Desktop. In *Proceedings of the Workshop on Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts, Digital Humanities 2012, Hamburg*.

Götz, T. and Suhre, O. (2004). Design and implementation of the UIMA Common Analysis System. *IBM Systems Journal*, 43(3):476–489.

Helfrich, P., Rieb, E., Abrami, G., Lücking, A., and Mehler, A. (2018). TreeAnnotator: Versatile Visual Annotation of Hierarchical Text Relations. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Hemati, W., Uslu, T., and Mehler, A. (2016). TextImager: a Distributed UIMA-based System for NLP. In *Proceedings of the COLING 2016 System Demonstrations*. Federated Conference on Computer Science and Information Systems.

Kano, Y., Baumgartner, J. W. A., McCrohon, L., Ananiadou, S., Cohen, K. B., Hunter, L., and Tsujii, J. (2009). U-Compare: share and compare text mining tools with UIMA. *Bioinformatics*, 25(15):1997–1998.

Kassner, L., Hirmer, P., Wieland, M., Steimle, F., Königsberger, J., and Mitschang, B. (2017). The Social Factory: Connecting People, Machines and Data in Manufacturing for Context-Aware Exception Escalation. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Kreimeyer, K., Foster, M., Pandey, A., Arya, N., Halford, G., Jones, S. F., Forshee, R., Walderhaug, M., and Botsis, T. (2017). Natural language processing systems for capturing and standardizing unstructured clinical information: A systematic review. *Journal of Biomedical Informatics*, 73(Supplement C):14–29.

Mehler, A., Abrami, G., Bruendel, S., Felder, L., Ostertag, T., and Spiekermann, C. (2017). Stolperwege: An App for a Digital Public History of the Holocaust. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, HT '17, pages 319–320, New York, NY, USA. ACM.

Niekler, A., Wiedemann, G., and Heyer, G. (2017). Leipzig Corpus Miner - A Text Mining Infrastructure for Qualitative Data Analysis. *CoRR*, abs/1707.03253.

Ogren, P. V., Wetzler, P. G., and Bethard, S. (2008). ClearTK: A UIMA Toolkit for Statistical Natural Language Processing. In *Towards Enhanced Interoperability for Large HLT Systems: UIMA for NLP. Proceedings of LREC 2008*.

Patterson, O. V., Freiberg, M. S., Skanderson, M., J. Fodeh, S., Brandt, C. A., and DuVall, S. L. (2017). Unlocking echocardiogram measurements for heart disease research through natural language processing. *BMC Cardiovascular Disorders*, 17(1):151, Jun.

Rudzewitz, B., Ziai, R., De Kuthy, K., and Meurers, D. (2017). Developing A Web-based Workbook for English Supporting the Interaction of Students and Teachers. In *Proceedings of the Joint 6th Workshop on NLP for Computer Assisted Language Learning and 2nd Workshop on NLP for Research on Language Acquisition at NoDaLiDa, Gothenburg*, number 134, pages 36–46. Linköping University Electronic Press.

Savova, G. K., Masanz, J. J., Ogren, P. V., Zheng, J., Sohn, S., Kipper-Schuler, K. C., and Chute, C. G. (2010). Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

Wilcock, G. (2017). The Evolution of Text Annotation Frameworks. In Nancy Ide et al., editors, *Handbook of*

[11] https://github.com/texttechnologylab/UIMADatabaseInterface
[12] https://www.elastic.co
[13] https://www.blazegraph.com/

Linguistic Annotation, pages 193–207. Springer Netherlands, Dordrecht.

Zeldes, A., Lüdeling, A., Ritz, J., and Chiarcos, C. (2009). ANNIS: a search tool for multi-layer annotated corpora. In *Proceedings of Corpus Linguistics*.

European Language Resource Coordination: Collecting Language Resources for Public Sector Multilingual Information Management

Andrea Lösch[1], Valérie Mapelli[2], Stelios Piperidis[3], Andrejs Vasiļjevs[4], Lilli Smal[1], Thierry Declerck[1], Eileen Schnur[1], Khalid Choukri[2] and Josef van Genabith[1]

[1]DFKI GmbH, [2]ELDA, [3]ILSP/Athena RC, [4]Tilde

[1]Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany

[2]9 rue des Cordelières, 75013 Paris, France

[3]Epidavrou & Artemidos, Maroussi, Athens, Greece

[4]Vienibas gatve 75a, LV1004, Riga, Latvia

[1]{andrea.loesch|lilli.smal|declerck|eileen.schnur|Josef.Van_Genabith}@dfki.de

[2]{mapelli|choukri}@elda.org, [3]spip@ilsp.gr, [4]andrejs@tilde.com

Abstract

In order to help improve the quality, coverage and performance of automated translation solutions for current and future Connecting Europe Facility (CEF) digital services, the European Language Resource Coordination (ELRC) consortium was set up through a service contract operating under the European Commission's CEF SMART 2014/1074 programme to initiate a number of actions to support the collection of Language Resources (LRs) within the public sector in EU member and CEF-affiliated countries. The first action focused on raising awareness in the public sector through the organisation of dedicated events: 2 international conferences and 29 country-specific workshops to engage national as well as regional/municipal governmental organisations, language competence centres, relevant European institutions and other potential holders of LRs from public service administrations and NGOs. In order to gather resources shared by the contributors, the ELRC-SHARE Repository was set up together with services supporting the sharing of LRs, such as the ELRC Helpdesk and Intellectual Property Rights (IPR) clearance support. All collected LRs pass a validation process developed by ELRC. The collected LRs cover all official EU languages, plus Icelandic and Norwegian.

Keywords: ELRC, Public Sector Information directive, LR evaluation, LR validation

1. Language Resources in and for the Public Sector

1.1 The European Public Sector Information Context

With the European Language Resource Coordination (ELRC), the European Commission (EC) has taken a decisive step towards supporting a truly multilingual Digital Single Market by enabling public services for Europe's citizens and businesses to operate freely across language barriers. The Connecting Europe Facility (CEF[1]) has various building blocks, one of them being "eTranslation", which is offering an automated translation platform[2] to facilitate multilingual communication and the exchange of documents and other linguistic content in Europe between national public administrations on the one hand and between these administrations and EU and CEF-affiliated country citizens and businesses (European Commission, 2017) on the other hand.

The CEF eTranslation platform provides machine translation (MT) services in all official languages of the EU as well as CEF-affiliated countries to address public administration scenarios in the areas of consumer rights, health, public procurement, social security, culture and others. eTranslation is technically managed by the EC Directorate-General for Translation (DGT) and will empower Europe's public online services such as the Online Dispute Resolution platform (ODR), eJustice, the Electronic Exchange of Social Security Information (EESSI), Business Registry Information System (BRIS), eProcurement and many others. Table 1 provides an overview of existing sector-specific Digital Service Infrastructures (DSIs) and their domains.

1.2 The PSI Directive

In accordance with the Public Sector Information (PSI) Directive 2003/98/EC (modified in 2013 by the Directive 2013/37/UE), Member States should ensure that documents, which are held by public sector bodies and accessible according to national access regimes, are re-usable for commercial or non-commercial purposes.

This obligation ensures that data produced by public administrations will be easier to share and made available for different types of usage.

CEF DSI	Domain
Online Dispute Resolution	Consumers' rights
Electronic Exchange of Social Security Information	Social security, insurance
eProcurement	Public procurement, contractual agreements
European e-Justice Portal	Justice, Law
eHealth	Health, Medicine
Business Registers Interconnection System	Business, market
Safer Internet	ICT
Cybersecurity	ICT
Public Open Data	Multiple domains
Europeana	Culture

Table 1: CEF Digital Service Infrastructures (DSIs) and their domains

[1] For more details on CEF, see: https://ec.europa.eu/digital-single-market/en/connecting-europe-facility.

[2] For more details on CEF eTranslation, see: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation.

Even though the PSI Directive is an important instrument to open up public sector data, there are many challenges in collecting LRs from public services, such as lack of awareness, lack of technical or legal competence, poor data management, etc. (Vasiļjevs et al., 2018).

1.3 Setting up a European Language Resource Coordination (ELRC)

European, national and regional public administrations deal with a huge amount of multilingual textual information in original and translated form. By sharing this linguistic data and turning it into language resources (LRs), they can improve the quality, coverage and performance of CEF eTranslation that needs multilingual LRs to train MT systems.

In April 2015, the ELRC Consortium was set up through EC's Connecting Europe Facility SMART 2014/1074 programme to initiate a number of actions with the aim to support the collection of such LRs. ELRC is coordinated by DFKI[3] (*Deutsches Forschungszentrum für Künstliche Intelligenz*, Germany), in partnership with ELDA[4] (Evaluations and Language Resources Distribution Agency, France), ILSP/Athena RC[5] (Institute for Language and Speech Processing/Athena Research Centre, Greece) and TILDE[6] (Latvia).

A Language Resource Board (LRB) was set up as governance and oversight body for the ELRC effort, consisting of National Anchor Points (NAPs), i.e. leading technological and public service representatives for each CEF country[7].

Through CEF eTranslation and the data collected by ELRC, European public services and public administrations across Europe will be one step closer towards operating without language barriers. By helping to improve eTranslation public service providers will gain access to better quality MT systems. These increase the efficiency of human translators and, in addition, the MT systems can be integrated in various online services.

This provides a major motivation for public services across Europe to donate and share data through ELRC: increased data provided for the content and languages relevant to particular administrations and/or services produces increased translation speed based on better accuracy of the machine translation output.

The more organisations contribute, the better their translated content will be. In addition, organisations benefit from the data donated by other public administrations, because they will also contribute content that may be relevant.

ELRC is responsible for the online coordination tool – ELRC website http://lr-coordination.eu – that integrates data sharing facilities, access to the LR catalogue, and information on ELRC events, Helpdesk and up-to-date information on the services provided by ELRC.

[3] http://dfki.de/en

[4] http://elda.org/en

[5] http:// www.ilsp.gr/en

[6] http://tilde.com

[7] http://lr-coordination.eu/anchor-points

2. Sharing Language Resources within ELRC

2.1 Raising Awareness in the Public Sector

In order to address the Public Sector and raise awareness of the need for data and sharing LRs to improve CEF eTranslation a large information dissemination initiative has been put into action by ELRC.

Two international ELRC conferences were organised: 1) as part of the Riga Summit 2015 on the Multilingual Digital Single Market[8] on 29 April 2015 and 2) as a satellite event of the Translating Europe Forum (TEF) in Brussels, on 26 October 2016. At each conference, more than 120 key stakeholders of the ELRC network (in particular potential data donors) participated.

In addition, in order to interact closely with national administrations, ELRC organised 29 country-specific workshops to meet with national and regional/municipal governmental organisations, language competence centres, relevant European institutions, other potential holders of LRs from the respective national public service administrations as well as important NGOs. Bringing ELRC to each country and getting engaged on the national level was essential to foster local ownership and local responsibility on which ELRC is built. Through the workshops, ELRC managed to identify more than 1.000 potential data sources. ELRC workshops established contacts to potential data holders, who are central to the subsequent data collection process.

The ELRC website provides detailed reports on these events, including presentations and many video recordings[9]. Vasiļjevs et al. (2018) details key discussions and findings at the workshops organised in Nordic and Baltic countries.

2.2 Collection Process

The process of sharing LRs with ELRC is straightforward and simple. Depending on the size of the data set, participating organisations may:

- Send the data to ELRC via email to data@lr-coordination.eu, as a zip file.
- Upload the data directly onto the ELRC-SHARE Repository, as a zip file, at https://www.elrc-share.eu (see Figure 1 below).
- Contact ELRC for other support such as setting up FTP services for specific deliveries.

ELRC-SHARE (Piperidis et al, 2018) covers the whole life cycle of LR sharing: uploading, documentation, uploading of accompanying documents, monitoring and reporting, updating, browsing, delivery and downloading.

The process is built on and inspired by META-SHARE (Piperidis, 2012) and is essentially an adaptation of its latest version v3.1.1, mainly in terms of the employed metadata schema, the identified user roles and the largely simplified operational workflow.

[8] http://www.rigasummit2015.eu/

[9] http://lr-coordination.eu/events



Figure 1: The ELRC-SHARE Repository

For all donations, ELRC needs to ensure that the material is in the public domain and follows the Public Sector Information (PSI) Directive transposition rules or that the necessary licenses have been obtained. Confidential and personal information is excluded from data to be shared.

2.3 Services to Facilitate the Sharing of LRs

In order to minimize efforts on the side of the donating institutions due to technical and legal obstacles, ELRC offers a range of support services for donating institutions. The ELRC Helpdesk[10] provides assistance and support to both technical and legal questions involved in the use, production, collection, processing and sharing of LRs.
The Helpdesk can be reached freely through several channels (including a telephone number associated to a web-conferencing desktop (Skype) and an email address). Moreover, a web forum platform[11] was set up so as to compile relevant questions and answers.
ELRC also supports donating organisations with different data processing services, including:

- Data conversion (e.g. to plain text or XML)
- Tag removal
- Re-formatting
- Data extraction
- Cleaning and alignment
- Meta-data validation
- Anonymization
- Etc.

If a public administration has potentially usable language data that needs assessment and processing, but that cannot be accessed outside the institution due to technical or legal reasons (e.g. data privacy, confidentiality), ELRC can provide on-site assistance. A member of the ELRC consortium with special knowledge of the particular language processing issue will travel to the organization on-site to provide technical/legal assistance required.
For public administrations, these services are provided for free in order to facilitate their data sharing efforts.

2.4 Data Processing

Each LR is analysed and processed by ELRC experts to ensure compliance with the *Language Resources Data Formats Specification* agreed with EC. According to this specification, resulting parallel data should be provided in the TMX format in UTF-8 encoding, without optional data fields (e.g. translator id, adjacent segments) and without non-printable control characters.
Monolingual corpora are to be delivered in plain text format without any additional annotation, in UTF-8 encoding, single file by language and resource, segmented into paragraphs. Terminology resources should be provided in the TBX format.
Several workflows including automated and manual tasks have been developed to ensure efficient processing of data and compliance to the required specifications. In a typical example, a public institution donates documents and translations in DOCX or PDF format. ELRC experts perform document level alignment of source and translated files, convert documents to plain text format, remove tags, perform sentence level alignment and also data cleaning by removing sentence pairs that include non-printable characters or text in language other than respective source or target language, and convert this data to TMX format. TMX files are validated using TMXValidator[12]. Resulting LRs are stored in the ELRC-SHARE repository and the corresponding metadata fields are filled.
Among the tools used in the data processing are DictMetric (Su & Babych, 2012) for document alignment, Microsoft Bilingual Sentence Aligner[13], language detection tool PYCLD2[14], and many others.

2.5 IPR Clearance

For a number of LRs, ELRC interacted with providers to clarify legal issues linked to IPR and licensing constraints. Licenses were drafted to address e.g. country-specific rules in relation to the PSI Directive, limitations in use within the CEF eTranslation platform, etc.
The ELRC-Repository legal part of the metadata was adapted in order to include open data licenses (such as Open Licence/Licence Ouverte, for France). LRs provided can be classified and viewed depending on the licence available: public domain, open under PSI, open licenses, standard licenses, and non-standard licenses.

3. Validation of Language Resources within ELRC

3.1 Validation Guidelines Implementation

Validation can be understood as the quality control of a LR against a list of relevant criteria (Schneller et al., 2017). Due to the high number of LRs required within the project, the ELRC consortium decided to complement the donated LRs with additional LRs produced from scratch through a website crawling process (Papavassiliou et al, 2018). Web crawling was conducted using ILSP-FC, a comprehensive end-to-end solution for the acquisition of domain-specific monolingual and bilingual corpora from the web.

[10] http://lr-coordination.eu/helpdesk
[11] http://helpdesk.lr-coordination.eu/overview
[12] https://www.maxprograms.com/products/tmxvalidator.html
[13] https://www.microsoft.com/en-us/download/details.aspx?id=52608
[14] https://github.com/aboSamoor/pycld2

The ELRC partners (the National Anchor Points) initially identified and documented public administration websites (e.g. websites of ministries, local authorities, museums, etc.) as candidate sources for the extraction of content relevant to the CEF Digital Service Infrastructures (DSIs) and subsequently deployed ILSP-FC to acquire language resources for specific (EN-X) language pairs, where X stands for official EU languages in CEF-affiliated countries.

All gathered LRs needed to be submitted to the ELRC validation process to check their conformity with the original requirements of the project. The goal of the ELRC validation procedures is to provide a methodology, known as "Validation Guidelines" to validate donated and crawled data, which was collected within the project. The validation of donated and crawled data had to be conducted in different ways.

- It was assumed that the donated data consist of high quality data in terms of content (in particular translations for multilingual data, data produced by human experts), but require a technical- and legal-oriented evaluation. Here validation consists of:
    - checking compliance of data with the ELRC objectives and scope,
    - checking the format of provided data, and
    - checking whether the legal information provided is compliant with the ELRC scope.
- As crawled data come from automatic processing, their validation necessitates deeper content validation, while crawling already delivers the format that corresponds to the Language Resource production requirements. Legal validation of crawled data is also required and had to be carried out according to the following steps:
    - checking whether crawled websites are under the scope of the Public Sector Information (PSI) Directive to make sure that the content can be re-used, and
    - estimating the quality of translation with a team of language expert validators.

All types of data were uploaded to the ELRC-Share Repository and corresponding metadata was validated and completed. Finally, a Validation Report was provided for each data set, and all available legal related documentation were asserting the quality of all data.

Full Validation Guidelines, including the Validation Report template are made available online: http://www.lr-coordination.eu/sites/default/files/common/ELRC Data Validation Guidelines.pdf.

3.2 Issues Regarding the Validation of Language Resources

As indicated above, the validation of donated data was limited to a "Quick Quality Check" that includes:

- compliance with the ELRC scope (relevant language, not data from or already available to EC), and
- checking the metadata elements against a number of minimal requirements in terms of technical information (i.e. format, content, alignment), as well as legal information (whether required information on licensing has been well filled in, e.g. PSI-compliant data, available licenses, attribution, etc.

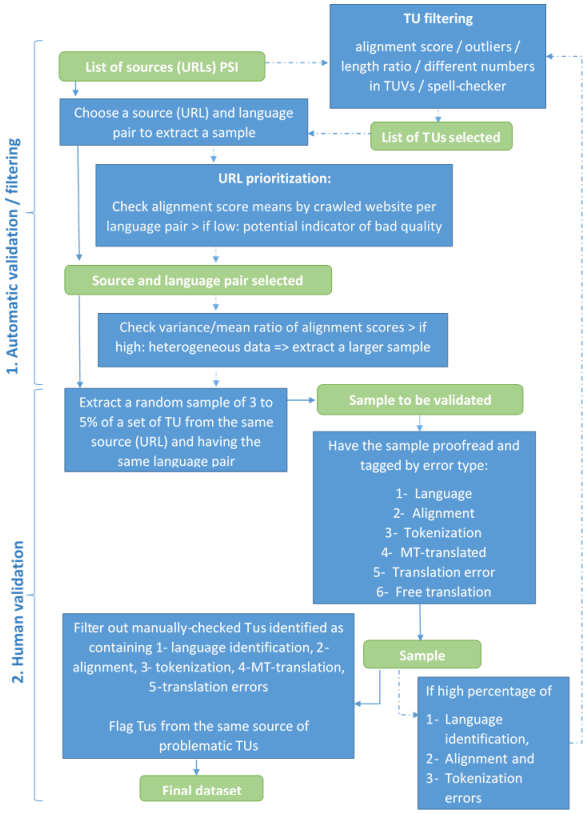


Figure 2: Workflow for Content Validation of Crawled Language Resources

For crawled data, the validation process was much more comprehensive, e.g. the list of crawled URLs was manually checked to assess if the websites are under the PSI scope. Content from websites that did not fall under the PSI Directive was excluded. Errors in Translations Units (TU) were reported and TUs marked up as containing errors were automatically removed; the remaining TUs were annotated with an indication on the probability of finding the same errors. The validation of the content consisted of both an automatic and a manual procedure. Figure 2 illustrates the workflow employed within ELRC for the validation of crawled data.

4. Impact and Results

Having received the first resources starting in Spring 2016, ELRC has managed to collect 225 LRs within one year, covering all official EU languages, plus Icelandic and both varieties of Norwegian, Bokmål and Nynorsk. This includes bi- or multi-lingual contents in digital editable formats ranging from reports, publications and other materials for internal and external use, web contents and brochures, but also terminologies and glossaries.
In 2017, ELRC processed 100 of these 225 LRs to make them compliant with the ELRC technical specification and be fully ready to be used in MT training. 82 new additional LRs were collected in 2017.

More than 58 public sector organisations across Europe have shared their language data with ELRC, including in particular national ministries, governmental bodies and public services.

5. Conclusion and future Work

This paper provides an overview of the major challenges faced by ELRC and how they have been addressed. We are in the process of identifying clear recommendations for future actions regarding Language Resources sharing and collection, in particular with regard to:

- The continuation of personal support structures provided by ELRC, complementary to the ELRC Helpdesk
- The organization of future conferences
- The organization of workshops (including in particular the further development of stakeholder involvement and data pipeline sustainability)
- The support of future work of the Language Resource Board, with for example additional country-specific reach-out events

6. Acknowledgements

The European Language Resource Coordination (ELRC) is a service contract operating under the EC's Connecting Europe Facility SMART 2014/1074 programme from April 2015 to April 2017 and will continue until December 2019 under SMART 2015/1091 LOT 2 "Language Resource coordination and collection with related legal and technical work" and SMART 2015/1091 LOT 3 "Acquisition of additional Language Resources and related refinement/processing services and their provision of the Language Resource Repository of CEF Automated Translation Platform".

7. Bibliographical References

European Commission (2017). eTranslation – Making European Digital Public Services Multilingual. Available at: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation [last accessed: 22.02.2018]

Schneller, Priscille ; Fernandez-Barrera, Meritxell ; Mapelli, Valérie ; Popescu, Vladimir, Choukri, Khalid ; Prokopidis, Prokopis ; Papavassiliou, Vassilis (2017). European Language Resource Coordination – Validation Guidelines. Available at http://www.lr-coordination.eu/sites/default/files/common/ELRC Data Validation Guidelines.pdf [last accessed: 22.02.2018]

CEF: https://ec.europa.eu/digital-single-market/en/connecting-europe-facility [last accessed: 22.02.2018]

CEF eTranslation: https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation

Directive 2013/37/UE: http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32013L0037 [last accessed: 22.09.2017]

ELRC-SHARE: http://www.lr-coordination.eu/resources [last accessed: 22.02.2018]

Piperidis, S.; Labropoulou, P.; Deligiannis, M.; Giagkou, M. (2018). Managing Public Sector Data for Multilingual Applications Development, In Proceedings of the 11th Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

Papavassiliou, V.; Prokopidis, P.; Piperidis, S. (2018) Discovering parallel language resources for training MT engines. In Proceedings of the 11th Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

Piperidis, S. (2012). The META-SHARE Language Resources Sharing Infrastructure: Principles, Challenges, Solutions. In Proceedings of the Eighth International Language Resources and Evaluation (LREC 2012), Istanbul, Turkey. European Language Resources Association (ELRA).

Su, F.; Babych, B. (2012). Measuring comparability of documents in non-parallel corpora for efficient extraction of (semi-) parallel translation equivalents. In Proceedings of the Joint Workshop on Exploiting Synergies between Information Retrieval and Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra) (pp. 10-19). Association for Computational Linguistics.

Vasiļjevs, A.; Rozis, R.; Kalniņš, R.; Bērziņš, A. (2018). Collecting Language Resources from Public Administrations in the Nordic and Baltic Countries. In Proceedings of the 11th Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

Tilde MT Platform for Developing Client Specific MT Solutions

Mārcis Pinnis, Andrejs Vasiļjevs, Rihards Kalniņš, Roberts Rozis, Raivis Skadiņš, Valters Šics
Tilde
Vienības gatve 75A, Riga, Latvia, LV-1004
{marcis.pinnis, rihards.kalnins, roberts.rozis, raivis.skadins, valters.sics, andrejs}@tilde.lv

Abstract

In this paper, we present Tilde MT, a custom machine translation (MT) platform that provides linguistic data storage (parallel, monolingual corpora, multilingual term collections), data cleaning and normalisation, statistical and neural machine translation system training and hosting functionality, as well as wide integration capabilities (a machine user API and popular computer-assisted translation tool plugins). We provide details for the most important features of the platform, as well as elaborate typical MT system training workflows for client-specific MT solution development.

Keywords: machine translation, cloud-based platform, data processing workflows

1. Introduction

Today, as globalisation and cross-border trade infuse more sectors of the economy, the need for translations and multilingual content is continuously increasing. The demand for translations is surpassing the supply that professional translation services can handle. As Common Sense Advisory reported in a 2016 market study, "enterprises intend to increase their translation volumes by 67% over current levels by 2020." (Lommel and DePalma, 2016) An obvious alternative to customers who cannot afford professional translation services or who have too much content to translate, and also an obvious choice for translation and localisation service providers who see the potential in increasing their productivity, is machine translation (MT). General MT providers (such as Google[1] and Microsoft[2]) cater for the masses with general-domain MT systems. However, customers who require purpose-built and highly customised MT systems, particular for complex or low-resourced languages, turn to MT service providers that offer MT system development and customization capabilities as a full-service package.

One such custom MT platform is Tilde MT[3], the successor of the LetsMT! platform (Vasiļjevs et al., 2012) for Statistical Machine Translation (SMT) system development, first launched in 2012. Tilde MT builds upon the LetsMT! platform by providing greater customisation capabilities, smarter data processing workflows, and current state-of-the-art Neural Machine Translation (NMT) system support.

The paper is further structured as follows: Section 2. provides an overview of the main features of the Tilde MT platform, Section 3. describes the MT system training workflow with the different tools used for data processing, Section 4. describes the MT system translation workflow, and Section 5. concludes the paper.

2. Overview of Tilde MT

Tilde MT is a cloud-based custom MT platform that allows users to store linguistic resources (such as parallel and monolingual corpora and multilingual term collections); train SMT and NMT systems; integrate the systems through computer assisted translation (CAT) tool plugins or the Tilde MT external API in users' translation and multilingual content creation workflows; and perform translation of text snippets, documents of various popular formats, and websites directly in the Tilde MT graphical user interface or using a widget that can be integrated in any website.

2.1. SMT and NMT System Support

Tilde MT supports two MT paradigms: statistical machine translation and neural machine translation. The platform allows to train Moses phrase-based SMT systems (Koehn et al., 2007) and attention-based encoder-decoder NMT systems with multiplicative long short-term memory units using the Nematus toolkit (Sennrich et al., 2017). The platform has been designed to allow switching to different NMT engines easily. For translation, Nematus NMT models are converted to Marian (formerly AmuNMT) NMT models (Junczys-Dowmunt et al., 2016) that allow reaching much higher translation speed (up to 10 times faster compared to Nematus in non-batched translation scenarios).

2.2. Cloud-based Infrastructure

To facilitate on-demand training and deployment of MT systems, it is important for the platform to be highly scalable, available, and reliable. To address these requirements, Tilde MT has been developed as a distributed cloud-based platform (see Figure 1) that is able to dynamically start and turn off computing nodes depending on current workloads. The computing nodes are responsible for running MT system training tasks and translation servers. To provide MT services also to customers with security concerns or customers whose data is not allowed to leave the customers' infrastructure, Tilde MT can be also deployed as an enterprise solution in customer infrastructure.

2.3. Tilde Data Library

The platform features a resource-rich data repository, the Tilde Data Library. The library is used as the central data repository for Tilde MT, as well as an open facility for registered users to upload their own corpora (both parallel and

[1] https://translate.google.com
[2] https://www.bing.com/translator
[3] https://tilde.com/mt

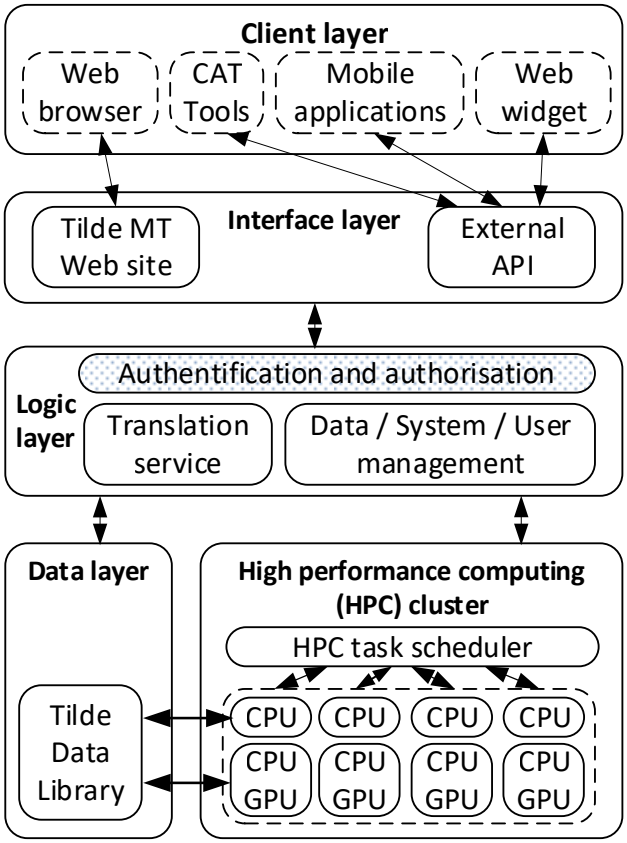


Figure 1: Tilde MT infrastructure design

monolingual) for training MT engines. It stores publicly available and proprietary parallel and monolingual corpora as well as multilingual term collections that users can use to train their MT systems within the Tilde MT platform. Several of the largest (publicly available) parallel corpora in the Tilde Data Library are the DGT-TM (Steinberger et al., 2012), Tilde MODEL (Rozis and Skadiņš, 2017), Open Subtitles (Tiedemann, 2009), MultiUN (Chen and Eisele, 2012), DCEP (Hajlaoui et al., 2014), JRC-Acquis (Steinberger et al., 2006), Europarl (Koehn, 2005), Microsoft Translation Memories and UI Strings Glossaries (Microsoft, 2015). The Tilde Data Library stores approximately 12.35 billion parallel segments for 58 languages and over 4 million terms for more than 125 languages.

2.4. Tilde Terminology Integration

Term collections are important linguistic resources that are often used by translators who work on domain-specific translation tasks. To alleviate the need for users to store and manage their terminological resources on multiple platforms, the Tilde MT platform allows users to access their Tilde Terminology (Pinnis et al., 2013) resources. Users can add their term collections to MT system training tasks for static terminology integration as well as to running SMT[4] systems for dynamic terminology integration (Pinnis, 2015). Integration of terminology has been shown to improve term translation accuracy by up to 52.6% (Pinnis, 2015).

[4]For NMT systems, dynamic integration is not yet available.

2.5. CAT Tool Plugins and External API

Professional translators often use specific computer-assisted translation tools in their professional duties. Depending on specific projects or customers, translators may have to use different CAT tools. Therefore, it is crucial to the success of an MT platform to provide integration capabilities for at least the most popular CAT tools. MT systems trained on Tilde MT can be accessed from at least four popular CAT tools[5]: MateCat (Federico et al., 2014), SDL Trados Studio, Memsource[6], and memoQ[7].

3. MT System Training Workflow

Tilde MT provides users with rich customisation possibilities when training MT systems. Users can specify which filtering and cleaning steps to take, which data pre-processing tools to use, and which MT models and with which configurations to train. Further, we describe in more detail the main MT system training capabilities of Tilde MT.

3.1. Data Filtering and Cleaning

Not all data that users upload in the Tilde Data Library as parallel data is actually parallel. For instance, the data may contain misalignment issues, formatting issues, encoding corruption issues, sentence breaking issues, etc. Therefore, the Tilde MT platform performs data filtering before MT system training. The following issues are addressed by various filters in Tilde MT:

1. Source-source or target-target entries in parallel data (equal source/target entries are filtered out).
2. Sentence splitting issues (segments $>$1000 symbols or $>$400 tokens are filtered out; the numerical thresholds here and further can be adjusted for each individual training task).
3. Data corruption through optical character recognition (OCR), e.g., when processing PDF documents (segments containing tokens with $>$50 symbols are filtered out).
4. Redundancy issues (duplicate entries are filtered out).
5. Partial translation (also sentence splitting) issues (entries where the length ratio between the source and target segments is too small (e.g., $<$0.3) are filtered out).
6. Foreign language data issues (entries containing letters from neither source nor target languages are filtered out).
7. Sentence misalignment issues (sentences failing a cross-lingual alignment test using c-eval (Zariņa et al., 2015) are filtered out).

In our previous research, we identified that NMT systems are sensitive to systematic noise that can be found in the parallel data (Pinnis et al., 2017a). Therefore, additionally

[5]For more details, refer to: https://www.tilde.com/products-and-services/machine-translation/features/integration-with-cat

[6]https://memsource.com

[7]https://www.memoq.com

to the filters used for SMT systems, for NMT systems Tilde MT performs also the following filtering steps to ensure that the parallel corpora are filtered more strictly (Pinnis et al., 2017c):

1. Incorrect language filtering using a language detection tool (Shuyo, 2010).
2. Low content overlap filtering using the cross-lingual alignment tool *MPAligner* (Pinnis, 2013).
3. Digit mismatch filtering, which showed to be effective in identifying parallel corpora sentence segmentation issues.

After filtering, each valid segment is cleaned in order to further reduce noise and to remove potential non-translatable text fragments, which will be processed by the formatting tag handling method in the translation workflow prior to decoding but are not necessary during training. The following are the main cleaning steps:

1. Removal of HTML and XML tags,
2. Removal of the byte order mark
3. Removal of escaped characters (e.g., "\n")
4. Decoding of XML entities, normalisation of whitespace characters
5. Removal of empty braces and curly tags (specific to parallel corpora extracted from some CAT tools)
6. Separation of ligatures into letters (specific to parallel corpora extracted using OCR methods)

3.2. Data Pre-processing

Next, the filtered and cleaned corpora are pre-processed using standard and custom tools. This step is identical for both the training and translation workflows. The following pre-processing steps are performed:

1. *Normalisation of punctuation*. Tilde MT allows limiting MT models to one standard of quotation marks and apostrophes.
2. *Identification of terminology*. For SMT systems, dynamic terminology integration support ensures that terms can be identified in the source text and possible translation equivalents can be provided to the SMT engine before the actual translation.
3. *Identification of non-translatable entities*. E-mail addresses, URLs, file addresses and XML tags can be identified and replaced with place-holders.
4. *Tokenisation*. Tilde MT uses a regular expression-based tokeniser that allows applying customised tokenisation rules for each language and customer.
5. *Truecasing*. The standard Moses truecasing tool *truecase.perl* can be used to truecase the first or all words of each sentence.
6. *Morphology-driven word splitting* (MWS) (Pinnis et al., 2017b) or *byte-pair encoding* (BPE) (Sennrich et al., 2015). For NMT systems, tokens can be split using a morphological analyser[8] and processed with BPE.
7. *Source side factorisation*. Tilde MT supports NMT models that use linguistic input features (Sennrich and Haddow, 2016). Therefore, the source side can be factored using language-specific factorisation tools (depending on the source language - either part-of-speech or morphological taggers or syntactic parsers).

3.3. SMT and NMT Model Training

After the data is pre-processed, SMT or NMT models are trained. During configuration of an MT system, users can freely select whether to train an SMT or an NMT model.
For SMT models, word alignment is performed using fast-align (Dyer et al., 2013), after which a 7-gram translation and the *wbe-msd-bidirectional-fe-allff*[9] reordering models are built. For language modelling, the KenLM (Heafield, 2011) toolkit is used (the n-gram order can be specified by the users). SMT systems are tuned using MERT (Bertoldi et al., 2009).
For NMT models, training data is further pre-processed by introducing unknown phenomena (i.e., unknown word tokens) within training data following the methodology by Pinnis et al. (2017b). Then, an NMT model is trained using the configuration specified by the user (e.g., the vocabulary size, embedding and hidden layer dimensions, whether to use dropout, the learning rate, gradient clipping, etc. parameters can be freely configured). To ensure the stability of the system, a maximum value restriction is applied for each configuration parameter that influences the hardware resources to be consumed.

4. Translation Workflow

When an MT model is trained, a translation server can be started. A typical translation server (see Figure 2 for a broad overview) allows to translate text snippets, translation segments (i.e., content that includes tags), documents (e.g., various OpenDocument[10] and Office Open XML[11] formats, translation and localisation formats, such as XML Localisation Interchange File Format (XLIFF)[12] (with different variations), Translation Memory eXchange (TMX)[13], etc.), and web sites (the latter two basically consist of zero to many translation segments).
When translating translation segments, first, formatting tags are removed from the segments and the tag positions are remembered for reinsertion after translation. Then, the text is pre-processed using the same steps that were used for the training data. Additionally to the previous pre-processing steps, the text is also split into sentences (before

[8] Currently, for Latvian and English only.
[9] For more information, see http://www.statmt.org/moses/?n=FactoredTraining.BuildReorderingModel
[10] http://opendocumentformat.org/
[11] See ISO/IEC 29500 at http://standards.iso.org/ittf/PubliclyAvailableStandards
[12] http://docs.oasis-open.org/xliff
[13] http://www.ttt.org/oscarstandards/tmx/tmx13.htm

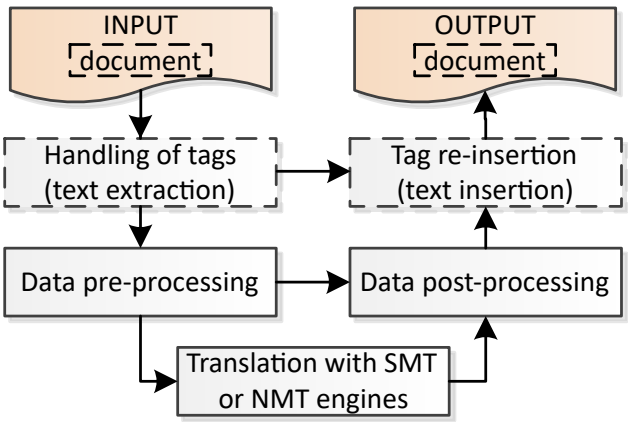


Figure 2: A broad overview of the translation workflow

MWS). For NMT systems, rare and unknown words are identified (based on word part unigram and bigram statistics in the training data) and replaced with unknown word tokens in order to assist the NMT model in the handling of rare and unknown phenomena (Pinnis et al., 2017c). Each pre-processed sentence is then translated using the MT engine that was used during training.

After translation, unknown word tokens are replaced back with their source language tokens, the text is recased and detokenised using language-specific detokenisation rules. Tilde MT provides also functionality for rule-based localisation, i.e., the transformation of certain types of tokens according to customer specified localisation rules. For instance, a customer can pre-set the desired styles of quotation marks and apostrophes, number formats (i.e., decimal point and thousand separators), even conversion rules for different units of measurement (e.g., imperial to metric units, etc.). Finally, if the source was a translation segment, formatting tags are re-inserted in the translated text and (in case of document and Web page translation) the translated segment is inserted in the final document.

In order for the whole translation workflow to work, it is important to keep track of word and phrase alignments and the changes of the word and phrase alignments at each step. When using SMT models, the word alignments are provided by the phrase-based translation model, and when using NMT models, the word alignments are extracted (Pinnis et al., 2017c) from the alignment matrices produced by the attention mechanism of the NMT model.

To address scalability requirements, Tilde MT allows starting multiple translation server instances of each MT system. To save computing resources, translation servers can be set to *fall asleep* after a certain time without any translation requests.

5. Conclusion

In this paper, we presented Tilde MT, a distributed cloud-based custom machine translation platform that is capable of supporting SMT and NMT systems. Tilde MT with its feature-rich MT system training workflow alleviates a lot of manual work necessary for data preparation prior to training. The workflows have been specifically adjusted to cater for NMT system, which are more sensitive to systematic noise, development. The platform also allows training more robust NMT models by preparing data in a way that it contains unknown phenomena in common contexts. We also described the translation workflow and its abilities to handle unknown phenomena and to facilitate customer specific customisation needs. Finally, we briefly discussed also the various integration possibilities offered by Tilde MT, such as the CAT tool plugins, the external API, and the Tilde Terminology integration).

6. Acknowledgements

In accordance with the contract No. 1.2.1.1/16/A/009 between the "Forest Sector Competence Centre" Ltd. and the Central Finance and Contracting Agency, concluded on 13th of October, 2016, the study is conducted by Tilde Ltd. with support from the European Regional Development Fund (ERDF) within the framework of the project "Forest Sector Competence Centre".

7. Bibliographical References

Bertoldi, N., Haddow, B., and Fouet, J.-B. (2009). Improved Minimum Error Rate Training in Moses. *The Prague Bulletin of Mathematical Linguistics*, 91(1):7–16.

Chen, Y. and Eisele, A. (2012). MultiUN v2: UN Documents with Multilingual Alignments. In *LREC*, pages 2500–2504.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013)*, number June, pages 644–648, Atlanta, USA.

Federico, M., Bertoldi, N., Cettolo, M., Negri, M., Turchi, M., Trombetti, M., Cattelan, A., Farina, A., Lupinetti, D., Martines, A., et al. (2014). The MateCat Tool. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 129–132.

Hajlaoui, N., Kolovratnik, D., Väyrynen, J., Steinberger, R., and Varga, D. (2014). DCEP-Digital Corpus of the European Parliament. In *LREC*, pages 3164–3171.

Heafield, K. (2011). KenLM : Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, number 2009, pages 187–197. Association for Computational Linguistics.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is Neural Machine Translation Ready for Deployment? A Case Study on 30 Translation Directions. *arXiv preprint arXiv:1610.01108*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koehn, P. (2005). Europarl: a Parallel Corpus for Statistical Machine Translation. In *MT summit*, volume 5, pages 79–86.

Lommel, A. R. and DePalma, D. A. (2016). Europe's Leading Role in Machine Translation. Technical report, Common Sense Advisory.

Microsoft. (2015). Translation and UI Strings Glossaries.

Pinnis, M., Gornostay, T., Skadiņš, R., and Vasiļjevs, A. (2013). Online Platform for Extracting, Managing, and Utilising Multilingual Terminology. In *Proceedings of the Third Biennial Conference on Electronic Lexicography, eLex 2013*, pages 122–131, Tallinn, Estonia. Trojina, Institute for Applied Slovene Studies (Ljubljana, Slovenia) / Eesti Keele Instituut (Tallinn, Estonia).

Pinnis, M., Krišlauks, R., Deksne, D., and Miks, T. (2017a). Evaluation of Neural Machine Translation for Highly Inflected and Small Languages. In *Proceedings of the 18th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2017)*, Budapest, Hungary.

Pinnis, M., Krišlauks, R., Deksne, D., and Miks, T. (2017b). Neural Machine Translation for Morphologically Rich Languages with Improved Sub-word Units and Synthetic Data. In *Proceedings of the 20th International Conference of Text, Speech and Dialogue (TSD2017)*, Prague, Czechia.

Pinnis, M., Krišlauks, R., Miks, T., Deksne, D., and Šics, V. (2017c). Tilde's Machine Translation Systems for WMT 2017. In *Proceedings of the Second Conference on Machine Translation (WMT 2017), Volume 2: Shared Task Papers*, pages 374–381, Copenhagen, Denmark. Association for Computational Linguistics.

Pinnis, M. (2013). Context Independent Term Mapper for European Languages. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2013)*, pages 562–570, Hissar, Bulgaria.

Pinnis, M. (2015). Dynamic Terminology Integration Methods in Statistical Machine Translation. In *Proceedings of the Eighteenth Annual Conference of the European Association for Machine Translation (EAMT 2015)*, pages 89–96, Antalya, Turkey. European Association for Machine Translation.

Rozis, R. and Skadiņš, R. (2017). Tilde MODEL-Multilingual Open Data for EU Languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265.

Sennrich, R. and Haddow, B. (2016). Linguistic Input Features Improve Neural Machine Translation. In *Proceedings of the First Conference on Machine Translation (WMT 2016) - Volume 1: Research Papers*, pages 83–91.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2015)*, Berlin, Germany. Association for Computational Linguistics.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Barone, A. V. M., Mokry, J., et al. (2017). Nematus: a Toolkit for Neural Machine Translation. *arXiv preprint arXiv:1703.04357*.

Shuyo, N. (2010). Language Detection Library for Java.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufis, D., and Varga, D. (2006). The JRC-Acquis: a Multilingual Aligned Parallel Corpus with 20+ Languages. *arXiv preprint cs/0609058*.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlter, P. (2012). DGT-TM: a Freely Available Translation Memory in 22 Languages. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 454–459.

Tiedemann, J. (2009). News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Recent advances in natural language processing*, volume 5, pages 237–248.

Vasiļjevs, A., Skadiņš, R., and Tiedemann, J. (2012). LetsMT!: A Cloud-Based Platform for Do-It-Yourself Machine Translation. In Min Zhang, editor, *Proceedings of the ACL 2012 System Demonstrations*, number July, pages 43–48, Jeju Island, Korea. Association for Computational Linguistics.

Zariņa, I., Ņikiforovs, P., and Skadiņš, R. (2015). Word alignment based parallel corpora evaluation and cleaning using machine learning techniques. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*.

Improving homograph disambiguation with supervised machine learning

Kyle Gorman, Gleb Mazovetskiy, Vitaly Nikolaev
Google Inc.
{kbg,glebm,vitalyn}@google.com

Abstract

We describe a pre-existing rule-based homograph disambiguation system used for text-to-speech synthesis at Google, and compare it to a novel system which performs disambiguation using classifiers trained on a small amount of labeled data. An evaluation of these systems, using a new, freely available English data set, finds that hybrid systems (making use of both rules and machine learning) are significantly more accurate than either hand-written rules or machine learning alone. The evaluation also finds minimal performance degradation when the hybrid system is configured to run on limited-resource mobile devices rather than on production servers. The two best systems described here are used for homograph disambiguation on all US English text-to-speech traffic at Google.

Keywords: Homograph disambiguation, machine learning, text normalization, text-to-speech synthesis

1. Introduction

Despite substantial progress in applying machine learning to text normalization and linguistic analysis for text-to-speech synthesis (TTS), it is still the case that most front-end processing in real-world TTS systems is done by language-specific hand-written rules. While rule systems provide an high degree of interpretability, they may require a great deal of development effort to obtain reasonable accuracy. Therefore, rule-based components represent substantial barriers for both quality control and internationalization.

1.1. Homograph disambiguation

When text input is sent to the Google TTS engine, it is first tokenized, and then pronunciations are selected for these tokens (Ebden and Sproat, 2014). For most tokens—such as in-vocabulary words with a single pronunciation—this requires only dictionary lookup. But other types of tokens, so-called *semiotic classes* (Taylor, 2009)—such as numbers, currencies and measures, dates and times, etc.—and out-of-vocabulary words, require additional language-specific processing (Sproat et al., 1992; Sproat et al., 2001). One particularly challenging class of tokens are *homographs*, polysemous words pronounced differently depending on the intended sense. One must analyze the context in which a homograph occurs to select a contextually appropriate pronunciation.

1.2. Sources of homography

Homography occurs any time two words pronounced distinctly are spelled the same. This may arise due to inflectional processes not indicated orthographically, as in the English irregular verb *read*, pronounced as either the past [ɹɛd] or the present [ɹiːd]. Alternatively, homographs may represent phonologically and semantically distinct lexical items which just happen to share a spelling, as in *bow*, pronounced either as [boʊ] or [baʊ]. We refer to these categories of homography as *morphosyntactic* and *lexical*, respectively.[1]

One may also discern a *mixed* category for those homographs which carry both morphosyntactic and semantic distinctions. For instance, *produce* may either be [pɹəˈdus], a verb referring to a process, or the noun [ˈpɹoʊdus], referring to the result of that process. At the same time, the verb has a number of semantic extensions not available for the noun; for instance, it may refer to overseeing the creation of musical recordings. Because the semantic relationship between the various pronunciations of a homograph are arbitrary, and because there may be hundreds or even thousands of homographs in a given language, homograph disambiguation requires substantial language-specific resources. Furthermore, it has been claimed that listeners' subjective evaluations of TTS are particularly sensitive to homograph disambiguation errors (Braga et al., 2007).

2. Rule-based disambiguation

Prior to the work described in this study, the Google TTS engine performed homograph disambiguation using language-specific rules curated by linguists and engineers. One major class of rules specifies the appropriate pronunciation to be used when the homograph occurs in the context of certain nearby words or phrases; for instance, one may specify that the homograph *used* is pronounced as [juːst] when immediately followed by the token *to*, as in the sentence "She *used* to smoke." Another class of rules selects a pronunciation to be applied when the homograph is tagged as a certain part of speech (POS). Finally, for each homograph one rule must be specified as a default, used when no more specific rules apply. Some limitations of such rule systems can be illustrated by considering a recent bug reported to us concerning the English lexical homograph *winds*. For this word, the nominal pronunciation [wɪndz] is the default, but rules selects the verbal pronunciation [waɪndz] when the homograph is tagged as a verb, or when it is immediately followed by the word *up*. Both the word context and the POS rule may seem sensible and intuitive, but both fail on a sentence like "There may be *winds* up to 20 miles per hour", for which the nominal pronunciation is required. Here, the word context rule will overapply, as will the POS rule if—as is usually the case in this context—the tagger incorrectly identifies *winds* as a verb. Furthermore, the choice of default rules is in large part based on introspection, and without empirical observations it is not clear, for example, which of the two pronun-

[1] This cuts across the traditional distinction between those homographs which can or cannot be disambiguated by part of speech (Yarowsky, 1997), since the distinct pronunciations of the latter category may or may not have overlapping parts of speech. For instance, the lexical homograph *console* is largely disambiguated by part of speech, but both pronunciations of the lexical homograph *bass*—[beɪs] and [bæs]—are usually nouns.

ciations of the English word *live*—[laɪv] or [lɪv]—ought to be considered the default. This issue is particularly important since default rules are the only rules yet available for many low-resource languages. Because of such limitations, prior to this work, homograph disambiguation was a major source of bug reports filed against the Google TTS front-end: it was the most buggy component for Russian, and the second most buggy one for English.

3. Machine learning-based disambiguation

We therefore consider the possibility of using machine learning methods to improve the existing rule-based system. Our basic design uses a set of multinomial classifiers—one per homograph—selecting the best resolution given a featural representation of the local context. Unlike prior work which uses complex and novel techniques to weigh conflicting sources of evidence and to prevent overfitting (Hearst, 1991; Sproat et al., 1992; Yarowsky, 1997; Silva et al., 2012), we simply use discriminative training and regularization.

3.1. Homograph disambiguation features

By representing this task as supervised classification, we are free to choose arbitrary features of the input context. We made use of the following set of features for both morphosyntactic and lexical homographs:

Word context features These represent tokens one or two to the left and/or right of the target homograph, left- and right-context bigrams, and a single skipgram centered on the target. For instance, consider the sentence "It was a terrific, riveting, really fast *read* and really exciting and really horrifying, but managed to be really touching." For this example the word context features are represented by the strings `WL2:really`, `WL1:fast`, `WR1:and`, `WR2:really`, `WL2:really_WL1:fast`, `WR1:and_WR2:really`, and `WL1:fast_WR1:and`. Following Yarowsky (1997), we use equivalence classes for context tokens classified as instances of specific *semiotic classes* such as numbers, currencies, measure expressions, or letter sequences; see Ebden and Sproat (2014) for a full list. Thus the featural representation of "1993 to *present*" and "2017 to *present*" are identical, since *1993* and *2017* are both instances of the `DATE` semiotic class.

POS tag feature Prior work on homograph disambiguation makes heavy use of features derived from part of speech (POS) tags or other automatic morphosyntactic analyses. We therefore process tokenized sentences with a POS tagger and extract a feature representing the hypothesized POS tag of the homograph itself.[2] However, this feature is only available when synthesis is performed on server; our embedded TTS engines (i.e., those running on mobile devices not connected to the internet) lack an on-device POS tagger.

Capitalization feature Following Sproat et al. (1992), we extract a feature indicating whether the target homograph is uppercase, titlecase, or lowercase. This feature is particularly useful for those homographs where one sense is a proper name and the other is not, as in *Polish* vs. *polish*.

[2] We also experimented with features derived from POS tags of nearby words, but this did not improve accuracy overall.

Type of ambiguity	Count	Example
Morphosyntactic	78	*read*: [ɹiːd] vs. [ɹɛd]
Lexical	62	*bow*: [boʊ] vs. [baʊ]
Mixed	23	*use*: [juːz] vs. [juːs]

Table 1: Counts and examples of three major categories of homographs in the data set.

3.2. Model training

Features are fed to a multinomial log-linear (i.e., maxent) model. This model is trained using an internal library employed for a number of other classification tasks in our TTS front-end, including sentence boundary detection and word stress prediction (Hall and Sproat, 2013; Sproat and Hall, 2014). During training, we use batched stochastic gradient descent with a fixed learning rate of $\alpha = .1$, and L_1 regularization. Separate models are trained for each homograph.

3.3. Model hybridization

While one could simply replace the rule-based system with machine-learned classifiers, we also consider a hybrid variant in which non-default rules pre-empt the learned classifiers, and learned classifiers pre-empt default rules. This approach is based on our intuition that non-default rules have high precision but low recall, and thus most errors are due to misapplication of default rules.

3.4. Outline

In what follows, we describe data collection and evaluation procedures, and compare the machine-learned model to the existing rule-based system using a manually labeled, publicly available database. We also estimate the performance degradation associated with embedded models which do not have access to a POS tagger.

4. Materials & methods

The following section describes data collection and evaluation methods used to compare systems.

4.1. Data collection

Querying an existing pronunciation lexicon for US English, we selected a set of 163 homographs for this experiment. Nearly all have two pronunciations, but two homographs have three pronunciations. To faciliate error analysis, one of the authors then coded each homograph as either *morphosyntactic*, *lexical*, or *mixed*; the counts of these three categories are given in Table 1. We then randomly sampled sentences containing these homographs from English-language Wikipedia articles, and manually filtered these to remove non-English text, bibliographic entries, and the like. This resulted in roughly 100 sentences per homograph. These examples were then labeled by a team of English-speaking annotators. Examples of the each homograph were grouped into batches of 20, and at the start of each batch, the annotator was provided with dictionary definitions for each pronunciation of a homograph as well as unique tags ("word IDs") for each. They were then presented with an example sentence and asked to select the best word ID for the homograph (marked with bold) in that sentence. Annotators

were also permitted to mark an example as "ambiguous", in which case the example was discarded.[3] Each example was labeled by three separate annotators; if all three did not assign the same word ID to an example, the disagreement was resolved by a fourth, more experienced annotator.

4.2. Data release

We have released this labeled data under an Apache 2.0 license.[4] The primary data consists of tab-separated values (TSV) files in which each row contains the identity of a homograph, its word ID label, the sentence itself, and byte-based indices for the exact location of the homograph token. For instance, a row might indicate that the homograph *read* occurs at bytes 41–45 in a certain given sentence and that it has the word ID `read_present` in this example. The data set is randomly split into two files, one containing a 10% random sample stratified by homograph, reserved for evaluation, and another contains the remaining data, intended for training and development. A supplementary TSV file can be used to match word IDs to an IPA transcription of the corresponding pronunciation; for example, this file indicates that `read_present` is pronounced [ɹiːd].

4.3. Evaluation

We evaluate models using two metrics: micro-accuracy and macro-accuracy (or *mean average accuracy*). The former is simply the percentage of examples correctly classified across all homographs; the latter is the arithmetic mean of the per-homograph accuracies. For the server model, the L_1 coefficient is tuned to maximize micro-accuracy on held-out data; for the embedded model it is also used to keep model size below a certain threshold for on-device data. [5]

5. Results

5.1. Baseline accuracies

Table 2 gives accuracies for three baseline models, evaluating over the entire data set. The first baseline simply selects most likely (the *maximum likelihood estimate*; MLE) pronunciation for each instance. The second and third consist of the existing set of disambiguation rules; the server system makes use of the full set of rules, whereas the embedded system excludes POS-tag-based rules, since, as mentioned above, the tagger they depend on cannot currently be run on-device. We see that the existing rules outperform the MLE baseline, but that there is substantial residual error.

5.2. Model accuracies

Table 3 reports accuracies using disjoint training and evaluation sets for all six systems. The two ML systems outperform the rules-only systems (for which the training set is irrelevant), and hybrid ("rules + ML") systems both substantially outperform ML-only systems. As expected, server-based systems outperform comparable embedded systems which lack access to POS tag features. This is particularly pronounced for the ML-only systems. On the other hand, hybridization minimizes this distinction; both hybrid systems perform near ceiling. The best systems (the hybrid systems) obtain a 12.0% absolute and a 92.3% relative error reduction over the worst (embedded rules-only). All six systems are ranked the same by micro- and macro-accuracy, suggesting there is no need to make a distinction between the two metrics on this relatively well-balanced data set.

	Micro	Macro
MLE baseline	.850	.849
Embedded: rules	.869	.863
Server: rules	.893	.890

Table 2: Overall micro- and macro-accuracies for the MLE baseline and two rule-based systems.

	Micro	Macro
Embedded: rules	.870	.867
Server: rules	.890	.886
Embedded: ML	.926	.924
Server: ML	.954	.951
Embedded: rules + ML	.990	.990
Server: rules + ML	.990	.990

Table 3: Evaluation set micro- and macro-accuracies.

5.3. Error analysis

For all six systems, morphosyntactic homographs like *read*, *live*, and *lives* are more challenging than lexical homographs like *bass*. However, the server model performs significantly better on morphosyntactic homographs than the embedded model, presumably due to the presence of POS tag features. One of the most challenging homographs is the mixed homograph *present*. All systems incorrectly predict the verb form [pɹɪˈzɛnt] in place of the noun/adjective [ˈpɹɛzənt] in sentences like "Smith has played Trophy matches for the county from 1993 to *present*." Here the server model incorrectly tags *present* as `VB` (a bare verb), which may contribute to a downstream classification error.

5.4. Limitations

One limitation of all six systems is that they depend on existing tokenization and token classifications, and errors during these steps may propagate. The same is true for POS tagging features. While POS tagging accuracies are quite high in general, morphosyntactic homographs like *present* are instances of a systematic noun-verb ambiguity in English known to be a major source of tagging errors (Toutanova and Manning, 2000). Rules allow such errors to propagate, but discriminative training may also help to correct such errors, i.e., by weighing features so that the evidence for

[3] We do not provide a wider context since some prior work claims that annotators very rarely require additional context to select the appropriate sense (Hearst, 1991; Gale et al., 1992), and single sentences account for the majority of our TTS traffic. In support of this, $< 1\%$ of examples were labeled "ambiguous".

[4] `http://github.com/google/WikipediaHomographData`

[5] These models are stored using a binary wire format based on protocol buffers. At the time of writing our in-production server model for US English has roughly 65k non-zero weights and its serialization is 1.5 MiB in size; our in-production embedded model has 39k non-zero weights and is roughly 800 KiB.

word context overwhelms competing evidence from unreliable POS tag features. It is an open question whether higher-level features, such as those derived from a dependency parse, might help to prevent such errors. Secondly, L_1 regularization is an appropriate remedy for overfitting but it may be suboptimal for controlling size of the embedded model, since the smallest-magnitude weights are not necessarily the least important; feature hashing may be more appropriate for this purpose. Finally, one may suspect that there is substantial redundancy between the existing rules and the ML classifier features in the hybrid systems, but as of yet we do not have a principled method to detect rules subsumed by the ML classifier (or vis versa).

6. Future work

We anticipate that additional languages will pose new challenges and require us to enrich our feature set. For example, consider some challenges posed by Russian and Thai, respectively. In Russian, most homographs are the result of morphologically conditioned stress shifts not indicated in the orthography, and we anticipate that features based on detailed morphological analyses will be required for such homographs. Furthermore, Russian is richly inflected so it may be desirable to use lemmatization or stemming to create equivalence classes for word context features. Thai, on the other hand, is written in a script which lacks capitalization distinctions—making the capitalization feature inapplicable—and does not mark word boundaries (Tesprasit et al., 2003), and thus it may require more sophisticated tokenization schemes. Future work will consider the possibility of replacing or augmenting manually labeled examples with sources of weakly labeled data derived using label propagation (Hearst, 1991), word-aligned bilingual text from a machine translation system—as in Gale et al. (1992)'s approach to word sense disambiguation—or phoneme-aligned transcriptions from a speech recognition system. Furthermore, some recent work on word sense disambiguation employs recurrent neural networks to encode the context (Yuan et al., 2016), allowing for a much wider context window than just the surrounding four tokens used here; we anticipate this technique will also be useful for homograph disambiguation and may even eliminate the need for morphosyntactic features such as POS tags.

7. Conclusion

We have shown that a simple application of machine learning produces significant improvements—in one case, a 12% absolute error reduction—in homograph disambiguation, a key part of high-quality text-to-speech synthesis. Since launching the hybrid server and embedded systems for US English text-to-speech traffic at Google, we have seen a substantial decline in the number of incoming bugs pertaining to homograph disambiguation. Furthermore, we find that once-challenging bug fixes can be completed simply by adding a small number of labeled examples and regenerating the models. Finally, we make our data freely available in the hopes it will encourage future academic research on this understudied problem.

8. Acknowledgements

Thanks to Keith Hall for assistance with the maximum entropy library and early pilot work, and to the many annotators and engineers who participated in data collection and software development.

Braga, D., Coelho, L., and Resende, Fernando Gil V., J. (2007). Homograph ambiguity resolution in front-end design for Portuguese TTS systems. In *INTERSPEECH*, pages 1761–1764.

Ebden, P. and Sproat, R. (2014). The Kestrel TTS text normalization system. *Natural Language Engineering*, 21(3):1–21.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992). Using bilingual materials to develop word sense disambiguation methods. In *Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 101–112.

Hall, K. and Sproat, R. (2013). Russian stress prediction using maximum entropy ranking. In *EMNLP*, pages 897–883.

Hearst, M. A. (1991). Noun homograph disambiguation using local context in large text corpora. In *Using Corpora*, pages 185–188.

Silva, D. C., Braga, D., and Resende, Fernando Gil V., J. (2012). A rule-based method for homograph disambiguation in Brazilian Portugese text-to-speech. *Journal of Communications and Information Systems*, 27(1):1–9.

Sproat, R. and Hall, K. (2014). Applications of maximum entropy rankers to problems in spoken language processing. In *INTERSPEECH*, pages 761–764.

Sproat, R., Hirschberg, J., and Yarowsky, D. (1992). A corpus-based synthesizer. In *ICSLP*, pages 563–566.

Sproat, R., Black, A. W., Chen, S., Kumar, S., Ostendorf, M., and Richards, C. (2001). Normalization of non-standard words. *Computer Speech & Language*, 15(3):287–333.

Taylor, P. (2009). *Text-to-speech synthesis*. Cambridge University Press, Cambridge.

Tesprasit, V., Charoenpornsawat, P., and Sornlertlamvanich, V. (2003). A context-sensitive homograph disambiguation in Thai text-to-speech synthesis. In *NAACL*, pages 103–105.

Toutanova, K. and Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *EMNLP*, pages 63–70.

Yarowsky, D. (1997). Homograph disambiguation in text-to-speech synthesis. In Jan P. H. van Santen, et al., editors, *Progress in speech synthesis*, pages 157–172. Springer, New York.

Yuan, D., Richardson, J., Doherty, R., Evans, C., and Altendorf, E. (2016). Semi-supervised word sense disambiguation with neural models. In *COLING*, pages 1374–1385.

Text Normalization Infrastructure that Scales to Hundreds of Language Varieties

Mason Chua, Daan van Esch, Noah Coccaro, Eunjoon Cho, Sujeet Bhandari, Libin Jia
Google LLC, 1600 Amphitheatre Parkway, Mountain View, CA 94043
{aftran, dvanesch, noahc}@google.com

Abstract

We describe the automated multi-language text normalization infrastructure that prepares textual data to train language models used in Google's keyboards and speech recognition systems, across hundreds of language varieties. Training corpora are sourced from various types of data sets, and the text is then normalized using a sequence of hand-written grammars and learned models. These systems need to scale to hundreds or thousands of language varieties in order to meet product needs. Frequent data refreshes, privacy considerations and simultaneous updates across such a high number of languages make manual inspection of the normalized training data infeasible, while there is ample opportunity for data normalization issues. By tracking metrics about the data and how it was processed, we are able to catch internal data preparation issues and external data corruption issues that can be hard to notice using standard extrinsic evaluation methods. Showing the importance of paying attention to data normalization behavior in large-scale pipelines, these metrics have highlighted issues in Google's real-world speech recognition system that have caused significant, but latent, quality degradation.

Keywords: language model, text normalization, internationalization, industrial systems, data mining, data processing, scale, less-resourced languages

1. Introduction

As technology adoption increases, products need to support ever more language varieties (ITU/UNESCO Broadband Commission, 2017). For example, there is a clear need for spell-checking and keyboards in hundreds or even thousands of language varieties. These applications typically require a language model, as do more advanced technologies like speech recognition systems. It would not be feasible to maintain separate training pipelines for all these use cases multiplied by all these language varieties, so we use one unified pipeline that is flexible enough to train all the models. The training includes automatic, configurable preprocessing for each data set, allowing text from a variety of sources to be normalized into the domains appropriate for keyboard prediction and speech recognition (Schwarm and Ostendorf, 2002; Ju et al., 2008; Scannell, 2014).

2. Training Pipeline

Each language model is produced by interpolation of component models, which are trained on individual data sources. The training set for the typical component model begins as publicly crawled or logs data. It is processed by language identification classifiers and privacy-enhancing systems (including for anonymization). The data is then filtered and edited by a sequence of normalizers (section 2.1) specific to the use case and language variety.

The creation of the interpolated language model is controlled by a configuration file specifying how to find and normalize each data set, as well as global model options like interpolation method. The configuration system separates language-specific data and its preprocessing from the training algorithms used. A single training binary can therefore train a model for any supported language, and new languages can be added simply by adding a configuration file, without changes to the trainer logic. Due to this automation and abstraction, the amount of manual work required to add new languages or train new models does not depend on the size of the data sets.

Having a single training pipeline controlled by language-specific configuration files is critical to operating at large scale, across many language varieties and use cases. For any given language, there may be half a dozen different language models, each with their own configuration file, reflecting varying use cases, including keyboard input (Hellsten et al., 2017), on-device or cloud-based speech recognition, handwriting recognition, and more. In total, we currently maintain about 2,700 distinct configuration files (not all of which are used for production applications). Each of these language models may include more than a dozen different types of data sources.

2.1. Text Normalization

One critical part of the configuration files is the specification of the normalizations to apply to the textual data. Each training example is potentially modified or discarded by several functions, usually implemented as Thrax grammars (Roark et al., 2012), including the data cleaning in section 2.1.1, annotators required for class-based language models (Vasserman et al., 2015; van Esch and Sproat, 2017), and enforcers of spelling consistency. Other types of normalizers deal with language identification, privacy enhancement and anonymization, and offensive content filtering. The most complex type of normalizer uses language models trained by this same pipeline, e.g. to ensure consistent string variation (section 2.1.2) or capitalization.

2.1.1. Thrax-based Normalization

Most of the data preprocessing is done by grammars written in Thrax, which was chosen for its composability with other finite state transducers used in our systems (Mohri et al., 2008). In order to support several languages without a proportional increase in engineering cost, we use a

Work done while at Google LLC.

shared language-independent normalizer template of about 200 lines of code. It can be instantiated for each language by adding a few dozen more lines of code that define language-specific character sets and rewrite rules, and call upon the language's pronunciation models and word lists. The language-independent template is the composition of the following rewrite rules, with only minor differences between speech recognition (presented here) and keyboard.

1. Ensuring that tokens are always separated by a single space. Turns "hi there" into "hi there".
2. Applying any language-specific formatting fixes defined when the template is instantiated for a specific language. There are usually no fixes, but in Azerbaijani and Turkish, for example, we use this step to make sure dotted and dotless I are preserved by the language-independent lowercaser in the next step.
3. Lowercasing the sentence in languages that have case. For example, "Hello, Dr. Nduom, how are you?" becomes "hello, dr. nduom, how are you?"
4. Discarding any examples that are not associated with a pronunciation by the language-specific verbalizer. For example, "bbbbbbbbbbbx a cat stepped on the keyboard".
5. Detaching punctuation from words. This step simplifies upcoming steps and makes sure the language model recognizes all occurrences of a vocabulary item as the same regardless of what sentence-level punctuation it appears next to. For example, "hello, dr. nduom, how are you?" becomes "hello , dr . nduom , how are you ?"
6. Reattaching punctuation that is part of the word, like in the English abbreviations "no." and "esq." For example, "hello , dr . nduom , we shipped a no . 2 pencil to peppler st . yesterday ." becomes "hello , dr. nduom , we shipped a no. 2 pencil to peppler st. yesterday ." This step and the previous one are separate for the sake of code readability.
7. Deleting each freestanding punctuation symbol unless the speech recognizer for that language supports inputting the symbol by saying its name ("hello comma Doctor Nduom comma how are you question mark").
8. Making the spelling consistent for some common or product-relevant words. For example, correcting "youtobe" to "youtube" or "color" to "colour" in many varieties of English. This step uses hand-curated lists.
9. Restoring proper capitalization for internal symbols that will be expanded during recognition. For example, "meet at $time" becomes "meet at $TIME".
10. Applying any language-specific rewrites that are more easily expressed on normalized data. In Italian, for example, we ensure that numbers are written as Roman numerals in certain street names.
11. Deleting any extra spaces left over due to step 7.

The rewrite rules are typically much less than twenty lines of code, possibly reusing shared character classes, definitions or word lists. Step 4, for example, is implemented as the PASS_ONLY_VALID_SENTENCES acceptor below.

```
core_token = Optimize[
  config.GRAPHEME+ |
  config.LOANWORD_GRAPHEME+ |
  config.VERBALIZABLE];

token = Optimize[
  (config.INITIAL_PUNCTUATION?
  core_token config.FINAL_PUNCTUATION*) |
  non_terminal];

export PASS_ONLY_VALID_SENTENCES =
Optimize[
  (token u.space)*
  token
  config.FINAL_PUNCTUATION*];
```

It defines an acceptor of zero or more word-and-space pairs followed by an additional word and zero or more sentence-final punctuations. `config` is the language-specific configuration, which must provide grammars defining valid punctuations and lists of allowed words.

2.1.2. Learned Normalization

Keyboards and speech recognition systems should consistently produce text in a particular style in order to be useful to the user. Since the training data includes Web text and search queries, there will be typos and intentional variation in spelling, capitalization and punctuation. We use a learned normalizer, called the *string variation normalizer*, to remove some of these inconsistencies and make the training data match the style of a reference corpus.

While the individual actions performed on the string are expressed as Thrax grammars, the string variation normalizer allows us to specify them without context. Instead, it learns when to apply them from the reference corpus. For example, for some output like "During the Renaissance period, people visited B&Bs on Valentine's Day", the normalizer had to decide whether to capitalize "Renaissance", whether to write "period" or ".", whether to use an ampersand or "and" in "B&Bs", whether to capitalize "Valentine's" and "Day", and whether to use an apostrophe in "Valentine's". The string variation normalizer evaluates all possible sequences of edits of this sort and chooses the result with the best score according to a language model trained on the reference corpus.

The language model used by the string variation normalizer is trained using the same pipeline as the end-use language model itself, except with a corpus that is cleaner and usually human-curated. It might seem overcomplicated to introduce a language model to normalize the text for training the final language model, but this complexity is outweighed by the added value in terms of accuracy and stylistic consistency. The use of the string variation normalizer typically results in at least a 1% relative decrease in word error rate and a 1% relative increase in sentence accuracy for speech recognition when evaluated on a voice search test set. In

one case, the recognizer used by voice search had relative changes of -5.7% word error rate and +2.1% sentence accuracy.

3. Monitoring to Allow for Scaling

The complexity and diversity of normalization processes described in section 2.1 are justified by user-facing quality gains, confirmed by experiments showing higher accuracy on extrinsic evaluations. However, this complexity also creates opportunities for data normalization issues that are not easily caught by extrinsic evaluation. Incorrect removal of punctuation, for example, is more easily caught by inspecting the data before and after normalization. There are also opportunities for data to become corrupted at its source before it reaches the normalization pipeline, for example due to incorrect character encoding or unexpected control symbols.

These issues could be caught by carefully inspecting the intermediate data between every normalization step for every new model trained. However, we avoid this approach for three reasons: privacy considerations; the sheer number of language models across languages (each one with many data sets that undergo many steps of normalization); and wanting staff to be able to develop improvements to models even when they do not speak the language involved.

We instead automate the inspection process by logging metrics during training, exemplified in the next section. The monitoring of these metrics has allowed us to identify normalization issues in several production language models. In US English, for example, the fixing of identified issues has resulted in a 6.6% relative word error rate decrease on a test set consisting of Google Home queries. Without automatic monitoring, problems in languages that are less familiar to research and engineering staff would be even more likely to occur.

Due to the size of the data sets, which can contain tens of billions of sentences, even simple metrics, like the size of the corpus, require parallel computation enabled by Flume (Chambers et al., 2010). Operations are automatically combined at runtime to avoid repeated iterations over the data.

3.1. Possible Data Issues and their Associated Monitoring

This section describes some potential data corruption issues that can arise and how they are caught using logged metrics.

Low quality for one dialect or region. When a language is spoken in more than one region or country, we often include multiple regions in a single model. Selecting training data from the right regions warrants extra care because changes can severely impact quality for populations of users in ways that can only be detected with expensive region-specific evaluations or test sets that are not always available. We therefore tag each piece of data with the region it came from, usually at the country level, and audit the histogram of tags for unexpected changes.

Language change. The language models are automatically re-trained with fresh data every few weeks in order to keep up with some types of language change, broadly construed. This process works well for neologisms, trending search terms and seasonal variations in the distribution of topics being written about. Changes that involve new characters, however, require manual intervention, because the normalization process (section 2.1.1) filters out unrecognized characters. If a new currency symbol is added, like the Indian rupee sign was in 2010 (Everson, 2010), we want to notice and expand the currency character lists used by our Thrax grammars. There are an infinite variety of possible external issues that could lead to malformed data. For example, if a popular keyboard were to have a bug where it used the wrong (but visually identical) Unicode codepoint, we would want to be alerted to fix the data, rather than discarding it in step 4.

To help identify character normalization issues, we log the number of times each character occurs before and after normalization. A sudden or gradual increase of this count for a particular character is reason to investigate whether there has been a change to the writing system. We also log how many vocabulary items each character appears in. Unexpected changes can indicate issues in the normalizer pipeline. A bug that causes sentence-level punctuation to be attached to tokens, for example, might be caught due to an increase in the number of vocabulary items that contain punctuation characters.

Deletion during normalization. Normalizers, explained in section 2.1, have the power to discard or corrupt all the data before it reaches the trainer. If Web markup is accidentally included in the data, for example, normalizers that assume markup has been removed will discard the data due to unrecognized tokens. To detect this type of issue, we log how many times each normalizer discards, accepts or edits the input. Excessive discarding or sudden changes in the distribution of actions suggest a bug in the normalizer logic or an incorrect assumption about its inputs. The counts are automatically presented to experimenters in a stacked bar chart (figure 1).

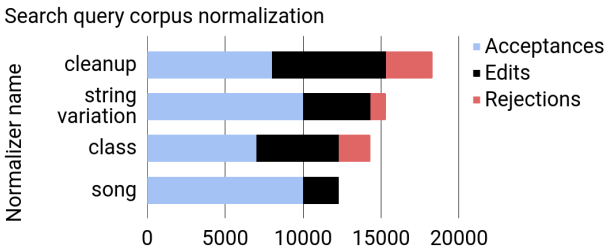


Figure 1: A made-up example of the horizontal stacked bar chart generated for each data set used by each model. The three colors of each bar show how many training examples were passed as-is, edited and deleted by each normalizer, respectively. The four normalizers were applied to the data in top-to-bottom order. Since the output of a normalizer is the input of the next normalizer, each bar is as long as the non-rejection part of the bar above it.

Data from the wrong language, script or corpus. Data might be annotated with the wrong language code due to quality issues with the internal language identification system or end-user devices misreporting their locale. Since

each corpus usually represents a different data source (such as voice search queries or crawled text), this issue is more likely to happen in a single corpus than all corpora at once. A model trained with significant amounts of data from the wrong language will be noticed before deployment due to worse sentence accuracy and word error rate. But those metrics do not directly show what happened, only that there is a problem. So we log the number of vocabulary items that are unique to each corpus. A high number may suggest incorrect language data in that corpus.

Similarly, normalizer grammar bugs can result in training data with non-canonical characters. For example, Greek text might contain Unicode Greek math symbols instead of standard Unicode Greek letters, which could decrease quality by creating both incorrect output text and more sparsity of contexts in the language model. So we also report character-based metrics, displaying each character's Unicode name (for example, "GREEK SMALL LETTER IOTA WITH DIALYTIKA AND TONOS") with its count in the data and the model's vocabulary. This also allows for faster detection of tokenisation issues that result in sentence-level punctuation attached to words.

Although we cannot foresee all data misclassification mistakes, many will involve sudden or gradual changes in corpus size. If, for example, all assistant-type queries were to start being classified as maps-type queries, every data refresh would shrink the assistant-type corpus and grow the maps-type corpus. Language model interpolation would also become less effective due to the apparent disappearing distinction between user behavior in the two products. So we monitor corpus sizes and interpolation weights for sudden changes and unexpected trends.

Inadequate pronunciation information. A speech recognizer is only as good as its implicit or explicit pronunciation model that associates phoneme sequences with text. To better diagnose quality issues, we log how many vocabulary items get a pronunciation in the recognizer purely by consulting a human-curated pronunciation dictionary, coupled with Thrax verbalization rules (Sak et al., 2013a; Sak et al., 2013b). The remaining items rely on some learned model (Wu et al., 2014; van Esch et al., 2016). If the number of words receiving a manual pronunciation is high, quality issues are unlikely to be due the learned model.

4. Conclusion

We described our text normalization infrastructure that trains language models for speech recognition and keyboard prediction in hundreds of language varieties. Most of its internals are language-independent, with language-specific behavior specified in configuration options, training data and a limited portion of normalizer grammars. The sharing of most infrastructure enables a common set of auditing techniques that have allowed us to catch text normalization errors across languages.

5. Bibliographical References

Chambers, C., Raniwala, A., Perry, F., Adams, S., Henry, R. R., Bradshaw, R., and Weizenbaum, N. (2010). Flumejava: easy, efficient data-parallel pipelines. In *ACM Sigplan Notices*, volume 45, pages 363–375. ACM.

Everson, M. (2010). Proposal to encode the INDIAN RUPEE SIGN in the UCS. In *Universal Multiple-Octet Coded Character Set*. International Organization for Standardization.

Hellsten, L., Roark, B., Goyal, P., Allauzen, C., Beaufays, F., Ouyang, T., Riley, M., and Rybach, D. (2017). Transliterated mobile keyboard input via weighted finite-state transducers. In *Proceedings of the 13th International Conference on Finite State Methods and Natural Language Processing (FSMNLP 2017)*, pages 10–19.

ITU/UNESCO Broadband Commission. (2017). The State of Broadband 2017: Broadband catalyzing sustainable development. *Broadband Commission for Sustainable Development*.

Ju, Y.-C., Odell, J., and Ju, Y. C. (2008). A language-modeling approach to inverse text normalization and data cleanup. International Speech Communication Association, September.

Mohri, M., Pereira, F., and Riley, M. (2008). Speech recognition with weighted finite-state transducers. In *Springer Handbook of Speech Processing*, pages 559–584. Springer.

Roark, B., Sproat, R., Allauzen, C., Riley, M., Sorensen, J., and Tai, T. (2012). The opengrm open-source finite-state grammar software libraries. In *Proceedings of the ACL 2012 System Demonstrations*, pages 61–66. Association for Computational Linguistics.

Sak, H., Beaufays, F., Nakajima, K., and Allauzen, C. (2013a). Language model verbalization for automatic speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8262–8266. IEEE.

Sak, H., Sung, Y.-h., Beaufays, F., and Allauzen, C. (2013b). Written-domain language modeling for automatic speech recognition. In *INTERSPEECH*, pages 675–679.

Scannell, K. (2014). Statistical models for text normalization and machine translation. In *Proceedings of the First Celtic Language Technology Workshop*, pages 33–40.

Schwarm, S. and Ostendorf, M. (2002). Text normalization with varied data sources for conversational speech language modeling. In *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, volume 1, pages I–789. IEEE.

van Esch, D. and Sproat, R. (2017). An expanded taxonomy of semiotic classes for text normalization. *Proc. Interspeech 2017*, pages 4016–4020.

van Esch, D., Chua, M., and Rao, K. (2016). Predicting pronunciations with syllabification and stress with recurrent neural networks. In *INTERSPEECH*, pages 2841–2845.

Vasserman, L., Schogol, V., and Hall, K. (2015). Sequence-based class tagging for robust transcription in asr. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Wu, K., Allauzen, C., Hall, K., Riley, M., and Roark, B. (2014). Encoding linear models as weighted finite-state transducers. In *Fifteenth Annual Conference of the International Speech Communication Association*.

DeModify: A Dataset for Analyzing Contextual Constraints on Modifier Deletion

Vivi Nastase[◇], Devon Fritz, Anette Frank[◇]
Leibniz ScienceCampus "Empirical Linguistics and Computational Language Modeling"[◇]
Institute of Computational Linguistics, Heidelberg University
Heidelberg, Germany
{nastase, frank}@cl.uni-heidelberg.de, devon.s.fritz@gmail.com

Abstract

Tasks such as knowledge extraction, text simplification and summarization have in common the fact that from a text fragment a smaller (not necessarily contiguous) portion is obtained by discarding part of the context. This may cause the text fragment to acquire a new meaning, or even to become false. The smallest units that can be considered disposable in a larger context are modifiers. In this paper we describe a dataset collected and annotated to facilitate the study of the influence of modifiers on the meaning of the context they are part of, and to support the development of models that can determine whether a modifier can be removed without undesirable semantic consequences.

Keywords: annotation, modifier deletion, text simplification

1. Introduction

Knowledge extraction, keyword identification, text simplification, or summarization are useful tasks that rely on the assumption that certain information from texts can be discarded without negative consequences. Certain details are peripheral and can be disregarded. Removing subordinate clauses is a common practice in extractive summarization, for example (Vanderwende et al., 2007; Zajic et al., 2007).
While modifiers can be quite complex – ranging from a single word to a full clause – we focus here on single-word modifiers, the smallest unit of context that can be removed. While even their name suggests a peripheral role in the overall meaning of a text, modifiers can impact both their local and global context. The impact of modifiers on their local context (themselves plus their syntactic head), also called modification distortion (Murphy, 2002), has been studied mostly out of context, and there are several datasets that allow for the study of this phenomenon (Kruszewski and Baroni, 2014; Schulte im Walde et al.,). The dataset we built and present here is focused on the effect of modifiers on the larger context. For example, the following title of a TED talk would mean something completely different should the modifier *nearly* be removed:

> AJ Jacobs: How healthy living nearly killed me

While *nearly* could be argued to be a special modifier, the same situation may arise for "normal" modifiers, such as the adjective *old* – it can be removed from the following context without a dramatic impact on the meaning of the sentence:

> Then she saw the old parish priest pull up in his car.

but it is an essential element of the story, when the entire context is included – a short story from the ROC corpus (Mostafazadeh et al., 2016):

> Joan entered the confessional and kneeled. She thought she was confessing to the old parish priest. Joan confessed she had fantasized about the young visiting priest. Joan felt relief as she left the confessional. Then she saw the old parish priest pull up in his car.

Understanding the influence of modifiers is important, as it affects compositional models of language, as well as higher-level tasks such as summarization or textual entailment. As seen in the above examples, modifiers, while syntactically omissible, can make important semantic contributions to the information conveyed by a larger context, such that deleting them may considerably alter the meaning of the sentence and its context.
In this paper we describe a dataset of complete short texts (approximately 5 sentences each) in which one open-class modifier has been annotated with one of three classes based on its impact on the text it appears in: crucial, not-crucial, ungrammatical. The dataset consists of 3632 instances, which we provide with their multiple annotations obtained through CrowdFlower. We describe in the paper two potential gold standards – one obtained by using only instances where all annotators agree consisting of 1767 instances, and one obtained with majority voting, consisting of 3542 instances. We also include a split into 5 folds, to be used for future experiments.[1]

2. Related work

Text simplification can occur at different levels of granularity – extracting a sentence from a document, deleting a phrase from a sentence or a sub-phrase from a larger chunk. Vanderwende et al. (2007), Zajic et al. (2007) propose syntax-based trimming, where branches of a syntactic tree are scored using a combination of features that

[1] http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/deModify/deModify_data.shtml

marks them for potential deletion. Wubben et al. (2012) approach the problem of text simplification as a machine translation problem trained on pairs of texts from Wikipedia and SimpleWikipedia. Wang et al. (2016), Zhang and Lapata (2017) reformulate this approach in the form of neural encoder-decoder models.

Focusing on the modifiers, modification can be viewed from many different perspectives. From a linguistic point of view, several typologies of modifiers have been proposed (McNally, 2013). Of these, the semantic impact of modifiers is taken into account in: (a) the *entailment-based* typology, in which modifiers are grouped into three broad categories based on the inferences they license, which stem from potential interpretations of the extension of modifiers, head nouns and the compounds as sets: *intersective* modifiers (*male nurse*), *subsective* modifiers (*molecular scientist*), *intensional* modifiers (*alleged crook*); (b) *pragmatic/discourse-related* typologies which partition modifiers based on their impact on the utterance in which they appear – those that affect the interpretation of the utterance, and those that do not, and modifiers are considered separately by POS or phrase type.

The entailment-based typology is the focus of Amoia and Gardent (2007), who study the inferential properties of adjectives and how these classes influence the omissibility of adjectives under truth-conditional aspects. Amoia and Gardent (2008) published a data set where these and other syntactic and semantic properties of adjectives are tested in an RTE (recognizing textual entailment) setting. In this work, the *context* in which the semantic effects of adjectives are tested is the sentence, and the relevant criterion is preservation of truth when, e.g., deleting the adjective or the head noun as in the following inference pairs: *Daisy is a big mouse* $\rightarrow$ *Daisy is a mouse* or *Daisy is a big mouse* $\rightarrow$ *Daisy is big*. Along similar lines, Stanovsky and Dagan (2016) describe the construction process and resulting dataset of non-restrictive noun phrase modification. Non-restrictive modifiers – e.g. *The speaker thanked president Obama who just came into the room* – can be removed to shorten sentences. The dataset gathered and annotated trough crowd sourcing has no restrictions on the length of the modifiers, which often span phrases. The context provided for annotation is a sentence for each modifier.

From a conceptual point of view, modifiers were studied with respect to the distortion effect they have on the concept denoted by the head noun (Murphy, 2002). Kruszewski and Baroni (2014) focus on the computational study of the effect of *modifier-triggered (head) distortion*, and built and annotated a dataset of (out of context) noun compounds, with respect to their "place" in a hierarchy of concepts. Each modifier-head compound (e.g., *perfume bottle*) is rated with membership and typicality scores against 3 criteria: how well it fits under the concept denoted by the head (*perfume bottle* $\rightarrow$ *bottle*), how well it fits under a superconcept of the head (*perfume bottle* $\rightarrow$ *drinkware*), and how prototypical the concept denoted by the head is of the super concept (*bottle* $\rightarrow$ *drinkware*). The final ratings are averages over individual scores gathered through surveys on CrowdFlower. The data thus collected is tested from the point of view of compositionality, expecting that more typical instances of a class are modelled more successfully using compositional operations on the individual vectors. Schulte im Walde et al. () built a dataset of 868 German noun-noun compounds, where one of the annotations quantifies the compositionality of the compound on a scale of 1 (semantically opaque) to 6 (semantically transparent).

In contrast to Kruszewski and Baroni (2014) and Schulte im Walde et al. (), our dataset focuses on the semantic effects that modifier deletion may have on the wider context, i.e., is not limited to the modified phrase. Modifiers in our dataset may be considered crucial to the larger context in which they appear, although they may not have a distorting effect on their local syntactic head – e.g. the adjective *old* in the noun phrase *old parish priest* ("old parish priest" is still "parish priest") – nor within the full sentence.

We also go beyond Amoia and Gardent (2008)'s and Stanovsky and Dagan (2016) work in that we consider a full (short) story context for judging omissibility – the example with the adjective *old* in the Introduction section shows that the sentence context (where the deletion might be deemed acceptable) can be overriden by the larger context. Furthermore, we judge *informativeness* as opposed to *preservation of truth*, which yields a more natural criterion for omissibility in e.g. text simplification tasks.

3. The *DeModify* Dataset: Data Selection, Annotation Process and Data Statistics

3.1. Data selection and annotation categories

The problem that data selection poses is that the phenomenon we target is not explicitly marked. A random selection of texts may not contain a significant number of positive instances. We follow a deliberate strategy, inspired by Grice's conversational maxims (Grice, 1975), and target collections of short texts with a specific communication intent, which are likely to contain (mostly) relevant and important details. Furthermore, we choose short self-contained texts as a basis for our annotations to ensure that at the onset we have a complete context which is easy and fast to read for the annotators. In these texts we mark one modifier (following the methodology described below), and we present the text and the marked modifier to annotators through CrowdFlower[2]. The annotators are asked to assign the marked modifier one of three categories:

crucial – the modifier is crucial for preserving the intended meaning, removing it leads to a different/erroneous/false interpretation of the remaining text:

> ... She thought she was confessing to the old parish priest. Joan confessed she had fantasized about the young visiting priest. ... Then she saw the old parish priest pull up in his car.

not crucial – the modifier is not crucial, removing it would not result in a distortion of the meaning of the remaining text:

> ... The trading of information is obviously driven by greed of gain. ...

[2] https://www.crowdflower.com

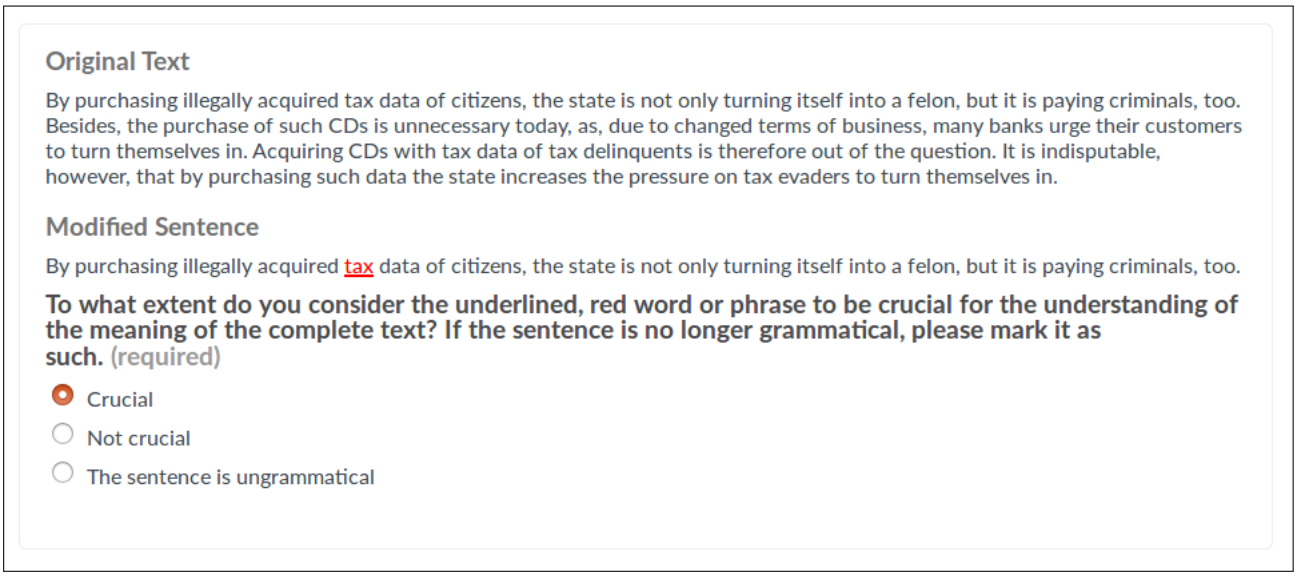


Figure 1: CrowdFlower interaction for annotation.

dependency relation	frequency in ROCStories	frequency in microtexts	examples
advmod	48,477	317	She is happy that she finally baked a cherry pie.
amod	69,066	576	Josh patted himself on the back for making a good decision.
compound	37,714	267	Bella made dessert for her family dinner.
nmod:tmod	9,140	10	Katarina lost her first tooth yesterday.

Table 1: Examples for the four selected dependency relations for modifier selection.

ungrammatical – removing the modifier results in an ungrammatical sentence:

> ... this ideal appears unworthy of support in many ways. ...

We selected instances in our dataset from short stories from the ROCStories corpus[3] (Mostafazadeh et al., 2016) and the short argumentative texts[4] (Peldszus and Stede, 2016). All these texts are approximately 5 sentences long, and self-contained.

Modifier selection is based on a syntactic analysis of the texts: the Stanford Dependency Parser is used to parse the data, and we compute statistics on the frequency of grammatical dependencies other than root/subject/object that connect open-class words. We select the 4 most frequent dependency relations, presented in Table 1.

Based on frequency statistics for the modifier lemmas that occur in the chosen dependency relations, they are split into 5 frequency bands. We then chose at most 15 instances for a random selection of modifiers from each band for inclusion in the final dataset, which contains a total of 3632 instances:

ROCStories – 3026 instances. The ROCStories corpus (Mostafazadeh et al., 2016) is a collection of about 49,000 self-contained stories of at most 5 sentences. From this large dataset we selected approximately 4000 stories. After the filtering criteria mentioned above, 3026 stories with one marked modifier from each story were kept.

Argumentative Microtexts – 606 instances. The argumentative microtexts corpus (Peldszus and Stede, 2016) consists of 112 texts of 3-5 sentences each. We selected more than one modifier from each of these texts according to the same process as for the ROCStories.

3.2. Annotation Process

This raw dataset is presented to CrowdFlower users. For each instance the complete text of the story/microtext is shown, and below it the targeted modifier is shown in red in its sentence context. The users are given three choices for annotation – *crucial (C), not crucial (N), ungrammatical (U)*. Before the actual annotation exercise started, the CrowdFlower users were given instructions and shown both positive and negative commented examples. A partial screenshot of the annotation interaction is shown in Figure 1 and the instructions for the annotation process are included in Figure 3 in the Appendix.

The set-up was first tested in several iterations (on friends and family). We started with 5+1 options: 5 options ranging in stages from *modifier deletable without consequences* to *modifier not deletable* + one additional label: *ungrammatical*. However, we found that the task was clearer with a binary decision regarding the impact of the modifier being *crucial* vs. *not crucial*, plus the ungrammaticality option.

The data was released in batches of 100. 32 control in-

[3] http://cs.rochester.edu/nlp/rocstories

[4] https://github.com/peldzsus/arg-microtexts

label combination	frequency	percentage
C C C	442	12.2%
N N N	1248	34.4%
U U U	77	2.1%
C C N	566	15.6%
C C U	132	3.6 %
N N C	843	23.2%
N N U	86	2.4%
U U C	120	3.3%
U U N	28	0.8%
C N U	89	2.4%

Table 2: Agreement percentages for the different label combinations: strict agreement: rows 1-3.

strict GS: 1089 unique heads, 399 unique modifiers

microtexts	265	N	174	U	7	C	84
ROCStories	1502	N	1074	U	70	C	358
all	1767	N	1248	U	77	C	442
in %	100	N	70.6	U	4.4	C	25.0

relaxed GS: 1773 unique heads, 585 unique modifiers

microtexts	587	N	336	U	32	C	219
ROCStories	2955	N	1841	U	193	C	921
all	3542	N	2177	U	225	C	1140
in %	100	N	61.4	U	6.4	C	32.2

Table 3: Gold standard datasets statistics: number of instances from each text source, and annotation frequencies.

stances – where the answer was known (annnotated by one of the authors who is a native speaker of English) – were included in the batches, and CrowdFlower's internal mechanisms were used to filter out annotators with low levels of (automatically computed) trust. In the end each instance was annotated by at least three CrowdFlower users. One of the authors (native English speaker) annotated the control instances and performed an additional evaluation on a preliminary test run of the system on 100 instances. On a random selection of 20 instances our annotator confirmed agreement with the judges (not all of whom are native English speakers) in 17 of the 20 cases. Table 2 shows the agreement counts for each combination of labels in the data.

3.3. Data Statistics

We derived two gold standard versions from the annotations: one by **strict** agreement (all annotators agree), and one by **relaxed** agreement (majority voting). The distribution of the instances into annotation classes for both versions, divided by text sources, is presented in Table 3.

The **strict GS** is about half the size of the **relaxed GS**. The proportion of categories (N, C, U) is comparable in both versions, with N(oncrucial) covering about 70/60 percent of the instances, followed by C(rucial) with about a quarter/third of the instances and a small U(ngrammatical) class. Table 2 shows the detailed counts for all annotation

gold standard	nb. of classes		modifiers	heads
strict	1		253	939
		C	88	209
		N	156	688
		U	9	42
	2		123	137
		C-N	110	122
		C-U	4	3
		N-U	9	12
	3		23	13
relaxed	1		332	1364
		C	132	408
		N	181	884
		U	19	72
	2		173	365
		C-N	154	306
		C-U	8	22
		N-U	11	37
	3		80	44

Table 4: Number of classes (out of our three – C, N, U) in which modifiers and heads appear, for strict and relaxed (majority voting) gold standards.

combinations.

The purpose for constructing this dataset was to facilitate the study of the impact of context on the deletability of modifiers. For a robust study it would be interesting to have a spread of the instances over the three classes, or rather, mostly *crucial* and *not-crucial* (which as Table 3 shows is the case), but it would also be extremely interesting if the same modifiers appear with different class annotations. This is an issue that we could not control for during annotation, but we have tested the resulting dataset whether such phenomena were captured. If the dataset includes modifiers and heads that appear in more than one class, an automatic system would be forced to take into account the context for prediction. The plots in Figure 2 and the statistics in Table 4 confirm that this is accomplished: 30% (39%) of the modifiers appear in both the *crucial* and *not-crucial* classes for the strict (relaxed) gold-standard. This also confirms the point we made in the Introduction, that the same modifiers will behave differently in different contexts with respect to the phenomenon targeted here.

An example from the annotated data of the same modifier belonging to different classes in different contexts is included in Table 5 for the adjective *long*.

We noted a certain degree of confusion between the *ungrammatical* and *crucial* for the adjective *long*: when it appears in the phrase *as long as* it is sometimes annotated as *crucial*, and sometimes as *ungrammatical*.

3.4. Data files

We provide an archive through our institute's website (`http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/deModify/deModify_data.shtml`) with two main files: the dataset with CrowdFlower annotations, and a file with a proposed split into 5 folds.

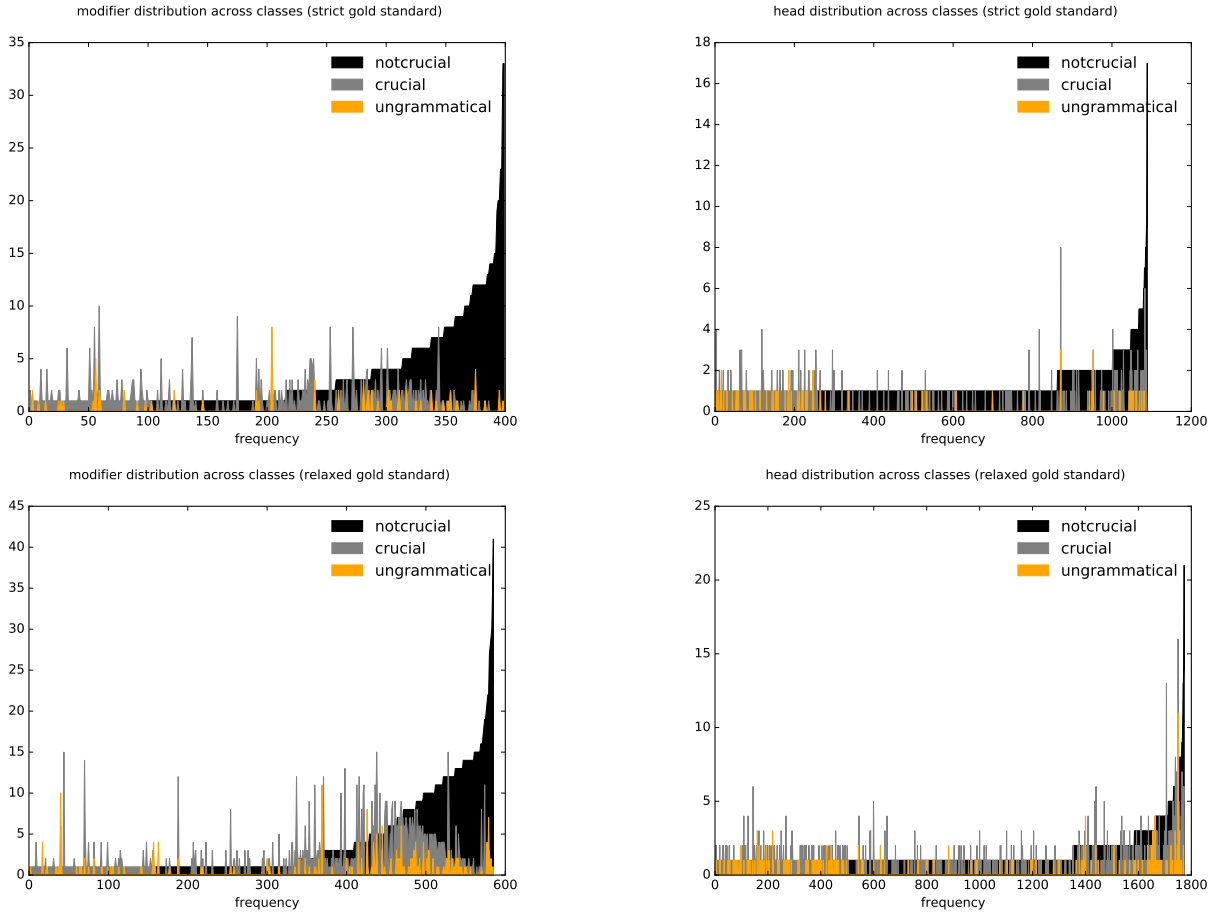


Figure 2: Histogram for modifier (left) and head (right) frequencies by annotation classes, when considering strict (all annotators agree, top) or relaxed (majority voting, bottom) gold standards: every point on the x axis corresponds to a word, and the colored bars show its frequency in each of the three classes (words on the x axis are ordered by increasing frequency in the *not-crucial* (N) class).

long: instances in different classes	
crucial	However, a death would not be of any more use to those affected and their relatives than if the felon receives a **long** sentence. (instance id: 1101714199)
not-crucial	They loved to go on **long** walks together. (instance id 1102590593)
ungrammatical	We put on **long** sleeves and jackets. (instance id 1102590621)
	You should watch less television, as too much TV makes you stupid in the **long** run, like your brother. (instance id 1101713906)

Table 5: Instances in different classes for the adjective **long**

The data (file demodify.tsv) consists of 3632 entries on 10896 lines, each entry consisting of 3 lines which provide the information listed in Table 6 (all lines provide the same information).

We include a file (demodify.data_split.tsv) that gives a proposed split of the data into 5 balanced folds (with respect to the classes). The file contains an assignment to fold for both the strict (full agreement) and relaxed (majority agreement) class assignments.

4. Conclusion

We described a novel dataset for the study of the influence of single-token modifiers on the larger textual context. The DEMODIFY dataset consists of short (up to 5 sentences) self-contained texts, in which selected modifiers are marked and annotated with one of three categories: *crucial* (removing them would distort the meaning of the remaining text), *not crucial* (the modifier can be removed without drastic consequences) or *ungrammatical* (removing the modifier would result in an ungrammatical sentence). The final dataset consists of 3632 instances annotated through CrowdFlower by at least 3 judges. Control instances and CrowdFlower trustiness measures were used to filter the judges and ensure high-quality annotations.

The DEMODIFY dataset is publicly available[5] It can be

[5] http://www.cl.uni-heidelberg.de/english/research/downloads/resource_pages/deModify/

_unit_id	unique instance id for each entry, shared by all lines that contain annotations for this entry
_created_at	time of creation
annotation	one of three classes: crucial, not crucial, ungrammatical
head	a head word
head_word_index	the position of this word in the sentence
modifier	the modifier that is marked for deletion, whose head is in the "head" column
modifier_type	the dependency relation between the head and the modifier
sentence_number	the sentence number relative to the full text
source	the name of the source corpus: microtexts/ROCStories
storyid	the id of the story/microtext
title	title of the story/microtext – if there is one
full_story	the full text of the story/microtext
original_sentence	the sentence in which the modifier appears
_trust	the trust measure of the annotator, computed automatically by CrowdFlower
_country	the country of the annotator
_region_city	the city of the annotator

Table 6: Data file description

used to investigate *linguistic factors* underlying the observed behaviour of modifiers in context, the range of influence of the modifier on a context and the interacting elements of the context. Ultimately we hope this dataset will contribute towards a better understanding of pragmatic influences in text semantics.

5. Acknowledgments

We thank our reviewers for their comments. This research was funded by the Leibniz Science Campus Empirical Linguistics & Computational Language Modeling, supported by Leibniz Association grant no. SAS-2015-IDS-LWC and by the Ministry of Science, Research, and Art of Baden-Württemberg.

6. Bibliographical References

Amoia, M. and Gardent, C. (2007). A first order semantic approach to adjectival inference. In *Workshop on textual entailment and paraphrasing (WTEP)*, Prague.

Amoia, M. and Gardent, C. (2008). A testsuite for inference involving adjectives. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.

Grice, H. P. (1975). Logic and conversation. In P. Cole et al., editors, *Syntax and Semantics 3: Speech Acts*, pages 41–58. Academic Press, New York, N.Y.

Kruszewski, G. and Baroni, M. (2014). Dead parrots make bad pets: Exploring modifier effects in noun phrases. In *Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014)*, pages 171–181, Dublin, Ireland.

McNally, L. (2013). Modification. In Maria Aloni et al., editors, *Cambridge Handbook of Semantics*. Cambridge University Press.

Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., Kohli, P., and Allen, J. (2016). A Corpus and Cloze Evaluation for Deeper Understanding of Commonsense Stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California.

Murphy, G. (2002). *The Big Book of Concepts*. MIT Press.

Peldszus, A. and Stede, M. (2016). An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon 2015*, pages 801–815, London. College Publications.

Schulte im Walde, S., Hätty, A., Bott, S., and Khvtisavrishvili, N.). GhoSt-NN: A Representative Gold Standard of German Noun-Noun Compounds. In *Proceedings of the 10th Conference on Language Resources and Evaluation (LREC)*, Portoroz, Slovenia.

Stanovsky, G. and Dagan, I. (2016). Annotating and Predicting Non-Restrictive Noun Phrase Modifications. In *Proc. of the 54th Ann. Meeting of the Assoc. for Computational Linguistics*, pages 1256–1265, Berlin, Germany.

Vanderwende, L., Suzuki, H., Brockett, C., and Nenkova, A. (2007). Beyond SumBasic: Task-focused Summarization with Sentence Simplification and Lexical Expansion. *Information Processing and Management*, 43(6):1606–1618, November.

Wang, T., Chen, P., Rochford, J., and Qiang, J. (2016). Text Simplification Using Neural Machine Translation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 4270–7271. AAAI Press.

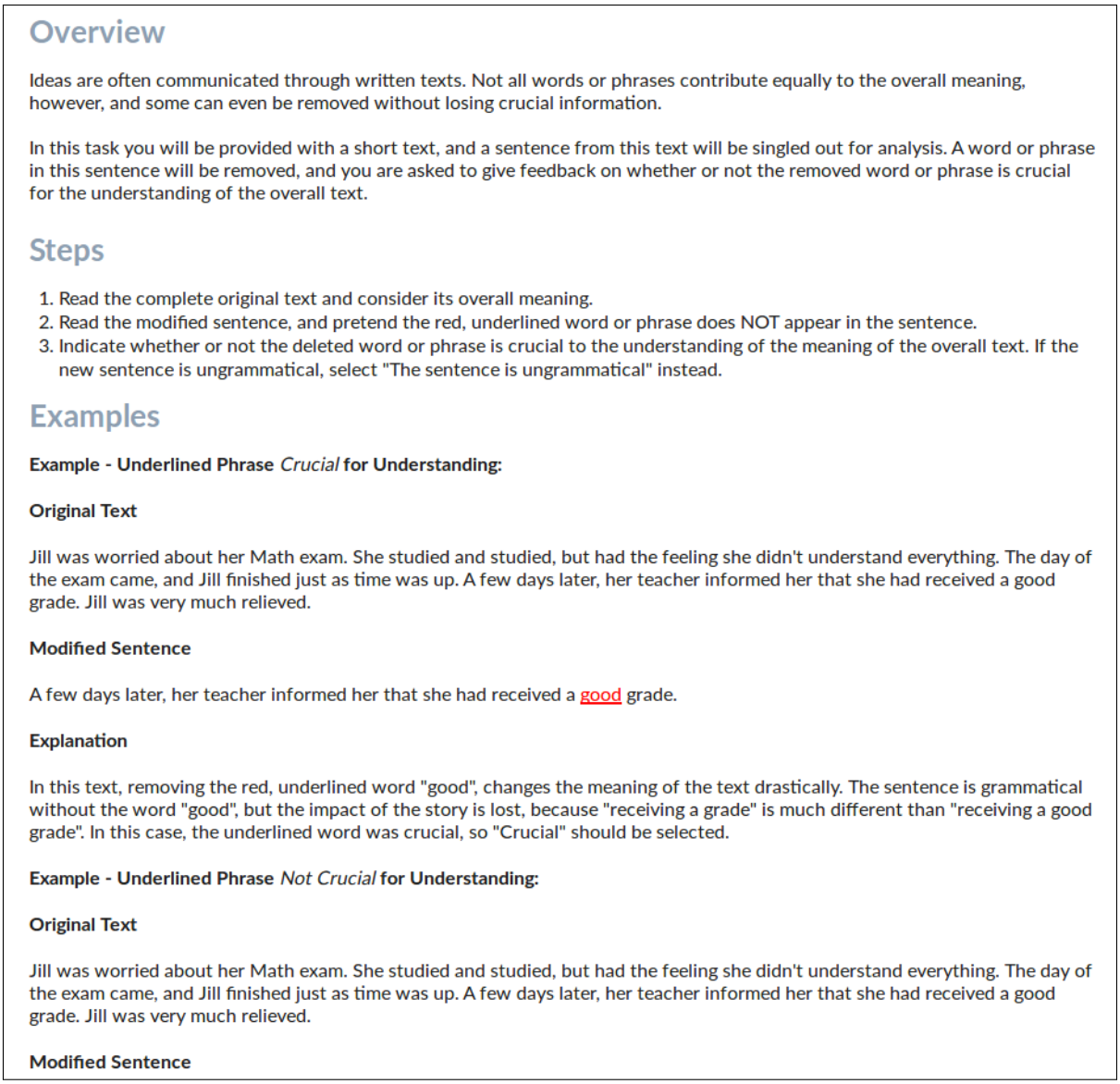
Wubben, S., Van Den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 1015–1024.

Zajic, D., Dorr, B., Lin, J., and Schwartz, R. (2007). Multi-Candidate Reduction: Sentence Compression as a Tool for Document Summarization Tasks. *Information Processing and Management*, 6(43):1549–1570.

Zhang, X. and Lapata, M. (2017). Sentence Simplification with Deep Reinforcement Learning. In *Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 595–605, Copenhagen, Denmark.

deModify_data.shtml

Appendix



Overview

Ideas are often communicated through written texts. Not all words or phrases contribute equally to the overall meaning, however, and some can even be removed without losing crucial information.

In this task you will be provided with a short text, and a sentence from this text will be singled out for analysis. A word or phrase in this sentence will be removed, and you are asked to give feedback on whether or not the removed word or phrase is crucial for the understanding of the overall text.

Steps

1. Read the complete original text and consider its overall meaning.
2. Read the modified sentence, and pretend the red, underlined word or phrase does NOT appear in the sentence.
3. Indicate whether or not the deleted word or phrase is crucial to the understanding of the meaning of the overall text. If the new sentence is ungrammatical, select "The sentence is ungrammatical" instead.

Examples

Example - Underlined Phrase *Crucial* for Understanding:

Original Text

Jill was worried about her Math exam. She studied and studied, but had the feeling she didn't understand everything. The day of the exam came, and Jill finished just as time was up. A few days later, her teacher informed her that she had received a good grade. Jill was very much relieved.

Modified Sentence

A few days later, her teacher informed her that she had received a good grade.

Explanation

In this text, removing the red, underlined word "good", changes the meaning of the text drastically. The sentence is grammatical without the word "good", but the impact of the story is lost, because "receiving a grade" is much different than "receiving a good grade". In this case, the underlined word was crucial, so "Crucial" should be selected.

Example - Underlined Phrase *Not Crucial* for Understanding:

Original Text

Jill was worried about her Math exam. She studied and studied, but had the feeling she didn't understand everything. The day of the exam came, and Jill finished just as time was up. A few days later, her teacher informed her that she had received a good grade. Jill was very much relieved.

Modified Sentence

Figure 3: CrowdFlower instructions including positive and negative examples for annotation.

Open Subtitles Paraphrase Corpus for Six Languages

Mathias Creutz
Department of Digital Humanities, Faculty of Arts, University of Helsinki, Finland
mathias.creutz@helsinki.fi

Abstract

This paper accompanies the release of *Opusparcus*, a new paraphrase corpus for six European languages: German, English, Finnish, French, Russian, and Swedish. The corpus consists of paraphrases, that is, pairs of sentences in the same language that mean approximately the same thing. The paraphrases are extracted from the OpenSubtitles2016 corpus, which contains subtitles from movies and TV shows. The informal and colloquial genre that occurs in subtitles makes such data a very interesting language resource, for instance, from the perspective of computer assisted language learning. For each target language, the Opusparcus data have been partitioned into three types of data sets: training, development and test sets. The training sets are large, consisting of millions of sentence pairs, and have been compiled automatically, with the help of probabilistic ranking functions. The development and test sets consist of sentence pairs that have been checked manually; each set contains approximately 1000 sentence pairs that have been verified to be acceptable paraphrases by two annotators.

Keywords: paraphrase, subtitle, colloquial language, annotation, ranking, German, English, Finnish, French, Russian, Swedish

1. Introduction

This paper introduces the first release of the Opusparcus multilingual corpus of paraphrases (Creutz, 2018). Paraphrases are pairs of phrases in the same language that essentially convey the same meaning, such as *"Have a seat."* versus *"Sit down."*. Paraphrase resources have been published earlier, for instance by Quirk et al. (2004), Dolan et al. (2004), Dolan and Brockett (2005), Ganitkevitch et al. (2013), Ganitkevitch and Callison-Burch (2014), and Pavlick et al. (2015). However, Opusparcus has a few distinctive characteristics.

Firstly, and most importantly, all paraphrases in Opusparcus (**Op**en**S**u**b**titles**Par**aphrase**C**orp**us**) consist of movie and TV subtitles extracted from the *OpenSubtitles2016* collection of parallel corpora (Lison and Tiedemann, 2016). Previous paraphrase collections mostly contain fairly formal language in the form of news text and transcripts of parliamentary proceedings. The more *colloquial* language used in subtitles can be a valuable addition, for instance, in computer assisted language learning, to help learners find natural and idiomatic expressions in real-life situations.

Secondly, in this work the *pivot* language technique introduced by Bannard and Callison-Burch (2005) is applied using multiple pivot languages rather than just one or a few. The technique consists in finding paraphrases in one target language by translating to another, so-called pivot language and then translating back. For example, English *"Have a seat."* can be translated to French *"Asseyez-vous."*, which can be translated back to *"Sit down."*. Now, a well known fact is that different languages make different distinctions; for instance, the English pronoun *you* corresponds to French *toi* or *vous*, depending on number and degree of politeness. If French paraphrases are extracted using English as a pivot, then the *toi/vous* distinction will typically disappear, such that *"Asseyez-vous."* and *"Assieds-toi."* emerge as paraphrases, because they can both be translated as *"Sit down."*. Whether this is desirable or not depends on the application. However, if multiple pivot languages are used rather than one, more distinctions can be preserved. Bannard and Callison-Burch (2005) use four pivot languages in order to identify English paraphrases. Denkowski and Lavie (2010) use one, two, or three pivot languages for their five target languages. Ganitkevitch and Callison-Burch (2014) produce paraphrases for an impressive number of 21 languages, but they limit themselves to using one language, English, as their pivot (in order to be able to use syntactic information, which is available only for English). Opusparcus contains paraphrases in six European languages representing four different language branches: German, English, Swedish (Germanic), French (Romance), Russian (Slavic), and Finnish (Finnic). For each of the six languages, all other five languages are used as pivots.

Thirdly, simplicity is reflected in several aspects of the work. On one hand, only full sentences, so called sentential paraphrases, are produced, unlike Ganitkevitch et al. (2013), Ganitkevitch and Callison-Burch (2014), and Pavlick et al. (2015), who also extract sub-sentential paraphrases, such as individual word pairs, and include the counts of all such fragments in their reported figures. On the other hand, typically subtitles are fairly short, which makes it easier to evaluate and annotate the paraphrase candidates, unlike the complex sentences in the news data of Dolan and Brockett (2005). Furthermore, sub-sentential features or syntactic constraints (Callison-Burch, 2008) are not utilized to assess the likelihood that two sentences are paraphrases. If one favors similar sentence structures, there is a risk to miss some interesting idiomatic variation, such as in *"It's what we do."* $\leftrightarrow$ *"This is our job."*. Finally, particular to this work is that paraphrases and scores for ranking paraphrases are *symmetric*. The two phrases are equal, for instance in contrast to the incorporation of fine-grained entailment relations (Pavlick et al., 2015; Bowman et al., 2015) and the asymmetric conditional probabilities used by Bannard and Callison-Burch (2005).

The rest of this article is split into two main blocks, followed by some concluding remarks. The data sets and annotation scheme are described in Section 2. Alternative ranking functions that can be utilized to produce large paraphrase corpora are evaluated in Section 3.

Category	Description	Examples
Good *"Green"*	The two sentences can be used in the same situation and essentially "mean the same thing".	*It was a last minute thing.* ↔ *This wasn't planned.* *Honey, look.* ↔ *Um, honey, listen.* *I have goose flesh.* ↔ *The hair's standing up on my arms.*
Mostly good *"Light green"*	It is acceptable to think that the two sentences refer to the same thing, although one sentence might be more specific than the other one, or there are differences in style, such as polite form versus familiar form.	*Hang that up.* ↔ *Hang up the phone.* *Go to your bedroom.* ↔ *Just go to sleep.* *Next man, move it.* ↔ *Next, please.* *Calvin, now what?* ↔ *What are we doing?* *Good job.* ↔ *Right, good game, good game.*
Mostly bad *"Yellow"*	There is some connection between the sentences that explains why they occur together, but one would not really consider them to mean the same thing.	*Another one?* ↔ *Partner again?* *Did you ask him?* ↔ *Have you asked her?* *Hello, operator?* ↔ *Yes, operator, I'm trying to get to the police.*
Bad *"Red"*	There is no obvious connection. The sentences mean different things.	*She's over there.* ↔ *Take me to him.* *All the cons.* ↔ *Nice and comfy.*

Table 1: The four annotation categories used, with examples. Each category is also associated with a color, which corresponds to the color of a button in the user interface of the annotation tool.

2. Data Sets and Annotation Scheme

OpenSubtitles2016 (Lison and Tiedemann, 2016) is a collection of translated movie and TV subtitles from `www.opensubtitles.org`. OpenSubtitles2016, which is a subset of the larger OPUS collection[1], provides a large number of sentence-aligned parallel corpora in 65 languages. When subtitles exist for the same film in multiple languages, then sentence alignments are available for each language pair. For the present work, fifteen such bitexts were used, that is, all language-pair combinations for the six target languages German, English, Finnish, French, Russian and Swedish.

In principle, we work on full sentences only, and thus the terms *sentence* and *phrase* are used fairly interchangeably in this paper. Only one-to-one aligned sentences are used, that is, one sentence in the target language must be aligned with one sentence in the pivot language. There are occasional OCR errors and incorrect sentence segmentation in the data.

For each language, the data have been partitioned into separate training, development, and test sets, based on the release year of the movie; the test sets were extracted from years ending in 4, development sets from years ending in 5, and training sets from the rest.

Four different categories have been used when annotating sentence pairs. The annotation scheme is illustrated in Table 1. Other annotation schemes exist as well, such as slightly more complex, five-level Likert scales (Callison-Burch, 2008).

2.1. Training Sets

The so-called training sets are orders of magnitudes larger than the development and test sets and consist of lists of automatically ranked sentence pairs, where a high rank means a higher probability that the two sentences are paraphrases.

[1]OPUS ("... the open parallel corpus"): `opus.lingfil.uu.se`

The training sets, or subsets of them, are intended to be used freely for any useful purpose.

The training sets were produced as follows: First, around 1000 randomly selected sentence pairs were annotated by the author for each of the six languages. Then, an automatic ranking function was applied to all sentence pairs (annotated and unannotated alike), as explained later in Section 3. By extrapolating from the manually annotated data points to the entire set, an estimate of the quality of the training sets can be obtained. The result for English is shown in Figure 1.

Another view to the quality of the training sets is provided in Table 2, where approximate corpus sizes are given for each of the six languages, at three different accuracy levels.

Language	95%	90%	75%
German (de)	590,000	1,000,000	4,700,000
English (en)	1,000,000	1,500,000	7,000,000
Finnish (fi)	480,000	640,000	3,000,000
French (fr)	940,000	2,400,000	11,000,000
Russian (ru)	150,000	170,000	3,400,000
Swedish (sv)	240,000	600,000	1,400,000

Table 2: Number of phrase pairs in the training sets at three different cut-off points, where 95%, 90%, and 75% of the sentence pairs are estimated to be "Good" or "Mostly good" paraphrases.

2.2. Development and Test Sets

Whereas the training sets have been produced semi-automatically, the development and test sets consist exclusively of sentence pairs that have been annotated manually. This is to guarantee the high quality of these sets. However, quality comes at the expense of quantity, so the development and test sets are smaller than the training sets. The number of annotations produced for each language are shown i Table 3. Half of the sentence pairs belong to the

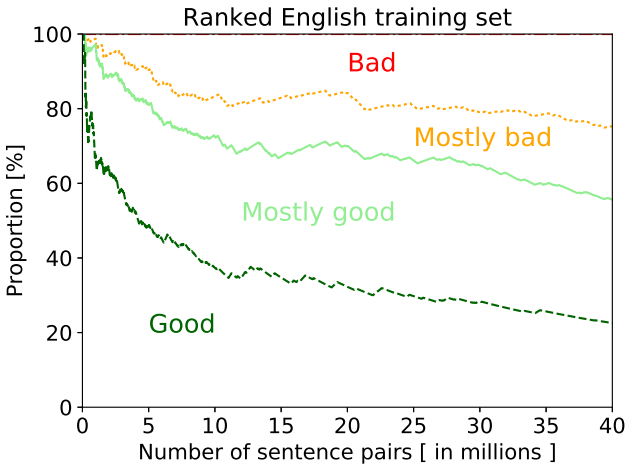


Figure 1: Estimated quality of the English training set. Proportions of the four annotation categories are calculated cumulatively, starting from the most highly ranked sentence pairs (on the left). By picking only highly ranked sentence pairs, one can achieve a high share of "Good" and "Mostly good" parapahrase candidates. The more data that is included, the more "Bad" or "Mostly bad" sentence pairs appear in the set.

development set and the other half to the test set.

Language	Total number of annotations	"Good" or "Mostly good" paraphrases
German (de)	3483	2060
English (en)	3088	1997
Finnish (fi)	3703	1921
French (fr)	3847	2004
Russian (ru)	4381	2088
Swedish (sv)	4178	1931
Total	22,680	12,001

Table 3: Total number of manual annotations in the development and test sets combined. Each sentence pair has been annotated independently by two annotators. For each language, approximately 2000 annotated sentence pairs qualify as acceptable paraphrases.

The development sets can be used to refine whatever training algorithms one might want to devise, and the test sets should be used in final evaluations only. The development sets contain only sentence pairs that do not occur in the training sets. The test sets consist of sentence pairs that do not occur in either the training sets or the development sets. The sentence pairs to be annotated manually were subject to more rigorous pre-filtering than the sentence pairs in the training sets. In the data, there are many sentences that differ only slightly from each other, such as: *"He is not your friend."* ↔ *"He isn't your friend."*. It would have been a waste of human labor to have such simple and predictable variations annotated manually. Therefore, only pairs of sentences that differ sufficiently from each other are accepted into the development and test sets. The difference is measured using relative edit distance; in general, the edit distance between the two sentences has to be at least 0.4 times the length of the shorter of the sentences (and for very short sentences containing less than 24 characters, the distance threshold is even higher).

Two persons annotated every sentence pair. If the annotators agreed on the category, the annotation was accepted as is. If the annotators disagreed but picked adjacent categories (such as "Good" versus "Mostly good" or "Mostly good" versus "Mostly bad"), then the annotation was also accepted, but the lower category was assigned (such that "Mostly good" and "Mostly bad" yields "Mostly bad"). If there was stronger disagreement between the annotators (such as "Mostly good" versus "Bad"), then the sentence pair was discarded. The annotators were also able to discard a sentence pair, if the language of either sentence was wrong or there were spelling or grammar errors. The number of trashed sentences turned out to be highest for French and Russian: It appears that French orthography is complex and mistakes are fairly common in written text. In the Russian data, some non-Russian Cyrillic as well as Latin characters show up occasionally, apparently because of inaccurate optical character recognition (OCR).

The detailed outcome of the annotation effort is summarized in Table 4 for the development sets and Table 5 for the test sets.

3. Automatic Ranking of Paraphrase Candidates

For the data sets that are intended to be used as training sets, a number of ranking schemes have been tested in order to identify paraphrases. Five of the ranking schemes are presented below, followed by a description how these approaches were evaluated. In the examples, English is used as our target language, and we are looking for English paraphrases. In the actual experiments, English was just one of the languages, and the same procedure was carried out for German, Finnish, French, Russian, and Swedish, as well.

3.1. Conditional Probability

Bannard and Callison-Burch (2005) propose a conditional paraphrase probability $P(e_2|e_1)$ as the probability that the English phrase e_1 is translated to a foreign phrase f_i, which in turn is translated back into another English phrase e_2. Since there are typically multiple possible foreign translations, we need to marginalize over the different possible f_i:

$$P(e_2|e_1) = \sum_i P(e_2|f_i)P(f_i|e_1) \quad (1)$$

This ranking formula tends to assign high ranks to phrase pairs, where e_1 is more specific than e_2. For instance, consider the case, where e_1 is *"I was taken from my family when I was a boy."* and e_2 is *"I was taken from my family."*. In the English-French parallel corpus, both English phrases have been aligned with the French phrase f_1: *"On m'a enlevé à ma famille."*. However, e_1 occurs aligned against f_1 only once, whereas e_2 21 times. Thus, $P(f_1|e_1)$ is high (=1), because e_1 is always translated as f_1. Also, $P(e_2|f_1)$ is high (= 21/22), because f_1 is almost always translated as e_2.

	Good	Mostly good		Mostly bad		Bad		Discarded	
	2 x Green	Gr. + Light gr.	2 x Light green	Light gr. + Yel.	2 x Yellow	Yellow + Red	2 x Red	Trash	Disagree
de	286 (16.4%)	333 (19.1%)	394 (22.6%)	189 (10.9%)	112 (6.4%)	100 (5.7%)	168 (9.6%)	81 (4.7%)	79 (4.5%)
en	409 (26.5%)	319 (20.7%)	287 (18.6%)	105 (6.8%)	74 (4.8%)	48 (3.1%)	213 (13.8%)	61 (4.0%)	28 (1.8%)
fi	351 (19.0%)	268 (14.5%)	344 (18.6%)	185 (10.0%)	135 (7.3%)	135 (7.3%)	342 (18.5%)	36 (1.9%)	56 (3.0%)
fr	252 (13.1%)	337 (17.5%)	408 (21.2%)	226 (11.7%)	207 (10.8%)	94 (4.9%)	106 (5.5%)	229 (11.9%)	65 (3.4%)
ru	473 (21.6%)	337 (15.4%)	210 (9.6%)	256 (11.7%)	155 (7.1%)	221 (10.1%)	202 (9.2%)	185 (8.4%)	152 (6.9%)
sv	376 (18.0%)	303 (14.5%)	305 (14.6%)	155 (7.4%)	86 (4.1%)	161 (7.7%)	501 (24.0%)	105 (5.0%)	97 (4.6%)

Table 4: Detailed breakdown of the results of the annotation of the development sets. A sentence pair qualifies as a "good" paraphrase, when both annotators have chosen the "good" category, visualized as a green button in the annotation tool. A sentence pair qualifies as "mostly good", when either one annotator has pushed the green button and the other annotator has pushed the light green button or both annotators have chosen the light green button. Similarly, sentence pairs have been categorized as "mostly bad" or "bad", if both annotators have agreed on the same category or if the annotators ended up pushing adjacent buttons. Sentence pairs were discarded in the following scenarios: The pair was trashed, if at least one of the annotators judged it to contain incorrect spelling or grammar. The sentence pair was also discarded, if the annotators disagreed about the category by more than one step on the four-level scale.

	Good	Mostly good		Mostly bad		Bad		Discarded	
	2 x Green	Gr. + Light gr.	2 x Light green	Light gr. + Yel.	2 x Yellow	Yellow + Red	2 x Red	Trash	Disagree
de	303 (17.4%)	333 (19.1%)	411 (23.6%)	177 (10.2%)	116 (6.7%)	85 (4.9%)	161 (9.2%)	77 (4.4%)	78 (4.5%)
en	450 (29.1%)	273 (17.7%)	259 (16.8%)	97 (6.3%)	56 (3.6%)	64 (4.1%)	246 (15.9%)	59 (3.8%)	40 (2.6%)
fi	376 (20.3%)	244 (13.2%)	338 (18.3%)	179 (9.7%)	138 (7.5%)	121 (6.5%)	353 (19.1%)	60 (3.2%)	42 (2.3%)
fr	261 (13.6%)	337 (17.5%)	409 (21.3%)	206 (10.7%)	204 (10.6%)	124 (6.4%)	133 (6.9%)	184 (9.6%)	65 (3.4%)
ru	462 (21.1%)	351 (16.0%)	255 (11.6%)	223 (10.2%)	151 (6.9%)	225 (10.3%)	188 (8.6%)	189 (8.6%)	146 (6.7%)
sv	379 (18.1%)	312 (14.9%)	256 (12.3%)	173 (8.3%)	107 (5.1%)	147 (7.0%)	527 (25.2%)	120 (5.7%)	68 (3.3%)

Table 5: Detailed breakdown of the results of the annotation of the test sets. Exactly the same procedure was applied as for the development sets. Annotators were unaware of which set a particular sentence pair belonged two; in fact, most annotators were unaware of the existence of separate development and test sets.

This tendency produces numerous errors, when there are occasional misaligned phrases in the corpus, such as in: *"We're staying in the army."* → *"Aah."*, where *"We're staying in the army."* has been aligned against French *"Aah."* once, which in turn has been aligned with English *"Aah."* 1401 times.

3.2. Joint Probability

Instead of a conditional probability, which is asymmetric, one can use the corresponding joint probability, which includes a prior probability, and is symmetric. Thus, the probability of e_1 being a paraphrase of e_2 is the same as the probability of e_2 being a paraphrase of e_1:

$$P(e_1, e_2) = P(e_2|e_1)P(e_1) = P(e_1|e_2)P(e_2) \quad (2)$$

$P(e_2|e_1)$ and $P(e_1|e_2)$ are calculated as in Equation (1), and $P(e_1)$ and $P(e_2)$ are the (prior) probabilities of the phrases, which are simply estimated as relative frequencies over all sentences in the corpus.

Now, at the top of the ranking, we find pairs consisting of frequently used phrases: *"Yes."* ↔ *"Yeah."*, *"Of course."* ↔ *"Sure."*, *"Hello."* ↔ *"Good morning."*, *"Are you okay?"* ↔ *"Are you all right?"*.

A few spurious phrase pairs also score high, where it appears that two frequent phrases might have found a common translation mostly by chance, by the fact that they occur frequently in general: *"You 're welcome."* ↔ *"Sure."*, *"Yeah."* ↔ *"I am."*, *"Hi."* ↔ *"Goodbye."*, *"I do."* ↔ *"I know."*

3.3. Pointwise Mutual Information

Pointwise Mutual Information (PMI) divides the joint probability by the probability that the two phrases $e_1$and e_2 occur independently. Thus, PMI penalizes phrase pairs that co-occur mostly by chance, by the fact that they occur frequently in general:

$$\begin{aligned} \text{pmi}(e_1; e_2) &= \log \frac{P(e_1, e_2)}{P(e_1)P(e_2)} \\ &= \log \frac{P(e_2|e_1)}{P(e_2)} = \log \frac{P(e_1|e_2)}{P(e_1)} \end{aligned} \quad (3)$$

This scoring favors phrase pairs e_1 and e_2 that have a limited set of translations f_i, such that e_1 and e_2 are not aligned with phrases other than f_i, and f_i are not aligned with other phrases than e_1 and e_2. For instance, the phrases *"You sound a little homesick."* ↔ *"Do you miss being home?"* have a common French translation *"Vous avez le mal du pays ?"*, which occurs twice in the corpus, aligned once against each of the two English phrases.

However, similarly to the conditional probability in Equation (1), PMI is sensitive to misaligned, infrequent sentences. The phrase pair *"Lost the phone now."* ↔ *"I'm from the agency."* scores high because *"Lost the phone now."*

has been misaligned against French *"Je viens de l'agence."*, which occurs only twice.

3.4. Joint Probability and PMI Combined

Our experiments show that rather than using joint probability (2) or PMI (3) in isolation, we obtain a better ranking by multiplying the two together:

$$P(e_1, e_2) \cdot \text{pmi}(e_1; e_2) = P(e_1, e_2) \log \frac{P(e_1, e_2)}{P(e_1)P(e_2)} \quad (4)$$

This leverages the strengths and alleviates the shortcomings of the two approaches.

3.5. Multiple Multilingual Parallel Corpora

The four formulae presented above, (1), (2), (3), and (4), easily generalize beyond bilingual parallel corpora. One can simply concatenate all parallel corpora, such that English is kept on one side and the other (pivot) languages on the other side. All frequencies and probabilities are then calculated over the merged bitext as a whole.

Interestingly, another approach can produce better results. PMI in Equation (3) may rank rare phrase pairs very high, and in case of misalignments, these pairs are unreliable. However, if a rare phrase pair ranks high in multiple bitexts, then this seems to signal much higher confidence. In order not to lose the information that a phrase pair emerges in multiple different corpora, rather than merging the parallel corpora into one, we can keep them separate. We then compute PMI scores separately for each bitext (English-German, English-Finnish, English-French, English-Russian, and English-Swedish). To obtain a combined score, we compute the sum of the PMI values obtained from each different corpus (English vs. pivot language $\mathcal{L}_i$):

$$\sum_i \text{pmi}(e_1; e_2|\mathcal{L}_i) = \sum_i \log \frac{P(e_1, e_2|\mathcal{L}_i)}{P(e_1|\mathcal{L}_i)P(e_2|\mathcal{L}_i)} \quad (5)$$

The probabilities are calculated exactly as previously. The notation merely highlights the fact that every value is conditioned on alignments between English and a specific pivot language $\mathcal{L}_i$.

Since the number of languages is constant, the sum in (5) can also be interpreted as the average PMI across pivot languages.

3.6. Evaluation of Ranking Schemes

A symmetric score is desired in this work, and therefore the conditional probability in Equation (1) cannot be used as such. However, one could obtain a symmetric score by combining $P(e_2|e_1)$ and $P(e_1|e_2)$ in some way, such as taking the minimum, maximum or average value. In practice, this would make this score behave fairly similarly as the more elegantly formulated PMI in Equation (3), so the conditional probability scheme was not investigated further. This leaves us with the four remaining schemes in Equations (2), (3), (4), and (5). They were compared with the help of a set of phrase pairs that were drawn randomly from the training set and annotated manually, as described in Section 2.1. The training set is then reordered using the ranking scheme to be tested. Depending on the ranking scheme, the manually annotated phrase pairs will appear at different ranks in the full, ordered collection. An ideal ranking scheme will place the phrase pairs that are true paraphrases at the head of the ordering and the phrase pairs that are not paraphrases at the tail.

The results were then plotted as in Figure 1 and compared visually. Across the six languages, the results were consistent: the best performing rankings were PMI summed over multiple corpora (5) followed by joint probability mutliplied by PMI (4). The types of phrase pairs that rank high are different in both cases: the former favors less frequent, more specific phrase pairs, such as *"It was a difficult and long delivery."* ↔ *"The delivery was difficult and long."*, whereas the latter favors frequent, less informative phrase pairs, such as: *"Excuse me."* ↔ *"I'm sorry."*. PMI summed over multiple corpora, in Equation (5), was judged to be the best ranking function. The final training sets were produced using this particular ranking.

4. Conclusion

Paraphrase extraction from movie subtitle data has been described in this paper. Six languages were included in this initial phase, but there is no principal reason why not more of the 65 languages in the OpenSubtitles2016 collection could be exploited. As there is considerable manual annotation effort involved, crowdsourcing could be considered; see, for instance, Tschirsich and Hintz (2013).

Another improvement could be to reduce the number of OCR errors that still occur in the data.

Acknowledgments

The following people have participated in the annotation effort: Thomas de Bluts, Aleksandr Semenov, Eetu Sjöblom, Mikko Aulamo, Olivia Engström, Janine Siewert, Carola Carpentier, Svante Creutz, Yves Scherrer, Anders Ahlbäck, Sami Itkonen, Riikka Raatikainen, Kaisla Kajava, Tiina Koho, Oksana Lehtonen, Sharid Loáiciga Sánchez, Tatiana Batanina, and the author himself. The annotation tool was implemented by Mikko Aulamo. The author is grateful to Mikko and the annotators for their very valuable contributions.

The author would also like to thank professor Jörg Tiedemann together with the anonymous reviewers for their valuable comments and suggestions as well as the Academy of Finland for the financial support of the annotations through Project 314062 in the ICT 2023 call on Computation, Machine Learning and Artificial Intelligence.

Furthermore, warm thanks go to Mietta Lennes and the Language Bank of Finland (Kielipankki) for publishing and hosting Opusparcus as part of their corpus collections.

5. Bibliographical References

Bannard, C. and Callison-Burch, C. (2005). Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 597–604, Ann Arbor, Michigan. Association for Computational Linguistics.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Callison-Burch, C. (2008). Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 196–205, Honolulu, Hawaii. Association for Computational Linguistics.

Denkowski, M. and Lavie, A. (2010). METEOR-NEXT and the METEOR paraphrase tables: Improved evaluation support for five target languages. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR, WMT '10*, pages 339–342, 08.

Dolan, B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*. Asia Federation of Natural Language Processing, January.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Geneva, Switzerland. Association for Computational Linguistics.

Ganitkevitch, J. and Callison-Burch, C. (2014). The multilingual paraphrase database. In *The 9th edition of the Language Resources and Evaluation Conference*, Reykjavik, Iceland, May. European Language Resources Association.

Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2013). PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764, Atlanta, Georgia, June. Association for Computational Linguistics.

Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*.

Pavlick, E., Rastogi, P., Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2015). PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pages 425–430, Beijing, China, July. Association for Computational Linguistics.

Quirk, C., Brockett, C., and B. Dolan, W. (2004). Monolingual machine translation for paraphrase generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP2004)*, pages 142–149, Barcelona, Spain, January.

Tschirsich, M. and Hintz, G. (2013). Leveraging crowdsourcing for paraphrase recognition. In *Proceedings of the 7th Linguistic Annotation Workshop & Interoperability with Discourse @ ACL 2013*, pages 205–213, Sofia, Bulgaria, August.

6. Language Resource References

Creutz, M. (2018). *Opusparcus: Open Subtitles Paraphrase Corpus for Six Languages,* version 1.0. Language Bank of Finland (Kielipankki), `http://urn.fi/urn:nbn:fi:lb-2018021221`.

Fine-grained Semantic Textual Similarity for Serbian

Vuk Batanović, Miloš Cvetanović, Boško Nikolić
School of Electrical Engineering, University of Belgrade
Bulevar kralja Aleksandra 73, 11120 Belgrade, Serbia
vuk.batanovic@student.etf.bg.ac.rs, cmilos@etf.bg.ac.rs, nbosko@etf.bg.ac.rs

Abstract

Although the task of semantic textual similarity (STS) has gained in prominence in the last few years, annotated STS datasets for model training and evaluation, particularly those with fine-grained similarity scores, remain scarce for languages other than English, and practically non-existent for minor ones. In this paper, we present the Serbian Semantic Textual Similarity News Corpus (*STS.news.sr*) – an STS dataset for Serbian that contains 1192 sentence pairs annotated with fine-grained semantic similarity scores. We describe the process of its creation and annotation, and we analyze and compare our corpus with the existing news-based STS datasets in English and other major languages. Several existing STS models are evaluated on the Serbian STS News Corpus, and a new supervised bag-of-words model that combines part-of-speech weighting with term frequency weighting is proposed and shown to outperform similar methods. Since Serbian is a morphologically rich language, the effect of various morphological normalization tools on STS model performances is considered as well. The Serbian STS News Corpus, the annotation tool and guidelines used in its creation, and the STS model framework used in the evaluation are all made publicly available.

Keywords: short-text semantic similarity, corpus annotation, morphological normalization

1. Introduction

Semantic Textual Similarity (STS), sometimes also referred to as Short-text Semantic Similarity (STSS), is the task of assigning a numerical score to a given pair of short texts based on the level of semantic equivalence between them. The minimal numerical score in a given range indicates complete semantic independence, while the maximal score indicates full semantic equality. Although STS has important implications for a whole range of other natural language processing tasks, including information retrieval, question answering, machine translation, textual entailment, etc., research on this topic started appearing only around a decade ago (Corley and Mihalcea, 2005; Mihalcea, Corley, and Strapparava, 2006; Islam and Inkpen, 2008). Semantic Textual Similarity has gained in prominence since 2012, with its inclusion in the annual *SemEval* shared tasks (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017).

A large collection of datasets with fine-grained semantic similarity scores has been annotated in this series of shared tasks, using a standardized methodology, and has been made publicly available. The move from binary similarity scores (Dolan and Brockett, 2005) to fine-grained ones has allowed for more precise model training and evaluation. However, most of this development has been limited to English. Several other major languages have been considered recently, including Spanish (Agirre et al., 2014, 2015; Cer et al., 2017), French (Vu et al., 2014), Portuguese (Fonseca et al., 2016), Chinese and Japanese (Hayashi and Luo, 2016), Arabic (Cer et al., 2017), and Hindi (Agarwal et al., 2017). Among them, only the datasets in Spanish, Portuguese, and Arabic have been made publicly available. To the best of our knowledge, there has been no development of STS datasets with fine-grained similarity scores for minor languages so far.

In this paper, we present the Serbian Semantic Textual Similarity News Corpus (*STS.news.sr*)[1] – a publicly available STS dataset for Serbian annotated with fine-grained semantic similarity scores. Although there has been some recent work on the broader task of semantic relatedness in Polish (Wróblewska and Krasnowska-Kieraś, 2017), our dataset is, as far as we know, the first STS dataset for a Slavic language.

The remainder of this paper is structured as follows: in Section 2 we describe the creation and annotation of *STS.news.sr*, while in Section 3 we analyze and compare it with the available STS datasets in other languages. Section 4 provides some baseline model results on *STS.news.sr*, as well as an evaluation of several supervised bag-of-words STS models. Within this section, we also assess the impact of morphological normalization methods for Serbian – a language with rich morphology – on STS models. Finally, in Section 5 we present our conclusions and some potential avenues of future research.

2. Dataset Creation and Annotation

The initial step in STS dataset creation is the acquisition of a suitable collection of short-text pairs. We deemed the existing Serbian Paraphrase Corpus (*paraphrase.sr*)[2] (Batanović, Furlan, and Nikolić, 2011; Furlan, Batanović, and Nikolić, 2013), a set of 1194 sentence pairs gathered from the news domain, to be a suitable source for this purpose. Firstly, we went through the corpus and manually corrected any typographical errors and restored any missing diacritical marks. Two sentence pairs were removed from the dataset since one was found to be a duplicate and the other included a text longer than one sentence. The remaining 1192 sentence pairs were then given to five annotators who independently assigned a semantic similarity score to each pair.

For the sake of standardization, we chose to follow the annotation methodology established in the *SemEval* STS tasks, and we adopted the scoring scheme (a 0 – 5 Likert scale) and the general annotation guidelines used therein (Agirre et al., 2013). However, our initial consultations with the annotators showed that the sentence pair examples for each score that are included in the *SemEval* annotation instructions can be somewhat unclear, particularly those for scores 2 – 4. This issue had an effect on lowering task comprehension and annotation quality. To rectify this, we replaced all examples with new ones, and we increased the

[1] http://vukbatanovic.github.io/STS.news.sr/

[2] http://vukbatanovic.github.io/paraphrase.sr/

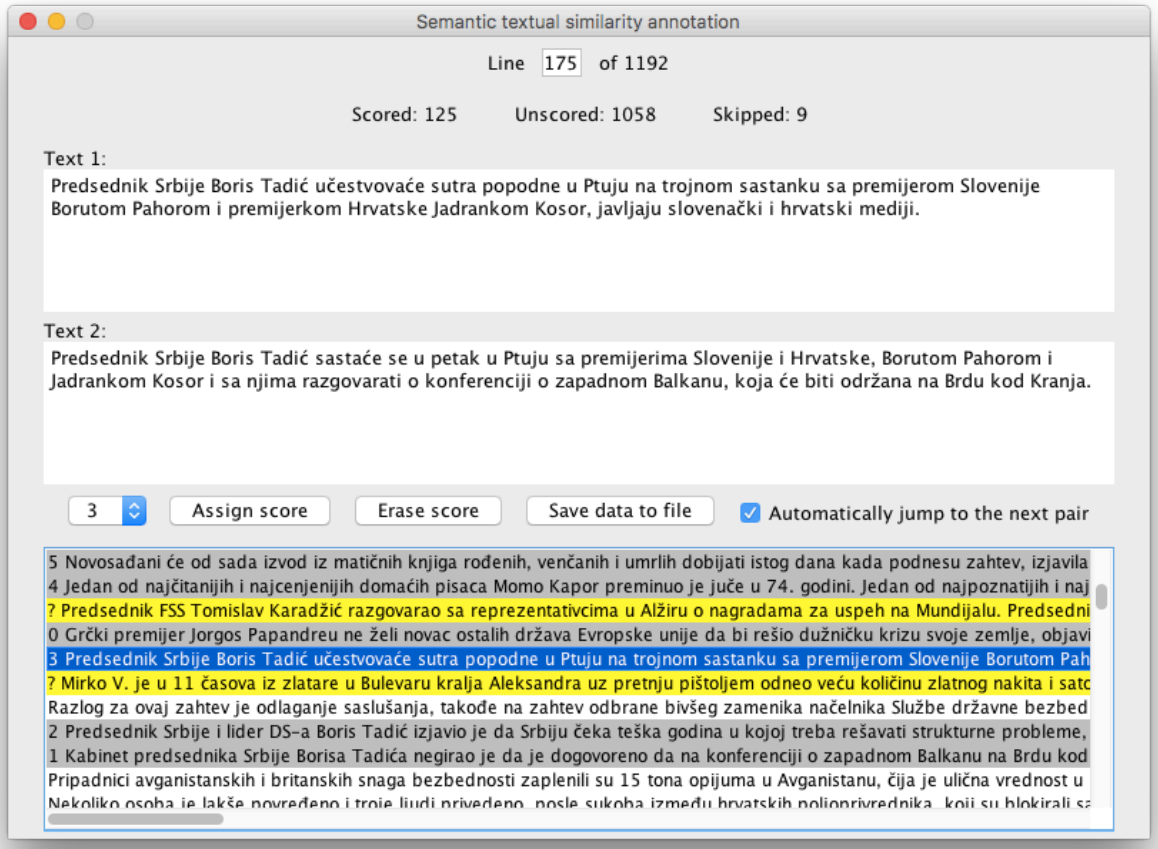


Figure 1: The *STSAnno* annotation tool interface

number of examples from one to three per score. In order to limit our own bias in the selection of new examples, we chose suitable pairs from the 2012 *MSRPar* and the 2013-2016 *HDL SemEval* STS corpora in English (since they all belong to the news domain), and we had them professionally translated into Serbian. We considered only those pairs whose averaged scores are integers – an integer average usually means that all annotators assigned the same similarity score to a particular pair, indicating its unambiguity. The final selection was made in consultation with the annotators to ensure the representativeness of each example. Our annotation guidelines and examples, in both Serbian and English, are available on the *STS.news.sr* repository.

Once the instructions were finalized, all annotators first scored a subset of 60 randomly selected pairs from the corpus (~5% of the total), before proceeding to annotate the entire dataset. This initial batch was subsequently used to calculate the annotator self-agreement scores. The annotation process was completed within approximately two months.

In order to make the annotation quicker and easier, we created *STSAnno*[3], a simple offline annotation tool. *STSAnno* allows an annotator to view in parallel the texts in a pair, assign a semantic similarity score to them, and change or erase existing scores. Annotators can also assign a special symbol to a pair to temporarily skip it, which can be useful when faced with difficult examples. Scored, unscored, and skipped pairs are highlighted in different colors to easily distinguish between them. Sentence pairs can be scored in the order in which they are given, or in any other order chosen by the annotator. At all times, a statistical overview of the annotation progress is displayed. A screenshot of *STSAnno* during the annotation of *STS.news.sr* is shown in Figure 1.

3. Dataset Analysis

The annotator self-agreements and the inter-annotator agreements were calculated using the Pearson correlation coefficient *r*. Table 1 contains the self-agreement scores and Table 2 the inter-annotator agreements. In addition to the pairwise inter-annotator scores, we also measure the agreement of each annotator with the average of the scores of all other annotators.

The agreement scores are generally very high. Even though the annotators had different backgrounds (annotator #1 is a computational linguist, annotators #2 and #3 are linguists, while annotators #4 and #5 are non-linguists) there is no major difference in correlation values due to this. This indicates that with well-chosen example pairs and clear guidelines, even non-experts can achieve very high levels of annotation quality on STS corpora. Our average inter-rater agreement between an annotator and the average of the scores of all other annotators is 0.92, which is therefore

[3] http://vukbatanovic.github.io/STSAnno/

Annotator	#1	#2	#3	#4	#5	Average
Self-agreement	0.95	0.97	0.97	0.85	0.92	0.93

Table 1: Annotator self-agreement scores

Annotator	#1	#2	#3	#4	#5	Average
#1	/					
#2	0.90	/				
#3	0.89	0.87	/			
#4	0.88	0.84	0.85	/		
#5	0.90	0.88	0.86	0.86	/	
Average of other annotators	0.94	0.92	0.91	0.90	0.92	0.92

Table 2: Inter-annotator agreement scores

the upper bound for STS model performance on this dataset. This agreement is higher than the ones reported for *SemEval* datasets from the news domain (Agirre et al., 2013, 2014, 2015) by around 0.05 – 0.1, most likely due to the increased number and quality of the examples in our annotation instructions.

The final similarity score for each sentence pair was obtained by averaging the scores of all five annotators. Figure 2 shows the distribution of sentence pairs within the Serbian STS News Corpus across the range of similarity score values. It is moderately balanced, with the exception of a large peak regarding the pairs with the score 3.0. However, similar distributional irregularities are also present in other news-based STS datasets.

A comparison between our dataset and other publicly available STS corpora created from the news domain is shown in Table 3. We consider the following corpora:

- In English: the 2012 *SemEval MSRPar* corpus (Agirre et al., 2012), the combined 2013-2016 collection of *SemEval HDL* corpora (Agirre et al., 2013, 2014, 2015, 2016), and the 2014 *SemEval Deft-news* corpus (Agirre et al., 2014).
- In Spanish: the combined 2014-2015 *SemEval News* corpora (Agirre et al., 2014, 2015).
- In Portuguese: the 2016 ASSIN corpus, divided into European and Brazilian Portuguese portions (Fonseca et al., 2016).
- In Arabic: the 2017 *SemEval* translation of a part of the *MSRPar* corpus into Arabic (Cer et al., 2017).

The size of the Serbian STS News Corpus is average when compared to the other available STS corpora, both in terms of the number of sentence pairs and in terms of token count (we counted only alphanumerical tokens). The average length of a sentence in *STS.news.sr* is greater than in most other STS datasets, while the average similarity score is 2.51 – almost ideal given the 0 – 5 score scale. In fact, *STS.news.sr* is much more balanced than the English *SemEval MSRPar* corpus, which is the one most similar to it in terms of source material, type, and size.

However, nearly all of the considered STS corpora exhibit strong distribution peaks around score values 3 and 4, in case of the 0 – 5 score scale, and scores 2 and 3 in case of the 0 – 4 scale. The ASSIN corpora score distribution is heavily skewed toward the central 2 – 4 values. The only corpus with a more uniform score distribution is the English *SemEval HDL*. This is probably at least in part a natural effect of the shortness of the texts (news headlines) in this corpus. With longer texts, the likelihood of coming across sentence pairs with near-identical semantics (score values close to 5 on a 0 – 5 scale) is lower. Similarly, in longer text pairs, there is a greater chance of encountering at least some semantic links between the sentences, lowering the probability of minimally scored pairs. We leave for further work the consideration of how and to what extent these distribution irregularities in most corpora affect the training and evaluation of STS models.

4. Evaluation

We evaluate several STS models on the Serbian STS News Corpus. As a performance metric, we utilize the Pearson correlation coefficient between the model output and the averaged annotated similarity scores, which we consider the gold standard. We first consider unsupervised models, and evaluate them on the entire *STS.news.sr*. Then, we move on to supervised algorithms, which are evaluated using 10-fold cross-validation with sorted stratification.

4.1 Unsupervised Models

The first unsupervised model we consider is the one used as a standard baseline in all *SemEval* STS shared tasks – a simple word overlap technique in which sentences are split into tokens using white space and then represented as bag-of-words vectors in the multidimensional token space (Agirre et al., 2012). Token counts in a sentence are binarized, so that each vector dimension has a value of one if that token appeared in the sentence, and zero otherwise. Cosine similarity is used to compute the similarity between such sentence vectors. We also improve upon this baseline by lowercasing the text, removing punctuation, and using the tokenizer for Serbian included in the ReLDI (*Regional Linguistic Data Initiative*) project repository[4] (Samardžić, Ljubešić, and Miličević, 2015; Ljubešić, Erjavec, et al., 2016). Since it proves to be highly beneficial, this improved tokenization approach is utilized for all subsequent models.

The second baseline model that we use is one based on averaging the embeddings of words in a sentence and calculating the cosine similarity of the mean vectors. We employ the w*ord2vec* algorithm (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013), as implemented in the *gensim* library (Řehůřek and Sojka, 2010), since Cer et al. (2017) showed it to be superior to the other word

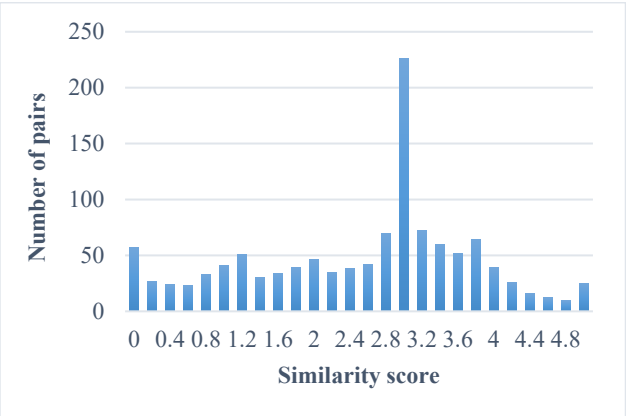


Figure 2: Sentence pair similarity score distribution in the Serbian STS News Corpus

[4] http://reldi.spur.uzh.ch/blog/tokeniser/

Corpus	Lang.	Score scale	Sentence pairs	Tokens	Average sentence length in tokens	Average similarity score	Percentage of sentence pairs with scores rounded to					
							0	1	2	3	4	5
STS.news.sr	SR	0 – 5	1192	64 K	~27	2.51	9.06%	14.93%	16.11%	39.43%	16.53%	3.94%
SemEval MSRPar	EN	0 – 5	1500	54 K	~18	3.30	0.13%	4.47%	13.87%	36.47%	36.2%	8.86%
SemEval HDL	EN	0 – 5	2499	37 K	~7	2.62	10.56%	17.89%	17.29%	19.93%	22.09%	12.24%
SemEval Deft-news	EN	0 – 5	300	9 K	~16	3.03	4.0%	11.0%	16.0%	27.33%	31.33%	10.33%
SemEval News	ES	0 – 4	980	68 K	~35	2.20	6.33%	16.33%	36.94%	29.08%	11.32%	/
ASSIN (PT)	PT-PT	1 – 5	5000	145 K	~14	3.04	/	2.72%	35.14%	24.16%	31.56%	6.42%
ASSIN (BR)	PT-BR	1 – 5	5000	130 K	~13	3.04	/	1.74%	33.94%	28.84%	30.76%	4.72%
SemEval MSRPar	AR	0 – 5	510	18 K	~18	3.36	0.2%	3.33%	13.14%	34.9%	40.39%	8.04%

Table 3: An overview of news-based STS corpora with fine-grained semantic similarity scores

embedding models when used in this context. The skip-gram *word2vec* architecture is trained on the Serbian web corpus *srWaC* (Ljubešić and Klubička, 2014), the largest publicly available text corpus in Serbian, containing 555 million tokens. The *srWaC* corpus is parsed to remove punctuation marks and words that are not in Serbian, and is then lowercased. This reduces the corpus to around 470 million tokens, with a vocabulary of around 3.8 million entries. We use 100-dimensional vectors and a window size of 10 for the skip-gram model. All other parameters are kept at the *gensim* default settings.

In both baseline models, we experiment with a simple negation-marking technique in which a single word after a negation word is marked with a special prefix in order to distinguish it from its non-negated form. Such techniques were previously found useful in simple models for other semantic tasks, such as sentiment analysis, both in English (Pang, Lee, and Vaithyanathan, 2002) and in Serbian (Batanović, Nikolić, and Milosavljević, 2016).

The results of basic unsupervised STS model evaluations are presented in Table 4. We find that the word overlap model outperforms the embedding-based one, indicating a higher level of string similarity between the sentences in *STS.news.sr*, which is to be expected given the method used to collect sentence pairs for the original *paraphrase.sr* corpus (Furlan, Batanović, and Nikolić, 2013). We therefore also consider a joint model in which the *word2vec* mean vector and the binarized bag-of-words vector of a sentence are concatenated and used in cosine similarity calculation. This joint baseline proves superior to both individual models.

Model	Pearson *r*
Word overlap (white space tokenizer)	0.6461
Word overlap (Serbian tokenizer)	0.6869
Word overlap (Serbian tokenizer + negation marking)	0.6862
word2vec averaging (Serbian tokenizer)	0.6211
word2vec averaging (Serbian tokenizer + negation marking)	0.6257
Word overlap + *word2vec* averaging (Serbian tokenizer)	**0.6949**
Word overlap + *word2vec* averaging (Serbian tokenizer + negation marking)	0.6943

Table 4: Unsupervised baseline STS model performances on the entire dataset

The proposed negation-marking technique is found beneficial on the *word2vec* baseline. However, since it has a slightly detrimental effect on the superior word overlap and joint models, we do not use it in further experiments.

4.1.1 Morphological Normalization

Next, having in mind the morphological complexity of Serbian, we evaluate the impact of morphological normalization methods on our baseline STS models. The results are presented in Table 5.

Three stemming algorithms developed for Serbian are considered – the optimal and the greedy algorithm of Kešelj and Šipka (2008), and the improvement of the greedy algorithm by Milošević (2012). We also evaluate a stemmer for Croatian, a language closely related to Serbian, by Ljubešić and Pandžić, which is a refinement of the approach presented in (Ljubešić, Boras, and Kubelka, 2007). We use the *SCStemmers* package (Batanović, Nikolić, and Milosavljević, 2016) in which all of the aforementioned algorithms are implemented.

Similarly, we consider two publicly available lemmatizers for Serbian and one for Croatian. The first lemmatizer for Serbian is *BTagger*, which is available in two variants – one that only normalizes word suffixes (Gesmundo and Samardžić, 2012b), and another that also deals with word prefixes, allowing for full lemmatization (Gesmundo and Samardžić, 2012a). In addition, we assess a lemmatization model for Croatian developed by Agić, Ljubešić, and Merkler (2013) for the CST lemmatizer (Jongejan and Dalianis, 2009). The final lemmatizer that is evaluated is the one for Serbian by Ljubešić, Klubička, et al. (2016), which relies on a large inflectional lexicon and an improved part-of-speech tagger.

Morphological normalizer	Model		
	Word overlap	*word2vec* averaging	Word overlap + *word2vec* averaging
None	0.6869	0.6211	0.6949
Stemmers			
Kešelj and Šipka (optimal)	0.7291	0.5971	0.7338
Kešelj and Šipka (greedy)	0.7218	0.5966	0.7271
Milošević	0.7210	0.5986	0.7266
Ljubešić and Pandžić	0.7287	0.6077	**0.7339**
Lemmatizers			
BTagger (suffix)	0.7031	0.5936	0.7126
BTagger (suffix + prefix)	0.7019	0.5921	0.7112
Agić et al.	0.7064	0.5915	0.7143
Ljubešić et al.	0.7225	0.5937	**0.7283**

Table 5: The effects of morphological normalization methods on unsupervised baseline STS model performances on the entire dataset

The results show that the application of morphological normalization has a consistently positive impact on the performance of word overlap and joint baseline models, and a consistently detrimental one on the purely embedding-based method. On average, stemmers tend to have a better effect on STS models than lemmatizers do. The best overall stemmer is Ljubešić and Pandžić's stemmer for Croatian, although the optimal stemmer of Kešelj and Šipka is a close second. Ljubešić and Pandžić's stemmer was also found to be the best option for sentiment classification in Serbian (Batanović and Nikolić, 2016, 2017), making it a good choice in general. The lemmatizer of Ljubešić et al. proves to be the best one in this setting, but it is still outmatched by the top two stemming algorithms.

4.2 Supervised Models

We limit the examination of supervised models to those that do not require more advanced syntactic tools, like dependency parsers, since the development of such tools for Serbian has only recently begun (Samardžić et al., 2017). We therefore evaluate the performance of several bag-of-words models. The approach proposed by Islam and Inkpen (2008) is the most basic one we consider. Within it, each word from the shorter sentence is paired to its most similar word in the longer sentence, and word pair similarities are calculated as a mixture of three string similarity metrics and one corpus-based semantic similarity measure. Supervision is used in this method to determine the optimal balance between the string and the corpus-based measures in the final score. As the corpus-based measure, we utilize the cosine similarity of the same *word2vec* 100-dimensional vectors as before.

We also evaluate three models derived from this basic approach. The first is *LInSTSS* (*Language-independent Short-text Semantic Similarity*), proposed by Furlan, Batanović, and Nikolić (2013), in which word pair similarities are weighted according to the term frequencies of the words in question. We calculate the TF values using the *srWaC* corpus.

The second one is *POST STSS* (*Part-of-speech Tag-supported Short-text Semantic Similarity*), proposed by Batanović and Bojić (2015), which utilizes similarity weighting based on the part of speech of each word in a pair. In order to obtain POS tags we use the Serbian morphosyntactic tagger developed by Ljubešić, Klubička, et al. (2016). This tagger produces morphosyntactic descriptors (MSDs) and POS tags according to the MULTEXT-East (Erjavec, 2017) version 5 standard for Croatian[5]. In this standard, a POS tag is simply the first letter of an MSD.

POST STSS relies on a two-stage optimization procedure in order to determine the best weighting settings, including the weights for each part of speech/each POS grouping, as well as the values within a POS interaction matrix that allow or disallow the pairing of words belonging to different parts of speech/POS groups. In the first phase of the *POST STSS* parameter optimization – pseudo-exhaustive search – we classify all MSDs into one of the following seven POS groups:

1. Nouns – MSDs start with *N*
2. Verbs – MSDs start with *V*
3. Adverbs – MSDs start with *R*
4. Adjectives – MSDs start with *A*
5. Pronouns – MSDs start with *P*
6. Numerals – MSDs start with *M*
7. Other – all other MSD values

In the second optimization phase – steepest ascent hill climbing – we expand the POS weights from these seven groups into 29 classes. Each class represents a different MSD category/type combination, according to the MULTEXT-East version 5 standard. For instance, the *Noun* MSD category is divided into two types – common nouns (*Nc*) and proper nouns (*Np*), while the *Numeral* category is divided into four types – cardinal numerals (*M_c*), ordinal numerals (*M_o*), multiple numerals (*M_m*), and special numerals (*M_s*). There are also MSD categories, such as adpositions (*S*), which are not divided into types – in these cases one weight is assigned to an entire category. The only exceptions to this classification scheme are the residuals (*X*), where a single weight is assigned to the entire category since only one type of residual (foreign) appears in *STS.news.sr*, and the punctuation category (*Z*), which is ignored, since punctuation is filtered out during tokenization. However, this category/type classification is applied only to those types for which actual MSD values are specified in the MULTEXT-East version 5 standard. For example, the standard allows for a separate type of copular verbs (*Vc*), but no tags are specified under this type and the utilized tagger does not employ it, so this category/type combination does not necessitate a separate weight.

POST STSS requires a nested cross-validation during the first optimization phase in order to tune the model hyperparameters – the initial POS weight values, the initial POS interaction values, the initial string similarity weight, the choice of the POS weighting function, and the option of using a special weight value minimization process at the end of the first optimization phase. Here, for the sake of

[5] http://nl.ijs.si/ME/V5/msd/html/msd-hr.html

efficiency, we only optimize the initial POS weights and the initial POS interaction values in a nested three-fold CV. We consider the same options for their initial values as in (Batanović and Bojić, 2015). For the remaining hyperparameters we use the settings found optimal in previous experiments (Batanović and Bojić, 2015) – the initial string similarity weight is set to 0.5, the arithmetic mean is the chosen POS weighting function, and the value minimization process is not used.

Finally, we propose and evaluate a mixture of *LInSTSS* and *POST STSS* that uses both TF-based and POS-based weighting of word similarities. Within this model, similarities between words i and j are calculated as follows:

$$Sim(i,j) = (w_{str} \times Str(i,j) + w_{sem} \times Sem(i,j)) \times TF(i,j) \times POS(i,j)$$

where w_{str} and w_{sem} are the string and the semantic similarity weights ($w_{str} + w_{sem} = 1$), $Str(i,j)$ is the mixture of three string similarity metrics as defined in (Islam and Inkpen, 2008), and $Sem(i,j)$ is the corpus-based semantic similarity measure (as noted, we use the cosine similarity of *word2vec* vectors in this paper). $TF(i,j)$ is the term frequency weighting function, as defined in (Furlan, Batanović, and Nikolić, 2013), while $POS(i,j)$ is the part-of-speech weighting function (as noted, in this paper we always use the arithmetic mean of the weights for the parts of speech of words i and j). The optimization procedure for this approach, which we name *POS-TF STSS*, is identical to the one used for *POST STSS*, since term frequencies are obtained from the *srWaC* corpus in an unsupervised way.

In all supervised models, weight values are optimized in steps of 0.1. The string similarity weight is chosen from the [0.3, 0.7] range, while the POS weights are optimized in the [0.7, 1.3] range. In order to minimize the chance of overfitting to the training set, the hill climbing part of the *POST/POS-TF STSS* optimization is stopped heuristically, when there are no hill climbing moves left whose error reduction on the training set is at least 5% of the error reduction of the first move made in the climb.

The 10-fold cross-validation results for all models are shown in Table 6. We repeat the unsupervised model evaluation using 10-fold CV to be able to make a fair comparison between the performances of the unsupervised and the supervised models. Furthermore, we measure the impact of morphological normalization on supervised models, but we limit the scope to the tools that were previously found to be the best in each category – the stemmer of Ljubešić and Pandžić, and the lemmatizer of Ljubešić et al. All of the evaluated STS models, both supervised and unsupervised, are made available as parts of *STSFineGrain*[6], a collection of STS models and a unified framework for their evaluation, implemented in Java.

Results show that supervised models perform noticeably better than unsupervised ones on non-normalized text, but the gap between the two narrows when stemming or lemmatization is applied. Stemming has a clearly positive effect on almost all models, while lemmatization only brings an improvement to word overlap methods, with mixed effects on supervised ones. The three models derived from Islam and Inkpen's approach consistently outperform the original algorithm. *LInSTSS* generally achieves results similar to *POST STSS*, but the *POS-TF STSS* mixture model performs better than both *LInSTSS* and *POST STSS* independently. In fact, when used in conjunction with the stemmer of Ljubešić and Pandžić, *POS-TF STSS* achieves the best result among all the models that we considered.

Model	Morphological normalizer		
	None	*Stemmer* Ljubešić and Pandžić	*Lemmatizer* Ljubešić et al.
Unsupervised models			
Word overlap	0.6970	0.7367	0.7278
word2vec averaging	0.6405	0.6295	0.6136
Word overlap + *word2vec* averaging	0.7050	0.7417	0.7335
Supervised models			
Islam and Inkpen	0.7387	0.7444	0.7350
LInSTSS (TF weighting)	0.7534	0.7573	0.7494
POST STSS (POS weighting)	0.7538	0.7593	0.7491
POS-TF STSS (POS and TF weighting)	0.7599	**0.7665**	0.7606

Table 6: STS model performances on 10-fold CV

4.2.1 Optimal Parameters

Naturally, the optimal parameter values for supervised models vary somewhat from one morphological normalization approach to another, but there are consistent patterns that can be observed. The optimal string similarity weight in the basic Islam and Inkpen approach tends to be 0.7, resulting in an optimal semantic similarity weight of 0.3. This is not surprising given the higher level of string similarity between the sentences in *STS.news.sr*. Nevertheless, in the *LInSTSS* model the optimal value of the string similarity weight is a bit lower (0.6) which indicates that the addition of TF weighting increases the importance of non-surface forms of similarity.

Some variation between the optimal POS weight settings of *POST STSS* and those of *POS-TF STSS* does exist. However, we did not encounter any systematic differences between the optimal parameters of these two algorithms, nor between their chosen optimal hyperparameter values. The optimal initial POS weights are most often set to the neutral value of 1.0, while the optimal initial POS interaction setting is usually to allow word pairings between all parts of speech.

Common nouns typically retain a neutral POS weight value of 1.0 and are found to be more important than proper nouns, whose weight revolves around the 0.8 – 0.9 mark. The weight for main verbs is almost universally set to the 1.3 maximum, indicating the central role of a verb in conveying the meaning of a sentence. This effect was also evident when applying *POST STSS* to data in English (Batanović and Bojić, 2015), and was previously noted by other researchers as well (Wiemer-Hastings, 2004). Auxiliary verbs, on the other hand, carry far less semantic

[6] http://vukbatanovic.github.io/STSFineGrain/

content and are therefore assigned a lower weight, most often in the 0.7 – 0.9 range. With regard to this, participial adjectives are consistently found to be the most important kind of adjectives, with the maximum POS weight value. The weight of possessive adjectives[7] also tends to be augmented, but to a lesser extent, while other adjectives are most often assigned the 0.9 weight value. The weights within the adverb category follow a similar pattern – adverbial participles are assigned higher weights, around 1.2, while the weight of adverbs proper is usually 1.0 or 1.1. Numerals, particularly ordinal ones, are found to be quite important – their weight values usually approach the upper POS weight bound. The high weight values assigned to numerals, adverbs, and most adjectives probably indicate their importance in correctly measuring the exact level of semantic similarity between sentences whose main actions/verbs are the same. The POS weights of pronouns are generally lower, ranging between 0.7 and 1.0, while the weight value assigned to abbreviations tends to fluctuate between 0.8 and 1.0. The remaining parts of speech mostly consist of functional words, such as conjunctions, adpositions, interjections, etc., which do not contain salient semantic content and are, thus, assigned low weight values in the 0.7 – 0.8 range.

The optimized POS interaction matrix allows the pairing of words belonging to different parts of speech in most cases, but some nonsensical pairings are generally prohibited, like the one of pronouns and purely functional words like conjunctions. However, the fact that most pairings remain permitted shows that such strict prohibitions are only useful in a very limited number of cases, and that, in performance optimization, the *POST/POS-TF STSS* models rely primarily on the modification of POS weight values. This conclusion is further validated by the fact that the optimal string similarity weight in these models usually remains at the starting value of 0.5. Consequently, the optimal semantic similarity weight has the same value.

5. Conclusion

In this paper, we have presented the Serbian STS News Corpus, the first STS corpus with fine-grained semantic similarity scores in a Slavic language. We have compared it to similar STS corpora in other languages and have evaluated several unsupervised baseline STS models on it. A number of previously presented supervised models have also been considered. In addition, we have proposed *POS-TF STSS*, a new bag-of-words method that uses both term frequency weighting and part-of-speech weighting, and outperforms similar algorithms on *STS.news.sr*. The effects of various morphological normalization techniques on STS model performances have also been evaluated. In particular, we have found that using the stemmer for Croatian by Ljubešić and Pandžić alongside the *POS-TF STSS* approach yields the best results among the evaluated models. Finally, the optimal values of supervised model parameters have been discussed.

In the future, we plan to construct additional, topically distinct STS corpora in Serbian, and to use them to conduct a more thorough model evaluation. We also aim to examine the influence of gold score distribution irregularities on the behavior of STS models.

[7] NB: Possessive adjectives in Serbian correspond to possessive noun forms in English.

6. Acknowledgements

This work was partially supported by the III 44009 research grant of the Ministry of Education, Science, and Technological Development of the Republic of Serbia. The annotation of the Serbian STS News Corpus was supported by the Regional Linguistic Data Initiative (*ReLDI*) via the Swiss National Science Foundation grant no. 160501. The authors would like to thank Aleksandar Milinković for assisting in *srWaC* lemmatization, and Ana Bjelogrlić, Jelena Bošnjak, Marko Janković, and Ognjen Krešić for annotating the *STS.news.sr* corpus. The first author also served as an annotator.

7. Bibliographical References

Agarwal, D., Mujadia, V., Sharma, D. M., and Mamidi, R. (2017). A Modified Annotation Scheme for Semantic Textual Similarity. In *Proceedings of the 18th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing-2017)*, Budapest, Hungary.

Agić, Ž., Ljubešić, N., and Merkler, D. (2013). Lemmatization and Morphosyntactic Tagging of Croatian and Serbian. In *Proceedings of the Fourth Biennial International Workshop on Balto-Slavic Natural Language Processing*, Sofia, Bulgaria, Association for Computational Linguistics, pp. 48–57.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2014). SemEval-2014 Task 10: Multilingual Semantic Textual Similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, Ireland, Association for Computational Linguistics, pp. 81–91.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2015). SemEval-2015 Task 2: Semantic Textual Similarity, English, Spanish and Pilot on Interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, Denver, Colorado, USA, Association for Computational Linguistics, pp. 252–263.

Agirre, E., Banea, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Mihalcea, R., … Wiebe, J. (2016). SemEval-2016 Task 1: Semantic Textual Similarity, Monolingual and Cross-Lingual Evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, San Diego, California, USA, Association for Computational Linguistics, pp. 497–511.

Agirre, E., Cer, D., Diab, M., and Gonzalez-Agirre, A. (2012). SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM)*, Montreal, Canada, Association for Computational Linguistics, pp. 385–393.

Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., and Guo, W. (2013). *SEM 2013 shared task: Semantic Textual Similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM)*, Atlanta, Georgia, USA, Association for Computational Linguistics, pp. 32–43.

Batanović, V., and Bojić, D. (2015). Using Part-of-Speech Tags as Deep-Syntax Indicators in Determining Short-Text Semantic Similarity. *Computer Science and*

Information Systems, 12(1), pp. 1–31.

Batanović, V., Furlan, B., and Nikolić, B. (2011). A software system for determining the semantic similarity of short texts in Serbian. In *Proceedings of the 19th Telecommunications Forum (TELFOR 2011)*, Belgrade, Serbia, IEEE, pp. 1249–1252. [in Serbian]

Batanović, V., and Nikolić, B. (2016). Sentiment Classification of Documents in Serbian: The Effects of Morphological Normalization. In *Proceedings of the 24th Telecommunications Forum (TELFOR 2016)*, Belgrade, Serbia, IEEE.

Batanović, V., and Nikolić, B. (2017). Sentiment Classification of Documents in Serbian: The Effects of Morphological Normalization and Word Embeddings. *Telfor Journal*, 9(2), pp. 104–109.

Batanović, V., Nikolić, B., and Milosavljević, M. (2016). Reliable Baselines for Sentiment Analysis in Resource-Limited Languages: The Serbian Movie Review Dataset. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, European Language Resources Association (ELRA), pp. 2688–2696.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017)*, Vancouver, Canada, Association for Computational Linguistics, pp. 1–14.

Corley, C., and Mihalcea, R. (2005). Measuring the Semantic Similarity of Texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment (EMSEE 2005)*, Ann Arbor, Michigan, USA, Association for Computational Linguistics, pp. 13–18.

Dolan, W. B., and Brockett, C. (2005). Automatically Constructing a Corpus of Sentential Paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP 2005)*, Jeju Island, South Korea, Asia Federation of Natural Language Processing, pp. 9–16.

Erjavec, T. (2017). MULTEXT-East. In *Handbook of Linguistic Annotation*, Springer, Dordrecht, pp. 441–462.

Fonseca, E. R., Santos, L. B. dos, Criscuolo, M., and Aluísio, S. M. (2016). Visão Geral da Avaliação de Similaridade Semântica e Inferência Textual. *Linguamática*, 8(2), pp. 3–13. [in Portuguese]

Furlan, B., Batanović, V., and Nikolić, B. (2013). Semantic similarity of short texts in languages with a deficient natural language processing support. *Decision Support Systems*, 55(3), pp. 710–719.

Gesmundo, A., and Samardžić, T. (2012a). Lemmatisation as a tagging task. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, Jeju Island, Korea, Association for Computational Linguistics, pp. 368–372.

Gesmundo, A., and Samardžić, T. (2012b). Lemmatising Serbian as Category Tagging with Bidirectional Sequence Classification. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul, Turkey, European Language Resources Association (ELRA), pp. 2103–2106.

Hayashi, Y., and Luo, W. (2016). Extending Monolingual Semantic Textual Similarity Task to Multiple Cross-lingual Settings. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, European Language Resources Association (ELRA), pp. 1233–1239.

Islam, A., and Inkpen, D. (2008). Semantic Text Similarity Using Corpus-Based Word Similarity and String Similarity. *ACM Transactions on Knowledge Discovery from Data*, 2(2), p. Article No. 10.

Jongejan, B., and Dalianis, H. (2009). Automatic training of lemmatization rules that handle morphological changes in pre- , in- and suffixes alike. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the Fourth International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL 2009)*, Suntec, Singapore, ACL and AFNLP, pp. 145–153.

Kešelj, V., and Šipka, D. (2008). A Suffix Subsumption-Based Approach to Building Stemmers and Lemmatizers for Highly Inflectional Languages with Sparse Resources. *INFOtheca*, 9(1–2), p. 23a–33a.

Ljubešić, N., Boras, D., and Kubelka, O. (2007). Retrieving Information in Croatian: Building a Simple and Efficient Rule-Based Stemmer. In *INFuture2007: Digital Information and Heritage*, Zagreb, Croatia, Department for Information Sciences, Faculty of Humanities and Social Sciences, pp. 313–320.

Ljubešić, N., Erjavec, T., Fišer, D., Samardžić, T., Miličević, M., Klubička, F., and Petkovski, F. (2016). Easily Accessible Language Technologies for Slovene , Croatian and Serbian. In *Proceedings of the Conference on Language Technologies & Digital Humanities*, Ljubljana, Slovenia, pp. 120–124.

Ljubešić, N., and Klubička, F. (2014). {bs,hr,sr}WaC – Web corpora of Bosnian, Croatian and Serbian. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, Gothenburg, Sweden, Association for Computational Linguistics, pp. 29–35.

Ljubešić, N., Klubička, F., Agić, Ž., and Jazbec, I.-P. (2016). New Inflectional Lexicons and Training Corpora for Improved Morphosyntactic Annotation of Croatian and Serbian. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, European Language Resources Association (ELRA), pp. 4264–4270.

Mihalcea, R., Corley, C., and Strapparava, C. (2006). Corpus-based and Knowledge-based Measures of Text Semantic Similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006)*, Boston, Massachusetts, USA, AAAI Press, pp. 775–780.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Learning Representations Workshop (ICLR 2013)*, Scottsdale, Arizona, USA.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS 2013)*, Lake Tahoe, Nevada, USA, Curran Associates, Inc., pp. 3111–3119.

Milošević, N. (2012). Stemmer for Serbian language, arXiV 1209.4471.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, Philadelphia, Pennsylvania, USA, Association for Computational Linguistics, pp. 79–86.

Řehůřek, R., and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, European Language Resources Association (ELRA), pp. 45–50.

Samardžić, T., Ljubešić, N., and Miličević, M. (2015). Regional Linguistic Data Initiative (ReLDI). In *Proceedings of the Fifth Workshop on Balto-Slavic Natural Language Processing (BSNLP 2015)*, Hissar, Bulgaria, pp. 40–42.

Samardžić, T., Starović, M., Agić, Ž., and Ljubešić, N. (2017). Universal Dependencies for Serbian in Comparison with Croatian and Other Slavic Languages. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2017)*, Valencia, Spain, Association for Computational Linguistics, pp. 39–44.

Vu, H. H., Villaneau, J., Saïd, F., and Marteau, P. F. (2014). Sentence similarity by combining explicit semantic analysis and overlapping n-grams. In *Proceedings of the 17th International Conference on Text, Speech and Dialogue (TSD 2014)*, Brno, Czech Republic, Springer International Publishing, pp. 201–208.

Wiemer-Hastings, P. (2004). All parts are not created equal: SIAM-LSA. In *Proceedings of the 26th Annual Conference of the Cognitive Science Society*, Chicago, Illinois, USA, Erlbaum.

Wróblewska, A., and Krasnowska-Kieraś, K. (2017). Polish evaluation dataset for compositional distributional semantics models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, Vancouver, Canada, Association for Computational Linguistics, pp. 784–792.

8. Language Resource References

Agirre, E., Cer, D., Diab, M., and Gonzalez-Agirre, A. (2012). MSRPar (*SemEval* 2012 STS shared task), distributed online: https://www.cs.york.ac.uk/semeval-2012/task6/index.php%3Fid=data.html, 1.0.

Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., and Guo, W. (2013). HDL (*SemEval* 2013 STS shared task), distributed online: http://ixa2.si.ehu.es/sts/index.php%3Foption=com_content&view=article&id=49&Itemid=56.html, 1.0.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2014). Deft-news (*SemEval* 2014 STS shared task), distributed online: http://alt.qcri.org/semeval2014/task10/index.php?id=data-and-tools, 1.0.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2014). HDL (*SemEval* 2014 STS shared task), distributed online: http://alt.qcri.org/semeval2014/task10/index.php?id=data-and-tools, 1.0.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2014). News [Spanish] (*SemEval* 2014 STS shared task), distributed online: http://alt.qcri.org/semeval2014/task10/index.php?id=data-and-tools, 1.0.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2015). HDL (*SemEval* 2015 STS shared task), distributed online: http://alt.qcri.org/semeval2015/task2/index.php?id=data-and-tools, 1.0.

Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., … Wiebe, J. (2015). News [Spanish] (*SemEval* 2015 STS shared task), distributed online: http://alt.qcri.org/semeval2015/task2/index.php?id=data-and-tools, 1.0.

Agirre, E., Banea, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Mihalcea, R., … Wiebe, J. (2016). HDL (*SemEval* 2016 STS shared task), distributed online: http://alt.qcri.org/semeval2016/task1/index.php?id=data-and-tools, 1.0.

Batanović, V., Furlan B., and Nikolić B. (2011). The Serbian Paraphrase Corpus (*paraphrase.sr*), distributed online: http://vukbatanovic.github.io/paraphrase.sr/, 1.0, ISLRN 192-200-046-033-9.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). MSRPar [Arabic] (*SemEval* 2017 STS shared task), distributed online: http://alt.qcri.org/semeval2017/task1/index.php?id=data-and-tools, 1.0.

Dolan, W. B., Brockett, and C., Quirk, C. (2005). Microsoft Research Paraphrase Corpus, distributed online: http://www.microsoft.com/en-us/download/details.aspx?id=52398, 1.0.

Fonseca, E. R., Santos, L. B. dos, Criscuolo, M., and Aluísio, S. M. (2016). ASSIN (Avaliação de Similaridade Semântica e INferência textual), distributed online: http://nilc.icmc.usp.br/assin/, 1.0.

Ljubešić, N., and Klubička, F. (2014). Serbian web corpus srWaC, distributed via CLARIN.SI: http://hdl.handle.net/11356/1063, 1.1.

SPADE: Evaluation Dataset for Monolingual Phrase Alignment

Yuki Arase[1⋆] **and Junichi Tsujii**[⋆2]
[1]Osaka University, Japan
[⋆]Artificial Intelligence Research Center (AIRC), AIST, Japan
[2]NaCTeM, School of Computer Science, University of Manchester, UK
arase@ist.osaka-u.ac.jp, j-tsujii@aist.go.jp

Abstract

We create the SPADE (Syntactic Phrase Alignment Dataset for Evaluation) for systematic research on syntactic phrase alignment in paraphrasal sentences. This is the first dataset to shed lights on syntactic and phrasal paraphrases under linguistically motivated grammar. Existing datasets available for evaluation on phrasal paraphrase detection define the unit of phrase as simply sequence of words without syntactic structures due to difficulties caused by the non-homographic nature of phrase correspondences in sentential paraphrases. Different from these, the SPADE provides annotations of gold parse trees by a linguistic expert and gold phrase alignments identified by three annotators. Consequently, $20,276$ phrases are extracted from 201 sentential paraphrases, on which $15,721$ alignments are obtained that at least one annotator regarded as paraphrases. The SPADE is available at Linguistic Data Consortium for future research on paraphrases. In addition, two metrics are proposed to evaluate to what extent the automatic phrase alignment results agree with the ones identified by humans. These metrics allow objective comparison of performances of different methods evaluated on the SPADE. Benchmarks to show performances of humans and the state-of-the-art method are presented as a reference for future SPADE users.

Keywords: phrase alignment, paraphrase detection

1. Introduction

Paraphrases have been applied to various NLP applications, and recently, they are recognized as a useful resource for natural language understanding, such as semantic parsing (Berant and Liang, 2014) and automatic question answering (Dong et al., 2017).

While most previous studies focused on sentential paraphrase detection, *e.g.*, (Dolan et al., 2004), finer grained paraphrases, *i.e.*, phrasal paraphrases, are desired by the applications. In addition, syntactic structures are important in modeling sentences, *e.g.*, their sentiments and semantic similarities (Socher et al., 2013; Tai et al., 2015). A few studies worked on phrasal paraphrase identification on sentential paraphrases (Yao et al., 2013); however, the units of correspondence in previous studies are defined as sequences of words and not syntactic phrases due to difficulties caused by the non-homographic nature of phrase correspondences. To overcome these challenges, one promising approach is phrase alignment on paraphrasal sentence pairs based on their syntactic structures derived by linguistically motived grammar. A flexible mechanism to allow non-compositional phrase correspondences is also required. We have published our initial attempt on this direction (Arase and Tsujii, 2017).

For systematic research on syntactic phrase alignment in paraphrases, an evaluation dataset as well as evaluation measures are essential. Hence, we constructed the SPADE (Syntactic Phrase Alignment Dataset for Evaluation) and released it through Linguistic Data Consortium[1] (catalog ID: LDC2018T09[2]). In the SPADE, 201 sentential paraphrases are annotated gold parse trees, on which $20,276$ phrases exist. Three annotators annotated alignments among these phrases as shown in Figure 1, resulted in $15,721$ alignments that at least an annotator regarded as paraphrases. We also propose two measures to evaluate the quality of phrase alignment on the SPADE, which have been used as official evaluation metrics in (Arase and Tsujii, 2017). These measures allow objective comparison of performances of different methods evaluated on the SPADE.

2. Related Work

Extensive research efforts have been made for sentential paraphrase detection. One of promising resources that provide paraphrases is machine translation evaluation corpora. In such a corpus, a source sentence is translated into multiple translations in a target language. These translations are called reference translations, which can be regarded as sentential paraphrases (Weese et al., 2014). Since the reference translations are constrained to convey the same information in similar structures with the source sentences, they can be regarded as authentic paraphrases containing purely a paraphrasal phenomenon.

Although the amount of reference translations are relatively large thanks to efforts by the research community, they require severe human workloads for creation and expansion. To explore more abundant resources to extract paraphrases, Microsoft Research Paraphrase Corpus (Dolan et al., 2004) aligns news texts published at the same timing as paraphrases. Twitter URL Corpus takes a similar approach on news headlines and comments to them published at Twitter[3]: it uses attached URLs as a primary clew to find paraphrasal candidates (Lan et al., 2017). Since paraphrases in these datasets are not strictly constrained like in the ones extracted from machine translation evaluation corpora, they involve variety of linguistic phenomena beyond the conven-

[1]https://www.ldc.upenn.edu/

[2]https://catalog.ldc.upenn.edu/LDC2018T09 (will be effective since March 2018)

[3]https://twitter.com/

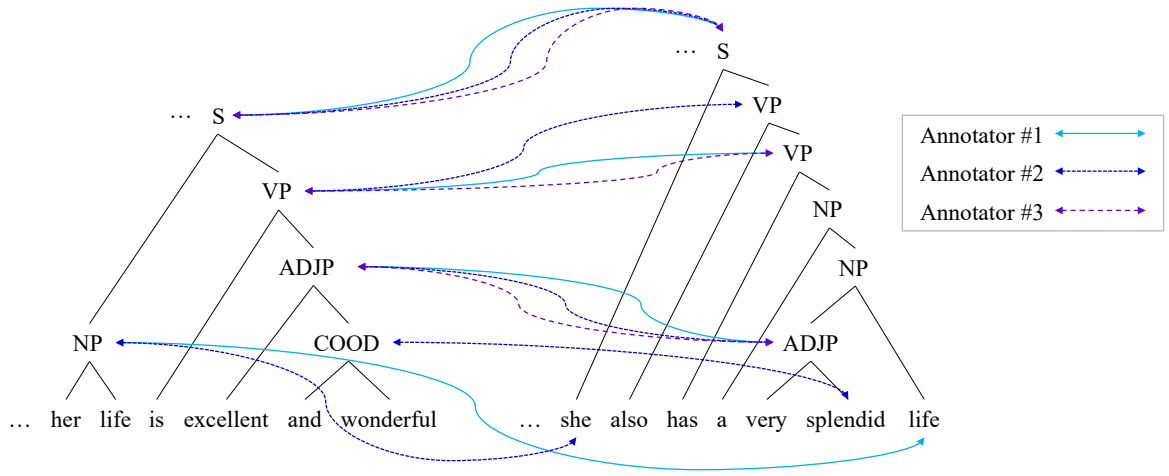


Figure 1: SPADE data example: part of gold trees and phrase alignments on a sentence pair of "Hence, I also have a reason to believe that her life is excellent and wonderful." and "So I have reason to believe that she also has a very splendid life."

tional scope of paraphrasing, such as entailment, inference, and drastic summarization.

Although costly, manually generating paraphrases is the way to produce a high-quality dataset. SICK (Marelli et al., 2014) was constructed from image and video captions through sentence alignment and careful edits to exclude undesired linguistic phenomena. In (Choe and McClosky, 2015), a linguist manually generated paraphrases to given sentences. To scale up the process trading off the quality, crowd-sourcing has been explored (Jiang et al., 2017).

As for phrasal paraphrase datasets, there are only a few; PPDB (Ganitkevitch et al., 2013) and its extension annotated levels of paraphrasability (Wieting et al., 2015). PPDB uses bilingual pivoting on parallel corpora; multiple translations of the same source phrase are regarded as paraphrases. While researchers proposed methods to identify phrasal correspondences for natural language inferences (MacCartney et al., 2008; Thadani et al., 2012; Yao et al., 2013), the unit of phrase was simply n-gram and syntax in paraphrases was out of their scope. Part of PPDB provides syntactic paraphrases under the synchronous context free grammar (SCFG); however, SCFG captures only a fraction of paraphrasing phenomenon (Weese et al., 2014). Hence, the SPADE is unique for providing fully syntactic and phrasal paraphrases.

3. Construction of SPADE

We create the SPADE for evaluation on syntactic phrase alignment in paraphrases. Two rounds of annotations were carefully conducted to annotate gold parse trees and phrase alignments.

3.1. Corpus

Paraphrasal sentence pairs to annotate were extracted from the NIST OpenMT[4] that are machine translation evaluation corpora. As discussed in Section 2., reference translations can be regarded as authentic paraphrases. Other types of paraphrase copora are left for our future study.

Reference translations of 10 to 30 words were randomly extracted for annotation. To diversify the data, only one reference pair per source sentence was chosen.

3.2. Gold-Standard Parse Tree

We essentially need phrase structure grammars to recognize phrases. In addition, we believe that rich syntactic information is useful for deriving rules or applying machine learning techniques in phrase alignment process. Hence, we decided to use Head-driven phrase structure grammar (HPSG) (Pollard and Sag, 1994) to assign gold parse trees to sentences.

We asked a linguistic expert with rich experience on annotating HPSG trees to annotate gold-trees to paraphrasal sentence pairs. Consequently, 201 paraphrased pairs with gold-trees (containing 20, 276 phrases) were obtained.

3.3. Gold-Standard Phrase Alignment

Next, three professional English translators identified paraphrased pairs including phrases with no correspondences given sets of phrases extracted from the gold-trees. These annotators independently annotated the same set.

The annotators were given an annotation guideline with detailed instructions and examples. They were also provided an annotation tool developed by us for simpler annotation process and easier management of progress. Figure 2 shows a screenshot of the annotation tool, in which three panes show (1) phrases extracted from a sentence, (2) phrases extracted from another sentence, and (3) annotation results, from left to right. Annotators select a phrase in each pane and assign an alignment label to declare if the alignment is either *sure* or *possible* according to their confidence in judgment. They can also quickly refer to tree structures by clicking the "Show parse trees" button, then trees are visualized as Figure 3 shows[5]. In the panes of (1) and (2), already-aligned phrases turn their surface colors into gray for easier recognition. The annotators can modify their

[4]LDC catalog numbers: LDC2010T14, LDC2010T17, LDC2010T21, LDC2010T23, LDC2013T03

[5]Script for visualization is borrowed from `http://www.nactem.ac.uk/enju/demo.html`.

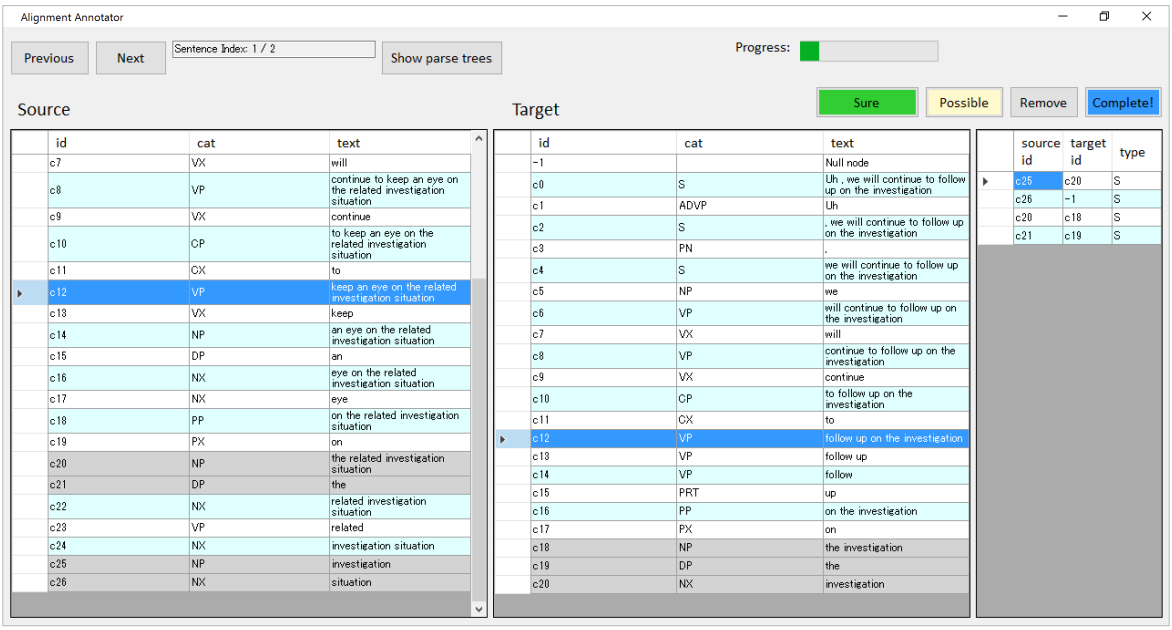


Figure 2: Annotation tool

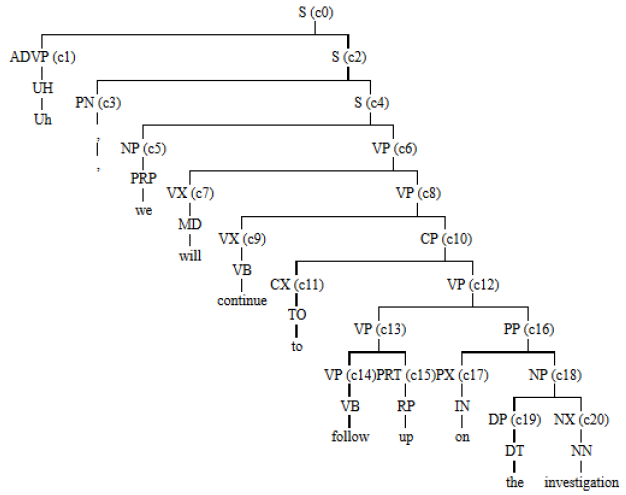


Figure 3: Tree structures of annotating sentences can be quickly referred on the annotation tool.

alignments by directly editing an alignment at the annotation result pane, or erasing it using the "Remove" button.

3.4. Statistics in SPADE

As results of annotation, 15,721 phrase alignments were obtained that at least one annotator regarded as paraphrases. These alignments contain ones in which a phrase does not have any correspondence in another sentence. Such a phrase is regarded as being aligned to a *null* phrase. Figure 1 visualizes gold parse trees and phrase alignments, where alignment types of *sure* and *possible* are not distinguished.

Table 1 shows detailed statistics of the SPADE. The annotated sentence pairs were split into 50 pairs for development and another 151 for testing. In the development and test sets, 3,932 and 11,789 alignments are the ones regarded as paraphrases by at least an annotator, respectively. Among them, 2,518 and 7,134 alignments are agreed by all annotators, respectively. Hence, the overall agreement rate is 61.4%. Although the numbers of sentences in the development and test sets sound limited, we deem that those of phrase alignments are sufficient for evaluation.

	Development	Test
# of sentence pairs	50	151
# of tokens	2,494	7,276
# of types	736	1,573
# of phrases (w/o tokens)	5,201	15,075
# of alignments (∪)	3,932	11,789
# of alignments (∩)	2,518	7,134

Table 1: Statistics of development and test sets in SPADE

4. Evaluation Measure

Phrase alignment quality should be evaluated by measuring the extent that automatic alignment results of a certain method agree with those of humans. We propose two measures, named ALIR (alignment recall) and ALIP (alignment precision) based on conventional recall and precision. These are used in (Arase and Tsujii, 2017) as official evaluation metrics. Together with the SPADE, researchers objectively compare performances of their methods measured by ALIR and ALIP with those of others .

Specifically, ALIR evaluates how gold-alignments can be replicated by automatic alignments and ALIP measures how automatic alignments overlap with alignments that at least an annotator aligned as:

$$\text{ALIR} = \frac{|\{\mathbb{h}|\mathbb{h} \in \mathbb{H}_a \wedge \mathbb{h} \in \mathbb{G} \cap \mathbb{G}'\}|}{|\mathbb{G} \cap \mathbb{G}'|},$$

$$\text{ALIP} = \frac{|\{\mathbb{h}|\mathbb{h} \in \mathbb{H}a \wedge \mathbb{h} \in \mathbb{G} \cup \mathbb{G}'\}|}{|\mathbb{H}a|},$$

Method	ALIR	ALIP
Human	90.65	88.21
(Arase and Tsujii, 2017)	83.64	78.91

Table 2: Benchmarks of ALIR and ALIP on the SPADE

where $\mathbb{H}a$ is a set of automatic alignments, while $\mathbb{G}$ and $\mathbb{G}'$ are the ones that two of annotators produce, respectively. The function of $|\cdot|$ counts the elements in a set.

Since we have three annotators, there are three combinations for $\mathbb{G}$ and $\mathbb{G}'$. The final ALIR and ALIP values are calculated by taking the averages.

4.1. Benchmark

As a benchmark of evaluation using the SPADE, Table 2 shows performances of humans and (Arase and Tsujii, 2017). Although we have *sure* and *possible* alignments, we did not distinguish them in the evaluation due to variance in annotators' decision to assign either label[6].

The performance of the human annotators was assessed by considering one annotator as the test and the other two as the gold-standard, and then taking the averages, which is the same setting as the automatic method. We regard this as the pseudo inter-annotator agreement, since the conventional inter-annotator agreement is not directly applicable due to variations in combinations of aligned phrases as depicted in Figure 1.

ALIR and ALIP of (Arase and Tsujii, 2017) reach 92% and 89% of those of humans, respectively, though there still significant gaps to the human performance. Research efforts from variety of groups are desired for further progress in syntactic phrase alignment research.

5. Conclusion

We created the SPADE that provides annotations of gold HPSG trees and phrase alignments on sentential paraphrases extracted from machine translation evaluation corpora. This dataset was released through LDC. Two evaluation measures, ALIR and ALIP, were also proposed, which allow to compare the extent that automatic phrase alignment results agree with the ones produced by humans.

The creation of SPADE is just initiated, and there remains room for future development. The most important task is employing other sentential paraphrase corpora for annotation. We are analyzing the linguistic phenomena observed in Twitter URL Corpus and Microsoft Research Paraphrase Corpus, where drastic summarization or lengthening by adding context, as well as entailments and inference due to people's world knowledge are happening. These are beyond the conventional scope of paraphrasing; however, considering such paraphrases spontaneously arise and widely available, we should definitely need technologies to handle such paraphrases in the wild. We will annotate paraphrases in these corpora to scale up the SPADE in terms of its size as well as the variety of paraphrasal phenomena.

[6]We also observed results using only *sure* alignments, and confirmed that they show the same trends.

Acknowledgments

We thank Dr. Yuka Tateishi for her contribution to HPSG tree annotation. This project is funded by Microsoft Research Asia and the Kayamori Foundation of Informational Science Advancement.

Bibliographical References

Arase, Y. and Tsujii, J. (2017). Monolingual phrase alignment on parse forests. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1–11, Copenhagen, Denmark, September.

Berant, J. and Liang, P. (2014). Semantic parsing via paraphrasing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1415–1425, Baltimore, Maryland, June.

Choe, D. K. and McClosky, D. (2015). Parsing paraphrases with joint inference. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1223–1233, Beijing, China, July.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceesings of the International Conference on Computational Linguistics (COLING)*, pages 350–356, Geneva, Switzerland, August.

Dong, L., Mallinson, J., Reddy, S., and Lapata, M. (2017). Learning to paraphrase for question answering. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 886–897, Copenhagen, Denmark, September.

Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2013). PPDB: The paraphrase database. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 758–764, Atlanta, Georgia, June.

Jiang, Y., Kummerfeld, J. K., and Lasecki, W. S. (2017). Understanding task design trade-offs in crowdsourced paraphrase collection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 103–109, Vancouver, Canada, July.

Lan, W., Qiu, S., He, H., and Xu, W. (2017). A continuously growing dataset of sentential paraphrases. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1235–1245, Copenhagen, Denmark, September.

MacCartney, B., Galley, M., and Manning, C. D. (2008). A phrase-based alignment model for natural language inference. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 802–811, Honolulu, Hawaii, October.

Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., and Zamparelli, R. (2014). A SICK cure for the evaluation of compositional distributional semantic models. In *Proceesings of the Language Re-*

sources and Evaluation Conference (LREC), pages 216–223, Reykjavik, Iceland.

Pollard, C. and Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar*. CSLI Publications.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642, Seattle, Washington, USA, October.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *ACL-IJCLNLP*, pages 1556–1566, Beijing, China, July.

Thadani, K., Martin, S., and White, M. (2012). A joint phrasal and dependency model for paraphrase alignment. In *Proceesings of the International Conference on Computational Linguistics (COLING)*, pages 1229–1238, Mumbai, India, December.

Weese, J., Ganitkevitch, J., and Callison-Burch, C. (2014). PARADIGM: Paraphrase diagnostics through grammar matching. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 192–201, Gothenburg, Sweden, April.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2015). From paraphrase database to compositional paraphrase model and back. *Transactions of the Association of Computational Linguistics (TACL)*, 3(1):345–358.

Yao, X., Van Durme, B., Callison-Burch, C., and Clark, P. (2013). Semi-Markov phrase-based monolingual alignment. In *Proceesings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 590–600, Seattle, Washington, USA, October.

ETPC - A Paraphrase Identification Corpus Annotated with Extended Paraphrase Typology and Negation

Venelin Kovatchev, M. Antònia Martí, Maria Salamó
Universitat de Barcelona
Gran Vía de les Corts Catalanes, 585, 08007 Barcelona, Spain
{vkovatchev, amarti, maria.salamo}@ub.edu

Abstract

We present the Extended Paraphrase Typology (EPT) and the Extended Typology Paraphrase Corpus (ETPC). The EPT typology addresses several practical limitations of existing paraphrase typologies: it is the first typology that copes with the non-paraphrase pairs in the paraphrase identification corpora and distinguishes between contextual and habitual paraphrase types. ETPC is the largest corpus to date annotated with atomic paraphrase types. It is the first corpus with detailed annotation of both the paraphrase and the non-paraphrase pairs and the first corpus annotated with paraphrase and negation. Both new resources contribute to better understanding the paraphrase phenomenon, and allow for studying the relationship between paraphrasing and negation. To the developers of Paraphrase Identification systems ETPC corpus offers better means for evaluation and error analysis. Furthermore, the EPT typology and ETPC corpus emphasize the relationship with other areas of NLP such as Semantic Similarity, Textual Entailment, Summarization and Simplification.

Keywords: Paraphrasing, Paraphrase Typology, Paraphrase Identification

1. Introduction

The task of Paraphrase Identification (PI) consists of comparing two texts of arbitrary size in order to determine whether they have approximately the same meaning. The most common approach to PI is as a binary classification problem, in which a system learns to make correct binary predictions (paraphrase or non-paraphrase) for a given pair of texts. The task of PI is challenging from more than one point of view. From the resource point of view, defining the task and preparing high quality corpora is a non-trivial problem due to the complex nature of "paraphrasing". From the application point of view, for a system to perform well on PI often requires a complex ML architecture and/or a large set of manually engineered features. From the evaluation point of view, the classical task of PI does not offer many possibilities for detailed error analysis, which in turn limits the reusability and the improvement of PI systems.

In the last few years, researchers in the field of paraphrasing have adopted the approach of decomposing the meta phenomenon of *"textual paraphrasing"* into a set of *"atomic paraphrase"* phenomena, which are more strictly defined and easier to work with. *"Atomic paraphrases"* are hierarchically organized into a typology, which provides a better means to study and understand paraphrasing. While the theoretical advantages of these approaches are clear, their practical implications have not been fully explored. The existing corpora annotated with paraphrase typology are limited in size, coverage and overall quality. The only corpus of sufficient size to date annotated with paraphrase typology is the corpus by Vila et al. (2015), which contains 3900 re-annotated *"textual paraphrase"* pairs from the MRPC corpus (Dolan et al., 2004).

The use of a paraphrase typology in practical tasks has several advantages. First, *"atomic paraphrases"* are much more strict in their definition, which makes the results more useful and understandable. Second, the more detailed annotation can be useful to (re)balance binary PI corpora in terms of type distribution. Third, annotating a corpus with paraphrase types provides much better feedback to the PI systems and allows for a detailed, per-type error analysis. Fourth, enriching the corpus and improving the evaluation can provide a linguistic insight into the workings of complex machine learning systems (i.e. Deep Learning) that are traditionally hard to interpret. Fifth, corpora annotated with a paraphrase typology open the way for new research and new tasks, such as "PI by type" or "Atomic PI in context". Finally, decomposing *"textual paraphrases"* can help relate the task of PI to tasks such as Recognizing Textual Entailment, Text Summarization, Text Simplification, and Question Answering.

In this paper, we present the Extended Typology Paraphrase Corpus (ETPC), the result of annotating the MRPC (Dolan et al., 2004) corpus with our Extended Paraphrase Typology (EPT). EPT is oriented towards practical applications and takes inspiration from several authors that work on the typology of paraphrasing and textual entailment. To the best of our knowledge, this is the first attempt to make a detailed annotation of the linguistic phenomena involved in both the positive (paraphrases) and negative (non-paraphrases) examples in the MRPC (for a total size of 5801 textual pairs). The focus on non-paraphrases and the qualitative and quantitative comparison between *"textual paraphrases"* and *"textual non-paraphrases"* provides a different perspective on the PI task and corpora.

As a separate layer of annotation, we have identified all pairs of texts that include negation and we have annotated the negation scope. This makes ETPC the first corpus that is annotated both with paraphrasing and with negation.

The rest of this article is organized as follows. Section 2. is devoted to the Related Work. Section 3. describes the proposed Extended Typology, the reasons and the practical considerations behind it. Section 4. explains the annotation process, the annotation scheme and instructions, the tool that we used and the corpus preprocessing. Section 5.

presents ETPC, with its structure and type distribution. It discusses the results of the annotation and outlines some of the practical applications of the corpus. Finally, Section 6. concludes the article and outlines the future work.

2. Related Work

The task of PI is one of the classical tasks in NLP. Several corpora can be used in the task for training and/or for evaluation. Traditionally, PI is addressed using the MRPC corpus (Dolan et al., 2004). The MRPC corpus consists of 5801 pairs, that have been manually annotated as paraphrases or non-paraphrases. More recently, Ganitkevitch et al. (2013) introduce PPDB - a very large automatic collection of paraphrases, which consists of 220 million pairs. The introduction of PPDB allowed for the training of deep learning systems, due to the significant increase of the available data. However, the quality of the PPDB pairs is much lower than those of MRPC, which makes it less reliable for evaluation. A common approach is to work on both datasets simultaneously - using the PPDB for training, and the MRPC for development and evaluation.

Closely related to the PI task is the yearly task of Recognizing Textual Entailment (RTE) (Dagan et al., 2006), which has also produced various datasets and multiple practical systems. The meta-phenomena of paraphrasing and textual entailment are very similar and are often studied together at least from a theoretical point of view. Androutsopoulos and Malakasiotis (2010) present a summary of the tasks related to both paraphrasing and textual entailment.

The idea of decomposing paraphrasing into simpler and easier to define phenomena has been growing in popularity in the last few years. Bhagat (2009) and later Bhagat and Hovy (2013) propose a simplified framework that identifies several possible phenomena involved in the paraphrasing relation. Vila et al. (2014) propose a more complex, hierarchically structured typology that studies the different phenomena at the corresponding linguistic levels (lexical, morphological, syntactic, and discourse). More recently, Benikova and Zesch (2017) approach the problem by focusing on the paraphrasing at the level of events, understood as predicate-argument structure.

A similar decomposition tendency is noticed in the field of Textual Entailment. Garoufi (2007), Sammons et al. (2010), and Cabrio and Magnini (2014) propose different frameworks for decomposing the textual "inference" into simple, atomic phenomena. It is important to note that the similarity and the relation between paraphrasing and textual entailment is even stronger in the context of the decomposition framework and the resulting typologies. The two most exhaustive typologies: Vila et al. (2014) for paraphrasing and Cabrio and Magnini (2014) for textual entailment share the majority of their atomic phenomena as well as the overall structure and organization of the typology.

One of the advantages of the decomposition approaches is that naturally they work towards bridging the gap between the research at different granularity levels. A corpora annotated with semantic relations at both the textual and the atomic (morphological, lexical, syntactic, discourse) levels can be a valuable resource for studying the relation between them. In this same line of work, Shwartz and Dagan (2016) emphasize the importance of studying lexical entailment "in context" and the lack of resources that can enable such work. The corpora annotated with atomic paraphrase and atomic entailment phenomena can be used for that purpose without adaptation or additional annotation.

The application of paraphrase typology for the creation of resources and in practical tasks is still very limited. Most of the authors annotate a very small subsamples of around 100 text pairs to illustrate the proposed typology. The largest available corpus annotated with paraphrase types to date is the one of Vila et al. (2015). Barrón-Cedeño et al. (2013) use this corpus to demonstrate some possible uses of the decomposition approach to paraphrasing.

3. Extended Paraphrase Typology

We propose the Extended Paraphrase Typology (EPT), which was created to address several of the practical limitations of the existing typologies and to provide better resources to the NLP community. EPT ha better coverage than previous typologies, including the annotation of non-paraphrases. This allows for a more in-depth understanding of the meta-phenomena and of the relation between *"textual paraphrases"* and *"atomic paraphrases"*.

3.1. Basic Terminology

In order to discuss the issues and limitations of existing paraphrase typologies, we first define *"paraphrasing"*, *"textual paraphrase"*, and *"atomic paraphrase"*.

We understand *"paraphrasing"* to be a specific semantic relation between two texts of arbitrary length. The two texts that are connected by a paraphrase relation have approximately the same meaning. We call them *"textual paraphrases"*. There is no limitation for *"textual paraphrases"* in terms of the nature of the linguistic phenomena involved. The concept of *"textual paraphrases"* is a practical simplification of a complex linguistic phenomenon, which is adopted in most paraphrase-related tasks, datasets, and applications. The original annotation of the MRPC and the PPDB corpora is built around the notion of textual paraphrases. Another term that we use in the article is *"textual non-paraphrases"*. With this term we refer to pairs of texts (of arbitrary length), which are not connected by a paraphrase relation.

"Atomic paraphrases" are paraphrases of a particular type. They must satisfy specific (linguistic) conditions, defined in the paraphrase typology. *"Atomic paraphrases"* are identified by the linguistic phenomenon which is responsible for the preservation of the meaning between the two texts. *"Atomic paraphrases"* have a (linguistically defined) scope, such as a word, a phrase, an event, or a discourse structure. The most complete typologies to date organize *"atomic paraphrases"* hierarchically, in terms of the linguistic level of the involved phenomenon. Unlike *"textual paraphrases"*, *"atomic paraphrases"* cannot be of arbitrary length. Their length is defined and restricted by their scope.

3.2. From Atomic to Textual Paraphrases

The relation between textual and atomic paraphrases is not easy to define and explore. It poses many challenges to

the researchers, annotators, and developers of practical systems. In this section, we illustrate several issues that we want to address with the creation of the EPT and the ETPC. The first issue to be addressed is that multiple atomic paraphrases can appear in a single textual paraphrase pair. The two texts in 1a and 1b are textual paraphrases[1]. However, they include more than one atomic paraphrase': *"magistrate"* and *"judge"* are an instance of *"same polarity substitution"*, while *"A federal magistrate ... ordered"* and *"Zuccarini was ordered by a federal judge..."* are an instance of *"diathesis alternation"*[2].

1a A federal **magistrate** in Fort Lauderdale ordered him held without bail.

1b Zuccarini was ordered held without bail Wednesday by a federal **judge** in Fort Lauderdale, Fla.

Second issue is that atomic paraphrases can appear in textual pairs that are not paraphrases. The two texts in 2a and 2b as a whole are not textual paraphrases, even if they have a high degree of lexical overlap and a similar syntactic structure. However, *"Microsoft"* and *"shares of Microsoft"* are an instance of *"same polarity substitution"* - both phrases have the same role and meaning in the context of the two sentences. This demonstrates the possibility of atomic paraphrases being present in textual non-paraphrases. [3]

2a Microsoft fell 5 percent before the open to $27.45 from Thursday's close of $28.91.

2b Shares in Microsoft slipped 4.7 percent in after-hours trade to $27.54 from a Nasdaq close of $28.91.

Third issue is that in certain cases, the semantic relation between the elements in an atomic paraphrase can only be interpreted within the context (as shown in the work of Shwartz and Dagan (2016)). The two texts in 3a and 3b are textual paraphrases. The out-of-context meaning of *"cargo"* and *"explosives"* differs significantly, however within the given context, they are an instance of *"same polarity substitution"*.

3a They had published an advertisement on the Internet on June 10, offering the cargo for sale, he added.

3b On June 10, the ship's owners had published an advertisement on the Internet, offering the explosives for sale.

And finally, 4a and 4b illustrate an issue that is often overlooked in theoretical paraphrase research: the linguistic phenomena behind certain atomic paraphrases do not always preserve the meaning. The meanings of *"beat"* and *"battled"* are similar, and play the same syntactic and discourse role in the structure of the texts. Therefore, the substitution of *"beat"* for *"battled"* fulfills the formal requirements of a *"same polarity substitution"*. However, after this substitution, the resulting texts are not paraphrases as they differ substantially in meaning.

4a He beat testicular cancer that had spread to his lungs and brain.

4b Armstrong, 31, battled testicular cancer that spread to his brain.

3.3. Objectives of EPT and Research Questions.

We argue that the objectives behind a paraphrase typology are twofold: 1) to classify and describe the linguistic phenomena involved in paraphrasing (at the atomic level); and 2) to provide the means to study the function of atomic paraphrases within pairs of texts of arbitrary size and with various semantic relations (such as, textual paraphrases, textual entailment pairs, contradictions, and unrelated texts).

Traditionally, the authors of paraphrase typologies have focused on the first objective while the latter is mentioned only briefly or ignored altogether. In our work, we want to extend the existing work on paraphrase typology in the direction of Objective 2, as we argue that it is crucial for applications. We pose four research questions, that we aim to address with the creation of EPT and ETPC:

RQ1 what is the relation between atomic and textual paraphrases considering the distribution of atomic paraphrases in textual paraphrases?

RQ2 what is the relation between atomic paraphrases and textual non-paraphrases considering the distribution of atomic paraphrases in textual non-paraphrases?

RQ3 what is the role of the context in atomic paraphrases?

RQ4 in which cases do the linguistic phenomena behind an atomic paraphrase preserve the meaning and in which they do not?

3.4. The Extended Paraphrase Typology

The full Extended Paraphrase Typology is shown in Table 1. It is organized in seven meta categories: "Morphology", "Lexicon", "Lexico-syntax", "Syntax", "Discourse", "Other", and "Extremes". Sense Preserving (Sens Pres.) shows whether a certain type can give raise to textual paraphrases (+), to textual non-paraphrases (-), or to both (+ / -)[4]. The typology contains 25 atomic paraphrase types (+) and 13 atomic non-paraphrase types (-). It is based on the work of Vila et al. (2014) and aims to extend it in two directions in order to address the four Research Questions.

[1] All examples in this subsection are from the MRPC corpus. When we say that the texts are textual paraphrases or textual non-paraphrases, we refer to the labels corresponding to these pairs in MRPC.

[2] These types and annotation are from Vila et al. (2015).

[3] In fact, it is possible to find atomic paraphrases within pairs of texts connected by various relations, such as entailment, simplification, summarization, contradiction, and question-answering, among others. This is illustrated by the significant overlap of atomic types in Paraphrase Typology research and typology research in Textual Entailment.

[4] A more detailed table of EPT, with additional examples for each atomic type is available at https://github.com/venelink/ETPC

ID	Type	Sense Pres.
	Morphology-based changes	
1	Inflectional changes	+ / -
2	Modal verb changes	+
3	Derivational changes	+
	Lexicon-based changes	
4	Spelling changes	+
5	Same polarity substitution (habitual)	+
6	Same polarity substitution (contextual)	+ / -
7	Same polarity sub. (named entity)	+ / -
8	Change of format	+
	Lexico-syntactic based changes	
9	Opposite polarity sub. (habitual)	+ / -
10	Opposite polarity sub. (contextual)	+ / -
11	Synthetic/analytic substitution	+
12	Converse substitution	+ / -
	Syntax-based changes	
13	Diathesis alternation	+ / -
14	Negation switching	+ / -
15	Ellipsis	+
16	Coordination changes	+
17	Subordination and nesting changes	+
	Discourse-based changes	
18	Punctuation changes	+
19	Direct/indirect style alternations	+ / -
20	Sentence modality changes	+
21	Syntax/discourse structure changes	+
	Other changes	
22	Addition/Deletion	+ / -
23	Change of order	+
24	Semantic (General Inferences)	+ / -
	Extremes	
25	Identity	+
26	Non-Paraphrase	-
27	Entailment	-

Table 1: Extended Paraphrase Typology

First, we have added three new atomic paraphrase types - we split the atomic types *"same polarity substitution"* and *"opposite polarity substitution"* into two separate types based on the nature of the relation between the substituted words: *"habitual"* and *"contextual"*. We have also added the type *"same polarity substitution (named entity)"*. While the principle behind all substitutions is the same, in practice there is a significant difference whether the replaced words are connected in their habitual meaning, contextually, or refer to related named entities in the world. Instances of the new types can be seen in sentence pairs 5 (*"same polarity substitution (habitual)"*), 6 (*"same polarity substitution (contextual)"*), 7 (*"same polarity substitution (named entity"*), 8 (*"opposite polarity substitution (habitual)"*), and 9 (*"opposite polarity substitution (contextual)"*)

5a A federal magistrate in Fort Lauderdale ordered him held without bail.

5b Zuccarini was ordered held without bail Wednesday by a federal judge in Fort Lauderdale, Fla.

6a Meanwhile, the global death toll approached 770 with more than 8,300 people sickened since the severe acute respiratory syndrome virus first appeared in southern China in November.

6b The global death toll from SARS was at least 767, with more than 8,300 people sickened since the virus first appeared in southern China in November.

7a He told The Sun newspaper that Mr. Hussein's daughters had British schools and hospitals in mind when they decided to ask for asylum.

7b "Saddam's daughters had British schools and hospitals in mind when they decided to ask for asylum – especially the schools," he told The Sun.

8a Leicester failed in both enterprises.

8b He did not succeed in either case.

9a A big surge in consumer confidence has provided the only positive economic news in recent weeks.

9b Only a big surge in consumer confidence has interrupted the bleak economic news.

Second, we have introduced the *"sense preserving"* feature in 13 of the atomic types. As we have shown in the previous section (examples 4a and 4b), the same atomic linguistic transformation (such as substitution, diathesis alternation, and negation switching) can give raise to different semantic relations at textual level: paraphrasing, entailment, and contradiction, among others. This idea has already been expressed by Cabrio and Magnini (2014) in the field of Recognizing Textual Entailment. Building on this idea, we identify 13 atomic types that can, in different instances, give rise to both paraphrases and non-paraphrases. Sentence pairs 10 and 11 show an example of sense preserving and non-sense preserving *"Inflection change"* types. In 10a and 10b, both *"streets"* and *"street"* are a generalization with the meaning *"all streets"*. In a similar way, in 11b, *"boats"* has the meaning as *"all boats"*. However in 11a, *"boat"* can have the meaning *"one particular boat"*, thus the inflectional change *"boat - boats"* is not sense-preserving.

10a It was with difficulty that the course of streets could be followed.

10b You couldn't even follow the path of the street.

11a You can't travel from Barcelona to Mallorca with the boat.

11b Boats can't travel from Barcelona to Mallorca.

The changes introduced in EPT allow us to work on all four Research Questions (RQs) defined in Section 3.3. This is a clear advantage over the existing paraphrase typologies, which are only suitable for addressing RQ1. For RQ1, we annotated all atomic types in the positive ("paraphrases") portion of MRPC and measured their distribution. For RQ2, we annotated all atomic types in the negative ("non-paraphrases") portion of MRPC and compared the distribution of the types in the positive and negative portions. For RQ3, the two newly added "contextual" types allow us to distinguish and compare context dependent from context independent atomic paraphrases. Finally, for RQ4, the addition of "sense preserving" allows us to annotate, isolate and compare the sense preserving and non-sense preserving instances of the same linguistic phenomena.

4. Annotation Scheme and Guidelines

We propose the Extended Paraphrase Typology (EPT) with a clear practical objective in mind: to create language resources that improve the performance, evaluation, and understanding of the systems competing on the task of PI and to open new research directions. We used the EPT to annotate the MRPC corpus with atomic paraphrases. We annotated all 5801 text pairs in the corpus, including both the pairs annotated as paraphrases (3900 pairs) and those annotated as non-paraphrases (1901 pairs).

As a basis, we used the MRPC-A corpus by Vila et al. (2015), which already contains some annotated atomic paraphrases. Our annotation consisted of three steps, corresponding to the three different layers of annotation.

First, we annotated the non-sense preserving atomic phenomena (Section 4.1.) in the textual non-paraphrases. Second, we annotated the sense preserving atomic paraphrase phenomena (Section 4.2.) in both textual paraphrases and textual non-paraphrases. And third, we identified all sentences in the corpus containing negation, and explicitly annotated the negation scope (Section 4.4.).

For the purpose of the annotation, we created a web-based annotation tool, Pair-Anno, capable of annotating aligned pairs of discontinuous scopes in two different texts[5]. As the scope of each atomic phenomena is one or more sets of tokens, prior to the annotation we automatically tokenized the corpus using NLTK (Bird et al., 2009).

4.1. Non-Sense Preserving Atomic Phenomena

Textual non-paraphrases in the MRPC corpus typically have a very high degree of lexical overlap and a similar syntactic and discourse structure. Normally, they differ only by a few elements (morphological, lexical, or structural), but the modification of these few elements leads to a substantial difference in the meaning of the two texts as a whole. The annotation of non-sense preserving phenomena aims to identify these key elements and study the linguistic nature of the modification.

When annotating atomic phenomena, our experts identified and annotated the type, the scope, and in some paraphrase types, the key element. Both the scope and the key are kept as a 0-indexed list of tokens. Examples 12a and 12b show a textual pair, annotated as non-paraphrase in the MRPC corpus. Table 2 shows the annotation of non-sense preserving atomic phenomena in 12a and 12b. The key differences are *"opposite polarity substitution (habitual)"* (type id 10) of "slip" with "rise", and the *"same polarity substitution (named entity)"* (type id 7) of "Friday" with "Thursday".

12a The loonie , meanwhile , continued to slip in early trading Friday .

12b The loonie , meanwhile , was on the rise again early Thursday .

type	pair	s1 scope	s2 scope	s1 text	s2 text
7	146	11	11	Friday	Thursday
10	146	7	8	slip	rise

Table 2: Non-sense preserving phenomena

The annotation of 12a and 12b illustrates one of the issues when annotating non-sense preserving phenomena. In many textual pairs, there is more than one "key" difference. In those cases, all of the phenomena were annotated separately. Nevertheless, the annotators were instructed to be conservative and only annotate phenomena that carry substantial differences in the meaning of the two texts. Determining which differences are substantial, and which are not was the main challenge for the annotators. Due to the difficulty of the task, we selected annotators that were expert linguists with a high proficiency of English[6].

When the two texts were substantially different and it was not possible to identify the atomic phenomena responsible for the difference, the pair was annotated with atomic type *"non-paraphrase"* (examples 13a and 13b) or *"entailment"* (examples 14a and 14b).

13a That compared with $35.18 million, or 24 cents per share, in the year-ago period.

13b Earnings were affected by a non-recurring $8 million tax benefit in the year-ago period.

14a The year-ago comparisons were restated to include Compaq results.

14b The year-ago numbers do not include figures from Compaq Computer.

4.2. Sense Preserving Atomic Phenomena

For the annotation of the sense preserving atomic phenomena, we used the same annotation scheme format as the one for the non-sense preserving phenomena. Each phenomenon is identified by a type, a scope, and, where applicable, a key. 15a and 15b show a textual pair, annotated as a paraphrase in the MRPC. An example of a single annotated atomic phenomenon can be seen in Table 3

[5]Screenshots of Pair-Anno can be seen at https://github.com/venelink/ETPC.

[6]The full annotation guidelines for both sense preserving and non-sense preserving phenomena can be found at https://github.com/venelink/ETPC

15a Amrozi accused his brother , whom he called " the witness " , of deliberately distorting his evidence .

15b Referring to him as only " the witness " , Amrozi accused his brother of deliberately distorting his evidence .

type	pair	s1 scope	s2 scope	s1 text	s2 text
6	1	5	1, 2	whom	to him

Table 3: Sense preserving phenomenon

For the 3900 text pairs already annotated by Vila et al. (2015), we worked with the existing corpus and we only re-annotated the 3 new sense preserving paraphrase types introduced in EPT. For the 1901 textual non-paraphrases, which were not annotated in MRPC-A, we performed a full annotation with all 25 sense preserving atomic types.

4.3. Inter-Annotator Agreement

In this section, we present the measures for calculating the inter-annotator agreement and the agreement score on the first two layers of annotation: non-sense preserving atomic phenomena and sense preserving atomic phenomena.

The measure that we use is the IAPTA TPO, introduced by Vila et al. (2015). It is a fine-grained measure, created specifically for the task of annotating paraphrase types. It takes into account the agreement with respect to both the label and the scope of the phenomena. It is a pairwise agreement measure, obtained by calculating the Precision, Recall and F1 of one of the annotators, while using another annotator as a gold standard. There are two versions of the measure - TPO-partial, which requires that the annotators select the same label and that the scopes overlap by at least one token; and TPO-total which requires full overlap of label and scope.

The classical TPO measures are pairwise, they calculate the agreement between two annotators. When the annotation process involves more than two annotators, we first calculate the pairwise TPO measure between any two annotators and then we use one of three different techniques for calculating the overall agreement for the corpus. TPO (avg) is the most simple score, as it is the average of all pairwise TPO scores. TPO (union) is the union of all pairwise TPO agreement tables. That is, any phenomena that is annotated with the same label and the same scope by any 2 annotators is part of the TPO (union). Finally, TPO (gold) is the average F1 score of the three annotators, when treating TPO (union) as a gold standard. TPO (union) and TPO (gold) are two new measures, that we propose as part of this paper. TPO (union) represents all the "high quality" phenomena (that is, phenomena annotated the same way by multiple annotators). TPO (gold) represents the probability that any of our annotators would annotate "high quality" phenomena.

The annotation of the sense preserving atomic paraphrases was carried out by two expert annotators, while the annotation of the non-sense preserving atomic phenomena was carried out by three expert annotators. For the purpose of calculating the inter-annotator agreement, all experts were given the same 180 text pairs (roughly 10 % of all non-paraphrase pairs in the corpus). The pairs were split in 3 equal parts and given to the annotators in three different stages of the annotation: one at the beginning, one in the middle, and one at the end of the annotation process. Table 4 shows the obtained scores, where ETPC (-) stands for the non-sense preserving layer, ETPC (+) stands for the sense-preserving layer of annotation and MRPC-A is the annotation of Vila et al. (2015). For ETPC (+) we only had two annotators, so we were not able to calculate TPO (union) and TPO (gold). Since these measures have been introduced by us in the current paper, the MRPC-A corpus by Vila et al. (2015) does not have values for them either.

Measure	ETPC (-)	ETPC (+)	MRPC-A
TPO-partial (avg)	0.72	0.86	0.78
TPO-total (avg)	0.68	0.68	0.51
TPO (union)	0.77	n-a	n-a
TPO (gold)	0.86	n-a	n-a

Table 4: Inter-annotator Agreement

ETPC (+) and MRPC-A are directly comparable as they measure the agreement on the same task (annotation of sense-preserving atomic phenomena). The results show much higher agreement score with respect to both TPO-partial (0.86 against 0.78) and TPO-total (0.68 against 0.51). ETPC (-) measures the agreement on a different task (annotation of non-sense preserving phenomena). The TPO-partial score of ETPC (-) is lower than both ETPC (+) and MRPC-A (0.72 against 0.86 and 0.78 respectively), however the TPO-total score is equal to that of ETPC (+) and much higher than that of MRPC-A. It is interesting to note that there is almost no difference between TPO-partial and TPO-total for ETPC (-) (0.72 against 0.68), while for ETPC (+) and MRPC-A, the difference is significant. The TPO (union) for ETPC (-) shows that 77% of all phenomena are annotated the same way by at least 2 of the annotators. The TPO (gold) indicates that the probability of any of our experts annotating a "gold" example is 86%. Considering the difficulty of the task, the obtained results indicate the high quality of the annotated corpus.

4.4. Annotation of Negation

During the first two steps of the annotation, we identified all sentences that contain negation. For every instance of negation we annotated the negation cues and the scope of negation. 16a and 16b illustrate an example of annotated negation.

16a (Moore had (**no** [negation marker]) immediate comment Tuesday [scope])

16b (Moore (**did not** [negation marker]) have an immediate response Tuesday [scope])

5. The ETPC corpus

This section presents the results of the annotation of the ETPC corpus. Section 5.1. shows the results of annotating non-sense preserving phenomena. Section 5.2. shows the

results of annotating sense preserving phenomena. Section 5.3. discusses the results and the Research Questions, and Section 5.4. lists some applications of ETPC.

5.1. Non-Sense Preserving Atomic Phenomena

Table 5 shows the distribution of the non-sense preserving phenomena. Type Relative Frequency (Type RF) shows the relative distribution of the atomic types. Occurrence Frequency (Type OF) shows the distribution of phenomena per sentence, that is in how many textual pairs each phenomenon can be found[7]. The total number of non-sense preserving phenomena is 3406 in 1901 text pairs.

Type	Type RF	Type OF
Inflectional	0.02%	0.04%
Same Polarity (con)	9.3%	15.5%
Same Polarity (ne)	27.5%	22.5%
Opp Polarity (hab)	2.7%	4.4%
Opp Polarity (con)	0.01%	0.02%
Converse	0.01%	0.02%
Diathesis	0.01%	0.01%
Negation	0.02%	0.03%
Direct/Indirect	0%	0%
Addition/Deletion	52%	65.5%
Semantic based	0%	0%
Non-paraphrase	7.6%	13.7%
Entailment	0.02%	0.04%

Table 5: Distribution of non-sense preserving phenomena

Both Type Relative Frequency (RF) and Occurrence Frequency (OF) indicate that the non-paraphrase portion of the corpus is not well balanced with respect to atomic phenomena. In 260 of the text pairs (13.7%), the annotators selected *"non-paraphrase"* indicating that the two texts were substantially different. In the rest of the pairs, the most common reason for the "non-paraphrase" label at textual level was *"Addition/Deletion"* (52% RF, 65.5% OF), followed by *"Same polarity substitution (named entity)"* (27% RF, 22.5% OF), *"Same polarity substitution (contextual)"* (RF 9,3%, OF 15.5%), and *"Opposite polarity substitution (habitual)"* (RF 2.8%, OF 4.6%). These are the only types with Type Relative Frequency and Occurrence Frequency above 1%, and they constitute over 99% of all non-sense preserving atomic phenomena annotated in the corpus. Six of the atomic phenomena are represented only with a few examples, while two are not represented at all.

5.2. Sense Preserving Atomic Phenomena

Table 6 shows the distribution of sense preserving atomic phenomena in the textual paraphrase and non-paraphrase portions of the corpus[8]. For the textual paraphrase portion, we used the numbers reported by Vila et al. (2015) with partial re-annotation to account for the new types in ETPC. For *"same polarity substitution"*, 35% of the phenomena were re-annotated as *"habitual"*, 47% as *"contextual"*, and 18% as *"named entity"*. For *"opposite polarity substitution"* 21% of the phenomena were *"contextual"* and 79% of the phenomena were *"habitual"*.

Type	Non Paraphrase	Paraphrase
Inflectional	2.13%	2.78%
Modal verb	0.59%	0.83%
Derivational	0.35%	0.85%
Spelling changes	1.30%	2.91%
Same Polarity (hab)	10.55%	8.68%
Same Polarity (con)	11.15%	11.66%
Same Polarity (ne)	7.11%	5.08%
Format	1.06%	1.1%
Opp Polarity (hab)	0%	0.07%
Opp Polarity (con)	0%	0.02%
Synthetic/analytic	7.82%	3.80%
Converse	0.12%	0.20%
Diathesis	0.83%	0.73%
Negation	0%	0.09%
Ellipsis	0.47%	0.30%
Coordination	0.24%	0.22%
Subord. and nesting	1.18%	2.14%
Punctuation	2.72%	3.77%
Direct/Indirect	0.24%	0.30%
Sentence modality	0%	0%
Synt./Disc. structure	1.30%	1.39%
Addition/Deletion	20.04%	25.94%
Change of order	3.08%	3.89%
Semantic	0%	1.53%
Identity	25.02%	17.54%
Non-Paraphrase	2.49%	3.81%
Entailment	0.12%	0.37%

Table 6: Distribution of Sense preserving phenomena in textual paraphrases and textual non-paraphrases

The results show that the distribution of sense-preserving phenomena is relatively consistent between the two portions of the corpus. The most notable differences between the two distributions are the frequencies of *"same polarity substitution (named entity)"*, *"synthetic/analytic"*, *"addition/deletion"*, and *"identity"*. Both distributions are not well balanced in terms of atomic types, with 8 types (*"addition/deletion"*, *"identity"*, *"same polarity substitution (contextual)"*, *"same polarity substitution (habitual)"*, *"synthetic/analytic"*, *"same polarity substitution (named entity)"*, *"change of order"*, and *"punctuation"*) responsible for over 80% of the phenomena.

5.3. Discussion

In this section we briefly discuss the annotation results and the Research Questions that we posed in Section 3.3.
With respect to RQ1 and RQ2, we measured the raw frequency distribution of the sense preserving atomic phenomena in both the paraphrase and non-paraphrase portions of

[7]The sum of all Occurrence Frequencies exceeds 100, as one sentence often contains more than one atomic phenomenon.

[8]At the time of the submission of this paper, the annotation of the non-paraphrase portion was not finished. The reported results are for 500 annotated pairs (about 30% of the corpus). The full figures will be made available at https://github.com/venelink/ETPC

the corpus. We make two important observations from the data. First, the corpus is not well balanced in terms of type distribution in either of the portions. It can be seen in Table 6 that 8 of the types are overrepresented while the rest are underrepresented. This imbalance is even more significant in terms of meta-categories. The structure meta-types "syntax" and "discourse" account for less than 10 % of all types. Second, the raw frequency distribution of atomic phenomena in textual paraphrases and textual non-paraphrases is very similar. This finding suggests that it is the non-sense preserving phenomena that are mostly responsible for the relation at textual level in this corpus. This makes the annotation of the non-sense preserving phenomena even more important for the PI task.

With respect to RQ3, we annotated the *"same polarity substitution (contextual)"* and *"opposite polarity substitution (contextual) "* types in all portions of the corpus. For *"same polarity substitution"*, over 40% of the sense-preserving and over 25% of the non-sense preserving instances were contextual. For *"opposite polarity substitution"*, 21% of the sense-preserving instances were annotated as contextual, while in the non-sense preserving portion we found almost no contextual instances.

With respect to RQ4, we measured the raw frequency distribution of the non-sense preserving phenomena. If we compare it with the distribution of sense preserving phenomena, we can see that the differences are noteworthy and we can easily differentiate between the two distributions. Non-sense preserving phenomena are even less balanced than sense preserving phenomena, with just 4 types responsible for almost all instances. The structure types "syntax" and "discourse" are not represented at all, with all frequent types being either "lexical", "lexico-syntactic", or "other".

Finally, it is worht mentioning that 13% of the sentences in the textual paraphrase portion of the corpus and 12% of the sentences in the textual non-paraphrase portion contain negation. The relative distribution in the paraphrase and in the non-paraphrase portion of the corpus is consistent. The negation scope for each of these sentences has been annotated in a separate layer.

5.4. Applications of ETPC

The ETPC corpus has clear advantages over the currently available PI corpora, and the MRPC in particular. It is much more informative and can be used in several ways.

First, ETPC can be used as a single PI corpus. The annotation with atomic types makes it much more informative for evaluation than any other existing PI corpus. PI systems are currently evaluated in terms of binary Precision, Recall, F1 and Accuracy. ETPC provides the developers with much more detailed information, without requiring any additional work on the developers' side. Knowing which atomic types are involved in the correct and incorrect classification helps the error analysis and should lead to an improvement in the these systems' performance. It also promotes reusability.

Second, ETPC can be used to provide quantitative and qualitative analysis of the MRPC corpus, as we have already shown in section 5.3. By having a detailed statistical analysis of the content of the corpus we can identify possible biases and promote the creation of better and more balanced corpora.

Third, ETPC can be easily split into various smaller corpora built around a certain atomic type or a class of types. Each of them can be used for a new task of Atomic Paraphrase Identification. It can be used to study the nature of the relation between atomic paraphrases and textual paraphrases.

Finally, ETPC can be used to study the role of negation in PI, a research question that, to date, has received very little attention.

6. Conclusions and Future work

In this paper we presented the ETPC corpus - the largest corpus annotated with detailed paraphrase typology to date. For the annotation we used the new Extended Paraphrase Typology, a practically oriented typology of atomic paraphrases. The annotation process included three expert linguists and covered the whole 5801 text pairs from the MRPC corpus. The full corpus is publicly available in two formats: SQL and XML[9].

ETPC is a high quality resource for paraphrase related research and the task of PI. It provides more in-depth analysis of the existing corpora and promotes better understanding of the phenomena, the data, and the task. It also identifies several problems, such as the under-representation of structure based types and the over-representation of lexical based types. ETPC sets an example for the development of new feature-rich corpora for paraphrasing research. It also promotes collaboration between similar areas, such as PI, RTE and Semantic Similarity.

Our work opens several lines of future research. First, the ETPC can be used to re-evaluate existing state-of-the-art PI systems. This detailed evaluation can lead to improvements of the existing PI systems and the creation of new ones. Second, it can be used to create new corpora for paraphrase research, which will be more balanced in terms of type distribution. Third, it can be used to study the nature of the paraphrase phenomenon and the relation between "atomic" and "textual" paraphrases. Finally, the EPT and ETPC can be extended to other research areas, such as lexical and textual entailment, semantic similarity, simplification, summarization, and question answering, among others.

7. Acknowledgments

We would like to thank dr. Horacio Rodriguez and dr. Irina Temnikova for their support and suggestions, Montse Nofre and Marina Bolea for their help with the annotation, and the anonymous reviewers for their feedback and comments.

This work has been funded by Spanish Ministery of Economy Project TIN2015-71147-C2-2, by the CLiC research group (2017 SGR 341), and by the APIF grant of the first author.

[9]We have also made publicly available all complementary data, such as annotation guidelines, screenshots of the interface, detailed statistics, as well as the ETPC_Neg corpus, composed only from the paraphrase and non-paraphrase pairs containing negation (https://github.com/venelink/ETPC).

8. Bibliographical References

Androutsopoulos, I. and Malakasiotis, P. (2010). A survey of paraphrasing and textual entailment methods. *J. Artif. Int. Res.*, 38(1):135–187, May.

Barrón-Cedeño, A., Vila, M., Martí, M. A., and Rosso, P. (2013). Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947.

Benikova, D. and Zesch, T. (2017). Same same, but different: Compositionality of paraphrase granularity levels. In *Proceedings of RANLP 2017*.

Bhagat, R. and Hovy, E. H. (2013). What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Bhagat, R. (2009). *Learning Paraphrases from Text*. Ph.D. thesis, Los Angeles, CA, USA. AAI3368694.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st edition.

Cabrio, E. and Magnini, B. (2014). Decomposing semantic inferences.

Dagan, I., Glickman, O., and Magnini, B. (2006). The pascal recognising textual entailment challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, MLCW'05, pages 177–190, Berlin, Heidelberg. Springer-Verlag.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of Coling 2004*, pages 350–356, Geneva, Switzerland, Aug 23–Aug 27. COLING.

Ganitkevitch, J., Durme, B. V., and Callison-Burch, C. (2013). Ppdb: The paraphrase database. In Lucy Vanderwende, et al., editors, *HLT-NAACL*, pages 758–764. The Association for Computational Linguistics.

Garoufi, K. (2007). Towards a better understanding of applied textual entailment: Annotation and evaluation of the rte-2 dataset. Master's thesis, Saarland University, September.

Sammons, M., Vydiswaran, V. G. V., and Roth, D. (2010). "ask not what textual entailment can do for you...". In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 1199–1208.

Shwartz, V. and Dagan, I. (2016). Adding context to semantic data-driven paraphrasing. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 108–113, Berlin, Germany, August. Association for Computational Linguistics.

Vila, M., Martí, M. A., and Rodríguez, H. (2014). "is this a paraphrase? what kind? paraphrase boundaries and typology. ". pages 205–218.

Vila, M., Bertran, M., Martí, M. A., and Rodríguez, H. (2015). Corpus annotation with paraphrase types: new annotation scheme and inter-annotator agreement measures. *Language Resources and Evaluation*, 49(1):77–105.

Introducing a Lexicon of Verbal Polarity Shifters for English

Marc Schulder*, Michael Wiegand*, Josef Ruppenhofer†, Stephanie Köser*
* Spoken Language Systems, Saarland University, Germany
† Institute for German Language, Mannheim, Germany
marc.schulder@lsv.uni-saarland.de, michael.wiegand@lsv.uni-saarland.de,
ruppenhofer@ids-mannheim.de, skoeser@coli.uni-saarland.de

Abstract

The sentiment polarity of a phrase does not only depend on the polarities of its words, but also on how these are affected by their context. Negation words (e.g. *not*, *no*, *never*) can change the polarity of a phrase. Similarly, verbs and other content words can also act as *polarity shifters* (e.g. *fail*, *deny*, *alleviate*). While individually more sparse, they are far more numerous. Among verbs alone, there are more than 1200 shifters. However, sentiment analysis systems barely consider polarity shifters other than negation words. A major reason for this is the scarcity of lexicons and corpora that provide information on them. We introduce a lexicon of verbal polarity shifters that covers the entirety of verbs found in WordNet. We provide a fine-grained annotation of individual word senses, as well as information for each verbal shifter on the syntactic scopes that it can affect.

Keywords: Sentiment Analysis, Polarity Shifter, Negation, Lexical Semantics, Lexicon

1. Introduction

Polarity shifters are content words that exhibit semantic properties similar to negation. For example, the negated statement in (1) can also be achieved by the verbal shifter *fail* instead of the negation *not*, as shown in (2).

(1) Peter did **not** pass the exam.
(2) Peter **failed**$_{\textbf{\textit{shifter}}}$ to pass the exam.

As with negation words, polarity shifters change the polarity of a statement. This can happen to both positive and negative statements. In (3) the positive polarity of *scholarship* is shifted by *denied*, resulting in a negative polarity for the phrase. Conversely, the overall polarity of (4) is positive despite the negative polarity of *pain*.

(3) She was [**denied**$_{\textbf{\textit{shifter}}}$ the [scholarship]$^{+}$]$^{-}$.
(4) The new treatment has [**alleviated**$_{\textbf{\textit{shifter}}}$ her [pain]$^{-}$]$^{+}$.

Polarity shifting is also caused by other content word classes, such as nouns (e.g. *downfall*) and adjectives (e.g. *devoid*). However, this work focusses on verbs, due to their importance as minimal semantic units, far-reaching scopes and potential basis for nominal shifter lexicons (see §2.2.).
Knowledge of polarity shifting is important for a variety of tasks, especially sentiment analysis (Wiegand et al., 2010; Liu, 2012; Wilson et al., 2005), as well as relation extraction (Sanchez-Graillet and Poesio, 2007) and textual entailment recognition (Harabagiu et al., 2006).
The majority of research into polarity shifting for sentiment analysis has focussed on negation words (Wiegand et al., 2010; Schouten and Frasincar, 2016; Pak and Paroubek, 2010). Negation words (e.g. *not*, *no*, *never*) are mostly function words, of which only a small number exists, so exhaustive coverage is comparatively simple. Content word classes, such as verbs, are considerably more difficult to cover comprehensively due to their sheer number. For example, WordNet (Miller et al., 1990) contains over 10k verbal lemmas. Most verbs are also far less frequent than common negation words, making individual verbal shifters seem less important. However, overall, verbal shifter lemmas occur 2.6 times as often as negation words (see §4.).
Most existing resources on negation and polarity shifting cover few to no instances of verbal shifters (see §2.3.). To remedy this, we introduce a **complete lexicon of verbal shifters** with annotations of polarity shifters and their shifting scope for each word sense.
Our **contributions** are as follows:

(i) A complete lexicon of verbal polarity shifters, covering all verbs found in WordNet 3.1.
(ii) A fine grained annotation, labelling every sense of a verb separately.
(iii) Annotations for shifter scope, indicating which parts of a sentence are affected by the shifting.

The entire dataset is **publicly available**.[1]

2. Background

In this section we will provide a formal definition of polarity shifters (§2.1.), motivate our focus on verbal shifters (§2.2.) and discuss related work (§2.3.).

2.1. Polarity Shifters

The notion of valence or polarity shifting was brought to broad awareness in the research community by the work of Polanyi and Zaenen (2006). Those authors drew attention to the fact that the basic valence of individual lexical items may be shifted in context due to (a) the presence of certain other lexical items, (b) the genre type and discourse structure of the text and (c) cultural factors. In subsequent research, the term *shifter* has since mostly been applied to the case of lexical items that influence polarity. Further, the notion of *shifting* is most prototypically used for situations where a discrete polarity *switch* occurs between the classes *positive*, *negative* and *neutral*. However, for other authors,

[1] https://github.com/marcschulder/lrec2018

including Polanyi and Zaenen (2006), intensification (e.g. *very disappointing*) and downtoning (*e.g. somewhat disappointing*) of polar intensity also falls within the scope of shifting. We partially follow this view in that we consider downtoning to be shifting, as it moves the polarity of a word in the opposite direction, i.e. making a positive expression less positive (*e.g. hardly satisfying*) and a negative one less negative (*e.g. slightly problematic*). We do not consider intensifiers as shifters, as they support the already existing polarity.

In most research, shifters are commonly illustrated and enumerated rather than formally defined. Polanyi and Zaenen (2006) for instance list negation words, intensifiers, modals and presuppositional items as lexical contextual polarity shifters.

Setting aside downtoners for now, the common denominator of shifting is negation. Negation marks contexts in which a situation that the speaker expected fails to occur or hold. When this situation is part of a binary opposition (*dead – alive*), one can firmly conclude that the complementary state of affairs holds (*not dead* $\Rightarrow$ *alive*). In cases where the negation affects a scalar notion, which is common in evaluative contexts, the understanding that arises depends on which kinds of scalar inferences and default assumptions are made in the context (Paradis and Willners, 2006). Thus, *not good* denies the applicability of an evaluation in the region of good or better, but leaves open just how far in the direction of badness the actual interpretation lies: *"It wasn't good"* may be continued with *"but it was ok"* to yield a neutral or mildly positive evaluation or with *"in fact, it was terrible"* to yield a strongly negative one.[2]

While downtoners (e.g. *somewhat*) applied to scalar predicates such as *good* do not directly express contradiction, they do give rise to negative entailments and inferences. Moreover, the structure of scales intrinsically provides shifting. Thus, while something being *good* allows it to be even more positive (*"The movie was good. In fact, it was excellent."*), something being *somewhat good* bounds its positiveness and opens up more negative meanings (*"The performance was somewhat good, but overall rather disappointing"*). Considering these properties of scales, one can see shifting at work even in the case of downtoning.

2.2. Verbal Shifters

While the inclusion of shifting and scalar semantics in semantic representations is not limited to lexical items of particular parts-of-speech – we also find shifter adjectives (e.g. *devoid*) and adverbs (e.g. *barely*) – we limit our work to verbal shifters for several reasons. As shown by the work of Schneider et al. (2016), verbs, together with nouns, are the most important minimal semantic units in text and thus are prime candidates for being tackled first. Verbs are usually the main syntactic predicates of clauses and sentences and thus verbal shifters can be expected to project far-reaching scopes. Most nominal shifters (e.g. *failure*, *loss*), on the other hand, have morphologically related verbs (e.g. *fail*, *lose*) and we expect that this connection can be exploited to spread shifter classification from verbs to nouns in the future. Related to this, the grammar of verbs, for instance with respect to the diversity of scope types, is more complex than that of nouns and so we expect it to be easier to project from verbs to nouns rather than in the opposite direction.

2.3. Related Work

Existing lexicons and corpora that cover polarity shifting focus almost exclusively on negation words. The most complex negation lexicon for sentiment analysis (Wilson et al., 2005) includes a mere 12 verbal shifters. In contrast, our resource covers over 1200 verbal shifter lemmas.

Corpora used as training data for negation processing, such as the Sentiment Treebank (Socher et al., 2013) or the BioScope corpus (Szarvas et al., 2008), are fairly small datasets, so only the most frequent negation words appear. The BioScope corpus, for example, contains only 6 verbal shifters (Morante, 2010). Schulder et al. (2017) show that state-of-the-art systems trained on such data do not reliably detect polarity shifting and should profit from explicit knowledge of verbal shifters.

The only work to date that covers a larger number of verbal shifters is Schulder et al. (2017), who annotate a sample of the English verbs found in WordNet for whether they exhibit polarity shifting. They start by manually annotating an initial 2000 verbs. These verbs are used to train an SVM classifier using linguistic features and common language resources. The classifier is then run on the remaining WordNet verbs to bootstrap a list of additional likely shifters. This list is then checked by a human annotator to detect false positives. Combining the initial annotation and the result of the bootstrapping process, they create a list of 3043 verbs.

While the lexicon by Schulder et al. (2017) is an important step towards full coverage of verbal polarity shifters, there are several aspects that we seek to improve upon. First of all, their lexicon covers less than a third of the verbs found in WordNet, likely missing a number of verbal shifters. Schulder et al. (2017) argue that their bootstrap process should cover the majority of shifters, however, this would mean that only 9% of all verbs are shifters.[3] Their initial annotation of 2000 randomly selected verbs puts the shifter ratio at 15% instead.

Another issue with their lexicon is that it only labels lemma forms, but does not differentiate between word senses. Many verbs do not actually exhibit shifting in all of their senses, so this information will be important for contextual classification.

Lastly, they forgo the question of shifter scope, i.e. which argument of a verb can be affected by its polarity shift.

[2] Note that the example also illustrates how distinguishing between items that induce a switch between polarities and others that affect intensity without changing overall polarity is an idealization. Simple syntactic negation of a polar adjective may influence intensity as well as polarity (e.g. *not terrible* $\neq$ *excellent*) (Kiritchenko and Mohammad, 2016).

[3] They find 980 shifter lemmas among the 3043 verbs that are annotated by a human annotator. An additional 7538 verbs are assumed to not be shifters without human confirmation.

3. Data

We treat this annotation effort as a binary labelling task where a word can either cause polarites to shift or not. However, instead of assigning a single label to an entire verb lemma, as Schulder et al. (2017) did, we label individual word senses. We outline the rationale for this in §3.1. In addition we explicitly specify the syntactic scope of the shifting. This is motivated and explained in §3.2. §3.3. describes the annotation process. §3.4. describes the data format of our main lexicon. Based on this main lexicon we also derive two auxiliary lexicons in §3.5., providing complete labelled lists of all WordNet verb lemmas and all WordNet verb synsets respectively.

3.1. Word Senses

Many words that shift polarities only do so for some of their word senses. For example, *mark down* acts as a shifter in (5), where it has the sense of *"reducing the value of something"*, but the sense of *"writing something down to have a record of it"* in (6) causes no shifting. In our work we found that among shifter lemmas with multiple word senses, only 23% caused shifting in each of their senses. An annotation on the basis of individual word senses is therefore required.

(5) The agency [**marked down**$_{\textbf{\textit{shifter}}}$ [their assets]$^{+}$]$^{-}$.

(6) She [**marked down**$_{\textbf{\textit{no shifter}}}$ [his confession of guilt]$^{-}$]$^{-}$.

To differentiate the senses of a verb, we use its synset affiliations found in WordNet. Words within the same synset share a shifter label. Shifter scope, on the other hand, can differ among words of the same synset (see §3.2.). The annotation introduced in §3.3. is therefore applied to individual lemma-sense pairs to capture the best of both worlds.

3.2. Shifter Scope

A verbal shifter usually only affects the parts of a sentence that are syntactically governed by the verb through its valency. However, not every argument of a verbal shifter is subject to polarity shifting. Which argument is affected by polarity shifting depends on the verb in question. In (7), *surrender* shifts only the polarity of its subject, but does not affect the object. Conversely, *defeat* only shifts its object in (8). The polarity of the subject of *defeat* does not play a role in this, as can be seen in (9).

(7) [[The villain]$^{-}$ **surrendered**]$^{+}$ [to the hero]$^{+}$.

(8) [The villain]$^{-}$ [**defeated** [the hero]$^{+}$]$^{-}$.

(9) Chance [**defeated** [the hero]$^{+}$]$^{-}$.

In the following, we present the shifting scopes we observed, their abbreviation in the annotation and examples for each:

Subject (`subj`): The verbal shifter affects its subject, e.g. *"[[His confidence]$^{+}_{subj}$* ***decreased****]$^{-}$."*

Direct Object (`dobj`): The verbal shifter affects its direct object, e.g. *"[The storm]$_{subj}$ [**ruined** [their party]$^{+}_{dobj}$]$^{-}$."*

Prepositional Object (`pobj_*`): The verbal shifter affects the object within a prepositional phrase. The preposition in question is included in the annotation. For example, *shield* as in *"[The wall]$_{subj}$ [**shielded** them$_{dobj}$ [from the explosion]$^{-}_{pobj}$]$^{+}$."* is annotated as `pobj_from` and *reimburse* as in *"[The company]$_{subj}$ [**reimbursed** him$_{dobj}$ [for his expenses]$^{-}_{pobj}$]$^{+}$."* as `pobj_for`.

Clausal complement (`comp`): The verbal shifter affects a clausal complement, such as infinitive clauses like *"He [**failed** [to pass the exam]$^{+}_{comp}$]$^{-}$."* or gerunds like *"She [**stopped** [using drugs]$^{-}_{comp}$]$^{+}$."*

The given scopes assume that verb phrases are in their active form. In passive phrases, subject and object roles are inverted. To avoid this issue, sentence structure normalization should be performed before computing shifter scope.
Synsets in WordNet only capture the semantic similarity of words, but almost no syntactic properties (Ruppenhofer and Brandes, 2015). The shifter scope of a verb depends on its syntactic arguments, which can differ between verbs of the same synset. For example, *discard* and *dispose* share the sense *"throw or cast away"*, but while *discard* shifts its direct object (10), *dispose* requires a prepositional object (11). For this reason we annotate lemma-synset pairs individually, instead of assigning scope labels to an entire synset.

(10) He [**discarded** [the evidence]$^{+}_{dobj}$]$^{-}$.

(11) He [**disposed** [of the evidence]$^{+}_{pobj}$]$^{-}$.

We also consider cases where a verbal shifter has more than one potential scope for the same lemma-sense pair. For example, *infringe* can shift its direct object or various prepositional objects, as seen in (12) – (14). Therefore, *infringe* receives the scope labels `dobj`, `pobj_on` and `pobj_upon`.

(12) The inquiry [**infringes** [people's privacy]$^{+}_{dobj}$]$^{-}$.

(13) The inquiry [**infringes** [on people's privacy]$^{+}_{pobj}$]$^{-}$.

(14) The inquiry [**infringes** [upon people's privacy]$^{+}_{pobj}$]$^{-}$.

A verbal shifter will only ever shift the polarity of one of its scopes. Which scope is affected by the shifting depends on the given sentence.

3.3. Annotation

The entire dataset was labelled by an expert annotator with experience in linguistics and annotation work. To measure inter-annotator agreement, a second annotator re-annotated 400 word senses for their shifter label. They achieved an agreement of $\kappa = 0.73$, indicating substantial agreement (Landis and Koch, 1977).
The annotation progressed as follows: Given a complete list of WordNet verb lemmas, the annotator would inspect one lemma at a time. For this lemma, all senses were looked up. For each such lemma-sense pair, the annotator decided whether it is a shifter or not. Decisions were based on the sense definition of the synset and whether sentences using

	Shifters		Non-shifters		Total
	#	%	#	%	#
Lemmas	1220	11.53	9357	88.47	10577
Synsets	924	6.88	12502	93.12	13426
LS Pairs	2131	8.88	21855	91.12	23986

Table 1: The ratio of verbal shifters in WordNet. Lemmas are counted as shifters when at least one sense is a shifter. *"LS Pairs"* represents lemma-synset pairs.

this sense of the lemma cause shifting. If a word sense was labelled as a shifter, it was subsequently also annotated for its potential shifter scopes.

In cases where label conflicts between different lemma-sense pairs of the same sense were encountered, these labels were reconsidered. This introduced an additional robustness to the annotation as it let the annotator revisit challenging cases from a new perspective.

The resulting list of lemma-sense pairs provides more fine-grained information than either an annotation for only word lemmas or only synsets could (see §3.1. and §3.2.).

3.4. Main Lexicon File Format

We provide our main lexicon as a comma-separated value (csv) file in which each line represents a specific lemma-sense-scope triple of a verbal shifter. Each line follows the format "`LEMMA,SYNSET,SCOPE`". The fields are defined as follows:

`LEMMA`: The lemma form of the verb.

`SYNSET`: The numeric identifier of the synset, commonly referred to as *offset* or *database location*. It consists of 8 digits, including leading zeroes (e.g. 00334568).

`SCOPE`: The scope of the shifting. Given as `subj` for subject position, `dobj` for direct object position and `comp` for clausal complements. Prepositional object positions are given as `pobj_*`, where `*` is replaced by the preposition in question, e.g. `pobj_from` for objects with the preposition *"from"* or `prep_of` for the preposition *"of"*.

When a lemma has multiple word senses, a separate entry is provided for each lemma-sense pair. When a lemma-sense pair has multiple potential shifting scopes, a separate entry is provided for each scope. Any combinations not provided are considered not to exhibit shifting. Take, for example, the set of entries for *"blow out"*:

(15)
```
blow out,00436247,subj
blow out,02767855,dobj
```

It tells us that *blow out* in the sense 00436247 (*"melt, break, or become otherwise unusable"*) is a shifter that affects its subject. The sense 02767855 (*"put out, as of fires, flames, or lights"*) also exhibits shifting, but this time affects the direct object. It is, however, not a shifter for sense 02766970 (*"erupt in an uncontrolled manner"*). For an example of multiple scopes for the same word sense, consider *cramp*:

(16)
```
cramp,00237139,dobj
cramp,00237139,pobj_in
```

	subj	dobj	pobj_*	comp	Total
Frequency	402	1574	212	32	2220

Table 2: Distribution of shifting scopes for individual word senses. Total is higher than number of lemma-synset pairs (Table 1) as 4% of shifters have multiple potential scopes.

Its sense 00237139 (*"prevent the progress or free movement of"*) can shift the polarity of either its direct object (e.g. *"it cramped his progress"*) or that of a prepositional object with the preposition *"in"* (e.g. *"he was cramped in his progress"*). The three other senses of *cramp* given by WordNet are not considered shifters.

3.5. Auxiliary Lexicons

Our main lexicon is labelled at the lemma-sense pair level to provide the most fine-grained level of information possible. It can, however, easily applied to more coarse-grained applications. As a convenience, we provide lemma- and synset-level auxiliary lexicons that list all WordNet lemmas and all WordNet synsets, respectively, accompanied with their shifter label. A lemma is labelled as a shifter if at least one of its senses is considered a shifter in our main lexicon. Similarly, synsets are labelled as shifters if at least one of its lemma-realizations is a shifter.

4. Statistics

In Table 1 we present the ratio of shifters among the verbs contained in WordNet. While only about 10% of verbs are shifters, this still results in 1220 lemmas and 924 synsets, more than covered in any other resource (see §2.3.).

49% of verbs in WordNet are polysemous, i.e. they have multiple meanings. Among verbal shifters, this ratio is considerably higher, reaching 73%. Of these, only 23% are shifters in all of their word senses.

To get an idea of how common verbal shifters are in actual use, we computed lemma frequencies over the *Amazon Product Review Data* corpus (Jindal and Liu, 2008), which comprises over 5.8 million reviews. We found this corpus suitable due to its size, sentiment-related content and use in related tasks (Schulder et al., 2017).

We observe 1163 different verbal shifter lemmas with an overall total of 34 million occurrences. Correcting for non-shifter senses of shifter lemmas[4], we still estimate 13 million occurrences, accounting for 5% of all verb occurrences in the corpus. To compare, the 15 negation words found in the valence shifter lexicon by Wilson et al. (2005) occur 13 million times as well. While the frequency of individual negation (function) words is unsurprisingly higher, the total number of verbal shifter occurrences highlights that verbal shifters are just as frequent and should not be ignored.

Statistics on the distribution of shifter scopes can be found in Table 2. 74% of verbal shifters have a direct object

[4]Due to the lack of robust word-sense disambiguation tools, we estimated the likelihood of a lemma instance being a shifter based on its ratio of shifter word senses. A lemma with 3 shifter senses and 1 non-shifter sense would, therefore, have a likelihood of 0.75 to be a shifter.

scope and 10% a prepositional object scope. Among these, *"from"* is the most common preposition at 51%, followed by *"of"* with 22%. 19% shift the polarity of their subject and only 1.5% shift that of a clausal complement. This distribution shows that shifting cannot be trivially assumed to always affect the direct object and that explicit knowledge of shifter scopes will be useful for judging the polarity of a phrase.

5. Conclusion

We introduced a lexicon of verbal polarity shifters that covers the entire verb vocabulary of WordNet. Our annotation labels each individual word sense of a verb, providing more fine-grained information than annotations on the lemma-level would. In addition, we also label the syntactic scopes of each verbal shifter that can be affected by the shifting.

This is a clear improvement over the list of verbal shifters provided by Schulder et al. (2017), which only provides labels at the lemma-level rather than for individual word senses and gives no information regarding shifting scope. It also only has human expert annotation for 30% of the verb vocabulary of WordNet, as opposed to our full coverage.

We hope this resource will help improve fine-grained sentiment analysis systems by providing explicit information on where polarities may shift in a sentence.

We also hope our work will encourage the creation of similar polarity shifter lexicons for nouns and adjectives. As they are more numerous than verbs (WordNet contains 20k adjectival and 110k nominal lemmas), creating such resources will come with its own challenges, especially in the case of nouns.

Acknowledgements

The authors were partially supported by the German Research Foundation (DFG) under grants RU 1873/2-1 and WI 4204/2-1.

6. Bibliographical References

Harabagiu, S., Hickl, A., and Lacatusu, F. (2006). Negation, Contrast and Contradiction in Text Processing. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*.

Jindal, N. and Liu, B. (2008). Opinion Spam and Analysis. In *Proceedings of the International Conference on Web Search and Data Mining*.

Kiritchenko, S. and Mohammad, S. M. (2016). The Effect of Negators, Modals, and Degree Adverbs on Sentiment Composition. In *Proceedings of WASSA*.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Liu, B. (2012). Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. (1990). Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3:235–244.

Morante, R. (2010). Descriptive Analysis of Negation Cues in Biomedical Texts. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Paradis, C. and Willners, C. (2006). Antonymy and Negation: The Boundedness Hypothesis. *Journal of Pragmatics*, 38(7):1051–1080.

Polanyi, L. and Zaenen, A. (2006). Contextual Valence Shifters. In James G. Shanahan, et al., editors, *Computing Attitude and Affect in Text: Theory and Applications*, pages 1–10. Springer Netherlands.

Ruppenhofer, J. and Brandes, J. (2015). Extending effect annotation with lexical decomposition. In *Proceedings of the Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA@EMNLP)*, pages 67–76, Lisboa, Portugal.

Sanchez-Graillet, O. and Poesio, M. (2007). Negation of protein–protein interactions: analysis and extraction. *Bioinformatics*, 23(13):i424–i432.

Schneider, N., Hovy, D., Johannsen, A., and Carpuat, M. (2016). SemEval-2016 Task 10: Detecting Minimal Semantic Units and their Meanings (DiMSUM). In *Proceedings of the International Workshop on Semantic Evaluation (SemEval@NAACL-HLT)*, pages 546–559.

Schouten, K. and Frasincar, F. (2016). Survey on Aspect-Level Sentiment Analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(3):813–830.

Schulder, M., Wiegand, M., Ruppenhofer, J., and Roth, B. (2017). Towards Bootstrapping a Polarity Shifter Lexicon using Linguistic Features. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, Taipei, Taiwan.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Szarvas, G., Vincze, V., Farkas, R., and Csirik, J. (2008). The BioScope Corpus: Annotation for Negation, Uncertainty and their Scope in Biomedical Texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing (BioNLP@ACL-HLT)*.

Wiegand, M., Balahur, A., Roth, B., Klakow, D., and Montoyo, A. (2010). A Survey on the Role of Negation in Sentiment Analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Joint Conferences on Human Language Technology and on Empirical Methods in Natural Language Processing (HLT/EMNLP)*.

JFCKB: Japanese Feature Change Knowledge Base

Tetsuaki Nakamura[†], **Daisuke Kawahara**[†‡]
†Kyoto University, ‡JST PRESTO
Yoshida-Honmachi, Sakyo-ku, Kyoto, Japan
tnakamura@nlp.ist.i.kyoto-u.ac.jp, dk@i.kyoto-u.ac.jp

Abstract

Commonsense knowledge plays an essential role in our language activities. Although many projects have aimed to develop language resources for commonsense knowledge, there is little work focusing on connotational meanings. This is because constructing commonsense knowledge including connotational meanings is challenging. In this paper, we present a Japanese knowledge base where arguments in event sentences are associated with various feature changes caused by the events. For example, "my child" in "my wife hits my child" is associated with some feature changes, such as *increase in pain*, *increase in anger*, *increase in disgust*, and *decrease in joy*. We constructed this knowledge base through crowdsourcing tasks by gathering feature changes of arguments in event sentences. After the construction of the knowledge base, we conducted an experiment in anaphora resolution using the knowledge base. We regarded anaphora resolution as an antecedent candidate ranking task and used Ranking SVM as the solver. Experimental results demonstrated the usefulness of our feature change knowledge base.

Keywords: emotion, commonsense knowledge, knowledge base, crowdsourcing, anaphora resolution

1. Introduction

Commonsense knowledge plays an essential role in our language activities. Such knowledge also plays an important role for computers to understand texts. In the area of language resource development, most studies on commonsense knowledge focus on denotational meanings such as predicate-argument structures (Baker et al., 1998; Palmer et al., 2005). Meanwhile, there are few studies focusing on connotational meanings (Nakamura and Kawahara, 2016; Rashkin et al., 2016). Even in such studies, abstract features such as polarities are used. However, concrete knowledge (that is, fine-grained knowledge) is better for computers than abstract knowledge because events actually cause various concrete feature changes to participants in the events or those who know of the events.

There are two approaches to acquiring commonsense knowledge. One is the automatic acquisition approach; the other is the manual acquisition approach. The automatic acquisition approach uses machine learning techniques and pattern matching methods. Although this approach is useful when the amount of data to be acquired is extremely large, the quality of acquired knowledge might be low. Moreover, the existence of *reporting bias* in texts (Gordon et al., 2010; Gordon and Durme, 2013) shows that infrequent events tend to appear more in texts than quotidian events. The manual approach is useful for gathering subjective information (such as emotional information) and quotidian commonsense knowledge, which are difficult to acquire automatically. This approach may use a fully manual technique (at a very early stage of artificial intelligence studies) or collective intelligence (e.g., crowdsourcing and games with a purpose).

In this study, we aim to solve some problems in previous work on connotational meanings by constructing a large-scale knowledge base of concrete feature changes of arguments in event sentences with a controlled granularity. For example, "my child" in "my wife hits my child" is associated with some feature changes such as *increase in pain*, *increase in anger*, *increase in disgust*, and *decrease in joy*. The proposed knowledge base is for Japanese. We combined automatic and manual approaches (collective intelligence) to exploit the merits of both techniques. After the construction of our feature change knowledge base, we conducted an experiment on anaphora resolution using the knowledge base. In this experiment, we used a Japanese translated version of the Winograd Schema Challenge (WSC) dataset (JWSC) (Levesque, 2011; Shibata et al., 2015).

2. Related Work

There are many studies focusing on events themselves. FrameNet (Baker et al., 1998) is a corpus in which deep cases (semantic roles) of predicates are defined. In FrameNet, each argument in an example sentence is labeled with a deep case. ConceptNet (Speer and Havasi, 2012) is a large semantic network constructed in the Open Mind Common Sense project (Singh et al., 2012). It is composed of relationships between concepts, where concepts are noun, verb or adjectival phrases.

To date, some knowledge bases have been developed to capture connotational meanings of predicates. Lexical Conceptual Structure (LCS) is a useful way to describe behavior of arguments in event sentences (Jackendoff, 1983). In LCS, not only behaviors of arguments but also relationships between arguments in event sentences are described, for each verb. However, LCS focuses only on direct and explicit information. It is necessary to develop a method to process indirect and implicit information because such knowledge is frequently used. Moreover, there is a possibility that many predicates are excessively generalized because the information used in LCS is abstract. WordNet-Affect (Strapparava and Valitutti, 2004) is an extended version of WordNet (Fellbaum, 1998). Although synsets of WordNet are associated with some emotions

in WordNet-Affect, it is hard to know who/what is associated with the emotions in events. *Connotation Frames* are a knowledge base of emotional implications of events (Rashkin et al., 2016). In Connotation Frames, implications of an event are represented by a set of polarities categorized into five types (writers' perspective, entities' perspective, effect, value, and mental state). To construct Connotation Frames, crowdsourcing and a news corpus were used. Although this knowledge base is useful for understanding information conveyed by event descriptions, it is hard to know details of the emotions associated with events because emotions themselves and associated behaviors are abstracted by polarities in the knowledge base.

As for connotational meanings, especially for mental states, there are many studies that exploit traditional emotion models proposed by psychologists (Tokuhisa et al., 2008; Tokuhisa et al., 2009; Hasegawa et al., 2013; Vu et al., 2014). In these studies, Ekman's Big Six Model (Ekman, 1992) or Plutchik's wheel of emotions (Plutchik, 1980) are used to automatically extract emotion knowledge from large corpora or Web documents. However, as pointed out in Gui et al. (2017), most natural language processing studies on emotion focus on emotion classification (Tokuhisa et al., 2008; Tokuhisa et al., 2009; Li et al., 2015; Zhou et al., 2016) and emotion information extraction (Vu et al., 2014). There are few studies on emotion cause extraction based on relationships between events and emotions caused by the events (Ghazi et al., 2015; Gui et al., 2016; Gui et al., 2017). These studies use explicit keywords in texts to recognize relationships between emotions and events.

There are many studies on automatic acquisition of relationships between events (Chambers and Jurafsky, 2009; Chambers and Jurafsky, 2010; Vanderwende, 2005; Shibata et al., 2014). Knowledge bases constructed in these studies are useful to determine what events happen after other events. However, these studies do not focus on motivations for events or effects caused by the events. As described in the previous section, this approach cannot guarantee quotidian connotational meanings from texts because of the reporting bias. Therefore, it is important to construct knowledge bases of such commonsense knowledge by manual intervention as well as through automatic acquisition.

3. Design of JFCKB

Our final goal is to construct a knowledge base of various feature changes of arguments in event sentences with the controlled granularity and to develop various methods for deep understanding of texts. To achieve the goal, we propose a knowledge base structure shown in Table 1. In this structure, each argument in an event sentence is associated with various feature changes caused by the event. For example, in the case of "my wife hits my child," "my child" is associated with changes of some features such as *increase in pain*, *increase in anger*, *increase in disgust*, *decrease in joy*, and *decrease in trust*. We gathered such information through collective intelligence (i.e., crowdsourcing). Event sentences to be stored in our knowledge base are created based on some knowledge bases automatically constructed from large-scale Web documents. Hereafter, we call our knowledge base *JFCKB* (Japanese Feature Change Knowledge Base). JFCKB is a Japanese knowledge base.

Sentence	**Case (word)**	**Probability**
My wife hits my child	*ga* (nominative) (wife)	*joy* (+, −, UNC) = (0, 1, 0)
		:
	wo (accusative) (child)	*anger* (+, −, UNC) = (1, 0, 0)
		:
	ni (dative) (NULL)	N/A
	reader (NULL)	*disgust* (+, −, UNC) = (0.99, 0, 0.01)
		:

Table 1: Example of the proposed knowledge base structure (JFCKB). This knowledge base is for Japanese. In JFCKB, each sentence has various feature changes for three cases (*ga* case, *wo* case, and *ni* case) and those of *readers*. The three cases are Japanese language specific syntactic roles. The *ga*, *wo*, and *ni* cases roughly correspond to nominative, accusative, dative, respectively. *Readers* are not arguments in the sentence. In the "Probability" column, symbols +, −, and UNC denote *increased*, *decreased*, and *unchanged*, respectively.

We pay attention to the controlled granularity for feature definition. Many studies suggest that it is important to focus on the types of changes caused by events to recognize the events. In infant cognitive development studies, there are several reports that even infants use information about the basic features of participants in events to understand the events (Massey and Gelman, 1988; Baillargeon et al., 1989; Spelke et al., 1995). In cognitive linguistics, *basic level category* is an extremely important concept (Rosch et al., 1976; Taylor, 1995). Basic level categories minimize attributes shared with other categories. Members belonging to a basic level category maximize attributes shared with other members of that same category. For example, "dog" is a basic level category while "Great Pyrenees" and "German shepherd" are not. Many cognitive linguists believe that our languages are built around basic level categories. Therefore, the granularity of knowledge is an important aspect of our language activity that needs to be captured.

Although there are many features altered by events, such features can be roughly classified into at least four categories: (1) physical features (such as size and temperature); (2) mental features (such as anger and joy); (3) sensory features (such as pain and sleepiness), and (4) relational features (such as position and social relation). According to these categories, we assumed 47 features shown in Table 3. These features represent the *basic level categories*

Verb: case frame ID	Case	Word
yaku: yaku1 (bake)	*ga* (nominative)	*watashi* (I): 114, *haha* (mom): 75, *musume* (daughter): 74, ...
	wo (accusative)	*pan* (bread): 54076, *ke-ki* (cake): 31693, *niku* (meat): 14059, ...
	de (tools/ingredients)	*koubo* (yeast): 888, *be-kari-* (bakery): 768, *o-bun* (oven): 515, ⋯
yaku: yaku2 (have difficulty)	*ga* (nominative)	*mina* (all persons): 23, *sensei* (teacher): 11, *hito* (person): 8, ...
	wo (accusative)	*te* (hand): 26449
	ni (dative)	*kodomo* (child): 168, *musuko* (son):108, ...
yaku: yaku3 (burn)	*ga* (nominative)	*daitouryou* (president): 1, śhidousya (mentor): 1, ...
	ni (dative)	CD:13812, DVD:12200, ...

Table 2: Case frame examples. Each row denotes one case frame. In the "Word" column, each number denotes the frequency of the noun in the Web corpus.

as closely as possible. This decision was based on a traditional emotion study (Plutchik, 1980), Japanese thesauri (Ikehara, 1997; NINJAL, 2004), sentiment analysis studies (Tokuhisa et al., 2008; Tokuhisa et al., 2009), and features used in the VerbCorner project (Hartshorne et al., 2014). Although our final version of the proposed knowledge base will have all the features in Table 3, at present, emotional and sensory features in the table have been mainly investigated through crowdsourcing tasks described in the following section.

4. Construction of JFCKB

In our previous work (Nakamura and Kawahara, 2016), we constructed a trial version of a knowledge base in which each argument of an event sentence was associated with various feature changes caused by the events. As a result of a subjective evaluation experiment, it was shown that such feature changes can be appropriately acquired using crowdsourcing. In this trial version, 857 sentences (including 144 verbs (types) and 1,788 arguments (tokens) (742 types), 391 case frames (types)) were used.

In this study, we constructed a full version of the knowledge base and extended the dataset as follows: (1) for the representation manner of behaviors, we used only triples (*increased*, *decreased*, and *unchanged*) whose values are probabilities. In the trial version, some feature changes were represented by the triples while the other feature changes were represented by pairs (*changed* and *unchanged*) whose values are probabilities. (2) In addition to event sentences based on the 200 most frequent verbs in the *Kyoto University Web Document Leads Corpus* (*KWDLC*) (Hangyo et al., 2012)[1] used in our previous work, we used sentences from other Japanese language resources. The resources are *Kyoto University Case Frames* (*KUCF*) (Kawahara and Kurohashi, 2006)[2] and the Japanese translated version of the WSC (*JWSC*) dataset.

KWDLC is a Japanese text corpus that comprises 5,000 documents (15,000 sentences) with annotations of morphology, named entities, dependencies, predicate-argument structures including zero anaphora and coreferences. KUCF is a database of case frames automatically constructed from a corpus of 10 billion Japanese sentences taken from the Web. Case frames describe what kinds of nouns are related to each verb. Many Japanese verbs have some meanings. Examples are shown in Table 2. In KUCF, each case frame is composed of case frame ID, verb, cases, nouns filled in the cases, frequencies of the nouns in the Web corpus. KUCF has about 110,000 predicates with 5.4 case frames on average for each predicate. WSC (JWSC) dataset is basically composed of two sentences including one anaphor, two antecedent candidates, and a correct antecedent.

Category	Sub category	Feature
physical	form	**length**, **size**, width, thickness (around) thickness (depth)
	color	redness, orangeness yellowness, greenness blueness, purpleness brownness, whiteness blackness, brightness
	touch	**temperature**, rigidness roughness, stickiness
	smell	goodness, badness
	sound	silence
	taste	sweetness, sourness bitterness, astringency hotness (not temperature)
	density	denseness
	amount	quantity
mental	emotion	**joy**, **trust**, **surprise** **disgust**, **fear**, **sadness** **anger**, **anticipation**
	evaluation	polarity
sensory	sensory	**pain**, **sleepiness** **tiredness**
relation	relation	interaction, possession physical contact physical force existence social relationship
	position	closeness

Table 3: Features assumed in this study. These features were decided by considering various studies such as traditional psychological studies, studies on cognitive development of infants, Japanese thesauri, sentiment analysis studies, and VerbCorner project. Features with bold fonts have been investigated so far.

[1] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KWDLC
[2] http://www.gsk.or.jp/en/catalog/gsk2008-b/

The construction of JFCKB was achieved using crowdsourcing. The details are described in the next subsections.

4.1. Phase 1: Event Sentence Creation

Event sentences in JFCKB are representative sentences of case frames of verbs in KUCF. The sentence creation procedure was as follows. Step 1: Selection of verbs. We used the 200 most frequent verbs in KWDLC, the 1,000 most frequent verbs in KUCF, and all verbs in the JWSC dataset. Step 2: Representative argument selection. For each case (syntactic role) of each case frame of each verb, the most frequent argument was used as the representative argument of the case. In the case of KWDLC, the two most frequent arguments were used. We used *ga* case, *wo* case, and *ni* case as arguments. These cases are Japanese grammatical syntactic roles and roughly correspond to nominative, accusative, and dative, respectively. Step 3: Representative sentence creation. Sentences were created by combining the verbs with the three types of representative arguments. Step 4: Incomprehensible sentence exclusion. Sentences difficult to understand were pruned based on a crowdsourcing task. The crowdsourcing workers were asked whether they could understand the sentences presented. The comprehensibility of each sentence was judged by ten workers. In total, 1,559 people participated in the task.

After the crowdsourcing task, we estimated the probability that each sentence was judged *comprehensible* by the participants based on the aggregation method proposed by Whitehill et al. (2009). Sentences with probabilities of *comprehensible* less than 0.9 were discarded. Unlike majority voting, this method calculates the probability based on worker agreements. As a result, 9,073 sentences remained out of 11,189 representative sentences. These 9,073 sentences included 19,052 arguments (tokens) (4,882 types).

4.2. Phase 2: Animacy Investigation

When a word does not denote a concrete object, it is not necessary to ask about feature changes of the word (e.g., "discussion"). Even if the word denotes a concrete object, it is not necessary to investigate feature changes of emotions and senses of the word when the word denotes an object without emotions and senses (e.g., "stone"). To avoid such redundant asking in the feature change investigation task, we conducted a crowdsourcing task to investigate the animacy of words (arguments) in event sentences in advance.

In this task, crowdsourcing workers were asked whether presented words denote concrete objects or may have emotions or senses. That is, we divided this phase into two sub phases. The former (sub phase 1) is a task to investigate the concreteness of arguments. The other (sub phase 2) is a task to investigate whether arguments have emotions. In total, 1,011 participants completed these two sub tasks. The method for calculating the probability of answers was the same as in the event sentence creation phase. We discarded words whose probabilities of *concrete object* or *object with emotion* were less than 0.5. As a result, as far as we investigated, 1,575 types of words were judged as those representing concrete objects out of 3,457 types of words. 677 types of words were judged as those with emotions out of 3,457 types of words. Note that the number of words whose animacies were investigated (3,457 types) is different from the number of words in JFCKB (4,882 types) because this phase was applied from the middle of constructing JFCKB.

How does *anger* of "my child" change before and after the event described by the sentence below? (select *increased*, *decreased*, or *unchanged*)
My wife hits my child

(a)

How did your *anger* change before and after you read the event described by the sentence below? (select *increased*, *decreased*, or *unchanged*)
My wife hits my child

(b)

Figure 1: Questions used for crowdsourcing tasks. These tasks are to acquire concrete feature changes caused by events described by the sentences.

4.3. Phase 3: Feature Change Investigation

According to results of the previous two phases (i.e., event sentence creation and animacy investigation), we conducted a crowdsourcing task to gather feature changes of arguments in event sentences. We also attempted to gather those of sentence readers. That is, crowdsourcing workers were asked to answer one of feature changes of presented arguments in sentences (Figure 1 (a)) or those of the workers themselves (Figure 1 (b)). In total, 33,683 people participated in this task. Probability calculations were carried out in the same way as the previous phases. The resulting knowledge base includes 5,647 case frames (types). These case frames are for 975 verbs (types). Note that one verb has one or more case frames because of the verbal polysemy as shown in Table 2.

5. Validating Usefulness of JFCKB: Anaphora Resolution

5.1. Evaluation Settings

To validate the usefulness of JFCKB, we conducted an experiment of anaphora resolution using JFCKB. We adopted JWSC as an anaphora resolution problem. Each problem in JWSC is basically composed of two sentences including one anaphor, two antecedent candidates, and a correct antecedent. This anaphora resolution task can be regarded as an antecedent candidate ranking task because it is a task to select the correct antecedent from two candidates. We used Ranking SVMs[3] (Joachims, 2002) as the solver and used two-degree polynomial kernel as the kernel function. We set the cost parameter (trade-off between training error and margin) to 0.01.

In this experiment, three conditions were compared: (1) the case using only feature change information (hereafter, *FC*); (2) the case using only word embeddings of word2vec (hereafter, *W2V*), and (3) the case using both feature change information and word embeddings (hereafter, *BOTH*).

[3]https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

S1 = The bee landed on the flower. S2 = Because it wanted pollen.
antecedent 1 = bee, antecedent 2 = flower, anaphora = it, correct antecedent = bee
diff1 = it − bee, diff2 = it − flower

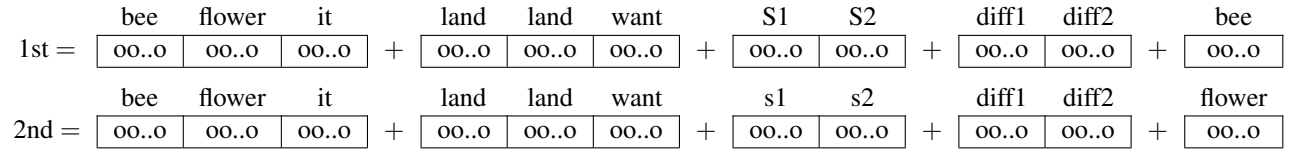


Figure 2: Example of anaphora resolution problem and vectors used in the anaphora resolution task. S1 and S2 denote the first sentence and the second sentence in the problem. Diff denotes the difference between an anaphor and an antecedent candidate. The vectors given to Ranking SVMs were composed of vectors expressing anaphors and antecedent candidates, vectors expressing the nearest (in parse trees) predicates of anaphoras and antecedent candidates, vectors expressing the first and second sentences, vectors expressing differences between anaphoras and antecedent candidates, and vectors expressing either of two antecedent candidates. As for the order in Ranking SVMs, the vector with the last component expressing the correct antecedent was regarded the first one.

The feature change vector of each argument is a 33 dimensional vector because we used eight emotion features (joy, trust, surprise, disgust, fear, sadness, anger, and anticipation) and three sensory featues (pain, sleepiness, and tiredness). Each feature change is composed of three probabilities (increased, decreased, and unchanged). The feature change vector of each predicate is a 99 dimensional vector because each predicate vector is composed of vectors for three cases (*ga* case (nominative), *wo* case (accusative), and *ni* case (dative)). Vectors in word2vec are 500 dimensional vectors learned from 100 million Web sentences analyzed by JUMAN++[4].

Vectors composed of the following feature vectors were given to Ranking SVM in all three conditions. (1) vectors expressing anaphors and antecedent candidates, (2) vectors expressing the nearest (in parse trees) predicates of anaphoras and antecedent candidates, (3) vectors expressing the first and second sentences, (4) vectors expressing differences between anaphoras and antecedent candidates, and (5) vectors expressing either of two antecedent candidates. We used KNP[5] for dependency parsing.

To solve JWSC problems, we created each vector of antecedent candidates based on five types of vectors described above and compared them. For each antecedent candidate, the five types of vectors were concatenated. Therefore, according to the dimension numbers of feature change vectors and word2vec vectors described above,vectors in *FC*, *W2V*, and *BOTH* are 693 dimensional, 5,500 dimensional, and 6,193 dimensional vectors respectively. As for the order in Ranking SVM, the vector with the last component expressing the correct antecedent was regarded the first one. For example, when the first sentence "the bee landed on the flower," the second sentence "Because it wanted pollen," two antecedent candidates "bee, flower," the anaphor "it," and the correct antecedent "bee" were given, the vector in which the last component represents "bee" is the first one (Figure 2).

We conducted ten-fold cross validation to evaluate system performance. Although JWSC is composed of 1,321 problems, we used 441 problems in this evaluation. These problems satisfied the following conditions: (1) each was composed of two sentences; (2) all the predicates of anaphors and those of antecedent candidates have information about their feature changes and word embeddings.

[4]http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?JUMAN++
[5]http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KNP

	FC	**W2V**	**BOTH**
Accuracy	52.15%	47.39%	47.62%

Table 4: Anaphora resolution result.

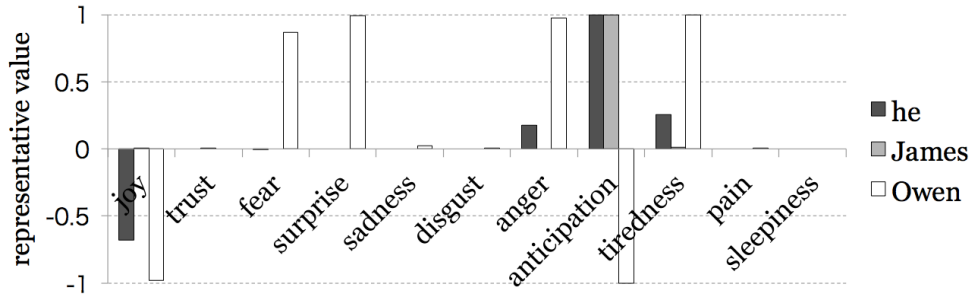
5.2. Results

The result of the cross validation is shown in Table 4. Table 4 shows that the accuracy of *FC* outperformed those of the others. An example that feature changes worked well was the case "James always gives orders to Owen. Because he is very bossy." Figure 3 (a) shows feature changes of this example. In this case, *FC* and *BOTH* estimated the correct answer while *W2V* failed. An example that feature changes did not work was the case "Bill punched Larry. And he was hurt." Figure 3 (b) shows feature changes of this example. In this case, *FC* and *BOTH* failed to estimate the correct answer while *W2V* succeeded. Note that Figure 3 shows representative feature change values. The representative value of each feature is the weighted average of the feature, where weights of *increased*, *decreased*, and *unchanged* are $+1$, -1, and 0, respectively. For example, when the feature change probabilities are 0.8, 0.15, and 0.05 for *increased*, *decreased*, and *unchanged*, respectively, the representative value of the feature is 0.65.

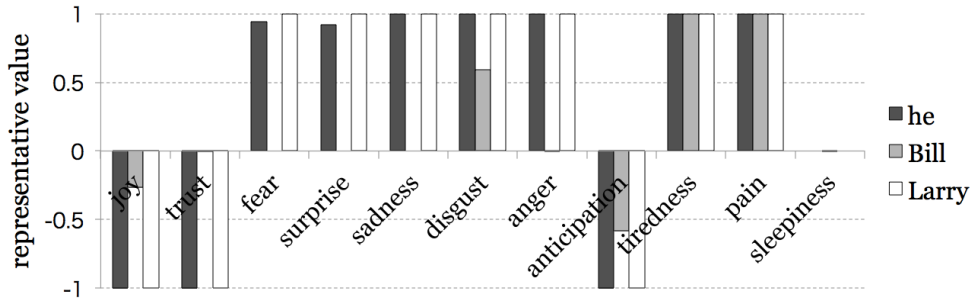
In the case that feature changes worked well, as can be seen in Figure 3 (a), the correct antecedent has some feature changes in common with the anaphors but the incorrect antecedent has a conflicting feature change (e.g., anticipation). In the case that feature changes did not work, as can be seen in Figure 3 (b), there are no conflicting feature changes. Considering these results, we speculate that such common feature changes and conflicting feature changes influence the estimation of the correct antecedent candidates in anaphora resolution.

6. Conclusion

In this study, we constructed a knowledge base where arguments in event sentences are associated with various feature



(a) James always gives orders to Owen. Because he is very bossy.



(b) Bill punched Larry. And he was hurt.

Figure 3: Examples of feature changes. (a) is an example where feature changes worked well. (b) is an example where feature changes did not work. The value of each feature is the representative feature change value. The representative value of each feature is the weighted average of the feature, where weights of *increased*, *decreased*, and *unchanged* are $+1$, -1, and 0, respectively.

changes caused by the events. Features used in the proposed knowledge base are determined by considering various studies such as traditional psychological studies, studies on cognitive development of infants, sentiment analysis studies. This knowledge base was constructed through crowdsourcing tasks. After the construction of our feature change knowledge base, we conducted an experiment of anaphora resolution using the knowledge base. In this experiment, we regarded anaphora resolution as an antecedent candidate ranking task and compare three conditions (the case using only the feature change information, the case using only the word2vec information, and the case using both of the feature change information and the word2vec information). As a result, it was shown that the condition where only the feature change information was used outperformed the other conditions. This result suggests the usefulness of our feature change knowledge base.

7. Acknowledgements

This work was supported by JST PRESTO Grant Number JPMJPR1402, Japan.

8. Bibliographical References

Baillargeon, R., Devos, J., and Graber, M. (1989). Location memory in 8-month-old infants in a non-search ab task: Further evidence. *Cognitive Development*, 4(4):345–367.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of ACL-COLING 1998*, pages 86–90.

Chambers, N. and Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 602 –610.

Chambers, N. and Jurafsky, D. (2010). A database of narrative schemas. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC2010)*.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6:169–200.

Christiane Fellbaum, editor. (1998). *WordNet: An Electronic Lexical Database*. The MIT Press.

Ghazi, D., Inkpen, D., and Szpakowicz, S., (2015). *Detecting Emotion Stimuli in Emotion-Bearing Sentences*, pages 152–165. Springer.

Gordon, J. and Durme, B. V. (2013). Reporting bias and knowledge acquisition. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*, pages 25–30.

Gordon, J., Durme, B. V., and Schubert, L. (2010). Learning from the web: Extracting general world knowledge from noisy text. In *Proceedings of the AAAI 2010 Workshop on Collaboratively-built Knowledge Sources and Artificial Intelligence (WikiAI 2010)*, pages 8–10.

Gui, L., Wu, D., Xu, R., Lu, Q., and Zhou, Y. (2016). Event-driven emotion cause extraction with corpus construction. In *Proceedings of EMNLP 2016*, pages 1639–1649.

Gui, L., Hu, J., He, Y., Xu, R., Lu, Q., and Du, J. (2017). A question answering approach to emotion cause extraction. In *arXiv:1708.05482v1*.

Hangyo, M., Kawahara, D., and Kurohashi, S. (2012).

Building a diverse document leads corpus annotated with semantic relations. In *Proceedings of the 26th Pacific Asia Conference on Language Information and Computing*, pages 535–544.

Hartshorne, J. K., Bonial, C., and Palmer, M. (2014). The verbcorner project: Findings from phase 1 of crowdsourcing a semantic decomposition of verbs. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 397–402.

Hasegawa, T., Kaji, N., Yoshinaga, N., and Toyoda, M. (2013). Predicting and eliciting addressee's emotion in online dialogue. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 964–972.

Ikehara, S. (1997). *Nihongo Goi Taikei*. Iwanami Syoten, Tokyo.

Jackendoff, R. (1983). *Semantics and Cognition*. MIT Press.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD), ACM, 2002*.

Kawahara, D. and Kurohashi, S. (2006). Case frame compilation from the web using high-performance computing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC2006)*, pages 1344–1347.

Levesque, H. J. (2011). The Winograd Schema Challenge. In *Proceedings of AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Li, S., Huang, L., Wang, R., and Zhou, G. (2015). Sentence-level emotion classification with label and context dependence. In *Proceedings of ACL 2015*, pages 1045–1053.

Massey, C. M. and Gelman, R. (1988). Preschoolers' ability to decide whether a photographed unfamiliar object can move itself. *Developmental Psychology*, pages 307–317.

Nakamura, T. and Kawahara, D. (2016). Constructing a dictionary describing feature changes of arguments in event sentences. In *Proceedings of the 4th Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 46–50.

NINJAL. (2004). *Bunrui Goihyo*. Dainippon Tosho, Tokyo.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Plutchik, R., (1980). *A General Psychoevolutionary Theory of Emotion, 1*, pages 3–33. Academic Press.

Rashkin, H., Singh, S., and Choi, Y. (2016). Connotation frames: A data-driven investigation. In *Proceedings of ACL 2016*, pages 264–269.

Rosch, E., Mervisa, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, 8:382–439.

Shibata, T., Kohama, S., and Kurohashi, S. (2014). A large scale database of strongly-related events in Japanese. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC2014)*, pages 3283–3288.

Shibata, T., Kohama, S., and Kurohashi, S. (2015). Nihongo Winograd Schema Challenge no kouchiku to bunseki (in Japanese). In *Proceedings of NLP2015*, pages 493–496.

Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., and Zhu, W. L. (2012). Open mind common sense: Knowledge acquisition from the general public. In *Proceedings of On the Move to Meaningful Internet Systems, 2002*, pages 1223–1237.

Speer, R. and Havasi, C. (2012). Representing general relational knowledge in conceptnet 5. In *Proceedings of LREC 2012*, pages 86–90.

Spelke, E. S., Phillips, A., and Woodward, A. L., (1995). *Infants' knowledge of object motion and human action*. Oxford University Press.

Strapparava, C. and Valitutti, A. (2004). Wordnet-affect: an affective extension of wordnet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1083–1086.

Taylor, J. R. (1995). *Linguistic Categorization: Prototypes in Linguistic Theory*. Clarendon Press.

Tokuhisa, R., Inui, K., and Matsumoto, Y. (2008). Emotion classification using massive examples extracted from the web. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 881–888.

Tokuhisa, R., Inui, K., and Matsumoto, Y. (2009). Emotion classification using massive examples extracted from the web. *Journal of Information Processing (in Japanese)*, 50(4):1365–1374.

Vanderwende, L. (2005). Volunteers created the web. In *Proceedings of the 2005 AAAI Spring Symposium, Knowledge Collection from Volunteer Contributors*.

Vu, H. T., Neubig, G., Sakti, S., Toda, T., and Nakamura, S. (2014). Acquiring a dictionary of emotion-provoking events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 128–132.

Whitehill, J., Ruvolo, P. L., Ting-fan Wu, J. B., and Movellan, J. R. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, et al., editors, *Advances in Neural Information Processing Systems 22*, pages 2035–2043. Curran Associates, Inc.

Zhou, D., Zhou, Y., Zhang, X., Zhao, Q., and Geng, X. (2016). Emotion distribution learning from texts. In *Proceedings of EMNLP 2016*, pages 638–647.

Quantifying Qualitative Data for Understanding Controversial Issues

Michael Wojatzki[†], Saif M. Mohammad[◇], Torsten Zesch[†], and Svetlana Kiritchenko[◇]
[†]Language Technology Lab, University of Duisburg-Essen
{michael.wojatzki,torsten.zesch}@uni-due.de
[◇]National Research Council Canada
{saif.mohammad,svetlana.kiritchenko}@nrc-cnrc.gc.ca

Abstract

Understanding public opinion on complex controversial issues such as '*Legalization of Marijuana*' and '*Gun Rights*' is of considerable importance for a number of objectives such as identifying the most divisive facets of the issue, developing a consensus, and making informed policy decisions. However, an individual's position on a controversial issue is often not just a binary support-or-oppose stance on the issue, but rather a conglomerate of nuanced opinions and beliefs on various aspects of the issue. These opinions and beliefs are often expressed qualitatively in free text in issue-focused surveys or on social media. However, quantifying vast amounts of qualitative information remains a significant challenge. The goal of this work is to provide a new approach for quantifying qualitative data for the understanding of controversial issues. First, we show how we can engage people directly through crowdsourcing to create a comprehensive dataset of assertions (claims, opinions, arguments, etc.) relevant to an issue. Next, the assertions are judged for agreement and strength of support or opposition, again by crowdsourcing. The collected Dataset of Nuanced Assertions on Controversial Issues (NAoCI dataset) consists of over 2,000 assertions on sixteen different controversial issues. It has over 100,000 judgments of whether people agree or disagree with the assertions, and of about 70,000 judgments indicating how strongly people support or oppose the assertions. This dataset allows for several useful analyses that help summarize public opinion. Across the sixteen issues, we find that when people judge a large set of assertions they often do not disagree with the individual assertions that the opposite side makes, but that they differently judge the relative importance of these assertions. We show how assertions that cause dissent or consensus can be identified by ranking the whole set of assertions based on the collected judgments. We also show how free-text assertions in social media can be analyzed in conjunction with the crowdsourced information to quantify and summarize public opinion on controversial issues.
Keywords: controversial issues, judging assertions, crowdsourcing, stance detection, argument mining, sentiment analysis

1. Introduction

Controversy is a state of sustained public debate on a topic or issue that evokes conflicting opinions, beliefs, claims, arguments, and points of view. In this paper, we will refer to utterances of opinion, belief, claim, argument, or point of view relevant to an issue as *assertions*. People make assertions on a controversy (or controversial issue) both in the physical world and on social media. Others might agree or disagree with these assertions. An individual's *position* on an issue is not simply a binary support-or-oppose stance on the issue, but rather a cumulative sum of many nuanced beliefs and opinions. Thus assertions are a useful means of capturing one's position on a controversial issue. Examples of common controversial issues include *the legalization of marijuana, government policy on refugees,* and *gun rights*. Examples of assertions include '*Marijuana alleviates the suffering of chronically ill patients*' and '*Expanding legal use of marijuana makes illegal use easier*'.

Controversial issues are usually complex, not only because of the many diverse and inter-related assertions they evoke, but also because they often subsume many inter-related sub-issues (e.g., '*Should marijuana be legalized for medical purposes?*'). An issue has many *stake holders*—people that are directly or indirectly affected by it—e.g., patients undergoing chemotherapy and parents of teenage children are just two of the many stakeholders affected by a possible legalization of marijuana. Given these complexities, it is difficult to attain consensus, and people often tend to talk past each other without really listening to the merits of opposing arguments (amplifying existing echo chambers). Even decision making bodies, such as local and national governments, can make more informed choices if they have a comprehensive understanding of the controversial issue. Simply obtaining the percentage of people that support or oppose an issue is not sufficient. It is often more useful to obtain information about the different aspects of an issue, what aspects of an issue are considered more important, what people's beliefs and opinions on various aspects are, who the main stakeholders are, what opposing groups agree and disagree on, etc.

A common approach to understanding complex controversial issues is to hire experts and conduct surveys. However, such an approach has inherent limitations: the survey creators inadvertently bring in biases, often these surveys fail to cover all relevant aspects, and the process is time intensive and expensive. Further, such surveys often only make use of questions whose responses can be easily aggregated, e.g., multiple-choice questions and questions with a small set of possible responses. Quantifying qualitative responses often requires manual interpretation and does not scale up to large surveys. In contrast, people naturally express their nuanced positions on an issue through free-text utterances in the real world and in on-line social networks. Capturing and quantifying information in free-text assertions remains a significant challenge in understanding controversial issues.

This work has two broad goals. First, we propose a method to obtain and quantify qualitative information relevant to a controversial issue by **engaging people directly via crowdsourcing**. Specifically, we create a comprehensive dataset of nuanced assertions relevant to sixteen controversial issues in the United States. Next, we rank the as-

sertions by both the degrees of agreement on them and by how strongly people support or oppose each of the assertions. We create this dataset by conducting crowdsourced surveys to: (1) collect about 150 unique assertions per issue, (2) determine agreement on these assertions by hundreds of respondents, and (3) rank the assertions based on how strongly people support or oppose them. For each of the sixteen issues we collect around 150 assertions (2,243 in total). For each of the assertions, we obtain 15 judgments on how strongly people support or oppose the assertions (67,290 in total) and about 45 judgments for agreement (101,133 in total). We will refer to this dataset as the dataset of *Nuanced Assertions on Controversial Issues (NAoCI)*.

We propose several metrics that can be calculated from the data and used for grouping, ranking, and clustering assertions and participants. We show how these metrics can be used to identify agreement and support on assertions, to rank issues based on controversial assertions, and to determine the similarity of assertions, users and groups. An analysis of the distribution of judgments shows that for all of the controversial issues there are more assertions with which the majority of the people agree than assertions with which the majority of the people disagree or assertions with which some agree and some do not (controversial assertions). However, the controversial assertions are often the ones that are supported or opposed to the greatest degree. The new approach to understanding argumentation proposed here goes well beyond simple positive–negative–neutral classification or overall stance detection from text. The NAoCI dataset will help foster new research that tackles difficult questions such as how people make arguments to support their stance on an issue, what is the distribution of assertions that people agree and disagree with across different groups, and how the position on a controversial assertion impacts overall stance.

Our second goal is to improve the **understanding of controversial issues using assertions made by people in social media**. As a first step towards achieving this goal, we propose several new natural language processing tasks in this paper. These tasks include identifying assertions implicit in free-text posts on social media, determining a speaker's position on various assertions, identifying the degree of agreement, support, and opposition for assertions by a large population of tweeters that post messages about a controversial issue, etc. These new natural language processing tasks are a way to summarize information about the controversial issues without necessarily having to do the crowdsourcing described above. However, the crowdsourced data will serve as a source of reference (gold) labels for the evaluation of these NLP algorithms.

All the annotation tasks described in this paper were approved by the National Research Council Canada's Institutional Review Board, which reviewed the proposed methods to ensure that they were ethical. All our data and crowdsourcing questionnaires are made available on the project webpage.[1]

[1]https://sites.google.com/view/you-on-issues

2. Related Work

In recent years, a number of web-based applications have been developed that help users share their opinions and explore public opinion on controversial issues. Specifically, Voting Advice Applications such as *votecompass.com*, *politicalcompass.org*, and *isidewith.com* ask visitors whether they agree or disagree with a set of pre-chosen assertions (Garzia and Marschall, 2012). The applications usually provide some form of visualization that depict which political party or candidate is closest to the user's own position. Voting Advice Applications have mainly been studied in political science, where researchers have examined questions such as: whether these applications have an effect on voting behavior (Ladner and Pianzola, 2010), what characteristics their users have (Wall et al., 2009), or how their design affects their outcome (Louwerse and Rosema, 2014). In contrast, here we focus on creating a language resource containing crowdsourced judgments on a large number of user generated assertions.

Our second goal of understanding and summarizing public opinion from posts on social media is related to work on detecting sentiment (Pang and Lee, 2008; Liu, 2012; Mohammad, 2016; Mohammad et al., 2018) and stance (Mohammad et al., 2016; Xu et al., 2016; Taulé et al., 2017), argumentation mining (Kwon et al., 2007; Walker et al., 2012; Rosenthal and McKeown, 2012; Stab and Gurevych, 2014; Peldszus and Stede, 2016; Habernal and Gurevych, 2016), and framing (Entman, 1993; Card et al., 2015; Tsur et al., 2015; Fulgoni et al., 2016; Johnson and Goldwasser, 2016). These approaches focus on identifying sentiment, stance, claims, premises, reasons, arguments, sentiment, etc. from individual utterances. In contrast, here we suggest quantifying information from a large number of social media utterances in order to gain the overall understanding of a complex issue. There exists work on opinion summarization (Hu and Liu, 2004; Zhuang et al., 2006; Titov and McDonald, 2008; Titov and McDonald, 2008; Ganesan et al., 2010; Gerani et al., 2014) and on the clustering of argumentative elements (Trabelsi and Zaıane, 2014; Misra et al., 2015; Boltužić and Šnajder, 2015; Barker and Gaizauskas, 2016); however, these studies mainly explore the grouping of utterances into positive and negative clusters, or extracting opinion utterances and claims to create a summary. Here, we first establish a comprehensive representation of assertions for a controversial issue, and then propose discovering elements of this representation from free-text utterances in social media.

3. Understanding Controversial Issues

Quantifiable and useful insights on a controversial issue can be obtained by having a large number of people vote on a large number of relevant assertions. As the manual creation of assertions is time-consuming, subject to personal bias, and potentially incomplete, we here rely on crowdsourcing to generate the assertions. In the subsections below, we describe how we: (1) engage people directly through crowdsourcing to obtain assertions and judgments on these assertions (Figure 1 gives the overview for an example issue), and (2) analyze and summarize the crowdsourced information through a number of ways

Issue	# of Assertions	# of Agreement Judgments	# of Support–Oppose Judgments
Black Lives Matter	135	6,154	4,050
Climate Change	142	6,473	4,260
Creationism in school	129	5,747	3,870
Foreign Aid	150	6,866	4,500
Gender Equality	130	5,969	3,900
Gun Rights	145	6,423	4,350
Marijuana	138	6,200	4,140
Same-sex Marriage	148	6,899	4,440
Mandatory Vaccination	134	5,962	4,020
Media Bias	133	5,877	3,990
Obama Care	154	6,940	4,620
US Electoral System	175	7,695	5,250
US in the Middle East	138	6,280	4,140
US Immigration	130	5,950	3,900
Vegetarianism & Veganism	128	5,806	3,840
War on Terrorism	134	5,892	4,020
Total	2,243	101,133	67,290

Table 1: Issues, number of generated assertions, and number of collected judgments.

including: ranking assertions based on how they were judged, clustering people and assertions by voting patterns, and ranking issues by degree of polarization. The sixteen controversial issues we explore are shown in Table 1. They were compiled from voting advice websites.

3.1. Collecting Public Opinion

Generating Assertions: Given an issue (name and a brief description), we asked each participant to come up with five assertions relevant to the issue. To guide the process of creating assertions, the participants were given the following directions. Participants had to formulate assertions in a way that a third person can agree or disagree with it. The assertions had to be self-contained and understandable. Hedged statements that included words such as '*maybe*', '*perhaps*', or '*possibly*' were prohibited.

As a quality control measure, participants were also required to answer test questions where they had to indicate whether the given assertions were applicable to the issue and whether they were in accordance with the rules. We discarded all responses from participants who incorrectly answered more than 10% of these questions.

We obtained 2,243 assertions from 69 participants for the sixteen issues. Duplicates and instances not in accordance with the guidelines were removed. Table 1 lists the number of remaining valid assertions for each issue.

Quantifying Agreement and Strength of Support and Opposition: Once the list of assertions was compiled, we obtained judgments on (a) whether people agree with the assertion, and (b) how strongly they support or oppose the assertion. A participant may not be inclined to judge all assertions. However, if a large enough number of judgments are obtained from many participants, then meaningful inferences can be drawn. Thus, individual participants were free to judge as many assertions as they wished. For both kinds of judgments, the exact questionnaires along with the directions and examples are shown in the Appendix.

To obtain agreement judgments on the assertions, we simply asked the subjects to indicate to us whether they agree or disagree with the collected assertions. The average number of agreement judgments per assertion was 45. Table 1 shows how many judgments we collected for each issue. From the agreement judgments, we created the agreement matrix AM, which contains one column per assertion, and one row per participant. Each cell $ad_{p,a}$ in this matrix has the judgment provided by participant p for assertion a. Consequently, $\vec{ad}_a$ is the vector of all judgments provided for assertion a, and $\vec{ad}_p$ is the vector of all judgments provided by participant p.

To obtain consistent and comparable fine-grained scores indicating the degree of support or opposition from multiple respondents, we used a technique known as Best–Worst Scaling (BWS) (Louviere et al., 2015b). BWS is an annotation scheme that addresses the limitations of traditional rating scale methods, such as inter- and intra-annotator inconsistency, by employing comparative annotations (Louviere, 1991; Louviere et al., 2015a; Kiritchenko and Mohammad, 2016; Kiritchenko and Mohammad, 2017). Annotators are given n items (an n-tuple, where $n > 1$ and commonly $n = 4$). They are asked which item is the *best* (highest in terms of the property of interest) and which is the *worst* (lowest in terms of the property of interest). When working on 4-tuples, best–worst annotations are particularly efficient because each best and worst annotation will reveal the order of five of the six item pairs. For example, for a 4-tuple with items A, B, C, and D, if A is the best, and D is the worst, then A $>$ B, A $>$ C, A $>$ D, B $>$ D, and C $>$ D. Real-valued scores of association between the items and the property of interest can be calculated from the BWS annotations (Orme, 2009; Flynn and Marley, 2014).

We generated 4,486 4-tuples of assertions from our list of 2,243 assertions using the code provided by Kiritchenko and Mohammad (2016). Participants were presented with four assertions at a time and asked two questions:

1. which of the assertions they support the most (or oppose the least),
2. which of the assertions they oppose the most (or support the least).

We obtained judgments from fifteen people for every 4-tuple. From the comparative judgments, we created the support–oppose matrix SM, that consists of one row per participant, and one column per assertion. In each cell $bd_{p,a}$ we store a tuple that indicates how often participant p has selected assertion a as the one they support the most (or oppose the least), and how often participant p has selected assertion a as the one they oppose the most (or support the least).

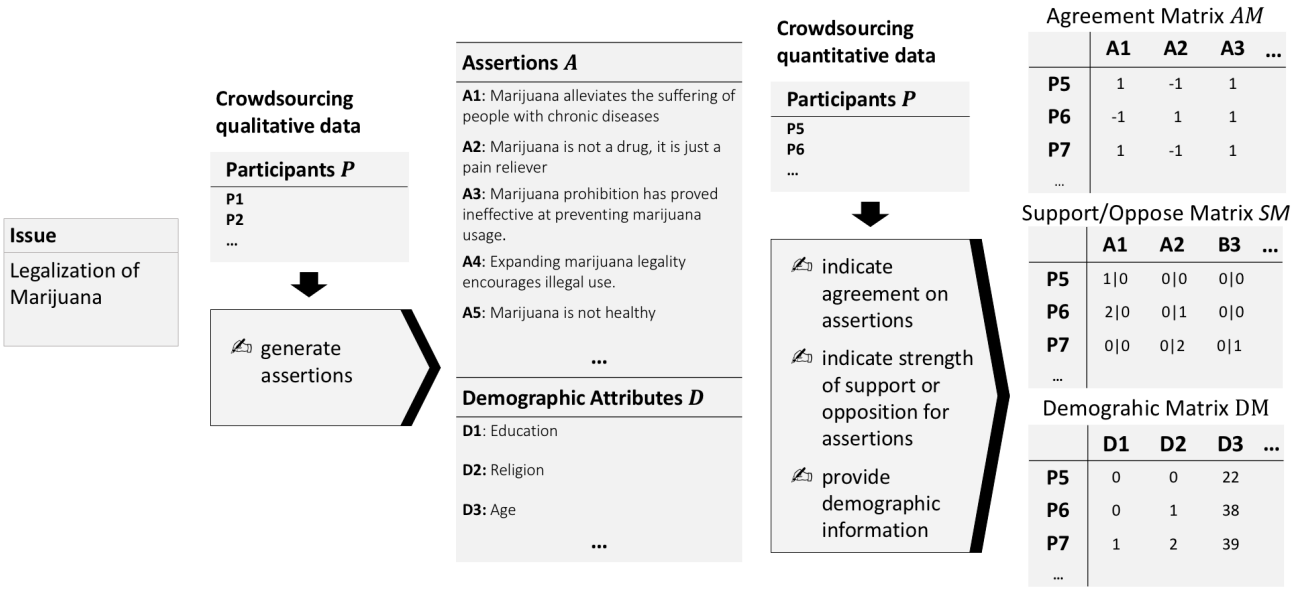


Figure 1: Quantifying qualitative assertions for the issue '*Legalization of Marijuana*'. The agreement matrix has values 1 for agreement and -1 for disagreement. The support/oppose matrix contains integer values indicating how many times the participants have selected an assertion as the one they most support (resp. least oppose) and most oppose (resp. least support). The demographic matrix contains the responses to the demographic questions such as education, religion, and age.

Demographic Data: A key determiner of one's beliefs and opinions is their personal experience, which in turn is often shaped by their demographic attributes. In order to determine the extent to which demographic attributes correlate with one's judgments on assertions, we asked the participants to provide us with their demographic information: age, gender, political affiliation, education, family status, profession, race, religion, whether the participants have ties to overseas and whether they are US citizens. Participants were free to not provide this demographic data if they wished.

Participants: Of the 230 subjects that participated in the quantitative phase, 85 (37%) submitted demographic information. This sample had a mean age of 34.9 years; 65% of the respondents were female. Just over 50% of the respondents had a bachelors or higher degree. With regard to political affiliation, a broad mix could be observed. However, the group of people who identified themselves as democrats was the largest. About 50% of respondents indicated that they were employed. Most of the participants had ties to overseas (68%) and were US citizens (89%). With respect to religion, the majority of the participants were Christians (Roman Catholic 28%, Protestant 25%, Russian Orthodox 1%). About 29% were atheists. 69% of our participants identified themselves as white, 10% as Hispanic, 8% as black or African American, 8% as Asian, 2% as American Indian or Alaskan native, and 3% as others. Further details of the demographic information can be found in the Appendix.

3.2. Summarizing Public Opinion

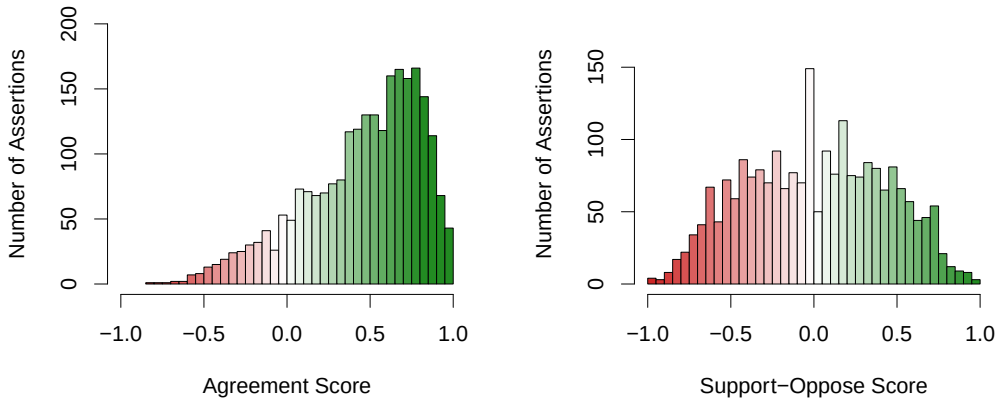
The data gathered from crowdsourcing efforts can be used to summarize, visualize, and gain insights into public opinion on controversial issues. In the subsections below, we describe how we calculate various metrics from the data that can be used to summarize various aspects of public opinion on an issue.

3.2.1. Ranking Agreement and Support

Assertions expressed by participants provide key insights into why an issue is controversial, what aspects of the issue people are particularly passionate about, etc. Thus organizing the assertions by amount of agreement (to quickly view the assertions with most and least agreement) and strength of support or opposition is particularly useful. We calculate the *agreement score* (ags) of an assertion a by simply subtracting the percentage of times the assertion was disagreed with from the percentage of times the assertion was agreed with:

$$ags(a) = \%\ \text{agree}(a) - \%\ \text{disagree}(a) \qquad (1)$$

The agreement score can be used to rank assertions from **least agreement** (-1) to **most agreement** (1). A score of 0 indicates that an equal number of participants agree and disagree with the assertion and that the assertion is therefore highly **controversial**. We can identify the most controversial assertions by sorting assertions by the absolute value of the agreement scores and selecting those assertions which have the lowest absolute scores. These agreement scores can be used to better understand the debate on an issue. For instance, for the issue *legalization of same-sex marriage*, the assertion '*Love is a right for everyone.*' has the highest agreement score, '*Saying that gay people should get married is like saying that a brother can marry his sister both are at higher risk of disease.*' has the lowest agreement score, and '*Allowing same-sex marriage will create a slippery slope where people will begin to fight for other alter-*



(a) Distribution of Agreement Scores

(b) Distribution of Support–Oppose Scores

Figure 2: Distribution of agreement and support–oppose scores across all issues. We group the agreement and support–oppose scores into bins of size 0.05. For both the agreement and support–oppose scores, the colors encode how positive (green) or negative (red) the scores are.

native marriages such as polygamy.' is the most controversial. In the Appendix, we show the top three assertions according to these rankings for all issues.

We transform the comparative BWS judgments for support and opposition into scores ranging from -1 (maximum opposition) to 1 (maximum support) through a simple counting method proposed by Orme (2009). For an assertion a we calculate a *support–oppose score* (sos) by subtracting the percentage of times an assertion was chosen as the most opposed from the percentage of times the assertion was chosen as the most supported:

$$sos(a) = \%\text{ most support}(a) - \%\text{ most opposed}(a) \quad (2)$$

These scores can be used to rank assertions from most strongly supported (1) to most strongly opposed (-1). Selecting an assertion as 'most opposed' in the comparative annotations, may mean that one either most opposes or least supports the assertion. We can infer which of the two interpretations applies from the agreement judgments; i.e. 'most oppose' can be considered as 'least support' if a participant agrees to a statement and it remains 'most oppose' if the person disagrees with the assertion. Analogously, 'most support' can be interpreted as 'least oppose' if one disagrees to the assertion and it can be interpreted as 'most support' if one agrees to the assertion. Thus, we additionally calculate a *support score* (ranging from 'least supported' (0) to 'most supported' (1)) and an *oppose score* (ranging from 'least opposed'(0) to 'most opposed' (1)). To calculate these scores, we reuse the formula shown in equation 2. However, the percentages are now calculated only on the set of persons that have agreed to (*support score*) or disagreed with (*oppose score*) the assertion. These scores can be used to differentiate between assertions where a *support–oppose score* of about zero indicates that an assertion is both strongly supported and strongly opposed, and assertions which have a *support–oppose score* of about zero but that are rarely strongly supported or opposed. An example for the former is the assertion '*Freedom of the press prevents the government or other third parties from controlling the media.*' that has a *support–oppose score* of 0, but a fairly high *support score* of 0.71, and a high *oppose score* of 1. An example for the latter is the assertion '*Women's rights have well found legal basis.*' that has a *support–oppose score* of -0.07 and both a low *oppose score* (0) and *support score* (0.39).

Figure 2 shows histograms of the agreement and support–oppose scores. For the support–oppose scores, we observe that the scores have a normal distribution. This distribution can also be found when looking at the distributions per issue. For the agreement scores, we observe that the mass of the distribution is concentrated in the positive range of possible values, indicating that the participants tend to agree with the assertions more often than they disagree. (We observe a similar distribution in each of the individual issues as well.) This is consistent with the hypothesis that people often do not disagree with the individual assertions that the other side makes, however, they might disagree on the relative importance of that assertion among the various other assertions in terms of reaching an overall stance on the broader issue.

Examples of assertions with a particular high *agreement score* for the *Legalization of Marijuana* issue are '*Drug abuse can kill.*', '*Cannabis has legitimate medical effects.*', and '*Exceptions should be considered allow people with medical issues that will benefit from legalization of Marijuana.*'.

3.2.2. Ranking Controversial Issues

Governments and policy makers often have to work with not one but several controversial issues. Thus it is useful to know which issues are particularly polarizing so as to prioritize those issues or to allocate appropriate resources to

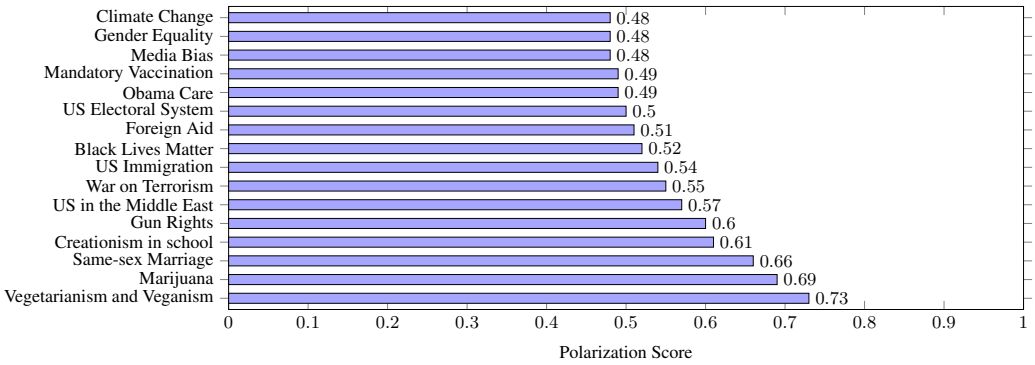


Figure 3: Issues ranked according to their polarization score.

them. One indicator of the degree of polarization is the extent to which the assertions associated with the issue evoke opposing responses. If all the assertions for an issue have an agreement score of zero (the number of respondents that agree is equal to the number of respondents that disagree), then the issue is maximally polarizing. Hence, we calculate the *polarization score* (ps) of an issue (a set of assertions) I by first calculating the average of the absolute value of the agreement score for each of the assertions, and then subtracting this value from one:

$$ps(I) = 1 - \frac{1}{|I|} \sum_{a \in I} |ags(a)| \tag{3}$$

A polarization score of 0 indicates that participants consistently agree or disagree with all assertions representing an issue. A polarization score of 1 indicates that – on average – an equal number of participants agree and disagree with the assertions on an issue. Consequently, a polarization score of 0.5 describes an issue in which more and less polarizing assertions keep a balance.

The polarization scores for the sixteen issues are shown in Figure 3. Interestingly, many of the polarization scores for the issues are around 0.5. For the issues *Climate Change*, *Gender Equality*, *Media Bias*, *Mandatory Vaccination*, and *Obama Care* the scores are even below 0.5, which means that on average there is more consensus than dissent in judging the assertions on these issues. However, as shown by the issues *Same-sex Marriage* (0.66), *Marijuana* (0.69) and *Vegetarianism & Veganism* (0.73), our data contains also more polarizing issues. In future work, we plan to examine whether linguistic properties of the assertions (e.g., whether they use superlatives) can be utilized to explain these differences. Also note that the score does not include external factors such as the social context in which a controversy takes place and should be used only as one (of the many possible ways) in which polarization can be measured.

3.2.3. Determining Similarity of Users and Assertions

The crowdsourced data can be used to determine which users show a similar response behavior and which assertions have been similarly voted on. Voting similarity between participants can be used to generate guesses about their judgments on assertions for which they have not voted. We determine the voting similarity between pairs of participants by computing the cosine of the vectors that represent the rows in the agreement matrix AM (see Figure 1):

$$cos(p1, p2) = \frac{\vec{ad}_{p1} \cdot \vec{ad}_{p2}}{|\vec{ad}_{p1}| \cdot |\vec{ad}_{p2}|} \tag{4}$$

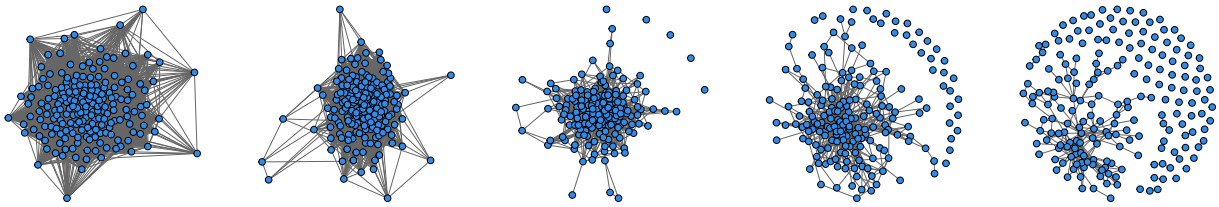
Being able to judge similarity between assertions helps identify inter-related assertions. We determine the degree by which two assertions are judged similarly by computing the cosine of the column vectors of the agreement matrix AM (see Figure 1):

$$cos(a1, a2) = \frac{\vec{ad}_{a1} \cdot \vec{ad}_{a2}}{|\vec{ad}_{a1}| \cdot |\vec{ad}_{a2}|} \tag{5}$$

The computed similarities between pairs of assertions are made public on the project's website. We manually inspected pairs of assertions with a particularly high or low judgment similarity. We found several reasons for high similarity between assertions (why people tend to agree with both assertions or they tend to disagree with both assertions): the two assertions are close paraphrases (e.g., '*Owning a gun can deter criminals.*' and '*Gun ownership deters crime.*'), one assertion entails the other (e.g., '*Oceans rise due to climate change.*' and '*Climate change has a big effect on the Earth.*'), underlying socio-cultural and political factors cause people to vote similarly on two (sometimes seemingly unrelated) assertions (e.g., '*US should hire skilled immigrants.*' and '*Inclusion should be facilitated for immigrants.*').

Reasons for low similarity between assertions (why people tend to agree with one assertion and disagree with the other) include: the two assertions are contradictory or contrasting (e.g., '*It is safe to use vaccines.*' and '*Vaccines cause autism.*'), and underlying socio-cultural and political factors cause people to vote dissimilarly on two (sometimes seemingly unrelated) assertions (e.g., '*Congress should immediately fund Trump's wall.*' and '*All immigrants should have the right to vote in the American elections.*').

To further explore the relation between judgment similarity and semantic textual similarity (Agirre et al., 2012), we compute the textual overlap between assertions using the Jaccard index (Lyon et al., 2001) and examine the agreement scores of textually similar assertions. We observe that assertions with high text similarity often have very similar



(a) cosine > 0.0 (b) cosine > 0.1 (c) cosine > 0.2 (d) cosine > 0.3 (e) cosine > 0.4

Figure 4: Similarity of participants visualized in an undirected graph for the issue *Black Lives Matter*. In the sub figures, we draw edges between two persons if their voting similarity is above a certain threshold.

agreement scores. An example for this case are the following assertions:

> '*Women should have the same rights than men.*' $(ags = 0.84)$
> '*Women should have the same rights as men.*' $(ags = 0.88)$
> '*Women should have the same right as men.*' $(ags = 0.86)$

Note that the first and third example are ungrammatical, which does not seem to significantly affect the rating.

However, there are also pairs with a high textual overlap and significantly different agreement scores. In these pairs, slight differences in wording have a marked influence on how the assertions are judged. Examples include:

> '*Foreign aid budget should be more effective.*' $(ags = 0.73)$
> '*The foreign aid budget should be made more effective.*' $(ags = 0.65)$

as well as

> '*Climate change is costing lives.*' $(ags = 0.76)$
> '*Climate change is already costing lives.*' $(ags = 0.52)$.

3.2.4. Clustering Participants With Similar Positions

It is often useful to identify which groups of people have similar positions on a controversial issue. This allows for focused engagement with individual groups. The agreement judgments in our data can be used to cluster participants according to their judgments on assertions.

We use the voting similarity between participants (c.f. equation 4) to find groups of people with similar overall beliefs on an issue. The same methods described in section 3.2.1. to calculate the agreement and support–oppose scores for an assertion can be used to calculate agreement and support–oppose scores for individual groups (instead of for the whole population). Then separate ranked lists of assertions can be generated for each group. This ranking can then be used to summarize the judgments of the groups. We can also determine which assertions a group agrees the most with, which assertions receive similar judgments across two groups, and which assertions the two groups disagree on.

When we cluster participants by judgment similarity, several scenarios are possible: a binary split into persons that support and oppose the overall issue (e.g., one cluster includes people in favour of legalizing marijuana and one cluster includes people that are against), several clusters that correspond to persons with more specific positions (e.g., being against legalizing marijuana for medical purposes or favoring the idea that marijuana is a gateway drug), or single cluster which expresses a mainstream of positions.

To examine the distribution of similar participants, we create an undirected graph in which the nodes represent participants and edges are drawn if the person–person similarity exceeds a certain threshold. Next, we compare graphs with different thresholds. We find that it is not uncommon for an issue that at a low threshold almost all participants are connected; i.e. that all subjects have a certain similarity to each other. If we increase the threshold, we do not observe the formation of several clusters, but of a single cluster and an increasing amount of single disconnected outliers. This indicates that the majority of persons in our data belongs to a mainstream. We visualize this experiment for the issue *Black Lives Matter* in Figure 4. We manually inspect the judgments of disconnected persons and observe that these indeed tend to have rather radical positions (e.g., disagreeing with the assertion '*Everyone is equal.*'). Note that clustering using cosine similarity is just one of the several ways to identify groups of similar positions. In future work, we plan to examine other ways of determining similarity between vectors (e.g., by considering their overlap).

4. Understanding Controversial Issues from Assertions in Social Media

The second goal of this work is to summarize information about controversial issues without necessarily being dependent on the described crowdsourcing. We propose to make use of the abundance of opinions and beliefs that are expressed on social media, especially on controversial issues. We now briefly outline a number of NLP tasks that can be developed for the understanding of controversial issues by identifying and analyzing assertions made in social media. Notably, the dataset described in the previous section can be

used as reference (gold) to evaluate these automatic methods.

- Identifying explicit and implicit assertions relevant to an issue in social media posts (e.g. tweets).
- Identifying social media posts that express the same assertions in different ways; Identifying posts that express contradictory or opposing assertions.
- Compiling a large list of assertions relevant to an issue from tweets; ranking them by an estimate of the degree of agreement; ranking them by an estimate of the support and oppose scores; automatically estimating polarization scores for issues.
- Determining semantic similarity between pairs of assertions; Clustering tweeters by the similarity of the assertions they make (agree with); Clustering assertions by their similarity.

As a first step, we have begun collecting tweets pertinent to each of the sixteen issues (that are part of this project) through a small manually identified list of query terms.[2] We have collected close to nine million tweets. The IDs of these tweets are available on the project webpage.

5. Conclusion

We proposed a method to quantify qualitative information relevant to a controversial issue by engaging people directly via crowdsourcing. This new approach to understanding argumentation goes well beyond simple positive–negative–neutral classification or overall stance detection from text. We created a dataset containing a comprehensive and nuanced list of assertions relevant to an issue, and then ranked them by both the agreement and degree of support for each assertion. We applied the proposed method for sixteen different controversial issues and collected a total of over 2,000 assertions. We obtained over 100,000 judgments of whether people agree or disagree with the assertions and about 70,000 judgments indicating how strongly people support or oppose the assertions. We also proposed several metrics such as an agreement score for an assertion and a polarization score for an issue that can be calculated from the data and used for grouping, ranking, and clustering issues, assertions and participants. Finally, we outlined a number of NLP tasks for the understanding of controversial issues through assertions made by people in social media. Our dataset, Nuanced Assertions on Controversial Issues (NAoCI), can be used as a source of reference labels in the evaluations of these tasks. We expect the NAoCI dataset and automatic algorithms produced as part of this project to be helpful—both to the lay person and expert—in making informed decisions about complex controversial issues.

6. Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media". We would also like to thank Tara Small for the helpful discussions.

[2]The full list of query terms is available on the project webpage: https://sites.google.com/view/you-on-issues.

7. Bibliographical References

Agirre, E., Diab, M., Cer, D., and Gonzalez-Agirre, A. (2012). Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393.

Barker, E. and Gaizauskas, R. (2016). Summarizing multi-party argumentative conversations in reader comment on news. In *Proceedings of the 3rd Workshop on Argument Mining*, pages 12–20, Berlin, Germany.

Boltužić, F. and Šnajder, J. (2015). Identifying Prominent Arguments in Online Debates Using Semantic Textual Similarity. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 110–115, Denver, USA. Association for Computational Linguistics.

Card, D., Boydstun, A. E., Gross, J. H., Resnik, P., and Smith, N. A. (2015). The media frames corpus: Annotations of frames across issues. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 438–444, Beijing, China.

Entman, R. M. (1993). Framing: Toward clarification of a fractured paradigm. *Journal of communication*, 43(4):51–58.

Flynn, T. N. and Marley, A. A. J. (2014). Best-worst scaling: theory and methods. *Handbook of Choice Modelling*, pages 178–201.

Fulgoni, D., Carpenter, J., Ungar, L. H., and Preotiuc-Pietro, D. (2016). An Empirical Exploration of Moral Foundations Theory in Partisan News Sources. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3730–3736, Portorož, Slovenia.

Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 340–348, Beijing, China.

Garzia, D. and Marschall, S. (2012). Voting advice applications under review: the state of research. *International Journal of Electronic Governance*, 5(3-4):203–222.

Gerani, S., Mehdad, Y., Carenini, G., Ng, R. T., and Nejat, B. (2014). Abstractive Summarization of Product Reviews Using Discourse Structure. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1602–1613, Doha, Qatar.

Habernal, I. and Gurevych, I. (2016). Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1589–1599.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, Seattle, USA.

Johnson, K. and Goldwasser, D. (2016). Identifying stance by analyzing political discourse on twitter. *Proceedings of the First Workshop on NLP and Computational Social Science 2016*, pages 66–75.

Kiritchenko, S. and Mohammad, S. M. (2016). Capturing

reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, San Diego, California.

Kiritchenko, S. and Mohammad, S. M. (2017). Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of The Annual Meeting of the Association for Computational Linguistics (ACL)*, Vancouver, Canada.

Kwon, N., Zhou, L., Hovy, E., and Shulman, S. W. (2007). Identifying and classifying subjective claims. In *Proceedings of the 8th annual international conference on Digital government research: bridging disciplines & domains*, pages 76–81. Digital Government Society of North America.

Ladner, A. and Pianzola, J. (2010). Do voting advice applications have an effect on electoral participation and voter turnout? Evidence from the 2007 Swiss Federal Elections. *Electronic participation*, pages 211–224.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Louviere, J. J., Flynn, T. N., and Marley, A. A. J. (2015a). *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge University Press.

Louviere, J. J., Flynn, T. N., and Marley, A. A. (2015b). *Best-worst scaling: Theory, methods and applications*. Cambridge University Press.

Louviere, J. J. (1991). Best-worst scaling: A model for the largest difference judgments. Working Paper.

Louwerse, T. and Rosema, M. (2014). The design effects of voting advice applications: Comparing methods of calculating matches. *Acta politica*, 49(3):286–312.

Lyon, C., Malcolm, J., and Dickerson, B. (2001). Detecting short passages of similar text in large document collections. In *Proceedings of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, pages 118–125, Pittsburgh, USA.

Misra, A., Anand, P., Tree, J., and Walker, M. (2015). Using summarization to discover argument facets in online idealogical dialog. In *Proceedings of the NAACL HLT*, pages 430–440, Denver, USA.

Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016). SemEval-2016 Task 6: Detecting Stance in Tweets. In *Proceedings of the International Workshop on Semantic Evaluation 2016*, San Diego, USA.

Mohammad, S. M., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Mohammad, S. M. (2016). Sentiment analysis: Detecting valence, emotions, and other affectual states from text. *Emotion Measurement*, pages 201–238.

Orme, B. (2009). Maxdiff analysis: Simple counting, individual-level logit, and HB. *Sawtooth Software*.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Peldszus, A. and Stede, M. (2016). An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon 2015 / Vol. 2*, pages 801–815, London, UK. College Publications.

Rosenthal, S. and McKeown, K. (2012). Detecting opinionated claims in online discussions. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, pages 30–37. IEEE.

Stab, C. and Gurevych, I. (2014). Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 46–56. Association for Computational Linguistics.

Taulé, M., Martı, M. A., Rangel, F., Rosso, P., Bosco, C., and Patti, V. (2017). Overview of the task of stance and gender detection in tweets on catalan independence at ibereval 2017. In *Notebook Papers of 2nd SEPLN Workshop on Evaluation of Human Language Technologies for Iberian Languages (IBEREVAL), Murcia, Spain, September*, volume 19.

Titov, I. and McDonald, R. (2008). A Joint Model of Text and Aspect Ratings for Sentiment Summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, USA.

Trabelsi, A. and Zaıane, O. R. (2014). Finding Arguing Expressions of Divergent Viewpoints in Online Debates. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)@ EACL*, pages 35–43, Gothenburg, Sweden.

Tsur, O., Calacci, D., and Lazer, D. (2015). A Frame of Mind: Using Statistical Models for Detection of Framing and Agenda Setting Campaigns. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1629–1638, Beijing, China.

Walker, M. A., Tree, J. E. F., Anand, P., Abbott, R., and King, J. (2012). A Corpus for Research on Deliberation and Debate. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2012)*, pages 812–817, Istanbul, Turkey.

Wall, M., Sudulich, M. L., Costello, R., and Leon, E. (2009). Picking your party online–an investigation of ireland's first online voting advice application. *Information Polity*, 14(3):203–218.

Xu, R., Zhou, Y., Wu, D., Gui, L., Du, J., and Xue, Y. (2016). Overview of NLPCC Shared Task 4: Stance Detection in Chinese Microblogs. In *International Conference on Computer Processing of Oriental Languages*, pages 907–916, Kunming, China.

Zhuang, L., Jing, F., and Zhu, X.-Y. (2006). Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 43–50. ACM.

Appendix

Table 2 summarizes the information provided by the participants in the demographic survey. Note that 85 (37%) of the total 230 participants submitted responses to the demographic survey. For each of the sixteen issues explored in this study, Tables 3, 4 and 5 list assertions that most people agree with, assertions that most people disagree with, and assertions that are the most controversial. Below, we show an example questionnaire used for collecting assertions and an example questionnaire used for collecting judgments on the collected assertions:

QUESTIONNAIRE I: CREATING ASSERTIONS ON CONTROVERSIAL ISSUES

Provide at least five relevant assertions on the given controversial issue. The assertions must be expressions that one can agree or disagree with. They can be claims, beliefs, opinions, reasons, arguments, or any statement that can be used to inform or support one's position on the issue. The assertions do not have to be reflective of your own opinions. The assertions can be about a sub-issue or an aspect of the issue.

The assertions should:

- support a position that is relevant to the issue.
- cover a diverse set of positions on the issue. (Avoid claims that rephrase the same argument in slightly different ways.)
- be formulated in a way that a third person can agree or contradict the assertion.
- be self contained and understandable without additional context. (Do not use 'it', 'she/her' or 'he/him/his' etc. to refer to an issue, a person or something else that is not directly mentioned in your assertion.)
- be precise. (Avoid vague formulations such as maybe, perhaps, presumably or possibly.)

The assertions should NOT:

- be a simple expression of agreeing/supporting or disagreeing/rejecting the overall issue.
- contain multiple positions (e.g. migrants are friendly and hardworking).
- contain expressions of personal perspective (e.g. I don't like immigrants).
- be the same as any of the provided examples; or simple negations or other variants of a provided example.

Issue:	Marijuana
Description:	This issue is about legalization of cannabis. This includes the legalization for recreational or medical usage and other positive or negative consequences of legalizing cannabis.

Q1: True or False: This issue is about risks of consuming Cocaine.

( ) true

(✓) false

Q2: Choose the assertion which meets the format requirements:

(✓) The government should discourage any drug usage.

( ) Maybe, the government should discourage any drug usage.

Q3: Enter assertion 1 about 'Marijuana': _____
Q4: Enter assertion 2 about 'Marijuana': _____
Q5: Enter assertion 3 about 'Marijuana': _____
Q6: Enter assertion 4 about 'Marijuana': _____
Q7: Enter assertion 5 about 'Marijuana': _____

QUESTIONNAIRE II: JUDGING ASSERTIONS ON CONTROVERSIAL ISSUES

We want to better understand common controversial issues such as immigration and same-sex marriage. Therefore, we have collected a large amount of assertions relevant to these issues. Your task is to:

- Indicate whether you agree or disagree with these assertions.
- Indicate how strongly you support or oppose these assertions. Since it is difficult to give a numerical score indicating the degree of support or degree of opposition, we will give you four assertions at a time, and ask you to indicate to us:
  - Which of the assertions do you support the most (or oppose the least)?
  - Which of the assertion do you oppose the most (or support the least)?
  - *If you support two assertions equally strongly, then select any one of them as the answer. The same applies for oppose.*

Q1: Indicate whether you agree or disagree with the given assertions on the issue 'Black Lives Matter'.

- Every race has experienced racism.
  ( ) agree ( ) disagree
- There is racial discrimination in the U.S..
  ( ) agree ( ) disagree
- The Black lives matter movement is important.
  ( ) agree ( ) disagree
- Black Lives Matter encourages racial hate.
  ( ) agree ( ) disagree

Q2: Which of these assertions on the issue 'Black Lives Matter' do you support the most (or oppose the least)?

( ) Every race has experienced racism.

( ) There is racial discrimination in the U.S..

( ) The Black lives matter movement is important.

( ) Black Lives Matter encourages racial hate.

Q3: Which of these assertions on the issue 'Black Lives Matter' do you oppose the most (or support the least)?

( ) Every race has experienced racism.

( ) There is racial discrimination in the U.S..

( ) The Black lives matter movement is important.

( ) Black Lives Matter encourages racial hate.

Variable	Answers	%
Education	Bachelor	39%
	some college	15%
	high school	12%
	Master	12%
	associate degree	11%
	vocational certification	9%
	PhD	2%
Race	White	69%
	Hispanic	10%
	Black / African American	8%
	Asian	8%
	other	3%
	American Indian	2%
Religion	Roman Catholic	28%
	Atheist / Agnostic	25%
	Protestant	19%
	other	11%
	Muslim	2%
	Buddhist	2%
	Jewish	1%
	Russian Orthodox	1%
Affiliation	Democrat	45%
	independent	22%
	none	16%
	Republican	15%
	other	1%
Family Status	married	29%
	single (living alone)	28%
	single (living with partner)	20%
	single (living with parents)	19%
	other	4%
Profession	employee	49%
	unemployed	14%
	self-employed	13%
	student	9%
	other	6%
	retired	6%
	civil servant	2%
U.S. Citizen	yes	89%
	no	11%
Ties to overseas	yes	73%
	no	27%

Table 2: Demographic information of the 85 subjects that participated in the voluntary demographic survey.

Issue	Metric	Top 3 Assertions
Black Lives Matter	highest agreement assertions	Every human is equal. Black people are as human as white people. We can all make a better world if we work together.
	most controversial assertions	People of color are more likely to be born into poverty. The Black Lives Matter movement encourages racial hate. Police racial profiling reduces minorities.
	lowest agreement assertions	Blacks are the scum of society. The world would be safer without black people. Not all people are equal.
Climate Change	highest agreement assertions	Global warming can change our climate. Climate change has a big affect on the Earth. Climate change will cause problems for future generations.
	most controversial assertions	Different changes in weather does not mean global warming. The climate change is caused by most developed country. Rising levels of atmospheric CO2 do not necessarily cause global warming.
	lowest agreement assertions	Global warming is not real. Global warming has nothing to do with the change in weather. Trump won the election due to his positions on the environment.
Creationism in school	highest agreement assertions	Freedom of expression is vital to our liberties. Each person should be free to choose what to believe in. People in schools should not be made fun of for their religion.
	most controversial assertions	Religion has no place in a school. Teaching creationism's truth is necessary in upholding the truth of the Bible. Creationism should not be tought in a classroom because it goes against some beliefs.
	lowest agreement assertions	Atheists should be banned from schools. Creationism should include just Christian beliefs versus evolution. Christian religion should be promoted in schools.
Foreign Aid	highest agreement assertions	People have the right to know where there taxes are going and why. We should clearly know where our funding is going and why. Foreign aid should be corruption free
	most controversial assertions	Helping other countries will only increase those nations' dependence on the U.S. Stop borrowing money and raise tax payers dollars. Foreign aid lead to further difficulties for both countries.
	lowest agreement assertions	US spending should focus on defence rather than aid. Foreign aid should be focused on African countries. Foreign aids uplift corruption.
Gender Equality	highest agreement assertions	All people should be treated equally. Wages should not be based on gender. Women can be as successful as men.
	most controversial assertions	Instead of fighting for gender equality, we must fight for gender equity. A woman's physical condition makes her unsuitable for certain jobs. Places don't hire people based on gender.
	lowest agreement assertions	Women should not be in the army (direct combat forces). Gender equality is stupid. The wage gap is a made up thing.
Gun Rights	highest agreement assertions	Gun owners need to be required to have a background check. Gun owners should register their arms. Gun owners should be required to take a gun safety course.
	most controversial assertions	In a certain part to eliminate the arms would be to end the delinquency. Guns should only be issued for hunting. People who own guns are not more likely to mass kill.
	lowest agreement assertions	Everyone should own a gun. The gun industry is too heavily regulated. Guns should be legal for everyone.
Marijuana	highest agreement assertions	Drug abuse can kill. Exceptions should be considered allow people with medical issues that will benefit from legalization of Marijuana. Marijuana is proven to have medical benefits.
	most controversial assertions	Legalization of marijuana will result to people not pushing hard drugs. Marijuana use will not increase just because it is legalized. People who use marijuana are mentally stuck at the age they were when they began using the drug.
	lowest agreement assertions	Marijuana should be as readily available as cigarettes are. Allowing the legal use of marijuana will prevent drug dealers from selling it. Cannabis is nontoxic

Table 3: The top three ranked assertions according to different metrics of the agreement score (issues 1-7).

Issue	Metric	Top 3 Assertions
Same-sex Marriage	highest agreement assertions	God loves all the people for equal. One person's believes should not affect another persons rights. Love is a right for everyone.
	most controversial assertions	Same sex marriage is unlike any other marriage. Allowing same-sex marriage will create a 'slippery slope' where people will begin to fight for other alternative marriages such as polygamy. Marriage was God's idea. He defined it as between a man and a woman only. End of conversation.
	lowest agreement assertions	Saying that gay people should get married is like saying that a brother can marry his sister both are at higher risk of disease. There is more domestic abuse in homosexual couples. Children of same-sex couples will turn out gay.
Mandatory Vaccination	highest agreement assertions	The vaccines must be given by people who are very prepared to put them. Vaccines should be free. Medical treatments like vaccines should be for all the people.
	most controversial assertions	Vaccines can cause serious and fatal side effects. Vaccinations are expensive and yet not 100% safe. The theory of herd immunity has never been proven.
	lowest agreement assertions	Vaccinations cause autism. Vaccines increase the risk of autism. Vaccines are not safe.
Media Bias	highest agreement assertions	The job of a reporter is to tell the truth. Every body has its owns preferences. The society deserves the truth.
	most controversial assertions	If you don't watch the media , bias isn't a problem. Journalists only cover stories that support their own opinion. Media is liberal biased because the liberals are right.
	lowest agreement assertions	Fox News is the only unbiased news source. Journalism focusses on Democrats views. Fake news is not a real problem.
Obama Care	highest agreement assertions	Every deserves a chance at life. Everyone needs insurance. Health care should not punish people for pre-existing conditions.
	most controversial assertions	People may rely on free health care to substitute there own role in maintaining unhealthy lifestyle choices. Obama care should be removed. An unregulated market ensures cheap medicine.
	lowest agreement assertions	Medical treatments should be only for people that can afford treatments. By giving people free health care it can lead to them being lazy. Free health care leads to people not taking care of their health.
US Electoral System	highest agreement assertions	The people should elect there leader. The electoral system must be fair. States should do what they can to avoid voter fraud.
	most controversial assertions	The electoral college protects minority interests. Popular vote should never be used as a determining factor because it give total control of the country to large coastal states. Our good electoral system puts US as a world example.
	lowest agreement assertions	Voting for an independent candidate only ensures that the Republican will win. Gerrymandering is a valid system. The voting age should be changed to 16.
US in the Middle East	highest agreement assertions	US Engagement in the middle east needs to be carefully addressed. US should protect America against terrorism. Middle east countries should be allowed to determine their own future.
	most controversial assertions	Withholding funds to Palestinian National Authority prevents anti-Semitism. Usa's engagement in middle east is not a public interest topic. The US needs to show its support for Israel by moving its embassy to Jerusalem. It would make a big statement to the world.
	lowest agreement assertions	All Muslims should be regarded as terrorists. It is our duty to spread Western ideals to the Middle East. The war in Iraq was worth the costs.
US Immigration	highest agreement assertions	Migrants have a positive impact on the economy. A person in need should not be denied help and life regardless of where they are from. Legal immigration has brought some great scientists to the United States.
	most controversial assertions	Immigration takes opportunities away from those born in the USA. Immigration should be elevated because it brings the best brains to the economity. Illegal immigrants take jobs away from Americans.
	lowest agreement assertions	Immigrants are more often criminals. Immigrants are bad for the US. Immigrants are better workers than US workers.

Table 4: The top three ranked assertions according to different metrics of the agreement score (issues 8-14).

Issue	Metric	Top 3 Assertions
Vegetarianism & Veganism	highest agreement assertions	People have the right to choose the kind of meal they want to eat. A persons diet should be there choice and the individual freedoms of choosing one food should not be shamed. Creating a nutrition plan is left for the person.
	most controversial assertions	Vegetarianism is healthier than veganism. Killing animals is harmful. Veganism is like a religion.
	lowest agreement assertions	Killing animals is against Gods law. I like cows, so I don't eat them. Humans are meant to be vegan.
War in Terror	highest agreement assertions	Terrorism is an international threat. The government must take care that the taken measures on counter terror does not affect innocents. Terrorism has destroyed too much and needs to end.
	most controversial assertions	As a bonus, war on terror almost guarantees safety worldwide. The war on terrorism is an invasion of privacy. It is ok that our taxes support the war on terrorism.
	lowest agreement assertions	All Muslims are terrorists. Terrorism is only a problem in the middle east. War on terriorism kept us safe so far.

Table 5: The top three ranked assertions according to different metrics of the agreement score (issues 15 & 16).

Distribution of Emotional Reactions to News Articles in Twitter

Omar Juárez Gambino* †**, Hiram Calvo**†**, Consuelo-Varinia García-Mendoza***
*Instituto Politécnico Nacional - ESCOM, J.D. Bátiz, Lindavista, G.A. Madero, Mexico City, 07738, Mexico
†Instituto Politécnico Nacional - CIC, J.D. Bátiz, Nueva Industrial Vallejo, G.A. Madero, Mexico City, 07738, Mexico
b150697@sagitario.cic.ipn.mx, hcalvo@cic.ipn.mx, cvgarcia@ipn.mx

Abstract

Several datasets of opinions expressed by Social networks' users have been created to explore Sentiment Analysis tasks like Sentiment Polarity and Emotion Mining. Most of these datasets are focused on the writers' perspective, that is, the post written by a user is analyzed to determine the expressed sentiment on it. This kind of datasets do not consider the source that provokes those opinions (e.g. a previous post). In this work, we propose a dataset focused on the readers' perspective. The developed dataset contains news articles published by three newspapers and the distribution of six predefined emotions expressed by readers of the articles in Twitter. This dataset was built aiming to explore how the six emotions are expressed by Twitter users' after reading a news article. We show some results of a machine learning method used to predict the distribution of emotions in unseen news articles.

Keywords: Twitter Sentiment Analysis, Reader's emotions, Emotion distribution

1. Introduction

Social media have fostered new ways of interaction between writers and readers. The writer of a post can receive several responses from readers as soon as it is available, and then a direct interchange of opinions begins. This particular form of communication has attracted attention of the Sentiment Analysis community because both writers and readers express emotions during their conversation. Twitter has become a popular media where users interchange opinions with strong emotional content. Since 2013 SemEval workshop has organized a Twitter Sentiment Analysis task (Nakov et al., 2013). Specific objectives of this task have changed over time, but all released datasets since 2013 have contained tweets and annotations related to sentiment polarity or emotions expressed on them. The Spanish Society of Natural Language Processing (SEPLN for its acronym in Spanish) has organized a similar task but specifically for Spanish since 2012 (Villena Román et al., 2013). The dataset for this workshop is composed by tweets and annotations corresponding to the sentiment polarity of each one. On both workshops the datasets were created focusing on writer's perspective; this means that the proposed tasks use the text written in the opinions to predict the sentiment polarity or emotions on them.

On the other hand, there have been fewer efforts focused on the reader's perspective. In this case, the idea is to determine which emotions or sentiment polarity would be expressed by a user as a reaction after reading a post. From this perspective, datasets comprise the written text read by users (e.g. a news article) and annotations corresponding to sentiment polarity or emotions expressed by the users. Predicting the sentiment polarity or emotions of readers is considered particularly difficult, because the text where readers express their opinions is not available; instead, the written text that was read by them is used for the prediction. Lin et al. (2007) and Lin et al. (2008) created datasets with news articles from Yahoo! China and the emotions users expressed after reading the articles, from a fixed set of 8 emotions. The authors pointed out that the emotions expressed in the news content are not necessarily the same as those expressed by the readers; this is an important characteristic that should be considered. Rao (2016) and Li et al. (2017) collected datasets with news articles and user ratings across 8 emotions from Sina website[1]. The tasks defined for these datasets are to predict the predominant emotion (also known as Acc@1) of each article and the percentage of votes that users will express for the defined set of emotions in each article (distribution of emotions). It is worth to mention that in all these three works, readers express their emotions by choosing emoticons from a fixed set defined by the websites. Authors argue that emoticons are related to some specific emotions, even though readers do not express their opinions in a written form.

In this work, we propose a dataset composed by news articles from three different newspapers and the corresponding distribution of emotions, from a predefined set of six emotions, expressed in texts by readers in Twitter. Our dataset is focused on reader's perspective but, unlike the aforementioned works above, the emotions are labeled by human annotators by analyzing the content of each tweet instead of a fixed emoticon set. To our knowledge there is no available dataset of news articles with their corresponding distribution of emotions from Twitter users.

2. Dataset development

News articles report information about current events. Journalists are expected to describe these events based on real facts and following three important principles: impartiality, neutrality and objectivity. From this point of view, news articles are supposed to lack personal opinions, emotions or anything additional to the reported facts. Although there are no explicit emotions in news, reported facts may trigger emotions in readers. For instance, consider the following headline news extracted from a newspaper. "Call for punishment for man who intentionally hit dog" (originally in Spanish). In the example it is clear that a factual event is reported, but a set of emotions like sadness and anger could be triggered on readers because of the news content. Therefore, we considered that news articles are a good source of emotional reactions, specially those published in Twitter.

[1] http://news.sina.com.cn/society/

2.1. Newspaper selection

There are several newspapers that have an online version of their articles. Some of them require a paid subscription to access articles while others have free access. We considered that free access allows more users to read the articles, so more and better responses can be available. Therefore, we discarded newspapers that required a paid subscription. Another aspect to consider is the kind of content that newspapers publish. There are newspapers specialized on subjects like finance or sports, while others include general information of different topics. We do not want to limit the scope of the dataset to specialized subjects because we think that any kind of news contents can trigger emotions on readers. The last consideration is more subjective because it is related to the bias of newspapers. We have identified that certain sensible topics like politics are published with different focus. Even though it is not our intention to affirm that a specific newspaper has bias to certain political preferences, we want to include diversity of opinions in the dataset. So at the end, we have selected three Mexican newspapers that comply with these characteristics. El Universal[2], La Jornada[3] and Excelsior[4]. In order to verify all of the above assumptions, a manual review of the collected data was carried out, finding that they were suitable for our purposes.

The three selected newspapers have active Twitter user accounts. These accounts are used to publish headlines of news articles and to share them with other Twitter users. Once users read an article, tweet responses (hereinafter replies) are posted. By analyzing these replies it is possible to identify emotions that the article triggered on users. Therefore, in our proposal we consider that it is possible to use replies as readers' emotional reactions to news articles. To illustrate this interaction between newspaper and readers, we present the following example.

News article headline published by La Jornada (originally in Spanish): "@lajornadaonline: He is not my president!, they shout in US cities after Trump's victory." A sample of its replies (translated from Spanish):

- @user1: @lajornadaonline @realDonaldTrump They are making pressure, and it is obvious that this is not necessarily going to change the state of things...
- @user2: @lajornadaonline Polarization has generated radicalizations that threaten to rise tone
- @user3: @lajornadaonline @realDonaldTrump That is democracy, accept it; it was a plot.

2.2. News and replies gathering

Twitter provides official APIs to access posts from different programming languages. The most serious limitation of the them is that there is no method available to get all post's replies, even though this request was made since 2008 to Twitter developers[5]. We have implemented an indirect process to get news articles and their replies. The pseudocode of the process is shown in Algorithm 1.

Algorithm 1: Algorithm to get news articles and their replies

```
function getNewsAndReplies (q);
  Input  : q is the query that specifies the data we want
           to get
  Output: A list of news and their replies
replies = searchReplies(q);
foreach reply in replies do
    replyID = reply.ID;
    replyContent = reply.content;
    newsID = getInReplyTo(replyID);
    url = getURLNews(newsID);
    newsContent = getNewsContent(url);
    save(newsContent, replyContent);
end
```

As can be seen in this algorithm, instead of obtaining the contents of the news and then the contents of the replies, we have to proceed the other way around. Contents of the news article are obtained by parsing the source code of the web page specified in the URL of the tweet. The procedure for gathering news and tweets was applied to the three selected newspapers and in the query q of Algorithm 1 we specified to recover replies published from 01-01-2016 to 01-01-2017. Table 1 shows the number of news articles and replies collected by newspaper. The number of replies may vary between newspapers depending on the number of readers the newspaper had and the impact of the articles.

Table 1: Number of news articles and replies collected

Newspaper	News articles	Replies
El Universal	90	1,000
Excelsior	100	1,136
La Jornada	98	1,406
Total	*288*	*3,542*

2.3. Set of emotions selection

The study of emotions is a multidisciplinary field which includes Neuroscience, Psychology and Cognitive Sciences. Every discipline has its own objectives and perspectives, in such a way that it has been historically hard to agree in a definition of what an emotion is (Kleinginna and Kleinginna, 1981). Nevertheless, there are well-founded proposals on types of emotions and how they are generated (Ortony and Turner, 1990). A remarkable study of emotions was done in Shaver et al. (1987), in which the authors argue that emotional knowledge should be represented hierarchically. To illustrate that, an experiment was developed, in which 112 Psychology students were asked to rate 213 words used to express emotions. By applying mean prototypicality rating, 135 words were selected as good emotions descriptors. Then another 100 psychology students were asked to do a similarity sorting over the

[2]http://www.eluniversal.com.mx/
[3]http://www.jornada.unam.mx
[4]http://www.excelsior.com.mx/
[5]https://code.google.com/archive/p/twitter-api/issues/142

135 words. The sorting resulted in a hierarchy with six main groups and every group had 2 internal groups. At the end, the authors proposed a 3 level hierarchy, with the first level containing basic emotions, while the second and third levels, represented more specific emotions. In Tables 2 and 3 we show the emotions, translated to Spanish, of every group and its corresponding level. It is important to mention that the number of words used to describe emotions was reduced from the original 135 to 110 because some of them are translated using the same word in Spanish [6]. We will adhere to this proposal and the words listed in the tables will be used as labels to tag the emotions in replies during the annotation process.

2.4. Annotation process

In order to identify emotions in replies, we selected 4 annotators: 3 undergraduate Computer Science students and a Computer Science professor. They were provided with 288 news articles collected from the three selected newspapers and 3,542 replies (with an average of 11 replies per article). Annotators were asked to follow the next procedure for each news article:

1. Read each reply to the article.
2. Identify the emotions expressed in each reply.
3. Mark all emotions identified in the replies using the words (tags) of Tables 2 and 3.

It was emphasized to the annotators that the emotions that should be annotated are those expressed in the replies, and not those that the news article triggered in them.

According to their experience when tagging emotions, it was easier for annotators to have an extensive list of labels than only the six ones defined in level 1 of each group: *love*, *joy*, *surprise*, *anger*, *sadness* and *fear*. As a result, we obtained a fine-grained annotated dataset. Although identifying very specific emotions such as *loathing* and *exasperation* is interesting, in this work we want to predict the general emotions expressed by Twitter users after reading a news article[7]. Therefore, all the emotions marked by the annotators were generalized to the emotions in level 1 because they are considered basic emotions and provide a good overall idea of the sentiment expressed in replies. To generalize emotions, we used the defined groups and replaced the words of levels 2 and 3 with the word in level 1. For instance, let us consider that an annotator tagged a reply with the labels *enjoyment*, *relief* and *surprise*. By looking in group 2, we identified that the labels enjoyment and relief are related to the basic emotion *joy*, while *surprise* is already the basic emotion of group 3 and hence the generalized tags are *joy* and *surprise*. These generalized tags are the ones that were used in the created dataset.

It is important to measure the agreement in tasks performed by humans. The higher the agreement, the better the quality of the dataset; but in this particular task there are some factors that make difficult to expect a high agreement. First of all, the content of news articles often leads to controversial opinions, for instance political stances, and in these kind of opinions is not easy to identify emotions because they are not explicit expressed or they contain sarcasm. In addition to that, the background of annotators may lead to variation in the identified emotions. Finally, because of the number of annotators and emotions to identify, a perfect agreement

Table 2: Emotions provided to annotators

Group 1	
Level	Emotions
1	**Amor (Love)**
2	Afecto (Affection), Deseo (Lust), Anhelo (Longing)
3	Adoración (Adoration), Amor (Love), Cariño (Fondness), Afición (Liking), Atracción (Attraction), Ternura (Tenderness), Compasión (Compassion), Sentimentalismo (Sentimentality), Agitación (Arousal), Deseo (Lust), Pasión (Passion), Encaprichamiento (Infatuation), Anhelo (Longing)
Group 2	
Level	Emotions
1	**Alegría (Joy)**
2	Alegría (Joy), Entusiasmo (Zest), Contentamiento (Contentment), Orgullo (Pride), Optimismo (Optimism), Encanto (Enthrallment), Alivio (Relief)
3	Diversión (Amusement), Dicha (Bliss), Alegría (Cheerfulness), Regocijo (Glee), Jovialidad (Joviality), Deleite (Delight), Placer (Enjoyment), Felicidad (Happiness), Júbilo (Jubilation), Euforia (Elation), Satisfacción (Satisfaction), Éxtasis (Ecstasy)
Group 3	
Level	Emotions
1	**Sorpresa (Surprise)**
2	Sorpresa (Surprise)
3	Asombro (Amazement), Sorpresa (Surprise), Estupor (Astonishment)
Group 4	
Level	Emotions
1	**Enojo (Anger)**
2	Irritación (Irritation), Exasperación (Exasperation), Rabia (Rage), Disgusto (Disgust), Envidia (Envy), Tormento (Torment)
3	Irritación (Irritation), Agitación (Agitation), Molestia (Annoyance), Mal humor (Grouchiness), Exasperación (Exasperation), Enojo (Anger), Rabia (Rage), Furia (Fury), Ira (Wrath), Hostilidad (Hostility), Agresividad (Ferocity), Rencor (Bitterness), Odio (Hate), Aversión (Loathing), Desprecio (Scorn), Desagrado (Dislike), Resentimiento (Resentment), Disgusto (Disgust), Envidia (Envy), Celos (Jealousy), Tormento (Torment)

[6]For further details of the original hierarchy and emotions see (Shaver et al., 1987)

[7]Prediction of emotions in fine-grained level as well as the use of the hierarchy are considered as future work

Table 3: Emotions provided to annotators 2

Group 5	
Level	Emotions
1	**Tristeza (Sadness)**
2	Sufrimiento (Suffering), Tristeza (Sadness), Decepción (Disappoinment), Pena (Shame), Desamparo (Neglect), Lástima (Sympathy)
3	Agonía (Agony), Sufrimiento (Suffering), Dolor (Hurt), Angustia (Anguish), Depresión (Depression), Desesperación (Despair), Desesperanza (Hopelessness), Pesadumbre (Gloom), Abatimiento (Glumness), Tristeza (Sadness), Infelicidad (Unhappiness), Aflicción (Grief), Pesar (Sorrow), Miseria (Misery), Melancolía (Melancholy), Consternación (Dismay), Decepción (Disappointment), Disgusto (Displeasure), Culpa (Guilt), Pena (Shame), Arrepentimiento (Regret), Remordimiento (Remorse), Aislamiento (Alienation), Desamparo (Neglect), Soledad (Loneliness), Rechazo (Rejection), Nostalgia (Homesickness), Derrota (Defeat), Abatimiento (Dejection), Inseguridad (Insecurity), Vergüenza (Embarrassment), Humillación (Humiliation), Insulto (Insult), Pena (Pity), Lástima (Sympathy)
Group 6	
Level	Emotions
1	**Miedo (Fear)**
2	Horror (Horror), Ansiedad (Nervousness)
3	Sobresalto (Alarm), Conmoción (Shock), Miedo (Fear), Temor (Fright), Horror (Horror), Terror (Terror), Pánico (Panic), Histeria (Hysteria), Mortificación (Mortification), Ansiedad (Anxiety), Nerviosismo (Nervousness), Tensión (Tenseness), Inquietud (Uneasiness), Aprensión (Apprehension), Preocupación (Worry), Angustia (Distress)

happens when the four annotators tag a reply with the same emotions from a set of six non-exclusive labels (love, joy, surprise, anger, sadness and fear) and this is already an unlikely condition. Despite all the above, the inter-annotator agreement was 0.48 using a generalized version of Cohen's kappa (multi-kappa) (Davies and Fleiss, 1982). This value is considered by Landis and Koch (1977) as a moderated inter-annotator agreement and we also consider it so for the purposes of this work.

2.5. Distribution of emotions

Users that reply to news articles can express multiple emotions in their posts. These emotions can be counted in order to determine the frequency (*votes*) of each emotion. To illustrate this idea, let us consider that a news article had ten replies and for each reply a set of the six emotions previously defined have been tagged by the annotators. Table 4 shows tagged emotions indicated with a checkmark.

It is possible to represent the information of Table 4 in terms of percentage applying a normalization process, dividing each count of tagged emotions by the number of replies, as shown in the last row. These percentage values is what we call *distribution of emotional reaction* and can be interpreted as the degree on which of these emotions were expressed by readers as a reaction to a news article. For instance, the news article of Table 4 provoked 50% of love, 100% of joy, 40% of surprise, 30% of sadness, 70% of anger and 0% of fear. It is worth mentioning that percentage values of emotions are independent because a news article may provoke 100% of joy and 70% of anger at the same time. Therefore, the sum of these values do not necessarily sum 100%. We have decided to use percentage ranges instead of total votes to specify the distribution of emotions in the dataset. The possible values for each emotion is composed by a set of 11 ranges from 0% to 100% with 10% increment, that is {0%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 100%}. The same normalization process was applied to each set of responses in order to have all news articles associated with their corresponding distribution of emotions.

In Figures 1, 2, 3 we show the distribution of percentage values per emotion in articles of each newspaper. We have also created an extra dataset that integrates the news articles and their corresponding tags of the three newspaper, which allows to test a more generalized prediction of emotion distribution. Emotions have different distribution values in news articles as can be seen in Figure 1, where 50% of anger was expressed as a reaction in 7 articles while the same percentage of sadness was expressed in 20 articles. In addition to that, the integrated version of newspapers (see Figure 4) clearly shows that some emotions are less frequently expressed in replies to news articles. Particularly love and surprise were not expressed at all (0%) in 229 and 189 articles respectively, it means that less than 40% of articles provoked those emotions. This characteristic leads to a unbalanced corpus in which some percentage values are much more frequent than others.

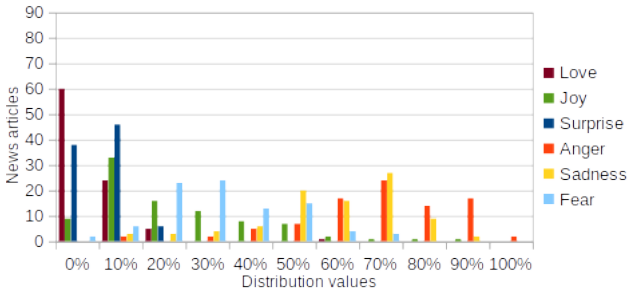


Figure 1: El Universal.

3. Data format

Data are presented in four different CSV text files. Each file is related to a specific newspaper and the integrated version of the three newspapers. Files contain one news article per line. Each line has three separated elements: tweet ID, the content of the news article, and the tagged distribution of emotions. Elements are separated by the

Table 4: Reactions of users to a news article

Responses	Emotions					
	Love	Joy	Surprise	Sadness	Anger	Fear
R_1		✓	✓			
R_2	✓	✓		✓	✓	
R_3		✓			✓	
R_4	✓	✓	✓		✓	
R_5		✓				
R_6	✓	✓			✓	
R_7		✓	✓	✓		
R_8	✓	✓			✓	
R_9		✓			✓	
R_{10}	✓	✓	✓	✓	✓	
Total	*5/10*	*10/10*	*4/10*	*3/10*	*7/10*	*0/10*
Normalized total	*0.5*	*1*	*0.4*	*0.3*	*0.7*	*0*

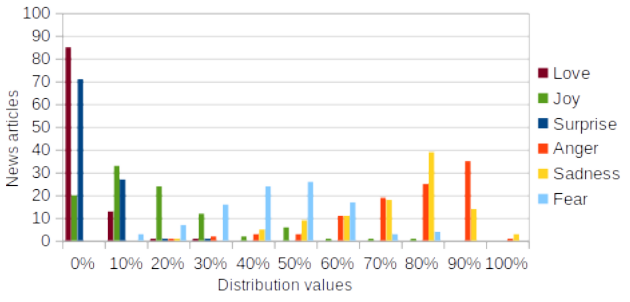


Figure 2: Excelsior.

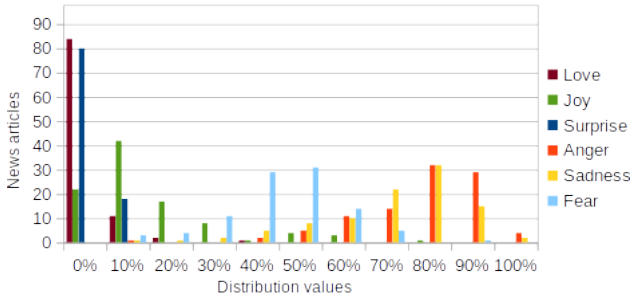


Figure 3: La Jornada.

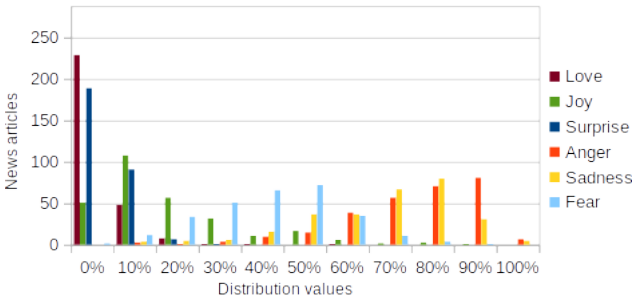


Figure 4: Integrated newspapers.

string "&&". A tuple of six elements is used to describe the distribution of emotions. The order of the elements in the tuples is associated with the emotions love, joy, surprise, anger, sadness and fear respectively. The values of the distribution of emotions are expressed in decimal form. An example of an annotated news article is the following: 123&&This is the content of the article&&0.1,0.2,0.1,0.7,0.8,0.5

4. Possible uses and early experiments

The distribution of emotions annotated in the proposed dataset is useful for several Sentiment Analysis tasks. Let us consider the distribution of emotions shown in Table 4 (50%, 100%, 40%, 30%, 70% and 0%). Given a news article, it is possible to determine the predominant emotion (Joy)—cf. (Lin et al., 2007), determine the ranking of emotions (Joy, Anger, Love, Surprise, Sadness and Fear)—cf. (Lin et al., 2008) and of course the very same distribution of emotions —cf. (Rao, 2016), (Li et al., 2017). Practical applications of predicting the distribution of emotions could be the following:

- Assisted writing for provoking particular emotions.
- Recover news articles that generate specific emotions (e.g. news articles that cause surprise)

In order to explore how machine learning methods could use this dataset for predicting distribution of emotions, we have performed some experiments. Because users can express more than one emotion in their replies and each emotion is associated with 11 possible distribution values (from 0% to 100%), we decided to use a multi-target classification strategy (Herrera et al., 2016) that supports multi-dimensional problems such as this. This strategy follows a supervised approach by using well known methods like Naïve Bayes and SVM as base classifiers. As a metric we used the average Pearson Correlation (AP). This metric has been used in (Rao, 2016) and (Li et al., 2017) to compare the predicted distribution of emotions against the real ones. Values of this metric ranges from -1 to 1, where 1 indicates a perfect positive correlation. Table 5 shows the average results from 10-fold cross validation, using 90% of data for training and 10% for testing. We believe that, when a newspaper has a clear political bias (left-wing or right-wing),

most of its readers will share the same ideologies and they will express similar emotions in their replies. On the other hand, when a newspaper has an intermediate position (centrist), the opinions expressed by its readers will be diverse as well as their emotions. This idea is reflected in the results of Table 5 in which La Jornada (left-wing) obtained the best results, following by Excelsior (right-wing), and finally El Universal (centrist). The Integrated version is a kind of average between the results of the three newspapers. Further details of the experiments are described in (Gambino and Calvo, 2018).

Table 5: AP results by each newspaper

Newspaper	AP
Excelsior	0.9241
La Jornada	0.9274
El Universal	0.8494
Integrated	0.8997

5. Conclusions and future work

In this work we presented a dataset of news articles and their corresponding distribution of emotions expressed by readers in Twitter. This dataset could be used for several Sentiment Analysis tasks. The expressed emotions are from the readers' perspective; and, the available datasets of this kind are scarce. Early experiments were performed to explore the application of machine learning methods for predicting emotion distribution and results seem promising. Increasing the size of the corpus is proposed as a future work.

6. Acknowledgements

We thank the support of Insituto Politécnico Nacional (IPN), ESCOM-IPN, CIC-IPN, SIP-IPN projects numbers: 20181102, 20180859, 20182114, COFAA-IPN and EDI-IPN.

7. Bibliographical References

Davies, M. and Fleiss, J. L. (1982). Measuring Agreement for Multinomial Data. *Biometrics*, 38(4):1047–1051.

Gambino, O. J. and Calvo, H. (2018). Modeling distribution of emotional reactions in social media using a multi-target strategy. *Jounal of Intelligent & Fuzzy Systems*, in press.

Herrera, F., Charte, F., Rivera, A. J., and Del Jesus, M. J. (2016). *Multilabel Classification: Problem Analysis, Metrics and Techniques*. Springer International Publishing.

Kleinginna, P. R. and Kleinginna, A. M. (1981). A Categorized List of Emotion Definitions, with Suggestions for a Consensual Definition. *Motivation and emotion*, 5(4):345–379.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Li, X., Peng, Q., Sun, Z., Chai, L., and Wang, Y. (2017). Predicting Social Emotions from Readers' Perspective. *IEEE Transactions on Affective Computing*, in press.

Lin, K. H.-Y., Yang, C., and Chen, H.-H. (2007). What emotions do news articles trigger in their readers? In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 733–734. ACM.

Lin, K. H.-Y., Yang, C., and Chen, H.-H. (2008). Emotion classification of online news articles from the reader's perspective. In *International Conference on Web Intelligence and Intelligent Agent Technology, 2008.* , volume 1, pages 220–226. IEEE.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 312–320.

Ortony, A. and Turner, T. J. (1990). What's basic about basic emotions? *Psychological review*, 97(3):315.

Rao, Y. (2016). Contextual sentiment topic model for adaptive social emotion classification. *IEEE Intelligent Systems*, 31(1):41–47.

Shaver, P., Schwartz, J., Kirson, D., and O'connor, C. (1987). Emotion Knowledge: Further Exploration of a Prototype Approach. *Journal of personality and social psychology*, 52(6):1061.

Villena Román, J., Lana Serrano, S., Martínez Cámara, E., and González Cristóbal, J. C. (2013). TASS - Workshop on Sentiment Analysis at SEPLN. *Procesamiento del Lenguaje Natural*, (50):37–44.

Aggression-annotated Corpus of Hindi-English Code-mixed Data

Ritesh Kumar, Aishwarya N. Reganti, Akshit Bhatia, Tushar Maheshwari

Dr. Bhim Rao Ambedkar University, Indian Institute of Information Technology-Sricity, Birla Institute of Technology and Science-Pilani, Indian Institute of Information Technology-Sricity

Agra, Sricity, Pilani, Sricity

riteshkrjnu@gmail.com, aishwarya.r14@iiits.in, akshit1811@gmail.com, tushar.m14@iiits.in

Abstract

As the interaction over the web has increased, incidents of aggression and related events like trolling, cyberbullying, flaming, hate speech, etc. too have increased manifold across the globe. While most of these behaviour like bullying or hate speech have predated the Internet, the reach and extent of the Internet has given these an unprecedented power and influence to affect the lives of billions of people. So it is of utmost significance and importance that some preventive measures be taken to provide safeguard to the people using the web such that the web remains a viable medium of communication and connection, in general. In this paper, we discuss the development of an aggression tagset and an annotated corpus of Hindi-English code-mixed data from two of the most popular social networking / social media platforms in India – Twitter and Facebook. The corpus is annotated using a hierarchical tagset of 3 top-level tags and 10 level 2 tags. The final dataset contains approximately 18k tweets and 21k facebook comments and is being released for further research in the field.

Keywords: aggression, hate speech, trolling, Hindi-English, code-mixing, Facebook, Twitter

1. Introduction

In the last few years, we have witnessed a gradual shift from largely static, read-only web to quickly expanding user-generated web. There has been an exponential growth in the availability and use of online platforms where users could put their own content. A major part of these platforms include social media websites, blogs, Q&A forums and several similar platforms. All of these are almost exclusively user-generated websites. As such they change and expand at a very rapid pace. In addition to this, most of the traditionally read-only web have started to give an option to the readers to interact with the website as well as the other users by posting their comments and replying to the comments of other users.

In all of these platforms and forums, humongous amount of data is created and circulated every minute. It has been estimated that there has been an increase of approximately 25% in the number of tweets per minutes and 22% increase in the number of Facebook posts per minute in the last 3 years. It is posited that approximately 500 million tweets are sent per day, 4.3 billion Facebook messages are posted each day, more than 200 million emails are sent each day, and approximately 2 million new blog posts are created daily over the web (Schultz, 2016). We still do not have a consolidated figure on the number of comments and opinion generated on different websites but it can be safely assumed that those would be comparably staggering.

As the number of people and this interaction over the web has increased, incidents of aggression and related events like trolling, cyberbullying, flaming, hate speech, etc. too have increased manifold across the globe. While most of these like bullying or hate speech have predated the Internet, the reach and extent of Internet has given these incidents an unprecedented power and influence to affect the lives of billions of people. It has been reported that these incidents have not only created mental and psychological agony to the users of the web but has in fact forced people to deactivate their accounts and in rare instances also commit suicides. Thus the incidents of aggression and unratified verbal behaviour has not remained just a minor nuisance but has acquired the form of a major criminal activity that affects a large number of people. So it is of utmost significance and importance that some preventive measures be taken to safeguard the interests of the people using the web as well as of the web such that it remains a viable medium of communication and connection, in general.

While the initial response to handling these aggressive behaviour was to manually monitor and moderate the user-generated content, the amount and pace at which new data is being created over the web has rendered these manual methods of moderation and intervention almost completely impractical and ineffective. As such it has become imperative that such behaviours be recognised and dealt with using automatic or semi-automatic means.

However, as much as we want to deal with this automatically, it is not quite that easy to automatically recognise these, especially, using the traditional dictionary look-up or similar methods. The systems need to be more intelligent and nuanced in order to be useful in a large number of cases. Moreover, the system should also be able to recognise incidents of both overt as well as covert aggression. At the same time, it must be able to distinguish in between the ratified and unratified aggressive behavior.

2. Verbal Aggression

Verbal aggression could be understood as any kind of linguistic behaviour which intends to damage the social identity of the target person and lower their status and prestige (Barron and Richardson 1994, cited in Culpeper 2011). It is any kind of behaviour that upsets the social equilibrium. In general, verbal aggression can be ratified as well as unratified and it is but obvious that we are mainly concerned with the unratified aggressive behaviour. In order to build an automatic aggression-

detection system, it is important that we have a good understanding of the structure of the phenomenon of aggression itself, especially the distinction between ratified and unratified behaviour such that the system flags only the most serious cases of aggression. Previous research in the field has been carried out to automatically recognise several related behaviours such as trolling (Cambria, et al., 2014; Kumar, Spezzano and Subrahmanian, 2013; Mojica, 2016; Mihaylov et al, 2015), cyberbullying (Kathick et al., 2012; Nitta et al., 2013; Dadvar, Trieschnigg and de Jong, 2014; Van Hee et al, 2015), flaming / insults (Sax, 2016; Nitin et al., 2012), abusive / offensive language (Chen et al, 2012; Nobata et al., 2016) and others. However, there is hardly any theoretical insight into the structure and formation of such behaviours (some notable exceptions include Hardaker, 2010, 2013), in general, and absolutely nothing in Indian scenario. Furthermore, there is a theoretical gap in the understanding of interrelationship among these phenomena – while most of the researchers have focussed on one of these phenomena and their computational processing, it seems there is a significant overlap among these phenomenon in the way they are understood in these studies. All of these are considered undesirable, aggressive and detrimental for those on the receiving end. However, besides focussing on the intention of the initiator of these behaviours, there is hardly any discussion around their pragmatic and/or syntactic structure. So, trolling is intended "to cause disruption and/or to trigger or exacerbate conflict for the purposes of their own amusement" (Hardaker, 2010). Cyberbullying is "humiliating and slandering behavior towards other people" [Nitta et al, 2013]. Flaming intends "to offend someone through e-mail, posting, commenting or any statement using insults, swearing and hostile, intense language, trolling, etc." (Krol, 1992). Going purely by these understandings, the overlap among these phenomena are pretty obvious – as we try to classify actual data in one of these categories, the overlap becomes even more prominent. As such it might be possible to tackle all of these using similar methods, rather than building separate systems for each of these.

3. Data Collection

The data for the current corpus was crawled from the public Facebook Pages and Twitter. The data was mainly collected from the pages/issues that is expected to be discussed more among the Indians (and in Hindi).

For Facebook, more than 40 pages were recognised and crawled to collect the data. It included the pages of following kinds:

- News websites / organizations like NDTV, ABP News, Zee News, etc.
- Web-based forums / portals like Firstost, The Logical Indian, etc.
- Political Parties / groups like INC, BJP, etc.
- Students' organisations / groups like SFI, JNUSU, AISA, etc.
- Support and opposition groups built around recent incidents in Indian Universities of higher education like Rohith Vemula's suicide in HCU, February 9 incident in JNU, etc.

For Twitter, the data was collected using some of the popular hashtags around such contentious themes as beef ban, India vs. Pakistan cricket match, election results, opinions on movies, etc.

During collection, the data was not sampled on the basis of language and so it included data from English, Hindi as well as some other Indian languages. At a later stage, the data from languages other than Hindi and Hindi-English code-mixed were handled.

4. Aggression Typology

Verbal aggression could be divided into two basic types based on how it is expressed -

- Overt Aggression
- Covert Aggression

Furthermore, it can be divided into 4 different types based on the target of aggression -

- Physical Threat
- Sexual Threat / Aggression
- Identity Threat / Aggression
  - Gendered Aggression
  - Geographical Aggression
  - Political Aggression
  - Casteist Aggression
  - Communal Aggression
  - Racial Aggression
- Non-threatening Aggression

In addition to these subtypes of aggression, there two distinctions are also drawn in between aggression and abuse and aggression analysis and sentiment analysis. All of the sub categories and these two distinctions are discussed in detail in the following subsections. We also discuss the annotation scheme developed on the basis of this typology of aggression.

4.1 Overt Aggression

Any speech / text (henceforth, text will mean both speech as well as text) in which aggression is overtly expressed – either through the use of specific kind of lexical items or lexical features which is considered aggressive and / or certain syntactic structures is overt aggression. An example is given below -

अबे कन्हइया सुना इंसहल्लाह (लाल चड्ढी) गैंग दे्ल्ही mcd में 31 सीट पे चुनाव लड़ा ओर इन्हें कुल 51 वोटों से आजादी मिली है। कमाल हो गया बे।

Oye Kanhaiya, I have heard that Insaallah (red chaddi) gang contested election on 31 seats in Delhi MCD and they got azadi (freedom) by a total of 51 votes. It is amazing.

4.2 Covert Aggression

Any text in which aggression is not overtly expressed is covert aggression. It is an indirect attack against the victim and is often packaged as (insincere) polite expressions (through the use of conventionalised polite structures), In general, lot of cases of satire, rhetorical questions, etc. may be classified as covert aggression. An example is given below -

Harish Om kya anti-national ko bail mil sakti hai? ? ?

Harish Om can an anti-national get bail?

4.3 Physical Threat

Any aggressive text that threatens to hurt the victim (an individual or a community) physically or even kill her/him can be classified as physical threat. It also includes suicide intentions, mass killings, etc. as well as potentially physically aggressive (verbal aggression transforming into physical aggression). It is potentially physically aggressive in the sense that verbal aggression might transform into actual physical aggression and as such it is essential that physical threats be recognised accurately. e.g. -

Muh kala hai dogle ka dil bhi kala gaddar hai mujhe tum dikh jaye sala juta marunga dogala deshdrohi

This hypocrite has lost his face, his heart is also bad, as soon as I shall see you moron, I will hit you with a shoe, you hypocritical anti-national

4.4 Sexual Threat / Aggression

Verbal aggression that includes graphic depiction of the actual act of sex or a threat to actually carry out these acts against the victim. e.g.

Bhosri ke jab kuchh pata nahi hai to bolta kyu hai ja ke apni gaar marwa halala me.

You fucker, when you do not know anything then why do speak. Go and get yourself fucked in Halala.

4.5 Identity Threat / Aggression

It refers to the threats to one or more of the identities of the victim. It includes aggression directed at social groups, communities, etc that the victim belongs to. It can again be of 6 kinds depending on which aspect of the identity of the victim is being attacked.

4.5.1 Gendered Aggression

Any text that attacks the victim because of / by referring to her/his gender. It includes homophobic and transgender attacks. It also includes attack against the victim owing to not fulfilling gender roles assigned to them or fulfilling the roles assigned to another gender. e.g.

Napushank tha Nehru... lesbo thi indira

Nehru was impotent, Indira was a lesbian

4.5.2 Geographical Aggression

Aggression aimed at the victim referring to one's place of birth / origin / living is geographical aggression. 'Geographical' in this case could imply a small area like a locality to the whole of the Earth and everything in between which one's identity is attached with. e.g.

Kahe ganda jhuth bolte ho, Sharminda h ki tum bihar se ho

Why do you speak dirty lies. I am ashamed that you are from Bihar.

4.5.3 Political Aggression

Aggression directed against the victim for her/his presumed / actual affinity / membership to a particular political group / community. It also includes aggression against the political group / community itself. e.g

bjp wale jyada dhindhora pitate h hindutva ka...aur hindu me hi equality nhi de pa rhe.isliye bjp ka virodh.baki states k compare me bjp ruled state me ye jyada hota h.isliye v.

The BJP people brag about Hindutva more than others and they are not able to ensure equality among Hindus. That is why this opposition against BJP. And also because this happens more in the BJP-ruled states.

4.5.4 Casteist Aggression

Aggression aimed at the caste of the victim. e.g.

Central govt k cabinet me dekho top k 10 ya 20 ministers ko.1-2 ko chor yahi 15% wale h.

Look in the cabinet of central government. Among the top 10 – 20 ministers, besides 1 – 2, they are only these people with 15% reservation.

4.5.5 Communal Aggression

Verbal aggression towards the religious affiliation and beliefs of the victim. e.g.

Ye bhi sala gandhi khan nehru khan rahul khan jaise kisi muslim ki hi olad h

This moron is also son of some Muslim like Gandhi Khan, Nehru Khan, Rahul Khan

4.5.6 Racial Aggression

Verbal aggression aimed towards the skin color as well as ethnic identity / origin of the victim. e.g.

Aur vishal tu to indian hai.lekin tu kab se hakka noodle chinki ban gaya.indian hone ka proud hai mujhe.tu kyon Japanese ban raha hai.saale chicken noodles.

And Vishal you are Indian. Since when have you become Hakka noodle chinki (Chinese). I am proud of being Indian. Why are you becoming Japanse. Moron chicken noodles.

4.6 Non-threatening Aggression

Aggression against individual traits and choices like color of the house, choice of food (non-communal), etc. are non-threatening (even though it might still be highly distressing for the victims). It also includes most instances of personal insults, cyberbullying, etc. e.g.

तुम सिर्फ पटर पटर बोलना जानते हो और कुछ नहीं।कुछ अच्छे कर्म भी कर लिया करो गरीब दिल से दुआ देंगे सुकून मिलेगा तुझे।

You only know how to blabber and nothing else. You do some good work also, people will bless you from their hear, you will get peace.

4.7 Aggression vs. Abuse

Abuses and aggression are often correlated but neither entails the other. In cases of certain pragmatic practices like 'banter' and 'jocular mockery', abusive constructions are used for establishing inter-personal relationships and increasing solidarity. So these instances cannot be labelled as aggressive. Moreover, most of the examples that I have given above are aggressive but do not contain abuse.

However, both do co-occur in a lot of cases and lot of times we are probably more concerned with (actual) abuses (and not the banter / teasing) than aggression itself. As such, we may consider abuse/curse as one aspect of aggression (even though not strictly a sub-type of aggression). However a more in-depth analysis is needed to discover the relationship between the two.

4.8 Aggression Analysis vs. Sentiment Analysis

At the theoretical level, sentiment analysis seeks to analyse the psychological state of the humans through the language usage while aggression analysis only seeks to analyse the language usage without getting into the question of intentionality. On a more practical level, aggression analysis may be informed by sentiment analysis to certain extent (such that a negative sentiment may strengthen the prediction of being aggressive), it cuts across the sentiment level such that even positive sentiment could be expressed aggressively and negative sentiment need not be aggressive at all. And thus the techniques for the two may overlap at certain places but largely they will depict different sets of features and characteristics.

4.9 Annotation Scheme

Based on the typology discussed above, we have come up with an annotation scheme for annotating the corpus with information related to its aggression level as well as the kind(s) of aggression it exhibits. The tagset contains 3 tags at the top-level and each of the the two aggressive levels contains 2 attributes – discursive role and discursive effects (See Table 1, 2 and 3 below). Discursive effects are based on aggression typology and are of 10 kinds (with all the sub-types of aggression and abuse merged into this level). Discursive roles define the 3 roles that a person might play in an aggressive discourse and they are discussed below.

Aggression Level	**TAG**	**Discursive Features**	
		Discursive Role	**Discursive Effect**
Overtly Aggressive	OAG	Yes	Yes
Covertly Aggressive	CAG	Yes	Yes
Non Aggressive	NAG	May be	No

Table 1 : Aggression Levels

Attribute	**TAG**
Attack	ATK
Defend	DFN
Abet	ABT

Table 2 : Discursive Roles

Discursive Effect	**TAG**
Physical Threat	PTH
Sexual Aggression	SAG
Gendered Aggression	GAG
Racial Aggression	RAG
Communal Aggression	CoAG
Casteist Aggression	CaAG
Political Aggression	PAG
Geographical Aggression	GeAG
General Non-threatening Aggression	NtAG
Curse / abuse	CuAG

Table 3 : Discursive Effects

4.9.1 Discursive Roles

The three kinds of discursive roles define the role of the current post / comment in the ongoing discourse. These are defined as below -

a. **Attack**: Any comment / post which attacks a previous comment / post. It can only be aggressive. An example is given below.

मोदी जी कपड़े भी उतारवा लें तो भी हमें लगेगा कि हिन्दू रक्षा और देश हित में उतारा होगा।

Even if Modiji would rip us in every possible way then also we would feel that it was in the interest of the Nation.

b. **Defend**: Any comment / post which defends or counter-attacks a previous comment / post. The previous comment / post must be an attack and the current one should be in support of the victim. It could be both aggressive as well as non-aggressive. An example of each case is given below

- Kitna dukhi hai bhai tu, lagta hai teri pool kholdi kanhaiya ne. Agar tu jo ilzam uspe laga raha hai wo sach hai toh wo kiyon jail me nahi hai. **(Counter-attack)**

How sad you are bro. It seems that Kanhaiya has shown your true face. If the accusation that you are labelling on him is true then why is he not in the prison.

- Av tak chargesheet file nhi kar payi h Delhi police...Aur lab ne is bat Ko confirm kiya ki anti-national slogan me Kanhaiya ki aawaz nhi h.ye dusre logo no kiya.kon kiya hoga ...Samjhte hi hoge. **(Defend)**

Delhi Police has not been able to file chargesheet till now. And the lab has confirmed that the anti-national slogan does not contain the voice of Kanhaiya. It has been done by other people. You must have an idea who has done it.

c. **Abet**: Any comment / post which lends support and/or encourages a previous aggressive comment / post. The previous comment / post must be an attack and the current one should be in support of the aggressor. It could be both aggressive as well as non-aggressive.

Great sachchai likha aapne

You have written great truth

4.10 Annotation Conventions

Annotation is carried out at the document level – it could be a complete post, a comment or any one unit of the discourse. While annotating, the annotators were given the following instructions (in addition to a detailed annotation guidelines describing the different tags, with examples) -

- Annotators were allowed to mark more than one discursive effects if it seem that certain tweets / comments may be annotated for more than one discursive effect. Thus the annotators were asked to choose ALL the discursive effects that a comment depicts.
- A lot of tweets / comments in the data contained one or the other form of 'abuse' – in those cases the annotators were required to mark the comment as 'abuse' and also at least one more effect must be marked in such cases. So any comment will have a minimum of two effects, if it contains abuse.
- If a tweet / comment is marked as exhibiting General Non-threatening Aggression by the annotators then it cannot be marked for any other effect. In other words, any comment can be marked as General Non-threatening aggression only if it does NOT contain any other kind of threat/aggression. However, General Non-threatening aggression can also contain abuse and if it does, it should be marked so. Thus NtAG comments can be only abuse in addition to itself and nothing else.
- If the tweet / comment was in a language other than English or Hindi (or something that the annotator did not understand), it was to be marked as non-aggressive.

5. Inter-annotator agreement

In order to test the validity and efficiency of the above tagset, we conducted an inter-annotator agreement experiment with 4 annotators using approximately 500 test instances. Kripendorff's Alpha for this experiment was 0.49 (for the top-level annotation). Since the agreement was below par, even going by the standard of pragmatic phenomenon like aggression, we made certain changes to the annotation guidelines and conducted a second round of agreement experiments. Initially, the annotators were allowed to annotate only one discursive effect. It turned out that it resulted in a lot of disagreement among the annotators since a lot of comments / tweets could be interpreted as depicting more than one effect. Moreover, most of the discursive effects were left undefined considering that they were self-explanatory - this left a room for different interpretations by different annotators, resulting in lower inter-annotator agreement. After the first round of experiments, based on the feedback from the annotators, two major changes were done in the guidelines. The annotators were given an option to annotate the comments / tweets with multiple discursive effects. Additionally, all the categories were defined more rigorously, thereby, reducing the scope of different interpretations by the annotators. However, given the fact that aggression is a pragmatic phenomenon, the guidelines still gave annotators the flexibility of giving judgments based on their interpretation, instead of fixing the structures, lexical items, etc for each effect.

The second round of agreement experiments was done over a crowdsourcing platform, Crowdflower, using approximately 1100 test instances. In these experiments, each instance was annotated by 3 annotators. A total of 77 annotators attempted the test. It is to be noted here that each annotator did not annotate equal number of instances. The number of annotations by each annotator ranged from 135 judgments – 10 judgments. At the end of these experiments, the inter-annotator agreement for the top-level was slightly above 72%. While the agreement for the 10-class annotation of discursive effect was approximately 57%. These agreement scores were significantly higher than the scores obtained from the previous scores and so we decided to continue with the data annotation task.

6. Final Dataset

The complete dataset contains approximately 18k tweets and 21k facebook comments annotated with aggression level and discursive effects. The annotation was again done using the Crowdflower platform but it was done by what is known is 'internal' annotators in the Crowdflower lingo. The whole of annotation was done by 4 annotators – all of them were native speakers of Hindi, with a native-like competence in English and were pursuing a doctoral degree in Linguistics.

A preliminary study of the final annotated dataset reveals a fundamental difference in between how people communicate over Facebook and Twitter. Length-wise, approximately ⅔ of the Facebook comments are of less than 150 characters (which is approximately equal to Twitter's restriction of 140 characters). However, a comparison of the aggression level of Twitter and

Facebook (given in Figure 1 and 2) clearly shows that both the platform crucially defines the predominant aggression level – people are more vocal and overtly aggressive on Facebook in comparison to Twitter where people are more subtle and covert in expressing aggression. A different observation, however, could be made about the discursive effect where it seems a majority of the tweets as well facebook comments in the current dataset revolve around the political aggression (Figure 3). Another interesting observation could be made about the interaction between the phenomenon of code-mixing and aggression – the data shows that a majority of code-mixed comments and tweets are aggressive, while for posts in Hindi, it is equally distributed and for posts in English, it is largely non-aggressive (Figure 4).

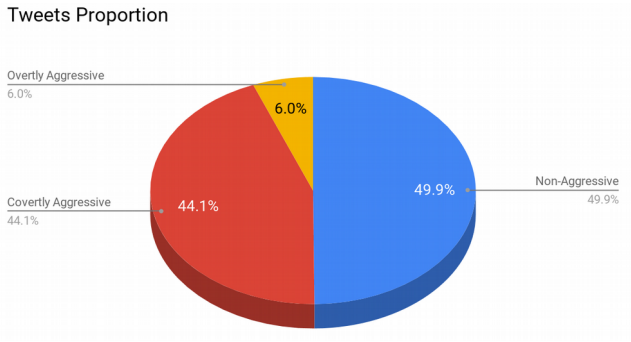
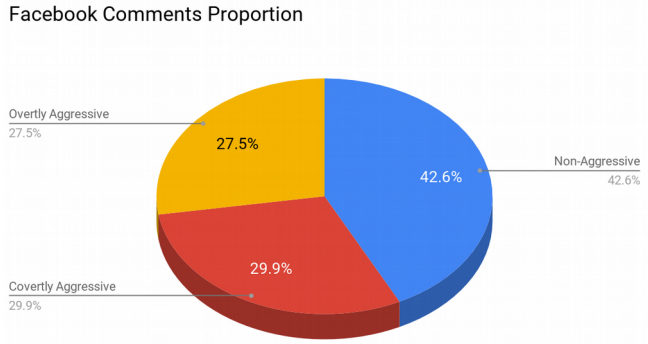


Figure 1 : Proportion of Aggressive and Non-aggressive Tweets in the dataset



Figue 2 : Proportion of Aggressive and Non-aggressive Facebook comments in the dataset

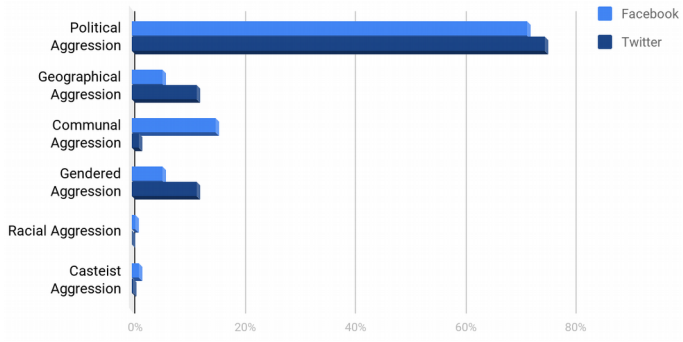


Figure 3 : Comparison of different (Identity) Discursive Effects in the dataset

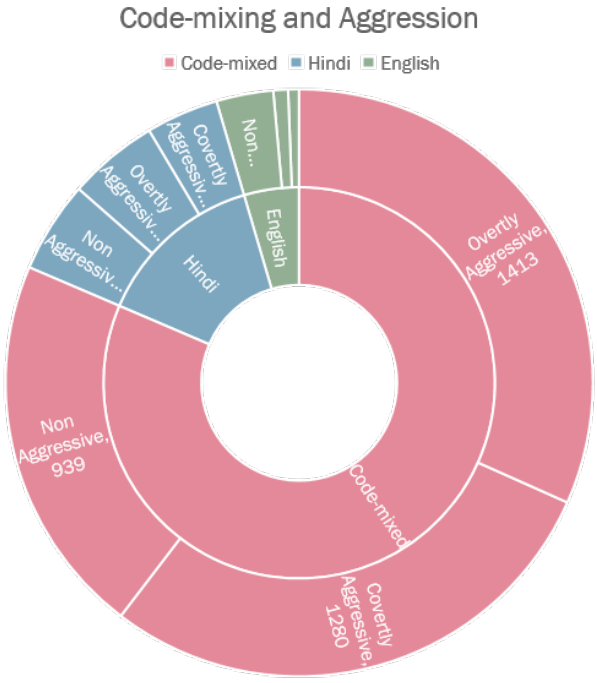


Figure 4 : Interaction of code-mixing and aggression

7. Summing Up

In this paper, we have discussed the development of an aggression-annotated dataset of approximately 18k tweets and 21k Facebook comments in English-Hindi code-mixed language. We have discussed the annotation scheme that was used to annotate the dataset with different levels and types of aggression. As far as we know, it is the first dataset to be annotated with different levels and kinds of aggression. We believe this dataset could be prove to be an invaluable resource for understanding as well as automatically identifying aggression and other related phenomenon like trolling and cyberbullying over the web, especially social media platforms. As such the dataset we will publicly released for free use in further research in the area.

We have recently started experimenting with the automatic identification of aggression using this dataset. However the initial results are not very encouraging with F1 score for the top-level classification - Overtly Aggressive, Covertly Aggressive and Non-aggressive - barely reaching 0.70. It shows the complexity of classifying aggression even at the most basic level and needs further investigation.

Acknowledgements

The annotation of the corpus described in this paper is supported by an Unrestricted Research Grant from Microsoft Research India.

We are very thankful to the annotators - Deepak Alok, Mayank, Atul Ojha and Dhirendra Kumar - for their untiring efforts in finishing the annotation within a very short span of time!

8. Bibliographical References

Cambria, Erik, Praphul Chandra, Avinash Sharma, and Amir Hussain. (2010). Do not feel the trolls. *ISWC, Shanghai.*

Chen, Ying, Yilu Zhou, Sencun Zhu, and Heng Xu. (2012). Detecting offensive language in social media to protect adolescent online safety. Privacy, Security, Risk and Trust (PASSAT), *International Conference on Social Computing (SocialCom)*. pp. 71–80

Cynthia Van Hee, Els Lefever, Ben Verhoeven, Julie Mennes, Bart Desmet, Guy De Pauw, Walter Daelemans, and Véronique Hoste. (2015). Detection and fine-grained classification of cyberbullying events. In *Proceedings of International Conference Recent Advances in Natural Language Processing (RANLP)*, pp. 672–680.

Culpeper, Jonathan. (2011). *Impoliteness Using Language to cause Offense.* Cambridge: Cambridge University Press.

Dadvar, Maral, Dolf Trieschnigg, and Franciska de Jong. (2014). Experts and machines against bullies: a hybrid approach to detect cyberbullies. In *Advances in Artificial Intelligence*, pp. 275–281. Berlin: Springer

Hardaker, Claire. (2010). Trolling in Asynchronous Computer-Mediated Communication: From User Discussions to Academic Definitions. *Journal of Politeness Research. Language, Behaviour, Culture*, 6 (2): 215–242.

Hardaker, Claire. (2013). "Uh. . . . not to be nitpicky,,,,,but...the past tense of drag is dragged, not drug." An overview of trolling strategies. *Journal of Language Aggression and Conflict*, 1(1): 58–86.

Karthik Dinakar, Birago Jones, Catherine Havasi, Henry Lieberman, and Rosalind Picard. (2012). Common sense reasoning for detection, prevention, and mitigation of cyberbullying. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(3):18.

Krol, E. (1992). *The whole internet: User's guide & catalog.* Sebastopol, CA: O'Reilly & Associates, Inc.

Kumar, Sudhakar, Francesca Spezzano, and VS Subrahmanian. (2014). Accurately detecting trolls in slashdot zoo via decluttering. In *Proceedings of IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. pp. 188–195.

Mojica, Luis G. (2016). Modeling Trolling in Social Media Conversations. *arXiv:1612.05310 [cs.CL]*. https://arxiv.org/pdf/1612.05310.pdf [Retrieved on Feb 16, 2017]

Mihaylov, Todor, Georgi D Georgiev, AD Ontotext, and Preslav Nakov. (2015). Finding opinion manipulation trolls in news community forums. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning, CoNLL*, pp. 310–314.

Nitin, Ankush Bansal, Siddhartha Mahadev Sharma, Kapil Kumar, Anuj Aggarwal, Sheenu Goyal, Kanika Choudhary, Kunal Chawla, Kunal Jain, Manav Bhasinar. (2012). Classification of Flames in Computer Mediated Communications. *arXiv:1202.0617 [cs.SI]*. https://arxiv.org/pdf/1202.0617.pdf [Retrieved on Feb 16, 2017].

Nitta, Taisei, Fumito Masui, Michal Ptaszynski, Yasutomo Kimura, Rafal Rzepka, and Kenji Araki. (2013). Detecting cyberbullying entries on informal school websites based on category relevance maximization. In *IJCNLP*, pp. 579–586.

Nobata, Chikashi, Joel Tetreault, Achint Thomas, Yashar Mehdad and Yi Chang. (2016). Abusive Language Detection in Online User Content. In *Proceedings of WWW2016*. pp. 145 – 153.

Sax, Sasha. (2016). Flame Wars: Automatic Insult Detection. *Stanford Reports*. http://cs224d.stanford.edu/reports/Sax.pdf [Retrieved on Feb 20, 2017]

Schultz, Jeff. (2016). *How Much Data is Created on the Internet Each Day?* https://www.gwava.com/blog/internet-data-created-daily . Retrieved on February 28, 2017.

Creating a Verb Synonym Lexicon Based on a Parallel Corpus

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, Jan Hajič
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Malostranské nám. 25
11800 Prague 1, Czech Republic
{uresova,fucikova,hajicova,hajic}@ufal.mff.cuni.cz

Abstract

This paper presents the first findings of our recently started project of building a new lexical resource called CzEngClass, which consists of bilingual verbal synonym groups. In order to create such a resource, we explore semantic 'equivalence' of verb senses (across different verb lexemes) in a bilingual (Czech-English) setting by using translational context of real-world texts in a parallel, richly annotated dependency corpus. When grouping semantically equivalent verb senses into classes of synonyms, we focus on valency (arguments as deep dependents with morphosyntactic features relevant for surface dependencies) and its mapping to a set of semantic "roles" for verb arguments, common within one class. We argue that the existence of core argument mappings and certain adjunct mappings to a common set of semantic roles is a suitable criterion for a reasonable verb synonymy definition, possibly accompanied with additional contextual restrictions. By mid-2018, the first version of the lexicon called CzEngClass will be publicly available.

Keywords: Lexical Resource, Parallel Corpus, Semantics, Syntax, Synonymy, Valency

1. Introduction

While synonym lexicons, such as the Roget's thesaurus (Associates, 1988),[1] or WordNet (Miller, 1995; Fellbaum, 1998) are well-known resources, used in both research and NLP applications, we believe that there is a significant gap in these resources (which concerns mainly verbs). We see two orthogonal problems: first, a granularity problem– current synonym classes (synsets, entries, frames, ...) as found in these resources are usually too broad (contain words which can be considered synonyms only in very specific contexts), or they use too fine-grained sense distinctions, such as in WordNet, which often distinguish too many and too close verb senses (Palmer et al., 2007); second, a coverage problem - simply there are not enough verbs covered or some verb senses are missing.[2]

In our project, we attempt to introduce criteria, based mainly on valency or predicate-argument structure (taking into account both semantic and morphosyntactic properties of verbs) to help define both the granularity of verb senses as well as the synonym classes themselves. Along with (Levin, 2013), we refer to predicate-argument structure as to those structures that involve the realization of lexical item's arguments, including morphosyntactic properties that affect the morphosyntactic realization of arguments. This is similar to our valency approach, the FGD Valency Theory (FGDVT), first described in (Panevová, 1974). As (Levin, 2014) notes, the meaning of a verb may be characterized via the relations that its arguments bear to it and "Semantic Roles" (SRs) can be seen as labels for certain recurring predicate-argument relations.

In addition, we have chosen to use multilingual, rather than monolingual-only evidence to help with both the sense distinctions and the synonymy relation proper. We have started with Czech and English, since there are resources (both lexical and textual) that allow us to do so.

The goal of the project is thus to group verbs used as synonyms in Czech and English into (cross-lingual) synonym classes representing a cross-lingual meaning of the state or event expressed by the set of verbs assigned to that class. We call the resulting lexicon CzEngClass.

For the purpose of this work, we use the term "synonym" in the "loose" interpretation (Lyons, 1968), i.e., the necessary semantic equivalence takes also wider context into account. We proceed strictly "bottom-up", i.e. from the corpus and existing lexical resources towards the new lexicon. The novel feature is the use of a richly annotated bilingual corpus (Sect. 2.) to achieve deeper insight into the usage of verbs (together with their arguments) in translation.

The results reported in our paper are based on a sample of 60 classes manually processed and entered into the CzEngClass lexicon. We also describe their linkage to additional existing resources, the relevant features of which are described, too (Sect. 3.). The tool used for creating this sample is described elsewhere in this volume (Urešová et al., 2018).

Our approach to synonymy, meaning and sense is captured in Section 4.. Section 5. deals with the design of CzEngclass lexicon. In Section 6., we exemplify the main (albeit first) findings. We comment on the CzEngClass' criteria for grouping synonymous, based on samples that have been annotated and processed while creating the first entries. Section 7. summarizes our approach and points to future work.

2. Corpus Resources

For our work we use primarily the parallel Prague Czech-English Dependency Treebank 2.0 (PCEDT 2.0) (Hajič et al., 2012). This corpus contains the Wall Street Journal (WSJ) part of the Penn Treebank (Marcus et al., 1993) and its manual (human) Czech translations. Each language part is enhanced with a rich manual linguistic an-

[1] http://www.roget.org

[2] It is left for future investigation whether other parts of speech, especially nouns, would be subject to similar problems, but now we concentrate on verbs since they are generally considered the core of every sentence (clause). We will extend our approach to nouns and adjectives later.

notation in The Prague Dependency Treebank (PDT 2.0) style (Hajič et al., 2006; Hajič et al., 2018), which is in turn based on the Functional Generative Dependency (FGD) framework (Sgall et al., 1986). The PDT annotation uses a "stratificational" (layered) approach containing multiple layers (morphology, surface syntax and deep syntax). The main annotated phenomena are (surface) dependency structure and (deep) syntactico-semantic labeling of predicate-argument structure. For the purpose of our study, it is important that the annotation contains interlinked surface dependency trees and deep syntactico-semantic (tectogrammatical) trees, both vertically and also horizontally across the two languages on sentence and node levels. At the deep layer, each verb node (occurrence) is additionally linked to the corresponding valency frame in the associated valency lexicons, PDT-Vallex and EngVallex (Sect. 3.), effectively providing also word sense labeling for all verb occurrences in the bilingual corpus.

3. Lexical Resources

PDT-Vallex (Urešová et al., 2014), (Urešová, 2011) is a Czech valency lexicon, manually created in a bottom-up way during the annotation of the PDT/PCEDT 2.0. Each entry in the lexicon contains a headword (lemma) with associated valency frames. A valency frame typically corresponds to one sense of the verb, even though for close verb senses[3] and identical valency frames only one valency frame may exist in the lexicon. Each valency frame includes labeled valency frame members, or valency "slots" (i.e., ACT for Actor, PAT for Patient, ADDR for Addressee, etc.), semantic "obligatoriness" attribute, and subcategorization information, i.e., required surface form(s) of the individual valency frame members. Most valency frames include a note or an example explaining their meaning and usage. The version of PDT-Vallex used here contains 11,933 valency frames for 7,121 verbs.

EngVallex (Cinková et al., 2014), (Cinková, 2006) is a valency lexicon of English verbs created on the same principles as PDT-Vallex by an automatic conversion from PropBank frame files (Palmer et al., 2005) which was manually refined afterwards.[4] EngVallex was used for the annotation of the English part of the PCEDT 2.0. Currently, it contains 7,148 valency frames for 4,337 verbs. For the most part, EngVallex does not contain explicitly formalized subcategorization information.

CzEngVallex (Urešová et al., 2015), (Urešová et al., 2016) is based on the treebank annotation of the PCEDT 2.0 corpus, covering about 86,000 aligned verbal pairs. It is a manually annotated Czech-English valency lexicon linking verbal entries of PDT-Vallex and EngVallex. Over 66% of English verbs and 72% of Czech verbs[5] in the PCEDT 2.0 have a verbal translation covered by the CzEngVallex mapping. CzEngVallex builds links not only between corresponding verbal frames but also between corresponding verb arguments for each pair of verb senses, providing an interlinked database of argument structures available for each verb and documenting a cross-lingual comparison of Czech and English valency behavior.

VALLEX[6] (Lopatková et al., 2016) is closely related to PDT-Vallex because it is built on the same theoretical framework. This lexicon is much more elaborated, however it is not based on the PDT data.

Among other resources we use, there are **FrameNet** (Baker et al., 1998; Fillmore et al., 2003), **FrameNet+** (Pavlick et al., 2015), **VerbNet** (Schuler, 2006), **SemLink** (Palmer, 2009; Bonial et al., 2012), **PropBank** (Palmer et al., 2005) and **English WordNet**[7] (Miller, 1995; Fellbaum, 1998) as well as **Czech WordNet** (Pala et al., 2011), (Pala and Smrž, 2004). These resources have been used for an initial set of semantic roles (taken mostly from FrameNet and VerbNet),[8] and their entries will be referred to explicitly from all the corresponding entries in the CzEngClass lexicon, if possible to the exact lexical units/frames/sense groups/synsets.

4. Synonymy

Before we address synonymy, we make a short digression to the term "meaning" and term "sense" as we interpret them in our work, due to sometimes wildly different understandings of these terms.

We understand the term "sense" in the same way as e.g., (Hofmann, 1993) saying "when a form has several different concepts associated with it, we sometimes call them different senses or readings of the word." (Wierzbicka, 1996) has the same approach, explaining the term "sense" on four different senses of the word *spring*. We also differentiate a single verb (type, lemma, possibly multiword) into one or more "senses," represented by its valency frame; in our lexical valency resources (PDT-Vallex and EngVallex) the individual senses (i.e., the valency frames) of a verb are technically represented by a unique ID. The term (lexical) "meaning" is understood here with regard to a context, i.e., syntactic and semantic surroundings of the word; similarly to Wittgenstein's understanding that "the meaning of an expression is a function of its use in a particular context" (Frawley, 1992). In our use of the terms "meaning" and "sense", the following holds:

- "sense" is used only to distinguish different meanings within one verb type (lemma), e.g., the verb *leave* has (at least) two senses, *leave sth somewhere* (leave book on the table) and *leave someplace [for some other place]* (leave Paris [for London]),
- "meaning" is not applied to verbs (lemmas, verb types), but only to their distinguished senses (lexical units), and can be compared across such units: e.g., *leave* in the sense *leave someplace [for some other place]* has similar meaning to *depart* in its sense *depart from somewhere*,
- consequently, two senses of the same verb can never be totally equivalent, i.e., they never have the same

[3] For explanation of the term "sense", as it is used in this paper, please see Sec. 4..

[4] EngVallex preserves most of the links to PropBank.

[5] The remaining pairings are verb-noun or verbs translated as structurally different constructions.

[6] http://ufal.mff.cuni.cz/vallex/3.0/

[7] https://wordnet.princeton.edu/

[8] The current theories of SRs are reviewed in more detail in (Levin and Hovav, 2005).

meaning (they would not be separated if they were equivalent),

- meanings can thus only be (non-trivially) compared *across* verbs (more precisely, across lexical units defined for ("within") different verbs).

In the working definition for establishing the CzEngClass entries, we use **"contextually-based"** synonymy. For two verbs (verb senses) to be considered contextually synonymous, and therefore be members of the same class, they have to convey the same or similar meaning, both monolingually and cross-lingually, and they must share the same Semantic Roles (SRs), albeit they can be expressed by different morphosyntactic realizations as well as subject to additional restrictions.

5. Structure of CzEngClass Lexicon

For our study, the following lexicon structure (Fig. 1) has been designed.

The CzEngClass lexicon builds upon the existing resources, as described above: CzEngVallex, PDT-Vallex and EngVallex and the PCEDT parallel corpus. In addition, the other lexicons listed (VALLEX, FrameNet, VerbNet, PropBank and WordNet(s)) are used as additional sources, and links will be kept between their entries and the CzEngClass entries.

At the core of the CzEngClass lexicon, there are Synonym Classes, which are, for the purpose of this project, defined as (multilingual, or rather cross-lingual)[9] groups of verb senses (of different words) that have the same meaning *and* the arguments of which can be mapped to a common set of SRs (cf. the purple boxes Agent, Item, Change in Fig. 1).

SRs have been reviewed (with FrameNet's core roles for the linked-to entry(ies) providing inspiration) and determined for all the members of the group, and mapped to arguments from EngVallex (and PDT-Vallex for Czech verbs), using also the pairings from the CzEngVallex lexicon, which is in turn linked to the PCEDT parallel Czech-English corpus (lower part of Fig. 1).

6. CzEngClass Examples

We capture the common background information of one synonym class realized through a set of common SRs in one "frame," called SynSemFrame (synonym semantic frame). We present two examples here.

In the relatively straightforward synonym verb class in Table 1, exemplified in the three sentences below, we consider all verbs to be synonyms and group them in the SynSemFrame COMPLAIN, since the valency pattern of the source verbs *complain, gripe, grumble* correlates (for almost all arguments) 1:1 with the valency pattern of the translational equivalent, i.e., verb *stěžovat si* (lit. *complain*). The exception is the Czech verb itself, where both ADDR and LOC can be mapped to the role `Addressee`. The reason is, however, simply the conventions applied in the FGDVT to cases where the surface expression is location rather than (= in place of) a (true) addressee (as an animate agent); typically, this happens for offices (*to the clerk*.ADDR vs. *at the court office*.LOC), government seats (*to the governor*.ADDR vs. *in Annapolis*.LOC), etc. Consequently, in all these cases, functors from the valency lexicons and the deep dependencies can easily be mapped to a common set of SRs, taken from FrameNet in this case.

Examples (Czech translations double as examples for the Czech verb *stěžovat si* in the class):

En: *Mrs. Yeargin*.`Complainer`/ACT *never complained to school officials*.`Addressee`/ADDR [*that the standardized test was unfair.*].`Complaint`/PAT

Cz: *Yearginová*.`Complainer`/ACT *si nikdy na školském úřadu*.`Location`/LOC *nestěžovala na nespravedlnost*.`Complaint`/PAT *standardizovaných testů.*

En: [*"For $10 million, you can move $100 million of stocks,"*].`Complaint`/PAT *a specialist*.`Complainer`/ACT *... gripes.* NULL.`Addressee`/ADDR

Cz: [*"Za 10 milionů dolarů můžete přesunout akcie za 100 milionů dolarů,"*].`Complaint`/PAT *stěžuje si ... odborník..*`Complainer`/ACT NULL.`Addressee`/ADDR

En: *Soviet consumers*.`Complainer`/ACT *grumble at the exorbitant black-market prices*.`Complaint`/PAT NULL.`Addressee`/ADDR

Cz: *Sovětští spotřebitelé*.`Complainer`/ACT *si stěžují na nehorázné ceny*.`Complaint`/PAT *za takové zboží na černém trhu.* NULL.`Addressee`/ADDR

	Roles		
	`Complainer`	`Addressee`	`Complaint`
stěžovat si	ACT	ADDR,LOC	PAT
complain	ACT	ADDR	PAT
gripe	ACT	ADDR	PAT
grumble	ACT	ADDR	PAT

Table 1: Mappings for COMPLAIN class

Table 2 shows the class OFFER, with additional complex examples. While the verbs *offer*, *bid*, *proffer* and *tender*, as well as the Czech verb *nabídnout* do not pose any problem in mapping their valency arguments to the four SRs associated with this class, the other aligned translations, described below, are more complex.

The verb *extend* (in the sense "extend an offer") seems not to fit the pattern; without the `Entity Offered`/PAT being a noun phrase meaning exactly what the class is about, namely *offers, bids, aid*, etc., its meaning is more similar to "hand over" than to "offer". However, looking up all the possible deep objects to this meaning of *extend*, it is clear that it cannot be complemented by a direct object *not* being an *offer*, *bid* or something similar. If such a noun is further modified, one of its dependents might describe the `Entity Offered` (e.g., *employment*: *employment offer* or *offer of employment*). If it is not present, this construction simply says that there was an *offer* while not specifying anything more specific about it, unless the word *offer* or its equivalent is left out and the `Entity Offered` is specified as a dependent directly on the verb *extend* itself, as in *the banks ... extended [company] up to $90 million in revolving loans*. Therefore, both the PAT as well as the restrictive attribute of the PAT (denoted as PAT(RSTR) can be the `Entity Offered`.

The verbal phrase *make available* is also used as a translation of the Czech verb *nabídnout*. In this case, *make* behaves almost as a light verb, which typically keeps some

[9] For the time being, bilingual: in Czech and English.

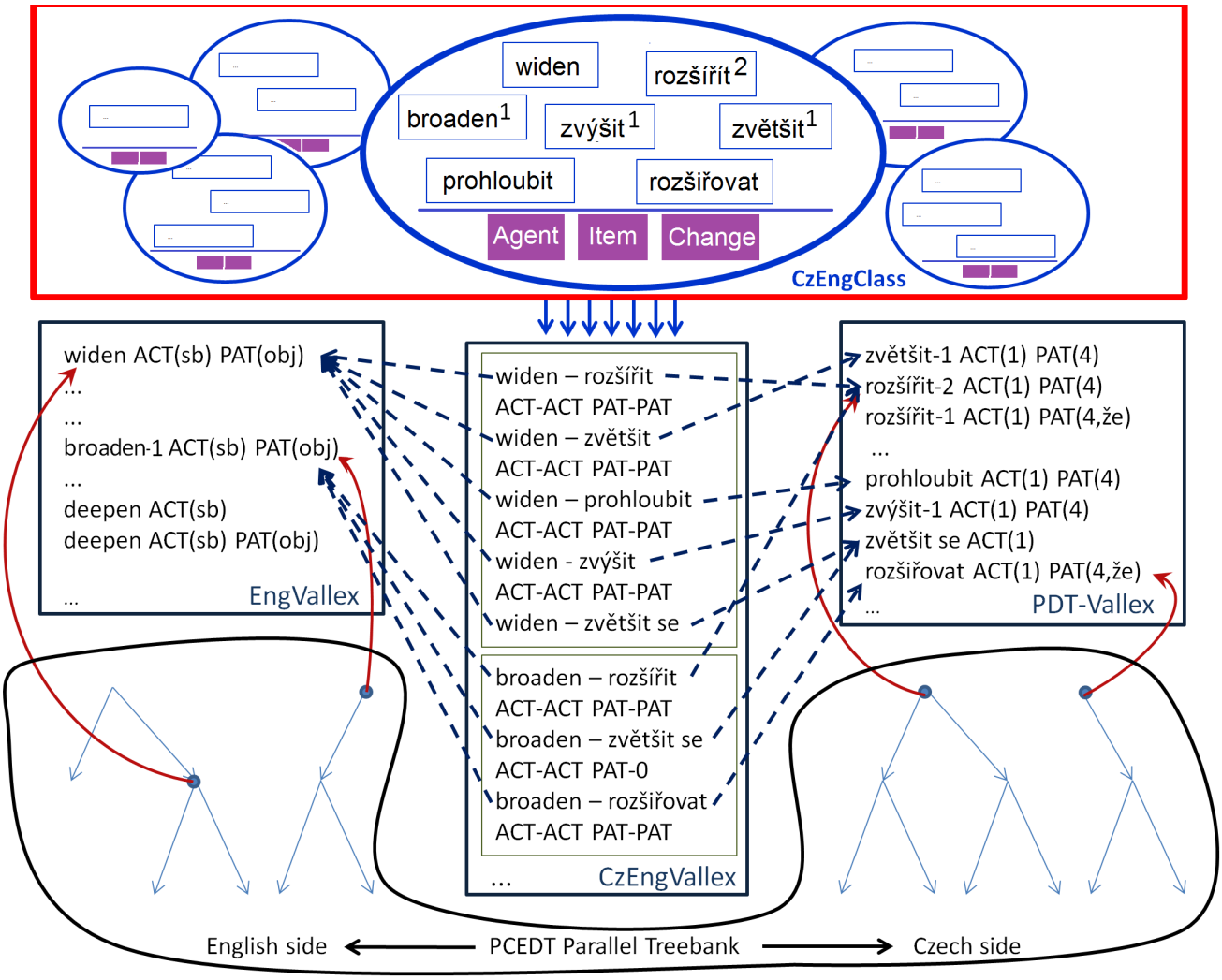


Figure 1: CzEngClass lexicon & other resources

arguments (mostly ACT) and "pushes" some other ones to the nominal part. In this case, the situation is similar, without *make* being considered a light verb (otherwise, a different functor would have been used for *available*). From the example below, it is clear that the translation was quite adequate and thus *make available* should be kept in the synonym class OFFER (see the deep dependency annotation of this example captured in Figure 2) where the `Recipient` is annotated with the relation BEN as a dependent on *available*, denoted as EFF(BEN) in Table 2:

En: *Japanese researchers*.`Offerer`/ACT ... *have made available*.EFF *three possible cures*.`Entity Offered`/PAT *to American researchers*.`Recipient`/BEN ... .

Cz: *Japonští výzkumníci nabídli ... americkým výzkumníkům tři možné léčebné postupy...*

Last translation used for *nabídnout* was *place (a value)*, e.g., in the sentence:

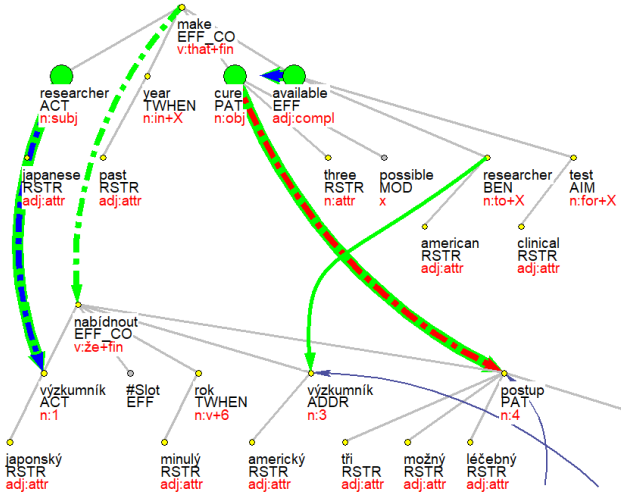
En: *... buyers who place the highest value on them.*

Cz: *...kupujícím, kteří za ně nabídnou nejvyšší hodnotu.*

However, this is considered pure nominalization (other sentences have been found where *offer, bid*, etc. as a head of a noun phrase is aligned with a Czech content verb), even if there was no other equivalent word for *place* in the Czech translation. This is an example where the translated text probably does not have the same effect on the target reader: the translator simply added content to a sentence based on her/his understanding of the context. Therefore, the verb *place* was, at present, not included in the OFFER synonym class. This also demonstrates the difference between *interlingual synonymy* (which defines the CzEngClass lexicon) and *translational equivalence* (not necessarily represented).

	Roles			
	`Offerer`	`Recipient`	`Entity Offered`	`Entity Received`
nabídnout	ACT	ADDR	PAT	EFF
offer	ACT	ADDR	PAT	EFF
bid	ACT	ADDR	PAT	EFF
proffer	ACT	ADDR	PAT	SUBS
tender	ACT	ADDR	PAT	SUBS
extend	ACT	ADDR	PAT, PAT(RSTR)	SUBS
make available	ACT	EFF(BEN)	PAT	SUBS
	Restriction: EFF[EXCHANGE]			

Table 2: Mapping for OFFER class



En: Japanese researchers... have made available three possible cures to American researchers.

Cz: Japonští výzkumníci nabídli ... americkým výzkumníkům tři možné léčebné postupy.

Figure 2: Dependency tree annotations for English sentence rooted in *make available* and its Czech translation.

7. Conclusions and Future Work

The CzEngClass lexicon is a contribution to a set of lexical resources important for many NLP tasks, such as event detection and linking, semantic relation extraction, etc. While these tasks often use unlabeled data for training, there is always a need for human-annotated data for evaluation, tuning, etc. Importantly, we believe that such a resource is currently missing in the offerings of lexical resources. In addition, it contains links to all the relevant related resources, such as VALLEX, FrameNet, VerbNet, PropBank and WordNet, making it suitable for comparative studies.

We have based our study on the assumption that mapping syntactic structure (verb arguments) of verbs to semantic roles (with some added restrictions) helps us discover their meaning affinity through which we can group these verbs into synonym classes. To help avoid single-language bias, we have used a parallel corpus and the alignments of verbs, previously manually checked and extracted. In such a corpus, aligned translations of a single verb should in principle be, in most cases, synonymous. The resulting sets of cross-lingual synonyms including the explicit mappings between the FGDVT valency functors and the semantic roles assigned to each synonym set are captured in the CzEngClass lexicon. So far, the roles used in FrameNet and VerbNet seem to form a suitable set for each synonym class and the mapping. However some adjustments, e.g., merging the roles, adding non-core roles, have to be made.

Once the lexicon reaches reasonable size (approx. 500 classes), we will compare the results with automated synonym discovery methods, such as (van der Plas et al., 2011; van der Plas et al., 2014), either using Deep Learning (looking, e.g., at embeddings based on argument-role mapping) or other previously well-researched methods, such as the LDA which has been already used for Czech, e.g., in (Materna, 2012).

Finally, we plan to publish the resulting CzEngClass lexicon as an open source dataset.

Acknowledgments

This work has been supported by the grant No. GA17-07313S of the Grant Agency of the Czech Republic, and it uses resources hosted by the LINDAT/CLARIN Research Infrastructure, projects No. LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781, supported by the Ministry of Education of the Czech Republic. The last co-author has also been supported in part by the project VIA-DAT, No. DG16P02R019, of the Ministry of Culture of the Czech Republic.

8. Bibliographical References

Associates, P. S. I. . (1988). *Roget's thesaurus of synonyms and antonyms*. P.S.I. & Associates.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bonial, C., Feely, W., Hwang, J. D., and Palmer, M. (2012). Empirically Validating VerbNet Using SemLink. In *Seventh Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, Istanbul, Turkey, May.

Cinková, S. (2006). From PropBank to EngValLex: adapting the PropBank-Lexicon to the valency theory of the functional generative description. In *Proceedings of LREC 2006, Genova, Italy*.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Fillmore, C. J., Johnson, C. R., and L.Petruck, M. R. (2003). Background to FrameNet: FrameNet and Frame Semantics. *International Journal of Lexicography*, 16(3):235–250.

Frawley, W. (1992). *Linguistic Semantics*. Lawrence Erlbaum, Hillsdale.

Hofmann, T. R. (1993). *Realms of Meaning. An Introduction to Semantics*. Longman, Oxford.

Levin, B. and Hovav, M. R. (2005). *Argument realization*. Cambridge Univ. Press, Cambridge.

Levin, B. (2013). Argument Structure. *Oxford Bibliographies in Linguistics*.

Levin, B. (2014). Semantic roles. *Oxford Bibliographies in Linguistics*.

Lopatková, M., Kettnerová, V., Bejček, E., Vernerová, A., and Žabokrtský, Z. (2016). *Valenční slovník českých sloves VALLEX*. Nakladatelství Karolinum, Praha, Czechia.

Lyons, J. (1968). *Introduction to Theoretical Linguistics*. Cambridge University Press.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330.

Materna, J. (2012). LDA-Frames: An Unsupervised Approach to Generating Semantic Frames. In *Proceedings of the 13th International Conference on Computational Linguistics and Intelligent Text Processing - Volume Part I*, CICLing'12, pages 376–387, Berlin, Heidelberg. Springer-Verlag.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41, November.

Pala, K. and Smrž, P. (2004). Building Czech Wordnet. *Romanian Journal of Information Science and Technology*, 7(1-2):79–88.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, March.

Palmer, M., Dang, H., and Fellbaum, C. (2007). Making Fine-grained and Coarse-grained sense distinctions, both manually and automatically. *Journal of Natural Language Engineering*, 13(2):137–163.

Palmer, M. (2009). Semlink: Linking PropBank, VerbNet and FrameNet. In *Proceedings of the Generative Lexicon Conference*, page 9–15.

Panevová, J. (1974). On verbal Frames in Functional Generative Description. *Prague Bulletin of Mathematical Linguistics*, 22:3–40.

Pavlick, E., Wolfe, T., Rastogi, P., Callison-Burch, C., Dredze, M., and Van Durme, B. (2015). Framenet+: Fast paraphrastic tripling of framenet. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 408–413, Beijing, China, July. Association for Computational Linguistics.

Schuler, K. K. (2006). *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania.

Sgall, P., Hajičová, E., and Panevová, J. (1986). *The meaning of the sentence in its semantic and pragmatic aspects*. D. Reidel, Dordrecht.

Urešová, Z., Fučíková, E., and Šindlerová, J. (2016). CzEngVallex: a bilingual Czech-English valency lexicon. *The Prague Bulletin of Mathematical Linguistics*, 105:17–50.

Urešová, Z., Fučíková, E., Hajičová, E., and Hajič, J. (2018). Tools for Building an Interlinked Synonym Lexicon Network. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC'18)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Urešová, Z. (2011). *Valenční slovník Pražského závislostního korpusu (PDT-Vallex)*. Studies in Computational and Theoretical Linguistics. Ústav formální a aplikované lingvistiky, Praha, Czechia.

van der Plas, L., Merlo, P., and Henderson, J. (2011). Scaling up cross-lingual semantic annotation transfer. In *Proceedings of ACL/HLT*, pages 299–304, Portland, US.

van der Plas, L., Apidianaki, M., and Chen, C. (2014). Global methods for cross-lingual semantic role and predicate labelling. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, pages 1279–1290, Dublin, Ireland.

Wierzbicka, A. (1996). *Semantics: primes and universals*. Oxford Univ. Press, Oxford & New York.

9. Language Resource References

Cinková, S., Fučíková, E., Šindlerová, J., and Hajič, J. (2014). *EngVallex - English Valency Lexicon*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0023-4337-2.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Cinková, S., Fučíková, E., Mikulová, M., Pajas, P., Popelka, J., Semecký, J., Šindlerová, J., Štěpánek, J., Toman, J., Urešová, Z., and Žabokrtský, Z. (2012). *Prague Czech-English Dependency Treebank 2.0, https://catalog.ldc.upenn.edu/LDC2004T25*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4.

Hajič, J., Bejček, E., Bémová, A., Buráňová, E., Hajičová, E., Havelka, J., Homola, P., Kárník, J., Kettnerová, V., Klyueva, N., Kolářová, V., Kučová, L., Lopatková, M., Mikulová, M., Mírovský, J., Nedoluzhko, A., Pajas, P., Panevová, J., Poláková, L., Rysová, M., Sgall, P., Spoustová, J., Straňák, P., Synková, P., Ševčíková, M., Štěpánek, J., Urešová, Z., Vidová Hladká, B., Zeman, D., Zikánová, Š., and Žabokrtský, Z. (2018). *Prague Dependency Treebank 3.5*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-2621.

Hajič, J., Panevová, J., Hajičová, E., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M., Žabokrtský, Z., Ševčíková Razímová, M., and Urešová, Z. (2006). *Prague Dependency Treebank 2.0*. LDC, number LDC2006T01, LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4.

Pala, K., Čapek, T., Zajíčková, B., Bartůšková, D., Kulková, K., Hoffmannová, P., Bejček, E., Straňák, P., and Hajič, J. (2011). *Czech WordNet 1.9 PDT*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0001-4880-3.

Urešová, Z., Štěpánek, J., Hajič, J., Panevová, J., and Mikulová, M. (2014). *PDT-Vallex: Czech Valency lexicon linked to treebanks*. LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0023-4338-F.

Urešová, Z., Fučíková, E., Hajič, J., and Šindlerová, J. (2015). *CzEngVallex - Czech English Valency Lexicon*. Charles University in Prague, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-1512.

Evaluation of Domain-specific Word Embeddings using Knowledge Resources

Farhad Nooralahzadeh, Lilja Øvrelid, Jan Tore Lønning
Department of Informatics, University of Oslo
{farhadno,liljao,jtl}@ifi.uio.no

Abstract

In this work we evaluate domain-specific embedding models induced from textual resources in the Oil and Gas domain. We conduct intrinsic and extrinsic evaluations of both general and domain-specific embeddings and we observe that constructing domain-specific word embeddings is worthwhile even with a considerably smaller corpus size. Although the intrinsic evaluation shows low performance in synonymy detection, an in-depth error analysis reveals the ability of these models to discover additional semantic relations such as hyponymy, co-hyponymy and relatedness in the target domain. Extrinsic evaluation of the embedding models is provided by a domain-specific sentence classification task, which we solve using a convolutional neural network. We further adapt embedding enhancement methods to provide vector representations for infrequent and unseen terms. Experiments show that the adapted technique can provide improvements both in intrinsic and extrinsic evaluation.

Keywords: word embeddings, intrinsic and extrinsic evaluation, domain knowledge resource, embeddings enhancement

1. Introduction

Domain-specific, technical vocabulary presents a challenge to NLP applications. Recently, word embedding models have been shown to capture a range of semantic relations relevant to the interpretation of lexical items (Mikolov et al., 2013b) and furthermore provide useful input representations for a range of downstream tasks (Collobert et al., 2011). The majority of work dealing with intrinsic evaluation of word embeddings has focused on general domain embeddings and semantic relations between frequent and generic terms. However, it has been shown that embeddings differ from one domain to another due to lexical and semantic variation (Hamilton et al., 2016; Bollegala et al., 2015). Domain-specific terms are challenging for general domain embeddings since there are few statistical clues in the underlying corpora for these items (Bollegala et al., 2015; Pilehvar and Collier, 2016).

The Oil and Gas domain is a highly technical and data-intensive domain. Experts working within this domain daily investigate selected geographical areas and use relevant information (scientific articles, reports and other textual sources) to evaluate the potential for undiscovered hydrocarbons. The vocabulary is technical and there is a real need for NLP tools to aid the work process. In this work we investigate whether word embedding models can capture domain-specific semantic relations by training domain-specific embeddings[1] and evaluating these against a terminological resource. We conduct a comprehensive study including a wide range of evaluation criteria, contrasting several general and domain specific embedding models. We augment the domain-specific embeddings using a domain knowledge resource. To supply embeddings for rare words, we extend the retrofitting method by Faruqui et al. (2015). We then go on to examine the contribution of these models in the performance of a downstream classification task.

[1]Link to the domain-specific model: `http://vectors.nlpl.eu/repository/11/75.zip`

2. Related work

Despite the pervasive use of word embedding in language technology, there is no agreement in the community on the best ways to evaluate these semantic representations of language[2]. There exist a variety of benchmarks which are widely employed to assess the quality of word representations and to compare different distributional semantic models. Existing evaluation methods can largely be separated into two categories: "intrinsic evaluation" and "extrinsic evaluation". *Intrinsic evaluation* tries to directly quantify how well various kinds of linguistic regularities can be detected with the model independent of its downstream applications (Baroni et al., 2014; Schnabel et al., 2015). On the other hand, the quality of a word vector may be assessed by its performance in downstream tasks through measuring changes in performance metrics specific to the tasks by *extrinsic evaluation*. The downstream language technology tasks on which the quality of a word embedding is examined, fall into syntactic (e.g. POS tagging, Chunking) and semantic (e.g. Named entity recognition, Sentiment Classification) categories (Schnabel et al., 2015; Chiu et al., 2016b). In this work we evaluate domain-specific word embedding models using both intrinsic and extrinsic evaluation schemes.

Although, word embeddings techniques have drawn significant interest in the field, they are not well equipped to deal with unseen and infrequent words, nor do they consider word relations found in knowledge resources. Recently, different solutions have been proposed to overcome these limitations (Pilehvar and Collier, 2016; Faruqui et al., 2015; Yu and Dredze, 2014). Among these, we choose Faruqui et al. (2015) in this work since it is a post-processing approach which is straightforward to apply.

3. Intrinsic evaluation setup

Intrinsic evaluation of word embeddings has two requirements. First, we require a query inventory as a gold standard, and second, a word embedding model that has been

[2]RepEval @ACL 2016: The First Workshop on Evaluating Vector Space Representations for NLP

Source	Abbr.	Description	Docs	Sentences
American Association of Petroleum Geologist	AAPG	Scientific articles	3,382	72,243
C&C Reservoirs-Digital Analogs	CCR	Field evaluation reports	1,140	244,017
Elsevier	ELS	Scientific articles, magazines	40,757	7,703,447
Geological Society, London Memoirs	GSL	Scientific articles	152	32,352
Norwegian Petroleum Directory	NPD	Norwegian Field info	514	49,426
Tellus	TELLUS	Basin info	1,478	179,450
Total			47,423	8,280,935

Table 1: Sources of the Oil and Gas corpus

trained on a specific corpus. In this section we describe how we build a domain specific query inventory by exploiting a domain-specific knowledge resource. Then, the domain specific corpus and the training of the embedding models will be described. We then go on to clarify the evaluation methodology.

3.1. Domain specific query inventory

For the general domain, there exists a wide range of gold standard resources for evaluating distributional semantic models in their ability to capture semantic relations of different types, for instance, *Simlex-999* (Hill et al., 2015). However, evaluating the domain specific embeddings by applying these gold standards will not provide an adequate picture of their quality, since they do not share a common vocabulary and word meanings. For this reason, we create a domain-specific gold standard using the Schlumberger oilfield glossary (*slb*).[3] The *slb* is a reference which defines major oilfield activities and has been created by technical experts. Terms are described by their part of speech, their discipline (e.g. *Well Completions*, *Geology*), as well as a textual definition. Terms are linked to other terms in the glossary by means of semantic and lexical relations such as *Synonyms*, *Antonyms* and *Alternative forms*. It provides a network of related terms that can be navigated through the glossary. We construct a domain query inventory by extracting all terms and their inter-glossary relations from the relational database. The glossary consists of *4,886* terms. Following the symmetric nature of the *Synonym*, *Antonym* and *Alternative form* relations we infer a relation if it is missing between terms. The final query inventory contains *878* synonym pairs, *284* antonym pairs and *934* alternative form pairs. We observe that the majority of terms in the query inventory are multi-word units (70%) and nouns (72%). This indicates that a large portion of the domain-specific vocabulary that we want to capture in our model consists of multi-word entities. Thus we should take this into account during the training of embeddings.

3.2. Corpora and Pre-processing

In order to train domain-specific embeddings we need a domain-specific corpus. We therefore compile a corpus consisting of technical reports and scientific articles in the Oil and Gas domain. Table 1 shows detailed information about these sources. The corpus contains $47,423$ documents and $8,280,935$ sentences. It is pre-processed using the following steps: 1) Tokenization and lemmatization using StanfordCoreNLP (Manning et al., 2014). English stop words and sentences with less than three words are also removed from the corpus. 2) Shuffling: we randomly shuffle the text in the dataset. During the training of embedding models the learning rate is linearly dropped as training progresses, text appearing early has a larger effect on the model. Shuffling makes the effect of all text almost equivalent (Chiu et al., 2016a).

3.3. Training of Word Embeddings

For training of the word embeddings, we exploit the available word2vec (Mikolov et al., 2013a) implementation *gensim* (Řehůřek and Sojka, 2010). The elements that have an impact on the performance of the model are the input corpus, model architecture and the hyper-parameters. In many works lemmatized, lowercased and shuffled input during training the word2vec are recommended; we carried out our experiments with these settings as detailed above. We employed the phrase model of *gensim* which automatically detects common phrases (multi-word expressions). The phrases are collocations (frequently co-occurring tokens) and we consider bi-grams and tri-grams in this extraction process. We further proceed with the domain specific model generation by creating two sets of embeddings, employing both the *CBOW* and the *Skip-gram* architectures with default settings. In the initial evaluation step, we compare the outcomes of these two models to determine the better architecture.

We then go on to compare different settings for the hyper-parameters, while keeping all other settings constant. It has been shown that optimizations of hyper parameters and certain system choices constitute the main causes of differences in performance rather than the algorithms themselves (Levy et al., 2015). Here we investigate the impact of various system design choices in the evaluation of domain specific embeddings across the following parameters [4]: I) Vector size: $dim \in 50, \mathbf{100}, 200, 300, 400, 500, 600$ II) Context window size: $win \in 2, 3, \mathbf{5}, 10, 15, 20$. III) Negative sampling size: $neg \in 3, \mathbf{5}, 10, 15$. IV) Frequency cut off: $min.count \in 2, 3, \mathbf{5}, 10$. V) n-most-similar: The parameter n for top n-most-similar as output is fixed at value 5 (the maximum number of terms that are involved in each relation set in the query inventory). We evaluate these different system design settings based on our intrinsic benchmark. We build various embeddings models by varying values of

[3] http://www.glossary.oilfield.slb.com/

[4] Default values are in bold

Model	Synonymy			Antonymy			Alt. form		
	A	R	P	A	R	P	A	R	P
Skip-gram	9.8	8.0	2.2	46.4	41.3	9.3	12.1	10.4	2.4
CBOW	**12.7**	**10.2**	**2.7**	**55.3**	**49.2**	**11.1**	**12.8**	**11.0**	**2.6**

Table 2: Evaluation results for different architectures

one hyper-parameter and keeping others as default. Thereafter, we perform evaluation over the domain specific query inventory.

3.4. Evaluation

For evaluation, we assume that for each term in the inventory an embedding model should be able to propose similar words which are related semantically as either *synonym*, *alternative form* or *antonym*. We will measure this by looking at a target word's relation set, for instance its synonyms, and top n-most-similar words based on the embeddings model. Since these relations are symmetric, the pairs (t_i, t_j) and (t_j, t_i) are considered equivalent in the evaluation. We calculate the *accuracy* (A) as the number of target words for which the model provides at least one correct prediction, the *recall* (R) as the number of correctly predicted word pairs over all word pairs and *precision* (P) as the number of correctly predicted word pairs over all predicted word pairs for each relation category.

4. Intrinsic evaluation experiments

In the following we present experiments that evaluate the domain-specific word embedding models intrinsically. We first present tuning experiments and then present an experimental comparison between domain-specific and general domain embedding models.

4.1. Model architecture: Skip-gram vs. CBOW

First, we compare the models obtained using different architectures (CBOW and Skip-gram) with default values for hyper-parameters i.e. $dim = 100$, $win = 5$, $min.count = 5$ and $neg = 5$. Table 2 presents the results for the two architectures broken down by semantic relation from the query inventory. In general we find that the CBOW based model shows better results than the Skip-gram in all semantic relation tasks. The results show that the embedding models have higher scores for *antonymy* prediction than *synonymy*, see Table 2. This result is consistent with previous studies such as van der Plas and Tiedemann (2006) and Leeuwenberg et al. (2016) in which they reported that using distributional similarity some word categories like antonyms, (co)hyponyms or hypernyms show up more often than synonyms.

4.2. Hyper-parameter tuning

We explore the impact of each hyper-parameter on detection of semantic relations. We observe that the performance of the embedding models can be notably improved over the default hyper-parameters but like the findings in other studies (Chiu et al., 2016a; Gladkova et al., 2016), the effects of different configurations are diverse and sometimes they are counter-intuitive. For example, different relation categories benefit from different context windows size in different ways, such as the model with larger context windows tends to capture antonymy relation while with smaller windows, learns synonymy relation of the words. On the other hand, negative sampling and frequency cut-off parameters have different impacts in the three relation categories.

dim	Synonymy			Antonymy			Alt. form		
	A	R	P	A	R	P	A	R	P
50	12.7	10.2	2.7	48.2	42.9	9.6	11.4	9.8	2.3
100	12.7	10.2	2.7	55.4	49.2	11.1	12.9	11.0	2.6
200	14.7	12.4	3.3	55.4	49.2	11.1	14.3	12.3	2.9
300	**15.7**	**13.1**	**3.5**	55.4	49.2	11.1	13.6	11.7	2.7
400	**15.7**	**13.1**	**3.5**	**57.1**	**50.8**	**11.4**	13.6	11.7	2.7
500	14.7	12.4	3.3	53.6	47.6	10.7	**15.0**	**12.9**	**3.0**
600	14.7	12.4	3.3	51.8	46.0	10.4	12.9	11.0	2.6
700	14.7	12.4	3.3	53.6	47.6	10.7	13.6	11.7	2.7

Table 3: Evaluation results for different vector size (default=100)

win	Synonymy			Antonymy			Alt. form		
	A	R	P	A	R	P	A	R	P
2	12.7	10.2	2.7	55.4	49.2	11.1	**13.6**	**12.3**	**2.9**
3	**13.7**	**12.4**	**3.3**	48.2	42.9	9.6	11.4	9.8	2.3
5	12.7	10.2	2.7	55.4	49.2	11.1	12.9	11.0	2.6
10	13.1	10.9	2.9	53.6	47.6	10.7	**13.6**	**12.3**	**2.9**
15	12.7	10.2	2.7	**67.1**	**50.8**	**11.4**	12.9	11.0	2.6
20	12.7	10.2	2.7	53.6	47.6	10.7	12.1	10.4	2.4

Table 4: Evaluation results for different context window size (default=5)

4.2.1. Vector size (*dim*)

The effect of vector size on the trained models is quite similar in all tasks (Table 3). It shows a large improvement in all evaluations when the dimensionality is increased. However, the improvement peaks at 400 for the *synonymy* and *antonymy* predictions and 500 for *alternative form*.

4.2.2. Context window size (*win*)

Table 4 depicts the impact of window size per evaluation task. The embedding model can detect well the synonymy relation in low windows size (w=3) while in antonymy and alternative form tasks the model performance fluctuates between lower and higher window sizes.

4.2.3. Negative sampling (*neg*)

Unlike the practical recommendation in Levy et al. (2015) which states that the skip-gram model prefers many negative samples, the CBOW model shows contradictory result with respect to this parameter in our evaluation benchmarks. As we can see in Table 5, results remain constant regardless of negative sampling number in the synonym prediction task. While its performance has correlation with an increase of this parameter in alternative form detection. For the antonym task, it reached a peak on neg equal to 5 and 10 before falling.

4.2.4. Frequency cut off (*min.count*)

The impact of excluding words that are less frequent regarding to the $min.count$ parameter is summarized in Table 6. This parameter shows different impact compared to the other parameters. While, ignoring more words has better effect in synonymy detection, it stops at $min.count = 3$ for antonymy and alternative form relations.

neg	Synonymy			Antonymy			Alt. form		
	A	*R*	*P*	*A*	*R*	*P*	*A*	*R*	*P*
3	12.7	10.2	2.7	53.6	47.6	10.7	12.1	10.4	2.4
5	12.7	10.2	2.7	**55.4**	**49.2**	**11.1**	12.9	11.0	2.6
10	12.7	10.2	2.7	**55.4**	**49.2**	**11.1**	13.0	11.7	2.7
15	12.7	10.2	2.7	51.8	46.0	10.4	**13.6**	**12.3**	**2.9**

Table 5: Evaluation results for different number of negative samples (default=5)

min.count	Synonymy			Antonymy			Alt. form		
	A	*R*	*P*	*A*	*R*	*P*	*A*	*R*	*P*
2	12.4	9.9	2.7	54.4	48.4	10.9	13.0	11.8	2.7
3	12.6	10.1	2.7	**56.1**	**50.0**	**11.2**	**13.2**	**12.0**	**2.8**
5	12.7	10.2	2.7	55.4	49.2	11.1	12.9	11.0	2.6
10	**13.1**	**10.4**	**2.8**	54.7	48.3	10.9	13.0	11.8	2.7

Table 6: Evaluation results for different value for frequency cut off (default=5)

Since the context window size (*win*), negative sampling (*neg*) and frequency cut off (*min.count*) parameters showed inconsistent results among the relations, we selected the CBOW model with vector size (*dim*) equal to 400 and we fixed the other parameters to their defaults i.e. $win = 5$, $min.count = 5$ and $neg = 5$. This configuration, hereinafter referred to as OILGAS.*d400*, showed the maximum improvement during the tuning experiments.

4.3. Comparative evaluation

In order to compare the domain-specific embeddings with general domain embeddings, we select two widely used pre-trained embedding sets: *Wiki+Giga* [5] and *GoogleNews* [6] to see how they perform in our evaluation benchmark. The input data in the *Wiki+Giga* has been tokenized and lowercased with the Stanford tokenizer, whereas the *GoogleNews* model is trained on a part of the Google News dataset and it contains both words and phrases. The phrases are obtained using the same approach as described in Section 3.2.. The words are not lemmatized in both models and the GoogleNews also contains capitalized words.

The results of the comparative evaluation of the domain-specific and pre-trained models are summarized in Table 7. Since the words in the vocabularies of both pre-trained models are not in lemma form, we consider the surface form of terms for the evaluation. We also report the proportion of query terms that are covered by the vocabulary of each model as coverage. We find that in spite of the large input and vocabulary size in both *GoogleNews* and *Wiki+Giga* models, they have less coverage than the domain specific model. We further observe that despite the considerably smaller training data set, the OILGAS.d400 performs better across all the tasks.

It is clear that this comparison is somewhat unfair due to differences in pre-processing and hyperparameter tuning. In order to investigate the impact of these differences, we apply the same pre-processing steps and hyperparameters to train the CBOW model over the English Wikipedia dump (20 September 2016), here dubbed *enwiki*. Furthermore, we conduct a similar experiment with a data set consisting of both the general and domain specific corpora (*enwiki+*OILGAS). However, these approaches do not show further improvements in our evaluation benchmark, as reported in Table 7. Surprisingly, the mixing of Wikipedia and OILGAS does not increase the coverage rate. It can be attributed to the fact that the phrase extraction method (Section 3.3.) is not able to capture the multi-word expressions, since in many case in mixed corpus the relative increase in the frequency of tokens individually is higher than relative increase of co-occurring tokens (e.g. the relative increase of the word "source" and the word "rock" in the *enwiki+*OILGAS are bigger than relative increase of the word "source rock" compared to the OILGAS corpus)

[5] https://nlp.stanford.edu/projects/glove/

[6] https://code.google.com/archive/p/word2vec/

5. Error Analysis

The results in Section 4.3. show that the domain-specific model provides better results than general domain models for a domain-specific benchmark. However, we also observe that performance is low for all three tasks, in particular for the synonymy detection task. In this section, we explore the reasons behind these low scores and gain insight into the domain specific model predictions, in particular the synonymy detection, through an in-depth error analysis.

As noted above, the primary cause of low performance is due to out of vocabulary (OOV) terms in the query inventory. The model vocabulary contains only 31% of the evaluation dataset. We find that the majority of terms that participate in synonymy relations are not included in the word embeddings model, this is in particular the case for multi-word items. The majority of these terms either do not occur or have a frequency lower than the cut off threshold in the domain dataset. Excluding the OOV terms from the evaluation tasks has an impact on the model performance for synonymy detection, recall (R) is 29% and precision (P) is 6.5%. Still these scores are low, we therefore examine the model predictions closer.

We choose randomly 100 terms from the reference inventory which are also in the model vocabulary and we manually categorize their 10-most-similar words provided in the word embeddings. In this section, we are inspired by the work of Leeuwenberg et al. (2016), where the authors categorized the result of embeddings for a synonym extraction task in the following categories (The categories with $*$ are added by us).

Spelling Variant: The prediction is an abbreviation or there are differences between prediction and target word because of hyphenation.

Alternative or derived form: The prediction is an alternative or derived form of the target word.

Reference-Synonyms: The prediction is a synonym of the target word in the oilfield glossary.

Human-judged Synonyms: The prediction is judged as true by the expert (but is not present in the glossary).

$*$**Antonyms**: The prediction is an antonym of a target term.

Hypernyms: The prediction is a more general category of the target term.

Hyponyms: The prediction is a more specific type of the target term.

Co-Hyponyms: The prediction and target term share a common hypernym.

Model	Coverage	dim	Synonymy			Antonymy			Alt. form		
			A	R	P	A	R	P	A	R	P
Google News	26% (100B, 3M)	300	9.0	7.0	1.8	51.2	37.0	8.1	4.1	1.6	0.4
Wiki+Giga	23% (6B, 400K)	300	4.0	3.2	0.8	40.4	43.8	10.2	1.8	3.7	0.8
OILGAS.d400	31% (108M, 330K)	400	**15.7**	**13.1**	**3.5**	**57.1**	**50.8**	**11.4**	**13.6**	**11.7**	**2.7**
enwiki	29% (1.8B, 2M)	400	8.2	6.7	1.8	39.1	33.3	7.5	8.3	8.1	1.9
enwiki+OILGAS	31% (1.9B, 2.3M)	400	11.1	7.8	2.1	55.3	47.7	10.7	8.6	8.9	2.0

Table 7: Results from the intrinsic comparative evaluation of general domain and domain-specific embedding models.

Category	Example [target→ prediction]	1^{st}:10^{th}(%)
1. Spelling Variant	borehole → bore-hole	2.4
2. Alternative or derived form	acidizing→ acidization	3.2
3. Reference-Synonyms	filter cake → mud cake	2.8
4. Human-judged Synonyms	seismometer → seismograph	8.4
5. Antonyms	transgressive → regressive	0.9
6. Hypernyms	acidizing → stimulation	1.3
7. Hyponyms	EOR → In-situ combustion	9.3
8. Co-Hyponyms	EOR → MEOR	13.1
9. Holonyms	shoe→ wellbore	1.1
10. Meronyms	rig → wellhead	2.8
11. Related	Kirchhoff migration → NMO correlation	35.2
12. Unrelated/Unknown	backflow → sediment-laden	19.5

Table 8: Manual error analysis results for the 10-most-similar words

∗**Holonyms** The prediction denotes a whole whose part is denoted by the target term.
∗**Meronyms**: The prediction is a part of the target term.
Related: The prediction is semantically related to target.
Unrelated/Unknown: The prediction and target terms are semantically unrelated.
Table 8 shows the result of this analysis. In general, the result of this analysis shows that the model predictions are semantically meaningful in a majority of cases and all categories except the *Unrelated/Unknown* represent one type of morphosyntactic or semantic relation between terms. Less than 20% of errors are assigned to the *Unrelated/Unknown* category. It reveals that if we consider the count of *human-judged synonyms* as true positives, the actual scores for precision and recall will be considerable higher than the ones that are reported in the evaluation section. Moreover, the embeddings model proposes more synonyms that are not in the reference, even though the reference is provided by manual procedure. The most frequent error type falls in the *related* category. The *hyponym* and *co-hyponym* relations are another frequent error type that were also reported in previous studies (van der Plas and Tiedemann, 2006; Leeuwenberg et al., 2016). The morphosyntactic type of relations such as *Alternative forms*, *spelling variant* cover another type of errors. The error analysis further reveals several meaningful relation types such as *Hypernyms*, *Meronyms* and *Holonyms* that are useful in many downstream applications.

6. Embedding Enrichment Using a Knowledge Resource

Even though the word embeddings clearly capture important semantic relations in the domain, the first experiment shows that the domain technical vocabulary has many elements which are generally disregarded by the distributional representation techniques. Since these approaches rely only on the statistics derived from textual input, they are incapable of providing representations for words which are not seen frequently in the training process. Furthermore, they do not include the valuable information that is accommodated in domain knowledge resources such as semantic lexicons and glossaries. In this section, we address these issues by applying the work of Faruqui et al. (2015) to exploit prior domain knowledge in enhancing the embeddings model, and induce representations for OOV terms. We then go on to evaluate the impact of the refinement method over an unseen terminological resource.

6.1. Embeddings for infrequent terms

Faruqui et al. (2015) proposed the retrofitting method as a post-processing step to apply to existing pre-trained embeddings. The goal is to refine word vector representations to capture relatedness suggested by semantic lexicons while preserving their similarity to the corresponding embeddings. The objective of the retrofitting method is to minimize the following:

$$\Psi(Q) = \sum_{i=1}^{n} \Big[\alpha_i \|q_i - \hat{q}_i\|^2 + \sum_{(i,j)\in E} \beta_{ij} \|q_i - q_j\|^2 \Big]$$

where $\hat{q} \in \hat{Q}$ is the observed vector representation for each term in the semantic lexicon and $q \in Q$ is the corresponding retrofitted vector. E is the set of relations among the terms in the semantic lexicons. α and β correspond to the relative weights of relation type. Since Ψ is convex in Q, an

Model	Synonymy			Narrower			Broader			Abbr. label		
	A	R	P	A	R	P	A	R	P	A	R	P
OILGAS	25.5	15.5	5.1	12.5	5.0	2.7	4.7	4.4	0.9	2.4	2.3	0.5
OILGAS.retrofitted	27.4	16.7	5.5	12.5	5.0	2.7	4.7	4.4	0.9	2.2	2.2	0.4
OILGAS.retrofitted+OOV	**30.2**	**18.4**	**6.1**	12.5	5.0	2.7	4.7	4.4	0.9	2.4	2.3	0.5

Table 9: Evaluation over the GeoSci knowledge resource.

efficient iterative updating method is used to solve the objective function. Retrofitted embeddings Q are initialized to be equal to the observed ones $\hat{Q}$. Then by taking first derivative of Ψ with respect to q_i the following online update is used for 10 iterations in order to reach convergence:

$$q_i = \frac{\sum_{j:(i,j)\in E} \beta_{ij} q_j + \alpha_i \hat{q}_i}{\sum_{j:(i,j)\in E} \beta_{ij} + \alpha_i} \quad (1)$$

The formula computes a new embedding for a term i which is in the pre-trained model and has relations of interest in the semantic lexicon, whereas its neighbours should be part of the pre-trained model. To provide an embedding for OOV words we extend $\hat{Q}$ in each iteration by adding the terms that are in semantic lexicon and connect to the terms that are in $\hat{Q}$ via relations of interest. Since there is no initial vector for these type of words in the observed model, α is set to zero and the online update formula for the OOV terms will be as follows:

$$q_i = \frac{\sum_{j:(i,j)\in E} \beta_{ij} q_j}{\sum_{j:(i,j)\in E} \beta_{ij}} \quad (2)$$

6.2. Test Data

We use another domain related glossary to perform a quantitative comparison of domain-specific word embeddings before and after the retrofitting process. We create a test query inventory using the same approach as explained in Section 3.1. using the Geoscience Vocabularies data set [7], here dubbed *GeoSci*. GeoSci covers the domain of geology and describes geological features, geological time, mineral occurrences, and mining-related features. It relates terms with syntactically and semantically aligned relations such as *Abbreviated label*, *Synonym* [8], *Broader* and *Narrower*. We construct a test query inventory by extracting all terms and their inter-glossary relations from the RDF files. The test set consists of *1,753* terms. It contains *196* synonym pairs, *1,639* broader pairs, *1,584* narrower pairs and *965* abbreviated label pairs. Like the *slb* glossary, the majority of terms are multi-word units (63%).

6.3. Evaluation

We use the structure of the *slb* glossary as prior domain knowledge to enrich the OILGAS.d400 embeddings model that is evaluated as a best candidate in Section 4.. Experiments in Faruqui et al. (2015) showed that including all semantic relations in the retrofitting process has a better impact than having only one of them. We therefore consider connections of a word to its synonyms, alternative forms and antonyms. Moreover, similar to the origin, all α_i are set to 1 and β_{ij} to be $degree(i)^{-1}$.

The Eq.1 is used to retrofit the OILGAS.d400 model by employing the structure of the semantic lexicon (".retrofitted"). To induce word vectors for OOV terms , we carry out the retrofitting process with Eq.2 (".retrofitted+OOV"). Table 9 shows the performance of the model in the test dataset as well as the retrofitted models with two different configurations. We observe that the retrofitting process provides improvement in the *synonymy* relation. The improvement is highest when we consider the adapted version (retrofitted+OOV). Interestingly the retrofitted models have no impact in the *narrower* and *broader* relationships, this can be attributed to the fact that the employed semantic lexicons do not include these kinds of associations to lead the retrofitting process. In the *abbreviated label* relation, there is a slightly negative effect when we apply the original retrofitting process.

7. Extrinsic evaluation

While the intrinsic evaluations attempt to interpret the encoding content of an embedding model in terms of lexical semantic relations, *extrinsic* evaluation investigates the contribution of an embedding model to the performance of a specific downstream task. In this section, we investigate the influence of our domain-specific model in a domain related classification task.

7.1. Classification Data Set

The task of the exploration department in Oil and Gas industry is to find exploitable deposits of hydrocarbons (oil or gas). Geoscientists in the exploration department model the subsurface geography by classifying rock layers according to multiple stratigraphic hierarchies using information from a wide range of different sources. The quality of the analysis depends on the availability and the ease of access to the relevant data. Previous technical studies, reports and surveys are crucial resources in this process.

We collect sentences from exploration textual documents which are then manually labeled with various geological type properties by domain experts. Example 1 shows an example sentence from the data set along with its assigned set of properties.

Example 1. *Submarine fans and deltaic/estuarine facies of the San Juan Formation were deposited during the Maastrichtian regression, which gave way during the Paleocene-Eocene to black marine shales and carbonates of the Vidoño Formation and the shelfal and pro-delta shales of the Caratas Formation.*

Properties: Lithology_RockType, Lithology_Main, DepEnv_Sub, DepEnv_General

[7] http://resource.geosciml.org/

[8] GeoSic vocabulary specifies this relation as *Alternative label*.

Model	DepEnv_Sub	Lith_Main	BasinType	DepEnv_Gen	Facies	DepEnv_Main	Lith_RockType
	$F1$	$F1$	$F1$	$F1$	$F1$	$F1$	$F1$
CNN.rand	28.9	90.6	0.0	0.0	63.0	57.1	68.2
CNN.domain	51.1	91.4	23.9	11.3	71.1	66.3	65.8
CNN.multi.rand	38.0	91.4	7.3	5.0	63.9	58.6	69.9
CNN.multi.enwiki	43.9	90.5	11.3	0.0	61.1	61.4	57.8
CNN.multi.domain	56.2	92.2	**33.8**	**15.0**	71.7	69.4	**72.5**
CNN.multi.retrofitted+OOV	64.0	91.3	11.3	0.0	67.6	68.8	72.2
CNN.multi.domain &retrofitted+OOV	**68.2**	**92.8**	32.0	9.4	**73.4**	**73.5**	71.1
CNN.multi.retrofitted+OOV&domain	53.4	92.6	20.9	10.0	71.8	67.0	70.7

Table 10: Results of the classification task with various configurations

Property	# Sentences
Lithology_ Main	1,193
Lithology_ RockType	191
DepEnv_ General	38
DepEnv_ Main	483
DepEnv_ Sub	298
Facies	387
BasinType	49

Table 11: Classification data set

The resulting data set contains 1,348 sentences in which experts assigned each sentence to 7 different properties. The sentences are pre-processed using the same approach as described in Section 3.2. Table 11 depicts the properties and number of sentences for each. It can be seen that the data set is unbalanced regarding to the properties and that the downstream task is a multi-label classification task.

7.2. Multi-label Classification Model

We use a slight variant of the *Convolutional Neural Network* (CNN) architecture that is proposed by Kim (2014) for a sentence classification task. We keep the value of hyperparameters equal to the ones that are reported in the original work, however we update the dimension of the embeddings layer according to the dimension of the domain-specific embeddings model. Furthermore, since the architecture aims to assign a single label to each sentence, we update the activation function to *sigmoid* at the output layer, which produces a probability for each of the potential properties. During training, these probabilities are used to compute the error, while during testing, we round each of the probabilities to 0 or 1 depending upon a set threshold (0.5).

7.3. Extrinsic evaluation experiments

Like Kim (2014), we run experiments with several variants of the model as follows: **CNN.rand**: As a baseline model, where all words in the embedding layer are randomly initialized and updated in the training process. **CNN.domain**: the embedding layer is initialized with a domain-specific model and fine-tuned for the target task. **CNN.multi.rand**: There are two embedding layers as a 'channel' in the CNN architecture. Both channels are initialized randomly and only one of them is updated during training while the other remains static. **CNN.multi.domain**: Same as before, but the channels are initialized with domain-specific vectors. **CNN.multi.enwiki:** The channels consider the general domain word vectors from section 4.3. using the English Wikipedia data. To deal with effects of an unbalanced dataset and guarantee that each fold in 5-fold cross validation will have the proportion of same classes during training and test, we apply the stratification of multi-label data proposed by Sechidis et al. (2011).

Results of the classification task with various CNN configurations are presented in the first section of Table 10. In general, the multi-channel mode performs better than the single channel setting. The results suggest that having a significant amount of sentences per property assists the CNN model to classify better. The baseline model does not perform well on its own. The use of the pre-trained embeddings model helps the model in property assignment. Particularly, domain-specific embeddings provide higher performance gain in the task-at-hand when it is used in both channels.

We also investigate the influence of the refined word embeddings model in our classification task. **CNN.multi.retrofitted+OOV:** We used the retrofitted domain embeddings including the OOV vectors generation for two channels. One channel is static and the other is non-static. **CNN.multi.domain&retrofitted+OOV:** First channel is initialized with original domain-specific embeddings with static mode and the second makes use of the retrofitted embeddings with a non-static mode. **CNN.multi.retrofitted+OOV&domain:** Same as previous setting, but the channels swap their input. In these experiments, because of having many multi-words as OOV terms in the model, we replaced tokens in the sentences with their bi-gram and tri-gram forms if their combination occurs in the model vocabulary (e.g. 'fracture porosity' is replaced to 'fracture_porosity' as a input unit). The experiment (second section of Table 10) shows that the enhanced embedding models provides better input representations for classes with a sufficient number of instances.

8. Conclusion

In the present work we demonstrate that constructing domain-specific word embeddings is beneficial even with limited input data. Nevertheless, the empirical evaluation shows that the distributional models have low performance in domain-specific synonymy detection, an in-depth manual error analysis reveals the striking ability of the embedding models to discover other semantic relations such as (co)hyponymy, hypernymy and relatedness. We further showed the importance of dealing with rare words in an embedding model in both intrinsic and extrinsic evaluation.

9. Bibliographical References

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 238–247. Association for Computational Linguistics.

Bollegala, D., Maehara, T., and ichi Kawarabayashi, K. (2015). Learning word representations from relational graphs. In *Proceedings of 53rd Annual Meeting of the Association for Computational Linguistics, and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 730–740. Association for Computational Linguistics.

Chiu, B., Crichton, G. K. O., Korhonen, A., and Pyysalo, S. (2016a). How to train good word embeddings for biomedical NLP. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing,BioNLP@ACL*, pages 166–174. Association for Computational Linguistics.

Chiu, B., Korhonen, A., and Pyysalo, S. (2016b). Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, RepEval@ACL*, pages 1–6. Association for Computational Linguistics.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E. H., and Smith, N. A. (2015). Retrofitting word vectors to semantic lexicons. In *The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615. Association for Computational Linguistics.

Gladkova, A., Drozd, A., and Matsuoka, S. (2016). Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the Student Research Workshop, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 8–15. Association for Computational Linguistics.

Hamilton, W. L., Clark, K., Leskovec, J., and Jurafsky, D. (2016). Inducing domain-specific sentiment lexicons from unlabeled corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 595–605. Association for Computational Linguistics.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751. Association for Computational Linguistics.

Leeuwenberg, A., Vela, M., Dehdari, J., and van Genabith, J. (2016). A minimally supervised approach for synonym extraction with word embeddings. *The Prague Bulletin of Mathematical Linguistics*, 105:111–142.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistic*, 3:211–225.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 55–60. Association for Computer Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *Computing Research Repository*, abs/1301.3781.

Mikolov, T., Yih, W., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*, pages 746–751. Association for Computational Linguistics.

Pilehvar, M. T. and Collier, N. (2016). Improved semantic representation for domain-specific entities. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, pages 12–16. Association for Computational Linguistics.

Řehůřek, R. and Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*, pages 46–50. European Language Resources Association.

Schnabel, T., Labutov, I., Mimno, D. M., and Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307. Association for Computational Linguistics.

Sechidis, K., Tsoumakas, G., and Vlahavas, I. (2011). On the stratification of multi-label data. In *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 145–158. Springer.

van der Plas, L. and Tiedemann, J. (2006). Finding synonyms using automatic word alignment and measures of distributional similarity. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 866–873. Association for Computational Linguistics.

Yu, M. and Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 545–550. Association for Computer Linguistics.

Automatic Thesaurus Construction for Modern Hebrew

°Chaya Liebeskind, *Ido Dagan, *Jonathan Schler
°Computer Science Department, Jerusalem College of Technology, Lev Academic Center, Israel
*Computer Science Department, Bar-Ilan University, Israel
liebchaya@gmail.com, dagan@cs.biu.ac.il, schler@gmail.com

Abstract

Automatic thesaurus construction for Modern Hebrew is a complicated task, due to its high degree of inflectional ambiguity. Linguistics tools, including morphological analyzers, part-of-speech taggers and parsers often have limited in performance on *Morphologically Rich Language*s (MRLs) such as Hebrew. In this paper, we adopted a schematic methodology for generating a co-occurrence based thesaurus in a MRL and extended the methodology to create distributional similarity thesaurus. We explored three alternative levels of morphological term representations, surface form, lemma, and multiple lemmas, all complemented by the clustering of morphological variants. First, we evaluated both the co-occurrence based method and the distributional similarity method using Hebrew WordNet as our gold standard. However, due to Hebrew WordNet's low coverage, we completed our analysis with a manual evaluation. The results showed that for Modern Hebrew corpus-based thesaurus construction, the most directly applied statistical collection, using linguistics tools at the lemma level, is not optimal.

Keywords: Hebrew, Morphologically Rich Language, thesaurus, co-occurrence, distributional similarity

1. Introduction and Background

A thesaurus is a lexical resource which groups words together by *semantic similarity*. The notion of semantic similarity includes semantic relations, such as synonyms (*car-automobile*), hyperonyms (*vehicle-car*), hyponyms (*car-vehicle*), meronyms (*wheel-car*), and antonyms (*acceleration-deceleration*).

Generally, two statistical approaches for corpus-based thesaurus construction were explored: a first-order, co-occurrence-based approach which assumes that words that frequently occur together are topically related (Schütze and Pedersen 1997) and a second-order, distributional similarity approach (Hindle 1990; Lin 1998; Gasperin et al. 2001; Weeds and Weir 2003; Kotlerman et al. 2010) which suggests that words occurring within similar contexts are semantically similar (Harris 1968).

The aim of this research was to construct a thesaurus for Modern Hebrew. Modern Hebrew lacks repositories of machine-readable knowledge (i.e, lexical resources) fundamental to many Natural Language Processing (NLP) tasks, such as machine translation (Irvine and Callison-Burch 2013), question answering (Yang et al. 2015), and word sense disambiguation (Chen et al. 2014). Therefore, in the last decade, a few semantic resources for modern Hebrew have been constructed. Examples for such resources are Hebrew WordNet[1] (Ordan et al. 2007), which groups words into sets of synonyms, and Hebrew FrameNet (Hayoun and Elhadad 2016), which defines formal structures for semantic frames, and various relationships between and within them.

Furthermore, Hebrew is a *Morphologically Rich Language* (MRL), a language in which significant information about syntactic units and relations is expressed at word-level. Due to the rich and challenging morphology of MRLs, tools, such as part-of-speech taggers, and parsers, often perform poorly. In MRLs, commonly used statistical extraction at the lemma level, using a morphological analyzer and tagger (Lindén and Piitulainen 2004; Peirsman et al. 2008; Rapp 2008), might not be the optimal choice.

Therefore, Liebeskind et al. (2012) suggested a methodology for generating a co-occurrence based thesaurus in MRL. They explored three options for term representation: surface form, lemma, and multiple lemmas, all supplemented by term variant clustering. While the default lemma representation is dependent on tagger performance, the two other representations avoid choosing the right lemma for each word occurrence. Instead, the multiple-lemma representation assumes that the correct analysis will accumulate enough statistical prominence throughout the corpus; while by clustering term variants at the end of the extraction process, the surface representation solves morphological disambiguation "in retrospect". The input is a thesaurus entry (target term) in one of the possible term representations (surface, best, or all). Their algorithm is as follow: for each target term, retrieve all the corpus documents where the target term appears. Then, Liebeskind et al. (2012) define a set of candidate terms that consists of all the terms that appear in all these documents (this again for each of the three possible term representations). Next, a co-occurrence score (e.g., *Dice coefficient* (Smadja et al. 1996), *Pointwise Mutual Information* (PMI) (Church and Hanks 1990) and *log-likelihood test* (Dunning 1993)) between the target term and each of the candidates is calculated. Then, candidates are sorted, and the highest rated candidate terms are clustered into lemma-oriented clusters. Finally, the clusters are ranked by their members' co-occurrence scores and the highest rated clusters become related terms in the thesaurus. The two choices for term representation are independent, resulting in nine possible configurations of the algorithm for representing both the target term and the candidate terms. The methodology provides a generic scheme for exploring the alternative representation levels, each corpus and language-specific tool set might yield a different optimal configuration.

In this paper, we adopt and extend Liebeskind et al. (2012) algorithmic scheme to deal with second-order distributional similarity methods. In the second-order statistical approach, there is an additional dimension, feature representation. For each term in the corpus, a

[1] http://cl.haifa.ac.il/projects/mwn/index.shtml

feature vector is constructed by collecting terms that appear in its context. Each *feature term* is assigned a weight indicating its association to the given term. Then, second-order similarity is calculated between the target term and all the other terms in the corpus. (e.g., *Jaccard's coefficient* (Gasperin et al. 2001), *Cosine-similarity* (Salton and McGill 1983; Ruge 1992; Caraballo 1999; Gauch et al. 1999; Pantel and Ravichandran 2004) and *Lin's mutual information* metric (Lin 1998)). Therefore, each term in the corpus is a potential *candidate term*. Yet, the ranked list of terms is considered as the *candidate terms* list. The three choices for term representation are independent, resulting in 27 configurations which cover the range of possibilities for term representation in second-order thesaurus. Exploring all of them in a systematic manner should reveal the best configuration for a particular setting.

In Section 2, we aim to describe our corpora, target term collection, and the experimental setting. Section 3 elaborates on the results of the algorithmic scheme for both first-order and second-order statistical extraction. We complete this section with a post-hoc manual evaluation and an error analysis. In Section 4, we suggest directions for future research.

2. Modern Hebrew Thesaurus Construction

2.1 Corpora

The MILA Knowledge Center[2] has acquired a number of Hebrew corpora from various domains. We used four corpora for the thesaurus construction experiment: 2 news corpora: HaAretz (11,097,790 word tokens) and Arutz 7 (15,107,618 word tokens), a financial newspaper: TheMarker (692,919 word tokens), and the sessions protocols of the Israeli parliament, HaKnesset (15,066,731 word tokens).

2.2 Target Terms Collection

We used Hebrew WordNet (Ordan et al. 2007) as our gold-standard. English WordNet (Fellbaum 1998) is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual semantic and lexical relations. WordNet superficially resembles a thesaurus; words are grouped based on their meanings.

Following the success of the English WordNet, parallel networks have been developed for a variety of languages. In particular, researchers from Italy have developed a methodology for parallel development of multilingual WordNets (Bentivogli et al. 2000). The system, called MultiWordNet[3], contains information on several aspects of multilingual dictionaries, including lexical relationships between words, semantic relations between lexical concepts, and several mappings of lexical concepts in different languages, etc.

Hebrew WordNet uses the MultiWordnet methodology (Bentivogli et al. 2000) and is thus aligned with English, Italian, Spanish, Portuguese, Romanian and Latin. The Hebrew WordNet currently contains 5261 synsets, with an average of 1.47 synonyms per synset, where nouns are much more frequent than other parts of speech (almost 78 percent).

Our target terms list consisted of all 3362 Hebrew WordNet terms with at least 5 appearances in our Modern Hebrew corpora. The target terms were of different parts of speech, including nouns, verbs, adjectives and adverbs.

2.3 Experimental Setting

2.3.1 Morphological Tools

We used the MILA's automatically morphologically analyzed texts. The texts are tokenized and tagged with all possible morphological analyses. Therefore, we directly extracted all the three possible term representations from the analyzed texts. The morphological analysis of the texts was performed by the MILA Hebrew Morphological Analyzer (Itai and Wintner 2008; Yona and Wintner 2008) and POS tagging was performed by the Bar Haim et al. (2008) tagger. The reported accuracy of the tagger for Modern Hebrew is 90%.

2.3.2 Evaluation measures

Following previous works (Lin 1998; Riedl and Biemann 2013; Melamud et al. 2014), we used Hebrew Wordnet to construct a large scale gold standard for semantic similarity evaluations. In our experiments, we compared the performance of our algorithms by Relative Recall (RR) and Mean Average Precision (MAP). Since each related terms list is of different length, these measures ensure that any recall increase is reflected in both scores. The recall scores are *micro-averaged*; we sum the number of extracted WordNet related terms for all the target terms for each configuration and then divide the sum by the total number of related terms for all target terms.

2.3.3 Methods

We applied the Liebeskind et al. (2012) methodology for generating a first-order co-occurrence based thesaurus and extended the methodology for generating a second-order distributional similarity thesaurus as described in Section 1. The Pointwise mutual information (Church and Hanks 1990) was used as our co-occurrence measure for both first-order similarity and feature weighting in second-order similarity. In this paper, we focused on the word window context representation, which is the most common one for lexical similarity extraction. This representation maintains comparable performance for alternative context representations (Bullinaria and Levy 2012; Kiela and Clark 2014; Lapesa and Evert 2014)). We used a sliding window of 3 words on each side of the represented word, not crossing sentence boundaries. At the end of the extraction process, we grouped the top 50 related term variants and ranked them based on the summation approach. The summation approach adds up the group members' scores as the group score. The approach accounts for the cumulative impact of all group members, which corresponds to the morphological variants of each candidate term.

3. Results

In general, due to the low coverage of Hebrew WordNet, we expected the measured precision of our methods to be low. For example, the statistical methods extracted related

[2] http://www.mila.cs.technion.ac.il/resources corpora.html

[3] http://multiwordnet.fbk.eu/english/home.php

terms such as m`lh[4] (degree) and clsiws (Celsius) for the target term TmprTwrh ((physics) temperature). However, these related terms do not appear in Hebrew WordNet and thus would be judged irrelevant. In addition, WordNet makes abstract generalizations that we did not expect to be extract by statistical measures. For example, the related terms: mišhw (someone), xi (living thing) and adm (person) for the target term ab (father), or the related terms: p`wlh (action) and m`šh (deed) for the target term ktivh (writing). Therefore, our measured recall was expected to be low. Still, even though the absolute values of our evaluation measures was expected to be low.

Candidate► / **Target▼**		**Surface**	**Best**	**All**
Surface	RR	0.0172	0.0122	0.0133
	MAP	0.0057	0.0022	0.0026
Best	RR	**0.0208**	0.0111	0.0124
	MAP	**0.0064**	0.0024	0.003
All	RR	0.0169	0.0092	0.0098
	MAP	0.0051	0.0019	0.0019

Table 1: Results for first-order method

Hebrew WordNet is still a valuable resource for comparing performance of different statistical methods. In addition, we performed a post-hoc manual evaluation, which better indicated our performance level in absolute terms. The manual evaluation also shows the contribution of statistical methods to the construction and enrichment of lexical resources for resource-poor languages, such as Hebrew.

In the next subsections, we report the results of the first-order and second-order thesaurus construction methods. We examine the thee levels of term representations: surface form (Surface), best lemma, as identified by a POS tagger (Best), and all possible lemmas produced by a morphological analyzer (All).

3.1 First-order Results

Table 1 compares the performance of all nine term representation configurations for the first-order method. The lemma-based representations of the target term outperformed its surface representation. The second-best results were obtained from the candidate representation at the surface level, which was complemented by grouping term variants to lemmas in the grouping phase. The improvement over the common default best lemma representation, for both target and candidate, is statistically significant at the 0.01 level for both MAP and RR.

Target►		**Surface**			**Best**			**All**		
Candidate►		Surface	Best	All	Surface	Best	All	Surface	Best	All
Feature▼	RR	0.0413	0.0137	0.0125	0.0452	0.0203	0.0216	0.0434	0.0216	0.0245
Surface	MAP	0.0174	0.0103	0.0107	0.0204	0.0181	0.0143	**0.0271**	0.0181	0.0209
	RR	0.0452	0.0161	0.017	0.048	*0.0208*	0.0269	0.0454	0.0244	0.0259
Best	MAP	0.0164	0.0127	0.0119	0.0197	*0.0064*	0.0147	0.0261	0.0183	0.0197
	RR	0.0414	0.0155	0.0151	**0.0488**	0.0209	0.0254	0.0441	0.0244	0.0245
All	MAP	0.0164	0.0127	0.01	0.0189	0.0124	0.0147	0.0266	0.018	0.0197

Table 2: Results for second-order method

Target term	**WordNet related terms**	**DistSim related terms**
akiph (enforcement)	bcw` (execution)	piqwx (supervision), bqrh (inspection)
bkwx (aggressively)	xzq (strong)	`wcmh (strength)
zkrwn (memory)	qwgnicih (psychology) cognition), zkr memory (of someone or something)	andrTh (monument), azkrh (memorial)
TmprTwrh ((physics) temperature)	qr (cold), xm (hot)	m_lh (degree), clsiws (Celsius)
nwrmTibi (normative, regular)	pwsqni ((linguistics) normative)	nwrmli (normal), tqin (in order), nwrmh (norm)
crxh (scream)	tq_srt (communication), cwxh (shout), amirh (saying)	c`q (to shout), dibwr (speaking)
šitwq (paralysis)	štwq (paralysis)	qipawn (standstill), hšbth (stoppage)
ti`wb (disgust)	mšTmh ((flowery) hatred), aibh (loathing), šnah (hatred)	bwz (contempt), slidh (disgust), ginwi (denunciation)

Table 3: WordNet vs. distributional similarity (DistSim) related terms

[4] To facilitate readability, we use a transliteration of Hebrew using Roman characters; the letters used, in Hebrew lexicographic order, are abgdhwzxTiklmns`pcqršt.

3.2 Second-order Results

The performance of all the 27 representation configurations is presented in Table 2. The best MAP score of the second-order method was obtained from target term representation at the all-lemma level, feature representation at the surface level and candidate representation at the surface level. The improvement over the common default best lemma representation, for target, feature and candidate, is statistically significant at the 0.01 level for both MAP and RR. The common default best lemma representation for target, feature and candidate is italicized in Table 2. This default configuration has relatively low performance. Its recall is ranked 20 out of the 27 configurations and its MAP is the lowest, while, even the default surface representation for both target, feature and candidate has higher rank (9 and 13 respectively).

Rapp (2002) observed that there is a relationship between the type of computation performed (first-order versus second-order) and the type of the extracted association (syntagmatic versus paradigmatic). Whereas the results of the second order computations are exclusively paradigmatic, the first order computations are a combination of both syntagmatic and paradigmatic associations. Since we limited preferences to lexical similarity, a thesaurus based on second order associations is better than one based on first-order associations.

We analyzed the results of the best configuration (*all-surface-surface*) and found that due to the limitations of the Hebrew WordNet some truly related terms found by our system were judged irrelevant. Since these terms do not appear in Hebrew WordNet, they were counted as false positives and decreased the overall MAP. Table 3 shows examples of target terms with their WordNet related terms and additional related terms that were extracted by the second-order algorithm.

3.3 Manual evaluation

Due to the low coverage of Hebrew WordNet and the poor results in the previous analysis, we performed a post-hoc manual evaluation to assess our preliminary findings. We randomly selected 30 target terms and annotated their results.

First, we compared the related term extracted by WordNet to the related terms extracted by the best distributional similarity configuration (*all-surface-surface*). While WordNet relative recall was only 0.23, the MAP of the best configuration was 0.38 and its relative recall was 0.78. The intersection between the two sources of related terms was low.

Next, we chose 4 configurations; 2 first-order configurations and 2 second-order configurations. In each configuration pair, the first was the best performing configuration and the second was the default best lemma representation for all the term levels. The comparison results are presented in Table 4.

The ranking of the four configurations is consistent with the full ranking of all the possible configurations of our previous evaluation (Tables 1 and 2). Since we did not manually compare all the configurations, we cannot guarantee that the full rankings are consistent. However, we can still assess the benefits of our methodology by assessing its statistical significance. We used the one-sided Wilcoxon signed-rank test (Wilcoxon 1945) and observed that the advantage of the best first-order configuration over the default best lemma representation is statistically significant at the 0.05 level for MAP. While the advantage of the best second-order configuration over the default best lemma representation is statistically significant at the 0.01 level for both RR and MAP. The relatively high MAP score of the best second-order configuration is consistent with the results in prior research for statistical thesaurus construction methods.

The best results for both the first and second order methods were obtained from candidate representation at the surface level, which was complemented by grouping term variants to lemmas in the grouping phase. The effectiveness of solving morphological disambiguation "in retrospect" was significant. Considering the relatively high accuracy of the tagger on Modern Hebrew corpora (90%), this finding is somewhat surprising.

Since available morphological tools were designed for processing modern Hebrew, we hardly recognized lemmatization errors in the grouping phase. The lemmatizer's lexicon covers modern Hebrew well and no invalid derivations were found in the modern corpora.

We also analyzed the results of the best configuration, the *all-surface-surface* configuration of the second-order method. We observed that about 15% of the related terms groups had a different lemma than the target term, but shared a similar root with the target term, often corresponding to a morphological derivation rather than an inflection. For example: kwšl (failing) - kišlwn (failure), akilh (eating) - makl (food), hmcah (invention) - mmcia (inventor), and dliqh (fire) - dliq (flammable).

In addition, we performed an error analysis and found that 62% of the related terms groups were indeed irrelevant. However, 38% of the groups shared a broader context with the target term and were judged irrelevant in the current use case. This broader contextual group may include verbs associated with the target term or its related terms, or different nouns associated with the related terms. For example: akilh (eating) - dg (fish), swkr (sugar); bkwx (aggressively) – xiil (soldier); bki (crying) - xrdh (fear); and lqixh (taking) - hlwwah (loan), pir`wn (redemption).

Due to the problematic gold-standard (Hebrew Wordnet), we do not have a decisive conclusion on the best configuration for term representation. However, we did demonstrate the importance of the methodological scheme by showing that the default configuration is definitely not the optimal one.

Configuration	RR	MAP
best-surface	0.18	0.05
best-best	0.15	0.03
all-surface-surface	0.55	0.28
best-best-best	0.19	0.1

Table 4: Manual comparison of 4 configurations

4. Conclusions and Future work

We constructed a Modern Hebrew thesaurus by adopting and extending a first-order co-occurrence based methodological method to second-order distributional similarity method. We showed that our methodology is effective for constructing a more comprehensive thesaurus for MRLs. Our automatic thesaurus construction tool outperforms the current term representation and yields an optimal configuration. We plan to extend our methodology to deal with Multi Word Expressions (MWE).

5. Bibliographical References

Bar-haim, R., Sima'an, K., and Winter, Y. (2008). Part-of-speech tagging of modern hebrew text. Natural Language Engineering, 14(2), 223–251.

Bullinaria, J. A., and Levy, J. P. (2012). Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD. Behavior research methods, 44(3), 890–907.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics (pp. 120–126). Association for Computational Linguistics.

Chen, X., Liu, Z., and Sun, M. (2014). A unified model for word sense representation and disambiguation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1025–1035).

Church, K. W., and Hanks, P. (1990). Word association norms, mutual information, and lexicography. Computational linguistics, 16(1), 22–29.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. Computational linguistics, 19(1), 61–74.

Gasperin, C., Gamallo, P., Agustini, A., Lopes, G., and Lima, V. de. (2001). Using Syntactic Contexts for Measuring Word Similarity. In the Workshop on Semantic Knowledge Acquisition and Categorisation (ESSLI 2001).

Gauch, S., Wang, J., and Rachakonda, S. M. (1999). A corpus analysis approach for automatic query expansion and its extension to multiple databases. ACM Transactions on Information Systems (TOIS), 17(3), 250–269.

Harris, Z. S. (1968). Mathematical Structures of Language. New York, NY, USA: Wiley.

Hindle, D. (1990). Noun classification from predicate-argument structures. In Proceedings of the 28th annual meeting on Association for Computational Linguistics (pp. 268–275). Association for Computational Linguistics.

Irvine, A., and Callison-Burch, C. (2013). Combining bilingual and comparable corpora for low resource machine translation. In Proceedings of the eighth workshop on statistical machine translation (pp. 262–270).

Itai, A., and Wintner, S. (2008). Language resources for Hebrew. Language Resources and Evaluation, 42(1), 75–98.

Kiela, D., and Clark, S. (2014). A systematic study of semantic vector space model parameters. In Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) at EACL (pp. 21–30).

Kotlerman, L., Dagan, I., Szpektor, I., and Zhitomirsky-Geffet, M. (2010). Directional distributional similarity for lexical inference. Natural Language Engineering, 16(4), 359–389.

Lapesa, G., and Evert, S. (2014). A Large Scale Evaluation of Distributional Semantic Models: Parameters, Interactions and Model Selection. Transactions of the Association for Computational Linguistics, 2, 531–545.

Liebeskind, C., Dagan, I., and Schler, J. (2012). Statistical Thesaurus Construction for a Morphologically Rich Language. In *SEM 2012: The First Joint Conference on Lexical and Computational Semantics (pp. 59–64). Montréal, Canada.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In Proceedings of the 17th international conference on Computational Linguistics-Volume 2 (pp. 768–774). Association for Computational Linguistics.

Lindén, K., and Piitulainen, J. (2004). Discovering Synonyms and Other Related Words. In S. Ananadiou and P. Zweigenbaum (Eds.), COLING 2004 CompuTerm 2004: 3rd International Workshop on Computational Terminology (pp. 63–70). Geneva, Switzerland.

Melamud, O., Dagan, I., Goldberger, J., Szpektor, I., and Yuret, D. (2014). Proceedings of the Eighteenth Conference on Computational Natural Language Learning (pp. 181–190). Ann Arbor, Michigan: Association for Computational Linguistics.

Pantel, P., and Ravichandran, D. (2004). Automatically labeling semantic classes. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004.

Peirsman, Y., Heylen, K., and Speelman, D. (2008). Putting things in order. First and second order context models for the calculation of semantic similarity. In 9es Journées internationales d'Analyse statistique des Données Textuelles (JADT). Lyon, France.

Rapp, R. (2002). The Computation of Word Associations: Comparing Syntagmatic and Paradigmatic Approaches. In COLING (Vol. 1, pp. 1–7). Stroudsburg, PA, USA.

Rapp, R. (2008). The automatic generation of thesauri of related words for English, French, German, and Russian. International Journal of Speech Technology, 11(3–4), 147–156.

Riedl, M., and Biemann, C. (2013). Scaling to Large3 Data: An Efficient and Effective Method to Compute Distributional Thesauri. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL (pp. 884–890).

Ruge, G. (1992). Experiments on Linguistically-Based Term Associations. Information Processing and Management, 28(3), 317–332.
Salton, G., and McGill (Eds.). (1983). Introduction to Modern Information Retrieval. McGraw-Hill.
Schütze, H., and Pedersen, J. O. (1997). A cooccurrence-based thesaurus and two applications to information retrieval. Information Processing and Management, 33(3), 307–318.
Smadja, F., McKeown, K. R., and Hatzivassiloglou, V. (1996). Translating collocations for bilingual lexicons: a statistical approach. Computational Linguistics, 22(1), 1–38.
Weeds, J., and Weir, D. (2003). A general framework for distributional similarity. In Proceedings of the 2003 conference on Empirical methods in natural language processing (pp. 81–88). Association for Computational Linguistics.
Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. Biometrics Bulletin, 1(6), 80–83.
Yang, Y., Yih, W., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 2013–2018).
Yona, S., and Wintner, S. (2008). A finite-state morphological grammar of hebrew. Natural Language Engineering, 14(2), 173–190.

6. Language Resource References

Bentivogli, L., Pianta, E., and Pianesi, F. (2000). Coping with Lexical Gaps when Building Aligned Multilingual Wordnets. In Proceedings of the Second International Conference on Language Resources and Evaluation, LREC 2000, 31 May - June 2, 2000, Athens, Greece.
Fellbaum, C. (1998). WordNet. Wiley Online Library.
Hayoun, A., and Elhadad, M. (2016). The Hebrew FrameNet Project. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Portorož, Slovenia: European Language Resources Association (ELRA).
Ordan, N., Ilan, B., Ordan, N., and Wintner, S. (2007). S.: Hebrew WordNet: a test case of aligning lexical databases across languages. International Journal of Translation, 19, 39–58.

Automatic Wordnet Mapping: from CoreNet to Princeton WordNet

Jiseong Kim, Younggyun Hahm, Sunggoo Kwon, Key-Sun Choi
Semantic Web Research Center, School of Computing, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon, Korea
{jiseong, hahmyg, fanafa, kschoi}@kaist.ac.kr

Abstract

CoreNet is a lexico-semantic network of 73,100 Korean word senses, which are categorized under 2,937 semantic categories organized in a taxonomy. Recently, to foster the more widespread use of CoreNet, there was an attempt to map the semantic categories of CoreNet into synsets of Princeton WordNet by lexical relations such as synonymy, hyponymy, and hypernymy relations. One of the limitations of the existing mapping is that it is only focused on mapping the semantic categories, but not on mapping the word senses, which are the majority part (96%) of CoreNet. To boost bridging the gap between CoreNet and WordNet, we introduce the automatic mapping approach to link the word senses of CoreNet into WordNet synsets. The evaluation shows that our approach successfully maps previously unmapped 38,028 word senses into WordNet synsets with the precision of 91.2% ($\pm$1.14 with 99% confidence).

Keywords: wordnet, automatic mapping, automatic construction

1. Introduction

CoreNet (Choi et al., 2004) is a semantic hierarchy of Korean word senses, which has been built by KAIST since 1994 based on CoreNet concept hierarchy originated from NTT Goi-Taikei (Ikehara et al., 1997) concept hierarchy.

The CoreNet hierarchy comprises mainly two parts, non-terminal part and terminal part. The non-terminal part of the hierarchy comprises 2,937 semantic categories, called CoreNet concept, as non-terminal nodes, which are organized by a taxonomic relation, while the terminal part of the hierarchy comprises 73,100 Korean word senses as terminal nodes, which are separated from each other (i.e., there is no link between the word senses) and there is only a link between word sense and its semantic category with unknown lexical relations such as *is-a* and *part-of*; the unknown relation means there may be *is-a* or *part-of* relation between word sense and its semantic category, but a label of the relation is not revealed. An example of the CoreNet hierarchy is shown in Figure 1.

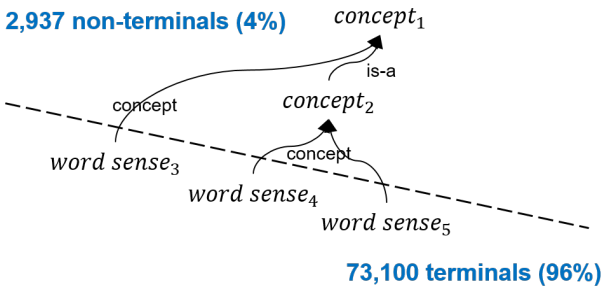


Figure 1: An example of the hierarchy of CoreNet

To extend CoreNet into other languages and to promote its broader utilization for diverse NLP application, Kang et al. (2010) made an attempt to map the CoreNet hierarchy into the Princeton WordNet hierarchy. The scope of the mapping encompassed all of 2,937 semantic categories, which were successfully mapped to WordNet synsets with synonymy, hypernymy, and hyponymy relations.

Although Kang et al. (2010) mapped the almost all of the semantic categories of CoreNet, word senses of CoreNet is not in the scope of the mapping, and still remains out of mapping; this leads to the fact that, from the perspective of NLP application, WordNet operations such as path similarity hard to be applied to the word senses, which are the majority part (96%) of CoreNet, because there is no mapping for the word senses, and, moreover, lexical relations between word senses and mapped part (semantic categories) of CoreNet are totally unknown.

To overcome this limitation, it can be an option that human annotators manually label WordNet synsets to all of the word senses of CoreNet with appropriate lexical relations; however, it requires the excessive cost of human labors.

By the motivation from these facts, in this paper, we introduce an automatic mapping approach that automatically maps the unmapped word senses of CoreNet into WordNet hierarchy to boost bridging the gap between CoreNet and WordNet.

Our contributions are as follows:

(1) We present a wordnet mapping approach that automatically maps word senses in wordnets of different languages, especially Korean and English, using novel semantic features between wordnet hierarchies.

(2) We present a new language resource that contains mappings between CoreNet word senses and WordNet synsets with synonymy relation. To the best of our knowledge, it is the first attempt to map CoreNet word senses into WordNet hierarchy.

In the following sections, we describe the problem to be dealt in this paper and our approach much in detail.

2. Problem Statement

Before mapping CoreNet word senses into WordNet synsets, synset candidates for each CoreNet word sense are selected by the following list of actions.

(1) Given a word sense of CoreNet, the word sense is translated into N English words by bilingual dic-

tionaries; the translation is done based on the exact matching of lemma and part-of-speech.

(2) M synsets are selected as synset candidates for the given word sense of CoreNet if lemma and part-of-speech are exactly matched with one of the N translated English words of the given word sense of CoreNet.

An example of the synset candidate selection is shown in Figure 2.

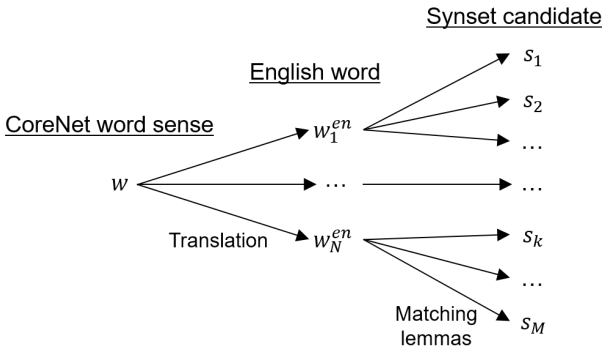


Figure 2: An example of synset candidates for a given word sense of CoreNet

We used 70 domain-specific bilingual dictionaries, Sejong electronic dictionary[1], CoreNet to WordNet mapping (Kang et al., 2010), Naver English dictionary[2], and Google translation[3]. Cumulative coverage of the dictionaries is shown in Table 1.

Table 1: Cumulative coverage of bilingual dictionaries to CoreNet word senses: *Domain* denotes 70 domain-specific dictionaries, *Sejong* denotes Sejong electronic dictionary, *Mapping* denotes CoreNet to WordNet mapping, *Naver* denotes Naver English dictionary, and *Google* denotes Google translation.

Dictionaries	Coverage
Domain	61.82%
Domain, *Sejong*	65.31%
Domain, *Sejong*, *Mapping*	65.35%
Domain, *Sejong*, *Mapping*, *Naver*	72.02%
Domain, *Sejong*, *Mapping*, *Naver*, *Google*	100.0%

After selecting synset candidates, the problem to be dealt in this paper can be translated into word sense disambiguation problem that is to select synonymous synsets from synset candidates for each word sense of CoreNet.
Our approach solves this problem by supervised classification with semantic features of wordnet hierarchies, which is described in the following section in detail.

[1]The Sejong electronic dictionary has been developed by several Korean linguistic researchers, funded by Ministry of Culture and Tourism, Republic of Korea. (http://www.sejong.or.kr)

[2]http://dic.naver.com

[3]https://translate.google.com

3. Mapping Approach

3.1. Semantic Feature Extraction

For a given CoreNet word sense and its synset candidates, three different scores are measured as a feature of semantic similarity between the given CoreNet word sense and its synset candidates.

Vertical similarity is to measure a vertical similarity between hierarchies of CoreNet word sense and its synset candidate; an example is shown in Figure 3.

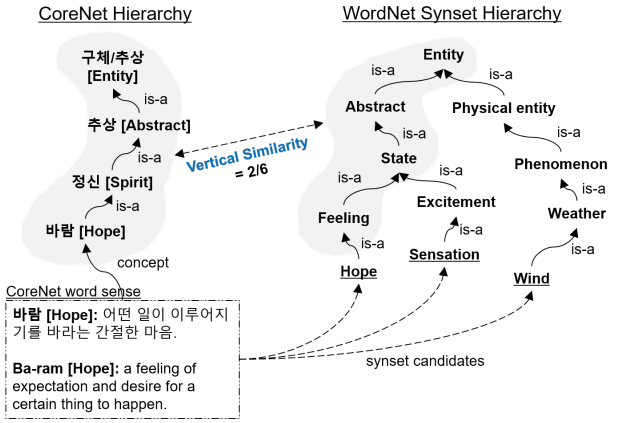


Figure 3: An example of vertical similarity between CoreNet word sense 'Ba-ram' and its synset candidates

The basic idea is that the vertical similarity increases as much as CoreNet word sense and its synset candidate share common ancestors on their hierarchies.
More precisely, the vertical similarity is a translation-based Jaccard similarity between a set of ancestral semantic categories of a given CoreNet word sense and a set of ancestral hypernym and holonym synsets of a synset candidate; the following formula explains the idea:

$$VertSim(w, s) = JaccardSim(Anc_{CN}(w), Anc_{WN}(s))$$

where w denotes a CoreNet word sense, s denotes a synset candidate for w, $Anc_{CN}(w)$ denotes a set of ancestral semantic categories of w, and $Anc_{WN}(s)$ denotes a set of ancestral hypernym and holonym synsets of s.

Horizontal similarity is to measure a horizontal similarity between hierarchies of CoreNet word sense and its synset candidates; an example is shown in Figure 4.
The basic idea is that the horizontal similarity increases as much as CoreNet word sense and its synset candidate share common siblings on their hierarchies.
More precisely, the horizontal similarity is a translation-based Jaccard similarity between a set of sibling word senses of a given CoreNet word sense and a set of sibling synsets of a synset candidate; the following formula explains the idea:

$$HoriSim(w, s) = JaccardSim(Sib_{CN}(w), Sib_{WN}(s))$$

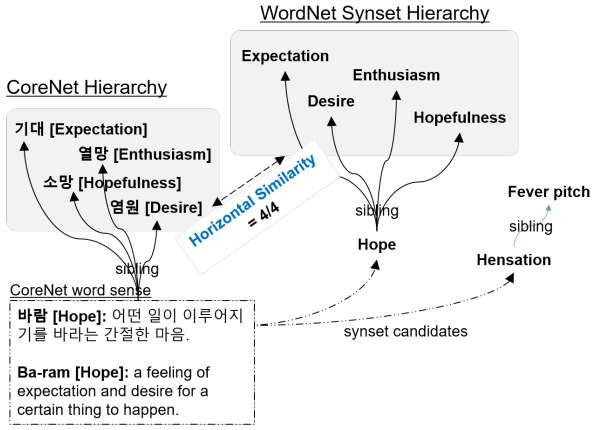


Figure 4: An example of horizontal similarity between CoreNet word sense 'Ba-ram' and its synset candidates

where w denotes a CoreNet word sense, s denotes a synset candidate for w, $Sib_{CN}(w)$ denotes a set of sibling word senses of w, and $Sib_{WN}(s)$ denotes a set of sibling synsets of s.

Conceptual word coverage is to measure a conceptual similarity between CoreNet word sense and its synset candidate based on their conceptual words contained in semantic categories, definition statements, and example sentences.

More precisely, the conceptual word coverage is the measurement of how many words contained in names of semantic categories for a given CoreNet word sense are covered by words contained in definition statements and example sentences of a synset candidate, based on translation; the following formula explains the idea:

$$ConceptCover(w,s) = \frac{|\{w'|w' \in Cwords(w) \cap \{Dwords(s) \cup Ewords(s)\}\}|}{|\{w'|w' \in Cwords(w))\}|}$$

where w denotes a CoreNet word sense, s denotes a synset candidate for w, $Cwords(w)$ denotes a set of words contained in names of w's semantic categories, $Dwords(w)$ denotes a set of words contained in definition statements of s, and $Ewords(w)$ denotes a set of words contained in example sentences of s.

3.2. Basic Feature Extraction

The above-mentioned three semantic features are mainly focused on information about a semantic relationship among word senses, not on word sense itself. To supplement features describing a word sense itself and support the above-mentioned three semantic features for a better performance in a classification task, the basic features of word senses, part-of-speech and semantic categories of word senses, are also used as a feature for training.

3.3. Mapping by Decision Tree Classifier

Given CoreNet word senses and their synset candidates with five different features, our goal is to combine the five features to classify synset candidates as linking or discarding.

The combination of the features is performed by a decision tree classifier which shows the best performance among other different classifiers in our experiments described in the following section.

To link CoreNet word senses into WordNet synsets, there are two phases for training a decision tree classifier (training phase) and linking/discarding synset candidates by the trained classifier (mapping phase).

In the training phase shown in Figure 5, a decision tree classifier is trained on the five features extracted from CoreNet word sense w and synset candidate s contained in manually labeled data.

The manually labeled data is built on the samples from all CoreNet word senses and their Top-2 synset candidates where Top-2 means only two synset candidates are selected from the front of the candidate list sorted by linear summation score of vertical similarity, horizontal similarity, and conceptual word coverage in a descending order. The selected Top-2 synset candidates are labeled as linking or discarding.

The reason why we picked only Top-2 synset candidates for each CoreNet word sense is to avoid imbalance of training and test datasets. If negative examples in the datasets overwhelm positive examples, the precision of classification results would be dropped rapidly by enormous false positives; it is showed in Table 2 that precision and coverage are dropped by increasing the ratio of negative examples to positive examples. There are also reports that standard classifiers such as decision trees give sub-optimal classification results when trained on imbalanced datasets (Lane et al., 2012; Haixiang et al., 2017).

Table 2: Dropping of precision and coverage by increasing the ratio of negative examples to positive examples

negative # / positive #	Precision	Coverage
0.5	0.9121	0.9347
1.0	0.908	0.9258
2.0	0.8789	0.9094
3.0	0.8612	0.8992
4.0	0.8489	0.8934
5.0	0.839	0.8846
6.0	0.8279	0.8821
7.0	0.8113	0.8812
8.0	0.8098	0.8783
9.0	0.7911	0.8752

In the mapping phase shown in Figure 6, a trained model of a decision tree classifier is applied to all the pairs of CoreNet word sense w and its Top-2 synset candidates s to classify as linking or discarding. As a result of classification, synset candidates classified as linking are finally mapped to the corresponding CoreNet word sense as synonymy relation.

4. Evaluation

In this section, we evaluate the performance of each of the five features as well as their combination.

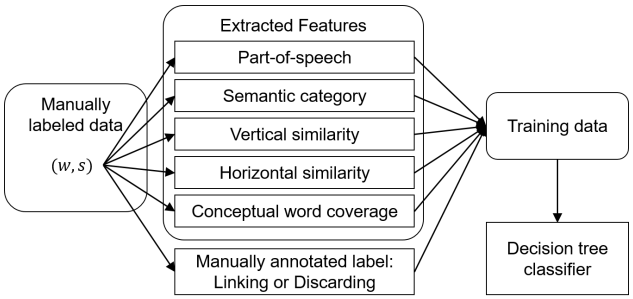


Figure 5: Training phase

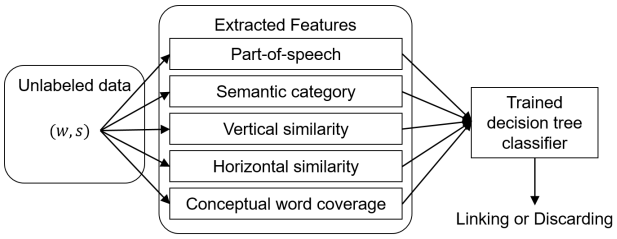


Figure 6: Mapping phase

For evaluation, we use the manually labeled 6,041 CoreNet word senses with 8,655 positive links to synonymous synsets and 2,700 negative links to nonsynonymous synsets.

In the evaluation, we use the two measurements; the one is *precision* defined as the proportion of correctly linked synonymous synsets over all of linking results, and the other is *coverage* defined as the proportion of CoreNet word senses linked to synonymous synsets over all CoreNet word senses to be linked.

All the performance scores are evaluated by 10-fold cross-validation with 90% of labeled data for training and the remaining 10% of labeled data for testing.

The performance scores of decision tree classifiers trained on each feature and combination of all the features are shown in Table 3. 'Random' in the table classifies synset candidates as linking or discarding in a random manner. In summary, the performance of each feature is not good enough, but, when they are combined, the performance is fairly improved up to 91.2% with 99% confidence level.

Table 3: Performance of each feature and the combination

	Precision	Coverage
Random	0.7627	0.7235
Part-of-speech	0.7613	1.0
Semantic category	0.7903	0.8425
VertSim	0.8308	0.9388
HoriSim	0.8107	0.9408
ConceptCover	0.7585	1.0
Combination	0.9121	0.9347

In Table 4, the performance scores of five different classifiers are shown. The classifiers are trained on the combination of all the features. Although the decision tree classifier shows the relatively low coverage, it achieves the best performance of 91.2% precision with 99% confidence level.

Table 4: Performance of different classification models

	Precision	Coverage
Logistic regression	0.8099	0.9815
Naive Bayes	0.8085	0.9321
Decision tree	0.9121	0.9347
SVM	0.7669	0.9983
Multilayer perceptron	0.8315	0.9668

By using the decision tree classifier trained on the combination of all the features, we classified all CoreNet word senses and obtained the mappings between 38,028 CoreNet word senses and their synonymous WordNet synsets. In other words, we constructed a Korean wordnet composed of 38,028 Korean word senses (33,956 nouns, 3,617 verbs, 355 adjectives) with the precision of 91.2% (± 1.14 with 99% confidence level).

5. Related Work

(Lee et al., 2000) introduced the automatic mapping between Korean word senses in bilingual dictionaries and synsets in Princeton WordNet by word sense disambiguation. They reported that 21,654 Korean word senses are mapped to WordNet synset with the precision of 93.59% by decision tree learning on six heuristic features.

In other languages, especially Persian, many works tried to map word senses in bilingual dictionaries to synsets in Princeton WordNet in a similar way (Dehkharghani and Shamsfard, 2009; Mousavi and Faili, 2017).

The above-mentioned works have a common point that they target to map word senses in bilingual dictionaries that are not organized in a semantic network. Inevitably, the features used in their approaches lack the use of hierarchical features in their own languages.

The difference of our work from them is that semantic features introduced in this paper fully utilize hierarchical features of both source language (Korean) and target language (English).

6. Conclusion

This paper has explored an automatic mapping of wordnets, especially CoreNet and Princeton WordNet, by supervised classification with novel semantic features between wordnet hierarchies.

The experiments showed that the combination of all the features introduced in this paper achieves the better performance than each of individual features, and a decision tree classifier is the best choice for performing the combination of all the features.

Our approach is not restricted to CoreNet and Princeton WordNet, but it can be applied on any wordnets with traditional wordnet structures whose word senses are organized in the same lexical relations and have definition statements and example sentences.

After applying our mapping approach on all the CoreNet word senses, we obtained the new synonym mapping between 38,028 word senses of CoreNet and corresponding WordNet synsets. A series of experiments showed that the accuracy of mapping is over 90%.

7. Acknowledgement

This research was financially supported by the Ministry of Trade, Industry and Energy(MOTIE) and Korea Institute for Advancement of Technology(KIAT) through the International Cooperative R&D program.

8. Bibliographical References

Choi, K.-S., Bae, H.-S., Kang, W., Lee, J., Kim, E., Kim, H., Kim, D., Song, Y., and Shin, H. (2004). Korean-chinese-japanese multilingual wordnet with shared semantic hierarchy. In *LREC*.

Dehkharghani, R. and Shamsfard, M. (2009). Mapping persian words to wordnet synsets. *International Journal of Interactive Multimedia and Artificial Intelligence*, 1(2).

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., and Bing, G. (2017). Learning from class-imbalanced data: review of methods and applications. *Expert Systems with Applications*, 73:220–239.

Ikehara, S., Miyazaki, M., Shirai, S., Yokoo, A., Nakaiwa, H., Ogura, K., Ooyama, Y., and Hayashi, Y. (1997). Goi-taikei-a japanese lexicon.

Kang, I.-S., Kang, S.-J., Nam, S.-J., and Choi, K.-S. (2010). Linking corenet to wordnet-some aspect and interim consideration. In *Proceedings of the 5th Global WordNet Conference*, pages 239–242.

Lane, P. C., Clarke, D., and Hender, P. (2012). On developing robust models for favourability analysis: Model choice, feature sets and imbalanced data. *Decision Support Systems*, 53(4):712–718.

Lee, C., Lee, G., and Yun, S. J. (2000). Automatic wordnet mapping using word sense disambiguation. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 142–147. Association for Computational Linguistics.

Mousavi, Z. and Faili, H. (2017). Persian wordnet construction using supervised learning. *arXiv preprint arXiv:1704.03223*.

The New Propbank: Aligning Propbank with AMR through POS Unification

Tim O'Gorman[1], Sameer Pradhan[23], Julia Bonn[1], Katie Conger[1], James Gung[1,4], Martha Palmer[1]
[1]University of Colorado [2]cemantix.org [3]Vassar College [4]IPSoft
ogormant@colorado.edu

Abstract

We present a corpus which converts the sense labels of existing Propbank resources to a new unified format which is more compatible with AMR and more robust to sparsity. This adopts an innovation of the Abstract Meaning Representation project (Banarescu et al., 2013) in which one abstracts away from different, related parts of speech, so that related forms such as "insert" and "insertion" could be represented by the same roleset and use the same semantic roles. We note that this conversion also serves to make the different English Propbank corpora released over the years consistent with each other, so that one might train and evaluate systems upon that larger combined data. We present analysis of some appealing characteristics of this final dataset, and present preliminary results of training and evaluating SRL systems on this combined set, to spur usage of this challenging new dataset.

Keywords: Propbank,SRL, Semantic Roles, Corpora

1. Introduction

We introduce the conversion of all existing Propbank data — constituting more than half a million predicate instances in English — into a format in which etymologically related senses from different parts are speech are merged, making that data compatible with the predicate senses used for Abstract Meaning Representation (Banarescu et al., 2013). This constitutes a large set of data made consistent to use the same frames and conventions, both increasing the amount of training data available for Propbank SRL and also providing a large corpus of semantic role labeling whose rolesets and numbered arguments match those of the Abstract Meaning Representation data (while containing over twice as many predicate instances).

We describe the combination of automatic and manual methods used in converting these corpora, provide some analysis to characterize the resulting corpora, and present preliminary SRL results against the test sets presented here, as a baseline for future evaluation of SRL. We suggest that evaluating against the combined test sets (OntoNotes, English Web Treebank and BOLT) can provide a challenging test set that could encourage the community to build SRL systems with a greater coverage over nominal, adjectival and light verb data, and with robustness to a range of difficult genres.

1.1. Motivations for Unifying Parts of Speech

Propbank (Kingsbury and Palmer, 2002) is a paradigm for the development of semantic role labeling corpora, designed for large-scale annotation. It focuses upon annotation of coarse-grained senses ("rolesets") which provide predicate-specific definitions of numbered arguments (ARG0, ARG1, etc.) to represent semantic roles. By using coarse-grained sense labels and these predicate-specific arguments (following the "individual thematic roles" of Dowty (1991)), Propbank approaches can achieve a high inter-annotator agreement rate, and the methodology has been adapted to Chinese (Palmer et al., 2005), Korean (Palmer et al., 2006), Hindi/Urdu (Bhatt et al., 2009), Finnish (Haverinen et al., 2013), Turkish (Sahin, 2016) and Brazilian Portuguese (Duran and Aluísio, 2011).

Such Propbank semantic role labels are generally annotated by labeling individual phrases within a constituency parse (which can then be converted into surface forms (Carreras and Màrquez, 2005; Pradhan et al., 2011) or dependency parses (Surdeanu et al., 2008)), but AMR annotation is done by directly building a semantic graph for a sentence – utilizing Propbank senses and numbered arguments – without explicit linking of that graph to phrases within a sentence.

While Propbank 1.0 (Kingsbury and Palmer, 2002) annotated only verbal predicates, it was later expanded to nouns (Hwang et al., 2010) and predicative adjectives (Bonial et al., 2017), creating new Propbank senses (called "rolesets") for those nouns and adjectives. Parallel work in the Abstract Meaning Representation project also handled nouns and adjectives, but did so by representing them with etymologically related verbal rolesets – so that a noun such as "insertion" would not have its own rolesets, but would instead be labeled with a verbal sense of "insert". This approach has a range of useful properties in reducing the number of senses with small amounts of training data, and also better conforms to the approach of FrameNet (Baker et al., 1998).

In order to merge these different rolesets into the new "unified" form, Propbank rolesets were given any number of "alias" entries, each of which expresses a different surface form and part of speech. These alias fields are analogous to the Lexical Units of FrameNet (Baker et al., 1998), although this does not result in a Framenet-like lexicon; Propbank rolesets rarely contain more than three aliases, and maintain the same coarse-grained sense distinctions developed in earlier Propbank works. For example, one might look at the different usages of "appeal" in that regard: there is a legal sense of the verb and noun "appeal", a begging sense for the same terms, or an attractiveness sense for which one might use verbal "appeal", nominal "appeal", the adjective "appealing" or the light verb "have appeal". In the pre-unification methodology, each sense of each lemma would receive its own roleset, resulting in seven different rolesets. Instead, these are merged into three rolesets, each with 2–3 aliases.

This approach results in a reduction in sparsity, and reduces

the number of redundant senses which must be added as a lexicon increases. Section 4 will attempt to quantify such gains. Conversion to unified forms was also an appropriate context to update a range of prior Propbank annotations in order to make them more compatible – resulting in a larger collective landscape of SRL resources for English.

2. Methodology of Corpus Conversion

2.1. Conversion of Lexicon

It is intuitive to a casual speaker of English that "insert" and "insertion", or "appeal" and "appealing", are related lemmas. We focus upon that simple, coarse-grained level of etymological relatedness, focusing upon clusters of lemmas that would be verbalized into the same verbal lemma. Thus, "appealing" is clustered with "appeal" we do not dive into distantly related terms with the same root, such as "appellation" or "compel." After determining which lemmas were to be treated as related, the challenge was (a) to get an alignment between those rolesets — discerning which verbal senses corresponded with which nominal senses — and (b) to get an alignment between the numbered arguments in each aligned pair of rolesets.

The nominal and adjectival rolesets were developed with an awareness of both the related verbal senses and related senses in Nombank (Meyers et al., 2004). When nominal or adjectival forms had the same meaning as a verbal form, care was therefore taken to maintain consistency in between their sense descriptions and numbered argument descriptions. These allowed many direct, automatic mappings to be done.

However, senses and numbered arguments did not always have such identical descriptions. To align senses without clear matches, we started with automatic pre-annotations using existing information, as rolesets were sometimes mapped to corresponding FrameNet classes (Palmer, 2009), but then overviewed all such alignments manually, using experienced Propbank frame builders. For aligning numbered arguments between those aligned senses, a range of resources exist that help one generalize beyond predicate-specific numbered arguments of Propbank, such as the Propbank function tags (labels such as LOC, TMP, PAG (more proto-agentive core role) and PPT (more proto-patientive core role); cf Bonial et al. (2014)), the number of the numbered argument itself, which are designed to capture general tendencies (Kingsbury and Palmer, 2002), and mappings to Verbnet or FrameNet semantic role types (Palmer, 2009). As with roleset alignments, these were then manually checked by expert framers to confirm each mapping.

This resulted not simply in a new lexicon, but in mappings from the older lexicon to the new one, with manually crafted labels regarding which mappings were deterministic, and which mappings might require manual checking. This resulted in the merging of 13,460 rolesets delineated by part of speech into 10,183 unified rolesets, 57 of which were determined by framers to require manual instance-by-instance sense disambiguation. The alignment of numbered arguments converted 34,469 different arguments down to a set of 25,452 unified numbered arguments, only 388 of which required manual instance-by-instance retrofitting.

2.2. Conversion of Annotated Data

The conversion of the Propbank lexicon resulted in a set of direct mappings between senses and numbered arguments, a small percentage of which required manual disambiguation. All senses which were labeled as ambiguous were double-annotated to revise them. In addition, whenever a numbered argument was labeled as ambiguous between numbered arguments in the new frames, every instance with that numbered argument was also selected for retrofitting. The manual retrofitting covered 12,000 instances in Propbank across all corpora (roughly 2% of all predicates), each of which was double-annotated and adjudicated. Subsequent data generated after that conversion was annotated directly using these unified frames.

2.3. Revision of Formatting

In the process of this conversion, other inconsistencies between different releases of Propbank have also been resolved in order to make the data more consistent in format and behavior. In addition to minor consistency decisions (such as replacement of a "ARGM-PNC" role with "ARGM-PRP"), we made the treatment of control and relative clause chains (appearing as LINK-PRO and LINK-SLC) consistent, using postprocessing tools in ClearNLP (Choi, 2012). A set of scripts used convert to that data to the bracketing over surface forms used in evaluations (Pradhan et al., 2011) has been revised and corrected, to improve the consistency of discontinuous material ("R-" and "C-" prefixed arguments in the CoNLL-2012 data).

2.4. Relationship of this Unified Data to AMR

The Abstract Meaning Representation project annotates predicate argument structures using the Propbank lexicon, supplemented with a small set of AMR-specific rolesets, ending with *-91*, for specific semantic functions and the reification of semantic roles, such as INCLUDE-91 for set operations or HAVE-ORG-ROLE-91 for organizational membership. Other than those AMR rolesets ending in *-91*, the rolesets used within AMR are a subset of the rolesets used in Propbank, adopting nearly every verbal form in Propbank and most nominal and adjectival senses. The most common reason why a verbal roleset is not in AMR is because AMR deletes "semantically light" predicates, such as copular "be" or auxiliary "have". As the use of existing SRL systems has been shown to help AMR parsing (Wang et al., 2015), we expect that the introduction of a larger SRL dataset with closer alignment to the AMR lexicon should increase that utility. The AMR SUBSET line in Table 1 illustrates the size of these corpora when limited to only the rolesets used in AMR; while there is a large drop in verbal senses (notably due to the omission of semantically light predicates), it still remains a very large corpus. Work is ongoing adding more of the Propbank nominal and adjectival rolesets to AMR.

2.5. Expansion of multi-word predicate coverage

More recent work has expanded coverage of the Propbank lexicon to encompass multi-word predicates as well, such as *take with a grain of salt*, *cut slack*, or *jump on bandwagon*. Propbank has long annotated certain classes of

multi-word predicates such as verb-particle constructions and light verbs (Bonial et al., 2014), and is now expanded to arbitrarily structured semi-fixed expressions. We added coverage to many of the most high-frequency multi-word predicates in the corpus, and created lexical entries for each multi-word predicate. The important contribution is not simply the detection of these MWP elements (which can also be found in larger resources such as PARSEME (Savary et al., 2017)), but the annotation of semantic roles for each MWP. For example, something like "jump on the bandwagon" would be as follows:

- **jump-on-bandwagon.09**: *join an activity or group because of its popularity*
  - Arg0: person jumping on the bandwagon
  - Arg1: popular thing joined
  - Arg2: action done which gets one on the bandwagon

This is a step forward in not simply detecting these MWPs, but being able to represent them in structured semantic representations such as AMR.

3. Larger Landscape of Propbank Resources

The OntoNotes corpus (Hovy et al., 2006), most recently released as part of the Conll-2012 (Pradhan et al., 2011), was developed during the DARPA-GALE annotation project covering a wide range of domains, and has been the largest resource for training and evaluating semantic role labeling systems. However, the range of other corpora that have been annotated with Propbank roles since OntoNotes – most notably, the English Web Treebank and the BOLT corpora – collectively constitute an amount of additional predicate annotations the same size as OntoNotes itself. The English Web Treebank encompasses a range of genres of the web, such as reviews and emails (Bies et al., 2012), and the BOLT datasets encompass informal corpora of English discussion forum data, SMS text, and translations of conversational data (Garland et al., 2012; Song et al., 2014). All of these resources are either currently released or in the process of being released, with stand-off SRL annotations available at propbank.github.io .

We suggest that the combination of the test sets of the three major corpora provides a more interesting and challenging dataset against which to evaluate a semantic role labeling system. This is due to both the challenging informal domains (such as SMS messages and discussion forum posts) as well as to the increase in the coverage of nouns and adjectives in the data. Table 1 illustrates the size of these corpora, broken down by the parts of speech seen. Due to annotations done at the end of the OntoNotes data collection phase, additional rolesets were also added to the OntoNotes corpus release.

Propbank data also exists for related domains. The SHARP and THYME clinical data (Albright et al., 2013) encompass nearly a million words of clinical text; there are also annotations in the same frames for the LORELEI English core data (Strassel and Tracey, 2016), image captions of the Flickr 8k corpus (Hodosh et al., 2013), MASC data (Ide et al., 2008), and Earth Science data affiliated with the ClearEarth project (Duerr et al., 2016). These additional corpora – as noted in Table 2 below – illustrate the range of domain-specific annotations of Propbank data which might be utilized for semantic role labeling in specific domains.

	verbs	nouns (light v.)	adjectives
OntoNotes (ON)	349,352	40,163 (2,215)	750
EWT	44,736	9,453 (732)	3,305
BOLT	132,642	18,839 (1973)	10,957
ON+EWT+BOLT	526,730	68,455 (4920)	15,012
in AMR Subset	349,783	63,585 (4714)	10,121
Conll-2012	319,239	20,305	0

Table 1: Core Corpora Annotated with Propbank rolesets for general English. Light verbs are annotated using nominal frames (Hwang et al. 2010) and therefore included in those counts

	verbs	nouns (light v.)	adjectives
LORELEI	18,871	4,089 (196)	780
MASC	14,150	70 (3)	0
flickr-5k	5,897	551 (91)	51
earth science	10,070	5713 (8)	468
clinical – SHARP	27,667	15,807 (22)	0
clinical – THYME	49,649	17,906 (89)	756
All current	653,034	112,591 (5,329)	17,067

Table 2: Additional corpora annotated with Propbank rolesets

4. Data Analysis

4.1. Effect of Unification on Sense Sparsity

The largest anticipated advantage to unifying across parts of speech is to reduce the number of very low frequency rolesets, which are therefore both hard to detect and whose numbered arguments are difficult to learn. Using the running example of the "appeal" senses: while three verbal senses of "appeal" each have 20+ examples in training data, the adjectival sense of "appealing" and three nouns of "appeal" all have fewer than six examples, making all of them problematic for learning. One way of quantifying this is to look at a new English dataset (we use the LORELEI corpus of disaster-related newswire texts) and to measure how many predicate instances have a sufficient number of examples in prior Propbank corpora. Figure 1 illustrates this over a range of different thresholds for sufficiency. One may see that there is a reduction of these out-of-vocabulary and low-frequency senses due to the added nominal rolesets now in OntoNotes, further reductions of sparsity when one looks at the combined set of OntoNotes, English Web Treebank, and BOLT, and further reductions of sparsity with the unification of rolesets across parts of speech.

	ON (CoNLL-2012)		ON (Unified)		BOLT		Google	
	Valid	Test	Valid	Test	Valid	Test	Valid	Test
Our System	78.7	79.0	78.2	78.3	65.2	66.0.	72.0	72.4
Comparable Past Results								
Täckström et al. (2015)	79.1	79.4	-	-	-	-	-	-
FitzGerald et al. (2015) (Single)	79.2	79.6	-	-	-	-	-	-
FitzGerald et al. (2015) (POE)	79.7	80.1	-	-	-	-	-	-
Zhou and Xu (2015)	81.1	81.3	-	-	-	-	-	-

Table 3: Preliminary results on the part-of-speech *unified* corpora along with results reported on existing partition by other researchers.

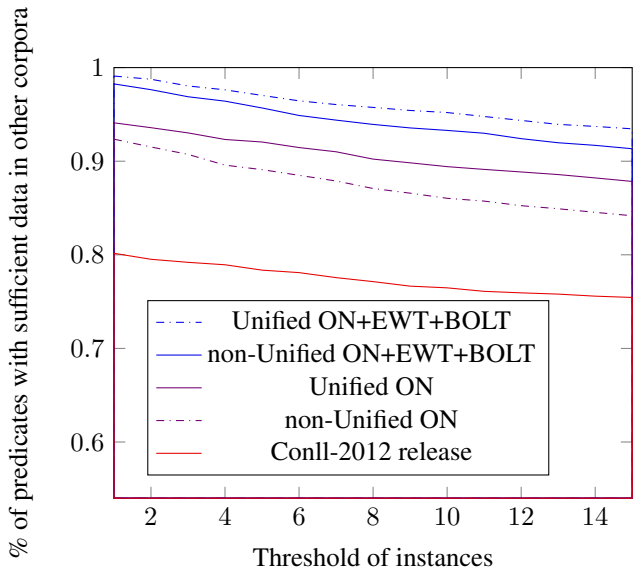


Figure 1: Percent of predicate instances in LORELEI with N or more training instances

4.2. Nature of the Expansion of Nominal Coverage

The coverage over nominals is dramatically expanded in this release of Propbank, having 3934 rolesets with nominal *aliases*, focused upon nouns that correspond with predicates or events, and roughly as many as NomBank corpus(Meyers et al., 2004). This differs from two other core resources for semantic roles of nominals, FrameNet (Baker et al., 1998) and NomBank (Meyers et al., 2004), which often annotated non-eventive nominals. Because of this, while Propbank has coverage over only 20% of nominal lexical units in FrameNet, and 30% of those of NomBank (Meyers et al., 2004), that coverage encompasses most of the nouns which assert an event or dynamic situation. Table 4 illustrates how the coverage differs between each resource, by illustrating the kinds of nominal predicates unique to each annotation project. Propbank rolesets which overlap with FrameNet and NomBank (the left column) illustrate prototypical nominalizations. In contrast, nouns not represented in Propbank but captured in FrameNet or NomBank show their coverage over more traditional entities, objects and relational nouns.

In all resources	FN only	NomBank only
permission	business card	pirate
assistance	knife	mound
prohibition	rotunda	normalcy
invasion	pattern	ire
kidnapping	raincoat	glamour

Table 4: Random samples of Propbank nominal rolesets overlapping with Framenet and NomBank (left), and those unique to FrameNet (center), or to NomBank (right), illustrating that most eventive nouns are in that intersection

5. Preliminary Results

5.1. Semantic Role Labeler

Our SRL system (Gung and Pradhan, 2018) uses a deep neural network model which does not include explicit syntactic information. We closely follow Zhou and Xu (2015), treating SRL as an IOB tagging problem and using deep bidirectional LSTMs with a linear chain conditional random field (Lafferty et al., 2001) loss function.

Long-short term memory networks (LSTMs) are a form of recurrent neural network (RNN) that has been successfully applied to many NLP tasks. Sequential inputs are often processed using pairs of RNNs, with one RNN processing from the first to last element and the other RNN processing from the last element to the first, concatenating outputs for each element (Graves et al., 2013). In our approach, and that of Zhou and Xu (2015), the result of the forward pass is used as input to the backward pass, enabling repeated stacking of these layers to form a deep topology. Instead of scoring each label locally, the addition of a CRF loss function allows for globally normalized scoring of all possible sequences of labels, maximizing the sequence-level log-likelihood (Collobert et al., 2011). This approach has been shown to improve performance on a variety of tasks (Huang et al., 2015; Ma and Hovy, 2016; Lample et al., 2016).

Following standard practices for applying neural architectures to NLP tasks, we initialize our network with word embeddings trained on orders of magnitude more data than is available for our task (SRL). Specifically, we use publicly available GloVe 100-dimensional vectors trained on 6 billion words from Wikipedia and Gigaword (Pennington et al., 2014). These embeddings are updated during training as network parameters along with a single out-of-vocabulary (OOV) vector, which is randomly initial-

ized. We simplify the features used in the original model of (Zhou and Xu, 2015), using only the vector associated with the current word as well as the distance from the current word to the predicate. The distance feature uses a trainable lookup table to map each discrete distance to a low-dimensional representation.

To improve the handling of OOV words, such as names and numbers not found in the original word vectors, we also use character-level vector representations produced using a convolutional neural network. Recent work has demonstrated the effectiveness of using neural networks to extract character-level (morphological) features (Chiu and Nichols, 2015; Lample et al., 2016; dos Santos et al., 2015; dos Santos and Zadrozny, 2014; Kim et al., 2015; Ma and Hovy, 2016). In these approaches, characters are assigned fixed-dimensional embeddings, which are composed into a single fixed-length vector using a neural network-based reduction function. We follow the approach described in Ma and Hovy (2016), using a convolutional neural network with max-over-time pooling. The resulting character-based representations are concatenated with each word and distance vector to form the input to the deep bidirectional LSTM. We train the full neural network end-to-end using Adam (Kingma and Ba, 2014).

5.2. Preliminary Results on the English Unified Set

In this section we will report preliminary results on this updated corpus using the aforementioned semantic role labeler model. For these experiments we used the OntoNotes training partition to train the model.

Table 3 summarizes the performance of our system on the validation and test partitions of the following three subcorpora— the CoNLL-2012 subset of OntoNotes, the revised OntoNotes with additional predicates and unification across the part-of-speeches, as well as the BOLT and Google corpora which will be released in their unified format. Further details of the partitions, and further analysis and genre-wise breakdowns of results, will be provided in the extended form of the paper.

6. Discussion

We've outlined the methods for converting Propbank to a unified form, and the advantages provided by that unified form and by the larger size of the Propbank corpora now available. We suggest that testing against the combination of OntoNotes, English Web Treebank and BOLT corpora can provide a more challenging SRL evaluation, requiring systems to better handle challenging web domains and non-verbal predicates.

7. Acknowledgements

This work is supported by Defense Advanced Research Projects Agency (DARPA) under DARPA HR001117S0026 - AIDA, and BOLT- HR0011-11-C-0145, DEFT-FA-8750-13-2-0045 and DARPA Lorelei via subcontracts from LDC. We thank Google in supporting the annotation of the English Web Treebank, and thank the LDC for cooperation in distributing the source material.

8. Bibliographical References

Albright, D., Lanfranchi, A., Fredriksen, A., Styler IV, W. F., Warner, C., Hwang, J. D., Choi, J. D., Dligach, D., Nielsen, R. D., Martin, J., et al. (2013). Towards comprehensive syntactic and semantic annotations of the clinical narrative. *Journal of the American Medical Informatics Association*, 20(5):922–930.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract meaning representation for sembanking.

Bonial, C., Bonn, J., Conger, K., Hwang, J., and Palmer, M. (2014). PropBank: Semantics of New Predicate Types. pages 3013–3019.

Bonial, C., Conger, K., Hwang, J. D., Mansouri, A., Aseri, Y., Bonn, J., OGorman, T., and Palmer, M. (2017). Current directions in english and arabic propbank. In *Handbook of Linguistic Annotation*, pages 737–769. Springer.

Carreras, X. and Màrquez, L. (2005). Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 152–164. Association for Computational Linguistics.

Chiu, J. P. and Nichols, E. (2015). Named entity recognition with bidirectional lstm-cnns. *arXiv preprint arXiv:1511.08308*.

Choi, J. D. (2012). *Optimization of natural language processing components for robustness and scalability*. Ph.D. thesis, University of Colorado at Boulder.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

dos Santos, C. N. and Zadrozny, B. (2014). Learning character-level representations for part-of-speech tagging. In *Proceedings of ICML-14*.

dos Santos, C., Guimaraes, V., Niterói, R., and de Janeiro, R. (2015). Boosting named entity recognition with neural character embeddings. In *Proceedings of NEWS 2015 The Fifth Named Entities Workshop*, page 25.

Duerr, R., Thessen, A., Jenkins, C., Palmer, M., Myers, S., and Ramdeen, S. (2016). The clearearth project: Preliminary findings from experiments in applying the cleartk nlp pipeline and annotation tools developed for biomedicine to the earth sciences. In *AGU Fall Meeting Abstracts*.

FitzGerald, N., Täckström, O., Ganchev, K., and Das, D. (2015). Semantic role labeling with neural network factors. In *Proceedings of EMNLP*, pages 960–970.

Graves, A., Jaitly, N., and Mohamed, A.-r. (2013). Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE.

Gung, J. and Pradhan, S. (2018). How redundant is syntax for deep semantic role labeling? Ms.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Hwang, J. D., Bhatia, A., Bonial, C., Mansouri, A., Vaidya,

A., Xue, N., and Palmer, M. (2010). Propbank annotation of multilingual light verb constructions. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 82–90. Association for Computational Linguistics.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2015). Character-aware neural language models. *arXiv preprint arXiv:1508.06615*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Palmer, M. (2009). Semlink: Linking propbank, verbnet and framenet. In *Proceedings of the generative lexicon conference*, pages 9–15. Pisa Italy.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., et al. (2017). The parseme shared task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47.

Surdeanu, M., Johansson, R., Meyers, A., Màrquez, L., and Nivre, J. (2008). The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 159–177. Association for Computational Linguistics.

Täckström, O., Ganchev, K., and Das, D. (2015). Efficient inference and structured learning for semantic role labeling. *Transactions of the Association for Computational Linguistics*, 3:29–41.

Wang, C., Xue, N., and Pradhan, S. (2015). Boosting transition-based amr parsing with refined actions and auxiliary analyzers. *Volume 2: Short Papers*, page 857.

Zhou, J. and Xu, W. (2015). End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

9. Language Resource References

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING/ACL-98)*, pages 86–90, Montreal.

Bhatt, R., Narasimhan, B., Palmer, M., Rambow, O., Sharma, D. M., and Xia, F. (2009). A multi-representational and multi-layered treebank for hindi/urdu. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 186–189. Association for Computational Linguistics.

Bies, A., Mott, J., Warner, C., and Kulick, S. (2012). English web treebank. *Linguistic Data Consortium, Philadelphia, PA*.

Duran, M. S. and Aluísio, S. M. (2011). Propbank-br: a brazilian portuguese corpus annotated with semantic role labels. In *Proceedings of the 8th Symposium in Information and Human Language Technology, Cuiabá/MT, Brazil*.

Garland, J., Strassel, S., Ismael, S., Song, Z., and Lee, H. (2012). Linguistic resources for genre-independent language technologies: user-generated content in bolt. In *Workshop Programme*, page 34.

Haverinen, K., Laippala, V., Kohonen, S., Missilä, A., Nyblom, J., Ojala, S., Viljanen, T., Salakoski, T., and Ginter, F. (2013). Towards a dependency-based propbank of general finnish. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013); May 22-24; 2013; Oslo University; Norway. NEALT Proceedings Series 16*, number 085, pages 41–57. Linköping University Electronic Press.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Ide, N., Baker, C., Fellbaum, C., and Fillmore, C. (2008). Masc: The manually annotated sub-corpus of american english. In *In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC*. Citeseer.

Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *LREC*, pages 1989–1993.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The nombank project: An interim report. In *HLT-NAACL 2004 workshop: Frontiers in corpus annotation*, volume 24, page 31.

Palmer, M., Xue, N., Babko-Malaya, O., Chen, J., and Snyder, B. (2005). A parallel proposition bank ii for chinese and english. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 61–67. Association for Computational Linguistics.

Palmer, M., Ryu, S., Choi, J., Yoon, S., and Jeon, Y. (2006).

Korean propbank. *LDC Catalog No.: LDC2006T03 ISBN*, pages 1–58563.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. (2011). Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27.

Sahin, G. (2016). Verb sense annotation for turkish propbank via crowdsourcing. In *Proceedings of 17th international conference on intelligent text processing and computational linguistics. CICLING*.

Song, Z., Strassel, S., Lee, H., Walker, K., Wright, J., Garland, J., Fore, D., Gainor, B., Cabe, P., Thomas, T., et al. (2014). Collecting natural sms and chat conversations in multiple languages: The bolt phase 2 corpus. In *LREC*, pages 1699–1704.

Strassel, S. and Tracey, J. (2016). Lorelei language packs: Data, tools, and resources for technology development in low resource languages. In *LREC*.

The Boarnsterhim Corpus: A Bilingual Frisian-Dutch Panel and Trend Study

Marjoleine Sloos, Eduard Drenth, Wilbert Heeringa
Fryske Akademy, Royal Netherlands Academy of Sciences (KNAW)
Doelestrjitte 8, 8911 DX Ljouwert, The Netherlands
{msloos, edrenth, wheeringa}@fryske-akademy.nl

Abstract
The Boarnsterhim Corpus consists of 250 hours of speech in both West Frisian and Dutch by the same sample of bilingual speakers. The corpus contains original recordings from 1982-1984 and a replication study recorded 35 years later. The data collection spans speech of four generations, and combines panel and trend data. This paper describes the Boarnsterhim Corpus halfway the project which started in 2016 and describes the way it was collected, the annotations, potential use, and the envisaged tools and end-user web application.

Keywords: West Frisian, Dutch, sociolinguistics, language variation and change, bilingualism, phonetics, phonology

1. Background

West Frisian is mostly spoken in the province of Fryslân in the north of the Netherlands. All its speakers are bilingual with Dutch, which is the dominant language. West Frisian is mainly used in informal settings (van Bezooijen 2009:302) but also the most formal, viz. in the provincial parliament. In semi-formal interactions (e.g.shopping, in church), Dutch is usually the preferred language. This dominant position of Dutch traces back to about 1500 when Fryslân lost its political independence and the upper class started to use a mixed Frisian-Dutch language (van Bezooijen 2009:302). Given this long-term contact, it was (and still is) often thought that West Frisian is slowly but steadily converging towards Dutch (e.g. Feitsma 1989).

However, an investigation in the 1980s into phonological change in West Frisian and Dutch of (the same) West Frisian speakers, suggests that, at least at the phonological level, the opposite holds: younger speakers were more likely to keep the phonological rules of the two languages apart; whereas older speakers were more likely to confuse them in either language (van der Kuip 1986, Feitsma et al. 1987, Feitsma 1989, Meekma 1989). This suggests language change, and we are currently investigating whether this change has been continuing over the past thirty-five years in a follow-up study—which is a replication of the original one (see also section 2).

The present contribution describes the unique dataset that underlies these studies. The database is longitudinal, spanning four generations. It combines trend data with a panel study in which the same speakers were recorded in the 1980s and 35 years later. For this sociolinguistic study, unique in a bilingual context, data were collected that have never been made publicly accessible before. Given the special language situation, the exclusive design, and the broad array of linguistic fields which could benefit from these data, we will make the corpus freely accessible.

The remainder of this paper describes the data collection and methodology in section 2. Section 3 describes the embedding in a larger infrastructure and section 4 provides background information about the tool that is used to retrieve lexical frequency. Section 5 discusses previous and ongoing research, and further research opportunities that this database may make possible. Finally, section 6 concludes.

2. Methods and Data

2.1 Data Collection

The Boarnsterhim Corpus (henceforth BHC) consists of two collections. BHC1 was recorded between 1982 and 1984; BHC2 is recorded in 2018-2019. The BHC1 data underlie the above mentioned sociolinguistic studies into variation and change of bilingual Frisian-Dutch speakers (van der Kuip 1986, Feitsma et al. 1987, Feitsma 1989, Meekma 1989). The recordings were made in the municipality of Boarnsterhim in central Fryslân (see Figure 1), because the inhabitants—more than other Frisians—advocated the monolinguistic use of Frisian (Feitsma 1989).[1]

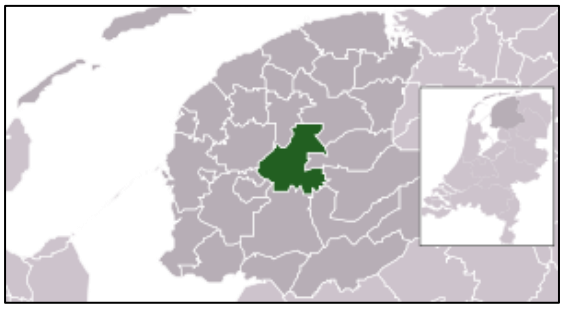


Figure 1: The municipality of Boarnsterhim 1984-2014. The inset represent the Netherlands (*Centraal Bureau voor de Statistiek* 'Statistics Netherlands').

[1] The municipality of Boarnsterhim (Dutch: Boornsterhem) was created through a division boundary alteration in 1984 in which three municipalities were combined into one. The area was 168.58 km^2 and consisted of 18 villages of which Grou was the principal one. It is an area with much water (17.04 km^2) and remained relatively isolated for quite a long time. In 1990, the population consisted of 17,710 people and slightly increased every year. As a result of another division boundary alteration in 2014, Boarnsterhim was again divided and combined with four other municipalities.
(https://nl.wikipedia.org/wiki/Boornsterhem.)

The speakers were recorded twice; once in West Frisian (with a West Frisian native interviewer) and once in Dutch (with a monolingual Dutch interviewer). To enhance informal and spontaneous speech—crucial for Frisian—the recordings took place at the speakers' homes. This lead to a variable amount of noise in the recordings, but overall the quality of the recordings is still acceptable and even valuable for phonetic research.

Within the same family, three speakers were recorded, each of a different generation. The three speakers of a single family were either male or female. The older speakers in BHC1 were born between 1898 and 1917 (65-86 years old at the time of the recording); middle-aged speakers were born between 1926 and 1946 (36-58 years old at the time of the recording); and younger speakers were born between 1952 and 1962 (20-32 years old at the time of the recording) (Feitsma et al. 1987, Feitsma 1989, Meekma 1989). In sum 87 speakers were recorded, from 29 families (two speakers were missing).

The recorded family members had comparable socio-economic status (SES) and were socially divided into three categories based on their levels of education: non-educated farmers, lower educated, or higher educated. The non-educated females were wives and daughter of farmers. Higher educated females of the older generations appeared too difficult to find at the time, so the social stratification in the BHC1 corpus consists of five groups (van der Kuip 1986, Feitsma et al. 1987, Feitsma 1989, Meekma 1989):

- higher-educated males
- lower-educated males
- lower-educated females
- male farmers (non-educated)
- non-educated females

Each recording consists of 20 read sentences, a read story (2-3 minutes), and an interview of about 40 minutes about the speaker's use of West Frisian, language attitude, and daily life activities. The data were originally recorded on TDK-AD (Japan) cassette tapes and digitalized in 2016 with a SONY TC-FX310 cassette deck connected to a PC with a stereo Jack-Tulp cable.

The BHC2 aims to be a replication of the BHC1, with the same number of speakers and same age groups. Since education levels gradually increased and all farmers are (lower or higher) educated nowadays, we adhere to the currently common two-way distinction in education level (and SES in general) in the Netherlands.[2]

The generation of the middle-aged speakers of the BHC1 serves as the oldest generation in the BHC2 and the generation of the youngest speakers of the BHC1 can be identified with the middle-aged speakers in the BHC2. Our aim is to record 50% of the original speakers of the BHC1 and 50% new speakers of these generations, plus an equal number of younger speakers born between 1982 and 1997. We follow the system of three family members of the same gender throughout the data collection. The recordings are made with Tascam DR-44WL recorders.

2.2 Annotation and Data Labelling

The data are manually aligned and annotated in Praat speech processing software (Boersma & Weenink 2017). The textgrids consist of orthographic, lexical, phonological, and phonetic annotations. The orthographic transcriptions are aligned at the phrase level in Standard West Frisian (cf. the *foarkarswurdlist* 'preferred wordlist' 2011) or Standard Dutch (cf. *Het Groene Boekje* 'The Green Booklet' 2017). Dutchisms in Frisian recordings and Frisisms in Dutch recordings are given between square brackets.

Separate tiers are made for the alignment and annotation of words, phonemes, and phonetic or allophonic realizations. A point tier is used to indicate deletion. Extra tiers for specific phonological processes will be added in the future, like a tier for the pronunciation of final -ən. In addition, a tier is used for general comments (see Figure 2).

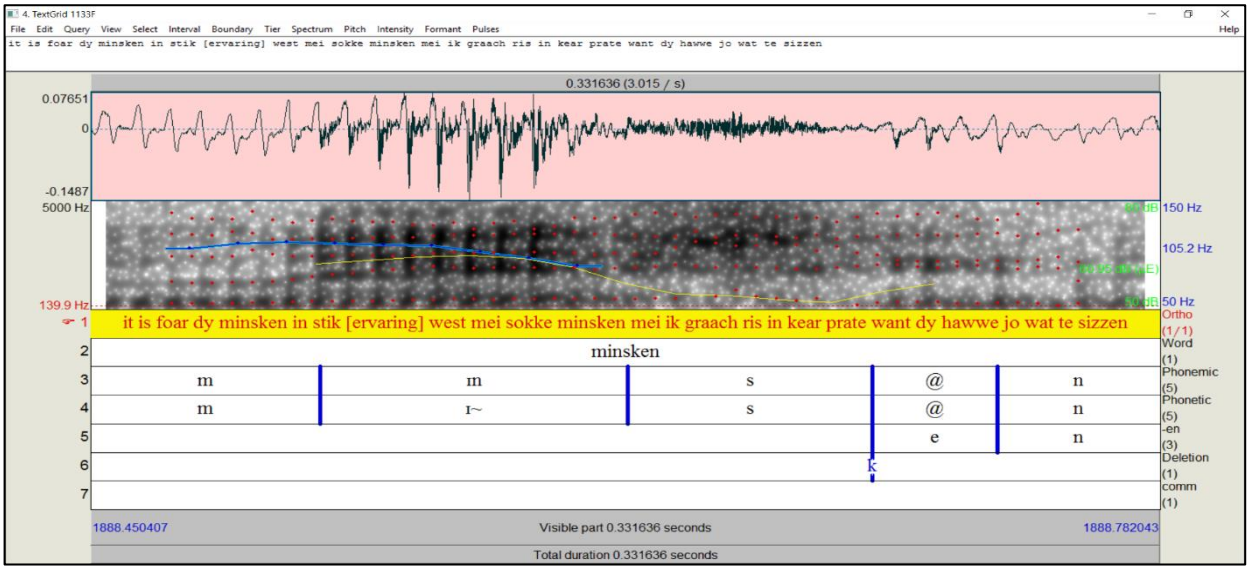


Figure 2: A textgrid from the BHC1.

[2] In the BHC1, higher education correspond to ULO (extended elementary education) for older speakers, and HAVO (senior general secondary education). Since the general education level further increased over the past decades, the cut-off point for the BHC2 is higher, i.e. HBO (higher vocational education) and WO (university) are regarded higher education.

3. Infrastructure

The BHC (1 &2) will be embedded in a large-scale Clarin infrastructure in the Netherlands (Odijk & van Hessen 2017), as part of the European digital infrastructure for the humanities CLARIAH (Odijk 2016). This provides the opportunity to connect information from the BHC to frequency and POS tagging of other databases as well. The corpus will be encoded in Text Encoding Initiative (TEI) (Ide & Véronis 1995, Sperberg-McQueen & Burnard 1995). The different parts of this portal will be gradually published online in the near future.

The BHC will be made available in different phases. Initially, the orthography (and other information in the textgrids) will be published and POS-tagged, so that it can be queried. Subsequently, in later stages, original recordings and linguistic metadata (anonymized speaker data), will be made available. Finally, textgrids and an XML version of them will be added. A webpage with general background information about the corpus is made available at the website of the Fryske Akademy.[3]

4. Frequency Analysis Tool

An important factor in language variation and change is lexical frequency (e.g. Bybee 2006, Phillips 2006, Diessel 2007). Therefore, it is crucial to offer frequency information to the end user. [4] But the corpus size of the complete BHC is nearly 125 hours of speech for both West Frisian and Dutch. Frequency based solely on the BHC may be biased, also because the population sample in the BHC is deliberately homogeneous, and the variety of topics discussed in the recordings restricted. To obtain more reliable frequency data, we turn to The Frisian Audio Mining Enterprise (FAME!) (Yilmaz et al. 2016). FAME! is a corpus consisting of more than 2600 hours of West Frisian Radio Broadcast from 1950-2016. The Corpus Spoken West Frisian contains orthographical annotations of speech by 60 speakers, recorded between 2003 and 2006. We will provide both token and lemma frequency of words.

The BHC will be both lemmatized and POS-tagged by using TreeTagger (Schmid 1994, 1995) or Frog (Avontuur et al. 2012). POS-tagging of the data makes it possible to easily find words of a particular word class, for example verbs, in the corpus. No lemmatizer/POS-tagger for West Frisian exists, but given that the morphological and syntactic structure of West Frisian and Dutch are highly similar, the West Frisian speech data will be translated from West Frisian to Dutch before being POS-tagged. We will implement an application which automatically translates Frisian to Dutch by using the existing API of the online translation system Oersetter (lit. 'Translator').[5]

For the Dutch part, we refer to frequency information in the existing Dutch corpora: the Corpus Spoken Dutch (*Corpus Gesproken Nederlands*) (Oostdijk 2000, 2003, Oostdijk & Broeder 2003), CELEX (Baayen et al. 1993), and Nederlab (Brugman et al. 2016). These corpora also provide token frequencies and lemma frequencies.

5. Research based on the BHC

5.1 Previous research

The BHC1 corpus was partly analysed (35 males and 8 females in separate studies) for five phonological rules in West Frisian:

- schwa deletion in final -/ən/
- nasal place assimilation in final -/ən/
- vowel nasalization
- assimilation of final -/s/ to [z d Ø]
- initial /d/ deletion of the definite non-neuter article [də]

Feitsma et al. (1987) and Feitsma (1989) report that among the male speakers, assimilation of -s and d-deletion occurred more in Frisian than in Dutch. They produced as many nasalized vowels in Dutch as in West Frisian (whereas in Standard Dutch nasalization does not occur). Younger speakers nasalized even more than older speakers, especially in Dutch, pointing at a transfer from Frisian to Dutch. Another age difference was observed for schwa-deletion in final -/ən/: younger speakers used more schwa deletion (the West Frisian rule) in West Frisian and n-deletion (the Dutch rule) in Dutch, whereas the older speakers mixed up the rules in the two languages more often. In this case, the younger speakers kept the two rules better apart than the older speakers. A later study on schwa deletion of eight females pointed towards the same direction (Meekma 1989).

5.2 Current research

Currently, we are investigating the large array of pronunciations of final -/ən/ based on the BHC1 in Frisian and Dutch in more detail. We also investigate the interaction between the final nasal that remains after ə-deletion and the preceding consonant(s). We study the phonological status of the remaining nasal after ə-deletion with regard to syllabicity (Sloos et al. submitted).

5.3 Future research possibilities

The BHC offers a wide range of applications for linguistic research. We plan a sociophonetic study into variation and change in final -/ən/. This will later be extended to a broader investigation into the variation and change of the phonological system of Frisian and Dutch of Frisian-Dutch bilinguals across four generations, spanning the pronunciation of speakers born between 1897 and 1997. The BHC also offers data for a comparative study, based on the two varieties, giving more insight into convergence and divergence in a bilingual contact situation in which one language is dominant.

Secondly, current phonological descriptions of West Frisian lack a phonetic basis. The BHC contains ample material for phonological (re)investigation.

Thirdly, since the same speakers are investigated in the two languages, this corpus serves studies into bilingualism. In addition, the West Frisian-accented Dutch data could be compared to other varieties Dutch, for instance, in the Spoken Dutch Corpus (CGN).

[3] https://www.fryske-akademy.nl/.

[4] The BHC1 studies did not take frequency into account as a factor of variation.

[5] http://oersetter.nl/ accessed at September 2017.

Fourthly, this corpus provides the opportunity to investigate real-time language change, and compare real-time and apparent-time studies into language change.

Finally, the corpus may facilitate studies into language attitude—and changing attitudes—toward the usage of Frisian by first- and second language users. Also the development of reading skills in West Frisian could be investigated, which is interesting given the increasing efforts regarding bilingual education at all levels in the province of Fryslân.

6. Conclusion

We have described in detail our efforts to make available a sociolinguistic corpus of West Frisian and Dutch of the same bilingual West Frisian-Dutch speakers. The corpus contains data for four generations (born between 1897 and 1997). One part consists of speech of speakers that were recorded twice, with an interval of more than 30 years. Along with the sound files and textgrids with much phonological information, anonymized metalinguistic information of the speakers will be made available for research purposes. We will also provide lexical frequency information (token and lemma frequency).

7. Acknowledgements

This research has been made possible through a VENI grant (number 275-75-10) by the Netherlands Organization for Scientific Research to the first author, matched by the Fryske Akademy, which is gratefully acknowledged. The BHC1 studies were funded by *Nederlandse Organisatie voor Zuiver Wetenschappelijk Onderzoek* (currently Netherlands Organization for Scientific Research), Stichting Taalwetenschap Fryske Akademy, and the Fryslân Bank.

8. Bibliographical References

Avontuur, T., Balemans, I., Elshof, L., van Noord, N., & van Zaanen, M. (2012). Developing a part-of-speech tagger for Dutch tweets. *Computational Linguistics in the Netherlands Journal* 2: 34—51.

Baayen, R., Piepenbrock, R. & Gulikers, L. (1995). CELEX2 LDC96L14. Web Download. Philadelphia: Linguistic Data Consortium.

Boersma, P. & Weenink, D. (2017). Praat, doing phonetics by computer.Version 6.0.30. www.praat.org.

Brugman, H., Reynaert, M., van der Sijs, N., van Stipriaan, R., Tjong Kim Sang, E. & van den Bosch, A. (2016). Nederlab: Towards a Single Portal and Research Environment for Diachronic Dutch Text Corpora. *LREC Proc. 2016.* 1277—1281.

Bybee, J. (2006). *Frequency of Use and the Organization of Language*. Oxford: Oxford University Press.

Diessel, H. (2007). Frequency effects in language acquisition, language use, and diachronic change. *New Ideas in Psychology* 25(2): 108—127.

Feitsma, A. (1989). Changes in the pronunciation of Frisian under the influence of Netherlandic. In Deprez, K. (ed.), *Language and Intergroup Relations in Flanders and in the Netherlands*, 181—193. Dordrecht: Foris.

Feitsma, T., van der Geest, E., van der Kuip, F.J. & Meekma, I. (1987). Variations and development in West Frisian sandhi phenomena. *International Journal of the Sociology of Language* 64: 81—94.

Ide, N., & Véronis, J. (Eds.). (1995). *Text encoding initiative: Background and contexts* (Vol. 29). Springer Science & Business Media.

Meekma, I. (1989). Frouljuspraat en it lytse ferskil. Oer útspraakferoaring yn 'e sandhi by froulju en manlju. *It Beaken* 51: 115—129.

Odijk, J. (2016). CLARIAH in the Netherlands. *Proceedings of LREC 2016*.

Odijk J. & van Hessen A. (2017). CLARIN in the Low Countries. London: Ubiquity Press. DOI: https://doi.org/10.5334/bbi

Oostdijk, N. (2000). The Spoken Dutch Corpus Project. In *The ELRA Newsletter*. Vol. 5 No. 2: 4—8.

Oostdijk, N. (2003). Het Corpus Gesproken Nederlands: Veelzijdig onderzoeksinstrument voor o.a. taalkundig en taal-en spraaktechnologisch onderzoek. In *LINK* 14(1): 3—6.

Oostdijk, N. & D. Broeder (2003). Een databank van het gesproken Nederlands. In *Philologia Frisica 2002*: 37—54.

Phillips, B. (2006). *Word frequency and lexical diffusion*. New York: Palgrave Macmillan.

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing.* Manchester, UK.

Schmid, H. (1995): Improvements in Part-of-Speech Tagging with an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*. Dublin, Ireland.

Sloos, M., Ariza Garcia, A. & van de Weijer, J. (submitted). Syllabic vs. non-syllabic nasals in West-Frisian. *Journal of the International Phonetic Association.*

Sperberg-McQueen, C. M., & Burnard, L. (1995). The design of the TEI encoding scheme. *Computers and the Humanities*, 29(1), 17—39.

van Bezooijen, R. (2009). The pronunciation of /r/ in West Frisian. In J.N. Stanford & D.R. Preston (Eds.), *Variation in Indigenous Minority Languages* (Studies in Language and Society 25). Amsterdam and Philadelphia: John Benjamins, pp. 299—318.

van der Kuip, F. J. (1986). Syllabisearring yn it Frysk en it Hollânsk fan Fryskpraters. *Tydskrift foar Fryske Taalkunde* 2: 69—92.

Yilmaz, E., Andringa, M., Kingma, S., Dijkstra, J., van der Kuip, F., Van de Velde, H. & van Leeuwen, D. (2016). A longitudinal bilingual Frisian-Dutch radio broadcast database designed for code-switching research. *LREC Proc. 2016.* 4666—4669.

The French-Algerian Code-Switching Triggered audio corpus (FACST)

Djegdjiga Amazouz[1], Martine Adda-Decker[1,2], Lori Lamel[2]
[1]LPP-CNRS Université Sorbonne Nouvelle Paris-III
[2]LIMSI, CNRS, Paris Saclay University, Orsay, France
djegdjiga.amazouz@univ-paris3.fr, {madda, lamel}@limsi.fr

Abstract

The French Algerian Code-Switching Triggered corpus (FACST) was created in order to support a variety of studies in phonetics, prosody and natural language processing. The first aim of the FACST corpus is to collect a spontaneous Code-switching speech (CS) corpus. In order to obtain a large quantity of spontaneous CS utterances in natural conversations experiments were carried out on how to elicit CS. Applying a triggering protocol by means of code-switched questions was found to be effective in eliciting CS in the responses. To ensure good audio quality, all recordings were made in a soundproof room or in a very calm room. This paper describes FACST corpus, along with the principal steps to build a CS speech corpus in French-Algerian languages and data collection steps. We also explain the selection criteria for the CS speakers and the recording protocols used. We present the methods used for data segmentation and annotation, and propose a conventional transcription of this type of speech in each language with the aim of being well-suited for both computational linguistic and acoustic-phonetic studies. We provide an a quantitative description of the FACST corpus along with results of linguistic studies, and discuss some of the challenges we faced in collecting CS data.

Keywords: Code-switching, bilingual speakers, oral speech data, French, Arabic

1. Introduction

When multilingual speakers communicate using two shared languages in a conversation, they may switch between these languages in a same sentence or utterance. Switching may become a common practice in social groups with strong language contact. This simultaneous use of two language codes can take several forms at the lexical, syntactic and communication levels, and is commonly referred to as code-switching (CS) (Muysken, 2000; Grosjean, 1995). CS may present complex structures in particular at morphological, syntactical and phonetic levels. Hence, CS has interested several research fields such as sociolinguistics and interactional linguistics (Gumperz, 1982; Auer, 2010; Bullock, 2012; Gardner-Chloros, 2009; Sebba, 2012; Tabouret-Keller and Page, 1970; Ziamari, 2008; Piccinini, 2012; Gullberg et al., 2012) with numerous studies describing CS phenomena, explaining the process of its realization and analyzing its productions. Computational linguistics enable the study of large textual and spoken corpora (Çetinoğlu et al., 2016; Schultz and Kirchhoff, 2006). Consequently, experiments have been carried out on language identification (LID) for both oral and written CS as well as automatic identification of CS languages pairs (Lyu et al., 2015; Lyu and Lyu, 2008; Modipa et al., 2013). Furthermore, quantifying CS in large corpora at the word and oral segment levels gives a precise idea about the use of CS in bilingual communities (Amazouz et al., 2016). It also allows the determination of which language is dominant in the CS speech and in the daily CS practices. Applying language processing tools can help to compare CS in different language pairs and to determine how frequent CS is for pairs of languages.

In this paper, we focus on eliciting spontaneous CS and how to collect CS data. And, we question about methods used to build this hybrid oral speech. So, this work describes the design, recording, and annotation of the French Algerian Code-switching Triggered corpus (FACST) being created to support a variety of linguistic studies. Using a methodology designed to elicit CS speech, of 20 bilingual Algerian/French speakers have been recorded in a quiet room.

The paper is organized as follow: presentation of FACST corpus, selection of CS speakers, the recording protocols and the stimuli of CS, and CS data processing.

2. Algerian Arabic dialect, French and CS

The Algerian Arabic dialect (AA) is an oral dialect of North African Arabic dialects group spoken in Algeria and it is the mother tongue of 80% of Algerians. AA is different from Modern Standard Arabic (MSA) at several levels: lexicon, phonetics, phonology, syntax and morphology (Saadane and Habash, 2015; Souag, 2006). MSA is mainly a written language while AA has few written resources. But AA written form becomes more and more widespread especially in social medias. Commonly, AA is written with Arabic characters Table 5 of Appendix. This script is written from right to left. Too, another form of AA script transliterated with Latin characters called "Arabizi transliteration" used heavily on the internet and SMS (Cotterell et al., 2014; Al-Badrashiny et al., 2014; Bies et al., 2014). Thus, AA can be written with two forms.

French language (FR) is the first foreign language spoken in the Algerian community and is for the most part the second language for Algerians. This bilingual community has tens of millions of bilingual speakers who live in Algeria and in France. CS is increasingly part of their daily communications.

Code-switching is a general term referring to language change within a given conversation or an utterance. Code-switching can take many different forms, such as inter-sentential CS (language changes at sentence boundaries), intra-sentential CS (CS within sentences) (Kebeya, 2013),

Code-mixing (insertion of L2 word within L1 utterance) (Muysken, 2000), borrowing, and bilingual verbs (L1 verb inserted in L2 form) (Muysken, 2000).
CS is also characterized by individual choices and individual forms as a placement of adverbs in the beginning or at the end of the sentences, the inclusion or deletion of articles/particle at the switch moment and the construction of CS sentences. In intra-sentential CS, speaker may produce in ungrammatical sentences due to the non-correlation of the two grammar codes but the sentences are semantically correct (Tossa, 1998). In FACST corpus, we used this general term CS for these transcodic marks.

3. FACST corpus

Table 1: Compact FACST presentation

Label	French Algerian Code-switching Triggered (FACST) corpus
Languages	French (FR), Algerian Arabic (AA)
Speakers	20 speakers: 10 male, 10 female Ages: 23-39
Duration	Recordings ranging from 15 to 40 minutes/speaker Total: 7 h 30 of speech
Content	Read speech and stimulated spontaneous speech.
Year	2016-2018

Building an oral CS corpus requires careful attention, in particular on oral languages as Algerian Arabic dialect (AA). Indeed, comparing French language (FR) and AA, AA is an oral language with low written resources unlike French which is standardized with more written resources and more steady in grammar. CS corpus involves selection of speakers who represent CS speech of the linguistic community. It requires too an appropriate methodology to ensure that the recordings contain sufficient instances of CS. Corpus annotation is challenging, requiring guidelines for segmentation of CS and transcription of dialectal speech. In the FACST corpus, these particularities are taken into account in the essential steps of oral corpus construction, that is, segmentation, annotation, transcription and alignment.

3.1. CS speaker selection

We selected participants using a socio-linguistic online questionnaire named *Experience of Code-switching practice* (ECSP). The ECSP form includes questions about linguistic autobiography of the potentially recruted speakers, their bilingualism, the environment in which French and AA languages are practiced, language acquisition/learning, CS habits. . . The participants were selected and recorded in 2016-2018 and were (young) adult speakers aged between 23 to 39 years, with an equal number of men and women. We selected speakers who stated that they used to code-switch in their daily lives and that they use CS at least in two domains among studies, work, family, friends. All the speakers have lived a part of their life in Algeria and another part in France. They all have studied in university. At this date, we transcribed and aligned 13 speakers.

3.2. Recording protocol

The speakers were recorded in a soundproof room at LPP (Labortoire de Phonétique et Phonologie) of Sorbonne-Nouvelle University in Paris or in a very calm room. The corpus is intended to serve multiple CS research purposes including phonetic, prosodic and lexical studies as well as more automatic processing challenges such as language identification and language boundary detection. The main challenge of recording the CS conversations in a lab environment was to meet two antagonistic requirements: ensure the collection of informal speech with a high quantity of CS and at the same time guarantee a high acoustic quality of the signal.
To this aim, each recording session started with a preliminary unrecorded conversation with the speaker, to get her/him in a relaxed setting, practicing both languages in the same interaction. We started the recording with the first task of reading texts in both French and AA. This was followed by discussions designed to elicit spontaneous speech with CS production. Figure 1 shows the corpus organization in two parts: controlled speech with reading texts task and spontaneous speech triggered by questions.

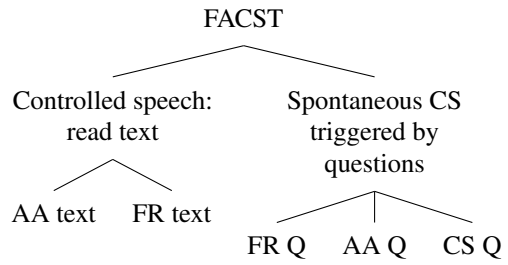


Figure 1: FACST data speech organization.

3.3. Reading tasks

The speakers were asked to perform oral readings of two texts, one in AA and another one in French, at three different speech rates (slow, normal, fast). For French, the text was an excerpt from "Le Petit Prince" ('The little Prince', by A. de Saint-Exupéry). For Algerian Arabic, we used an excerpt from an Algerian movie scenario "Bab El-Oued City" (by M. Allouache) transcribed in Arabic letters. The controlled read speech recordings are summarized in Table 2. The first goal of the read-speech

Table 2: Read speech in French and AA. Number of words including repetitions and average reading times in seconds (slow-medium-fast)

Language	**# words**	**Ave. reading times for 3 rates**
FR	185	92s - 60s - 55s
AA	102	50s - 37s - 30s

recordings was to obtain a controlled monolingual speech

corpus in AA and FR for the bilingual speakers before proceeding to the bilingual speech. Second, the recordings can serve to highlight potential pronunciation differences of consonants and vowels in each language separately. Third, studying the productions at the three speech rates provides data to investigate rate-related differences in the realization of consonants and vowels in each language. So, the controlled speech data helps to apply acoustic analysis and serves to beacon for phonetic observations in CS spontaneous speech.

3.4. Elicited CS speech

In this step, we recorded dual conversations between the linguist (who is a bilingual speaker of both languages) and the speaker. The CS conversations are triggered by questions. The principal questions were inspired by the feedback in the ECSP questionnaire of each speaker. Most of the subjects covered by the speakers were about describing and comparing life in both countries, studies in both countries and conversations about language and bilingualism practices. Other sub-themes were also addressed in the conversations following a free speech approach. So, the role of the linguist in this task is to ask the questions and let the speaker answer freely, making spontaneous use of CS. The recordings of this task lasted from 10 to 25 minutes for each speaker. To elicit CS, we asked four main types of questions in each speech sequence. These questions are summarized as follows:

1. AA questions about studies and work.
2. FR questions about life and studies in Algeria.
3. FR/CS: code-switched questions using French as the base language.
4. AA/CS: code-switched questions using AA as a base.

In these stimuli, the linguist tries to implicitly suggest the use of AA and French in a same conversation to obtain CS spontaneous speech.

3.5. Segmentation and annotation

First, manual segmentation of the recordings is based on language change, breath groups and speaker turns. There are two types of segments: language segments which correspond to one language, and breath group segments which correspond to a rhythmic group of an utterance or an oral sentence. The lengths of the segments are quite variable, ranging from very short segments (less than 2s) to longer ones, with an average segment length of 6s. Indeed, a switch can be limited to a very short word such as an article or a particle. The aim of this segmentation is to get boundaries for each language and label them "ALG" for AA segments and "FRA" for French segments. This type of segmentation and annotation helps thereafter when processing and manipulating the data. We used Transcriber program (Barras et al., 2001) to segment and annotate FACST data. Figure 2 shows an example of segmentation with the segment-level annotations.

```
SP1 12.211 12.771 <male> FRA: quatre vingt
SP1 12.771 13.264 <male> ALG: zaAyid fiy
SP1 13.264 17.856 <male> FRA: en quatre vingt huit. Il a commencé à
                              travailler en tant que pâtissier
SP1 17.856 19.201 <male> ALG: cand xaAliy
SP1 19.201 22.962 <male> FRA: pour huit mille dinars
SP1 22.962 25.917 <male> ALG: baAX yacTiyhum li maAmaA
SP1 25.917 26.378 <male> FRA: parce que
SP1 26.378 28.058 <male> ALG: maA kaAnX candanaA  hiya kaAntX taxdam
SP1 28.058 28.667 <male> FRA: Etcetera
SP1 28.667 29.740 <male> ALG: xwaAliy
SP1 29.740 38.848 <male> FRA: il nous ont beaucoup , il nous ont beaucoup aidé
                              et limite c'est grâce à ça que je suis devenu,  ce que on
                              est devenu aujourd'hui
SP1 38.848 41.346 <male> ALG: maA xalaAwnanAX maA xalaAwX xuthum
```

Figure 2: Example of segments annotation by speaker code (SPx), time-codes (columns 2-3), gender, language code (FRA/ALG) and transcription.

3.6. CS Transcription

The segments were manually transcribed using an orthographic transcription for French and a transliterated transcription for AA inspired by Buckwalter Arabic transcription (BKW) (Buckwalter, 2002) and modified according to AA specificities. The aim of the transliteration is to get a scripts with the same characters in both languages and to get a script which is written in the same direction (Arabic script is written from right to left). This transcription convention has been created in order to facilitate the use of the manual transcriptions, without special characters, for phonetic analyses while keeping the possibility to convert the transliterated characters to Arabic characters in future studies. Table 5 of Appendix illustrates the characters chosen for FACST and the corresponding symbols for each characters in Arabic letters, BKW convention and IPA symbols. For AA and FRA languages, the transcription also includes pauses, repetitions, hesitations, speech backchannels and various linguistic disfluencies. 13 of the speakers are transcribed at this date. Table 3 shows statistics about the number of segments and words counted in CS speech of the transcribed speakers.

In Arabic script, the articles and some particles are attached to the word: اَلبَاب (بَاب + **اَل**) "the door", لِلبَاب (بَاب + اَل + **لِ**) "for/to the door". For AA segments, we transcribed the articles and a large number of particles, placed initially at the beginning of the words, separately from the word. This separation is applied in order to count the number of words and to compare speech production at word level in CS FR-AA. So, in FACST transcription the utterance لِلبَاب is transcribed "li al baAb" "for/to the door".We used this to readily separate the languages in intra-sentential CS. Example: "liy" in AA, a mark placed at the end of a word refers to pronoun suffixes, conjugation morphemes, and number and gender marks.Thus, due to the morphological construction of AA (Souag, 2006), we did not apply separation for attached morphemes at the end of words in this corpus, examples: attached objects "jabt**hum**" *I brought **them** back.*

With the help of these transcriptions, we counted too the

Table 3: FACST spontaneous speech: number of segments and words counted for each language and for each speaker (13 speakers)

Spks#	FR seg	FR wrd	AA seg	AA wrd
SP1 M	28	112	36	108
SP2 F	126	583	170	734
SP3 F	219	1283	126	369
SP4 F	113	720	67	230
SP5 M	175	1619	93	348
SP7 M	180	1157	98	243
SP8 F	288	829	186	277
SP10 M	84	793	11	45
SP11 F	307	1746	236	948
SP12 F	334	1960	223	598
SP16 M	419	2075	277	1108
SP19 F	372	2122	248	710
SP20 F	335	1906	202	1172

highest occurrences of words in AA and FRA in order to know the words and the utterances which appears frequently in this pair of languages. Especially, getting a precise overview of AA insertion as an embedded language in FRA as a matrix language. The lists bellow present words and groups of words occurrences > 20 for AA and > 60 for FRA.

AA: waAHad, allah, bazzaAf, gaAc, waAluw, in, kaAmal, bacd, hnaAyaA, kiymaA, wiyn, kaAyan, laAzam, claY, hnaA, kunt, liy, mca, baAX, iyh, min, li, kaAn, lak, taAc, anaA, waAX, dzaAyar, kiy, alliy, laA, fiy, maA, wa, al, salaAm.

FRA: c'est-à-dire, un petit peu, il y a , deux, non, dans, très, par, beaucoup, quand, voilà, enfin, fait, plus, ils, tout, même, oui, tu, pour, donc, on, qui, parce que, mais, une, ça, en, j'ai, un, des, il, à, et, le, la, les, c'est, pas, que, je ...

A lot of AA words introduce CS, they start AA segment after FRA segment or end AA segment to switch to FRA. Mostly this words are particles:"fiy" in, "li" for/to, "taAc" for, "wiyn" where, "claY" on, "alliy" ... They connect the two languages and introduce a significant part of switches with a high number of occurrences as shown in table 4.

The following examples illustrate the dispositions of theses words in CS speech.

"FRA: internet
ALG: wa tXuwf al **fiy**
FRA: les forums"
"FRA: une licence
ALG: **fiy** al carbiyyaT
FRA: français et anglais"
"FRA: et du coup
ALG: **fiy**
FRA: la faculté"
"FRA: il avait des notes très très basses
ALG: **fiy**
FRA: français"
"FRA: le bouchon
ALG: **alliy** kaAyan **fiy**
FRA: le réservoir"

word	#occurrences	#occ introducing CS
fiy	208	112
wa	224	54
alliy	60	21
li	34	14
taAc	32	25

Table 4: AA word occurrences introducing CS > 14 occurrences

3.7. Data alignment

The data has been aligned using LIMSI speech recognizer in a forced alignment mode (Gauvain et al., 2003), assigning word and phone level time codes. The alignments made use of acoustic models from different ASR systems (Gauvain et al., 2003; Laurent et al., 2016) in parallel: a French system, and an Algerian Arabic (dialect) system. The alignment is composed by two separate parts. First alignment for French segments using pronunciation dictionary of standard french.The second alignment for ALG segments with pronunciation dictionary of AA created on the basis of MSA model and adapted on AA with specific phonemes used in this dialect as /p, v, g, ʒ/. Thereafter, both alignments are combined to the speech signal as shown in Figure 3.

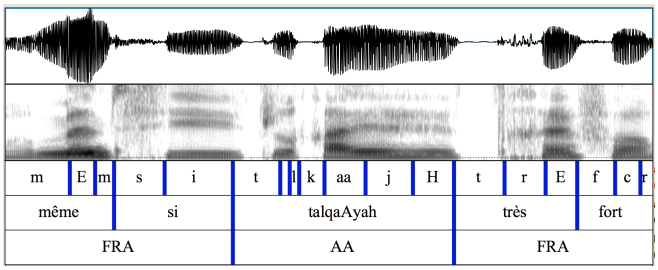


Figure 3: Example of FACST alignment on intra-sentential CS audio segment with spectrogram. Three levels of segmentation and transcription are applied. From top to bottom of tiers: phoneme segmentation and phonetic transcription, word segmentation and orthographic transcription, language segmentation and annotation.

4. Discussion and future work

In spite of a high CS production in the FR-AA bilingual community, recording speakers in laboratory requires a careful preparation of the communicative situation and of the setting of CS usage (Gumperz, 1982). The role of the linguist, in this experience is, to be an interactive speaker who can adapt himself to the daily CS usage of the

participant in order to incite a CS conversation.
We found that frequent CS could be elicited by different questions, but the number of segments in FR and AA is not equal. FR has a higher number of segments and of produced words than Algerian for almost all speakers Table 3 shows clearly the quantity of AA and FRA distribution in CS spontaneous speech. So, we can say that in the FACST spontaneous speech corpus, FR can be considered as the dominant language and AA as dominated one. In the segmentation of spontaneous speech portions of FACST, some segments are very short (less than 1 second) because of the speed of language change. These segments generally correspond to particles and articles of both languages. One of the major difficulties of language annotation in CS FR-AA is the articles **al** /l/ in AA and **l'** /l/ in french at the switches. The two articles are pronounced identically but it is very difficult to identify the language of this word at the switch moment. E.G.: (fiy **l'**école fiy-AA, **l'**-FR, école-FR) or (fiy **al** école fiy-AA, **al**-AA, école-FR)
CS verbs can take a base in one language but are redesigned with the other language form. An illustration is the following CS example of AA-FR:
ypartaAj**iy** بيَرتَاجِي **/j**parta:ʒ**i:**/ "he shares"
This type of neologism is not easy to classify as French or as AA because it doesn't conventionally belong to none of the two languages. The root of the verb in bold ***ypartaAjiy*** is in French *partager* "to share". The prefix and suffix *y - iy* are in Arabic: the present form of the verb with the pronoun "he".
In summary, this paper describes the steps used to construct a CS corpus containing spontaneous speech and read bilingual speech. We present the methods employed to obtain CS in a dual conversation between a linguist and a speaker. We developed a methodology to annotate the CS corpus and we proposed transcription conventions for dialectal Algerian Arabic speech. An acoustico-phonetic study will be planned to analyze CS phenomena, in particular, phonemes at the moment of switches and the question of articles mentioned above.

5. Acknowledgements

This work is supported in one hand by the French National Agency for Research as part of the SALSA project (Speech and Language technologies for Security Applications) project under grant ANR-14-CE28-0021 and in the other hand by the French Investissements d'Avenir - Labex EFL program (ANR-10-LABX-0083). The study was conducted at LPP-CNRS (UMR 7018) Paris-III Sorbonne Nouvelle University and LIMSI-CNRS Paris-Orsay University. Many thanks to the 20 participants recorded who contributed to realize the FACST corpus.

6. Bibliographical References

Al-Badrashiny, M., Eskander, R., Habash, N., and Rambow, O. (2014). Automatic transliteration of romanized dialectal arabic. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 30–38.

Amazouz, D., Adda-Decker, M., and Lamel, L. (2016). Arabic-french code-switching across maghreb arabic dialects : a quantitative analysis. In *Workshop "Corpus-driven studies of heterogeneous and multilingual corpora"*, pages 5–7.

Auer, P. (2010). *Language and Space: An International Handbook of Linguistic Variation. Theories and Methods*, volume 1. Walter de Gruyter.

Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (2001). Transcriber: development and use of a tool for assisting speech corpora production. *Speech Communication*, 33(1):5–22.

Bies, A., Song, Z., Maamouri, M., Grimes, S., Lee, H., Wright, J., Strassel, S., Habash, N., Eskander, R., and Rambow, O. (2014). Transliteration of arabizi into arabic orthography: Developing a parallel annotated arabizi-arabic script sms/chat corpus. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 93–103.

Buckwalter, T. (2002). Arabic transliteration. *URL http://www. qamus. org/transliteration. htm*.

Bullock, B. E., (2012). *Phonetic reflexes of code-switching*, chapter 10, pages 163–181. Cambridge University Press, Cambridge.

Çetinoğlu, Ö., Schulz, S., and Vu, N. T. (2016). Challenges of computational processing of code-switching. *arXiv preprint arXiv:1610.02213*.

Cotterell, R., Renduchintala, A., Saphra, N., and Callison-Burch, C. (2014). An algerian arabic-french code-switched corpus. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme*, page 34.

Gardner-Chloros, P. (2009). *Code-Switching*. Cambridge University Press, Cambridge.

Gauvain, J.-L., Lamel, L., Schwenk, H., Adda, G., Chen, L., and Lefevre, F. (2003). Conversational telephone speech recognition. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–I. IEEE.

Grosjean, F. (1995). A psycholinguistic approach to code-switching: The recognition of guest words by bilinguals. *One speaker, two languages: Cross-disciplinary perspectives on code-switching*, pages 259–275.

Gullberg, M., Indefrey, P., and Muysken, P., (2012). *Research techniques for the study of code-switching*, chapter 2, pages 21–39. Cambridge University Press, Cambridge.

Gumperz, J. J. (1982). *Discourse strategies*, volume 1. Cambridge University Press.

Kebeya, H. (2013). *Inter-and intra-sentential switching: are they really comparable?* Ph.D. thesis, Kenyatta University.

Laurent, A., Fraga-Silva, T., Lamel, L., and Gauvain, J.-L. (2016). Investigating techniques for low resource conversational speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5975–5979. IEEE.

Lyu, D.-C. and Lyu, R.-Y. (2008). Language identification on code-switching utterances using multiple cues. In *Interspeech*, pages 711–714.

Lyu, D.-C., Tan, T.-P., Chng, E.-S., and Li, H. (2015). Mandarin–english code-switching speech corpus in south-east asia: Seame. *Language Resources and Evaluation*, 49(3):581–600.

Modipa, T. I., Davel, M. H., and De Wet, F. (2013). Implications of sepedi/english code switching for asr systems. *PRASA 2013, Johannesburg, South Africa*.

Muysken, P. (2000). *Bilingual speech: A typology of code-mixing*, volume 11. Cambridge University Press.

Piccinini, P. E. (2012). Cross-language activation and the phonetics of code-switching. *Speech Communication*, 52(11-12):930–942.

Saadane, H. and Habash, N. (2015). A conventional orthography for algerian arabic. In *ANLP Workshop 2015*, page 69.

Schultz, T. and Kirchhoff, K. (2006). *Multilingual speech processing*. Academic Press.

Sebba, M. (2012). On the notions of congruence and convergence in code-switching. In Almeida Jacqueline Toribio et al., editors, *The Cambridge handbook of linguistic code-switching*, chapter 3, pages 40–57. Cambridge University Press, Cambridge.

Souag, M. L. (2006). Explorations in the syntactic cartography of algerian arabic. Master's thesis, School of Oriental and African Studies (University of London).

Tabouret-Keller, A. and Page, R. B. L. (1970). L'enquête sociolinguistique à grande échelle: Un exemple: Sociolinguistic survey of multilingual communities. part i, british honduras survey. *La Linguistique*, 6(2):103–118.

Tossa, C.-Z. (1998). Phénomènes de contact de langues dans le parler bilingue fongbe-français. *Linx. Revue des linguistes de l'université Paris X Nanterre*, 1(38):197–220.

Ziamari, K. (2008). *Le code switching au Maroc: l'arabe marocain au contact du français*. Collection Espaces discursifs. L'Harmattan.

7. Appendix

Transliteration convention for AA consonants and vowels

	IPA	**FACST symbol**	Arabic symbol	BKW symbol	Examples	Transl
Plosives	p	**p**	پ		پلَاصَا plaSaA	place
	b	**b**	ب	b	بَاطل bATal	free
	t	**t**	ت	t	تَبَّع tabbac	follow
	t	**M**	ة	t	حيَاة HyaAM	life
	tˤ	**T**	ط	T	طبِيب Tbiyb	doctor
	d	**d**	د	d	دَبزَة dabzaM	punch
	dˤ	**D**	ض	D	ضَحكَة DaHkaM	smile
	k	**k**	ك	k	كتَاب ktaAb	book
	g	**g**	گ	g	گَاع gaAc	all
	q	**q**	ق	q	قَرعة qarcaM	bottle
	ʕ	**E**	أ	>	أَمَل Emal	hope
Affrc	dʒ	**j**	ج	j	جُوع juwc	hunger
	dz	**dz**	جز		جزَايَر dzaAyar	Algérie
Nasals	m	**m**	م	m	مَاضي maADiy	past
	n	**n**	ن	n	نُوم nuwm	slumber
Fricatives	f	**f**	ف	f	فُوق fuwq	on
	v	**v**	ڤ		ڤِيلَا viylaA	villa
	θ	**F**	ث	v	ثُوم Fuwm	garlic
	ð	**V**	ذ	*	هَذَا haVaA	this
	s	**s**	س	s	سفَر safar	travel
	sˤ	S	ص	S	صُور Suwr	wall
	z	**z**	ز	z	زيت ziyt	oil
	ʃ	**X**	ش	$	شَاف XaAf	saw(you)
	x	**x**	خ	x	خرِيف xriyf	autumn
	ɣ,ʁ	**G**	غ	g	غرِيب Griyb	foreign
	ħ	**H**	ح	H	حِيط HiyT	wall
	ʕ	**c**	ع	E	عِين ciyn	eye
	h	**h**	ه	h	هُومَ huwma	they
Laterals	l	**l**	ل	l	لِيل liyl	night
	r	**r**	ر	r	رَاس raAs	head
Spirants	w	**w**	و	w	وَردَة wardaM	rose
	j	**y**	ي	y	يَد yad	hand
vowels	i	**i**	اِ	i	ضِيَّع Diyyac	lose
	u	**u**	اُ	u	جُملَة jumlaM	phrase
	a	**a**	اَ	a	فَرجَة farjaM	show
	a	**Y**	ى	Y	علَى claY	above
	iː	**iy**	اِي	iy	خشِين xXiyn	bold
	uː	**uw**	اُو	uw	غُول Guwl	monster
	aː	**aA**	اَا	aA	مَاكلَة MaAklaM	food

Table 5: Transliteration of AA consonants and vowels in FACST corpus with corresponding symbols in IPA, BKW, Arabic symbols and examples of words. FASCT symbols represent the phonological pronunciation of AA (Algiers region)

Strategies and Challenges for Crowdsourcing Regional Dialect Perception Data for Swiss German and Swiss French

Jean-Philippe Goldman, Simon Clematide, Matthieu Avanzi, Raphael Tandler
University of Zurich, Switzerland
University of Geneva, Switzerland
FNRS/Université catholique de Louvain, Belgium

jeanphilippegoldman@gmail.com

Abstract

Following the dynamics of several recent crowdsourcing projects with the aim of collecting linguistic data, this paper focuses on such a project in the field of Swiss German dialects and Swiss French accents. The main scientific goal of the data collected is to understand people's perception of dialects and accents, and provide a resource for future computational systems such as automatic dialect recognition. A gamified crowdsourcing platform was set up and launched for both main locales of Switzerland: "din dialäkt" ('your dialect') for Swiss German dialects and "ton accent" ('your accent') for Swiss French. The main activity for the participant is to localize preselected audio samples by clicking on a map of Switzerland.

The media was highly interested in the two platforms and many reports appeared in newspapers, television and radio, which increased the public's awareness of the project and thus also the traffic on the page. At this point of the project, 7,500 registered users (beside 30,000 anonymous visitors), have provided 470,000 localizations. By connecting user's results of this localization task to their socio-demographic information, a quantitative analysis of the localization data can reveal which factors play a role in their performance. Preliminary results showed that age and childhood residence influence the how well dialects/accents are recognized.

Nevertheless, quantity does not ensure quality when it comes to data. Crowdsourcing such linguistic data revealed traps to avoid such as scammers, or the participants' quick loss of motivation causing them to click randomly. Such obstacles need to be taken into account when assessing the reliability of data and require a number of preliminary steps before an analysis of the data.

Keywords: Swiss German dialects, French accents, regional variation, cartography, crowdsourcing

1. Introduction

Voices are highly individual and people often wonder where this individuality stems from. Not only we can often guess sociolinguistic attributes like gender, age, and mood from an unknown voice, but also a vast number of characteristics inform us on the speaker's specific regional and social background.

In recent years, several academic projects as well as media initiatives on linguistic variation were launched: see the numerous applications by Leemann et al. (2016) only for English and German dialectal variation; Avanzi et al. (2016) for European and Canadian French; *Atlas der deutschen Alltagssprache* (Möller & Elspaß 2015) for regional varieties of German; the *Harvard Dialect Survey* (Vaux and Bert 2013) for regional variation in the USA; *VerbaAlpina* (Krefeld and Lücke 2014) for dialects spoken in the Alps; and more recently *Donnez Votre Français* (Goldman 2018). They were developed for the web (accessible via computer, tablet or smartphone) or as smartphone applications. A significant number of these projects were initiated in Switzerland, whose variety of languages and unique dialect landscape allows for interesting studies. All of these projects showed the great interest of the public in regiolects, and beyond that, the public's will to understand more about their own voices and the regional linguistic variations of their country.

Following the dynamics of several recent crowdsourcing projects for collecting linguistic data (Cook et al. 2013), the framework presented in this communication focuses on dialects and accents perception. The main goal is to understand people's perception of dialects and accents and provide material for automatic dialect recognition. For this, a gamified crowdsourcing platform was set up and launched in April 2017 for both main locales in Switzerland: "din dialäkt" ('your dialect') for Swiss German dialects and "ton accent" ('your accent') for Swiss French. The main activity is for the participant to localize pre-selected audio samples within Switzerland by clicking on a map. The media was highly interested in the two platforms and many reports appeared in newspapers, television and radio, which increased the public's awareness of the project and thus also the traffic on the page. At this point of the project, 7,500 registered users (beside 30,000 anonymous visitors), have provided 470,000 localizations. By connecting user's results of this localization task to their socio-demographic information, a quantitative analysis of the localization data can reveal which factors play a role in their performance. Preliminary results showed that age and childhood residence influence the how well dialects/accents are recognized. A detailed analysis of each factors' influence will be presented in a paper later this year (Hundt et al. 2018).

Nevertheless, quantity does not ensure quality when it comes to data. Crowdsourcing such linguistic data

revealed traps to avoid such as scammers, or the participants' quick loss of motivation causing them to click randomly. Such obstacles need to be taken into account when assessing the reliability of data and require a number of preliminary steps before an analysis of the data.

We present the linguistic situation in Switzerland, i.e Swiss German dialects and Swiss French accents (Section 2), then describe the audio samples (Section 3) used in the game. In Section 4, we discuss the gamification of the localization task. After describing the participation on the platforms (Section 5), we finally discuss about the drawbacks of a geolocating perception task, the relevance of the socio-demographic information from the users and the reliability of data.

2. Swiss linguistic landscape

In Switzerland, standard German is the official language of 4.5 millions of people (which represents 65% of the population), spread across almost three quarters of the country. French is the standard language of 25% of the population (1.8 millions), gathered on the very western part of the country, in an area called Romandy. Italian and Romansh, the two other official languages of the Confederation, constitute linguistic minorities: they are spoken in small areas in the south and in the south-west of the country.

From a linguistic point of view, varieties of German spoken in Switzerland are noticeably different from standard German: standard German speakers often encounter difficulties when trying to understand Swiss German speakers when the latter communicate in their dialect. Practically, the two linguistic system are in a situation of diglossia: standard German is preferentially used for formal contexts (in particular for writing), while in everyday oral communication, Swiss German dialects are spoken (Ferguson 1959). In addition, it is important to underline that Swiss German is not one uniform language, but consists of a variety of mutually intelligible dialects that differ in terms of lexicon, syntax, morphology and intonation (Siebenhaar 1997: 30). For instance, morphology differs between northern and western dialects, as the former have one or two desinences for the plural of the verb, while the latter have three. Another example for variation in pronunciation would be the presence or absence of diphthongization, etymological [k] evolution, etc., see Sprachatlas der Deutschen Schweiz (1962-2003). This linguistic variation depends, among other factors, on the geographic distance that separate the dialects at stake. While the isoglosses between dialect features draw a highly complex picture of the dialect continuum (visible e.g. in Hotzenköcherle 1984) that allow for a multitude of ways to group small-scale dialects into dialect regions, the general public usually thinks about dialects in terms of political canton boundaries (Siebenhaar 1997: 30).

In the Romandy, the linguistic situation is quite different. Ancestral Gallo-Romance dialects herited from Latin were replaced by French at different stages of history (the process started in the 15th century), and those earlier dialects are barely spoken anymore. Nevertheless, they have left some traces in the French spoken within this area, especially regarding lexicon and pronunciation, which render Swiss French a quite well-identified regiolect when compared with other varieties of European French (Northern/Standard French, Belgian French, Southern French, etc. see Avanzi and Boula-de-Mareüil, 2017, and references therein). In contrast to the aforementioned difficulties for a German to understand a Swiss-German speaking dialect, there are no such obstacles for Standard and Swiss French speakers: a Parisian will easily understand a Swiss French speaker despite his accent, since the two varieties differ only in minor points. Within the Romandy, it is common practice to distinguish Swiss French varieties based on the canton where they are spoken, even if some people will claim that they can make finer distinctions between speakers from different places within the same canton. Lexical cues are usually mentioned by Swiss people when asked which feature they based a localization on, but usually it is the accent (pronunciation variants) that allows one to identify the region of origin of a given Swiss French speaker.

3. Audio data

The auditory input in the games was drawn from several different corpora for French and German. For the French game, all the samples were drawn from the OFROM corpus, which was recorded and transcribed by the University of Neuchatel between 2008 and 2014 (Avanzi et al. 2015). The samples for the German game were extracted mainly from three corpora: the SDS Phonogramme (Phonogrammarchiv Zürich 2001), Archimob (Vereinigung Archimob 2000; Samardžić et al. 2016) and Stimmen der Schweiz (Glaser/Loporcaro 2012). Due to a lack of speakers in the cantons of Appenzell Innerrhoden, Grisons, Aargau and Fribourg in the corpora mentioned above, a few samples were extracted from other sources (see acknowledgments section).

Here is a transcribed example for each language:

- a male French-speaker from Vaud, born in 1925 (87 years-old during the recording), retired, former wine-producer:

ben maintenant ils | ils remplissent leur cageot donc fin euh enfin leur caisse | c'est toujours les caisses | mais | ils vont plus laisser y a y a des | les dames qui vendangent | pis y a des jeunes qui portent la caisse

well now they | they fill their crate | so | well their boxes| it's always boxes | but | they won't let, there is | there is some | the ladies who harvest | and there are young people who carry the box

- a male German-speaker from Appenzell Innerrhoden born in 1957 (60 years-old during the recording), butcher

Aso gfloge simmer no nü meng mal | mier sind, äh, mal, ääh | vor sechs Jahr simmer uf Gran Canaria | mit de Gove | d'Meitli hend gmeint sie wettid öu e chli as Meer id Ferie | de simmer uf Gran Canaria | und denn hemmer gseit, da hemmer jetzt gseh, etz müemmer nöd unbedingt

So we haven't flown many times | we went, uh, once, uuh | to Gran Canaria six years ago, with the kids | the girls said they wanted to go to the beach a little during holidays | so we went to Gran Canaria | and then we said, we've seen it now, now we don't have to [anymore] necessarily

Certain constraints on the selection of the samples were introduced. In terms of duration, the samples had to be between 10 and 20 seconds. Content-wise, samples could neither contain any personal names, in order to ensure the anonymity of the speakers, nor any geographical clues that would hint towards the speaker's location. Furthermore, the aim was to exclude any hints pointing towards regional culture in the content, for instance in form of regional legends or politics.

Among the pre-selection of 3 to 9 samples that were extracted from the audio-files in accordance with the aforementioned constraints, one sample was chosen based on the auditory quality (not too many long breaks; no great divergence of loudness; no distracting sounds) and whether the content would be interesting to the players. No speaker appears more than once, in order to ensure that users do not perform better by remembering a voice.

Unlike other studies in the field of perceptual dialectology, such as Baker et al. (2009), linguistic features were not taken into account for the selection of the samples. In contrast to the corpus of the Baker study that consisted of speech based on a pre-written dialogue, the corpora for this project contained mostly free speech. Considering the variety of the many recordings, controlling the salient features would not have been possible.

As a consequence of not controlling the linguistic features within the single samples, the difficulty of locating a specific sample might also have differed depending on the amount of salient features of a dialect that appeared in a sample, especially in the case of lexical or certain phonological cues. However, as all the samples are longer than 10 seconds, it is very unlikely that no salient features would be included in any of the samples. Furthermore, the difference of linguistic features can also be used as a fruitful basis for a study on salience by comparing the recognition rate of samples of the same dialect that include or exclude a certain feature.

There are a number of factors concerning the samples that need to be taken into account in the analysis. As mentioned above, the samples in the game had to be drawn from a variety of sources due to the lack of an extensive spoken corpus of contemporary Swiss German. Two of these sources, the SDS Phonogramme and the Stimmen der Schweiz, which form a substantial part of the samples, are recordings made approximately between 1940 and 1960. For this reason, the audio quality of some of the older samples is not of the same standard as the newer ones. Furthermore, the speakers recorded in these two projects were older people whose dialects nowadays are not spoken in their way anymore. This means that not only knowledge of current, but also of older dialects determined the performance of the player. As a consequence, age might become a more important factor, as the older players might still have had contact with those older stages of the dialect.

4. Gamification

In order to keep up the motivation of the participant, we had go beyond his/her interest into regional variation and we turned the localisation task into a facilitated and gamified activity by several means:

- Instead of offering a long-haul series of samples to localise on a map (150 samples in French and 114 samples in Swiss-German), we **organized the game in rounds** of 10 samples. Dividing this task in sub-tasks is much more attractive.
- Another way to motivate users is to offer them simple tasks at first then lead them **progressively more difficult tasks**. In that sense, we split our game in two modes (see Table 1):
  - In the so-called **easy mode** ("einsteigen" task in the German game, "amateur" for the French one), the player has to choose one canton among a pre-selection of cantons. For each of the Swiss-German samples, a different set of 5 cantons (out of 20 existing ones) was selected. For the Swiss French game, all the cantons were selectable, as there are only seven. Scoring is binary, i.e either the player scores 100 points or nothing (see Figure 1).
  - In the **expert mode**, the player has to click on an exact location within the Swiss map. The score depends on the distance between the user's click and the exact location of the audio sample. A *grace area* of 10 kilometers in radius around the sample's location gives the user the maximum number of points (see Figure 2).
  - As the "klettern" task with its unrestricted localization could be rather challenging and frustrating, the users first had to complete the "einsteigen" task, where most players would be more successful.
- Newcomers discover the game with a **trial round** of 3 samples in each mode (easy and expert), then they are incented to register with email address, username

and password only. This trial round is also considered as a training stage to ensure that they have understood the mechanics of the game before their performance is recorded. In order to avoid that our participants do not complete the registration process, we delay the presentation of the form with socio-linguistic questions (after few answers), we also ask these various informations one by one after each round. Of course, they are offered to fill up the socio-linguistic form at once.

- We also **gamified** the tasks with points and set up a leaderboard where users can see their own ranking as well as the top scorers.
- Finally, some efforts were done to facilitate the **user experience** (e.g simplified instructions, blinking buttons to assist navigation)

	Easy mode	Expert mode
de	trial round of 3 samples 3 rounds of 7 samples total = 24 samples	trial round of 3 samples 9 rounds of 10 samples total = 93 samples
fr	trial round of 3 samples 2 rounds of 7 samples total = 17 samples	trial round of 3 samples 13 rounds of 10 samples total = 133 samples

Table.1 Nb. of rounds and samples per language and mode

	regis. users	childhood residence	birth decade	gender
de	3705	62%	55%	49 %
fr	2519	50 %	42 %	36%

Table 2. Proportion of sociolinguistic information provided by the participants.

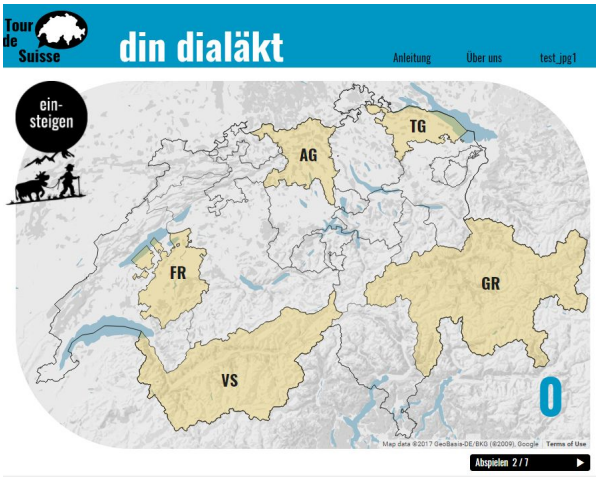


Figure 1. Easy mode for Swiss German (5 possible answers)



Figure 2. Expert mode for Swiss German (after the participant clicked onto the map)

5. Crowdsourced results

After few months online, the platform had a great success and allowed us to gather a lot of data. About 7'500 users have registered. The birth decade was given by 43% of them and their childhood residence by 56 % of them. In the end, about 470'000 tasks on map were achieved. The logs showed that a number of players quickly lost interest (Figure 3 and 4) and that the proportion of them who went through all the rounds is rather limited (19% for DE and 6% for FR, see the rightmost bins). Nevertheless, preliminary results showed interesting effect of age and childhood residence to be further evaluated in final version of this paper.

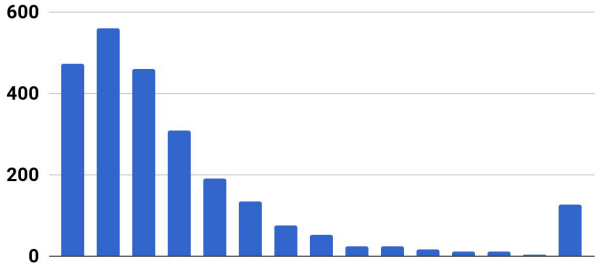


Figure 3. Number of French users as number of completed rounds (from 1 to 15 rounds)

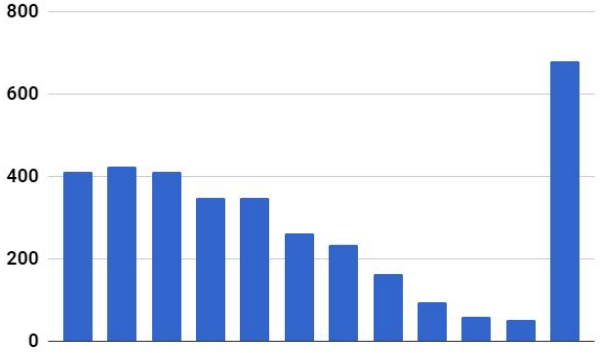


Figure 4. Number of Swiss German users as number of completed rounds (from 1 to 12 rounds)

Here are some map results of the expert game for German and French (Figures 5 to 8), where the individual answers

(red dots) are superimposed with "heat-map" modeling and the ground truth (green cross):

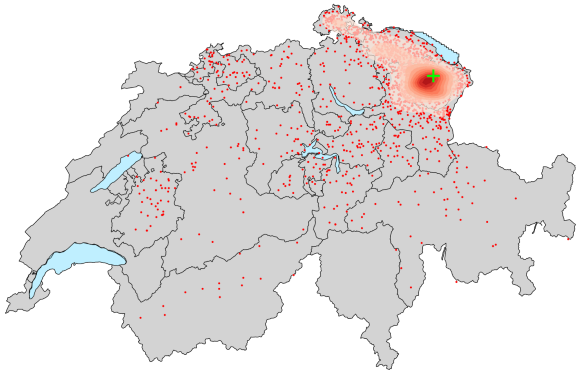


Figure 5: Correct global identification of a German-speaker from St.Gallen (task #494)

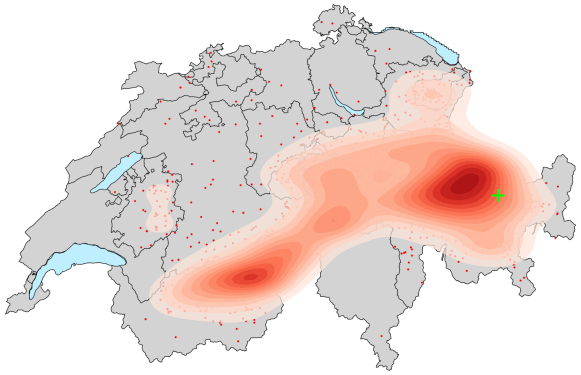


Figure 6: Bi-modal recognition of a German-speaker from Grisons (task #492)

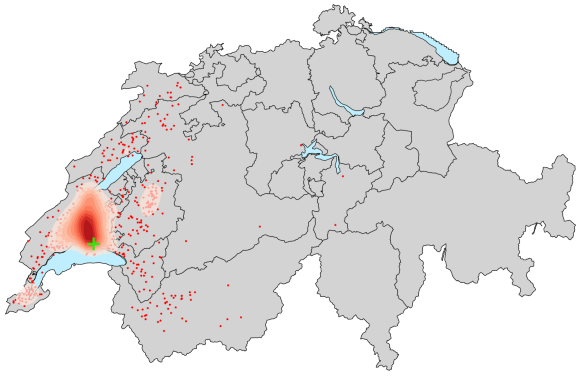


Figure 7: Correct global identification of a French-speaker from Lausanne (task #331)

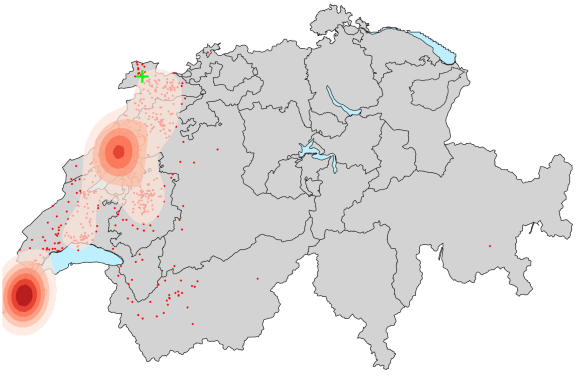


Figure 8: Bi-modal recognition of a French-speaker from Jura (task #332). The two modes are located in Geneva (lower group) and Neuchâtel (upper group).

6. Discussion

On the basis of this experiment, various questions can be addressed and debated:

1. **Danger of geolocating: are we testing the geographical competence and/or dialect perception competence?** The selection of geographic information displayed on the map is crucial (administrative boundaries, cities, main roads, water bodies) as they may influence the answer regarding both, the geographical knowledge and the dialectal landscape representation of the player. Beside, dialectal areas may not always coincide with administrative ones. We stuck to these latter ones in the easy game, as people think in terms of those cantonal boundaries, and there was no other simple alternative that would have been a lot more exact that those boundaries. Finally, geographical information represented as points, for instance, cities, can be attractors and introduce a bias in the spatial distribution of the answers.
2. **Getting the relevant socio-demographic information from the users: what should we ask as personal info?** Asking too much information from the player about his/her linguistic history may be frightening or boring. At the same time, information that is not precise enough might render an analysis more difficult. Thus, one has to balance between not getting enough information if it is too personal, and not getting the relevant data if it is too general.
3. **Population of informants: do we get even distribution of participants?** Such online surveys, even relayed by popular media, ineluctably bring up a younger and more connected population. Another consequence is that towns are overrepresented compared to countryside areas. This problem of an uneven distribution of players in terms of age and location needs to be addressed.
4. **Reliability of data: is everyone who takes part of the game doing it seriously?** Closer observations of the data showed some types of players that should be

certainly discarded, but the criteria of removal need to be set carefully; for example, users that perform exceptionally badly (which could be unmotivated players, children or people with a very low knowledge of the dialectal variety, or elderly people struggling with the technological aspect), or users with a very high score (who could be cheaters aiming to get points and a high ranking in the leaderboard, or linguistic experts).

The aim of the paper being a presentation of the crowd-sourcing framework, some complementary data and further analyses of the results are necessary, assessing in more details the results as well as the benefits and drawbacks of such crowdsourced linguistic data.

7. Acknowledgements

This research is supported by the Swiss National Science Foundation as Agora project n°164811. We thank the GISLab team from URPP Language and Space (Research Priority Program) and the SIVIC team (Scientific Visualisation and Visual Communication) of the University of Zurich, Switzerland.
Audio source: for the samples from Grisons, we thank Hanna Ruch who recorded speakers from there for her project "Akkommodation in Dialektkontaktsituationen" financed by the URPP Language and Space; for the ones from Fribourg we thank Ivan Schmutz from the Sensler Museum in Tafers; for the ones from Aargau we thank Nicole Studer-Joho and Dieter Studer-Joho. The samples from Appenzell-Innerrhoden were recorded by Raphael Tandler from the dindialaekt-team.

8. References

Avanzi, M., Béguelin, M.-J. Diémoz, F. (2015). « OFROM – corpus oral de français de Suisse romande, v. 2.2 », Ms, Université de Neuchâtel, http://www.unine.ch/ofrom

Avanzi, M., C. Barbet, J. Glikman, J. Peuvergne (2016). Présentation d'une enquête pour l'étude les régionalismes du français. In: Actes du 5ème congrès mondial de linguistique française (CMLF). Tours, France, 1-15.

Avanzi, M. and Boula-de-Mareuil, Ph. (2017). Perceptual identification of regional French accents in (northern) France, Belgium and Switzerland, *Journal of Linguistic Geography*, 5(1):17-40.

Baker, Wendy, David Eddington and Lyndsey Nay (2009). Dialect identification: The effects of region of origin and amount of experience. *American Speech*, 84(1): 48-71.

Cook, M., J. Barker & Lecumberri, M. L. G. (2013). Crowdsourcing in Speech Perception. In Eskénazi, M., Levow, G.A., Meng, H., Parent, G. & Suendermann, D. (eds), *Crowdsourcing for Speech Processing: Applications to Data Collection, Transcription and Assessment*. Hoboken: John Wiley & Sons, 137-172.

Möller, R., S. Elspaß (2015): Atlas zur deutschen Alltagssprache. In: Kehrein, R., A. Lameli, S. Rabanus (eds.): Regionale Variation des Deutschen – Projekte und Perspektiven. Berlin, Boston: de Gruyter, 519-540.

Ferguson, Charles A. 1959. "Diglossia". *Word* 15: 324–340.

Glaser, Elvira and Michele Loporcaro (eds.) (2001). *Stimmen der Schweiz*. Frauenfeld: Huber. 1 book and 2 CDs.

Goldman J.Ph. et al (2018) Crowdsourcing Regional Variables and Automatic Geolocalisation of Speakers of European French, LREC Conference, Miyasaki, Japan.

Hotzenköcherle, Rudolf. 1984. *Die Sprachlandschaften der Schweiz*. Ed. N. Bigler, R. Schläpfer and R. Börlin. Aarau/Frankfurt: Sauerländer.

Hundt, Marianne, Raphael Tandler and Karina Frick. Localization and perception of Swiss German dialects: comparing performance and judgement data. Forthcoming 2018.

Krefeld, T., S. Lücke (2014). VerbaAlpina - Der alpine Kulturraum im Spiegel seiner Mehrsprachigkeit. In Ladinia XXXVIII, 189-211.

Leemann, A., M.-J. Kolly, R. Purves, D. Britain, E. Glaser (2016). Crowdsourcing language change with smartphone applications. PLOS ONE.

Phonogrammarchiv Zürich (eds.) (2001). *SDS-Phonogramme*. 8 CDs + 4 CD booklets.

Samardžić, Tanja, Yves Scherrer and Elvira Glaser (2016). *ArchiMob – A Corpus of Spoken Swiss German*. Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Online.

Siebenhaar, Beat and Aflred Wyler. 1997. *Dialekt und Hochsprache in der deutschsprachigen Schweiz. Zürich: Pro Helvetia.*

Sprachatlas der Deutschen Schweiz. (2017). Bern (I-IV), Basel:Francke (VII-VIII).

Vaux, B., S. Golder. 2003. *The Harvard Dialect Survey*. Cambridge, MA: Harvard University Linguistics Department.

Vereinigung Archimob (eds.) (2000). *Archive der Zeit des zweiten Weltkrieges in der Schweiz: 383 deutschsprachige Zeugen*.

Phonetically Balanced Code-Mixed Speech Corpus for Hindi-English Automatic Speech Recognition

Ayushi Pandey[1], B M L Srivastava[2], Rohit Kumar[3]*, B T Nellore[1], K S Teja[4]*, S V Gangashetty[1]
IIIT-Hyderabad[1], Microsoft Research[2], NIT Patna[3], MIT Manipal[4]
ayushi.pandey@research.iiit.ac.in, t-brsriv@microsoft.com, svg@iiit.ac.in

Abstract

The paper presents the development of a phonetically balanced read speech corpus of code-mixed Hindi-English. Phonetic balance in the corpus has been created by selecting sentences that contained triphones lower in frequency than a predefined threshold. The assumption with a compulsory inclusion of such rare units was that the high frequency triphones will inevitably be included. Using this metric, the Pearson's correlation coefficient of the phonetically balanced corpus with a large code-mixed reference corpus was recorded to be 0.996. The data for corpus creation has been extracted from selected sections of Hindi newspapers.These sections contain frequent English insertions in a matrix of Hindi sentence. Statistics on the phone and triphone distribution have been presented, to graphically display the phonetic likeness between the reference corpus and the corpus sampled through our method.

Keywords: code-mix speech, phonetic balance, newspaper corpus

1. Introduction

Code-mixing is a frequently encountered phenomenon in day-to-day natural language communication, especially in multilingual and bilingual communities. Code-switching is considered to be the phenomenon of alternating languages at the sentential or clausal level, and code-mixing is the word-level insertions from one language into the sentential frame of another. The phenomenon is particularly prevalent in speech communities where the native language and medium of education are recognized as two separate languages. According to the census of 2001, 12.1% of the speakers in India are speakers of English as their second or third language. Additionally, the popularity of English in social media, print media, and also entertainment make English widely accessible to most such bilingual speakers. The ubiquitous prestige associated with English in the diglossic Indian situation also motivates Indian bilinguals to show abundant code-mixing and code-switching patterns between English and other regional languages. The widespread usage and growth of this phenomenon of code-mixing mandates a shift in paradigm from monolingual automatic speech recognition (ASR) studies into code-mixed speech recognition. However, computational studies for both textual and speech processing of code-mixing suffer from a sincere disadvantage: lack of data.

In this paper, we present a Phonetically Balanced Code Mixed (PBCM) speech corpus, sampled from a standardized code-mixed text corpus, the Large Code Mixed (LCM) corpus. An optimal text selection procedure has been used to extract 6,126 utterances from the LCM. The PBCM corpus is currently in the process of being recorded and post-processed for speech recognition purposes at IIIT-Hyderabad.

The paper is organized as follows: Section 2 describes some popular methods in corpora creation, and also mentions the development of code-mixed corpora for various language pairs. Section 3 details the procedure of optimal text selection that we employed to design the PBCM corpus. Section 4 describes the recording procedure, and the progress of speech recording so far. Section 5 presents the conclusion, and Section 6 discusses some future directions.

2. Prevalent methods in corpus design

It is popularly believed that the success of the recognition and/or synthesis system depends significantly on the quality of the speech corpus. Careful attention therefore, has been paid to designing corpora that ensure a phonemic distribution appropriate for training and testing of the system. Ensuring minimal redundancy in phonetic coverage is also crucial to optimize the time consumed in post-processing. From a large and usually diverse textual database, a set of either phonetically rich or phonetically balanced sentences are selected. Phonetically rich sentences (Radová and Vopálka, 1999) contain an approximately uniform distribution of all phonemes in the language. Phonetically balanced sentences, on the other hand, represent the frequency distribution of phonemes proportionate to the "natural" phonemic distribution in the concerned language. For a phonetically transcribed corpus, the *add-on* procedure is a popular method (Falaschi, 1989). The sentence with a frequency distribution score proportionate to that of the already selected sentences gets added on to the corpus. Corpora designed for speech recognition studies require a context-sensitive phone; a triphone or another subword unit containing sequence information. For synthesis systems, corpora must contain adequate distribution of word-joins, in addition to a phonetic coverage. It is also common to optimize phonetic coverage based on a suprasegmental feature vector such as lexical stress, pitch, prosody etc (Black and Lenzo, 2001). Santen et al emphasize the importance of the preparedness of a system towards rare phonetic units (Van Santen and Buchsbaum, 1997) . To optimize coverage of all phonetic units, ASR studies

*participated in project while interning with IIIT-Hyderabad

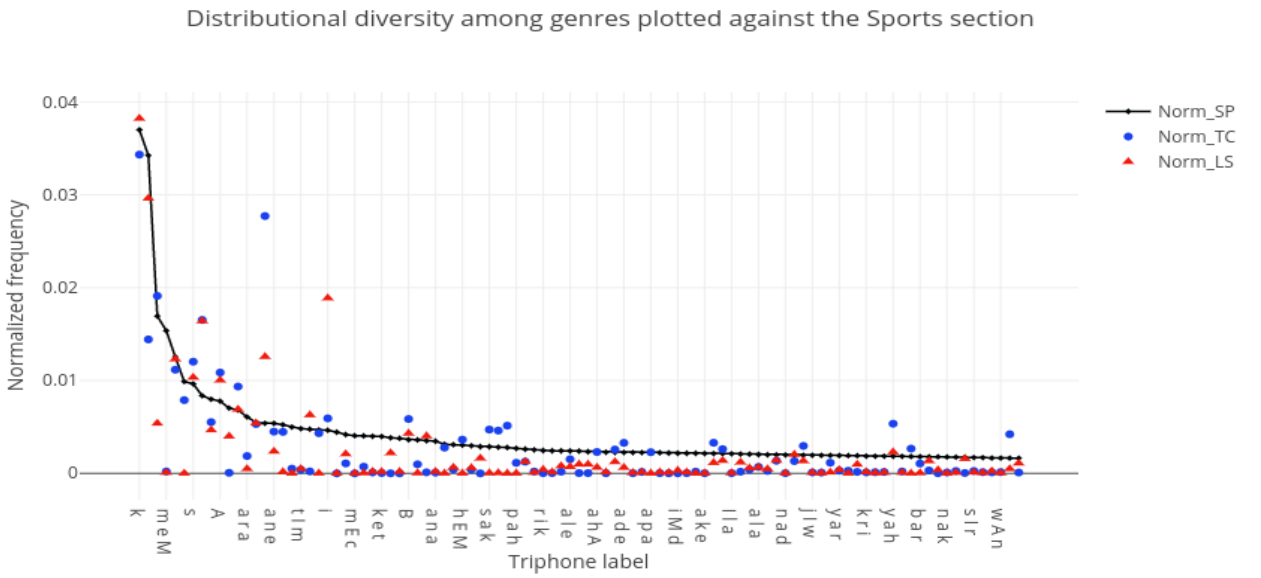


Figure 1: Plot between the disproportionate triphone coverage across two genres (Lifestyle and Technology), plotted against the third (Sports).

benefit from a greedy sentence selection approach with weighted frequencies of triphones, where the weights are the inverse of frequencies. This ensures an inclusion of rare phones in the corpus. (Van Santen and Buchsbaum, 1997). In India, there has been consistent effort to develop both phonetically rich and phonetically balanced corpora for Indian languages. (Kumar et al., 2005), (Godambe and Samudravijaya, 2011), (Arora et al., 2010), (Samudravijaya et al., 2000), (Upadhyay and Riyal, 2010) However, large-scale development of code-mixed corpora still needs attention. There have been several attempts to create speech corpora for language pairs like Mandarin-English, Cantonese-English, Frisian-Dutch, Swahili-English and so on.(Yılmaz et al., 2016), (Chan et al., 2005), (Lyu et al., 2015), (Lyu et al., 2010) (van der Westhuizen and Niesler, 2016), (Kleynhans et al., 2016) As this research field remains in the nascent stage of investigation, a read speech corpus can provide insightful contribution into modeling the acoustic properties of code mixing. A corpus designed in such a manner could offer enormous control on the lexical content, optimal phonetic coverage, choice of speakers, recording environments and reduce the dependence on post-processing. However, one of the largest challenges in approaching the development of an large vocabulary read speech corpus, is the lack of standardized code-mixed text data. The following section describes our approaches towards selecting standardized textual material which reflects patterns of Hindi-English code-mixing in print media.

3. Design of the data corpus

Conversational communication between bilingual speakers represents the dynamic nature of code-mixing in nearly all its entirety. However, there are large sections of print media now that employ recurrent patterns of code-mixing, if not switching. Columns specifically dedicated to content like technology, sports, gadgets and fashion trends show frequent word-level English embeddings in the matrix of a Hindi sentence.

Lexical diversity in phonetic coverage is also recognised as a major concern in corpora design, because the coverage of the recognition unit (triphone, syllable etc) may differ significantly from domain to domain. While monolingual corpora achieve this diversity by selecting portions from various genres, designing a code-mixed corpus requires selections that are exhaustive in English insertions. 1 displays the distributional diversity among the genres of the LCM corpus. As a first step, a large body of data was scraped from three sections, namely Gadgets and Technology, Lifestyle and Sports from the newspapers DainikBhaskar (http://epaper.bhaskar.com/) and Sanjeevani (http://www.sanjeevnitoday.com/). The following example represents the word level English insertion in the matrix of a Hindi sentence.

Example:
अनहैल्थी फूड्स को अधिकतर अवॉइड करना चाहिए ।
Gloss:
[unhealthy-ENG] [foods-ENG] [case marker-HIN] [avoid-ENG] [mostly-HIN] [do-HIN] [should-HIN]

Translation:
One should mostly avoid unhealthy foods.

Here, the English insertion has been transcribed in a matrix sentence of Devanagari. The newspaper corpus contains both English words transcribed in Devanagari, as in the example above, but also a sizeable amount of English words in their Roman transcriptions. The size of the scraped corpus is 46,595 sentences and it has been named the Large Code Mixed (LCM) Corpus. The development

of a Phonetically Balanced Code-Mixed Corpus will be detailed in the succeeding paragraphs.

3.1. Sampling corpus through triphone frequency

Triphone, as a recognition unit has been given primary importance in development of most ASR corpora. The primary reason for this consideration is the sensitivity of triphone towards both its preceding and the succeeding context. To obtain an optimal selection of sentences, the corpus needed to be balanced not only in a set of unique phones, but also the contexts that they occurred in.
A common phonetic scheme was required to cover all the possible contexts in this combination of scripts. As a large section of the vocabulary was transcribed in Devanagari, the WX notation [1] was chosen to develop a consistent phonetic representation of the entire corpus. The following paragraphs detail the formation of a bilingual dictionary and a combined phoneset.
The design on the optimal text selection was created using the following steps:

1. *Grapheme to phoneme (Roman):* Phoneme sequences for unique Roman words in the LCM corpus were generated using a grapheme to phoneme (G2P) converter trained using the CMUdict sequence-to-sequence (Yao and Zweig, 2015) model. Using the conversion map described in Table 1, the ARPABet [2] characters were transformed into their respective WX counterparts.
2. *Grapheme to phoneme (Devanagari):* Phoneme sequences for unique Devanagari words in the LCM corpus were generated by converting them to their corresponding WX notation (Bharati et al., 1995), (Bhat, 2016). . Normalization of this phoneset was achieved through pruning out the word-final schwa, and other special characters such as "nukta".
3. *Bilingual dictionary:* Concatenating the two phonetic dictionaries generated in steps 1. and 2., a bilingual pronunciation dictionary was created. The total number of unique phones in the corpus, derived from the combination of WX and the ARPAbet-adapted WX was recorded to be 65.
4. *Preprocessing :* As a pre-processing step for creation of a read-speech corpus, sentences only sentences of length 5-15 words were selected. Punctuations (except '.',' |', ',') were pruned out. Web addresses were replaced by the single word "website". Numerals were converted to their Hindi word expansion equivalents. The size of this cleaned corpus was 23,389 sentences.
5. *Triphone coverage:* The cleaned and pre-processed corpus (in step 4.) was converted to its corresponding phonetic representation (mapped from the dictionary generated in step 3.). Word-internal triphones were collected and arranged based on the descending order of their frequency of occurrence. To ensure the coverage of rare phones, all the unique sentences that contained words that were composed of the triphones lower in frequency than the threshold, were selected and added to the corpus. The threshold was set to 10.
6. *Correlation computation:* After this selection process of sentences, a metric that compared the true distribution in the sampled corpus with the LCM was required. Unique phones (monophones) from both the LCM corpus and the PBCM corpus were collected as vectors, and a Pearson's correlation coefficient was computed.

[1] https://en.wikipedia.org/wiki/WX_notation
[2] https://en.wikipedia.org/wiki/ARPABET

Table 1: ARPABet to WX notation conversion table

ARPABet	WX notation	Devanagari
AA	A	आ
AE	E	ऐ
AH	a	अ
AO	O	औ
AW	a u	आउ
AY	a i	आ ई
B	b	ब
CH	c	च
D	d	ड
DH	x	द
EH	e	ए
ER	a r	अ र
EY	e	ए
F	P	फ
G	g	ग
HH	h	ह
IH	i	आ
IY	I	ई
JH	j	ज
K	k	क
L	l	ल
M	m	म
N	n	न
NG	M g	न्ग
OW	o	ओ
OY	O i	औ ई
P	p	प
R	r	र
S	s	स
SH	S	श
T	t	ट
TH	W	थ
UH	u	उ
UW	U	ऊ
V	v	व
W	v	व
Y	y	य
Z	j	ज
ZH	j	ज

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x^2)][n\sum y^2 - (\sum y^2)]}} \quad (1)$$

Equation (1) describes the Pearson's correlation r, where n is the number of pairs to be scored, x is the value contained in the first variable (in our case, the phonetic distribution of the LCM corpus), and y is the value contained in the second variable (phonetic distribution of the PBCM corpus).

The Phonetically Balanced Code-Mixed (PBCM) corpus of 6,126 sentences was created through the culmination of steps described above. Table 1 describes the conversion mapping between the ARPABet and WX notation. Phonemes that lacked a direct equivalent were either approximated to their closest sounding phoneme in WX, or represented as a combination of more than one phone.

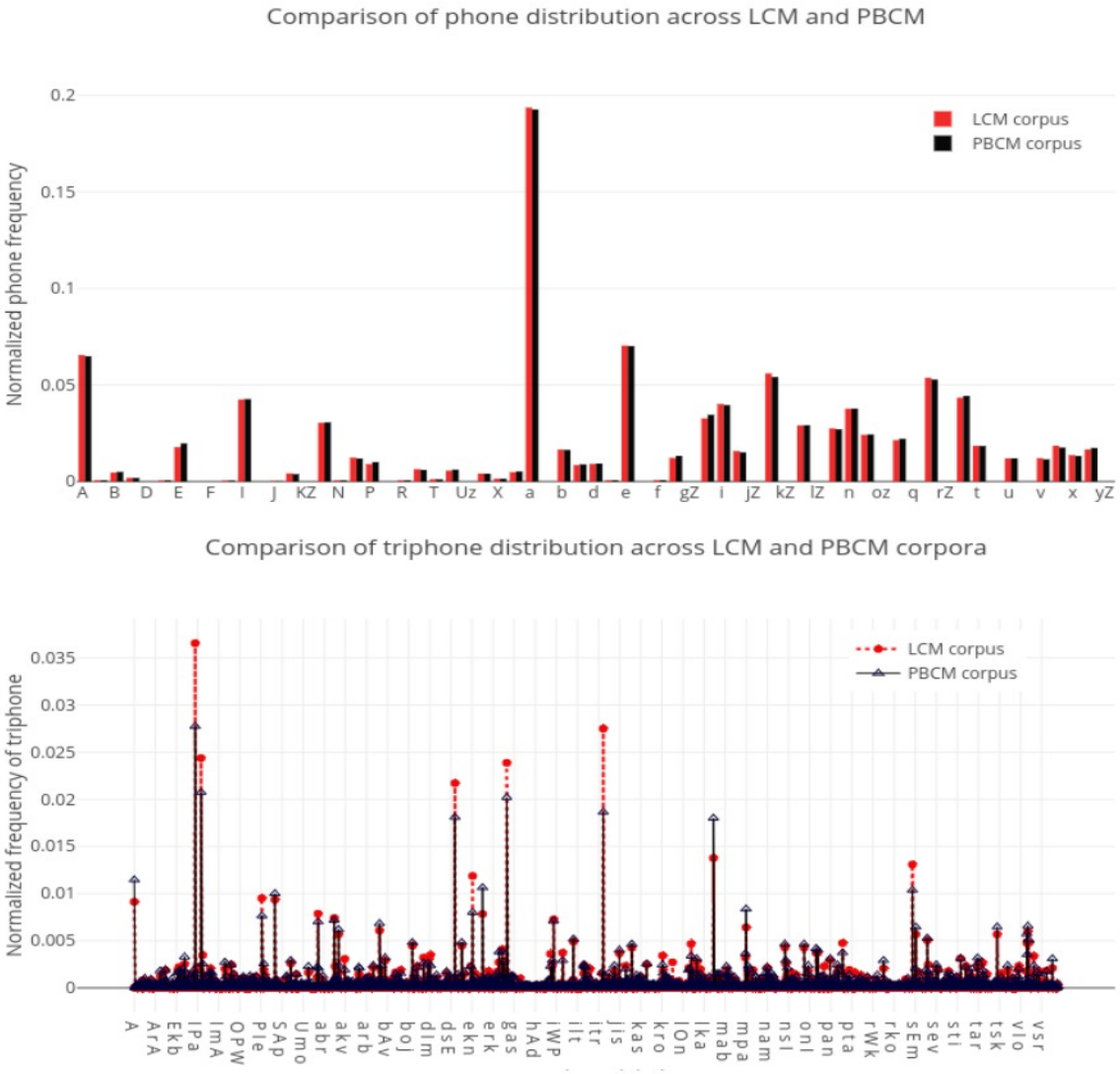


Figure 2: The top panel traces the sum-normalized frequency for phones in the LCM and the PBCM corpus, against the phoneme label. A similar distribution has been described for the triphones. The dotted red curve represents the sum normalized phone/triphone distribution in the LCM corpus, whereas the solid black line represents the same in the PBCM corpus. Both graphs are plotted with the common triphone threshold = 10.

The Pearson's correlation coefficient between the LCM and the PBCM corpora was found to be 0.996. A high correlation value indicates a proportionate distribution of phones between the sampled corpus (PBCM), and the reference corpus (LCM). Figure 1 displays the phone and triphone coverage in the LCM and PBCM corpora. Table 2 displays the comparison between the corresponding sentence, word, triphone and phone units. From Step 4, we observe that the size of the LCM corpus has considerably reduced. This results in the loss of certain phones, giving the phone distribution between LCM and PBCM an imbalance (Table 2, col. 4). It can be observed that a high correlation with the LCM is maintained despite having a non-equal phonetic distribution in the PBCM. We believe that this is because phones that show no occurrence (frequency: 0) in the PBCM are also recorded to be fairly low in frequency in the LCM. It may be observed that a large amount of code-mixed speech utterances have been removed in this way. The purpose of the corpus so created is to be able to achieve maximal phonetic coverage in the least possible amount of recordings.

4. The recording procedure

After the sentence selection procedure is completed, the next step is to conduct the actual recordings. This section presents a detailed description of volunteer speakers, recording environment and the equipment setup utilized for recording.

4.1. Description of speakers

Speech recordings are being collected from 100 volunteer speakers (50 male and 50 female), who are each a native speaker of Hindi and received education in English medium schools. All speakers are students of IIIT-Hyderabad. The age range of these speakers was between 18-35 years. The PBCM corpus is equally divided among the speakers, so that every speaker records around 62 sentences.

4.2. Recording environment and equipment

The recording of the speech utterances of the PBCM corpus are being conducted in a soundproof voice recording studio (Speech and Vision Lab, IIIT, Hyderabad). The recordings are

Table 2: Distribution across the LCM and PBCM

	Sentences	Words	Triphones	Phones
LCM	46,595	30,578	11,370	65
PBCM	6,126	8,683	6,599	57

administered through the OctaCore speech processing software, with a high fidelity noise free microphone. Wavfiles are being recorded through the open-source, cross-platform audio software, Audacity. The recordings are sampled at 48kHz and recorded at 24-bit resolution.

Each volunteer speaker is instructed to maintain a distance 5-6 inches from the microphone. Speakers record 20 sentences in one pass, after which they were given a water-break and vocal rest of 2-5 minutes. Before each recording session, the speaker is *primed* by having the sentences read out aloud to them, in order to minimize hesitation while speaking. After every 100 sentences, the speaker is given a vocal rest for 10 minutes. So far, 78 speakers (40 male and 38 females) have been recorded.

4.3. Post-processing of audio files

At this stage, the data is completely unsegmented, which means that there is only wavefile per speaker. After completing 100 speakers and exhausting all the utterances of the PBCM speech corpus, the data will be post-processed as a final step. A long sound file of 62 utterances will be manually split into one sound file per sentence format, using Praat. Non-verbal sounds and repetitions will also be manually removed, and only noise-free sentences will be compiled. For preparing the data suitable for use for speech recognition, we plan to give each sound file a unique ID, which will contain the speaker information and the serial number of recording. A silence of 1 second will be appended to each sound file, both before and after the utterance. The sound files, initially recorded at 48 kHz and 24-bit resolution, will also downsampled to 16 kHz and a 16-bit resolution.

5. Conclusion

The paper presents a phonetically balanced read speech corpus for code-mixed Hindi-English automatic speech recognition. The PBCM corpus has been sampled from a Large newspaper Corpus (LCM), which contains rich lexical insertions from English in a matrix of Hindi sentences. The inclusion of rare triphones in the sampled corpus has resulted in a high phonetic coverage (correlation: **0.996**), even with a small number of sentences. To the best of our knowledge, the PBCM can be safely proposed as one of the first phonetically balanced corpus of code-mixed speech in an Indian language pair. Recordings through the contribution of 100 Hindi-English bilinguals is aimed for, of which 78 speakers have been recorded. Once post-processed, the PBCM corpus will be made available for research and related purposes.

6. Future direction

We have observed that increasing the threshold value increases the number of sentences, while maintaining a steadfast correlation. We hope to provide an adaptive measure for selecting sentences by choosing the appropriate threshold within a given range. We also hope to compare the results of our proposed metric for selecting sentences, with the already existing methods of phonetic balance in terms of optimum corpus size, and correlation measure.

7. References

Arora, S., Saxena, B., Arora, K., and Agarwal, S. (2010). Hindi asr for travel domain. In *Oriental COCOSDA*.

Bharati, A., Chaitanya, V., Sangal, R., and Ramakrishnamacharyulu, K. (1995). *Natural language processing: a Paninian perspective*. Prentice-Hall of India New Delhi.

Bhat, I. A. (2016). indic-wx-converter. https://github.com/ltrc/indic-wx-converter.

Black, A. W. and Lenzo, K. A. (2001). Optimal data selection for unit selection synthesis. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*.

Chan, J. Y., Ching, P., and Lee, T. (2005). Development of a cantonese-english code-mixing speech corpus. In *Ninth European Conference on Speech Communication and Technology*.

Falaschi, A. (1989). An automated procedure for minimum size phonetically balanced phrases selection. In *Speech Input/Output Assessment and Speech Databases*.

Godambe, T. and Samudravijaya, K. (2011). Speech data acquisition for voice based agricultural information retrieval. In *Proc. Of 39th All India DLA Conference, Punjabi University, Patiala, June*.

Kleynhans, N., Hartman, W., van Niekerk, D., van Heerden, C., Schwartz, R., Tsakalidis, S., and Davel, M. (2016). Code-switched english pronunciation modeling for swahili spoken term detection. *Procedia Computer Science*, 81:128–135.

Kumar, R., Kishore, S., Gopalakrishna, A., Chitturi, R., Joshi, S., Singh, S., and Sitaram, R. (2005). Development of indian language speech databases for large vocabulary speech recognition systems. In *Proceedings of SPECOM*.

Lyu, D.-C., Tan, T. P., Chng, E., and Li, H. (2010). Seame: a mandarin-english code-switching speech corpus in south-east asia. In *INTERSPEECH*, volume 10, pages 1986–1989.

Lyu, D.-C., Tan, T.-P., Chng, E.-S., and Li, H. (2015). Mandarin–english code-switching speech corpus in south-east asia: Seame. *Language Resources and Evaluation*, 49(3):581–600.

Radová, V. and Vopálka, P. (1999). Methods of sentences selection for read-speech corpus design. In *International Workshop on Text, Speech and Dialogue*, pages 165–170. Springer.

Samudravijaya, K., Rao, P., and Agrawal, S. (2000). Hindi speech database. In *Proceedings of International Conference on Spoken Language Processing*.

Upadhyay, R. and Riyal, M. (2010). Garhwali speech database. *Proceedings of O-COCOSDA*.

van der Westhuizen, E. and Niesler, T. (2016). Automatic speech recognition of english-isizulu code-switched speech from south african soap operas. *Procedia Computer Science*, 81(Supplement C):121 – 127. SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia.

Van Santen, J. P. and Buchsbaum, A. L. (1997). Methods for optimal text selection. In *EuroSpeech*.

Yao, K. and Zweig, G. (2015). Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. *arXiv preprint arXiv:1506.00196*.

Yılmaz, E., van den Heuvel, H., and van Leeuwen, D. (2016). Investigating bilingual deep neural networks for automatic recognition of code-switching frisian speech. *Procedia Computer Science*, 81(Supplement C):159 – 166. SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia.

Chinese–Portuguese Machine Translation: A Study on Building Parallel Corpora from Comparable Texts

Siyou Liu[♥] **Longyue Wang**[♠] **Chao-Hong Liu**[♠]

[♥] School of Languages and Translation, Macao Polytechnic Institute, Macao S.A.R., China
[♠] ADAPT Centre, School of Computing, Dublin City University, Ireland
violetal@ipm.edu.mo
{longyue.wang, chaohong.liu}@adaptcentre.ie

Abstract

Although there are increasing and significant ties between China and Portuguese-speaking countries, there is not much parallel corpora in the Chinese–Portuguese language pair. Both languages are very populous, with 1.2 billion native Chinese speakers and 279 million native Portuguese speakers, the language pair, however, could be considered as low-resource in terms of available parallel corpora. In this paper, we describe our methods to curate Chinese–Portuguese parallel corpora and evaluate their quality. We extracted bilingual data from Macao government websites and proposed a hierarchical strategy to build a large parallel corpus. Experiments are conducted on existing and our corpora using both Phrased-Based Machine Translation (PBMT) and the state-of-the-art Neural Machine Translation (NMT) models. The results of this work can be used as a benchmark for future Chinese–Portuguese MT systems. The approach we used in this paper also shows a good example on how to boost performance of MT systems for low-resource language pairs.

Keywords: Chinese–Portuguese, Low-Resource, Statistical Machine Translation, Neural Machine Translation, Parallel Corpus

1. Introduction

Chinese and Portuguese are widely used by a large amount of people in the world. With the development of economic globalization, communications between Chinese and Portuguese-speaking countries are increasing in a fast path. Translation services between these two languages is becoming more and more demanding. However, Chinese and Portuguese belong to distinct language families (Sino-Tibetan and Romance, respectively) and only a relative much smaller proportion of people have bilingual proficiency of the language pair. Therefore, the use of Chinese–Portuguese MT systems to provide auxiliary translation services between the two sides is highly demanded.

Pivot-based machine translation is a commonly used method when large quantities of parallel data are not readily available for some language pairs. Utiyama and Isahara (2007), Wu and Wang (2007), Bertoldi et al. (2008) investigated phrase-level, sentence-level and system-level pivot strategies for low resource translation in SMT. A pivot language, which is usually English, can bridge the source and target languages and make translation possible. However, the domains of these two are often different and thus results in low performance and even ambiguities.

A few researchers have investigated how to improve the Chinese–Portuguese MT by incorporating linguistic knowledge into the systems. For instance, Wong and Chao (2010) proposed a hybrid MT system combining rule-based and example-based components. Oliveira et al. (2010) explored Constraint Synchronous Grammar parsing for SMT. Lu et al. (2014) and Liu and Leal (2016) focused on specific linguistic phenomena (i.e. present articles and temporal adverbials) in translation. Although NMT has been rapidly developed in recent years (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Bahdanau et al., 2015; Tu et al., 2016), Chinese–Portuguese MT has not received much attention using NMT because training data are not readily enough. Therefore the performance is still low using these state-of-the-art approaches.

To date, there are only a few Chinese–Portuguese parallel corpora available[1] (Tiedemann, 2012). OpenSubtitles2018[2] (Lison and Tiedemann, 2016) has released 6.7 millions Chinese–Portuguese sentence pairs, which are extracted from movie subtitles. These sentences are usually short and simple as most of them are transcripts of conversations in movies, therefore they alone are not suitable to train general-domain MT systems. News-Commentary11[3] contains data in newswire domain. However, there are only 21.8 thousands of sentence pairs and is thus not sufficient to train robust MT models.

To alleviate data scarcity problem, we extracted bilingual data from Macao government websites.[4] Macao government documents, as requested by law, are written and archived in both languages. Domains contained in these documents include international communication, trade exchanges, technological cooperation, etc. In order to build a high-quality parallel corpus, we propose a hierarchical strategy to deal with document-level, paragraph-level and sentence-level alignment. In total more than 800 thousands of Chinese–Portuguese sentence pairs in newswire, law and travelling domains, among others, are curated. Finally, we conducted experiments on Chinese–Portuguese machine translation tasks using both OpenSubtitles2018 and the curated corpus to evaluate the quality of the cor-

[1] http://opus.nlpl.eu.

[2] http://opus.nlpl.eu/OpenSubtitles2018.php.

[3] http://www.casmacat.eu/corpus/news-commentary.html.

[4] Macao is a multi-cultural society in which both Mandarin Chinese and Portuguese are recognized as official languages.



Figure 1: An example from *The Government Information Bureau* website. The texts in the boxes with the same color are parallel data.

pus. The experimental results show that the performance of Chinese–Portuguese MT has significantly improved and outperforms the results using pivot-based method. The contributions of this paper are listed as follows:

- We propose a hierarchical alignment approach to build a large and high-quality general-domain corpus, which in total contains more than 800 thousands of Chinese–Portuguese sentence pairs;
- We evaluate the quality of the curated corpus with MT performance of Chinese–Portuguese MT systems trained on the two large corpora (i.e., OpenSubtitles2018 and ours);
- We investigate both SMT and NMT models and compare them using pivot-based method. The experimental results can be used as Chinese–Portuguese MT benchmark for future work.

The rest of the paper is organized as follows. In Section 2, we introduce our approach to build the Chinese–Portuguese corpus. The experimental results of MT tasks, which are used to evaluate the quality of corpus, are reported in Section 3. Analyses of out-of-vocabulary (OOV) and pivot-based MT are given in Section 4. Finally, Section 5 presents our conclusions and future work.

2. Building a Chinese–Portuguese Corpus

There are a number of Macao websites (e.g. *The Government Information Bureau* website,[5] *The Macao Law*,[6] and *The Government Printing Bureau*[7]) containing bilingual resources. *The Government Information Bureau* website, for example, contains more than 100 thousands of local news articles from the year 2000 to date and more than 80% of the articles are written in both Portuguese and Chinese languages. As shown in Figure 1, the same news is written in a "comparable" way.[8] Although not completely aligned, there are still paragraphs can be aligned to each other. Therefore, these are still good resources to curate parallel corpora. We crawl all similar websites in Macao government website list[9] and we only use *The Government Information Bureau* website for detailed discussions in the rest of the paper.

As shown in Figure 2, we develop an end-to-end system to automatically build our parallel corpus from bilingual websites. However, there are still some challenges: 1) how to identify parallel/comparable news articles (bilingual document alignment tasks); 2) bilingual news articles are not direct translations to each other but written separately by Chinese authors and Portuguese authors of the same story. Therefore these articles are mostly comparable rather than parallel texts (paragraph alignment tasks); 3) Chinese texts are usually written in chronicle style, while its corresponding Portuguese texts written in several sentences. Thus, these sentences are not one-to-one aligned (sentence alignment tasks). In order to obtain high-quality sentence pairs, we propose a hierarchical alignment strategy including document-level, paragraph-level and sentence-level alignment. At each level, we employ hybrid alignment approaches such as rule-based and translation-based. The architecture of our method can be described in a pipeline as follows:

(1) We crawl all accessible web pages from each website and then extract meta-data (e.g. title, author, date and content) using HTML tags;

[5] http://www.gcs.gov.mo.
[6] http://www.macaolaw.gov.mo.
[7] http://www.io.gov.mo.

[8] A Comparable Corpus is a collection of "similar" texts in different languages or in different presentation forms of a language.
[9] https://www.gov.mo/en/about-government/departments-and-agencies/.

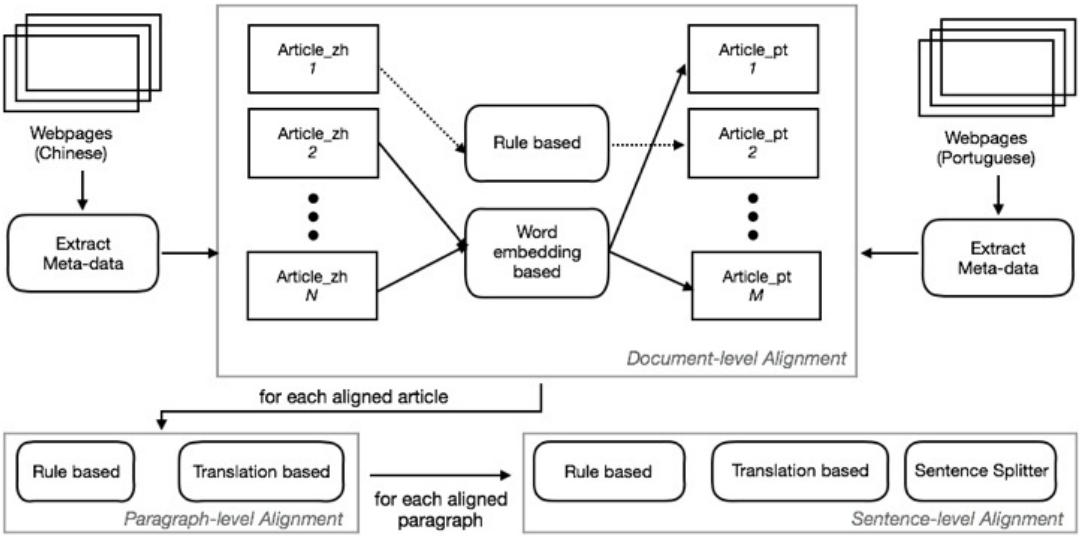


Figure 2: The workflow of building our Chinese–Portuguese corpus.

(2) For document-level alignment, we first use URLs to align portions in articles. For other articles in which the heuristic rules do not apply, we employ a document alignment algorithm that calculates semantic similarity based on word embedding;

(3) For each aligned articles, we align the paragraphs with a simple but effective method: if the number of paragraphs in the two aligned articles is equal, we align all paragraphs one by one in the same order. Otherwise, we employ translation-based alignment algorithm to find parallel paragraphs;

(4) Within each aligned paragraph, we firstly use sentence boundary detection toolkit to split the paragraph into sentences and then do sentence-level alignment, which is similar to the process used in paragraph-level alignment.

2.1. Rule-based Alignment

We found that around 10% of web pages have already been aligned by URL links. As shown in Figure 1, there is usually a language switch button on the top of a web page which can be used to extract its corresponding page in the other language. Thus, we extract these useful links and align documents based on this heuristic rule. For other web pages the rule does not apply, we employ word embedding based alignment approach as described in Section 2.2.

At paragraph- and sentence-level alignment, we propose a simple but effective rules: 1) in each article/paragraph, we count the number of paragraphs/sentences on two sides. If the numbers are equal to each other, we align these paragraphs/sentences one by one in the same order. Otherwise, for example, the number of the source-side sentences is 10 while the number of the target-side is 12, we use translation-based alignment method (in Section 2.3) instead.

Using rule-based methods, we can easily obtain a reliable parallel sub-corpus. Although the corpus size is relatively small, these data are in the same domain. Thus the corpus can still be used to train a MT system for further steps. For instance, we could train a SMT system on the sub-corpus in newswire domain and use the system to translate sentences for translation based alignment method.

2.2. Word Embedding based Alignment

To align articles/documents, we consider the problem as cross-lingual document alignment task (Wang et al., 2012). We employ a document alignment approach using word embedding (Lohar et al., 2016): 1) we initially construct a pseudo-query from a source-language document; 2) and then represent both the target-language documents and the pseudo-query as word vectors to find the average similarity measure between them; 3) finally the word vector based similarity is combined with the term-overlap-based similarity.

The Vector Space Model (VSM) (Salton et al., 1975) is one of the overlap based methods. Each document is represented as a vector of terms. The ith document D_i in target-side is represented as a vector $D_i = [w_{1,i}, w_{2,i}, ...w_{k,i}]$, in which k is the size of the term vocabulary. Here we employ the cosine distance to calculate the similarity between two document vectors:

$$sim(d_i, d_j) = \sum_{k=1}^{N} w_{i,k} \cdot w_{j,k} \sqrt{\sum_{k=1}^{N} w_{i,k}} \cdot \sqrt{\sum_{k=1}^{N} w_{j,k}} \quad (1)$$

where N is the number of terms in a vector, and $w_{i,k}$ and $w_{j,k}$ represent the weight of the ith/jth term in D_i/D_j respectively. Technically, the distance between documents in VSM is calculated by comparing the deviation of angles between vectors. A Boolean Retrieval Model sets a term weight to be either 0 or 1, while an alternative solution is to calculate the term weights according to the appearance of a term within the document collection. To calculate the term weights according to the appearance of a term within the document collection, we use term frequency-inverse document frequency (*TF-IDF*) (Ramos, 2003) as the term-weighting model.

Corpus	Set	$\|S\|$	$\|W\|$		$\|V\|$		$\|L\|$	
			Zh	Pt	Zh	Pt	Zh	Pt
Opensub	Train	6.51M	48.14M	53.45M	0.40M	0.26M	7.39	8.21
	Tune	2.07K	15.43K	18.22K	3.16K	2.92K	7.46	8.81
	Test	2.16K	1243K	17.49K	2.69K	2.94K	5.74	8.08
Our Corpus	Train	0.84M	17.02M	22.49M	0.21M	0.23M	20.36	26.90
	Tune	1.00K	19.90K	30.83K	3.88K	4.20K	19.90	30.83
	Test	1.00K	26.79K	38.60K	4.68K	4.99K	26.79	38.60

Table 1: Number of sentences ($|S|$), words ($|W|$), vocabulary ($|V|$), and averaged sentence length ($|L|$) in the corpus. K stands for thousands and M for millions.

We also use the vector embedding of words and incorporate them with the VSM approach as mentioned above to estimate the semantic similarity between the source-language and the target-language documents. In practice, we indexed articles in both sides and then generate a query for each source-side article. Then we use a Chinese–Portuguese SMT system (training data are extracted using the method in Section 2.1) to obtain translated queries.

2.3. Translation based Alignment

Through exploring various sentence-alignment methods (e.g. length-based, dictionary-based), we found that translation based alignment is a robust approach especially for comparable data (Sennrich and Volk, 2010; Sennrich and Volk, 2011). The idea is to use machine translated text and BLEU as a similarity score to find reliable alignments which are used as anchor points. The gaps between these anchor points are then filled using BLEU-based and length-based heuristics.

We use this method to align unaligned paragraphs and sentences. A Chinese–Portuguese SMT system (training data are extracted using the method in Section 2.1) is used to obtain translated paragraphs/sentences.

2.4. Machine Translation

MT is a sequence-to-sequence prediction task, which aims to find for the source language sentence the most probable target language sentence that shares the same meaning. We can formulate SMT as: $\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$ (Brown et al., 1993), where $\mathbf{x}$ and $\mathbf{y}$ are sentences in source and target sides, respectively. $\hat{\mathbf{y}}$ denotes the translation output with the highest translation probability. $p(\mathbf{y}|\mathbf{x})$ is usually decomposed using the log-linear model:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \frac{\exp(\sum_{i=1}^{I} \lambda_i h_i(\mathbf{x}, \mathbf{y}))}{\sum_{\mathbf{y}'} \exp(\sum_{i=1}^{I} \lambda_i h_i(\mathbf{x}, \mathbf{y}'))} \quad (2)$$

where $h_i(\cdot)$ indicates the translation feature and λ_i is its corresponding optimal weight, which is learned by maximizing with a development set. I indicates the total feature number. We employ phrase-based SMT in our experiments. NMT is a new paradigm for MT in which a large neural network is trained to maximize the conditional likelihood on the bilingual training data. It directly models the probability of translation from the source sentence to the target sentence word by word (Kalchbrenner and Blunsom, 2013):

$$P(\mathbf{y}|\mathbf{x}) = \prod_{j=0}^{N} P(y_j|y_{<j}, \mathbf{x}) \quad (3)$$

in which given $\mathbf{x}$ and previous target translations $y_{<j}$ $(y_1, ..., y_{j-1})$, we need to compute the probability of the next word y_j ($j \in \{1, ..., N\}$). We employ both RNNsearch (Sutskever et al., 2014) and Transformer (Vaswani et al., 2017) architectures in our experiments.

3. Experiments

3.1. Data

The general domain parallel corpus is built using the approaches introduced in Section 2. We randomly sampled 1000 sentences and found that the alignment accuracy is over 94%, indicates the corpus could be used for MT training. Regarding the document-level alignment, we use FaDA toolkit[10] and for the translation based alignment, we employ Bleualign[11]. To pre-process the raw data, we apply a series of procedures (Wang et al., 2016b) including: full/half-width conversion, Unicode conversation, simplified/traditional Chinese conversion, punctuation normalization, English/Chinese tokenization and sentence boundary detection, letter casing and word stemming, etc. For Portuguese tokenization and sentence splitting, we use Moses toolkit[12]. We randomly select one thousand (for the curated corpus) and approximately two thousands (for OpenSubtitles2018) of sentences as development and test sets; the numbers of sentences selected reflect the average sentence lengths of the two corpora. Table 1 lists the statistics of our corpus and OpenSubtitles2018 (*Opensub*). Movie subtitle corpus is much larger than ours, however, its sentences are mostly simple and short.

3.2. Setup

We carry out our experiments on both Chinese-to-Portuguese and Portuguese-to-Chinese translation directions. We investigate various MT models: phrase-based SMT, RNNsearch NMT and Transformer NMT. We used

[10] https://github.com/gdebasis/cldocalign.

[11] https://github.com/rsennrich/Bleualign.

[12] https://github.com/moses-smt/mosesdecoder.

Corpus	System	Dev	Test
Opensub	SMT	15.05	6.73
	RNNsearch	13.37	13.34
	Transformer	17.00	17.43
Ours	SMT	33.78	27.42
	RNNsearch	31.36	24.74
	Transformer	32.55	25.11

Table 2: Results of Chinese-to-Portuguese translation.

Corpus	System	Dev	Test
Opensub	SMT	11.62	5.11
	RNNsearch	10.71	11.76
	Transformer	12.78	12.43
Ours	SMT	26.14	19.29
	RNNsearch	25.13	17.82
	Transformer	26.23	18.68

Table 3: Results of Portuguese-to-Chinese translation.

case-insensitive 4-gram NIST BLEU metrics (Papineni et al., 2002) for evaluation, and *sign-test* (Collins et al., 2005) for statistical significance test.

SMT We employ Moses (Koehn et al., 2007) to build phrase-based SMT model. The 5-gram language model are trained using the SRI Language Toolkit (Stolcke, 2002). To obtain word alignment, we run GIZA++ (Och and Ney, 2003) on the training data together with News-Commentary11 corpora. We use minimum error rate training (Och, 2003) to optimize the feature weights. The maximum length of sentences is set as 80.

RNNsearch We use our re-implemented attention-based NMT system, which incorporates dropout (Hinton et al., 2012) on the output layer and improves the attention model by feeding the most recently generated word. We limited the source and target vocabularies to the most frequent 50K and 30K words in Chinese and Portuguese, covering more than 97% of the words in both languages. Each model was trained on sentences of lengths up to 80 words with early stopping. Mini-bataches were shuffled during processing with a mini-batch size of 80. The word-embedding dimension was 620 and the hidden layer size was 1,000. We trained for 20 epochs using Adadelta (Zeiler, 2012), and selected the model that yields best performance on the validation set.

Transformer We use our re-implemented Transformer NMT system. Most parameters are same as RNNsearch model except that 1) the encoder and decoder are both composed of a stack of 6 identical layers; 2) the hidden layer size is 512; 3) the batch size is 4096 tokens and 4) we use two GPUs for training.

3.3. Results

Table 2 and Table 3 show the performances of different MT systems on Chinese-to-Portuguese and Portuguese-to-Chinese, respectively.

Chinese-to-Portuguese Translation On *Opensub*, SMT system only obtain 6.73 in BLEU score. NMT systems outperform it by 6–10 BLEU scores. We think SMT model is weak in translating informal domain (e.g. spoken domain) data, while distributed word representations can facilitate the computation of semantic distance. With our corpus, the SMT system can achieve 27.42 in BLEU while the best NMT model (Transformer) is around 2 points lower than SMT model. It is not surprising that the performance of NMT models have not surpassed that of traditional SMT (25.11 vs. 27.42). There are three main reasons: 1) the vocabulary size of *Opensub* is very large, which results in a lot of out-of-vocabulary words (OOVs) in NMT training; 2) *Opensub* is of small scale and the corpus is not big enough for NMT models to learn some general translation knowledge; 3) the sentences in our corpus is much longer than that in *Opensub*. We will discuss these in Section 4. Furthermore, Transformer is the best one among NMT models in both corpora. The Transformer model is usually better than RNNsearch even for low-resource MT.

Generally, the BLEU scores using *Opensub* (i.e. OpenSubtitle2018 corpus) are much lower than the scores with our corpus. For example, the performance on *Opensub* is 6–17 in BLEU while it is 24–27 points on our data. Because sentences in movie subtitles are usually compact (i.e. short sentences with rich information) and contain multiple expressions, it results in a number of problems such as ambiguities for MT.

Portuguese-to-Chinese Translation As shown in Table 3, the translation performances of different systems on inverse direction are similar to those in Table 2. For example, NMT models still perform better than SMT model on *Opensub* while worse on our general domain corpus. However, with our corpus, the performance of Transformer is close to that of SMT (i.e. -0.61 in BLEU).

The BLEU scores in the Chinese-to-Portuguese direction are much higher than those in the inverse direction. Taking the performance on our data for instance, Chinese-to-Portuguese MT systems can usually achieve around 25 in BLEU whereas the systems for the inverse direction can only obtain about 18. It indicates that generating fluent and adequate Chinese translations is a more difficult task to MT systems.

The performance of Chinese–Portuguese machine translation is relatively lower; Chinese–English systems can usually achieve 36–40 in BLEU on NIST test sets. One of the reasons is that the Chinese–Portuguese language pair is low-resource. Another reason is the great distance between Chinese and Portuguese in terms of language families; there are extensive differences in syntax, semantics and discourse structures.

4. Analysis

In this section, we first discuss the OOV problem observed in the experimental results on our corpus (as described in Section 3.3). We also compare training directly using the

parallel corpus we curated with the pivot-based method, which is a common approach for low-resource MT (as described in Section 1).

4.1. Out-of-Vocabulary

As shown in Table 1, vocabulary size is very big on both corpora. However, NMT models typically operate with a fixed vocabulary, which results in the OOV problem. This might contribute to the under-performance of NMT models compared SMT as observed in our experimental results.

Joint byte-pair encoding (BPE) (Sennrich et al., 2016) is a simpler and more effective method to handle the OOV problem. It encodes rare and unknown words as sequences of subword units. We use the BPE toolkit[13] to process our corpus and train an NMT system on this processed data. The procedures are as follows. The Portuguese and Chinese data are first pre-processed using the same method introduced in Section 3.1. We then train single BPE models on tokenized/segmented both Portuguese and Chinese sides. Finally, we use BPE models to segment the sentences into subword units.

After 59,500 joint BPE operations, the network vocabulary sizes are reduced to 67K and 59K for Chinese and Portuguese sides, respectively. Compared with the original vocabulary sizes (i.e. 210K and 230K), BPE method has significantly alleviate the problem of OOV. We train a new Chinese-to-Portuguese NMT model with Transformer on BPE-based data. As shown in Table 4, the NMT model trained on BPE data increase 1.44 points in BLEU compared that without BPE. The performance is getting closer to that of SMT, which shows using BPE with subword units does deal with OOV problem to some extent.

System	Dev	Test
SMT	33.78	27.42
Transformer	32.55	25.11
Transformer + BPE	33.96	26.55

Table 4: Comparisons of results using SMT and NMT trained with/without BPE (Chinese-to-Portuguese).

4.2. Pivot-based MT

As discussed in Section 1, pivot method is commonly used for low-resource MT. To show that increasing parallel data is still essential to improve low-resource MT, we also compare MT models trained with parallel corpus of direct translation pair, with the pivot-based models.

We build four Transformer NMT models on large Chinese–English[14] and English–Portuguese[15] parallel corpora: Chinese-to-English, English-to-Portuguese, English-to-Chinese, Portuguese-to-English NMT models. Taking English as the pivot language, we firstly use Chinese-to-English model to translate the Chinese input into English. Secondly, we use English-to-Portuguese model to translate the machine translated English sentences into Chinese output. For the Portuguese-to-Chinese translation direction, the experiments are administered in the same manner.

As shown in Table 5, the pivot-based systems perform poor than those trained on direct parallel corpora. For instance, Portuguese-to-English-to-Chinese ("PT-EN-ZH") system obtains only 11.29 in BLEU on our test set which is 7.39 points lower than our Transformer model. It indicates that increasing the amount of parallel data does help improve low-resource MT systems.

Direction	Corpus	Dev	Test
ZH-EN-PT	Opensub	8.23	10.52
	Ours	15.38	14.60
PT-EN-ZH	Opensub	8.88	10.33
	Ours	12.31	11.29

Table 5: Results of pivot-based translation.

5. Conclusions and Future Work

In this paper we described our methods to build a large Chinese–Portuguese corpus. Despite both Chinese and Portuguese are populous languages, the language pair itself could be considered as low-resource. Therefore the same technologies could be used to improve machine translation quality in other low-resource language pairs. We conduct experiments on existing and the curated corpora, and compare the performance of different MT models using these corpora. This results of this work can be used by Chinese–Portuguese MT research for comparison purposes.

In the future, we will investigate other approaches such as universal low-resource NMT (Gu et al., 2018) and discourse-aware approaches (Wang et al., 2018; Wang et al., 2017; Wang et al., 2016a) for Chinese–Portuguese MT task. Furthermore, we will keep exploring simple yet effective methods to build larger and domain-specific Chinese–Portuguese parallel corpora to further improve MT performance in this language pair.

6. Acknowledgements

The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant No. 13/RC/2106) and is co-funded under the European Regional Development Fund. This project has partially received funding from the European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska-Curie Actions (Grant No. 734211).

7. References

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, pages 1–15, San Diego, USA.

Bertoldi, N., Barbaiani, M., Federico, M., and Cattoni, R. (2008). Phrase-based statistical machine translation with

[13] https://github.com/rsennrich/subword-nmt.
[14] http://www.statmt.org/wmt17.
[15] http://www.statmt.org/europarl.

pivot languages. In *Proceedings of the 2008 International Workshop on Spoken Language Translation*, pages 143–149, Honolulu, Hawaii, USA.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Collins, M., Koehn, P., and Kucerova, I. (2005). Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 531–540, Ann Arbor, Michigan, USA.

Gu, J., Hassan, H., Devlin, J., and Li, V. O. (2018). Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics*. In press.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180, Prague, Czech Republic.

Lison, P. and Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation LREC*, pages 923–929, Portorož, Slovenia.

Liu, S. and Leal, A. L. V. (2016). Analysis of temporal adverbial phrases for portuguese–chinese machine translation. In *Proceedings of the 12th International Conference on Computational Processing of the Portuguese Language*, pages 62–73, Tomar, Portugal.

Lohar, P., Ganguly, D., Afli, H., Way, A., and Jones, G. J. (2016). FaDA: Fast document aligner using word embedding. *The Prague Bulletin of Mathematical Linguistics*, 106(1):169–179.

Lu, C., Leal, A., Quaresma, P., and Schmaltz, M. (2014). Analysis of the chinese–portuguese machine translation of chinese localizers qian and hou. In *Proceedings of the 10th China Workshop on Machine Translation*, pages 70–79, Macao, China.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.

Oliveira, F., Wong, F., Hong, I.-S., and Dong, M.-C. (2010). Parsing extended constraint synchronous grammar in chinese-portuguese machine translation. In *International Conference on Computational Processing of the Portuguese Language*, pages 62–73. Springer.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA.

Ramos, J. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the 1st Instructional Conference on Machine Learning*, pages 133–142, Piscataway, NJ USA.

Salton, G., Wong, A., and Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.

Sennrich, R. and Volk, M. (2010). MT-based sentence alignment for ocr-generated parallel texts. In *The 9th Conference of the Association for Machine Translation in the Americas*, Denver, Colorado, USA.

Sennrich, R. and Volk, M. (2011). Iterative, mt-based sentence alignment of parallel texts. In *Proceedings of the 18th Nordic Conference of Computational Linguistics*, pages 175–182, Riga, Latvia.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725, Berlin, Germany.

Stolcke, A. (2002). Srilm - an extensible language modeling toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing*, pages 901–904, Colorado, USA.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 2014 Neural Information Processing Systems*, pages 3104–3112, Montreal, Canada.

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In *Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC*, pages 2214–2218, Istanbul, Turkey.

Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). Modeling coverage for neural machine translation. In *Proceedings of the 54th annual meeting of the Association for Computational Linguistics*, pages 76–85, Berlin, Germany.

Utiyama, M. and Isahara, H. (2007). A comparison of pivot methods for phrase-based statistical machine translation. In *Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 484–491, Rochester, New York, USA.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st Annual Conference on Neural Information Processing Systems*, pages 6000–6010, Long Beach, California, USA.

Wang, L., Wong, D. F., and Chao, L. S. (2012). An im-

provement in cross-language document retrieval based on statistical models. In *Proceedings of the 24th Conference on Computational Linguistics and Speech Processing*, pages 144–155.

Wang, L., Tu, Z., Zhang, X., Li, H., Way, A., and Liu, Q. (2016a). A novel approach for dropped pronoun translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 983–993, San Diego, California, USA.

Wang, L., Zhang, X., Tu, Z., Way, A., and Liu, Q. (2016b). The automatic construction of discourse corpus for dialogue translation. In *Proceedings of the 10th Language Resources and Evaluation Conference*, Portorož, Slovenia.

Wang, L., Tu, Z., Way, A., and Liu, Q. (2017). Exploiting cross-sentence context for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2816–2821, Copenhagen, Denmark.

Wang, L., Tu, Z., Shi, S., Zhang, T., Graham, Y., and Liu, Q. (2018). Translating pro-drop languages with reconstruction models. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA. AAAI Press.

Wong, F. and Chao, S. (2010). PCT: Portuguese-chinese machine translation systems. *Journal of Translation Studies*, 13(1-2):181–196.

Wu, H. and Wang, H. (2007). Pivot language approach for phrase-based statistical machine translation. *Machine Translation*, 21(3):165–181.

Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Evaluating the WordsEye Text-to-Scene System: Imaginative and Realistic Sentences

Morgan Ulinski, Bob Coyne, Julia Hirschberg
Department of Computer Science
Columbia University
New York, NY, USA
{mulinski, coyne, julia}@cs.columbia.edu

Abstract

We describe our evaluation of the WordsEye text-to-scene generation system. We address the problem of evaluating the output of such a system vs. simple search methods to find a picture to illustrate a sentence. To do this, we constructed two sets of test sentences: a set of crowdsourced *imaginative sentences* and a set of *realistic sentences* extracted from the PASCAL image caption corpus (Rashtchian et al., 2010). For each sentence, we compared sample pictures found using Google Image Search to those produced by WordsEye. We then crowdsourced judgments as to which picture best illustrated each sentence. For imaginative sentences, pictures produced by WordsEye were preferred, but for realistic sentences, Google Image Search results were preferred. We also used crowdsourcing to obtain a rating for how well each picture illustrated the sentence, from 1 (completely correct) to 5 (completely incorrect). WordsEye pictures had an average rating of 2.58 on imaginative sentences and 2.54 on realistic sentences; Google images had an average rating of 3.82 on imaginative sentences and 1.87 on realistic sentences. We also discuss the sources of errors in the WordsEye system.

Keywords: text-to-scene, evaluation, corpus creation, crowdsourcing

1. Introduction

WordsEye (Coyne and Sproat, 2001) is a system for automatically converting natural language text into 3D scenes representing the meaning of that text. WordsEye supports language-based control of spatial relations, spatial properties, surface textures and colors, and cardinality; it handles simple anaphora and coreference resolution, allowing for a variety of ways to refer to objects and describe scenes. Scenes are assembled from a library of 3,000 3D objects and 10,000 2D images tied to a lexicon of 15,000 nouns. WordsEye is a web application (http://www.wordseye.com) with 27,000 registered real-world users. During the two year period from November 2015 to December 2017, approximately 2,200 users posted 14,000 finished scenes to an online gallery.

One task that WordsEye addresses is the problem of creating vs. automatically finding a picture to illustrate a sentence. Standard image search engines are limited to pictures that already exist in their databases, biasing them toward retrieving images of mundane and real-world scenarios. In contrast, a scene generation system like WordsEye can illustrate a much wider range of images, allowing users to visualize unusual and fantastical scenes. When users are freed from the normal constraints of what is possible or already exists they will often describe what they imagine – from situational, to iconic, to abstract, to fantastical. The majority of scenes created by actual users of the online WordsEye system are imaginative. One user commented "I truly enjoy watching people unleash their minds here." Some examples of imaginative scenes that have been created in WordsEye are shown in Figure 1.

The ability to generate both realistic and imaginative scenes from text input demonstrates that text-to-scene generation systems such as WordsEye can be used to supplement the results of image search engines such as Google. In evaluating WordsEye vs. image search engines, we therefore compare *imaginative sentences* and *realistic sentences* as items to be visualized. We use crowdsourcing to collect imaginative sentences and extract realistic sentences from the PASCAL image caption corpus (Rashtchian et al., 2010). In Section 2. we discuss related work. In Section 3. we introduce the WordsEye text-to-scene system. In Section 4. we describe the construction of *imaginative* sentences and *realistic* sentences for system evaluation. In Section 5. we explain the collection of potential illustrations for these, using Google Image Search or WordsEye. In Section 6., we discuss the use of crowdsourcing to evaluate the illustrations. We discuss the results of the evaluation in Section 7. and conclude in Section 8..



Figure 1: Imaginative Images: Situational, Iconic, Abstract, Fantastic

2. Related Work

Several systems exist for producing graphics from natural language sources. Glass and Bangay (2008) describe a system for transforming text sourced from popular fiction into

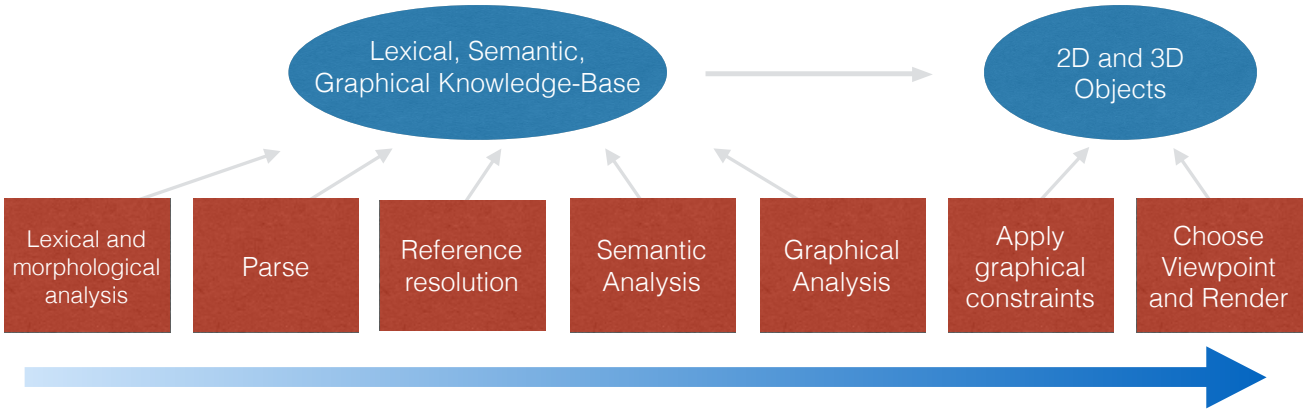


Figure 2: WordsEye System Architecture

corresponding 3D animations without prior language simplification. 3SVD (Zeng et al., 2005) is a 3D scene creation system using story-based descriptions. Parisi et al. (2007) describe an ontology-driven generation of 3D animations for training and maintenance. CONFUCIUS (Ma, 2006) is a multimodal text-to-animation system that generates animations of virtual humans from single sentences containing an action verb. In all these systems the referenced objects, attributes, and actions are typically relatively small in number or targeted to specific pre-existing domains. Seversky and Yin (2006) is a system tailored for interactively generating 3D scenes from natural language and voice and text input where the user can adjust the scene they create. Chang et al. (2014) created a text-to-scene system focused on learning spatial relations by supplying examples of indoor scenes and letting the user teach it by adjusting the scene to match the text. A survey of these and other text-to-graphics systems is given in Hassani and Lee (2016).

Others have used crowdsourcing to collect human-generated sentences, e.g. to create image captions. This includes the PASCAL image caption corpus (Rashtchian et al., 2010), Flickr8k (Hodosh et al., 2013) and Microsoft COCO (Chen et al., 2015). Our work differs in that we want to collect sentences describing anything users can imagine, as opposed to descriptions of existing photographs.

Zitnick and Parikh (2013) crowdsourced the evaluation of their scene generation system using 2D clip art: subjects created an initial set of scenes and wrote descriptions of the scenes. Zitnick et al. (2013) used several methods to automatically generate scenes for these descriptions and asked subjects which picture matched the description better. While the pictures that the sentences describe are human-constructed scenes rather than photographs from sources like Flickr, the scenes use a fixed set of 80 objects and are limited to the domain of children playing outside. Chang et al. (2015) evaluate their text-to-scene system by asking people to rate the degree to which scenes match the text used to generate them. Their test corpus includes a much larger number of objects than Zitnick et al. (2013), but the sentences and scenes are realistic descriptions of the configuration of objects in a room.

The WordsEye text-to-scene system was previously evaluated as an educational tool, as a means of helping students develop language skills. We found that students using the system had significantly greater improvement in their literary character and story descriptions in pre- and post-test essays compared with a control. In this paper, we focus on evaluating the pictures produced by the system more directly as accurate illustrations of input sentences.

3. The WordsEye Text-to-Scene System

In this section, we provide more details on the WordsEye text-to-scene generation system. WordsEye includes a library of approximately 3,000 3D objects and 10,000 2D images tied to a lexicon of 15,000 nouns. These include a wide variety of common objects (including variations of the same basic type, such as different types of doors or chairs) and textures (e.g. wood, grass, granite). WordsEye also supports several dozen graphical primitives and properties that are used for spatial relations (different senses of “in”, “on”, lateral relations, etc.), spatial properties (absolute and relative sizes and aspect ratios), and surface properties (colors, opacity, reflectivity, etc.). These primitives in conjunction with the objects and semantic knowledge about those objects (such as defaults for size, orientation, and top surface regions) allow the scene to be composed.

The system operates by first tokenizing each input sentence into lexical items (including modifiers like contractions or possessives) and possible parts-of-speech. The tokens are parsed into a labeled syntactic dependency structure. The dependency structure is then processed for anaphora and other co-reference, which is especially important for depicting multi-sentence input. These resolved structures are converted to lexical-semantic relations using lexical valence patterns and other lexical and semantic information. The resulting semantic relations are converted to a set of graphical constraints, representing the position, orientation, size, color, texture, cardinality, and poses of objects in the scene. The graphical constraints (other than poses, which currently are ignored) are applied to construct a fully specified 3D scene which is then rendered. Throughout this process, the system relies on a knowledge-base of lexical, semantic, and graphical information and a library of 2D and 3D objects. The architecture is shown in Figure 2.

4. Elicitation and Selection of Sentences

In this section we describe our use of crowdsourcing to collect imaginative sentences and filtering the PASCAL image caption corpus (Rashtchian et al., 2010) to obtain realistic sentences.

Category	Definition	Examples
PROP	Small objects that could be held or carried	*cellphone, apple, diamond*
FIXTURE	large objects such as furniture, vehicles, plants	*couch, sailing ship, oak tree*
ANIMAL	Animals	*dolphin, chicken, llama*
SPATIAL TERM	terms representing spatial relations	*above, against, facing, on*
NUMBER	small numbers	*one, four, nine, twelve*
COLOR	common colors	*beige, green, scarlet, black*
SIZE	general size or specific dimensions	*big, tiny, thin, 5 feet long*
DISTANCE	distances	*4 inches, five meters, 10 feet*
SURFACE PROPERTY	surface properties	*opaque, shiny, transparent*
LOCATION	terms representing terrain types and locations	*field, driveway, lake, forest*
BUILDING	buildings and architectural structures	*doghouse, castle, skyscraper*

Figure 3: Categories of words in the lexicon

<table>
<tr><th>AMT Column Headings</th><th colspan="2">WordsEye Lexical Categories</th></tr>
<tr><td>1.Noun1, Noun2, Spatial Term, Adjective
2.Adjective, Noun1, Noun2, Spatial Term</td><td>Noun1 is PROP. Noun2 is FIXTURE.</td><td rowspan="5">Spatial Term is SPATIAL TERM. Adjective is SIZE, COLOR or SURFACE PROPERTY. Distance is DISTANCE. Location is BUILDING or LOCATION. Size is SIZE. Color is COLOR. Number is NUMBER.</td></tr>
<tr><td>3.Adjective, Noun1, Noun2, Spatial Term</td><td>Noun1 is ANIMAL. Noun2 is FIXTURE.</td></tr>
<tr><td>4.Noun1, Noun2, Spatial Term, Distance, Adjective</td><td>Noun1, Noun2 are PROP, FIXTURE or ANIMAL.</td></tr>
<tr><td>5.Noun1, Noun2, Spatial Term, Location</td><td>Noun1 is ANIMAL. Noun2 is PROP or FIXTURE.</td></tr>
<tr><td>6.Noun1, Noun2, Spatial Term, Distance, Adjective
7.Adjective, Noun1, Noun2, Spatial Term, Distance
8.Noun1, Noun2, Spatial Term, Location
9.Noun1, Noun2, Spatial Term, Color, Size
10.Noun1, Noun2, Spatial term, Number
11.Noun1, Noun2, Spatial term, Number, Adjective
12.Adjective, Noun1, Noun2, Spatial Term, Number</td><td>Noun1, Noun2 are PROP or FIXTURE.</td></tr>
</table>

Figure 4: Possible combinations of categories for the sentence construction task

4.1. Imaginative Sentences

We used Amazon Mechanical Turk to obtain imaginative sentences for our evaluation. We gave Turkers short lists of words divided into several categories and asked them to write a short sentence using at least one word from each category. The words provided to the Turkers represent the objects, properties, and relations supported by the text-to-scene system.

To help Turkers construct sentences of different types, we organized the objects, properties, and relations into a few basic categories. The categories are listed in Figure 3. We restricted the lexicon to include only commonly known words that could be easily understood and recognized visually. We excluded super-types such as "invertebrate" and sub-types such as "european elk". We omitted obscure terms such as "octahedron" or "diadem". The resulting lexicon included about 1500 terms and phrases.

We created 12 different combinations of categories with 20 HITs per combination. Each HIT randomly presented different words for each category in order to elicit different types of sentences from the Turkers. This involved varying the types and number of categories as well as the order of the items in the categories. We wanted to encourage sentences such as "there is a blue dog on the large table" as well as different orders and constructs like "the dog on the large table is blue". Each HIT showed 4 or 5 categories, with three words per category. Figure 4 shows all the combinations of categories.

Our instructions specified that Turkers write one sentence using a maximum of 12 words. Words could be in any order as long as the resulting sentence was grammatical. We allowed the use of any form of a given word; for example, using a plural noun instead of a singular. We also allowed the use of *filler words* not listed in the categories, but asked Turkers not to add any unlisted *content words*. We defined *filler words* as words with "little meaning on their own, but that are used to make the sentence grammatical (e.g. *the*, *has*, *is*, *with*)" and *content words* as words that "refer to an object, action, or characteristic (e.g. *eat*, *shallow*, *organization*)." An example HIT is shown in Figure 5.

We restricted our task to workers who had completed at least 100 HITs previously with an approval rate of at least 98%. We paid $.04 per assignment. We started with 240 unique combinations of words and collected one sentence for each of these. After filtering out ungrammatical sentences, we ended up with a total of 209 imaginative sentences. Some examples are shown in Figure 6(a).

4.2. Realistic Sentences

We began with image captions collected by Rashtchian et al. (2010) for the PASCAL Data Set (Everingham et al., 2011), which consists of 5000 descriptive sentences, 5 cap-

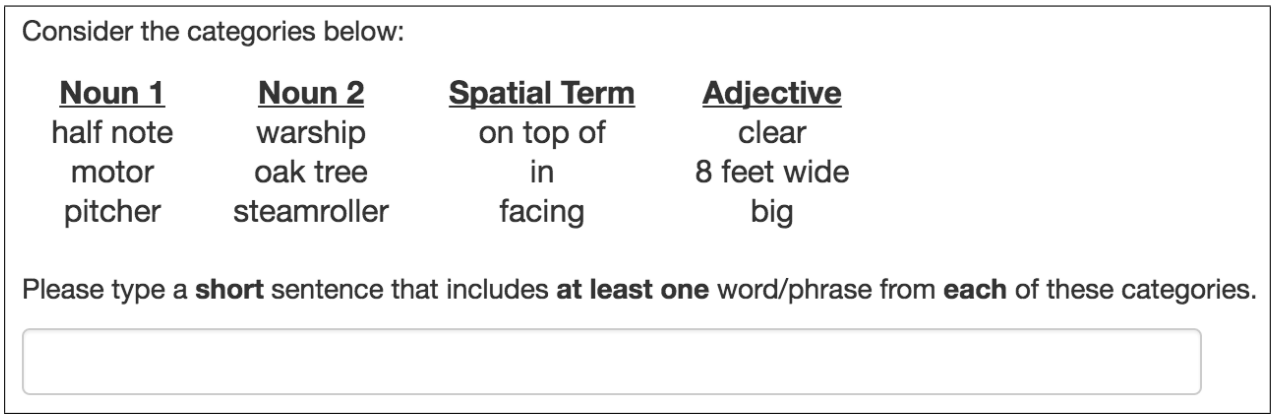


Figure 5: Example of sentence collection HIT

(a) Imaginative

- *The huge jewel is in front of the red rolling pin.*
- *Five pears are under the martini glass.*
- *The large prawn is on top of the stool.*
- *The red clock is three feet above the desk.*
- *Two tulip trees are close to a seashell.*

(b) Realistic

- *A brown duck and white duck stand on the grass.*
- *A man is standing next to a yellow sports car.*
- *A black dog in a grass field.*
- *The big white boat is in the ocean.*
- *A child sits in a large black leather chair.*

Figure 6: Examples of imaginative and realistic sentences

tions each for 1000 images. The images cover 20 object categories from the original PASCAL task, including *people, animals, vehicles*, and *indoor objects*. We used at most a single caption for each photograph.

To select a usable caption, we manually removed all ungrammatical sentences and fed the remaining sentences into WordsEye, which was able to create scenes for about one third of the image captions; the captions that were rejected were omitted due to current limitations of the system's lexicon, object library, or parser. Since our goal is to evaluate WordsEye we excluded these sentences which are outside the domain of the system. For example we omitted most sentences using action verbs since the system currently cannot pose characters to represent those verbs. We kept simple stative pose-related verbs such as "sit" and "stand" so the system could capture other aspects of the sentence. We also omitted sentences that could not be parsed or that had concrete nouns with no corresponding 3D object. This resulted in a total of 250 realistic sentences. Some examples are shown in Figure 6(b).

5. Collection/Generation of Illustrations

In this section, we describe how we obtained the possible illustrations for each sentence.

Google Image Search: We used each sentence as a query for Google image search. We did not strip punctuation, add quotation marks, or otherwise modify sentences. The first 4 results were downloaded and resized to a uniform width.

WordsEye Scene Generation: Since WordsEye images are rendered 3D scenes, they can be easily viewed from different angles. Normally, users can interactively change the viewpoint in the scene they are creating and choose the best view. So our approach was to automatically generate four WordsEye scenes with slightly different camera views. If one of the objects was occluded by another (and hence not visible in a front view of the scene), we automatically produced an alternate view of the scene from the back. Likewise, the elevation of the camera was varied to allow an object to potentially be more visible. We randomized the objects chosen in the scene from those compatible with the sentence. For example, Figure 7 shows the four scenes generated for the sentence "A furry dog lying on a glass table."

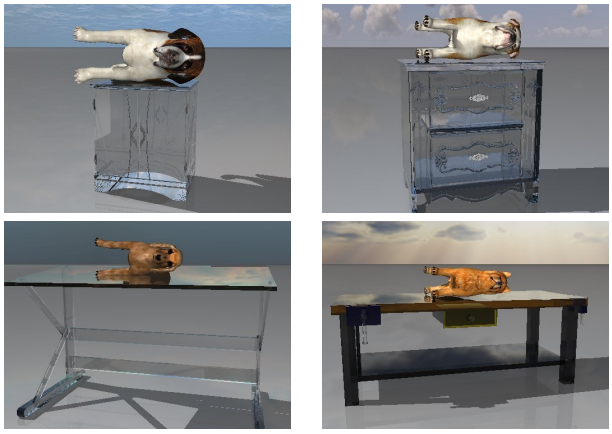
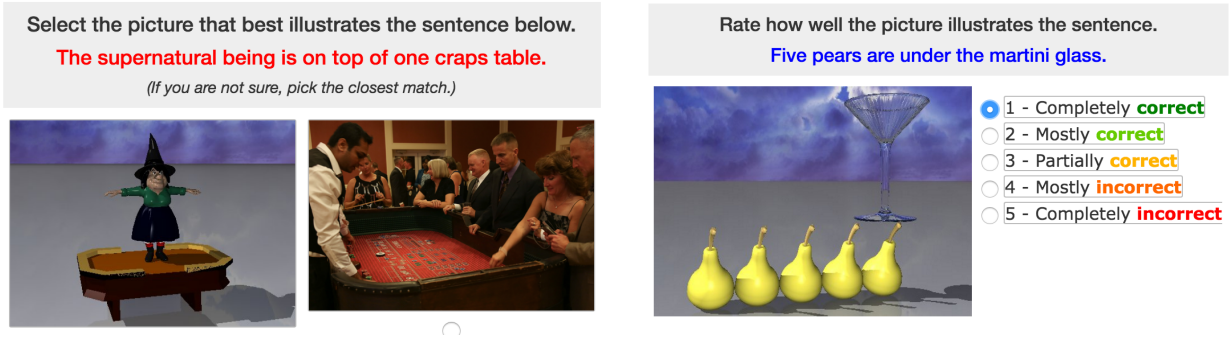


Figure 7: Generated scenes for the sentence "A furry dog lying on a glass table."

6. Evaluating Illustrations with AMT

The evaluation of the quality of the illustrations was done in two phases. In the first phase we asked Turkers to determine the best image for each sentence from the downloaded Google results and (separately) for each sentence among the Wordseye-generated images. In the second phase, Turkers evaluated the quality of the best Google image and the best WordsEye image. We did this second phase evaluation with two separate crowdsourced tasks. In the first, we asked Turkers to compare the best Google image with the best WordsEye image directly. In the second, we obtained a rating for how well each of the images illustrated the sentence. For all tasks, we required turkers to have previously



(a) (b)

Figure 8: Examples of the second phase AMT tasks: (a) image comparison task and (b) rating task. (Google image source: `https://en.wikipedia.org/wiki/Craps`)

completed at least 500 HITs and to have a 98% approval rate. We paid $0.01 per assignment.

Each image comparison HIT showed a single sentence with the possible images below it. Turkers were asked to select the picture that best illustrated the sentence. In the first phase, we showed four pictures and collected 5 judgments for each HIT. In case of ties, we published additional assignments for that sentence until one image had more votes than any of the others. The image that received the most votes was used in the next phase, which compared the winning Google image with the winning WordsEye image. In the second phase, we collected 3 judgments for each HIT, which guaranteed no ties. A sample HIT from the second phase is shown in Figure 8(a).

For the rating task, each HIT showed a single sentence and a single image. Turkers were asked to rate how well the picture illustrated the sentence. The scale was from 1 (completely correct) to 5 (completely incorrect). We collected 3 judgments for each HIT and averaged these ratings to obtain the final rating for each picture. An example of the rating HIT is shown in Figure 8(b).

7. Results and Discussion

In this section, we discuss results from the second phase of evaluation. In the image comparison task, we asked 3 Turkers to choose the picture that best illustrated the sentence. The distribution of outcomes is shown in Figure 9. The winner is shown in bold for each category.

Next, we obtained a rating for each image from 1 (completely correct) to 5 (completely incorrect). Figure 10(a) shows average ratings for Google and WordsEye for each category of sentence, with the better rating in each category shown in bold. We also calculated the winning image for each category based on the ratings. For each sentence, the winning image was the one with the lower rating. These are shown in Figure 10(b), with the winner for each category shown in bold.

The trend for both votes and ratings is the same: WordsEye is superior for imaginative sentences and Google for realistic sentences. The winning image based on votes is not always the same as the winner based on rating. Figure 11 compares the distribution based on ratings and votes.

For imaginative sentences, when the Google and WordsEye ratings were tied, WordsEye tended to win the votes. Even

Winner (votes)	**Imaginative**	
WordsEye (3 to 0)	60.3% (126)	**85.6% (179)**
WordsEye (2 to 1)	25.4% (53)	
Google (2 to 1)	10.0% (21)	14.4% (30)
Google (3 to 0)	4.3% (9)	
Total	100.0% (209)	

(a)

Winner (votes)	**Realistic**	
WordsEye (3 to 0)	8.8% (22)	16.4% (41)
WordsEye (2 to 1)	7.6% (19)	
Google (2 to 1)	14.4% (36)	**83.6% (209)**
Google (3 to 0)	69.2% (173)	
Total	100.0% (250)	

(b)

Figure 9: Distribution of Turkers' Votes for WordsEye vs. Google Images for (a) imaginative sentences and (b) realistic sentences.

	Imaginative	**Realistic**
WordsEye	**2.581**	2.536
Google	3.825	**1.871**

(a)

Winner	**Imaginative**	**Realistic**
WordsEye	**74.6% (156)**	25.6% (64)
Tie	5.3% (11)	13.6% (34)
Google	20.1% (42)	**60.8% (152)**
Total	100.0% (209)	100.0% (250)

(b)

Figure 10: (a) Avg. ratings for WordsEye and Google images. (b) Distribution of winner based on ratings.

when Google had a better rating than WordsEye, WordsEye still tended to win by votes. In particular, out of the 42 cases where the Google image received a better rating, Turkers chose the WordsEye image for 24 (more than half) of them. This pattern is reversed for the realistic sentences. For realistic sentences, when both images had the same rating, Turkers tended to choose the Google image. However,

	WordsEye won rating	Tie rating	Google won rating	Total
WordsEye won votes	70.3% (147)	3.8% (8)	11.5% (24)	85.6% (179)
Google won votes	4.3% (9)	1.4% (3)	8.6% (18)	14.4% (30)
Total	74.6% (156)	5.3% (11)	20.1% (42)	100.0% (209)

(a) Imaginative sentences

	WordsEye won rating	Tie rating	Google won rating	Total
WordsEye won votes	14.0% (35)	0.8% (2)	1.6% (4)	16.4% (41)
Google won votes	11.6% (29)	12.8% (32)	59.2% (148)	83.6% (209)
Total	11.6% (64)	13.6% (34)	60.8% (152)	100.0% (250)

(b) Realistic sentences

Figure 11: Distribution of winner (WordsEye vs Google) based on ratings and votes.



(a) (b) (c)

Figure 12: Example WordsEye errors: (a) camera viewpoint partial occlusion: (*The hotdog is next to the chocolate cake in the booth.)* and (b) graphical interpretation and knowledge-base: (*Two men in a small wooden canoe on the water*) and (c) semantic interpretation: (*a gray house with a red door*)

when WordsEye had the better rating for a realistic sentence, Turkers still tended to choose the WordsEye image. Thus, while Turkers seemed to prefer to associate imaginative sentences with WordsEye-style pictures when forced to make a binary choice (even when the Google image had a lower rating), the reverse bias does not hold for realistic sentences: when an WordsEye image illustrated a realistic sentence better based on rating, the binary choices made by Turkers usually favored the WordsEye image as well.

7.1. Error Analysis

In this section we examine the sources of errors in the WordsEye system. One common cause of errors was a poorly placed camera (3D viewpoint). This was especially an issue for imaginative sentences which could involve very small objects in the same scene with large ones, making it hard to see both at the same time, given our default algorithm for positioning the camera to frame the full scene. A better strategy would be to position the camera aimed to frame the small object with the large object in the background. In other cases, one object was inside another (e.g. within an enclosed area such as a building) and the default generated camera positions were outside the building, making it impossible to see the inner object. Another source of errors was from missing graphical primitives. For example, sentences that required a person or animal to be in a particular pose (e.g. sitting) are internally represented, but the system is currently unable to actually put the 3D character into a pose. A third source of errors was in anaphora resolution in text like *A field with many black cows in it*. The WordsEye system currently processes anaphora and other co-reference across sentences but not within a sentence. Other errors occurred because of incorrect information stored in the knowledge base (e.g. incorrect real-world sizes resulting in strange relative sizes between objects in a scene) or from incorrect or unexpected semantic and graphical interpretations.

A description of the kinds of errors that could occur in each WordsEye module (see Figure 2) is presented here. Examples of some of these are shown in Figure 12.

- **Knowledge Base:** missing lexical entry or word sense; incorrect object (or part) properties.
- **Graphics Library:** missing or unrepresentative 3D object.
- **Parsing:** problem with syntax or punctuation.
- **Reference resolution:** unresolved anaphora.
- **Semantic analysis:** syntax-to-semantic conversion, including object selection and ambiguity.
- **Graphical analysis:** incorrect graphical interpretation (backgrounds, materials, spatial layout).
- **Apply graphical constraints:** spatial constraint maintenance issues, missing graphical primitives.
- **Camera/Render:** camera angle, zoom, occlusions.

We tagged each WordsEye image that had a rating worse than 2 with the type of error it exhibited and the WordsEye module where the error occurred. The WordsEye pictures for 114 imaginative sentences and 137 realistic sentences were tagged with errors. Figure 13 shows the distribution of errors per module. Note that since some pictures were tagged with errors from multiple modules, the total of each column is greater than 100%. WordsEye made

WordsEye Module	Imaginative	Realistic
Knowledge Base	10.0% (21)	2.4% (6)
Graphics Library	3.3% (7)	6.0% (15)
Parsing	2.9% (6)	2.0% (5)
Reference resolution	0.5% (1)	2.0% (5)
Semantic analysis	3.8% (8)	15.2% (38)
Graphical analysis	10.0% (21)	25.6% (64)
Apply graphical constraints	4.3% (9)	21.2% (53)
Camera/Render	42.1% (88)	6.8% (17)
No Error	45.5% (95)	45.2% (113)

Figure 13: Errors per module. Note: a given sentence could have more than one error.

more knowledge-base errors and camera errors on imaginative sentences. It made more semantic analysis, graphical analysis, and apply graphical constraints errors on realistic sentences.

8. Summary

We have described our evaluation of the WordsEye text-to-scene system; specifically, we have evaluated WordsEye's ability to create a picture that illustrates *imaginative* and *realistic* sentences, as compared to traditional image search methods (i.e. Google search). We found that WordsEye performs very similarly on both kinds of sentences (average rating of 2.581 and 2.536, respectively - on our rating scale, halfway between "mostly correct" and "partially correct"). While Google search does perform better than WordsEye on realistic sentences (average rating of 1.871 - between "completely correct" and "mostly correct"), performance breaks down when faced with imaginative sentences (average rating of 3.825 - between "partially correct" and "mostly incorrect"). Thus, we have shown that WordsEye is superior for imaginative sentences, and Google search is superior for realistic sentences. While this result is not unexpected, we can now quantify what the gap in performance actually is. In particular, while the average rating of WordsEye on realistic sentences was just 0.665 below that of Google, WordsEye's ratings on imaginative sentences was 1.244 higher than Google's. This suggests that as WordsEye and text-to-scene technology in general improve, they may become a viable alternative to image search even for realistic sentences, but it might be difficult to adapt traditional image search techniques to retrieve illustrations for imaginative sentences. In addition, as sentences get longer and more complicated (or if multiple sentences are involved), Google might be begin to have more trouble with realistic sentences as well.

Creativity is something that too often gets overlooked in technology development, and our results show that research into text-to-scene generation could play an important role in addressing the issue. Our new corpus of imaginative sentences may also have applications for other researchers studying language in a visual context or those interested in spatial language in general.

9. Bibliographical References

Chang, A. X., Savva, M., and Manning, C. D. (2014). Semantic parsing for text to 3D scene generation. In *Association for Computational Linguistics (ACL) Workshop on Semantic Parsing*.

Chang, A. X., Monroe, W., Savva, M., Potts, C., and Manning, C. D. (2015). Text to 3d scene generation with rich lexical grounding. *CoRR*, abs/1505.06289.

Chen, X., Fang, H., Lin, T., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. (2015). Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

Coyne, B. and Sproat, R. (2001). WordsEye: An automatic text-to-scene conversion system. In *SIGGRAPH*.

Everingham, M., Van Gool, L., Williams, C., Winn, J., and Zisserman, A. (2011). The pascal visual object classes challenge results (voc2011).

Glass, K. and Bangay, S. (2008). Automating the creation of 3d animation from annotated fiction text. In *IADIS 2008: Proceedings of the International Conference on Computer Graphics and Visualization 2008*, pages 3–10. IADIS Press.

Hassani, K. and Lee, W.-S. (2016). Visualizing natural language descriptions: A survey. *ACM Computing Surveys (CSUR)*, 49(1):17.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, 47(1):853–899, May.

Ma, M. (2006). *Automatic conversion of natural language to 3D animation*. Ph.D. thesis, University of Ulster.

Parisi, S., Bauch, J., Berssenbrugge, J., and Radkowski, R. (2007). Ontology-driven generation of 3d animations for training and maintenance. In *2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*, pages 608–614, April.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. (2010). Collecting image annotations using amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 139–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Seversky, L. M. and Yin, L. (2006). Real-time automatic 3d scene generation from natural language voice and text descriptions. In *Proceedings of the 14th ACM International Conference on Multimedia*, MM '06, pages 61–64, New York, NY, USA. ACM.

Zeng, X., Mehdi, Q. H., and Gough, N. E. (2005). From visual semantic parameterization to graphic visualization. In *Ninth International Conference on Information Visualisation (IV'05)*, pages 488–493, July.

Zitnick, C. L. and Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June.

Zitnick, C. L., Parikh, D., and Vanderwende, L. (2013). Learning the visual interpretation of sentences. In *The IEEE International Conference on Computer Vision (ICCV)*, December.

Computer-assisted Speaker Diarization: How to Evaluate Human Corrections

Broux Pierre-Alexandre[1,2], Doukhan David[2], Petitrenaud Simon[1], Meignier Sylvain[1], Carrive Jean[2]
1: Computer Science Laboratory Le Mans University (LIUM - EA 4023), Le Mans, France
2: French National Audiovisual Institute (INA), Paris, France
pabroux@ina.fr, ddoukhan@ina.fr, simon.petit-renaud@univ-lemans.fr, sylvain.meignier@univ-lemans.fr, jcarrive@ina.fr

Abstract

In this paper, we present a framework to evaluate the human corrections of a speaker diarization. We propose four elementary actions to correct the diarization and an automaton to simulate the correction sequence. A metric is described to evaluate the correction cost. The framework is evaluated using French broadcast news drawn from the REPERE corpus.

Keywords: Speaker diarization, annotation, Human-Computer Interaction (HCI), evaluation

1. Introduction

The work presented in this paper has been realized to cope with some needs of the French National Audiovisual Institute (INA). INA is a public institution in charge of the preservation and the promotion of French audiovisual heritage. The promotion task partly relies on the annotation of audiovisual document collections. The annotation consists in enriching the documents with summaries, keywords or participant names in order to satisfy the complex queries elaborated by INA customers or researchers within media databases.

However, due to the increasing number of documents and the limited number of annotators, many documents remain undocumented or only partly documented. The information provided by the annotation greatly varies according to the kind of archives: the broadcast news is usually finely annotated, while the other programs such as games, documentaries, variety shows or reality shows are much less annotated. Thus, enterprises owning large undocumented or partly documented collections such as INA need to exploit their resources even better. One of the solutions to facilitate the annotation and improve the access to its documents is to use automatic speech and speaker recognition technologies as proposed in Charhad et al. (2005; Ordelman et al. (2009; Vallet et al. (2016).

The speaker diarization task is a necessary pre-processing step for speaker identification (Bonastre et al., 2000) or speech transcription (Anguera et al., 2012) in broadcast shows. The speaker diarization and speaker identification tasks allow to determine « who spoke when ». Speaker diarization systems are generally based on unsupervised segmentation and clustering methods, in charge of estimating the number of speakers, and splitting the audio stream into labeled speech segments assigned to anonymous speakers. However, state-of-the-art speaker diarization systems are still not sufficiently accurate to be employed into most of INA's applications, mainly because of the wide variety of INA's collections. The variety relates to the time period (from the end of the nineteenth century to nowadays), the type of broadcast or the recording conditions. For these various reasons, human interventions are most of the time required to obtain robust annotations.

Entirely manual annotation of speech cannot be a reasonable solution as it is a very expensive process. Indeed, nine hours are required to perform the manual annotation corresponding to one hour of spontaneous speech (speech transcription and speaker identity) (Bazillon et al., 2008). Thus, a human annotator should be assisted by an automatic system to be efficient.

In this paper, we propose a framework to experiment human assisted diarization methods. More precisely, the aims are to build an automaton which simulates the annotator corrections and to propose a metric to evaluate these corrections.

In this paper, firstly, we present the state of the art in the field of annotation in speech or speaker recognition systems. Then, we propose an overview of a human assisted diarization system and we propose a new metric to evaluate such systems. In the following part, we describe the human actions used to correct the diarization. Before concluding, we measure the duration of each action to build the proposed metric and we evaluate an oracle system based on the automaton.

2. Related work

The human annotation of an audio document is time consuming. This task is generally manually realized with annotation software like *Transcriber* (Barras et al., 2001) or *ELAN* (Wittenburg et al., 2006). In Bazillon et al. (2008) the authors have shown that the output correction of an automatic speech transcription system decreases the time devoted to the annotation process. An active learning method, proposed in Budnik et al. (2014), used in conjunction with various systems, for example with a speaker diarization system and a face-recognition system, further reduces the number of human-machine interactions. The authors proposed to apply their method to the output of a multilayer perceptron (ML) classifier, based on lip activity and other temporal characteristics. This classifier was used on both speaker and face tracks extracted from videos so as to find associations between them and create multimodal clusters. These clusters were initially labeled thanks to an optical character recognition (OCR). Recently, in Broux et al. (2016), we proposed a system assisting an annotator for

the correction of a diarization system reducing the number of human interventions. In that paper, the annotator only corrects speaker clustering errors and the segmentation was assumed to be perfect. More precisely, the input of our system was a segmentation obtained from the ground truth and a hierarchical agglomerative clustering was applied to this segmentation. These last two papers are focused on the correction of clustering errors and forget the segmentation errors. Moreover, the authors unfoundedly assumed that every kind of correction has the same cost. This assumption is not judicious, since each correction requires specific actions from the annotator. These actions require different mental efforts, a different physical effort and provides a different result. For example, it can be assumed that it is easier to change the speaker label (with the use of an exhaustive list of potential speakers) than to create a new speaker label, since it is not provided and the annotator may need much time to find it.

3. Human assisted diarization system

In this section, we propose an overview of a human assisted diarization system as well as a new metric to evaluate such systems.

3.1. Description of the system

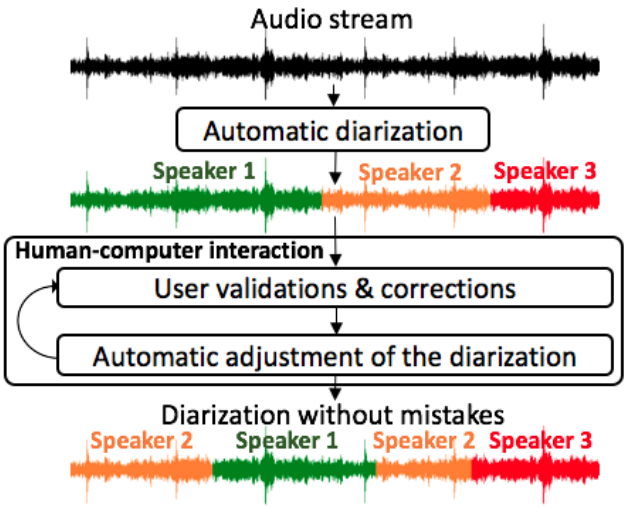


Figure 1: Architecture of a diarization system assisted by the human

Figure 1 presents the architecture of a diarization system assisted by a human. It is composed of two main parts. The first one consists in providing an automatic diarization from an audio stream. An initial segmentation of the stream is then obtained. The second one consists of asking a human to correct the output of the first part. Each human correction is in turn taken into account by a system which improves the diarization, generally making easier the remaining actions of the annotator. According to the target, the annotator can achieve corrections for the clustering task and/or the segmentation task. At the end of the process, the diarization error rate (DER (NIST, 2003; Galibert, 2013)) is expected to have decreased thanks to human and system corrections. It is recalled that the DER is the fraction of speaking time which is not attributed to the correct speaker by using the best matching between speaker labels of the references and the hypotheses.

3.2. Experimental framework

From the framework presented in the preceding section, several rules have been defined:

1. the annotator is simulated by an automaton and does not make any error (it is a noiseless automaton);
2. the annotator corrects the show from the beginning to the end in temporal order so as to validate the automatic annotation done a posteriori;
3. only the current speech turn can be corrected by the annotator.

The noiseless automaton rule allows to avoid the complex random modeling of the error according to the annotator. Moreover, this natural simplification allows to have a document without any error at the end of the correction process and thus a DER equals to 0%.Correcting from the beginning to the end is supposed to help the annotator understanding and improving the correction. This rule and the last one are experimental conditions chosen to facilitate our problem. They can be questioned thereafter.

3.3. Proposed metric: HCIQ

Since the DER measures the quality of a diarization (NIST, 2003), it is not relevant to evaluate human interactions. A new metric similar to the Keystroke Saving used in word prediction for people with communication difficulty (Wood and Lewis, 1996) is proposed. We called it Human-Computer Interaction Quantity (HCIQ). This metric estimates the human intervention cost for the diarization correction. The HCIQ can be computed both for assisted systems and systems where a human corrects the diarization alone. Furthermore, as well as the DER, the HCIQ can be computed for each recording or for a set of audio/video recordings. It is defined by the formula:

$$HCIQ = \sum_{i=1}^{K} w_i n_i,$$

where the index i corresponds to a correction action type in the interface, w_i is its associated cost, n_i the number of times the annotator has applied this action type and K is the number of different action types.
The lower the HCIQ measure, the lower the correction duration of the annotation will be. By the way, it allows to compare different assisted diarization systems in a objective way.
The HCIQ measure, in its current form, does not allow to compare different corpora. In order to resolve it, we propose the following formula:

$$HCIQ_n = \frac{HCIQ}{d},$$

where $HCIQ$ is normalized by the corpus duration d on which the $HCIQ$ was assessed. The HCIQ$_n$ measure is a

ratio of the number of corrections to do for a unit of time. For a given corpus, when $HCIQ_n$ increases, the amount of corrections increases also.
The HCIQ is close to others metrics that assess the amount of effort needed to correct a given kind of errors. For instance, the word error rate (WER (McCowan et al., 2004)) estimates the number of incorrect words in a transcription and the translation edit rate (TER (Snover et al., 2006)), which is derived from the WER, assesses the number of corrections needed for a human to reach the reference translation.

4. Annotator and assisted diarization tools

In this section, we describe the annotation software and the authorized correction actions which we used to correct a diarization.

4.1. Annotation software: Transcriber

To choose the human actions needed to the correction, we relied on *Transcriber*, a reference software in speech transcription and annotation.
This software allows to cut an audio stream into segments. Each segment corresponds to a speech zone and is labeled with a speaker name. This label may be enriched by information such as the gender or the native language of the speaker.
In *Transcriber*, the segmentation actions are "*Create a boundary*", "*Delete a boundary*" and "*Move a boundary*". The "*Create a boundary*" action adds a boundary by cutting a segment into two pieces, the "*Delete a boundary*" action merges two consecutive segments and the "*Move a boundary*" action moves the boundary of a segment. Concerning the clustering actions, *Transcriber* offers the "*Create a speaker label*" and "*Change the speaker label*" actions. The former allows to create a new speaker label for the current segment whereas the latter allows to change the speaker label.

4.2. Correction actions

In order to facilitate the creation of a simulated annotator, the series of actions will be deterministic. No action can be substituted by a set of actions providing the same correction. One of the *Transcriber* actions does not fulfill this criterion. The "*Move a boundary*" action can be replaced by the two following actions: "*Create a boundary*" and "*Delete a boundary*".
To sum up, we kept two actions for modifying the segment boundaries and two actions for modifying the labels. By combining these actions, we can describe all the corrections in a unique way. Finally, the four selected actions used in the HCIQ metric are :

- "*Create a boundary*",
- "*Delete a boundary*",
- "*Create a speaker label*",
- "*Change the speaker label*".

5. Experiments

In this section, we present the corpus used for our experiments, the measure of the action duration in order to build the proposed metric and the evaluation of an oracle system based on the automaton.

5.1. Corpus

The experiments have been applied on TV recordings from the 2013 evaluation campaign of the challenge ANR-REPERE[1]. The TV shows come from two French channels (BFM and LCP). They are mainly composed of talk shows and new broadcasts.

Show number	7
Recording number	28
Recording time	14h17
Annotation time	2h57
Speaker number	212

Table 1: REPERE test 2013 description

Table 1 describes the corpus used in the experiments. The corpus is balanced : it contains spontaneous and prepared speech. It is made up of street interviews, debates and information shows but only a part of the data is annotated (Kahn et al., 2012).

5.2. Measure of the action duration

According to Arora et al. (2009), three variables can directly affect the assessment of the action duration: the interface, the annotator and the annotated document. So as to obtain an accurate assessment, each of these variables is studied.

5.2.1. The interface

The assessment of action duration varies according to the used interface. The more ergonomic it is, the more a user quickly annotates and the more the annotation time decreases. *Transcriber* offers good ergonomics. In *Transcriber*, a human generally cuts the audio stream by putting boundaries on speaker changes, silences or breaths. In our framework, only speaker changes are useful and are annotated.

5.2.2. The annotator

The annotation time varies according to the experience the annotator can have both in annotation itself and in the annotation software. If a person is used to annotate, he or she will effective and the annotation will take little time. Moreover, people frequently using an annotation software correct more quickly than people discovering it. Therefore, there are two possible strategies to measure the action duration:

1. using the average of time of annotators having the same experience;
2. using the average of time of annotators having several experiences.

[1] http://www.defi-repere.fr/

The former one allows to obtain specific time which can be useful in accordance with the aim. The latter one, that we chose in our experiments, offers global time covering different annotator profiles with variable expertise levels. Unfortunately, this time is constrained by the annotator profiles which we have at our disposal.

5.2.3. The annotated document

As far as speech is concerned, the audio documents can mainly be separated in three groups: telephone, meeting and (radio/TV) show. These three groups essentially varies according to two points: the audio stream quality and the spontaneous degree. The spontaneous degree is correlated with the disfluency number. When the degree is high, the disfluency number is also high and the clauses are more ungrammatical. The spontaneous degree implies various phenomena such as overlapped speech, false starts, etc. (Bazillon et al., 2008; Dufour et al., 2009). When the signal quality is low, it is more difficult to annotate. Furthermore, when the spontaneous degree is high due to the overlapped speech, it is more complicated to determine who speaks when, and then it is more difficult to annotate. The signal quality and the spontaneous degree have consequences on the human annotation time (Bazillon et al., 2008). Therefore, two strategies can also be conceivable:

1. annotating various documents;
2. annotating specific documents.

We choose the latter strategy. Thus the documents to annotate are good audio quality with a low spontaneous degree. This choice allows to facilitate the measure of annotation time.

5.2.4. Evaluation

In section 4.2., we selected four correction actions. Now, we propose a method to estimate the average duration of each action, in the framework we defined previously. The history of mouse clicks and keyboard strokes in *Transcriber* permits to indirectly determine the successive actions and to precisely assess the duration of each action. To record this detailed log file, we had to modify the *Transcriber* code. A log file input, i.e. a mouse click or a keyboard stroke, contains three types of information : the click or the stroke time, the active module name and a comment (figure 2). The module identifies an element of the user interface whereas the comment gives accurate details on the event in progress.

The log file itself is not enough to determine the actions in an automatic way. Indeed, the annotator can make some mistakes or take a break during the annotation session. To solve this problem, the recording of the user screen is manually segmented into one or several actions thanks to the log file. Thus the measured duration accurately correspond to the actual actions.

In accordance with 5.2.3., only the regions with few spontaneous speech and without overlapping speech are annotated to assess the annotation time for each action. Table 2 shows the results of the duration of actions.

The most time consuming actions are the ones consisting to "*Create a speaker label*" and "*Create a boundary*", with an average of about 12-13 seconds.The first action requires to enter a speaker label (and possibly other speaker meta data), while the second action requires to look and listen to the signal to detect the speaker boundary. Moreover, it is generally necessary to listen to the signal several times to place a new boundary. The action called "*Change the speaker label*" has an average of 7.6 seconds. Thanks to a contextual window, it consists in selecting the correct speaker label in a drop-down list. Looking for a label in a drop-down list takes a less mental effort compared to the boundary creation. The fastest action is the "*Delete a boundary*" action. It requires to stop listening when a false boundary is detected and to delete it by a simple keyboard key combination. Correcting an error is in reality built upon three phases:

1. detecting the presence of an error;
2. finding the place of the error;
3. correcting the error.

Each action duration in table 2 represents the sum of the time of these three phases. However, these durations do not take into account the listening time.

```
[1319509750] :: [Player] [Strategy: Play/Pause; Play at 639.961]
[1319494751] :: [Player] [Strategy: Play/Pause; Pause at 654.942]
[1319493381] :: [LabelWindow] [Open window; Edit an existing Turn]
[1319491287] :: [Label] [Edit an existing speaker thanks to LabelWindow]
[1319480451] :: [LabelWindow] [Validate; Close Window]
[1319480451] :: [Label] [Cancel the edited speaker thanks to LabelWindow]
[1319477766] :: [LabelWindow] [Open window; Edit an existing Turn]
[1319475399] :: [Label] [Edit an existing speaker thanks to LabelWindow]
[1319469222] :: [LabelWindow] [Validate; Close Window]
[1319469222] :: [Label] [Cancel the edited speaker thanks to LabelWindow]
[1319465719] :: [Player] [Strategy: Play/Pause; Play at 654.942]
[1319326169] :: [Player] [Strategy: Play/Pause; Pause at 794.463]
[1319324161] :: [LabelWindow] [Open window; Edit an existing Turn]
[1319321769] :: [Label] [Edit an existing speaker thanks to LabelWindow]
[1319313024] :: [LabelWindow] [Validate; Close Window]
[1319313024] :: [Label] [Cancel the edited speaker thanks to LabelWindow]
[1319310256] :: [LabelWindow] [Open window; Edit an existing Turn]
[1319307961] :: [Label] [Edit an existing speaker thanks to LabelWindow]
[1319303527] :: [LabelWindow] [Validate; Close Window]
[1319303527] :: [Label] [Cancel the edited speaker thanks to LabelWindow]
[1319300105] :: [Player] [Strategy: Play/Pause; Play at 794.463]
[1319262090] :: [Player] [Strategy: Play/Pause; Pause at 832.461]
[1319258313] :: [LabelWindow] [Open window; Edit an existing Turn]
[1319256490] :: [Label] [Edit an existing speaker thanks to LabelWindow]
[1319249522] :: [LabelWindow] [Validate; Close Window]
[1319249522] :: [Label] [Cancel the edited speaker thanks to LabelWindow]
[1319246850] :: [Cursor] [Change: 832.461 to 829.673]
```

Figure 2: Example of a log file

Action	Nb	Avg (sec)	Std (sec)
Create a speaker label	28	12.7	6.0
Change the speaker label	32	7.6	3.8
Create a boundary	38	12	7.6
Delete a boundary	46	5.1	2.3

Table 2: Duration of actions - 20 min of REPERE test 2013 data. Nb: Number of occurrences; Avg: Average duration; Std: Standard deviation.

5.3. Evaluation of an oracle system

The simulated annotator relies on two types of information to determine whether a correction is required at time t:

- the correspondence between the reference segment (ground truth) and the hypothesis segment;
- the matching between the reference and the hypothesis speaker labels which minimize the DER.

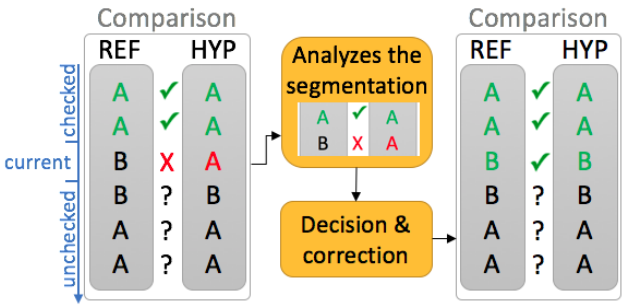


Figure 3: Illustration of the simulated annotator

If there is at least one discordance in the diarization at time t between the reference and the hypothesis, the simulated annotator firstly corrects the segmentation errors and then the speaker clustering errors (figure 3). After each correction, the system can run a diarization system on the unchecked part (segments with start time $> t$) by taking into consideration the already checked segments (segments with end time $\leq t$).

Without segmentation errors, the clustering correction is easy to set up (Broux et al., 2016). The segmentation correction is more difficult, the simulated annotator needs to deal with the accuracy of the reference boundaries. To solve this problem, a tolerance of more or less 250 ms is generally applied to the boundaries of the reference segments for the DER computation. We applied the same tolerance to avoid the numerous and generally useless corrections. So before the assessment of the potential discordance between the reference and the hypothesis, any hypothesis boundary belonging to a tolerance area is moved in order to be aligned with the reference boundary.

The simulated annotator becomes an oracle system when no automatic adjustment is performed as corrections. The oracle system evaluation is reported in table 3. The HCIQ of the test corpus is 331.6 minutes and corresponds to the sum of all duration estimations (table 3). The input diarization of the oracle is provided by the full-automatic diarization system described in Meignier and Merlin (2010). The DER of the input diarization is 13.80%.

Action	Nb	D (min)
Create a speaker label	295	62.4
Change the speaker label	463	58.7
Create a boundary	986	197.2
Delete a boundary	156	13.3

Table 3: Correction for the oracle system - REPERE test 2013. Nb: Number of occurrences; D: duration estimation (Avg duration $\times$ number of occ.)

The occurrence number of the segmentation actions is about one and a half greater than the clustering action number (respectively 1142 and 758). Segmentation corrections represent about 65% of the total correction time (210.5 minutes). The "*Create a boundary*" action is the most costly action, since it corresponds to about 52% of the overall corrections. For an audio recording of 2h57 (177 min), an annotator will take 3h17 (197.2 min) to create boundaries.

If the simulated annotator only corrects the clustering errors, the DER is 5.59% at the end of the correction process. These 5.59% errors are due to the wrong segmentation. This result shows that segmentation errors and clustering errors approximately contribute to 40% and 60% respectively of the DER. Comparatively, the segmentation errors correspond to the main correction cost in terms of HCIQ.

Corpus	HCIQ (min)	AT (min)	HCIQ_n
ESTER test 2003	477.2	592	0.81
ESTER test 2009	482.0	430	1.12
ETAPE test 2012	793.7	418	1.90
REPERE test 2013	331.6	177	1.87

Table 4: Comparison of HCIQ_n obtained from corrections of the oracle system on several corpora. AT: annotation time

Table 4 compares the HCIQ_n of the REPERE test 2013 corpus to others. It shows that the REPERE corpus is one of the corpora which requires the most corrections for a unit of time since it needs on average 1.87 minutes of human corrections for 1 minute of audio signal. Moreover, it demonstrates that the ETAPE and REPERE corpora, being mainly more composed of spontaneous speech (false starts, repetition, overlapped speech, interjections, etc (Bazillon et al., 2008)) than the ESTER corpora, are enhanced with high HCIQ_n.

6. Conclusions

In this paper, we proposed a framework to assess any interactive system of diarization taking into account the human corrections. The combination of four actions permits to describe the correction steps in a single way. We proposed a metric used to precisely determine the duration of each action in order to assess the cost of the human-computer interactions. The evaluation of an oracle system on REPERE test 2013 shows that segmentation corrections take longer than the clustering corrections. The results of the oracle demonstrates the importance of segmentation errors on the HCIQ and the DER as well. The correction of segmentation errors increases the HCIQ measure whereas it affects the DER in a negligible way. Only the correction of clustering errors directly decreases the DER measure. Future work will be focus on the development of an embedded speaker diarization system to reduce the correction time. Then, a study will be done to determine how to call a human with parsimony (i.e. in some parts of the document) by having a sufficient annotation of the document for a targeted application. The input diarization system will be modified as well, in particular the segmentation step.

7. Bibliographical References

Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., and Vinyals, O. (2012). Speaker diarization: A review of recent research. ieee-tsap, 20(2):356–370, Feb.

Arora, S., Nyberg, E., and Rosé, C. P. (2009). Estimating annotation cost for active learning in a multi-annotator environment. In Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing, pages 18–26. Association for Computational Linguistics.

Barras, C., Geoffrois, E., Wu, Z., and Liberman, M. (2001). Transcriber: development and use of a tool for assisting speech corpora production. Speech Communication, 33(1):5–22.

Bazillon, T., Estève, Y., and Luzzati, D. (2008). Transcription manuelle vs assistée de la parole préparé et spontanée. Revue TAL.

Bonastre, J.-F., Delacourt, P., Fredouille, C., Merlin, T., and Wellekens, C. (2000). A speaker tracking system based on speaker turn detection for nist evaluation. In Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on, volume 2, pages II1177–II1180. IEEE.

Broux, P.-A., Doukhan, D., Petitrenaud, S., Meignier, S., and Carrive, J. (2016). An active learning method for speaker identity annotation in audio recordings. In 1st International Workshop on Multimodal Media Data Analytics (MMDA), In conjuction with the 22nd European Conference on Artificial Intelligence (ECAI).

Budnik, M., Poignant, J., Besacier, L., and Quénot, G. (2014). Automatic propagation of manual annotations for multimodal person identification in tv shows. In Content-Based Multimedia Indexing (CBMI), 2014 12th International Workshop on, pages 1–4. IEEE.

Charhad, M., Moraru, D., Ayache, S., and Quénot, G. (2005). Speaker identity indexing in audio-visual documents. In Content-Based Multimedia Indexing (CBMI2005).

Dufour, R., Jousse, V., Estève, Y., Béchet, F., and Linarès, G. (2009). Spontaneous speech characterization and detection in large audio database. SPECOM, St. Petersburg.

Galibert, O. (2013). Methodologies for the evaluation of speaker diarization and automatic speech recognition in the presence of overlapping speech. In INTERSPEECH, pages 1131–1134.

Kahn, J., Galibert, O., Quintard, L., Carré, M., Giraudel, A., and Joly, P. (2012). A presentation of the repere challenge. In Content-Based Multimedia Indexing (CBMI), 2012 10th International Workshop on, pages 1–6. IEEE.

McCowan, I. A., Moore, D., Dines, J., Gatica-Perez, D., Flynn, M., Wellner, P., and Bourlard, H. (2004). On the use of information retrieval measures for speech recognition evaluation. Technical report, IDIAP.

Meignier, S. and Merlin, T. (2010). Lium spkdiarization: an open source toolkit for diarization. In CMU SPUD Workshop, volume 2010.

NIST. (2003). The rich transcription spring 2003 (RT-03S) evaluation plan. `http://www.itl.nist.gov/iad/mig/tests/rt/2003-spring/docs/rt03-spring-eval-plan-v4.pdf`, February.

Ordelman, R., De Jong, F., and Larson, M. (2009). Enhanced multimedia content access and exploitation using semantic speech retrieval. In Semantic Computing, 2009. ICSC'09. IEEE International Conference on, pages 521–528. IEEE.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In Proceedings of association for machine translation in the Americas, volume 200.

Vallet, F., Uro, J., Andriamakaoly, J., Nabi, H., Derval, M., and Carrive, J. (2016). Speech trax: A bottom to the top approach for speaker tracking and indexing in an archiving context. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC). European Language Resources Association (ELRA).

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). Elan: a professional framework for multimodality research. In Proceedings of LREC, volume 2006, page 5th.

Wood, M. E. and Lewis, E. (1996). Windmill-the use of a parsing algorithm to produce predictions for disabled persons. PROCEEDINGS-INSTITUTE OF ACOUSTICS, 18:315–322.

Performance Impact Caused by Hidden Bias of Training Data for Recognizing Textual Entailment

Masatoshi Tsuchiya
Toyohashi University of Technology
1–1 Hibarigaoka, Tempaku-cho, Toyohashi, Aichi, Japan
tsuchiya@imc.tut.ac.jp

Abstract

The quality of training data is one of the crucial problems when a learning-centered approach is employed. This paper proposes a new method to investigate the quality of a large corpus designed for the recognizing textual entailment (RTE) task. The proposed method, which is inspired by a statistical hypothesis test, consists of two phases: the first phase is to introduce the predictability of textual entailment labels as a null hypothesis which is extremely unacceptable if a target corpus has no hidden bias, and the second phase is to test the null hypothesis using a Naive Bayes model. The experimental result of the Stanford Natural Language Inference (SNLI) corpus does not reject the null hypothesis. Therefore, it indicates that the SNLI corpus has a hidden bias which allows prediction of textual entailment labels from hypothesis sentences even if no context information is given by a premise sentence. This paper also presents the performance impact of NN models for RTE caused by this hidden bias.

Keywords: Recognizing Textual Entailment; Evaluation Methodology; Naive Bayes

1. Introduction

The quality of a training data is one of the crucial problems when a learning-centered approach including neural network (NN) is employed. (Reidsma and Carletta, 2008) demonstrated that annotation errors of the dialog act corpus, which follows a certain systematic pattern, mislead the learning result of Bayesian network. (Zhang et al., 2017) described that the capacity of a NN model is large enough for brute-force memorizing the entire data set, even if its labels are random. Thus, an influence for a NN model caused by a certain systematic pattern may become more serious than the influence for other learning-centered methods.

Both a method to improve the quality of a training data and a metric to evaluate its reliability are important. As the former method, majority vote of human annotators was widely used (Sabou et al., 2014; Bowman et al., 2015; Al Khatib et al., 2016) As the latter metric, many kind of inter-annotator agreement metrics based on multiple human annotation results were employed, because direct assessment of data quality is difficult (Craggs and Wood, 2005; Artstein and Poesio, 2008; Mathet et al., 2012).

This paper proposes a new empirical method to investigate a quality of a large corpus designed for the recognizing textual entailment (RTE) task (Condoravdi et al., 2003; Bos and Markert, 2005; MacCartney and Manning, 2009; Marelli et al., 2014a). The proposed method, which is inspired by a statistical hypothesis test, assesses a quality of a target corpus directly and does not depend on multiple human annotation results unlike the existing metrics. The proposed method consists of two phases: the first phase is to introduce the predictability of textual entailment (TE) labels as a null hypothesis which is extremely unacceptable when a target corpus has no hidden bias, and the second phase is to test the null hypothesis for the target corpus using a Naive Bayes (NB) model.

In order to demonstrate the efficiency of the proposed method, we investigate two RTE corpora: the Stanford Natural Language Inference (SNLI) corpus (Bowman et al., 2015) and the Sentences Involving Compositional Knowledge (SICK) corpus (Marelli et al., 2014b). Although the experimental result of the SICK corpus rejects the null hypothesis, the result of the SNLI corpus does not reject it. Thus, the SNLI corpus has a hidden bias to allow prediction of TE labels from hypothesis sentences even if no context information is given by a premise sentence. The other experiment shows that this hidden bias causes the risk that a NN model for RTE works as a entirely different model than its constructor expects.

s_1	Two boys are swimming in the pool.
s_2	Two girls are playing basketball.
s_3	Two women are swimming in the pool.
s_h	Two children are swimming in the pool.

Figure 1: Example sentences of RTE. The textual entailment label of s_h is determinable if and only if context information is given by a premise sentence.

The major contributions of this paper are following three points:

- This paper proposes a new empirical method to reveal a hidden bias of a large RTE corpus (Section 2.).
- This paper applies the proposed method on the SNLI corpus and the SICK corpus, and reveals the hidden bias of the SNLI corpus (Section 3.).
- This paper also presents that this hidden bias causes the risk that a NN model proposed for RTE works as a entirely different model than its constructor expects (Section 4.).

2. Proposed Method

This section describes the detail of the proposed method, which consists of two phases.

Predicted labels	Corpus labels		
	Entailment	Neutral	Contradiction
Entailment	2275	644	706
Neutral	508	1976	563
Contradiction	585	599	1968

(a) The confusion matrix obtained by the TE label prediction model trained and tested on the SNLI corpus. It tries to predict an appropriate TE label for each individual hypothesis sentence.

Predicted labels	Corpus labels		
	Entailment	Neutral	Contradiction
Entailment	3	3	2
Neutral	1411	2790	718
Contradiction	0	0	0

(b) The confusion matrix obtained by the TE label prediction model trained and tested on the SICK corpus. It simply tries the major TE label 'neutral' for almost all hypothesis sentences without prediction.

Figure 2: Confusion matrices of TE Label Prediction Models

2.1. Predictability of TE Labels without Premise Sentences

The first phase of the proposed method is to derive a null hypothesis which is extremely unacceptable when a target corpus has no hidden bias. We focus the task definition of the target corpus for this phase.

(Marelli et al., 2014a) defined RTE as a task to partition relationships between a premise sentence and a hypothesis sentence into three categories: entailment, neutral and contradiction. Consider the example sentences shown in Figure 1. When s_1 is given as a premise sentence and s_h is given as a hypothesis sentence, the relationship between s_1 and s_h is labeled entailment. The relationship between s_2 and s_h is labeled neutral, and the relationship between s_3 and s_h is labeled contradiction. These examples indicate that the TE label is determinable if and only if context information is given by a premise sentence. Based on this observation, the null hypothesis of the proposed method is defined as follows:

Definition of the null hypothesis

> The TE label of the hypothesis sentence is determinable without the premise sentence.

Because the null hypothesis looks extremely unacceptable, it is appropriate to reveal a hidden bias of the target RTE corpus.

2.2. TE Label Prediction Model

The second phase of the proposed method is to test statistical significance of the null hypothesis for a target corpus. This phase requires two models: the first model is the statistical model of the null hypothesis, henceforth referred to as the *TE label prediction model*, and the second model is the statistical model of the alternative hypothesis, henceforth referred to as the baseline model.

The TE label prediction model is a model which predicts TE labels for hypothesis sentences without context information of premise sentences. This paper employs a multinomial NB model (Wang and Manning, 2012) as the TE label prediction model, which is defined by the following equation:

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, P(y) \prod_{i=1}^{n} P(x_i|y), \tag{1}$$

where y is a TE label and x_i is a feature. This paper simply employs all word unigrams of hypothesis sentences as features.

Corpus	TE label prediction model	Baseline model
SNLI	**63.3%**	34.3%
SICK	56.7%	56.7%

Table 1: Performance of the TE label prediction model trained and tested on two RTE corpora. The TE label prediction model trained and tested on the SNLI corpus achieves statistically significant accuracy than the baseline model.

The baseline model assigns TE labels for hypothesis sentences when no information is given by either premise sentences or hypothesis sentences but only the TE label distribution $P(y)$ of the target corpus is available. In such case, it is reasonable to assign the TE label which occurs most frequently in the target corpus for all hypothesis sentences. This baseline assignment is defined as follows:

$$\breve{y} = \underset{y}{\operatorname{argmax}}\, P(y) \tag{2}$$

If there is a statistically significant difference between the TE label prediction model and the baseline model, the null hypothesis is not rejected for the target corpus, and it indicates that the target corpus contains a hidden bias. Otherwise, the null hypothesis is rejected for the target corpus,

3. Experiment

This section presents the detailed experimental conditions and the experimental results. The highlight of these results is that the TE label prediction model achieves 63.3% accuracy for the SNLI corpus.

Table 1 shows the performances of the TE label prediction models trained and tested on two RTE corpora[1]. The TE label prediction model, which is trained on the SNLI training hypothesis sentences and their TE labels, achieves 63.3% accuracy on the SNLI test hypothesis sentences without premise sentences. The baseline model based on the SNLI TE label distribution achieves 34.3% accuracy on the same hypothesis sentences. The sign test indicates that there is a statistically significant difference between these models ($p = 5.7e^{-202}$). On the other hand, the performance of the TE label prediction model trained and tested on the SICK corpus is close to the performance of the baseline model (56.7%). The sign test indicates that there is no statistically significant difference between these models ($p = 0.65$).

[1](Pedregosa et al., 2011) is employed to implement the TE label prediction model.

	E_e	H_e
Entailment	2,275 (36.6%)	1,093 (30.3%)
Neutral	1,976 (31.8%)	1,243 (34.5%)
Contradiction	1,968 (31.6%)	1,269 (35.2%)
Total	6,219 (63.3%)	3,605 (36.7%)

Table 2: Empirical classification of the SNLI corpus using the TE label prediction model

Figure 2 clearly shows the difference between the behavior of the TE label prediction model trained on the SNLI corpus and the model trained on the SICK corpus. The left matrix is obtained by the model trained and tested on the SNLI corpus, and the right matrix is obtained by the model trained and tested on the SICK corpus. The model of the SICK corpus simply tries the major TE label 'neutral' for almost all hypothesis sentences without prediction, although the model of the SNLI corpus tries to predict an appropriate TE label for each individual hypothesis sentence.

These results indicate that the null hypothesis is rejected for the SICK corpus, but it is not rejected for the SNLI corpus. Therefore, hypothesis sentences of the SNLI corpus have a hidden bias to allow prediction of their TE labels without premise sentences.

4. Discussion

As described in Section 3., hypothesis sentences of the SNLI corpus have a hidden bias to allow prediction of their TE labels without premise sentences. The magnitude of the performance impact caused by the hidden bias is important, because the SNLI corpus is widely used as training data by many NN models for RTE (Bowman et al., 2015; Rocktäschel et al., 2015; Yin et al., 2016; Mou et al., 2016a; Wang and Jiang, 2016; Liu et al., 2016a; Liu et al., 2016b; Cheng et al., 2016; Parikh et al., 2016; Sha et al., 2016). This section discusses the performance impact of the NN models caused by the hidden bias.

4.1. Empirical Classification of SNLI Corpus

The test pairs of the SNLI corpus are classified into two subsets using the TE label prediction model trained on the SNLI corpus. The first subset is the empirical easy test set E_e, which consists of all test pairs whose TE labels are predicted correctly by the TE label prediction model. The second subset is the empirical hard test set H_e, which consists of the rest pairs. Table 2 shows the classification result. 63.3% test pairs of the SNLI corpus were classified as E_e, and the rest pairs were classified as H_e.

4.2. Definitions of NN Models for RTE

Two NN models are prepared to evaluate performance impacts caused by the hidden bias. The first model (henceforth denoted as the parallel LSTM model) was proposed by (Bowman et al., 2015) for RTE, and was evaluated on the performance difference between RTE corpora by (Mou et al., 2016b). This model is defined by the following equations.

$$
\begin{aligned}
\boldsymbol{h}_{p,i} &= \text{LSTM}_p(W_e x_{p,i} + W_{hp}\boldsymbol{h}_{p,i-1}) \\
\boldsymbol{h}_{h,i} &= \text{LSTM}_h(W_e x_{h,i} + W_{hh}\boldsymbol{h}_{h,i-1}) \\
\boldsymbol{l}_1 &= \tanh(W_1[\boldsymbol{h}_{p,|\boldsymbol{x_p}|}, \boldsymbol{h}_{h,|\boldsymbol{x_h}|}] + B_1) \\
\boldsymbol{l}_2 &= \tanh(W_2\boldsymbol{l}_1 + B_2) \\
\boldsymbol{l}_3 &= \tanh(W_3\boldsymbol{l}_3 + B_3) \\
\boldsymbol{y} &= \text{softmax}(\boldsymbol{l}_3)
\end{aligned}
$$

The first step is to convert a premise sentence $\boldsymbol{x}_p$ and a hypothesis sentence $\boldsymbol{x}_h$ into embedding vectors using the word embedding matrix W_e, which is initialized with the 300-dimension reference GloVe vectors (Pennington et al., 2014). The second step is to convert embedding vectors into two 100-dimension sentence vectors with LSTMs, and they are concatenated into a 200-dimension vector. The remaining steps are to predict a TE label with three tanh fully connected layers and then to apply the softmax function.

The second model (henceforth denoted as the sequential LSTM model), which was proposed by (Rocktäschel et al., 2015) for RTE, is defined as follows.

$$
\begin{aligned}
\boldsymbol{h}_{p,i} &= \text{LSTM}_p(W_e x_{p,i} + W_{hp}\boldsymbol{h}_{p,i-1}) \\
\boldsymbol{h}_{h,0} &= \text{LSTM}_h(W_{hh}\boldsymbol{h}_{p,|\boldsymbol{x_p}|}) \\
\boldsymbol{h}_{h,i} &= \text{LSTM}_h(W_e x_{h,i} + W_{hh}\boldsymbol{h}_{h,i-1}) \\
\boldsymbol{l} &= \tanh(W_l\boldsymbol{h}_{h,|\boldsymbol{x_h}|} + B_l) \\
\boldsymbol{y} &= \text{softmax}(\boldsymbol{l})
\end{aligned}
$$

In the second model, two LSTMs are sequentially connected. Thus, it is possible to consider that the memory cells of these LSTMs are directly modeling a recognition process unlike the parallel LSTM model. All vectors of the sequential LSTM model are 100-dimension. Although (Rocktäschel et al., 2015) proposed the variants with attentions between a premise sentence and a hypothesis sentence, the attention-less model is employed in this experiment, because of its simplicity.

4.3. Performance Impact of NN Models for RTE

This subsection presents the big performance drop of the NN models caused by the hidden bias.

Table 3 shows the experimental results of these NN models trained and tested on the SNLI corpus. Although both NN models achieve high accuracy for the whole test set and for the empirical easy test set E_e, they achieve drastic low accuracy for the empirical hard test set H_e. These performance drops mean that a large portion of the high accuracy achieved by both NN models benefits from the empirical easy test set E_e.

Table 4 shows the performance which is achieved by the same NN models when all words of premise sentences are replaced by unknown word symbols. Because this replacement removes all context information from premise sentences, thus the performance of the NN models must drop close to the chance ratio, if the NN models decide TE labels based on context information of premise sentences. Despite this expectation, both NN models achieve obviously higher performance than the chance ratio for E_e (36.8%, shown in Table 2). This result indicates that both NN models do not

Model	Related models	$E_e \cup H_e$	E_e	H_e
Parallel LSTM model	76.3% (Mou et al., 2016b)	76.8%	87.8%	57.8%
Sequential LSTM model	80.9% (Rocktäschel et al., 2015)	81.4%	90.1%	65.6%

Table 3: Performance of the NN models for RTE. Performance drops from the empirical easy test set E_e to the empirical hard test set H_e are observed for both NN models.

Model	$E_e \cup H_e$	E_e	H_e
Parallel LSTM model	54.1%	66.0%	33.7%
Sequential LSTM model	48.6%	56.7%	34.7%

Table 4: Performance of the NN models for RTE, when all words of premise sentences are replaced by unknown word symbols. Although all context information is removed by this replacement, both NN models achieve obviously higher performance than the chance ratio for the empirical easy test set E_e. This result indicates that both NN models do not work as RTE models for E_e, but work as TE label prediction models for E_e.

	SNLI			SICK		
	Training	Development	Test	Training	Development	Test
Entailment	183,416 (33.4%)	3,329 (33.8%)	3,368 (34.3%)	1,299 (28.9%)	144 (28.8%)	1,414 (28.7%)
Neutral	182,764 (33.3%)	3,235 (32.9%)	3,219 (32.8%)	2,536 (56.4%)	282 (56.4%)	2,793 (56.7%)
Contradiction	183,187 (33.4%)	3,278 (33.3%)	3,237 (33.0%)	665 (14.8%)	74 (14.8%)	720 (14.6%)
Total	549,367	9,842	9,824	4,500	500	4,927

Table 5: Statistics of TE labels of two RTE corpora. Although TE labels of the SNLI corpus are balanced, TE labels of the SICK corpus are not balanced.

	SNLI	SICK
# of training pairs	55k	4,500
# of development pairs	10k	500
# of test pairs	10k	4,927
Premise mean token count	14.1	9.8
Hypothesis mean token count	8.3	9.5
Vocabulary size of training pairs	36,427	2,178
Vocabulary size of test pairs	6,548	2,188
OOV ratio of test pairs	0.25%	0.32%

Table 6: Comparison of the SNLI and SICK corpora. Both corpora are extremely similar from the view point of descriptive statistics.

work as RTE models for E_e, but work as TE label prediction models for E_e. This behavior of NN models for E_e must be entirely different than their constructor expected.

4.4. Comparison of SNLI and SICK corpora

The SNLI and SICK corpora, are entirely similar in their sentence domains, English scene descriptions. Both of them use the Flickr30k corpus (Young et al., 2014) as origins of their sentences. It is also exhibited by the small differences of sentence token mean counts as shown in Table 6. The second is about their vocabulary. The out-of-vocabulary (OOV) ratio of SICK test pairs is 0.15%, when words of SNLI training pairs are regarded as known. This small OOV ratio indicates that SICK test pairs and SNLI training pairs are quite close from the view point of their vocabulary.

The SNLI and SICK corpora are different in the method of composing sentences. Hypothesis sentences of the SNLI corpus are composed by human workers, but all sentences of the SICK corpus are derived from original sentences using hand-crafted rules. We think that this difference may be a cause of the hidden bias revealed by this paper.

5. Conclusion

This paper proposes a new empirical method to investigate the quality of large RTE corpus. The proposed method consists of two phases: the first phase is to introduce the predictability of TE labels as a null hypothesis, and the second phase is to test the null hypothesis using a NB model. The proposed method reveals a hidden bias of the SNLI corpus, which allows prediction of TE labels from hypothesis sentences without context information given by premise sentences.

This paper also presents that this hidden bias makes a large performance impact on the NN models for RTE. The experimental result shows that a large portion of the high accuracy achieved by the NN models benefits from the hidden bias. The other experimental result shows that a NN model trained on the SNLI corpus does not work as an RTE model, but works as a TE label prediction model, when biased test pairs are given. These results arise a risk that a complex NN model works as an entirely different model than its constructor expects.

6. Acknowledgments

A part of this research was supported by JSPS KAKENHI Grant No. 15K12097. I would like to express my sincere appreciation to Dr. Mitsuo Yoshida and Dr. Adam Meyers for their valuable comments.

7. Bibliographical References

Al Khatib, K., Wachsmuth, H., Kiesel, J., Hagen, M., and Stein, B. (2016). A news editorial corpus for mining argumentation strategies. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3433–3443,

Osaka, Japan, December. The COLING 2016 Organizing Committee.

Artstein, R. and Poesio, M. (2008). Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Bos, J. and Markert, K. (2005). Recognising textual entailment with logical inference. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 628–635, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September. Association for Computational Linguistics.

Cheng, J., Dong, L., and Lapata, M. (2016). Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas, November. Association for Computational Linguistics.

Condoravdi, C., Crouch, D., de Paiva, V., Stolle, R., and Bobrow, D. G. (2003). Entailment, intensionality and text understanding. In Graeme Hirst et al., editors, *Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning*, pages 38–45.

Craggs, R. and Wood, M. M. (2005). Evaluating discourse and dialogue coding schemes. *Computational Linguistics*, 31(3):289–296, September.

Liu, P., Qiu, X., Chen, J., and Huang, X. (2016a). Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1034–1043, Berlin, Germany, August. Association for Computational Linguistics.

Liu, P., Qiu, X., Zhou, Y., Chen, J., and Huang, X. (2016b). Modelling interaction of sentence pair with coupled-lstms. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1703–1712, Austin, Texas, November. Association for Computational Linguistics.

MacCartney, B. and Manning, C. D. (2009). An extended model of natural logic. In *Proceedings of the Eight International Conference on Computational Semantics*, pages 140–156, Tilburg, The Netherlands, January. Association for Computational Linguistics.

Marelli, M., Bentivogli, L., Baroni, M., Bernardi, R., Menini, S., and Zamparelli, R. (2014a). Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 1–8, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., and Zamparelli, R. (2014b). A SICK cure for the evaluation of compositional distributional semantic models. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Mathet, Y., WidlÃ¶cher, A., Fort, K., François, C., Galibert, O., Grouin, C., Kahn, J., Rosset, S., and Zweigenbaum, P. (2012). Manual corpus annotation: Giving meaning to the evaluation metrics. In *Proceedings of COLING 2012: Posters*, pages 809–818, Mumbai, India, December. The COLING 2012 Organizing Committee.

Mou, L., Men, R., Li, G., Xu, Y., Zhang, L., Yan, R., and Jin, Z. (2016a). Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 130–136, Berlin, Germany, August. Association for Computational Linguistics.

Mou, L., Meng, Z., Yan, R., Li, G., Xu, Y., Zhang, L., and Jin, Z. (2016b). How transferable are neural networks in nlp applications? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 479–489, Austin, Texas, November. Association for Computational Linguistics.

Parikh, A., Täckström, O., Das, D., and Uszkoreit, J. (2016). A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255, Austin, Texas, November. Association for Computational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Reidsma, D. and Carletta, J. (2008). Reliability measurement without limits. *Computational Linguistics*, 34(3):319–326, September.

Rocktäschel, T., Grefenstette, E., Hermann, K. M., Kociský, T., and Blunsom, P. (2015). Reasoning about entailment with neural attention. In *Proceedings of The International Conference on Learning Representations (ICLR2016)*.

Sabou, M., Bontcheva, K., Derczynski, L., and Scharl, A. (2014). Corpus annotation through crowdsourcing: Towards best practice guidelines. pages 859–866.

Sha, L., Chang, B., Sui, Z., and Li, S. (2016). Reading and thinking: Re-read lstm unit for textual entailment recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2870–2879, Osaka, Japan, De-

cember. The COLING 2016 Organizing Committee.

Wang, S. and Jiang, J. (2016). Learning natural language inference with lstm. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1442–1451, San Diego, California, June. Association for Computational Linguistics.

Wang, S. and Manning, C. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 90–94, Jeju Island, Korea, July. Association for Computational Linguistics.

Yin, W., SchÃ$\frac{1}{4}$tze, H., Xiang, B., and Zhou, B. (2016). Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *Transactions of the Association for Computational Linguistics*, 4:259–272.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2017). Understanding deep learning requires rethinking generalization. In *Proceedings of The International Conference on Learning Representations (ICLR2016)*.

Evaluation of Croatian Word Embeddings

Lukáš Svoboda[1], **Slobodan Beliga**[2]
1) Department of Computer Science and Engineering, University of West Bohemia
Univerzitní 22, 306 14 Plzeň, Czech Republic
2) Department of Informatics, University of Rijeka
Radmile Matejčić 2, 51000 Rijeka, Croatia
svobikl@kiv.zcu.cz[1], sbeliga@inf.uniri.hr[2]

Abstract

Croatian is poorly resourced and highly inflected language from Slavic language family. Nowadays, research is focusing mostly on English. We created a new word analogy dataset based on the original English *Word2vec* word analogy dataset and added some of the specific linguistic aspects from the Croatian language. Next, we created Croatian WordSim353 and RG65 datasets for a basic evaluation of word similarities. We compared created datasets on two popular word representation models, based on *Word2Vec* tool and *fastText* tool. Models have been trained on 1.37B tokens training data corpus and tested on a new robust Croatian word analogy dataset. Results show that models are able to create meaningful word representation. This research has shown that free word order and the higher morphological complexity of Croatian language influences the quality of resulting word embeddings.

Keywords: Croatian word embeddings, Croatian word analogy, Croatian language, Slavic language family, Word2Vec, FastText, Croatian word similarity dataset, WordSim353, RG65

1. Introduction

Word2Vec and FastText are tools that create models representing words as vectors of real numbers from high-dimensional space. Word representations are based on Distributional Hypothesis (Harris, 1954), where the context for each word is given by its nearby words. The goal of such representations is to capture the syntactic and semantic relationship between words.

It was shown that the word vectors can be successfully used in order to improve and/or simplify many NLP applications (Collobert and Weston, 2008; Collobert et al., 2011). There are also NLP tasks, where word embeddings does not help much (Andreas and Klein, 2014).

Most of the work is focused on English. Recently the community has realized that the research should focus on other languages with rich morphology and different syntax (Berardi et al., 2015; Elrazzaz et al., 2017; Köper et al., 2015; Svoboda and Brychcín, 2016), but there is still a little attention to languages from Slavic family. These languages are highly inflected and have a relatively free word order. Since there are open questions related to the embeddings in the Slavic language family, in this paper, we will focus mainly on Croatian word embeddings, from the South Slavic language family. With the aim of expanding existing findings about Croatian word embeddings, in this paper we will:

1. Compare different word embeddings methods on Croatian language which is not deeply explored, and according to its features, belongs to highly inflected language (words can have seven different cases for singular and seven for plural, genders, and numbers).

2. For the purposes of the word embeddings experiments, we will create three new datasets. Two basic word similarity datasets based on original WordSim353 (Finkelstein et al., 2002) and RG65 (Rubenstein and Goodenough, 1965) will be translated to the Croatian. Except for the similarity between words, we would like to explore other semantic and syntactic properties which are hidden in word embeddings. A new evaluation scheme based on word analogies were presented in (Mikolov et al., 2013a). Based on this popular evaluation scheme, we will create a Croatian version of original *Word2Vec* analogy dataset in order to qualitatively compare the performance of different models.

3. Empirically compare the results obtained from the Croatian language, which belongs to the group of Balto-Slavic (subgroup: Slavic) languages, to the results obtained from the English – the most commonly studied language, which belongs to the group of Germanic language family (subgroup: West).

Nowadays, word embeddings are typically obtained as a product of training neural network-based language models. Language modeling is a classical NLP task of predicting the probability distribution over the "next" word. In these models, a word embedding is a vector in $\mathbb{R}^n$, with the value of each dimension being a feature that weights the relation of the word with a "latent" aspect of the language. These features are jointly learned from plain unannotated text data. This principle is known as the *Distributional Hypothesis* (Harris, 1954). The direct implication of this hypothesis is that the word meaning is related to the context where it usually occurs and thus it is possible to compare the meanings of two words by statistical comparisons of their contexts. This implication was confirmed by empirical tests carried out on human groups in (Rubenstein and Goodenough, 1965; Charles, 2000).

There is a variety of datasets for evaluating semantic relatedness between English words, such as *WordSimilarity-353* (Finkelstein et al., 2002), *Rubenstein and Goodenough (RG)* (Rubenstein and Goodenough, 1965), *Rare-words* (Luong et al., 2013), *Word pair similarity in con-*

text (Huang et al., 2012), and many others. Mikolov et al. reported in (Mikolov et al., 2013a) that word vectors trained with a simplified neural language model (Bengio et al., 2006) encodes syntactic and semantic properties of language, which can be recovered directly from space through linear translations, to solve analogies such as: $\vec{king} - \vec{man} = \vec{queen} - \vec{woman}$. Evaluation scheme based on word analogies were presented in (Mikolov et al., 2013a).

To the best of our knowledge, only small portion of recent studies attempted evaluating Croatian word embeddings. In a review of works that evaluate syntactic and semantic analogies, we have encountered only a few datasets. In (Zuanović et al., 2014) authors translated small portion from English analogy dataset to Croatian in order to evaluate their Neural based model. However, this translation of syntactic analogy reasoning dataset was only made for a total of 350 questions based on positive-comparative form relationship in adjectives. In addition to syntactic, they also prepare semantic analogy reasoning dataset. It was based on countries and their capitals, originally proposed for the English by (Mikolov et al., 2013a) and translated into Croatian. The dataset comprises 506 entries. Recently Mrkšić et al. also trained word embeddings (Mrkšić et al., 2017), and produced a translation in the Croatian language and reannotation of gold standard resource SimLex-999 (Hill et al., 2015) with 999 word pairs.

There is only one analogy dataset representing Slavic language family – Czech word analogy dataset presented in (Svoboda and Brychcín, 2016).

In general, many methods have been proposed to learn such word vector representations. One of the Neural Network based models for word vector representation which outperforms previous methods on word similarity tasks was introduced in (Huang et al., 2012). Word embeddings methods implemented in tool *Word2Vec* (Mikolov et al., 2013a) and *GloVe* (Pennington et al., 2014) significantly outperform other methods for word embeddings. Word vector representations made by these methods have been successfully adapted on a variety of core NLP tasks. Recent library *fastText* (Bojanowski et al., 2017) tool is derived from *Word2Vec* and enriches word embeddings vectors with subword information.

2. Proposed Datasets

Original *Word2Vec* analogy dataset is composed of 19,558 questions divided into two tested group: semantic and syntactic questions, e.g. king : man = queen : woman. The fourth word in question is typically the predicted one.

Our Croatian analogy dataset has 115,085 question divided in the same manner as for English into two tested group: semantic and syntactic questions. Dataset has been created by three annotators (two native speakers).

Semantic questions are divided into 9 categories, each having around 20-100 word question pairs. Combination of question pairs gives overall 36,880 semantics questions:

- `capital-common-countries`: This group consist of 23 the most common countries. These countries were adopted from original *Word2Vec* analogies and having the highest number of occurrences in the text between all languages.
- `chemical-elements`: Represents 119 pairs of chemical elements with their shortcut symbol (e.g. O – Oxygen).
- `city-state`: Gives 20 regions (states) inside Croatia and gives one of city example in such region.
- `city-state-USA`: 67 pairs of cities and corresponding states in the USA. This category is adopted from original English word analogy test.
- `country-world`: 118 pairs of countries with main cities from all over the world. Translated from original *Word2Vec* analogies.
- `currency-shortcut`: 20 pairs of state currencies with its shortcut name (e.g. Switzerland – CHF).
- `currency`: 20 pairs of states with their currencies (e.g. Japan – yen). Translated from original EN analogy dataset.
- `eu-cities-states`: 40 word pairs of states from EU and their corresponding main city (e.g. Belgium – Brussels).
- `family`: 41 word pairs with family relation in masculine vs. feminine form (e.g. brother – sister).

Syntactic part of the dataset is divided into 14 categories, consisting of 78,205 questions:

- `jobs`: This category is language-specific, consist of 109 pairs of job positions in masculine $\times$ feminine form.
- `adjective-to-adverb`: 32 pairs of adjectives and its representatives in adverb form.
- `opposite`: 29 pairs of adjectives with its opposites. This category collects words from which is easy to make its opposites usually with preposition "un" or "in", the respective preposition "ne" in Croatian (e.g. certain – uncertain). Adopted from original EN word analogies.
- `comparative`: 77 pairs of adjectives and its comparative form (e.g. good – better).
- `superlative`: 77 pairs of adjectives and its superlative form.
- `nationality-man`: 84 pairs of states and humans representing its nationalities in masculine form. (e.g. Switzerland – Swiss).
- `nationality-female`: 84 pairs of states and its nationalities in feminine form. This is language specific.
- `past-tense`: 40 pairs of verbs and its past tense form.

- `plural`: 46 pairs of nouns and its plural form.

- `nouns-antonyms`: 100 pairs of nouns and its antonyms.

- `adjectives-antonyms`: Similar to *opposite* category, it consists of 96 word pairs of adjectives and their antonyms. However, words are much more complex (e.g. good – bad).

- `verbs-antonyms`: 51 pairs of verbs and its antonyms.

- `verbs-pastToFemale`: 83 pairs of verbs and its past tense in feminine form. This category is extended from category *past-tense* and is language-specific.

- `verbs-pastToMale`: 83 pairs of verbs and its past tense masculine form. This category is the same as *past-tense*, only its extended variation to be comparable with category *verbs-pastToFemale*.

2.1. Word Similarities Corpora

For basic comparison with English, we have translated state-of-the-art English word similarity datasets WordSim353 (Finkelstein et al., 2002) and RG65 (Rubenstein and Goodenough, 1965). These datasets have 353 and 65 word pairs respectively. Each word pair is manually annotated with similarity. We kept similarities untouched. The words in WordSim353 are assessed on a scale from 0 to 10, in RG65 from 0 to 5.

3. Distributional Semantic Models

We experimented with state-of-the-art models used for generating word embeddings. Neural network based models *CBOW* and *Skip-gram* from *Word2Vec* (Mikolov et al., 2013a) tool and tool *fastText* that promises better score for morphologically rich languages.

3.1. CBOW

CBOW (Continuous Bag-of-Words) (Mikolov et al., 2013a) tries to predict the current word according to the small context window around the word. The architecture is similar to the feed-forward NNLM (Neural Network Language Model) which has been proposed in (Bengio et al., 2006). The NNLM is computationally expensive between the projection and the hidden layer. Thus, *CBOW* proposed an architecture, where the (non-linear) hidden layer is removed and projection layer is shared between all words. The word order in the context does not influence the projection. This architecture also proved low computational complexity.

3.2. Skip-gram

Skip-gram architecture is similar to *CBOW*. Although instead of predicting the current word based on the context, it tries to predict a word's context based on the word itself (Mikolov et al., 2013b). Thus, the intention of the *Skip-gram* model is to find word patterns that are useful for predicting the surrounding words within a certain range in a sentence. *Skip-gram* model estimates the syntactic properties of words slightly worse than the *CBOW* model, but it is much better for modeling the word semantics on English test set (Mikolov et al., 2013a) (Mikolov et al., 2013b). Training of the *Skip-gram* model does not involve dense matrix multiplications and that makes training also extremely efficient (Mikolov et al., 2013b).

3.3. FastText

FastText (Bojanowski et al., 2017) combines concepts of *CBOW* (resp. *Skip-gram*) architectures introduced earlier in Section 3.1. and 3.2. These include representing sentences with bag-of-words and bag-of-n–grams, as well as using subword information, and sharing information across classes through a hidden representation.

4. Experimental Results

4.1. Training Data

We trained our models on two datasets in the Croatian language. Initially, we made the entire dump of Croatian Wikipedia – dated from August 2017 with approximately 275,000 articles. We have tokenized the text, removed non-alphanumeric tokens and extracted only sentences with at least 5 tokens. Resulting corpus has 92,446,973 tokens. Secondly, we merged data from Wikipedia with Croatian corpus presented in (Šnajder et al., 2013), and originally proposed in (Ljubešić and Erjavec, 2011), that has over 1.2B tokens. Resulting corpus has 1.37 tokens and 56,623,398 sentences. Such corpus has a vocabulary of 955,905 words with at least 10 occurrences.
For the English version of data, we used Wikipedia dump from June 2016. This dump was made of 5,164,793 articles and has 2.2B tokens. We tested analogies and similarity corpora for both languages with most frequent 300,000 words.

	Vocabulary $tf > 10$	Tokens	
EN corpus	3,234,907	2,201,735,114	
HR corpus	955,905	1,370,836,176	

Table 1: Properties of Croatian (HR corpus) and English (EN corpus) training data.

In total, we tested models on 68,986 out of 115,085 questions. It means that almost 40% question was unknown by the model. All question contained OOV words were discarded from the testing process. We tested *semantic* group on 16,968 known questions and part of corpus testing *syntactic* properties were measured on 52,018 questions.
Only 10 out of 353 question was unknown for *WordSim353* corpus and all 65 questions of *RG65* were in vocabulary. Unknown words in *WordSim353* were represented as word vector averaged from 10 least common words in a vocabulary.

Semantic tests reveal overall poor performance on all tested models, as we can see in Table 2. The opposite is true for English, where semantic tests give usually similar score as syntactic tests. This behavior we already saw on Czech corpus presented in (Svoboda and Brychcín, 2016). It seems that free word order and other properties of highly inflected languages from Slavic family have a big

Model	CBOW	Skip-gram	fastText-Skip	fastText-CBOW
Capital	44.17	62.5	59.58	21.25
Chemical-elements	1.02	2.25	0.74	0.41
City-state	22.11	37.89	47.63	46.32
City-state-USA	5.78	8.23	4.30	0.37
Country-world	23.93	44.49	40.15	7.31
Currency	4.68	8.19	6.43	0.58
Currency-shortcut	2.08	8.19	2.50	0.42
EU-cities-states	21.59	41.95	42.33	6.16
Family	34.83	41.82	42.72	34.76
Jobs	68.94	64.06	88.54	95.45
Adj-to-adverb	18.36	21.36	35.33	62.01
Opposite	17.34	18.05	59.03	86.10
Comparative	34.90	33.57	43.22	41.46
Superlative	33.22	27.70	40.50	51.77
Nationality-man	17.01	23.87	60.05	62.13
Nationality-female	14.38	55.66	57.77	53.98
Past-tense	67.31	61.03	66.67	78.21
Plural	37.12	44.65	44.24	35.10
Nouns-ant.	12.70	10.96	10.80	21.24
Adjectives-ant.	13.39	13.11	18.59	12.59
Verbs-antonyms	9.18	6.18	7.25	9.71
Verbs-pastFemale	60.92	19.47	71.04	80.50
Verbs-pastMale	66.68	62.89	76.04	85.04
SEMANTICS_EN	73.63	83.64	68.77	68.27
SYNTACTIC_EN	67.55	66.8	67.94	76.58
SEMANTICS_HR	16.60	28.54	25.94	7.76
SYNTACTIC_HR	37.06	35.63	49.60	54.56
ALL_HR	32.03	33.89	43.83	43.13

Table 2: Detailed results of Croatian word analogy dataset (the results of the semantic test at the top of the table; the results of the syntactic test in the middle part of the table; total results for English and Croatian at the bottom of the table).

	English		
Models	WordSim353	RG65	EN-analogies
CBOW	57.94	68.69	69.98 (44.02)
Skip-gram	64.73	78.27	73.57 (46.28)
fastText-Skip	46.13	76.31	68.27 (42.94)
fastText-CBOW	44.64	73.64	76.58 (48.17)
	Croatian		
CBOW	37.61	52.01	32.03 (19.19)
Skip-gram	52.16	58.47	33.89 (20.31)
fastText-Skip	52.98	64.31	43.83 (25.79)
fastText-CBOW	30.41	51.06	43.14 (25.79)

Table 3: Comparison with English models. Measurement in brackets gives the results including OOV questions.

impact on the performance of current state-of-the-art word embeddings methods.

From results of *City-state* and *City-state-USA* category it can be seen that knowledge of the topic in training data has the significant impact on the performance of a model. We wanted to show differences between two similar categories in case we have an insufficient amount of training data covering a particular topic. Category *City-state* is showing that model is able to carry such knowledge – if the topic is sufficiently represented in a training data, the model is able to carry this type of information. This behavior is seen in regions from Croatia mentioned in many articles on Croatian Wikipedia, but this was not a case with states from the USA. All questions of *City-state* were covered, but only around 50% of questions in category *City-state-USA* were in vocabulary. On categories *Country-world* and *EU-cities-states* it can be seen that there is no difference between knowledge about states and main cities from EU again state-city pairs from all over the world. Another very poor performance gives group *Currency*, but this group is usually weak across all languages and shows the weaknesses of the model.

Syntactic tests reveal better performance than tests oriented to semantic, but they still have significantly worse performance rather than on English. This part of corpus includes language-specific group of tests - such as *Verbs-pastMale/Female*, *Nationality-man/female*. Simple *Past-tense* tests give surprisingly high score – similar findings were presented for the Czech language in (Svoboda and Brychcín, 2016). We could say, that languages from Slavic family tend to have easier patterns for past tense. From language-specific groups we see that slightly better score is given in categories with word pairs in the masculine form, these results also correspond with the fact that there are more articles written in the masculine form in the training data.

4.2. Testing Data

As previously mentioned, in our experiments two word similarity datasets were used WordSim353 (Finkelstein et al., 2002) and RG65 (Rubenstein and Goodenough, 1965). In this subsection, we will discuss the issues we have encountered in the process of creating datasets by translating the original English versions into Croatian. The translation process can bring noise to data very insensibly due to difference and specificity contained in two languages (Croatian and English). Some specific cases are as follows:

- **mapping M:1 (the problem caused by lack of context)**: two (or more) words in the English language can have slightly different meanings in different contexts. In the translation process of such words to the Croatian language without taking into account the context in which the word appears, it is difficult to determine which translation is correct. To be quite clear, the context is missing because of the word-tuples form preserved in the dataset instead of the plain text form. For example, words *coast* and *shore* both appear in the dataset but in translation to the Croatian (without the context) both have one common meaning and unique translation *obala*. This problem could be entitled as 2:1 mapping (2 original words and 1 target). The problem of such mapping can be solved easily by using synonyms *priobalje* or *kopno*. The situation is unfavorable in case we use a synonym for translation as well as in the case we use the same Croatian word in the translation of two different English words. In both cases, we are not sure how much noise is introduced into the data.

- **the absence of a synonym pair**: this is a special case of the problem mentioned in the previous point as mapping M:1. In particular, two or more words have the same translation into the target language. However, the problem can not be solved by a synonym pair because it does not exist. For example, both *midday* and *noon* appear in our English version of the dataset,

but in the Croatian, they can be translated only as *podne*. However, *midday* could be translated differently (as *podnevni*), but in such translation, it becomes an adjective rather than a noun. Such a scenario certainly may allow noise introduction into the dataset, and therefore is not desirable.

- **mapping 1:M (the problem caused by lack of context)**: a problem is similar to the one mentioned above, with the difference in mapping between source and target words (1 original word from the English dataset can be translated into two or more Croatian words), and translation again depends on the context. For example, English word *drug* can be mapped into the Croatian words *lijek* (drug with the positive connotation; used for medical purposes, i.e. medicament) or *droga* (drug with the negative connotation; causes addiction, i.e. narcotic), depends again on missing context.

- **the problem of using multiple words in the translation**: for some English terms, there are no translations consist of just one word in Croatian, instead, two or more words must be used (phrase; set of words). For example, *seafood* can be translated with syntagma of two words *morski plodovi* or *plodovi mora*.

- **the problem caused by cultural differences**: In the geographical area of the Croatian, established word for *football* and *soccer* is *nogomet*, unlike the US area where there is a clear difference between these two words and sports. In the example of our dataset, both words *football* and *soccer* are present. Again, we can use a different Croatian translation *ragbi*, but we use the risk of introducing the noise into the dataset.

- **the problem of non-standard words (slang)**: some words in the slang may have different meanings than those in the standard language. For example, word *cock* can belong to the standard language but also to the slang, and depending on it has two meanings and two corresponding translations into the Croatian language: *pjetao* (kind of a bird) or *penis* (genitalia).

Besides, it is important to emphasize that although the invested efforts and high linguistic expertise, the created dataset may have unintentionally included noise into the data, which is inevitable due to restrictions in translation caused by specificities in different languages.

5. Conclusion

In this paper, evaluation of Croatian word embeddings is performed. New word embeddings are derived using different models. Additionally, some of the specific linguistic aspects of the Croatian language was added. Two popular word representation models were compared, *Word2Vec* and *fastText*. Models have been trained on a new robust Croatian analogy dataset. WordSim353 and RG65 datasets were translated from the English to the Croatian, in order to perform basic semantic measurements. Results show that models are able to create meaningful word representation.

However, it is important to note that this paper presents the first comparative study of word embeddings for Croatian and English, and therefore, new insights for NLP community according to the behavior of the Croatian word embeddings. The Croatian language belongs to the group of Slavic languages and has only preliminary and basic knowledge insights from word embeddings. In addition, another contribution of this work is certainly new datasets for the Croatian language, which are publicly available from: `https://github.com/Svobikl/cr-analogy`. It is also worth mentioning that these are also the first parallel English-Croatian word embeddings datasets.

Finally, we can figure out from experiments that models for Croatian does not achieve such good results as for English. In fact, results are mostly lower for the Croatian than for the English, with the exception of one case: fastText-Skip for WordSim-353. Such results can be explained theoretically through two perspectives – technical and linguistic.

1) Firstly, **the technical one** rests on the fact related to the corpus statistics used in the experiments (i.e. the size of the training corpus). It is evident that there is more English training data than Croatian. Therefore, expectations could be higher for English than for Croatian.

2) Secondly, **the linguistic one** withdraws its arguments from two sources.

   a) **Testing data** is the first one. In the process of translating datasets (testing data) from the English to the Croatian, there are possibilities of unintentional entering of the noise into the data (for example, by using synonyms) which hance make the task harder. Due to this fact it is reasonable to expect slightly worse results for Croatian.

   b) **Training data** is the second one. Croatian an English corpuses used for the training have serious differences in morphological complexity according to regularities of the Croatian and the English language. In particular, the difference in English (Germanic language) and Croatian (Slavic language) morphology is huge. Compared to the Croatian language, English language morphology is considerably poor. The Croatian language is a highly inflected language with mostly free word ordering in sentence structure, unlike the English, which is inflectional language and has a strict word ordering in a sentence (*subject-verb-object*). For example, three Croatian words are enough to construct two different sentence constructions with the same meaning: "*Ana voli Milovana.*" and "*Milovana voli Ana.*". Unlike the English, which requires up to 5 words for the same language construction: "*Ana loves Milovan.*" and "*Milovan is loved by Ana.*". These differences are reflected in the results of embeddings modeling. It is plausible that higher degree of inflection leads to higher data sparsity, which could reduce performance. Models presented in

this paper give good approximations to the English, they are better tailored to the English language morphology and better match the structure of such a language.

The most important conclusion of this research suggests that models for the Croatian do not achieve such good results as for the English. According to (Svoboda and Brychcín, 2016), this is also true for the Czech language, another one from Slavic language family. Following this, we would like to point out that future research should be focused on model improvements for Slavic languages. It would be worth to explore which Slavic languages specificities would be advisable to incorporate into models, in order to achieve better modeling of complex morphological structures. On the other hand, corpora preprocessing which simplifies morphological variations (and reduces data sparsity), such as stemming or lemmatization procedures, could also have an effect on word embeddings and should be one of the future research directions. Besides, we would also like to further investigate properties of other models for word embeddings and try to use external sources of information (such as part-of-speech tags, referenced information on Wikipedia, etc.) and experiment with the tree structure of sentence during the training process.

6. Acknowledgements

This work was supported by the project LO1506 of the Czech Ministry of Education, Youth and Sports and by Grant No. SGS-2016-018 Data and Software Engineering for Advanced Applications.

7. Bibliographical References

Andreas, J. and Klein, D. (2014). How much do word embeddings encode about syntax? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 822–827, Baltimore, Maryland, June. Association for Computational Linguistics.

Bengio, Y., Schwenk, H., Senécal, J.-S., Morin, F., and Gauvain, J.-L. (2006). Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer.

Berardi, G., Esuli, A., and Marcheggiani, D. (2015). Word embeddings go to italy: A comparison of models and training datasets. In *IIR*.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Charles, W. G. (2000). Contextual correlates of meaning. *Applied Psycholinguistics*, 21(04):505–524.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. P. (2011). Natural language processing (almost) from scratch. *CoRR*, abs/1103.0398.

Elrazzaz, M., Elbassuoni, S., Shaban, K., and Helwe, C. (2017). Methodical evaluation of arabic word embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 454–458.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Harris, Z. (1954). Distributional structure. *Word*, 10(23):146–162.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with genuine similarity estimation. *Comput. Linguist.*, 41(4):665–695, December.

Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 873–882, Stroudsburg, PA, USA. Association for Computational Linguistics.

Köper, M., Scheible, C., and im Walde, S. S. (2015). Multilingual reliability and" semantic" structure of continuous word spaces. In *IWCS*, pages 40–45.

Ljubešić, N. and Erjavec, T. (2011). hrwac and slwac: Compiling web corpora for croatian and slovene. In Ivan Habernal et al., editors, *Text, Speech and Dialogue - 14th International Conference, TSD 2011, Pilsen, Czech Republic, September 1-5, 2011. Proceedings*, Lecture Notes in Computer Science, pages 395–402. Springer.

Luong, M.-T., Socher, R., and Manning, C. D. (2013). Better word representations with recursive neural networks for morphology. In *CoNLL*, Sofia, Bulgaria.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Mrkšić, N., Vulić, I., Ó Séaghdha, D., Leviant, I., Reichart, R., Gašić, M., Korhonen, A., and Young, S. (2017). Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, October.

Šnajder, J., Padó, S., and Agić, Ž. (2013). Building and evaluating a distributional memory for croatian. In *Proceedings of the 51st Annual Meeting of the Association*

for Computational Linguistics (Volume 2: Short Papers), volume 2, pages 784–789.

Svoboda, L. and Brychcín, T. (2016). New word analogy corpus for exploring embeddings of czech words. *CoRR*, abs/1608.00789.

Zuanović, L., Karan, M., and Šnajder, J. (2014). Experiments with neural word embeddings for croatian. In *Proceedings of the 9th Language Technologies Conference*, pages 69–72.

C-HTS: A Concept-based Hierarchical Text Segmentation approach

Mostafa Bayomi, Séamus Lawless
ADAPT Centre, Knowledge and Data Engineering Group,
School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland
bayomim@tcd.ie, seamus.lawless@scss.tcd.ie

Abstract

Hierarchical Text Segmentation is the task of building a hierarchical structure out of text to reflect its sub-topic hierarchy. Current text segmentation approaches are based upon using lexical and/or syntactic similarity to identify the coherent segments of text. However, the relationship between segments may be semantic, rather than lexical or syntactic. In this paper we propose *C-HTS*, a Concept-based Hierarchical Text Segmentation approach that uses the semantic relatedness between text constituents. In this approach, we use the explicit semantic representation of text, a method that replaces keyword-based text representation with concept-based features, automatically extracted from massive human knowledge repositories such as Wikipedia. *C-HTS* represents the meaning of a piece of text as a weighted vector of knowledge concepts, in order to reason about text. We evaluate the performance of *C-HTS* on two publicly available datasets. The results show that *C-HTS* compares favourably with previous state-of-the-art approaches. As Wikipedia is continuously growing, we measured the impact of its growth on segmentation performance. We used three different snapshots of Wikipedia from different years in order to achieve this. The experimental results show that an increase in the size of the knowledge base leads, on average, to greater improvements in hierarchical text segmentation.

Keywords: Hierarchical Text Segmentation, Explicit Semantic Analysis, Semantic Relatedness, Wikipedia

1. Introduction

Text segmentation aims to divide text into coherent segments which reflect the sub-topic structure of the text. It is widely used as a pre-processing task for Information Retrieval (Prince and Labadié, 2007) and several Natural Language Processing (NLP) tasks such as Text Summarization (Boguraev and Neff, 2000) and Question Answering (Tellex et al., 2003). Text segmentation is essentially based on measuring the coherence between atomic units of text (sentences or paragraphs). Some approaches use lexical similarity, which compares units of text that are represented as vectors of the same, or similar, words (Choi, 2000). These approaches are limited, in that they rely on endogenous knowledge extracted from the documents themselves. Relying on such knowledge does not reveal much about the meaning beyond the text.

Some approaches started to enrich the text representation by exploiting its semantic meaning by using the Latent Semantic Analysis (LSA) (Choi et al., 2001). However, these approaches require a very large corpus, and consequently the pre-processing effort required is significant. On the other hand, some other approaches started to use external resources such as WordNet to enrich text (Stokes et al., 2004). However, such resources cover only a small fragment of the language lexicon. Furthermore, the use of such lexical resources offers little knowledge about the different word representations.

Buchanan and Feigenbaum (1982) stated that: "*The power of an intelligent program to perform its task well depends primarily on the quantity and quality of knowledge it has about that task.*" Hence, in this research, we are trying to enrich the text representation by replacing traditional text representation methods with a concept-based representation that exploits an external knowledge base to reveal more knowledge about the text.

When a person reads a text, the eyes read the words (the lexical representation of text) and send these words to the human's cognitive system, the brain. The brain starts to make sense of these words based on the knowledge of the reader. For example, the name "*Albert Einstein*" in a text document is read by the eyes and then sent to the brain, which starts to map the name to the different concepts that the person knows about Einstein such as: "*Theory of Relativity*", "*Physics*", "*Nobel Prize*", etc. The information that the brain maps the name to, is dependent upon how much knowledge this person has. If the individual does not know about Einstein, the brain would make no sense of that name. The individual could potentially ask other people who have different collections of knowledge for assistance, creating an intellectual representation through collaboration. In this research, we are trying to recreate this methodology in a segmentation algorithm. It is our contention that using this approach to understand text would make a more accurate approach to text segmentation.

The essential task in any text segmentation algorithm is to measure the coherence between two adjacent text blocks. Being inherently limited to lexical representation, current approaches cannot reveal much about the coherence between text blocks. Consider the following two sentences for example:

- Albert Einstein is a German scientist who was born on the 14th of March 1879.
- Mileva Marić was born on December 19, 1875 into a wealthy family in Titel, Serbia.

Lexically, the two sentences are not similar because both have different names, cities and dates. For a segmentation approach that solely relies upon a lexical representation of text, the two sentences are not similar or even related to each other. Even for an approach that uses a learning model to learn text representation, if it has not seen the entities mentioned in the sentences together in a training set it will be difficult for it to infer the relation between the two sentences. In fact, Mileva Marić is Einstein's ex-wife, they both worked in physics and they had three children. Hence, an ideal approach to reveal such information about the two sentences, and to measure their relatedness, would use the explicit semantic representation of text based on a knowledge base. Such a knowledge base should be based on human cognition and be intuitive to use and reason over, with no limits on domain coverage or conceptual granularity. Creating and maintaining such knowledge base requires enormous effort on the part of many people. Luckily, such a collection already exists in the form of

Wikipedia, which is one of the largest knowledge repositories on the Web. Hence, relying on such human-organised intensive knowledge reveals more meaning of the text that we want to segment regardless of the approach (linear or hierarchical) or the algorithm that we use for segmentation.

In this paper we propose *C-HTS*, a Concept-based Hierarchical Text Segmentation approach that uses semantic representation to measure the relatedness between text blocks and then builds a tree-like hierarchy of the document to reveal its semantic structure. *C-HTS* capitalises on the human knowledge encoded in Wikipedia and uses its concepts to leverage information about text that cannot be deduced solely from the input texts being processed. We evaluate *C-HTS* on two hierarchical datasets and we compare its performance against selected state-of-the-art approaches. We also use different snapshots from Wikipedia to assess the influences of knowledge base size on the hierarchical segmentation task.

The contributions of this paper are threefold. First, we propose a new approach to hierarchical text segmentation that uses explicit semantic representation of text as a substitute for traditional lexical representation. Second, we assess the impact of knowledge base size on the segmentation task through an experiment where we use different snapshots of Wikipedia from different years. Third, we have processed a recent Wikipedia snapshot (April 2017) as described in (Gabrilovich & Markovitch 2009) to use as the underlying concept space for the explicit semantic analysis of text. This processed Wikipedia snapshot[1]s along with a Java implementation of *C-HTS*[2] are publicly available.

2. Related Work

Natural Language Processing (NLP) has different tasks (Lawless et al. 2015; Bayomi et al. 2016; Naili et al. 2016). One of these tasks is Text Segmentation that is considered an essential task for other NLP tasks (Boguraev & Neff 2000). Text Segmentation can be classified into two main broad classes: Linear and Hierarchical text segmentation. Linear text segmentation approaches focus on segmenting text into coherent segments where each segment represents a specific topic (Choi 2000). An early linear text segmentation algorithm was the TextTiling approach introduced by Hearst (1997). TextTiling applied linear text segmentation by measuring the lexical similarity between text blocks. Text blocks are the smallest units that constitute the text. They range from one sentence (Ye et al. 2008) to multiple sentences (paragraphs) (Bayomi et al. 2015). TextTiling is a content-based text segmentation algorithm that uses a sliding window to segment a text. The calculation is accomplished using two vectors containing the number of terms occurring in each block. Utiyama and Isahara (2001) proposed a linear approach, U00, that is based on language models, where they use dynamic programming and the probability distribution of words to rank and select the best segments. Eisenstein and Barzilay (2008) proposed a Bayesian approach to unsupervised topic segmentation. They showed that lexical cohesion between text segments can be placed in a Bayesian context by modelling the words in each topic segment. Galley et al. (2003) proposed *LcSeg*, a *TextTiling*-based algorithm that uses *tf-idf* term weights, which improved the text segmentation results. Another well-known linear text segmentation algorithm is *C99*, introduced by Choi (2000). *C99* segments a text by combining a rank matrix, transformed from the sentence-similarity matrix, and divisive clustering.

Hierarchical text segmentation, on the other hand, focuses on discovering more fine grained subtopic structures in texts (Kazantseva & Szpakowicz 2014). An early hierarchical text segmentation approach was proposed by Yaari (1997). Yaari used paragraphs as the elementary units for his algorithm and he measured the cohesion between them using cosine similarity. An agglomerative clustering approach is then applied to induce a dendrogram over paragraphs. Eisenstein (2009) proposed a hierarchical Bayesian algorithm based on Latent Dirichlet Allocation (LDA). Eisenstein modelled each word token as a draw from a pyramid of latent topic models to create topical trees.

Both, hierarchical and linear approaches attempt to place boundaries between utterances. There are three main approaches to detect boundaries within text (Prince & Labadié 2007):

1) *Similarity-based methods*: where text blocks are represented as vectors of their terms and then a measure is used to find the proximity by using (most of the time) the cosine of the angle between these vectors. For example, C99 (Choi 2000) uses a similarity matrix to generate a local classification of sentences and isolate topical segments.

2) *Graphical methods*: a representation of term frequencies is plotted on a graph to identify topical segments (which are dense dot clouds on the graph). The DotPlotting algorithm (Reynar 1994) is the most common example of the use of a graphical approach of text segmentation.

3) *Lexical chain-based methods*: the notion behind lexical chains is to chain semantically related words together via a thesaurus. It was proposed by Morris and Hirst (1991). Multiple occurrences of a term in a document are linked together through a chain. This chain is considered broken when there are too many sentences between two occurrences of a term. Segmenter (Kan et al. 1998) is an example of approaches that use lexical chains in text segmentation. It uses lexical chains with a subtle adjustment as it determines the number of necessary sentences to break a chain in function of the syntactical category of the term.

All these approaches rely upon the traditional bag-of-words representation of text. However, a representation based solely on the endogenous knowledge in the documents themselves does not reveal much about the meaning of the text. Hence, some approaches started to enrich the text representation by exploiting its semantic meaning. Choi et al. (2001) enriched their approach, C99, by using the Latent Semantic Analysis (LSA). They applied latent concept modelling to the similarity metric. They proved that using

[1] Wikipedia 2017 snapshot processed for ESA: https://goo.gl/JZhEvm,

[2] https://github.com/bayomim/C-HTS

LSA improved the quality of their segmenter. However, these LSA-based approaches require a very large corpus, and consequently the pre-processing effort required is significant.

Some other approaches started, on the other hand, to use external resources to enrich text. Stokes et al. (2004) proposed a new approach, SeLeCT, that uses the WordNet thesaurus as an external lexical resource to add semantic links between words to create lexical chains from these links with respect to a set of chain membership rules. However, the use of such lexical resources offers little information about the different word representations. Furthermore, such resources cover only a small fragment of the language lexicon.

Ontologies have been widely used in different tasks to give a conceptual representation of entities (Bayomi & Lawless 2016). Recently, some approaches have emerged that segment text by exploiting the conceptual representation of its constituent terms. For example, we proposed OntoSeg (Bayomi et al. 2015) as a hierarchical text segmentation approach that is based on the ontological similarity between text blocks. The approach annotates text using a named entity recognition algorithm and text entities are extracted. The extracted entities are then mapped to their concepts (classes) from an ontology (DBpedia in the experiments). The sentences in the text are then represented as vectors of concepts from the ontology. The similarity between two text blocks (one or more sentences) is measured based on the similarity between the concepts of their entities in the ontology using the *is-a* relation. Naili et al. (2016) integrated a domain ontology in the topic segmentation in order to add external semantic knowledge to the segmentation process. They proposed two topic segmenters called TSS-Ont and TSB-Ont based on C99 and TextTiling respectively. They used the same techniques as C99 and TextTiling but replaced lexical similarity with concept similarity.

Although these approaches relied on an external resource and used an ontology to add a semantic layer to the segmentation process, they suffer from some drawbacks, such as: they solely extract named entities from text, and in a text with few entities or with poor performance from the named entity extraction algorithm, measuring the similarity between text blocks is not feasible. Furthermore, these approaches measure the semantic similarity between entities rather than the semantic relatedness. As argued by Budanitsky and Hirst (2006), relatedness is more general than similarity. Furthermore, dissimilar entities may also be semantically related by other relationships such as meronymy, antonymy, functional relationship or frequent association.

In this paper we propose C-HTS, a hierarchical model of text segmentation that uses the semantic relatedness between text blocks to produce a tree-like structure of a text document. C-HTS uses the explicit semantic representation of text to measure how text blocks are semantically related based on concepts from a knowledge base. C-HTS uses the exogenous knowledge (externally supplied), rather than the endogenous knowledge extracted from the documents themselves. The approach uses Wikipedia as an external knowledge base to enrich the text representation in a very high-dimensional space of concepts.

The purpose of measuring semantic relatedness is to allow computers to reason about text. Various approaches have been proposed in the literature to measure the semantic relatedness between terms using an external knowledge source. Explicit Semantic Analysis (ESA) (Gabrilovich & Markovitch 2007) is a technique that provides a semantic representation of text in a space of concepts derived from Wikipedia. ESA defines concepts from Wikipedia articles e.g., BARACK OBAMA and ACOMPUTER SCIENCE. A target term is essentially represented as a vector of concepts in Wikipedia based on how this term is mentioned in the concept's article. Relatedness is then calculated as the cosine similarity between the two vectors of the target terms. Another approach that uses the link structure of Wikipedia to measure semantic relatedness is the Wikipedia Link-based Measure (WLM) (Witten & Milne 2008). WLM measures the relatedness between two terms using the links found within their corresponding Wikipedia articles rather than using the articles' textual content. In this research we use Explicit Semantic Analysis (ESA) and we use Wikipedia as its source of knowledge. ESA has been widely used in a variety of tasks such as concept-based information retrieval (Egozi et al. 2011) and text classification (Chang et al. 2008) among other tasks. The efficacy of ESA has been proven compared to other approaches that do not rely on explicit knowledge bases.

3. Semantic Relatedness Using Wikipedia

The core idea of our algorithm is the use of an external knowledge base to enrich text representation to measure the semantic relatedness between terms, and thus sentences, and to utilise this in hierarchical text segmentation. The notion behind using explicit semantic relatedness is that it relies on a knowledge base that is built and continuously maintained by humans. The knowledge base that we use in this research is Wikipedia, the largest and fastest growing encyclopaedia in existence. This knowledge base is considered a collaborative effort that combines the knowledge of hundreds of thousands of people. Many approaches have exploited Wikipedia to measure the semantic relatedness between terms. In this research, we use Explicit Semantic Analysis (ESA) (Gabrilovich & Markovitch 2007) for this task. ESA is a method that represents meaning in a high-dimensional space of concepts, automatically driven from human-built knowledge repositories such as Wikipedia.

ESA maps a term to a concept vector, this vector contains the term's association strengths to concepts in Wikipedia. A concept is a Wikipedia article (e.g. ALBERT EINSTEIN). This concept is represented as a vector of terms which occur in that article weighted by their *tf-idf* score. After generating terms from the concept article, an inverted index is created that maps each term to a list of concepts in which this term appears. The name, Explicit Semantic Analysis, stems from the way vectors are comprised of concepts that are manually defined, as opposed to the mathematically derived contexts used by Latent Semantic Analysis.

Each input term in a text processing task (e.g. segmentation) can be represented as a vector of concepts that the term is associated with, accompanied by the degree of association between the term and each concept. The semantic relatedness between two given terms is measured by computing the cosine similarity between the concept vectors of the two terms. For larger text fragments

(sentence or paragraph), a concept vector is retrieved for each term in the fragment, then the semantic relatedness between two text fragments is measured by computing the cosine similarity between the centroid of the vectors representing the two fragments.

To elaborate on the notion of the semantic relatedness using ESA, consider the two sentences in the example mentioned in Section 1. Each term in each sentence is mapped to a vector of concepts from the vector space. Each sentence is then represented as the centroid of the vectors of the sentence's terms. For the first sentence, the centroid of the vectors contains the following concepts (among other concepts):

- ALBERT EINSTEIN AWARD
- THE EVOLUTION OF PHYSICS
- HANS ALBERT EINSTEIN → (*second child and first son of Albert Einstein and Mileva Marić*)
- ELSA EINSTEIN → (*the second wife of Einstein*)

And the centroid of the vectors of the second sentence contains the following (among other concepts):

- MILEVA MARIĆ
- HANS ALBERT EINSTEIN
- ELSA EINSTEIN
- EINSTEIN FAMILY

From these vectors, we can see that the two sentences have concepts in common. This shows that although the two sentences are not lexically similar, they are semantically related to each other.

4. The C-HTS Algorithm

4.1 C-HTS Components

The *C-HTS* algorithm proposed by this research consists of three phases:

4.1.1 Morphological Analysis

In this phase, the target is processed to be split it into sentences and remove stopwords as they are generally assumed to be of less, or no, informational value. The remaining words are then stemmed and converted to their root. In this research we use the Porter stemmer (Porter, 1980). This morphological analysis technique has been used in processing the Wikipedia terms and concepts while building the concept space from Wikipedia. The remaining terms are then used as input for the next phase.

4.1.2 Calculating the Semantic Relatedness

The key idea in *C-HTS* consists of treating the segmentation of text as an examination of the semantic relatedness between text blocks rather than traditional lexical similarity. A text block is the elementary unit of the segmentation algorithm, which is one sentence in *C-HTS*. For each sentence, and for each term in that sentence, the term is mapped to a vector of concepts from the concept space that was created from Wikipedia. The semantic relatedness between two (adjacent) sentences is calculated as the cosine similarity between the centroid of the vectors representing the individual terms in each sentence.

4.1.3 Hierarchical Agglomerative Clustering

C-HTS is an iterative approach that uses the bottom-up Hierarchical Agglomerative Clustering (HAC) algorithm for text segmentation. Hierarchical clustering algorithms have been studied extensively in the clustering literature (Jain and Dubes 1988). A typical agglomerative clustering algorithm successively merges documents into clusters based on a specific criterion such as their similarity with one another. In *C-HTS* we transfer the agglomerative clustering technique from document level to text level. The clustering process is done in *C-HTS* between text blocks within one document as opposed to across documents. The main topic for research in the HAC algorithm is the proximity test. In *C-HTS*, we apply the semantic relatedness between text blocks as the proximity test.

When applying hierarchical agglomerative clustering on text blocks the algorithm successively agglomerates blocks that are deemed to be semantically related to each other, thus forming a text structure. *C-HTS* uses HAC because it is a bottom-up clustering approach. The idea behind using a bottom-up approach in text segmentation is that it starts from the smallest clusters, that are considered the seeds of the text, and then builds the text structure by successively merging the semantically coherent clusters. This way of building the document structure can be regarded as a hierarchically coherent tree that is useful to support a variety of search methods as it provides different levels of granularity for the underlying content.

Conceptually, applying the HAC algorithm on text blocks produces a hierarchy tree or a dendrogram. In this tree, the leaf nodes correspond to individual blocks (sentences). When two blocks are merged together, a new node is created in this tree corresponding to this larger merged group. Figure 1 shows the resulting dendrogram from *C-HTS* for a sample text. In the dendrogram depicted in Figure 1, we can see that for each iteration of *C-HTS* a new level (horizontal dotted lines) is constructed from the agglomeration process on the previous level. Each level is considered a different representation of the document granularity. The level of granularity increases as we move from the root to the bottom of the tree (the leaves). For example, in level 5 in the dendrogram, we can see that the document at that level of granularity can be segmented into two segments with boundaries 19 & 25.

4.2 Complexity analysis

Hierarchical agglomerative clustering algorithms for clustering text documents in general takes an order of $O(N^2)$ steps. This is because at each stage in the algorithm the proximity of the newly merged object to all other available segments is computed. On the other hand, in *C-HTS*, we apply the hierarchical agglomerative clustering on text level. Since we need to preserve the linear order in text, we only compute the proximity between a cluster and its surrounding clusters. Hence, *C-HTS* takes an order of $O(N)$ steps on text level.

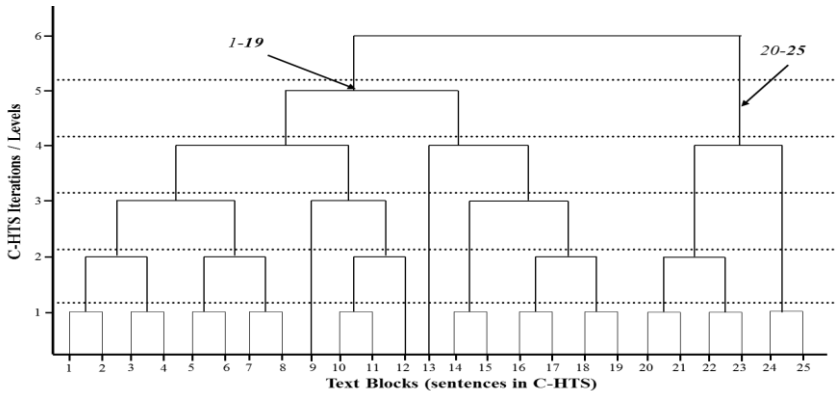


Figure 1 C-HTS output as a dendrogram of a sample text

4.3 Word Sense Disambiguation

Word Sense Disambiguation (WSD) is the task of identifying the meaning of a term, when the term has multiple meanings, based upon the context of where it appears (Navigli 2009). For example, "light" can mean "not heavy" or "illumination", what identifies its meaning is the context of where "light" is used. For a natural language processing task like text segmentation, disambiguating such words would allow the task to better understand the meaning of the sentence and to reason about it and thus enhance the quality of the segmentation. For lexical segmenters, being inherently limited to lexical representation of text, these approaches require an extra level of sophistication to disambiguate words.

In *C-HTS*, we measure the relatedness between sentences as the cosine similarity between the centroid of the vectors representing the two sentences. This interpretation of text is considered an implicit disambiguation of terms. For example, a sentence that has the term "*Apple*" amongst other computer related terms, taking the centroid of the vectors will boost the computer-related concepts and will disambiguate the term effectively. To illustrate how words are disambiguated in *C-HTS*, consider the following sentence: "*I love fruit, particularly a nice apple*". In this sentence, after applying morphological analysis (Section 4.1.1), the remaining prominent terms are *love*, *fruit, particularli*, *nice* and *appl*. Among these terms, the word "apple" has different interpretations. From the underlying concept space that we have created from Wikipedia, the top concepts generated for the word *apple* are:

- APPLE DAY → (*related to apple fruit*)
- APPLE SPECIALIST → (*related to apple Inc.*)
- APPLE EXTENDED KEYBOARD → (*related to apple Inc.*)
- EMPIRE (APPLE) → (*related to apple fruit*)
- APPLE STORE (ONLINE) → (*related to apple Inc.*)

These concepts are mostly talking about the company, Apple Inc. When considering the centroid of the vectors representing this sentence, the top generated concepts are (among others):

- FRUIT PICKING
- ROME APPLE → (*a kind of apple originating near Rome Township, Ohio*)
- LIST OF APPLE CULTIVARS
- EMPIRE (APPLE) → (*a kind of apple derived from a seed grown in 1945*)

From these concepts, we can see that they all are related to the fruit apple. This proves that considering the centroid of the vectors of a sentence disambiguates the terms without adding extra sophisticated text processing layers. This vector can also be seen as a representation of the context of that sentence. This in turn enhances the understandability of text and enhances the segmentation quality.

5. C-HTS Evaluation

Research on hierarchical text segmentation has been scarce and most state-of-the-art approaches evaluated their hierarchical approaches on linear segmentation datasets. For example, Yaari (1997) evaluated his approach on the *Stargazers* article. He compared his approach against a linear text segmentation approach, TextTiling. OntoSeg (Bayomi et al. 2015) was evaluated using Choi's dataset for linear text segmentation evaluation. Evaluating a hierarchical text segmentation algorithm using a linear dataset does not give a realistic picture of the performance of the hierarchical algorithm. This is because the output of a hierarchical algorithm is a tree structure, while a linear dataset has consequently segmented chunks of text. Hence, selecting an appropriate dataset is a critical step in the evaluation process.

5.1 Datasets

In this research, we argue that *C-HTS* is applying the hierarchical text segmentation as if a human would perform the task. To prove this assumption, a gold standard dataset that is created by humans is needed. Furthermore, the dataset needs to be suitable for a hierarchical text segmentation task. Luckily, Kazantseva and Szpakowicz (2014) proposed two datasets that are suitable for evaluating hierarchical text segmentation and both were annotated by humans. The authors evaluated their approach, Hierarchical Affinity Propagation for Segmentation (HAPS), against two well-defined datasets: the *Moonstone* dataset and the Wikipedia dataset compiled by Carroll (2010).

Moonstone dataset: This dataset consists of nine chapters of the Moonstone novel. Kazantseva and Szpakowicz (2014) employed human annotators to annotate the dataset and to identify the hierarchical structure of each text document. The annotators were asked to read a chapter and split it into top-level segments according to where they can

see a shift in topic. Each chapter was annotated by 3-6 people (4.8 on average)[3].

Wikipedia dataset: This dataset was compiled by Carroll (2010). The dataset consists of 66 Wikipedia articles on various topics. The html pages were converted to flat text, and unneeded content such as navigation boxes, and image captions were removed. The hierarchical structure for each article is created automatically from the structure of the Wikipedia page, i.e. heading text was replaced with a boundary marker, indicating the heading depth. This depth represents the level in the text's hierarchical structure. While the levels in the Wikipedia dataset were created automatically, the original structure of the documents is created by the human authors who contribute to Wikipedia. Thus it is considered a human annotated dataset.

Since *C-HTS* is based on the external knowledge base to enrich text representation, evaluating it on these two datasets will give us a realistic picture of the performance of *C-HTS* as a concept based approach. This is due to the inherent human involvement in the construction process of the two datasets.

5.2 Baselines

To evaluate the quality of segmentations produced by *C-HTS*, there is a need to compare its performance against hierarchical text segmentation approaches. Work on hierarchical text segmentation has been scarce. To the best of our knowledge, the only publicly available hierarchical segmenter (along with a dataset) is HAPS that was proposed by Kazantseva and Szpakowicz (2014). HAPS[4] is a hierarchical text segmentation approach that is based on a graphical model for hierarchical clustering called Hierarchical Affinity Propagation (Givoni et al. 2011). The input for HAPS is a matrix of similarity between text blocks. HAPS requires the desired number of levels to be in the produced topical tree and a preference value for each data point and each level. HAPS also finds a centre for each segment at every level of the produced topical tree, a data point which best describes the segment.

HAPS was compared against two linear segmenters *MCSeg* (Minimum Cut Segmenter) (Malioutov & Barzilay 2006) and *BSeg* (Bayesian based Segmenter) (Eisenstein 2009). These two systems were chosen because they are representative of the existing text segmentation methods, and their implementations are freely available on the internet. *MCSeg* casts text segmentation in a graph-theoretic framework. In this approach, text is abstracted into a weighted undirected graph, where the nodes of the graph correspond to text blocks and edge weights represent the pairwise block similarity. Text segmentation in *MCSeg* corresponds to a graph partitioning that optimises the normalised-cut criterion. In *BSeg*, the lexical cohesion between segments is placed in a Bayesian context. The words are modelled in each topic segment as draws from a multinomial language model associated with the segment.

To obtain hierarchical segmentation from these two linear segmentation systems, both systems were run first to produce top-level segmentations. Each segment thus computed was a new input document for segmentation. The procedure was repeated twice to obtain a three level structure of the text.

In this research we compare our approach, *C-HTS*, with *HAPS* and the other two baselines proposed by Kazantseva and Szpakowicz. For evaluation consistency, we use their experimental settings by evaluating the top three levels (excluding the root) of the document structure produced by *C-HTS*.

5.3 Evaluation Metric

We evaluate *C-HTS* using the well-known metric *windowDiff* (Pevzner & Hearst 2002). *windowDiff* is a penalty measurement metric, which means that lower scores indicate higher segmentation accuracy. *windowDiff* was proposed by Pevzner and Hearst as a modification to the *Pk* evaluation metric proposed by Beeferman et al. (1997). *windowDiff* is computed by sliding a window across the input sequence and at each step examining whether the hypothesised segmentation is correct about the separation (or not) of the two ends of the window. It counts the difference of the number of segment boundaries in the given window between the two partitions. *windowDiff* is defined as:

$$windowDiff\ (ref\ , hyp) = \frac{1}{K-k} \sum_{i=1}^{K-k} (|b(ref_i, ref_{i+k}) - b(hyp_i, hyp_{i+k})| > 0)$$

where *ref* is the correct segmentation for reference, *hyp* is the segmentation produced by the model, *K* is the number of sentences in the text, *k* is the size of the sliding window and *b(i, j)* is the number of boundaries between sentences *i* and *j*.

windowDiff is designed to evaluate linear text segmentation not hierarchical trees. Hence, in our evaluation, and for the sake of comparability we follow the same technique as Kazantseva and Szpakowicz (2014). Each level of the text hierarchy is treated as a separate segmentation and each hypothetical level is compared against a corresponding level in the reference segmentation.

5.4 Results

The *Moonstone* dataset has on average 4.8 annotations per chapter. To obtain a realistic picture of the results across the different annotators per file, each hypothetical segmentation is separately compared against each available gold standard. After that, the averages across all annotators are taken as the final score. For the two datasets, Table 1 shows results of the comparison between *C-HTS* and the other three baselines using the *windowDiff* evaluation metric. Since *C-HTS* and *HAPS* are inherent hierarchical text segmentation approaches, both were run without knowing the number of segments. *BSeg* was able to run with and without that parameter. In our results, we report the *BSeg* run without this parameter. *MCSeg*, on the other hand, required that the exact number of segments to be specified. This makes it considerably more informed than others.

The results show that *C-HTS* performs well on both datasets compared to the baselines, even when compared to more informed baseline. For the Wikipedia dataset, *C-HTS* performs better than the baselines on all three levels.

[3] For more details about the *Moonstone* dataset, the reader is referred to (Kazantseva and Szpakowicz, 2014).

[4] HAPS implementation and the Moonstone dataset are available here: https://github.com/anna-ka/HAPS

This proves that using the explicit semantic representation of text gives more understanding of the meaning of the text, and thus enhances the process of hierarchical text segmentation.

	Level	*Moonstone windowDiff*	*Wikipedia windowDiff*
C-HTS	3 (top)	**0.320**	**0.360**
	2 (middle)	0.507	**0.400**
	1 (bottom)	**0.488**	**0.409**
HAPS	3 (top)	0.337	0.421
	2 (middle)	**0.422**	0.447
	1 (bottom)	0.556	0.617
MCSeg	3 (top)	0.375	0.440
	2 (middle)	0.541	0.424
	1 (bottom)	0.601	0.471
BSeg	3 (top)	0.600	0.637
	2 (middle)	0.447	0.877
	1 (bottom)	0.545	0.952

Table 1. Evaluation of *C-HTS*, HAPS and iterative versions of MCseg and BSeg using *windowdiff* per level

For the Moonstone dataset, *C-HTS* performs favourably on the top and bottom levels but we notice that its performance on the middle level is not better than HAPS and BSeg. We argue that this is because in the Moonstone dataset the boundary for each level, in each document, was placed by a number of different annotators, hence, there can be mixed agreement between those annotators on the correct placement of the level boundary. On the other hand, in Wikipedia dataset, the original article hierarchy (where levels are obtained from) was created and updated with the agreement of the Wikipedia article contributors.

6. Discussion

6.1 Elementary units for the segmenter

The bottom-up hierarchical text segmentation algorithms start with atomic text pieces as their elementary units and then successively grow areas of coherence at the most appropriate place. The elementary units could be of a fixed size, such as a specific number of sentences, or could be of a mutable size such as paragraphs. For example, Yaari (1997) and Kazantseva & Szpakowicz (2014) used paragraphs as the elementary units for their segmenters, while in *C-HTS* we use one sentence as the elementary unit.

The size of the elementary units is an influential parameter for the segmentation algorithm and it has implications on the segmentation accuracy. Previously, we experimented with the influence of different elementary unit sizes on the hierarchical segmentation task (Bayomi et al. 2015). We experimented with sizes ranging from one to four sentences per unit. The best run we reported in our experiments was when we used one sentence as the elementary unit. The results also concluded that the higher the size of the elementary unit, the lower the accuracy of the segmentation.

This also adds to the understanding of the inconsistency of *C-HTS* performance on the Moonstone dataset. Besides the disagreement between the human annotators about the correct placement of level boundary, the elementary units presented to the annotators, to build the gold standard, were paragraphs. As a result, and for the evaluation consistency, we had to set the elementary units for *C-HITS* to be paragraphs which impacted the performance of the algorithm. This can be seen in the results of the first experiment (and the following experiment) where the performance of *C-HTS* on the Wikipedia dataset, where we use one sentence as the elementary unit, gives, on average, lower error rates than its performance on the Moonstone dataset.

6.2 Text Granularity

Hierarchical text segmentation approaches produce a structural representation of text that represents different levels of granularity. In *HAPS*, the desired number of levels needs to be passed as a parameter to the algorithm. In contrast, in *C-HTS*, it does not need to know number of levels that are needed in the output structure because the structure produced by *C-HTS* depends on the coherence between the atomic units of the text. This way of building the structure makes the output more granular and facilitates its use in different tasks like Information Retrieval. Identifying the number of levels of the output limits the usage of the produced hierarchy, as each task requires a different level of granularity. Hence, from this point of view, *HAPS* is considered a task-dependent approach, as its parameters need to be set depending on the task in question. On the other hand, *C-HTS* is considered a task-independent approach as it produces all the available levels of granularity in the processed document, hence it can be used with different tasks.

6.3 Multilingual C-HTS

C-HTS is based on the concept space built from Wikipedia. Wikipedia is the largest encyclopaedia in existence that is available in dozens of languages. Building a concept space for these languages would help an ESA-based task to be used with texts in different languages. Gurevych et al. (2007) applied ESA to the German-language Wikipedia and used it for semantic relatedness and information retrieval tasks. Their experiments showed that using ESA was superior compared to a system based on the German version of WordNet (GermaNet).

The core of *C-HTS* is the process of measuring the semantic relatedness between clusters using the explicit semantic interpretation of text. This process is essentially based on the underlying concept space that we have built from Wikipedia. Moving *C-HTS* from one language to another can be done easily. Changing the language of the underlying concept space would make no difference in the running process of *C-HTS*. The only step which must be changed is the morphological analysis to filter out the prominent terms in text. This step is relatively easy to implement as there has been a large volume of work completed on morphological analysis for languages other than English (Rafferty & Manning 2008). Hence, *C-HTS* can be seen as a multilingual hierarchical text segmentation approach that can semantically represent text and reason about it regardless the language of the text.

7. The Impact of Knowledge Breadth

In this research, we use a concept space that is built from the text of a knowledge base articles (Wikipedia). Anderka and Stein (2009) showed that the nature of the text collection used to build the concept space has much less impact on the explicit semantic analysis performance than its size.

Wikipedia is being constantly expanded and updated by different contributors who add new articles and extend the existing ones. Consequently, the amount of knowledge in Wikipedia is expanding. We conjecture that such expansion, and the growth of information available in the knowledge base should impact the accuracy of the segmentation process. To test this assumption, we acquired different snapshots of the entire Wikipedia knowledge base from three different years: 2006, 2013 and 2017. The snapshots from 2006 and 2013 were processed by Carvalho et al. (2014) and ready for use. For the 2017 snapshot, we processed it ourselves following the instructions in (Gabrilovich and Markovitch, 2009) and (Carvalho et al., 2014)[5].

7.1 Experiment and Results

Table 2 presents a comparison of the amount of information contained in the three used Wikipedia snapshots. In this experiment, we ran our approach on the two aforementioned datasets but using different concept spaces built from the three different Wikipedia snapshots. The purpose of this experiment is to examine the effect using different versions of the underlying knowledge base has on *C-HTS*.

Table 3 shows the results of the experiment. As we can see, increasing the amount of knowledge in the knowledge base leads, on average, to improvements in hierarchical text segmentation. Although the difference in performance of the three versions is admittedly small, it is consistent across the datasets.

8. Conclusion and Future Work

In this paper, we proposed *C-HTS*, a new Concept-based Hierarchical Text Segmentation approach. The core idea of *C-HTS* is the use of external knowledge to enhance the text representation by adding a semantic layer of concepts that represents the text in a high dimensional semantic space. Relatedness between the atomic units of text is measured using this semantic representation. A Hierarchical Agglomerative Clustering (HAC) algorithm is then used to grow areas of coherent segments. The output of *C-HTS* is a tree-like structure of the input text. We compared *C-HTS* against the state-of-the-art approaches across two different datasets. The results showed that *C-HTS* performed favourably against other approaches.

We also evaluated the influence of the size of the knowledge base that *C-HTS* uses to reason about text. Since *C-HTS* uses Wikipedia as the underlying knowledge base, we measured its performance when using different snapshots of Wikipedia over different years: 2006, 2013 and 2017. The results show that there is a measurable impact upon segmentation performance, and while the difference is small, it is consistent across the two datasets. We also processed a recent Wikipedia snapshot (April 2017) to create a concept space. This processed Wikipedia snapshot along with the implementation of *C-HTS* are publicly available.

Moving forward, and in light of our results, viable future work may involve experimenting *C-HTS* with other familiar languages that have a rich representation in Wikipedia such as French and German.

As text segmentation is widely used as a pre-processing task for Information Retrieval, we plan to use C-HTS with a concept-based retrieval task for content adaptation (Bayomi 2015). The hierarchical structure produced by C-HTS is generated from the use of a concept space that generates new text features automatically. Indexing documents based on their conceptual representation along with these features can be exploited to make the retrieval process more focused.

	2006 Snapshot	*2013 Snapshot*	*2017 Snapshot*
Number of articles	895,000	4,133,000	5,373,241
Concepts used	369,767	1,270,521	1,446,243
Distinct terms	598,391	1,615,525	1,825,353
Concept space size after processing	11 Gb	21 Gb	12.5 Gb[6]

Table 2 Comparison of the three Wikipedia snapshots

	Level	*Moonstone windowDiff*	*Wikipedia windowDiff*
2006 Snapshot	3 (top)	0.347	0.365
	2 (middle)	0.545	0.404
	1 (bottom)	0.504	0.411
	Average	0.465	0.3933
2013 Snapshot	3 (top)	0.346	0.366
	2 (middle)	0.539	0.399
	1 (bottom)	0.509	0.405
	Average	0.464	0.390
2017 Snapshot	3 (top)	0.320	0.360
	2 (middle)	0.507	0.400
	1 (bottom)	0.488	0.409
	Average	**0.438**	**0.3897**

Table 3 *windowDiff* Evaluation of *C-HTS* using different versions of the underlying knowledge source (Wikipedia)

9. Acknowledgements.

The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is cofunded under the European Regional Development Fund.

10. Bibliographical References

Anderka, M. & Stein, B., 2009. The ESA Retrieval Model Revisited. In Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '09. New York, NY, USA: ACM, pp. 670–671.

Bayomi, M., 2015. A Framework to Provide Customized Reuse of Open Corpus Content for Adaptive Systems. In Proceedings of the 26th ACM Conference on Hypertext & Social Media. HT '15. New York, NY, USA: ACM, pp. 315–318.

[5] The technical instructions and snapshots can be found here: http://treo.deri.ie/easyesa/

[6] We indexed the 2017 snapshot in MongoDB v3 that uses the WiredTiger storage engine which applies more compression than the old *mmapv1* engine in MongoDB version used in indexing both 2006 and 2013 snapshots.

Bayomi, M. et al., 2015. OntoSeg: A Novel Approach to Text Segmentation Using Ontological Similarity. In 2015 IEEE International Conference on Data Mining Workshop (ICDMW). pp. 1274–1283.

Bayomi, M. et al., 2016. Towards Evaluating the Impact of Anaphora Resolution on Text Summarisation from a Human Perspective. In the 21st International Conference on Applications of Natural Language to Information Systems, NLDB 2016, Salford, UK, June 22-24, 2016, pp. 187–199.

Bayomi, M. & Lawless, S., 2016. ADAPT_TCD: An Ontology-Based Context Aware Approach for Contextual Suggestion. In The Twenty-Fifth Text REtrieval Conference Proceedings (TREC 2016), Contextual Suggestion Track. Gaithersburg, Maryland, USA: National Institute for Standards and Technology, NIST Special Publication pp500-321.

Beeferman, D., Berger, A. & Lafferty, J., 1997. Text Segmentation Using Exponential Models. In Proceedings of the Second Conference on Empirical Methods in Natural Language Processing. pp. 35–46.

Boguraev, B.K. & Neff, M.S., 2000. Discourse segmentation in aid of document summarization. Proceedings of the 33rd Annual Hawaii International Conference on System Sciences.

Davis, Randall, and Douglas B. Lenat. Knowledge-Based Systems in Artificial Intelligence: 2 Case Studies. McGraw-Hill, Inc., 1982.

Budanitsky, A. & Hirst, G., 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. Computational Linguistics 32.1 (2006): pp13-47.

Carroll, L., 2010. Evaluating Hierarchical Discourse Segmentation. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL Conference. NAACL 2010. pp. 993–1001.

Carvalho, D. et al., 2014. EasyESA: A Low-effort Infrastructure for Explicit Semantic Analysis. In Proceedings of the 2014 International Semantic Web Conference (Posters & Demos), pp. 177-180.

Chang, M.-W. et al., 2008. Importance of Semantic Representation Dataless Classification. In AAAI 2008 vol. 2, pp. 830–835.

Choi, F.Y.Y., 2000. Advances in Domain Independent Linear Text Segmentation. In Proceedings of the 1st North American Chapter of the ACL Conference. NAACL 2000. pp. 26–33.

Choi, F.Y.Y., Wiemer-Hastings, P. & Moore, J., 2001. Latent Semantic Analysis for Text Segmentation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP 2001. pp. 109–117.

Egozi, O., Markovitch, S. & Gabrilovich, E., 2011. Concept-Based Information Retrieval Using Explicit Semantic Analysis. ACM Transactions of Information Systems, p.8:1--8:34.

Eisenstein, J., 2009. Hierarchical Text Segmentation from Multi-Scale Lexical Cohesion. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL Conference, NAACL 2009 , pp.353–361.

Eisenstein, J. & Barzilay, R., 2008. Bayesian Unsupervised Topic Segmentation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP '08., pp. 334–343.

Gabrilovich, E. & Markovitch, S., 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In IJcAI 2007. Vol. 7, pp. 1606–1611.

Gabrilovich, E. & Markovitch, S., 2009. Wikipedia-based semantic interpretation for natural language processing. Journal of Artificial Intelligence Research, pp.443–498.

Galley, M. et al., 2003. Discourse Segmentation of Multi-party Conversation. In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - ACL 2003, pp. 562–569.

Givoni, I.E., Chung, C. & Frey, B.J., 2011. Hierarchical Affinity Propagation. Proceedings of the Twenty-Seventh Conference on Uncertainty in AI 2011. pp. 238–246.

Gurevych, I., Müller, C. & Zesch, T., 2007. What to be?-electronic career guidance based on semantic relatedness. In Annual Meeting-Association for Computational Linguistics – ACL 2007, pp. 1032–1039.

Hearst, M.A., 1997. TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. In Journal of Computational Linguistics, 23(1), pp.33–64.

Jain, A.K. & Dubes, R.C., 1988. Algorithms for clustering data, Prentice-Hall, Inc.

Kan, M.-Y., Klavans, J.L. & McKeown, K., 1998. Linear Segmentation and Segment Significance. CoRR.

Kazantseva, A. & Szpakowicz, S., 2014. Hierarchical Topical Segmentation with Affinity Propagation. In COLING 2014. pp. 37–47.

Lawless, S., Lavin, P., Bayomi, M., Cabral, J. P., & Ghorab, M. R 2015. Text Summarization and Speech Synthesis for the Automated Generation of Personalized Audio Presentations. In International Conference on Applications of Natural Language to Information Systems 2015, pp. 307–320.

Malioutov, I. & Barzilay, R., 2006. Minimum Cut Model for Spoken Lecture Segmentation. In Proceedings of the COLING 2006. pp. 25–32.

Morris, J. & Hirst, G., 1991. Lexical Cohesion Computed by Thesaural Relations As an Indicator of the Structure of Text. Computational Linguistics, pp.21–48.

Naili, M., Habacha Chaibi, A. & Hajjami Ben Ghezala, H., 2016. Exogenous approach to improve topic segmentation. International Journal of Intelligent Computing and Cybernetics, pp.165–178.

Navigli, R., 2009. Word Sense Disambiguation: A Survey. ACM Computing Surveys (CSUR),p.10:1--10:69.

Pevzner, L. & Hearst, M.A., 2002. A Critique and Improvement of an Evaluation Metric for Text Segmentation. Computational Linguistics, pp.19–36.

Porter, M.F., 1980. An algorithm for suffix stripping. Program, 14(3), pp.130–137.

Prince, V. & Labadié, A., 2007. Text Segmentation Based on Document Understanding for Information Retrieval. Natural Language Processing and Information Systems, pp. 295–304.

Rafferty, A.N. & Manning, C.D., 2008. Parsing three German treebanks: Lexicalized and unlexicalized baselines. In Proceedings of the Workshop on Parsing German. pp. 40–46.

Reynar, J.C., 1994. An Automatic Method of Finding Topic Boundaries. In Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics. ACL '94, pp. 331–333.

Stokes, N., Carthy, J. & Smeaton, A.F., 2004. SeLeCT: a lexical cohesion based news story segmentation system. AI communications, 17(1), pp.3–12.

Tellex, S. et al., 2003. Quantitative Evaluation of Passage Retrieval Algorithms for Question Answering. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. SIGIR '03, pp. 41–47.

Utiyama, M. & Isahara, H., 2001. A Statistical Model for Domain-independent Text Segmentation. In Proceedings of the 39th Annual Meeting on Association for Computational Linguistics, pp. 499–506.

Witten, I. & Milne, D., 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, pp. 25–30.

Yaari, Y., 1997. Segmentation of Expository Texts by Hierarchical Agglomerative Clustering. In Recent Advances in NLP (RANLP'97).

Ye, N. et al., 2008. A Dynamic Programming Model for Text Segmentation Based on Min-Max Similarity. In Information Retrieval Technology - Lecture Notes in Computer Science, pp. 141–152.

Semantic Supersenses for English Possessives

Austin Blodgett, Nathan Schneider
Georgetown University
Washington, DC
{ajb341, nathan.schneider}@georgetown.edu

Abstract

We adapt an approach to annotating the semantics of adpositions to also include English possessives, showing that the supersense inventory of Schneider et al. (2017) works for the genitive **'s** clitic and possessive pronouns as well as prepositional **of**. By comprehensively annotating such possessives in an English corpus of web reviews, we demonstrate that the existing supersense categories are readily applicable to possessives. Our corpus will facilitate empirical study of the semantics of the genitive alternation and the development of semantic disambiguation systems.

Keywords: Genitive, Possessive, Supersense, Lexical Semantics

1. Introduction

Possessive constructions can be used to express a wide variety of semantic relations. English has two canonical constructions for attributive possession. The first uses the clitic **'s** or a possessive pronoun and is known as the **s-genitive** (or Saxon genitive): e.g., *the car**'s**/**its** windshield*. The second uses the preposition **of** and is known as the **of-genitive**: *the windshield **of** the car*. Both constructions are highly polysemous as to the nature of the relation between the two noun phrases. A few of the semantic relations associated with possessives include:

- Alienable possession: *John**'s** computer*.
- Kinship: ***My** sister was surprisingly late.*
- Part–whole relations: *The car**'s** windshield*.
- Thematic roles: *The boy**'s** murder was never reported.* (This is actually ambiguous: The role that the boy fills with respect to the predicate *murder* could be either agent or patient, depending on whether he was the victim or perpetrator.)

Previous work on annotating the semantics of possessive constructions has taken a sense disambiguation perspective, with semantic categories specific to relations between nominals (Badulescu and Moldovan, 2009) or s-genitives (Tratz and Hovy, 2013). In this paper, we show that a tagset for broad-coverage semantic annotation of prepositions and postpositions can be applied—mostly as is—to English possessive constructions.

We use the **adposition supersense** inventory (Schneider et al., 2015, 2016, 2017), which was designed with adpositions (including **of**) in mind, to annotate all of-genitive and s-genitive tokens in a 55,000 word corpus of English web reviews (§3). In so doing, we demonstrate that the existing supersense categories are readily applicable to English possessives.

The latest version consists of 50 general supersense categories including thematic role labels (AGENT, THEME, RECIPIENT, etc.) and relations that hold between entities (POSSESSOR, WHOLE, SOCIALREL, etc.). §4 describes the supersenses that proved useful for possessive constructions. In §5 we examine their distribution in the corpus and their relationship to categories from prior work.

Our corpus will facilitate the development of semantic disambiguation systems, as well as the empirical study of the semantics of the English genitive alternation (i.e., what factors influence the choice of **'s** vs. **of**), building on the work of Rosenbach (2002) and Shih et al. (2012). As the latest version of the supersense inventory was designed to be cross-linguistically applicable, we anticipate that this approach will eventually accommodate possessives and genitives in other languages, yielding similar analyses of supersense coverage and distribution to the analysis in this paper. It has been investigated for adpositions in English, Korean, Hebrew, and Hindi thus far (Schneider et al., 2017; Hwang et al., 2017).

2. Related work

Linguistic study has shown that possessive constructions in English and other languages can denote a number of semantic relations (Taylor, 1996; Nikiforidou, 1991; Rosenbach, 2002; Heine, 2006), in particular described as prototypical forms of possession (*legal ownership, kinship, body parts, part-whole relations*) and non-prototypical possession (other semantic relations) (Rosenbach, 2002; Wolk et al., 2013). Nikiforidou (1991) shows that possession in classical Greek can denote a comparison relation ("better than Plato"). Stefanowitsch (2003) shows that s-genitive and of-genitive are not interchangeable, and further the genitive alternation is not fully predictable from *animacy*, *givenness*, or *syntactic weight*. S-genitive and of-genitive are each polysemous but not interchangeable. We will discuss their semantic differences further in §5.

The computational linguistics community has seen a few studies of relations between nominals—including possessive constructions—and their automatic disambiguation (e.g., Badulescu and Moldovan, 2009; Tratz and Hovy, 2013). In §5.2, we compare the adposition supersense scheme to previous proposals for English possessives that were based on noun-noun relations. We explore whether the English preposition system, by contrast, is a useful departure point for characterizing the semantics of genitives.

Supersense	's+PRP$ %	of %	Supersense	's+PRP$ %	of %	Supersense	's+PRP$ %	of %
Agent	11.2	0.4	Identity		5.5	SocialRel	19.5	0.8
Beneficiary	0.9		Instrument		0.2	Species		7.7
Causer	0.4	0.4	Locus	0.1	1.6	Stimulus		2.4
Characteristic		1.4	Manner		0.2	Stuff		3.3
ComparisonRef		0.2	OrgRole	2.9	1.0	Theme	1.1	3.1
Cost		0.6	Originator	6.9	0.2	Time	0.3	0.6
Duration	0.2		Possession		0.6	Topic		6.3
Experiencer	7.7	0.4	Possessor	34.2		Whole	6.0	12.2
Extent		0.4	Quantity		36.9			
Gestalt	7.7	13.8	Recipient	1.0		**# tokens**	1104	509

Table 1: Token distribution of supersenses: **of**-PP vs. genitive case ('**s** and possessive pronouns). The percentages in each column sum to 100%. In addition, there were 11 s-genitive tokens that were used in idioms.

3. Corpus annotation

For this work, 2 linguists annotated a 55,000 word corpus of English web reviews. We annotate a total of 1,613 possessive constructions with 28 attested supersense labels (out of 50 total). Their lexical distribution is given in figure 1. One annotator labeled all genitive case tokens from scratch. The other annotator revised the **of** tokens from Schneider et al.'s (2016) supersenses to reflect the inventory of Schneider et al. (2017). From this data, 100 sentences containing s-genitives were randomly sampled and independently labeled by both annotators.[1] 109 s-genitive tokens are in this sample. In our study, each s-genitive or of-genitive token was assigned a single label. (See Hwang et al. (2017); Schneider et al. (2018) for a strategy of representing construal with two supersense labels.)

The 2 annotators agreed 72% of the time; Cohen's κ was 0.66, falling into a range which has been said to indicate "substantial" agreement (Viera et al., 2005). Disagreements are discussed in §4.2 below.

4. Supersense inventory for possessives

4.1. Categories

Of the 50 supersenses in Schneider et al.'s (2017) inventory, 28 appear to be relevant to English possessive constructions. Their distribution in our corpus appears in table 1. We exemplify and discuss these categories below.[2]

CONFIGURATION subtypes. The most prototypical possessive scenarios are stative relationships between entities. Excluding relations of place and time (which may apply to static or dynamic scenes), these fall under the CONFIGURATION portion of the supersense hierarchy. Together, these account for 83.3% of the **of** tokens and 70.3% of the s-genitive tokens in our corpus.

CHARACTERISTIC: *a place **of** beauty*
"NP_1 **of** NP_2", where NP_1 refers to an entity and NP_2 to a quality of that entity (also: *a person **of** honor*); seems to be a rare construction that does not admit an **'s** paraphrase.

COMPARISONREF: *the opposite **of** cheap*
This category exists primarily for **than**, **like**, **as**, and similar prepositions that can be used in various statements of comparison, contrast, similarity, or differentiation. Occasionally, a predicate like *opposite* uses **of** to mark such a role.

GESTALT: ***Her** flexibility and accessibility, Quality **of** work*
GESTALT is the inverse of CHARACTERISTIC: it is the holder of a property.

IDENTITY: *a neat gem **of** a restaurant*
This label is used for constructions that establish some sort of equation between the two noun phrases. With **of**, the head noun is a category being ascribed to the dependent. The above example can be paraphrased as *a restaurant that is a neat gem*.

ORGROLE: ***his** firm, a customer **of** this store*
The dependent NP is an organization/institution with which somebody (denoted by the head NP) has an association.

POSSESSION: *the owner **of** a new car*
For of-genitives and s-genitives, this is restricted to arguments of a predicate of ownership. In a way, the example above is also a THEME. With *the car **'s** owner*, the labels POSSESSION and GESTALT both seem to apply (because the owner is a piece of information typically associated with cars).

POSSESSOR: ***her** dog*
Strikingly, there were no clear **of**-POSSESSORs in our corpus—for the most prototypical forms of possession (human ownership of a nonhuman entity), English speakers exhibit a strong preference for the s-genitive, though *the dog **of** hers* is a valid paraphrase of ***her** dog*.

QUANTITY: *plenty **of** parking*
The preposition **of** is frequently used in various expressions of quantity which cannot be paraphrased with the s-genitive.

SOCIALREL: ***her** brother, a friend **of** mine*
This category covers relationships between persons, including kinship, friendship, and business or other social associations (e.g., **my** teacher). The s-genitive is far more popular than **of** for this relation in our corpus.

SPECIES: *that kind **of** behavior*

[1] Specifically: Annotator A annotated all genitive case markers in the corpus, and selected examples for discussion with Annotator B. After the discussion, A revised the original annotations and B annotated the random sample of 100 sentences. So as not to skew the results, B skipped 6 tokens that B recalled specifically from the discussion. For the tokens in the sample, A applied 9 distinct supersenses while B found use for a superset of 15 supersenses.

[2] Most examples are drawn or adapted from our corpus. Where the names of the categories are conventional, we do not provide a definition. Further definitions and guidelines on applying these categories appear in Schneider et al. (2018).

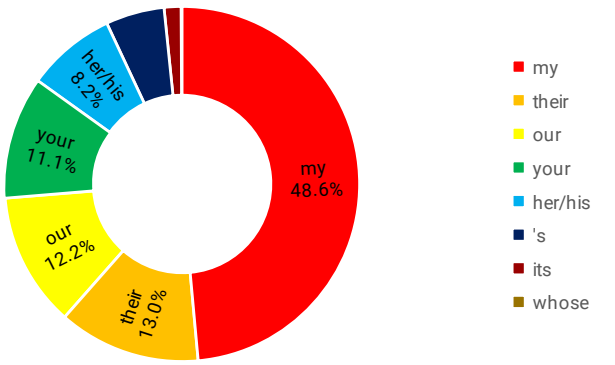


Figure 1: Distribution of s-genitive forms with supersense labels. Nonstandard spellings like **ur** have been normalized.

STUFF: *the piece **of** metal*
WHOLE: *judge a book by **its** cover, the inside **of** my car*

PARTICIPANT subtypes. Nominalization and other processes result in the realization of participant roles with possessive marking. In our data, these account for 13.9% of **of** tokens and 29.2% of s-genitives.

AGENT: ***her** help, the fault **of** the parts supplier*
BENEFICIARY: *children **'s** clothes*
CAUSER: *the fire **'s** damage, victim **of** adversity*
COST: *a tax bill **of** $6,000*
EXPERIENCER: *will exceed **your** expectations, a favorite **of** our family*
INSTRUMENT: *Usage **of** product barcodes*
ORIGINATOR: ***his** advice, in the words **of** my son*
ORIGINATOR is defined as "animate who is the initial possessor or creator/producer of something, including the speaker/communicator of information" (Schneider et al., 2017, p. 17).
RECIPIENT: ***my** delivery*
STIMULUS: *a fear **of** snakes*
THEME: ***my** next haircut, spraying **of** pesticides, a dish full **of** filler vegetables*
TOPIC: *kept me apprised **of** status*

AGENT, EXPERIENCER, and ORIGINATOR are not infrequently expressed with the s-genitive. STIMULUS, THEME, and TOPIC are not infrequently expressed with **of**. The rest are rarely observed in possessive constructions.

CIRCUMSTANCE subtypes. Rarely expressed with possessive constructions, these comprise 2.8% of **of** tokens and 0.5% of s-genitives.

DURATION: *a year **'s** worth of dirty clothes*
EXTENT: *a production increase **of** 10%*
LOCUS: *Miami **'s** best beach, I am just south **of** Walnut*
LOCUS is used for concrete as well as abstract locations, including states and values.
MANNER: *My room reeks **of** old cigarette smoke.*
TIME: *today **'s** tough times, March **of** 2010*

4.2. Further conventions

Multiword expressions and idioms. Schneider et al. (2014) had already annotated the online reviews corpus for **multiword expressions** (Baldwin and Kim, 2010), including proper names and idioms. We did not apply supersenses to possessives used within a completely fixed phrase, such as the proper name *Ben **'s** Chili Bowl* or the shorter 'local genitive': *Ben **'s** is a great restaurant* (Quirk and Greenbaum, 1973, pp. 329–330).

Supersense	's %	PRP$ %
Agent	8.5	11.4
Beneficiary	3.4	0.8
Causer	0.0	0.4
Duration	3.4	0.0
Experiencer	6.8	7.8
Gestalt	15.3	7.3
Locus	1.7	0.0
OrgRole	5.1	2.8
Originator	10.2	6.7
Possessor	25.4	34.7
Recipient	0.0	1.1
SocialRel	10.2	20.0
Theme	1.7	1.1
Time	5.1	0.0
Whole	3.4	6.1
# tokens	59	1045

Table 2: Token distribution of supersenses: **'s** vs. possessive pronouns. The percentages in each column sum to 100%.

In addition, various idioms license a noun phrase constituent that is required to be possessively marked (but the NP itself is not fixed). These include **possessed idioms**, where the pronoun in the possessive slot agrees with an NP in another syntactic position (Bond et al., 2013, 2015): e.g., *try [one] **'s** best* (which can be instantiated as *I tried **my** best*, not **I tried Mary **'s** best* or **I tried **her** best*); *be quick on [one] **'s** feet*; and *be on [one] **'s** own*. Other idioms with a possessive slot do not enforce agreement: e.g., *[one] **'s** hour/time of need* (*I helped in Mary **'s** hour of need*). If the **'s** clitic was used, it was annotated as a fixed part of the idiom, so we did not assign it a label. In labeling possessive pronouns used in idioms, our policy was to assign a semantic label only if the possessive relationship was compositional, and a special label `$ otherwise (11 tokens). 4 of these were the generic pronoun **your** in the expression ***your** average/typical/usual* NP.

Borderline cases. As is par for the course in broad-coverage semantic annotation, many constructions in our corpus could be considered to fit into more than one supersense, or a different supersense depending on context or interpretation. We give two examples:

1. *Jane **'s** knowledge of the subject matter*: Possessive nominals modifying *knowledge*, like the example above, can plausibly be interpreted as POSSESSOR, EXPERIENCER, GESTALT, and possibly others. Speakers can disagree in how they interpret knowledge with respect to these different perspectives. For the sake of annotation, it is helpful to have a clear and consistent convention. We annotate *Jane **'s** knowledge* as EXPERIENCER.

2. *I put in **my** order* vs. *I picked up **my** order*: The noun *order* can denote a command or request (where the possessive construction is marked as ORIGINATOR) or, by metonymy, an

A1 / A2	AGENT	BENEFICIARY	CHARACTERISTIC	EXPERIENCER	GESTALT	ORGROLE	ORIGINATOR	PART/PORTION	POSSESSION	POSSESSOR	RECIPIENT	SOCIALREL	SOURCE	THEME	WHOLE
AGENT	9		1				3			2				1	
BENEFICIARY				1											
EXPERIENCER				4										1	
GESTALT					2								2		
ORIGINATOR					1		8								
POSSESSOR		1	2	1	2		1		2	22					
SOCIALREL	1				1	1	1	1		3	1	21			
THEME											1				
WHOLE															12

Table 3: Interannotator confusion matrix for s-genitives.

item which had been requested (in which case the possessive marking on the orderer is labeled POSSESSOR). Annotators should judge by context.

Table 3 provides a confusion matrix between the two annotators. Annotator 1 was more prone to using prototypical possessive supersenses while Annotator 2 was more willing to use PARTICIPANT subtypes. As examples of disagreement:

- ***my*** *home team* was labeled SOCIALREL by A1 and GESTALT by A2—a distinction in whether the team has an interpersonal relationship with or is a property of the speaker;
- in *This one won't be getting* ***my*** *business*, **my** was labeled AGENT by A1 and ORIGINATOR by A2—a distinction in whether business is considered an action or something transfered;
- in *I have many fond memories of* ***my*** *college evenings*, **my** was labeled POSSESSOR by A1 and EXPERIENCER by A2—a distinction in whether evenings are thought to be possessed or experienced.

Recent improvements to the guidelines have sought to address some of these sources of disagreement.

Apart from occasional disagreements about the semantic boundaries of categories, we have noticed that annotating the s-genitive can feel counterintuitive: it is easy to accidentally focus on the role of the head noun and apply the inverse label, e.g. POSSESSION instead of POSSESSOR or CHARACTERISTIC instead of GESTALT. Annotators should therefore be vigilant about s-genitives, and a warning should be generated if an unlikely s-genitive supersense is applied (for instance, POSSESSION or CHARACTERISTIC, neither of which is attested for any s-genitives in our data).

5. Discussion

5.1. Distribution: s-genitives vs. of-genitives

As table 1 shows, there are noticeable differences in distribution of supersenses between **of** and genitive case markers (**'s** and possessive pronouns). (The distribution of **'s** is roughly similar to that of possessive pronouns, though there are only 59 tokens of the former. See table 2.) The differences are stark. Supersenses that are only attested for **of** include CHARACTERISTIC, IDENTITY, QUANTITY, SPECIES, STIMULUS, STUFF, and TOPIC. Supersenses that are only attested for **'s** and possessive prepositions include BENEFICIARY, ORGROLE, RECIPIENT, and perhaps most interestingly POSSESSOR.

Semantic differences in distribution are known to play a role in English's *genitive alternation* studied in previous work (Shih et al., 2012; Wolk et al., 2013). Many other factors have been established as well. Our data augments the empirical record.

5.2. Comparison to previous schemes

Previous annotation schemes for English possessive constructions have been based on attempts to characterize relations between nominals. Badulescu and Moldovan (2009, herafter "BM") adapted one such set of 35 semantic categories (from Moldovan et al., 2004), resulting in 22 labels for English possessive constructions. Tratz and Hovy (2013, "TH"), building on prior work by BM and others, developed a set of 18 semantic categories specific to the s-genitive. They did so iteratively, adjusting the categories as needed to reduce disagreements between annotators. TH's study and inventory were limited to the s-genitive (**'s** and possessive pronouns).

Our approach, by contrast, uses an adposition annotation scheme as the point of departure. We find that indeed, many of the semantic relations expressed with prepositions like **in**, **with**, etc. can also be conveyed with **of** and **'s** (*the man* ***in/with*** *a coat* vs. *the man's coat*; *the financial markets* ***in/of*** *the largest European capitals*). While **'s** was not considered as a preposition in the development of the supersenses, we find that nearly all its usages in a corpus are covered by the supersense categories (rare exceptions are due to idioms). A direct comparison between the schemes is given in table 4. Overall, the supersense scheme is slightly finer-grained than the others, which is not surprising as it has more labels (29, including the idiom category, versus 18 for TH and 22 for BM). Most of the supersenses have a counterpart in at least the BM scheme, which covers both kinds of genitive constructions. A notable difference is that certain supersenses distinguish directionality where the corresponding TH labels do not—e.g. GESTALT is distinguished from CHARACTERISTIC, and POSSESSOR from POSSESSION, which

Supersense	(Tratz and Hovy, 2013)	(Badulescu and Moldovan, 2009)	example
Agent	Subjective	Agent	**her** help
Beneficiary	Recipient	Recipient	children**'s** clothes
Causer	Producer's Product	Cause	victim **of** hard times
Characteristic		Property	a place **of** beauty
ComparisonRef			the opposite **of** QuikTrip
Cost	Extent	Measure	a price **of** $160
Duration	Extent	Temporal	a year**'s** worth of dirty clothes
Experiencer	Mental Experiencer	Experiencer	**our** needs
Extent	Extent	Extent	a production increase **of** 10%
Gestalt	Attribute	Property	**Her** flexibility and accessibility
Identity		Hypernymy	the city **of** Dallas
Instrument		Means	usage **of** product barcodes
Locus	Location	Location/Space	Miami**'s** best beach
Manner			reeks **of** old cigarette smoke
OrgRole	Member's Collection	Associated With, Source/From	**his** firm, prime minister **of** Japan
Originator	Producer's Product	Make/Produce	**his** advice
Possession		Possession	the owner **of** a new car
Possessor	Controller/Owner/User	Possession	**her** dog
Quantity		Measure	plenty **of** parking
Recipient	Recipient	Recipient	**my** delivery
SocialRel	Kinship, Member's Collection, Other Relational Noun	Kinship	**her** brother, **her** family **her** friend
Species			that kind **of** behavior
Stimulus		Stimulus	a fear **of** snakes
Stuff			the piece **of** metal
Theme	Objective	Theme, Accompaniment, Result	**my** next haircut, "solution **of** the problem" B., "result of the review" B.
Time	Temporal	Temporal	today**'s** tough times
Topic		Topic, Depiction-Depicted	apprised **of** status, a picture **of** the moon
Whole	Partitive	Part-Whole	the inside **of** my car
(idiom)	Adjective Determined, Possessive Coumpound, Other		Ben**'s** Chili Bowl, "**his** fellow brit" T.
N/A	Other	Other	"state **of** emergency" B., "**your** lordship" T.

Table 4: Mappings between possessive categories. Quoted examples followed by B. are attributed to Badulescu and Moldovan (2009) and ones ending in T. are from Tratz and Hovy (2013). Tratz and Hovy's (2013) labels only apply to s-genitives.

could be useful for making inferences about the related NPs. BM distinguishes direction with a suffix '\R' for reversed relations—e.g. 'Possession' vs. 'Possession\R'.

The supersense schema aims to be applicable to adpositional constructions in other languages. Based on our findings, we speculate that it will be applicable to other language's possessive constructions as well, though this will need to be tested in future work. Our data is a step toward future work comparing genitive constructions across languages.

6. Conclusion

We have shown that an existing broad-coverage semantic annotation scheme for adpositions can be applied to English possessive constructions. Annotation reveals major distributional differences between prepositional **of** and genitive case marking (the s-genitive). Our policies for s-genitives have been incorporated into the latest version of the English annotation guidelines for adposition and case supersenses (Schneider et al., 2018). Our annotated corpus is available for download at `https://github.com/nert-gu/streusle/releases/tag/v4.0`.

Acknowledgments

We are grateful for the advice and expertise of Omri Abend, Archna Bhatia, Na-Rae Han, Jena D. Hwang, Sarah R. Moeller, Tim O'Gorman, and Vivek Srikumar. We also thank the MASC-SLL conference attendees for feedback on a presentation of an early version of this work.

References

Badulescu, Adriana and Moldovan, Dan (2009). A Semantic Scattering model for the automatic interpretation of English genitives. *Natural Language Engineering*, 15(2):215–239.

Baldwin, Timothy and Kim, Su Nam (2010). Multiword expressions. In Indurkhya, Nitin and Damerau, Fred J., editors, *Handbook of Natural Language Processing, Second Edition*, pages 267–292. CRC Press, Taylor and Francis Group, Boca Raton, FL.

Bond, Francis, Ho, Jia Qian, and Flickinger, Daniel (2015). Feeling our way to an analysis of English possessed idioms. In *Proc. of HPSG*, pages 61–74. Singapore.

Bond, Francis, Sameha, Sheefa Samara, and Flickinger, Dan (2013). Making English possessed idioms our own. In *Proc. of HPSG*. Berlin, Germany.

Heine, Bernd (2006). *Possession: Cognitive Sources, Forces, and Grammaticalization*. Cambridge University Press, Cambridge, UK.

Hwang, Jena D., Bhatia, Archna, Han, Na-Rae, O'Gorman, Tim, Srikumar, Vivek, and Schneider, Nathan (2017). Double trouble: the problem of construal in semantic annotation of adpositions. In *Proc. of *SEM*, pages 178–188. Vancouver, Canada.

Moldovan, Dan, Badulescu, Adriana, Tatu, Marta, Antohe, Daniel, and Girju, Roxana (2004). Models for the semantic classification of noun phrases. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, pages 60–67. Association for Computational Linguistics.

Nikiforidou, Kiki (1991). The meanings of the genitive: a case study in semantic structure and semantic change. *Cognitive Linguistics*, 2(2):149–205.

Quirk, Randolph and Greenbaum, Sidney (1973). *A concise grammar of contemporary English*. Harcourt School.

Rosenbach, Anette (2002). *Genitive Variation in English: Conceptual Factors in Synchronic and Diachronic Studies*. Walter de Gruyter.

Schneider, Nathan, Hwang, Jena D., Bhatia, Archna, Han, Na-Rae, Srikumar, Vivek, O'Gorman, Tim, and Abend, Omri (2017). Adposition Supersenses v2. arXiv preprint: `http://arxiv.org/abs/1704.02134v1`.

Schneider, Nathan, Hwang, Jena D., Bhatia, Archna, Han, Na-Rae, Srikumar, Vivek, O'Gorman, Tim, Moeller, Sarah R., Abend, Omri, Blodgett, Austin, and Prange, Jakob (2018). Adposition and Case Supersenses v2: Guidelines for English. arXiv preprint: `https://arxiv.org/abs/1704.02134`.

Schneider, Nathan, Hwang, Jena D., Srikumar, Vivek, Green, Meredith, Suresh, Abhijit, Conger, Kathryn, O'Gorman, Tim, and Palmer, Martha (2016). A corpus of preposition supersenses. In *Proc. of LAW X – the 10th Linguistic Annotation Workshop*, pages 99–109. Berlin, Germany.

Schneider, Nathan, Onuffer, Spencer, Kazour, Nora, Danchik, Emily, Mordowanec, Michael T., Conrad, Henrietta, and Smith, Noah A. (2014). Comprehensive annotation of multiword expressions in a social web corpus. In Calzolari, Nicoletta, Choukri, Khalid, Declerck, Thierry, Loftsson, Hrafn, Maegaard, Bente, Mariani, Joseph, Moreno, Asuncion, Odijk, Jan, and Piperidis, Stelios, editors, *Proc. of LREC*, pages 455–461. Reykjavík, Iceland.

Schneider, Nathan, Srikumar, Vivek, Hwang, Jena D., and Palmer, Martha (2015). A hierarchy with, of, and for preposition supersenses. In *Proc. of The 9th Linguistic Annotation Workshop*, pages 112–123. Denver, Colorado, USA.

Shih, Stephanie, Grafmiller, Jason, Futrell, Richard, and Bresnan, Joan (2012). Rhythm's role in genitive construction choice in spoken English.

Stefanowitsch, Anatol (2003). Constructional semantics as a limit to grammatical alternation: The two genitives of english. *Topics in English Linguistics*, 43:413–444.

Taylor, John R. (1996). *Possessives in English: An Exploration in Cognitive Grammar*. Clarendon Press, Oxford, UK.

Tratz, Stephen and Hovy, Eduard (2013). Automatic interpretation of the English possessive. In *Proc. of ACL*, pages 372–381. Sofia, Bulgaria.

Viera, Anthony J., Garrett, Joanne M., et al. (2005). Understanding interobserver agreement: the kappa statistic. *Fam Med*, 37(5):360–363.

Wolk, Christoph, Bresnan, Joan, Rosenbach, Anette, and Szmrecsanyi, Benedikt (2013). Dative and genitive variability in Late Modern English: Exploring cross-constructional variation and change. *Diachronica*, 30(3):382–419.

A Corpus of Metaphor Novelty Scores for Syntactically-Related Word Pairs

Natalie Parde and Rodney D. Nielsen
Department of Computer Science and Engineering
University of North Texas
{natalie.parde, rodney.nielsen}@unt.edu

Abstract

Automatically scoring metaphor novelty is an unexplored topic in natural language processing, and research in this area could benefit a wide range of NLP tasks. However, no publicly available metaphor novelty datasets currently exist, making it difficult to perform research on this topic. We introduce a large corpus of metaphor novelty scores for syntactically related word pairs, and release it freely to the research community. We describe the corpus here, and include an analysis of its score distribution and the types of word pairs included in the corpus. We also provide a brief overview of standard metaphor detection corpora, to provide the reader with greater context regarding how this corpus compares to other datasets used for different types of computational metaphor processing. Finally, we establish a performance benchmark to which future researchers can compare, and show that it is possible to learn to score metaphor novelty on our dataset at a rate ignificantly better than chance or naïve strategies.

Keywords: metaphor, metaphor novelty, figurative language, crowdsourcing

1. Introduction

Metaphors exist along a novelty continuum, with highly conventional metaphors (those that a person is likely to encounter on a day-to-day basis, e.g., "I *spent* an *hour* on my homework.") at one end, and highly novel metaphors (those that stand out as being particularly uncommon or creative, e.g., "She *frowned* like a *thunderstorm*.") at the other, with many falling somewhere in between. Despite this, research on computational metaphor processing to date has largely viewed metaphor as a binary phenomenon. This simplification has at least partially arisen due to a lack of metaphor novelty-annotated data, as pointed out by Haagsma and Bjerva (2016).

Resource scarcity notwithstanding, having a reliable means of automatically scoring metaphor novelty could be of benefit to many tasks. Word sense disambiguation, for example, can be quite adept at handling conventional metaphors (simply treating them as additional word senses), but novel metaphors require more complex processing (Shutova, 2015; Haagsma and Bjerva, 2016). Automatically grading essays, asessing text difficulty levels, and identifying interesting topics for book discussion systems could all benefit from an ability to automatically score metaphors. Metaphor novelty can also be leveraged for psychological and cognitive health assessment. For example, Gutierrez et al. (2017) applied standard, binary metaphor detection to open-ended patient interviews to aid in the detection of schizophrenia, motivated by clinical research indicating that schizophrenic patients tend to produce particularly bizarre figurative speech (Kuperberg, 2010; Billow et al., 1997). Although the binary metaphor labels were useful, one can hypothesize that performance could be further magnified with continuous novelty scores. Likewise, research in cognitive neuroscience has found that patients with Alzheimer's disease often struggle with comprehending novel metaphors but not conventional metaphors (Amanzio et al., 2008). Systems capable of distinguishing between different grades of metaphor novelty could thus learn ways to assess cognitive health based on a user's perceived comprehension of different metaphors.

However, data scarcity currently acts as a barrier to research activity in these promising application areas. In this work, we remove that barrier by presenting a large (18,000+ instances) corpus of syntactically-related word pairs from four domains annotated for metaphor novelty, and make it publicly available to researchers under the Creative Commons Attribution ShareAlike 3.0 Unported License. We first provide an overview of existing metaphor datasets. Then, we describe our data collection process, and present an analysis of this new corpus. Finally, as a proof-of-concept we develop a metaphor novelty scoring approach and train it on this new dataset to establish a performance benchmark for this task.

2. Related Work

Although no datasets annotated for metaphor novelty currently exist, a number of datasets have been released in the past for research on traditional metaphor detection. Details about currently-existing free, publicly-available, English-language metaphor datasets are provided in Table 1.

The most widely-used of these has been the Vrije Universiteit Amsterdam Metaphor Corpus (VUAMC) (Steen et al., 2010). The VUAMC is a subset of the BNC Baby corpus (Consortium, 2005), comprised of document fragments from four domains: news, academic text, fiction, and transcribed conversations. Individual words in those documents are labeled as metaphors. Those metaphors are of all part-of-speech types, ranging from conventional to novel; however, novelty is not specified in the annotations.

Other existing metaphor datasets have filled more specialized needs for different tasks. Some of these have included data from multiple languages. Levin et al. (2014), Tsvetkov et al. (2014), and Mohler et al. (2016) all included instances in English, Spanish, Russian, and Farsi in their data. Levin et al.'s dataset contains conventional metaphors from each of those languages belonging to three target domains: poverty, wealth, and taxation. Tsvetkov et al. (2014) included only English instances in their training set, but built

Name	Citation	Instance Type	Label Type	Word Type(s)	Size	Link
VUAMC	(Steen et al., 2010)	Word	Binary	All	187,570	http://ota.ahds.ac.uk/headers/2541.xml
Levin	(Levin et al., 2014)	Word Pair	Binary	Verb-Sub., Verb-Obj., Verb-Adv., Noun-Adj., Noun-Noun, Noun-Compound	7657 (English) 24,277 (All)	Available upon request from the authors.
TroFi	(Birke and Sarkar, 2006)	Word	Binary	Verb	3737	http://natlang.cs.sfu.ca/software/trofi.html
Tsvetkov	(Tsvetkov et al., 2014)	Word Pair, Word Triple	Binary	Adj.-Noun, Sub.-Verb-Obj.	1968 (Eng., AN) 1831 (Eng., SVO) 2608 (All, AN) 2335 (All, SVO)	https://github.com/ytsvetko/metaphor
Mohammad	(Mohammad et al., 2016)	Word	Binary	Verb	1639	http://saifmohammad.com/WebPages/metaphor.html
LCC Metaphor Dataset	(Mohler et al., 2016)	Word Pair	Discrete Range (0-3)	Pairs with nouns, verbs, multi-word expr., adjs., advs.	8724 (Eng., Free) 15,837 (All, Free)	Available upon request from the authors.
Master Metaphor List	(Lakoff, 1994)	Concept Pair	List	All	208	https://tinyurl.com/mastermetaphor
Metaphors of Mind	(Pasanek, 2015)	Sentence	List	All	14,000+	http://metaphors.iath.virginia.edu/
Our Corpus	—	Word Pair	Contin. Range (0-3)	Pairs with nouns, verbs, adjs., advs., personal pronouns	18,439	http://hilt.cse.unt.edu/resources.html

Table 1: Metaphor Datasets

small test sets in Spanish, Russian, and Farsi (as well as English) to study whether metaphor detection models trained on English samples could be applied to other languages. Mohler et al. (2016) produced *metaphoricity* annotations on a scale from 0-3; theirs is the only existing metaphor dataset to use non-binary labels. The free version of their dataset contains English and Spanish instances, and the paid version contains additional instances for those languages as well as instances in Russian and Farsi.

Some metaphor datasets have been principally concerned with the metaphoric and literal uses of different verbs. The TroFi Metaphor Dataset (Birke and Sarkar, 2006) is one of these, comprised of sentences originating in the Wall Street Journal Corpus (Charniak et al., 2000). The sentences contain metaphoric or literal uses of 50 verbs. Similarly, Mohammad et al. (2016) created a dataset containing sentences with metaphoric or literal uses of 440 verbs. Their data also includes emotionality annotations, and annotations indicating relative metaphoricness and emotionality (*more*, *less*, or *equal*) for paired uses of the same verb.

Finally, a few metaphor datasets are simply lists of metaphors. These datasets were developed for research in linguistics or the humanities, rather than for computational analysis. Perhaps the most famous collection of metaphors, the Master Metaphor List (Lakoff, 1994), falls under this category. The Master Metaphor List contains conceptual metaphor mappings that guide the generation of linguistic metaphors (e.g., the conceptual metaphor, TIME is MONEY, gives rise to the linguistic metaphor, "I *spent* an *hour* on my homework."). Also a member of this category is the large list of metaphors created by Pasanek (2015). This dataset is comprised of sentences from primarily 18th-century British literature that convey metaphors of the mind (a sample from the dataset is Jane Austen's *"Astonishment and doubt first seized them; and a shortly succeeding ray of common sense added some bitter emotions of shame."*).

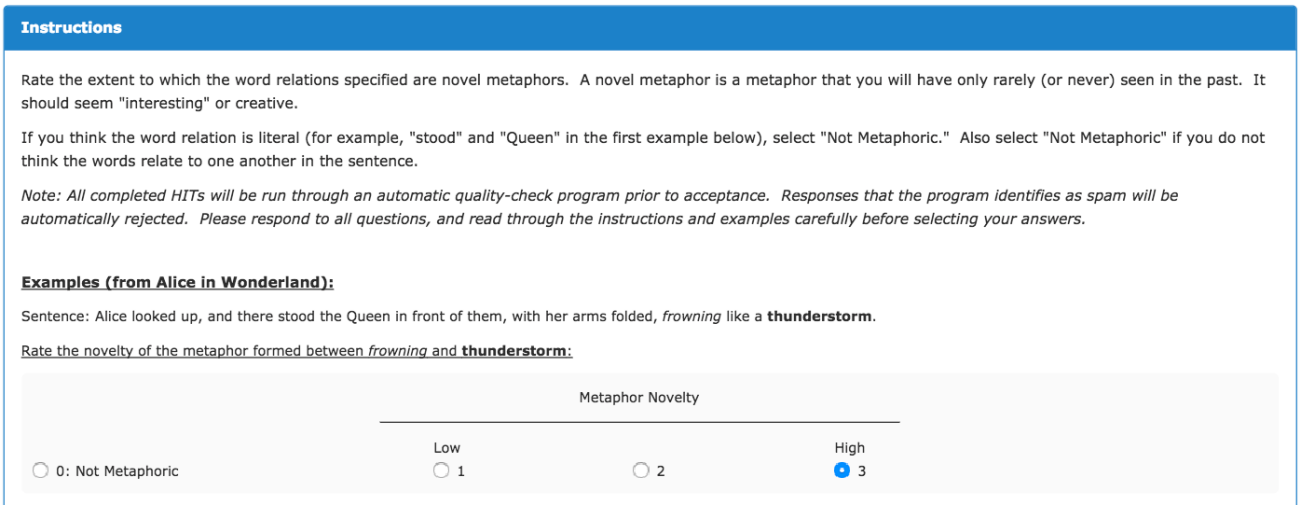


Figure 1: Annotation Instructions

The Metaphors of Mind dataset is still growing, as Pasanek continues to add to it.

Of the datasets described, the closest to ours is the LCC Metaphor Dataset, since it labels word pairs with a range of scores rather than with binary labels. However, the focus of the LCC Metaphor Dataset is on *metaphoricity* (ranging from *no metaphoricity* to *clear metaphor*) rather than *metaphor novelty*. We found upon analysis of the dataset that instances rated with high metaphoricity were not necessarily novel metaphors. The following sample from the LCC Metaphor Dataset illustrates this:

> A measure of the protection provided to an industry by the entire *structure* of *tariffs*, taking into account the effects of tariffs on inputs as well as on outputs.

In the LCC dataset, the word pair *structure* of *tariffs* is scored as a "3" (high metaphoricity), and this is a reasonable metaphoricity score; clearly, a tariff cannot have a physical structure. However, this metaphor is also quite conventional, and unlikely to strike any reader as particularly creative (in fact, many readers may not notice that the expression is figurative at all unless it is explicitly pointed out to them). Thus, its *novelty* score should be low.

3. Data Collection

Our dataset contains continuous metaphor novelty annotations for syntactically-related word pairs extracted from the VUAMC. Word pairs are comprised either of two *content words* (nouns, verbs, adverbs, and adjectives, excluding stopwords, proper nouns, and some auxiliary verbs) or a content word and a personal pronoun. Syntactic relations were identified using Stanford CoreNLP (Manning et al., 2014). We required that each word pair contain at least one content word labeled as a metaphor using the original binary annotations provided with the VUAMC, thus entailing that each word pair was *potentially metaphoric*.[1] Annotators were asked to score the metaphor novelty of each word pair, given the enclosing sentence as context, on a discrete scale from 0 (non-metaphoric) to 3 (highly novel metaphor); the multiple annotations collected for a given word pair were later aggregated to a single continuous label. A sample of the annotation instructions is shown in Figure 1, and Figure 2 illustrates both the annotations included in the original VUAMC and the final, continuous labels included in our dataset, for the same source sentence.

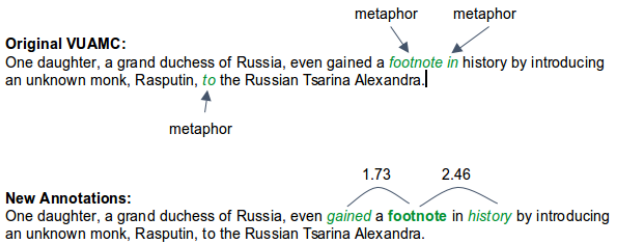


Figure 2: Original VUAMC Corpus and Our Annotations

We collected five annotations from crowd workers for all instances (18,439 word pairs) using Amazon Mechanical Turk (`https://www.mturk.com`). Additionally, we collected annotations from two trained annotators for each instance in the test set (3162 word pairs). The trained annotators used the same labeling interface as the crowd workers. Instances were grouped into Human Intelligence Tasks (HITs), each containing all of the instances associated with 10 sentences, and crowd workers were paid $0.20 per HIT. Overall, 479 crowd workers participated in annotating the dataset. Workers were required to have an overall HIT approval rate (across all HITs on Amazon Mechanical Turk) greater than or equal to 90%.

Crowdsourced annotations were filtered using the algorithm described in Parde and Nielsen (2017), which identifies substandard workers based on their poor correlation with other workers. Workers who were filtered by the algorithm for a given batch of HITs were also disqualified from accepting future HITs.

[1] A word's metaphoric usage is rarely conveyed in all word pair contexts: for example, in "Her *laughter* **waltzed** through the *courtyard*," the pair {*laughter*, **waltzed**} is a novel metaphor but the pair {**waltzed**, *courtyard*} is non-metaphoric.

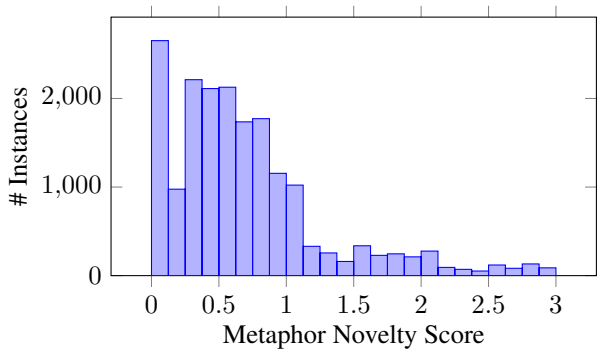


Figure 3: Score Distribution

		Metaphor Word				
		Noun	**Verb**	**Adj.**	**Adv.**	**Total**
Other Word	**Noun**	4352	5630	1533	98	11613
	Verb	2227	1282	275	124	3908
	Adj.	1618	512	83	32	2245
	Adv.	148	418	83	5	654
	Pro.	3	13	3	0	19
	Total	8348	7855	1977	259	18439

Table 2: Word Pair Distribution

3.1. Adjudication

The crowdsourced annotations for the training instances were automatically aggregated to continuous labels using the regression-based approach developed by Parde and Nielsen (2017). Briefly, this approach trains a random subspace regression model on data that has been labeled by both crowd workers and experts, using features based on annotation distribution and estimated annotator quality, to predict optimal aggregations of crowd labels.

The annotations provided by trained annotators for the test set were averaged, unless the annotators disagreed strongly (e.g., a 0 and a 3) or if one of the annotators did not agree with the score produced by averaging. In those cases (111 total), instances were forwarded to a third-party adjudicator to make the final decision. The adjudicated labels comprise the dataset's *gold standard* test labels. Inter-annotator agreement between trained annotators was measured across all 3162 test instances, prior to any adjudication or annotator discussion, using kappa (κ) with quadratic weights between the four "classes" of 0, 1, 2, and 3, resulting in $\kappa = 0.435$. This highlights the fact that scoring metaphor novelty is a difficult task even for humans; however, most of the annotators' disagreements were just between two adjacent (similar) scores. We also computed κ with the more relaxed constraints established by Mohler et al. (2016) (considering scores within a distance of 1 from one another to agree), and this resulted in $\kappa = 0.897$. In our published dataset, we include the original crowdsourced annotations for each instance, the aggregated label for each training instance, and the gold standard label for each test instance.

4. Corpus Analysis

4.1. Score Distribution

The corpus contains many more conventional metaphors than novel metaphors, which mirrors the distribution seen in naturally-occurring text. The score distribution across the full corpus, with scores binned in 0.125 intervals, is shown in Figure 3. Aggregated training scores were scaled to fit the 0-3 range (the aggregation model originally predicted scores within a more narrow range). Scores from 2.0-3.0 represent novel metaphors, scores from 0.0-1.0 represent conventional (fossilized and non-) metaphors, and scores from 1.0-2.0 are metaphors that fall somewhere between conventional and novel.

4.2. Pair Type Distribution

Unlike many existing metaphor detection corpora, our corpus contains a wide variety of word pairs. This makes it possible to capture metaphors that may be missed by more constrained datasets. Table 2 lists the frequencies with which different types of word pairs occur in our corpus, with the horizontal axis corresponding to the pair's focus word (the word labeled as a metaphor in the VUAMC), and the vertical axis corresponding to the word paired with that term. The most common pair type in our dataset is verb metaphors paired with nouns, followed closely by noun metaphors paired with nouns. Other common pair types include noun metaphors paired with verbs, noun metaphors paired with adjectives, adjective metaphors paired with nouns, and verb metaphors paired with verbs (e.g., "She **aims** to *explore* the Denton coffee scene.").

4.3. Dependency Type Distribution

Finally, we analyze the various syntactic relations included in our corpus. Table 3 shows the frequency of each dependency type, the average score associated with it, and the standard deviation of the scores associated with it. The most common relation in our corpus is *nmod* (nominal modifier), followed by *amod* (adjectival modifier), *dobj* (direct object), and *nsubj* (nominal subject). Instances with dependency *amod* had the highest average score, and instances with dependency *nmod:tmod* (temporal modifier) had the lowest. The largest standard deviation among scores was found with instances of type *amod*.

5. Benchmark

Although the focus of this paper is on the corpus itself, as a proof of learnability and to establish a benchmark we also train a metaphor novelty scoring model using our data. To do so, we construct a simple neural network with one hidden layer using Keras (`https://keras.io/`) with a TensorFlow (`https://www.tensorflow.org/`) backend. We apply a random normal initialization function, *tanh* activation, and dropout of 0.1 to both the input layer and hidden layer, and set the number of output nodes to 256 for the input layer and 32 for the hidden layer. We optimize the network using *RMSprop*, an optimization technique that updates learning rates for weights based on running averages of the magnitudes of their recent gradients (Hinton et al., 2012), and use a mean squared error loss function. We train the network (5 epochs, with a training batch size of 32) using the following features:

Relation Type	Freq.	Avg.	SD
acl	599	0.62	0.50
acl:relcl	573	0.51	0.45
advcl	689	0.31	0.32
advmod	709	0.80	0.63
amod	2668	1.07	0.70
appos	123	0.42	0.54
compound	761	0.93	0.66
csubj	42	0.33	0.30
csubjpass	3	0.35	0.09
dep	423	0.45	0.49
dobj	2276	0.73	0.48
iobj	2	0.72	0.01
nmod	4807	0.69	0.59
nmod:npmod	14	0.47	0.37
nmod:poss	62	0.59	0.43
nmod:tmod	45	0.29	0.27
nsubj	1460	0.68	0.54
nsubjpass	444	0.68	0.44
reln-obj[2]	693	0.33	0.31
reln-subj[2]	344	0.42	0.37
subj-obj[2]	949	0.37	0.41
xcomp	753	0.53	0.39

Table 3: Dependency Type Distribution

- **Word Embeddings:** 300-dimensional vectors for the word pair's governor and modifier, extracted from the pretrained Google News word embeddings (Mikolov et al., 2013).
- **Dependency Relation:** One-hot encoded vector indicating the relation between the words in the pair.
- **Word Distance:** Absolute distance between the words in the sentence from which the pair was extracted.

We compare our benchmark approach to several baselines, described in Table 4. Our results are shown in Table 5; we ran our approach 10 times (weights from our approach were randomly initialized) and report average correlation (r) and root mean squared error (RMSE). Differences between our approach and others are statistically significant ($p < 0.0001$).

The results illustrate that it is possible to learn to score metaphor novelty from our dataset at a rate significantly better than chance or naïve strategies like predicting the mean training label. They also provide a performance benchmark for future researchers who use this dataset to develop their own approaches. We explore the task of automatically scoring metaphor novelty further in Parde and Nielsen (2018).

6. Conclusion

In this work, we describe a new, large dataset (18,000+ instances) annotated for metaphor novelty, built on top of the VUAMC. We make this dataset publicly available[3] under the Creative Commons Attribution ShareAlike 3.0 Unported License. We provide an overview of existing metaphor datasets, and a comprehensive analysis of our new corpus. Finally, we establish a performance benchmark on this dataset to which other researchers may compare their work[4]. In the future, we plan to improve upon the performance of our metaphor novelty scoring system. It is our hope that the availability of this dataset stimulates further research in this area by others as well.

[2]Extracted from subject-relation-object triples generated by Stanford OpenIE (Angeli et al., 2015). Note that upon manual inspection of these pairs, we found many of the reln-obj and reln-subj pairs to be syntactically invalid.

Baseline	Description
RANDOM	Predicts a random continuous value in the range of 0-3 for each instance.
DISTRIBUTION-AWARE RANDOM	Learns a probability density function from the training set and predicts random continuous values from 0-3 with that same distribution.
AVERAGE	Computes the average value from the training set and then predicts that value for each instance.

Table 4: Baseline Approaches

Approach	*r*	RMSE
RANDOM	0.0048	1.4658
DISTRIBUTION-AWARE RANDOM	0.0007	0.8143
AVERAGE	0.0000	0.7191
OURS	**0.4380**	**0.6528**

Table 5: Results

7. Acknowledgements

This material is based upon work supported by the NSF Graduate Research Fellowship Program under Grant 1144248, and the NSF under Grant 1262860. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

8. Bibliographical References

Amanzio, M., Geminiani, G., Leotta, D., and Cappa, S. (2008). Metaphor comprehension in alzheimer's disease: Novelty matters. *Brain and Language*, 107(1):1 – 10.

Angeli, G., Johnson Premkumar, M. J., and Manning, C. D. (2015). Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354, Beijing, China. Association for Computational Linguistics.

[3]http://hilt.cse.unt.edu/resources.html

[4]https://github.com/natalieparde/metaphor-novelty-scoring

Billow, R. M., Rossman, J., Lewis, N., Goldman, D., and Raps, C. (1997). Observing expressive and deviant language in schizophrenia. *Metaphor and Symbol*, 12(3):205–216.

Birke, J. and Sarkar, A. (2006). Clustering approach for the nearly unsupervised recognition of nonliteral language. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, April.

Charniak, Eugene and Blaheta, Don and Ge, Niyu and Hall, Keith and Hale, John and Johnson, Mark. (2000). *BLLIP 1987-89 WSJ Corpus*. Linguistic Data Consortium, Penn Treebank, Release 1, ISLRN 233-420-716-637-7.

BNC Consortium. (2005). *The BNC Baby*. Oxford University Computing Services, British National Corpus, version 2.

Gutierrez, E. D., Cecchi, G., Corcoran, C., and Corlett, P. (2017). Using automated metaphor identification to aid in detection and prediction of first-episode schizophrenia. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2913–2920.

Haagsma, H. and Bjerva, J. (2016). Detecting novel metaphor using selectional preference information. In *Proceedings of the Fourth Workshop on Metaphor in NLP*, pages 10–17, San Diego, California, June. Association for Computational Linguistics.

Hinton, G., Srivastava, N., and Swersky, K. (2012). Neural networks for machine learning-lecture 6a-overview of mini-batch gradient descent.

Kuperberg, G. R. (2010). Language in schizophrenia part 1: an introduction. *Language and linguistics compass*, 4(8):576–589.

Lakoff, G. (1994). *Master Metaphor List*. University of California.

Levin, L., Mitamura, T., Macwhinney, B., Fromm, D., Carbonell, J., Feely, W., Frederking, R., Gershman, A., and Ramirez, C. (2014). Resources for the detection of conventionalized metaphors in four languages. In *Proce. of the Ninth International Conf. on Language Resources and Evaluation*. European Language Resources Association, may.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Mohammad, S. M., Shutova, E., and Turney, P. D. (2016). Metaphor as a medium for emotion: An empirical study. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (*Sem)*, Berlin, Germany.

Mohler, M., Brunson, M., Rink, B., and Tomlinson, M. (2016). Introducing the lcc metaphor datasets. In *Proc. of the Tenth International Conf. on Language Resources and Evaluation*, May.

Parde, N. and Nielsen, R. (2017). Finding patterns in noisy crowds: Regression-based annotation aggregation for crowdsourced data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1908–1913, Copenhagen, Denmark, September. Association for Computational Linguistics.

Parde, N. and Nielsen, R. D. (2018). Exploring the terrain of metaphor novelty: A regression-based approach for automatically scoring metaphors. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana.

Pasanek, B. (2015). *Metaphors of Mind: An Eighteenth-Century Dictionary*. Johns Hopkins University Press.

Shutova, E. (2015). Design and evaluation of metaphor processing systems. *Computational Linguistics*, 41(4):579–623, December.

Steen, Gerard J. and Dorst, Aletta G. and Hermann, J. Berenike and Kaal, Anna and Krennmayr, Tina and Pasma, Trijntje. (2010). *A Method for Linguistic Metaphor Identification: From MIP to MIPVU*. John Benjamins Publishing Company, Converging evidence in language and communication research.

Tsvetkov, Y., Boytsov, L., Gershman, A., Nyberg, E., and Dyer, C. (2014). Metaphor detection with cross-lingual model transfer. In *Proc. of the 52nd Annual Meeting of the ACL*, pages 248–258, June.

Improving Hypernymy Extraction with Distributional Semantic Classes

Alexander Panchenko[1*], **Dmitry Ustalov**[2,3*], **Stefano Faralli**[3], **Simone P. Ponzetto**[3], **Chris Biemann**[1]
[*] these authors contributed equally
[1] University of Hamburg, Department of Informatics, Language Technology Group, Germany
[2] University of Mannheim, School of Business Informatics and Mathematics, Data and Web Science Group, Germany
[3] Ural Federal University, Institute of Natural Sciences and Mathematics, Russia
{panchenko,biemann}@informatik.uni-hamburg.de
{dmitry,stefano,simone}@informatik.uni-mannheim.de

Abstract

In this paper, we show how distributionally-induced semantic classes can be helpful for extracting hypernyms. We present methods for inducing sense-aware semantic classes using distributional semantics and using these induced semantic classes for filtering noisy hypernymy relations. Denoising of hypernyms is performed by labeling each semantic class with its hypernyms. On the one hand, this allows us to filter out wrong extractions using the global structure of distributionally similar senses. On the other hand, we infer missing hypernyms via label propagation to cluster terms. We conduct a large-scale crowdsourcing study showing that processing of automatically extracted hypernyms using our approach improves the quality of the hypernymy extraction in terms of both precision and recall. Furthermore, we show the utility of our method in the domain taxonomy induction task, achieving the state-of-the-art results on a SemEval'16 task on taxonomy induction.

Keywords: semantic classes, distributional semantics, hypernyms, co-hyponyms, word sense induction

1. Introduction

Hypernyms are useful in various applications, such as question answering (Zhou et al., 2013), query expansion (Gong et al., 2005), and semantic role labelling (Shi and Mihalcea, 2005) as they can help to overcome sparsity of statistical models. Hypernyms are also the building blocks for learning taxonomies from text (Bordea et al., 2016). Consider the following sentence: "This café serves fresh *mangosteen* juice". Here the infrequent word "mangosteen" may be poorly represented or even absent in the vocabulary of a statistical model, yet it can be substituted by lexical items with better representations, which carry close meaning, such as its hypernym "fruit" or one of its close co-hyponyms, e.g. "mango".

Currently available approaches to hypernymy extraction focus on the acquisition of individual binary hypernymy relations (Hearst, 1992; Snow et al., 2004; Weeds et al., 2014; Shwartz et al., 2016; Glavaš and Ponzetto, 2017). Frequencies of the extracted relations usually follow a power-law, with a long tail of noisy extractions containing rare words. We propose a method that performs post-processing of such noisy binary hypernyms using distributional semantics, cf. Figure 1. Namely, we use the observation that distributionally related words are often are co-hyponyms (Wandmacher, 2005; Heylen et al., 2008) and operationalize it to perform filtering of noisy relations by finding dense graphs composed of both hypernyms and co-hyponyms.

The contribution of the paper is an unsupervised method for post-processing of noisy hypernymy relations based on clustering of graphs of word senses induced from text. The idea to use distributional semantics to find hypernyms seems natural and has been widely used. However, the existing methods used distributional, yet *sense-unaware* and *local* features. We are the first to use *global sense-aware distributional structure* via the induced semantic classes to improve hypernymy extraction. The implementation of our method and the induced language resources (distributional semantic classes and cleansed hypernymy relations) are available online.[1]

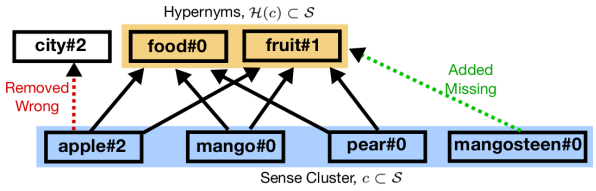


Figure 1: Our approach performs post-processing of hypernymy relations using distributionally induced semantic classes, represented by clusters of induced word senses labeled with noisy hypernyms. The word postfix, such as `#1`, is an ID of an induced sense. The wrong hypernyms outside the cluster labels are removed, while the missing ones not present in the noisy database of hypernyms are added.

2. Related Work

2.1. Extraction of Hypernyms

In her pioneering work, Hearst (1992) proposed to extract hypernyms based on lexical-syntactic patterns from text. Snow et al. (2004) learned such patterns automatically based on a set of hyponym-hypernym pairs. Pantel and Pennacchiotti (2006) presented another approach for weakly supervised extraction of similar extraction patterns. These approaches use some training pairs of hypernyms to bootstrap the pattern discovery process. For instance, Tjong Kim Sang (2007) used web snippets as a corpus for extraction of hypernyms. More recent approaches exploring the use of distributional word representations for extraction of

[1] https://github.com/uhh-lt/mangosteen

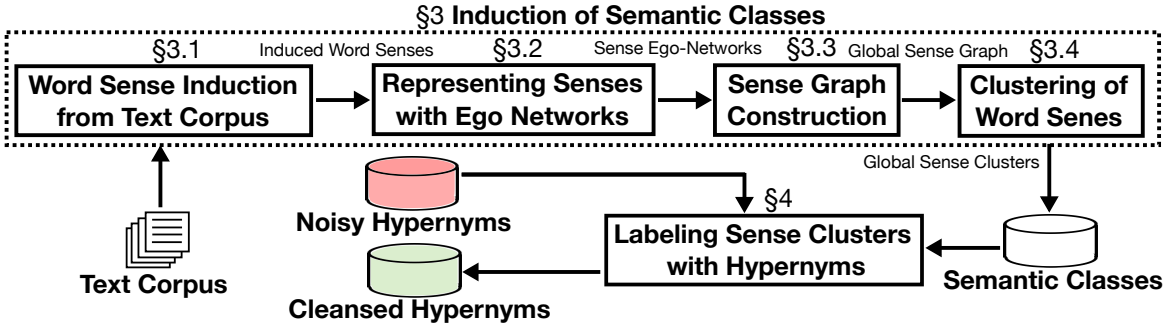


Figure 2: Outline of our approach: sense-aware distributional semantic classes are induced from a text corpus and then used to filter noisy hypernyms database (e.g. extracted by an external method from a text corpus).

hypernyms and co-hyponyms include (Roller et al., 2014; Weeds et al., 2014; Necsulescu et al., 2015; Vylomova et al., 2016). They rely on two distributional vectors to characterize a relation between two words, e.g. on the basis of the difference of such vectors or their concatenation. Levy et al. (2015) discovered a tendency to lexical memorization of such approaches, hampering their generalization to other domains.

Fu et al. (2014) relied on an alternative approach where a projection matrix is learned, which transforms a distributional vector of a hyponym to the vector of its hypernym. Ustalov et al. (2017a) improved this method by adding regularizers in the model that take into account negative training samples and the asymmetric nature of the hypernyms.

Recent approaches to hypernym extraction focused on learning *supervised* models based on a combination of syntactic patterns and distributional features (Shwartz et al., 2016). Note that while methods, such as (Mirkin et al., 2006) and (Shwartz et al., 2016) use distributional features for extraction of hypernyms, in contrast to our method, they do not take into account word senses and global distributional structure.

Seitner et al. (2016) performed extraction of hypernyms from the web-scale Common Crawl[2] text corpus to ensure high lexical coverage. In our experiments, we use this web-scale database of noisy hypernyms, as the large-scale repository of automatically extracted hypernyms to date.

2.2. Taxonomy and Ontology Learning

Most relevant in the context of automatic construction of lexical resource are methods for building resources from text (Caraballo, 1999; Biemann, 2005; Cimiano, 2006; Bordea et al., 2015; Velardi et al., 2013) as opposed to methods that automatically construct resources from semi-structured data (Auer et al., 2007; Navigli and Ponzetto, 2012) or using crowdsourcing (Biemann, 2013; Braslavski et al., 2016).

Our representation differs from the global hierarchy of words as constructed e.g. by (Berant et al., 2011; Faralli et al., 2016), as we are grouping many lexical items into a labeled sense cluster as opposed to organizing them in deep hierarchies. Kozareva and Hovy (2013) proposed a taxonomy induction method based on extraction of hypernyms using the doubly-anchored lexical patterns. Graph algorithms are used to induce a proper tree from the binary relations harvested from text.

2.3. Induction of Semantic Classes

This line of research starts with (Lin and Pantel, 2001), where sets of similar words are clustered into concepts. While this approach performs a hard clustering and does not label clusters, these drawbacks are addressed in (Pantel and Lin, 2002), where words can belong to several clusters, thus representing senses, and in (Pantel and Ravichandran, 2004), where authors aggregate hypernyms per cluster, which come from Hearst patterns. The main difference to our approach is that we explicitly represent senses both in clusters and in their hypernym labels, which enables us to connect our sense clusters into a global taxonomic structure. Consequently, we are the first to use semantic classes to improve hypernymy extraction.

Ustalov et al. (2017b) proposed a synset induction approach based on global clustering of word senses. The authors used the graph constructed of dictionary synonyms, while we use distributionally-induced graphs of senses.

3. Unsupervised Induction of Distributional Sense-Aware Semantic Classes

As illustrated in Figure 2, our method induces a sense inventory from a text corpus using the method of (Faralli et al., 2016; Biemann et al., 2018), and clusters these senses. Sample word senses from the induced sense inventory are presented in Table 1. The difference of the induced sense inventory from the sense clustering presented in Table 2 is that word senses in the induced resource are specific to a given target word, e.g. words "apple" and "mango" have distinct "fruit" senses, represented by a list of related senses. On the other hand, sense clusters represent a global and not a local clustering of senses, i.e. the "apple" in the "fruit" sense can be a member of only one cluster. This is similar to WordNet, where one sense can only belong to a single synset. Below we describe each step of our method.

3.1. Word Sense Induction from a Text Corpus

Each word sense s in the induced sense inventory $\mathcal{S}$ is represented by a list of neighbors $\mathcal{N}(s)$, see Table 1 for an example. Extraction of this network is performed using the method of Faralli et al. (2016) and involves three steps: (1) building a distributional thesaurus, i.e. a graph

[2] http://www.commoncrawl.org

ID: Word Sense, $s \in \mathcal{S}$	Local Sense Cluster: Related Senses, $\mathcal{N}(s) \subset \mathcal{S}$	Hypernyms, $\mathcal{H}(s) \subset \mathcal{S}$
mango#0	peach#1, grape#0, plum#0, apple#0, apricot#0, watermelon#1, banana#1, coconut#0, pear#0, fig#0, melon#0, **mangosteen#0**, ...	fruit#0, food#0, ...
apple#0	mango#0, pineapple#0, banana#1, melon#0, grape#0, peach#1, watermelon#1, apricot#0, cranberry#0, pumpkin#0, **mangosteen#0**, ...	fruit#0, crop#0, ...
Java#1	C#4, Python#3, Apache#3, Ruby#6, Flash#1, C++#0, SQL#0, ASP#2, Visual Basic#1, CSS#0, Delphi#2, MySQL#0, Excel#0, Pascal#0, ...	programming language#3, language#0, ...
Python#3	PHP#0, Pascal#0, Java#1, SQL#0, Visual Basic#1, C++#0, JavaScript#0, Apache#3, Haskell#5, .NET#1, C#4, SQL Server#0, ...	language#0, technology#0, ...

Table 1: Sample induced sense inventory entries representing "fruits" and "programming language" senses. Each word sense s is represented with a list of related senses $\mathcal{N}(s)$ and the list of hypernyms $\mathcal{H}(s)$. The hypernyms can be used as human-interpretable sense labels of the sense clusters. One sense s, such as "apple#0", can appear in multiple entries.

ID	Global Sense Cluster: Semantic Class, $c \subset \mathcal{S}$	Hypernyms, $\mathcal{H}(c) \subset \mathcal{S}$
1	peach#1, banana#1, pineapple#0, berry#0, blackberry#0, grapefruit#0, strawberry#0, blueberry#0, fruit#0, grape#0, melon#0, orange#0, pear#0, plum#0, raspberry#0, watermelon#0, apple#0, apricot#0, watermelon#0, pumpkin#0, berry#0, **mangosteen#0**, ...	vegetable#0, fruit#0, crop#0, ingredient#0, food#0, ·
2	C#4, Basic#2, Haskell#5, Flash#1, Java#1, Pascal#0, Ruby#6, PHP#0, Ada#1, Oracle#3, Python#3, Apache#3, Visual Basic#1, ASP#2, Delphi#2, SQL Server#0, CSS#0, AJAX#0, JavaScript#0, SQL Server#0, Apache#3, Delphi#2, Haskell#5, .NET#1, CSS#0, ...	programming language#3, technology#0, language#0, format#2, app#0

Table 2: Sample of the induced sense clusters representing "fruits" and "programming language" semantic classes. Similarly to the induced word senses, the semantic classes are labeled with hypernyms. In contrast to the induced word senses, which represent a local clustering of word senses (related to a given word) semantic classes represent a global sense clustering of word senses. One sense c, such as "apple#0", can appear only in a single cluster.

of related ambiguous terms (Biemann and Riedl, 2013); (2) word sense induction via clustering of ego networks (Widdows and Dorow, 2002; Everett and Borgatti, 2005) of related words using the Chinese Whispers graph clustering algorithm (Biemann, 2006); (3) disambiguation of related words and hypernyms. The word sense inventory used in our experiment[3] was extracted from a 9.3 billion tokens corpus, which is a concatenation of Wikipedia[4], ukWac (Ferraresi et al., 2008), LCC (Richter et al., 2006) and Gigaword (Graff and Cieri, 2003). Note that analogous graphs of senses can be obtained using word sense embeddings, see (Neelakantan et al., 2014; Bartunov et al., 2016). Similarly to any other distributional word graph, the induced sense inventory sense network is scale-free, cf. (Steyvers and Tenenbaum, 2005). Our experiments show that a global clustering of this network can lead to a discovery of giant components, which are useless in our context as they represent no semantic class. To overcome this problem, we re-build the sense network as described below.

3.2. Representing Senses with Ego Networks

To perform a global clustering of senses, we represent each induced sense s by a second-order *ego network* (Everett and Borgatti, 2005). An ego network is a graph consisting of all related senses $\mathcal{R}(s)$ of the ego sense s reachable via a path of length one or two, defined as:

$$\{s_j : (s_j \in \mathcal{N}(s)) \vee (s_i \in \mathcal{N}(s) \wedge s_j \in \mathcal{N}(s_i))\}. \quad (1)$$

Each edge weight $\mathcal{W}_s(s_i, s_j)$ between two senses is taken from the induced sense inventory network (Faralli et al., 2016) and is equal to a distributional semantic relatedness score between s_i and s_j.

Senses in the induced sense inventory may contain a mixture of different senses introducing noise in a global clustering: cf. Figure 3, where "Python" in the animal sense is related to both car and snake senses. To minimize the impact of the word sense induction errors, we filter out ego networks with a highly segmented structure. Namely, we cluster each ego network with the Chinese Whispers algorithm and discard networks for which the cluster containing the target sense s contains less than 80% nodes of the respective network to ensure semantic coherence inside the word groups. Besides, all nodes of a network not appearing in the cluster containing the ego sense s are also discarded.

3.3. Global Sense Graph Construction

The goal of this step is to merge ego networks of individual senses constructed at the previous step into a global graph. We compute weights of the edges of the global graph by counting the number of co-occurrences of the same edge in different networks:

$$\mathcal{W}(s_i, s_j) = \sum_{s \in \mathcal{S}} \mathcal{W}_s(s_i, s_j). \quad (2)$$

For filtering out noisy edges, we remove all edges with the weight less than a threshold t. Finally, we apply the function $E(w)$ that re-scales edge weights. We tested identity function (count) and the natural logarithm (log):

$$\mathcal{W}(s_i, s_j) = \begin{cases} E(\mathcal{W}(s_i, s_j)) & \text{if } \mathcal{W}(s_i, s_j) \geq T, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

[3]The input and output datasets are available for download at https://doi.org/10.5281/zenodo.1174041

[4]https://doi.org/10.5281/zenodo.229904

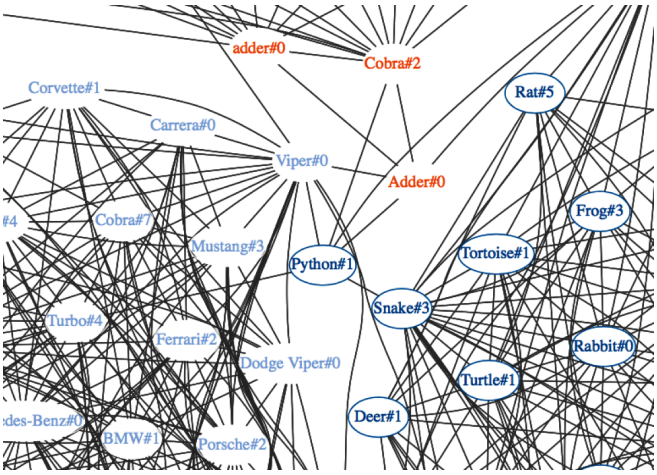


Figure 3: An example of a non-coherent ego network of the automatically induced sense `Python#1`, representing the "animal" sense. We prune it to remove terms not relevant to the animal sense.

3.4. Clustering of Word Senses

The core of our method is the induction of semantic classes by clustering the global graph of word senses. We use the Chinese Whispers algorithm to make every sense appear only in one cluster c. Results of the algorithm are groups of strongly related word senses that represent different concepts (cf. Figure 4). Hypernymy is by definition a relation between nouns. Thus optionally, we remove all single-word senses that do not correspond to nouns using the Pattern library (De Smedt and Daelemans, 2012). This optional mode is configured by the boolean parameter N.

We use two clustering versions in our experiments: the *fine-grained* model clusters 208,871 induced word senses into 1,870 semantic classes, and the *coarse-grained* model that groups 18,028 word senses into 734 semantic classes. To find optimal parameters of our method, we compare the induced labeled sense clusters to lexical semantic knowledge from WordNet 3.1 (Fellbaum, 1998) and BabelNet 3.7 (Navigli and Ponzetto, 2012).

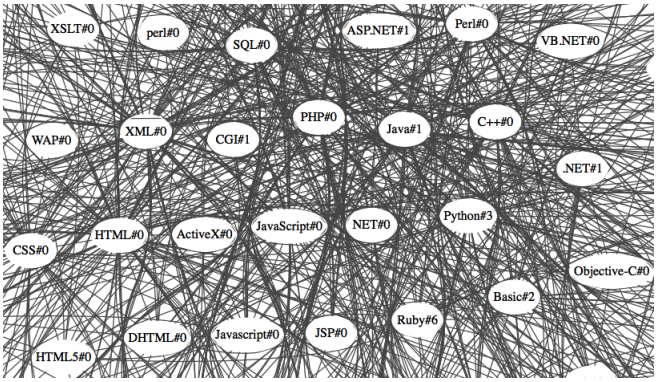


Figure 4: Senses referring to programming languages co-occur in global sense cluster entries, resulting in a densely connected set of co-hyponyms.

4. Denoising Hypernyms using the Induced Distributional Semantic Classes

By labeling the induced semantic classes with hypernyms we can thereby remove wrong ones or add those that are missing as illustrated in Figure 1. Each sense cluster is labeled with the noisy input hypernyms, where the labels are the common hypernyms of the cluster word (cf. Table 2). Hypernyms that label no sense cluster are filtered out. In addition, new hypernyms can be generated as a result of labeling. Additional hypernyms are discovered by propagating cluster labels to the rare words without hypernyms, e.g. "mangosteen" in Figure 1. For labeling we used the tf-idf weighting. Hypernyms that appear in many senses s are weighted down:

$$\text{tf-idf}(h) = \sum_{s \in c} \mathcal{H}(s) \cdot \log \frac{|\mathcal{S}|}{|h \in \mathcal{H}(s) : \forall s \in \mathcal{S}|}, \quad (4)$$

where $\sum_{s \in c} \mathcal{H}(s)$ is a sum of weights for all hypernyms for each sense s, per each cluster c.

We label each sense cluster c with its top five hypernyms $\mathcal{H}(c)$. Each hypernym is disambiguated using the method of Faralli et al. (2016). Namely, we calculate the cosine similarity between the context (the current sense cluster) and the induced senses (local clusters of the ambiguous word).

Distributional representations of rare words, such as "mangosteen" can be less precise than those of frequent words. However, co-occurrence of a hyponym and a hypernym in a single sentence is not required in our approach, while it is the case for the path-based hypernymy extraction methods.

5. Finding an Optimal Configuration of Meta Parameters of the Method

The approach consists of several sequential stages, as depicted in Figure 2, with each stage having a few meta parameters. This study is designed to find promising combinations of these meta parameters. In this section, we propose several metrics which aim at finding an optimal configuration of all these meta parameters jointly. In particular, to compare different configurations of our approach, we compare the labeled sense clusters to WordNet 3.1 (Fellbaum, 1998) and BabelNet 3.7 (Navigli and Ponzetto, 2012). The assumption is that the optimal model contains lexical semantic knowledge similar to the knowledge in the lexical resources. To implement the evaluation metrics we used the NLTK library (Bird et al., 2009) and the BabelNet Java API.[5]

5.1. Metrics Quantifying Goodness of Fit of the Induced Structures to Lexical Resources

To summarize various aspects of the lexical resource, we propose a score that is maximized if labeled sense clusters are generated directly from a lexical resource:

$$hpc\text{-score}(c) = \frac{h\text{-score}(c) + 1}{p\text{-score}(c) + 1} \cdot \text{coverage}(c). \quad (5)$$

p-score(c) quantifies the plausibility of the sense cluster c.

[5] `http://www.babelnet.org`

	Min. sense co-occurrences, t	**Edge weight**, E	**Only nouns**, N	**Hypernym weight**, H	**Number of clusters**	**Number of senses**	*hpc*-avg, **WordNet**	*hpc*-avg, **BabelNet**
coarse-gr.	100	log	yes	tf-idf	734	18 028	**0.092**	**0.304**
	100	log	no	tf-idf	763	27 149	0.090	0.303
	100	count	no	tf-idf	765	27 149	0.089	0.302
	100	log	no	tf	784	27 149	0.090	0.300
	100	count	yes	tf	733	18 028	**0.092**	0.299
	100	count	no	tf	772	27 149	0.089	0.297
	100	count	yes	tf-idf	732	18 028	0.091	0.295
	100	log	yes	tf	726	18 028	0.088	0.293
fine-gr.	0	count	no	tf-idf	1870	208 871	**0.041**	**0.279**
	0	count	no	tf	1877	208 871	**0.041**	0.278
	0	count	yes	tf	2070	144 336	0.037	0.240
	0	count	yes	tf-idf	2080	144 336	0.038	0.240
	0	log	yes	tf-idf	4709	144 336	0.027	0.138
	0	log	yes	tf	4679	144 336	0.027	0.136
	0	log	no	tf-idf	5960	208 871	0.035	0.127
	0	log	no	tf	5905	208 871	0.036	0.126

Table 3: Performance of different configurations of the hypernymy labeled global sense clusters in terms of their similarity to WordNet/BabelNet. The results are sorted by performance on BabelNet dataset, the best values in each section are boldfaced. The two underlined configurations are respectively the best *coarse-grained* and *fine-grained* grained semantic class models used in all experiments. The coarse grained model contains less semantic classes, but they tend to be more consistent than those of the fine-grained model, which contains more senses and classes.

It reflects the distance of co-hyponyms in a lexical resource:

$$p\text{-score}(c) = \frac{1}{|c|}\sum_{i=1}^{|c|}\sum_{j=1}^{i}\text{dist}(w_i, w_j). \quad (6)$$

The lower the p-score is, the closer the hyponyms are located in the gold standard resource. For each pair of distinct lemmas (w_i, w_j) in the cluster of co-hyponyms c, we search for the minimal shortest path distance (SPD) between the synsets corresponding to each word in the pair, i.e. $S(w_i)$ is the set of synsets having the w_i lemma and $S(w_j)$ is the similar set with respect to the w_j lemma:

$$\text{dist}(w_i, w_j) = \min_{\substack{s'\in S(w_i),\\ s''\in S(w_j)}} \text{SPD}(s', s''). \quad (7)$$

h-score(c) quantifies plausibility of the hypernyms $\mathcal{H}(c)$ of a sense cluster c measuring the precision of extracted hypernyms:

$$h\text{-score}(c) = \frac{|\mathcal{H}(c) \cap \text{gold}(c)|}{|\mathcal{H}(c)|}. \quad (8)$$

The gold(c) is composed of the lowest common hypernyms (LCH) in the lexical resource for each pair of lemmas in the sense cluster c:

$$\text{gold}(c) = \bigcup_{\substack{w_i\in c,\\ w_j\in c}} \bigcup_{\substack{s'\in S(w_i),\\ s''\in S(w_j)}} \{\text{LCH}(s', s'')\}. \quad (9)$$

coverage(c) quantifies how well cluster words are represented in the gold standard resource. Thus, errors in poorly represented clusters are discounted via coverage. Coverage is the fraction of the lemmas appearing both in the cluster c and in the vocabulary of the resource $\mathcal{V}$:

$$\text{coverage}(c) = \frac{|c \cap \mathcal{V}|}{|c|}. \quad (10)$$

The total score used to rank various configurations of our approach averages *hpc*-score scores for all induced sense clusters:

$$hpc\text{-avg} = \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} hpc\text{-score}(c). \quad (11)$$

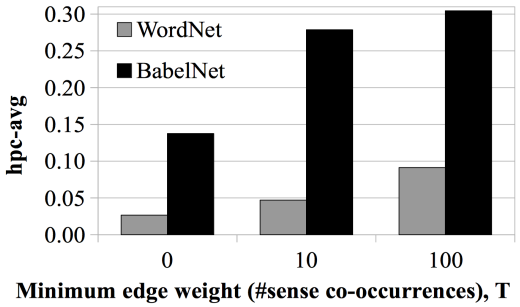


Figure 5: Impact of the min. edge weight t.

5.2. Results

Meta parameter search results based on the comparison to WordNet and BabelNet are provided in Figure 5 and Table 3. The minimal edge weight t trades off between the size of the resulting resource (number of words and senses) and its similarity to the gold lexical resources. The higher the threshold, the fewer nodes remain in the graph, yet these remaining nodes form densely interlinked communities. For t of 100, each pair of senses in the graph is observed in at least 100 ego networks. Secondly, for the unpruned model ($t = 0$), edge weights based on counts worked better than logarithmic weights. However, when pruned ($t > 0$), logarithmic edge weighting shows better results. Thirdly, the tf-idf weights proved to yield consistent improvements over the basic tf weighting. For the pruned model, the variation in scores across different configurations is small as the underlying graphs are of high quality, while for the unpruned model the choice of parameters has

much more impact as the sense graphs are noisier.
We selected the best-performed configuration according to BabelNet (hpc-avg of 0.304), which also is the second best configuration according to WordNet (hpc-avg of 0.092). This model is based on the edge threshold t of 100, logarithmic weights of edges contains only nouns and hypernyms ranked according to tf-idf. Note also that, the best-unpruned model ($t = 0$) has BabelNet hpc-avg of 0.279, which is only 10% lower than the best model, yet the unpruned model has an order of magnitude larger vocabulary and a more fine-grained representation (734 vs. 1,870 clusters). Thus, if coverage is important, the unpruned model is recommended. In the remainder of this paper, we continue with the first model listed in Table 3 and evaluate it in the following experiments.

6. Evaluation

To evaluate our approach, we conduct two intrinsic evaluations and one extrinsic evaluation. The first experiment aims to estimate the fraction of spurious sense clusters, the second one evaluates the quality of the post-processed hypernyms. Finally, we evaluate the induced semantic classes in application to the taxonomy induction task.

6.1. Experiment 1: Plausibility of the Induced Semantic Classes

Comparison to gold standard resources allows us to gauge the relative performances of various configurations of our method. To measure the absolute quality of the best configuration selected in the previous section, we rely on microtask-based crowdsourcing with CrowdFlower[6].

6.1.1. Task Design

We used two crowdsourcing tasks based on word intruder detection (Chang et al., 2009) to measure how humans perceive the extracted lexical-semantic structures. Namely, the tasks are designed to evaluate the quality of the extracted sense clusters and their labels. The input form presented to an annotator is illustrated in Figure 6. A crowdworker is asked to identify words that do not match the context represented by words from a sense cluster or its label. To generate an intruder, following the original design of Chang et al. (2009), we select a random word from a cluster and replace it with a word of similar frequency that does not belong to any cluster (bias here is low as the evaluated model contains 27,149 out of 313,841 induced word senses). In both tasks, the workers have been provided with concise instructions and test questions.

6.1.2. Evaluation Metrics

We compute two metrics on the basis on annotation results: (1) *accuracy* is the fraction of tasks where annotators correctly identified the intruder, thus the words from the cluster are consistent; (2) *badness* is the fraction of tasks for which non-intruder words were selected. In this experiment, we assume that it is easy to identify the intruder in a correct sense cluster and difficult in a noisy, implausible sense cluster. We compute *accuracy* as the fraction of tasks where annotators correctly identified the intruder, thus the words from the cluster are consistent.

[6] https://www.crowdflower.com

Topics:
- vegetable
- fruit
- crop

For these topics we have the list of the following words:
- peach
- pineapple
- winchester
- watermelon
- cherry
- blackberry

Select the words that are non-relevant for the topics above:
- ☐ peach
- ☐ pineapple
- ☐ winchester
- ☐ watermelon
- ☐ cherry
- ☐ blackberry

Figure 6: Layout of the sense cluster evaluation crowdsourcing task, the entry "winchester" is the intruder.

	Accuracy	Badness	Randolph κ
Sense clusters, c	0.859	0.248	0.739
Hyper. labels, $\mathcal{H}(c)$	0.919	0.208	0.705

Table 4: Plausibility of the sense clusters according to human judgments via an intruder detection experiment for the coarse-grained semantic class model.

6.1.3. Results

Table 4 summarizes the results of the intruder detection experiment. Overall, 68 annotators provided 2,035 judgments about the quality of sense clusters. Regarding hypernyms, 98 annotators provided 2,245 judgments. The majority of the induced semantic classes and their labels are highly plausible according to human judgments: the accuracy of the sense clusters based on the intruder detection is 0.859 (agreement of 87%), while the accuracy of hypernyms is 0.919 (agreement of 85%). The Randolph κ of respectively 0.739 and 0.705 indicates substantial inter-observer agreement (Randolph, 2005).
According to the feedback mechanism of the CrowdFlower, the co-hyponymy task received a 4.0 out of 5.0 rating, while the hypernymy task received a 4.4 out of 5.0 rating. The crowdworkers show a *substantial* agreement according to Randolph κ coefficient computed 0.739 for the cluster evaluation task and 0.705 for the hypernym evaluation task.
Major sources of errors for crowdworkers are rare words and entities. While clusters with well-known entities, such as "Richard Nixon" and "Windows Vista" are correctly labeled, examples of other less-known named entities, e.g. cricket players, are sometimes wrongly labeled as implausible. Another source of errors during crowdsourcing were wrongly assigned hypernyms: in rare cases, sense clusters are labeled with hypernyms like "thing" or "object" that are

	Precision	Recall	F-score
Original hypernymy relations extracted from the Common Crawl corpus (Seitner et al., 2016)	0.475	0.546	0.508
Enhanced hypernyms with the *coarse-grained* semantic classes	**0.541**	**0.679**	**0.602**

Table 5: Results of post-processing of a noisy hypernymy database with our approach, evaluated using human judgements.

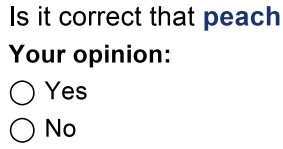


Figure 7: Layout of the hypernymy annotation task.

too generic even under tf-idf weighting.

6.2. Experiment 2: Improving Binary Hypernymy Relations

In this experiment, we test whether our post-processing based on the semantic class improves the quality of hypernymy relations (cf. Figure 2).

6.2.1. Generation of Binary Hypernyms.

We evaluated the best coarse-grained model identified in the first experiment (t of 100). Each sense cluster of this model is split into the set $H_{cluster}$ of binary hypernyms, as illustrated in Figure 1. Overall, we gathered 85,290 hypernym relations for 17,058 unique hyponyms. Next, we gathered the set H_{orig} of 75,486 original hypernyms for exactly the same 17,058 hyponyms. For each word from the sense cluster we looked up top five hypernyms under the best ones when sorting them by extraction frequency from the hypernym relation database of Seitner et al. (2016) as in our model each sense cluster is labeled with five hypernyms from the same database. The database of Seitner et al. (2016) is extracted using lexical patterns. Note that any other method for extraction of binary hypernyms can be used at this point, e.g. (Weeds et al., 2014; Roller et al., 2014; Shwartz et al., 2016; Glavaš and Ponzetto, 2017). For the comparison, we gathered up to five hypernyms for each word, using (1) the most frequent hypernym relations from (Seitner et al., 2016) vs. (2) the cluster labeling method as described above.

6.2.2. Task Design

We drew a random sample of 4,870 relations using lexical split by hyponyms. All relations from $H_{cluster}$ and H_{orig} of one hyponym were included in the sample. These relations were subsequently annotated by human judges using crowdsourcing. We asked crowdworkers to provide a binary judgment about the correctness of each hypernymy relation as illustrated in Figure 7.

6.2.3. Results

Overall, 298 annotators completed 4,870 unique tasks each labeled 6.9 times on average, resulting in a total of 33,719 binary human judgments about hypernyms. We obtained a *fair* agreement among annotators of 0.548 in terms of the Randolph κ (Meyer et al., 2014). Since CrowdFlower reports a confidence for each answer, we selected $N = 3$ most confident answers per pair and aggregated them using weighted majority voting. The ties were broken pessimistically, i.e. by treating a hypernym as irrelevant. Results for $N \in 3, 5, 6$ varied less than by 0.002 in terms of F-score. The task received the rating of a 4.4 out of 5.0 according to the annotator's feedback mechanism.

Table 5 presents results of the experiment. Since each pair received a binary score, we calculated Precision, Recall, and F-measure of two compared methods. Our denoising method improves the quality of the original hypernyms by a large margin both in terms of precision and recall, leading to an overall improvement of 10 F-score points. The improvements of recall are due to the fact that to label a cluster of co-hyponyms it is sufficient to lookup hypernyms for only a fraction of words in the clusters. However, binary relations will be generated between all cluster hypernyms and the cluster words potentially generating hypernyms missing in the input database. For instance, a cluster of fruits can contain common entries like "apple" and "mango" which ensure labeling it with the word "fruit". Rare words in the same cluster, like "mangosteen", which have no hypernyms in the original resource due to the sparsity of the pattern-based approach, will also obtain the hypernym "fruit" as they are distributionally related to frequent words with reliable hypernym relations, cf. Figure 1. We also observed this effect frequently with clusters of named entities, like cricket players. Improvements in precision are due to filtering of wrong extractions, which are different for different words and thus top hypernyms of a cluster contain only hypernyms confirmed by several co-hyponyms.

Finally, note that all previous hypernymy extraction methods output binary relations between undisambiguated words (cf. Section 2.). Therefore, our approach could be used to improve results of other state-of-the-art hypernymy extraction approaches, such as HypeNET (Shwartz et al., 2016).

6.3. Experiment 3: Improving Domain Taxonomy Induction

In this section, we show how the labeled semantic classes can be used for induction of domain taxonomies.

6.3.1. SemEval 2016 Task 13

We use the taxonomy extraction evaluation dataset by Bordea et al. (2016), featuring gold standard taxonomies for three domains (Food, Science, Environment) and four languages (English, Dutch, French, and Italian) on the basis of existing lexical resources, such as WordNet and Eurovoc (Steinberger et al., 2006).[7] Participants were supposed to build a taxonomy provided a vocabulary of a domain. Since our other experiments were conducted on English, we used the English part of the task. The evaluation is

[7] http://eurovoc.europa.eu

System / Domain, Dataset	Food, WordNet	Science, WordNet	Food, Combined	Science, Combined	Science, Eurovoc	Environment, Eurovoc
WordNet	1.0000	1.0000	0.5870	0.5760	0.6243	n.a.
Baseline	0.0022	0.0016	0.0019	0.0163	0.0056	0.0000
JUNLP	0.1925	0.0494	0.2608	0.1774	0.1373	0.0814
NUIG-UNLP	n.a.	0.0027	n.a.	0.0090	0.1517	0.0007
QASSIT	n.a.	0.2255	n.a.	0.5757	0.3893	0.4349
TAXI	0.3260	0.2255	0.2021	0.3634	0.3893	0.2384
USAAR	0.0021	0.0008	0.0000	0.0020	0.0023	0.0007
Semantic Classes (fine-grained)	0.4540	0.4181	0.5147	0.6359	**0.5831**	0.5600
Semantic Classes (coarse-grained)	**0.4774**	**0.5927**	**0.5799**	**0.6539**	0.5515	**0.6326**

Table 6: Comparison of the our taxonomy induction method on the SemEval 2016 Task 13 on Taxonomy Extraction Evaluation (Bordea et al., 2016) for English in terms of cumulative Fowlkes&Mallows measure (F&M).

Domain	#Seeds words	#Expand. words	#Clusters, fine-gr.	#Clusters, coarse-gr.
Food	2 834	3 047	29	21
Science	806	1 137	73	35
Environ.	261	909	111	39

Table 7: Summary of the domain-specific sense clusters.

based on the Fowlkes&Mallows Measure (F&M), a cumulative measure of the similarity of both taxonomies (Velardi et al., 2013).

6.3.2. Taxonomy Induction using Semantic Classes

Our method for taxonomy induction takes as input a vocabulary of the domain and outputs a taxonomy of the domain. The method consists of three steps: (1) retrieving sense clusters relevant to the target domain; (2) generation of binary relations though a Cartesian product of words in a sense cluster and its labels; (3) attaching disconnected components to the root (the name of the domain). We retrieve domain-specific senses for each domain of the SemEval datasets by a lexical filtering. First, we build an extended lexicon of each domain on the basis of the seed vocabulary of the domain provided in the SemEval dataset. Namely, for each seed term, we retrieve all semantically similar terms. To filter out noisy expansions, related terms are added to the expanded vocabulary only if there are at least $k = 5$ common terms between the seed vocabulary and the list of related terms. Second, we retrieve all sense clusters that contain at least one term from the expanded vocabulary among its sense clusters or hypernyms. Table 7 summarizes results of this domain filtering. After, we generate binary hypernymy relations by linking every word in the semantic class to each hypernymy label as shown in Figure 1. Finally, we link roots of each disconnected components to the root of the taxonomy, e.g. "food" for the Food domain. Note that this step was used by SemEval participants, e.g. in the TAXI system (Panchenko et al., 2016).

6.3.3. Results

Table 6 presents results of the taxonomy extraction experiment. We evaluated two best models of our method: a *coarse* and a *fine* grained clusterings featuring respectively 734 and 1870 semantic classes identified in Section 5. with different levels of pruning: $t \in \{0, 100\}$. As one can observe, our model based on the labeled sense clusters significantly outperforms the substring-based baseline and all participating system by a large margin on all domains. For the "Science (Eurovoc)" and "Food" domains our method yields results comparable to WordNet while remaining unsupervised and knowledge-free. Besides, for the "Science" domain our method outperforms WordNet, indicating on the high quality of the extracted lexical semantic knowledge. Overall, the *coarse-grained* more pruned model yielded better results as compared to *fine-grained* un-pruned model for all domains but "Science (Eurovoc)".

7. Conclusion

In this paper, we presented an unsupervised method for the induction of sense-aware semantic classes using distributional semantics and graph clustering and showed how these can be used for post-processing of noisy hypernymy databases extracted from text. We determined optimal parameters of our approach by a comparison to existing lexical-semantic networks. To evaluate our approach, we performed three experiments. A large-scale crowdsourcing study indicated a high plausibility of extracted semantic classes according to human judgment. Besides, we demonstrated that our approach helps to improve precision and recall of a hypernymy extraction method. Finally, we showed how the proposed semantic classes can be used to improve domain taxonomy induction from text.

While we have demonstrated the utility of our approach for hypernym extraction and taxonomy induction, we believe that the induced semantic classes can be useful in other tasks. For instance, in (Panchenko et al., 2017) these semantic classes were used as an inventory for word sense disambiguation to deal with out of vocabulary words.

8. Acknowledgements

This research was supported by Deutscher Akademischer Austauschdienst (DAAD), Deutsche Forschungsgemeinschaft (DFG) under the project "Joining Ontologies and Semantics Induced from Text" (JOIN-T), and by the Ministry of Education and Science of the Russian Federation Agreement no. 02.A03.21.0006. We are grateful to three anonymous reviewers for their helpful comments. Finally, we are grateful to Dirk Johannßen for providing feedback on an early version of this paper.

9. Bibliographical References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). DBpedia: A Nucleus for a Web of Open Data. In *Proceedings of the 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007: The Semantic Web*, pages 722–735. Springer Berlin Heidelberg, Busan, Korea.

Bartunov, S., Kondrashkin, D., Osokin, A., and Vetrov, D. (2016). Breaking sticks and ambiguities with adaptive skip-gram. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 130–138, Cadiz, Spain.

Berant, J., Dagan, I., and Goldberger, J. (2011). Global Learning of Typed Entailment Rules. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 610–619, Portland, Oregon, USA. Association for Computational Linguistics.

Biemann, C. and Riedl, M. (2013). Text: now in 2D! A framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Biemann, C., Faralli, S., Panchenko, A., and Ponzetto, S. P. (2018). A framework for enriching lexical semantic resources with distributional semantics. *Natural Language Engineering*, pages 1–48.

Biemann, C. (2005). Ontology Learning from Text: A Survey of Methods. *GLDV-Journal for Computational Linguistics and Language Technology*, 20(2):75–93.

Biemann, C. (2006). Chinese Whispers: An Efficient Graph Clustering Algorithm and Its Application to Natural Language Processing Problems. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, TextGraphs-1, pages 73–80, New York, NY, USA. Association for Computational Linguistics.

Biemann, C. (2013). Creating a system for lexical substitutions from scratch using crowdsourcing. *Language Resources and Evaluation*, 47(1):97–122.

Bird, S., Loper, E., and Klein, E. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

Bordea, G., Buitelaar, P., Faralli, S., and Navigli, R. (2015). SemEval-2015 Task 17: Taxonomy Extraction Evaluation (TExEval). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 902–910, Denver, CO, USA, June. Association for Computational Linguistics.

Bordea, G., Lefever, E., and Buitelaar, P. (2016). SemEval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *SemEval-2016*, pages 1081–1091, San Diego, CA, USA. Association for Computational Linguistics.

Braslavski, P., Ustalov, D., Mukhin, M., and Kiselev, Y. (2016). YARN: Spinning-in-Progress. In *Proceedings of the 8th Global WordNet Conference*, GWC 2016, pages 58–65, Bucharest, Romania. Global WordNet Association.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 120–126, College Park, MD, USA. Association for Computational Linguistics.

Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., and Blei, D. M. (2009). Reading Tea Leaves: How Humans Interpret Topic Models. In *Advances in Neural Information Processing Systems 22*, pages 288–296. Curran Associates, Inc.

Cimiano, P., (2006). *Ontology Learning from Text*, pages 19–34. Springer US, Boston, MA, USA.

De Smedt, T. and Daelemans, W. (2012). Pattern for Python. *Journal of Machine Learning Research*, 13:2031–2035.

Everett, M. and Borgatti, S. P. (2005). Ego network betweenness. *Social networks*, 27(1):31–38.

Faralli, S., Panchenko, A., Biemann, C., and Ponzetto, S. P. (2016). Linked Disambiguated Distributional Semantic Networks. In *The Semantic Web – ISWC 2016: 15th International Semantic Web Conference, Part II*, Lecture Notes in Computer Science, pages 56–64, Kobe, Japan. Springer International Publishing.

Fellbaum, C. (1998). *WordNet: An Electronic Database*. MIT Press.

Ferraresi, A., Zanchetta, E., Baroni, M., and Bernardini, S. (2008). Introducing and evaluating ukWaC, a very large web-derived corpus of English. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4): Can we beat Google?*, pages 47–54, Marrakech, Morocco. European Language Resources Association (ELRA).

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning Semantic Hierarchies via Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209, Baltimore, MD, USA. Association for Computational Linguistics.

Glavaš, G. and Ponzetto, S. P. (2017). Dual tensor model for detecting asymmetric lexico-semantic relations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1758–1768, Copenhagen, Denmark, September. Association for Computational Linguistics.

Gong, Z., Cheang, C. W., and Leong Hou, U. (2005). Web Query Expansion by WordNet. In *Proceedings of the 16th International Conference on Database and Expert Systems Applications - DEXA '05*, pages 166–175, Copenhagen, Denmark. Springer Berlin Heidelberg.

Graff, D. and Cieri, C. (2003). English Gigaword corpus. *Linguistic Data Consortium*.

Hearst, M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, COLING '92, pages 539–545, Nantes, France. Association for Computational Linguistics.

Heylen, K., Peirsman, Y., Geeraerts, D., and Speelman, D. (2008). Modelling Word Similarity: an Evaluation of Automatic Synonymy Extraction Algorithms. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation*, LREC 2008, pages

3243–3249, Marrakech, Morocco. European Language Resources Association (ELRA).

Kozareva, Z. and Hovy, E. (2013). Tailoring the automated construction of large-scale taxonomies using the web. *Language resources and evaluation*, 47(3):859–890.

Levy, O., Remus, S., Biemann, C., and Dagan, I. (2015). Do Supervised Distributional Methods Really Learn Lexical Inference Relations? In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 970–976, Denver, CO, USA. Association for Computational Linguistics.

Lin, D. and Pantel, P. (2001). Induction of Semantic Classes from Natural Language Text. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 317–322, San Francisco, CA, USA. ACM.

Meyer, C. M., Mieskes, M., Stab, C., and Gurevych, I. (2014). DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 105–109, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Mirkin, S., Dagan, I., and Geffet, M. (2006). Integrating pattern-based and distributional similarity methods for lexical entailment acquisition. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 579–586, Sydney, Australia. Association for Computational Linguistics.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Necsulescu, S., Mendes, S., Jurgens, D., Bel, N., and Navigli, R. (2015). Reading Between the Lines: Overcoming Data Sparsity for Accurate Classification of Lexical Relationships. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 182–192, Denver, CO, USA. Association for Computational Linguistics.

Neelakantan, A., Shankar, J., Passos, A., and McCallum, A. (2014). Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar.

Panchenko, A., Faralli, S., Ruppert, E., Remus, S., Naets, H., Fairon, C., Ponzetto, S. P., and Biemann, C. (2016). TAXI at SemEval-2016 Task 13: a Taxonomy Induction Method based on Lexico-Syntactic Patterns, Substrings and Focused Crawling. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1320–1327, San Diego, California, USA. Association for Computational Linguistics.

Panchenko, A., Marten, F., Ruppert, E., Faralli, S., Ustalov, D., Ponzetto, S. P., and Biemann, C. (2017). Unsupervised, knowledge-free, and interpretable word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 91–96, Copenhagen, Denmark, September. Association for Computational Linguistics.

Pantel, P. and Lin, D. (2002). Discovering Word Senses from Text. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 613–619, Edmonton, AB, Canada. ACM.

Pantel, P. and Pennacchiotti, M. (2006). Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 113–120, Sydney, Australia. Association for Computational Linguistics.

Pantel, P. and Ravichandran, D. (2004). Automatically Labeling Semantic Classes. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL'2004)*, pages 321–328, Boston, MA, USA. Association for Computational Linguistics.

Randolph, J. J. (2005). Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Online submission*.

Richter, M., Quasthoff, U., Hallsteinsdóttir, E., and Biemann, C. (2006). Exploiting the Leipzig Corpora Collection. In *Proceedings of the Fifth Slovenian and First International Language Technologies Conference (IS-LTC)*, Ljubljana, Slovenia. European Language Resources Association (ELRA).

Roller, S., Erk, K., and Boleda, G. (2014). Inclusive yet Selective: Supervised Distributional Hypernymy Detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1025–1036, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Seitner, J., Bizer, C., Eckert, K., Faralli, S., Meusel, R., Paulheim, H., and Ponzetto, S. P. (2016). A Large DataBase of Hypernymy Relations Extracted from the Web. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, LREC 2016, pages 360–367, Portorož, Slovenia. European Language Resources Association (ELRA).

Shi, L. and Mihalcea, R. (2005). Putting Pieces Together: Combining FrameNet, VerbNet and WordNet for Robust Semantic Parsing. In *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing 2005, pages 100–111, Mexico City, Mexico. Springer Berlin Heidelberg.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving Hypernymy Detection with an Integrated Path-based and Distributional Method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany, August. Association for Computational Linguistics.

Snow, R., Jurafsky, D., and Ng, A. Y. (2004). Learning Syntactic Patterns for Automatic Hypernym Discovery.

In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, NIPS'04, pages 1297–1304, Vancouver, BC, Canada. MIT Press.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufis, D., and Varga, D. (2006). The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. *arXiv preprint cs/0609058*.

Steyvers, M. and Tenenbaum, J. B. (2005). The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth. *Cognitive science*, 29(1):41–78.

Tjong Kim Sang, E. (2007). Extracting Hypernym Pairs from the Web. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 165–168, Prague, Czech Republic. Association for Computational Linguistics.

Ustalov, D., Arefyev, N., Biemann, C., and Panchenko, A. (2017a). Negative sampling improves hypernymy extraction based on projection learning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 543–550, Valencia, Spain. Association for Computational Linguistics.

Ustalov, D., Panchenko, A., and Biemann, C. (2017b). Watset: Automatic induction of synsets from a graph of synonyms. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1579–1590, Vancouver, Canada. Association for Computational Linguistics.

Velardi, P., Faralli, S., and Navigli, R. (2013). OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction. *Computational Linguistics*, 39(3):665–707.

Vylomova, E., Rimell, L., Cohn, T., and Baldwin, T. (2016). Take and Took, Gaggle and Goose, Book and Read: Evaluating the Utility of Vector Differences for Lexical Relation Learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1671–1682, Berlin, Germany. Association for Computational Linguistics.

Wandmacher, T. (2005). How semantic is Latent Semantic Analysis? In *Proceedings of RÉCITAL 2005*, pages 525–534, Dourdan, France.

Weeds, J., Clarke, D., Reffin, J., Weir, D. J., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Widdows, D. and Dorow, B. (2002). A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Zhou, G., Liu, Y., Liu, F., Zeng, D., and Zhao, J. (2013). Improving question retrieval in community question answering using world knowledge. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 2239–2245, Beijing, China. AAAI Press.

Laying the Groundwork for Knowledge Base Population: Nine Years of Linguistic Resources for TAC KBP

Jeremy Getman, Joe Ellis, Stephanie Strassel, Zhiyi Song, Jennifer Tracey
Linguistic Data Consortium, University of Pennsylvania
3600 Market Street, Suite 810, Philadelphia, PA 19104
{jgetman, joellis, strassel, zhiyi, garjen}@ldc.upenn.edu

Abstract

Knowledge Base Population (KBP) is an evaluation series within the Text Analysis Conference (TAC) evaluation campaign conducted by the National Institute of Standards and Technology (NIST). Over the past nine years TAC KBP evaluations have targeted information extraction technologies for the population of knowledge bases comprised of entities, relations, and events. Linguistic Data Consortium (LDC) has supported TAC KBP since 2009, developing, maintaining, and distributing linguistic resources in three languages for seven distinct evaluation tracks. This paper describes LDC's resource creation efforts for the various KBP tracks, and highlights changes made over the years to support evolving evaluation requirements.

Keywords: knowledge base population, information extraction, linguistic resources

1. Introduction

Text Analysis Conference (TAC) an annual series of open technology evaluations organized by the National Institute of Standards and Technology (NIST). The Knowledge Base Population (KBP) evaluation track (McNamee et al. 2010) encourages the development of systems that can extract information from unstructured multilingual text and in order to populate an existing or emergent knowledge base. Since the start of TAC KBP in 2009 Linguistic Data Consortium (LDC) has provided linguistic resources by building labeled training and test sets and by assessing system results.

Each year's KBP evaluation comprises a number of component evaluation tasks. Over time the KBP evaluations, and by extension the data to support those evaluations, has moved in the direction of greater complexity, with more integration of individual evaluation tracks and data sets, a greater emphasis on multilingual data for all tracks, and an increasing focus on events (in addition to entities and slots). These developments culminated in 2017 with an expanded end-to-end Cold Start track that evaluated systems' ability to combine the other KBP tracks and extract all entities, relations, sentiments, and events from a document collection, perform corpus-wide clustering of the extracted items, and build a structured knowledge base from scratch (NIST, 2017).

In the sections that follow we describe the KBP corpora developed by LDC for each of the seven primary evaluation tracks, including source data, knowledge bases and annotation and assessment efforts between 2009-2017. TAC KBP will continue in 2018 with new evaluation tracks.

2. Source Data and Knowledge Bases

Source data for the KBP evaluations consists of English, Spanish and Chinese text in two primary genres: formal newswire (NW) and informal web text (primarily blogs and discussion forums) drawn from existing LDC collections as well as newly collected material. In 2009-2015 we selected a separate set of documents for each evaluation track. Starting in 2016, the consolidation of evaluations into a smaller number of more complex and challenging tasks led to the use of a single set of test data shared by all tasks. This put additional demands on data selection given the need for a single corpus to support multiple, sometimes mutually contradictory, requirements. For instance, the list of required features for the 2016 source data included all of the following:

- Roughly equal representation of all three languages (ENG, SPA, CMN)
- Roughly equal representation of both genres (formal and informal)
- Corpus should cover a relatively short time span
- 800 tokens max per document excluding quote regions for all 90K documents
- Discussion forum documents will start from the beginning of the thread
- Each event sub/type in the ontology must be present in each genre/language
- Each event type and subtype in the ontology must have at least one mention in each of 30 or more documents in each language
- Cross-lingual event hoppers in the corpus, for at least half of the event types in the ontology, made up of 2-3 event mentions
- 50 instances of relatively simple, non-confusable events that are mentioned in 3 or more documents
- 10 or more event hoppers with mentions in 10 or more documents
- Presence of only unnamed mentions for some specific, individual entities in some documents
- Presence of synonymous entities (entities referred to by more than one non-matching string in the corpus)
- Presence of polysemous entities (distinct entities referred to by equivalent strings in the corpus)

Table 1 below summarizes KBP source data provided by LDC for each year's evaluations.

For every new evaluation track introduced to KBP, LDC produced a small amount of labeled training data to illustrate the annotation approach and data properties for the track. After this initial training set no new labeled training data was produced; instead, labeled test sets from prior years were released to evaluation participants for use as training and development data. LDC also made related

resources produced under other efforts available to KBP participants to use as supplemental training data.

Year	Genre	ENG	CMN	SPA
2009	formal	1,286,609	-	-
	informal	3,040	-	-
2010	formal	1,287,292	-	-
	informal	490,596	-	-
2011	formal	1,287,292	1,000,000	-
	informal	490,596	-	-
2012	formal	2,287,549	2,000,256	1,000,020
	informal	1,490,595	815,886	-
2013	formal	1,000,257	2,000,256	910,734
	informal	1,099,062	1,015,027	-
2014	formal	1,000,562	2,000,256	910,734
	informal	1,099,423	1,015,027	649,095
2015	formal	9,270	84	84
	informal	40,521	82	83
2016	formal	15,000	15,000	15,000
	informal	15,000	15,000	15,000
2017	formal	15,000	15,000	15,000
	informal	15,000	15,000	15,000

Table 1: TAC KBP evaluation source documents

In addition to creating the training and test corpora required for each year's evaluation tracks, LDC also produced the knowledge bases used for evaluation. KBP has used two distinct KBs since its inception. Prior to the first KBP evaluation in 2009, LDC created a KB based on a Wikipedia snapshot, extracting information from page titles, infoboxes and article text from over 800,000 entries (Simpson et. al., 2010). Each node in the reference KB corresponds to a Wikipedia page for a person, organization, or geopolitical entity and consists of predefined attributes derived from infoboxes (Linguistic Data Consortium, 2014). In 2015 a decision was made by evaluation coordinators to instead use BaseKB (http://basekb.com), which is a subset of Freebase represented in RDF (Linguistic Data Consortium, 2015). Because of the additional complexity of BaseKB compared to the Wikipedia-based KB, LDC prepared a custom human-readable version for use during annotation and assessment.

3. Evolution of Data for KBP Evaluation Tracks: 2009-2017

3.1 Entity Linking

Entity Linking (EL) requires systems to link named mentions of person, organization, and geopolitical entities in text to entries in a knowledge base (KB), report if no matching entries exist, and group mentions without entries according to identity coreference. Entity Linking began in 2009 in English (Simpson et al., 2010), added a Chinese cross-lingual version of the task in 2011, and further expanded with a Spanish version of the task in 2012 (Ellis et al., 2012; Li et al., 2012). The 2013 EL task remained largely unchanged from the tri-lingual version established in 2012. In 2014, only cross-lingual Chinese and Spanish EL evaluations were held, as English was replaced by Entity Discovery & Linking.

Although the languages in Entity Linking changed over the track's six years, the goals of query selection for Entity Linking did not (Ellis et al., 2014). Annotators sought to collect the most confusable named entity mentions they could find in the corpus. A query's confusability was measured both by the number of distinct entities in the set of queries that are referred to by its namestring (polysemy) as well as the number of distinct namestrings in the pool that refer to the entity (synonymy). Entity Linking queries were selected with the intention of representing as evenly as possible the three entity types, as well as the ratio of entities in the KB to those not. For cross-lingual EL, English entity mentions co-referential with non-English queries were selected when possible.

The following table summarizes Entity Linking resources developed by LDC.

Year	Source Documents			Queries		
	ENG	CMN	SPA	ENG	CMN	SPA
2009	3688	-	-	3904	-	-
2010	3684	-	-	3750	-	-
2011	2231	4329	-	2250	4347	-
2012	2016	2271	3772	2226	2280	3890
2013	1820	2143	1832	2190	2155	2117
2014	-	2860	2207	-	3253	2596

Table 2: Entity Linking resources

3.2 Entity Discovery & Linking

A new variant of Entity Linking, named Entity Discovery & Linking (EDL), was performed for the first time in 2014 in English. The goal of EDL is full entity extraction from a collection of documents, followed by linking entities to a KB and clustering any entities not in the KB. EDL differed from Entity Linking in that systems and annotators exhaustively annotated documents to create a gold standard for system scoring, instead of cherry-picking ambiguous entities from the corpus (Ellis et al., 2014). While the 2014 task dealt only with named mentions of person, organization, and geopolitical entities, locations and facilities were also annotated in subsequent years (2015-2017), as were nominal mentions. EDL was expanded from English-only to English, Chinese, and Spanish starting in 2015.

In 2014-2015, EDL gold standard annotation required identification and classification of all valid mentions of the targeted entity types within the source corpus (Ellis et al., 2015). Titles were also annotated in 2015 to help systems distinguish between titles and nominal mentions of persons (e.g. "president"). In 2016-2017, rather than starting with a blank slate, annotators worked with entity mentions from Entities, Relations, and Events (ERE), an annotation task developed by LDC for DARPA's Deep Exploration and Filtering of Text program (DEFT) (DARPA, 2012). ERE exhaustively labels entities, relations and events, along with their attributes, according to specified taxonomies (Song et al., 2015). In EDL, ERE entity annotations were displayed in the context of their source documents, so annotators could check for errors and misses, as well as ERE annotations at variance with EDL guidelines (though correct for ERE). In all years,

labeled entities were then linked to the KB or marked as NIL (not in the KB) or Unknown (insufficient information in the source data to know if an entity was in the KB). After document-level KB linking, senior annotators then performed cross-document, cross-language NIL clustering, aided by English descriptions of non-English entities.

The following table summarizes Entity Discovery & Linking resources developed by LDC.

Year	Task	Source Documents	Queries		
			ENG	CMN	SPA
2014	training	138	5598	-	-
2014	eval	160	6349	-	-
2015	pilot	15	200	266	220
2015	training	444	13545	13116	4177
2015	eval	500	15645	11066	5822
2016	eval	505	9231	8845	6964
2017	eval	500	6915	10246	7212

Table 3: Entity Discovery & Linking resources

3.3 Slot Filling

The regular Slot Filling (SF) task involves mining information about entities from text. Systems and LDC annotators search a corpus for information about persons and organizations and add new information to an existing knowledge base. LDC produced data in support of English Slot Filling from 2009 through 2014. In 2014, LDC also produced data for a Chinese Slot Filling pilot evaluation. In addition to regular Slot Filling, LDC also produced data in support of Surprise Slot Filling, in 2010, Temporal Slot Filling, in 2011 and 2013 (Ellis et al., 2013), and Sentiment Slot Filling, in 2013 and 2014. The Surprise and Sentiment SF tasks followed the same guidelines as regular SF with the exception that the slots evaluated targeted novel types of information. Surprise SF added four slots in the same vein as regular SF slots, and Sentiment SF sought to extract positive and negative sentiment held by entities toward other entities, including geopolitical entities (unlike regular SF). Temporal SF utilized the regular SF slots, and involved annotation of temporal information indicating when a given SF relation held true.

Entities (the basis of SF queries), were selected for their non-confusability and productivity. An entity was considered non-confusable if there existed one or more canonical references to it in the source corpus. Productivity was determined by searching the source corpus to find whether at least two answers existed for the entity. After 2009, LDC also developed manual runs, the set of valid human-produced responses to each of the SF queries. From 2012, responses included a justification (the minimum text extract from the source corpus supporting the validity of a response). Valid justifications included all three elements of a relation: its subject entity, slot type, and answer. During assessment, annotators judged the validity of human and system responses, and grouped instances of the same response. Answers were marked correct if they adhered to the slot definitions and were supported in the text. For the 2009-2013 evaluations, attributes in the KB were mapped to the set of SF queries before assessment, thereby indicating returned responses' redundancy with the KB.

The following tables summarize Slot Filling resources developed by LDC.

Year	Task	Lang.	Queries	LDC Responses	Assessed Responses
2009	eval	ENG	53	-	10416
2010	training	ENG	98	336	-
2010	eval	ENG	100	799	24515
2011	training	ENG	198	1627	-
2011	eval	ENG	100	796	28041
2012	eval	ENG	80	1553	22885
2013	eval	ENG	100	2383	27655
2014	eval	ENG	100	2216	21956
2014	training	CMN	32	967	-
2014	eval	CMN	103	2858	2878

Table 4: Regular Slot Filling resources

Year	Task	Lang.	Queries	LDC Responses	Assessed Responses
2010	training	ENG	32	83	-
2010	eval	ENG	40	252	996

Table 5: Surprise Slot Filling resources

Year	Task	Lang.	Queries	LDC Responses	Assessed Responses
2013	training	ENG	163	986	-
2013	eval	ENG	160	977	5160
2013	dual	ENG	-	-	1145
2014	eval	ENG	400	594	6383

Table 6: Sentiment Slot Filling resources

Year	Task	Lang.	Queries	LDC Responses	Assessed Responses
2011	training	ENG	50	1258	-
2011	eval	ENG	100	1413	-
2013	training	ENG	7	16	-
2013	eval	ENG	271	1519	2035

Table 7: Temporal Slot Filling resources

3.4 Event Argument Linking

The Event Argument Linking (EAL) task requires systems and annotators to extract event arguments (entities or attributes playing a role in an event), indicate their role, link the arguments involved in the same event, and format the information in a manner suitable as input to a knowledge base. LDC produced data in support of EAL in each of its four years, from 2014-2017. In 2014 and 2015, the EAL evaluation used an assessment paradigm; 2016 and 2017 instead used a gold standard. In the 2014 EAL evaluation, annotators marked one mention of each valid, unique event argument within the EAL source corpus. In 2015, the task was expanded to include the linking of related event arguments; annotators marked each unique

event argument and clustered related arguments into event hoppers.

For 2014 assessment, annotators performed coreference on all responses to a document, then judged the parts of each response – the event type, the role a response played in the event, and the entity filling that role. Annotators also indicated each response's modality as well as its mention type (name or nominal). In 2015, a final step was added, wherein annotators grouped responses into event hoppers (Song et al., 2015) to indicate a type of event coreference. In 2016, instead of an assessment paradigm, LDC created a set of gold standard EAL annotations against which system submissions were scored (Ellis et al., 2016). The gold standard was an expanded version of ERE data augmented by a script developed by the EAL evaluation track coordinators. Annotators reviewed the results of this augmentation, which added inferred arguments invalid for ERE (but not EAL) and/or difficult for annotators to find. The same approach was taken in 2017, except that instead of reviewing automatically generated augmentations, annotators performed a fully manual augmentation pass, which increased the number of augmented event arguments, as compared with 2016. An English cross-document task was also added in 2016 only, in which LDC selected queries, produced responses, and assessed human and system responses. Annotators selected queries comprised of single event arguments each indicating an event hopper in the EAL gold standard, and searched for all responses to those queries. During assessment, annotators decided if a response's justification proved that a document contained an instance of the corresponding query event.

The following table summarizes Event Argument Linking resources developed by LDC.

Year	Task	Lang.	Source Docs	LDC Responses	Assessed Responses
2014	pilot	ENG	60	-	32054
2014	eval	ENG	528	5947	57599
2015	training	ENG	55	-	-
2015	eval	ENG	500	5207	45391
2016	gold standard eval	ENG, CMN, SPA	505	17809	-
2016	x-doc pilot	ENG	2092	98	2689
2016	x-doc eval	ENG	30000	700	7697
2017	gold standard eval	ENG, CMN, SPA	500	27109	-

Table 8: Event Argument Linking resources

3.5 Event Nugget

The Event Nugget track seeks to evaluate system performance in detection and coreference of event references in text (Mitamura et al., 2015). An event 'nugget', as defined by the task, includes a text extent, a classification of event type and subtypes, and an indication of whether realis mood was used to describe the event (Ellis et al., 2015). Event Nugget started as a pilot evaluation within the DEFT program in 2014. In 2015, event nuggets were redefined to align with the treatment of events in DEFT Rich ERE (Song et al., 2015). Also in 2015, coreference of event nuggets was added, using the definition of event hoppers developed in Rich ERE. In 2016 and 2017, there was no separate annotation task conducted solely to support the Event Nugget evaluations; the data were entirely produced by running a script over ERE data to extract and reformat a subset for use by Event Nugget. Additionally, Chinese, Spanish, and English source documents were used as inputs in 2016 and 2017, whereas the task had been English-only in previous iterations.

Year	Task	Lang.	Source Docs	Event Nuggets	Event Hoppers
2014	training	ENG	151	3782	-
2014	eval	ENG	200	6921	-
2015	training	ENG	446	12301	7481
2015	eval	ENG	202	6438	4125
2016	eval	ENG, CMN, SPA	500	9042	6799
2017	eval	ENG, CMN, SPA	500	11687	8039

Table 9: Event Nugget resources

3.6 Belief and Sentiment

Belief and Sentiment (BeSt), part of TAC KBP in 2016 and 2017, emerged as a task from DARPA's DEFT program, with the goal of augmenting information about entities, relations, and events in a knowledge base with beliefs and sentiment (Ellis et al., 2016). BeSt requires that belief and sentiment be annotated with respect to entities, relations, and events as annotated in ERE. Entities can be holders/reporters of belief and sentiment, as well as targets of sentiment; relations and events can be the targets of belief and/or sentiment. BeSt annotation also labels an entity's role in an event as a target of belief, separate from belief in the event itself. Input to the BeSt annotation task is an ERE-annotated document. A single annotator performs two passes over the list of ERE annotations: one for belief, and one for sentiment. Belief annotation marks the belief-holder's commitment to a belief in the occurrence of an event (event-target), the participation of an entity in an annotated event (entity-target), and/or the existence of a relation (relation-target). In addition to the target and belief-type, the holder of the belief is explicitly indicated, as is the polarity of the belief. Positive and negative sentiment is annotated with entities, relations, and events as targets, and, as in Belief annotation, the holder of the sentiment is indicated.

Table 10 below summarizes BeSt resources developed by LDC.

Year	Task	Lang.	Source Docs	Belief Annotations	Sentiment Annotations
2016	eval	ENG, CMN, SPA	494	45897	61693
2017	eval	ENG, CMN, SPA	500	54412	65753

Table 10: Belief and Sentiment resources

3.7 Cold Start

Cold Start, part of TAC KBP from 2012 through 2017, is designed to evaluate a system's ability to construct a new knowledge base from the information provided in a text collection, by combining technologies developed via other KBP tracks. Like Slot Filling, Cold Start involves mining information about entities from text, and as in Entity Discovery & Linking, Cold Start systems must also find all entities mentioned in the text (Ji et al., 2016). From 2012 through 2016, Cold Start focused on person, organization, and geopolitical entities; facilities and locations were added in 2017. From 2012 to 2015, Cold Start was English-only, but following a Spanish-English pilot in 2016 the track expanded to English, Chinese, and Spanish for 2016 and 2017. On top of regular Slot Filling slots, the focus of Cold Start since 2012, the 2017 task also incorporated the Sentiment SF slots, as well as a new set of event-focused slots, derived from Event Argument Linking, that sought to extract the events in which entities were involved (Getman et al., 2017).

In Cold Start query development, annotators created queries defined by an entity initiating a chain of relations. Unlike Slot Filling, which generates single binary relations, Cold Start strings multiple relations together, so that the object of one relation becomes the subject of another. An example query could be phrased: "Find all shareholders of organizations at which Jane Doe has been an employee." After developing queries, annotators searched for all responses to those queries that could be found in the source data. Responses included justification, extents of text proving the validity of a response. Through 2015, Cold Start queries and responses were developed concurrently, such that annotators could switch between investigating candidate queries and annotating responses. In 2016 and 2017, however, query and response development were separated, as a result of the addition of Spanish and Chinese, which meant a single annotator could no longer annotate all responses to a query, as each query required searching for answers in three languages. In assessment, annotators judged human and system responses. Cold Start followed the same paradigm as that in Slot Filling assessment. However, unlike Slot Filling, because Cold Start queries involved chains of slots, Cold Start assessment necessarily happened in multiple stages. The first stage mirrored Slot Filling assessment, but the second stage involved assessment of slots branching off of answers marked correct in the first stage.

The following table summarizes Cold Start resources developed by LDC.

Year	Task	Lang.	Queries	LDC Responses	Assessed Responses
2012	eval	ENG	385	979	5015
2013	eval	ENG	326	1627	6745
2014	eval	ENG	247	1386	7254
2015	eval	ENG	2539	2218	30654
2016	pilot	SPA, ENG	2118	1238	818
2016	eval	ENG, CMN, SPA	1077	4739	25416
2017	eval	ENG, CMN, SPA	1392	3495	26802

Table 11: Cold Start resources

4. Discussion and Conclusions

We have described LDC's resource creation efforts in support of TAC KBP evaluations since 2009, highlighting changes in our approach required to meet new requirements within and across KBP tasks. In considering the evolution of KBP and its data over time, one strong trend has been the consolidation of simpler monolingual tasks (like English Entity Linking into more challenging multilingual tasks (like Trilingual Entity Discovery and Linking and Cold Start). In part this evolution reflects the maturation of KBP technology, but it also highlights the fact that starting in 2012 KBP also served as the primary framework for evaluating system capabilities in the DARPA Deep Exploration and Filtering of Text (DEFT) program (DARPA, 2012). The goal of the DEFT program is to develop technologies capable of extracting knowledge from unstructured text in multiple languages and genres. DEFT's growing focus on multi-lingual technologies and whole-corpus (as opposed to sentence- or document-level) understanding is reflected in the evolution of KBP tracks over the past 5 years. The table below presents a summary of KBP tracks and their languages between 2009-2017.

	2009	2010	2011	2012	2013	2014	2015	2016	2017
Entity Linking	E	E	E,C	E,C,S	E,C,S	C,S	n/a	n/a	n/a
Entity Discovery & Linking	n/a	n/a	n/a	n/a	n/a	E	E,C,S	E,C,S	E,C,S
Regular Slot Filling	E	E	E	E	E	E,C	n/a	n/a	n/a
Surprise Slot Filling	n/a	E	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Sentiment Slot Filling	n/a	n/a	n/a	n/a	E	E	n/a	n/a	n/a
Temporal Slot Filling	n/a	n/a	E	n/a	E	n/a	n/a	n/a	n/a
Event Argument Linking	n/a	n/a	n/a	n/a	n/a	E	E	E,C,S	E,C,S
Event Nugget	n/a	n/a	n/a	n/a	n/a	E	E	E,C,S	E,C,S
BeSt	n/a	n/a	n/a	n/a	n/a	n/a	n/a	E,C,S	E,C,S
Cold Start	n/a	n/a	n/a	E	E	E	E	E,C,S	E,C,S

Table 12: Increasing complexity and multilinguality in KBP over time

As illustrated here, KBP data introduced greater complexity over time, with more integration of distinct evaluation tracks and data sets, a greater emphasis on multilingual data for all tracks, and an increasing focus on events (as well as entities and slots). In 2013 we expanded Slot Filling to include data both Sentiment Slot Filling and Temporal Slot Filling. In 2014 we added data for two

event-focused tracks Event Argument Linking and Event Nugget, and expanded Entity Linking annotation to cross-document entity extraction and clustering for English (EDL); Chinese and Spanish EDL followed in 2015, which meant cross-document clustering was necessarily also cross-lingual. In 2016 we made the move to using the same source corpus for all KBP evaluation tracks, and we expanded Cold Start from monolingual English to cross-lingual English/Chinese/Spanish; we also added new information about entities, relations, and events with beliefs and sentiment. In 2017, Cold Start was further expanded to include event and sentiment slots, making Cold Start very nearly the sum total of all component KBP evaluations, testing extraction and clustering of entities, relations, events, and sentiment. In all LDC has produced over 150 distinct KBP corpora, comprising over to 150,000 queries, 84,000 manual runs and 310,000 system assessments.

After the conclusion of an evaluation series, resources are consolidated into one or more comprehensive packages and released into LDC's public catalog, making them generally available for language-related research, education and technology development. To date LDC has published several KBP corpora including the pre-2015 knowledge base (LDC2014T16); training and evaluation data for cross-lingual entity linking in Spanish (LDC2016T26) and Chinese (LDC2017T17); and the source data used in all English evaluations between 2009-2014 (LDC2018T03). An additional 15-20 KBP corpora will be published in the catalog in the coming months and years. The TAC KBP evaluation series will continue into 2018 with the introduction of new tracks.

5. Bibliographical References

Dang, H.T., Lin, J., and Kelly, D. 2006. Overview of the TREC 2006 Question Answering Track. In Fifteenth Text Retrieval Conference (TREC 2006) Proceedings, Gaithersburg, MD.

DARPA. 2012. *Broad Agency Announcement: Deep Exploration and Filtering of Text (DEFT)*. Defense Advanced Research Projects Agency, DARPA-BAA-12-47.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. 2004. Automatic Content Extraction (ACE) program - task definitions and performance measures. In Proceedings of the Fourth International Language Resources and Evaluation Conference, Lisbon, Portugal.

Ellis, J., Li, X., Griffitt, K., Strassel, S., and Wright, J. 2012. Linguistic Resources for 2012 Knowledge Base Population Evaluations. TAC KBP Workshop 2012: National Institute of Standards and Technology, Gaithersburg, MD.

Ellis, J., Getman, J., Mott, J., Li, X., Griffitt, K., and Strassel, S. 2013. Linguistic Resources for 2013 Knowledge Base Population Evaluations. TAC KBP Workshop 2013: National Institute of Standards and Technology, Gaithersburg, MD.

Ellis, J., Getman, J., and Strassel, S. 2014. Overview of Linguistic Resources for the TAC KBP 2014 Evaluations: Planning, Execution, and Results. TAC KBP Workshop 2014: National Institute of Standards and Technology, Gaithersburg, MD.

Ellis, J., Getman, J., Fore, D., Kuster, N., Song, Z., Bies, A., and Strassel, S. 2015. Overview of Linguistic Resources for the TAC KBP 2015 Evaluations: Methodologies and Results. TAC KBP Workshop 2015: National Institute of Standards and Technology, Gaithersburg, MD.

Ellis, J., Getman, J., Kuster, N., Song, Z., Bies, A., and Strassel, S. 2016. Overview of Linguistic Resources for the TAC KBP 2016 Evaluations: Methodologies and Results. TAC KBP Workshop 2016: National Institute of Standards and Technology, Gaithersburg, MD.

Getman, J., Ellis, J., Song, Z., Tracey, J., and Strassel, S. 2017. Overview of Linguistic Resources for the TAC KBP 2017 Evaluations: Methodologies and Results. TAC KBP 2017 Workshop: National Institute of Standards and Technology, Gaithersburg, MD.

Ji, H., Nothman, J., Dang, H. T., & Hub, S. I. (2016). Overview of TAC-KBP2016 Tri-lingual EDL and its impact on end-to-end Cold-Start KBP. Proceedings of TAC 2016.

Li, X., Strassel, S. M., Ji, H., Griffitt, K., & Ellis, J. (2012). Linguistic resources for entity linking evaluation: from monolingual to cross-lingual. Annotation. In Proceedings of LREC 2012.

Mitamura, T., Yamakawa, Y., Holm, S., Song, Z., Bies, A., Kulick, S., & Strassel, S. (2015). Event nugget annotation: Processes and issues. In Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation (pp. 66-76).

McNamee, P., Dang, H.T., Simpson, H., Schone, P., and Strassel, S. (2010). An Evaluation of Technologies for Knowledge Base Population. In Proceedings of the Seventh International Language Resources and Evaluation Conference (LREC), Valleta, Malta.

NIST. (2017). TAC 2017 Cold Start KB Track. https://tac.nist.gov/2017/KBP/ColdStart/index.html. May 2017. Accessed September, 2017.

Simpson, H., Strassel, S., Parker, R., and McNamee, P. 2010. Wikipedia and the Web of Confusable Entities: Experience from Entity Linking Query Creation for TAC 2009 Knowledge Base Population. In Proceedings of the Seventh International Language Resources and Evaluation Conference (LREC), Valleta, Malta.

Song, Z., Bies, A., Riese, T., Mott, J., Wright, J., Kulick, S., Ryant, N., Strassel, S., and Ma, X. 2015. From Light to Rich ERE: Annotation of Entities, Relations, and Events. 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation, at the 2015 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015).

Song Z, Bies A, Strassel S, Ellis J, Mitamura T, Dang HT, Yamakawa Y, Holm S. (2016). Event nugget and event coreference annotation. In Proceedings of the The 4th Workshop on EVENTS: Definition, Detection, Coreference, and Representation (pp. 37-45).

Walker, C., Strassel, S., Medero, J., and Maeda, K. 2006. ACE 2005 Multilingual Training Corpus. Linguistic Data Consortium, LDC Catalog No.: LDC2006T06.

6. Language Resource References

Simpson, Heather, et al. TAC KBP Reference Knowledge Base LDC2014T16. Web Download. Philadelphia: Linguistic Data Consortium, 2014.

Ellis, Joe, Jeremy Getman, and Stephanie Strassel. TAC KBP Spanish Cross-lingual Entity Linking - Comprehensive Training and Evaluation Data 2012-2014 LDC2016T26. Web Download. Philadelphia: Linguistic Data Consortium, 2016.

Ellis, Joe, Jeremy Getman, and Stephanie Strassel. TAC KBP Chinese Cross-lingual Entity Linking - Comprehensive Training and Evaluation Data 2011-2014 LDC2017T17. Web Download. Philadelphia: Linguistic Data Consortium, 2017.

Ellis, Joe, et al. TAC KBP Comprehensive English Source Corpora 2009-2014 LDC2018T03. Web Download. Philadelphia: Linguistic Data Consortium, 2018.

A Dataset for Inter-Sentence Relation Extraction using Distant Supervision

Angrosh Mandya, Danushka Bollegala, Frans Coenen, Katie Atkinson
Department of Computer Science, University of Liverpool, UK
{angrosh,danushka,coenen,katie}@liverpool.ac.uk

Abstract

This paper presents a benchmark dataset for the task of inter-sentence relation extraction. The paper explains the distant supervision method followed for creating the dataset for inter-sentence relation extraction, involving relations previously used for standard intra-sentence relation extraction task. The study evaluates baseline models such as bag-of-words and sequence based recurrent neural network models on the developed dataset and shows that recurrent neural network models are more useful for the task of intra-sentence relation extraction. Comparing the results of the present work on iner-sentence relation extraction with previous work on intra-sentence relation extraction, the study suggests the need for more sophisticated models to handle long-range information between entities across sentences.

Keywords: Inter-sentence Relation Extraction, Relation Extraction, Inter-sentence Relation Extraction Dataset, Distant Supervision for Inter-sentence Relation Extraction

1. Introduction

In recent times, the field of relation extraction has received significant research attention due to its importance in information retrieval (Culotta and Sorensen, 2004; Mintz et al., 2009; Banko et al., 2007; Etzioni et al., 2011). The key task in relation extraction is to recognise the semantic relation that exist between two given entities. Depending on the scope of the co-occurrences of the two entities, relation extraction methods can be broadly categorised into two groups: (a) *intra-sentence relation extraction* methods (Brin, 1998; Banko et al., 2007; Mintz et al., 2009; Riedel et al., 2010), and (b) *inter-sentence relation extraction* methods (Swampillai and Stevenson, 2010; Gu et al., 2017; Quirk and Poon, 2016; Peng et al., 2017). While intra-sentence relation extraction attempts to extract relations between two entities that co-occur within the same sentence, inter-sentence relation extraction methods consider entities that might not necessarily co-occur in the same sentence.

In more detail, the distinction between intra and inter-sentence relation extraction tasks can be illustrated as follows. Let us assume that a relation r takes e_1 as the first argument and e_2 as the second argument. Further, let us also assume that e_1 is included in a sentence s_i and e_2 is included in a sentence s_j. Then, we define intra-sentence relation extraction as the task of extracting relation r from s_i and s_j, when $i = j$. We define inter-sentence relation extraction as the task of extracting relation r from s_i and s_j, when $i \neq j$. In this work, we limit i and j such that $|i - j| = 1$. Examples of intra-sentence and inter-sentence relation extraction are provided below in Listings 1 and 2, respectively.

LISTING 1: EXAMPLE OF INTRA-SENTENCE RELATION EXTRACTION

> S1: In 1957, <e1>**Ayn Rand**</e1> published her best-known work, the novel <e2>**Atlas Shrugged**</e2>.

LISTING 2: EXAMPLE OF INTER-SENTENCE RELATION EXTRACTION

> S1: <e1>**Ayn Rand**</e1> (born <e1>**Alisa Zinovyevna Rosenbaum**</e1>, March 6, 1982) was a Russian-born American novelist, philosopher, playwright, and screenwriter.
>
> S2: She is known for her two best-selling novels, <e2>**The Fountainhead**</e2> and <e2>**Atlas Shrugged**</e2> and for developing a philosophical system she called Objectivism.

As seen in Listing 1, intra-sentence relation extraction attempts to extract related entities (*Ayn Rand, Atlas Shrugged*) for the relation book/author/works_written (Freebase relation) appearing in the same sentence. However, as seen in Listing 2, the related entities appear in different sentences, with the author name present in S1 and the published novels in S2. Traditional relation extraction methods focussing on intra-sentence relation extraction will fail to extract the relation book/author/works_written, between the entities *(Ayan Rand, The Fountainhead), (Ayn Rand, Atlas Shrugged), (Alisa Zinovyevna Rosenbaum, The Fountainhead), (Alisa Zinovyevna Rosenbaum, Atlas Shrugged)* from these two sentences. Thus, in order to extract these relationships, both sentences must be considered collectively.

Inter-sentence relations extraction is important as since significant portion of relations appear across sentences. Swampillai and Stevenson (2010) identify nearly 28.6% of the relations appearing across sentences in the MUC6 dataset. Similarly Roberts et al. (2008) recognize 23% of relation mentions in a biomedical dataset as inter-sentence relation instances. However, a major bottleneck for investigating inter-sentence relation extraction is the absence of a significantly large dataset with inter-sentence relation mentions. Previous studies on inter-sentence relation extraction have employed smaller datasets (Swampillai and Stevenson, 2010; D'Souza and Ng, 2014; Gu et al., 2017). Recently Quirk and Poon (2016; Peng et al. (2017) have investigated inter-sentence relation extraction on a large dataset. However, the study is focused on a specialised domain such drug-gene interaction. Thus, given the absence of a large dataset of inter-sentence relation mentions for generic relations, this study proposes to follow distant supervision approach for developing a datsaet for inter-sentence relation

extraction.
The main contributions of this paper are as follows:

a. A large (benchmark) dataset for inter-sentence relation extraction generated following distant supervision method. The approach employed the resource used by Mintz et al. (2009) to develop a balanced dataset comprising 31,970 sentence pairs with inter-sentence relation mentions involving 17 different relations. The test set for evaluation purposes is created manually by choosing 100 sentence pairs with explicit relation mentions for each of the 17 relations.

b. Present performance of baseline models such as the bag-of-words model and sequence-based neural network models on the developed dataset.

2. Related Work

The related work for the present study can be grouped into the following three strands:
Intra-sentence relation extraction. Mintz et al. (2009) identify atleast three paradigms applied for the task of intra-sentence relation extraction. These are: (a) *supervised learning* approaches focussing on creating hand-labeled data and experimenting with a variety of lexical, syntactic and sematnic features (GuoDong et al., 2005; Surdeanu and Ciaramita, 2007); (b) *unsupervised learning* methods aiming to cluster strings of words exracted from large collections of text (Shinyama and Sekine, 2006; Banko et al., 2007); and (c) *bootstrapping methods* employing small seed sets that focus on pattern-based relation extraction (Brin, 1998; Riloff et al., 1999). Recently, deep learning models such as CNN (Zeng et al., 2014; Santos et al., 2015; Xu et al., 2015a), Recurrent neural networks based models such as LSTM model (Miwa and Bansal, 2016; Xu et al., 2015b) and BiLSTM model (Wu et al., 2017) are shown to be quite useful for intra-sentence relation extraction.
Inter-sentence relation extraction. As explained in the previous section, several studies have focussed on relation extraction across sentences due to its contribution to th eoverall task of relation extraction. Further, comparing intra-sentence and inter-sentence features for clinical research relationship extraction, Roberts et al. (2008) show that intra-sentence features are not very useful for inter-sentence relation extraction. Swampillai and Stevenson (2010) employed features drawn from combining parse trees of sentences for extracting relations across sentences in the MUC6 dataset. Targeting inter-sentence time-event relation extraction, Moschitti et al. (2013) proposed an SVM-model using tree kernels, whichwere evaluated on Machine Reading Program (MRP) and TimeBank datasets. Tree kernels are also shown to be useful for inter-sentence relation extraction in the Chemical-Induced-Disease domain (Nagesh, 2016). More recently, Quirk and Poon (2016) and Peng et al. (2017) developed a large dataset for drug-gene interactions and experimented with graphLSTM models to extract cross-sentence n-ary relation extraction.
Distant supervision for relation extraction. While several methods are employed for dataset creation for relation extraction across entities in a single sentence, distant supervision method has been shown as an useful method for such intra-sentence relation extraction tasks (Mintz et al., 2009; Riedel et al., 2010). The approach of distant supervision facilitates creation of large datasets using seed instances. Although, distant supervision follows a strong assumption that sentences with any two entity mentions for a particular relation, qualify as a candidate for relation extraction, it combines the usefulness of supervised learning approaches, unsupervised learning methods and bootstrapping systems for relation extraction, and is particularly useful in creating large datasets, without manual annotation.
Given the above three strands of research related to the field of relation extraction, it can be clearly noticed that significantly a large number of studies have focused on intra-sentence relation extraction in comparison to the research work on inter-sentence relation extraction. Further, it can also be seen that most of the work in the context of inter-sentence relation extraction have employed smaller datasets (Swampillai and Stevenson, 2010; Gu et al., 2017). Furthermore, more recent works (Quirk and Poon, 2016; Peng et al., 2017) have focused on specialized domains such as Bioinfomatics. Thus, given the absence of a large dataset to investigate inter-sentence relation extraction, particulary involving generic relations, this study proposes to develop a dataset of reasonable size, involving generic relations to facilitate research in the field of inter-sentence relation extraction. Without doubt, the availability of such a dataset can help to explore novel ways of relation extraction across sentences. With this motivation, this study looks at developing a dataset for inter-sentence relation extraction, involving generic relations. Further, although distant supervision method suffers from the "strict assumption" (Riedel et al., 2010), given the usefulness of distant supervision for relation extraction, we propose to follow the distant supervision method for developing dataset for inter-sentence relation extraction. This could serve as a starting point to examine the task of inter-sentence relation extraction. Further, given the recent success of deep learning models for relation extraction (Zeng et al., 2014; Santos et al., 2015; Xu et al., 2015a; Xu et al., 2015b), we propose to evaluate some of these techniques on the developed dataset.

3. Inter-sentence Relation Extraction Dataset

The process of creating inter-sentence relation extraction dataset is described in this section.

3.1. Approach

In the past, Freebase relations have been successfully used for examining relation extraction (Mintz et al., 2009; Bordes et al., 2013; Wanf et al., 2014). The initial work on using distant supervision for relation extraction was proposed by Mintz et al. (2009). The authors developed a large dataset comprising 1.8 million instances using 102 Freebase relations, connecting 940,000 entities. Since then the dataset has been extensively used for evaluation purposes (Riedel et al., 2010; Surdeanu et al., 2012). Thus, given the usefulness of the dataset developed by Mintz et al. (2009), this study proposes to use this resource (102 Freebase relations) for developing a benchmark dataset for inter-sentence relation extraction. Using this resource,

Relation	SI
american/football/football/position/players	9812
architecture/structure/architect	2288
automotive/model/year/body/styles	4806
automotive/model/year/engines	4564
automotive/model/year/exterior/colors	3072
automotive/model/year/make	2740
automotive/model/year/model	2753
automotive/model/year/next/model/year	2354
automotive/model/year/previous/model/year	2415
automotive/model/year/transmissions	4709

Table 1: Sample set of Freebase relations from Mintz et al.(2009) dataset. SI - seed instances.

which is previously examined for intra-sentence relation extraction for developing dataset for inter-sentence relation extraction will facilitate generation of a corpus for generic relations and also help in understanding the scope of inter-sentence relation extraction in the context of intra-sentence relation extraction. As defined previously in §1., this study focuses on extracting sentence pairs with inter-sentence relation mentions. An example of candidate sentence pair for inter-sentence relation extraction was previously seen in Listing 2, where entities e_1 and e_2 are present in the first and the second sentence, respectively. The different statistics of extracted sentence pairs with inter-sentence relation mentions are explained in the following section.

3.2. Dataset Statistics

The process of extracting sentence pairs using the 102 Freebase relations resulted in obtaining nearly 101042 sentence pairs with relation mentions between them, i.e., entity e_1 being present in the first sentence and entity e_2 being present in the second sentence. Table 2 provides the list of 17 Freebase relations with the largest number of sentence pairs with inter-sentence relation mentions.

As can be seen in Table 2, a varied number of sentence pairs are obtained for different relations. For example, while the relation "location/location/contains" obtains a large number of 26599 sentence pairs, the relation "business/company/industry" obtains a lower number of 1529 sentence pairs. There were also other relations that had lesser than 1500 sentence pairs, which are not listed in Table 2, as we do not include those relations in the dataset. Given this varied set of sentence pairs for Freebase relations, in order to develop a balanced dataset for inter-sentence relation extraction, we randomly selected 2000 sentence pairs for all those relations that had more than 2000 sentence pairs and retained all the available sentence pairs for those relations that had less than 2000 sentence pairs. Further, a filtering process was carried out to remove problem sentence pairs, containing reference and hyperlink tags. This resulted in a balanced dataset comprising 35895 sentences for 17 different Freebase relations as shown in Table 2. Table 2 also shows the unique entities used for each relation. The various characteristics of the developed dataset are explained below.

	Initial Set	Balanced Set	
Relation	SP	SP	UE
location/location/contains	26599	1981	1249
film/film/country	16438	1971	458
location/country/administrative/divisions	13113	1978	112
language/human/language/main/country	3939	1982	111
film/film/genre	2946	1955	800
geography/river/basin/countries	2799	1982	685
government/political/party/country	2478	1987	452
film/writer/film	2434	1988	1493
film/director/film	2324	1990	1569
people/person/place_of_birth	2003	1967	1312
tv/tv/program/country_of_origin	1902	1867	611
book/author/works_written	1923	1895	1331
people/person/nationality	1894	1866	1325
people/person/profession	1782	1731	1080
film/producer/film	1715	1705	1169
tv/tv/program_genre	1696	1666	563
business/company/industry	1529	1459	541
Total	87514	31970	14861

Table 2: List of 19 Freebase relations with initial set of inter-sentence relation instances and balanced set created using the initial set. SP - Sentence Pairs, UE - Unique Entities.

3.3. Characteristics of the Dataset

The following are some of the aspects of the developed dataset:

1. Distant supervision assumption. The distant supervision assumption (mentioned in previous section) is preserved while developing the dataset for inter-sentence relation extraction. This results in obtaining a number sentences pairs, where the relation between the related entity pairs is not evident directly. For instance, as seen in Listing 3, the sample sentence pair for the relation business/company/industry does not provide an explicitly visible relationship between the entities for the said relation. However, the seed instances used identifies "Google" as a "Search" industry, resulting in obtaining this sentence pair as a suitable candidate for inter-sentence relation extraction.

LISTING 3: EXAMPLE OF SENTENCE PAIR FOR BUSINESS/COMPANY/INDUSTRY RELATION

> s1: $<e1>$Search$</e1>$ engines also frequently make web pages they have indexed available from their cache.
>
> s2: For example, $<e2>$Google$</e2>$ provides a "Cached" link next to each search result"

2. Filter instances with multiple entity mentions. Further, instances with multiple entity mentions were considered only once in order to remove duplicates across the training and test set. For instance, the example shown in Listing 2, qualify as two instances for the relation book/author/works_written(*Ayan Rand, The Fountainhead*) and book/author/works_written(*Ayan Rand, Atlas Shrugged*). However, we retain only one instance of such relations, by randomly selecting between multiple instances.

3. Coreference resolution. While handling relations across sentences, coreference resolution plays an important role in disambiguating entities between sentences. For instance, in the sample sentence pair for book/author/works_written relation provided in Listing 4, the surname 'Christie' appears in $s2$, referring to 'Agatha

Christie' in $s1$. Such instances are included in the dataset without resolving coreferences, as a suitable candidate for inter-sentence relation extraction, between the entities (*Agatha Christie, The Murder of Roger Ackroyd*) for the relation book/author/works_written.

LISTING 4: EXAMPLE OF SENTENCE PAIR FOR BOOK/AUTHOR/WORKS_WRITTEN RELATION

> s1: $<e1>$Agatha Christie$</e1>$ attributed the inspiration for the character of Miss Marple to a number of sources.
>
> s2: Christie also used material from her fictional creation, spinster Caroline Sheppard, who appeared in $<e2>$The Murder of Roger Ackroyd$</e2>$.

4. Dataset split. To evaluate the dataset, we manually selected 100 instances for each relation resulting in a test dataset comprising about 1900 instances for 17 relations. The remaining corpus i.e., dataset excluding sentences manually selected for test dataset, was randomly split into two sets in the ratio 80:20 to create the training and validation set, respectively. The mode was developed using the validation set, which was tested on the manually created test dataset. Our dataset is created following the distant supervision approach. Therefore, it remains unclear how accurate the annotations produced by following distant supervision. To empirically evaluate the validity of the distant supervision assumption, we randomly select 50 sentence pairs for each of the four relation types book/author/works_written, film/director/film, film/producer/film/ film/writer/film separately, and manually verify whether the relation holds between the two entities in each pair of sentences. This analysis reveals that on average 79% of the sentence pairs annotated using distant supervision are indeed correct. Therefore, we believe that our dataset is sufficiently accurate for training and testing purposes.

4. Experiments

The different baselines models, evaluation metrics and the results of this study are presented here.

4.1. Models

The following models are evaluated for inter-sentence relation extraction.

4.1.1. Bag-of-Words Model

The bag-of-words model provides a simple baseline to evaluate inter-sentence relation extraction by simply concatenating the two sentences with inter-sentence relation mentions. While the bag-of-words model simply combines words in the sentence pair without differentiating between them, it would be interesting to investigate whether differentiating between the words in the sentence pair would help in classification. Accordingly, the following two bag-of-words model, using different feature sets are examined:

1. **BOW-WITHOUT-SB**: bag-of-words model without sentence boundary;
2. **BOW-WITH-SB**: bag-of-words model with sentence boundary.

4.1.2. Sequence based Deep Learning Models

Although recurrent neural networks (RNNs) are useful models for relation extraction through sequential learning, the difficulty in training RNNs using backpropagation through time (Rumelhart et al., 1988), usually results in the vanishing gradient problem (Bengio et al., 1994), wherein the gradient propagated through the network over time either decays or grows exponentially. The Long Short-Term Memory (LSTM) model was proposed to overcome the vanishing gradient by regulating the information in a cell state using input, output and forget gates and thereby learn long-term dependencies (Hochreiter and Schmidhuber, 1997). Recently, LSTM and Bidirectional-LSTM (BiLSTM) models have also been successfully applied for the task of intra-sentence relation extraction (Ma and Hovy, 2016). Inter-sentence relation classification can be considered as a sequence classification problem, with the task to predict the relation given the sequence of words across the sentences. Theoretically, LSTM should be helpful for inter-sentence relation extraction, with its capability of handling long-term dependencies from long sequences of words. Further, while LSTM model captures the context only in the forward direction, BiLSTM models process the data in both directions with two separate hidden layers, which are then provided to the output layer. Accordingly, the following LSTM-based models are evaluated for the task of inter-sentence relation extraction:

1. **LSTM-MODEL** LSTM model for inter sentence relation extraction which uses the words between the two entities across sentences and also learns embeddings for entities in different sentences;
2. **BILSTM-MODEL**: Bidirectional LSTM model for inter sentence relation extraction which uses the words between the two entities across sentences and also learns embeddings for entities in different sentences

4.2. Evaluation Metrics

The Precision P, Recall R and F-Score F, as defined below are measured in order to evaluate the performance of different models.

$$P = \frac{\text{Number of correctly extracted entity relations}}{\text{Total number of extracted entity relations}}$$

$$R = \frac{\text{Number of correctly extracted entity relations}}{\text{Actual number of extracted entity relations}} \quad F = \frac{2PR}{P+R}$$

4.3. Results and Discussion

The performance of different models for inter-sentence relation on the proposed dataset is shown in Table 3. The precision, recall and F-scores scored by different models for individual relations is provided in Table 4. The following explains the results of this study.

4.4. Best performing model

As seen in Table 3, the LSTM-based models (LSTM-MODEL and BILSTM-MODEL) achieve a comparatively higher F-score of (0.70 and 0.72 respectively) against the bag-of-words models (BOW-WITHOUT-SB and BOW-WITH-SB), which score an F-score of 0.65 and 0.66, respectively

Model	P	R	F
BOW-WITHOUT-SB	0.67	0.65	0.65
BOW-WITH-SB	0.68	0.66	0.66
LSTM-MODEL	0.72	0.70	0.70
BILSTM-MODEL	0.73	0.72	0.72

Table 3: Performance of BAG-OF-WORDS, LSTM and BILSTM models for Inter-sentence Relation Extraction on test dataset, P - Precision, R - Recall, F - F-score

on the test set. This shows that sequence based recurrent neural network models are more useful for the task of inter-sentence relation extraction, in comparison to simple bag-of-words models due to their ability in learning from long range sequential information between entities across sentences. The inclusion of information to distinguish between information obtained from different sentences does not seem to help much in increasing the performance since the BOW-WITH-SB achieved a mere improvement of 1 point in terms of F-score obtained against the BOW-WITHOUT-SB model.

Among the different evaluated models, the BILSTM model achieved the highest F-score of 0.72 in comparison to other models. However, the performance obtained using BILSTM does not provide a statistically significant improvement against the other examined models. In comparison to the regular LSTM model which achieves an F-score of 0.70, the BILSTM model achieves a little improvement by obtaining an F-score of 0.72. Although the difference between these two models is not statistically significant, the results indicate that it would be useful to use models such as BILSTM for the task of inter-sentence relation extraction, as these models learn from sequential information from both directions.

4.5. Poor performing relations

The precision, recall and F-scores obtained for individual relations (shown in Table 4) shows that the models perform significantly better for certain relations and poor for some relations[1]. For example, almost all models achieve significantly higher F-scores for the following relations 'business/company/industry', 'people/person/profession', 'location/country/administrative/divisions', 'tv/tv/program_genre' and 'government/political/party/country'. These results shows that these models are able to easily learn from the available features (words) for these relations.

However, as seen in Table 4, there are a number of relations, where the models achieve a significantly lower F-score. For example, for the following relations 'film/director/film', 'film/film/country', 'film/writer/film', and 'film/producer/film', the models achieve a significantly poor F-score. The confusion matrix for these relations indicates that a large number of instances for these relations are wrongly classified as other relations. For example, instances for the 'film/film/country' relation are wrongly classified as 'tv/tv/program/country_of_origin' relation. This indicates that the features between these two relations are so common that the classifier fails to differentiate between the two relations. For instance, consider instances in Listings 5 and 6 for the relation film/film/country. Though these instances are annotated for 'film/film/country' relation, features such as 'television' can render the instance to be classified as 'tv/tv/program/country'.

LISTING 5: EXAMPLE FOR FILM/FILM/COUNTRY RELATION

S1: Victoria Schmidt is a $<e1>$New Zealand$</e1>$ theater, film and television actress.

S2: She is most known for her role as Aaliyah in $<e2>$Siones Wedding$</e2>$ (2006).

LISTING 6: EXAMPLE FOR FILM/FILM/COUNTRY RELATION

S1: The Legendary Fok is a $<e1>$Hong Kong$</e1>$ television series.

S2: It includes a subplot based on the protagonist of the 1972 film $<e2>$Fist of Fury$</e2>$.

Similarly, a number of instances for the relation 'film/writer/film' are classified as 'film/director/film' and vice-versa. The reason for this confusion is that many a times, the director himself is the writer of the story or the screenplay for the film. For example, consider the instance in Listing 7. The writer of the movie has different roles in terms of actor, writer and director of the film, making it difficult for the classifier to identify the correct relation.

LISTING 7: EXAMPLE FOR FILM/WRITER/FILM RELATION

S1: $<e1>$Luis Valdez$</e1>$ is an American playwright, actor, writer and film director.

S2: He is best known for his movie $<e2>$La Bamba$</e2>$.

4.5.1. Intra-sentence vs. inter-sentence relation extraction

As explained in the preceding sections, the lstm-based models (LSTM and BILSTM) achieve higher performance on the task of inter-sentence relation extraction. However, working in the context of intra-sentence relation extraction, Xu et al. (2015b) report an F-score of 0.82 by training an LSTM model using word embeddings. However, instead of using all the words between the entities in the sentence, Xu et al. (2015b) use the words in the shortest dependency path between the two entities in the sentence. The results (F-score of 0.82) achieved by Xu et al. (2015b) is significantly higher in comparison to the F-scores achieved by both LSTM and BILSTM models (F-score of 0.70 and 0.72) for the task of intra-sentence relation extraction. The comparison of these results clearly indicate that the task of inter-sentence relation extraction focusing on extracting relations between entities across sentences is more difficult than intra-sentence relation extraction, which focuses on extracting relations between entities in a single sentence. The major challenge is to model the long-range information between the entities across sentences and thus, requires more sophisticated models other than simple LSTM and BILSTM models that use words between entities across sentences.

[1]The relations are sorted in decreasing order according to their performance $w.r.t$ BILSTM model

Relation	P	R	F	P	R	F	P	R	F	P	R	F
business/company/industry	0.94	0.89	0.91	0.96	0.90	0.92	0.97	0.97	0.97	0.85	0.89	0.86
people/person/profession	0.79	0.73	0.75	0.89	0.80	0.84	0.99	0.95	0.96	0.90	0.90	0.90
geography/river/basin/countries	0.35	0.37	0.35	0.36	0.38	0.36	0.87	0.92	0.89	0.69	0.62	0.65
location/country/administrative/divisions	0.80	0.81	0.80	0.82	0.82	0.82	0.92	0.82	0.86	0.80	0.91	0.85
tv/tv/program_genre	0.67	0.57	0.61	0.74	0.60	0.66	0.93	0.75	0.83	0.37	0.51	0.42
government/political/party/country	0.86	0.87	0.86	0.86	0.88	0.86	0.79	0.86	0.82	0.78	0.79	0.78
film/film/genre	0.54	0.53	0.53	0.57	0.62	0.59	0.81	0.92	0.86	0.97	0.99	0.97
location/location/contains	0.72	0.68	0.69	0.74	0.68	0.70	0.79	0.79	0.79	0.96	0.76	0.84
people/person/place_of_birth	0.71	0.62	0.66	0.74	0.65	0.69	0.82	0.71	0.79	0.80	0.75	0.77
book/author/works_written	0.67	0.84	0.74	0.65	0.85	0.73	0.86	0.75	0.80	0.54	0.46	0.49
people/person/nationality	0.53	0.42	0.46	0.49	0.35	0.40	0.72	0.54	0.61	0.93	0.92	0.92
tv/tv/program/country_of_origin	0.66	0.77	0.71	0.66	0.82	0.73	0.49	0.90	0.63	0.39	0.39	0.39
language/human/language/main/country	0.91	0.88	0.88	0.96	0.87	0.91	0.75	0.64	0.69	0.27	0.14	0.18
film/producer/film	0.85	0.91	0.87	0.81	0.92	0.86	0.65	0.41	0.50	0.84	0.85	0.84
film/writer/film	0.47	0.21	0.29	0.40	0.17	0.23	0.18	0.04	0.06	0.98	0.93	0.95
film/film/country	0.67	0.61	0.63	0.70	0.61	0.65	0.37	041	0.41	0.55	0.78	0.64
film/director/film	0.27	044	0.33	0.28	0.45	0.34	0.38	0.62	0.47	0.80	0.74	0.76
Average	0.67	0.65	0.65	0.68	0.66	0.66	0.72	0.70	0.70	**0.73**	**0.72**	**0.72**

Table 4: Precision (P), Recall (R) and F-scores (F) obtained by different models for individual relations.

5. Conclusion

To conclude, this study resulted in creating a benchmark dataset for inter-sentence relation extraction. The study followed distant supervision method for creating the dataset involving relations previously used for creating resources for intra-sentence relation extraction. Accordingly, this study resulted in developing a balanced dataset comprising a large number of sentence pairs with inter-sentence relation mentions for 17 different relations. The study also evaluated certain baseline models such as bag-of-words and sequence based recurrent neural network models on the developed dataset. The study shows that recurrent neural network models are more useful for the task of intra-sentence relation extraction, in comparison to bag-of-words model. However, the intra-sentence relation extraction results obtained in this study in comparison to intra-sentence relation extraction, indicate the need for more sophisticated models for handling the long-range information between entities across sentences.

6. Bibliographical References

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *IJCAI*, volume 7, pages 2670–2676.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhenko, O. (2013). Translating embeddings for modeling multi-relational data. In *Proc. of NIPS*.

Brin, S. (1998). Extracting patterns and relations from the world wide web. In *International Workshop on The World Wide Web and Databases*, pages 172–183. Springer.

Culotta, A. and Sorensen, J. (2004). Dependency tree kernels for relation extraction. In *Proceedings of the 42nd annual meeting on association for computational linguistics*, page 423. Association for Computational Linguistics.

D'Souza, J. and Ng, V. (2014). Annotating inter-sentence temporal relations in clinical notes. In *LREC*, pages 2758–2765.

Etzioni, O., Fader, A., Christensen, J., Soderland, S., and Mausam, M. (2011). Open information extraction: The second generation. In *IJCAI*, volume 11, pages 3–10.

Gu, J., Sun, F., Qian, L., and Zhou, G. (2017). Chemical-induced disease relation extraction via convolutional neural network. *Database*, 2017.

GuoDong, Z., Jian, S., Jie, Z., and Min, Z. (2005). Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Ma, X. and Hovy, E. H. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Miwa, M. and Bansal, M. (2016). End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Moschitti, A., Patwardhan, S., and Welty, C. (2013). Long-distance time-event relation extraction. In *IJCNLP*, pages 1330–1338.

Nagesh, P. (2016). Exploiting tree kernels for high perfor-

mance chemical induced disease relation extraction. In *4TH ANNUAL DOCTORAL COLLOQUIUM*, page 15.

Peng, N., Poon, H., Quirk, C., Toutanova, K., and Yih, W.-t. (2017). Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Quirk, C. and Poon, H. (2016). Distant supervision for relation extraction beyond the sentence boundary. *arXiv preprint arXiv:1609.04873*.

Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Riloff, E., Jones, R., et al. (1999). Learning dictionaries for information extraction by multi-level bootstrapping. In *AAAI/IAAI*, pages 474–479.

Roberts, A., Gaizauskas, R., and Hepple, M. (2008). Extracting clinical relationships from patient narratives. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 10–18. Association for Computational Linguistics.

Rumelhart, D. E., Hinton, G. E., Williams, R. J., et al. (1988). Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1.

Santos, C. N. d., Xiang, B., and Zhou, B. (2015). Classifying relations by ranking with convolutional neural networks. *arXiv preprint arXiv:1504.06580*.

Shinyama, Y. and Sekine, S. (2006). Preemptive information extraction using unrestricted relation discovery. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 304–311. Association for Computational Linguistics.

Surdeanu, M. and Ciaramita, M. (2007). Robust information extraction with perceptrons. In *Proceedings of the NIST 2007 Automatic Content Extraction Workshop (ACE07)*.

Surdeanu, M., Tibshirani, J., Nallapati, R., and Manning, C. D. (2012). Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics.

Swampillai, K. and Stevenson, M. (2010). Inter-sentential relations in information extraction corpora. In *LREC*.

Wanf, Z., Zhang, J., Feng, J., and Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. In *Proc. of AAAI*, pages 1112 – 1119.

Wu, M., Liu, L., Yao, W., Yin, C., and Wang, J. (2017). Semantic relation classification by bi-directional lstm architecture. *Advanced Sciences and Technology Letters*.

Xu, K., Feng, Y., Huang, S., and Zhao, D. (2015a). Semantic relation classification via convolutional neural networks with simple negative sampling. *arXiv preprint arXiv:1506.07650*.

Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., and Jin, Z. (2015b). Classifying relations via long short term memory networks along shortest dependency paths. In *EMNLP*, pages 1785–1794.

Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al. (2014). Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344.

Diacritics Restoration Using Neural Networks

Jakub Náplava, Milan Straka, Pavel Straňák, Jan Hajič
Institute of Formal and Applied Linguistics
Charles University, Faculty of Mathematics and Physics
Malostranské náměstí 25, Prague, Czech Republic
{naplava,straka,stranak,hajic}@ufal.mff.cuni.cz

Abstract

In this paper, we describe a novel combination of a character-level recurrent neural-network based model and a language model applied to diacritics restoration. In many cases in the past and still at present, people often replace characters with diacritics with their ASCII counterparts. Despite the fact that the resulting text is usually easy to understand for humans, it is much harder for further computational processing. This paper opens with a discussion of applicability of restoration of diacritics in selected languages. Next, we present a neural network-based approach to diacritics generation. The core component of our model is a bidirectional recurrent neural network operating at a character level. We evaluate the model on two existing datasets consisting of four European languages. When combined with a language model, our model reduces the error of current best systems by 20% to 64%. Finally, we propose a pipeline for obtaining consistent diacritics restoration datasets for twelve languages and evaluate our model on it. All the code is available under open source license on https://github.com/arahusky/diacritics_restoration.

Keywords: neural networks, diacritics, diacritics generation, error correction

1. Introduction

When writing emails, tweets or texts in certain languages, people for various reasons sometimes write without diacritics. When using Latin script, they replace characters with diacritics (e.g. c with acute or caron) by the underlying basic character without diacritics. Practically speaking, they write in ASCII. We offer several possible reasons for this phenomenon:

- Historically, many devices offered only an English keyboard and/or ASCII encoding (for example older mobile phones and SMS).
- Before Unicode became widespread, there were encoding problems among platforms and even among programs on the same platform, and many people still have this in mind.
- Even though text encoding is rarely a problem any more and all modern devices offer native keyboards, some problems persist. In situations of frequent code switching between English and a language with a substantially different keyboard layout, it is very hard to touch type in both layouts. It is much easier to type both languages using the same layout, although one of them without proper diacritics.
- In some circumstances typing with diacritical marks is significantly slower than using just basic latin characters. The most common example is on-screen keyboards on mobile devices. These keyboards do not include the top row (numerical on US English), so languages that use that row for accented characters are much slower to type. Naturally, users type without explicit accents and rely on the auto-completion systems. However, these systems are usually simple, unigram-based, and based on the word form ambiguity for a given language (cf. Table 1), which introduces many errors. Postponing the step of diacritics generation would be beneficial both for typing speed and accuracy.
- For example in Vietnamese, the language with most diacritics in our data (cf. Table 1), both the above problems are very pronounced: Because Vietnamese uses diacritical marks to distinguish both tones (6) and quality of vowels (up to 3), a vowel can have (and often has) 2 marks. This need to provide efficient typing of very many accented characters led to the invention of systems like unikey.org that allow a user to type all the accented characters using sequences of basic letters. For instance to typeset "đường" a user types "dduwowngf". While this system elegantly solves the problems above with switching keyboard layouts and missing top row of keys, it requires a special software package and it still results in 9 keystrokes to type 5 characters. That is why typing without accents in informal situations like emails or text messages is still common and system for efficient generation of diacritics would be very useful.

Typical languages where approximately half of the words contain diacritics are Czech, Hungarian or Latvian. Nevertheless, as we discuss in Sections 2 and 3, diacritics restoration (also known as diacritics generation or diacritization) is an active problem also in many languages with substantially lower diacritics appearance.

Current approaches to restoration of diacritics (see Section 3) are mostly based on traditional statistical methods. However, in recent years, deep neural networks have shown remarkable results in many areas. To explore their capabilities, we propose a neural based model in Section 4 and evaluate its performance on two datasets in Section 5.

In Section 6, we describe a way to obtain a consistent multilingual dataset for diacritics restoration and evaluate our model on them. The dataset can be downloaded from the published link. Finally, Section 7 concludes this paper with a summary of outcomes.

Language	Words with diacritics	Word error rate of dictionary baseline
Vietnamese	88.4%	40.53%
Romanian	31.0%	29.71%
Latvian	47.7%	8.45%
Czech	52.5%	4.09%
Slovak	41.4%	3.35%
Irish	29.5%	3.15%
French	16.7%	2.86%
Hungarian	50.7%	2.80%
Polish	36.9%	2.52%
Swedish	26.4%	1.88%
Portuguese	13.3%	1.83%
Galician	13.3%	1.62%
Estonian	19.7%	1.41%
Spanish	11.3%	1.28%
Norwegian-Nynorsk	12.8%	1.20%
Turkish	30.0%	1.16%
Catalan	11.1%	1.10%
Slovenian	14.0%	0.97%
Finnish	23.5%	0.89%
Norwegian-Bokmaal	11.7%	0.79%
Danish	10.2%	0.69%
German	8.3%	0.59%
Croatian	16.7%	0.34%

Table 1: Analysis of percentage of words with diacritics and the word error rate of a dictionary baseline. Measured on UD 2.0 data, using the *uninames* method, and CoNLL 17 UD shared task raw data for dictionary. Only words containing at least one alphabetical character are considered.

2. Diacritics Restoration in Languages using Latin Script

Table 1 presents languages using (usually some extended version of) a Latin script. Employing UD 2.0 (Nivre et al., 2017) plain text data, we measure the ratio of words with diacritics, omitting languages with less than 5% of words with diacritics. In eleven of the languages, at least every fifth word contains diacritics; in another eleven languages, at least every tenth word does.

Naturally, high occurrence of words with diacritics does not imply that generating diacritics is an ambiguous task. Consequently, we also evaluate word error rate of a simple dictionary baseline to diacritics restoration: according to a large raw text corpora we construct a dictionary of the most frequent variant with diacritics for a given word without diacritics, and use the dictionary to perform the diacritics restoration.

Table 1 presents the results. We utilized the raw corpora by Ginter et al. (2017) released as supplementary material of CoNLL 2017 UD Shared task (Zeman et al., 2017), which contain circa a gigaword for each language, therefore providing a strong baseline. For nine languages, the word error rate is larger than 2%, and eight more languages have word error rate still above 1%. We conclude that even with a very large dictionary, the diacritics restoration is a challenging task for many languages, and better method is needed.

Letter	Hex code	Unicode name
ø	00F8	LATIN SMALL LETTER O *WITH STROKE*
ł	0142	LATIN SMALL LETTER L *WITH STROKE*
đ	0111	LATIN SMALL LETTER D *WITH STROKE*
ʂ	0282	LATIN SMALL LETTER S *WITH HOOK*
ç	00E7	LATIN SMALL LETTER C *WITH CEDILLA*
š	0161	LATIN SMALL LETTER S *WITH CARON*

Table 2: Unicode characters that cannot be decomposed using NFD (first 4 lines), and those that can. The suffix of the name removed by the *uninames* method is show in italics. As we can see, the strucutre of names is identical, so the method works for all of these characters.

2.1. Methods of Diacritics Stripping

Although there is no standard way of stripping diacritics, a commonly used method is to convert input word to NFD (The Unicode Consortium, 2017, Normalization Form D) which decomposes composite characters into a base character and a sequence of combining marks, then remove the combining marks, and convert the result back to NFC (Unicode Normalization Form C). We dub this method *uninorms*.

We however noted that this method does not strip diacritics for some characters (e.g. for đ and ł).[1] We therefore propose a new method *uninames*, which operates as follows: In order to remove diacritics from a given character, we inspect its name in the Unicode Character Database (The Unicode Consortium, 2017). If it contains a word *WITH*, we remove the longest suffix starting with it, try looking up a character with the remaining name and yield the character if it exists.

The method is illustrated in Table 2, which presents four characters that do not decompose under NFD, but whose diacritics can be stripped by the proposed method.

As shown in Table 3, the proposed *uninames* method recognizes all characters the *uninorms* method does, and some additional ones. Therefore, we employ the *uninames* method to strip diacritics in the paper.

3. Related Work

One of the first papers to describe systems for automatic diacritics restoration is a seminal work by Yarowsky (1999), who compares several algorithms for restoration of diacritics in French and Spanish. Later, models for diacrization in Vietnamese (Nguyen and Ock, 2010), Czech (Richter et al., 2012), Turkish (Adali and Eryiğit, 2014), Ara-

[1] What constitutes a "diacritic mark" is a bit of a problem. On one hand not all characters with a graphical element added to a letter letter contain diacritics, e.g. ¥ (symbol of Japanese Yen) or Đ/ð (Icelandic "eth"). On the other end of the spectrum we have clear diacritics with Unicode canonical decomposition into a letter and a combining mark. Between these clear borders there are the characters that do not have a unicode decomposition, but their names still indicate they are latin letters with some modifier and often they are used the same as characters that do have decomposition. E.g. Norwegian/Danish ø is used exactly like ö in Swedish, it is just an orthographic variation. However while the latter has canonical decomposition in Unicode, the first does not. This is why we opted to treat these characters also as "letters with diacritics".

Lowercased letters with diacritics

í 15.7%	ů 2.5%	ş	ī	ņ	ō
á 11.7%	ú 1.6%	ā	ē	ū	ŕ
é 9.8%	ö 1.5%	ñ	ą	ï	Ĺ
ě 6.6%	ă 1.3%	**ł**	ò	ļ	ţ
ä 6.0%	**ø** 1.1%	ć	ż	ń	ì
č 5.5%	à 0.9%	ň	ğ	ù	ó
ř 5.2%	ç 0.9%	â	ś	û	ḥ
ž 4.9%	ü 0.8%	õ	ď	ű	ă
ý 4.5%	ã 0.8%	ť	ô	ķ	ί
š 4.4%	è 0.6%	ê	**đ**	ģ	έ
ó 3.2%	î 0.5%	ę	ő	ź	ά
å 2.9%	ț 0.5%	ş	ľ	ë	**ş**

Table 3: Most frequent characters with diacritics from data listed in Table 1, together with their relative frequency. The bold characters are recognized only using the *uninames* method.

bic (Azmi and Almajed, 2015) Croatian, Slovenian, Serbian (Ljubešic et al., 2016), and many other languages were published. The system complexity ranges from simplest models, that for each word apply its most frequent translation as observed in the training data, to models that incorporate language models, part-of-speech tags, morphological and many other features. One of the most similar model to ours is a system by Belinkov and Glass (2015) who used recurrent neural networks for Arabic diacritization.

4. Model Architecture

The core of our model (see Figure 1) is a bidirectional recurrent neural network, which for each input character outputs its correct label (e.g. its variant with diacritics). The input and output vocabularies contain a special out-of-alphabet symbol.

The input characters are embedded, i.e. each character in the input sentence is represented by a vector of d real numbers. The character embeddings are initialized randomly and updated during training.

The embeddings are fed to a bidirectional RNN (Graves and Schmidhuber, 2005). The bidirectional RNN consists of two unidirectional RNNs, one reading the inputs in standard order (forward RNN) and the other in reverse order (backward RNN). The output is then a sum of forward and backward RNN outputs. This way, bidirectional RNN is processing information from both preceding and following context. The model allows an arbitrary number of stacked bidirectional RNN layers.

The output of the (possibly multilayer) bidirectional RNN is at each time step reduced by an identical fully connected layer to an o-dimensional vector, where o is the size of the output alphabet. A nonlinearity is then applied to these reduced vectors.

Finally, we use a softmax layer to produce a probability distribution over output alphabet at each time step.

The loss function is the cross-entropy loss summed over all outputs.

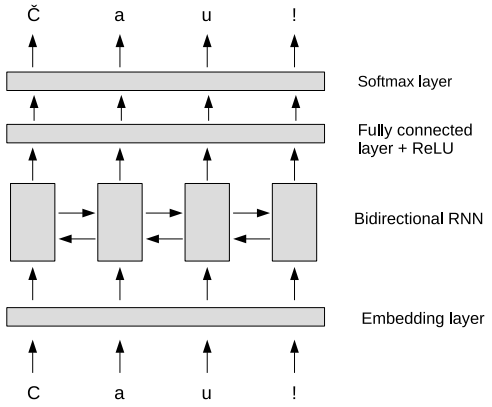


Figure 1: Visualisation of our model.

4.1. Residual connections

The proposed model allows an arbitrary number of stacked RNN layers. The model with multiple layers allows each stacked layer to process more complex representation of current input. This naturally brings potential to improve accuracy of the model.

As stated by (Wu et al., 2016), simple stacking of more RNN layers works only up to a certain number of layers. Beyond this limit, the model becomes too difficult to train, which is most likely caused by vanishing and exploding gradient problems (Pascanu et al., 2013). To improve the gradient flow, (Wu et al., 2016) incorporate residual connections to the model. To formalize this idea, let RNN_i be the i-th RNN layer in a stack and $x_0 = (\mathit{inp}_1, \mathit{inp}_2, \ldots, \mathit{inp}_N)$ input to the first stacked RNN layer RNN_0. The model we have proposed so far works as follows:

$$o_i, c_i = \mathit{RNN}_i(x_i)$$
$$x_{i+1} = o_i$$
$$o_{i+1}, c_{i+1} = \mathit{RNN}_{i+1}(x_{i+1}),$$

where o_i is the output of i-th stacked RNN layer and c_i is a sequence of its hidden states. The model with residual connections between stacked RNN layers then works as follows:

$$o_i, c_i = \mathit{RNN}_i(x_i)$$
$$x_{i+1} = o_i + x_i$$
$$o_{i+1}, c_{i+1} = \mathit{RNN}_{i+1}(x_{i+1})$$

4.2. Decoding

For inference we use a left-to-right beam search decoder combining the neural network and the language model likelihoods. The process is a modified version of standard beam search used by Xie et al. (2016) for decoding sequence-to-sequence models.

Let b denote the beam size. The hypotheses in the beam are initialized with the b most probable first characters. In each step, all beam hypotheses are extended with b most probable variants of the respective character, creating b^2 hypotheses. These are then sorted and the top b of them are kept.

Whenever a space is observed in the output, all affected hypotheses are reranked using both the RNN model output

probabilities and language model probabilities. The hypothesis probability in step k can be computed as:

$$P(y_{1:k}|x) = (1-\alpha)\log P_{NN}(y_{1:k}|x) + \alpha \log P_{LM}(y_{1:k}),$$

where x denotes the input sequence, y stands for the decoded symbols contained within current hypothesis, P_{NN} and P_{LM} are neural network and language model probabilities and the hyper-parameter α determines the weight of the language model. To keep both $\log P_{NN}$ and $\log P_{LM}$ terms within a similar range in the decoding, we compute the $\log P_{NN}$ as the mean of output token log probabilities and additionally normalize P_{LM} by the number of words in the sequence.
To train the language model as well as to run it, we use the open-source KenLM toolkit (Heafield, 2011).

5. Experiments

To compare performance of our model with current approaches, we perform experiments using two existing datasets. The first one, created by Ljubešic et al. (2016), consists of Croatian, Serbian and Slovenian sentences from three sources: Wikipedia texts, general Web texts and texts from Twitter. Since Web data are assumed to be the noisiest, they are used only for training. Wikipedia and Twitter testing sets should then cover both standard and non-standard language. The second evaluation dataset we utilize consists of Czech sentences collected mainly from newspapers, thus it covers mostly standard Czech.

5.1. Training and Decoding Details

We used the same model configuration for all experiments. The bidirectional RNN has 2 stacked layers with residual connections and utilizes LSTM units (Hochreiter and Schmidhuber, 1997) of dimension 300. Dropout (Srivastava et al., 2014) at a rate of 0.2 is used both on the embedded inputs and after each bidirectional layer. All weights are initialized using Xavier uniform initializer (Glorot and Bengio, 2010).
The vocabulary of each experiment consists of top 200 most occurring characters in a training set and a special symbol (<UNK>) for unknown characters.
To train the model, we use the Adam optimizer (Kingma and Ba, 2014) with learning rate 0.0003 and a minibatch size of 200. Each model was trained on a single GeForce GTX 1080 Ti for approximately 4 days. After training, the model with the highest accuracy on the corresponding development set was selected.
To estimate the decoding parameter α, we performed an exhaustive search over [0,1] with a step size of 0.1. The parameter was selected to maximize model performance on a particular development set. All results were obtained using a beam width of 8.

5.2. Croatian, Serbian and Slovenian

The original dataset contains training files divided into Web, Twitter and Wikipedia subsets. However, Ljubešic et al. (2016) showed that concatenating all these language-specific sets for training yields best results. Therefore, we used only concatenated files for training the models for each of three languages in our experiments. The training files contain 17 968 828 sentences for Croatian, 11 223 924 sentences for Slovenian and 8 376 810 sentences for Serbian.[2] All letters in the dataset are lowercased.
To remove diacritics from the collected texts, Ljubešic et al. (2016) used a simple script that replaced four letters (ž, ć, č, š) with their ASCII counterparts (z, c, c, s), and one letter (đ) with its phonetic transcription (dj). This results in the input and target sentences having different length. Since our model requires both input and target sentences to have the same length, additional data preprocessing was required before feeding the data into the model: we replace all occurrences of the *dj* sequence in both input and target sentences by a special token, and replace it back to *dj* after decoding.
The results of the experiment with comparison to previous best system (*Lexicon*, *Corpus*) are presented in Table 4. The *Lexicon* method replaces each word by its most frequent translation as observed in the training data. The *Corpus* method extends it via log-linear model with context probability. These methods were evaluated by Ljubešic et al. (2016) and the *Corpus* method is to the best of our knowledge state-of-the-art system for all three languages. System accuracy is measured, similarly to the original paper, on all words which have at least one alphanumerical character.
We incorporated the same language models as used by the authors of the original paper. There are two points in the results we would like to stress:

- Our system with language model reduces error by more than 30% on wiki data and by more than 20% on tweet data. Moreover, our model outperforms the current best system on wiki data even if it does not incorporate the additional language model, which makes the model much smaller (˜30MB instead of several gigabytes of the language model).
- Diacritics restoration problem is easier on standard language (wiki) than on non-standard data (tweets). This has, in our opinion, two reasons. First, the amount of wiki data in the training sets is substantially higher than the amount of non-standard data (tweets). This makes the model fit more standard data. Second, due to lower language quality in Twitter data, we suppose that the amount of errors in the gold data is higher.

5.3. Czech

The second experiment we conducted is devoted to diacritics restoration in Czech texts. To train both the neural network and language models, we used the SYN2010 corpus (Křen et al., 2010), which contains 8 182 870 sentences collected from Czech literature and newspapers. To evaluate the model, PDT3.0 (Hajič et al., 2018) testing set with 13 136 sentences originating from Czech newspapers is used. Both the training and testing set, thus, contain mainly standard Czech. For language model training, we consider only those {2,3,4,5}-grams that occurred at least twice, and use default KenLM options.
Table 5 presents a comparison of our model performance with Microsoft Office Word 2010, ASpell, CZACCENT (Rychlý, 2012) and Korektor (Richter et al., 2012), the latter being the state-of-the-art system of diacritics restoration

[2]The Serbian dataset is based on a Latin script.

System	wiki			tweet		
	hr	sr	sl	hr	sr	sl
Lexicon	0.9936	0.9924	0.9933	0.9917	0.9893	0.9820
Corpus	0.9957	0.9947	0.9962	0.9938	0.9917	0.9912
Our model	0.9967	0.9961	0.9970	0.9932	0.9939	0.9882
Our model + LM	**0.9973**	**0.9968**	**0.9974**	**0.9951**	**0.9944**	**0.9930**
Error reduction	36.81%	39.74%	30.45%	21.62%	32.14%	20.77%

Table 4: Results obtained on Crotian (HR), Serbian (SR) and Slovenian (SL) Wikipedia and Twitter testing sets. Note that the word accuracy presented in the table is not measured on all words, but only on words having at least one alphanumerical character.

System	Word accuracy
Microsoft Office Word 2010 (*)	0.8910
ASpell (*)	0.8839
Lexicon	0.9527
CZACCENT	0.9607
Corpus	0.9713
Korektor	0.9861
Our model	0.9887
Our model + LM	**0.9951**
Error reduction	64.75%

Table 5: Comparison of several models of restoration of diacritics for Czech. The (*) denotes reduced test data (see text).

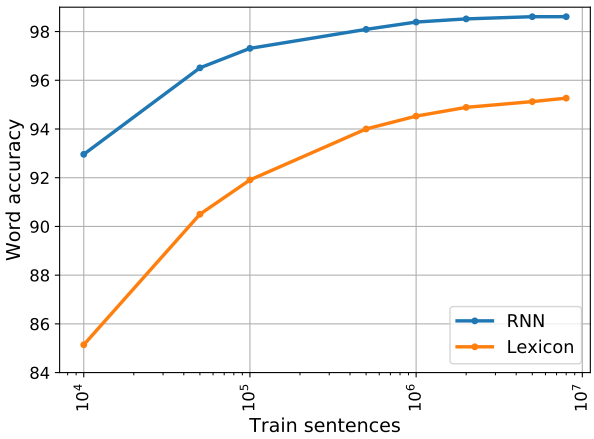


Figure 2: Comparison of RNN and Lexicon performance with varrying training data size.

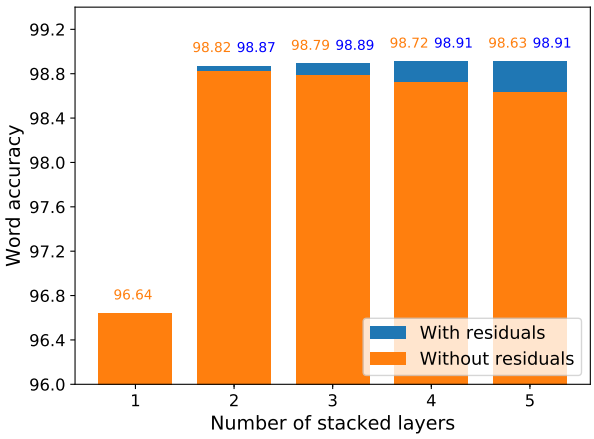


Figure 3: Effect of using residual connections with respect to the number of stacked layers.

for Czech. Note that evaluation using Microsoft Office Word 2010 and ASpell was performed only on the first 746 (636) sentences, because it requires user interaction (confirming the suggested alternatives).

As the results show, models that are not tuned to the task of diacritics restoration perform poorly. Our model combined with a language model reduces the error of the previous state-of-the-art system by more than 60%; our model achieves slightly higher accuracy than Korektor even if no language model is utilized.

5.3.1. Ablation Experiments

One of the reasons why deep learning works so well is the availability of large training corpora. This motivates us to explore the amount of data our model needs to perform well. As Figure 2 shows, the RNN model trained on 50 000 random sentences from SYN2010 corpus performs better on the PDT3.0 testing set than the *Lexicon* baseline trained on full SYN2010 corpus. Further, up to 5M sentences the performance of the RNN model increases with the growing training set size. We do not observe any performance difference between the RNN model trained on 5M and 8M sentences.

The second ablation experiment examines the effect of residual connections. We trained models with 2, 3, 4 and 5 stacked layers each with and without residual connections. We also trained a simple model with 1 bidirectional layer without residual connections. The results of this experiment are presented in Figure 3. Apart from the big difference in word accuracy between the model with 1 layer and other models, we can see that models with residual connections perform generally better than when no residual connections are incorporated. It is also evident that when more layers are added in stack, performance of models without residual connections deteriorates while performance of models with additional residual connections does not.

6. New Multilingual Dataset

As discussed in the preceding sections, diacritics restoration is an active field of research. However, to the best of our knowledge, there is no consistent approach to obtaining datasets for this task. When a new diacritics restoration sys-

tem is published, a new dataset is typically created both for training and testing. This makes it difficult to compare performance across systems. We thus propose a new pipeline for obtaining consistent multilingual datasets for the task of diacritics restoration.

6.1. Dataset

As the data for diacritics restoration need to be clean, we decided to utilize Wikipedia for both development and testing sets. Because there may be not enough data to train potential diacritics restoration systems on Wikipedia texts only, we further decided to create training sets from the (general) Web. We chose two corpora for this task: the W2C corpus (Majliš, 2011) with texts from Wikipedia and the general Web in 120 languages, and the CommonCrawl corpus with language annotations generated by Buck et al. (2014) with a substantially larger amount of general Web texts in more than 150 languages.

To create training, development and testing data from the Wikipedia part of the W2C corpus, its data are first segmented into sentences, these are then converted to lowercase and finally split into disjoint training, development and testing set. The split was performed in such a way that all three sets consist of sentences collected from whole articles rather then being randomly sampled across all articles. Each testing set consists of 30 000 sentences, development set of approximatelly 15 000 sentences and the rest of the data are preserved for training set.

The pipeline for creating additional training data from the CommonCrawl corpus starts with the removal of invalid UTF8 data and Wikipedia data. These filtered data are then segmented into sentences and converted to lowercase. Since these data come from general Web and may be noisy (e.g. contain sentences with missing diacritics), only those sentences that have at least 100 characters and contain at least a certain amount of diacritics are preserved. The constant determining the minimum amount of diacritics is language specific and is derived from Table 1. Finally, sentence intersection with existing development and testing set is removed and maximally ten similar sentences are preserved in the training data. Since both baseline methods (*Lexicon* and *Corpus*) require data to be word tokenized, all texts are also word tokenized.

The dataset was created for 12 languages (see Table 6), where the additional training sets were generated from the *2017_17* web crawl. Complete dataset can be downloaded from `http://hdl.handle.net/11234/1-2607`.

6.2. Experiments

We train and evaluate our model on the created dataset and compare its performance to two baseline methods. The same model hyperparameters as described in Section 5.1 are used, except for the RNN cell dimension, which is 500.

Training was performed in two phases. First, each language specific model was trained on particular CommonCrawl (Web) training set for approximately four days. Then, each model was fine-tuned with a smaller learning rate 0.0001 on respective Wikipedia training set for three more days. Finally, as all models seemed to be continuously improving on the development sets, we took the last model checkpoints for evaluating.

Both baseline methods and language models were trained on concatenation of Wikipedia and CommonCrawl training data. For language model training, we considered only those {2,3,4,5}-grams that occurred at least twice, and used default KenLM options.

To measure model performance, modified word error accuracy is used. The alpha-word accuracy considers only words that consist of at least one alphabetical character, because only these can be potentially diacritized. The testing set results of *Lexicon* and *Corpus* baselines, as well as of our models before and after fine-tuning, and with a language model are presented in Table 6.

As results show, our model outperforms both baselines even if no language model is used. Moreover, incorporation of the language model helps the model perform better as well as does model fine-tuning. Without fine-tuning, all models but the Romanian outperform baselines. We suspect that the reason why the Romanian model before fine-tuning performs worse than the *Corpus* method is that non-standard Web data differ too much from standard data from Wikipedia. It is also an interesting fact that the biggest error reduction is at Vietnamese and Romanian which seem to be most difficult for both baseline methods.

7. Conclusion

In this work, we propose a novel combination of recurrent neural network and a language model for performing diacritics restoration. The proposed system is language agnostic as it is trained solely from parallel corpora of texts without diacritics and diacritized texts. We test our system on two existing datasets comprising of four languages, and we show that it outperforms previous state-of-the-art systems. Moreover, we propose a pipeline for generating consistent multilingual diacritics restoration datasets, run it on twelve languages, publish the created dataset, evaluate our system on it and provide a comparison with two baseline methods. Our method outperforms even the stronger contextual baseline method on the new dataset by a big margin.

Future work includes detailed error analysis, which could reveal types of errors made by our system. Since certain words may be correctly diacritized in several ways given the context of the whole sentence, such error analysis could also set the language specific limit on the accuracy that can be achieved. Further, when designing our multilingual dataset we decided to use testing sets with sentences from Wikipedia articles. This was well motivated as we wanted it to contain sentences with proper diacritics. However, such testing sets contain mainly standard language and are thus worse for comparison of models aiming to generate diacritics for non-standard language. Therefore, we plan to create additional development and testing sets in the future work.

While experimenting with the model on Czech we found out that when it is trained to output instructions (e.g. add caron) instead of letters, it performs better. Future work thus also includes thorough inspection of this behavior when applied to all languages.

Finally, the system achieves better results when a language model is incorporated while inferring. Because the use of an external model both slows down the inferring process and

Language	Wiki sentences	Web sentences	Words with diacritics	Lexicon	Corpus	Our model w/o finetuning	Our model	Our model + LM	Error reduction
Vietnamese	819 918	25 932 077	73.63%	0.7164	0.8639	0.9622	0.9755	0.9773	83.33%
Romanian	837 647	16 560 534	24.33%	0.8533	0.9046	0.9018	0.9799	0.9837	82.96%
Latvian	315 807	3 827 443	39.39%	0.9101	0.9457	0.9608	0.9657	0.9749	53.81%
Czech	952 909	52 639 067	41.52%	0.9590	0.9814	0.9852	0.9871	0.9906	49.20%
Polish	1 069 841	36 449 109	27.09%	0.9708	0.9841	0.9891	0.9903	0.9955	71.64%
Slovak	613 727	12 687 699	35.60%	0.9734	0.9837	0.9868	0.9884	0.9909	44.21%
Irish	50 825	279 266	26.30%	0.9735	0.9800	0.9842	0.9846	0.9871	35.55%
Hungarian	1 294 605	46 399 979	40.33%	0.9749	0.9832	0.9888	0.9902	0.9929	58.04%
French	1 818 618	78 600 777	14.65%	0.9793	0.9931	0.9948	0.9954	0.9971	58.11%
Turkish	875 781	72 179 352	25.34%	0.9878	0.9905	0.9912	0.9918	0.9928	24.14%
Spanish	1 735 516	80 031 113	10.41%	0.9911	0.9953	0.9956	0.9958	0.9965	25.57%
Croatian	802 610	7 254 410	12.39%	0.9931	0.9947	0.9951	0.9951	0.9967	36.92%

Table 6: Results obtained on new multilingual dataset. Note that the alpha-word accuracy presented in the table is measured only on those words that have at least one alphabetical character. The last column presents errror reduction of our model combined with language model compared to the Corpus method.

requires significantly more memory, it would be desirable to train the model in such way that no additional language model is needed. We suspect that multitask learning (e.g. training the model also to predict next/previous letter) may compensate for the absence of a language model.

8. Acknowledgements

The research described herein has been supported by the Grant No. DG16P02R019 of the Ministry of Culture of the Czech Republic. Data has been used from and stored into the repository of LINDAT/CLARIN, a large research infrastructure supported by the Ministry of Education, Youth and Sports of the Czech Republic under projects LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781.
Thanks to NGỤY Giang Linh for the information and examples on Vietnamese.

9. Bibliographical References

Adali, K. and Eryiğit, G. (2014). Vowel and diacritic restoration for social media texts. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pages 53–61.

Azmi, A. M. and Almajed, R. S. (2015). A survey of automatic arabic diacritization techniques. *Natural Language Engineering*, 21(3):477–495.

Belinkov, Y. and Glass, J. (2015). Arabic diacritization with recurrent neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2281–2285.

Buck, C., Heafield, K., and Van Ooyen, B. (2014). N-gram counts and language models from the common crawl. In *LREC*, volume 2, page 4.

Ginter, F., Hajič, J., Luotolahti, J., Straka, M., and Zeman, D. (2017). CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, pages 5–6.

Hajič, Jan and Bejček, Eduard and Bémová, Alevtina and Buráňová, Eva and Hajičová, Eva and Havelka, Jiří and Homola, Petr and Kárník, Jiří and Kettnerová, Václava and Klyueva, Natalia and Kolářová, Veronika and Kučová, Lucie and Lopatková, Markéta and Mikulová, Marie and Mírovský, Jiří and Nedoluzhko, Anna and Pajas, Petr and Panevová, Jarmila and Poláková, Lucie and Rysová, Magdaléna and Sgall, Petr and Spoustová, Johanka and Straňák, Pavel and Synková, Pavlína and Ševčíková, Magda and Štěpánek, Jan and Urešová, Zdeňka and Vidová Hladká, Barbora and Zeman, Daniel and Zikánová, Šárka and Žabokrtský, Zdeněk. (2018). *Prague Dependency Treebank 3.5*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Křen, M., Bartoň, T., Cvrček, V., Hnátková, M., Jelínek, T., Kocek, J., Novotná, R., Petkevič, V., Procházka, P., Schmiedtová, V., et al. (2010). Syn2010: žánrově vyváženỳ korpus psané češtiny. *Ústav Českého národního korpusu FF UK, Praha*.

Ljubešic, N., Erjavec, T., and Fišer, D. (2016). Corpus-based diacritic restoration for south slavic languages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA)(may 2016)*.

Majliš, M. (2011). W2c–web to corpus–corpora.

Nguyen, K.-H. and Ock, C.-Y. (2010). Diacritics restoration in vietnamese: letter based vs. syllable based model. *PRICAI 2010: Trends in Artificial Intelligence*, pages 631–

636.

Nivre, J., Agić, Ž., Ahrenberg, L., et al. (2017). Universal Dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, Prague.

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *ICML (3)*, 28:1310–1318.

Richter, M., Straňák, P., and Rosen, A. (2012). Korektor-a system for contextual spell-checking and diacritics completion. In *COLING (Posters)*, pages 1019–1028.

Rychlý, P. (2012). Czaccent–simple tool for restoring accents in czech texts. *RASLAN 2012 Recent Advances in Slavonic Natural Language Processing*, page 85.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

The Unicode Consortium. (2017). *The Unicode Standard, Version 10.0.0*. The Unicode Consortium, Mountain View, CA.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Xie, Z., Avati, A., Arivazhagan, N., Jurafsky, D., and Ng, A. Y. (2016). Neural language correction with character-based attention. *arXiv preprint arXiv:1603.09727*.

Yarowsky, D. (1999). A comparison of corpus-based techniques for restoring accents in spanish and french text. In *Natural language processing using very large corpora*, pages 99–120. Springer.

Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinková, S., Hajič jr., J., Hlaváčová, J., Kettnerová, V., Urešová, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., de Paiva, V., Droganova, K., Martínez Alonso, H., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Fernandez Alcalde, H., Strnadova, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonça, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics.

Ensemble Romanian Dependency Parsing with Neural Networks

Radu Ion, Elena Irimia, Verginica Barbu Mititelu
Research Institute for Artificial Intelligence, "Mihai Drăgănescu"
13 "Calea 13 Septembrie", Bucharest 050711, Romania
{radu, elena, vergi}@racai.ro

Abstract

SSPR (Semantics-driven Syntactic Parser for Romanian) is a neural network ensemble parser developed for Romanian (a Python 3.5 application based on the Microsoft Cognitive Toolkit 2.0 Python API) that combines the parsing decisions of a varying number (in our experiments, 3) of other parsers (MALT, RGB and MATE), using information from additional lexical, morpho-syntactic and semantic features. SSPR outperforms the best individual parser (MATE in our case) with 1.6% LAS points and it is in the same class with the top 5 Romanian performers at the CONLL 2017 dependency parsing shared task. The train and test sets were extracted from a Romanian dependency treebank we developed and validated in the Universal Dependencies format. The treebank, used in the CONLL 2017 Romanian track as well, is open licenced; the parser is available on request.

Keywords: parsing, treebank, neural networks.

1. Introduction

A reliable, fast and freely available syntactic parser for Romanian was needed (by both linguists and computer scientists working in Natural Language Processing) when we decided to develop SSPR. Even if other attempts at creating Romanian parsers were documented before, none proved to be accessible or performant enough: Călăcean and Nivre (2009) reported a dependency parser based on MaltParser (Nivre et al., 2007) and trained on a small treebank (Hristea and Popescu, 2003), whose impressive results, 88.6% labelled attachment score (LAS) and 92% unlabelled attachment score (UAS), are expectable given the shortness and syntactic simplicity of the sentences in the data set; the Fips constituency rule-based parser was adapted for Romanian (Seretan et al., 2010) and is available for online use (http://www.latl.unige.ch/), but no evaluation is provided for this language; Colhon and Cristea (2016) focus only on noun phrases, using patterns to identify relations inside such structures; the dependency parser of Perez et al. (2016) reported 78.04% LAS when trained on an 8000 sentences treebank. Dumitrescu et al. (2017) RBG-based parser obtained a LAS of 79.44% in the CONLL 2017 Romanian track but new developments of Boroș and Dumitrescu (2018) employ a Bidirectional LSTM network that is still training at the time of the writing and whose foreseen results are much better.

In a nutshell, the SSPR ensemble parser we have developed works by training a feed-forward neural network (NN) to recognize "good" vs. "bad" dependency relations from the minimum spanning tree (MST) of the directed graph of all existing (as many as possible) parses for the same sentence. For the training part, each edge of the graph is weighted by the inverse of the size of the set of parsers that found that edge (the "majority voting" combination strategy). The aim of the SSPR ensemble parser is to improve on the baseline "majority voting" combination of dependency parses by using the additional, lexical, morpho-syntactic and semantic features of the words participating in a relation, coupled with information related to the state of the partially completed parse.

2. Working data

The train/dev/test data that the SSPR NN used to train and evaluate came from the Romanian treebank annotated in Universal Dependencies (CONLL-U) format, named RoRefTrees (Barbu Mititelu et al., 2016b). It is a relatively large treebank (9523 sentences, 218365 words) with complex sentences (23 tokens/sentence on average) and a large coverage: it contains several text genres (literature - 1818 sentences, law - 1606 sentences, medical - 1210 sentences, FrameNet translations - 1092 sentences, academic writing - 950 sentences, news - 933 sentences, science - 362 sentences, Wikipedia - 251 sentences, miscellanea - 1302 sentences). The corpus, consistently annotated and manually validated, is freely available as a release at universaldependencies.org and it was used as training/development/testing data for Romanian in the CONLL 2017 dependency parsing shared task. Additionally, it can be queried online with user-friendly tools: http://clarino.uib.no/iness/page?page-id=iness-main-page, http://bionlp-www.utu.fi/dep_search, http://lindat.mff.cuni.cz/services/pmltq/#!/home.

3. Features

3.1 Linguistic Features

SSPR started as a project that aimed to elude errors typical to statistical parsing by employing additional semantic information. The standard practice in machine learning/statistical approaches to parsing is to use *morpho-syntactic* (POS and/or morpho-syntactic descriptions) and *lexical* (lemmas/word forms of the words and word embeddings computed from large raw corpora) features of the words to be linked and of those in a context (*n*-word windows) around the targeted words. Also, *first-order syntactic* features (extracted directly from the dependency relations themselves) and *second-order syntactic* features (the partial analysis of a word, when already linked in the partial tree) complete the linguistic information usually exploited for automatic parsing. However, additional features, whenever available, are easy to integrate in data-driven systems and our initial intuition was that *semantic* features, like wordnet relations, sub-categorization frames and semantic classes can increase parsing performance. These features were provided by external programs and supplementary resources (e.g. an inventory of subcategorisation frames of verbs occurring in RoRefTrees, semi-automatically developed and hand validated by linguists). Moreover, information about constituency of noun, verb, adjectival and adverbial phrases was available through TTL (Ion,

2007), a tool that does the tokenization, lemmatisation and POS-tagging of the corpus.

The set of linguistic features that SSPR used to improve the "majority voting" classifier is:

- Sets of semantically-related words, derived by automatically clustering word embeddings (the feature *SemClass*) extracted from the CoRoLa corpus (Tufiș et al., 2016);
- Manually validated subcategorisation frames for all the main verbs in the treebank, enriched with semantic restrictions on the arguments (Barbu Mititelu et al., 2016a) (the features *VFrame*, identifying the frame, and the *VArg*, identifying the associated argument(s), restricted to specified Romanian WordNet (Tufiș and Barbu Mititelu, 2014) senses);
- Lexical chains between the words in the sentence (Ion and Ștefănescu, 2009): a lexical chain between a word w_1 and a word w_2 is a path in the Romanian WordNet that seeks to get from a specific sense of the word w_1 to the relevant sense of the word w_2 by following semantic relations in wordnet (the feature *LXC*);
- Morpho-syntactic features provided by TTL (the features *Type*, *Gender*, *Case*, etc.);
- Continuous constituents provided by TTL (the feature *Chunk*).

3.2 The individual parsers and their performances

The following open-source parsers were trained and evaluated using all the available linguistic features (see the scores marked with "+") and using only the standard, lexical and morpho-syntactic features: 1) MALT parser with the LIBSVM classifier and with the "arc-eager" algorithm; 2) RBG parser with the standard, projective analysis model (Lei et al., 2014); 3) MATE parser with "default" parameters (Bohnet, 2010).

Parsers performance was evaluated in a standard, 10-fold cross validation manner (with 80% training data, 10% development data and 10% test data) and Table 1 shows the average scores on the 10 different runs.

The evaluation measures we employed were: 1) **UAS**, which gives the number of relations identified in the test set (the head and the dependent are both correctly identified), but it is not concerned with the relation type (subject, object, etc.); 2) **LAS**, which also considers the dependency relations label; 3) **UAS/LAS without punct (UASp, LASp)**, which refers to the scores UAS and LAS as defined above, but ignoring the dependency relation that links the punctuation (**punct**, in our relation inventory); 4) **Relaxed LAS (LASr)**, which refers to LAS as defined above, but ignoring the relations specific to the Romanian treebank, like "nmod:tmod", which is a temporal noun modifier. This relation is an adaptation of the "nmod" relation, universally defined within the Universal Depenedency project (de Marneffe et al., 2014), i.e. valid for all natural languages. The project allows for language-specific relations, but only as subtypes of the universal ones and labelled accordingly. Thus, "nmod:tmod" is a language-specific relation, a subtype of the "nmod" relation, namely a temporal "nmod" (nominal modifier). All the relations of the type "universal:particular" are automatically reduced to "universal"; in our example, if the parser detects the relation "nmod", it is classified as correct when compared to "nmod:tmod"; 5) **Relaxed LAS without punct (LASpr)**, which is the relaxed LAS score, as defined above, that also ignores the dependency relation that links punctuation (it is the most permissive LAS score of all).

Table 1 shows little but consistent improvement of all the parsers performances when using the additional syntactic-semantic features over the standard scenarios (see LAS+ vs. LAS, UAS+ vs. UAS, etc.). We underlined the best values for each score and MATE parser superior yield is visible for all measures, except UASp+ (where the difference to the best parser's performance is insignificant). It is important to note that we could obtain these improvements in the various UAS/LAS scores of *all the three* parsers only when using *all* of our supplementary features. Otherwise, there are certain subsets that work well for some parsers but degrade performance for other parsers. We thus think that it is important to optimize the supplementary feature set according to the parsers one wants to combine, such that the supplementary feature set provides improvements for all the parsers of the ensemble.

Score	MALT	RBG	MATE
LAS	0.7978	0.7906	0.809
LAS+	0.8038	0.8091	0.8138
UAS	0.8544	0.8603	0.8668
UAS+	0.859	0.8693	0.8707
UASp	0.8639	0.8706	0.8758
UASp+	0.8683	0.8808	0.88
LASp	0.7992	0.7913	0.8096
LASp+	0.8054	0.8116	0.8152
LASr	0.813	0.805	0.8229
LASr+	0.8185	0.8225	0.8276
LASpr	0.8165	0.8077	0.8256
LASpr+	0.822	0.8271	0.8309

Table 1: The contribution of supplementary syntactic-semantic features to the UAS and LAS average scores for MALT, RBG and MATE parsers.

4. SSPR implementation

SSPR is a feed-forward neural network (NN) that learns to recognize a "correct" syntactic dependency relation vs. an "incorrect" one, from the MST "Majority Voting" combination strategy of multiple parses, taking into account its context as modelled by linguistic and structural features. There is no limit on the number of different parsers that can be combined by this NN, but it is of utmost importance that the parsers are as different as possible, from the analysis algorithm point of view, so that the reunion of all the dependency relations of a

sentence, generated by all available parsers (i.e. the directed graph of all existing parses of the sentence), should cover the correct analysis of the sentence. This way, we can make sure the combination algorithm will have a pool of choices from which to produce a correct analysis.

SSPR is a Python 3.5 application that uses Microsoft Cognitive Toolkit 2.0 Python API to train and run the NN whose structure is presented in Figure 1. The input vector *X* is a binary vector and has N positions (corresponding to N values for all the features available for one training example, i.e. a dependency relation). There are two hidden layers, the first with 100 neurons (*H*), and the second with 10 neurons (*G*). The activation function for both hidden layers is *tanh*. The structure was obtained experimenting with NNs having 0, 1, or two hidden layers, and with 10, 50, 100, 500 or 1000 neurons in a hidden layer. The NN in Figure 1 was the one that gave the best results. The NN outputs a real vector of two positions $\hat{Y}$, which is a probability distribution of the two possible states (correct/incorrect), computed with the *softmax* function.

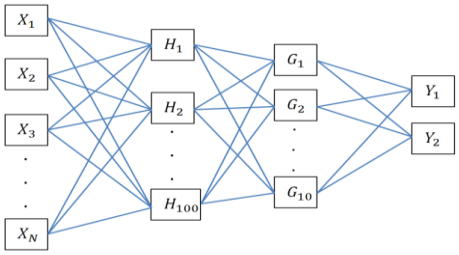


Figure 1: SSPR neural network

The training examples for SSPR NN are acquired as follows:

- **Step 1.** For each sentence in the training set, all the individual parsers were run and a directed graph A was constructed with the reunion of all the resulting analyses. The nodes of the graph A are lexical units, while the edges are dependency relations. Each edge is labelled with the name of the corresponding dependency relation and with the name of the parser that discovered it;
- **Step 2.** If a relation *r* is identified between the words *HEAD* and *DEP* by more than one parser, then all the edges between *HEAD* and *DEP* labelled *r* are replaced by a single edge, which bears the label *r* and the set V of the parsers that discovered *r*;
- **Step 3**. From the directed graph A, a Minimum Spanning Tree (MST) is constructed, using Chu-Liu-Edmonds algorithm (Chu & Liu, 1965; Edmonds, 1967) where the weight of each edge is 1/|*V*|. In other words, we computed the tree that covers all the sentence and whose dependency relations are discovered simultaneously by as many of the individual parsers as possible (i.e. the real number 1/|*V*| is minimised). Steps 1-3 are reproducing the "Majority Voting" (MV, Surdeanu and Manning, 2010) strategy of combining independently-trained parsing models;
- **Step 4.** The MST constructed at step 3 contains both correct and incorrect relations (given that the natural language may pose complicated parsing problems that none of the parsers available knows how to solve). Each relation in this tree is transformed in a training example *(X, Y)* for the SSPR NN, where *X* is the binary feature vector (see below) and *Y* is a binary vector with 2 positions computed by checking if the relation is also present in the gold-standard train set, which guarantees its correctness (*Y*= [1; 0] for a correct relation and [0; 1] for an erroneous one).

For each dependency relation that has to be classified as correct or incorrect, we construct and then concatenate in the input vector *X* the following "one-hot" vectors: 1) The binary vector of the parsers that identified the relation; 2) The binary vector of the morpho-syntactic properties of the dependent in the relation; 3) The binary vector of the label of the relation; 4) The binary vector of the dependent's lemma; 5) The binary vector of the syntactic-semantic features of the relation; 6) The binary vectors obtained by concatenating the vectors obtained at steps 1-5 for the regent of the relation, for the leftmost child of the dependent and for the regent of the regent.

SSPR NN is then trained to minimise the average (over the train set) of the cross entropy of the *Y* and $\hat{Y}$ vectors, standing for the ground-truth value and the estimated value with *softmax*.

At run-time, the SSPR parser ensemble works as follows:

- **Step 1.** All individual (the same as in training) parsers are run on a new test sentence and the directed graph A of all individual analyses is constructed;
- **Step 2.** The NN is applied on each edge a_i in A and the probability vector is computed as $\hat{Y}$ = [*P*(*correct*|*X*); *P*(*incorrect*|*X*)];
- **Step 3.** A Minimum Spanning Tree is constructed on the graph A, but this time the weight of each edge is given by the first position in the vector Y as computed by NN, namely by *P*(*correct*|*X*).

In this manner, SSPR tries to improve on the MV combination strategy by proposing more informed weights for the MST construction.

5. SSPR Evaluation

To demonstrate the improvements that the SSPR parser brings in comparison with the MV ensemble baseline, we first compute a breakdown of the LAS score of the MV ensemble as a function of the number of individual parsers that find the correct dependency relation, averaged over our 10-fold cross validation test run. Table 2 shows the components of LAS for our three parsers MV ensemble, namely the LAS when all three parsers agree on the correct dependency relation (see the "3/3" row in Table 2) and the LAS when two out of three parsers agree on the correct dependency relation ("2/3"). When only one

parser finds the correct dependency relation ("1/3"), its LAS, in this case, contributes to the "oracle" LAS score, which is the maximum possible score that can be achieved by any ensemble of these three parsers.

3/3 LAS+	**0.7218**
2/3 LAS+	**0.0968**
MV LAS+	**0.8186**
1/3 LAS+	**0.068**
ORACLE LAS+	**0.8866**

Table 2: A breakdown of the LAS score as a function of how many parsers found the correct dependency relation

Table 3 shows the SSPR scores compared to the MATE scores, the best individual parser from Table 1.

Score	MATE average	SSPR average
LAS+	**0.8138**	**0.83**
UAS+	**0.8707**	**0.8844**
UASp+	**0.88**	**0.8928**
LASp+	**0.8152**	**0.8307**
LASr+	**0.8276**	**0.8434**
LASpr+	**0.8309**	**0.8461**

Table 3: Comparison between MATE and SSPR (averages over 10 train/dev/test random samples)

The evaluation setting of SSPR is the same as the one described in Section 3.2. The individual parsers were trained on the train set, using all the available syntactic-semantic features, and were run on the development set. The neural network SSPR was trained on the development set combining the three individual parsers outputs and was tested, together with each individual parser, on the test set. As Surdeanu and Manning (2010) note, our approach in combining independent parses of the same sentences can be described, as they put it, as "a meta-classifier that selects candidate dependencies based on their likelihood of belonging to the correct tree." Furthermore, they assert that this strategy did not offer significant gains over the standard, "unweighted" MV but, as far as our experiments on Romanian dependency parsing go, Tables 2 and 3 show that the SSPR meta-classifier is able to improve the LAS of the MV ensemble by 1.1% (from 81.86% to 83%). We can explain this significant increase as follows:

- Surdeanu and Manning use seven parsers in their ensemble but six out of these seven parsers are, in fact, the same parser (the MALT parser) with different linking strategies. We postulate that these six flavours of the MALT parser output very similar parse trees such that the case where three or less parsers (the minority subset) agree on the correct dependency is poorly represented;
- We use three different parsers, each with its own modelling of the parsing problem, such that the case where only one out of the three parsers (the minority subset) finds the correct dependency relation adds a *significant* 6.8% to the LAS of the MV ensemble. It is this pool of correctly found dependency relations from which SSPR is able to extract the 1.1% improvement of the MV ensemble LAS;
- Lastly, we think that our NN-based meta-classifier is superior to the L2-regularized logistic classifier that Surdeanu and Manning used to combine parses, mainly because we rely on a richer set of features that also include semantic features and second-order features.

Table 3 shows that the SSPR NN ensemble parser is 1.6% LAS points and 1.5% LASpr points better than the best individual parser **(**MATE**)**. The medium LASpr score of 84.61% places SSPR in the same class with the best parsers in CoNLL-2017 task for Romanian (see Figure 2). Even though our evaluation setting is not identical to the one for CONLL 2017, the working corpus is the same and our 10-fold cross validation strategy provides a more robust result, *for Romanian*, than the CONLL 2017 1-fold run (the CONLL final parser ranking is backed up by the runs in various languages).



Figure 2: LASpr scores in CONLL 2017 parsing shared task

The LAS medium score of 83% reflects the general performance of SSPR, regardless of its domain of application, because the treebank we worked on is balanced and contains texts from various domains. Table 4 shows the variation of LAS according to the domain. It can be noted that juridical, medical and scientific (from mathematics, physics and computer science) texts have over the medium scores: in their case, it is relatively simple to identify the elements between which a relation is very probable and this relation type is easy to predict (or machine learn): the topic is quite stable, ellipses are rare (a consequence of the need for clarity), predicates saturate their valences locally by words from predictable semantic classes. At the opposite pole, the literature domain, even if it does not contain long sentences, raises the most parsing problems: the authors' creativity manifests both in the sentence structure (dislocated arguments, unlexicalised ones, unusual word order, etc.) and in unusual combination of words (with spectacular

stylistic results), namely words from unexpected semantic classes taken as arguments.

SSPR	Average
Belletrist	**0.7839**
Juridical	**0.8694**
Medical	**0.8571**
Academic	**0.8332**
Encyclopaedic	**0.8001**
Journalistic	**0.8373**
Scientific	**0.8812**
Miscellanea	**0.8053**

Table 4: LAS for SSPR according to the domain

Conclusions

We presented evidence that Romanian dependency parsing can be improved by using "better informed" ensembles of individual dependency parsers, in our case in the form of a neural network that learns to distinguish correct links from incorrect ones. The improvement margin can only increase when adding more, *as different as possible*, parsers into the mix that uses our supplementary linguistic features and we plan to extend this work by adding more such parsers to the ensemble. Specifically, we need to see what parsers from the CONLL 2017 shared task are open-sourced and readily available such that new ways of modelling dependency parsing could be of benefit to our ensemble parsing. One such example is the best parser from the CONLL 2017 shared task on Romanian, namely the Stanford parser which is a NN-based dependency parser.

Another direction we could take to improve ensemble parsing (or individual parsing for that matter) would be to better generalize a dependency relation. Our approach of using semantic features worked to a certain degree but, since NNs work well with "embeddings", we can think of constructing "dependency relation embeddings" as real-valued vectors that are computed with recursive NNs.

Bibliografical References

Barbu Mititelu, V., Ion, R., Simionescu, R., Scutelnicu, A., Irimia E. (2016a) Improving parsing using morpho-syntactic and semantic information, in *Revista Romana de Interactiune Om-Calculator 9(4)*, 285-304, 2016.

Barbu Mititelu, V., Ion, R., Simionescu, R., Irimia, E., Perez C.-A. (2016b) The Romanian Treebank Annotated According to Universal Dependencies., Proceedings of The Tenth International Conference on Natural Language Processing (HrTAL2016), Dubrovnik, Croatia, 29 September – 1 October 2016

Bohnet, B. (2010). Very High Accuracy and Fast Dependency Parsing is not a Contradiction. The 23rd International Conference on Computational Linguistics (COLING 2010), Beijing, China.

Boroș T. and Dumitrescu, S.D., Multilingual tokenization and part-of-speech tagging. Lightweight versus heavyweight algorithms, to be published in Lecture Notes in Artificial Intelligence, 2018.

Călăcean, M., Nivre, J. (2009) A Data-Driven Dependency Parser for Romanian, Proceedings of the Seventh International Workshop on Treebanks and Linguistic Theories, 65-76.

Chu, Y. J.; Liu, T. H. (1965). "On the Shortest Arborescence of a Directed Graph", *Science Sinica*, 14: 1396–1400

Colhon, M., Cristea, D. (2016). Dependency Parsing within Noun Phrases with Pattern-based Approaches, Proc. of the 12th Int. Conf. Linguistic Resources and Tools for Processing the Romanian Language, 51-60.

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., Manning, Ch. D. (2014). Universal Stanford Dependencies: A cross-linguistic typology. In *Proceedings of LREC*, 4585-4592.

Dumitrescu, S. D., Boroș T., Tufiș, D. (2017). RACAI's Natural Language Processing pipeline for Universal Dependencies, Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, August, 2017, Vancouver, Canada, Association for Computational Linguistics, 174-181.

Edmonds, J. (1967), "Optimum Branchings", *J. Res. Nat. Bur. Standards*, 71B: 233–240.

Hristea, F.,Popescu, M. (2003) A Dependency Grammar Approach to Syntactic Analysis with Special Reference to Romanian. In F. Hristea and M. Popescu (eds.), *Building Awareness in Language Technology*, Bucharest, University of Bucharest Publishing House, 9-16.

Ion, R. and Ștefănescu, D. Unsupervised Word Sense Disambiguation with Lexical Chains and Graph-based Context Formalization. In Proceedings of the 4th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics (Vetulani, Zygmunt). November 2009, Poznan, Poland, pp. 190-194,

Ion, Radu. (2007). Word sense disambiguation methods applied to English and Romanian, PhD Thesis, Romanian Academy.

Lei, T., Xin, Y., Zhang, Y., Barzilay R. and Jaakkola, T. (2014) Low-Rank Tensors for Scoring Dependency Structures. ACL 2014.

Nivre, J., J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov and E. Marsi (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2), 95-135.

Perez, A.C., Mărănduc, C., Simionescu, R. (2016) Including Social Media – A Very Dynamic Style – in the Corpora for Processing Romanian Language. In: Trandabăț D., Gîfu D. (eds) Linguistic Linked Open Data. RUMOUR 2015. *Communications in Computer and Information Science*, vol 588. Springer, Cham, 139-153.

Seretan, V., Wehrli, E., Nerima, L., Soare, G. (2010) FipsRomanian: Towards a Romanian Version of the Fips Syntactic Parser, Proceedings of LREC 2010, 1972-1977.

Surdeanu, M. and Manning, C. D. (2010). Ensemble Models for Dependency Parsing: Cheap and Good? In Proceedings of the HLT 2010 Human Language Technologies: The 2010 Annual Conference of the

North American Chapter of the Association for Computational Linguistics, 649-652.

Tufiș, D., Barbu Mititelu, V., Irimia, E., Dumitrescu, S.D., Boroș, T. (2016) The IPR-cleared Corpus of Contemporary Written and Spoken Romanian Language. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). Portorož, Slovenia, pp. 2516-2521, May 2016

Tufiş, D., Barbu Mititelu, V. (2014) The Lexical Ontology for Romanian, in Nuria Gala, Reinhard Rapp, Nuria Bel-Enguix (Ed.), *Language Production, Cognition, and the Lexicon*, series Text, Speech and Language Technology, vol. 48, Springer, p. 491-504

Classifying Sluice Occurrences in Dialogue

Austin Baird, Anissa Hamza, Daniel Hardt
UCSC, LiLPa/Unistra, Copenhagen Business School/UCSC
arbaird@ucsc.edu, anissa.hamza@etu.unistra.fr, dhardt@ucsc.edu

Abstract

Ellipsis is an important challenge for natural language processing systems, and addressing that challenge requires large collections of relevant data. The dataset described by Anand and McCloskey (2015), consisting of 4100 occurrences, is an important step towards addressing this issue. However, many NLP technologies require much larger collections of data. Furthermore, previous collections of ellipsis are primarily restricted to news data, although sluicing presents a particularly important challenge for dialogue systems. In this paper we classify sluices as Direct, Reprise, Clarification. We perform manual annotation with acceptable inter-coder agreement. We build classifier models with Decision Trees and Naive Bayes, with accuracy of 67%. We deploy a classifier to automatically classify sluice occurrences in OpenSubtitles, resulting in a corpus with 1.7 million occurrences. This will support empirical research into sluicing in dialogue, and it will also make it possible to build NLP systems using very large datasets. This is a noisy dataset; based on a small manually annotated sample, we found that only 80% of instances are in fact sluices, and the accuracy of sluice classification is lower. Despite this, the corpus can be of great use in research on sluicing and development of systems, and we are making the corpus freely available on request. Furthermore, we are in the process of improving the accuracy of sluice identification and annotation for the purpose of created a subsequent version of this corpus.

Keywords: sluicing, ellipsis, dialogue

1. Introduction

Ellipsis is a major challenge for NLP systems, as well as an important topic in theoretical linguistics. The most extensive empirical work to date on ellipsis is described in Anand and Hardt (2016) and Anand and McCloskey (2015). This work involves a corpus of some 4100 sluice occurrences, extracted from the NYTimes Gigaword Corpus. These occurrences have been manually annotated in a detailed fashion.

Sluices are elliptical questions, where all but the interrogative phrase of a question is omitted, leaving a wh-word remnant, as in the following example, with the sluice wh-word in bold (Anand and Hardt, 2016):

(1) Harry traveled to southern Denmark to study botany. I want to know **why**.

In this paper, we construct a very large corpus of sluice occurrences in dialog. We build on previous work (Anand and McCloskey, 2015; Fernández et al., 2004; Fernández et al., 2007) in developing methods to automatically identify and classify sluice occurrences. We apply these methods to the English portion of OpenSubtitles, resulting in a corpus of over 1.7 million sluice occurrences. This is orders of magnitude larger that any previous collections of ellipsis occurrences and it has been automatically annotated with linguistically relevant features.

2. Related Work

(Fernández et al., 2004; Fernández et al., 2007) describe an approach to the classification of sluice occurrences in the British National Corpus (BNC). Fernandez et al. focus on what they call bare sluices: utterances in dialog consisting of only a wh-word (they also consider the form *which N*). They extract 5343 bare sluices from the dialogue transcripts of the BNC. Fernández et al. (2004) classify dialogue sluices as follows.

Feature	Description
sluice	type of sluice
mood	declarative or non-declarative
polarity	positive or negative
frag	fragment or not
quant	presence of a quantified expression
deictic	presence of a deictic pronoun
proper_n	presence of a proper name
pro	presence of a pronoun
def_desc	presence of a definite description
Wh	presence of a wh-word
overt	presence of other potential antecedent expression

Table 1: Features

Direct: the sluice queries for additional information that was explicitly or implicitly quantified away in the previous utterance.

Reprise: The utterer of the sluice cannot understand some aspect of the previous utterance which the previous speaker assumed as presupposed.

Clarification: the sluice used to ask for clarification about the previous utterance as a whole.

Wh-anaphor: the antecedent is a *wh*-phrase.

(They also use a category **Unclear**, which we will ignore.) Fernandez et al. build models to classify sluice occurrences, using the above five-way classification scheme. They define the features as shown in Table 1: the first is the type of sluice; the other features all apply to the antecedent utterance.

A total of 351 data points were used to train the classifiers. Table 2 gives the distribution of these data points by the

Sluice	Direct n (%)	Reprise n (%)	Clarification n (%)	Wh-anaphor n (%)
What	7 (9.60)	17 (23.3)	17 (23.3)	1 (1.3)
Why	55 (68.7)	24 (30.0)	0 (0)	1 (1.2)
Who	10 (13.0)	65 (84.4)	0 (0)	2 (2.6)
Where	31 (34.4)	56 (62.2)	0 (0)	3 (3.3)
When	50 (63.3)	27 (34.1)	0 (0)	2 (2.5)
Which	1 (8.3)	11 (91.6)	0 (0)	0 (0)
whichN	19 (21.1)	71 (78.8)	0 (0)	0 (0)
How	23 (79.3)	3 (10.3)	3 (10.3)	0 (0)
Total	106(30.2)	203(57.8)	24 (6.8)	18 (5.1)

Table 2: Sluice Cats and Wh Types

Reading	Recall	Precision	F1
Direct	71.70	79.20	75.20
Reprise	85.70	83.70	84.70
Clarification	100.00	68.60	81.40
Wh anaphor	66.70	100.00	80.00
weighted score	81.47	82.14	81.80

Table 3: BNC Sluice Classification

wh-word and classification (Fernández et al. (2007), table 3).

Four machine learning classifiers were run on this dataset annotated with the 11 features, with weighted f-scores ranging from 73.24 - 81.62. Table 3 shows the results obtained by the most accurate learner (Fernández et al., 2007) (Appendix A)

3. The Data

3.1. Opensubtitles

The English portion of Opensubtitles (http://www.opensubtitles.org/)[1] contains 2,125,277,188 words and 327,968,003 lines. Building on methods described in (Anand and McCloskey, 2015), we locate both root sluices and embedded sluices. As explained in (Anand and McCloskey, 2016) a root sluice is unembedded (2), while non-root sluices are "sub-parts of larger structures", as in (3).

(2) A: We should go home. B: Why/when/what for/how?

(3) The university has to change, but it's not clear in what ways.

In order to locate sluices, the entire corpus was first POS tagged using the Stanford POS tagger, described in (Toutanova et al., 2003). We define two regular expressions to search for sluices in the corpus. The first identifies embedded sluices with a pattern including an embedding verb

[1]Pierre Lison and Jörg Tiedemann, 2016, OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)

WH word	root count	embedded count
What	1,097,382	14,421
Why	352,047	29,805
How	122,256	6,453
Who	98,330	2,335
Where	70,312	2,677
When	29,171	1,473
Which	18,491	308
Whom	8,874	60

Table 4: Root vs. embedded sluice in Opensubtitles

followed by a wh-word. The wh-word is optionally followed by an adjective, adverb, preposition, or noun. Following this is an optional punctuation, followed by end-of-string. This pattern defines the following list of embedding verbs: *know, knew, ask, say, understand, wonder, remember, tell, explain, imagine, care, forget, worry*. The second pattern applies to lines that did not match the first pattern. It is the same as the one described above, except that it does not contain an embedding verb and the sentence must end with a '?'.
Of the sluices found in the corpus, a total of 57,532 were embedded sluices and a total of 1,796,863 were root sluices. Table 4 gives the breakdown of root and embedded sluices by wh word.

3.2. Annotating Sluice Types

We construct two samples of sluice occurrences for the purpose of manual annotation: the first includes the first 100 root sluices. Since the distribution of wh-words is quite unbalanced, we construct a more balanced sub-corpus, which includes 1000 randomly selected examples of what, and 500 of each remaining wh-word – why, how, who, where, when, which, whom – making up a total of 4500 examples. We used four categories, following Fernández et al. (2007), with the following revised definitions:

Direct questions an indefinite part of the antecedent that is implicitly or explicitly expressed, and is not necessarily known by the speaker.
A: He didn't come.
B: **Why?**
A: Break up.

Clarification questions the entire antecedent, typically expressing surprise or confusion.
A: Captain ! It 's the Tomb of Heroes !
B: **What?**
How can it be?

It also includes illocutionary uses of *wh*-words as in:
A: Congratulations on your promotion !
B: Should I thank you ?
A: **Why ?**

Sluices lacking a linguistic antecedent are also classified under Clarification.
A: It 's Colonel Gelovani.

Sluice Type	Agreement
Clarification	87.9%
Direct	83.8%
Reprise	61.5%

Table 5: Intercoder Agreement for Opensubtitles Sluices

B: Yes .
A: **What ?**

Reprise addresses a definite and explicit part of the antecedent. The questioned element is definitely known to the speaker.
A: They made her mad.
B: **Who ?**
A: The devils

None occurrences are not in fact sluices. These are often due to incorrect POS labels, or frozen questions which typically occur in spontaneous and oral discourse.
A : She teases him a lot .
B : That 's natural in a girl
A : Yes , I suppose so . What about Claudius ?

3.3. Interannotator Agreement

The three authors of this paper individually annotated the same sample of 100 sluices, resulting in 84% average agreement. The kappa score is 80%.
Out of the sample, 51% of the sluices were unanimously classified as Clarification, 8% as Reprise, 26% as direct, and 3% as None. The agreement rates for sluices are given in table 5. (The agreement rate for None was 100%)
Although Reprise sluices were the least frequent in the sample (besides None), they had the highest amount of disagreement. In all disagreements involving a Reprise sluice, the alternative classification by the disagreeing annotator was Direct. For all disagreements involving a Clarification sluice, the disagreeing annotator always annotated as Direct as well. These disagreements overwhelmingly occurred in sluices containing a single 'what?', where the preceding and succeeding context was needed in order to determine the type. All instances in which all three annotators disagreed on the sluice type are not included for the percentage calculations. Due to the relatively high overall agreement among the authors, a single author annotated all the samples used in training the classifier for this paper.

4. Predictive Model

4.1. Training Data

Two sets of training data were used in building classifiers, as shown in Table 6. Set2 roughly matches the distribution of classes in OpenSubtitles, while Set1 is more balanced.
The decision tree classifiers are built using scikit-learn (Pedregosa et al., 2011), and the Naive Bayes classifiers are using nltk (Bird et al., 2009).
The features used to train the were identical to nine of the features used by the authors of Fernández et al. (2007). All of the features described in section 2 are used except for frag and overt. All of the features, except for 'type', take on boolean values. The value for 'type' is the wh word contained in the sluice. Unlike the features in Fernández et al. (2007), there is no distinction between WhichN and Which for the classifier used in this paper.
The separate datasets were used to train both a NaiveBayes classifier and a Decision Tree classifier. Both classifiers have accuracies scored using 10-fold cross validation. In what follows, we focus on the Decision Tree classifier results on the balanced dataset, Set1, as these were the best results.

	Set 1					Set 2				
	C	D	R	N	Total	C	D	R	N	Total
What	674	126	54	107	961	711	126	54	109	1000
Why	126	224	109	4	463	68	203	77	2	350
How	0	380	28	15	423	0	105	13	2	120
Who	0	40	191	45	276	0	28	63	9	100
Where	0	20	86	66	172	0	47	14	9	70
When	0	0	56	30	86	0	20	8	2	30
Which	0	10	33	35	78	0	1	15	4	20
Whom	0	0	43	30	73	0	9	0	1	10
Total	800	800	600	332	2532	779	539	244	136	1700

Table 6: Two Training Datasets

class	Precision	Recall	f1-score
clar	0.71	0.91	0.80
dir	0.80	0.81	0.81
rep	0.86	0.74	0.79
none	0.85	0.53	0.65
avg / total	0.80	0.75	0.78

Table 7: Decision Tree Set1

4.2. Classifier Results

Table 7 shows the results using the decision tree classifier with Set1. The majority baseline results are shown in Table 8.
This classifier beats the majority baseline overall and performs relatively well in most areas. However, it has a very low recall when identifying None type sluices. We suspect that this is because other features are relevant to identifying this class.

5. Classifying All OpenSubtitles Sluices

The decision tree classifier was used to classify all of the sluices detected in OpenSubtitles. The number of classifications assigned to the sluices are shown in Table 9. The

class	Precision	Recall	f1-score
clar	0.31	1.00	0.47
dir	0.00	0.00	0.00
rep	0.00	0.00	0.00
none	0.00	0.00	0.00
avg / total	0.10	0.31	0.15

Table 8: Baseline Set1

Class	Amount	Percentage
Clarification	1,110,210	61.8%
Direct	379,420	21.1%
Reprise	226,568	12.6%
None	80,665	4.5%

Table 9: Resulting Dataset

	Clar	Dir	Rep	None
What	1,059,912	13,578	2,947	19,759
Why	50,298	232,768	69,006	98
How	0	120,498	1,411	422
Who	0	2,514	87,911	7,938
Where	0	7,188	29,120	34,033
When	0	0	21,674	7,539
Which	0	2,874	8,539	7,088
Whom	0	0	5,960	3,788
Total	1,110,210	379,420	226,568	80,665

Table 10: Class by wh-word in OpenSubtitles

type of sluice as a percentage of all sluices detected are also shown in this table. Note that the total number of sluices includes those that the classifier classified as None.

Table 9 gives the resulting dataset, broken down by class; this is further broken down by wh-word in Table 10.

A random sample of 103 examples classified by the model were selected and hand annotated to compute the classifier's accuracy on this sample. Table 11 shows two sets of percentages about the classifier's predictions. First, of all the sluices in categories Direct, Clarification, and Reprise, the percentage of which are actually sluices (not annotated as being of the None class). Second, of the sluices categorized as Direct, Clarification, or Reprise, what percent are correct.

Table 11 shows that overall, 80% of the examples that the classifier predicted to be a sluice were actually sluices, and 67% were categorized correctly by the classifier.

6. Conclusion

Ellipsis is an important challenge for natural language processing systems, and addressing that challenge requires large collections of relevant data. The dataset described by Anand and McCloskey (2015), consisting of 4100 occurrences, is an important step towards addressing this issue. However, many NLP technologies require much larger collections of data. Furthermore, previous collections of ellipsis are primarily restricted to news data, although sluicing presents a particularly important challenge for dialogue systems.

Predicted Class	True Sluices	Correctly Categorized
clar	0.81	0.76
dir	0.69	0.61
rep	0.57	0.57
Total	0.80	0.67

Table 11: Classifier Accuracy Results

In this paper we present an ellipsis corpus with 1.7 million occurrences. This will support empirical research into sluicing in dialogue, and it will also make it possible to build NLP systems using very large datasets. This is a noisy dataset; based on a small manually annotated sample, we found that only 80% of instances are in fact sluices, and the accuracy of sluice classification is lower. Despite this, the corpus can be of great use in research on sluicing and development of systems, and we are making the corpus freely available on request. Furthermore, we are in the process of improving the accuracy of sluice identification and annotation for the purpose of created a subsequent version of this corpus.

7. Acknowledgments

Thanks to three reviewers for helpful comments. We gratefully acknowledge Pranav Anand and Jim McCloskey for their work on sluicing annotation and analysis on the project *The Implicit Content of Sluicing*, which provides the inspiration and background for the current work. This research has been sponsored by NSF grant 1451819.

8. Bibliographical References

Anand, P. and Hardt, D. (2016). Antecedent selection for sluicing: Structure and content. In *EMNLP*, pages 1234–1243.

Anand, P. and McCloskey, J. (2015). Annotating the implicit content of sluices.

Anand, P. and McCloskey, J. (2016). Annotation guide.

Fernández, R., Ginzburg, J., and Lappin, S. (2004). Classifying ellipsis in dialogue: A machine learning approach. In *Proceedings of the 20th international conference on computational linguistics*, page 240. Association for Computational Linguistics.

Fernández, R., Ginzburg, J., and Lappin, S. (2007). Classifying non-sentential utterances in dialogue: A machine learning approach. *Computational Linguistics*, 33(3):397–427.

Toutanova, K., Klein, D., Manning, C., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL 2003*, pages 252–259.

9. Language Resource References

Bird, Steven and Loper, Edward and Klein, Ewan. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python*.

Collection of Multimodal Dialog Data and Analysis of the Result of Annotation of Users' Interest Level

Masahiro Araki[1], Sayaka Tomimasu[1], Mikio Nakano[2], Kazunori Komatani[3], Shogo Okada[4], Shinya Fujie[5], Hiroaki Sugiyama[6]

[1]Kyoto Institute of Technology, [2]Honda Research Institute Japan Co., Ltd., [3]Osaka University, [4]Japan Advanced Institute of Science and Technology [5]Chiba Institute of Technology, [6]Nippon Telegraph and Telephone Corporation

araki@kit.ac.jp, tomi@ii.is.kit.ac.jp, nakano@jp.honda-ri.com, komatani@sanken.osaka-u.ac.jp, okada-s@jaist.ac.jp, shinya.fujie@p.chibakoudai.jp, sugiyama.hiroaki@lab.ntt.co.jp

Abstract

The Human-System Multimodal Dialogue Sharing Corpus Building Group is acting as a working group of SIG-SLUD for the purpose of constructing a corpus for evaluating elemental technologies of the multimodal dialogue system. In this paper, we report the results of recording chat dialogue data between a human and a virtual agent by the Wizard of OZ method conducted in 2016, and the results of the analysis of annotations of users' interest level in the data.

Keywords: Multimodal dialogue corpus, users' interest level, dialogue system

1. Introduction

Recently, multimodal dialogue systems that utilize not only spoken language but also image and other modalities are getting much attention, thanks to improvements in language, speech, and image processing techniques and their underlying machine learning technologies as well as the advancement of hardware such as computers, sensors, display, and robots. However, how such systems should use multimodal information when communicating with humans is still under investigation. One of the reasons is that there are not enough shared multimodal dialogue data with annotations referring to users' intentions, emotions, attitudes, and dialogue situations. Since a lot of effort is needed for annotation, it is desirable that multiple research institutions collaborate to annotate.

With this background, we initiated an activity to build a shared corpus of human-system multimodal dialogues[1]. So far, a number of shared corpora of multimodal dialogues among humans have been built and there has been plenty of work on the analysis of these corpora (e.g., (Carletta, 2007; Janin et al., 2003)). In contrast, the goal of this project is to contribute to component technology development for multimodal dialogue systems. Therefore, we focused on analyzing how humans behave toward multimodal dialogue systems, rather than analyzing human-human dialogues. In human-system dialogues, user behaviors are different from human behaviors in human-human dialogues because the users realize that they are talking to a system. By collecting data of human-system dialogues, we can analyze user behaviors to contribute to improve components of dialogue systems.

We collected multimodal dialogues between a user and a virtual agent, which was operated by the Wizard of Oz method, and annotated them with labels indicating whether the user is interested in the topic or not. We used a virtual agent because we consider virtual agents and robots are promising natural user interfaces as users would feel easier in talking to them than to devices without human-like characters, and virtual agents are easier to use in data collection than robots.

This paper reports our multimodal dialogue data collection protocol (Sec. 2.) and annotation method (Sec. 3.), and discusses future data collection and annotation through the results of the analysis of annotations (Sec. 4.). Then, by considering the differences with related work (Sec. 5.), we list issues related to sharing the corpus that we will build with the research community (Sec. 6.).

2. Multimodal dialogue corpus

We implemented an environment for recording multimodal dialogue between a human and a virtual agent (Tomimasu and Araki, 2016). The virtual agent [2] was manipulated by an operator via the Wizard of Oz (WoZ) method, which facilitates data collection. A human so-called "Wizard" simulates a dialogue system that interacts with the human users via the GUI interface shown in Figure 1. The interface features a topic selector, a selectable list of typical utterances for each topic, and a selectable list of general responses in chat dialogue.

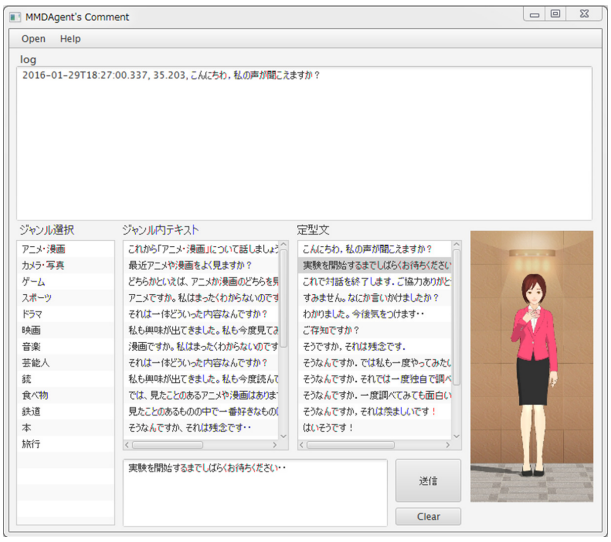
Before the data collection, the participants assessed their interest in 12 topics. During the data collection, six topics were selected as dialogue theme from both the favorite and non-favorite groups of topics. The operator tried to follow the dialogue engagement of the participants by selecting the utterance with the different initiative. The operator also limited the number of exchanges for each topic to approximately 10. The behavior of the participants was recorded via a video camera and Microsoft Kinect ®V2 sensor.

An example of the dialogue is shown in Figure 2 (it is originally Japanese that translated to English.).

In April 2016, data from four people (hereafter referred to as data 1) were experimentally recorded and the problems in the recording environment were examined. At this time,

[1]This activity is being conducted by the Human-System Multimodal Dialogue Sharing Corpus Building Group of SIG-SLUD of the Japanese Society for Artificial Intelligence.

[2]http://www.mmdagent.jp/



Copyright 2009-2013 Nagoya Institute of Technology (MMDAgent Model “Mei”)

Figure 1: GUI for Wizard.

```
S: Let's talk about railway.
U: Yes.
S: Do you like trains?
U: Well, not so much.
S: Then do you often take a train?
U: I have not taken the train recently.
S: In what kinds of situations do you ride
   the train?
U: Well, when going out of the prefecture
   ... and when I go out for a drink.
   I cannot use my motorcycle.
```

Figure 2: An example of dialogue

a trial annotation of participants' interest level in the topics was carried out by members of the working group. We made improvements such as randomizing the order of presenting topics and clarifying instructions to participants; in January 2017 data from 10 people (hereafter data 2) were recorded.

Regarding data 2, after the preliminary annotation by three annotators and improvement of the annotation manual based on their work, a release version of the annotation was created by another three annotators.

In the annotation of chat dialogue between a person and a dialogue system, we assigned one of the following labels to each exchange: interest (o), unknown (t), and no interest (x) to each exchange (i.e., a pair of ”system utterance S” and ”user utterance U”). The assessment was done by considering the various sources of participant's information, such as facial expressions, prosody, speech content, etc.

3. Annotation of users' interest level

The estimation of users' interest level in a dialogue topic enables the system to capture the users' preferences, which is essential when developing user-adapted dialogue systems. In this section, we analyze our corpus with the annotation of users' interest level to examine the reliability of the annotations and to improve the annotation manual.

Table 1: Annotation results

Label	First	Second
Interested	907	992
Unknown	162	267
Not interested	1276	1108
Error	22	0
Fleiss' kappa	0.407	0.490

The specific procedure of the analysis is as follows. At first, three graduate students studying human-computer interaction annotated data 2 (10 persons, 789 interactions) with very intuitive instructions. Through the analysis and discussion about differently judged interactions, we developed an annotation manual. Then we compared the first round of annotations with the annotations carried out by three new annotators, who received the annotation manual for instruction.

In the first round of annotations, the graduate students intuitively judged whether the participant was interested in the topic or not. It should be noted that they judged the presence of the interest in the current dialogue topics, not whether the participants enjoyed the dialogue or not. The column labeled *First* in Table 1 shows the results of the first annotation. Here, the simple hearing back actions were annotated as errors.

Since we controlled the number of *interested* and *not interested* topics based on the preliminary survey, we expected the number of annotated labels (*interested* and *not interested*) to be almost the same; however, the number of *not interested* labels is higher. In the first annotation, Fleiss' kappa was 0.407, which is interpreted as between Fair agreement and Moderate agreement.

We analyzed the criteria of the annotators in cases where they did not agree and looked for possible causes of the disagreement as follows.

- At the beginning of the dialogue, some annotators tried to judge a dialogue as *interested* or *not interested* using little information, but others thought those interactions should be judged as *unknown*.
- Some annotators judged interactions based on only part of the interactions, others based their judgment on the entire interaction.
- Some annotators always utilized single or a few modalities such as smile instead of the contents of the interaction.
- Some annotators consider that the criteria of the labels are consistent among participants, but the others consider that the criteria differ for each participant.

According to the results of this analysis, we developed an annotation manual with the indications that follow, and had three annotators who work at another research institute carry out a second round of annotations.

- The beginning of the dialogue should be labeled as *unknown* if you are not completely sure.

- The judgment should be done based on the whole interaction instead of a part of it.
- Annotators firstly watch the whole video and understands the participant's habits of emotional expressions before the judgment.
- The interaction should be labeled *Error* when the interaction is a completely unexpected one.

The column labeled *Second* in Table 1 shows the results of the second round of annotations.
In the second annotation, Fleiss' kappa improved to 0.490, which is interpreted as Moderate agreement.
Our experiment shows that, even with a difficult problem such as judging the presence or absence of interest in an unfamiliar situation as in human-system dialogue, we can obtain reliable annotations with a well-designed annotation manual.

4. Analysis of label distribution

The subjectivity of coders influences the annotation of participants' interest level in the dialog. Data 1 was collected for analysis of the subjectivity of coders and the label distribution. In this section, we analyze the distribution of annotated labels by calculating the level of agreement (κ coefficient) between coders of data 1. A total of eight coders annotated the interest level for four participants who participated in the experiments. For each exchange (interaction), the coders were asked to annotate the labels: interest (o), unknown (t), and no interest (x).
Fleiss's κ (κ_f) was calculated as the agreement between the eight coders. The agreement was 0.26, which is considered low. The agreement between each pair of coders was calculated to analyze the reasons for the low agreement. Cohen's κ (κ_c) was calculated for each pair of eight coders (A1-A8). Figure 3 shows the matrix of κ_c among coders. Although the κ_c was less than 0.3 in almost all cases, the agreement between some pairs was more than 0.4 (Moderate agreement). Furthermore, we analyzed the similarity of annotations between coders through hierarchical clustering with Ward's method using κ_c as a distance measure. The similarity between one coder and the other coders was also calculated as the average of κ_c based on the matrix (Figure 3).
Figure 4 shows the dendrogram that denotes the clustering results and average κ_c. From the figure, coders A5 and A7 annotated interest level in a different manner compared to the other coders because the averages of A5 and A7 were the lowest and second lowest, respectively, among all the coders. The clustering analysis results show that the annotation task was influenced by the coders' subjectivity. It also clarified the dissimilarity of label distributions among coders.

5. Related work

For analyzing human-human multimodal conversations, several meeting conversation corpora with multiple participants have been released and shared such as the AMI (Augmented Multi-party Interaction) corpus (Carletta, 2007) and the ICSI meeting corpus (Janin et al., 2003). The CHIL

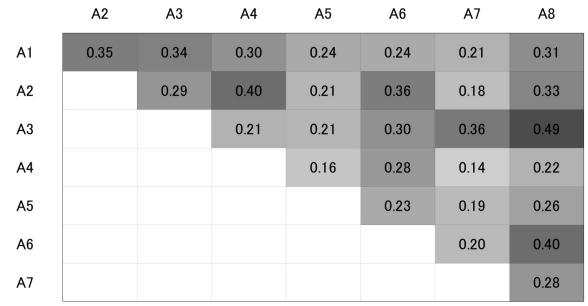


Figure 3: Cohen's kappa (κ_c) between each pair of coders

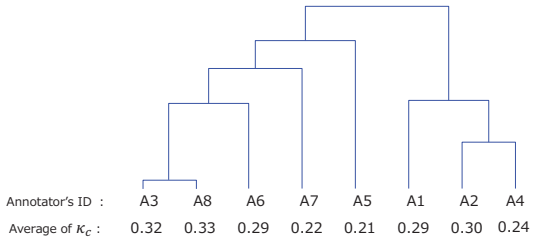


Figure 4: Clustering results of annotators based on Cohen's kappa (κ_c)

(Computers in Human Interaction Loop) treats human-human interactions in offices and classrooms (Waibel and Stiefelhagen, 2009), and VACE (Video Analysis and Content Extraction) treats human-human interactions in battle-game sessions in the air force (Chen et al., 2006). A corpus of political debates in a TV program has also been shared for analyzing social interactions (Vinciarelli et al., 2009).

Several multimodal corpora, which are not those of interactions, have also been published, such as the ones for assessing public speaking ability and anxiety (Chollet et al., 2016) and for recognizing group-level emotion on in-the-wild data (Dhall et al., 2017). Systems based on such multimodal analyses have been constructed; for example, a system that analyzes nonverbal human behaviors was developed and used to assess indicators of psychological distress, such as depression and post-traumatic stress disorder (PTSD) (Stratou and Morency, 2017).

Our goal in this project is to contribute to component technology development for multimodal dialogue systems. Therefore, our target is to construct dialogue corpora; more specifically, those are not of human-human dialogues but of human-system dialogues. The most important difference between human-human and human-system dialogues is whether users realize that they are talking to a system. User behaviors differ when they talk to a system or a human. Data on human-system dialogues contain real behaviors of users, which can be used to predict user actions in actual dialogue systems. Furthermore, dialogues have inherent characteristics such as the fact that verbal interactions are composed of multiple turns and the existence of certain dialogue states. In the WoZ system currently used for our data collection, dialogues can be divided by topics

presented by the system, which can be regarded as one of the dialog states. Collecting such dialogue data can lead to a novel system design that will not easily bore users by considering such states.

In terms of sharing the human-system dialogue corpus, there is an ongoing project to collect and share a text chat corpus, which also conducts shared tasks using it (Higashinaka et al., 2015). Currently, a competition for speech-input chatbots, the Amazon Alexa Challenge[3], is also being conducted. The target of our project is not text-input or speech-input chatbots but multimodal dialogue systems.

Constructing dialogue robots is one of the ultimate goals of dialogue system research, and thus has been investigated by many researchers (Bohus and Horvitz, 2009; Al Moubayed et al., 2012; Sugiyama et al., 2015; Matsuyama et al., 2015; Lala et al., 2016). Some studies focused on the detection of user interests to enable a system to adapt topics to user preferences (Hirayama et al., 2011; Chiba et al., 2014; Tomimasu and Araki, 2016). Our corpus can be used for developing and improving such systems.

6. Issues on privacy and future work

The collected multimodal dialogue data includes the personal information of participants, such as their faces and voices. For the purpose of privacy protection, several fundamental rules for treating personal data have been established by the Japanese government. Thus, the following issues should be addressed prior to sharing the corpus, even for research purposes.

- Careful discussions are required on what kind of research will be done prior to data collection since the participants have to sign an agreement on the range of the data use.
- It is preferable that participants be allowed to reject the agreement even after they have signed it. Thus, the data should be distributed by an agency that specializes in research data distribution.
- Since crowdsourcing cannot be used for data annotation because of these limitations, the research group should be maintained continuously to preserve the quality of the collected data.

Currently, the working group is discussing the abovementioned issues in order to release the collected data. We are planning to share our corpus with the annotation results by multiple annotators, in the same way as the text chat corpus was shared (Higashinaka et al., 2015). Since the user's interest level to be annotated in our project is based on the subjective judgments of annotators, the annotation process is not to give a reference label that can be uniquely determined, and thus differences among individual annotators are inevitable. Therefore, the corpus will also be used for studying how to handle such subjective annotation results and how to apply such labels to dialogue system research.

[3]https://developer.amazon.com/alexaprize

7. Bibliographical References

Al Moubayed, S., Skantze, G., Beskow, J., Stefanov, K., and Gustafson, J. (2012). Multimodal multiparty social interaction with the furhat head. In *Proceedings of the 14th ACM International Conference on Multimodal Interaction (ICMI)*, pages 293–294.

Bohus, D. and Horvitz, E. (2009). Models for multiparty engagement in open-world dialog. In *Proceedings of Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 225–234.

Carletta, J. (2007). Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

Chen, L., Rose, R. T., Qiao, Y., Kimbara, I., Parrill, F., Welji, H., Han, T. X., Tu, J., Huang, Z., Harper, M., Quek, F., Xiong, Y., McNeill, D., Tuttle, R., and Huang, T. (2006). Vace multimodal meeting corpus. In *Proceedings of the Second International Conference on Machine Learning for Multimodal Interaction (MLMI05)*, pages 40–51.

Chiba, Y., Ito, M., Nose, T., and Ito, A. (2014). User modeling by using bag-of-behaviors for building a dialog system sensitive to the interlocutor's internal state. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 74–78, June.

Chollet, M., Wortwein, T., Morency, L.-P., and Scherer, S. (2016). A multimodal corpus for the assessment of public speaking ability and anxiety. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA).

Dhall, A., Goecke, R., Ghosh, S., Joshi, J., Hoey, J., and Gedeon, T. (2017). From individual to group-level emotion recognition: EmotiW 5.0. In *Proc. International Conference on Multimodal Interaction (ICMI)*, pages 524–528, New York, NY, USA. ACM.

Higashinaka, R., Funakoshi, K., Araki, M., Tsukahara, H., Kobayashi, Y., and Mizukami, M. (2015). Towards taxonomy of errors in chat-oriented dialogue systems. In *Proceedings of Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 87–95, September.

Hirayama, T., Sumi, Y., Kawahara, T., and Matsuyama, T. (2011). Info-concierge: Proactive multi-modal interaction through mind probing. In *The Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC 2011)*.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., and Wooters, C. (2003). The ICSI meeting corpus. In *Proceedings of IEEE International Conference on Acoustics, Speech & Signal Processing (ICASSP)*, pages I–364–I–367, April.

Lala, D., Milhorat, P., Inoue, K., Zhao, T., and Kawahara, T. (2016). Multimodal interaction with the autonomous android ERICA. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction (ICMI)*, pages 417–418.

Matsuyama, Y., Akiba, I., Fujie, S., and Kobayashi, T. (2015). Four-participant group conversation: A facilitation robot controlling engagement density as the fourth participant. *Computer Speech & Language*, 33(1):1 – 24.

Stratou, G. and Morency, L.-P. (2017). Multisense—context-aware nonverbal behavior analysis framework: A psychological distress use case. *IEEE Transactions on Affective Computing*, 8(2):190–203, April.

Sugiyama, T., Funakoshi, K., Nakano, M., and Komatani, K. (2015). Estimating response obligation in multi-party human-robot dialogues. In *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 166–172, Seoul, South Korea.

Tomimasu, S. and Araki, M. (2016). Assessment of users' interests in multimodal dialog based on exchange unit. In *Proceedings of the Workshop on Multimodal Analyses Enabling Artificial Agents in Human-Machine Interaction*, MA3HMI '16, pages 33–37, New York, NY, USA. ACM.

Vinciarelli, A., Dielmann, A., Favre, S., and Salamin, H. (2009). Canal9: A database of political debates for analysis of social interactions. In *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–4, Sept.

Waibel, A. and Stiefelhagen, R. (2009). *Computers in the Human Interaction Loop*. Springer Publishing Company, Incorporated, 1st edition.

Recognizing Behavioral Factors while Driving: A Real-World Multimodal Corpus to Monitor the Driver's Affective State

Alicia Lotz[1], **Klas Ihme**[2], **Audrey Charnoz**[3], **Pantelis Maroudis**[4], **Ivan Dmitriev**[5], **Andreas Wendemuth**[1]

[1] Otto-von-Guericke-Universität Magdeburg, D-39106 Magdeburg, {firstname}.{surname}@ovgu.de

[2] German Aerospace Center, D-38108 Braunschweig, klas.ihme@dlr.de

[3] École polytechnique fédérale de Lausanne, CH-1015 Lausanne, audrey.charnoz@epfl.ch

[4] VALEO Comfort & Driving Assistance Systems, F-93012 Bobigny, pantelis.maroudis@valeo.com

[5] Institut VEDECOM Du Véhicle Décarboné et Communicant et de sa Mobilité, F-78000 Versailles, ivan.dmitriev@vedecom.fr

Abstract

The presented study concentrates on the collection of emotional multimodal real-world in-car audio, video and physiological signal recordings while driving. To do so, three sensor systems were integrated in the car and four relevant emotional states of the driver were defined: neutral, positive, frustrated and anxious. To gather as natural as possible emotional data of the driver, the subjects needed to be unbiased and were therefore kept unaware of the detailed research objective. The emotions were induced using so-called Wizard-of-Oz experiments, where the drivers believed to be interacting with an automated technical system, which in fact was controlled by a human. Additionally, on board interviews while driving were conducted by an instructed psychologist. To evaluate the collected data, questionnaires were filled out by the subjects before, during and after the data collection. These included monitoring of the drivers perceived state of emotion, stress, sleepiness and thermal sensation but also detailed questionnaires on their driving experience, attitude towards technology and big five OCEAN personality traits. Afterwards, the data was annotated by expert labelers. Exemplary results of the evaluation of the experiments are given in the result section of this paper. They indicate that the emotional states were successfully induced and the annotation results are consistent for both performed annotation approaches.

Keywords: in-car emotions, multimodal corpus, multimodal interaction, affective computing, natural emotions

1. Introduction

While driving in a car, the driver can be affected by various emotionally challenging situations. They can either be triggered by the current driving situation, e.g. being cut off by another driver, or caused by a personal event, e.g. receiving good news. On the one hand, emotions can effect the driving behavior in positive and negative ways. By sensing fear, the driver is able to perceive a situation as a possible risk and adapt his driving towards the situation, while anger may lead to an underestimation of the risk level and therefore may increase the risk of causing an accident (Lu et al., 2013). On the other hand, positive as well as negative emotions can influence the driving performance in a negative way (Pêcher et al., 2009; Rhodes and Pivik, 2011; Taubman-Ben-Ari, 2012). Pêcher et al. show that positive emotions (by listening to music) with high intensity lead the participants to drive with risky behavior (e.g. degraded lateral control, sudden speed decreases). In a similar way, Taubman-Ben-Ari shows that positive emotions (with raw intensity) tempt people to drive in a reckless manner. However, positive emotions are rarely considered in terms of road safety, as their occurrence is less common compared to negative emotions (Lewis et al., 2007).

Especially negative emotions such as anger can seriously influence the driving behavior. Already slight provocation can lead to aggressive, violent and hostile driving and can result in road rage. Road rage or aggressive driving is a syndrome of frustration-driven behaviors, enabled by the driver's environment (Shinar, 1998). Frustrating situations, as traffic congestion or delays, can lead to anger emotions in a driving context that conclude in aggressive driving (Shinar, 1998; Zhang et al., 2015). Road rage can be expressed in extenuated ways such as verbal abuse or headlight flashing, but also in more dangerous ways such as traffic weaving, tailgating or aggressive braking (Garase, 2006).

Another negative emotion which should be addressed is fear. This includes anxious drivers who are afraid of driving itself, for example caused by stressful situations while driving, but also anxiety induced by driving in a self-driving/autonomous car. In this study we will focus on the second cause. A survey from the American Automobile Association revealed that 3/4 of U.S. drivers reported to feel afraid to drive in a self-driving car (American Automotive Association, 2017). A reason of this new form of driving anxiety is the fact that recent car functionalities (e.g. adaptive cruise control, lane keeping, self-parking system) take over more and more control from the driver. This transmits a feeling of not being in control of the situation (Koo et al., 2015).

To mitigate this negative safety impact of emotionally affected drivers, we aim to develop a driver monitoring system which detects emotional states of the driver and is able to take over control in critical situations. End users' surveys have shown, that potential users are willing to hand over the control to the car in safety critical situation (e.g. bad weather or road conditions), as well as for comfort reasons (e.g. traffic jam or road works) (Willstrand et al., 2017). 64% of the users were positive to get informed and warned by the car when being in a critical driver state, 16% would even consider a full handover of the control to the car. In the study presented here a special focus will be drawn on the impact of negative emotions. Therefore, a suitable database

is needed, aligned to the needs of this research objective. The presented study will focus on four emotional states of the driver: neutral, positive, frustration and anxiety. For each of these emotional states a specific experiment is designed. The developed system should be able to distinguish between these four types of emotional drivers and draw a conclusion on the capability of the driver to safely interact in road traffic.

2. Related work

There are various studies concentrating on the appearance of affects while driving (e.g. (Grimm et al., 2007a) and (Eyben et al., 2010)). Available datasets are mostly limited to the application of in-car speech recognition but not designed to evaluate the driver's emotional state. These are for example the AV@CAR Spanish multichannel multimodal corpus for in-vehicle automatic audio-visual speech recognition (Ortega et al., 2004) and AVICAR audio-visual speech corpus in car environment (Lee et al., 2004). The number of corpora consisting of natural real-world in-car emotional speech data is still limited, a corpus of multimodal audio-visual and physiological data is yet unknown. Available corpora concentrate either on already existing well known emotional datasets like the Berlin Database of Emotional Speech (Burkhardt et al., 2005), the Danish Emotional Speech corpus (Engbert and Hansen, 1996) or the eNTERFACE'05 Audio-Visual Emotion Database (Martin et al., 2006) and additive real car noises in different conditions as presented in (Grimm et al., 2007b), or focus only on basic modalities such as separate video and audio (Tawari and Trivedi, 2010) or physiological signals (Katsis et al., 2008). Other authors like Abdić et al. and Malta et al. focus on the evaluation of frustration but leave other car-related emotional states disregarded (Abdić et al., 2016; Malta et al., 2011; Ihme et al., in press).

The main disadvantage of most of the presented works is the disregard of in-car acoustics. By superimposing noise, the acoustic characteristic of the car is being left unconsidered. This cannot be compensated by additively overlaying real in-car noise recordings as presented in (Jones and Jonsson, 2007), as the replaying of car noises through stereo speakers differs significantly from real in-car acoustics. In (Lotz et al., 2018) it was shown that it is also not sufficient to replay standard emotional corpora in a real car environment, as the SNR and classification performance can differ significantly for different recording setups of the original dataset. Botinhao et al. investigate the effect of different speaking styles, noise levels and listener age on speech intelligibility. Further, not only different in-car acoustics influence the speech signals' quality significantly but also different types of route taken, weather conditions, background noise in the car and whether the windows are open or closed (Botinhao and Yamagishi, 2017). Furthermore, also the movement of the car needs to be considered. This effects the mounting of all considered sensors and will lead to random noise in the recorded signals.

A big advantage of the developed data set is that all recordings were done in a real-world in-car environment while driving. By not only letting the subject perform small interaction tasks with car assistant systems, which will not lead to pure natural speech, but by also conducting conversation-like interviews, we obtain more natural information considering the driver's speech and facial expressions. By inducing natural emotions, we are able to also consider physiological signals for further analysis.

3. Test Environment

3.1. Test Vehicle and Environment

The test vehicle of the data collection was the research vehicle FASCar (Fischer et al., 2014). The FASCar is a test vehicle for testing driver assistance systems and automated driving functions. It is equipped with a unique steer-by-wire system to support innovative haptic feedback and intervention strategies. For the safety driver/co-driver an additional brake pedal is available (DLR, 2017).

All experiments were carried out on the DLR test ground at the DLR compound in Braunschweig, Germany, which is a designated test ground for driving experiments. As the test vehicle is not permitted to drive on open road, this ensured the most natural driving experience and driving environment for the participants, comparable to road traffic in quiet residential areas. On the site, driving is allowed with a maximum speed of 30 km/h on a fixed driving course (see Figure 1). One round course of roughly 900 meters on the available streets in the site took approx. 2.5 min. To ensure comparability of all recordings, the data was collected during day light and under similar and constant weather conditions. Termination criteria for in-car audio recordings were strong rain and/or thunderstorm. In addition to the participant, two further persons were in the car, one investigator sitting on the passenger seat, and one technician for the supervision of the sensor data recording sitting on the rear bench behind the passenger seat.

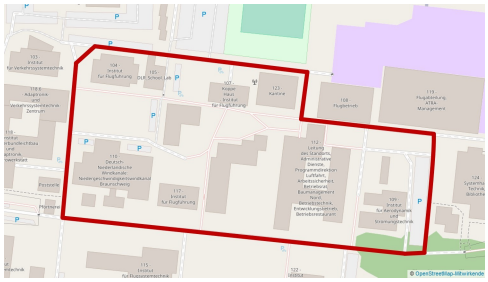


Figure 1: Driving course at DLR compound in Braunschweig, Germany (map taken from `https://www.openstreetmap.de/`).

3.2. Sensor integration

The test vehicle was equipped with a microphone, video and physiological sensing system.

The audio speech stream was recorded using two Shure VP 82 shotgun microphones attached to the dashboard above the steering wheel and close to the right A-pillar using elastic mounting to dampen the car's movement. Additionally, to collect high quality reference recordings, a Sennheiser HSP-4 EW-3 headset microphone was worn by the driver. The microphone tracks were synchronized using a Steinberg UR44 audio interface stored in the trunk of the car. Video images were captured using a Smart Eye Pro



Figure 2: Setup of the microphone and SEP camera system (Smart Eye AB, Gothenburg, Sweden, www.smarteye.se) on the dashboard of the FASCar research vehicle.

(SEP) Multi Camera System (Smart Eye AB, Gothenburg, Sweden, www.smarteye.se) including two high resolution cameras with infrared (IR) filters and active IR illumination attached to the dashboard on both sides of the steering wheel. Peripheral physiological data was recorded using the wireless sensor system Spacebit Heally to measure the electrocardiogram (ECG) and galvanic skin response (GSR). It consists of a finger sensor and a standard 3-lead ECG wearable. The signals are transmitted via Bluetooth to a computer. The sensors integrated onto the dashboard of the test vehicle are depicted in figure 2.
All sensor systems were triggered by a time synchronous signal coming from SEP, to ensure the synchronicity of all systems.

4. Data Characteristics

4.1. Involved Participants

Data was gathered from 30 participants of one age group (25 - 40 years). All of them were native standard German speakers without speaking disorders. For safety reasons, only drivers with a valid driver's license and an annual mileage of at least 5000 km were considered. Further exclusion criteria were: Pregnancy, physical impairment, heart and/or neurological problems, partial or total deafness and medical or alcohol consumption. All participants provided written informed consent to participate in the study and received 30 €as reimbursement for participation. After the study, the participants were fully informed about the goals of the study. The procedure of the study was reviewed and approved by the ethics committee of the Otto-von-Guericke University of Magdeburg, Germany (ref.-number: 153/17).

4.2. Considered Emotional States

This study concentrates only on the driver-relevant states. Therefore, four emotional states occurring frequently while driving a car were defined. These include neutral, positive, frustrated and anxious drivers. It can be assumed that the most commonly occurring emotional state is neutral. To be able to consider a broad range of emotional states and at the same time distinguish between emotions having a highly risky impact on the driving performance, all positive emotions were summarized into the driver state "positive", while the relevant negative emotions were subdivided into "frustrated/angry" and "anxious/fearful". As the collected data will be used to train machine learning algorithms, it is not reasonable to distinguish between separate states of frustration and anger, and anxiety and fear, respectively (Devillers et al., 2005). The characteristic features of the data collected for these states is assumed to be very similar. Therefore, the algorithms will not be able to differentiate between them. The authors of this paper are aware of the fact that from a psychological point of view there is a difference between frustration and anger, and anxiety and fear, but this issue will not be addressed in this paper.
To define these states, the circumplex model of emotions concepts (Russell and Lemay, 2000) was used. This defines the neutral emotional state in a region around the origin of the valence/arousal-axis with moderate arousal and neutral valence, the positive emotional state as all positive expressions with a positive valence, and frustration and anxiety both with high arousal and negative valence (cf. Figure 3). Therefore, to be able to distinguish between frustration and anxiety, additional definitions were concluded. Frustration/anger is defined as the unpleasant feeling occurring in situations in which a person is detained from reaching a desired outcome/goal and anxiety as the unpleasant feeling of dread over anticipated negative events (Lazarus, 1991; Schmidt-Daffy, 2013). According to these definitions, scenarios for each emotional state were designed.

4.3. Emotional Scenarios

The scenarios were designed such that the driving itself would only minimally influence the emotional state of the driver. Participants drove five rounds per scenario. Each scenario started with one round as baseline. The four following rounds depended on the scenario and are described in the remainder of this section. The order of the scenarios was kept constant starting with the neutral scenario followed by positive, frustrated, and anxious. The emotions were induced by conducting experimental studies and interview-like conversations. The experimental studies were assisted by the interviewer reinforcing the situation. In the presented scenarios the co-driver was a trained psychologist who took the role of the interviewer. He did not hesitate to react to the driver's answers and ask follow up questions to keep the conversation alive. Afterwards, the driver was asked to narrate a situation where he felt the considered emotion. By memorizing/narrating an emotional experience, this emotion can be recalled by the driver

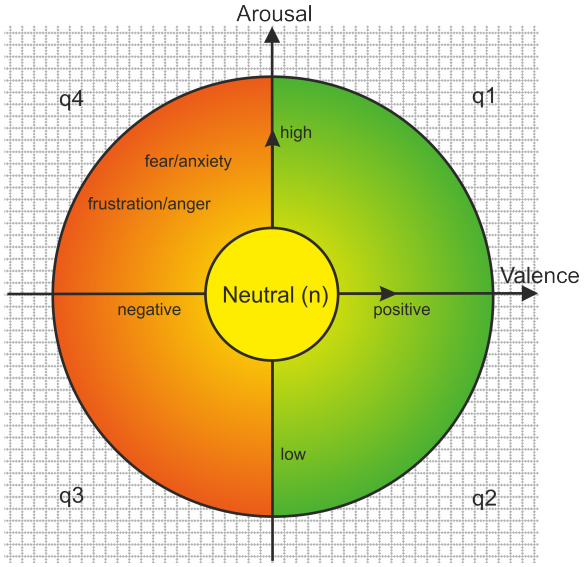


Figure 3: Defined emotional states as areas in the circumplex model of emotions concept by Russell and Lemay (2000).

and is reflected in his facial expressions, speech and physiological signals. This is a commonly used method to induce emotions also used in (Martin et al., 2006), (Amir et al., 2000) and (Forgas, 2002).

neutral: After the baseline round, the investigator initiated a conversation with the participant on a neutral topic (e.g. on educational background, basic personal information, weather etc.) which lasted roughly two rounds. In the last two rounds, the participant drove without conversation to gather baseline information on the facial expression of the driver.

positive: After the baseline round, to not reveal the research objective of the study, the participants were told that a check of the sound quality is necessary and a sound file needed to be played during the following two rounds. This sound file comprised two episodes of a funny radio podcast "Wir sind die Freeses." of the radio station NDR2 (Altenburg, 2017), which is well-known in this region of Germany. In the last two rounds, the investigator started a conversation with the participant on the show, followed up with positive topics (happy situations, holidays) and repeatedly asked the participant to narrate situations in which s/he felt positive/happy.

frustrated/angry: For this scenario, participants were told that the goal of this drive is to evaluate a speech-based navigation system, which was briefly introduced by the investigator before the drive. The participants had the task to enter a certain address and start the navigation. The participants were also told that one of the main innovations of the navigation system was its capability to recognize whether the user is talking to the system or to other people in the car. The navigation system was a mock-up created with MS Power Point, which was controlled by the technician on the rear bench in a Wizard-of-Oz (WOZ)-like scenario. Participants were asked to "think aloud" while interacting with the system, which means that they should utter any thoughts about and experiences with the systems.

The drive started again with one baseline round. Following this, the participant interacted with the navigation system for two rounds. In order to induce frustration, the technician regularly "misunderstood" certain commands of the participant and elicited wrong selections, so that the participant could not finalize the selection for a couple of minutes. After that the investigator initiated a conversation on similarly frustrating experiences, e.g. in interaction with technological systems or while driving. During the course of the conversation (for another two rounds), the investigator encouraged the participant to narrate frustrating situations.

anxious: Again the scenario started with a baseline round. Then, the participant was told that a usability evaluation of an automated brake assistant would follow, in which the participant was asked to "think aloud" again. The brake assistant would only brake at certain locations, namely when the car is about to pass two traffic cones on the side of the road. Before braking, the system would present three audio warning tones. However, similar to the previous scenario, the system was controlled by a WOZ (which was the investigator on the passenger seat who had a second brake pedal). In total, three locations on the track were marked with traffic cones. Initially, the system worked just fine and braked at the marking preceded by the warning tone. Then, however, the WOZ started to brake or play the tone without braking at random locations on the track.
The goal of this procedure was that the participant would pair the tone with the sudden brake. In the following this happened over and over again, so that the participants anticipated the negative event of the brake when the tone started, which created the anxiety of the upcoming abrupt brake. The brake assistant was "active" for two rounds. After this, the investigator initiated a conversation on similar situations or experiences in the car, in which other road participants or assistance systems behaved in a way that created anxiety or uncertainty. The conversation lasted for two rounds.

5. Assessment Methodology

5.1. Questionnaires

Before the data collection, the participants filled out a basic demographic questionnaire including questions about their driving experience. In addition, the participants had to complete the ATI-scale measuring their attitude towards technology (Franke et al., 2017) and the Big Five Inventory (BFI-10) (Rammstedt and John, 2007) to assess the big five OCEAN personality traits: openness on experience (O), conscientiousness (C), extraversion (E), agreeableness (A), neuroticism (N). Moreover, the participants' emotional baseline was assessed using the Self-Assessment Manikin (SAM) (Bradley and Lang, 1994) on the dimensions valence, arousal and dominance as well as the Geneva Emotion Wheel (GEW) (Scherer et al., 2013). For the GEW, the participants were asked to rate all emotions in the wheel (alternative 3). In addition, one item assessing the novelty of the situation was filled in by the participants. After each drive, the participants filled in the GEW plus three dummy questionnaires to camouflage the actual purpose of

the study (Karolinska Sleepiness Scale, (Akerstedt and Gillberg, 1990); Stress Scale, (Dahlgren et al., 2005); Scale of Thermal Sensation, (Gagge et al., 1967)). After all drives, the participants filled out a questionnaire asking detailed questions on their emotional experiences during the drives. These included among others the SAM scales and novelty item as well as free input on their experienced emotions during each drive.

5.2. Ground Truth

The considered emotional states were defined using the circumplex model of emotions concepts (see section 4.2.). By using the valence and arousal level to define the emotions, we could overcome the use of a garbage class in the annotation of the data. We opted for an annotation in two stages, this made the annotation process more controllable for the labelers. The annotation was conducted as followed:

1. Annotation of valence and arousal using the 5-point SAM scale (Bradley and Lang, 1994). The outcome of the labeling was averaged over all labelers.

2. Categorical annotation of emotions into *neutral*, *positive*, *frustrated/angry*, *anxious/fearful* and free space to insert a *different emotional state*.

Additionally to the perceived emotional state, the labelers should also give feedback on how satisfied they were with their decision. From the satisfaction level the reliance of their decision could be concluded, e.g. if they were very dissatisfied with their emotional assessment. This implied that they were very indecisive in their decision and vice versa. To do so, a 5-point Likert-scale (1-Very dissatisfied, 2-Dissatisfied, 3-Unsure, 4-Satisfied, 5-Very satisfied) was used.
For all considered modalities, the data was rated by expert labelers. To receive feasible results by the labelers, at least 3 labelers were employed and their inter-rate reliability of the assessed emotional states was determined using Krippendorf's alpha. Using the inter-rater reliability, we could exclude outliers before averaging the result over the remaining labelers (Siegert et al., 2014).
For acoustic annotation, the ikannotate labeling tool (Böck et al., 2011) was used. The collected audio data was divided into short audio snippets of the same length and annotated afterwards. For the video annotation, the CAPTIV-L2100 software was used. The video signal was annotated in sequences of the same emotional state. All tools were adapted to the above mentioned requirement of the annotation.
The outcome of the audio and video annotation were used as ground truth for the physiological data. Depending on the reliance of the decision of the labelers, either the annotation results of the audio or video data were used.
Additionally to the expert rating, the results of the subjective self-reported feedback forms, filled out by each driver after every emotion scenario, were used as reference value of success for the emotion induction.

6. Study Realization

Before the data collection the participant filled out a self-reported personality questionnaire on the big five OCEAN personality traits and a general questionnaire on personal information. Then the driver was equipped with the physiological sensors (wristband and ECG electrodes on chest), a headset microphone, and the SEP camera system was calibrated. The outfitting of the driver took approx. 20 min. Afterwards, a short period of acclimatization (5 min) was given to the driver to get used to the equipment. During this time, the driver was introduced to the co-driver of the experiment and became acquainted with him. This was important as the co-driver took the role of the investigator in the different emotion scenarios. Therefore, the co-driver needed to be a trained psychologist.
After the acclimatization, the driver got in the car. In total, three persons were present in the car while conducting the experiments: the participant himself, the investigator sitting on the passenger seat, and one technician for the supervision of the sensor data recording sitting on the rear bench behind the passenger seat. The driving scenarios were fixed for all participants as depicted in figure 1.
For each emotional state, the experimental setup was as followed: 2.5 min of baseline driving without distraction of the driver to gather baseline data of the physiological data of the participant; 5 min of driving while conducting a task; 5 min of conversation with the investigator; 2 min filling out questionnaires.
The completion of the different emotional scenarios took in total approx. 10 to 15 min. In between every emotion scenario, the driver had at least 5 minutes of recess to get back to baseline. Depending on the intensity of the driver's emotional state, this time span was extended. At the beginning of each break, the driver was asked to fill out a short subjective self-reported feedback form on his emotional, sleepiness, stress and thermal state.
At the end of the study, the driver was debriefed by the test leader and a special debriefing information sheet was handed out to the participant. This debriefing also included discovering the detailed research objective of the data collection, which was detained from the participant previously. During an interview-like conversation, the driver was asked to give detailed subjective feedback on the recently experienced situations/moments. This was assisted by the interviewer who filled out a specially designed feedback form containing the answers given by the participant. This took approx. 15 min.
The data collection of one participant took approx. 180 min, in total.

7. Exemplary Results

This section will give a broad overview on the results by presenting the audio annotation of one exemplary participant. This participant was a male driver of 29 years. The evaluation of the demographic questionnaire shows a frequent usage of motorized vehicles with little experience in using advanced driver assistance systems. Form the ATI-scale a value of 4.8 was evaluated, which indicates an above-average positive attitude towards technology. From the BFI-10, it was drawn that the participant is strongly open to experience, shows high conscientiousness and high agreeableness. The results of the audio annotation will be presented in the remainder of this section, including a com-

parison with the self-reported feedback gathered from the GEW and detailed end-questionnaire.

7.1. Audio Annotation

For the audio annotation, three female labelers were employed. The inter-rater reliability of all three labelers can be seen in the first row of table 1. It shows an accordance for the labels of 0.27 for the annotation of the valence, 0.21 for annotation of arousal and 0.27 for the categorical annotation. By considering all labelers separately it was noticed, that one of the three labelers showed significant differences in the annotation result. This labeler was also not satisfied with some of the annotation results, while the other labelers never chose "dissatisfied" or "very dissatisfied" as satisfaction level. Therefore, this labeler was excluded for the further analysis of the results. The inter-rater reliability for the remaining two labelers could be increased to 0.39, 0.33 and 0.38, respectively. This is a reasonable result for the annotation of highly natural emotional audio data samples (Siegert et al., 2014).

	IRR		
Labelers	Valence	Arousal	Categories
All	0.27	0.21	0.27
Best	0.39	0.33	0.38

Table 1: Inter-rater reliability of all labelers vs. best two labelers.

In total, the labelers annotated 14 min of speech resulting in 556 speech samples. The labeling process took on average 4.3 hrs. In this time, the samples were annotated into the dimensions valence-arousal, into the categories neural, positive, frustrated/angry and anxious/fearful, and the labelers rated the satisfactory-level of the annotation. By considering the labelers' satisfactory-level, outliers were excluded from the sample-set before averaging the results over the remaining labelers. For the presented participant, the remaining labelers were never "dissatisfied" or "very dissatisfied" with their decision, this is why no samples were excluded from the sample-set.

For the valence-arousal annotation, the annotation results are presented as mapping onto the four quadrants of the circumplex model as pictured in figure 3. To transform the valence-arousal annotation results into these dimensional categories, the annotated SAM scale values (1-5) were averaged over the remaining two labelers. The origin of the two dimensions, in which the region of the neutral emotional state is located, is allocated to the valence and arousal value of 3. The neutral emotional state is defined as the region around the origin, with valence and arousal values lying within the scale (2-4). All values outside of this region are located in "q1", "q2", "q3" and "q4", or on the x/y-axes of the two dimensions. This approach resulted in 297 samples in region "n", 0 in "q1", 43 in "q2", 145 in "q3" and 3 in "q4". From those speech samples lying on the x/y-axes of the two dimensions, only samples with low arousal and neutral valence were recognized (68).

For the annotation of the emotional categories, only those samples annotated consistently by both remaining labelers were evaluated. In case of a high inter-rater reliability for all three labelers, a majority voting would have been carried out. This is not possible in case of two labelers. The results of the labelers was consistent for 263 samples. The corresponding annotation results are: 68 samples labeled as neutral, 48 as positive, 60 as frustrated/angry, 87 as anxious/fearful and none of the speech samples was annotated as different emotional states. The majority of the confusion between samples where the labelers did not find a consistent label was distributed equally between neutral and any other emotional state (202). This is expectable because of the high naturalness of the recorded samples.

		Dimensional					
		n	q1	q2	q3	q4	low
Categorical	Neutral	17	0	3	7	0	41
	Positive	35	0	13	0	0	0
	Anxious/fearful	27	0	1	55	0	4
	Frustrated/angry	48	0	0	9	3	0

Table 2: Confusion matrix of the categorical and dimensional annotation results. The entry "low" for the dimensional annotation denotes those samples annotated with low arousal and neutral valence.

Table 2 shows the confusion matrix of the categorical and dimensional annotation results. The entries marked in green indicate a suitable assignment between the categorical and dimensional annotation. Red entries indicate an unsuitable assignment. This implies a high consistency of the dimensional and categorical annotation in case of high values for green entries and low values for red values, which is given in this case. Entries marked in yellow are confusions of the categorical results with the neutral region of the dimensional approach. This is reasonable, as highly natural emotional samples of low emotional content and expressivity were examined. By optimizing the transformation of the dimensional annotation results, the accuracy of the two annotation approaches can be improved. Because of the high naturalness of the recorded audio samples, the neutral region around the origin of the dimensional approach needs to be reduced, to also be able to track small changes in the valence and arousal level. It is also noticed that those samples categorically annotated as neutral were of low arousal and neutral valence. This is in line with the emotional models presented in (Holzapfel et al., 2002) and (Almeida et al., 2016).

7.2. Evaluation of Questionnaires

The subjective self-reported questionnaires, conducted before and after the experiment, as well as the GEW, filled out by the participant in between each emotional scenario, were used to confirm a successful inducement of emotions. Evaluating the questionnaires and scales, the participant stated to be in a neutral emotional state while conducting the *neutral experiment*, with neutral valence, low arousal and moderate dominance. While conducting the *positive experiment*, he stated to be in a positive mood with moderate arousal and moderate dominance. While conducting the *frustrated experiment*, he stated to be in a negative state of valence, moderate arousal and very low dominance. He verified this statement by mentioning that he felt frustrated

and a bit ashamed. For the *anxious experiment*, the participant stated a negative valence, high arousal and moderate dominance, which was confirmed by his statement of being insecure while conducting the experiment leading to an anxious and confused feeling. From these statements we can assume that the inducement of the emotional states was realized successfully.

8. Conclusion and Outlook

This paper presents the collection of emotional multimodal real-world in-car audio, video and physiological signal recordings. The aim of this study was to be able to recognize behavioral factors of drivers while driving in a car, focusing on the emotional state of the driver. By conducting the presented experiments, it is possible to obtain multimodal, essentially natural emotional data, which enables a monitoring of the driver's emotional state. This could also be confirmed by the annotation results and participants' subjective feedback on the conducted experiments. This was exemplarily presented by evaluating the results of one randomly chosen participant of the study. It could be shown that the outcome of the two presented annotation approaches are consistent in their results and indicate a high naturalness of the annotated speech samples.

Description of the full data set, and corresponding annotation statistics, will be reported in a forthcoming paper. This will also include results of usability of the data for developing monitoring systems for mitigating the negative safety impact of emotionally affected drivers.

9. Acknowledgment

This paper has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 68890.

10. Bibliographical References

Abdić, I., Fridman, L., McDuff, D., Marchi, E., Reimer, B., and Schuller, B. (2016). Driver Frustration Detection From Audio and Video in the Wild. In *Proc. of the IJCAI-2016*, pages 1354–1360, New York, NY, USA.

Akerstedt, T. and Gillberg, M. (1990). Subjective and objective sleepiness in the active individual. *International Journal of Neuroscience*, 52(1–2):29–37.

Almeida, P. R., Ferreria-Santos, F., Chaves, P. L., Paiva, T. O., Barbosa, F., and Marques-Teixeira, J. (2016). Perceived arousal of facial expressions of emotion modulates the N170, regardless of emotional category: Time domain and time-frequency dynamics. *International Journal of Psychophysiology*, 99:48–56.

Altenburg, A. (2017). Wir sind die Freeses. http://www.ndr.de/ndr2/wir_sind_die_freeses/podcast4250.html.

American Automotive Association. (2017). Americans Feel Unsafe Sharing the Road with Fully Self-Driving Cars. http://newsroom.aaa.com/2017/03/americans-feel-unsafe-sharing-road-fully-self-driving-cars/.

Amir, N., Ron, S., and Laor, N. (2000). Analysis of an emotional speech corpus in Hebrew based on objective criteria. In *ITRW on Speech and Emotion*, pages 29–33, Newcastle, Northern Ireland, UK.

Böck, R., Siegert, I., Haase, M., Lange, J., and Wendemuth, A. (2011). ikannotate - A Tool for Labelling, Transcription, and Annotation of Emotionally Coloured Speech. In Sidney D'Mello, et al., editors, *Proc. of the ACII-2011*, volume 6974 of *LNCS*, pages 25–34, Memphis, TN, USA. Springer Verlag Berlin, Germany.

Botinhao, C. V. and Yamagishi, J. (2017). Speech intelligibility in cars: the effect of speaking style, noise and listener age. In *Proc. of the INTERSPEECH-2017*, pages 2944–2948, Stockholm, Sweden.

Bradley, M. M. and Lang, P. J. (1994). Measuring emotion: the Self-Assessment Manikin and the Semantic Differential. *Journal of Behavior Therapy and Experimental Psychiatry*, 25(1):49–59.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., and Weiss, B. (2005). A Database of German Emotional Speech. In *Proc. of the INTERSPEECH-2005*, pages 1517–1520, Lisbon, Portugal.

Dahlgren, A., Kecklund, G., and Akerstedt, T. (2005). Different levels of work-related stress and the effects on sleep, fatigue and cortisol. *Scandinavian Journal of Work, Environment and Health*, 31(4):277–285.

Devillers, L., Vidrascu, L., and Lamel, L. (2005). Challenges in real-life emotion annotation and machine learning based detection. *Neural Networks*, 18(4):407–422.

DLR. (2017). FASCar - test vehicle for assistance and automation. http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-11367/19950_read-46557/.

Engbert, I. S. and Hansen, A. V. (1996). Documentation of the Danish Emotional Speech Database DES. Technical report, Center for Person Kommunikation, Aalborg University, Denmark.

Eyben, F., Wöllmer, M., Poitschke, T., Schuller, B., Blaschke, C., Färber, B., and Nguyen-Thien, N. (2010). Emotion on the Road - Necessity, Acceptance, and Feasibility of Affective Computing in the Car. *Advances in Human-Computer Ineraction*, 2010:s.p.

Fischer, M., Richter, A., Schindler, J., Plättner, J., Temme, G., Kelsch, J., Assmann, D., and Köster, F. (2014). Modular and Scalable Driving Simulator Hardware and Software for the Development of Future Driver Assistence and Automation Systems. In *New Developments in Driving Simulation Design and Experiments*, Driving Simulation Conference, pages 223–229. Paris, France.

Forgas, J. P. (2002). Feeling and Doing: Affective Influences on Interpersonal Behavior. *Psychological Inquiry - An International Journal for the Advancement of Psychological Theory*, 13(1):1–28.

Franke, T., Attig, C., and Wessel, D. (2017). Affinity for technology interaction - a personal-resource perspective. In *Proc. of HFES Europe-2017*, Rome, Italy.

Gagge, A. P., Stolwijk, J. A. J., and Hardy, J. D. (1967). Comfort and thermal sensations and associated physiological responses at various ambient temperatures. *Environmental Research*, 1(1):1–20.

Garase, M. L. (2006). *Road Rage*. LFB Scholarly Publishing LLC.

Grimm, M., Kroschel, K., Harris, H., Nass, C., Schuller,

B., Rigoll, G., and Moosmayr, T. (2007a). On the Necessity and Feasibility of Detecting a Drivers Emotional State While Driving. In *Proc. of the ACII-2007*, volume 4738 of *LNCS*, pages 126–138, Lisbon, Protugal. Springer, Berlin, Heidelberg.

Grimm, M., Kroschel, K., Schuller, B., Rigoll, G., and Moosmayr, T. (2007b). Acoustic Emotion Recognition in Car Environment Using a 3D Emotion Space Approach. In *Proc. of the DAGA-2007*, Stuttgart, Germany.

Holzapfel, H., Fuegen, C., Denecke, M., and Waibel, A. (2002). Integrating Emotional Cues into a Framework for Dialogue Management. In *Proc. of the ICMI-2002*, pages 141–148, Pittsburgh, PA, USA.

Ihme, K., Dömeland, C., Freese, M., and Jipp, M. (in press). Frustration in the face of the driver: A driving simulator study on facial muscle activity during frustrated driving. *Interaction Studies*.

Jones, C. M. and Jonsson, I.-M. (2007). Performance Analysis of Acoustic Emotion Recognition for In-Car Conversational Interfaces. In C. Stephanidis, editor, *Proc. of the UAHCI-2007*, volume 4555 of *LNCS*, pages 411–420, Beijing, China. Springer, Berlin, Heidelberg.

Katsis, C. D., Katertsidis, N., Ganiatsas, G., and Fotiadis, D. I. (2008). Toward Emotion Recognition in Car-Racing Drivers: A Biosignal Processing Approach. *IEEE Transactions on Systems, Man, and Cyberenetics - Part A: Systems and Humans*, 38(3):502–512.

Koo, J., Kwac, J., Ju, W., Steinert, M., Leifer, L., and Nass, C. (2015). Why did my car just do that? Explaining semi-autonomous driving actions to improve driver understanding, trust, and performance. *International Journal on Interactive Design and Manufacturing*, 9(4):269–275.

Lazarus, R. S. (1991). Progress on a cognitive-motivational-relational theory of emotion. *American Psychologist*, 46(8):819–834.

Lee, B., Hasegawa-Johnson, M., Goudeseune, C., Kamdar, S., Borys, S., Liu, M., and Huang, T. (2004). AVICAR: Audio-Visual Speech Corpus in a Car Environment. In *Proc. of the INTERSPEECH-2004*, pages 2489–2492, Jeju, Jeju Island, South Korea.

Lewis, I. M., Watson, B. C., Tay, R. S., and White, K. M. (2007). The Role of Fear Appeals in Improving Driver Safety: A Review of the Effectiveness of Fear-arousing (threat) Appeals in Road Safety Advertising. *International Journal of Behavioral Consultation and Therapy*, 3(2):203–222.

Lotz, A. F., Faller, F., Siegert, I., and Wendemuth, A. (2018). Emotion Recognition from Disturbed Speech - Towards Affective Computing in Real-World In-Car Environments. In André Berton, et al., editors, *Elektronische Sprachsignalverarbeitung 2018*, Studientexte zur Sprachkommunikation, page s.p. TUDpress, Ulm, Germany.

Lu, J., Xie, X., and Zhang, R. (2013). Focusing on appraisals: how and why anger and fear influence driving risk perception. *Journal of Safety Research*, 45:65–73.

Malta, L., Miyajima, C., Kitaoka, N., and Takeda, K. (2011). Analysis of Real-World Driver's Frustration. *IEEE Transactions on Intelligent Transportation Systems*, 12(1):109–118.

Martin, O., Kotsia, I., Macq, B., and Pitas, I. (2006). The eNTERFACE'05 Audio-Visual Emotion Database. In *Proc. of the ICDEW-2006*, Atlanta, GA, USA.

Ortega, A., Sukno, F., Lleida, E., Frangi, A., Miguel, A., Buera, L., and Zacur, E. (2004). AV@CAR: A Spanish Multichannel Multimodal Corpus for In-Vehicle Automatic Audio-Visual Speech Recognition. In *Proc. of the LREC-2004*, pages 763–767, Lisbon, Protugal.

Pêcher, C., Lemercier, C., and Cellier, J.-M. (2009). Emotions drive attention: Effects on driver's behaviour. *Safety Science*, 47(9):1254–1259.

Rammstedt, B. and John, O. P. (2007). Measuring personality in one minute or less: A 10-item short version of the Big Five Inventory in English and German. *Journal of Research in Personality*, 41(1):203–212.

Rhodes, N. and Pivik, K. (2011). Age and gender differences in risky driving: the roles of positive affect and risk perception. *Accident Analysis and Prevention*, 43(3):923–931.

Russell, J. and Lemay, G. (2000). Emotion Concepts. In Michael Lewis et al., editors, *Handbook of Emotions*, pages 491–503. Guilford Press, 2 edition.

Scherer, K. R., Shuman, V., Fontaine, J. J. R., and Soriano, C. (2013). The GRID meets the Wheel: Assessing emotional feeling via self-report. In Johnny J. R. Fontaine, et al., editors, *Components of Emotional Meaning: A sourcebook*, pages 281–298. Oxford University Press.

Schmidt-Daffy, M. (2013). Fear and anxiety while driving: Differential impact of task demands, speed and motivation. *Transportation Research Part F: Traffic Psychology and Behaviour*, 16:14–28.

Shinar, D. (1998). Aggressive driving: the contribution of the drivers and the situation. *Transportation Research Part F: Traffic Psychology and Behaviour*, 1(2):137–160.

Siegert, I., Böck, R., and Wendemuth, A. (2014). Inter-rater reliability for emotion annotation in human-computer interaction: comparison and methodological improvements. *Journal on Multimodal User Interfaces*, 8(1):17–28.

Taubman-Ben-Ari, O. (2012). The effects of positive emotion priming on self-reported reckless driving. *Accident Analysis and Prevention*, 45:718–725.

Tawari, A. and Trivedi, M. (2010). Speech Based Emotion Classification Framework for Driver Assistance System. In *Proc. of 2010 IEEE Intelligent Vehicle Symposium*, pages 174–178, San Diego, CA, USA.

Willstrand, T. D., Anund, A., Strand, N., Nikolaou, S., Touliou, K., Gemou, M., and Faller, F. (2017). Deliverable 1.2 - Driver/Rider models, Use Cases and implementation scenarios. http://www.adasandme.com/wp-content/uploads/2017/06/ADASME-20170619-D1.2-Driver-Rider-models-Use-Cases-and-implementation-scenarios.pdf.

Zhang, T., Chan, A. H. S., and Zhang, W. (2015). Dimensions of driving anger and their relationships with aberrant driving. *Accident Analysis & Prevention*, 81:124–133.

EmotionLines: An Emotion Corpus of Multi-Party Conversations

Sheng-Yeh Chen[1*], Chao-Chun Hsu[1*], Chuan-Chun Kuo[1], Ting-Hao (Kenneth) Huang[2], Lun-Wei Ku[1]
Academia Sinica, Taiwan
Carnegie Mellon University, USA
[1]{alan012102, joe32140, william0617, lwku}@iis.sinica.edu.tw
[2]tinghaoh@cs.cmu.edu

Abstract

Feeling emotion is a critical characteristic to distinguish people from machines. Among all the multi-modal resources for emotion detection, textual datasets are those containing the least additional information in addition to semantics, and hence are adopted widely for testing the developed systems. However, most of the textual emotional datasets consist of emotion labels of only individual words, sentences or documents, which makes it challenging to discuss the contextual flow of emotions. In this paper, we introduce EmotionLines, the first dataset with emotions labeling on all utterances in each dialogue only based on their textual content. Dialogues in EmotionLines are collected from Friends TV scripts and private Facebook messenger dialogues. Then one of seven emotions, six Ekman's basic emotions plus the neutral emotion, is labeled on each utterance by 5 Amazon MTurkers. A total of 29,245 utterances from 2,000 dialogues are labeled in EmotionLines. We also provide several strong baselines for emotion detection models on EmotionLines in this paper.

Keywords: emotion detection, emotional dialogue dataset

1. Introduction

There are two major kinds of dialogue systems: a task-oriented dialogue system and the social (chit-chat) dialogue system. The former focuses on designing a personal assistant which can accomplish certain tasks, and for the latter it is important to capture the conversation flow which emphasizes more on the feelings of the speaker. Many researchers try to build a "smart" dialogue system by enhancing dialogue breadth (coverage), dialogue depth (complexity) or both. Those who want to increase dialogue breadth try to transfer dialogue acts across domains (Chen et al., 2016) to establish multi-domain or even open domain dialogue system, and those who want to deepen dialogue complexity pay their attention to transform a knowledge-based systems to common sense or even empathetic systems that can recognize emotion features, generate emotion-aware responses (Fung et al., 2016), or learn how to plan the dialogues while users interact via high-level descriptions (Sun et al., 2016). No matter what kind of dialogue system we want to build, a useful and large dialogue dataset is indispensable.

When building a task-oriented dialogue system, dialogue corpora with dialogue act information is accessible and hence are commonly utilized. However when building a chit-chat conversational bot, though the importance of emotion detection has been noticed, pure conversation content such as movie, TV scripts or chat logs without emotion labels are more available: no emotion labels on utterances can be used for learning. Moreover, when we turn to other datasets with annotated emotion information such as data crawled from Twitter (Mohammad and Bravo-Marquez, 2017), the labeled units (posts or sentences) are independent. As a result, models built with these datasets lack the ability to consider contextual information essential in dialogue systems, not to mention the ability to capture the emotion flow. We illustrate this issue with examples shown in Table 1.

Post	Label
Just got back from seeing @GaryDelaney in Burslem. AMAZING!! Face still hurts from laughing so much #hilarious	Joy
Feeling worthless as always #depression	Sadness
I get so nervous even thinking about talking to *** *** I wanna die	Fear
Wont use using @mothercareuk @Mothercarehelp again!! These guys cant get nothing right!! #fuming	Anger

Table 1: Emotion labeled posts without contextual information (selected from WASSA-2017 Shared Task on Emotion Intensity)

Modeling emotion on one single utterance without contextual information may encounter another issue that the same utterance can express different emotions depending on its context. Table 2 shows some examples of saying "Okay!" with different emotions.

The IEMOCAP database (Busso et al., 2008), to the best of our knowledge, is the only dataset that provides emotion labels for each utterance. However, IEMOCAP was created by actors performing emotions, and hence carries the risk of overacting. Moreover, the annotators label the emotions by watching the videos instead of reading the transcripts which means the annotators may make the decision only depend on the facial expression or the prosodic features without realizing the meaning of the words.

To tackle these problems, we create EmotionLines: an

*These authors contributed equally to this work.

Chandler	Matthew Perry talking about signs in Las Vegas. (Neutral)
Chandler	I guess it must've been some movie I saw. (Neutral)
Chandler	What do you say? (Neutral)
Monica	***Okay! (Joy)***
Chandler	Okay! Come on! Let's go! All right! (Joy)
Rachel	Oh okay, I'll fix that to. What's her e-mail address? (Neutral)
Ross	Rachel! (Anger)
Rachel	All right, I promise. I'll fix this. I swear. I'll-I'll- I'll-I'll talk to her. (Non-neutral)
Ross	***Okay! (Anger)***
Rachel	Okay. (Neutral)

Table 2: "Okay!" of different emotions from Friends TV scripts.

emotion dialogue dataset with emotion labels on each utterance. The collected textual dialogues are from not only scripts of TV shows but also real, private, human-to-human chat logs. We establish several strong baselines for the emotion detection task on dialogues, and motivate an automatic metric to benchmark progress. Modeling sequential emotions in dialogues, as provided in EmotionLines, has the potential to move dialog systems from generating understandable messages to more human-like responses. To the best of our knowledge, this is the first textual emotion dialogue dataset with the emotion label on each utterance in dialogues.

2. Related Work

Sentiment analysis, which can help users and companies capture people's opinion, is getting a growing attention by both research community and business word because of the research challenge and the potential value to make profits. Since the social media and the instant message platforms become the important part of our daily life, we can easily obtain a large amount of these user-generated content to get better understanding of emotion. Thus improve the satisfaction for web services and call centers (Devillers and Vidrascu, 2006).

In 1974, Ekman conducted extensive studies on emotion recognition research over 6 basic emotions: anger, disgust, fear, happiness, sadness, and surprise. His study shows it is possible to detect emotions given enough features (Ekman et al., 1987). Later studies on text-based emotion recognition are mainly divided into three categories: keyword-based, learning-based, and hybrid recommendation approaches (Kao et al., 2009). Recently, emotion recognition researches on text focus on the learning-based methods. Kim proposed CNN(Convolutional neural network) text classification, which is widely used for extracting sentence information (Kim, 2014). However, single sentence emotion recognition is lack of contextual emotion flow within a dialogue. Therefore, contextual LSTM(Long short-term memory) architecture is proposed to measure the inter-dependency of utterances in the dialogue (Poria et al., 2017). In this paper, we report the performance of the CNN model and the contextual LSTM architecture on the proposed EmotionLines dataset as baselines.

3. Corpus

3.1. Data Source

To bring conversations closer to real-word dialogues, we selected sources from both TV shows scripts and Human-to-human chat logs. First, we crawled the scripts of seasons 1 to 9 of *Friends* TV shows[1]. Second, we requested private dialogues from Wang (2016), which are conversations between friends on Facebook Messenger collected by an app called *EmotionPush* [2].

3.1.1. Friends TV Scripts

The crawled scripts are separated as episodes, and we viewed each scene in every episode as a dialogue. Then, the collected dialogues were categorized according to their dialogue length, i.e. the number of utterances in a dialogue, into four classes of which bucket length ranges are [5, 9], [10, 14], [15, 19], and [20, 24]. Finally, we randomly sampled 250 dialogues from each class to construct a dataset containing 1,000 dialogues.

3.1.2. EmotionPush Chat Logs

For the private dialogues from *EmotionPush*, we assumed that a dialogue would not sustain more than 30 minutes, and messages separated in time by less than 300 seconds were put in the same dialogue. At last, the dialogues are categorized and sampled using the same procedure as that for the Friends TV scripts, and we obtained 1,000 dialogues from EmotionPush chat logs.

3.2. Human-level Labeling

We placed our dialogues on the Amazon Mechanical Turk, and each dialogue is regarded as an annotation task where each utterance is labeled with one of Ekman's six basic emotions (1987) anger, disgust,fear, happiness, sadness, surprise, and the additional emotion neutral. The total of seven labels are Neutral, Joy, Sadness, Fear, Anger, Surprise, and Disgust respectively. For every MTurk HIT, we designed a web interface like Figure 1, and asked crowd workers to mark each utterance in a dialogue considering the context in the whole dialogue. Workers should think for at least 3 seconds before selecting an answer. For HITs with different dialogue length, we assign distinctive payments according to the bucket length ranges mentioned above, where the award is 0.1, 0.15, 0.2, and 0.25 dollars per HIT respectively.

Each HIT was accomplished by 5 workers, and for each utterance in a HIT, the emotion with the highest number of votes was set as the gold label of the utterance. Those utterances with more than two different emotions voted were put into the non-neutral category.

[1]Scripts of seasons 1-9 of "Friends": `http://www.livesinabox.com/friends/scripts.shtml`

[2]Participants consented to make their private conversations available for research purposes.

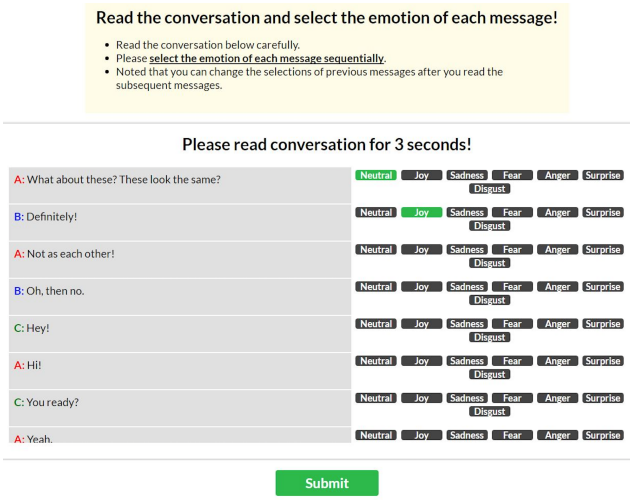


Figure 1: Worker interface on Amazon Mechanical Turk

speaker	Rachel
utterance	Hi Joey! What are you doing here?
emotion	joy
speaker	Joey
utterance	Uhh, well I've got an audition down the street and I spilled sauce all over the front of my shirt. You got an extra one?
emotion	neutral
speaker	Rachel
utterance	Yeah, sure. Umm... here.
emotion	neutral

Table 3: Data format of EmotionLines

3.3. De-identification

The EmotionPush chat logs are from private conversations. Therefore, the logs may contain personal information such as names of real people, locations, organizations, and email addresses. In order to protect the privacy of EmotionPush users, we performed a two-step masking process. First, Stanford Named Entity Recognizer (Finkel et al., 2005) was adopted to detect named entities mentioned by each utterance, which were later replaced with their entity types.

After this step, however, we still found named entities like lowercase or foreign names and emails. Therefore, we manually checked and cleaned utterances in the first step again to prevent the accidental reveal of personal data.

Since the conversations collected in the EmotionPush chat logs involve not only our participants, we want to carefully protect their identity. Therefore, we hired a native speaker whose occupation is an editor to rewrite all the messages of participants' friends. The rewriting process follows the guideline from Bruckman (Bruckman, 2006). In addition, we asked the rewriter to check all named entities and mask them by its categories. For example, "my mother" remains the same, but "Sam" is replaced by "person n" indicating the n^{th} person being de-identified; "the park near my house" remains the same but "Taipei 101" is replaced by "location m" indicating the m^{th} location being de-identified.

3.4. Data Format

Both Friends TV scripts and EmotionPush chat logs contain 1,000 dialogues. The lengths of dialogues vary. Each utterance of the dialogue has the same format, which involves information of the speaker, the content, and the emotion labels of the utterance. We show an example in Table 3.

4. Analysis

4.1. Data Information

The analysis of data from two sources is shown in Table 4. We calculated the numbers of utterances of each type of emotion, and found that except *neutral* type, *joy* and *surprise* appear more frequently in the dataset. Besides, EmotionPush chat logs have more skewed label distribution than Friends TV scripts. Interestingly, the average length of real private utterances is much shorter than the length of those of TV show scripts (10.67 vs. 6.84).

We adopted Fleiss' kappa to measure the agreement among annotators of the labeling task of the dataset. The kappa scores are above 0.33 for labels of both the Friends scripts and EmotionPush, which indicates a solid basis for a subjective labeling task.

4.2. Train-/Dev-/Test-Set Split

We not only constructed an emotion dialogue corpus, but also split the dataset from two sources into training, development, and testing set separately. In order to preserve completeness of any dialogue, we divided the corpus by the dialogues, not the utterances. Table 6 shows the information of each set.

5. Experiments

5.1. Modeling a Single Utterance

Given a utterance of M words, the one-hot encoding for utterance words is denoted by $U = \{w_1, w_2, w_3, ..., w_M\}$. We first embed the words to the word embedding , which is publicly available 300-dimensional GloVe pre-trained on Common Crawl data (Pennington et al., 2014). Thus each utterance in u_i is represented by a feature matrix $F \in \mathbb{R}^{M \times 300}$. Then, a 1-D convolution with k filters of r window sizes from 1 to r, followed by a 1-D max-pooling is applied on F. The concatenation of max-pooling outputs of different window sizes is denoted as f with dimension $k \times r$. k is set to 64 and r is set to 5 in the experiment.

5.2. Modeling on the Whole Dialogue

In a paragraph, the sentiment of each utterance is dependent on the context. Thus, within a dialogue, there is a high probability of inter-dependency with respect to their sentimental clues. When we classify an utterance, other utterances may provide import contextual information. To measure this information flow, we apply the contextual LSTM architecture. The inputs of contextual LSTM for each dia-

	# of Utterances	Utterance Length	Emotion Label Distribution (%)								kappa (%)
			Neu	Joy	Sad	Fea	Ang	Sur	Dis	Non	
Friends	14,503	10.67	45.03	11.79	3.43	1.70	5.23	11.43	2.28	19.11	33.83
EmotionPush	14,742	6.84	66.85	14.25	3.49	0.28	0.95	3.85	0.72	9.62	33.64

Table 4: Detail information of Friends TV scripts and EmotionPush chat logs

		WA	UWA	Neu	Joy	Sad	Fea	Ang	Sur	Non
CNN	**Friends**	**59.2**	**45.2**	64.3	60.2	41.2	21.9	46.6	61.5	20.6
	EmotionPush*	**71.5**	**41.7**	80.8	46.9	43.7	0.0	27.0	53.8	40.0
CNN-BiLSTM	**Friends**	**63.9**	**43.1**	74.7	61.8	45.9	12.5	46.6	51.0	8.8
	EmotionPush*	**77.4**	**39.4**	87.0	60.3	28.7	0.0	32.4	40.9	26.7

Table 5: Weighted and unweighted accuracy on Friends and EmotionPush

	Friends			EmotionPush		
	train	dev	test	train	dev	test
# of D	720	80	200	720	80	200
# of U	10,561	1,178	2,764	10,733	1,202	2,807
D-len	14.67	14.73	13.82	14.91	15.03	14.04
U-len	10.20	10.08	10.44	6.73	6.96	7.24

Table 6: Information of train/dev/test set of Friends and EmotionPush dataset (D and U represent dialogue and utterance correspondingly, and the dialogue/utterance lengths were averaged.)

logue with length L are denoted as $X = \{x_1, ..., x_L\}$.

$$x_i = \tanh(W_x \cdot f_i + b_x) \quad (1)$$

The output of LSTM cell h_i is then fed to the dense layer followed by a softmax layer. Then we compute loss by cross-entropy as follows:

$$loss = -\frac{1}{\sum_{c \in C} N_j} \sum_{c \in C} \sum_{i=1}^{N_c} \sum_{l \in C} (y_l^i) log(\hat{y}_l^i) \quad (2)$$

where C is the emotion class set for evaluation, N_c denotes the number of utterances in class c, y_l^i is the original output, and $\hat{y}_l^i$ is the predicted output for the i-th utterance in emotion class l.

5.3. Performance on EmotionLines

We conduct experiments on EmotionLines with the CNN model and the CNN-Bidirectional LSTM(CNN-BiLSTM) model. Results are shown in Table 5. The performance of the CNN and CNN-BiLSTM model is evaluated by both the weighted accuracy (WA) and the unweighted accuracy (UWA) shown as follows.

$$\text{WA} = \sum_{l \in C} s_l a_l \quad (3)$$

$$\text{UWA} = \frac{1}{|C|} \sum_{l \in C} a_l \quad (4)$$

where a_l denotes the accuracy of emotion class l and s_l denotes the percentage of utterances in emotion class l.

The improvements of weighted accuracy from 59.2% to 63.9% on the Friends dataset and from 71.5% to 77.4% on the EmotionPush dataset show that using the contextual information (CNN-BiLSTM) can help recognize emotions. Note that the reported performance is from the experiments conducted on the raw data, which are not de-identified yet. Updated results will be provided later in the dataset download webpage[3].

6. Conclusion and Future Work

We have constructed EmotionLines, the emotion dialogue dataset containing the text content for each utterance annotated with one of seven emotion-categorical labels. The kappa value shows the good quality of these generated labels. In addition, several experiments were performed to provide baselines and to show contextual information is beneficial for the dialogue emotion recognition. The provided strong baselines are weighted accuracy 63.9% and 77.4% for Friends and EmotionPush, respectively.

Due to the imbalanced nature of emotion label distribution, one of our future work is to collect specific types of the label to enrich the minor emotion categories, e.g., trying horror movies scripts to get more *fear* utterances and tragedies for *sadness* utterances. EmotionLines is now available at `http://academiasinicanlplab.github.io/#download`.

7. Acknowledgement

This research is partially supported by Ministry of Science and Technology, Taiwan, under Grant no. MOST 106-2218-E-002-043-.

[3] http://academiasinicanlplab.github.io/#download

8. References

Bruckman, A. (2006). Teaching students to study online communities ethically. *Journal of Information Ethics*, page 82.

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. (2008). Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335.

Chen, Y.-N., Hakkani-Tür, D., and He, X. (2016). Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 6045–6049. IEEE.

Devillers, L. and Vidrascu, L. (2006). Real-life emotions detection with lexical and paralinguistic cues on human-human call center dialogs. In *Ninth International Conference on Spoken Language Processing*.

Ekman, P., Friesen, W. V., O'sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., Krause, R., LeCompte, W. A., Pitcairn, T., Ricci-Bitti, P. E., et al. (1987). Universals and cultural differences in the judgments of facial expressions of emotion. *Journal of personality and social psychology*, 53(4):712.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

Fung, P., Bertero, D., Wan, Y., Dey, A., Chan, R. H. Y., Siddique, F. B., Yang, Y., Wu, C.-S., and Lin, R. (2016). Towards empathetic human-robot interactions. *arXiv preprint arXiv:1605.04072*.

Kao, E. C. C., Liu, C. C., Yang, T. H., Hsieh, C. T., and Soo, V. W. (2009). Towards text-based emotion detection a survey and possible improvements. In *2009 International Conference on Information Management and Engineering*, pages 70–74, April.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Mohammad, S. M. and Bravo-Marquez, F. (2017). Wassa-2017 shared task on emotion intensity. *arXiv preprint arXiv:1708.03700*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Poria, S., Cambria, E., Hazarika, D., Mazumder, N., Zadeh, A., and Morency, L.-P. (2017). Context-dependent sentiment analysis in user-generated videos. In *Association for Computational Linguistics*.

Sun, M., Chen, Y.-N., Hua, Z., Tamres-Rudnicky, Y., Dash, A., and Rudnicky, A. I. (2016). Appdialogue: Multi-app dialogues for intelligent assistants. In *LREC*.

Wang, S.-M., Li, C.-H., Lo, Y.-C., Huang, T.-H. K., and Ku, L.-W. (2016). Sensing emotions in text messages: An application and deployment study of emotionpush. *arXiv preprint arXiv:1610.04758*.

Academic-Industrial Perspective on the Development and Deployment of a Moderation System for a Newspaper Website

Dietmar Schabus, Marcin Skowron
Austrian Research Institute for Artificial Intelligence (OFAI)
Freyung 6/6, 1010 Vienna, Austria
{dietmar.schabus, marcin.skowron}@ofai.at

Abstract

This paper describes an approach and our experiences from the development, deployment and usability testing of a Natural Language Processing (NLP) and Information Retrieval system that supports the moderation of user comments on a large newspaper website. We highlight some of the differences between industry-oriented and academic research settings and their influence on the decisions made in the data collection and annotation processes, selection of document representation and machine learning methods. We report on classification results, where the problems to solve and the data to work with come from a commercial enterprise. In this context typical for NLP research, we discuss relevant industrial aspects. We believe that the challenges faced as well as the solutions proposed for addressing them can provide insights to others working in a similar setting. Data and experiment code related to this paper are available for download at https://ofai.github.io/million-post-corpus.

1. Background

For about two years, we have been working on an applied research project in a collaboration between a research institute and a large Austrian broadsheet newspaper (DER STANDARD), which supports the moderation of the comments posted to the newspaper's website[1] by its readers.

Like many newspaper websites, DER STANDARD's website features a comment section below each newspaper article, where users engage in discussion. In the year 2017, more than 9.5 million comments were posted by more than 55,000 distinct users. To ensure high quality in the discourse, the newspaper's community management department invests considerable effort in the moderation of the discussion fora, using both machine-learning-based tools and a team of professional human forum moderators.

With the project goal to improve the moderation, the moderators have defined eight relevant categories of posts, and have annotated a collection of posts with respect to these categories. The annotated categories are "negative sentiment", "positive sentiment", "off-topic", "inappropriate", "discriminating", "feedback", "personal stories" and "arguments used". The detailed description of the categories and the annotation process, as well as the resulting data set and the baseline classification results are provided in Schabus et al. (2017). Both the data set and the classification experiment code are available online for research purposes.[2]

We have designed, developed and deployed a moderator dashboard that provides various ways of searching, filtering, sorting and aggregating according to the introduced set of categories to help the moderators find locations in the discussions where moderation actions are required. In this process, we have addressed the following tasks that often interconnect predominately research- and industry-oriented aspects:

- Automatic labeling of new user comments according to the defined categories – a text classification problem,
- Using these predictions and other (meta-)data for finding posts and/or entire discussions that require moderator attention – an information retrieval problem,
- Providing a user interface to the moderators so they can use the results of the above in their workflow – user interface design,
- Integrating the resulting system into the existing IT infrastructure – system integration.

Since we want to deliver a (prototype) system that is usable in practice, our setting differs considerably from academic research in several aspects. For example, both user interface design and system integration are not typically relevant in NLP research.

2. Challenges

In this section, we describe a few challenges we have faced in detail and how the issues were addressed. We have grouped them under the four terms *holism*, *specificity*, *"messy" data* and *integration*, highlighting differences between academic and industrial settings.

2.1. Holism

In academic research we are often focused on a highly specific problem, and we can make extensive assumptions about aspects that are not in the immediate center of attention. In contrast, industrial endeavors require a more holistic view; they need to work with given practical settings and address specific requirements of live systems. In particular, we have identified three relevant perspectives to our project, all of which need to be considered simultaneously:

The *scientific/technical perspective* focuses on questions such as which methods to apply for particular subproblems, how to best make use of the available data, which evaluation metrics to apply, etc.

The *industrial perspective* focuses on the operational realization and deals with topics like integration and interfaces, software quality, performance and scalability, privacy, security, backups, etc.

[1] https://derstandard.at

[2] https://ofai.github.io/million-post-corpus

Finally, the *user perspective* is concerned with the benefits the system is able to deliver to the end-users, in our case the moderators working for the newspaper.
For example, using evaluation metrics like precision, recall and F_1-score for a classification problem is well-established in machine learning research (scientific perspective), but measuring the time savings for a well-defined moderation task tells us more adequately how well we are addressing the needs of the moderators (user perspective).
It has shown to be beneficial to frequently switch between these perspectives during the project timespan or to consider and address them simultaneously.

2.2. Specificity

Even though we have identified the requirement for holistic thinking as one challenge for our endeavor in the previous subsection, we can at the same time also name challenges that come from the highly specific practical needs in the given real-world setting. For example, the applied classification scheme (i.e., the annotated categories) could be criticized in a purely academic setting as being specifically tailored to the needs of one particular newspaper. Indeed, it is difficult to find related work that deals with text classification according to categories like "arguments used", "personal stories" or "feedback" originating from the concrete moderation needs at DER STANDARD. And even for our category negative/positive sentiment, the related research literature in sentiment analysis often relates to online *reviews* for different things like movies (Pang and Lee, 2005; Socher et al., 2013; Maas et al., 2011; Le and Mikolov, 2014), restaurants (Snyder and Barzilay, 2007) or books (Sakunkoo and Sakunkoo, 2009), where a clear indication of sentiment can be expected in every document, and the domain restriction can be expected to facilitate sentiment classification. In our data, sentiment has yet another specific meaning stemming from the application; in particular, the moderators are interested in locating negative sentiment in order to prevent escalation in user discussions.
In general, in a concrete industrial setting it is likely that one has to deal with very specific phenomena, use cases and goals, and results from academic research might not carry over directly.

2.3. "Messy" Data

Data annotation by humans is time-consuming and thus costly, especially when specific domain expertise is necessary, as is the case with the categories the moderators have defined with their particular moderation goals at DER STANDARD in mind. We need to ensure efficient use of moderator time in data annotation for model training and evaluation. Furthermore, most categories can be considered rare anomalies, resulting in strongly unbalanced data (e.g., the binary category "discriminating" has a prevalence of about 8% in our data set). These two factors explain the somewhat complicated, exploratory annotation procedure described in our data set paper (Schabus et al., 2017): In the first attempt, where 1,000 user comments were selected randomly for annotation, some categories were very weakly represented. Subsequently making use of the moderators' experience in selecting suitable topics (e.g., articles about the refugee crisis or gender mainstreaming for finding discriminating posts) turned out to be helpful for acquiring additional positive instances.[3] However, this also has unwanted side-effects; First, it means that many posts are annotated only according to one particular category, i.e., the data sets for the categories are mostly disjoint and consequently separate classification models must be trained, rather than a single multi-label model. Second, the class distributions in the labeled data are no longer necessarily indicative of the real class distributions in practice. And even if we accept these disadvantages, it still does not mean that we have vast amounts of data: In our data collection, even the better represented categories have less than 2,000 positive examples.
While in academic research settings methods are typically evaluated on carefully compiled benchmark data sets, which have reasonable balance and size, in practical industry applied research settings these might not always be available in a similar quality and quantity. Our situation is also different from what one might associate with an industrial setting typical for large companies which have less limitations in terms of available data or capacities to conduct large scale annotations. In the presented approach we thus focused on the efficient usage of the available data acknowledging its characteristics which are neither typical for academia nor for large enterprises.

2.4. Integration

The goal of the project is to deliver a prototype system applicable in practice, i.e., supporting the moderation of current online discussions. Therefore, a connection to the production forum system is required, such that the prototype works with the live stream of new postings in near real-time. To achieve this, the prototype needed to be integrated into the existing IT infrastructure at the newspaper.
The IT environments typically used in research institutions (operating systems, programming languages, software libraries, database systems, etc.) differ significantly from those used in commercial enterprises. In the former case, open-source libraries are often used, where new methodological advances become available quickly. In the latter case, systems from large commercial vendors are often preferred, with certifications, support services, etc. Letting researchers use the tools they are accustomed to is beneficial for flexibility and agility in experimental prototype development; on the other hand, a prototype using the same technologies as the existing environment facilitates integration. Our approach to this situation was to compromise: give the researchers flexibility in the core area of experimentation (e.g., machine learning frameworks), but adapt to the enterprise systems in other areas (e.g., database system).
Another important aspect of adding an experimental prototype from a collaborative research project to a production environment is (data) security and privacy. No enterprise would tolerate the risks involved with a prototype system directly manipulating its production databases for a service used by thousands of users every day. Therefore, the data

[3] Positive here means that the property in question is present, e.g., that a given posting does exhibit the characteristics of the "discriminating" category.

was mirrored to a dedicated database server for the prototype system, such that the production system is shielded from potential bugs. By restricting this mirroring to the data that is actually required, most privacy risks can be avoided (e.g., no personal data of users are mirrored).

In a practical setting, scalability and performance under peak loads become key factors. In our setting with up to 200 new comments per minute and eight different labels to predict, the run-time performance requirements for prediction (time and memory) influenced the choice of methods. By keeping the models for comment representation and classification small enough to all fit into main memory simultaneously, and by processing new comments in batches, we achieve a performance of almost 20,000 classified comments per minute on a machine with 16 cores.

Finally, we need to keep in mind that the system needs to be completed on time before the end of the project and operated and maintained by the industry partner after that. Therefore, clean code, suitable error handling and documentation are essential; these aspects typically can and are neglected in purely academic settings.

3. Experimental Results

To better illustrate some of the challenges we face in our concrete industrial setting, we report the results of new experiments using our data set, which extend the experiments from our previous work (Schabus et al., 2017). There, the most promising method was a (linear) support vector machine on a paragraph vector representation (Le and Mikolov, 2014), which we further investigate in our new experiments, using the old setup as a baseline.

The first extension we consider is to train two paragraph vector models (one using the distributed memory method and one using the distributed bag-of-words method) and to then represent each document by the concatenation of the two vectors, as proposed by Le and Mikolov (2014). We used a vector size of 100 dimensions for each of the two models instead of 300 as we did in the baseline setup as this turned out to be superior in preliminary experiments, and it also keeps the dimensionality from becoming too excessive when two vectors are concatenated.

Secondly, we add topic features to the representation: Each of our user comments belongs to a news article, and for each news article we have meta-data including a topic path such as *sports / motorsports / formula 1*. We have selected 17 top level topics (e.g., *sports*, *economy*, *science*, etc.) and added the resulting 17 binary dimensions indicating topic membership to the representation for each comment.

Finally, we compare support vector machines with linear kernels against Radial Basis Function (RBF) kernels, motivated by the hypothesis that the new composite feature space requires more complex decision boundaries for accurate classification.

The evaluation results of a 10-fold stratified cross-validation on the data set from (Schabus et al., 2017) are given in Table 1, where Method 1 represents the baseline results from our prior study. Methods 2–9 represent all combinations of the three configuration options described above. Note that Method 2 is identical to the baseline except with regard to the number of dimensions (100 vs. 300).

In terms of F_1-score, Method 9 (concatenation, topics and RBF kernel) outperforms the baseline on five of eight categories, and Method 8 (concatenation, topics and linear kernel) outperforms the baseline on an additional category ("negative sentiment"). For the two remaining categories "inappropriate" and "personal stories", the results of Method 9 are less than 0.01 below the baseline results.

Using the concatenated representation generally helps to improve the prediction results (Methods 4 and 5 vs. Methods 2 and 3), most noticeably for the categories "feedback" and "personal stories". Adding topic information also generally improves the prediction results (Methods 6 and 7 vs. Methods 2 and 3), this time most noticeably for categories "negative sentiment", "off-topic", "discriminating", "feedback" and "personal stories". Combining both representation extensions results in further improvement, especially for the "negative sentiment", "off-topic" and "feedback" categories. Even when Methods 8 and 9 are not the best performing, the differences are insignificant for practical settings and therefore we choose one of these two for deployment, favoring a more uniform overall setup.

With respect to the challenges discussed in Section 2., the specificity of the data we work with becomes apparent in the context of the experiments. For example, there are no directly applicable baseline results to compare against for most of our categories, with the exception of the two "sentiment" categories where extensive prior work exists. Here however the differences are in the definition and scope of the labels, hindering direct comparison of classification results. For example, Le and Mikolov (2014) report sentiment classification accuracies above 90% on a balanced data set of movie reviews, while our best result for negative sentiment in terms of minority class F_1-score (0.6063) corresponds to only 63% accuracy on our set of user comments, which are highly diverse in terms of topic, style, length, author intention, etc.

Finally, the integration aspect also plays a role in selecting the classification method to use in practice. For example, with deep LSTM models, which achieved competitive results in our previous work, we need to sequentially load separate large models onto a GPU for efficient classification, increasing the required efforts in operation and maintenance of the system after the "hand-over" to the industry partner. On the other hand, paragraph vectors are an efficient representation in our scenario, because they are computed only once for all eight categories, and then fed into separate SVM models. The resulting system is relatively light-weight and easier to deploy and maintain in the long term.

4. Conclusions

In this "industry track" paper, we have shared our experiences from a collaborative applied research project involving a small research institution and a medium size commercial enterprise. The goal of the project is to develop and deploy a prototype system that supports the moderation of user discussions on a large newspaper website. A key building block of this system is a text classification module predicting eight moderator-defined category labels. We have described a number of challenges faced in this context and

		Method								
		Concat	✗	✗	✓	✓	✗	✗	✓	✓
		Topics	✗	✗	✗	✗	✓	✓	✓	✓
		Kernel	Linear	RBF	Linear	RBF	Linear	RBF	Linear	RBF
Category	Measure	1 (BL)	2	3	4	5	6	7	8	9
Negative	Precision	0.5842	0.5653	0.5755	0.5832	0.5893	0.5975	0.6106	0.6112	**0.6216**
	Recall	0.5624	0.5659	0.5228	0.5908	0.5192	0.5310	0.4684	**0.6014**	0.4837
	F_1	0.5731	0.5656	0.5479	0.5870	0.5520	0.5623	0.5301	**0.6063**	0.5441
Positive	Precision	0.0397	0.0644	0.0707	0.0845	0.0618	**0.1020**	0.0851	0.0804	0.0977
	Recall	**0.4651**	0.3488	0.3023	0.2791	0.2558	0.3488	0.2791	0.2093	0.3023
	F_1	0.0731	0.1087	0.1145	0.1297	0.0995	**0.1579**	0.1304	0.1161	0.1477
OffTopic	Precision	0.2065	0.1930	0.2039	0.2010	0.2090	0.2284	**0.2579**	0.2472	0.2524
	Recall	**0.6241**	0.5897	0.4552	0.5707	0.4724	0.5741	0.4759	0.6086	0.4534
	F_1	0.3103	0.2908	0.2816	0.2973	0.2898	0.3268	0.3345	**0.3516**	0.3243
Inappr	Precision	0.1340	0.1074	0.1382	0.1218	**0.1475**	0.1203	0.1340	0.1179	0.1433
	Recall	0.5776	0.5347	0.4059	0.5116	0.4059	**0.5974**	0.4158	0.5248	0.4125
	F_1	**0.2175**	0.1789	0.2062	0.1967	0.2164	0.2002	0.2027	0.1925	0.2128
Discrim	Precision	0.1111	0.1038	0.1206	0.1115	0.1402	0.1207	0.1343	0.1223	**0.1547**
	Recall	0.3936	0.4574	0.2057	0.4610	0.1844	**0.5922**	0.3440	0.5071	0.2837
	F_1	0.1733	0.1692	0.1520	0.1796	0.1593	**0.2005**	0.1932	0.1971	0.2003
Feedb	Precision	0.5240	0.4604	0.5039	0.4865	**0.5393**	0.4520	0.4743	0.4839	0.5311
	Recall	0.7056	0.7233	0.6472	0.7317	0.7018	**0.7679**	0.7294	0.7633	0.7356
	F_1	0.6014	0.5626	0.5666	0.5844	0.6099	0.5691	0.5748	0.5923	**0.6168**
Personal	Precision	**0.6247**	0.5525	0.5462	0.5995	0.5835	0.5563	0.5771	0.5952	0.5898
	Recall	0.8123	0.8160	0.8252	0.8271	0.8388	0.8394	0.8498	0.8332	**0.8505**
	F_1	**0.7063**	0.6589	0.6574	0.6951	0.6882	0.6691	0.6874	0.6944	0.6966
Argum	Precision	**0.5657**	0.5636	0.5398	0.5594	0.5457	0.5631	0.5434	0.5581	0.5458
	Recall	0.6614	0.7114	**0.7769**	0.6722	0.7652	0.7250	0.7652	0.6908	0.7632
	F_1	0.6098	0.6289	0.6370	0.6107	**0.6371**	0.6339	0.6355	0.6174	0.6365
$>$ BL	Precision		1	3	2	6	4	5	4	6
	Recall		5	2	5	2	5	3	5	3
	F_1		2	2	4	3	4	4	5	5

Table 1: Classification results: precision, recall and F_1-scores per method and category. BL indicates the baseline from Schabus et al. (2017). Values outperforming the baseline are underlined, the best value per row is in **bold**. The last three rows indicate the number of times the baseline was outperformed per method and measure.

grouped them under the terms holism, specificity, "messy" data and integration, highlighting identified differences between academic and industrial perspectives. Finally, we reported new results on our data set to illustrate some of these challenges and proposed solutions more concretely.

5. Acknowledgments

This research was partially funded by the Google Digital News Initiative.[4] We thank DER STANDARD and their moderators for the interesting collaboration.

6. Bibliographical References

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proc. ICML*, pages 1188–1196, Bejing, China.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proc. ACL*, pages 142–150, Portland, OR, USA.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proc. ACL*, pages 115–124, Ann Arbor, MI, USA.

Sakunkoo, P. and Sakunkoo, N. (2009). Analysis of social influence in online book reviews. In *Proc. AAAI*, pages 308–310, San Jose, CA, USA.

Schabus, D., Skowron, M., and Trapp, M. (2017). One million posts: A data set of German online discussions. In *Proc. SIGIR*, pages 1241–1244, Tokyo, Japan.

Snyder, B. and Barzilay, R. (2007). Multiple aspect ranking using the good grief algorithm. In *Proc. ACL*, pages 300–307, Rochester, NY, USA.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. EMNLP*, pages 1631–1642, Seattle, WA, USA.

[4] https://www.digitalnewsinitiative.com

Community-Driven Crowdsourcing: Data Collection with Local Developers

Christina Funk, Michael Tseng, Ravindran Rajakumar, Linne Ha
Google
1600 Amphitheatre Parkway, Mountain View, CA 94043
{christinafunk, michaeltseng, ravirajakumar, linne}@google.com

Abstract

We tested the viability of partnering with local developers to create custom annotation applications and to recruit and motivate crowd contributors from their communities to perform an annotation task consisting of the assignment of toxicity ratings to Wikipedia comments. We discuss the background of the project, the design of the community-driven approach, the developers' execution of their applications and crowdsourcing programs, and the quantity, quality, and cost of judgments, in comparison with previous approaches. The community-driven approach resulted in local developers successfully creating four unique tools and collecting labeled data of sufficiently high quantity and quality. The creative approaches to the rating task presentation and crowdsourcing program design drew upon developers' local knowledge of their own social networks, who also reported interest in the underlying problem that the data collection addresses. We consider the lessons that may be drawn from this project for implementing future iterations of the community-driven approach.

Keywords: crowdsourcing, data diversity, sentiment annotation, tools and platforms

1. Introduction

Labeled datasets for machine learning algorithms can contribute to robust and accurate models. Crowdsourcing presents one way to obtain more representative labeled datasets, but it is difficult to recruit crowd contributors from a diverse cross-section of communities. Furthermore, transactional crowdsourcing frameworks, where crowd contributors are remote performers of an annotation task designed solely by the modelers of the problem, do not fully benefit from the potential insights of the crowd. In this paper, we present an approach for involving developer communities in both annotation task development and crowdsourcing program design, so that they engage their local communities to build labeled datasets that represent their world.

Developers rooted in local communities are ideal interlopers to create tools that are effective for their particular context, as they can employ their community insights to build labeled datasets used to inform models that exemplify their locale and its people. While doing this, they can also contribute to a research space to which they might not normally have access, and as they learn, we learn with them.

For the implementation of this approach, we chose a problem in the sentiment annotation domain, where diverse judgments representative of several communities are valuable due to the subjective nature of evaluating sentiment in a given context. Specifically, we chose to collect judgments on the toxicity of Wikipedia discussion comments, where a "toxic" comment is defined as any kind of hateful, aggressive, or disrespectful comment that is likely to make someone leave a discussion. The Conversation AI team at Jigsaw, a part of Alphabet, works on technology to promote civility in online discourse. They have shown that a classifier trained using data labeled through crowdsourcing can be as effective in identifying personal attacks as the aggregate work of three contributors (Wulczyn et al., 2017).

Toxicity judgment is well-suited for the community-driven approach because toxic language is one component of the broader global problem of online harassment. According to the Pew Research Center, 41% of Americans have been the targets of harassment online, from offensive name-calling to physical threats (Duggan, 2017). In a survey commissioned by Amnesty International of women in eight countries (Denmark, Italy, New Zealand, Poland, Spain, Sweden, the United Kingdom, and the United States), 23% reported that they had been harassed or abused online, and of these, 41% felt that their physical safety was threatened (Magill, 2017). In China, over 56% of students polled at 1,438 secondary schools reported that they had been a target of online bullying (Zhou et al., 2013).

With the goal of harnessing community insights around this problem, we partnered with the Conversation AI team to test the viability of a crowdsourcing approach where both the application development and the data collection are external to the company. The expected benefits of this approach include: distributing and diversifying the sources of labeled data for a given annotation task; motivating developers and their communities to become participants in the ongoing creation of data resources; and replacing one-off tools development and maintenance with developers who are invested directly in the creative process.

2. Background

We conducted two rounds of data collection with internal or onsite developers that informed our iteration toward the community-driven approach.

2.1. Vendor-Mediated Approach

To establish a baseline for the task, we started with an industry-standard approach of developing a task interface and contracting a vendor to execute the annotation operations. The application we developed for the toxicity rating task is hosted on a Google platform. The user interface displays the Wikipedia comment, presents a range of rating options from very toxic to very healthy, supplies a box for optional comments, and lets the contributor submit the judgment. The dataset consists of 4,500 Wikipedia comments that were annotated at a redundancy of 10, resulting

in 45,000 judgments. The contributors who provided the judgments were managed by a vendor company that was selected after completing a pilot task that demonstrated delivery of labeled data at an acceptable level of quality.

2.2. Internally Managed Approach

In preparation for the community-driven approach, we worked with contractors in Singapore who had no prior knowledge of the project. We split them into two teams and conducted a three-day boot camp on Android development and the annotation task. Each team used the rest of the week to build an Android application, following simple designs, which incorporated a display of the Wikipedia comment to be rated, buttons for rating options, and space for optional comments. The backend for storing task data and judgments was again a Google platform. This exercise demonstrated that it is feasible for moderately experienced developers to create custom crowdsourcing applications without deep background knowledge of the task and within a short time frame.

Outside volunteers from local universities spent an hour of their time rating the toxicity of comments in exchange for a Google office tour and a token of appreciation (a canvas bag). Over three days, they submitted 11,809 judgments.

3. Community-Driven Approach

The community-driven approach was conducted as a challenge event in Colombo, Sri Lanka. 17 teams submitted proposals; we selected the most promising proposals, and four teams of three developers participated in the event. We provided one day of boot camp to brief the participants on the toxicity rating task, technical implementation details, and crowdsourcing program design. The backend used to store task data and judgments was available through a public API developed by the Conversation AI team.

During the Singapore round, we had observed that crowd contributors tended to get bored with the same repetitive task for an hour. We encouraged the Sri Lanka developers to be creative in their presentation of the task, and they came up with different gaming options. Since the game concepts for the applications were more involved in this round, we gave the teams a full week to complete development, and some continued to improve their applications throughout the data collection period.

As in the previous rounds, the main task was still to rate the toxicity of Wikipedia comments, but developers were given the option to extend the task to include classification of the toxicity type: insult, identity hate, obscenity, or threat.

Developers tapped into their own local networks to recruit and motivate crowd contributors. We gave the developers a month to find contributors and collect judgments, and we paid them awards for successfully building applications and for the quantity and quality of judgments that they submitted. Teams were permitted to use their own quality control methods to decide which judgments to submit. The API interspersed golden items (i.e., items with expected ratings withheld from the developers) into the dataset, allowing us to estimate the accuracy of judgments in each team's submissions based on the percentage of golden items returned with ratings consistent with the expected ratings.

We discuss each team's Android application and crowdsourcing plan in the following sections.

3.1. Forager Application

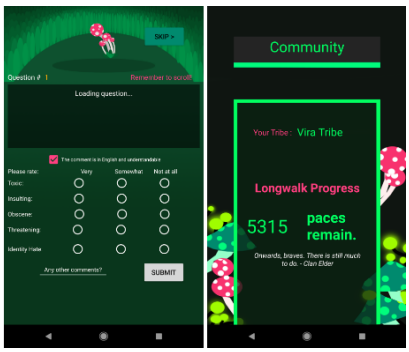


Figure 1: Screenshots from the Forager application.

In the Forager application, each user is conceptualized as a member of the same tribe on a journey. Mushrooms representing Wikipedia comments are to be rated for toxicity. In the game world, these judgments inform whether the mushrooms are safe to eat as the tribe advances along its journey. The team used word of mouth and social media platforms to advertise the application. To motivate contributors, the team offered to donate a portion of its award earnings to charity. The game appealed to contributors' sense of solidarity by placing everyone in the same tribe on a common journey. After completing a certain number of judgments, the player acquires mushrooms that count as credit for in-app purchases, designed to entice repeated engagement.

3.2. Jury Application

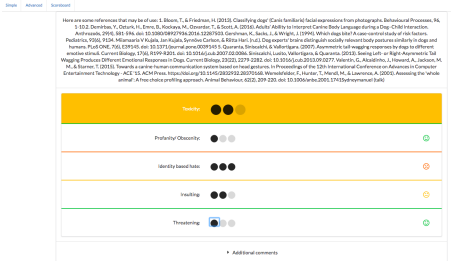


Figure 2: Screenshot from the Jury web application.

In the Jury application, users rate the toxicity of comments to earn coins. (The team's initial idea was to have users deliberate on a rating but scaled the concept down given the short development time frame.) In "simple" mode, users rate toxicity on a three-point scale; in "advanced" mode, users also classify the type of toxicity. In addition to an Android application, the team created a web application.

To recruit contributors, the team messaged friends, classmates, and colleagues about the application and shared details on social media. The application has a leaderboard and users earn coins for completing judgments. As a final motivation, the team took the three contributors with the highest number of judgments to watch the film *Justice League*.

3.3. ToxicMania Application

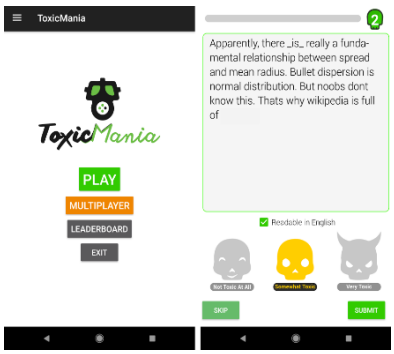


Figure 3: Screenshots from the ToxicMania application.

In single-player mode of the ToxicMania application, users compete to be at the top of the leaderboard, and they earn achievement badges as they rate comments. In multiplayer mode, users compete to outpace their friends in completing the most judgments.

The team recruited contributors through a Facebook page and through awareness sessions at their university. To motivate people to continue using the application, the team offered T-shirts to the highest-ranked players on the leaderboard. The team chose to filter out task items with lengthy comment text to reduce fatigue while playing the game.

3.4. MemifyX Application



Figure 4: Screenshots from the MemifyX application.

In the MemifyX application, users complete judgments to earn "swipes" to view or upload Internet memes. The application includes an achievements page where users earn recognition based on the judgments they have contributed. The team distributed the application to friends and sought to maintain engagement by offering the entertainment product of memes and the chance to create and view them using points. They also printed stickers to give to contributors.

4. Results

The community-driven approach of incentivizing external developers to build annotation applications and to crowdsource data collection through their own networks resulted in four different and creative solutions that brought in a high volume of labeled data at an acceptable level of accuracy, all within the time frame of a few weeks. Across all four applications, we collected 69,042 judgments with an average estimated accuracy of 76.55%. This approach was also more cost-effective than the previous approaches.

4.1. Volume

Team	Number of Judgments
Forager	23,085
Jury	22,436
ToxicMania	19,252
MemifyX	4,269

Table 1: The number of judgments submitted by each application created in the community-driven approach.

Among the four applications, Forager submitted the most judgments, while MemifyX submitted the fewest by a large margin (see table 1). One reason for this could be a waning of interest on the part of the team in promoting its application to potential contributors. It is worth noting that developers bring with them their own motivations, which may change during the course of the challenge. Including multiple teams in the challenge helped mitigate this factor. Each team's scheme for engaging contributors also likely factors into the quantity results. The Forager team motivated contributors by encouraging group togetherness and offering to donate some of the award proceeds to charity. One Forager developer said that while the points system drew in some, the application "managed to attract quite a few contributors due to our concept and artwork." ToxicMania application users reported on its Google Play Store page that it let them "contribute to something that matters while you are on commute" and "help the world while having fun." The MemifyX team, however, concentrated contributors' motivation on a scarcity of access to memes within the application. With easier ways of finding memes outside of the application, contributors may have found this motivation less appealing than contributing to a good cause.

4.2. Quality

Team	Percentage of Golden Items Correct	Golden Answers Submitted	Baseline
Forager	77.69%	3,178	57.05%
Jury	79.36%	1,857	54.53%
ToxicMania	70.84%	562	47.85%
MemifyX	81.34%	201	63.81%

Table 2: The accuracy of golden item judgments for each application in the community-driven approach.

Table 2 shows the breakdown for each application of the percentage of golden items returned with ratings matching the (withheld) expected ratings. In this case, "accuracy" refers to the percentage of golden items the team returned with ratings that matched those previously established by

the Conversation AI team. Because each team's application collects and resolves judgments from crowd contributors in a different way, we consider the ratings only in the final judgments submitted by each team. The baseline for each team indicates the expected percentage of golden items that would be returned with correct ratings by random guessing, given the distribution of golden items in that team's queue. All four teams maintained above 70% estimated accuracy, which was acceptable to the Conversation AI team and indicates that for this particular annotation task, it is possible to obtain labeled data of acceptable quality without hiring vendors, building applications internally, or finding and supervising crowd contributors.

4.3. Cost

	Vendor-Mediated	Internally Managed	Community-Driven
Number of Judgments	45,000	11,809	69,042
Cost per Judgment	$0.1168	$0.9091	$0.0628

Table 3: The number of judgments collected and cost per judgment broken down by approach.

For the vendor-mediated approach, the cost per judgment is the flat rate negotiated with the vendor. For the internally managed approach, the cost per judgment factors in payments to the onsite contractors and the cost of the canvas bag gifts for the crowd contributors. For the community-driven approach, the cost per judgment factors in: (1) a fixed award for each team that successfully developed an application; (2) a second fixed award for each team that collected at least 10,000 judgments while maintaining at least 70% estimated accuracy; and (3) a variable award for each team calculated by multiplying the number of judgments by a rate that scales with estimated accuracy.

We do not factor into the costs the company's technical investment for each approach. Externalizing annotation application development reduces Google's outlays toward recurring tools design and implementation, but it also requires ongoing maintenance of a public API.

In terms of cost per judgment, the community-driven approach is the clear winner (see table 3). This, however, is from the point of view of the payer, not controlling for other factors that may affect cost, such as the country location or quality of the data.

5. Discussion and Conclusions

Developers customized tools and incentives to fit the interests of their local community, accessing their own social networks into which we lacked insight. This community-driven approach allowed us to diversify and parallelize data collection efforts as we partnered with developer teams on different ways to solving the same problem. It afforded developers more creative control over their applications, and rather than being limited to a strict list of requirements, they could leverage their understanding of local communities and to become entrepreneurs who decided the designs and strategies that would result in the highest returns on their efforts. The approach enabled us to collect more data at a lower cost per judgment than we had previously, while still maintaining an acceptable level of overall accuracy.

The community-driven approach requires a public API and has risks, giving us less control over the development or data collection processes. Instead, it relies on the robustness of the quality control methods. It also may introduce group bias, as developers' incentive schemes could appeal to particular populations. Completing the same task in multiple communities is a possible way to mitigate this effect.

In future iterations, we plan to expand to new locations and annotation tasks. While most of the participating developers were comfortable with developing an application in a hackathon format, they were not as familiar with the basics of crowdsourcing. In order to obtain a higher quality and quantity of labeled data, we would like to provide a more in-depth training curriculum on how to create crowdsourcing programs that optimize crowd contributors' engagement and minimize their biases in judgments.

We will also consider allowing developers to choose their platform rather than requiring an Android application. The Jury team reported that potential contributors were less inclined to download an application, which prompted the team to create a web application so contributors could start rating as soon as they received the link. In Sri Lanka, 56.9% of devices used to connect to the Internet in the first half of 2017 were smartphones, while 38.1% were desktops or laptops (Department of Census and Statistics, 2017). Relaxing platform restrictions would allow developers to accommodate the usage habits of their communities and potentially expand their contributor base.

6. Acknowledgments

We thank our colleagues in Google Research and Machine Intelligence as well as Lucas Dixon, C. J. Adams, and the Conversation AI team for their work on the project's design and implementation. We also thank Keshan Sodimana for his help with the Sri Lanka event. We are grateful to the developers and crowd contributors in Singapore and Sri Lanka whose participation made this project possible. We also thank our LREC reviewers for their insights.

7. Bibliographical References

Department of Census and Statistics. (2017). Computer Literacy Statistics–2017 (First six months). Technical report, Battaramulla, Sri Lanka, June.

Duggan, M. (2017). Online harassment 2017. Technical report, Pew Research Center, 11 July.

Magill, T. (2017). Online abuse and harassment. Technical report, Ipsos MORI on behalf of Amnesty International, 19 November.

Wulczyn, E., Thain, N., and Dixon, L. (2017). Ex machina: Personal attacks seen at scale. *Proceedings of the 26th International Conference on World Wide Web*, pages 1391–1399.

Zhou, Z., Tang, H., Tian, Y., Wei, H., Zhang, F., and Morrison, C. M. (2013). Cyberbullying and its risk factors among Chinese high school students. *School Psychology International*, 34(6):630–647.

Building Open Javanese and Sundanese Corpora for Multilingual Text-to-Speech

Jaka Aris Eko Wibawa[2], Supheakmungkol Sarin[1], Chenfang Li[3], Knot Pipatsrisawat[1], Keshan Sodimana, Oddur Kjartansson[1], Alexander Gutkin[1], Martin Jansche[1], Linne Ha[1]
[1]Google,[2]Google (on contract from Pactera Technologies NA Inc),[3]Google (on contract from Optimum Solutions Pte Ltd)
{jwibawa,mungkol,chenfangl,thammaknot,ksodimana,oddur,agutkin,mjansche,linne}@google.com

Abstract

We present multi-speaker text-to-speech corpora for Javanese and Sundanese, the second and third largest languages of Indonesia spoken by well over a hundred million people. The key objectives were to collect high-quality data in an affordable way and to share the data publicly with the speech community. To achieve this, we collaborated with two local universities in Java and streamlined our recording and crowdsourcing processes to produce corpora consisting of 5,800 (Javanese) and 4,200 (Sundanese) mixed-gender recordings. We used these corpora to build several configurations of multi-speaker neural network-based text-to-speech systems for Javanese and Sundanese. Subjective evaluations performed on these configurations demonstrate that multilingual configurations for which Javanese and Sundanese are trained jointly with a larger corpus of Standard Indonesian significantly outperform the systems constructed from a single language. We hope that sharing these corpora publicly and presenting our multilingual approach to text-to-speech will help the community to scale up text-to-speech technologies to other lesser resourced languages of Indonesia.
Keywords: low-resource languages, corpora, multilingual, text-to-speech

1. Introduction

One of the important trends in modern speech and language technology is the increasing focus on scaling up the current state of the art to the large number of language communities in the world. Progress in this direction is difficult, because, more often than not, the languages in the long tail of the distribution of the majority of the world's languages lack adequate linguistic resources required for supporting speech and natural language research (Besacier et al., 2014; O'Horan et al., 2016).

A prime example of such a challenge are the languages of Indonesia. Lewis et al. (2015) identify 707 distinct living languages (not dialects) that are spoken throughout the archipelago by over 255 million people. About twenty of these languages are spoken by over a million people. Standard Indonesian is the official national language of Indonesia and is spoken natively or as a second language by more than 200 million people (Paauw, 2009). Javanese with roughly 90 million native speakers and Sundanese with approximately 40 million native speakers constitute the two largest regional languages of Indonesia. Unlike Indonesian, which received a lot of attention over the years, e.g. (Sakti et al., 2008; Manurung et al., 2010; Koto, 2016), both Javanese and Sundanese are currently under-resourced due to the lack of openly available high-quality corpora.

We address this problem for Javanese and Sundanese by building crowdsourced text-to-speech corpora for these languages. This work is part of an ongoing effort by Google to build multi-speaker corpora for low-resource languages in an affordable way and to share the resulting data publicly. Building on our initial work on Bengali (Gutkin et al., 2016), Sinhala and other languages, we aimed to design the process from the ground up, starting with script building and recording equipment selection to logistics and operations. Similar to other languages, we are releasing the resulting corpora under a liberal license on OpenSLR[1].

We collaborated with two universities in Indonesia to conduct data collections locally. For Javanese we worked with Universitas Gadjah Mada (UGM), Faculty of Cultural Science in Yogyakarta. For Sundanese we worked with Universitas Pendidikan Indonesia (UPI), Faculty of Language and Literature Education in Bandung. The universities assisted us with finding volunteers to help manage the data collection, as well as with adequate recording environments. Together with several researchers from Reykjavík University in Iceland we also used this opportunity to collect open-source Javanese and Sundanese[2] data for automatic speech recognition (ASR) (Guðnason et al., 2017).

We used the collected multi-speaker data to build statistical parametric speech synthesis (SPSS) systems for Javanese and Sundanese. While it is possible to use single-language corpora to bootstrap text-to-speech for individual languages, we also experimented with an alternative approach that capitalizes on language similarities between Javanese and Sundanese on the one hand, and Standard Indonesian on the other. We hypothesize that constructing a jointly trained multilingual system may result in significant improvements over the systems constructed using a "classical", single-language approach.

2. Overview of Javanese and Sundanese

Since we are building speech corpora, a key consideration is the design of the phoneme inventory of each language. In order to facilitate multi-lingual experiments, the phoneme inventories of different languages have to be bridged somehow. In the present case this was particularly straightforward, as the languages are related (they all belong to the Malayo-Polynesian subgroup of Austronesian) and their phoneme inventories overlap heavily.

[1] http://www.openslr.org/
[2] http://www.openslr.org/{35,36}/

Language		*Segments*
Indonesian	id	32 see below (23 consonants, 9 vowels)
Sundanese	su	33 Indonesian plus ɤ
Javanese	jv	35 Indonesian plus ʈ ɖ ɔ
Joint total		36 (25 consonants, 11 vowels)

Consonants

p b	t d	ʈ ɖ	tʃ dʒ	k g		ʔ
m	n		ɲ	ŋ		
	r					
f	s z		ʃ	x		h
w	l		j			

Vowels

i			u
e	ə	ɤ	o
a			ɔ
	ɑɪ	ɑʊ	ɔɪ

Table 1: Joint phoneme inventory of Indonesian, Javanese, and Sundanese in International Phonetic Alphabet notation

We started from a pre-existing phoneme inventory of Standard Indonesian, which consists of 32 segmental phonemes. Table 1 shows the basic Indonesian inventory in IPA notation, with the additional segments for Javanese and Sundanese highlighted. The overall joint phoneme inventory consists of 36 segments.

Sundanese has one additional vowel phoneme, which is not shared with Indonesian or Javanese. This is the back unrounded vowel /ɤ/, written as *eu* in the modern Sundanese orthography. It occurs in words like *seueur* /sɤ.ɤr/ (many) and *henteu* /hən.tɤ/ (not) and contrasts with schwa /ə/.

Javanese has three additional phonemes in our analysis, which are not shared with Indonesian or Sundanese. One is the open back vowel /ɔ/. The other two are, remarkably, the retroflex stops /ʈ/ and /ɖ/.

The retroflex stops occur, inter alia, in loanwords that ultimately trace back to languages of India, for example *garudha* /ga.ru.ɖɔ/ (Garuda, eagle; from Sanskrit *garuḍa* or *kutha* /ku.ʈɔ/ (town; compare Indonesian *kota*). Here we also see that orthographic final *a* is read as phonemic /ɔ/. By contrast in *gajah* /ga.dʒah/ (elephant) the last vowel does not round to /ɔ/, due to the presence of a final consonant. Phonemic /ɔ/ further arises as the reading of orthographic *o* in closed syllables, potentially spreading to preceding syllables: compare *koyo* /ko.jo/ with *koyok* /kɔ.jɔʔ/.

Our phoneme inventory for Javanese is largely identical to the one described by Ogloblin (2005).[3] Ogloblin lists an additional open-mid front vowel /ɛ/, which is not reflected in our inventory. We chose to treat this distinct sound as an allophone of /e/, but this choice is debatable.

Our joint inventory inherited several choices made for Indonesian, which we decided not to revisit. For example, the diphthongs are rare and could easily be analyzed as a combination of two vowels or as a vowel plus off-glide. The diphthong /ɔɪ/ is exceedingly rare and occurs mostly in loanwords. Notably all inventories include /ʃ/ and /x/ as distinct phonemes, as they have a robust presence in loanwords (and have distinct letters in e.g. the traditional orthography). Relatively recent orthographic reforms have made the modern Javanese and Sundanese orthographies highly regular and phonemically transparent. However, in conventional usage several ambiguities arise. Most notably, and shared with Indonesian, the distinction between /ə/ and /e/ is not always reliably indicated in the orthography. While /e/ can be written as *é*, this is not always done systematically, and often it appears as plain *e*. As a result, the orthography alone is insufficient to derive accurate pronunciations for a text-to-speech system. We therefore asked native speakers to transcribe substantial pronunciation dictionaries for Javanese (54,000 words) and Sundanese (42,000 words).

3. Data Collection

3.1. Recording Script

To build the script efficiently, we asked native speakers to list some of the important named entities (e.g., local place names), time expressions (e.g., months of the year), numbers (e.g., all numbers smaller than 100) and so on.

For Javanese, we asked the native speakers to create sentences that include these elements. The sentences should be easy to read, rich enough to include orthographic variants, not offensive, and span five to twenty words in length. For Sundanese, we scaled the process further by starting from *templates* constructed from elements in the the above lists. For example, "[global celebrity name] goes to [global place name city] with [local celebrity name] during [time expression season]" is one such template where the fillers are shown in square brackets. We then generated sentences from these templates, which were reviewed by native speakers. Finally, we computed grapheme and phoneme coverage of the resulting sentences to make sure that we cover most sounds of the language.

3.2. Hardware and Recording Setup

The equipment used was an ASUS Zenbook UX305CA fanless laptop, a Neumann KM 184 microphone, Blue Icicle XLR-USB A/D converter and a portable acoustic vocal booth[4]. Parameters such as distance between speaker and microphone, height of the microphone and the angle at which it is pointed to the speaker were kept as constant as possible.

All audio was recorded using ChitChat, our in-house recording tool described in (Gutkin et al., 2016). ChitChat is a web-based recording software that allows audio data to be collected, managed and quality controlled. Each volunteer is presented with a series of sentences assigned to them for recording. The tool records at 48 kHz (16 bits per sample), detecting audio clipping to ensure quality, and ambient noise prior to recording each sentence, with a high noise level triggering an alert preventing further recording. Audio is initially stored on the client, and is uploaded asynchronously to a server when requested.

3.3. Crowdsourced Data Collection

The staff of the Javanese Literature Department of UGM put us in contact with the volunteers from that university who helped record samples. For Sundanese, we worked with UPI who helped us find volunteer speakers. A portion of the recordings was done at the student-run CompFest 2016 (an annual Computer Science exhibition event organized by students from the Faculty of Computer Science at

[3] http://phoible.org/inventories/view/1675. Their /y/ should be understood as IPA /j/.

[4] https://www.vocalboothtogo.com/

Gender	Collected Data		Cleaned Data	
	jv-ID	su-ID	jv-ID	su-ID
Female	3912 (20)	2475 (21)	2864 (19)	2401 (21)
Male	3918 (19)	2594 (21)	2958 (19)	1810 (21)

Table 2: Javanese and Sundanese database details.

Universitas Indonesia)[5]. Volunteers were between 18 and 35 years old.

We asked the speakers to read the script that we prepared. Each one spent an average of one hour contributing approximately 80 recordings. They were advised to read the script naturally. The Javanese recordings were done at UGM; half of the data was collected in a studio, the rest in a quiet room, inside a portable vocal booth. All the Sundanese recordings were collected in a portable vocal booth placed inside a quiet room. We performed quality control (QC) using ChitChat after the data was collected. Our objective was to make sure that the text matches the recordings and that the recordings are free of any artifacts: background noise, breathing, extraneous noises, etc.

Table 2 shows the details for the Javanese and Sundanese databases for female and male speakers after the data collection was complete (columns on the left) and after the recordings passed the QC (columns on the right). For each stage a number of recorded sentences along with the number of speakers (in parentheses) are shown. The total duration of the female multi-speaker datasets is 3.5 hours for Javanese and 3.2 hours for Sundanese. For the male multi-speaker datasets the total duration is 3.5 hours for Javanese and 2.2 hours for Sundanese.

4. Experiments

The experiments focus on building individual text-to-speech systems for Javanese and Sundanese languages from the multi-speaker data described in Section 3. The goals of the experiments are as follows: The first goal is to determine whether the individual corpora that we collected are good enough to construct a text-to-speech voice of adequate quality for the languages in question. The second goal is to see whether the quality of the systems can be improved by utilizing joint training, where both languages are trained together in a single model. In addition, we are interested in including a high-quality single-speaker Indonesian language database in this experiment. Our hypothesis is that the presence of a larger and professional-quality corpus from a related language should positively impact the training (Li and Zen, 2016; Gutkin, 2017).

4.1. Language Data and System Details

In this experiment we use the Javanese, Sundanese (both described in Section 3) and Indonesian datasets. The Indonesian dataset is the largest of the three, being approximately six times bigger than Javanese and Sundanese datasets. The original audio for all three corpora was recorded at 48 kHz. We selected the female multi-speaker datasets for both Javanese and Sundanese for the purpose of these experiments.

[5] http://www.compfest.web.id

Code	Description	Speaker	
		jv	su
id	Indonesian single speaker		
jv	Javanese single speaker	✓	
jv+id	Javanese with Indonesian	✓	
jv+su+id	Javanese with Sundanese and Indonesian	✓	✓
su	Sundanese single speaker		✓
su+id	Sundanese with Indonesian		✓
su+jv+id	Sundanese with Javanese and Indonesian	✓	✓

Table 3: Different acoustic model configurations.

We constructed several statistical parametric speech synthesis (SPSS) systems for Indonesian, Javanese, and Sundanese. Each system consists of a linguistic front-end, followed by an LSTM-RNN acoustic model, and finally a vocoder (Zen and Sak, 2015; Zen et al., 2016). The details of the LSTM-RNN model configuration (neural network training parameters, multilingual input and output features and so on) are similar to the ones described by Gutkin (2017). The main difference is that in this work we use fewer neural network parameters because we are dealing with a much smaller set of languages.

For each of the Javanese and Sundanese multi-speaker datasets we selected the "best" (according to our subjective analysis) sounding speaker to be used as the target speaker for biasing the acoustic models during run-time. The use of speaker features have been demonstrated to improve the quality of a multilingual multi-speaker synthesis (Li and Zen, 2016; Gutkin, 2017). Various LSTM-RNN configurations we built are described next.

4.2. System Configurations

Overall we constructed seven configurations corresponding to various combinations of languages shown in Table 3. We built a single-speaker Indonesian system (id) from an Indonesian corpus to see how well the Indonesian system fares on its own, as a reference point. Similarly, we used the Javanese and Sundanese multi-speaker corpora individually to construct single-language (but multi-speaker) systems jv and su, which serve as baselines in our experiments. In addition, we jointly trained Javanese with Indonesian (jv+id) and Sundanese with Indonesian (su+id) to produce the bilingual Javanese and Sundanese systems. Finally, we jointly trained a single combined Javanese, Sundanese and Indonesian acoustic model in order to produce a trilingual Javanese (jv+su+id) and Sundanese (su+jv+id) systems.

Each resulting system was evaluated using a subjective Mean Opinion Score (MOS) listening test. For each test we used 100 sentences not included in the training data for evaluation. Each rater was a native speaker of the language and was asked to evaluate a maximum of 100 stimuli. Each item was required to have at least three ratings. The raters used headphones. After listening to a stimulus, the raters were asked to rate the naturalness of the stimulus on a 5-point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). Each participant had one minute to rate each stimulus. The rater pool included eight raters for Javanese and seven raters for Sundanese. For each language, all configurations were evaluated in a single experiment. For Indonesian, which is the "bigger" reference language in our experiments, we

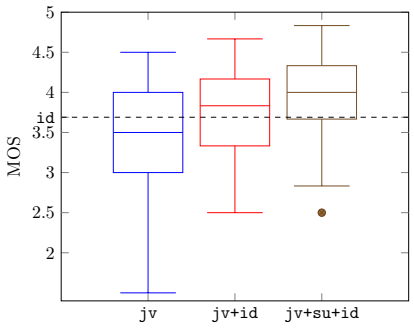


Figure 1: Mean Opinion Scores (MOS) box plot for three Javanese configurations.

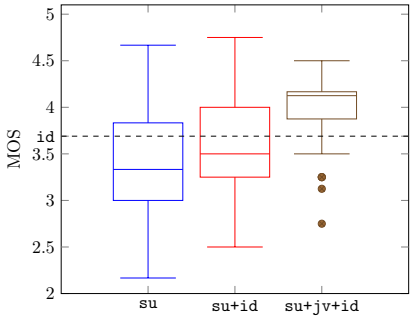


Figure 2: Mean Opinion Scores (MOS) box plot for three Sundanese configurations.

required over eight ratings per item and a larger rater pool.

4.3. Evaluation Results and Discussion

The distribution of the MOS scores for each of the three Javanese and three Sundanese systems described in the previous section are shown in Figures 1 and 2, respectively. The distributions are represented in box plot format, where, for each distribution the minimum, first quartile, median, third quartile, and maximum are displayed. Outlier values are shown as circles. The reference MOS score for Indonesian (`id`) is shown as a dashed line. The distributions are computed over all the available ratings for the 100 stimuli in each language.

Table 4 shows the mean opinion scores corresponding to the box plots for Javanese (Figure 1) and Sundanese (Figure 2). The mean opinion score for Indonesian is included for reference. Each mean opinion score is shown along with the corresponding 95% confidence interval (Recommendation ITU-T P.1401, 2012). The best scores are shown in bold.

As can be seen from the MOS box plots and Table 4, the baseline Javanese (`jv`) and Sundanese (`su`) systems bootstrapped solely from their respective multi-speaker corpora are inferior to the Indonesian system (`id`). In particular, the Sundanese system is significantly worse. This is not surprising because the Indonesian corpus is significantly larger, of studio quality, and consists of recordings from a single professional speaker.

These results, however, are improved when Javanese is jointly trained with Indonesian (`jv+id`) and Sundanese is jointly trained with Indonesian (`su+id`). These new configurations outperform both the baseline systems (`jv` and `su`) and their performance is tied with the Indonesian system (the differences with Indonesian are not statistically significant).

Configuration	Language	MOS
`id`	Indonesian	3.692±0.054
`jv`	Javanese	3.484±0.122
`jv+id`	Javanese	3.780±0.114
`jv+su+id`	Javanese	**3.998±0.103**
`su`	Sundanese	3.333±0.122
`su+id`	Sundanese	3.597±0.096
`su+jv+id`	Sundanese	**4.000±0.061**

Table 4: Subjective Mean Opinion Scores (MOS) (along with 95% confidence intervals) for languages synthesized with various acoustic model configurations. Best scores are shown in bold.

The best results are obtained by jointly training all the available languages. These Javanese (`jv+su+id`) and Sundanese (`su+jv+id`) systems outperform the systems built from two languages and also significantly improve upon the single-language multi-speaker baselines. It is interesting to note that despite using different speaker identity features and linguistic front-ends, the MOS scores for the two trilingual systems are virtually identical. We hypothesize that these additional improvements stem from both the availability of increased amounts of data from a related language (Indonesian) and from more data from a related language recorded in similar conditions (Javanese and Sundanese).

5. Conclusion and Future Work

We assembled open crowdsourced Javanese and Sundanese multi-speaker corpora collected for the purpose of building text-to-speech applications. Building on our previous work on Bengali (Gutkin et al., 2016) we further optimized the low-resource language data collection process to be more affordable.

We used the corpora to build several configurations of text-to-speech systems for Javanese and Sundanese. We demonstrated that the best results are obtained by constructing individual systems that share a multilingual acoustic model that is jointly trained on Javanese, Sundanese and a larger Indonesian dataset. This configuration significantly outperforms the baselines.

We hope that the process, the data, and the approaches described in this paper can be used to scale up our system to other low-resource languages of Indonesia, for example large languages such as Madurese (Davies, 2010) and Minangkabau (Adelaar, 1995).

One potentially very interesting venue for future research is to investigate different approaches to reducing the data collection costs by recording less data without sacrificing the synthesis quality.

6. Acknowledgments

The authors would like to thank Ruli Manurung, Maftuhah Ismail, Neneng Nurjanah, Mollyna Ezyando, Laila Rahmawati and Kharisma Ulinnuha for providing the linguistic expertise and for helping with the data collection. We also thank Universitas Gadjah Mada (UGM) and Universitas Pendidikan Indonesia (UPI) for their help with the recordings.

7. Bibliographical References

Adelaar, K. A. (1995). Minangkabau. In D. T. Tryon, editor, *Comparative Austronesian Dictionary 1*, pages 433–442. Mouton de Gruyter, Berlin / New York.

Besacier, L., Barnard, E., Karpov, A., and Schultz, T. (2014). Automatic Speech Recognition for Under-Resourced Languages: A Survey. *Speech Communication*, 56:85–100.

Davies, W. D. (2010). *A Grammar of Madurese*. Mouton Grammar Library. Mouton de Gruyter, August.

Guðnason, J., Pétursson, M., Kjaran, R., Klüpfel, S., and Nikulásdóttir, A. B. (2017). Building ASR corpora using Eyra. In *Proc. of Interspeech 2017*, pages 2173–2177, Sweden, August.

Gutkin, A., Ha, L., Jansche, M., Pipatsrisawat, K., and Sproat, R. (2016). TTS for Low Resource Languages: A Bangla Synthesizer. In *10th edition of the Language Resources and Evaluation Conference (LREC)*, pages 2005–2010, Portorož, Slovenia, May.

Gutkin, A. (2017). Uniform Multilingual Multi-Speaker Acoustic Model for Statistical Parametric Speech Synthesis of Low-Resourced Languages. In *Proc. of Interspeech 2017*, pages 2183–2187, Sweden, August.

Koto, F. (2016). A Publicly Available Indonesian corpora for Automatic Abstractive and Extractive Chat Summarization. In *10th edition of the Language Resources and Evaluation Conference (LREC)*, pages 801–805, Portorož, Slovenia, May.

Lewis, M. P., Simons, G. F., Fennig, C. D., et al. (2015). *Ethnologue: Languages of the world*, volume 20. Dallas, TX: SIL International, `http://www.ethnologue.com/`.

Li, B. and Zen, H. (2016). Multi-Language Multi-Speaker Acoustic Modeling for LSTM-RNN based Statistical Parametric Speech Synthesis. In *Proc. of Interspeech 2016*, pages 2468–2472, San Francisco, September.

Manurung, R., Distiawan, B., and Putra, D. D. (2010). Developing an Online Indonesian Corpora Repository. In *Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation*.

Ogloblin, A. K. (2005). Javanese. In Alexander Adelaar et al., editors, *The Austronesian Language of Asia and Madagascar*, pages 590–624. Routledge.

O'Horan, H., Berzak, Y., Vulić, I., Reichart, R., and Korhonen, A. (2016). Survey on the Use of Typological Information in Natural Language Processing. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 1297–1308, Japan.

Paauw, S. (2009). One Land, One Nation, One Language: An Analysis of Indonesia's National Language Policy. *University of Rochester Working Papers in the Language Sciences*, 5(1):2–16.

Recommendation ITU-T P.1401. (2012). Methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models. *International Telecommunication Union*, July.

Sakti, S., Kelana, E., Riza, H., Sakai, S., Markov, K., and Nakamura, S. (2008). Development of indonesian large vocabulary continuous speech recognition system within A-STAR project. In *Proceedings of the Workshop on Technologies and Corpora for Asia-Pacific Speech Translation (TCAST)*.

Zen, H. and Sak, H. (2015). Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In *Proc. of Acoustics, Speech and Signal Processing (ICASSP)*, pages 4470–4474, Brisbane, Australia, April. IEEE.

Zen, H., Agiomyrgiannakis, Y., Egberts, N., Henderson, F., and Szczepaniak, P. (2016). Fast, Compact, and High Quality LSTM-RNN Based Statistical Parametric Speech Synthesizers for Mobile Devices. In *Proc. of Interspeech*, San Francisco, September. ISCA.

An Integrated Representation of Linguistic and Social Functions of Code-Switching

Silvana Hartmann, Monojit Choudhury, Kalika Bali
Microsoft Research Lab India
Bangalore, India
nlp@silvanahartmann.de,{monojitc,kalikab}@microsoft.com

Abstract

We present an integrated representation of code-switching (CS) functions, i.e., a representation that includes various CS phenomena (intra-/inter-sentential) and modalities (written/spoken), and aims to derive CS functions from local and global properties of the code-switched discourse. By applying it to several English/Hindi CS datasets, we show that our model contributes i) to the standardization and re-use of CS data collections by creating a resource footprint, and ii) to the study of CS functions by creating a systematic description and hierarchy of reported functions together with the (local and social) properties that may affect them. At the same time, the model provides a flexible framework to add emerging functions, supporting theoretical studies as well as the automatic detection of CS functions.
Keywords: Code-Switching, Multilingual, Bilingual, Pragmatics, Discourse

1. Introduction

Code-Switching (CS), or alternating between two or more languages in a single conversation, is a marked feature of multilingual communities. Linguists have studied this phenomenon in great detail and recently, with the rise of social media, the processing and generating of CS content has gained attention in the NLP community as well. The amount and type of code-switching depends on a number of structural, functional and social factors (Begum et al., 2016). Linguistic studies in the past have focused on two aspects that may be summarized as the "how" and the "why" of code-switching (Poplack, 2015). The "how" aims to explain the grammatical principles that underlie code-switching performance; the "why" aims to explain the function of code-switching in discourse, defining social, pragmatic, and discourse functions of code-switching, for instance *addressee specification*, *emphasis*, or *marking quotations*. These studies are mostly based on limited recordings of conversations, concentrating on a small subset of functions, either linguistic, based on local properties of the discourse, or social and other global aspects. There are no large empirical studies on the interaction of linguistic and social aspects of CS functions. Further, there is no unified theory on the different CS functions and how they are expressed in discourse. Previous theoretical work identifies lists of functions (see e.g., Abdul-Zahra (2010)), and recent work in NLP picks specific functions and aims to identify them in large collections of CS texts (Begum et al., 2016; Rudra et al., 2016). Thus, such studies use different levels of analysis, and often confuse observations (such as observing code-switched *reiteration* or *translation*) with CS functions (*reiteration* is often used for *emphasis*), as in the following example from Begum et al. (2016) that uses the translation of an utterance for emphasis:

> dimaag mein bhoosa bhara hai [gloss:*up in their heads with fodder*]. up in their heads with fodder.

The task of *defining* a comprehensive set of functions is very difficult, if not elusive. The same switching phenomenon, in our example *translation*, is used for emphasis in some communities and for weakening request in others (Poplack, 1988). In such cases it is crucial to distinguish between surface form and (community-specific) function of a switching event. In this work, we present a new integrated framework for representing code-switching functions that builds the foundations for studying the functions of code-switching empirically at a large scale, particularly the interaction between the linguistic and social CS functions. Instead of defining a static set of functions, our framework provides the tools to define functions according to community-specific analyses. We define a flexible set of code-switching functions by first outlining a number of properties that characterize code-switched discourse, and then using these properties to characterize different switching phenomena and associate them to their functions. The proportion of properties and phenomena in a dataset can also be used to create a footprint and compare different CS styles, for instance across language pairs or communities. We apply our framework to several English/Hindi code-switched datasets, i) creating footprints of CS corpora for comparison and standardization (adding a wider context to the statistics introduced by Guzmán et al. (2017)) and ii) systematically describing and deriving CS functions for code-switched conversations.

2. Related Work

Early work on the functions of CS distinguishes between high-level *social functions* and lower-level linguistic functions that depend on the surface form of a specific code-switched utterance. Gumperz (1982) distinguishes between situational and metaphorical/conversational switching. Situational switching operates on the social level and is performed to accommodate different discourse partners, metaphorical switching happens on the utterance level. Later work stresses the discourse-structuring function of CS (Auer, 2013): specific functions are considered to emerge dynamically from the discourse. These functions are strongly tied to a specific code-switched utterance and its context. Following this theory, creating exhaustive lists of CS functions is problematic, since new functions can emerge any time.
Previous work identified a large number of functions of CS, ranging from utterance-level functions (marking switches

from narrative to evaluative language) to conversation-level functions (marking an addressee in a conversation), or even higher-level social functions (showing an affilliation to a certain community). Gumperz (1982) lists six functions: *quotation*, *addressee specification*, *interjection*, *reiteration*, *message qualification*, and the semantic function of *personalization versus objectivization*. Crystal (1987) reports the functions *lack of competence*, *solidarity with a group*, and *communicating a certain attitude*. Reyes (2004) lists functions such as *reported speech*, *imitation quotation*, *turn accommodation*, *topic shift*, *situation switch* (e.g., from academic to non-academic), *insistence non-command* (by reiteration), *emphasis on command*, *clarification/persuasion*, *person specification*, *question shift*, *discourse marker*, and *other*. Abdul-Zahra (2010) adds *quotation* and *addressee qualification* to Crystal (1987)'s list.

Recently, large empirical studies have started to use Twitter as a data source to explore the functions of code-switching: Begum et al. (2016) analyze a set of English/Hindi Tweets and distinguish between several structural and pragmatic functions. Rudra et al. (2016) discover tendencies of bilingual speakers to express negative sentiment in (what is perceived as) their primary language. Rijhwani et al. (2017) is one of the first works to explicitly explore diverse CS communities on Twitter: they study different CS communities based on different language pairs and different localities. Due to the broadcasting nature of Twitter most of this work is focused on linguistic functions, ignoring functions that require global knowledge of the conversation setting.

Quantitative studies have proposed metrics on the distribution of languages in CS corpora. Gambäck and Das (2016) introduce the Code-Mixing Index, and Guzmán et al. (2017) present a set of six metrics to quantify CS statistics, including the equality and burstiness of the language distribution, language entropy, and switching probability. These metrics consider the distribution of the different languages in the code-switched discourse. These metrics can thus, provide detailed statistics on the granularity of code-switching as a means to systematically study CS functions.

Mapping the above-mentioned function lists to a unified model is very difficult for several reasons: i) they cover social and local functions to different degrees, ii) small terminological differences are not clear (the terms *reiteration*, *repetition*, and *translation* are used for similar phenomena), and iii) they show varying degrees of granularity (e.g., whether or not to distinguish between *quotation*, *imitation quotation*, and *reported speech*). Thus, confusing properties of a CS utterance with its function: an observed *translation* can have the function of *emphasis* or *de-emphasis* depending on the community. In general, the relation between surface forms and potential functions available for a surface form is not well-defined. In addition, the lists are often incomplete and tailored to community-specific observations. A comprehensive model should provide descriptive capabilities for representing conflicting functions across communities. We aim to create a framework that represents the properties of CS and uses them to define CS functions dynamically, in an attempt to resolve these problems.

3. New Representation

In this section we introduce our new representation of CS functions that aims to fill the research gap discussed above. Our goal is to organize the functions observed in the literature into a systematic hierarchy that considers the individual functions and the functional levels on which they apply. This new representation of code-switching functions lays the foundation for the study of the interaction between different functional levels of code-switching in the future.

Requirements. Our representation should be a) the basis for analyzing the relationship between social and local CS functions, b) the means to distinguish surface properties of CS phenomena from their functions, c) the means to define functions based on their properties, and d) extensible by other functions and properties, for instance prosody and pauses in speech. We now define code-switching and the analysis dimensions used in our model.

Definition of code-switching. We define code-switching, short CS, as the use of at least two languages by the same speaker that is fluent in these languages within a single conversation. We call the alternating sequences in the different languages **CS segments**. As a prerequisite to defining code-switching functions, we define the unit of analysis for CS functions: a **CS pair** $p({s_1}^a, {s_2}^b)$ consists of two **CS segments** s_1 and s_2 where s_1 is in language l_a and s_2 is in language l_b. The **switch point** marks the boundary between s_1 and s_2. The segments s_1 and s_2 can be of variable length (number of tokens); s_2 extends to the next switch point or the end of the utterance.

3.1. Analysis Dimensions

We define five analysis dimensions relevant to the identification of CS functions, four global dimensions (Granularity, Modality, Discourse, and Social dimension), and the **Local** dimension that considers properties of the specific CS pair. **Granularity** considers properties on the level of CS segments. Its properties are represented by basic CS statistics and the metrics from Guzmán et al. (2017) and Gambäck and Das (2016). **Modality** defines different modes of recording CS language, such as written, spoken, multi-modal (including visual information). **Discourse** relates to the type of communication that is studied, e.g., monolog, dialog, multi-party dialog or broadcasting situations. Certain CS functions such as turn-taking have only been observed for dialogs. The **Social** dimension includes the social aspects of CS discussed earlier, representing information on the relationship and hierarchy between the conversation partners, including familiarity and politeness, but also whether the language community uses certain functions, e.g., repetition for emphasis. In the following paragraphs we provide more detail on the properties we measure for each analysis dimension.

Modality dimension The modality dimension defines different modes of producing code-switched language, such as written, speech-only, and speech with visual information, see the first line in Fig. 1. The categories in this dimension differ in the way an utterance is transmitted and in the presence or absence of linguistic and meta-linguistic information relevant to communication such as prosody, facial

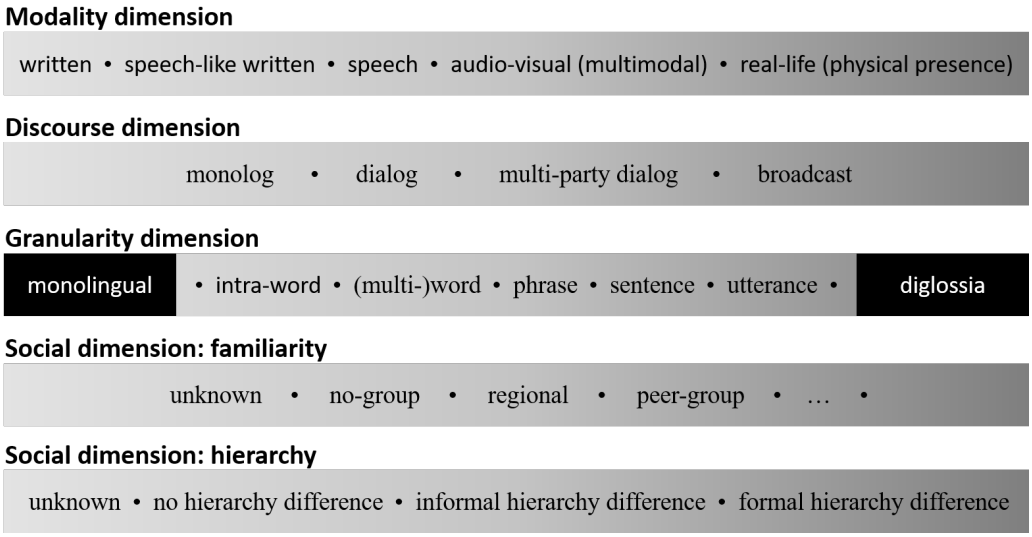


Figure 1: Global analysis dimensions for code-switched discourse.

expression, posture, gaze, gestures, etc. The modality of CS datasets depends on the recording medium. Previous work mostly uses speech recordings (and their transcripts), as well as written texts. It is thus focused on their inherent properties, such as emphasis and hesitations to structure utterances in spoken conversation, and punctuation to structure them in written text. Socio-linguistic studies of CS however frequently refer to discourse-structuring functions of CS, and other ways to express the same functions, for instance gestures when addressing a speaker.[1] We propose the following categories for the modality dimension, of which at least written or audio are typically present: *written, informal speech-like written, speech*, and *audio-visual*.

Discourse dimension The discourse dimension relates to the type of communication that is studied, e.g., whether we are studying at a monolog, dialog, multi-party-dialog or a broadcasting situation (which can be assumed for some Twitter messages), see the second line in Fig. 1. Certain functions that have been observed for CS contributions, such as turn-taking, are only applicable to dialog. The discourse dimension is closely related to the modality dimension: turn-taking is more important in immediate dialog, in particular in speech settings (be it present or via phone call), but may also occur in computer-mediated communication. We propose the following categories for the discourse dimension: *monolog, dialog, multi-party dialog, broadcast*.

Granularity dimension The granularity dimension describes the code-switched segments based on their size and syntactic type. It is illustrated in the third line of Fig. 1. The segment sizes can range from parts of words, single words and multi-word-expressions, via phrases and sentences to full utterances (up to sequences of sentences). One of the most frequently studied phenomena in CS research is intra-sentential switching, which maps to the segments up to the phrase level.

The granularity dimension helps to demarcate the boundaries of switching: we do not consider the use of loan-words by a predominantly monolingual speaker part of CS. The other end of the scale (as shown in Fig. 1) shows diglossia, where one speaker is capable of speaking two languages, but does not mix them in the same conversation, also outside of our definition of CS. Therefore, they are shown with a black background in Fig. 1.

Segments of different type will, of course, occur in the same conversation and in the language of the same speaker or community. The distribution over segment types, or dominant segment size can be used to characterize a specific CS dataset. Syntactic analysis, identifying sentences and phrases, might not be available for the studied language pair. In this case, the number of tokens in a segment can be used as a proxy for these categories. Previous work on CS in Twitter messages exploits quantitative statistics on segment size, studying CS with segment sizes of at least 3 words (Rudra et al., 2016). We propose the following list of categories for the granularity dimension: *sub-word, single word, phrase, sentence, utterance*.

Social dimension Code-switching speakers are known to modulate their use of switching to specific situations, depending on the participants involved. This information is captured in the social dimension. For a particular speaker, different social groups and situations may evoke different degrees of CS and affect their use of CS.

The social dimension includes aspects relevant to the meta-data associated with the conversation partner(s) or the addressed audience, such as formality and politeness, familiarity, and the existence of certain group-internal (cultural) conventions of expressions. Details of the social dimension are typically recorded when gathering code-switched conversations, but they are mostly unknown when studying large Twitter datasets. As a consequence, most work on CS on Twitter did not consider the social dimension.

We propose the following two sub-categories for the social dimension, each with a list of tentative values that are subject

[1] In how far emoticons and other frequent phenomena in written computer-mediated communication, such as expressive lengthening, emulate other modalities is an interesting research question in this context.

to further development: a) familiarity (*unknown*, *no-group* (no common group membership), *regional* (group defined by geographic properties), *peer-group* (various social factors, with further specification of sub-groups)) and b) hierarchy levels (*unknown, no hierarchy difference, informal hierarchy difference, formal hierarchy difference*).

3.2. Local Dimension

Since they play a large role in our representation model, we introduce local properties in detail. We distinguish between *simple atomic properties* on the CS segment level and *compound properties* that build upon atomic properties. Both are relevant to identifying local CS functions. Relevant properties include syntactic, semantic, and pragmatic properties of the CS segment:

Syntactic properties

- $syntactic(s_i) = x \in \{$sentence, clause, phrase, word, sub-word$\}$
- $quotation(s_i) = x \in \{0,1\}$, $x = 1$ if s_i is highlighted by quotation marks[2]
- $continuation(s_i) = x \in \{0,1\}$, $x = 1$ if s_i starts a new sentence
- $tag(s_i) = x \in \{0,1\}, x = 1$ if s_i is a tag (a fixed phrase used for greeting, etc.
- $ne(s_i) = x \in \{0,1\}, x = 1$ if s_i is a named-entity

Semantic properties

- $topic(s_i) = x$ in a set of topics or in a distribution over topics
- $content(s_i) = x$ where x is a semantic representation of s_i (e.g., vector representation, logic form)

Sentiment properties

- $sentiment(s_i) = x \in$ SENT $= \{$-1,0,+1$\}$, where -1 stands for negative, 0 for neutral, and +1 for positive sentiment, cf. Nakov et al. (2013).

Pragmatic properties

- $speechact(s_i) = x \in SACS = \{$question, request, command,...$\}$, cf. Searle (1969).
- $dm(s_i) = x \in \{0,1\}$, where $x = 1$ if s_i contains a discourse marker, cf. Prasad et al. (2014).

Compound properties (on the *CS pair* level) Just like s_i and s_{i+1} are combined to build a CS pair $p(s_i, s_{i+1})$, properties of s_i and s_{i+1} can be combined to build compound properties. Some of the possible combinations are relevant with respect to the analysis of CS functions, for instance discovering the function of sentiment change.

In general, we create compound properties by a) pairing arbitrary properties of s_i and s_{i+1} and b) by comparing the property value of s_i and s_{i+1} for the same property. For the $sentiment$ property, a) leads to a sentiment-pair(s_i, s_{i+1}) = $\langle x, y\rangle$ with 4^2 values for $\langle x, y\rangle$, e.g., $\langle x, y\rangle \in$ SENT×SENT=$\{\langle 0,0\rangle, \langle 0,-1\rangle, \langle 0,+1\rangle, \ldots, \langle +1,+1\rangle\}$. The latter, b) can be used to define topic change: topic-change$(s_i, s_{i+1}) = x \in \{0,1\}, x = 1$ if $topic(s_i) \neq topic(s_{i+1})$.

In the following, we define a number compound properties that are relevant for the definition of CS functions:

- topic-change$(s_i, s_{i+1}) = x \in \{0,1\}, x = 1$ if $topic(s_i) \neq topic(s_{i+1})$
- translation$(s_i, s_{i+1}) = x \in \{0,1\}, x = 1$ if $content(s_i) = content(s_{i+1})$
- sentiment-pair$(s_i, s_{i+1}) = \langle x, y\rangle$ with 4^2 values for $\langle x, y\rangle \in$ SENT×SENT
- speechact-pair$(s_i, s_{i+1}) = \langle x, y\rangle$ with $|SACT|^2$ values for x, e.g., $\langle x, y\rangle \in$ SACT×SACT
- discourse-rel(s_i, s_{i+1}) = x ∈ a set of discourse relations SREL that connect s_i and s_{i+1}

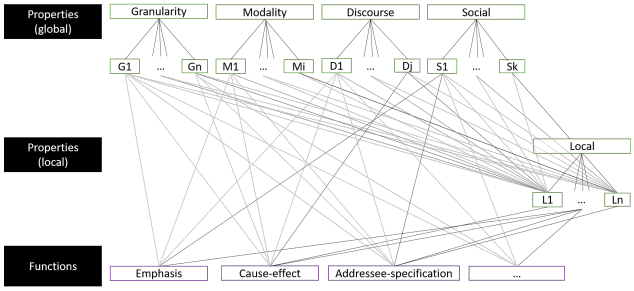


Figure 2: Dependency between CS properties and functions.

These properties can be used to describe a CS phenomenon independently of its function in discourse. We use these properties to define several common CS functions below.

3.3. Deriving CS Functions from Properties

We use simple and compound properties of a CS pair to create operationalized definitions of common CS functions. We show three examples of common CS functions and property derivations here.

Example 1: the *narrative-evaluative* function that can be identified based on the discourse relation between the two CS segments and the sentiment involved, changing from neutral to positive or negative sentiment.

Example 2: *negative reinforcement*: if the sentiment-pair(s_i, s_{i+1}) consists of two sentiment expressions ⟨a,b⟩ and b $= -1$, this indicates negative reinforcement.

Example 3: *marking quotation*: if the second CS segment is highlighted by quotation marks (quotation(s_{i+1})=1), this indicates the use of CS to introduce a quotation or reported speech.

Example 4: the definition of Riloff et al. (2013) can be used to identify *sarcasm* in CS tweets as tweets with positive sentiment in one segment and a negative situation (determined by topic) in the other CS segment.

Example 5: the function of *topic shift* is observed when topic-change(s_i, s_{i+1})=1.

[2]In our introduction of local properties, we focus on their identification in written texts and speech transcripts. For the modalities *audio* and*visual*, quotation marks can be replaced by gestures or prosodic markers of quotation.

The integration of local and global properties in a joint representation of CS functions is a core motivation of this paper: besides local properties, CS functions are often also dependent on higher-level global properties (e.g., translation for *emphasis* is community-specific and depends on social contexts, social and discourse dimensions are relevant for *addressee specification*). These dependencies are illustrated in an abstract fashion (i.e., without relating to specific properties) in Fig. 2. The black lines mark observed dependencies between types of properties, grey lines show hypothetical dependencies that may exist between two property hierarchies or between a property and a function. One goal of this work is to provide a representation framework for observed and expected dependencies for a given CS corpus.

Example 6: a CS function that incorporates local and global properties is *addressee qualification*: to identify addressee qualification, we need some change in the switching behavior (e.g., monolingual to code switching), some deictic aspect (e.g., a greeting tag or a gesture addressing the speaker) and information on the relation between the speaker and addressee, e.g., belonging to a peer group.

Example 7: an instance of the local function *topic switch* that occurs in a certain social setting, for instance a conversation among colleagues in a professional setting, may be a marker of a *situation switch*. The *situation switch* can only be identified if the social factors of the interaction are known.

Summary: this section shows several examples on how local properties indicate local CS functions, and how local and global properties, for instance social properties, can be jointly used to identify CS functions. The local and global properties provide necessary, but not sufficient information to identify CS functions. Still, they help to characterize CS phenomena and provide prerequisites for the large-scale corpus-based analysis of CS functions. Many local properties can be determined using automatic methods and corpus meta-data capture the smaller set of global properties to a certain degree. We believe that a stronger formalization and standardization of corpus meta-data along the lines proposed in this work can support corpus-based analysis of the interaction of local and global CS properties in the future. In the next sections, we show two applications of our integrated representation of CS functions.

4. Application Scenario 1: Footprinting Corpora

To demonstrate its use, we apply our model to several English/Hindi CS datasets. The first dataset D1 is based on speech transcripts of informal conversations, the other datasets M1 to M6 are based on dialogs from six Indian movies titled *Pink*, *Kapoor and Sons*, *Neerja*, *Talvar*, *Ek Main Aur Ek Tu*, and *D-day*. These movies contain code-switched dialog in Hindi and English, whereby the majority of the tokens was labeled as Hindi for all datasets. Transcripts of the movie dialogs are available for non-commercial purposes from `https://moifightclub.com/category/scripts`. We performed automatic language labeling and annotated all datasets with the properties introduced above, providing elaborate annotation guidelines and an extensive training for our three annotators. Automatically created language tags were corrected as needed during the manual annotation of local properties. We plan to publish the annotations and datasets subject to permissions. We present two applications of our representation: footprinting CS discourse according to properties and functions in this section, and supporting automatic functions derivations in Section 5.

Footprinting CS discourse We can use the property values defined above to create *footprints* of a specific code-switched dataset. This is done by accumulating statistics for the different analysis dimensions as shown in Table 2. Fine-grained Granularity properties are provided by the CS metrics from Guzmán et al. (2017) and Gambäck and Das (2016). To save space, this is shown exemplary for the corpora D1 and M1 in Table 1. We refer to the original publications for detailed explanations of the different metrics.

Collecting the property values leads to a feature vector that can be compared across datasets to categorize and compare different manifestations of CS. Table 2 contrasts a subset of the global and local properties for D1 and M1 to M6. We were able to access D1 as speech transcript and speech recording, while the movie datasets are based on written transcript, but could in theory be accessed in their original audio-visual format. Conversations in D1 are generally based on two speakers, but also include a single monolog, while the movie datasets contain multi-party dialog with up to 13 speakers. There are no explicit hierarchy levels in D1, and the speakers do not know each other, while social properties are varied in the movie dialogs. The local properties in the lower part of the table are split up by the language of the CS segments, for Compound properties, the language column corresponds to the language of the second CS segment s_2, contrasting the two switching directions. The information in Table 2 goes beyond the detailed analysis of language distribution in Table 1, that shows similar Span Entropy, but different CMI, M-Metric, Burstiness, and Language Entropy for the two corpora D1 and M1. Relevant for studying CS functions is the more even distribution of languages in M1 (indicating more instances of switching and less borrowing) and its larger variety of social properties (e.g., speakers with different relationship and hierarchy levels). To contrast CS for different social aspects, statistics on interactions between specific pairs of speakers can be analyzed as sub-corpora of M1, as proposed in (Pratapa and Choudhury, 2017). The local properties also show a larger proportion of positive and negative sentiment, sentiment change, and discourse relations in M1 compared to D1, which indicates that it is a more promising source for studying CS functions related to these properties. One exception are Hindi discourse relations that seem to be more prevalent in D1 compared to the movie datasets.

The other movie datasets, M2 to M6 are fairly similar to M1. The proportion of word-level switches is lower in all movie datasets compared to D1. In contrast to previous work on Twitter datasets (Rudra et al., 2016), there is no strong prevalence for expressing negative sentiment in Hindi or to switch to Hindi for expressing negative sentiment, but a slight tendency to express positive sentiment in English for M2 and M5. CS at the sentence level and cross-speaker switches are more prevalent when the second CS segment s_2

is Hindi. Tag-switching on the other hand is more prevalent for switches to English in all datasets, which supports the notion that Hindi is the main language in the movies. The use of English discourse relations at the beginning of s_2 is more prevalent in the movie datasets compared to the speech dataset D1.

We observe that many property combinations associated with CS functions are fairly rare in our datasets. An example is translation for emphasis. This function has been widely reported for English/Hindi code-switching, but there are only 13 instances even in the largest dataset D1. Finding evidence of CS functions in corpus data is difficult, which highlights the need for large corpus-based analyses.

Source	Metric	D1	M1
Guzmán et al. (2017)	M-Metric	0.275	0.772
	I-Metric	0.153	0.222
	Burstiness	0.293	0.141
	Memory	-0.174	-0.076
	Language Entropy	0.538	0.05
	Span Entropy	3.527	3.245
Gambäck and Das (2016)	CMI Index (C_c)	45.51	71.87
	% EN	12.23	27.44
	% HI	67.20	58.14

Table 1: CS metrics for detailed Granularity properties.

5. Application Scenario 2: Function Derivation

For the second application, deriving functions from properties, we focus on the movie dataset M1 that is based on the movie *Pink*, a court-room drama centering around a group of three girlfriends. One of the friends is accused of attempted murder, but claims she injured the victim, a young man, in self-defense. The interactions among the three friends and dialogs with the judge and attorney in the court room show different social settings, thus providing a good background for our analysis of the interaction of local and social functions of CS.

We first show examples for deriving local functions from properties, then discuss the interaction of local and global properties and their effects on deriving CS functions.

Deriving local functions from local properties The following example from M1 shows the *negative reinforcement* function:

> [This is b***s**t]s_1 [Bakwaas hai poori ki poori]s_2 - jhooth bol rahein hain yeh ladke aur aap bhi
>
> **Gloss**: [This is nonsense]s_1 [This is all rubbish.]s_2 - These boys are lying and so are you.

The English segment s_1 and the Hindi segment s_2 both have negative sentiment (sentiment(s_2)=sentiment(s_1) $= -1$), which together with the knowledge of a social property, i.e. the tendency of the English/Hindi-speaking community to use switching to emphasize negative emotions (Rudra et al., 2016) indicates the function of *negative reinforcement*. The same applies to the following example from M1:

> [Aur Minal ko tumhari ek nahi dono aankhein phhodni chahiye thi]s_1 [and you think you can scare us.]s_2
>
> **Gloss**: [And Minal should have taken out not one but both your eyes ...]s_1 [and you think you can scare us.]s_2

The following example from M1 shows the *narrative-evaluative* function:

> [Chaar din ho gaye - kuchh hua nahin na ...]s_1 [So why are we tense ?]s_2
>
> **Gloss**: [It's been four days, nothing has happened ...]s_1 [So why are we tense ?]s_2

In this example, a neutral statement is followed by an evaluation that shows negative sentiment. In the following example from M1, the *narrative-evaluative* function appears in inverse order: the evaluative s_1, thats shows negative sentiment, is followed by the statement s_2.

> [Look roz subah-subah tense hone se kya faayda !]s_1 [We'll go mad ...]s_2
>
> **Gloss**: [Look, what is the use of getting tense every day from the morning ...]s_1 [We'll go mad ...]s_2

The interaction of local and global properties The following example from *Kapoor and Sons* (M2) shows a social function of CS, namely the assertion of Neetu Chachi's identity through the use of English. Neetu Chachi has come to visit from New York. The mother, who has a higher status in age and relationship, continues using Hindi, even though she does know English. Here the switch to English by Neetu Chachi (in s_2 and s_4), who also knows Hindi perfectly well, is used to denote her identity:

> **Neetu Chachi:** [And look what I found in ...]s_1
>
> **Mother:** [Arre ... haan ...!Ye toh pata nahin kitni purani hain. Yeh *taste* kar.]s_1
>
> **Gloss**: [Oh yes! God knows how old this is! Taste this.]s_1
>
> **Neetu Chachi:** [It's delicious!]s_2 [Timmy ko bolo na mujhe kuch mutton recipes bheje]s_3 [Sharic just loves this stuff!]s_4
>
> **Gloss**: [It's delicious!]s_2 [Tell Timmy to send me some mutton recipes ...]s_3 [Sharic just loves this stuff!]s_4
>
> **Mother:** [Haan usse bolti hoon *email* karne ke liye.]s_5
>
> **Gloss**: [Ok, I will tell her to email them to you.]s_5

The examples shown in the previous paragraphs are based on informal conversation among friends and acquaintances. The following example from the movie *Pink* (M1) shows an interaction in a formal court-room setting. In this exchange, CS is aligned with situational changes based on social factors. The following statement is made by Deepak, the attorney defending Minal in the trial:

> [Minal Honourable judge sahab aapko baithne ke liye keh rahein hain]s_1 [and please be quiet]s_2
>
> **Gloss**: [Minal, the honourable judge is asking you to sit down ...]s_1 [and please be quiet]s_2

Dimension	Property	D1 Speech		M1 Pink		M2 Kapoor …		M3 Neerja		M4 Talvar		M5 Ek main …		M6 D-day	
Global Properties (of corpora)															
Granularity	languages	EN, HI		EN, HI		EN, HI		EN, HI		EN, HI		EN, HI		EN, HI	
	# tokens	38,624		10,541		8,227		3,438		4,404		5,386		5,420	
	# switch points	6,561		2,790		2,421		1,204		1,534		2,052		1,364	
	more Granularity: see Table 1														
Modality	written, formal	-		-		-		-		-		-		-	
	written, informal	+		+		+		+		+		+		+	
	speech	+		(+)		(+)		(+)		(+)		(+)		(+)	
	audio-visual	-		(+)		(+)		(+)		(+)		(+)		(+)	
Discourse	monolog	+		-		-		-		-		-		-	
	dialog	+		+		+		+		+		+		+	
	multi-party	-		+		+		+		+		+		+	
	max # speakers	2		13		11		9		7		15		7	
Social	hierarchy levels	-		-, +		-, +		-, +		-, +		-, +		-, +	
	familiarity	-		-, +		-, +		-, +		-, +		-, +		-, +	
	repetition/emphasis	+		+		+		+		+		+		+	
Local Properties (of CS segments and CS pairs)															
		EN	HI	EN	HI	EN	HI	EN	HI	EN	HI	EN	HI	EN	HI
Syntactic	sentence	0.014	0.078	0.105	0.073	0.064	0.096	0.080	0.096	0.056	0.063	0.076	0.067	0.007	0.069
	phrase	0.098	0.579	0.197	0.568	0.025	0.578	0.252	0.578	0.266	0.553	0.315	0.543	0.251	0.608
	word	0.853	0.229	0.664	0.315	0.666	0.248	0.666	0.252	0.668	0.357	0.581	0.331	0.702	0.294
	tag	0.025	0.010	0.066	0.019	0.197	0.047	0.140	0.030	0.087	0.050	0.144	0.048	0.034	0.026
	named-entity	0.107	0.010	0.043	0.071	0.043	0.057	0.041	0.096	0.050	0.087	0.054	0.045	0.083	0.069
	continuation=0	0.096	0.254	0.147	0.163	0.164	0.219	0.184	0.248	0.171	0.238	0.191	0.214	0.124	0.231
	continuation=1	0.386	0.240	0.349	0.335	0.334	0.280	0.319	0.266	0.325	0.263	0.309	0.286	0.380	0.283
Sentiment	sentiment=-1	0.167	0.171	0.371	0.378	0.401	0.401	0.120	0.144	0.157	0.158	0.352	0.359	0.158	0.165
	sentiment=0	0.690	0.699	0.571	0.571	0.851	0.463	0.871	0.875	0.821	0.829	0.546	0.577	0.829	0.843
	sentiment=+1	0.117	0.121	0.048	0.043	0.162	0.138	0.023	0.014	0.016	0.015	0.103	0.067	0.022	0.018
Pragmatic	has-DM	0.037	0.438	0.061	0.253	0.030	0.206	0.022	0.144	0.018	0.196	0.047	0.212	0.014	0.213
Compound	speaker-change=1	0.027	0.041	0.046	0.076	0.059	0.102	0.061	0.098	0.059	0.082	0.056	0.087	0.032	0.067
	speaker-change=0	0.460	0.455	0.454	0.425	0.441	0.399	0.446	0.419	0.441	0.421	0.445	0.414	0.473	0.446
	topic-change	0.024	0.019	0.022	0.027	0.027	0.025	0.015	0.017	0.016	0.020	0.024	0.027	0.010	0.021
	translation	0.001	0.001	0.001	0.001	0.000	0.001	0.000	0.003	0.000	0.002	0.000	0.001	0.001	0.001
	senti-pair $\langle -1, 1\rangle$	0.000	0.001	0.001	0.004	0.003	0.010	0.000	0.000	0.000	0.000	0.001	0.005	0.000	0.002
	senti-pair $\langle 1, -1\rangle$	0.001	0.002	0.001	0.003	0.005	0.005	0.000	0.000	0.000	0.001	0.004	0.002	0.001	0.001
	senti-pair $\langle 1, 0\rangle$	0.002	0.005	0.003	0.004	0.013	0.009	0.007	0.001	0.003	0.001	0.017	0.004	0.001	0.001
	senti-pair $\langle 0, 1\rangle$	0.003	0.005	0.001	0.006	0.004	0.019	0.002	0.005	0.001	0.002	0.003	0.020	0.000	0.002
	senti-pair $\langle -1, 0\rangle$	0.004	0.006	0.013	0.032	0.014	0.028	0.006	0.018	0.011	0.018	0.018	0.023	0.006	0.018
	senti-pair $\langle 0, -1\rangle$	0.004	0.005	0.015	0.030	0.014	0.033	0.018	0.012	0.007	0.014	0.017	0.022	0.007	0.014
	discourse relation†	0.010	0.040	0.032	0.014	0.025	0.009	0.012	0.003	0.021	0.003	0.022	0.007	0.015	0.007

Table 2: Footprinting CS corpora; DM = Discourse marker; + and - stand for boolean true and false; † marks property labels created automatically. Statistics in the lower half of the table are relative to the number of switch points in each dataset.

English is the marked language in the court-room setting, and switching to English emphasizes and reinforces the strength of the request. The emphasis is further enhanced by a local function, namely starting the English segment with discourse marker *and*. Deepak in his role as the attorney uses code-switching to English to emphasize the seriousness of his request, which is licensed by the hierarchy induced by the formal court-room setting. The characters Deepak and Minal are not familiar figures, and they are not separated by a hierarchy outside of the courtroom setting, so it would be rude for Deepak to switch to English to reinforce his request in this way in an informal setting.

A third interesting example of the interaction of social and global functions of code-switching occurs in the movie *Neerja* (M3). In this movie, a group of terrorists kidnaps a plane. They are opposed by a group of police officials. Depending on who has the upper hand in the negotiations, each of the groups changes between formal, polite language (i.e. talking up when being lower in the hierarchy), or rude, impolite language (i.e. talking down when being higher in the hierarchy). One main characteristic of the impolite language is the stronger use of code-switching. These changes between up-talk and down-talk show an interaction between social hierarchy and use of code-switching in discourse.

Summary The above examples show an application of our representation to the identification and derivation of CS functions. To further validate our strategy for the identification of CS functions based on properties, we will contrast the statistics on functions automatically derived from properties to manually annotated functions for our datasets. This will help to identify property configurations that are *sufficient* for the (automatic) identification of CS functions in future work.

6. Conclusion

We present an integrated representation of code-switching functions to facilitate their systematic empirical study, particularly the interaction between local and social aspects of the CS functions. The proposed representation is language-independent and can be extended by additional properties (for instance gestures for multi-modal corpora) and emerging functions.

Comparative and systematic corpus-based study of CS is desired (Gullberg et al., 2009; Myslín and Levy, 2015) and facilitated by an increasing number of available corpora (Diab et al., 2014; Çetinoğlu, 2017; Rudra et al., 2016); With our framework we aim to contribute to the comparative corpus-based study of code-switching and to foster the discussion of the interaction between local and social functions. Besides extending and improving the proposed representation in our future work, we plan further applications of the framework to exemplify in which ways it can be used for CS research. Moreover, we will study the automatic derivation of functions in more detail, with a particular focus on discourse relations.

Acknowledgments

This paper is built upon two annotation studies. We gratefully acknowledge the support by Prof. Niloy Ganguly, Koustav Rudra, and Bidisha Samanta during the course of the two studies. Adithya Pratapa kindly provided the language tags and CMI metrics. Last, but not least, we acknowledge the contributions of our annotators Sanjana Arora, Shivangi Shukla, and Jyothsna Chakravarthy.

7. Bibliographical References

Abdul-Zahra, S. (2010). Code-switching in language: An applied study. *Journal Of College Of Education For Women*, 21(1):283–296.

Auer, P. (2013). *Code-switching in conversation: Language, interaction and identity*. Routledge.

Begum, R., Bali, K., Choudhury, M., Rudra, K., and Ganguly, N. (2016). Functions of Code-Switching in Tweets: An Annotation Framework and Some Initial Experiments. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1644–1650, Portorož, Slovenia.

Çetinoğlu, Ö. (2017). A code-switching corpus of turkish-german conversations. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 34–40, Valencia, Spain.

Crystal, D. (1987). *The Cambridge Encyclopedia of Language*. Cambridge University Press, Cambridge, U.K.

Diab, M., Hirschberg, J., Fung, P., and Solorio, T. (2014). *Proceedings of the First Workshop on Computational Approaches to Code Switching*. Doha, Qatar.

Gambäck, B. and Das, A. (2016). Comparing the Level of Code-Switching in Corpora. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1850–1855.

Gullberg, M., Indefrey, P., and Muysken, P., (2009). *Research techniques for the study of code-switching*, pages 21–39. Cambridge University Press, Cambridge.

Gumperz, J. (1982). *Discourse Strategies*. Cambridge University Press, Cambridge.

Guzmán, G., Ricard, J., Serigos, J., Bullock, B. E., and Toribio, A. J. (2017). Metrics for modeling code-switching across corpora. In *Proceedings of the Interspeech 2017 Special Session on Code Switching*, Stockholm, Sweden.

Myslín, M. and Levy, R. (2015). Code-switching and predictability of meaning in discourse. *Language*, 91(4):871–905.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). SemEval-2013 Task 2: Sentiment Analysis in Twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320, Atlanta, GA, USA.

Poplack, S. (1988). Contrasting patterns of code-switching in two communities. *Codeswitching: Anthropological and sociolinguistic perspectives*, 48:215–44.

Poplack, S. (2015). Code switching: Linguistic. In *International Encyclopedia of the Social and Behavioral Sciences, 2nd edition*, pages 918–925. Elsevier Ltd.

Prasad, R., Webber, B., and Joshi, A. (2014). Reflections on the Penn Discourse TreeBank, Comparable Corpora, and Complementary Annotation. *Computational Linguistics*, 40(4):921–950.

Pratapa, A. and Choudhury, M. (2017). Quantitative characterization of code switching patterns in complex multiparty conversations: A case study on hindi movie scripts. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 75–84, Kolkata, India, December. NLP Association of India.

Reyes, I. (2004). Functions of code switching in school children's conversations. *Bilingual Research Journal*, 28:83–96.

Rijhwani, S., Sequiera, R., Choudhury, M., Bali, K., and Maddila, C. S. (2017). An Exploratory Study of Structural and Functional Aspects of English-Hindi Code Switching in Twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1971–1982, Vancouver, Canada.

Riloff, E., Qadir, A., Surve, P., De Silva, A., Gilbert, N., and Huang, R. (2013). Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, WA, USA.

Rudra, K., Rijhwani, S., Begum, R., Bali, K., Choudhury, M., and Ganguly, N. (2016). Understanding Language Preference for Expression of Opinion and Sentiment: What do Hindi-English Speakers do on Twitter? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1131–1141, Austin, TX, USA.

Searle, J. R. (1969). *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge University Press, Cambridge, U.K.

A Corpus of eRulemaking User Comments for Measuring Evaluability of Arguments

Joonsuk Park and Claire Cardie
Department of Computer Science, Williams College, Massachusetts, USA
Department of Computer Science, Cornell University, New York, USA
jpark@cs.williams.edu, cardie@cs.cornell.edu

Abstract

eRulemaking is a means for government agencies to directly reach citizens to solicit their opinions and experiences regarding newly proposed rules. The effort, however, is partly hampered by citizens' comments that lack reasoning and evidence, which are largely ignored since government agencies are unable to evaluate the validity and strength. We present *Cornell eRulemaking Corpus – CDCP*, an argument mining corpus annotated with argumentative structure information capturing the evaluability of arguments. The corpus consists of 731 user comments on Consumer Debt Collection Practices (CDCP) rule by the Consumer Financial Protection Bureau (CFPB); the resulting dataset contains 4931 elementary unit and 1221 support relation annotations. It is a resource for building argument mining systems that can not only extract arguments from unstructured text, but also identify what additional information is necessary for readers to understand and evaluate a given argument. Immediate applications include providing real-time feedback to commenters, specifying which types of support for which propositions can be added to construct better-formed arguments.

Keywords: argument mining, e-government, e-rulemaking, text analytics

1 Introduction

The U.S. federal agencies amend rules in a highly transparent manner, inviting public participation as they are finalized. This is legally ensured in part by the requirement that agencies publish descriptions and rationale behind newly proposed rules and solicit feedback from the public (Park et al., 2012; Farina and Newhart, 2013). However, the public participation tends to be dominated by large corporations and interest groups (CSFFR, 2009); *eRulemaking* is an ongoing effort to promote citizens' participation in federal policymaking by using the latest information technology to directly reach citizens and incorporate their feedback (Lubbers et al., 2012).

Government agencies consider reasoning and validity of supporting evidence, rather than a mere number of citizens supporting an argument, to determine how the rules should be adjusted to meet the needs of those who are directly affected. Thus, useful feedback consists of clear reasoning and objective evidence supporting factual claims (Park et al., 2015). However, many comments are not written this way, thwarting the government agencies' effort to communicate with citizens.

Consider the following comments from `www.regulationroom.org`, an eRulemaking website:

> (1) \$400 is enough compensation,$_A$ as it can cover a one-way fare across the US.$_B$ I checked in a passenger on a \$98.00 fare from east coast to Las Vegas the other day.$_C$
>
> (2) All airfare costs should include the passenger's right to check at least one standard piece of baggage.$_A$ All fees should be fully disclosed at the time of airfare purchase, regardless of nature (i.e. optional or mandatory).$_B$ Any changes in fees should be identified by air carriers at least 6 months prior to taking effect.$_C$

Comment 1 consists of propositions in support relations that collectively form a single argument: Proposition 1.C is an anecdotal evidence supporting Proposition 1.B, which in turn is a reason explaining why Proposition 1.A is true. Readers are able to make sense of the argument and evaluate its validity and strength, because each proposition is accompanied with a support of an appropriate type. (Figure 1 shows a sample annotation capturing the above discussion; see Section 3 for more details on the types of support and when they are appropriate.) In contrast, the propositions in Comment 2 are in no support relation with one another. In fact, each proposition functions as the conclusion of its own argument, where each argument contains no support for its conclusion. This renders it difficult for readers to understand the arguments, let alone evaluate them. Thus, Comment 1 is much more desirable for readers, whether it be government agencies or fellow citizens.

The aforementioned difference between the arguments

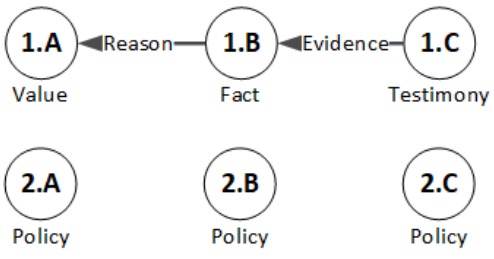


Figure 1: Annotated Example Comments

made in Comments 1 and 2 is captured by the notion of *evaluability* of argument proposed by Park et al. (2015)—are the propositions comprising a given argument adequately supported so as for readers to understand and evaluate the validity or strength of the whole argument?

We present *Cornell eRulemaking Corpus – CDCP*, an argument mining corpus annotated with argumentative structure information capturing the evaluability of arguments. We annotated 731 user comments on Consumer Debt Collection Practices (CDCP) rule by the Consumer Financial Protection Bureau (CFPB) posted on `www.regulationroom.org`; the resulting dataset contains 4931 elementary unit and 1221 support relation annotations. It will be a valuable resource for building argument mining systems that can not only extract arguments from unstructured text, but also identify ways in which a given argument can be improved with respect to its evaluability. Immediate applications include automatically ranking arguments based on their evaluability for a (crude) identification of read-worthy comments and providing real-time feedback to writers, specifying which types of support for which propositions can be added to construct better-formed arguments.

The remainder of this paper is organized as follows: We discuss related work (Section 2), provide an overview of the annotation scheme (Section 3), present an annotation study (Section 4) and describe the resulting dataset (Section 5).

2 Related Work

This paper presents a corpus for the purpose of mining and evaluating arguments in eRulemaking user comments. It is closely related to two areas of research: argument mining and argument quality assessment.

2.1 Argument Mining

Argument mining is a developing field of computational linguistics that aims at identifying argumentative structures in unstructured text. Extracting claims together with their respective premises allows us to go beyond opinion mining by considering the reasoning and rationale behind people's opinions. (Peldszus and Stede, 2013; Lippi and Torroni, 2016) Argument mining systems build on theoretical models of argument, which define argumentative components and their relations in a variety of ways. Famous models include the Toulmin Model (Toulmin, 1958) and argument schemes (Walton et al., 2008). The Toulmin Model is a general model of practical argumentation that can be instantiated in many forms. The three major components of the model are *claim*, *warrant*, and *data*, where *warrant* explains how *data* supports the *claim*. One criticism, which in turn make it challenging to build an argument mining system base on this model, is that the model leaves room for multiple interpretations. For example, according to Eemeren et al. (1987), *warrant* is indistinguishable from *data*. On the other hand, argument schemes capture specific patterns of argument that are in use; each argument scheme specifies specific premises for the given conclusion, as well as critical questions that can be used to examine the strength of the given argument (Walton, 1996; Blair, 2001). Having many specific premises, a subset of which may not be present in the text, makes it difficult for manual annotation and automatic classification. The sheer number of argument schemes also causes additional challenges in gathering enough examples for each scheme. In this work, we adopt a model uniquely designed to capture the evaluability of arguments, which is general enough to model diverse argumentative structures that appear in practical argumentation (Park et al., 2015).

Argument mining systems also differ in the domain, resulting in datasets consisting of newspaper articles (Reed et al., 2008), legal documents (Mochales and Moens, 2011), student essays (Stab and Gurevych, 2014), and eRulemaking user comments (Park and Cardie, 2014; Konat et al., 2016), to name a few. While ours is not the first eRulemaking dataset, the task is different; Park and Cardie (2014) targets elementary unit classification only, and Konat et al. (2016) focuses on identifying divisive issues between commenters by analyzing conflict relations found across multiple comments in a thread. In contrast, we examine support structures within a comment; our dataset contains both elementary unit and support relation annotation without cross-comment conflict annotation. Also, the user comments comprising our dataset are different from those in the aforementioned datasets.

2.2 Argument Quality Assessment

Measuring the quality of argument has long been a subject of discussion and research, leading to a variety of dimensions of quality (Toulmin, 1958; Perelman et al., 1969; van Eemeren and Grootendorst, 2004; Johnson and Blair, 2006; Wachsmuth et al., 2017). More recently, argument mining research is conducted with specific measures of quality depending on the domain and purpose, such as persuasiveness (Tan et al., 2016), strength (Persing and Ng, 2015), acceptability (Cabrio and Villata, 2012), and convincingness (Habernal and Gurevych, 2016). The measure of quality we are interested in is evaluability (Park et al., 2015). By examining arguments' evaluability, we aim to identify ways to improve them so that they can be better understood and evaluated. For example, we answer questions like, "Which propositions need additional reasons or evidence supporting them?" This is the type of constructive feedback that can help commenter improve their arguments, unlike quality measures that results in a single numeric score without specifying how an argument can be improved.

3 Annotation Scheme

The annotators annotated the elementary units and support relations defined in the argumentation model proposed by Park et al. (2015). In this section, we provide a brief overview of the model; please refer to the original paper for more details.

The goal of the model is to capture whether an argument consists of explicitly stated premises that allow readers to understand and evaluate the given argument. The model defines five types of elementary units that are prevalent in online comments, along with two types of support relations between the units.

3.1 Elementary Units

Proposition of Non-Experiential Fact (FACT) : This refers to an objective proposition "expressing or dealing with facts or conditions as perceived without distortion by personal feelings, prejudices, or interpretations."[1] By definition a FACT proposition has a truth value that can be verified with objective evidence. We restrict the notion of verifiability to pieces of evidence that may be available at the time the claim is made; predictions about future are considered unverifiable. Here are examples from the dataset:

- Recently, courts have held that debt collectors can escape 1692i's venue provisions entirely by pursuing debt collection through arbitration instead.
- banks can simply add this provision to their Loan Sale Agreements.
- That process usually takes as much a 2 years or more.

Proposition of Experiential Fact (TESTIMONY) : This refers to an objective proposition about the author's personal state or experience. One major characteristic of this type of objective propositions, as opposed to the non-experiential counterparts classified as FACT, is that it is often practically impossible to provide objective evidence in online commenting setting, in the form of URL or citation. That is, evidence for TESTIMONY is not publicly available in most cases. For example:

- Informing them that we wanted all debt collection to be written was also ignored.
- A neighbor who has since moved away has had her debts turned over to collection agencies.
- We receive repeated calls trying to get contact information, even though we request to be taken off their list.

Proposition of Value (VALUE) : This refers to a proposition containing value judgments without making specific claims about what should be done (If so, then it is a POLICY proposition.). Because of the subjectivity of value judgments, a VALUE proposition cannot be proved directly with objective evidence; however, providing a reason as support is feasible and appropriate. For example:

- That would be a starting point that can be expanded on as the system is fine tuned.
- Admittedly, their system is much more complex and dives much deeper than would be required for the debt industry.
- However, the double penalty against the consumer is certainly unfair.

Proposition of Policy (POLICY) : This refers to a proposition proposing a specific course of action to be taken. It typically contains modal verbs like "should" and "ought to." Just like VALUE, a POLICY proposition cannot be directly proved with objective evidence, and a proper type of support is a logical reason from which the proposition can be inferred. For example:

- They should not be allowed to contact anyone (other than the debtor him/herself) more than once.
- I say there ought to be sanctions, monetary sanctions, against these credit reporting agencies for making these mistakes and their cavalier attitude.
- Set up a system where the consumer is on equal footing with the debt collectors.

Reference to a Resource (REFERENCE) : This refers to a reference to a source of objective evidence. In online comments, a REFERENCE is typically a citation or a URL of a published work from a renowned source.For example:

- http://files.consumerfinance.gov/f/201309_cfpb_agency-brief_12-cv-04057.pdf
- http://www.myfico.com/CreditEducation/ImproveYourScore.aspx
- www.optoutprescreen.com

3.2 Support Relations

Reason : An elementary unit X is a *reason* for a proposition Y (of type POLICY, VALUE, FACT, or TESTIMONY) if X provides rationale for Y. For example:

- **Y:** I urge the CFPB to include in a rule language interpreting 1692i as requiring debt collectors to proceed in court, not through largely-unregulated arbitral forums. **X:** As the NAF studies reflect, arbitration has not proven a satisfactory alternative.

Evidence : An elementary unit X is *evidence* for a proposition Y (of type POLICY, VALUE, FACT, or TESTIMONY) if it proves whether proposition Y is true or not. The possible types of evidence are limited to TESTIMONY or REFERENCE based on previous studies on what constitutes justified grounds (Toulmin and Janik, 1979; Hitchcock, 2005). For example:

- **Y:** At least in Illinois there is a Caller ID spoofing law. **X:** http://www.ilga.gov/legislation/ilcs/ilcs3.asp?ActID=1355ChapterID=24

[1] http://www.merriam-webster.com/

3.3 Evaluability

An argument is *evaluable* if all propositions comprising the given argument is supported by an explicit premise of an appropriate type, as summarized in Table 1. The underlying assumption is that readers are able to understand the gist of an argument—and at least roughly evaluate its strength—as long as one premise of an appropriate type is explicitly stated for each proposition.[2]

Once elementary units and support relations comprising an argument are identified, the evaluability of the given argument can be determined. This is done by comparing the appropriate types of support and the types of support present in the argument, if any. In the process, additional support that is necessary to make the given argument evaluable (e.g."A reason for proposition X needs to be provided.") can also be identified.

Proposition Type	POLICY	VALUE	FACT	TESTIMONY
Reason	✓	✓	✓	✓*
Evidence			✓	✓*

Table 1: Appropriate Support Types for Propositions
* Support can be provided, but it is not required.

4 Annotation Study

We annotated user comments on the Consumer Debt Collection Practices (CDCP) rule. The discussion regarding CDCP rule was hosted on `www.regulationroom.org` with a partnership with the CFPB. The goal was for the CFPB to hear about the first-hand experiences and concerns regarding debt collection practices. According to a voluntary user survey that asked the commenters to self-identify themselves, about 64% of the comments came from consumers, 22% from debt collectors, and the remainder from others, such as consumer advocates and counsellor organizations (Farina et al., 2017).

Each user comment was annotated by two annotators, who independently determined the types of elementary units and support relations among them using the GATE annotation tool (Cunningham et al., 2011). A third annotator manually resolved the conflicts to produce the final dataset.

An elementary unit is either a sentence or a clause; a sentence is split into smaller units if there are multiple independent clauses or an independent clause with a subordinate clause of interest, such as a because-clause. Non-argumentative portions of comments, such as greetings and names, were removed as elementary unit boundaries are determined in this way.

Inter-annotator agreement between 2 annotators is measured with Krippendorf's α (Krippendorff, 1980) with respect to elementary unit type (α=64.8%) and support relations (α=44.1%); IDs of supported elementary units are treated as labels for the supporting elementary units.[3]

The disagreements in elementary unit type annotation mostly occurred between VALUE vs TESTIMONY and VALUE vs FACT. The former is the case when a testimony spans multiple propositions and a few of them are subjective opinions about the experience. The latter often happens with an elementary unit that contains both subjective and objective expressions, e.g. "Unfortunate, but yes they are allowed to deny due process and get away with it." In this case, annotators had to determine the commenter's main intention—is it to express the emotion or state the fact? Depending on the answer, the given elementary unit was either marked as VALUE or FACT. (Allowing a more granular boundaries for elementary units can solve this type of disagreement; however, an undesirable effect of this is that automatic segmentation becomes more challenging.)

5 Dataset

The resulting dataset, *Cornell eRulemaking Corpus – CDCP*, consists of 731 comments, 4931 elementary units, and 1221 support relations as summarized in Table 2. About 45% of the elementary units are VALUE type, and most support relations are reasons. Table 3 describes annotated information in this dataset.

Figure 2 shows the types of supported elementary units and those of supporting elementary units. The percentage of supported elementary units decreases as the elementary unit's objectivity goes from the least objective (POLICY) to the most objective (REFERENCE). One reason is that it is easier to provide a reason as to why one thinks or feels something (POLICY and VALUE) than to justify factual propositions (FACT and TESTIMONY). Interestingly, even though both POLICY and VALUE are subjective, there is a notable difference in the support pattern; 51% of POLICY propositions are supported, whereas only 28% of VALUE propositions are supported. This means that when commenters propose a specific course of actions to be taken, they are more likely to provide support for it. This is because POLICY propositions are often the central claims of the comments, thus other propositions naturally support them. Also, unlike VALUE, a simple expression of one's thoughts and feelings, POLICY, a proposal to act in a certain way, is associated with persuasion, which benefits from explicitly stated reasoning.

A significant portion, roughly 75%, of support relation annotations are between adjacent elementary units. While commenters certainly tend to provide reasons immediately after the proposition to be supported, it is also easier for annotators to identify support relations in proximity. Thus, support relations in the wild may be not as skewed toward

[2] Please refer to Park et al. (2015) for formal definitions.

[3] Krippendorf's α is suitable for our purpose as it is compatible with various types of labeling, along with the ability to handle missing annotations.

POLICY	VALUE	FACT	TESTIMONY	REFERENCE	Elementary Units	Reason	Evidence	Support Relations
815	2182	785	1117	32	4931	1174	46	1220

Table 2: Number of Elementary Units and Support Relations in the Dataset (731 comments)

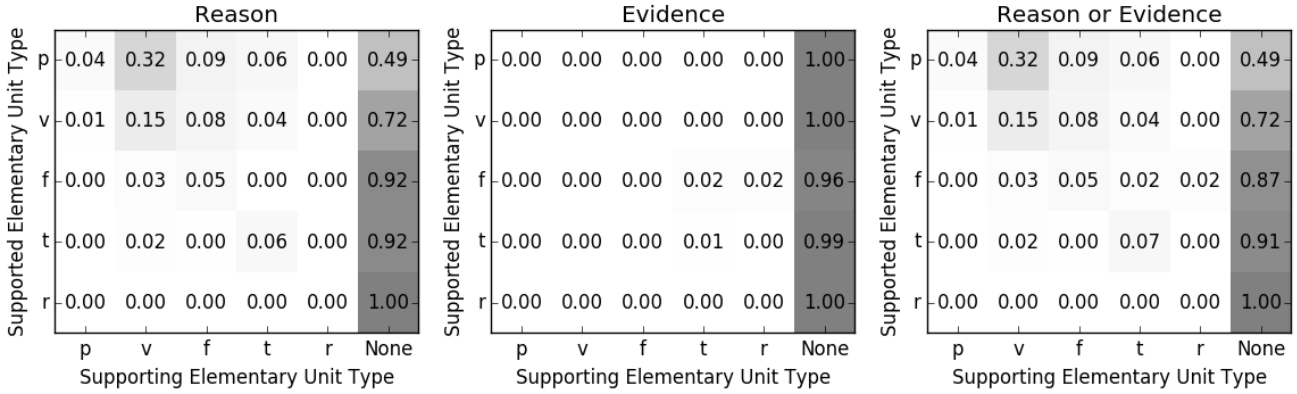


Figure 2: Types of Elementary Units in Support Relations (%)

Field	Description
ID	ID of the elementary unit
Text	Text of the elementary unit
Type	POLICY, VALUE, FACT, TESTIMONY or REFERENCE
Reasons	List of elementary unit IDs serving as reasons
Evidence	List of elementary unit IDs serving as evidence

Table 3: Annotated Information: Each comment annotation consists of a list of elementary units in the given comment with fields described in this table.

those between adjacent elementary units.

6 Conclusion

We have presented *Cornell eRulemaking Corpus – CDCP*, an argument mining corpus annotated with argumentative structure information capturing the evaluability of arguments. The corpus consists of 731 user comments on Consumer Debt Collection Practices (CDCP) rule by by the Consumer Financial Protection Bureau (CFPB) posted on `www.regulationroom.org`; the resulting dataset consists of 4931 elementary unit and 1221 support relation annotations. It will be a valuable resource for building argument mining systems that can not only extract arguments from unstructured text, but also identify which additional information is necessary for readers to understand and evaluate a given argument.

Future work includes: (1) construction of a larger corpus using the same or similar annotation scheme and (2) making use of the resources to train argument mining systems (Niculae et al., 2017) and subsequent applications, such as a commenting interface that provides real-time feedback to help commenters construct evaluable arguments. Domain adaptation is also desirable, since building an argument mining dataset for individual domains incurs a significant cost.

7 Bibliographical References

Blair, J. A. (2001). Walton's argumentation schemes for presumptive reasoning: A critique and development. *Argumentation*, 15(4):365–379.

Cabrio, E. and Villata, S. (2012). Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 208–212, Jeju Island, Korea, July. Association for Computational Linguistics.

CSFFR. (2009). Achieving the potential:the future of federal e-rulemaking. Technical report, Committee on the Status & Future of Federal e-Rulemaking/American Bar Association, Washington, DC.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Farina, C. R. and Newhart, M. J. (2013). Rulemaking 2.0: Understanding and getting better public participation.

Farina, C. R., Blake, C. L., Newhart, M. J., and Nam, C. (2017). Digital support for enhanced democratic participation in us rulemaking. In C. Prins, et al., editors, *Digital Democracy in a Globalized World*, chapter 10. Edward Elgar Publishing.

Habernal, I. and Gurevych, I. (2016). Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume Volume 1: Long Papers, page (to appear). Association for Computational Linguistics, August.

Hitchcock, D. (2005). Good reasoning on the toulmin model. *Argumentation*, 19(3):373–391.

Johnson, R. and Blair, J. (2006). *Logical Self-defense*. Key titles in rhetoric, argumentation, and debate series. International Debate Education Association.

Konat, B., Lawrence, J., Park, J., Budzynska, K., and Reed, C. (2016). A corpus of argument networks: Using graph properties to analyse divisive issues. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Krippendorff, K. (1980). *Content Analysis: An Introduction to Its Methodology*. Sage commtext series. Sage Publications.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technol.*, 16(2):10:1–10:25, March.

Lubbers, J., of Administrative Law, A. B. A. S., Practice, R., Government, A. B. A., and Division, P. S. L. (2012). *A Guide to Federal Agency Rulemaking*. ABA Section of Administrative Law and Regulatory Practice and Government and Public Sector Lawyers Division.

Mochales, R. and Moens, M.-F. (2011). Argumentation mining. *Artif. Intell. Law*, 19(1):1–22, March.

Niculae, V., Park, J., and Cardie, C. (2017). Argument mining with structured svms and rnns. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 985–995. Association for Computational Linguistics.

Park, J. and Cardie, C. (2014). Identifying appropriate support for propositions in online user comments. In *Proceedings of the First Workshop on Argumentation Mining*, pages 29–38, Baltimore, Maryland, June. Association for Computational Linguistics.

Park, J., Klingel, S., Cardie, C., Newhart, M., Farina, C., and Vallbé, J.-J. (2012). Facilitative moderation for online participation in erulemaking. In *Proceedings of the 13th Annual International Conference on Digital Government Research*, pages 173–182. ACM.

Park, J., Blake, C., and Cardie, C. (2015). Toward machine-assisted participation in erulemaking: An argumentation model of evaluability. In *Proceedings of the 15th International Conference on Artificial Intelligence and Law*, ICAIL '15, pages 206–210, New York, NY, USA. ACM.

Peldszus, A. and Stede, M. (2013). From argument diagrams to argumentation mining in texts: A survey. *Int. J. Cogn. Inform. Nat. Intell.*, 7(1):1–31, January.

Perelman, C., Olbrechts-tyteca, L., Wilkinson, J., and Weaver, P. (1969). *The New Rhetoric*. University of Notre Dame P X.

Persing, I. and Ng, V. (2015). Modeling argument strength in student essays. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 543–552.

Reed, C., Palau, R. M., Rowe, G., and Moens, M.-F. (2008). Language resources for studying argument. In *LREC*. European Language Resources Association.

Stab, C. and Gurevych, I. (2014). Annotating argument components and relations in persuasive essays. In Junichi Tsujii et al., editors, *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1501–1510, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Tan, C., Niculae, V., Danescu-Niculescu-Mizil, C., and Lee, L. (2016). Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 613–624, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Toulmin, S.E., R. R. and Janik, A. (1979). *An Introduction to Reasoning*. Macmillan Publishing Company.

Toulmin, S. E. (1958). *The uses of argument*. Cambridge University Press.

van Eemeren, F. and Grootendorst, R. (2004). *A Systematic Theory of Argumentation: The Pragma-dialectical Approach*. A Systematic Theory of Argumentation: The Pragma-dialectical Approach. Cambridge University Press.

van Eemeren, F., Grootendorst, R., and Kruiger, T. (1987). *Handbook of argumentation theory ; a critical survey of classical backgrounds and modern studies*. PDA Series. Foris Publications.

Wachsmuth, H., Naderi, N., Hou, Y., Bilu, Y., Prabhakaran, V., Thijm, T. A., Hirst, G., and Stein, B. (2017). Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 176–187, Valencia, Spain, April. Association for Computational Linguistics.

Walton, D., Reed, C., and Macagno, F. (2008). *Argumentation Schemes*. Cambridge University Press.

Walton, D. (1996). *Argumentation schemes for presumptive reasoning*. Lawrence Erlbaum Associates.

A Multi-layer Annotated Corpus of Argumentative Text: From Argument Schemes to Discourse Relations

Elena Musi[§], Tariq Alhindi[§], Manfred Stede [†], Leonard Kriese [†], Smaranda Muresan [§], Andrea Rocci
Columbia University[§], Potsdam University[†], Universita della Svizzera italiana
475 Riverside Drive New York (USA) [§], Karl-Liebknecht-Str. 24-25 Potsdam (Germany) [†], Via Buffi 13 Lugano (Switzerland)
em3202/ta2509/sm761@columbia.edu [§], stede@uni-potsdam.de, leonard.kriese@hotmail.de, andrea.rocci@usi.ch

Abstract

We present a multi-layer annotated corpus of 112 argumentative microtexts encompassing not only argument structure and discourse relations (Stede et al., 2016), but also argument schemes — the inferential relations linking premises to claims. We propose a set of guidelines for the annotation of argument schemes both for support and attack relations, and a new user-friendly annotation tool. The multi-layer annotated corpus allows us to conduct an initial study of dependencies between discourse relations (according to Rhetorical Structure Theory (Mann and Thompson, 1988)) and argument schemes. Our main contribution is that of offering the first resource for the combined study of (argumentative) discourse relations and inferential moves.

Keywords: argumentation mining, argument schemes, discourse relations

1. Introduction

Recent interest in Argumentation Mining (e.g., (Lippi and Torroni, 2016; Habernal and Gurevych, 2017) has brought to the fore the need for corpora annotated with argument information, which can be used as training data. Generally, the automatic search for arguments encompasses the following steps (Peldszus and Stede, 2013):

1. the segmentation of texts into *argumentative discourse units*(ADUs);
2. the classification of the role (e.g., claim, premise) played by each ADU;
3. the analysis of the relations linking ADUs (e.g., support, attack); and
4. the identification argument schemes, namely the implicit and explicit inferential relations within and across ADUs.

Annotation efforts have so far provided corpora that focus on the first three steps, with genres ranging from persuasive essays and scientific articles to online debates (e.g., (Kirschner et al., 2015; Ghosh et al., 2014; Stab and Gurevych, 2014; Walker et al., 2012). Step 4, which builds a bridge to reasoning, has not received nearly as much attention as the others. A notable exception is the *Araucaria* corpus (Reed and Rowe, 2004), which provides annotations based on Walton et al. (2008a) argument schemes. This annotated corpus has led to work on automatically classifying argument schemes focusing on the five most frequent ones (Feng and Hirst, 2011). Other projects looked at restricted subsets of argument schemes (Green, 2017; Schneider et al., 2013) and have not yet led to publicly-available data.

In this paper, we report on an annotation project that adds information about inferential rules, in the shape of argument schemes, to an existing corpus that already holds annotations of argumentation structure as well as discourse structure based on Rhetorical Structure Theory (RST) and Segmented Discourse Representation Theory(SDRT) (Stede et al., 2016). Our emphasis thus is on a *multi-layer* resource that allows for correlating different levels and for studying dependencies between discourse relations and argument structure, or — enabled by our new result — discourse relations and underlying inferential moves.

Specifically, we propose guidelines for the annotation of argument schemes for both SUPPORT and ATTACK relations using the *Argumentum Model of Topics* framework (Rigotti and Morasso, 2010) (Section 2). These guidelines, based on semantic principles, are scalable to other text genres as well as languages. In addition, we present a new annotation tool for argument schemes with a user-friendly interface to support the annotation process (Section 3). The reliability of the guidelines is tested through a pilot annotation project on top of 40 microtexts, obtaining moderate agreement (Section 4). Finally, we report on initial experiments on the mapping of rhetorical discourse relations and types of argument schemes (Section 5). The annotation guidelines, the corpus and the annotation tool are available at $http://angcl.ling.uni-potsdam.de/resources/argmicro.html$.

2. Data and methods

2.1. Corpus

The 'argumentative microtext' corpus includes 112 short texts that have been produced by means of a text generation experiment: 23 subjects were asked to choose a controversial topic from a fixed list and write a short argumentative text that should contain a clear major claim and arguments supporting or attacking it (Peldszus and Stede, 2016). The texts were written in German by native speakers and later professionally translated into English. The argument structure was annotated according to the scheme proposed by Peldszus and Stede (2013), which builds on the ideas of Freeman (2011). Briefly, the texts are segmented into argumentative units, each unit has an argumentative role and is related to another unit, except for the single main claim (resulting overall in a tree structure). Furthermore, the corpus was enriched with discourse structure information based on

RST and SDRT theories by Stede et al. (2016), and Becker et al. (2016) have provided the additional layer of situation entity types (Becker et al., 2016). Annotating the argument schemes, covering the underlying inferential moves, will provide a valuable annotation layer for studying the mechanics of argumentation from a theoretical, yet empirically grounded perspective, and for argumentation mining.

2.2. Annotation guidelines

Our annotation of argument schemes is grounded in the *Argumentum Model of Topics* (Rigotti and Morasso, 2010). Unlike other approaches (Walton et al., 2008b; Kienpointner and others, 1986; Van Eemeren and Houtlosser, 2006), it offers a *taxonomic hierarchy* of argument schemes with a limited number of schemes that are based on semantic, mutually exclusive distinctive criteria (Figure 1).

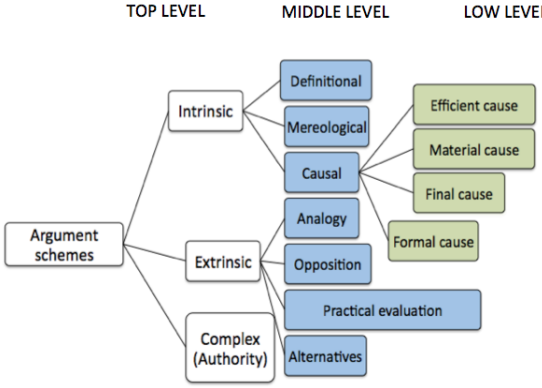


Figure 1: the AMT taxonomy of argument schemes

The annotation consists of two subtasks: 1) given a SUPPORT or REBUT relation, identify the argument scheme among the 8 middle level schemes (DEFINITIONAL, CAUSAL, MEREOLOGICAL, ANALOGY, OPPOSITION, PRACTICAL_EVALUATION, AUTHORITY) or NONE if no reasoning is present; and 2) identify the associated inference rule.

An early pilot annotation testing the first version of guidelines had been carried out on top of 30 persuasive essays from the corpus of Stab and Gurevych (2014), obtaining a fair inter-annotator agreement with trained but non-expert annotators (Musi et al., 2016).

The guidelines contain identification questions, linguistic clues and inferential rules for each argument scheme. Annotators are asked to first browse the identification yes/no questions and check whether inferential rules apply and linguistic clues are indeed present. The description of CAUSAL argument schemes contains, for example, the following information:

- *Identification Question*: is x a cause/effect of y or is it a means to obtain y?
- *Other clues*: Evaluations about actions play a role as common ground knowledge but do not constitute the premise textually expressed.
- *Inferential rules:*
  - if the cause is the case, the effect is the case
  - if the effect is the case, the cause is probably the case
  - if a quality characterizes the cause, then such quality characterizes the effect too
  - if the realization of the goal necessitates the means x, x must be adopted
  - if an action does not allow to achieve the goal, it should not be undertaken
  - if somebody has the means to achieve a certain goal, he will achieve that goal

Compared to the previous version, although the taxonomy and related theoretical insights have been maintained, the definition of the argument schemes that gave rise to the highest disagreement in the previous study have been clarified pointing to distinctive features. In particular, the CAUSAL argument schemes (from means to end) that we mentioned above have been frequently annotated by others as PRACTICAL EVALUATION, described as follows:

- *Identification Question*: does x express an evaluation about a state of affairs and does y express an advice/a recommendation about stopping/continuing the state of affairs the premise refer to?
- *Other clues*: the premise contains adjectives or other linguistic items which qualify something as more or less good
- *Inferential rules:*
  - if something is of important value, it should not be terminated
  - if something has a positive value, it should be supported/continued/promoted/maintained
  - if something has positive effects, it should be supported/continued/promoted/maintained
  - if something has a negative effect it should be terminated

These two argument schemes are perceived as similar due to the common reference to intentionality and to the frame of human action. To overcome this ambiguity, the guidelines have been refined to stress that the distinctive presence of evaluative propositions working as premises and of recommendation speech acts working as claims indicate a PRACTICAL EVALUATION scheme, but not a CAUSAL argument schemes. The reference to the linguistic clues has been retained only when they are argument scheme specific, and the list of inferential rules per argument scheme has been enriched with more examples (e.g. "if the state of affairs x shows a set of features which are also present in the state of affairs y and z holds for x, then z holds for y too" for ANALOGY argument schemes). A complete list of the inferential rules is available in the annotation guidelines. Furthermore, guidelines for the description of attack relations of the rebutt type have been added. Rebuttals are counterarguments which directly challenge the truth of a statement. Therefore, they have been conceived as arguments which support the negation of the proposition functioning as claim.

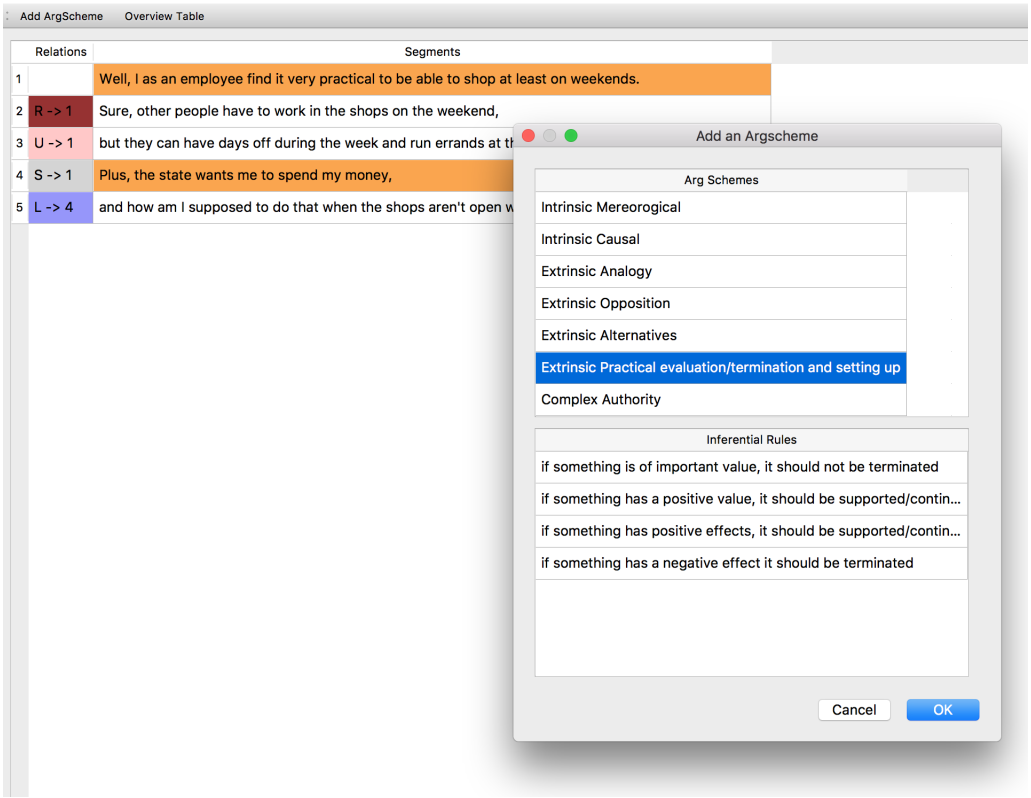


Figure 2: The ArgScheme Annotation Tool

3. The ArgScheme annotation tool

Human annotation is made most efficient when it is supported by an easy-to-use annotation tool. Since we are adding a new layer to the already-existing annotation of argument trees, none of the standard tools for text annotation can be used for our purposes. Therefore, we built the *ArgScheme* Annotator Tool, designed to provide a user-friendly interface for the labeling of support and attack relations with argument schemes. The annotation tool is written in C++ and was built within the Qt framework, which allows for building cross-platform GUI applications. We make the tool available for Linux, Windows, and Mac OS X.

Once a microtext is loaded, the tool shows each proposition paired with the type of relation it instantiates, as well as the index of the proposition it is connected to. After clicking on a relation, a window containing the list of argument schemes opens up. Once they have chosen the right argument scheme, annotators are shown a set of scheme-specific inference rules and are asked to select one. See Figure 2 for illustration. The result of the annotation is stored as additional attributes on the original XML format used in the microtext corpus, allowing for an easy mapping between all different annotation layers.

4. Results

4.1. Inter-Annotator Agreement

To measure inter-annotator agreement, two sets of 20 microtexts have been annotated by three annotators each (4 relations per microtext on average). The six annotators involved included four students with background in Linguistics and Argumentation, and 2 PhD students in NLP. Each annotator underwent a one-hour training session during which the guidelines have been explained and an exemplary analysis collaboratively discussed. To evaluate the reliability of the guidelines, we measured the inter-annotators agreement (IAA) using Fleiss κ to account for multiple annotators (Fleiss and Cohen, 1973). We obtained an IAA (k=0.296), which corresponds to fair agreement. To verify whether the idiosyncratic behavior of some annotators plays a role, we calculated the inter-annotator agreement pairwise for both sets. The IAAs for the first and second sets of microtexts (20 microtexts each) is given in Table 1. The set of three annotators is different for the 2 sets of 20 microtexts.

Annotators	κ **(1st set)**	κ **(2nd set)**
1,2	0.404	0.213
2,3	0.231	0.260
1,3	0.231	0.409

Table 1: pairwise IAA.

As can be noticed in Table 1, the 3rd annotator in the first set and the 2nd annotator in the second set are outliers. Zooming into their annotations, we noticed that annotator 3 has a tendency to signal the lack of argumentation schemes: s(he) annotated NONE in 21 cases, while the other two annotators always picked an argument scheme. Due to these individual tendencies, we have chosen majority voting as gold annotation. A manual check by an expert analyst has revealed the consistency of the choices made by the anno-

	Definitional	Mereorogical	Causal	Analogy	Opposition	Practical_evaluation	Alternatives	Authority	None
Definitional	8	1	10	0	2	14	1	0	16
Mereorogical		42	6	2	2	15	0	0	6
Causal			60	1	12	42	2	1	8
Analogy				12	0	3	0	0	2
opposition					8	5	3	3	3
Practical_evaluation						218	10	8	29
alternatives							4	0	4
Authority								24	0
none									6

Table 2: Confusion Matrix (since the matrix is symmetric we only show its upper triangular part)

tators who were in agreement. Considering the inherent difficulty of the task, which necessarily gives rise to ambiguities and disagreement (even among argumentation analysts) and the presence of outliers, this IAA score can be considered a positive index of the guidelines' reliability. On this basis, we have asked an additional set of seven highly expert annotators (researchers in Linguistics and Argumentation Theory) to annotate a set of 10 microtexts each. As a result, we release to the community a set of 112 microtexts fully annotated as to argument schemes.

4.2. Analysis of the disagreement space

With the aim of understanding the reasons underlying disagreements among annotators, we have built a confusion matrix as shown in Table 2.

We notice that CAUSAL scheme is still confused with PRACTICAL_EVALUATION, while DEFINITIONAL scheme is often confused with PRACTICAL_EVALUATION and NONE (i.e., no argument). The confusion between CAUSAL and PRACTICAL_EVALUATION has already been attested in the previous annotation project (Musi et al., 2016), as a confirmation of the perceived closeness of the two types of argument schemes.

From the qualitative analysis it emerges that DEFINITIONAL argument schemes have not been recognized by some annotators and have been annotated as NONE in particular for rebuttal relations (e.g., "Anti-virus software protects the users of a computer from dangers from the internet". "Admittedly they do not generally prevent you from catching a virus"). A possible explanation is that in those cases, the presence of category in the claim is less salient since it has to be recognized after the claim has been negated.

Finally, DEFINITIONAL argument schemes tend to be confused with PRACTICAL_EVALUATION argument schemes since the evaluative propositions featuring as premises in PRACTICAL_EVALUATION have been conceived as instances of categorizations. The premise-claim pair "Supermarkets and shopping centers should be allowed to open on any Sundays and holidays"."Considering the growing digitization of society, the traditional model of the weekend will soon be obsolete" constitutes an instance of PRACTICAL_EVALUATION argument scheme: a recommendation about carrying out an action is proposed on the basis of the positive/negative evaluation (e.g., "obsolete") of another state of affairs. The annotator that has chosen DEFINITIONAL has most likely interpreted the premise as a categorization. However, the categorization process is not at the basis of the inference that allows to support the conclusion. Even though the available data are not enough to draw generalizations, we consider them as hints to be used to refine the guidelines. In particular, we plan to stress on the differences between the propositional and the inferential levels of analysis.

5. Argument schemes and rhetorical relations

RST discourse relations (Mann and Thompson, 1988) overlap with argumentative discourse relations, particularly the subset of *Presentational* relations meant to increase the reader's positive inclination for the proposition functioning as nucleus clearly performing an argumentative act. The mapping of rhetorical discourse relations onto argumentative relations carried out by Stede et al. (2016) confirms this pragmatic similarity: the rhetorical relation REASON co-occurs for example 99 times with SUPPORT relations. Attack relations of the REBUT type tend, instead, to co-occur with the relation ANTITHESIS, while those of the UNDERCUT type with CONCESSIONS. In addition, Stede et al. (2016) showed that some *Subject-matter* relations, meant to allow the reader to recognize a semantic relation, may instantiate an argumentative relation in specific contexts. In order to extend this investigation by analyzing the correlations among rhetorical relations and argument schemes, we have mapped the annotation of argument schemes on top of RST relations and observed as well as explained attested overlaps or mismatches.

5.1. Mapping argument schemes to discourse relations

The mapping between RST discourse relations and argument schemes is done using an approach similar to the one introduced by Stede et al. (2016). Specifically, we convert both annotations to a common dependency structure where each relation has exactly one ADU in both its source and target. Argument scheme annotations follow a dependency structure for all relations except for the UNDERCUT relation where the target is another relation between ADUs,

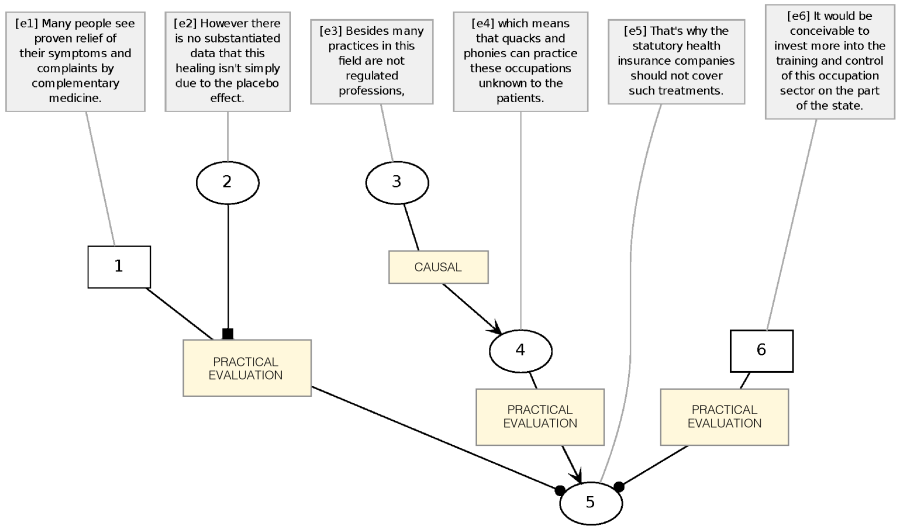


Figure 3: Argumentation structure and argument schemes of an example microtext

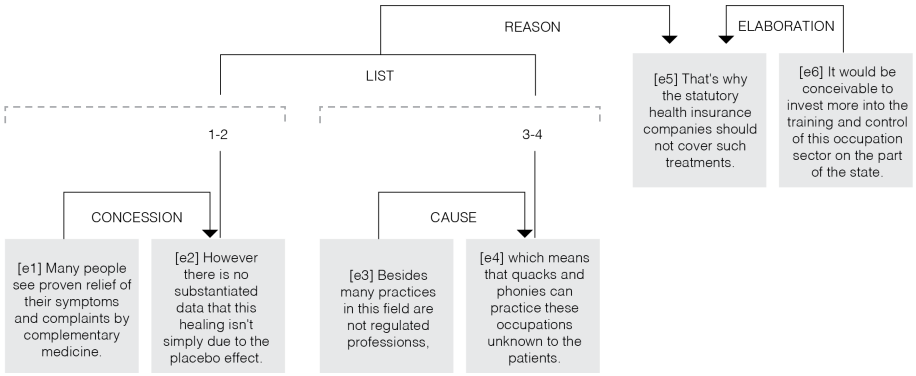
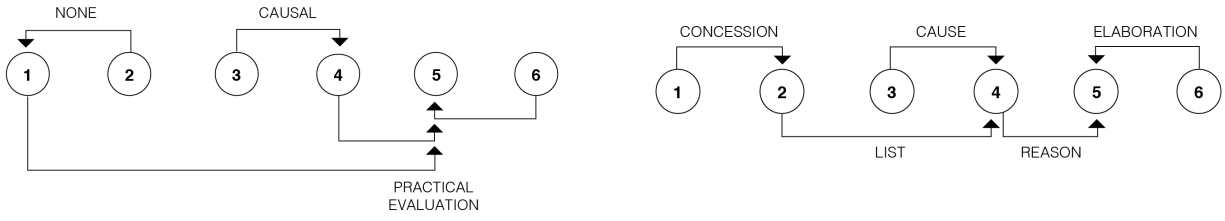


Figure 4: RST discourse structure of the same microtext given in Figure 3



(a) Argument Schemes

(b) RST

Figure 5: Conversion of argument schemes and RST relations to dependency structures

which is converted to the source of that relation (Stede et al., 2016). For RST, converting the relations to dependency structure is done in two steps. First, each *elementary discourse unit* (EDU) in RST is linked to its corresponding ADU in the argumentation structure. This is a one-to-one mapping when there is no join relation. If there is a join relation, multiple EDUs are mapped to a single ADU. Second, multi-nuclear nodes in RST are mapped with the ADU of its leftmost child as done in (Stede et al., 2016).

To illustrate, Figure 3 shows an example of argument schemes annotation, where the argument structure of the essay has six ADUs. Four out of the six ADUs represent the proponent's stances, who presents and defends the claim (round nodes), and the other two give voice to an imaginary opponent, who questions the claim (square nodes). The example has two REBUT relations, two SUPPORT relations and one UNDERCUT relation shown by the end symbol of each edge (circle for REBUT, arrow-head for SUPPORT an square for UNDERCUT). The argument schemes, which are available only for SUPPORT and REBUT relations, are shown as

the labels on each edge.

The corresponding RST structure for the same microtext is shown in Figure 4. It has the same six EDUs (there is a one-to-one mapping between EDUs are ADUs in this example) with five different discourse relations (REASON, LIST, CONCESSION, CAUSE, ELABORATION). It is worth noting that multi-nuclear discourse relations do not have an EDU as their target. They target another relation instead, which is the case for the LIST relation in the illustrated example.

The argumentation structure of the example shown in Figure 3 has an UNDERCUT relation. Therefore, the dependency structure of the argument scheme shown in Figure 5(a) is equivalent to the argument structure in Figure 3 except for the UNDERCUT relation, which maintains the same source and changes the target to the source of its target relation (i.e., $2 \rightarrow (1 \rightarrow 5)$ is changed to $2 \rightarrow 1$). To convert the RST structure of the microtext in Figure 4 to its dependency structure shown in Figure 5(b), the following steps are done:

1. All EDUs are labeled with their corresponding ADU using the argument structure of the microtext as a reference.

2. Converting multi-nuclear relations, LIST in this example, to the two following relations:

   (a) A relation with the source as the leftmost child of LIST and the target as the target of the next relation (i.e., target of REASON) and label as REASON.

   (b) A relation with the source as rightmost child of LIST and the target as its leftmost child and label as LIST.

The resulting common dependency structure of the argument schemes and RST discourse relations (Figure 5), allows us to analyze the overlap between the two annotations. We can see that three relations ($3 \rightarrow 4$, $4 \rightarrow 5$ and $6 \rightarrow 5$) exist in both RST and argument schemes, while one relation in each annotation does not exist in the other ($1 \rightarrow 5$ in argument schemes and $2 \rightarrow 4$ in RST). A study of the overlap between RST and argument schemes across the whole dataset is presented in the following section.

5.2. Analysis of the correlations

The overlap between argument schemes for SUPPORT and REBUT relations and RST discourse relations in our corpus is visualized in Table 3 and Table 4, respectively.

As a premise, it has to be stated that, with respect to the traditional taxonomy of RST relations (Mann and Thompson, 1988), two relations have been added by (Stede et al., 2016) according to Stede and Taboada (2017). Among *Presentational* relations, the new label REASON shows as nucleus "a subjective statement/thesis/claim, which R might not accept or might not regard as sufficiently important or positive" and as satellite "a subjective statement/thesis/claim"; the link between nucleus and satellite implies that "understanding S makes it easier for R to accept N, or to share the particular viewpoint of W". While sharing with REASON the same constraints on the nucleus, as well as the same effect ("R's belief in N is increased"), the relations EVIDENCE and JUSTIFY imply different constraints on the satellite, that has to state an objective description of a fact —EVIDENCE — or a fundamental attitude of the acting per-

	AUTHORITY	ALTERNATIVES	ANALOGY	OPPOSITION	PRACTICAL_EVALUATION	CAUSAL	DEFINITIONAL	MEREOROGICAL	NO SCHEME	NONE
ANTITHESIS	0	1	0	0	0	0	0	0	0	0
BACKGROUND	1	0	0	1	1	3	0	0	0	0
CAUSE	0	0	0	0	1	8	2	1	0	0
CIRCUMSTANCE	0	0	0	0	0	0	0	0	0	0
CONCESSION	0	0	0	0	0	0	0	0	0	0
CONDITION	0	0	0	0	0	1	0	0	0	0
CONJUNCTION	0	0	0	0	0	0	0	0	0	0
CONTRAST	0	0	0	0	0	0	0	0	0	0
DISJUNCTION	0	0	0	0	0	0	0	0	0	0
E-ELABORATION	0	0	0	0	0	0	0	0	0	0
ELABORATION	0	0	0	0	0	1	0	3	0	0
EVIDENCE	0	0	0	0	1	2	1	4	0	0
INTERPRETATION	0	0	0	0	0	0	0	0	0	0
JOINT	0	1	0	0	2	1	0	0	0	0
JUSTIFY	0	0	0	0	4	0	0	0	0	0
LIST	0	0	0	0	1	1	0	0	0	0
MOTIVATION	0	0	0	0	2	0	0	0	0	0
REASON	3	6	4	4	52	29	5	4	1	0
RESTATEMENT	0	0	0	0	0	1	1	0	0	0
RESULT	0	0	0	0	1	0	0	0	0	0
SAMEUNIT	0	0	0	0	1	0	0	0	0	0
SOLUTIONHOOD	0	0	0	0	0	0	0	0	0	0
UNLESS	0	0	0	0	0	0	0	0	0	0
NONE	3	1	3	0	45	17	6	4	1	1

Table 3: Overlap between RST relations and argument schemes for SUPPORT relations

	AUTHORITY	ALTERNATIVES	ANALOGY	OPPOSITION	PRACTICAL_EVALUATION	CAUSAL	DEFINITIONAL	MEREOROGICAL	NO SCHEME	NONE
ANTITHESIS	1	0	0	1	2	3	2	1	0	1
BACKGROUND	0	0	0	0	0	0	0	0	0	0
CAUSE	0	0	0	0	0	0	0	0	0	0
CIRCUMSTANCE	0	0	0	0	0	0	0	0	0	0
CONCESSION	1	0	0	1	3	0	1	1	0	0
CONDITION	0	0	0	0	1	0	0	0	0	0
CONJUNCTION	0	0	0	0	0	0	0	0	0	0
CONTRAST	0	0	0	1	0	0	0	0	0	0
DISJUNCTION	0	0	0	0	0	0	0	0	0	0
E-ELABORATION	0	0	0	0	0	0	0	0	0	0
ELABORATION	0	1	0	0	1	0	0	0	0	0
EVIDENCE	0	0	0	0	0	0	0	0	0	0
INTERPRETATION	0	0	0	0	0	0	0	0	0	0
JOINT	1	0	0	0	0	0	0	0	0	0
JUSTIFY	0	0	0	0	0	0	0	0	0	0
LIST	0	0	0	0	1	0	0	0	0	0
MOTIVATION	0	0	0	0	0	0	0	0	0	0
REASON	0	0	0	0	3	0	0	0	0	0
RESTATEMENT	0	0	0	0	0	0	0	0	0	0
RESULT	0	0	0	0	0	0	0	0	0	0
SAMEUNIT	0	0	0	0	0	1	0	0	0	0
SOLUTIONHOOD	0	0	0	0	0	0	0	0	0	0
UNLESS	0	0	0	0	1	0	0	0	0	0
NONE	2	3	3	6	33	11	6	5	1	0

Table 4: Overlap between RST relations and argument schemes for REBUT relations

son — JUSTIFY —. Among *Subject Matter* RST relations, the label E-ELABORATION differs from simple ELABORATION since the satellite provides more information about a single entity mentioned in the nucleus, rather than about the described state of affairs.
Comparing Table 3 and Table 4 it is clear that some RST relations overlap with argument schemes only in the presence of REBUT relations or SUPPORT relations, respectively.

Presentational RST Relations and Argument Schemes. As far as *Presentational* RST relations are concerned, ANTITHESIS and CONCESSION instantiate inferences only with REBUT relations, but do not seem to constrain the type of argument scheme at issue. Both relations entail a hierarchy of importance between the content of the nucleus (considered more important) and that of the satellite. The two relations differ in the type of comparison that is established: with ANTITHESIS, the propositions forming the nucleus and the satellite are deemed incompatible while with CONCESSIONS they are presented as both valid, while the nucleus is emphasized (e.g. "Although IBM s numbers haven't been staggering recently. You should buy IBM shares if you want to invest."). In both cases the inference underlying the creation of such a hierarchy can vary from context to context. However, it cannot serve e support function: the opposition relation signaled by the two relations makes the satellite support the negation of the nucleus and viceversa. In terms of frequency, the most frequent RST relation is REASON (92), followed by EVIDENCE (8), JUSTIFY (4) and MOTIVATION (2). While there is no attested overlap between argument schemes and EVIDENCE, JUSTIFY and MOTIVATION with REBUT relations, there is no incompatibility from a theoretical point of view.
The RST relation REASON overlaps with the widest range of argument schemes, with a preference for PRACTICAL EVALUATIONS — more than half of overlapping cases (e.g. "Actually only those people should pay a TV and radio licence fee who really watch ARD, ZDF, Arte etc. It is in fact good to support sophisticated programming through fees") — and CAUSAL argument schemes — ca. one third of overlapping cases. The subjectivity constraint imposed on the satellite well matches the semantic type of premises occurring with the argument scheme PRACTICAL_EVALUATION: evaluations, containing sentiment, presuppose a subjective perspective. CAUSAL argument argument schemes, coinciding with the REASON relation, show premises which are not descriptions of states of affairs, but speakers' interpretations (e.g."The universities in Germany should not under any circumstances charge tuition fees. The anticipated objectives of tuition fees can be achieved by other means."). Premises encoding factual states of affairs are, instead, present when CAUSAL argument schemes overlap with the RST relation EVIDENCE, due to the objectivity constraint imposed by this discourse relation on the satellite (e.g., "Our society is in danger of overheating due to 'never clocking off'.The last 20 years have seen the Sunday off work increasingly sacrificed to commerce."). From a semantic point of view, the most suitable argument scheme to overlap with EVIDENCE is MEREOLOGICAL: premises express states of affairs whom realization is attestable and, thus, objective ("Especially the home games have seen the team stay behind expectations. A sad highlight: Losing 60:87 to the weakish Mannheimers in a sold-out stadium."). However, due to the scarcity of data, it is not possible to statistically evaluate the significance of this preferential association.
The RST relations JUSTIFICATION and MOTIVATION overlap in our corpus with the argument scheme PRACTICAL EVALUATION only. This is probably due to the fact that the recommendation about setting up or terminating a certain action featuring as a claim with this type of argument scheme mirrors the agentive constraint imposed by the relations JUSTIFY (e.g. "Today I will delete my Facebook account.[Not only because] I'm constantly astonished about who wants to be my friend") and MOTIVATION (e.g "[But] still Germany produces way too much rubbish.We Berliners should take the chance and become pioneers in waste separation!") on their nuclei.

Subject-matter RST Relations and Argument Schemes. Turning to *Subject-matter* relations, BACKGROUND and RESTATEMENT overlap only with arguments schemes used in SUPPORT relations. These two relations are *per definition* incompatible with REBUT relations since the BACKGROUND relation has as effect that of increasing the reader's ability to comprehend the nucleus, while RESTATEMENT links a nucleus and a satellite with almost equivalent propositional contents. Although RESTATEMENT is primarily a textual and not a pragmatic relation, serving mainly to organize the text, it can acquire an argumentative function when the satellite is a rephrase that my help the reader in understanding the nucleus ("Intelligence services apparently enjoy indiscriminate liberties. No one can follow their activities in detail."). The inferential link between the satellite and nucleus is necessarily belonging to the INTRINSIC class since the state of affairs encoded by a rephrase necessarily belong to the same frame of that expressed by the rephrased proposition.
The RST relation CAUSE recurrently correlates with argument schemes (11 cases of overlap) that are for the majority, with no surprise, of the CAUSAL type. Although not constituting an inherently argumentative effect, making the reader aware of a cause/effect relation might have a persuasive outcome, especially when the effect/cause works as an argument in support of a general claim: for instance, the claim "Fees result in longer durations of studies" supported by the premise "That's costly!" through a causal relation, works as an argument for the standpoint "Tuition fees should not be charged in Germany."

6. Conclusion

We presented a *multi-layer* annotated corpus of 112 short argumentative texts that allows for correlating different levels of annotation and for studying dependencies between discourse relations and the underlying inferential moves. We presented a theoretically grounded annotation study of argument schemes both for support and attack relations together with a novel annotation tool. In the pilot annotation project to test the guidelines' reliability we obtained fair agreement. The qualitative analysis has shown the presence of outliers among the annotators and it has shed light on the main reasons underlying disagreement. We

plan to refine the guidelines accordingly. We have shown initial results from the correlation between rhetorical relations and argument schemes: *Presentational* RST relations such as REASON and ANTITHESIS select SUPPORT and REBUT relations, respectively, but do not point to specific argument schemes, while the relation EVIDENCE correlates with MEREOLOGICAL and AUTHORITY argument schemes. *Subject-matter* rhetorical relations such as CAUSE seems to constrain the argument scheme at issue.

7. Bibliographical References

Becker, M., Palmer, A., and Frank, A. (2016). Argumentative texts and clause types. In *Proceedings of the Third Workshop on Argumentation Mining*, Berlin. Association for Computational Linguistics.

Feng, V. W. and Hirst, G. (2011). Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 987–996. Association for Computational Linguistics.

Fleiss, J. L. and Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Freeman, J. B. (2011). *Argument Structure: Representation and Theory*. Argumentation Library (18). Springer.

Ghosh, D., Muresan, S., Wacholder, N., Aakhus, M., and Mitsui, M. (2014). Analyzing argumentative discourse units in online interactions. In *ArgMining@ ACL*, pages 39–48.

Green, N. (2017). Manual identification of arguments with implicit conclusions using semantic rules for argument mining. In *Proceedings of the 4th Workshop on Argument Mining*, pages 73–78.

Habernal, I. and Gurevych, I. (2017). Argumentation mining in user-generated web discourse. *Computational linguistics*, 43(1):125–179.

Kienpointner, M. et al. (1986). Towards a typology of argument schemes. *Proceedings of ISSA 1986*.

Kirschner, C., Eckle-Kohler, J., and Gurevych, I. (2015). Linking the thoughts: Analysis of argumentation structures in scientific publications. In *ArgMining@ HLT-NAACL*, pages 1–11.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technol.*, 16(2):10:1–10:25.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Musi, E., Ghosh, D., and Muresan, S. (2016). Towards feasible guidelines for the annotation of argument schemes. In *Proceedings of the third workshop on argument mining (ArgMining2016)*, pages 82–93.

Peldszus, A. and Stede, M. (2013). From argument diagrams to argumentation mining in texts: A survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 7(1):1–31.

Peldszus, A. and Stede, M. (2016). An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon 2015 / Vol. 2*, pages 801–816, London. College Publications.

Reed, C. and Rowe, G. (2004). Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(04):961–979.

Rigotti, E. and Morasso, S. G. (2010). Comparing the argumentum model of topics to other contemporary approaches to argument schemes: the procedural and material components. *Argumentation*, 24(4):489–512.

Schneider, J., Samp, K., Passant, A., and Decker, S. (2013). Arguments about deletion: How experience improves the acceptability of arguments in ad-hoc online task groups. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pages 1069–1080. ACM.

Stab, C. and Gurevych, I. (2014). Annotating argument components and relations in persuasive essays. In *COLING*, pages 1501–1510.

Stede, M. and Taboada, M. (2017). Annotation guidelines for rhetorical structure. *manuscript*.

Stede, M., Afantenos, S. D., Peldszus, A., Asher, N., and Perret, J. (2016). Parallel discourse annotations on a corpus of short texts. In *LREC*.

Van Eemeren, F. H. and Houtlosser, P. (2006). Strategic maneuvering: A synthetic recapitulation. *Argumentation*, 20(4):381–392.

Walker, M., Tree, J. F., Anand, P., Abbott, R., and King, J. (2012). A corpus for research on deliberation and debate. In *Proceedings of the Eighth Conference on International Language Resources and Evaluation (LREC)*, Istanbul/Turkey.

Walton, D., Reed, C., and Macagno, F. (2008a). *Argumentation schemes*. Cambridge University Press, Cambridge.

Walton, D., Reed, C., and Macagno, F. (2008b). *Argumentation schemes*. Cambridge University Press.

Discourse Coherence Through the Lens of an Annotated Text Corpus: A Case Study

Eva Hajičová, Jiří Mírovský
Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Malostranské nám. 25, 118 00 Praha 1, Czech Republic
{hajicova, mirovsky}@ufal.mff.cuni.cz

Abstract

A corpus-based study of local coherence as established by anaphoric links between the elements in the thematic (Topic) and the rhematic (Focus) parts of sentences in different genres of discourse. The study uses the Czech data present in the Prague Dependency Treebank and annotated for surface and underlying syntactic relations, the contextual boundness of tree nodes (from which the bi-partition of the sentence into Topic and Focus can be derived) and the coreference and bridging relations. Among the four possible types of the relations between anaphoric links and the Topic–Focus bipartition of the sentence, the most frequently occurring type is a link between the Topic of the sentence to the Focus of the immediately preceding sentence. In case there is an anaphoric link leading from the Focus of one sentence to the Topic or Focus of the immediately preceding sentence, this link frequently leads from a contextually bound element of the Focus, which supports the assumption that it is convenient to distinguish between "overall" Topic and Focus and the local Topic and Focus and/or the anaphoric relation is of the type of bridging and the relationship is often interpreted as a contrast. As for the relationship between the relations of the Topic-to-Topic type, due to the word order typological difference for Czech and English, these relations in Czech are not at all related to the syntactic function of subject.

Keywords: discourse, coherence, coreference

1. Introduction

One way how to look at discourse is to view it as a sequence of utterances linked by coherence relations (Halliday and Hasan, 1976). There are several possible ways how to account for these relations: e.g. they may be described on the basis of coreferential links between the elements present in the utterances, or on the basis of some discoursive links between segments of the adjacent utterances. In the present contribution, we first characterize some of the hitherto described approaches to this issue passing over to an examination of a possibility to look at the text coherence taking into account both the information structure of the utterances and the anaphoric relations. The proposed approach is based on a sample of Czech text corpus (the Prague Dependency Treebank 3.0) annotated for deep syntactic relations, information structure, and coreference and discourse relations.

2. Related Theories

One of the most deeply elaborated and best known theories of discourse (local) coherence is the so-called *centering theory* (Grosz, Joshi and Weinstein, 1983) based on the model of the local attentional states of speakers and hearers as proposed by Grosz and Sidner (1986). Each utterance in discourse is considered to contain a *backward looking center*, which links it with the preceding utterance, and a set of entities called *forward looking centers*; these entities are ranked according to language-specific ranking principles stated in terms of syntactic functions of the referring expressions. The highest ranked entity on the list is the so-called *preferred center*, i.e. the most likely link to the next following utterance. The *transitions* from one utterance to the following one are then specified by rules that capture their ordering: the most preferred is '*continue*', which means that the backward looking center of a given utterance equals the backward looking center of the preceding utterance and at the same time is the preferred center of the given utterance, followed by '*retain*' (the backward looking center of a given utterance equals the backward looking center of the preceding utterance but is not the preferred center of the given utterance), '*smooth shift*' (the backward looking center of a given utterance differs from the backward looking center of the preceding utterance but at the same time is the preferred center of the given utterance), and '*rough shift*' (the backward looking center of a given utterance differs from the backward looking center of the preceding utterance and is not the preferred center of the given utterance), in this order. The intuition which is behind this ranking of transitions is very close to those behind the notion of the low cost effort (Fais, 2004, p.120): "utterances that 'continue' the 'topic' of a previous sentence in a prominent position impose a lower inferential load, and are thus more coherent, than utterances which relegate the topic to less prominent position or which change the topic".

Interesting experiments investigating the effects of utterance structure and anaphoric reference on discourse comprehension examined in the context of utterance pairs with parallel constituent structure (e.g., *Josh criticized Paul. Then Marie insulted him.*) are reported in Chambers (1998). In previous studies of structural parallelism it was shown that an ambiguous pronoun (e.g., *him*) is biased to corefer with an antecedent in the same structural position (e.g., *Paul*). Of interest was whether parallelism can also influence the capacity for a pronoun to facilitate discourse comprehension and whether the centering model of discourse coherence can account for such effects. Most generally, centering predicts that a pronoun will increase coherence when it corefers with the subject of the previous utterance and that a single pronoun is sufficient to optimize local coherence. Three experiments are reported in the study addressing the interpretation of ambiguous pronouns, the comprehension of utterances containing a pronoun whose antecedent occupies either a parallel or nonparallel position, and also evaluating how the presence of multiple anaphoric links facilitates

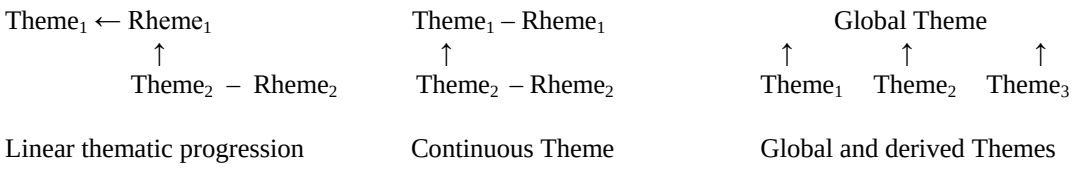


Figure 1: Types of thematic progressions according to Daneš (1970)

comprehension. Overall, the results reveal several limitations in centering theory and suggest that a more detailed account of utterance structure is necessary to capture how coreference influences the coherence of discourse.

A corpus-based evaluation of the preferences proposed in Centering theory is given by Poesio et al. (2004). The data used for that work are texts from two of the three domains of the GNOME corpus. The annotation included e.g. the break-up of sentences into clauses and the assignment of grammatical functions and anaphoric relations incl. bridging reference. An automatic script uses this information to compute utterances and the ranking of forward and backward looking centers. The study has reached some interesting results. For example, only the rule stating that if any forward looking center is pronominalized then the backward looking center is also pronominalized has been confirmed. The results concerning the constraint in its strong version that all utterances of a segment except for the first have exactly one backward looking center are especially negative; only if indirect realization is allowed, the constraint holds and is violated by between 20 to 25% of utterances. Another interesting observation is that if ranking is only required to be partial, some utterances end up with more than one backward looking center. As for the 'shifts' rule stating that (sequences of) continuations are preferred over (sequences of) retains, which are preferred over (sequences of) shifts, the tests revealed that there are more shifts than retains.

3. Thematic Progressions

Our long-time study of the information structure of the sentence (its Topic-Focus Articulation) has led us to the conviction that this aspect of the sentence structure is a good "bridge" towards a study of (at least one aspect of) the dynamic development of discourse. This, of course, is not a new idea: to our knowledge, its first comprehensive treatment, though clad in psychological rather than linguistic considerations, was given by Weil (1844, quoted here from the 1978 English translation). Weil recognized two types of the "movement of ideas", namely *marche parallèle* and *progression*: "If the initial notion is related to the united notion of the preceding sentence, the march of the two sentences is to some extent parallel; if it is related to the goal of the sentence which precedes, there is a progression in the march of the discourse" (p. 41). (It should not be overlooked that Weil also noticed a possibility of a reverse order called by him 'pathetic': "When the imagination is vividly impressed, or when the sensibilities of the soul are deeply stirred, the speaker enters into the matter of his discourse at the goal.", p. 45.)

In Czech linguistics, this idea is later reflected in Daneš' notion of *thematic progressions* (Daneš, 1970; 1974), explicitly referring to the relation between the theme and the rheme of a sentence and the theme or rheme of the next following sentence (a simple linear thematic progression and a thematic progression with a continuous theme), or to a 'global' theme (derived themes) of the (segment of the) discourse. Schematically, these three types can be captured as follows (see Fig. 1); the arrow denotes the direction of the relation.

In a slightly different but closely related vein, Firbas develops his ideas of the thematic and rhematic layers of a text (1995).

4. Corpus-Based Case Study

4.1 Methodology and its testing on a small sample

In our present corpus-based case study we focus our attention on the issue of local coherence as established by links between the thematic (Topic) and rhematic (Focus) parts of sentences. In particular, we want to verify if the classical observations valid for English as a language with a grammatically fixed word order, namely that there is a prevailance of "constant theme" (based on Mathesius´ 1947 study on the thematicity and "continuity" of English subject, and further analyzed esp. by Dušková, 2008; 2010), are also valid for a typologically different language, namely Czech, in which the word order is not guided by grammatical rules.

For this purpose, we use the data from the Prague Dependency Treebank (PDT in the sequel, for the most recent version see Hajič et al., 2018), which offers a good testing bed as it provides – in addition to the dependency underlying (deep) syntactic relations – annotation of (i) contextual boundness[1] from which the Topic–Focus

[1] The Topic-Focus bipartition of the sentence has been carried out automatically based on the primary opposition of contextually bound and non-bound items reflected in the PDT by a manual assignment of one of three values of the attribute of TFA. The distinction of contextual boundness should not be understood in a straightforward etymological way: an *nb* element may be 'known' in a cognitive sense (from the context or on the basis of background knowledge) but structured as non-bound, 'new', in Focus; see e.g (1) *John entered the room.* (2) *He first went to the window.* In (2), the window refers to cognitively 'known' object, i.e. known from the preceding context (the window of the room), but the sentence is structured in such a way that this element is contextually non-bound, it belongs to the focus of (2), as documented by placing the pitch on them if the sentence is read aloud.

bipartition of the sentence (TFA) can be derived and (ii) basic anaphoric relations, incl. some types of bridging. Such an annotation has allowed us to follow the occurrence of the two basic types of thematic progressions mentioned above, namely (i) the "progressive" rheme (Focus) in linear thematic progression, i.e. the Topic of the given sentence is anaphorically related to the Focus of the previous sentence, and (ii) continuous theme (Topic), i.e. the Topic of the given sentence is anaphorically related to the Topic of the previous sentence.

For the first step, in which we wanted to test whether our research methodology and the corpus material available may lead to some interesting and representative results, we have randomly chosen 6 documents of 5 genres with the total of 150 sentences and applied the algorithm for the division of the sentence into Topic and Focus based on the values of the TFA attribute (with values non-contrastive contextually bound, contrastive contextually bound and contextually non-bound, see Sgall, 1979, p. 180; Sgall et al. 1986, pp. 216ff; the original algorithm was later implemented and then tested on the whole of PDT and the results were reported in Hajičová et al., 2005, see also Rysová et al., 2015). As a result, we had at our disposal the total of 150 dependency trees with marked (binary) division into Topic and Focus and with the annotation of coreference and basic bridging relations between referring expressions of the adjacent sentences.

On this sample, we have followed four possible "thematic" relations between neighbouring sentences (the boundary between Topic and Focus is indicated in our examples by a slash):[2]

(i) (some element of the) Topic of the sentence *n* refers to (some element of the) Topic of the sentence *n-1* (denoted below as $T_{n-1} \leftarrow T_n$ and called above continuous Topic):

Myšlenka stručného ústavního zákona, který by prostě stanovil, že výdaje státního rozpočtu mají být kryty příjmy téhož roku, / se vyskytla v řadě zemí. Nejrozsáhlejší diskuse na toto téma / se odehrála v 80. letech ve Spojených státech.

The idea of a concise constitutional law, which would simply state that the state budget expenditures are to be covered by the same year's income,/ has occurred in a number of countries. The most extensive discussion on this issue / took place in the 1980s in the United States.

(ii) (some element of the) Topic of the sentence *n* refers to (some element of the) Focus of the sentence *n-1* (denoted below as $F_{n-1} \leftarrow T_n$ and called above progression of Focus):

Dnes je každý / pod novinářskou diktaturou. Diktatura jest / nehlučná, ale jest.

Today everybody is / under a journalist dictatorship. Dictatorship is / not noisy, but it is.

(iii) (some element of the) Focus of the sentence *n* refers to (some element of the) Focus of the sentence *n-1* (denoted below as $F_{n-1} \leftarrow F_n$):

Barevný terčík / usnadňuje nakládání pošty do kontejnerů. Během přepravy barva / zlepšuje přehled o tom, zda se zásilka nezpožďuje.

The coloured disc / makes easier the loading of the mail into containers. During the transport the colour / makes the information easier whether the article is not delayed.

(iv) (some element of the) Focus of the sentence *n* refers to (some element of the) Topic of the sentence *n-1* (denoted below as $T_{n-1} \leftarrow F_n$).

Novináři jsou / hlídací psi společnosti. Taková je / všeobecně sdílená představa o poslání novinářů.

Journalists are / watching dogs of the society. This is / a generally shared image of the mission of journalists.

"An element x refers to an element y" means that there is an anaphoric link (be it a proper coreference or a bridging relation) between the referring expressions x and y in adjacent sentences.

The genres of the selected documents were (i) interviews, (ii) plot, (iii) news, (iv) letter, and (v) essay, all the documents in PDT being of a journalistic domain. A first perfunctory look at the annotated data indicated that the interviews are a special kind of text, basically with two speakers, and that anaphoric links to the speakers (identified by pronouns or "dropped" pronouns) prevail, being mostly of the $T_{n-1} \leftarrow T_n$. type. Also the news and the texts marked as 'plot' did not provide an interesting material for the kind of analysis we aimed at, so that our attention was focussed first on the essay and letter genre (but see below Sect. 4.3 for an extended search).

Our starting assumption was that if the sentence is to be "about" something (i.e. about the Topic of the sentence), this "something" has to be somehow established (anchored) in the memory of the addressees. This anchor often is reflected in the text by an anaphoric reference from the Topic. This is why we first examined the types (assumed as prototypical) $T_{n-1} \leftarrow T_n$ and $F_{n-1} \leftarrow T_n$, that is the pairs of sentences in which the Topic refers to the Topic of the previous sentence ("continuous Topic") or in which the Topic refers to the Focus of the previous sentence ("progression of Focus").

This assumption has been confirmed in both genres, but there was a difference which of the two types prevails in which genre: $T_{n-1} \leftarrow T_n$ occurred twice as often than $F_{n-1} \leftarrow T_n$ in the letter document, while in the essay genre, $F_{n-1} \leftarrow T_n$ occurred three times as often than $T_{n-1} \leftarrow T_n$. With the other, non-prototypical relations, both types occurred rather rarely in the letter genre but the type $F_{n-1} \leftarrow F_n$ was surprisingly frequent in the essay type (13 occurrences as compared to 20 of $F_{n-1} \leftarrow T_n$ and 8 of $T_{n-1} \leftarrow T_n$). Under a more detailed inspection, it has been found that in most of these cases the anaphoric relation of an element in F_n leads from a contextually bound element of Focus. This finding is in an agreement with the assumption (made explicit in Hajičová, Partee and Sgall, 1998) of the theory of TFA we subscribe to that the recursive character of this articulation makes it possible (or even necessary) to distinguish between the "overall" bipartition of the sentence into its Topic and Focus and the local partitioning within these two parts into what may be

[2] The examples in this section are original sentences from the PDT.

called "local Topic" and "local Focus". An illustrative example are the sentences in (iii) above, repeated here for convenience:

Barevný terčík / usnadňuje nakládání pošty do kontejnerů. Během přepravy barva / zlepšuje přehled o tom, zda se zásilka // nezpožďuje.

The coloured disc / makes easier the loading of the mail into containers. During the transport the colour / makes the information easier whether the article // is not delayed.

The two expressions connected by an anaphoric link (bridging type) are *pošty* [mail] and *zásilka* [article], both in the Focus part of the sentences they are part of. However, the element *zásilka* [article] in the (global) Focus of the second sentence of the segment carries the TFA value "contextually bound" and as such is in the local Topic of the sentence; the division of the (underlined) embedded clause into its own Topic and Focus is indicated by a double bar.

4.2 Verification of the results on a larger corpus

To confirm our observations and to find a more substantial support for our initial assumption, we carried out a second step in the analysis, in which: (i) we put under scrutiny a larger amount of data from the essay genre, (ii) we complemented our identification of the "overall" Topic and Focus by a more detailed analysis of the inner structure of these parts as for the value of contextual boundness within the TFA attribute, and (iii) we paid a more detailed attention to the type of anaphoric relations, to see whether the difference between (pure) coreference and bridging plays some important role.

The new sample contained another 100 annotated sentences from the genre of essay, 79 of which were linked by links of coreference or bridging relations. (It should be noted that we followed only links between two adjacent sentences and did not analyze sentences the links from which pointed to some more distant preceding context.) In this sample, the $F_{n-1} \leftarrow T_n$ sequences prevailed only slightly (28 cases) followed by the $F_{n-1} \leftarrow F_n$ type (24 cases), the $T_{n-1} \leftarrow T_n$ type (11 cases) and the $T_{n-1} \leftarrow F_n$ type (8). This is to say that the ratio between what we consider to be typical relations (from the Topic in the second sentence of the pair) and the non-typical relations (from the Focus of the second pair) was almost balanced (39 versus 32). Under a more detailed analysis of the 24 cases of the $F_{n-1} \leftarrow F_n$ type relations, it has been confirmed that in most cases, the anaphoric link leads from a contextually bound element of F_n which again may serve as a support to distinguish local topics and local foci from the overall Topic and Focus. Another explanation of the unexpected links between elements of the Foci of the adjacent sentences is the fact that 12 out of the 24 $F_{n-1} \leftarrow F_n$ links were bridging relations in which the mentioning in the second sentence has a contrastive character (i.e. the contrast between a whole and a part of the whole, set or subset) or is accompanied by a particle (such as *only*) with a focusing function which by itself is contrastive.

To obtain a more general picture of the distribution of the different types of anaphoric relations as attested in larger data, we applied the analysis onto the whole subset of the essay genre in the PDT corpus; this sample contains 189 documents with the total of 6 858 sentences, among which 4 606 adjacent sentences contained a pairwise anaphoric link. The figures obtained confirm even more clearly the picture described above: the $F_{n-1} \leftarrow T_n$ sequences prevailed considerably (1 771 cases) over the $T_{n-1} \leftarrow T_n$ type (1 278 cases) while the number of the non-prototypical links was much lower (the $F_{n-1} \leftarrow F_n$ type with 1 004 cases and the $T_{n-1} \leftarrow F_n$ type with 553 cases).

4.3 Application of the analysis to different genres

As the PDT annotated material allows for a comparison of the obtained results with respect to different genres,[3] we applied the analysis onto a collection of 10 genres, namely (i) advice, (ii) comment, (iii) description, (iv) essay, (v) invitation, (vi) letter, (vii) news, (viii) overview, (ix) review and (x) survey. We put under scrutiny documents containing more than 20 sentences; we have identified the total of 17 307 anaphoric links and the results obtained for all these genres taken together are as follows: as for the relations leading from the Topic of the given sentence to the preceding sentence, the $F_{n-1} \leftarrow T_n$ sequences again prevail (6 292 cases, i.e. 36%) over the $T_{n-1} \leftarrow T_n$ type (5 029 cases, i.e. 29%); the total number of these typical relations is 11 321 (65%). This result indicates that continuous topic, i.e. the anaphoric relations between Topics of two sentences, are considerably less frequent than the progression of focus, i.e. anaphoric reference from the Topic of the given sentence to an element in the Focus of the preceding sentence. As for the non-typical relations, i.e. relations leading from the Focus of the given sentence to the Topic or Focus of the preceding sentence, they occur only in 5 986 cases (35%); among them, the $F_{n-1} \leftarrow F_n$ sequences prevail (3 665, 21%) over the $T_{n-1} \leftarrow F_n$ type (2 321 cases, 14%), see Table 1 for the distribution of the types of thematic progressions according to the type of anaphoric relation.

	All anaphora		Coreference		Bridging	
F ← T	6 292	36%	5 091	38%	1 201	31%
T ← T	5 029	29%	3 834	29%	777	20%
F ← F	3 665	21%	2 888	21%	1 195	31%
T ← F	2 321	14%	1 629	12%	692	18%

Table 1: Distribution of the types of thematic progressions according to the type of anaphoric relation

Looking at the genres separately, nine of the ten analyzed genres provide a similar distribution of thematic types, the only conspicuous differences being attested in the genre of "overview" with a balanced occurrence of the $F_{n-1} \leftarrow T_n$, $T_{n-1} \leftarrow T_n$ and $F_{n-1} \leftarrow F_n$ sequences (29%, 29% and 31%, respectively) and a considerably lower frequency of the $T_{n-1} \leftarrow F_n$ type (11%, which corresponds to the general situation). So far, we cannot offer any explanation for this phenomenon. Another observation relates to the genre of "letter": there the prevalence of the $F_{n-1} \leftarrow T_n$ is even more perspicuous than with the other types (44% compared to 28% of $T_{n-1} \leftarrow T_n$ for all relations, and 50%

[3] There are 20 different labels for the genre categories assigned to the PDT documents (Zikánová et al. 2015, p. 27f.)

vs. 35% for bridging only). As a matter of fact, looking at the bridging type of anaphoric relations separately, the distribution of the types of thematic relations differs almost in all genres from the distribution of the thematic relations identified by coreferential links. The explanation may concern two points: (i) first and most importantly, the manual annotation of bridging relations is very difficult in general and open for annotation inaccuracy, the more so in our corpus where we recognize only a few basic types of bridging; (ii) with the genre of letters and advice in particular, we may expect a more or less intimate, personal relations between the writer/author and the addressee, which may lead to a more frequent use of (indirect) anaphoric links (bridging), and the interaction has also a more familiar character, which may lead to a higher occurrence of contrast in focus.

As the PDT corpus offers the possibility to examine relations present at different layers of annotation rather than to restrict the attention to a single level, we also used our material to look at a possible relationship between the anaphoric links and the surface syntactic function of the elements in question. As mentioned above, the analysis of thematic progressions in the writings on English (e.g. Dušková, 2008; 2010) suggests that there is a predominance of "constant theme", which, for English, means, that there is a "continuity" of subject (see Mathesius, 1947 for the function of subject in English). Our material documents that this is not the case in Czech: in the second sample of 100 sentences, among the $T_{n-1} \leftarrow T_n$ type (i.e. the type relevant for the issue under investigation), there was no instance of a subject-to-subject link.

5. Summary and Future Work

To sum up our observations based on annotated corpus (the PDT multi-layered annotation of Czech sentences), the following results have been reached:

(a) among the four possible types of the relationship between anaphoric links and the Topic-Focus bipartition of the sentence, the most frequently occurring type is a link between the Topic of the sentence to the Focus of the previous sentence; this is in contrast to the assumption of Fais (2004) based on the low cost and Chamber's (1998) assumption of structural parallelism, but in favour of Poesio et al.'s (2004) finding on the predominance of shifts to retain relation;

(b) in case there is an anaphoric link leading from a sentence to the Focus of the next following sentence,

(i) this link frequently leads to a contextually bound element of the Focus of the next sentence, which supports the assumption that it is convenient to distinguish between "overall" Topic and Focus and the local Topic and Focus; and/or

(ii) the anaphoric relation is of the type of bridging, which is often interpreted as a contrast;

(c) as for the relationship between the relations of the Topic-to-Topic type, due to the word order typological difference for Czech and English, these relations in Czech are not at all related to the syntactic function of subject.

We are aware that the observations and results presented in this paper are only first steps in the corpus-based study of the relationships between the Topic-Focus articulation of the sentence and the anaphoric relations. The resource we have at our hands offers a possibility to follow several aspects of this relationship, out of which the following issues are on our future programme:

(i) The multilayered annotation of the PDT allows for a systematic study of the relation between the syntactic structure – both surface (in terms of subject, object, etc.) and deep (in terms of Actor, Patient, Addressee, etc.) – and the frequency of the types of thematic progressions; in this way we may arrive at explanations based on the typological differences between different languages.

(ii) Not only the adjacent sentences but also sentences linked by anaphoric relations "at a distance" should be examined, with the aim to investigate whether the length of the chain of anaphoric relations and the size of the "gap" ("hole") in between the coreferring expressions make a difference.

(iii) In connection with the point (ii), it will be interesting to examine whether the size of the above-mentioned "gap" makes a difference in the use of a particular anaphoric type or a preference of the use of a particular type of surface expression (pronoun, bare noun, nominal group etc.).

(iv) A more complex task consists in the examination of the dynamics of discourse in terms of the activation of elements of the knowledge shared by the speaker/author and the addressee (for the formulation of the task and the first text analysis see Hajičová, 1993; 2003, and Hajičová and Hladká, 2008). The multilayered PDT annotation offers a useful resource by capturing the forms of the referring expressions, their syntactic functions and the values of contextual boundness, all being the factors determining or influencing the discourse flow.

Acknowledgements

The authors gratefully acknowledge support from the Grant Agency of the Czech Republic (projects GA17-03461S and GA17-06123S) and the Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071). The research reported in the present contribution has been using language resources developed, stored and distributed by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071).

Bibliographical References

Chambers, C. (1998). *Structural Parallelism and Discourse Coherence: A Test of Centering Theory,* Journal of Memory and Language, Volume 39, Issue 4, November 1998, 593–608.

Daneš, F. (1970). *Zur linguistischen Analyse der Textstruktur.* Folia linguistica 4:72–78.

Daneš, F. (1974). Functional Sentence Perspective and the organization of the text. In: Daneš, Ed. *Papers on*

Functional Sentence Perspective. Prague: Academia, 106–128.

Dušková, L. (2008). Theme movement in academic discourse. In: M. Procházka and J. Čermák, Eds., *Shakespeare between the Middle Ages and Modernity.* From translators art to academic discourse. Prague, FF UK, 221–247.

Dušková, L. (2010). Rozvíjení tématu v akademickém a narativním textu [The development of theme in an academic and narrative text]. In Čmejrková S., Hoffmannová J., Havlová E., eds.: *Užívání a prožívání jazyka. K 90. narozeninám Františka Daneše.* Praha, Karolinum, 253–260.

Fais, L. (2004). Inferable centers, centering transitions, and the notion of coherence. *Computational linguistics 30*, 119–150.

Firbas, J. (1995): On the thematic and rhematic layers of a text. In: Warvick B. et al., *Organization of Discourse. Proceedings of the Turku Conference 1995*, pp. 59–72.

Grosz, B. (1977). The representation and use of focus in dialog understanding. *Technical Note 15,* Artificial Intelligence Center, SRI International, Menlo Park, California.

Grosz, B. J., Joshi, A. K. and S. Weinstein (1983). Providing a unified account of definite noun phrases in discourse. In: *Proceedings of the 21st Annual meeting of the Association for Computational Linguistics,* Cambridge, Mass., 44–50.

Grosz, B. and Sidner, C. L. (1986). Attention, Intentions and the structure of discourse. *Computational Linguistics, 12,* 175–204.

Grosz, B. J., Joshi, A. K. and S. Weinstein (1995). Centering: A Framework for modeling the local coherence of discourse. *Computational Linguistics, 21*, 203–225.

Hajičová, E. (1993). *Issues of sentence structure and discourse patterns*. Charles University, Prague.

Hajičová, E. (2003). Aspects of Discourse Structure. In: *Natural Language Processing between Linguistic Inquiry and System Engineering* (ed. by W. Menzel and C. Vertan), Iasi, 47–56.

Hajičová E., Havelka J. and K. Veselá (2005). Corpus Evidence of Contextual Boundness and Focus. In: *Proceedings of the Corpus Linguistics Conference Series*, University of Birmingham, Birmingham, UK, ISSN 1747-9398, pp. 1–9.

Hajičová, E. and B. Hladká (2008). What does sentence annotation say about discourse? In *18th International Congress of Linguists*, Abstracts, The Linguistic Society of Korea, Seoul, Korea, 125–126.

Hajičová, E., Partee, B. H. and P. Sgall (1998). *Topic-Focus Articulation, Tripartite Structures, and Semantic Content,* Dordrecht , Kluwer Academic Publishers.

Hajičová, E. and J. Vrbová (1982). On the role of the hierarchy of activation in the process of natural language understanding. In: Horecký J., Ed., *Coling 82 – Proceedings of the Ninth International Congress of Computational Linguistics.* Amsterdam: John Benjamins. 107–113.

Halliday, M. A. K. – Hasan, R. (1976): *Cohesion in English*. Longman: London

Mathesius, V. (1947). O funkci podmětu. [On the function of subject.]. In *Čeština a obecný jazykozpyt,* Melantrich, Prague, 100–103.

Poesio, M., Stevenson, R., Di Eugenio, B. and J. Hitzeman (2004). Centering: a parametric theory and its instantiations. *Computational Linguistics 30,* 309–363.

Rysová, K., Mírovský, J. and E. Hajičová (2015). On an apparent freedom of Czech word order. A case study. In: *14th International Workshop on Treebanks and Linguistic Theories (TLT 2015)*, Warszawa, Poland, 93–105.

Sgall, P. (1979): Towards a Definition of Focus and Topic. In *Prague Bulletin of Mathematical Linguistics,* 31:3–25; 32:24–32; reprinted in *Prague Studies in Mathematical Linguistics*, 7, 1981, 78:173–198Sgall, P. Hajičová, E. andJ. Panevová (1986): *The Meaning of the Sentence in its Semantic and Pragmatic Aspects*. In Mey, J. L. (ed). Dordrecht: Reidel and Prague:Academia.

Weil, H. (1844). *De l'order des mots dans les langues anciennes comparées aux langues modernes*, Paris: Joubert. Translated by Charles W. Super as *The order of words in the ancient languages compared with that of the modern languages,* Boston: Ginn, 1887, reedited and published by John Benjamins, Amsterdam 1978.

Zikánová, Š. et al. (2015). *Discourse and Coherence.* Prague: Charles University, MFF ÚFAL, Prague, Czech Republic.

Language Resource References

Hajič, J., Bejček, E., Bémová, A., Buráňová, E., Hajičová, E., Havelka, J., Homola, P., Kárník, J., Kettnerová, V., Klyueva, N., Kolářová, V., Kučová, L., Lopatková, M., Mikulová, M., Mírovský, J., Nedoluzhko, A., Pajas, P., Panevová, J., Poláková, L., Rysová, M., Sgall, P., Spoustová, J., Straňák, P., Synková, P., Ševčíková, M., Štěpánek, J., Urešová, Z., Vidová Hladká, B., Zeman, D., Zikánová, Š. and Z. Žabokrtský (2018). *Prague Dependency Treebank 3.5.* Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University, Prague, Czech Republic, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-2621.

Automatic Prediction of Discourse Connectives

Eric Malmi[1,*], **Daniele Pighin**[2], **Sebastian Krause**[2,*], **Mikhail Kozhevnikov**[2]
[1]Aalto University
Espoo, Finland
eric.malmi@aalto.fi
[2]Google
Zürich, Switzerland
{biondo,bastik,qnan}@google.com

Abstract

Accurate prediction of suitable discourse connectives (*however*, *furthermore*, etc.) is a key component of any system aimed at building coherent and fluent discourses from shorter sentences and passages. As an example, a dialog system might assemble a long and informative answer by sampling passages extracted from different documents retrieved from the Web. We formulate the task of discourse connective prediction and release a dataset of 2.9M sentence pairs separated by discourse connectives for this task. Then, we evaluate the hardness of the task for human raters, apply a recently proposed decomposable attention (DA) model to this task and observe that the automatic predictor has a higher F_1 than human raters (32 vs. 30). Nevertheless, under specific conditions the raters still outperform the DA model, suggesting that there is headroom for future improvements.

Keywords: discourse connectives, decomposable attention model, discourse relation prediction

1. Introduction

Discourse connectives, also referred to as discourse markers, discourse cues, or discourse adverbials, are used to bind together and to explicate the relation between pieces of text. It is a common language class exercise to be asked to fill in suitable connectives to a text in order to improve the text flow. Similarly, it is important for computational summarization and text adaptation systems to be able to fill in suitable discourse connectives to produce natural-sounding utterances.

In this work, we study the problem of automatic discourse connective prediction. We limit ourselves to connectives which appear at the beginning of a sentence, linking the sentence to the preceding one. Even in this limited setting, an automatic discourse connective predictor has many concrete use cases. For example, in a question-answering setting it could help to generate answers by collating sentences from multiple sources. In extractive text summarization, it could be used to determine what is the best way to join two sentences that used to be separated by one or more sentences. As part of a text-authoring application, it could suggest suitable connectives at the beginning of a sentence. In the literature, discourse connective prediction has been recently studied merely as an intermediate step for the well-studied problem of implicit discourse relation prediction (Xu et al., 2012; Zhou et al., 2010). However, considering the aforementioned applications, we argue that connective prediction makes an interesting and relevant problem in its own right.

The contributions of this work are twofold:

1. We present an extensive experimental study on the problem of discourse connective prediction and show that a recently proposed decomposable attention model (Parikh et al., 2016) yields a good performance on this task. The model clearly outperforms a popular word-pair model and obtains a better performance than human raters on the same task and data.

2. We describe the dataset that we collected, consisting of 2.9 million adjacent sentence pairs (with and without a connective) extracted from the English Wikipedia. For 10 000 sentences, we also include connectives filled in by human raters. The dataset is publicly available at: https://github.com/ekQ/discourse-connectives

[*]Work performed during an internship at Google.

2. Related Work

A few earlier works study discourse connective prediction alone, but recently it has been studied merely as an intermediate step for discourse relation prediction. Next we provide a brief overview of these two lines of work, starting from the latter.

2.1. Predicting Connectives for Implicit Discourse Relation Prediction

Implicit discourse relation prediction has attracted considerable attention in recent years (Braud and Denis, 2016; Liu and Li, 2016; Qin et al., 2016; Qin et al., 2017; Rutherford and Xue, 2015; Wu et al., 2017; Zhang et al., 2016). Earlier Pitler et al. (2008) showed that if a discourse connective is known, the explicit discourse relation[1] can be inferred with a 93.09% accuracy, which has inspired several efforts at predicting connectives to improve implicit discourse relation prediction. Zhou et al. (2010) predicted connectives using an N-gram language model, whereas Xu et al. (2012) employed word pairs and a selection of linguistically informed features. Liu et al. (2016), on the other hand, showed that predicting both connectives and relations using a convolutional neural network in a multi-task setting improves the relation prediction performance.

2.2. Discourse Connective Prediction

Some earlier works have focused on connective prediction alone and developed various hand-crafted features for distinguishing between connectives. For example, Elhadad and McKeown (1990) explored pragmatic features for distinguishing between the connectives `but` and `although`,

[1]Later, it has been shown that a single discourse connective can actually convey multiple discourse relations (Rohde et al., 2015; Rohde et al., 2016).

and between because and since. Later Grote et al. (1998) developed a specialized lexicon for discourse connectives based on the relevant constraints and preferences associated with the connectives. While these works do not present an experimental evaluation of the proposed systems, we evaluate our connective prediction models extensively in order to understand their applicability to real-life scenarios. Furthermore, our aim is to learn the representations of the two arguments and their relationship automatically which allows us to distinguish between a large set of connectives without extensive manual efforts required to craft features that separate the connectives.

In addition to predicting the most suitable discourse connective, several methods have been developed for predicting the presence of a discourse connective (Yung et al., 2017; Di Eugenio et al., 1997; Patterson and Kehler, 2013). We also predict the presence of a connective by considering [No connective] as one of the classes to be predicted.

3. Data Collection

We compile a list of 79 discourse connectives based on the Penn Discourse Treebank (PDTB) (Prasad et al., 2008). Since our focus is on sentence concatenation, we ignore the (forward) connectives, which typically point to the following sentence rather than the previous one, such as "After the election, [...]". However, for several ambiguous connectives, the forward use can be ruled out by requiring a comma after the connective (e.g. Instead,); we include such connectives in our data. Discontinuous connectives, such as "If [...] then [...]", are not included.

Data samples for discourse connective prediction can be collected from any large unannotated text corpus. In this instance, we use the English Wikipedia[2] and collect every pair of consecutive sentences within the same paragraph where the latter sentence begins with one of the 79 discourse connectives. As a result, we obtain a dataset of 1.95 million sentence pairs separated by a connective. Additionally, we collect 0.91 million examples of consecutive sentences not separated by a discourse connective, labeled as [No connective], for a total of 2.86 million sentence pairs.[3]

The frequency distribution of the connectives is very skewed; however occurs 720 334 times, whereas else, only 43 times in the beginning of a sentence. In order to make the connective prediction task more feasible for the models and for human raters, we select a subset of sufficiently frequent and distinct connectives (e.g. for example is included but for instance is not since it conveys the same meaning and is less frequent). The details of the selection process are omitted in the interest of space, but the resulting 19 connectives are listed in Table 1.

Finally, we split the data into train, development, and test sets. We balance the connective classes, since in an unbalanced dataset most examples would be labeled as [No connective] and many connectives would be extremely under-represented, limiting the applicability of the resulting classifier. For the development and test sets, we pick 500 samples per connective (including [No connective]) by under-sampling without replacement. This results in two balanced datasets of 10 000 samples. For the training set, we pick 20 000 samples per connective by under-sampling the majority classes and oversampling the minority classes, creating a balanced dataset of 400 000 samples. Connective samples from a single Wikipedia article are not included in more than one of the three datasets to avoid over-fitting through potential repetition within a single article.

In comparison with the PDTB dataset, which contains information about both discourse connectives and discourse relations, the main advantage of the collected dataset is its size. PDTB contains only 40 600 examples (1.4% of the size of the collected dataset), which causes sparsity issues (Li and Nenkova, 2014). This can slow down the development of new models, particularly complex neural models that often require large training datasets to generalize well.

Connective	Occurrences	Connective	Occurrences
however	720 334	on the other hand	20 301
for example	111 711	in particular,	16 011
and	73 644	indeed,	15 286
meanwhile,	57 971	overall,	9 513
therefore	44 064	in other words	8 888
finally,	33 076	rather,	5 596
nevertheless	32 952	by contrast,	4 605
instead,	30 973	by then	4 279
moreover	25 583	otherwise,	3 563
then,	21 731		

Table 1: The list of 19 connectives studied in the experiments in addition to the [No connective] class. Only the connectives which are sufficiently frequent and distinct in their meaning have been selected.

4. Connective Prediction Models

The decomposable attention (DA) model was recently introduced by Parikh et al. (2016) for the *natural language inference* (NLI) problem which aims to classify entailment and contradiction relations between a premise and a hypothesis. Discourse connective prediction is related to the NLI problem since entailment and contradiction can be explicitly indicated by certain connectives (for instance, therefore and by contrast, respectively). However, the larger number of classes makes connective prediction more challenging. DA was shown to yield a state-of-the-art performance on the NLI task while requiring almost an order of magnitude fewer parameters than previous approaches. For all these reasons, it seems natural to apply the DA model to the connective prediction problem.

Marcu and Echihabi (2002) proposed to use word-pair features to predict discourse relations based on discourse connectives mapped to these relations. Similarly, many later implicit discourse relation prediction models are based on word-pair features (Marcu and Echihabi, 2002; Pitler et al., 2009; Xu et al., 2012; Zhou et al., 2010) or aggregated word-pair features (Biran and McKeown, 2013; Rutherford and Xue, 2014). Therefore, we use a model called WORDPAIRS to have a baseline for the DA model.

[2] A snapshot from September 5, 2016.

[3] Note that our models are tested only on consecutive sentences, for which the ground truth connectives are known, but they can be applied to connect also disjoint sentences.

4.1. The Decomposable Attention Model

The DA model consists of three steps, *attend*, *compare*, and *aggregate*, which are executed by three different feed-forward neural networks F, G, and H, respectively. As input, the model takes two sentences $\mathbf{a}$ and $\mathbf{b}$ represented by sequences of word embeddings. The sequences are padded by "NULL" tokens to fix their lengths to 50 tokens.

In the **attend** step, the model computes non-negative attention scores for each pair of tokens across the two input sentences. This computation ignores the order of the tokens and it produces soft-alignments from $\mathbf{a}$ to $\mathbf{b}$ and *vice versa*.

In the **compare** step, the model computes comparison vectors between each input token and its aligned sub-phrase. The aligned sub-phrase is a linear combination of the embedding vectors of the other sentence weighted by the attention scores.

Finally, in the **aggregate** step, the comparison vectors are summed over the tokens of a sentence and then the aggregate vectors of the two sentences are concatenated. The resulting vector is fed into the third feed-forward network which outputs $\hat{\mathbf{y}}$ containing scores for each class. The predicted class is given by $\hat{y} = \arg\max_i \hat{y}_i$.

The weights of the three networks are randomly initialized, after which the model is trained in an end-to-end manner. Our implementation of the DA model has the following differences compared to the original model described by Parikh et al. (2016): (i) we do not use the self-attention mechanism which was reported to provide only a small improvement over the vanilla version of DA; (ii) we do not project down the embedding vectors but use 100-dimensional word2vec embeddings (Mikolov et al., 2013) which are updated during the training; (iii) we use layer normalization (Ba et al., 2016) which makes the model converge faster.

4.2. The Word-Pair Model

The WORDPAIRS model considers as features all word pairs which appear across the two arguments (e.g. word A appears in Arg 1 and word B in Arg 2) in at least five samples in the training dataset. Such features are employed by many implicit discourse relation prediction models (Marcu and Echihabi, 2002; Pitler et al., 2009; Zhou et al., 2010; Xu et al., 2012). Additionally, we incorporate single word features (e.g. word *A* appears in Arg 2) since these slightly improved the results. With these binary features, we train logistic regressors using the one-vs-rest scheme to predict one of the 20 different connectives.[4]

5. Experiments

Next we present experimental results on discourse connective prediction using human raters, the DA model and the WORDPAIRS model. For this task, we remove the connective (if any) from the second sentence in each test pair, and measure the ability of the model (or the raters) to identify the removed connective.

[4]We trained two versions of the WORDPAIRS model: using stochastic gradient descent with mini-batches and using LIBLINEAR with 100k samples (i.e. 25% of the training data) which we could fit into the memory of a 256 GB machine. The reported results are based on the latter approach, which performed better.

Model	Macro F_1	Accuracy
RANDOM	5.00	5.00
WORDPAIRS	14.81	15.60
DA	**31.80**	**32.71**
Human Raters	23.72	23.12

Table 2: Discourse connective prediction performance of a RANDOM baseline, the WORDPAIRS model, the decomposable attention model (DA) and human raters.

5.1. Accuracy of Human Raters

To better understand what is reasonable to expect from an automatic predictor, we use a crowd-sourcing platform to ask human raters to reconstruct the removed connectives for each of the 10 000 test sentence pairs. Each sentence pair is annotated by three (not necessarily the same three) native English speakers. The raters are shown the two sentences, the latter of which starts with a *[Connective goes here]* placeholder, and asked to select the most suitable connective from the 20 options, including `[No connective]`. This layer of human annotations is also released as part of the connective dataset. The raters are instructed to pick the most natural connective in case there are multiple suitable options. Furthermore, they are asked to pick `[No connective]` only if adding a connective would make the concatenation sound ungrammatical or artificial, or if the two sentences seem to be completely disconnected. The sentences are not pre-processed apart from upper-casing the first character of the second sentence to avoid giving away the presence of a connective in the original sentence. The order of the connectives is randomized, except for `[No connective]` which is always shown last.

On the whole test-set, human annotators achieve a macro-averaged F_1 score of 23.72. The confusion matrix generated by the raters' decisions is presented on the left side of Figure 1. It shows that the raters are strongly biased towards `[No connective]` despite the indication to refrain from using it. A similar bias was observed by Rohde et al. (2016) for the task of filling in a suitable conjunction before a discourse connective. There are at least two possible explanations for this bias: (i) in the sake of clarity and in line with common scientific writing guidelines, Wikipedia editors tend to use connectives quite generously, and (ii) the artificial balancing of the datasets makes `[No connective]` under-represented in the test data compared to the actual distribution of discourse connectives vs. `[No connective]`. The confusion matrix also shows that there are clusters of connectives that raters tend to confuse, even though they do not necessarily encode exactly the same relation. Examples are `rather,` and `instead,`, `for example` and `in particular,`, `on the other hand` and `by contrast,`. For 57.1% of the test questions, there is a consensus among at least two raters and for 11.4%, all three raters agree on the most suitable connective.

5.2. Accuracy of the Models

In this section, the DA model and the WORDPAIRS model are employed to perform the same task as the human raters,

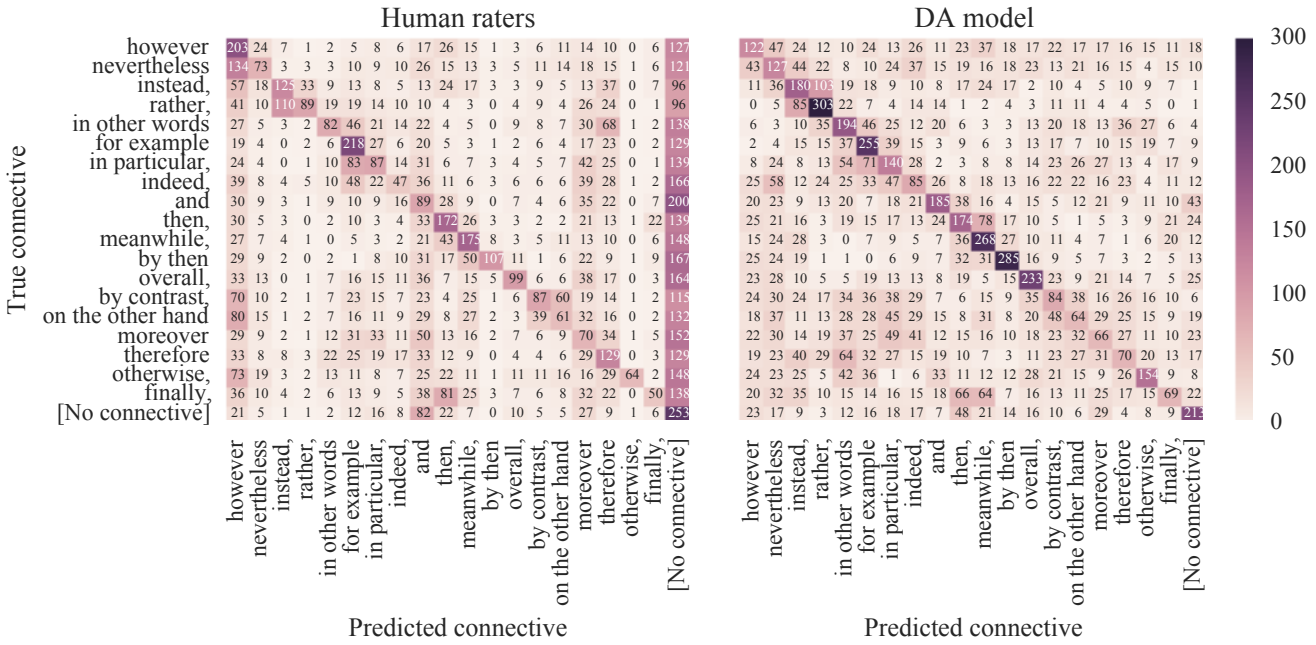


Figure 1: Confusion matrices for the human raters (left) and DA model (right) predicting the discourse connectives used by Wikipedia authors. The counts of the human raters are divided by three (i.e., the number of raters) for easier comparison.

i.e., learning to reconstruct the connective possibly removed from the beginning of the second sentence in each test pair. A balanced dataset is used for both training and testing the models as described in Section 3. The DA model is evaluated using the following hyper-parameters optimized on the development set: network size (one hidden layer with 200 neurons), batch size (64), dropout ratios for the F, G, and H networks (0.68, 0.14, and 0.44, respectively), and learning rate (0.0018). The model is implemented in TensorFlow (Abadi et al., 2015) and the training is run for 300 000 batch steps. The results, reported in Table 2, show that DA clearly outperforms the WORDPAIRS baseline with an F_1 score of 31.80 vs. 14.81.

5.3. Comparison Between the Raters and the Decomposable Attention Model

Table 3 compares the accuracy of DA predictions to the rater decisions. The macro-averaged F_1 score of human raters is 23.72 which is, quite surprisingly, lower than the F_1 score of the DA model, 31.80. The difference is smaller when considering majority votes on the subset of 5 714 tasks for which there is a consensus among at least 2 out of 3 raters, which results in a 30.36 F_1 score for the raters. On these less ambiguous cases, the model performance also increases to 32.68.

As we mentioned in Section 5.1., human raters are clearly less eager to introduce a connective than Wikipedia editors. Therefore, we also evaluate the setting in which we exclude the questions for which either the ground-truth label, or the rater-assigned majority label, or the model-assigned label is `[No connective]`. The results, listed in the last line of Table 3, show that under these conditions human raters actually outperform the model.

The confusion matrix of DA is shown on the right side of Figure 1. For each connective, the true connective is the most frequent prediction. Connective `on the other hand` has the lowest F_1 score (15.06), whereas `by then` has the highest (57.29). Some of the most frequent mistakes are between similar connectives, such as `however` vs. `nevertheless`, and `instead,` vs. `rather,`. These errors are by and large consistent with those of human raters (left side of the figure). This confirms that the model is accurately capturing the meaning of the relation, and when it does not select the gold connective it is making similar approximations to what people would do. Furthermore, the Figure 1 shows that raters have a more pronounced tendency to select frequent connectives, such as `however` and `and`. To further exemplify, in Table 4 we show a selection of wrong and correct decisions made by DA and human raters. A manual inspection of these and other examples shows that in some cases a larger context than the previous sentence is required for inferring the connective. For instance, to correctly decide whether `finally,` is more suitable than `then,` one may have to inspect a larger context.

Setting	n	Raters (F_1)	Model (F_1)
A	10 000	23.72	**31.80**
B	5 714	30.36	**32.68**
C	3 204	**41.97**	36.65

Table 3: Macro-averaged F_1 scores for human raters and the decomposable attention model. The three settings are: (A) All test set items; (B) Only the items for which there is a consensus among at least 2 out of 3 raters; (C) Consensus items ignoring those where either ground truth, or the rater assigned, or the model assigned label is `[No connective]`.

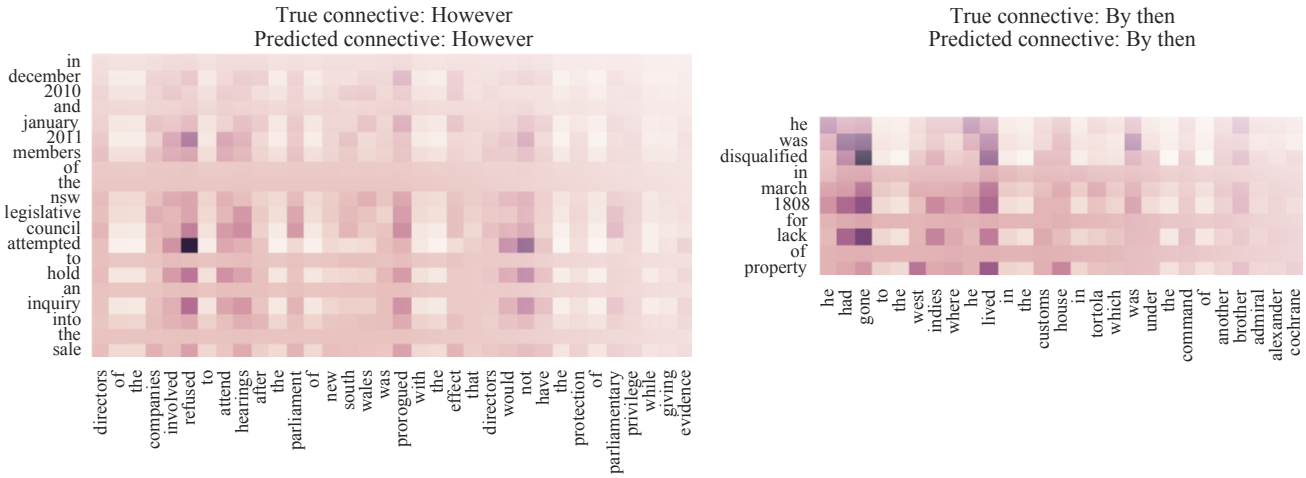


Figure 2: Two examples of the alignment matrix between the first (y-axis) and the second argument (x-axis) generated by the DA predictor. The darker the color, the higher the alignment score.

Arguments	Gold	DA	Raters
Arg 1: From 1913 to 1917 Lucey served as Illinois Attorney General. Arg 2: Lucey was appointed to the Illinois Public Utilities Commission in 1917 and served until 1920.	**then**	**then**	[No connective]
Arg 1: In Ahmedgarh Municipal Council, Female Sex Ratio is of 901 against state average of 895. Arg 2: Child Sex Ratio in Ahmedgarh is around 841 compared to Punjab state average of 846.	**moreover**	**moreover**	meanwhile
Arg 1: If the question was answered correctly, the team would receive the next clue. Arg 2: The chosen team member would have to try again.	**otherwise**	**otherwise**	then
Arg 1: Lee then continued on to Boston, arriving 25 June. Arg 2: The ranks of General Washington's Navy were being thinned by captures.	**meanwhile**	by then	**meanwhile**
Arg 1: Cooper was promoted as an alternate leader to Ahern. Arg 2: It was thought he could shore up the National Party's vote in its conservative rural heartland.	**in particular**	however	**in particular**
Arg 1: Taylor urged Pius XII to explicitly condemn Nazi atrocities. Arg 2: Pius XII spoke against the "evils of modern warfare", but did not go further.	**instead**	in particular	**instead**

Table 4: Examples of mistakes and correct predictions made by the DA model and by the raters.

5.4. Model Interpretation

An advantage of the DA model is that it is possible to examine which words the model attends to when inferring a connective. In some cases, the attended words are clearly meaningful semantically or linguistically, whereas in other cases the soft-alignment matrix that the model produces is harder to interpret. Examples of the former case are represented in Figure 2, which shows the alignment matrices from the tokens of the first sentence (y-axis) to the tokens of the second sentence (x-axis) so that the rows sum to 1. In the left example, the model correctly predicts `however` as the connective after aligning the word *attempt* with *refuse* and *not*. These word pairs indicate contrast which makes `however` a likely connective. In the right example, the model aligns the phrase *was disqualified* with *had gone* and correctly predicts `by then` as the connective. The corresponding tenses, i.e., past and past perfect, respectively, are likely clues of the presence of `by then`.

6. Conclusions

We studied the problem of discourse connective prediction, which has many useful applications in text summarization, adaptation and conversationalization. We collected a dataset of 2.9 million pairs of consecutive sentences and connectives, and made it publicly available to facilitate further research on this problem, as well as other related bi-sequence classification tasks. We showed that the recently proposed decomposable attention model performs surprisingly well on the connective prediction task, even better than human raters on the same representative test set consisting of 10 000 samples. We also observed that, unlike the model, human raters have a preference for implicit connectives, as they do outperform the model if the comparison is restricted to the cases in which the majority of raters agrees on an explicit connective. The alignment matrices produced by the model suggest that the predictor is picking up relevant lexical, syntactic and semantic clues. The confusion matrix of the predictor shows very similar error patterns to the matrix generated from human raters, further confirming the meaningfulness of the decisions made by the model.

7. Acknowledgements

We would like to thank Cesar Ilharco for his help on running the experiments.

8. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2015). Tensorflow: Large-scale machine learning on heterogeneous systems. *Software available from tensorflow.org*.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Biran, O. and McKeown, K. (2013). Aggregated word pair features for implicit discourse relation disambiguation. In *Proc. ACL*.

Braud, C. and Denis, P. (2016). Learning connective-based word representations for implicit discourse relation identification. In *Proc. EMNLP*.

Di Eugenio, B., Moore, J. D., and Paolucci, M. (1997). Learning features that predict cue usage. In *Proc. EACL*.

Elhadad, M. and McKeown, K. R. (1990). Generating connectives. In *Proc. the 13th Conference on Computational Linguistics*.

Grote, B., Stede, M., et al. (1998). Discourse marker choice in sentence planning. In *Proc. Ninth International Workshop on Natural Language Generation*.

Li, J. J. and Nenkova, A. (2014). Reducing sparsity improves the recognition of implicit discourse relations. In *Proc. SIGDIAL*.

Liu, Y. and Li, S. (2016). Recognizing implicit discourse relations via repeated reading: Neural networks with multi-level attention. In *Proc. EMNLP*.

Liu, Y., Li, S., Zhang, X., and Sui, Z. (2016). Implicit discourse relation classification via multi-task neural networks. In *Proc. AAAI*.

Marcu, D. and Echihabi, A. (2002). An unsupervised approach to recognizing discourse relations. In *Proc. ACL*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proc. NIPS*.

Parikh, A. P., Täckström, O., Das, D., and Uszkoreit, J. (2016). A decomposable attention model for natural language inference. In *Proc. EMNLP*.

Patterson, G. and Kehler, A. (2013). Predicting the presence of discourse connectives. In *Proc. EMNLP*.

Pitler, E., Raghupathy, M., Mehta, H., Nenkova, A., Lee, A., and Joshi, A. K. (2008). Easily identifiable discourse relations. In *Proc. Coling 2008: Companion volume: Posters and Demonstrations*.

Pitler, E., Louis, A., and Nenkova, A. (2009). Automatic sense prediction for implicit discourse relations in text. In *Proc. ACL-IJCNLP*.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A. K., and Webber, B. L. (2008). The penn discourse treebank 2.0. In *Proc. LREC*.

Qin, L., Zhang, Z., and Zhao, H. (2016). A stacking gated neural architecture for implicit discourse relation classification. In *Proc. EMNLP*.

Qin, L., Zhang, Z., Zhao, H., Hu, Z., and Xing, E. P. (2017). Adversarial connective-exploiting networks for implicit discourse relation classification.

Rohde, H., Dickinson, A., Clark, C., Louis, A., and Webber, B. (2015). Recovering discourse relations: Varying influence of discourse adverbials. In *Proc. LSDSem Workshop*.

Rohde, H., Dickinson, A., Schneider, N., Clark, C. N., Louis, A., and Webber, B. (2016). Filling in the blanks in understanding discourse adverbials: Consistency, conflict, and context-dependence in a crowdsourced elicitation task. In *Proc. LAW X Workshop*.

Rutherford, A. and Xue, N. (2014). Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. In *Proc. EACL*.

Rutherford, A. and Xue, N. (2015). Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In *Proc. NAACL-HLT*.

Wu, C., Shi, X., Chen, Y., Huang, Y., and Su, J. (2017). Leveraging bilingually-constrained synthetic data via multi-task neural networks for implicit discourse relation recognition. *Neurocomputing*, 243:69–79.

Xu, Y., Lan, M., Lu, Y., Niu, Z. Y., and Tan, C. L. (2012). Connective prediction using machine learning for implicit discourse relation classification. In *Proc. IJCNN*.

Yung, F., Duh, K., Komura, T., and Matsumoto, Y. (2017). A psycholinguistic model for the marking of discourse relations. *Dialogue & Discourse*, 8(1):106–131.

Zhang, B., Xiong, D., Su, J., Liu, Q., Ji, R., Duan, H., and Zhang, M. (2016). Variational neural discourse relation recognizer. In *Proc. EMNLP*.

Zhou, Z. M., Lan, M., Niu, Z. Y., Xu, Y., and Su, J. (2010). The effects of discourse connectives prediction on implicit discourse relation recognition. In *Proc. SIGDIAL*.

Handling Rare Word Problem using Synthetic Training Data for Sinhala and Tamil Neural Machine Translation

Pasindu Tennage, Prabath Sandaruwan, Malith Thilakarathne, Achini Herath, Surangika Ranathunga
Department of Computer Science and Engineering
University of Moratuwa
Katubedda 10400, Sri Lanka
{pasindu.13, prabath.sandaruwan.13, malith.13, narmada.ah.13,surangika,}@cse.mrt.ac.lk

Abstract
Lack of parallel training data influences the rare word problem in Neural Machine Translation (NMT) systems, particularly for under-resourced languages. Using synthetic parallel training data (data augmentation) is a promising approach to handle the rare word problem. Previously proposed methods for data augmentation do not consider language syntax when generating synthetic training data. This leads to generation of sentences that lower the overall quality of parallel training data. In this paper, we discuss the suitability of using Parts of Speech (POS) tagging and morphological analysis as syntactic features to prune the generated synthetic sentence pairs that do not adhere to language syntax. Our models show an overall 2.16 and 5.00 BLEU score gains over our benchmark Sinhala to Tamil and Tamil to Sinhala translation systems, respectively. Although we focus on Sinhala and Tamil NMT for the domain of official government documents, we believe that these synthetic data pruning techniques can be generalized to any language pair.

Keywords: Neural Machine Translation, POS Tagging, Morphological Analysis, Data Augmentation

1. Introduction

Neural Machine Translation (NMT) is the current state-of-the-art machine translation architecture that aims at building a single neural network that can be jointly tuned to maximize the translation performance (Bahdanau et al, 2014). Despite being successful in producing acceptable outputs for language pairs having large parallel corpora (Sennrich et al, 2016), NMT performs poorly for language pairs that lack the luxury of having sufficiently large parallel data (Tennage et al, 2017). This is due to the requirement of numerous instances of sentence pairs with words occurring in different contexts in order to accurately train a NMT model. Being an under-resourced language pair that is unable to satisfy this requirement, Sinhala and Tamil NMT falls short of reaching state-of-the-art performances (Tennage et al, 2017).

Limited corpus size directly influences the rare word problem. Rare word problem refers to the inability of the neural network to properly model words that appear in the corpus only a very few times. Being morphologically rich languages, there exist many inflections for each word in Sinhala and Tamil languages. Hence having many rare words in the corpus is inevitable.

One way to handle the rare word problem is to increase corpus size using synthetic parallel training data. To generate synthetic data, Fadaee et al. (2017) presented a possible technique. This creates new contexts for rare words when generating synthetic training data, thus giving a possible solution for the rare word problem. However, the main limitation of this approach is that this synthetic sentence pair generation technique does not take language syntax into consideration, which eventually lowers expected BLEU score gain.

In this paper, we present two sentence pruning techniques based on Parts of Speech (POS) tagging and morphological analysis to remove synthetic sentence pairs that do not preserve language syntax. Compared to Fadaee et al.'s (2017) method, POS tagging method shows an improvement of 1.04 and 2.12 BLEU score gains for Sinhala to Tamil (Si-Ta) and Tamil to Sinhala (Ta-Si) models, respectively. Use of morphological analysis improves the quality of translation by 1.26 and 2.98 BLEU scores, respectively. Overall, synthetic parallel training data methods yield an improvement of 2.16 and 5.00 BLEU score gains over our benchmark Si-Ta and Ta-Si models.

2. Background and Related Work

2.1 Sinhala and Tamil Languages

Sinhala language descends from Indic language family and Tamil from Dravidian family (Pushpananda et al, 2014). Being morphologically rich, Sinhala has up to 110 noun word forms and up to 282 verb word forms and Tamil has 40 noun word forms and up to 240 verb word forms. Both these languages have the same word order of Subject-Object-Verb.

Tennage et al, (2017) have built the first NMT system for this language pair. Lack of language resources and data sparseness that is caused by morphological variances have been identified as the key factors that hinder the translation performance (Tennage et al, 2017).

2.2 Neural Machine Translation

NMT is an end-to-end translation process that treats a word as the smallest unit (Bahdanau et al, 2014). Encoder Decoder (ED) architecture with attention mechanism is the current state-of-the-art NMT architecture (Cho et al, 2014). In the ED architecture, recurrent activation function is applied recursively over the input sequence until the end when the final internal state of the recurrent neural network (RNN) contains the summary of the whole input sentence. Decoder computes RNN's internal state based on the summary vector, the previous predicted word, and the previous internal state. Using internal hidden state of the decoder, it is possible to score each target word based on how likely it is to follow all the preceding translated words. Using softmax normalization, scores are turned into probabilities.

2.3 Data Augmentation

Fadaee et al. (2017) presented a data augmentation approach that targets low-frequency words by generating

new sentence pairs containing rare words in new, synthetically created contexts. They have produced experimental results on low-resource settings and have achieved considerable improvement over the benchmark systems. They have focused mainly on fluency and grammatical structure of synthetic training data, and have disregarded its syntax correctness.

Strategies to train with monolingual data without changing the neural network architecture have been proposed by Sennrich et al. (2016). It is based on the intuition that encoder-decoder NMT architecture already has the capacity to learn the same information as a language model. By pairing monolingual training data with an automatic back-translation, synthetic parallel training data are generated. Quality of synthetic training data generated using this method highly depends on the machine translator that is used for back translation.

3. Methodology

3.1 Initial Data Augmentation

For initial synthetic sentence generation, we use the technique used by Fadaee et al. (2017). Initially we obtain a list of rare words by considering the unique words and their counts. Words that appear only *R* (rare word threshold) times or less are considered as rare words. For each rare word *r*, we iterate through each sentence pair in our parallel corpus.

In the below expressions, s_i and t_i denote the i^{th} word in the source sentence and target sentence, respectively. Each word in the source sentence is iterated through and substituted by *r*. Trigram language probability around *r* is checked thereafter.

If the i^{th} source word is substituted,

Original language probability p_1 = LM (s_{i-1}, s_i, s_{i+1})
Synthetic language probability p_2= LM (s_{i-1}, r, s_{i+1})

if ($p_2 > M * p_1$): this is a valid source substitution (M is fluency threshold).

To generate the target side synthetic sentence, we need to substitute the translation of r to the word in the corresponding original target sentence that is aligned to the word that we removed from the source sentence. Statistical approach of automatic word alignment (Och, Ney, 2004) is used to accomplish this task. Using automatic word alignment, it is possible to get the index of the target word that is aligned with the source word that was removed.

To get the translation of a rare word *r*, phrase tables that are generated using word alignment are used. For a given word *e*, there exist several possible translations *f* according to the generated phrase tables. To find the exact translation, we use a two-way translation probability as follows.

$$\text{translation}(e) = \text{argmax}_{f \in \text{possible translations}} (p(f|e) * p(e|f))$$

where,

p(*f*|*e*) : Probability of *f* being the translation of *e*.
p(*e*|*f*) : Probability of *e* being the translation of *f*.

If there exists a target side word *q* corresponding to *r*, with two-way translation probability greater than *T* (translation threshold), we select it as a viable translation for *r*. *q* is substituted to the word that is aligned to the word that was removed in source side. If the trigram language probability around that word is greater than *M* times the original trigram language probability, then we select it as a correct target word substitution.

A synthetic sentence pair that satisfies all these conditions is added to the synthetic parallel corpus. To reduce distortion of the meaning, only a single rare word substitution per sentence was allowed. Use of language modeling ensures the fluency of synthetic sentences whereas the use of the translation modeling ensures the correspondence between source sentence and target sentence. Table 1 depicts an example synthetic sentence pair.

Original Sentence Pair	**Synthetic Sentence Pair**
එසේ පවරා දෙනු ලැබුවේ කවරෙකුටද (/esea pavaraa denu lAbuwea kavarekuTada/) - (It was assigned to whom?)	එසේ පවරා දෙනු ලැබුවේ ඔබටද? (/esea pavaraa denu labuwea obaTada /) (It was assigned to you?)
அவ்வாறு யாருக்கு ஒப்படைக்கப்பட்டுள்ளது? (/avvaaRu yaarukku oppataikkappattuLLadhu/) (It was assigned to whom?)	அவ்வாறு யாருக்கு உங்களுக்கும்? (/avvaaRu yaarukku ungkaLukkum/) (It was to whom and to you)[1]

Table 1: Initially Generated Synthetic Sentence Pair

Human evaluation of the synthetic parallel training data generated using this method revealed that the resulting sentences do not preserve language syntax. Hence, we investigated on methods to prune the synthetic sentence pairs that do not preserve language semantics.

3.2 Parts of Speech Tagging

POS tags contain important syntactic information about the word in the context that it appears. Based on this property, we further increased the quality of synthetic training data by checking the POS tag of each rare word that is substituted.

Initially, the original parallel corpus is POS tagged. Then using the methodology proposed in section 3.1, all possible synthetic sentence pairs are generated. Then the synthetic parallel sentences are also POS tagged. Algorithm 1 describes this method.

Here,
s_i= word that was removed from source sentence.
t_i= word that was removed from target sentence.
r= rare word that was introduced to source sentence.
t= translation of r that was introduced to target side.

Algorithm 1 – POS tag based pruning

1: **if** (*POS tag of* s_i *== POS tag of r*) and (*POS tag of* t_i *== POS tag of t*) **then**
2: *Keep the synthetic sentence pair*
3: **else**
4: *Remove it from the corpus*

[1]Word ordering in Sinhala is different from English. The exact English translations are “To whom was it assigned?” / “Was it assigned to you?” (“To whom” is replaced by “to you”)

3.3 Morphological Analysis

To further preserve language semantics, we use morphological features. In this research, we pay attention to morphological features of Sinhala nouns only, since most of the rare words are noun word forms. We use two morphological features of Sinhala nouns,

1. Count (වචනය /*wachanaya*/)
2. Case (විභක්තිය /*wibhaktiya*/)

Grammatical number or the count is an inflectional feature belonging tothe realm of morphosyntax. Count can take three values, definite singular (DS), indefinite singular (IS) and definite plural (DP). Case is a suffix that is added to a stem to derive nouns in different meanings. Case forms differ in terms of the syntactic contexts in which they may occur. Sinhala language consists of 9 cases, ප්‍රථමා (/prathamaa/) - Nominative, කර්ම (/karma/) - Accusative, කර්තෘ (/kartru/) – Auxiliary, කරණ (/karaNa/) - Instrumental, සම්ප්‍රදාන (/sampradaana/) - Dative, අවදි (/awadi/) - Abalative, සම්බන්ධ (/sambandha/) - Possesive, ආධාර (/aadhaara/) - Locative, ආලපන (/aalapana/) - Vocative(Priyanga, Ranatunga& Dias, n.d.).

Synthetic parallel corpus that was generated in section 3.2 is further improved using morphological features. For a given word, there exists a variable number of case - count combinations. In this approach, we check whether the case - count combinations of the word that was removed have an intersection with the case - count combinations of the word that is introduced synthetically. We consider it as a syntax preserving sentence pair only if there exists an intersection of at least one element.

4. Experimental Setup

4.1 Data Collection and Preprocessing

Official government document translation is the domain used in this translation task. We used the parallel corpus developed by Farhath et al. (2017). Parallel corpus features government documents, annual reports, gazette papers, establishment codes, order papers, official letters and parliament documents. Characteristics of the Sinhala-Tamil parallel dataset are shown in Table 2.

Language	Total Words	Unique Words	Sentences
Sinhala	267,613	21,548	19,153
Tamil	226,160	38,651	

Table 2: Characteristics of the parallel dataset

Parallel corpus was divided into 3 parts: training set (14653 sentence pairs), validation set (4000 sentence pairs), and testing set (500 sentence pairs).

4.2 Experimental Setup

The open source NMT system: OpenNMT (Klein et al. 2017) was used for the experiments. GIZA++ (Och, Ney, 2004) was used for automatic word alignment. It uses the standard alignment heuristic grow-diag-final for word alignment. Tri-gram language models were trained for both source side and target side using the Stanford Research Institute Language Modeling toolkit (Stolcke et al, 2002) with Kneser- Ney smoothing. For translation evaluation, Bilingual Evaluation Understudy (BLEU) metric (Papineni et al, 2002) was used.

4.3 Benchmark Training

Using the above parallel corpus, Si-Ta and Ta-Si translation models were trained. Training involved two steps: pre-processing and model training. After completing the pre-processing step, two dictionaries (source dictionary and target dictionary) were generated to index mappings. Using two dictionaries and the serialized file, a model was trained with a 2-layer LSTM with 500 hidden units on both encoder and decoder.

4.4 Initial Data Augmentation

Out of 15383 number of unique words in the corpus, 6421 words appeared only once in the Sinhala language side, whereas corresponding values for Tamil side was found to be 31186 and 17238, respectively. Hence, we chose rare word threshold R to be 1. Considering the tradeoff between the number of sentences generated and semantic preservation of synthetic data, we chose fluency threshold M to be 2 and translation threshold T to be 0.9.

4.5 POS Tagging

Both original and synthetic parallel corpora that were generated in the previous section were POS tagged. We used the POS tagger developed by Fernando et al. (2016) for Sinhala, and the POS tagger developed by the Computational Linguistic Research Group (2017) for Tamil.

4.6 Morphological Analysis

We considered morphological features only when generating the Sinhala side of the parallel corpus. We used Helabasa - Noun Analyzer (2017) to retrieve Sinhala morphological features.

When training the translation model for each technique, we appended the synthetic corpus generated from that technique to our original corpus in one to one ratio and trained a separate model.

5. Results and Analysis

Table 3 provides examples resulting from each augmentation procedure.

Considering Table 3, in the initial data augmentation method (first row), substituting බඳවාගැනීමට (/banndhavaagAniemaTa/ - to hire) with දන්වන්නෙහිද (/danwannehida/ - will inform?) makes the resulting synthetic sentence meaningless. Sentences that are generated by analyzing POS tags seem to have an edge over initial data augmentation method. Since සැලැස්මට (/salasmataTa/ - for plan) and බඳවාගැනීමට (/banndhavaagAniemaTa/ - to hire) (second row) have identical POS tags, high fluency is achieved in the resulting synthetic sentence pair. Synthetic sentence pair generated using the method mentioned in 3.3, preserves the meaning to a better extent. Word ලදුපතක්(/ladupatak/ - receipt) and කබායක්(/kabaayak/ - coat) have indefinite singular – Nominative, Accusative, Auxiliary, and Locative morphological features in common.

Method	Example
3.1	Si: එතුමා මෙම සභාවට [දන්වන්නෙහිද / **බඳවාගැනීමට**]? (/etumaa mema sabhaavaTa [danwannehida / **banndhavaagAniemaTa**] ?/) Ta: அவர் இச்சபைக்குத் [தெரிவிப்பாரா/ *ஆட்சேர்ப்பிற்கு*] ? (/avar issapaikkudh [dherivippaaraa/ *aatseerppiRku*] /) (En: For this session, he [will inform/ for hiring]?)[2]
3.2	Si: පුහුණු [සැලැස්මට / **බඳවාගැනීමට**] අදාල තොරතුරු (/puhuNu [sAlAsmaTa / **banndhavaagAniemaTa**] adaala toraturu/) Ta: பயிற்சித் திட்டத்திற்கு [பொருத்தமான / *ஆட்சேர்ப்பிற்கு*] தகவல்கள் (/payiRsidh dhittadhdhiRku [porudhdhamaana / *aatseerppiRku*] dhakavalkaL/) (En: Information [related to training planning/ related to training hiring])[3]
3.3	Si: පහත සඳහන් ලිපිනයට කරුණාකර [ලදුපතක් / **කබායක්**] එවීමට කටයුතු කරන්න. (/pahata sanndhahan lipinayaTa karuNaakara [ladupatak / **kabaayak**]ewiemaTa kaTayutu karanna./) Ta: கீழ் காணும் முகவரிக்கு தயவு செய்து [பற்றுச் சீட்டொன்றை / *மழைக்காப்பு*]அனுப்ப நடவடிக்கை எடுக்கவும். (/kiiz kaaNum mukavarikku dhayavu seydhu [paRRus siittonRai / *mazaikkaappu*]anuppa watavatikkai etukkavum./) (En: Kindly send a [receipt/coat] for the following address)

Table 3: Examples synthetic data with highlighted [original / **substituted**] and [original /*translated*] words

Table 4 depicts the BLEU scores obtained for each method.

Method	Si-Ta	Ta-Si
Benchmark training	6.78	6.84
+ Initial data augmentation	7.68	8.86
+POS tagging	8.72	10.98
+Morphological Analysis	8.94	11.84

Table 4: BLEU scores

Synthetic data generated using the initial data augmentation method have improved the performance of Si-Ta and Ta-Si translation by 0.9 and 2.02 amounts, respectively. To verify that this gain is due to the rare word substitutions and not just due to the repetition of a part of the training data, we performed an experiment where each sentence pair selected for augmentation is added to the training data unchanged (i.e. without creating synthetic data). This simple form of sampled data replication delivered 0.53 and 1.42 BLEU score gains for Si-Ta and Ta-Si, respectively. Hence initial data augmentation models have performed better compared to simple data replication method.

Use of POS tags has achieved 1.04 and 2.12 BLEU score gains over the initial data augmentation for Si-Ta and Ta-Si, respectively. Human evaluators who oversaw the quality of generated sentences revealed that the use of POS tags has increased the fluency of language and rare word translation performance by a significant amount. Thus, we can empirically prove that the use of POS tags improves the quality of synthetic training data, which in turn reduces the rare word problem in NMT.

Morphological features have played a vital role in reducing the rare word problem. When generating synthetic sentence pairs, we considered only Sinhala language morphological features. Sinhala being morphologically rich, there exist many number of variations for a given root word. Hence checking the case-count combinations of a word when substituting, helps to preserve language semantics of the generated sentence. This is evident by analyzing the BLUE score gains of 1.26 and 2.98 for Si-Ta and Ta-Si translations compared to the initial data augmentation method.

Table 5 depicts an example of successful translation of a rare word.

Reference Source	விசேட பிரிவு(/*viseeta pirivu*/)
Reference Target	විශේෂ ඒකකය(/*viSeasha eakakaya*/) (Special unit)
Benchmark Model	විශේෂ අංශය(/*viSeasha a\nSaya*/) (Special sector) - erroneous
Our method (3.2 and 3.3)	විශේෂ ඒකකය(/*viSeashaeakakaya*/) (Special unit) - correct

Table 5: Rare Word example

To examine the impact of augmenting training data by creating contexts for rare words on the target side, we tested how each model performs on rare words. Most of the rare words are not 'rare' anymore in the augmented data since they were augmented sufficiently many times.

BLUE score gains are consistent across both translation directions, regardless of whether rare word substitutions are first applied to Sinhala or Tamil. Hence it can be verified that using POS tagging and morphological features results in generating quality synthetic parallel data that preserve language semantics, which eventually leads to better translation performance. Though overall rare word translation quality was improved by our methods, there were several cases where augmentation resulted in incorrect outputs that were correctly translated by our benchmark system. Table 6 corresponds to such an incorrect translation.

Our benchmark model has been able to correctly translate දේශීය (/*deaSieya*/ - local) and විදේශීය (/*videaSieya*/ - foreign) terms, whereas our new model has not been able to translate any of them. If the language model selects substitutions that have low probabilities, it results in generating outputs with low fluency. Another possible reason is errors in word alignments. If the word alignments are erroneous and phrase table contains faulty

[2]Exact English translation:"Will he inform this session?" / "For this session will he be hiring?" ("will inform" is replaced by "for hiring")

[3]Exact English translation:"Information related for training planning" / "Information related for training hiring"("for training" is replaced by "for hiring")

probabilities, this may lead to synthetic sentence pairs that do not correspond to each other.

Source Sentence	8. உள்ளூர்/ வெளிநாட்டு திரைப்ப ட தயாரிப்பாளர்களுடன் /uLLur/ veLiwaattu dhiraippa ta dhayaarippaaLarkaLutan/ - (With local and foreign film producers)
Reference Translation	8 .දේශීය / විදේශීය චිත්‍රපට නිෂ්පාදකයින් සමඟ /8 .deaSieya / videaSieya citrpaTa nishpaadakayin samannga / - (With local /foreign film producers)
Benchmark translation	8 .දේශීය විදේශීය විකාශන කටයුතු සඳහා/ 8 .deaSieya videaSieya vikaaSana kaTayutu sanndhahaa/ - (For local/ foreign broadcasting)
Our method (3.2 and 3.3)	(8)විදේශ චිත්‍රපට ගනුදෙනු කිරීම / (8)videaSa citrapaTa ganudenu kiriema . / - (For trading foreign films)

Table 6: Incorrect outputs

6. Conclusion

The purpose of this research was to find out syntax preserving techniques for synthetic data generation to solve the rare word problem in NMT for the under-resourced language pair Sinhala and Tamil. POS tagging and morphological analysis show impressive results in increasing the quality of synthetic sentence pairs that reduces the rare word problem. Being morphologically rich, there exist a number of morphological features in Sinhala and Tamil that can be exploited to enhance the quality of augmented data. We expect to experiment with these features in the future.

7. Acknowledgements

The authors would like to thank the reviewers for their helpful comments and suggestions. The authors are grateful to members of the National Languages Processing Centre at University of Moratuwa for their significant contribution in developing the basic linguistic resources, and the Department of official languages of Sri Lanka for providing corpus data needed to carry out the research.

8. Bibliographical References

Bahdanau, D., Cho, K., &Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. Arxiv Preprint Arxiv:1409.0473 [Cs.CL].

Cho, K., van Merrienboer, B., Bahdanau, D., &Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8) ,pp. 103-111.

Fadaee, M., Bisazza, A., &Monz, C. (2017). Data Augmentation for Low-Resource Neural Machine Translation. In Proceedings of the Annual Meeting of the Association for Computational Linguistics ,pp. 567-573.

Klein, G., Kim, Y., Deng, Y., Senellart, J., & M. Rush, A. (2017). OpenNMT: Open-source toolkit for neural machine translation",.Arxiv Preprint Arxiv:1701.02810 [Cs.CL],.

Och, F. J., & Ney, H. (2004). The alignment template approach to statistical machine translation. Computational linguistics, 30(4),pp. 417-449.

Papineni, K., Roukos, S., Ward, T., & Zhu, W. (2002). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics ,pp. 311-318.

Priyanga, R., Ranatunga, S., & Dias, G. An Inflectional Morphological Generator for Sinhala Nouns. (Unpublished)

Pushpananda, R., Weerasinghe, R., & Niranjan, M. (2014). Sinhala-Tamil Machine Translation: Towards better Translation Quality. In proceedings of Australasian Language Technology Association Workshop, 129, pp. 129-133.

Sennrich, R., Haddow, B., & Birch, A. (2016). Edinburgh Neural Machine Translation Systems for WMT 16. In Proceedings of the First Conference on Machine Translation (WMT) ,pp. 371-376. Association for Computational Linguistics.

Sennrich, R., Haddow, B., & Birch, A. (2016). Improving Neural Machine Translation Models with Monolingual Data. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 86-96.

Stolcke, A. (2002). SRILM-an extensible language modeling toolkit. In Interspeech. pp. 901-904.

Tennage, P., Sandaruwan, P., Thilakarathne, M., Herath, A., Ranathunga, S., Dias, G., &Jayasena, (2017). Neural Machine Translation for Sinhala and Tamil Languages, *(Submitted).*

9. Language Resource References

Computational Linguistic Research Group. (2017). Au-kbc.org. Retrieved 2 October 2017, from http://www.au-kbc.org/nlp/corpusrelease.html

Farhath, F., Ranathunga, S., Jayasena, S., Dias, G., &Thayasivam, U. (2017). Improving Domain-Specific Statistical Machine Translation for Sinhala-Tamil using Bilingual Lists, *(Submitted).*

Fernando, S., Ranathunga, S., Jayasena, S., & Dias, G. (2016). Comprehensive Part-Of-Speech Tag Set and SVM based POS Tagger for Sinhala. In Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (pp. 173 - 182).

Helabasa - Noun Analyzer. (2017). Translation.projects.mrt.ac.lk. Retrieved 1 October 2017, from http://translation.projects.mrt.ac.lk:8081/helabasa/noun_analyzer

BDPROTO: A Database of Phonological Inventories from Ancient and Reconstructed Languages

Egidio Marsico[1], Sebastien Flavier[1], Annemarie Verkerk[2], Steven Moran[3]
[1]Dynamique Du Langage, CNRS, Université de Lyon, Lyon, France
[2]Max Planck Institute for the Science of Human History, Jena, Germany
[3]Department of Comparative Linguistics, University of Zurich
{egidio.marsico, sebastien.flavier}@cnrs.fr, verkerk@shh.mpg.de, steven.moran@uzh.ch

Abstract

Here we present BDPROTO, a database comprised of phonological inventory data from 137 ancient and reconstructed languages. These data were extracted from historical linguistic reconstructions and brought together into a single unified, normalized, accessible, and Unicode-compliant language resource. This dataset is publicly available and we aim to engage language scientists doing research on language change and language evolution. We provide a short case study to highlight BDPROTO's research viability; using phylogenetic comparative methods and high-resolution language family trees, we investigate whether consonantal and vocalic systems differ in their rates of change over the last 10,000 years.

Keywords: historical linguistics, phonological inventories, rates of change, language evolution

1. Overview

First we provide some background on ancient language reconstruction and why it is interesting in light of studying the evolution of human language. Then we describe the BDPROTO language sample and our data extraction and aggregation pipelines. Afterwards we present a short case study using ancient language data from BDPROTO together with phonological inventory data from currently spoken languages to investigate evolutionary trends in consonant and vowel systems. Finally, we discuss how consonant and vowel inventories have changed over the last 10,000 years and we detail avenues for further research in historical and evolutionary linguistics.

2. Background

The development of the comparative method in the 19th century is one of the hallmarks of modern linguistics. It allows language scientists to reliably reconstruct ancient languages from prehistory in terms of form and meaning, including their vocabulary, phonology, grammar, and even their speakers' cultural practices. The comparative method is used to demonstrate genealogical relationships between languages and to reconstruct a proto-language, i.e. the common ancestor of a set of languages before their divergence. In short, when applying the comparative method technique on modern languages, linguists perform a feature-by-feature comparison of two or more languages that share a common ancestor, so that they can extrapolate the properties of their linguistic ancestor (the so-called parent language).

Comparative reconstruction is accomplished through systematic comparison of word forms in descendant languages. By identifying groups of potentially related words, cognates between related languages are identified (words that share form and meaning and are genealogically related). Consider the modern word for 'tooth' in four Romance languages: Spanish 'diente', Italian 'dente', French 'dent', and Portuguese 'dente'. The similarities in these words could be due to chance correspondence, word borrowing, or linguistic universals, cf. Blasi et al. (2016). However, these three factors are highly unlikely in this example because each descendent form shares the same meaning and a similar phonetic form. That is, each word begins with a 'd' sound and it contains a consonant cluster 'nt'. Thus it is highly probable that the ancestral word contained a form resembling 'd ... nt ...' (Fortson, 2004, 3).

Using the principle of maximum parsimony, the proto-language word form contained simply an 'e', instead of the diphthong 'ie', as in Spanish today. Additionally, it is not uncommon in the world's languages to drop sounds at the end of words, particularly vowels, as was probably done in French. Therefore the proto-form for the word 'tooth' shared by these Romance languages had the shape 'dente', written *dente to denote a reconstructed form. Sometimes such forms are preserved in historical records (in which they have been used to verify the accuracy of the comparative method), but more often they are hypothetical reconstructions. With a large enough amount of reconstructed vocabulary, language scientists can posit that the parent language of modern languages, in this case so-called Proto-Romance, contained a 'd' sound in its phonological inventory, i.e. its repertoire of contrastive speech sounds.

Recently the comparative reconstruction approach outlined above has been implemented programmatically (Steiner et al., 2011), so that many of the time-consuming and redundant tasks of the historical linguist are automated, for example inferring regular sound change (Bouchard-Côté et al., 2013; Hruschka et al., 2015). The resulting score of similarity from pairwise sets of words across all languages in a sample can help to identify cognates. Expert judgment is still needed, but tools (List and Moran, 2013) and interfaces (List, 2017) allow even the non-tech-savvy linguist to quickly identify cognates from masses of raw data, such as word lists from thousands of languages (Wichmann et al., 2017). Word lists that are coded for cognacy and phonetic similarity scores can be used as input for one of many al-

gorithms that generate language family phylogenies. These language family trees can then be used as input to phylogenetic comparative methods developed by biologists for investigating the tree of life, but adopted and adapted by linguists and evolutionary anthropologists to address research questions about ancient language structures, cultures and population movements, e.g. Dunn et al. (2011), Gray et al. (2009), Bouckaert et al. (2012).

3. Data extraction and aggregation

The phonological inventories in BDPROTO were extracted manually from source texts,[1] interpreted by experts, and then codified according to standardized Unicode conventions (Moran and Cysouw, In press) for the International Phonetic Alphabet (International Phonetic Association, 1999). The resulting dataset was put into a Github repository, additional metadata were added, and an aggregation script was written to bring three independent and disparate input data sources together.[2]

The datasets include the original BDPROTO data from Marsico (1999) and reconstruction data collected more recently at the Department of Comparative Linguistics at the University of Zurich. The former was originally stored in SQL tables in ISO 8859-1 encoding and was for this work transformed into CSV files in UTF-8 NFC with LF and no BOM. Given the legacy character encoding, we standardized character representations using the PHOIBLE conventions.[3] Additional phonological inventories were entered by hand into Excel spreadsheets and exported as compliant Unicode Standard UTF-8 to complement and extend the sample in Marsico (1999). For each of the 478 unique speech sounds reported in BDPROTO, we integrated a distinctive feature vector from the 37 phonetic features described in PHOIBLE (Moran et al., 2014).

Supplemental metadata for each inventory was collected and is stored in the BDPROTO repository, including for each language: estimates for its age and the homeland where it was spoken. Both the time depth and the homeland of language families are hotly debated issues; see for example the discussion of the age and heartland of Indo-European (Bouckaert et al., 2012; Chang et al., 2015). Each language data point in BDPROTO is also associated with a Glottolog language identifier, so that it is positioned within a language family phylogeny (Hammarström et al., 2017).[4] Each inventory has one or more bibliographic citations, which are stored in a text-based BibTeX file, where the BDPROTO ID is mapped to the BibTeX key for easy perusal of original data sources.

The aggregation of the phonological inventory data and metadata is accomplished with a script written in the R programming language (R Core Team, 2013). This script combines the inventory data from CSV files, joins in the additional linguistic and non-linguistic metadata described above, and outputs the combined data sources as an R data object and CSV files.

[1] https://github.com/bdproto/raw-data/metadata/bdproto-references.bib

[2] https://github.com/bdproto

[3] http://phoible.github.io/conventions/

[4] http://glottolog.org/

4. The language sample

There are 137 phonological inventories in the current BDPROTO sample, which represent 126 distinct reconstructed and ancient languages from 67 different language families. The original BDPROTO sample was devised without duplicates by considering the coherence of the proposed reconstructions and their relations to their modern daughter languages. The aggregation of the original BDPROTO sample with our more recent work of collecting inventories results in duplicate data points. We consider multiple entries a feature of our database, thus allowing the user to explore and compare different reconstructions by different experts.

Figure 1 lists 25 of the oldest languages in the sample and approximately when they were spoken. Some data points in BDPROTO represent root-level language family nodes, such as Indo-European. Other data points in BDPROTO are intermediate nodes in existing proposed phylogenies. For example, there are expert reconstructions of ancient Germanic, Nordic, and Anatolian, each of which represents an intermediate node within the branches of Indo-European tree, i.e. daughter languages of Indo-European but also parent languages of currently spoken languages. Note that it is generally agreed-upon that 10,000 years is the maximum time depth of reconstruction for the comparative method (Nichols, 1992). Past this time depth, languages have simply had too much time to mutate in vocabulary through regular processes of sound change and it has not yet been discovered how to peer further back in time (although this is an active area of research, e.g. Pagel et al. (2013)). The fact that most language families have resided in geographically disparate areas and have been influenced by many other factors, including linguistic and cultural, is not beneficial for deep reconstruction.

5. Case study: consonant vs vowel rates

In a study of whether phonological inventories have become more or less complex over time, Marsico (1999) shows that languages dating back as far as 10,000 years are equally-complex in terms of their number of segments, consonant/vowel ratio, average number of consonants and vowels, and frequency hierarchy of the segments. However, Marsico (1999) also notes that modern languages tend to have slightly more consonants today than their ancestors did in the past. The same does not apply to vowels. On average the number of consonants and vowels across proto-languages in BDPROTO are 18 and 8, respectively. In comparison, modern spoken languages have on average 22 consonants and 8 vowels (Maddieson, 1984).[5]

Why is it that we observe more consonants in phonological inventories today than we see in reconstructed ancient languages of the past? We decided to test whether six language families show greater rates of change in consonant inventory size as compared to vowel inventory size using phylogenetic comparative methods. Specifically, we use BayesTraits V2 (Meade and Pagel, 2014), which implements a generalized least squares approach to modeling the evolution of continuously varying traits (Pagel,

[5] Note that these averages are not adjusted for phylogeny because so far there is a lack of high-resolution language phylogenies for most language families.

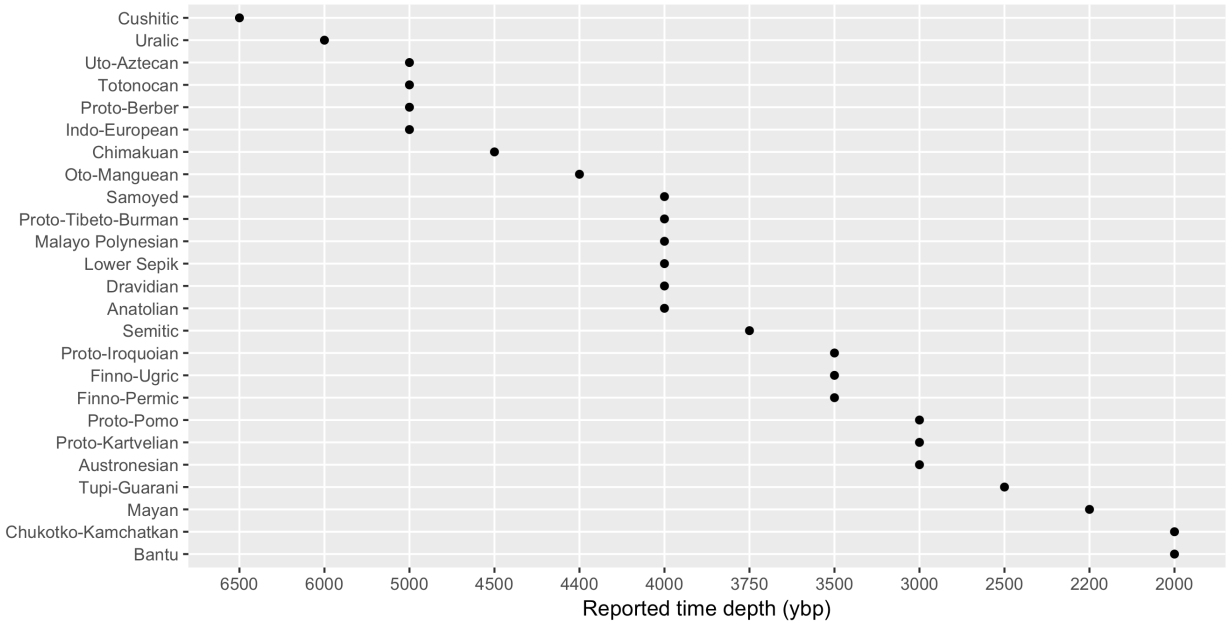


Figure 1: Approximate age of 25 language families in the BDPROTO sample

1997; Pagel, 1999). We chose these six language families because they have high-resolution expert-created phylogenies: Arawakan (language sample n=38; Walker and Ribeiro (2011)), Austronesian (83; Gray et al. (2009)), Bantu (114; Grollemund et al. (2015)), Indo-European (58; Bouckaert et al. (2012)), Pama-Nyungan (134; Bowern and Atkinson (2012)), and Tupi-Guarani (30; Michael et al. (2015)) and because they are in the BDPROTO sample. Figure 2 shows box plots of the ranges of vowel and consonant inventory size in the language samples used for phylogenetic ancestral state estimations. P gives proto-language reconstruction from BDPROTO. R gives the ancestral state estimation. An asterisk * indicates whether the rate of change of vowel or consonant inventory size is faster.[6]

Language family	Consonants	Vowels
Tupi-Guarani	5.02 ± 0.60	4.07 ± 0.52
Pama-Nyungan	3.28 ± 0.18	0.74 ± 0.04
Arawakan	17.57 ± 1.51	31.41 ± 2.76
Austronesian	129.07 ± 3.85	43.67 ± 1.22
Bantu	63.98 ± 2.94	8.52 ± 0.23
Indo-European	29.21 ± 2.42	47.00 ± 3.88

Table 1: Rates of change in 1000s of years

Table 1 gives the mean rates of change of consonant and vowels on the branches of the listed phylogenetic tree sets by 1000s of years. Our results suggest a mixed picture for the acquisition of new consonants vs vowels over the last 10,000 years. In two-thirds of the language families sampled, the rate of change in consonants systems is greater. But in Arawakan and Indo-European vowel inventory size changes faster than consonant inventory size. These two language families have in common a wider range of vowel inventory sizes as compared to the other families. However, if we take into account the mean and standard deviations of the rates given in Table 1, a high variance does not always entail a high rate and vice versa. Austronesian, for instance, has the highest rate of change for consonants, but the Austronesian languages are less variable in their inventory size than Bantu and Indo-European.

Thus our results suggest differential rates of change in consonants and vowels by language family. This finding is surprising to us because the synchronic data suggest that there is a diachronic pressure on languages to expand their consonant inventories at a greater rate than vowels; in line with the finding by Marsico (1999). For example, on average languages have more consonants than vowels, so we might expect phonological inventories to universally acquire consonants at a faster rate. Consonants are more likely to be borrowed than vowels (Moran et al., 2014). The synchronic data also show more phonetic diversity in consonant inventories, suggesting a greater number of lexical contrasts available by consonants. For example, there are three times as many contrastive consonants than vowels in the world's languages. Consonant inventories also range more in size from 6–90 (Rotokas in Papua New Guinea vs the click language !Xu, spoken in Botswana and Namibia) and vowel qualities from 2–14 (Maddieson, 2013a; Maddieson, 2013b).

Our finding warrants further research, but we might already speculate on where to look next. Inventories of both vow-

[6]We also observe that ancestral state estimates of vowel and consonant inventory sizes are generally closer to the mean of the range than expert reconstructions of proto-languages. This means there is a difference between the well-worked historical comparative method used by linguists to reconstruct proto-languages and the automated ancestral rates generated through pyhylogenetic analysis. This observation warrants a closer evaluation using directional models of feature change.

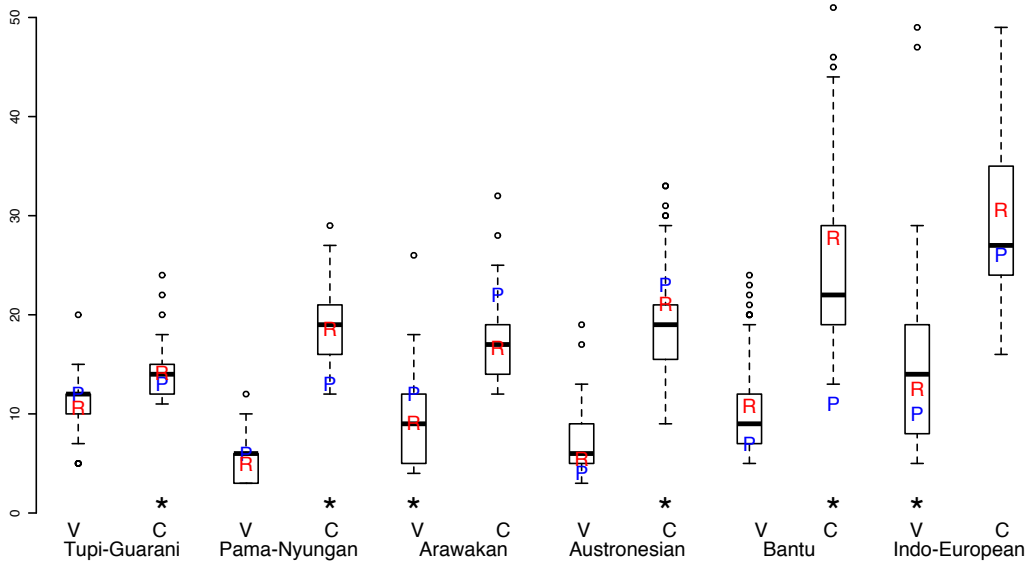


Figure 2: Ranges, reconstructions, and ancestral state estimations of vowel and consonant inventory size

els and consonants can be extended through the use of secondary articulatory features. For example, a vowel space can be expanded straightforwardly by the contrastive features length and nasalization. On the other hand, length and labialization, palatalization, and velarization, can expand consonant inventories.[7] It may be the case that mathematically there is a greater number of dimensions for consonant inventories to expand, but that there are other constraints on how consonant or vowel inventories increase in size. Hence to create more and more vocabulary, increasing the number of contrastive sounds in the phonological inventory while keeping the number of distinctive phonetic features at a minimum is said to encompass the principle of feature economy (Clements, 2009). An example is given in Moran (2012), who shows that vowel systems tend to expand from the cardinal vowels through the highly economic features length and nasalization before filling in the vowel space with peripheral vowels that require finer articulatory features and distinctions. Furthermore, Coupé et al. (2011) show that there is an asymmetry between feature economy in which vowel inventories tend to be more economical than consonant inventories. Thus the articulatory and perceptual constraints that may govern the changes in phonological inventories over time must be incorporated into models of the evolution of spoken languages.

6. Summary

Here we present BDPROTO, an open-access database of phonological inventories from a sample of 137 ancient and reconstructed languages. BDPROTO provides a rich resource for investigating historically reconstructed languages and whether they show any significant changes with languages spoken today. After an initial brief overview of the historical comparative method, we describe the data extraction and aggregation pipelines that we used to create the BDPROTO database. Finally, in a short case study we use phylogenetic methods to show that the evolution of consonant and vowel systems have differential rates of change – an unexpected observation given what we know about the ancient and reconstructed languages in the BDPROTO sample and their modern descendants.

[7] Consider for example palatalization in Russian, which increases the number of possible lexical contrasts in Russian, while being as perceptually salient a feature as primary features like voicing (Kavitskaya, 2006).

7. Acknowledgments

Special thanks to Ian Maddieson for his help with the data analysis and standardization. We also thank Balthasar Bickel, Joël Brogniart, Paul Widmer, and three anonymous reviewers.

8. Author contributions

EM, SF, SM designed the research. EM, SM collected the data. EM, SF, SM designed the database. AV, SM designed and implemented the case study. SM, AV wrote the paper.

9. Bibliographical References

Blasi, D. E., Wichmann, S., Hammarström, H., Stadler, P. F., and Christiansen, M. H. (2016). Sound–meaning association biases evidenced across thousands of languages. *Proceedings of the National Academy of Sciences*, page 201605782.

Bouchard-Côté, A., Hall, D., Griffiths, T. L., and Klein, D. (2013). Automated reconstruction of ancient languages using probabilistic models of sound change. *Proceedings of the National Academy of Sciences*, 110(11):4224–4229.

Bouckaert, R., Lemey, P., Dunn, M., Greenhill, S. J., Alekseyenko, A. V., Drummond, A. J., Gray, R. D., Suchard, M. A., and Atkinson, Q. D. (2012). Mapping the origins and expansion of the Indo-European language family. *Science*, 337(6097):957–960.

Bowern, C. and Atkinson, Q. D. (2012). Computational phylogenetics and the internal structure of Pama-Nyungan. *Language*, 88:817–845.

Chang, W., Cathcart, C., Hall, D., and Garrett, A. (2015). Ancestry-constrained phylogenetic analysis supports the Indo-European steppe hypothesis. *Language*, 91(1):194–244.

Clements, G. N. (2009). The Role of Features in Phonological Inventories. In Eric Raimy et al., editors, *Contemporary Views on Architecture and Representations in Phonology*, pages 19–68. MIT Press.

Coupé, C., Marsico, E., and Philippson, G. (2011). How economical are phonological inventories? In *Proceedings of the 17th International Congress of Phonetic Sciences*.

Dunn, M., Greenhill, S. J., Levinson, S. C., and Gray, R. D. (2011). Evolved structure of language shows lineage-specific trends in word-order universals. *Nature*, 473(7345):79–82.

Fortson, B. W. (2004). *Indo-European Language and Culture: An Introduction*. Number 13 in Blackwell Textbooks in Linguistics. Blackwell, Oxford.

Gray, R. D., Drummond, A. J., and Greenhill, S. J. (2009). Language phylogenies reveal expansion pulses and pauses in Pacific settlement. *Science*, 323(5913):479–483.

Grollemund, R., Branford, S., Bostoen, K., Meade, A., Venditti, C., and Pagel, M. (2015). Bantu expansion shows that habitat alters the route and pace of human dispersals. *Proceedings of the National Academy of Sciences*, 112(43):13296–13301.

Hammarström, H., Forkel, R., and Haspelmath, M. (2017). Glottolog 3.0. Online: `http://glottolog.org`.

Hruschka, D. J., Branford, S., Smith, E. D., Wilkins, J., Meade, A., Pagel, M., and Bhattacharya, T. (2015). Detecting regular sound changes in linguistics as events of concerted evolution. *Current Biology*, 25(1):1–9.

International Phonetic Association. (1999). *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet*. Cambridge University Press.

Kavitskaya, D. (2006). Perceptual salience and palatalization in russian. *Laboratory Phonology*, 8:589–610.

List, J.-M. and Moran, S. (2013). An open source toolkit for quantitative historical linguistics. In *ACL (Conference System Demonstrations)*, pages 13–18.

List, J.-M. (2017). A Web-based interactive tool for creating, inspecting, editing, and publishing etymological datasets. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. System Demonstrations*, pages 9–12, Valencia. Association for Computational Linguistics.

Maddieson, I. (1984). *Pattern of Sounds*. Cambridge University Press, Cambridge, UK.

Maddieson, I. (2013a). Consonant inventories. In Matthew S. Dryer et al., editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Maddieson, I. (2013b). Vowel quality inventories. In Matthew S. Dryer et al., editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Marsico, E. (1999). What can a database of proto-languages tell us about the last 10,000 years of sound changes? In *Proceedings of the XIVth International Congress of Phonetic Sciences*, pages 353–356, San Francisco.

Meade, A. and Pagel, M. (2014). Bayestraits v2. Online: `http://www.evolution.rdg.ac.uk/BayesTraitsV3/BayesTraitsV3.html`.

Michael, L., Chousou-Polydouri, N., Keith, B., Donnelly, E., Meira, S., Wauters, V., and O'Hagan, Z. (2015). A bayesian phylogenetic classification of Tupí-Guaraní. *LIAMES - Línguas Indígenas Americanas*, 15:193–221.

Moran, S. and Cysouw, M. (In press). *The Unicode Cookbook for Linguists*. Language Science Press.

Moran, S., McCloy, D., and Wright, R. (2014). *PHOIBLE Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Moran, S. (2012). *Phonetics Information Base and Lexicon*. Ph.D. thesis, University of Washington.

Nichols, J. (1992). *Linguistic diversity in space and time*. University of Chicago Press.

Pagel, M., Atkinson, Q. D., Calude, A. S., and Meade, A. (2013). Ultraconserved words point to deep language ancestry across Eurasia. *Proceedings of the National Academy of Sciences*, 110(21):8471–8476.

Pagel, M. (1997). Inferring evolutionary processes from phylogenies. *Zoologica Scripta*, 26(4):331–348.

Pagel, M. (1999). Inferring the historical patterns of biological evolution. *Nature*, 401(6756):877.

R Core Team, (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Steiner, L., Stadler, P. F., and Cysouw, M. (2011). A pipeline for computational historical linguistics. *Language Dynamics and Change*, 1(1):89–127.

Walker, R. S. and Ribeiro, L. A. (2011). Bayesian phylogeography of the Arawak expansion in lowland South America. *Proceedings of the Royal Society of London B: Biological Sciences*, 278(1718):2562–2567.

Wichmann, S., Holman, E. W., and Brown, C. H. (2017). The ASJP Database (version 17). Online: `http://asjp.clld.org/`.

Creating a Translation Matrix of the Bible's Names Across 591 Languages

Winston Wu[1]*, Nidhi Vyas[2]*, David Yarowsky[1]
[1]Johns Hopkins University, [2]Carnegie Mellon University
{wswu,yarowsky}@jhu.edu, nkvyas@cs.cmu.edu

Abstract

For many of the world's languages, the Bible is the only significant bilingual, or even monolingual, text, making it a unique training resource for tasks such as translation, named entity analysis, and transliteration. Given the Bible's small size, however, the output of standard word alignment tools can be extremely noisy, making downstream tasks difficult. In this work, we develop and release a novel resource of 1129 aligned Bible person and place names across 591 languages, which was constructed and improved using several approaches including weighted edit distance, machine-translation-based transliteration models, and affixal induction and transformation models. Our models outperform a widely used word aligner on 97% of test words, showing the particular efficacy of our approach on the impactful task of broadly multilingual named-entity alignment and translation across a remarkably large number of world languages. We further illustrate the utility of our translation matrix for the multilingual learning of name-related affixes and their semantics as well as transliteration of named entities.

Keywords: Bible, alignment, named entities, translation, transliteration

1. Introduction

In a statistical machine translation (SMT) pipeline, word alignment is important for extracting phrase translations. However, for low-resource languages with very little data, these alignments may be extremely noisy or may not exist at all. Thus, improving the quality of word alignments leads to more accurate phrase pairs, which in turn improves the quality of an SMT system (Och and Ney, 2003; Fraser and Marcu, 2006). For many low-resource languages, the Bible is the only text available, making it a valuable resource to train machine translation (MT) systems. This paper focuses on the translation and transliteration of named entities from the Bible, which are a rich resource for studying lexical borrowing (Tsvetkov and Dyer, 2016), since they are usually borrowed between languages rather than translated[1](Whitney, 1881; Moravcsik, 1978; Myers-Scotton, 2002).

Like cognates, names are often phonetically or orthographically similar across languages, which make them suited for training transliteration systems, in contrast to words which may just be translations of each other. In addition, especially for low-resource languages, names can be an invaluable source of information for learning morphemes and their semantics. However, finding their optimal translation is a challenging task, due to various reasons, including low occurrence counts and because certain names have high translation entropy or are translated into their localized proper names. In this work, we present: (1) our creation of a novel resource of 1129 English Bible named entities aligned across 591 languages; (2) novel methods to produce and clean the resource; (3) challenges and findings in this process; and (4) potential use cases of our resource. Our Bible names translation matrix is available at `github.com/wswu/trabina`. We believe this resource will be of great linguistic importance in studying low resource languages and will be applicable in several areas such as transliteration and morphological analysis of named entities.

*Denotes equal contribution.

[1]The opposite case is also very interesting, e.g. in many languages, *Caesar* is translated as *emperor* rather than transliterated.

2. Related Work

Due to low frequency words, word alignments can suffer from misalignments, which in turn can be detrimental to downstream tasks like MT. Previously, researchers have worked on improving word alignment to improve MT using a variety of approaches: combining hypotheses generated from bridge languages (Kumar et al., 2007), using semantic relationships (Songyot and Chiang, 2014), prior distributions (Mermer and Saraclar, 2011; Vaswani et al., 2012), discriminative alignment models (Moore, 2005; Taskar et al., 2005; Riesa and Marcu, 2010) and part-of-speech (POS) tagging (Lee et al., 2006; Sanchis and Sánchez, 2008). Our work uses the assumption that names are often orthographically or phonetically similar across languages. We use MT as an intermediate step to generate hypotheses for citation form alignments. This is akin to training phrase-based MT systems for transliteration (Song et al., 2009; Jia et al., 2009; Dasigi and Diab, 2011). (Dasigi and Diab, 2011) used a character-based Moses system and post-edited the output using linguistic rules, which is similar to our approach of using MT and applying transformation rules. Our approach is unique in that we use cross-language joint models of variant/latent forms to expand and refine the candidate space of citation forms.

3. Translation Matrix

The primary contribution of this paper is a translation matrix of 1129 English names aligned and translated into 591 languages. We produced around 14,000 alignments, providing better coverage than Wikipedia for several names.[2] On average, a name contains realizations in 52% of all languages.[3] The names in this matrix were extracted from the parallel Bibles corpus (Mayer and Cysouw, 2014), in which verses

[2]For example, in Wikipedia, *Egypt* is translated (including Romanization) into 55 languages, whereas our resource contains translations of Egypt in the majority of 591 languages.

[3]Names that occur frequently, like *Jesus*, are covered in almost all 591 languages, while uncommon names or name variants, like *Antiochia* (variant of *Antioch*), appear in fewer languages.

English	Yipma	Hanga	Koongo	Hanunoo	cnw*	Maltese	Latvian	gug*
jesus	jizaazai	yeesu	yesu	hisus	jesuh	ġesú	jēzus	jesús
christ	kɨraazɨ	kristu	klisto	kiristu	khrih	kristu	kristus	cristo
israel	yɨzɨrelɨ	juusi	isaeli	israil	israel	iżrael	izraēli	israel
david	devitɨ	dawuda	davidi	dabid	david	david	dāvida	david
paul	polɨ	pool	pawulu	pablu	paul	pawlu	pāvils	pablo
peter	pitai	piita	petelo	pidru	piter	pietru	pēteri	pedro
egypt	yɨzɨpɨ	yijipi	ngipiti	ihiptu	izip	eġittu	ēġipti	egípto
jerusalem	jeruzaalemɨ	jiruusilim	yelusalemi	hirusalim	jerusalem	ġerusalemm	jeruzalemē	jerusalén

Table 1: Example translation matrix of Bible named entities (*cnw = Chin Ngwan, *gug = Paraguayan Guaraní)

are aligned across all languages. Each cell in the translation matrix contains the best guess of the citation form of the English name in the target language. This form is the consensus of four different methods, which are described in the following sections. An excerpt of the name translation matrix is shown in Table 1.

4. Improving Named Entity Alignment

The source data from Mayer and Cysouw (2014) contains 24 English editions of the Bible. For 591 target language bibles, we word aligned each verse with each English Bible verse using the Berkeley Aligner (Liang et al., 2006) and performed POS tagging to extracted proper nouns from these alignments. A total of 1129 English named entities were extracted. For each English name, we considered multiple citation hypotheses in all target languages from the following four approaches.

4.1. Most frequent alignment

For every target language, the baseline hypothesis for a name's translation is the most frequent alignment obtained from the aligner. These initial hypotheses contained several alignment problems which we broadly classify into three categories: (1) incorrect alignment, (2) missing alignment, and (3) non-base form alignment.[4] Examples of these alignment errors are presented in Table 2. We improve on these initial alignments by tackling each of these problems in turn.

4.2. Distance-based approach

Incorrect alignments (Issue 1) are obviously problematic. Using the assumption that names are borrowed very frequently across languages and undergo minimal orthographic change, we employed an edit distance based approach to produce a hypothesis. If we let A be the top 5 most frequent alignments within a language, and T be the top 5 most frequent alignments combined across all languages,[5] the distance-based approach selects the name $h_{dist} \in A$ with the smallest Levenshtein distance (Levenshtein, 1966) to any word in T. This approach resolved many incorrect alignments.

For example, in the Manikion bible, the English name *Boas* was aligned 21 times to four words: Obed (9), Boas (7), eici (4), and Nahson (1). Clearly, the original alignment (Obed) is not the best translation. Taking the top 5 most common alignments for *Boas* across all languages (Booz, Boas, Boaz, —[6], and Boasi), we calculated the edit distance between each pair of words. The Manikion name with the minimum distance to any of the language-independent names (Boas) is a better translation than the original baseline alignment (Obed).

	Booz	Boas	Boaz	–	Boasi
Obed	4	4	4	4	5
Boas	2	**0**	1	4	1
eici	4	4	4	4	4
Nahson	5	5	6	5	5

4.3. Producing hypotheses with MT

Missing alignments (Issue 2) are common for low-frequency words, due to a weak alignment signal, or in certain languages, because these words may not exist at all in the Bible[7]. Such issues cannot be overcome using the distance-based approach, which selects a hypothesis from existing alignments. Thus, we employ character-based machine translation to suggest possible foreign names to which an English name is aligned.

Using the translation matrix with the distance-based hypotheses as bitext, we generate multiple hypotheses for plausible translations/transliterations by training ten MT systems for a single target language. The source languages were: four pivot languages (English, French, Spanish, Italian) and the six nearest languages in an ordering based on the language tree in Ethnologue. The pivot languages were chosen for their near-complete coverage over the 1129 names, and we utilized the six nearest languages due to the potential for names to be orthographically similar in related languages.

For each language pair, treating foreign-English name pairs as bitext, we split the data in half and trained two systems A and B, such that system A decodes the data that system B was trained on, and vice versa; this was done to ensure that the test set was never seen by the system performing decoding. We used a standard Moses (Koehn et al., 2007) setup with 5-gram KenLM (Heafield, 2011) language model and MERT (Och, 2003) for tuning. Each system generated a unique 200-best list of hypotheses.

[4] Throughout this paper, *italicized* names in tables refer to incorrect alignments/hypotheses.

[5] These five names represent a language-independent consensus; the intuition is that regardless of the language, the best translation of a name should be similar if not equal to one of these top five. We selected a threshold of five names because we found that this covered the major realization variants of a name.

[6] Denotes missing alignment, which in this case is frequent enough to make it into the top 5.

[7] For example, a language may not have a translation of the Old Testament, so names appearing only in the Old Testament will have missing alignments.

(a) Issue 1: Incorrect alignments

English	Foreign	Lang
Boaz	*Obedarinchichitam*	cbu
Boaz	*Obed*	mnx
Obed	*Jeseyrinchichitam*	cbu
David	*Luwiy*	agu
Eliezer	*Yesua*	mnx
Julia	*Pirorogasomi*	aak

(b) Issue 2: Missing alignments

English	For.	Lang
Mnason	—	cbu
Phoenix	—	cbu
Dionysius	—	mnx
Illyricum	—	mnx
Ephphatha	—	aak
Colossae	—	aak

(c) Issue 3: Non-lemma alignments

English	Foreign	Lang
David	*Dapiyarin*	cbu
Eliezer	*Eriesaori*	aak
Eliezer	*Elíyaserarinchichitam*	cbu
Aram	*Ramaho*	mcq

Table 2: MLE alignment problems: cbu = Candoshi, mnx = Manikion, aak = Muak Sa-aak, mcq = Ese, agu = Awakateko

4.3.1. Filtering and scoring hypotheses

The hypotheses generated by the MT approach were not necessarily valid names. To rectify this, we combined the n-best lists of all ten systems, filtered out hypotheses that did not occur in the target language's bible, and rescored the remaining hypotheses with a weighted combination of four features:

Fraction of observed to total count: The observed verse count of a hypothesis is a measure of how often it is aligned to the corresponding English name. It was calculated as follows: for each Bible verse where the English name occurred, we looked into the corresponding foreign Bible verse and within a window of ± 3 verses[8], and incremented a count if the hypothesis was present. The total count is the number of times the hypothesis appeared in the entire foreign Bible.

Model score: This feature incorporates the MT decoder's score, which we observed to generally fall within the range -5 to 5. We normalized it to bring it into the range [0, 1].

Closeness to expected English count: This feature gave weight to the actual coverage of a hypothesis within a given Bible. For a good hypothesis, we expect its observed verse count c_v to be close to the corresponding expected count in English. Since there are 24 English bibles, the expected English count c_e is the total English count $\div$ 24. We defined closeness as $\frac{-(c_v - c_e)^2}{c_e^2}$ if $c_v < c_e$ and c_e/c_v otherwise. The rationale for using a piece-wise function is that if the observed verse count is less than the expected count, then it might be due to inflected forms being aligned to the word. However, if the observed verse count is higher than the expected count, then it is likely not the correct alignment.

Matches the gold: 1 or 0 if the hypothesis matches the gold name h_{dist}.

We manually tuned the weights to $[0.4, 0.4, 0.05, 0.05]$ for the above features, respectively. Higher scores indicate better hypotheses. Unlike the baseline aligner and distance-based approaches, the MT approach can generate multiple hypotheses for a single English name and can produce plausible translations for missing alignments (see Table 3).

4.4. Transformation rules

Non-base form alignments (Issue 3) are common and occur when an English name, which does not mark case, is aligned to an inflected foreign name. String transformation rules were employed to recover the best candidate h_{tr} for the citation form of the English name in the target language from a set of candidate hypotheses generated by the previous approaches (Sections 4.1. to 4.3.).

[8]This compensated for possible incorrect verse alignments in the data that we started with.

English	Aligner	MT Hypotheses	Lang
Ephphatha	—	Epata	aak
Colossae	—	Korosi	aak
Colossae	—	Kolose	mnx
Dionysius	—	Dionisius	mnx
Illyricum	—	Ilirikum	mnx
Mahalaleel	—	Mahalelel	mnx
Dalmatia	—	*Tármatiyap*	cbu
Phoenix	—	Finíase	cbu
Sergius	—	Sírjiyu	cbu
Mnason	—	*Nasónap*; Nasón	cbu

Table 3: Hypotheses generated by MT approach, filling in missing alignments. aak = Muak Sa-aak, cbu = Candoshi, mnx = Manikion

Pairs in C-Set	LCS	T-Rule
Yose, Yose	Yose	$\emptyset \leftrightarrow \emptyset$
Yose, Yose'nin	Yose	$\emptyset \leftrightarrow$ 'nin
Yose, Yusuf	Y-s-	o-e $\leftrightarrow$ u-uf
Yose'nin, Yusuf	Y-s-	o-e'nin $\leftrightarrow$ u-uf
Beytanya, Beytanya	Beytanya	$\emptyset \leftrightarrow \emptyset$
Beytanya, Beytanya'ya	Beytanya	$\emptyset \leftrightarrow$ 'ya
Beytanya, Beytanya'dan	Beytanya	$\emptyset \leftrightarrow$ 'dan
Beytanya'ya, Beytanya'dan	Beytanya	'ya $\leftrightarrow$ 'dan

Affix	Freq
$\emptyset$	7
'dan	2
'ya	2
-u-uf	2
'nin	1
o-e	1
o-e'nin	1

Table 4: Transformation rules extracted for Turkish hypotheses of the English names *Joses* and *Bethany*, and their affix frequency.

Candidate hypotheses set (C-Set) This set consists of a total of 6 hypotheses (or less if any hypothesis from the previous iterations was missing):

1. Baseline alignment
2. Hypothesis from distance-based approach
3. Most frequent 1-best hypothesis from each MT system
4. Most frequent hypothesis from combined hypotheses
5. Best scoring 1-best hypothesis from each system

English	Aligner	T-Rule	Lang
Boaz	*Bowasom~~i~~*	Bowaso	aak
Eliezer	*Eriesaor~~i~~n~~i~~*	Ereasao	aak
Esau	*Isom~~i~~*	Iso	aak
Israel	*Is~~i~~rer~~i~~yí*	*Is~~i~~rene*	aak
David	*Dapiyarini*	Tapít	cbu
Eliezer	*Elíyaserarinchichitam*	Elíyas	cbu
Rama	*Aramho*	Ramo	mcq

Table 5: String transformation rules recover the lemma form of inflected alignments. mcq = Ese

6. 2nd best scoring 1-best hypothesis from each system

Transformation Rules (T-Rule) We define T-Rules as language-specific string transformation rules that change one word form to another. For example, the T-Rule $\{\emptyset \leftrightarrow s\}$ can change a singular to a plural word in English (e.g. *Cat* $\leftrightarrow$ *Cats*). For an English name's C-Set, we construct a T-Rule for every pair of words in the C-Set by removing the longest common subsequence between them (see Table 5). After obtaining rules for all English names in a given target language, we combined and sorted them by their frequency (Table 4). The hypothesis in the C-Set with the maximum underlying transformation frequency in the combined list of affixes was selected as the best translation hypothesis h_{tr}. For example (Section 4.4. and table 4), the empty affix is the most frequent affix, so this approach takes this to be the affix for the citation form of a name and thus selects *Yose* and *Beytanya* as the new base translations for Joses and Bethany, respectively. This method leverages the globally distributed information of word transformations in the target language to select the citation form. Table 5 shows examples of such cases.

5. Evaluation

To evaluate each alignment-improving approach, we first acquired a manually annotated test set of 30 randomly-chosen names across 591 languages, for a total of 17,730 examples. Annotators were asked to verify that the foreign best was indeed the best translation of that name. If not, they were to replace it with either a name from the alternatives (hypotheses from the various methods described above) or a blank if they deemed that none of the choices were a good translation. We used three annotators, and each test example was examined by two annotators. The data was shown in a tabular format, which facilitated inspection as well as provided stimuli from nearby languages. If there was a disagreement, we randomly chose one of the words to be the gold. Although the annotators do not know all 591 languages, the average inter-annotator agreement over the entire test set was 0.92, indicating that they have a good intuition of what names should look like even in languages that they are not familiar with.[9]

[9]Obviously annotators do not know all 591 languages. However, the large majority of names are related across languages (e.g. Mesir/Masr/... or Eiypta/Ehipto/...) and typically undergo systematic sound/orthography shifts between languages. Because of this, annotators can readily pick out the correct name with high interannotator agreement, especially after having seen the name's realizations in related languages.

	Align	Dist	MT	TR	Maj.	WC
Average	.682	.762	.805	.499	.778	.779

Table 6: Average 1-best accuracy on test words. TR = transformation rules, Maj. = majority vote, WC = weighted combination

English	Lng	BA	Distance	MT	TR	Consensus
Pharaoh	ctu	*egipto*	faraón	faraón	faraón	faraón
Tobias	por	—	tobias	tobias	tobias	tobias
Caesar	bzj	seeza	*roam*	seeza	*koam*	seeza
Phoenix	bcw	fenikəsə	fenikəsə	fenikəsə	feniki	fenikəsə
Pyrrhus	gbi	*sopater*	*sopater*	*sopater*	pirus	*sopater*
Zion	msm	sion	sion	sion	*siam*	sion
Claudia	agg	krodia	krodia	krodia	*kardia*	krodia
Claudia	cbu	*linu*	*linu*	*linu*	*linu*	*linu*
Pyrrhus	lac	*sópater*	*sópater*	*sópater*	*berea*	*sópater*

Table 7: Examples of best hypotheses from each system

6. Results

We evaluated the accuracy of the four methods on predicting the correct test word, in addition to system combination via majority voting and weighted combination (Table 6). Note that these methods are not independent of each other, since the MT approach builds on the results of the distance-based approach, and the transformation rules approach build upon the previous two. Results are across all 591 languages, and the size column indicates the number of non-empty gold words evaluated against. The weighted combination was done by taking a weighted consensus of the top hypothesis by each of the four approaches (Aligner, Distance, MT, and T-Rule), weighted by the average performance of each approach (.683, .763, .805, and .449, respectively).

On average, we find that the distance based approach and the MT approach perform comparably, showing large improvements over the baseline. We see that using a simple majority consensus to combine the outputs of the four methods, while not as good as MT alone on average, obtains the best performance on 12 of the test words, in contrast to 10 for MT. Our methods effectively generated citation forms for words that were not seen before by the aligner, which suggests that these methods can be effective in generating new vocabulary for low resource languages.

Some examples from each approach are presented in Table 7. Pharaoh and Tobias represent typical cases where our approaches improve upon the aligner baseline. The next few names show examples of disagreement between the different approaches. Claudia and Pyrrhus exemplify cases where none of the approaches chose the correct translation, largely due to high co-occurrence count of these incorrect names and their English counterparts, which caused misalignments.[10] On test words with low frequency across languages (Havilah, Jarmuth, Shishak, Mordecai and To-

[10]In the Bible, Claudia and Linus occur frequently together, as do Pyrrhus and Sopater (Pyrrhus' son) and Berea (the city they were from).

bias), our methods show 13–72% improvements over the baseline, suggesting that our approaches can overcome the problem of limited size of the data set.

In total, we found that the distance-based approach changed 1935 alignments (∼98/name), the MT approach changed 5516 existing alignments (∼280/name) and generated 3170 missing alignments (∼161/name), and the transformation rule approach changed 9053 alignments (∼460/name).

Further examination of the results reveals variations in the translation of certain names, which is linguistically interesting. Some notable examples include Caesar, which is split across languages whether to be pronounced with a hard *c* (e.g. Kaiser, Keizer, etc.), as in Classical Latin, or a softer *s* (e.g. Sisa) or *ts* (e.g. Czar/Tsar). In addition, Caesar is often translated as "Emperor" or "King" rather than transliterated. Likewise, Sheol is more often translated as the language's word for "Hades" or "Hell" than transliterated. This phenomenon is likely attributed to cultural influences.[11]

7. Applications

We present preliminary investigations into two potential use cases of our resource.

7.1. Named Entity Morphology

In the process of aligning a morphologically rich language to English, the aligner encounters many morphological variants of the same name. For example, the English name *David* is best aligned to the lemma form *Depito* in the Ankave language. However, *David* is also aligned to words of the form *Depito* + some affixes. By examining the morphological variants of *Depito*, we gather the following transformation rules:

Depito	$\emptyset \leftrightarrow \emptyset$
Depitoyá	$\emptyset \leftrightarrow$ **-yá**
Depitom~~i~~	$\emptyset \leftrightarrow$ -m~~i~~
Depitor~~i~~n~~i~~	$\emptyset \leftrightarrow$ -r~~i~~n~~i~~

A rule's right hand side can be considered an affix that denotes some aspect of morphology. Then, we can discover this affix's semantics by modeling the context surrounding the English word to which the morphological variant was aligned. The intuition is that inflected foreign forms of proper names should co-occur often with English prepositions and thus is correlated with a semantic case; or with preceding conjunctions, which can indicate plurality. By performing this process over all names in a language, we can determine the meaning of a given affix. For example, Ankave words ending with *-yá* occur frequently with the English preposition *of*, so *-yá* is very likely a marker for the genitive case. This process is especially applicable for identify the meaning of morphological affixes in low-resource languages where a grammar of the language may not exist and would be time-consuming to create by hand.

7.2. Transliteration

The Bible name translation matrix is naturally suited for training transliteration systems. By treating the weighted consensus names in our translation matrix as bitext, we trained character-based Moses SMT systems in a 80-10-10 train-dev-test split on a random subset of 40 languages, with the target language being English. We use a standard setup with a 4-gram language model, tuning with MERT, and no distortion to prevent reordering.

We compare against a simple baseline, Unidecode[12], which provides context-free Unicode to ASCII character mappings. While this is a naive baseline, it is reasonable for many low-resource languages for which this is perhaps the only form of transliteration available. Even on the order of several hundred training examples, underscoring the low-resource nature, the MT systems trained on this data transliterated on average much better than the baseline (Figure 1).

Figure 1 compares the accuracy of the baseline versus a Moses-based transliterator. The languages are represented by their ISO 639-3 language codes. Overall, the average accuracy of the baseline was 0.17, compared to 0.23 for Moses. Note that for the two highest scoring languages, ifb (Ifugao, a Malayo-Polynesian language) and bvr (Burarra, an Australian Aboriginal language), the Unidecode baseline performed better than Moses. For these two languages, most of the source and target words were identical. In such cases, Moses would just learn character identity mappings, and we suspect that the language model was biasing the system away from the correct answer. For example, the Ifugao-English system incorrectly transliterated *Amminadab* as *Aminadab*, whereas passing the source through unchanged would have achieved a higher accuracy. More investigation is necessary to determine the role of the language model in transliteration, especially of low-resource languages.

A followup on this work is presented in Wu and Yarowsky (2018), who compare the performance of several methods of transliteration, including phrase-based and neural machine translation, on our Bible names dataset.

[11] For example, Caesar was originally just a name but eventually became a title for the Roman emperor.

[12] `https://pypi.python.org/pypi/Unidecode`

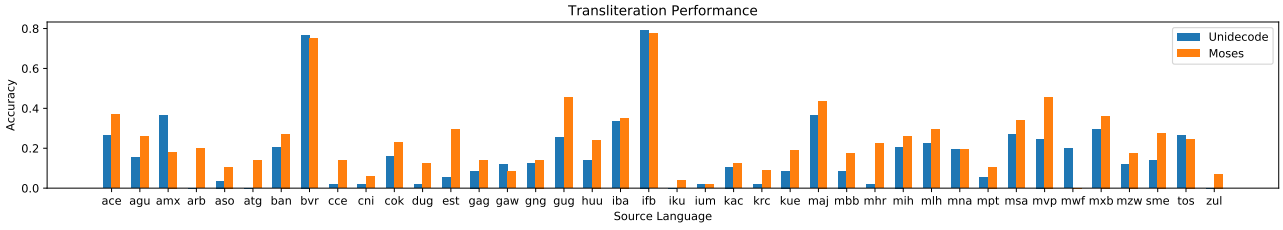


Figure 1: Comparing the performance of a baseline transliterator with a SMT-based transliterator trained on our translation matrix

8. Conclusion

We have developed and presented a novel and impactful dataset of 1129 person and place names in the Bible aligned over 591 languages. We have presented and empirically contrasted several techniques, including distance-based metrics, machine-translation-based transliteration models, and affixal transformation rules, for iteratively refining alignments. Our improved alignments outperform baseline alignments from a widely-used word alignment software in 97% of words in the test set. We release our Bible names translation matrix dataset, which we believe will be of value to researchers looking to build transliteration systems or other applications for low resource languages, for which the Bible may be the significant available bilingual, or even monolingual, text.

9. Acknowledgments

This work was supported in part by the DARPA LORELEI program. The findings, conclusions, and opinions found in this work are those of the authors and do not necessarily represent the views of the funding agency.

10. Bibliographical References

Dasigi, P. and Diab, M., (2011). *Proceedings of the 3rd Named Entities Workshop (NEWS 2011)*, chapter Named Entity Transliteration Generation Leveraging Statistical Machine Translation Technology, pages 106–111. Asian Federation of Natural Language Processing.

Fraser, A. and Marcu, D. (2006). Semi-supervised training for statistical word alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 769–776. Association for Computational Linguistics.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Jia, Y., Zhu, D., and Yu, S., (2009). *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration (NEWS 2009)*, chapter A Noisy Channel Model for Grapheme-based Machine Transliteration, pages 88–91. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics.

Kumar, S., J. Och, F., and Macherey, W. (2007). Improving word alignment with bridge languages. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Lee, J., Lee, D., and Lee, G. G. (2006). Improving phrase-based korean-english statistical machine translation. In *INTERSPEECH*.

Levenshtein, V. I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707–710, February.

Liang, P., Taskar, B., and Klein, D. (2006). Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*.

Mayer, T. and Cysouw, M. (2014). Creating a massively parallel bible corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Mermer, C. and Saraclar, M. (2011). Bayesian word alignment for statistical machine translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 182–187. Association for Computational Linguistics.

Moore, R. C. (2005). A discriminative framework for bilingual word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Moravcsik, E. (1978). Language contact. *Universals of human language*, 1:93–122.

Myers-Scotton, C. (2002). *Contact linguistics: Bilingual encounters and grammatical outcomes*. Oxford University Press on Demand.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics, Volume 29, Number 1, March 2003*.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Riesa, J. and Marcu, D. (2010). Hierarchical search for word alignment. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 157–166. Association for Computational Linguistics.

Sanchis, G. and Sánchez, J. A. (2008). Vocabulary extension via POS information for SMT. *Mixing Approaches to Machine Translation*.

Song, Y., Kit, C., and Chen, X., (2009). *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration (NEWS 2009)*, chapter Transliteration of Name Entity via Improved Statistical Translation on Character Sequences, pages 57–60. Association for Computational Linguistics.

Songyot, T. and Chiang, D. (2014). Improving word alignment using word similarity. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1840–1845. Association for Computational Linguistics.

Taskar, B., Simon, L.-J., and Dan, K. (2005). A discriminative matching approach to word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Tsvetkov, Y. and Dyer, C. (2016). Cross-lingual bridges

with models of lexical borrowing. *J. Artif. Intell. Res.(JAIR)*, 55:63–93.

Vaswani, A., Huang, L., and Chiang, D. (2012). Smaller alignment models for better translations: Unsupervised word alignment with the l0-norm. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 311–319. Association for Computational Linguistics.

Whitney, W. D. (1881). On mixture in language. *Transactions of the American Philological Association (1869-1896)*, 12:5–26.

Wu, W. and Yarowsky, D. (2018). A comparative study of extremely low-resource transliteration of the world's languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*. European Language Resources Association (ELRA).

Building a Word Segmenter for Sanskrit Overnight

Vikas Reddy*[1], Amrith Krishna*[2], Vishnu Dutt Sharma[3],
Prateek Gupta[4], Vineeth M R[5], Pawan Goyal[2]
[1]Dept. of Mining Engineering, [2]Dept. of Computer Science and Engineering, [4]Dept. of Mathematics,
[5]Dept. of Electrical Engineering, [1,2,4,5]IIT Kharagpur
[3]American Express India Pvt Ltd
{vikas.challaram, amrith}@iitkgp.ac.in, pawang@cse.iitkgp.ernet.in

Abstract

There is abundance of digitised texts available in Sanskrit. However, the word segmentation task in such texts are challenging due to the issue of *Sandhi*. In Sandhi, words in a sentence often fuse together to form a single chunk of text, where the word delimiter vanishes and sounds at the word boundaries undergo transformations, which is also reflected in the written text. Here, we propose an approach that uses a deep sequence to sequence (seq2seq) model that takes only the sandhied string as the input and predicts the unsandhied string. The state of the art models are linguistically involved and have external dependencies for the lexical and morphological analysis of the input. Our model can be trained "overnight" and be used for production. In spite of the knowledge lean approach, our system preforms better than the current state of the art by gaining a percentage increase of 16.79 % than the current state of the art.

Keywords: Word Segmentation, Sanskrit, Low-Resource Languages, Sequence to sequence, seq2seq, Deep Learning

1. Introduction

Sanskrit had profound influence as the knowledge preserving language for centuries in India. The tradition of learning and teaching Sanskrit, though limited, still exists in India. There have been tremendous advancements in digitisation of ancient manuscripts in Sanskrit in the last decade. Numerous initiatives such as the Digital Corpus of Sanskrit[1], GRETIL[2], The Sanskrit Library[3] and others from the Sanskrit Linguistic and Computational Linguistic community is a fine example of such efforts (Goyal et al., 2012; Krishna et al., 2017).

The digitisation efforts have made the Sanskrit manuscripts easily available in the public domain. However, the accessibility of such digitised manuscripts is still limited. Numerous technical challenges in indexing and retrieval of such resources in a digital repository arise due to the linguistic peculiarities posed by the language. Word Segmentation in Sanskrit is an important yet non-trivial prerequisite for facilitating efficient processing of Sanskrit texts. Sanskrit has been primarily communicated orally. Due to its oral tradition, the phonemes in Sanskrit undergo euphonic assimilation in spoken format. This gets reflected in writing as well and leads to the phenomena of *Sandhi* (Goyal and Huet, 2016). *Sandhi* leads to phonetic transformations at word boundaries of a written chunk, and the sounds at the end of a word join together to form a single chunk of character sequence. This not only makes the word boundaries indistinguishable, but transformations occur to the characters at the word boundaries. The transformations can be deletion, insertion or substitution of one or more sounds at the word ends. There are about 281 sandhi rules, each denoting a unique combination of phonetic transformations, documented in the grammatical tradition of Sanskrit. The proximity between two compatible sounds as per any one of the 281 rules is the sole criteria for sandhi. The *Sandhi* do not make any syntactic or semantic changes to the words involved. *Sandhi* is an optional operation relied solely on the discretion of the writer.

While the *Sandhi* formation is deterministic, the analysis of *Sandhi* is non-deterministic and leads to high level of ambiguity. For example, the chunk '*gardabhaścāśvaśca*' (the ass and the horse) has 625 possible phonetically and lexically valid splits (Hellwig, 2015). Now, the correct split relies on the semantic compatibility between the split words.

The word segmentation problem is a well studied problem across various languages where the segmentation is non-trivial. For languages such as Chinese and Japanese, where there is no explicit boundary markers between the words (Xue, 2003), numerous sequence labelling approaches have been proposed. In Sanskrit, it can be seen that the merging of word boundaries is the discretion of the writer. In this work, we propose a purely engineering based pipeline for segmentation of Sanskrit sentences. The word segmentation problem is a structured prediction problem and we propose a deep sequence to sequence (seq2seq) model to solve the task. We use an encoder-decoder framework where the *sandhied* (unsegmented) and the *unsandhied* (segmented) sequences are treated as the input at the encoder and the output at the decoder, respectively. We train the model so as to maximise the conditional probability of predicting the unsandhied sequence given its corresponding sandhied sequence (Cho et al., 2014). We propose a knowledge-lean data-centric approach for the segmentation task. Our approach will help to scale the segmentation process in comparison with the challenges posed by knowledge involved processes in the current systems (Krishna et al., 2017). We only use parallel segmented and unsegmented sentences during training. At run-time, we only require the input sentence. Our model can literally be trained overnight. The best performing model of ours takes less than 12 hours to

*The first two authors contributed equally
[1]http://kjc-sv013.kjc.uni-heidelberg.de/dcs/
[2]http://gretil.sub.uni-goettingen.de/
[3]http://sanskritlibrary.org/

train in a 'Titan X' 12 GB memory, 3584 GPU Cores system. Our title for the paper is inspired from the title for the work by Wang et al. (2015). As with the original paper, we want to emphasise on the ease with which our system can be used for training and at runtime, as it do not require any linguistically involved preprocessing. Such requirements often limit the scalability of a system and tediousness involved in the process limits the usability of a system.

Since Sanskrit is a resource scarce language, we use the *sentencepiece* (Schuster and Nakajima, 2012), an unsupervised text tokeniser to obtain a new vocabulary for a corpus, that maximises the likelihood of the language model so learnt. We propose a pipeline for finding the semantically most valid segmented word-forms for a given sentence. Our model uses multiple layers of LSTM cells with attention. Our model outperforms the current state of the art by 16.79 %.

2. Models for Word Segmentation in Sanskrit

A number of methods have been proposed for word segmentation in Sanskrit. Hellwig (2015) treats the problem as a character level RNN sequence labelling task. The author, in addition to reporting sandhi splits to upto 155 cases, additionally categorises the rules to 5 different types. Since, the results reported by the author are not at word-level, as is the standard with word segmentation systems in general, a direct comparison with the other systems is not meaningful. Mittal (2010) proposed a method based on Finite State Transducers by incorporating rules of *sandhi*. The system generates all possible splits and then provides a ranking of various splits, based on probabilistic ranking inferred from a dataset of 25000 split points. Using the same dataset, Natarajan and Charniak (2011) proposed a sandhi splitter for Sanskrit. The method is an extension of Bayesian word segmentation approach by Goldwater et al. (2006). Krishna et al. (2016) is currently the state of the art in Sanskrit word segmentation. The system treats the problem as an iterative query expansion problem. Using a shallow parser for Sanskrit (Goyal et al., 2012), an input sentence is first converted to a graph of possible candidates and desirable nodes are iteratively selected using Path Constrained Random Walks (Lao and Cohen, 2010).

To further catalyse the research in word segmentation for Sanskrit, Krishna et al. (2017) has released a dataset for the word segmentation task. The work releases a dataset of 119,000 sentences in Sanskrit along with the lexical and morphological analysis from a shallow parser. The work emphasises the need for not just predicting the inflected word form but also the prediction of the associated morphological information of the word. The additional information will be beneficial in further processing of Sanskrit texts, such as Dependency parsing or summarisation (Krishna et al., 2017).So far, no system successfully predicts the morphological information of the words in addition to the final word form. Though Krishna et al. (2016) has designed their system with this requirement in mind and outlined the possible extension of their system for the purpose, the system currently only predicts the final word-form.

3. Method

We use an encoder-decoder framework for tackling our segmentation problem, and propose a deep seq2seq model using LSTMs for our prediction task. Our model follows the architecture from Wu et al. (2016), originally proposed for neural machine translation. We consider the pair of *sandhied* and *unsandhied* sentences as source and target sentences, respectively. Following the insights from Sutskever et al. (2014), we reverse the sequence order at the input and we find that the reversal of the string leads to improvement in the results. We also use a deep architecture with 3 layers each at the encoder and decoder, as it is shown that deeper models perform better than shallow LSTM Models. We also experiment with models with and without attention and find that the model with attention leads to considerable improvement in performance of the system (Wu et al., 2016). Given the training set $\mathcal{S}$, our training objective is to maximise the log probability of the segmented sequences T where the unsegmented sequences S are given. The training objective is to maximise (Sutskever et al., 2014)

$$\frac{1}{|S|} \sum_{(T,S)\in\mathcal{S}} log\, p(T|S)$$

For a new sentence, we need to output a sequence T' with maximum likelihood for the given input (Sutskever et al., 2014).

$$T' = \underset{T}{\operatorname{argmax}}\ p(T|S)$$

LSTMs are used both at the encoder and decoder. We use softmax layer at the decoder and perform greedy decoding to obtain the final prediction. The outputs are then passed to the loss function which calculates the log-perplexity over the data samples in the batch. We then update the parameters via backpropagation and use Adam optimiser (Kingma and Ba, 2015) for our model.

Vocabulary Enhancement for the model - Sanskrit, being a resource poor language, the major challenge is to obtain enough data for the supervised task. While there are plenty of sandhied texts available for Sanskrit, it is hard to find parallel or unsandhied texts alone, as it is deterministic to get sandhied text from unsandhied texts.

In our case we use 105,000 parallel strings from the Digital Corpus of Sanskrit as released in Krishna et al. (2017). To handle the data sparsity, we adopt a purely engineering based approach for our model. Rather than relying on the real word boundaries, we use the '*sentencepiece*' model, an unsupervised text tokeniser (Schuster and Nakajima, 2012) to obtain a new vocabulary for the corpus. The method was originally proposed for segmentation problem in Japanese and Korean speech recognition systems. In the method, a greedy approach is used to identify new word units from a corpus that maximises the likelihood of the language model so learnt (Schuster and Nakajima, 2012).

Figure 1 shows the instance of words learnt from the *sentencepiece* model corresponding to the original input from the corpus. In the *sentencepiece* model, the 'space' in the original input is also treated as a character and is replaced

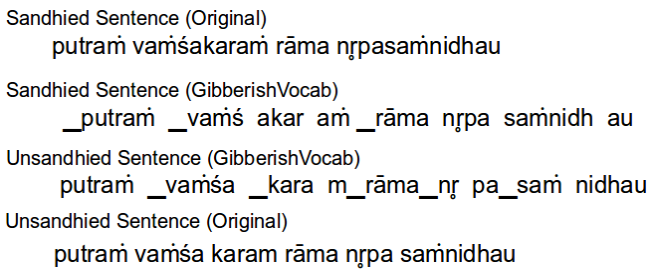


Figure 1: Sandhied and unsandhied sentence expressed in original writing and with the new learnt vocabulary 'GibberishVocab'.

with the special symbol '_'. So 'aṃ_rāma' is a word in our vocabulary, which originally is part of two words.
Our model is fed only the 'words' from the new vocabulary, henceforth to be referred to as '*GibberishVocab*'. Note that the decoder also outputs words from *GibberishVocab*. The output from decoder is then converted to the original vocabulary for evaluating the outputs. This is trivially done by reclaiming the original 'space' as the delimiter for the old vocabulary.

4. Experiments

4.1. Dataset

We used a dataset of 107,000 sentences from the Sanskrit Word Segmentation Dataset (Krishna et al., 2017). The dataset is a subset of the Digital Corpus of Sanskrit. From the dataset we only use the input sentences and the ground truth inflected word-forms. We ignore all the other morphological and lemma information available in the dataset.

4.2. Baselines

We compare the performance of our system with two other baseline systems.
supervisedPCRW - This is the current state of the art for word segmentation in Sanskrit. The method treats the problem as an iterative query expansion task. This is a linguistically involved approach, as at first a lexicon driven shallow parser is used to obtain all the phonetically valid segments for a given sentence. The sentence is then converted into a graph with the segments as the nodes. The edges are formed between every pair of nodes which can co-exist in a sentence and are not competing for the same input position. The edge weights are formed by weighted sum of random walks across typed paths. The authors use typed paths to obtain extended contexts about the word pairs from the candidate pool. The typed paths are designed with human supervision which is also linguistically involved.
GraphCRF - We use a structured prediction approach using graph based Conditional Random Fields, where we first obtain the possible candidate segments using the shallow parser and then convert the segments into a graph. For every node segment, we learn a word vector using fastText (Bojanowski et al., 2016).
segSeq2Seq - This is the proposed model as described in Section 3. but without attention.
attnSegSeq2Seq - This is the proposed model as described in Section 3. with attention.

Model	Precision	Recall	F-Score
GraphCRF	65.20	66.50	65.84
SupervisedPCRW	76.30	79.47	77.85
segSeq2Seq	73.44	73.04	73.24
attnsegSeq2Seq	90.77	90.3	90.53

Table 1: Micro-averaged Precision, Recall and F-Score for the competing systems on the test dataset of 4200 strings.

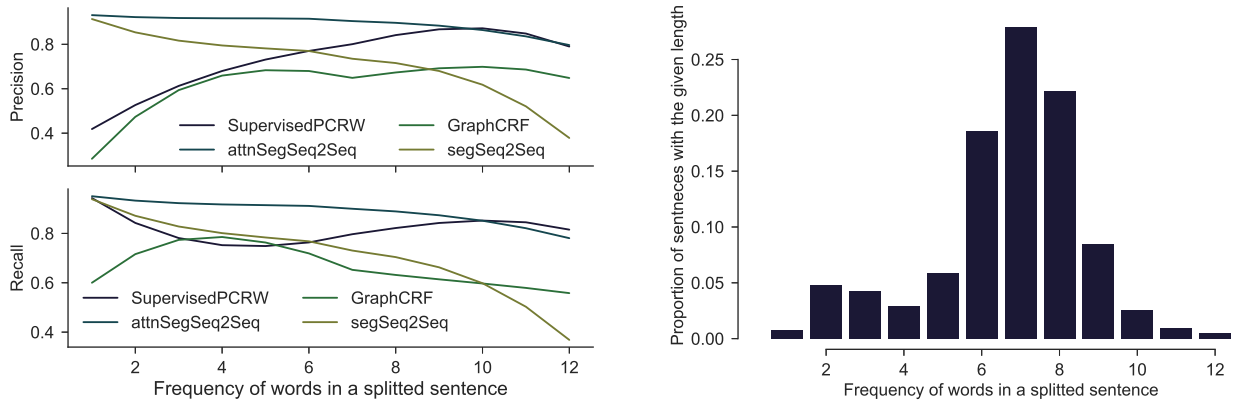
We report all our results on a test data of 4,200 sentences which was not used in any part of the training. From the dataset we ignore about 7,500 sentences which are neither part of training nor the test set. We used 214,000 strings both from input and output strings in the training data to obtain the *GibberishVocab* using *sentencepiece* model.
We use string-wise micro averaged precision, recall and F-Score to evaluate our model as is the standard with evaluating word segmentation models. We find that the default vocabulary size of 8,000 for the *GibberishVocab* works best. Of the 8,000 'words', the encoder vocabulary size is 7,944 and the decoder vocabulary size is 7,464. This shows the high overlap in the vocabulary in *GibberishVocab* at both input and output sides, in spite of the difference in phonetic transformations due to sandhi. Originally the training data contained 60,308 segmented words at the output side. By reducing the vocabulary size at decoder side to 7,464, we make the probability distribution (softmax) at the decoder layer denser. Even if we followed a linguistic approach there were 16,473 unique lemmas in the training dataset.

4.3. Training Procedure and Hyperparameters

Our models have 3 layers at both the encoder and decoder. The models contain an embedding layer which is a trainable matrix with individual word vector having a size of 128. Our LSTM layers consist of 128 cells at both the encoder and decoder layers. We train the sentences in a batch size of 128 and keep the sequence length of each sequence to be 35. The initial learning rate was set at 0.001 and we trained our system for 80 epochs after which the network parameters converged. We used Adam optimiser with parameter values β_1,β_2 as 0.9 and 0.999, respectively. We use dropout in the hidden layers with different settings from 0.1 to 0.4 in step sizes of 0.1. We find that a dropout of 0.2 is the best performing configuration. Dropout helps to avoid over-fitting of data (Srivastava et al., 2014). Both the 'segSeq2Seq' and 'attnSegSeq2Seq' models follow the same architecture and have the same hyperparameter settings and vary only on the attention mechanism.

4.4. Results

Table 1 shows the performance of the competing systems. We can find that the system 'attnSegSeq2Seq' outperforms the current state of the art with a percent increase of 16.29 % in F-Score. The model 'segSeq2Seq' falls short of the current state of the art with a percent decrease of 6.29 % in F-Score. It needs to be noted that the systems 'attnSegSeq2Seq' and 'segSeq2Seq' are exactly the same architectures other than the addition of attention in the former. But there is a percentage increase of 23.61 % for



(a) Precision and Recall for the competing systems grouped based on the count of words in each of the test sentence.

(b) Distribution of strings in test dataset grouped based on the count of words in each sentence.

Figure 2: Results on the test dataset. The sentences are grouped based on the count of words in the segmented sentences

both the systems. One probable reason for this is due to the free word order nature of sentences in Sanskrit. Since there are multiple permutations of words in a sentence which are valid syntactically and convey the same semantic meaning, the entire input context is required to understand the meaning of a sentence for any distributional semantic model.

Figure 2 shows the results of the competing systems on strings of different lengths in terms of words in the sentence. This should not be confused with sequence length. Here, we mean the 'word' as per the original vocabulary and is common for all the competing systems. For all the strings with up to 10 words, our system 'attnSegSeq2Seq' consistently outperforms all the systems in terms of both precision and recall. The current state of the art performs slightly better than our system, for sentences with more than 10 words. It needs to be noted that the average length of a string in the Digital Corpus of Sanskrit is 6.7 (Krishna et al., 2016). The proportion of sentences with more than 10 words in our dataset is less than 1 %. The test dataset has slightly more than 4 % sentences with 10 or more words. The 'segSeq2Seq' model performs better than the state of the art for both Precision and Recall for strings with less than or equal to 6 words. Figure 2a shows the proportion of sentences in the test data based on the frequency of words in it. Figure 2b shows the proportion of strings in the test dataset based on the number of words in the strings. Our systems attnSegSeq2Seq takes overall 11 hours 40 minutes and for 80 epochs in a 'Titan X' 12GB GPU memory, 3584 GPU Cores, 62GB RAM and Intel Xeon CPU E5-2620 2.40GHz system. For segSeq2Seq it takes 7 hours for the same setting.

5. Discussion

The purpose of our proposed model is purely to identify the word splits and correctness of the inflected word forms from a sandhied string. The word-level indexing in retrieval systems is often affected by phonetic transformations in words due to *sandhi*. For example, the term '*parameśvaraḥ*' is split as '*parama*' (ultimate) and '*īśvaraḥ*' (god). Now, a search for instances of the word '*īśvaraḥ*' might lead to missing search results without proper indexing. String matching approaches often result in low precision results. Using a lexicon driven system might alleviate the said issues, but can lead to possible splits which are not semantically compatible. For *parameśvaraḥ*', it can be split as '*parama*' (ultimate), '*śva*' (dog) and '*raḥ*' (to facilitate). Though this is not semantically meaningful it is lexically valid. Such tools are put to use by some of the existing systems (Krishna et al., 2016; Mittal, 2010) to obtain additional morphological or syntactic information about the sentences. This limits the scalability of those systems, as they cannot handle out of vocabulary words. Scalability of such systems is further restricted as the sentences often need to undergo linguistically involved preprocessing steps that lead to human in the loop processing. The systems by Krishna et al. (2016) and Krishna et al. (2017) assume that the parser by Goyal et al. (2012), identifies all the possible candidate chunks.

Our proposed model is built with precisely one purpose in mind, which is to predict the final word-forms in a given sequence. Krishna et al. (2017) states that it is desirable to predict the morphological information of a word from along with the final word-form as the information will be helpful in further processing of Sanskrit. The segmentation task is seen as a means and not an end itself. Here, we overlook this aspect and see the segmentation task as an end in itself. So we achieve scalability at the cost of missing out on providing valuable linguistic information. Models that use linguistic resources are at an advantage here. Those systems such as Krishna et al. (2016) can be used to identify the morphological tags of the system as they currently store the morphological information of predicted candidates, but do not use them for evaluation as of now. Currently, no system exists that performs the prediction of wordform and morphological information jointly for Sanskrit. In our case, since we learn a new vocabulary altogether, the real word boundaries are opaque to the system. The decoder predicts from its own vocabulary. But predicting morphological information requires the knowledge of exact word boundaries. This should be seen as a multitask learning set up. One possible solution is to learn '*GibberishVocab*' on the set of words rather than sentences. But

this leads to increased vocabulary at decoder which is not beneficial, given the scarcity of the data we have. Given the importance of morphological segmentation in morphologically rich languages such as Hebrew and Arabic (Seeker and Çetinoğlu, 2015), the same applies to the morphologically rich Sanskrit as well (Krishna et al., 2017). But, we leave this work for future.

6. Conclusion

In this work we presented a model for word segmentation in Sanskrit using a purely engineering based appraoch. Our model with attention outperforms the current state of the art (Krishna et al., 2016). Since, we tackle the problem with a non-linguistic approach, we hope to extend the work to other Indic languages as well where sandhi is prevalent such as Hindi, Marathi, Malayalam, Telugu etc. Since we find that the inclusion of attention is highly beneficial in improving the performance of the system, we intend to experiment with recent advances in the encoder-decoder architectures, such as Vaswani et al. (2017) and Gehring et al. (2017), where different novel approaches in using attention are experimented with. Our experiments in line with the measures reported in Krishna et al. (2016) show that our system performs robustly across strings of varying word size.

Code and Dataset

All our working code can be downloaded at `https://github.com/cvikasreddy/skt`. The dataset for training can be downloaded at `https://zenodo.org/record/803508#.WTuKbSa9UUs`

7. Bibliographical References

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October. Association for Computational Linguistics.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2006). Contextual dependencies in unsupervised word segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 673–680. Association for Computational Linguistics.

Goyal, P. and Huet, G. (2016). Design and analysis of a lean interface for sanskrit corpus annotation. *Journal of Language Modelling*, 4(2):145–182.

Goyal, P., Huet, G. P., Kulkarni, A. P., Scharf, P. M., and Bunker, R. (2012). A distributed platform for sanskrit processing. In *COLING*, pages 1011–1028.

Hellwig, O. (2015). Using recurrent neural networks for joint compound splitting and sandhi resolution in sanskrit. In *4th Biennial Workshop on Less-Resourced Languages*.

Kingma, D. and Ba, J. (2015). Adam: A method for stochastic optimization. In *In 3rd International Conference on Learning Representations (ICLR) 2015*.

Krishna, A., Santra, B., Satuluri, P., Bandaru, S. P., Faldu, B., Singh, Y., and Goyal, P. (2016). Word segmentation in sanskrit using path constrained random walks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 494–504, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Krishna, A., Satuluri, P. K., and Goyal, P. (2017). A dataset for sanskrit word segmentation. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 105–114, Vancouver, Canada, August. Association for Computational Linguistics.

Lao, N. and Cohen, W. W. (2010). Relational retrieval using a combination of path-constrained random walks. *Machine learning*, 81(1):53–67.

Mittal, V. (2010). Automatic sanskrit segmentizer using finite state transducers. In *Proceedings of the ACL 2010 Student Research Workshop*, pages 85–90. Association for Computational Linguistics.

Natarajan, A. and Charniak, E. (2011). S3-statistical saṃdhi splitting. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 301–308. Association for Computational Linguistics.

Schuster, M. and Nakajima, K. (2012). Japanese and korean voice search. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 5149–5152. IEEE.

Seeker, W. and Çetinoğlu, Ö. (2015). A graph-based lattice dependency parser for joint morphological segmentation and syntactic analysis. *Transactions of the Association for Computational Linguistics*, 3:359–373.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Wang, Y., Berant, J., and Liang, P. (2015). Building a semantic parser overnight. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on*

Natural Language Processing (Volume 1: Long Papers), pages 1332–1342, Beijing, China, July. Association for Computational Linguistics.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Xue, N. (2003). Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29–48.

Simple Semantic Annotation and Situation Frames: Two Approaches to Basic Text Understanding in LORELEI

Kira Griffitt, Jennifer Tracey, Ann Bies, Stephanie Strassel
Linguistic Data Consortium. University of Pennsylvania
3600 Market Street, Suite 810, Philadelphia, PA. 19104 USA
{kiragrif, garjen, bies, strassel}@ldc.upenn.edu

Abstract
We present two types of semantic annotation developed for the DARPA Low Resource Languages for Emerging Incidents (LORELEI) program: Simple Semantic Annotation (SSA) and Situation Frames (SF). Both of these annotation approaches are concerned with labeling basic semantic information relevant to humanitarian aid and disaster relief (HADR) scenarios, with SSA serving as a more general resource and SF more directly supporting the evaluation of LORELEI technology. Mapping between information in different annotation tasks is an area of ongoing research for both system developers and data providers. We discuss the similarities and differences between the two types of LORELEI semantic annotation, along with ways in which the general semantic information captured in SSA can be leveraged in order to recognize HADR-oriented information captured by SF. To date we have produced annotations for nineteen LORELEI languages; by the program's end both SF and SSA will be available for over two dozen typologically diverse languages. Initially data is provided to LORELEI performers and to participants in NIST's Low Resource Human Language Technologies (LoReHLT) evaluation series. After their use in LORELEI and LoReHLT evaluations the data sets will be published in the LDC catalog.

Keywords: semantic annotation, information extraction, linguistic resources

1. Introduction

Most of the world's languages are under-resourced for Human Language Technology (METANET, 2010; Rehm and Uszkoreit, 2012), but lack of resources does not correlate with lack of need for such technologies. The DARPA Low Resource Languages for Emerging Incidents (LORELEI) Program aims to advance the capabilities of NLP in low-resource languages, with a particular focus on using NLP to obtain situational awareness for an incident like a natural disaster involving a low-resource language within a short time of the emergence of that incident (DARPA, 2014).

Linguistic Data Consortium (LDC) is building a variety of linguistic resources for nearly three dozen low resource languages for the LORELEI program (see Table 1). Representative Language Packs – consisting of large volumes of formal and informal monolingual and parallel (with English) text with a variety of manual annotations to support situational awareness, plus a lexicon, grammatical sketch and basic processing tools – are designed to enable research into language universals and cross-language projection. Incident Language Packs contain manually labeled evaluation data designed to test system performance on tasks related to situational awareness for one or more surprise languages per year that remain unknown until the start of the annual evaluation (Strassel and Tracey, 2016).

Akan (Twi)	Hungarian	Tagalog	Vietnamese
Amharic	Indonesian	Tamil	Wolof
Arabic	Mandarin	Thai	Yoruba
Bengali	***Oromo***	***Tigrinya***	Zulu
English	Russian	Turkish	
Farsi	Somali	***Ukrainian***	
Hausa	Spanish	***Uyghur***	
Hindi	Swahili	Uzbek	

Table 1: LORELEI Representative and ***Incident*** Languages

This paper focuses on two semantic annotation tasks developed by LDC to support LORELEI research and evaluation: Simple Semantic Annotation (SSA) and Situation Frame (SF). Both SSA and SF label basic information relevant to humanitarian aid and disaster relief (HADR) scenarios. Situation Frame annotation directly corresponds to the LORELEI SF evaluation task, with a focus on actionable information contained in HADR-related documents, where "actionable" refers to the kind of content that mission planners might require in order to mount a response to an incident. SSA represents a more general approach to semantic annotation, albeit in the HADR domain.

By design, no training data of any kind is provided for LORELEI incident languages, since data of this type is unlikely to be available at the start of an incident involving a low resource language. Instead systems must make use of more general linguistic resources, through transfer learning, annotation projection and language universals, in order to rapidly respond to the need for situational awareness in a new language. SSA serves as a general semantic resource that can be used by system developers to build language-independent algorithms capable of labeling actionable HADR information in documents from a surprise incident language at test time. In the original LORELEI data plan SSA annotation is provided for all Representative Languages while no Situation Frame annotation is provided (apart from answer key annotations on the incident language test set). After the Year 2 evaluation the data plan was augmented to provide a small amount of SF annotation for all Representative Languages, though it remains the case that no training data -- whether SSA or SF -- is provided for any incident language.

2. Simple Semantic Annotation

Simple Semantic Annotation supports LORELEI's goal of situational awareness by labeling basic information about physical events and disaster-relevant situations, their participants, and their locations in text data. Given

LORELEI's low resource language setting combined with the need to simultaneously create resources for dozens of languages, the SSA task was designed with a naïve annotator in mind (i.e. without formal linguistic training or prior annotation experience). In this way it contrasts with more complex predicate-argument focused semantic representation schemes like Abstract Meaning Representation (AMR) (Banarescu, et al., 2013) Automatic Content Extraction (ACE) (Doddington, et al., 2004), PropBank (Palmer, et al., 2005), FrameNet (Baker, et al. 1998), Richer Event Description (RED) (Ikuta, et al., 2014), and Universal Decompositional Semantics (UDS) (White, et al., 2016), which require a background in linguistics and/or a long training period.

In order to make SSA feasible for non-experts to master quickly, we annotate a small number of broad, underspecified predicate and argument categories that do not require fine-grained semantic distinctions. We generally select names, pronouns, or heads of nominal phrases as annotation extents, but annotators are allowed to select "intuitive extents" if needed (e.g. for multiword expressions), meaning that strict rules about selecting head words are not enforced. Each sentence is annotated independently, with reference to the full document as needed for additional context, and there is no coreference of arguments or predicates. The following sections describe the predicate and argument categories annotated in SSA.

2.1 Predicate Categories

SSA has two coarse-grained predicate categories: Acts and States. Acts are event-like predications describing change, while States are situation-like, describing non-changing or ongoing circumstances. In SSA, Acts and States are semi-open classes. There is no fixed set of predicates made available to annotators and no typing of predicates beyond the broad categories of Act and State. Since exhaustive annotation of all Acts and States is impractical, and since SSA is focused on tagging information relevant to situational awareness, SSA annotators are instructed to restrict their annotations to capturing the following types of Acts and States:

- Physical Acts, which are events, actions, or activities that take place in the observable, material world (e.g. bombing) as opposed to events that do not (e.g. thinking)
- Disaster-Relevant States, which are situations that constitute, are caused by, or provide information relevant to a disaster or disaster-relief effort (e.g. scared, without water, etc.), but not those that bear no relationship to a disaster scenario (e.g. "married", "excited [about a movie]," etc.).

2.2 Argument Categories

We define three coarse-grained argument categories for SSA: Agents, Patients, and Places. SSA Agents are similar to the traditional linguistic notion of Agent, though slightly broader, encompassing the person or thing that does or performs an Act, or the person or thing that causes or enables an Act to occur or a State to arise. Agents are often typical entities like persons, organizations, and geopolitical entities (e.g. if the United Nations delivered supplies after a disaster, we would annotate "United Nations" as the Agent of the predicate "delivered"). SSA Patients include the traditional linguistic notion of Patient, but also include recipients, beneficiaries, experiencers, and purposes/goals. Arguments may be typical entities, or they may be other Acts or States. Place includes the physical location where an Act or State occurred, as well as directional locations.

3. Situation Frame Annotation

The Situation Frame annotation task was defined to support LORELEI technology evaluations, and is directly aligned with the goal of situational awareness in disaster response scenarios (Strassel, et al., 2016). The objective of SF is to aggregate information into a comprehensive, actionable understanding of the basic facts needed to mount a response to an emerging situation, including the following:

- Characterization of the situation type
- Status of need/issue and resolution of need
- Localization of the situation to a place
- Sentiment, Emotion, or Cognitive State (SEC).

The information is arranged into "frames," which represent *needs* that may require a response (e.g., food, shelter, etc.), or *issues* that may affect the ability to deliver aid (e.g., widespread crime in the area). The frame contains all of the type, status, place, entity and SEC information elements for a given need or issue. Note that the term "frame" is used here in the general sense of a linguistic frame, rather than with any more specific reference to PropBank frame files (Palmer, et al. 2005) or the FrameNet lexical database (Baker, et al. 1998). Situation Frame annotation captures information about needs and issues at the document level, rather than capturing semantics at the event or word level.

3.1 Situation Type

For each frame, annotators characterize the situation by indicating the type of need or issue that exists, selecting from the types shown in Table 2. Multiple needs or issues in a document result in multiple frames, one for each unique combination of type, status, and place. Need and issue types were defined with input from LORELEI stakeholders and from existing annotation schemes such as MicroMappers (Imran, et al., 2014).

Need Types	Issue Types
• Evacuation • Food Supply • Search/Rescue • Utilities/Energy/Sanitation • Infrastructure • Medical Assistance • Shelter • Water Supply	• Civil Unrest/ Widespread Crime • Regime Change • Terrorism/Extreme Violence

Table 2: Situation Frame Types

Situation Frame annotation requires the use of inference, since annotators must be able to recognize that an implied need exists even when it is not explicitly stated (e.g. when a document about a hurricane says that "housing across the island was destroyed", annotators should label a shelter need). However, inference is a slippery slope, and too much use of inference can lead annotators to create frames for all possible needs typically associated with an

incident – even when they are not implied by the document (e.g. creating search/rescue, shelter, water supply, utilities/energy/sanitation, medical assistance and infrastructure needs anytime an earthquake is mentioned, even when the document does not imply such needs currently exist.) A major challenge for SF annotation is creating guidelines and training annotators to use the right amount of inference such that annotators create frames for needs (or issues) that are strongly implied or inevitable, but not when it is merely possible or even likely. Because of the inherent challenge in achieving highly consistent annotations involving inference, SF evaluation data is labeled by a panel of annotators, which is reflected in system scoring (NIST, 2017).

3.2 Situation Status

Every situation frame is labeled for the status of the need or issue (current or not) as well as the status of the resolution (sufficient or not) for needs. Annotators also label the source(s) of information about the need and/or its resolution, as well as the entity/entities involved in resolving the need. For example, if the Red Cross and the government of Mexico are both mentioned in the document as contributing to the relief of a food need, "Red Cross" and "Mexico" would be added to the frame's "resolved by" element. Only entities named in the document can be selected as "reporting" or "resolving" the need or issue.

3.3 Situation Location

Annotators localize the situation by specifying the place where it occurs, selecting a single Location or Geopolitical Entity for each frame, or an indication that no named place entity relevant to the frame is mentioned in the document. Only named entities can be selected as locations for a situation frame. If there are multiple mentions of a situation (e.g. same place but different status; same need/issue in different places) multiple frames are created. Labeling SF place can be challenging due to the fact that news reports, tweets, etc. may be ambiguous or vague about the exact location of a need, even when the type of need is clear. An incident may be discussed in connection with several different (related or adjacent) places, but it is not always clear whether they are all affected by the same set of needs or whether the status or urgency differs among the various locations.

3.4 Situation SEC

Finally, annotators denote SEC for a situation by indicating whether it is urgent. Urgency can be both a property of the emotional/cognitive state of those affected as well as a property of the situation itself, regardless of any emotional component. For example, both "We're in desperate need of water. It's awful!" and "Officials say it is imperative that the drinking water be brought to the area immediately" are tagged as urgent. In pilot experiments the SF task has been augmented with additional SEC information, including positive or negative sentiment and two specific emotions: anger, and fear. These are labeled if they are present in the document and related to the frames, and annotation includes the sentiment holder, target, and sentiment/emotion value (positive, negative, anger, fear).

4. Mapping from SSA to SF

While SSA and SF annotation have different scopes (SSA is sentence-level while SF is document-level), because they both target disaster-relevant semantic information it is conceivable that the general-purpose semantic information captured by SSA could be utilized for improved performance on the incident language Situation Frames task through transfer learning. In this section we consider the possibility of direct mappings or at least inferential correspondences between the more general-purpose, sentence-level information captured in SSA and the use-case oriented, document-level information labeled in SF. Note that there are no automated techniques used to generate the SSA-SF mappings in the following sections. All mappings discussed in this paper have been generated by manual comparison of the annotations.

Let us take the following document excerpt as an example and point of comparison for SSA and SF annotation:

> Suak Beukah, Indonesia: Airdrops have provided enough food for the survivors, but in a village where half of the people were wiped out by the tsunami, the Red Cross now fears malaria could kill more if medical supplies don't arrive soon.

Figure 1: Sample document excerpt

For this document excerpt, SF would annotate two frames, one a Food Need, the other a Medical Assistance Need:

Type: Food Place: Suak Beukah Proxy: Yes Status: Current Urgent: No Resolution: Sufficient Reported by: N/A Resolved by: N/A SEC: N/A	Type: Medical Assistance Place: Suak Beukah Proxy: Yes Status: Future only Urgent: No Resolution: Insufficient/Unknown Reported by: Red Cross Resolved by: N/A SEC: Negative, Fear; Source: Red Cross; Target: Frame

Figure 2: Sample document with SF annotation

This information provides an aggregated, actionable understanding that there are food and medical assistance needs that exist in Suak Beukah as a result of the tsunami described in the document.

Act: Airdrops Patient: food Place: Suak Beukah Place: Indonesia Act: provided Agent: airdrops Patient: enough Patient: survivors Place: Suak Beukah Place: Indonesia State: enough Agent: food Patient: survivors Place: Suak Beukah Place: Indonesia Act: wiped out Agent: tsunami Patient: half Place: Suak Beukah Place: Indonesia	Act: tsunami Place: Suak Beukah Place: Indonesia State: fear Agent: kill Patient: Red Cross State: malaria Place: Suak Beukah Place: Indonesia Act: kill Agent: malaria Patient: more Place: Suak Beukah Place: Indonesia Act: arrive Patient: supplies Place: Suak Beukah Place: Indonesia

Figure 3. Sample document with SSA annotation

In contrast, SSA captures several specific, individual physical Acts or disaster-relevant States, including: airdrops of food, there being enough food, a tsunami, being wiped out by a tsunami, the occurrence of malaria, and deaths from malaria.

The Place arguments for these SSA annotations also indicate that the events and situations described in the document are located in Suak Beukah. These SSA annotations provide a detailed, but uncategorized picture of the individual events and situations that are occurring in the wake of the tsunami as described in the document.

Given each task's annotations for this excerpt, we will now identify and examine informational correspondences between the SSA annotations and two SF frames.

4.1 Mapping SF Frame Type from SSA

Looking at the second column of SSA annotations, we identify several correspondences between the SSA elements and SF frame type (indicated by shaded boxes): The SSA annotations do not contain the lexical items "medical" or "assistance", and so direct mapping based on the lexical items identified in SSA and the names of the SF need type is not possible. However, using world knowledge, we can identify "malaria" as a disease or medical condition that causes deaths, and thus infer that the document is describing a medical situation. Further, we can use world knowledge to identify the Red Cross as an organization providing assistance, and observe that supplies are being provided in the area where a medical situation exists. These two inferences let us infer a Medical Assistance frame type.

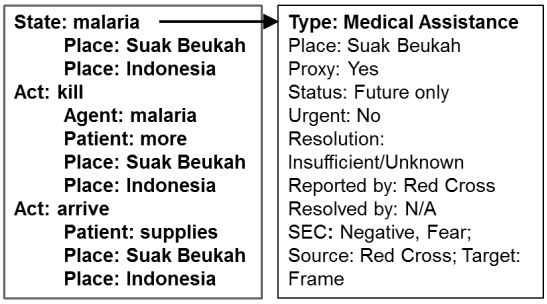


Figure 4: Mapping frame type

4.2 Mapping SF Place from SSA

Turning to the Place information type, we can see that it is possible to directly ascertain Place information for the Medical Assistance Frame from the SSA annotations:

Here, as shown in Figure 5, all the medically-relevant SSA Predicates have "Suak Beukah" as their Place, which lets us map Suak Beukah as the Place for the Medical Assistance frame.

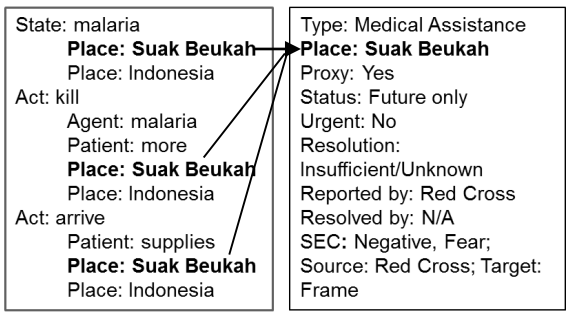


Figure 5: Mapping place information

4.3 Understanding SF Resolution

Looking at the SF Food need frame, we see that it is also possible to ascertain Resolution information:

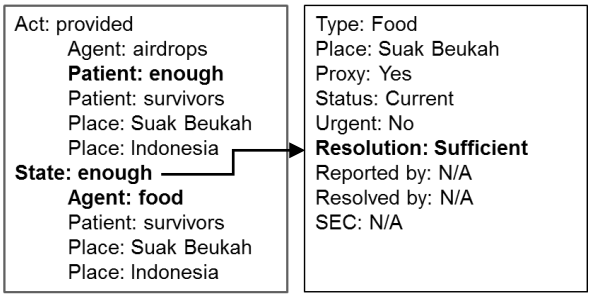


Figure 6: Mapping resolution information

Looking at the SSA annotations in the first column of Figure 6 that are relevant to food needs, we can see in the "Act: provided" Predicate that "enough" of something has been provided, and in the "State: enough" Predicate that the thing there is enough of is food.

Based on this information, we can use lexical knowledge to infer that there being "enough" of something corresponds to there being a sufficient amount of that thing, which allows us to recognize that the Resolution of the Food need frame is "Sufficient".

4.4 Mapping SF SEC from SSA

Finally, returning to the Medical Assistance frame, we can observe correspondences that allow us to identify SEC information for the Medical Assistance frame from SSA.

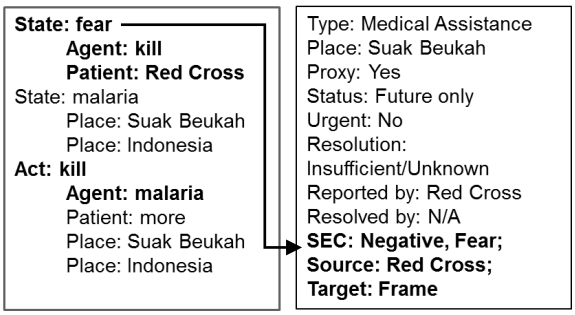


Figure 7: Mapping SEC information

Looking at the SSA Predicates with correspondence to the Medical Assistance frame in Figure 7, we can use the SSA Predicate "State: fear" to directly recognize an SEC value of Fear for the frame, and use world knowledge to identify "fear" as a negative emotion, which allows us to map a Negative sentiment for the frame. Further, the SSA Argument "Patient: Red Cross" to identifies the Red Cross as the experiencer "State: fear", which we can use to recognize the Red Cross as the Source of the Negative sentiment and Fear SEC values.

Finally we can use the SSA Argument "Agent: kill" to identify people being killed as the reason the Red Cross is experiencing fear. We can then trace through the other SSA annotations "Act: kill, Agent: malaria" and "State: malaria" to understand that malaria as the cause of people being killed. Since this set of Predicates and Arguments all correspond to the Medical Assistance frame, we can infer the frame itself as the SEC Target value for this frame.

5. Conclusion

We have presented two types of semantic annotation that were developed at LDC to support the humanitarian aid and disaster relief use case for the LORELEI program: Simple Semantic Annotation and Situation Frames. We have further presented some ways in which the general-purpose semantic information captured in SSA can be leveraged in order to map to the use-case information captured by SF. This is an area of on-going research for both system developers and annotation creation.

The linguistic resources described here have been distributed to LORELEI performers and to participants in the NIST Open Low Resource Human Language Technologies (LoReHLT) evaluation (NIST, 2017). As the data sets are completed under LORELEI they will be published in the LDC catalog, making them generally available to the broader research community.

We have produced manually annotated SSA data for 25K words in each of 9 representative languages (Amharic, Arabic, Chinese, Hungarian, Farsi, Russian, Spanish, Vietnamese, Yoruba). This data has been used as training data as part of the representative language packs. We have also manually annotated SF for 3 incident languages thus far: 200Kw in Uyghur, and 50Kw in each of Tigrinya and Oromo. This data has been used as evaluation data as part of the LoReHLT 2016 (Uyghur) and 2017 (Tigrinya and Oromo) evaluations. These resources have been released in language packs to participants in the LORELEI program and LoReHLT evaluation participants, and all will be made available to the larger research community as part of the LDC catalog starting in February 2018.

In addition, we have produced a multi-way annotated SF dataset in English, as part of an experiment on the degree of inference that is possible with SF annotation. Annotation for both SSA and SF is on-going in additional languages, and we expect to complete 25Kw of SSA for an additional 12 languages, and 25Kw of SF for an additional 24 languages by the end of 2018.

9. Acknowledgements

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-15-C-0123. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA.

6. Bibliographical References

Baker, C., Fillmore, C., and Lowe, J. (1998). The Berkeley FrameNet project. In Proceedings of COLING/ACL, pages 86–90, Montreal.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M. and Schneider N. (2013). Abstract Meaning Representation for Sembanking. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse.

DARPA. 2014. Broad Agency Announcement: I2O Low Resource Languages for Emergent Incidents (LORELEI). Defense Advanced Research Projects Agency, DARPA-BAA-15-04.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., Weischedel, R. (2004). Automatic Content Extraction (ACE) Program - Task Definitions and Performance Measures. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04), Lisbon, Portugal, May 24-30. European Language Resource Association (ELRA).

Ikuta, R., Styler, W., Hamang, M., O'Gorman, T., and Palmer, M. (2014). Challenges of adding causation to richer event descriptions. In Proceedings. of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation, pages 12–20.

Imran, M., Castillo, C., Lucas, J., Meier, P., Vieweg, S. (2014). AIDR: Artificial Intelligence for Disaster Response. In Proceedings of the 23rd International World Wide Web Conference, Seoul, South Korea, April 7-11.

METANET. (2010). META-NET White Paper Series: Press Release, http://www.meta-net.eu/whitepapers/press-release-en, accessed March 16, 2016.

NIST. (2017). Low Resource Human Language Technologies (LoReHLT) evaluations. https://www.nist.gov/itl/iad/mig/lorehlt17-evaluations. August 2017. Accessed September, 2017.

Palmer, M., Gildea, D., Kingsbury, P. (2005). The Proposition Bank: A Corpus Annotated with Semantic Roles. *Computational Linguistics Journal*, 31:1.

Rehm, Georg, Hans Uszkoreit, eds. 2012. META-NET White Paper Series: Europe's Languages in the Digital Age, URL: www.meta-net.eu/whitepapers.

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In Proceedings of the Fourteenth Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies, Third Workshop on EVENTS: Definition, Detection, Coreference, and Representation, Denver, Colorado, June 4.

Strassel, S., Bies, A. and Tracey, J. (2017). Situational Awareness for Low Resource Languages: the LORELEI Situation Frame Annotation Task. In Proceedings of the First Workshop on Exploitation of Social Media for Emergency Relief and Preparedness, Aberdeen, Scotland, April 9.

Strassel, S., and Tracey, J. (2016). LORELEI Language Packs: Data, Tools, and Resources for Technology Development in Low Resource Languages. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 3273-3280, Portoroz, Slovenia, May 23-28. European Language Resource Association (ELRA).

White, A., Reisinger, D., Sakaguchi, K., Vieira, T., Zhang, S., Rudinger, R., Rawlins, K., and Van Durme, B. (2016). Universal decompositional semantics on universal dependencies. In Proceedings. of EMNLP. pages 1713– 1723.

Abstract Meaning Representation of Constructions: The More We Include, the Better the Representation

Claire Bonial,[1] Bianca Badarau,[2] Kira Griffitt,[3] Ulf Hermjakob,[4] Kevin Knight,[4] Tim O'Gorman,[5] Martha Palmer,[5] Nathan Schneider[6]

[1]U.S. Army Research Laboratory, [2]SDL, [3]Linguistic Data Consortium, [4]USC Information Sciences Institute, [5]University of Colorado, [6]Georgetown University

claire.n.bonial.civ@mail.mil

Abstract

We describe the expansion of the Abstract Meaning Representation (AMR) project to provide coverage for the annotation of certain types of constructions. Past AMR annotations generally followed a practice of assigning the semantic roles associated with an individual lexical item, as opposed to a flexible pattern or template of multiple lexical items, which characterizes constructions such as 'The X-er, The Y-er' (exemplified in the title). Furthermore, a goal of AMR is to provide consistent semantic representation despite language-specific syntactic idiosyncracies. Thus, representing the meanings associated with fully syntactic patterns required a novel annotation approach. As one strategy in our approach, we expanded the AMR lexicon of predicate senses, or semantic 'rolesets,' to include entries for a growing set of constructions. Despite the challenging practical and theoretical questions encountered, the additions and updates to AMR annotation described here ensure more comprehensive semantic representations capturing both lexical and constructional meaning.

Keywords: Semantics, Constructions, Meaning Representation

1. Introduction

The Abstract Meaning Representation (AMR) project (Banarescu et al., 2013) has created a manually annotated "semantics bank" of text. A goal of AMR is to capture core facets of meaning while abstracting away from idiosyncratic syntactic facts; thus, for example, *She adjusted the machine* and *She made an adjustment to the machine* share the same AMR. The purpose of these annotations is to support natural language processing (NLP) applications such as natural language understanding, generation, and summarization (Liu et al., 2015; Pourdamghani et al., 2016), machine translation, question answering (Mitra and Baral, 2016), information extraction (Pan et al., 2015), and biomedical text mining (Garg et al., 2016; Rao et al., 2017; Wang et al., 2017). With a growing body of over 70 research papers using AMR,[1] the corpus is becoming a benchmark dataset.

As a practical NLP resource, AMR annotation has focused on providing coverage for the more frequent and predictably patterned linguistic phenomena, and therefore has not necessarily provided adequate representations for some of the rarer structures found in the long tail of language (Zipf, 1949). However, as the project matures, we are aiming to expand the representation to go beyond capturing the semantics of purely compositional language to better capture the semantics of constructions in which the meaning is more than the sum of its parts (lexical meanings). From its inception, in 'abstracting away' from language-specific syntactic facts to represent core meaning elements, AMR has annotated a variety of semi- and non-compositional phrases according to a more general lexicon entry capturing the meaning, independent of any particular syntactic derivation. For example, certain realizations of *but, while, whereas, however* and *on the other hand* are represented using a single lexicon entry, contrast-01. Nonetheless, there were gaps and inconsistencies in the treatment of various constructions, including those with the common feature of expressing degrees and quantities of properties and things. There are 8,117 instances of the AMR Degree modifier and 11,785 instances of the Quantity modifier in the last AMR release corpus of 39,260 sentences (Knight et al., 2017). Of course, many of these are not instances of the constructions of focus in this paper; nonetheless, this work provides a deeper and more consistent representation of the semantics of these relatively prevalent phenomena across both compositional and constructional usages. In order to capture the meaning of degree/quantity constructions, we expanded the AMR lexicon of predicate senses, or semantic 'rolesets,' to include entries for a growing set of constructions. Here, we describe the theoretical challenges involved in developing the rolesets and implementing clear guidelines, provide details on the novel rolesets and examples, as well as evaluation in piloting.

[1]Bibliographies: `https://amr.isi.edu/research.html`, `http://people.cs.georgetown.edu/nschneid/cosc672/s17/amr-papers.html`

2. Background & Related Work

2.1. Constructions

Everyday language is built up of prefabricated parts and templates that form a speaker's individual discourse experience (Hopper, 1998; MacWhinney, 2001; Bybee and McClelland, 2005). These templates can be thought of as 'constructions,' which are generally defined as any pairing of form and meaning, including both lexical items and phrases (Fillmore et al., 1988; Kay and Fillmore, 1999; Michaelis and Lambrecht, 1996). Fully syntactic patterns arguably have associated meanings: e.g., 'argument structure constructions' are thought to license a verb and its arguments within a clause (Goldberg, 1995). For example, the verb *blink*, for which we might typically expect the mention of a *blinker* and perhaps *eyes*, can be found with roles typical of a caused

motion verb: *Jerry blinked the snow off of his eyelashes*. In the case of adjectival predicates, typically one-place predicates (i.e., licensing only one argument), such as *tall*, can be found in contexts with comparisons, superlatives, and even result or consequence clauses: *The boy was not tall enough to ride the rollercoaster.*[2] Whereas lexicosemantic approaches (Frege, 1879; Chomsky, 1981; Pinker, 1989) may take these structures as evidence of a distinct sense of *blink* or *tall* licensing additional arguments, in a construction grammar approach, the construction itself can license the arguments. This distinction also affects what is thought to be stored in the lexicon and, perhaps most relevant to AMR, what would need to be represented in a computational lexicon: additional senses tied to a lexical predicate, or constructional entries.

2.2. Abstract Meaning Representation Annotation

The AMR project annotations are completed on a sentence-by-sentence basis, where each sentence is represented by a rooted directed acyclic graph (DAG). See Figure 1.

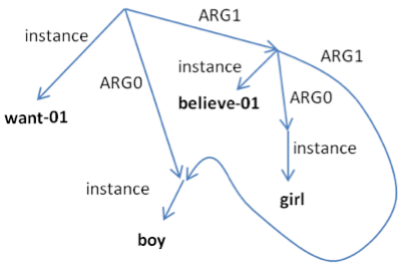


Figure 1: AMR graph for *The boy wants the girl to believe him.*

For ease of creation and manipulation, annotators work with the PENMAN representation of the same information (Penman Natural Language Group, 1989). For example, the following AMR can be glossed as:

1. The boy wants the girl to believe him.

```
(w / want-01
   :ARG0 (b / boy)
   :ARG1 (b2 / believe-01
             :ARG0 (g / girl)
             :ARG1 b))
```

In neo-Davidsonian fashion (Davidson, 1969; Parsons, 1990), we introduce variables (or graph nodes) for entities, events, properties, and states. Leaves are labeled with concepts, so that `(b / boy)` refers to an instance (`b`) of the concept *boy*. Relations link entities, so that `(w / walk-01 :location (p/ park))` means the walking (`w`) was in the park (`p`). When an entity plays multiple roles in a sentence (e.g., `(b / boy)` above), we employ re-entrancy in graph notation (nodes with multiple parents) or variable re-use in PENMAN notation.

AMR concepts are either English words (`boy`), PropBank (Palmer et al., 2005) rolesets (`want-01`), or special keywords indicating generic entity types: `date-entity`, `world-region`, `distance-quantity`, etc. In addition to the PropBank lexicon of rolesets, which associate argument numbers (ARG 0–6) with predicate-specific[3] semantic roles (e.g., ARG0=*wanter* in ex. 1), AMR uses approximately 100 relations of its own (e.g., `:time`, `:age`, `:quantity`, `:destination`, etc.). These AMR-specific relations can be thought of as a fine-grained inventory of modifier role labels.

AMR abstracts away from language-specific, idiosyncratic facts, such that distinct syntactic realizations of the same basic meaning are represented with the same AMR. This includes representing related parts of speech in the same way (e.g., *describe* and *description* are both represented as `describe-01`), as well as representing light verb constructions and related lexical verb counterparts in the same way (e.g., *She made an adjustment to the machine* and *She adjusted the machine* are both represented with `adjust-01`). Because AMR annotators directly build a representation of the meaning of an utterance, they are not limited to meanings which are introduced by lexical items in the sentence, but may introduce meanings which are derived solely through the constructional semantics; for example, `cause-01` and `move-01` can be introduced in a representation for a caused motion construction (e.g., *She talked me into a corner*).

2.3. Constructions in Semantic Resources

Other semantic role labeling resources have taken on the challenge of annotating semantic roles assigned by individual predicates and constructions. Although FrameNet (Fillmore et al., 2003) has long included entries for certain light verb constructions and multi-word expressions, the more recent FrameNet Constructicon (Fillmore et al., 2012) represents an effort to extend FrameNet to capture the semantics and roles of the construction itself (i.e. the template of fixed/flexible syntactic slots). The Constructicon lists constructions, such as the Way_manner construction (e.g., *She whistled her way down the lane*), and lists the roles associated with the construction and the Construction Evoking Element (CEE) (e.g., *one's way* in this context). The roles can be thought of as the flexible slots for the construction, while the CEE is the more fixed element.

Of particular relevance to this research, the Constructicon lists a Comparison parent frame (Hasegawa et al., 2010), but indicates that the construction is always instantiated in daughter constructions, including Comparison_equality and Comparison_inequality frames. The Superlative is treated as a separate entry, but notably the roles are similar to those of Comparison, except that what is termed the Standard (the compared-to entity) in the Comparison frame is replaced by a Comparison Set. Our treatment generalizes somewhat over the FrameNet treatment by exploiting a single roleset

[2] Note that these degree-related contexts can be compatible with both gradable adjectives and what are typically thought of as ungradable adjectives, as in the humorous *He's too American to vote.*

[3] For ARG0 and ARG1 only, an effort is made to map to Dowty's prototypical agent and patient (Dowty, 1991), respectively.

for all of these constructions (see Section 5.1).

Our general roleset is very similar (in definition, albeit distinct in labeling) to that of Bakhshandeh and Allen (2015), who aim to predict the predicate-argument structure of comparison sentences to support semantic parsing (Bakhshandeh et al., 2016). Also related is the constructional annotation scheme of Dunietz et al. (2017), which targets *causal* language: in its current form their scheme includes the Degree-Consequence and The X-er, The Y-er constructions but excludes argument structure constructions like Caused Motion.

PropBank has always included multi-word verbal predicates (e.g., *eat up*), and has recently been expanding its lexicon of rolesets and annotations to include light verb constructions and the degree and quantity-related constructions that will be discussed here (Hwang et al., 2010; Bonial et al., 2014; Bonial and Palmer, 2016). The expansion of the roleset lexicon was necessary not only for PropBank to provide construction-based annotations, but also because PropBank and AMR maintain a shared lexicon of rolesets that ensures symmetry between the two projects.

3. AMR Approach to Constructions

In building the AMR corpus, we have run across a variety of cases where a predicate appeared to be in an atypical context: none of the senses listed in the lexicon of rolesets provided the appropriate role label choices for novel arguments encountered. For example, none of the PropBank rolesets for *blink* list appropriate semantic roles for the thing-moved and path arguments in *Jerry blinked the snow off of his eyelashes*. This left us with a choice of how to deal with such cases, and the options, to some extent, align either with a lexicosemantic approach or a constructional approach: either add a variety of individual rolesets to the lexicon reflecting, for example, a caused-motion sense of *blink*, or add rolesets to the lexicon for particular constructions, such as the caused motion construction.

Some of the practical ramifications of this choice had previously been researched to allow for semantic role labeling annotations of constructions in the PropBank corpus (Bonial et al., 2014). It was found that one must either add many individual rolesets for every predicate compatible with a particular construction or add a single roleset for the construction, to be used freely with all predicates (including previously unseen predicates) occurring in this context.[4] Putting theoretical arguments aside, the latter option clearly had a practical advantage of requiring far less time-consuming manual expansion of the lexicon.

AMR opted to balance both the lexicosemantic and constructional approach by employing two distinct strategies: 1) making use of existing lexical rolesets and AMR's extensive inventory of modifier roles (e.g., Source, Destination) to the greatest extent possible, while 2) adding constructional rolesets only where this alternative allows us to avoid creating numerous rolesets for all predicates compatible with a construction—thereby avoiding, for example, new entries for all gradable adjectives compatible with the comparative construction. The cases in which each strategy was employed are discussed in turn in the sections to follow.

[4]This is not to say that which predicates occur within a particular constructional context is unimportant. On the contrary, generalizing and explicitly marking the construction pattern allows for further research into which predicates (and potentially which semantic features of those predicates) can combine felicitously with the constructional semantics.

4. Exploiting Lexical Rolesets and Modifier Roles

Previously unseen usages of verbs in the ditransitive (e.g., *The same friend Facebooked me the invite*), resultative (e.g., *Sandra kissed him unconscious*), caused motion (e.g., *The crowd booed him off the stage*), and intransitive motion constructions (e.g., *The fly buzzed into the room*), are handled with the first strategy. Namely, these constructions are annotated using an existing roleset reflecting a canonical, or relatively frequent and commonly recognized, sense of the lexical predicate. Roles and semantics arguably invoked by the construction (and therefore not expected or covered in the existing lexical predicate's roleset) are then represented with existing AMR modifier roles/relations, such as Source, Destination, and Domain (expressing statehood), as well as the introduction of implicit predicates, such as Cause-01 and Move-01. For example, the sound emission sense rumble-01 provides coverage for the Arg0 "rumbler" role while AMR's Path modifier relation provides coverage for the path argument licensed by the intransitive motion construction:

Rumble-01
Arg0: entity rumbling
Arg1: sound/utterance
Arg2: hearer

2. The troops rumbled along the main road.

```
(r / rumble-01
      :ARG0 (t / troop)
      :path (a / along
            :op1 (r2 / road
                  :mod (m / main))))
```

Similarly, note the use of blink-01, meaning "close eyes for a second," in combination with implicit predicates cause-01 and move-01 in the AMR for the following caused motion construction:

Blink-01
Arg0: blinker
Arg1: eyes (usually unstated)

3. He blinked the snow off his eyelashes.

```
(b / blink-01
      :ARG0 (h/ he))
      :ARG0-of (c5 / cause-01
            :ARG1 (m2 / move-01
                  :ARG1 (s / snow)
                  :source (e / eyelash
                        :part-of h))))
```

i.e. *He blinked, the blinking caused the snow to move from his eyelashes.*

5. Expanding the Lexicon with Constructions

We chose to add constructional rolesets for a family of constructions involving degrees and quantities. This choice was motivated in part by the frequency and productivity of these constructions and by their fairly nuanced semantics, which could not adequately be captured with existing rolesets or modifier roles.

5.1. Comparative, Superlative, & Degree-Consequence Constructions

Comparative, superlative, and what we term the 'Degree-Consequence' construction are all handled with a single roleset, Have-Degree-91,[5] which is a semantically finer-grained replacement for many cases of the existing AMR modifier role, Degree. Degree will continue to be used in cases of intensifiers or downtoners (e.g., *She was a **little bit** nervous*; *Carthage was **utterly** destroyed*).

Have-Degree-91
Arg1: domain, entity characterized by attribute
Arg2: attribute (e.g. tall)
Arg3: degree itself (e.g. more/most, less/least, equal)
Arg4: compared-to
Arg5: superlative: reference to superset
Arg6: consequence, result of degree

The comparative construction licenses an additional argument of the entity that another entity is being compared to, with respect to a particular attribute. This is captured by Arg4 in the roleset, for example:

4. The girl is taller than the boy.
```
(h / have-degree-91
      :ARG1 (g / girl)
      :ARG2 (t / tall)
      :ARG3 (m / more)
      :ARG4 (b / boy))
```
i.e. *The girl is more tall compared to the boy.*

Note that the Arg3 'degree itself' may not appear explicitly in the sentence and may instead be realized by a comparative form of the adjective. Annotators are instructed to use the base form of adjectives in all cases, and introduce into the Arg3 slot *more* (as seen above), *less* (for cases such as *This book is less expensive*, or *equal* (for cases like *The girl is as tall as the boy*).

Previously, the compared-to entity (Arg4) was represented by a modifier role, Compared-to, in AMR; this role is superseded by Have-Degree-91.[6] The previous role was found to be somewhat unintuitive to annotators in its attachment position, and, in the absence of a full roleset, it failed to convey that two things were being compared with respect to a particular attribute. This led to some confusion and inconsistencies in annotation.

Although very similar to the comparative, the superlative invokes a subset/superset relation—an entity or subset of entities is compared to a superset of relevant entities with respect to some property. The reference to this superset is captured with Arg5, for example:

5. She is the tallest girl on the team.
```
(h / have-degree-91
      :ARG1 (s / she)
      :ARG2 (t / tall)
      :ARG3 (m / most)
      :ARG5 (g / girl
            :ARG0-of (h2 / have-org-role-91
                  :ARG1 (t2 / team))))
```
i.e. *She is the most tall of the girls on the team.*

Previously, superlatives were treated identically to comparatives, using the Compared-to role. This was a more superficial representation, where the Compared-to for the above example would be *team* alone, as opposed to the more precise representation wherein the tallness of one girl is being compared to the tallness of the set of girls on the team.

One of the rarer, but also more problematic, constructions involving Degree that we encountered were cases of the 'Degree-Consequence' construction. This construction licenses an argument representing the result or consequence of the degree to which a state holds. This construction was particularly problematic because annotators lacked any good way of representing the consequence argument or connecting it to the degree information, thereby leading to inconsistent and superficial treatments. The FrameNet Constructicon provides coverage for one species of this construction, which it calls the 'Degree-so' construction. While this is limited to constructions involving the word *so* (e.g., *The smell is so terrible, you want to throw up*), our definition of this construction is broader, allowing for a greater variety of degree words in this slot. The result or consequence argument is captured by Arg6:

6. The watch is too wide for my wrist.
```
(h / have-degree-91
      :ARG1 (w / watch)
      :ARG2 (w2 / wide-02
            :ARG1 w)
      :ARG3 (t / too)
      :ARG6 (f / fit-06
            :ARG1 w
            :ARG2 (w3 / wrist
                  :part-of (i / i))))
```
Note that the implicit predicate *fit* is introduced, capturing the fact that the watch is too wide[7] with respect to fitting.

[5] AMR uses a numbering convention in which rolesets introduced to the shared PropBank/AMR lexicon for the purposes of enriching AMR specifically are numbered "-91." Other rolesets in the lexicon are numbered starting with 1, increasing sequentially. This convention should not be taken to mean that there are 90 other senses of a particular relation.

[6] Retrofitting efforts for existing data are described in Section 6.

[7] The AMR in ex. 6 repeats the variable `w`, making explicit that the watch is the wide entity as well as the first item in the comparison. We only do this when the concept in the Arg2 slot (corresponding to the adjective) is itself a predicate with a roleset in the lexicon. This is not the case in ex. 5, in which *tall* does

Although we initially piloted a deeper representation that could potentially capture deontic modality, possibility, and polarity (e.g., above, including negative polarity on the fitting event, indicating that the consequence is that the watch does NOT fit), we discovered that the multitude of contexts in which this construction can be used makes adding such information too challenging for consistent annotation.[8] For example, if one says *I was too tired to drive*, context may or may not make clear whether the driving actually took place. Thus, this is one case in which practical considerations of what can be captured consistently outweighed the desire for a deeper representation.

5.2. Parallel Quantity Constructions

Paralleling Have-Degree-91 is Have-Quant-91, which is used for comparisons and superlatives relating to quantities of things as opposed to qualities or properties. Have-Quant-91 has existed in AMR in past releases to serve as the reification of the modifier role Quantity. However, the roleset was expanded and refined as we noted the many parallel cases of Degree-based constructions and Quantity-based constructions. Note the similar arguments to Have-Degree-91:

Have-Quant-91
ARG1: entity (thing being quantified)
ARG2: quantity (numerical or quantifier: many, much)
ARG3: degree mention (more, less, equal, too)
ARG4: compared-to
ARG5: superlative: reference to superset
ARG6: consequence, result

The roles of this roleset carry largely the same semantics as Have-Degree-91, applied to quantities of things, including the comparative:

7. He sold as many cars as his competitor.

```
(h / have-quant-91
      :ARG1 (c / car
            :ARG1-of (s / sell-01
                  :ARG0 (h2 / he)))
      :ARG3 (e / equal)
      :ARG4 (c3 / car
           :ARG1-of (s2 / sell-01
              :ARG0 (p / person
                    :ARG0-of (c2 / compete-02
                          :ARG1 h2)))))
```

and the 'Quantity-Consequence' construction:

8. I had too many books to carry them all.

```
(h2 / have-quant-91
      :ARG1 (b / book
            :ARG1-of (h / have-03
                  :ARG0 (i / i)))
      :ARG2 (m / many)
      :ARG3 (t / too)
      :ARG6 (c / carry-01
            :ARG0 i
            :ARG1 b))
```

5.3. Comparing Resemblance Construction

These constructions generally fit the pattern 'X [verb] more/less like Y than Z' and involve the comparison of two separate resemblances. To our knowledge, this type of construction has not received much attention in linguistic literature. After struggling and failing to find a good fit for Have-Degree-91 with the slightly distinct semantics of this construction and the arguments it licenses, we opted to introduce a new roleset, which showed reasonably high agreement in piloting—Have-Degree-of-Resemblance-91:

Have-Degree-of-Resemblance-91
Arg1: thing resembling other things
Arg2: first resemblance under comparison
Arg3: second resemblance under comparison
Arg4: degree word comparing Arg2 to Arg3

For example:

9. They dance more like the natives here than normal people.

```
(h / have-degree-of-resemblance-91
      :ARG1 (d / dance-01
            :ARG0 (t / they))
      :ARG2 (d2 / dance-01
            :ARG0 (n / native
                  :location (h2 / here)))
      :ARG3 (d3 / dance-01
            :ARG0 (p / person
                  :ARG1-of (n2 / normal-02)))
      :ARG4 (m / more))
```

i.e. *Their dancing resembles the dancing of natives here more than it resembles the dancing of normal people.*

Admittedly, this is a somewhat shallow representation of the rich semantics of this construction, which pushes off much of what is needed to interpret the sentence into the semantics of the roleset as opposed to the inventory of relations generalizable across the AMR corpus. Nonetheless, including specific guidelines for such constructions does allow them to be captured consistently, and we leave off a deeper interpretation of "resemblance" to future work.

5.4. The X-er, The Y-er Construction

A final construction requiring the introduction of an additional roleset is The X-er, The Y-er construction, otherwise known as the Covariational-Conditional construction (Fillmore et al., 1988; Culicover and Jackendoff, 1999; Goldberg, 2003). To capture the specific type of correlation expressed by this construction, we introduced a new roleset:

not yet have a corresponding roleset in the PropBank lexicon, and hence `s` (the entity that is tall) only occurs once.

[8] Even this relatively superficial representation, allowing for the introduction of implicit fit-06, may lead to disagreements regarding precisely what implicit predicate should be used.

Correlate-91
Arg1: X, degree/quant word modifying first item changing in relation to Arg2
Arg2: Y, degree/quant word modifying second item changing in relation to Arg1

A challenge in creating and implementing this roleset was determining what should head the arguments of the construction: the items changing, or the degree mentions? After piloting with an alternative roleset, we decided that the clearest and most precise roleset would focus on the degree words, since these are easily recognizable to annotators, and semantically it is the degree to which something holds that is correlated with another degree. For example:

10. The longer he is around, the more miserable I will be.

```
(c / correlate-91
      :ARG1 (m2 / more
            :ARG3-of (h2 / have-degree-91
                  :ARG1 (b / be-located-at-91
                        :ARG1 (h / he)
                        :ARG2 (a / around))
                  :ARG2 (l2 / long-03
                        :ARG1 b)))
      :ARG2 (m3 / more
            :ARG3-of (h3 / have-degree-91
                  :ARG1 (i / i)
                  :ARG2 (m / miserable))))
```

i.e. *An increase in how long he is around correlates with an increase in how miserable I am.*

6. Implementation and Evaluation

After a relatively stable draft of guidelines was decided upon and new rolesets were implemented, these guidelines were given to two annotators at two different sites. We piloted using a 'Challenge Set' of sentences selected from the AMR corpus using keyword searches followed by manual selection in an attempt to represent the variety of degree and quantity-based constructions described thus far, including tricky cases with clear inconsistencies in past annotations. On this set of 50 sentences, annotators achieved an overall agreement rate, calculated as a 'smatch' score (Cai and Knight, 2013), of 88.6%, which is a relatively high agreement rate for AMR. Although we cannot provide past agreement rates on these specific instances prior to the piloting of these guidelines (only certain portions of the AMR corpus are double-annotated), we can anecdotally report that our efforts have reduced the inconsistency and superficiality with which annotators were handling cases previously.

The bulk of the disagreements in the challenge set related to what concept served as the root or top node of the AMR annotation, with one annotator rooting the AMR with the Have-Degree-91 roleset, while another would embed this further down into the annotation, heading the AMR instead with, for example, a focal item in a comparison. Additional guidance was given to annotators on this subject in the guidelines: annotators are encouraged to use the constructional roleset as the root concept in cases where the comparison and/or correlation is focal, including cases of the copula (e.g., *She **is** the tallest and the youngest in the class*). Nonetheless, the selection of the appropriate root is necessarily somewhat subjective and remains a source of disagreement throughout other annotations as well.

Given the success in piloting, we have adopted the new guidelines and rolesets in recent annotations, and we have also completed retrofitting of past annotations. Using keyword searches over the annotations and text (e.g., 'Compared-to'), we discovered an initial set of about 4,600 annotations that potentially needed retrofitting for Have-Degree/Quant-91 (the remaining cases were found to be simple usages of degree or quantity modifiers, such as *very*, which remain unchanged), and about 30 potential Correlate-91 cases. These cases were flagged and, if needed, the annotations were updated by the team at the site where the instance was originally annotated.

7. Conclusions & Future Work

The next release of the AMR corpus in early 2018, totaling 59,783 AMRs, includes the revised annotations and additional annotated sentences completed under the new guidelines. The corpus counts of the constructions of focus here are given in Table 1.

Use Case	Roleset/Relation	Count
Downtoners, intensifiers	Degree	4547
Comparison, superlative, degree-consequence	Have-Degree-91	4943
Comparison, superlative, quantity-consequence, quantity reification	Have-Quant-91	1122
Comparing resemblances	Have-Degree-of-Resemblance-91	9
The X-er, The Y-er	Correlate-91	38

Table 1: Counts of described degree/quantity related constructions in forthcoming AMR release corpus.

Although some of the new representations fall short of what we could ideally capture (for example, the nuances of the consequence argument in Degree/Quantity-Consequence constructions), we are optimistic that these additions will increase the depth and consistency of annotations. In the future, we hope to add some of the more detailed aspectual and modal properties in a second layer of annotation, in a fashion similar to recent explorations of adding Richer Event Description (O'Gorman et al., 2016) annotation on top of AMR.

8. Bibliographical References

Bakhshandeh, O. and Allen, J. (2015). Semantic framework for comparison structures in natural language. In *Proc. of EMNLP*, pages 993–1002, Lisbon, Portugal, September.

Bakhshandeh, O., Wellwood, A., and Allen, J. (2016). Learning to jointly predict ellipsis and comparison structures. In *Proc. of CoNLL*, pages 62–74, Berlin, Germany.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Bonial, C. and Palmer, M. (2016). Comprehensive and consistent PropBank light verb annotation. In *Proc. of LREC*.

Bonial, C., Bonn, J., Conger, K., Hwang, J. D., and Palmer, M. (2014). PropBank: Semantics of new predicate types. In *Proc. of LREC*, pages 3013–3019.

Bybee, J. and McClelland, J. L. (2005). Alternatives to the combinatorial paradigm of linguistic theory based on domain general principles of human cognition. *The Linguistic Review*, 22(2-4):381–410.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proc. of ACL*, pages 748–752.

Chomsky, N. (1981). Lectures on government and binding. *Dordrecht: Foris*.

Culicover, P. W. and Jackendoff, R. (1999). The view from the periphery: The English comparative correlative. *Linguistic Inquiry*, 30(4):543–571.

Davidson, D. (1969). The individuation of events. In *Essays in honor of Carl G. Hempel*, pages 216–234. Springer.

Dowty, D. (1991). Thematic proto-roles and argument selection. *language*, pages 547–619.

Dunietz, J., Levin, L., and Carbonell, J. (2017). The BECauSE Corpus 2.0: annotating causality and overlapping relations. In *Proc. of the 11th Linguistic Annotation Workshop*, pages 95–104, Valencia, Spain, April.

Fillmore, C. J., Kay, P., and O'Connor, M. C. (1988). Regularity and idiomaticity in grammatical constructions: the case of 'let alone'. *Language*, 64(3):501–538, September.

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to FrameNet. *International journal of lexicography*, 16(3):235–250.

Fillmore, C. J., Lee-Goldman, R., and Rhodes, R. (2012). The FrameNet Constructicon. *Sign-based construction grammar*, pages 309–372.

Frege, G. (1879). *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. L. Nebert.

Garg, S., Galstyan, A., Hermjakob, U., and Marcu, D. (2016). Extracting biomolecular interactions using semantic parsing of biomedical text. In *Proc. of AAAI*, Phoenix, Arizona, USA, February.

Goldberg, A. E. (1995). *Constructions: A construction grammar approach to argument structure*. University of Chicago Press.

Goldberg, A. E. (2003). Constructions: a new theoretical approach to language. *Trends in cognitive sciences*, 7(5):219–224.

Hasegawa, Y., Lee-Goldman, R., Ohara, K. H., Fujii, S., and Fillmore, C. J. (2010). On expressing measurement and comparison in English and Japanese. In Hans C. Boas, editor, *Contrastive Studies in Construction Grammar*, pages 169–200. John Benjamins, Amsterdam.

Hopper, P. J. (1998). Emergent grammar. *The new psychology of language: Cognitive and functional approaches to language structure*, 1:155–176.

Hwang, J. D., Bhatia, A., Bonial, C., Mansouri, A., Vaidya, A., Xue, N., and Palmer, M. (2010). PropBank annotation of multilingual light verb constructions. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 82–90. Association for Computational Linguistics.

Kay, P. and Fillmore, C. J. (1999). Grammatical constructions and linguistic generalizations: the What's X doing Y? construction. *Language*, 75(1):1–33, March.

Knight, K., Badarau, B., Banarescu, L., Bonial, C., Bardocz, M., Griffitt, K., Hermjakob, U., Marcu, D., Palmer, M., O'Gorman, T., and Schneider, N. (2017). Abstract Meaning Representation (AMR) Annotation Release 2.0. Technical Report LDC2017T10, Linguistic Data Consortium, Philadelphia, PA, June.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proc. of NAACL*.

MacWhinney, B. (2001). Emergentist approaches to language. *Typological Studies in Language*, 45:449–470.

Michaelis, L. A. and Lambrecht, K. (1996). Toward a construction-based theory of language function: The case of nominal extraposition. *Language*, pages 215–247.

Mitra, A. and Baral, C. (2016). Addressing a question answering challenge by combining statistical methods with inductive rule learning and reasoning. In *Proc. of AAAI*, pages 2779–2785.

O'Gorman, T., Wright-Bettner, K., and Palmer, M. (2016). Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proc. of the 2nd Workshop on Computing News Storylines*, pages 47–56.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Pan, X., Cassidy, T., Hermjakob, U., Ji, H., and Knight, K. (2015). Unsupervised entity linking with Abstract Meaning Representation. In *Proc. of HLT-NAACL*, pages 1130–1139.

Parsons, T. (1990). *Events in the Semantics of English*, volume 5. MIT Press, Cambridge, MA.

Penman Natural Language Group. (1989). The Penman user guide. *Technical report, Information Sciences Institute*.

Pinker, S. (1989). *Resolving a learnability paradox in the acquisition of the verb lexicon.* Paul H. Brookes Publishing.

Pourdamghani, N., Knight, K., and Hermjakob, U. (2016). Generating English from Abstract Meaning Representations. In *Proc. of INLG*, pages 21–25.

Rao, S., Marcu, D., Knight, K., and Daumé III, H. (2017). Biomedical event extraction using Abstract Meaning Representation. In *Proc. of BioNLP*, pages 126–135, Vancouver, Canada, August.

Wang, Y., Liu, S., Rastegar-Mojarad, M., Wang, L., Shen, F., Liu, F., and Liu, H. (2017). Dependency and AMR embeddings for drug-drug interaction extraction from biomedical literature. In *Proc. of ACM-BCB*, pages 36–43, New York, NY, USA.

Zipf, G. K. (1949). *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, Reading, MA.

Evaluating Scoped Meaning Representations

Rik van Noord, Lasha Abzianidze, Hessel Haagsma, Johan Bos
CLCG, University of Groningen
{r.i.k.van.noord, l.abzianidze, hessel.haagsma, johan.bos}@rug.nl

Abstract

Semantic parsing offers many opportunities to improve natural language understanding. We present a semantically annotated parallel corpus for English, German, Italian, and Dutch where sentences are aligned with scoped meaning representations in order to capture the semantics of negation, modals, quantification, and presupposition triggers. The semantic formalism is based on Discourse Representation Theory, but concepts are represented by WordNet synsets and thematic roles by VerbNet relations. Translating scoped meaning representations to sets of clauses enables us to compare them for the purpose of semantic parser evaluation and checking translations. This is done by computing precision and recall on matching clauses, in a similar way as is done for Abstract Meaning Representations. We show that our matching tool for evaluating scoped meaning representations is both accurate and efficient. Applying this matching tool to three baseline semantic parsers yields F-scores between 43% and 54%. A pilot study is performed to automatically find changes in meaning by comparing meaning representations of translations. This comparison turns out to be an additional way of (i) finding annotation mistakes and (ii) finding instances where our semantic analysis needs to be improved.

Keywords: parallel corpus, semantic annotation, discourse representation structure, evaluation, semantic scope

1. Introduction

Semantic parsing is the task of assigning meaning representations to natural language expressions. Informally speaking, a meaning representation describes *who did what to whom, when, and where, and to what extent this is the case or not*. The availability of open-domain, wide coverage semantic parsers has the potential to add new functionality, such as detecting contradictions, verifying translations, and getting more accurate search results. Current research on open-domain semantic parsing focuses on supervised learning methods, using large semantically annotated corpora as training data.

However, there are not many annotated corpora available. We present a parallel corpus annotated with formal meaning representations for English, Dutch, German, and Italian, and a way to evaluate the quality of machine-generated meaning representations by comparing them to gold standard annotations. Our work shows many similarities with recent annotation and parsing efforts around Abstract Meaning Representations, (AMR; Banarescu et al., 2013) in that we abstract away from syntax, use first-order meaning representations, and use an adapted version of SMATCH (Cai and Knight, 2013) for evaluation. However, we deviate from AMR on several points: meanings are represented by scoped meaning representations (arriving at a more linguistically motivated treatment of modals, negation, presupposition, and quantification), and the non-logical symbols that we use are grounded in WordNet (concepts) and VerbNet (thematic roles), rather than PropBank (Palmer et al., 2005). We also provide a syntactic analysis in the annotated corpus, in order to derive the semantic analyses in a compositional way.

We make the following contributions:

- A meaning representation with explicit scopes that combines WordNet and VerbNet with elements of formal logic (Section 2).
- A gold standard annotated parallel corpus of formal meaning representations for four languages (Section 3).
- A tool that compares two scoped meaning representations for the purpose of evaluation (Section 4 and Section 5).

2. Scoped Meaning Representations

2.1. Discourse Representation Structures

The backbone of the meaning representations in our annotated corpus is formed by the Discourse Representation Structures (DRS) of Discourse Representation Theory (Kamp and Reyle, 1993). Our version of DRS integrates WordNet senses (Fellbaum, 1998), adopts a neo-Davidsonian analysis of events employing VerbNet roles (Bonial et al., 2011), and includes an extensive set of comparison operators. More formally, a DRS is an ordered pair of a set of variables (discourse referents) and a set of conditions. There are basic and complex conditions. Terms are either variables or constants, where the latter ones are used to account for indexicals (Bos, 2017). Basic conditions are defined as follows:

- If W is a symbol denoting a WordNet concept and x is a term, then W(x) is a basic condition;
- If V is a symbol denoting a thematic role and x and y are terms, then V(x,y) is a basic condition;
- If x and y are terms, then $x{=}y$, $x{\neq}y$, $x{\sim}y$, $x{<}y$, $x{\leq}y$, $x{\prec}y$, and $x{\bowtie}y$ are basic conditions formed with comparison operators.

WordNet concepts are represented as word.`POS`.`SenseNum`, denoting a unique synset within WordNet. Thematic roles, including the VerbNet roles, always have two arguments and start with an uppercase character. Complex conditions introduce scopes in the meaning representation. They are defined using logical operators as follows:

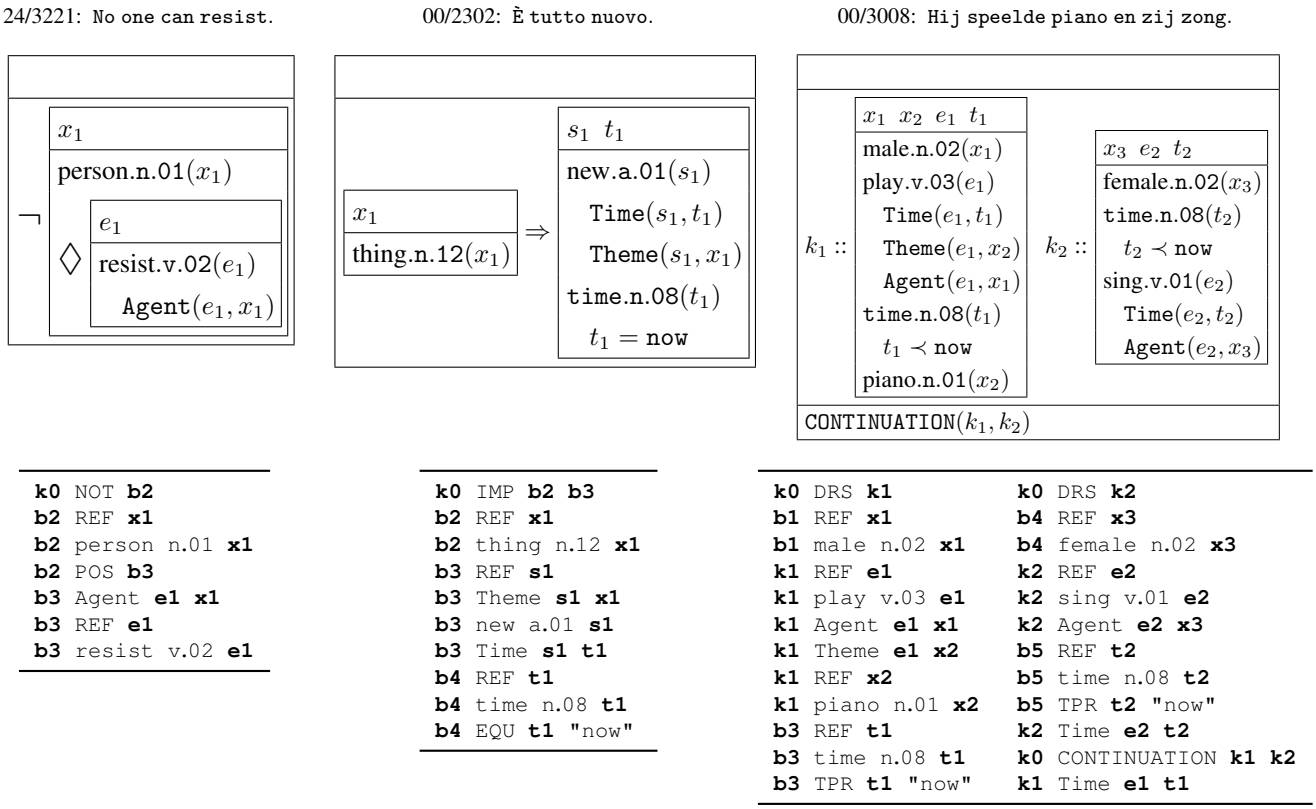


Figure 1: Examples of PMB documents with their scoped meaning representations and the corresponding clausal form. The first two structures are basic DRSs while the last one is a segmented DRS.

- If B is a DRS, then ¬B, ◇B, □B are complex conditions;
- If x is a variable, and B is a DRS, then x:B is a complex condition;
- If B and B' are DRSs, then B⇒B' and B∨B' are complex conditions.

Besides basic DRSs, we also have segmented DRSs, following Asher (1993) and Asher and Lascarides (2003). Hence, DRSs are formally defined as follows:

- If D is a (possibly empty) set of discourse referents, and C a (possibly empty) set of DRS-conditions, then <D,C> is a (basic) DRS;
- If B is a (basic) DRS, and B' a DRS, then B↓B' is a (segmented) DRS;
- If U is a set of labelled DRSs, and R a set of discourse relations, then <U,R> is a (segmented) DRS.

DRSs can be visualized in different ways. While the compact linear format saves space, the box notation increases readability. In this paper we use the latter notation. The examples of DRSs in the box notation are presented in Figure 1.

However, for evaluation and comparison purposes, we convert a DRS into a flat clausal form, i.e. a set of clauses. This is carried out by using the labels for DRSs as introduced in Venhuizen (2015) and Venhuizen et al. (2018), and breaking down the recursive structure of DRS by assigning them a label of the DRS in which they appear. Let t, t', and t" be meta-variables ranging over DRSs or terms. Let $\mathcal{C}$ be a set of WordNet concepts, $\mathcal{T}$ a set of the thematic roles, and $\mathcal{O}$ the set of DRS operators (REF, NOT, POS, NEC, EQU, NEQ, APX, LES, LEQ, TPR, TAB, IMP, DIS, PRP, DRS). The resulting clauses are then of the form t R t' or t R t' t" where R $\in \mathcal{C} \cup \mathcal{T} \cup \mathcal{O}$. The result of translating DRSs to sets of clauses is shown in Figure 1. In a clausal form, it is assumed that different variables are represented with different variable names and vice versa. Due to this, before translating a DRS to a clausal form, different discourse referents in the DRS must be represented with different variable names. This assumption significantly simplifies the matching process between clausal forms (Section 4) and makes it possible to recover the original box notation of a DRS from its clausal form.

2.2. Comparing DRSs to AMRs

Since DRSs in a clausal form come close to the triple notation of AMRs (Cai and Knight, 2013), and both aim to model meaning of natural language expressions, it is instructive to compare these two meaning representations. The main difference between AMRs and DRSs is that the latter ones have explicit scopes (boxes) and scopal operators such as negation. Due to the presence of scope in DRSs, their clauses are more complex than AMR triples. The length of DRS clauses varies from three to four, in contrast to the constant length of AMR triples. Additionally, DRS clauses contain two different types of variables, for scopes and discourse referents, whereas AMR triples have just one type.

Type	Description	Example	Total
REF	Discourse referent	`b3 REF x2`	7,592
NOT	Negation	`b1 NOT b2`	204
POS	Possibility (◇)	`b4 POS b5`	55
NEC	Necessity (□)	`b2 NEC b3`	14
IMP	Implication (⇒)	`b1 IMP b2 b3`	104
PRP	Proposition (:)	`b1 PRP x6`	50
REL	Discourse relation	`b1 CONTINUATION b2`	71
DRS	DRS as a condition	`b4 DRS b5`	84
Compare	Comparison operators	`x1 APX x2`	2,100
Concept	WordNet senses	`b2 hurt v.02 e3`	7,545
Role	Semantic roles	`b2 Agent e3 x4`	7,516

Table 1: Distribution of clause types for 2,049 gold DRSs.

Unlike AMRs, DRSs model tense. In general, the tense related information is encoded in a clausal form with three additional clauses, which express a WordNet concept, semantic role and a comparison operator. In order to give an intuition about the diversity of clauses in DRSs, Table 1 shows a distribution of various types of clauses in a corpus of DRSs (see Section 3). Since every logical operator carries a scope, their number represents a lower bound of the number of scopes in the meaning representations. In addition to logical operators, scopes are introduced by presupposition triggers like proper names or pronouns.

To make a meaningful comparison between AMRs and DRSs in terms of size, we compare the DRSs of 250,000 English sentences from the Parallel Meaning Bank (PMB; Abzianidze et al., 2017) to AMRs of the same sentences, produced by the state-of-the-art AMR parser from van Noord and Bos (2017). Statistics of the comparison are shown in Figure 2. On average, DRSs are about twice as large as AMRs, in terms of the number of clauses as well as the number of unique variables. This is obviously due to the explicit presence of scope in the meaning representation. However, for both meaning representations the number of clauses and variables increase linearly with sentence length.

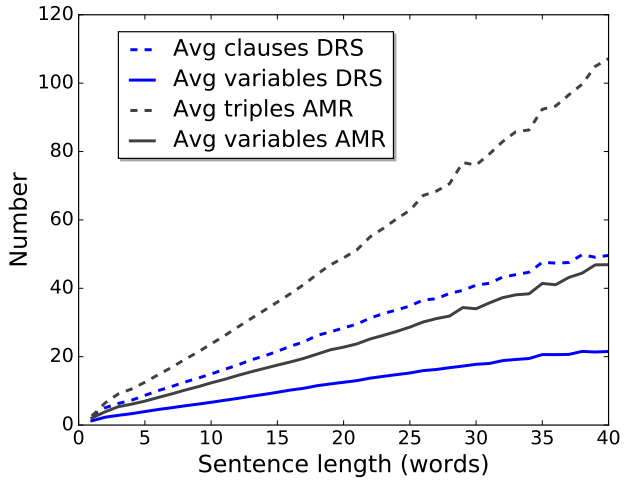


Figure 2: Comparison of the number of triples/clauses and variables between AMRs and DRSs for sentences of different length.

	Documents	Sentences	Tokens
English	2,049	2,057	11,664
German	641	642	3,430
Italian	387	387	1,944
Dutch	394	395	2,268

Table 2: Statistics of the first PMB release.

3. The Parallel Meaning Bank

The scoped meaning representations, integrating word senses, thematic roles, and the list of operators, form the final product of our semantically annotated corpus: the Parallel Meaning Bank. The PMB is a semantically annotated corpus of English texts aligned with translations in Dutch, German and Italian (Abzianidze et al., 2017). It uses the same framework as the Groningen Meaning Bank (Bos et al., 2017), but aims to abstract away from language-specific annotation models. There are five annotation layers present in the PMB: segmentation of words, multi-word expressions and sentences (Evang et al., 2013), semantic tagging (Bjerva et al., 2016; Abzianidze and Bos, 2017), syntactic analysis based on CCG (Lewis and Steedman, 2014), word senses based on WordNet (Fellbaum, 1998), and thematic role labelling (Bos et al., 2012). The semantic analysis for English is projected on the other languages, to save manual annotation efforts (Evang, 2016; Evang and Bos, 2016). All the information provided by these layers is combined into a single meaning representation using the semantic parser Boxer (Bos, 2015), in the form of Discourse Representation Structures. Note that the goal is to produce annotations that capture the most probable interpretation of a sentence; no ambiguities or under-specification techniques are employed.

At each step in this pipeline, a single component produces the automatic annotation for all four languages, using language-specific models. Human annotators can correct machine output by adding 'Bits of Wisdom' (Basile et al., 2012). These corrections serve as data for training better models, and create a gold standard annotated subset of the data. Annotation quality is defined per layer and language, at three levels: bronze (fully automatic), silver (automatic with some manual corrections), and gold (fully manually checked and corrected). If all layers are marked as gold, it follows that the resulting DRS can be considered gold standard, too.

The first public release[1] of the PMB contains gold standard scoped meaning representations for over 3,000 sentences in total (see Table 2). The release includes mainly relatively short sentences involving several semantic scope phenomena. A detailed distribution of clause types in the dataset is given in Table 1. A larger amount of texts and more complex linguistic phenomena will be included in future releases.

In addition to the released data, the PMB documents are publicly accessible through a web interface, called the PMB explorer.[2] In the explorer, visitors can view natural lan-

[1] http://pmb.let.rug.nl/data.php
[2] http://pmb.let.rug.nl/explorer

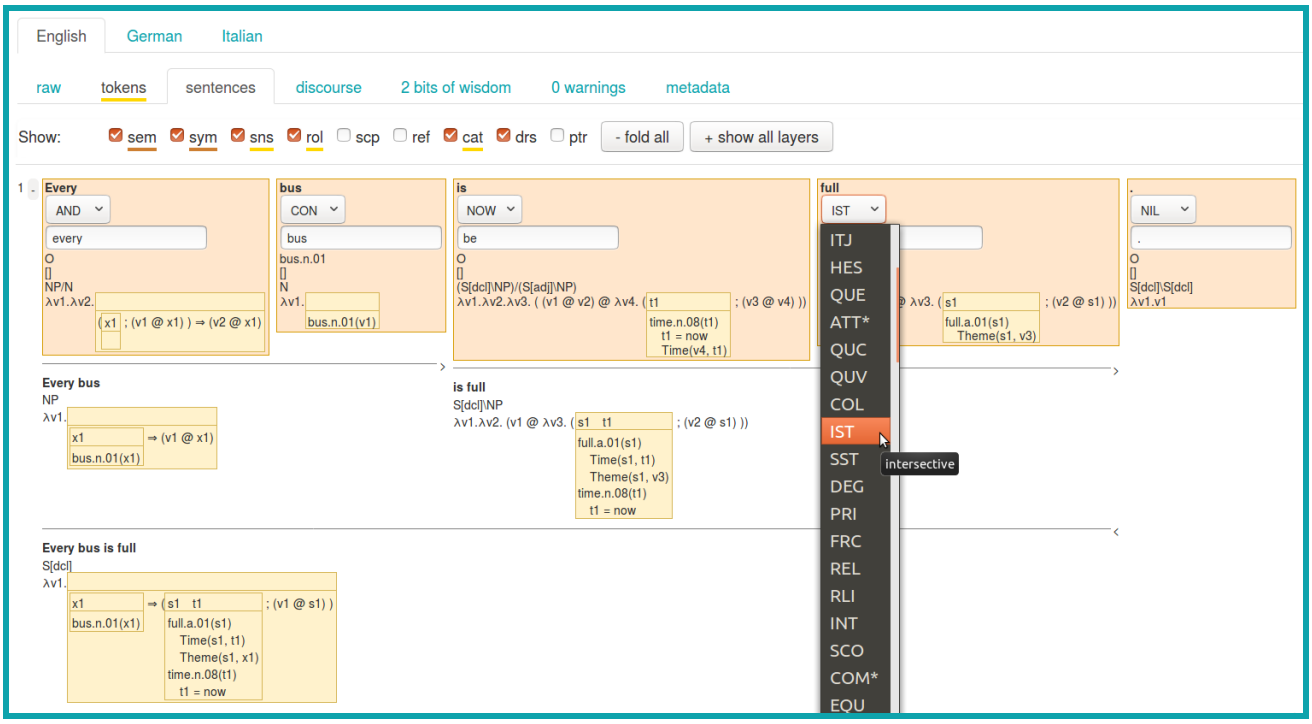


Figure 3: The edit mode of the PMB explorer: semantic tag (`sem`) and symbol (`sym`) layers of the document are bronze and therefore editable, while the word sense (`sns`), semantic role (`rol`) and CCG category (`cat`) layers are gold and uneditable.

guage texts with several layers of annotations and compositionally derived meaning representations, and, after registration, edit the annotations. It is also possible to use a word or a phrase search to find certain words or constructions with their semantic analyses. Figure 3 shows the PMB explorer with the semantic analysis of a sentence in the edit mode.

4. Matching Scoped Representations

4.1. Evaluation by Matching

In the context of the Parallel Meaning Bank there are two main reasons to verify whether two scoped meaning representations capture the same meaning or not: (1) to be able to evaluate semantic parsers that produce scoped meaning representations by comparing gold-standard DRSs to system output; and (2) to check whether translations are meaning-preserving; a discrepancy in meaning between source and target could indicate a mistranslation.

The ideal way to compare two meaning representations would be one based on inference. This can be implemented by translating DRSs to first-order formulas and using an off-the-shelf theorem prover to find out whether the two meanings are logically equivalent (Blackburn and Bos, 2005). This method can compare meaning representation that have different syntactic structures but still are equivalent in meaning. The disadvantage of this approach is that it yields just a binary answer: if a proof is found the meanings are the same, else they are not.

An alternative way of comparing meaning representations is comparing the corresponding clausal forms by computing precision and recall over matched clauses (Allen et al., 2008). The advantage of this approach is that it returns a score between 0 and 1, preferring meaning representations that better approximate the gold standard over those that are completely different. Since the variables of different clausal forms are independent from each other, the comparison of two clausal forms boils down to finding a (partial) one-to-one variable mapping that maximizes intersection of the clausal forms. For example, the maximal matching for the clausal forms in Figure 4 is achieved by the following partial mapping from the variables of the left form into the variables of the right one: $\{\texttt{k0}\mapsto\texttt{b0}, \texttt{e1}\mapsto\texttt{v1}\}$.

For AMRs, finding a maximal matching is done using a hill-climbing algorithm called SMATCH (Cai and Knight, 2013). This algorithm is based on a simple principle: it checks if a single change in the current mapping results in a better matching mapping. If this is the case, it continues with the new mapping. Otherwise, the algorithm stops and has arrived at the final mapping. This means that it can easily get stuck in local optima. To avoid this, SMATCH does a predefined number of restarts of this process, where each restart starts with a new and random initial mapping. The first restart always uses a 'smart' initial mapping, based on matching concepts.

Our evaluation system, called COUNTER[3], is a modified version of SMATCH. Even though clausal forms do not form a graph and clauses consist of either three or four components, the principle behind the variable matching is the same. The actual implementation differs, mainly because SMATCH was not designed to handle clauses with three variables, e.g. $\langle$`k0 Agent e1 x1`$\rangle$.

In contrast to SMATCH, COUNTER takes a set of clauses directly as input. COUNTER also uses two smart initial map-

[3] `http://github.com/RikVN/DRS_parsing/`

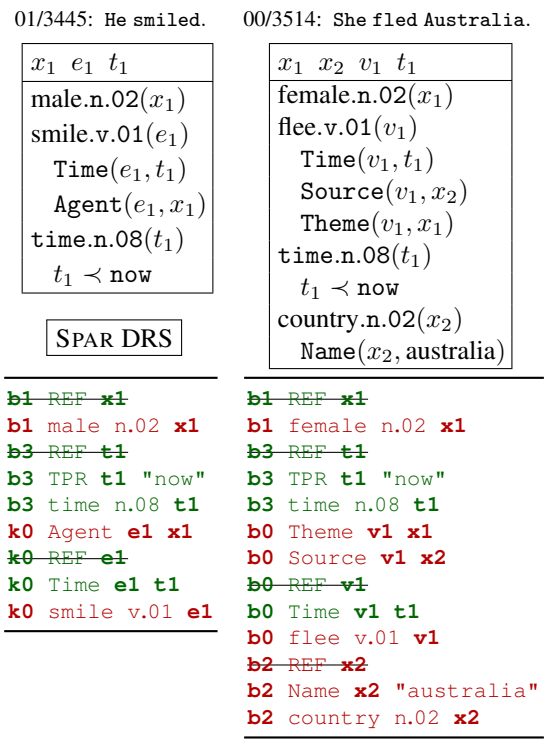


Figure 4: The SPAR DRS (Section 5.1) matches the DRS of 00/3514 PMB document with an F-score of 54.5%. If redundant REF-clauses are ignored, the F-score drops to 40%. These results are achieved with the help of the mapping {k0↦b0, e1↦v1}.

pings, based on either role-clauses, like ⟨k0 Agent e1 x1⟩, or concept-clauses, like ⟨k0 smile v.01 e1⟩. Also specific to this method is the treatment of REF-clauses in the matching process. Before matching two DRSs, redundant REF-clauses are removed. A REF-clause ⟨b1 REF x1⟩ is redundant if its discourse referent x1 occurs in some basic condition of the same DRS b1. Figure 4 shows some examples of redundant REF-clauses. Not removing these redundant clauses would lead to inflated matching scores since for each matched variable the corresponding REF-clause will also match. Comparison of the clausal forms in Figure 4 demonstrates this fact. Note that not all REF-clauses are redundant: if a discourse referent is declared outside the scope of negation or an other scope operator, the REF-clause is kept. This is very infrequent in our data, since only a single REF-clause was preserved in 2,049 examples.

4.2. Evaluating Matching

As we showed in Figure 2, DRSs are about twice as large as AMRs. This increase in size might be problematic, since it increases the average runtime for comparing DRSs. Moreover, if there are more variables, more restarts might be needed to ensure a reliable score, again increasing runtime. Therefore, our goal is that COUNTER gets close to optimal performance in reasonable time. Since we want to be sure that this also holds for longer sentences, we use a balanced data set. We take 1,000 DRSs produced by the semantic parser Boxer for each sentence length from 2 to 20 (punctuation excluded), resulting in a set of 19,000 DRSs.

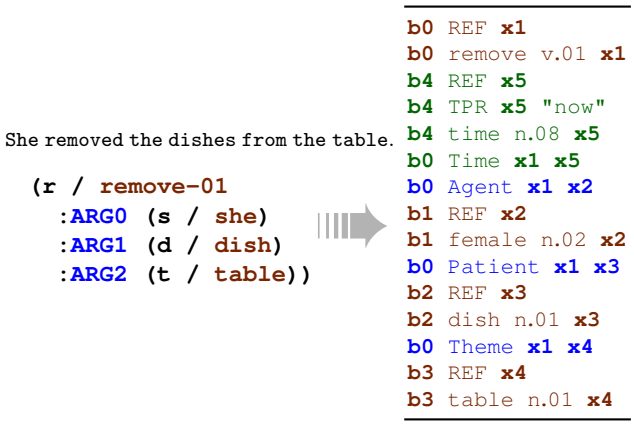


Figure 5: A clausal form obtained from an automatically generated AMR of the document 14/0849.

To test COUNTER in a realistic setting, we cannot compare the DRSs to themselves or to a DRS of the translation, since those are too similar. Therefore, the 19,000 English sentences of the DRS are parsed by an existing AMR parser (van Noord and Bos, 2017) and subsequently converted into a DRS by a rule-based system, AMR2DRS, as motivated by Bos (2016). An example of translating an AMR to a clausal form of a DRS is shown in Figure 5. We convert AMR relations to DRS roles by employing a manually created translation dictionary, including rules for semantic roles (e.g. :ARG0 ↦ Agent and :ARG1 ↦ Patient) and pronouns (e.g. she ↦ female.n.02). Since AMRs do not contain tense information, past tense clauses[4] are produced for the first verb in the AMR (see four tense related clauses in Figure 5). Also, since AMRs do not use WordNet synsets, all concepts get a default first sense, except for concepts that are added by concept-specific rules, such as female.n.02 and time.n.08.

We compare the sets of DRSs using different numbers of restarts to find the best trade-off between speed and accuracy. The results are shown in Table 3. The optimal scores are obtained using a Prolog script that performs an exhaustive search for the optimal mapping. As expected, increasing the number of restarts benefits performance. Cai and Knight (2013) consider four restarts the optimal trade-off between accuracy and speed, showing no improvement in F-score when using more than ten restarts.[5] Contrary to SMATCH, performance for COUNTER still increases with more than 4 restarts. In our case, it is a bit harder to select an optimal number of restarts, since this number depends on the length of the sentence, as shown in Figure 6. We see that for long sentences, 5 and 10 restarts are not sufficient to get close to the optimal, while for short sentences 5 restarts might be considered enough. In general, the best trade-off between speed and accuracy is approximately 20 restarts.

[4] Past tense was chosen because it is the most frequent tense in the data set.

[5] However, we found that, in practice, SMATCH still improves when using more restarts. Parsing the development set of the AMR dataset LDC2016E25 with the baseline parser of van Noord and Bos (2017) yields an F-score of 55.0 for 10 restarts, but 55.4 for 100 restarts.

Restarts	P%	R%	F1%	Time (h:m:s)
(random) 1	27.20	22.71	24.75	4:19
(smart concepts) 1	27.45	22.92	24.98	4:35
(smart roles) 1	27.27	22.76	24.81	4:37
5	30.25	25.25	27.53	19:33
10	30.65	25.59	27.89	37:08
20	30.84	25.75	28.07	1:10:13
30	30.90	25.80	28.12	1:41:43
50	30.94	25.83	28.16	2:41:38
75	30.96	25.85	28.17	3:53:01
100	30.97	25.85	28.18	5:01:25
Optimal	30.98	25.86	28.19	

Table 3: Results of comparing 19,000 Boxer-produced DRSs to DRSs produced by AMR2DRS, for different number of restarts. For three or more restarts, we always use the smart role and concept mapping.

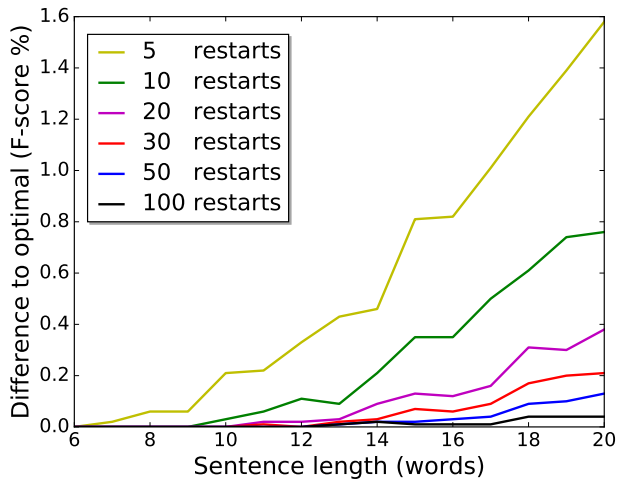


Figure 6: Comparison of the differences to the optimal F-score per sentence length for different number of restarts.

5. COUNTER in Action

5.1. Semantic Parsing

The first purpose of COUNTER is to evaluate semantic parsers for DRSs. Since this is a new task, there are no existing systems that are able to do this. Therefore, we show the results of three baseline systems PMB PIPELINE, SPAR, and AMR2DRS (Subsection 4.2).[6]

The PMB PIPELINE produces a DRS via the pipeline of the tools used for automatic annotation of the PMB.[7] This means that it has no access to manual corrections, and hence it uses the most frequent word senses and default VerbNet roles. SPAR is a trivial semantic 'parser' which always outputs the DRS that is most similar to all other DRSs in the most recent PMB release (the left-hand DRS in Figure 4).

The results of the three baseline parsers are shown in Table 4. The surprisingly high score of SPAR is explained by the fact that the first PMB release mainly contains relatively short sentences with little structural diversity. The average number of clauses per clausal form (excluding redundant REF-clauses) is 8.7, where a substantial share (approximately 3) comes from tense related clauses. Due to this fact, guessing temporal clauses for short sentences has a big impact on F-score. This is illustrated by the comparison of the clausal forms in Figure 4, where matching only temporal clauses results in an F-score of 40%.

[6] SPAR and AMR2DRS are available at: https://github.com/RikVN/DRS_parsing/

[7] http://pmb.let.rug.nl/software.php

	Precision%	Recall%	F-score%
SPAR	53.1	36.6	43.3
AMR2DRS	46.5	48.2	47.3
PMB PIPELINE	53.0	54.8	53.9

Table 4: Comparison of three baseline DRS parsers to the gold-standard data set.

AMR2DRS outperforms SPAR by a considerable margin, but is still far from optimal. This is also the case for PMB PIPELINE, which shows that, within the PMB, manual annotation is still required to obtain gold standard meaning representations.

5.2. Comparing Translations

The second purpose of COUNTER is checking whether translations are meaning-preserving. As a pilot study, we compare the gold standard meaning representations of German, Italian and Dutch translations in the release to their English counterparts. The results are shown in Table 5. The high F-scores indicate that the meaning representations are often syntactically very similar, if not identical. However, there is a considerable subset of meaning representations which are different from the English ones, indicating that there is at least a slight discrepancy in meaning for those translations.

	F-score%	Docs	F<1.0	% total
German	98.4	579	61	10.5
Italian	97.6	341	46	13.5
Dutch	98.3	355	37	10.4

Table 5: Comparing meaning representations of English texts to those of German, Italian and Dutch translations.

Manual analysis of these discrepancies showed that there are several different causes for a discrepancy to arise. In most of the cases (38%), a human annotation error was made. In 34% of cases, a definite description was used in one language but not in the other. Examples are 'has long hair' with the Italian translation 'ha i capelli lunghi', and 'escape from prison' with the Dutch translation 'vluchtte uit de gevangenis'. In 15% of cases proper names were translated (e.g. 'United States' and 'Stati Uniti'). This is not accounted for, since we do not currently make use of grounding proper names to a unique identifier, for instance by wikification (Cucerzan, 2007), or by using a language-independent transliteration of names. In 13% of cases the translation was either non-literal or incorrect. Examples are 'Tom lacks experience' with the Dutch translation 'Tom

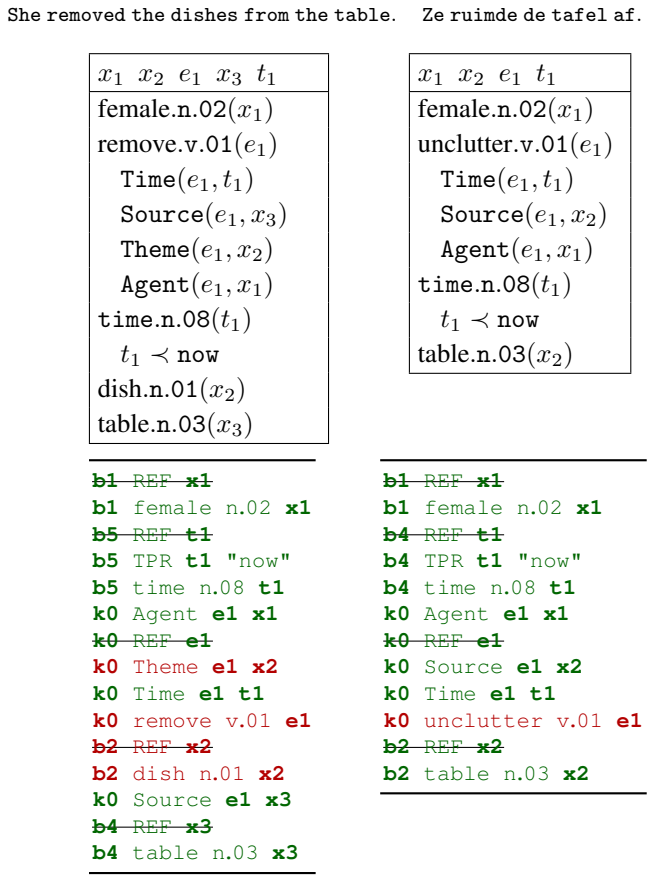


Figure 7: English and Dutch non-literal translations of the document 14/0849. Their clausal forms match each other (excl. redundant REF-clauses) with an F-score of 77.8%. This matching is achieved by the mapping of variables {b5↦b4, b4↦b2}.

heeft geen ervaring' (lit. 'Tom has no experience'), 'can't use chopsticks' with the German 'kann nicht mit Stäbchen essen' (lit. 'cannot eat with sticks'), and 'remove the dishes from the table' with the Dutch translation 'ruimde de tafel af' (lit. 'uncluttered the table').

The mapping of clausal forms involving non-literal translations is illustrated in Figure 7. This preliminary analysis shows that this comparison of meaning representations provides an an additional method for detecting mistakes in annotation. It also showed that there are cases where our semantic analysis needs to be revised and improved.

6. Conclusions and Future Work

Large semantically annotated corpora are rare. Within the Parallel Meaning Bank project, we are creating a large, open-domain corpus annotated with formal meaning representations. We take advantage of parallel corpora, enabling the production of meaning representations for several languages at the same time. Currently, the these are languages similar to English, two Germanic languages (Dutch and German) and one Romance language (Italian). Ideally, future work would include more non-Germanic languages.

The DRSs that we present are meaning representations with substantial expressive power. They deal with negation, universal quantification, modals, tense, and presupposition. As a consequence, semantic parsing for DRSs is a challenging task. Compared to Abstract Meaning Representations, the number of clauses and variables in a DRS is about two times larger on average. Moreover, compared to AMRs, DRSs rarely contain clauses with single variables. All non-logical symbols used in DRSs are grounded in WordNet and VerbNet (with a few extensions). This makes evaluation using matching computationally challenging, in particular for long sentences, but our matching system COUNTER achieves a reasonable trade-off between speed and accuracy.

Several extensions to the annotation scheme are possible. Currently, the DRSs for the non-English languages contain references to synsets of the English WordNet. Conceptually, there is nothing wrong with this (as synsets can be viewed as identifiers for concepts that are language-independent), but for practical reasons it makes more sense to provide links to synsets of the original language (Hamp and Feldweg, 1997; Postma et al., 2016; Roventini et al., 2000; Pianta et al., 2002). In addition, we consider implementing semantic grounding such as wikification in the Parallel Meaning Bank.

As for other future work, we plan to include a more fine-grained matching regarding WordNet synsets, since the current evaluation of concepts is purely string-based, with only identical strings resulting in a matching clause. For many synsets, however, it is possible to refer to them with more than one word.POS.SenseNum triple, and this should be accounted for (e.g. fox.n.02 and dodger.n.01 both refer to the same synset). In a similar vein, we plan to experiment with including WordNet concept similarity techniques in COUNTER to compute semantic distances between synsets, in case they do not fully match.

Finally, we would like to stimulate research on semantic parsing with scoped meaning representations. Not only are we planning to extend the coverage of phenomena and the number of texts with gold-standard meaning representations for the four languages, we also aim to organize a shared task on DRS parsing for English, German, Dutch and Italian in the near future.

7. Bibliographical References

Abzianidze, L. and Bos, J. (2017). Towards universal semantic tagging. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS 2017) – Short Papers*, Montpellier, France, September. Association for Computational Linguistics.

Abzianidze, L., Bjerva, J., Evang, K., Haagsma, H., van Noord, R., Ludmann, P., Nguyen, D.-D., and Bos, J. (2017). The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain, April. Association for Computational Linguistics.

Allen, J. F., Swift, M., and de Beaumont, W. (2008). Deep Semantic Analysis of Text. In Johan Bos et al., editors, *Semantics in Text Processing. STEP 2008 Conference*

Proceedings, volume 1 of *Research in Computational Semantics*, pages 343–354. College Publications.

Asher, N. and Lascarides, A. (2003). *Logics of conversation*. Studies in natural language processing. Cambridge University Press.

Asher, N. (1993). *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers.

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria.

Basile, V., Bos, J., Evang, K., and Venhuizen, N. (2012). A platform for collaborative semantic annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012)*, pages 92–96, Avignon, France.

Bjerva, J., Plank, B., and Bos, J. (2016). Semantic tagging with deep residual networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3531–3541, Osaka, Japan.

Blackburn, P. and Bos, J. (2005). *Representation and Inference for Natural Language. A First Course in Computational Semantics*. CSLI.

Bonial, C., Corvey, W. J., Palmer, M., Petukhova, V., and Bunt, H. (2011). A hierarchical unification of LIRICS and VerbNet semantic roles. In *Proceedings of the 5th IEEE International Conference on Semantic Computing (ICSC 2011)*, pages 483–489.

Bos, J., Evang, K., and Nissim, M. (2012). Annotating semantic roles in a lexicalised grammar environment. In *Proceedings of the Eighth Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-8)*, pages 9–12, Pisa, Italy.

Bos, J., Basile, V., Evang, K., Venhuizen, N., and Bjerva, J. (2017). The Groningen Meaning Bank. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer.

Bos, J. (2015). Open-domain semantic parsing with Boxer. In Beáta Megyesi, editor, *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 301–304.

Bos, J. (2016). Expressive power of Abstract Meaning Representations. *Computational Linguistics*, 42(3):527–535.

Bos, J. (2017). Indexicals and compositionality: Inside-out or outside-in? In *Proceedings of the 12th International Conference on Computational Semantics (IWCS 2017) – Short Papers*, Montpellier, France, September. Association for Computational Linguistics.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752, Sofia, Bulgaria, August. Association for Computational Linguistics.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716, Prague, Czech Republic. Association for Computational Linguistics.

Evang, K. and Bos, J. (2016). Cross-lingual learning of an open-domain semantic parser. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 579–588, Osaka, Japan.

Evang, K., Basile, V., Chrupała, G., and Bos, J. (2013). Elephant: Sequence labeling for word and sentence segmentation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1422–1426, Seattle, Washington, USA.

Evang, K. (2016). *Cross-lingual Semantic Parsing with Categorial Grammars*. Ph.D. thesis, University of Groningen.

Christiane Fellbaum, editor. (1998). *WordNet. An Electronic Lexical Database*. The MIT Press, Cambridge, Ma., USA.

Hamp, B. and Feldweg, H. (1997). GermaNet - a lexical-semantic net for German. In *In Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic; An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer, Dordrecht.

Lewis, M. and Steedman, M. (2014). A* CCG parsing with a supertag-factored model. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 990–1000, Doha, Qatar.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1).

Pianta, E., Bentivogli, L., and Girardi, C. (2002). MultiWordNet: developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*, pages 293–302.

Postma, M., van Miltenburg, E., Segers, R., Schoen, A., and Vossen, P. (2016). Open Dutch WordNet. In *Proceedings of the Eight Global Wordnet Conference*, Bucharest, Romania.

Roventini, A., Alonge, A., Calzolari, N., Magnini, B., and Bertagna, F. (2000). ItalWordNet: a large semantic database for Italian. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2000)*, pages 783–790.

van Noord, R. and Bos, J. (2017). Neural semantic parsing by character-based translation: Experiments with Abstract Meaning Representations. *Computational Linguistics in the Netherlands Journal*, 7:93–108.

Venhuizen, N. J., Bos, J., Hendriks, P., and Brouwer, H. (2018). Discourse semantics with information structure. *Journal of Semantics*.

Venhuizen, N. J. (2015). *Projection in Discourse: A data-driven formal semantic analysis*. Ph.D. thesis, University of Groningen.

Huge Automatically Extracted Training Sets for Multilingual Word Sense Disambiguation

Tommaso Pasini, Francesco Elia, Roberto Navigli
Sapienza University of Rome
{pasini, elia, navigli}@di.uniroma1.it

Abstract

We release to the community six large-scale sense-annotated datasets in multiple language to pave the way for supervised multilingual Word Sense Disambiguation. Our datasets cover all the nouns in the English WordNet and their translations in other languages for a total of millions of sense-tagged sentences . Experiments prove that these corpora can be effectively used as training sets for supervised WSD systems, surpassing the state of the art for low-resourced languages and providing competitive results for English, where manually annotated training sets are accessible. The data is available at `trainomatic.org`.

Keywords: Multilingual Word Sense Disambiguation, Resource, Dataset

1. Introduction

Word Sense Disambiguation is a crucial task in Natural Language Processing as it can be beneficial to several downstream applications, i.e., natural language understanding, semantic parsing and question answering. Despite the task has been around for a long time, it is far from being solved as it presents several challenges that have not fully been addressed yet, starting from the theoretical difficulty of formally establishing what a "word sense" is and choosing a corresponding sense inventory to the more pragmatic problems of finding large-scale sense-annotated corpora to train supervised systems on. Although WordNet (Fellbaum, 1998) virtually solved the first problem at least for English, a wide range of other issues still remain open. In fact, since supervised WSD systems need to be trained on a word-by-word basis, creating effective datasets requires a huge effort, which is beyond reach even for resource-rich languages like English. Clearly, this issue is even more severe for systems that need both lexicographic and encyclopedic knowledge (Schubert, 2006) and/or need to work in a multilingual or domain-specific setting. Knowledge-based WSD, on the other hand, exploits the knowledge contained in resources like WordNet to build algorithms (e.g. densest subgraph (Moro et al., 2014) or personalized page rank (Agirre and Soroa, 2009)) that can choose the sense of a word in context, thus not requiring training data but usually adopting bag-of-words approaches that neglect the lexical and syntactic context of the word (information that is more easily exploited by supervised systems), which may be essential in some scenarios. Furthermore, performances of both types of systems are highly affected by distribution of word senses that are usually different for each domain of application (Pasini and Navigli, 2018).

In order to address these issues different solutions have been proposed in the past years, ranging from manually annotated resources that can be used to train WSD systems to automatic or semi-automatic approaches that aim at exploiting parallel corpora or partially annotated data in order to produce training corpora. One of the first attempt to produce a sense annotated corpus is SemCor (Miller et al., 1993), a collection of thousand sentences manually tagged with WordNet senses. While its quality is very high thanks to the effort of specialized annotators, it is far from covering the whole English vocabulary of words and senses. Moreover, such manual resources need extra effort to be maintained and updated to integrate new senses and words appearing in everyday language. Thus, in order to overcome these issues, semi-automatic or fully automatic approaches have been proposed over the past years.

Taghipour and Ng (2015) exploit a parallel corpus and the manual translations of senses to annotate the words in the corpus with senses. Similarly, but without the need for human intervention, Delli Bovi et al. (2017) and Camacho-Collados et al. (2016), rely on aligned sentences in order to create a richer context that can be beneficial to their disambiguation. Raganato et al. (2016), instead, designed a set of heuristics which exploit the human effort of the Wikipedia community in order to propagate and add sense annotations to the Wikipedia pages. Similarly Pasini and Navigli (2017) exploit a knowledge base in order to annotate sentences with sense tags and uses a measure of confidence in order to select the most correct annotated sentences. They show that, relying on a multilingual semantic network as the underlying knowledge base, they are able to create high-quality sense-tagged corpora for any languages supported by the semantic network.

Our work builds upon that of Pasini and Navigli (2017) in order to generate sense-tagged corpora for 5 major European languages (English, French, German, Spanish and Italian) and the most spoken language of Asia (Chinese) and paves the way for supervised Word Sense Disambiguation in multiple languages. Exploiting the knowledge contained in BabelNet (Navigli and Ponzetto, 2010; Navigli and Ponzetto, 2012) – a huge and multilingual semantic network containing both lexicographic and encyclopedic knowledge – and Wikipedia, we generated large corpora annotated with BabelNet senses for the 6 languages listed above.

Experiments and statistics prove that these automatically created corpora are rich in terms of number of different lemmas annotated with a sense and number of sentences, and as such they can be a valuable resource for supervised WSD systems: in fact, systems trained on our datasets perform better or comparably to the state of the art across different languages. The added value is even more visible on low-resourced languages where such data is very scarce, if

at all available. We now give an overview of our corpus building procedure, including a brief description of Train-o-Matic; we then discuss features of the created datasets, our experimental setup for evaluation and its results.

2. Building the corpus

In order to build a sense annotated corpus for a given language L, our system takes as input a corpus of raw sentences C in the language L, a list of words W_L in the target language L and a semantic network G[1]. For each language L we apply (Pasini and Navigli, 2017)[Train-o-Matic] in order to annotate each target word $w \in W_L$ with a distribution over its senses.

For example given the ambiguous sentence *"A match is a lighter."* and the target word *"match"*, Train-o-Matic will output a sense distribution of the target word similar to the following:

$$[\text{match}_n^1 : 0.74, \text{match}_n^2 : 0.16, \text{match}_n^3 : 0.10]$$

where $word_{pos}^n$ follows the notation introduced in (Navigli, 2009) to indicate the *n-th* WordNet sense of $word$ with Part-of-Speech pos.

We chose Wikipedia in the language L as raw corpus C_L and BabelNet as the underlying semantic graph G because both are available for all the 6 languages of interest. BabelNet is also exploited in order to generate the lexicon W_L for each language L by collecting all the lexicaliztions of a synset in the graph in the given language L. Given the size of BabelNet we chose not to include all of its synsets, limiting our graph only to those that contain at least a sense from WordNet. We choose to keep all the BabelNet edges because they add many syntagmatic relations on top of the manually curated paradigmatic edges of WordNet.

To build each corpus we select all the sentences in each Wikipedia that contain at least one of the target words in W_L and then apply Train-o-Matic.

2.1. Train-o-Matic Overview

Train-o-Matic is a 3-step method to annotate a raw corpus of sentences.

1. Lexical Profiling Train-o-Matic exploits the semantic graph G in order to generate a lexical profile for each of the synsets in G. Such profile is computed by running the Personalized PageRank algorithm (Brin and Page, 1998) for each node in the graph. This means that, given the following formula:

$$v^{(t+1)} = (1-\alpha)v^{(0)} + \alpha M v^{(t)} \quad (1)$$

we set a 1 in the probability distribution v to the component that corresponds to the node for which we want to build the lexical profile. This procedure can also be interpreted as a random walk on the graph G where the walk is always restarted from the same initial node.

At the end of this step each synset s (i.e. node) in the graph has an associated vector in which each component represents another synset s' in the graph and the value of the component expresses the probability of reaching s' from s; this probability can be interpreted as a measure of relatedness between s and s'.

2. Sentence Scoring Once we have a distribution over the most related concepts for each synset in the graph, Train-o-Matic exploits them in order to annotate each target word in the raw corpus. For example, given the target word $w =$ "match", its set of senses retrieved from the semantic network $S_{\text{match}} = [match_n^1, match_n^2]$ and the sentence *"Messi didn't play the last match."* which contains the target word, the system creates a distribution over the senses in S_{match}.

This is done by approximating the probability of a sense given the target word and the sentence as follows:

$$P(s|\sigma, w) = \frac{P(\sigma|s,w)P(s|w)}{P(\sigma|w)} \quad (2)$$

$$\approx P(w_1|s,w)\dots P(w_n|s,w)P(s|w) \quad (3)$$

which assumes the independence of the words and removes the constant denominator. Each probability in (3) is computed exploiting the vectors previously computed. In fact, grounding the formula on our example, we have:

$$P(\text{match}_n^1|\text{Messi didn't play the last match, match}) = \quad (4)$$

$$P(\text{match}|\text{match}_n^1, \text{match})\times \quad (5)$$

$$P(\text{play}|\text{match}_n^1, \text{match})\times \quad (6)$$

$$P(\text{Messi}|\text{match}_n^1, \text{match}) \quad (7)$$

and each individual probability for the words w_i is computed by taking the value of the synset with the highest probability in the lexical profile of $match_n^1$ that contains the lemma w_i.

3. Sentence Ranking The last step aims at sorting and removing the sentences which are less likely to be correctly tagged. The sentences are in fact ranked by a confidence score which is computed by considering the difference between the most likely and second most likely senses of the target word. For example, referring to the previous example sentence, if match_n^1 received a probability of .7 and match_n^2 one of .3 then the sentence score will be .4. For each sense of a given word w, the candidate sentences are sorted using the confidence score. In order to select how many sentences to include in total, we set a parameter K that represents how many sentences must be included for the first sense of the given target word (i.e., the most common sense), with subsequent senses (according to the BabelNet ordering) for the same word receiving a decreasing number of examples computed according to a Zipf's distribution.

The following formula better explains the computation of the number of sentences assigned to each sense in a given ordering o.

$$\text{examples}_s = K \times \frac{1}{index(o,s)^z}$$

where $index(o, s)$ is a function that returns the position of a synset s in the ordering o. So, for example, if K is set to

[1]We consider a WordNet-like structure of the semantic network, where the nodes are synsets (concepts) which contain a set of lemmas that can express that concept.

	Total	English	French	German	Italian	Spanish	Chinese
Number of Annotations	17,987,488	12,722,530	1,597,230	1,213,634	1,037,253	935,713	481,128
Distinct lemmas covered	146,068	51,395	25,689	22,300	19,192	14,596	12,896
Distinct senses covered	63,613	56,229	33,843	23,526	22,587	21,388	12,485
Average # of sentences per sense	75.5	226.3	47.2	51.6	45.9	43.7	38.5
Average confidence score	56.74	71.64	22.07	89.19	19.40	50.41	87.75
Average Polisemy	1.71	1.56	1.78	1.66	1.80	1.74	1.76

Table 1: Statistics for each corpus in each language.

Corpus	Sentences	Annotations	Unique Words
SemCor	37,176	226,036	22,436
SemCor+OMSTI	850,974	1,137,170	22,437
Train-o-Matic	12,722,530	12,722,530	51,395

Table 2: Statistics of SemCor, OMSTI and Train-o-Matic about the number of sentences, annotations and unique words.

100, the first sense of the target word will receive K examples, the second one $\frac{K}{2^z}$ and so on; z is another parameter of the system.

3. Statistics

In this section we report some features of the corpora produced by Train-o-Matic, in order to give a complete overview of the data.

In Table 1 we show the number of annotations for each language as well as the number of distinct words and senses that have at least one example in our corpora and the number of sentences for each sense on average.

Train-o-Matic was able to generate around $18M$ annotated sentences for roughly $146K$ distinct lemmas and $63K$ distinct senses across languages. These corpora proved also to be of high quality, taking supervised system on par with or beyond state of the art results (Section 4.1.). The number of annotations is bigger for English and comparable across other languages: this is both because, for English, we set the value of the parameter K (see Section 2.1.) to 500 instead of 100, and because BabelNet, on average, contains more English senses compared to other languages.

As can be seen, each language has an average of 75 different sentences for each sense in the corpus, with English having the highest number of sentences per sense. Note that the total number of distinct senses covered is not equal to the sum of distinct senses for each sense due to the fact that we use a language-independent sense inventory (i.e. BabelNet) similarly to Otegi et al. (2016) and Delli Bovi et al. (2017). Thus many senses are shared across languages. The average confidence score measures how confident the system was on average when annotating the given language, meaning that the resulting data is most likely better: this score depends on both the average ambiguity of each lemma and on the quality of the relations in BabelNet. As expected, the system confidence score is highest in languages that have the lowest polisemy, i.e. English and German, which have the lowest average number of senses for nouns. As regards the average number of sentences for each sense, it directly depends on the parameter K and z that we set experimentally (see Section 2.1.). All corpora but English proved to lead supervised system to better performance when K was set to 100 and z between 2.0 and 3.0, thus we preferred to keep a lower number of more accurate sentences (50 for each sense). The English corpus, instead, was generated with K equal to 500 and z equal to 2.0 and thus it has a higher average number of sentences for each sense.

Table 2, instead, shows the comparison, in terms of number of sentences, annotations and unique words covered, between our automatically generated English corpus and two other corpora:

- SemCor (Miller et al., 1993), a corpus containing about 226,000 tokens annotated manually with WordNet senses.
- One Million Sense-Tagged Instances (Taghipour and Ng, 2015)[SemCor+OMSTI], a sense-annotated dataset obtained via a semi-automatic approach based on the disambiguation of a parallel corpus, i.e., the United Nations Parallel Corpus, performed by exploiting manually translated word senses. It also contains SemCor.

In terms of number of annotated sentences and number of annotations, our corpus is significantly bigger than SemCor and SemCor+OMSTI (by a factor of 200 and 10 respectively). More importantly, however, it covers double the amount of nouns that are covered by these two corpora, allowing supervised systems to have higher recall and to rely less on the Most Frequent Sense heuristic.

4. Experimental Setup

In order to evaluate the quality of the corpora we tested the performance of IMS, a state-of-the-art WSD system, when trained on our datasets.

English setup: For English, we compare the performance of IMS when trained on Train-o-Matic to that obtained against training with:

Test Set	Language	Train-o-Matic			Best System
		Precision	Recall	F1	F1
SemEval 2013	German	0.66	0.61	**0.63**	0.62
	French	0.61	0.60	**0.61**	**0.61**
	Spanish	0.68	0.66	0.67	**0.71**
	Italian	0.71	0.66	**0.68**	0.66
SemEval 2015	Spanish	61.3	54.8	**57.9**	56.3
	Italian	65.1	55.6	**59.9**	56.6

Table 3: Precision, Recall and F1 of IMS trained on Train-o-Matic, against the best performing system on SemEval-13 and SemEval-15.

Dataset	Train-o-Matic	OMSTI	SemCor	MFS
Senseval-2	70.5	74.1	**76.8**	72.1
Senseval-3	67.4	67.2	**73.8**	72.0
SemEval-07	59.8	62.3	**67.3**	65.4
SemEval-13	**65.5**	62.8	**65.5**	63.0
SemEval-15	**68.6**	63.1	66.1	66.3
ALL	67.3	66.4	**70.4**	67.6

Table 4: F1 of IMS trained on Train-o-Matic, OMSTI and SemCor, and MFS for the Senseval-2, Senseval-3, SemEval-07, SemEval-13 and SemEval-15 datasets.

The evaluation has been performed using the unified evaluation framework for Word Sense Disambiguation made available by Raganato et al. (2017), thus considering the following WSD shared tasks: Senseval-2 (Edmonds and Cotton, 2001), Senseval-3 (Snyder and Palmer, 2004), SemEval-2007 (Navigli et al., 2007), SemEval-2013 (Navigli et al., 2013) and SemEval-2015 (Moro and Navigli, 2015). We set the two Train-o-Matic parameters K to 500 and z to 2.0 experimentally, testing the models learned by IMS on a small in-house development set[2] and choosing the one with the highest performance.

Multilingual setup: For the other languages we tuned the two paramenter K and z in the same way we did for English. The corpora proved to be more effective with K set to 100, for all the languages, and z ranging in $[2.0, 3.0]$. To prove that the generated data in the other languages are also high quality we also report the performance of IMS when trained on Train-o-Matic corpora for Italian and Spanish on the Multilingual WSD task of SemEval-2015 (Moro and Navigli, 2015), and for German, French, Spanish and Italian on the Multilingual WSD task of SemEval-2013 (Navigli et al., 2013) which focuses on nouns only. Given that no supervised system have been submitted to this task[3] we compare against the best performing knowledge-based systems of the two SemEvals.

4.1. Results

English results: As can be seen in Table 4 IMS trained on our corpus is always comparable, if not better (from 2 to 3 points), than OMSTI[4]. SemCor, instead, provides better training data for 3 out of 5 datasets, while the performance of IMS is comparable on the SemEval-2013 and SemEval-2015. This shows that our automatically generated data can lead to better performance than semi-automatic datasets and, in some situations, even surpass that of manually annotated ones. More interestingly, the ability to automatically generate high-quality sense-annotated data enables the creation of domain-specific datasets that could be used to train WSD systems on particular domains of interest. Given that such a system would most likely outperform a system trained on non-specialized data (e.g. because the latter may have learned a Most Frequent Sense bias that is not accurate for the domain at hand), this is often a need for companies which need to specialize their software on a specific use case (see (Pasini and Navigli, 2017) for experiments on domain specific tasks).

Multilingual results: Looking now at results in Table 3 it is clear that the best improvement in performance, compared to the current state of the art, is obtained on low-resourced languages, which was our main objective. We note that IMS, when trained on Train-o-Matic corpora, is able to score from 1 to 3 points more than the best system of each language and each of the two SemEval (i.e. SemEval 2013 and SemEval 2015) but Spanish in SemEval 2013.
This comes as expected as supervised systems perform better than knowledge-based ones (Raganato et al., 2017) when enough training data is available. Still, it is not the purpose of this paper to show that these datasets provide the best possible training sets in all scenarios, but rather that they can be very valuable in low-resourced languages, for which training supervised systems would be otherwise impossible.

5. Conclusion

We release to the community 6 sense-annotated corpora for the 5 major European languages (English, French, Spanish, German and Italian) plus Chinese, each containing on average more than 1 million sentences from Wikipedia articles and automatically annotated using Train-o-Matic.
Our experiments proved that these corpora provide effective training ground for supervised WSD system, especially in a multilingual setting where sense annotated data is scarce, if at all available. As a matter of fact, the performance of supervised systems trained on this data is better or comparable to those trained on semi-automatically and, in some cases, manually-curated data. Given the lack of

[2]The development set contains roughly 50 items per language.
[3]Note that no supervised system have ever been submitted for a multilingual WSD task.
[4]We recall that OMSTI has been built using a semi-automatic approach and contains SemCor

such data for languages other than English, most WSD systems that target these languages usually adopt a knowledge-based approach, thus neglecting syntactic and contextual information that may be essential in some scenarios. This point is confirmed by the fact that we are able to outperform such systems by using these corpora as training set. All these points show that our corpora are able to address the need for sense-annotate data in low-resources languages. The data is available at `trainomatic.org`.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Consolidator Grant MOUSSE No. 726487.

6. Bibliographical References

Agirre, E. and Soroa, A. (2009). Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Athens, Greece, 30 March–3 April 2009*, pages 33–41.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117.

Camacho-Collados, J., Bovi, C. D., Raganato, A., and Navigli, R. (2016). A Large-Scale Multilingual Disambiguation of Glosses. In *Proceedings of LREC*, pages 1701–1708, Portoroz, Slovenia.

Delli Bovi, C., Camacho-Collados, J., Raganato, A., and Navigli, R. (2017). EuroSense: Automatic harvesting of multilingual sense annotations from parallel text. In *Proc.of ACL*, volume 2, pages 594–600.

Edmonds, P. and Cotton, S. (2001). Senseval-2: overview. In *Proc. of SensEval 2*, pages 1–5. ACL.

Christiane Fellbaum, editor. (1998). *WordNet: An Electronic Database*. MIT Press, Cambridge, MA.

Moro, A. and Navigli, R. (2015). Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proc. of SemEval-2015*.

Moro, A., Raganato, A., and Navigli, R. (2014). Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transaction of ACL (TACL)*, 2:231–244.

Navigli, R. and Ponzetto, S. P. (2010). BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 11–16 July 2010*, pages 216–225.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). Semeval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic, 23–24 June 2007*, pages 30–35.

Navigli, R., Jurgens, D., and Vannella, D. (2013). Semeval-2013 task 12: Multilingual word sense disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (* SEM '13)*, volume 2, pages 222–231.

Navigli, R. (2009). Word Sense Disambiguation: A survey. *ACM Computing Surveys*, 41(2):1–69.

Otegi, A., Aranberri, N., Branco, A., Hajic, J., Neale, S., Osenova, P., Pereira, R., Popel, M., Silva, J., Simov, K., et al. (2016). Qtleap wsd/ned corpora: Semantic annotation of parallel corpora in six languages. In *Proceedings of the 10th Language Resources and Evaluation Conference, LREC*, pages 3023–3030.

Pasini, T. and Navigli, R. (2017). Train-o-matic: Large-scale supervised word sense disambiguation in multiple languages without manual training data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 78–88. Association for Computational Linguistics.

Pasini, T. and Navigli, R. (2018). Two knowledge-based methods for high-performance sense distribution learning. In *AAAI-2018*.

Raganato, A., Delli Bovi, C., and Navigli, R. (2016). Automatic Construction and Evaluation of a Large Semantically Enriched Wikipedia. In *Proceedings of IJCAI*, pages 2894–2900, New York City, NY, USA, July.

Raganato, A., Camacho-Collados, J., and Navigli, R. (2017). Word Sense Disambiguation: A Unified Evaluation Framework and Empirical Comparison. In *Proc. of EACL*, pages 99–110, Valencia, Spain.

Schubert, L. K. (2006). Turing's dream and the knowledge challenge. In *Proc. of AAAI-06*, pages 1534–1538.

Snyder, B. and Palmer, M. (2004). The english all-words task. In *Proc. of Senseval-3*, pages 41–43, Barcelona, Spain.

Taghipour, Kaveh and Ng, Hwee Tou. (2015). *One Million Sense-Tagged Instances for Word Sense Disambiguation and Induction*. Association for Computational Linguistics.

7. Language Resource References

Miller, G. A., Leacock, C., Tengi, R., and Bunker, R. (1993). A semantic concordance. In *Proceedings of the 3rd DARPA Workshop on Human Language Technology*, pages 303–308, Plainsboro, N.J.

SentEval: An Evaluation Toolkit for Universal Sentence Representations

Alexis Conneau* **and Douwe Kiela**
Facebook Artificial Intelligence Research
{aconneau,dkiela}@fb.com

Abstract

We introduce SentEval, a toolkit for evaluating the quality of universal sentence representations. SentEval encompasses a variety of tasks, including binary and multi-class classification, natural language inference and sentence similarity. The set of tasks was selected based on what appears to be the community consensus regarding the appropriate evaluations for universal sentence representations. The toolkit comes with scripts to download and preprocess datasets, and an easy interface to evaluate sentence encoders. The aim is to provide a fairer, less cumbersome and more centralized way for evaluating sentence representations.

Keywords: representation learning, evaluation

1. Introduction

Following the recent word embedding upheaval, one of NLP's next challenges has become the hunt for universal general-purpose sentence representations. What distinguishes these representations, or embeddings, is that they are not necessarily trained to perform well on one specific task. Rather, their value lies in their transferability, i.e., their ability to capture information that can be of use in any kind of system or pipeline, on a variety of tasks.

Word embeddings are particularly useful in cases where there is limited training data, leading to sparsity and poor vocabulary coverage, which in turn lead to poor generalization capabilities. Similarly, sentence embeddings (which are often built on top of word embeddings) can be used to further increase generalization capabilities, composing unseen combinations of words and encoding grammatical constructions that are not present in the task-specific training data. Hence, high-quality universal sentence representations are highly desirable for a variety of downstream NLP tasks.

The evaluation of general-purpose word and sentence embeddings has been problematic (Chiu et al., 2016; Faruqui et al., 2016), leading to much discussion about the best way to go about it[1]. On the one hand, people have measured performance on intrinsic evaluations, e.g. of human judgments of word or sentence similarity ratings (Agirre et al., 2012; Hill et al., 2016b) or of word associations (Vulić et al., 2017). On the other hand, it has been argued that the focus should be on downstream tasks where these representations would actually be applied (Ettinger et al., 2016; Nayak et al., 2016). In the case of sentence representations, there is a wide variety of evaluations available, many from before the "embedding era", that can be used to assess representational quality on that particular task. Over the years, something of a consensus has been established, mostly based on the evaluations in seminal papers such as SkipThought (Kiros et al., 2015), concerning what evaluations to use. Recent works in which various alternative sentence encoders are compared use a similar set of tasks (Hill et al., 2016a; Conneau et al., 2017).

Implementing pipelines for this large set of evaluations, each with its own peculiarities, is cumbersome and induces unnecessary wheel reinventions. Another well-known problem with the current status quo, where everyone uses their own evaluation pipeline, is that different preprocessing schemes, evaluation architectures and hyperparameters are used. The datasets are often small, meaning that minor differences in evaluation setup may lead to very different outcomes, which implies that results reported in papers are not always fully comparable.

In order to overcome these issues, we introduce SentEval[2]: a toolkit that makes it easy to evaluate universal sentence representation encoders on a large set of evaluation tasks that has been established by community consensus.

2. Aims

The aim of SentEval is to make research on universal sentence representations fairer, less cumbersome and more centralized. To achieve this goal, SentEval encompasses the following:

- one central set of evaluations, based on what appears to be community consensus;
- one common evaluation pipeline with fixed standard hyperparameters, apart from those tuned on validation sets, in order to avoid discrepancies in reported results; and
- easy access for anyone, meaning: a straightforward interface in Python, and scripts necessary to download and preprocess the relevant datasets.

In addition, we provide examples of models, such as a simple bag-of-words model. These could potentially also be used to extrinsically evaluate the quality of word embeddings in NLP tasks.

3. Evaluations

Our aim is to obtain general-purpose sentence embeddings that capture generic information, which should be useful

*LIUM, Université Le Mans

[1] See also recent workshops on evaluating representations for NLP, e.g. RepEval: https://repeval2017.github.io/

[2] https://github.com/facebookresearch/SentEval

name	N	task	C	examples	label(s)
MR	11k	sentiment (movies)	2	"Too slow for a younger crowd , too shallow for an older one."	neg
CR	4k	product reviews	2	"We tried it out christmas night and it worked great ."	pos
SUBJ	10k	subjectivity/objectivity	2	"A movie that doesn't aim too high , but doesn't need to."	subj
MPQA	11k	opinion polarity	2	"don't want"; "would like to tell";	neg, pos
TREC	6k	question-type	6	"What are the twin cities ?"	LOC:city
SST-2	70k	sentiment (movies)	2	"Audrey Tautou has a knack for picking roles that magnify her [..]"	pos
SST-5	12k	sentiment (movies)	5	"nothing about this movie works."	0

Table 1: **Classification tasks**. C is the number of classes and N is the number of samples.

name	N	task	output	premise	hypothesis	label
SNLI	560k	NLI	3	"A small girl wearing a pink jacket is riding on a carousel."	"The carousel is moving."	entailment
SICK-E	10k	NLI	3	"A man is sitting on a chair and rubbing his eyes"	"There is no man sitting on a chair and rubbing his eyes"	contradiction
SICK-R	10k	STS	$[0, 5]$	"A man is singing a song and playing the guitar"	"A man is opening a package that contains headphones"	1.6
STS14	4.5k	STS	$[0, 5]$	"Liquid ammonia leak kills 15 in Shanghai"	"Liquid ammonia leak kills at least 15 in Shanghai"	4.6
MRPC	5.7k	PD	2	"The procedure is generally performed in the second or third trimester."	"The technique is used during the second and, occasionally, third trimester of pregnancy."	paraphrase
COCO	565k	ICR	sim	 [3]	"A group of people on some horses riding through the beach."	rank

Table 2: **Natural Language Inference and Semantic Similarity tasks**. NLI labels are contradiction, neutral and entailment. STS labels are scores between 0 and 5. PD=paraphrase detection, ICR=image-caption retrieval.

for a broad set of tasks. To evaluate the quality of these representations, we use them as features in various transfer tasks.

Binary and multi-class classification We use a set of binary classification tasks (see Table 1) that covers various types of sentence classification, including sentiment analysis (MR and both binary and fine-grained SST) (Pang and Lee, 2005; Socher et al., 2013), question-type (TREC) (Voorhees and Tice, 2000), product reviews (CR) (Hu and Liu, 2004), subjectivity/objectivity (SUBJ) (Pang and Lee, 2004) and opinion polarity (MPQA) (Wiebe et al., 2005). We generate sentence vectors and classifier on top, either in the form of a Logistic Regression or an MLP. For MR, CR, SUBJ and MPQA, we use nested 10-fold cross-validation, for TREC cross-validation and for SST standard validation.

Entailment and semantic relatedness We also include the SICK dataset (Marelli et al., 2014) for entailment (SICK-E), and semantic relatedness datasets including SICK-R and the STS Benchmark dataset (Cer et al., 2017). For semantic relatedness, which consists of predicting a semantic score between 0 and 5 from two input sentences, we follow the approach of Tai et al. (2015a) and learn to predict the probability distribution of relatedness scores. SentEval reports Pearson and Spearman correlation. In addition, we include the SNLI dataset (Bowman et al., 2015), a collection of 570k human-written English supporting the task of natural language inference (NLI), also known as recognizing textual entailment (RTE) which consists of predicting whether two input sentences are entailed, neutral or contradictory. SNLI was specifically designed to serve as a benchmark for evaluating text representation learning methods.

Semantic Textual Similarity While semantic relatedness requires training a model on top of the sentence embeddings, we also evaluate embeddings on the unsupervised SemEval tasks. These datasets include pairs of sentences taken from news articles, forum discussions, news conversations, headlines, image and video descriptions labeled with a similarity score between 0 and 5. The goal is to evaluate how the cosine distance between two sentences correlate with a human-labeled similarity score through Pearson and Spearman correlations. We include STS tasks from 2012 (Agirre et al., 2012), 2013[4] (Agirre et al., 2013), 2014 (Agirre et al., 2014), 2015 (Agirre et al., 2015) and 2016 (Agirre et al., 2016). Each of these tasks includes several subtasks. SentEval reports both the average and the weighted average (by number of samples in each subtask) of the Pearson and Spearman correlations.

Paraphrase detection The Microsoft Research Paraphrase Corpus (MRPC) (Dolan et al., 2004) is composed of pairs of sentences which have been extracted from news sources on the Web. Sentence pairs have been human-annotated according to whether they capture a paraphrase/semantic equivalence relationship. We use the same

[3] Antonio Rivera - CC BY 2.0 - flickr

[4] Due to License issues, we do not include the SMT subtask.

approach as with SICK-E, except that our classifier has only 2 classes, i.e., the aim is to predict whether the sentences are paraphrases or not.

Caption-Image retrieval The caption-image retrieval task evaluates joint image and language feature models (Lin et al., 2014). The goal is either to rank a large collection of images by their relevance with respect to a given query caption (Image Retrieval), or ranking captions by their relevance for a given query image (Caption Retrieval). The COCO dataset provides a training set of 113k images with 5 captions each. The objective consists of learning a caption-image compatibility score $\mathcal{L}_{\text{cir}}(x, y)$ from a set of aligned image-caption pairs as training data. We use a pairwise ranking-loss $\mathcal{L}_{\text{cir}}(x, y)$:

$$\sum_y \sum_k \max(0, \alpha - s(Vy, Ux) + s(Vy, Ux_k)) +$$
$$\sum_x \sum_{k'} \max(0, \alpha - s(Ux, Vy) + s(Ux, Vy_{k'})),$$

where (x, y) consists of an image y with one of its associated captions x, $(y_k)_k$ and $(y_{k'})_{k'}$ are negative examples of the ranking loss, α is the margin and s corresponds to the cosine similarity. U and V are learned linear transformations that project the caption x and the image y to the same embedding space. We measure Recall@K, with K $\in \{1, 5, 10\}$, i.e., the percentage of images/captions for which the corresponding caption/image is one of the first K retrieved; and median rank. We use the same splits as Karpathy and Fei-Fei (2015), i.e., we use 113k images (each containing 5 captions) for training, 5k images for validation and 5k images for test. For evaluation, we split the 5k images in 5 random sets of 1k images on which we compute the mean $R@1$, $R@5$, $R@10$ and median (Med r) over the 5 splits. We include 2048-dimensional pretrained ResNet-101 (He et al., 2016) features for all images.

4. Usage and Requirements

Our evaluations comprise two different types: ones where we need to learn on top of the provided sentence representations (e.g. classification/regression) and ones where we simply take the cosine similarity between the two representations, as in the STS tasks. In the binary and multi-class classification tasks, we fit either a Logistic Regression classifier or an MLP with one hidden layer on top of the sentence representations. For the natural language inference tasks, where we are given two sentences u and v, we provide the classifier with the input $\langle u, v, |u - v|, u * v\rangle$. To fit the Pytorch models, we use Adam (Kingma and Ba, 2014), with a batch size 64. We tune the L2 penalty of the classifier with grid-search on the validation set. When using SentEval, two functions should be implemented by the user:

- `prepare(params, dataset)`: sees the whole dataset and applies any necessary preprocessing, such as constructing a lookup table of word embeddings (this function is optional); and
- `batcher(params, batch)`: given a batch of input sentences, returns an array of the sentence embeddings for the respective inputs.

The main `batcher` function allows the user to encode text sentences using any Python framework. For example, the batcher function might be a wrapper around a model written in Pytorch, TensorFlow, Theano, DyNet, or any other framework[5]. To illustrate the use, here is an example of what an evaluation script looks like, having defined the prepare and batcher functions:

```
import senteval
se = senteval.engine.SE(
        params, batcher, prepare)
transfer_tasks = ['MR', 'CR']
results = se.eval(transfer_tasks)
```

Parameters Both functions make use of a `params` object, which contains the settings of the network and the evaluation. SentEval has several parameters that influence the evaluation procedure. These include the following:

- `task_path` (str, required): path to the data.
- `seed` (int): random seed for reproducibility.
- `batch_size` (int): size of minibatch of text sentences provided to batcher (sentences are sorted by length).
- `kfold` (int): k in the kfold-validation (default: 10).

The default config is:

```
params = {'task_path': PATH_TO_DATA,
          'usepytorch': True,
          'kfold': 10}
```

We also give the user the ability to customize the classifier used for the classification tasks.

Classifier To be comparable to the results published in the literature, users should use the following parameters for Logistic Regression:

```
params['classifier'] =
    {'nhid': 0, 'optim': 'adam',
     'batch_size': 64, 'tenacity': 5,
     'epoch_size': 4}
```

The parameters of the classifier include:

- `nhid` (int): number of hidden units of the MLP; if nhid>0, a Multi-Layer Perceptron with one hidden layer and a Sigmoid nonlinearity is used.
- `optim` (str): classifier optimizer (default: adam).
- `batch_size` (int): batch size for training the classifier (default: 64).
- `tenacity` (int): stopping criterion; maximum number of times the validation error does not decrease.
- `epoch_size` (int): number of passes through the training set for one epoch.

[5]Or any other programming language, as long as the vectors can be passed to, or loaded from, code written in Python.

Model	MR	CR	SUBJ	MPQA	SST-2	SST-5	TREC	MRPC	SICK-E
Representation learning (transfer)									
GloVe LogReg	77.4	78.7	91.2	87.7	80.3	44.7	83.0	72.7/81.0	78.5
GloVe MLP	77.7	79.9	92.2	88.7	82.3	45.4	85.2	73.0/80.9	79.0
fastText LogReg	78.2	80.2	91.8	88.0	82.3	45.1	83.4	74.4/82.4	78.9
fastText MLP	78.0	81.4	92.9	88.5	84.0	45.1	85.6	74.4/82.3	80.2
SkipThought	79.4	83.1	93.7	89.3	82.9	-	88.4	72.4/81.6	79.5
InferSent	81.1	86.3	92.4	90.2	84.6	46.3	88.2	76.2/83.1	86.3
Supervised methods directly trained for each task (no transfer)									
SOTA	83.1[1]	86.3[1]	95.5[1]	93.3[1]	89.5[2]	52.4[2]	96.1[2]	80.4/85.9[3]	84.5[4]

Table 3: Transfer test results for various baseline methods. We include supervised results trained directly on each task (no transfer). Results [1] correspond to AdaSent (Zhao et al., 2015), [2] to BLSTM-2DCNN (Zhou et al., 2016), [3] to TF-KLD (Ji and Eisenstein, 2013) and [4] to Illinois-LH system (Lai and Hockenmaier, 2014).

- `dropout` (float): dropout rate in the case of MLP.

For use cases where there are multiple calls to SentEval, e.g when evaluating the sentence encoder at every epoch of training, we propose the following prototyping set of parameters, which will lead to slightly worse results but will make the evaluation significantly faster:

```
params['classifier'] =
    {'nhid': 0, 'optim': 'rmsprop',
     'batch_size': 128, 'tenacity': 3,
     'epoch_size': 2}
```

You may also pass additional parameters to the `params` object in order which will further be accessible from the prepare and batcher functions (e.g a pretrained model).

Datasets In order to obtain the data and preprocess it so that it can be fed into SentEval, we provide the `get_transfer_data.bash` script in the data directory. The script fetches the different datasets from their known locations, unpacks them and preprocesses them. We tokenize each of the datasets with the MOSES tokenizer (Koehn et al., 2007) and convert all files to UTF-8 encoding. Once this script has been executed, the task_path parameter can be set to indicate the path of the data directory.

Requirements SentEval is written in Python. In order to run the evaluations, the user will need to install numpy, scipy and recent versions of pytorch and scikit-learn. In order to facilitate research where no GPUs are available, we offer for the evaluations to be run on CPU (using scikit-learn) where possible. For the bigger datasets, where more complicated models are often required, for instance STS Benchmark, SNLI, SICK-R and the image-caption retrieval tasks, we recommend pytorch models on a single GPU.

5. Baselines

Several baseline models are evaluated in Table 3:

- Continuous bag-of-words embeddings (average of word vectors). We consider the most commonly used pretrained word vectors available, namely the fastText (Mikolov et al., 2017) and the GloVe (Pennington et al., 2014) vectors trained on CommonCrawl.
- SkipThought vectors (Ba et al., 2016)
- InferSent vectors (Conneau et al., 2017)

In addition to these methods, we include the results of current state-of-the-art methods for which both the encoder and the classifier are trained on each task (no transfer). For GloVe and fastText bag-of-words representations, we report the results for Logistic Regression and Multi-Layer Perceptron (MLP). For the MLP classifier, we tune the dropout rate and the number of hidden units in addition to the L2 regularization. We do not observe any improvement over Logistic Regression for methods that already have a large embedding size (4096 for Infersent and 4800 for SkipThought). On most transfer tasks, supervised methods that are trained directly on each task still outperform transfer methods. Our hope is that SentEval will help the community build sentence representations with better generalization power that can outperform both the transfer and the supervised methods.

6. Conclusion

Universal sentence representations are a hot topic in NLP research. Making use of a generic sentence encoder allows models to generalize and transfer better, even when trained on relatively small datasets, which makes them highly desirable for downstream NLP tasks.

We introduced SentEval as a fair, straightforward and centralized toolkit for evaluating sentence representations. We have aimed to make evaluation as easy as possible: sentence encoders can be evaluated by implementing a simple Python interface, and we provide a script to download the necessary evaluation datasets. In future work, we plan to enrich SentEval with additional tasks as the consensus on the best evaluation for sentence embeddings evolves. In particular, tasks that probe for specific linguistic properties of the sentence embeddings (Shi et al., 2016; Adi et al., 2017) are interesting directions towards understanding how the encoder understands language. We hope that our toolkit will be used by the community in order to ensure that fully comparable results are published in research papers.

Model	SST'12	SST'13	SST'14	SST'15	SST'16	SICK-R	SST-B
Representation learning (transfer)							
GloVe BoW	52.1	49.6	54.6	56.1	51.4	79.9	64.7
fastText BoW	58.3	57.9	64.9	67.6	64.3	82.0	70.2
SkipThought-LN	30.8	24.8	31.4	31.0	-	85.8	72.1
InferSent	59.2	58.9	69.6	71.3	71.5	88.3	75.6
Char-phrase	66.1	57.2	74.7	76.1	-	-	-
Supervised methods directly trained for each task (no transfer)							
PP-Proj	60.0[1]	56.8[1]	71.3[1]	74.8[1]	-	86.8[2]	-

Table 4: Evaluation of sentence representations on the semantic textual similarity benchmarks. Numbers reported are Pearson correlations x100. We use the average of Pearson correlations for STS'12 to STS'16 which are composed of several subtasks. Charagram-phrase numbers were taken from (Wieting et al., 2016). Results [1] correspond to PP-Proj (Wieting et al., 2015) and [2] from Tree-LSTM (Tai et al., 2015b).

7. Bibliographical References

Adi, Y., Kermany, E., Belinkov, Y., Lavi, O., and Goldberg, Y. (2017). Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *Proceedings of ICLR Conference Track*, Toulon, France. Published online: `https://openreview.net/group?id=ICLR.cc/2017/conference`.

Agirre, E., Diab, M., Cer, D., and Gonzalez-Agirre, A. (2012). Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of Semeval-2012*, pages 385–393.

Agirre, E., Cer, D., Diab, M., Gonzalez-agirre, A., and Guo, W. (2013). sem 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In *In *SEM 2013: The Second Joint Conference on Lexical and Computational Semantics. Association for Computational Linguistics*.

Agirre, E., Baneab, C., Cardiec, C., Cerd, D., Diabe, M., Gonzalez-Agirre, A., Guof, W., Mihalceab, R., Rigaua, G., and Wiebeg, J. (2014). Semeval-2014 task 10: Multilingual semantic textual similarity. *SemEval 2014*, page 81.

Agirre, E., Banea, C., Cardie, C., Cer, D. M., Diab, M. T., Gonzalez-Agirre, A., Guo, W., Lopez-Gazpio, I., Maritxalar, M., Mihalcea, R., et al. (2015). Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In *SemEval@ NAACL-HLT*, pages 252–263.

Agirre, E., Baneab, C., Cerd, D., Diabe, M., Gonzalez-Agirre, A., Mihalceab, R., Rigaua, G., Wiebef, J., and Donostia, B. C. (2016). Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. *Proceedings of SemEval*, pages 497–511.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *Advances in neural information processing systems (NIPS)*.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of EMNLP*.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Chiu, B., Korhonen, A., and Pyysalo, S. (2016). Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *First Workshop on Evaluating Vector Space Representations for NLP (RepEval)*.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of EMNLP*, Copenhagen, Denmark.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of ACL*, page 350.

Ettinger, A., Elgohary, A., and Resnik, P. (2016). Probing for semantic evidence of composition by means of simple classification tasks. In *First Workshop on Evaluating Vector Space Representations for NLP (RepEval)*, page 134.

Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. *arXiv preprint arXiv:1605.02276*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of CVPR*.

Hill, F., Cho, K., and Korhonen, A. (2016a). Learning distributed representations of sentences from unlabelled data. In *Proceedings of NAACL*.

Hill, F., Reichart, R., and Korhonen, A. (2016b). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of SIGKDD*, pages 168–177.

Ji, Y. and Eisenstein, J. (2013). Discriminative improvements to distributional sentence similarity. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of CVPR*, pages 3128–3137.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. In *Proceedings of the 3rd*

International Conference on Learning Representations (ICLR).

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lai, A. and Hockenmaier, J. (2014). Illinois-lh: A denotational and distributional approach to semantics. *Proc. SemEval*, 2:5.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer International Publishing.

Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., and Zamparelli, R. (2014). A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of LREC*.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2017). Advances in pre-training distributed word representations.

Nayak, N., Angeli, G., and Manning, C. D. (2016). Evaluating word embeddings using a representative suite of practical tasks. In *First Workshop on Evaluating Vector Space Representations for NLP (RepEval)*, page 19.

Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of ACL*, page 271.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of ACL*, pages 115–124.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, volume 14, pages 1532–1543.

Shi, X., Padhi, I., and Knight, K. (2016). Does string-based neural MT learn source syntax? In *Proceedings of EMNLP*, pages 1526–1534, Austin, Texas.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., Potts, C., et al. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, pages 1631—1642.

Tai, K. S., Socher, R., and Manning, C. D. (2015a). Improved semantic representations from tree-structured long short-term memory networks. *Proceedings of ACL*.

Tai, K. S., Socher, R., and Manning, C. D. (2015b). Improved semantic representations from tree-structured long short-term memory networks. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Voorhees, E. M. and Tice, D. M. (2000). Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207. ACM.

Vulić, I., Kiela, D., and Korhonen, A. (2017). Evaluation by association: A systematic study of quantitative word association evaluation. In *Proceedings of EACL*, volume 1, pages 163–175.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2015). Towards universal paraphrastic sentence embeddings. *Proceedings of the 4th International Conference on Learning Representations (ICLR)*.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2016). Charagram: Embedding words and sentences via character n-grams. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Zhao, H., Lu, Z., and Poupart, P. (2015). Self-adaptive hierarchical sentence model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 4069–4076. AAAI Press.

Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., and Xu, B. (2016). Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*.

A Survey on Automatically-Constructed WordNets and their Evaluation: Lexical and Word Embedding-based Approaches

Steven Neale
School of English, Communication and Philosophy
Cardiff University, Wales, United Kingdom
NealeS2@cardiff.ac.uk

Abstract

WordNets – lexical databases in which groups of synonyms are arranged according to the semantic relationships between them – are crucial resources in semantically-focused natural language processing tasks, but are extremely costly and labour intensive to produce. In languages besides English, this has led to growing interest in constructing and extending WordNets automatically, as an alternative to producing them from scratch. This paper describes various approaches to constructing WordNets automatically – by leveraging traditional lexical resources and newer trends such as word embeddings – and also offers a discussion of the issues affecting the evaluation of automatically constructed WordNets.

Keywords: WordNet, evaluation, automatic synset extraction, bilingual dictionaries, word embeddings

1. Introduction

WordNets (Fellbaum, 1998) are lexical databases in which open-class words – nouns, verbs, adjectives and adverbs – are stored as sets of synonyms or 'synsets' and linked to each other by various semantic relationships. These relationships include antonymy (opposites, i.e. 'wet' and 'dry'), hypernymy and hyponymy (types and types-of, i.e. 'animal' and 'cat'), and meronymy (parts-of, i.e. 'toe' as a meronym of 'foot'), among others. The *Princeton WordNet (PWN)*[1] is the original and pioneering English language WordNet – now totalling over 117,000 synsets in version 3.0 – and its format has become the gold standard for lexical databases representing meanings and concepts.

Besides its adoption as a standard for lexical databases, the PWN has become a vital resource across a range of semantic processing tasks – it has been used to produce gold-standard semantically-annotated corpora such as *SemCor*[2] and to determine the correct meanings of words in natural language processing tasks such as word-sense disambiguation, text summarization, and semantic textual similarity. Naturally, its extensive usage in English has inspired the construction of new WordNets in many other languages. The *Global WordNet Association (GWA)*[3], for example, was set up to provide a platform for discussing, sharing and connecting WordNets in any language, while large-scale research projects such as *EuroWordNet* (Vossen, 2004) and *MultiWordNet* (Pianta et al., 2002) have focused on aligning synsets from multiple WordNets in different languages.

However, while the PWN has been constructed, extended and refined over decades, building new WordNets from scratch is an enormous undertaking. Manual construction of course ensures accurate synsets covering as many concepts as possible, but it requires lexicographers to spend hours crafting synsets and is thus far too labour-intensive to be feasible in most cases. Inevitably, this has led to a growing and improving body of research into techniques and approaches for constructing and extending WordNets automatically.

In this paper, a number of approaches to automatically construct and extend WordNets are presented, taking into account both lexical and word embedding-based approaches. In addition, a discussion of the evaluation of these resources – and particular bottlenecks that a lack of evaluation principles and guidelines cause – is offered. The contribution of the paper is twofold: a summary of current techniques and approaches for automatic WordNet construction and extension, and a first step in encouraging the further discussion and development of common guidelines for improving the area – particularly in terms of evaluation – moving forward.

2. Background - WordNet Construction

Generally speaking, WordNets are constructed using one of two approaches (Vossen, 1998):

- The *merge* approach – whereby an exhaustive repository of senses (meanings) of each word is compiled, with synsets then created that contain all of the applicable words for a given sense.
- The *expansion* approach – whereby existing synsets from a reference WordNet are used as a guide to create corresponding synsets in a new WordNet, by gathering applicable words that represent the meaning of the synset and ordering them by frequency.

Since the introduction of the PWN and the success of early projects such as *EuroWordNet* that were built around its principles, many projects have focused on building new WordNets in diverse languages using these methods. These endeavours have highlighted various advantages and disadvantages of both the merge and the expansion approaches. Bhattacharyya (2010) describes how the merge approach results in WordNets of high quality, on account of expert lexicographers working in detail on only one language; however, the process is typically very slow. Conversely, the expansion approach can allow the construction of the WordNet to take place much more quickly, with construction

[1] http://wordnet.princeton.edu/wordnet/

[2] http://web.eecs.umich.edu/~mihalcea/downloads.html#semcor

[3] http://globalwordnet.org/

guided by synsets and semantic relationships in the source (or reference) WordNet; however, lexicographers still need to dedicate time to constructing language-specific synsets (meanings or concepts which may not be represented or have a place in the source WordNet), and there is a danger of specific concepts only applicable to the target language being overlooked altogether (Bhattacharyya, 2010).

Years of work on constructing new WordNets have also contributed to the development of guidelines and principles for creating them, largely based on leveraging existing knowledge using the expansion approach. The GWA outlines the importance of 'base concepts'[4] – those concepts that occupy a high position in a semantic hierarchy and have many relations to other concepts – as playing a vital role in constructing WordNets. Base concepts are defined by their *universality* – common (to at least two languages), local (to only one language), or global (across all languages) – with an initial set of 1024 common base concepts being released as part of the *EuroWordNet* project[5]. As a starting point, the GWA proposes that WordNets be constructed in two steps:

- A *core* WordNet of between 5,000 and 10,000 synsets is constructed around the common base concepts,
- An *extended* WordNet is built (semi-automatically, given the semantic basis of the core WordNet) to increase the total number of synsets to 20,000 and beyond.

Given that for many languages these core synsets are readily-constructed, it makes sense to leverage them when constructing new WordNets, and to 'borrow' the semantic relationships that have already been created (Bhattacharyya, 2010). Given the amount of time that can be saved by re-using existing work, there is a tendency to see the expansion approach favoured over the merge approach – it also lends itself extremely well to the automatic construction of synsets, where input from lexicographers is minimal to zero. Thus, the research described in this paper (and particularly in section 3.) largely follows the expansion approach, with synsets being constructed by automatically extracting lexical data from a range of resources in order to build a skeleton framework of meanings based on a reference WordNet.

3. Automatic WordNet Construction from Lexical Resources

Automatic construction and extension of WordNets has traditionally relied on existing lexical resources. Most of the existing research on the subject describes that to create the 'target' WordNet, at least two resources are needed:

- A 'source' WordNet (usually the PWN),
- Lexical resources such as on-line encyclopedias, bilingual dictionaries, or parallel corpora – possibly leveraged in conjunction with additional processing techniques such as machine translation.

[4] http://globalwordnet.org/gwa-base-concepts/

[5] http://www.globalwordnet.org/gwa/ewn_to_bc/topont.htm

3.1. On-line Encyclopedias

An early approach to constructing WordNet synsets automatically explored the idea of extracting semantic relationships from an on-line encyclopedia – namely Wikipedia[6] – in order to extend the PWN (Ruiz-Casado et al., 2005). The approach identifies lexical patterns representing the semantic relationships between entities (links) in Wikipedia, and then uses these patterns to extend existing or create new WordNet synsets. This process takes place over four steps:

- 'Disambiguating' Wikipedia entities to associate them with their corresponding WordNet synsets,
- Extracting patterns of context between Wikipedia entities and other terms they are associated with via a hyperlink,
- Generalising the extracted patterns by comparing them with each other and finding matching ones,
- Applying the patterns to find new semantic relationships not already present in the WordNet.

A manual evaluation of 360 'disambiguated' Wikipedia entities showed that 92% were accurately associated with their corresponding synsets in PWN. As for the patterns extracted from them and used to extend PWN, a total of 1204 hyponomy relations, 418 holonymy relations, and 184 meronymy relations were added to the existing WordNet, with precisions of 0.69, 0.61, and 0.61 respectively. Only 4 hypernymy patterns were extracted by the process (all of which were already present in PWN), perhaps showing that most broader concepts (hypernyms being 'types') are a) likely to be already present in WordNets, and b) not likely to be found in Wikipedia definitions, which by their very nature tend to describe known concepts in more detail.

3.2. Bilingual Dictionaries

The most common technique for populating new WordNets automatically has been to leverage the information in bilingual dictionaries in the source and target languages. Use of bilingual dictionaries for this purpose goes back to very early work on building Catalan and Spanish WordNets as part of the *EuroWordNet* project. In this work, translations of English words in the source WordNet were found, and these translations classified by features such as polysemy (number of translations for each word), structure (the semantic relationships between translations in the source WordNet), and 'conceptual distance' (length of the path between two words in a graph-based representation of the source WordNet) to create a skeleton WordNet in the target language, which could be extended later using bilingual taxonomies (Farreres et al., 1998).

Since then, bilingual dictionaries have continued to be a popular resource for the automatic construction of WordNets. A Romanian WordNet was built by using a range of heuristics to:

- Analyse the relationships between synsets in the source (English) WordNet,

[6] https://en.wikipedia.org

- Identify semantic relationships in various target language resources,

- Map these relationships to each other in the target (Romanian) WordNet using a bilingual dictionary (Barbu and Barbu Mititelu, 2005).

The method was evaluated using 9716 synsets from a pre-existing Romanian WordNet that also had entries in PWN, from which these 9716 synsets were extracted and used as the source (English) WordNet – the synsets used were limited to hypernymy and meronymy relations, and all 19,624 literal words within the synsets had an entry in the bilingual dictionary. The resulting automatically-constructed Romanian WordNet contained 9610 synsets connected by approximately 11,969 semantic relationships, which were reported to be 91% accurate when compared to the 9716 synsets from the pre-existing Romanian WordNet (Barbu and Barbu Mititelu, 2005).

In more recent work on building a Persian WordNet, a bilingual dictionary was used to extract a group of 'candidate' synsets containing English translations of a given Persian word from a source WordNet (PWN). These candidate synsets were then ranked by calculating the Mutual Information of the given Persian word and its English translations in both source and target language corpora, and based on this ranking the most appropriate candidate synset to use for the target (Persian) WordNet was selected (Montazery and Faili, 2010). An extension of this work specifically aimed at lesser-resourced languages was also described, in which a Persian WordNet is constructed by finding the English translations of Persian words in small corpora using a bilingual dictionary. These translations are then used to perform word sense disambiguation (WSD) on a Persian sentence using a source (English) WordNet, and the English synsets returned by the WSD algorithm are mapped to the target (Persian) WordNet (Taghizadeh and Faili, 2016).

Again, these techniques have been shown to be able to automatically construct WordNets with a good degree of accuracy. Montazery and Faili (2010) report that a manual evaluation of 500 synsets from their automatically-constructed target WordNet (which in total covered 29,716 synsets from PWN) resulted in an accuracy of 82.6% (95.8% for synsets whose mapping from source to target WordNet was unambiguous and 76.4% for synsets whose correct mapping had to be decided by ranking multiple candidates). Taghizadeh and Faili (2016) manually evaluated 1,750 word/synset pairs from their target WordNet, and describe how a threshold value (between 0 and 1) used by their WSD algorithm to remove low-scoring candidate synsets had a significant impact on their results. Higher threshold values resulted in the WordNet being more precise (90% with a threshold value of 0.1) but with low recall (fewer synsets in the target WordNet), while lower threshold values resulted in a WordNet with higher recall (more synsets) but with low precision (74% with the threshold value set to 0).

3.3. Machine Translation

While bilingual dictionaries have been the most commonly-used resources in the automatic construction of WordNets, they have also been leveraged in conjunction with additional processing techniques. Recent work by Lam et al. (2014) focused on the construction of WordNets in a variety of languages – Arabic, Assamese, Dimasa, Karbi, and Vietnamese – using machine translation as well as (or in some cases instead of) a bilingual dictionary. For example, they describe:

- An *intermediate WordNets (IW)* approach whereby the same synset from WordNets in a number of different languages is translated into the target language using machine translation,

- An *intermediate WordNets and dictionary (IWND)* approach whereby the same synset from WordNets in a number of different languages is translated into English using a bilingual dictionary, and then from English to the target language using machine translation.

For both techniques, after a set of translated candidate synsets in the target language has been produced, the candidates are ranked based on various heuristics to decide on the most appropriate target language translation for the original synset in question.

Matching subsets of 500 automatically-constructed synsets in Arabic, Assamesse, and Vietnamese were evaluated using a 5-point Likert scale, with an average score of 3.82, 3.78, and 3.75 respectively (3-4, average to good on the Likert scale). It was also shown that the coverage (number of synsets in the automatically-constructed WordNet compared to PWN) of the bilingual dictionary-based IWND technique was generally higher (70,536 synsets in Arabic, for example) than with the IW technique (48,245 synsets with 2 intermediate WordNets, or 61,354 with 3) unless the IW technique was used with 4 intermediate WordNets (75,234 synsets) (Lam et al., 2014).

3.4. Parallel Corpora

A recently-explored alternative to leveraging bilingual dictionaries to automatically construct WordNets has been to exploit parallel corpora in order to map synsets between source and target languages. Oliver and Climent (2014) experimented with extracting synsets for target WordNets by aligning English terms tagged with PWN synset identifiers to their corresponding lemmas in parallel corpora in Spanish, French, German, Italian and Portuguese. This is achieved by assigning synset identifiers from PWN to the English side of the parallel corpora using the *Freeling*[7] and *UKB*[8] word sense disambiguation toolkits, tagging the target language sides with simple POS tags (nouns, verbs, adjectives and adverbs), and then using a simple word alignment algorithm based on most frequent translations to map the subsets of the English synsets to their corresponding source language words.

The resulting automatically-constructed WordNets were evaluated using reference WordNets for each language. For each source language synset for which target language variants were extracted using the method, those target language

[7] http://nlp.lsi.upc.edu/freeling/node/1

[8] http://ixa2.si.ehu.es/ukb/

variants were compared with the corresponding synset in the target language reference WordNet, with a mapping being considered correct if the target language variant proposed by the method was present in the reference WordNet. Experimenting with three different parallel corpora, the automatically-constructed WordNets were considered quite precise, with results ranging between 75.73% and 85.03% for Spanish, French, Italian and Portuguese (although the extracted German WordNet was noticeably less precise at 45.64-53.15% across the three parallel corpora). However, as with the bilingual dictionary-based approaches to automatic WordNet construction, the reported recall from the parallel corpora-based approach was low (ranging from 10.96 to as low as 2.93 for French synset variants extracted from one of the three corpora). The number of extracted variants in each target language was considered to be few, given the size and number of English PWN synsets present in the English sides of the corpora.

4. Word Embeddings and WordNet Synsets

Given the increasing popularity of word embeddings (vector space representations of word meanings based on their distribution within large datasets), it should come as no surprise that the links between embeddings and traditional, WordNet-style representations of word senses have recently been explored. Nayak (2015) demonstrated that word embeddings can be classified according to words and can also be used to predict hypernymy relations between them, while Rothe and Schütze (2015) report that sets of embeddings trained not just on words but also on synsets (groups of synonyms) and lexemes (word-synset pairs) achieve state-of-the-art performance on WSD and semantic similarity tasks. This kind of research shows the potential of word embeddings for capturing the kinds of relationships (and being useful in the types of tasks) commonly associated with WordNet-style word senses – potential which is further compounded by the reported high precision with which multi-sense word embeddings can be mapped to WordNet-style synset entries in Babelnet[9] (Panchenko, 2016).

4.1. Extending and Constructing Synsets from Word Embeddings

Naturally, recent research has explored leveraging the links between word embeddings and synsets in order to automatically construct new synsets from the embeddings themselves. Sand et al. (2017) describe using word embeddings to extend an existing Norwegian WordNet by finding candidate hypernyms for a given word based on its nearest neighbours in the WordNet, and then scoring these candidate hypernyms by distributional similarity (using the vector space of the embeddings) and distance in a graph-based representation of the WordNet. Based on an evaluation of 1388 target words occurring 5 times or more in the news corpus on which the embeddings were trained, an *accuracy* (percentage of newly-added target words correctly placed under the appropriate hypernym) of 55.80% and an *attachment* score (percentage of target words actually added to the Norwegian WordNet) of 96.33% were recorded. This accuracy is increased when only evaluating on target words that occurred more than 100, or more than 500 times in the corpus, but at the cost of diminished coverage (fewer target words available with which to extend the Norwegian WordNet).

An alternative approach to using word embeddings to extend an existing WordNet has been described by Al tarouti and Kalita (2016), who in fact use word embeddings to extend an automatically-constructed Arabic WordNet built using the machine translation / bilingual dictionary method described by Lam et al. (2014). They leverage word embeddings to compute the cosine similarity of a) words within candidate synsets, and b) words within pairs of semantically-related synsets, allowing them to discard candidate synsets (and words within them) whose cosine similarity is below a given threshold value. 600 automatically-constructed word pairs (of synonym, hypernym, holonym, and meronym types) were evaluated by Arabic speakers using a 5-point Likert scale, with the average score then converted to a percentage – the resulting precision of the synonyms, hypernyms, holonyms, and meronyms was 78.4%, 84.4%, 90.4%, and 79.6% respectively, slightly higher than the precision (as a percentage) reported by Lam et al. (2014) for Arabic.

A similar method has also been described by Khodak et al. (2017), who report on the automatic construction of whole WordNets in French and Russian from scratch using bilingual dictionaries and word embeddings. After producing candidate synsets by finding the corresponding source language synsets for target language words as given by the bilingual dictionaries, word sense embeddings and word sense induction (WSI) techniques are used to cluster only the most relevant translations of lemmas from the source language synset together, ensuring that the correct target language candidate synset is 'matched' as correct. Evaluating these methods using subsets of 200 nouns, verbs and adjectives from each of the target language WordNets, the resulting $F_{\dot{5}}$ scores – used as a precision-centric alternative to the usual F_1 score – were reported to outperform those yielded using a baseline similarity method by 6% and 10% for French and Russian respectively.

5. Issues for Evaluating Automatically-Constructed WordNets

One of the biggest issues for the automatic construction of WordNets is how to properly and effectively evaluate their accuracy and/or precision. Across all of the different lexical resource and word embedding-based approaches to automatic synset construction described in the previous two sections, evaluation methods can be split across two types:

- Comparison against a reference WordNet,
- Manual evaluations against fixed samples of automatically constructed synsets.

Focusing first on comparisons with reference WordNets, much of the research referenced in the preceding sections reports on problems with this kind of evaluation. Khodak et al. (2017) describe an attempted comparison with their

[9] http://babelnet.org/

automatically-constructed WordNets and reference WordNets being difficult, with the ELRA French WordNet 2 being only around half the size of their new French WordNet and most Russian WordNets being a) even smaller, and b) not easily linked to (or compared with) PWN. Similarly, Taghizadeh and Faili (2016) cite 'the lack of correct links' in the pre-existing *FarsNet* (an ontology of Persian words mapped to PWN synsets) as being troublesome when attempting to compare their automatically-constructed Persian WordNet to it – they reported after comparing their automatically-constructed Persian WordNet to *Farsnet* that that the precision of their new WordNet was just 19% and its recall 49%, too low 'to be considered as a reliable resource'.

The discrepancies between size and coverage of reference WordNets and the original PWN can be viewed as an issue of granularity: PWN is large enough that its senses are fine-grained, and so several PWN synsets can generally mapped onto one synset in a reference WordNet (such as *FarsNet*) while other PWN synsets will not be present at all (Khodak et al., 2017). This makes it difficult, when comparing automatically-constructed WordNets to reference WordNets in a target language, to decide whether newly-created synsets are correct or not. For example, Taghizadeh and Faili (2016) use the following criteria to decide whether word and synset pairs in an automatically-constructed Persian WordNet are correct, using *FarsNet* as their reference WordNet:

- If a Persian word does not exist in *FarsNet*, it *IS NOT* correct,
- If a Persian word exists in *FarsNet* but is not linked to a PWN synset, it *IS NOT* correct,
- If a Persian word exists in *Farsnet* and and at least one PWN synset is linked to it:
  - If the automatically-constructed synset is not one of the linked PWN synsets in *FarsNet*, it *IS NOT* correct,
  - IF the automatically-constructed synset is one of the linked PWN sysnets in *FarsNet*, it *IS* correct.

Out of three options here, two of them lead to the word in the automatically constructed Persian WordNet being classed as incorrect – and even if the word is in both *FarsNet* and PWN, that word still has to be linked between *those* resources to be accepted as correct. This approach is therefore totally dependent on the quality of *FarsNet*, and any words in the automatically constructed WordNet that are in PWN and that *should* be in *FarsNet* will, unfortunately, be classed as incorrect. As Oliver and Climent (2014) – who considered an automatically extracted synset correct only if it was also present in a reference WordNet – highlight, automatic comparisons with reference WordNets inevitably mean that if the reference WordNets are not complete, then correctly extracted synsets in the automatically constructed WordNet can be evaluated as incorrect – and this is a major problem when reporting on their accuracy and legitimacy as a lexical resource.

Sand et al. (2017) also touch on potential discrepancies between automatically extracted synsets and their equivalent synsets in reference WordNets or in PWN, noting that hypernymy relations 'can be right or wrong by varying degrees'. They describe a 'soft accuracy' measure whereby the accuracy of an automatically extracted synset is weighted according to the number of links (or edges) between words in different synsets that separate a given word from what would be its correct position in a graph-based representation of the WordNet. Weighting the accuracy of automatically extracted synsets according to how comparable they are with their fully-formed PWN equivalents is certainly more logical than evaluating strictly on 'correct insertions' – an automatically extracted synset containing 8 of the 10 links to other words present in the same sysnet in PWN, for example, is surely more correct than an automatically extracted synset containing only 2 or 3 of the 10 links.

The alternative to automatic evaluations of synset correctness is of course manual evaluation, which is widely used both in isolation and in conjunction with automated evaluations in the works cited in Section 3. (Ruiz-Casado et al., 2005; Montazery and Faili, 2010; Lam et al., 2014; Taghizadeh and Faili, 2016). However, as Ruiz-Casado et al. (2005) point out, it is 'difficult to know how accurate manually-evaluated synsets are without some common guidelines. Some works simply describe having manual annotators decide if an automatically extracted is or is not semantically similar to a reference synset (Taghizadeh and Faili, 2016), while others – much more in line with the idea of weighting accuracy according to a degree of correctness (Sand et al., 2017) – have used a Likert scale for conducting manual evaluations (Lam et al., 2014).

6. Conclusions

This paper has examined a number of approaches to automatically construct and extend WordNets, taking into account both lexical and word embedding-based approaches to extracting synsets. A common trend across evaluations of WordNets extracted from lexical resources is that while the synsets themselves are reasonably precise, recall is often very low – that is, although extracted synsets are accurate when compared to reference WordNets such as PWN, not enough synsets are actually being extracted using automatic methods. However, these results do not necessarily paint the full picture – there are few agreed principles or common guidelines for evaluating extracted synsets, and it is often difficult to decide what constitutes a correctly or incorrectly extracted synset.

Automatic WordNet construction is a promising research area, particularly in the context of lesser-resourced languages – the fact that the works outlined in Sections 3. and 4. cover typically under-resourced languages such as Arabic, Catalan, Persian (Farsi), Romanian and Russian demonstrates that researchers see the value in exploring how to improve it. Constructing WordNets manually takes many years of linguistic knowledge, making them a costly and labour-intensive endeavour, but the availability of lexical resources in many languages and widespread adoption of approaches such as word embeddings for extract-

ing meaning and relationships from free text have made the concept of automatically constructing accurate WordNets in lesser-resourced languages very feasible.

This paper is intended to provide a summary of commonplace techniques in automatic WordNet construction and extension. In highlighting some of the underlying issues that have been a bottleneck for evaluating them, it is also hoped that it can serve as a first step in encouraging further discussion and the development of common guidelines for improving the area moving forward. A set of agreed principles that help researchers paint a clearer picture of what constitutes a correctly or incorrectly extracted synset will be an important next step in encouraging the automatic construction and extension of WordNets, particularly for those working with lesser-resourced languages.

References

Al tarouti, F. and Kalita, J. (2016). Enhancing Automatic WordNet Construction Using Word Embeddings. In *Proceedings of the Workshop on Multilingual and Cross-lingual Methods in NLP*, pages 30–34, San Diego, California. Association for Computational Linguistics.

Barbu, E. and Barbu Mititelu, V. (2005). Automatic building of Wordnets. In *Proceedings of International Conference on Recent Advances in Natural Language Processing*, RANLP 2005.

Bhattacharyya, P. (2010). IndoWordnet. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, LREC '10. European Language Resources Association.

Farreres, X., Rigau, G., and Rodríguez, H. (1998). Using WordNet for Building WordNets. In *Proceedings of the COLING-ACL '98 Workshop on Usage of WordNet in Natural Language Processing Systems*.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Khodak, M., Risteski, A., Fellbaum, C., and Arora, S. (2017). Automated wordnet construction using word embeddings. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 12–23, Valencia, Spain. Association for Computational Linguistics.

Lam, K. N., Al Tarouti, F., and Kalita, J. (2014). Automatically Constructing Wordnet Synsets. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers)*, ACL 2014, pages 106–111. Association for Computational Linguistics.

Montazery, M. and Faili, H. (2010). Automatic Persian WordNet Construction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 846–850. Association for Computational Linguistics.

Nayak, N. (2015). Learning hypernymy over word embeddings. Technical report, Stanford University.

Oliver, A. and Climent, S. (2014). Automatic Creation of WordNets from Parallel Corpora. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, LREC '14. European Language Resources Association.

Panchenko, A. (2016). Best of Both Worlds: Making Word Sense Embeddings Interpretable. In *Proceedings of the 10th Language Resources and Evaluation Conference*, pages 2649–2655, Portorož, Slovenia. European Language Resources Association.

Pianta, E., Bentivogli, L., and Girardi, C. (2002). MultiWordNet: Developing an Aligned Multilingual Database. In *Proceedings of the First International Global WordNet Conference*.

Rothe, S. and Schütze, H. (2015). Autoextend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, ACL 2015, pages 1793–1803, Beijing, China. Association for Computational Linguistics.

Ruiz-Casado, M., Alfonseca, E., and Castells, P. (2005). Automatic Extraction of Semantic Relationships for WordNet by Means of Pattern Learning from Wikipedia. In *Proceedings of the 10th International Conference on Natural Language Processing and Information Systems*, NLDB '05, pages 67–79. Association for Computational Linguistics.

Sand, H., Velldal, E., and Øvrelid, L. (2017). WordNet Extension via Word Embeddings: Experiments on the Norwegian WordNet. In *Proceedings of the 21st Nordic Conference of Computational Linguistics*, pages 298–302, Gothenburg, Sweden. Linköping University Electronic Press.

Taghizadeh, N. and Faili, H. (2016). Automatic Wordnet Development for Low-resource Languages Using Cross-lingual WSD. *Journal of Artificial Intelligence Research*, 56(1):61–87.

Vossen, P. (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Springer Netherlands.

Vossen, P. (2004). EuroWordNet: A Multilingual Database of Autonomous and Language-Specific WordNets Connected via an Inter-Lingual Index. *International Journal of Lexicography*, 17(2):161–173.

Linguistically-driven Framework for Computationally Efficient and Scalable Sign Recognition

Dimitris Metaxas*, Mark Dilsizian*, Carol Neidle**

*Rutgers University, **Boston University
*Rutgers University, CBIM, Department of Computer Science, 617 Bowser Road, Piscataway, NJ 08854
**Boston University Linguistics, 621 Commonwealth Ave., Boston, MA 02215
dnm@cs.rutgers.edu, mdil@cs.rutgers.edu, carol@bu.edu

Abstract

We introduce a new general framework for sign recognition from monocular video using limited quantities of annotated data. The novelty of the hybrid framework we describe here is that we exploit state-of-the art learning methods while also incorporating features based on what we know about the linguistic composition of lexical signs. In particular, we analyze hand shape, orientation, location, and motion trajectories, and then use CRFs to combine this linguistically significant information for purposes of sign recognition. Our robust modeling and recognition of these sub-components of sign production allow an efficient parameterization of the sign recognition problem as compared with purely data-driven methods. This parameterization enables a scalable and extendable time-series learning approach that advances the state of the art in sign recognition, as shown by the results reported here for recognition of isolated, citation-form, lexical signs from American Sign Language (ASL).

Keywords: Sign Recognition, Model-based Machine Learning, Computer Vision, American Sign Language (ASL).

1. Introduction

Automatic sign recognition is a difficult problem given the complexity of the linguistic structures in sign languages and the challenges in modeling 3D configurations and movements from 2D video. To address this problem, we use certain known linguistic properties of the language to structure the problem, and to inform, enhance, and correct visual recognition tasks. By combining these components into a unified optimization framework, we recognize isolated, citation-form lexical signs from American Sign Language (ASL) in a fully scalable manner.

Whereas prior vision-based approaches to sign recognition by computer had focused on detection of linguistically important components, such as handshape and motion trajectory, neural networks have recently been applied to the overall problem of end-to-end sign recognition without attending to linguistic structure. The framework described here (1) exploits recent discriminative neural net-based learning approaches, coupled with generative model-based methods, to improve the detection and analysis of the linguistically relevant components and features; (2) integrates knowledge of linguistic structures and dependencies to derive additional parameters; and (3) uses CRF learning methods to integrate these features for sign recognition. This enhances visual recognition capabilities for the critical sign components and offers a unified framework for sign recognition. This approach is successful working with limited quantities of annotated data and is scalable.

2. Previous Work

Previous computer vision research on sign recognition has generally focused on aspects of sign production known to be linguistically important, including analysis of handshapes, upper body pose, and movement trajectories.

Prior work on hand pose recognition in general includes Heap and Hogg (1996), Athitsos and Sclaroff (2001, 2003) and Tompson et al. (2014). Lu et al. (2003), Vogler and Metaxas (2004), and Isaacs and Foo (2004) focus specifically on the recognition of handshapes in sign languages. Yuntao and Weng (2000), Ding and Martinez (2007, 2009) , Ricco and Tomasi (2009), Thangali et al. (2011), Dilsizian et al. (2014), and Koller et al. (2016) constrain handshape recognition to fit those handshapes that are used linguistically in the sign language. Koller et al. (2016) is notable in the use of convolutional neural nets to achieve state-of-the art handshape recognition on a large dataset. Thangali et al. (2011) and later Dilsizian et al. (2014) leverage phonological constraints on start and end handshape co-occurrence to improve handshape recognition accuracy.

Other research has explored hand motion trajectories as an intermediate step towards sign recognition (Han, Awad, and Sutherland, 2009; Dilsizian et al., 2016; Pu et al., 2016). Ding and Martinez (2007, 2009) combine motion trajectories with face and hand configuration for sign recognition. Dilsizian et al. (2016) demonstrates the importance of 3D motion trajectories for sign recognition.

There have been some attempts to build full sign recognition frameworks for isolated signs, which have had limited success. Cooper, Holt, and Bowden (2011) combine 2D motion trajectories and handshape features to achieve 71.4% top-1 accuracy on 984 signs from a single signer. Wang et al. (2016) achieve 70.9% accuracy on 1000 isolated signs across multiple signers. Guo et al. (2016) propose an adaptive Gaussian Mixture Model HMM framework, and from vocabulary of 370 signs, they achieve a top-1 accuracy of 33.54% and a top-5 accuracy of 59.79%. However, they rely on an RGBD sensor for 3D information.

More recently there have been several purely data-driven end-to-end approaches to sign recognition from continuous signing based on Recurrent Neural Net (RNN) architectures (Cui, Liu, and Zhang, 2017; Koller, Zargaran, and Ney, 2017). However, the performance of these image-based approaches is held back by limitations in the data-

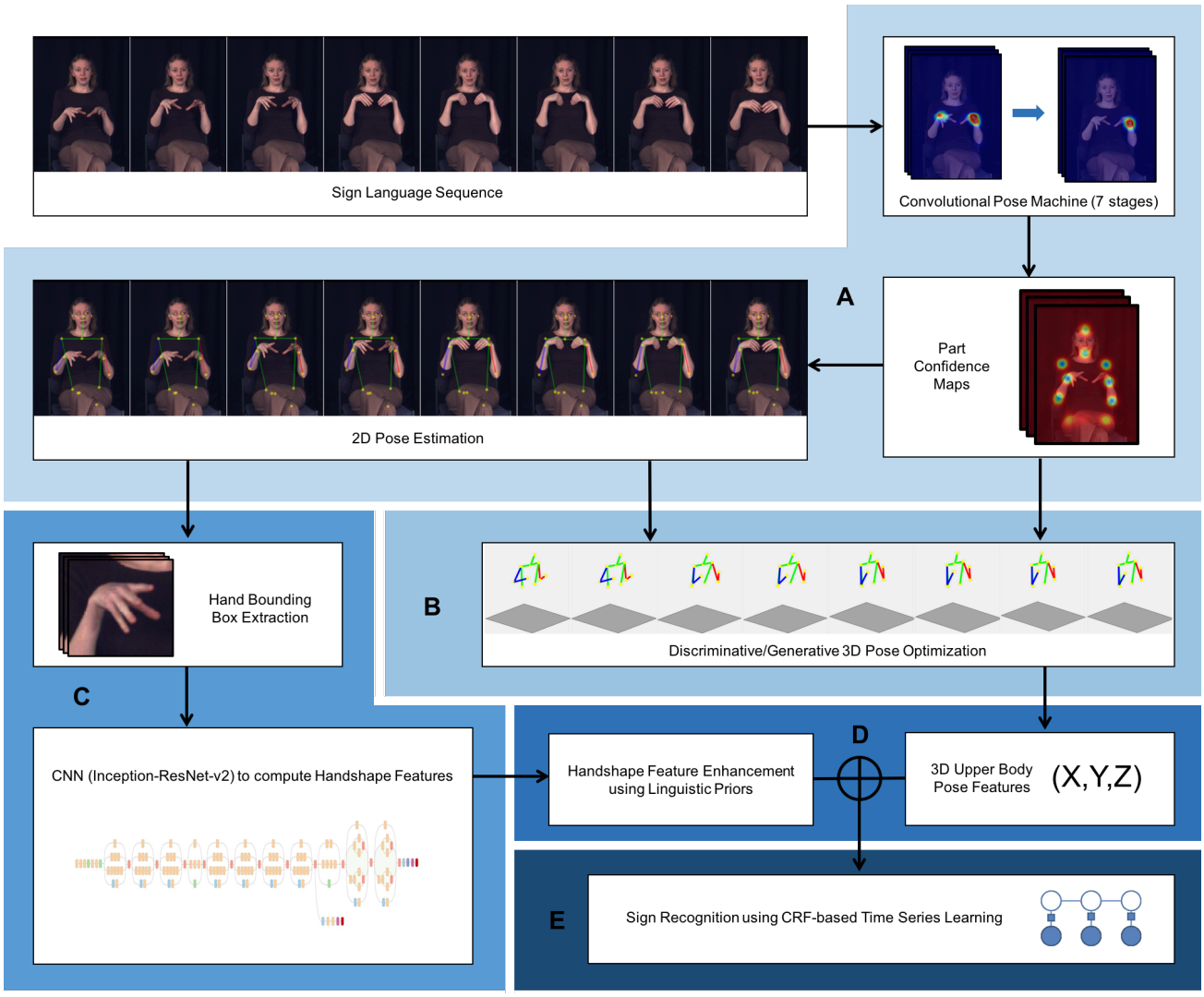


Figure 1: Framework Overview

sets and the fact that they do not integrate linguistic knowledge and perform 3D analysis. They report limited success in sign recognition, and these methods do not generalize well across multiple signers. The previous work in sign language recognition clearly demonstrates the need for a new computational approach.

3. New Framework for Sign Recognition

We propose a learning-based approach with three sub-components: 1) new discriminative learning-based computer vision methods (based on advances in deep learning) coupled with generative methods for hand and pose feature extraction and related parameters (Section 3.1); 2) additional linguistically driven parameters (Section 3.2), with enhancement of parameters from known linguistic dependencies (Section 3.3); and machine learning methods for sign recognition using the extracted parameters (Section 3.4).

This gives rise to a reduced parameterization and a significantly more efficient algorithm capable of coping with limited quantities of annotated data. This results in improved sign recognition compared to previous approaches. See Figure 1 for an overview of our framework.

.

3.1 Summary of Features Used for Sign Recognition

Using the methods to be described below, we estimated a comprehensive set of features (a total of 110), with regard to: **a)** handshapes, **b)** number of hands, **c)** 3D upper body locations, movements of the hands and arms, and distance between the hands, **d)** facial features and head movements (which have been shown to improve manual sign recognition (von Agris, Knorr, and Kraiss, 2008; Koller, Forster, and Ney, 2015)), and **e)** contact. The features for the face include 66 points (visible in Figure 2) from 3D estimates for the forehead, ear, eye, nose, and mouth regions, and their velocities across frames. The contact features are extracted from our 3D face and upper body movement estimation, and relate to the possibilities of the hand touching specific parts of the head or body. The parameter extraction is described in the next section.

3.2 Coupling of Discriminative and Generative Methods for Feature Extraction

In order to build a robust and scalable framework for sign recognition, we model individual components of the sign recognition problem. In this section, we present our methodology for upper body trajectory and handshape estimation, and related feature extraction.

3.2.1 Upper Body and Hand/Arm Movement Trajectories

Previous work has shown that tracking upper body pose, especially in 3D, is critical to sign recognition (Fillbrandt, Akyol, and Kraiss, 2003; Vogler and Metaxas, 2004; Zafrulla et al., 2011; Dilsizian et al., 2014). In our framework, we model upper body pose and use the 3D joint locations as features.

To develop an accurate 3D pose estimation suitable for ASL, we integrate state-of-the-art neural net-based 2D and 3D pose estimation (fine-tuned on ASL upper body videos) and a generative, deformable model-based fitting approach to further refine the 3D pose (Dilsizian, 2016). We start with a Convolutional Pose Machine (Wei et al., 2016) trained on a combination of the MPII human pose dataset (Andriluka et al., 2014) and the Kinect-based dataset for upper body pose in Dilsizian et al. (2016) to better match the 2D pose projection. Next, we use nearest neighbor matching with a 3D Pose library that includes the Human 3.6M dataset (Ionescu et al., 2014) and is also combined with the Kinect-based dataset from Dilsizian et al. (2016).

We formulate generative human pose recognition as a search problem in 3D (Euclidean) space (Dilsizian, 2016); solving this problem entails finding optimal pose parameters of a human model whose learned part appearance representation has the best matching score based on the image. Confidence maps returned by the Convolutional Pose Machine are used as the cost surface and inversely projected in 3D space along the tensor normal to the camera. The neural-net based 3D prediction is then used as an initialization to our generative approach to search for 3D candidate locations for each part.

In our novel 3D generative approach (Dilsizian, 2016), the global 3D upper body pose is modeled by an I-node relational graph representing I human skeleton parts and joints. Each node represents a part center, and edges denote skeletal links. For image X, the function

$$\varphi^{\lambda} : \boldsymbol{X} \times \boldsymbol{L} \rightarrow R^{d}$$

extracts features for C candidate 2D image projections

$$\boldsymbol{L} = l_1, l_2, \ldots, l_C$$

of each 3D candidate location

$$\mathbf{Y} = \mathrm{y}_1, \mathrm{y}_2, \ldots, \mathrm{y}_C$$

at the scale space associated with the depth parameter

$$\boldsymbol{\lambda} = \lambda_1, \lambda_2, \ldots, \lambda_C.$$

In order to conduct a local search for the j^{th} part across 3D candidate locations $\boldsymbol{Y}$j , we compute the corresponding 2D projections $\boldsymbol{L}_j = f_c(\boldsymbol{Y}_j)$ and the quantized depth parameters $\boldsymbol{\lambda}_j$ of the part. The local matching score makes use of the learned part templates $t^{\lambda j, \boldsymbol{mj}}$ and the local mixture parameter $\boldsymbol{b}_{\mathrm{i}}^{\boldsymbol{mj}}$; it is computed as:

$$\psi_j(\boldsymbol{Y}_j, \boldsymbol{m}_j) = \boldsymbol{t}_j^{\boldsymbol{\lambda}_j, \boldsymbol{m}_j} * \boldsymbol{\phi}_j^{\boldsymbol{\lambda}_j}(\boldsymbol{X}, f_c(\boldsymbol{Y}_j)) + \boldsymbol{b}_j^{\boldsymbol{m}_j} \quad (1)$$

These local scores are optimized efficiently through the use of a dynamic programming approach similar to those of Felzenszwalb and Huttenlocher (2005) and Yang and Ramanan (2011). The optimization includes the passing of a message $\boldsymbol{S}$ from child to parent nodes. The optimal parameters of a part i over the candidate locations, mixtures, and scales are obtained by optimizing the local score and the sum over each message S passed from each child node j:

$$\boldsymbol{\Psi}(\boldsymbol{Y}_i, \boldsymbol{m}_i) = \psi_i(\boldsymbol{Y}_i, \boldsymbol{m}_i) + \sum_{j \in \mathrm{child}(i)} \boldsymbol{S}_j(\boldsymbol{Y}_i, \boldsymbol{m}_i) \quad (2)$$

Message $\boldsymbol{Sj}$ makes use of the learned pairwise mixture parameter $\boldsymbol{b}_{\mathrm{ij}}$, as well as the pairwise distance term $\boldsymbol{w}$, which scores the relative location of part j with respect to its parent i.

$$\boldsymbol{S}_j(\boldsymbol{Y}_i, \boldsymbol{m}_i) = \max_{\boldsymbol{y}_j \in \boldsymbol{Y}_j, m_j \in \boldsymbol{m}_j} \left[\boldsymbol{\Psi}(\boldsymbol{y}_j, m_j) + \boldsymbol{w}(\boldsymbol{Y}_i, \boldsymbol{y}_j) + \boldsymbol{b}_{ij}^{\boldsymbol{m}_i, m_j} \right] \quad (3)$$

Equations 2 and 3 recursively compute the score of a part at each location and scale using the learned part templates. The global optimization is computed from the head node, which has index 1. We find the joint configuration with the maximum score:

$$\hat{\boldsymbol{y}}_1, \hat{m}_1 = \arg\max_{\boldsymbol{y}_1 \in \boldsymbol{Y}_1, m_1 \in \boldsymbol{m}_1} \boldsymbol{\Psi}(\boldsymbol{y}_1, m_1) \quad (4)$$

Starting with $\hat{\boldsymbol{y}}_1$ and $\hat{m}_1$, we backtrack through location and mixture indices to find all the part locations $\hat{\boldsymbol{Y}}$ of the optimal configuration.

Examples of 3D upper body pose reconstructions projected onto the image and from an alternative view can be seen in Figure 2.

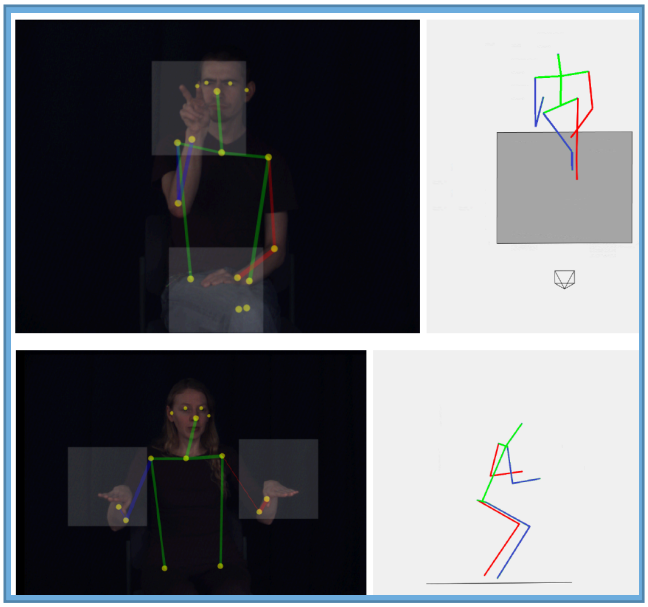


Figure 2: 3D upper body pose reconstruction and hand bounding boxes during an example of the signs glossed as LOOK (top row) and SHELF/FLOOR (bottom row). [The pictures depict the full body reconstruction that our system generates, although we only use the upper body for this research.]

Because our framework is capable of integrating a wide variety of disparate features, we can further mine the 3D trajectories for additional discriminative information. Another relevant set of features is extracted from linguistically recognized events, namely: we identify 1- vs. 2-handed signs, categorize hand touching events, and analyze motion trajectory. These additional features are particularly discriminative when combined with features relating to handshape appearance.

3.2.2 Handshape Parameters

Our parameterization also includes feature extraction from hand images specifically. We focus on the handshapes at the start and end of each sign, because those are the most linguistically informative handshapes.

We extract features derived from a neural net trained for handshape recognition. Additional features are then derived based on the relationship between handshapes on the dominant and non-dominant hands, as well as at the start and end of the sign (factoring in linguistic dependencies derived from frequencies of co-occurrence in our dataset).

In order to avoid overfitting and capture both the local and global appearance of the hand, we train Inception-ResNet-v2 (Szegedy et al., 2017) on hand images extracted from our upper body pose prediction. The handshape CNN returns top-1 accuracy of 70.1%; top-5 accuracy reaches 92.3%. However, we use the entire set of handshape probabilities from the output of the neural net as features for sign recognition.

3.3 Linguistic Structure

Once motion trajectories and handshape parameters have been extracted, we enhance features from Section 3.1 by focusing on properties known to be linguistically important and by leveraging known linguistic dependencies.

3.3.1 Upper Body Parameters Related to Contact

Some signs involve linguistically significant contact between the two hands, or hands contacting some specific part of the head, face, or body. We include parameters extracted from recognizing when such contact occurs and classifying the types of contact. For present purposes, the face and body were divided into different regions, based on the linguistically significant distinctions.

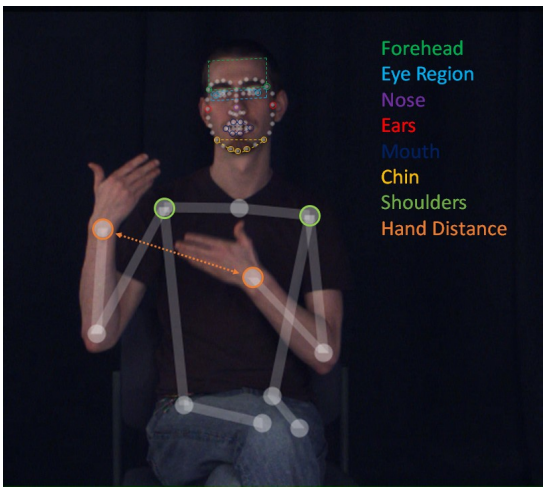


Figure 3: Visualization of Locations where Contact Occurs

Contact events are considered relative to 3D keypoints identified on the face and body from our pose estimation. A threshold is used for each touch event that is determined empirically based on what leads to improvement to sign recognition accuracy. Locations for contact events are visualized in Figure 3.

3.3.2 Number of Hands

Whereas some signs are produced using only one hand, others are normally produced using two hands (and in such cases, there are some dependencies with regard to what happens on the two hands, as discussed in Section 3.4). Thus we introduce an additional parameter related to this distinction. Our dataset of motion trajectories from citation-form ASL examples is used to train an HMM to predict 1- vs. 2- handed sequences from the dataset.

In summary, flags for signs involving contact between hands, face, and body, and flags for 1- vs. 2-handed signs function as additional parameters in our sign recognition feature vector.

3.4 Linguistic Dependencies

3.4.1 Dependencies between Start & End Handshapes

Following Thangali et al. (2011), Thangali (2013), and Dilsizian et al. (2014), we enhance handshape recognition by leveraging phonological constraints that hold between start and end handshapes in lexical signs, as reflected in the co-occurrence probabilities from our data set. After extracting, for each sign, the above per-frame start/end handshape parameters, we adjust those parameters based on the computation of the probabilities of co-occurrence of specific start and end handshapes. These co-occurrence statistics are multiplied by the joint probabilities of one window from the beginning of the sequence, which contains the start handshape, and one from the end, which contains the end handshape. The new start/end handshape priors are then used to adjust and improve handshape parameters for each frame of the sequence that falls in a start or end window.

3.4.2 Dependencies between Dominant & Non-dominant Handshapes in 2-handed Signs

In addition to enhancing handshape parameters through use of start/end handshape co-occurrence statistics, we extract another relevant parameter from comparing left and right handshapes. Many 2-handed lexical signs are produced with the same handshape on both hands. Furthermore, when the handshapes differ, the options for the non-dominant hand are severely reduced to a small set of unmarked handshapes. By computing the probability that the two hands are producing the same shape, the learning algorithm can benefit from this additional parameter with known discriminative value.

To compare the shapes, we compute a simple Mahalanobis distance between the dominant and non-dominant hand probability sets for each frame. This distance, which captures the similarity between the two handshapes, is added to our sign recognition feature vector.

3.5 Summary of Feature Vector Extraction

Based on our previously described hybrid approach, we assemble the features from the upper body and the handshapes into a feature vector to be used for sign recognition. This final feature vector consists, for each frame of a sign, of the features outlined in 3.1.

3.6 Sign Recognition

Our current sign recognition approach combines detection of upper body pose and hand configurations, the latter leveraging statistical properties resulting from linguistic constraints on sign formation, as just discussed. In doing this, we face several challenges. First, the signal for handshape recognition is noisy. Although for the training samples, we can rely on human annotations of start and end frames of a given sign (and these are the handshapes that are most important for sign identification), for the testing samples, there needs to be estimation of the start and end frames containing the handshapes to be taken as representative for the given sign. Second, the motion trajectories for a sign vary spatially and temporally from one instance to another of a given sign produced by same or different signers.

To process the motion trajectories, we normalize all upper body locations to the sternum location. In order to capture dependencies between our various features and to explicitly model the structure of the language, we employ a structured CRF-based method. We employ Hidden Conditional Ordinal Random Fields (HCORF), which explicitly model sequence dynamics as the dynamics of ordinal categories (Walecki et al., 2015); in our case, the ordinal categories are start and end handshape labels. We modify the HCORF objective function to include an additional error term that compares handshape predictions to ground truth labels for the two ordinal states (start/end).

Given normalized 3D body part locations (including the face and the head) and handshape features for each frame of a sequence, our resulting optimization minimizes the error of sign recognition while locally minimizing the error of start/end handshape prediction. As demonstrated in our Experiments section, this ordinal, structured approach is flexible and robust enough to overcome various types of failures in the different components of our framework.

In summary, our approach to sign recognition takes advantage of the fact that ASL has structure, and we achieve a significant reduction in the parameters used, which results in more efficient and robust ASL learning, as demonstrated in the next section.

4. Experiments and Results

4.1 Dataset

This research exploits the publicly accessible American Sign Language Lexicon Video Dataset (ASLLVD) (Neidle, Thangali, and Sclaroff, 2012).[1] This includes over 8500 examples corresponding to almost 2800 monomorphemic lexical signs in citation form from 6 native ASL signers. Although the entire ASLLVD dataset contains between 1 and 6 signers for all signs, we chose to use a subset of 350 signs, from among those with the highest numbers of signers and examples. On average, there were 4.7 signers and 6.9 total examples per sign for this set of 350 signs (a total of about 2400 examples). For each sign, 2 examples were randomly selected to be in the testing set, and the remaining examples were used for training.

4.2 Experiments

For each frame in each sequence, we extract a feature vector of dimension 110, which, as explained in Section 3.4, includes features for handshape, motion trajectory, and other linguistically motivated features discussed above. Then this feature vector is used as input to our modified HCORD-based framework for sign recognition. We trained on our data from 6 signers, using about 80% of the data for training and 20% for testing. We tested on vocabularies of differing sizes (175 vs. 350 signs) as a first step in demonstrating the efficiency and scalability of our approach. The set of 175 was chosen randomly from the signs in the larger set of 350.

We also performed a series of experiments to separate out the contributions of the different parameters, including those based on linguistically motivated features. This linguistic parameterization is especially useful in the current context of sign recognition research, where large amounts of data with ground truth are not available.

4.3 Results

4.3.1 Recognition Accuracy

As shown in Figure 4, from a vocabulary of 350 signs (including both 1- and 2-handed signs), using all of our parameters, we achieve a top-1 accuracy of 93.3% and a top-5 accuracy of 97.9%.

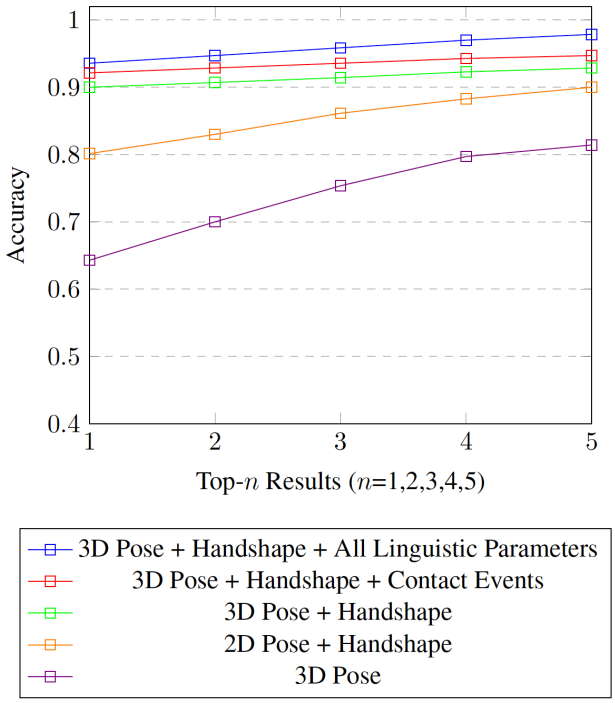


Figure 4: ASL sign recognition accuracy for top-n predictions over a 350 sign vocabulary.

[1] See http://www.bu.edu/av/asllrp/dai-asllvd.html. This dataset is also available at http://secrets.rutgers.edu/dai/queryPages/search/search.php and forms the basis for our new Web-accessible ASLLRP Sign Bank, accessible at http://dai.cs.rutgers.edu/dai/s/signbank (Neidle et al., 2018).

Figure 4 demonstrates the importance of integrating the different features computed by our framework. Sign recognition based solely on 3D upper body trajectories (violet) achieves reasonable accuracy. Although the handshape features by themselves are not sufficient to recognize signs, when combined with 3D trajectories, a significant boost in recognition accuracy is achieved (green). The addition of the contact events (shown in red) and linguistic information (blue) also improves recognition accuracy, particularly when considering top-5 accuracy, where it penalizes low probability observations that would result in impossible or improbable start/end combinations.

In addition, the importance of 3D trajectories over 2D is demonstrated here. The use of 3D motion trajectories (green) results in a significant boost in performance over 2D (yellow). This shows that there is significant discriminatory information built into the depth component of the motion trajectories. This important dimension is captured through the explicitly modeling of upper body 3D pose, and would not be captured by an end-to-end sign recognition CNN that implicitly encodes 2D motion patterns.

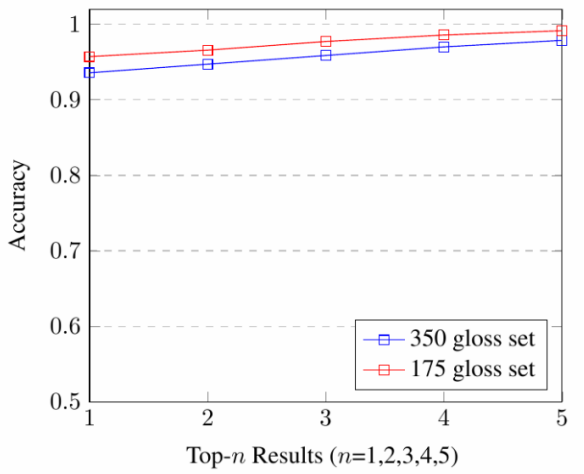


Figure 5: Comparing the Results on Vocabularies of 175 vs. 350 Signs

When increasing the vocabulary size from 175 to 350 signs (Figure 5), accuracy declines by only 2.1% for top-1, and by only 1.3% for top-5. This provides evidence for the scalability of the approach.

4.3.2 Error Analysis

An analysis of the errors in sign recognition revealed that some of the confusion involved signs with strong similarities in handshapes or movement trajectories. For example, there was one case where CAN was confused with COLD. Both signs involve closed fists in front of the body and have a similar movement pattern, but they differ critically in orientation (with the palm/fist facing downward for CAN but sideward, facing the center of the body (and shaking a bit from side to side), for COLD) as seen in Figure 6. Other recognition errors similarly involved orientation of the hands or confused locations relative to the body, or handshapes or movement patterns in some cases.

In any case, the success rate already achieved through the reduction in the number of parameters offers promise for scalability of these methods to large databases. We will investigate modifying the current parameterizations to better capture certain types of distinctions that are linguistically significant but that were not reliably exploited in the current set of results. For example, we expect that further improvements can be achieved by incorporating additional information about linguistic dependencies related to movement patterns of the two hands.

4.4 Discussion

There is a limited number of previous studies on isolated sign recognition available for comparison. In addition, many of the reports in the literature are for different sign languages (e.g., (von Agris et al., 2006; von Agris, Knorr, and Kraiss, 2008; Cooper, Holt, and Bowden, 2011; Wang et al., 2016)). Furthermore, research focused on ASL generally uses datasets we don't have (which in many cases contain small numbers of signs (e.g., (Zahedi et al., 2005; Zaki and Shaheen, 2011)), and/or the authors do not provide enough details or code to enable direct comparisons).

One relevant comparison is Guo et al. (2016), which uses a dataset of a size comparable to ours (370 signs) and a similar number of signers (5). Despite using RGBD data and a number of examples for training about 4 times greater than in our experiments, their adaptive GMM-HMM method results in recognition accuracy of only 33.54% for top-1, 59.79% for top-5, and 69.41% for top-10.

Conly (2016) is based on the same ASLLVD data we use, but he supplements that with additional data that he



Figure 6: CAN on the left (hands move downward); COLD on the right (hands move side to side)

collected that include depth information; and, despite that, and even taking into account the fact that he is working with a larger set of signs, he still gets quite a bit lower recognition accuracy. From a vocabulary of just over 1,100 signs, Conly reports the correct sign match 14.7% (top-1) of the time, and 36.0% for top-5. Although we use a smaller set of signs, 350 total, we get the correct match 93.3% (top-1) and 97.6% (top-5) of the time.

Thus, our recognition accuracy compares favorably with the two approaches just mentioned, and furthermore has the advantage of being scalable.

5. Conclusions

We have established a framework for sign recognition that relies on combining 3D modeling of start and end hand-shapes as distributions based on a few initial and final frames (enhanced by statistical information about their linguistic dependencies) and of 3D movement patterns of the hands, arms, and upper body during sign production. In particular, we have developed a statistical approach that combines distributions of the initial and final hand shapes and pose coupled with the spatiotemporal patterning of the arm and upper torso. Using this approach, we achieve high accuracy for recognition of ASL signs. This statistical parameterization of the linguistically important components of lexical signs makes it possible to employ learning methods that can take advantage of large amounts of data relevan t to each parameter, without requiring large numbers of examples of each individual sign in the vocabulary. As a result of the use of a reduced parameter representation, this method will also scale to larger sign vocabularies. To improve our sign recognition results in the future, we intend to expand the proposed parameterization to incorporate additional linguistic information about location, orientation, and movement patterns that are relevant to discrimination of signs. The ability to incorporate such improvements represents yet another advantage over purely data-driven approaches.

6. Acknowledgments

The research reported here has been supported in part by grants from the National Science Foundation (#1748016, 1748022, 1059218, 0705749, 0958247, 1703883, 1567289, 1555408, 1451292, 1447037). We are also grateful for contributions from many students and colleagues at Boston, Gallaudet, and Rutgers Universities. Thanks also to our many ASL linguistic consultants. We wish also to thank Stan Sclaroff, Ashwin Thangali, Joan Nash, and Vassilis Athitsos for their participation in data collection efforts for the ASLLVD, the dataset that served as the basis for the research reported here.

7. Bibliographical References

Andriluka, M., Pishchulin, L., Gehler, P., and B., S. (2014). 2D Human Pose Estimation: New benchmark and state of the art analysis. *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*.

Athitsos, V. and Sclaroff, S. (2001). 3D hand pose estimation by finding appearance-based matches in a large database of training views. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2001*.

Athitsos, V. and Sclaroff, S. (2003). Estimating 3D hand pose from a cluttered image. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2003,* Wisconsin.

Conly, C. (2016) Improving Accuracy in Large Vocabulary Sign Search Systems. Unpublished Doctoral Dissertation, University of Texas at Arlington.

Cooper, H., Holt, B., and Bowden, R. (2011) Sign Language Recognition. In Moeslund, T. B., Hilton, A., Krüger, V. and Sigal, L., (eds.) *Visual Analysis of Humans: Looking at People*: Springer. pp. 539–562.

Cui, R., Liu, H., and Zhang, C. (2017). Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization. *Proceedings of the CVPR 2017,* Honolulu, Hawaii.

Dilsizian, M., Yanovich, P., Wang, S., Neidle, C., and Metaxas, D. (2014). New Framework for Sign Recognition based on 3D Handshape Identification and Linguistic Modeling. *Proceedings of the LREC 2014, Reykjavik, Iceland. May 2014.*

Dilsizian, M. (2016) Hybrid Discriminative-Generative Methods for Human Pose Reconstruction from Monocular Imagery. Unpublished PhD Dissertation, Rutgers University.

Dilsizian, M., Tang, Z., Metaxas, D., Huenerfauth, M., and Neidle, C. (2016). The Importance of 3D Motion Trajectories for Computer-based Sign Recognition. *Proceedings of the 7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining. LREC 2016, Portorož, Slovenia. May 2016.*

Ding, L. and Martinez, A.M. (2007) Recovering the linguistic components of the manual signs in American Sign Language. *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, September 5-7, 2007.*, pp. 447-452.

Ding, L. and Martinez, A.M. (2009) Modelling and Recognition of the Linguistic Components in American Sign Language. *Image and Vision Computing, 27*(12), pp. 1826-44.

Felzenszwalb, P.F. and Huttenlocher, D.P. (2005) Pictorial Structures for Object Recognition. *International Journal of Computer Vision, 61*(1), pp. 55–79.

Fillbrandt, H., Akyol, S., and Kraiss, K.-F. (2003) Extraction of 3D Hand Shape and Posture from Image Sequences for Sign Language Recognition. *AMFG, Vol. 3*.

Guo, D., Zhou, W., Wang, M., and Li, H. (2016). Sign Language Recognition Based on Adaptive HMMs with Data Augmentation. *Proceedings of the IEEE Conference on Image Processing*.

Han, J., Awad, G., and Sutherland, A. (2009) Modelling and segmenting subunits for sign language

recognition based on hand motion analysis. *Pattern Recognition Letters, 30*(6), pp. 623-633.

Heap, T. and Hogg, D. (1996). Towards 3D Hand Tracking Using a Deformable Model. *Proceedings of the International Conference on Automatic Face and Gesture Recognition*.

Ionescu, C., Papava, D., Olaru, V., and Sminchisescu, C. (2014) Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(7), pp. 1325-1339.

Isaacs, J. and Foo, S. (2004) System Theory. Proceedings of the Thirty-Sixth Southeastern Symposium on IEEE, 2004. *Hand pose estimation for American sign language recognition*.

Koller, D., Ney, H., and Bowden, R. (2016). Deep Hand: How to Train a CNN on 1 Million Hand Images when your Data is Continuous and Weakly Labelled. *Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Koller, O., Forster, J., and Ney, H. (2015) Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding, 141*, pp. 108-125.

Koller, O., Zargaran, S., and Ney, H. (2017). Re-Sign: Re-Aligned End-To-End Sequence Modelling With Deep Recurrent CNN-HMMs. *Proceedings of the CVPR 2017,* Honolulu, Hawaii.

Lu, S., Metaxas, D., Samaras, D., and Oliensis, J. (2003). Using Multiple Cues for Hand Tracking and Model Refinement. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2003*.

Neidle, C., Thangali, A., and Sclaroff, S. (2012). Challenges in Development of the American Sign Language Lexicon Video Dataset (ASLLVD) Corpus. *Proceedings of the 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon. LREC 2012, Istanbul, Turkey. May 2012*.

Neidle, C., Opoku, A., Dimitriadis, G., and Metaxas, D. (2018). NEW Shared & Interconnected ASL Resources: SignStream® 3 Software; DAI 2 for Web Access to Linguistically Annotated Video Corpora; and a Sign Bank. *Proceedings of the 8th Workshop on the Representation and Processing of Sign Languages: Involving the Language Community. LREC 2018, Miyagawa, Japan. May 2018.* .

Pu, J., Zhou, W., Zhang, J., and Li, H. (2016). Sign Language Recognition based on Trajectory Modeling with HMMs. *Proceedings of the International Conference on Multimedia Modeling*.

Ricco, S. and Tomasi, C. (2009) Fingerspelling Recognition through Classification of Letter-to-Letter Transitions. *9th Asian Conference on Computer Vision, Xi'an, China*.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A.A. (2017) Inception-v4, inception-resnet and the impact of residual connections on learning. *AAAI*, pp. 4278–4284.

Thangali, A., Nash, J.P., Sclaroff, S., and Neidle, C. (2011). Exploiting Phonological Constraints for Handshape Inference in ASL Video. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2011*.

Tompson, J., Stein, M., LeCun, Y., and Perlin, K. (2014) Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (ToG), 33*(5), pp. 169.

Vogler, C. and Metaxas, D. (2004) Handshapes and movements: Multiple-channel ASL recognition. *Springer Lecture Notes in Artificial Intelligence (Proceedings of the Gesture Workshop '03, Genova, Italy), 2915*, pp. 247-258.

von Agris, U., Schneider, D., Zieren, J., and Kraiss, K.-F. (2006). Rapid signer adaptation for isolated sign language recognition. *Proceedings of the Workshop on Vision for Human Computer Interaction (V4HCI)*.

von Agris, U., Knorr, M., and Kraiss, K.F. (2008). The significance of facial features for automatic sign language recognition. *Proceedings of the International Conference on Automatic Face & Gesture Recognition*.

Walecki, R., Rudovic, O., Pavlovic, V., and Pantic, M. (2015). Variable-state latent conditional random fields for facial expression recognition and action unit detection. *Proceedings of the 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2015, IEEE*.

Wang, H., Chai, X., Hong, X., Zhao, G., and Chen, X. (2016) Isolated Sign Language Recognition with Grassmann Covariance Matrices. *ACM Transactions on Accessible Computing (TACCESS), 8*(14).

Wei, S.-E., Ramakrishna, V., Kanade, T., and Sheikh, Y. (2016). Convolutional pose machines. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Yang, Y. and Ramanan, D. (2011). Articulated Pose Estimation with Flexible Mixtures-of-parts. *Proceedings of the Computer Vision and Pattern Recognition (CVPR), IEEE, 2011*.

Yuntao, C. and Weng, J. (2000) Appearance-based Hand Sign Recognition from Intensity Image Sequences. *Computer Vision and Image Understanding, 78*(2), pp. 157-176.

Zafrulla, Z., Brashear, H., Starner, T., Hamilton, H., and Presti, P. (2011). American Sign Language Recognition with the Kinect. *Proceedings of the 13th international conference on multimodal interfaces. ACM, 2011*.

Zahedi, M., Keysers, D., Deselaers, T., and Ney, H. (2005) Combination of Tangent Distance and an Image Distortion Model for Appearance-Based Sign Language Recognition In *Pattern Recognition*, Berlin / Heidelberg: Springer. pp. 401-408.

Zaki, M.M. and Shaheen, S.I. (2011) Sign Language Recognition using a Combination of New Vision Based Features. *Pattern Recognition Letters, 32*(4), pp. 572-77.

CONDUCT: An Expressive Conducting Gesture Dataset for Sound Control

Lei Chen, Sylvie Gibet, Camille Marteau
IRISA, Université Bretagne Sud
Vannes, France
{lei.chen, sylvie.gibet}@univ-ubs.fr, cam.marteau@hotmail.fr

Abstract

Recent research in music-gesture relationship has paid more attention on the sound variations and its corresponding gesture expressiveness. In this study we are interested by gestures performed by orchestral conductors, with a focus on the expressive gestures made by the non dominant hand. We make the assumption that these gestures convey some meaning shared by most of conductors, and that they implicitly correspond to sound effects which can be encoded in musical scores. Following this hypothesis, we defined a collection of gestures for musical direction. These gestures are designed to correspond to well known functional effect on sounds, and they can be modulated to vary this effect by simply modifying one of their structural component (hand movement or hand shape). This paper presents the design of the gesture and sound sets and the protocol that has led to the database construction. The relevant musical excerpts and the related expressive gestures have been first defined by one expert musician. The gestures were then recorded through motion capture by two non experts who performed them along with recorded music. This database will serve as a basis for training gesture recognition system for live sound control and modulation.

Keywords: Corpus, sound-control gestures, expressive gesture, conducting

1. Introduction

The use of 3D mid-air gestures for controlling sound interactively has gained attention over this past decade (R. Aigner et al., 2012) One of the main difficulties to design such interfaces is the lack of guidelines to help designers in creating a set of gestures that can be significant for the given application and adopted by a large number of users.

The purpose of this study is to be able to control and modulate a live sound with expressive gestures. In live performances indeed the use of human movements to control both the visual and sound outputs have brought an intuitive and natural dimension to the interaction, leading to unprecedented performances, rich in body sensations and artistic creation. Among gestures controlling sound we are seeking a subset of meaningful and expressive gestures performed in the air, without any instrument. Furthermore, we expect the message conveyed by each gesture to correspond to an underlying desired function, and the quality encoded in the movement to correspond to an understandable expressive intent. The gestures should also be sufficiently different from each other, so that they may be automatically discriminable and recognizable. And finally, as in high-level languages, the structure of the produced gestures should allow to efficiently carry expressive variations by changing only a few gesture features such as kinematics, dynamics, geometry, or hand shape features.

Our search for a gestural language to drive sound applications naturally led us to consider conducting gestures performed by orchestral conductors. These gestures are of particular interest, since they gather most of the mentioned properties. Moreover, they are highly coded gestures which are potentially understandable by most of musicians around the world (Meier, 2009). Even if each conductor has his own style and defines his own set of gestures, we may find a subset of those gestures that share common meaning and form.

Most descriptions of the conducting gestures are concerned by gestures executed by the dominant hand, i.e. beating gestures that indicate the structural and temporal organization of the musical piece (tempo, rhythm) and lead to precise, efficient, and unambiguous orders to the orchestral musicians. Other gestures performed by the non dominant hand are dedicated to show other aspects of the music execution and interpretation, among which variations in dynamics and intensity, musical phrasing or articulation, accentuation, entries and endings, sound quality and color, etc. These musical cues are sometimes noted as specific symbols or semantic terms in the score, but there is no agreement between musicians, musicologists or composers to precisely define the meaning of these additional notations on the scores. When the musical piece is performed, these indications can be translated into gestures that express the desired musical expression. For the musician, it comes down to interpreting, through his instrumental gesture, the nuance indicated on the score. For the conductor, the interpretation is understood more generally at the level of the ensemble of the musicians and the intention of the musical excerpt, and his gestures indicate in a slightly anticipatory manner the nuance that the musicians must achieve. We propose here to define a new data set of expressive gestures that are inspired by these conducting gestures performed by the non dominant hand. Our gesture selection has been partly guided by the joint work of professional conductors and linguists studying sign languages (Braem and Bräm, 2001). However, since we target non musicians in our interactive application, we chose intuitive, expressive, and easy to perform gestures, in order to efficiently control a musical performance.

In this paper, we present the design and construction of this new multimodal data set, called CONDUCT, composed of associated gestures and musical excerpts. To this end, we defined a set of interacting gestures that are likely to express and control the performance of music excerpts. We have identified four functional categories of gestures that corre-

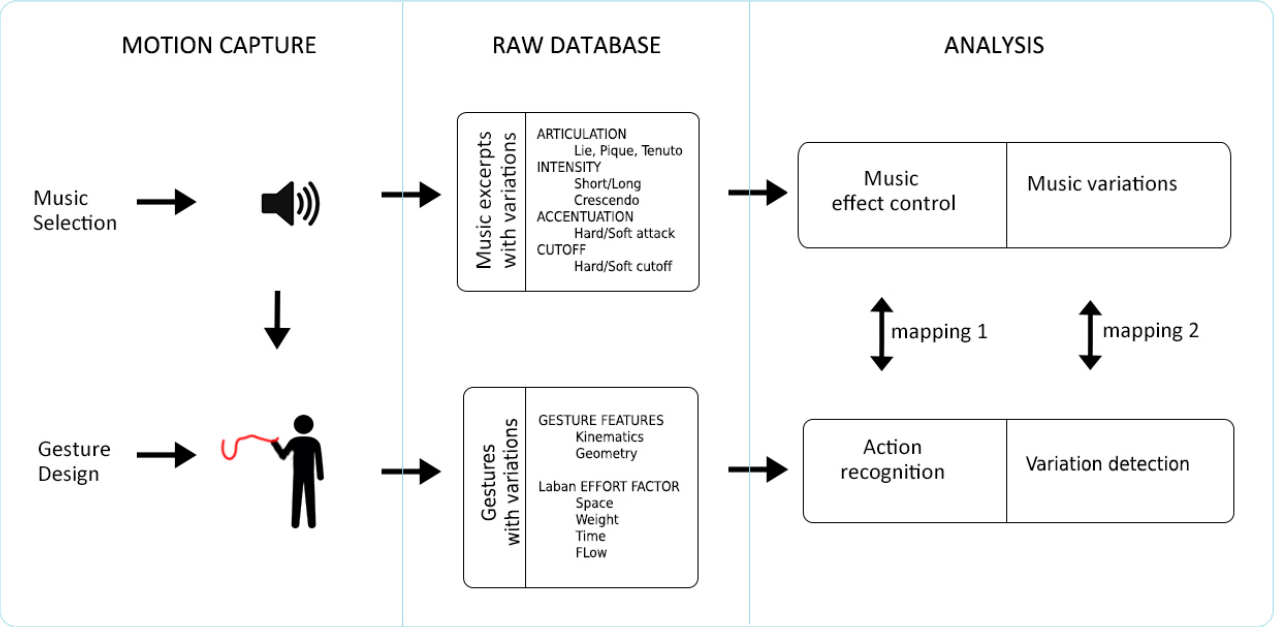


Figure 1: Work flow to collect the data and analyze them for the application needs.

spond to classically used musical effects. These categories relate to indications in Articulation, Intensity, Attack, or Cutoff. Within each category, we have characterized several expressive variations.

The paper is organized as follows: section 2 describes the main principles that have led to the design of the data set, both for the sound excerpts and for the design of associated gestures. Section 3 details the experimental setup, section 4 presents a first analysis of the data and section 5 concludes the paper and raises the main perspectives.

2. Overview of the Study

Figure 1 illustrates the way the data are collected (motion capture and sound recording data) and the future use of the database (recognition of actions and their variations). Concerning the experimental protocol, we first selected a set of appropriate musical excerpts that illustrate the main categories of sound effects (Articulation, Intensity, Accentuation, Cutoff) and their variations within each category. Our grammar of gestures follows the same protocol, by defining one specific gesture per category and variation.

Many researchers have designed experiments to record and analyze music-related gestures. Previous work has shown that gesture-sound correspondence is mutual, i.e. musical gestures help to understand the music, and, conversely, gestures can be used in turn to control the music. (Godøy et al., 2005) have studied *air-instrument* playing gestures where subjects were asked to observe real musician performances and imitate them as if they were playing the piano. In another study, they have asked the subjects to listen to the sound and draw induced sketches on a tablet (Godøy et al., 2006). Both studies contribute to the understanding of gesture-sound mapping. F. Bevilacqua et al. also proposed a procedure to learn the gesture-sound mapping where the users' gestures are recorded while they are listening to specific sounds (Bevilacqua et al., 2007) (Bevilacqua et al., 2011). On the basis of these theoretical studies, J. Françoise et al. have design several gesture-to-sound applications (Françoise et al., 2012) (Françoise et al., 2013). This work supports the approach we have adopted, by building the database from the music guiding the conductor rather than vice versa.

As illustrated in the right part of Figure 1 (Analysis), the aim of this gesture database is to serve as a basis for further gesture recognition for live sound control. Each recognized action will directly affect the sound effect (whether real or synthesized), and each detected gesture variation will correspond to a sound variation. In our approach the gestures are codified following sign language compositional structure, by identifying for each gesture the couple of components (hand movement, hand shape). With such a structure, it is possible to express the gesture variations as one modulation of either the hand movement (varying for example the form of the trajectory or the kinematic of the movement), or the hand shape (varying the shape of the hand). Previous work in sign recognition from video has demonstrated some success on isolated and continuous signing. Adding some linguistic knowledge about the composition of lexical signs considerably improves the performances of the recognition system (Dilsizian et al., 2014). Although these results are not achieved on motion capture data, they are promising for our training recognition system based on similar linguistic components.

3. Corpus Design

The purpose of this new data set was to be able to modulate a live sound with expressive gestures. From this hypothesis, the first questions that arose were: Which variables of the sound should we change? How much? In which categories should we consider such variations? Which variations should we take into account within the category? What qualifies as an expressive gesture? How is it related

to the sound variation? Instead of directly defining a set of non dominant hand gestures, as was done in (Braem and Bräm, 2001), we first listed different sound effects and grouped them in main functional categories. Among each category, we identified different variations that are commonly found in musical scores. We have then defined possible conducting-like gestures corresponding to these categories and variations, strongly inspired by the structural and iconic description of deaf sign languages.

3.1. Sound Categories and Variations

A musical score provides all the information needed to build an interpretation of the musical piece. Through this score, musicians (instrumentalists, singers) are able to read a musical piece and transform it in a gestural performance with musical sound. The challenge of the conductor is to have a global idea of the composer's musical intention, to imagine sounds and colors, and to read and understand all the scores of all the instruments. Among the information not contained in the temporal organization (tempo, rhythm) of the musical excerpt, we have identified four main categories: Articulation, Intensity, Attack, and Cutoff.

- The Articulation category is related to the phrasing of the musical discourse which is strongly dependent on the style of the piece. This expresses the way in which specific parts or notes in a piece are played within the musical phrasing, how they are linked and co-articulated, taking into account the timing and the quality of the musical sequencing. Among the techniques of articulation, we retained three of them: *Legato* (linked notes), *Staccato* (short and detached notes), and *Tenuto* (completely held and sustained). In our examples, we are aware that these terms and their meaning might differ according to the instrument and the musical context.

- The Intensity category, also called Dynamics in musicology, characterizes the music loudness. In our study we are interested in variations of intensity. These variations can be progressive (smooth) or abrupt with an increase or decrease in intensity. Four Intensity variations have been retained: *Long Crescendo*, *Long/Medium Decrescendo*, *Short Crescendo*, *Short Decrescendo*.

- The Attack category gathers different types of accents which are indicated in the score by different symbols but also by terms such as *sforzato (sfz)*. In our study, we have identified two main discriminating attacks: *Hard hit*, *Soft Hit*.

- The Cutoff category expresses the way a musical phrase ends. We have retained two main variations within this last category: *Hard Cutoff*, *Soft Cutoff*.

These categories and variations have been retained for two kinds of musical excerpts:

- Orchestral classical music; some of the excerpts are taken from conducting scores (Meier, 2009);

- Two musical phrases with different variations played on a piano (one variation at a time, keeping the same tempo); we have chosen excerpts extracted from work of J. S. Bach: Prelude no. 1 in C Major, and Cantate Bwv 147.

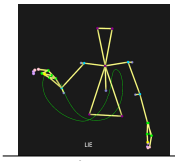
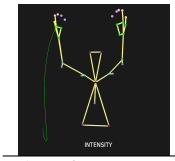
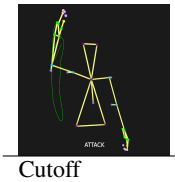
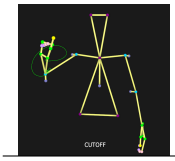
We also aim to add to this sound database two synthesized musical phrases similar to the above piano playing excerpts from J.S. Bach, along with corresponding generated variations for the four categories of actions.

3.2. Grammar of Gestures and their Modulation

For the conductor, the challenge is to interpret along the musical piece the sound effects and to give proper orders in terms of physical gestures to the musicians. Our aim was to find a set of intuitive and expressive gestures, in adequacy with the sound categories and variations. Beyond command and coverbal gestures (those that accompany the speech), several gestural languages have been developed for the exclusive expression by means of gestures. This is the case with gestures used by conductors or choirmasters, for whom it is impossible to express verbally. In addition, the need to teach the musical direction and to transmit it over time requires the definition of a form of codification of gestures and the definition of a written grammar. Another community that invented languages to express themselves and communicate with gestures is the one that practices the sign languages of the Deaf. Interestingly is the fact that there are strong similarities between these communities expressing themselves through gestures: they say by showing, they exploit the iconicity of gestures, and they mimic actions that make sense. Inspired by expressive conducting gestures we chose a subset of gestures that are shared by most of musicians around the world (Meier, 2009). In this study, we selected gestures composed of short actions, so that they can be used as isolated gestures, or repetitively as flows of actions. These gestures should implicitly contain the richness and the expressiveness of the conducting gestures, and they should be sufficiently simple to be codified and used in real-time through an automatic recognition process.

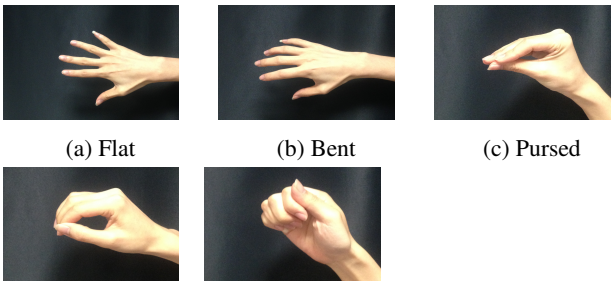
A novel grammar of gestures has therefore been defined in Table 1, whose structure is closely linked to sign language gestures. These gestures share indeed some common properties with conducting gestures. They are both visual-gestural languages: the produced information is multimodal and conveyed by hand-arm movements, facial expression and eye's gaze direction, and the perceivable information is interpreted by visual receptors, which means that the gestures have to be well articulated and visually comprehensible. Both of them have also a metaphoric function, i.e. they may fall into the category of describing objects (iconic gestures that describe the shape and the size of an object), or manipulating objects (like hitting or touching an object). As they may accurately express various meanings, they are characterized by a linguistic structure that can be described by a set of components also called phonological units in sign languages, including the location of the gesture, the movement, the hand shape, the wrist orientation and the facial expression. Therefore, as in high-level lan-

Table 1: List of gestures with their movement and handshape description.

Gesture with demo	**Hand movement**	**Handshape**
Articulation 	Phrasing gesture: Legato (sagital plane); Staccato (horizontal); Tenuto (vertical)	Flat/Bent for Legato or Tenuto, O for Staccato
Intensity 	Upward (crescendo) or downward (decrescendo)	Alternating a Flat and a Bent handshape
Attack 	Metaphor of "hitting a hard object"	O for a hard attack and Fist for a soft attack
Cutoff 	Metaphor of "taking an object out of the view"	Flat handshape closes to a Pursed (soft cutoff) or O (hard cutoff) handshape.

guages, the combination of these components allow to efficiently express various gestures and sequences of gestures. Modulating these gestures to express various meanings and nuances in the desired performances can be achieved by changing only one or few components (like the hand shape, or the kinematics and dynamics of the movement).

Our gesture database is composed of hand-arm gestures, and we only retain as structural components the hand shape and the hand movement. In sign languages the number and nature of hand shapes change according to the context of the corpus (around 60 basic hand shapes are identified in French Sign Language). This number is much more limited for conducting gestures. In our database we selected five basic hand shapes, as illustrated in Figure 2.



(a) Flat (b) Bent (c) Pursed

(d) O (e) Fist

Figure 2: List of the five selected handshapes.

4. Experimental Protocol

To capture the conducting gestures, we used a Qualisys motion capture system which is a marker-based system composed of 12 high speed and high resolution cameras and one video camera. As these gestures mostly involve the hands and the upper body, we have fixed 8 markers on each hand, and 18 markers on the upper body, which counts for 34 markers in total. Figure 3 shows the MoCap marker setting. The MoCap frequency rate was set to 200 Hz to deal with the rapid movements of the hands. Two subjects of different musical levels (one non musician, and one musician highly graduated in instrument playing) participated in the experiment, both of them being right-handed. We call S_1 and S_2 these subjects.

Previously to the experiment, one expert musician (called E), with extensive experience in orchestra, chamber music and instrumental violin practice, chose the musical excerpts, designed the corresponding gestures, and evaluated them qualitatively.

The MoCap recording experimentation can be described as follows: the musical excerpts were played as many times as necessary and each subject S_1 and S_2, in two different sessions (each session lasted about four hours), was instructed to execute the corresponding gestures according to a specific musical effect. For each gesture, several variations were defined, induced by the musical excerpts. The users had to perform each gesture after a short training session, by visualizing some referent videos performed by the expert musician E. Each variation was expressed in several different musical excerpts. For example, for the orchestra music, we had about three excerpts per variation. Moreover, within the same musical excerpt, we could have different nuances of the same variation played at different times of the excerpt (for example several attacks, or several cutoffs). Each musical excerpt was played several times (up to 10 times).

For each subject, two recording sessions were considered, for two classes of classical music pieces:

- 30 classical orchestra music excerpts; these excerpts were validated by the expert musician E;
- two piano excerpts from J.S. Bach played by a piano expert with the instructed variations.

All these excerpts covered the four sound control effects corresponding to the four categories (Articulation, Intensity, Attack, Cutoff), and for each category they covered the previously identified variations. Note that for the orchestra music excerpts, there were other variations related to the musical piece (style, articulation, rhythm, etc.) and the interpretation (tempo, loudness, etc.), whereas for the Bach piano pieces, the variations were more constrained. In the latter case, indeed, the same phrases were replayed several times with the instructed variations, but only one variation at a time, with the same tempo.

To sum up, we captured 50 gesture sequences from each participant, which corresponds to 50 musical excerpts. For each gesture sequence, each participant repeated the gesture at least 5 times. After pre-processing and manual cut-

ting, we obtained a conducting gesture data set of 1265 gesture units.

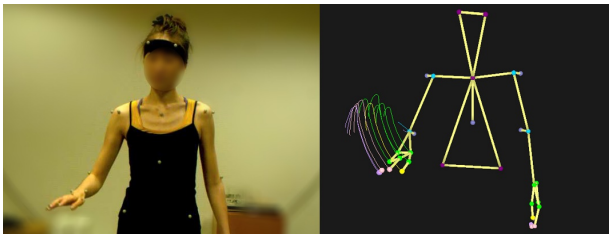


Figure 3: Mocap marker settings and marker-model reconstruction.

5. Analysis

Human movement can be represented by a regularly sampled sequence (also called time series) of T body postures. Each body posture corresponds in turn to a set of m joint positions $\mathbf{x}(t) = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}(t)$, where $\mathbf{x}_k(t)$ (with $1 \leq k \leq m$) is the 3D Cartesian position associated to the k_{th} joint and $1 \leq t \leq T$. Thus, a movement amounts to a matrix $\mathbf{X} = \{\mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(T)\}$ of dimensionality $3 \times m \times T$. The variation in movement category is illustrated

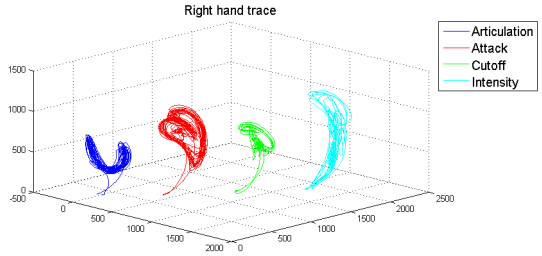


Figure 4: Right hand traces of one subject performing repetitively different gestures among the four categories: Articulation, Intensity, Attack, Cutoff.

in Figure 4. Four different right hand traces corresponding to the four categories (Articulation, Intensity, Attack, Cutoff) are repeated several times by one subject. We can see both the regularity of the traces within each gesture, and the variability of each gesture in shape and duration.

5.1. Feature Extraction and Vectorization

In order to characterize the expressive content of a given movement $\mathbf{X}$, we wished to consider a significant variety of descriptors, both for the recognition of gestures and for the detection of expressiveness, while ensuring that the selected quantities could be computed for several distinct movement categories and variations.

Based on a review of the motion descriptors traditionally used to characterize gestures (Karg et al., 2013), (Kapadia et al., 2013), (Larboulette and Gibet, 2015), we selected, (i) for the hand movement, kinematic features such as position, velocity, acceleration, and curvature, and, (ii) for the hand configuration, geometrical features measuring distances between the wrist and the extremity of the fingers (thumb, index finger, and middle fingers), as well as the volume covered by the hand. The selected hand movement features have proven to be generic and sufficient to cover most variations of affect in movements (Carreno-Medrano et al., 2015). For example, for a right hand Cutoff gesture, illustrated by the trace of its captured data in Figure 5(a), we show the four kinematic features, i.e. the norm of the position, velocity, acceleration, and jerk (see Figure 5(b)). With regard to hand configuration features, they are specifically defined for the hand shapes of Figure 2.

In addition to these low-level descriptors, the Laban Effort descriptors (Maletic, 1987) are frequently used to characterize bodily expressiveness, especially in music-related gestures (Glowinski et al., 2011). Concerning conducting gestures, an experiment was also conducted to show how Laban Effort-Shape descriptors can help young conductors build expressive vocabulary and better recognize the level of expressiveness of different artistic disciplines (Jean, 2004). On the basis of these results, Laban Effort descriptors become a good candidate for characterizing the expressive variations of the conducting gestures. Focusing on the quality of motion in terms of dynamics, energy and expressiveness, these descriptors are defined by four subcategories (Weight, Time, Space, and Flow). The Weight Effort refers to physical properties of the motion; the Time Effort represents the sense of urgency; the Space Effort defines the directness of the movement which is related to the attention to the surroundings; the Flow Effort defines the continuity of the movement. In summary, the chosen descriptors, whether they are kinematic, geometric, or inspired by Laban's Effort components, are computed from the equations defined in (Larboulette and Gibet, 2015). They are listed in Table 2.

Category	Features
Kinematics	Normalized velocity, acceleration, curvature, and jerk norms.
Geometry and space	Displacements between hand joint (wrist) and each finger extremity.
Laban Effort	Flow, Space, Time, and Weight.

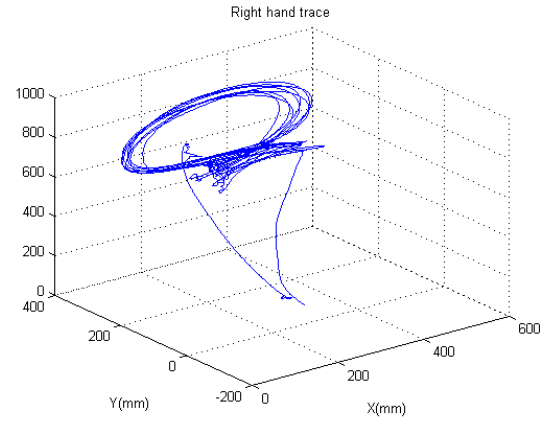
Table 2: List of features used to characterize gesture and motion expressiveness.

5.2. Recognition

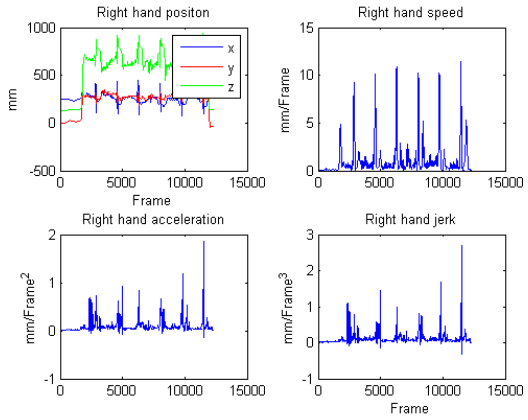
For recognition purposes, we had to cut the gestures in elementary units, each cut gesture corresponding to a specific command and variation that will be used for controlling sound in live performances. The splitting has been achieved through a semi-automatic process, using the velocity information for all the hand traces. Moreover, to manage the variation of duration in our recognition process, we proceeded in two different ways, whether we consider off-line or on-line recognition.

5.2.1. Validation through Off-line Recognition

In the scope of validating our database, both for the gesture categories and the gesture variations within each category, we used an off-line classification approach applied on our gesture units. Two methods were considered to deal with the temporal variation of the gesture units. In the first

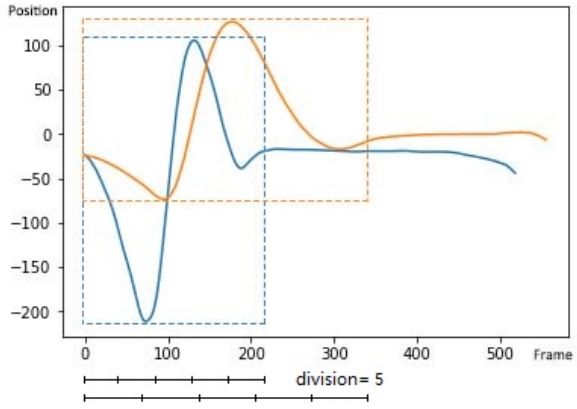


(a) 3D hand trace of the right hand.

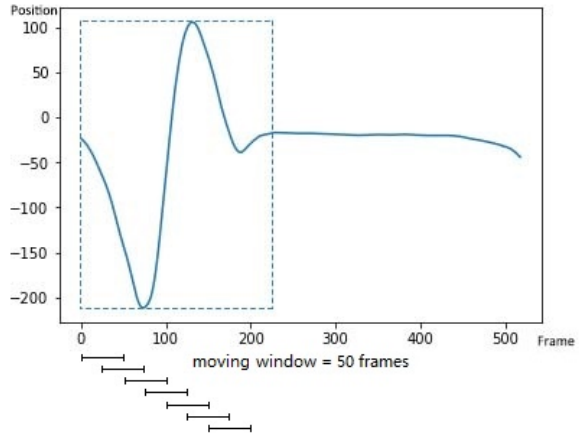


(b) Feature sets.

Figure 5: Kinematic features of a Cutoff sequence.



(a) Equal division of two Cutoff gestures.



(b) Moving window on a Cutoff gesture.

Figure 6: Kinematic features of a Cutoff sequence.

method, we divided each gesture unit into an equivalent number of segments (1, 5, 10 or 20 divisions), and computed a vector of normalized descriptors for each segment composing the gesture, using average and standard deviations of the kinematic and geometric features. In this way, each gesture is represented by a vector of the same size, which facilitates the recognition process. We plan to use in a second method elastic distances associated to Support Vector Machines methods.

5.2.2. On-line Recognition

Secondly, for real-time recognition, we used a moving window and compute the average and standard deviation in this window. Figure 6 shows how the division method (part (a) of the Figure) or a moving window (part (b) of the Figure) works on a Cutoff gesture. These classification methods for off-line and on-line recognition are currently being implemented and tested.

6. Conclusion and Perspectives

In this article we have designed and recorded a novel database of expressive gestures induced by sound effects. The originality of this database lies in the fact that we have separated the functional components of the gestures from their expressive variation. This may lead to interesting applications where it is possible to control sound both from the recognition of the action (gesture category) and from the recognition of the gesture expressiveness. In order to quantitatively validate the database, a recognition system is currently under development, using machine learning algorithms. This ongoing work currently focuses on two aspects: finding the best feature set for on-line classification, and adapting the algorithms for real-time recognition.

In the near future, we also intend to evaluate perceptually the database. The gestures will be evaluated by qualitative information related to Laban Effort components (Energy, Timing, Flow), and by multi-choice questionnaires using semantic terms, while the music excerpts will be evaluated through linguistic questionnaires.

7. Bibliographical References

Bevilacqua, F., Guédy, F., Schnell, N., Fléty, E., and Leroy, N. (2007). Wireless sensor interface and gesture-follower for music pedagogy. In *Proceedings of the 7th international conference on New interfaces for musical expression*, pages 124–129. ACM.

Bevilacqua, F., Schnell, N., Rasamimanana, N., Zamborlin, B., and Guédy, F. (2011). Online gesture analysis and control of audio processing. In *Musical Robots and Interactive Multimodal Systems*, pages 127–142. Springer.

Braem, P. and Bräm, T. (2001). A pilot study of the ex-

pressive gestures used by classical orchestra conductors. *Journal of the Conductor's Guild*, 22(1-2):14–29.

Carreno-Medrano, P., Gibet, S., and Marteau, P.-F. (2015). End-effectors trajectories: An efficient low-dimensional characterization of affective-expressive body motions. In *2015 International Conference on Affective Computing and Intelligent Interaction, ACII 2015, Xi'an, China, September 21-24, 2015*, pages 435–441.

Dilsizian, M., Yanovich, P., Wang, S., Neidle, C., and Metaxas, D. (2014). A new framework for sign language recognition based on 3d handshape identification and linguistic modeling. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Françoise, J., Caramiaux, B., and Bevilacqua, F. (2012). A hierarchical approach for the design of gesture-to-sound mappings. In *9th Sound and Music Computing Conference*, pages 233–240.

Françoise, J., Schnell, N., and Bevilacqua, F. (2013). A multimodal probabilistic model for gesture–based control of sound synthesis. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 705–708. ACM.

Glowinski, D., Dael, N., Camurri, A., Volpe, G., Mortillaro, M., and Scherer, K. (2011). Toward a minimal representation of affective gestures. *IEEE Transactions on Affective Computing*, 2(2):106–118.

Godøy, R. I., Haga, E., and Jensenius, A. R. (2005). Playing "air instruments": mimicry of sound-producing gestures by novices and experts. In *International Gesture Workshop*, pages 256–267. Springer.

Godøy, R. I., Haga, E., and Jensenius, A. R. (2006). Exploring music-related gestures by sound-tracing: A preliminary study. In *Proceedings of the COST287-ConGAS 2nd International Symposium on Gesture Interfaces for Multimedia Systems (GIMS2006)*, pages 27–33.

Jean, N. E. (2004). *The effect of Laban Effort-Shape instruction on young conductors' perception of expressiveness across arts disciplines.* Ph.D. thesis, University of Minnesota.

Kapadia, M., Chiang, I., Thomas, T., Badler, N. I., and Jr., J. T. K. (2013). Efficient motion retrieval in large motion databases. In *Symposium on Interactive 3D Graphics and Games, I3D '13, Orlando, FL, USA, March 22-24, 2013*, pages 19–28.

Karg, M., Samadani, A., Gorbet, R., Kühnlenz, K., Hoey, J., and Kulic, D. (2013). Body movements for affective expression: A survey of automatic recognition and generation. *IEEE Transactions on Affective Computing*, 4(4):341–359.

Larboulette, C. and Gibet, S. (2015). A review of computable expressive descriptors of human motion. In *Proceedings of the 2nd International MoCo Workshop*, pages 21–28. ACM.

Maletic, V. (1987). *Body-space-expression: The development of Rudolf Laban's movement and dance concepts*, volume 75. Walter de Gruyter.

Meier, G. (2009). *The Score, the Orchestra, and the Conductor*. Oxford University Press.

R. Aigner, D., Wigdor, H., Benko, M., Haller, D., Lindbauer, A., Ion, S., Zhao, J., and Koh, T. (2012). Understanding mid-air hand gestures: A study of human preferences in usage of gesture types for hci. Technical report, November.

Neural Caption Generation for News Images

Vishwash Batra, Yulan He, George Vogiatzis
School of Engineering and Applied Science, Aston University
{batrav1, y.he9, g.vogiatzis}@aston.ac.uk

Abstract

Automatic caption generation of images has gained significant interest. It gives rise to a lot of interesting image-related applications. For example, it could help in image/video retrieval and management of vast amount of multimedia data available on the Internet. It could also help in development of tools that can aid visually impaired individuals in accessing multimedia content. In this paper, we particularly focus on news images and propose a methodology for automatically generating captions for news paper articles consisting of a text paragraph and an image. We propose several deep neural network architectures built upon Recurrent Neural Networks. Results on a BBC News dataset show that our proposed approach outperforms a traditional method based on Latent Dirichlet Allocation using both automatic evaluation based on BLEU scores and human evaluation.

Keywords: Recurrent Neural Networks, Image caption generation, Deep learning, Order Embedding.

1. Introduction

There is rich information available on the Internet. Many on-line news sites like CNN, Yahoo, BBC etc. publish images with their stories and even provide photo feeds related to current events. These news sites are a good resource for multimedia files containing information in the form of videos, images and natural language texts. Presence of this vast amount of multimedia data has provided strong impetus to develop machine-learning based applications that jointly model data from different modalities. For example, Ngiam et al. (2011) develops a speech recognition system where they jointly model audio and visual modality. They focus on learning representations of audio data which are coupled with the videos of the lips. Another such application is image caption or visual description generation, which aims to generate text descriptions for an image, often times capturing all the different objects depicted and their spatial relationships.

News image caption generation, however, is different from the typical image captioning task. The input to news image caption generation is both a news article and its accompanying image, as opposed to the traditional image captioning task where the input is only an image. Hence, rather than enumerating objects in a given image and describing their properties or relationships to each other as in the traditional image captioning task, the output of news image caption generation is informative text not only describing the key semantics conveyed in the given image, but also summarising the content of its relating news article (Berg et al., 2004). An example is shown in Figure 1. It can be seen that the captions of news images provide more information than what have been depicted in images only. For example, a reasonable caption for the second image would be "A building". But its actual caption conveys much more information and it is evident that the text content of news articles would also need to be considered when generating good captions for news images.

News caption generation tools can assist journalists in creating descriptions for the images associated with their articles or in finding images that appropriately illustrate their text. It also helps in increased accessibility of web for visually impaired individuals (blind or people with partially impaired vision) users who cannot access the content of many sites in the same way sighted users can (Ferres et al., 2006).



Immigration Service staff have not dealt with reports, agencies say

Ministers are set to admit that they may have significantly under-estimated the number of failed asylum seekers living in Britain, the BBC has learnt. Last year the National Audit Office estimated that the figure could be as much as 283,000 - but at the time the Home Office insisted that was too high. But a trawl of files in the Immigration and Nationality department has produced between 400,000 and 450,000 case files...



A report estimated more than 2,000 children a year are detained

The government is contravening legal guidelines by detaining children whose parents are seeking asylum, a report for a coalition of charities says. The report for the No Place for a Child campaign, co-written by a Labour peer and two opposition MPs, highlights concerns over the issue. Labour's Lord Dubs said there were "workable alternatives" to detention. The Home Office has said detention is used sparingly, especially when children are involved...

Figure 1: BBC News Corpus shows sample news articles containing text, image and caption in the bold.

A wide variety of techniques exist for caption generation ranging from semantic space learning (Karpathy and Fei-Fei, 2017), where both supervised and unsupervised methods exist to learn associations between features extracted from image and words, to latent variable models (Feng and Lapata, 2013). There are models inspired by information retrieval and instantiations of noisy-channel model (Lavrenko et al., 2004). Semantic space learning models learn parameters to map an image to a caption, whereas latent variable models are probabilistic in nature. Recently, there has been a surge of interests in neural caption generation methods due to ground-breaking results produced by deep learning. Mainly, they all have a fundamental architecture in common which is inspired by encoder-decoder architecture from neural machine translation. (You et al., 2016) (Karpathy et al., 2014) (Chen and Zitnick, 2015)

In the encoder-decoder models, caption generation is seen as a translation problem where image is translated to a natural language. Convolutional Neural Networks (CNNs)

are typically used as an image encoder, whereas Recurrent Neural Networks (RNNs) are used for decoding sentences, because of their sequence modeling capability. Although there are other variants proposed, for example, with attention mechanisms included, the encoder-decoder architecture is at the heart of these methodologies (Xu et al., 2015).

Existing work to news image captioning generation is scarce. An early approach tackled the problem with a two-stage process, content selection and surface realization. The first stage consists of an image annotation model, where a given image is tagged with a set of keywords based on topics learned from both news article texts and images using a variant of Latent Dirichlet Allocation (LDA) (Blei, 2004). The second stage uses extractive and abstractive summarisation techniques in forming a sentence from these set of keywords. Word-based models are highly specific in nature and may results in ambiguous results. There is need of sentential integration with the images, as a sentence describes an image without any ambiguity.

In this paper, we propose a sequence-to-sequence deep learning model to address the news image caption generation problem. Specifically, we first encode each sentence of a given news article using an order-embedding vector and extract semantic features from the accompanying image using a pre-trained CNN Network, which are further projected to same semantic space, such that both text and image vectors reside in a common semantic space (Vendrov et al., 2015). We then feed the sentence vectors together with the image vector to a Long Short-Term Memory (LSTM) network (Sak et al., 2014) to generate a vector representation of the image caption. Finally, we use the generated vector to retrieve the most similar sentence from the original news article based on cosine similarity measurement as the caption of the given image. We also explore a number of variants of our proposed architecture and compare them with the previous work on the news image captioning task.

Our experimental results on the BBC News Corpus show that our proposed strategies outperform traditional methods according to automatic evaluation metrics like BLEU scores (Papineni et al., 2002) and are comparable in terms of Meteor Scores (Lavie and Agarwal, 2007). Since automatic evaluation metrics are currently limited by their capability to measure the quality of caption generation models, a human evaluation experiment has also been conducted, where users were shown the news articles from our test dataset.

Our evaluation results show that captions generated by our proposed approach were more favoured than captions generated by an existing model based on LDA. In what follows, we first discuss related work and then describe our proposed methodology, followed by experiments and results, and finally conclude the paper and outline future research directions.

2. Related Work

Our work is related to two lines of research, image captioning and encoder-decoder architecture.

2.1. Image Captioning

The most fundamental problem that connects computer vision and natural language processing is of automatic caption generation of an image. For a long time, there has been significant work in image classification, object detection and image annotation, but a relatively little focus on generating sentential descriptions. So, some of the obvious solutions consist of using the results of these methods, that annotates an image with a set of keywords. These keywords are fed to another stage, that arranges these keywords in the form of a sentence. All of these methods fall under two-stage architecture methods.

Two stages are content selection and surface realization. The former stage, content selection consists of an image annotation model that analyses the content of an image and identifies "what to say" of the image. The latter stage, surface realization consists of a language model, that analyses the keywords and identifies "how to say" of the image.

Image Annotation Methods. Much work within computer vision has focused on image annotation, a task which is very much related but distinct from image description generation. The goal in image annotation is to label an image with keywords relating to its content. Image annotation methods can be classified as *supervised* and *unsupervised*. Supervised image annotation is similar to image classification, as the keywords (or categories) are fixed and pre-defined at training time. The fixed set of categories are identified usually in the form of classes of vocabulary words. Machine learning algorithms are applied to learn a one-to-one correspondence between an image and these categories. The core notion behind is to learn a mapping between visual feature vectors and semantics of the image. A detailed review of supervised methods for image annotation can be found in F Tsai and Hung (2008). Unsupervised image annotation methods do not have a fixed set of pre-defined classes. Instead, algorithms attempt to learn the connections between visual features and words and automatically cluster them into classes of words, which will finally denote the semantics of the image. Typical solutions to this involve introducing latent variables such as LDA models (Pan et al., 2004). Standard latent semantic analysis (LSA) and it's probabilistic variant (PLSA) have been applied to this task (Pan et al., 2004). Barnard et al. (2003) provide a more sophisticated model, they estimate the joint-distribution of words and regional image features while treating annotation as a problem of statistical inference in a graphical model. The final output is clusters of words, which appropriately describe the content of the image.

Surface Realization. The output of the previous stage is a set of keywords that appropriately describe the content of the image. The aim of this stage is to go from keywords to a sentence. Two methods are generally popular for this approach, *extractive methods* and *abstractive methods*. The main idea behind extractive methods is to retrieve most relevant sentences from a document database given the keywords identified in image annotation. Various metrics could be used to calculate the relevance of a sentence with a set of keywords, for example, word-overlap based sentence selection score, vector-space based sentence selection score

or topic-based sentence selection score. Jones (1998) provided a comprehensive overview on sentence-selection algorithms. Although extractive methods help in coming up with grammatically correct sentences and require relatively less linguistic analysis, there are few caveats to consider. Sometimes, there is no single sentence in the document database that best describes the image and is one big limitation for such methods. Abstractive methods try to compose a sentence from the words based on language models. Examples of language models are probabilistic generative models or neural-language based models.

In Farhadi et al. (2010), images are parsed into $< object, action, scene >$ triplets. A more complex graph of detections beyond triplets is used by Kulkarni et al. (2013). State-of-the-art object recognition and language generation techniques are used in their model Babytalk. Feng and Lapata (2013) provide a news article caption generator. They use an LDA-based model for image annotation and use wide variety of surface realization techniques.

All of these two-stage architecture methods have some serious limitations. As mentioned before, a list of keywords is often ambiguous. A set of keywords "blue, sky, car, green" could depict "a blue sky and a green car" or "a blue car and a green sky". Therefore, the models should be designed such that there is strong correlation between phrases and images or sentences and images that are semantically relevant. In other words, a direct leap is taken from image to sentence and vice versa. Moreover, these approaches are heavily hand-designed and rigid when it comes to text generation. So, their applicability becomes limited and they cannot be generalized for new domains (Hofmann, 2001).

In recent years, some deep learning approaches like neural networks are used to co-embed images and sentences in the same vector space also called semantic space (Socher et al.,). Karpathy et al. (2014) co-embed image-crops and subsentences into semantic space. But even such approaches cannot solve the problem of limited applicability. These cannot describe unseen compositions of features even though individual features might have been observed in the training data.

Recently, an encoder-decoder architecture inspired from machine translation has been applied to image captioning and has achieved state-of-the-art performance. In the next subsection, we describe this architecture.

2.2. Encoder-Decoder Architecture

In neural machine translation, an encoder is used to read a sentence in the source language and transforms it into a rich fixed-length embedding vector representation. This embedding vector is in turn fed to a decoder that generates the sentence in the target language. This encoder-decoder architecture has been adopted in image captioning and a class of methods called "neural image captioning" methods have been developed. The idea here is to view captioning as a translation problem, where image is a source language and caption is a target language. Typically a CNN is used as an encoder for the image and RNN as a decoder.

Over the last few years, it has been convincingly shown the CNNs can produce rich representation of the input image by embedding it to a fixed-length vector, such that this representation can be used by a variety of vision tasks (Sermanet et al., 2013). Therefore, it is natural to use CNN as image encoder,

In specific, CNN is first pre-trained for classification. This network is subsequently used as an off-the-shelf feature extractor, where the last hidden layer of the network is used as a feature vector. This hidden representation is fed to the decoder to generate descriptions for the image. Vinyals et al. (2014) proposed a model with a similar architecture. Karpathy and Fei-Fei (2017) developed a deep neural network that infers the latent alignment between segments of sentences and region of image they describe. They use CNN for encoder and a bi-directional RNN over sentences as decoder.

Rather than compressing an entire image into a static representation, attention mechanisms have been introduced which allow salient features to dynamically come to forefront as needed. Using representations from top layer of a CNN that distill information in an image down to the most salient objects is one effective solution. But it has a potential drawback of losing information present in the lower layers which could be useful for generating richer and more descriptive captions. You et al. (2016) propose a soft and hard attention mechanism for image captioning tasks. They use a CNN to encode the images and a RNN with attention mechanism to generate a description. By visualizing attention weights, they switch what the model is looking at while generating a word. You et al. (2016) propose a CNN with an attention mechanism that weights the image features and RNN to generate captions to describe weighted image features.

3. Methodology

Our problem is formulated as follows: given a news image I, and its associated article D, create a sentence description S that best describes the image given D. The training data thus consists of document-image-caption tuples like the ones shown in Figure 1. During testing, we are given a document and an associated image for which we need to generate a caption.

In this section, we propose a novel deep Neural Network (NN) architecture to automatic caption generation of news images. Figure 2 provides a block diagram of the model architecture. We first convert sentences in a news article into a sequence of vectors using a pre-trained order-embedding model (Vendrov et al., 2015). We then encode the accompanying image into an image embedding using the pre-trained Oxford VGGNet (Simonyan and Zisserman, 2014) as an off-the-shelf feature extractor. The VGG Features are further projected to the same order-embedding space. Both sentence and image vectors are represented in a 1,024 dimensional semantic space. The sentence vector sequence is then fed to a LSTM network, which is a specific type of Recurrent Neural Network (RNN). The output of the the network is fed into another LSTM cell which also takes the image vector as an additional input. The final output is considered as a representation which captures the semantics conveyed in both text and image. The cross entropy between the output vector and caption order-embedding vector is used as an objective function to train the LSTM pa-

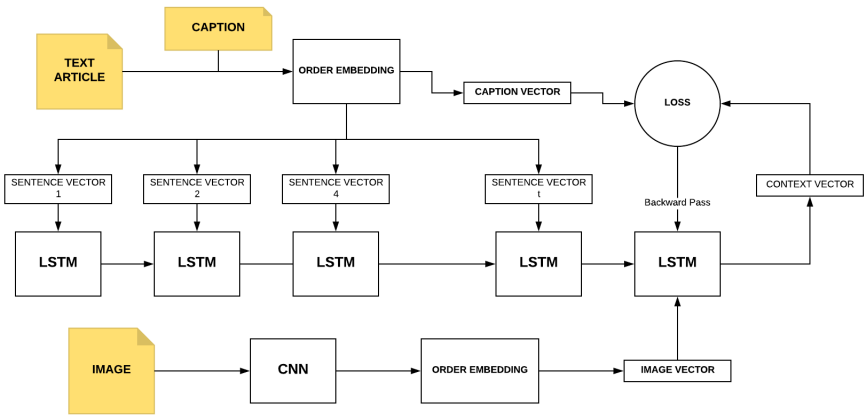


Figure 2: Our proposed deep Neural Network (NN) architecture for news image caption generation.

rameters.

3.1. Text and Image Representation

For encoding sentences, we use a pre-trained order-embedding model (Vendrov et al., 2015) to encode sentences using distributed representations. Order-embeddings exploit the partial order structure of the visual-semantic hierarchy by learning a mapping between sentences and semantic vector space. This projects each sentence into a 1024-dimensional embedding space.

For encoding images, we first use a pre-trained Convolutional Neural Network (CNN), which is an important class of learnable representations applicable, among others, to numerous computer vision problems. Deep CNNs, in particular, are composed of several layers of processing, each involving linear as well as non-linear operators. We use pre-trained Oxford VGGNet as an off-the-shelf feature extractor. The whole network consists of 22 layers. We use the fc7 features, that is the output of the penultimate fully-connected layer, as a representation for the image. The VGG features are projected to same order-embedding space, where sentence vectors reside. As such, both image and sentence vectors reside in a common semantic space which enables direct comparison between them.

3.2. LSTM Training

RNNs surely do a great job at modelling sequences. Unfortunately, the shortcoming of such networks is that they are unable to carry forward information when the length of the chain grows beyond a measure. This is called vanishing gradient effect. To solve this problem, a forgetting mechanism has been proposed in LSTM. LSTMs have many variations. One cell consists of three gates i.e. input, output and forget. Gates typically use sigmoid activation, while input and cell state is often transformed with the hyperbolic tangent function, $\tanh$.

At timestep t, an LSTM has two inputs, x_t the input vector at that timestep and h_{t-1}, the hidden state vector of previous timestep. All the W are weight matrices and b are biases, which are learnable model parameters. In the forward pass, this is how updates are done in the input gate i_t, forget gate f_t, the output gate o_t, the input transform cin_t is taken and the state c_t and h_t is updated in this manner.

$$i_t = g\left(W_{xi}x_t + W_{hi}h_{t-1} + b_i\right)$$

$$f_t = g\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right)$$

$$o_t = g\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right)$$

$$cin_t = \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_{cin}\right)$$

$$c_t = f_t c_{t-1} + i_t cin_t$$

$$h_t = o_t . \tanh\left(c_t\right)$$

In encoder-decoder based models, information is encoded to a context vector which is then fed to the decoder.

At training time, in the forward pass, both sentence vectors and an image vector are fed to a LSTM network to obtain a context vector, as shown in Figure 2. It is assumed that the context vector summarises the information conveyed in both textual and visual formats. We use the cross-entropy between the output of the LSTM network and the order-embedding vector of the image caption as the loss function to backpropagate and update model weights. We set the learning rate to 0.6, momentum to 0.9 and train the model with 30 epochs using stochastic gradient descent.

During testing, given a news article and its accompanied image, we retrieve the most relevant sentence from the article based on the cosine similarity measurement between the output vector from the LSTM and the order-embedding vector of each sentence.

3.3. Variant Architecture

There are multiple ways, in which sequential information can be propagated through an LSTM network. Another variant of the proposed architecture is to feed the image vector at each timestep of the LSTM such that the input to each LSTM cell is a concatenation of a sentence vector and the image vector. Figure 3 shows a variant of our proposed architecture which is called the Deep NN Dual Architecture.

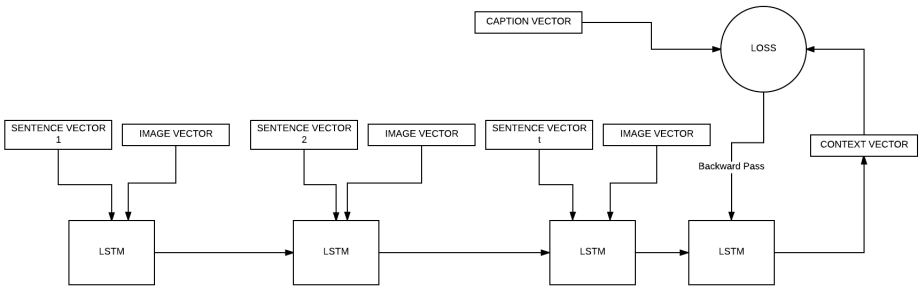


Figure 3: A Deep NN Dual architecture for news image caption generation.

4. Experiments

4.1. Experimental Setup

Dataset. We use the BBC News dataset collected in Feng and Lapata (2013), which contains 3,361 news documents in total. The dataset covers a wide range of topics. Each news article consists of a text article, an image which are normally 200 pixels wide and 150 pixels high, and a caption of the image which has an average length of 20.5 words. On an average each news article contain 421.5 words. The caption vocabulary is 6,180 words and the document vocabulary is 26,795 words. The vocabulary shared between captions and documents is 5,921 words. Some example news articles with their accompanying images and image captures are shown in Figure 1. The original dataset was split into a training set consisting of 3,115 news articles, and a test set consisting of 237 remaining news articles.

Baselines. We compare our proposed model with the following baselines:

- **LDA-based (KL)**. We reproduced the results from Feng and Lapata (2013). For content selection, we first synthesized textual and visual dictionaries where a textual dictionary was created by assigning a unique token id to each word present in any of the articles and visual dictionary was made by clustering SIFT descriptors into 2,000 different visual words. We then trained a LDA model with 1,000 topics on the BBC news dataset containing both text and images. For surface realization, we only used extractive summarisation. It has been shown in Feng and Lapata (2013) that retrieving sentences based on the Kullback-Leibler (KL) divergence between the topic distribution of a sentence and the topic distribution of a news article with its accompanying image gives the best results in terms of human evaluation.

- **Nearest Neighbour**. We also implemented a Nearest Neighbour approach in the order-embedding space. Since both sentences and images are projected to the same semantic space, we can simply choose the sentence which is nearest to a given image as its caption. We use cosine similarity measurement to calculate the similarity score between a sentence vector and an image vector.

- **Deep NN (text input only)**. We explore a variant of our proposed architecture where the input is only text from news articles. This is similar to news headline generation based on text input only except that what we generated here are image captions.

- **Deep NN (dual)**. This is the variant of the architecture shown in Figure 3 where the input to an LSTM cell at each timestep is the concatenation of a sentence vector and the image vector.

Evaluation Metrics. We compare the generated image captions with the actual captions using both BLEU and Meteor scores. The BLEU scores are typically used to evaluate machine translation models. They are calculated based on number of n-gram matches. The Meteor score overcomes the limitation of BLEU by also taking synonyms into consideration. Apart from objective evaluation using BLEU and Meteor, we have also invited human participants to evaluate the generated results by various models. For human evaluation, we have invited 16 human evaluators to choose between the caption generated by the baseline models and our approach for each pair of news article and image presented to them. If human evaluators found none of the captions generated can describe the image well, they can choose the option "none".

4.2. Experimental Results

Method	BLEU	Meteor
LDA-based (KL)	0.3002	**0.0706**
Nearest Neighbour	0.3237	0.0672
Deep NN (text only)	0.3315	0.0642
Deep NN (dual)	0.3303	0.0609
Deep NN	**0.3427**	0.0677

Table 1: News image caption generation results.

The objective evaluation results are shown in Table 2. It can be observed that the simple Nearest Neighbour approach already outperforms the LDA-based method in terms of the BLEU score. Deep NN with text input only improves Nearest Neighbour slightly on BLEU. Deep NN (dual) performs almost the same as Deep NN (dual). This shows that feeding an image vector at each time step somehow diffuse the semantic information captured in images. Our model (deep NN), where the image vector was only fed in the last

timestep in the LSTM network, gives the best overall BLEU score of 0.3427, which outperforms the LDA approach by 4%.

In terms of Meteor scores, both Deep NN and Nearest Neighbour give similar results and they slightly outperform other variants of the deep NN model. Deep NN also performs on par with LDA since the difference of their Meteor scores is only 0.003.

For human evaluation, 38.3 percent of times, the caption generated by our approach was selected as the most appropriate image description by the users, whereas only 28.8 percent of times, the caption generated by the LDA-based model was preferred. We also notice that a staggering 32.91 percent of times, no caption was picked by the users, which could be due to the limited capability of extractive summarisation techniques. Figure 4 shows qualitative study of generated captions.

When only using text content of news articles as the input to our NN architecture, the original model reduces to one-sentence summarisation based purely on text content. As expected, without taking into account the image information, the model has a difficulty in producing appropriate description of a given image. As such, the results are worse than the full approach taking both text and image as input.

Method	Human Evaluation
LDA-based (KL)	28.8%
Deep NN	**38.3%**

Table 2: Human Evaluation results.

5. Error Analysis

In this section, we present more results from the experiments conducted. Figure 4 shows three cases of results. The first case, shows the case, where majority of users picked "Deep NN" caption as a right caption for the given article. In this case, Deep NN methodology is clearly able to identify the subject "Chris Langham" in the picture. It is also able to capture background knowledge of the article. The second case, is where the majority of users picked "LDA" caption as a right caption for the given article. In this case, LDA methodology is able to identify the subject. However, the third case shows, where majority of users picked "No" caption as a suitable description for the given article. This is an example case, where both "LDA" as well as "Deep NN" methodologies have failed to capture the content of the articles. It is quite a challenging case. The gold standard caption is "Parts of Charlie and Chocolate factory were also filmed there.", which is not clearly evident from the image.

38.3% of times, "Deep NN" caption has been picked as a right choice by the users. 32.9% of times, "No" caption has been picked as a suitable choice. 28.8% of times "LDA" caption has been picked as a right choice by majority of users.

6. Conclusion

In this paper, we have proposed a novel deep NN-based architecture for the task of automatic caption generation for news images. The experimental evaluation on the BBC News corpus show that proposed methodology gives a better BLEU score than baseline models and performs similarly compared to the LDA approach on Meteor scores. Nevertheless, we notice that the captions generated by our approach were favoured over the captions generated by the LDA based model most of time by human evaluators. In future, this model can be extended to a full-fledged encoder-decoder architecture, where the context vector from the LSTM cell used in our model can be passed to another LSTM cell, which acts as a decoder for word sequence generation.

7. References

Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D. M., and Jordan, M. I. (2003). Matching words and pictures. *J. Mach. Learn. Res.*, 3:1107–1135, March.

Berg, T. L., Berg, A. C., Edwards, J., Maire, M., White, R., Teh, Y.-W., Learned-Miller, E., and Forsyth, D. A. (2004). Names and faces in the news. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–848–II–854 Vol.2, June.

Blei, D. M. (2004). *Probabilistic Models of Text and Images*. Ph.D. thesis, Berkeley, CA, USA. AAI3183785.

Chen, X. and Zitnick, C. L. (2015). Mind's eye: A recurrent visual representation for image caption generation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2422–2431, June.

F Tsai, C. and Hung, C. (2008). Automatically annotating images with keywords: A review of image annotation systems. 1:55–68, 01.

Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D. (2010). Every picture tells a story: Generating sentences from images. In Kostas Daniilidis, et al., editors, *Computer Vision – ECCV 2010*, pages 15–29, Berlin, Heidelberg. Springer Berlin Heidelberg.

Feng, Y. and Lapata, M. (2013). Automatic caption generation for news images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):797–812, April.

Ferres, L., Parush, A., Roberts, S., and Lindgaard, G. (2006). Helping people with visual impairments gain access to graphical information through natural language: The igraph system. In *Proceedings of the 10th International Conference on Computers Helping People with Special Needs*, ICCHP'06, pages 1122–1130, Berlin, Heidelberg. Springer-Verlag.

Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42(1/2):177–196, January.

Jones, K. S. (1998). Automatic summarising: Factors and directions. In *Advances in Automatic Text Summarization*, pages 1–12. MIT Press.

Karpathy, A. and Fei-Fei, L. (2017). Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):664–676, April.



Figure 4: Error Analysis

Karpathy, A., Joulin, A., and Fei-Fei, L. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 1889–1897, Cambridge, MA, USA. MIT Press.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. (2013). Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2891–2903, Dec.

Lavie, A. and Agarwal, A. (2007). Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lavrenko, V., Manmatha, R., and Jeon, J. (2004). A model for learning the semantics of pictures. In S. Thrun, et al., editors, *Advances in Neural Information Processing Systems 16*, pages 553–560. MIT Press.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 689–696, USA. Omnipress.

Pan, J.-Y., Yang, H., and Faloutsos, C. (2004). Mmss: multi-modal story-oriented video summarization. In *Data Mining, 2004. ICDM '04. Fourth IEEE International Conference on*, pages 491–494, Nov.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sak, H., Senior, A. W., and Beaufays, F. (2014). Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *CoRR*, abs/1402.1128.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (). Recursive deep models for semantic compositionality over a sentiment treebank.

Vendrov, I., Kiros, R., Fidler, S., and Urtasun, R. (2015). Order-embeddings of images and language. *CoRR*, abs/1511.06361.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2014). Show and tell: A neural image caption generator. *CoRR*, abs/1411.4555.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In Francis Bach et al., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul. PMLR.

You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J. (2016). Image captioning with semantic attention. *CoRR*, abs/1603.03925.

MPST: A Corpus of Movie Plot Synopses with Tags

Sudipta Kar, Suraj Maharjan, A. Pastor López-Monroy and Thamar Solorio
Department of Computer Science
University of Houston
Houston, TX 77204-3010
{skar3, smaharjan2, alopezmonroy, tsolorio}@uh.edu

Abstract

Social tagging of movies reveals a wide range of heterogeneous information about movies, like the genre, plot structure, soundtracks, metadata, visual and emotional experiences. Such information can be valuable in building automatic systems to create tags for movies. Automatic tagging systems can help recommendation engines to improve the retrieval of similar movies as well as help viewers to know what to expect from a movie in advance. In this paper, we set out to the task of collecting a corpus of movie plot synopses and tags. We describe a methodology that enabled us to build a fine-grained set of around 70 tags exposing heterogeneous characteristics of movie plots and the multi-label associations of these tags with some 14K movie plot synopses. We investigate how these tags correlate with movies and the flow of emotions throughout different types of movies. Finally, we use this corpus to explore the feasibility of inferring tags from plot synopses. We expect the corpus will be useful in other tasks where analysis of narratives is relevant.

Keywords: Tag generation for movies, Movie plot analysis, Multi-label dataset, Narrative texts

1. Introduction

Folksonomy (Vander Wal, 2005), also known as collaborative tagging or social tagging, is a popular way to gather community feedback about online items in the form of tags. User-generated tags in recommendation systems like IMDb[1] and MovieLens[2] provide different types of summarized attributes of movies. These tags are effective search keywords, are also useful for discovering social interests, and improving recommendation performance (Lambiotte and Ausloos, 2006; Szomszor et al., 2007; Li et al., 2008; Borne, 2013). In this regard, an interesting research question is: *Can we learn to predict tags for a movie from its written plot synopsis?* This question enables an enormous potential to understand the properties of plot synopses that correlate with the tags. For instance, a movie can be tagged with *fantasy, murder,* and *insanity*, that represent different summarized attributes of the movie. The inference of multiple tags by analyzing the written plot synopsis of movies can benefit the recommendation engines. In addition, the consumers would have a useful set of tags representing the plot of a movie. Notwithstanding the usefulness of tags, its proper use in computational methods is challenging as the tag spaces are noisy and redundant (Katakis et al., 2008). Noise and redundancy issues arise because of differences in user perspectives and use of semantically similar tags. For example, the *Movielens 20M dataset* (Harper and Konstan, 2016), which provides tag assignments between ≈27K movies and ≈1,100 unique tags also suffers from these problems. Thus, a fine-grained tagset and their assignment to movie plots can help to overcome these obstacles.

In this work, (i) we present the MPST corpus that contains plot synopses of 14,828 movies and their associations with a set of 71 fine-grained tags; where each movie is tagged with one or more tags. (ii) We discuss the expected properties of this tagset and present the methodology we followed to create such tagset from multiple noisy tag spaces (Section 2.). We also present the process of mapping these tags to a set of movies and collecting the plot synopses for these movies. (iii) We analyze the correlations between the tags and track the flow of emotions throughout the plot synopses to investigate if the associations between tags and movies fit with what we expect in the real world (Section 3.). We also try to estimate the possible difficulty level of a multi-label classification approach to predict tags from the plot synopses. (iv) Finally, we create a benchmark system to predict tags using a set of traditional linguistic features extracted from plot synopses. To the best of our knowledge, this is the first corpus that provides multi-label associations between written plot synopses of movies and a fine-grained tagset. The corpus is freely available to download[3].

	**A Nightmare on Elm Street 3: Dream Warriors** **Tags**: *fantasy, murder, cult, violence, horror, insanity*
	**50 First Dates** **Tags**: *comedy, prank, entertaining, romantic, flashback*

Table 1: Examples of tag assignments to movies from the corpus.

2. Creating the Movie Plot Synopses with Tags (MPST) Corpus

There are several datasets that provide plots or scripts of movies. Since their utilization in this work was difficult, we created a fine-grained tagset first and collected the synopses by ourselves. For example, MM-IMDb (Arevalo et al., 2017) provides plot summaries, posters, and metadata of ≈25K movies collected from IMDb. But these plot summaries are very short to capture different attributes of

[1] http://www.imdb.com
[2] https://www.movielens.org
[3] http://ritual.uh.edu/mpst-2018

movies (average words per summary is 92.5 versus 986.47 in MPST). Another example is ScriptBase (Gorinski and Lapata, 2015), which provides scripts of 1,276 movies collected from IMSDb[4]. But plot synopses are more readily available than the scripts and that helped us to create a bigger dataset. Finally, CMU Movie summary corpus (Bamman et al., 2013) contains ≈42K plot synopses of movies collected from Wikipedia. Due to the absence of IMDb id for these movies, we could not retrieve the tag association information for the movies in that corpus.

We created the corpus using *MovieLens 20M dataset*, *Internet Movie Data Base (IMDb)*, and *Wikipedia*. To create a good corpus, we first defined some expected properties of the corpus (Section 2.1.). Then we created a fine-grained set of tags that satisfies the expected properties (Section 2.2.). We created mappings between the tags and a set of movies and collected the plot synopses for those movies. Figure 1 shows an overview of the data collection process that we will discuss in this section.

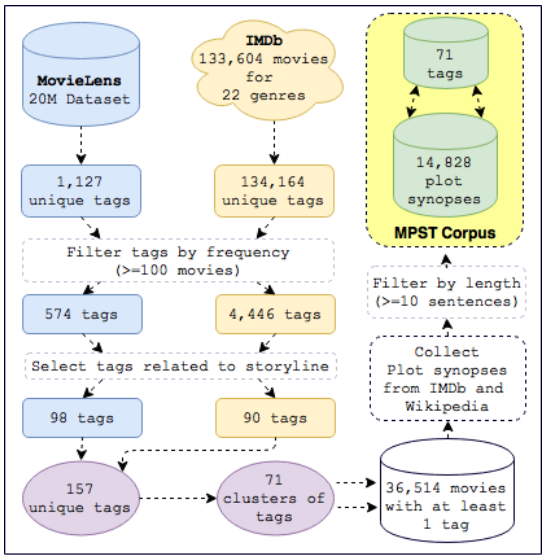


Figure 1: Overview of the data collection process.

2.1. Corpus Requirements

We set the following expected properties for the corpus to make it ideal for future works:

- *Tags should express plot-related attributes that are easy to understand by people.*
  The goal is to predict tags from the written movie plots. Therefore relevant tags are those that capture properties of movie plots (e.g. structure of the plot, genre, emotional experience, storytelling style), and not attributes of the movie foreign to the plot, such as metadata.
- *The tagset should not be redundant.*
  Because we are interested in designing methods to automatically assign tags, having multiple tags that represent the same property is not desirable. For example, tags like *cult*, *cult film*, *cult movie* are closely related and should all be mapped to a single tag.
- *Tags should be well represented.*
  For each tag, there should be a sufficient number of plot synopses, so that the process of characterizing a tag does not become difficult for a machine learning system due to data sparseness.
- *Plot synopses should be free of noise and adequate in content.*
  Plot synopses should be free of noise like IMDb notifications and HTML tags. Each synopsis should have at least 10 sentences, as understanding stories from very short texts would be difficult for any learning system.

2.2. Towards a Fine-Grained Set of Tags

As shown in Figure 1, we collected a large number of tags from *MovieLens 20M dataset* and *IMDb*. To extract the tags commonly used by the users, we only kept the tags that were assigned to at least 100 movies. We manually examined these tags to shortlist the tags that could be relevant to movie plots. We discarded the tags that do not conform to our requirements. At the next step, we manually examined the tags in this shortlist to group semantically similar tags



Figure 2: Tag cloud created by the tags from the dataset. Size of the tags depends on their frequency in the dataset.

together. We got 71 clusters of tags by this process and set a generalized tag label to represent the tags of each cluster. For example, *suspenseful*, *suspense*, and *tense* were grouped into a cluster labeled *suspenseful*. Through this step, we overcame the redundancy issues in the tagset and created a more generalized version of the common tags related to the plot synopses. The tagset is shown as a word cloud in Figure 2.

We created the mapping between the movies and the 71 clusters using the tag assignment information we collected from *MovieLens 20M dataset* and *IMDb*. If a movie was tagged with one or more tags from any cluster, we assigned the respective cluster label to that movie. We used the IMDb IDs to crawl the plot synopses of the movies from IMDb. We collected synopses from Wikipedia for the movies without plot synopses in IMDb or if the synopses in Wikipedia were longer than the synopses in IMDb. These steps resulted in the MPST corpus that contains 14,828 movie plot synopses where each movie has one or more tags.

[4] http://www.imsdb.com

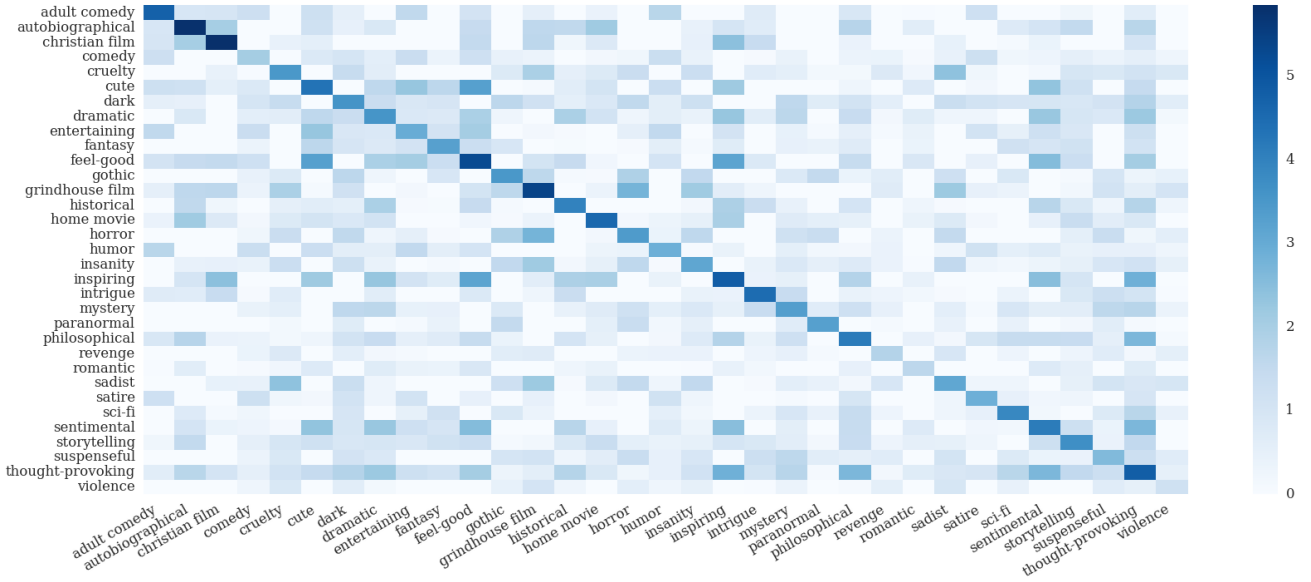


Figure 3: Heatmap of Positive Pointwise Mutual Information (PPMI) between the tags. Dark blue squares represent high PPMI, and white squares represent low PPMI.

Total plot synopses	14,828
Total tags	71
Average tags per movie	2.98
Median value of tags per movie	2
STD of tags for a movie	2.60
Lowest number of tags for a movie	1
Highest number of tags for a movie	25
Average sentences per synopsis	43.59
Median value of sentences per synopsis	32
STD of sentences per synopsis	47.5
Highest number of sentences in a synopsis	1,434
Lowest number of sentences in a synopsis	10
Average words per synopsis	986.47
Median value of words per synopsis	728
STD of words per synopsis	966.16
Highest number of words in a synopsis	13,576
Lowest number of words in a synopsis	72

Table 2: Brief statistics of the MPST corpus.

3. Data Statistics

Table 2 shows that the distribution of the number of tags assigned to movies, number of sentences, and number of words per movie are skewed. Most of the synopses are small in terms of the number of sentences, although the corpus contains some really large synopses with more than 1K sentences. Around half of the synopses have less than 33 sentences. A similar pattern is noticeable for the average number of tags assigned to the movies. Some movies have a large number of tags, but most of the movies are tagged with one or two tags only. *Murder, violence, flashback,* and *romantic* are the most frequent four tags in the corpus that are assigned to 5,732; 4,426; 2,937 and 2,906 movies respectively. Least frequent tags like *non-fiction, christian film, autobiographical, and suicidal* are assigned to less than 55 movies each.

3.1. Multi-label Statistics

Label cardinality (LC) and *label density* (LD) are two statistics that can influence the performance of multi-label learning methods (Tsoumakas and Katakis, 2006; Tsoumakas et al., 2010). Label cardinality is the average number of labels per example in the dataset as defined by Equation 1.

$$LC(D) = \frac{1}{|D|}\sum_{i=1}^{|D|}|Y_i| \quad (1)$$

Here, $|D|$ is the number of examples in dataset D and Y_i is number of labels for the i^{th} example. Label density is the average number of labels per example in the dataset divided by the total number of labels, as defined by Equation 2.

$$LD(D) = \frac{1}{|D|}\sum_{i=1}^{|D|}\frac{|Y_i|}{|L|} \quad (2)$$

Here, $|L|$ is the total number of labels in the dataset. Bernardini et al. (2014) analyzed the effects of cardinality and density on multiple datasets. They showed that, for two datasets with similar cardinalities, learning is harder for the one with lower density. And if the density is similar, learning is harder for the one with higher cardinality. For example, learning performance was better for the *Genbase* dataset (LC: 1.252, LD: 0.046) as compared to the *Medical* dataset (LC: 1.245, LD: 0.028), where they had similar cardinalities but the *Medical* dataset was less dense. On the other hand, performance was better for the *Emotions* dataset (LC: 1.869, LD: 0.311) as compared to the *Yeast* dataset (LC: 4.237, LD: 0.303), where they had similar density but cardinality of the *Yeast* dataset was higher. The label cardinality and label density of our dataset are *2.98* and *0.042*, respectively. Based on the mentioned experiments, we suspect that a traditional multi-label classification approach for this dataset will be a challenge that opens the scope for exploring more scalable approaches.

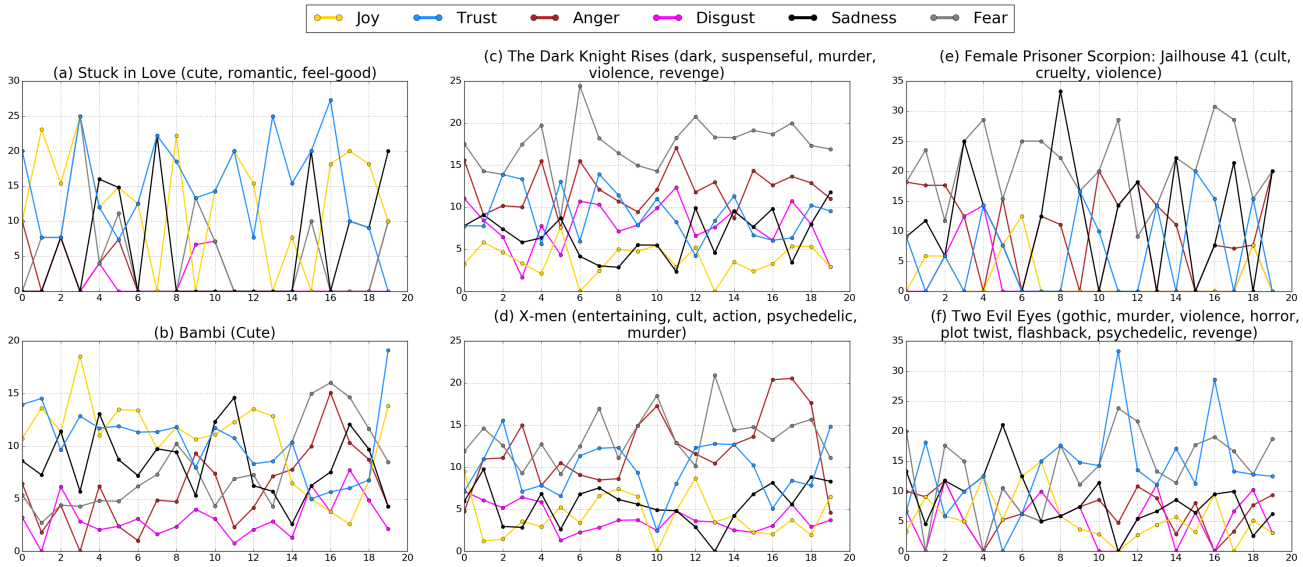


Figure 4: Tracking flow of emotions in the synopses of six movies. Each synopsis was divided into equally sized 20 segments based on the words and percentage of the emotions for each segment were calculated using NRC emotion lexicons. The y axis represents the percentage of emotions in each segment; whereas, the x axis represents the segments.

3.2. Correlation between Tags

To find out significant correlations in the tagset, we compute the Positive Pointwise Mutual Information (PPMI) between the tags, which is a modification over the standard PMI (Church and Hanks, 1990; Dagan et al., 1993; Niwa and Nitta, 1994). PPMI between two tags $t1$ and $t2$ is computed by the following equation:

$$PPMI(t1;t2) \equiv max(log_2 \frac{P(t1,t2)}{P(t1)P(t2)}, 0) \qquad (3)$$

where, $P(t1,t2)$ is the probability of tags $t1$ and $t2$ occurring together and $P(t1)$ and $P(t2)$ are the probabilities of tag $t1$ and $t2$, respectively. Figure 3 shows the heatmap correlation of PPMI values between a subset of tags. The figure shows interesting relations between the tags and supports our understanding of the real world scenario.

High PPMI scores show that *cute, entertaining, dramatic,* and *sentimental* movies can evoke *feel-good* mood, whereas lower PPMI scores between *feel-good* and *sadist, cruelty, insanity,* and *violence* suggest that these movies usually create a different type of impression on people. Also note that, these movies have stronger relations with *horror, cruelty,* and *darkness* which make them difficult to create the *feel-good* experience. It suggest that people tend to get *inspiration* from *dramatic, thought-provoking, historical,* and *home movies*. *Christian films* and *science fictions* are also good sources of *inspiration*. *Grind-house, Christian,* and *non-fiction* films do not usually have *romantic* elements. *Romantic* movies are usually *cute* and *sentimental*. *Autobiographical* movies usually have *storytelling* style and they are *thought-provoking* and *philosophical*. These relations, in fact, show that the movie tags within our corpus seem to portray a reasonable view of movie types based on our understanding of possible impressions from different types of movies.

3.3. Emotion Flow in the Synopses

NRC Emotion Lexicons (Mohammad and Turney, 2010) have been shown effective to capture the flow of emotions in narrative stories (Mohammad, 2011). It is a list of 14,182 words[5] and their binary associations with eight types of elementary emotions from the Hourglass of Emotions model (Cambria et al., 2012) (*anger, anticipation, joy, trust, disgust, sadness, surprise,* and *fear*) with polarity.

In Figure 4, we try to inspect how the flows of emotions look like in different types of plots. The reason behind this investigation is to get a shallow idea about the potential feasibility of the collected plot synopses to predict tags. As general users have written the collected plot synopses and created the tags for movies on the web, there is always a possibility to have noise in the data. For example, in a real world scenario we will expect that horror movies will contain fear and sadness. On the other hand, comedy or funny movies will be filled with happiness.

In the figure we can observe that, emotions like *joy* and *trust* are dominant over *disgust* and *anger* in *cute, feel-good, and romantic* movie's plots (a, b). We can observe sudden spikes in *sadness* in segment 4. The animated movie *Bambi (1942)* shows an interesting flow of different types of emotions. The dominance of *joy* and *trust* suddenly gets low at segment 14 and gets high again at segment 18, where *fear, sadness,* and *anger* get high at segment 14. It is quite self-explanatory that the plot are mixtures of positive and negative emotions where the lead characters go through difficult situations, fight enemies and face a happy ending (spike in *joy* and *trust* at the end) after climax scenes where enemies get defeated. The final segments of (b) indicate happy endings, but the rise of *sadness* and *fear* in (a) indicates that *Stuck in Love (2012)* does not have a happy ending.

We observe the opposite scenarios in cases of *violent, dark, gothic,* and *suspenseful* movies (c, d, e, and f) where

[5] Version 0.92

fear, anger, and *sadness* dominate over *joy* and *trust*. The dominance of *anger* and *fear* is a good indicator of a movie having *action, violence,* and *suspense*. *Female Prisoner Scorpion: Jailhouse 41 (1972)* (e), has dominance of *fear, sadness,* and *anger* throughout the whole movie, and it is easy to guess that this movie has *violence* and *cruelty* portrayed through the lead characters. The flow of *joy, trust, sadness,* and *fear* alters at the middle of the movie *Two Evil Eyes (1990)* (f). Maybe it is the reason why people tagged it with *plot twist*. These observations give evidence of the connection between the flow of emotion in the plot synopses and the experience people can have from the movies, and they also match with what we expected.

4. A Machine Learning Approach for Predicting Tags using Plot Synopses

In this section, we will discuss about some preliminary experiments we conduct with the corpus for predicting tags for movies. We approach the task of predicting tags for movies as a multi-label classification problem and use various traditional linguistic features.

4.1. Hand-crafted Features

Lexical: We extract word n-grams (n=1,2,3), character n-grams (n=3,4) and two skip n-grams (n=2,3) from the plot synopses as they are strong lexical representations. We use term frequency-inverse document frequency (TF-IDF) as the weighting scheme.

Sentiments and Emotions: Sentiments are inherent part of stories and one of the key elements that determine the possible experiences found from a story. For example, *depressive* stories are expected to be full of *sadness, anger, disgust* and negativity, whereas a funny movie is possibly full of *joy* and *surprise*. In this work, we employ two approaches to capture sentiment related features.

- **Bag of Concepts**: As concept-level information have showed effectiveness in sentiment analysis (Cambria, 2013), we extract around 10K unique concepts from the plot synopses using the Sentic Concept parser[6]. It breaks sentences into verb and noun clauses and extracts concepts from them using Parts of Speech (POS) based bigram rules (Rajagopal et al., 2013).

- **Affective Dimensions Scores**: The hourglass of emotions model (Cambria et al., 2012) categorized human emotions into four affective dimensions (*attention, sensitivity, aptitude and pleasantness*) starting from the study on human emotions by Plutchik (2001). Each of these affective dimensions is represented by six different activation levels called 'sentic levels'. These make up to 24 distinct labels called 'elementary emotions' that represent the total emotional state of the human mind. SenticNet 4.0 (Cambria et al., 2016) knowledge base consists of 50,000 commonsense concepts with their semantics, polarity value and scores for the basic four affective dimensions. We used this knowledge base to compute average polarity, attention, sensitivity, aptitude, and pleasantness for the synopses.

We divide the plot synopses into three equal chunks based on words and extracted these two sentiment features for each chunk. We will discuss more about chunk-based sentiment representation later.

Semantic Frames: Semantic role labeling is a useful technique to assign abstract roles to the arguments of predicates or verbs of sentences. We use SEMAFOR[7] frame-semantic parser to parse the frame-semantic structure using the FrameNet (Baker et al., 1998) frames. For each synopsis, we use the bag of frames representation weighted by normalized frequency as feature.

Word Embeddings: Word embeddings have been shown effectiveness in text classification problems by capturing semantic information. Hence, in order to capture the semantic representation of the plots, we average the word vectors of every word in the plot. We use the publicly available FastText pre-trained word embeddings[8].

Agent Verbs and Patient Verbs: Actions done and received by the characters can help to identify attributes of plots. For example, if the characters of a movie *kill, take revenge, shoot, smuggle, chase*; we can expect violence, murder, action from that story. We use the agent and patient verbs found in synopses to capture the actions. In this regard, we use Stanford CoreNLP library to parse the dependencies of the synopses. Then we extract the agent verbs (using *nsubj or agent* dependencies) and the patient verbs (using *dobj, nsubjpass, iobj* dependencies) as described in Bamman et al. (2013). We group these verbs into 500 clusters using the pre-trained word embeddings with the K-means clustering algorithm to reduce noise. We use the distribution of these clusters of the agent verbs and patient verbs over the synopses. We experimented with different values of K (K=100, 500, 1000, 1500), and 500 clusters helped to achieve better results.

4.2. Experimental Setup

Section 3. shows that the distribution of the number tags assigned to per movies is skewed. The average number of tags per movie is approximately three. We thus begin by experimenting with predicting a fixed number of three tags for each movie. Moreover, to get more detailed idea about movies, we create another set of five tags by predicting two additional tags.

We use random stratified split to divide the data into 80:20 train to test ratio[9]. We use the One-versus-Rest approach to predict multiple tags for an instance. We experiment with logistic regression as the base classifier. We run five-fold cross-validation on the training data to evaluate different features and combinations. We tune the regularization parameter (C) using grid search technique over the best feature combination that includes all of the extracted features. We use the best parameter value (*C=0.1*) for training

[6] https://github.com/SenticNet/concept-parser

[7] http://www.cs.cmu.edu/~ark/SEMAFOR

[8] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[9] Train-test partition information is available with the dataset.

a model with all the training data and used that model for predicting tags for the test data.

Majority and Random Baseline: We define majority and random baselines to compare the performance of our proposed model in the task of predicting tags for movies. The majority baseline method assigns the most frequent three or five tags to all the movies. We chose three tags per movie as this is the average number of tags per movie in the dataset. Similarly, the random baseline assigns at random three or five tags to each movie.

Evaluation Metrics: Wu and Zhou (2016) illustrate the complications in evaluating multi-label classifiers by an example of determining the significance of mistakes for the following cases: *one instance with three incorrect labels* vs. *three instances each with one incorrect label*. It is complicated to tell which of these mistakes is more serious. Due to such complications, several evaluation methodologies have been proposed for this type of tasks (Tsoumakas and Katakis, 2006; Wu and Zhou, 2016). For example, *hamming loss, average precision, ranking loss, one-error, coverage,* (Schapire and Singer, 2000; Fürnkranz et al., 2008), *micro* and *macro* averaged versions of *F1* and *AUC* score (Tsoumakas et al., 2010; Tsoumakas et al., 2011; Lipton et al., 2015).

Another complication arises when the label distribution is sparse in a dataset. Less frequent tags could be underrepresented by models, but an ideal model should be able to discriminate among all the possible labels. Such an issue is very common in problems like image annotation, and existing works use *mean per label recall* and *labels with recall*>0 to measure the effectiveness of models in learning individual labels (Lavrenko et al., 2003; Feng et al., 2004; Carneiro et al., 2007; Wang et al., 2009). Here, we use two similar metrics: *tag recall* (TR) and *tags learned* (TL), along with traditional micro-F1 metric. Tag recall computes the average recall per tag and defined by the following equation.

$$TR = \frac{\sum_{i=1}^{|T|} |R_i|}{|T|} \quad (4)$$

Here, $|T|$ is the size of tagset in the corpus, and R_i is recall of i^{th} tag. Tags learned (TL) computes how many unique tags are being predicted by the system for the test data. These evaluation metrics will help us to investigate how well and how many distinct tags are being learned by the models. We evaluate the models using these three metrics in two settings. One is selecting the top three tags and another is selecting the top five tags.

4.3. Results and Analysis

Table 3 shows the performance of the hand-crafted features for predicting tags for movies. All the features beat the baselines in terms of micro-F1 and tag recall (TR). But another significant criterion to evaluate the performances is the number of unique tags predicted by the models, which is measured by the tags learned (TL) metric. We prefer such a model that is capable of creating diverse tagsets by capturing varieties of attributes of movies with reasonable accuracy. For instance, the random baseline used all of the tags in the dataset to assign to the movies but its accuracy is very poor. On the other hand, the majority baseline has better accuracy but it does not have diversity in the tagset. We can see that most of the individual features achieved almost similar micro-F1 scores, but they demonstrate difference in effectiveness to create diversity in predicted tags. Feature combinations seem to improve in TR and TL, but micro-F1 scores are almost similar to the individual features.

	Top 3			Top 5		
	F1	**TR**	**TL**	**F1**	**TR**	**TL**
Baseline: Most Frequent	29.7	4.225	3	31.5	7.042	5
Baseline: Random	4.20	4.328	**71**	5.40	7.281	**71**
Unigram (U)	37.6	7.883	22.6	37.1	11.945	27.4
Bigram (B)	36.5	7.216	19.6	36.1	10.808	24.8
Trigram (T)	31.3	5.204	15.4	32.4	8.461	21
Char 3-gram (C3)	37.0	7.419	22	36.6	11.264	27.4
Char 4-gram (C4)	37.7	7.799	22.6	37.0	11.582	27.2
2 skip 2 gram (2S2)	34.2	6.289	19.4	34.5	9.875	25.2
2 skip 3 gram (2S3)	30.8	4.951	12.8	32.1	8.109	18.2
Bag of Concepts (BoC)	35.7	7.984	29	35.9	12.473	34.8
Concepts Scores (CS)	31.1	4.662	7.8	32.4	7.512	8.2
Word Embeddings	36.8	6.744	13.2	36.1	10.074	17.8
Semantic Frame	33.4	5.551	13.4	33.9	8.394	15.2
Agent Verbs	32.9	5.050	7.2	33.2	7.714	8
Patient Verbs	33.1	5.134	7.4	33.5	7.843	8
U+B+T	37.2	8.732	30	36.8	13.576	36.8
C3+C4	**37.8**	8.662	28.8	**37.4**	13.395	33.6
U+B+T+C3+C4	37.1	9.991	36.8	36.8	15.871	45.8
Al lexical	36.7	10.046	37.6	36.5	15.838	46.4
BoC + CS	35.7	8.165	29.4	36.0	12.754	35.4
All features	36.9	**10.364**	39.6	36.8	**16.271**	47.8

Table 3: Performance of the hand-crafted features using 5-fold cross-validation on the training data. We use three metrics (*F1: micro averaged F1, TR: tag recall, and TL: tags learned*) to evaluate the features.

	Top 3			Top 5		
	F1	**TR**	**TL**	**F1**	**TR**	**TL**
Baseline: Most Frequent	29.7	4.23	3	28.4	14.08	5
Baseline: Random	4.20	4.21	**7**1	6.36	15.04	**71**
System	**37.3**	**10.52**	47	**37.3**	**16.77**	52

Table 4: Results achieved on the test data using the best feature combination (all features) with tuned regularization parameter *C*.

The lexical features show better performance compared to other features. Bag of concepts (BoC) shows similarity in performance. Combination of all lexical features demonstrates effectiveness in capturing a wide range of attributes of movies from the synopses, which is reflected by the better TR and TL scores.

We present the results achieved on the test data in Table 4. Although the result is similar to the result we got with all features during cross-validation, number of predicted unique tags is higher in the test set. This result could be used as a baseline system to compare other methods developed in future as it uses several traditional linguistic features combination to predict tags.

Chunk-based Sentiment Representation: Narratives have patterns in ups and downs of sentiments (Vonnegut, 1981). Reagan et al. (2016) showed that the pattern of changes in sentiments is significant for consumer experiences that results in success of stories. To capture such changes, we experiment with chunk-based sentiments

Chunks	Top 3			Top 5		
	F1	TR	TL	F1	TR	TL
1	35.2	6.550	18.2	35.1	9.928	23.4
2	35.0	7.031	23.0	35.2	10.68	26.8
3	**35.7**	8.165	29.4	**36.0**	**12.754**	35.4
4	35.1	8.153	30.6	35.4	12.723	**36.8**
5	34.8	**8.185**	30.4	35.1	12.553	**36.8**
6	34.3	7.976	**31.2**	34.9	12.725	36.0

Table 5: Experimental results obtained by 5-fold cross-validation using chunk-based sentiment representations. Chunk-based sentiment features were combined with the other features described in Section 4.1.

and emotions representation. We divide the plot synopses into equally sized n chunks based on the word tokens and extract the sentiment and emotion features for each chunk. Then we run 5-fold cross validation on the training data to observe the effect of chunk-based sentiments and emotions representation. We report the results in Table 5. Results show that dividing synopses into multiple chunks and using sentiment and emotion features from each chunk improves the performance of tag prediction. Although we observe noticeable improvements up to three chunks, TL remains similar where micro-F1 scores start to drop when we use more than three chunks. We suspect that higher number of chunks create sparseness in the representation of sentiments and emotions that hurts the performance. So we use sentiments and emotions features using three chunks in further experiments. As the chunk-based representation shows improvement in results, we plan to work capturing the flow of sentiments throughout the plots more efficiently in future work.

5. Conclusion

We have presented a new corpus of ≈70 fine-grained tags and their associations with ≈14K plot synopses of movies. In order to create the tagset, we tackled the challenge of extracting tags related to movie plots from noisy and redundant tag spaces created by user communities in MovieLens and IMDb. In this regard, we describe the methodology for creating the fine-grained tagset and mapping the tags to the plot synopses.

We present an analysis, where we try to find out the correlations between tags. These correlations seem to portray a reasonable set of movie types based on what we expect from certain types of movies in the real world. We also try to analyze the structure of some plots by tracking the flow of emotions throughout the synopses, where we observed that movies with similar tag groups seem to have similarities in the flow of emotions throughout the plots.

Finally, we create a benchmark system to predict tags from the synopses using a set of hand-crafted linguistic features. This dataset will be helpful to analyze and understand the linguistic characteristics of plot synopses of movies, which will in turn help to model certain types of abstractions as tags. For example, what type of events, word choices, character personas, relationships between characters, and plot structure make a movie *mysterious* or *suspenseful* or *paranormal*? Such investigations can help the research community to better exploit high-level information from narrative texts, and also help to build automatic systems to create tags for movies. The generation of tags from movie plots or narrative texts could also be a significant step towards solving the problem of automatic movie profile generation. Methodologies designed using the MPST corpus could also be used to analyze narrative texts from other domains, such as books and storyline of video games.

Acknowledgements

We would like to thank the National Science Foundation for partially funding this work under award 1462141. We are also grateful to Prasha Shrestha, Giovanni Molina, Deepthi Mave, and Gustavo Aguilar for reviewing and providing valuable feedback during the process of creating tag clusters.

6. Bibliographical References

Arevalo, J., Solorio, T., y Gómez, M. M., and González, F. A. (2017). Gated multimodal units for information fusion. In *5th International Conference on Learning Representations (ICLR) 2017 -Workshop Track*.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Bamman, D., O'Connor, B., and Smith, N. A. (2013). Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria, August. Association for Computational Linguistics.

Bernardini, F. C., da Silva, R. B., Rodovalho, R. M., and Meza, E. B. M. (2014). Cardinality and density measures and their influence to multi-label learning methods. *Dens*, 1:1.

Borne, K. (2013). Collaborative annotation for scientific data discovery and reuse. *Bulletin of the American Society for Information Science and Technology*, 39(4):44–45.

Cambria, E., Livingstone, A., and Hussain, A. (2012). The hourglass of emotions. In *Proceedings of the 2011 International Conference on Cognitive Behavioural Systems*, COST'11, pages 144–157, Berlin, Heidelberg. Springer-Verlag.

Cambria, E., Poria, S., Bajpai, R., and Schuller, B. (2016). Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2666–2677, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Cambria, E. (2013). An introduction to concept-level sentiment analysis. In *Mexican International Conference on Artificial Intelligence*, pages 478–483. Springer.

Carneiro, G., Chan, A. B., Moreno, P. J., and Vasconcelos, N. (2007). Supervised learning of semantic classes

for image annotation and retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(3):394–410, March.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, March.

Dagan, I., Marcus, S., and Markovitch, S. (1993). Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics*, ACL '93, pages 164–171, Stroudsburg, PA, USA. Association for Computational Linguistics.

Feng, S. L., Manmatha, R., and Lavrenko, V. (2004). Multiple bernoulli relevance models for image and video annotation. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'04, pages 1002–1009, Washington, DC, USA. IEEE Computer Society.

Fürnkranz, J., Hüllermeier, E., LozaMencía, E., and Brinker, K. (2008). Multilabel classification via calibrated label ranking. *Machine Learning*, 73(2):133–153, Nov.

Gorinski, P. J. and Lapata, M. (2015). Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado, May–June. Association for Computational Linguistics.

Katakis, I., Tsoumakas, G., and Vlahavas, I. (2008). Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD 2008 Discovery Challenge*.

Lambiotte, R. and Ausloos, M., (2006). *Collaborative Tagging as a Tripartite Network*, pages 1114–1117. Springer Berlin Heidelberg, Berlin, Heidelberg.

Lavrenko, V., Manmatha, R., and Jeon, J. (2003). A model for learning the semantics of pictures. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, pages 553–560, Cambridge, MA, USA. MIT Press.

Li, X., Guo, L., and Zhao, Y. E. (2008). Tag-based social interest discovery. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 675–684, New York, NY, USA. ACM.

Lipton, Z. C., Kale, D. C., Elkan, C., and Wetzel, R. C. (2015). Learning to diagnose with LSTM recurrent neural networks. *CoRR*, abs/1511.03677.

Mohammad, S. (2011). From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, LaTeCH '11, pages 105–114, Stroudsburg, PA, USA. Association for Computational Linguistics.

Niwa, Y. and Nitta, Y. (1994). Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*, COLING '94, pages 304–309, Stroudsburg, PA, USA. Association for Computational Linguistics.

Plutchik, R. (2001). The nature of emotions human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350.

Rajagopal, D., Cambria, E., Olsher, D., and Kwok, K. (2013). A graph-based approach to commonsense concept extraction and semantic similarity detection. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 565–570. ACM.

Reagan, A. J., Mitchell, L., Kiley, D., Danforth, C. M., and Dodds, P. S. (2016). The emotional arcs of stories are dominated by six basic shapes. *CoRR*, abs/1606.07772.

Schapire, R. E. and Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine learning*, 39(2-3):135–168.

Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., and Servedio, V. D. (2007). Folksonomies, the semantic web, and movie recommendation.

Tsoumakas, G. and Katakis, I. (2006). Multi-label classification: An overview. *Dept. of Informatics, Aristotle University of Thessaloniki, Greece*.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). Mining multi-label data. In *In Data Mining and Knowledge Discovery Handbook*, pages 667–685.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2011). Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089.

Vander Wal, T. (2005). Folksonomy definition and wikipedia. *vanderwal. net*.

Vonnegut, K. (1981). Palm sunday: An autobiographical collage.

Wang, C., Yan, S., Zhang, L., and Zhang, H.-J. (2009). Multi-label sparse coding for automatic image annotation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1643–1650, June.

Wu, X. and Zhou, Z. (2016). A unified view of multi-label performance measures. *CoRR*, abs/1609.00288.

7. Language Resource References

Harper, F Maxwell and Konstan, Joseph A. (2016). *The movielens datasets: History and context*. ACM.

Mohammad, Saif and Turney, Peter D. (2010). *Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon*. Association for Computational Linguistics.

OpenSubtitles2018: Statistical Rescoring of Sentence Alignments in Large, Noisy Parallel Corpora

Pierre Lison[*], **Jörg Tiedemann**[†], **Milen Kouylekov**[‡]
[*]Norwegian Computing Center
plison@nr.no

[†] Department of Modern Languages, University of Helsinki
jorg.tiedemann@helsinki.fi

[‡] Department of Informatics, University of Oslo
milen.kouylekov@usit.uio.no

Abstract

Movie and TV subtitles are a highly valuable resource for the compilation of parallel corpora thanks to their availability in large numbers and across many languages. However, the quality of the resulting sentence alignments is often lower than for other parallel corpora. This paper presents a new major release of the OpenSubtitles collection of parallel corpora, which is extracted from a total of 3.7 million subtitles spread over 60 languages. In addition to a substantial increase in the corpus size (about 30 % compared to the previous version), this new release associates explicit *quality scores* to each sentence alignment. These scores are determined by a feedforward neural network based on simple language-independent features and estimated on a sample of aligned sentence pairs. Evaluation results show that the model is able predict lexical translation probabilities with a root mean square error of 0.07 (coefficient of determination R^2 = 0.47). Based on the scores produced by this regression model, the parallel corpora can be filtered to prune out low-quality alignments.

Keywords: Parallel Corpora, Machine Translation, Bitext alignment

1 Introduction

Movie and TV subtitles are used in a wide range of language technology applications. Their availability in a large number of languages makes them well-suited for the creation of parallel multilingual corpora. These corpora are a central resource for learning machine translation models (Koehn, 2009) but can also be used for corpus-driven lexicography, cross-lingual NLP or translation research (Paetzold, 2016; Akbik et al., 2016; Mikhailov and Cooper, 2016). Recent work on neural conversation models also showed that subtitles can be used to train dialogue agents (Vinyals and Le, 2015; Lison and Bibauw, 2017).

Parallel corpora derived from subtitles have a number of benefits. The first one is their size: the OpenSubtitles dataset is (to the best of our knowledge) the world's largest open collection of parallel corpora. The latest release, which is presented in this paper, contains no less than 3.4 billion sentences (amounting to 22.2 billion tokens) spread over 60 languages and a total of 1782 language pairs. As subtitles are annotated with timestamps, they can also be efficiently synchronised using a linear-time algorithm (Tiedemann, 2008). Finally, their conversational nature make them ideal for exploring dialogue phenomena and properties of everyday language (Paetzold and Specia, 2016; van der Wees et al., 2016).

However, the extraction of parallel corpora from subtitles must also face some challenges. One difficulty stems from the fact that subtitles are typicall not direct translations of one another. Rather, they should better be viewed as boiled-down *transcriptions* of the same conversations across several languages. Subtitles will inevitably differ in how they "compress" the conversations, notably due to structural divergences between languages, cultural differences and disparities in subtitling traditions/conventions. As a consequence, sentence alignments extracted from subtitles often have a higher degree of insertions and deletions compared to alignments derived from other sources.

We present in this paper a new release of the OpenSubtitles collection. In addition to increasing the global volume of the dataset (+30 % of the total number of sentences), the release includes several technical improvements in the preprocessing and alignments of the sentences. The most important improvement is the calculation of explicit *quality scores* for all sentence pairs. As explained in Section 3, these quality scores are determined by a neural model based on simple features extracted from the sentence pairs. This regression model is fitted based on a sample of sentence pairs and can be subsequently applied to the full collection of bilingual corpora.

The paper is organised as follows. Section 2 describes the preprocessing and alignment steps involved in compiling the parallel corpora, while Section 3 presents the alignment (re)scoring model. Section 5 provides our conclusions.

2 Dataset

2.1 Source Data

The raw data consists of a full database dump of the OpenSubtitles website[1], encompassing a total of 3.98 million subtitle files. In addition to the files themselves, the database dump contains information about the source material (through IMDB identifiers[2]), the subtitling language and format (usually `.srt` format), as well as miscellaneous meta-data such as the upload date and user ratings. The dataset covers a total of 208 K movies or TV episodes (as determined by their IMDb identifier). 69 % of the IMDb

[1] http://www.opensubtitles.org

[2] http://www.imdb.com

Language	2016 release Subtitle files	2018 release Subtitle files	Covered IMDbs	Sentences	Tokens
Afrikaans	32	63	57	61.3K	450K
Albanian	3.0K	3.1K	2.0K	3.6M	24.4M
Arabic	67.3K	94.1K	45.0K	83.6M	458M
Armenian	1	9	9	4.1K	33.0K
Basque	188	0.9K	0.9K	1.0M	5.8M
Bengali	76	0.5K	440	0.7M	3.7M
Bosnian	30.5K	37.3K	21.1K	34.1M	216M
Breton	32	32	28	23.1K	165K
Bulgarian	90.4K	108K	59.1K	94.6M	0.6G
Catalan	0.7K	0.8K	0.8K	0.6M	4.7M
Chinese (simplified)	22.4K	29.1K	15.2K	31.2M	191M
Chinese (traditional)	6.7K	9.9K	6.0K	10.7M	66.2M
Bilingual Chinese-English	4.5K	8.6K	4.7K	9.2M/8.3M	56.0M/73.9M
Croatian	96.8K	126K	52.9K	113M	0.7G
Czech	125K	157K	63.8K	136M	0.9G
Danish	24.1K	32.4K	19.5K	30.2M	208M
Dutch	98.2K	125K	58.9K	105M	0.8G
English	322K	447K	140K	441M	3.2G
Esperanto	89	103	95	93.1K	0.6M
Estonian	23.5K	28.8K	16.2K	27.5M	168M
Finnish	44.6K	64.4K	45.2K	52.0M	282M
French	105K	127K	66.7K	107M	0.8G
Galician	370	449	424	309K	2.4M
Georgian	271	293	268	281K	1.7M
German	27.7K	46.5K	34.5K	41.6M	288M
Greek	114K	143K	61.1K	126M	0.9G
Hebrew	79.7K	98.7K	43.5K	87.5M	0.5G
Hindi	57	102	92	144K	1.0M
Hungarian	99.3K	131K	66.7K	104M	0.6G
Icelandic	1.3K	1.5K	1.3K	1.9M	12.2M
Indonesian	11.0K	21.6K	12.2K	22.8M	138M
Italian	96.5K	135K	55.8K	105M	0.8G
Japanese	2.6K	3.5K	3.0K	3.2M	23.7M
Kazakh	0	4	4	4.1K	19.8K
Korean	0.7K	2.2K	1.9K	2.3M	10.2M
Latvian	392	493	459	0.6M	3.5M
Lithuanian	1.5K	2.0K	1.8K	2.1M	11.6M
Macedonian	5.6K	7.9K	4.6K	7.9M	50.3M
Malay	1.0K	3.2K	2.2K	3.8M	22.9M
Malayalam	251	421	379	0.5M	2.8M
Norwegian	8.9K	14.2K	11.8K	13.0M	86.8M
Persian	6.5K	12.2K	8.0K	13.0M	78.8M
Polish	161K	279K	66.5K	237M	1.4G
Portuguese	96.3K	131K	48.1K	118M	0.8G
Portuguese (BR)	220K	289K	101K	252M	1.7G
Romanian	162K	205K	72.3K	193M	1.3G
Russian	38.7K	56.0K	39.9K	44.9M	291M
Serbian	148K	180K	67.8K	168M	1.1G
Sinhalese	0.5K	0.9K	0.8K	1.0M	5.7M
Slovak	14.7K	18.1K	12.4K	16.1M	104M
Slovenian	52.6K	60.4K	27.1K	59.6M	361M
Spanish	192K	234K	91.7K	214M	1.5G
Swedish	27.3K	41.1K	26.2K	36.2M	245M
Tagalog	52	60	59	19.3K	130K
Tamil	17	32	30	40.2K	206K
Telugu	20	22	22	30.4K	160K
Thai	10.2K	11.0K	5.8K	9.1M	18.8M
Turkish	159K	189K	65.0K	173M	1.0G
Ukrainian	1.0K	1.6K	1.4K	1.3M	7.9M
Urdu	14	35	32	46.5K	358K
Vietnamese	3.1K	5.2K	4.1K	5.1M	41.9M
Total	2.8M	3.7M		3.4G	22.2G

Table 1: Statistics for the 60 languages in the extracted corpus. The *subtitles files* corresponds to the number of converted subtitles (which may be lower than the number of raw subtitles in the database due to discarded files). The *covered IMDbs* represent the number of distinct movies or TV episodes (denoted by their IMDb identifier) covered by the subtitles.

identifiers are associated with subtitles in at least two languages and 29 % with at least 10 languages.

2.2 Preprocessing

A number of steps are required to preprocess the subtitle files, as detailed in (Lison and Tiedemann, 2016):

1. **Format conversion**: The `.srt` subtitles are parsed to extract their constitutive blocks. This step includes detecting the file encoding.

2. **Sentence segmentation**: There no one-to-one correspondence between sentences and subtitle blocks displayed on the screen, as illustrated in this small example (where the first sentence is spread over 2 subtitle blocks, while the third block contains 2 sentences):

```
140
00:07:12,502 --> 00:07:15,812
Quando abbiamo estratto l'energia
blu positiva dal frammento

141
00:07:15,902 --> 00:07:19,019
ci siamo ritrovati con questo
sottoprodotto altamente instabile.

142
00:07:19,102 --> 00:07:21,935
- l'energia rossa negativa.
- Ah, quella mi piace.
```

   The sentences are segmented using language-specific heuristics based on punctuation markers, time gaps between blocks, and capitalisation.

3. **Tokenisation**: Once the sentences are segmented, they are tokenised, using either the tokenisation scripts from Moses (Koehn et al., 2007) or the Kytea toolkit for Chinese word segmentation (Neubig et al., 2011).

4. **Correction of OCR errors**: Some subtitles are extracted from video streams using OCR (Optical Character Recognition), generating a number of recognition errors. A noisy channel approach is presented in (Lison and Tiedemann, 2016) to correct these errors based on language models derived from the Google N-grams. This spellchecking model is also used here with some minor improvements to better handle e.g. accented characters and proper names. In total, more than 9 million tokens were corrected using this approach (with 4 million tokens just for English).

5. **Inclusion of meta-data**: Finally, the subtitles are enriched with meta-data extracted from IMDb, providing details such as the film genre and the original (spoken) language of the movie or TV episode. The new release contains additional information such as the version number of the subtitles and flags indicating (a) whether the subtitle is intended for hearing-impaired audiences and (b) whether the subtitles were generated automatically using machine translation.

After preprocessing, we obtain a total of 3.7 million subtitles (180 thousand subtitles were discarded due to formatting errors or erroneous meta-data). Each subtitle is encoded in a separate XML file including the tokenised sentences (annotated with timestamps) together with meta-data about the subtitle and its associated movie / TV episode.

2.3 Alignment

Sentence alignment is done using the time-overlap algorithm proposed by (Tiedemann, 2008). The procedure searches for the alignment that maximises the time-overlap between aligned units based on the time stamps given in the subtitles. Time information is extrapolated in correlation to string length in cases where it is not available at the sentence boundary. To further improve the synchronisation, we use lexical cues to estimate offset and speed parameters using bilingual dictionaries extracted from word-aligned subtitles (Tiedemann, 2008). In contrast to our earlier releases, we now also keep alignments between alternative subtitle files besides the ones that show the best match according to an overlap measure. Those alternative links are stored in separate alignment files and may be used to complement the selection of top-ranked subtitle pairs. Furthermore, intralingual links will also be offered again based on the procedures of (Tiedemann, 2016).

3 Rescoring model

As mentioned in the introduction, sentence alignments extracted from subtitle are often less literal than alignments from other types of bilingual corpora. Subtitle must indeed obey strong space and time constraints: a maximum of two lines with at most 40-50 characters per line and an on-screen display between 1 and 6 seconds (Aziz et al., 2012). Subtitles must therefore be crisp and boil down the spoken conversations to a small number of words. The everyday language used in subtitles also leaves more room for translation choices than technical or legal texts. Here are two examples of non-literal alignments:

English:	Oh, I bet it does
French:	Le contraire m'aurait surpris.
	[The contrary would have surprised me]
Arabic:	أنت الأهدأ في مركز العاصفةً
	[You are the calmest in the center of the storm]
Spanish:	Dijeron que no tenías nervios.
	[They said you had no nerves.]

These less literal alignments (along with other types of misalignments due to e.g. timing differenes) may lead to problems for downstream NLP tasks. Fortunately, some surface cues that can be exploited to predict whether a sentence pair is closely aligned or not. For instance, a large difference in the number of tokens in the source and target language may be indicative of a low-quality alignment. On the other side, the presence of cognates or the use of identical punctuation markers increases the likelihood of a good alignment.

3.1 Measures of alignment quality

The first step towards building the rescoring model is to determine a measure of alignment quality that can be used

as target variable. The approach used in this paper relies on extracting a sample of sentence pairs from the bilingual corpora, computing their lexical translation probabilities (in both directions) based on existing lexical translation tables and using these probabilities as a proxy for the alignment quality[3]. More specifically, we rely on the expectation formula of IBM Model 1 (Brown et al., 1993) to compute the log-probabilities of the target sentence t given the source s and the source s given the target t:

$$\log P(s|t) = \alpha \sum_{j=1}^{l_s} \log \left(\sum_{i=0}^{l_t} t(s_j|t_i) \right) \quad (1)$$

$$\log P(t|s) = \alpha \sum_{j=1}^{l_t} \log \left(\sum_{i=0}^{l_s} t(t_j|s_i) \right) \quad (2)$$

In the two formulae above, α represents a normalising factor and $t(x|y)$ the translation probability of token x from token y given by the lexical translation table. As done in IBM Model 1, s_0 and t_0 represent a default "null" value allowing for tokens to appear in the target without direct equivalent in the source language. To obtain the lexical translation tables, word alignments are first generated by running GIZA (Och and Ney, 2000) on the existing bitexts from OpenSubtitles and estimating the probability of each word pair through maximum likelihood.

In order to be useful as measures of alignment quality, the log-probabilities in (1) and (2) must, however, be slightly modified. First, the log-probabilities will typically be lower for a long source sentence than for a short one, since the number of translation choices increases with the sentence length. This is unfortunate, as we do not want to penalise long sentence pairs in the scoring model. To address this issue, the log-probabilities in (1) and (2) are divided by the length of the source sentence, such that the average log-probability remains roughly constant as a function of the sentence length. We can then define a raw score of alignment quality between the two sentences s and t as the minimum of these two rescaled log-probabilities:

$$\text{score}_{\text{raw}}(s,t) = \min \left(\frac{\log P(t|s)}{l_s}, \frac{\log P(s|t)}{l_t} \right) \quad (3)$$

Furthermore, the lexical translation probabilities may also vary according to the language pair. This variation may be due to the size of the bitext on which the translation tables were trained (a larger bitext will lead to a higher number of alternative translations for each token), or to the linguistic distance between the source and target language. To avoid penalising language pairs with lower average translation probabilities, the raw scores of (3) are rescaled separately for each language pair using quantile normalisation (Bolstad et al., 2003). Quantile normalisation is a non-linear transformation that matches the quantiles of the original distribution to the quantiles of a target distribution, in this case a normal distribution. After this quantile transform, the scores are mapped to a range of $[0, 1]$. The final scores are thus computed as:

$$\text{score}_{\text{final}}(s,t) = \text{scale}_{L_s,L_t}\left(\text{score}_{\text{raw}}(s,t)\right) \quad (4)$$

where scale_{L_s,L_t} is the quantile transform of the raw scores for the source and target languages L_s and L_t followed by rescaling to [0,1]. A score of 0.5 will therefore indicate a sentence pair whose log-probabilities (per token) revolve around the arithmetic mean for that particular language pair.

3.2 Features

Three families of features are extracted from the sentence alignments:

- Features extracted at the level of sentence pairs, such as the ratio of sentence length (measured in number of tokens or characters) in the source and target, or the use of similar punctuations.
- Features extracted at the subtitle level, such as the number of empty alignments or the ratio between the total number of tokens in the two subtitles.
- Features extracted from meta-data, in particular the languages used, movie or TV genre, release year and user rating.

These three families of features are enumerated in Table 2. All features are rescaled separately for each language pair. This per-language rescaling is necessary since the distribution of many features will correlate with the language pair – for instance, the number of cognates will be higher for Spanish-Catalan than for Arabic-Chinese. It is worth noting that the features defined in Table 2 are all based on simple, surface-level measures that do not rely on external language resources or NLP tools. This is important as many of the languages found in the OpenSubtitles corpus have relatively few available linguistic resources.

4 Evaluation

4.1 Model selection

A set of 8.3 million sentence pairs was extracted from the OpenSubtitles corpus, covering 760 distinct language pairs. This set corresponds to 0.24 % of the total number of sentences in the corpus. The features and quality scores were extracted for these sentence pairs based on the approach described in the previous section. Several machine learning models for regression were tested:

- Ridge regression, which is a simple linear model with L_2 regularisation.
- Lasso regression, another linear model with L_1 regularisation.
- Gradient boosting, which builds a predictor from an ensemble of simpler models, here regression trees.

[3] A previous version of this paper used position-independent word error rates computed from Google translations as response variable. However, the average error rates were too high to be practically useful to estimate the quality of sentence pairs.

Feature	Description
Features extracted from the sentence pair:	
tokens_{ratio,ndiff}	Ratio and normalised difference between the number of tokens in source and target sentences.
chars_{ratio,ndiff}	Ratio and normalised difference between the nb. of characters in the two sentences.
nb_identical_{all,cap}	Nb. of identical tokens in the two sentences, considering all tokens or only capitalized tokens.
nb_cognates	Nb. of near-identical tokens (shared substring of at least 4 characters) in the two sentences.
nb_corrected	Nb. of tokens corrected by the spellchecker(s) in the two sentences.
same_timings	Whether the start and end timestamps of the sentences in the two subtitles are identical or not.
time_overlap	Time overlap between the source and target sentences (as given by the timestaps).
nb_aligns	Total nb. of sentences in the pair (2 for 1:1 alignments, 3 for 1:2, etc.)
final_punct	Whether the final punctuation is the same in the source and target sentence or not.
llcsr	Length of the longest common substring between the two sentences, normalised by length.
nonalpha_seq	Length of longest common subsequence for non-letter characters (punctuation and numbers).
Features extracted at subtitle-level:	
sentences_{ratio,ndiff}	Ratio and normalised difference between the total number of sentences in the two subtitles.
tokens_{ratio, ndiff}	Ratio and normalised difference between the total number of tokens.
duration_{ratio, ndiff}	Ratio and normalised difference between the duration (in seconds)
corrected_words	Total number of tokens corrected by the spellcheckers (when available).
unknown_words	Total number of tokens unknown to the spellcheckers (when available).
ratio_{0:1,1:1,1:2,1:3}	Ratio of alignments of a particular type among the full list of alignments.
Meta-data features:	
language	Source and target languages (one-hot encoding)
genre	Movie or TV genre (one-hot encoding).
year	Release year of movie or TV episode.
original	Original language used in the movie or TV episode (one-hot encoding)
MT-translated	Whether zero, one or both subtitles are marked as translated by MT engines
confidence	Confidence score from language identification tool on the subtitles
rating_{min,max,avg}	Minimum, maximum and average rating of the two subtitles.

Table 2: Features used in the scoring model.

- Feedforward neural networks (multilayer perceptron) including either one or two hidden layers.

The performance of these models are evaluated through 10-fold cross validation, using the mean-square error, root-mean-square error and coefficient of determination (R^2) as evaluation metrics. The baseline is simply the prediction of the mean value for the quality score.

Table 3 summarises the results. We can observe that the best performing model is a feedforward neural network with two hidden layers of 100 units each. The neural network obtains a R^2 for coefficient of determination for 0.47, which means that 47 % of the variance in the quality score can be predicted from the input features using this model. The good performance of neural networks seems to indicate the presence of complex, non-linear relations between the features and the quality score which cannot be accounted for by simpler models. The bad performance of Lasso regression (which favours using a small number of features due to the L_1 regularisation) also shows that there is no single feature that works as a good predictor for the task.

Once learned on the dataset of 8.3 M sentence pairs, the

Model	MSE	RMSE	R^2
Baseline (predict mean)	0.009	0.096	0.0
Lasso regression ($\alpha = 0.01$)	0.008	0.092	0.091
Lasso regression ($\alpha = 0.001$)	0.006	0.081	0.303
Ridge regression ($\alpha = 1$)	0.006	0.077	0.356
Gradient boosting (10 regression trees)	0.007	0.085	0.224
Feedforward NN (one hidden layer, dim=100)	0.005	0.071	0.457
Feedforward NN (two hidden layers, dim=100)	**0.005**	**0.070**	**0.470**

Table 3: Evaluation results for various machine learning models on the task of predicting the value of $\text{score}_{\text{final}}(s,t)$ from the features listed in Table 2. MSE stands for mean-square error, RMSE for the root mean-square error and R^2 for coefficient of determination.

neural network can then be straightforwardly applied on the full list of aligned sentences in OpenSubtitles in order to assign each alignment to a quality score.

We also conducted a small-scale manual analysis of the quality scores predicted by the neural network. Here are a few examples of alignments assigned to a low quality score (<0.25) by the neural network:

Afrikaans:	Kalmeer
	[Calm down]
Polish:	Dlatego byłem w Wiedniu.
	[That's why I was in Vienna]
Bosnian:	Tačno tako
	[Exactly]
Danish:	Og du er tidligere straffet?
	[And you had previous convictions?]
Greek:	Θεέ μου
	[Oh my god]
Portuguese:	Residência Mainwaring.
	[Mainwaring Residence.]
German	(Mystische Musik)
	[(Mystical music)]
Turkish	Lordum...
	[My Lord...]

4.2 MT Experiments

In this section we look at machine translation models trained on filtered data sets in order to test the impact of rescoring on a downstream task. For this purpose, we apply an attentional sequence-to-sequence model implemented in the Helsinki neural MT system (Östling et al., 2017) with byte-pair encoding (BPE) to the language pairs of news translation task at WMT 2017. Our models are trained on OpenSubtitles data only from the 2018 release presented in this paper and we leave subtitles released in 2017 as held-out data from which we extract 10,000 sentences and their alignments as an in-domain test set for each language pair. We restrict the test data to one-to-one sentence pairs with a time-overlap of over 80% to reduce noise in the data.

Using this setup we can now compare two different systems for each language pair: One that is trained on all data (excluding the heldout data) and one that is trained on filtered data using a rescoring threshold of 0.6. We then apply both models to in-domain test data from the subtitle corpus and to out-of-domain news data from WMT 2017.

All systems apply the same setup with the same number of training batches without any language-specific tuning of any of them. In particular, we use 256 dimensions for word embeddings, 512 dimensions for the hidden LSTM-layer in the encoder, 1024 dimensions for the decoder LSTM's and 256 dimensions for the attention layer. We use savepoint ensembling of the last five models (each of them saved after 5,000 batches of size 16) and stop training after 45,000 batches. We use a vocabulary of 50,000 items for each language and split the data using BPE (Sennrich et al., 2016) trained on the subtitle trainng data with 50,000 merge operations. We also apply the hybrid encoder of HNMT to avoid unknown words in translation.

Note that the results on news data will be much lower than official results from WMT 2017 due to the domain mismatch and limited training that we apply in our experiments. We only give BLEU scores here to provide some indication of the translation quality that can be expected. Further analyses of the MT results is outside of the scope of this paper. Table 4 summarises the results of our experiments.

	2016		2018		filtered	
system	subs	news	subs	news	subs	news
en-cs	28.36	12.02	**28.76**	**12.94**	28.35	12.05
en-fi	23.51	11.00	24.00	11.13	**24.12**	**11.49**
en-de	28.71	14.48	**28.92**	**16.07**	**28.92**	14.71
en-ru	23.21	14.21	**23.74**	**15.94**	23.68	15.25
en-tr	**18.67**	6.46	18.58	**7.36**	18.24	6.81
cs-en	38.14	17.18	38.34	**17.26**	**38.37**	16.90
fi-en	26.58	13.80	26.94	10.77	**27.08**	**15.88**
de-en	33.02	18.88	**33.40**	19.16	33.01	**19.24**
ru-en	30.52	18.40	30.15	17.67	**30.58**	**18.71**
tr-en	**25.84**	10.34	25.64	**10.79**	25.32	10.65

Table 4: NMT models trained on subtitle data with and without filtering. BLEU scores in % on heldout data and news test data from WMT 2017.

We can see that the differences are small and the effect of filtering is not always beneficial. There are several reasons why the results are inconclusive. First of all, our training procedures were rather limited and all models have only seen a fraction of the entire data. Furthermore, our choice of leaving out all subtitles from 2017 as heldout data removed a large portion of the additional data that we include in our new release and the positive effect is not as visible as it could be. Finally, we did not perform a systematic study on optimising the threshold for filtering the training data. It is possible that too much of the valuable training data is left out and the coverage is reduced.

5 Conclusion

This paper presented OpenSubtitles2018, a new major release of the OpenSubtitles collection of parallel corpora. One important addition to this release is the estimation of a quality score associated with each sentence pair. The scores are determined through a feedforward neural network estimated from a sample of sentence pairs.

To train the neural network, lexical translation probabilities are computed for each sentence pair and employed as an indirect measure of alignment quality. The approach relies on simple, generic features such as the relative sentence length, number of empty alignments, number of cognates, or similar punctuation.

The neural network is able to explain 47 % of the variance of the quality score based on these features. A subsequent manual analysis also showed that sentence pairs assigned with a low quality score were indeed the result of misalignments. However, initial experiments with neural machine translation models do not demonstrate a conclusive advantage to filtering the bitexts based on these quality scores, suggesting that more work is needed to find the right filtering threshold.

References

Akbik, A., Kumar, V., and Li, Y. (2016). Towards semi-automatic generation of proposition banks for low-resource languages. In *EMNLP*, pages 993–998.

Aziz, W., de Sousa, S. C. M., and Specia, L. (2012). Cross-lingual sentence compression for subtitles. In *16th Annual Conference of the European Association for Machine Translation (EAMT 2012)*, pages 103–110, Trento, Italy.

Bolstad, B. M., Irizarry, R. A., Åstrand, M., and Speed, T. P. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C. J., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)*, pages 177–180, Prague, Czech Republic.

Koehn, P. (2009). *Statistical machine translation*. Cambridge University Press.

Lison, P. and Bibauw, S. (2017). Not all dialogues are created equal: Instance weighting for neural conversational models. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2017)*, pages 384–394, Saarbrücken, Germany. ACL.

Lison, P. and Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*.

Mikhailov, M. and Cooper, R. (2016). *Corpus Linguistics for Translation and Contrastive Studies: a guide for research*. Routledge.

Neubig, G., Nakata, Y., and Mori, S. (2011). Pointwise prediction for robust, adaptable japanese morphological analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, pages 529–533, Portland, Oregon, USA.

Och, F. J. and Ney, H. (2000). Giza++: Training of statistical translation models.

Östling, R., Scherrer, Y., Tiedemann, J., Tang, G., and Nieminen, T. (2017). The Helsinki neural machine translation system. In *Proceedings of the Second Conference on Machine Translation*, Copenhagen, Denmark, September. Association for Computational Linguistics.

Paetzold, G. and Specia, L. (2016). Collecting and exploring everyday language for predicting psycholinguistic properties of words. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1669–1679, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Paetzold, G. H. (2016). *Lexical Simplification for Non-Native English Speakers*. Ph.D. thesis, University of Sheffield.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Tiedemann, J. (2008). Synchronizing translated movie subtitles. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*.

Tiedemann, J. (2016). Finding alternative translations in a large corpus of movie subtitles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC-2016)*.

van der Wees, M., Bisazza, A., and Monz, C. (2016). Measuring the effect of conversational aspects on machine translation quality. In *COLING*, pages 2571–2581.

Vinyals, O. and Le, Q. (2015). A Neural Conversational Model. *CoRR*, abs/1506.05869.

Building an Ellipsis-aware Chinese Dependency Treebank for Web Text

Xuancheng Ren[1,2], Xu Sun[1,2], Ji Wen[1,2], Bingzhen Wei[1,2], Weidong Zhan[1,3], Zhiyuan Zhang[1,2]
[1]MOE Key Laboratory of Computational Linguistics, Peking University
[2]School of Electronics Engineering and Computer Science, Peking University
[3]Department of Chinese Language and Literature, Peking University
Beijing, China
{renxc, xusun, wenjics, weibz, zwd, zzy1210}@pku.edu.cn

Abstract

Web 2.0 has brought with it numerous user-produced data revealing one's thoughts, experiences, and knowledge, which are a great source for many tasks, such as information extraction, and knowledge base construction. However, the colloquial nature of the texts poses new challenges for current natural language processing techniques, which are more adapt to the formal form of the language. Ellipsis is a common linguistic phenomenon that some words are left out as they are understood from the context, especially in oral utterance, hindering the improvement of dependency parsing, which is of great importance for tasks relied on the meaning of the sentence. In order to promote research in this area, we are releasing a Chinese dependency treebank of 319 weibos, containing 572 sentences with omissions restored and contexts reserved.

Keywords: ellipsis, dependency treebank, web text

1. Introduction

With the rapid development of Web 2.0, the internet provides us with numerous raw text, which is a great source for structured learning such as information extraction. However, the information in Web 2.0 is usually highly unstructured and of various forms, written in a more casual way, which makes it hard to directly apply techniques that works for regular text. One probable way to structure the text is through syntax parsing, which exposes the syntax relation, and implicitly semantic relation, between words, especially dependency parsing (Tesnière, 1959). However, due to the different construction of the sentences in web text, the current parsing methods face greatly challenge (Kübler et al., 2009; Petrov and McDonald, 2012). Web text usually represents more of a spoken language aspect than the written language aspect, which means the social effect is more dominant in its formation, and the efficiency in conveying the information is more valued. In this work, we focus on building a resource to facilitate the parsing of the web text, specifically text from online microblogs.

A common issue of the current techniques is that they lack the ability to deal with omitted elements in the text, which are more often referred as ellipsis in linguistics. Ellipsis is a common linguistic phenomenon across languages, which facilitates the communication in real-world conversations, as the omitted elements should be understood from the context. The down side is that the sentence is often made less structured, and cannot easily be understood if extracted from the context. This work is a first step towards establishing a more robust parsing framework through recognizing the ubiquitous ellipsis in web text and identifying the omitted elements from such type of sentences. We build a Chinese dependency treebank based on the messages extracted from Weibo[1], in which we restore the omitted elements, and hope it could help advance the study in this area and improve the peformance of dependency parsing of web text.

To our knowledge, this is the first study that seeks to build a treebank with focus on ellipsis in context for Chinese. Chinese Treebank (Palmer et al., 2005), which is initially a constituent treebank, and then converted to a dependency treebank (de Marneffe et al., 2006), incorporates the idea of empty category from the government and binding theory of Chomsky (1993), but is fundamentally different from ours, because empty categories usually cannot appear in a legal sentence, while ellipsis means the omitted elements can always be restored, and are omitted just for the convenience of communication. In practice, the empty category annotation is often being ignored when building a parser, making the annotation more or less useless (Campbell, 2004). Universal Treebanks (McDonald et al., 2013; de Marneffe et al., 2014; Nivre et al., 2016), considering the ellipsis as a less important phenomenon, deal with it by promoting the omitted words' dependents, or use the special *orphan* relation, which either produces a confusing dependency, or isolates the relation between the part with ellipsis and the part without ellipsis. This solution makes it harder for the parser to learn from the data.

It is also necessary to point out the difference from the study of pronoun-dropping, where certain classes of pronouns may be omitted when they are inferable. For example, in the sentence "谁说的？"(Who say $DE_{particle}$, i.e., "Who said it?"), the object of "说"("say"), that is, "it" in the translation, is omitted or dropped. There is a huge amount of research regarding Chinese from a theoretical linguistic view. The concept is overlapped with the concept of

The treebank is available at https://github.com/lancopku/Chinese-Dependency-Treebank-with-Ellipsis.

[1]Weibo is China's most popular microblogging service, in comparison with Twitter. In the following, we will refer to the platform as Weibo, and each message the user posts as a weibo, just like a tweet.

ellipsis, more specifically contained in the concept of ellipsis. While in pronoun-dropping the omitted elements is supposed to be pronouns or noun phrases, the constraint does not apply to ellipsis. In spoken Chinese, verb phrases can also be omitted. To our knowledge, pronoun-dropping is also not applied or reflected when building treebanks.
In what follows, we will start by explaining ellipsis in Chinese Language, then briefly review the main considerations and the main steps we take in order to produce consistent and helpful annotations for the related natural language processing tasks, and finally introduce the annotated dataset with basic analysis of its attributes.

2. Ellipsis

To annotate ellipsis in sentences, we first need to define what should be considered as ellipsis. In linguistic theory, the specification of ellipsis has been a unsolved issue for a long time. In traditional natural language processing, ellipsis is not recognized as an important factor, which may be a measure of expediency at the time, as the goal is to process the regular form first. That no longer holds with the rapid adoption of Web 2.0, characterized by the publicness, informality, and the causal expression.
Following the theories in Chinese linguistics (Chao, 1965), we try to give a definition of ellipsis in the practice of natural language processing, whose goal is to assist the dependency parsing, which reveals the syntactic role and, to some extent, semantic role of the word in a sentence.

2.1. Ellipsis in Chinese Language

We define ellipsis in Chinese as textual omission of syntactic components, specifically words or phrases, expressing a semantic role in a sentence, that are optional but not obligatory in an utterance, and if elided, given the context of the sentence, the exact wording or, if referring to a object or concept, at least a board category of what the ellipsis refers to, shall be determined. There are four main parts in the definition, which are:

- It happens at the level of words and phrases. The omission of characters or morphemes is not considered ellipsis.
- The elided words must express meaning related to the sentence. If the words are not helpful to the understanding of the sentence, the omission is ignored.
- The elided words can be said. If they are said but the resulting sentence becoming incorrect or illegal, that kind of omission is not what ellipsis considers.
- The elided words can be determined from the context. If we do not know what the elided refers to or stands for, we do not treat the omission as ellipsis.

The definition is the guiding principle in our annotation of the text. It is worth noticing that the definition is not from a pure syntax view. As the goal of the annotation is to broaden the use of the web text, the semantic side is paid more attention to.

3. Construction of the Dataset

Following the definition, we develop several considerations in the construction of the dataset, which we will introduce in Section 3.1., then we will show our annotation procedure in Section 3.2., and finally we will explain the format of the dataset in Section 3.3..

3.1. Considerations

Speakability The omission can be said in a regular sentence, which means the omission is optional rather than obligatory. For example, in the well-known example, “我请他吃饭。”(I invite him eat meal, i.e., “I invite him to eat a meal.”), “他”(“him”) is both the object of “请”(“invite”) and the subject of “吃”(“eat”), which causes an illegal situation where two dependency point at “他”(“him”). It’s common belief that a “他”(“him” or “he”) is dropped from the sentence[2]. However, if restored, the resulting sentence “我请他他吃饭。”(I invite him he eat meal, i.e., “I invite him to eat a meal.”) is unspeakable for a Chinese speaker. Although the omitted word play a syntactic and semantic role in the sentence, due to the speakablility, we do not restore the dropped elements[3].

Identifiability The omission must be known from the context. In traditional Chinese grammar, this constraint is so strong that it requires the omission can be restored uniquely and unambiguously. However, in practice, to make a single sentence semantically reasonable, we are aware that some words are missing, and we know vaguely what they refer to, but we cannot restore the words uniquely. For example, in the sentence “这样就没了。” (This way just not-exist $LE_{particle}$, i.e., “It has gone this way.”), the subject of “没”(not-exist, i.e., “vanish”) is omitted, which causes the problem that most parser will treat “这样”(this way, i.e., “this way”) as the subject. However, most of the time, we don’t know what the exact wording of the omission is, but we are aware that it must be a noun and represent a thing. Due to the semantic importance, we introduce several categories in this case, that is THG, EVT, PPL, and OTH, representing concrete or abstract things, events, people-like perceivers, and others respectively.

Necessity If the omission does not affect the syntactic or semantic side of the sentence, we just ignore it and do not restore the omission. For example, in the sentence “我喜欢看书，但她不喜欢。” (I like read book, but she not like, i.e., “I like reading books, but she doesn’t.”), “但”(“but”), which is an adversative conjunction, is a short form of “但是”(but be, i.e, “but”) with the character “是”(be) omitted. However, the character “是”(be) here does not affect the syntax or meaning of the sentence. Hence, we do not restore “是”(be) here, although in a more formal utterance, “是”(be) is needed[4].

[2]However, it has not reached consensus whether the subject is dropped or the object is dropped.

[3]This is called PRO-drop from the view of empty category. (Huang, 1989)

[4] Nonetheless, we must point out that the sentence is indeed an elliptical construction. But the formation is different between Chinese and English. The Chinese one elides the object of “喜欢”(“like”), but keeps the predicate, while the English one omits

3.2. Annotation Procedure

We will first give a brief introduction to the raw text we used, and then describe the annotation procedure we practiced.

The raw text we used is Leiden Weibo Corpus (LWC)[5], which consists of more than 5 million messages posted on Weibo in January 2012. The period contains normal weeks, as well as the Chinese New Year holiday, which makes the corpus includes general topics in a normal weeks and also a prominent topic, i.e. the Chinese New Year.

The main reason that we choose the text from microblogs is that text from microblogs is more useful than the text from typical blogs for information extraction. Besides, the text from microblogs is often more oral and casual, which means the ellipsis probably occurs more. While some may argue that the ellipsis comes from the 140 characters limitation rather than the attributes of the language itself, we believe that, different from the 140 letters limitation of a tweet, a lot things can be conveyed in 140 Chinese characters of a weibo, and the character limitation is not a dominant factor to the ellipsis.

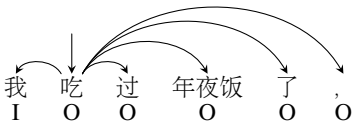
(1) 吃过年夜饭了，还有亲没吃吗?

(2) a. 吃过年夜饭了，
    b. 还有亲没吃吗?

(3) a. ` 我` 吃过年夜饭了，
    b. 还有亲没吃` 年夜饭` 吗?

(4) a. ` 我` /吃/过/年夜饭/了/，
    ` I` eat *HAVE*$_{particle}$ family-reunion-dinner *LE*$_{particle}$,
    "I have eaten the family reunion dinner."
    b. 还有/亲/没/吃/` 年夜饭` /吗/?
    any-other person not eat ` family-reunion-dinner` *MA*$_{particle}$?
    "Has any of you not eaten the family reunion dinner?"

(5) a.



b.

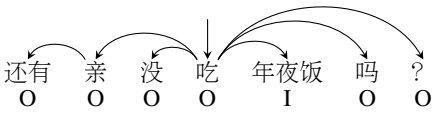


Figure 1: An example of the annotation procedure.

The annotation is done in the following order, and an example is given in Table 1.

Selection As the corpus is much too large for our purpose, we randomly select weibos from the corpus. A major attribute of ellipsis is that it should be understood from the context. However, in LWC, a single weibo is a stored unit, lacking the relations between weibos, such as forwarding and replying. Hence, some omissions just cannot be restored, even if we use the aforesaid general categories. Besides, there are also tons of advertisements and chaos messages in the corpus. To deal with the problem, we further purge the randomly selected weibos by considering the content of the weibos. We only keep the weibos that are roughly about normal daily topics, and drop the rest. In the end, we got 319 weibos from 500 random chosen weibos. The weibos contain a total of 8,382 tokens.

Sentence Split We then manually split the sentences in a weibo, as a weibo generally contains more than one sentences, but the sentence split is often implicit, due to the casual use of punctuations. Besides, sentences, in the sense of English or other Indo-European languages, in Chinese does not only end at the periods, but also can stop at the commas. The reason is that sentences are normally formed on a topic level, and several sentences, which are independent from each other grammatically, and lack conjunctions in between, are grouped as a sentence because they related to a certain topic. By manually splitting the sentences, a weibo can be seen as the context of its sentences, which is necessary for omission restoration tasks. The split standard is that if the topic or the subject is changed between the parts split by commas, we split the parts. This step gives us 572 sentences from 319 weibos.

Restoration Restoration is the major step in our annotation, and also the step we pay the most attention to. The considerations in annotation is explained in Section 3.1.. In this step, we restored 208 words of 256 characters.

All of the previous steps are done in the .txt files.

Word Segmentation From this step, the annotation becomes more demanding, and we gain assistance from the annotation tool brat (Stenetorp et al., 2012)[6] and the Stanford CoreNLP tool[7]. The word segmentation procedure basically follows the guideline of CTB (Xia, 2000). The difference is that we treat all the words, which typically only appear in web text, as single words, regardless of the boundness of its morphemes/characters. For example, the word "给力" (give power, i.e. "forceful") is not segmented into "给" ("give") and "力" ("power"). We also mark the restored words with the tag "I" to distinguish from the original words. The word segmentation annotation generated from Stanford CoreNLP tool are considered as baselines, which are already available in LWC. A total of 8,590 words are found in this step, including the restored words.

Dependency Arc Annotation This is the core of any dependency treebanks, which is also the most challenging part. With limited resources, this step is simplified as much as possible. We only annotate the dependency relation between words, excluding the type of the dependency, and some words are dropped in this step, if the words are emoji converted words, cause the dependency tree to be non-projectable or are less relevant to the sentence. For example, there are a lot of interjections, or called exclamations, such as "嗯", "哦", and "哈" in the sentences. Although they are part of the sentence, and often are at the start, we do not annotate their dependencies, because the relation between them and the sentence is rather weak. Nonetheless,

both the predicate and its object, which is particularly frequent in English, commonly known as verb phrase ellipsis or VPE.

[5]http://lwc.daanvanesch.nl/

[6]http://brat.nlplab.org/

[7]https://stanfordnlp.github.io/CoreNLP/

we do annotate the modal particles' dependencies, such as "吗", "呢", and "吧", as they do affect the modal of the sentence. For example, "吗" is a distinctive mark for interrogative sentence. This gives us 8,018 dependencies.

3.3. Annotation Format

There are two types of files after the annotation. One is a text file containing the split weibos and restored content, and the other is an annotation file, containing the word segmentation, omission restoration, and the dependency annotation.

In the text file, each line contains a sentence, and a blank line separates weibos. Each restored element is surrounded by the grave accent, i.e., the back quote mark " ` ". The original " ` " in the file is changed to "⟨ ` ⟩", which is very rare in our annotation.

The annotation file is of the same format with the brat's default format. Each line is an annotation entry, either a word or a dependency.

For the word annotation, the line is of format "⟨wid⟩\t⟨wtag⟩ ⟨start⟩ ⟨end⟩\t⟨word⟩", where ⟨wid⟩ is the word's unqiue identifier in a file, \t means a tab, ⟨wtag⟩ is either "I", meaning a restored element, or "O", meaning an original element, the ⟨start⟩ and the ⟨end⟩ give the start offset and the offset after the end of the word from the start of the file, and ⟨word⟩ is the original form of the word in the text. Please notice, the replaced " ` " is also changed to its original form, and the restored element mark " ` " is not annotated.

For the dependency annotation, the line is of format "⟨did⟩\t⟨dtag⟩ Arg1:⟨hid⟩ Arg2:⟨cid⟩", where ⟨did⟩ is the dependency's unqiue identifier in a file, ⟨dtag⟩ can only be "dep" as we do not differentiate the dependency types, and ⟨hid⟩ and ⟨cid⟩ give the head and dependent word ID of the dependency.

For natural language processing usage, the annotation file is adequate for ellipsis restoration, word segmentation, and dependency parsing tasks, as the related annotations are all reflected in the file.

Furthermore, we combined the annotation files into a single file in the tsv format. There are four columns in the file. The first column is the token's index in the sentence, starting from 1. The second column is the textual form of the token. The third column indicates whether the token is a restored one. "O" stands for original tokens, and "I" stands for restored tokens. The fourth column is the head of the token. 0 indicates the token is the root. There is an empty line between sentences, and an extra empty line between weibos.

4. Dataset

We are releasing a first version of the dataset, containing 8,590 tokens, 572 sentences, and 319 weibos (Table 1). The raw text is from LWC, a weibo corpus. Unsurprisingly, due to the characteristics of microblogging, the average length of the sentences are quite short, around 15.0 tokens per sentence, comparing to 27.0 tokens per sentence in CTB5. We have restored 256 characters and 208 words in the dataset. As shown in Table 1, ellipsis is indeed a common phenomenon in web text, which requires more attention, as 162 of the sentences, and 122 of the weibos contain ellipsis, meaning 38.24% of the weibos involve ellipsis.

Type	#Token	#Word	#Sentence	#Weibo
Original	12,508	8,382	572	319
Ellipsis	256	208	162	122
Overall	12,764	8,590	572	319
Percentage (%)	2	2	28	38

Table 1: Statistics of the dataset. Each column representing the term that are being counted, 1: the type of the term; 2: number of tokens or characters; 3: number of words; 4: number of sentences containing the type of the term; 5: number of weibos containing the type of the term. We can see that ellipsis is indeed common in the annotated dataset, as more than half of the weibos containing ellipsis.

Head → Dependent	#Depedency
Original → Original	7,762
Original → Ellipsis	187
Ellipsis → Original	61
Ellipsis → Ellipsis	8
Overall	8,018

Table 2: Statistics of the treebank. The first column describes the type of the elements involved in a dependency. The second column gives the count of the dependency.

We further show the statistics of the annotated dependencies of the dataset in Table 2. There are in total 256 dependencies involving ellipsis. As there are only 208 restored words, some restored words are linked to multiple original words. In addition, restored words serve as heads in 61 dependencies, which means, if not restored, the omission will cause dependent promotion or orphan relation just like in Universal Treebanks, harming the syntactic soundness of the dependency tree. To our surprise, in 8 dependencies, both the head and the dependent are omitted, which may indicate some of the restored omissions may be redundant. We will further investigate the case and try to refine the guidelines, if needed, in the next version of the dataset.

5. Efforts in Maintaining Quality

At the beginning of the annotation, it is unsurprising to see that the annotators hardly fully agree on annotations of one weibo. As the annotation task involves several steps, some of which are really hard even to senior students majoring in linguistics, such as restoration and dependency parsing. Besides, the corpus is rather informal, usually with unusual utterance and wrong characters. In order to maintain the quality of the annotations, and the consistency across the dataset, we make the following supervision efforts.

Several Peking University students majoring in computational linguistics were chosen to conduct the annotation study. Each student was given an annotation guideline and ten sample weibos, which we developed gold standard annotations. We compared their answers with the standard annotations, pointed out the differences, then provided specialized guidance for the students and revised the guideline as needed. (Pyysalo and Ginter, 2014) The test was

repeated, until the quality of the annotations of the ten samples are met. Then, the students were given real text to annotate.

Due to limited annotators, each weibo was only annotated once after the test phase. The workload is divided into units of 10 weibos. Each time, a student is given a unit to annotate. After the unit is annotated, we randomly choose one weibo to double-check. If the quality is met, the unit is considered okay. If not, the rest weibos are all checked, and we select the qualified weibos, and discard the bad ones. By doing this, we could maximize the annotation speed, and, at the same time, keep a basic and satisfying quality of the first version of the dataset.

As the restoration standard is hard to unify, we make a list of the aspects that should be considered when restoring an element. For example, if a sentence is a subject-predicate construction, the elided subject must be restored, because it is very common in oral Chinese to omit the subject, which most of the time is also the topic of the conversation, but it can cause serious problems for parsers, as the subjects in consecutive sentences may refer to different things.

In annotation, especially the word segmentation step and dependency parsing step, to alleviate the workload of the annotators, we use off-the-self tools, specifically the Stanford CoreNLP tools, to automatically generate the related references. The annotators could use the annotations generated as a baseline, and rectify the wrong dependencies as they recognize. This could further facilitate the annotation procedure, and maintain the quality of the dependency at least above the performance of the used parser, which is about 83.9 in terms of UAS on the test set of CTB5 (Chen and Manning, 2014). In future revisions, tools with better accuracy and faster speed may be considered (Sun et al., 2013; Sun et al., 2014; Dyer et al., 2015; Sun, 2014; Xu and Sun, 2016; Kiperwasser and Goldberg, 2016; Sun, 2016; Sun et al., 2017; Zhang et al., 2017; Ren and Sun, 2017).

The reason for these relaxations is that the main focus of the treebank is the annotation of ellipsis, and the study of the effect of ellipsis on dependency parsing, so as to other semantic related tasks. We shift our annotation focus to the annotation of sentence split, omission discovery and restoration, which are less demanding than the annotation of dependency trees, after we find the annotation process is too slow.

6. Conclusion

In this study, we introduce a practical definition for ellipsis in Chinese. We also introduce a practical scheme for ellipsis annotation, and build an ellipsis-aware Chinese dependency treebank for web text, where the elided is restored, and the necessary context is reserved. The dataset contains 572 sentences from 319 weibos, including 208 restored omissions, forming 8,018 dependencies. We are releasing an initial version of the dataset, and we hope the dataset will advance the study in ellipsis restoration and the following tasks in a natural language processing pipeline.

7. Acknowledgements

We thank the anonymous reviewers for their insightful comments. This work was supported in part by National Natural Science Foundation of China (No. 61673028), National High Technology Research and Development Program of China (863 Program, No. 2015AA015404), and an Okawa Research Grant (2016). Email correspondence to Xu Sun.

8. Bibliographical References

Campbell, R. (2004). Using linguistic principles to recover empty categories. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 645–652. ACL.

Chao, Y. R. (1965). *A grammar of spoken Chinese.* ERIC.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 740–750. ACL.

Chomsky, N. (1993). *Lectures on government and binding: The Pisa lectures*. Walter de Gruyter.

de Marneffe, M., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006*, pages 449–454. European Language Resources Association (ELRA).

de Marneffe, M., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal stanford dependencies: A cross-linguistic typology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 4585–4592. European Language Resources Association (ELRA).

Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015*, pages 334–343.

Huang, C. J. (1989). Pro-drop in chinese: A generalized control theory. In *The null subject parameter*, pages 185–214. Springer.

Kiperwasser, E. and Goldberg, Y. (2016). Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL*, 4:313–327.

Kübler, S., McDonald, R., and Nivre, J. (2009). Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127.

McDonald, R. T., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K. B., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Castelló, N. B., and Lee, J. (2013). Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguis-*

tics, ACL 2013, Volume 2: Short Papers, pages 92–97. The Association for Computer Linguistics.

Nivre, J., de Marneffe, M., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation, LREC 2016*. European Language Resources Association (ELRA).

Petrov, S. and McDonald, R. (2012). Overview of the 2012 shared task on parsing the web. In *First Workshop on Syntactic Analysis of Non-Canonical Language*.

Pyysalo, S. and Ginter, F. (2014). Collaborative development of annotation guidelines with application to universal dependencies. In *Proceedings of The Fifth Swedish Language Technology Conference, SLTC 2014*.

Ren, X. and Sun, X. (2017). Hybrid oracle: Making use of ambiguity in transition-based chinese dependency parsing. *CoRR*, abs/1711.10163.

Stenetorp, P., Pyysalo, S., Topic, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a web-based tool for nlp-assisted text annotation. In *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. The Association for Computer Linguistics.

Sun, X., Zhang, Y., Matsuzaki, T., Tsuruoka, Y., and Tsujii, J. (2013). Probabilistic chinese word segmentation with non-local information and stochastic training. *Inf. Process. Manage.*, 49(3):626–636.

Sun, X., Li, W., Wang, H., and Lu, Q. (2014). Feature-frequency-adaptive on-line training for fast and accurate natural language processing. *Computational Linguistics*, 40(3):563–586.

Sun, X., Ren, X., Ma, S., and Wang, H. (2017). meProp: Sparsified back propagation for accelerated deep learning with reduced overfitting. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3299–3308, International Convention Centre, Sydney, Australia, 06–11 Aug. PMLR.

Sun, X. (2014). Structure regularization for structured prediction. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 2402–2410.

Sun, X. (2016). Asynchronous parallel learning for neural networks and structured models with dense features. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference*, pages 192–202.

Tesnière, L. (1959). *Eléments de syntaxe structurale*. éd. Klincksieck.

Xia, F. (2000). The segmentation guidelines for the penn chinese treebank (3.0). Technical report, University of Pennsylvania Institute for Research in Cognitive Science.

Xu, J. and Sun, X. (2016). Dependency-based gated recursive neural network for chinese word segmentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016*.

Zhang, X., Sun, W., and Wan, X. (2017). The covert helps parse the overt. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 343–353, Vancouver, Canada, August. Association for Computational Linguistics.

9. Language Resource References

Palmer, Martha and Chiou, Fu-Dong and Xue, Nianwen and Lee, Tsan-Kuang. (2005). *Chinese Treebank 5.0*. Linguistic Data Consortium (LDC), Chinese Treebank, 5.0, ISLRN 426-628-131-806-1.

EuroGames16: Evaluating Change Detection in Online Conversation

Cyril Goutte, Yunli Wang, Fangming Liao,[†] Zachary Zanussi,[‡] Samuel Larkin, Yuri Grinberg
National Research Council Canada
1200 Montreal Rd, Ottawa ON, Canada
{Cyril.Goutte, Yunli.Wang, Samuel.Larkin, Yuri.Grinberg}@nrc.ca, will.liao@mail.utoronto.ca, zzanussi@shaw.ca

Abstract

We introduce the challenging task of detecting changes from an online conversation. Our goal is to detect significant changes in, for example, sentiment or topic in a stream of messages that are part of an ongoing conversation. Our approach relies on first applying linguistic preprocessing or collecting simple statistics on the messages in the conversation in order to build a time series. Change point detection algorithms are then applied to identify the location of significant changes in the distribution of the underlying time series. We present a collection of sport events on which we can evaluate the performance of our change detection method. Our experiments, using several change point detection algorithms and several types of time series, show that it is possible to detect salient changes in an on-line conversation with relatively high accuracy.

Keywords: social media, change detection, online conversation

1. Overview

Social media, microblogs and news sources produce massive streams of textual data. Real world events and changes have an impact on these streams. For example, a significant action in a sport event will result in a flurry of positive or negative posts on twitter. Monitoring message streams to detect these changes requires automatic methods relying on a mixture of natural language processing and statistical modeling.

Social media analysis typically operates from two types of data: *raw* stream data and *filtered* stream data. Event detection detects either emerging events or specific events from raw stream data. In particular, the large amount of work done on *Topic Detection and Tracking* (TDT) attempts to detect emerging events. As a recent example, Laban and Hearst (2017) collected 4 million news articles, generated topics from the news, merged them into stories, and visualized the stories along a timeline. For *specific* event detection, Atkinson et al. (2017) created a corpus of security-related events extracted from news data by learning lexico-semantic patterns, and classified events into security-related categories.

Filtered stream data is usually generated by filtering social media using keywords related to public security, natural disaster etc. The TREC 2014 temporal summarization track focused on monitoring events by detecting sub-events, extracting relevant sentences and summarizing them, from a sequence of stream data (Aslam et al., 2014). Zhao et al. (2014) retrieved relevant documents, calculated the text similarity of sentences within the time period, and used k-means to cluster relevant sentences, using the cluster centers and the top sentences as summarization.

In our work, we do not address topic detection and tracking *per se*, but target the detection of significant changes, typically within an existing topic or event. We do not summarize changes like in the TREC 2014 temporal summarization task, but summarization can be used as a post-processing step. Earlier work used change point detection techniques to detect significant changes from sensor signals (Guralnik and Srivastava, 1999; James et al., 2014), but these do not use the textual content of message streams. Some recent studies focus on detecting changes within a storyline. For example, Bruggermann et al. (2016) used the dynamic topic model (Blei and Lafferty, 2006) to identify topics from news and the changes in the word distributions from the topic model were used to represent changes within the storyline. Also, Wang and Goutte (2017) detected changes within events from the temporal profile of hashtags in tweets and evaluated the resulting performance on two twitter datasets.

Our approach relates to some of this prior work, with clear distinctions. First, we target the detection of significant changes *within* an existing event or storyline, rather than detect and track events as in the TDT setup. Second, we focus on detecting the locations of significant changes from the message stream, rather than extract descriptive phrases from the text. Also, instead of detecting changes from external signals obtained from sensors or signals such as stock ticks, we use linguistically motivated signals obtained through text analysis pre-processing.

To establish a benchmark on detecting changes in online conversation, we collected a dataset of 16 sport events, with reference change points for each event. For the purpose of this paper, we will refer to a stream of messages related to a specific topic (e.g. a game, or a current event) as an *online conversation*. Our purpose is to detect, from that online conversation, the location of significant changes within the event. We do this by analyzing the content of the messages in order to produce one or several *time series* describing, for example, the sentiment or the topic of the conversation. We then use *change point detection* algorithms on these time series in order to detect locations where the underlying stochastic process changes. This typically is a change in mean, but also e.g. in variance or other distributional property. We apply this approach and benchmark it on the acquired collection of sport games conversations. In partic-

[†] from University of Toronto, ON, Canada
[‡] from Carleton University, ON, Canada

Game	Hashtag	# Eng.	# Total
Croatia-Spain	CROESP	52,953	115,328
England-Iceland	ENGISL	191,384	209,851
France-Albania	FRAALB	61,748	433,954
France-Ireland	FRAIRL	172,872	665,476
France-Iceland	FRAISL	158,457	720,771
Germany-France	GERFRA	273,074	496,498
Germany-Italy	GERITA	426,381	709,453
Germany-Poland	GERPOL	82,132	232,200
Poland-Portugal	POLPOR	128,079	663,612
Portugal-Austria	PORAUT	72,644	170,526
Portugal-France	FRAPOR	229,000	1,000,000
Portugal-Wales	PORWAL	287,417	461,343
Russia-Wales	RUSWAL	110,165	141,994
Switzerland-France	SUIFRA	36,507	468,043
Wales-Belgium	WALBEL	288,312	378,852
Wales-N. Ireland	WALNIR	95,679	114,723
Total	-	2.69M	7.04M

Table 1: Basic statistics on collection content.

ular, we test several change point detection algorithm and uncover their strengths and weaknesses.

In Section 2. we describe the collection that we acquired in order to benchmark change point detection algorithms. In Sections 3. and 4., we describe the linguistic preprocessing and change point detection algorithms, respectively. We then present experimental results, testing and validating this approach, in Section 5.

2. Collection

Our dataset contains messages related to 16 games from the 2016 UEFA European Championship[1], a continental football (soccer) competition held from 10 June to 10 July 2016. Dedicated hashtags comprised of three letter codes for each country were used to harvest messages from the twitter API, e.g. #FRAPOR or #PORFRA for the France-Portugal final game. Messages were also filtered by language, including English for all games. In the following, we only consider English messages, for which the relevant sentiment analysis tool is available. Statistics on the number of messages are given in Table 1. The number of messages per game is between 100k and one million, for a total of 7M messages. Most of the messages are posted over a period of about two hours, starting shortly before the game and ending shortly after.

In order to obtain a gold standard of reference events, we obtained game reports from sport websites, from which we semi-automatically extracted the main game events such as start/end of each period, goals, on-target attempts, substitutions, etc. The collection we release contains the multivariate sentiment time series, produced as described below, together with the reference labels for each of the 16 games. The processed dataset is available from: https://github.com/cyrilgoutte/EuroGames16

[1] https://en.wikipedia.org/wiki/UEFA_Euro_2016

3. Linguistic Preprocessing

The first stage in our approach is to turn the stream of messages into one or several time series. This can be done by using existing linguistic preprocessing methods such as sentiment analysis or topic models. As a running example below as well as in our experiments we use sentiment analysis for that purpose. Application to other linguistic preprocessing methods is similar.

We formalize a stream of N messages as a set $\mathcal{C} = \{(t_i, \mathbf{c}_i), i = 1 \dots N\}$, where t_i is the time associated with message i, and $\mathbf{c}_i$ is its textual content. We assume that the linguistic preprocessing produces, for each content $\mathbf{c}$, a set of scores describing the target linguistic properties. For example, we use the NRC sentiment analyzer (Kiritchenko et al., 2014) to produce, for each textual content $\mathbf{c}$, a set of three scores estimating the positive, negative and neutral polarity of the message: $\mathbf{c} \rightarrow \left(s^+, s^-, s^0\right)$. Processing the entire collection $\mathcal{C}$ results in a large table containing time and scores for each message, $\mathbf{S} = \left[t_i, s_i^+, s_i^-, s_i^0\right]_{i=1\dots N}$.

The posting times for messages are not uniformly distributed. In order to produce three times series for the positive, negative and neutral scores, we first *bin* the messages in time intervals of equal sizes, then average the scores within each bin. For simplicity and without loss of generalization, let us assume that posting times range from 0 to T, i.e. $0 \leq t_i \leq T, \forall i = 1 \dots N$. Dividing the range from 0 to T into $\lceil T/\Delta \rceil$ bins, each of width Δ, we build time series by averaging scores within each bin:[2]

$$s^+(t) = \frac{1}{|\mathcal{B}_t|} \sum_{i \in \mathcal{B}_t} s_i^+, \quad \mathcal{B}_t = \{i, (t-1)\Delta \leq t_i < t\Delta\}, \tag{1}$$

and similarly for $s^-(t)$ and $s^0(t)$ for the negative and neutral scores. We therefore obtain three time series of sentiment scores, or alternatively a multivariate, three dimensional time series, on which we run the change point detection algorithms. We later compare it with detection from raw message counts. This corresponds to a trivial preprocessing where each message is associated with a single score of 1 (performing binning and averaging as before), resulting in the time series $n(t) = |\mathcal{B}_t|$. Keyword profiles (Sec. 5.4.) are obtained similarly by recording the number of keywords in each bin.

4. Change Point Detection

In time series analysis, a change point is a location where the underlying stochastic process changes. Although it may seem superficially related, this is a different problem than anomaly detection, where the purpose is to identify observations that do not conform to an expected pattern or distribution in the data. In change point detection, we assume that data before the change point conforms to one distribution, while data after the change point comes from a second, different distribution. In our case, we are interested in identifying where the change has occurred, as soon as possible after it occurs.

Many algorithms have been proposed to detect change points. Most work on univariate time series, and use the

[2] When no message falls within a bin, it yields a missing value.

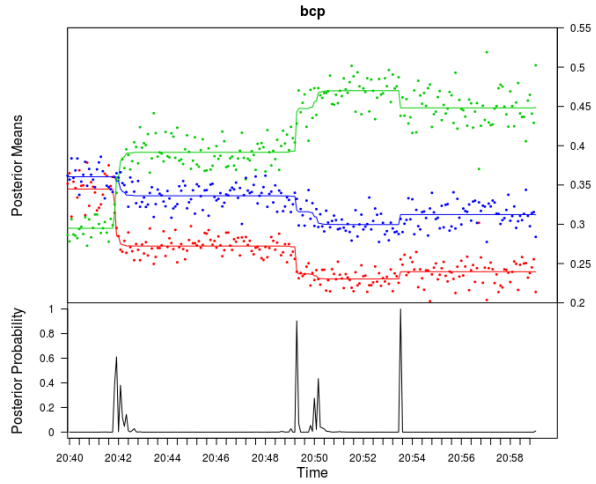


Figure 1: Top: Sentiment time series for the Wales v.s Belgium game in green/blue/red (positive/neutral/negative); Bottom: Posterior probability of change from `bcp`. At 20:42, Wales scores their third goal; at 20:50, Belgium is awarded three consecutive corner kicks, then injury time starts.

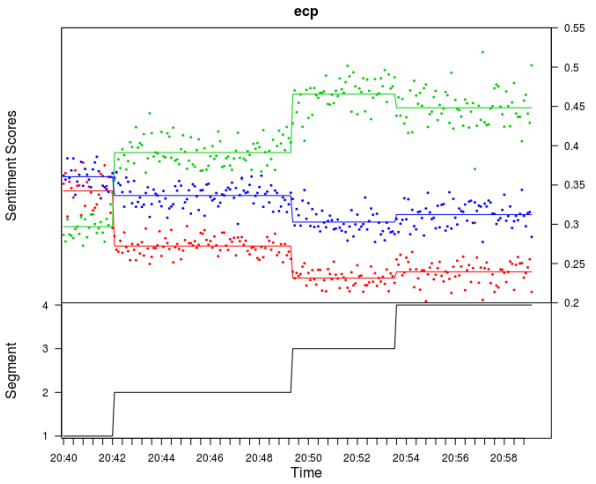


Figure 2: Top: Sentiment time series for the Wales v.s Belgium game in green/blue/red (positive/neutral/negative); Bottom: Segments output by `ecp`. At 20:42, Wales scores their third goal; at 20:50, Belgium is awarded three consecutive corner kicks, then injury time starts.

entire time series to detect change points. We will focus on techniques that can be used with multivariate time series:

bcp

The Bayesian change point detection of Barry and Hartigan (1993) assumes that each block between two change points arises from a (multivariate) normal distribution. It outputs the posterior probability that a change occurred at each point in the time series. Figure 1 illustrates this on a short extract from the sentiment time series for one of the games in the collection. We use the implementation from the `R` package `bcp` (Erdman and Emerson, 2007). The `bcp` algorithm runs fast: it is linear in the length of the time series, and handles multivariate time series. The biggest limitations are that it is designed to detect changes in the mean of independent Gaussian observations, and that it works off-line, once the entire time series is available.

ecp

The nonparametric, hierarchical divisive algorithm of James and Matteson (2015) uses recursive bisections, identifying change points using a non-parametric divergence measure from Székely and Rizzo (2005). As the divergence measure is non-parametric, this makes `ecp` suitable to detect changes with minimal assumptions on the underlying distributions. The divisive approach by recursive bisections returns a number of consecutive *segments* between change points, without knowing the number of change points *a priori*. In addition, the implementation from the `R` package `ecp` handles multivariate time series, as illustrated in Figure 2 on the same data as Fig. 1. One remaining limitation is that it works only in off-line mode, once the entire time series are available.

ocpd

The Bayesian online change point detection algorithm of Adams and MacKay (2007) is designed to update the detection of change points sequentially as new data points are acquired, rather than wait until the entire times series are available. It relies on two components: a probabilistic model $P(r_t|\mathbf{s}(1\dots t))$ of the length of a *run* during which the underlying distribution is stable, given observations until time t; and an underlying predictive model (UPM) $P(\mathbf{s}(t+1)|\mathbf{s}(1\dots t), r_t)$ governing the stochastic generation of new data in each run. Our basic implementation, available in the `R` package `onlineCPD`, uses a multivariate Gaussian UPM. Figure 3 shows the results obtained from `ocpd` on the same sentiment time series as before.

ocpd+

We extend the basic `ocpd` algorithm beyond the simple Gaussian assumption by using a more flexible UPM. We model the linear trends within each run, using a multivariate linear regression with additive Gaussian noise. This allows modeling drifts in the time series without forcing multiple change points. Our implementation in `R` will shortly be included in the `onlineCPD` package.

4.1. Online vs. offline

Despite different modeling inspirations and assumptions, the key differentiating feature of `ocpd`/`ocpd+` is that they work in *online* mode, updating the model and the detection at each step. When analyzing short events such as sports event in our collection, the difference may seem contrived. However, many real-life monitoring situations span days or months. This is the case when tracking changes in the days after a terror attack (Wang and Goutte, 2017) or in public health when following events related to epidemics over months or years. In those cases, it is clearly impractical to wait until all data acquisition is finished before running the

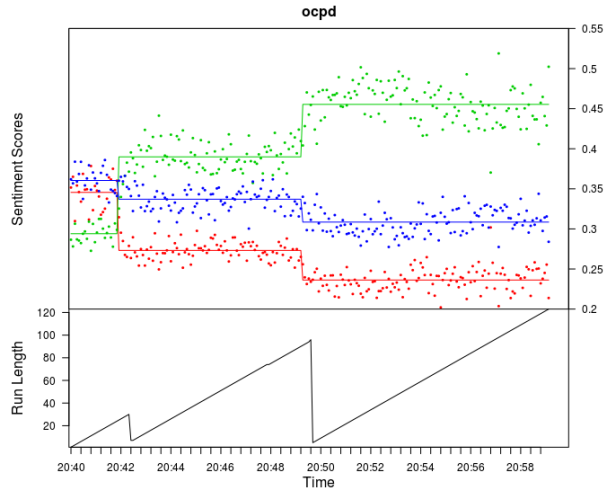


Figure 3: Top: Sentiment time series for the Wales v.s Belgium game in green/blue/red (positive/neutral/negative); Bottom: Maximum run length probability from `ocpd`. At 20:42, Wales scores their third goal; at 20:50, Belgium is awarded three consecutive corner kicks, then injury time starts.

change detection algorithm. Section 5.5. discusses this in more details.

4.2. Finding the number of change points

Both `bcp` and `ecp` can be used for univariate and multivariate time series without a priori knowledge of the number of change points. The posterior probability of change output by `bcp` may be thresholded to tune the output of the method, and in `ecp`, similarly, a threshold may be applied to the p-value estimating the significance of a divergence in distributions. However, in our experience, both methods work much better when a target number of changes is provided. We investigate the impact of this in our experimental section (Section 5.6.). On the other hand, `ocpd` and `ocpd+` provide a posterior distribution on the run length, from which it is possible to automatically detect the number of change points. Light post-processing may be used to avoid multiple detections around the same change.

5. Results

Each of the games listed above (Section 2.) was pre-processed (as described in Section 3.) in order to produce multivariate time series estimating the positive, negative and neutral sentiment[3] during each game. We use a bin size of $\Delta = 15s$ for these experiments, which yields good performance. We first look at the results from a particular game (Sec. 5.1.) before presenting our systematic evaluation in Sections 5.2. to 5.7..

5.1. Example

We use the Wales vs. Belgium game that took place on July 1st, 2016 to illustrate what the data and the output of the change point detection algorithm look like. Figure 4 shows the timeline from a few minutes before the game starts to minutes after it stopped. The shaded area in the background represents the volume of tweets, and shows high variability. Most spikes are associated to significant events in the game (light green, labeled lines), but this is not always the case (e.g. first and third yellow cards). Changes in the positive and negative scores (blue and red curves, smoothed) during the game also tend to match peaks in the tweet volumes, but are quite variable everywhere.

The detections produced by `ocpd+` (blue ticks) and `ecp` (bottom red ticks) show very different behaviours. `ocpd+` yields high precision: the detections are usually close to reference events; but it also misses several. On the other hand, `ecp` detects too many changes. As a consequence it yields high recall, but lower precision. Note that there is a qualitative difference between changes. Those related to predictable events such as half time or end of game tend to be detected early, while unpredictable game plays such as goals or yellow cards tend to be detected with a slight delay. This makes sense, as people may start tweeting about predictable events in anticipation, before they actually occur, while unpredictable events, by definition, can not be anticipated. More analysis would be required to check, e.g. whether false positives correspond to notable game plays that may not be recorded in our gold standard.

5.2. Off-line CPD

A systematic evaluation was carried out using precision, recall and F-score (Goutte and Gaussier, 2005) to evaluate the performance of change point detection algorithms on the 16 games. A detected change point was considered as a true positive if it falls into the time window 180 seconds on either sides of a reference change point. This allows to take into account that tweets need to be written and posted, as well as slight inaccuracies in the real timing of the reference game plays.

In these experiments, we run all algorithms *off-line*, on the entire dataset, which is the standard mode of operation for `bcp` and `ecp`. The `ocpd` and `ocpd+` algorithm are run one data point at a time, as designed, simulating an on-line operation over the entire dataset.

Table 2 shows the performance on all games for all change point detection algorithms applied to the sentiment time series. We see that `bcp` performs poorly overall. This may be due to the strong underlying Gaussian assumption, and the fact that it is very sensitive to non normally distributed noise, a problem we confirmed using simulated data (not reported here). The `ecp` algorithm performs well, obtaining the best F-score for 7 out of 16 games, often by a small margin (e.g. ENGISL, WALBEL, WALNIR). The `ocpd` algorithm behaves sometimes quite poorly, maybe due to the Gaussian assumption again, but it is typically close to (and sometimes better than) `ecp`. Finally, the more flexible assumptions underlying the `ocpd+` algorithm allows it to get the best results for 7 games and overall (Tables 3, 4).

5.3. Detection from Message Counts

We have run the change point detection algorithms on the three sentiment signals (positive/negative/neutral), taking advantage of the fact that they handle multivariate data. However, it is simple to run them on the univariate tweet

[3]Time series are not independent: sentiment scores sum to one.

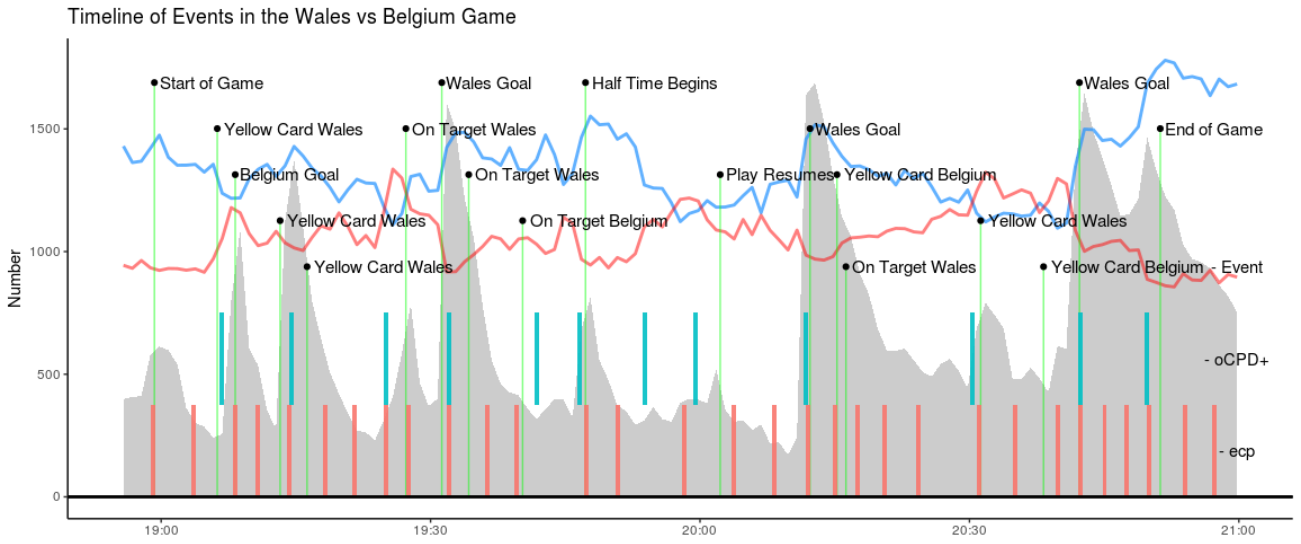


Figure 4: Timeline for the Wales v.s Belgium game. Tweet volume is in grey shade, and +/- sentiment time series in blue/red; reference events in light green, detected events in blue (ocpd+) and red (ecp).

Table 2: The F-scores of four algorithms on sentiment of Euro games in 15 seconds time interval

Game	bcp	ecp	ocpd	ocpd+
CROESP	0.267	**0.643**	0.444	0.500
ENGISL	0.085	**0.429**	0.333	0.414
FRAALB	0.390	0.480	0.111	**0.583**
FRAIRL	0.290	0.240	0.250	**0.400**
FRAISL	0.225	0.370	**0.538**	0.400
GERFRA	0.186	0.462	0.231	**0.467**
GERITA	0.217	**0.571**	0.300	0.516
GERPOL	0.211	0.462	0.400	**0.552**
POLPOR	0.212	**0.500**	0.174	0.370
PORAUT	0.069	0.267	0.160	**0.400**
FRAPOR	0.143	0.182	0.154	**0.303**
PORWAL	0.367	**0.563**	0.435	0.389
RUSWAL	0.266	0.417	**0.571**	0.444
SUIFRA	0.333	0.435	0.167	**0.500**
WALBEL	0.419	**0.703**	0.370	0.684
WALNIR	0.086	**0.190**	0.111	0.174
Average	.2354	.4315	.3047	**.4428**

frequency signal $n(t)$. On datasets such as sports games where tweet volume is correlated with significant changes, this works surprisingly well, cf. first row of Table 3. Pairwise comparisons (Table 4) also shows that ecp and ocpd+ are overwhelmingly more effective than the other two algorithms. As algorithms handle multivariate signal, we can also *add* counts as a fourth time series, in addition to the sentiment scores. In that case (last row, Table 3), performance jumps above what we obtained on counts and sentiments separately, and reaches 50.6% F-score using ocpd+.

Table 3: Performance on raw counts (top), sentiment (middle, with/without providing the target number of change points) and combined (bottom).

Input	bcp	ecp	ocpd	ocpd+
Count	.3434	.4645	.4250	**.4725**
Sentiment	.2354	.4315	.3047	**.4428**
+ # references	.3918	**.4860**	.3047	.4380
Count+Sentiment	.4251	.4645	.4010	**.5062**

Table 4: Pairwise comparisons of four algorithms on counts and sentiment scores, with/without providing the target number of change points. Cells report how many games the row method wins/ties/loses vs. the column method.

Counts	ecp	ocpd	ocpd+
bcp	2/0/14	5/0/11	2/0/14
ecp	-	11/1/4	**7/1/8**
ocpd	-	-	4/1/11
Sentiment	**ecp**	**ocpd**	**ocpd+**
bcp	1/0/15	5/1/11	0/0/16
ecp	-	12/0/4	**7/1/8**
ocpd	-	-	3/0/13
+ references	**ecp**	**ocpd**	**ocpd+**
bcp	0/0/16	4/0/16	0/0/16
ecp	-	14/0/2	**10/0/6**
ocpd	-	-	3/0/13

5.4. Detection from Keyword Usage

Another simple comparison is with the detection of change points from temporal profiles of keywords such as "goals", "begin", "end", etc. and hope that changes in the usage of these words can be captured by change point detection algorithm. There are two serious issues with this approach, however. One is obviously that it requires specific domain knowledge to pick the appropriate keywords: obvi-

Table 5: Performance of change point detection algorithms on temporal profiles of specific keyword usage.

Time Interval	bcp	ecp	ocpd	ocpd+
5 mins	**0.3748**	0.2413	0.1692	0.1773
1 min	0.1495	**0.2046**	0.1107	0.112

Table 6: Performance of change point detection algorithms in on-line mode, with different time intervals for binning.

Time Interval	bcp	ecp	ocpd	ocpd+
60 seconds	**0.4799**	0.3274	0.2012	0.3826
30 seconds	0.4693	0.4260	0.2996	**0.4810**
15 seconds	0.4391	0.4313	0.3625	**0.4624**

ously "goal" of "red card" would not be very useful keywords when tracking health- or security-related document streams. Another is that even domain-related keywords will be sparse in the collection. This may require to use larger bin sizes in order to capture any statistics at all on keyword usage patterns, leading to short temporal profiles and little information for the CPD algorithms to work with. We will see below (Sec. 5.5.) that this tends to yield inferior performance for most algorithms. Despite these objections, we benchmarked this baseline on the same 16 games. Table 5 presents the results using 1min and 5min time intervals. The pattern in the result is similar to what we saw in Table 6: bcp performs better on the larger time interval, ecp performs better on the shorter time interval, while ocpd and ocpd+ do not perform well on these large intervals and short time series. Overall, the performance is clearly much lower than what we obtained with our approach.

5.5. On-line CPD

In practice, it is often more useful to run the change point detection on-line. As discussed above, it would be inconvenient to wait until all data is acquired before the analysis is run. Since bcp and ecp are off-line change point detection methods, we simulate on-line operation by running sequentially over a sliding window on the time series: We do a first run on the first 50 minutes, then offset the window by 25 minutes and run the algorithms again on the data from minute 25 to minute 75, and so on until the end of the time series. In principle, this provides change point detections with at most a 25 minute delay. Although the ocpd and ocpd+ algorithms are on-line by design, we run them over the same sliding window in order to have a fair comparison and evaluate the impact on performance.

Further, we investigate the impact of the time interval used for binning messages in order to produce the time series. In addition to the 15 seconds interval used earlier, we experiment with 30 seconds and 60 seconds. Note that this has a direct impact on the length of the time series: with a 15s interval, times series have 240 points per hour, versus 60 with a 60s interval. Experiments were run on the Count+Sentiment time series, which produced the best performance in Table 3.

Table 6 suggests that the performance of bcp is impacted favourably by running in a sliding window. Its performance increases slightly with larger time window. The three other

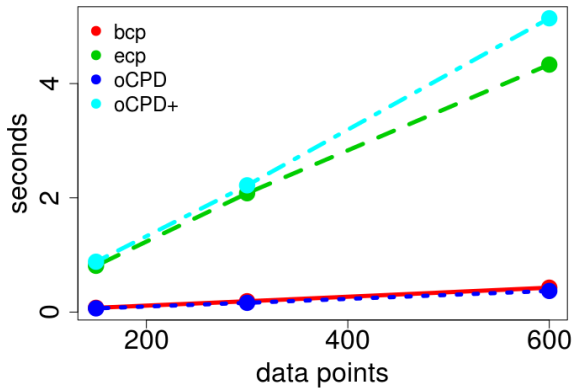


Figure 5: Average computational time of change point detection algorithms in on-line mode over 16 games, for 60s, 30s and 15s intervals (150, 300 and 600 data points).

algorithm suffer when the large time window is used, possibly because this reduces the length of the time series and limits the amount of statistics the underlying model can work with. On the other hand, bcp does better on larger time intervals, possibly because averaging scores over more messages makes the resulting data points more Gaussian (according to the central limit theorem). The performance of ecp improves with decreasing time interval. This may be due to more robust permutation tests when more data is available. ocpd+ performs best on the 30s and 15s time interval, and the best F-score of 48.1% is obtained at 30s. It also does better than ocpd (see also Tables 2, 4), which is consistent with the fact that the underlying predictive model is more flexible in ocpd+. All algorithm apart from bcp do somewhat worse with the sliding window than in off-line mode. This suggest that it is beneficial to run in true on-line mode, one data point at a time, as ocpd and ocpd+ are designed to do.

5.6. Number of Changepoints

Although changepoint detection algorithms try to guess the correct number of changes, they usually work better if the target number of changes is given. We evaluate this effect by running experiments in which we provide the correct number of reference changes. Results reported in the 3rd line of Table 3 should be compared to the average of Table 2 (also second line in Table 3). bcp and ecp clearly benefit from knowing how many changes to detect, and ecp now yields the best results, and wins over ocpd+ in the majority of cases (Table 4). It is understood, however, that this is an unrealistic scenario: in practice, we do not know the correct number of changes. In addition, ocpd and ocpd+provide a key functionality that the other two do not have: they process the data *online*, one point at a time, and can detect changes as soon as (or soon after) they occur instead of waiting for the entire collection to be acquired.

5.7. Computational Time

The computational time of online change point detection algorithms is another important factor for real-time CPD. We

use the same sliding windows as before (50 minute windows in steps of 25 minutes) to assess the computational time of the four CPD algorithms. The average computation time needed to analyze all 16 games is shown in Figure 5 for increasing time series lengths corresponding to decreasing time intervals. This shows that `ocpd` and `bcp` are much faster than `ocpd+` and `ecp`. This reflects the computational cost of running permutation tests repeatedly in `ecp`, although computational effort can be adjusted by lowering the number of permutations. For `ocpd+`, this reflects that the added flexibility in the linear model fitting the trend comes with the increase in computational cost. Note that despite theoretical upper bounds suggesting quadratic runtime for some of these algorithm, actual runtime seems to increase linearly.

6. Conclusion

We introduced a framework for detecting significant changes from on-line streams of messages. It relies on linguistic preprocessing producing semantically or linguistically relevant times series, which we run through a multivariate change point detection algorithm. The EuroGame16 collection was used to benchmark this approach, showing that we can detect around half the significant game plays in sports events, from the content of the twitter messages alone. The collection is made available to allow researchers to try other approaches to improve on our results. In addition, we contribute two variants of change point detection, `ocpd` and `ocpd+`. They show competitive performance with the state of the art `ecp` package. In addition, they can be used in a fully *on-line* mode, which allows the detection of changes soon after they occur instead of waiting until the entire time series can be processed. This is a key feature when monitoring social media streams in real time.

7. Bibliographical References

Adams, R. P. and MacKay, D. J. (2007). Bayesian online changepoint detection. arXiv:0710.3742.

Aslam, J. A., Ekstrand-Abueg, M., Pavlu, V., Diaz, F., McCreadie, R., and Sakai, T. (2014). TREC 2014 temporal summarization track overview. In Voorhees and Ellis (Voorhees and Ellis, 2014).

Atkinson, M., Piskorski, J., Tanev, H., and Zavarella, V. (2017). On the creation of a security-related event corpus. In *Proceedings of the Events and Stories in the News Workshop*, pages 59–65. Association for Computational Linguistics.

Barry, D. and Hartigan, J. (1993). A bayesian analysis for change point problems. *Journal of the American Statistical Association*, 35(3):309–319.

Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 113–120, New York, NY, USA. ACM.

Bruggermann, D., Hermey, Y., Orth, C., Schneider, D., Selzer, S., and Spanakis, G. (2016). Storyline detection and tracking using dynamic latent dirichlet allocation. In *Proceedings of 2nd Workshop on Computing News Storylines*, pages 9–19. Association for Computational Linguistics.

Erdman, C. and Emerson, J. (2007). bcp: An R package for performing a bayesian analysis of change point problems. *Journal of Statistical Software*, 23(1):1–13.

Goutte, C. and Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In D.E. Losada et al., editors, *Advances in Information Retrieval - 27th European Conference on IR Research*, pages 345–359.

Guralnik, V. and Srivastava, J. (1999). Event detection from time series data. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '99, pages 33–42, New York, NY, USA. ACM.

James, N. A. and Matteson, D. (2015). ecp: An R package for nonparametric multiple change point analysis of multivariate data. *Journal of Statistical Software*, 62(1):1–25.

James, N. A., Kejariwal, A., and Matteson, D. S. (2014). Leveraging cloud data to mitigate user experience from "breaking bad". *eprint arXiv:1411.7955*.

Kiritchenko, S., Zhu, X., and Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762.

Laban, P. and Hearst, M. (2017). newslens: building and visualizing long-ranging news stories. In *Proceedings of the Events and Stories in the News Workshop*, pages 1–9. Association for Computational Linguistics.

Székely, G. J. and Rizzo, M. L. (2005). Hierarchical clustering via joint between-within distances: Extending ward's minimum variance method. *Journal of Classification*, 22(2):151–183.

Ellen M. Voorhees et al., editors. (2014). *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*, volume Special Publication 500-308. National Institute of Standards and Technology (NIST).

Wang, Y. and Goutte, C. (2017). Detecting changes in twitter streams using temporal clusters of hashtags. In *Proceedings of the Events and Stories in the News Workshop*, pages 10–14. Association for Computational Linguistics.

Zhao, Y., Yao, F., Sun, H., and Yang, Z. (2014). BJUT at TREC 2014 temporal summarization track. In Voorhees and Ellis (Voorhees and Ellis, 2014).

A Deep Neural Network based Approach for Entity Extraction in Code-Mixed Indian Social Media Text

Deepak Gupta, Asif Ekbal, Pushpak Bhattacharyya
Indian Institute of Technology Patna, India
{deepak.pcs16, asif, pb}@iitp.ac.in

Abstract

The rise in accessibility of web to the mass has led to a spurt in the use of social media making it convenient and powerful way to express and exchange information in their own language(s). India, being enormously diversified country have more than 168 millions users on social media. This diversity is also reflected in their scripts where a majority of users often switch between their native languages to be more expressive. These linguistic variations make automatic entity extraction both a necessary and a challenging problem. In this paper, we report our work for entity extraction in a code-mixed environment. Our proposed approach is based on the popular deep neural network based Gated Recurrent Unit (GRU) archirecture that automatically discovers the higher level features from the text. We do not make use of any handcrafted features or rules, and therefore our proposed model is quite generic in nature. Our experiments on two benchmark datasets of English-Hindi and English-Tamil language pairs show the F-scores of 66.04% and 53.85%, respectively.

Keywords: Code-Mixing, Gated Recurrent Unit (GRU), Entity Extraction, Indian Language

1. Introduction

The phenomenal growth in user-generated contents on popular social media platforms like Twitter and Facebook has established a new perspective in Natural Language Processing (NLP). It incurs several challenges ranging from the incorrect spellings to the highly creative words ("f9" for "fine"), word lengthening ("gooooood" for "good"), phonetic typing, abbreviation ("IDK" for "I don't know") and so on. In recent times there has been a rising interest in building several applications on social media contents such as mood identification, opinion mining, etc. However, a majority of the existing research is restricted towards resource-rich languages such as English, some of the European languages and few Asian languages. The growing contents of underprivileged languages in the Web has necessitated investigating automated techniques that could build solutions involving these. Code-mixing refers to the mixing of more than one language that often makes the task more complex for building automated techniques. Non-native English speakers do not always use unicode encoding scheme to write in social media. Instead, they often use the most frequently used transliterated forms with the English words or phrases. These multilingual speakers often mix multiple languages in addition to anglicisms to be more expressive. There are 22 official languages mentioned in the Indian constitution. The report also reveals that 30 languages are spoken by more than one million native speakers, and 122 are spoken by more than $10,000$ people. Language diversity and dialect changes instigate frequent code-mixing in Indian languages. Hence, Indians are multilingual by adaptation and necessity, and frequently change and mix languages while writing in social media platforms. These pose additional difficulties in building automated tools for social media analytics. Named Entity Recognition (NER) is a primary task in information extraction. It aims at identifying the names of entities and classifying them into some predefined categories such as people, location, organization and product. This task can also be thought of as a two-step process, *viz.* entity detection and entity classification (Grishman and Sundheim, 1996). There are a significantly large body of works existing in Indian languages, but these are mostly related to the domains such as newswire. Nowadays, information extraction in micro-blogs has become an active research topic (Cano Basave et al., 2013), following the early experiments which showed this genre to be extremely challenging for the state-of-the-art algorithms (Derczynski et al., 2015; Bontcheva et al., 2014). The shortness of micro-blogs makes them hard to interpret. The social media text normally carries less discourse information per document, and threaded structure is fragmented across multiple documents. Apart from this, short text (tweets) also exhibits more language variations, tend to be less grammatical than the longer posts, contains unorthodox capitalization, and makes use of frequent abbreviations, hashtags and emoticons. These information also participates to decide the intention or meaning of a short text. To combat these problems, researcher has focused on microblog-specific information extraction algorithms, e.g. NER on Twitter data using Conditional Random Field (CRF) (Ritter et al., 2011) or hybrid methods (Van Erp et al., 2013). Particular attention is given to micro-text normalization (Han and Baldwin, 2011), as a way of removing some of the linguistic noise prior to Part-of-Speech (PoS) tagging and NER. Several Machine Learning (ML) techniques have already been applied for the NER tasks such as Hidden Markov Model(HMM) (Bikel et al., 1997), Maximum Entropy (Borthwick, 1999; Kumar and Bhattacharyya, 2006), Support vector Machine (SVM) (Isozaki and Kazawa, 2002), Conditional Random Field (CRF) (Li and McCallum, 2003) etc. Few systems for entity extraction in social media texts involving Indian language have been reported in FIRE-2015 workshop (Rao et al.,). In recent times, a benchmark setup for entity extraction involving Indian languages was introduced in FIRE-2016 shared

task (Rao and Devi, 2016). Some of the challenges for entity extraction in a code-mixed environment are as follows:

- The dataset contains tweets utterance in code mixed as well as in mono-lingual text.
- Introduction of a diverse set of entities, not limited to the only traditional set of entity types such as person, location, organization etc. There are 22 different types of entities that need to be extracted from text.
- Various resources and/or tools such as sentence splitter, tokenizer, PoS tagger, chunker etc. are not readily available in the required measure. In our current work, we develop a system for entity extraction using a deep Gated Recurrent Unit (GRU) architecture. We evaluate the proposed system for two language pairs, namely English-Hindi (EN-HI) and English-Tamil (EN-TA).

2. Methodology

In this section at first we define the code mixed entity extraction problem, and then present our proposed deep neural network based model.

2.1. Problem Definition

The problem of code-mixed entity extraction comprises of two sub-problems, *viz.* entity extraction and entity classification. Mathematically, the problem of code-mixed entity extraction can be described as follows: Let be S a code-mixed sentence having n tokens $t_1, t_2 \dots t_n$. A set of pre-defined entity category $C = \{C_1, C_2, \dots C_k\}$. The goal is to extract set of code mixed entities CME=$\{x \in S \mid x$ is a token or pharse$\}$. Thereafter, each of the code mixed entities $ce \in$ CME has to be classified into one of the pre-defined categories denoted by C. More formally the input of the system is a tweet associated with a user id and tweet id. The output of the system is all the entities with their corresponding categories present in the tweet. Some of the tweets from the dataset are shown in Table 1.

Langauage Pair	Sample Tweet
English-Hindi	awesome track ☺ listening to Toota Jo Kabhi Tara - A Flying Jatt - Atif Aslam; Sumedha Karmahe
	RiftWood Productions presents to you the season finale of Le' Bill & Giddy,La Muje'r
English-Tamil	@adhu_idu111 @7hillsm @beingsalmankhan @memephobhia @rajinifc salman sultan did 580cr. and for salman 200 crs cakewlk;
	IruMugan will be a Class + Mass movie like Thani Oruvan. The biggest plus is the screenplay - Thambi Ramaiyah..

Table 1: Some of the tweets from the dataset. The colored words represent the entities.

2.2. Approach for Entity Extraction

In order to recognize the entity from a code-mixed sentence, it is necessary to have a model which can process forward and backward tokens together in order to decide the type of the current token. Inspired by these works (Dernoncourt et al., 2017; Yao et al., 2013), we adapted a bi-directional GRU-based deep learning model for code mixed entity extraction which can use their memory to process arbitrary sequences of inputs in both the directions. Our proposed model is a layered architecture that follows the following steps:

1. Character and word level embedding layer
2. Input sequence processing layer
3. Output sequence optimization layer

We will begin by the first describing the bi-directional gated recurrent unit (GRU). Thereafter, each of the component layers is described.

2.2.1. Bi-directional gated recurrent units

Gated recurrent unit (GRU) was proposed by (Cho et al., 2014) to make each recurrent unit to adaptively capture dependencies at different time scales. GRU is very similar to the Long Short Term Memory (LSTM) model (Hochreiter and Schmidhuber, 1997). Similar to LSTM unit, GRU has gating units that modulate the flow of information inside the unit, however, without having separate memory cells. GRU has two neural gates, *update* and *reset*, that control the flow of information, as in LSTM. The update gate controls how much of past state should matter now. Similarly, reset gate allows the model to drop information that is irrelevant in future. Specifically, a GRU network successively reads the input token t_i, as well as the previous hidden state h_{i-1}, and generate the new memory content c_i and hidden unit h_i.

$$\begin{aligned} \mathbf{z}_i &= \sigma(\mathbf{W}_z t_i + \mathbf{V}_z \mathbf{h}_{i-1} + \mathbf{b}_z) \\ \mathbf{r}_i &= \sigma(\mathbf{W}_r t_i + \mathbf{V}_r \mathbf{h}_{i-1} + \mathbf{b}_r) \\ \mathbf{c}_i &= tanh(\mathbf{W} t_i + \mathbf{V}(\mathbf{r}_i \odot \mathbf{h}_{i-1}) + \mathbf{b}) \\ \mathbf{h_i} &= z_i \odot \mathbf{h}_{i-1} + (1 - \mathbf{z}_i) \odot \mathbf{c}_i \end{aligned} \tag{1}$$

where $\mathbf{z}$ and $\mathbf{r}$ are the input and reset gates, respectively. Here σ, $\odot$ represents the sigmoid function and element-wise multiplication, respectively. The $\{W_z, W_r, W, V_z, V_r, V\}$ and $\{b_z, b_r, b\}$ are the weight-matrices and bias vectors, respectively. Eq 1 can compute the forward an backward hidden state at time i by moving the forward or backward along the input sequence. We compute the forward hidden state $\overrightarrow{h_t}$ and backward hidden state $\overleftarrow{h_t}$. The final hidden state z_i computed by augmenting both the hidden states, i.e. $z_i = [\overrightarrow{h_i} \oplus \overleftarrow{h_i}]$ where $\oplus$ is the concatenation operator.

2.2.2. Character and word level embedding layer

The input to this layer is the word t at time i from the code mixed sentence S. It produces output as the word vector. The representation of the word in higher dimensional space is known as word embedding which can be obtained from the pre-trained embedding model trained on the huge unlabeled corpus using several methods, such as Glove (Pennington et al., 2014), word2vec (Mikolov et al., 2013) or fastText (Bojanowski et al., 2016). The word embedding is

Entity Types	No. of Entities (EN-HI / EN-TA)	Entity Type	No. of Entities (EN-HI / EN-TA)
COUNT	132 / 94	FACILITIES	10 / 23
PLANTS	1 / 3	PERSON	712 / 661
PERIOD	44 / 53	MATERIALS	24 / 28
LOCOMOTIVE	13 / 5	LOCATION	194 / 188
ENTERTAINMENT	810 / 260	YEAR	143 / 54
MONEY	25 / 66	DATE	33 / 14
TIME	22 / 18	ORGANIZATION	109 / 68
LIVTHINGS	7 / 16	QUANTITY	2 / 0
DISEASE	7 / 5	DAY	67 / 15
ARTIFACT	25 / 18	SDAY	23 / 6
MONTH	10 / 25	DISTANCE	0 / 4
Total no. of entity (EN-HI /EN-TA)		**2413 / 1624**	

Table 2: Statistics of the dataset for English-Hindi and English-Tamil language pairs

obtained through the Glove pre-trained word embedding[1]. It has the desirable properties in capturing the syntactic and semantic representation of words. This property helps the model to identify the similar entities. However, it still suffers from several issues such as, they cannot handle out-of-vocabulary words, misspelled words and variations in noun or verb phrase. When comes to the code mixing text this issue even becomes more crucial to resolve. One way to resolve this issue is by utilizing character based word embedding which incorporates each individual character of a token to generate its vector representation. This approach, therefore, allows the model to learn lexical patterns (e.g. suffix or prefix) which eventually can help in capturing out-of-vocabulary words and some other information which are difficult to capture through word embedding.

Now we define the way to compute the character embedding. Let $c_1^i, \dots c_l^i$ be the character sequence of word t_i having length l. We initialize the character representation $V(c_k^i)$ randomly of each character k in token t_i. By this way, we generate the sequence of character representations $c_{1:k}$. The generated sequence is provided as an input to the Bi-GRU model, which outputs the character word embedding $C(i)$. Finally, the output of word embedding for the i_{th} word is the concatenation of word embedding $W(i)$ and the character word embedding $C(i)$. In short, the layer takes the sequence of words $t_{1:n}$ as input and produces a sequence of word embeddings $e_{1:n}$ as output.

2.2.3. Input sequence processing layer

This layer takes the previous layer's output i.e. the sequence of word embedding $e_{1:n}$ as the input and passes this sequence to a Bi-GRU unit. Bi-GRU unit as discussed in Section-2.2.1. generates the output states h_i for the i^{th} element of the input sequence $e_{1:n}$. Thereafter, each h_i of the Bi-GRU unit is given to a feed-forward neural network. This network computes the vectors of probability score p_i. The length of the probability vector $len(p_i)$ = $(2 \times$ number of possible named entity category + $1)$. This length exhibits the BIO (B-beginning, I-intermediate and O-outside) encoding of the entity classes.

2.2.4. Output sequence optimization layer

This layer takes the output (sequence of probability vector) of the previous layer as input and produces the label sequence as the output. However, the label sequence $C_{1:n}$ can be obtained for C_i as follows: $C_i = argmax_k(p_i[k])$. This greedy approach costs the performance of a model where the final output is the sequence of labels. This strategy does not account the dependencies between the subsequent labels. However, these dependencies capture at certain level in the input sequencing layer. But, we can still allow the model to directly learn these dependencies in the last layer of the model. To achieve this transition probability two subsequent labels are utilized. The final score of a label sequence can be defined as follows:

$$Score(C_{1:n}) = \sum_{k=2}^{n} M[C_{k-1}][C_k] + \sum_{k=1}^{n} p_i[C_k] \quad (2)$$

Where M is 2-D matrix that contains the transition probabilities between the two subsequent labels. While training, model maximizes the log-likelihood probability of the gold label sequence. In the testing phase, a sequence of predicted labels which maximize the score is chosen as the final label sequence.

3. Dataset and Experimental Setup

In this section, we report the datasets used in the experiments and experimental setup.

3.1. Dataset

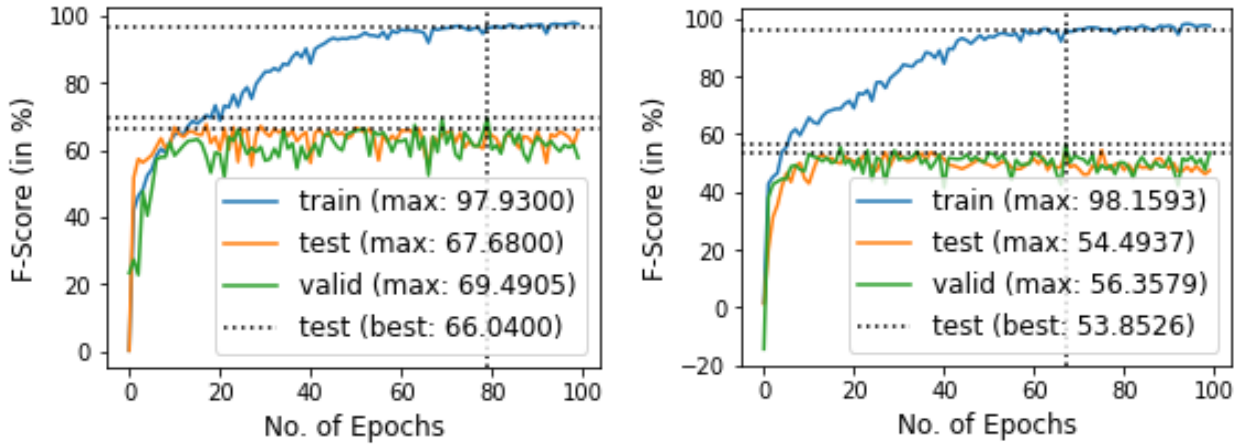
In order to evaluate the system performance, we use two language pairs: English-Hindi (EN-HI) and English-Tamil (EN-TA), which have been made available through the CMEE shared task (Rao and Devi, 2016). Datasets are crawled from the tweeter. The tweets are mixed in nature containing English-Hindi and English-Tamil scripts. The dataset contains a total of 22 different type of entities. Majority of entities are from 'Entertainment', 'Person' 'Location' and 'Organization' categories. A brief statistics of the dataset is provided in Table-2. English-Hindi tweet dataset contains a total of $2,700$ tweets from $2,699$ tweeter users. Similarly, English-Tamil tweet dataset contains a total of $2,183$ tweets from $1,866$ tweeter users. As some of the tweets from EN-TA language pair dataset are in Tamil script, in order to keep everything in English we transliterate[2] those tweets into English. We evaluate the system performance using the standard metrics, precision, recall and F-score.

[1] http://nlp.stanford.edu/data/glove.840B.300d.zip

[2] https://github.com/deepak1357/indic-trans

Models	English-Hindi			English-Tamil		
	Precision	Recall	F-Score	Precision	Recall	F-Score
SVM	72.25	50.12	59.18	69.38	36.12	47.50
MEMM	73.89	51.07	60.39	68.75	37.61	48.62
CRF	75.14	52.29	61.66	69.27	38.02	49.09
Our model	74.29	59.44	66.04	67.93	44.15	53.85

Table 3: Performance comparison of our proposed model with ML based baseline classifier.



(a) English-Hindi

(b) English-Tamil

Figure 1: Validation curve for both pair of dataset

3.2. Experimental Setup

We perform 5-fold cross validation on the both language pair datasets as described in Section 3.1.

We optimize the network by setting the following hyper-parameters through 5-fold cross-validation experiments: word embedding dimension=300, character embedding dimension=50, character-based word embedding GRU dimension=50, input sequence processing GRU dimension: 100, dropout probability: 0.5, dropout probability=0.3, batch size=50, # epochs=100.The network is trained using Adam optimization algorithm (Kingma and Ba, 2014) by updating all the parameters. This reflects the generalization of our optimum hyper-parameter selection over two completely different language pair datasets.

4. Result and Analysis

For comparison, we design three strong baselines based on traditional supervised techniques. Our baseline systems are based on Support vector machine (SVM) [3], Maximum entropy markov model [4] and Conditional random field [5] classifiers. In order to train the classifier the feature set described in (Gupta et al., 2016) is extracted. We evaluate the performance of all three baselines using 5-fold cross validation to be consistent with our proposed neural network.

Table-3 reports the performance of our proposed approach with all the three baselines for both the language pair datasets. We also plot the validation curve for both the language pair datasets. Fig 1 shows the validation curve between F-score and no. of epochs. The model achieves the best F-score value of 66.04% in 79 epochs for EN-HI (i.e. English-Hindi). Similarly, we obtain the best F-score value of 53.85% for EN-TA language pair dataset with 67 epochs. There could be two possible reasons behind the low F-score values on EN-TA dataset :

1. less instances of entities in the dataset as compared to EN-HI.
2. lower hit ratio (availability of word) of EN-TA word (37.46%) as compared to EN-HI word (55.70%) in word embedding matrix.

We further improve the system performance by using the word embedding trained on an appropriate code-mixing dataset. Our proposed approach outperforms all the baselines for both the language pairs. We observe that we achieve significant gains in terms of recall as compared to the baselines. This may be attributed to the fact our proposed network learns the hidden and useful features from a text, which handcrafted features can not infer always.

The best-performing system (Bhat et al., 2016) in the shared task reported the F-score value of 68.24% in English-Hindi language pair. They have used several hand-crafted features to train the feed-forward neural network. For English-Tamil language pair the best- performing system (Gupta et al., 2016) reported the F-score value of 44.12 using the handcrafted features with CRF as the underlying classifier. The evaluation shows that our

[3] http://chasen.org/ taku/software/yamcha/

[4] https://nlp.stanford.edu/software/classifier.html

[5] https://taku910.github.io/crfpp/

proposed system achieves higher performance compared to these systems. However, it is to be noted that our reported results are on cross-validation and we have not been able to perform experiments on the test data as this is not publicly available. Hence, it will not be fair to compare these existing systems with our proposed system.
Our obtained results are reliable as we perform experiments on cross-validation Significance test shows that the improvement over all the baselines are statistically significant as (p-*value* < 0.044).

4.1. Error Analysis

In order to get an idea about the errors, we perform in-depth analysis of the outputs of the systems. We have segregated our prediction inaccuracies based on its genre and have mentioned some of the observations below:
(1) Incorrect Entity: This error is caused when the predicted entity is misclassified into the incorrect entity type. This appears mainly due to the irregular casing of multiple words in a tweet. One possible reason for incorrect classification of tags also includes words that convey more than one semantic information.
(2) Missed Entity: This error is generated when the system fails to predict the entity tag and mis-classified it into other-than-NE. The possible causes for this error include:

- **Contextual information:** Words appearing at the start and end of tweets are easily recognized as NE, but the system fails to predict many entities present in the middle of a tweet.
- **Presence of abbreviated words:** Due to the presence of a large number of abbreviations, many words were left un-categorized by our system. Words like *SRK* and *FB* were often confused with *"Shah Rukh Khan"* and *Facebook*, leading to incorrect predictions by our system.

5. Conclusion and Future Works

In this paper, we have described an approach for entity extraction from the code-mixed tweets. We have proposed a deep learning model utilizing bi-directional GRU architecture. The proposed deep learning architectures discover the hidden features from the tweets automatically to categorize them into one of the pre-defined classes. The system has been evaluated for two language pairs, namely English-Hindi and English-Tamil. Experimental results show that our system achieves encouraging performance for both the language pairs. Empirical evaluation shows that our proposed system performs better as compared to the feature-based supervised model. In future, we will investigate the usages of shared representation of code mixed words in different languages.

Acknowledgements

Asif Ekbal greatfully acknowledges the Young Faculty Research Fellowship (YFRF) Award, supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

6. Bibliographical References

Bhat, I. A., Shrivastava, M., and Bhat, R. A. (2016). Code mixed entity extraction in indian languages using neural networks. In *FIRE (Working Notes)*, pages 296–297.

Bikel, D. M., Miller, S., Schwartz, R., and Weischedel, R. (1997). Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing*, pages 194–201. Association for Computational Linguistics.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Bontcheva, K., Derczynski, L., and Roberts, I. (2014). Crowdsourcing named entity recognition and entity linking corpora. *The Handbook of Linguistic Annotation (to appear)*.

Borthwick, A. (1999). *A maximum entropy approach to named entity recognition*. Ph.D. thesis, Citeseer.

Cano Basave, A. E., Varga, A., Rowe, M., Stankovic, M., and Dadzie, A.-S. (2013). Making sense of microposts (# msm2013) concept extraction challenge.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October. Association for Computational Linguistics.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., and Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49.

Dernoncourt, F., Lee, J. Y., Uzuner, O., and Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606.

Grishman, R. and Sundheim, B. (1996). Message understanding conference-6: A brief history. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gupta, D., Tripathi, S., Ekbal, A., and Bhattacharyya, P. (2016). A hybrid approach for entity extraction in code-mixed social media data. *MONEY*, 25:66.

Han, B. and Baldwin, T. (2011). Lexical normalisation of short text messages: Makn sens a# twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 368–378. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Isozaki, H. and Kazawa, H. (2002). Efficient support vector classifiers for named entity recognition. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kumar, N. and Bhattacharyya, P. (2006). Named entity recognition in hindi using memm. *Techbical Report, IIT Mumbai*.

Li, W. and McCallum, A. (2003). Rapid development of hindi named entity recognition using conditional random fields and feature induction. *ACM Transactions on Asian Language Information Processing (TALIP)*, 2(3):290–294.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43.

Rao, P. R. and Devi, S. L. (2016). Cmee-il: Code mix entity extraction in indian languages from social media text@ fire 2016-an overview. In *FIRE (Working Notes)*, pages 289–295.

Rao, P. R., Malarkodi, C., and Devi, S. L.). Esm-il: Entity extraction from social media text for indian languages@ fire 2015–an overview.

Ritter, A., Clark, S., Etzioni, O., et al. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534. Association for Computational Linguistics.

Van Erp, M., Rizzo, G., and Troncy, R. (2013). Learning with the web: Spotting named entities on the intersection of nerd and machine learning. In *# MSM*, pages 27–30.

Yao, K., Zweig, G., Hwang, M.-Y., Shi, Y., and Yu, D. (2013). Recurrent neural networks for language understanding. In *Interspeech*, pages 2524–2528.

7. Language Resource References

Rao, P. R. and Devi, S. L. (2016). Cmee-il: Code mix entity extraction in indian languages from social media text@ fire 2016-an overview. In *FIRE (Working Notes)*, pages 289–295.

PoSTWITA-UD: an Italian Twitter Treebank in Universal Dependencies

Manuela Sanguinetti[⋆], Cristina Bosco[⋆], Alberto Lavelli[•], Alessandro Mazzei[⋆], Oronzo Antonelli[‡], Fabio Tamburini[◇]

[⋆]Department of Computer Science, University of Turin, Italy

[•]FBK-ict, Trento, Italy

[‡]DISI / [◇]FICLIT, University of Bologna, Italy

{msanguin, bosco, mazzei}@di.unito.it, lavelli@fbk.eu, antonelli.oronzo@gmail.com, fabio.tamburini@unibo.it

Abstract

Due to the spread of social media-based applications and the challenges posed by the treatment of social media texts in NLP tools, tailored approaches and *ad hoc* resources are required to provide the proper coverage of specific linguistic phenomena. Various attempts to produce this kind of specialized resources and tools are described in literature. However, most of these attempts mainly focus on PoS-tagged corpora and only a few of them deal with syntactic annotation. This is particularly true for the Italian language, for which such a resource is currently missing. We thus propose the development of PoSTWITA-UD, a collection of tweets annotated according to a well-known dependency-based annotation format: the Universal Dependencies. The goal of this work is manifold, and it mainly consists in creating a resource that, especially for Italian, can be exploited for the training of NLP systems so as to enhance their performance on social media texts. In this paper we focus on the current state of the resource.

Keywords: social media language, Twitter, Italian, Universal Dependencies

1. Introduction

The increasing reliance on the popularity of the Internet and social media in every-day life has led, among other things, to the proliferation of the so-called user-generated contents. As one of the most popular social media, Twitter is among the main providers of this type of contents, the usage of which in scientific research ranges from data analysis, sentiment analysis and opinion mining, to language technologies. Often, though, Twitter user-generated contents are not edited and/or revised for grammatical accuracy (Lynn et al., 2015). Furthermore, the limited number of characters for each tweet can stimulate creativity and encourage an innovative and non-standard usage of language conventions. As regards NLP, dealing with this kind of linguistic data presents a series of challenges, which are reflected in the lower output quality of various automatic tools and at different linguistic levels (see, e.g., Gimpel et al.(2011), Foster et al. (2011) and Ritter et al. (2011)).

These considerations highlight the need for properly annotated resources to provide adequate coverage of such a phenomenon. This is especially true considering the (relatively) little progress made in this field: for the Italian language in particular, at the time of writing, the only linguistically-annotated social media corpora we are aware of are those of Bosco et al. (2016) (i.e. the PoSTWITA corpus) and Rei et al. (2016), both annotated at PoS level only.

The contribution of the work hereby presented aims at filling this gap, by creating a treebank of Italian non-canonical texts retrieved from Twitter. The treebank has been built using as starting data the PoSTWITA corpus (mentioned above), and it has been syntactically annotated in compliance with the Universal Dependencies format.

Several goals motivate this work, among these: *a*) to provide a resource that can be used for parser training on standard (Bosco et al., 2008; Bosco et al., 2010; Bosco and Mazzei, 2013) and non-standard texts (whose results, in turn, can be exploited in sentiment analysis applications[1]), as well as for systematic linguistic analysis related to social media language (similar to what proposed in Hu et al. (2013)); *b*) in a long-term perspective, to encourage the creation of similar resources in languages other than Italian, supported by the availability of a shared representation format, namely the Universal Dependencies (possibly extended to cover social media linguistic phenomena).

As for the second point, that is the choice of the annotation format, Universal Dependencies is a recent project that has gained broad consensus over the last few years, becoming the reference framework for dependency annotation. Despite the critical points raised, for example, on some annotation choices (Gerdes and Kahane, 2016), or on the cross-resource consistency problem (de Marneffe et al., 2017), an ever increasing number of languages and resources have been (and are about to be) made available in this format; also two CoNLL Shared Task have been organized in 2017 and 2018, using UD treebanks as datasets. This highlights the need for a widely recognized standard to refer to, either in the process of creating a resource (from scratch or by conversion) or in the evaluation of NLP tools whose training is based on such a resource. Finally, one more factor that made us lean towards using the UD format is the possibility to extend the basic labels with subtypes that are useful for the representation of specific phenomena and are based both on the language at issue and, as in our case, on the peculiar linguistic features of the text type.

The paper focuses on the current state of the resource and is organized as follows. A brief survey is presented about related work in the next section. Section 3 describes the resource considering the conversion steps into UD format.

[1]In this sense, the creation of this resource is linked to the projects coordinated by the Computer Science Department of the University of Turin for the creation of automatic systems for online hate speech identification, in particular on Twitter. See: `http://hatespeech.di.unito.it/`

Section 4. focuses on the syntactic layer in particular, showing the main phenomena we dealt with for what concerns manual annotation. Finally, Section 5. describes the experiments we carried out by training and testing state-of-the-art parsers on the novel resource, while Section 6. closes the paper with few remarks on how we intend to follow up on this work.

2. Related Work

Social media texts fall under the broader language variety often referred to as non-canonical, or non-standard, language; its automatic processing and analysis is challenged namely by all those linguistic phenomena that deviate from what is conventionally conceived as the "norm", i.e. the standard language. In this section, we mention some of the attempts made in other related resources to tackle these challenges, especially as regards syntactic annotation.

Tweebank (Kong et al., 2014) is a corpus that presents a simplified, though linguistically-grounded, dependency-based scheme; the resource consists of unlabeled dependency graphs that allow multiple roots in case a tweet contains more than one utterance, and where just nodes with a syntactic function are explicitly selected.

Another attempt to properly annotate Web data was made in the English Web Treebank (Silveira et al., 2014), a collection of more than 16k sentences taken from various media, also available in UD format. In this resource, the treatment of Internet-related phenomena mainly entailed the revision of the inventory of dependency relations; in particular, new labels were introduced, that since then became an integral part of the UD scheme[2].

Other examples of non-canonical texts annotated in compliance with UD specifications are the Treebank of Learner English (Berzak et al., 2016) and the Singlish treebank (Wang et al., 2017). The first one is a collection of English as a Second Language (ESL) sentences, which thus contains a large number of non-standard syntactic structures due to grammatical errors made by the non-native English speakers. As regards their annotation, the main guiding principle prescribes to follow the *literal meaning*, emphasizing a syntactic analysis that is more faithful to the observed language usage. This is reflected, for example, in the annotation of a direct object as a non-core predicate dependent, if (wrongly) preceded by a preposition, or conversely, a non-core dependent annotated as predicate argument because of an elided preposition. The second example of UD format applied to non-standard texts is the one presented in Wang et al. (2017) and regarding the syntactically-annotated resource of Colloquial Singapore English (or Singlish)[3], an English-based creole language, frequently used in written forms of Web media. Most of the problems encountered in the annotation of such texts had to do with the treatment of terms and expressions imported from local languages, whose annotation is mainly based on the conventions of such languages rather than English, as well as on topic-prominence phenomena, copula and NP deletions, and inversions, all linguistic constructions that eventually have been modeled successfully with UD representation.

To conclude, we mention here the work presented by Martinéz Alonso et al. (2016), on the creation of a French corpus of user-generated (UGC) content with automatic PoS tagging and an experimental syntactic annotation (on a smaller sub-section of the corpus) using UD. Besides confirming, once again, the challenges posed to the treatment of UGC-related phenomena, the study also brings to light some critical points of UD format and specifications when it comes to deal with such issues, with an eye in particular on the tokenization, and the consequent syntactic annotation, of non-standard conflated tokens, as well as elliptical structures and disfluencies resulting from the time and space limitations posed by the medium used.

In Section 4., we describe our approach to such phenomena.

3. Introducing PoSTWITA-UD: Conversion and Current Dataset

PoSTWITA-UD has been created by enriching the dataset used for the EVALITA 2016 task of Part-of-Speech tagging of Social Media (Bosco et al., 2016). The original corpus consists of 6,438 tweets in the development set (114,967 tokens) and 300 tweets in the test set (4,759 tokens), annotated at PoS level only. The format of the resource, also shown in Figure 1, appears as a two-column text file with tweets identified by their IDs (in the header) and separated by blank lines; each word in the tweet has its own line, which in turn contains two tab-separated fields, for the word form and its Part of Speech respectively.

```
_____579013335921885184_____
@LudovicaCagnino    MENTION
Grazie  INTJ
AMORE   NOUN
```

Figure 1: Example of PoSTWITA original format.

The corpus was automatically tokenized with an adapted version of the Tweet-NLP tokenizer(Gimpel et al., 2011), PoS-tagged with the TnT tagger (Brants, 2000) trained on the UD_Italian treebank v1.3 (Bosco et al., 2013), and then manually corrected.

The whole process of conversion into UD and its annotation has been described in Sanguinetti et al. (2017), but for the sake of clarity, we summarize here the main steps we followed in order to get a fully UD-compliant resource.

Tokenization: no particular changes have been made in this sense from PoSTWITA to PoSTWITA-UD, except for preposition-article and verb-clitic contractions, that were left as single tokens in PoSTWITA and then splitted into the corresponding syntactic words during conversion into UD. All other tokenization choices remained unchanged in PoSTWITA-UD; this also entailed the occurrence of cases where multiple tokens were kept as a single one, whether mistakenly or on purpose, i.e. either because some typo occurred (e.g. "*anchio*" instead of "*anch'io*", "me too") or in

[2]These are `discourse`, `goeswith`, `list` and `vocative`.

[3]The resource is not available in the official UD repository, but here: `https://github.com/wanghm92/Sing_Par/tree/ud_tf0.12/Singlish/treebank`

	PoSTWITA	PoSTWITA-UD v2.1	PoSTWITA-UD v2.2
Annotation Layers	Part of Speech	Lemma Language-specific Tag (xpos) Morphological Features Syntactic Relation	
# of tweets	6,738	3,510	6,712
# of (syntactic) words	119,726	64,536	124,410

Table 1: Treebank basic statistics and differences between the original PoSTWITA corpus, used in the EVALITA campaign in 2016, and the converted versions in UD format: the one released in November 2017 (v2.1), containing just the first half of the corrected dataset, and the v2.2 that finally comprises the complete dataset. The overall number of tweets in PoSTWITA-UD v2.2 differs from the one in the original PoSTWITA due to the removal of duplicate tweets, while the change in the number of syntactic words also depends on the tokenization steps carried out during conversion (see Section 3.).

case of abbreviations of two or more words (e.g. "TT", that stands for "trend topic"), or even because of expressive intents (e.g. "*éStataPremiataUna*", "itHasBeenAwardedA"). Such cases, however, are not particularly frequent in our corpus, neither systematic. Considering that UD do not force the splitting of such conflated tokens, we decided to keep them unchanged (see Section 4.2. for their syntactic treatment).

PoS tagging: the original PoSTWITA corpus already contained UD PoS tags; however the standard UD tagset was extended by adding *a*) two labels for the contracted forms mentioned above (i.e. `ADP_A` and `VERB_CLIT` respectively), *b*) a number of other new labels for non-standard elements typically found in tweets, such as URLs (`URL`), email addresses (`EMAIL`), pictograms (`EMO`), hashtags (`HASHTAG`) and mentions (`MENTION`).
In PoSTWITA-UD, `ADP_A` and `VERB_CLIT` were completely removed, because of the splitting of such contractions, while the other Internet-specific tags all conflated into `SYM`, the tag used for symbols. Any other unconventional token whose tagging was not possible for some reason was assigned a `X` tag.

Lemmatization and morphological analysis: lemmas and mophological features were retrieved using AnIta (Tamburini and Melandri, 2012). However, as expected, the corpus also contains a whole host of non-standard word forms that were not recognized by the lemmatizer. In the spirit of leaving the texts as much intact as possible, we decided not to normalize such forms, which still appear in the resource as they do in the original tweet. On the other hand, in case of abbreviations (*ke*⇒ *che*, 'that'), word lengthening (*pizzaaaaaaa*⇒pizza), capitalization (*GOVERNO*⇒*governo*, 'government'), minor typos and grammatical errors (*anno* instead of *hanno*, 3rd pers.plur of *avere*, 'to have'), we manually inserted the lemma of their standard counterpart; for other out-of-vocabulary words, such as dialectal and foreign terms or unintelligible forms, the lemma remained the same as the word form. Note also that for abbreviations of multiple words we kept the abbreviation in the lemma field as well.

Syntactic analysis: this step has been performed first automatically, by training three parsers on Italian standard texts, namely those included in UD_Italian v2. The tools used were the graph-based (Bohnet, 2010) and transition-based (Bohnet and Nivre, 2012; Bohnet and Kuhn, 2012) MATE parsers, and RBG (Lei et al., 2014; Zhang et al., 2014b; Zhang et al., 2014a). The parsing step was performed on the entire resource and relied on the previously-annotated layers.
A first set of 300 tweets was then revised by two independent annotators in order to calculate the inter-annotator agreement (a Cohen's k = 0.92) and to test the parsers results.
Finally, the output that gained the best results (i.e. the one from the transition-based MATE parser) was chosen as the final version, and the two annotators completely revised it.
The first part (about 3,500 tweets) of the manually-corrected corpus was made available in November 2017, as part of the v2.1 release of Universal Dependencies. In view of the 2.2 version, whose official release is scheduled for April 2018, and of the upcomping CoNLL shared task, the second half as well is expected to appear in the resource, thus completing the whole annotation process.
To conclude this introductory section, Table 1 summarizes the treebank basic statistics, also highlighting the differences among the various versions of the resource.
The next section describes the guiding principles we adopted to annotate the treebank.

4. A UD-based Analysis of Italian Tweets: Updated Guidelines

As already described in Sanguinetti et al. (2017), the first tentative guidelines were drafted in parallel with an initial annotation experiment of 300 tweets. Since then, those guidelines have been further revised and integrated, in order to cover a wider range of UGC-related and other tricky phenomena.
The main guiding principle followed during the whole annotation process provides that what is understandable by a human should be annotated accordingly: this means that even in the presence of non-canonical tokens or structures, whenever the annotator is able to grasp their meaning with a certain degree of confidence, he/she is expected to encode it properly, according to such interpretation. On the other hand, while this served as a general guideline, more peculiar issues have also been encountered, the treatment of which required specific solving strategies. We grouped such issues into the following categories:

- meta-language tokens

- non-standard word forms
- juxtaposition and sentence linking
- elliptical structures

and we attempted to deal with them in a more systematic way.

4.1. Meta-Language Tokens

We include in this category the non-conventional textual elements typical of Twitter (as well as other social platforms), especially hashtags, mentions, pictograms, RTs (i.e. the tokens that usually precede a retweet), URLs and similar. These elements, if not inherent to the syntactic structure of the sentence, have been annotated using different criteria, depending on their type and their distinguishing features, also by introducing new specific label extensions.
More specifically, hashtags are considered as juxtaposed elements, and thus annotated using the `parataxis:hashtag` relation; mentions are annotated with the `vocative:mention` label while emoticons and emojis are treated as discourse markers, therefore they all bear the `discourse:emo` relation. Finally, the `dep` relation has been systematically used with URLs that are just appended at the end of a tweet, as well as with RT tokens. Figure 2 shows an example for each case:

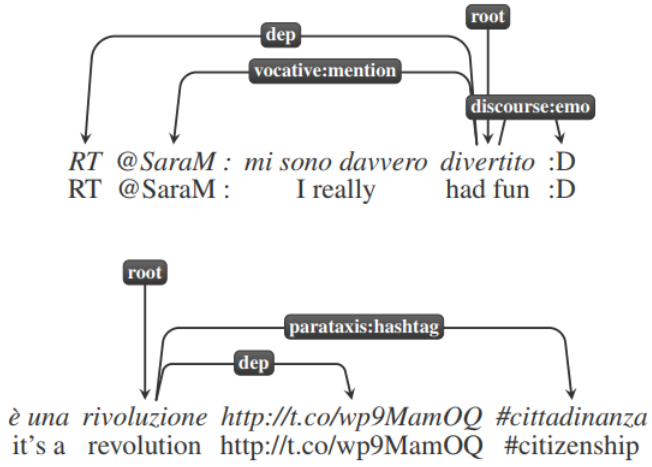


Figure 2: Examples of relations for meta-language tokens in PoSTWTITA-UD.

On the other hand, if syntactically integrated within the sentence, these same elements are annotated taking into account their actual syntactic role. In the sentence:

@AttilioFontains ha risposto al mio messaggio
(@AttilioFontains replied to my message)

the mention is considered as the subject of the sentence.

4.2. Non-Standard Word Forms

This category includes a wide range of different examples of various nature, from foreign and dialectal terms to mistakenly conflated forms (*cosè*, instead of *cos'è*, 'what is'), but also truncated words (due to space constraints) and completely unintelligible forms.
As also stated in Section 3., such forms were all PoS-tagged as `X` elements; still, the identification of their syntactic role remained quite unclear. A further distinction has thus been made, and each distinct case has been treated differently.

Code switching: if a single foreign or dialectal word occurs within the sentences, it is considered for its actual syntactic role, as also prescribed in similar cases for meta-language tokens (Section 4.1.); however, a phrase involving more than one token is considered as a flat structure[4] specified by the `:foreign` subtype.

Conflations: conflated forms are treated by assigning the syntactic relation associated with the word in the token that is promoted as head. In the sentence:

Fedenon ha proferito parola
(Fede-did-not say a word)

the token *Fedenon* is the concatenation of the proper noun *Fede* (abbreviation of *Federica*), which in this context is the subject of the verb *proferito* ('said')), and *non* ('not'), an adverbial modifier. Being a core argument, the proper noun is thus promoted as the head word and the whole token is annotated as `nsbuj`.

Truncated and unknown words: we also found a number of cases where the last word in the tweet was cut off, because of the character limits posed to tweets; even in these cases, we annotate the word according to its supposed syntactic role, whenever possible; if not, and, more generally, in case of unintelligible word forms, a `dep` relation is used.

4.3. Juxtaposition and Sentence Linking

From a syntactic point of view, UGC is also characterized by an abundance of paratactic and juxtaposed sentences; this is particularly true for Twitter posts, that often consist of more than one sentence, as below:

#michelebravi un nome una garanzia? Vediamo. Anzi sentiamo
(#michelebravi a name a guarantee? We'll see. Or rather, we'll hear)

Note also that, given the intended use of the resource as training set for NLP tools in a real-world setting, the main segmentation unit in PoSTWITA-UD is the complete tweet, rather than the single sentence. However, establishing a dependency link among such sentences is not a trivial task, especially considering the single-root constraint posed by UD scheme. Therefore, we systematically resorted to `parataxis` even to represent such inter-sentential links, although aware of the theoretical limits and the risk of an "overuse" of the label. The tweet above would thus be annotated as shown in Figure 3:
Furthermore, besides extending, in a sense, the applicability of this label to such a wider range of cases, we have also

[4] See `http://universaldependencies.org/u/dep/flat.html`

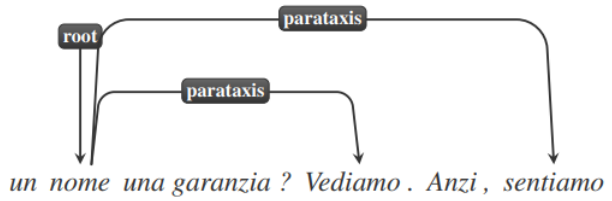


Figure 3: Example of `parataxis` used as inter-sentential link.

introduced a further distinction in its use with respect to other ones. More specifically, we have included new label extensions for five cases in particular (in addition to the one mentioned above for hashtags), even by partly mutuating the dependency labels from already existing treebanks[5]. Such cases comprise the dependency link used to identify appositive sentences (`parataxis:appos`), the semantically-void clauses used as discourse markers (`parataxis:discourse`), the parenthetical clauses that cannot be considered independent from the governing predicate (`parataxis:insert`), and the paratactic sentences having an implicit argumental role with respect to the governing predicate (`parataxis:nsubj` and `parataxis:obj`). All these relations are shown with practical examples in Figure 4.

4.4. Elliptical Structures

The need for immediacy in computer-mediated communication, its interactive nature, as well as the space constraints posed by the medium used, often result in a fragmentary writing (Martínez-Alonso et al., 2016) and very concise utterances, where more or less meaningful portions of a sentence are omitted[6]. As a consequence, and in a way that recalls the so-called *headlinese*, a given sentence may have function words removed, such as:

Manovra Governo Monti
((The) budget measures (of) Monti administration)

Given the preference of UD scheme in assigning headedness to content words, no solving strategy has been necessary for such cases. Even when copulas in copulative sentences were omitted, the main constituents of the sentence (i.e. the nominal predicate and its subject, if present) preserved their function, and the missing copula has been simply ignored.
However, elliptical structures can also reach a higher degree of complexity; in such cases, or at least whenever possible, we followed the main guideline, by attempting to interpret the missing context and to annotate the tweet accordingly. For example, in the sentence:

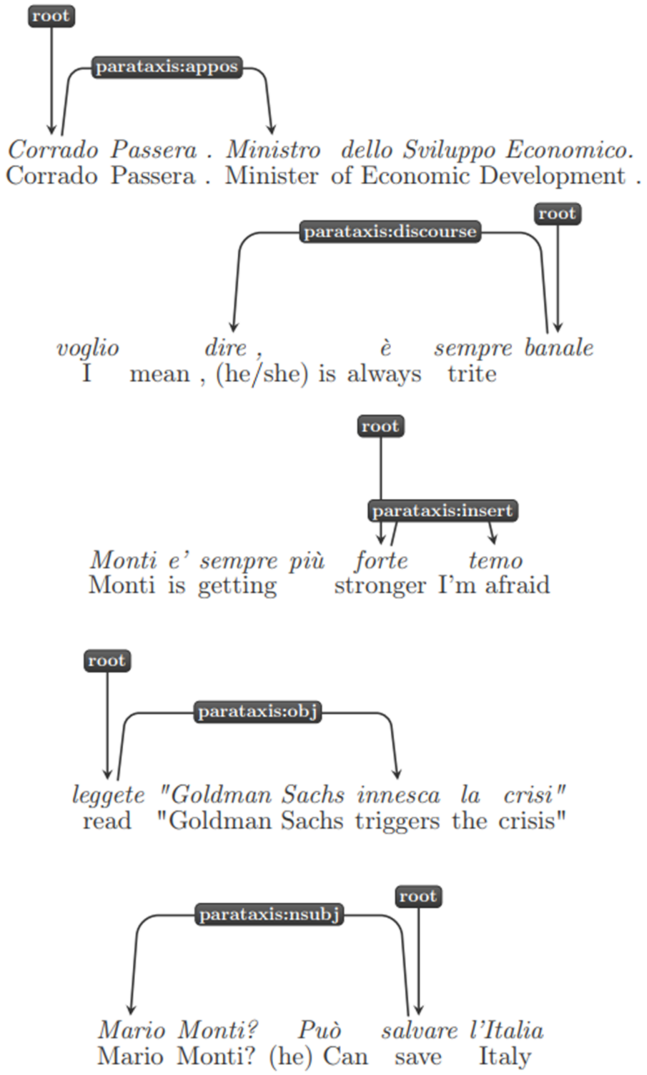


Figure 4: Examples of newly-introduced label extensions for paratactic structures in PoSTWTITA-UD.

Innalzata età minima donne 62 e uomini 66 dal 2018
(Raised (the) retirement age (of) women (to) 62 and (of) men (to) 66 (starting) from 2018)

besides the function words removal, a complex predicate ellipsis also occurs, which requires the special relation `orphan`, applied as shown in Figure 5.
In case the degree of uncertainty is such that it made the interpretation effort completely pointless, we rather linked the disconnected fragments, again, either with a `parataxis` or with an even more generic `dep` relation, depending on the tweet context.

5. Some Parsing Experiments

In order to test the effectiveness of the proposed annotations and to gather information about the difficulties in parsing tweet texts, we carried out some parsing experiments using state-of-the-art parsers available to the community. We

[5]Namely the UD_French-Spoken (Gerdes and Kahane, 2017) and UD_Slovenian-SST (Dobrovoljc and Nivre, 2016)

[6]In this respect, it is also similar to sign languages, where the lack of determiners and prepositions can be compensated with a sort of visual/spatial organization of the sentence (Mazzei et al., 2013).

Parser	UD_It		UD_PoSTW		UD_It+PoSTW	
	UAS	LAS	UAS	LAS	UAS	LAS
(Chen and Manning, 2014)	72.25%	63.98%	81.75%	76.34%	82.94%	77.60%
(Ballesteros et al., 2015)	73.98%	65.71%	84.28%	78.97%	85.21%	79.93%
(Kiperwasser and Goldberg, 2016): Transition	77.17%	68.12%	77.46%	68.95%	80.79%	73.36%
(Kiperwasser and Goldberg, 2016): Graph	75.96%	67.54%	79.49%	71.19%	81.43%	73.49%
(Andor et al., 2016)	65.72%	52.31%	77.88%	67.11%	79.52%	69.04%
(Cheng et al., 2016)	76.94%	67.54%	86.12%	79.89%	86.85%	80.93%
(Dozat and Manning, 2017)	**77.48%**	**68.22%**	**86.38%**	**80.53%**	**86.95%**	**81.49%**
(Shi et al., 2017a; Shi et al., 2017b)	71.69%	66.89%	81.41%	74.73%	83.48%	76.54%
(Nguyen et al., 2017)	70.84%	61.21%	83.37%	76.95%	84.03%	77.98%

Table 2: Evaluation results on the PoSTWITA 2.2 test set using the different setups: training using only UD_Italian 2.1 (UD_It), training using only PoSTWITA 2.2 (UD_PoSTW) and training using both resources (UD_It+PoSTW). All the parser outputs were evaluated by using the standard script devised for the CoNLL-X evaluation.

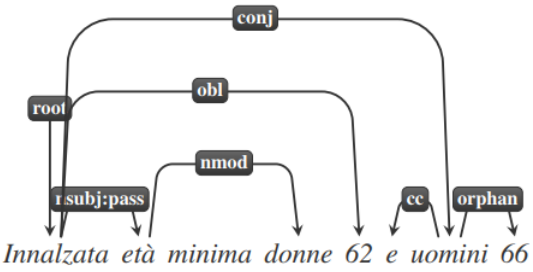


Figure 5: Annotation example of complex predicate ellipsis.

organized the experiments into three different setups:

1. in the first setup we trained and tuned the parsers by using the train and development sets belonging to the general Italian UD treebank UD_Italian v2.1 (UD_It). This setup is useful for comparing the parsers' performance when trained with general texts (out-of-domain) with the results obtained on Twitter data, similarly to what already proposed in Sanguinetti et al. (2017);
2. considering the importance of training with in-domain data, especially when it comes to non-standard texts (and UGC in particular), in the second setup we trained and tuned the parsers by using the UD_PoSTWITA v2.2 only (UD_PoSTW);
3. in the last set of experiments we trained and tuned the parsers by using both resources (UD_It+PoSTW).

In all the experiments the parsers were tested on the UD_PoSTWITA v2.2 test set. Table 2 shows the results for all setups. An in-depth analysis of parsing problems for tweets is well beyond the scope of this paper, but we can draw some provisional observations by examining the obtained results: first of all, the parser from Dozat and Manning (2017) consistently outperforms all the other competitors exhibiting a nice robustness also for this kind of texts. Second, even if the data sets from UD_PoSTWITA v2.2 are smaller than UD_Italian v2.1, in-domain data are fundamental for getting reliable results. Third, adding the UD_Italian 2.1 treebank does increase the performance, but only to a limited extent, suggesting, again, that out-of-domain data are less useful for obtaining good results.

Finally, considering the use of *parataxis* also for sentence linking (which is an unusual application of this label, as explained in Section 4.3.), we observed the performance of Dozat and Manning parser, trained using the UD_It+PoSTW setup, on this relation. The results are reported in Table 3 and show that, probably due to such specific use of the label, the parser results on its proper annotation are (relatively) low. We finally compared LAS and UAS scores on `parataxis` subtypes, observing that, except for hashtags, they were very poorly annotated, mainly because of their far lower frequency also in the training set.

Relation	Tot.	LAS	UAS
parataxis	509	60.31	66.60
parataxis:appos	16	12.50	75.00
parataxis:discourse	6	0	0
parataxis:hashtag	216	61.11	83.33
parataxis:insert	4	25	50.00
parataxis:nsubj	3	0	66.67
parataxis:obj	13	0	67.93

Table 3: Evaluation results on the PoSTWITA 2.2 test set for the specific *parataxis* relation and its subtypes, using the UD_It+PoSTW setup.

As already stated above, a more in-depth analysis would be necessary to study and better understand the main challenges of automatic tools in providing an accurate analysis of social media texts, specifically for Italian. We leave this point to future work.

6. Conclusion

In this paper, we presented the development of an Italian social media corpus annotated according to Universal Dependencies, and included in the official UD repositories since v2.1. This work aimed at showing how a *de facto* standard such as UD can be extended and applied also to one of the currently more widespread types of non-standard texts. Moreover, provided that the resource is especially tailored for NLP tools training, we also proposed a preliminary set of experiments with state-of-the-art parsing systems, in order to pave the way for an in-depth error analysis that takes into account all the annotation issues discussed in Section

4. in a more systematic way, thus overcoming the limitations of the present work. In the next future, we intend to further investigate this line of research also testing this approach on new Twitter data; as a side effect, this will result in a richer resource for training purposes.

Acknowledgements

The work of Cristina Bosco and Manuela Sanguinetti has been partially funded by Fondazione CRT (*Hate Speech and Social Media*, project n. 2016.0688) and by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media*, project S1618_L2_BOSC_01).

7. References

Andor, D., Alberti, C., Weiss, D., Severyn, A., Presta, A., Ganchev, K., Petrov, S., and Collins, M. (2016). Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2442–2452, Berlin, Germany, August. Association for Computational Linguistics.

Ballesteros, M., Dyer, C., and Smith, N. A. (2015). Improved transition-based parsing by modeling characters instead of words with lstms. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 349–359, Lisbon, Portugal, September. Association for Computational Linguistics.

Berzak, Y., Kenney, J., Spadine, C., Wang, J. X., Lam, L., Mori, K. S., Garza, S., and Katz, B. (2016). Universal dependencies for learner english. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 737–746, Berlin, Germany, August. Association for Computational Linguistics.

Bohnet, B. and Kuhn, J. (2012). The best of both worlds – a graph-based completion model for transition-based parsers. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 77–87, Avignon, France. Association for Computational Linguistics.

Bohnet, B. and Nivre, J. (2012). A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465, Jeju Island, Korea. Association for Computational Linguistics.

Bohnet, B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China. Coling 2010 Organizing Committee.

Bosco, C. and Mazzei, A. (2013). The evalita dependency parsing task: From 2007 to 2011. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7689 LNAI:1–12.

Bosco, C., Mazzei, A., Lombardo, V., Attardi, G., Corazza, A., Lavelli, A., Lesmo, L., Satta, G., and Simi, M. (2008). Comparing italian parsers on a common treebank: The evalita experience. pages 2066–2073.

Bosco, C., Montemagni, S., Mazzei, A., Lombardo, V., Dell'Orletta, F., Lenci, A., Lesmo, L., Attardi, G., Simi, M., Lavelli, A., Hall, J., Nilsson, J., and Nivre, J. (2010). Comparing the influence of different treebank annotations on dependency parsing. pages 1794–1801. cited By 9.

Bosco, C., Montemagni, S., and Simi, M. (2013). Converting Italian treebanks: Towards an Italian Stanford Dependency treebank. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 61–69.

Bosco, C., Tamburini, F., Bolioli, A., and Mazzei, A. (2016). Overview of the EVALITA 2016 Part Of Speech on TWitter for ITAlian task. In *Proceedings of Evalita 2016*.

Brants, T. (2000). Tnt: A statistical part-of-speech tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, ANLC '00, pages 224–231, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chen, D. and Manning, C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar, October. Association for Computational Linguistics.

Cheng, H., Fang, H., He, X., Gao, J., and Deng, L. (2016). Bi-directional attention with agreement for dependency parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2204–2214, Austin, Texas, November. Association for Computational Linguistics.

de Marneffe, M.-C., Grioni, M., Kanerva, J., and Ginter, F. (2017). Assessing the annotation consistency of the universal dependencies corpora. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017), September 18-20, 2017, Università di Pisa, Italy*, pages 108–115.

Dobrovoljc, K. and Nivre, J. (2016). The Universal Dependencies treebank of spoken Slovenian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28*, pages 1566–1571. European Language Resources Association (ELRA).

Dozat, T. and Manning, C. D. (2017). Deep biaffine attention for neural dependency parsing. In *Proceedings of the 2017 International Conference on Learning Representations*.

Foster, J., Çetinoglu, Ö., Wagner, J., Roux, J. L., Hogan, S., Nivre, J., Hogan, D., and van Genabith, J. (2011). #hardtoparse: POS tagging and parsing the twitterverse. In *Analyzing Microtext, Papers from the 2011 AAAI Workshop, San Francisco, California, USA, August 8, 2011*.

Gerdes, K. and Kahane, S. (2016). Dependency annotation choices: Assessing theoretical and practical issues of universal dependencies. In *Proceedings of Linguistic*

Annotation Workshop (LAW).

Gerdes, K. and Kahane, S. (2017). Trois schémas d'annotation syntaxique en dépendance pour un même corpus de français oral: le cas de la macrosyntaxe. In *Actes de l'atelier "ACor4French -– Les corpus annotés du français"*, pages 1–9, Orléans, France.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 42–47, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hu, Y., Talamadupula, K., and Kambhampati, S. (2013). *Dude, srsly?*: The surprisingly formal nature of twitter's language. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pages 244–253.

Kiperwasser, E. and Goldberg, Y. (2016). Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4.

Kong, L., Schneider, N., Swayamdipta, S., Bhatia, A., Dyer, C., and Smith, N. A. (2014). A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1001–1012, Doha, Qatar. Association for Computational Linguistics.

Lei, T., Xin, Y., Zhang, Y., Barzilay, R., and Jaakkola, T. (2014). Low-rank tensors for scoring dependency structures. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1381–1391, Baltimore, Maryland. Association for Computational Linguistics.

Lynn, T., Scannell, K., and Maguire, E. (2015). Minority language twitter: Part-of-speech tagging and analysis of Irish tweets. In *Workshop on Noisy User-generated Text*, Beijing, China.

Martínez-Alonso, H., Seddah, D., and Sagot, B. (2016). From noisy questions to minecraft texts: Annotation challenges in extreme syntax scenario. In *Proceedings of the 2nd Workshop on Noisy User-generated Text, NUT@COLING 2016, Osaka, Japan, December 11, 2016*, pages 13–23.

Mazzei, A., Lesmo, L., Battaglino, C., Vendrame, M., and Bucciarelli, M. (2013). Deep natural language processing for italian sign language translation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8249 LNAI:193–204.

Nguyen, D. Q., Dras, M., and Johnson, M. (2017). A novel neural network model for joint pos tagging and graph-based dependency parsing. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 134–142, Vancouver, Canada, August. Association for Computational Linguistics.

Rei, L., Mladenić, D., and Krek, S. (2016). A multilingual social media linguistic corpus. In *Proceedings of the 4th Conference on CMC and Social Media Corpora for the Humanities*, Ljubljana, Slovenia.

Ritter, A., Mausam, S. C., and Etzioni, O. (2011). Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11*, pages 1524––1534, Stroudsburg, PA, USA.

Sanguinetti, M., Bosco, C., Mazzei, A., Lavelli, A., and Tamburini, F. (2017). Annotating italian social media texts in universal dependencies. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017), September 18-20, 2017, Università di Pisa, Italy*, pages 229–239.

Shi, T., Huang, L., and Lee, L. (2017a). Fast(er) exact decoding and global training for transition-based dependency parsing via a minimal feature set. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 12–23, Copenhagen, Denmark, September. Association for Computational Linguistics.

Shi, T., Wu, F. G., Chen, X., and Cheng, Y. (2017b). Combining global models for parsing universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 31–39, Vancouver, Canada, August. Association for Computational Linguistics.

Silveira, N., Dozat, T., Marneffe, M.-C. D., Bowman, S., Connor, M., Bauer, J., and Manning, C. (2014). A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Tamburini, F. and Melandri, M. (2012). AnIta: a powerful morphological analyser for Italian. In *Proceedings of Language Resources and Evaluation Conference 2012*, pages 941–947.

Wang, H., Zhang, Y., Chan, G. L., Yang, J., and Chieu, H. L. (2017). Universal dependencies parsing for colloquial singaporean english. In Regina Barzilay et al., editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1732–1744. Association for Computational Linguistics.

Zhang, Y., Lei, T., Barzilay, R., and Jaakkola, T. (2014a). Greed is good if randomized: New inference for dependency parsing. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1013–1024, Doha, Qatar. Association for Computational Linguistics.

Zhang, Y., Lei, T., Barzilay, R., Jaakkola, T., and Globerson, A. (2014b). Steps to excellence: Simple inference with refined scoring of dependency trees. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Baltimore, Maryland. Association for Computational Linguistics.

Annotating If Authors of Tweets are Located in the Locations They Tweet About

Vivek Doudagiri, Alakananda Vempala and **Eduardo Blanco**
Human Intelligence and Language Technologies Lab
University of North Texas
VivekReddyDoudagiri@my.unt.edu, AlakanandaVempala@my.unt.edu, eduardo.blanco@unt.edu

Abstract

The locations in a tweet do not always indicate spatial information involving the author of the tweet. In this paper, we investigate whether authors are located or not located in the locations they tweet about, and temporally anchor this spatial information in the tweet timestamp. Specifically, we work with temporal tags centered around the tweet timestamp: longer than 24 hours before or after tweeting, within 24 hours before or after tweeting, and at the time of tweeting. We introduce a corpus of 1,200 location mentions from 1,062 tweets, discuss several annotation samples, and analyze annotator disagreements.

Keywords: Spatial knowledge, Social Networks, Twitter

1. Introduction

Twitter has quickly become one of the most popular social media sites. It has 313 million monthly active users, and 500 million tweets are published daily. People tweet about their current locations (e.g., *I'm at Walgreens in Anaheim, California*), as well as past (e.g., *I miss living in Kerman*) and (probable) future locations (e.g., *Can't wait to visit Italy!*). Tweets often contain hints regarding how long the author is in a particular location, either implicitly (e.g., people usually stay at pharmacies such as Walgreens for a few minutes to an hour, not days) or explicitly (e.g., *My 2 day vacation to San Diego beach starts tomorrow*).

In this work, we present a corpus annotating whether the author of a tweet is located in the locations mentioned in his tweets. Going beyond named entity recognition, we annotate whether the author is located or *not* located in the location he tweets about with respect to the time he tweeted (before, during and after). To the best of our knowledge, this problem has not been explored before. We found that no spatial relationship can be inferred between authors of tweets and the locations they tweet about in 21% of instances. In other words, 1 out of 5 locations in a tweet do not indicate any spatial information about the author.

The major contributions of this paper are:

1. We create a corpus of 1,062 tweets containing 1,200 location named entities, and annotate whether the authors are located or *not* located in those locations with respect to the time they tweeted (when the author tweeted, within 24 hours before and after he tweeted, and longer than 24 hours before and after he tweeted).[1]

2. We provide several annotation examples and the label distributions per temporal tag.

3. We present detailed inter-annotator agreement calculations, including Cohen's κ and confusion matrices per temporal tag.

[1]Available at https://alakanandav.bitbucket.io/

2. Previous Work

The work presented here is inspired by Li and Sun (2014), who work with points of interest in tweets using the Foursquare API (their points of interest are similar to our locations). Li and Sun determine if the author of a tweet is present at a point of interest in the past, present or future with respect to the tweet timestamp (three binary decisions). In their corpus, 47.3% of points of interest are annotated invalid, meaning that their methodology to extract points of interest using Foursquare is not very effective. In contrast to their work, we (a) present a corpus with few invalid locations ($\approx 6\%$), and (b) work with finer-grained temporal information (when somebody tweets, within 24 hours before and after he tweeted, and longer than 24 hours before and after he tweeted).

Jurgens et al. (2015) present a thorough evaluation of nine state-of-the-art network-based approaches to perform geolocation inference. They propose several evaluation methods, discuss possible sources of ground truth and their soundness, and conclude, among others, that real-world performance is much lower than initially reported. Mahmud et al. (2014) extract home locations of Twitter users at different granularities (e.g., city, state, time zone, geographic region). Their approach is a combination of statistical classifiers and heuristics, and takes into account the content of tweets (the actual words). Unlike them, we annotate spatial information from any location a person tweets about, we do not target the place of residence. Jurgens (2013) shows that social relationships help determining locations. They present an algorithm grounded on propagating spatial information through a user's social network. The algorithm does not rely on the specifics of any social network (the only requirement is that there are social relationships), and pinpoints the location of 50% of the users in a Twitter-based social network within 10 km. In contrast to this previous work, we work with specific locations mentioned in a tweet, and annotate whether the author was located there with respect to the time he tweeted.

	Tweet	Location	Before >24	Before <24	During	After <24	After >24
1	I'm at Walgreens in Anaheim, Calif	Calif	PY	CY	CY	CY	PY
2	Just got home from Vegas and I'm cooking omg	Vegas	CY	PY	CN	PN	UNK
3	First time in Squaw Valley and it could not have been more perfect! #squawvalley	Squaw Valley	CN	CY	CY	CY	UNK
4	I adopted a child while in Mexico	Mexico	CY	PN	CN	PN	UNK
5	Found some really cute couches in Oakland and did not have a car big enough to carry them back RIP	Oakland	PY	CY	CN	PN	UNK
6	Tomorrow we are driving to Yosemite Valley	Yosemite Valley	UNK	PN	CN	PY	PY
7	I can't wait to be in Hawaii permanently.	Hawaii	UNK	UNK	UNK	UNK	UNK
8	Bruh im bouta fall asleep on Keyasia	Keyasia	INV	INV	INV	INV	INV

Table 1: Annotation examples. We show the original tweet, the location being annotated (we only detail one location per tweet), and the labels for the five temporal tags.

3. Corpus Creation

In this section, we detail the creation of the corpus. First, we present the steps to gather tweets and select locations to be annotated. Second, we describe the annotation process and the kind of spatial information annotators were asked about. Finally, we present annotation examples.

3.1. Selecting Tweets and Locations

We collected tweets containing at least one location named entity following 4 steps:

1. We downloaded over one million tweets published from California along with their metadata using the Twitter API.[2] Then, we discarded tweets (a) consisting of less than 3 tokens, or (b) not containing at least one pronoun.[3]
2. We extracted named entities using spaCy[4] and Stanford CoreNLP (Manning et al., 2014) after removing emoticons, URLs and newline characters from the original tweets. The corpus contains both the original tweet and the preprocessed version.
3. We identified locations in tweets if both Spacy and Stanford CoreNLP recognized a LOC or GPE named entity (location and geopolitical named entities respectively) spanning exactly the same tokens.
4. Finally, we randomly selected 1,200 locations from 1,062 tweets for annotation.

3.2. Annotation process

For each location in the selected tweets, we asked annotators the following question: "Is the author of the tweet present in the location ...":

1. at any point of time earlier than 24 hours before tweeting (Before $>$ 24)?
2. at any point of time within 24 hours before tweeting (Before $<$ 24)?
3. at the time he tweeted (During)?
4. at any point of time within 24 hours after tweeting (After $<$ 24)?
5. at any point of time later than 24 hours after tweeting (After $>$ 24)?

We decided to work with these temporal tags because people usually tweet about what is happening, about what has happened recently, or about what is about to happen (Sanagavarapu et al., 2017).
We allow annotators to choose from six labels inspired by previous work on factuality (Saurí and Pustejovsky, 2012):

- Certainly Yes (CY): I am certain that the author is located in the given location at the specified time.
- Probably Yes (PY): I am not certain that the author is located in the given location at the specified time, but it is probably the case.
- Certainly No (CN): I am certain that the author is *not* located in the given location at the specified time.
- Probably No (PN): I am not certain that the author is *not* located in the given location at the specified time, but it is probably the case.
- Unknown (UNK): There is not enough information to answer any of the four labels above.
- Invalid (INV): The location is invalid, it is nonsensical to ask whether the author is (or is not) located there.

The entire corpus was annotated independently by two graduate students. Disagreements were adjudicated after in-person discussions between both annotators. Section 4.1. details the inter-annotator agreements.

[2] `https://developer.twitter.com/en/docs`

[3] Full list of pronouns: *I*, *we*, *you*, *he*, *she*, *it*, *they*, *him*, *her* and *them*. Tweets containing these pronouns are more likely to be authored by an individual (as opposed to companies or institutions).

[4] `https://spacy.io/`

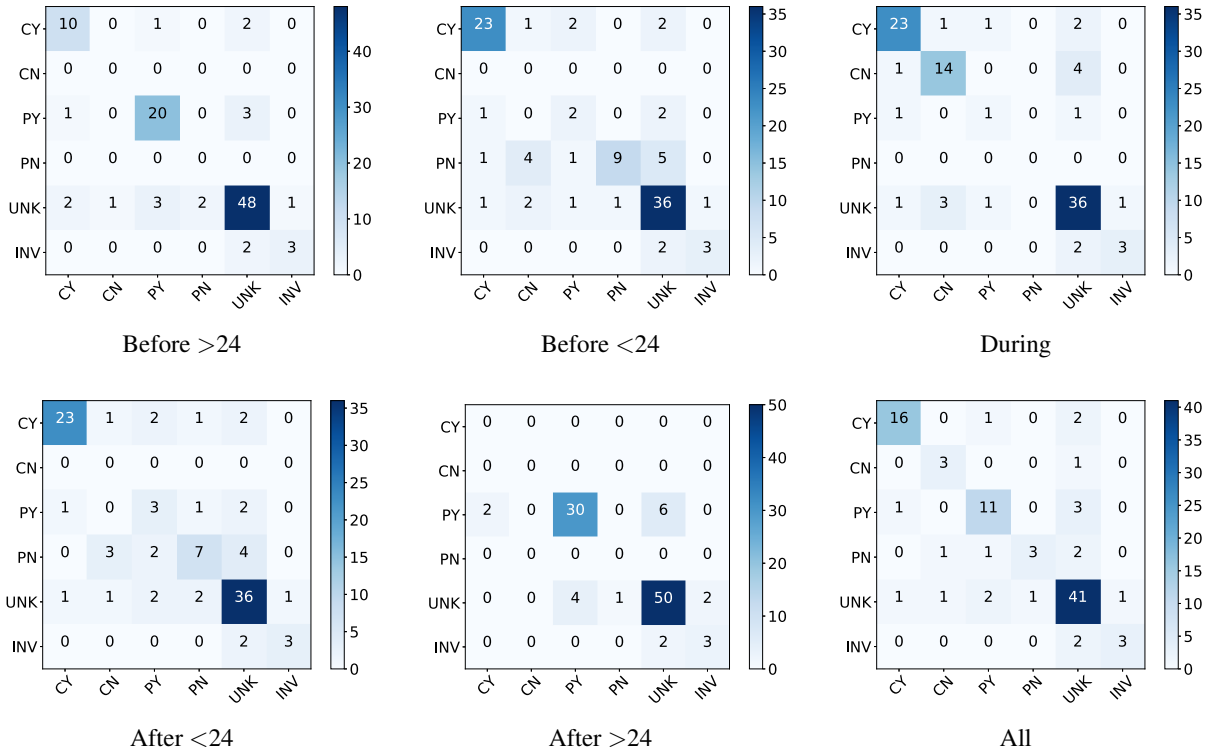


Figure 1: Inter-annotator confusion matrices (percentages for all pairs of labels from both annotators) per temporal tag. Note that annotators almost never disagree between (CY, PY) and (CN, PN).

3.3. Annotation Examples

Table 1 presents sample tweets and the annotations for all temporal spans. We briefly interpret the annotations below:

- Regarding tweet 1, annotators understood that the author is certainly in *Calif* when he tweeted, and also at some point of time within 24 hours before and after (one cannot leave *Calif* in a split second). However, they annotated that the author is probably still in *Calif* at some point of time 24 hours before and after, as it is not guaranteed that they author is there for an extended period of time.

- In tweet 2, the author describes the return from a trip to *Vegas*. Annotations reveal that the author is not in *Vegas* when he tweeted or shortly after (after <24, people usually don't travel to to the same destination within a day), but it is unknown whether he will go back in the long run (after >24). Annotations also reveal that the author was in *Vegas* prior to tweeting.

- In tweet 3, the author is also tweeting about a trip, this time to *Squaw Valley*. Annotators understood that the author tweeted shortly after arriving (before <24: CY) and he had not been there earlier (before >24: CN, keywords: *First time in*). They also annotated that the author is in *Squaw Valley* within 24 hours after tweeting, or in other words, that he was not leaving when he tweeted. Finally, there is not information to determine whether he was (or will be) there 24 hours after tweeting (after >24: UNK).

- Tweet 4 describes a past event. It is clear that the author was a participant in the event, and thus he was in *Mexico* at some point of time in the past. Annotators were certain that the author (a) was not in *Mexico* when he tweeted and (b) he was in *Mexico* at some point of time 24 hours before he tweeted (presumably adopting a child takes longer than a day).

- In tweet 5, the author describes a recent event (*Found some really cute couches*) that took place in *Oakland*. Annotators interpreted that *finding couches* occurred in the immediate past (before <24: CY), and maybe in the more distant past (before >24: PY). They also understood that the author left *Oakland* before tweeting (during: CN).

- Tweet 6 describes future plans for a trip to *Yosemite Valley*. Annotators understood that the author is likely to be there within 24 hours of tweeting (after <24: PY), and he will stay for longer than a day (after >24: PY). They also annotated that the author is certainly not there when he tweeted, and probably not there within a day before tweeting (before <24: PN). Finally, they couldn't determine if the author has or has not been at *Yosemite Valley* prior to 24 hours from tweeting (before >24: UNK).

- Tweet 7 describes a (possible) future state (the author wishes to *be in Hawaii*). Unlike in tweet 6, where the author appears to have made plans to drive to *Yosemite Valley* the day after tweeting, in tweet 7 it is not clear

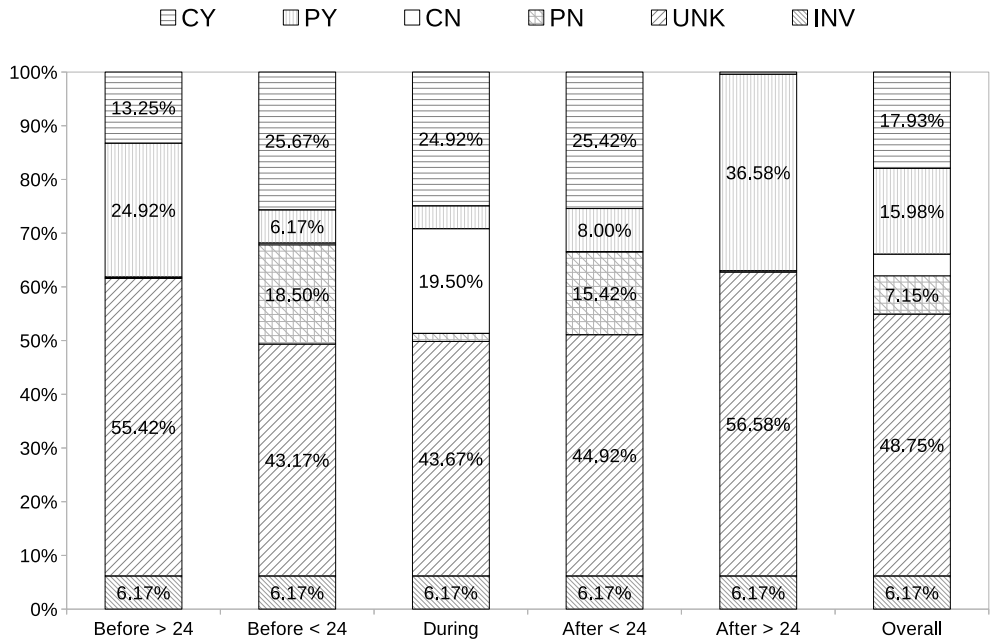


Figure 2: Label distribution per temporal tag and overall distribution after adjudicating disagreements. Percentage values are not shown if they are lesser than 5 %

	Before > 24	Before < 24	During	After < 24	After > 24
α	0.71	0.72	0.73	0.66	0.70

Table 2: Cohen's kappa (k) agreements per temporal tag.

if the author is simply stating a wish. Thus, annotators chose UNK for all temporal tags.

- Finally, tweet 8 contains an invalid location. *Keyasia* is a girl's name, but it was selected for annotation because both spaCy and Stanford CoreNLP tagged it as a location.

4. Corpus Analysis

In this section we analyze the corpus. Specifically, we discuss the quality in terms of inter-annotator agreement and the label distribution. Our corpus contains 1,200 location mentions extracted from 1,062 tweets (i.e., 1.14 locations per tweet on average).

4.1. Quality

Figure 1 presents confusion matrices per temporal tag and for all temporal tags (each matrix shows percentages for all pairs of labels from both annotators). Note that disagreements are relatively minor: most disagreements are between CY and PY, CY and UNK, CN and PN, CN and UNK, or UNK and INV. In other words, while annotators do not agree all the time, the sources of disagreements are not a major source of concern.

Table 2 presents inter-annotator agreements prior to adjudicating disagreements. The overall Cohen's kappa coefficient is 0.71, which is considered substantial (Landis and Koch, 1977). Coefficients range between 0.66 and 0.73 depending on the temporal tag. The highest agreement was obtained when annotating whether the author is (or is not) in a location when he tweeted (*during*: 0.73), and the lowest agreement when annotating whether the author is (or is not) in a location within 24 hours after tweeting (*after <24*: 0.66). The remaining agreements range from 0.70 to 0.72.

4.2. Label Distribution

Figure 2 presents the label distribution per temporal tag and overall after adjudicating disagreements. Overall, labels that allow us to extract spatial knowledge (CY, PY, CN and PN labels) account for 45.08% of labels. This percentage is similar across temporal tags, although we observe larger percentages of UNK with *before >24* and *after >24* (55.42% and 56.58% vs. 43.17%–44.92%). Authors are more likely to be located at the locations they tweet about than *not* located (CY+PY: 33.91% vs. CN+PN: 11.15%). Overall, we can infer the location of authors of tweets with around 20% certainty (CY+CN). For within a day before, during and within a day after posting the tweet, we can infer the location of the author with greater certainty (CY+CN: 26%, 44.42%, 26% vs. CY+CN: 13.25%, <5%). Also, we can infer the location of the author with greater certainty for before rather than after posting the tweet.

5. Conclusions

In this paper, we present a corpus of tweets annotated with temporally-anchored spatial information involving the author. Specifically, we annotated whether authors of tweets are located in the locations they tweet about when they tweet, within 24 hours before or after tweeting, and longer than 24 hours before or after tweeting.

The corpus has substantial inter-annotator agreement. Label distributions indicate that many locations present in tweets do not indicate any spatial information about the author. Additionally, annotators were more certain (CY and CN labels) when annotating spatial information for temporal tags close to the tweet timestamp (during, and within 24 hours before and after) than for more distant temporal tags (longer than 24 hours before and after).

6. Bibliographical References

Jurgens, D., Finethy, T., McCorriston, J., Xu, Y. T., and Ruths, D. (2015). Geolocation prediction in twitter using social networks: A critical analysis and review of current practice. *ICWSM*, 15:188–197.

Jurgens, D. (2013). That's what friends are for: Inferring location in online social media platforms based on social relationships. *ICWSM*, 13(13):273–282.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Li, C. and Sun, A. (2014). Fine-grained location extraction from tweets with temporal awareness. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 43–52. ACM.

Mahmud, J., Nichols, J., and Drews, C. (2014). Home location identification of twitter users. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):47.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Sanagavarapu, K. C., Vempala, A., and Blanco, E. (2017). Determining whether and when people participate in the events they tweet about. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 641–646, Vancouver, Canada, July. Association for Computational Linguistics.

Saurí, R. and Pustejovsky, J. (2012). Are you sure that this happened? assessing the factuality degree of events in text. *Comput. Linguist.*, 38(2):261–299, June.

MOCCA: Measure of Confidence for Corpus Analysis - Automatic Reliability Check of Transcript and Automatic Segmentation

Thomas Kisler, Florian Schiel
Institute of Phonetics and Speech Processing
Schellingstr. 3, 80799 München, Germany
{kisler, schiel}@bas.uni-muenchen.de

Abstract

The production of speech corpora typically involves manual labor to verify and correct the output of automatic transcription/segmentation processes. This study investigates the possibility of speeding up this correction process using techniques borrowed from automatic speech recognition to predict the location of transcription or segmentation errors in the signal. This was achieved with functionals of features derived from a typical Hidden Markov Model (HMM)-based speech segmentation system and a classification/regression approach based on Support Vector Machine (SVM)/Support Vector Regression (SVR) and Random Forest (RF). Classifiers were tuned in a 10-fold cross validation on an annotated corpus of spontaneous speech. Tests on an independent speech corpus from a different domain showed that transcription errors were predicted with an accuracy of 78% using an SVM, while segmentation errors were predicted in the form of an overlap-measure which showed a Pearson correlation of 0.64 to a ground truth using SVR. The methods described here will be implemented as free-to-use Common Language and Resources and Technology Infrastucture (CLARIN) web services.

Keywords: automatic segmentation, MAUS, confidence measure

1. Introduction

The creation of a new speech corpus typically involves three major steps: (1) recording, (2) (orthographic) transcription and (3) alignment of the transcription to the recorded signal, referred to hereafter as segmentation and labeling (S&L). The quality of these three production steps more or less defines the usefulness of the speech resource. The transcription of a speech recording (2) can be done either manually or via Automatic Speech Recognition (ASR). In both cases the transcription may contain errors in the form of deviations between the transcribed words and the words that were actually spoken. The S&L (3) can also be done either manually or automatically based on the transcript created at step (2). Since manual S&L is more time-consuming than (2) (slower by a factor of around 20 to 100), step (3) is often first done automatically (applying text-to-speech alignment or similar techniques) and then manually corrected afterwards. Both tasks – the manual correction of the transcript or the manual correction of the S&L – are expensive and time-consuming because every part of every utterance must be checked manually.

This study is concerned with the automatic and reliable detection of errors in the S&L either caused by falsely transcribed words[1] or by errors of the applied S&L system: Measure of Confidence for Corpus Analysis (MOCCA). More specifically, MOCCA consists of two methods for the reliable detection of word errors in the transcription and for the word-by-word estimation of the quality of the S&L in the form of an overlap measure. Both MOCCA methods can be used to automatically identify parts in the S&L where 'something went wrong', and thus facilitate the manual correction task.

The estimation of the correctness of a word label in a transcription based on the speech signal is similar to the assessment of a hypothesized word in ASR systems during or after recognition. In ASR research such confidence measures have attracted significant attention and have been used to detect recognition errors or to detect out-of-vocabulary (OOV) words. Good overviews for confidence measures are Jiang (2005) and Seigel (2013). Both classify confidence measures into three categories: 1) the *posterior probability approach*, which estimates the true posterior probabilities by approximating the probability mass function of all possible acoustic feature vectors, 2) the *utterance verification approach*, which treats the problem of confidence estimation as a statistical hypothesis testing problem (using Likelihood Ratio Testing), where an approximation of the alternate hypothesis is needed for a reliable decision, and 3) the *classification approach*, in which a model is trained to estimate whether or not a word is correctly recognized.

Kemp et al. (1997) applied a *classification approach* with a linear classifier and a neural net on features that were extracted from the ASR decoder process. As Zhang and Rudnicky (2001) and also Seigel (2013) pointed out, a fundamental problem of ASR decoder features is that the features to generate the hypotheses and the features that are used to make the prediction about the quality of those hypotheses are the same, and are therefore not optimal to assess the quality of the ASR output in a post-processing step. This fundamental problem does not apply in our case because the S&L system has two inputs: the speech signal and the transcript, which is produced independently from the S&L decoding.

Paulo and Oliveira (2004) used features from a forced-alignment system to estimate the quality of automatic S&L. They estimated a measure called the Overlap Ratio (OvR), in which the overlap on phoneme level was estimated (cf. section 4.2.). In contrast to that earlier study, we aim to estimate the OvR over a complete word.

A S&L decoding process essentially resembles that of a speech recognizer, but is not identical since it lacks some

[1] Note that for the purpose of this study it is irrelevant whether these errors stem from ASR or a manual transcription

features (Kemp et al., 1997), for example the number of times the model switches to a lower N-gram model, or the number of active final word states. Therefore we use a subset of the features described in Kemp et al. (1997) that were suitable for the present study and could be extracted without additional processing (e.g. estimating the Signal-to-Noise ratio of a word segment, etc.).

The remainder of this paper is organized as follows: the next section briefly describes the basic experimental setup, the S&L system to be evaluated, the features used and the classifiers that were applied. Section 3. outlines the speech data on which we test the proposed MOCCA methods, and in section 4. we describe the two experiments and discuss their results.

2. Method

2.1. Overview

The MOCCA tagger was based on a *classification approach*, which introduces a post-processing step after the actual alignment. The general setup of the experiments was as follows: test data consisting of the speech signal and the corresponding transcript were processed by the S&L system Munich AUtomatic Segmentation System (MAUS) described in Schiel (1999). Based on features derived from the MAUS decoding process, MOCCA tagged each word of the input transcript as to whether it matched the speech signal or not (experiment 1) and at the same time estimated the degree of overlap (OvR) between the calculated segmentation and the ground truth segmentation (experiment 2). The estimation of whether a word label is correct is a two-class classification problem, while the prediction of the OvR is a regression task; for both tasks, classification and prediction, we tested a SVM, which was reported to give good results in Zhang and Rudnicky (2001), and a RF.

2.2. S&L System MAUS

The transcript text input was converted into a canonical phonological transcript using the grapheme-to-phoneme service G2P (Reichel, 2012). The phonological transcript was then passed to the MAUS service, which first generated a probability graph for all predictable pronunciations together with their prior probabilities (Schiel, 2015), and then decoded this graph into the most likely S&L using the Hidden Markov Toolkit (HTK, Young et al. (2002); for details about the MAUS technique see Schiel (1999)). Features for the confidence measure experiments were extracted from the HTK Viterbi decoding as described in the following section.

2.3. Features

Kemp et al. (1997) showed that features from the output of an ASR system can be utilized to predict the correctness of the recognized words. Since the automatic segmentation obtained with MAUS is not exactly the same as the output of an ASR system, only a subset of the features described in Kemp et al. (1997) were extracted for each segmented word:

logLM: the log prior language model probability

logAP: the log posterior probability as produced by the HTK Viterbi decoder (Young et al., 2002)

logAPNorm: the log posterior probability normalized by log prior probability: $logAPNorm = logAP - logLM$

Duration: the duration after alignment of the phoneme sequence as output by G2P of the segmented word

SpkRate: the local speaking rate, calculated as the ratio of the mean phoneme sequence duration of the word in the training data $MeanDuration$ and $Duration$: $SpkRate = \frac{MeanDuration}{Duration}$

logNPhones: the logarithm of the number of phonemes in the target word according to segmentation

The MAUS system models phones not words. Since words have variable numbers of phones n, it follows that for each word, n feature values **logLM**, **logAP** and **logAPNorm** are produced. To circumvent the problem of feature vectors with variable lengths, we used the following functionals of these features: *sum, mean, median, range, standard deviation, variance* and *Discrete Cosine Transform (DCT) coefficients 1-3*.

- sum: $sum(X) = \sum_{i=1}^{n} x_i$
- mean: $\overline{X} = \frac{1}{n} \sum_{i=1}^{n} x_i$
- median: $med(X) = \begin{cases} x_{\frac{n+1}{2}} & n \text{ odd} \\ \frac{1}{2}\left(x_{\frac{n}{2}} + x_{\frac{n+1}{2}}\right) & n \text{ even} \end{cases}$
- range: $range(X) = \max(x) - \min(x)$
- standard deviation: $\sigma(X) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{N-1}}$
- variance: $Var(X) = \sigma^2$
- DCT coefficients 1-3: $C_k(X) = \sum_{i=1}^{n} x_i \cos\left[\frac{\pi}{n}(i + \frac{1}{2})k\right]$, for $k = 1, 2, 3$

where n is the number of phonemes and x_i is the feature value of the $i-th$ phoneme of a given feature. This yields a feature vector of constant dimensionality $p = 30$ for each word.

2.4. Classifiers

For both experiments we tested two different classification/regression algorithms: SVM and RF (Meyer et al., 2015; Wright and Ziegler, 2015). Both classifiers support binary classification and regression (with minor differences, e.g. the splitting criterion).

SVM The two best SVM kernels reported in Zhang and Rudnicky (2001) were a Gaussian Radial Basis Function (RBF) kernel of the form:

$$k(u, v) = exp(-\gamma \|u - v\|^2) \quad (1)$$

and the ANOVA RBF kernel of the form:

$$k(u, v) = \sum_{k=1}^{n} exp(-\sigma(u^k - v^k)^2)^d \quad (2)$$

Since the training of the ANOVA kernel is quite time-consuming, and Karatzoglou et al. (2005) report that ANOVA RBF kernels generally perform well in regression problems, we applied this kernel only for the regression task in experiment 2. The Gaussian RBF kernel was applied in both experiments 1 and 2.

Since the SVM is sensitive to its hyperparameters, we tuned them by performing a standard grid search: in the case of the Gaussian RBF kernel, we tuned the parameters C (values tested: $C = 0.0001, 0.001, 0.01, 0.1, 1, 10, 100$) and γ (values tested: $\gamma = 0.0001, 0.001, 0.01, 0.1, 1, 10, 100$). For the ANOVA RBF kernel we tuned the parameters C (values tested: $C = 0.1, 1, 10$), σ (values tested: $\sigma = 0.1, 1, 10$) and $degree$ (values tested: $degree = 1, 2, 3$).

For implementation we applied the R Programming Language (R) package *e1071* (Meyer et al., 2015) which uses the *LibSVM* library (Chang and Lin, 2011), a parallelizable implementation of SVMs.

Random Forest Fernández-Delgado et al. (2014) showed that RFs often have similar or better performance in classification problems than SVMs. Additionally, RFs have the advantage that they are less sensitive to their (tunable) parameters (Breiman, 2001; Archer and Kimes, 2008; Díaz-Uriarte and De Andres, 2006), and that they can be parallelized more efficiently than SVM. The two RF parameters we tuned were the number of trees to grow ($ntree = 50, 100, 200, 500$) and the number of features to consider at each split in the tree ($mtry = \sqrt{p}, 8, \frac{p}{3}$).

We used the *R* package *Random Forest Generator (ranger)* to train the random forests, since ranger is to our knowledge the fastest RF implementation available in R (Wright and Ziegler, 2015).

3. Test Data

We tested MOCCA on recordings from two different speech corpora. For training and parameter tuning in a 10-fold Cross Validation (CV)[2] we used a subset of the Kiel Corpus (Kohler, 1995). To evaluate the performance of MOCCA, we used recordings from the PhonDat2 (PD2) corpus (The ASR Consortium, 1995) as an independent test set. Both corpora have a manually verified orthographic transcript and a manual S&L which was produced by trained phoneticians. Both corpora contain German read and spontaneous speech produced by native German speakers.

Kiel Corpus: The subset of the Kiel corpus used in the present study consists of 2225 utterances from spontaneous speech produced by 30 speakers doing the appointment scheduling task and while performing a map task[3] (John, 2012).

PD2 The subset of the PD2 corpus used in the present study consists of read speech produced by 16 speakers, who each produced 64 semi-spontaneous utterances doing an information query task (1024 utterances total).

[2]In each fold a speaker is either part of the test or training set and the number of observations is balanced so that each of the 10 test and 10 training sets has roughly the same size.

[3]For a detailed description of a map task please refer to (Anderson et al., 1991)

4. Experiments and Results

4.1. Experiment 1: Correctness of Transcription

Overview: This section describes experiment 1, the goal of which was to correctly recognize, whether a word in the given transcript is correct or not. An example of the assignment of correct confidence measure labels to an incorrect transcript can be found in Table 1.

In this section we tested the following hypothesis: The features described in section 2.3. carry enough information to classify each segmented word into the classes 'correct transcript' versus 'incorrect transcript'. Since there were no transcription errors in the test sets, we applied a replacement strategy on every test recording to introduce artificial transcription errors as explained in the following paragraphs.

Real utterance:	Have	a	great	day
Confidence Measure Label:	*C*	*C*	**I**	*C*
Given Transcription:	Have	a	bad	day

Table 1: Illustration of the mode of operation of MOCCA by labeling incorrectly transcribed parts of speech (correct: *C*; incorrect: **I**).

Artificial Transcription Errors: First a MAUS S&L was performed on the test recording and features were extracted from the decoder output for all words. Since we assume that the transcript is correct, these features represent the "correct transcript" case. We then repeated the S&L over the complete recording, once for each word w_o that had an OvR (see section 4.2.) of more than 90% between the MAUS S&L and the ground truth segmentation, but with w_o replaced by another (wrong) word w_r in the transcript. Again, features were extracted from the decoder output for the replaced word w_r, this time representing the "incorrect transcript" case. w_r was randomly selected from the corpus' word list with two restrictions: first, the number of characters $length(w_r)$ had to be in the range of ± 1 compared to the length of original word $length(w_o)$; second, the word-length normalized Levenshtein distance (Levenshtein, 1966) between w_r and w_o had to be at least 75%. If no word could be found in the range ± 1, the range was incrementally increased, until a replacement could be made. For example, a rejected replacement for "train" would be "rain", since "rain" only fulfills the length requirement and not the Levenshtein requirement; a valid replacement for "train" would be for instance "wash".

The word length restriction was introduced so that replacement words had roughly the same amount of phonemes. This is crucial in cases for words that are originally very short and are replaced by much longer words e.g. "ja" by "Zugverbindung" ("yes" by "train connection"). In this case the MAUS S&L may fail, since the available time frame is too short for the number of phonemes to be aligned.

The Levenshtein condition was introduced to ensure a significant difference in pronunciation between the original

and the replacement word. Although this probably simplified the recognition task to a certain degree, it allows us to test the feasibility of transcription error detection based on features from the Viterbi decoder in the first place.[4]
This replacement strategy had two benefits: firstly, many training examples could be generated automatically, and secondly the training set was balanced with regard to the output classes (every word in the transcript was analyzed once as correct and once as incorrect). The procedure applied to the test data yielded a total of 26,649 training examples.

Results: Table 2 summarizes the results of the classification. Hyperparameters of both classifiers were optimized to a 10-fold CV of Kiel Corpus (see values in the caption of Table 2). We report accuracy, precision and recall for the model yielding the best accuracy, defined as:

$$\text{Accuracy} = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}$$

$$\text{Precision} = \frac{t_p}{t_p + f_p}, \quad \text{Recall} = \frac{t_p}{t_p + f_n}$$

where t_p are the true positives, t_n the true negatives, f_p the false positives and f_n the false negatives ("bad" is the positive class).
The lower half of 2 shows the results for tests on the independent test set PD2 using the same parametrization.

Corpus	Class.	Accuracy	Precision	Recall
Kiel	SVM	0.7822	0.7897	0.7672
	RF	0.7908	0.7862	0.7968
PD2	SVM	0.7876	0.7785	0.7868
	RF	0.7526	0.6923	0.7794

Table 2: Results of the classifiers SVM and RF, when tuned to the Kiel corpus (maximal accuracy); the SVM was built with tuned parameters $C = 100$ and $\gamma = 0.1$; the RF was built with parameters $ntree = 500$ and $mtry = 8$.

The SVM and the RF both showed similar performance metrics in the 10-fold CV (Kiel); the RF had a slightly better accuracy than the SVM. This result is consistent with Fernández-Delgado et al. (2014) who also found that RFs and SVMs had similar accuracies in classification tasks.
When testing against the independent test set PD2, the accuracy obtained for the SVM was very close to the one on the Kiel data set; it therefore seems that the SVM generalized better than the RF. The RF also showed a skewed distribution towards predicting more false negative results f_n which lead to a decrease in precision by more than 10% on the independent data set.

Example: An example can be seen in Table 3, where MOCCA used the best SVM model to predict the correctness of each word in a transcript taken from the PD2 corpus. The classification results of two transcripts are shown: one correct (top) and one where the word "es" ("it") was replaced by "man" ("one") following the replacement strategy described in section (bottom). In both cases the aligned transcript is shown together with class probabilities and classification result of the SVM.
As expected the class probability was decreased for the replaced word in the wrong transcript (underlined). Additionally, the class probability for the following word was decreased as well. This is due to the fact that the wrong transcript also influenced the segmentation of the following word.

Class prob.:	0.9299	0.7325	0.7523	0.7225
Class labels:	*C*	*C*	*C*	*C*
Correct Transcr.:	Geht	es	nicht	eher
Wrong Transcr.:	Geht	man	nicht	eher
Class labels:	*C*	**I**	*C*	*C*
Class prob.:	0.9330	0.1252	0.6821	0.7225

Table 3: A real live example from the PD2 corpus of the German sentence "Geht es nicht eher" (which loosely translates in this context to "Isn't there an earlier connection"). The replaced (wrong) word in the transcript is underlined (see text for details).

4.2. Experiment 2: Segmentation Quality

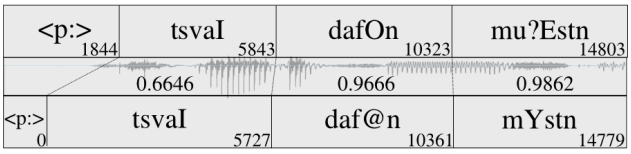


Figure 1: A real example of phoneme strings and their alignment: an automatic and erroneous S&L (top), a manual and correct S&L (bottom) and the resulting OvR values (middle).

Overview: In this experiment the predictive power of the extracted features (cf. Section 2.3.) with regard to the segmentation quality was evaluated by predicting the values for the OvR and comparing these to the OvR from the ground truth segmentation.
The OvR is a measure of the amount of overlap between two given time segments. It is independent of the duration of the segments t_i and t_j and is defined as (Paulo and Oliveira, 2004):

$$OvR = \frac{t_{ij}}{t_i + t_j - tij} \tag{3}$$

where t_i and t_j are the duration of the segments i and j respectively (see Figure 2).
The range of the OvR is from $]-\infty, 1]$. However, in our case it made no difference whether something was "negatively overlapped" ($OvR < 0$), meaning that a gap existed between the segments, or $OvR = 0$, meaning that the end of segment t_i was the beginning of segment t_j. We therefore forced all $OvR < 0$ to $OvR = 0$ so that the range of

[4] It would be interesting to measure the Levenshtein distance in real data, and study how this influences the outcome of the error detection, but at the time of writing such data were not available.

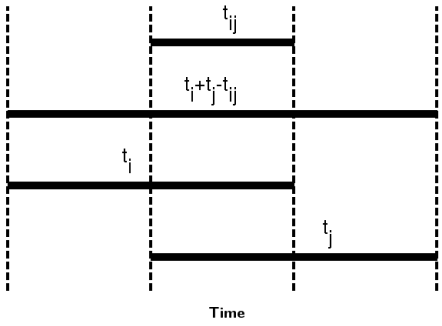


Figure 2: The visualization of the OvR as described in equation 3 (adapted from Paulo and Oliveira (2004)).

OvR became $[0, 1]$ (where 1: perfect match of segment to ground truth; 0: total mismatch).
Unfortunately the overlap ratio was not equally distributed over the possible range of values: there were very many values close to 1 (indicating an almost total overlap of word segments to the ground truth segmentation) and very few values < 1 (indicating a small overlap). While this is actually a good sign since it means that the MAUS S&L was in most cases correct, it made it difficult to train a model that can predict the OvR equally well over the complete range of possible values.

Binning: To address this problem we divided the OvR into 20 equally sized bins of width 0.05 between 0 and 1 and restricted all bins to the same number of measurements. We set this number of measurements to the average over all bins (1890 observations), and selected these randomly from the available measurements. This corresponds to an under-sampling strategy, in which the bins with a higher number of observations are under-sampled more often than the bins with fewer observations. This strategy resulted in a total of 55,739 training observations from the Kiel Corpus.

Results: Table 4 summarizes the results for the SVR as well as the RF. The values reported are the Pearson correlation coefficient between real and predicted OvR (CorCoeff), mean absolute error (MAE), and root mean squared error (RMSE). Again we only report values resulting from the best parametrization of the hyperparameters (see caption), in this case tuned according to the correlation coefficient (CorCoeff). The results of the ANOVA RBF kernel were omitted, because the best parametrization of the SVR model based on a standard hyperparameter grid search resulted in a weak negative correlation (-0.26).
The results were again very similar for the SVR and RF regression in the cross validation (Kiel). When applied to the independent test set (PD2), the correlation significantly decreased for all classifiers, which indicates that the models did not generalize well for this prediction task. Again the values for the RF deteriorated more than those of the SVR.

Example: An example for the prediction of the overlap ratio by MOCCA is shown in Figure 3. The example consists of a sentence of 10 words, for which the OvR was calculated on the ground truth segmentation (red) and estimated using the best SVM model described in Table 4 (blue). A perfect model would have resulted in identical values. It can be seen that the prediction generally followed the trend of the true OvR values.

Corpus	Class.	CorCoeff	MAE	RMSE
Kiel	SVR GRBF	0.7336	0.1277	0.1856
	RF	0.7430	0.1290	0.1814
PD2	SVR GRBF	0.6415	0.09464	0.1346
	RF	0.5955	0.1143	0.1535

Table 4: Results of the SVR with Gaussian RBF (GRBF) kernel and of the RF (both parametrizations were optimized according to the Pearson correlation coefficient (CorCoeff)). The GRBF SVR was built with parameters $C = 1$ and $\gamma = 0.1$; the RF was built with parameters $ntree = 500$ and $mtry = \frac{p}{3}$.

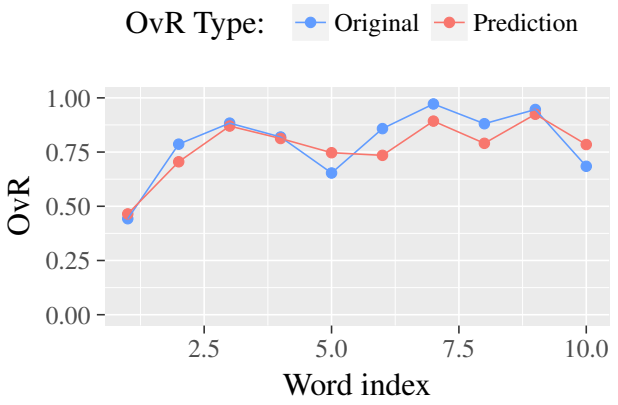


Figure 3: The true overlap ratio and the predicted overlap ratio of an example sentence with 10 words.

5. Discussion

Experiment 1 suggests evidence that erroneous words in a transcript can be detected from the results of a MAUS S&L procedure with about 78% accuracy (at roughly equal error types). The SVM classifier outperforms the RF in terms of generalization when applied to data from another corpus.
The advantage of the classification applied in this study compared to confidence measure estimation in ASR systems is that it uses two knowledge sources by combining the information from the independent transcriber (be it a human or an ASR system) with the MAUS alignment features; this partly explains the high accuracy.
Experiment 2 is similar to confidence measure estimation in speech recognition, because the same features are used for deciding word boundaries and then afterwards for estimating the quality of these boundaries. We showed that the prediction of the confidence measure is possible, but only to a Pearson correlation of about 0.64 tested on the independent data set. It remains to be seen whether this prediction

is good enough to be useful in a practical corpus correction scheme. Regression using a ANOVA RBF kernel did not yield any usable results in experiment 2. Thus, the positive results reported by Zhang and Rudnicky (2001) could not be replicated in our setting.

In addition to the undersampling strategy in experiment 2, an oversampling strategy e.g. as in Torgo et al. (2013) could improve the regression analysis. This could be especially beneficial for the detection of overlap ratios that are close to 0 (no overlap), as the data set could be balanced out better than with the simple undersampling strategy.

To summarize, the prediction of transcription word errors as described in this study appears to be a promising method to make the process of speech corpus annotation more efficient; the method based on SVM will be implemented and made available via a web-interface and as a web service within the CLARIN infrastructure (see `http://clarin.phonetik.uni-muenchen.de/BASWebServices`). The prediction of S&L time-alignment errors turned out to be more challenging and will need further attention in the future.

6. Acknowledgments

This work has been partly supported by the German Federal Ministry of Education and Research (BMBF) in the CLARIN-D project (CLARIN, 2017).

7. Bibliographical References

Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., et al. (1991). The HCRC map task corpus. *Language and speech*, 34(4):351–366.

Archer, K. J. and Kimes, R. V. (2008). Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis*, 52(4):2249–2260.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27.

CLARIN. (2017). CLARIN-D web page. last accessed: 2017-09-5.

Díaz-Uriarte, R. and De Andres, S. A. (2006). Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3.

Fernández-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, 15(1):3133–3181.

Jiang, H. (2005). Confidence measures for speech recognition: A survey. *Speech communication*, 45(4):455–470.

John, T. (2012). *EMU Speech Database System: praxisorientierte Weiterentwicklung der Funktionalität, Benutzerfreundlichkeit und Interoperabilität sowie die Aufbereitung des Kiel Corpus als EMU-Sprachdatenbank*. Ph.D. thesis, München, Ludwig-Maximilians-Universität, Diss., 2012.

Karatzoglou, A., Meyer, D., and Hornik, K. (2005). Support vector machines in r.

Kemp, T., Schaaf, T., et al. (1997). Estimating confidence using word lattices. In *EuroSpeech*, pages 827–830.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10:707, February.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F., (2015). *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*. R package version 1.6-7.

Paulo, S. and Oliveira, L. C. (2004). Automatic phonetic alignment and its confidence measures. In José Luis Vicedo, et al., editors, *Proceedings of Advances in Natural Language Processing: 4th International Conference (EsTAL), Alicante, Spain*, pages 36–44. Springer Berlin Heidelberg.

Reichel, U. (2012). PermA and Balloon: Tools for string alignment and text processing. In *Proc. Interspeech*, pages 1874–1877, Portland, Oregon.

Schiel, F. (1999). Automatic Phonetic Transcription of Non-Prompted Speech. In *Proc. of the International Conference on Phonetic Sciences*, pages 607–610, San Francisco, August.

Schiel, F. (2015). A statistical model for predicting pronunciation. In *Proc. of the International Conference on Phonetic Sciences*, page paper 195, Glasgow, United Kingdom, August.

Seigel, M. S. (2013). *Confidence estimation for automatic speech recognition hypotheses*. Ph.D. thesis, University of Cambridge.

Torgo, L., Ribeiro, R. P., Pfahringer, B., and Branco, P., (2013). *SMOTEfor Regression*, pages 378–389. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wright, M. N. and Ziegler, A. (2015). ranger: A fast implementation of random forests for high dimensional data in C++ and R. *arXiv preprint arXiv:1508.04409*.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., et al. (2002). The HTK book. *Cambridge university engineering department*, 3:175.

Zhang, R. and Rudnicky, A. I. (2001). Word level confidence annotation using combinations of features.

8. Language Resource References

Kohler, Klaus. (1995). *Das Kiel Corpus*. IPdS, Albrecht-Universität Kiel, 1 + 2, ISLRN 613-489-674-355-0.

The ASR Consortium. (1995). *Phondat2 Corpus (PD2)*. Bavarian Archive for Speech Signals, 3.0, ISLRN 937-744-173-899-5.

Towards an ISO Standard for the Annotation of Quantification

Harry Bunt[1], Kiyong Lee[3], and James Pustejovsky[2]
[1]Tilburg Center for Communication and Cognition, Tilburg University, The Netherlands
[2]Brandeis University, Waltham, MA, USA
[3]Korea University, Seoul, Korea
bunt@uvt.nl, pustejovsky@cs.brandeis.edu, ikiyong@gmail.com

Abstract

This paper presents an approach to the annotation of quantification that is being developed in preparation of the specification of a quantification annotation scheme, as part of an effort of the International Organisation for Standardisation ISO to define interoperable semantic annotation schemes. The paper focuses on the theoretical basis for an ISO standard annotation scheme in this area. It is argued that the combination of Generalized Quantifier Theory, neo-Davidsonian event-bases semantics, Discourse Representation Theory, and the ISO Principles of semantic annotation forms a powerful and solid foundation for defining annotations of quantification phenomena with an abstract and a concrete syntax and a compositional semantics. The coverage of the proposed annotation scheme includes both count and mass NP quantifiers, as well as NPs with syntactically and semantically complex heads with internal quantification and scoping structures, such as inverse linking by prepositional phrases and relative clauses.

Keywords: semantic annotation, annotation standaard, quantification

1. Introduction

Quantification occurs in every sentence of written text or spoken discourse. This is because the application of a predicate to one or more sets of objects gives rise to questions of relative scope, of cardinality, and of the distribution (or 'distributivity') of the predicate over the sets of arguments. Dealing with these questions is of crucial importance for understanding, translating, or responding to sentences, and for correct information extraction and question answering from natural language text. Example sentence (1a) illustrates the influence of the relative scopes of quantifications on the information that an utterance contains (how many papers weere read altogether?); (1b) illustrates the importance of the collective or individual distribution of a predicate over a set of arguments (assuming that a certain piano was carried collectively by certain men, who individually had a beer); and (1c) shows that a quantified relation may involve parts of individuals, such as pizza halves.

(1) a. All the students read two papers on quantification.

b. The men had a beer before carrying the piano upstairs.

c. The three boys ate four and a half pizzas.

The International Organisation for Standardisation ISO is pursuing the establishment of standards for linguistic annotation in general, and semantic annotation in particular, in view of the importance of annotated language resources, both for empirically-based linguistic research and for developing NLP applications, In order to be widely applicable across theories and platforms, annotation standards should on the one hand be theory-neutral, but they should on the other hand also take established theoretical insights into account. This paper outlines an ISO standard annotation scheme under development for quantification that builds on logical and linguistic theories, notably on the theory of generalized quantifiers, on neo-Davidsonian event-based semantics, and on Discourse Representation Theory. in combination with established principles for semantic annotation.

2. Theoretical background

2.1. Generalized Quantifier Theory

Philosophers, linguists, and AI researchers have extensively studied quantifiation phenomena. Logicians from Aristotle to van Benthem have studied quantification for its role in reasoning and thinking. It was noted relatively recently that quantifiers can be viewed as expressing a property of the involvement in a predication of sets of individual objects: $\forall x$ expresses that all the individual objects in the universe of discourse are involved; $\exists x$ that at least one such object is involved (Mostowski, 1957; Lindström, 1966). This notion of a quantifier has been generalized to properties such as those expressed in English by *"two", "most", "less than half of"*, *"more than 2000"*. The concepts in this broader class of quantifiers are called *generalized quantifiers*.
Quantifications in formal logic are statements about all individual objects in a given universe of discourse; natural languages, by contrast, have quantifying expressions like *"all the students", "a book", "some wine"*, and *"more than five sonatas"*, which indicate a specific domain that the quantification is restricted to. Generalized quantifier theory (GQT) therefore views noun phrases, rather than determiners, as the quantifiers of natural language (Barwise and Cooper, 1981). According to GQT, words like *all* and *some* in English do not form the counterparts of the universal and existential quantifiers of formal logic, and neither do words like *three*, and *most*, which have been called 'cardinal quantifiers' and 'proportional quantifiers' (Partee, 1988).

2.2. Event-based semantics

Some aspects of sentence meaning can be accounted for only if verbs are viewed as introducing events - in a broad sense of 'event' that includes states, facts, processes, and

their negations. Examples of such aspects are adverbials, but also quantifying expressions such as *"always", "twice", "never"* and *"more than three times"*, which refer to sets of events. These expressions say something about the absolute or relative cardinality of a set of events of a certain type. Similarly for expressions of frequency, as in *"I will call you twice every day"*. Adverbs, such as adverbs of manner, often express a property of events. Observations like these led Davidson (1967) to propose to treat events as individual objects.

Following Parsons (1990) this view can be expressed in semantic representations by means of one-place predicates applied to existentially quantified event variables, using semantic role relations to indicate the roles of the participants in an event. This approach has been widely accepted in modern semantics, and has been adopted in the ISO annotation standards 24617-1 (Time and events), 24617-4 (Semantic roles), and 24617-7 (Spatial information).

For the semantic annotation of quantification, we propose an approach that combines GQT with the neo-Davidsonian treatment of predicate-argument relations, including the use of semantic roles as defined in ISO 24617-4. This approach allows the expression in annotations of quantification aspects such as the collective/individual distinction as a property of the way in which a set of individuals participates in a set of events. For example, the ISO 24617-4 annotation[1] of example sentence (2) would look as in (3).

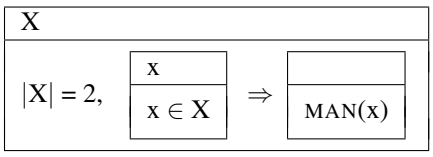
(2) Two men lifted the piano.

(3) (Markables: m1="Two men", m2="lifted, m3="the piano".)

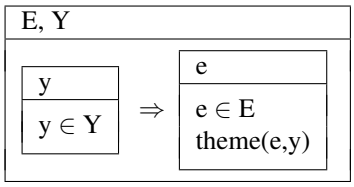
```
<entity xml:id="x1" target="#m1" pred="man"
  involvement="2" definiteness="indef"/>
<event xml:id="e1" target="#m2" pred="lift"/>
<srLink event="#e1" participant="#x1"
  semRole="agent">
<entity xml:id="x2" target="#m3" type="piano"
  involvement="1" definiteness="def"/>
<srLink event="#e1" participant="#x2"
  semRole="theme"/>
```

This theoretical basis is also brought out in the semantics of the annotations, which makes use of Discourse Representation Structures (DRSs) that involve sets of events with sets of participants. For example, the annotation of the NP *"Two men"* is interpreted as the DRS in (4a), which can be read as follows: *There is a set X of cardinality 2 that consists of men*. A semantic role link, like the one for the theme role with individual distributivity, is interpreted as the DRS in (4b), and the sentence *"Two men lifted a piano"* is interpreted as the DRS (4c), obtained by combining the DRSs for the NPs, the verb, and the semantic role relations.

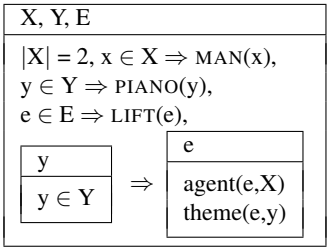
(4) a.



b.



c.



2.3. Principles of Semantic Annotation

A third pillar of the approach to quantification annotation proposed in this paper is formed by the ISO principles of semantic annotation (ISO standard 24617-6; see also (Bunt, 2015) and (Pustejovsky et al., 201), which require an annotation scheme to have a three-part definition consisting of (1) an abstract syntax that specifies the possible annotation structures in set-theoretical terms, such as pairs and triples of concepts; (2) a concrete syntax, that specifies a representation format of annotation structures as XML expressions; (3) a semantics that specifies the meaning of annotation structures. This formal definition is supported by a metamodel that captures the fundamental concepts used in annotations and the way they are related. This organization ensuress that semantic annotations have a semantics, as required by ISO 24617-6 (the principle of 'semantic adequacy'), and by defining the semantics at the level of the abstract syntax it puts the focus of an annotation standard at the conceptual level, rather than at the level of concrete representations, as required by the ISO Linguistic Annotation Framework (ISO 24612, see also (Ide and Romary, 2004). Annotators have to deal with the concrete representations only, but they can rely on the existence of an underlying abstract syntax and semantics, which can be generated automatically from the XML representations.

A systematic specification of the semantics of annotation structures is hardly possible if it cannot be done in a *compositional* way, i.e., the semantic interpretation of an annotation structure is obtained by combining the interpretations of its components. Two kinds of components are distinguished: *'entity structures'* and *'link structures'*. An entity structure specifies certain semantic information about a markable, a link structure specifies certain semantic information about the way two or more entity structures are semantically related. For example, the annotation of an NP corresponds to an entity structure, and scope relations correspond to link structures (see below, Section 5).

3. Related Work

ISO-TimeML (ISO 24617-1) has certain limited provisions for dealing with time-related quantification. For example, a temporal quantifier like *"weekly"* is represented as follows, where "P7D" stands for "period of seven days":

[1]The <entity> element of ISO 24617-4 has been extended here with the attributes @involvement and @definiteness, and the attribute @entityType has been renamed @pred.

(5) <TIMEX3 xml:id="t5" target="#token0" type="SET" value="P7D" quant="EVERY"/>

Here, @quant is one of the attributes of temporal entities, used to indicate that the entity is involved in a quantification. The limitations of this approach for annotating temporal quantification have been discussed by Bunt and Pustejovsky (2010), and improvements have been suggested by Lee and Bunt (2012).

ISO-Space (ISO 24617-7) uses the @quant attribute as well, applying it to spatial entities, and in addition uses the attribute @scopes to specify a scoping relation. The following example, taken from ISO 24617-7:2014, illustrates this:

(6) a. There's a computer$_{se1}$ on$_{ss1}$ every desk$_{se2}$.

b. <spatialEntity id="se1" target="'#token2" form="nom" countable="true" quant="1" scopes="∅"/>
<spatialEntity id="se2" target="#token5" form="nom" pred="desk" countable="true" quant="every" scopes="#se1"/>
<spatialSignal id="ss1" target="#token3" type="dirTop" />
<qsLink id="qsl1" relType="EC" figure="#se1" ground="#se2" trigger="#ss1"/>
<oLink id="ol1" relType="above" figure="#se1" ground="#se2" trigger="#ss1" frameType="intrinsic" referencePt="#se2" projective="false" />

From a semantic point of view, this use of the @scopes attribute is not very satisfactory since the relative scoping of quantifications over different sets of entities is not a local property of one of these quantifications; therefore an annotation such as (6) does not have a compositional semantics. Instead, we propose to use a link structure to represent scope relations among quantifying NPS, which would come down to adding an element like the following:

(7) <scopeLink arg1="#se2" arg2="#se1" relType="wider"/>

Indirectly related to the definition of an annotation scheme for quantification is the Groningen Meaning Bank project (Bos et al., 2017), which is developing a resource consisting of sentences paired with DRSs that represent their meanings. This work cannot be compared directly with the usual kind of annotation work, which associates pieces of semantic information with markables like individual words and small stretches of text, whereas in the Groningen Meaning Bank DRSs are associated with full sentences. It may however be interesting to compare these DRSs with those that come out of the compositional interpretation of annotations as proposed here.

4. Aspects of Quantification

4.1. Restrictors

A natural language quantifier, a noun phrase (NP), consists in its full form of three parts: (1) the head noun; (2) a sequence of determiners; and (3) pre-nominal or post-nominal adjectives, prepositional phrases (PPs), relative clauses and other modifiers of the head noun. The head noun together with its modifiers is called the *restrictor* of the quantifier, and indicates a domain that is considered in the quantification, the *'source domain'*. Quantification in natural language is nearly always restricted to a contextually determined part of the source domain, called the *"reference domain'* or *'context set'* (Westerståhl, 1985). For example, the quantifier *"all the students"* in (8) does not apply to every student, but only to those present at some meeting or performance.

(8) All the students applauded.

The definiteness of an NP is an indication that the domain of a quantification is restricted to a certain reference domain, rather than to its source domain; definiteness is therefore an item of quantification information that should be annotated, as illustrated above in (3).

4.2. Scope

Relative scope is one of the most studied aspects of quantification in natural language – see e.g. (Cooper, 1983); (Kamp and Reyle, 1993; Szabolcsi, 2010); (Ruys and Winter, 2011). Studies of scope have focused almost exclusively on the relative scopes of sets of participants, as in the classical example *"Everybody in this room speaks two languages"*. Not only the relative scoping of sets of quantified *participants* is a semantically important issue, but also the relative scoping of participants and *events*. This is illustrated by the two possible readings of the sentence in (9):

(9) All the students protested

On one reading, each of the students protested, for instance by sending a letter of protest; on another, all students participated in a single protest event, such as a demonstration. (Note that the latter interpretation involves the consideration of events in which multiple participants occupy the same role. Several approaches, such as those of the VerbNet and PropBank frameworks allow only a single occupant for each semantic role; the ISO approach to semantic role annotation (ISO 24617-4), does allow multiple participants with the same semantic role.)
The relative scopes of events and participants can be marked up on the link structure that expresses the participation in a set of events, as shown in (10) with the attribute 'eventScope' added to the <srLink> element.

(10) (Markables: m1="All the students", m2="protested")
<entity xml:id="x1" target="#m1" pred="student" involvement="all" definiteness="def"/>
<event xml:id="e1" target="#m2" pred="protest"/>
<srLink event="#e1" participant="#x1" semRole="agent" eventScope="wide"/>

In logic it is customary to assume that the relative scopes in a sequence of quantifiers are linearly ordered (but see e.g. Hintikka, 1973 on 'branching quantifiers'); in natural language sentences it may happen that two or more quantifications have equal scope, see e.g. (11):

(11) Three breweries supplied five inns

The intended reading here is that in total three breweries supplied in total five inns. In this total-total, or *cumulative* reading (Scha, 1981) the two quantifications have equal scope; the two cardinal determiners both indicate the amount of involvement of the respective reference domains in the predication. This can be represented in annotations as follows:

(12) <scopeLink arg1="#se2 arg2="#se1" relType="equal"/>

The sentence in (13a) has the same syntactic form as the one in (11), but here the intended reading is not cumulative; it is from a report about a tournament of (European) football where teams of boys and teams of girls participated, and whenever a team of boys played against a team of girls, its size would be reduced from 11 to 7. So the two cardinal determiners are indicators not of reference domain involvement or of scoped involvement of subsets of the reference domain, but rather of group size associated with the participation of groups of boys and girls. This can be accounted for in annotations by introducing a "group" value for the @distr attribute of a participation link, as shown in (13b). Semantically, an annotation with such a distributivity will be interpreted as describing a quantification over groups of seven boys and eleven girls.

(13) a. Seven boys played against eleven girls.

b. (Markables: m1="Seven boys", m2="played against", m3="eleven girls")
<entity xml:id="x1" target="#m1" pred="boy"
involvement="7" definiteness="indef"/>
<event xml:id="e1" target="#m2" pred="play"/>
<srLink event="#e1" participant="#x1"
semRole="agent" distr="group"/>
<entity xml:id="x1" target="#m3" pred="girl"
involvement="11" definiteness="indef"/>
<srLink event="#e1" participant="#x2"
semRole="agent" distr="group"/>

4.3. Distributivity

Distributivity comes in an obvious form in the distinction between individual (or 'distributive') and collective participation; the group distribution illustrated by (13) forms another case. Yet another form of distributivity occurs in (14), where the three boys involved did not necessarily do all the carrying either collectively or individually, but where they may have carried some heavy boxes collectively and some lighter boxes individually:

(14) The boys carried all the boxes upstairs.

The quantifications in this sentence over sets of boys and sets of boxes have 'unspecific' distributivity (Bunt, 1985); the sentence just says that all the boxes were *somehow* carried upstairs by the boys. Following (Kamp and Reyle, 1993), we use the notation X^* to designate the set consisting of the members of X and the subsets of X, and P^* to designate the characteristic function of the set X^*, where P is the characteristic function of X. Using moreover the notation R_C to indicate the characteristic function of a reference domain that forms a subset of a source domain with characteristic function R, the interpretation of (14) can be represented in second-order predicate logic as follows:

(15) $\forall x.[\text{box}_C(x) \rightarrow \exists y.\exists e.[\text{boy}_C{}^*(y) \wedge \text{carry-up}(e) \wedge \text{agent}(e,y) \wedge \exists z.[\text{box}_C{}^*(z) \wedge [x{=}z \vee x{\in}z] \wedge \text{theme}(e,z)]]]$

Note that the distributivity of a quantification is not a property of the set of participants in a set of events, but a property of the way of participating. This was already illustrated above by example (1c), assuming that *"the men"* individually had a beer, and collectively carried the piano upstairs. Distributivity should thus be marked up on the participation relation in the drinking and carrying events.

(16) (Markables: m1="The men", m2="had", m3="a beer", m4="carrying" upstairs, m5="the piano")
<entity xml:id="x1" target="#m1" pred="man"
involvement=">1" definiteness="def"/>
<event xml:id="e1" target="#m2" pred="drink"/>
<entity xml:id="x2" target="#m3" pred="beer"
involvement="1" definiteness="indef"/>
<event xml:id="e2" target="#m4" pred="carry"/>
<entity xml:id="x3" target="#m5" pred="piano"
involvement="1" definiteness="def"/>
<srLink event="#e1" participant="#x1" semRole="agent"
distr="individual"/>
<srLink event="#e2" participant="#x1" semRole="agent"
distr="collective"/>

4.4. Involvement and size/cardinality

The prenominal part of an NP may contain a sequence of determiners of different type. Grammars commonly distinguish different classes of determiners, with different possible sequencing and co-occurrence restrictions. For example, in English grammar it is customary to distinguish between predeterminers, central determiners, and postdeterminers (e.g. Quirk et al., 1972), each of which has a different function:

- predeterminers express the (absolute or proportional) quantitative involvement of the reference domain, and may, additionally, provide information about the distribution of a quantification over the reference domain;
- central determiners determine the definiteness of the NP, and thus co-determine a reference domain;
- postdeterminers contain information about the cardinality of the reference domain (for count NPs) or its size (for mass NPs).

This is illustrated by the NP *"All my nine grandchildren"*, where *"all"* is a predeterminer, *"my"* a central determiner, and *"nine"* a postdeterminer. Note also that the value of an @involvement attribute in an entity structure that has widest scope represents the total involvement of the reference domain, but in an entity structure within the scope of another, it represents a 'scoped' involvement, like the determiner *"two"* in *"Everybody in this room speaks two languages"*.

involv.	*distributivity*	*interpretation*	*example*
all	homogeneous	For all quantities of M	(21a)
total	unspecific	For the elements in a set of quantities of M that together make up the whole of M	(21c)
all	collective	For M as a whole	(21b)

Table 1: Involvement and distributivity in mass NP quantification.

4.5. Quantification in modifications

The restrictor part in a full-fledged NP may be simply a noun, but in general may contain adjectives and other expressions that modify the noun, such as other nouns, as in *"waste dump"*, prepositional phrases, or relative clauses. Moreover, conjunctions of nouns (possibly with modifications) may add further complexity to restrictors.

Head noun modifications bring certain issues of quantification, such as scope and distributivity, e.g. the restrictor *"heavy books"* in the sentence *"Peter carried some heavy books"* may be interpreted as referring to certain books that are heavy each (individual reading) or to a heavy pile of books (collective reading). To express this in annotations, an <adNLink> structure is introduced with the attribute @distr:

(17) (Markables: m1="heavy", "books")
<entity id="x1" target="#m2" pred="book"/>
<entity id="x2" target=#m1 pred="heavy"/>
<adNLink head="#x1" mod="#x2" distr="collective"/>

An adjective can be used not only for modifying an NP head noun, but also for modifying a noun that modifies another noun, as in *"(toxic waste) dump"* or *"(natural language) processing"*. The QuantML representation (18) shows how the adjective scope in *"(toxic waste) dump"* can be indicated (see also example (22)).

(18) (Markables: m1="toxic", m2="toxic waste", m3="toxic waste dump", m4="waste", m5="dump")
<qDomain xml:id="x1" target="#m3"
source="#x2" restrictions="#r1"/>
<sourceDomain xml:id="x2" target="#m5"
pred="dump" />
<nnLink xml:id="r1" target="#m2" pred="waste"
restr="#r2" />
<adNLink xml:id="r2" target="#m!" pred="toxic"
distr="individual"/>

Quantifier scope issues arise when a head noun is modified by a PP or a restrictive clause, as in (19), where a quantification inside the PP takes scope over the one of the head noun.

(19) The committee spoke with two students from every university.

This phenomenon is known as 'inverse linking' (May, 1977; May and Bale, 2005; Szabolcsi, 2010; Ruys and Winter, 2011; Barker, 2014). Inverse linking in PP modification is widespread; especially the case of an existentially quantified main NP and universally quantified PP, as in (19), is quite common.

To capture the relevant information related to quantification within a complex restrictor, the annotation of complex restrictors needs to be articulated in marking up the head noun that is central to the restrictor, and the various possible modifiers with indications of their distributivity and possible scope inversion.

4.6. Mass Quantifiers

Most studies of quantification in natural language have been restricted to cases where the NP head is a count noun. Quantification by NPs where the head is a mass noun are in many ways similar; compare, for example for example, (20a) and (20b) :

(20) a. The salesmen sold all the cars.
b. The vendors sold all the ice cream.

In (20a) a predicate is applied to sets of salesmen and cars, and in (20b) to sets of vendors and quantities of ice cream. A difference is that (20a) can be analysed as: "Every one of the cars was the object in a sell-event with one of the salesmen as the agent", but it is not clear that the analogous analysis where every quantity of ice cream was the object in a sell-event would make sense. A universal quantification like *"all the ice cream"* does not necessarily refer to *all* the quantities of ice cream that the vendors possessed, but rather to a certain subset of quantities that has the property of together making up the whole of the vendors' ice cream. Such a situation commonly arises for mass NP quantifiers, and may arise also for count NPs in case the individuals in the quantification domain have an internal part-whole structure, as in *"The boys ate all the pizzas"*.

A detailed analysis of quantification in relation to mass terms can be found in (Bunt, 1985), whick analyses the notion 'quantity of' as a part-whole relation, and defines a 'merge' operation $\cup$ on quantities such that the merge of two or more quantities of M is again a quantity of M. An expression of the form 'all the M' with a mass noun M can be interpreted as referring to a set of quantities of M whose merge forms the whole of all M.

Properties of mass quantification, like those of count NP quantification, are distributivity, scope, definiteness, domain involvement, and size of the reference domain (or of parts of it), but there are some notable differences in distributivity and in the expression of involvement and size.

Since mass nouns do not individuate their reference, quantification by mass NPs would seem not to allow individual distribution. Yet there is a distinction somewhat similar to the individual/collective distinction of count NP quantifiers, as illustrated by (21).

(21) a. The water in these lakes is polluted.
b. The sand in the truck weighs 20 tons.
c. The boys carried all the sand to the back yard.

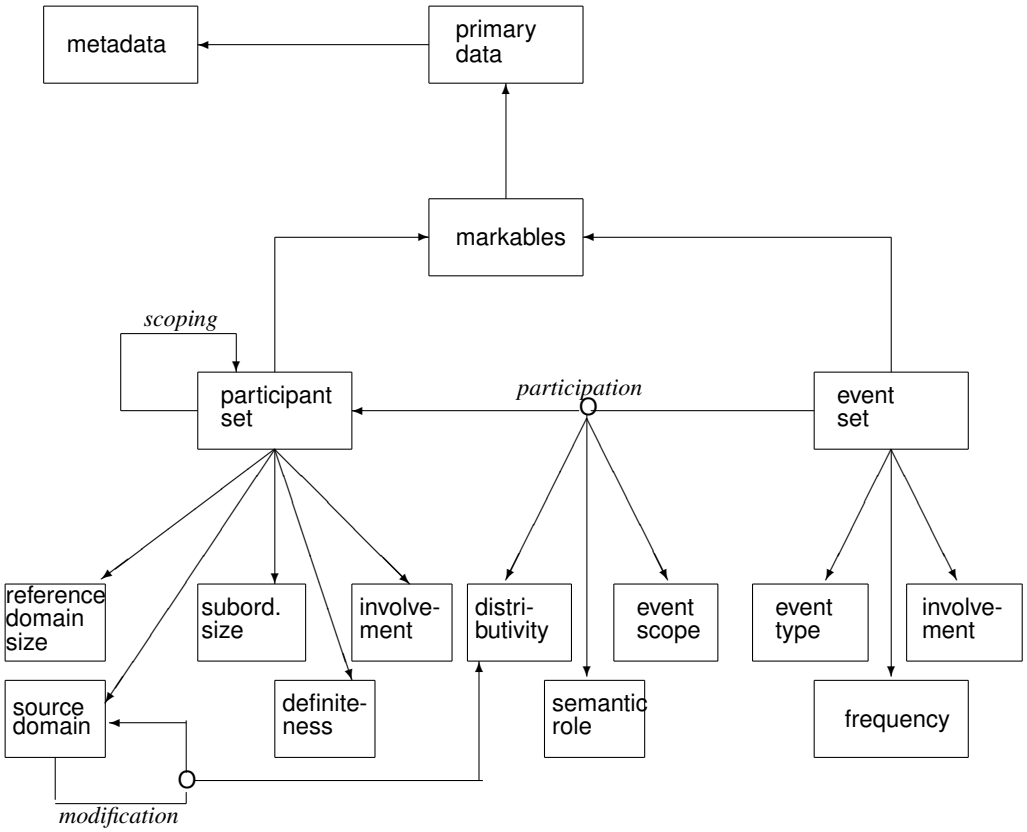


Figure 1: Metamodel for the annotation of quantification

The quantification in (21b) is collective, as the weight applies to the whole formed by all the quantities of sand. In (21a) the predicate of being polluted applies to any sample of *"the water in the lake"*; this distribution is called *homogeneous*. In (21c) the boys carried certain quantities of sand that together make up "all the sand"; in this case the distribution is called *unspecific*, as this case is rather similar to the count NP case where both individuals, sets of individuals, and parts of individuals may be involved.

Expressions of proportional involvement, like *"some pasta", "most of the pasta","all the pasta"* cannot be interpreted in terms of number of quantities. As the examples in (21) illustrate, complete involvement of a mass NP reference domain means that the merge of the quantities involved forms the entire domain: if Q_M is the set of all quantities of M, and X_M is the set of quantities of M involved in the quantifying predication, then $\cup X_M = \cup Q_M$. Non-zero involvement means that the merge of the quantities in M has non-zero size, and a *"most M"* quantification over reference domain M means that, for any unit of measurement u, $|\cup X|_u > |M|_u/2$, where $|.|_u$ indicates size measured in units u. (See (Bunt, 1985) for a calculus of size measurement.)

The examples in (21) illustrate three different ways in which the quantification domain of a mass NP can be 'completely' involved in a predication, corresponding to three different senses of expressions of the form *(all) the M* in English, and similarly in other languages. In (21a), *"The water"* refers to the set of all (contextually relevant) quantities of water; this involvement will be indicated in annotations as *"all"*. In (21c), *"all the sand"* refers to a subset of quantities of sand that together make up all the (contextually relevant) sand; this involvement is indicated as "total" in annotations of reference domain involvement. Finally, in (21b), *"The sand"* refers to the single quantity of sand formed by all contextually relevant quantities of sand. This involvement will be annotated as "all", just as in the case of collective count NP quantification. Table 1 summarizes these possible annotation choices.

5. QuantML Annotation Scheme

5.1. ISO scheme organization

An ISO standard annotation scheme for quantification should fit within the series of semantic annotation standards known collectively as the Semantic Annotation Framework (SemAF), ISO 24617. It should as such be compatible with the existing parts of SemAF: Part 1, Time and events; Part 2, Dialogue acts; Part 4: Semantic roles; Part 7: Spatial information, and Part 8: Discourse relations. Moreover, it should be defined according to the ISO Principles for semantic annotation, as mentioned in Section 2.3. This means in particular that the definition of the annotation scheme includes an abstract syntax, a concrete syntax, and a semantics, supported by a conceptual metamodel. Figure 1 shows the proposed metamodel, reflecting the conceptual analysis of quantification in the previous sections.

Specifying the semantics of annotations at the level of abstract syntax ensures that changes in representation format do not need to require adaptations in the semantics. The ISO principles require representation formats to be 'ideal', i.e. (1) complete, in the sense that every annotation structure defined by the abstract syntax has a representation; (2) unambiguous, i.e. every representation is a rendering of only one annotation structure defined by the abstract syntax

(22) a. Alex owns some (valuable ancient (Chinese books and Japanese paintings)).

b. Markables: m1=Alex, m2=owns, m3=some valuable ancient Chinese books and Japanese paintings, m4= valuable, m5=valuable ancient Chinese books and paintings, m6=ancient, m7=Chinese, m8=Chinese books, m9=books, m10=Japanese, m11=Japanese paintings, m12=paintings

c. QuantML Representation:

<entity xml:id="x1" target="#m1" domain="#x1" involvement="1" definiteness="def"/>
<sourceDomain xml:id="x1" target="#m1" pred="alex"/>
<event xml:id="e1" target="#m2" pred="own"/>
<entity xml:id="x2" target="#m3" domain="#x3" involvement="some" definiteness="indef"?/>
<qDomain xml:id="x3" target="#m5" source="#x4 #x6" restrictions="#r1 #r2?"/>
<qDomain xml:id="x4" target="#m8" source="#x5" restrictions="#r3"/>
<sourceDomain xml:id="x5" target="#m9" pred="book"/>
<qDomain xml:id="x6" target="#m11" source="#x7" restrictions="#r4"/>
<sourceDomain xml:id="x7" target="#m12" pred="painting"/>
<adNLink xml:id="r1" target="#m4" pred="valuable" distr="individual"/>
<adNLink xml:id="r2" target="#m6" pred="ancient" distr="individual"/>
<adNLink xml:id="r3" target="#m7" pred="chinese" distr="individual"/>
<adNLink xml:id="r4" target="#m10" pred="japanese" distr="individual"/>
<participationLink event="#e1" participant="#x1" semRole="theme" distr="individual" eventScope="narrow"/>

(see Bunt, 2010). This approach supports the design of alternative user-friendly representations, allowing for example to use tabular forms or other formats that are more convenient for human annotators and researchers than XML representations. This has been exploited in the DialogBank (Bunt et al., 2016), a resource of dialogues annotated according to the ISO 24617-2 annotation scheme, with alternative representations and the possibility to convert from one representation to another.

5.2. QuantML Abstract Syntax, Concrete Syntax, and Semantics

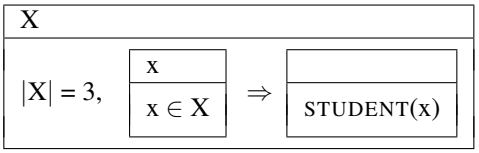
The annotation structures defined by the QuantML abstract syntax consist of entity structures and link structures. The concrete syntax specifies a pivot XML format for representing these structures. The examples earlier in this paper gave a slightly simplified impression of these representations. The example in Fig. 2 provides more detail in case the quantifier has a structured source domain, caused by the NP head being a modified conjunction of two modified nouns. (The <srLink> used in the examples above has been replaced here by the QuantML element <participationLink>.) The semantics specifies an interpretation-by-translation by means of a compositional interpretation function that defines DRSs for all well-formed annotation structures; compositional in the sense that the DRS expressing the meaning of an annotation structure is constructed out of the DRSs interpreting the component entity and link structures.

Example (23) shows the abstract entity structure for the quantifier *"Three students"* with its concrete XML representation and its semantics. The abstract annotation structure pairs the markable for an occurrence of the expression *"three students"* with a quadruple of concepts as shown in (23a); a concrete representation in XML may look as in (23b); and the semantic interpretation as shown in (23c), obtained by applying the interpretation function as partly defined by (24), where 'P' stands for the characteristic predicate of a source domain, and N is a predicate that expresses reference domain involvement.

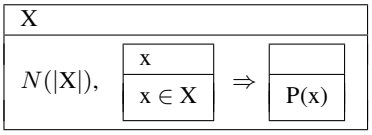
(23) a. $\langle$man, some, indef, $\lambda z.|z| = 3\rangle$

b. <entity xml:id="x1" target="#m1"
type="student" involvement="3"/>

c.



(24) $I_A(\langle$P, 3, indef, $C\rangle)$ =



6. Conclusions and Further Work

In this paper we have outlined a theoretically well-founded approach to the annotation of quantification in natural language which observes the most important requirements for establishing an ISO annotation standard. These requirements concern:

1. theoretical adequacy, taking well-established results of studies in logic, linguistics, and computation into account (notably those of Generalized Quantifier Theory, of neo-Davidsonian event-based semantics, and the ISO Principles of semantic annotation);
2. empirical adequacy, providing a wide coverage of quantification phenomena;
3. semantic adequacy, in specifying a well-defined semantics of annotations;
4. compliance with the ISO Linguistic Annotation Framework[2] and the ISO Principles of semantic anno-

[2]ISO 24612; cf. Ide and Romary, 2004

tation[3], resulting in the separation of the abstract and concrete syntax of annotations, with a compositional semantics that provides interpretations in the form of DRSs for the structures of the abstract syntax;

5. compatibility with existing standards for the semantic annotation of time and events, spatial information, and semantic roles.

The paper specifies a metamodel that specifies a number of concepts relating to properties of quantification in natural language (including quantification phenomena within head noun modifications) that have to be taken into account in an adequate annotation scheme, including those that occur in noun modification structures with quantifier restrictors, such as the distributivity of adjectival modification and the relative scoping of quantifiers when inverse linking occurs in modification by preposition phrases.

There is still work to be done: further development of he approach outlined here will try to incorporate solutions for some of the well-known semantically hard puzzles relating to quantification, such as the treatment of reciprocals, reflexives, generics and habituals.

7. Bibliographical References

Barker, C. (2014). Scope. In S. Lappin and C. Fox, editors, *The Handbook of Contemporary Semantic Theory*, pages 40–76. John Wiley.

Barwise, J. and Cooper, R. (1981). Generalized quantifiers and natural language. *Linguistics and Philosophy*, pages 159–219.

Bos, J., Basile, V., Evang, K., Venhuizen, N., and Bjerva, J. (2017). The Groningen Meaning Bank. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 463–496. Springer, Berlin.

Bunt, H. and Pustejovsky, J. (2010). Annotating temporal and event quantification. In *Proceedings ISA-5, Fifth International Workshop on Interoperable Semantic Annotation*, pages 15–22. City University of Hong Kong.

Bunt, H., Petukhova, V., Malchanau, A., Fang, A., and Wijnhoven, K. (2016). The DialogBank. In *Proceedings 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz.

Bunt, H. (1985). *Mass terms and model-theoretic semantics*. Cambridge University Press, Cambridge, UK.

Bunt, H. (2010). A methodology for designing semantic annotation languages exploiting syntactic-semantic isomorphisms. In Alex Fang, Nancy Ide, and Jonathan Webster, editors, *Proceedings of ICGL 2010, Second International Conference on Global Interoperability for Language Resources*, pages 29–45. City University of Hong Kong.

Bunt, H. (2015). On the principles of semantic annotation. In *Proceedings 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-11)*, London.

Cooper, R. (1983). *Quantification and syntactic theory*. Reidel, Dordrecht.

Davidson, D. (1967). The logical form of action sentences. In N. Rescher, editor, *The Logic of Decision and Action*. University of Pittsburgh Press.

Hintikka, J. (1973). Quantifiers vs. quantification theory. *Dialectica*, 27:329–358.

Ide, N. and Romary, L. (2004). International standard for a Linguistic Annotation Framework. *Natural Language Engineering*, 10:221–225.

ISO. (2010). *ISO 24612: Language resource management: Linguistic Annotation Framework (LAF)*. International Organisation for Standardisation ISO, Geneva.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht.

Lee, K. and Bunt, H. (2012). Counting time and events. In *Proceedings 8th Joint ISO-ACL Workshop on Interoperable Semantic Annotation (ISA-8)*. ILC-CNR, Pisa.

Lindström, P. (1966). First order predicate logic with generalized quantifiers. *Theoria*, 32:186–195.

May, R. and Bale, A. (2005). Inverse linking. In Martin Everaert and Henk van Riemsdijk, editors, *The Blackwell Companion to syntax, Vol. 2*, pages 639–667. Blackwell, Oxford.

May, R. (1977). *The Grammar of Quantification*. Ph.D. Dissertation, MIT.

Mostowski, A. (1957). On a generalization of quantifiers. *Fundamentae Mathematicae*, 44:12–36.

Parsons, T. (1990). *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press, Cambridge, MA.

Partee, B. (1988). Many quantifiers. In *ESCOL 89: Proceedings of the Eastern States Conference on Linguistics, Reprinted in Compositionality in Formal Semantics: Selected Papers by Barbara Partee*, pages 241–158, Oxford. Blackwell 2004.

Pustejovsky, J. and Bunt, H., and Zaenen, A. (201). Designing annotation schemes: From theory to model. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*. Springer, Berlin.

Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. (1972). *A grammar of contemporary English*. Longman, London.

Ruys, E. and Winter, Y. (2011). Scope ambiguities in formal linguistics. In D. Gabbay et al., editors, *Handbook of Philosophical Logic*. Foris, Dordrecht.

Scha, R. (1981). Collective, distributive and cumulative quantification. In J. Groenendijk et al., editors, *Formal Methods in the Study of Language*, pages 483–512. Mathematisch Centrum, Amsterdam.

Szabolcsi, A. (2010). *Quantification*. Cambridge University Press, Cambridge, UK.

Westerståhl, D. (1985). Determiners and context sets. In Johan van Benthem and Alice Ter Meulen, editors, *Generalized Quantifiers in Natural Language*, pages 45–71. Foris, Dordrecht.

[3]ISO 24617-6; cf. Bunt, 2015

Lightweight Grammatical Annotation in the TEI: New Perspectives

Piotr Bański[1], Susanne Haaf[2], Martin Mueller[3]
[1]Institute for the German Language, Mannheim (Germany)
[2]Berlin-Brandenburg Academy of Sciences and Humanities, Berlin (Germany)
[3]Northwestern University, Chicago (USA)
[1]banski@ids-mannheim.de, [2]haaf@bbaw.de, [3]martinmueller@northwestern.edu

Abstract

In mid-2017, as part of our activities within the TEI Special Interest Group for Linguists (LingSIG), we submitted to the TEI Technical Council a proposal for a new attribute class that would gather attributes facilitating simple token-level linguistic annotation. With this proposal, we addressed community feedback complaining about the lack of a specific tagset for lightweight linguistic annotation within the TEI. Apart from `@lemma` and `@lemmaRef`, up till now TEI encoders could only resort to using the generic attribute `@ana` for inline linguistic annotation, or to the quite complex system of feature structures for robust linguistic annotation, the latter requiring relatively complex processing even for the most basic types of linguistic features. As a result, there now exists a small set of basic descriptive devices which have been made available at the cost of only very small changes to the TEI tagset. The merit of a predefined TEI tagset for lightweight linguistic annotation is the homogeneity of tagging and thus better interoperability of simple linguistic resources encoded in the TEI. The present paper introduces the new attributes, makes a case for one more addition, and presents the advantages of the new system over the legacy TEI solutions.

Keywords: linguistic annotation, lightweight annotation, TEI, TEI LingSIG

1. Introduction

In July 2017, as part of our activities within the TEI Special Interest Group for Linguists (LingSIG), we submitted to the TEI Technical Council a proposal for the definition of a new attribute class that gathers token-level attributes facilitating simple linguistic annotation. With this proposal, we addressed repeated requests from various corpus projects to facilitate combined annotation of customary TEI text structures and basic token-level linguistic features. Most of our proposed modifications have been accepted and are part of the new release of the TEI Guidelines (3.3.0, published in January 2018).[1] The present paper is therefore partially a report of success and a review of new features that the encoder has at her disposal, and in part, it provides arguments for extending the descriptive power of the Guidelines even further, towards supporting a complete chain of token-level grammatical analysis in various kinds of text collections.

In the out-of-the-box TEI markup, it is now possible to use the following attributes of the elements `<w>` (« word ») and `<pc>` (« punctuation character »):

- `@pos` (for part-of-speech)
- `@msd` (for morpho-syntactic description)
- `@join` (to signal string concatenation with a neighbouring element)

These attributes – together with `@lemma` and `@lemmaRef`, previously defined inside `<w>` – are now encapsulated in a new attribute class, *att.linguistic*, supplying means crucial for basic grammatical annotation at the token level, i.e., for lightweight grammatical annotation.[2]

In the sections to follow, we provide a description of the new attributes and a discussion of the alternatives that have been in use up till now. A further, argumentative part of the present paper focuses on an additional attribute that facilitates the encoding of historical corpora and literary collections, namely on `@norm`, encoding normalized/ regularized forms.

2. TEI and the TEI Guidelines

« TEI » stands for « Text Encoding Initiative », a consortium of institutions and individuals aiming at developing guidelines for consistent and explicit encoding of a wide array of textual types. The TEI Guidelines are freely available at http://www.tei-c.org/, in the form of prose, documented schemas, and ready-to-use customizations, together with various tools. This section provides a brief overview of the TEI Guidelines essential for contextualizing the rest of the paper.

The TEI Guidelines are encoded in, and customized by, a TEI-based specialized literate markup language called ODD.[3] ODD allows for the definition of TEI-specific constructs (modules, element classes, attribute classes) which can be combined in various ways in order to form descriptive apparatus for a variety of phenomena encountered in Digital Humanities' research. ODD combines such definitions with potentially very extensive documentation (the most extreme form of which is found in the multi-chapter prose of the Guidelines themselves). Apart from tools that can assist the text modeller by manipulating ODD documents in order to produce customized schemas (the most well-known such tool is Roma), encoders may use so-called TEI Extensions which are essentially out-of-the-box customizations designed for particular research areas.

[1] The changes were merged in a pull request which also references detailed discussion, see https://github.com/TEIC/TEI/pull/1671 and issue #1670.

[2] We enclose element names in angle brackets and prepend attribute names with a '@'.

[3] See http://www.tei-c.org/Guidelines/Customization/odds.xml and https://wiki.tei-c.org/index.php/ODD.

Crucially for the proposal presented here, the TEI data model includes constructs known as element classes and attribute classes. Element classes are sets of TEI elements that are found in similar structural contexts. Attribute classes group attributes that have something in common in terms of features modelled in a particular domain. When an element is a member of an attribute class, it can use all the attributes found inside that class.

Our focus here is on the elements <w> and <pc> that are members of the model.segLike element class and are specialized for the description of linguistic tokens and punctuation characters, respectively. Each of these elements is a member of several attribute classes. An element can also, in principle, define attributes that are specific to it and ideally not needed anywhere else – that is currently the case of <pc>, and it was the case of <w>, which used to define @lemma and @lemmaRef before the changes described here came into effect.[4]

The present paper is centred around the newly added attribute class *att.linguistic*. Postulating a new attribute class is a theoretical statement as well as a practical move. From the theoretical point of view, attributes contained in a single class should have something in common: in our case, they are necessary for a reasonably flexible description of basic grammatical features of tokens. From the practical perspective, defining a single class of attributes that function together means that the set of properties that it provides can be made available to other elements, if needed, via ODD customization.

3. Grammatical Annotation in the TEI prior to the *att.linguistic* class

Enrichment of TEI-annotated text with even the most basic grammatical information (part-of-speech and morphosyntactic information in addition to lemma identification) drastically expands search options, options for sorting the results, or for investigating author-specific traits. It also facilitates the combined analysis of token-based linguistic information with specific TEI structures, e.g. investigation of adjective usage in headlines, etc. (see e.g. Schöch 2016, Haaf 2016). Over the years, many robust structural solutions have been suggested, but no single standardized approach for lightweight annotation at the token level has emerged. This section looks briefly at the solutions suggested so far and shows why some of them are not optimal.

3.1 Hierarchical solutions

The TEI possesses tools for complex description of linguistic structures by means of element hierarchies, among others by exploiting the extremely powerful mechanism of feature structures, defined in a joint ISO-TEI standard (ISO 24610-1, TEI Consortium, 2018, Ch. 18 ; see fig. 1 below for an illustration). Indeed, as shown by Stegmann and Witt (2009), it is conceivable, even if not practical, to model an entire linguistic corpus as a complex feature matrix. Typically, feature structures are used for modelling local bundles of features, grammatical and other, that would otherwise not fit into the format prescribed by the TEI. They are a very handy descriptive device in all formats that use the stand-off approach, where annotations are not part of the (sub)document containing annotated text.[5]

An example of a feature structure that at the same time illustrates much of the functionality that *att.linguistic* now provides, comes from the National Corpus of Polish[6] and represents a set of potential interpretations of the adjectival form *kategoryczne* (« categorical » ; fig. 1). In the example, the feature « base » has the lemma as its value, and « ctag » encodes the part-of-speech. The feature « msd » lists all possible morphological interpretations of the token (the first option encodes the features : plural, nominative, animate masculine, and positive).[7] Another feature, not shown in the example, is used to point at the value(s) disambiguated in the given morphosyntactic context.

```
<f name="interps">
 <fs type="lex">
  <f name="base">
   <string>kategoryczny</string>
  </f>
  <f name="ctag">
   <symbol value="adj"/>
  </f>
  <f name="msd">
   <vAlt>
    <symbol value="pl:nom:m2:pos"/>
    <symbol value="pl:nom:m3:pos"/>
    <symbol value="pl:nom:f:pos"/>
    <symbol value="sg:nom:n:pos"/>
    <symbol value="pl:nom:n:pos"/>
    <symbol value="pl:acc:m2:pos"/>
    <symbol value="pl:acc:m3:pos"/>
    <symbol value="pl:acc:f:pos"/>
    <symbol value="sg:acc:n:pos"/>
    <symbol value="pl:acc:n:pos"/>
    <symbol value="pl:voc:m2:pos"/>
    <symbol value="pl:voc:m3:pos"/>
    <symbol value="pl:voc:f:pos"/>
    <symbol value="sg:voc:n:pos"/>
    <symbol value="pl:voc:n:pos"/>
   </vAlt>
  </f>
 </fs>
</f>
```

Figure 1. National Corpus of Polish, file « ann_morphosyntax.xml », with ID attributes stripped off for the sake of conciseness

Another way to use feature structures is by defining feature libraries and feature-value libraries, where individual elements can be referenced by the attribute

[4] For <w> see the penultimate release: http://www.tei-c.org/Vault/P5/3.2.0/doc/tei-p5-doc/en/html/ref-w.html; for <pc> see: http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-pc.html.

[5] For references and remarks on TEI standoff techniques, see a.o. Bański, 2010, Pose et al., 2014, and Bański et al., 2016, as well as https://github.com/laurentromary/stdfSpec.

[6] See http://nkjp.pl/ for an entry point and information, and http://nlp.ipipan.waw.pl/TEI4NKJP/ for ODD and examples.

[7] See the tagset documentation for other categories and values: http://nkjp.pl/poliqarp/help/ense2.html.

@ana, as in fig. 2, which shows an example copied from Budin et al., 2012:11.

```
<fLib>
 <f xml:id="pos.verb" name="pos">
  <symbol value="verb"/> </f>
 ...
 <f xml:id="tns.pres" name="tense">
  <symbol value="present"/> </f>
 ...
 <f xml:id="mood.ind" name="mood">
  <symbol value="indicative"/> </f>
 ...
 <f xml:id="num.pl" name="number">
  <symbol value="plural"/> </f>
 ...
 <f xml:id="pers.1" name="person">
  <symbol value="1"/> </f>
 ...</fLib>

<fvLib> ...
 <fs   xml:id="v_pres_ind_sg_p2"
       name="v_pres_ind_sg_p2"
       feats="#pos.verb #tns.pres #mood.ind
              #num.pl #pers.2">
 ...</fvLib>

<form  type="inflected"
       ana="#v_pres_ind_pl_p1
              #v_pres_ind_pl_p3 ">
 <orth>gehen</orth>
</form>
```

Figure 2. <fLib> contains simple attribute-value pairings, while <fvLib> can be used to create complex values for re-use (multiple references to single pairings are grouped into bundles). Both simple and complex features can then be referenced, from any language resource, by means of the @ana attribute. (Copied from Budin et al., 2012:11)

A decision to use such complex hierarchical element-based devices for grammatical description indicates a commitment in terms of various resources: manpower, time, finances needed to create and/or customize and maintain specialized tools capable of interpreting such robust structures. In practice, this kind of commitment is not always possible or needed. A tokenized corpus where element hierarchy is relatively simple and where grammatical features are bundled inside word-sized textual segments, is able to support or reject many linguistic hypotheses at a much lower cost than a robust resource would incur. Similar observations are true of an average case of literary encoding, to which grammatical information gets added for the purpose of enhancing searches or for basic measurements – it can be added as a separate document, i.e. in a stand-off manner, but in many cases it is much simpler and cheaper to add the relevant attributes to the individual <w>-sized segments.

3.2 Solutions for lightweight annotation before TEI version 3.3.0

Before the initiative described here, for the purpose of encoding the results of simple grammatical analysis, the TEI encoder had to resort to non-standardized devices, essentially to using semantically unspecified attributes to carry linguistic description. The primary candidates in this context were @ana, @corresp and @type, where:

- @ana: "indicates one or more elements containing interpretations of the element on which the @ana attribute appears" (TEI 2018: att.global.analytic)
- @corresp: "points to elements that correspond to the current element in some way" (TEI 2018: att.global.linking)
- @type: "characterizes the element in some sense, using any convenient classification scheme or typology" (TEI 2018: att.typed)

Of this attribute set, @ana and @corresp are pointer-based, while @type can hold sequences of whitespace-delimited tokens. Fig. 3 illustrates the use of @ana with an actual example from the Guidelines.

```
<s>
 <w ana="#AT0">The</w>
 <w ana="#NN1">victim</w>
 <w ana="#POS">'s</w>
 <w ana="#NN2">friends</w>
 <w ana="#VVD">told</w>
 <w ana="#NN2">police</w>
 <w ana="#CJT">that</w>
 [...]
<s>
```

Figure 3. Partial example of using @ana advocated by the Guidelines (TEI 2018:17.4)

Our main arguments against the use of the above-mentioned attributes for lightweight markup are based on the notions of simplicity and practicality. Firstly, a large part of our target group, namely corpus linguists and creators of language resources, need a straightforward way to serialize the output of analysis tools, using well-established labels, concepts, and datatypes. An approach using @ana or @corresp would essentially involve creating pseudo-URIs out of the labels produced by morphological analyzers, only to pre-process those URIs for querying and visualisation in order to convert them back to simple labels.

The other major part of our target group, namely the creators and curators of resources for other disciplines (e.g., literary and historical text collections), frequently use any of the three attributes, in any combination, for the purpose of domain-specific text analysis. When these resources become subject to enrichment with linguistic markup, there will be no way to guarantee a uniform choice of containers for grammatical information (or, in the case of @ana and @corresp with added pseudo-URIs, a uniform structuring of values), unless the *att.linguistic* class can be used for this purpose. Reserving some of the generic attributes for linguistic purposes

would mean (a) removing them from the general pool of attributes available for non-linguistic uses and (b) excluding some of the legacy non-linguistic resources that already use those attributes.

On the basis of the above considerations and being aware that identification of the lemma, the POS and the morphosyntactic features are the most basic requirements of linguistic or linguistically-informed analysis at any level of processing, we decided to propose analytical attributes specifically fitted for the linguistic domain and thus to separate linguistic token-based annotation from that used in other domains.

4. Description of the *att.linguistic* class

One notorious point of criticism concerning the TEI is that it 'allows for too much'. This is apparently based on an expectation that there should be a single way to encode any given kind of textual phenomena. However, anyone with moderate awareness of the richness of modern day Digital Humanities will know that there is definitely no single way to approach the different kinds of data, information needs, visualisation requirements and the varying foci of interest of Digital Humanities' scholars. The TEI is a toolkit from which a researcher needs to define and document a particular customization, given the plethora of options on offer. For a "tinkerer", the TEI is a nearly endless collection of parts from which numerous schemas can be created. Not every researcher wants to be a tinkerer, however – some would prefer to be end-users of predesigned solutions and to have at their disposal a ready-made standardized format, which only needs to be filled in with text and annotated values. The solution described here adds low-level devices from which a tinkerer can choose for the purpose of enriching already existing schemas but which at the same time can straightforwardly be used by a corpus linguist planning to create a new digital resource consisting of only crudely structured tokenized text with the basic linguistic features – which now have clearly labelled containers in the form of *att.linguistic* attributes.

An example applying the crucial *att.linguistic* components follows below.

```
<s>
 <w pos="PPER" msd="pers:subst:p1:nom:pl"
       lemma="wir">Wir</w>
 <w pos="VVFIN" msd="p1:pl:pres:ind"
       lemma="fahren">fahren</w>
 <w pos="APPR" msd="--" lemma="in">in</w>
 <w pos="ART" msd="def:acc:sing:masc"
       lemma="d">den</w>
 <w pos="NN" msd="acc:sing:masc"
       lemma="Urlaub">Urlaub</w>
  <pc pos="$." msd="--" lemma=".">.</pc>
</s>
```

Figure 4. The results of an analysis of the sentence « Wir fahren in den Urlaub » encoded by means of *att.linguistic*. Source of analysis: WebLicht (2018)

The example above contains the following attributes:

- `@pos`: « (part of speech) indicates the part of speech assigned to a token, usually according to some official reference vocabulary (e.g. for German: STTS, for English: CLAWS, for Polish: NKJP, etc.) » (TEI 2018 : att.linguistic)
- `@msd`: « (morphosyntactic description) supplies morphosyntactic information for a token, usually according to some official reference vocabulary (e.g. for German: STTS-large tagset, for Polish, NKJP) »[8] (TEI 2018 : att.linguistic)
- `@lemma`: « provides a lemma (base form) for the word, typically uninflected and serving both as an identifier (e.g. in dictionary contexts, as a headword), and as a basis for potential inflections. » (TEI 2018 : att.linguistic)

Note that due to various compromises that have to be made between linguistic description and technological efficiency, it is not unnatural to expect projects to use only one of `@pos` and `@msd` for storing complex information, or to use them redundantly, e.g. with `@pos` containing part-of-speech symbols extracted from composite morphosyntactic labels stored inside `@msd`. It often happens, especially in languages with impoverished inflection, that morphosyntactic categories are merged into parts of speech (this is partially responsible for the difference between, e.g., CLAWS-5 and CLAWS-8). It is expected that each project will document the particular grammatical annotation practices in the corpus header.

In order to illustrate the next member of *att.linguistic*, we repeat the example sentence from fig. 4 in a somewhat different arrangement, and with the attributes removed, for the sake of clarity (fig. 5).

```
<s><w>Wir</w> <w>fahren</w> <w>in</w>
<w>den</w> <w>Urlaub</w><pc>.</pc></s>
```

Figure 5. Tagging of example sentence from fig. 4 as inline representation

If we contrast the inline representation in fig. 5 with the sequential representation in fig. 4, it becomes clear that the example in fig. 5 provides more information, because it uses whitespace as additional typographical markup. The representation in fig. 4 only lists the tokens according to their order in the sentence, but loses the information on the lack (or the presence) of the neighbouring whitespace. In order to preserve this kind of information, the `@join` attribute should be used. With this attribute, the final fragment of the sequential example looks as illustrated in fig. 6.

An issue may arise concerning the redundancy of marking the absence of whitespace on two neighbouring elements. From the top-down, global perspective, it is indeed redundant. From the bottom-up, 'streamable' perspective, it is not redundant, and we assume that decisions on which stance to adopt are going to be project-specific. The TEI

[8] For a feature description system designed as (pragmatically) universal for use with Universal Dependencies, see http://universaldependencies.org/u/feat/index.html; for the corresponding system of parts of speech see http://universaldependencies.org/u/pos/index.html.

provides support for the general, redundant, 'streamable' case, of which project-specific decisions can be subsets.

```
<s>
    <w>Wir</w>
    <w>fahren</w>
    <w>in</w>
    <w>den</w>
    <w join="right">Urlaub</w>
    <pc join="left">.</pc>
</s>
```

Figure 6. Example sentence from fig. 4 in the sequential arrangement, with redundant (=streamable) use of `@join`.

The last attribute of att.linguistic is `@lemmaRef`, previously defined directly inside `<w>`. It provides a pointer to a definition of the lemma for the word, e.g. in an online lexicon, as illustrated in fig. 7 (copied from the Guidelines).

```
<w type="verb" lemma="hit"
lemmaRef="http://www.example.com/lexicon/hi
tvb.xml">hitt<m type="suffix">ing</m></w>
```

Figure 7 : Example for the usage of `@lemmaRef`

The current documentation for the class may be accessed at http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.linguistic.html.

5. Normalization of forms

In corpora of historical texts with non-standardized spelling, regularization is a very frequent matter which may be applied not only to words but also to punctuation characters. For corpus queries, the normalization of historical spellings may be useful for search purposes because users do not have to think of possible spellings of a search term. For further linguistic analysis, the normalization of writings may be a significant first step towards providing homogeneous input data for applications based on lexica with standardized spelling (e.g., lemmatization and POS analysis usually expects modernized spellings, cf. Jurish 2012[2008]:13). The TEI has at its disposal the powerful but also a somewhat costly descriptive mechanism of `choice/reg|orig` for the annotation of regularization, whereby the element `<choice>` contains the element `<orig>` with the original version of the text, and the element `<reg>` with the normalized version. For the purpose of lightweight linguistic description and for the sake of coherence with the already adopted proposals, we suggest an addition of an alternative device – an attribute to store the normalized equivalent of the text content of `<w>` or `<pc>`. That attribute is already part of the TEI repertoire, but it is defined by another attribute class, namely *att.lexicographic*:[9]

[9] In contrast to research areas dealing with strictly numeric data of various sorts, the langage-resource community uses the terms « normalization », « standardization », « regularization » (and often also « modernization ») to a large extent interchangeably, with nuances getting teased out only at the level of specific project guidelines and only when needed. This is why there is nothing untoward in suggesting the use of `@norm` for the functionality otherwise claimed for the element named `<reg>` (defined in the Guidelines as containing « a reading which has been regularized or normalized in some sense »).

`@norm` : « (normalized) gives a normalized form of information given by the source text in a non-normalized form. » (TEI 2018 : att.lexicographic) ; status: optional; datatype: teidata.text[10]

We propose that this attribute should also be available within the class *att.linguistic*. In other words, we do not postulate an introduction of a new device, but merely an extension of the structural context in which an existing TEI attribute may be used, keeping its definition and data type intact.

The `@norm` attribute would complete the set of attributes for token-based linguistic annotation. As stated above, (automatic) normalization is a crucial basic step for further linguistic analysis on the token level. In this sense, tokenization, orthographic normalization, lemmatization, POS tagging and morphosyntactic analysis form a sequence of analytic steps based on one another and thus connected in terms of an analysis chain. The initial point of this analysis is the textual content of the `<w>` element, i.e. a token of the historical source text. The `<w>` element, together with its *att.linguistic* attributes, would then form a single coherent unit encapsulating token-level linguistic information.

By using `choice/reg|orig`, the initial step of the above-mentioned information chain is moved out of the immediate context of the `<w>` element and into another subset of TEI elements. Regarding consistency, it doesn't seem appropriate that some linguistic analysis results for tokens lead to further embedding of the source text whereas others do not. Furthermore, such encoding adds significantly to the complexity of annotation itself. Linguistic annotation becomes mixed with customary TEI encoding, e.g. with the annotation of highlighting (`<hi>`; see fig. 8 and 9), of erroneous text (`choice/sic|corr`), or even with text interrupting the running text (`<fw>`; see fig. 10). From the perspective of processing and post-processing, this mixture of approaches necessitates efforts (e.g. by increasing the possibility of exceptions, or by simply enforcing the usage of different routines to be able to add and extract similar-level information to and from the text) that seem avoidable by allowing for the homogeneity of linguistic markup.

In contrast, the TEI tagset can easily tolerate an addition of an encoding variant that provides a localized alternative to existing tagging solutions. The TEI has not been maintained as a tagset without ambiguities but has rather been created with the motive of suiting as many communities and project necessities as possible:

> « Because of its roots in the humanities research community, the TEI scheme is driven by its original goal of serving the needs of research, and is therefore committed to providing a maximum of comprehensibility, flexibility, and extensibility. [...] This has led to a number of important design decisions, such as: [...] alternative encodings for the same textual features » (http://www.tei-c.org/release/doc/tei-p5-doc/en/html/AB.html#ABTEI2)

[10] See http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.lexicographic.html.

The ODD mechanism was created to allow projects to reduce the TEI tagset to a subset adjusted for the respective contexts of its usage:

« Because the TEI Guidelines must cover such a broad domain and user community, it is essential that they be customizable: [...] Customization is a central aspect of TEI usage and the Guidelines are designed with customization in mind. » (http://www.tei-c.org/Guidelines/Customization/)

```
<choice>
 <orig>
  <w lemma="wohlstilisierend" pos="ADJA">
Wohl-<hi rendition="#aq">ſtylifi</hi>rende
  </w>
 </orig>
 <reg>
  <w lemma="wohlstilisierend" pos="ADJA">
Wohl<hi rendition="#aq">stylisie</hi>rende
    </w>
  </reg>
</choice>
```

Figure 8: From Marperger (1717: 655), tagging with `<choice>`

```
<w lemma="wohlstilisierend" pos="ADJA"
  reg="Wohlstilisierende">
Wohl-<hi rendition="#aq">ſtylifi</hi>rende
</w>
```

Figure 9: From Marperger (1717: 655), tagging with `@reg`

```
<w>Flecken</w>
<w>oder</w>
<w norm="Dorf">Dorff</w>
<w norm="desselbigen">deſ-<lb/>
  <fw place="bottom" type="catch">
    ſelbi-</fw>
  <pb facs="#f0672" n="656"/>
  <fw place="top" type="header">
    <hi rendition="#b">Von der</hi>
    <hi rendition="#aq #i">Præſtan</hi>
    <hi rendition="#b">tz
    und Vortreflichkeit</hi>
  </fw><lb/>
  ſelbigen</w>
<w>Landes</w>
```

Figure 10: Fragment from: Marperger (1717: 671); tagging of a token interrupted by page break with `@norm` attribute

It is therefore common that the TEI offers several possible encodings for similar phenomena. Hence, the recurring strand of thought in the discussion of any modifications or enrichment of the TEI, that the system should prevent the encoder from 'making mistakes' in choosing the wrong one out of several tagging solutions, does not fit the constitutive design of the ODD-based TEI. The responsibility of ensuring the 'right' markup within a project is in the hands of the encoder and/or the tools and validation scenarios, which may even be independent of the TEI-defined schemas. From our point of view, the multitude of potential research foci, tools for linguistic analysis and visualisation, and language-specific constraints, necessitate a variety of approaches to tagging for projects to choose from.

We therefore stress that adding `@norm` to *att.linguistic* is not meant to be a replacement for the `choice/reg|orig` system, but rather a localized alternative, to be used where feasible, and primarily for the purpose of adding basic token-level linguistic information.

In terms of an actual implementation, we propose to extract `@norm` from the *att.lexicographic* class into a separate class (call it e.g. *att.normalize*), of which both *att.lexicographic* and *att.linguistic* will then become members. This would create an inheritance hierarchy that would make it possible to avoid duplication of the attribute definition.

6. Application of the Format

At Northwestern University, Philip R. Burns and Martin Mueller have worked on a project to apply simple linguistic annotation and normalization to the not quite two billion words in 60,000 English texts before 1700 and American texts before 1800 and originally transcribed by the Text Creation Partnership (TCP). They have used `@lemma` and `@pos` attributes, as well as a `@reg` attribute that is functionally equivalent to the proposed `@norm` and could be easily replaced by it. This is a coarse-grained and large-scale enterprise where ease of processing becomes an important concern. Things become a lot simpler if all relevant properties of a given 'word' or lexical item can be contained within the element (`<w>` or `<pc>`) that encloses it. It also helps if each property has a readily understood name (`@pos`, `@norm`). If these properties are isolated within an element you can ignore them: they are a sideshow that does not complicate the hierarchical XML structure. You can also extract them more readily: for any analysis of the 'bag of words' type each word comes in a bag that contains the available data about it.

Approximately 7,000 of those texts are currently available at https://texts.earlyprint.org/, to be joined later in 2018 by another 18,000 texts currently in the public domain. These texts exist in an environment that supports collaborative curation. The adoption of the `@reg` (or `@norm`) attribute has made it significantly easier and cheaper to maintain a corpus that is subject to iterative correction and completion.

The format for lightweight linguistic annotation has also been successfully tested on a sample of 46 texts of the Deutsches Textarchiv project (DTA 2018). The DTA consists of a large corpus of historical German texts, dating back to the 17th to 19th century. For normalization, lemmatization, and POS tagging the DTA applies the integrated system CAB (Jurish 2012) which provides various XML-based output formats, some of which already include the linguistic features discussed here in a similar way as the proposed format and may hence easily be converted to the format proposed here. Figure 11 and 12 illustrate this by example of the sentence : « Es iſt ja von Natur nicht gut/ daß der Menſch allein ſey [...] » (Abel 1699: 49).

```
<w t="Es" exlex="Es" errid="ec" msafe="1">
<moot word="Es" tag="PPER" lemma="es"/></w>
<w t="ift" exlex="ist" errid="25429"
msafe="1">
  <moot word="ist" tag="VAFIN"
lemma="sein"/></w>
...
<w t="Menſch" exlex="Mensch" errid="ec"
msafe="1">
  <moot word="Mensch" tag="NN"
lemma="Mensch"/>
</w>
...
<w t="ſey" exlex="sei" errid="57805"
msafe="0">
  <moot word="sei" tag="VAFIN"
lemma="sein"/>
</w>
```

Figure 11: XML (TokWrapFast)-Output by CAB[11]

```
<w norm="Es" pos="PPER"
 lemma="es">Es</w>
<w norm="ist" pos="VAFIN"
 lemma="sein">iſt</w>
<w norm="ja" pos="ADV" lemma="ja">
 ja</w>
<w norm="von" pos="APPR"
 lemma="von">von</w>
<w norm="Natur" pos="NN"
 lemma="Natur">Natur</w>
<w norm="nicht" pos="PTKNEG"
 lemma="nicht">nicht</w>
<w norm="gut" pos="ADJD"
 lemma="gut">gut</w>
<pc norm="/" pos="$("
 lemma="/">/</pc>
<w norm="daß" pos="KOUS"
 lemma="daß">daß</w>
<w norm="der" pos="ART"
 lemma="d">der</w>
<w norm="Mensch" pos="NN"
 lemma="Mensch">Menſch</w>
<w norm="allein" pos="ADV"
 lemma="allein">allein</w>
<w norm="sei" pos="VAFIN"
 lemma="sein">ſey</w>
<pc norm="/" pos="$("
 lemma="/">/</pc>
```

Figure 12: Output of CAB (see fig. 11) converted to TEI with *att.linguistic*

7. Limits of application

The mechanism introduced here is intentionally minimalistic: it serves to adorn tokenized text with the basic information labels essential for useful linguistic processing. It is not to be used for cases where multiple grammatical interpretations need to be listed and disambiguated, as in fig. 1 above. It is also far from optimal for handling multi-word units, especially if they involve discontinuity and/or changes in ordering, both of which can be found e.g. in separable German prefixes, illustrated below. The ideal usage scenario is where tokens (pieces of text) match word forms (as defined by ISO MAF[12]) one-to-one. This is not easy to achieve in natural languages, and therefore some repair strategies will usually be necessary.

The examples in fig. 13 and 14 illustrate two possible strategies of handling word forms which do not match tokens 1:1. In the German sentence *Ich stimme dir zu* (« I agree with you »), the base, infinitive form of the verb, is *zustimmen*, with the prefix attached to the verb. In some contexts, however, the prefix gets separated, yielding the correspondence between a single word form and two tokens. The typical way to handle this is by expressing the dependency between the tokens by modifying the repertoire of part-of-speech symbols – in this very case, the label « PTKVZ » of the STTS tagset[13] signals the prefix of a split form, so that the two parts can be reassembled at some higher level of representation (fig. 13). The approach taken in fig. 14 reassembles the morphological parts already at the level of tokens, and uses a convention whereby grammatical information describing the entire word form is represented on its first token.

```
<w pos="PPER" lemma="ich">Ich</w>
<w pos="VVFIN" lemma="stimmen">
stimme</w>
<w pos="PRF" lemma="du">dir</w>
<w pos="PTKVZ" lemma="zu">zu</w>
```

Figure 13 : Dependency between the prefix *zu-* and the verb stem *stimme* encoded indirectly, by means of a POS label (“ PTKVZ ”).

```
<w xml:id="t2" pos="VVFIN"
lemma="zustimmen" next="#t4">
 stimme</w>
<w pos="PRF" lemma="du">dir</w>
<w xml:id="t4" prev="#t2">zu</w>
```

Figure 14: A fragment of fig. 13 with the dependency captured at the level of markup

However, while the representation proposed here is able to handle mild deviations from the 1:1 correspondence between word forms and tokens, it is not sufficient for handling complex multi-word units or for syntactic description – these require more powerful descriptive mechanisms. Similarly, in systems which rely on the presence of `@norm` and use its content for further linguistic analysis, cases where a historical token corresponds to more than one normalized token may also turn out to be beyond the scope of lightweight descriptive mechanisms. Nevertheless, the fact that both archives mentioned in Section 6 have used the proposal described here successfully shows that many other projects can benefit from it.

[11] See http://www.deutschestextarchiv.de/demo/cab/ for the web service.

[12] While ISO specifications created outside the scope of the ISO-TEI liaison need to be purchased, ISO makes all definitions publicly viewable in the new ISO Online Browsing Platform : https://www.iso.org/obp/ui.

[13] http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/TagSets/stts-table.html.

8. Summary and outlook

The goal of *att.linguistic* together with `@norm` is not to facilitate in-depth linguistic annotation, but rather to equip 'off-the-shelf' TEI in the very basic tools that linguists can use, and that non-linguists can safely add to their existing resources in order to enhance them. Where more elaborate analysis is needed, with explicit distinction between tokens and word forms and/or with hierarchical or dependency structures and the like, other TEI-based devices should be used.

Introduction of the `@norm` attribute will be the topic of a forthcoming LingSIG feature request directed at the TEI Technical Council.

9. Acknowledgements

We would like to thank the anonymous LREC reviewers for their comments on an earlier version of this paper. We are grateful to the TEI community at large, and the LingSIG community in particular, for numerous discussions at the stage of the preparation of this proposal and during the process of approval by the TEI Technical Council. Finally, many thanks are due to Peter Stadler, who acted as a liaison between the LingSIG and the TEI Council.

10. References

10.1 Research Articles

Bański, P. (2010). Why TEI standoff annotation doesn't quite work: and why you might want to use it nevertheless. In Proceedings of Balisage: The Markup Conference. Vol. 5 of Balisage Series on Markup Technologies. DOI:10.4242/BalisageVol5.Banski01.

Bański, P, Gaiffe, B., Lopez, P., Meoni, S., Romary, L., et al. (2016). Wake up, standOff!. TEI Conference 2016, Sep 2016, Vienna, Austria. ⟨hal-01374102⟩

Budin, G., Majewski, S., and Mörth, K. (2012). Creating Lexical Resources in TEI P5. *Journal of the Text Encoding Initiative* 3. DOI : 10.4000/jtei.522.

Haaf, S. (2016): Corpus Analysis based on Structural Phenomena in Texts: Exploiting TEI Encoding for Linguistic Research. In Nicoletta Calzolari (Conference Chair) et al.: Proceedings of the 10th LREC, pages 4365–4372, Portorož, Slovenia, May. ELRA. http://www.lrec-conf.org/proceedings/lrec2016/pdf/1154_Paper.pdf

Jurish, B. (2012). Finite-state Canonicalization Techniques for Historical German. PhD thesis, University of Potsdam. urn:nbn:de:kobv:517-opus-55789.

Ide, N. (1998). Encoding Linguistic Corpora. In Proceedings of the 6th Workshop on Very Large Corpora, pages 9–17, Montreal, Canada. http://www.cs.vassar.edu/~ide/papers/ces.wvlc.pdf .

Pose, J., Lopez, P., Romary, L. (2014). A Generic Formalism for Encoding Stand-off annotations in TEI. 2014. ⟨hal-01061548⟩

Schöch, Ch. (2016): Ein digitales Textformat für die Literaturwissenschaften. Die Richtlinien der Text Encoding INitiative und ihr Nutzen für Textedition und Textanalyse", *Romanische Studien* 4.

Stegmann, J., and Witt, A. (2009). TEI Feature Structures as a Representation Format for Multiple Annotation and Generic XML Documents. In Proceedings of Balisage: The Markup Conference. Vol. 3 of Balisage Series on Markup Technologies.
DOI: 10.4242/BalisageVol3.Stegmann01

10.2 (Language) Resource References

CLAWS TAGSET C8. Last changed - N.I.S. 14 Jan 2001. http://ucrel.lancs.ac.uk/claws8tags.pdf.

Deutsches Textarchiv (DTA). Grundlage für ein Referenzkorpus der neuhochdeutschen Sprache. Berlin-Brandenburg Academy of Sciences and Humanities. Berlin 2007-2018. http://www.deutschestextarchiv.de/.

EEBO-TCP (2018): Early English Books Online. Text Creation Partnership. University of Michigan Library. www.textcreationpartnership.org/.

ISO 24610-1:2006. Language resource management, Feature structures, Part 1: Feature structure representation.

ISO 24611:2012. Language resource management, Morpho-syntactic annotation framework (MAF)

LingSIG "word attributes" project (2017): https://github.com/LingSIG/wordAttributes

NKJP: National Corpus of Polish – Narodowy Korpus Języka Polskiego (2008-2012). http://nkjp.pl/.

Text Encoding Initiative Consortium (eds.). TEI P5: Guidelines for Electronic Text Encoding and Interchange. Version 3.3.0. Last updated 31st January 2018. http://www.tei-c.org/Vault/P5/3.3.0/doc/tei-p5-doc/en/html/.

TEI Elements : http://www.tei-c.org/release/doc/tei-p5-doc/en/html/REF-ELEMENTS.html

Universal Dependencies Tagset: POS tags. Version 2. http://universaldependencies.org/u/pos/ (2014)

WebLicht: https://weblicht.sfs.uni-tuebingen.de/ (2018)

10.3 Sources

Abel, H. K. Wohlerfahrner Leib-Medicus der Studenten. Leipzig, 1699. In Deutsches Textarchiv. URN: urn:nbn:de:kobv:b4-200905199878

Gottfried, J. L. Newe Welt Vnd Americanische Historien. Frankfurt (Main) 1631. In Deutsches Textarchiv. URN: urn:nbn:de:kobv:b4-200905199012

Marperger, P. J.: Der allzeit-fertige Handels-Correspondent. 4Th ed. Hamburg 1717. In Deutsches Textarchiv. URN: urn:nbn:de:kobv:b4-20887-0

A Gold Standard for Multilingual Automatic Term Extraction from Comparable Corpora: Term Structure and Translation Equivalents

Ayla Rigouts Terryn, Véronique Hoste, Els Lefever
LT3 Language and Translation Technology Team,
Department of Translation, Interpreting and Communication, Ghent University
Groot-Brittanniëlaan 45, 9000 Gent, Belgium
{ayla.rigoutsterryn, veronique.hoste, els.lefever}@ugent.be

Abstract

Terms are notoriously difficult to identify, both automatically and manually. This complicates the evaluation of the already challenging task of automatic term extraction. With the advent of multilingual automatic term extraction from comparable corpora, accurate evaluation becomes increasingly difficult, since term linking must be evaluated as well as term extraction. A gold standard with manual annotations for a complete comparable corpus has been developed, based on a novel methodology created to accommodate for the intrinsic difficulties of this task. In this contribution, we show how the effort involved in the development of this gold standard resulted, not only in a tool for evaluation, but also in a rich source of information about terms. A detailed analysis of term characteristics illustrates how such knowledge about terms may inspire improvements for automatic term extraction.

Keywords: terminology extraction, gold standard, comparable corpora

1. Introduction

Automatic term extraction (ATE) has been a productive and successful field of research within natural language processing, yet the evaluation of ATE remains particularly difficult. The main difficulty of the task lies in the ambiguous nature of terms; there are no objective rules to distinguish terms from non-terms. While there are many definitions or terms found in the literature, such as "words that are assigned to concepts used in the special languages that occur in subject-fields or domain-related texts" (Wright, 1997, p. 13), these definitions, however accurate, aren't always helpful when deciding whether or not a lexical unit can be considered a term. Not only does this make ATE a challenge, it also poses a problem for the evaluation. The typical way to evaluate such a task is to compare the automatic output against a manually constructed gold standard (GS), in other words, to compare automatic versus human performance. In this way, precision (how many of the automatically extracted terms are correct) and recall (how many of the terms in the text are automatically extracted) can be calculated. To construct this GS, text must be manually annotated. The corpus must be large and domain-specific enough to be used as input for ATE and it must be annotated entirely to calculate recall. Owing to the ambiguous nature of terms and the necessary volume of text, this is an arduous, time consuming task. Moreover, it results in low inter-annotator agreement scores (Rigouts Terryn et al., Submitted), which are supposed to be an indication of the objectivity and quality of the annotations.

Recently, research on ATE has shifted from monolingual ATE, to bilingual ATE, first from parallel corpora and currently also from comparable corpora. ATE from comparable corpora (ATECC) attempts not only to recognise terms in a text, but also to find equivalent terms in the different languages of a comparable corpus. Comparable corpora are collections of texts in different languages, on the same subject (and preferably in the same style), but the texts are not each other's translations. Using comparable corpora is much more difficult than using parallel corpora, since it is impossible to know beforehand where to look for term translation equivalents or even whether appropriate equivalents are available in the corpus. However, comparable corpora have the great advantage of availability: it is much easier and less costly to collect comparable corpora (manually or automatically) than parallel corpora, which require aligned human translations. ATECC can therefore be used for languages with fewer resources or rare and specialised domains for which data is too scarce to compile a parallel corpus.

The result of ATECC is usually an ordered list of potentially equivalent candidate terms in the target language, for each candidate term in the source language. Consequently, the evaluation needs to include an evaluation of both term extraction, and term linking. Current research in ATECC is mostly evaluated by using reference translations from a source other than the input corpus or by using only a limited set of manually evaluated term equivalents from the input corpus. However, to accurately evaluate the entire output and be able to trace mistakes back to their source, a new type of GS is needed. To the best of our knowledge, we are the first to undertake the challenge of constructing a completely manually annotated GS for ATECC, thus requiring the development of a novel methodology to annotate and structure the data. To address the problem of low inter-annotator agreement due to the subjective nature of terms, a new term annotation scheme was developed and tested in combination with detailed annotation guidelines. The investment of time and effort is not to be underestimated, but the result is both an informative instrument for evaluation and an invaluable source of information about the nature of terms.

The remainder of this contribution is dedicated, first, to a summary of the state-of-the-art, subsequently, to a description of the GS and, next, to a discussion of what can be learnt about terms and term equivalents from this GS. The results will be recapitulated in the conclusions.

2. State of the Art

Researchers have been creative in finding ways to evaluate ATE, especially since the traditional way, i.e. calculating precision and recall, requires a fully annotated corpus. For instance, in the EVALDA-CESART project (Mustafa El Hadi et al., 2004), existing reference word lists were completed by domain specialists to calculate precision, but

recall remained a problem, since it requires a complete GS. Another problem emerged because of the need for term annotation instructions, resulting in a plethora of diverging annotation schemes and guidelines, rendering re-use and comparison problematic. For instance, Bernier-Colborne (2012) developed detailed, but very domain-specific annotation guidelines. Another example is the limitation to certain parts-of-speech (POS) patterns: sometimes, only nouns and noun phrases are annotated (Bernier-Colborne & Drouin, 2014); others allow more POS patterns (Schumann & Fischer, 2016).

Similar problems exist for the evaluation of ATECC. Constructing a complete record of all translation equivalents for all terms in a comparable corpus is a daunting task, so alternative solutions have been invented. For instance, Laroche and Langlais (2010) use translations from an existing thesaurus; Kontonatsios (2015) also uses a limited set of reference translations and sets the maximum system performance to an estimate of the percentage of translations that are present in the corpus. In the TTC project (Loginova et al., 2012), a GS was created based on the input corpus, but it was limited to ±100 term pairs.

As for the investigation of the structure of terms, surprisingly little empirical research can be found on term length or structure based on manual annotations. Justeson and Kats (1995) started from dictionaries of technical terminology and selected 200 technical terms from four different dictionaries. Only 35 of the resulting 800 terms weren't noun phrases, which led them to focus on noun phrases alone. Out of 800 noun phrase terms, 30% were single-word terms (SWTs), 55% were 2WTs, 12% 3WTs and the remaining terms were multi-word terms (MWTs) of four or more words. Only in the medical domain did they find more SWTs than 2WTs, which they attribute to the presence of more Latin or Greek single-word compounds. Around the same time, Nkwenti-Azeh (1994) had reached similar results.

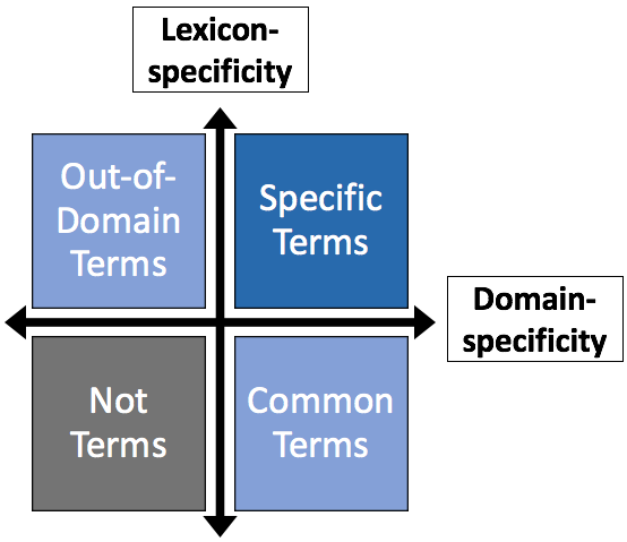


Figure 1: Annotation scheme

3. Gold Standard

3.1 Monolingual Gold Standard for ATE

Three languages were included in this project: English (EN), French (FR) and Dutch (NL). They provide a good contrast between well- and less-resourced languages (EN/FR vs. NL), Romance and Germanic languages (FR vs. EN/NL) and, particularly, languages with very different compounding strategies. Compound terms in English are often concatenations of nouns, separated by a whitespace, whereas, in French, the different parts of the compound are typically connected by prepositions and Dutch compound terms are characteristically one, long compound word. Besides language, the structure of terms may also be influenced by domain, so three different domains were selected: medical (heart failure), technical (wind energy) and juridical (corruption). Each corpus was manually checked and enhanced, but they were all based on pre-existing resources. The medical corpus of medical abstracts and short papers was based on previous research about terminology (Hoste et al., Accepted), as was the technical corpus (Daille, 2012). The juridical corpus was assembled based on a collection of titles provided by the DGT of the European Commission. The corpora are the same size for each language. Per language, the medical corpus has ± 46k words, the technical one ± 310k and the juridical corpus contains ± 670k words per language.

The medical corpus has been completely annotated and large parts of the other corpora have been annotated as well (see Table 1). The annotation of the juridical and technical corpora is an ongoing work, but a sufficient portion has already been annotated to provide a useful resource for the evaluation of ATE. There were two main concerns for the development of the annotation scheme. First, it should be intuitive and uncomplicated for the annotators and improve inter-annotator agreement. Second, the need for a "highly parametrizable" (Vivaldi & Rodríguez, 2007) GS had been expressed before, to encourage a more detailed analysis of terms and ATE. These considerations led to the development of three term labels: *Specific Terms*, *Out-of-Domain Terms* and *Common Terms*. These are defined by splitting termhood into two parameters: lexicon-specificity and domain-specificity. By representing these on two sliding scales, it results in the matrix shown in Figure 1.

	EN	FR	NL
Heart Failure (HF)	45.788	46.751	47.888
Corruption (CR)	50.322	49.180	50.676
Wind Energy (WE)	76.488	83.259	84.207

Table 1: Number of annotated tokens per corpus

Lexicon-specificity is defined as the degree to which a term belongs to either the general language or to the lexicon of specialists. Domain-specificity shows how relevant the term is to the subject. According to the strictest definitions of terms, they should score high on both scales. Terms in this category were labelled *Specific Terms*. In the domain of heart failure, "ejection fraction" would be an example of a *Specific Term*. However, no matter how well constructed the corpus, there may also be terms which are lexicon-specific, but not domain-specific. These are called *Out-of-Domain Terms*, or, abbreviated, *OOD Terms*. For instance, the medical corpus contains some terms about statistics, such as "p value", which is, in this case, an *OOD Term*. Finally, the opposite may be true as well: *Common Terms* are relevant to the domain, but are also part of the general vocabulary. In the heart failure corpus, a good example would be "heart", which is clearly domain-specific, but, since non-specialists are familiar with the term as well, not

lexicon-specific. While this doesn't eliminate the factor of subjectivity, previous experiments have shown that the labels help the annotators and increase inter-annotator agreement (Rigouts Terryn et al., Submitted). This is especially true for *Specific Terms*, which are often the most relevant terms to ATE users.

The annotation scheme is helpful for deciding whether a linguistic unit is a term. However, there is a second difficulty in term annotation, namely deciding the term boundaries. Elaborate annotation guidelines were constructed to address such added annotation difficulties (https://biblio.ugent.be/download/8503113/8517085.pdf). These guidelines also provide more information on how to determine the degree of lexicon- or domain-specificity as objectively as possible.

ID	**112**
Annotation	beta-blockers
Label	Specific Term
Frequency	4
Texts	144; 096
EN	EN
FR	2801; 3664; 4738; 5268; […]
NL	7558; 5774; 6015; 5998; […]
Lemma	Beta-blocker
Synonym	1971; 1450
Abbreviation	2567
Alt. Spelling	2099; 1509; 2243
Hypernym	87; 393; 1430; 1893; 1303; 111 […]
Hyponym	235; 1577; 2441; 2324; 2669; 222
Other	1027; 1035; 2462; 1563; 776; […]

Table 2: Example term record in the GS for ATECC

3.2 Multilingual Gold Standard for ATECC

For the evaluation of ATECC, the GS should provide more information than only termhood. Ideally, it would also include a record of all possible translation equivalents in the corpus and even additional semantic relations, to encourage a more nuanced evaluation. With such information, it would be possible to tell whether the suggested target language candidate term is a correct translation equivalent and, if not, if it is still in some way related to a correct equivalent. Most importantly, a wrong term suggestion could be traced back to its origins: either the system simply was not able to find the correct equivalent in the target language corpus, or the correct translation was not present in the corpus. It is important to remember that, since comparable corpora aren't in any way aligned, there is no guarantee that the translation equivalents for all terms are present in the corpus. Being able to trace the origins of mistakes in the term linking module of ATECC can be a useful tool to identify areas of improvement. To accommodate all this information in a single document, each unique annotation got an ID number, which could be used as a reference. The annotation, its label, frequency and the texts in which it was found were automatically extracted from the monolingual gold standards. Three fields for each language were added to indicate the source language and refer to the IDs of any equivalents in the other languages. To identify term variants, lemma, synonyms, abbreviations and alternative spellings were added manually. Other semantic links could be indicated as hypernyms, hyponyms or 'other'. Table 2 is an example of the term record for "beta-blockers". All numbers (except frequency and texts) refer to the IDs of other term records. In total, 6818 unique terms were thus created, with an additional 567 records for named entities.

4. Term Analysis

Besides being a useful tool for the evaluation of ATECC, this GS contains a wealth of information about comparable corpora, term frequency, term variation, differences per domain and language etc. In this contribution, we will focus on what can be learnt about the structure of terms, more specifically: term length and term POS patterns. In the following analyses, only term annotations are considered (no named entities) and the numbers we report are calculated on unique terms (one count per term record) instead of absolute frequencies (one count per term occurrence). Nevertheless, both calculations were made, to rule out any discrepancies. Barring some minor variations, they both lead to the same conclusions. It is also important to note, that, according to the annotation guidelines, all content words can be terms and no minimum or maximum term length was stipulated.

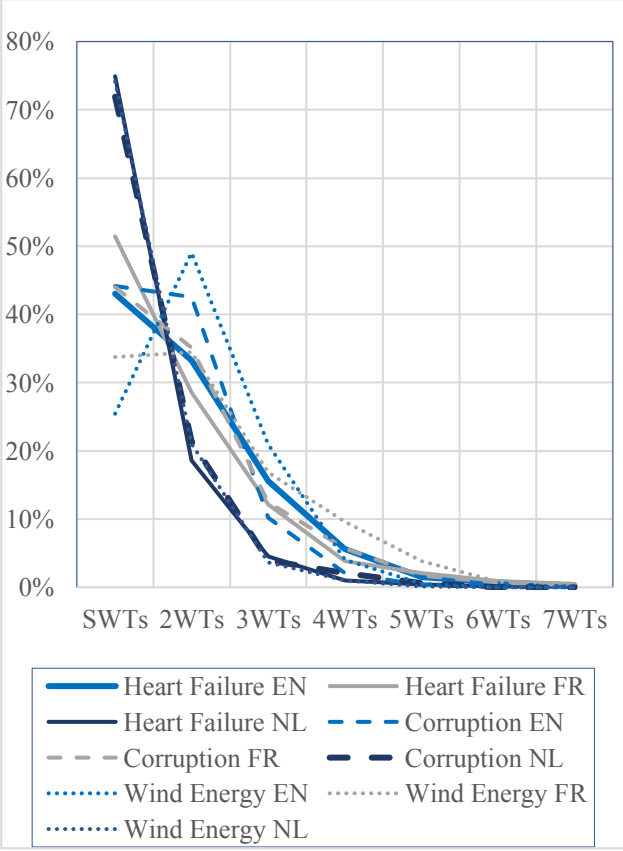


Figure 2: Percentage of terms per term length, comparing languages and domains

4.1 Word Length

When distinguishing solely between SWTs and MWTs, it became quickly apparent that both language and domain have an impact on term length. In this analysis, complex single-word compound terms were processed as SWTs. The percentage of SWTs ranged from 25% (English corpus on wind energy), to 75% (Dutch corpus on heart failure).

Figure 2 shows a chart of the term length for each language and domain. A first observation is that very few terms contain more than five words. In Dutch, over 90% of terms are no longer than 2 words. In English and French, there are more three-word terms, but no more than 10% of the terms are longer than that. Dutch term length (dark blue) appears very consistent, with hardly any variation per domain. French term length (grey) has more variation, especially for wind energy, where there is an equal percentage of SWTs and two-word terms (2WTs). In English, however, the variation per domain is more apparent. For the English corpus on wind energy, there are even more 2WTs (49%) than SWTs (25%). However, apart from the variations for SWTs and 2WTs, the numbers are rather consistent for all languages and domains. As the term length reaches 7 words, the number of terms of that length becomes negligible.

Another interesting observation regarding term length is the difference between the different term categories. There were too few *OOD Terms* to be relevant, but there was a notable difference in term length between *Specific* and *Common Terms*. As shown in Figure 3, *Common Terms* are more often SWTs, and *Specific Terms* are more likely to be longer. Only 1% of *Common Terms* are longer than 3 words, compared to almost 10% of *Specific Terms*. A potential explanation is, that *Specific Terms* are made up of several *Common Terms*, which, when combined, become more lexicon-specific.



Figure 3: Percentage of terms per term length, comparing term categories (averages over all languages and domains)

4.2 POS patterns - monolingual

For a more in-depth analysis of term structure, the LeTs preprocess toolkit (van de Kauter et al., 2013) was used for the automatic linguistic processing of all corpora. This included tokenisation and POS-tagging (i.e. automatically assigning a POS to each token). Only the results for term POS-patterns with large frequencies will be discussed here to avoid overgeneralisation. This is necessary, since automatic POS-tagging isn't flawless and some (parts of) terms were not assigned a POS, since they weren't tokenised separately. For instance, some terms were annotated that were connected to other words by "-", e.g. within the term "angiotensin-converting enzyme", "angiotensin" was annotated, as well as the full term. Since LeTs doesn't tokenise words connected by a hyphen separately, they weren't assigned any POS. Whenever the same term had received different POS-tags in the processed texts, the most frequent tag was used. When there was any ambiguity about the tag, the decision was made manually. For instance, the tagger had difficulties distinguishing between the nouns and named entities, especially for abbreviations, which sometimes lead to different tags for the same abbreviation in different sentences (e.g. "cTnT" occurred in the corpus six times, was tagged as a noun four times and as a named entity twice).

Figure 4 shows how, on average, over 80% of all terms (in all languages and domains) are one of eight POS patterns: single nouns (N), a noun and an adjective (N+A), a single adjective (A), a named entity (NE), two nouns (N+N), two nouns separated by a preposition (N+P+N), two adjectives and a noun (N+A+A) or a single verb (V). The order of nouns and adjectives varies depending on the language. Verbs are not often extracted by ATE, since the frequency of terminological verbs is considered so low, that attempting to extract them introduces more noise in the output than improved recall. Justeson and Katz (1995), for instance, found that only 3 out of 800 technical terms chosen from a dictionary were verbs. Therefore, it was surprising to find single verbs as a rather common POS pattern in our corpus. However, verbs rarely appear within MWTs. This may be explained by the fact that verbs aren't often combined with other words in the exact same way and that the lack of these set combinations leads annotators to only annotate the verb separately, but not as part of a larger MWT.

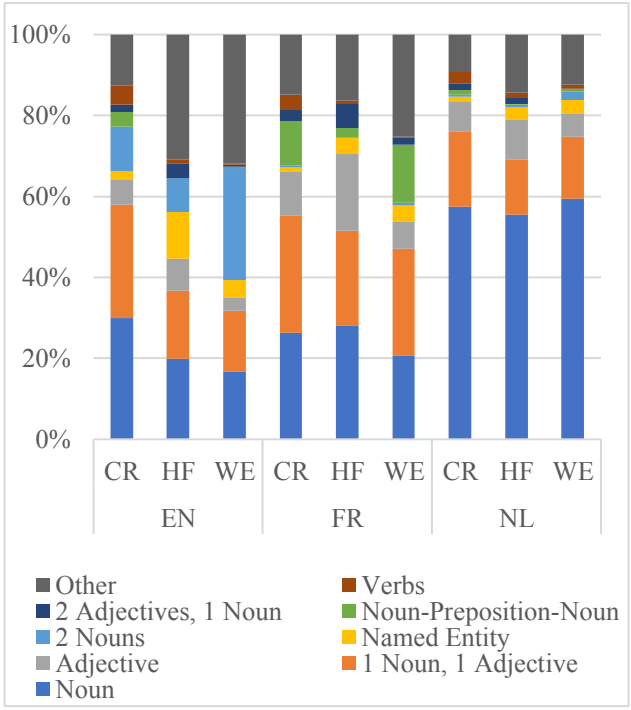


Figure 4: POS patterns

While there are some shared characteristics across all corpora (there are always many N and N+A patterns), substantial differences can be observed between different domains and, especially, between different languages. Some of these differences are easily explained. For instance, the different compounding strategies of the three languages are clearly visible. In English and French, the N+N pattern, which is almost non-existent in the other

languages, is often used for complex terms. In French, a similar phenomenon can be seen with the N+P+N pattern, which is rare in English and Dutch. In Dutch, complex terms are mostly formed by one, long, single-word compound, so there are much more N terms. Another feature may be explained by these compounding rules, namely, that Dutch seems to be less creative with different term POS patterns. A higher percentage of all Dutch terms falls into the above stated eight categories and, when comparing the number of unique POS patterns to the number of unique terms, Dutch has, on average, less different POS patterns (5%) than English and French (both 7%). The most likely explanation is that a single-word Dutch compound can be a combination of several nouns and adjectives, even though the assigned POS tag is N.

During the analysis of term length in the previous section, it was found that *Specific Terms* are less likely to be SWTs than *Common Terms*. The POS pattern analysis confirms this and provides a more detailed picture of the differences. The most common POS pattern for *Common Terms* is invariably N, whereas *Specific Terms* are more likely to be N+A (except in Dutch). Some other observations are more difficult to explain, such as the popularity of N+N terms in the English corpus on wind energy, or the fact that there are less A terms in English than in the other two languages. While the corpora are small enough that such variations may be due to chance, such peculiarities are worth keeping in mind when defining the parameters for ATE(CC).

4.3 POS patterns – translation equivalents

Analysing the term POS patterns per language already inspired some hypotheses about potential translation patterns (e.g. N+N in English ~ N+P+N in French ~ N in Dutch) and, thanks to the multilingual GS on heart failure, it is possible to substantiate these. Again, to avoid overgeneralisation, we only look at the most frequent patterns. For every POS pattern, the POS patterns of all available equivalents in the other languages were analysed. This revealed, for instance, that, out of the 365 Dutch translations for English N terms, 303 of them were also N terms. So, to find Dutch term equivalents for English N terms, the search should be focussed on Dutch N terms. However, in the opposite direction (English translations for Dutch N terms), the search should probably be widened to include other POS patterns, since only 260 out of 564 of the English term equivalents for Dutch N terms are also N terms. Other common POS patterns for the English equivalents are A+N (75), N+N (59) and NE (51). A similar pattern can be discerned between Dutch and French and for longer POS patterns, where English and French terms can often be mapped to shorter Dutch terms. In combination with the frequency of the N pattern, as discussed in the previous section, it is safe to say that single noun compound terms are a common occurrence in Dutch. A potential conclusion for the improvement of Dutch ATECC could be to incorporate automatic decompounding.

While this isn't very surprising, it is a good example of how the GS can help to define the parameters for ATECC. Another, more striking example, is the N+A pattern in English and French. Since the pattern is very common in both languages and single nouns are very rarely compounds in either language, it could be expected that the term equivalents of these patterns would correspond nicely. However, this is only true for 60% of the French equivalents found for English N+A terms and for only 43% of the English equivalents for French N+A terms. Some peculiarities may be due to differences in the automatic POS tagging for the different languages. For instance, there is no special tag for abbreviations in English, but there is one for Dutch and French. Equivalents for English terms with the NE tag do not often have the same tag in French (18%) or Dutch (12%) and there are more of NEs in English as well. A detailed look at the terms tagged as NEs in English revealed that at least part of this incongruity is due to the lack of a special tag for abbreviations in English, since it appears that these abbreviations are often tagged as NEs and, as was already discovered in a previous analysis, there are much more abbreviations in the English texts.

This is only a selection of some of the observations resulting from the GS, since the conclusions differ for each pattern and each language. However, the examples presented in this contribution do illustrate the usefulness of all the annotation work performed for the GSs and the potential of a bottom-up approach for ATE(CC).

5. Conclusion

Terms are a very ambiguous concept, which makes the development of algorithms for ATE a challenge, but also provides difficulties for the evaluation of the task. ATECC, which involves not only a term recognition module, but also a multilingual term linking module is even more difficult to evaluate. Therefore, several corpora in three different languages and domains were manually annotated. Based on the monolingual annotations about heart failure, a multilingual GS was created for ATECC. These resources will be made publically available in due course, just like the annotation guidelines, which already are. As illustrated in this contribution, such a GS can be used for more than evaluation purposes alone, as it is also a rich source of information about terms, which may inspire ideas for the improvement of ATECC.

We showed that, in general, terms are mostly SWTs or 2WTs and that very few terms are longer than 5 words. However, *Specific Terms* tend to be longer and less likely SWTs than *Common Terms*. Next, the POS patterns of the annotated terms were analysed. Apart from the popularity of N and N+A terms, there are notable differences across the three languages and even some differences per domain. Most of the differences per language were related to the language structure, such as compounding rules, but other differences may be the result of differences in the automatic POS tagging. Finally, it was shown how the data also provide information about the POS patterns of term equivalents in the different languages, which could be useful for the term linking module of ATECC.

Some of the ideas for the improvement of ATE(CC) based on these data are a maximum term length to improve ATE precision, Dutch decompounding for term linking and the inclusion of terms that aren't noun phrases, such as adjectives and even verbs and adverbs.

6. Bibliographical References

Bernier-Colborne, G. (2012). Defining a Gold Standard for the Evaluation of Term Extractors. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC)*. Istanbul, Turkey: ELRA.

Bernier-Colborne, G., & Drouin, P. (2014). Creating a test corpus for term extractors through term annotation. *Terminology*, *20*(1), 50–73.

Daille, B. (2012). Building Bilingual Terminologies from Comparable Corpora: The TTC TermSuite. In *Proceedings of the 5th Workshop on Building and Using Comparable Corpora with special topic "Language Resources for Machine Translation in Less-Resourced Languages and Domains", co-located with LREC 2012*. Istambul, Turkey.

Hoste, V., Vanopstal, K., Rigouts Terryn, A., & Lefever, E. (Accepted). The trade-off between quantity and quality. Comparing a large web corpus and a small focused corpus for medical terminology extraction. *Across Languages and Cultures*.

Justeson, J., & Katz, S. (1995). Technical Terminology: Some Linguistic Properties and an Algorithm for Identification in Text. *Natural Language Engineering*, *1*(1), 9–27.

Kontonatsios, G. (2015). *Automatic Compilation of Bilingual Terminologies from Comparable Corpora* (Doctor of Philosophy). University of Manchester, Manchester.

Laroche, A., & Langlais, P. (2010). Revisiting Context-based Projection Methods for Term-Translation Spotting in Comparable Corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)* (pp. 617–625). Beijing, China.

Loginova, E., Gojun, A., Blancafort, H., Guégan, M., Gornostay, T., & Heid, U. (2012). Reference Lists for the Evaluation of Term Extraction Tools. In *Proceedings of the 10th International Congress on Terminology and Knowledge Engineering*. Madrid, Spain: ACL.

Mustafa El Hadi, W., Timimi, I., & Dabbadie, M. (2004). EVALDA-CESART Project: Terminological Resources Acquisition Tools Evaluation Campaign. In *proceedings of LREC 2004* (pp. 515–518). Lisbon, Portugal.

Nkwenti-Azeh, B. (1994). Positional and Combinational Characteristics of Terms: Consequences for Corpus-based Terminography. *Terminology*, *1*(1), 61–95.

Rigouts Terryn, A., Hoste, V., & Lefever, E. (Submitted). In No Uncertain Terms: A Gold Standard for Automatic Terminology Extraction from Comparable Corpora. *Terminology*.

Schumann, A.-K., & Fischer, S. (2016). Compasses, Magnets, Water Microscopes. In *Proceedings of LREC 2016* (pp. 3578–3584). Portorož, Slovenia: ELRA.

van de Kauter, M., Coorman, G., Lefever, E., Desmet, B., Macken, L., & Hoste, V. (2013). LeTs Preprocess: The multilingual LT3 linguistic preprocessing toolkit. *Computational Linguistics in the Netherlands Journal*, *3*, 103–120.

Vivaldi, J., & Rodríguez, H. (2007). Evaluation of terms and term extraction systems: A practical approach. *Terminology*, *13*(2), 225–248.

Wright, S. E. (1997). Term Selection: The Initial Phase of Terminology management. In *Handbook of Terminology management* (Vol. 1, pp. 13–23). Amsterdam: John Benjamins.

Handling Big Data and Sensitive Data Using EUDAT's Generic Execution Framework and the WebLicht Workflow Engine.

Claus Zinn, Wei Qui, Marie Hinrichs, Emanuel Dima, and Alexandr Chernov
University of Tübingen
Wilhelmstrasse 19, 72074 Tübingen, Germany
{claus.zinn, wei.qui, marie.hinrichs, emanuel.dima, alexandr.chernov}@sfs.uni-tuebingen.de

Abstract

Web-based tools and workflow engines can often not be applied to data with restrictive property rights and to big data. In both cases, it is better to move the tools to the data rather than having the data travel to the tools. In this paper, we report on the progress to bring together the CLARIN-based WebLicht workflow engine with the EUDAT-based Generic Execution Framework to address this issue.

Keywords: Generic Execution Framework, WebLicht, Big Data, Sensitive Data

1. Introduction

Our work addresses two challenges that affect the applicability of workflows for some data sets. First, restrictive property rights may forbid research data to leave their home institution, and therefore, data transferal to other institutions for processing is not allowed. The second issue concerns the size of the data, with big data often causing prohibitive overhead once it is necessary to send such data back and forth to the various tools of a scientific tool pipeline. In both cases, it is desirable to bring the workflow engine to the data, rather than having the data travel to the tools.

The EUDAT project is currently developing the Generic Execution Framework (GEF). The GEF aims at providing a framework that allows the execution of scientific workflows in a computing environment close to the data. In this paper, we describe how WebLicht needs to be adapted to render its services compatible with the GEF.

The remainder of the paper is structured as follows. In Sect. 2, we give some technical background on the Generic Execution Framework and WebLicht. Sect. 3 specifies the GEF-WebLicht integration from WebLicht's side. This includes user-centric front-end considerations as well as necessary adaptations to WebLicht's back-end and the processing services connected to WebLicht, which need to be converted to GEF compatible web services. An integral part is a translation mechanism that liaises the WebLicht workflow engine with the GEF environment, and which ensures that all data transfers occur at the GEF's host institution, which also hosts all data. Sect. 4 reviews the service compatibility of WebLicht from the point of view of a GEF administrator, and in Sect. 5, we reconsider WebLicht's perspective. In Sect. 6, we conclude.

2. Background

2.1. The Generic Execution Framework

One underlying motivation for the GEF is that datasets have become much larger than the tools that process them. It is thus more efficient to move the tools to the data rather than the data to the tools, given that the tools do not require a large amount of processing power. This argument is often supported by restrictive property or privacy rights attached to the data; here data transfer to tools not under control by the property holder is often not possible. Also, the transfer of big data sets across the tools of a tool chain often leads to bottlenecks, and transfer time is sometimes higher than the actual processing time.

The GEF aims at a framework that allows developers to pack tools (and their computation) into movable containers that can be moved towards the data. GEF makes use of Docker [U1], a virtualization mechanism that wraps an application into a container that capsules and sandboxes all computation, that is, everything needed to run the application, including all system tools, system libraries, and settings. Docker-based technology shows a very good performance, with container images that can have a small image size, and are fast to start. Container images are immutable by design and relatively easy to specify, using Docker scripts and pre-built container images to build-upon.

Fig. 1 shows the design rationale of GEF (Dima et al., 2015). Docker software runs on the operating system of the host, on the host's hardware, and is able to execute Docker containers (holding the tools' virtualization). The

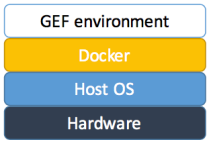


Figure 1: The GEF architecture.

GEF software is built-upon Docker allowing community admin users to upload their tools together with a Dockerfile from which Docker container images are generated. These images become GEF services and can be invoked on any data set in the EUDAT Collaborative Data Infrastructure (EUDAT, 2017), see [U2].

The GEF source code is available at GitHub [U3], and there is available a 1.0.0-beta release. The software's back-end is built upon the Go programming language [U4], and its front-end is built upon the React Javascript library for building user interfaces [U5]. Interested parties can use the software, as a GEF environment has been set-up for public use, see [U8].

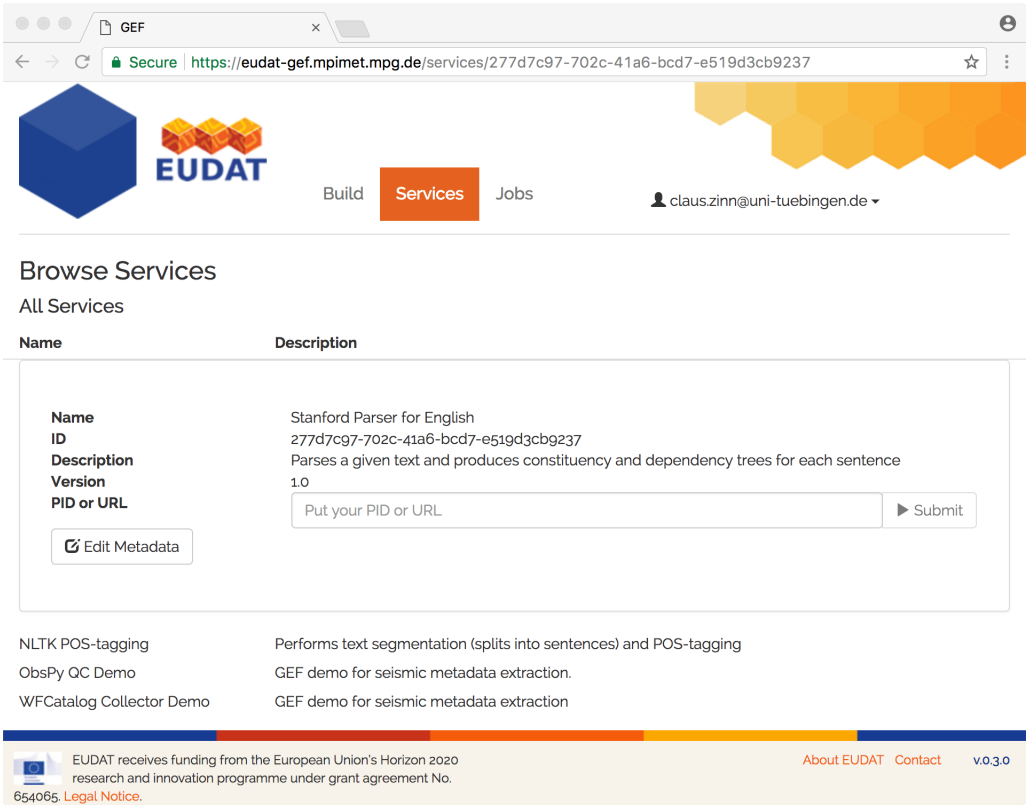


Figure 2: Main Page of the GEF front-end, see [U8].

Fig. 2 shows the front-end of the GEF. In the top pane, there are three items: "Build", "Services" and "Jobs". When choosing the first item, the GEF admin ("power user") can build a new GEF service. For this, the user needs to upload a Dockerfile, together with other files which should be part of the container. With the file upload complete, GEF's underlying Docker software will build a Docker-based container image, which the GEF will use to provide the corresponding GEF service of the application.

Fig. 2 shows four services from the GEF test site. A GEF service can be run by supplying a PID or URL that points to the input to be processed. Hitting the "Submit" button, will start the service asynchronously. All running jobs can be inspected through "Jobs". Once a job terminates, its output volume holds the result of the computation.

2.2. The WebLicht Workflow Engine

WebLicht is a workflow engine giving users a web-based access to over fifty tools for analysing digital texts (Hinrichs et al., 2010). Its pipelining engine offers predefined workflows ("Easy Mode") and supports users in configuring their own ("Advanced Mode"). With WebLicht, users can analyse texts at different levels such as morphology analysis, tokenization, part-of-speech tagging, lemmatization, dependency parsing, constituent parsing, co-occurrence analysis and word frequency analysis, supporting mainly German, English, and Dutch. Note that WebLicht does not implement any of the tools itself but mediates their use via pre-defined as well as user-configurable process pipelines. These workflows schedule the succession of tools so that one tool is called after another to achieve a given task, say, named entity recognition.

WebLicht is a good step forward in increasing (web-based) tool access and usability as its TCF format mediates between the various input and output formats the tools require; and it calls the tools (hosted on many different servers located nation and world-wide) without any user engagement. WebLicht has now been used for many years in the linguistics community, and it is the workflow engine of choice for many national and European researchers in the CLARIN context. WebLicht is actively maintained, profits from regular tool updates and new tool integrations, and has recently been integrated with TüNDRA, a treebank search and visualization tool that allows WebLicht users to inspect linguistically annotated data (Chernov et al., 2017). With TüNDRA, users can search for specific linguistic phenomena at the word and sentence level, and visualize such phenomena. The WebLicht workflow engine is closed source, but all members of the CLARIN Service Provider Federation can access its services [U6]. WebLicht-related technology is maintained in a public Github repository [U7].

Workflow definition and execution. Each tool in the workflow is identified by a persistent identifier that WebLicht's execution system uses to invoke the tool. With each tool invocation, WebLicht passes on a TCF-compliant data file that contains the tool's input. The tool processes selected parts of the input, and enriches the TCF file with the output of the computation, which is then sent back to the WebLicht orchestrator. WebLicht then passes the enriched TCF file to the next tool of the workflow until all tools in

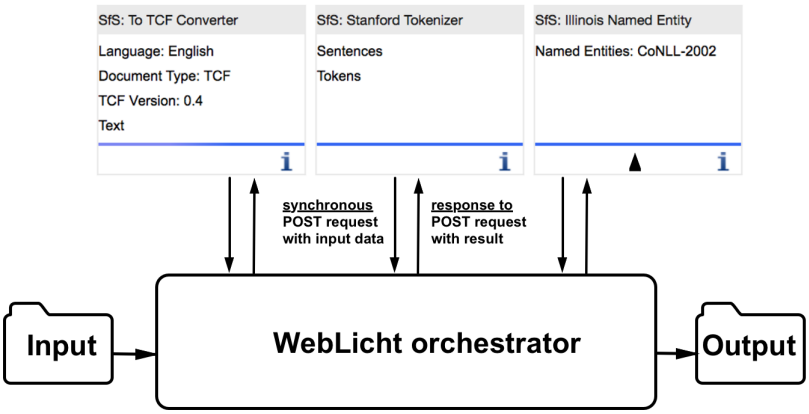


Figure 3: Data Flow in WebLicht.

the workflow have been executed in the given order.

Fig. 3 shows the data flow between the WebLicht orchestrator and the tools. Note that each tool is invoked via the Hypertext Transfer Protocol (HTTP) using a POST method, with the tools' output data being captured in the response body of the request. This design decision makes it hard to use WebLicht on huge data sets, or on data that cannot leave the host institution for property rights or privacy concerns.

In the GEF mind set, WebLicht's tools need to be "migrated" to a GEF environment that encapsulates all computation. To keep all data transfer local to the GEF environment that also hosts the data, the way the WebLicht orchestrator invokes the services needs to be changed.

3. Integration of WebLicht into GEF

GEF is not capable of executing workflows by itself. To execute a GEF-based workflow, it must be orchestrated by an external engine, which makes reference to a GEF-based repository of (immutable) application containers, each of which can be referenced by a PID. In the sequel, we will describe how the WebLicht orchestrator can be used to execute a GEF-based WebLicht workflow; we also describe how any data transfer between the GEF-external WebLicht orchestrator and the GEF-internal services is handled.

3.1. WebLicht Adaptations

WebLicht Front-End. Fig. 4 depicts a proposed UI change to WebLicht's entry page. In addition to the "Input Selection" pane, there is a "Tool Location" pane, where users can select the tool environment that the WebLicht orchestrator should use. The environment "GLOBAL – all tools" is the default tool environment where WebLicht assumes traditional operation (all tools are eligible for inclusion in the workflow). When the user selects the environment "GEF@sfs_tuebingen", the WebLicht orchestrator will only make use of workflows whose tools are running in the GEF environment at the SfS in Tübingen. In GEF mode, no data is uploaded to WebLicht, rather users specify a data URL that points to the data, usually located close to the chosen GEF computing environment.

WebLicht Back-End. The backend adaptations to WebLicht are more substantial. Two changes to WebLicht's back-end are discussed: how should the orchestrator organise its tool space; and how should the orchestrator call the individual tools to minimize data transfers?

The WebLicht orchestrator uses the tools' metadata to help users define workflows. For this, WebLicht is harvesting tool metadata from tool providers (as ingested to the metadata repositories of the CLARIN centre registries). We assume that providers of GEF-ified tools advertise them in the same way so that WebLicht is aware of them. The description of a WebLicht tool is based on the CMDI Profile `WebLichtServiceProfile`, which is addressed via a persistent identifier in the tool's metadata. Among other information, the metadata provides information about the tool's invocation URL and its input and output parameters. By default, the WebLicht Orchestrator works on the entire tool space that results from the tools' metadata harvest. When a WebLicht user specifies a GEF environment of her choice, the orchestrator must cut through this space. In Easy Mode, it needs to identify which predefined workflows are available in the GEF organisation specified by the user (a workflow is available in a GEF environment if each tool of the workflow is available in the GEF environment). In Advanced Mode, the Weblicht orchestrator ignores all tools not originating in the given GEF environment; only tools that are part of a given GEF environment are considered for workflow construction.

Fig. 3 shows the back-and-forth movement of data between the WebLicht orchestrator and the tools it is executing. If a given workflow consists of n tools, then the data is transferred 2 times n between the orchestrator and the tools, potentially across networks of varying bandwidth and latency. With data required to stay in the GEF environment, the interplay between the orchestrator and the tools must change; rather than passing along actual data, now references to data originating in the GEF environment are being passed.

This requires the "wrapping" of tools. A wrapped tool: (i) retrieves the input data from a given URL, (ii) starts the tool it wraps by posting the tool the input data via HTTP, (iii) accepts the response of the HTTP request, (iv) stores

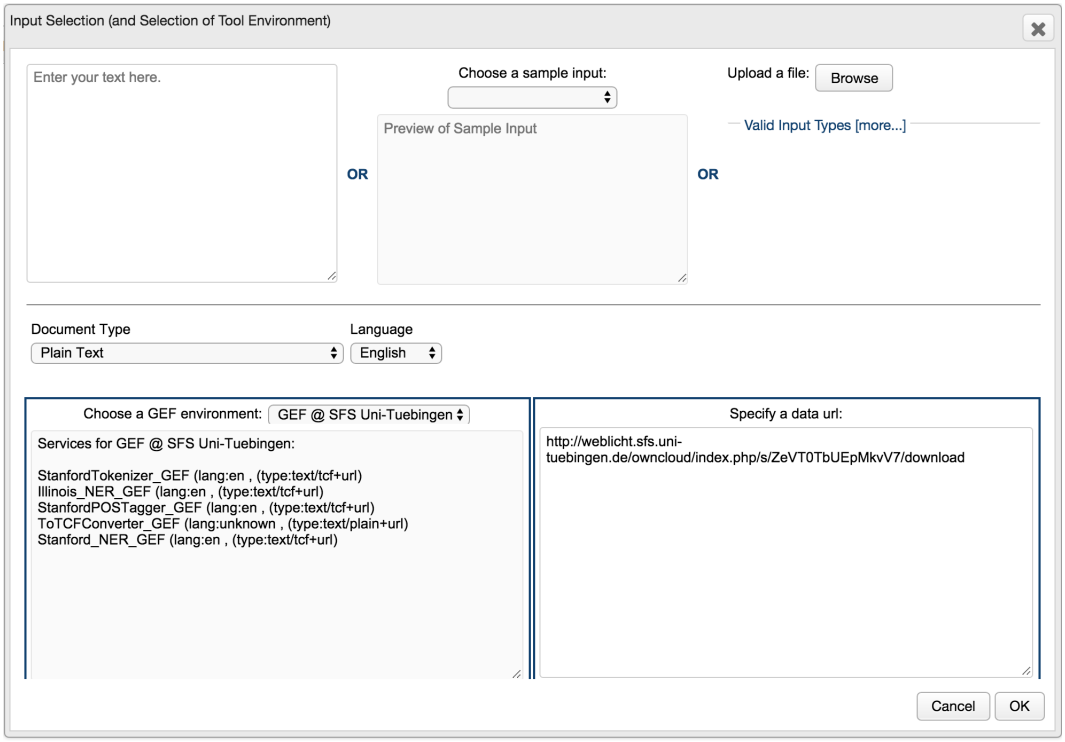


Figure 4: Screenshot of WebLicht GEF extension.

the output data locally (no transfer), and (v) returns this location with a corresponding URL. With the workflow consisting only of tools with such wrappers no data is sent back and forth to WebLicht, but rather URL-based pointers to the data. Note that the wrappers will be required to monitor the execution of the tools they wrap. That is, when an HTTP POST request yields an error message, then this message is sent back to the WebLicht orchestrator, which then prematurely terminates the execution of the workflow with an appropriate error message. Once, a tool is wrapped, it needs to get Dockerized and ingested into the GEF.

3.2. Liaising between WebLicht and the GEF

A translation mechanism is needed to liaise between WebLicht and the GEF-enabled services in a GEF environment. Consider a user wanting to annotate sensitive data, say an English text with named entities; for this, the user selects a GEF-based WebLicht workflow, whose corresponding XML representation is given in Fig. 5.

```
<cmd:Toolchain>
    <cmd:ToolInChain>
        <cmd:PID>http://hdl.handle.net/11858/00-1778-GEF-0004-BA56-7</cmd:PID>
        <cmd:Parameter value="en" name="lang"/>
        <cmd:Parameter value="text/plain" name="type"/>
    </cmd:ToolInChain>
    <cmd:ToolInChain>
        <cmd:PID>http://hdl.handle.net/11022/0000-GEF-2518-C</cmd:PID>
    </cmd:ToolInChain>
    <cmd:ToolInChain>
        <cmd:PID>http://hdl.handle.net/11022/0000-GEF-839F-9</cmd:PID>
    </cmd:ToolInChain>
</cmd:Toolchain>
```

Figure 5: The NER EasyChain in XML (fragment).

Each of the three tools in the tool chain is referenced with a persistent URL, pointing to metadata that describes the tool, see for instance, Fig. 6.

Stanford Tokenizer Stanford Tokenizer is a an efficient, fast, deterministic tokenizer.

Stanford Tokenizer is a an efficient, fast, deterministic tokenizer.

PID	http://hdl.handle.net/11022/0000-0000-2518-C		
URL	http://bridgit.sfs.uni-tuebingen.de/StanfordTokenizer		
Email	wlsupport@sfs.uni-tuebingen.de	Status	production
Created	2014-07-07T11:45:58.789+02:00	Modified	2014-07-07
Input		Output	
type	text/tcf+ url	sentences	
version	0.4	tokens	
text			
lang	en		

Figure 6: Metadata for the Stanford Tokenizer..

The metadata has a URL slot that is used to invoke the tool from WebLicht. Note that the URL does not point to the GEF environment, but rather to the translation mechanism, see below. Also note that the type of the input is not "text/tcf+xml" but "text/tcf+url", indicating that WebLicht will need to invoke the tool by passing a reference to the data to the tool, rather than the actual data.

Fig. 7 summarises the interaction between the WebLicht orchestrator and the GEF environment. As with all other web services connected to WebLicht, the orchestrator invokes the given URL with a synchronous POST request. Given the new type "text/tcf+url", no input data is being passed, only a URI reference to the data. The synchronous request invokes BridgIt, the translation mechanism, which liaises with the GEF environment. BridgIt keeps a record of:

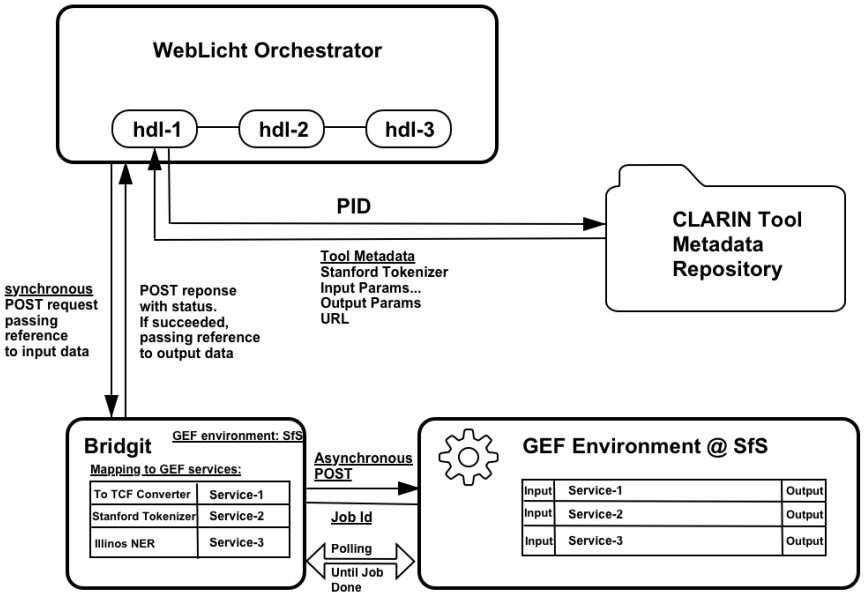


Figure 7: Architecture of WebLicht - GEF interaction.

- a pointer to the GEF environment, say, `http://gef.sfs.uni-tuebingen.de:8443`.
- a JSON-based table that maps URL-encoded information such as "/StanfordTokenizer" to a service code that identifies the corresponding GEF service.

With this information the URL `http://gef.sfs.uni-tuebingen.de:4041/jobs?serviceId=<id>&pid=<pointerToData>` is constructed and BridgIt sends this URL as POST request to start the GEF service. The given GEF environment answers with a JSON structure that gives the job identifier of the corresponding GEF service. Bridgit then polls the GEF environment at regular intervals (200 ms) to check whether the job has terminated. This is a GET request of the form `http://gef.sfs.uni-tuebingen.de:4041/jobs/<jobId>`. This GET request returns all information about the GEF job with the given jobID: the job's id, state, and output volume (for the result of the processing). When the job terminated successfully, then BridgIt returns a reference to the output volume associated with the job id; otherwise BridgIt passes on the error code encoded in the response of the GET request. Any result is returned to WebLicht as response to the synchronous POST request to BridgIt.

4. GEF Perspective

Setting-up a GEF environment comes at a substantial cost. Technical and logistical support is needed to keep costs manageable. From the perspective of a GEF community manager, it is highly desirable that all tools he or she wants to integrate into a GEF environment are pre-packaged. The dockerization of a tool requires intricate knowledge about the tool and its best runtime environment. In general, such information is rarely available. As a consequence, we assume a GEF community manager to contact the WebLicht developers and asks them for support. The step from a Dockerized web service and the GEF-wrapper is small.

Setting-up a GEF environment must also include the provision of metadata records that describe all GEF-ified tools using the CMDI profile `WebLichtWebService`. The metadata must be included in the GEF host's CLARIN centre repository so that WebLicht can harvest this data, and hence knows about the existence of the GEF environment. For this, the existing metadata of the original web service should be adapted accordingly.

The GEF community manager must also complement the GEF environment with the translation mechanism. Here, we assume that a reference implementation of BridgIt will be provided by the WebLicht team so that a GEF admin can use and configure the translation mechanism easily.

Given the required expertise and high cost of setting-up a GEF environment, the WebLicht team may provide GEF environments for popular natural language processing chains. Once a GEF environment has been set-up, the cost of moving it to a new location (close to the data) should come at relatively cheap cost.

5. The WebLicht Perspective

The standard, browser-based use of WebLicht [U6] does not handle big data well; it imposes file size limits and the processing is threatened by session timeouts. For files larger than 3 MB, users are advised to us WebLicht as a Service (WaaS), which is a REST service that executes WebLicht chains [U9]. WaaS allows users to run WebLicht chains from their UNIX shell, scripts, or programs, and hence, users are not exposed to browser-based timeouts, which vary across browsers. Note however, that chain

execution in WaaS produces the same data flow than the browser-based version, see Fig. 3. That is, data is sent back and forth to all web services in the chain via HTTP post requests, which is expensive. The work described in this paper improves the situation for larger files as the WebLicht orchestator now invokes GEFified tools by sending them references to the data rather then the actual data. But there is ample potential for optimization. The wrapping of services described earlier introduces an unnecessary overhead. Here, it is advisable to Dockerize the initial tools (that is, not their REST-based variants), and have them accessing their input via mounted Docker volumes (instead of responding to HTTP requests).

Another path for optimization is WebLicht's TCF format, which is rather verbose when compared with other competing formats (Lapponi et al., 2014). As a matter of fact, each tool in a WebLicht workflow enriches the TCF input with its processing result. Reconsider the easy-chain displayed in Fig. 3. Here, the Stanford Tokenizer tokenizes all input as preprocessed by the TCF converter, and then adds all tokens to the TCF file; this stage hence more than doubles the data, which is then sent to the Illinois Named Entity Recognizer. Assuming that the Illionois NER relies on the tokens only, the original input could have been consumed by the Stanford Tokenizer, nearly halving the data then sent to the next stage. Here, WebLicht's original design rationale is to blame. It regards all web services connected to WebLicht as black boxes and abstracts from their inner workings. For big data, this abstraction may prove too wasteful. Future tool metadata descriptions might need to specify whether a tool, say, requires only tokenized input or whether they also need access to the original input. WebLicht's workflow engine would then invoke the services with a 'consume' flag set to true or false, respectively.

In addition, the TCF format itself could be compacted in many different ways without inflicting any information loss. A JSON-based representation of TCF, for instance, promises to cut space requirements by half.

In the paper, WebLicht breaks out of its current boundaries. It empowers users to not only select the tools for their workflow, it lets them now also choose where the tools should run (see the selection of the GEF environment in Fig. 4). Here, more flexibility can be added by allowing *mixed* workflows: here, the first stages of the workflow, for instance, must be processed by in-house (*i.e.*, GEFified) tools, while the latter processing stages could be processed by non-GEF services (assuming that the original input is 'consumed' by the prior stages, see above).

To use WebLicht and its associated tools, users need to be authenticated with an account of the CLARIN Service Provider Federation. For users to process sensitive data with WebLicht, a future version will need to ask users for additional credentials. For this, reconsider, Fig. 4. When users specify a data url for the input data (bottom right), WebLicht could mediate users' credentials with the data site to check whether a given user is authorized to access the data. Only after successful AAI, users will be delegated to the next GUI page to identify or construct workflows.

6. Discussion and Conclusion

We are in the process of setting-up a proof of concept GEF environment for a popular language processing workflow (Named Entity Recognition for English). So far, we have dockerized all three tools of this workflow (including their service wrapping). This is the essential step towards "movable computation". The BridgIT liaison device is currently being developed and tested, and CMDI-based metadata descriptions for GEF-enabled services are being authored and made available for harvest by WebLicht. For testing purposes, we will create a WebLicht branch that offers GEF-enabled services to users.

Our work has already had a positive impact on the development of both WebLicht and GEF. To limit the amount of data transfer between the original WebLicht orchestrator and the individual services that define a workflow, it is now being discussed whether services shall also be invokable by passing references to the data rather than by posting the data to them. Our work also informed the EUDAT-GEF development team to ensure that user requirements stemming from the WebLicht use case lead to GEF feature requests that will find their way into the official GEF specification and implementation. One feature request concerns the invocation of a GEF service. At the time of writing, starting a GEF-ified tool means starting a new Docker container. Some tools, however, consume significant resources at start-up. The Illinois Named Entity Recognizer, for instance, loads large models to inform its processing. Here, it is more advisable to not terminate the GEF service (that is, to not stop the Docker container) once NER processing finished. In this case, the GEF service should continue listening to and serving future incoming processing requests.

We have started our article with GEF's underlying motivation that datasets have become much larger than the tools that process them, or that there are datasets that are not allowed to leave their hosting institition for legal reasons. Moving the tools to the data rather than the data to the tools seems reasonable. In linguistics, there is sensitive data, but big data issues become more prominent, too. Take, for instance, the Newsreader project with the aim to parse 100.000+ news articles live on a daily basis (Vossen et al., 2016). For these projects, a future version of WebLicht based on our approach can play a key role in orchestrating and executing a variety of workflows to gather, collect and post-process such data. The integration of the CLARIN WebLicht workflow engine and its services with EUDAT's Generic Execution Framework makes it possible to bring the language processing tools to an execution environment that also hosts the data, hence allowing language processing of sensitive and big data.

7. Acknowledgements

This work has been supported by the EUDAT (grant agreement No. 654065) and CLARIN-PLUS (grant agreement no. 676529) projects, both funded from the European Union's Horizon 2020 research and innovation programme. We would like to thank the anonymous reviewers for their valuable feedback.

8. URL References

[U1] https://www.docker.com/what-docker
[U2] https://www.eudat.eu/eudat-cdi/about
[U3] https://github.com/EUDAT-GEF/GEF
[U4] https://golang.org
[U5] https://facebook.github.io/react
[U6] https://weblicht.sfs.uni-tuebingen.de
[U7] https://github.com/weblicht
[U8] https://eudat-gef.mpimet.mpg.de
[U9] https://weblicht.sfs.uni-tuebingen.de/WaaS

9. Bibliographical References

Chernov, A., Hinrichs, E., and Hinrichs, M. (2017). Search your own treebank. In *Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories (TLT15)*, pages 25–34. Bloomington, IN, USA.

Dima, E., Pagé, C., and Budich, R. (2015). D7.5.2: Technology Adaptation and Development Framework (final). Technical report, EUDAT deliverable.

EUDAT. (2017). The EUDAT Collaborative Data Infrastructure (CDI). See the website at https://eudat.eu/eudat-collaborative-data-infrastructure-cdi.

Hinrichs, E., Hinrichs, M., and Zastrow:, T. (2010). Weblicht: Web-Based LRT Services for German. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (System Demonstrations)*.

Lapponi, E., Velldal, E., Oepen, S., and Knudsen, R. L. (2014). Off-road laf: Encoding and processing annotations in nlp workflows. In N. Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A.-L., Aprosio, A. P., Rigau, G., Rospocher, M., and Segers, R. (2016). Newsreader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. *Knowledge-Based Systems*, 110:60 – 85.

Building a Web-Scale Dependency-Parsed Corpus from Common Crawl

Alexander Panchenko[‡], Eugen Ruppert[‡], Stefano Faralli[†], Simone P. Ponzetto[†], Chris Biemann[‡]
‡ University of Hamburg, Department of Informatics, Language Technology Group, Germany
† University of Mannheim, School of Business Informatics and Mathematics, Data and Web Science Group, Germany
{panchenko,ruppert,biemann}@informatik.uni-hamburg.de
{stefano,simone}@informatik.uni-mannheim.de

Abstract

We present DEPCC, the largest-to-date linguistically analyzed corpus in English including 365 million documents, composed of 252 billion tokens and 7.5 billion of named entity occurrences in 14.3 billion sentences from a web-scale crawl of the COMMON CRAWL project. The sentences are processed with a dependency parser and with a named entity tagger and contain provenance information, enabling various applications ranging from training syntax-based word embeddings to open information extraction and question answering. We built an index of all sentences and their linguistic meta-data enabling quick search across the corpus. We demonstrate the utility of this corpus on the verb similarity task by showing that a distributional model trained on our corpus yields better results than models trained on smaller corpora, like Wikipedia. This distributional model outperforms the state of art models of verb similarity trained on smaller corpora on the SimVerb3500 dataset.

Keywords: text corpus, Web as a corpus, Common Crawl, dependency parsing, verb similarity, distributional semantics

1. Introduction

Large corpora are essential for the modern data-driven approaches to natural language processing (NLP), especially for unsupervised methods, such as word embeddings (Mikolov et al., 2013) or open information extraction (Banko et al., 2007) due to the "unreasonable effectiveness of big data" (Halevy et al., 2009). However, the size of commonly used text collections in the NLP community, such as BNC[1] or Wikipedia is in the range 0.1–3 billion tokens, which potentially limits coverage and performance of the developed models. To overcome this limitation, larger corpora can be composed of books, e.g. in (Goldberg and Orwant, 2013) a dataset of syntactic n-grams[2] was built from the 345 billion token corpus of the Google Books project.[3] However, access to books is often restricted, which limits use-cases of book-derived datasets. Another source of large amounts of texts is the Web. Multiple researchers investigated the use of the Web for construction of text corpora, producing resources, such as PUKWAC (Baroni et al., 2009) (2 billion of tokens) and ENCOW16 (Schäfer, 2015) (17 billion of tokens), yet the size of these corpora is still at least one order of magnitude smaller than the web-scale crawls, e.g. CLUEWEB[4] and COMMON CRAWL[5]. On the other hand, directly using the web crawl dumps is problematic for researchers as: (1) the documents are not preprocessed, containing irrelevant information, e.g. HTML markup; (2) big data infrastructure and skills are required; (3) (near)duplicates of pages disbalance the corpus; (4) documents are not linguistically analyzed, thus only shallow models can be used. The mentioned factors substantially limit the use of web-scale corpora in natural language processing research and applications.

The objective of this work is to address these issues and *make access to web-scale corpora a commodity* by providing a web-scale corpus that is ready for NLP experiments as it is linguistically analyzed and cleansed from noisy irrelevant content. Namely, in this paper, we present a technology for constructing of linguistically analyzed corpora from the Web and release DEPCC, the largest-to-date dependency-parsed corpus of English texts.

The COMMON CRAWL project regularly produces web-scale crawls featuring a substantial fraction of all public web pages. For instance, as of October 2017, the estimated number of pages on the Web is 47 billion[6], while the corresponding crawl contains over 3 billion pages. To put this number into perspective, according to the same source, the indexed Web contains about 5 billion pages.

COMMON CRAWL provides the data in the Web ARChive (WARC) format. Each crawl is provided in the raw form features full HTML pages with metadata or in the form of preprocessed archives containing texts (WET). The WET archives contain extracted plaintext from the raw crawls. For instance, the 29.5 Tb raw crawl archive (cf. Table 2) has a corresponding 4.8 Tb WET version with texts. The preprocessing used for producing the WET archives is limited to removal of HTML tags. After a manual check, we also noticed that in WET archives (1) some documents still contain HTML markup; (2) the archives contain document duplicates; (3) documents are written in various languages making it difficult to train language-specific linguistic models. Finally, most importantly, the WET dumps are not linguistically analyzed, which significantly limits their utility for language processing applications.

In this work, we address these limitations by constructing a text corpus from COMMON CRAWL, which is filtered from irrelevant and duplicate documents and is linguistically an-

[1] http://www.natcorp.ox.ac.uk

[2] http://commondatastorage.googleapis.com/books/syntactic-ngrams/index.html

[3] https://books.google.com

[4] http://lemurproject.org/clueweb12

[5] http://www.commoncrawl.org

[6] http://www.worldwidewebsize.com at 02.10.2017

	WaCkypedia	Wikipedia	PukWaC	GigaWord	ENCOW16	ClueWeb12	Syn.Ngrams	DepCC
Tokens (billions)	0.80	2.90	1.91	1.76	16.82	N/A	345.00	251.92
Documents (millions)	1.10	5.47	5.69	4.11	9.22	733.02	3.50	364.80
Type	Encyclop.	Encyclop.	Web	News	Web	Web	Books	Web
Source texts	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Preprocessing	Yes	No	Yes	No	Yes	No	No	Yes
NER	No	No	No	No	Yes	No	No	Yes
Dependency-parsed	Yes	No	Yes	No	Yes	No	Yes	Yes

Table 1: Comparison of existing large text corpora for English with the DepCC corpus.

alyzed. Namely, the contributions of this paper are the following:

1. We present a *methodology* for the creation of the text corpus from the web-scale crawls of Common Crawl.
2. We present a *software* implementing the methodology in a scalable way using the MapReduce framework.
3. We present the largest-to-date *dependency parsed corpus* of English texts obtained using the developed methodology, also featuring *named entity tags*.
4. We show the utility of the web-scale corpora on the *verb similarity* task by outperforming the state of the art on the SimVerb3500 dataset (Gerz et al., 2016).

The corpus and the software tools are available online.[7] Namely, the corpus can be directly used without the need to download it on the Amazon S3 distributed file system, cf. Section 3.7.[8] The software tools used to build the corpus are distributed under an open source license. The terms of use of the corpus are described in Section 4.

2. Related Work

2.1. Large Scale Text Collections

In Table 1 we compare the DepCC corpus to seven existing large-scale English corpora, described below.

WaCkypedia (Baroni et al., 2009) is a parsed version of English Wikipedia as of 2009. The articles are part-of-speech tagged with the TreeTagger (Schmid, 1994) and dependency parsed with the Malt parser (Nivre et al., 2007). Similarly to our corpus, the results are presented in the CoNLL format.[9] The 2017 version of Wikipedia contains three times more tokens, compared to the version of 2009, yet the there are no distributions of linguistically analyzed recent dumps. PukWaC is a dependency-parsed version of the ukWaC corpus (Baroni et al., 2009), which is processed in the same way as the WaCkypedia corpus.

GigaWord (Parker et al., 2011) is a large corpus of newswire, which is not dependency parsed. The ClueWeb12 is a corpus similar to the raw Common Crawl corpus: it contains archives of linguistically unprocessed web pages.

Stage of the Processing	**Size (.gz)**
Input raw web crawl (HTML, WARC)	29,539.4 Gb
Preprocessed corpus (simple HTML)	832.0 Gb
Preprocessed corpus English (simple HTML)	683.4 Gb
Dependency-parsed English corpus (CoNLL)	2,624.6 Gb

Table 2: Various stages of development of the corpus based on the Common Crawl 2016-07 web crawl dump.

The authors of the Google Syntactic Ngrams corpus (Goldberg and Orwant, 2013) parsed a huge collection of books and released a dataset of syntactic dependencies. However, the source texts are not shared due to copyright restrictions, which limits potential use-cases of this resource.

Finally, ENCOW16 (Schäfer, 2015) is a large-scale web corpus, which is arguably the most similar one to DepCC. The authors also rely on the Malt parser and perform named entity tagging. However, this corpus contains roughly 15 times less tokens than DepCC.

2.2. Common Crawl as a Corpus

Kolias et al. (2014) present an exploratory study of one of the early versions of the Common Crawl. The authors provide various descriptive statistics of the dataset regarding language distribution, formats of the documents, etc.

Common Crawl was used construct a large-scale Finnish Parsebank consisting of 1.5 billion tokens in 116 million sentences (Laippala and Ginter, 2014). The texts were morphologically and syntactically analyzed. In addition, distributional vector space representations of the words were obtained using the word2vec toolkit (Mikolov et al., 2013). The resources were made available under an open license.

GloVe (Pennington et al., 2014) is an unsupervised model for learning distributional word representations similar to word2vec. The authors distribute[10] two models trained on the English part of a Common Crawl corpus (comprising respectively 42 and 820 billion of tokens), which are often used to build neural NLP systems, such as (Tsuboi, 2014). The models were trained on the Common Crawl documents texts tokenized with the Stanford tokenizer. In addition, the smaller training corpus was lowercased.

[7] https://www.inf.uni-hamburg.de/en/inst/ab/lt/resources/data/depcc.html

[8] https://commoncrawl.s3.amazonaws.com/contrib/depcc/CC-MAIN-2016-07/index.html

[9] http://www.universaldependencies.org/format.html

[10] https://nlp.stanford.edu/projects/glove

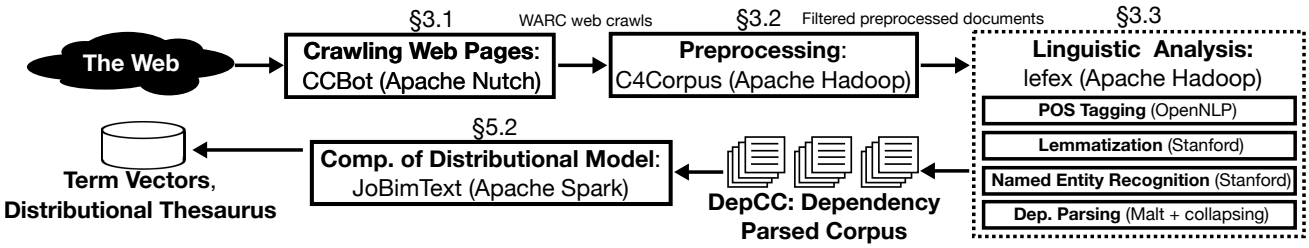


Figure 1: Outline of the corpus construction approach and experiments described in the paper.

3. Building a Web-Scale Dependency-Parsed Corpus in English from Common Crawl

Figure 1 shows how a linguistically analyzed corpus is built from the Web. First, web pages are downloaded by the web crawler of COMMON CRAWL, called CCBot. Second, preprocessing, involving elimination of duplicates and language detection, is performed using the C4Corpus tool. Finally, we perform linguistic analysis of the corpus and save the results in the CoNLL format (cf. Section 3.4.).

3.1. Input Web Crawl: the Common Crawl

The DEPCC corpus is based on the crawl of February 2016[11] containing more than 1.73 billion URLs. The COMMON CRAWL URL index for this crawl is available online[12], while the original files are located in the "commoncrawl" bucket on the S3 distributed file system.[13] As summarized in Table 2, the total size of the compressed HTML WARC files is about 30 Tb.

3.2. Preprocessing of Texts: the C4Corpus Tool

The raw corpus was processed with the C4Corpus tool (Habernal et al., 2016) and is available on the distributed cloud-based file system Amazon S3.[14] The tool performs preprocessing of the raw corpus, in five phases:

1. Language detection, license detection, and removal of boilerplate page elements, such as menus;
2. "Exact match" document de-duplication;
3. Detecting near duplicate documents;
4. Removing near duplicate documents;
5. Grouping the final corpus by language and license.

The resulting output is a gzip-compressed corpus with a total size of 0.83 Tb (cf. Table 2). For further processing, we selected only English texts with the total size of 0.68 Tb, based on the language detection in the first phase. Note that we use all texts written in English, not only those published under the CC-BY license.

3.3. Linguistic Analysis of Texts

Linguistic analysis consists of four stages presented in Figure 1 and is implemented using the Apache Hadoop framework[15] for parallelization and the Apache UIMA framework[16] for integration of linguistic analysers via the DKPro Core library (Eckart de Castilho and Gurevych, 2014).[17]

3.3.1. POS Tagging and Lemmatization

For morphological analysis of texts, we used OpenNLP part-of-speech tagger and Stanford lemmatizer.

3.3.2. Named Entity Recognition

To detect occurrences of persons, locations, and organizations we use the Stanford NER tool (Finkel et al., 2005).[18] Overall, 7.48 billion occurrences of named entities were identified in the 251.92 billion tokens output corpus.

3.3.3. Dependency Parsing

To make large-scale parsing of texts possible, a parser needs to be not only reasonably accurate but also fast. Unfortunately, the most accurate parsers, such as Stanford parser based on the PCFG grammar (De Marneffe et al., 2006), according to our experiments, take up to 60 minutes to process 1 Mb of text on a single core, which was prohibitively slow for our use-case (details of the hardware configuration are available in Section 3.5.). We tested all versions of the Stanford, Malt (Hall et al., 2010), and Mate (Ballesteros and Bohnet, 2014) parsers for English available via the DKPro Core framework. To dependency-parse texts, we selected the Malt parser, due to an optimal ratio of efficiency and effectiveness (parsing of 1 Mb of text per core in 1–4 minutes). This parser was successfully used in the past for the construction of linguistically analyzed web corpora, such as PUKWAC (Baroni et al., 2009) and ENCOW16 (Schäfer, 2015). While more accurate parsers exist, e.g. the Stanford parser, according to our experiments, even the neural-based version of this parser is substantially slower. On the other hand, as shown by Chen and Manning (2014), the performance of the Malt parser is only about 1.5–2.5 points below the neural-based Stanford parser. In particular, we used the stack model based on the projective transition system with the Malt.[19]

[11] http://commoncrawl.org/2016/02/

[12] http://index.commoncrawl.org/CC-MAIN-2016-07

[13] s3://commoncrawl/crawl-data/CC-MAIN-2016-07

[14] s3://commoncrawl/contrib/c4corpus/CC-MAIN-2016-07

[15] https://hadoop.apache.org

[16] https://uima.apache.org

[17] https://github.com/uhh-lt/lefex

[18] stanfordnlp-model-ner-en-all.3class.distsim.crf, 20.04.2015

[19] The used model is de.tudarmstadt.ukp.dkpro.core.maltparser-upstream-parser-en-linear, version 20120312.

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	FEATS	HEAD	DEPREL	DEPS	NER
# newdoc url = http://www.poweredbyosteons.org/2012/01/brief-history-of-bioarchaeological.html									
# newdoc s3 = s3://aws-publicdatasets/common-crawl/crawl-data/CC-MAIN-2016-07/segments...									
...									
# sent_id = http://www.poweredbyosteons.org/2012/01/brief-history-of-bioarchaeological.html#60									
# text = The American Museum of Natural History was established in New York in 1869.									
0	The	the	DT	DT	_	2	det	2:det	O
1	American	American	NNP	NNP	_	2	nn	2:nn	B-Organization
2	Museum	Museum	NNP	NNP	_	7	nsubjpass	7:nsubjpass	I-Organization
3	of	of	IN	IN	_	2	prep	_	I-Organization
4	Natural	Natural	NNP	NNP	_	5	nn	5:nn	I-Organization
5	History	History	NNP	NNP	_	3	pobj	2:prep_of	I-Organization
6	was	be	VBD	VBD	_	7	auxpass	7:auxpass	O
7	established	establish	VBN	VBN	_	7	ROOT	7:ROOT	O
8	in	in	IN	IN	_	7	prep	_	O
9	New	New	NNP	NNP	_	10	nn	10:nn	B-Location
10	York	York	NNP	NNP	_	8	pobj	7:prep_in	I-Location
11	in	in	IN	IN	_	7	prep	_	O
12	1869	1869	CD	CD	_	11	pobj	7:prep_in	O
13	.	.	.	.	_	7	punct	7:punct	O
...									

Table 3: An excerpt from an output document in the CoNLL format: a document header plus a sentence are shown. Here, "ID" is a word index, "FORM" is word form, "LEMMA" is lemma or stem of word form, "UPOSTAG" is universal part-of-speech tag, "XPOSTAG" is language-specific part-of-speech tag, "FEATS" is a list of morphological features, "HEAD" is head of the current word, which is either a value of ID or zero, "DEPREL" is universal dependency relation to the "HEAD", "DEPS" is enhanced dependency graph in the form of head-deprel pairs, and "NER" is named entity tag.

The text downloaded from the Web has highly variable quality due to the inherent nature of user-generated content, but also unavoidable pre-processing errors, e.g. during the cleanup of incomplete HTML markup. To avoid crashes of the dependency parser caused by excessively long sentences, we filter all sentences longer than 50 tokens. Our manual analysis revealed that there are hardly any well-formed sentences of 50 tokens or more in this corpus.

3.3.4. Collapsing of Syntactic Dependencies

Collapsed and enhanced dependencies, such as the Stanford Dependencies (De Marneffe et al., 2006)[20] can be useful in various NLP tasks as they provide a more compact syntactic trees of a sentence, compared to the original dependency tree, thus reducing sparsity of syntax-aware representations.

To compensate the lack of the dependency enhancement in Malt, we use the system of (Ruppert et al., 2015)[21] to perform collapsing and enhancing of dependencies. The authors of the toolkit shown that (1) using the collapsed dependency representations substantially improves quality of construction of distributional thesauri based on sparse syntactic features; (2) the performance of the Stanford enhanced dependencies and the collapsed Malt dependencies on the same task are comparable. The advantage of using Malt with an external collapsing with respect to the Stanford parser, in our case, is speed.

Note that, both original and enhanced versions are saved respectively into the columns "DEPREL" and "DEPS" as illustrated in Table 3.

[20] https://nlp.stanford.edu/software/stanford-dependencies.shtml

[21] http://jobimtext.org/dependency-collapsing

3.4. Format of the Output Documents

The documents are encoded in the CoNLL format as illustrated in Table 3. The corpus is released as a collection of 19,101 gzip-compressed files.

Each file is relatively small (around 150Mb) and is easy to download and work with locally during the development phase. However, to work with the entire corpus we recommend using some kind of parallelism, e.g. based on multiprocessing/multithreading or frameworks for distributed computing, such as Apache Hadoop/Spark/Flink.

3.5. Computational Settings

The linguistic analysis was performed on an Apache Hadoop 2.6 cluster using 341 containers each provided with one Intel Xeon CPU E5-2603v4@1.70GHz and 8Gb of RAM. The computational cluster consisted of 16 nodes plus a single head node. The job used 2.75 TB out of 2.82 TB available RAM and 356 out of 640 available Vcores.

3.6. Running Time

In total, the computations were completed in 110 hours in 19101 tasks each processing a block of 100 Mb input data. The median running time of one task was 1 hour 10 minutes. This corresponds to the processing time of about 1.4 Mb/min for such a median task and 0.84 Mb/min for on average for the entire corpus, including compression of the output CoNLL files (cf. Section 3.3.3.). The minimum time of processing of a task was 38 minutes while the maximum time was 9 hours and 4 minutes.

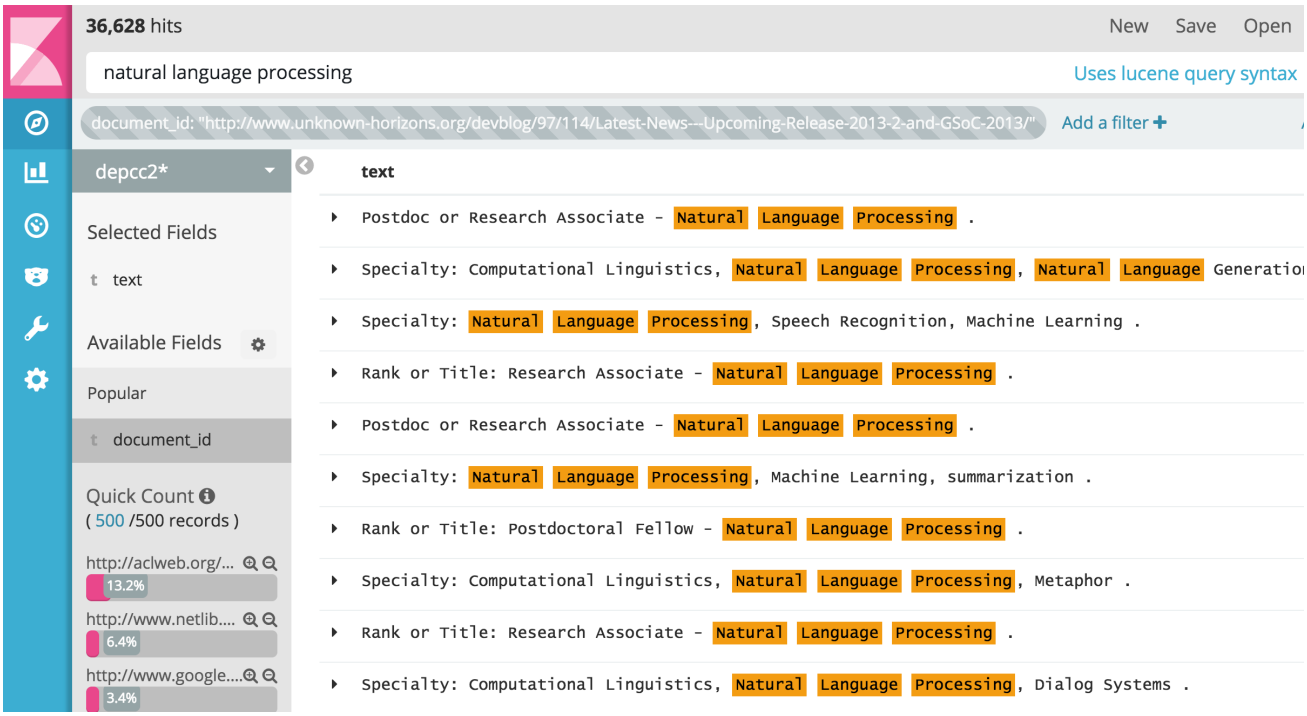


Figure 2: Interactive graphical user interface providing a full text search over 14.3 billion of sentences in the DEPCC corpus and their linguistic meta-data. A user can search for sentences containing specified keywords, named entities or syntactic dependencies.

3.7. Using the Corpus in the Amazon Cloud

The COMMON CRAWL datasets are hosted in the Amazon computing cloud platform.[22] As mentioned in the introduction, our corpus is also was made directly available on Amazon S3 distributed file system in the `us-east-1` region (US East, North Virginia) as a part of the COMMON CRAWL contributed datasets. This means that *you do not need to download the corpus* to be able to work with it. Instead, you can run the jobs directly against the respective bucket on the Amazon S3 file system which contains the DEPCC corpus. For optimal performance, you need to run instances which perform computations with the corpus inside the Amazon cloud (e.g. using the EC2[23] or EMR[24] services) in the `us-east-1` region.

3.8. Index of the Corpus

An access to a full-text search of all 14.3 billion sentences and their dependency relations of the DEPCC corpus is available upon request. This service is free and aims at facilitating access to the corpus for use-cases that do not require download of the entire collection of documents. Using the index, users can quickly retrieve sentences matching various linguistic criteria (using the Lucene query syntax[25]), e.g. presence of keyword in a sentence, presence of a specific named entity in a sentence, presence of a specific dependency relation, provenance of a document from a specific web domain, etc. Each retrieved sentence contains all the meta-data depicted in Table 3, such as the provenance of the sentence.

The corpus can be queried via a RESTful API based on the ElasticSearch search engine[26] or via a web-based graphical user interface based on Kibana[27] graphical interface to ElasticSearch. Results of a sample query visualized using Kibana are presented in Figure 2.

We do not distribute the index itself due to its huge size. However, users can re-create the index using the open source software provided as a part of the JoBimText package[28] from the the CoNLL files. While these compressed CoNLL files occupy only around 2.6 Tb, the size of the full index is about 15 Tb or more, depending on the replication factor of ElasticSearch. For this reason, for most practical applications, re-creation of the index is faster and more straightforward than download of a pre-computed index and its subsequent deployment. Besides, some major versions of the ElasticSeach indices are not compatible between one another.

4. Terms of Use

The DEPCC corpus is based on a COMMON CRAWL dataset. We do not reserve any copyrights as the authors of this derivative resource, but while using the DEPCC corpus you need to make sure to respect the Terms of Use of the original COMMON CRAWL dataset it is based on.[29]

[22] `https://aws.amazon.com`
[23] `https://aws.amazon.com/ec2`
[24] `https://aws.amazon.com/emr`
[25] `https://www.elastic.co/guide/en/kibana/current/lucene-query.html`
[26] `https://www.elastic.co`
[27] `https://www.elastic.co/products/kibana`
[28] `https://github.com/uhh-lt/josimtext`
[29] `http://commoncrawl.org/terms-of-use`

Model	SimVerb3500	SimVerb3000	SimVerb500	SimLex222
Wikipedia+ukWaC+BNC: Count SVD 500-dim (Baroni et al., 2014)	0.196	0.186	0.259	0.200
PolyglotWikipedia: SGNS BOW 300-dim (Gerz et al., 2016)	0.274	0.333	0.265	0.328
8B: SGNS BOW 500-dim (Gerz et al., 2016)	0.348	0.350	0.378	0.307
8B: SGNS DEPS 500-dim (Gerz et al., 2016)	**0.356**	**0.351**	0.389	0.385
PolyglotWikipedia:SGNS DEPS 300-dim (Gerz et al., 2016)	0.313	0.304	**0.401**	**0.390**
Wikipedia: LMI DEPS wpf-1000 fpw-2000	0.283	0.284	0.271	0.268
Wikipedia+ukWac+GigaWord: LMI DEPS wpf-1000 fpw-2000	0.376	0.368	0.419	0.183
DEPCC: LMI DEPS wpf-1000 fpw-1000	0.400	0.387	**0.477**	0.285
DEPCC: LMI DEPS wpf-1000 fpw-2000	**0.404**	**0.392**	**0.477**	**0.292**
DEPCC: LMI DEPS wpf-2000 fpw-2000	0.399	0.388	0.459	0.268
DEPCC: LMI DEPS wpf-5000 fpw-5000	0.382	0.372	0.442	0.226

Table 4: Evaluation results on the verb semantic similarity task. Sparse count-based distributional models (LMI) trained on the DEPCC corpus are compared to models trained on the smaller corpora, such as Wikipedia and a combination of Wikipedia, UKWAC, and GIGAWORD. Rows and columns of each LMI-weighted distributional model are pruned as in (Biemann and Riedl, 2013): the *wpf* indicates the number of words per feature, and the *fpw* indicates the number of features per word. We also compare our models to the best verb similarity models from the state of the art. Here the "BOW" denotes models based on bag-of-word features, while "DEPS" denotes syntax-based models. SimVerb3000 and SimVerb500 are train and test partitions of the SimVerb3500, while the SimLex222 dataset is composed of verb pairs from the SimLex999 dataset. The best results in a section are boldfaced, the best results overall are underlined.

5. Evaluation: Verb Similarity Task

As an example of potential use-case, we demonstrate the utility of the corpus and the overall methodology on a verb similarity task.

This task structurally is the same as the word similarity tasks based on such datasets as SimLex-999 (Hill et al., 2015). Namely, a system is given two words as input and needs to predict a scalar value which characterizes semantic similarity of the input words. While in the word similarity task the input pairs are words of various parts of speeches (nouns, adjectives, etc.), in this paper we only consider verb pairs.

We chose this task since verb meaning is largely defined by the meaning of its arguments (Fillmore, 1982), therefore dependency-based features seem relevant for building distributional representations of verbs.

5.1. Datasets: SimVerb3500 and SimLex222

Recently a new challenging dataset for verb relatedness was introduced, called SimVerb3500 (Gerz et al., 2016). The dataset is composed of 3500 pairs of verbs and is split into the train and test parts, called respectively SimVerb3000 and SimVerb500. In addition to this benchmark, in our experiments, we also test the performance of the models on the SimLex222, which is the verb part of SimLex999 dataset (Hill et al., 2015) composed of 222 verb pairs. Historically, the SimVerb3500 dataset was created after the SimLex222, addressing its shortcomings related to the verb coverage. As in our experiments, we do not use the dataset SimVerb3000 for training, and to be consistent with the results reported in (Gerz et al., 2016) we report performance of the tested verb similarity models on all four datasets: SimVerb3500/3000/500, and SimLex222.

5.2. A Distributional Model for Verb Similarity

We compute syntactic count-based distributional representations of words using the JoBimText framework (Biemann and Riedl, 2013).[30] The sparse vectors are weighted using the LMI weighting schema and converted to unit length. In our experiments, we varied also the maximum number of salient features per word (*fpw*) and words per feature (*wpf*). Conceptually, each row and column of the sparse term-feature matrix is pruned such that at most *wpf* non-zero elements in a row and *fpw* elements in a column are retained.

5.3. Discussion of Results

Table 4 presents results of the experiments.

5.3.1. Baselines

The top part of the table lists five top systems in various categories (Gerz et al., 2016), representing the current state-of-art result on this dataset. Namely, the Count based SVD system is from (Baroni et al., 2014). In the original paper, two corpora were used: the "8B" is a 8 billion tokens corpus produced by a script in the word2vec toolkit, which gathers the texts from various sources (Mikolov et al., 2013) and the "PolyglotWikipedia" is the English Polyglot Wikipedia corpus consisting of 1.9 billion tokens (Al-Rfou et al., 2013).

We use the baselines in the top of the table to indicate the best results on the dataset: our goal is to show the impact of the large corpora on performance and not to present a new model for verb similarity.

5.3.2. Impact of the Corpora on Performance

The bottom part of the table presents the distributional model described in Section 5.2. trained on the corpora of various sizes. Note, that the preprocessing steps for each corpus are exactly the same as for the DEPCC corpus. We observe that the smallest corpus (Wikipedia) yields the worst results. While the scores go up on the larger corpus, which is a combination of Wikipedia with two

[30] https://github.com/uhh-lt/josimtext

other corpora, we can reach the even better result by training the model (with exactly the same parameters) on the dependency-based features extracted from the full DEPCC corpus. This model substantially outperforms also the prior state of the art models, e.g. (Baroni et al., 2014) and (Gerz et al., 2016), on the SimVerb dataset, through the sheer size of the input corpus, as previously shown, e.g. (Banko and Brill, 2001) inter alia.

5.3.3. Differences in Performance for Test/Train Sets

For the SimVerb dataset, the absolute performance on the test part (SimVerb500) is higher than the absolute performance on the train part (SimVerb300) for almost all models, including the baselines. We attribute this to a specific split of the data in the dataset: our models do not use the training data to learn verb representations.

6. Conclusion

In this paper, we introduced a new web-scale corpus of English texts extracted from the COMMON CRAWL, the largest openly available linguistically analyzed corpus to date, according to the best of our knowledge.

The documents were de-duplicated and linguistically processed with part-of-speech and named entity taggers and a dependency parser, making it possible to easily start large-scale experiments with syntax-aware models without the need of long and resource-intensive preprocessing. We built an index of sentences and their linguistic meta-data accessible though an interactive web-based search interface or via a RESTful API.

In our experiments on the verb similarity task, a distributional model trained on the new corpus outperformed models trained on the smaller corpora, like Wikipedia, reaching new state of the art of verb similarity on the SimVerb3500 dataset. The corpus can be used in various contexts, ranging from training of syntax-based word embeddings (Levy and Goldberg, 2014) to unsupervised induction of word senses (Biemann et al., 2018) and frame structures (Kawahara et al., 2014). A promising direction of future work is using the proposed technology for building corpora in multiple languages.

7. Acknowledgements

This research was supported by the Deutsche Forschungsgemeinschaft (DFG) under the project "Joining Ontologies and Semantics Induced from Text" (JOIN-T). We are grateful to Amazon for providing required computational resources though the "AWS Cloud Credits for Research" program. Finally, we thank Kiril Gashteovski and three anonymous reviewers for their most helpful feedback.

8. Bibliographical References

Ballesteros, M. and Bohnet, B. (2014). Automatic feature selection for agenda-based dependency parsing. In *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers (COLING'2014)*, pages 794–805, Dublin, Ireland.

Banko, M. and Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 26–33, Toulouse, France. Association for Computational Linguistics.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, volume 7, pages 2670–2676, Hyderabad, India. AAAI Press.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, MD, USA. Association for Computational Linguistics.

Biemann, C. and Riedl, M. (2013). Text: now in 2d! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Biemann, C., Faralli, S., Panchenko, A., and Ponzetto, S. P. (2018). A framework for enriching lexical semantic resources with distributional semantics. *Natural Language Engineering*, pages 1–48.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Dohar, Qatar. Association for Computational Linguistics.

De Marneffe, M.-C., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of Conference on Language Resources and Evaluation (LREC'2006)*, pages 449–454. Genova, Italy, ELRA.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Fillmore, C. J., (1982). *Frame semantics*, pages 111–137. Hanshin Publishing Co., Seoul, South Korea.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Halevy, A., Norvig, P., and Pereira, F. (2009). The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12.

Hall, J., Nilsson, J., and Nivre, J. (2010). Single malt or blended? a study in multilingual parser optimization. In *Trends in Parsing Technology*, pages 19–33. Springer.

Kawahara, D., Peterson, D. W., and Palmer, M. (2014). A step-wise usage-based method for inducing polysemy-aware verb classes. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1030–1040, Bal-

timore, MD, USA. Association for Computational Linguistics.

Kolias, V., Anagnostopoulos, I., and Kayafas, E. (2014). Exploratory analysis of a terabyte scale web corpus. *arXiv preprint arXiv:1409.5443*.

Levy, O. and Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 302–308, Baltimore, MD, USA. Association for Computational Linguistics.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., Harrahs and Harveys, NV, USA.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Ruppert, E., Klesy, J., Riedl, M., and Biemann, C. (2015). Rule-based Dependency Parse Collapsing and Propagation for German and English. In *Proceedings of GSCL*, pages 58–66, Essen, Germany.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Tsuboi, Y. (2014). Neural networks leverage corpus-wide information for part-of-speech tagging. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 938–950, Doha, Qatar. Association for Computational Linguistics.

9. Language Resource References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Gerz, D., Vulić, I., Hill, F., Reichart, R., and Korhonen, A. (2016). Simverb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182, Austin, TX, USA. Association for Computational Linguistics.

Goldberg, Y. and Orwant, J. (2013). A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Proceedings of *SEM in conjunction with NAACL-HLT*, pages 241–247, Portland, OR, USA. Association for Computational Linguistics.

Habernal, I., Zayed, O., and Gurevych, I. (2016). C4Corpus: Multilingual web-size corpus with free license. In *Proceedings of the Language Resources and Evaluation Conference (LREC'2016)*, pages 914–922, Portorož, Slovenia. ELRA.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with genuine similarity estimation. *Computational Linguistics*, 41(4):665–695.

Laippala, V. and Ginter, F. (2014). Syntactic N-gram collection from a large-scale corpus of internet Finnish. In *Human Language Technologies-The Baltic Perspective: Proceedings of the Sixth International Conference Baltic HLT 2014*, volume 268, page 184, Kaunas, Lithuania. IOS Press.

Parker, R., Graff, D., Kong, J., Chen, K., and Maeda, K. (2011). English Gigaword. *Linguistic Data Consortium*.

Schäfer, R. (2015). Processing and querying large web corpora with the COW14 architecture. In Piotr Baiaski, et al., editors, *Proceedings of Challenges in the Management of Large Corpora 3 (CMLC-3)*, pages 28–34, Lancaster. UCREL, IDS.

Universal Dependencies Version 2 for Japanese

Masayuki Asahara♡†, Hiroshi Kanayama♠, Takaaki Tanaka♣
Yusuke Miyao♢, Sumire Uematsu♢, Shinsuke Mori○
Yuji Matsumoto●, Mai Omura♡, Yugo Murawaki○
♡ National Institute for Japanese Language and Linguistics, ♠ IBM Research AI
♣ NTT Communication Science Labolartories, ♢ National Institute of Informatics
○ Kyoto University, ● Nara Institute of Science and Technology
masayu-a@ninjal.ac.jp†

Abstract

In this paper, we are presenting our work to build Universal Dependencies (UD) resources for Japanese. The UD Japanese resources are built based on automatic conversion from several treebanks. The word delimitation, POS, and syntactic relations of the existing treebanks are ported for the UD annotation scheme. We discuss the issues of the UD scheme found through porting of the Japanese language.

1. Introduction

The Universal Dependencies (UD) project (McDonald et al., 2013) has defined a consistent, cross-linguistic target and syntactic structure representation format. In this presentation, we will show the work of the UD Japanese team. The UD Japanese team was organised by interested people who are developing their own treebanks or parsers. We developed and maintained several UD guidelines (version 2.0) compatible data for Japanese. Most of the data are made through automatic conversion from the existing treebank. The UD annotation guidelines were updated from version 1 to version 2 in early 2017. The automatic conversion enabled us to adapt the existing annotation based on traditional Japanese grammar conventions for the UD annotation guidelines changes. In this paper, we discuss the current issues of UD Japanese resources until today. These issues come from the difficulty to perform cross-linguistically consistent annotation for the different grammatical system from western European languages. The points at the issues related to the conversions are split into the delimitation (word, phrase and clause), undefined policies of UD guidelines, typological systems for UD, and copyright of Japanese language resources.

2. UD Japanese Resources

We have the following UD Japanese resources: UD Japanese-BCCWJ, UD Japanese-KTC (Tanaka et al., 2016), UD Japanese-GSD, UD Japanese-PUD, and UD Japanese-Modern (Omura et al., 2017). Table 1 presents the current status of UD Japanese resources. Below, we describe these resources briefly.

UD Japanese-BCCWJ is UD data based on the 'Balanced Corpus of Contemporary Written Japanese' (hereafter BCCWJ) (Maekawa et al., 2014). The BCCWJ defines 1 million word-scale core data samples in which the morphological information is manually annotated with three layers of word delimitations: Short Unit Word (SUW), Long Unit Word (LUW), and *bunsetsu*. The BCCWJ has several syntactic annotations. The BCCWJ-DepPara (Asahara and Matsumoto, 2016) is a *bunsetsu*-based syntactic dependency and coordinate structure annotation. The BCCWJ-PAS (Ueda et al., 2015) is a predicate-argument relation annotation with the NAIST Text Corpus annotation schema (Iida et al., 2007). We maintain conversion rules based on these annotations.

UD Japanese-KTC (Tanaka et al., 2016) is based on the NTT Japanese Phrase Structure Treebank (Tanaka and Nagata, 2013) which contains the same original text as the Kyoto Text Corpus (KTC) (Kurohashi and Nagao, 2003). KTC is a *bunsetsu*, namely base phrase, based dependency treebank with its own word delimitation schema and POS tagset. The NTT Japanese Phrase Structure Treebank is a phrase structure-based treebank. The word delimitation and POS are adapted to the UniDic SUW standard. The data is still in version 1.0 schema as of February 2018. We are now modifying UD Japanese KTC from version 1.0 schema to version 2.0.

UD Japanese-GSD (formerly known as UD Japanese) consists of sentences from Wikipedia. The version 2.0 of this annotated corpus was provided for the CoNLL 2017 Shared Task (Zeman et al., 2017). In the release of version 2.0, the sentences have been automatically split into words by IBM's word segmenter. The segmentation errors were removed by adding lexicons specific to the data. In addition, the dependencies are automatically resolved using the bunsetsu-level dependency parser (Kanayama et al., 2000) with the attachment rules for functional words defined in UD Japanese (Tanaka et al., 2016). Complex sentences with parenthesis were removed to avoid parsing errors. In the version 2.1 released in November 2017, manual annotations were merged with the semi-automatic annotations to reduce remaining errors.

UD Japanese-PUD was created in the same manner as UD Japanese-GSD, with the goal of maintaining consistency with UD Japanese-GSD. Since it is a parallel corpus with other languages, no sentences were removed from the corpus, including the ones containing parenthesis.

UD Japanese-Modern (Omura et al., 2017) is a small UD annotation data based on the 'Corpus of Historical Japanese: Meiji-Taisho Series I - Magazines' (CHJ) (Ogiso et al., 2017). The CHJ has morphological information compatible with the BCCWJ. We annotated *bunsetsu*-based syntactic dependency and coordinated structures using BCCWJ-DepPara annotation schema. We also an-

Table 1: Status of UD Japanese resources.

	UD Japanese-BCCWJ	UD Japanese-KTC	UD Japanese-GSD	UD Japanese-PUD	UD Japanese-Modern
Tokens (SUW)	1,098K	189K	186K	26K	14K
Copyright	masked surface	masked surface	CC BY-NC-SA	CC BY-SA	CC BY-NC-SA
UD version	v2	v1	v2	v2	v2
Media	Newspaper, Books, Magazines, Whitepaper, Yahoo! Answers, Yahoo! Blogs	Newspaper	Wikipedia	Parallel Corpus	Magazines in 19th century
Method	conversion from *bunsetsu* dependency	conversion from phrase structure tree	integration of bunsetsu-level manual annotations and word-level rule-based parser outputs	integration of bunsetsu-level manual annotations and word-level rule-based parser outputs	conversion from *bunsetsu* dependency
Plan	will be released in March 2018	the modified version (v2) will be released in September 2018	annotating by BCCWJ-DepPara schema from scratch	annotating by BCCWJ-DepPara schema from scratch	will be released in March 2018

notated predicate-argument relations based on the NAIST Text Corpus annotation schema. We utilised the conversion script used for UD Japanese-BCCWJ, because the two corpora share the same annotation schema. There are two characteristic syntactic structures in modern Japanese. One is inversion in Sino-Japanese literary styles. The other is predicative adnominals.

3. Issues

3.1. Overview

The first issue is word delimiters. Japanese has no overt word delimitation. We have to define what is the syntactic word in UD for such languages. The same issue can be found in the discrimination between phrases and clauses. Because obligatory cases in Japanese can be omitted, the definition of adjective and adverbial clauses is vague which requires some clarification for the UD syntactic relations.

The second issue is undefined policies in the UD guidelines. The design of POS can be split into two principles: one is lexicon-based morphology level; the other is the usage-based syntax level. The design of syntactic relations also has several levels. One level is that the relations are defined only by the surface form. Another level is that the relations are defined by the contextual words. The UD guidelines are not entirely clear as to define which levels are appropriate for the designs of POS and syntactic relations.

The third issue is the guidelines related to linguistic typology. The UD guidelines are biased toward subject prominent languages such as English. However, Japanese is one of topic prominent languages. For example, in the topic prominent languages, whereas the expletive label `expl` cannot appeared, a label for the topic marker is needed.

Though Japanese is a strictly head-final language, the UD guidelines define head-initial construction for coordinate structures. We hope to introduce the head directionality parameter for UD guidelines.

The last issue is the copyright of the original texts for Japanese language resources.

3.2. Word Delimitation

Word delimitation is a critical issue for building UD annotation, because Japanese is written without word boundaries. UD guidelines specify that the basic units of annotation are *syntactic words*.

We tend to define morphemic units which are smaller than the word unit in order to maintain unit uniformity. Therefore, when we define the morpheme unit as the Universal Dependency word unit, we must annotate the compound word construction, as defined in the morphological layer of Japanese linguistics. Although the smaller word unit can be produced with high precision, this is not suitable for Japanese syntactic dependency annotation. In Japanese NLP, *bunsetsu* (base phrase) tends to be used as a syntactic dependency annotation unit. The morphology level including multi-word expressions is encapsulated within a bunsetsu. Therefore we can concentrate on the annotation of purely syntactic phenomena.

We must define the *syntactic words* for UD annotation for Japanese. We used the BCCWJ morpheme annotation standard, which is based on UniDic word boundary definition. The definition contains three layers: SUW, LUW, and *bunsetsu*. SUW can be produced by the morphological analyser MeCab.[1] with UniDic[2] LUW and *bunsetsu* can be produced by the pre-trained chunker Comainu.[3]

NINJAL[4] defined five sorts of word unit definitions by operationalism. The most fine-grained unit is NINJAL Minimum Unit Word. SUW (Short Unit Word: 短単位) is constructively defined by the NINJAL Minimum Unit Word (最小単位). MUW (Middle Unit Word: 中単位) is a basic unit where a sound may change at the beginning or the ending of a word and/or an accent may change (Uchimoto and Den, 2008). The Middle Unit Word defines voiced compound ("rendaku") (van de Weijer et al., 2005).

[1] `taku910.github.io/mecab/`
[2] `unidic.ninjal.ac.jp/`
[3] `osdn.net/projects/comainu/`
[4] National Institute for Japanese Language and Linguistics.

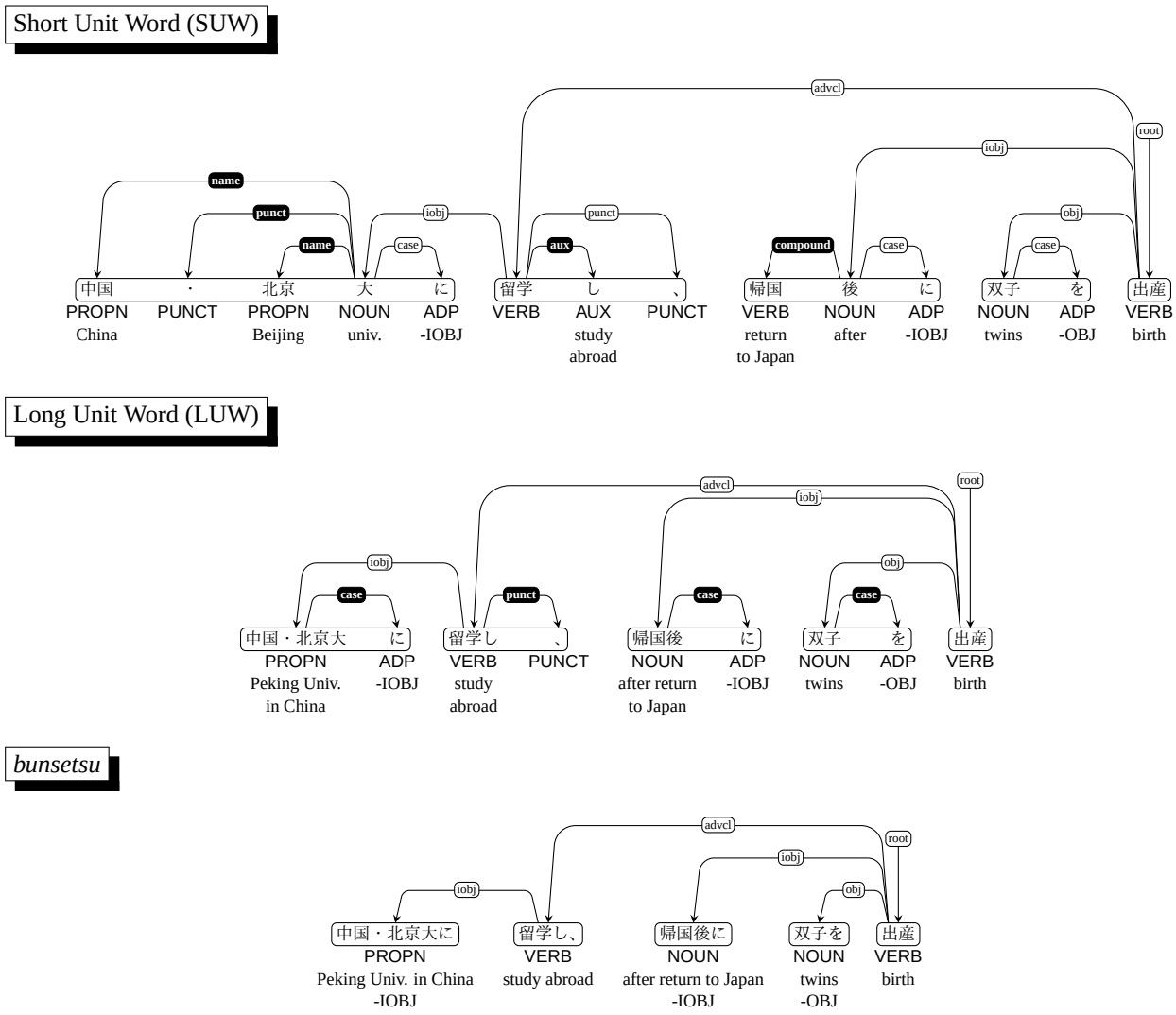


Figure 1: Word delimitation standards in Japanese.

The 'named entity' and 'compound word construction' related labels (in the black ground) which show up SUW disappear in the LUW (Long Unit Word: 長単位) layer. In the most coarse-grained level, *bunsetsu*, case marker and auxiliary verbs that show up in LUW disappear. Currently, all UD Japanese resources (in versions 1 and 2) are based on SUW word delimitations. Figure 1 shows the SUW, LUW and *bunsetsu* examples. The group of words in a squircle defines a *bunsetsu* chunk.

UD schema defines that the basic units of annotation are syntactic words, though a renowned typologist failed to identify words consistently across languages (Haspelmath, 2011). However, SUW is not suitable for syntactic dependency annotations. We need to define syntactic word delimitation for UD Japanese. Structuralists (Hattori, 1960; Miyaoka, 2015) discuss Japanese syntactic word definition. A blog[5] also discusses the plausible unit for UD Japanese resources. These discussions are not based on any language resources.

[5] www.cjvlang.com/Spicks/udjapanese.html

3.3. POS

In Japanese NLP, we have several POS tagsets used in IPADIC, JUMAN, and UniDic. IPADIC and its POS tagset have not been maintained for 10 years. The JUMAN POS tagset is based on the Masuoka-Takubo POS tagset (Masuoka and Takubo, 1992). UniDic has a two layered POS tagset for SUW and LUW delimitations. The two layered UniDic POS tagset is split in to two categories: lexicon-based (語彙主義) and usage-based (用法主義). The lexicon-based approach involves all possible categories for one word as labels. For example, the label '名詞-普通名詞-サ変形状詞可能' means that the word can be a noun, verbal noun or adjective depending on the context. The POS labels of UniDic SUW are lexicon-based. The POS labels are maintained in a large-scale POS-tagged lexicon and used in semi-Markov model-based morphological analysers. Usage-based labelling is determined by contextual information in a sentence. The POS labels of UniDic LUW are usage-based. The POS labels are produced by chunking from the UniDic SUW sequences with contextual features. The UD guidelines do not present POS design princi-

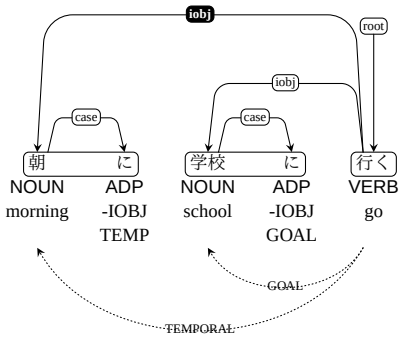


Figure 2: Double indirect object markers.

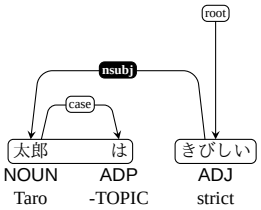


Figure 3: Topic marker.

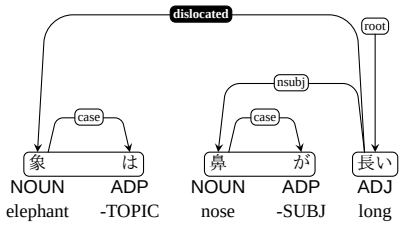


Figure 4: Double subject sentence.

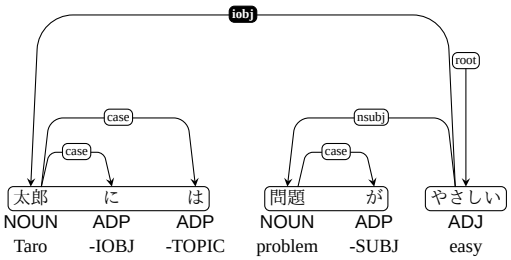


Figure 5: Indirect object marker with topic marker.

ples with this respect. We used usage-based POS tags from UniDic based on lexicon/corpora/morphological analysers to align the Universal POS tags.

3.4. Japanese case marker and syntactic relations

A phrase-based treebank includes syntactic relation information for UD. However, a *bunsetsu* dependency-based treebank does not include the syntactic relation information. Our policy is to produce UD syntactic relations from the surface forms of *bunsetsu* dependency-based dependency structures.

We assigned the label `nsubj` to nominal phrases with the case marker 'が' (ga), the label `obj` to nominal phrases with the case marker 'を' (o), and the label `iobj` to nominal phrases with the case marker 'に' (ni), respectively.

Though the indirect object marker 'に' (surface case) is given the label `iobj`, it can appear in more than one phrase with different deep cases for one predicate. The example in Figure 2 shows two indirect objects with the deep cases of TEMPORAL and GOAL presented below the sentences. If the UD guideline define deep case based labels for the syntactic relations, we need to assign the label `obl` to the TEMPORAL case. It is difficult to determine deep cases. Though the information to discriminate `iobj` from `obl` is available in BCCWJ-PAS or compatible annotation, it is not reproducible for other language resouces.

Japanese obligatory cases for predicates do not necessarily appear overtly in Japanese sentences. If 'が' does not overtly appear in the clause, the label `nsubj` is assigned to a nominal phrase with a topic marker 'は' (wa). Figure 3 shows an example of `nsubj` indicated by the topic marker. The topic marker 'は' and subject marker 'が' can appear for the same predicate. Figure 4 shows such a double subject sentence. The UD guidelines specify that the example needs the label `dislocated`.

The topic marker 'は' can immediately follow an indirect object marker 'に' as in Figure 5.

The labels for the topic marker are instable because of the design of UD syntactic labels. The design of UD syntactic labels are biased to subject-prominent languages. We argue for a new label for the topic marker for the topic-prominent languages such as Japanese, Korean, Chinese and Indonesian.

Table 2: The UD dependency labels between word/phrase and clause.

word/phrase	clause
`nsubj`	`csubj`
`obj`, `iobj`	`ccomp`, `xcomp`
`amod`	`acl`
`advmod`	`advcl`

3.5. Clause

The UD dependency labels are designed to split between word/phrase and clause (Table 2). The difference between clause and word/phrase is vague in Japanese, because obligatory cases, including subject, do not necessarily overtly appear in the sentences.

We define the dependency label `csubj` only for the clause with a particle (準体助詞) 'の' (no) with UD POS SCONJ in UD Japanese resources (Figure 6).

We define the dependency label `ccomp` only for the clause with a case particle 'と' (to) with UD POS ADP in UD Japanese resources (Figure 7).

`xcomp` is a label generally used for the open clausal complement of a verb or an adjective that is a predicative or clausal complement without its own subject. However, most subjects in Japanese clauses do not overtly appear. Therefore, we do not use `xcomp` for Japanese.

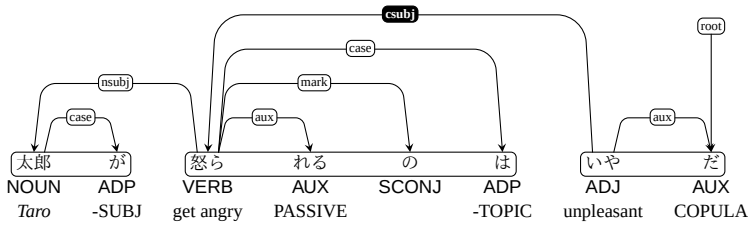


Figure 6: Clausal subject `csubj`.

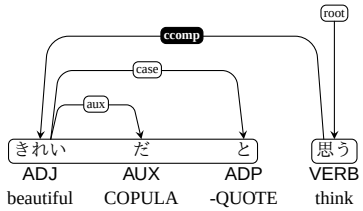


Figure 7: Clausal complement `ccomp`.

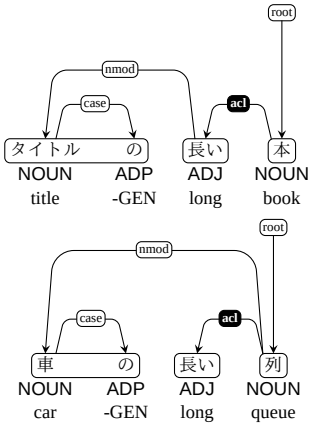


Figure 8: Adjective clause `acl`.

In the case of adjectives, attributive and predicative usages are not exclusive. There are two examples of attributive usages given in Figure 8. Whereas the latter example does not have an overt sense subject, the former has an overt sense subject and can also indicate predicative usage. The absence of an overt sense subject does not mean that the adjective is not used in the predicate. Therefore, we regard both adjectives as clauses, and we assign `acl`. The same goes for `advcl` in which there are adverbial forms (連用形) of adjectives.

Furthermore, `acl` has an ambiguity with `appos`. The UD guidelines (v2) define the label (`acl`) for an adjective clause with a predicate-argument relation (関係節内の関係) between the modifier and head (nominal phrase). The Japanese language has a similar clause – an appositional clause (関係節外の関係) without predicate-argument relations between the modifier and head. The difference between the adjective clause and appositional clause is not clearly defined in Japanese, since the case for predicates including adjectives tends to be omitted. Figure 9 shows some ambiguous examples. The arcs below the sentences indicate predicate-argument relations compatible with BCCWJ-PAS annotation. One interpretation (in the upper example) is that「中間報告」(interrim report) can be the subject of the 「設け」(set) as an adjective clause. The other interpretation (in the lower example) is that 「考え」(idea) cannot be the subject of the 「設け」(set) as an apposition clause. However, both interpretations are assigned `acl` in the UD annotation. Note that the label `appos` is assigned not for clause but for word/phrase.

3.6. Coordinate Structure

The Japanese language is a strictly head-final language with a *bunsetsu*-based dependency structure. However, UD guidelines define head-initial relations for coordinate structures, where in the attachment of conjuctions or punctuation is to the right-side constituent in the coordinate structures. Figure 10 shows simple coordinate structure examples. The grouping of the words indicates the *bunsetsu*. The word-based UD dependency relations intersect the *bunsetsu* boundaries.

Figure 11 shows a coordinate structure with three constituents. The root of the dependency tree is the leftmost constituent in the coordinate structure.

Figure 12 shows an example of non-constituent coordination. The upper example is the coordination of '花子はりんごを二つ買い' and '太郎はみかんを三つ買っ'. The root node is the rightmost word in the leftmost coordinate chunk. The lower example has an ellipsis of 'りんごを' before the numeral. The dependency relations of the omitted node are reduced and attached to the daughter node '三つ'.

Figure 13 shows another example of non-constituent coordination. The upper example is the coordination of '太郎は山に行き' and '花子は川に行っ'. The root node is the rightmost word in the leftmost coordinate chunk. The lower example is the ellipsis of the '行き' part, which is the root node in the upper example. The dependency relations of the omitted node that include the root are reduced and attached to the daughter node '太郎'. We assigned the label `orphan` between '太郎' (SUBJ) and '山' (IOBJ).

These peculiar tree constructions are caused by the head-initial definition of coordinate structures for a strictly head-final language.

3.7. Copyright

UD Japanese BCCWJ is based on BCCWJ. The annotation will be provided under an open license. However, users need to buy the original text in the form of the DVD edition

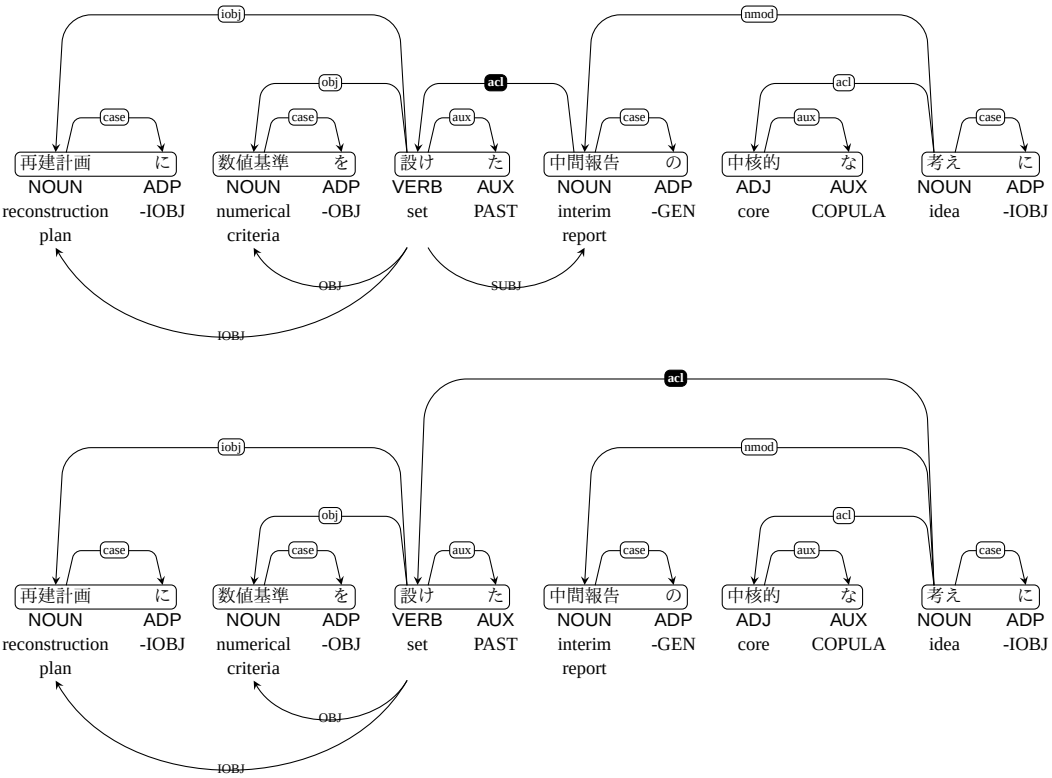


Figure 9: Ambiguous example of adjective clause (`acl`) (LUW delimitation).

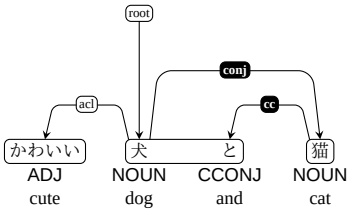


Figure 10: Simple coordinate structures.

of BCCWJ.[6]

UD Japanese-KTC is the first UD annotation schema data for Japanese. The data is based on Mainichi Newspaper in 1995. Though the annotation is available under the Creative Commons License (CC BY-SA), users are required to purchase the original text data CD-ROM.[7]

UD Japanese-GSD and UD Japanese-PUD data are based on open data. The original text is also under Creative Commons License (CC BY-NC-SA for UD Japanese-GSD and CC BY-SA for UD Japanese-PUD).

UD Japanese-Modern is based on CHJ. The copyright of the original text has already expired. We will provide the data under open license.

A newswire company asked us to provide a million scale newspaper texts for the UD Japanese project. We will annotate BCCWJ-like morphological information and *bunsetsu*-based syntactic structures for the data. The data will published on the Linguistic Data Consortium by the newswire company.

[6] `pj.ninjal.ac.jp/corpus_center/bccwj/en/dvd-index.html`

[7] `www.nichigai.co.jp/sales/mainichi/mainichi-series.html`

4. Conclusion

This paper presented our work to build UD compatible language resources for Japanese. We discussed the issues of using the UD annotation scheme on Japanese language. We are still working on conversion rules around word delimitation, clause definitions and coordinate structure.

Table 1 presents the plans to develop UD Japanese resources.

UD Japanese-KTC is based on the UD schema of version 1.0. The data is derived by conversion rules based on phrase structure trees. We are adapting the conversion rules to the version 2.0 schema. The newer version will be released on September 2018.

UD Japanese-GSD and **UD Japanese-PUD** are not based on UniDic word boundaries (SUW/LUW). We are annotating the word boundaries of SUW and LUW with morphological information in UniDic definition. The next step is to annotate *bunsetsu*-based dependency structures based on BCCWJ-DepPara schema.

UD Japanese-BCCWJ and **UD Japanese-Modern** are developped based on version 2.0 schema. These two data will be released in March 2018.

5. Acknowledgements

This work was supported by JSPS KAKENHI Grants Numbers JP15K12888 and JP17H00917 and is a project of the

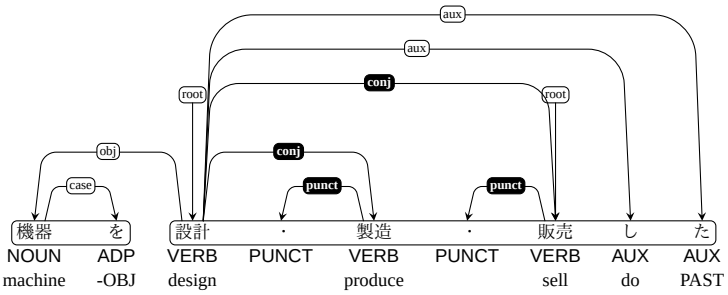


Figure 11: Coordinate structure with three constituents.

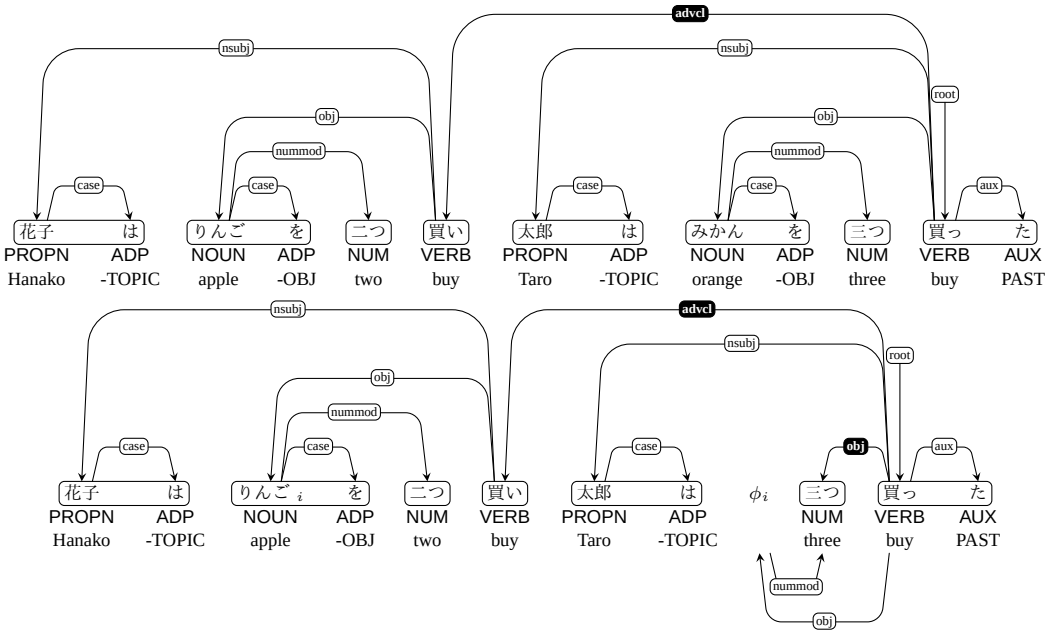


Figure 12: Nominal phrase ellipsis in the non-constituent conjunct coordination.

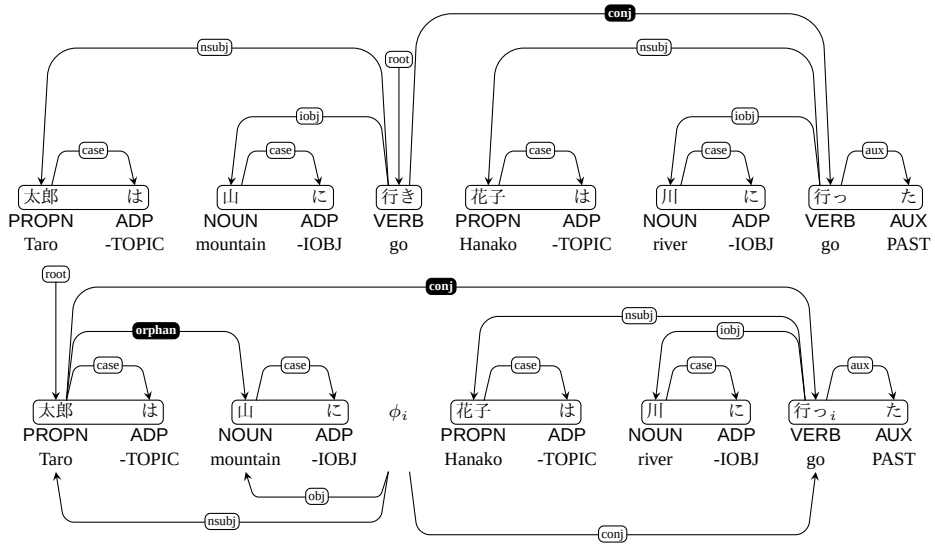


Figure 13: Predicate ellipsis in the non-constituent conjunct coordination.

Center for Corpus Development, NINJAL.

6. Bibliographical References

Asahara, M. and Matsumoto, Y. (2016). BCCWJ-DepPara: A Syntactic Annotation Treebank on the 'Balanced Cor-

pus of Contemporary Written Japanese'. In *Proceedings of the 12th Workshop on Asian Langauge Resources (ALR12)*, pages 49–58.

Haspelmath, M. (2011). The indeterminacy of word segmentation and the nature of morphology and syntax. *Folia Linguistica*, 45(1):31–80.

Hattori, S., (1960). *Gengogaku no hoho*, chapter Fuzokugo to Fuzokukeishiki, pages 461–491. Iwanami Shoten. (in Japanese).

Iida, R., Komachi, M., Inui, K., and Matsumoto, Y. (2007). Annotating a Japanese Text Corpus with Predicate-argument and Coreference Relations. In *LAW '07: Proceedings of the Linguistic Annotation Workshop*.

Kanayama, H., Torisawa, K., Mitsuishi, Y., and Tsujii, J. (2000). A hybrid Japanese parser with hand-crafted grammar and statistics. In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 411–417.

Kurohashi, S. and Nagao, M., (2003). *Building a Japanese Parsed Corpus – while Improving the Parsing System*, chapter 14, pages 249–260. Kluwer Academic Publishers.

Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M., and Den, Y. (2014). Balanced Corpus of Contemporary Written Japanese. *Language Resources and Evaluation*, 48:345–371.

Masuoka, T. and Takubo, Y. (1992). *Kiso Nihongo Bunpou – Kaitei Ban*. Kuroshio Publisher.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., K.Ganchev, Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Castelló, N. B., and Lee, J. (2013). Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 92–97.

Miyaoka, O. (2015). *Go to wa Nani ka Saiko (Reconsidering What is the "Word"?)*. Sanseido. (in Japanese).

Ogiso, T., Kondo, A., Mabuchi, Y., and Hattori, N. (2017). Construction of the "Corpus of Historical Japanese: Meiji-Taisho Series I - Magazines". In *Proceedings of Digital Humanities 2017*.

Omura, M., Takahashi, Y., and Asahara, M. (2017). Universal Dependency for Modern Japanese. In *Proceedings of the 7th Conference of Japanese Association for Digital Humanities (JADH2017)*.

Tanaka, T. and Nagata, M. (2013). Constructing a Practical Constituent Parser from a Japanese Treebank with Function Labels. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically Rich Language*, pages 108–118.

Tanaka, T., Miyao, Y., Asahara, M., Uematsu, S., Kanayama, H., Mori, S., and Matsumoto, Y. (2016). Universal Dependencies for Japanese. In *Proceedings of LREC 2016*.

Uchimoto, K. and Den, Y. (2008). Word-level dependency-structure annotation to corpus of spontaneous japanese and its application. In *Proceedings of LREC 2008*.

Ueda, Y., Iida, R., Asahara, M., Matsumoto, Y., and Tokunaga, T. (2015). A Predicate-Argument and Coreference Relation Annotation on the 'Balanced Corpus of Contemporary Written Japanese', *gendai nihongo kakikotoba kinkou koopasu ni taisuru jutugo koukouzou kyosanshou kankei annoteeshon* (in Japanese). In *8th Corpus Japanese Linguistics Workshop*, pages 205–214.

van de Weijer, J., Nanjo, K., and Nishihara, T. (2005). *Voicing in Japanese*. Studies in Generative Grammar. Walter De Gruyter Inc.

Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinková, S., Hajič jr., J., Hlaváčová, J., Kettnerová, V., Urešová, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., de Paiva, V., Droganova, K., Martínez Alonso, H., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Fernandez Alcalde, H., Strnadova, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonça, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics.

Developing the Bangla RST Discourse Treebank

Debopam Das and Manfred Stede
University of Potsdam
Karl-Liebknecht Strasse 24-25,
14476 Potsdam, Germany
{debdas, stede}@uni-potsdam.de

Abstract

We present a corpus development project which builds a corpus in Bangla called the Bangla RST Discourse Treebank. The corpus contains a collection of 266 Bangla text, which are annotated for coherence relations (relations between propositions, such as *Cause* or *Evidence*). The texts represent the newspaper genre, which is further divided into eight sub-genres, such as business-related news, editorial columns and sport reports. We use Rhetorical Structure Theory (Mann and Thompson, 1988) as the theoretical framework of the corpus. In particular, we develop our annotation guidelines based on the guidelines used in the Potsdam Commentary Corpus (Stede, 2016). In the initial phase of the corpus development process, we have annotated 16 texts, and also conducted an inter-annotator agreement study, evaluating the reliability of our guidelines and the reproducibility of our annotation. The corpus upon its completion could be used as a valuable resource for conducting (cross-linguistic) discourse studies for Bangla, and also for developing various NLP applications, such as text summarization, machine translation or sentiment analysis.

Keywords: corpus, discourse annotation, coherence relations, Rhetorical Structure Theory, Bangla

1. Introduction

Coherence in discourse is, to a large extent, achieved through the use of coherence relations. Coherence relations (also known as discourse relations or rhetorical relations) refer to the semantic or pragmatic relations between text segments representing propositions, such as *Contrast* or *Elaboration*. For example, consider the following text fragment[1]:

(1) [The U.S. Coast Guard closed six miles of the Houston Ship Channel, where about 150 companies have operations,] [because the thick, black smoke obscured the area.] [wsj-1309].

In Example 1, there are two text segments (marked by square brackets) which are connected to each other by a *Reason* relation in which the second segment serves as a reason for the first one.

We are interested in creating a discourse-annotated corpus in which the texts are annotated with respect to coherence relations. Corpora with relational annotation have become increasingly useful in discourse studies, and they have been developed in various languages. These corpora have also served as valuable resources for developing various computational applications, such as discourse parsing, text summarization, or argumentation mining, to name a few.

We have recently begun to develop a corpus of Bangla texts annotated for coherence relations. We use Rhetorical Structure Theory or RST (Mann and Thompson, 1988) as the theoretical framework of the corpus, and we call the corpus the Bangla RST Discourse Treebank or Bangla RST-DT. In this paper, we present the corpus development project, describing our annotation schemes and annotation procedure. In addition, we also report on an inter-annotator agreement study, by having the initial subset of the corpus annotated by a team of annotators and evaluated thereon.

The paper is organized as follows: In Section 2., we provide a brief introduction to Rhetorical Structure Theory. Section 3. gives an account of previous works on RST-based corpora in different languages. In Section 4., we describe the annotation guidelines used to build the Bangla RST-DT. In Section 5., we provide the characteristics of the corpus, describing the training of the annotators (in Section 5.2.) and annotation procedure (in Section 5.3.). Reliability of annotation including an inter-annotator agreement study is discussed in Section 6.. Finally, Section 7. summarizes the paper, and highlights a few potential future applications of the corpus.

2. Rhetorical Structure Theory

The concept of coherence relations has been extensively studied in different discourse frameworks such as Rhetorical Structure Theory or RST (Mann and Thompson, 1988), Segmented Discourse Representation Theory or SDRT (Asher and Lascarides, 2003), the Penn Discourse Treebank or PDTB framework (Prasad et al., 2008), the Cognitive approach to Coherence Relations or CCR (Sanders et al., 1992), the Unified Linguistic Discourse Model or ULDM (Polanyi et al., 2004), or Hobbs' theory (Hobbs, 1985), further expanded by Kehler (Kehler, 2002). For our purpose, we use RST, because we believe that RST provides a healthy mix of an explanatory account of certain aspects of text organization and has proven practical applicability to a wide range of text types. In addition, RST is a language-independent theory, and it has been successfully used in a number of areas in computational discourse processing, such as text generation, discourse

[1]Example source: RST Discourse Treebank (Carlson et al., 2002). The content inside the square brackets following the example refers to the file number in the corpus from which the text fragment has been taken.

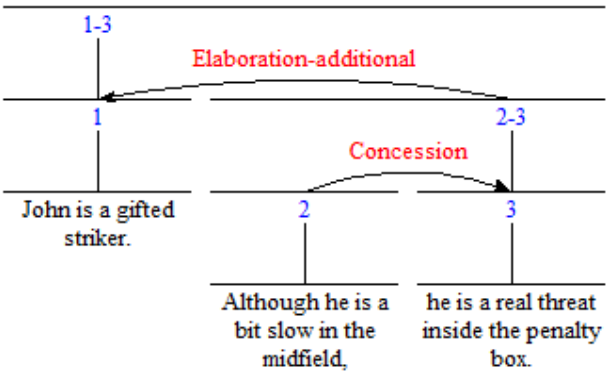


Figure 1: Graphical representation of an RST analysis

parsing, and text summarization (see Taboada and Mann (2006a) for an overview).

Rhetorical Structure Theory or RST is a functional theory of text organization (Mann and Thompson, 1988; Taboada and Mann, 2006b). It describes what parts a text is made of, what kinds of relationships exist between these parts, and how these parts are organized with respect to each other to constitute a coherent piece of discourse. In RST, relations hold between two (or sometimes more) non-overlapping text spans, and can be multinuclear, reflecting a paratactic relationship, or nucleus-satellite, a hypotactic type of relation. The names nucleus and satellite refer to the relative importance of each of the relation components. The relation inventory suggested by Mann and Thompson consisted of 25 relations, but the authors emphasized that additions may be possible for specific kinds of text. Relations that are used in many projects include *Cause, Concession, Condition, Elaboration, Result* or *Summary*.

Texts, according to RST, are built out of basic clausal units (also called elementary discourse units or EDUs) that enter into rhetorical (or discourse, or coherence) relations with each other in a recursive manner. Mann and Thompson (1988) proposed that most texts can be analyzed in their entirety as recursive applications of different types of relations. In effect, this means that an entire text can be analyzed as a tree structure, with clausal units being the leavees and relations the nodes.

For illustration purposes, we provide the annotation of a short (invented) text, represented by the tree diagram[2] in Figure 1. The text is segmented for three EDUs (minimal spans), which are marked by the cardinal numbers 1, 2 and 3, respectively. In the diagram, the arrow points to a span called the nucleus, and away from another span called the satellite. Span 2 (satellite) is connected to Span 3 (nucleus) by a *Concession* relation, and together they make the combined Span 2-3, which is further linked as a satellite to Span 1 (nucleus) by an *Elaboration-additional* relation.

[2]The RST diagram is created by RSTTool (O'Donnell, 2000) which provides a graphical representation of the RST analysis of a text in the form of tree diagrams.

3. Related work

The tradition of building discourse-annotated corpora began with the introduction of the RST Discourse Treebank or RST-DT (Carlson et al., 2002). The RST-DT contains a collection of 385 Wall Street Journal articles annotated for coherence relations. The corpus provides annotation for more than 20,000 relation instances, and the corpus has been extensively used for developing a number of RST-based discourse parsers, including Hernault et al. (2010), Ji and Eisenstein (2013), Feng and Hirst (2014) and Braud et al. (2016).

Considering the success of the RST-DT, similar corpora have been developed, in English and also in other languages. Taboada and Renkema develop the Discourse Relations Reference Corpus in English (Taboada and Renkema, 2008), annotating a set of 65 texts taken from the RST website[3], RST-DT (Carlson et al., 2002) and SFU Review Corpus (Taboada, 2008). In Dutch, an RST corpus is developed with annotation of discourse structure and also lexical cohesion (van der Vliet et al., 2011). For Brazilian Portuguese, RST is used to create the CSTNews corpus (Cardoso et al., 2011), which includes annotation of news texts and single/multi-document summaries[4]. In Spanish, da Cunha and colleagues (da Cunha et al., 2011a; da Cunha et al., 2011b) develop the RST Spanish Discourse Treebank, which includes a collection of over 250 RST-annotated texts from different specialized domains (Astrophysics, Law, Mathematics, Psychology, etc.). The Basque version of the RST corpus called the RST Basque Treebank (Iruskieta et al., 2013) is annotated not only for coherence relations, but for their signals as well. For German, the Potsdam Commentary Corpus or PCC (Stede, 2016) is built over a collection of 170 newspaper commentaries. The texts in PCC are annotated for RST relations, and also for five other different layers of annotation, such as syntax, co-reference and information structure. Initiatives to develop RST corpora have also been taken for Chinese (Cao et al., 2017) and Russian (Toldova et al., 2017), and those corpora are currently under production.

We chose to develop an RST corpus for Bangla. Bangla is an Indo-Aryan language spoken in India and Bangladesh, with an estimated 177 million speakers in the Indian subcontinent (leaving aside the diasporic Bangla speakers living elsewhere) (Dasgupta, 2003). While Bangla has remained a relatively well-studied language, unfortunately there are only a handful of linguistic corpora available for Bangla, mainly either transcribed for speech (Das et al., 2011; Bills et al., 2016), or annotated for lemmatization, POS tags or similar phenomena (Bali et al., 2010; Chaudhury et al., 2017; Ekbal and Bandyopadhyay, 2008), or unannotated (Al Mumin et al., 2014). To our knowledge, there is no discourse-annotated text corpus available

[3]http://www.sfu.ca/rst/

[4]CSTNews is also annotated based on Cross-document Structure Theory (Radev, 2000), which explains how text passages from different topics on the same topic are related to each other.

in Bangla, and in this respect, the present Bangla RST-DT is going to be the first data set of its kind.

4. Annotation Guidelines

The success of a corpus annotation task depends much on the reliability of the guidelines to be followed in the annotation. Our annotation guidelines for the Bangla RST-DT[5] are based on the guidelines used to annotate the Potsdam Commentary Corpus or PCC (Stede, 2016)[6], and are more closely related to an updated version of the PCC guidelines used in Das et al. (2017)[7]. In the present project, we adopt a modified version of these guidelines for annotating texts in Bangla, because these original guidelines (although used for German and English texts) are based on RST, which is essentially a language-independent theory.

An RST annotation of a text comprises three steps: (1) segmenting the text into EDUs (elementary discourse units), (2) assigning the relations between EDUs and larger spans, and finally (3) building the hierarchical RST tree, comprising all the connected spans stemming from a single root node at the top of the tree.

Our segmentation guidelines closely follow those used for German texts in the PCC (Stede, 2016) and for English texts in SLSeg (syntactic and lexically based discourse segmenter) (Tofiloski et al., 2009). These guidelines were also used in Das et al. (2017) for segmenting German and English texts, respectively. Both PCC and SLSeg guidelines closely adhere to the original definition of spans in RST (Mann and Thompson, 1988), which specifies that only adjunct clauses (rather than complement clauses) are considered to constitute legitimate EDUs. According to this principle, every EDU must contain a verb, either finite or non-finite. Broadly, we consider coordinated clauses (but not coordinated verb phrases), adjunct clauses and non-restrictive relative clauses to establish legitimate EDUs. In all cases, this strategy is, however, complemented by the annotator's decision on whether a discourse relation could hold between the resulting segments.

Since Bangla employs a different morphological and syntactic structure than German and English, it is important to determine how clause-based discourse segmentation strategies could be used for Bangla. For this purpose, we consult some notable works on Bangla grammar, such as Chatterji (1988), Chakraborty (1992), Chaki (1996) and Sarkar (2006). In addition, we closely examine the clausal structures in the Bangla texts from our corpus. Based on our understanding and observation of the Bangla syntactic structures, we ultimately decide to retain the basic segmentation principles from the PCC and SLSeg guidelines, but at the same time, we also modify some of those guidelines (or even develop some new segmentation strategies) to account for certain constructions in Bangla.

We enumerate the most significant segmentation principles followed in our annotation below. For more information about the segmentation guidelines, see (Das, under review).

1. Clausal subjects, represented as verbal nouns or complete clauses (with a finite verb) in Bangla, are not considered to be EDUs.
2. Clausal complements (including clausal objects of verbs, expressed as verbal nouns or infinitival clauses) are not considered to constitute EDUs.
3. Attribution clauses are represented either by reported speech, both directly (by direct quotes) or indirectly. They can also be represented through cognitive predicates (containing verbs expressing feelings, thoughts or opinions, such as *think*, *know*, *estimate* or *wonder* in English). Attribution clauses are not considered to form EDUs.
4. Non-restrictive relative clauses which encode a coherence relation with their host clauses are considered to be EDUs. However, restrictive relative clauses which typically elaborate on an entity in their host clauses are not considered as EDUs.
5. Participial clauses, conditional clauses, infinitival clauses (if they are not complement clauses) and verbal nouns with a postposition are treated as EDUs.

In an RST relational annotation task, the next step after segmentation is to determine the suitable coherence relations that hold between EDUs (or larger spans comprising multiple EDUs). This is done by selecting a relation type from a relational taxonomy that specifies a range of all relation types (along with their definitions) which could possibly occur in a corpus. Whenever a new relational instance in the corpus is encountered and interpreted, it is assigned a relation type which best represents the relational instance in the corpus.

The relational taxonomy used in our annotation is the one used in the PCC (Stede, 2016), which is based on the relation set proposed in the original RST paper (Mann and Thompson, 1988). This means that the relation set is much smaller than that of the RST-DT (Carlson et al., 2002) which employs a large set of 78 relations (divided into 16 broad relation classes). This is because our taxonomy does not use the many nucleus-satellite variants, and it deliberately left out suggestions like *Topic-Comment* or *Attribution*, which are not considered as coherence relations in the same way as those of "classic" RST[8].

Our taxonomy includes 31 relations which are organized in a slight different way from Mann and Thompson (1988). It

[5]The Bangla RST-DT annotation guidelines are available at: http://angcl.ling.uni-potsdam.de/pdfs/Bangla-RST-DT-Annotation-Guidelines.pdf

[6]http://angcl.ling.uni-potsdam.de/resources/pcc.html

[7]http://www.sfu.ca/~mtaboada/docs/research/RST_Annotation_Guidelines.pdf

[8]We do not claim that phenomena of Topic-Comment and Attribution do not exist. Instead, notions of information structure in our view belong to a separate level of analysis, and not to that of coherence relations.

retains the original binary classification of subject-matter and presentational relations (semantic and pragmatic relations, respectively, in our taxonomy). We also have an extra category for *textual* relations (e.g., *List, Summary*). The taxonomy for mononuclear relations is provided in Table 1.

Semantic	Pragmatic	Textual
Circumstance	Background	Preparation
Condition	Antithesis	Restatement
Otherwise	Concession	Summary
Unless	Evidence	
Elaboration	Reason	
E-elaboration	Reason-N	
Interpretation	Justify	
Means	Evaluation-S	
Cause	Evaluation-N	
Result	Motivation	
Purpose	Enablement	
Solutionhood		

Table 1: Mononuclear relations in Bangla RST-DT

In addition, there are five multinuclear relations in our taxonomy: *Sequence, Contrast, Conjunction, List* and *Joint*. Among these, *Sequence* is a semantic relation, while the remaining four relations can function as semantic, pragmatic or textual, depending on context.

5. Corpus Development Process

5.1. Characteristics of the Corpus

The Bangla RST-DT is being developed as a corpus of Bangla annotated for coherence relations following RST. The corpus contains 266 texts, comprising 71,009 words, with an average of 267 words per text. The corpus represents newspaper genre. The texts have been collected from a popular Bangla daily called *Anandabazar Patrika* published in India. The texts in the corpus come from eight different sub-genres: (1) business-related news, (2) editorial columns, (3) international affairs, (4) cityscape (stories on *Kolkata*, the home city of the newspaper), (5) letters to the editor, (6) articles on nature, (7) features on science, and (8) reports on sports. The distribution of the texts for these sub-genres in the corpus is provided in Table 2.

Sub-genre	Number of texts
Business	31
Editorial column	32
Internatinal affairs	31
Cityscape	32
Letters to the editor	41
Nature	31
Science	34
Sports	34
TOTAL	266

Table 2: Distribution of texts in Bangla RST-DT

5.2. Annotator Profile and Training

The initial subset of the corpus (also used for the inter-annotator agreement study) was annotated by a team of three annotators who are native Bangla speakers. The team includes two graduate students and one of the authors of the present paper. The subset comprises 16 texts which were separately annotated by each annotator.

The two graduate students hired as the annotators have prior experience in other types of text annotation. They were extensively trained in RST by the third annotator (who has many years of experience with various RST annotation projects, and served as the *expert* annotator in this project). The training roughly consists of three phases. In the first phase, the student annotators were introduced to RST, and they learned to operate RSTTool (O'Donnell, 2000)[9] which was used for annotation. In this phase, they also did a fair amount of practice by independently annotating many texts (in English) from different genres (newspaper reports, scientific articles, undergraduate textbooks, etc.). In the second phase, the annotators were introduced to the annotation guidelines of the present project, and following those guidelines, they annotated a number of Bangla texts (which were collected from *Anandabazar Patrika*, and are similar to the target texts) as part of their practice. In the final phase, all three annotators annotated three Bangla texts separately, and compared the annotations with each other. The results were jointly discussed and adjudicated in order to resolve disagreements in annotation, arising from issues such as assigning nuclearity or choosing a relation label. The overall procedure stretched over two months, and each student annotator spent approximately 35 hours on the training phase.

5.3. Annotation Procedure

For annotating the target 16 texts in the corpus, we used pre-segmented texts. The texts were segmented beforehand by the expert annotator, following the segmentation guidelines described in the previous section. This is because we believe that segmentation is essentially a different kind of task from other tasks in relational annotation (such as deciding on mono- vs. multinuclear relations, or choosing a certain relation label). Segmentation can be factored out and evaluated separately, which has the advantage that it reduces the effort of the annotators, and makes it much easier to quantitatively evaluate the nuclearity-assignment and relation-tagging decisions (and also to perform a qualitative analysis of disagreement).

The annotation procedure consists of the following steps:

1. Identifying the macro structure: We read a text first, and identify the main topic(s), and henceforth, the most important nucleus (or nuclei). This helps us divide the text into larger units.

[9] We use the source version of RSTTool which works on Unicode scripts (with UTF-8 encoding for the Bangla script).

2. Identifying the nucleus-EDUs: We select the EDUs that play an important role in the text. If one EDU can be singled out as representing the central statement of the text, we mark it as such, and also the other important EDUs.

3. Connecting EDUs with relations: We consider each EDU and its direct neighbours, in order to see if there is a clearly recognizable relation between such a pair. This will often be the case with syntactically dependent pairs, and sometimes when two independent EDUs are linked with a discourse marker. If we find such an EDU-pair, we link the EDUs with an appropriate relation, based on the nuclearity decision made in the previous step.

4. Connecting larger spans: When the EDUs are connected, they make larger spans. We link these larger spans with appropriate relations. Sometimes, the relation between larger units (and also between smaller units/EDUs) are signalled by a discourse marker, but in many cases, they can also be indicated by other textual signals (such as syntactic or lexical features) as used in the RST Signalling Corpus (Das et al., 2015). We continue to connect even larger spans until we notice all the spans are connected and we have a single root node in the tree.

In practice, as mentioned earlier, we use RSTTool to do our annotation. The texts (in .txt format) were first imported to RSTTool, and segmented by the expert annotator beforehand. The segmented texts were then distributed among all the three annotators who further did the nuclearity and relational annotation. The annotated texts are saved in .rs3 (an XML) format.

We observed that the annotators took approximately between 30 minutes and one hour to annotate a single text. The time varies mainly according to the length of the text, and also its type. For example, argumentative texts (editorial columns, letters to the editor, etc.) usually take longer time to annotate than simple news reports (business related news, sports reports, etc.).

6. Inter-annotator Agreement

In order to check the validity of our annotation guidelines and test the reproducibility of our annotation, we conducted an inter-annotator agreement study. We followed the method proposed in Marcu (2000) which evaluates agreement between competing analyses with respect to four individual dimensions: unit, span, nuclearity and relation. Since in our study we use pre-segmented texts, we calculated agreement only for span, nuclearity and relation.

We computed the agreement between a pair of annotators in terms of precision and recall. First, we calculated the agreement between the expert annotation and student annotation, considering the former as the "gold annotation". For this purpose, we made use of RSTEval, a tool that provides precision and recall statistics between a "gold" human annotation and a parser-produced annotation[10]. The results of the pairwise comparisons are provided in Table 3.

	Exp vs. Student1		**Exp vs. Student2**	
Dimension	**Precision**	**Recall**	**Precision**	**Recall**
Span	0.87	0.87	0.86	0.86
Nuclearity	0.69	0.69	0.68	0.68
Relation	0.51	0.51	0.49	0.49

Table 3: Precision and recall for expert versus student annotators

Table 3 show that the agreement between annotators is high for span, fair for nuclearity and moderate for relation This is in line with earlier studies on relational annotation (Carlson et al., 2003; Cardoso et al., 2011; da Cunha et al., 2011a), which suggest that spans are easier to identify than nuclearity status, while relation assignment is particularly difficult.

We would like to point out that the precision and recall values in a pairwise comparison are same. The identical values stem from the use of pre-segmented text files by all annotators. The same set of EDUs for a text generated the identical number of spans and also the identical number of relations across annotations. This further resulted in producing an equal number of relevant items (in the gold annotation) and retrieved items (in the annotation to be tested), which were used as the denominators for the precision and recall formula, respectively.

Second, we computed the agreement between two student annotators, again in terms of precision-recall values. Here, the annotation produced by student annotator 1 was (technically) used as the "gold annotation". The results are provided in Table 4.

Dimension	**Precision**	**Recall**
Span	0.90	0.90
Nuclearity	0.74	0.74
Relation	0.59	0.59

Table 4: Precision and recall for student annotator 1 versus student annotator 2

A comparison of all pairwise results (in Table 3 and 4) shows that the agreement is higher when the annotations produced by the student annotators were examined. This suggests that the annotators were successfully trained and were able to adhere to the annotation guidelines, which in turn yielded higher agreement between them. We thus feel that our annotations are reliable, and believe that we can use the guidelines and infrastructure to train further annotators, in order to complete the corpus annotation.

[10] http://www.nilc.icmc.usp.br/rsteval/

7. Conclusion

We have presented the development of the Bangla RST Discourse Treebank, and described our annotation guidelines and annotation procedure. The corpus started with the annotation of 16 texts, which were evaluated for agreement among the annotators. The currently-ongoing work includes annotation of the remaining 250 more texts, representative of different sub-genres in the newspaper genre.

The Bangla RST-DT when completed will be made publicly available. The corpus will have two clear applications. First, from a theoretical point of view, it will provide empirical support to the existing research on the discourse structure of Bangla, an Indo-Aryan language, and provide a valuable resource for conducting cross-linguistic discourse studies. Second, the corpus will be used to develop discourse parsing systems for Bangla texts, which may further be used for NLP applications such as automatic summarization, machine translation, sentiment analysis, or argumentation mining.

Furthermore, since our annotation guidelines have now been successfully applied to German, English, and Bangla, with only minor changes, we believe that they are now quite stable across languages and can serve for projects by other researchers, possibly involving further languages.

8. Acknowledgements

We would like to thank our two student annotators, Lahari Chatterjee and Soumya Sankar Ghosh, for their valuable contribution to the development the Bangla RST Discourse Treebank.

9. Bibliographical References

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge.

Braud, C., Plank, B., and Søgaard, A. (2016). Multi-view and multi-task training of RST discourse parsers. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1903–1913, Osaka, Japan.

Chaki, J. (1996). *Bangla Bhashar Byakaran*. Ananda Publishers, Kolkata, India.

Chakraborty, U. K. (1992). *Bangla Padaguccher Sangathan (Structure of Bengali Phrases)*. Dey's Publishing, Kolkata, India.

Chatterji, S. K. (1988). *Bhasha-Prakash Bangla Vyakaran*. Rupa and Company, New Delhi, India.

Das, D., Taboada, M., and Stede, M. (2017). The good, the bad, and the disagreement: Complex ground truth in rhetorical structure analysis. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms*, pages 11–19.

Dasgupta, P. (2003). Bangla. In George Cardona et al., editors, *The Indo-Aryan Languages*, pages 386–427. Routledge.

Das, D. (under review). Discourse Segmentation in Bangla. In *4th Workshop on Indian Language Data: Resources and Evaluation (WILDRE-4)*.

Feng, V. W. and Hirst, G. (2014). A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 511–521, Baltimore, MA.

Hernault, H., Prendinger, H., duVerle, D., and Ishizuka, M. (2010). Hilda: A discourse parser using support vector machine classification. *Dialogue and Discourse*, 1(3):1–33.

Hobbs, J. (1985). On the coherence and structure of discourse. Report, CSLI.

Ji, Y. and Eisenstein, J. (2013). Discriminative improvements to distributional sentence similarity. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 891–896, Seattle, Washington, USA, October. Association for Computational Linguistics.

Kehler, A. (2002). *Coherence, Reference, and the Theory of Grammar*. Center for the Study of Language and Information, Stanford, CA.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Marcu, D. (2000). *The Theory and Practice of Discourse Parsing and Summarization*. MIT Press, Cambridge, MA.

O'Donnell, M. (2000). RSTTool 2.4 – A markup tool for Rhetorical Structure Theory. In *Proceedings of the International Natural Language Generation Conference*, pages 253–256, Mizpe Ramon/Israel.

Polanyi, L., Culy, C., van den Berg, M., Thione, G. L., and Ahn, D. (2004). A rule based approach to discourse parsing. In *the 5th SIGdial Workshop on Discourse and Dialogue, ACL*.

Radev, D. R. (2000). A Common Theory of Information Fusion from Multiple Text Sources Step One: Cross-document Structure. In *Proceedings of the 1st SIGdial Workshop on Discourse and Dialogue - Volume 10*, SIGDIAL '00, pages 74–83, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sanders, T., Spooren, W., and Noordman, L. (1992). Toward a taxonomy of coherence relations. *Discourse Processes*, 15:1–35.

Sarkar, P. (2006). *Bangla Byakaran Prasanga*. Dey's Publishing, Kolkata, India.

Taboada, M. and Mann, W. C. (2006a). Applications of Rhetorical Structure Theory. *Discourse Studies*, 8(4):567–588.

Taboada, M. and Mann, W. C. (2006b). Rhetorical structure theory: Looking back and moving ahead. *Discourse Studies*, 8(3):423–459.

Tofiloski, M., Julian, B., and Taboada, M. (2009). A Syntactic and Lexical-Based Discourse Segmenter. In *47th Annual Meeting of the Association for Computational Linguistics*, pages 77–80.

10. Language Resource References

Al Mumin, M. A., Shoeb, A. A. M., Selim3, M. R., and Iqbal, M. Z. (2014). Sumono: A representative modern

bengali corpus. *SUST Journal of Science and Technology*, 21(1):78–86.

Bali, K., Choudhury, M., and Biswas, P. (2010). Indian language part-of-speech tagset: Bengali ldc2010t16.

Bills, A., David, A., Dubinski, E., Fiscus, J., Gillies, B., Harper, M., Jarrett, A., Molina, M., Ray, J., Rytting, A., Paget, S., Shen, W., Silber, R., Tzoukermann, E., and Wong, J. (2016). Iarpa babel bengali language pack iarpa-babel103b-v0.4b.

Cao, S., Xue, N., da Cunha, I., Iruskieta, M., and Wang, C. (2017). Discourse Segmentation for Building a RST Chinese Treebank. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms*, pages 73–81.

Cardoso, P., Maziero, E., Jorge, M. L. C., Seno, E., Di Felippo, A., Rino, L., Nunes, M. d. G., and Pardo, T. (2011). CSTNews - A Discourse-Annotated Corpus for Single and Multi-Document Summarization of News Texts in Brazilian Portuguese. In *Proceedings of the 3rd RST Brazilian Meeting*, pages 88–105.

Carlson, L., Marcu, D., and Okurowski, M. E. (2002). RST Discourse Treebank, ldc2002t07.

Carlson, L., Marcu, D., and Okurowski, M. E. (2003). Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In Jan van Kuppevelt et al., editors, *Current Directions in Discourse and Dialogue*, pages 85–112. Kluwer, Dordrecht.

Chaudhury, T. H., Matin, A., Hossain, M., Uzzaman, A., and Masum, M. (2017). Annotated bangla news corpus and lexicon development with pos tagging and stemming. *Global Journal of Researches in Engineering*, 17(1).

da Cunha, I., Torres-Moreno, J.-M., and Sierra, G. (2011a). On the Development of the RST Spanish Treebank. In *Proceedings of the 5th Linguistic Annotation Workshop*, LAW V '11, pages 1–10, Stroudsburg, PA, USA. Association for Computational Linguistics.

da Cunha, I., Torres-Moreno, J.-M., Sierra, G., Cabrera-Diego, L. A., Rolón, B. G. C., and Bartilotti, J. M. R. (2011b). The Rst Spanish Treebank On-line Interface. In *Proceedings of the Recent Advances in Natural Language Processing*, pages 698–703.

Das, B., Mandal, S., and Mitra, P. (2011). Bengali speech corpus for continuous automatic speech recognition system. In *Proceedings of Conf. Speech Database and Assessments (Oriental COCOSDA)*, pages 51–55.

Das, Debopam and Taboada, Maite and McFetridge, Paul. (2015). *RST Signalling Corpus, LDC2015T10*. Linguistic Data Consortium.

Ekbal, A. and Bandyopadhyay, S. (2008). Web-based bengali news corpus for lexicon development and pos tagging. *Polibits*, 37.

Iruskieta, M., Aranzabe, M. J., de Ilarraza, A. D., Gonzalez-Dios, I., Lersundi, M., and de Lacalle, O. L. (2013). The RST Basque Treebank: An online search interface to check rhetorical relations. In *Proceedings of the 4th workshop RST and discourse studies*, pages 40–49.

Prasad, Rashmi and Dinesh, N. and Lee, A. and Miltsakaki, E. and Robaldo, L. and Joshi, A. and Webber, B. (2008). *The penn discourse treebank 2.0*.

Stede, M. (2016). Rhetorische Struktur. In Manfred Stede, editor, *Handbuch Textannotation: Potsdamer Kommentarkorpus 2.0*. Universitätsverlag, Potsdam.

Taboada, M. and Renkema, J. (2008). Discourse relations reference corpus.

Taboada, M. (2008). SFU Review Corpus [Corpus].

Toldova, S., Pisarevskaya, D., Ananyeva, M., Kobozeva, M., Nasedkin, A., Nikiforova, S., Pavlova, I., and Shelepov, A. (2017). Rhetorical relations markers in Russian RST Treebank. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms*, pages 29–33.

van der Vliet, N., Berzlánovich, I., Bouma, G., Egg, M., and Redeker, G. (2011). Building a discourse-annotated Dutch text corpus. In *Beyond Semantics, Bochumer Linguistische Arbeitsberichte 3*, pages 157–171.

A New Version of the *Składnica* Treebank of Polish Harmonised with the *Walenty* Valency Dictionary

Marcin Woliński, Elżbieta Hajnicz, Tomasz Bartosiak
Institute of Computer Science, Polish Academy of Sciences
Jana Kazimierza 5, 01-248 Warsaw, Poland
{wolinski,hajnicz}@ipipan.waw.pl, tomasz.bartosiak@gmail.com

Abstract

This paper reports on developments in the Składnica treebank of Polish which were possible due to the switch to the Walenty valency dictionary. The change required several modifications in the Świgra parser, such as implementing unlike coordination, semantically motivated phrases, and non-standard case values. A procedure to upgrade manually disambiguated trees of Składnica was required as well. Modifications introduced in the treebank included systematic changes of notation and resolving ambiguity between semantically motivated phrases.

The procedure of confronting Składnica treebank with the trees generated with the new version of the Świgra parser using Walenty dictionary allowed us to check the consistency of all the resources. This resulted in several corrections introduced in both the treebank and the valence dictionary.

Keywords: treebank, constituency parsing, valency dictionary

1. Introduction

This article concerns adapting the Składnica treebank to the valence dictionary Walenty. Składnica predates Walenty, so initially the treebank was based on the valence dictionary of Świdziński (1994). Walenty surpasses this dictionary both in size and the number of linguistic phenomena being represented. Therefore, deploying Walenty was an obvious choice for the further development of Składnica.

The procedure to adapt Składnica to the new dictionary was to a large extent automatic. However, the differences between the resources made it necessary to manually correct some parse trees. In this article, we present the method of automatic mapping and the problems that needed manual intervention.

The article is organised as follows. First we present the resources – the treebank Składnica together with the parser Świgra used to generate the trees (Section 2) and the valence dictionary Walenty (Section 3). Next, we discuss adapting the parser to Walenty (Section 4) and then adapting the treebank (Section 5).

2. The treebank and the parser

Składnica is a treebank of Polish (Woliński et al., 2011) built on a sub-corpus sampled from the one million word manually annotated sub-corpus (NKJP1M) of the National Corpus of Polish – NKJP (Przepiórkowski et al., 2012). Corpus samples consist of a few sentences each, and they sum up to 20,000 sentences.

The primary form of the resource comprises constituency trees generated with the DCG (Pereira and Warren, 1980) parser Świgra (Woliński, 2004; Świdziński and Woliński, 2010) and then manually disambiguated and validated (Woliński, 2010). It was assumed as a construction rule for the treebank that all accepted trees have to be actually generated by the parser. The treebank annotators are not allowed to modify trees in any way nor to provide trees for sentences rejected by the parser. The representation of sentences without a proper parse tree has to be corrected by improving the parser's grammar. This leads to an iterative development of the grammar and the treebank. The grammar feeds the treebank and the treebank documents the coverage of the grammar.

The grammar used by Świgra stems from Świdziński's grammar (Świdziński, 1992), but it was deeply restructured. The trees generated by the present version are much simpler and more intuitive. The grammar was also extended in many ways, in particular to describe various forms of coordinated structures.

Figure 1 shows Składnica/Świgra annotation for the sentence:

(1) *Trzeba określić i zbadać rodzaj infekcji*
must determine and study kind infection
oraz co ją powoduje.
and what it causes

'It's necessary to determine and confirm the kind of infection and what causes it.'

Leaves of the tree correspond to terminals (represented with a form and a lemma in the picture). Internal nodes correspond to phrases. They are represented by the name of the non-terminal category in Fig. 1. The labels use abbreviations of Polish names, which are explained in Table 1. In fact, each node carries several attributes specifying its syntactic features (all such features of a fw node are shown in Fig. 2). Children of a given node are its constituents, as determined by some rule of the grammar being used. An important feature of Składnica trees is the fact that one of the constituents is marked as the syntactic head (marked with an edge with a thick grey background), which allows to convert constituency trees to dependency trees. Such conversion has in fact been performed resulting in a dependency version of Składnica (Wróblewska and Woliński, 2012), later on converted also to Universal Dependencies (Seddah et al., 2013). Non-terminals of the grammar fall into several types or layers in the tree (Świdziński and Woliński, 2010). From the bottom up, these are:

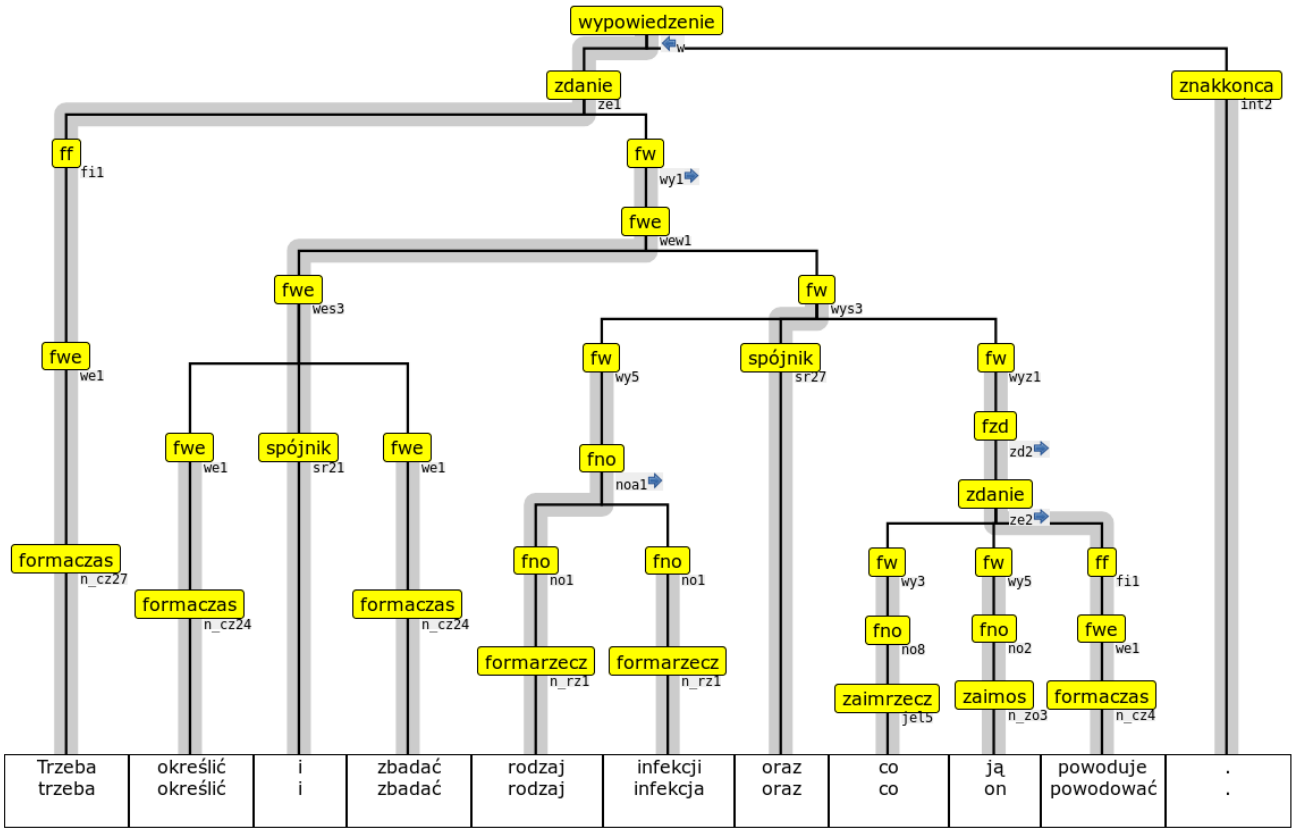


Figure 1: A Świgra parse tree for the example (1)

syntactic words	
formaczas	verbal form
formarzecz	nominal form
zaimrzecz	nominal pronoun
zaimos	personal pronoun
spójnik	conjunction
constituent phrases	
fno	nominal phrase
fwe	verbal phrase
ff	finite phrase (an fwe that can constitute a clause)
fzd	clausal phrase
valency phrases	
fw	required phrase (argument)
fl	free phrase (adjunct)
clauses	
zdanie	clause/sentence
wypowiedzenie	utterance

Table 1: Non-terminal categories of Fig. 1

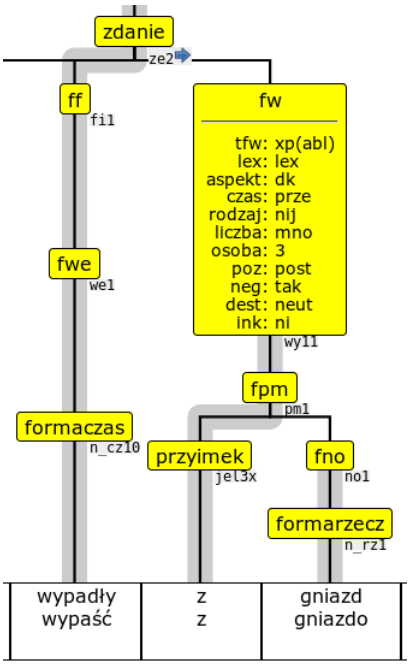
- *Syntactic words* form the syntactic counterpart of terminals. Typical examples are the units formaczas, formarzecz, zaimrzecz, and zaimos in Figure 1. However, units of this level can also represent multi-token verbal forms (e.g., analytical future forms of verbs *będzie czytać* 'will read') and other cases where one form, from the syntactic viewpoint, corresponds to several tokens in the NKJP tagset, e.g. two-word prepositions *wraz z* 'together with' and adverbs *po ciemku* 'in the dark'.

- *Constituent phrases* are used to describe the attachment of various modifiers to verbal, nominal, adjectival, and adverbial heads. Also at this level, prepositional-nominal phrases and subordinate clauses are formed. Constituent phrases can also be coordinated structures (with a conjunction as a head).

- *Valency phrases*, as proposed by Świdziński (1992), denote functions played by constituent phrases. These differentiate dependents into required phrases (arguments) fw and free phrases (adjuncts) fl. Thanks to their presence, the valency structure gets visible in the tree.

- The fourth layer comprises *clauses*. Simple clauses consist of a finite phrase and valency phrases. Coordinate clauses, based upon a conjunction as their head, have other clauses as their constituents.

As can be seen in Fig. 2, the first attribute of a required phrase fw is tfw – the 'type of required phrase'. This attribute shows the characteristic of the phrase assigned by the valence dictionary. Some of the other attributes of fw and fl are, somewhat counter-intuitively, shared with the head of the clause. This allows to confront some of the attributes of the given dependent with the attributes of the head, for example to require gender and number agreement of the verb and the subject.



wypadły *z* *gniazd*
fall from nests

'[they] have fallen out of nests'

Figure 2: Complete set of features for one required phrase fw of type xp(abl)

3. Valency dictionary Walenty

A valency dictionary specifies what types of arguments are possible for a given predicate. In Polish, the need of such information is most obvious for verbs, which differ widely in possible arguments, e.g., some verbs allow for a complement in the form of a verbal phrase in the infinitive and others don't.

Initially, Świgra used a valency dictionary based on (Świdziński, 1994). This dictionary was extended during the treebank development project. At the end of the project the dictionary consisted of 6400 schemata for 1450 Polish verbs (covering about 75% of verb occurrences in NKJP1M).

Later, this dictionary became a seed for a new one, which is currently being developed at ICS PAS. Walenty is a comprehensive valency dictionary of Polish based on corpus data (Hajnicz et al., 2016b; Przepiórkowski et al., 2014c; Przepiórkowski et al., 2014b; Przepiórkowski et al., 2014a; Hajnicz et al., 2016a). After a few years of development, Walenty contains 99,000 schemata for 18,100 predicates, which include 13,000 verbs, 4,000 nouns and 1100 adjectives and adverbs. Walenty covers 99,8% of occurrences of verbal forms in the 300 millions word balanced sub-corpus of NKJP. Moreover, Walenty is much more rich in information than the original Świgra dictionary.

The dictionary uses so-called structural case, i.e. case whose morphological realisation depends on the syntactic context. It is used in two contexts – as the case of nominal subjects and as the case of nominal phrases underlying the genitive of negation. For Świgra, a nominal subject is simply in the nominative (np(nom))[1]. For the structural case underlying the genitive of negation we use a mnemotechnical symbol np(accgen), since this type of phrase is realised in the accusative or in the genitive, depending on negation.

A similar mechanism is used to represent so called partitive nominal phrases, which can be realised in the accusative or in the genitive case with a slight difference in the meaning. This type of phrase is represented in Walenty with a special value part for the grammatical case.

The dictionary provides semantic classification of some adverbial-like arguments (e.g., ablative and adlative), denoted in Walenty as xp(...). Such valency positions can be filled mainly with adverbs and prepositional phrases, but the subtype of xp explicitly specifies a semantically motivated set of allowed phrase types. For example xp(abl) – ablative phrase, marking departure point of a motion – can be realised (among others) by adverbs *stąd* 'from here', *znikąd* 'out of nowhere', or prepnp(z,gen) – phrases with the preposition *z* 'from'. Adlative phrases xp(adl) denote point of arrival: *tutaj* 'here', *naprzód* 'forward', prepnp(do,gen) – *do* 'towards', complex preposition comprepnp(w kierunku) 'in the direction of', or even clauses, e.g. cp(rel[dokąd;gdzie]) – a relative clause limited to two relative pronouns *dokąd* 'where to' and *gdzie* 'where'.

In total, there are 10 specific subtypes of xp – expressing time, duration, place, starting or ending point, path, tool, manner, cause, or aim.

The following example depicts one of syntactic schemata for the verb określić 'specify/determine' that is used in the tree of Figure 1:

subj	obj		
np(nom)	np(accgen) cp(int) cp(że) ncp(accgen,int) ncp(accgen,że)	prepnp(w,loc)	xp(mod)

Each column represents a single syntactic position. Two positions are labelled: subject subj (the argument in this position influences morphological features of the finite verb) and passivable object obj (the argument in this position turns into a subject in passive voice).

Phrases that can fill a given position are specified by their non-terminal category and selected grammatical features. In the example, the subject position can be realised by a nominal phrase in the nominative case np(nom). The object position can be realised by a nominal phrase in a structural case np(accgen) or an interrogative clause cp(int) or a clause with the complementizer że 'that' – cp(że), and two other. (For a full list of available phrase types see (Hajnicz et al., 2016b).)

Walenty describes coordination of syntactically different arguments within a single syntactic position (so called unlike coordination). The fact that given phrase type specifications are listed within a single position is to mean that arguments of these types can be coordinated. An example of such coordination can be seen in Figure 1, where a nominal

[1] Świgra does not follow Przepiórkowski's concept of numeral subjects being in the accusative (Przepiórkowski, 2004).

phrase *rodzaj infekcji* 'kind of infection' gets coordinated with a clause *co ją powoduje* 'what causes it'. If the coordination was not possible, separate schemata with respective phrase types would be given.
The following table shows a schema for the other verb occurring in the example sentence — ZBADAĆ 'examine':

subj	obj	
np(nom)	np(accgen) ncp(accgen,int) cp(int)	xp(instr)

The third position in this schema can be realised by a phrase of type xp and subtype instr – instrument/tool.
Other phenomena represented in Walenty include syntactic control and raising, but these are not implemented in Świgra yet.
Walenty includes a rich phraseology component (Przepiórkowski et al., 2014a; Hajnicz et al., 2016b). It aims at precise representation of the structure of lexicalised arguments. In particular, it is used to represent all acceptable dependants of complex prepositions.
In Walenty, due to the free word order of Polish, the order of positions within a schema and the order of argument types within a position is not important.
Valency schemata given by Walenty are maximal – the dictionary does not list possible sub-schemata of a given schema. In Polish most of arguments are optional (in particular subjects are often omitted).
The syntactic layer of Walenty is being currently complemented with semantic frames (Hajnicz et al., 2016a).

4. Adapting Świgra to Walenty

Adopting Walenty was an obvious step in the development of Świgra but it meant that some changes needed to be introduced in the grammar to take advantage of the more detailed description. Simple changes included translating the symbols used in the old dictionary, which were based on Polish abbreviations for some grammatical categories, to those used in Walenty (Latin/English based).
A much more fundamental change was the introduction of coordination within syntactic positions. An example of such coordination can be seen in Fig. 1. The phrase *rodzaj infekcji* 'kind of infection' is analysed as a nominal phrase fno in the accusative, which turns into a required phrase fw of type np(accgen). The sentence *co ją powoduje* 'what causes it' becomes a clausal phrase fzd of type int – interrogative (*co* 'what' is an interrogative pronoun), and then a required phrase fw of type cp(int). These two required phrases get coordinated to become a phrase of type [np(accgen),cp(int)].
A mechanism was introduced that checks that this composite type is a subset of the appropriate position in some schema for the given verb. As can be seen, it is the case with the quoted schemata of the verbs OKREŚLIĆ 'determine' and ZBADAĆ 'study'.
What makes the example even more interesting, the two verbs are also coordinated and form a complex verbal phrase *określić i zbadać*. Syntactic schemata for both verbs differ and even the respective obj positions differ. Nonetheless, both schemata contain a position that is a superset of the type [np(accgen),cp(int)], which allows to accept the sentence. As can be seen, the use of Walenty's schemata can be quite complicated, which means an efficient way of using those had to be developed (Woliński, 2015).
Walenty applies the label subj not only to nominal phrases in the nominative, but also to some other phrases, e.g. cp(że). It was decided to interpret subjects in the same way in Świgra, which means new rules had to be added for those realisations. As a result, much fewer verbs are inherently subjectless in this new interpretation.
New rules had also been added to implement special types of arguments present in Walenty: nonch, refl, cp(żeby2) and complex prepositions.
Semantically motivated phrase types xp(...) had also to be implemented. The old dictionary used a much less precise general type advp, so respective rules had to be replaced with ones defining the possible subtypes.
To use lexicalised schemata from Walenty it was necessary to make the lemma of the head of each phrase available. In DCG information is only available "locally" – a grammar rule can only access the category and the information available as attributes of a given node. So it was necessary to add attributes that carry the information on the lexical head along the 'head branch' of each subtree. With these changes Świgra now uses phraseological schemata of Walenty (although the complete analysis of embedded modifiers of lexicalised items is not yet performed).

5. Adapting Składnica to Walenty

The core reason to use Walenty in Świgra was to introduce its rich information to the Składnica treebank. But that required some non-trivial operations to be performed on the treebank.
The system used to manually disambiguate trees generated with Świgra (Woliński, 2010) includes a module to automatically re-annotate a parse forest generated with a changed grammar preserving the tree previously chosen by annotators. However, in the form previously implemented, the system sought for a tree literally identical to the previously selected one. Due to new features from Walenty, some systematic changes had to be allowed between the old and the new trees. An algorithm was implemented that accepts the tree as matching if it differs only in the pre-specified way.
To make the upgrade procedure easier to manage, the changes required to adopt Walenty were split into a few sets of independent changes, which were applied incrementally. Each set of changes was tested against the treebank and necessary corrections were performed. The corrections involved the rules of the grammar, valency schemata of Walenty, or arguments selected for particular sentences in the treebank. This way all three resources were tested against each other.
In the first step, Walenty was mapped to a form close to the original dictionary with the intention to detect incompatible changes in valency schemata. At this stage Polish names of grammatical categories were used; all xp(...) phrases were mapped to generic advp; and lexical heads were introduced in the grammar and confronted with lexicalised schemata of Walenty. After re-parsing of the corpus, schemata from

Walenty have been confronted with arguments selected by annotators.

At the beginning of procedure there were 10,673 accepted trees in Składnica. The tree previously accepted by the annotators was found among new parses in 10,193 cases (95.5%). For the remaining 480 sentences (4.5%) the parser using Walenty did not produce a compatible tree (in 255 cases (2.4%) the new parse forest was empty). Preliminary analysis has shown that these sentences exhibit several problems including errors both in Składnica and in Walenty. In particular, for some verbs the two dictionaries differ whether a given dependent should be considered a complement or an adjunct. We have decided to upgrade the rest of the treebank and present those problematic sentences for a new assessment of treebank annotators.

In following steps, which were mostly automatic, the symbols used for types of phrases were made consistent with Walenty and the subj label was added to respective phrases. The last step was devoted to the introduction of semantically motivated xp(...) phrases. The advp specification in the old dictionary was very general: this type of phrase could be realised by any adverbial phrase or any prepositional-nominal phrase prepnp. The annotators were free to decide whether a particular prepositional phrase can be interpreted as advp in a given context. We expected many problems in matching these types.

It turned out that in about 300 sentences some of the advp phrases in the old trees did not match any subtype of xp in the new ones. The list of sentences with this problem was analysed and the problems resolved in one of the following ways:

- the old advp was replaced in the treebank with a specific prepositional phrase in accordance with a schema present in Walenty,
- a schema of Walenty needed to be amended by a particular subtype of xp or prepnp phrase,
- the offending phrase was changed from required fw to free fl (adjunct) in the treebank,
- a new realisation for some subtype of xp had to be added.

Another type of a problem, that showed up in the process, was ambiguity of the advp specification. Some phrases can be interpreted as xp of various subtypes. For example *gdzieś* 'somewhere' can be xp(loc) – locative or xp(adl) – ablative. The phrases *przez most* 'through a bridge' and *przez godzinę* 'during one hour' both are prepnp(przez,acc) in Polish, so they both qualify as realisations of xp(perl) or xp(dur), but only the first is really perlative (expresses a path of a movement) and the second – durative. The list of about 200 sentences containing such ambiguities was given to an expert, who decided for each of them, which interpretation to choose.

We are aware that some problems remain after the update procedure. Nominal phrases are not typical realisations of xp phrases. The only exception is np(inst), which is a possible realisation of xp(dur) (*czekać godzinami* 'to wait for hours') and xp(perl) (*jechać drogą* 'to drive along the road'). Such realisations were absent in the old valency dictionary, so such phrases were considered adjuncts in the treebank. Moreover, for some verbs of movement, which allow for an xp(perl) argument, the schemata of Walenty contain both xp(perl) and np(inst) (*jechać samochodem* 'to drive a car'). Only the np(inst) argument was present in the old dictionary, and could be used for both types of arguments. Unfortunately, occurrences of these problems could not be detected automatically. To make the annotation consistent with Walenty some more manual corrections will be needed.

In total, Świgra with the changes described in Section 4 was able to accept 14,103 sentences of the Składnica corpus (70.5%), while the version with the old dictionary accepted 13,194 (66%). These newly accepted sentences include many sentences with verbs missing in the old dictionary, but also interesting examples of various coordinated phrases. These sentences are currently being assessed by the annotators.

6. Conclusions

Składnica is the first treebank of Polish of a considerable size. The resource is now coupled with an independently developed valency dictionary, which marks an important turning point in its development. The fact that Walenty is actively maintained makes further development of the parser easier. From the other point of view Składnica provides verification for schemata of Walenty.

The current version of Składnica can be downloaded from the address `http://zil.ipipan.waw.pl/Składnica`. The new version is also already available in the treebank search engine: `http://treebank.nlp.ipipan.waw.pl/`.

As said above, the treebank is being currently enlarged with the sentences accepted thanks to Walenty. We hope to reach the level of 70% analysed sentences in a few months.

Future plans for Świgra and Składnica include deployment of the non-verbal part of Walenty (this is are relatively easy task since non-verbal schemata are simpler and use the same types of phrases as verbs). The new version of Składnica will also be converted to the dependency form and used for training dependency parsers. An interesting question is whether the new features of the treebank (in particular types of xp phrases) can help in training statistical disambiguation tools and parsers. Another direction of development is to use the semantic layer of Walenty to generate predicate-argument structures using semantic role labels. In fact, the work in this direction has already started.

Acknowledgements

Work financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

Bibliographical References

Hajnicz, E., Andrzejczuk, A., and Bartosiak, T. (2016a). Semantic layer of the valence dictionary of Polish *Walenty*. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language*

Resources and Evaluation, LREC 2016, pages 2625–2632, Portorož, Slovenia. ELRA, European Language Resources Association (ELRA).

Hajnicz, E., Patejuk, A., Przepiórkowski, A., and Woliński, M. (2016b). Walenty: słownik walencyjny języka polskiego z bogatym komponentem frazeologicznym. In Karolina Skwarska et al., editors, *Výzkum slovesné valence ve slovanských zemích*, pages 71–102. Slovanský ústav AV ČR, Prague.

Pereira, F. and Warren, D. H. D. (1980). Definite clause grammars for language analysis–a survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence*, 13:231–278.

Przepiórkowski, A., Bańko, M., Górski, R.L., and Lewandowska-Tomaszczyk, B., editors. (2012). *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Przepiórkowski, A., Hajnicz, E., Patejuk, A., and Woliński, M. (2014a). Extended phraseological information in a valence dictionary for NLP applications. In *Proceedings of the Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP 2014)*, pages 83–91, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Przepiórkowski, A., Hajnicz, E., Patejuk, A., Woliński, M., Skwarski, F., and Świdziński, M. (2014b). Walenty: Towards a comprehensive valence dictionary of Polish. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 2785–2792, Reykjavík, Iceland. ELRA.

Przepiórkowski, A., Skwarski, F., Hajnicz, E., Patejuk, A., Świdziński, M., and Woliński, M. (2014c). Modelowanie własności składniowych czasowników w nowym słowniku walencyjnym języka polskiego. *Polonica*, XXXIII:159–178.

Przepiórkowski, A. (2004). O wartości przypadka podmiotów liczebnikowych. *Biuletyn Polskiego Towarzystwa Językoznawczego*, LX:133–143.

Seddah, D., Tsarfaty, R., Kübler, S., Candito, M., Choi, J. D., Farkas, R., Foster, J., Goenaga, I., Gojenola Galletebeitia, K., Goldberg, Y., Green, S., Habash, N., Kuhlmann, M., Maier, W., Marton, Y., Nivre, J., Przepiórkowski, A., Roth, R., Seeker, W., Versley, Y., Vincze, V., Woliński, M., and Wróblewska, A. (2013). Overview of the SPMRL 2013 shared task: A cross-framework evaluation of parsing morphologically rich languages. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 146–182, Seattle, WA. Association for Computational Linguistics.

Świdziński, M. and Woliński, M. (2010). Towards a bank of constituent parse trees for Polish. In Petr Sojka, et al., editors, *Text, Speech and Dialogue: 13th International Conference, TSD 2010, Brno, Czech Republic*, volume 6231 of *Lecture Notes in Artificial Intelligence*, pages 197–204, Heidelberg. Springer-Verlag.

Świdziński, M. (1992). *Gramatyka formalna języka polskiego*. Rozprawy Uniwersytetu Warszawskiego. Wydawnictwa Uniwersytetu Warszawskiego, Warszawa.

Świdziński, M. (1994). Syntactic dictionary of Polish verbs. Manuscript, Uniwersytet Warszawski i Universiteit van Amsterdam.

Woliński, M., Głowińska, K., and Świdziński, M. (2011). A preliminary version of Składnica—a treebank of Polish. In Zygmunt Vetulani, editor, *Proceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pages 299–303, Poznań, Poland.

Woliński, M. (2004). *Komputerowa weryfikacja gramatyki Świdzińskiego*. Ph.D. dissertation, Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Woliński, M. (2010). Dendrarium – an open source tool for treebank building. In Mieczysław A. Kłopotek, et al., editors, *Proceedings of IIS'2010*, pages 193–204. Wydawnictwo Akademii Podlaskiej.

Woliński, M. (2015). Deploying the new valency dictionary Walenty in a DCG parser of Polish. In Markus Dickinson, et al., editors, *Proceedings of the Fourteenth International Workshop on Treebanks and Linguistic Theories (TLT 14)*, pages 221–229, Warsaw. Institute of Computer Science, Polish Academy of Sciences.

Wróblewska, A. and Woliński, M. (2012). Preliminary experiments in Polish dependency parsing. In Pascal Bouvry, et al., editors, *Security and Intelligent Information Systems: International Joint Conference, SIIS 2011, Warsaw, Poland, June 13-14, 2011, Revised Selected Papers*, volume 7053 of *Lecture Notes in Computer Science*, pages 279–292. Springer-Verlag.

Language Resource References

Elżbieta Hajnicz *et al.* (2017). *Walenty valency dictionary of Polish*. Institute of Computer Science, Polish Academy of Sciences, `http://walenty.ipipan.waw.pl`, v. 1.2.

Marcin Woliński. (2011). *Składnica – a constituency and dependency treebank of Polish*. Institute of Computer Science, Polish Academy of Sciences, `http://zil.ipipan.waw.pl/Składnica`, v. 0.5.

Marcin Woliński. (2017a). *Składnica – a constituency and dependency treebank of Polish*. Institute of Computer Science, Polish Academy of Sciences, v. 0.8.

Marcin Woliński. (2017b). *Świgra 2 – a constituency parser of Polish*. Institute of Computer Science, Polish Academy of Sciences, v. 1.9.

Parse Me if You Can: Artificial Treebanks for Parsing Experiments on Elliptical Constructions

Kira Droganova,[1] **Daniel Zeman,**[1] **Jenna Kanerva,**[2] **Filip Ginter**[2]
[1]Charles University, Faculty of Mathematics and Physics, Praha, Czechia
[2]TurkuNLP Group, University of Turku, Finland
[1]{droganova, zeman}@ufal.mff.cuni.cz, [2]{jmnybl, figint}@utu.fi

Abstract

In this work we focus on a particular linguistic phenomenon, ellipsis, and explore the latest parsers in order to learn about parsing accuracy and typical errors from the perspective of elliptical constructions. For this purpose we collected and processed outputs of several state-of-the art parsers that took part in the CoNLL 2017 Shared Task. We extended the official shared task evaluation software to obtain focused evaluation of elliptical constructions. Since the studied structures are comparatively rare, and consequently there is not enough data for experimentation, we further describe the creation of a new resource, a semi-artificially constructed treebank of ellipsis.

Keywords: Ellipsis, Syntactic parsing, Evaluation, Universal Dependencies

1. Introduction

Ellipsis, i.e. omission of linguistic content that is silently understood by both the speaker and the addressee, is a phenomenon present—in various forms—in many natural languages. Ellipsis obviously makes natural language understanding harder; but sometimes it also complicates syntactic parsing of the content that is not omitted. In dependency syntax (which is the framework within which we operate), a parent node may be missing while its dependents are present. One might either create an "empty" node for the missing word, or choose a substitute parent among the words that are not missing. Both options make parsing difficult: in the former case, the parser must learn where to generate empty nodes; in the latter, relations are drawn between nodes that would not be connected otherwise, hence they are not easily learned from data.

In any case, modern dependency parsers *are* data-driven and they can hardly account for those types of ellipsis that are not represented in training data. If the data contains empty nodes, the parser can try to learn generating them. If the data does not contain any specific annotation of ellipsis, we have to hope that the parser learns to occasionally attach dependents to strange parents, even without knowing that it is ellipsis what caused the lack of better options.

In this study we focus on elliptical constructions in the so-called *basic representation* of Universal Dependencies (UD) (Nivre et al., 2016). The annotation style of UD does not mark ellipsis explicitly when it does not have to: most types are solved by simply promoting one orphaned dependent to the position of its missing parent. Admittedly, there are treebanks that overtly annotate a wider range of elliptical structures. Our main reason for working with UD is practical: substantial data is available in this annotation style for several dozens of languages, and state-of-the art parsers have been trained and tested on UD.

The one exception where UD explicitly marks ellipsis are certain types of gapping and stripping (Droganova and Zeman, 2017), where multiple orphaned dependents of a missing predicate have to be connected using a special relation called `orphan` (Figure 1). In the present work we investigate how frequent are the `orphan` relations in data, how well can existing parsers learn to recognize them, and how can we extend the data to provide more training material and improve parsing accuracy.

2. Data

For the purpose of the experiments we use the system outputs from the CoNLL 2017 Shared Task (Zeman et al., 2017), that are now available as a corpus. We chose 12 teams whose systems surpassed baseline results (Zeman et al., 2017) on labelled attachment score (LAS): C2L2, darc, HIT-SCIR, IMS, Koç University, LATTICE, NAIST-SATO, Orange-Deskiñ, Stanford, TurkuNLP, ÚFAL-UDPipe 1.2 and UParse.

3. Experiments

The idea behind this work is to look closely at the current parsers regarding their ability to parse non trivial linguistic constructions such as elliptical constructions, and collect the information about typical errors, how they differ from parser to parser.

For the purpose of this experiment we adapted and extended the evaluation script which had been created to evaluate system output files for the 2017 Shared Task. The main idea of such adaptation is to save evaluation techniques that were proposed and implemented by the 2017 task organizers. Since the data was selected relying on these techniques, we hope that following the same line, especially regarding word alignments and sentence segmentation, helps us to be more precise. The script is available at the Shared Task page.[1] The adapted script can be found on GitHub.[2]

The adapted script provides information of two types:

- Statistics on correctly predicted `orphan` relations;
- Statistics on erroneously predicted or missed `orphan` relations and typical errors.

[1] `http://universaldependencies.org/conll17/evaluation.html`

[2] `https://github.com/Kira-D/conll2017/tree/deprelCalc`

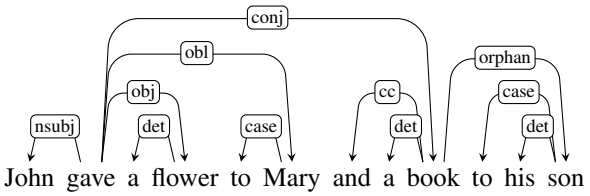


Figure 1: UD v2 uses the orphan relation to attach unpromoted dependents of a predicate to the promoted dependent.

4. Evaluation

Table 1 shows the statistics on correctly predicted orphan relations. In these calculations we use the relative number of all orphan nodes for every team, which is based on alignment between system output words and gold standard words. In other words, only successfully aligned orphan nodes from gold standard are included in this number. It is clearly seen that both Recall and F-measure are rather low. At the same time, percentage of correctly predicted dependency labels for head nodes is quite high.

Table 2 shows the statistics on erroneously predicted or missed orphan labels. For every parser that we selected for the experiment, we calculate error pairs "relation1-relation2", where the first relation was taken from the aligned gold word and the second relation was assigned by the system. Table 2 provides top 5 error pairs. Every cell contains the following information:

- the error pair;
- the contribution of the pair to the number of all errors concerning orphan label (percentage);
- the number of instances of the error type (frequency);
- h.error shows erroneously predicted head nodes (percentage and absolute number).

It seems that parsers make mistakes in similar conditions: the error types and their frequencies are almost the same from parser to parser.

What is important, the number of orphan labels is just a tiny fraction of all labels and the contribution of their low values of Recall and F-measure to the final figures calculated on the whole amount of data goes virtually unseen. Hence, the question is if the parsers perform really poorly on elliptical constructions or it is simply the lack of data. To answer that question, we would need more data; since naturally occurring ellipsis is rare, we decided to artificially create a set of sentences that are structurally similar to natural elliptical constructions.

Parser	All	Correct	Recall	F1	Parent	Parent %
C2L2	1420	217	15.28%	26.48%	192	88.48%
darc	1411	194	13.75%	19.06%	180	92.78%
HIT-SCIR	1411	341	24.17%	34.13%	292	85.63%
IMS	1421	241	16.96%	28.83%	208	86.31%
Koc-University	1420	194	13.66%	20.78%	161	82.99%
LATTICE	1420	200	14.08%	20.62%	166	83.0%
NAIST-SATO	1420	391	27.54%	41.53%	357	91.3%
Orange-Deskin	1420	369	25.99%	35.16%	280	75.88%
Stanford	1420	454	31.97%	49.11%	408	89.87%
TurkuNLP	1420	218	15.35%	23.37%	189	86.7%
UFAL-UDPipe-1-2	1423	226	15.88%	23.69%	182	80.53%
UParse	1420	326	22.96%	33.44%	288	88.34%

Table 1: Correctly predicted orphan relations. Parser: names of the teams in alphabetic order; All: number of orphan labels; Correct: number of correctly predicted orphan labels; Recall: number of correct orphan labels divided by the number of gold-standard orphan nodes; F1: F-measure: 2PR / (P+R); Parent: number of correctly predicted parent nodes; Parent %: percent of correctly predicted parent nodes;

5. Creating Artificial Treebanks

Recent research (Schuster et al., 2017; Droganova and Zeman, 2017) provides a detailed overview of elliptical constructions within the UD framework and presents typical patterns that can be used for detection of elliptic constructions. This information allows us to develop a script that transforms non-elliptic UD style trees to elliptic trees.

Figure 2 shows a subtree pattern that matches sentences where gapping (Johnson, 2009) could potentially occur (but it did not, or at least it was not annotated following the UD guidelines, because there is no orphan relation). An example of an English sentence that matches the pattern: "But not always do those three agree, and not always are their decisions equal."

Figure 3 provides the tree structure of this sentence. It matches the pattern because 1. its "root" node is a verb; 2. the verb has an "aux" child; 3. the verb is linked with another clause via a "conj" relation; 4. the other clause is headed by an adjective and has a "cop" (copula) dependent; 5. both clauses contain a "nsubj" (subject) and an "advmod" (adverbial modifier). After transformation the sentence would lose an adjective and its dependent. The new structure is shown in Figure 4.

It should be mentioned that patterns do not require a particular word order, only particular dependents. Thus the sentence in Figure 5 is a match as well.

The methodology requires manual efforts. After application of the script, the data have to be checked and corrected:

- After artificial omission sentences must remain grammatically correct (Figure 10, Figure 11);
- The patterns are designed to match as many instances as possible, so the erroneous instances have to be filtered out or manually corrected. All sentences at Figures 3, 5 and 10 match the pattern at Figure 2, but after conversion sentence at the Figure 11 becomes ungrammatical. However, it can be fixed manually (Figure 12).

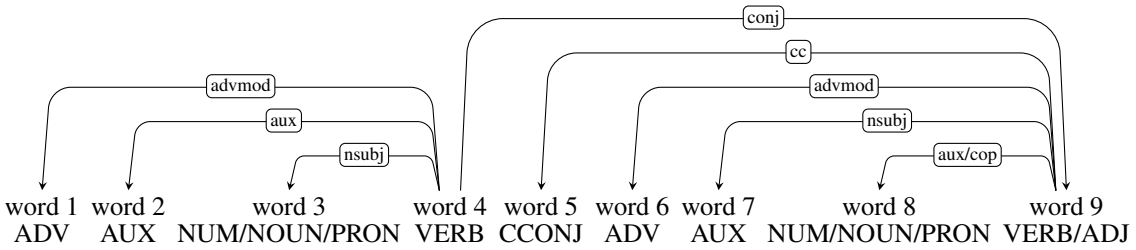


Figure 2: An example of a gapping pattern. Unless explicitly said otherwise, patterns are not word-order-sensitive.

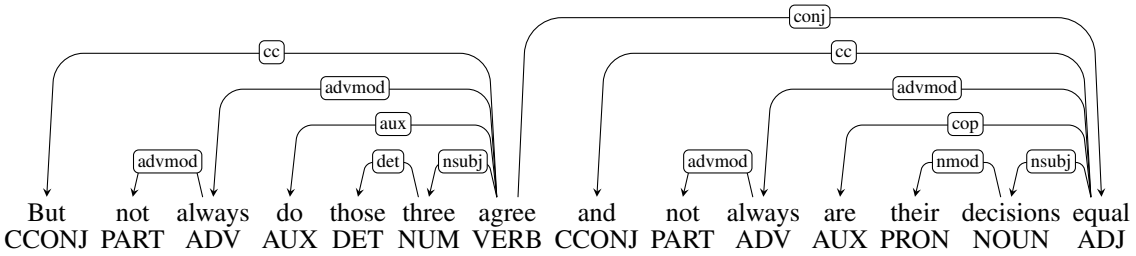


Figure 3: An example of a matched sentence (before conversion).

5.1. Input Data

Our methodology can in principle be applied to any UD treebank. The dataset and experiments presented in this paper are based on Czech, English and Finnish treebanks from UD 2.1 (Nivre et al., 2017).

In addition, large web corpora of the three languages (Zeman et al., 2017) (Ginter et al., 2017) were parsed by two parsers (Stanford (Dozat et al., 2017) and Baseline UDPipe (Zeman et al., 2017) entries in the CoNLL17 Shared Task on Multilingual Parsing from Raw Text to Universal Dependencies) and used as an additional source of trees to get more candidate material. After double parsing, only trees with identical analysis were kept to ensure the quality of the automatic parses. An informal manual inspection of these trees confirmed that the quality is sufficient. All input data are in the CoNLL-U format.[3]

For English, there are four UD treebanks: "Original", LinES, ParTUT, PUD. For Czech, we used the training parts of "Original", CAC and FicTree. For Finnish, there are three treebanks in UD but only the "Original" UD_Finnish was used in the present work, and only for development of the rules, which were then applied to the parsed web corpus to create the final artificial dataset.

5.2. Conversion

The conversion was performed using Udapi, an open-source framework that provides an application programming interface for processing Universal Dependencies data (Popel et al., 2017). Udapi components that were developed for the conversion are not integrated into the framework and cannot be obtained from the the official Udapi project repository due to their extremely narrow specialization. The components can be found on GitHub.[4]

The algorithm consists of three steps. First, we use existing Udapi functionality to filter out sentences that cannot be converted into artificial sentences in any way. For this purpose we composed a filtering query that restricts the structure of a candidate sentence. For instance, gapping occurs in coordinate structures and typically the second clause contains a gap (Coppock, 2001), therefore a candidate sentence must contain at least two clauses and the candidate word must be in the second. By candidate word we mean the token that will be deleted or transformed in the next step.

Another example of a filtering rule is the requirement that the heads of both clauses have at least 3 dependents that are neither function words nor punctuation. (That is, they are arguments or adjuncts.)

One of our observations suggests that elided elements can hardly be restored if some other clause is encountered between the two clauses. Therefore we delete such sentences in the first step.

The filtering query simplifies visualization of candidate trees which is helpful for adjustment of the rules.

Second, we use the Udapi components that we developed in order to propose the conversion. The first component analyzes candidate words within a sentence and duplicates the sentence if there are two or more words marked as candidate and not all of them are verbs. Simultaneously, the component leaves only one word marked as candidate for each copy. The verb restriction prevents such sentences from duplication:

(1) Mary won gold, Peter won silver, and Jane won bronze.

This rule was designed for the following purposes:

- If a candidate sentence contains three or more coordinate clauses, we do not want to allow gapping in the second clause if it does not occur in the third clause. Either both must be gapped or none of them.

[3] http://universaldependencies.org/format.html

[4] https://github.com/Kira-D/UDapy_block_artificial

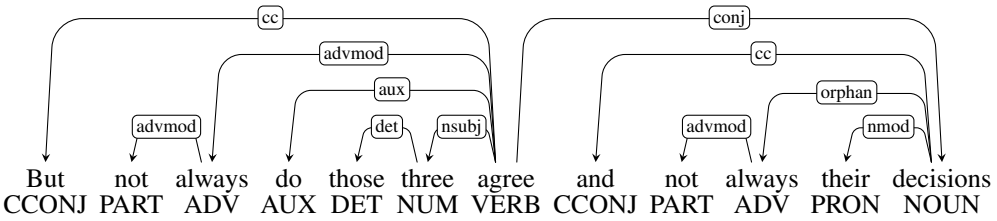


Figure 4: The matched sentence from Figure 3 after conversion.

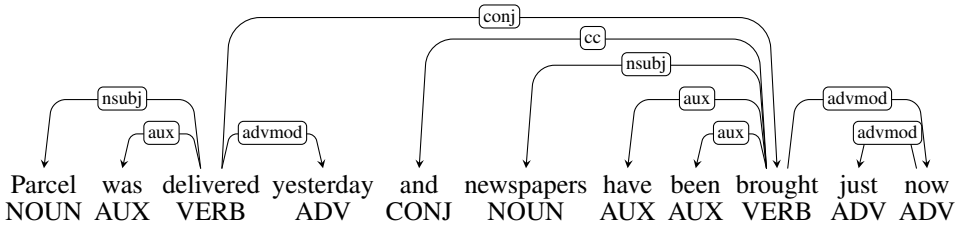


Figure 5: An example of a matched sentence.

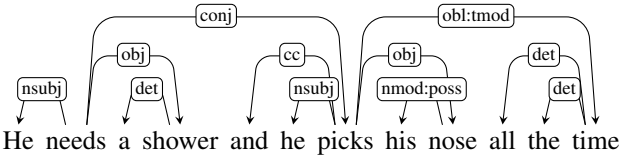


Figure 6: An example of a sentence before Type 1 conversion.

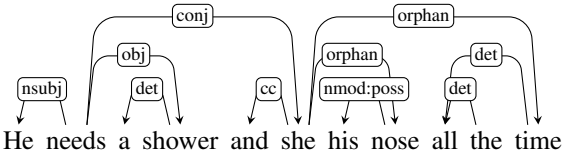


Figure 7: The sentence from Figure 6 after Type 1 conversion.

- If a candidate sentence contains three or more coordinate clauses and some of them are copular clauses, due to the conversion rules the output sentence will contain different types of ellipsis. Consequently, the output structure is not parallel—dependent clauses do not repeat the structure of the main clause. Conversion rules accept copular clauses, but can convert only non-core ellipsis (Figure 8) from such constructions and only if they have enough non-core dependents.

The second component analyzes candidate words and proposes the conversion according to the rules.
The proposed conversion is stored in the CoNLL-U file[5] for every token that should be changed. The information contains either new position in the sentence, parent token and dependency relation, or a marker that the token should be deleted.
The component proposes conversion of two types:

- Type 1 (Core argument ellipsis)—a verb and its auxiliary verbs are marked as 'to be deleted'. Figure 6 and Figure 7 show an example of such conversion.
- Type 2 (Non-core argument ellipsis)—some or even all core arguments are marked as 'to be deleted'. Figure 8 and Figure 9 illustrate this type.

If no rule can be applied to the sentence, the sentence is marked as 'to be deleted'.

[5]The tenth column, MISC, allows for storing arbitrary additional annotation.

Some additional markers were used during the conversion process to adjust the rules and to make sure that the rules cover most of the sentences present in the data.
Third, we use the Udapi component that was developed to perform the final conversion:

- Delete the sentences that are marked as 'to be deleted';
- Delete the tokens that are marked as 'to be deleted';
- Change the sentence structure according to the information that is stored in the MISC column;
- Delete all markers that were added during conversion.

5.3. Complex Issues

We iteratively adjusted the conversion rules, manually checking output samples, making the rules more strict and precise, and re-running them.
However, the output data is still not perfect. We attribute this issue to lack of information: Our rules do not have access to a dictionary and cannot always assess the degree to which the dependents of the two verbs are semantically compatible (Figure 11). Some of the complex sentences may be correctly converted using grammatical cases (see section 5.4.). Some language-specific relation types may help, for instance, English has *obl:tmod* (temporal modifier), a subtype of *obl* (oblique argument or adjunct). Unfortunately, rules relying on relation subtypes are not always applicable to trees produced by parsers because some parsers are trained to only generate the main dependency type—*obl* in this case.

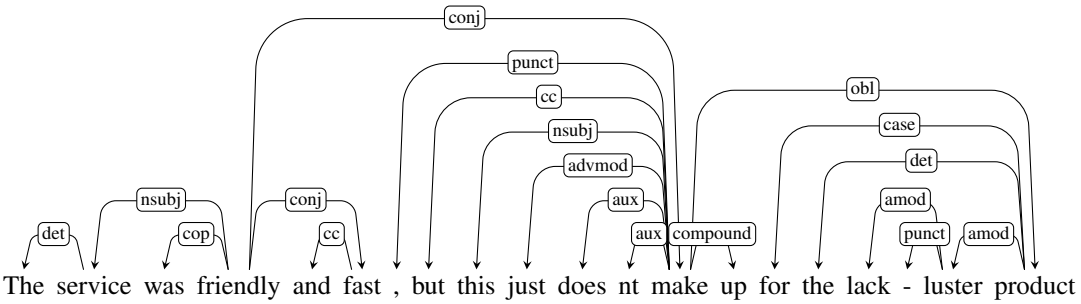


Figure 8: An example of a sentence before Type 2 conversion.

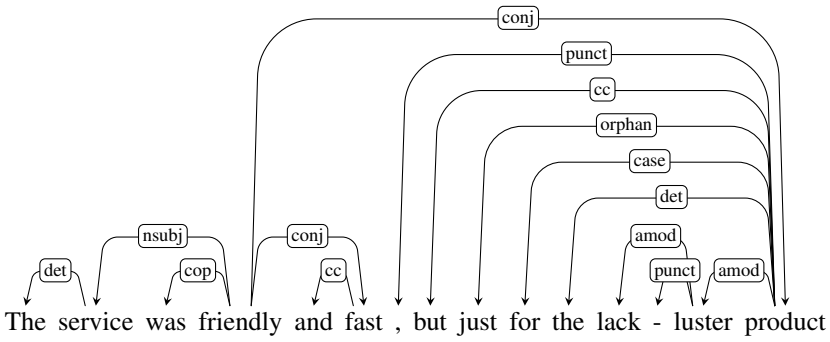


Figure 9: The sentence from Figure 8 after Type 2 conversion.

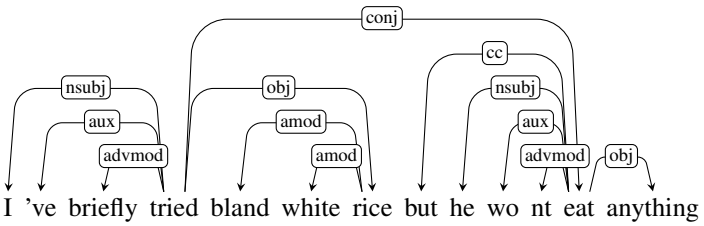


Figure 10: An example of a matched sentence before conversion, problematic case.

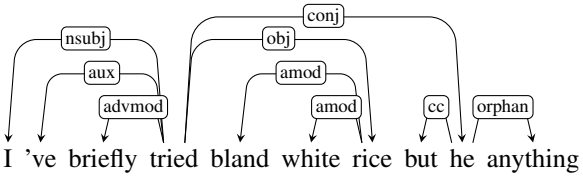


Figure 11: The sentence from Figure 10 after conversion.

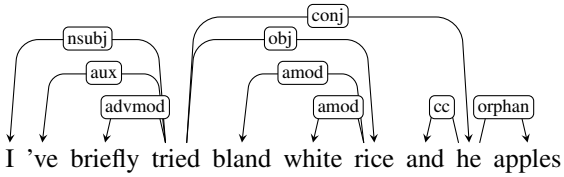


Figure 12: The sentence from Figure 10 after manual correction.

5.4. Language-specific Rules

It is not possible to avoid language-specific rules, although some of them contradict each other. For instance, Finnish allows elliptic sentences with omitted subjects in both clauses and remaining objects (Figure 16, Figure 17), which hardly can be found in English. We thus maintain separate processing pipelines for each language.

An example of an English-specific rule is substitution of personal pronouns. The original sentence typically describes a sequence of actions performed by one actor. By

C2L2	orphan-conj 23.0% 327 h.error: 85.63% 280 orphan-nmod 15.05% 214 h.error: 42.52% 91 orphan-obl 6.12% 87 h.error: 62.07% 54 orphan-advmod 6.05% 86 h.error: 72.09% 62 conj-orphan 5.91% 84 h.error: 61.9% 52
darc	orphan-conj 14.5% 267 h.error: 76.4% 204 orphan-nmod 12.6% 232 h.error: 47.84% 111 conj-orphan 7.98% 147 h.error: 78.23% 115 orphan-obl 5.75% 106 h.error: 55.66% 59 orphan-advmod 5.05% 93 h.error: 60.22% 56
HIT-SCIR	orphan-conj 16.05% 266 h.error: 79.32% 211 orphan-nmod 10.38% 172 h.error: 51.74% 89 conj-orphan 8.99% 149 h.error: 69.13% 103 orphan-obl 6.46% 107 h.error: 68.22% 73 orphan-advmod 4.59% 76 h.error: 80.26% 61
IMS	orphan-conj 22.92% 328 h.error: 81.4% 267 orphan-nmod 14.12% 202 h.error: 43.56% 88 orphan-obl 6.92% 99 h.error: 56.57% 56 orphan-advmod 6.5% 93 h.error: 67.74% 63 conj-orphan 5.87% 84 h.error: 69.05% 58
Koc-University	orphan-conj 20.5% 343 h.error: 79.01% 271 orphan-nmod 12.67% 212 h.error: 52.83% 112 conj-orphan 6.16% 103 h.error: 69.9% 72 orphan-obl 5.5% 92 h.error: 66.3% 61 orphan-advmod 4.72% 79 h.error: 70.89% 56
LATTICE	orphan-conj 17.24% 300 h.error: 82.0% 246 orphan-nmod 13.33% 232 h.error: 49.57% 115 conj-orphan 7.93% 138 h.error: 68.84% 95 orphan-obl 6.03% 105 h.error: 63.81% 67 orphan-advmod 5.11% 89 h.error: 67.42% 60
NAIST-SATO	orphan-conj 17.23% 257 h.error: 82.1% 211 orphan-nmod 11.73% 175 h.error: 45.71% 80 conj-orphan 9.72% 145 h.error: 56.55% 82 orphan-obl 6.7% 100 h.error: 67.0% 67 orphan-advmod 4.83% 72 h.error: 63.89% 46
Orange-Deskin	orphan-conj 15.14% 262 h.error: 71.37% 187 conj-orphan 10.34% 179 h.error: 69.27% 124 orphan-nmod 9.76% 169 h.error: 45.56% 77 orphan-obl 5.6% 97 h.error: 65.98% 64 orphan-advmod 4.68% 81 h.error: 66.67% 54
Stanford	orphan-conj 17.71% 247 h.error: 85.43% 211 orphan-nmod 12.19% 170 h.error: 45.88% 78 conj-orphan 10.9% 152 h.error: 61.84% 94 orphan-obl 5.3% 74 h.error: 64.86% 48 orphan-advmod 5.23% 73 h.error: 65.75% 48
TurkuNLP	orphan-conj 19.96% 329 h.error: 74.77% 246 orphan-nmod 12.38% 204 h.error: 47.55% 97 conj-orphan 8.56% 141 h.error: 73.05% 103 orphan-obl 6.37% 105 h.error: 67.62% 71 orphan-advmod 5.95% 98 h.error: 63.27% 62
UFAL-UDPipe-1-2	orphan-conj 17.12% 288 h.error: 81.94% 236 orphan-nmod 12.31% 207 h.error: 44.93% 93 conj-orphan 8.62% 145 h.error: 73.79% 107 orphan-obl 5.77% 97 h.error: 59.79% 58 orphan-advmod 4.88% 82 h.error: 56.1% 46
UParse	orphan-conj 14.96% 243 h.error: 81.48% 198 orphan-nmod 12.5% 203 h.error: 50.74% 103 conj-orphan 9.11% 148 h.error: 59.46% 88 orphan-obl 5.91% 96 h.error: 69.79% 67 orphan-advmod 4.37% 71 h.error: 59.15% 42

Table 2: Erroneously predicted or missed `orphan` labels and their frequencies

substituting the second subject with a different pronoun, we create an elliptical sentence where two actors perform presumably the same action but with different patients (Figure 6, Figure 7). Another example would be a rule for copular constructions. If the main clause contains a copula, the sentence must be converted into Type 2 structure (Figure 8,

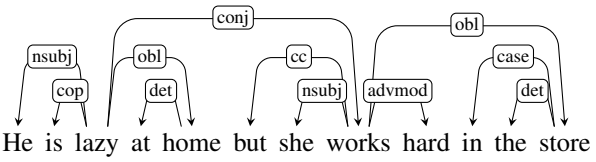


Figure 13: An example of an English sentence with copula before conversion.

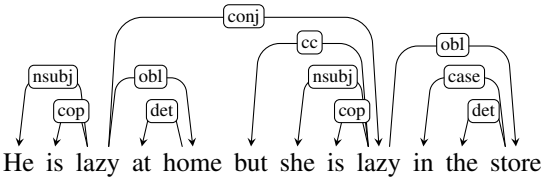


Figure 14: The sentence from Figure 13, copula reconstruction

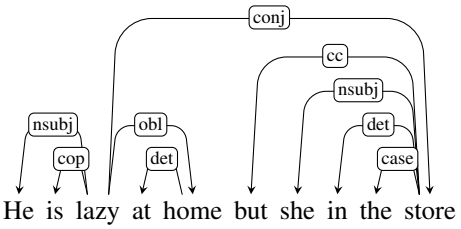


Figure 15: The sentence from Figure 13, correct analysis

Figure 9). Consider an example at Figure 13. After deletion of the verb, the structure would have to be reconstructed as there was a copula (Figure 14). And in this case, if the copula is not repeated, the second clause is interpreted just as a clause with a nonverbal predicate (Figure 15).

In all languages it is important to check that the deleted predicate has at least two arguments or adjuncts whose type (e.g., "nsubj", "obj", "obl:agent") matches dependents of the first predicate; otherwise no gapping can occur. If the dependent is a prepositional phrase, the meaning of the prepositions must be compatible, too (note that it does not necessarily mean that the prepositions are identical). Furthermore, in Czech and Finnish the morphological case of the nouns is important. (English does not have cases, with the exception of personal pronouns.)

Obviously, the rules described above filter out a substantial amount of sentences. Therefore, we currently do not check prepositions in English; instead, we manually fix sentences where prepositions are not compatible.

A good Czech example from the parsed web corpus is in Figures 18 and 19. Both verbs take a directional adjunct with the preposition *do* "to" and a genitive noun, which makes the gapped sentence sound very natural.

6. Results

We provide artificial ellipsis treebanks for three languages, Czech, English and Finnish, using our processing pipeline explained in previous sections. Furthermore, the data for English and Finnish is manually checked and fixed to be grammatical and naturally-sound using UD Annotatrix (Tyers et al., 2017) annotation tool. This further ensures the

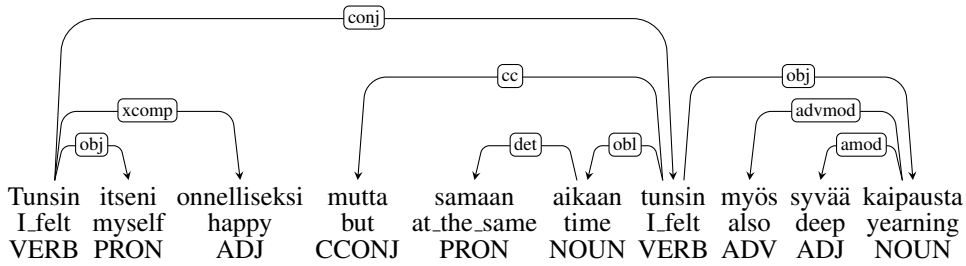


Figure 16: An example of a Finnish sentence automatically identified in the parsed Finnish web corpus.

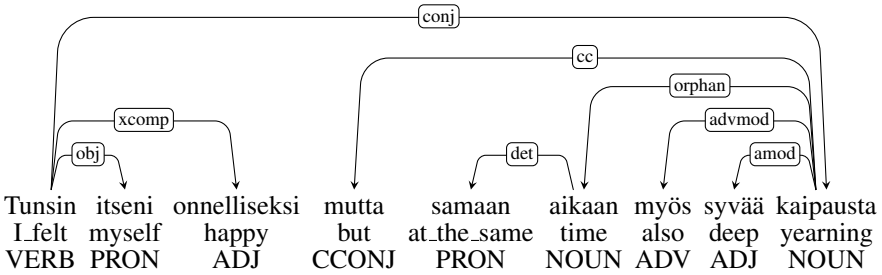


Figure 17: The Finnish sentence from Figure 16 after conversion.

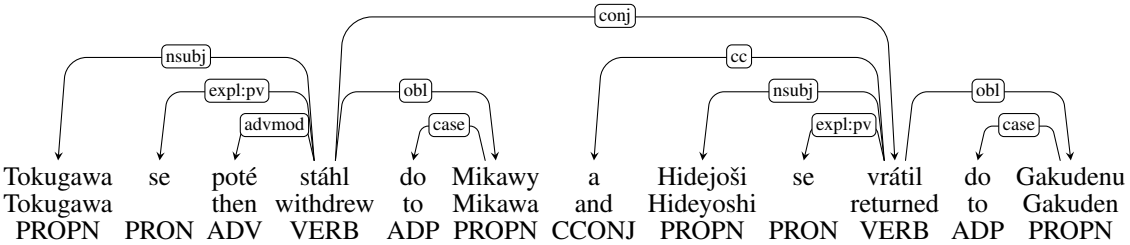


Figure 18: A perfect example automatically identified in the parsed Czech web corpus.

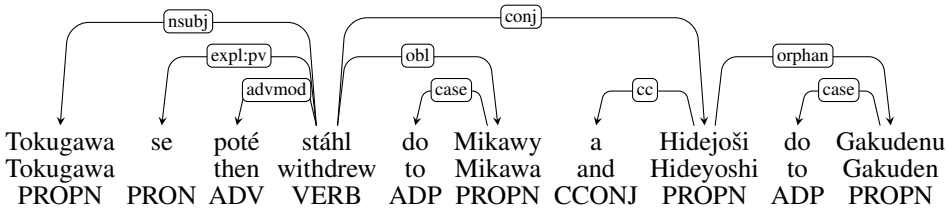


Figure 19: The Czech sentence from Figure 18 after conversion.

good quality of the provided data. Table 3 provides information concerning the sizes of these three datasets.

7. Related Work

The idea of artificial generation or modification of corpora is not new and it has been occasionally applied to various areas of language learning, whenever the studied phenomenon is underrepresented in existing resources. To name just a few: In (van der Plas et al., 2009), creation of an artificial treebank from an existing text treebank helps to overcome domain differences. (Khoshnavataher et al., 2015) artificialy modify text to look like obfuscated plagiarism; the resulting corpus is used to train a plagiarism-detecting system for Persian. And (Gulordava and Merlo, 2016) generate word-order permutations to study the im-

	Initial	Processed	Manual
Czech	1.7M / 102K	13K / 498	NA
Czech web	23M / 2M	37K / 2369	NA
English	408K / 24K	6.8K / 284	3.7K / 183
English web	883K / 89K	6.4K / 422	3.6K / 238
Finnish web	31M / 4.3M	31K / 2442	13K / 1000

Table 3: The size of the data. Initial: the size of the input data, tokens/sentences; Processed: the size of the data after the application of the conversion pipeline, tokens/sentences; Manual: the size of the data after manual correction, tokens/sentences

pact of word order on parsing accuracy in twelve different languages.

8. Conclusion

We have presented experiments that provide a closer look at the current parsers regarding their ability to parse elliptical constructions. We have proposed and described a methodology for creation of artificial treebanks for parsing experiments. The parsing experiments with the artificial data are the natural next step, which we are going to take.

The first version of the artificial elliptic UD treebanks is publicly available in CoNLL-U format via the LINDAT/CLARIN repository at `http://hdl.handle.net/11234/1-2616`.

The artificial data provides a decent amount of ellipsis-like structures. The number of sentences may not look large but it is still several times bigger than naturally occurring elliptical sentences in large treebanks. It will significantly increase the basis that can be used to train ellipsis-aware parsers.

9. Acknowledgements

The work was partially supported by the grant 15-10472S of the Czech Science Foundation (GAČR), by the GA UK grant 794417, Academy of Finland, and Nokia Foundation. The data is available via the LINDAT/CLARIN repository, which is part of a research infrastructure project funded by the Ministry of Education, Youth and Sports of the Czech Republic, Project No. LM2015071. Computational resources were provided by CSC - IT Center for Science, Finland.

10. Bibliographical References

Coppock, E. (2001). Gapping: In defense of deletion. In *Proceedings of the Chicago Linguistics Society*, volume 37, pages 133–148.

Dozat, T., Qi, P., and Manning, C. D. (2017). Stanford's graph-based neural dependency parser at the conll 2017 shared task. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30.

Droganova, K. and Zeman, D. (2017). Elliptic constructions: Spotting patterns in ud treebanks. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, number 135, pages 48–57.

Gulordava, K. and Merlo, P. (2016). Multi-lingual dependency parsing evaluation: a large-scale analysis of word order properties using artificial data. *Transactions of the Association for Computational Linguistics (TACL)*, 4. Presented at ACL 2016.

Johnson, K. (2009). Gapping is not (vp) ellipsis. *Linguistic Inquiry*, 40(2):289–328.

Khoshnavataher, K., Zarrabi, V., Mohtaj, S., and Asghari, H. (2015). Developing monolingual persian corpus for extrinsic plagiarism detection using artificial obfuscation. In *Notebook for PAN at CLEF 2015*.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association.

Popel, M., Žabokrtský, Z., and Vojtek, M. (2017). Udapi: Universal API for universal dependencies. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 96–101.

Schuster, S., Lamm, M., and Manning, C. D. (2017). Gapping constructions in Universal Dependencies v2. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*.

Tyers, F. M., Sheyanova, M., and Washington, J. (2017). Ud annotatrix: An annotation tool for universal dependencies. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 10–17.

van der Plas, L., Henderson, J., and Merlo, P. (2009). Domain adaptation with artificial data for semantic parsing of speech. In *Proceedings of NAACL HLT 2009: Short Papers*, pages 125–128, Boulder, Colorado, USA, June.

Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinková, S., Hajič jr., J., Hlaváčová, J., Kettnerová, V., Urešová, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., de Paiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, Ç., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Fernandez Alcalde, H., Strnadova, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonça, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.

11. Language Resource References

Ginter, F., Hajič, J., Luotolahti, J., Straka, M., and Zeman, D. (2017). *CoNLL 2017 Shared Task - Automatically Annotated Raw Texts and Word Embeddings*. ISLRN `http://hdl.handle.net/11234/1-1989`.

Nivre, J., Agić, Ž., Ahrenberg, L., et al. (2017). *Universal Dependencies 2.1*. ISLRN `http://hdl.handle.net/11234/1-2515`.

Zeman, D., Potthast, M., Straka, M., et al. (2017). *CoNLL 2017 Shared Task System Outputs*. ISLRN `http://hdl.handle.net/11234/1-2424`.

Semi-Automatic Construction of Word-Formation Networks (for Polish and Spanish)

Mateusz Lango[1,2], Magda Ševčíková[2] and Zdeněk Žabokrtský[2]
[1]Poznan University of Technology, Faculty of Computing, Institute of Computing Science
[2]Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
mlango@cs.put.edu.pl,{sevcikova,zabokrtsky}@ufal.mff.cuni.cz

Abstract

The paper presents a semi-automatic method for the construction of derivational networks. The proposed approach applies a sequential pattern mining technique in order to construct useful morphological features in an unsupervised manner. The features take the form of regular expressions and later are used to feed a machine-learned ranking model. The network is constructed by applying resulting model to sort the lists of possible base words and selecting the most probable ones. This approach, besides relatively small training set and a lexicon, does not require any additional language resources such as a list of alternations groups, POS tags etc. The proposed approach is applied to the lexeme sets of two languages, namely Polish and Spanish, which results in the establishment of two novel word-formation networks. Finally, the network constructed for Polish is merged with the derivational connections extracted from the Polish WordNet and those resulting from the derivational rules developed by a linguist, resulting in the biggest word-formation network for that language. The presented approach is general enough to be adopted for other languages.

Keywords: derivation, derivational morphology, Polish, Spanish, lexical network, learning to rank, sequential pattern mining

1. Introduction

Derivational morphology has moved into focus of Natural Language Processing (NLP) only recently. For some languages, we observe a significant research effort in the construction of resources specialized in derivation, e.g. DerivBase (Zeller et al., 2013) for German, Démonette (Hathout and Namer, 2014) for French, DerivBase.Hr (Šnajder, 2014) for Croatian, DeriNet (Ševčíková and Žabokrtský, 2014; Žabokrtský et al., 2016) for Czech, or Word Formation Latin (Litta et al., 2016). However, for many other languages the data resources which provide information about derived words are scarce or even lacking. Unfortunately, the creation of such resources requires a considerable human effort and is highly time-consuming.

In this paper, we propose a method for semi-automatic construction of a derivational network which can be applied to under-resourced languages. The proposed approach requires only two resources: a set of lexemes and a relatively small training set which should contain examples of derived lexemes with their base words.

First, the method is looking for frequent patterns in a given lexeme set in order to automatically detect character-level regularities in the word construction of the language under consideration. The mentioned process is conducted in a completely unsupervised manner and no hand-crafted rules are used. Given those frequent patterns, each lexeme can be described by the presence (or absence) of a given pattern. Such lexeme descriptions serve as feature vectors for machine learning techniques and allow to train a ranking model.

Later, we start the network construction by making an attempt to find a base word for each lexeme. In order to avoid the consideration of all possible parents for each lexeme, only a candidate set of most morphologically similar words is considered. As a measure of morphological similarity, the Proxinette distance (Hathout, 2009; Hathout, 2014) is used. Next, the previously trained ranker is applied to order each candidate set. Finally, for each lexeme the highest ranked candidates are selected but only if the confidence of the ranker is high.

In order to evaluate the proposed approach, the method is applied on sets of Polish and Spanish lexemes, resulting in two new derivational networks for those languages. Even though these languages belong to different families, both have rich inflectional and derivational morphology, with the derivation as the most productive word-formation process. For Polish more than 50 thousand derivational pairs were automatically detected, and above 18 thousand for Spanish.[1] Moreover, the experimental evaluation demonstrates the high precision of constructed networks.

Finally, the Polish network was enriched with additional connections extracted from Polish WordNet (Maziarz et al., 2016) resulting in the Polish Word-Formation Network which contains 261 822 lexemes with more than 192 thousand connections. To the best of our knowledge, it is the biggest language resource of derivational morphology for Polish.

The rest of the paper is organized as follows. In Section 2. we briefly present related work. Section 3. contains a description of the proposed approach. In Section 4., we proceed with the discussion of experimental evaluation and with the analysis of the resulting resources. Finally, in Section 5. we draw conclusions and discuss lines of future research.

2. Related work

Morphology was intensively studied by the NLP community, with the research primarily concentrated on inflectional morphology. However, in recent years researchers

[1]Spanish lexeme set was considerably smaller.

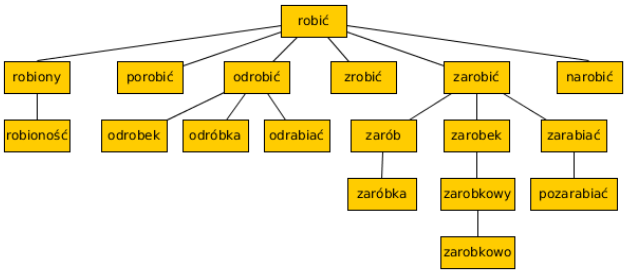


Figure 1: An example of a tree structure in word-formation network.

noticed the potential of derivational morphology to improve the performance in many important areas of NLP, which caused the development of novel language resources which focus on word formation.

One novel type of such resource is word-formation network which represents information about derivational morphology in a form of the graph. In such networks, the derivational relations are represented as directed edges between lexemes. In this work, as in the majority of related works, we consider word-formation networks whose connected components have a tree structure (see an example on Figure 1). This means that derivatives can have only one base word, hence, e.g. compounding is not considered.

Although such networks were created for some languages (see Sect. 1.), there is still a demand for the creation of such resources for many other languages. For instance, there is no word-formation network for two languages which are considered in the present work, Polish and Spanish. For those languages the number of works on automatic detection of derivatives is also quite limited.

Piasecki et al. (2012) propose an approach based on bootstrapping and supervised learning in order to construct derivational rules for Polish. Also, a hand-crafted list of alternation groups is used in order to make the derived words as similar as possible to the base word, allowing for a more effective application of derivational rules. Besides using this additional resource, the approach uses a relatively large training set of 15 718 examples (approx. 10 times bigger than the one used in this work) and deals only with derivation by suffixation or prefixation. Recently, an algorithm for the automatic pairing of perfective and imperfective Polish verb forms has been developed (Kaleta, 2017). The algorithm fully relies on the hand-crafted rules of Polish morphology, hence it is inapplicable for the construction of the network for other languages.

The literature related to Spanish is quite more extensive. Vilares et al. (2001) develop a system for automatic generation of morphological families[2] in order to improve their information retrieval systems. However, the derivational rules are incorporated into the system by a human expert rather than automatically learned. There were also some attempts to automatically discover affixes and suffixes used in Spanish derivations (Urrea, 2000), but no language resources were created. Spanish was also one of the languages studied by Baranes and Sagot (2014) in their language-independent approach. Although their approach is completely unsupervised, it requires POS tagging and, particularly for Spanish, it provides a considerably lower accuracy of extracted relations than for the other languages (73% for Spanish compared with 98% for English and German). In our opinion, it creates an opportunity for a use of supervised approaches which do not require large training sets nor additional handcrafted features or resources.

Sequential pattern mining Sequential pattern mining is one of the most important topics in the area of frequent pattern mining, and in the data mining in general (Han et al., 2007). The problem of sequential pattern mining is the extraction of all frequent subsequences with the support[3] greater than a specified threshold. Informally, a sequence a is a subsequence of the sequence b if one can remove items from the sequence b (without changing the order of them), to finally get the sequence a. Due to the importance of the task, a lot of approaches have been proposed. Among them, SPADE (Zaki, 2001) which bases on breadth-first search and Apriori pruning on the vertical data format. For formal definitions and a detailed review consult e.g. Mabroukeh and Ezeife (2010).

Learning to rank Learning to rank is a widely studied area of machine learning which was originally researched in the context of automatic ranking of web search results in the information retrieval community. However, it proved to be useful in many other areas such as statistical machine translation, see (Watanabe, 2012). The task of learning to rank is the construction of a model which is able to sort new objects according to their degrees of importance. The approaches for machine-learned ranking can be divided into three groups: the pointwise, the pairwise and the listwise approaches. Pointwise methods make use of classification or regression techniques in order to predict a score for each object given the query. An idea of predicting the order of each pair of objects is explored by pairwise methods. Finally, listwise approaches directly optimize metrics defined on a whole list of objects. For a review of those methods see e.g. Liu (2009).

3. Proposed approach

For the construction of the networks for Polish and Spanish, a machine-learned model is constructed first. This allows for building a large part of the word-formation network automatically, using a small training set only. Our focus is rather on the precision of discovered connections than on the coverage, hence a connection is constructed only if its probability exceeds a predefined threshold. For Polish then, a second step is done, namely the network established in the first step is merged with derivational connections extracted from Polish WordNet.

3.1. Machine learning approach: Polish and Spanish

Our machine learning approach can be split into several steps. First, a sequential pattern mining method is used to

[2]In the context of word-formation networks, a morphological family is a connected component of the network.

[3]The support of a subsequence is equal to the number of sequences which contain that subsequence in the database.

construct features describing lexemes in an unsupervised manner. Then, for each lexeme in the lexicon, we construct a list of possible base words using nearest-neighbors search. Finally, a machine-learned ranker is trained and consecutively applied to each candidate list. If the model ranks one position on the list much higher than any other, we create a connection between the lexeme and the selected candidate. In the following paragraphs, we will discuss each step in detail.

As the first stop of our approach, a sequential pattern mining is applied to the lexicon in order to find frequent subsequences. To perform this task we selected the SPADE algorithm (Zaki, 2001) since it is computationally efficient for usual lexicon sizes and its implementation was easily available. Any other algorithm which solves the sequential pattern mining problem could be adopted in a straightforward way. The algorithm treats each word as a sequence of characters and the lexicon is interpreted as a database of them. Hence, the resulting subsequences are in fact lists of characters which often occur in a particular order. The examples of such frequent subsequences in Polish lexicon are {n,i,e}, {o,w,y} and {n,o,ś,ć}. Our hypothesis is that by finding subsequences covered by a lot of words, we will be able to discover useful morphological patterns in the lexeme set. We hope that those patterns will be useful in the feature construction for machine learning techniques.

To unify notation with the following paragraphs, we will represent each frequent subsequence as a regular expression. For example, the aforementioned subsequence {n,i,e} will be further denoted as ^⋆n⋆i⋆e⋆$ where ^ and $ mark the beginning and the end of the word, and ⋆ represents any string (including an empty one). At this point one faces two problems: first, the extracted patterns are too general and second, the number of frequent subsequences is large. Hence, we proceed with a procedure whose goal is to make the patterns more specific but also more meaningful. Later, a method for pruning the set of frequent subsequences is introduced.

In order to make patterns more specific, we apply a greedy approach which iteratively tries to delete one of the symbols of any string (⋆) from the pattern. If the deletion of that symbol results in a small decrease in the support (less than a threshold provided by the user), the newly created regular expression is accepted. The execution of this procedure results in more specific patterns with some of them having a linguistic interpretation. For example, ^⋆n⋆i⋆e⋆$ is replaced by ^nie⋆$ which is a prefix used for the creation of negated forms in Polish, e.g. *dobry* (good) and *niedobry* (bad); ^⋆o⋆ś⋆ć⋆$ is converted to ^⋆ość$ which is a common suffix for Polish nouns, e.g. *męski* (manly) and *męskość* (manhood); *żwawo* (briskly) and *żwawość* (briskness). Since many other methods for automatic discovery of suffixes and prefixes has been proposed in the literature, it is important to note that our approach is able to detect more complex patterns than affixation only. For example patterns like ^nie⋆ość$ or ^⋆cz⋆ność$ are also constructed.

Next, one must deal with a high number of patterns generated. We have observed that some of the patterns match almost the same lexemes. For example, ^⋆cz⋆noś⋆$ and ^⋆cz⋆ność$ have approximately the same support and cover the same lexemes, so keeping both of them seems to be redundant. In order to detect such pairs of redundant patterns, we perform a specific correlation analysis. First, we describe each lexeme in the lexicon by a binary feature vector. Each previously created regular expression is converted into one feature which takes 1 when the pattern matches the lexeme, and 0 otherwise. Having such representation of the lexicon, we are able to measure the association between patterns by calculating the phi coefficient (see e.g. Kotz et al., 2006). We identify a pair of patterns as a redundant one if its phi coefficient is greater than 95 %. Then, we shrink the number of patterns by selecting the most specific pattern from each indicated pair. This results in a considerably smaller set of patterns.

The aforementioned representation of lexemes as binary feature vectors enable us to use machine learning techniques for the construction of word-formation network. Since the network has a tree-like structure, we construct it by finding a base word for each lexeme. We approach this problem by sorting the list of possible base words for a lexeme from the most plausible ones to the least probable ones. Because sorting a whole lexicon for each lexeme is infeasible, we restrict the list of possible base words to the 100 most similar words according to Proxinette measure (Hathout, 2009; Hathout, 2014). Such constructed candidate list is sorted by a ranker, which is previously trained on a relatively small training set provided by a linguist. A connection in the network is established when the difference between the rank of the first and second element on the sorted list exceeds a threshold provided by the user.

3.2. Enriching the Polish network

3.2.1. Extraction of derivational connections from WordNet

The connections constructed in the Polish network by the machine learning method, the connections extracted from Polish WordNet (Maziarz et al., 2016) are added. Polish WordNet contains many relations which store information about derived words such as "feminity" which links masculine nouns with its feminine counterparts, "inhabitant" which connects geographical names with the name of their inhabitants, "aspectuality" which relates verbs of different aspects and many others. A list of the main relations related to derivation, together with descriptions can be found in (Piasecki et al., 2012). In order to merge the automatically constructed network with the connections from the WordNet, we iteratively analyze each lexeme for which a base word was not discovered by the machine learning approach. For each such lexeme, we try to find a base word using one of the 53 relations which were extracted from the WordNet. Additionally, the direction of some of the relations was reversed, in order to obtain a coherent network structure.

3.2.2. Adding connections by derivational rules

Although our machine learning approach is able to detect pairs of words which are derived in a fairly complex way, we discovered that many new connections can be added to the network by following some simple handcrafted rules.

A language expert created 20 derivational rules for Polish which are most productive. The derivational rules are given in the form of regular expressions and express suffix/prefix addition or substitution. In order to improve rule's performance, the base word indicated by the rules is accepted only if it already exists in a lemma set.

3.2.3. Elimination of cycles

One disadvantage of constructing a network in an iterative way is the lack of any prevention from creating a cycle. In our approach, a network is constructed by searching a parent for each lemma. This ensures that every node in a graph has at most one outgoing edge and causes that many of the graph's components will have the desired tree structure. Nevertheless, there is no guarantee that a cycle in a graph will not be created (e.g. $A \rightarrow B \rightarrow C \rightarrow A$). To handle such situation, we use a simple heuristic to eliminate graph's cycles. Our heuristic relies on the rather naive assumption that the derivatives are usually longer than the base word since they are often created by adding to it a suffix or a prefix. Hence, we eliminate cycles by iterating over them and removing the first connection between a shorter child and a longer parent. If all words in a cycle have the same length, we drop a random connection.

4. Experimental evaluation

4.1. Machine learning approach

As the base for the creation of the Polish Word-Formation Network, we have used the Grammatical Dictionary of Polish (Saloni et al., 2017), which is a comprehensive lexicon of the Polish language, covering more than 261 thousand lexemes. This dictionary is quite popular in the Polish NLP community as it was used in the creation of many resources and tools such as Morfeusz morphological analyzer (Woliński, 2006) or the Great Dictionary of Polish (Żmigrodzki, 2011).

Using this lexeme set, we created a training set which consists of 1500 pairs of base words with their derivatives. Polish native speakers, who created the training set, were asked to provide examples of as many different derivation schemas as possible. The construction of the training set took approximately 12 man-hours.

In the implementation of our machine learning approach, we have used the SPMF data mining library (Fournier-Viger et al., 2016) for the extraction of frequent subsequences. The implementation of the ranker based on Gradient Boosting Decision Trees was taken from XGBoost machine learning library (Chen and Guestrin, 2016). The ensemble of 100 decision trees was used with the maximum depth of a tree set to 40.

The application of the SPADE algorithm with the minimal support set to the 1% of the lexicon size resulted in roughly 27 thousand frequent subsequences. However, our filtering technique based on the phi coefficient limited the set of patterns to 13 441 regular expressions. Using that set of expressions, we created a feature vector together with two additional features: the length of the common prefix and the length of the common suffix. The training set for a ranker consist of automatically constructed groups containing a lexeme together with 100 candidates. In each group, the rank of the correct base word is set to 1, whereas the rank of the rest of candidates is set to 0.

We evaluated our approach using 5-fold cross-validation and we obtained the accuracy of 82.33% without applying any threshold on the confidence of the ranker. However, since we prefer precision to coverage, we have chosen a threshold which allowed us to obtain 98.8% of precision with the recall of 38.2%. By applying this thresholded model to the set of lexemes, we were able to create more than 53.5 thousand links in the network.

Since our dataset does not contain negative examples (all of the words are derivatives), we decided to perform a manual verification of the precision of discovered connections. We created a random sample of 200 connections and manually checked each of them. Only 6 of them were incorrect, so we estimate the precision of created connections to 97%. Then, we estimated the recall to 26.5% by sampling 200 lexemes and verifying if their parent exists in the network.

Encouraged by the high precision of constructed network, we decided to augment the training set with the connections from the network. In this way, we have around 55 thousands lexeme pairs in the training set without any additional effort of language experts, although the training set has become somewhat noisy. By applying our approach to this larger set, we have obtained almost 75 thousand connections. The manual evaluation on a random sample of 200 connections yielded 95% of precision and 34% of recall.

Since our machine learning approach is general enough to be adopted for other languages, we have also evaluated its performance on the set of 159 035 Spanish lexemes taken from Leffe (Molinero et al., 2009) lexicon. Using the training set of 1026 examples, we obtained 98.1% precision with the recall of 27.7%. The model was able to discover 18.5 thousand connections with the manually evaluated precision of 85% and recall of 44%. The precision for Spanish is considerably lower than for Polish due to a large amount of French proper names in the lexicon which were not taken into account by linguists while creating the training set. The precision without taking them into account is much closer to the one for Polish (94.9%).

Language	Method	Precision	Recall
Polish	Rules	90.0%	66.7%
	ML	95.0%	34.0%
Spanish	Rules	84.0%	59.0%
	ML	94.9%	44.0%

Table 1: A comparison of precision and recall for machine learning and rule-based approach

4.2. Comparison against the rule-based approach

We compared the performance of our machine learning approach with substitution rules provided by a human expert. We asked a linguist to provide us with 20 highly productive and reliable derivational rules for each language. The results are presented in Table 1.

For both languages, the recalls obtained by hand-crafted

Step	# of conn.	Precision	Recall
Machine Learning	53 487	97.0%	26.5%
Machine Learning (retraining)	74 985	95.0%	34.0%
Merge with WordNet	110 553	94.5%	47.0%
Derivational rules	192 289	95.0%	72.0%

Table 2: The number of connections, precision and recall of the Polish Word-Formation Network evaluated after each step of the construction.

rules are considerably higher than those for machine learning approach. However, our method is significantly better in terms of precision. Since the provided rules were the very productive ones, we would like to emphasize that further development of a higher number of hand-crafted rules will result in rather small increases of the recall. Furthermore, the impact of applying the rules in a particular order will significantly grow with their number. This will cause that substitution rules will have to be designed with special caution and be thoroughly tested. Conversely, one can expect the quality of machine learning approach to improve together with the addition of more training examples. Also, a particular rule typically handles only one type of derivation whereas machine learning approach it is able to connect lexemes derived in a plenty of different ways.

4.3. Construction of the Polish Word-Formation Network

In order to create a bigger language resource, we decided to merge our automatically detected connections with other available resources.

First, we extracted derivational connection from *Słowosieć*, the Polish WordNet. By applying these connections to our network's lexemes, we were able to construct about 52 thousand links. Finally, merging the automatically discovered connections together with links constructed on the basis of the information from the WordNet resulted in a network of more than 110.5 thousand connections.

The manual verification of precision and recall, performed as was previously described in Section 4.1., yielded 94.5% of precision and 47% of recall. The addition of connections from the Polish WordNet, introduced some new errors mainly due to the incorrect direction of the link or due to the omission of lexemes in the chain of derivation e.g. *poatomowy* is directly connected with *atom* whereas we would expect *atom* (atom) →*atomowy* (atomic) →*poatomowy* (property of the result of a nuclear explosion).

Finally, a set of 20 highly-productive hand-constructed derivational rules was applied to the lemma set. The resulting network has 192 289 connections and, to the best of our knowledge, it is the biggest word-formation network for Polish language. The manual verification of the resource yielded 95% of precision and 72% of recall. This demonstrate that the resource is reliable and has significant coverage of Polish derivations.

Both the Polish Word-Formation network and the data for Spanish are available at `ufal.mff.cuni.cz/derinet`.

4.4. Visualization of word-formation networks

To better evaluate the created resource, we decided to visualize some parts of the Polish Word-Formation Network using a visualization tool developed by Vidra and Žabokrtský (2017). In Figure 2 one can see the derivation tree for a proper noun *Szczecin* (a Polish city). The derived adjective (*szczeciński*), the name of its inhabitants (*szczecinianin*) and even a name of another smaller city (*Szczecinek*) are correctly connected in a network, despite the fact that e.g. the word-formation process of *szczeciński* is potentially difficult because of occurring alternation of *n* to *ń*. One can also observe that the tree has multiple layers with further derivatives e.g. *podszczeciński* (property of Szczecin's surrounding area). Furthermore, a derivation tree for *herbata* (tea) in Figure 3 also seems correct, containing derived adjectives (e.g. *herbaciany*), nouns (e.g. *herbaciarnia* tea shop), negated forms (e.g. *nieherbaciany*) and diminutives (*herbatka*). One can notice the lack of adverbs, however they are absent in our lemma set for this particular case.

The presented approach can also handle some untypical cases. For example, *ponauczać* is a verb which does not have an imperfective form (see Figure 4). Even though a blind application of morphological rules would result with *ponauczyć* (e.g. *nauczyć* and *nauczać* are correctly connected), this verb is correctly connected with another perfective form *nauczać*.

However, on visualizations also some errors in the network structure can be found. For example, in Figures 5 and 6 we present the derivation trees for *karta* (sheet) and *karty* (card game), respectively. First of all, we would rather expect these two trees to be mutually connected. The lack of the connection between *karta* and *karty* causes the absence of a whole derivational subtree in the derivations of *karta*. Second, the words *Djakarta*/*Dzakarta* (Jakarta) are definitely not derived from *karta* and *gokartowość* (noun from go-cart) is not derived from *kartowość*. Such errors related to the words adopted from other languages are quite frequent. Another example of such error can be found in Figure 7 where *pastoforium* is derived from Greek.

We have also visualized some parts of the Spanish Word-Formation Network. On Figure 8 the derivation tree for the verb *ilustrar* (to illustrate) is presented. Similarly to the visualizations of the Polish network, one can see that the lexeme is correctly linked to many derivatives. The derived adjectives like *ilustrativo* illustrative or *ilustrado* illustrated as well as derived nouns (e.g. *ilustración* illustration) are properly connected in the network. However, as a result of a low recall, many lexemes have too small derivation trees. A good example of this is the derivation tree for the verb *hacer* (to do) which is presented on Figure 9. Only the noun *hecedor* (maker) is connected with this verb and the absence of many other lexemes is rather evident e.g. *rehacer* (to redo) or *deshacer* (to undo) are lacking.

5. Conclusion and future research

In the present paper, a new semi-automatic approach for the construction of word-formation networks is presented. In particular, it applies sequential pattern mining to construct useful morphological features. To the best of our knowledge, no one has used these techniques in that context.



Figure 2: The noun *Szczecin* (the capital of West Pomeranian Voivodeship) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.



Figure 3: The noun *herbata* (tea) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.

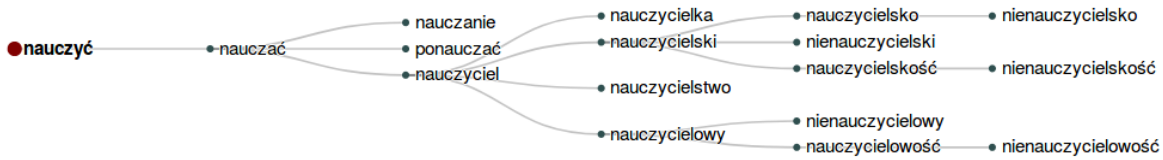


Figure 4: The verb *nauczyć* (to learn *perf*) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.



Figure 5: The noun *karta* (sheet) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.

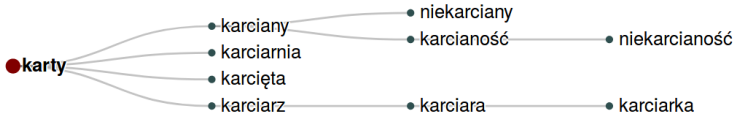


Figure 6: The noun *karty* (card game) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.



Figure 7: The noun *pastor* (pastor) and derivationally related lexemes (displayed as a tree structure) in the Polish Word-Formation Network.

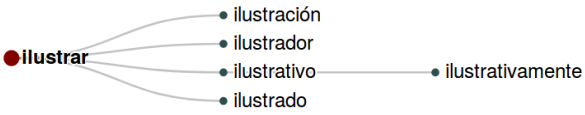


Figure 8: The verb *ilustrar* (to illustrate) and derivationally related lexemes (displayed as a tree structure) in the Spanish Word-Formation Network.



Figure 9: The verb *hacer* (to do) and derivationally related lexemes (displayed as a tree structure) in the Spanish Word-Formation Network.

Moreover, the approach was successfully evaluated on one Slavic and one Romance language, namely Polish and Spanish. For both languages, the newly proposed method discovered plenty of true connections between base words and their derivatives. The resources created by the method are characterized by a high precision, and their creation does not require large human effort.

Finally, this paper introduces the Polish Word-Formation Network which is the result of merging Polish WordNet with automatically discovered connections by our machine learning approach. The network is constructed over a large lexeme set and is characterized by high precision. Furthermore, it is also the biggest free language resource about Polish derivations.

However, the coverage of the created resources still needs to be improved. As a future research, the active learning framework (Settles, 2010) could be incorporated into our approach. The active learning methods allow for the iterative construction of machine learning models with the participation of human experts. In such approaches, training examples which should be annotated are chosen by the algorithm itself with the ultimate goal of constructing an accurate model using as few examples as possible. We hope that such extension of our work would lead to the significant improvement in coverage of constructed networks without significantly increasing the effort of human experts.

Currently, we explore the possibility of using a cross-lingual transfer to the fully automatic construction of word-formation networks. As we mentioned in the introduction, derivational resources have been already developed for a limited number of languages. This creates the possibility of employing those resources to the construction of word-formation networks for related languages. For instance, the Czech DeriNet (Vidra et al., 2017) may be used to extend the proposed Polish network. Each connection from the Czech network could be translated into Polish using one of the available dictionaries such as Treq (Škrabal and Martin, 2017). Then, some notion of morphological similarity should be adopted in order to eliminate incorrect connections. In the context of the present work, the constructed candidate sets may be used to check if the indicated base word is morphologically related. The preliminary experiments which we have run so far are quite promising. The described approach was able to find over 40 thousand connections but with rather poor precision. Roughly 40% of the discovered connections were correct, however, one-third of errors could be simply fixed by reversing the direction of relation. We believe that there are many possibilities to improve this result e.g. by taking the probability of translation into account.

Another issue of the current approach is that it is limited to derivation only. Hence, there is a need of developing novel methods for the automatic discovery of other word-formation processes, such as compounding. Additionally, some effort to design the representation of such enhanced resources is also needed.

Acknowledgements

This work was supported by the Grant No. GA16-18177S of the Grant Agency of the Czech Republic. It has been using language resources developed, stored, and distributed by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071).

Mateusz Lango acknowledges also the support from the Institute of Computing Science Statutory Funds and from the internship program of the Institute of Formal and Applied Linguistics.

6. Bibliographical References

Baranes, M. and Sagot, B. (2014). A Language-independent Approach to Extracting Derivational Relations from an Inflectional Lexicon. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 2793–2799, Reykjavik, Iceland, May.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM*

SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, pages 785–794, New York, NY, USA. ACM.

Fournier-Viger, P., Lin, C., Gomariz, A., Gueniche, T., Soltani, A., Deng, Z., and Lam, H. T. (2016). The SPMF Open-Source Data Mining Library Version 2. In *Proc. 19th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD 2016) Part III*.

Han, J., Cheng, H., Xin, D., and Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15(1):55–86.

Hathout, N. and Namer, F. (2014). Démonette, a French derivational morpho-semantic network. *Linguistic Issues in Language Technology*, 11(5):125–168.

Hathout, N. (2009). Acquisition of Morphological Families and Derivational Series from a Machine Readable Dictionary. In Fabio Montermini, et al., editors, *Selected Proceedings of the 6th Décembrettes*, Cascadilla Proceedings Project, pages 166–180, Bordeaux, France.

Hathout, N. (2014). Phonotactics in morphological similarity metrics. *Language Sciences*, 46:71 – 83. Theoretical and empirical approaches to phonotactics and morphonotactics.

Kaleta, Z. (2017). Automatic Pairing of Perfective and Imperfective Verbs in Polish. In *Proceedings of the Eight Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*.

Kotz, S., Read, C. B., Balakrishnan, N., and Vidakovic, B. (2006). *Encyclopedia of statistical sciences*. Wiley, 2nd edition.

Litta, E., Passarotti, M., and Culy, C. (2016). Formatio formosa est. Building a Word Formation Based Lexicon for Latin. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it 2016)*, pages 185–189, Naples.

Liu, T.-Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331.

Mabroukeh, N. R. and Ezeife, C. I. (2010). A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys (CSUR)*, 43(1):3.

Piasecki, M., Ramocki, R., and Maziarz, M. (2012). Recognition of Polish Derivational Relations Based on Supervised Learning Scheme. In *LREC*, pages 916–922.

Settles, B. (2010). Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11.

Ševčíková, M. and Žabokrtský, Z. (2014). Word-Formation Network for Czech. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 1087–1093, Reykjavik, Iceland, May.

Šnajder, J. (2014). DerivBase.hr: A High-Coverage Derivational Morphology Resource for Croatian. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 3371–3377, Reykjavik, Iceland, May.

Urrea, A. M. (2000). Automatic discovery of affixes by means of a corpus: A catalog of Spanish affixes. *Journal of quantitative linguistics*, 7(2):97–114.

Vidra, J. and Žabokrtský, Z. (2017). Online software components for accessing derivational networks. In *Proceedings of the Workshop on Resources and Tools for Derivational Morphology*, pages 129–139.

Vilares, J., Cabrero, D., and Alonso, M. A. (2001). Applying productive derivational morphology to term indexing of Spanish texts. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 336–348. Springer.

Watanabe, T. (2012). Optimized online rank learning for machine translation. In *Proceedings of the 2012 conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies*, pages 253–262. Association for Computational Linguistics.

Woliński, M. (2006). Morfeusz - a practical tool for the morphological analysis of Polish. In *Intelligent information processing and web mining*, pages 511–520. Springer.

Zaki, M. J. (2001). SPADE: An efficient algorithm for mining frequent sequences. *Machine learning*, 42(1):31–60.

Zeller, B., Šnajder, J., and Padó, S. (2013). DErivBase: Inducing and evaluating a derivational morphology resource for German. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1201–1211.

Żmigrodzki, P. (2011). Polish Academy of Sciences Great Dictionary of Polish: history, presence, prospects. *Studies in Polish Linguistics*, 6(1):7–26.

Žabokrtský, Z., Ševčíková, M., Straka, M., Vidra, J., and Limburská, A. (2016). Merging Data Resources for Inflectional and Derivational Morphology in Czech. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1307–1314, Portoroz, Slovenia.

7. Language Resource References

Maziarz, Marek and Piasecki, Maciej and Szpakowicz, Stanisław. (2016). *plWordNet*. Wrocław University of Technology, 3.0.

Molinero, M., Sagot, B., and Nicolas, L. (2009). A morphological and syntactic wide-coverage lexicon for Spanish: The Leffe. In *RANLP 2009-Recent Advances in Natural Language Processing*.

Saloni, Zygmunt and Gruszczyński, Włodzimierz and Woliński, Marcin and Wlosz, Robert and Skowrńska, Danuta. (2017). *Slownik gramatyczny języka polskiego*. University of Warmia and Mazury.

Vidra, Jonáš and Žabokrtský, Zdeněk and Ševčíková, Magda and Kalužová, Adéla and Mediankin, Nikita and Straka, Milan. (2017). *DeriNet 1.5*. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. http://hdl.handle.net/11234/1-2422.

Škrabal, M. and Martin, V. (2017). Databáze překladových ekvivalentů Treq. *Časopis pro moderní filologii*, 99(2):245–260.

A multilingual collection of CoNLL-U-compatible morphological lexicons

Benoît Sagot
Inria
2 rue Simone Iff, CS 42112, 75589 Paris Cedex 12, France
benoit.sagot@inria.fr

Abstract

We introduce UDLexicons, a multilingual collection of morphological lexicons that follow the guidelines and format of the *Universal Dependencies* initiative. We describe the three approaches we use to create 53 morphological lexicons covering 38 languages, based on existing resources. These lexicons, which are freely available, have already proven useful for improving part-of-speech tagging accuracy in state-of-the-art architectures.

Keywords: Morphological Lexicons, Universal Dependencies, Freely Available Language Resources

1. Introduction

Morphological information belongs to the most fundamental types of linguistic knowledge. It is often either encoded into morphological analysers or gathered in the form of morphological lexicons. Such lexicons, which constitute the focus of this paper, are collections of lexical entries that typically associate a wordform with a part-of-speech (or morphosyntactic category), morphological features (such as gender, tense, etc.) and a lemma. Beyond direct lexicon lookup, used in virtually all types of natural language processing applications and computational linguistic studies, morphological lexicons have been shown to significantly improve tasks such as part-of-speech tagging and parsing.

There is currently no universally accepted way to encode morphological lexical information. Past multilingual projects such as MULTEXT/MULTEXT-East (Ide and Véronis, 1994; Erjavec, 2010) have resulted in the publication of morphological lexicons for a number of languages based on the same set of categories and morphosyntactic features, but they are still limited in scope.

Yet another type of language resource embeds morphological lexical information, namely treebanks. This type of resource has recently seen the emergence of a *de facto* trans-lingual standard and the publication of an increasing number of treebanks for numerous languages following a universal set of guidelines, encoded in the CoNLL-U format and gathered under the name *Universal Dependencies* (hereafter UD).[1] This treebank collection (Nivre et al., 2016; Nivre et al., 2017) follows several previous initiatives, such as the proposal of a universal part-of-speech tagset (Petrov et al., 2012) and the multilingual datasets released in the context of several shared tasks and projects (Buchholz and Marsi, 2006; Nivre et al., 2007; Zeman et al., 2012; Seddah et al., 2013).

The UD initiative has therefore allowed a simpler, unified access to treebank resources, giving a new impetus to research in topics such as multilingual and cross-lingual tokenisation, part-of-speech tagging, dependency parsing and quantitative linguistics. It is therefore important for morphological lexicons, another major source of linguistic information for such tasks, to also be available for many languages following a universal set of guidelines. The obvious choice would be to make use of the UD guidelines themselves.

We have therefore developed a multilingual collection of morphological lexicons that follow the UD guidelines regarding part-of-speech and morphological features. We used three main sources of lexical information:

- In the context of the CoNLL 2017 UD morphological and syntactic analysis shared task (Zeman et al., 2017) based on UD treebank data, we used lexical information available in the Apertium[2] and Giellatekno[3] projects. This information, which consists of morphological lexicons and analysers, allowed us to provide additional features to our part-of-speech tagging architecture, with high-accuracy results (Villemonte de La Clergerie et al., 2017). In some cases, information from Apertium or Giellatekno converted into the UD format was complemented with information extracted from the training sections of the UD treebanks (v2.0). We also developed a simple, unsupervised yet original algorithm for transferring morphological lexical information from a resourced language to a closely-related one not covered by Apertium and Giellatekno.
- We converted into the UD format a variety of freely available lexicons, among which the lexicons developed in the Alexina framework (Sagot, 2010).[4]
- We used the UD treebanks themselves (v2.0) in order to complete our lexicons with two types of information: (i) multi-word tokens and (ii) entries for categories (UPOS) not covered by the lexical sources mentioned above.

Our contribution thereby lies in a collection of 53 UD-compatible lexicons covering 38 distinct languages and the methods used to create this collection.[5] They are encoded

[1] http://www.universaldependencies.org

[2] https://svn.code.sf.net/p/apertium/svn/languages

[3] https://victorio.uit.no/langtech/trunk/langs

[4] These resources were not allowed in the CoNLL 2017 UD parsing shared task and were not used in this context.

[5] All resulting lexicons are available as free resources via the following website, together with their respective licences: http://pauillac.inria.fr/ sagot/udlexicons.html.

in the CoNLL-UL format, an extension of the CoNLL-U format introduced by More et al. (2018) aimed at representing morphological information, in particular the output of tokenisation and morphological analysis tools, even when they are non-deterministic.

In the remainder of this paper, we first describe how we converted Apertium and Giellatekno morphological analysers and lexicons into CoNLL-UL morphological lexicons. We also briefly sketch our cross-lingual transfer algorithm aimed at building lexicons for poorly resourced languages. We then briefly explain how we converted other existing morphological lexicons to CoNLL-UL. Next we provide a summary of the lexicons we obtained and the languages covered. In the conclusion, we mention a few results on the use of some of these lexicons in part-of-speech tagging experiments, using both statistical and neural approaches.

2. CoNLL-UL lexicon creation

2.1. Creation of CoNLL-UL lexicons from Apertium and Giellatekno resources

The Apertium and Giellatekno projects publish freely available tools and resources for a growing number of languages. These projects rely on morphological analysers, which, in turn, are primarily based either on morphological lexicons or on finite-state morphological grammars. Each language makes use of its own guidelines regarding the inventory of categories and the detailed definition of morphological features and feature values. Yet many categories, features and feature values are used consistently across languages. This allowed us to manually develop a single conversion script that interprets Apertium and Giellatekno categories and morphological features and rewrites them in terms of the UD guidelines, using the Universal Part-Of-Speech (hereafter UPOS)[6] and the Universal morphological Feature (UFEAT)[7] inventories.

Before this conversion could take place, morphological lexical information had to be extracted from the Apertium or Giellatekno resources. Depending on the type of resource available for a given language, we adopted one of the two following strategies:

- Direct extraction and reformatting of the monolingual morphological lexicon provided by Apertium (lexicon type code "AP" in Table 2);
- Automatic morphological analysis of the raw monolingual corpora provided by the ConLL 2017 shared task organisers, using Apertium or Giellatekno morphological analysers (codes "APma" or "GTma"). More precisely, we downloaded the corresponding monolingual part of OPUS's OpenSubtitles2016 corpus,[8] tokenised it using a basic language-independent rule-based tokeniser, extracted the 1 million most frequent tokens, and retrieved all their morphological analyses by the corresponding morphological analyser provided by Apertium (or, failing that, Giellatekno). All these analyses were then gathered in the form of a lexicon.

We applied the direct conversion technique to 27 lexicons/languages, whereas the approach based on morphological analysers allowed us to cover 4 additional languages (2 via Apertium, 2 via Giellatekno). We also experimented the extension of these lexicons with data from the shared task training sets and the raw corpus automatically analysed by the UDPipe tool (Straka et al., 2016) by the organisers. Whenever it helped increasing our tagging results, as measured on the development sets, we applied these extensions. They are respectively indicated in Table 2 by "+T" and "+U" (this also applies to lexicons created in the next section).

For a few languages, we also created expanded versions of our lexicons using word embeddings re-computed on the raw data provided by the CoNLL 2017 shared task organisers. We assigned to words unknown to the lexicon the morphological information associated with the closest known word (using a simple Euclidian distance in the word embedding space).[9] When the best performing lexicon is one of these extended lexicons, it is indicated in Table 2 by the "-e" suffix.

2.2. Unsupervised cross-lingual transfer of Apertium and Giellatekno lexicons

In the context of our participation to the CoNLL 2017 shared task (Villemonte de La Clergerie et al., 2017), we were interested in producing as many morphological lexicons as possible for all languages involved, in order to optimise part-of-speech tagging and morphological annotation as much as possible. Yet the closed setting disallowed the use of any resources outside a pre-defined list, which included Apertium and Giellatekno monolingual morphological resources. For several languages, including Slovak, these projects did not provide adequate resources. However, Apertium does include a morphological lexicon for Czech, a language closely related to Slovak. We therefore decided to set up an unsupervised cross-lingual transfer technique to produce a Slovak morphological lexicon from the Apertium Czech one. In doing this, we rely on the hypothesis that the same morphological features and feature values are valid for Slovak as for Czech, and that part-of-speech and morphological features are stable across word-alignment links. This technique is therefore more relevant for closely related languages that are typologically similar.

The main cross-lingual resource we were allowed to use was the OpenSubtitles2016 corpus set, which provides monolingual as well as sentence-aligned bilingual parallel subtitle corpora. We took advantage of the parallel data in two steps, which we now illustrate on the example of Slovak and Czech. Firstly, we performed an endogenous, unsupervised extraction of a bilingual lexicon from the Slovak-Czech OpenSubtitles2016 parallel corpus, after a basic tokenisation using the same tool as mentioned

[6] http://universaldependencies.org/u/pos/

[7] http://universaldependencies.org/u/feat/

[8] Exploiting this dataset was allowed as per the CoNLL 2017 shared task rules.

[9] We did not used the embeddings provided by the organisers because we experimentally found that the 10-token window used to train these embeddings resulted in less accurate results than when using smaller windows, especially when the raw corpus available was of a limited size.

above. For this extraction process, we defined a matching metric between Czech tokens and Slovak tokens, which takes into account both the distribution of the tokens across sentences—the more often the Slovak token is found in sentences aligned with Czech sentences containing the Czech token, the higher the score—and the weighted Levenshtein edit distance between the two tokens, in order to give an advantage to cognate pairs.[10]

We first computed this distance using default weights for the Levenshtein distance (all operations cost 1). Character-to-character alignments produced by the Levenshtein algorithm allowed us to update the weight matrix[11] and recomputed all matching metrics. After reaching stability (i.e. after a few iterations) and metric-based thresholding, we obtained a bilingual Czech-Slovak lexicon. We then retrieved part-of-speech and morphological features for each Czech word in our bilingual lexicon and projected it onto all Slovak words it is aligned with.

We applied this technique to the following language pairs, in which the source (resourced) language is indicated first: Czech-Slovak, Italian-Latin, Russian-Ukrainian, Slovene-Croatian.The evaluation of the accuracy of this transfer technique is currently ongoing for the Czech-Slovak pair. Task-based evaluation of the output lexicons by using them as additional sources of information for a statistical part-of-speech tagger has already proven successful (see Section 3.). In Table 2 we use the code "TR*source_language*" to identify lexicons created in this cross-language way.

2.3. Creation of CoNLL-UL lexicons from other freely available lexicons

Independently of the Apertium and Giellatekno projects, many research teams have developed large-scale morphological lexicons for a variety of languages. We therefore designed a conversion process aimed at producing high-quality CoNLL-UL lexicons from these existing resources. The process is the following:

1. For each word, we register its corresponding ⟨category, morphological feature values, lemma⟩ triplets found in the source lexicon. We also extract its corresponding ⟨UPOS, lemma⟩ pairs from the UD v2 training set from the language's main treebank.[12] We then extract the most frequent UPOS for each source ⟨category, morphological feature values⟩ pair, based on identical ⟨wordform, lemma⟩ pairs.

2. We then apply heuristics based on the source-lexicon-to-UPOS correspondence obtained in the first step to overcome lemmatisation mismatches between the source lexicon and the UD treebank. We store the corresponding ⟨wordform, category, morphological features, source lemma, UD lemma⟩ 5-tuples for later use. We also update the ⟨source category, source morphological feature values, UPOS⟩ triples created during step 1 with the results of this step.

3. We use these updated ⟨source category, source morphological feature values, UPOS⟩ as a basis for extracting full correspondence patterns, i.e. 4-tuples of the form ⟨source category, source morphological feature values, UPOS, UFEAT⟩. More precisely, for each triple of the form ⟨source category, source morphological feature values, UPOS⟩ we select the most frequent UFEAT in the UD data among those associated with wordforms known to the lexicon with the category and morphological feature values at hand.

4. We output the updated ⟨source category, source morphological values, UPOS, UFEAT⟩ 4-tuples in an Excel file for manual reviewing and completion. We also include ⟨source category, source morphological values⟩ pairs for which no UPOS was found—often because they are not attested in the UD data—as well as UPOS for which no such triplet was created. This allows for the manual work to be exhaustive. We then perform a full manual review and correction of the Excel correspondence file.

5. We use the manually corrected file to automatically convert the source lexicon into a CoNLL-UL lexicon.

We applied this strategy on 18 freely available lexicons (see Table 2):

- 8 lexicons developed in the Alexina framework (Sagot, 2010), covering French (Sagot, 2010, Le*fff*), Polish (Sagot, 2007, PolLex), Slovak (Sagot, 2005, SkLex), Spanish (Molinero et al., 2009, Le*ffe*), Galician (Le*ffga*), Persian (Sagot and Walther, 2010, PerLex) German (Sagot, 2014, DeLex) and English (EnLex);
- 10 other lexicons covering Italian (Zanchetta and Baroni, 2005, Morph-it!), Swedish (Borin et al., 2008,

[10] More formally, we define the distributional distance $DS(s,t)$ between a token s in the source (resourced) language and a token t in the target language as

$$DS(s,t) = 2 \cdot \left(\frac{\mathrm{occ}(s)}{\mathrm{nbsentpairs}(s,t)} + \frac{\mathrm{occ}(t)}{\mathrm{nbsentpairs}(s,t)} \right)^{-1},$$

where $\mathrm{nbsentpairs}(s,t)$ is the number of sentence pairs (a source sentence aligned with a target sentence) such that the source sentence contains s and the target sentence contains t. This distance equals 1 if and only if all occurrences of s are in sentences aligned with sentences containing t, and vice versa. Next, we call $d_{\vec{w}}(s,t)$ the weighted Levenshtein distance between s and t, where $\vec{w}$ stores the weight of each possible operation (e.g. replacing "ŕ" by "r"). Finally, our matching metric is defined as $M(s,t) = DS(s,t) - \frac{1}{10} \cdot d^2_{\vec{w}} \cdot \max\left(\frac{1}{10}, \frac{1}{2} - DS\right)$. The idea underlying this metric is that s and t are likely to be translations of each another if their distributional score is high or if they are formally close, this latter criterion becoming more important if the distributional score decreases.

[11] The weight of an operation tranforming a source character c_s (or the empty string in the case of an insertion) into a target character c_t (or the empty string in the case of a deletion) is defined as $1 - \sqrt{\frac{\mathrm{transfocc}(c_s,c_t)}{\mathrm{occ}(c_s)}}$, where $\mathrm{transfocc}(c_s,c_t)$ is the number of times c_s was transformed into c_t in the previous iteration, and $\mathrm{occ}(c_s)$ is the total number of c_s's in the source lexicon.

[12] I.e. the treebank whose identifier is the language code itself (e.g. `fr` rather than `fr-sequoia`).

From	To	Form or Token	Lemma	UPOS	CPOS	UFEAT	Misc
0	1	*encodent*	encoder	VERB	_	Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin	_
0-2		auxquels					
0	1	*à*	à	ADP	_	_	_
1	2	*lesquels*	lequel	PRON	_	Gender=Masc|Number=Plur	_

Table 1: Two entries resulting from the conversion of the Le*fff* into the CoNLL-UL format (for space reasons, the CPOS column is displayed as if it were empty).

saldo), Ancient Greek (Heslin, 2007, Diogenes Ancient Greek lexicon), Latin (Heslin, 2007, Diogenes Latin lexicon), Croatian (Oliver and Tadić, 2004, hml), Irish (Měchura, 2014, INMDB), Norwegian (Bokmål) (The Language Council of Norway, 2011, OrdBank-BM), Portuguese (Ranchhod et al., 1999, Labellex-PT) and Slovenian (Krek et al., 2008, SloLeks).

Some of the above-listed Alexina lexicons include information about multi-word tokens. For instance, the French token auxquels 'to which$_{PL}$' is described in Le*fff* as the contraction of the two wordforms *à* 'to' and *lesquels* 'which$_{PL}$'. Moreover, the part-of-speech of the wordforms involved is sometimes specified. We automatically extended our converted lexicons with CoNLL-UL entries for these multi-word tokens by combining existing entries for the underlying wordforms. Whenever part-of-speech information is provided for a wordfrom, we limit ourselves to entries with the corresponding UPOS. The entry generated for auxquels is shown in Table 1.

2.4. Final treebank-based extension

We complemented all lexicons created using one of the three techniques described above with information extracted from the training part of the UD treebanks. For each language, we first extracted all entries, both simple entries and multi-word tokens, from the training part of the corresponding UD treebank.[13]

Next, in order to limit the bias towards UD training data, we automatically computed a frequency threshold and discarded those entries occurring less frequently.[14]

Finally, we discard all simple entries whose UPOS is already attested in our lexicon. The reason for this is that we want to preserve the consistency of our lexicons in terms of morphological annotation guidelines (UFEAT). UFEAT annotations in UD treebanks is not always consistent with UFEAT information found in our lexicons (for instance, a feature can be provided in one resource and not in the other). Merging entries extracted from UD treebanks with entries already present in our lexicons could therefore result in inconsistencies. Doing so for UPOS not yet covered (e.g. PUNCT in several Apertium- and Giellatekno-based lexicons) does not create such inconsistencies.

[13]For several languages, more than one UD treebank is available. In such cases, we use the concatenatation of the training parts of all of them.

[14]Our threshold is computed using the following heuristics: we order entries in decreasing order of frequency, and identify the minimum number of entries necessary to cover 90% of the data, starting from the most frequent one and accumulating entries in decreasing order of frequency. We then extract the number of occurrences $\text{occ}_{90\%}$ of the last selected entry. If $\text{occ}_{90\%} = 1$, which typically occurs on small datasets, we also compute $\text{occ}_{75\%}$. We then fix our frequency threshold via a minimum number of occurrences $\text{occ}_{\text{threshold}}$ defined as follows:

$$\text{occ}_{\text{threshold}} = \begin{cases} \max(3, \text{occ}_{90\%}) & \text{if } \text{occ}_{90\%} > 1 \\ \max(2, \text{occ}_{75\%}) & \text{if } \text{occ}_{90\%} = 1 \end{cases}$$

2.5. Results

Table 2 gives quantitative information about the lexicons produced by the three methods described in Sections 2.1. to 2.3. followed by the extension step described in Section 2.4.. They constitute the version 0.2 of the UDLexicons collection.

3. Preliminary task-based evaluation in part-of-speech taggers

The CoNLL-UL lexicons we extracted from Apertium and Giellatekno data, including those created via cross-lingual transfer, served as a source of additional features in two different part-of-speech tagging experiments. The first one is our participation in the CoNLL 2017 shared task. For this shared task, we wanted to explore many ways of creating such additional features from lexicons, and to compare different lexicon variants (cf. the "+U" and "+T" extensions). We therefore developed a new statistical MEMM tagger, following our previous work in this direction (Denis and Sagot, 2012), but this time based on the Vowpal Wabbit architecture. We selected our pre-processing architecture based on parsing results on the development sets. In particular, we chose to use either this new tagger or the UDPipe baseline provided by the organisers. Although our new tagger had higher accuracies than UDPipe on all datasets but 4 (Villemonte de La Clergerie et al., 2017, Table 1), we ended up using our tagger on only half of the testing datasets. This allowed us to be ranked 3/33 for UPOS tagging.[15,16]

In another recent experiment (Sagot and Martínez Alonso, 2017), we have shown that these lexicons also improve tagging results when a state-of-the-art neural architecture, namely that of Plank et al. (2016), is extended to take into account external lexical information, even when word encodings and character-based encodings are used.

[15]More recent, unofficial results using improved parsers have resulted in our tagger being selected more often rather than UDPipe, thus further improving our overall UPOS tagging results.

[16]Our UFEAT scores in this shared task are not meaningful, because we explicitly decided to only predict a subset of all morphological features.

Apertium/Giellatekno-based resources				
lang.	**type**	**#simple entries**	**#complex entries**	**#distinct wforms**
ar	AP-e	660K	3,055	246K
bg	AP	93K		76K
ca	AP-e_{100}	381K	47	261K
cs	AP	1,875K	10	480K
da	AP	683K		377K
de	AP	2,180K	28	411K
el	AP	47K		29K
en	AP	127K	1	96K
es	AP	325K	161	273K
et	GTma	44K		33K
eu	AP	49K		45K
fi	GTma	228K	17	156K
fr	AP-e^6_{1000}	156K	21	123K
gl	AP	241K	28	191K
he	AP	268K	4425	206K
hi	AP	159K		66K
hr	TRsl	14K		11K
id	AP	12K		12K
it	AP	278K	105	229K
kk	APma	434K		274K
la	TRit+T-e_{100}	13K		11K
lv	AP	314K		166K
nl	AP	167K		78K
no	AP	2,470K		1,373K
pl	AP	1,316K	291	525K
pt	AP	159K	184	119K
ro	AP	229K		151K
ru	AP	4,401K		2,159K
sk	TRcs	66K		36K
sl	AP	654K		203K
sv	AP	2,378K		1,319K
tr	APma	417K	697	246K
uk	TRru	23K		12K
ur	AP	98K		54K
zh	AP+U	17K		10K
Other resources				
lang.	**type**	**#simple entries**	**#complex entries**	**#distinct wforms**
de	DeLex	1,138K	282	264K
en	EnLex	695K	186	508K
es	Le*ff*e	843K	163	680K
fa	PerLex	178K	68	168K
fr	Le*ffff*	650K	22	456K
ga	INMDB	57K		39K
gl	Le*ff*ga	949K	28	402K
grc	Diogenes	4,490K		1,004K
hr	HML5	3,854K		1,208K
it	Morph-it!	785K	105	378K
la	Diogenes	1,870K		425K
nl	Alpino	122K		73K
no	OrdBank$_{\mathrm{BM}}$	878K		636K
pl	PolLex	1,368K	291	355K
pt	labellex-pt	1,841K	184	820K
sk	SkLex	750K		419K
sl	SloLeks	2,626K		880K
sv	Saldo	1,241K		701K

Table 2: CoNLL-UL lexicons, version 0.2 (see text for an explanation of the 'type' column and for references).

4. Conclusion and future steps

We have developed a collection of morphological lexicons compatible with the Universal Dependencies guidelines and format. These lexicons are freely available, under licences that depend on those of the original resources. Such a lexicon collection could serve as a starting point for providing the community with a set of lexical resources that will consistently complement the UD treebank collection, as well as the morphological analysers developed as companions to the CoNLL-UL proposal (More et al., 2018). These initiatives give easier access to morphological information in multilingual settings in contexts such as parsing and information extraction.

Our future steps are now threefold. Firstly, as mentioned above, the conversion mappings from the original categories and features to UPOS and UFEATs must be thoroughly reviewed and improved, in a way consistent with the UD treebanks. Secondly, the converted lexicons will be carefully evaluated, both in terms of precision and coverage. Thirdly, the availability of different lexicons for the same languages will make it possible to merge them in different ways, in order to optimise their coverage and accuracy. This is particularly true for lexicons that are likely to be less reliable, such as those created using the cross-lingual transfer technique described in Section 2.2. Finally, more lexicons will be included in the collection over time, such as the MULTEXT/MULTEXT-East lexicons that are distributed under a free licence.

5. References

Borin, L., Forsberg, M., and Lönngren, L. (2008). The hunting of the BLARK - SALDO, a freely available lexical database for Swedish language technology. In *Resourceful language technology. Festschrift in honor of Anna Sågvall Hein*, pages 21–32. Uppsala University, Uppsala, Sweden.

Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *Proc. of the Tenth Conference on Computational Natural Language Learning*, pages 149–164, New York City, USA.

Denis, P. and Sagot, B. (2012). Coupling an annotated corpus and a lexicon for state-of-the-art POS tagging. *Language Resources and Evaluation*, 46(4):721–736.

Erjavec, T. (2010). MULTEXT-East version 4: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. In *Proceedings of the 7th international conference on Language Resources and Evaluation (LREC'2010)*, Valletta, Malta.

Heslin, P. J. (2007). Diogenes, version 3.1. http://www.dur.ac.uk/p.j.heslin/Software/Diogenes/.

Ide, N. and Véronis, J. (1994). MULTEXT: Multilingual Text Tools and Corpora. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING'94)*, Kyoto, Japan.

Krek, S., Erjavec, T., and Holozan, P. (2008). Specifikacije za leksikon besednih oblik (kazalnik 3). Technical report, Projekt Sporazumevanje v slovenskem jeziku, Ljubljana, Slovenia.

Molinero, M. A., Sagot, B., and Nicolas, L. (2009). A morphological and syntactic wide-coverage lexicon for

Spanish: The leffe. In *Proc. of the 7th conference on Recent Advances in Natural Language Processing (RANLP 2009)*, Borovets, Bulgaria.

More, A., Çetinoğlu, Ö., Çöltekin, Ç., Habash, N., Sagot, B., Seddah, D., Taji, D., and Tsarfaty, R. (2018). CoNLL-UL: Universal Morphological Lattices for Universal Dependency Parsing. In *Proc. of LREC 2018*, Miyazaki, Japan.

Měchura, M. B. (2014). Irish National Morphology Database: A High-Accuracy Open-Source Dataset of Irish Words. In *Proc. of the Celtic Language Technology Workshop at CoLing*, Dublin, Ireland.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 shared task on dependency parsing. In *Proc. of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal dependencies v1: A multilingual treebank collection. In *LREC*.

Nivre, J., Agić, Ž., Ahrenberg, L., Aranzabe, M. J., Asahara, M., Atutxa, A., Ballesteros, M., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bosco, C., Bouma, G., Bowman, S., Candito, M., Cebiroğlu Eryiğit, G., Celano, G. G. A., Chalub, F., Choi, J., Çöltekin, Ç., Connor, M., Davidson, E., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dobrovoljc, K., Dozat, T., Droganova, K., Dwivedi, P., Eli, M., Erjavec, T., Farkas, R., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D., Garcia, M., Ginter, F., Goenaga, I., Gojenola, K., Gökırmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Grūzītis, N., Guillaume, B., Habash, N., Hajič, J., Hà Mỹ, L., Haug, D., Hladká, B., Hohle, P., Ion, R., Irimia, E., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kotsyba, N., Krek, S., Laippala, V., Lê Hồng, P., Lenci, A., Ljubešić, N., Lyashevskaya, O., Lynn, T., Makazhanov, A., Manning, C., Mărănduc, C., Mareček, D., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Missilä, A., Mititelu, V., Miyao, Y., Montemagni, S., More, A., Mori, S., Moskalevskyi, B., Muischnek, K., Mustafina, N., Müürisep, K., Nguyễn Thị, L., Nguyễn Thị Minh, H., Nikolaev, V., Nurmi, H., Ojala, S., Osenova, P., Øvrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Ramasamy, L., Real, L., Rituma, L., Rosa, R., Saleh, S., Sanguinetti, M., Saulīte, B., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shakurova, L., Shen, M., Sichinava, D., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Tsarfaty, R., Tyers, F., Uematsu, S., Uria, L., van Noord, G., Varga, V., Vincze, V., Washington, J. N., Žabokrtský, Z., Zeldes, A., Zeman, D., and Zhu, H. (2017). Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Oliver, A. and Tadić, M. (2004). Enlarging the Croatian morphological lexicon by automatic lexical acquisition from raw corpora. In *Proc. of LREC 2004*, pages 1259–1262, Lisbon, Portugal.

Petrov, S., Das, D., and McDonald, R. (2012). A Universal Part-of-Speech Tagset. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Plank, B., Søgaard, A., and Goldberg, Y. (2016). Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany.

Ranchhod, E., Mota, C., and Baptista, J. (1999). A Computational Lexicon of Portuguese for Automatic Text Parsing. In *Proc. of the SIGLEX99 workshop on Standardizing Lexical Resources*, College Park, Maryland, USA.

Sagot, B. and Martínez Alonso, H. (2017). Improving neural tagging with lexical information. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 25–31, Pisa, Italy.

Sagot, B. and Walther, G. (2010). A morphological lexicon for the Persian language. In *Proc. of LREC 2010*, Valletta, Malta.

Sagot, B. (2005). Automatic acquisition of a Slovak lexicon from a raw corpus. In *Lecture Notes in Artificial Intelligence 3658, Proc. of TSD'05*, pages 156–163, Karlovy Vary, République tchèque. Springer-Verlag.

Sagot, B. (2007). Building a morphosyntactic lexicon and a pre-syntactic processing chain for Polish. In *Proc. of LTC 2005*, pages 423–427, Poznań, Poland.

Sagot, B. (2010). The Lefff, a freely available, accurate and large-coverage lexicon for french. In *Proceedings of the 7th international conference on Language Resources and Evaluation (LREC'2010)*, Valletta, Malta.

Sagot, B. (2014). DeLex, a freely-avaible, large-scale and linguistically grounded morphological lexicon for German. In *Language Resources and Evaluation Conf.*, Reykjavik, Iceland.

Seddah, D., Tsarfaty, R., Kübler, S., Candito, M., Choi, J. D., Farkas, R., Foster, J., Goenaga, I., Gojenola Galletebeitia, K., Goldberg, Y., Green, S., Habash, N., Kuhlmann, M., Maier, W., Nivre, J., Przepiórkowski, A., Roth, R., Seeker, W., Versley, Y., Vincze, V., Woliński, M., Wróblewska, A., and Villemonte de La Clergerie, E. (2013). Overview of the SPMRL 2013 Shared Task: A Cross-Framework Evaluation of Parsing Morphologically Rich Languages. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 146–182, Seattle, Washington, USA.

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evalua-*

tion (LREC 2016), Portorož, Slovenia.

The Language Council of Norway. (2011). Norsk Ordbank in Norwegian Bokmål. Common Language Resources and Technology Infrastructure Norway (CLARINO) Bergen Repository, http://hdl.handle.net/11509/136.

Villemonte de La Clergerie, É., Sagot, B., and Seddah, D. (2017). The ParisNLP entry at the ConLL UD Shared Task 2017: A Tale of a #ParsingTragedy. In *Conference on Computational Natural Language Learning*, Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pages 243–252, Vancouver, Canada, August.

Zanchetta, E. and Baroni, M. (2005). Morph-it! a free corpus-based morphological resource for the Italian language. In *Proc. of the Corpus linguistics Conf.*, pages 1–12, Birmingham, UK.

Zeman, D., Mareček, D., Popel, M., Ramasamy, L., Štěpánek, J., Žabokrtský, Z., and Hajič, J. (2012). Hamledt: To parse or not to parse? In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, Ç., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada, August. Association for Computational Linguistics.

UniMorph 2.0: Universal Morphology

Christo Kirov[1], **Ryan Cotterell**[1], **John Sylak-Glassman**[1], **Géraldine Walther**[2]
Ekaterina Vylomova[3], **Patrick Xia**[1], **Manaal Faruqui**[4], **Sebastian Mielke**[1], **Arya D. McCarthy**[1]
Sandra Kübler[5], **David Yarowsky**[1], **Jason Eisner**[1], **Mans Hulden**[6]
[1]Johns Hopkins University, [2]University of Zurich, [3]University of Melbourne
[4]Google, [5]Indiana University, [6]University of Colorado
Baltimore, Zurich, Melbourne, New York, Bloomington, Boulder
{ckirov1, ryan.cotterell, jcsg, paxia, sjmielke, arya, yarowsky, eisner}@jhu.edu, geraldine.walther@uzh.ch
evylomova@gmail.com, mfaruqui@google.com, skuebler@indiana.edu, mans.hulden@colorado.edu

Abstract

The Universal Morphology (UniMorph) project is a collaborative effort to improve how NLP handles complex morphology across the world's languages. The project releases annotated morphological data using a universal tagset, the UniMorph schema. Each inflected form is associated with a lemma, which typically carries its underlying lexical meaning, and a bundle of morphological features from our schema. Additional supporting data and tools are also released on a per-language basis when available. UniMorph is based at the Center for Language and Speech Processing (CLSP) at Johns Hopkins University in Baltimore, Maryland. This paper details advances made to the collection, annotation, and dissemination of project resources since the initial UniMorph release described at LREC 2016.

Keywords: morphology, multilingual, lexical resources

1. Introduction

Complex morphology is ubiquitous among the languages of the world. For example, roughly 80% of languages use morphology to mark verbal tense and 65% mark nominal case (Haspelmath et al., 2005). While overlooked in the past, explicit modeling of morphology has been shown to improve performance on a number of downstream HLT tasks, including including machine translation (MT) (Dyer et al., 2008), speech recognition (Creutz et al., 2007), parsing (Seeker and Çetinoğlu, 2015), keyword spotting (Narasimhan et al., 2014), and word embedding (Cotterell et al., 2016b). This has led to a surge of new interest and work in this area (Durrett and DeNero, 2013; Ahlberg et al., 2014; Nicolai et al., 2015; Faruqui et al., 2016).

The Universal Morphology (UniMorph) project, centered at the Center for Language and Speech Processing (CLSP) at Johns Hopkins University is a collaborative effort to improve how NLP systems handle complex morphology across the world's languages. The project releases annotated morphological data using a universal tagset, the UniMorph schema. Each inflected form is associated with a lemma, which typically carries its underlying lexical meaning, and a bundle of morphological features from our schema. Additional supporting data and tools are also released on a per-language basis when available.

Kirov et al. (2016) introduced version 1.0 of the UniMorph morphological database, created by extracting and normalizing the inflectional paradigms included in Wiktionary (`www.wiktionary.org`), a large, broadly multi-lingual crowd-sourced collection of lexical data. This paper describes UniMorph 2.0. It details improvements in Wiktionary extraction and annotation, as well as normalization of non-Wiktionary resources, leading to a much higher quality morphological database. The new dataset spans 52 languages representing a range of language families. As in UniMorph 1.0, we provide paradigms from highly-inflected open-class word categories — nouns, verbs, and adjectives. Many of the included languages are extremely low-resource, e.g., Quechua, Navajo, and Haida. This data was used as the basis for the CoNLL 2017 Shared Task on Morphological Learning (`http://sigmorphon.org/conll2017`) (Cotterell et al., 2017).

2. Wiktionary Extraction

In Kirov et al. (2016), we introduced version 1.0 of the UniMorph morphological database, based on a very large-scale parsing and normalization of Wiktionary. Wiktionary is a broadly multilingual resource with many crowd-sourced morphological paradigms in the form of custom HTML tables. Figure 1 illustrates the challenge associated with extracting this data. Wiktionary is designed for human, rather than machine readability, and authors have extensive freedom in formatting data. This leads to wildly differing table layouts across languages which need to be converted to a consistent tabular format.

The extraction process developed for UniMorph 1.0 relied heavily on statistical, visual, and positional heuristics (Sylak-Glassman et al., 2015b) to:

1. Determine which entries in an HTML table are inflected forms and which are grammatical descriptors.
2. Link each inflected form with its appropriate descriptors.
3. Convert each set of linked descriptors into a universal feature annotation schema, described in detail in Sylak-Glassman (2016).[1]

This led to a large dataset of 952,530 unique noun, verb, and adjective lemmas across 350 languages. Unfortunately,

[1] `unimorph.github.io/doc/unimorph-schema.pdf`

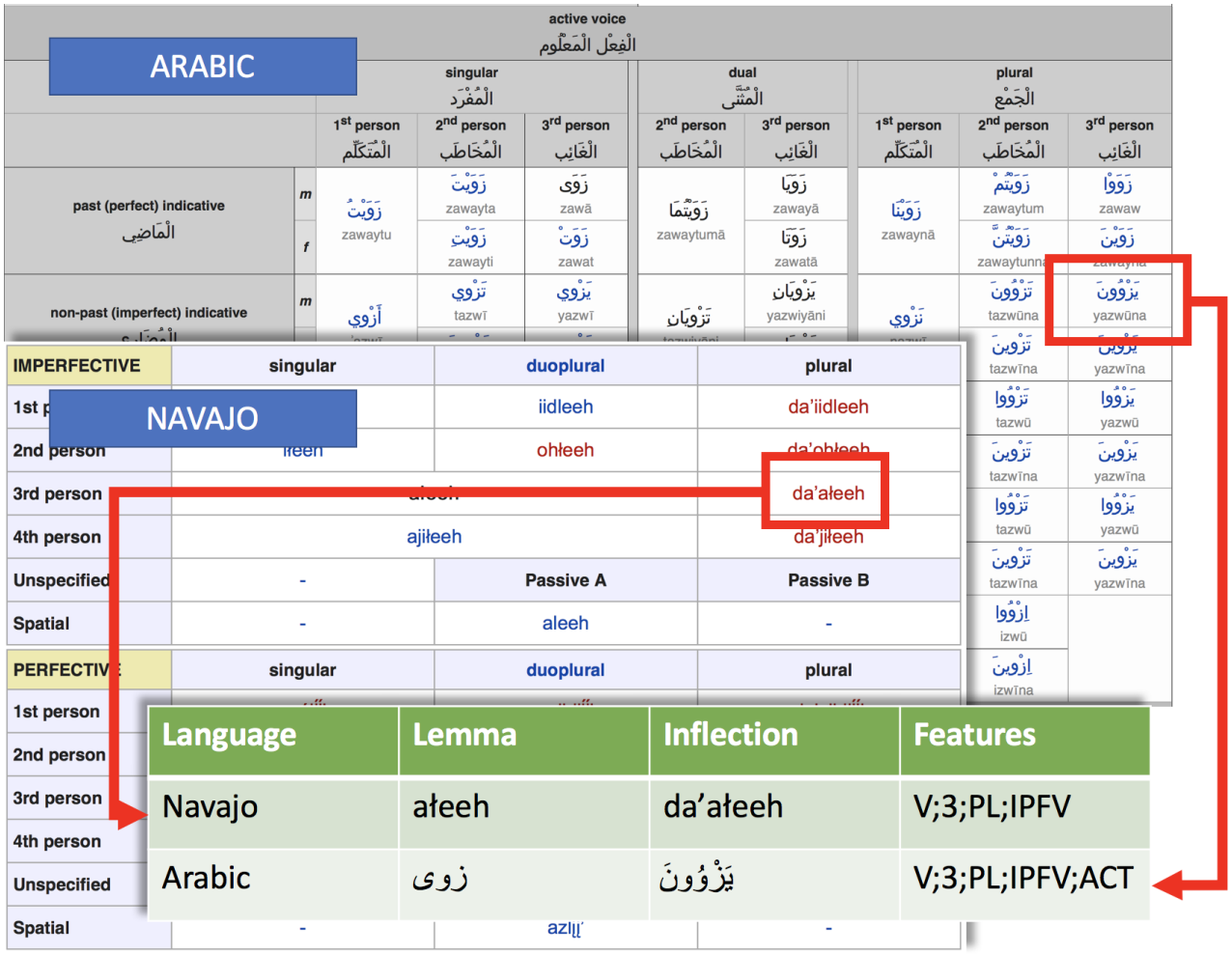


Figure 1: Paradigm extraction and normalization.

the UniMorph 1.0 dataset was very error-prone due to the inability of our heuristics to fully cover the degree of inconsistency found in Wiktionary. For many lemmas, inflected forms were systematically linked to incorrect feature vectors. To correct these errors, we noted that for each part-of-speech within a language in Wiktionary, authors use only a handful of distinct table layouts. Thus, it was sufficient for a human to verify and correct a single lemma parse from a particular layout, and apply those corrections to all similar lemma parses. A custom verification and correction process was created and applied to 8 languages (Arabic, Finnish, Georgian, German, Navajo, Russian, Spanish, and Turkish) in preparation for the SIGMORPHON 2016 Shared Task on Morphological Reinflection (Cotterell et al., 2016a).

For UniMorph 2.0, we noticed that the effort required to verify and correct an automatic parse of a Wiktionary table layout was greater than or equal to the effort required for a human to *directly* annotate a table with UniMorph features instead. Figure 2 illustrates this simplified process. Each language's HTML tables were parsed using Python's `pandas` library (`pandas.pydata.org`) and grouped according to their tabular structure and number of cells. Each group represents a different type of paradigm (e.g., regular verb).

For each group, a sample table was selected, and an annotator replaced each inflected form in the table with the appropriate UniMorph features. All annotation was compliant with the UniMorph Schema, which was designed to represent the full range of semantic distinctions that can be captured by inflectional morphology in any language (Sylak-Glassman et al., 2015a). The schema is similar in form and spirit to other tagset universalization efforts, such as the Universal Dependencies Project (Choi et al., 2015) and Interset (Zeman, 2008), but is designed specifically for typological completeness for inflectional morphology, including a focus on the morphology of especially low-resource languages. It includes over 200 base features distributed among 23 dimensions of meaning (i.e., morphological categories), including both common dimensions like tense and aspect as well as rarer dimensions like evidentiality and switch-reference. Despite the high coverage of the UniMorph tagset, for UniMorph 2.0, annotators were allowed to employ additional 'language specific' LGSPEC(1, 2, 3, etc.) features to mark any missing distinctions, or purely optional form variants that are not associated with a semantic difference. The Spanish imperfect subjunctive, for example, has two interchangeable forms (*-ra* and *-se*):



	singular	plural
nominative	лéмма lémma	лéммы lémmy
genitive	лéммы lémmy	лéмм lémm
dative	лéмме lémme	лéммам lémmam
accusative	лéмму lémmu	лéммы lémmy
instrumental	лéммой, лéммою lémmoj, lémmoju	лéммами lémmami
prepositional	лéмме lémme	лéммах lémmax

(a) Raw Wiktionary

	singular	plural
nominative	лéмма	лéммы
genitive	лéммы	лéмм
dative	лéмме	лéммам
accusative	лéмму	лéммы
instrumental	лéммой	лéммами
prepositional	лéмме	лéммах

(b) Unannotated Table

	singular	plural
nominative	N;NOM;SG	N;NOM;PL
genitive	N;GEN;SG	N;GEN;PL
dative	N;DAT;SG	N;DAT;PL
accusative	N;ACC;SG	N;ACC;PL
instrumental	N;INS;SG	N;INS;PL
prepositional	N;ESS;SG	N;ESS;PL

(c) Annotated Table

Figure 2: Annotation process

lemma	form	features
gravitar	gravitaras	V;SBJV;PST;2;SG;LGSPEC1
gravitar	gravitases	V;SBJV;PST;2;SG;LGSPEC2

As each example table is identical in structure to all members in the same layout group, annotating just one example allows mapping every inflected form in every table in the group to its corresponding morphological features. This minimizes the human annotation effort required per language, to the point that only 3 annotators were able to produce a complete initial dataset for 47 Wiktionary languages in a matter of days (data for these 47 languages, listed in Table 2, supplants the corresponding language data in the UniMorph 1.0 dataset).

Some of the extracted paradigms from Wiktionary were subject to additional post-processing. In particular, some Wiktionary tables contain multiple forms in the same cell. In the case of multiple forms, we separated them into their own entries. Looking at another Spanish example, we separate *tu* and *vos* forms corresponding to dialect differences in the choice of second person pronoun.

gravitar	gravitas(tú) gravitás(vos)	V;IND;PRS;2;SG
gravitar	gravitas	V;SBJV;PST;2;SG;LGSPEC3
gravitar	gravitás	V;SBJV;PST;2;SG;LGSPEC4

Finally, the content of all initial annotations was also verified as linguistically sensible by a second, larger set of adjudicators who were either native speakers of the language they reviewed or had significant expertise through research. The final dataset sizes are given by language in table 2.

3. Non-Wiktionary Data Sources

In additional to our large database of annotated inflected forms derived from Wiktionary, UniMorph 2.0 includes morphological data for several additional languages from non-Wiktionary sources. Data for Khaling, Kurmanji Kurdish, and Sorani Kurdish was derived from the Alexina project (Walther et al., 2013; Walther et al., 2010; Walther and Sagot, 2010).[2] Novel data for Haida, a severely endangered North American language isolate, was prepared by Jordan Lachler (University of Alberta). Basque language data was extracted from a manually designed finite-state morphological analyzer (Alegria et al., 2009). Data for all these additional languages was reformatted to match the Wiktionary-derived data using custom Python scripts. Any dataset-specific annotation was manually mapped to the UniMorph schema standard.

4. Supplementary Structured Data

As discussed in Kirov et al. (2016), we also mine additional structured data from Wiktionary. A number of Wiktionary pages contain lists of derived words under the HTML heading 'Related/Derived Terms' — 'sunflower' for example, appears on the list for the base lemma 'flower.' Furthermore, Wiktionary also contains tables of lemma translations. The English lemma 'flower' contains the translation entry 'Danish: blomstre.' As part of UniMorph 1.0, we collected an average of 3.42 derived terms per lemma across 76,038 lemmas, and an average of 3.54 translations per annotated lemma.

For UniMorph 2.0, we are releasing two additional resource types. First, only a subset of Wiktionary languages and lemmas contain embedded morphological tables. There are many more bare lemmas with no form of morphological annotation. We also scrape these lemmas, and provide a list of them along with their associated part of speech.

Second, for a number of languages in UniMorph, we provide multi-word English glosses for complex inflected wordforms. For example, the Spanish word *comprábamos* is mapped to the gloss '(we) were buying.' These glosses are generated for languages where adequately-sized lemma-to-lemma translation dictionaries are available, via the following general process:

1. Perform a generally language-independent conversion of UniMorph feature vectors to an English gloss template, e.g., V;1;PL;PST;IPFV → '(we) were VBG.' Here, VBG is a Penn Treebank tag which indicates that the template can be filled with the *-ing* form of an English verb.

2. Given an inflected lemma in the language with a particular feature vector and lemma translation, find the corresponding gloss template, e.g., *comprábamos*, *comprar*, V;1;PL;IPFV → 'buy: (we) were VBG'

3. Replace the English lemma placeholder in the template with the appropriately generated form of the English lemma, 'buy, (we) were VBG' → '(we) were buying'

[2] https://gforge.inria.fr/projects/alexina/

Language	Inflections	Glosses
Amharic	566553	1736981
Farsi	206711	582449
Hausa	55860	124492
Hungarian	2814006	9754197
Oromo	26690	246856
Russian	560067	2219960
Somali	451217	1144096
Spanish	153121	368636
Ukrainian	20288	41590
Yoruba	127833	356502
Total	**4982569**	**16575759**

Table 1: English glosses by language.

Generating complicated tenses of multi-word lemmata (e.g. "They will not have looked it up") and robustly generating appropriate English inflections for diverse and noisy translation dictionaries, are both a challenge and strength of this work.

Table 1 shows the a summary of the current resource sizes of selected languages, along with the number of distinct inflections covered, and the number of expanded phrasal glosses generated given multiple translations per lemma.

5. Community Features

Following the model of Universal Dependencies (UD),[3], UniMorph is intended to be a highly collaborative project. To that end, all data and tools associated with the project are released on a rolling basis with a permissive open source license. The main portal for the UniMorph project, which provides a high-level overview of project goals and activities, is `www.unimorph.org`. The hub for downloadable data and resources is `unimorph.github.io`. A full specification of the UniMorph annotation schema is available. For each language, the site indicates how many forms and paradigms have been extracted, the source of the data, and available parts of speech. The site is also designed to encourage community involvement. Each language is associated with a public issue tracker that allows users to discuss errors and issues in the available data and annotations. Interested users can also become part of the UniMorph mailing list.

Moving forward, we also intend to develop connections with other morphological resources. The Universal Dependencies project, for example, provides a token-level corpus complementary to the UniMorph type-level data. A preliminary survey of UD annotations shows that approximately 68% of UD features map directly to UniMorph schema equivalents. This set covers 97.04% of complete UD tags. Some UD features lie outside the current scope of UniMorph, which marks primarily morphosyntactic and morphosemantic distinctions. These include, for example, markers for abbreviated forms and foreign borrowings.

[3] `universaldependencies.org`

Language	Family	Lemmata / Forms
Albanian	Indo-European	589 / 33483
Arabic	Semitic	4134 / 140003
Armenian	Indo-European	7033 / 338461
Basque	Isolate	26 / 11889
Bengali	Indo-Aryan	136 / 4443
Bulgarian	Slavic	2468 / 55730
Catalan	Romance	1547 / 81576
Czech	Slavic	5125 / 134527
Danish	Germanic	3193 / 25503
Dutch	Germanic	4993 / 55467
English	Germanic	22765 / 115523
Estonian	Uralic	886 / 38215
Faroese	Germanic	3077 / 45474
Finnish	Uralic	57642 / 2490377
French	Romance	7535 / 367732
Georgian	Kartvelian	3782 / 74412
German	Germanic	15060 / 179339
Haida	Isolate	41 / 7040
Hebrew	Semitic	510 / 13818
Hindi	Indo-Aryan	258 / 54438
Hungarian	Uralic	13989 / 490394
Icelandic	Germanic	4775 / 76915
Irish	Celtic	7464 / 107298
Italian	Romance	10009 / 509574
Khaling	Sino-Tibetan	591 / 156097
Kurmanji Kurdish	Iranian	15083 / 216370
Latin	Romance	17214 / 509182
Latvian	Baltic	7548 / 136998
Lithuanian	Baltic	1458 / 34130
Lower Sorbian	Germanic	994 / 20121
Macedonian	Slavic	10313 / 168057
Navajo	Athabaskan	674 / 12354
Northern Sami	Uralic	2103 / 62677
Norwegian Bokmål	Germanic	5527 / 19238
Norwegian Nynorsk	Germanic	4689 / 15319
Persian	Iranian	273 / 37128
Polish	Slavic	10185 / 201024
Portuguese	Romance	4001 / 303996
Quechua	Quechuan	1006 / 180004
Romanian	Romance	4405 / 80266
Russian	Slavic	28068 / 473481
Scottish Gaelic	Celtic	73 / 781
Serbo-Croatian	Slavic	24419 / 840799
Slovak	Slavic	1046 / 14796
Slovene	Slavic	2535 / 60110
Sorani Kurdish	Iranian	274 / 22990
Spanish	Romance	5460 / 382955
Swedish	Germanic	10553 / 78411
Turkish	Turkic	3579 / 275460
Ukrainian	Slavic	1493 / 20904
Urdu	Indo-Aryan	182 / 12572
Welsh	Celtic	183 / 10641

Table 2: Total number of lemmata and forms available for each language in the morphological database.

6. Conclusion

As part of the UniMorph project, we are releasing the largest available database of high-quality morphological paradigms across a typologically-diverse set of languages. To create this dataset, we developed a type-based annotation procedure that enables extracting a large amount of data from Wiktionary with minimal effort from human annotators. The procedure successfully handles idiosyncratic variation in formatting across the languages in Wiktionary. UniMorph also prescribes a universal tagging schema and data formats that allow data to be incorporated from non-

Wiktionary data sources. The project welcomes community involvement, and all data and tools are released under a permissive open-source license at `unimorph.github.io`. UniMorph 2.0 data has already been used as the basis for the successful CoNLL 2017 Shared Task on Morphological Learning, the first shared task on morphology in the CoNLL community (Cotterell et al., 2017).

7. Bibliographical References

Ahlberg, M., Forsberg, M., and Hulden, M. (2014). Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 569–578, Gothenburg, Sweden, April. Association for Computational Linguistics.

Alegria, I., Etxeberria, I., Hulden, M., and Maritxalar, M. (2009). Porting Basque morphological grammars to *foma*, an open-source tool. In *International Workshop on Finite-State Methods and Natural Language Processing*, pages 105–113. Springer.

Choi, J., de Marneffe, M.-C., Dozat, T., Ginter, F., Goldberg, Y., Hajič, J., Manning, C., McDonald, R., Nivre, J., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2015). Universal Dependencies. Accessible at: http://universaldependencies.github.io/docs/, January.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Yarowsky, D., Eisner, J., and Hulden, M. (2016a). The SIGMORPHON 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22, Berlin, Germany, August. Association for Computational Linguistics.

Cotterell, R., Schütze, H., and Eisner, J. (2016b). Morphological smoothing and extrapolation of word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1651–1660, Berlin, Germany, August. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., Xia, P., Faruqui, M., Kübler, S., Yarowsky, D., Eisner, J., and Hulden, M. (2017). The CoNLL-SIGMORPHON 2017 shared task. In *CoNLL-SIGMORPHON 2017 Shared Task*.

Creutz, M., Hirsimäki, T., Kurimo, M., Puurula, A., Pylkkönen, J., Siivola, V., Varjokallio, M., Arisoy, E., Saraclar, M., and Stolcke, A. (2007). Analysis of morph-based speech recognition and the modeling of out-of-vocabulary words across languages. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 380–387. Association for Computational Linguistics.

Durrett, G. and DeNero, J. (2013). Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1185–1195, Atlanta, Georgia, June. Association for Computational Linguistics.

Dyer, C., Muresan, S., and Resnik, P. (2008). Generalizing word lattice translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, pages 1012–1020, Columbus, Ohio, June. Association for Computational Linguistics.

Faruqui, M., Tsvetkov, Y., Neubig, G., and Dyer, C. (2016). Morphological inflection generation using character sequence to sequence learning. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 634–643, San Diego, California, June. Association for Computational Linguistics.

Haspelmath, M., Dryer, M., Gil, D., and Comrie, B. (2005). The world atlas of language structures (WALS).

Kirov, C., Sylak-Glassman, J., Que, R., and Yarowsky, D. (2016). Very-large scale parsing and normalization of Wiktionary morphological paradigms. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3121–3126. European Language Resources Association (ELRA), May.

Narasimhan, K., Karakos, D., Schwartz, R., Tsakalidis, S., and Barzilay, R. (2014). Morphological segmentation for keyword spotting. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 880–885, Doha, Qatar, October. Association for Computational Linguistics.

Nicolai, G., Cherry, C., and Kondrak, G. (2015). Inflection generation as discriminative string transduction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 922–931, Denver, Colorado, May–June. Association for Computational Linguistics.

Seeker, W. and Çetinoğlu, O. (2015). A graph-based lattice dependency parser for joint morphological segmentation and syntactic analysis. *Transactions of the Association for Computational Linguistics*, 3:359–373.

Sylak-Glassman, J., Kirov, C., Post, M., Que, R., and Yarowsky, D. (2015a). A universal feature schema for rich morphological annotation and fine-grained cross-lingual part-of-speech tagging. In Cerstin Mahlow et al., editors, *Proceedings of the 4th Workshop on Systems and Frameworks for Computational Morphology (SFCM)*, Communications in Computer and Information Science, pages 72–93. Springer, Berlin, September.

Sylak-Glassman, J., Kirov, C., Yarowsky, D., and Que, R. (2015b). A language-independent feature schema for inflectional morphology. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 674–680, Beijing, July. Association for Computational Linguistics.

Walther, G. and Sagot, B. (2010). Developing a large-scale lexicon for a less-resourced language: General methodology and preliminary experiments on Sorani Kurdish.

In *Proceedings of the SaLTMiL Workshop on Creation and Use of Basic Lexical Resources for Less-Resourced Languages (at LREC)*, Valetta, Malta. European Language Resources Association (ELRA).

Walther, G., Sagot, B., and Fort, K. (2010). Fast development of basic NLP tools: Towards a lexicon and a POS tagger for Kurmanji Kurdish. In *Proceedings of the 29th International Conference on Lexis and Grammar*, Belgrade.

Walther, G., Jacques, G., and Sagot, B. (2013). Uncovering the inner architecture of Khaling verbal morphology, September. Presentation at the 3rd Workshop on Sino-Tibetan Languages of Sichuan, Paris, September 2013.

Zeman, D. (2008). Reusable tagset conversion using tagset drivers. In *Proceedings of LREC 2008*, pages 213–218.

A Computational Architecture for the Morphology of Upper Tanana

Olga Lovick[1], **Christopher Cox**[2], **Miikka Silfverberg**[3], **Antti Arppe**[4], **Mans Hulden**[3]
First Nations University of Canada[1] Carleton University[2] University of Colorado[3]
University of Alberta[4]
Olga@lithophile.com, christopher.cox@carleton.ca, Miikka.Silfverberg@colorado.edu, arppe@ualberta.ca,
mans.hulden@colorado.edu

Abstract

In this paper, we describe a computational model of Upper Tanana, a highly endangered Dene (Athabaskan) language spoken in eastern interior Alaska (USA) and in the Yukon Territory (Canada). This model not only parses and generates inflected Upper Tanana verb forms, but uses the language's verb theme category system, a system of lexical-inflectional verb classes, to additionally predict possible derivations and their morphological behavior. This allows us to model a large portion of the Upper Tanana verb lexicon, making it more accessible to learners and scholars alike. Generated derivations will be compared against the narrative corpus of the language as well to the (much more comprehensive) lexical documentation of closely related languages.

Keywords: Dene languages, inflection, derivation

1. Upper Tanana

Upper Tanana (ISO 639-3: tau) belongs to the Alaskan subgroup of the Northern Dene language family. Documentation of the language began in 1959. There are two published text collections (Tyone, 1996; David, 2017) as well as two corpora of unpublished texts collected by James Kari and Olga Lovick. The language is severely endangered with fewer than 50 speakers, most of them elderly.

The verbal morphology of Upper Tanana is typical of a Dene language (Rice, 2000), and often involves a complex interweaving of non-continuous lexical, derivational, and inflectional prefix sequences. Verb stems vary with aspect and mode in a synchronically opaque fashion. The verb can be represented as a template:

12	11	10	9	8	7	#	6	5	4	3	2	1	0	-1
Postpositional object	Postposition	Adverbial-derivational	Iterative	Incorporate	Distributive	– Disjunct boundary –	Pronominal	Qualifier	Conjugation	Mode	Subject	Voice/valence	Stem	Suffix

Table 1: The Upper Tanana verb, represented as a template

This templatic approach, albeit popular in Dene linguistics from its inception (cf. Goddard (1911; Hoijer (1945; Li (1946; Sapir and Hoijer (1967)) is not without challenges. It is common for the prefixes in Positions 1–6 in the above template to coalesce into one or two syllables, often through relatively opaque morphophonological processes. Templatic analyses also typically require large numbers of Ø-morphemes in analysis, even for comparatively simple verb forms, and rely heavily on complex systems of morpheme co-occurrence restrictions to prevent overgeneration. All of these factors contribute to the perception that Dene languages are inherently "difficult", which undermines both revitalization and computational modelling efforts.

A possible solution to this problem is to represent inflection in Dene languages paradigmatically as sets of phonologically precomposed portmanteau morphemes (cf. Arppe et al. (2017) and Cox et al. (2016)). This allows learners to learn sets of paradigmatic 'chunks' like the ones in Table 2, required by all verbs belonging to same lexical-inflectional class of "operative" verb themes.[1]

	Ø-Ipfv	aa-Pfv	Fut	Opt
1s	ag	ag	t-ag	og
2s	i̱l	i̱l	t-i̱l	u̱l
3s	el	al	t-al	ul
1p	ts'-el	ts'-al	ts'&t-al	ts'-ul
2p	al	at	t-al	al
3p	h-el	h-et	h&t-al	h-ul

Table 2: Inflection pattern for operative verb themes with L-voice/valence marker

When learning a new verb that belongs to the operative class and thus requires the inflection pattern in Table 2, a learner only needs to know where in the lexical entry the inflectional chunk is inserted. This is illustrated in Table 3 for the verb theme ch'+L+dzüh 'dance' in the Perfective mode; the precomposed chunks from Table 2 are inserted following the *ch'-* prefix and before the stem. The remaining morphophonemic adjustments (in this example, in the first and third person plural) are relatively minor and quite regular, which facilitates both language learning and the computational modelling discussed below.

The basic lexical entry for verbs in Dene languages is known as a verb theme (Kari, 1979), and consists of a voice/valence marker and a stem, as well as zero or more re-

[1]The hyphen in all 1p and 3p forms indicates that morphological material may intervene here; the ampersand in the Future forms signals that additional morphophonemic adjustments have to be made.

	Surface form	Breakdown
1s	ch'agndzia'	ch'-ag-dzia'
2s	ch'įldzia'	ch'-įl-dzia'
3s	ch'aldzia'	ch'-al-dzia'
1p	nts'aldzia'	n-ch'-ts'al-dzia'
2p	ch'aldzia'	ch'-al-dzia'
3p	ch'ihaldzia'	ch'-h-al-dzia'

Table 3: *aa*-perfective of ch'+L+dzüh (*op.*) 'dance'

quired lexical prefixes. Verb themes fall into larger lexical-inflectional classes referred to in Dene linguistics as verb theme categories. These are characterized by shared conjugation markers in the Imperfective and Perfective, a shared primary aspect indicated by (historic) stem suffixation pattern, derivational potential, and semantic properties (durativity, telicity...). There are 10 verb theme categories, as shown in Table 4.

VTC	Example	Cnj	Aspect
Motion	Ø+haayh 's. arrive'	n, n	Momentaneous
Successive	O+Ø+got 'punch O'	Ø, aa	Durative
Operative	ch'+L+dzüh 'dance'	Ø, aa	Durative
Conversive	O+H+tsįį 'make s. O'	Ø, dh	Conclusive
Extension	Ø+'ah 'it extends'	n, aa	Neuter
Classific.	Ø+'ąą 'classify CO'	dh, aa	Neuter
Positional	Ø+dah 's. sit'	dh, aa	Neuter
Stative	H+ts'iik 'be sick'	dh, aa	Neuter
Dimens.	Ø+chaa 'be big'	Ø, aa	Neuter
Descriptive	Ø+łįį 'be'	Ø, aa	Neuter

Table 4: Verb theme categories in Upper Tanana

The verb theme category system is quite powerful at predicting a verb theme's behavior (Kari, 1979). By knowing that Ø+haayh 's. arrive' and O+Ø+'ąą 'handle compact O' belong to the motion theme category, we also know that they (1) require (n, n) conjugation markers, (2) have punctual, telic meaning, (3) require the Momentaneous Aspect in their primary derivation, (4) allow aspectual derivations in the Momentaneous, Perambulative, Reversative Aspects, (5) have enormous derivational potential (55 directional aspectual prefix strings plus many non-aspectual derivations), (6) allow the Progressive superaspect, (7) allow the Inceptive derivation, (8) restrict the Imperfective to second person subjects in most syntactic environments, and so on. As we discuss in 3., modelling verb theme categories allows much of the derivational morphology of Upper Tanana to be captured in a straightforward way, building on the representation of inflectional processes described in 2..

2. Computational model

Following prior work for related Tsuut'ina initially presented in Cox et al. (2016) and detailed in Arppe et al. (2017), we make use of finite state machines (FSMs) (see e.g. Beesley and Karttunen (2003)) as the computational formalism for implementing our model for Upper Tanana. FSMs have become one standard way for computationally modeling the morphological structure of words, and there are currently several open source implementations of FSM compilers, e.g. *xfst* (Beesley and Karttunen, 2003), *foma* (Hulden, 2009) and HFST (Lindén et al., 2011). The key advantages of FSMs are most crucially that they provide a calculus for powerful manipulations and are designed for rule-based definition of paradigms, which does not require large corpora from which to learn such rules, usually lacking for endangered languages. Furthermore, as well-established computational data structures, FSMs allow for easy integration with other software applications, for instance as spell-checking modules within word-processors, morphologically "intelligent" electronic dictionaries, and "intelligent" computer-aided language-learning applications. Here, we make use of the Giella infrastructure, developed by the Giellatekno and Divvun research teams at the University of Tromsø (Trosterud, 2006), which provide ready-made solutions for the integration of an FSM-based computational model as part of end-user applications that all types of "language workers" benefit from in practice.

While Upper Tanana verb structure is clearly complex and multimorphemic, it is possible to divide it into a more basic, three-zone structure, as has been common practice in many preceding descriptive studies of other Dene languages (cf. Kari (1975), Kari (1989)). The three primary zones—the disjunct domain at the left edge of the verb (template positions 12–7), the inner conjunct domain (positions 6–1), and the rightmost stem domain (position 0), cf. Table 5—are distinguished not only by their linear order within the verb, but also by a number of phonological and morphosyntactic criteria, including the set of phonological processes that are observed in each zone (Kari, 1975; Kari, 1989). These differences between zones have immediate consequences for computational modelling: using only general-purpose morphophonological adaptation rules that apply indiscriminately to all parts of the verb-word risks modifying prefix combinations in zones where those rules should not apply, and makes it difficult to model processes that operate only in one of these domains.

12	11	10	9	8	7	=	6	5	4	3	2	1	.	0	-1
Postpositional object	Postposition	Adverbial-derivational	Iterative	Incorporate	Distributive	=	Pronominal	Qualifier	Conjugation	Mode	Subject	Voice/valence	.	Stem	Suffix
Disjunct						=	Conjunct						.	Stem	

Table 5: The Upper Tanana verb, represented as a template

We thus follow Arppe et al. (2017) in including boundary symbols in our lexical representations to mark off the disjunct (=), conjunct (_), and stem (.) zones in our lexical entries, which can thus together be seen as defining a discontinuous lexical 'tier'. Where boundary markers are not explicitly indicated in a lexical entry, it is possible to conclude that no morphological material appears within the corresponding zone, and thus insert the missing boundary markers automatically. Thus, a lexical entry for a verb like na#D+kuyh 'vomit' (imperfective), which contains only the

stem *kuyh* and a single lexical disjunct prefix *na-*, can be specified as `na=kuyh`, and subsequently expanded by the model automatically into `na=_.kuyh`. This automatic insertion of unspecified boundary markers serves to align the form of our lexical entries more closely with common lexicographic practice in Dene linguistics (e.g., Kari (1990), Jetté and Jones (2000), among others), while avoiding one source of possible human error when such boundary markers are added manually.

Automatically inserting disjunct, conjunct, and stem zone marker symbols into lexical entries also provides an opportunity to flag the presence or absence of morphological material in each of these zones for use in other parts of the model. As an example, when this model encounters the lexical entry `na=kuyh` 'vomit', it not only inserts the appropriate additional boundary markers, but also sets a corresponding flag diacritic (`@P.PREFIX.OUTER@`) to indicate that a disjunct prefix is present in this verb. This information is linguistically important, as the forms that inflectional morphology takes in Dene languages vary considerably based on the kind of prefixes that precede it. Using flag diacritics allows the model to insert the correct set of inflectional allomorphs for the observed prefixation condition (e.g., selecting the allomorphs of the Ø-imperfective paradigm that appear with a preceding conjunct prefix when the conjunct prefix flag diacritic is set, or the allomorphs of that same inflectional paradigm that appear with a preceding disjunct prefix when the disjunct prefix flag diacritic is set, etc.).

This approach to capturing recurring patterns of inflectional allomorphy associated with different prefixation conditions allows for considerable simplification of the final computational model. In the case of Upper Tanana, each inflectional paradigm (e.g., the *Ø*-imperfective mentioned above) can thus be represented as a finite number of sets of allomorphs that appear under particular prefixation conditions (e.g., when preceded by no other prefixes, or preceded by one or more conjunct prefixes, or by one more disjunct prefixes). Within each such prefixation condition, different phonological forms for each subject person-number combination are hard-coded into the model (e.g., the first-person singular form of the *Ø*-imperfective paradigm when preceded by a disjunct prefix). In practice, these inflectional 'chunks' are treated as portmanteau morphemes that combine not only subject person and number, but also the voice-valence markers found in position 1 of the verbal template given above. Thus, a lexical entry for a verb tagged as taking the *Ø*-imperfective paradigm, which has no preceding prefixes, and has an **l-* voice-valence marker (cf. 2) would follow a series of continuation lexica to finally arrive at the following set of inflectional chunks, which give the forms of the core inflectional markers that appear in this context:

```
LEXICON 0-IPFV-L-AFF-NoPrefix
@U.SBJPERSON.1@@U.SBJNUMBER.SG@ag      #;
@U.SBJPERSON.2@@U.SBJNUMBER.SG@į̨l      #;
@U.SBJPERSON.3@@U.SBJNUMBER.SG@el      #;
@U.SBJPERSON.1@@U.SBJNUMBER.PL@el      #;
@U.SBJPERSON.2@@U.SBJNUMBER.PL@al      #;
@U.SBJPERSON.3@@U.SBJNUMBER.PL@el      #;
@U.SBJPERSON.INDEF@@U.SBJNUMBER.SG@el #;
```

On this approach, lexical entries in the model can be represented parsimoniously as lexical combinations of prefixes and stems that are associated with a particular aspect (e.g., imperfective, perfective) and a lemma (in most cases, the third-person singular imperfective form), and subsequently 'tagged' through continuation lexica for their transitivity (e.g., transitive, intransitive, etc.), conjugation class and aspect (e.g., *n*-perfective, *aa*-perfective, etc.), and voice-valence marker (i.e., Ø-, **d-*, **l-*, or **ł-*). Whole verb themes, such as for the verbs 'vomit' and 'cry' below, are captured by listing the distinctive stem forms and inflectional patterns associated with each aspect in which this verb appears in the corresponding *lexc* definitions:

```
LEXICON VerbThemes
etsüh[cry]:tsüh              INTR-0-IPFV-0;
etsüh[cry]:tsia'             INTR-aa-PFV-0;
etsüh[cry]:tsüh              INTR-aa-FUT-0;
etsüh[cry]:tsüü              INTR-u-OPT-0;

na'etkuyh[vomit]:na=kuyh     INTR-0-IPFV-D;
na'etkuyh[vomit]:na=kuyh     INTR-dh-PFV-D;
na'etkuyh[vomit]:na=kuyh     INTR-aa-FUT-D;
na'etkuyh[vomit]:na=kuyh     INTR-u-OPT-D;
```

Many core features of the Upper Tanana verbal lexicon are thus captured through tag-like continuation lexica: the imperfective form of 'cry' is marked as being intransitive (`INTR-`) and taking the Ø-voice/valence marker allomorphs of the Ø-imperfective paradigm (`0-IPFV-0`) by the lexical entry proceeding into the continuation lexicon associated with these features (`INTR-0-IPFV-0`). These continuation lexica lead through a series of unification flags that constrain the range of inflected forms that are ultimately recognized and produced by the model, as in the example below:

```
LEXICON INTR-0-IPFV-0
@U.VALENCE.INTRANSITIVE@
@U.ASPECT.IPFV@@U.TAMA.%0@@U.VV.%0@ Verbsuffixes;
```

The overall morphological model for Upper Tanana can be seen as a concatenation of several smaller models such as these, interleaving a discontinuous 'inflectional tier' (the 'chunks' mentioned above as well as morphemes for direct and postpositional objects) with a potentially discontinuous lexical tier for the verb theme. In terms of our computational implementation, this is achieved by first compiling the specifications of the verb theme as well as the applicable morphemes or morpheme chunks in the three slots representing the inflectional tier, as four separate FSMs. Then, the three inflectional tier FSMs are inserted using finite-state calculus within the lexical tier FSM, by replacing the special characters (`=`), conjunct (`_`), and stem (`.`) marking the inflectional boundaries with the entire corresponding inflectional FSMs. Careful use of flag diacritics shared by all of these components ensures that only forms licensed by the morphology are generated and accepted by the resulting composite model (cf. Arppe et al. (2017)).

Despite the morphological complexity evidenced by the template presented in Table 1, this approach to representing the basic morphological 'scaffolding' of Upper Tanana verb forms appears to be adequate for the purposes of computational modelling. A small number of morphophonological

rules apply to the model described above, handling regular processes of prefix vowel syllabification and epenthesis, as well as well-attested reductions in certain common prefix sequences. For example, the third-person plural subject marker *he-* may be realized (a) as *hih-* word-initially before /t, n, d/; (b) as *hi-* before /h/; or (c) as either *hi-* or *he-* (depending on dialect and/or speaker) when neither preceded nor followed by another vowel. These patterns can be modelled phonologically by treating the /e/ in this prefix as epenthetic, using the special character `E` to represent this vowel in the model (i.e., as `hE`). In the regular expression notation included in FST compilers (e.g. *foma*), these realizations can be captured through the following expressions:

```
def hEpenthesis E -> i h || .#. h _ [t|n|d];
def Epenthesis  E -> i || _ h         .o.
                E -> 0 || _ PrefixV   .o.
                E -> [i|e];
```

These few phonological rules, together with another set that deals with common reductions in prefix consonant sequences and instances of resyllabification, appear to be adequate in modelling much of the morphophonology associated with the 'core' verbal inflection implemented thus far.

3. Modelling verb themes

Utilizing the verb theme category system allows us to further increase the efficiency of the model. By tagging each verb theme for verb theme category, we predict much of its morphological behaviour, as described in section 1. This allows us to simplify the lexical entry for a verb theme. Instead of indicating that `na=kuyh` 'vomit' requires the Ø-Imperfective and *dh*-Perfective, we indicate that it is a member of the Conversive verb theme category, which is characterized i.a. by these two conjugation markers. This requires some adjustment to the structure of the lexicon, which would now be organized by verb theme category, transitivity, and voice/valence marker:

```
LEXICON Operative-Intransitive-0
etsüh[cry]:tsüh             IPFV ;
etsüh[cry]:tsia'            PFV ;
etsüh[cry]:tsüh             FUT ;
etsüh[cry]:tsüü             OPT ;

LEXICON Conversive-Intransitive-0
na'etkuyh[vomit]:na=kuyh    IPFV ;
na'etkuyh[vomit]:na=kuyh    PFV ;
na'etkuyh[vomit]:na=kuyh    FUT ;
na'etkuyh[vomit]:na=kuyh    OPT ;

LEXICON Motion-Intransitive-0
nihaayh[s.go]:haayh         IPFV ;
nihaayh[s.go]:shyah         PFV ;
nihaayh[s.go]:haaɬ          FUT ;
nihaayh[s.go]:shya'         OPT ;
```

This approach has two advantages. First, it minimizes lexical data entry by allowing us to only list the information unique to each verb theme—its prefixes and aspectual stem variation. The remaining information—conjugation markers, transitivity, voice/valence-marker—are all given in the heading of each section of the lexicon (i.e., through these continuation lexica and associated flag diacritics that they set). Second, and more importantly, this approach allows us to model significant aspects of the derivational system of Upper Tanana. By marking each verb theme category and any associated derivational prefix strings with their own flag diacritic (e.g. `@U.VTC.OPERATIVE@` for the operative verb theme category), we ensure that these forms of derivation will occur with all and only the members of this verb theme. Since many derivational prefix strings select their own conjugation markers, this further allows the model to override the specification of a given verb theme. We illustrate this with an example from the Motion verb theme category.

If we flag a verb theme such as Ø+haayh 's. go' as a member of the Motion verb theme category, we indicate that it has certain semantic (+punctual, +telic situation type, Momentaneous aspect) and morphological (e.g. requiring *n*-Imperfective and *n*-Perfective) characteristics. Any Motion verb theme is however also compatible (theoretically) with all 55 derivational strings associated with Motion verbs, such as *ski-* 'across', *ki-* 'climbing', or *da+tl'a* 'up onto shore', as well as certain aspectual and superaspectual derivations, some of which are shown in (1). As the example shows, each derivational string is specified for conjugation marker and stem aspect, which overrides the primary theme's specifications. All forms in (1) are in the third person singular perfective; only (1f) is imperfective.

(1) a. primary theme (n, n), Momentaneous
nɻɻshyah
's/he arrived'

b. ski# (n, n), Momentaneous
skinɻɻshyah
's/he went across'

c. ki# (Ø, aa), Momentaneous
ki'ɻɻshyah
's/he climbed up [e.g. a tree]'

d. da+tl'a# (Ø, dh), Momentaneous
datl'a'eeshyah
's/he went up onto the bank'

e. na#t+D (aa, ee), Perambulative
natetshyah
's/he was walking around (pfv.)'

f. – (aa, –), Progressive (ipfv. only)
aahaaɬ
's/he is/was walking along'

By flagging both prefix strings and verb themes for verb theme category, we can harness the enormous derivational power of this system for our model. Since all prefix strings in (1) are available in principle to all motion verb themes, our model can correctly generate derived forms from basic

themes as schematized in Table 6. Some semantic restrictions apply: while all of the forms in Table 6 are morphologically and semantically well-formed and are attested in Lovick's corpus, ki# 'up' cannot be combined with the verb theme Ø+mbeeyh 'swim'. The form *ki'įįmįį* 'she climbed up while swimming' generated by our model is not accepted by speakers, presumably for semantic reasons. Such exceptions will need to be specified manually.

Prefix string	L+tthiit 's. run'	Ø+mbeeyh 'swim'
ski	ski'iltthät	skiniįmiį
'across'	's/he ran across'	's/he swam across'
ki	ki'altthät	–
'up'	's/he ran up [a tree]'	–
da+tl'a	datl'a'eltthät	datl'a'eemiį
'onto bank'	's/he ran onto the bank'	's/he swam onto the bank'
na#t+D	natelttthät	natetmiį
'Peramb.'	's/he was running around'	's/he was swimming around'
–	altthäł	almbeeł
'Prog.'	's/he was running along'	's/he was swimming along'

Table 6: Derivation of motion verb themes

In total, there are fewer than 100 primary motion themes and about 55 derivational strings that can apply to motion themes. By flagging each primary theme as compatible with each derivational string, we allow our model to generate 5,500 derived verb themes: a sizeable portion of the Upper Tanana verb lexicon. We follow the same process for the other verb theme categories and their associated derivational strings.

The accuracy of this model can be evaluated by comparing the resulting forms against the Upper Tanana documentation record. Attested forms might be listed in the model and weighted above predicted forms. Given the relatively small size of the corpus, however—Lovick's narrative corpus comprises about 9,000 utterances—it is likely that our model will produce many forms that are not attested in the documentation. We will not be able to check all generated forms with a speaker, due to the severe endangerment of the language. Our solution to this is to compare predicted forms against their equivalents in related languages with substantially larger corpora and, importantly, larger lexical databases, such as Ahtna (Kari, 1990) or Koyukon (Jetté and Jones, 2000). Forms with cognates in those languages may then also be listed in our model, albeit with a lower weighting than forms attested in Upper Tanana. In this fashion, the richness of the inflectional as well as the derivational system of Upper Tanana can be made accessible to language learners even when they do not have access to a speaker.

4. Conclusion

The computational model of Upper Tanana presented here builds on and expands that designed by Cox et al. (2016) and Arppe et al. (2017) for Tsuut'ina, as part of a more general research program to model Dene and other Indigenous languages (cf. `altlab.artsrn.ualberta.ca`). Verb lexica structured by verb theme category, transitivity, and voice/valence marker allow our model to predict much of a verb theme's inflectional and derivational behaviour, which drastically reduces the amount of information that needs to be entered manually. By adding a derivational component to our model, we make the rich derivational system of this conservative Dene language accessible to language learners.

5. Acknowledgements

This work has been funded by the Social Sciences and Humanities Research Council of Canada (SSHRC) through a Connections Outreach Grant (611-2016-0207) and a KIAS Research Cluster Grant during 2015-2018 (University of Alberta). Lovick's research on Upper Tanana grammar has been funded as part of her International Polar Year Postdoctoral Fellowship at the University of Alaska Fairbanks (2006-2008) as well as by a Documenting Endangered Languages grant from the National Science Foundation (2012-2015, held jointly with Dr. Siri Tuttle).

6. Bibliographical References

Arppe, A., Cox, C., Hulden, M., Lachler, J., Moshagen, S. N., Silfverberg, M., and Trosterud, T. (2017). Computational modeling of the verb in Dene languages. the case of Tsuut'ina. In *Working Papers in Athabascan Linguistics ("Red Book" series)*, Fairbanks. Alaska Native Language Center.

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications, Stanford, CA.

Cox, C., Hulden, M., Silfverberg, M., Lachler, J., Rice, S., Moshagen, S. N., Trosterud, T., and Arppe, A. (2016). Computational modeling of the verb in Dene languages. the case of Tsuut'ina. In *Dene Languages Conference, Yellowknife, Northwest Territories, Canada, 6–7 June 2016*. http://altlab.artsrn.ualberta.ca/wp-content/uploads/2016/07/Tsuutina-FST-DLC-160607B.pdf.

David, C. (2017). *Teedlǫy t'iin naholndak niign: Stories by the Tetlin people. Expanded edition. Edited by Olga Lovick*. Alaska Native Language Center, Fairbanks.

Goddard, P. E., (1911). *Athapascan (Hupa)*, pages 85–158. Number 40 in Smithsonian Institution Bureau of American Ethnology Bulletin. Government Printing Office, Washington, DC.

Hoijer, H. (1945). The Apachean verb, part I: Verb structure and pronominal prefixes. *International Journal of American Linguistics*, 11(4):193–203.

Hulden, M. (2009). Foma: a finite-state compiler and library. In *Proceedings of EACL*, pages 29—32, Athens. Association for Computational Linguistics.

Jetté, J. and Jones, E. (2000). *Koyukon Athabaskan Dictionary*. Alaska Native Language Center, Fairbanks.

Kari, J. (1975). The disjunct boundary in the Navajo and Tanaina verb prefix complexes. *International Journal of American Linguistics*, 41(4):330–345.

Kari, J. (1979). *Athabaskan verb theme categories: Ahtna*. Alaska Native Language Center Research Papers, Number 2. Alaska Native Language Center, Fairbanks.

Kari, J. (1989). Affix positions and zones in the Athapaskan verb complex: Ahtna and Navajo. *International Journal of American Linguistics*, 55(4):424–454.

Kari, J. (1990). *Ahtna Athabaskan Dictionary*. Alaska Native Language Center, Fairbanks.

Li, F.-K. (1946). Chipewyan. In Cornelius Osgood et al., editors, *Linguistic structures of native America*, pages 398–423. The Viking Fund publications in anthropology No. 6, New York.

Lindén, K., Axelson, E., Hardwick, S., Silfverberg, M., and Pirinen, T. (2011). HFST - framework for compiling and applying morphologies. In *Proceedings of the Second International Workshop on Systems and Frameworks for Computational Morphology (SFCM)*, pages 67–85. Springer, Berlin, Heidelberg.

Rice, K. (2000). *Morpheme order and semantic scope*. John Benjamins, Amsterdam, New York.

Sapir, E. and Hoijer, H. (1967). *The Phonology and Morphology of the Navaho Language*, volume 50 of *University of California Publications in Linguistics*. University of California Press, Berkeley, CA / Los Angeles, CA.

Trosterud, T. (2006). Grammatically based language technology for minority languages. In *Lesser known languages of South Asia*, pages 293—316. Mouton de Gruyter, The Hague.

Tyone, M. (1996). *Ttheek'ädn Ut'iin Yaaniidą' Oonign': Old Time Stories of the Scottie Creek People. Stories told in Upper Tanana Athabaskan by Mary Tyone, Ts'ą' Yahnik.* Alaska Native Language Center, Fairbanks.

Expanding Abbreviations in a Strongly Inflected Language: Are Morphosyntactic Tags Sufficient?

Piotr Żelasko
AGH University of Science and Technology, Kraków,
www.dsp.agh.edu.pl,
pzelasko@agh.edu.pl

Abstract

In this paper, the problem of recovery of morphological information lost in abbreviated forms is addressed with a focus on highly inflected languages. Evidence is presented that the correct inflected form of an expanded abbreviation can in many cases be deduced solely from the morphosyntactic tags of the context. The prediction model is a deep bidirectional LSTM network with tag embedding. The training and evaluation data are gathered by finding the words which could have been abbreviated and using their corresponding morphosyntactic tags as the labels, while the tags of the context words are used as the input features for classification. The network is trained on over 10 million words from the Polish Sejm Corpus and achieves 74.2% prediction accuracy on a smaller, but more general National Corpus of Polish. The analysis of errors suggests that performance in this task may improve if some prior knowledge about the abbreviated word is incorporated into the model.

Keywords: abbreviation expansion, morphosyntactic tags, strongly inflected langauges

1. Introduction

In this paper, we address the problem of recovery of morphological information lost in abbreviated forms, focusing on highly inflected languages. Significance of the problem can be shown in the domain of text normalization, especially in cases of expanding numerals and abbreviations to their full forms. Although in English the translation of token "14" to "fourteen", or token "yr." to "year" appears trivial, it is not so in a highly inflected language. For example in Polish, the abbreviation "r." could be expanded into one of {*rok*, *roku*, *rokowi*, *rokiem*}, all of which mean "year" in English, but are inflected differently, based on the context in which they are used. Correct choice of the grammatical number and case is essential towards successful text normalization, which is an indispensable part of any text-to-speech (TTS) system (Sproat and Jaitly, 2016), and can also be useful in text preprocessing for language model (LM) training (Pakhomov, 2002).

Under the assumption that an abbreviation is not ambiguous, the conversion of a token into its abbreviated form can be seen as a removal of morphological information from the token. Nonetheless, a person (e.g. a native speaker) reading the text is able to deduce the information missing from the abbrevation by using the context. Furthermore, we suspect that it is possible to abstract away from concrete words and infer the missing morphological information based only on morphological and syntactic properties of the context. Consequently, we formulate a hypothesis, that based purely on the morphosyntactic information present in other words in the sentence (i.e. the context), the correct morphosyntactic tag can be inferred for the abbreviated word in a highly inflected language.

Given recent major achievements of deep neural networks (DNN) in Natural Language Processing (NLP) tasks (Sundermeyer et al., 2012) (Huang et al., 2015), we apply a bidirectional recurrent neural network (RNN) (Graves and Schmidhuber, 2005) based on Long Short-Term Memory (LSTM) cells (Hochreiter and Schmidhuber, 1997) to the task of missing morphosyntactic tag prediction. The network is trained on a large collection of Polish texts analysed with a morphosyntactic tagger and evaluated on a manually annotated corpus in order to verify the hypothesis. To the best of our knowledge, no previous attempt has been made in order to infer the correct form of an abbreviated word with an RNN, while using solely morphosyntactic information of its context within the sentence.

In section 2. we present previous relevant work in the area of text normalization. Section 3. describes the problem of inflected expansion of abbreviations and the DNN architecture used to solve it. The data sets, experimental setting and results are shown in section 4. Finally, we conclude our work in section 5.

2. Related Work

Although there have been several studies regarding text normalization, it can be safely stated that it did not receive as much attention as some other areas of NLP. In (Pakhomov, 2002), the author employed a Maximum Entropy (ME) model in order to disambiguate and expand the abbreviations present in medical documents. In order to provide the labels for supervised training of the model, sentences containing non-abbreviated forms were identified and treated as if the term had been abbreviated. The model was then trained to recognize correct expansion based on a context of 7 words on each side of the abbreviation. Even though in our work we adopted similar approach to training data collection, we use the morphosyntactic tags instead of words and utilize the full sentence instead of a fixed context window. Furthermore, our goal is to infer morphological information and not to disambiguate the abbreviation.

Similar efforts have been undertaken in the task of numerals normalization in the Russian language, which is also highly inflected. In (Sproat, 2010), the author had investigated the viability of n-gram, perceptron and decision list models and found that a trigram model slightly outperforms the others. He also suggested further investigation into application of

more complex discriminative models.

A different branch of text normalization is focused on abbreviation discovery. One approach includes a character-level Conditional Random Field (CRF) model used to generate most likely deletion-based abbreviations for words, which are then reverse-mapped and disambiguated using an additional language model (Pennell and Liu, 2011). In (Roark and Sproat, 2014), the authors identified abbreviations by noticing that in a large data set, they appear in the same contexts as the non-abbreviated words. To expand them, they perform a fusion of N-gram and Support Vector Machine (SVM) models which is characterized by a low false alarm rate.

Recently, there has been growing interest in text normalization seen as a machine translation task - in order to foster more research in this direction, a challenge with an open data set has been published (Sproat and Jaitly, 2016). Regardless of promising results, the authors observed that the RNN tend to fail in some cases in quite a weird manner - such as translating abbreviated *hours* as *gigabytes*. Our approach differs from the one presented in (Sproat and Jaitly, 2016) in the way the RNN is used - instead of constructing a sequence to sequence (seq2seq) model, we use a many-to-one mapping in our network in order to identify only one missing piece of information at a time.

3. Methods

Let us consider a sentence $W = \{w_i\}$, where w_i is the i-th word of the sentence. Each word (lemma) w_i can be described as a combination of a basic form (lexeme) l_i and morphosyntactic information m_i, expressed as a set of particular morphosyntactic tags. These tags inform about both morphological and syntactical properties of the word, such as grammatical class, number, case, gender, etc.

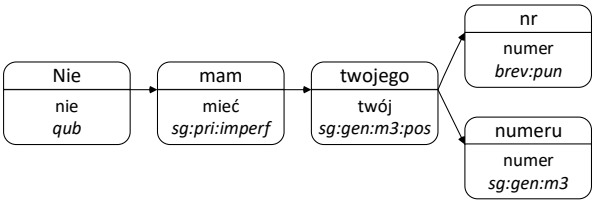


Figure 1: A comparison of abbreviated and full form of the lemma *numeru* in Polish equivalent of English sentence *"I don't have your number"*. The upper block contains lemmas and the lower block their corresponding lexemes and morphological tags.

In a sentence without any abbreviations, each lemma explicitly presents its morphosyntactic information through inflectional morphemes, which are composed together with the lexeme (e.g. lexeme *pasture* composed with morpheme *s* results in lemma *pastures*, which informs that the grammatical number is plural). However, this is not the case in abbreviated words, which are not subject to inflection, regardless of their context. Nonetheless, the obfuscated morphosyntactic properties of an abbreviation can usually be deduced by a native speaker of an inflectional language through analysis of the context. Consequently, it can be said that abbreviations, presented within a context, contain *implicit* morphosyntactic information. This phenomenon is illustrated in a sentence[1] in figure 1, where the abbreviated form is shown to have unknown morphological properties (i.e. a tag *brev:pun*, which indicates an abbreviation followed by punctuation), in contrast to the full form.

In order to predict this implicit morphosyntactic information, we propose the application of an RNN - in particular, a bidirectional LSTM network - which analyses the whole sentence.

The input of the network is a sequence of morphosyntactic tags m_i, each one corresponding to the word w_i from the original sentence W. The unknown tag of the abbreviation is substituted by a special token, which indicates that this tag is to be inferred. The input layer is then connected to an embedding layer, which casts the tags into a low-dimensional real-valued vector, similarly as it is done in case of regular words in modern neural language models (Mikolov et al., 2013).

The embeddings are then fed into several recurrent layers. After the final recurrent layer, the last output of the forward net and the backward net are concatenated and serve as input to a fully-connected layer, followed by a softmax layer, which infers the missing tag in the sentence. The architecture is shown in figure 2.

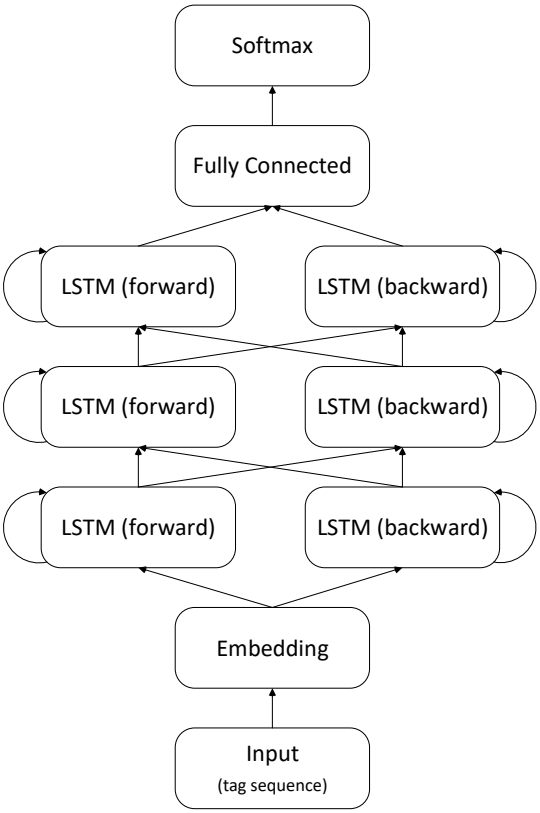


Figure 2: Tag prediction neural network architecture.

[1] The usage of this abbreviated form in this context is not typical in Polish writing, but may be representative of an abbreviation used in a short message sent to a peer.

4. Experiment

In this section we describe the data sets, the specific model architecture settings used in the experiment and the results.

4.1. Data

In order to evaluate the effectiveness of the proposed method, we used two publicly available Polish text corpora - the Polish Sejm Corpus (PSC) (Ogrodniczuk, 2012) and a subset of the National Corpus of Polish (NCP) (Przepiórkowski et al., 2008) (Przepiórkowski et al., 2010). The PSC is a large collection of speeches and interpellations of parliamentary sessions of the Polish government. Although PSC is provided along with automatically extracted morphosyntactic tags, we used the WCRFT tagger (Radziszewski, 2013) to extract them from plain PSC text, because of its ability to predict the tags of numerals written as digits. We used default tagger model supplied with the WCRFT software, which was trained on the NCP. The PSC corpus was split and served as training (99%) and validation (1%) data sets.

As the evaluation data set, we used the publicly available 1-million word subset of the NCP. The main feature of this subset is its manual annotation. Due to the fact that NCP was designed to be representative of the Polish language in general, the corpus consists partly of the PSC, as well as books, newspapers and transcripts of informal speech. Due to its richness, we encountered some tags not found in our annotation of the PSC, which resulted in a total of 321 (8%) of sentences being removed from the evaluation set.

Even though the NCP corpus could be considered better candidate for training set due to its manual annotations, we chose to train on PSC, which is a much larger corpus, more suitable for training a DNN. We also wanted to make the evaluation criterion more strict, since in this scenario, we also evaluate how well does the model generalize beyond the domain of the Polish parliament language.

Data set	Corpus	Sentences	Words
Train	PSC	521251	10684799
Validation	PSC	5265	107511
Evaluation	NCP	3491	71895

Table 1: Data sets used in the experiment.

The morphosyntactic information is typically described by a particular tagset. In this work, we adopted the NKJP tagset (Przepiórkowski, 2009), which is being used in the NCP. A major difficulty is the lack of availability of the true morphosyntactic tags for the abbreviations - one of the NKJP tagset assumptions was to manually annotate abbreviations with tags *brev*:*pun* and *brev*:*npun*, which indicate an abbreviation followed/not followed by punctuation, respectively. Since the true tags are available neither in automatic nor manual annotation, we select 34 Polish words which are often abbreviated 4.1., look up their inflected forms in the Polimorf morphological dictionary (Wolinski et al., 2012), and gather sentences which contain at least one of these words. As a result, we obtain sentences where an abbreviation might have been possible, and consequently acquire labels (the true morphosyntactic tags) for a supervised machine learning algorithm. We did not consider abbreviations of multiple words in this work.

The size of each data set is presented in table 4.1.. Sentences containing more than one abbreviation are used repeatedly with different word abbreviated each time in order not to introduce additional complexity into the experiment (which would involve handling more than one *unknown* tag by the network). Also, in order to reduce the training time, we used only sentences of length up to 30 words.

Abbrev.	Base form	Translation	Freq. [%]
r	rok	year	4.29
woj	województwo	voievoidship	1.44
pkt	punkt	point	1.32
wyd	wydanie	edition	0.46
art	artykuł	article	0.41
ok	około	around	0.39
godz	godzina	hour	0.35
poz	pozycja	position	0.30
jedn	jednolity	uniform	0.24
zł	złoty	zloty (PLN)	0.23
tys	tysiąc	thousand	0.23
mln	milion	million	0.15
nr	numer	number	0.14
prof	profesor	professor	0.11
św	święty	saint	0.11
proc	procent	percent	0.10
ul	ulica	street	0.08
ks	ksiądz	priest	0.06
mld	miliard	billion	0.05
ozn	oznaczenie	mark	0.05
ha	hektar	hectare	0.04
dr	doktor	doctor	0.04
gen	generał	general	0.04
ust	ustęp	paragraph	0.03
l	litr	liter	0.03
km	kilometr	kilometer	0.03
m	metr	meter	0.03
im	imię	name	0.02
del	delegowany	delegate	0.02
sygn	sygnatura	signature	0.02
par	paragraf	paragraph	0.02
kg	kilogram	kilogram	0.01
rep	repertorium	repertory	0.003
cm	centymetr	centimeter	0.001

Table 2: Abbreviations expanded in the experiment, along with their base form and frequency relative to the total word count in PSC corpus.

4.2. Experimental setup

We have tested several configurations of the neural network architecture and present the one which achieved the best results. In the training data, we found 950 different morphosyntactic tags (including the special *unknown* tag) and feed them in one-hot-encoded form to the embedding layer, obtaining 32-dimensional real-valued vectors. The bidirectional LSTM layers have 64 hidden units for each direction. After each recurrent layer, we apply dropout (Srivastava et al., 2014) with a factor of 0.2. After the last recurrent layer,

the fully connected layer has 128 hidden units followed by dropout with a factor of 0.5 and ReLU activation (Nair and Hinton, 2010). Finally, the softmax layer has 257 output units - this is different from the input layer due to the fact that the abbreviations in our experiment are mostly nouns and are described using only a subset of the original tagset. The last two layers are additionally regularized with weight decay of 0.0005.

The training was performed using the error backpropagation algorithm (Rumelhart et al., 1988) with cross-entropy criterion and Adam optimizer (Kingma and Ba, 2014) with the same settings of learning rate and momentum parameters as suggested in the paper. Final parameters of the model were selected from the epoch with minimal validation loss. We used the TensorFlow (Abadi et al., 2015) and Keras (Chollet, 2015) frameworks for the experiment.

The correctly inflected abbreviation expansion is obtained by a lookup of the base form and predicted morphological tag in a morphological dictionary - in our case, the Polimorf.

4.3. Results

After 29 epochs of training, the DNN achieved 84.5% accuracy on the training set, 85.7% accuracy on the validation set and 74.2% accuracy on the evaluation set. We hypothesize that the major factor behind the degradation of performance on the evaluation set (in contrast to consistent performance on the train and validation sets) is the difference between tag distribution in both corpora. As mentioned before, the NCP sentences were collected from several different language domains, such as casual language, letters, newspapers and parliament speeches, while the PSC consists entirely of the kind of language encountered in the parliament. It is likely that the tag distributions are different enough to introduce generalization errors, when the classifier was trained on a single domain exclusively.

By contrast, when a simple baseline which predicts the most frequent tag for each abbreviation is assumed, it achieves the following accuracies: training set - 42.8%, validation set - 42.6%, evaluation set - 40.3%. The result obtained by the DNN is significantly stronger and showcases the importance of the abbreviation context, which is not surprising. These results constitute significant evidence that morphosyntactic information is sufficient in many cases.

We would like to showcase and discuss some of the problematic cases in which the DNN failed to predict properly in the evaluation set. First of all, we noticed that about 10% of accuracy is lost due to confusing singular and plural grammatical number. While most of these are unacceptable, there are a few mistakes which are less severe - e.g. "*2.7 l*" should have been translated as "*2.7 litra*" (*2.7 liter*), but the output was "*2.7 litrów*" (*2.7 liters*). Also, 49 mistakes (about 5% of accuracy lost) concerned the inflection of the *procent* word, expanded from the % symbol. It appears that the network learned the rule that in Polish, a noun which has its quantity specified (e.g. *5 centimeters*), is generally inflected with case and number, even though *procent* is an exception to this rule. It follows reason, since the network had no way of knowing which word was used and the exceptional case is less frequent than the general one.

Another category of errors is related to grammatical gender. The Polish words *ulica* (*street*) and *godzina* (*hour*) are feminine, but in 40 cases their abbrevations (*ul.* and *godz.*) were classified as masculine. We suspect that the nature of this error is similar to the previous ones - the network had no way of acquiring prior knowledge about the correct gender of these words.

5. Conclusions

We discussed the problem of inflected abbreviation expansion and investigated the viability of expansion based on morphological tags of the abbreviation context combined with lookup in a predefined abbreviation dictionary. We successfully applied a bidirectional LSTM in this task and achieved a reasonable accuracy in an experiment conducted on two Polish corpora. Given the error analysis of our model, we conclude that the morphosyntactic information of the context is not sufficient to deduce the morphosyntactic tags for the expanded abbreviation - although it works in a significant number of cases, prior knowledge about factors such as base form or grammatical gender of the expanded abbreviation is required for correct prediction. We expect that incorporation of this prior knowledge in the model will yield significantly better expansion accuracy.

6. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Chollet, F. (2015). Keras. `https://github.com/fchollet/keras`.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5):602–610.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In *Proceedings*

of the 27th international conference on machine learning (ICML-10), pages 807–814.

Pakhomov, S. (2002). Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical texts. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 160–167. Association for Computational Linguistics.

Pennell, D. and Liu, Y. (2011). Toward text message normalization: Modeling abbreviation generation. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5364–5367. IEEE.

Roark, B. and Sproat, R. (2014). Hippocratic abbreviation expansion. In *ACL (2)*, pages 364–369.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):213–220.

Sproat, R. (2010). Lightly supervised learning of text normalization: Russian number names. In *Spoken Language Technology Workshop (SLT), 2010 IEEE*, pages 436–441. IEEE.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Sundermeyer, M., Schlüter, R., and Ney, H. (2012). LSTM neural networks for language modeling. In *Interspeech*, pages 194–197.

7. Language Resource References

Ogrodniczuk, M. (2012). The Polish Sejm Corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2219–2223.

Przepiórkowski, A., Górski, R. L., Lewandowska-Tomaszyk, B., and Lazinski, M. (2008). Towards the National Corpus of Polish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC-2008)*.

Przepiórkowski, A., Górski, R. L., Lazinski, M., and Pezik, P. (2010). Recent developments in the National Corpus of Polish. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC-2010)*.

Przepiórkowski, A. (2009). A comparison of two morphosyntactic tagsets of Polish. pages 138–144. Citeseer.

Radziszewski, A. (2013). A tiered CRF tagger for Polish. In *Intelligent tools for building a scientific information platform*, pages 215–230. Springer.

Sproat, R. and Jaitly, N. (2016). RNN approaches to text normalization: A challenge. *arXiv preprint arXiv:1611.00068*.

Wolinski, M., Milkowski, M., Ogrodniczuk, M., and Przepiórkowski, A. (2012). Polimorf: a (not so) new open morphological dictionary for Polish. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 860–864.

A High-Quality Gold Standard for Citation-based Tasks

Michael Färber[1,2], Alexander Thiemann[1], Adam Jatowt[2]
[1] University of Freiburg, [2] Kyoto University
michael.faerber@cs.uni-freiburg.de, mail@athiemann.net, adam@dl.kuis.kyoto-u.ac.jp

Abstract

Analyzing and recommending citations within their specific citation contexts has recently received much attention due to the growing number of available publications. Although data sets such as CiteSeerX have been created for evaluating approaches for such tasks, those data sets exhibit striking defects. This is understandable when one considers that both information extraction and entity linking, as well as entity resolution, need to be performed. In this paper, we propose a new evaluation data set for citation-dependent tasks based on arXiv.org publications. Our data set is characterized by the fact that it exhibits almost zero noise in its extracted content and that all citations are linked to their correct publications. Besides the pure content, available on a sentence-by-sentence basis, cited publications are annotated directly in the text via global identifiers. As far as possible, referenced publications are further linked to the DBLP Computer Science Bibliography. Our data set consists of over 15 million sentences and is freely available for research purposes. It can be used for training and testing citation-based tasks, such as recommending citations, determining the functions or importance of citations, and summarizing documents based on their citations.

Keywords: Citations, References, Scholarly Data, Citation Recommendation, arXiv.org, Digital Libraries

1. Introduction

Many tasks concerning digital libraries deal with citations mentioned in scientific texts. These include citation recommendation, which deals with recommending relevant citations for a given citation context. For instance, for the sentence "Models of linear logic have provided a fresh point of view and new intuitions that were applied to traditional fields of study, such as game semantics [?]," the aim would be to recommend appropriate citations such as (Abramsky and McCusker, 1995). Citation recommendation has turned out to be a task attracting increased interest and with significant impact due to the rapidly growing numbers of scientific publications released each year.

Approaches for citation recommendation are mostly evaluated by removing all citations in the considered publications and by letting the tested approach "re-predict" the publications which were cited. In order to allow large-scale experiments, some evaluation data sets have been created, such as CiteSeerX (Caragea et al., 2014). These data sets are sizeable in terms of the number of publications and citation contexts.[1] However, all of them have considerable drawbacks, making it difficult to use those data sets as realistic evaluation data sets – as partially pointed out by (Roy et al., 2016). Two of those drawbacks are (1) *the citation contexts are very noisy* and (2) *there is no interlinking or annotation of citations in the text with a noise-free structured representation of the cited publications* (especially across documents). To the best of our knowledge, CiteSeerX is the only data set which provides not only information about references between papers, but also extracted citation contexts for each citation, thereby solving problem (2) to some extent. However, the citation contexts are very noisy (see Sec. 2.), thereby it suffers from the drawback (1). Furthermore, the CiteSeerX data set contains not only publications, but any manuscripts, since it is built based on crawling web pages. A cleaner data set was created by Carageas et al. (Caragea et al., 2014) based on CiteSeerX, but this data set still does not reach the desired quality for a real-world evaluation of citation recommendation and other tasks.

In this paper, we propose a newly-created gold standard data set for citation-based tasks. This gold standard is based on all computer science papers in arXiv.org and is of very high quality: (1) the extracted sentences are almost always clean and complete, and (2) 100% of the citations in the text are linked to their correct publications. This is due to the fact that for each citation in TEX we know which cited publication is referenced and that we will not miss any citations due to explicitly given `cite` commands. Besides the fact that arXiv.org is a valuable source, arXiv.org is also being used with increasing frequency,[2] making our data set creation approach even more promising in the future.

This paper details how we created the data set and how it can be used. The data set files and associated key figures can be obtained for research purposes at `http://www.citation-recommendation.org/publications/`.[3]

2. Existing Data Sets

CiteSeerX can be regarded as the most frequently used evaluation data set for citation-based tasks. The first version of CiteSeerX was published in 1998 under the name CiteSeer (Giles et al., 1998) and presented a sample of 5,000 documents. For our investigation, we use the snapshot of the entire CiteSeerX dataset as of October 2013,

[1] For instance, the CiteSeerX version by (Huang et al., 2015) consists of over 1M papers and 10.8M citation contexts. As of May 2013, CiteSeerX had up to 52M citations from up to 2M documents (Caragea et al., 2014).

[2] See `https://arxiv.org/year/cs/17`, `https://arxiv.org/year/cs/16`, and so forth.

[3] Note that most articles in arXiv are submitted with the default arXiv license which grants arXiv a perpetual, non-exclusive license. For our data set, we follow this licensing and refer to `https://arxiv.org/help/bulk_data`

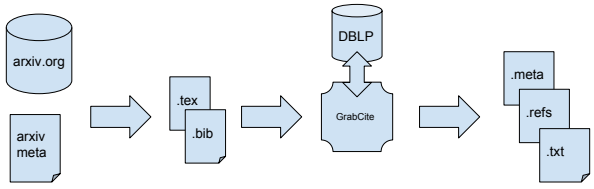


Figure 1: The pipeline used for creating our data set.

published in 2015 by (Huang et al., 2015). This data set consists of 1,017,457 papers, together with 10,760,318 automatically extracted citation contexts. Based on this data set, we can outline the most significant drawbacks of CiteSeerX as follows (cf. also (Roy et al., 2016)):

1. The provided meta-information about cited publications is often not accurate. In particular, the information about the title, the authors, and the venue of cited publications are sometimes incorrectly segmented. Furthermore, a publication's title can be mixed with information about the venue or with the header of the first content paragraph.
2. The citation contexts can contain noise from non-ASCII characters, formulas, section titles, missed references and/or other "unrelated" references, and do not begin with a complete word; instead, a cut-off at a fixed character length position is used.
3. The actual citation in a context is marked with delimiters ("=–=" and "–=–"), but sometimes characters or symbols from preceding words are included.
4. It also seems to be rather difficult to recover the original text of a given paper – meaning that one is essentially limited by the the length of citation context.

Beside CiteSeer and CiteSeerX, there are other collections of scientific publications. Among them are the ACL Anthology corpus (Bird et al., 2008) and Scholarly Dataset 2 (Sugiyama and Kan, 2015). Note that these data sets only contain the publications themselves, typically in PDF format. Therefore, using such data sets for citation-based task evaluation is troublesome, since one must preprocess the data (i.e., (1) extract the content without introducing too much noise, (2) build global identifiers for cited papers, and (3) annotate citations with those identifiers.) Last but not least, data sets for evaluating paper recommendation tasks, such as CiteULike,[4] only provide information on a document level, but no citation contexts.

3. Data Set Creation

The workflow for creating the proposed corpus of arXiv.org publications annotated with citations is presented in Fig. 1. The basic procedure is as follows: We first downloaded all arXiv source files, which are provided by arXiv via Amazon S3.[5] The provided data consists of multiple tar file bundles. Each tar file contains the files of the individual publications. A paper is either a single LaTeX-file, or a compressed folder containing a LaTeX file (at least one), optionally a bibtex file and other resources. We then use the metadata API of arXiv.org[6] to determine the domain of the paper (e.g., "CS" for computer science). In this data set, we only include papers of the computer science domain in order to be able to retrieve meta-information about those papers from DBLP.

The next step consists of processing each individual paper stored either in a single file or in a compressed folder. In the single file case, we directly parse the LaTeX file into a simple abstract syntax tree (AST). Otherwise, we uncompress the folder, identify all bibtex and LaTeX files, and parse them as described in Section 3.1. Then, we traverse both ASTs (the LaTeX AST and the bibtex AST if available) to extract title, text body, references and citations from the paper and represent it in a structured way.

Having obtained the references of each paper, we attempt to generate a globally unique (descriptive) ID for all references and all papers. In the optimal case, this is the DBLP URL of the paper/reference. This step is outlined in Sec. 3.2.

After having obtained all identifiers for all citations as the "offline step," we replace all citation markers (e.g., "`\cite{FooBar}`") with the global publications' identifiers and split the body text of all publications into single sentences (See Sec. 3.3.). For each publication, its sentences (annotated by identifiers) are written to a plain text file. In total, three files were created for each considered publication: A plain text file containing all sentences with global citation identifiers (each sentence on one line), a file with meta-information about this paper, and a file with mappings between the global citation identifiers (used in citations) and the titles of the cited publications (as written in the citing document). The full data format is described in Sec. 5.

In the following sections, we provide more details concerning key steps of our pipeline. The full pipeline is implemented in Haskell in our tool GrabCite, which is freely available on GitHub.[7] Note that in the following, all code snippets are simplified for clarity and brevity.

3.1. Parsing TeX

Working directly on TeX files instead of PDF enables us (1) to know with the utmost certainty the corresponding reference for each citation, and (2) to not miss any citations.

However, using TeX is non-trivial: LaTeX is a very complicated format to parse (Knuth, 1984). Among other things, this is due to the fact that LaTeX is fully customizable and programmable (cf. Turing completeness). Thus, few LaTeX parsers are sufficiently accurate and fast while producing an accessible AST for further processing. For example, the existing Haskell libraries like the popular pandoc (Krijnen et al., 2014) or HaTeX[8] quickly failed on most arXiv documents in our experiments. Most other tools for LaTeX parsing invoke the LaTeX-Engine and work on DVI outputs.[9] Math constructs such as equations are hard to write linearly in text and, moreover, they are not needed for tasks such

[4] See `http://citeulike.org/`.

[5] See `https://arxiv.org/help/bulk_data`.

[6] See `https://arxiv.org/help/oa/arXiv_meta_format.html`.

[7] See `https://github.com/agrafix/grabcite`.

[8] See `https://github.com/Daniel-Diaz/HaTeX`.

[9] See, for instance, TeX4ht, `http://www.tug.org/tex4ht/`.

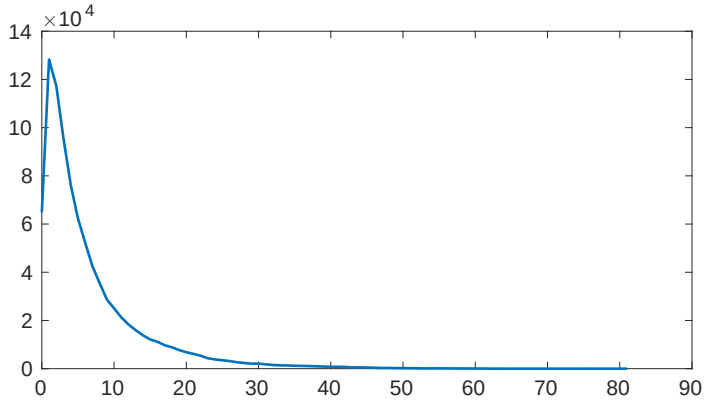


Figure 2: Distribution of time differences between the citing papers and the cited papers in years in our data set.

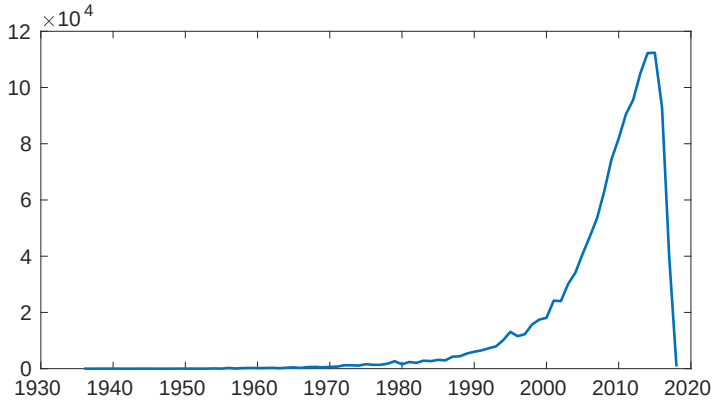


Figure 3: Distribution of the publication dates of all references in our data set.

as citation recommendation. Indeed, mathematical expressions rather disturb the generic learning of machine learning models. Therefore, we ignore math constructs and similar content for our needs.

Our TEX parser is implemented using parser combinators from the Haskell megaparsec library (Karpov, 2015). With the parsed AST at hand, we extract title, text bodies, and references and represent those in a structured way.

All references are written to a map, with the citation key as key and the title and authors as value. The title of the paper is also trivially extracted from the title LATEX command.

Evaluations showed that our extraction method currently fails on 24,762 of all 115,040 TEX input files (21.5%). Multiple factors account for these failures:

- The corresponding TEX input file only includes a PDF; hence, there is no raw content.
- Our heuristic algorithm picked the wrong TEX file from a zip archive.
- Our TEX parser fails due to unimplemented features in our parser.
- The TEX is invalid.

Note, however, that failures result in empty output files. Hence, the high quality of our gold standard is maintained.

3.2. ID Generation

In the ID Generation step, we want to replace local citation markers (e.g., `\cite{FooBar}`) and their associated reference with global identifiers (e.g., `DBLP:http://dblp.org/rec/journals/mscs/Berline06`). We also want to annotate the source papers with DBLP URLs in order to have meta-information about them, as well as graph-based statistics. To obtain the DBLP URL for a given title or reference, we generate a search string for DBLP by tokenizing the input and by using the first n words longer than two characters, thereby increasing n until we have 40 or more characters. We also look for year numbers and include them in the query. In our experiments, this returned the most accurate search results, as noisy words were removed and the query strings were not too long. If the search query returns a result, we use that. Otherwise, we query our own full text search index[10] of the DBLP XML dump[11] (Ley, 2009) which indexes titles and authors. We found that there are some cases where the DBLP API search does not return any results, but our custom full text search does. This full text search is based on PostgreSQL's built in `tsquery` and `similarity` functionality. If this still does not return any meaningful results, we generate as a fallback option our own global ID. First, we look for identifiers like the DOI or the arXiv.org ID to use, and if those are not detectable, we generate an identifier by extracting all words longer than two characters, sorting them, and taking the first 5 in concatenation. In Section 4., we outline the distribution of used references.

[10] See `https://github.com/agrafix/papergrep`.
[11] As of February 15, 2018.

3.3. Sentence Tokenizing

The final step is breaking the input into sentences. We implemented a custom sentence-splitting step due to the fact that many existing sentence tokenizers like `sent_tokenize` from NLTK (Loper and Bird, 2002) became confused by our global citation identifiers. Our sentence splitting uses heuristics to identify abbreviations, numberings, and ellipses, and can correctly handle our global citation markers. It also uses some basic metrics such as word count, character count and punctuation-to-character ratio to detect invalid sentences and to remove them.

4. Data Set Key Figures

Our data set, based on all arXiv.org publications in the computer science domain published until December 31, 2017, contains 90,278 papers. 62,337 (69%) of the papers could be found on DBLP and have been assigned the corresponding DBLP URL in the meta file. We extracted 15,530,204 sentences, resulting in 172 sentences per paper on average. 1,822,836 (11.7%) sentences contain at least one reference. All papers reference 277,227 unique papers using 2,448,826 citation markers in total (i.e., on average 27.1 citation markers per citing paper). Of these references, 962,084 could be found on DBLP and we could assign them a DBLP URL. Furthermore, the 90,278 citing papers cited 18,045 papers which are already in our arXiv data set (i.e., within-arXiv citations; in total, 153,555 single citations), while 259,182 (unique) cited papers are outside of our arXiv data set. For this calculation, we only considered papers with DBLP URLs, so that the value is likely to be under-approximated.

The temporal difference between a citing paper and a cited paper (see Fig. 2) is on average 6.7 years. For over half of all citations (53.1%), the cited paper is at most five years older than the citing paper. The largest gap is 81 years, with the oldest paper referenced having been written in 1936.

In Fig. 3, we show the distribution of the publication dates of all cited papers, with the oldest papers (from 1936) on the left and the most recent papers (from 2018) on the right. In total, 269,194 different authors and 1,489 different publication venues are referenced, with the most popular venues being *Computing Research Repository (CoRR)*[12] (citation count: 67,291) and *IEEE Trans. Information Theory*[13](citation count: 41,436). All mentioned key figures are available online.[14]

5. Data Set Format

The data set is provided as a compressed folder. The folder contains three documents per processed paper: a `.txt`, a `.meta`, and a `.refs` file. The name of each file corresponds to the paper's arXiv.org identifier.[15]

Listing 1: Example excerpt of an output .txt file.

```
In order to parallelize SGD, the standard
    approach is to employ minibatch
    training, which samples multiple
    examples uniformly at each step.
============
The uniformly sampled minibatch stochastic
    gradient is an unbiased estimation of
    the true gradient <DBLP:http://dblp.org
    /rec/conf/icml/Zhang04> <DBLP:http://
    dblp.org/rec/conf/aistats/RakhlinSS13>
    <DBLP:http://dblp.org/rec/conf/icml/
    Shamir013> <DBLP:http://dblp.org/rec/
    journals/jmlr/DuchiS09>, but the
    resulting estimator may have relatively
     high variance.
============
Throughout this paper, we will denote <
    formula> as <formula> for simplicity.
============
[..]
```

Listing 2: Example excerpt of an output .refs file.

```
DBLP:http://dblp.org/rec/conf/icml/ZhaoZ15;
    Peilin Zhao and Tong Zhang. Stochastic
    optimization with importance sampling.
    , abs/1401.2753, 2014.;
DBLP:http://dblp.org/rec/conf/icml/Zhang04;
    Tong Zhang. Solving large scale linear
    prediction problems using stochastic
    gradient descent algorithms. In ICML,
    2004.;
[..]
```

The `.txt` file contains all sentences extracted from the original paper, with local citation markers replaced with our global citation markers. There is one sentence per line, followed by a line containing a separator as shown in Listing 1. This allows the files to be easily skimmed by a human reader while also remaining optimal for machines parsing. Formulas and variables entered in math mode are represented by a `<formula>` token. Figures, tables and other listings, as well as the corresponding captions of the original input, are ignored and cannot be found in the output file.

The `.refs` file (see an example in Listing 2) contains a delimiter-separated dictionary mapping all global citation markers to their original reference descriptions. This al-

Listing 3: Example excerpt of an output .meta file.

```
{ "url":"http://dblp.org/rec/journals/corr/
    ZhaoZ14b"
, "authors":["Peilin Zhao","Tong Zhang"]
, "title":"Accelerating Minibatch
    Stochastic Gradient Descent using
    Stratified Sampling."
}
```

[12] See `https://arxiv.org/corr/home/`.

[13] See `http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?reload=true&punumber=18`.

[14] See `http://www.citation-recommendation.org/publications/`.

[15] See `https://arxiv.org/help/arxiv_identifier`

lows users of the data set to search for the paper in other sources if a DBLP identifier could not be determined by our processing pipeline.

The `.meta` file contains a JSON Document which is generated from data extracted from the paper merged with metadata returned from a search for the document in DBLP. It contains basic metadata such as the title of the paper, the authors of the paper and the DBLP URL of the paper. Note that the DBLP URL is very useful, as it allows users to download more context and metadata corresponding to the paper. For example, we can obtain BibTeX entries, RDF triples and other XML data for each paper using the provided URL. An example of a `.meta` file can be seen in Listing 3.

6. Conclusions

Approaches for citation-based tasks, especially those using machine learning, require clean, high-quality data sets. In this paper, we proposed a new high-quality data set for this purpose: The data set contains 15.5 million sentences of arXiv.org publications in the computer science domain. In those sentences, the citation markers were replaced by global paper identifiers. All citing and cited papers are linked to DBLP as much as possible. The data set can be used for a variety of citation-based tasks, such as citation recommendation, citation function determination, and citation-based document summarization.

In the future, besides improving our TEX parser, we will explore how to link arXiv.org papers to further established identifiers besides DBLP identifiers in order to incorporate arXiv.org papers from further disciplines into our data set.

Acknowledgments. Michael Färber is an International Research Fellow of the Japan Society for the Promotion of Science (JSPS). The work was partially supported by MIC SCOPE (171507010).

7. Bibliographical References

Abramsky, S. and McCusker, G. (1995). Games and Full Abstraction for the Lazy lambda-Calculus. In *Proceedings of the 10th Annual IEEE Symposium on Logic in Computer Science*, pages 234–243.

Bird, S., Dale, R., Dorr, B. J., Gibson, B. R., Joseph, M. T., Kan, M., Lee, D., Powley, B., Radev, D. R., and Tan, Y. F. (2008). The ACL anthology reference corpus: A reference dataset for bibliographic research in computational linguistics. In *Proceedings of the International Conference on Language Resources and Evaluation*, LREC 2008.

Caragea, C., Wu, J., Ciobanu, A. M., Williams, K., Ramírez, J. P. F., Chen, H., Wu, Z., and Giles, C. L. (2014). CiteSeer x : A Scholarly Big Dataset. In *Proceedings of the 36th European Conference on IR Research*, ECIR 2014, pages 311–322.

Giles, C. L., Bollacker, K. D., and Lawrence, S. (1998). CiteSeer: An Automatic Citation Indexing System. In *Proceedings of the 3rd ACM International Conference on Digital Libraries*, pages 89–98.

Huang, W., Wu, Z., Chen, L., Mitra, P., and Giles, C. L. (2015). A Neural Probabilistic Model for Context Based Citation Recommendation. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2404–2410.

Karpov, M. (2015). Hackage: megaparsec. `https://http://hackage.haskell.org/package/megaparsec`. Accessed: 2017-09-21.

Knuth, D. E. (1984). A Torture Test for TEX. Technical report, Stanford, CA, USA.

Krijnen, J., Swierstra, S. D., and Viera, M. (2014). Expand: Towards an Extensible Pandoc System. In *Practical Aspects of Declarative Languages – 16th International Symposium*, PADL 2014, pages 200–215.

Ley, M. (2009). DBLP - some lessons learned. *PVLDB*, 2(2):1493–1500.

Loper, E. and Bird, S. (2002). NLTK: the natural language toolkit. *CoRR*, cs.CL/0205028.

Roy, D., Ray, K., and Mitra, M. (2016). From a Scholarly Big Dataset to a Test Collection for Bibliographic Citation Recommendation. In *Scholarly Big Data: AI Perspectives, Challenges, and Ideas, Papers from the 2016 AAAI Workshop*.

Sugiyama, K. and Kan, M. (2015). A comprehensive evaluation of scholarly paper recommendation using potential citation papers. *Int. J. on Digital Libraries*, 16(2):91–109.

Measuring Innovation in Speech and Language Processing Publications.

Joseph Mariani[1], Gil Francopoulo[2], Patrick Paroubek[1]
[1]LIMSI, CNRS, Université Paris-Saclay, [2]Tagmatica
[1]rue John von Neumann, 91400 ORSAY (France), [2]126 rue de Picpus, 75012 PARIS (France)
Joseph.Mariani@limsi.fr, gil.francopoulo@wanadoo.fr, pap@limsi.fr

Abstract

The goal of this paper is to propose measures of innovation through the study of publications in the field of speech and language processing. It is based on the NLP4NLP corpus, which contains the articles published in major conferences and journals related to speech and language processing over 50 years (1965-2015). It represents 65,003 documents from 34 different sources, conferences and journals, published by 48,894 different authors in 558 events, for a total of more than 270 million words and 324,422 bibliographical references. The data was obtained in textual form or as an image that had to be converted into text. This resulted in a lower quality for the most ancient papers, that we measured through the computation of an unknown word ratio. The multi-word technical terms were automatically extracted after parsing, using a set of general language text corpora. The occurrences, frequencies, existences and presences of the terms were then computed overall, for each year and for each document. It resulted in a list of 3.5 million different terms and 24 million term occurrences. The evolution of the research topics over the year, as reflected by the terms presence, was then computed and we propose a measure of the topic popularity based on this computation. The author(s) who introduced the terms were searched for, together with the year when the term was first introduced and the publication where it was introduced. We then studied the global and evolutional contributions of authors to a given topic. We also studied the global and evolutional contributions of the various publications to a given topic. We finally propose a measure of innovativeness for authors and publications.

Keywords: Speech Processing, Natural Language Processing, Text Analytics, Bibliometrics, Scientometrics.

1. Introduction

1.1. Text Analytics of Scientific Papers

The application of text analytics to bodies of scientific papers has become an active area of research in recent years (see for example (Ding Y. et al., 2014), (Banchs R.E., 2012)[1] or the Saffron[2] project. The authors of this paper were invited to conduct several analyses on various conferences: ISCA-Interspeech (J. Mariani et al., 2013), ELRA-LREC (J. Mariani et al., 2014), L&TC (J. Mariani et al., 2015), that they now enlarge to the Speech and Natural Language Processing (SNLP) field in general. They also investigated various aspects of scholar contributions and their evolution over time, such as the production of papers, the collaborations between authors, the citations of papers and authors, the trends in research topics, paper plagiarism and reuse. They proposed measures of the authors' activity based on **paper production**, various kinds of **centrality in collaboration** networks, and **paper citation**. The present paper makes a link between those analyses and proposes in addition a measure of **innovation** that could be attached to research topics, authors or publications. It is based on the detection of the introduction of new terms and on their use in the SNLP research community, assuming that they correspond to a new research topic and exploring who introduced them, when and where, and how successful has the research topic been since then as reflected by the use of the corresponding term after its introduction.

1.2. The NLP4NLP Corpus

In order to conduct our study, we produced a corpus containing research papers on spoken and written language processing, called the NLP4NLP corpus, a name chosen to reflect the fact that the study uses NLP methods that are the subject of the corpus content itself (G. Francopoulo et al., 2015a, G. Francopoulo et al., 2015b). It contains papers from 34 publications, conferences and journals, on SNLP published over 50 years (1965-2015), thereby providing a good picture of research within the international SNLP community. The time span, number and frequency of the events (venues for the conferences, or issues for the journals) and number of papers may strongly vary across the publications. The number of sources globally increased over the year but seems now to be stabilizing at 34. The number of documents also fluctuates over the years, mainly due to the biennial frequency of some conferences. However the total number of papers itself increases steadily reaching a total of more than 65,000 documents as of 2015. In order to study the possible differences across different communities, we considered 3 different research areas, Speech, NLP and Information Retrieval (IR), and we attached the sources to each of those areas, given that some sources may be attached to several areas. The number of documents related to Speech is larger than the one related to NLP, and both are much larger than the one related to IR.

1.3. Data Acquisition

Most of the documents are available in PDF. Those that contain scanned images instead of plain text had to be converted with Tesseract-OC[3] before having their textual content extracted with PDFBox (B. Litchfield, 2005) like the others. A benchmark to estimate the error rate of the extracted content was established based on the ratio of unknown words, using the morphological module of TagParser[4] (G. Francopoulo, 2007), a deep industrial parser based on a broad English lexicon and Global Atlas (a knowledge base containing more than one million words from 18 Wikipedias) (G. Francopoulo, 2013).

[1] The results of these analyses together with corresponding data and tools are also available on-line at the University of Michigan. http://clair.eecs.umich.edu/aan/index.php.
[2] http://saffron.deri.ie
[3] https://code.google.com/p/tesseract-ocr/
[4] www.tagmatica.com

Following this content extraction, another step in our preprocessing was dedicated to split the content into abstract, body and references sections, when they exist. It resulted in a corpus that contains about 270 million words, the quality of which got improved over time. The study of authors is problematic due to variations in the rendering of names (family name and given name, initials, middle initials, ordering, married name, etc.). It therefore required a tedious semi-automatic cleaning process (J. Mariani et al., 2014b), which resulted in a list of 48,894 different authors. The number of authors also varies across the sources. The most productive author published 358 papers, while 26,870 authors (55% of the authors) published only one paper.

2. Terms and Topics

2.1. Term Extraction

Modeling the topics of a research field is a challenge in NLP (see for example (M. Paul et al. 2009), (D. Hall et al., 2008)). Here, our objectives were twofold: i) to compute the most frequent terms used in the domain, ii) to study their variation over time. Like the study of citations, our initial input is the textual content of the papers available extracted from the original electronic documents. Over these 50 years, the archives contain a grand total of 271,934,391 words, mostly in English. Because our aim is to study the terms of the NLP domain, it was necessary to avoid noise from phrases that are used in other senses in the English language. We therefore adopted a contrastive approach, using the same strategy implemented in TermoStat (P. Drouin, 2004). As a first step, we processed a vast number of English texts that were not research papers in order to compute a statistical language profile, using the TagParser deep syntactic parser applied on a corpus containing the British National Corpus (aka BNC), the Open American National Corpus (aka OANC), the Suzanne corpus release-5, the English EuroParl archives (years 1999 until 2009), plus a small collection of newspapers in the domain of sports, politics and economy, taking care of avoiding any texts dealing with SNLP. In a second step, we parsed the NLP4NLP corpus with the same filters and used our language model to distinguish SNLP-specific terms from common ones. We worked from the hypothesis that when a sequence of words is *inside* the NLP4NLP corpus and *not inside* the general language profile, the term is specific to the field of SNLP. The 65,003 documents written by 48,894 authors reduce to 61,661 documents written by 42,278 authors when considering only the papers written in English. They include 3,485,408 different terms (unigrams, bigrams and trigrams) and 23,803,462 term occurrences, provided that this number counts all the occurrences of all the sizes.

Rank	Term	Variants of all sorts	Date when the term appeared	Authors who introduced the term	Documents	Archive #Occurrences	Archive frequency	Archive #Existences	Archive Presence	Archive Rank Occurrence	Archive Rank Presence	Archive Ratio occurrences / existences	# occurrences in the last year	# existences in the last year	Frequency in the last year	Presence in the last year
1	dataset	data-set, data-sets, datasets	1966	Laurence Urdang	cath1966-3	65250	0.003	9940	0.16	11	18	6.6	14039	1472	0.0092	0.44
2	metric	metrics	1965	A Andreyewsky	C65-1002	50679	0.002	11335	0.18	19	10	4.5	5425	1108	0.0036	0.34
3	subset	sub set, sub sets, sub-set, sub-sets, subsets	1965	Denis M Manelski, E D Pendergraft, Gilbert K Krulee, Itiroo Sakai, N Dale, Wojciech Skalmowski	C65-1006 C65-1018 C65-1021 C65-1025	45616	0.002	16939	0.27	22	2	2.7	3463	1095	0.0023	0.33
4	neural network	ANN, ANNs, Artificial Neural Network, Artificial Neural Networks, NN, NNs, Neural Network, Neural Networks, NeuralNet, NeuralNets, neural net, neural nets, neural networks	1980	Bonnie Lynn Webber	P80-1032	54790	0.002	8885	0.14	16	27	6.2	8024	1037	0.0053	0.31
5	classifier	classifiers	1967	Aravind K Joshi, Danuta Hiz	C67-1007	98229	0.004	11546	0.18	7	9	8.5	8202	1000	0.0054	0.30
6	SR	ASR, ASRs, Automatic Speech Recognition, SRs, Speech Recognition, automatic speech recognition, speech recognition	1970	Josse De Kock	cath1970-9	129979	0.006	20382	0.32	2	1	6.4	8524	1000	0.0056	0.30
7	optimization	optimisation, optimisations, optimizations	1967	Ellis B Page	C67-1032	35257	0.002	10196	0.16	35	16	3.5	3331	903	0.0022	0.27
8	annotation	annotations	1967	Kenneth Janda, Martin Kay	cath1967-12 cath1967-8	111084	0.005	11975	0.19	4	7	9.3	7515	896	0.0049	0.27
9	POS	POSs, Part Of Speech, Part of Speech, Part-Of-Speech, Part-of-Speech, Parts Of Speech, Parts of Speech, Pos, part of speech, part-of-speech, parts of speech, parts-of-speech	1965	Denis M Manelski, Dániel Várga, Gilbert K Krulee, Makoto Nagao, Toshiyuki Sakai	C65-1018 C65-1022 C65-1029	102057	0.005	13823	0.22	5	4	7.4	7489	860	0.0049	0.26
10	LM	LMs, Language Model, Language Models, language model, language models	1965	Sheldon Klein	C65-1014	116684	0.005	13117	0.21	3	5	8.9	8522	851	0.0056	0.26

Table 1: 10 most present terms in 2015, with variants, date, authors and publications where they were first introduced, number of occurrences and existences in 2015, number of occurrences, frequency, number of existences and presence in the 50 year archive, with ranking and average number of occurrences of the terms in the documents

The 500 most frequent terms were computed over the period of 50 years, according to the following strategy. First, the most frequent terms were computed from raw occurrence counts, and secondly the synonyms sets (aka *synsets*) for the most 200 frequent terms of each year were manually declared in the lexicon of TagParser. We gather in the term *synset*, the variation in upper / lower case, singular / plural number, US / UK difference, abbreviation / expanded form and absence / presence of a semantically neutral adjective, like "artificial" in "artificial neural network". Thirdly, the most frequent terms were recomputed with the amended lexicon. We will call "*existence*"[5] the fact that a term exists in a document and "*presence*" the percentage of documents where the term exists. We computed in that way the occurrences, frequencies, existences and presences of the terms globally and over time (1965-2015), and the average number of occurrences of the terms in the documents where they exist (see Table 1). The ranking of the terms slightly differs if we consider the frequency or the presence. The most frequent term overall is "HMM" (*Hidden Markov Models*), which doesn't appear on Table

[5] sometimes called "Boolean frequency" or "binary frequency"

1 as it is ranked 16th in 2015, while the most present term is "*Speech Recognition*". The average number of occurrences of the terms in the documents where they exist varies a lot (from 9.3 for "*annotation*" to 2.7 for "*subset*" in Table 1).

2.2. New Terms Introduced by the Authors

We then studied when and who introduced new terms, as a mark of the innovative ability of various authors, which may also provide an estimate of their contribution to the advances of the scientific domain. We make the hypothesis that an innovation is induced by the introduction of a term which was previously unused in the community and then became popular. We then take into account the terms that are of scientific interest (excluding author's names, unless they correspond to a specific algorithm or method, city names, laboratory names, etc.). For each of these terms, starting from 1965, we determine the author(s) who introduced the term, referred to as the "inventor(s)" of the term. This may yield several names, as the papers could be co-authored or the term could be mentioned in more than one paper on a given year.

Table 1 provides the ranked list of the 10 most popular terms in 2015 based on their presence. For example, the term *Dataset* appeared first in the year 1966, when it was mentioned in a single paper authored by L. Urdang[6] while it was mentioned 14,039 times in 1,472 papers in 2015, and 65,250 times in 9,940 papers overall (i.e. in 16% of the papers!). From its first mention in the introduction of a panel session by Bonnie Lynn Webber at ACL[7] in 1980 to 2015, the number of papers mentioning *Neural Networks* increased from 1 to 1037, and the number of occurrences reached 8,024. *Metric, Subset, Classifier, Speech Recognition, Optimization, Annotation, Part-of-Speech* and *Language Model* are other examples of terms that are presently most popular.

3. Measuring Innovation

3.1. Measuring the Importance of Topics

We then considered the possibility to measure the importance of a term. Fig. 1 gives the annual presence (percentage of papers containing the term) for the term "*cross validation*", which was encountered for the first time in 2 papers in 2000. In order to measure the success of the term over time, we compute the sum of the annual presences. We may choose to consider all papers or only those ("external papers" marked in orange) that are written by authors who are different than those who introduced the term (marked in blue).

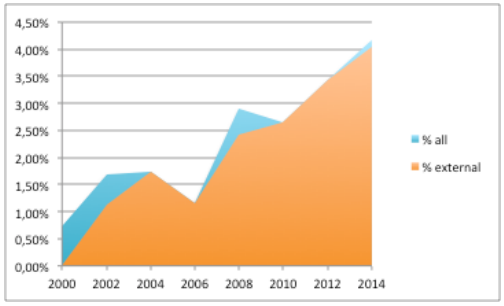


Fig. 1: Presence of the term "*cross validation*" over the years (% of all papers)

We propose to consider as the annual innovation score the presence of the term on that year. In this example, it went from 0.75% of the papers in 2000 to 4% of the papers in 2014. We propose to consider as the global innovation score of the term the corresponding surface, taking into account the inventors' papers in the year of introduction and all the papers in the subsequent years. We see here that it takes into account the periods when the term gets more present (2000 to 2004, 2006 to 2008 and 2010 to 2014), as well as those when it looses popularity (2004 to 2006 and 2008 to 2010). The innovation score for the term is the sum of the yearly presences of the term and amounts to 0.17 (17%). This approach emphasizes the importance of the term in the first years when it is mentioned, as the total number of papers is then lower. Some non-scientific terms may not have been filtered out, but their influence will be small as their presence is limited and random.

We considered the 1,000 most frequent terms over the 50-year period, as we believe they contain most of the important scientific advances in the field of SNLP. Given the poor quality and low number of different sources and papers in the first years, we decided to only consider the period from 1975 to 2015. This innovation measure provides an overall ranking of the terms. We also computed separate rankings for NLP and for Speech (Table 2).

Rank	Terms		
	Overall	NLP	Speech
1	Speech Recognition	semantic	Speech Recognition
2	Subset	syntactic	Spectral
3	Semantic	NP	Acoustics
4	Filtering	POS	Gaussian
5	HMM	parser	HMM
6	Spectral	parsing	Filtering
7	Linear	subset	Linear
8	iteration	lexical	Fourier
9	Language Model	Machine Translation	Subset
10	POS	predicate	Acoustic

Table 2: Global ranking of the importance of the terms overall and separately for Speech and NLP

We studied the evolution of the presence of the terms over the years, in order to check the changes in paradigm. However, the fact that some conferences are annual, while others are biennial brings noise. Instead of considering the annual presence of the terms (percentage of papers containing a given term **on** a given year), we therefore considered the cumulative presence of the terms

[6] Laurence Urdang (1966), The Systems Designs and Devices Used to Process The Random House Dictionary of the English Language. Computer and the Humanities. Interestingly, the author writes: *"Each unit of information-regardless of length-was called a dataset, a name which we coined at the time. (For various reasons, this word does not happen to be an entry in The Random House Dictionary of the English Language, our new book, which I shall refer to as the RHD)."* a statement which witnesses her authorship of the term.

[7] Interestingly, she mentions the Arthur Clarke's "2001, Space Odyssey" movie: *"Barring Clarke's reliance on the triumph of automatic neural network generation, what are the major hurdles that still need to be overcome before Natural Language Interactive Systems become practical?"*, which may appear as a premonition in 1980!

(percentage of papers containing a given term **up to** a given year) (Fig. 2).

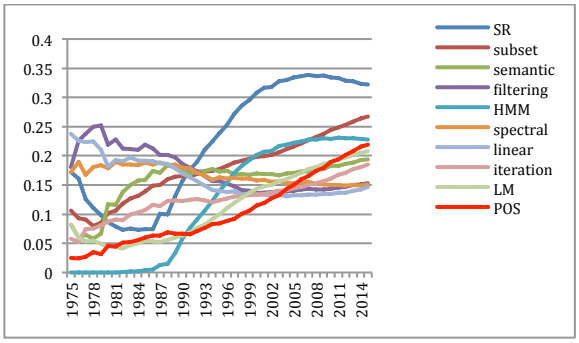


Fig. 2: Cumulative presence of the 10 most important terms over time (% of all papers)

We see that *Speech Recognition* has been a very popular topic over the years, reaching a presence in close to 35% of the papers published up to 2008. Its shape coincides with *Hidden Markov Models* that accompanied the effort on *Speech Recognition* as the most successful method over a long period and had then been mentioned in close to 25% of the papers. *Semantic* processing was a hot topic of research by the end of the 80's, and regained interest recently. *Language Models* and *Part-of-Speech* received continuing marks of interest over the years.

3.2. Measuring Authors' Innovation

We also computed in a similar way an *innovation score* for each author, illustrating his or her contribution in the introduction and early use of new terms that subsequently became popular. The score is computed as the sum over the years of the annual presence of the terms in papers published by the authors (percentage of papers containing the term and signed by the author on a given year). This innovation measure provided an overall ranking of the authors. We also computed separate rankings for NLP and for Speech Processing (Table 3).

Rank	Authors		
	Overall	NLP	Speech
1	Lawrence R Rabiner	Ralph Grishman	Lawrence R Rabiner
2	Hermann Ney	Kathleen R Mckeown	John H L Hansen
3	John H L Hansen	Jun'Ichi Tsujii	Shrikanth S Narayanan
4	Shrikanth S Narayanan	Aravind K Joshi	Hermann Ney
5	Chin Hui P Lee	Jaime G Carbonell	Chin Hui P Lee
6	Li Deng	Ralph M Weischedel	Li Deng
7	Mari Ostendorf	Mark A Johnson	Mark J F Gales
8	Alex Waibel	Fernando C N Pereira	Frank K Soong
9	Haizhou Li	Christopher D Manning	Haizhou Li
10	John Makhoul	Ted Briscoe	Thomas Kailath

Table 3: Global ranking of authors overall and separately for Speech and NLP

We should stress that this measure doesn't place on the forefront uniquely the "inventors" of a new topic, as it is difficult to identify them given that we only consider a subset of the scientific literature over a limited period. It rather helps identifying the early adopters who published a lot after the topic was initially introduced. We studied several cases, such as F. Jelinek and S. Levinson regarding *Hidden Markov Models*, where renowned authors don't appear within the 10 top authors contributing to those terms. We often see that they initially published in a different research field than SNLP (the *IEEE Transactions on Information Theory* in the case of F. Jelinek, for example) that we don't consider in our corpus. This measure also reflects the size of the production of papers from the authors on emerging topics, with an emphasis on the pioneering most ancient authors, such as L. Rabiner and J. Makhoul, at a time when the total number of papers was low. The overall ranking also favors those who published both in Speech and Language Processing, such as H. Ney or A. Waibel.

We may study the domains where the authors brought their main contributions, and how it evolves over time. We faced the same problem due to the noise brought by the different frequency of the conferences as we did when studying the evolution of the terms, and we rather considered the cumulative contribution of the author specific to that term (percentage of papers signed by the author among the papers containing a given term (that we will call "*topical papers*") **up to** a given year). We see for example that L. Rabiner brought important early contributions to the fields of *Acoustics*, *Signal Processing* and *Speech Recognition* in general, and specifically to *Linear Prediction Coding (LPC)* and *filtering* (Fig. 3). He even authored 30% of the papers dealing with *LPC* which were published up to 1976 and the only paper mentioning *endpoint detection* in 1975.

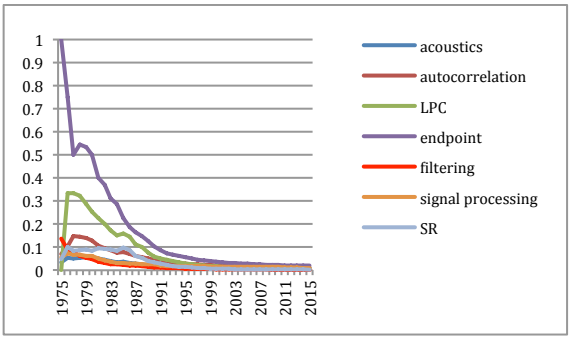


Fig. 3: Main contributions areas for L. Rabiner (% of topical papers)

H. Ney brought important contributions to the study of *perplexity* (authoring 10% of the papers which were published on that topic up to 1988) in *Language Models* (LM) using trigrams and bigrams (Figure 4).

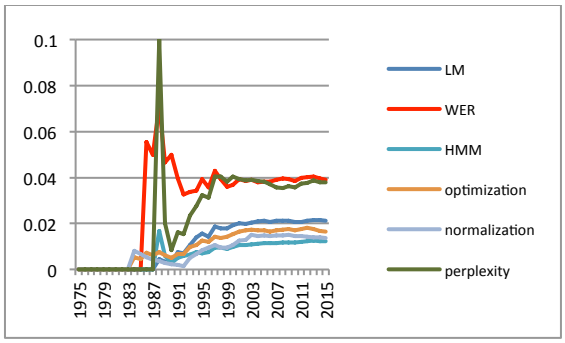


Fig. 4: Main contribution areas for H. Ney (% of topical papers)

A. Waibel brought important contributions in the use of *HMM* and even more of *Neural Networks* for speech and language processing already in the early 90s (Figure 5).

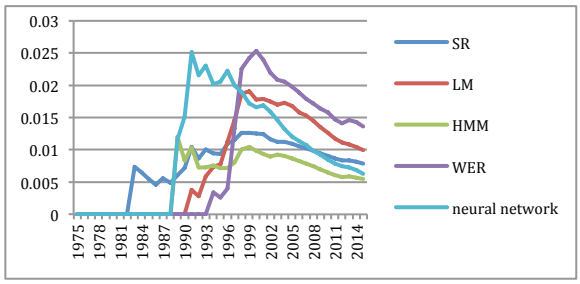


Fig. 5: Main contribution areas for A. Waibel (% of topical papers)

We may also wish to study the contributions of authors on a specific topic, using the same cumulative score. Fig. 6 provides the cumulative percentage of papers containing the term *HMM* published up to a given year by the 10 most contributing authors. We also added F. Jelinek as a well-known pioneer in that field and S. Levinson as the author of the first article containing that term in our corpus, which represented 0.4% of the papers published in 1982. We see the contributions of pioneers such as F. Soong, of important contributors in an early stage such as C. H. Lee, S. Furui or K. Shikano or a later stage such as M. Gales.

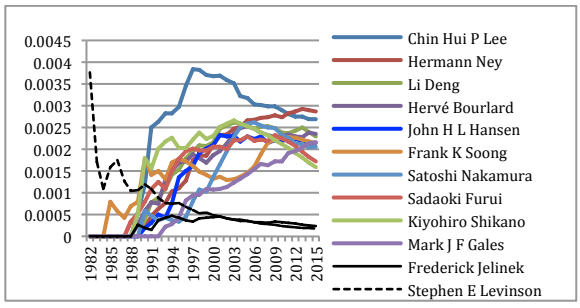


Fig. 6: Authors' contributions to *HMM* in SNLP (% of all papers)

Similarly, we studied the authors' contributions to *Deep Neural Networks (DNN*) which recently gained a large audience (Figure 7). We see the strong contribution of Asian authors on this topic, with the pioneering contributions of Dong Yu and Li Deng up to 2012 where they represented altogether about 50% of the papers mentioning DNN since 2009, while Deliang Wang published later but with a large productivity which finally places him at the second rank globally.

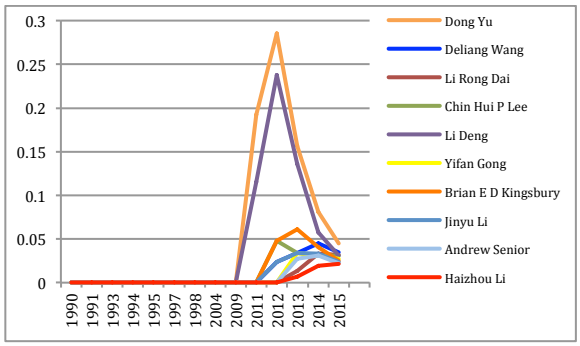


Fig. 7: Authors' contributions to the study of *DNN* in speech and language processing (% of topical papers)

3.3. Measuring the Innovation in Publications

We finally computed with the same approach an *innovation score* for each publication. The score is similarly computed as the sum over the years of the annual presence of the terms in papers published in the source, conference or journal (percentage of papers containing the term which were published in the publication on a given year). This innovation measure provided an overall ranking of the publication. We also computed separate rankings for NLP and for Speech Processing (Table 4).

Rank	Sources		
	Overall	NLP	Speech
1	taslp	acl	taslp
2	isca	coling	isca
3	icassps	cath	icassps
4	acl	lrec	lrec
5	coling	cl	csal
6	lrec	hlt	speechc
7	hlt	eacl	mts
8	emnlp	emnlp	ltc
9	cl	trec	lre
10	cath	mts	acmtslp

Table 4 : Global ranking of the importance of the sources overall and separately for Speech and NLP

Just as in the case of authors, the measure also reflects here the productivity, which favors the Speech Processing field where more papers have been published, and the pioneering activities, as reflected by the ranking of *IEEE TASLP*. In the overall ranking, publications that concern both Speech and Language Processing (LREC, HLT) also get a bonus here.

We may study the domains where the publications brought their main contributions, and how it evolves over time. We faced the same problem due to the noise brought by the different frequency of the conferences as we did when studying the evolution of the terms and authors, and we rather considered the cumulative contribution of the publication specific to that term (percentage of papers published in the source among the papers containing the term **up to** a given year). We see for example (Fig. 8) that ACL showed a strong activity and represented 40% of papers published about *parsing*, 35% of papers published about *semantic, syntactic* and *lexical* and 25% of papers published about *Machine Translation* up to 1985. Its share in those areas then globally decreases to about 15% of the total number of publications, due to the launching of new conferences and journals, while the share of publications on *Machine Translation* within ACL recently increased.

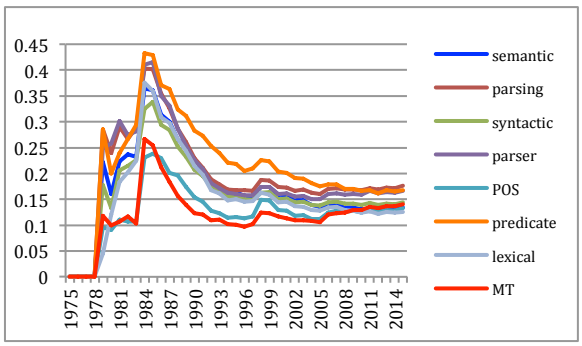


Fig. 8: Main domains within the ACL conference series (% of topical papers)

We may also wish to study the contributions of publications to a specific term, using the same cumulative score. Fig. 9 provides the cumulative percentage of papers containing the term *HMM* published up to a given year by the 10 most contributing publications. We see that all papers were initially published in the *IEEE Transactions on Speech and Audio Processing*. Other publications took a share of those contributions when they were created (*Computer Speech and Language* starting in 1986, *ISCA Conference series* starting in 1987) or when we start having access to them (*IEEE-ICASSP*, starting in 1990). We see that *ISCA Conference series* represents 45% of the papers published on HMM up to 2015, while *IEEE-ICASSP* represents 25%. We also see that HMMs were first used in speech processing related publications, then in NLP publications as well (ACL, EMNLP), while publications that are placed in both (CSL, HLT, LREC) helped spreading the approach from speech to NLP.

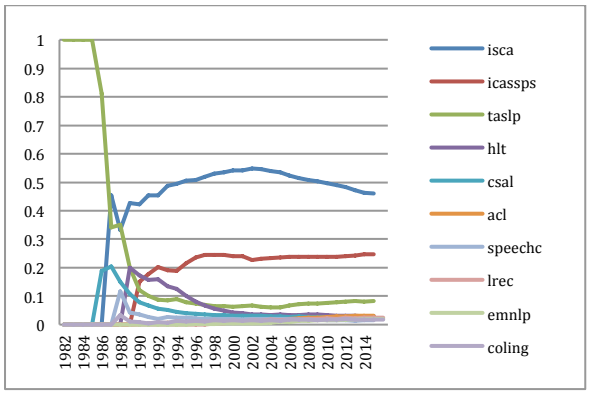


Fig. 9: Sources' contributions to the study of *HMM*. *(% of topical papers)*

4. Perspectives and Conclusions

We proposed in this analysis a measure of innovation for terms, authors and sources. This measure gives an image of the scientific community that seems acceptable. However, it emphasizes the eldest contributions and the productivity. We plan to further refine this measure. We already experimented some variants of the algorithm, such as only considering the periods when the popularity of a term is increasing, without getting very different results. In this analysis, we faced the problem of the lack of quality of the most ancient data that was obtained through OCR from the paper version of the proceedings, which sometimes even contain handwritten comments! For that reason, we focused the study on the period starting in 1975 and we still had to carry out some manual corrections. We plan to develop an automatic term extraction process taking into account the context in which the term is identified. This would allow making the distinction between real and false occurrences of the terms, especially when they have acronyms as variants. It would avoid the tedious manual checking that we presently conduct and would improve the overall process.

5. Bibliographical References

Banchs, Rafael E. (2012), Proceedings of the *ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries* Association for Computational Linguistics 2012 Jeju, Korea. http://aclweb.org/anthology/W12-32

Ding, Ying; Rousseau, Ronald and Wolfram, Dietmar ed. (2014), *Measuring Scholarly Impact*, Springer. 2014, ISBN: 978-3-319-10376-1.

Drouin, Patrick (2004), Detection of Domain Specific Terminology Using Corpora Comparison. In Proceedings of the Language Resources and Evaluation Conference (LREC 2004), Lisbon,

Francopoulo, Gil (2007), TagParser: well on the way to ISO-TC37 conformance. ICGL (International Conference on Global Interoperability for Language Resources), Hong Kong.

Francopoulo, Gil; Marcoul, Frédéric; Causse, David and Piparo, Grégory (2013), Global Atlas: Proper Nouns, from Wikipedia to LMF, in LMF-Lexical Markup Framework, Gil Francopoulo ed, ISTE/Wiley.

Francopoulo, Gil; Mariani, Joseph and Paroubek, Patrick (2015a), NLP4NLP: The Cobbler's Children Won't Go Unshod, 4th International Workshop on Mining Scientific Publications (WOSP2015), Joint Conference on Digital Libraries 2015 (JCDL 2015), Knoxville (USA), June 24, 2015.

Francopoulo, Gil; Mariani, Joseph and Paroubek, Patrick (2015b), NLP4NLP: Applying NLP to written and spoken scientific NLP corpora, Workshop on Mining Scientific Papers: Computational Linguistics and Bibliometrics, 15th International Society of Scientometrics and Informetrics Conference (ISSI 2015), Istanbul (Turkey), June 29, 2015.

Litchfield, Ben (2005), Making PDFs Portable: Integrating PDF and Java Technology, March 24, 2005, Java Developers Journal. http://java.sys-con.com/node/48543

Mariani, Joseph; Paroubek, Patrick; Francopoulo, Gil and Delaborde, Marine (2013), Rediscovering 25 Years of Discoveries in Spoken Language Processing: a Preliminary ISCA Archive Analysis, Proceedings of Interspeech 2013, 26-29 August 2013, Lyon, France.

Mariani, Joseph; Paroubek, Patrick; Francopoulo, Gil and Hamon, Olivier (2014a), Rediscovering 15 Years of Discoveries in Language Resources and Evaluation: The LREC Anthology Analysis, Proceedings of LREC 2014, 26-31 May 2014, Reykjavik, Iceland.

Mariani, Joseph; Cieri, Christopher; Francopoulo, Gil; Paroubek, Patrick and Delaborde, Marine (2014b), Facing the Identification Problem in Language-Related Scientific Data Analysis, Proceedings of LREC 2014, 26-31 May 2014, Reykjavik, Iceland.

Mariani, Joseph; Francopoulo, Gil; Paroubek, Patrick and Vetulani, Zygmunt (2015), Rediscovering 10 to 20 Years of Discoveries in Language & Technology, L&TC 2015, 27-29 November 2015, Poznan, Poland.

Paul, Michael and Roxana Girju (2009), Topic Modeling of Research Fields: An Interdisciplinary Perspective, In Recent Advances in Natural Language Processing (RANLP 2009), Borovets, Bulgaria.

6. Language Resource References

Francopoulo, Gil; Mariani, Joseph and Paroubek, Patrick (2016), NLP4NLP: NLP scientific papers for processing with NLP technology http://www.nlp4nlp.org/

PDFdigest: an Adaptable Layout-Aware PDF-to-XML Textual Content Extractor for Scientific Articles

Daniel Ferrés*, Horacio Saggion*, Francesco Ronzano†, and Àlex Bravo*
* Large Scale Text Understanding Systems Lab
TALN Research Group, DTIC Department
Universitat Pompeu Fabra, Barcelona (Spain)
{daniel.ferres,horacio.saggion,alex.bravo}@upf.edu
† Integrative Biomedical Informatics Group (IBI)
Research Programme on Biomedical Informatics (GRIB)
IMIM-UPF, Barcelona (Spain)
francesco.ronzano@upf.edu

Abstract

The availability of automated approaches and tools to extract structured textual content from PDF articles is essential to enable scientific text mining. This paper describes and evaluates the PDFdigest tool, a PDF-to-XML textual content extraction system specially designed to extract scientific articles' headings and logical structure (title, authors, abstract,...) and its textual content. The extractor deals with both text-based and image-based PDF articles using custom rule-based algorithms implemented with existing state-of-the-art open-source tools for both PDF-to-HTML conversion and image-based PDF Optical Character Recognition.

Keywords: Language Resources, Scientific Text Mining, Digital Libraries, Information Extraction, PDF Conversion

1. Introduction

Nowadays automated approaches to mine scientific literature are essential to support researchers - as well as any other interested actor - in taking full advantage of the huge number of articles available online: (Ware and Mabe, 2015) estimate that more than 2.5 million papers are published on the Web every year, and the percentage of publications distributed as Open Access content is constantly increasing. Even if XML-based formats are emerging, the majority of scientific articles are still accessed as PDF files. As a consequence, effective tools to extract structured textual content from PDF files represent a key technology to enable scientific text mining (Ronzano and Saggion, 2016). Such tools are essential components to develop a varied range of applications in different academic and industrial contexts, thus supporting the implementation of new, intelligent patterns to access scientific information (Accuosto et al., 2017). Once textual contents are extracted from scientific publications, they can be mined and enriched using existing Natural Language Processing (NLP) techniques such as Named Entity Recognition and Classification, Parsing, Relation Extraction, among others, thus helping researchers to improve their access to the scientific knowledge.

In this paper we present PDFdigest, a tool that extracts structured textual contents from scientific articles in PDF format. PDFdigest performs the conversion of files from PDF to XML format while preserving both the textual content and the layout details of the input PDF document.

The contributions of this paper can be summarized as follows:

- PDFdigest, a PDF to XML system that extracts the content of both text-based and image-based PDF files[1] by:
  - dealing with specific paper layouts as well as with text in multiple languages;
  - storing the layout of the original PDF in an HTML file, thus enabling the possibility of Visual Analytics over the original PDF document;
  - implementing a customizable approach to adapt the content extraction process to different PDF paper layouts by manually changing the extraction offsets and thresholds.
- An evaluation of the effectiveness of the text-based and image-based PDF textual content extraction algorithms with papers from a bilingual (English/Spanish) journal in the Natural Language Processing field (i.e. the SEPLN journal).

The rest of the paper is organized as follows: Section 2 provides an overview of the related work. Section 3 describes the main features of PDFdigest. The textual content extraction approach for text-based PDF files is described in Section 4. Section 5 introduces the approach to process image-based PDF files. Then, in Section 6 we evaluate the text-based and the image-based PDF extraction algorithms. In Section 7 we describe the online web-service demo. Finally, we discuss the key PDF mining advantages provided by PDFdigest (Section 8) and formulate our conclusions and future work plans (Sections 9 and 10 respectively).

2. Related work

During the last few years, several PDF-to-text extraction tools have been proposed, tailored to extract textual contents from PDF articles identifying their structural organization by spotting a set of common elements like the title, the authors' names and their affiliations, the abstract, one

[1] Web-service: `http://taln.upf.edu/pdfdigest`

or more sections and subsections, and the bibliographic entries included in the bibliography. Some examples of tools with such functionalities are:

- **PDFX**[2] (Constantin et al., 2013) that parses PDF files of scientific publications by exploiting several heuristics related to the layout and the lexical features of a paper to identify its structural elements;
- **Cermine**[3] (Tkaczyk et al., 2014) that processes the contents of PDF articles and properly classifies text zones as belonging to four general classes: metadata, references, body and other. In the metadata zone another classifier is responsible for the identification of the title of the paper, the authors, the affiliation, the keywords and other relevant information usually included in the header of an article. The zone classifiers exploited by Cermine are Support Vector Machines (Chang and Lin, 2011) that rely on lexical, geometrical, sequential and formatting features of the different zones of a paper to classify;
- **GROBID**[4] (Lopez, 2009) that exploits a chain of Conditional Random Field classifiers (from the Mallet library [5]) to extract a hierarchical set of structural elements from PDF papers;
- **SectLabel**[6] (Luong et al., 2012) that exploits a sequence tagging model (Conditional Random Fields) to associate to each sentence of a scientific paper a structural and rhetorical category chosen among a set of 23 structural categories (title, figure, section header, etc.) and 13 rhetorical categories (abstract, introduction, background, etc.);
- **SideNoter** (Abekawa and Aizawa, 2016) a PDF extraction tool that allows the possibility to refer the extracted text spans to the original layout of the PDF document processed.

GROBID, CERMINE and SectLabel are distributed as open-source software, while PDFX can be accessed as an on-line Web Service. At the time of writing, we are not aware of any implementation of Web Service available to process PDF papers by means of SideNoter.

3. System Description

The PDFdigest textual content extractor tool is a Java-based application that extracts some specific relevant logic and textual content from scientific articles in PDF format and stores its output in a file with XML format. It can detect both one-column and two-column articles in one or several languages for the same article (i.e. the SEPLN articles we consider in our evaluation combine Spanish and English).
The extraction algorithm extracts the following parts of scientific articles:

- *titles*: main title, second title (in case it exists)
- *authors' names,*
- *authors' affiliation*
- *authors' email*
- *abstract(s)*
- *categories*
- *keywords*
- *sections' titles*: (including subsections and subsubsections).
- *sections' textual content*: (detected at paragraph level and can be associated also to subsections and subsubsections)
- *bibliographic references*: including the identification of each single bibliographic entry individually.
- other fields: such as *acknowledgements,* annexes, *author's biographies,* captions (from tables and figures), and other supporting information.

The extractor deals with both text and image-based PDF files with two approaches that share some similarities (see in Figure 1 a diagram that shows the PDFdigest architecture).

4. PDFdigest: Text-Based PDF-to-XML Extraction

The text-based PDF-to-XML extraction algorithm has 6 steps that are executed sequentially: 1) PDF to HTML conversion, 2) HTML tag properties and CSS properties' values statistics computation, 3) content filtering, 4) rule-based content detection and extraction, 5) language prediction, and finally, 6) XML generation.

4.1. PDF-to-HTML Conversion

The PDF to HTML conversion is performed using the pdf2htmlEX[7] extractor which converts each PDF file into an HTML file with the content structured in several HTML tag elements with associated CSS properties and inner textual content[8]. The HTML file includes DIV elements defining the position and style of small portions of the paper, preserving the paper's original layout by means of CSS properties.

4.2. HTML and CSS Tag Properties Statistics

The following phases of statistics computation and content extraction take profit of the generated HTML syntax and the CSS properties' values to extract the content. The JSoup[9] HTML parsing library is used in these phases. The second phase calculates some data statistics of the HTML document and its CSS properties (e.g. the most used textual font in the paper and its size).

[2] http://pdfx.cs.man.ac.uk/
[3] https://github.com/CeON/CERMINE
[4] https://github.com/kermitt2/grobid
[5] http://mallet.cs.umass.edu/
[6] https://github.com/knmnyn/ParsCit/tree/master/bin/sectLabel
[7] https://github.com/coolwanglu/pdf2htmlEX
[8] In some cases the pdf2htmlEX cannot extract textual content of PDF files that contain images instead of text.
[9] https://jsoup.org/

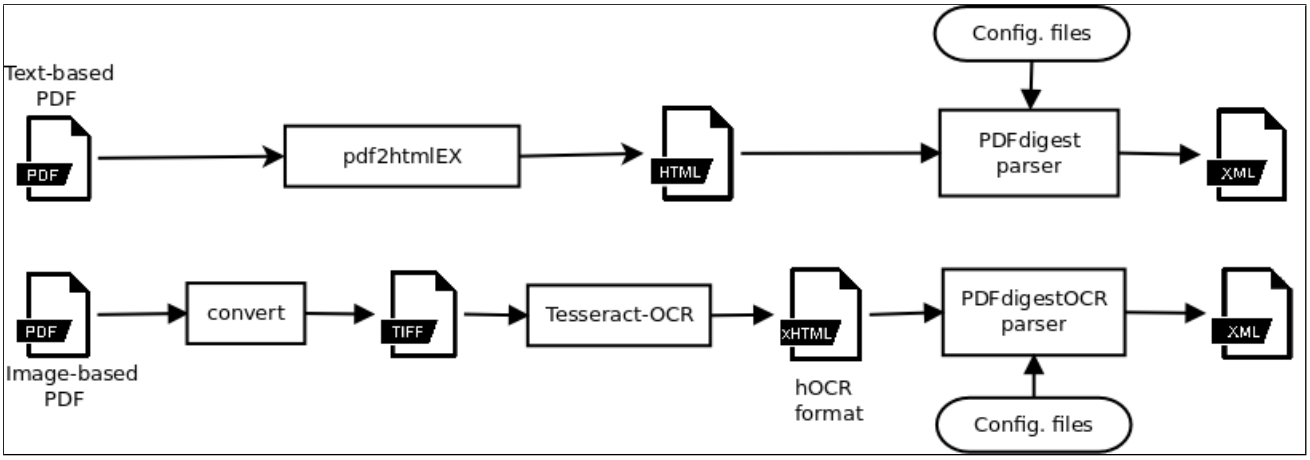


Figure 1: PDFdigest System Architecture for text-based and image-based PDF files.

4.3. Content Filtering

This phase filters out some textual content that is not extracted in the final XML file. The parts of the paper that are filtered out are: running heads, page numbers, footnotes and table contents. The following filtering strategies are the following:

- Pixel percentage margin-based filtering: this strategy filters out text that appears in the upper and lower margins of the document using the document height and predefined pixel percentage thresholds of this height for these margins.
- Text-based margin detection filtering: detects the top y-axis and bottom y-axis position of the basic textual content in the paper and uses it to filter out content in the top and bottom margins.
- Footnotes and table contents filtering functions.

4.4. Rule-Based Content Extraction

Then the rule-based extraction phase iterates over some specific tags of the HTML file and uses several manually generated rules to detect some specific content markers and consume its content. Each part of the textual content (i.e. title, abstract, acknowledgements) has its own extraction rules and consumption procedures. The rules are based on information from statistics, from the content markers detected previously, and from a set of language-dependent and content-specific regular expressions that can be manually modified or extended.
The rule-based extraction uses the following information:

1. Finite state machines that capture the regular logic structure of papers (i.e. abstract goes always after title detection, bibliographic entries go always after sections' textual content).
2. HTML attributes' statistics (i.e the most used textual font in the paper and its size).
3. CSS properties' values (e.g. values indicating the y-axis position of an element tag in the page).
4. Regular expressions that can cover several languages (currently the system supports Spanish and English). As an example, the regular expressions can find strings such as 'Abstract:' or 'Resumen:' expressions to detect the abstract titles in English and Spanish respectively.
5. HTML local element specific data (i.e. distance between the current and previous HTML tag elements).
6. Manually tuned offsets and thresholds (i.e. the maximum distance between the title and a line of text below the title to be considered part of the title).
7. Lists of language dependent hyphenated words extracted from the *Spanish LMF Freeling Lexicon* (UPC-TALP, UA-InterNostrum, UB-CLiC, and UPF-IULA , 2011) and the *Freeling*[10] dictionary for English (Padró and Stanilovsky, 2012). These lists are used to detect and mark real and false hyphens from the original text to the extracted one."Real hyphens" are the ones that appear at the end of a line but they coincide with a hyphen char from a hyphenated word (e.g. walkie-talkie). On the other hand, "false hyphens" are the ones appearing at the end of line but are not part of a hyphenated word.

The creation of rules to extract the content and the layout information was based on a set of examples from the following Computer Science conferences and journals: Computer Animation Virtual Worlds, ACM SIGGRAPH, Association for Computational Linguistics (ACL) Conference, and Language and Resources Evaluation Conference (LREC).

4.5. Language Prediction

The final phases consist in the language prediction of the recognized content and the generation of the output file in XML format. A set of language probabilities are calculated individually for each part of the content extracted and globally for the whole textual content extracted from the ar-

[10] http://nlp.cs.upc.edu/freeling/

ticles' sections. The language prediction is computed using the *optimaize language detector* [11] java API.

4.6. XML Generation and Validation

The XML generates and validates the XML output file. XML validation checking is performed with both JTidy[12] and SAX[13] Java parsers. The XML output contains a set of tags referring to the different logical sections of the paper (e.g. Abstract, Title, Author, Keywords, H1 (section title), Paragraph, BibEntry...). Each logical tag includes its related extracted content (including normalization of ligatures) within the "div" tags that relate the content to the original PDF2htmlEX HTML output "div" tags. Moreover each tag contains an attribute with the predicted language.

5. PDFdigestOCR: Image-Based PDF-to-XML Extraction

The image-based PDF-to-XML extraction is more difficult than the text-based one and needs special technologies to convert images into text. The image-based PDF-to-XML extraction algorithm has 5 steps executed sequentially: 1) image to hOCR format and textual lines layout bounding boxes extraction, 2) content filtering, 3) rule-based content detection and extraction, 4) language prediction, and finally, 5) XML generation and validation. The two last phases perform the same functionality described in subsections 4.5. and 4.6.

5.1. PDF-to-hOCR Conversion

This phase uses the Tesseract-OCR[14] state-of-the-art Optical Character Recognizer (OCR) engine to extract information from PDF files. First, PDF files are converted to images using the *convert*[15] utility. Then the OCR extracts the information of these files in a single file with the hOCR format using language specific pre-trained models. hOCR is a representation of text obtained from an OCR that stores text, style, layout information, and recognition confidence metrics in xHTML. The extraction of the bounding boxes of each textual line recognized by the OCR is crucial to detect useful data for the heuristics such as the distance between lines.

5.2. Content Filtering

This phase filters out running heads, page numbers and other textual contents in the top and bottom margins of the papers. The strategy used is the pixel percentage margin-based filtering (explained in Section 4.3.).

5.3. Rule-Based Content Extraction

Then the rule-based extraction phase, similar to the one presented in section 4.4., iterates over some specific tags of the hOCR file (the ones that have the attribute *class* equal to '*ocr_line*') and uses several manually generated rules to detect some specific content markers and consume its content. The rule-based extraction uses the following information (some points have been explained in Section 4.4.):

1. Finite state machines.
2. hOCR line-based bounding boxes data.
3. Regular expressions.
4. Manually tuned offsets and thresholds.
5. Lists of language dependent hyphenated words.

6. Evaluation

An evaluation of the quality extraction of both the PDFdigest text-based and the PDFdigestOCR image-based PDF-to-XML approaches was done with the creation of a gold standard set[16] of 27 bilingual (English/Spanish) SEPLN articles[17] that were manually annotated in order to spot: title(s), abstract(s), list of keywords, section headers (up to a depth of three levels), paragraphs, table captions, figure captions and bibliographic entries. Moreover, in order to evaluate the PDFdigestOCR image-based approach, the original dataset of 27 articles was also converted to image-based PDFs using the following Linux tools executed sequentially: 1) *pdftoppm* with a resolution of 300 DPI and 2) *convert* with the A4 layout format.
The PDFdigest text-based evaluation has been performed comparing the sets of "div" tags predicted to pertain to each class with the gold standard annotated "div" tags. The result of the PDFdigest text-based evaluation has an average F1 score of 0.917 (see global and specific results in Table 1).

Table 1: PDFdigest text-based evaluation

Class	Prec.	Recall	F1
Title	1.000	1.000	1.000
Abstract	0.921	0.928	0.890
Keywords	0.963	0.917	0.915
H1 -section title-	0.978	0.807	0.876
H2 -subsection title-	0.875	0.876	0.864
H3 -subsubsection title-	0.875	1.000	0.917
Paragraph	0.994	0.874	0.923
Table caption	0.549	0.879	0.664
Figure caption	0.584	0.922	0.691
BibEntry	0.923	0.915	0.919
Avg. weighted by class freq.	0.878	0.976	0.917

Most of the extraction errors of the PDFdigest text-based algorithm are due to: 1) papers without textual content extracted from the original PDF to the HTML file (includes cases in which wrong characters have been extracted and

[11] https://github.com/optimaize/language-detector
[12] http://jtidy.sourceforge.net/
[13] http://www.saxproject.org/
[14] https://github.com/tesseract-ocr/tesseract
[15] https://www.imagemagick.org

[16] The evaluation dataset with the gold annotated data and the PDF articles in text-based and image-based versions is avalaible for download at this site: http://taln.upf.edu/pdfdigest/resources.php
[17] SEPLN Journal http://journal.sepln.org

cases with only images of the full paper extracted), 2) papers that do not follow the logical structure of a regular paper (i.e. papers without abstract, papers without section titles,..).
On the other hand, the evaluation of the PDFdigestOCR algorithm for image-based PDF files uses the Normalized Levenshtein Similarity[18] between strings. The textual strings of each field extracted by the PDFdigestOCR are compared with the original strings generated by the pdf2htmlEX using the marked annotations of the gold dataset.
The results of the PDFdigestOCR image-based evaluation are reported in Table 2. The results of this evaluation show good performance in such elements as the Abstract, Title, Keywords, BibEntry and H3 fields.

Table 2: PDFdigestOCR image-based evaluation

Class	**Avg. Normalized Levenshtein Similarity**
Title	0.9183
Abstract[19]	0.9498
Keywords[20]	0.8856
BibEntry	0.9203
H1 -section title-	0.2628
H2 -subsection title-	0.4145
H3 -subsubsecion title-	0.9463

The extraction errors of the PDFdigestOCR image-based algorithm are due to: 1) OCR recognition errors (e.g. ”a.” for ”a”, ”a” for ”â”, or ”0 Name” for ”* Name”), 2) parsing and extraction errors due to bad OCR Recognitions (e.g, the error of recognizing ”0 Name” instead of ”* Name” can lead to extract a wrong section title header instead of a simple bullet list).

7. Online Demo

A web demonstration of the PDFdigest text-based PDF-to-XML tool can be accessed and tested in the following web address: `http://taln.upf.edu/pdfdigest`[21]. A screenshot of the web interface is shown in Figure 2. The demo allows to browse the local computer disk and select a set of PDF papers (up to a maximum of 10) to be converted to XML.
The output results are returned in 3 different files:

- A XML file with the content extracted and linked to the HTML file.
- An HTML file, generated by the pdf2htmlEX software, that clones the appearance of the original PDF.



Figure 2: Screenshot of the PDFdigest tool online demo.

- A GATE XML file that includes tokenization and sentence segmentation calculated using the Dr Inventor pre-processing library[22] (Ronzano and Saggion, 2015).

8. Discussion

The PDFdigest tool can be easily adapted to different styles of scientific articles and languages. This adaptation can be realized by modifying the following data: 1) language-dependent regular expressions configuration files, 2) hyphenated words lists files, 3) the offsets and thresholds configuration files, and, in some special cases, 4) the finite states that define the logical structure of the paper style and the extraction and consumption rules themselves[23].

9. Conclusions

This paper describes and evaluates PDFdigest, an adaptable layout-aware PDF-to-XML textual content extraction tool. PDFdigest has been designed to extract textual content from both text-based and image-based PDF files. Although there are several fields that would need improvements in extraction, the evaluation of the PDFdigest text-based and the PDFdigestOCR image-based extractors shows a good performance for both algorithms in most of the PDF fields (classes) considered. The classes *Title*, *Abstract*, *Keywords*, and *Bibentry* achieved performances between 88% and 100% of average F1 (text-based) and average Normalized Levenshtein Similarity (image-based) in both approaches. This results indicate that with proper adaptation to paper layouts this tool can be valuable for scientific text mining.

[18] `https://github.com/tdebatty/java-string-similarity`

[19] Includes the detection of both the Abstract header (e.g. “*Abstract:*” in English) and the Abstract text.

[20] Includes the detection of both the Keywords header (e.g. ”*Keywords:*” in English) and the Keywords text.

[21] Note that the online demo only treats text-based PDF files and uses a set of predefined thresholds and offsets and it is not yet possible to modify them online to adapt to different layout styles.

[22] `http://backingdata.org/dri/library/`

[23] To make changes at the finite states and the extraction and consumption rules is necessary to modify the source code

10. Further Work

In the future, we plan to extend the set of content extraction rules to cover special cases of papers that do not follow the logical structure of the standard PDF articles. Further work could explore: 1) the addition of rules that can cover the extraction of more sophisticated layouts in the Computer Science conference papers, 2) an extension of the online demo that will permit the access to the configuration thresholds and offsets in order to allow the end user the option to adapt these parameters to new layouts, 3) the adaptation of the XML output format to the JATS standard[24], 4) the evaluation with different scientific datasets related to Computer Science research, and 5) the comparative evaluation with other state-of-the-art tools with the same evaluation datasets.

11. Acknowledgements

This work was partly funded by the TUNER project (TIN2015-65308-C5-5-R, MINECO/FEDER, UE) and the Spanish MINECO Ministry (MDM-2015-0502).

12. Bibliographical References

Abekawa, T. and Aizawa, A. (2016). SideNoter: Scholarly Paper Browsing System based on PDF Restructuring and Text Annotation. In Hideo Watanabe, editor, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference System Demonstrations, December 11-16, 2016, Osaka, Japan*, pages 136–140. ACL.

Accuosto, P., Ronzano, F., Ferrés, D., and Saggion, H. (2017). Multi-level Mining and Visualization of Scientific Text Collections. In *Proceedings of the 6th International Workshop on Mining Scientific Publications (WOSP 2017)*.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May.

Constantin, A., Pettifer, S., and Voronkov, A. (2013). PDFX: Fully-automated PDF-to-XML Conversion of Scientific Literature. In *Proceedings of the 2013 ACM symposium on Document engineering*, pages 177–180. ACM.

Lopez, P., (2009). *GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications*, pages 473–474. Springer Berlin Heidelberg, Berlin, Heidelberg.

Luong, M.-T., Nguyen, T. D., and Kan, M.-Y. (2012). Logical Structure Recovery in Scholarly Articles with Rich Document Features. *Multimedia Storage and Retrieval Innovations for Digital Library Systems*, 270:2.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Ronzano, F. and Saggion, H. (2015). Dr. inventor framework: Extracting structured information from scientific publications. In Nathalie Japkowicz et al., editors, *Discovery Science*, pages 209–220, Cham. Springer International Publishing.

Ronzano, F. and Saggion, H., (2016). *Knowledge Extraction and Modeling from Scientific Publications*, pages 11–25. Springer International Publishing, Cham.

Tkaczyk, D., Szostek, P., Dendek, P. J., Fedoryszak, M., and Bolikowski, L. (2014). CERMINE – Automatic Extraction of Metadata and References from Scientific Literature. In *Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on*, pages 217–221. IEEE.

Ware, M. and Mabe, M. (2015). The STM Report: An overview of scientific and scholarly journal publishing.

13. Language Resource References

UPC-TALP, UA-InterNostrum, UB-CLiC, and UPF-IULA . (2011). *Spanish LMF Freeling Lexicon*. UPC-TALP, UA-InterNostrum, UB-CLiC, and UPF-IULA, 1, ISLRN 587-970-144-991-4.

[24] http://jats.niso.org

Automatic Identification of Research Fields in Scientific Papers

Eric Kergosien[1], Amin Farvardin[5], Maguelonne Teisseire[3], Marie-Noëlle Bessagnet[2], Joachim Schöpfel[1,6], Stéphane Chaudiron[1], Bernard Jacquemin[1], Annig Lacayrelle[2], Mathieu Roche[3,4], Christian Sallaberry[2], Jean Philippe Tonneau[3,4]

[1] Univ. Lille, EA 4073 – GERiiCO, F-59000 Lille, France
prenom.nom@univ-lille.fr
[2] LIUPPA, Université de Pau et des Pays de l'Adour, Pau, France
prenom.nom@univ-pau.fr
[3] TETIS, Univ. Montpellier, APT, Cirad, CNRS, Irstea, Montpellier, France
prenom.nom@teledetection.fr
[4] Cirad, Montpellier, France
prenom.nom@cirad.fr
[5] LAMSADE, Université Paris-Dauphine, Paris, France
MohammadAmin.Farvardin@dauphine.eu
[6] ANRT, Lille, France
Joachim.Schopfel@univ-lille3.fr

Abstract

The TERRE-ISTEX project aims to identify scientific research dealing with specific geographical territories areas based on heterogeneous digital content available in scientific papers. The project is divided into three main work packages: (1) identification of the periods and places of empirical studies, and which reflect the publications resulting from the analyzed text samples, (2) identification of the themes which appear in these documents, and (3) development of a web-based geographical information retrieval tool (GIR). The first two actions combine Natural Language Processing patterns with text mining methods. The integration of the spatial, thematic and temporal dimensions in a GIR contributes to a better understanding of what kind of research has been carried out, of its topics and its geographical and historical coverage. Another originality of the TERRE-ISTEX project is the heterogeneous character of the corpus, including PhD theses and scientific articles from the ISTEX digital libraries and the CIRAD research center.

Keywords: text mining, natural language processing, geographical information retrieval, scientometrics, document analysis

1. Introduction

Widespread access to digital resources, via academic platforms – for example, the Gallica project (BnF)[1], the ISTEX[2] platform, electronic theses and dissertation repositories (TEL), content federation services (Isidore), or electronic publishing platforms (OpenEdition) – offers numerous possibilities for users. The ISTEX initiative was launched to create innovative information retrieval services and provide access to digital resources through different search processes. The increasing adoption of information and communication technologies by researchers in different academic disciplines, especially in the social sciences and humanities, is changing the conditions of knowledge appropriation. Digital humanities have made it possible to develop platforms, providing researchers with large volumes of academic papers and with support services to add value and make use of them (e.g., the TELMA[3] application).

The TERRE-ISTEX project was developed in this research context and proposes (1) to identify the covered territories and areas from scientific papers available in digital versions within and outside the ISTEX library, and (2) to evaluate the academic disciplines involved (e.g. history, geography, information sciences, etc.) as well as the evolution of disciplinary and multi-disciplinary research paradigms in selected topics. The results of this project will help scientists working on a given territory (areas at different scales, such as township, region, country, or continent) to retrieve papers on the same territory.

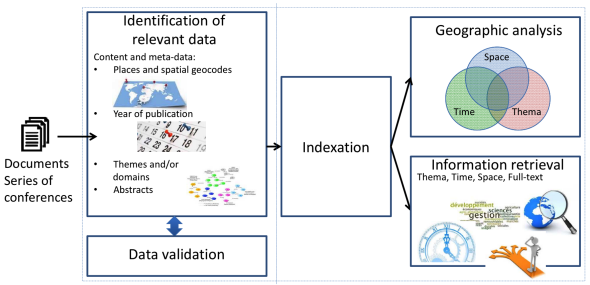


Figure 1: Generic Approach to Multidimensional Analysis of Scientific Corpus

2. The TERRE-ISTEX Project

The generic approach used in the TERRE-ISTEX project is described in Figure 1. Regardless of any scientific publications corpus, a first step is to standardize textual documents. A second step is to identify, in metadata and contents of documents, the research fields as well as the scientific disciplines involved. The research field is defined as the locations constituting the territory in which the research is conducted on a given date or period of time.

In the experimental section, we present a Web application called SISO to help domain experts to analyze and to vali-

[1] http://gallica.bnf.fr/
[2] http://www.istex.fr/
[3] http://www.cn-telma.fr/

date annotated text samples. The indexed and validated data are then integrated into a documentary database in order to allow, on the one hand, the analysis of data and, on the other hand, the retrieval of papers on the same field and/or the same period and/or the same discipline or sub-domain using a Web-demonstrator of geographical information search.
The next section describes the corpus used within the framework of our project.

2.1. The corpus

Elaborating a corpus is a major prerequisite in the process of analysis and information retrieval. We targeted three sources of scientific papers, namely the ISTEX[4] platform, the Agritrop[5] open archive (CIRAD[6]), and a sample of PhD theses from the ANRT[7] with associated metadata available on the portal `theses.fr`.
We conducted a case study on the topic of climate change in Senegal and Madagascar. We collected an initial corpus of documents from the ISTEX platform (about 170,000 documents) using queries with the following keywords: "climate change", "changement climatique", "Senegal", "Sénégal", "Madagascar". From the same keywords, we collected 400 theses from the ANRT database. Finally, the documents from Agritrop are related to studies dealing with Madagascar and the Senegal River. The 92,000 references and 25,000 full-text documents include different types of academic papers, i.e., scientific publications (i.e., articles, etc.), grey literature (e.g. reports, etc.), and technical documentation. Each document is associated with metadata, including an abstract.
The metadata formats of the different items depend on the document origin: MODS[8] (ISTEX), XML based on the Dublin Core (CIRAD), and RDF (ANRT). The corpus is multilingual: most of the documents are either in French or in English, but there are also documents using both languages (for example, with a summary in French and a summary in English). The corpus is thus composed of multilingual and heterogeneous documents, both in terms of content and format.

2.2. An important step to standardize data

2.2.1. The process TERRE-ISTEX

Initially, the process developed by the TERRE-ISTEX project is applied to metadata and abstracts. Because of the data heterogeneity, we chose to standardize metadata using the pivotal MODS (Metadata Object Description Schema) format, recommended on the ISTEX platform. The MODS format has several advantages: (a) it is suitable for describing any type of document and any medium (digital or print); (b) it is richer than the Dublin Core; and (c) it is similar to the models for structuring bibliographic information used in libraries (e.g. MARC format). For these reasons, we apply, in a first step, an algorithm of model transformation to those 92,400 documents of the corpus which do not comply with this format. The second step concerns the annotation in the abstracts of spatial, temporal, and thematic entities. This step is detailed below. As a result, the MODS-TI data model expands the MODS format to describe spatial, temporal, and thematic entities extracted from documents. The MODS-TI model is detailed in the following section. Step 3 implements a new algorithm for transforming the MODS-TI format in order to create indexes so that all data can then be processed in the final stages of analysis and information retrieval.

2.2.2. The TERRE-ISTEX data model

The TERRE-ISTEX data model expands the MODS format to describe spatial, temporal, and thematic information extracted from documents and from corresponding metadata. The choice of MODS was determined by the fact that MODS is the main format on the ISTEX platform and by the advantages described above.
We added three tag sub-trees to a MODS document:

- `<spatialAnnotations>`,
- `<temporalAnnotations>`,
- `<thematicAnnotations>`.

In the following, we give an example of the sub-tree for spatial entities (ES). The tag `<spatialAnnotations>` contains a set of spatial entities (tag `<es>`), with the annotated text for each of them. (tag `<text>`) as well as its spatial footprint obtained by querying the GeoNames resource. The corresponding DTD is shown in Figure 2.

2.3. Named Entity annotation

2.3.1. Spatial entities

In the TERRE-ISTEX project, the methodology is based on linguistic patterns for the automatic identification of spatial entities (ES) (Tahrat et al., 2013). An ES consists of at least one named entity and one or more spatial indicators specifying its location. An ES can be identified in two ways (Sallaberry et al., 2009): as an absolute ES (ESA), it is a direct reference to a geo-locatable space (e.g. "the Plateau d'Allada"); as a relative ES (ESR), it is defined using at least one ESA and topological spatial indicators (e.g. "in Southern Benin"). These spatial indicators represent relationships, and five types of relationships are considered: orientation, distance, adjacentness, inclusion, and geometric figure that defines the union or intersection, linking at least two ES. An example of this type of ES is "near Paris". Note that ESA and ESR integrated representation significantly reduces the ambiguities related to the identification of the right spatial footprint. Indeed, taking into account spatial indicators (e.g. "river" for "Senegal River") allows us to identify in GeoNames the right spatial footprint. To deal with distinct spatial entities with the same name (e.g. Bayonne in France and Bayonne in the United States), a disambiguation task is proposed in order to analyze the context in the textual documents (Kergosien et al., 2015).
In order to identify spatial entities, we apply and extend a Natural Language Process (NLP) adapted to Geographic Information Retrieval (GIR) domain. In this context, some rules (patterns) of (Tahrat et al., 2013) have

[4] `http://www.istex.fr/category/plateforme/`
[5] `https://agritrop.cirad.fr/`
[6] `http://www.cirad.fr`
[7] `https://anrt.univ-lille3.fr/`
[8] `http://www.bnf.fr/fr/professionnels/f_mods/s.mods_presentation.html`

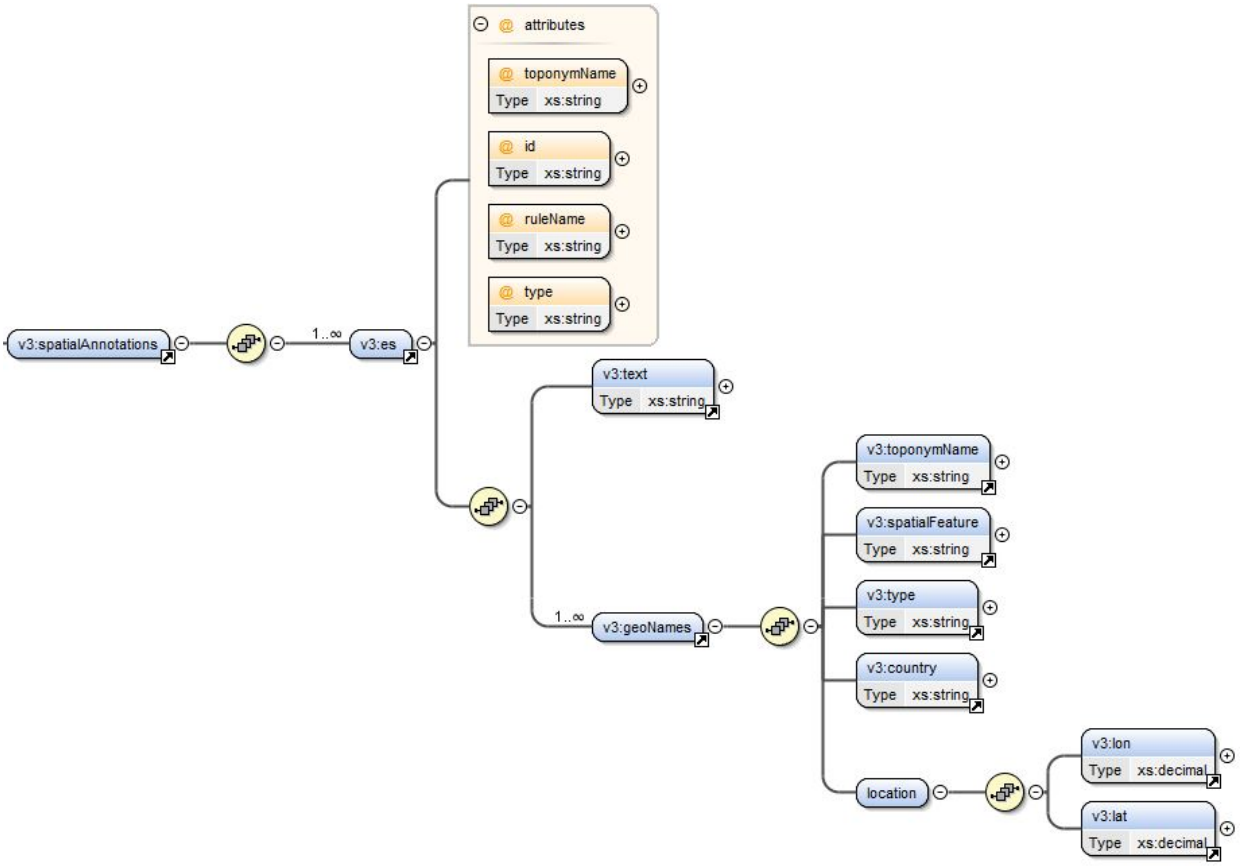


Figure 2: DTD describing the tag `<spatialAnnotations>`

been integrated and improved to identify absolute and relative spatial entities in French (e.g. "sud-ouest de l'Arabie Saoudite" (ESR), "dans la région du Mackenzie" (ESR), "golfe de Guinée" (ESA), "lac Eyre" (ESA)). These rules have been translated in English to analyze English corpora (e.g. "Willamette River" (ESA), "Indian Ocean" (ESA), "Wujiang River Basin" (ESA)) of TERRE-ISTEX project. Moreover, we propose new types of rules in order to identify Organization (for example, an "Organization is followed by an action verb"). These different rules developed with GATE1 enable to disambiguate extracted entities and then to improve named entities recognition.

2.3.2. Thematic and temporal entities

In order to enhance the knowledge identified in metadata and to specify the sub-domains, we apply modules in text mining to the content of the publications to extract domain vocabularies. First, we use domain semantic resources for lexical annotation. As, in our case, the thematic entities to be annotated are linked to climate change, we rely on the Agrovoc resource (Rajbhandari and Keizer, 2012). Agrovoc is formalized in XML SKOS. In the indexing phase, we mark for each term the content of an article coming from Agrovoc with the terms "used for" and the generic terms, information that will be exploited in the search engine. In the long term, we aim to propose a generic approach by giving the possibility of easily integrating a new semantic domain resource formalized in XML SKOS. Also, we plan to integrate the BioTex module developed by the TETIS team in Montpellier (Lossio-Ventura et al., 2016) combining statistical and linguistic approaches to extract terminology from free texts. The statistical information provides a weighting of the extracted applicant terms. However, the frequency of a term is not necessarily an appropriate selection criterion. In this context, BioTex proposes to measure the association between the words composing a term by using a measure called C-value while integrating different weights (TF-IDF, Okapi). The goal of C-value is to improve the extraction of multi-words terms that are particularly suitable for specialist fields.

For the temporal entities, we have integrated the HeidelTime processing chain (Strötgen and Gertz, 2013) to mark calendar entities (dates and periods). HeidelTime is a free, rule-based, time-sensitive labeling system for temporal expressions, available in several languages. Regarding English, several corpora of documents (i.e., scientific articles, press) have been treated (Strötgen and Gertz, 2013). The evaluation of this system shows better results for the extraction and standardization of temporal expressions for English, in the context of the TempEval-2 and TempEval-3 campaigns (UzZaman et al., 2013) and extended to 11 languages including French (Moriceau and Tannier, 2013). HeidelTime produces annotations in the ISO-TimeML format, which distinguishes between four categories of temporal expressions: dates, times, durations and frequencies. Since our objective is to know the periods covered in the documents, we are only interested in temporal expressions with a calendar connotation.

3. Experiments

3.1. First experiments

The sociologists and geographers working in the project evaluated the spatial entities extraction process. The French corpus is composed of 4,328 words (71 spatial features and 117 organizations). The evaluations (with classical measures, i.e., precision, recall, and F-measure) have been investigated by comparing the manual extraction done by experts with the web service results. For spatial entities, we obtain a good recall (91%) and an acceptable precision (62%). The F-measure is 0.74. The great majority of Spatial Features (SF) are extracted but there are still some errors. The rules to identify organization are very efficient and give high precision (85%) but the value of recall is lower (67%). The F-measure for organization identification is 0.74. The rules for organization extraction seem well adapted to the domain but they have to be extended in order to improve the recall that remains low.

Moreover, in order to evaluate our approach of annotation of the ESA and ESR on a scientific corpus, we manually annotated 10 scientific articles in French, and 10 in English from the corpus on the 'climate change' topic. Items are randomly selected. The documents averaged 230 words and contained 39 spatial entities (i.e., ESA, ESR). We then annotated these documents with two processes: the CasEN chain, a reference in the field for the marking of named entities (Maurel et al., 2011), and ours.

We obtained good results in terms of precision, recall, and F-measure with our process (see Tables 1 and 2). Nevertheless, we have issues with disambiguation of named entities (Organization and spatial entities) in English and we have to improve our process for scientific articles in English.

It is significant that the results coming from the CasEN chain are far better when processing French than English too. Assuming this difference is not directly linked to the test corpus, and besides the above disambiguation issue, we consider English linguistic specificities as an explanation. In particular, links between items of a multi-word spatial entity are seldom made explicit, neither by a common morphological feature, nor by a linking preposition.

	ESA, ESR (TERRE-ISTEX)	ESA, ESR (CasEN)
Precision	100%	93%
Recall	90%	77%
F-Measure	.947	.842

Table 1: Evaluation of spatial entity annotation on 10 articles from the French corpus

	ESA, ESR (TERRE-ISTEX)	ESA, ESR (CasEN)
Precision	90%	94%
Recall	60%	53%
F-Measure	0.72	0.68

Table 2: Evaluation of spatial entity annotation on 10 articles from the English corpus

In order to validate these initial results, we are currently working on an evaluation concerning a corpus of 600 scientific articles from the ISTEX platform, 300 written in French and 300 in English. The articles were randomly selected from the 40,000 scientific articles related to the theme of 'climate change'. In this case, on the first 140 documents manually annotated by experts working in the project, we obtain a precision of 78%. The complete evaluation process is in progress. Producing a bigger annotated scientific corpus (with spatial entities, temporal entities and topics) is also an objective for other ISTEX research groups.

To help experts to analyze corpora, and particularly information related to territories, a Web application SISO[9] has been developed.

3.2. SISO Web Application to index and analyze corpora

The SISO Web application (Figure 3) allows users to upload corpora, to index documents with specific web services in order to mark different kinds of information (spatial features, organizations, temporal features, and themes), to visualize and to correct the results, and to download validated results in XML format. More specifically, it is possible to upload corpora (frame 1), each marked corpus is saved on the server and automatically available in the web application (frame 2). After having downloaded documents, users can select the marked features (frame 5), see the results on the selected documents in frame 3. By selecting different categories from frame 5, the related marked information will be kindles in frame 3 and listed by type in frame 4. In case of finding any mistakes, users can unselect marked information (frame 4). Finally experts can export the selected corrected documents as a new corpus by clicking the top right button. The downloaded corpus, in XML MODS format, consists of selected documents with the marked information except those removed by the user.

The administration page allows users to upload, edit, and delete pipelines defined in the GATE format. It is also possible to remove processed corpora, to edit the uploaded pipeline rules and the available lexicons.

In order to provide experts with a web tool to process big data related to their domain, the TERRE-ISTEX approach was improved. The performance of the system, tested on 8,500 documents, are presented below:

- Temporal entity annotation: 8,196 seconds,
- Agrovoc entity annotation: 1,606 seconds,
- Search of concept and linked concept using the offline Agrovoc ontology: 36 seconds,
- Spatial entities annotation (French and English): 4,940 seconds,
- JSON index file generation: 55 seconds.

The global process takes 16,105 seconds for processing all documents, i.e., 1.9 seconds per document. This result is very encouraging.

[9] http://geriico-demo.univ-lille3.fr/siso/

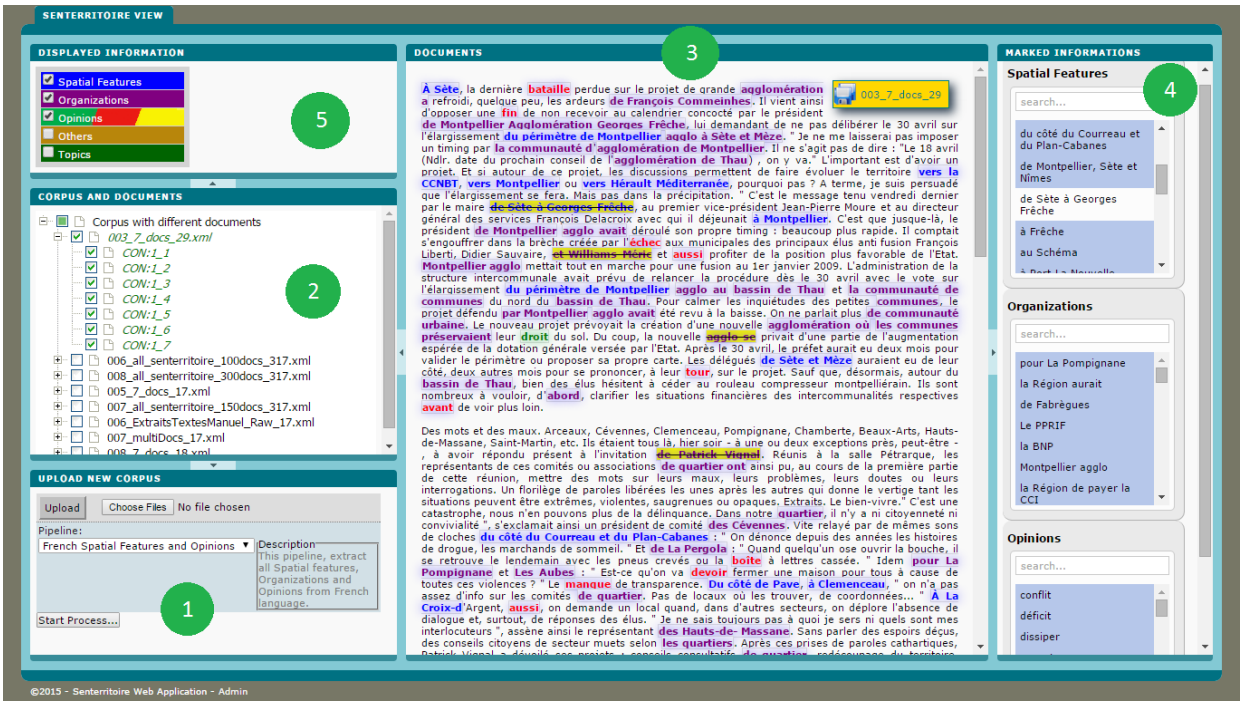


Figure 3: The Web application called SISO

4. Conclusion

In this article, we describe a method to deal with scientific literature on climate change from three different corpora of scientific papers. One main issue was the standardizing of data. Therefore, we have developed algorithms and a unified data model. Then, we have defined an automatic process to identify information related to a territory (spatial, temporal, and thematic information) in the documents.

Up to now, the entire corpus is indexed in JSON format. We are currently working on the enrichment of the temporal entity marking chain to integrate the BioTex tool, and on the extension of tagging assessments of marked (spatial, temporal and thematic) entities in voluminous corpus. Recently, we started to index our data with the Lucene-based search engine Elasticsearch1. Elasticsearch will facilitate the test of defined work use-cases. The main goal is to help researchers analyze big data corpora, and especially those who are interested in research related to a territory.

In our future work, we plan to use machine-learning approaches in order to improve the disambiguation process of spatial entities (i.e., ES vs. Organization). More precisely, based on our previous work (Tahrat et al., 2013), we will propose to integrate the patterns described in section 3.1. as features in the supervised learning model based on the SVM algorithm. Then, we plan to compare our final model to the state-of-the-art, and specifically to the ISO-Space model produced to annotate Spatial Information from textual data (Pustejovsky, 2017).

5. Acknowledgements

This work is funded by ISTEX (https://www.istex.fr/), SONGES project (Occitanie and European Regional Development Funds), and DYNAMITEF project (CNES).

6. Bibliographical References

Kergosien, E., Alatrista-Salas, H., Gaio, M., Güttler, F., Roche, M., and Teisseire, M. (2015). When Textual Information Becomes Spatial Information Compatible with Satellite Images. In *Proceedings of the 7th International Conference on Knowledge Discovery and Information Retrieval*, KDIR 2015 at IC3K, pages 301–306, Lisbon, Portugal.

Lossio-Ventura, J. A., Jonquet, C., Roche, M., and Teisseire, M. (2016). Biomedical Term Extraction: Overview and a New Methodology. *Information Retrieval Journal*, 19(1-2):59–99.

Maurel, D., Friburger, N., Antoine, J.-Y., Eshkol-Taravella, I., and Nouvel, D. (2011). CasEN: a transducer cascade to recognize French Named Entities. *TAL*, 52(1):69–96.

Moriceau, V. and Tannier, X. (2013). French resources for extraction and normalization of temporal expressions with HeidelTime. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, LREC 2014, pages 3239–3243, Reykjavík, Iceland.

Pustejovsky, J. (2017). ISO-Space: Annotating Static and Dynamic Spatial Information. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 989–1024. Springer, Dordrecht. DOI: 10.1007/978-94-024-0881-2_37.

Rajbhandari, S. and Keizer, J. (2012). The AGROVOC Concept Scheme: A Walkthrough. *Journal of Integrative Agriculture*, 11(5):694–699.

Sallaberry, C., Gaio, M., Lesbegueries, J., and Loustau, P. (2009). A semantic approach for geospatial information extraction from unstructured documents. In Arno Scharl et al., editors, *The Geospatial Web*, Advanced Information and Knowledge Processing, pages 93–104. Springer, London.

Strötgen, J. and Gertz, M. (2013). Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Tahrat, S., Kergosien, E., Bringay, S., Roche, M., and Teisseire, M. (2013). Text2geo: From Textual Data to Geospatial Information. In *Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics*, WIMS '13, pages 23:1–23:4, New York, NY, USA. ACM.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, USA. ACL.

A «Portrait» Approach to Multichannel Discourse

Andrej A. Kibrik, Olga V. Fedorova
Institute of Linguistics RAS & Lomonosov Moscow State University
B. Kislovskij per., 1 & Leninskie Gory, 1, Moscow, Russia
aakibrik@gmail.com, olga.fedorova@msu.ru

Abstract

This paper contributes to the research field of multichannel discourse analysis. Multimodal discourse analysis explores numerous channels involved in natural communication, such as verbal structure, prosody, gesticulation, facial expression, eye gaze, etc., and treats them as parts of an integral process. Among the key issues in multichannel studies is the question of the individual variation in multichannel behavior. We address this issue with the help of a multichannel resource "Russian Pear Chats and Stories" that is currently under construction (multidiscourse.ru). This corpus is based on a novel methodology of data collection and is produced with the help of state of the art technology including eyetracking. To address the issue of individual variation, we introduce the notion of a speaker's individual portrait. In particular, we consider the Prosodic Portrait, the Oculomotor Portrait, and the Gesticulation Portrait. The proposed methodology is crucially important for fine-grained annotation procedures as well as for accurate statistic analyses of multichannel data.

Keywords: multichannel discourse, prosody, gesticulation, eye gaze, a speaker's portrait

1. Introduction

People communicate with each other, using words, intonation, gestures, gaze, facial expression (Kress, 2002; Loehr, 2012; McNeill, 2005; Goldin-Meadow, 2014; Church et al. eds., 2017, inter alia), see Fig. 1.

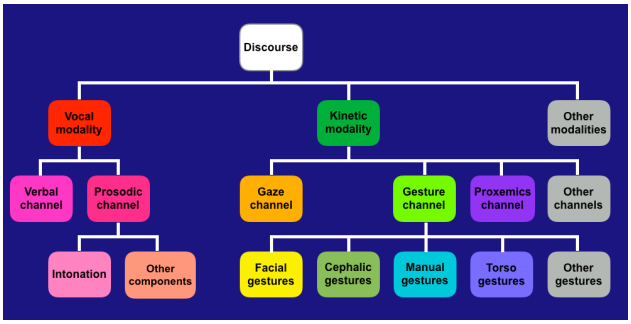


Figure 1: Spoken multichannel discourse.

All of these communication channels are employed simultaneously and in conjunction with each other. Therefore, everyday human communication is a multichannel process. We are immersed in this communication throughout our lives, but still it remains an underexplored phenomenon. There are at least two causes leading to this situation. First, the process of multichannel communication is ephemeral and passes by without leaving material traces. Second, the content of various communication channels traditionally belongs to the competence of different disciplines, weakly connected with each other. In particular, the verbal component is studied by linguists, while gestures and eye movements are explored, primarily, by psychologists. Our research group set as its goal a comprehensive study of multichannel discourse, based on an integrated theoretical and methodological approach.

A "multi-modal corpus" is defined as "an annotated collection of coordinated content on communication channels including speech, gaze, hand gesture and body language, and is generally based on recorded human behavior" (Foster, Oberlander, 2007: 307–308). As compared to monomodal corpora that already have a substantial history and tradition, multimodal corpora are still at their incipient stage. The most natural data have been assembled in the Fruit Carts Corpus that contains 240 videorecordings of 12 participants, each four to eight minutes long (Aist et al., 2012), the corpus D64, created for studies of everyday communication (Campbell, 2009), the InSight Interaction Corpus consisting of 15 recorded face-to-face conversations 20 min long each (Brône, Oben, 2015), as well as corpora created in the tradition of Conversation Analysis (Mondada, 2014; 2016, inter alia).

When exploring any linguistic phenomenon, one addresses two opposite (and complementary) issues: general trends, on the one hand, and individual variation, on the other. In the domain of multichannel discourse, exploration of either of the two issues requires a good quality, representative corpus. At this time we have created the resource "Russian Pear Chats and Stories" that includes a number of recorded sessions of natural communication between several participants, as well as vocal and kinetic annotations of these sessions. The sessions were recorded with the help of original technical solutions, including high quality audio and video recording, as well as eye tracking methods. The vocal (verbal and prosodic) annotation used in this project follows the principles previously developed for spoken Russian discourse, for more detail see Kibrik, Podlesskaja, 2009 and the website spokencorpora.ru. Within the framework of the present project, we have developed principles of kinetic and oculomotor annotation (see Section 2).

In this paper we address ***individual differences*** and propose a "portrait" approach to multichannel discourse. We demonstrate that individual portraits of interlocutors is a necessary prerequisite for fine-grained annotation and for in-depth analysis of multichannel discourse (see Sections 3–5).

2. Russian Pear Chats and Stories Corpus

2.1 Stimulus Material

We use the well known Pear Film (Chafe ed., 1980) as the stimulus material for collecting the data. The film contains no speech, and the events shown are relatively clear to any viewer. The film was constructed so that the shown scenes incline participants to describe landscape, explain cause-effect relations, account for the characters' thoughts and emotions, and resolve ambiguities.

2.2 Data Collection Setup

We have developed a new ***method*** of collecting data on the basis of the Pear Film. Each session, including the instructions we provided and participants' filling out the written consent, lasted for about one hour. Each session involved four participants with fixed roles, see Fig. 2.

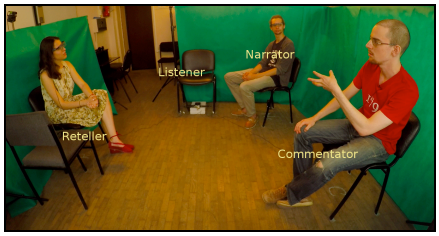


Figure 2: Data collection setup. Cover shot.

Three of them – the Narrator (N), the Commentator (C), and the Reteller (R) – took part in the main part of the recording, while the fourth participant, the Listener (L), joined them towards the end. The Narrator and the Commentator each watched the film on a personal computer and tried to memorize the plot and the details as precisely as possible. Then an interaction, also involving the Reteller, began. The Narrator told the Reteller about the plot of the film; this is a monologic stage – *first telling*. During an interactive stage (*conversation*), the Commentator supplied additional details and corrected the Narrator's story where necessary; the Reteller checked his/her understanding of the plot, asking questions to the Narrator and the Commentator. Then another monologic stage, *retelling*, followed, during which the Reteller was retelling the film to the Listener. Finally, the Listener wrote down the content of the film, as s/he had understood it from the Reteller's account.

2.3 Recording Devices

The participants' talk was recorded with the help of a six-channel recorder ZOOM H6 Handy Recorder (96 kHz / 24 bit). Three channels were used to record three speakers individually, with the lapel microphones SONY ECM-88B. Two more channels were used to produce a general recording of the whole speech signal, by using a stereo mic provided with the recorder.

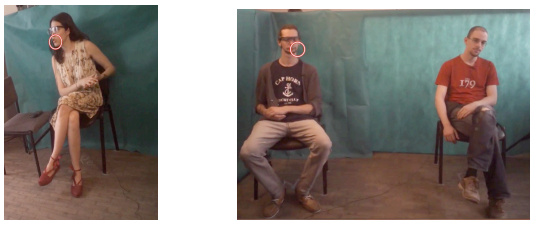
Three industrial video cameras JAI GO-5000M-USB (100 frames per second and 1392x1000 pixels) recorded three participants, shooting individually from a frontal perspective (see Fig. 3). These cameras use the mjpeg format of recording, which is free of interframe compression; this is a crucial prerequisite for subsequent frame-by-frame annotation. In addition, the camera GoPro Hero 4 (50 frames per second and 2700x1500 pixels) was used to record the whole scene (see Fig. 2 above).



Narrator Commentator Reteller

Figure 3: Individual frontal shots.

In order to record eye gaze, two head-mounted eyetrackers were used (Tobii Glasses II, 50 Hz and 1920x1080 pixels). The Narrator and the Reteller were wearing eyetrackers, see Fig. 4. The eyetrackers provide two types of data: (i) videofiles produced by an inbuilt scene camera and (ii) data files representing eye movements. The screenshots in Fig. 4 result from an overlay of videofiles from the scene camera and the gaze coordinates from the data files; the circles are generated by the eyetrackers and indicate the targets of interlocutors' gaze.



From the N's tracker From the R's tracker

Figure 4: Video scene, as recorded by a camera built into the eyetrackers, with a superimposed marker of visual attention.

2.4 Participants and Corpus Size

The corpus includes 24 conversations among 96 Russian native speakers aged 18-36. It consists of 9 hours of recording (the average length of a recording was 24 min). We work with the data coming from 10 different sources: three video files from the individual industrial cameras, one video file from the cover shot camera, two files from the eyetrackers, and four audio files.

2.5 Annotations

2.5.1 Vocal Annotation

Verbal-prosodic (vocal) annotation consists of spoken discourse divided into relevant elements (elementary discourse units – EDUs, words, absolute and filled pauses, nonspeech sounds), as well as of attributes assigned to EDUs and their parts (pitch accents, accelerated tempo, reduced pronunciation, lowered tonal register, etc.); see Kibrik 2011 for details. As a result of vocal annotation (performed by Nikolay Korotaev and Vera Podlesskaya) transcripts were obtained, in the form of text documents, as well as textgrid files with multi-layer annotation prepared with the help of speech analyzing program Praat (fon.hum.uva.nl/praat) and reflecting temporal anchoring of all annotated phenomena.

2.5.2 Annotation of Manual Gestures

For the transcription of the videodata we used the annotation software ELAN (lat-mpi.eu/tools/elan/) and followed the annotation system developed in (Litvinenko et al., 2017); annotation was done by Alla Litvinenko and Julia Nikolaeva. The proposed annotation scheme proceeds from the basic level of the distinction between motion and stillness to more complex structures at the next level. At the first stage we annotate simplest motion units, or movements, for each hand separately. Each of these movements functions as a gesture phase (Kendon, 1980; 2004; Ladewig, Bressem, 2013), a self-adaptor phase, or a position change movement. All the stillness intervals are also annotated and classified. At the next stage, we annotate hand postures. Gesture and movement chains are the final component of the scheme. A gesture chain is an uninterrupted series of gestures; a movement chain is an uninterrupted series of movements.

The proposed annotation scheme was originally created for manual movements. At the same time, the scheme is broadly applicable to other gestural components, in particular head movements.

2.5.3 Annotation of Gaze

In the course of annotation of the eye gaze component we conducted the export of the eyetracking data onto the video scene, and used the Tobii Pro Glasses Analyzer program to extract the information on the temporal structure of all fixations longer than 100 ms (that is, a participant's fixation on a target must last for at least 100 ms to be recognized as a gaze event). The possible targets include "Interlocutor" ("Narrator / Reteller", "Commentator" or "Listener"), further subdivided into "face", "hands", "torso", and "other", and "Surroundings", see Fedorova 2017 for more detail.

2.5.4 Multilayer Annotation

Fig. 5 provides an example of a full multichannel annotation, including the above discussed channels, as well as additional components of phonetic realization, facial expressions, torso gestures, and proxemics; for more details see our website multidiscourse.ru/annotation.

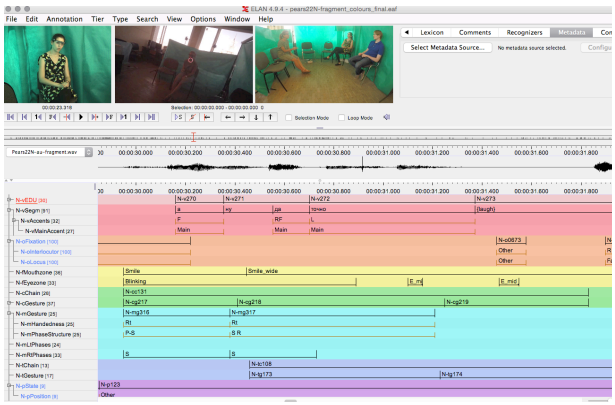


Figure 5: Multilayer annotation.

3. Prosodic Portrait

Many prosodic phenomena are of a relative, rather than absolute, nature: their specific realizations can be assessed and identified only with respect to neutral characteristics of a given speaker's voice. In particular, two kinds of the falling intonation must be distinguished: a final falling (so-called period intonation) and a non-final falling (a falling comma intonation). In order to posit punctuation marks in a transcript belonging to a certain speaker, one needs to explore and describe the speaker's prosodic system. That is, a complete account of a spoken discourse presupposes a speaker's Prosodic Portrait, i.e. a range of his or her prosodic characteristics. Compiling such a portrait is stage zero in the work on a transcript, preceding vocal annotation as such; see Kibrik, 2009, 2011 for further details.

Possible components of a Prosodic Portrait are shown in Table 1; for the annotation in Word and Praat see multidiscourse.ru/corpus. Minimal and maximal F0 values are point values, as they indicate the limits of a range in which F0 of a speaker's voice fluctuates. Other measurements are interval values (rare and extreme outliers are ignored). Where possible, e.g. in the case of a target level of falling, such interval values are centered around a median value (which also is a modal value). In other instances specific values are so widely scattered that no median or modal value is indicated; for example, in this particular speaker a wide interval is found in his target levels of rises in the canonical comma intonation.

Minimal F0 value, Hz	91
Maximal F0 vaue, Hz	214
Standard level of EDU onsets, Hz	140±5
Target level of final fallings, Hz	105±3
Target level of non-final fallings, Hz	115±3
Target level of rises in the canonical comma intonation, Hz	140~185
Standard level of falling on post-accent syllables, typical of the canonical comma intonation, Hz	120±5
Target level of rises in the "three dots" intonation, Hz	118~132

Table 1: Prosodic Portrait for participant 04C.

4. Oculomotor Portrait

Whereas a Prosodic Portrait is an important element of the transcription process, an oculomotor portrait fulfils a different function: it is created after the procedure of semi-automatic annotation and serves at the stage of data analysis. Individual differences between speakers are so substantial that one cannot properly compare the data belonging to different participants without averaging them, or, more precisely, without normalizing them and reducing them to quantiles for the purposes of the subsequent analysis of variance. This is what is done with the help of oculomotor portraits. Table 2 illustrates a so-called Standard Oculomotor Portrait of Narrator 04, involving the data of the summary quantity of fixations throughout the duration of the session; the summary duration of the fixations; mean, minimal, and maximal durations; as well as 25%, 50%, and 75% quantiles. In

addition, the portrait also includes analogous distributions of fixations on R, C, L, and the Surroundings. An Extended Oculomotor Portrait includes analogous data on the participants' fixations distributions during the stages of *first telling*, *conversation* and *retelling*, as well as a more complex distribution table of the fixations on the participant's body parts: face, hands, torso, and other. (Application developer is Ivan Zherdev, see github.com/ivan866/readTobiiGlasses; for the annotation in Excel and ELAN see multidiscourse.ru/corpus).

Total duration	1170.667
Total count	2249
Mean duration & std	0.52 & 0.66
Min, Quantile 25, 50, 75, Max	0.06, 0.16, 0.26, 0.58, 10.5
R: count & ratio	1033 & 0.46
R: duration & ratio	848.55 & 0.72
R: mean & std	0.82 & 0.85
R: Min, Quantile 25, 50, 75, Max	0.06, 0.24, 0.5, 1.16, 10.5
C: count & ratio	129 & 0.057
C: duration & ratio	43.8 & 0.036
C: mean & std	0.33 & 0.28
C: Min, Quantile 25, 50, 75, Max	0.06, 0.16, 0.24, 0.4, 1.32
L: count & ratio	16 & 0.007
L: duration & ratio	3.28 & 0.003
L: mean duration & std	0.2 & 0.12
L: Min, Quantile 25, 50, 75, Max	0.06, 0.12, 0.19, 0.26, 0.44
Surroundings: count & ratio	1071 & 0.48
Surroundings: duration & ratio	275.02 & 0.23
Surroundings: mean & std	0.26 & 0.21
Surroundings: Min, Quantile 25, 50, 75, Max	0.06, 0.13, 0.2, 0.3, 2.2

Table 2: Standard Oculomotor Portrait for participant 04N (durations shown in seconds).

5. Gesticulation Portrait

Finally, a Gesticulation Portrait fulfils a double function. It is necessary both at the stage of manual gesture annotation and at the stage of gesture analysis. Table 3 contains a Standard Gesticulation Portrait for Reteller 04; for ELAN annotation see the website multidiscourse.ru/corpus. The data necessary at the stage of annotation include: (dis)inclination to stillness; (dis)inclination to self-adaptors; typical amplitude; typical velocity; and preferences in gesture handedness: predominance of two-handed or one-handed gestures, as well as a distribution of one-handed gestures in accordance with handedness. Data necessary for a subsequent comparison include: a summary number of manual gestures throughout a session; their summary duration; their mean, minimal, and maximal durations; as well as 25%, 50%, and 75% quantiles. In addtion, a portrait includes an analogous distribution of manual gestures in accordance with the stages of *first telling*, *conversation*, and *retelling*. An Extended Gesticulation Portrait contains the same data, listed separately for the right and left hands.

Inclination to stillness	low
Inclination to adaptors	high
Amplitude of manual gesture	low
Velocity of manual gesture	high
Two-handed vs. one-handed	0.32 vs. 0.68
One-handed gestures in accordance with handedness: right vs. left	0.68 vs. 0.32
Total count	481
Total gesticulation duration	414.53
Mean duration & std	0.86
Min, Quantile 25, 50, 75, Max	0.09, 0.42, 0.64, 1, 8.7
Conversation: count & ratio	210 & 0.44
Conversation: duration & ratio	198.72 & 0.48
Conversation: Min, Quantile 25, 50, 75, Max	0.16, 0.45, 0.68, 1.15, 8.73
Retelling: count & ratio	271 & 0.56
Retelling: duration & ratio	215.81 & 0.52
Retelling: Min, Quantile 25, 50, 75, Max	0.09, 0.38, 0.62, 0.96, 3.73

Table 3: Standard Gesticulation Portrait for participant 04R (durations shown in seconds).

6. Conclusion

In modern linguistics, as well as in other domains of cognitive science, there is a growing understanding that human communication is inherently multimodal. A research program of multimodal linguistics is gradually evolving (Kibrik, 2010; Knight, 2011; Adolphs, Carter, 2013; Kibrik, Molchanova, 2013; Müller et al. eds., 2014) that treats verbal structure on a par with non-verbal devices. Among non-verbal devices, sometimes only kinetic-visual behaviors are considered. But we find it very important to include prosody (see e.g. Kodzasov, 2009), that is non-segmental aspects of the vocal signal, as a distinct communication channel.

In the course of the project we create a multimodal resource of natural Russian discourse that does not have direct analogs among the contemporary resources. It is created for a wide range of research goals (coordination between units belonging to different channels: clause, EDU, gesture; visual attention and units of communicative behavior; multimodal turn-taking; reinterpretation of "pause" in the multimodal perspective, inter alia). Our resource is based on a novel methodology of data collection and is produced with the help of state of the art technology. It is annotated on the basis of a multilayer discourse transcription system and is freely available online to anyone interested (website multidiscourse.ru). The corpus will serve as a data source for multichannel communication research by linguists, as well as by specialists in other domains of cognitive science.

In this paper we have considered prosodic, gesticulation, and oculomotor portraits of the participants and have demonstrated that such portraits constitute an essential part of both annotation process and data analysis.

7. Acknowledgements

This study is supported by Russian Science Foundation (grant #14-18-03819).

8. Bibliographical References

Adolphs, S., Carter, R. (2013). *Spoken corpus linguistics: From monomodal to multimodal*. N.-Y.: Routledge.

Aist, G., Campana, E., Allen, J., Swift, M., Tanenhaus, M.K. (2012). Fruit carts: A domain and corpus for research in dialogue systems and psycholinguistics. *Computational Linguistics* 38 (3): 469–478.

Brône, G., Oben, B. (2015). InSight Interaction: a multimodal and multifocal dialogue corpus. *Language Resources and Evaluation* 49 (1): 195–214.

Campbell, N. (2009). Tools and resources for visualising conversational-speech Interaction. In *M. Kipp et al. (eds.) Multimodal corpora: From models of natural interaction to systems and applications*. Springer: Heidelberg.

Chafe, W. (Ed.) (1980). *The pear stories: Cognitive, cultural, and linguistic aspects of narrative production*. Norwood: Ablex.

Church, R.B., Alibali, M.W., Kelly, S.D. (Eds.) 2017. *Why gesture? How the hands function in speaking, thinking and communicating*. Benjamins.

Foster, M.E., Oberlander, J. (2007). Corpus-based generation of head and eyebrow motion for an embodied conversational agent. *Language Resources and Evaluation,* 41 (3/4): 305–323.

Goldin-Meadow, S. (2014). Widening the lens: What the manual modality reveals about language, learning, and cognition. *Philosophical Transactions of the Royal society*, 369.

Kendon, A. (1980). Gesticulation and speech: Two aspects of the process of utterance. In M.R. Key (Ed.) *The relationship of verbal and nonverbal communication* (pp. 207–227).

Kendon, A. (2004). *Gesture: Visible Action as Utterance*. New York: Cambridge University Press.

Kibrik, A.A. (2009). A speaker's prosodic portrait. In A.A. Kibrik, V.I Podlesskaja (Eds.) *Night Dream Stories: A corpus study of spoken Russian discourse*. Moscow: Jazyki slavjanskix kul'tur.

Kibrik, A.A. (2010). Multimodal linguistics. In Yu.I. Aleksandrov, V.D. Solov'jev (Eds.) *Cognitive studies*, IV. Moscow: Institute of psychology.

Kibrik, A.A. (2011). Cognitive discourse analysis: local discourse structure. In: M. Grygiel and L.A. Janda (Eds.) *Slavic Linguistics in a Cognitive Framework*. (pp. 273-304). Frankfurt/New York: Peter Lang Publishing Company,

Kibrik, A.A., Podlesskaja, V.I. (Eds.) (2009). *Night Dream Stories: A corpus study of spoken Russian discourse*. Moscow: Jazyki slavjanskix kul'tur.

Kibrik, A.A., Molchanova, N.B. (2013). Channels of multimodal communication: Relative contributions to discourse understanding. In M. Knauff, M. Pauen, N. Sebanz, I. Wachsmuth (Eds.) *Proceedings of the 35th Annual Conference of the Cognitive Science Society* (pp. 2704–2709). Austin, TX: Cognitive Science Society.

Kodzasov, S.V. (2009). *Studies in the field of Russian prosody*. Moscow: Jazyki slavjanskix kul'tur.

Knight, D. (2011). *Multimodality and active listenership: A corpus approach*. London: Bloomsbury.

Kress, G. (2002). The multimodal landscape of communication. *Medien Journal*, 4: 4–19.

Ladewig, S.H., Bressem, J. (2013). A linguistic perspective on the notation of gesture phases. In *C. Müller, A. Cienki, E. Fricke, S.H. Ladewig, D. McNeill, S Teßendorf (Eds) Body — Language — Communication: An International Handbook on Multimodality in Human Interaction* (Vol. 2, pp. 1060–1078). Berlin: Mouton.

Litvinenko, A.O., Nikolaeva Ju.V., and Kibrik, A.A. (2017). Annotation of Russian manual gestures: Theoretical and practical issues. In *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2017"* (pp. 255–268). Moscow: RGGU.

Loehr, D. (2012). Temporal, structural, and pragmatic synchrony between intonation and gesture. *Laboratory Phonology*, 3(1): 71–89.

McNeill, D. (2005). *Gesture and thought*. Chicago.

Mondada, L. (2014). Bodies in action. *Language and Dialogue* 4 (3): 357–403.

Mondada, L. (2016). Challenges of multimodality: Language and the body in social interaction. *Journal of Sociolinguistics*, 20: 336–366.

Müller, C., Fricke, E., Cienki, A., McNeill, D. (Eds.) (2014). *Body – Language – Communication: An International Handbook on Multimodality in Human Interaction*. Berlin: Mouton de Gruyter.

Multilingual Extension of PDTB-Style Annotation: The Case of TED Multilingual Discourse Bank

Deniz Zeyrek[1], **Amália Mendes**[2], **Murathan Kurfalı**[1]
[1]Informatics Institute, Middle East Technical University, Ankara,
[2]Center of Linguistics, University of Lisbon, Lisbon
dezeyrek@metu.edu.tr, amaliamendes@letras.ulisboa.pt, kurfali@metu.edu.tr

Abstract

We introduce TED-Multilingual Discourse Bank, a corpus of TED talks transcripts in 6 languages (English, German, Polish, European Portuguese, Russian and Turkish), where the ultimate aim is to provide a clearly described level of discourse structure and semantics in multiple languages. The corpus is manually annotated following the goals and principles of PDTB, involving explicit and implicit discourse connectives, entity relations, alternative lexicalizations and no relations. In the corpus, we also aim to capture the characteristics of spoken language that exist in the transcripts and adapt the PDTB scheme according to our aims; for example, we introduce hypophora. We spot other aspects of spoken discourse such as the discourse marker use of connectives to keep them distinct from their discourse connective use. TED-MDB is, to the best of our knowledge, one of the few multilingual discourse treebanks and is hoped to be a source of parallel data for contrastive linguistic analysis as well as language technology applications. We describe the corpus, the annotation procedure and provide preliminary corpus statistics.

Keywords: discourse, parallel, multilingual corpus

1. Introduction

Over the last decade, there has been a growing interest in parallel corpora for linguistic research or language technology applications, such as Cettolo et al. (2012), Tiedemann and Nygaard (2004). While most of the existing resources are annotated at the word or syntactic level, as in Bentivogli and Pianta (2005), Haug et al. (2009), corpora enriched with discourse-level annotations are scarce but they exist. For example, Stede et al. (2016) introduce a corpus of parallel argumentative texts (German-English) annotated with respect to RST and SDRT; Popescu-Belis et al. (2012) describe pronoun and connective annotation over Europarl for English and French, and Samy and González-Ledesma (2008) report the development of an annotated corpus of UN documents in English, Spanish and Arabic by adopting a pragmatic perspective. Given the scarcity of discourse-annotated parallel corpora, we believe the community would benefit from new resources which involve various languages. Here we describe TED-Multilingual Discourse Bank (TED-MDB), a corpus of TED talks transcripts involving European languages as well as a non-European language, Turkish, annotated at the discourse level following the PDTB approach (Prasad et al., 2014). Our effort is based on years of discourse research and the principles that are being established in the science of annotation (Hovy and Lavid, 2010; Ide and Pustejovsky, 2017).[1]

1.1. Discourse Relations and the Role of Discourse Connectives in a Text

Discourse is a unit above the sentence level mainly structured in terms of lexical links, anaphoric relations and discourse relations. In this paper, our focus is on discourse relations (DRs), i.e. informational relations such as contrast, elaboration, causal, temporal, etc. that hold between two discourse units. DRs are low-level relations indicating discourse structure and they may be made salient by devices referred to as discourse connectives.[2] Discourse connectives are typically coordinating or subordinating conjunctions (*and, but, because*), prepositional phrases (*in sum*) or adverbs (*similarly*). Human readers easily infer a DR while reading a piece of text, for example, adjacent clauses or sentences often trigger a relation even when a discourse connective is absent.

DRs may be realized inter-sententially, as in example 1 or intra-sententially, as in 2. Examples 1 and 2 are referred to as explicit DRs due to the presence of an explicit connective relating two text spans consisting of clauses or clause complexes. These text spans are referred to as the arguments of a discourse connective (DC). In the examples throughout the paper, explicit DCs are underlined, implicit DCs are shown in parentheses. The arguments are rendered differentially, using italic fonts for the first argument and bold fonts for the second argument. The sense of the relation is shown in square parentheses where relevant. All the examples are from TED-MDB.

1. *About 80 percent of global CEOs see sustainability as the root to growth in innovation* .. <u>But</u> **93 percent see ESG as the future, or as important to the future of their business**.[Comparison:Contrast]
2. *I think it's reckless to ignore these things*, <u>because</u> **doing so can jeopardize future long-term return**. [Contingency:Cause:reason]

Example 3 below shows a text where a DC is lacking; these have been referred to as implicit relations. The inferred DR can be made explicit by inserting a discourse connective;

[1]The researchers and annotators involved in the creation of TED-MDB are members of the consortium initiated by the COST project Textlink (http://textlink.ii.metu.edu.tr). For information regarding the corpus, see http://www.textlink.ii.metu.edu.tr/ted-multilingual-discourse-bank.

[2]Low-level discourse structures differ from high-level discourse structures such as genre and topic Hobbs (1985).

thus, example 3 can be expressed with the discourse adverb *specifically* suggesting that the second clause specifies the meaning of the first clause. Here, it is both the adjacency of the clauses and their lexical content that guide a human reader to insert an appropriate adverb that would make the DR salient.

3. *We have a population that's both growing and aging*; (specifically) **we have seven billion souls today heading to 10 billion at the end of the century**. [Expansion: Level-of-detail-Arg2-as-detail]

1.2. PDTB Principles and the Goals of TED-MDB

Among the currently available frameworks for the investigation and annotation of discourse such as RST (Mann and Thompson, 1988) and SDRT (Asher and Lascarides, 2003), we have settled on PDTB. Our choice is motivated by several factors. For example, unlike RST and SDRT, PDTB aims to reveal discourse coherence and discourse structure at the local level and to the extent the relations are made explicit by DCs (Prasad et al., 2014). We believe the local, lexically based approach provides an easy starting point for the annotation process. Secondly, PDTB has a theory-natural approach, which would be appealing to linguists and NLP researchers working in different theoretical frameworks. Finally, the PDTB annotation guidelines have produced reliable results in corpora of written discourse developed for a range of languages, such as Hindi (Kolachina et al., 2012); Chinese (Zhou and Xue, 2015); Arabic (Al-Saif and Markert, 2010) and Turkish (Zeyrek et al., 2013). The PDTB principles have also been implemented in the creation of a corpus of conversational dialogues in Italian (Tonelli et al., 2010).

In the PDTB approach, DCs are discourse-level predicates with binary arguments. The arguments of a discourse connective (both explicit and implicit) are referred to as Arg1, Arg2 and always have an abstract object interpretation (Asher, 2012), i.e. in order for text spans to be selected as arguments to a DC, they need to be understood as eventualities, facts, propositions, etc. Arg2 is the text span that syntactically hosts the connective while Arg1 is conveniently the other argument.[3]

PDTB annotates texts for 5 major DR types to reveal relations between adjacent text spans: explicit relations, implicit relations, alternative lexicalizations (AltLex), entity relations (EntRel) and no relations (NoRel) (PDTB Group, 2008). By definition, explicit DCs are overtly expressed in texts and are easy to recognize (Pitler et al., 2008), as in examples 1, 2 above. By contrast, implicit relations lack an overt connective, where the relation between discourse units is inferred and shown by a potential explicit connective, as in example 3. AltLexs are any alternative means of lexicalizing a relation and can vary from fixed expressions such as *this is why* (example 4) to free expressions such as *that's the equivalent of* (example 5) (Prasad et al., 2010).

4. *.. long-term value creation requires the effective management of three forms of capital: financial, human, and physical*. This is why **we are concerned with ESG**. [AltLex: Contingency:Cause+Belief: Result+Belief]

5. *.. they yield savings of 23 million dollars in operating costs annually, and avoid the emissions of a 100,000 metric tons of carbon*. That's the equivalent of **taking 21,000 cars off the road** .. [Expansion:Equivalence]

EntRels are relations that hold between two entities rather than eventualities, as in example 6. Finally, NoRels are taken as those cases between adjacent sentences where no discernible DR exists (example 7).

6. *.. CalPERS is another example.* **CalPERS is the pension fund for public employees in California**. [EntRel]

7. Now *over almost eight years, they've outperformed by about two thirds*. So **yes, this is correlation**. [NoRel]

Adopting the PDTB principles, in the TED-MDB project, we annotate 5 DR types together with their binary arguments and sense (or senses), where relevant.[4] In all cases, PDTB's minimality principle is adopted; that is, the argument spans of a relation are annotated as minimally as possible as allowed by the sense of the relation.

While PDTB has inspired discourse annotation projects in various languages mentioned above, each team has developed their own corpora, made use of different annotation tools and used the PDTB annotation scheme in different ways. For example, Turkish Discourse Bank is a multi-genre corpus using a revised subset of the PDTB annotation categories. Arabic, Hindi and Chinese discourse banks involve news texts and again use revised subsets of the PDTB scheme. While such differences may be necessary for developing discourse-level corpora for different languages, they make cross-linguistic comparison difficult. With parallel corpora similarly annotated for discourse phenomena, (a) we may be able to avoid some of the differences arising from annotation efforts carried out separately for each language, (b) help discourse-level corpus annotation efforts to improve our understanding of discourse structure, and in turn (c) help to enhance language technology applications. Thus, our aim in developing TED-MDB is to investigate the phenomena surrounding discourse relations on the basis of texts belonging to the same genre by annotating them similarly across languages. The ultimate goal is to reveal a clearly defined level of discourse structure and semantics for multiple languages for linguistic and natural language processing research.

The structure of the rest of the paper is as follows. §2. introduces the corpus characteristics and its current coverage. §3. describes the annotation procedure, involving training of annotators (§3.1.), the steps in annotating the corpus

[3] Although arguments to a discourse connective can be adjacent or non-adjacent, for the sake of consistency, TED-MDB focuses on arguments that are adjacent to each other. This strategy also facilitates annotation without annotation projection, as explained in §3.2..

[4] Sense is not assigned to EntRels.

(§3.2.), an explanation of how sense annotation is carried out (3.3.), and an evaluation of the corpus covering English, Turkish and Portuguese (§3.4.). §4. describes how PDTB guidelines are extended to capture the interactive nature of TED talks reflected in the transcripts. Finally, §5. summarizes the paper and offers some future directions.

2. The Corpus

TED talks are examples of prepared (possibly scripted), formal monologues (cf. the structure of ICE, (Greenbaum, 1996)) delivered to a live audience. They are video-typed and stored in the TED website. The TED website also provides the transcripts of the talks together with the timestamps on the videos. The transcripts are prepared according to the norms of written language, e.g., they include punctuation and paragraph divisions matching the timestamps. The talks are translated to various languages by named volunteers who are required to follow the instructions provided in the TED website. The translated transcripts are checked by named experts.

We use the WIT3 website (Cettolo et al., 2012) to obtain the transcriptions in the original language, English and their existing translations in the target languages we focus on. We initially identified the common transcripts in English and two target languages (Turkish and European Portuguese). Out of this set of transcripts, we selected the texts for our corpus by reading them to make sure they had the expected level of translation quality and that they were easy to read and comprehend. Ambiguity was tolerated unless it hampered the natural flow of discourse or changed the meaning of the original text. We also ensured that the topics were varied and the texts that mostly relied on images and videos were not selected. Subsequently, the set of parallel texts was expanded to three more languages (German, Russian, and Polish). Table 1 shows the set of transcripts selected and annotated in 6 languages. This is the current coverage of the corpus.

ID	Title and author
1927	The investment of logic for sustainability (Chris McKnett)
1971	The sore problem of prosthetic limbs (David Sengeh)
1976	The flower-shaped starshade that might help us detect Earthlike planets (Jeremy Kasdin)
1978	Embrace the near win (Sarah Lewis)
2009	A glimpse of life on the road (Kitra Cahana)
2150	Social maps that reveal a city's intersections and separations (Dave Troy)

Table 1: Annotated TED talks transcripts

3. Annotation Procedure

In this section we describe the annotation procedure we followed in creating the TED-MDB corpus.

3.1. Training the annotators

The teams involved in developing TED-MDB work with a primary annotator (often the leading researcher of the team) and an experienced secondary annotator or a researcher in discourse. The annotators and researchers are native speakers of the languages we focus on. For annotators already well-trained in discourse, a tutorial on PDTB guidelines explaining major issues such as the position and the span of the arguments, implicit relations and the method of sense assignment is often sufficient. Annotators who are less experienced in discourse-level annotation are trained differently, starting with PDTB annotation guidelines followed by a pre-annotation phase on .doc files. Then, they are asked to create sample annotations on TED talks transcripts proceeding sense by sense, annotating one top-level sense (and its lower levels) at a time. They discuss the annotations with the researcher of the team and proceed to the new top-level sense. The circle is completed when sample annotations are created for all the top-level senses. [5]

3.2. Annotating the Corpus

Unlike most annotated parallel corpora, TED-MDB does not start with annotation projection, where the annotation on one of the languages seeds the annotation on another language (Yarowsky et al., 2001; Ambati and Chen, 2007). Instead, trained or experienced annotators go through each transcribed text sentence by sentence independently and individually. Independent sentence-by-sentence annotation procedure ensures that each team annotates pairs of sentences incrementally and assigns a sense to each relation independent of the others in the team. In the usual annotation projection procedure, there is a risk for a specific language to influence the annotations of other languages. In our procedure, such potential influence is avoided.

Once each team has annotated a text, the annotations are discussed in multilingual group meetings where all TED-MDB members are present. The aim in the discussions is to secure annotation consistency rather than to convince others on a specific type of annotation. The annotations are created using the PDTB annotation tool, which allows selecting discontinuous text spans (Lee et al., 2016) (see Figure 2.

3.3. Sense Annotation

For sense annotation, we use PDTB 3.0 relation hierarchy, which has 4 top-level senses (Expansion, Temporal, Contingency, Contrast) and their second- or in some cases third-level senses, as shown in Figure 1 (Webber et al., 2016).

PDTB 3.0 relation hierarchy is an enriched version capturing a larger number of cases missing in the PDTB 2.0 sense hierarchy. It also simplifies Level-3 senses either by moving them to Level-2 or eliminating them since they were

[5] English, Turkish, Portuguese and Polish annotations are provided by the primary annotator and checked by the researcher of the team. German and Russian annotations are provided by a primary annotator who is also a discourse researcher.

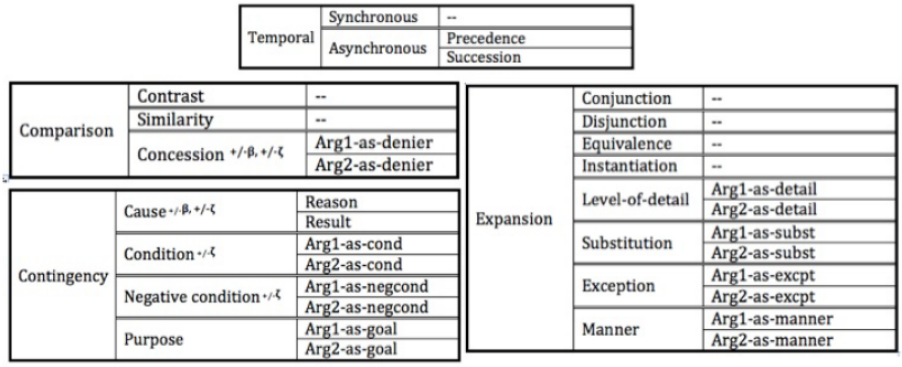


Figure 1: PDTB 3.0 Relation hierarchy. The superscripts indicate the implicit belief (-/+β) or speech-act (-/+ζ) features.

rare and difficult to find. With these improvements and the additional senses, PDTB 3.0 relation hierarchy has been an attractive option for the needs of a multilingual corpus; thus we chose to use it.

Following the PDTB guidelines, we also assign multiple senses for a DC (explicit or implicit). Example 8 provides an explicit DC token with multiple senses. An explicit DC may even trigger the inference of an implicit DC as in example 9 (Rohde et al., 2016). Such cases are annotated as two tokens, the first with the explicit connective (*and*), the second with an implicit adverbial connective (*consequently*). Each corresponding sense is also annotated.

8. .. when **they .. decide whether to invest**, *they look at financial data, metrics like sales growth, cash flow, market share, valuation ...* [Temporal:Synchronous], [Contingency:Condition:Arg2-as-cond]

9. .. *they are really complex* and (consequently) **they can seem really far off**.. [Expansion:Conjunction], [Contingency:Cause:result]

3.4. Intra-Annotator Agreement

To evaluate annotation stability, we measured intra-annotator agreement for three languages for now – English, Portuguese, Turkish. Each primary annotator recreated annotations for 20-23% of the total number of annotated tokens in the corpus. To avoid recall bias, we gave 8-10 months between the first annotations and the reannotations. We measured intra-annotator agreement both on the discourse relation type, i.e. whether the annotator was consistent in annotating the type of a specific relation, and on the top-level sense, i.e. whether the annotator was consistent in annotating the sense of a specific relation at the top level. We regard the original annotations as gold-standard and calculate precision, recall and f-score following the equations given below. In equation (1), the denominator is the sum of reannotations. In equation (2), the denominator is the sum of gold standard annotations. In both equations, the numerator is the number of reannotated tokens.

$$Precision = \frac{\#\ of\ correct\ found\ annotations}{Total\ \#\ of\ found\ annotations} \quad (1)$$

$$Recall = \frac{\#\ of\ correct\ found\ annotations}{\#of\ correct\ expected\ annotations} \quad (2)$$

The intra-annotator agreement results are presented in Tables 2 and 3.

Language	Precision	Recall	F-Score
English	91.92%	91.92%	0.92%
Portuguese	75.97%	75.97%	0.76%
Turkish	71.8%	70.06%	0.71%

Table 2: Intra-annotator agreement results for discourse relation type in three languages

Language	Precision	Recall	F-Score
English	91.73%	93.27%	0.92%
Portuguese	73.28%	75.88%	0.74%
Turkish	72.6%	70.8%	0.71%

Table 3: Intra-annotator agreement results for top-level senses in three languages

We obtained high agreement results for English both for the discourse relation type and the top-level senses (≥ 0.9). For Turkish and Portuguese, the coders achieved ≥ 0.7 in each case, which is acceptable for coherence phenomena (Spooren and Degand, 2010). This score is particularly satisfactory for our task, which presents the added difficulty of resolving ambiguities that arise due to the nature of translation. These results suggest that our annotation guidelines can be used consistently by the annotators.

4. Extending the PDTB Scheme

In the TED talks transcripts we looked at, we find that certain features of spoken discourse are maintained quite faithfully. For example, we observe that the connectives sometimes fulfill roles other than linking two text spans semantically. In particular, the connectives *but, so* exhibit a range of functions that could be defined as sequential linkage (speech management, topic structure, etc.) (Fischer, 2006; Redeker, 2000; Crible and Zufferey, 2015). We will

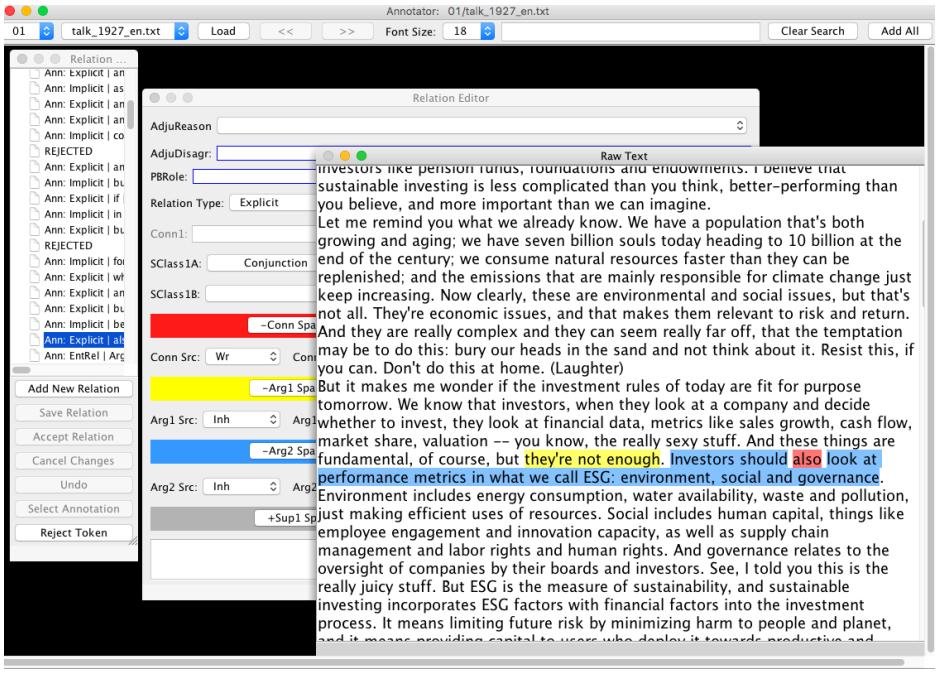


Figure 2: An explicit relation in the annotation environment

refer to these as discourse markers (Schiffrin, 1988). In addition, discourse particles (English *well*, Portuguese *bem*) that underlie the interactive nature of TED talks are also retained in the transcripts. Thus, TED talks transcripts represent a modality where certain features of spoken discourse is shared with written discourse. Our aim is to take this into consideration and annotate the properties of spoken discourse when they occur in the transcripts.

For example, we find that the transcripts involve hypophora, where the speaker asks a question and immediately provides a response himself. We have decided to add Hypophora to the annotation scheme as a feature impelled by the interactive nature of TED talks. We annotate Hypophora as a case of AltLex, where the question word is taken as the anchor, as in example 10 and its translated versions in Portuguese (11), Turkish (12) and Polish (13).

10. Why **is that hard**? Well *to see, let's imagine we take the Hubble Space Telescope and we turn it around ... We'll see something like that, a slightly blurry picture of the Earth. ..* [AltLex: Hypophora] [En]

11. Porque **é tão difícil**? Bem, *imaginemos que pegamos no Telescópio Espacial Hubble e o rodamos e o deslocamos.. Veríamos algo assim, Uma imagem algo difusa de a Terra.* [AltLex : Hypophora] [Por]

12. Neden **bu kadar zor**? *Bunu anlamak için, Hubble Uzay Teleskobu'nu tutup döndürdüğümüzü.. varsayalım. .. Görebildiğimiz tek şey, şuradaki .. yıldızın parıldayan büyük görüntüsü ..* [AltLex : Hypophora] [Tu]

13. Czemu **tego nie zrobiliśmy i czemu to takie trudne**? *Wyobraźmy sobie, że bierzemy Kosmiczny Teleskop Hubble'a, .. Zobaczymy coś takiego, nieco rozmazany obraz Ziemi* ..[AltLex: Hypophora] [Pol]

Hypophora cases may be annotated with a second sense, as shown in the German and the Russian equivalents of 10:

14. Warum **ist das schwer**? *Stellen Sie sich vor, wir nähmen das Hubble Weltraumteleskop , drehten es um .. Wir würden vermutlich ein leicht unscharfes Bild der Erde sehen..* [AltLex: Hypophora], [Contingency:Cause:Result] [Ger]

15. Почему это трудно? Чтоб понять, нужно представить, что мы берём космический телескоп Хаббл и поворачиваем его и перемещаем на орбиту Марса. Мы увидим что-то такое , слегка размытое фото Земли, ... [AltLex: Hypophora], [Contingency:Cause:Result] [Ru]

In addition to this, we spot the discourse marker use of connectives that otherwise signal a DR, such as *but, so*. In such cases, we annotate the arguments in the usual way and assign the label NoRel to the relation (see *so* in example 7 above). Thus, we tease apart discourse connectives from their discourse marker use in the speeches. In case of discourse particles such as *well*, we follow the same procedure. An example annotation from Portuguese *bem* is provided in 16.

16. *Seria o mesmo se erguesse o meu polegar e bloqueasse o ponto luminoso à frente dos meus olhos, poderia vê-los na última fila*. Bem, **o que está a acontecer**? [NoRel] [Por]
    It would be the same thing if I put my thumb up and blocked that spotlight that's getting right in my eye, I can see you in the back row. Well, what's going on?

Finally, PDTB annotates attribution, i.e. whether the information in the arguments of discourse relations is categorized as fact or opinion. Here, PDTB's aim is to capture "the source and degree of factuality of abstract objects"

(Prasad et al., 2006). Given our scarce resources, we leave the annotation of attribution to a further stage.

Some preliminary corpus statistics regarding the current stage of TED-MDB are provided in Tables 4 and 5 below. Table 4 presents the total number of words as well as the total number of annotated tokens for 5 DR types per language. Table 5 presents the total number of annotations on 5 top-level senses per language.

5. Conclusion and Outlook

We introduced TED-MDB, a new multilingual corpus annotated at the discourse level following the approach and annotation principles of PDTB. We described the steps in corpus development and explained how we adapt the PDTB annotation categories. We introduced a new category, Hypophora, an aspect of spoken discourse kept in the transcripts. We also described how we teased apart the discourse connectives from their discourse marker use. A more precise characterization of such features of spoken discourse that exist in TED talks transcripts is further work.

We believe that in annotation tasks, annotation pace can be compromised for annotation quality; hence, we proceed carefully in a step-wise manner at all stages of our effort. Our initial intra-annotator results are promising. In an upcoming paper, we aim to report new statistics and intra-annotator agreement results for all the languages involved (Zeyrek et al., in preparation).

We believe that TED-MDB opens up various interesting research possibilities. Studying the specific strategies that each language (or language pair) encode discourse relations will be one line of research that would lead to understanding discourse structure and semantics across languages. Given that TED-MDB involves translated texts, this line of investigation would contribute to understanding native speakers' translation preferences in structuring the discourse of TED talks. Finally, a concrete output of TED-MDB would be its contribution to the discourse connective lexicon of individual languages and their translation to other languages, which can be used in various monolingual or multilingual natural language processing tasks.

6. Acknowledgements

We thank the members of TED-MDB: Maciej Ogrodniczuk, Yulia Grishina, Sam Gibbon. Thanks to Nuno Martins (for annotating the Portuguese transcripts), Celina Heliasz, Joanna Bilińska, Daniel Ziembicki (for annotating the Polish transcripts) and to Bonnie Webber, Manfred Stede, Alan Lee, Julia Lavid for helpful comments. We acknowledge the support of ISCH COST Action IS1312 Textlink, of TÜBİTAK, The Scientific Research and Technology Council of Turkey (project ID: 2219/1059B191500896), and of FCT, Fundação para a Ciência e a Tecnologia (project ID: PEst-OE/LIN/UI0214/2013). We are grateful to three anonymous reviewers, although any errors are ours.

7. Bibliographical References

Al-Saif, A. and Markert, K. (2010). The Leeds Arabic Discourse Treebank: Annotating discourse connectives for Arabic. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation - LREC 2010*.

Ambati, V. and Chen, W. (2007). Cross lingual syntax projection for resource-poor languages. CMU.

Asher, N. and Lascarides, A. (2003). *Logics of conversation*. Cambridge University Press.

Asher, N. (2012). *Reference to abstract objects in discourse*, volume 50. Springer Science & Business Media.

Bentivogli, L. and Pianta, E. (2005). Exploiting parallel texts in the creation of multilingual semantically annotated resources: the multisemcor corpus. *Natural Language Engineering*, 11(3):247–261.

Cettolo, M., Girardi, C., and Federico, M. (2012). Wit3: Web inventory of transcribed and translated talks. In *Proc of the 16th Conference of the European Association for Machine Translation (EAMT)*, volume 261, page 268.

Crible, L. and Zufferey, S. (2015). Using a unified taxonomy to annotate discourse markers in speech and writing. In *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (isa-11)*.

Fischer, K. (2006). Towards an understanding of the spectrum of approaches to discourse particles: introduction to the volume. *Approaches to discourse particles*, pages 1–20.

Greenbaum, S. (1996). *Comparing English world wide: The International Corpus of English*. Oxford University Press.

Haug, D. T., Jøhndal, M. L., Eckhoff, H. M., Welo, E., Hertzenberg, M. J., and Müth, A. (2009). Computational and linguistic issues in designing a syntactically annotated parallel corpus of Indo-European languages. *TAL*, 50(2):17–45.

Hobbs, Jerry, R. (1985). On the coherence and structure of discourse. *Technical report, CSLI* .

Hovy, E. and Lavid, J. (2010). Towards a 'science'of corpus annotation: a new methodological challenge for corpus linguistics. *International journal of translation*, 22(1):13–36.

Ide, N. and Pustejovsky, J. (2017). *Handbook of Linguistic Annotation*. Springer.

Kolachina, S., Prasad, R., Sharma, D. M., and Joshi, A. K. (2012). Evaluation of discourse relation annotation in the Hindi Discourse Relation Bank. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation - LREC 2012*, pages 823–828.

Lee, A., Prasad, R., Webber, B. L., and Joshi, A. K. (2016). Annotating discourse relations with the PDTB Annotator. In *COLING (Demos)*, pages 121–125.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Language	Word count	Explicit	Implicit	AltLex	EntRel	NoRel	Total
English	7012	290	198	46	78	49	661
Russsian	5623	237	221	20	57	30	565
Polish	6520	218	195	11	104	52	580
Portuguese	7166	269	256	29	38	33	625
German	6366	240	214	17	59	30	560
Turkish	5164	276	202	59	70	51	658

Table 4: Number of words per language and distribution of discourse relation types across the corpus

Language	Temporal	Comparison	Expansion	Contingency	Hypophora	Total
English	46	71	281	132	11	541
Russsian	30	56	270	114	12	482
Polish	44	82	183	108	8	425
Portuguese	54	71	288	143	14	570
German	31	56	259	120	9	475
Turkish	41	74	307	146	14	582

Table 5: Distribution of top-level senses across the corpus

PDTB Group. (2008). The Penn Discourse Treebank 2.0 annotation manual. Technical report, Institute for Research in Cognitive Science, University of Philadelphia.

Pitler, E., Raghupathy, M., Mehta, H., Nenkova, A., Lee, A., and Joshi, A. K. (2008). Easily identifiable discourse relations. *Technical Reports (CIS)*, page 884.

Popescu-Belis, A., Meyer, T., Liyanapathirana, J., Cartoni, B., and Zufferey, S. (2012). Discourse-level annotation over Europarl for machine translation: Connectives and pronouns. In *Proc of the Eighth International Conference on Language Resources and Evaluation (LREC)*.

Prasad, R., Dinesh, N., Lee, A., Joshi, A. K., and Webber, B. L. (2006). Attribution and its annotation in the Penn Discourse Treebank. *TAL*, 47(2):43–64.

Prasad, R., Joshi, A., and Webber, B. (2010). Realization of discourse relations by other means: Alternative lexicalizations. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1023–1031. Association for Computational Linguistics.

Prasad, R., Webber, B., and Joshi, A. (2014). Reflections on the Penn Discourse Treebank, comparable corpora, and complementary annotation. *Computational Linguistics*.

Redeker, G. (2000). Coherence and structure in text and discourse. *Abduction, Belief and Context in Dialogue*, pages 233–263.

Rohde, H., Dickinson, A., Schneider, N., Clark, C. N., Louis, A., and Webber, B. (2016). Filling in the blanks in understanding discourse adverbials: Consistency, conflict, and context-dependence in a crowdsourced elicitation task. *LAW X*, page 49.

Samy, D. and González-Ledesma, A. (2008). Pragmatic annotation of discourse markers in a multilingual parallel corpus (Arabic-Spanish-English). In *Proceedings of the sixth International Conference on Language Resources and Evaluation - LREC 2008*.

Schiffrin, D. (1988). *Discourse markers*. Number 5. Cambridge University Press.

Spooren, W. and Degand, L. (2010). Coding coherence relations: Reliability and validity. *Corpus Linguistics and Linguistic Theory*, 6(2):241–266.

Stede, M., Afantenos, S. D., Peldszus, A., Asher, N., and Perret, J. (2016). Parallel discourse annotations on a corpus of short texts. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation - LREC 2016*.

Tiedemann, J. and Nygaard, L. (2004). The OPUS Corpus-parallel and free: http://logos. uio. no/opus. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation - LREC 2004*.

Tonelli, S., Riccardi, G., Prasad, R., and Joshi, A. K. (2010). Annotation of discourse relations for conversational spoken dialogs. In *Proc. of the 8th International Conference on Language Resources and Evaluation-LREC 2010*.

Webber, B., Prasad, R., Lee, A., and Joshi, A. (2016). A discourse-annotated corpus of conjoined VPs. In *Proceedings of the LAW X*, pages 22–31.

Yarowsky, D., Ngai, G., and Wicentowski, R. (2001). Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proc. of the 1st international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.

Zeyrek, D., Demirşahin, I., Sevdik-Çallı, A., and Çakıcı, R. (2013). Turkish Discourse Bank: Porting a discourse annotation style to a morphologically rich language. *Dialogue and Discourse*, 4(2):174–184.

Zeyrek, D., Mendes, A., Grishina, Y., Gibbon, S., Kurfalı, M., and Ogrodniczuk, M. (in preparation). TED Multilingual Discourse Bank: A parallel corpus annotated in the PDTB style.

Zhou, Y. and Xue, N. (2015). The Chinese Discourse Treebank: a Chinese corpus annotated with discourse relations. *Language Resources and Evaluation*, 49(2):397–431.

Building a Macro Chinese Discourse Treebank

Xiaomin Chu, Feng Jiang, Sheng Xu, Qiaoming Zhu
Natural Language Processing Lab
Soochow University
{xmchu, fjiang, sxu}@stu.suda.edu.cn, qmzhu@suda.edu.cn

Abstract

Discourse structure analysis is an important research topic in natural language processing. Discourse structure analysis not only helps to understand the discourse structure and semantics, but also provides strong support for deep applications of natural language processing, such as automatic summarization, statistical machine translation, question and answering, etc. At present, the analyses of discourse structure are mainly concentrated on the micro level, while the analyses on macro level are few. Therefore, this paper focuses on the construction of representation schema and corpus resources on the macro level of discourse structure. This paper puts forward a macro discourse structure framework and constructs the logical semantic structure and functional pragmatic structure respectively. On this basis, a macro Chinese discourse structure treebank is annotated, consisting of 147 Newswire articles. Preliminary experimental results show that the representation schema and corpus resource constructed in this paper can lay the foundation for further analysis of macro discourse structure.

Keywords: discourse structure, macro discourse structure corpus, logical semantic structure, functional pragmatic structure

1. Introduction

A discourse is not formed by independent and isolated discourse units, but by related and structure units. The task of discourse analysis is to segment sentences into elementary discourse units(EDUs) and recognize the relations among them to form a complete discourse structure. Due to the semantic integrity of discourse units, discourse relations and their well-formed structure, discourse informations have been applied to many natural language processing applications, such as information retrieval (Zou et al., 2014), automatic summarization (Ferreira et al., 2014; Cohan and Goharian, 2017), question and answering (Sadek and Meziane, 2016) and statistical machine translation (Guzmán et al., 2014). Previous research works have proven that discourse informations are beneficial to these NLP applications.

The advent of large-scale collections of annotated data shifted the research community of natural language processing. These corpora have accelerated the development efforts and energized the research community.

Generally speaking, there exist two hierarchical levels of discourse structures: micro level and macro level. At present, the analyses of discourse structure are mainly concentrated on the micro level, that is, the relations and structures between sentences or sentence groups. But the analyses on macro level are relatively few, that is, the relations and structures between paragraphs or documents.

Through the above analysis, it is obvious to realize that developing a macro discourse structure corpus is helpful to understand the overall discourse information and quite necessary for macro natural language processing tasks.

2. Macro Discourse Structure Framework

The overall discourse structure is relevant to the discourse genre and discourse pattern. Thus discourse structures vary if the genres are different. For example, news articles are commonly described in "summary-story" structure, and academic papers are consist of "abstract, introduction, related work, experimentation, conclusion", while court documents are recorded in the structure of "in what way, for what reason, where, according to what inference".

We focus on the news genre in this paper, and will expand the research scope to other discourse genres in future studies. We expand the discourse analysis from intra-paragraph to the overall discourse on the basis of original discourse structure analysis.

Inspired by Rhetorical Structure Theory (Mann and Thompson, 1987) and Macrostructure Theory (Van Dijk, 1980), we explore a macro discourse structure representation schema. Furthermore, we construct the logical semantic structure and functional pragmatic structure on the macro level of discourse analysis respectively. For each structure we define the structural elements such as leaf nodes, non-leaf nodes and edges pointing.

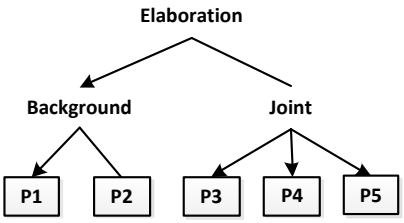


Figure 1: Logical semantic structure of chtb_0019.

Take the chtb_0019 for example, which is a typical news wire article from CTB 8.0 (Xue et al., 2002). There are five paragraphs in the news "Significant achievements in the construction of Ningbo Bonded Area", and the discourse logical semantic structure of this article is shown as Figure 1. Limited to the length of this paper, the full discourse text of this example is not included, please refer to the corpus CTB 8.0. The main contents of the five paragraphs respectively are: P1) Ningbo Bonded Area achieved fruitful results after three years of construction; P2) the basic situation of the Ningbo Bonded Area; P3) the situation of import

and export trade, warehouses, storage area, etc. P4) the situation of industrial processing projects and enterprises; P5) the situation of administrative services and information construction.

This news report is made up of 5 paragraphs(P1, P2, P3, P4 and P5). The paragraphs and paragraphs are connected by discourse relations. In the structure tree shown in Figure 1, leaf nodes represent paragraphs, and non-leaf nodes represent discourse relations. The edges connect the discourse units, while the arrows pointing to the primary discourse units. In this example, paragraph P1 points out the theme of the overall article. According to the direction of arrows in the discourse structure tree, we can quickly locate the most important part(P1) in this article.

From the discourse structure tree of this example, we can see that the analysis of discourse structure contributes to the understanding of the content and the theme of the discourse. Based on the macro discourse structure analysis, we can further enhance the performance of natural language processing applications, such as, information extraction base on the discourse structure, question answering system base on the discourse relations, and automatic summarization base on the primary-secondary relations, etc.

The detailed definitions of macro discourse structure are described as follows.

2.1. Leaf Nodes

Unlike the definition on micro level (the elementary units are treated as leaf nodes), we directly treat the paragraphs which are naturally segmented in the discourses as leaf nodes on the macro level. The natural segmentation of paragraphs are paragraphs segmented by the author's intention and the logical meaning of his writing. For example, there are five paragraphs in the news "Significant achievements in the construction of Ningbo Bonded Area" (chtb_0019), so we treat the five paragraphs(P1,P2,P3,P4,P5) as leaf nodes directly, and the discourse structure of this article is shown as Figure 1.

2.2. Non-leaf Nodes

Discourse relations connect discourse units, which are treated as non-leaf nodes in our macro discourse structure. In the representation scheme, we classify the discourse relations into three categories and fifteen subcategories, listed in Table 1.

Categories	Subcategories
Coordination	Joint, Sequence, Progression, Contrast, Supplement
Causality	Cause-Result, Result-Cause, Background, Behavior-Purpose, Purpose-Behavior
Elaboration	Elaboration, Summary, Evaluation, Statement-Illustration, Illustration-Statement

Table 1: Discourse Relations in our framework

As shown in Figure 1, the relations *Elaboration*, *Background*, and *Joint* are non-leaf nodes in the logical semantic structure tree of the chtb_0019. Specifically, P2 elaborates the background of "Ningbo Bonded Area", which forms a *Background* relation together with P1. P3, P4 and P5 elaborate the "fruitful results" achieved by Ningbo Bonded Area from three aspects respectively, and the three paragraphs form a *Joint* relation. The whole unit constituted by P3, P4 and P5 elaborate the whole unit constituted by P1 and P2, which form an *Elaboration* relation.

2.3. Edges pointing

A discourse relation generally includes two or more discourse units. These discourse units belong to the same relation layer. If one of the discourse units can generalize the intention and content of the relation layer it belongs to, and can connect to other layers on behalf of the relation layer, this discourse unit is a primary unit, while others are secondary ones. There are also some discourse relations, which have no primary and secondary differences between the discourse units they connected, so the discourse units are equally important. We define three types of primary-secondary relations: 1) primary-secondary(PS), the former unit is primary, and the latter unit is secondary; 2) secondary-primary(SP), the former unit is secondary, and the latter unit is primary; 3) equal importance(EI), the discourse units are equally important.

In the macro logical semantic discourse structure, we use the edges pointing to represent the primary-secondary relations. As shown in Figure 1, the arrows point to the primary units. P2 introduces the approval and development situation of "Ningbo Bonded Area", which is the background of the event "Ningbo Bonded Area achieved fruitful results" mentioned in P1. Obviously, P1 expresses more important semantic information, and therefore, this *Background* relation is a PS relation. In the *Joint* relation formed by P3, P4 and P5, the three paragraphs are equally important, so this *Joint* relation is an EI relation. In the *Elaboration* relation formed by P1-P2 and P3-P5, the elaborated unit P1-P2 is more import than the elaborate unit P3-P5, so this *Elaboration* relation is a PS relation. In the overall discourse, P1 can best express the discourse topic "Significant achievements in the construction of Ningbo Bonded Area" (also the discourse title), P1 is therefore the most important paragraph among all these discourse units. Based on the edges pointing of the logical semantic structure tree, readers can also get the same conclusion.

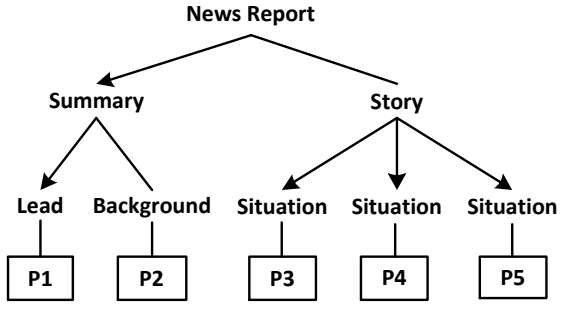


Figure 2: Functional Pragmatic Structure of chtb_0019.

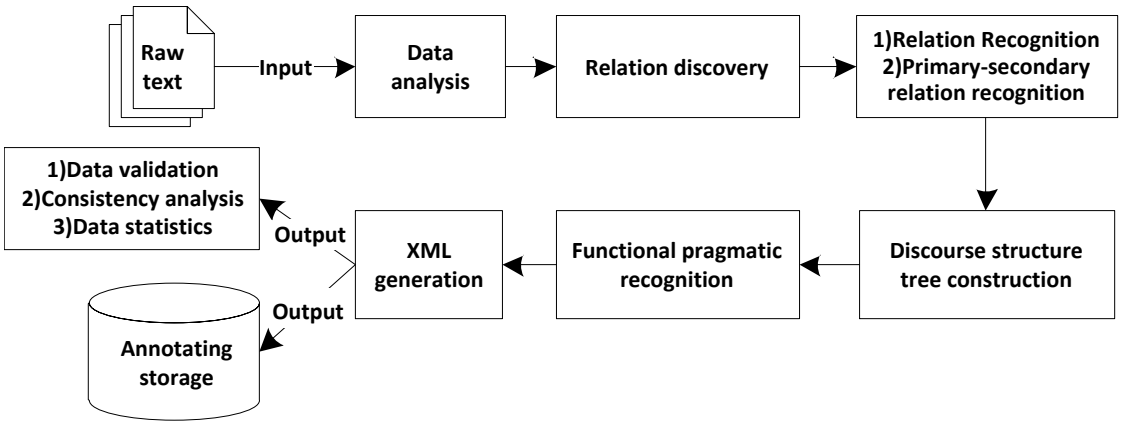


Figure 3: The processing flow of corpus annotating.

2.4. Macro Discourse Structure

In the representation schema we define, the discourse is organized as a tree structure, in which paragraphs appear in the leaf nodes and the discourse relations appear in the non-leaf nodes. The tree structure is an appropriate representation of discourse structure, which expresses the hierarchical relationship of the discourse. Essentially, the depth of the hierarchical structure indicates the depth of the corresponding discourse semantic.

2.5. Functional Pragmatic Structure

In addition to the logical semantic structure, we also define the functional pragmatic structure. Specifically, on the basis of the logical semantic structure, we add function pragmatic attribute to each node. This paper defines 18 functional types, including *News Report*, *Lead*, *Sub-Summary*, *Summary*, *Situation*, *Story*, *Cause*, *Sum-up*, *Result*, *Behavior*, *Purpose*, *Statement*, *Illustration*, *Background*, *Comment*, *Supplement*, *Contrast*, and *Progression*.

In the functional pragmatic structure of chtb_0019(as shown in Figure 2), the root node is *News Report*. P1 is the *Lead* of the article. P2 elaborates the background of P1, so its function is *Background*. P1 and P2 form the node of *Summary*. P3, P4 and P5 elaborate the *Summary* in detail, and their corresponding functions are *Situation*. The parent node of P3-P5 is defined as *Story*. Each node has its corresponding function in the article, and all the functional nodes constitute a complete article.

3. Annotation Task

Guided by the macro discourse structure framework defined above, we annotate a Macro Chinese Discourse Treebank (MCDTB) consisting of 147 Xinhua newswire articles on the top of Chinese Discourse Treebank (CDTB) (Li et al., 2014).

Because the discourse units are not isolated from the overall discourse, it's difficult to judge whether the discourse units are important or not and what relations are between the discourse units simply from the units themselves. It is necessary to have a comprehensive understanding of the overall article before the annotation work.

We have three annotators. We independently annotated articles based on an initial set of annotating guidelines, and then held discusses to compare results. At the meeting, we discussed the issue of segments, relations and primary-secondary relations, and analyzed the causes of differences. These exploratory sessions led to enhancements in the annotating guidelines and annotation quality. On the basis of the annotated logical semantic structure, we annotate the function of each node and complete the tagging of functional pragmatics. According to the contrast relation between semantic and pragmatic annotations, we summarize the transform rules from semantic to pragmatic, and construct a rule-based pragmatic transformation model by which can automatically tag functional pragmatics directly.

3.1. Annotation Strategy

We employ a combination of top-down and bottom-up strategy in the annotation work. **1) Top-down**: We determine the overall level first and then analysis goes on step by step to the individual discourse units. Such a top-down strategy can easily grasp the overall discourse structure, which consistent with the reading habit of human beings. **2) Bottom-up**: Meanwhile, we determine whether the lower discourse units need to be combined first according to the similarity of their forms and contents, and combine them together as a whole unit to contact with other parts. The annotation work shows our annotation strategy is effective.

3.2. Annotation process

Given a a piece of raw materials, we read and analyze the text first. A complete tree structure of the discourse is constructed, after the steps of relation discovery, discourse relation recognition, primary-secondary relation recognition, discourse structure tree construction. Then the annotation of functional pragmatics is automatically completed by the rule-based logical semantics to the functional pragmatics conversion program. The annotation results are saved in the form of XML. In order to ensure the consistency of the annotation, we verify the annotation results and calculate the consistency. After all the annotation work is completed, data statistics and analysis of the annotation results are car-

ried out. The specific process is shown in Figure 3)
We develop an annotation platform in order to enable annotators to construct discourse structures visually. Annotators annotate the discourse topic, lead, abstract, paragraph segmentations, paragraph topics, discourse relations, and primary-secondary relations for each discourse with the annotation platform. A complete discourse structure tree can be automatically generated and all annotation informations are saved in a XML file.

4. Quality Assurance

To ensure the quality of our corpus, we adopt the annotator consistency using agreement and kappa. Table 2 illustrates the annotator consistency in detail. We measure the agreement and kappa of discourse spans, primary-secondary relations and discourse relations. It's very difficult to achieve high consistency because the judgments of relation and structure are very subjective.
The method of consistency calculation used in this paper refers to the work of the corpus of RST (Marcu et al., 1999), and the appropriate adjustment is made according to the contents of our annotation.

Categories	Agreement	Kappa
Discourse Spans	88.54%	0.771
P-S Relations	80.67%	0.694
Discourse Relations	83.05%	0.556

Table 2: Annotating consistency (P-S Relations refers to primary-secondary relations in this table.)

5. Corpus Details

Our corpus consists of 147 newswire articles from Chinese Treebank 8.0. There are 648 paragraphs with 670 discourse relations annotated. There are 5.56 paragraphs and 624 Chinese characters in each article on average. Detailed statistical data are shown in Table 3 and Table 4.

Statistics Items	Value
Count of documents	147
Count of paragraphs	817
Amount of macro discourse relations	670
Average paragraphs (paragraphs/document)	5.56
Maximal of paragraphs	13
Minimal of paragraphs	2
Count of sentences	1,802
Average sentences (sentences/paragraph)	2.2
Count of characters	91,709
Average characters (characters /paragraph)	624

Table 3: Corpus basic statistic data

In terms of discourse relations, compared with the categories of *Coordination* and *Elaboration*, the amount of *Causality* relations is less and the data set is unbalanced. In terms of primary-secondary relations, compared with PS and EI relations, the amount of SP relation is very small, and the data set is quite unbalanced.

6. Preliminary Experiment

Based on the corpus we built, we can do the following analysis: discourse span segmentation, discourse relation recognition, primary-secondary relationship recognition, and discourse structure tree construction. In this section, we evaluated our annotated corpus with the task of recognition of primary-secondary relationship.
According to the characteristics of macro level primary-secondary relations and feature information used in the researches before, this paper adds the semantic information and takes the topic similarity as an important feature. The topic similarity refers to the semantic similarity between the discourse unit and the discourse topic. This paper puts forward two calculation methods of the topic similarity based on the word2vec (Mikolov et al., 2013) and LDA (Blei et al., 2003) respectively.
We conducted four sets of experiments to verify the feasibility of our proposed semantic similarity model and efficient. (S1) The first set of experiments used structural features, and the result was used as the benchmark system. (S2) The second groups added the LDA topic similarity as a semantic feature. (S3) The third groups added the word2vec topic similarity as a semantic feature. (S4) The fourth groups added the word2vec and LDA topic similarity both. Experimental results are shown in Table 5.

7. Discussion

Why don't we directly use the discourse relations defined in RST, but redefine new discourse relations? 1) the expressions in Chinese and English are different; 2) the expressions on micro level and macro level are different. For example, a paragraph not only discusses the background of the event, but also describes the circumstance of the event, then it cannot be determined whether it corresponds to *Circumstance* or *Background* in the RST definition.
These differences have been identified in previous studies, so there are many attempts at annotating corpora in both Chinese and English, such as Carlson et al. (2003), Yue (2008), and Li et al. (2014). Carlson et al. (2003) also discussed the annotation problem of macro-level. In the study of computational models, more and more researchers have constructed intra-sentential, multi-sentential, and multi-paragraph models separately to achieve higher performance (Joty et al., 2013; Wang et al., 2017).
The difficulties of annotating work: 1) the annotation processing is very subjectivity because of the different understandings among different annotators, so the consistency is not very high; 2) a lot of discussion is needed to achieve consistent understanding; 3) the structure framework and relation set have been repeatedly confirmed to give structure definition and annotation guidelines more clearly. Because of these reasons, the current annotation scale is not large enough, and our following research work will continue to expand the scale.

8. Conclusion

In this paper, we expand the discourse structure analysis from intra-paragraph to the overall discourse. We propose

Categories	PS	SP	EI	Subtotal	Percentage
Coordination	65	4	243	312	46.57%
Causality	94	8	7	109	16.27%
Elaboration	232	14	3	249	37.16%
Total	391	26	253	670	100.00%
Percentage	58.36%	3.88%	37.76%	100.00%	-

Table 4: Statistics of discourse relations and primary-secondary relations

Feature set	Accuracy	F-score
S1	81.96%	80.4%
S2	82.11%	80.5%
S3	82.26%	80.6%
S4	82.70%	81.1%

Table 5: Experimental results using different feature sets

a macro discourse structure representation scheme, and describe the scheme in detail. We also annotate a Marco Chinese Discourse Treebank consisted of 147 news wire articles based on the representation schema we defined. To evaluate our annotated corpus, we take the preliminary experiment with the task of recognition of primary-secondary relations. In the future work, we will enlarge the scale of the MCDTB corpus and explore the macro discourse structure computational models.

9. Acknowledgements

The work is supported by the National Natural Science Foundation of China (61773276, 61673290, 61331011) and Jiangsu Provincial Science and Technology Plan (BK20151222).

10. Bibliographical References

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Carlson, L., Marcu, D., and Okurowski, M. E. (2003). Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*, pages 85–112. Springer.

Cohan, A. and Goharian, N. (2017). Scientific document summarization via citation contextualization and scientific discourse. *International Journal on Digital Libraries*, pages 1–17.

Ferreira, R., de Souza Cabral, L., Freitas, F., Lins, R. D., de França Silva, G., Simske, S. J., and Favaro, L. (2014). A multi-document summarization system based on statistics and linguistic treatment. *Expert Systems with Applications*, 41(13):5780–5787.

Guzmán, F., Joty, S., Màrquez, L., and Nakov, P. (2014). Using discourse structure improves machine translation evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 687–698.

Joty, S. R., Carenini, G., Ng, R. T., and Mehdad, Y. (2013). Combining intra-and multi-sentential rhetorical parsing for document-level discourse analysis. In *ACL (1)*, pages 486–496.

Li, Y., Feng, W., Sun, J., Kong, F., and Zhou, G. (2014). Building chinese discourse corpus with connective-driven dependency tree structure. In *EMNLP*, pages 2105–2114. Citeseer.

Mann, W. C. and Thompson, S. A. (1987). Rhetorical structure theory: A theory of text organization (no. isi/rs-87-190). marina del rey. *CA: Information Sciences Institute*.

Marcu, D., Amorrortu, E., and Romera, M. (1999). Experiments in constructing a corpus of discourse trees. *Towards Standards and Tools for Discourse Tagging*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Sadek, J. and Meziane, F. (2016). A discourse-based approach for arabic question answering. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(2):11.

Van Dijk, T. A. (1980). *Macrostructures: An interdisciplinary study of global structures in discourse, interaction, and cognition*. Lawrence Erlbaum Associates.

Wang, Y., Li, S., and Wang, H. (2017). A two-stage parsing method for text-level discourse analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 184–188.

Xue, N., Chiou, F.-D., and Palmer, M. (2002). Building a large-scale annotated chinese corpus. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–8. Association for Computational Linguistics.

Yue, M. (2008). Rhetorical structure annotation of chinese news commentaries. *Journal of Chinese Information Processing*, 4(002).

Zou, B., Zhou, G., and Zhu, Q. (2014). Negation focus identification with contextual discourse information. In *ACL (1)*, pages 522–530.

Enhancing the AI2 Diagrams Dataset Using Rhetorical Structure Theory

Tuomo Hiippala and Serafina Orekhova
Department of Languages, Digital Geography Lab
University of Helsinki, Finland
tuomo.hiippala, serafina.orekhova @helsinki.fi

Abstract

This paper describes ongoing work on a multimodal resource based on the Allen Institute AI2 Diagrams (AI2D) dataset, which contains nearly 5000 grade-school level science diagrams that have been annotated for their elements and the semantic relations that hold between them. This emerging resource, named AI2D-RST, aims to provide a drop-in replacement for the annotation of semantic relations between diagram elements, whose description is informed by recent theories of multimodality and text-image relations. As the name of the resource suggests, the revised annotation schema is based on Rhetorical Structure Theory (RST), which has been previously used to describe the multimodal structure of diagrams and entire documents. The paper documents the proposed annotation schema, describes challenges in applying RST to diagrams, and reports on inter-annotator agreement for this task. Finally, the paper discusses the use of AI2D-RST for research on multimodality and artificial intelligence.

Keywords: multimodality, diagrams, rhetorical structure theory, artificial intelligence

1. Introduction

The Allen Institute AI2 Diagrams dataset (hereafter AI2D)[1] contains nearly 5 000 grade-school level science diagrams, which have been annotated for their elements and the semantic relations that hold between them (Kembhavi et al., 2016). The AI2D dataset was initially developed with two emerging computer vision tasks in mind: diagram understanding and visual question answering. Both tasks are challenging, as diagrams are inherently multimodal: they frequently combine various diagrammatic elements, such as arrows and lines with natural language, two-dimensional graphic elements, illustrations and photographs, and draw a multitude of semantic relations between them.

AI2D describes the semantic relations between diagram elements using a set of relations drawn from the framework for describing diagrams proposed in Engelhardt (2002). This paper proposes an alternative scheme for describing the semantic relations that hold between diagram elements, building on Rhetorical Structure Theory, or RST for short (Mann and Thompson, 1988; Taboada and Mann, 2006). To distinguish the original AI2D dataset from the emerging language resource described in this paper, we will adopt the name AI2D-RST to refer to the RST-enhanced dataset in the following discussion.

Although RST was originally developed for describing the organization of entire texts, the framework has also been applied to the generation of diagrammatic representations (André and Rist, 1995; Bateman et al., 2001). More recently, RST has been used to describe discourse structures in various multimodal artefacts ranging from newspapers and journals (Kong, 2013; Taboada and Habel, 2013) and product packaging (Thomas, 2014) to tourist brochures (Hiippala, 2015) and health care posters (Zhang, 2018). When combined with additional layers of description to capture the logical structure of content, its visual appearance and layout, even relatively small RST-annotated corpora have been able to reveal patterns characteristic to the structure of the aforementioned multimodal artefacts.

This suggests that RST can provide descriptions that are sufficiently fine-grained to bring out distinctions in the structure of various multimodal artefacts. As AI2D already contains annotations for the diagram elements and their layout, we assume that the application of RST to the dataset will provide multimodality researchers with a valuable resource for studying the structure of diagrammatic representations. Researchers working in the domain of artificial intelligence, in turn, may evaluate whether diagram understanding and visual question answering algorithms can learn better from RST annotation than the schema originally used for AI2D.

The paper itself is structured as follows: after introducing the AI2D dataset, we evaluate its original annotation from the viewpoint of multimodality research. Next, we proceed to describe the changes introduced in AI2D-RST and the challenges in applying RST to diagrams, and report on an experiment measuring inter-annotator agreement. Finally, we consider how the emerging resource may be used in research on multimodality and artificial intelligence.

2. The Original AI2D Dataset

Explicating the motivation for developing AI2D, Kembhavi et al. (2016, 235) note that research on computer vision has mainly focused on photographic images, while "rich visual illustrations", such as diagrams and information graphics, have received relatively little attention. In diagrams, this richness emerges from the combination of multiple modes of communication: diagrammatic elements, such as arrows and lines, are typically used alongside natural language and various types of images ranging from photographs to illustrations, and combined into meaningful ensembles by taking advantage of the layout space (Bateman et al., 2017, 281). As Kembhavi et al. (2016) observe, this makes diagrams radically different from photographic images, and for this reason, their computational processing involves an entirely different set of problems.

[1] Available at http://allenai.org/plato/diagram-understanding/ (accessed 13 February 2018).

Relationship	Definition	Count
ARROWHEAD ASSIGNMENT	An arrow head associated to an arrow tail	18541
INTRA-OBJECT LABEL	A text box naming the entire object	16281
INTER-OBJECT LINKAGE	Two objects related to one another via an arrow	15802
INTRA-OBJECT LINKAGE	A text box referring to a region within an object via an arrow	15664
INTRA-OBJECT REGION LABEL	A text box referring to a region within an object	2002
IMAGE TITLE	The title of the entire image	1458
IMAGE MISC	Decorative elements in the diagram	1148
IMAGE SECTION TITLE	Text box that serves as a title for a section of the image	807
IMAGE CAPTION	A text box that adds information about the entire image, but does not serve as the image title	690
ARROW DESCRIPTOR	A text box describing a process that an arrow refers to	681

Table 1: Semantic relations defined between diagram elements in the AI2D annotation by Kembhavi et al. (2016, 239). The column on the right-hand side gives their number of occurrences in the AI2D dataset.

To drive forward research on computational processing of diagrams, Kembhavi et al. (2016, 242–243) present a dataset containing 4 907 grade-school level science diagrams, which are described using 150 000 annotations that capture their elements, semantic interrelations and position in the layout. In addition, the dataset contains 15 000 multiple choice questions about the content of the diagrams for experiments involving diagram understanding and visual question answering.

The diagrams, which AI2D provides as PNG images accompanied by their annotations in JSON, were scraped from Google Images using the chapter headings of primary school science textbooks for grades one to six as seed terms. The result is a diverse dataset containing diagrams from various sources, ranging from professionally-produced diagrams in school textbooks to diagrams produced for learning materials by the teachers or by the students themselves. Figure 1 illustrates one diagram in the dataset, which clearly belongs to the first category of professionally-produced diagrams.

The diagrams were annotated by workers on Amazon Mechanical Turk[2], a crowdsourcing platform frequently used to create datasets for AI research (Kovashka et al., 2016). Due to their complex structure, the diagram annotation process was broken down into separate stages to ensure agreement between annotators. These stages involved, for instance, identifying diagram elements, categorising them, labelling their interrelations and answering multiple choice questions about the content of the diagram (Kembhavi et al., 2016, 243).

Building on the annotated corpus, Kembhavi et al. (2016, 239) propose to represent diagrams using graphs, which they refer to as *Diagram Parse Graphs* (hereafter DPG). DPG uses nodes to represent diagram elements, such as blobs (illustrations), text boxes, arrows and arrowheads, while the edges between nodes represent relationships between the elements. These relationships, which are listed in Table 1, are drawn from the framework developed by Engelhardt (2002) for analysing the syntax and semantics of maps, charts and diagrams.

That being said, Kembhavi et al. (2016) use the AI2D dataset to train deep neural networks for two distinct tasks, which also reflect the original distinction defined by Engelhardt (2002): *syntactic parsing* and *semantic interpretation*. Whereas syntactic parsing refers to the task of learning to infer a DPG that captures the diagram structure, semantic interpretation is concerned with interpreting a DPG to answer questions about diagram content. The goal of AI2D is to enable and support the development and evaluation of algorithms for both tasks.

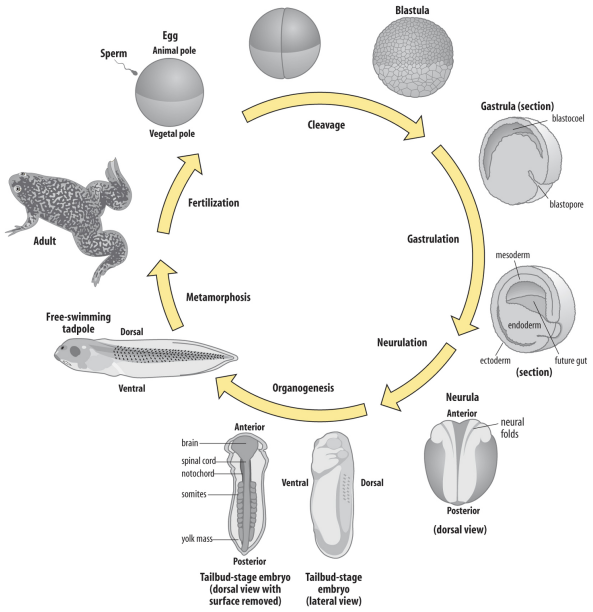


Figure 1: Diagram number 2240 in the AI2D dataset

3. Evaluating AI2D from the Perspective of Multimodality Research

The work of Kembhavi et al. (2016) assumes that resolving the relations that hold between the elements of a diagram is crucial for their computational understanding. In research on multimodality, these relations are often discussed under the heading of text-image relations, or more broadly, relations that hold between contributions from different modes of expression. Bateman (2014a) provides an extensive re-

[2] https://www.mturk.com/

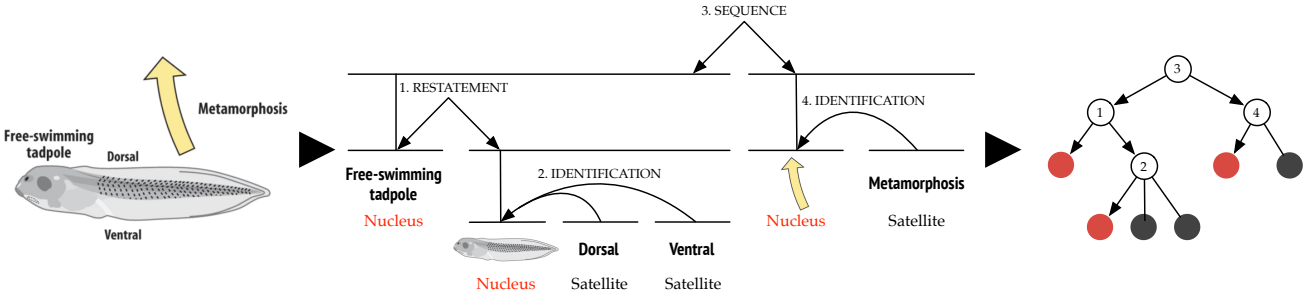


Figure 2: A partial RST analysis attempting to capture the "phase-and-process" structure of the diagram in Figure 1, represented using a schema for visualising RST relations. The straight lines indicate *multinuclear* relations between nuclei (such as RESTATEMENT and SEQUENCE), whereas the curved lines indicate *mononuclear* relations between nuclei and satellites (such as IDENTIFICATION). These relations are grouped together under *spans*, which are indicated by the horizontal lines. The information carried by spans, which reflects how diagram elements are chunked together, is carried over to the graph in the form of nodes that join related nuclei and satellites together. These nodes are coloured white, whereas nuclei are red and satellites are black.

view of research in this area, which may also be used to evaluate the semantic relations defined within AI2D from a multimodal perspective.

To begin with, several of the relations defined in AI2D appear to share roughly the same semantic function: Kembhavi et al. (2016, 245) acknowledge this explicitly by stating that relations 1–6 in Table 1 serve the purpose of relating one or more elements to one another. There seems to be, however, significant overlap among these relations, particularly between INTRA-OBJECT LABEL, INTRA-OBJECT REGION LABEL, INTRA-OBJECT LINKAGE and ARROW DESCRIPTOR, as they all appear to serve the general purpose of *identifying* some object.

It should be noted that such a semantic relation – identification – may be realised in different ways, either using an explicit diagrammatic element such as an arrow or a line, or spatially using layout, by placing the identifier and the identified close to each other. Both alternatives may be found on both inner and outer circle of the diagram shown in Figure 1 (cf. e.g. *Blastula* and *Gastrula (section)*). From a multimodal perspective, the question is whether these relations could be reduced into fewer, more generic categories, which would also unify the description.

Furthermore, the relation of ARROWHEAD ASSIGNMENT does not indicate a semantic, but rather a logical relation between diagram elements, indicating which elements belong together. Although defining such relations explicitly may be necessary for the current algorithms developed for diagram understanding, corpus-driven frameworks developed for describing complex multimodal artefacts such as the one presented in Bateman (2008) and its extension in Hiippala (2015) argue against conflating different descriptions of multimodal structure. This demarcation is intended to enable pulling apart different types of structure, in order to provide a clearer view of their distinct contribution to the meaning of the diagram.

In the case of AI2D, having multiple layers of annotation at hand – one for the logical structure, and one for the semantic relations – would also remove the need for relations such as IMAGE MISC, which is used to indicate decorative elements in the diagrams (see the last item in Table 1). If these descriptions were represented using multiple graphs, this would simply mean that some nodes present in the graph representing logical structure would be absent from the graph representing semantic relations. Because AI2D already contains rich annotations covering aspects of both logical and semantic structure, the existing annotation may be used as a basis for generating separate descriptions as a part of AI2D-RST.

4. Towards AI2D-RST

To clearly separate and describe different kinds of multimodal structures typically found in diagrams, AI2D-RST leverages the rich annotation provided by the original AI2D dataset to produce two distinct types of Diagram Parse Graphs (DPG). These DPGs, whose descriptions are also connected using cross-referenced identifiers, account for two kinds of multimodal structures in diagrams: DPG-L for the logical structure, and DPG-R for the rhetorical structure. The motivation for representing these structures using two different graphs is to make them more focused, that is, avoiding the problems associated with conflated descriptions of multimodal structure discussed in Section 3.

To begin with DPG-L, the purpose of this graph is to indicate which elements are generally presented as belonging together in the logical structure, without making assumptions about any semantic relations that may hold between them. As pointed out above, DPG-L covers all elements identified in the original AI2D annotation, such as those that do not necessarily contribute to the rhetorical structure, such as decorative elements or authorship attributions identified using the IMAGE MISC relation. This representation bears close resemblance to the original DPG defined in AI2D, as the analytical decomposition of diagrams remains the same in AI2D-RST.

DPG-R, in turn, contains only the nodes that participate in the rhetorical structure, whose relations are described using Rhetorical Structure Theory (Taboada and Mann, 2006),

the multimodal extension of RST proposed in Bateman (2008). Essentially, this extension contains the original relations defined within 'classical' RST and a set of 'subnuclear' relations necessary for decomposing fragments of discourse, which classical RST would treat as a single analytical unit (Bateman, 2008, 162). Finally, certain diagram-specific additions to DPG-R are presented shortly below in Section 5.

Another feature that DPG-R introduces to the description is what RST terms *nuclearity*. This assumption states that some parts of a text or a multimodal ensemble act as *nuclei*, which carry the most relevant meanings for the communicative task at hand, whereas optional parts – *satellites* – enhance them by providing additional information. Figure 2 illustrates this by defining RST relations between the elements and representing these relations using a graph. Here the assumption is that algorithms developed for semantic interpretation of diagrams could learn to attend to the elements that carry information relevant to the task at hand, for instance, by searching for sequences among nuclei and additional information among satellites.

The application of RST also requires information on the chunking of elements, that is, how they are grouped together in the diagram. As Taboada and Mann (2006, 427) point out, RST relations are applied recursively to the artefact under analysis in order to capture the intended communicative effect, which may be achieved using a combination of rhetorical relations. This phenomenon is well-known from the analysis of entire multimodal documents, in which the chunks often constrain the process of interpretation by limiting the possible RST relations to elements belonging to the same chunk (Hiippala, 2015, 168). However, diagrams present certain requirements for RST analysis, which are discussed below.

5. Applying RST to AI2D

5.1. Feasibility

To assess the feasibility of using RST to describe semantic relations in AI2D, we sampled the data without replacement for 545 semantic relations and annotated them for both rhetorical relations and their nuclearity using RST. This number amounted to roughly 1% of semantic relations reported in Table 1, excluding the relationship of ARROWHEAD ASSIGNMENT, which we considered to belong to the logical structure, as set out in Section 3.

In order to measure the level of agreement between our annotations, we used the common metrics surveyed in Artstein and Poesio (2008), such as Scott's π and Krippendorff's α, as implemented in Natural Language Toolkit (NLTK) (Bird et al., 2009). Table 2 shows the scores for average observed agreement and the aforementioned metrics for RST relations and nuclearity. In the case of nuclearity, we assigned the role of *nucleus*, *satellite* or *none* (for elements with no RST relation) to the elements acting as the origin or destination of a semantic relation, as defined in the original AI2D annotation.

For annotating the sample, we used a total of 10 out of 35 available RST relations to describe how the elements relate to each other: CONTRAST, EFFECT, ELABORATION, IDENTIFICATION, INTERPRETATION, PREPARATION, PROPERTY-ASCRIPTION, RESTATEMENT, SEQUENCE and TITLE. We also used an additional relation, NONE, to mark cases in which no relation was deemed to hold between the diagram elements, providing a final number of 11 categories. The chance-corrected scores for measuring agreement between our annotation for RST relations, namely Krippendorff's α (0.7245) and Scott's π (0.7242) suggest that we may draw only tentative conclusions about semantic relations in diagrams on the basis of our initial annotation.

Metric	Relation	Origin	Destination
Average agreement	0.7835	0.8128	0.9083
Krippendorff's α	0.7245	0.5490	0.4520
Scott's π	0.7242	0.5486	0.4515

Table 2: Inter-annotator agreement between annotators

The results are nevertheless promising, as it should be noted that there was a considerable difference in our level of experience in applying RST, as the second author received only minimal training before the experiment, which may have affected the α and π scores for relations. The low scores for nuclearity (origin/destination), in turn, reflect the presence of just three categories, which obviously increases the possibility of chance agreement.

In order to provide a measure of agreement that would take into account the differences in our expertise in applying RST, we follow Das et al. (2017) and report precision, recall and F1-scores for expert (first author) vs. novice (second author) annotation in Table 3, as implemented in scikit-learn (Pedregosa et al., 2011).

	Precision	Recall	F1-score
RST relation	0.78	0.78	0.78
Origin role	0.81	0.81	0.81
Destination role	0.93	0.91	0.92

Table 3: Prevalence-weighted macro-average scores for precision, recall and F1 for expert vs. novice annotator

The precision, recall and F1-scores appear promising despite the limited training, and are likely to be improved by revising the annotation manual for AI2D-RST as the work proceeds. This involves extending RST to diagrammatic representations by redefining some of the relations and their definitions, as we will describe below.

5.2. Extending RST

Evaluating inter-annotator agreement, in connection with close analyses of selected examples, helped to identify a core set of RST relations applicable to the description of semantic relations between diagram elements, which are given in Table 4. At this point, however, it is also useful to highlight some challenges in applying RST to diagrams. To begin with, classical RST was designed to capture relations that hold between sequential units of discourse in written language, mainly at the level of clause and beyond. The assumption of sequentiality, however, which is at the

Relationship	Description	Source
EFFECT	A generic mononuclear relation for describing processes that take place between entities, which are often reinforced using lines or arrows (see Figure 3). The affected entity acts as the nucleus, while the origin of the effect acts as the satellite.	AI2D RST
ELABORATION	A more extensive verbal description, such as a phrase or a clause, which provides more specific information about some entity or its part(s).	Classical RST
IDENTIFICATION	A short text segment, such as a single noun or a noun group, which identifies an entity or its part(s). A typical example would be a label for a part of an entity (see Figure 2).	GeM RST
PROPERTY-ASCRIPTION	A mononuclear relation between an entity (nucleus) and something predicated of that entity (satellite).	GeM RST
RESTATEMENT	A multinuclear relation holding between two entities that could act as a substitute for each other, such as the name of an entity and its visualisation (see Figure 2).	Classical RST
SEQUENCE	A multinuclear relation indicating a temporal or spatial sequence holding between two or more entities.	Classical RST
TITLE	A text segment acting as the title for the entire diagram or its parts.	GeM RST

Table 4: Common RST relations encountered in preliminary studies of applying RST to the AI2D dataset. The column on the right gives the work in which the relation was originally defined. Classic RST refers to the foundational work in Mann and Thompson (1988) and Taboada and Mann (2006), whereas GeM RST refers to the multimodal extension presented in Bateman (2008). AI2D RST, in turn, refers to the work presented here.

heart of classical RST, rarely holds for entire multimodal artefacts or their parts (Hiippala, 2015, 50–51).

For the analysis of written texts, this assumption enabled classical RST to control the number of potential relations drawn between different units of discourse, as this prevented drawing relations between non-adjacent units of discourse in written text. In a complete and well-formed description of rhetorical structure, each discourse unit would participate in a single relation, and textual progression would naturally structure the discourse units into a recursive organisation (Mann et al., 1992).

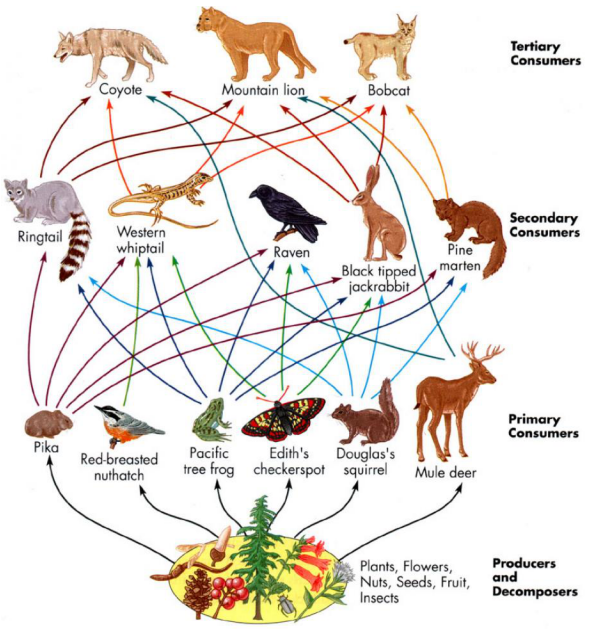


Figure 3: Diagram number 1807 in the AI2D dataset

In multimodal extensions of RST, the sequentiality assumption has been discarded in favour of alternative criteria such as spatial proximity (Bateman, 2008, 158). Applying RST to diagrams is likely to require additional criteria to complement spatiality. *Connectedness* emerges as one criterion for addressing graph-like diagrams, in which diagram elements participate in multiple relations, which do not necessarily respect the criterion of spatiality.

This may be exemplified using Figure 2, which shows how several diagram elements, namely the combinations of an illustration and its caption, relate to multiple elements of the same kind, which are not positioned close to each other in the diagram layout (see e.g. the arrow drawn from *Mule deer* to *Coyote*).

To capture the relations that build on the property of connectedness, we introduce a generic relation termed EFFECT, which describes any relation between two interconnected elements that affect each other in some way (see Table 4). The target of this effect is marked as the nucleus, whereas the origin acts as the satellite. Introducing this relation allows taking on graph-like diagrams, whose elements participate in multiple rhetorical relations. This, however, requires an alternative, graph-like means of visualising RST structures, as these interconnections are difficult to visualise using hierarchical trees. Finally, it should be noted that we do not incorporate the diagram elements responsible for signalling EFFECT, such as arrows and lines, into the description of rhetorical relations, but consider them as a part of the logical structure (see Section 4).

6. AI2D-RST as a Multimodal Resource

AI2D-RST is intended as a resource for researchers working on multimodality and artificial intelligence. The research community focusing on multimodality has long called for larger datasets, which would enable empirical research in the manner of corpus linguistics (Kaltenbacher, 2004; Bateman, 2014b; O'Halloran et al., 2018). AI2D-RST takes the first step towards this long-term goal, enabling the empirical study of diagrammatic representations

and testing hypotheses about their multimodal characteristics against a sufficiently large dataset. If such a resource is found useful, this may also lead to the adoption of crowdsourcing techniques for generating low-level annotations for multimodal corpora in the future.

For the AI community, AI2D-RST offers a dataset containing mixed annotations, sourced from non-experts in the form of original, crowd-sourced annotations describing the diagram elements, and from experts in the form of the RST annotation capturing the relations between these elements. Moreover, the RST-based description in AI2D-RST provides the AI research community with the first annotation schema informed by theories of multimodality. This novel resource may be used to evaluate whether expert annotations improve the performance of algorithms for the tasks defined by Kembhavi et al. (2016).

To support further research on diagrams and potential applications of the dataset, we also provide tools for visualising the annotation in both AI2D and AI2D-RST, and their respective Diagram Parse Graphs, in addition to the annotation tool used for creating the AI2D-RST corpus. These tools, written in Python 3.6, and the AI2D-RST corpus will be available online.[3]

7. Conclusion

This paper has presented an emerging multimodal language resource based on the AI2 Diagrams dataset (Kembhavi et al., 2016), which contains nearly 5000 grade-school level diagrams for developing algorithms that can process the structure and contents of diagrams. The emerging resource described in this paper, named AI2D-RST, enriches the annotation contained in the original AI2D dataset by using Rhetorical Structure Theory (RST) to describe the relations holding between elements participating in the diagram. The motivation for developing AI2D-RST is to provide an alterative annotation informed by recent research on multimodality, which has shown RST to be a powerful analytical tool for describing semantic relations, particularly when combined with descriptions of multimodal structure, which are already in place in AI2D. In the long run, AI2D-RST is expected to contribute to research on multimodality and artificial intelligence, improving the understanding of diagrams for both humans and computers.

8. Bibliographical References

André, E. and Rist, T. (1995). Generating coherent presentations employing textual and visual material. *Artificial Intelligence Review*, 9:147–165.

Artstein, R. and Poesio, M. (2008). Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Bateman, J. A., Kamps, T., Reichenberger, K., and Kleinz, J. (2001). Towards constructive text, diagram, and layout generation for information presentation. *Computational Linguistics*, 27(3):409–449.

Bateman, J. A., Wildfeuer, J., and Hiippala, T. (2017). *Multimodality: Foundations, Research and Analysis – A Problem-Oriented Introduction*. De Gruyter Mouton, Berlin.

Bateman, J. A. (2008). *Multimodality and Genre: A Foundation for the Systematic Analysis of Multimodal Documents*. Palgrave Macmillan, London.

Bateman, J. A. (2014a). *Text and Image: A Critical Introduction to the Visual/Verbal Divide*. Routledge, London and New York.

Bateman, J. A. (2014b). Using multimodal corpora for empirical research. In Carey Jewitt, editor, *The Routledge Handbook of Multimodal Analysis*, pages 238–252. Routledge, London and New York, second edition.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly, Sebastopol, CA.

Das, D., Stede, M., and Taboada, M. (2017). The Good, the Bad, and the Disagreement: Complex ground truth in rhetorical structure analysis. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms*, pages 11–19. Association for Computational Linguistics.

Engelhardt, Y. (2002). *The Language of Graphics: A Framework for the Analysis of Syntax and Meaning in Maps, Charts and Diagrams*. Ph.D. thesis, Institute for Logic, Language and Computation, University of Amsterdam.

Hiippala, T. (2015). *The Structure of Multimodal Documents: An Empirical Approach*. Routledge, New York and London.

Kaltenbacher, M. (2004). Perspectives on multimodality: from the early beginnings to the state of the art. *Information Design Journal + Document Design*, 12(3):190–207.

Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., and Farhadi, A. (2016). A diagram is worth a dozen images. In *Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV*, pages 235–251, Cham. Springer.

Kong, K. C. C. (2013). A corpus-based study in comparing the multimodality of Chinese- and English-language newspapers. *Visual Communication*, 12(2):173–196.

Kovashka, A., Russakovsky, O., Fei-Fei, L., and Grauman, K. (2016). Crowdsourcing in computer vision. *Foundations and Trends in Computer Graphics and Vision*, 10(3):177–243.

Mann, W. C. and Thompson, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Mann, W. C., Matthiessen, C. M., and Thompson, S. A. (1992). Rhetorical structure theory and text analysis. In William C. Mann et al., editors, *Discourse Description: Diverse Linguistic Analyses of a Fund-raising Text*, pages 39–78. Benjamins, Amsterdam.

O'Halloran, K. L., Tan, S., Pham, D.-S., Bateman, J. A., and Vande Moere, A. (2018). A digital mixed methods research design: Integrating multimodal analysis with data mining and information visualization for big

[3]The resources will be made available at: `https://www.github.com/DigitalGeographyLab/AI2D-RST`

data analytics. *Journal of Mixed Methods Research*, 12(1):11–30.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Taboada, M. and Habel, C. (2013). Rhetorical relations in multimodal documents. *Discourse Studies*, 15(1):65–89.

Taboada, M. and Mann, W. C. (2006). Rhetorical Structure Theory: looking back and moving ahead. *Discourse Studies*, 8(3):423–459.

Thomas, M. (2014). Evidence and circularity in multimodal discourse analysis. *Visual Communication*, 13(2):163–189.

Zhang, K. (2018). *Public Health Education Through Posters in Two World Cities: A Multimodal Corpus-Based Analysis*. Ph.D. thesis, The Hong Kong Polytechnic University.

QUD-Based Annotation of Discourse Structure and Information Structure: Tool and Evaluation

Kordula De Kuthy*, Nils Reiter†, Arndt Riester†
*Department of Linguistics, Wilhelmstr. 19, 72074 Tübingen, Germany
†Institute for Natural Language Processing, Pfaffenwaldring 5B, 70569 Stuttgart, Germany
kdk@sfs.uni-tuebingen.de,{nils.reiter,arndt.riester}@ims.uni-stuttgart.de

Abstract

We discuss and evaluate a new annotation scheme and discourse-analytic method, the *QUD-tree* framework. We present an annotation study, in which the framework, based on the concept of *Questions under Discussion,* is applied to English and German interview data, using *TreeAnno,* an annotation tool specially developed for this new kind of discourse annotation. The results of an inter-annotator agreement study show that the new annotation method allows for reasonable agreement with regard to discourse structure and good agreement with regard to the annotation of information structure, which covers *focus, background, contrastive topic* and *non-at-issue material*.

Keywords: discourse structure, information structure, annotation, inter-annotator agreement, Question under Discussion

1. Introduction

In this paper, we evaluate a new annotation scheme and discourse-analytic method, the *QUD-tree* framework, developed in Reyle and Riester (2016), Riester (to appear) and Riester et al. (to appear). Its purpose is the cross-linguistic analysis of *information structure* and *discourse structure* of textual data. We furthermore introduce a new tool, *TreeAnno*, which enables the analyst to semi-automatically segment texts, systematically enhance them with implicit *Questions under Discussion (QUDs)*, and transform the data into a new kind of discourse tree called *QUD tree*.

2. QUD trees

For several decades scholars have been claiming that implicit questions (so-called *Questions under Discussion*, or QUDs) are constitutive of the internal structure of texts, e.g. Polanyi (1988), Stutterheim and Klein (1989), van Kuppevelt (1995) or, recently, e.g. Onea (2016), Velleman and Beaver (2016), Riester (to appear). This means that every statement contained in a text is seen as the immediate answer to precisely one implicit or explicit QUD, and potentially also as an indirect answer to one or several more general QUDs. QUDs can be thought of as fine-grained, silent headlines of sections, subsections etc., down to the bottom-level of atomic assertions. The content of the latter bottom-level QUDs is, at the same time, the *background* (or *topic*) of the assertions dominated by it, while the constituent of the assertion that answers the question is called the *focus*. Therefore, QUDs also determine the information structure of the assertions contained in the text (Roberts, 2012), which is then reflected in specific choices of constituent order or the use of cleft constructions and, as to spoken discourse, in a characteristic prosodic realization.

To our knowledge, the QUD-tree method (Riester et al., to appear), briefly sketched in the following, is the first corpus annotation framework which actually *implements* the reconstruction of QUDs, which have, so far, mainly been discussed from a theoretical or experimental perspective. It is at the same time one of the few frameworks that deals with discourse structure and information structure simultaneously; but compare e.g. Kruijff-Korbayová and Steedman (2003), Polanyi et al. (2003), Poláková et al. (2013).

In our framework, we opt for a particular compact representation format for QUD trees, in which the textual assertions (A) represent the terminal nodes of a discourse tree (preserving the linear order of the text from left to right) while (implicit or explicit) QUDs (Q) form the non-terminal nodes. An abstract QUD tree is shown in Figure 1.

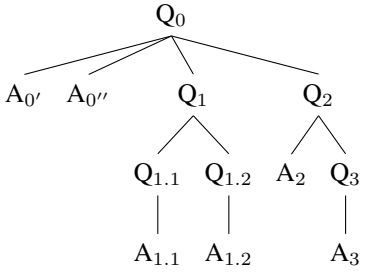


Figure 1: QUD tree

QUD trees combine properties of two types of structures from the literature. On the one hand, they comprise *d-trees* (Büring, 2003), which were designed to capture question-subquestion relations (e.g. $Q_1 > Q_{1.1}$) that guide the occurrence of so called *contrastive topics*. On the other hand, QUD trees can be systematically mapped onto discourse graphs from Segmented Discourse Representation Theory (SDRT; Asher and Lascarides, 2003). In a QUD tree, two assertions (e.g. $A_{0'}$, $A_{0''}$) count as coordinated whenever they are siblings under a joint QUD. An assertion (e.g. A_3) is subordinate to another assertion (A_2) whenever A_3 is the answer to a QUD Q_3 that is a sibling to A_2, which also means that Q_3 contentwise depends on A_2. The crucial difference between SDRT graphs and QUD trees is, of course, that the former contain discourse relations whereas the latter contain question nodes instead. For details and other tree formats representing the same information see Riester (to appear).

3. QUDs and information structure

In line with e.g. Rooth (1992), Büring (2003), Beaver and Clark (2008), Krifka (2008) or Roberts (2012), we assume a question-based definition of information-structural categories, shown in Table 1, which is also used in the examples below.[1]

Category (Label)	Definition
Focus domain ($\sim$)	Piece of discourse that has the same background as the current QUD and that contains a focus
Focus (F)	Constituent that answers the current QUD
Background (BG)	Material mentioned in the current QUD
Contrastive topic (CT)	Material backgrounded w.r.t. the current QUD and focal w.r.t. a super-question
Non-at-issue material (NAI)	Optional material w.r.t. the current QUD

Table 1: Information structure: Label inventory

(1) Q_{15}: {What did the President want to do?}
$>$ A_{15}: [It was clear from the President's speech that]$_{\text{NAI}}$ [[he wanted to]$_{\text{BG}}$ [make minor changes]$_{\text{F}}$]$\sim$

(1) is an example demonstrating the assignment of information-structure labels in the context of a QUD (in curly brackets). Note that the indentation ($>$) of A_{15} in the textual representation marks subordination in the discourse tree, as shown in Figure 2.



Figure 2: Question-answer pair as part of a discourse tree

4. QUDs and discourse structure

The QUD-tree framework can be applied to any kind of written or spoken discourse or conversation. It is not language-specific and can, in principle, be used in order to investigate data from any language the analyst is able to understand. Since the analysis procedure is described at great length within a separate guidelines document (Riester et al., to appear), we will limit the account to a minimum here.

4.1. Segmentation

Raw texts are segmented into atomic assertions. Apart from orthographic sentence boundaries, segmentation also applies at (2) (information-structurally relevant) coordinations and (3) before (optional) syntactic adjuncts. (Obligatory) sentential arguments (4) are not split off.

(2) A_1: You were working until last summer for the NSA
A_2: and during this time you secretly collected thousands of confidential documents.

(3) A_{27}: So there is a sort of a trading dynamic there
A_{28}: but it's not. . . it's not open.

(4) A_{30}: What they are saying is **that they will not then target people within that data**.

4.2. QUD principles for given information

The actual identification of a QUD for each assertion is guided by a number of well-established principles adapted from the formal literature on information structure (Rooth, 1992; Schwarzschild, 1999; Büring, 2008; Büring, 2016), cf. Riester et al. (to appear):

Q-A-CONGRUENCE: QUDs must be answerable by the assertion(s) that they immediately dominate.

Q-GIVENNESS: Implicit QUDs can only consist of given (or, at least, highly salient) material.

MAXIMIZE-Q-ANAPHORICITY: Implicit QUDs should contain as much given (or salient) material as possible.

Example (5) shows that from these principles we can derive QUD Q_{32} for assertion A_{32} in the context of A_{31}, whereas any of the questions in (6), used in place of Q_{32}, would violate at least one of the QUD constraints in the same context.

(5) A_{31}: So, if they want to spy on a British citizen, they can spy on a British citizen.
Q_{32}: {What can they do with that data?}
$>$ A_{32}: and [[then they can even]$_{\text{BG}}$ [share]$_{\text{F}}$ [that data]$_{\text{BG}}$ [with the British government]$_{\text{F}}$]$\sim$

(6) a. {What about spying?} (#Q-A-CONGRUENCE)
b. {What about the British government?} (#Q-GIVENNESS)
c. {What can they do next?} (#MAXIMIZE-Q-ANAPHORICITY)

The tree corresponding to (5) is shown in Figure 3. As a rule, the fact that an assertion contains given material leads to the subordination of that assertion under the assertion containing the antecedent. (Recall our definition of *subordination* at the end of Section 2.)

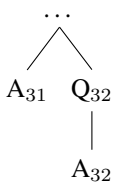


Figure 3: Subordinated assertion A_{32}, containing given material

4.3. QUD principle for parallel information

Two or more assertions are defined as parallel if and only if they share some semantically identical content and represent partial answers to the same QUD, see Example (7).

[1] Aboutness topics (T), defined in the guidelines as "referential entities (terms) in the background", were left out of consideration, in order to avoid categorical embedding.

PARALLELISM: The background of a QUD with two or more parallel answers consists of the (semantically) common material of the answers.

(7) Q_{19}: {What about the programs?}
$> A_{19'}$: [[they've]$_{BG}$ [never stopped a terrorist attack in the United States]$_{F}$]$\sim$
$> A_{19''}$: and [[they have]$_{BG}$ [marginal utility at best for other things]$_{F}$]$\sim$.

The resulting tree structure is shown in Figure 4.

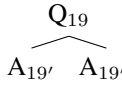


Figure 4: Two coordinated (parallel) assertions

5. TreeAnno

To support the annotation of QUDs according to the above described QUD principles, we developed *TreeAnno*, a web-based tool for the transformation of written text into QUD trees. Conceptually, two distinct steps are involved in the annotation: The segmentation of the text into appropriate units, and their hierarchical organization in the form of QUD trees. The tool imports plain text files, and first adds automatically detected sentence (and token) boundaries (using LanguageTool[2]), as an initial segmentation. The boundaries can later be changed by the annotators. The tree annotation works by indenting segments, similarly to an outliner. This allows for fast annotation of large text segments. Implicit QUDs can be inserted as well. Figure 5 shows a screenshot of the annotation view of *TreeAnno*.

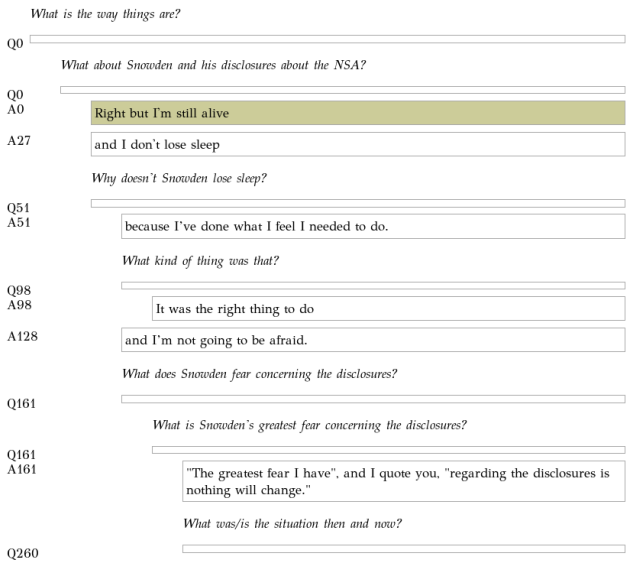


Figure 5: Annotation view of *TreeAnno*: discourse structuring and added QUDs *(italics)*

Behind the scenes, the sentences (and implicit QUDs) are represented as annotations in a UIMA[3] document. The connection to actual textual positions is always kept and allows future integration with other linguistic annotation layers. The tree structure is provided by maintaining a reference to the parent of each annotation.
In addition, TreeAnno supports the export of annotated documents as a simple, tree-oriented XML format, bracket expression (with or without node ids), visually rendered trees (by using GraphViz) or a chart-like matrix.
To sum up, *TreeAnno* allows for an easy transfer of QUD-tree annotations, which have, for instance, initially been carried out by various annotators in a text editor, into a generic XML or bracketing format. It furthermore provides the possibility to visualize, compare and evaluate different annotations, e.g. the ones shown in Figure 6. The tool will be made available as open source software released with this publication.[4]

6. Evaluation: Discourse structure

In a first evaluation of the QUD guidelines, our goal is to show that the above described method of discourse annotation in terms of QUDs can be applied reliably to naturally occurring data. We conducted an empirical study, in which annotators followed the QUD guidelines described in Riester et al. (to appear) to annotate English and German interview data with QUDs, using the above described TreeAnno tool.

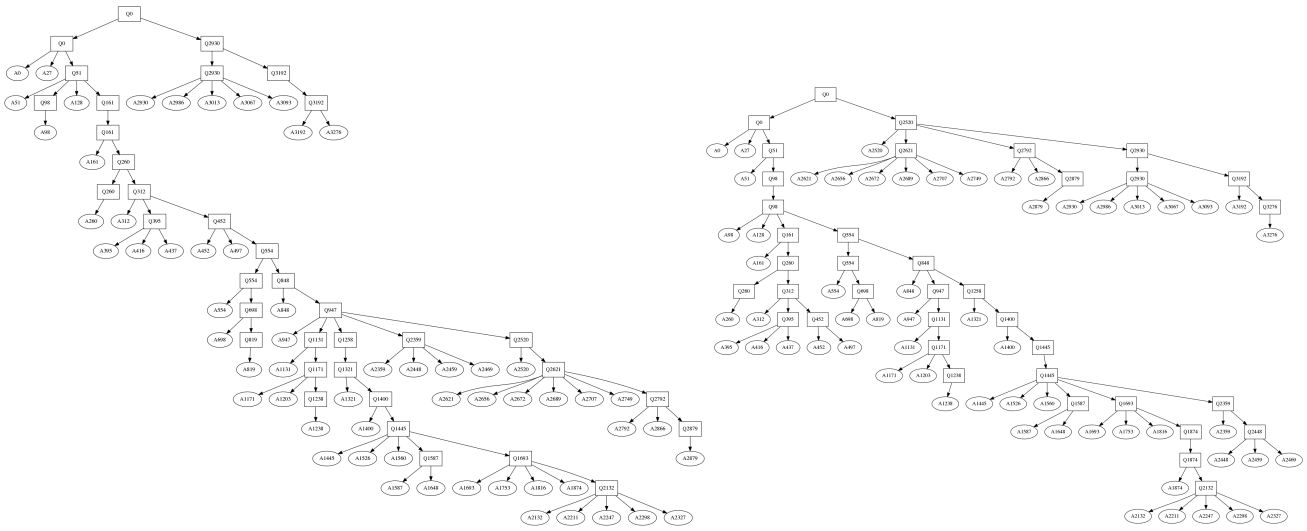
6.1. Evaluation setup

Two trained annotators analyzed two sections from a transcript of an (English) interview with Edward Snowden, broadcast on German ARD TV on Jan. 26, 2014.[5] The first section of the transcript consists of 60 text segments, the second has 69 text segments. The two resulting discourse trees for the first segment are shown in Figure 6.
Two other trained annotators analyzed a German radio interview (SWR2 radio interview with Thomas Oppermann, Social Democratic Party, Sept. 12, 2015), in the form of a single document consisting of 158 segments.[6]

6.2. Method and results

For the comparison of two QUD annotations we need to be able to calculate an inter-annotator agreement score that takes into account, for every segment and every possible span of segments, whether a QUD is present or not. In order to compute a κ statistics based on our QUD annotations, we follow the method described in Marcu et al. (1999), which was developed for measuring agreement in the labeling of rhetorical structure categories in texts. The method is based on the idea of mapping the hierarchical structure of a discourse tree onto a matrix filled with categorical values (in our case whether there exists a (Q)uestion spanning the

[2] https://languagetool.org
[3] http://apache.uima.org
[4] http://hdl.handle.net/11022/1007-0000-0007-C634-F
[5] https://archive.org/details/snowden_interview_en
[6] The interview is part of the GRAIN corpus (Schweitzer et al., 2018), which, among other data and annotation layers, comprises twelve more interviews of the same kind, which are all analyzed for QUD trees and information structure.



QUD tree of annotator 1 QUD tree of annotator 2

Figure 6: Two different QUD tree analyses for the same document

respective segments – start to end – or (n)ot). The resulting chart for the sample QUD tree of Figure 1 is shown in Figure 7, in which we observe, for instance, that the root node Q_0 spans over all the tree segments, i.e. from $A_{0'}$ to A_3. Note that the indices on the segments/assertions and questions in this Figure are marked only for the purpose of demonstration, while, in fact, the labels are binary.

	$A_{0'}$	$A_{0''}$	$A_{1.1}$	$A_{1.2}$	A_2	A_3
$A_{0'}$	n	n	n	n	n	Q_0
$A_{0''}$	–	n	n	n	n	n
$A_{1.1}$	–	–	$Q_{1.1}$	Q_1	n	n
$A_{1.2}$	–	–	–	$Q_{1.1}$	n	n
A_2	–	–	–	–	n	Q_2
A_3	–	–	–	–	–	Q_3

Figure 7: A text segment chart representing a QUD tree

A κ statistics can now be computed between two charts that represent two different QUD annotations for the same text; more precisely between the two resulting sets of cells in the upper half of each chart. In the case of Figure 7, this amounts to sets of 21 pairs of cells. Generally, for n segments contained in a document, the number of cells is $\frac{n\times(n+1)}{2}$.

For our three annotated documents we calculated κ (Cohen, 1960), based on the the described method. For the text *Snowden 1,* consisting of 60 segments, we calculated the κ statistics based on 1,830 items, for *Snowden 2* with 69 segments based on 2,415 items. And for the German *Oppermann* text, the κ is based on 12,535 items resulting from 158 segments. The results are shown in Table 2

The values show moderate agreement between the annotator pairs. It is entirely clear, though, that the basis of computation in this new task is rather different than, for instance, in a word-based classification task. It is, therefore, perhaps still too early to interpret the results, due to the overall complexity of the task and the lack of a reasonable baseline. However, the results can provide a point of reference for future developments in this area.

Text	Segments	Cells	κ
Snowden 1 (ENG)	60	1,830	.50
Snowden 2 (ENG)	69	2,415	.53
Oppermann (GER)	158	12,561	.45

Table 2: Kappa values for QUD-annotated spoken dialogue

7. Evaluation: Information structure

The second major issue we are interested in is to evaluate the reliability of information-structure annotation based on the previous identification of QUDs.

7.1. Evaluation setup

For the evaluation of the information structure mark-up, the same two documents from the English Snowden interview as well as the transcript of the German Opperman interview were annotated. The same pairs of trained annotators now performed an information-structure annotation of the text segments, still in keeping with the guidelines of Riester et al. (to appear). To keep matters simple, we concen-

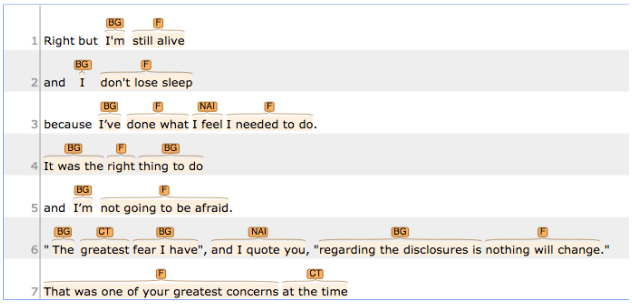


Figure 8: Annotation of information structure in WebAnno

trated on the four categories focus (F), background (BG), non-at-issue material (NAI) and contrastive topic (CT). Focus domain labels (∼) were not annotated, since each text segment (assertion) already corresponds to one focus domain. The annotators based their annotations on the previously performed QUD analysis in the TreeAnno tool described in Section 6.2. As an annotation tool for the token-based information-structure annotation, WebAnno (Yimam et al., 2013) was chosen. Figure 8 shows a screenshot of the information-structure annotation of the beginning part of *Snowden 1*.

7.2. Method and results

Following previous work on the evaluation of information structure annotation (Ritz et al., 2008; Calhoun et al., 2010), we calculated κ values on the annotated data based on tokens. In addition to the specifications in Riester et al. (to appear), in particular the QUD-to-information-structure mapping from Table 1, we defined a number of heuristic (but potentially debatable) rules in order to prevent disagreement due to theoretically unclear issues:

- Discourse connectors (*but, and, although, because, therefore* etc.) at the beginning of discourse segments are not annotated.

- All pronouns (including possessive pronouns), unless contrastive, receive the label BG.

- Function words like auxiliaries, prepositions, discourse particles, articles or complementizers are either labeled as BG or as F, depending on what they adjoin to. In case a function word occurs between a focus and a background, it is backgrounded, on the assumption that it represents salient information.

  (8) Q: {What about John?}
  A: [He **has been**]$_{\text{BG}}$ [lucky]$_{\text{F}}$.

  Exceptions:

  - If the wh-word provides that a function word is part of the focus, or if two function words are explicitly contrasted against each other, then they receive the label F.

    (9) Q: {Whom did she meet?}
    A: [She met]$_{\text{BG}}$ [**the** Pope]$_{\text{F}}$.

  - If a function word is adjoint to an overtly contrastive word, it stays in the background.

    (10) Q: {What kind of cars were there?}
    A': [**A**]$_{\text{BG}}$ [red]$_{\text{F}}$ [**one**]$_{\text{BG}}$
    A": and [**a**]$_{\text{BG}}$ [green]$_{\text{F}}$ [**one**]$_{\text{BG}}$.

- Punctuation: Quotation marks around an expression, commas within and at the right edge of an expression are part of the markable. Periods, colons, semicolons, exclamation marks are not.

Text	Label	Tokens	κ
Snowden 1	all	657	.69
(ENG)	F		.67
	BG		.46
	CT		.55
	NAI		.71
Snowden 2	all	842	.67
(ENG)	F		.65
	BG		.57
	CT		.61
	NAI		.71
Oppermann	all	1646	.67
(GER)	F		.63
	BG		.60
	CT		.14
	NAI		.69

Table 3: Kappa for information structure annotation

Results are shown in Table 3, divided into scores for all labels taken together, and individual scores for each of the four labels.
The results show that the described method supports the successful information-structure annotation with substantial agreement (i.e. $\kappa > .6$) between two annotators: On the *Snowden 1* text, the agreement score for all annotated categories taken together is at .69, for the category F (focus) the score is .67, while on *Snowden 2*, the score for all categories is at .67, for F at .65. As for the German text, the overall agreement score is again .67, for the category F the score is .63. However, the score for CT (contrastive topic) was very low here. Moreover, in general, there was a high agreement for the classification of non-at-issue material (NAI).
These agreement scores are much higher and exhibit more reliable annotation results than, for example, the results reported in (Ritz et al., 2008) for a similar annotation study on naturally occurring data. In their study, the highest κ value calculated for focus on all tokens of a spoken dialogue is at .44. Other studies reporting higher κ values usually did not base their annotation on all tokens of a text or used fewer categories in the annotation. Calhoun et al. (2010), for example, report a κ value of .67 for the binary distinction between kontrast (their terminology for focus) and background. However, in their study not all tokens of a text but only certain words were annotated, i.e. nouns, verbs, adjectives, adverbs and pronouns. Summing up, these high agreement scores show that the successful annotation of information structure in spoken-language data based on explicit QUDs is very promising, despite the fact that there is still some degree of disagreement on the QUD-based discourse structures.

8. The formulation of QUDs

Finally, we would like to address the similarity of the actual QUD formulations chosen by the two annotators. It might seem surprising at first that we are not evaluating this issue, although a substantial amount of variation can be expected here and indeed occurs, as shown, for instance, in Figure 9. We think that it would be rather futile to evaluate the string match between two annotators' free QUD

formulations, since language allows for endless possibilities of variation when expressing any statement or question. But this is not an important point, as far as our task is concerned. What counts is that the two QUDs chosen by the annotators have the same denotation, i.e. give rise to the same discourse-structure analysis and information-structure classification, which in the case of Figure 9 is fulfilled.

	Annotator 1:	**Annotator 2:**
Q_9:	{What did the political class do?}	{What exactly did the government do?}
$> A_9$:	[Instead of circling around the public and protecting their rights]$_{\text{NAI}}$ [the political class]$_{\text{BG}}$ [circled around the security state]$_{\text{F}}$	

Figure 9: Two different QUDs by two annotators, which lead to the same discourse and information structure

9. Conclusion

We have presented a novel method for the annotation of information structure which achieves good inter-annotator scores. The method is based on the reconstruction of QUDs, which moreover leads to the definition of a new kind of discourse structure, QUD trees. Although initially earmarked for the annotation of information structure, QUD trees represent an interesting contribution to discourse theory itself, which can be analyzed with a reasonable agreement. Finally, we introduced a new annotation tool for QUD trees, TreeAnno, which is made available as open source software.

10. Acknowledgements

We would like to thank our student annotators Daniel Knaus and Nadja Schäfer for their dedicated work and helpful comments. We would also like to thank Detmar Meurers for his insightful comments and discussion. This work was funded by the German Research Foundation (DFG) via the Tübingen Sonderforschungsbereich 833, Project A4 and the Stuttgart Sonderforschungsbereich 732, Project A6. TreeAnno has been developed within the Center for Reflected Text Analytics (CRETA), funded by the German Ministry of Education and Research.

11. Bibliographical References

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press.

Beaver, D. and Clark, B. (2008). *Sense and Sensitivity. How Focus Determines Meaning*. Wiley-Blackwell, Oxford.

Büring, D. (2003). On D-trees, beans, and B-accents. *Linguistics and Philosophy*, 26(5):511–545.

Büring, D. (2008). What's new (and what's given) in the theory of focus? In *Proceedings of the 34th Annual Meeting of the Berkeley Linguistics Society*, pages 403–424.

Büring, D. (2016). *Intonation and Meaning*. Oxford University Press.

Calhoun, S., Carletta, J., Brenier, J., Mayo, N., Jurafsky, D., Steedman, M., and Beaver, D. (2010). The NXT-format Switchboard corpus: A rich resource for investigating the syntax, semantics, pragmatics and prosody of dialogue. *Language Resources and Evaluation*, 44:387–419.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Krifka, M. (2008). Basic notions of information structure. *Acta Linguistica Hungarica*, 55(3-4):243–276.

Kruijff-Korbayová, I. and Steedman, M. (2003). Discourse and information structure. *Journal of Logic, Language and Information*, 12:249–259.

Marcu, D., Amorrortu, E., and Romera, M. (1999). Experiments in constructing a corpus of discourse trees. In *Proceedings of the ACL Workshop on Standards and Tools for Discourse Tagging*, pages 48–57, College Park, MD, USA.

Onea, E. (2016). *Potential Questions at the Semantics-Pragmatics Interface*. Brill, Leiden.

Poláková, L., Mírovský, J., Nedoluzhko, A., Jínová, P., Zikánová, Š., and Hajičová, E. (2013). Introducing the Prague Discourse Treebank 1.0. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 91–99, Nagoya.

Polanyi, L., van den Berg, M., and Ahn, D. (2003). Discourse structure and sentential information structure: An initial proposal. *Journal of Logic, Language and Information*, 12:337–350.

Polanyi, L. (1988). A formal model of the structure of discourse. *Journal of Pragmatics*, 12(5-6):601–638.

Reyle, U. and Riester, A. (2016). Joint information structure and discourse structure analysis in an Underspecified DRT framework. In Julie Hunter, et al., editors, *Proceedings of the 20th Workshop on the Semantics and Pragmatics of Dialogue (JerSem)*, pages 15–24, New Brunswick, NJ, USA.

Riester, A., Brunetti, L., and De Kuthy, K. (to appear). Annotation guidelines for Questions under Discussion and information structure. In Evangelia Adamou, et al., editors, *Information Structure in Lesser-Described Languages: Studies in Syntax and Prosody*. Benjamins, Amsterdam.

Riester, A. (to appear). Constructing QUD trees. In Malte Zimmermann, et al., editors, *Questions in Discourse. Vol. 2: Pragmatics*. Brill, Leiden.

Ritz, J., Dipper, S., and Götze, M. (2008). Annotation of information structure: An evaluation across different types of texts. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, pages 2137–2142, Marrakesh.

Roberts, C. (2012). Information structure in discourse: Towards an integrated formal theory of pragmatics. *Semantics and Pragmatics*, 5(6):1–69.

Rooth, M. (1992). A theory of focus interpretation. *Natural Language Semantics*, 1(1):75–116.

Schwarzschild, R. (1999). GIVENness, AvoidF, and other constraints on the placement of accent. *Natural Language Semantics*, 7(2):141–177.

Schweitzer, K., Eckart, K., Gärtner, M., Faleńska, A., Riester, A., Rösiger, I., Schweitzer, A., Stehwien, S., and Kuhn, J. (2018). German Radio Interviews: The GRAIN release of the SFB732 silver standard collection. In *Proceedings of the Eleventh Language Resources and Evaluation Conference (LREC)*, Miyazaki, JP.

Stutterheim, C. v. and Klein, W. (1989). Referential movement in descriptive and narrative discourse. In Rainer Dietrich et al., editors, *Language Processing in Social Context*, pages 39–76. North Holland, Amsterdam.

van Kuppevelt, J. (1995). Discourse structure, topicality and questioning. *Journal of Linguistics*, 31:109–149.

Velleman, L. and Beaver, D. (2016). Question-based models of information structure. In Caroline Féry et al., editors, *The Oxford Handbook of Information Structure*, pages 86–107. Oxford University Press.

Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). WebAnno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Sofia.

The Spot the Difference corpus: a multi-modal corpus of spontaneous task oriented spoken interactions

José Lopes, Nils Hemmingsson, Oliver Åstrand
KTH Royal Institute of Technology
Stockholm, Sweden
jdlopes@kth.se

Abstract

This paper describes the Spot the Difference Corpus which contains 54 interactions between pairs of subjects interacting to find differences in two very similar scenes. The setup used, the participants' metadata and details about collection are described. We are releasing this corpus of task-oriented spontaneous dialogues. This release includes rich transcriptions, annotations, audio and video. We believe that this dataset constitutes a valuable resource to study several dimensions of human communication that go from turn-taking to the study of referring expressions. In our preliminary analyses we have looked at task success (how many differences were found out of the total number of differences) and how it evolves over time. In addition we have looked at scene complexity provided by the RGB components' entropy and how it could relate to speech overlaps, interruptions and the expression of uncertainty. We found there is a tendency that more complex scenes have more competitive interruptions.

Keywords: Dialogues, Spontaneous, Multi-modal

1. Introduction

Despite the recent advances in the field of Spoken Dialogue Systems (SDSs), non task-oriented spontaneous dialogue is still a very challenging problem since its structure is often difficult to represent, unlike task-oriented dialogues which could easily be represented by a flow chart. Therefore, DeVault (DeVault, 2008) compared task-oriented dialogue to assembling furniture. On the other hand, non task-oriented dialogue may be compared to dancing. To be able to dance one needs to learn the steps, but this might not be enough since dancing is a collaborative task where coordination is extremely important. Unlike what happens with the furniture assembling case, there is no going back to the point where the mistake was done. In spontaneous dialogues as in dancing, the dialogue should continue, despite the mistakes. There are mechanisms that help to regulate these situations, for instance language, turn-taking and other types of non-verbal behavior. SDSs would greatly benefit from if they could understand these mechanisms in order to be able to anticipate moments in a dialogue where some sort of communication breakdown is about to happen, so they could act in a more human like fashion both in realizing the breakdown and finding an appropriate solution to it.

While replicating and studying these phenomena in a non task-oriented dialogue might be too complex given the current state-of-the-art of spoken dialogue systems, there are intermediate steps that can help us in this process. The step that we present in this paper is the Spot the Difference corpus. This is a corpus of task-oriented collaborative dialogues between humans using spontaneous speech. We think that this corpus is a valuable resource in the study of spontaneous dialogues both in terms of verbal and non-verbal behavior since the corpus release includes several annotations, audio and video data from the interactions. Unlike other similar tasks such as the Map Task (Anderson et al., 1991), participants are free to choose the order in which they could discuss the objects in the scene. These resulted in a less structured data, but certainly richer in spontaneity.

In this paper we will describe the Spot the Difference corpus in detail: the experimental setup used, the whole procedure, the participants data and the annotation already performed on this data. To prove the usefulness of the corpus, we performed a preliminary analysis where we studied if certain aspects of communication are related to the complexity of the task.

2. Background

There are several mechanisms and efforts that can be used to improve human communication. In the particular case of dialogues occurring in the scope of a collaborative tasks, these efforts serve the purpose of achieving the common goal by the participants in the dialogue.

These mechanisms or efforts can be expressed at the linguistic level. For instance, several studies have shown how important entrainment and coordination can relate to success in task oriented dialogues. Whereas entrainment presupposes an adaptation between speakers over time, coordination can be present from the very beginning of the dialogue in the way the speakers interact with each other. There is a fair number of studies which are focused on the role of lexical and syntactic coordination in both in task and non task-oriented dialogues. For instance, studies by (Garrod and Anderson, 1987) and (Brennan and Clark, 1996) have focused on participants' coordination in terms of lexical items. (Reitter and Moore, 2007) showed that for task solving in dialogue, lexical and syntactic repetition is a reliable predictor of task success given the first five minutes of task oriented dialogue. (Friedberg et al., 2012) found a significant difference in the performance of student engineering groups related to lexical entrainment. The high performing groups, increased their entrainment over time, whereas the low performing groups tended to decrease their lexical entrainment with time. (Nenkova et al., 2008) in-

vestigated entrainment in the use of the most commonly used words in the Switchboard (Godfrey et al., 1992) and the Columbia Games corpus (Benus et al., 2007), as well as its perceived naturalness, flow and task success. Their results indicate that entrainment in commonly used words, predicts of the perceived naturalness of dialogues and is significantly correlated with task success. The aforementioned efforts are ways of optimizing the dialogue. Other linguistic mechanisms are normally used when the dialogue falls below the optimization line, for instance repairs. These have been previously studied in the context of the Map Task (Colman and Healey, 2011), where it was shown that patterns are cross-person and cross-turn. Given the nature of the task, the mechanisms above memtioned often go together with the study of referring expressions (RE) and reference resolution (RR). The corpora presented in (Zarrieß et al., 2016) include examples where REs are improved over time to achieve a common ground. This process may require participants to use self-repairs in their utterances, and adjust them with their partner over time for efficiency purposes. The fact that the corpus we are releasing is multimodal could benefit an integrated approach to improve the understanding of dialogue utterances such as the one presented in (Kennington et al., 2013).

But these mechanisms and efforts are not exclusively linguistic. (Nenkova et al., 2008) found that higher degrees of entrainment are associated with more overlaps and fewer interruptions. (Oviatt et al., 2015) investigated overlapped speech in groups of students trying to jointly solve math problems. They found that during the most productive phases of the interactions the amount of overlap was higher when compared to other phases of the problem solving. Moreover, they could also show that the domain experts differed in the kind of interruptions they made from non-domain experts. (Goldberg, 1990) stated that interruptions may be used to convey rapport in competitive settings and (Poesio and Rieser, 2010) mentioned them as signs of coordination and alignment. However, in a similar set up to the one used in this study, (Bull and Aylett, 1998) found that the complexity and the lack of familiarity with the tasks could result in longer gaps.

Uncertainty display is another mechanism that indicates that the dialogue might approaching a point where some recovery strategy might be needed. It has already been studied in the scope of tutoring dialogues (Liscombe et al., 2005), but also in spontaneous speech (Schrank and Schuppler, 2015). Although tutoring dialogues can be seen as a collaborative dialogue, there is no short term goal, and therefore we hypothesize that the display of uncertainty will be different that we observe in our corpus, thus reinforcing the importance of the resource that we are releasing.

3. Data collection

3.1. Procedure

The Spot the Difference corpus was recorded during 2016 at KTH. Participants were recruited via mailing lists and word of mouth. Participants were required to speak English and had to fill in a small personality questionnaire before the experiment. This questionnaire included a small subset of the Big Five Inventory (John et al., 2008) with the 8 questions used to place individual in the introvert/extrovert axis. 36 participants took part in the experiment. Participants were informed that they were participating in an experiment to investigate human dialogues in a collaborative setting.

They were briefly instructed about the task: they had to collaborate to find all the differences in two very similar scenes, such as those shown in Figure 1. Since they were sitting in different rooms with no visual contact, they were forced talk to each other in order to discuss the scene they had in front of them. To enable communication between the two of them, one of the head-mounted microphones used was connected to their partners computer speakers. They would describe the pictures in front of them and as they found differences, participants should engage in a sub-dialogue to locate where the difference was as precisely as possible and use the mouse to click in that area. They were informed that if they would not click the same area, the difference would not be recorded as found. There were two roles in attributed in the beginning of the dialogue: the Instruction Giver (IG) and the other the Instruction Follower (IF). The IG had to lead the discussion by describing and locating the objects in the scene, whereas the IF had to follow to the IG's instructions, make clarification requests when necessary. An excerpt of a dialogue can be found in Table 1. The roles were randomly assigned once the participants did the first set of three scenes and kept over the course of the experiment. To get familiarized with the task participants had a chance to do a training scene before the experiment started.



Figure 1: Example from the beach scenes used.

In 1 we have transcribed a sample dialogue, where the participants are discussing the scene in Figure 1.

IG: OKAY THEN I HAVE LIKE <F-> <%aa> THREE <%aa> <L-> BEACH LYING CHAIRS
IF: OKAY
IG: YEAH
IG: YOU TOO?
IF: YES TWO OF THEM BLUE ONE IS ORANGE
IF: THAT
IG: OH THEY'RE <THE> THE THREE BLUE
IF: OKAY THE ONE ORANGE IS THE ONE IN THE BOTTOM
IG: YEAH OKAY LET'S CLICK THERE I DON'T HAVE THAT

Table 1: Excerpt of a dialogue transcriptions captured during the discussion about the scene pictured in Figure 1.

Whenever the time limit (200 seconds) was reached or the participants agreed to click the button to show the solution, the correct solution and the score was shown to both participants while the audio channel was kept open. Facing the solution the participants would have the possibility to discuss the missed differences and refine their strategy for the coming scenes. The IG had to click a button to continue to the following scene, once both partners had agreed to do so, unless it was the last scene of the set. In that case, no button would be shown. The procedure was repeated until each participant had completed three different sets. Including the necessary set ups, the experiment took about an hour. The participants would receive a cinema ticket as a compensation for their participation.

3.2. Infrastructure

The game was implemented with IrisTK (Skantze and Al Moubayed, 2012), already envisaging a future implementation in a dialogue system. IrisTK was running in parallel in two different machines, they were communicating with one another sending event messages, namely those generated by the eye-tracker and the mouse clicks on the scenes and respective coordinates. Each scene had a corresponding XML file with the description of the spatial arrangement of the objects in the scene, including the coordinates of the object, the radius, color, if the object corresponded to a difference, if the object was visible and possible ways of referring to that object.
In the beginning of each set the streams were synchronized using a beep sound. This step was necessary since the timestamps for the IrisTK events (mouse clicks and eye tracking among others) were from the native computer.

3.3. Participants

From the 36 participants 14 were female subjects and 22 were male subjects. 9 of the female speakers were assigned the role of IG and 4 the role of IF. 9 of the male subjects were assigned the role of IG and the remaining 13 the role of IF. The average age of the participants was 34.3. Among the participants there were 18 different mother languages and only one native English speaker. All the non-native speakers but one were fluent in English and claimed to have English as their language at work. The most represented native language was Swedish (7 subjects), followed by Portuguese (6 subjects), Spanish and Farsi (4 participants) and French (2 participants). All the other native languages only had one participant. Only 4 dialogues out of the 54 were held between subjects with the same mother tongue. We avoided subjects that were acquainted from before to take part in the same session since this might have implications in their interaction.

3.4. Experimental setup

Each set was composed of three scenes: an easy one, an average one and a difficult one. The difficulty level was assigned after the number of differences between pictures (more differences meaning higher difficulty). All the participants did each scene once, but the order of the scenes and the sets was randomized in order to avoid that the order of the scenes had an effect on the experiment. Since one of our goals was to create a corpus of spontaneous speech where we could study language, turn-taking and non-verbal behavior and how these evolve over time if partners don't change, we divided the subjects in two groups. The first group performed each set with a different partner (3 scenes with each partner). This will further on be referred as condition A. The second group performed all the 3 sets with the same partner, and this will be hereafter called condition B.

3.5. Collected data

The data recorded includes audio from the two head mounted microphones used by each participant. One of the microphones was the one used to communicate with the partner. From these microphones two mono audio files were generated per scene, one for each speaker. The other two microphones were connected to the same sound-card. From these microphones a stereo audio file was generated per set, with one speaker in each channel. Since these microphones recorded a complete set, the discussion that occurs once the solution is shown to the partipants until a new started is also included. We also recorded eye-tracking data with two eye-trackers placed under each screen. The eye-tracking data contains both raw fixation data and fixation data for objects specified in the XML scene description. The raw fixation data was stored in the IrisTK log file. The fixation data for the objects was also saved in the IrisTK log file, but was, in addition, converted into a Praat Tier where each interval was labelled with the corresponding object identifier. Mouse click coordinates were also saved as a point tier in Praat. Finally, videos from two GoPro placed on top of each eye-trackers was also recorded for each set.

3.5.1. Transcription

The audio for each speaker was recognized using the IBM Watson speech recognizer[1]. The output of the ASR was converted into a Praat interval tier. This allowed a reasonably accurate estimate of the content that was actually said, but most important, accurate time boundaries for the speech utterances in order to study turn taking behavior.
The data was also manually transcribed including disfluency annotations using the coding scheme defined in (Moniz et al., 2014). Part of this work was done on transcription made from scratch, while another part was done correcting the ASR output and inserting the disfluency annotation.

3.5.2. Annotation

Filled pauses were annotated separately by one annotator, without taking the transcription data into account. Annotated filled pauses were English filled pauses "eh, ah, aa, ahm" as well as filled pauses for the mother tongue of the participants. Since the transcription also coded Filled Pauses and was performed by a different annotator, it was possible to compute the agreement for Filled-Pause annotation which was in this case 0.61, which can be seen as moderate agreement.
Each conversation was manually annotated per topic by one annotator. The topic labels used were Describing scene (DS), Describing object (DO), Locating difference (LD)

[1] https://www.ibm.com/watson/services/speech-to-text

and End of dialogue (EOD). DP corresponds to the segments where a participant is describing the scene focusing in the spatial relation between the objects in the scene. DO was used for segments where participants were describing some characteristic of a specific object. LD was used for the segments where participants were discussing the exact location of the difference. EOD dialogue corresponds to segments where participants are negotiating whether they should press the button to show the solution. These choice of topics followed an hypothesis that participants behaviour would change between topics as the graph in Figure 2 shows for participation equality (Lai et al., 2013).

We also automatically extracted all the speech overlaps found in the data and we annotated those which corresponded to floor changes as interruptions or not. Among overlaps, we considered interruptions whenever the interrupted speaker was not able complete the sentence. Therefore there might be other interruptions in the data that do not follow overlaps which, for the current analysis, were not taken into account. The interruptions were further divided into collaborative and competitive interruptions. We labeled as collaborative the interruptions those where the interrupting speaker completes the sentence and competitive interruptions were labeled whenever the interrupting speaker utters something unrelated to the interrupted sentence.

Furthermore, the video data was annotated for uncertainty using ELAN. In doing so, we formulated the binary definition "A conversation participant is *uncertain* when they feel they do not understand what the counterpart is trying to communicate or that they do not know what to say". Those intervals in the videos where the annotator had this perception were annotated as uncertain and the remainder as certain. Each video was annotated by one annotator.

3.5.3. Data formats

Information about the dataset will be released in a JSON file. This JSON files contains all the dialogue ids and a link to a JSON set file. This JSON file includes metadata about the participants can be found, together with other relevant data about the session such as the scenes used, the data files (audio and video) and respective offsets, the log files and the annotations files for the scene: the Praat annotation file and the respective tiers for each participants and the uncertainty annotation ELAN file. Whenever the audio for the whole set was available, all the turns in the set, including those where participants are refining their strategy between scenes are included, with information about time boundaries, speaker, turn index, topic and rich transcriptions. The set JSON files further link to scene files. The scene JSON include information about the dialogue success, duration of the dialogue, scene audio files, and turns corresponding to the scene. In both set and scene JSON files, there is information about overlaps. The overlaps contain the time boundaries of the overlap, the speaker before and after the overlap, and the turn-index of the overlapping turn. The dataset can be accessed in `https://github.com/zedavid/SpotTheDifferenceData`.

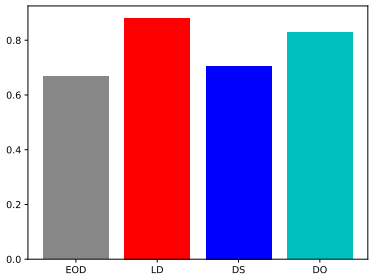


Figure 2: Participation Equality per Topic, the closer to 1 the more even is the participation.

4. Data analysis

Considering a dialogue as set of three scenes, 54 dialogues were recorded with this setup. Due to technical issues 4 of these dialogues could not be used in our data analysis. For this data we found that for each scene the average duration time was 188.3 seconds, meaning that there were scenes where the participants decided to unveil the solution before the time limit was reached. The average number of turns per dialogue was 121.4 (standard deviation 44.4).

Despite the fact that participants were assigned a specific role, in some dialogues we even observed these somehow reversed during the course of the interaction. In addition, as we mentioned before the participants could discuss their performance after the scene and refined their strategy, which can be seen as engaging factor and an incentive to improve the way they collaborate in the coming scenes.

From the experimental design, we implicitly hypothesized that there was a learning curve both to get used to the task and to interact with each partner. If the latter would have played a significant role, we would expect that the participants in condition B would, as they accumulate experience, perform better that those in condition A. As a matter of fact, as seen in Figure 3, we see that there is a similar progression in the task success achieved in each condition. According to the linear approximation, participants in condition A have improved more their task success than those in condition B during the course of the experiment. This means that the hypothesis there is a learning curve associated with the partner does not stand in our data.

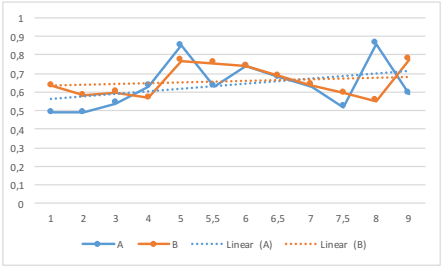


Figure 3: Average relative number of differences found (y-axis) against average expertise in the task between the participants (x-axis).

Another interesting result is the plot displayed in Figure 4.

The initial division of the scenes according to the difficulty level does not seem to correspond to the average relative number of differences found. The scenes beach (Figure 1) and sea (Figure 11) seemed more complex than all the scenes in the average group and the sheep scene (Figure 12) in the difficult group.

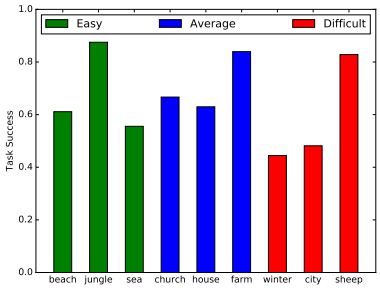


Figure 4: Task success per scene and initially defined level of difficulty for each scene.

These results seem to indicate that there are other factors that contribute to the scene complexity other than the number of differences between them. Therefore, we hypothesize that there could be a way to correlate the scene complexity with the task success. For this, first we have computed the entropy of the histogram of the RGB components. We found out that, except for house (Figure 9), the scenes with highest success rate (jungle in Figure 10, farm in Figure 8 and sheep) were those with the lowest entropy values. Another factor that could have also contributed to the complexity of the scene is the number of objects in the picture. This information could be easily obtained parsing the XML files for each picture and counting the number of objects defined there. In this process we can even differentiate between group and single objects. Combining this information with the entropy of the scene, we can hypothesize that the complexity of the scene in this setup could be given by a combination between entropy and the number of objects. The house scene (Figure 9) has low entropy, since there is very little color variation, but it has 21 objects (median value is 15). On the other hand, the scene sea (Figure 11) has 9 objects in the XML description, but the entropy value in 3.99 (median value is 3.57).

We hypothesize that the complexity of the scene would also have an impact on the mechanisms that mediate the interaction. Therefore we present a preliminary study where we tried to relate interruptions and uncertainty to the scene complexity. Just like in dancing, when the steps of the dance are more difficult, there is a higher chance that people step on each other's feet. Figure 5a shows the comparison between the number the total of overlaps and the number of overlaps followed by an interruption, Figure 5b shows the comparison between the number of competitive and collaborative interruptions per scene and Figure 5c shows the average time where participants showed uncertainty for each picture. The graph from Figures 5a, particularly concerning the number of interruptions per picture shows a similar trend to the graph show in Figure 4, that is the scenes with high average relative number of differences found were those with lower average number of interruptions, particularly competitive interruptions.

Overlaps do not show a similar trend. In Figure 5b similar trends hold, and the pictures with the lowest number of collaborative interruptions are exactly those where participants where the success rate was higher. The graph in Figure 5c, does not present the same trends a the previous two. We further explored interactions between uncertainty, interruptions and overlaps. We found a significant interaction between interruptions and uncertainty for both IGs, IFs and both participants combined in Chi-square test performed ($p - value < 0.05$, $p - value < 0.001$ and $p < 0.01$, respectively). Regarding overlaps and uncertainty, we also found significant differences for IFs and both participants combined ($p - value < 0.05$ and $p < 0.001$, respectively), but not for IGs according to the Chi-square test performed. These are interesting results, which can indicate that a multi-modal approach for breakdown detection can be worth investigating. In dialogue like in dancing, if someone steps on the partners foot, this might have impacts in the rest of the movements the body needs to perform.

We made a further analysis regarding uncertainty to assess the difficulty of predicting uncertainty from facial features and gaze. Using OpenFace (Baltrušaitis et al., 2016) to extract a set of facial features together with the variance of the gaze movement. After trying several different techniques we achieved an 62% accuracy as our best result (54% was be the majority baseline) using Artificial Neural Networks in an evenly distributed subset of the data (the original data set is highly skewed towards certain segments).

5. Conclusion

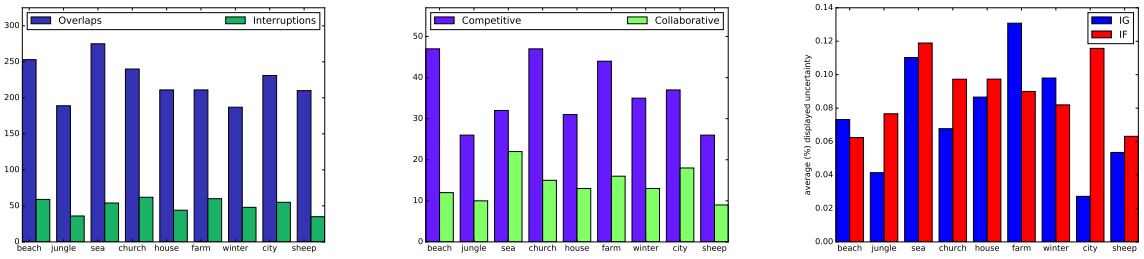
This paper presented the Spot the Difference corpus, a corpus of spontaneous task-oriented spoken dialogues. The set-up used, the experimental procedure and the participants data were described in detail, together with the annotations performed. Parts of data is publicly available online and the complete dataset can be obtain via the first author. It includes a meta-description of the data, audio, video, pictures and respective XML descriptions, the annotations described in this paper and code to parse the data logs.

As we have shown in our preliminary analyses, this data offers various possibilities regarding the study of mechanisms that regulate human communication. For instance, one could look at how humans ground different representation of very similar scenes and how it interacts with turn-taking.

To sum up, we think that this corpus contributes to future research in the multi-modal spoken interaction, so that in the future we can have spoken dialogue systems which are able to dance with their users.

Acknowledgments

The authors would like to thank Anna Hjalmarsson and Catharine Oertel for the discussions on data collection, annotation and analysis, Zofia Malisz for the help to with stats, Cláudia Velhas for transcribing the data, Casey Kennington for the helpful comments and discussions, and reviewers for the sharp suggestions.



(a) Overlaps. (b) Interruptions. (c) Uncertainty.

Figure 5: Overlaps (left), interruptions (center) and uncertainty (right) per scene.

6. Bibliographical References

Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., et al. (1991). The hcrc map task corpus. *Language and speech*, 34(4):351–366.

Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2016). Openface: an open source facial behavior analysis toolkit. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE.

Benus, S., Gravano, A., and Hirschberg, J. (2007). The prosody of backchannels in american english. In *Proceedings of ICPhS*, volume 2007, pages 1065–1068.

Brennan, S. E. and Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6):1482.

Bull, M. and Aylett, M. P. (1998). An analysis of the timing of turn-taking in a corpus of goal-oriented dialogue. In *ICSLP*.

Colman, M. and Healey, P. (2011). The distribution of repair in dialogue. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33.

DeVault, D. (2008). *Contribution tracking: participating in task-oriented dialogue under uncertainty*. Rutgers The State University of New Jersey-New Brunswick.

Friedberg, H., Litman, D., and Paletz, S. B. (2012). Lexical entrainment and success in student engineering groups. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 404–409. IEEE.

Garrod, S. C. and Anderson, A. (1987). Saying what you mean in dialogue: a study in conceptual and semantic co-ordination. *Cognition*, 27:181–218.

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520. IEEE.

Goldberg, J. A. (1990). Interrupting the discourse on interruptions: An analysis in terms of relationally neutral, power-and rapport-oriented acts. *Journal of Pragmatics*, 14(6):883–903.

John, O. P., Naumann, L. P., and Soto, C. J. (2008). Paradigm shift to the integrative big five trait taxonomy. *Handbook of personality: Theory and research*, 3:114–158.

Kennington, C., Kousidis, S., and Schlangen, D. (2013). Interpreting situated dialogue utterances: an update model that uses speech, gaze, and gesture information. In *Proceedings of the SIGDIAL 2013 Conference*, pages 173–182.

Lai, C., Carletta, J., and Renals, S. (2013). Detecting summarization hot spots in meetings using group level involvement and turn-taking features. In *Proc. Interspeech 2013, Lyon, France*.

Liscombe, J., Hirschberg, J., and Venditti, J. J. (2005). Detecting certainness in spoken tutorial dialogues. In *Interspeech 2005*.

Moniz, H., Batista, F., Mata, A. I., and Trancoso, I. (2014). Speaking style effects in the production of disfluencies. *Speech Communication*, 65:20–35.

Nenkova, A., Gravano, A., and Hirschberg, J. (2008). High frequency word entrainment in spoken dialogue. In *Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: Short papers*, pages 169–172. Association for Computational Linguistics.

Oviatt, S., Hang, K., Zhou, J., and Chen, F. (2015). Spoken interruptions signal productive problem solving and domain expertise in mathematics. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 311–318. ACM.

Poesio, M. and Rieser, H. (2010). Completions, coordination, and alignment in dialogue. *Dialogue and Discourse*, 1(1):1–89.

Reitter, D. and Moore, J. D. (2007). Predicting success in dialogue. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, page 808–815, Prague, Czech Republic, June.

Schrank, T. and Schuppler, B. (2015). Automatic detection of uncertainty in spontaneous german dialogue. In *Sixteenth Annual Conference of the International Speech Communication Association (Interspeech)*.

Skantze, G. and Al Moubayed, S. (2012). Iristk: a statechart-based toolkit for multi-party face-to-face interaction. In *Proceedings of ICMI*.

Zarrieß, S., Hough, J., Kennington, C., Manuvinakurike, R., DeVault, D., Fernandez, R., and Schlangen, D. (2016). Pentoref: A corpus of spoken references in task-oriented dialogues. In *10th edition of the Language Resources and Evaluation Conference*.

Appendices

A Scenes



Figure 6: Church scene.

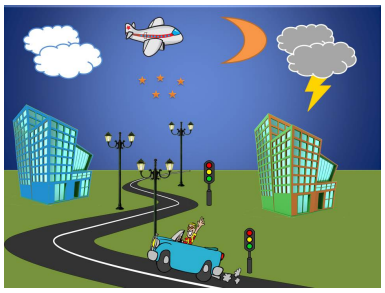


Figure 7: City scene.



Figure 8: Farm scene.

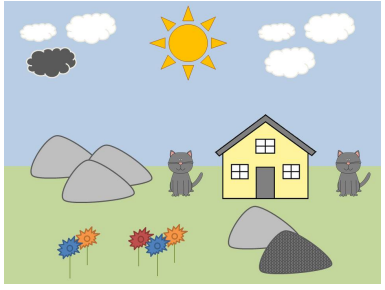


Figure 9: House scene.

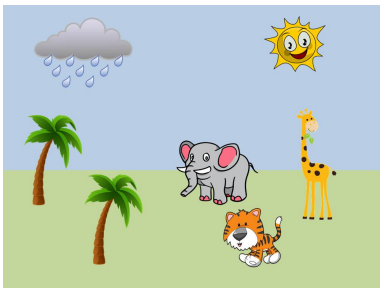


Figure 10: Jungle scene.

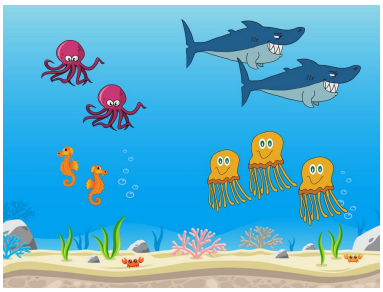


Figure 11: Sea scene.

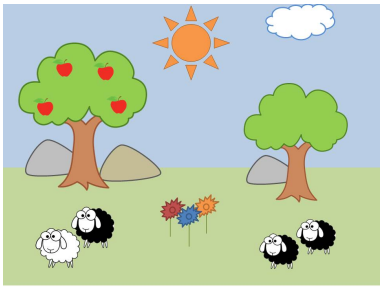


Figure 12: Sheep scene.



Figure 13: Winter scene.

Attention for Implicit Discourse Relation Recognition

Andre Cianflone, Leila Kosseim
Department of Engineering and Computer Science
Concordia University
{a_cianfl, kosseim}@encs.concordia.ca

Abstract

Implicit discourse relation recognition remains a challenging task as state-of-the-art approaches reach F1 scores ranging from 9.95% to 37.67% on the 2016 CoNLL shared task. In our work, we explore the use of a neural network which exploits the strong correlation between pairs of words across two discourse arguments that implicitly signal a discourse relation. We present a novel approach to Implicit Discourse Relation Recognition that uses an encoder-decoder model with attention. Our approach is based on the assumption that a discourse argument is "generated" from a previous argument and conditioned on a latent discourse relation, which we detect. Experiments show that our model achieves an F1 score of 38.25% on fine-grained classification, outperforming previous approaches and performing comparatively with state-of-the-art on coarse-grained classification, while computing alignment parameters without the need for additional pooling and fully connected layers.

Keywords: discourse relation recognition, sequence-to-sequence

1. Introduction

Shallow discourse relation recognition refers to the automatic identification of the relation between two segments of text. For example in:

(1) *I will go to Scotland* after **I complete my studies.**

The underlined discourse connective connects the first discourse argument (in italic) to the second discourse argument (in bold) via a *temporal* relation. Connectives constitute strong signals to identify discourse relations. In fact, given two arguments and a discourse connective many discourse parsers at the 2016 CoNLL Shared Task on Multilingual Shallow Discourse Parsing (SDP) (Xue et al., 2016) were around 78% accurate in recognizing the discourse relation on the SDP blind dataset. On the other hand, in implicit relations no connective is used. This is the case in:

(2) *I need to file my taxes.***Tomorrow is the deadline.**

In (2) the connective *because* is implied and the *contingency* relation is understood by the context. Unfortunately when the connective is absent, identifying the relation automatically becomes much more challenging. At the same 2016 CoNLL SDP shared task, the best implicit discourse relation (IDR) score on the blind test set without connectives reached 37.67% (Xue et al., 2016). In this paper we present a model to automatically recognize implicit discourse relations using an encoder-decoder with attention, a cross-argument word-pair alignment statistic in this context. We show that our model, with an F1 score of 38.25, outperforms other approaches on fine-grained classification, while performing comparatively with the state-of-the-art on coarse-grained classification.

2. Previous Work

Beginning with (Zhang et al., 2015a) and notably in the past year with the CoNLL SDP (Xue et al., 2016), neural network techniques have been used for IDR. Most of these models are based on convolutional neural networks (CNN), inspired by (Zhang et al., 2015a) and other work on sentence classification with CNN (such as (Kim, 2014; Zhang et al., 2015b)). The insight into these many works is that neural networks are better suited at capturing semantic clues between the two arguments of an implicit relation than traditional methods heavily reliant on feature engineering, as in (Pitler et al., 2009; Xue et al., 2015).

Given our correlation assumption, we sought a model that could successfully identify and exploit word pairs across arguments that are strong signals of a discourse relation, leading us to explore attention models. Although several neural network approaches have been proposed for IDR, to our knowledge none have investigated the use of encoder-decoder models with attention, an approach successfully applied to many applications including machine translation (Bahdanau et al., 2015), coreference resolution (Lee et al., 2017) and cloze-style reading comprehension (Cui et al., 2017). To improve translation, notably for longer sentences, a neural translation model is augmented with an attention mechanism uniquely purposed for capturing alignment (Bahdanau et al., 2015). The alignment model scores how well the input words from the source language match output words in the target language. Inspired by recent advances in the use of attention, we used attention to detect alignment scoring for IDR as word-pair features have be shown to contribute to IDR (Pitler et al., 2009; Biran and McKeown, 2013). However, unlike these methods we make no feature engineering. (Rönnqvist et al., 2017) also uses an attention mechanism to recognize implicit discourse relations. However, their approach differs from ours in two important ways: in (Rönnqvist et al., 2017), the two discourse arguments are concatenated to form a single input and the attention mechanism is applied over the entire input, which is fundamentally different to our sequence-to-sequence approach. Furthermore, their work is evaluated on the Chinese Discourse Treebank (Zhou and Xue, 2012).

Top Level	Nb Implicit Instances
Temporal	950
Contingency	4185
Comparison	2832
Expansion	8861
Total	**16828**

Table 1: Top-level breakdown of the PDTB with *entrel* merged into *expansion*

3. Datasets & Tasks

3.1. Datasets

Following the standard in the field, we used both the PDTB and the CoNLL SDP datasets. The PDTB dataset (Rashmi Prasad, 2008) contains 40,600 annotated discourse relations and their arguments over the 1 million word Wall Street Journal (WSJ) corpus (Prasad et al., 2008). The dataset includes four top-level classes of discourse relations; *temporal*, *contingency*, *comparison* and *expansion*; as well as level 2 and lever 3 types. For example, in the PDTB:

(3) *USAir has great promise.***By the second half of 1990, USAir stock could hit 60.**

is labeled as "Contingency.Cause.Reason". A fifth top-level relation, *entrel* (short for *entity-based coherence*), is also defined but has no lower-level types. Table 1 shows statistics of the PDTB dataset.

The CoNLL SDP dataset consists of the full PDTB dataset with a minor reduction in the number of subtypes (Xue et al., 2016). Additionally, the SDP dataset includes a *blind test set*, a second test set created specifically for the 2015 and 2016 editions of the shared task. The blind test set consists of newswire text selected from English Wikinews[1] consistent with WSJ-style text and manually annotated with discourse relations and connectives (Xue et al., 2015).

3.2. Tasks

Given the difficulty of automatic IDR, most work focuses only on top-level classification; i.e. classifying only the four top-level relations with *entrel* merged into *expansion* as preferred by (Pitler et al., 2009; Rutherford and Xue, 2014; Ji and Eisenstein, 2015). The standard WSJ section breakdown is to use sections 2-20 for training, sections 21-22 for testing, and the other sections for development. Given the unbalanced dataset, as shown in Table 1, the task has traditionally been formulated as four binary classifiers. For the development and test sets, the negative samples consist of all other relations. The training set is evenly balanced between positive and negative where negatives samples are randomly drawn from WSJ sections 2 to 20 (excluding positives).

A notable exception to only top-level IDR was the 2015 and 2016 edition of the CoNLL SDP, which included fine-grained non-explicit discourse relation recognition.[2] The fine-grained task is to recognize the 16 low-level subtypes with a single classifier. Additionally, the WSJ section breakdown is different compared to the top-level dataset. The SDP training set consists of WSJ sections 2-21, section 22 for development, and section 23 for testing.

4. Our Model

We describe our model in two modules, the encoder-decoder Recurrent Neural Network (RNN) with attention and two varieties of the classifier.

4.1. Encoder-Decoder RNN with Attention

The standard encoder (Cho et al., 2014) encodes an input vector $\mathbf{x}$, where $\mathbf{x}$ is represented as a sequence of word embedding vectors, into a single context vector $c = q(h_1, \ldots, h_{T_x})$ and hidden state $h_t = f(x_t, h_{t-1})$. Functions f and q are nonlinearities, in our case Bidirectional RNN (Schuster and Paliwal, 1997) of type long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997).

Normally, the decoder predicts a sequence of words y_t where each y_t prediction is conditioned on past predictions and context vector c, maximizing the following joint probability:

$$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \ldots, y_{t-1}\}, c) \quad (1)$$

In the context of RNNs, the conditional probability of each y_t in the joint probability of Eq.1 is modeled as a nonlinear function g with input y_t, context vector c and hidden state s_t:

$$p(y_t | \{y_1, \ldots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c) \quad (2)$$

(Bahdanau et al., 2015) propose a unique context vector c_i for each decoding time step, redefining the decoder conditional probability for each word y_i as:

$$p(y_i | y_1, \ldots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i) \quad (3)$$

The context vector c_i is a weighted sum over all input hidden states $(h_1, \ldots, h_T)$:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \quad (4)$$

where weights $a_{ij} = \text{softmax}(e_{ij})$, $e_{ij} = a(s_{i-1}, h_j)$ and a is a feedforward neural network.

Using attention leads to a vectorized representation of the second argument (decoder output) which is not only informed of its context but also of its alignment, unlike Gated Relevance Networks (GRN) (Chen et al., 2016) where the arguments are not informed of the alignment. In the case of GRN, the two discourse arguments are vectorized with separate RNN layers (no interaction), followed by relevance layers (that compute word-pair interaction), and finally pooling and fully connected layers.

[1] https://en.wikinews.org

[2] Non-explicit discourse includes types *implicit*, *entrel*, and a third *altlex*, short for *alternative lexicalization*. Only a small fraction of the dataset, around 3%, consists of *altlex*. For this reason we will not discuss *altlex* and consider the terms "non-explicit" and "implicit" discourse interchangeably.

4.2. Classifiers

Given our classification task and since the decoder inputs (the second discourse argument words) are known and not predicted, the model is not trained by maximizing the likelihood of the decoder targets, as in Eq.1, but rather by minimizing the cross-entropy error between the predicted label $\hat{y}$ and the true label y for all possible labels l:

$$E(y, \hat{y}) = -\sum_{i=1}^{l} y_i \log(\hat{y}) \quad (5)$$

We experimented with two classifiers to predict $\hat{y}$. In the simplest case:

$$\hat{y} = f(Wh_T + b) \quad (6)$$

where f is the softmax function, h_T is the final decoder hidden state of size d, $W \in \mathbb{R}^{l \times d}$ is a parameter matrix and $b \in \mathbb{R}^l$ a bias vector. In this case the classifier only relies on the last hidden state, minimizing the total number of parameters at the expense of information loss. We denote this Classifier with Attention CA, shown in Figure 1.
In the second classifier, $\hat{y}$ is a function of:

$$p = \max_{t=1}^{T}(h_{dec_1}, \ldots, h_{dec_t}) \quad (7)$$
$$h = g(W_d p + b_d) \quad (8)$$
$$\hat{y} = f(W_s h + b_s) \quad (9)$$

where p is a T sized concatenated vector of the maximum values over each decoder hidden state h_{dec}, i.e. 1D max pooling. $W_d \in \mathbb{R}^{v \times T}$ and $Ws \in \mathbb{R}^{l \times v}$ are parameter matrices, $b \in \mathbb{R}^v$ and $b \in \mathbb{R}^l$ are bias terms, and g a nonlinearity. In this case each decoded time step informs the relation classification. We denote this Classifier from Sequence with Attention CSA, as shown in Figure 2

5. Experiments

In this section we outline our data preprocessing and experiments. The raw texts from the PDTB and the CoNLL SDP are converted to lower case and tokenized. Then we keep only the 10,000 most common words. After forming a dictionary of unique tokens, we substitute each token with a dense word embedding from a pretrained model. Following the preferred embeddings used at the 2016 CoNLL SDP (Xue et al., 2016), we used the 300 dimensional pretrained Word2Vec binaries[3], trained by continuous skip-gram (Mikolov et al., 2013) for both top-level and fine-grained classification. While the PDTB samples contain additional data such as part-of-speech tags and parse trees, no additional data is used.
The top-level classification consists of four separately trained binary classifiers, while we train a single classifier for the fine-grained classification. We experiment using LSTM and GRU (Cho et al., 2014) cells, opting for LSTM since it showed slightly better results. The number of cell parameters were randomly searched at each training run. We randomly switched between bidirectional encoder or single direction. For the CSA, we additionally performed hyper-parameter search on the number of hidden units. Our main parameters that produced the best performance are listed in Table 2. Our models were optimized with the Adam algorithm (Kingma and Ba, 2015). Models evaluated on the test sets are based on optimal validation set F1 score.

[3]https://code.google.com/archive/p/word2vec/

Model		Parameter	Value
CSA	CA	batch size	32
		embedding size	300
		cell type	LSTM
		cell units	100
		pooling	1D max
		dense layer units	60

Table 2: Architecture parameters. Dense layer refers to the CSA model's fully connected layer between pooling and softmax layers.

ID	Author	Blind	Test	Dev
ecnucs	Wang	34.18	40.91	46.40
tbmihaylov	Mihaylov	34.51	39.19	40.32
tao0920	Qin	35.38	38.20	46.33
gtnlp	n/a	36.75	34.95	40.72
ttr	Rutherford	37.67	36.13	40.32
CSA	ours	35.07	28.05	36.58
CA	ours	**38.25**	35.63	39.42

Table 3: F1 scores of fine-grained IDR compared to top 5 teams. (Wang and Lan, 2016; Mihaylov and Frank, 2016; Qin et al., 2016; Rutherford and Xue, 2016)

6. Results & Analysis

Given the unbalanced datasets, performance is evaluated solely on F1 scores. Table 3, summarizes our top-level classification results on the PDTB dataset in comparison with other authors and Table 4 our fine-grained classification results[4] on the CoNLL SDP dataset.
As shown in Table 3, our CA model scored 38.25% on the fine-grained classification, over state-of-the-art F1 score of 37.67%. Observing the blind test set results in Table 3 we note how our model generalizes well to a different dataset (Wikinews). Other top models such as "gtnlp" and "ecnucs" have a more than 10 point difference between the development score and blind test score compared to 2 points in the CA case.
For the top-level classification, our CA model (see Table 4) scored well in the case of *expansion* with 80.72% F1 score, the largest relation class, and *contingency*, while*temporal* was better than most other approaches. The F1 of 30.56% for *comparison* was far from the top result in Table 4, likely due to the small dataset size.
It is interesting to note that the results achieved by the CA model are based on a relatively shallow, single bidirectional RNN encoder layer and single RNN decoder layer with attention. It is possible that the chosen input embedding had a minor impact on our results. We would have liked to measure the embedding effect to compare with (Chen et al., 2016), but to our knowledge the embedding is not publicly available.

[4]We used the official CoNLL scorer for comparison: https://github.com/attapol/conll16st

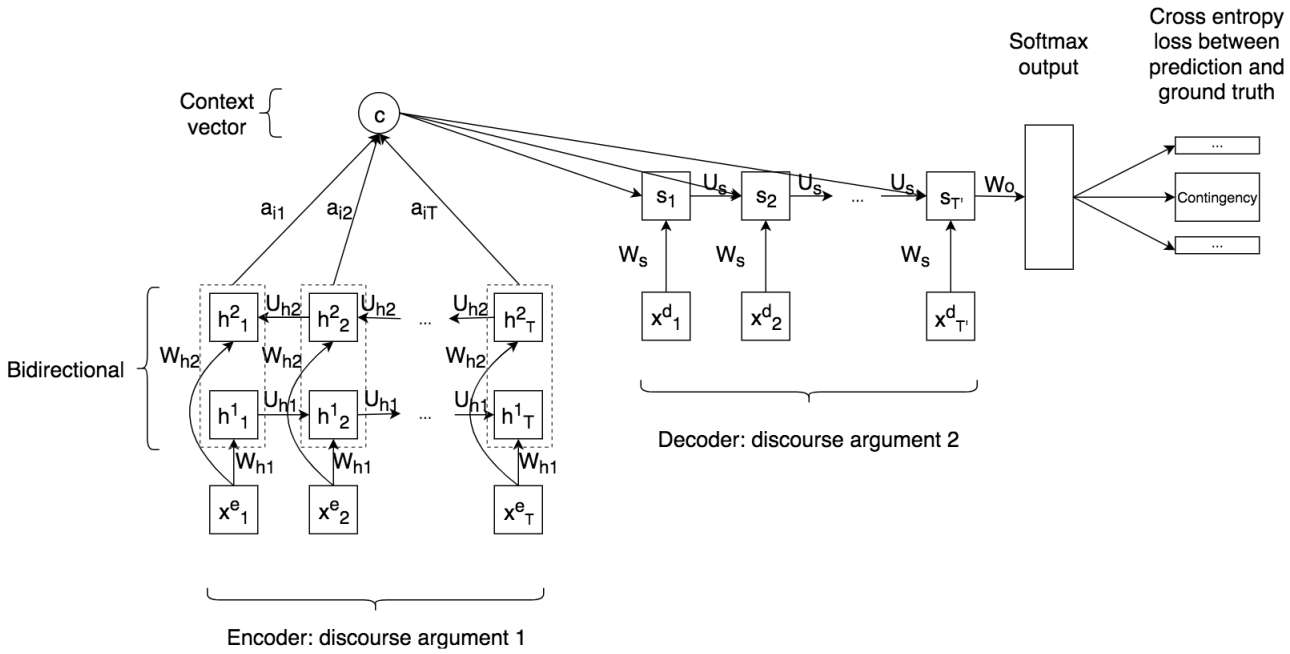


Figure 1: Our classifier with attention (CA): an encoder-decoder recurrent neural network with attention with the last hidden state used for classification. In the doted rectangles, the forward and backward hidden states are concatenated. Note there is no backpropagation through time from output predictions at each time step. Only the final cross-entropy error is backpropagated through time.

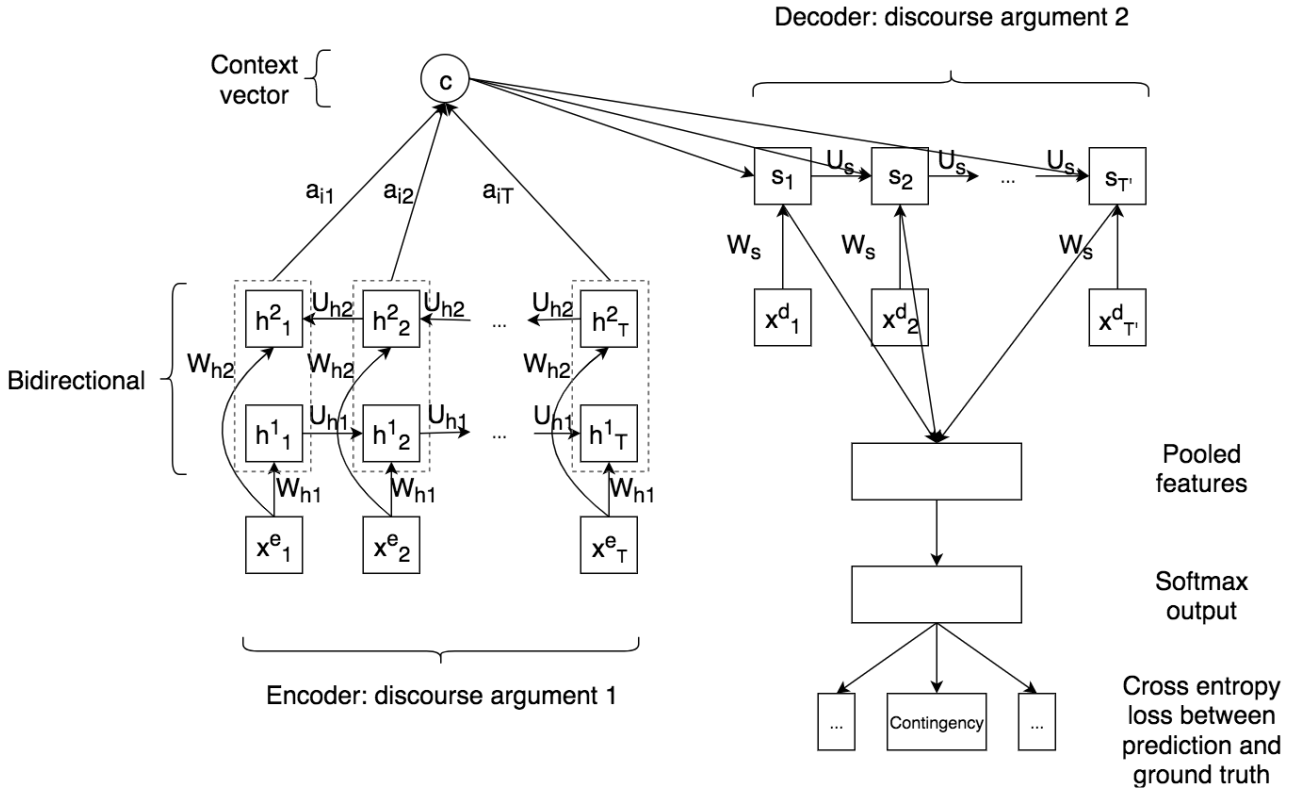


Figure 2: Our classifier with sequence of attention (CSA): encoder-decoder recurrent neural network with attention. The decoder hidden states are used for classification. Note that there is no backpropagation through time from output predictions at each time step.

Author	Comp.	Cont.	Exp.	Temp.
Pitler	21.96	47.13	76.42	16.76
Zhou	31.79	47.16	70.11	20.30
Park	31.32	49.82	79.22	26.57
Rutherford	39.70	54.42	80.44	28.69
Ji	35.93	52.78	80.02	27.63
Chen	**40.17**	54.76	80.62	**31.32**
CSA	27.02	49.86	77.45	24.43
CA	30.56	**54.80**	**80.72**	27.15

Table 4: F1 scores of top-level IDR for: *comparison, contingency, expansion, temporal*. Note that *entrel* is merged into *expansion*, as done in previous works. (Pitler et al., 2009; Zhou et al., 2010; Park and Cardie, 2012; Rutherford and Xue, 2014; Ji and Eisenstein, 2015; Chen et al., 2016)

We were surprised by the CSA's lower performance in all cases. We believed the model would be more robust if the classification layer had inputs from all decoded hidden states directly. However, using only the final state vector resulted in higher classification score while using less parameters. This may be due to overfitting. We would need to reevaluate the model on a larger dataset.

7. Conclusion

We presented an efficient encoder-decoder model with attention for implicit discourse relation recognition. Our model computes attention between discourse argument word pairs without feature engineering and without the need for additional fully connected layers, minimizing the number of trainable parameters. Finally, we show that our model generalizes well to unseen datasets on fine-grained classification, outperforming state-of-the-art without large variance in scoring between development and test sets, and outperforms in two categories in the coarse-grained case. In future work we would like to explore in more detail automatically learned alignment for IDR and text generation based on these models.

Acknowledgement

The authors would like to thank the anonymous reviewers for their feedback on the paper. We gratefully acknowledge the financial support of NSERC and the NVIDIA Corporation for the donation of the Titan X Pascal GPU used for this research.

8. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Advances in Neural Information Processing Systems (NIPS 2015)*, pages 649–657, Montreal, Canada, December.

Biran, O. and McKeown, K. (2013). Aggregated word pair features for implicit discourse relation disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 69–73, Sofia, Bulgaria.

Chen, J., Zhang, Q., Liu, P., Qiu, X., and Huang, X. (2016). Implicit discourse relation detection via a deep architecture with gated relevance network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL-2016)*, pages 1726–1735.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. pages 1724â–1734, Doha, Qatar.

Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., and Hu, G. (2017). Attention-over-attention neural networks for reading comprehension. pages 593–602.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Ji, Y. and Eisenstein, J. (2015). One vector is not enough: Entity-augmented distributional semantics for discourse relations. *Transactions of the Association for Computational Linguistics*, pages 329–344, June.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kingma, D. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceeding of the 2015 International Conference on Learning Representation (ICLR 2015)*, San Diego, California.

Lee, K., He, L., Lewis, M., and Zettlemoyer, L. (2017). End-to-end neural coreference resolution.

Mihaylov, T. and Frank, A. (2016). Discourse relation sense classification using cross-argument semantic similarity based on word embeddings. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 100–107, Berlin, Germany.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems (NIPS 2013)*, pages 3111–3119, Lake Tahoe, USA, December.

Park, J. and Cardie, C. (2012). Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 108–112. Association for Computational Linguistics.

Pitler, E., Louis, A., and Nenkova, A. (2009). Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processinf (ACL/IJCNLP). Volume 2*, pages 683–691. Association for Computational Linguistics.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A. K., and Webber, B. L. (2008). The Penn Discourse TreeBank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

Qin, L., Zhang, Z., and Zhao, H. (2016). Shallow discourse parsing using convolutional neural network. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 70–77, Berlin, Germany.

Rönnqvist, S., Schenk, N., and Chiarcos, C. (2017). A re-

current neural model with attention for the recognition of chinese implicit discourse relations. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 256–262.

Rutherford, A. and Xue, N. (2014). Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 645–654, Gothenburg, Sweden, April. Association for Computational Linguistics.

Rutherford, A. and Xue, N. (2016). Robust non-explicit neural discourse parser in english and chinese. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 55–59, Berlin, Germany.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Wang, J. and Lan, M. (2016). Two end-to-end shallow discourse parsers for english and chinese in conll-2016 shared task. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 33–40, Berlin, Germany.

Xue, N., Ng, H. T., Pradhan, S., Bryant, R. P. C., and Rutherford, A. T. (2015). The conll-2015 shared task on shallow discourse parsing. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 1–16, Beijing, China.

Xue, N., Ng, H. T., Rutherford, A., Webber, B., Wang, C., and Wang, H. (2016). Conll 2016 shared task on multilingual shallow discourse parsing. *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task (CoNLL)*, pages 1–19.

Zhang, B., Su, J., Xiong, D., Lu, Y., Duan, H., and Yao, J. (2015a). Shallow convolutional neural network for implicit discourse relation recognition. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2230–2235, Lisbon, Portugal, September. Association for Computational Linguistics.

Zhang, X., Zhao, J., and LeCun, Y. (2015b). Character-level convolutional networks for text classification. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS 2015)*, pages 649–657, Montreal, Canada, December.

Zhou, Y. and Xue, N. (2012). Pdtb-style discourse annotation of chinese text. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: (ACL)*, pages 69–77. Association for Computational Linguistics.

Zhou, Z.-M., Xu, Y., Niu, Z.-Y., Lan, M., Su, J., and Tan, C. L. (2010). Predicting discourse connectives for implicit discourse relation recognition. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1507–1514. Association for Computational Linguistics.

Language Resource References

Rashmi Prasad, Alan Lee, Nikhil Dinesh, Eleni Miltsakaki, Geraud Campion, Aravind Joshi, Bonnie Webber. (2008). *Penn Discourse Treebank Version 2.0*. Linguistic Data Consortium, 2.0, ISLRN 488-589-036-315-2.

A Context-based Approach for Dialogue Act Recognition using Simple Recurrent Neural Networks

Chandrakant Bothe, Cornelius Weber, Sven Magg, and Stefan Wermter
Knowledge Technology, Department of Informatics,
University of Hamburg,
Vogt-Koelln-Str. 30, 22527 Hamburg, Germany
http://www.informatik.uni-hamburg.de/WTM/
{bothe,weber,magg,wermter}@informatik.uni-hamburg.de

Abstract

Dialogue act recognition is an important part of natural language understanding. We investigate the way dialogue act corpora are annotated and the learning approaches used so far. We find that the dialogue act is context-sensitive within the conversation for most of the classes. Nevertheless, previous models of dialogue act classification work on the utterance-level and only very few consider context. We propose a novel context-based learning method to classify dialogue acts using a character-level language model utterance representation, and we notice significant improvement. We evaluate this method on the Switchboard Dialogue Act corpus, and our results show that the consideration of the preceding utterances as a context of the current utterance improves dialogue act detection.

Keywords: Dialogue Acts Detection, Recurrent Neural Networks, Context-based Learning

1. Introduction

In natural language processing research, the dialogue act (DA) concept plays an important role. Its recognition, in most cases, is considered a lexical-based or syntax-based classification at utterance-level. However, the discourse compositionality is context sensitive, meaning that the DA of an utterance can be elicited from the preceding utterances (Grosz, 1982). Hence, classifying only utterances is not enough because their DA class arises from their context. For example, the utterance containing only the lexical entry *'yeah'* might appear in several DA classes such as *Backchannel*, *Yes-Answer*, etc. For certain DA classes, the utterances are short, and most of them share similar lexical and syntactic cues (Jurafsky et al., 1998).

The aim of this article has two subgoals: first, we investigate the annotation process of DA corpora and review the modelling so far used for DA classification, and second, we present a novel model and compare its results with the state of the art. We propose to use context-based learning for the identification of the DA classes. First, we show the results without context, i.e., classifying only utterances. Including context leads to 3% higher accuracy. We use a simple recurrent neural network (RNN) for context learning of the discourse compositionality. We feed the preceding and current utterances to the RNN model to predict its DA class. The main contributions of this work are as follows:

- We provide detailed insight on the annotation and modelling of dialogue act corpora. We suggest to model discourse within the context of a conversation.
- We propose a context-based learning approach for DA identification. In our approach, we represent utterances by a character-level language model trained on domain-independent data.
- We evaluate the model on the Switchboard Dialogue Act (SwDA[1]) corpus and show how using context affects the results. For the SwDA corpus, our model achieved an accuracy of 77.3% compared to 73.9% as state of the art, where the context-based learning is used for the DA classification (Kalchbrenner and Blunsom, 2013).
- Benefits of using context arise from using only a few preceding utterances making the model suitable for dialogue system in real time, in contrast to feeding the whole conversation, which can achieve high accuracy, but includes future utterances (Liu et al., 2017; Kumar et al., 2017).

[1]Available at https://github.com/cgpotts/swda

2. Related Work

2.1. Annotation of Dialogue Act Corpora

Annotation Process and Standards: Research on dialogue acts became important with the commercial reality of spoken dialogue systems. There have been many taxonomies to it: speech act (Austin, 1962) which was later modified into five classes (Assertive, Directive, Commissive, Expressive, Declarative) (Searle, 1979), and the Dialogue Act Markup in Several Layers (DAMSL) tag set where each DA has a forward-looking function (such as Statement, Info-request, Thanking) and a backward-looking function (such as Accept, Reject, Answer) (Allen and Core, 1997). There are many such standard taxonomies and schemes to annotate conversational data, some of them follow the concept of discourse compositionality. These schemes are important for analysing dialogues or building a dialogue system (Skantze, 2007). However, there can never be a unique scheme that considers all aspects of dialogue.

Corpus Insight: We have investigated the annotation method for two corpora: Switchboard (SWBD) (Godfrey et al., 1992; Jurafsky et al., 1997) and ICSI Meeting Recorder Dialogue Act (MRDA) (Shriberg et al., 2004). They are annotated with the DAMSL tag set. The annotation includes not only the utterance-level but also the segmented-utterance labelling. The DAMSL tag set provides very fine-grained and detailed DA classes and follows the discourse compositionality. For example, the SWBD-DAMSL is the variant of DAMSL specific to the Switchboard cor-

Table 1: Example of a labeled conversation (portions) from the Switchboard Dialogue Act corpus

Speaker	Dialogue Act	Utterance
A	Backchannel	Uh-huh.
B	Statement	About twelve foot in diameter
B	Abandoned	and, there is a lot of pressure to get that much weight up in the air.
A	Backchannel	Oh, yeah.
B	Abandoned	So it's interesting, though.
		...
B	Statement-opinion	it's a very complex, uh, situation to go into space.
A	Agree/Accept	Oh, yeah,
		...
A	Yes-No Question	You never think about that do you?
B	Yes-Answer	Yeah.
A	Statement-opinion	I would think it would be harder to get up than it would be
B	Backchannel	Yeah.

pus. It distinguishes *wh-questions (qw)*, *yes-no questions (qy)*, *open-ended (qo)*, and *or-questions (qr)* classes, not just because these questions are syntactically distinct, but also because they have different forward functions (Jurafsky, 1997). A *yes-no question* is more likely to get a *"yes"* answer than a *wh-question*. This also gives an intuition that the answers follow the syntactic formulation of question which provides a context. For example *qy* is used for a question that from a discourse perspective expects a *Yes* or *No* answer.

Nature of Discourse in Conversation: The dialogue act is a context-based discourse concept that means the DA class of a current utterance can be derived from its preceding utterance. We will elaborate this argument with an example given in Table 1. Speaker *A* utters *'Oh, yeah.'* twice in the first portion, and each time it is labelled with two different DA labels. This is simply due to the context of the previously conversed utterances. If we see the last four utterances of the example, when speaker *A* utters the *'Yes-No Question'* DA, speaker *B* answers with *'yeah'* which is labelled as *'Yes-Answer'* DA. However, after the *'Statement-opinion'* from the same speaker, the same utterance *'yeah'* is labelled as *'Backchannel'* and not *'Yes-Answer'*. This gives evidence that when we process the text of a conversation, we can see the context of a current utterance in the preceding utterances.

Prosodic Cues for DA Recognition: It has also been noted that prosodic knowledge plays a major role in DA identification for certain DA types (Jurafsky et al., 1998; Stolcke et al., 2000). The main reason is that the acoustic signal of the same utterance can be very different in a different DA class. This indicates that if one wants to classify DA classes only from the text, the context must be an important aspect to consider: simply classifying single utterances might not be enough, but considering the preceding utterances as a context is important.

2.2. Modelling Approaches

Lexical, Prosodic, and Syntactic Cues: Many studies have been carried out to find out the lexical, prosodic and syntactic cues (Stolcke et al., 2000; Surendran and Levow, 2006; O'Shea et al., 2012; Yang et al., 2014). For the SwDA corpus, the state-of-the-art baseline result was 71% for more than a decade using a standard Hidden Markov Model (HMM) with language features such as words and n-grams (Stolcke et al., 2000). The inter-annotator agreement accuracy for the same corpus is 84%, and in this particular case, we are still far from achieving human accuracy. However, words like *'yeah'* appear in many classes such as *backchannel*, *yes-answer*, *agree/accept* etc. Here, the prosodic cues play a very important role in identifying the DA classes, as the same utterance can acoustically differ a lot which helps to distinguish the specific DA class (Shriberg et al., 1998). There are several approaches like traditional Naive Bayes and HMM models, which use minimal information and certainly ignore the dependency of the context within the communication (Grau et al., 2004; Tavafi et al., 2013). They achieved 66% and 74.32% respectively on the SwDA test set.

Utterance-level Classification: Perhaps most research in modelling dialogue act identification is conducted at utterance-level (Stolcke et al., 2000; Grau et al., 2004; Tavafi et al., 2013; Ji et al., ; Khanpour et al., 2016; Lee and Dernoncourt, 2016). The emergence of deep learning also gave a big push to DA classification. In a natural language conversation, most utterances are very short; hence it is also referred to as short text classification. Lee and Dernoncourt (2016) achieved 73.1% accuracy on the SwDA corpus by using advanced deep learning frameworks such as RNNs and convolutional neural networks (CNN) with word-level feature embeddings.

A Novel Approach: Context-based Learning: Classifying the DA classes at single utterance-level might fail when it comes to DA classes where the utterances share similar lexical and syntactic cues (words and phrases) like the *backchannel*, *yes-answer* and *accept/agree* classes. Some researchers proposed an utterance-dependent learning approach (Kalchbrenner and Blunsom, 2013; Ji et al., ; Kumar et al., 2017; Tran et al., 2017; Liu et al., 2017; Ortega and Vu, 2017; Meng et al., 2017). Kalchbrenner and Blunsom (2013) and Ortega and Vu (2017) have proposed context-based learning, where they represent the utterance as a compressed vector of the word embeddings using CNNs and use these utterance representations to model discourse within a conversation using RNNs. In their architecture,

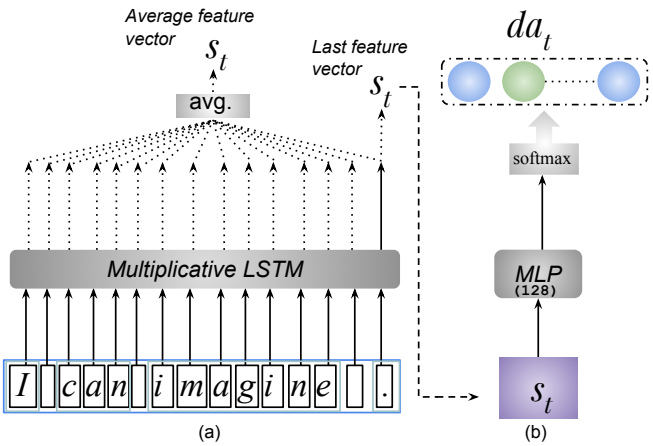


Figure 1: (a) Multiplicative LSTM (mLSTM) character-level language model to produce the sentence representation s_t. The character-level language model is pre-trained and produces the feature (hidden unit states of mLSTM at the last character) or average (average of all hidden unit states of every character) vector representation of the given utterance. (b) Utterance-level classification using a simple MLP layer with a *softmax* function (our baseline model).

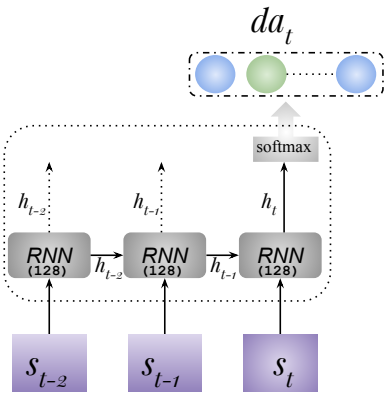


Figure 2: The RNN setup for learning the dialogue act recognition with the previous sentences as context. s_t is an utterance representation derived with a character-level language model and has a dialogue act label da_t. s_{t-1} and s_{t-2} are the preceding utterances of s_t. The RNN is trained to learn the recurrency through previous utterances s_{t-1} and s_{t-2} derived as h_{t-1} and h_{t-2} as a context to recognize the dialogue act of current utterance s_t which is represented by h_t used to detect da_t.

they also give importance to turn-taking by providing the speaker identity but do not analyse their model in this regard. This approach achieves about 73.9% accuracy on the SwDA corpus. In another line of research (Ji et al., ; Kumar et al., 2017), authors claim that their models take care of the dependency of the utterances within a conversation. Ji et al. (2016) use discourse annotation for the word-level language modelling on the SwDA corpus and also highlight a limitation that this approach is not scalable to large data.

In other approaches a hierarchical convolutional and recurrent neural encoder model are used to learn utterance representation by feeding a whole conversation (Kumar et al., 2017; Liu et al., 2017). The utterance representations are further used to classify DA classes using the conditional random field (CRF) as a linear classifier. The model can see the past and future utterances at the same time within a conversation, which limits usage in a dialogue system where one can only perceive the preceding utterance as a context but does not know the upcoming utterances. Hence, we use a context-based learning approach and regard the 73.9% accuracy (Kalchbrenner and Blunsom, 2013) on the SwDA corpus as a current state of the art for this task.

3. Our Approach

Our approach takes care of discourse compositionality while recognising dialogue acts. The DA class of the current utterance is predicted using the context of the preceding utterances. We represent each utterance by the hidden state of the multiplicative recurrent neural network trained on domain-independent data using a character-level language model. We use RNNs to feed the sequence of the utterances and eventually predict the DA class of the corresponding utterance.

3.1. Utterance Representation

Character-level encoding allows processing words and whole sentences based on their smallest units and still capturing punctuation and permutation of words. We represent a character-level utterance by encoding the whole sentence with a pre-trained character language model[2]. This model consists of a single multiplicative long-short-term memory (mLSTM) network (Krause et al., 2016) layer with 4,096 hidden units. The mLSTM is composed of an LSTM and a multiplicative RNN and considers each possible input in a recurrent transition function. It is trained as a character language model on ∼80 million Amazon product reviews (Radford et al., 2017). We sequentially input the characters of an utterance to the mLSTM and take the hidden state values after the last character as shown in Figure 1 (a).

The hidden vector s_t obtained after the last character is called the last feature vector, as it stores the information related to the character language model and the sentiment of the utterance. However, it was shown that the average vector over all characters in the utterance works better for emotion detection (Lakomkin et al., 2017). Hence, we extract the last feature vector and also the average feature vector representations for each utterance. We classify these representations with a multi-layer perceptron (MLP) as shown in Figure 1 (b). The results are shown in Table 2. The standard deviation (SD) is computed over ten runs. The average vector seems to carry more information related to the DA; hence we use it for future experiments. There is an advantage of using domain-independent data: it is rich regarding features being trained on big data, perhaps surpassing the limitation of scalability as mentioned in Ji et al. (2016).

[2] https://github.com/openai/generating-reviews-discovering-sentiment

Table 2: Accuracy of the dialogue act identification using the character-level language model utterance representation for 42 classes using a single MLP layer with 64 neurons.

Model input	Acc.(%)	SD
Last feature vector	71.48	0.28
Average feature vector	**73.96**	0.26
Concatenated vector	73.18	0.31

3.2. Context Learning with RNNs

We apply context learning with the help of RNNs. As shown in Figure 2, the utterances with their character-level language model representation s_t are fed to the RNN with the preceding utterances (s_{t-1}, s_{t-2}) being the context. We use the RNN, which gets the input s_t, and stores the hidden vector h_t at time t (Elman, 1990), which is calculated as:

$$h_t = f\left(W_h * h_{t-1} + I * s_t + b\right) \quad (1)$$

where $f()$ is a sigmoid function, W_h and I are recurrent and input weight matrices respectively and b is a bias vector learned during training. h_t is computed using the previous hidden vector h_{t-1} which is computed in a same way for preceding utterance s_{t-1}. The output da_t is the dialogue act label of the current utterance s_t calculated using h_t, as:

$$da_t = g\left(W_{out} * h_t\right) \quad (2)$$

where W_{out} is the output weight matrix. The weight matrices are learned using back-propagation through time. The task is to classify several classes; hence we use a $softmax$ function $g()$ on the output. The input is the sequence of the current and preceding utterances, e.g., s_t, s_{t-1}, and s_{t-2}. We reset the RNN when it sees the current utterance s_t. We also give the information related to a speaker to let the network find the change in the speaker's turn. The speaker id 'A' is represented by [1,0] and id 'B' by [0,1] and it is concatenated with the corresponding utterances s_t.

The Adam optimiser (Kingma and Ba, 2014) was used with a learning rate $1e-4$, which decays to zero during training, and clipping gradients at norm 1. Early stopping was used to avoid over-fitting of the network, 20% of training samples were used for validation. In all learning cases, we minimise the categorical cross-entropy.

3.3. Results

We follow the same data split of 1115 training and 19 test conversations as in the baseline approach (Stolcke et al., 2000; Kalchbrenner and Blunsom, 2013). Table 3 shows the results of the proposed model with several setups, first without the context, then with one, two, and so on preceding utterances in the context. We examined different values for the number of the hidden units of the RNN, empirically 64 was identified as best and used throughout the experiments. We also experimented with the various representations for the speaker id that is concatenated with the respective utterances but could find no differences. As a result, our proposed model uses minimal information for the context. The performance increases from 74% to about 77% with context. We run each experiment for ten times and take the average. The model shows robustness providing minimal variance, and using a minimum number of preceding utterances as a context can produce fair results.

Table 3: Accuracy of the dialogue act identification with the context-learning approach.

Model setup	Acc.(%)	SD
Baseline		
Most common class	31.50	
Related previous work		
Stolcke et al. (2000)	71.00	
Kalchbrenner and Blunsom (2013)	73.90	
Our work		
Our baseline (without context)	73.96	0.26
RNN (1 utt. in context w. SpeakerID)	76.48	0.33
RNN (1 utt. in context)	76.57	0.28
RNN (2 utts. in context)	76.81	0.24
RNN (3 utts. in context)	**77.34**	0.21
RNN (4 utts. in context)	77.28	0.22

4. Conclusion

In this article, we detail the annotation and modelling of dialogue act corpora, and we find that there is a difference in the way DAs are annotated and the way they are modelled. We argue to generalise the discourse modelling for conversation within the context of communication. Hence, we propose to use the context-based learning approach for the DA identification task. We used simple RNN to model the context of preceding utterances. We used the domain-independent pre-trained character language model to represent the utterances. We evaluated the proposed model on the Switchboard Dialogue Act corpus and show the results with and without context. For this corpus, our model achieved an accuracy of 77.34% with context compared to 73.96% without context. We also compare our model with Kalchbrenner and Blunsom (2013) who used the context-based learning approach achieving 73.9%. Our model uses minimal information, such as the context of a few preceding utterances which can be adapted to an online learning tool such as a spoken dialogue system where one can naturally see the preceding utterances but not the future ones. This makes our model suitable for human-robot/computer interaction which can be easily plugged into any spoken dialogue system.

5. Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement number 642667 (SECURE).

6. Bibliographical References

Allen, J. and Core, M. (1997). Draft of DAMSL: Dialogue Act Markup in Several Layers.

Austin, J. L. (1962). *How to Do Things with Words*. Oxford University Press.

Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. In *Proc. of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520.

Grau, S., Sanchis, E., Castro, M. J., and Vilar, D. (2004). Dialogue act classification using a Bayesian approach. In *9th Conference Speech and Computer SPECOM 2004*.

Grosz, B. J. (1982). Discourse Analysis. *Sublanguage. Studies of Language in Restricted Semantic Domains*, pages 138–174.

Ji, Y., Haffari, G., and Eisenstein, J.). A Latent Variable Recurrent Neural Network for Discourse Relation Language Models. In *Proc. of NAACL-HLT*, pages 332–342.

Jurafsky, D., Shriberg, E., and Biasca, D. (1997). Switchboard Dialog Act Corpus. Technical report, International Computer Science Inst. Berkeley CA.

Jurafsky, D., Shribergy, E., Fox, B., and Curl, T. (1998). Lexical, Prosodic, and Syntactic Cues for Dialog Acts. In *The ACL/COLING Workshop on Discourse Relations and Discourse Markers*.

Jurafsky, D. (1997). Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual, draft 13. *Technical Report 97-01, University of Colorado Institute of Cognitive Science*, pages 225–233.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent Convolutional Neural Networks for Discourse Compositionality. In *Workshop on Continuous Vector Space Models and their Compositionality, ACL*, pages 119–126.

Khanpour, H., Guntakandla, N., and Nielsen, R. (2016). Dialogue Act Classification in Domain-Independent Conversations Using a Deep Recurrent Neural Network. In *COLING*, pages 2012–2021.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv: 1412.6980*.

Krause, B., Lu, L., Murray, I., and Renals, S. (2016). Multiplicative LSTM for sequence modelling. *Workshop track - ICLR 2016*.

Kumar, H., Agarwal, A., Dasgupta, R., Joshi, S., and Kumar, A. (2017). Dialogue Act Sequence Labeling using Hierarchical encoder with CRF. *arXiv:1709.04250v2*.

Lakomkin, E., Bothe, C., and Wermter, S. (2017). GradAscent at EmoInt-2017: Character and Word Level Recurrent Neural Network Models for Tweet Emotion Intensity Detection. In *Proc. of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis at EMNLP 2017*, pages 169–174. ACL.

Lee, J. Y. and Dernoncourt, F. (2016). Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks. *arXiv preprint arXiv:1603.03827*.

Liu, Y., Han, K., Tan, Z., and Lei, Y. (2017). Using Context Information for Dialog Act Classification in DNN Framework. In *Proc. of the 2017 Conference on EMNLP*, pages 2160–2168. ACL.

Meng, Z., Mou, L., and Jin, Z. (2017). Hierarchical RNN with Static Sentence-Level Attention for Text-Based Speaker Change Detection. In *Proc. of ACM Conference on Information and Knowledge Management*, pages 2203–2206.

Ortega, D. and Vu, N. T. (2017). Neural-based Context Representation Learning for Dialog Act Classification. *Proc. of the SIGDIAL 2017 Conference*, pages 247–252.

O'Shea, J., Bandar, Z., and Crockett, K. (2012). A Multi-classifier Approach to Dialogue Act Classification Using Function Words. In *Transactions on Computational Collective Intelligence VII*, pages 119–143. Springer.

Radford, A., Jozefowicz, R., and Sutskever, I. (2017). Learning to Generate Reviews and Discovering Sentiment. *arXiv: 1704.01444*.

Searle, J. R. (1979). *Expression and Meaning: Studies in the Theory of Speech Acts*. Cambridge University Press.

Shriberg, E., Stolcke, A., Jurafsky, D., Coccaro, N., Meteer, M., Bates, R., Taylor, P., Ries, K., Martin, R., and Van Ess-Dykema, C. (1998). Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech? *Language and Speech*, 41(3-4):443–492.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The ICSI meeting recorder dialog act (MRDA) corpus. Technical report, International Computer Science Inst. Berkeley CA.

Skantze, G. (2007). Error Handling in Spoken Dialogue Systems-Managing Uncertainty, Grounding and Miscommunication: Chapter 2, Spoken Dialogue Systems. *KTH Computer Science and Communication*.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-Dykema, C., and Meteer, M. (2000). Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics*, 26(3):339–373.

Surendran, D. and Levow, G.-A. (2006). Dialog act tagging with support vector machines and hidden Markov models. In *Interspeech – ICSLP*.

Tavafi, M., Mehdad, Y., Joty, S. R., Carenini, G., and Ng, R. T. (2013). Dialogue Act Recognition in Synchronous and Asynchronous Conversations. In *SIGDIAL Conference*, pages 117–121. ACL.

Tran, Q. H., Zukerman, I., and Haffari, G. (2017). Preserving Distributional Information in Dialogue Act Classification. In *Proc. of Conference on EMNLP*, pages 2141–2146. ACL.

Yang, X., Liu, J., Chen, Z., and Wu, W. (2014). Semi-supervised Learning of Dialogue Acts Using Sentence Similarity Based on Word Embeddings. In *Proc. of International Conference on Audio, Language and Image Processing*, pages 882–886.

Appendix: Analysis of the state of the RNN

We also analyze the internal state h_t of the RNNs for a two-utterance setup. We plot them on a 2D graph with the t-SNE algorithm for the first 2,000 utterances of the SwDA test set. Figure 3 shows the clusters of all the DA classes. The classes which do not share any information are grouped without any interference such as *Non-verbal*, and *Abandoned*. Figure 4 shows some particular classes with utterances in their vector spaces, the (1) current utterance and (2) a preceding utterance in the context.

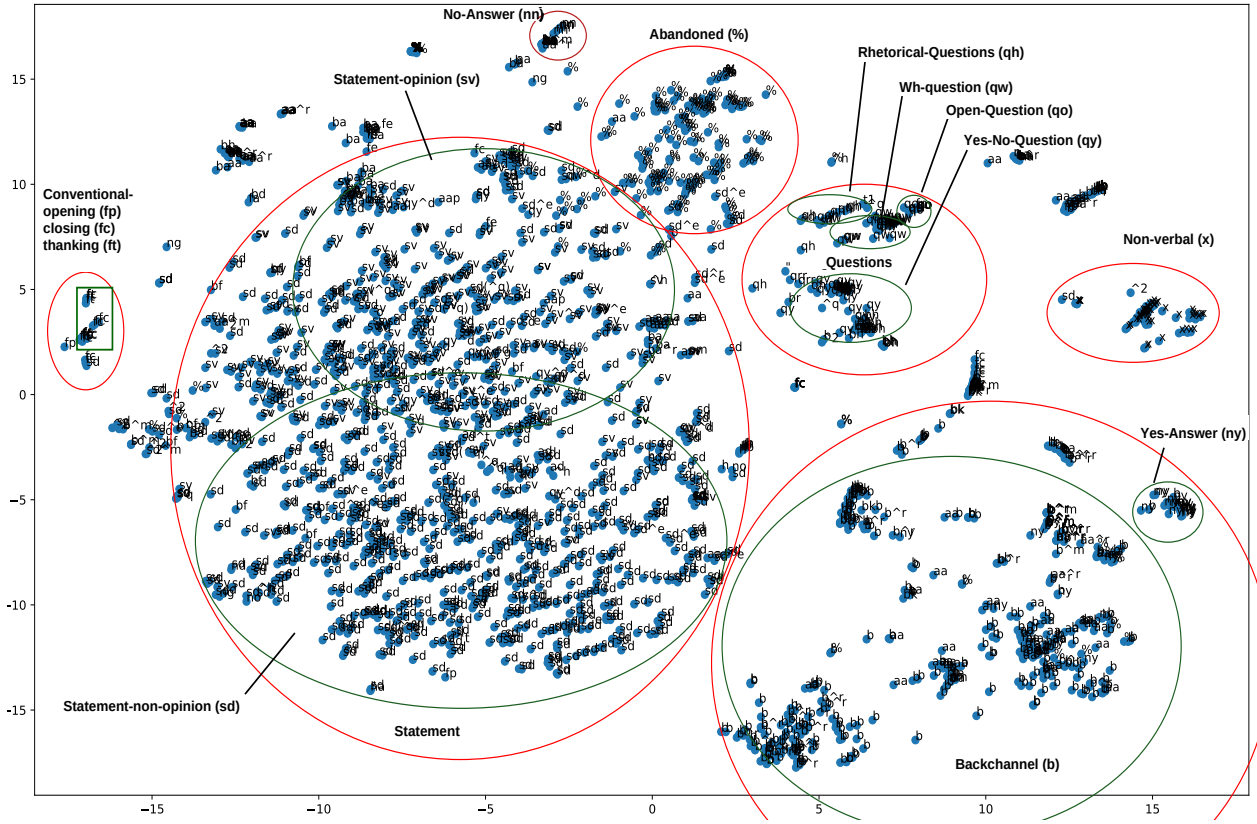


Figure 3: Clusters of all classes. Big clusters belong to the dominating *Statement* classes, *sv* and *sd*. The *Question* classes, *qy*, *qw*, *qh* and *qo* are clustered within the big class. The classes *Backchannel*, *Yes-answers*, and *Agree/Accept* share a lot of syntactic information hence they are clustered together, and our approach makes those classes separable within the cluster.

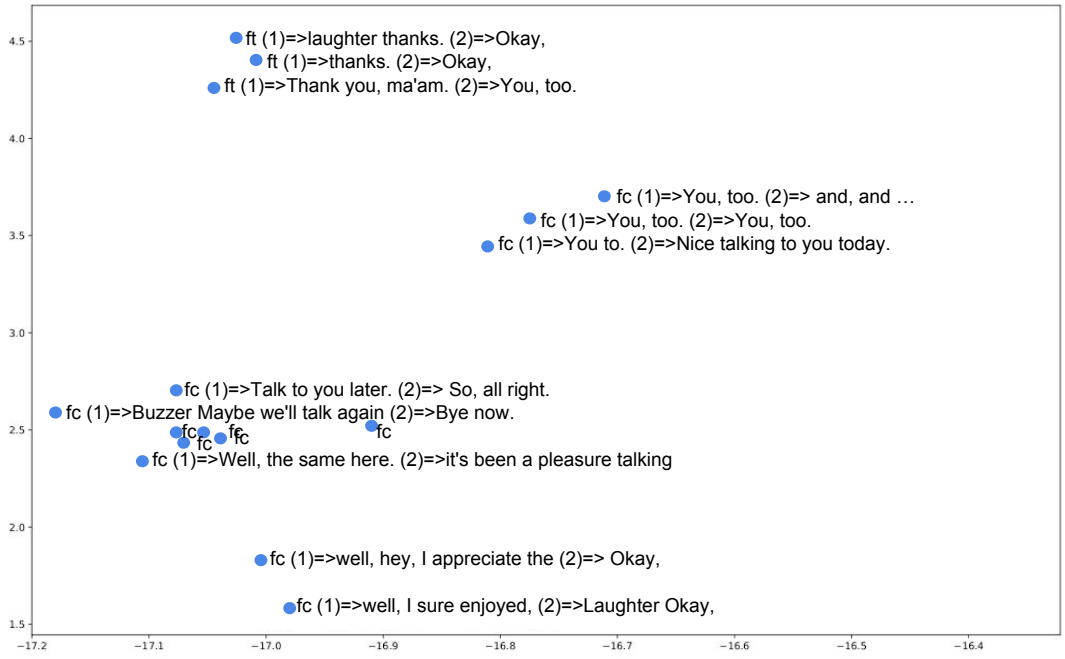


Figure 4: A blowup of the rectangle in Figure 3 from the *Conventional closing (fc)* and *thanking (ft) function* classes with their utterances. For readability, some utterances have been omitted and we show only the labels. These are examples of the context-sensitive dialogues, where we can see one cluster of the *ft* class and three groups of the *fc* class.

TREEANNOTATOR: Versatile Visual Annotation of Hierarchical Text Relations

Philipp Helfrich, Elias Rieb, Giuseppe Abrami, Andy Lücking, Alexander Mehler
Goethe University Frankfurt, Text-Technology Lab
Robert-Mayer-Straße 10, 60325 Frankfurt am Main
helfrich@cs.uni-frankfurt.de, s0294036@stud.uni-frankfurt.de, {abrami, luecking, mehler}@em.uni-frankfurt.de

Abstract

We introduce TREEANNOTATOR, a graphical tool for annotating tree-like structures, in particular structures that jointly map dependency relations and inclusion hierarchies, as used by Rhetorical Structure Theory (RST). TREEANNOTATOR is browser-based, embedded within the UIMA framework and provides two visualization modes. TREEANNOTATOR's interoperability exceeds similar tools, providing a wider range of formats, while annotation work can be completed more quickly due to a revised input method for RST dependency relations. TREEANNOTATOR offers a multiple window view, which allows users to inspect several annotations side by side. For storing and versioning annotations, the UIMA Database Interface (UIMA DI) was developed to save documents based on a pre-defined type system. These features not only connect TREEANNOTATOR annotations to modern technological and dialog theoretical work, but set it apart from related tools. The ease of use of TREEANNOTATOR and its newly designed user interface is evaluated in a user study consisting of annotating rhetorical relations with TREEANNOTATOR and the classic *RSTTool*.

Keywords: text annotation, text visualization, multiple views, rhetorical structure theory, UIMA, database

1. Introduction

Over the past years, a lot of annotation tools have been developed for various tasks (for a recent process and feature oriented overview see (Finlayson and Erjavec, 2017); for an overview of multimodal tools see (Cassidy and Schmidt, 2017)).

For some reason, however, the domain of discourse annotation, most notably in terms of *Rhetorical Structure Theory* (RST) (Mann and Thompson, 1988), has not gained much attention for quite a long time. Most RST annotations, with the RST Discourse Treebank (Carlson et al., 2003)(Carlson et al., 2014) leading the way, are carried out with the original *RST Annotation Tool* (O'Donnell, 1997) or its extension, the *ISI RST Annotation Tool* (Marcu, 1999). While still usable, both tools, however, are not maintained any more and do not comply to current annotation frameworks. Due to such reasons, a third, browser-based RST annotation tool has been developed recently: *rstWeb* (Zeldes, 2016). Why, then, is there need for a new tool like TREEANNOTATOR?

We highlight three reasons in the following: Firstly, TREEANNOTATOR, unlike *rstWeb*, follows the UIMA framework, which supports current requirements of annotation pipelines (Wilcock, 2017). Note that also the Atomic/ANNIS frameworks (Druskat et al., 2014; Krause and Zeldes, 2016) are not based on UIMA, and hence TREEANNOTATOR is a module following a technologically complementary approach, more related to the "NLP-affine" text annotation tool *BRAT* (Stenetorp et al., 2012) (which in turn is the technological and design source for UIMA-based *WebAnno* (de Castilho et al., 2014)). Even extending such frameworks, we developed an UIMA database interface in order to provide a fine-grained type system-compliant storage of documents.

Secondly, users of TREEANNOTATOR benefit from different visualizations. The usefulness of visualizing annotations has been independently highlighted, for instance by (Finlayson and Erjavec, 2017, p. 183) and (Biemann et al., 2017, p. 236). In case of polynuclear hierarchies there are several ways to display the resulting structures, which in turn emphasize different structural features. Accordingly, TREEANNOTATOR provides – in addition to the classical RST view – an inclusion view that adopts rhetorical structure representations in the tradition of *Discourse Representation Theory* (DRT) (Kamp and Reyle, 1993), namely *Segmented Discourse Representation Theory* (SDRT) (Asher and Lascarides, 2003). The visualization flexibility not only provides alternative views on given rhetorical structures, it also makes TREEANNOTATOR's output more "readable" to modern theoretical linguistics. Thus, TREEANNOTATOR may reach a new group of users.

Thirdly, the newly designed graphical interface of TREEANNOTATOR significantly improves the tool's usability for rhetorical annotations: this has been evaluated in a study in comparison to the *RSTTool*.

The components, annotation processes, and visualizations of TREEANNOTATOR are described in Sec. 2. The usability study is reported in Sec. 3. Future developments are summarized in Sec. 4.

2. TreeAnnotator

2.1. Annotating Tree-like Text Structures in TreeAnnotator

TREEANNOTATOR is an up-to-date, browser-based and freely available device for discourse annotation. It is designed as a module of the still-developing annotation suite TEXTANNOTATOR. TEXTANNOTATOR in turn draws on the architecture of TEXTIMAGER (Hemati et al., 2016), an UIMA-based framework that offers a wide range of NLP and visualization tools through a user-friendly GUI. All NLP tools that are available via TEXTIMAGER can be utilized for preprocessing input documents, say, in terms of tokenization, lemmatization or pos tagging. TREEANNOTATOR's front-end is based on the *Ext JS* framework[1] and

[1] https://www.sencha.com/products/extjs/#overview

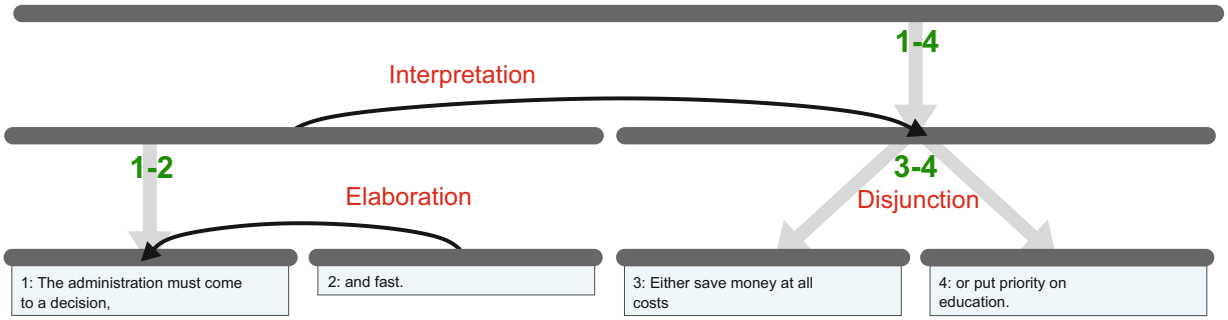


Figure 1: TREEANNOTATOR's classical RST mode, illustrating the same RST structure as shown in Fig. 2

uses the *D3.js*[2] library for visualizations. It was designed for annotating RST trees (Mann and Thompson, 1988) and structures that jointly map dependency relations and inclusion hierarchies. As such, TREEANNOTATOR covers and extends the functionality of previous RST annotation tools, but is completely newly developed in order to meet current technological standards and requirements.

The user interface characteristics of TREEANNOTATOR give rise to the following features, which are elaborated subsequently:

- the number of mouse clicks for carrying out an RST annotation is significantly reduced, thanks to a new approach of how to annotate rhetorical structures. In particular, spans no longer have to be added manually, but are included automatically based on selected relation types – a respective user study is reported in Sec. 3.;
- the flexibility and interoperability exceeds comparable tools by offering new output formats (e.g. LaTeX and XMI; see Sec. 2.2. and Sec. 2.3., respectively);
- TREEANNOTATOR provides an alternative mode of interactive visualization (and, hence, annotation), including a split-window facility, which extends the classical RST view (Sec. 2.2.);
- finally, TREEANNOTATOR includes a redo/undo functionality due to an underlying versioning system (Sec. 2.4.).

2.2. Visualizing Annotations in TextAnnotator

TREEANNOTATOR's visualization addresses current workflow needs more specifically than comparable tools: In particular, (a) a second mode of visualization is available, independent from classical RST trees; (b) LaTeX output for scientific publications can be generated; (c) a multiple window view to inspect annotations side by side is provided.

(a) Visualization Modes Representing data by means of alternative modes helps in understanding and interpreting the subject matter. To this end, TREEANNOTATOR supports two modes of visualizing tree-like structures: *classical RST trees* and *dependency-oriented inclusion hierarchies* (DIH) (see Fig. 1 and 2). TREEANNOTATOR allows for switching between the two available visualizations even during annotation.

Visualizations in RST mode are similar to those of *(ISI) RSTTool* and *rstWeb*. However, TREEANNOTATOR's mouse wheel zooming and dragging functionality adds a useful feature, especially for annotating larger tree structures: TREEANNOTATOR has been tested with trees of more than one hundred *Elementary Discourse Units* (EDUs, the text spans connected by rhetorical relations); *RSTTool* and *rstWeb* in comparison are limited to displaying roughly two dozen EDUs simultaneously on screen.

Unlike *ANNIS*, which was developed for generic graph visualizations, TREEANNOTATOR offers two dedicated RST modes, specifically designed for quick annotation work. ANNIS' latest version, ANNIS3, provides a solution for RST trees by means of a visualization plugin (Krause and Zeldes, 2016). However, the application is limited to only displaying previously annotated corpora. The corresponding annotation tool *Atomic* (Druskat et al., 2014) is currently still in development and not available as a browser-based tool.

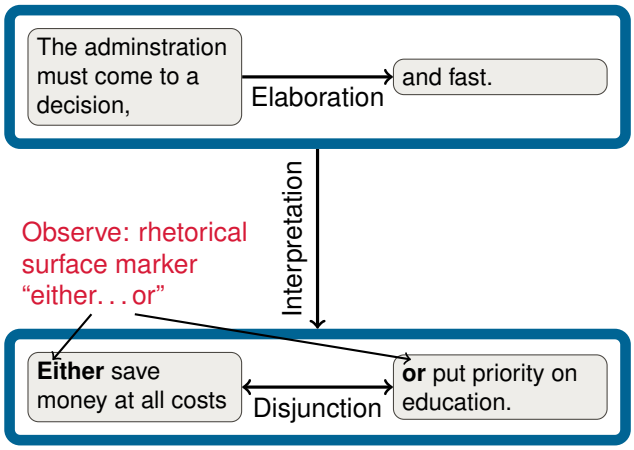


Figure 2: DIH: Mode to view integrated dependency and inclusion hierarchies in TREEANNOTATOR. The comment is added to the PGF/TikZ source code.

The DIH mode is not just a further means of visualization, it opens up a new domain of application, since it connects RST annotations to dynamic semantics in the DRT (Kamp and Reyle, 1993) tradition. The nested boxes representation format is familiar to formal semanticists from SDRT (Asher and Lascarides, 2003, e.g. p. 33), from where it is adopted in the first place. A particularly nice feature of the inclusion hierarchy representation – in contrast to the original RST rendering – is that the right frontier of a text becomes visible: the right frontier constitutes the structural

[2] https://d3js.org/

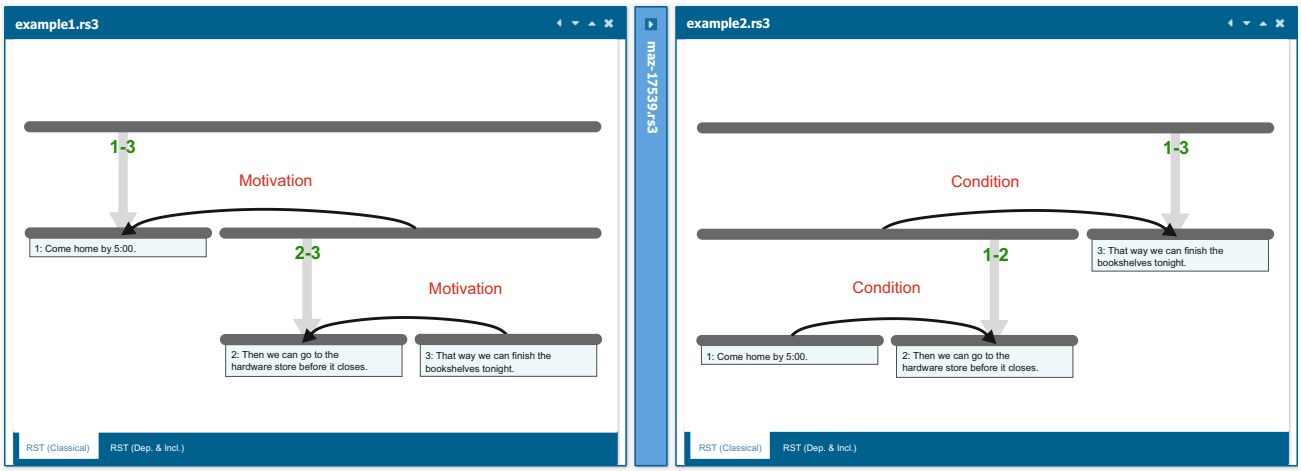


Figure 3: Multiple window view, showing two concurring annotations (Moore and Pollack, 1992, p. 542 f.). Several collapsible annotation views can be visualized.

boundary for anaphoric attachment sites and, hence, is of importance for any further co-reference annotation.

(b) Graphics Output Besides screenshot-like graphics (PNG), TREEANNOTATOR allows for generating LaTeX output: RST trees are exported using the *rst package* (Reitter, 2002), DIHs are output to PGF/TikZ (Tantau, 2015). PGF/TikZ are formal languages that can be interpreted by TeX (Knuth, 1984), LaTeX (Lamport, 1994) (including XeLaTeX), and ConTeXt (Hagen and Hoekwater, 2013) (including LuaTeX) to produce vector graphics output. The PGF/TikZ format is not only suitable for direct inclusion in LaTeX documents – the standard format of many scientific publication organs –, but enables further modifications within the graphic's source code. By this means, relevant features can be highlighted for publications – as illustrated in Fig. 2 –, or visualizations can be uncovered piecewise in presentations for talks.

(c) Multiple Window View A well-known problem of RST annotations is that there may be several, equally justified but concurring rhetorical analyses and structures of text spans (Moore and Pollack, 1992). Since this pluralism of rhetorical analyses is perfectly legitimate according to RST (Das et al., 2017), it has to be accounted for in the annotation workflow. Accordingly, a multiple window view was developed, which allows users to simultaneously inspect multiple annotations of the same text side by side, see Fig. 3. In the current version, this is achieved by simply creating a copy of the document. In future releases, the different annotations will be stored in the same document by using different *Subjects of Analysis*, that is, specific UIMA objects[3], which will further open up the possibility of having different annotations even for fragments of a text.

2.3. Storing Annotations in TextAnnotator

Since the Apache UIMA framework (Ferrucci et al., 2009) is one of the standards in projects within an NLP context (Wilcock, 2017), it is also used in the TEXTANNOTATOR suite. However, somewhat surprisingly this document-based schema does not provide native database support. To this end, we developed the *UIMA Database Interface* (UIMA DI) (Abrami and Mehler, 2018): it stores arbitrary UIMA documents based on their associated *UIMA Type System Descriptions* (UIMA TSD) in any database back-end (see Fig. 4). In the current version[4], the connections to MongoDB[5] and Neo4J[6] are already implemented and further database back-ends can be added easily.

For the general purpose of storing and using tree-like structures in UIMA frameworks, we developed a new UIMA TSD for graphs[7], compliant to the GraphML[8] format. Based on the UIMA TSD we defined an object model for RST trees, which is used in TEXTANNOTATOR (and thereby in TREEANNOTATOR). In order to efficiently store and retrieve annotations created by those tools, we employ established database technologies. Within this technological context, annotations are stored in the XMI[9] format during annotation processes. The resulting XMI document can be stored in any database integrated into UIMA DI or downloaded to a text file.

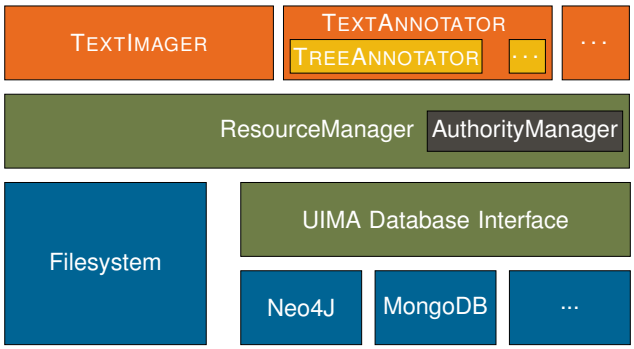


Figure 4: Components of the architecture

Table 1 summarizes the input/output formats which are cur-

[3] https://uima.apache.org

[4] Available on GitHub, under GPLv3.

[5] https://www.mongodb.com/

[6] https://neo4j.com

[7] http://www.textannotator.hucompute.org/typesystem/graph

[8] http://graphml.graphdrawing.org/index.html

[9] http://www.omg.org/spec/XMI/

Input	Data output	Graphical output
plain text, RST file format	UIMA-conform XMI file	LaTeX (PGF/TikZ), PNG

Table 1: Input/output formats of TREEANNOTATOR.

rently supported by TREEANNOTATOR.

2.4. Versioning Annotations in TextAnnotator

When dealing with error-prone, highly interpretive annotation tasks like RST analysis, undoing and redoing operations should be a fundamental part of the workflow (see e.g. the results in Sec. 3.2.). TEXTANNOTATOR currently supports an undo/redo functionality within the same session by versioning the XMI file in the internal memory. TREEANNOTATOR inherits TEXTANNOTATOR's versioning facility and thus provides the same functionality as *rstWeb* (v2.0) and *RSTTool* (v3.0), but exceeds tools such as *WebAnno* (v3.2.2), which does not offer an undo function. Thus, any annotation can be revised during the actual session. Since storing a copy of the complete file for every revision leads to a high memory usage, especially for larger documents, we are currently looking into genuine version control systems. Using such a system will not only lead to reduced memory usage, but will also allow TREEANNOTATOR to provide an undo/redo functionality even beyond sessions.

2.5. Managing Annotations in TextAnnotator

Since it cannot be guaranteed in any case that the natural language texts to be annotated are free of copyright restrictions, a rights and resource management is required. To this end, TEXTANNOTATOR uses components of the *eHumanities Desktop* (Gleim et al., 2012) which allows managing user rights and resources: *AuthorityManager* and *ResourceManager* (see Fig. 4). By integrating UIMA DI into *ResourceManager*, any UIMA document becomes accessible as a resource and organized in projects, which are defined as repositories. That is, user rights can be inherited from higher-level groups via restriction or extension in order to specify resource-specific permissions. Apart from that, TEXTANNOTATOR can also be used anonymously.

The need for a fine-grained rights management is evidenced by sample use-cases. A common classroom situation, for instance, involves students that work on an annotation task. The original texts are provided in a folder for which the group of students have only reading rights so that it is secured that the files cannot be modified or deleted by accident. Annotation results are stored in a second folder where students have write permission. The results of the other students cannot be changed or seen, as the students do not have reading rights to each other's documents, as required for inter-annotator evaluations.

In the context of research projects, files to be annotated are usually stored in a folder that is shared among the different project partners. All project partners have at least writing permissions and can therefore jointly work on the annotation sources. Now annotation results can be published in a targeted manner: they can be shared with individual users or made public to a wider audience simply by assigning appropriate authentications to files and users.

Tab. 2 compares TEXTANNOTATOR's rights and resource management, as facilitated by *AuthorityManager* and *ResourceManager*, to that of *rstWeb* and the widespread *WebAnno*: The access options applied by TEXTANNOTATOR are more elaborate than those of the compared tools. Especially, TEXTANNOTATOR acknowledges the level of groups, allows to share results with third parties and provides data access for external tools.

Feature	WebAnno	rstWeb	TextAnnotator
user management	✓	✓	✓
group management	✗	✗	✓
user permissions on projects / collections	✓	✓	✓
user permissions on documents	✗	✓	✓
group permissions on projects	✗	✗	✓
group permissions on documents	✗	✗	✓
project-independent document usage	✗	✗	✓
document organisation in repositories	✗	✗	✓
hierarchical repositories	✗	✗	✓

Table 2: Comparison of TEXTANNOTATOR, *WebAnno* and *rstWeb* with regards to rights and resource management

3. Evaluation

3.1. Setting

In a usability study, two groups of five test subjects each were asked to annotate pre-specified RST trees, comparing TEXTANNOTATOR to the most widely used RST annotation software, *RSTTool*. The subjects had no prior acquaintance with RST tools and RST annotations. However, they had a brief introduction into Rhetorical Rtructure Theory and its polynuclear relations. Twelve text segments (with three, five and seven RST relations, respectively) of the *Potsdam Commentary Corpus* (Stede, 2004) (Applied CL Discourse Research Lab, 2017) were handed out on paper. Group one annotated using TREEANNOTATOR, group two worked with *RSTTool*. The results are shown below.

3.2. Results

Using TREEANNOTATOR resulted in a decrease of mouse clicks per annotation. On average, annotators working with TREEANNOTATOR needed 14.9 clicks less compared to annotators using *RSTTool*; the median was improved by 18.0 clicks (or −32.7%, see Tab. 3).

	RSTTool		TreeAnnotator		improvement	
	avg	med.	avg	med.	avg %	med. %
short	36.8	36.5	22.4	19.0	−39.0%	−47.9%
medium	45.4	46.5	32.1	28.0	−29.3%	−39.8%
long	54.9	55.5	37.8	36.0	−31.2%	−35.1%
total	45.7	46.5	30.8	28.5	−32.7%	−38.7%

avg – average (arithmetic mean), med. – median

Table 3: Number of mouse clicks for 12 pre-specified RST annotations with *RSTTool* and TREEANNOTATOR

In terms of annotation time, TREEANNOTATOR again performed better in all measured categories. Using *RSTTool*,

annotators needed 104.9 sec for the twelve texts on average, while user of TREEANNOTATOR managed an average time of 96.3 sec (an improvement of −8.6 sec or −8.2%; see Tab. 4).

	RSTTool		TreeAnnotator		improvement	
	avg	med.	avg	med.	avg %	med. %
short	84.3	78.0	70.8	58.0	−16.0%	−25.6%
medium	101.9	101.5	92.9	82.5	−8.8%	−18.7%
long	128.6	124.5	125.3	116.5	−2.5%	−6.4%
total	104.9	101.5	96.3	95.0	−8.2%	−6.4%

avg – average (arithmetic mean), med. – median

Table 4: Required time (in sec) for 12 pre-specified RST annotations with *RSTTool* and TREEANNOTATOR

The number of annotation errors for both tools is virtually the same (see Tab. 5). However, the types of errors differed: annotators working with *RSTTool* mainly erred with respect to RST tree structures, while annotators using TREEANNOTATOR mainly produced incorrect text segmentations. As a result, TREEANNOTATOR's text segmentation was subsequently altered to make the currently selected segment more obvious for users.

	RSTTool	TreeAnnotator
Incorrect relation name	3	1
Incorrect RST structure	6	2
Incorrect text segmentation	0	5
Output file missing entirely	1	1
total	10	9

Table 5: Number of errors for each annotation tool (60 pre-specified RST annotations per group in total)

Users of TREEANNOTATOR (on average) needed 28.5% of their time for segmentation, while the program's undo function was used very sparsely. One concern illustrated in Tab. 6 is the fact that the time per newly included relation increases with text length, a problem partly due to the hardware used during the study, which did not allow for smooth zooming and moving of larger RST trees.

text length	time for segmentation (% of total time)	Undo actions (per text)	Time per new EDU (in sec)	Time per new relation (in sec)
short	29.6%	0.30	4.4	5.1
medium	28.6%	0.35	4.1	8.2
long	27.3%	0.15	4.1	12.1
total	28.5%	0.27	4.2	8.5

Table 6: Average user behavior for pre-specified RST annotations in TREEANNOTATOR

3.3. Summary

TREEANNOTATOR provides a state of the art approach to rhetorical structure annotation. TREEANNOTATOR outperforms *RSTTool* in all categories shown above (albeit only minimally in terms of error-proneness). Improvements in the number of needed mouse actions is a particularly remarkable result. Furthermore, we found a reduced annotation time, especially for shorter to medium length texts.

4. Outlook

TREEANNOTATOR offers a modern and arguably intuitive approach to discourse annotations for RST-like structures, with unique features that set it apart from related tools.

Future versions will add support for annotations based on representations from DRT and SDRT and a newly designed module for semantic analysis, utilizing various lexical resources to create a comprehensive model of semantic representation. However, rhetorical structure annotations will not be overloaded by adding, say, semantic or anaphoric annotations. Rather, a multilayer annotation scenario will be implemented, which has already proven to be useful, for instance, in the GUM project (Zeldes, 2017). TEXTANNOTATOR's code will be made available on GitHub in the future. Meanwhile, the tool can be reviewed at http://www.textannotator.hucompute.org (login: textAnnotator, password: textAnnotator).

5. Bibliographical References

Abrami, G. and Mehler, A. (2018). A UIMA Database Interface for Managing NLP-related Text Annotations. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press, Cambridge.

Biemann, C., Bontcheva, K., Eckart de Castilho, R., Gurevych, I., and Yimam, S. M. (2017). Collaborative web-based tools for multi-layer text annotation. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 229–256. Springer Netherlands, Dordrecht.

Carlson, L., Marcu, D., and Okurowski, M. E. (2003). Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In Jan van Kuppevelt et al., editors, *Current and New Directions in Discourse and Dialogue*, pages 85–112. Springer Netherlands, Dordrecht.

Cassidy, S. and Schmidt, T. (2017). Tools for multimodal annotation. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 209–227. Springer Netherlands, Dordrecht.

Das, D., Taboada, M., and Stede, M. (2017). The good, the bad, and the disagreement: Complex ground truth in rhetorical structure analysis. In *Proceedings of the 6th Workshop Recent Advances in RST and Related Formalisms*, pages 11–19, Santiago de Compostela, Spain.

de Castilho, R. E., Biemann, C., Gurevych, I., and Yimam, S. M. (2014). WebAnno: a flexible, web-based annotation tool for CLARIN. In *Proceedings of the CLARIN Annual Conference 2014*, CAC'14, Utrecht, Netherlands.

Druskat, S., Bierkandt, L., Gast, V., Rzymski, C., and Zipser, F. (2014). Atomic: an open-source software platform for multi-layer corpus annotation. In *Proceedings of the 12th Konferenz zur Verarbeitung natürlicher Sprache*, KONVENS 2014, pages 228–234.

Ferrucci, D., Lally, A., Verspoor, K., and Nyberg, E. (2009). Unstructured information management architecture (UIMA) version 1.0. OASIS Standard, mar.

Finlayson, M. A. and Erjavec, T. (2017). Overview of annotation creation: Processes and tools. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 167–191. Springer Netherlands, Dordrecht.

Gleim, R., Mehler, A., and Ernst, A. (2012). Soa implementation of the ehumanities desktop. In *Proceedings of the Workshop on Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts, Digital Humanities 2012, Hamburg, Germany*.

Hagen, H. and Hoekwater, T., (2013). *ConTEXt reference manual*.

Hemati, W., Uslu, T., and Mehler, A. (2016). Textimager: a distributed UIMA-based system for NLP. In *Proceedings of the COLING 2016 System Demonstrations*. Federated Conference on Computer Science and Information Systems.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht.

Knuth, D. E. K. (1984). *The TEX Book*. Addison-Wesley.

Krause, T. and Zeldes, A. (2016). ANNIS3: A new architecture for generic corpus query and visualization. *Digital Scholarship in the Humanities*, 31(1):118–139.

Lamport, L. (1994). *LATEX – A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley.

Mann, W. and Thompson, S. (1988). Rhethorical structure theory: Towards a functional theory of text organization. *Text*, 8(3):243–281.

Marcu, D., (1999). *Instructions for Installing the Rhetorical Annotation Tool*.

Moore, J. D. and Pollack, M. E. (1992). A problem for RST: The need for multi-level discourse analysis. *Computational Linguistics*, 18(4):537–544.

O'Donnell, M. (1997). RST-Tool: An RST analysis tool. In *Proceedings of the 6th European Workshop on Natural Language Generation*.

Reitter, D., (2002). *Rhetorical theory in LATEX with the rst package. Technical Manual*.

Stede, M. (2004). The Potsdam Commentary Corpus. In *Proceedings of the 2004 ACL Workshop on Discourse Annotation*. Association for Computational Linguistics.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL 2012, pages 102–107, Avignon, France.

Tantau, T., (2015). *The TikZ and PGF Packages*, manual for version 3.0.1a edition.

Wilcock, G. (2017). The evolution of text annotation frameworks. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, pages 193–207. Springer Netherlands, Dordrecht.

Zeldes, A. (2016). rstWeb – a browser-based annotation interface for rhetorical structure theory and discourse relations. In *Proceedings of NAACL-HLT 2016 System Demonstrations*, pages 1–5.

Zeldes, A. (2017). The GUM corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

6. Language Resource References

Applied CL Discourse Research Lab. (2017). *Potsdam Commentary Corpus*. 2.0, ISLRN 867-681-112-435-1.

Carlson, Lynn and Marcu, Daniel and Okurowski, Mary Ellen. (2014). *RST Discourse Treebank*. Linguistic Data Consortium, 1.0, ISLRN 299-735-991-930-2.

Chats and Chunks: Annotation and Analysis of Multiparty Long Casual Conversations

Emer Gilmartin, Carl Vogel, Nick Campbell
Trinity College Dublin
Ireland
{gilmare, vogel, nick}@tcd.ie

Abstract

Casual talk or social conversation is a fundamental form of spoken interaction. Corpora of casual talk often comprise relatively short dyadic conversations, although research into such talk has found longer multiparty interaction to be very common. This genre of spoken interaction is attracting more interest with attempts to build more friendly and natural spoken dialog systems. To study longer multiparty casual talk, we have assembled a collection of conversations from three existing corpora. We describe the collection, organization, and annotation of structural chat and chunk phases in these conversations. we then review our preliminary results, noting significant differences in the distribution of overlap, laughter and disfluency in chat and chunk phases, and finding that chunk dominates as conversations get longer. We outline our continuing work on gaining greater understanding of this genre of spoken interaction, with implications for the design of spoken dialog systems.

Keywords: casual conversation, multimodal corpus, annotation

1. Introduction

Many of the practicalities of life are managed through spoken interaction. Such interaction is task-based or practical, facilitating activities such as service encounters (buying a pizza), workplace projects (meetings), or education (tutorials, discussions). People also talk for social reasons – such interactional talk or casual conversation happens whenever people congregate, from short greetings at bus stops to interactions such as dinner party conversations lasting several hours. Although casual conversation is ubiquitous, it is not as well studied as practical talk. We investigate the structure of such talk, particularly in multiparty long (c. 1 hour) conversations. Dialog research has advanced greatly through studies of corpora of relevant speech interaction. In recent years, several high quality corpora have been made available and have underpinned a wide range of research. However, there remain very few collections of longer stretches of multiparty casual talk, and although the use of existing corpora of multiparty talk in scenarios such as meetings is convenient, it is unclear that results would generalise to casual talk. We have gathered a set of six 3 to 5 party conversations, drawn from three corpora, which have been manually segmented, transcribed, and annotated structurally into conversational phases - interactive chat phases and chunk phases where one speaker dominates. We have performed several analyses of the structure of these phases in terms of the occurrence of speech, laughter, and silence, and in the distribution of disfluencies and overlap. The motivation for this work is two-fold. First, the scientific goal of greater understanding of this fundamental human behaviour, and second, to better model conversation in human-machine interaction, particularly in dialog systems which engage in social, companionable talk. Below, we briefly review theories of casual conversation, describe the collection and annotation of the data used, and report on our current and future work towards a fuller account of the structure of multiparty casual conversation.

2. Casual Conversation

Studies of casual conversation have focussed on form and content, and on discourse and sociolinguistic functions.

2.1. Descriptions of Social Talk

Casual social conversation is described as 'talking just for the sake of talking'(Eggins and Slade, 2004), and its subgenres include smalltalk, gossip, and conversational narrative. Aimless social talk or 'phatic communion' has been described as an emergent activity of congregating people, and viewed as the most basic use of speech (Malinowski, 1936). Researchers in fields including anthropology, evolutionary psychology, and communication have theorized that such talk functions to build social bonds and avoid unfriendly or threatening silence, rather than simply to exchange information or express thought, as postulated in much linguistic theory. Instances of these views are found in the phatic component in Jakobson's model of communication (Jakobson, 1960), distinctions between interactional and instrumental language (Brown and Yule, 1983), and theories that language evolved to maintain social cohesion through verbal grooming (Dunbar, 1998). It has long been speculated that the prosodic and gestural aspects of social talk carry much of its communicative load, that 'how' things are said is as important as 'what' is said (Abercrombie, 1956; Hayakawa, 1990). Slade and Eggins view casual conversation as the space in which people form and refine their social reality (Eggins and Slade, 2004) citing gossip between workmates, where participants reaffirm their solidarity, and examples of conversation between friends at a dinner party where greater intimacy allows differences of opinion. Schneider collected and analysed audio recordings of naturally occurring small talk, concentrating on the linguistic content of entire dialogues (Schneider, 1988), highlighting how the relative paucity of propositional information flow casual talk which did not seem to conform to Gricean ideas of dialogue - in particular, idling sequences of repetitions of agreeing tails such as 'Yes, of course',

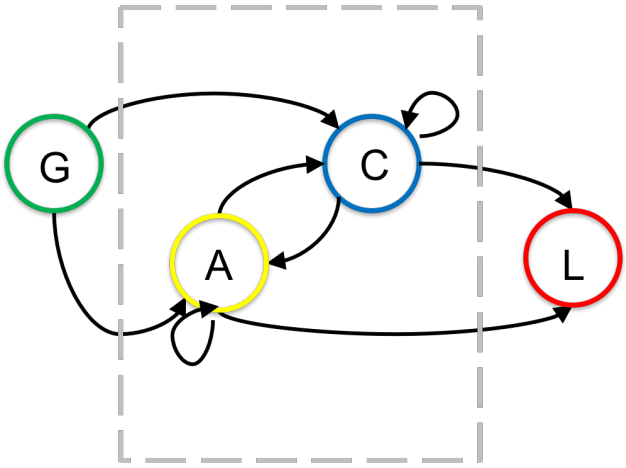


Figure 1: Greeting, Approach, Centring, and Leave-taking phases of casual conversation; a simplified version of Ventola's model.

'MmHmm' which seem to keep the conversation going rather than add any new information. He proposed a set of maxims concentrated on the importance of avoiding silence and maintaining politeness, and suggested that Grice's Co-operative Principle itself remained relevant to small talk although several of the related maxims, particularly those related to quantity and quality, did not apply as strongly as in more practical talk. Syntactical, lexical, and discourse differences between (casual) conversation and more formal spoken and written genres are described in Biber and Leech's work on the Longman Corpus of Spoken and Written English (LSWE), and particularly in their chapter on the grammar of conversation (Biber et al., 1999). Many researchers have also cited a possible text bias in linguistics as a retarding factor on the analysis of spoken interaction (Ong, 1982; Chafe and Danielewicz, 1987; Halliday, 1989).

2.2. Structure of Casual Talk

In early work on social conversation or smalltalk, Laver focussed on the edges of conversation,'psychologically crucial margins of interaction', suggesting that light smalltalk allowed transition from initial greetings to the main business of the interaction and back to closing sequences and to leave taking (Laver, 1975). Laver proposed that such talk suspended power or social differentials between interlocutors, and tended to treat subjects which were uncontroversial. Ventola described casual conversation in terms of distinct phases (Ventola, 1979); often beginning with ritualised opening greetings, followed by approach segments of light uncontroversial small talk, and in longer conversations leading to more informative centre phases consisting of sequential but overlapping topics, and then back to ritualised leavetakings, as shown in Figure 1.

Slade and Eggins contend that casual talk can be seen as sequences of 'chat' and 'chunk' elements (Eggins and Slade, 2004, p. 230). Chunks are segments where 'one speaker takes the floor and is allowed to dominate the conversation for an extended period', and the chunk appears to move through predictable stages – that is, it is generic. 'Chat' segments, on the other hand, are described as highly interactive, appearing to be managed locally, unfolding move by move or turn by turn, and thus amenable to Conversation Analysis style study. Figure 2 shows examples of chat and chunk phases taken from the dataset described in this paper.

Text type	Percentage
Storytelling	43.4
Observation/Comment	19.75
Opinion	16.8
Gossip	13.8
Joke-telling	6.3

Table 1: Relative frequencies of chunk phase genres in Slade's workplace conversations

In a study of three hours of conversational data collected during coffee breaks in three different workplaces, Slade found that around fifty percent of all talk could be classified as chat, while the rest comprised longer form chunks from the following genres: storytelling, observation/comment, opinion, gossip, joke-telling and ridicule. Excluding ridicule and chat, which are not amenable to genre analysis, Table 1 shows the relative frequency of the genres encountered by Slade in chunk phases in her conversational data.

Slade and Eggins also report that casual conversation tends to involve multiple participants rather than the dyads normally found in instrumental interactions or examples from conversation analysis. Instrumental and interactional exchanges differ in duration; task-based conversations are bounded by task completion and tend to be short, while casual conversation can go on indefinitely. Several researchers on casual conversation have noted that their analyses were limited as they were based on transcripts and thus lacked vital timing and multimodal information.

In the work described below, We focus on the chat and chunk phases in casual conversation. We examine casual conversations (c. 1 hour each in duration) to better understand chat and chunk structure in terms of speech and silence distribution, and the occurrence of laughter and disfluency.

3. Conversational Data

Multimodal corpora of spoken interaction have proven invaluable to researchers in understanding the bundle of signals present in face to face communication. Many multimodal and indeed audio corpora created in laboratory and 'real-world' conditions have been collections of performances of the same spoken task by different subjects, or of interactions specific to particular domains where lexical content was fundamental to progress towards a practical goal - such corpora include collections of information gap activities such as the HCRC MapTask corpus of dyadic information gap task-based conversations (Anderson et al., 1991). Other corpora have focussed on collecting recordings of real or staged meetings, such as the ICSI and AMI multiparty meeting corpora (Janin et al., 2003; McCowan et al., 2005), or recordings of particular genres of interaction, such as televised political interviews (Beattie, 1983).

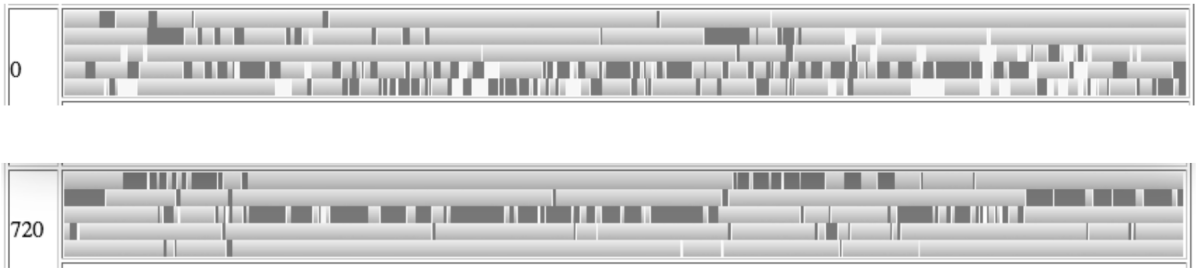


Figure 2: Examples of chat (top) and chunk (bottom) phases in two stretches from a 5-party conversation in the dataset. Each row denotes the activity of one speaker across 120 seconds. Speech is dark grey, and laughter is white on a light grey background (silence).The chat frame, taken at the beginning of the conversation, can be seen to involve shorter contributions from all participants with frequent laughter. The chunk frame shows longer single speaker stretches.

All of these corpora have contributed greatly to research into areas of spoken dialogue such as timing, turntaking, and dialogue architecture. However, the speech in these resources, while spontaneous and conversational, cannot be considered casual talk, and the results obtained from their analysis may not transfer to casual conversation.

Collections of non-task interaction include audio collections of casual talk, often telephonic, such as SWITCHBOARD (Godfrey et al., 1992) and the ESP-C collection of Japanese telephone conversations (Campbell, 2007) There are also collections of face-to-face talk as in the Santa Barbara Corpus (DuBois et al., 2000), and sections of the ICE (Greenbaum, 1991) and British National Corpus (BNC-Consortium, 2000). The Gothenburg Corpus of recordings of different types of human activity contains both audio and video recordings including casual or small talk (Allwood et al., 2000).

There is a growing number of multimodal corpora of mostly dyadic 'first encounters' where strangers were recorded engaged in casual conversation for periods of 5 to 20 minutes or so (Edlund et al., 2010; Aubrey et al., 2013; Paggio et al., 2010). These corpora are very valuable for the study of dyadic interaction, particularly at the opening and early stages of interaction. For a fuller review of available corpora and the challenges of genre in conversation, see (Gilmartin et al., 2015a). However, we are interested in the substance of longer casual conversation beyond these first encounters, and thus we have collected a number of multimodal recordings of conversations of multiparty casual speech to form a dataset for preliminary explorations.

4. Data and Annotation

We assembled a dataset of six informal conversations with three to five participants, each around an hour long. The conversations were drawn from three multimodal corpora, d64, DANS, and TableTalk (Oertel et al., 2010; Hennig et al., 2014; Campbell, 2008). Details of the dataset can be seen in Table 2, and further details of the annotation process can be found in (Gilmartin and Campbell, 2016).

In each of the corpora used, participants were recorded in casual conversation in a living room setting or around a table, with no instructions on topic of type of conversation to be carried out - participants were also clearly informed that they could speak or stay silent as the mood took them. All of the conversations were recorded in audio and video, using chest mounted or adjacent microphones and multiple video angles.

Corpus	Participants	Gender	Duration (s)
D64	5	2F/3M	4164
DANS	3	1F/2M	4672
DANS	4	1F/3M	4378
DANS	3	2F/1M	3004
TableTalk	4	2F/2M	2072
TableTalk	5	3F/2M	4740

Table 2: Source corpora and details for the conversations used in dataset

4.1. Segmentation

The recordings were found to be unsuitable for automatic segmentation using voice activity detection (VAD). While VAD could handle stretches where participants were talking without overlap ('in the clear'), many turn changes involved overlap and there was significant choral production of short utterances and laughter as well as within turn overlap when listeners produced backchannels. Although recordings were made with microphones worn by participants or adjacent to them, there was considerable bleedover between the recordings. After manual synchronisation, the audio files for each speaker were segmented manually into speech and silence intervals using Praat (Boersma and Weenink, 2010) on 10 and 4-second or smaller windows as necessary, and unclear cases were resolved using Elan (Wittenburg et al., 2006) to refer to the video recordings taken at the same time. Human annotators are not completely reliable – in listening tests humans have been shown to interpret the gist of spoken language rather than what exactly they heard, resulting in listeners missing or imagining the existence of short pauses, especially when there is elongation of previous or following syllables (Martin, 1970) or having difficulties recalling disfluencies (Deese, 1980). However, the annotators here had the benefit of visual representations of the waveform and spectogram in Praat, and

thus it is hoped that segmentation is accurate. The segmentation and transcription was carried out at the intonational phrase (IP) level rather than the more commonly used interpausal unit (IPU) as IPs are a basic unit for intonation study and can easily be concatentated to the interpausal unit (IPU) and turn level as required.

4.2. Transcription, Annotation of Disfluency, Prosodic Annotation

After segmentation the data were manually transcribed and non-verbal vocalizations were annotated, using a scheme largely derived from the TRAINS transcription scheme (Heeman and Allen, 1995). Words, hesitations, filled and unfilled pauses, unfinished words, laughs and coughs were transcribed and marked. To facilitate annotation of disfluency, the transcriptions and audio for each IPU were force aligned at the word and phoneme level with the Penn Aligner (Yuan and Liberman, 2008). Sections which could not be automatically aligned, where there was significant overlap or cut off words, were manually aligned.

Symbol	Note
.	interruption point
-	unfinished word
~	unfinished utterance
^	contracted word
r	repeated word
s	substituted word
d	deleted word
f	filled pause
x	pause
o	overlap

Table 3: The annotation code used for basic disfluencies.

The word level transcription was then used with the sound files to annotate disfluencies. The scheme and procedures used were based on those outlined in Shriberg's and Eklund's respective theses (Shriberg, 1994; Eklund, 2004), and Lickley's annotation manual for the MapTask corpus (Lickley, 1998), with extra labels and conventions for recycled turn beginnings (Schegloff, 1987), disfluencies in the presence of overlap, and unfinished utterances. Complex, or nested, disfluencies were labelled following Shriberg's method (Shriberg, 1994). A fuller account of the segmentation, transcription, and disfluency annotation process can be found in (Gilmartin and Campbell, 2016)
We have also manually annotated phrase final pitch movements in a subsection of the corpus in order to investigate how turn-taking cues manifest in chat and chunk segments. Annotation of intonation contours was carried out using the IViE system (Grabe, 2001).

4.3. Chat and Chunk Annotation

All six conversations were segmented into phases by first identifying the 'chunks' using the first, structural part of Slade and Eggins' definition - 'a segment where one speaker takes the floor and is allowed to dominate the conversation for an extended period' (Eggins and Slade, 2004). All other interaction was considered chat.

Code	Details
Type	chat:o, chunk:x
Owner	speaker code/z-everybody
Chunk Genre	story:s, observation/comment:c opinion:o, gossip:g

Table 4: Labelling Scheme for Chat and Chunk Phases

The chunks were first roughly identified from the transcriptions. The type of chunk was then decided with reference to Slade and Eggins taxonomy. Temporal boundaries of each chunk marked off on a 'phases' tier in Praat. Intervals were labelled using the code shown in Table 4, marking type of phase (chat - o or chunk - x), subtype of phase (narrative, story, discussion..), name of phase (roughly equivalent to the topic under discussion), and phase 'owner' (main speaker in chunks and everyone in chat phases). As an example, the code **x_s_g_cats** would denote a chunk phase where the main speaker is **g** and the chunk, which was in story form, was about cats. A total of 213 chat and 358 chunk phases were identified across the six conversations.

5. Overview of Preliminary Results

The dataset has been used to explore chat and chunk phases in conversation, in terms of timing, distribution of speech silence, laughter,overlap, and disfluency. Below we review our preliminary results and implications for dialog system technology.

5.1. Chat and Chunk Description and Distribution

There were a total of 571 segments of chat or chunk in the dataset, comprising 213 chat segments and 358 chunk segments. The number and total durations of chat and chunk segments per conversation can be seen in Table 5.

Conv	Chat No.	Chat Dur.	Chunk No.	Chunk Dur.
A	42 (36%)	1636 (39%)	73	2527
B	38 (32%)	1371 (29%)	82	3300
C	53 (44%)	1363 (31%)	68	3014
D	18 (26%)	660 (22%)	51	2343
E	17 (41%)	909 (44%)	24	1159
F	45 (43%)	2168 (46%)	60	2571

Table 5: Number and duration of chat and chunk segments per conversation, with percentage of conversation in terms of chat phase number, and chat phase duration

It can be seen that in all conversations there were more chunk phases and the time spent overall in chunk phases is greater than that spent in chat phases. Our data differs from Slade's in terms of the proportion of conversational time devoted to chat or chunk. In the dataset examined here, chunk accounts for more conversational time in all conversations than does chat, in contrast to Slade's finding of a 50/50 split.

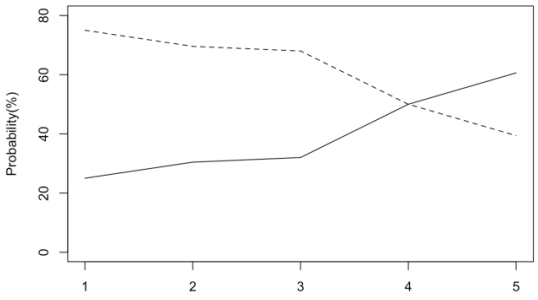


Figure 3: Probability of chunk-chunk transition (solid) and chunk-chat transition (dotted) as conversation elapses (x-axis = time) for first 30 minutes of conversation data in 6-minute bins

We also observed more chat at conversation beginnings, with chat predominating for the first 8-10 minutes of conversations. Although our sample size is small, this observation conforms to descriptions of casual talk in the literature, and reflects the structure of 'first encounter' recordings. However, as the conversation develops, chunks start to occur much more frequently, and the structure alternates between single-speaker chunks interleaved with shorter chat segments. In longer conversations, the likelihood of a chunk being followed by chat decreases and chunk to chunk transitions become more common, with the conversation entering a 'swapping stories' phase, as shown in Figure 3. It seems likely that the difference in composition of talk in our data is due to the longer lengths of conversations, and the increased prevalence of chunks and indeed chunk to chunk transition with greater conversational length. From these results it would appear that existing corpora of first encounters or the initial extended chat segments found in longer conversations can be used to model 'getting to know you' interactions or brief casual talk. However, it is clear that we need to model the chunk heavy central segments of longer conversation if we want to create systems which form a longer-term dialogic relationship with users.

From our analysis of the dataset we have found that the distributions of durations of chat and chunk phases are different, with chat phases durations varying more while chunk durations have a more consistent clustering around the mean. Chat phase durations (Mean=28s) tend to be shorter than chunk durations (Mean=34s). These findings are not speaker specific in our preliminary experiments and seem to indicate a natural limit for the time one speaker should dominate a conversation. The dimensions of chat and chunk durations observed would indicate that social talk should 'dose' or package information to fit chat and chunk segments of roughly these lengths. In particular, the tendency towards chunks of around half a minute could help in the design of narrative or education-delivering speech applications, by allowing designers to partition content optimally.

5.2. Speech, Overlap, Laughter and Disfluency Distribution

The commonest conversational state was a single participant speaking 'in the clear' (68%), with global silence accounting for 23% of the conversational time. The remaining time (9%) comprised overlapping speech by two or more participants, with instances dropping sharply as the number of overlapping speakers increases. The vast bulk of overlap in all conversations involved two speakers. There is significantly less overlap and more single party speech in chunk phases than in chat phases. We have also been investigating the frequency and distribution of laughter and disfluencies. Early experiments showed that laughter, and particularly shared laughter, appears more common in social talk than in meeting data, and that laughter happens more around topic endings/topic changes (Gilmartin et al., 2013a; Bonin et al., 2012). This is consistent with our current work on chat and chunk phases, as we are seeing that laughter is significantly more common in chat phases – which provide a 'buffer' between single speaker and topic chunks. In the current dataset we have found that laughter accounts for approximately 10% of vocal time in chat phases while it only accounts for 4% of chunk phases. For disfluencies, a pilot study has shown differences in the occurrence and distribution of disfluency types for chunk owners in chunks and all other speakers (Gilmartin et al., 2015b). In the chunk modality one speaker holds the floor for an extended period and this behaviour is different to that of all other speakers in chunks, to that of all speakers in chat, and indeed to that of the chunk owner when in somebody else's chunk.

6. Current and Future Work

We are studying the patterning of speaker contributions in both chat and chunk phases, particularly the length of gap or overlap in the vicinity of speaker and phase changes. We are performing prosodic analysis of the utterance final pitch movements in different contexts, using the IViE annotations, and believe the results of this work will provide information helpful in developing more finegrained 'endpointing' systems to determine *when* a system should speak; with knowledge of how turntaking occurs in different phases of talk we can work towards providing systems with turntaking behaviour appropriate to the current conversational phase. We are also currently completing dialog act annotation using the ISO 24617-2 standard in order to see how well the ISO standard covers casual conversation, and whether additional acts are necessary to reflect the goals and mechanisms of non-task conversation. In addition, we are further analysing the composition of chunk phases (narrative, gossip, etc) at different stages of conversation, in order to form a clearer picture of how longer conversations develop. This work will aid understanding of the genre, and also have useful applications in the design of artificial dialog.

7. Conclusions

We have described the segmentation, transcription and annotation of a dataset of six long multiparty casual conversations, introducing annotation of chat and chunk phases

within such conversations. The annotations have already enabled fruitful investigations into this omnipresent form of spoken interaction, in areas including laughter, overlap and disfluency (Gilmartin et al., 2013b; Gilmartin et al., 2015b), and the structure and timing of chat and chunk phases. We believe that greater understanding of casual conversation can lead to improvements in timing, dialog management, and natural language understanding in spoken dialog systems. The bulk of the recordings and the resulting annotations are available to interested researchers. Our investigations are preliminary and restricted by the lack of corpora of non task-based conversation, and particularly of long form casual or social talk. This kind of interaction is now of interest in the creation of the next generation of dialog systems - where social talk capacity will be important. We hope that these early explorations strengthen the case for creation of further collections of longer form casual talk, and encourage more investigation of aspects of this genre of spoken interaction.

8. Acknowledgements

This work is supported by the European Coordinated Research on Long-term Challenges in Information and Communication Sciences and Technologies ERA-NET (CHISTERA) JOKER project, JOKe and Empathy of a Robot/ECA: Towards social and affective relations with a robot, and by the ADAPT Centre for Digital Content Technology, which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

9. Bibliographical References

Abercrombie, D. (1956). *Problems and principles: Studies in the Teaching of English as a Second Language*. Longmans, Green.

Allwood, J., Bjornberg, M., Gronqvist, L., Ahlsen, E., and Ottesjo, C. (2000). The spoken language corpus at the department of linguistics, Goteborg University. In *FQS-Forum Qualitative Social Research*, volume 1.

Anderson, A., Bader, M., Bard, E., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., et al. (1991). The HCRC map task corpus. *Language and speech*, 34(4):351–366.

Aubrey, A. J., Marshall, D., Rosin, P. L., Vandeventer, J., Cunningham, D. W., and Wallraven, C. (2013). Cardiff Conversation Database (CCDb): A Database of Natural Dyadic Conversations. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 277–282, June.

Beattie, G. (1983). *Talk: An analysis of speech and non-verbal behaviour in conversation*. Open University Press.

Biber, D., Johansson, S., Leech, G., Conrad, S., Finegan, E., and Quirk, R. (1999). *Longman Grammar of Spoken and Written English*, volume 2. Longman London.

BNC-Consortium. (2000). British national corpus. *URL http://www. hcu. ox. ac. uk/BNC*.

Boersma, P. and Weenink, D. (2010). Praat: doing phonetics by computer [Computer program], Version 5.1. 44.

Bonin, F., Campbell, N., and Vogel, C. (2012). Laughter and topic changes: Temporal distribution and information flow. In *Cognitive Infocommunications (CogInfoCom), 2012 IEEE 3rd International Conference on*, pages 53–58.

Brown, G. and Yule, G. (1983). *Teaching the Spoken Language*, volume 2. Cambridge University Press.

Campbell, N. (2007). Approaches to conversational speech rhythm: Speech activity in two-person telephone dialogues. In *Proc XVIth International Congress of the Phonetic Sciences, Saarbrucken, Germany*, pages 343–348.

Campbell, N. (2008). Multimodal processing of discourse information; the effect of synchrony. In *Universal Communication, 2008. ISUC'08. Second International Symposium on*, pages 12–15.

Chafe, W. and Danielewicz, J. (1987). *Properties of spoken and written language.* Academic Press.

Deese, J. (1980). *Pauses, prosody, and the demands of production in language*. Mouton Publishers.

DuBois, J. W., Chafe, W. L., Meyer, C., and Thompson, S. A. (2000). *Santa Barbara Corpus of Spoken American English. CD-ROM. Philadelphia: Linguistic Data Consortium*.

Dunbar, R. (1998). *Grooming, gossip, and the evolution of language*. Harvard Univ Press.

Edlund, J., Beskow, J., Elenius, K., Hellmer, K., Strombergsson, S., and House, D. (2010). Spontal: A Swedish Spontaneous Dialogue Corpus of Audio, Video and Motion Capture. In *LREC*.

Eggins, S. and Slade, D. (2004). *Analysing Casual Conversation*. Equinox Publishing Ltd.

Eklund, R. (2004). *Disfluency in Swedish human-human and human-machine travel booking dialogues*. Dept. of Computer and Information Science, Linkoping Studies in Science and Technology, Linkoping, Sweden.

Gilmartin, E. and Campbell, N. (2016). Capturing Chat: Annotation and Tools for Multiparty Casual Conversation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Gilmartin, E., Bonin, F., Vogel, C., and Campbell, N. (2013a). Laugher and Topic Transition in Multiparty Conversation. In *Proceedings of the SIGDIAL 2013 Conference*, pages 304–308, Metz, France, August. Association for Computational Linguistics.

Gilmartin, E., Hennig, S., Chellali, R., and Campbell, N. (2013b). Exploring sounded and silent laughter in multiparty social interaction - audio, video and biometric signals. Valetta, Malta, October.

Gilmartin, E., Bonin, F., Cerrato, L., Vogel, C., and Campbell, N. (2015a). What's the Game and Who's Got the Ball? Genre in Spoken Interaction. In *2015 AAAI Spring Symposium Series*.

Gilmartin, E., Vogel, C., and Campbell, N. (2015b). Disfluency in Multiparty Social Talk. In *Proceedings of DISS 2015*, Edinburgh.

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520.

Grabe, E. (2001). The IViE labelling guide. *Journal of the Acoustical Society of America*, 101:3728–3740.

Greenbaum, S. (1991). ICE: The international corpus of English. *English Today*, 28(7.4):3–7.

Halliday, M. A. (1989). Spoken and written language.

Hayakawa, S. I. (1990). *Language in thought and action*. Houghton Mifflin Harcourt.

Heeman, P. A. and Allen, J. F. (1995). The TRAINS 93 Dialogues. Technical report, DTIC Document.

Hennig, S., Chellali, R., and Campbell, N. (2014). The D-ANS corpus: the Dublin-Autonomous Nervous System corpus of biosignal and multimodal recordings of conversational speech. Reykjavik, Iceland.

Jakobson, R. (1960). Linguistics and poetics. In Th. A. Sebeok, editor, *Style in language*, pages 350–377. MA: MIT Press, Cambridge.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., and Stolcke, A. (2003). The ICSI meeting corpus. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–364.

Laver, J. (1975). Communicative Functions of Phatic Communion. In Adam Kendon, et al., editors, *Organization of behavior in face-to-face interaction*, pages 215–238. Mouton, Oxford, England.

Lickley, R. J. (1998). HCRC disfluency coding manual. *Human Communication Research Centre, University of Edinburgh*.

Malinowski, B. (1936). The Problem of Meaning in Primitive Languages. In *The meaning of meaning: a study of the influence of language upon thought and of the science of symbolism*, pages 296–336. Kegan Paul, Trench, Trübner, London, 4th ed. rev edition.

Martin, J. G. (1970). On judging pauses in spontaneous speech. *Journal of Verbal Learning and Verbal Behavior*, 9(1):75–78.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., and Karaiskos, V. (2005). The AMI meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88.

Oertel, C., Cummins, F., Edlund, J., Wagner, P., and Campbell, N. (2010). D64: A corpus of richly recorded conversational interaction. *Journal on Multimodal User Interfaces*, pages 1–10.

Ong, W. J. (1982). *Orality and literacy: the technologizing of the word*. New accents. Methuen, London.

Paggio, P., Allwood, J., Ahlsen, E., and Jokinen, K. (2010). The NOMCO multimodal Nordic resource - goals and characteristics.

Schegloff, E. A. (1987). Recycled turn beginnings: A precise repair mechanism in conversation's turn-taking organization. *Talk and social organization*, pages 70–85.

Schneider, K. P. (1988). *Small Talk: Analysing Phatic Discourse*, volume 1. Hitzeroth Marburg.

Shriberg, E. E. (1994). *Preliminaries to a theory of speech disfluencies*. Ph.D. thesis, University of California.

Ventola, E. (1979). The Structure of Casual Conversation in English. *Journal of Pragmatics*, 3(3):267–298.

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). Elan: a professional framework for multimodality research. In *Proceedings of LREC*, volume 2006.

Yuan, J. and Liberman, M. (2008). Speaker identification on the SCOTUS corpus. *Journal of the Acoustical Society of America*, 123(5):3878.

Extending the Gold Standard for a Lexical Substitution Task: is it worth it?

Ludovic Tanguy (1), Cécile Fabre (1) , Laura Rivière (2)
(1) CLLE-ERSS: CNRS & University of Toulouse, France
(2) University of Toulouse, France
ludovic.tanguy@univ-tlse2.fr, cecile.fabre@univ-tlse2.fr, laurariviere12@gmail.com

Abstract

We present a new evaluation scheme for the lexical substitution task. Following (McCarthy and Navigli, 2007) we conducted an annotation task for French that mixes two datasets: in the first one, 300 sentences containing a target word (among 30 different) were submitted to annotators who were asked to provide substitutes. The second one contains the propositions of the systems that participated to the lexical substitution task based on the same data. The idea is first, to assess the capacity of the systems to provide good substitutes that would not have been proposed by the annotators and second, to measure the impact on the task evaluation of a new gold standard that incorporates these additional data. While (McCarthy and Navigli, 2009) have conducted a similar post hoc analysis, re-evaluation of the systems' performances has not been carried out to our knowledge. This experiment shows interesting differences between the two resulting datasets and gives insight on how automatically retrieved substitutes can provide complementary data to a lexical production task, without however a major impact on the evaluation of the systems.

Keywords: Lexical substitution, Gold Standard, Post hoc

1. Introduction

The lexical substitution task consists in providing the best substitute for a given word in a given context, usually the sentence. This is a crucial task in Natural Language Processing that requires systems to identify words that are semantically close to the target, and to select among candidates those that best fit the context. The task has been introduced in *SemEval-2007* by (McCarthy and Navigli, 2007) and involves annotators that are asked to provide substitutes for a single target word in context. This task has been reproduced with variations on the language (Cholakov et al., 2014; Fabre et al., 2014) or the size of the dataset (Kremer et al., 2014).

In this paper, we estimate the reliability of this setup by performing an additional annotation task for French, based on the results of the systems. We follow and extend (McCarthy and Navigli, 2009) who conducted a post hoc analysis to evaluate the reliability of the gold standard. The idea is to assess the capacity of the systems to provide good substitutes that would not have been proposed by the annotators. This question is particularly relevant for the lexical substitution task for two main reasons: first, finding a lexical substitute for a target word in the context of a sentence is a fairly difficult production task; second, systems that rely on huge corpora and lexical resources to detect semantic equivalents are likely to provide supplementary candidates that may be mistakenly ruled out by the evaluation procedure. Following (McCarthy and Navigli, 2009) we created a new annotation task mixing man-made and automatically-retrieved substitutes. But whereas (McCarthy and Navigli, 2009) simply evaluated the discrepancy between the two datasets, we go a step further and use this new source of data to evaluate the performance of the systems. This experiment shows interesting differences between the two resulting datasets. Yet it enables us to conclude that these differences have little impact on the evaluation of the systems.

2. The SemDis campaign: lexical substitution task in French

The SemDis 2014 evaluation campaign (Fabre et al., 2014), organized jointly by CLLE and IRIT research laboratories (University of Toulouse, France) was dedicated to a lexical substitution task in French, adapting the procedure defined for English by (McCarthy and Navigli, 2007). A dataset has been designed to evaluate and rank the systems that participated in the task, consisting of:

- 30 target words: 10 adjectives, 10 nouns and 10 verbs, whose frequency and degree of polysemy have been controlled;
- 300 sentences: 10 for each target word, selected from FRWaC (Baroni et al., 2009) and exemplifying different senses of the words;
- a gold standard made up of the substitutes that have been proposed by 7 annotators for each sentence. Each annotator could provide up to 3 answers per sentence.

For example, one of the target words is the noun *espace* (space), and one of the 10 target sentences for this word is: *No 208: Les sièges sont plus étroits, il y a moins d'espace entre les rangées.* (The seats are narrower, there is less space between the rows.)
Substitutes proposed by the annotators are: *distance* (distance), *place* (room), *écart* (gap), *espacement* (spacing), *volume* (volume).
Each system competing for this task could propose up to 10 substitutes per sentence. The first substitute in each list was considered the best candidate, but those that come next were not sorted in any particular order (see below for the details on evaluation measures).
This gold standard was used to evaluate the 9 participating systems and a simple baseline (which proposed the synonyms of the target words found in a dictionary, ordered

by decreasing frequency in a reference corpus). It is freely available for further use by the community[1].

3. Two stages of annotation

3.1. First version: annotation prior to runs

A total of 1,771 substitutes[2] were proposed by at least one annotator (see (Fabre et al., 2014) for the details on filtering and normalizing the substitutes). Inter-annotator agreement was measured with two techniques:

- *pairwise inter-annotator agreement*: 25.8%. This is the average rate of similar answers (proposed/not proposed) over every substitute in the dataset and for each pair of annotators;
- *mode inter-annotator agreement*: 73%. This is the rate of propositions which are the most common substitute (calculated for the 77% items for which a mode exists).

The base score for each substitute for this dataset was the number of different annotators who proposed it ($1 \leq score_1 \leq 7$). For example, the values for the substitutes of sentence 208 shown above are: *distance* (4), *place* (4), *écart* (2), *espacement* (2), *volume* (1).

3.2. Second version: post hoc annotation

It is common practice in Information Retrieval to use the results of the systems to build the set of documents that will be submitted to the annotators. The pooling method (Teufel, 2007) is a solution to the problem of non exhaustive relevance judgments. Following (McCarthy and Navigli, 2009), we adapted this principle to the task, by complementing the initial annotation dataset with all the substitutes provided by the systems: for each target sentence, every substitute either proposed by one of the participant systems, the baseline, or found in the first gold standard dataset has been evaluated. This gave a total of 13,089 candidate substitutes, among which 983 (7.5%) were in the gold standard and had been proposed by at least one system; 788 (6.0%) were in the gold standard only. The bulk (86.4%) of the candidate substitutes were proposed only by the systems, and thus were not taken into account in the first evaluation of the systems.

The 13,089 candidate substitutes were evaluated by 3 to 7 different anonymous annotators (with an average of 4.2 annotators per item). This annotation process was spanned over several months using an online survey platform. The annotators were contacted by word of mouth and had the sole constraint of being native French speakers; we did not control other variables (such as age and education). Each annotator could participate by evaluating any number of the 90 subsets in which these substitutes were dispatched in order to reduce the time spent on a single session. Dispatching the substitutes was pseudo-random, as we made sure that each subset for a given sentence contained at least one item from the initial gold standard, to prevent cases where the annotator was presented with only unsuitable substitutes. Evaluation itself required the annotator to choose a value on a scale ranging from 0 (inappropriate substitute) to 3 (perfect substitute). Annotators were instructed to focus on the meaning, and to be permissive of slight agrammaticalities induced by the substitution (inflection, elision, choice of preposition, etc.). Intermediate values were to be used for substitutes that were acceptable but induced a slight modification in the meaning of the sentence. Overall, 55,000 scoring decisions were made that could be exploited after filtering out inconsistent and incomplete answers.

The base score for each substitute in this dataset is the average score over the 3 to 7 annotators who rated it ($0 \leq score_2 \leq 3$). In the end, a total of 6,034 substitutes received a positive score, the average score being 0.51. Median score is 0, as 7,055 substitutes received an (unanimous) score of 0.

If we go back to sentence no 208, in addition to the 5 substitutes from the gold standard, 47 other candidates were found in the systems' propositions. In the end, 22 words received a positive score for this sentence. Here is a subset (words absent from the first gold standard are in italics): distance (3), place (3), espacement (3), écart (2.75), *écartement* (2.5), *intervalle* (2.5), *éloignement* (2), *interstice* (2), *marge* (1), *surface* (1), volume (0.75), *étendue* (0.75), *ouverture* (0.75), *air* (0.5), [...], *zone* (0.25), *an* (0), *atelier* (0), *attribut* (0), [...], *blanc* (0), *centre* (0), [...], *interligne* (0), *jardin* (0), [...] *univers* (0), *visa* (0)

Inter-annotator agreement was measured differently from the first dataset, because the annotated value is now a scale rating. We used:

- *rate of unanimous decision*: 0.56. 7,289 substitutes (out of 13,089) were given the exact same score by all annotators.
- *average standard deviation*: 0.38 ($\pm$ 0.01, 95% CI). Standard deviation is calculated independently on the annotators' rate for each substitute.

Although the two tasks clearly differ in their scope and nature (lexical production *vs* acceptability rating), it seems that the relevance of the collected data is on a par with what is expected when annotating lexical semantic phenomenons. In the next section we have a closer look at the differences between the two datasets.

This second dataset is also freely available[1].

4. Comparison of the two datasets

4.1. Quantitative differences

The two datasets agree on most items, as indicated by a high correlation coefficient measured between the two base scores. On the smaller dataset (first version, N=1,771), $\rho = 0.42$. On the larger dataset (N=13,089), $\rho = 0.61$ (considering a $score_1$ of 0 for substitutes absent from the first gold standard). More detail of this high correlation

[1] http://redac.univ-tlse2.fr/datasets/semdis-gold/lexicalsubstitution/

[2] The initial dataset contained 2,152 substitutes. However, we removed the multi-word substitutes as all the systems submitted single-word candidates. We also removed the data related to the adjective *compris* (understood/included), as most target sentences contained occurrences of the past participle instead of the adjective.

is indicated in the boxplots in Figure 1, where it appears clearly that substitutes initially proposed by 2 or more annotators systematically receive a very high score in the post hoc annotation.

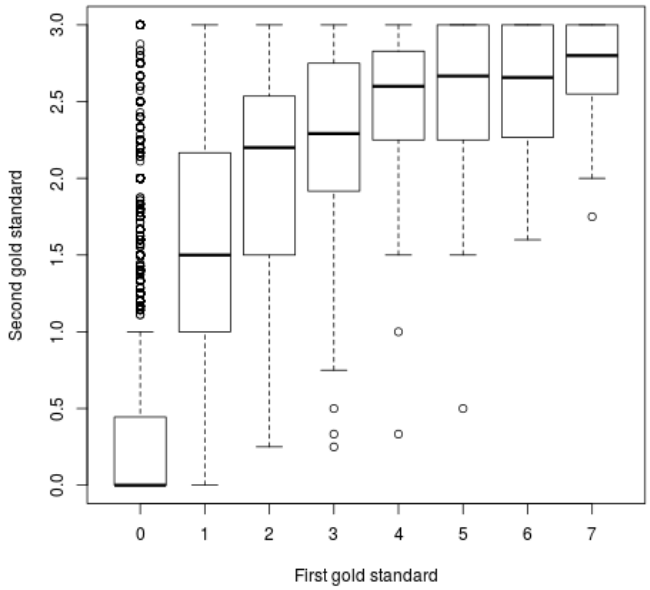


Figure 1: Correlation between the two annotations

Yet we notice contrast: as expected, the second dataset is much larger than the first one. More precisely, it contains a total of 4,336 new substitutes, which are candidate words with a positive score that did not appear in the first stage. Conversely, 73 substitutes from the initial gold standard (out of 1117) received a null score with the post hoc evaluation (i.e. no annotator considered they could be a possible substitute). In every such case, the substitute had an initial score of 1 (i.e. was proposed by only one annotator in the first dataset).

In Table 1 is shown the breakdown of the candidate substitutes based on their origin, i.e. whether they were proposed by the annotators, submitted by the competing systems (or baseline), or both. It appears clearly that the substitutes which were proposed both by the annotators and the systems get the highest scores (more than 78% with a score of 2 or more), even higher than the substitutes which were proposed by the annotators only.

$\mathbf{Score_2}$	Annotators only (788)	Systems only (11,318)	Annotators & systems (983)
$[0, 1[$	16%	80%	7%
$[1, 2[$	28%	14%	15%
$[2, 3[$	43%	6%	50%
$[3]$	13%	1%	28%

Table 1: Breakdown of substitutes across annotations

We also measured the approximate frequency for each substitute, based on the FrWac corpus, and found that, if in the first dataset the scores are positively correlated to the logarithm of the frequency ($\rho = 0.12$, $p < 0.01$) this is not true for the second ($\rho = 0.01$, $p > 0.05$). In addition, it appears that the substitutes which were added in the second dataset have a significantly lower frequency than those present in the first one (Wilcoxon test, $p < 0.01$).

4.2. Qualitative differences

Sentence 208 presented in Section 2. shows a typical example of the results obtained with the second annotation: as expected, only a few substitutes receive a very high score, most of which appearing in the first gold standard, while there is a long tail of words that were unanimously rejected by the annotators (corresponding to the very low box on the left of Figure 1). Some rejected candidates share with the target *space* the broad idea of location (*jardin* (garden), *univers* (universe)), others are related to a different sense than the one used in the sentence (such as *interligne* (line space)), while some are totally unrelated (*visa* (visa)). This illustrates the difficulty of the systems to capture the right level of semantic proximity or to perform disambiguation.

We had a closer look at words which were never proposed in the first annotation but got a high score in the second one (i.e. the outliers of the 0 value box on the left of figure 1). We identified several patterns. First, in some sentences, good substitutes are too numerous to be all proposed by annotators who focus on the most obvious synonyms within the limit of 3 answers. As the systems provide additional solutions the second task enables annotators to rate a larger set of substitutes. For example, considering again sentence 208, the word *écartement*, which is a morphological variant of the substitute *écart*, appears as a new valid substitute. In contrast, the annotators may find no answer for a given sentence because there is only one good substitute corresponding to a rare word (e.g. *diction* (diction) for *débit* (speech delivery)). The annotators are not able to produce this word, but they rate it highly. More generally, new substitutes often depart from common vocabulary: colloquial or formal words do not come to mind but they are considered acceptable in the rating task. This could be an explanation to the difference in frequencies measured across the two sets. Lastly, some new substitutes exhibit a looser semantic relation with the target word (e.g. hypernyms such as *progression* (progression) as a substitute for *montée* (climbing)).

These new substitutes are the most interesting contribution to this second stage: they illustrate the differences between the two tasks (producing substitutes or annotating candidate substitutes) and they are likely to have an impact on the evaluation of the systems.

5. Impact on the system rankings

In this last section we compare the overall ratings of each participant system on the two benchmarks.

5.1. Evaluation measures

Two scoring measures were initially used to assess the participants. In the formulae below, for sentence number i, G_i is the set of substitutes in the gold standard, P_i is the set of candidate substitutes proposed among which $best_i$ is the first one, and $score_i(c)$ is the score given to candidate c in the reference.

System	BEST (Gold1)	Rank	BEST (Gold2)	Rank	Rank difference
Proxteam_JDM_Syn	0.29	1	0.48	1	0
Proxteam_AxeParaProx_JDM_Syn	0.20	3	0.37	2	-1
CEA_LIST-word_cos_sent	0.23	2	0.33	3	+1
Alpage_WoDiS	0.17	4	0.29	4	0
CEA_LIST-fredist_cos_sent	0.12	7	0.25	5	-2
CEA_LIST-isc_cos_w2	0.12	8	0.22	6	-2
Proxteam_LM	0.15	5	0.18	7	+2
CEA_LIST-isc_cos_sent	0.11	9	0.18	8	-1
Baseline	0.13	6	0.17	9	+3
CEA_LIST-isc_l2_sent	0.03	10	0.09	10	0

Table 2: Normalized BEST scores and ranks for all systems as evaluated on both versions of the gold standard

System	OOT (Gold1)	Rank	OOT (Gold2)	Rank	Rank difference
Proxteam_JDM_Syn	0.41	1	0.38	1	0
Proxteam_AxeParaProx_JDM_Syn	0.37	2	0.35	2	0
CEA_LIST-isc_cos_sent	0.29	4	0.33	3	-1
CEA_LIST-isc_cos_w2	0.29	5	0.33	4	-1
Baseline	0.33	3	0.28	5	+2
CEA_LIST-fredist_cos_sent	0.24	6	0.23	6	0
CEA_LIST-isc_l2_sent	0.23	8	0.23	7	-1
Proxteam_LM	0.23	9	0.22	8	-1
CEA_LIST-word_cos_sent	0.24	7	0.19	9	+2
Alpage_WoDiS	0.22	10	0.19	10	0

Table 3: Normalized OOT scores and ranks for all systems as evaluated on both versions of the gold standard

- BEST: considers only the first proposal. The score is based on the raw gold standard score of the proposed substitute.

$$best(i) = \frac{score_i(best_i)}{\sum_{a \in G_i} score_i(a)} \quad (1)$$

- OOT (Out Of Ten): considers the 10 proposals without taking the order into consideration.

$$oot(i) = \frac{\sum_{a \in P_i} score_i(a)}{\sum_{a \in G_i} score_i(a)} \quad (2)$$

As can be seen, for both measures the scores are scaled by the sum scores of all substitutes for the target word. However, there are important differences from one target sentence to another, the number of substitutes with a positive score in the second gold standard varying from 5 to 56. It is thus impossible for a system to get a high score for a sentence with a large (> 10) number of suitable substitutes. For further evaluation, we use a normalized version of both evaluation measures, by dividing the measure value with the maximum expected value (i.e. the score obtained by a perfect system that proposes the 10 best substitutes in decreasing order of their gold standard score). We thus get a score that can reach a value of 1, meaning that it either proposed the highest rated substitute (BEST) or the 10 highest rated substitutes (OOT).

5.2. Comparison

We computed the scores for each initial participant submission and the baseline using both benchmarks, in order to measure the impact of the second annotation on the results of the substitution task. For details on the competing systems, please refer to (Fabre et al., 2014), (Desalle et al., 2014), (Ferret, 2014) and (Gábor, 2014).

Table 2 (resp. 3) gives the normalized BEST (resp. OOT) scores for all submitted participants with their scores according to both versions of the gold standards. As can be seen, for both evaluation measures there are little changes in terms of system ranking. Systems at both ends of the score range remain the same. The cases of bigger variations (up to three ranks) occur when the initial scores were very close to each others (e.g. for BEST there were 5 systems in the 0.11-0.15 range), this explains the relative changes.

6. Conclusion

The work presented here originated in a legitimate interrogation from the participants to the SemDis 2014 lexical substitution task. There was a possibility that the candidate systems proposed better (more accurate or more varied) substitutes than the ones proposed by the annotators. To answer this question, we had to evaluate all the proposed substitutes. It required a substantial annotation effort, given the sheer number of items and the need for a cross-evaluation as we wanted to get reliable data for such a difficult task. We finally opted for a pseudo crowdsourcing process in which all the annotators were contacted individually.

The main result of this work is of course the second gold standard dataset itself. Much more extended than the first, we measured that it is at least as reliable. It is an important added value to the test set itself, and we hope that it can still be of use for further experiments on lexical semantic

processing techniques for French.

The answer to the title question is twofold. As far as ranking the participants is concerned, the changes exist but are quite marginal: the best systems remain on top, and for the others the relative changes are mostly irrelevant. So the short answer is that this second annotation was not worth the effort, and we hope that this can be of use for future work on the development of such evaluation data.

However, the comparison of the two data sets gives us useful insights on the task itself, and helps us understand the gaps between the systems. By beginning to identify the main differences in terms of substitutes proposed by humans and NLP systems, we can complete the initial analysis proposed by (Tanguy et al., 2016) who found that there are also important differences in the difficulty encountered for specific target sentences.

7. Acknowledgements

We would like to thank the organizers of the SemDis 2014 task (Nabil Hathout, Lydia-Mai Ho-Dac, François Morlane-Hondère, Philippe Muller, Franck Sajous and Tim Van de Cruys); the participants (Yannick Chudy, Yann Desalle, Olivier Ferret, Kata Gábor, Bruno Gaume, Pierre Magistry and Emmanuel Navarro) and of course all the annotators who participated in the creation of the data sets.

8. Bibliographical References

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The Wacky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Language Resources and Evaluation*, 43(3):209–226.

Cholakov, K., Biemann, C., Eckle-Kohler, J., and Gurevych, I. (2014). Lexical substitution dataset for german. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 1406–1411, Reykjavik, Iceland.

Desalle, Y., Navarro, E., Chudy, Y., Magistry, P., and Gaume, B. (2014). BACANAL : Balades Aléatoires Courtes pour ANAlyses Lexicales : Application à la substitution lexicale. In *Proceedings of the TALN Conference (Traitement Automatique du Langage Naturel), SemDis workshop*, Marseille, France.

Fabre, C., Hathout, N., Ho-Dac, L.-M., Morlane-Hondère, F., Muller, P., Sajous, F., Tanguy, L., and Van de Cruys, T. (2014). Présentation de l'atelier SemDis 2014 : sémantique distributionnelle pour la substitution lexicale et l'exploration de corpus spécialisés. In *Proceedings of the TALN Conference (Traitement Automatique du Langage Naturel), SemDis workshop*, Marseille, France.

Ferret, O. (2014). Utiliser un modèle neuronal générique pour la substitution lexicale. In *Proceedings of the TALN Conference (Traitement Automatique du Langage Naturel), SemDis workshop*, Marseille, France.

Gábor, K. (2014). Le système WoDiS - WOlf & DIStributions pour la substitution lexicale. In *Proceedings of the TALN Conference (Traitement Automatique du Langage Naturel), SemDis workshop*, Marseille, France.

Kremer, G., Erk, K., Padó, S., and Thater, S. (2014). What Substitutes tell us - analysis of an" all-words" lexical substitution corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics , EACL*, pages 540–549, Gothenburg, Sweden.

McCarthy, D. and Navigli, R. (2007). Semeval-2007 task 10: English Lexical Substitution Task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53. Association for Computational Linguistics.

McCarthy, D. and Navigli, R. (2009). The English Lexical Substitution Task. *Language Resources and Evaluation*, 43(2):139–159.

Tanguy, L., Fabre, C., and Mercier, C. (2016). Analyse d'une tâche de substitution lexicale : quelles sont les sources de difficulté ? In *Proceedings of the TALN Conference (Traitement Automatique du Langage Naturel)*, Paris, France.

Teufel, S. (2007). An Overview of Evaluation Methods in TREC ad hoc Information Retrieval and TREC Question Answering. In L. Dybkjaer, et al., editors, *Evaluation of text and speech systems*, pages 163–186. Springer.

Lexical and Semantic Features for Cross-lingual Text Reuse Classification: an Experiment in English and Latin Paraphrases

Maria Moritz[1], David Steding[2]
[1]Institute of Computer Science, University of Göttingen, Germany
[2]Computer Science Faculty, University of Bremen, Germany
maria.moritz@stud.uni-goettingen.de, dsteding@uni-bremen.de

Abstract

Analyzing historical languages, such as Ancient Greek and Latin, is challenging. Such languages are often under-resourced and lack primary material for certain time periods. This prevents applying advanced natural-language processing (NLP) techniques and requires resorting to basic NLP not relying on machine learning. An important analysis is the discovery and classification of paraphrastic text reuse in historical languages. This reuse is often paraphrastic and challenges basic NLP techniques. Our goal is to improve the applicability of advanced NLP techniques on historical text reuse. We present an experiment of cross-applying classifiers—that we trained for paraphrase recognition on modern English text corpora—on historical texts. We analyze the impact of four different lexical and semantic features, on the resulting reuse-detection accuracy. We find out that—against initial conjecture—word embeddings can help to drastically improve accuracy if lexical features (such as the overlap of similar words) fail.

Keywords: cross-lingual classification, collocations, semantic features, word vectors

1. Introduction

Linguistic analyses, such as paraphrastic text-reuse discovery and classification, typically require advanced natural-language processing (NLP) techniques relying on machine learning. Yet, under-resourced historical languages, such as Ancient Greek and Latin, often lack enough primary material for certain time periods to properly train machine-learning models (a.k.a., classifiers). Consequently, only basic NLP techniques (e.g., checking similarity thresholds over string- and n-gram-shingles), which are independent from an advanced global knowledge or training experience, are applicable for analyzing historical corpora (Büchler et al., 2010). To improve the applicability of advanced NLP techniques, we need to experiment and systematically study the performance of such techniques on historical texts. Specifically, we must understand if—and in what way—ancient languages behave differently than contemporary languages when they are transferred and reused, especially when reused paraphrastically (where the reuse is not a literal copy of the source).

In this work, we focus on an important linguistic analysis—the detection and classification of paraphrastic text reuse in historical texts. We study a range of different machine-learning classifiers trained on modern and applied to ancient text-reuse excerpts. We analyze how the trained classifier models behave different or similar to each other for detecting paraphrastic text reuse.

We use two modern English text corpora—one containing about 2700 original-and-reuse pairs, the other containing 2600 compressed news articles (the so called "banner") together with their headlines—and a data set of Latin Bible reuse with around 1100 pairs of original and reused Bible verse.

We test whether three classifiers (K-Nearest Neighbor, Decision Tree, Support Vector Machine) relying on a handful of lexical features (e.g., no. of repeating words) and semantic features (e.g., word vector-based features) can correctly classify the Latin text excerpts as reuse when trained on the modern English data. By identifying features that positively affect the classification, we show their usefulness for cross-lingual reuse detection. The result of this study will help us to understand if cross-lingual application from modern to ancient is worth pursuing.

2. Related Work

Most research focuses on modern corpora. For example, Islam and Inkpen (2008) measure the semantic similarity of texts using corpus-based semantic word similarity and a modified version of the longest common substring algorithm. In contrast, the automated detection of historical text reuse is not thoroughly investigated yet. Büchler et al. (2013; 2010) combine basic NLP techniques for detecting reuse with overlapping features for historical texts using a fingerprinting approach (selecting n-grams from a pre-segmentized corpus).

Historical Text Reuse is studied by Lee (2007) who investigates reuse among the Gospels of the Bible's New Testament, aimed at aligning similar sentences. Similar to techniques used in the field of query expansion and retrieval, they develop so-called alternation patterns using the cosine similarity measure (a source-verse proximity measure) and the source-verse order. The field of paraphrastic reuse detection in historical texts is even more sparse. Bamman (2011) uses word-sense disambiguation. Utilizing a bilingual sense inventory, up to 72% of the word senses are classified correctly.

Cross-lingual Training was performed before. Rigutini et al. (2005) propose an algorithm based on expectation maximization. They train a classifier using a predefined category set and a labeled training data set for one language, followed by training a classifier for a different language on unlabeled documents, but using a translation of the label set. Wang et al. (2008) use labeled data from a related domain as auxiliary information to classify Wikipedia data in

a target domain. Showing the latent semantic relationship between two domains by co-clustering, they can propagate inter-domain labels, which capture common words and semantic concepts. Li et al. (2012) propose Topic Correlation Analysis, which enables the grouping of shared and domain-specific latent features. By inferring correlations between both groups, a mapping to domain-specific topics from different domains is established. The newly derived topics then open a shared feature space. Pan et al. (2008) use dimension reduction to apply transfer learning in a target domain with different feature distributions. They select a low-dimensional feature space, which minimizes the distribution distance in different domains. Via projection in related domains, standard learning is applied.

Cross-lingual training with Word Embeddings: Upadhyay et al. (2016) perform a comparative study by investigating four techniques for inducing cross-lingual embeddings on four different languages. The tasks described range from mono-lingual to cross-lingual similarity evaluation, and from rather "word-centered" semantic to more syntactic cases, where each requires a different degree of supervision. Upadhyay et al. show that models working on expensive cross-lingual knowledge often perform best.

Our work is motivated by the lack of studies on machine-learning approaches for historical text-reuse and the existence of promising results of cross-lingual training in the literature.

3. Study Design

We are interested to learn if we can cross-apply models for paraphrastic reuse detection from English paraphrases to Latin text reuse. Therefore, we identify text features that are language independent. We prepare a corpus with specific characteristics. Some of these characteristics (i.e., length reuse compared to original text, paraphrasic reuse) are similar to a Latin reuse corpus—our test set—and, hence, enables us with a comparable data base. Below, we define our research questions to formulate our work:

Overarchingly we ask: Is it possible to cross-apply classification models for non-literal reuse between modern and ancient-language text? We split this question into two more specific question that we address in this work:

RQ1: *What features support such a cross-lingual reuse classification?*

RQ2: *What characteristics should a source training text have to enable classification of the target language?*

3.1. Methodology

We first define three features (explained shortly) that i) are language independent, and ii) that we can directly infer from the text. We then train and test three classifiers on modern English data to create a baseline for the accuracy that could potentially be achieved when cross-applying each classifier. We finally cross-apply the generated models of English reuse to our Latin data set. In a second experiment (see 4.2.) we introduce a new feature based on word vectors and compare it against the results from the former experiment. This helps us to obtain a comprehensive view on current techniques and how they support our research.

3.2. Data sets

i) The Microsoft Research Paraphrase Corpus (henceforth MSRP) (Dolan and Brockett, 2005) consists of 5801 paraphrastic text-reuse pairs of modern English having a similar length. Out of these, about two thirds are considered semantically equivalent based on manual judgment. Positive and negative training examples are provided in the form of a training set and a test set.

ii) About 2600 instances of Antonio Gulli's English news articles corpus (henceforth Gulli's) collected in 2004 and 2005.[1] It contains XML-formatted data where the headline of a news article is associated with its banner—a very short summary (one sentence) of the article.

iii) Extracts from a total of twelve works and two work collections from the medieval Latin writer Bernard of Clairvaux (henceforth Bernard) who lived in the 12th century and reused text from the Bible. The latter was manually extracted by the Biblindex team (Mellerin, 2016) into a data set of over 1,100 reuse instances in alphabetical order. Every instance relates to a Bible verse. Typically, the reuse is about half as long as the verse. The Bible edition used to obtain the verses is the Biblia Sacra Juxta Vulgatam Versionem (Weber R., 1969 1994 2007). The works from Bernard where the texts are extracted from were published between 1957 and 2010 in the Sources Chrétiennes edition.

We tokenize all three data sets with the inbuilt tokenizer of NLTK (Bird and Loper, 2004) and lemmatize with TreeTagger's (Schmid, 1999) corresponding models for Latin and English. For Gulli's, we extract the text inside "<title></title>" and "<description></description >." We discard tokens from the title if they contain an opening or closing bracket, which often introduces the publisher name. In the description, we replace backslashes by white-spaces and discard dashes, because they indicate a topic-unrelated prefix (e.g., publisher place or newspaper name). Only if neither title nor banner exceed nine tokens and do not contain any character beyond the English alphabet, the exclamation mark, comma, full stop, and question mark, we use them for our experiment. Henceforth, we refer to a text snippet as "text1" and to its reuse as "text2."

3.3. Classification Procedure

Classifiers. We train a variety of well-known classifiers: a K-Nearest Neighbors (KNN), a Support Vector Machine (SVM), and a Decision Tree (DT). These are established approaches and proved successful for many text classification problems. The DT classifier is our own implementation using a maximum information gain metric to decide early on which feature the data has to be split. Our implementation uses discrete feature values only, hence our non-discrete features have to be discretized before they can be handed to the DT classifer. The SVM and KNN classifiers stem from the *sklearn* (Pedregosa et al., 2011) package.

Features. The classifiers are fed with feature values for both, negative and positive training examples. These feature values are calculated on positive reuse couples and negative reuse couples (i.e., two text excepts that are no reuse

[1] https://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

of each other). We use three features that can be quickly calculated and directly derived from the text—without requiring any extra resources or annotations. These are: the no. of words that text1 and text2 have in common, similar words that both texts have in common, as well as collocations that both texts have in common. The latter is defined as follows: How many words of text2 are collocations of words from text1, where collocations are considered within a three-word windows and collocations are only considered when they appear at least twice in the union of text1 and text2. Following, these features are listed and formally defined:

1. # words in common relative
2. # words in common .8 similarity relative
3. # 3-window collocations relative

Feature one $f1$ is defined as the relative amount of words that both text excerpts have in common:

$$f1(text1, text2) = \frac{|text1 \cap text2|}{min(|text1|, |text2|)} \quad (1)$$

Feature two $f2$ is defined as the relative amount of similar words that both text excerpts have in common:

$$f2(text1, text2) = \frac{|text2_{sim2_text1}|}{|text2|} \quad (2)$$

Where $|text2_{sim2_text1}|$ is the number of words form $text2$ that match with at least one word of $text1$ so that:

$$0.2 \geq \frac{edit(a, b)}{\frac{|a|+|b|}{2}} \quad (3)$$

Where $edit$ is the common edit distance by Levenshtein (Levenshtein, 1965).
Feature three $f3$ is defined as the relative amount of words from $text2$ that are collocations of the words from $text1$:

$$f3(text1, text2) = \frac{|text2_{sim3_text1}|}{|text2|} \quad (4)$$

Where $|text2_{sim3_text1}|$ determines the number of words from $text2$ that are collocations of any word from $text1$. Where collocations are calculated within $text1$ or $text2$ for each word of the corresponding text. Collocations are defined in a window of 3 with a maximum distance of 2 from a given word.
The classifiers calculate their models based on the feature values that are calculated form the examples. Examples with shorter reuse than source text may results in higher feature values, because the probability that all words form a short reuse are found in a longer source increases.

Sampling and Training. We use 10-fold-cross validation to train the classifiers. The ten complementing data parts for the training evaluation are generated randomly for each corpus, but every classifier gets the same input data. We train on 70% of the overall data sets and test on the remaining 30%. Note that to obtain data sets of negative examples (with similar sizes as those of the positive examples) for Gulli's and Bernard, we randomly generate links between any two text pairs.

4. Results

We first show the baseline performance of the classifiers on our corpora. Thereafter, we introduce a new feature, which we add to our feature set. We repeat the experiment afterwards.

4.1. Initial Experiment

We are interested in the classifiers' performance when trained on the modern and applied to the ancient Bernard corpus. Table 1 shows the precision where all implemented features are considered. It shows first how a trained model performs when applied to the test set of the data set it is trained on, and below, how it performs when applied to the Latin data set (Bernard). We see clearly that classifiers trained on Gulli's perform stable when their model is applied to the Latin data set. This can be explained by the fact that—even though a news headline is much shorter than its banner, both text excerpts do strongly overlap in their content words.
The models trained on MSRP show a comparable poor performance on the negative data when applied to the 30% test set of MSRP, which is partly caused by the comparably high ratio of positive examples compared to negative examples. Another reason is the characteristic of MSRP, which—serving as a benchmark for semantic equivalence—contains examples of close similarity, and its negative samples are surfacially very similar to the positive examples. (Finch et al., 2005) discuss some ambiguous characteristics of the MSRP corpus and give related examples. Thus, when two sentences are annotated for semantic equivalence that does not necessarily require them to be parahrasaes of each other. We take (Finch et al., 2005)'s examples to demonstrate this.

Example 1 (semantically equivalent following MSRP's annotators):

1. Amrozi accused his brother, whom he called "the witness", of deliberately distorting his evidence.

2. Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.

Example 2 (not semantically equivalent following MSRP's annotators):

1. Yucaipa owned Dominick's before selling the chain to Safeway in 1998 for $2.5 billion.

2. Yucaipa bought Dominick's in 1995 for $693 million and sold it to Safeway for $1.8 billion in 1998.

Example 3 (semantically equivalent following MSRP's annotators):

1. The stock rose $2.11, or about 11 percent, to close Friday at $21.51 on the New York Stock Exchange.

2. PG&E Corp. shares jumped $1.63 or 8 percent to $21.03 on the New York Stock Exchange on Friday.

train	test	precision	recall	fscore	precision	recall	fscore	accuracy		accuracy_new
KNN		**positive**			**negative**					
MSRP	MSRP	.74	.68	.71	.42	.50	.46	**.62**		**.65**
MSRP	Bernard	.62	.45	.53	.58	.73	.65	**.60**		**.68**
Gulli's	Gulli's	.83	.81	.82	.83	.85	.84	**.83**		**.85**
Gulli's	Bernard	.82	.82	.82	.83	.83	.83	**.82**		**.84**
DT		**positive**			**negative**					
MSRP	MSRP	.72	.86	.78	.50	.29	.37	**.68**		**.68**
MSRP	Bernard	.49	1.0	.66	-	0.0	-	**.49**		**.65**
Gulli's	Gulli's	.88	.82	.85	.84	.90	.87	**.86**		**.85**
Gulli's	Bernard	.86	.34	.48	.59	.94	.73	**.64**		**.78**
SVM		**positive**			**negative**					
MSRP	MSRP	.72	.94	.81	.62	.21	.31	**.71**		**.71**
MSRP	Bernard	.96	.51	.67	.67	.98	.80	**.75**		**.76**
Gulli's	Gulli's	.87	.84	.86	.86	.88	.87	**.86**		**.88**
Gulli's	Bernard	.87	.83	.86	.84	.90	.87	**.86**		**.87**

Table 1: Performance of the classifiers K-Nearest Neighbors, Decision Tree and Support Vector Machine showing precision, recall, fscore, accuracy, and accuracy of the new feature accuracy_new

Especially using the SVM classifier, precision is high for the positive test sample when the MSRP model is applied to Bernard, but recall is low. This again can be explained by the fact that some similar texts which are marked as negative text reuse in MSRP would be marked as positive text reuse examples in the Bernard data set.

The DT classifier performs particularly bad especially for the cross-application task. This is intuitive considering the comparably primitive model behind these types of classifiers. During discretizing our DT implementation maps the feature values to 50 different intervals. When trained on MSRP and Gulli's the DT classifier prefers a feature which's values enable the soonest and highest information gain. From 1 we can see that a feature that is significant for negative reuse, however, is not a good choice for the negative reuse in Bernard's data (84% vs. 59% precision on the negative data set). The SVM classifier treats feature values better in that respect that it creates a hyper plain equally based on all features.

In the following, we add a new feature to our experiment: We calculate a normalized context vector for each side of each reuse pair and add the angle between the two text excerpts as an extra feature.

4.2. Adding the Angle between the Context Vector as Feature

We now show how a new feature affects the accuracy in our experiment. We use normalized word vectors that represent context information for each word of a text excerpt. The angle between two vectors representing one text excerpt each serves as a new feature for reuse classification. To conceive word vectors for the English data, we use the pre-trained word vectors from GloVe (c.f., Pennington et al. (2014)), which are calculated on a dump of the English Wikipedia in 2014 and the Gigaword5 (Robert Parker, 2011) corpus. To conceive word vectors for our Latin data set, we pretrain vectors on the corpus from the Latin Library (Johnson, 2014) of the CLTK (Johnson et al., 2014 2016). We determine the new feature from the positive and negative training set for Gulli's, MSRP and Bernard. This new feature $f4$ is defined as the cosine of the angle between the averaged word vectors of $text1$ and $text2$. Those averaged vectors are defined as vec_{text1} and vec_{text2}:

$$vec_{text1} = \frac{\sum_{i=0}^{|text1|} v_{w_i}}{|text1|} \tag{5}$$

$$vec_{text2} = \frac{\sum_{j=0}^{|text2|} v_{w_j}}{|text2|} \tag{6}$$

Where v_w is the word embedding vector of a running word in $text1$ and $text2$ respectively.

The last column of Table 1 shows the results for this experiment. Nearly every classifier model on every data set shows an increase in accuracy. Especially for classifiers with a less complex model, the new feature causes a huge accuracy gain applied to both, the data it is trained on and the new target data set of the cross-lingual application task.

4.3. Discussion

RQ1: We learn from the experiments that lexical features can serve well for classification in a cross-lingual task, and that semantic characteristics, such as those that can be derived from word embeddings, support the identification of paraphrastic reuse. One should, however, be aware that—for our features to be suitable for the task—our training and testing data share common characteristics, i.e., texts behave similar in their surface and semantic features.

RQ2: When the training text is composed similarly to the testing text, a well-working cross-lingual classification can be achieved. If semantic equivalence is defined on a more granular level—as it its the case in the MSRP training data—recall and precision scores tend to excel each other widely. Further, MSRP corpus data differ from the other two data sets in the length of the text excerpts of a reuse pair. In the MSRP reuse data, the lengths are largely equal, as opposed to the other data sets. The reuse (text2) in Gulli's and Bernard corpus is about half as long as text1 and often contains words that are repeated from the former text

or slightly modified. This is another characteristic that explains why especially the more advanced classifier methods work better when trained on Gulli's and applied to Bernard. Summarizing, it can be useful to consider the advantage of available, modern text corpora for a learning task on historical text if the properties for which a classifier shall be trained remain comparable.

5. Conclusion

We presented a feasibility study of classification for cross-lingual training. Our study shows that the approach under a simple feature selection (based on shared similar words and collocations) can perform well with an accuracy of up to 86%, and even higher for models trained additionally on a new feature that is determined by the angle between the normalized word vectors of a reuse pair. We found that especially for less advanced classifiers, this new feature drastically improves the accuracy. We showed that it is valuable to consider using modern resources in a classification task for historical languages when the investigated data sets share similar features, such as structural characteristics.

Acknowledgments

Our work is funded by the German Federal Ministry of Education and Research (grant 01UG1509).

Bibliographical References

Bamman, D. and Crane, G. (2011). Measuring historical word sense variation. In *Proceedings of the 11th ACM/IEEE-CS Joint Conference on Digital libraries (JCDL 2011)*, pages 1–10. ACM Digital Library.

Bird, S. and Loper, E. (2004). Nltk: The natural language toolkit. In *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, ACLdemo '04. Association for Computational Linguistics.

Büchler, M., Geßner, A., Eckart, T., and Heyer, G. (2010). Unsupervised detection and visualisation of textual reuse on ancient greek texts. *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*, 1(2).

Büchler, M. (2013). *Informationstechnische Aspekte des Historical Text Re-use (English: Computational Aspects of Historical Text Re-use*. Ph.D. thesis, Leipzig University, Germany.

Dolan, B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*. Asia Federation of Natural Language Processing.

Finch, A., Hwang, Y.-S., and Sumita, E. (2005). Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Proceedings of the Third International Workshop on Paraphrasing*, pages 17–24. Association for Computational Linguistics.

Islam, A. and Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data*, 2(2):10:1–10:25.

Johnson, K. P., Burns, P. J., Hollis, L., Pozzi, M., Shilo, A., Margheim, S., Badger, G., and Bell, E. (2014–2016). Cltk: The classical language toolkit. `https://github.com/cltk/cltk`. DOI 10.5281/zenodo.44555 v0.1.32.

Johnson, K. P. (2014). Cltk latin library. `https://github.com/cltk/latin_text_latin_library`. Acsessed Aug. 2017.

Lee, J. (2007). A computational model of text reuse in ancient literary texts. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic*, pages 472–479. Association for Computational Linguistics.

Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848. (1966) Russisch, Englische Übersetzung. In: Soviet Physics Doklady Vol. 10, No. 8: 707–710.

Li, L., Jin, X., and Long, M. (2012). Topic correlation analysis for cross-domain text classification.

Mellerin, L. (2016). Biblindex. `http://www.biblindex.mom.fr/`.

Pan, S. J., Kwok, J. T., and Yang, Q. (2008). Transfer learning via dimensionality reduction. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, pages 677–682. AAAI Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Rigutini, L., Maggini, M., and Liu, B. (2005). An em based training algorithm for cross-language text categorization. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '05, pages 529–535, Washington, DC, USA. IEEE Computer Society.

Robert Parker, David Graff, J. K. K. C. K. M. (2011). English gigaword fifth edition. ldc2011t07. dvd.

Schmid, H. (1999). Improvements in part-of-speech tagging with an application to german. In *Natural language processing using very large corpora*, pages 13–25. Springer.

Upadhyay, S., Faruqui, M., Dyer, C., and Roth, D. (2016). Cross-lingual models of word embeddings: An empirical comparison. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1661–1670, Berlin, Germany, August. Association for Computational Linguistics.

Wang, P., Domeniconi, C., and Hu, J. (2008). Using wikipedia for co-clustering based cross-domain text classification. In *2008 Eighth IEEE International Conference on Data Mining*, pages 1085–1090, Dec.

Gribomont J. Weber R., Fischer B., editor. (1969, 1994, 2007). *Biblia sacra juxta vulgatam versionem*. Deutsche Bibelgesellschaft.

Investigating the Influence of Bilingual Multiword Units on Trainee Translation Quality

Yu Yuan[†] [‡], **Serge Sharoff**[†]
[‡] School of Languages & Cultures, Nanjing University of Information Science & Technology, 210044, China
[†] Centre for Translation Studies, University of Leeds, LS2 9JT, United Kingdom
hittle.yuan@gmail.com, s.sharoff@leeds.ac.uk

Abstract

We applied a method for automatic extraction of bilingual multiword units (BMWUs) from a parallel corpus in order to investigate their contribution to translation quality in terms of adequacy and fluency. Our statistical analysis is based on generalized additive modelling. It has been shown that normalised BMWU ratios can be useful for estimating human translation quality. The normalized alignment ratios for BMWUs longer than two words have the greatest impact on measuring translation quality. It is also found that the alignment ratio for longer BMWUs is statistically closer to adequacy than to fluency.

Keywords: Bilingual multiword unit, normalised alignment ratio, translation quality estimation, trainee translators

1. Introduction

The overall goal of this study is to investigate automatic quality assessment for translators, especially for the trainees. Many problems related to quality stem from inappropriate translation of MWUs, such as idioms (e.g. *barking at the wrong tree*, *add insult to injury*, etc.), terms (e.g. *recursive function*, *closed captioning*, etc.), phrasal verbs (e.g. *fall out*, *give up*, etc.), or named entities (e.g. *Vice-Chancellor Sir Timothy Michael Martin O'Shea*, *New York City*, etc.). Baker (2011, pp:72-73) identified four types of errors when translating idiomatic expressions: 'no equivalent', 'similar counterpart but different context', 'idiom play', 'different conventions of using idioms'. A trainee translator in our dataset produced an unacceptable calque of *as brave as a lion* into Chinese 勇猛如狮, while the correct idiom should have been 勇猛如虎 'as brave as a tiger'. This erroneous treatment is what Baker terms 'different conventions of using idioms'.

Based on observations of source language (SL) idioms and their Persian translation, researhers have found that explicit loss,implicit loss, modified loss and complete loss are common resultant categories of cultural losses (Zebardast and AbuSaeedi, 2015; Zebardast, 2015). Translation of MWUs is problematic because it is associated with cultural issues (Min, 2007), systematic language variation (Wang and Nian, 2004) and the translator's failure to decipher the meanings of MWUs in question (Abu-Ssaydeh, 2004). Therefore, this paper assesses the contribution of translating MWUs to the overall translation quality, particularly for trainee translations.

In this paper we investigate:

1. how can we extract BMWUs for trainee translation quality evaluation?
2. how is the use of BMWUs related to the quality of trainee translations?
3. How do BMWUs of different lengths contribute to the trainee translation quality?

The term BMWUs refers to recurrent sequences of words that are translations to each other in aligned bilingual texts. In other studies they are also known as (bilingual) phrase alignments, aligned phrases units (sequences), bilingual alignments, bilingual phraseological units, bilingual phraseology, etc. In this paper, we use the term bilingual multiword units (BMWUs) hereinafter.

We start with the automatic acquisition of BMWUs from parallel corpora and use them as base against which to query and compare trainee translations. Our main contribution is designing and using BMWU-related features as quality indicators for human translation quality estimation.

2. Methodology

In this section, we will describe how BMWUs are used in this study, the method we use to extract BMWUs and our exploration of the relationship between BMWUs of different lengths and translation quality in terms of fluency and adequacy using the mixed-effect modelling.

2.1. Extraction of BMWUs

As we work with the parallel corpora we can combine the task of identification of monolingual MWUs with the process of bilingual alignment, saving us from the trouble of a difficult task of monolingual MWU identification (Sag et al., 2002). The alignment process which is aimed at producing phrase tables for statistical machine translation (SMT) (Koehn et al., 2003) can be based on flat models or on hierarchical models. In traditionally used flat IBM family models, the phrase tables are generated in two steps, first generating word alignments and then extracting a scored table of phrase pairs. However, this often yields a large proportion of unwanted word alignments, as there are only minimal phrases memorized by the model(DeNero and Klein, 2008), so it has to be combined with heuristic phrase extraction to exhaustively combine adjacent phrases permitted by the word alignment (Och et al., 1999).

In contrast, Bayesian-based phrase alignment as proposed in (Neubig et al., 2011) is a model for joint phrase alignment and extraction using non-parametric Bayesian methods and inversion transduction grammars (ITGs). A hierarchical ITG model relies on the Pitman-Yor process (Pitman and Yor, 1997) to directly use probabilities of the model as

Table 1: Professional Translations for the Same Source Text Phrase

Source	Translation	Frequency
总而言之	in sum	4
	in summary	4
	all in all	3
	in short	5
	in conclusion	2
	in general	4
	overall	4
	$--$ (omitted)	2

a replacement for the phrase table generated by heuristic techniques, e.g. intersection, grow-diag in Giza ++ (Och and Ney, 2003). Because of its compactness and competitive accuracy, we choose this method over other standard heuristic alignment tools (e.g. Giza ++, fast_align (Dyer et al., 2013), etc.) to obtain an aligned list of MWUs.

Our study is mainly focused on investigating the contribution of BMWUs of varying lengths to the translation quality of trainee translations. For this task, we need an authentic database of BMWUs from a sizeable bilingual corpus of professional translations to have enough statistics for MWU identification and pruning. The extracted BMWUs will be used to measure the degree of adequacy and fluency of trainee translations. If professional translators view language expressions that can transfer meaning unambiguously as the basic translation units (Baobao et al., 2002), we believe, such a database of BMWUs can be a useful resource for human translation quality estimation. Investigation of BMWUs of different lengths can be viewed as part of the feature engineering for human quality estimation task.

It is often observed that human translators translate group of words as a whole and words are rarely treated as the working translation units individually. Variation in translations from a large corpus and distribution of frequencies, as illustrated in Table 1[1] will lead to varying probabilities of the aligned BMWUs. Though the list of candidate translations is not exhaustive, if translations by students are not in this list, they are more likely to be inappropriate translations.

Therefore, our working hypothesis is that the consistency of BMWUs in trainee translations in relation to professional translations can be interpreted as a higher degree of semantic adequacy and stylistic fluency. In the following experiment, we will use the Bayesian-based ITG method to automatically extract phrasal alignments of different lengths (1-4 words) from a large parallel corpus, and then compute the normalized ratios of these aligned phrasal sequences for each trainee translation at the document level. Here is how the normalized ratio of BMU is calculated:

$$\boldsymbol{R}_{norm} = \frac{Count_{trg} * Len_{srt}}{C_{[100]}} \quad (1)$$

where R_{norm} is the normalized ratio of BMWUs in proportion to the length of source text (Len_{srt}) in terms of the number of tokens, and $Count_{trg}$ is the count of BMWUs in the target text, with $C_{[100]}$ a constant number serving as the normalization base [2]. Note that in our calculation the ratio is computed in relation to the source text length. Therefore, the returned ratio is equivalent to the recall of BMWUs. This implies we can also compute the precision of BMWUs in relation to the target text length.

2.2. Multilevel Mixed Effects Modelling

Multilevel mixed effects modelling is a way of understanding the types of relationships you can examine or consider of your data. This technique of data analysis is especially useful when observations at one level of analysis are nested within observations at another, and when other categorical or hierarchical data types are involved. In our case, we are investigating how normalized ratios of BMWUs of varying lengths contribute to translation adequacy and fluency. Thus, our data (normalized ratios of BMWUs) are nested in a different category, e.g., BMWU length.

It has advantages of analysing phenomena at multiple levels simultaneously and identifying important relationships across different levels of analysis. The recent development of statistical modelling has made it feasible to analyse group and individual variance simultaneously. This modelling is based on a generalized linear model with a linear predictor involving a sum of smooth functions of covariates (Wood, 2017, pp:161). In general, the model has the form :

$$g(\mu_i) = \mathbf{A}_i\gamma + \sum\nolimits_i f_j(x_{ji}), y_i \sim \mathbf{EF}_{(\mu_i,\phi)} \quad (2)$$

where $\mathbf{A}_i$ is the i^{th} row of a parametric model matrix, with corresponding parameters γ, f_j is a smooth function of covariate x_j, and $\mathbf{EF}(\mu_i, \phi)$ denotes an exponential family distribution with mean μ_i and scale parameter ϕ (Wood, 2017, pp:249). This modelling technique allows for the flexibility and convenience of specifying the model in terms of smooth functions, which can then be estimated form data using cross validation or marginal likelihood maximization.

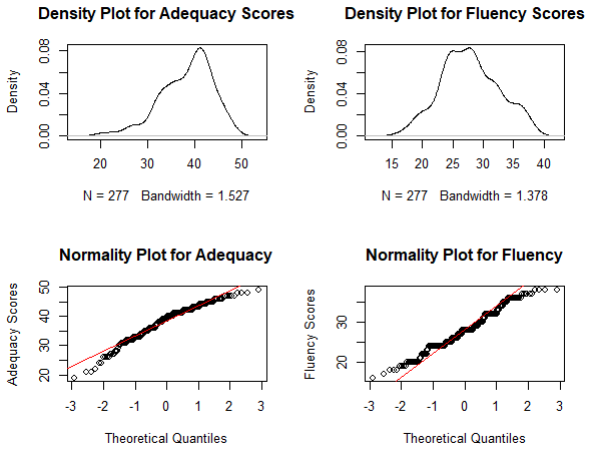
In this study, a mixed-effect modelling is carried out, aiming to explore the significance of BMWUs accounting for students' translation scores in adequacy and fluency and how BMWU alignments of different lengths impact the quality scores, when there are multiple levels of translation quality and alignment lengths are involved.

The response variables in our study are the adequacy and fluency scores, and the explanatory variables in our experiment are the normalized alignment ratios (see Equation (1)), coded as NrmALR, and MWU length information, coded as AL1, AL2, AL3, in the translations. Meanwhile, in order to answer the questions listed above, we are also interested in knowing the interaction between the BMWU length and their normalized ratios. Thus, in our design, we keep these two as the fixed effects and the training corpus size (two sizes) (coded as TrCorpS1, TrCorpS2) and each individual translation as the random effects.

[1] Generated from http://www.linguee.com

[2] The setting of the normalization base depends on the average length of the trainee translations, and in our case they are around $2 - 400$ words. Hence, we set 100 as the base for normalization.

Figure 1: Normality Test for Adequacy and Fluency Scores



Both adequacy and fluency scores are not normally distributed (Shapiro-Wilk normality tests $p < .05$), as shown in Figure 1, Adequacy scores are left-skewed due to central tendency towards upper bound (more scores above the third quartile), and Fluency scores are short tailed, indicating very small variances between them.

To proceed, we converted the adequacy scores and fluency scores into ordered categorical data (Liu and Agresti, 2005) so that it fits the condition that the expected value of the response variable (e.g. adequacy, fluency) follows a logistic distribution. Analysis was performed in the R package mgcv[3] for generalized additive mixed modelling.

We treat the normalized alignment ratio and phrase alignment length as main effects and several combinations of random effects of training corpora and sample IDs, using the ordered categorical data distribution family. This family of method is for use with generalized additive model, implementing regression for data following a logistic distribution. The observed categories are coded $1, 2, 3 \ldots$, up to the number of categories (Wood et al., 2016).

3. Parallel Corpus and the Trainee Data

For this study we use the English Chinese parallel UM corpus of mixed domains (Tian et al., 2014). It is a multi-domain and balanced parallel corpus covering several topics and text genres, including education, law, microblogs, news, science, spoken, subtitles and theses. The English part is tokenised with the scripts included within the Statistical Machine Translation system moses (Koehn et al., 2007). The Chinese part is segmented with Jieba Chinese word segmentation module.[4]

As for trainee translations, we have 277 student translations in six different domains scored by two raters in terms of their adequacy and fluency on a scale of 60 points (mean=38.23, interquartile range=7, range=18) for content adequacy and 40 points (mean= 27.84, interquartile range=8, range=22) for language fluency, so that the total score of a student translation can be in the [0-100] range (Table 3).

Table 2: Statistics of UM Parallel Corpora

Domains	Languages	Tokens	Average Length	Vocabularies	Sentences
News	English	8,646,174	19.21	274,546	45,000
	Chinese	15,277,414	33.95	47,902	
Spoken	English	1,836,670	8.35	107,923	220,000
	Chinese	3,033,052	13.79	9,011	
Laws	English	5,926,316	26.94	66,330	220,000
	Chinese	8,783,941	39.93	14,723	
Thesis	English	5,962,590	19.88	378,679	300,000
	Chinese	10,514,430	35.05	149,110	
Education	English	8,401,095	18.67	293,595	450,000
	Chinese	13,749,570	30.56	38,663	
Science	English	598,050	2.22	115,968	270,000
	Chinese	1,527,849	5.66	8,927	
Subtitles	English	2,299,742	7.67	101,423	300,000
	Chinese	3,818,490	12.73	13,854	
Microblog	English	72,144	14.43	12,083	5,000
	Chinese	125,415	25.08	3,525	
Total	English	33,742,781	13.29	832,518	2,215,000
	Chinese	56,830,161	22.51	209,729	

Table 3: Basic Statistics of English-Chinese Translational Data

Source Text	Domain	Topic	Statistics			
			Source Text		Translation	
			# of Sentences	# of words	# of sentences (mean)	# of words (mean)
ST1	Science fiction	Insects	11	261	10	317
ST2	Scoial life	Marriage	15	259	14	311
ST3	Sports	Walking	13	289	12	353
ST4	Short story	Preseverance	15	313	14	410
ST5	Literature	Essayist	5	229	4	246
ST6	Science	xenotransplantation	13	266	11	372

Table 4: Range Finders for Different Grades of Translation

Grades	Usefulness/transfer	Terminology/style	Idiomatic Writing	Target Mechanics
Standard	29-35	21-25	21-25	13-15
Strong	22-28	16-20	16-20	10-12
Acceptable	15-21	11-15	11-15	7-9
Deficient	8-14	6-10	6-10	4-6
Minimal	1-7	1-5	1-5	1-3

Two Chinese native annotators, both are PhD students in Translation Studies, following the scoring scheme of ATA Certification Programme Rubric for Grading (Version 2011),[5] measure the performance of a translator against four dimensions ranging from Content Transfer (CT), terminology (T), idiomatic writing (I) and target language conventions (TC), see Table 4. They evaluated the translations based on the degree to which learner translators have transferred the meaning completely (combining CT and T into the Adequacy score) and followed the rules and conventions of the target language (I and TC combined into the Fluency score). The inter-annotator agreement is substantial (Krippendorff's $\alpha = .77$ for Adequacy and .89 for Fluency).

[3] https://cran.r-project.org/web/packages/mgcv/index.html

[4] https://github.com/fxsjy/jieba

[5] http://www.atanet.org/certification/aboutexams_rubic.pdf

Table 5: Length and Type Distribution of BMWUs

Length	Counts	Alignment Types	Counts
1-word	648,611	one-to-one	374,840
2-word	1,835,261	one-to-many	2,73,771
3-word	1,889,009	many-to-one	362,849
4-word	1,344,276	many-to-many	4,705,697
Total	5,717,157	Total	5,717,157

Table 6: BMWU Alignment Accuracy (threshold DTP >= 0.2)

Alignment	Top1000(%)	Bottom1000(%)
BMWU(Single word)	96.3	26.9
BMWU(2-more words)	98.7	97.2

Table 7: An Excerpt of Aligned BP Extracted

BP(source)	BP(target)	DTP	ITP
tax credits	税额 减免	0.37	0.25
	税款 抵减	0.52	0.34
	税收 减免	0.08	0.05
	的 税收 优惠	0.90	0.05
	税收 优惠	0.05	0.05
	税额 抵免	1.00	0.05
	税收 抵免	0.59	0.17
and tax credits	或 减免	0.50	0.50
	和 税 减免	1.00	0.50
as tax credits	诸如 税收 扣减	1.00	1.00
continuous tax credits	持续性 税收 扣除	1.00	1.00
for tax credits	享受 税额 减免	1.00	1.00
investment tax credits	投资 税额 减免	1.00	1.00
in tax credits	税收 抵免	0.02	1.00
production tax credits	生产 税额 减免	0.33	1.00

4. Findings and Discussion

4.1. Considerations of BMWU Alignment Quality

We extracted BMWUs of a maximum length of 4. The reason we did not go up to longer sequences of phrases is that longer alignments (> 4 words) are very rare in students' translations. We include one word MWUs to account for one-to-many and many-to-one alignments, i.e. some MWUs are produced as translations of a single source word, and single words are translations of source MWUs. We eventually have 9.63 million pairs of phrasal alignments without exclusion and 5.72 million pairs after filtering per direct translation probability (DTP) and inverse translation probability (ITP), the selected thresholds of DTP and ITP $>= 0.01$, See Table 5.

Longer MWU alignments from the training corpus are generally more accurate than shorter ones. Accuracy of BMWU alignments remain stable even though their direct translation probabilities decrease. In order to evaluate the validity of entries in the extract list of BMWUs, we set the DTP threshold for all alignments to be 0.01 and then sort them per their DTP values and discard all non-English-Chinese pairs, i.e., Null alignments, punctuations, symbols, strings alignments and English-English alignments. That is to say, our evaluation is based on English-Chinese alignment pairs only. We compared the top 1000 and bottom 1000 entries for single word alignments and phrasal alignments (BMWUs of 2 to more words). See Table 6 for details. We believe that the reason one-word alignments contain many false matches is because of word segmentation, e.g. 无用功 ('unproductive work') should be matched by two English words to be equivalent, and word associations in the context, e.g. 拖垮 ('drag down') should be the cause of lack of productivity and it is supposed to occur within the near context. Nevertheless, as Table 6 shows, these BMWUs, particularly those longer than one word, when selected as bilingual correspondences with reasonably good accuracy, can be readily usable.

In preparing the dictionary of bilingual phrase alignments for query their occurrences in trainees' translations, four-word alignments are eventually discarded because their extremely low frequencies in the target translations. Analysis of the phrase alignment dictionary will be presented in the final paper. Note that while DTP may be a useful signal of alignment certainty we cannot take it for granted that lower probabilities nullify the legitimate translation equivalents, as many pairs with very low probabilities are valid translations to each other, for instance 报价和目录 ('quote and directory') and *quotation and catalogues* are aligned at a probability of 0.08 but they are clearly valid alignments. This again explains why in Table 6 there is no significant difference for longer BMWUs ranked by DTPs (top and bottom). Also this shows that we need keep good coverage of BMWUs. For this study, we set the threshold of direct translation probability at 0.02. This cutting-off value eventually allows us to have 3.5 millions pairs of bilingual phraseological pairs, a much slimmer list of phrase table, as illustrated in Table 7. In the Table, the third and fourth column are direct (conditional) translation probabilities (DTP, i.e. translation probability from English to Chinese) and inverse translation probabilities (ITP, i.e. translation probability from Chinese to English).

With this acquired probabilistic dictionary of BMWUs, each 1- word, 2-word and 3-word lexical unit in the TT is then queried against each trainee translation to find the matches between the corresponding units. When calculating the translation probability of a TT, the parallel sentences of ST and TT are taken as a unit, matched against the bilingual lexicon trained from last step and the results is the logarithm of all matched translation probabilities in each translation. Normalized BMWU ratios of varying lengths are calculated as per Equation (1). The rationale for this score is these four indexes come from the conception that good translations are often close to professional or expert translations and therefore direct translation probabilities trained from large parallel corpora of professional translations can be a good criterion for the word choices by translators. True alignments of higher probabilities are more likely to be the right candidates and in this way, for a translation, Therefore, there would be variation of the normalized phrase alignment ratios across translations of different quality. Better translations would have higher normalized ratios of aligned BMWUs than inferior ones. For the later modelling, translations are divided into three levels of quality groups, i.e. poor ([39, 54.7]), average ([54.7, 70.3]) and good ([70.3, 86]) per their final scores on ade-

Table 8: Illustration of Automatically Aligned Phrases(1-3 words)

Phrase Length	English Phrases	Chinese Phrases	DTP	Gloss
1-word	offences	所犯	0.33	committed by
		犯罪行为	0.29	crimes
		而犯	0.44	commit
		系由该	1.00	This is committed by
		犯罪	0.22	commit a crime
2-word	traffic offences	违犯 交通法规	0.91	violation of traffic regulations
		交通 罪行 的	1.00	traffic offence (about)
		交通 罪行	1.00	traffic offence
3-word	international drug trafficking	国际 贩毒	0.39	international drug trafficking
		国际 毒品 贩运	0.86	international drug trafficking
		国际 药物 贩运	1.00	international medicine trafficking

quacy and fluency. The BMWU ratio for each translation is further divided into three length groups, i.e. one, two and three word alignment ratio.

4.2. Mixed Effects Modelling

A model is eventually selected per Akaike's Information Criterion (AIC=4287.18) using a restricted maximum likelihood as our final model for adequacy. Results are reported in Table 9 below. The random effect of interaction between specific sample and the normalized alignment ratio (NrmALR) is also significant (${\chi_{(246.7,276)}}^2 = 2214$, $p < .0001$). The model has revealed that longer BMWUs contribute more to the adequacy scores

Table 9: Fixed Effects of Alignment Ratio and Alignment Length on Adequacy

Response Variable	Effect	Estimate	SD	Z	$Pr. > t$ or Z	AIC
Adequacy	Intercept	-4.91	0.53	-9.14	< 0.0001	4287.18
	NrmALR	4.35	0.39	11.15	< 0.0001	
	AL2	3.19	0.69	4.64	< 0.0001	
	AL3	6.52	0.73	8.82	< 0.0001	
	TrCorpS2	-0.67	0.09	-7.70	< 0.0001	
	NrmALR:AL2	5.04	0.48	10.59	< 0.0001	
	NrmALR:AL3	11.22	0.75	14.93	< 0.0001	

The selected model tells that, ignoring other variables, the normalized BMWU alignment ratio has a significant impact on the adequacy scores (${\chi_{(2)}}^2 = 314.37$, $p < .0001$; $z = 11.15$, $p < 0.0001$), and similarly, three word alignments (AL3) and two word alignments (AL2), in contrast to one word, have a significant influence on adequacy scores. In addition, there is a significant interaction between two word alignment (AL2) and the three word alignment (AL3) and the normalized BP alignment ratio, which suggests the adequacy scores across different alignment length groups (specifically, longer than one word BMWs) are significantly different in terms of the normalized alignment ratio.

We fitted three mixed effect model with the same method using fluency scores as the response variable. Significant random effect of interaction between sample translations and the normalized alignment ratio (${\chi_{(245.9,276)}}^2 = 1983$, $p < .0001$) can also be found. The output is reported below in Table 10. As the output shows, both alignment ratio and the longer alignments have significant impacts on the fluency scores as well, and apparently three word BMWUs

Table 10: Fixed Effects of Alignment Ratio and Alignment Length on Fluency

Response Variable	Effect	Estimate	SD	Z	$Pr. > tor Z$	AIC
Fluency	Intercept	5.15	0.74	6.98	< 0.0001	7149.1
	NrmALR	0.90	0.39	2.29	< 0.05	
	AL2	1.35	0.69	1.94	< 0.05	
	AL3	1.89	0.75	2.52	< 0.05	
	TrCorpS2	-0.05	0.09	-0.55	< 0.58	
	NrmALR:AL2	-0.09	0.47	-0.21	> 0.05	
	NrmALR:AL3	-1.49	0.75	-2.00	< 0.05	

have more weight over other two. The interaction of normalized BMWU alignment ratio and BMWU length also suggests that longer alignments contribute more to fluency than shorter alignments (one and two word alignments) in our case.

4.3. Implications

Generalized additive modelling suggests that BMWU alignments (two words and above) play a significant role in determining the quality of students' translations. There is also very strong indication that alignment length interacts with the normalized alignment ratios in students' translations and impact on their quality, i.e. the normalized alignment ratios of different lengths vary in their contribution to the quality scores for adequacy and fluency. As Table 1 shows, translators tend to resort to prefabricated translation pairs (e.g. BMWUs) available to them. This decision-making conforms to the idiom principle (Sinclair, 1991) or formulaic language (Wray, 2001), which help the translators produce native-like selections and reduce the cognitive processing effort.

However, it seems that our BMWU alignments have less effect on fluency. This contradicts our intuition but can be explained by the fact that alignment places more emphasis on correspondences, which are often oriented at semantic equivalence. In terms of the fluency scores, our BMWUs are relatively short (up to 4 words in this study), so longer units to capture the discourse markers, cohesion devices, etc.

5. Related Work

Recent years have seen attempts using word alignment information for translation quality estimation (QE), for either machine translation or human translation (Ueffing et al., 2003; Abdelsalam et al., 2016; Specia et al., 2015; Camargo de Souza et al., 2013; Bach et al., 2011; Popović et al., 2011; Popovic, 2012; Yuan et al., 2016) . As Abdelsalam et al. (2016) noted, the majority of these research focus on exploiting alignment related information for word-level QE. Among the few studies Abdelsalam et al. (2016), Camargo de Souza et al. (2013) and Yuan et al. (2016) actually try to tackle QE at the sentence-level or above, some features are too complex and not friendly interpretable to humans. For instance, Bach et al. (2011) use the source and target alignment context and even combine alignment context with PoS tags, and Camargo de Souza et al. (2013)

implement features, such as proportion of alignments connecting words with the same PoS tag and proportion of words in ST and TT that share the same PoS tag. We argue that on the one hand, such features are not computing cost-effective, and on the other hand, they are against our intuition that linguistic attributes, such as PoS, will hardly remain the same during the translation of two drastically different language pairs. We continue our way of obtaining alignment precision and recall in (Yuan et al., 2016), which compute the proportion of aligned words in source sentences or documents (precision) and the proportion of aligned words in target sentences or documents (recall), similar to two of many alignment features[6] by (Camargo de Souza et al., 2013). However, our BP alignment features differ by considering the sentence or document length information of ST and TT, and they are normalized [7]. Meanwhile, to be clear, we are not investigating how these features contribute in the QE task, but as part of feature engineering process, we explore statistically how different lengths and types of BP alignments interact with human rated translation quality scores (adequacy and fluency) to prove that they are useful in the future QE tasks.

6. Conclusion

In this study we investigated the effects of BMWUs extracted from parallel data to measure the adequacy and fluency of human translations.

Statistical analysis shows that the normalized alignment ratios for the phrase alignments longer than two words have the greatest impact on measuring translation quality. It is also found that longer phrase alignment ratios are statistically closer to adequacy than to fluency. We plan exploiting aligned BMWUs in the QE task (for evaluating both human and machine translations) in the form of normalized BMWU alignment ratios. These features can be used for QE at the sentence and document level. The latter task is particularly important, as the summative evaluation of trainee translators is typically done at the level of document translations, in contrast to conventional MT QE at the sentence level. Extending the alignment features to the phrasal level is consistent with human translation intuition and language production hypothesis, e.g. idiom principle (Sinclair, 1991). Most importantly, to the best of our knowledge, this paper is the first attempt of investigating the effects of phrasal alignment on human translation quality, and the method of computing alignment ratios has extended further beyond the word-level alignment features in previous MT QE studies. We share scripts from this study at `https://github.com/hittle2015/Bi-MWU.git`.

We can expect several extensions to the proposed model. First, our experiment uses general purpose parallel corpora, while the trainee translations come from a specific domain. It is interesting to investigate prediction of translation quality using parallel corpora from the same domain to measure the contribution of the proposed BMWU alignment ratios. Second, our experiment reported here accepts any phrase alignment from the professionally translated corpus as matching the trainee translations without taking their neighbouring contexts into consideration. We can try including a model for the context by using Recurrent Neural Networks methods from Neural Machine Translation when the neighbouring words contribute to the translation decisions (Koehn and Knowles, 2017).

Another extension for this study concerns increasing the amount of reliable BMWUs by extracting them from the comparable corpora (Sharoff et al., 2013), since the amount of data from monolingual corpora is much greater than what comes from parallel corpora, especially for specific domains. There has been extensive research on alignment of the monolingual embedding spaces for individual words, see an overview in (Conneau et al., 2017), but so far not much on BMWUs.

[6] They use proportion of aligned words and proportion of aligned n-grams. The latter is similar to our proposed feature of phrase alignment.

[7] We also propose that the summation of the logarithmized probabilities (IBM scores) of all aligned words in the documents (sentences) could be a potential quality indicator, and so is the geometric mean of these probabilistic scores.

7. Acknowledgements

This study is partially funded by the Jiangsu Provincial Social Science Fund (No.: 17YYB013) and the teaching reform project of Jinling Institute of Technology (No.: JYJG2017-34).

8. References

Abdelsalam, A., Bojar, O., and El-Beltagy, S. (2016). Bilingual embeddings and word alignments for translation quality estimation. In *Proceedings of the First Conference on Machine Translation*, pages 764–771, Berlin, Germany, August. Association for Computational Linguistics.

Abu-Ssaydeh, A.-F. (2004). Translation of english idioms into arabic. *Babel*, 50(2):114–131.

Bach, N., Huang, F., and Al-Onaizan, Y. (2011). Goodness: A method for measuring machine translation confidence. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Techologies*, pages 211–219, Portland, Oregon, USA, June. Association for Computational Linguistics.

Baker, M. (2011). *In other words: a coursebook on translation*. Routledge, London, 2nd edition.

Baobao, C., Danielsson, P., and Teubert, W. (2002). Extraction of translation unit from chinese-english parallel corpora. In *Proceedings of the First SIGHAN Workshop on Chinese Language Processing - Volume 18*, SIGHAN '02, pages 1–5, Stroudsburg, PA, USA. Association for Computational Linguistics.

Camargo de Souza, J. G., Buck, C., Turchi, M., and Negri, M. (2013). FBK-UEdin participation to the WMT13 quality estimation shared task. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 352–358, Sofia, Bulgaria, August. Association for Computational Linguistics.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. (2017). Word translation without parallel data. *CoRR*, abs/1710.04087.

DeNero, J. and Klein, D. (2008). The complexity of phrase alignment problems. In *Proceedings of ACL-08: HLT, Short Papers*, pages 25–28, Columbus, Ohio, June. Association for Computational Linguistics.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia, June. Association for Computational Linguistics.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proc Neural Machine Translation Workshop*, Vancouver.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

Liu, I. and Agresti, A. (2005). The analysis of ordered categorical data: An overview and a survey of recent developments. *Test*, 14(1):1–73.

Min, F. (2007). Cultural issues in chinese idioms translation. *Perspectives*, 15(4):215–229.

Neubig, G., Watanabe, T., Sumita, E., Mori, S., and Kawahara, T. (2011). An unsupervised model for joint phrase alignment and extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 632–641, Portland, Oregon, USA, June. Association for Computational Linguistics.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Och, F. J., Tillmann, C., Ney, H., et al. (1999). Improved alignment models for statistical machine translation. In Pascale Fung et al., editors, *Proceedings of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, College Park,MD,USA, 21-22 June.

Pitman, J. and Yor, M. (1997). The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, pages 855–900.

Popović, M., Vilar, D., Avramidis, E., and Burchardt, A. (2011). Evaluation without references: Ibm1 scores as evaluation metrics. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 99–103, Edinburgh, Scotland, July. Association for Computational Linguistics.

Popovic, M. (2012). Morpheme- and pos-based ibm1 and language model scores for translation quality estimation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 130–134, Montreal, Canada, June. Association for Computational Linguistics.

Sag, I., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. *Computational Linguistics and Intelligent Text Processing*, pages 189–206.

Sharoff, S., Rapp, R., and Zweigenbaum, P. (2013). Overviewing important aspects of the last twenty years of research in comparable corpora. In Serge Sharoff, et al., editors, *BUCC: Building and Using Comparable Corpora*, pages 1–17. Springer.

Sinclair, J. M. (1991). *Corpus, concordance, collocation*. Oxford University Press, Oxford.

Specia, L., Paetzold, G., and Scarton, C. (2015). Multilevel translation quality prediction with quest++. In *ACL-IJCNLP 2015 System Demonstrations*, pages 115–120, Beijing, China.

Tian, L., Wong, D. F., Chao, L. S., Quaresma, P., Oliveira, F., and Yi, L. (2014). Um-corpus: A large english-chinese parallel corpus for statistical machine translation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Ueffing, N., Macherey, K., and Ney, H. (2003). Confidence measures for statistical machine translation. In *Proceedings of the MT Summit IX*, New Orleans, USA, September. the Association for Machine Translation in the Americas.

Wang, W.-l. and Nian, X.-p. (2004). Loss of beauty in translation of idiom and compensation. *Hefei Gongye Daxue Xuebao Shehui Kexue Ban/Journal of Hefei University of Technology (Social Sciences)*, 18(6):118–120.

Wood, S. N., Pya, N., and Säfken, B. (2016). Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association*, 111(516):1548–1563.

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R, Second Edition*. Chapman & Hall / CRC texts in statistical science. Chapman and Hall/CRC, second edition edition.

Wray, A. (2001). *Formulaic language and the lexicon*. Cambridge University Press, Cambridge.

Yuan, Y., Sharoff, S., and Babych, B. (2016). Mobil: A hybrid feature set for automatic human translation quality assessment. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May. European Language Resources Association (ELRA).

Zebardast, M. R. and AbuSaeedi, A. A. R. (2015). Cultural loss versus hermeneutic of translation: A case study on the idioms of "death of a salesman" (miller, 1949). *Journal of Academic and Applied Studies*, 5(11).

Zebardast, M. R. (2015). Perceptions about cultural loss in translating idioms from english into persian: A case study on the “death of a salesman” (miller, 1949). *Journal of Academic and Applied Studies*, 5(10).

Evaluation of Dictionary Creating Methods for Finno-Ugric Minority Languages

Zsanett Ferenczi, Iván Mittelholcz, Eszter Simon, Tamás Váradi
Research Institute for Linguistics, Hungarian Academy of Sciences,
H-1068 Budapest, Benczúr u. 33.
{ferenczi.zsanett, mittelholcz.ivan, simon.eszter, varadi.tamas}@nytud.mta.hu

Abstract

In this paper, we present the evaluation of several bilingual dictionary building methods applied to {Komi-Permyak, Komi-Zyrian, Hill Mari, Meadow Mari, Northern Saami, Udmurt}–{English, Finnish, Hungarian, Russian} language pairs. Since these Finno-Ugric minority languages are under-resourced and standard dictionary building methods require a large amount of pre-processed data, we had to find alternative methods. In a thorough evaluation, we compare the results for each method, which proved our expectations that the precision of standard lexicon building methods is quite low for under-resourced languages. However, utilizing Wikipedia title pairs extracted via inter-language links and Wiktionary-based methods provided useful results. The newly created word pairs enriched with several linguistic information are to be deployed on the web in the framework of Wiktionary. With our dictionaries, the number of Wiktionary entries in the above mentioned Finno-Ugric minority languages can be multiplied.

Keywords: bilingual dictionaries, evaluation, under-resourced languages, dictionary building methods

1. Introduction

The research presented in this paper is part of a project whose general objective is to provide linguistically based support for several small Finno-Ugric (FU) digital communities to generate online content and help revitalize the digital functions of some FU minority languages. The practical objective of the project is to create bilingual dictionaries for six small FU languages (Komi-Permyak, Komi-Zyrian, Meadow Mari, Hill Mari, Northern Saami, and Udmurt) paired with four major languages that are important for these small communities (English, Finnish, Hungarian, Russian) as well as to deploy the enriched lexical material on the web in the framework of the collaborative dictionary project Wiktionary.

The status of any particular language of the world is usually described using the Expanded Graded Intergenerational Disruption Scale (EGIDS) (Lewis and Simons, 2010), which gives an estimation of the overall development versus endangerment of the language. In this scale, the highest level is 0, where languages are world-wide used *koiné*s, while languages on level 10 are already extinct. Northern Saami is on the highest level among the above mentioned FU languages: its level is 2 (provincial), thus it is used in education, work, mass media, and government within some officially bilingual region of Norway, Sweden, and Finland. In the case of the Meadow Mari language, the EGIDS level is 4 (educational), which means that it is in vigorous use, with standardization and literature being sustained through a widespread system of institutionally supported education. The EGIDS level of the other FU languages (Komi-Permyak, Komi-Zyrian, Hill Mari, Udmurt) is 5, i.e. they are developing, which means that there is literature which is available in a standardized form, though it is not yet widespread or sustainable.

The above mentioned FU languages are not endangered but under-resourced, hence we could not collect enough data for building parallel and comparable corpora, on which the standard dictionary building methods are based. The standard approach of bilingual lexicon extraction from parallel and comparable corpora is based on context similarity methods (e.g. Fung and Yee (1998; Rapp (1995)). Recently, source and target vectors are learned as word embeddings in neural networks based on gigaword corpora (e.g. Vulić and Moens (2015)). However, these methods need a large amount of (pre-processed) data and a seed lexicon which is then used to acquire additional translations of the context words. One of the shortcomings of this approach is that it is sensitive to the choice of parameters such as the size of the context, the size of the corpus, the size of the seed lexicon, and the choice of the association and similarity measures.

For these reasons, the above mentioned standard dictionary building methods cannot be used for our purposes. Therefore, it was necessary to conduct experiments with alternative methods. We made experiments with several lexicon building methods utilizing crowd-sourced language resources, such as Wikipedia and Wiktionary (Simon et al., 2015; Benyeda et al., 2016). Completely automatic generation of clean bilingual resources is not possible according to the state of the art, but it is possible to create certain lexical resources, termed proto-dictionaries, that can support lexicographic and NLP work. Proto-dictionaries contain candidate translation pairs produced by bilingual dictionary building methods. Depending on the method used, they either comprise more incorrect translation candidates and provide greater coverage, or provide precise word pairs at the expense of some decrease in recall; their right size depends on the specific needs.

Once the proto-dictionaries were prepared, they were merged for each language pair and repeated lines were filtered out. These files were then the object of manual validation by native speakers and linguist experts of the languages. These validated dictionaries containing translation units were the input of generating new Wiktionary entries which were created fully automatically. As the last step of

the project, we upload the entries to Wiktionary.
The rest of the article is as follows. In Section 2., the methods used for creating the proto-dictionaries are shortly presented. We conducted thorough evaluation of the dictionaries produced for each language pair. In Section 3., the results of the evaluation is presented: in Section 3.1., we present the workflow of the manual validation of the automatically generated dictionaries, in Section 3.2., we detail the precision of each dictionary creating method applied here, and in Section 3.3., we figure out a kind of coverage for the newly created dictionaries in each language pair. The article ends with some conclusions and future directions in Section 4..

2. Creating the Proto-dictionaries

For the creation of the proto-dictionaries, we applied several lexicon building methods utilizing Wikipedia and Wiktionary. For more details on the dictionary creating methods we used, see Benyeda et al. (2016) and Simon and Mittelholcz (2017) – here we only provide a short description.
Wikipedia is not only the largest publicly available database of comparable documents, but it can also be used for bilingual lexicon extraction in several ways. For example, Erdmann et al. (2009) used pairs of article titles for creating bilingual dictionaries, which were later expanded with translation pairs extracted from the article texts. Mohammadi and Ghasem-Aghaee (2010) extracted parallel sentences from the English and Persian Wikipedia using a bilingual dictionary generated from Wikipedia titles as a seed lexicon. We followed the approach which is common in both articles, thus we created bilingual dictionaries from Wikipedia title pairs using the interwiki links.
Besides Wikipedia, *Wiktionary* is also considered as a crowd-sourced language resource that can serve as a source of bilingual dictionary extraction. Although Wiktionary is primarily for human audience, the extraction of underlying data can be automated to a certain degree. Ács et al. (2013) extracted translations from the so-called translation tables. Since their tool `Wikt2dict` is freely available[1], we could apply it for our language pairs. We parsed the English, Finnish, Russian and Hungarian editions of Wiktionary looking for translations in the small FU languages we deal with.
Ács (2014) expanded the collection of translation pairs, discovering previously non-existent links between translations with a triangulation method. It is based on the assumption that two expressions are likely to be translations, if they are translations of the same word in a third language.

3. Evaluation

The proto-dictionaries for each language pair were merged, and repeated lines were filtered out. Besides the above mentioned proto-dictionaries, the large merged file also contains a proto-dictionary which was not created by us but was downloaded from the Opus corpus (Tiedemann, 2009). For the Northern Saami–{English, Finnish, Hungarian} language pairs, there are available dictionaries which are lists of "reliable" alphabetic token links extracted from the automatic word alignment created with GIZA++ and the Moses toolkit. First, word pairs where the source and target words were character-level equivalents of each other were removed, since they are probably incorrect word pairs and remaining parts after (or in the lack of) boilerplate removal. The remaining part of the dictionary was also merged into the large dictionary, serving as an interesting example of applying standard lexicon extraction tools for an under-resourced language. The text material from which the Opus proto-dictionaries come is a parallel corpus of KDE4 localization files, where the Northern Saami–English parallel data contain 0.9M tokens, the Northern Saami–Finnish data contain 0.6M tokens, and the Northern Saami–Hungarian data contain 0.8M tokens. At the time of creating the proto-dictionaries, there were no dic files available for the other language pairs besides the three mentioned above in the Opus corpus.

3.1. Manual Validation

The large merged files were then manually validated by native speakers and linguist experts of the FU languages. The instructions for the validators were as follows. The source and the target word must be a valid word in the language concerned, they must be dictionary forms, and they must be translations of each other. If the source word is not a valid word in the FU language, the word pair is treated as wrong. If the source word is a valid word but not a dictionary form, the correct dictionary form should be manually added. If the target word is a good translation of the source word but is not a dictionary form, the correct dictionary form should be added. If the target word is not a good translation, a new translation should be given.
The following categories come from these instructions:

- ok-ok: The source and the target word are valid words, they are dictionary forms, and they are translations of each other.
- ok-nd: The source and the target word are valid words, they are translations of each other, but the target word is not a dictionary form.
- nd-ok: The source and the target word are valid words, they are translations of each other, but the source word is not a dictionary form.
- nd-nd: The source and the target word are valid words, they are translations of each other, but none of them are dictionary forms.
- ok-wr: The source word is a valid word, it is a dictionary form, but the target word is not a valid word or it is not a correct translation of the source word.
- nd-wr: The source word is a valid word but not a dictionary form, and the target word is not a valid word or it is not a correct translation of the source word.
- wr-xx: The source word is not a valid word.

The validated dictionaries, however, were not fully clean and ready-to-use, thus several checking and correcting steps were required. As a sanity check, we made sure that

[1] `https://github.com/juditacs/wikt2dict`

the dictionary contains a source and a target word, checked whether any cells contain suspicious characters, etc. As a consistency check, cases when the target word was provided with a dictionary form as well as a new translation and cases when the source word was treated as wrong but a new translation was added for the target word were filtered out. A cross-language consistency check was also done, in which we checked whether source words were treated consistently in all languages. At the end of this workflow, we got the validated dictionaries containing the translation units, which served then as the input of the evaluation and the newly generated Wiktionary entries.

As mentioned in Section 1., the manually validated word pairs are used as the source material of newly created Wiktionary entries, which contain several obligatory elements. These elements containing morphological and phonetical information are generated fully automatically. For example, in the case of the Northern Saami–English language pair, the Northern Saami word will be an entry in the English Wiktionary: the title of the entry will be the Northern Saami word, while its English definition will be its English translation equivalent.

The manual validation and correction of the automatically generated proto-dictionaries has a twofold aim. First, the performance of dictionary creating methods can be compared. Second, we get the number of word pairs which can be used for upload to the Wiktionary.

3.2. Precision

Category tags given to word pairs in the merged dictionaries were projected onto the corresponding word pairs in the proto-dictionaries. Results for each method were then summed up across all language pairs, as can be seen in Table 1. Besides category tags, the total number of dictionary entries of proto-dictionaries is presented in the first column. Abbreviations of the name of the methods are as follows: W2D ext: `Wikt2dict` extraction mode, W2D tri: `Wikt2dict` triangulation mode, WikiTit: Wikipedia title pairs, Opus: dic files downloaded from the Opus corpus.

In Table 1, methods are presented in a descending order based on their performance in the ok-ok category. This score is the precision of a method, i.e. the ratio of the number of the correct word pair to the total number of word pairs. Depending on the research purpose in question, word pairs containing non-dictionary forms can also be treated as correct translations, thus precision metrics may vary among approaches. Here we use it in a strict sense, thus a word pair is correct iff it is in the ok-ok category.

Some precision-like metrics are generally used for the evaluation of automatically generated bilingual dictionaries. For example, Vulić et al. (2011) use Precision@1 score, which is the percentage of words where the first word from the list of translations is the correct one, and mean reciprocal rank (MRR), where for a source word w, $rank_w$ denotes the rank of its correct translation within the retrieved list of potential translations. All these metrics are based on the assumption that the method used produces a list of translation candidates along with some confidence or probability measures. Even though it is not the case in our work, we can treat figures in the ok-ok column in Table 1 as Precision@1 scores calculated for a one-unit list of translation candidates.

The most precise method is using `Wikt2dict` in extraction mode thus extracting translation equivalents from Wiktionary translation tables. Word pairs coming from this method are quite reliable, since Wiktionary entries are manually created.

The second method is using `Wikt2dict` in triangulation mode, but there is a 15% decrease in the precision of this method compared to that of the first one. As this method does not directly use manually created links, its output may contain incorrect translations. The ok-wr figure for this method is the highest, mainly due to polysemy.

Wikipedia has very valuable translation texts since these translations were manually made by editors. Therefore, it is quite surprising, that using Wikipedia title pairs as a dictionary proved to be just the third most precise method. Consider the high nd-nd figure, which may be due to the fact that Wikipedia titles sometimes are not lemmas but plural forms, for example in the case of the names of families of plants and animals.

The worst result was produced by the method used in the Opus corpus, which is a standard dictionary building method based on parallel text material, using standard alignment and word pair extraction tools developed for well-resourced languages. Figures of this method are more flat, i.e. word pairs are distributed more uniformly across the categories compared to the other methods. This may be due to several reasons. First, the Opus dictionaries were generated from running text containing inflected and derived word forms and lemmas as well. Therefore, the number of non-dictionary forms and wrong translations is higher. (Inflected word forms were treated as valid words in non-dictionary form, while derived forms were categorized as wrong by the validators.) Second, the tools used within the Opus corpus project are not really feasible for under-resourced languages, therefore they produced more non-dictionary forms and wrong word pairs.

The large merged dictionary of each language pair was then evaluated for each category described in 3.1.; the results can be seen in Table 2. We use ISO 639-3 language codes for the individual languages: koi: Komi-Permyak, kpv: Komi-Zyrian, mhr: Meadow Mari, mrj: Hill Mari, sme: Northern Saami, udm: Udmurt; eng: English, fin: Finnish, hun: Hungarian, rus: Russian.

The first column of the table shows the total number of word pairs gathered with all methods for the language pair. As can be seen, hundreds or thousands of translation candidates were generated for each language pair. The best language pair in this sense is sme–fin, which may be because Northern Saami is by far the best-resourced minority language of the ones we deal with, and it is an official language in several regions of Finland, where the Saami–Finnish bilingual population is quite large.

Since the validated dictionaries are the input of generating new Wiktionary entries, we need to extract all useful word pairs from the merged dictionary for each language pair. The second column of the table contains the ratio of the number of useful word pairs to the number of all word pairs.

method	all (#)	ok-ok (%)	ok-nd (%)	nd-ok (%)	nd-nd (%)	ok-wr (%)	nd-wr (%)	wr-xx (%)
W2D ext	1,965	71.76	1.22	5.75	15.17	4.63	0.36	0.76
W2D tri	23,066	56.61	1.79	2.98	3.06	30.38	1.1	3.82
WikiTit	16,854	54.11	2.97	5.57	32.5	2.92	0.49	0.75
Opus	8,401	27.57	3.99	10.4	18.64	13.99	14.57	10.69

Table 1: Results for the methods.

lang pair	all (#)	useful (%)	ok-ok (%)	ok-nd (%)	nd-ok (%)	nd-nd (%)	ok-wr (%)	nd-wr (%)	wr-xx (%)
koi–eng	1,251	96.64	74.82	0.16	7.83	0.00	13.67	0.16	3.36
koi–fin	592	98.82	65.20	3.04	9.97	0.84	19.59	0.17	1.18
koi–hun	540	93.15	70.19	3.33	4.63	1.30	13.52	0.19	6.85
koi–rus	611	98.85	65.47	2.95	16.69	1.47	11.62	0.65	1.15
kpv–eng	902	100.00	66.30	0.22	0.55	30.16	2.55	0.22	0.00
kpv–fin	577	100.00	57.89	3.29	0.69	37.09	0.87	0.17	0.00
kpv–hun	523	99.81	49.71	1.34	0.96	43.98	3.82	0.00	0.19
kpv–rus	544	100.00	63.60	8.64	9.93	14.52	3.31	0.00	0.00
mhr–eng	2,549	100.00	44.41	2.55	4.04	22.40	26.09	0.51	0.00
mhr–fin	2,565	100.00	50.80	1.05	3.31	20.74	23.63	0.47	0.00
mhr–hun	1,647	100.00	52.64	0.97	5.89	25.20	14.15	1.15	0.00
mhr–rus	1,707	100.00	40.01	2.11	4.28	17.28	35.56	0.76	0.00
mrj–eng	2,334	100.00	44.09	0.17	9.04	43.10	3.08	0.51	0.00
mrj–fin	1,013	100.00	20.24	7.70	9.77	52.32	8.59	1.38	0.00
mrj–hun	942	100.00	34.18	4.99	12.95	41.08	5.20	1.59	0.00
mrj–rus	835	100.00	27.07	11.26	9.58	31.38	16.89	3.83	0.00
sme–eng	6,041	91.97	47.57	3.77	7.33	6.56	21.65	5.08	8.03
sme–fin	7,100	91.03	42.03	3.42	5.42	12.56	19.92	7.66	8.97
sme–hun	4,969	90.78	48.48	1.67	6.72	6.62	17.05	10.24	9.22
sme–rus	4,373	95.40	71.35	0.50	2.56	0.18	20.05	0.75	4.60
udm–eng	2,087	99.14	77.19	3.07	0.91	0.29	17.59	0.10	0.86
udm–fin	1,700	99.65	49.12	2.06	1.06	18.82	28.06	0.53	0.35
udm–hun	1,204	99.50	57.14	1.74	1.50	23.17	15.45	0.50	0.50
udm–rus	1,226	98.78	8.56	2.04	0.98	65.25	20.64	1.31	1.22

Table 2: Results for the merged dictionaries.

In this case, useful word pairs comprise all word pairs minus the wr-xx category, since correct dictionary forms and translation equivalents were manually added by the human validators.

The remaining columns contain the results for each category coming from the instructions given to the validators (see Section 3.1.). A typical pattern can be recognized: if the ok-ok figure is low, the nd-nd figure will be high. It may be because of the high number of non-dictionary forms amongst Wikipedia titles, such as in the case of families of animals and plants, e.g. мерач-влак ∼ *nyúlfélék* ∼ *Leporidae*.

3.3. Coverage

If the number of the created dictionary entries can be treated as a kind of coverage, based on the figures of Table 1, it can be said that the `Wikt2dict` triangulation method has the best coverage, since it produced the largest number of translation candidates. As usual, the most precise method has the lowest coverage. We could gather much more word pairs from Wikipedia titles than from Wiktionary translation tables, which is likely due to the fact that Wikipedia contains more articles compared to the number of translations in Wiktionary's translation tables. Moreover, the number of entries highly depends on the activity of editors knowing these FU languages and willing to create new entries.

Coverage of a dictionary can also be measured by comparing the number of its entries to that of a hand-crafted dictionary. Since our newly created word pairs are to be transformed into Wiktionary articles, for this purpose, here we used Wiktionary, which is not an expert-built lexicon but manually edited by thousands of contributors.

Table 3 contains the figures for this kind of coverage evaluation. Several Wiktionary editors do not differentiate between individual languages but use macrolanguage codes (chm for Mari languages, kom for Komi languages), therefore we had to merge the dictionaries for the two Mari and for the two Komi languages.

The first column of the table ('all') shows the total number of word pairs gathered with all methods for the language pair. As can be seen, thousands of translation candidates were generated for each language pair. However, not all of these word pairs are correct translation candidates, therefore we needed to extract the useful word pairs from the merged dictionary for each language pair. The second column ('useful %') shows the percentage of useful word pairs, while the third column ('useful #') contains the number of useful word pairs. In this case, useful word pairs comprise all word pairs minus the wr-xx category, since correct dictionary forms and translation equivalents were manually added by human validators.

As mentioned above, our Wiktionary articles are generated fully automatically. The part-of-speech (POS) tag of an entry is a compulsory element of an article, which is gathered from the output of morphological analyzers available for these languages through several disambiguating steps, as detailed in Ferenczi et al. (2018). The number of the useful word pairs drops in line with the increase of source language words for which we could not provide a POS tag.

Before uploading new entries, it must be checked whether an entry with the same word already exists in Wiktionary. If yes, it also decreases the number of uploadable word pairs. Column 'remaining' contains the decreased number of the word pairs ready for upload. We have also got the number of the source language words already existing in the target language Wiktionary ('Wiktionary'), along with the number of the words being in both lists ('common'). These numbers come from the Wiktionary dumps[2] and are "theoretical" numbers in the sense that they are not the numbers of actually uploaded entries, which can only be known after uploading.

From the columns 'Wiktionary' and 'common', the number of brand new entries created by us ('new') can be easily counted, along with a kind of coverage ('coverage'), which is a ratio of the number of common words to the number of words already being in Wiktionary, thus it is the degree of overlap with Wiktionary. Consider that the coverage for each language pair drops as the size of the relevant Wiktionary grows. The last column ('improvement') contains the ratio of the number of the new Wiktionary entries to one of the already existing ones which shows the improvement in the amount of Wiktionary entries of the given source language in the given target language edition of Wiktionary.

4. Conclusions and Future Work

We presented several bilingual dictionary building methods applied to the {Komi-Permyak, Komi-Zyrian, Meadow Mari, Hill Mari, Northern Saami, Udmurt}–{English, Finnish, Hungarian, Russian} language pairs. Since these FU languages are under-resourced and standard dictionary building methods require a large amount of pre-processed data, we had to find alternative methods. In a thorough evaluation, we compared the results for each method, which proved our expectations that the precision of standard lexicon building methods is quite low.

The Wiktionary-based methods proved to be the most precise, but using Wikipedia title pairs extracted via interlanguage links also provided useful results.

Wiktionary is not only used for extracting data from it, but we want to give our results back to the community, thus translation pairs enriched with the required pieces of linguistic information are to be uploaded as new entries into Wiktionary. Before uploading new entries, it must be checked whether an entry with the same word already exists in Wiktionary. From this, the number of brand new entries created by us could be easily counted, along with a kind of coverage and improvement in the number of Wiktionary entries. As can be seen from the results, the latter is very impressive, thus, with our dictionaries, we could multiply the number of Wiktionary entries in the above mentioned Finno-Ugric minority languages. Since automatic uploading of entries is not supported by the Wiktionary community, we must obtain the permission to upload our newly created entries into Wiktionary. We have already permitted to upload the new entries into the Finnish and Hungarian versions of Wiktionary; in the time of writing, however, we still do not have the permission from the English and Russian Wiktionary editors.

We provide freely available professional online multilingual lexical data for digital communities of some FU minority languages with Wiktionary entries. However, lexical data can be provided in several other ways. We plan to make them available in standard data formats (e.g. `tsv`, `XML`) which are easy to apply in further lexicographic or NLP work. We also want to convert our data into the data format following the conventions of linguistic linked open data and provide them via our web site[3] or via the repositories of dictionary families such as Giellatekno[4].

5. Acknowledgements

The research reported in the paper was conducted with the support of the Hungarian Scientific Research Fund (OTKA) grant #107885.

6. Bibliographical References

Ács, J., Pajkossy, K., and Kornai, A. (2013). Building basic vocabulary across 40 languages. In *6th Workshop on Building and Using Comparable Corpora*, pages 52–58, Sofia. ACL.

Ács, J. (2014). Pivot-based multilingual dictionary building using Wiktionary. In *9th Language Resources and Evaluation Conference*, Reykjavik. ELRA.

Benyeda, I., Koczka, P., and Váradi, T. (2016). Creating seed lexicons for under-resourced languages. In *GLOBALEX 2016 workshop*, Portorož. ELRA.

[2]Wiktionary dumps used in the evaluation: eng: 06-Nov-2017, fin: 05-Nov-2017, rus: 07-Nov-2017, hun: 06-Nov-2017.

[3]`http://finnotka.nytud.hu`

[4]`http://giellatekno.uit.no`

lang pair	all (#)	useful (%)	useful (#)	remaining (#)	Wiktionary (#)	common (#)	new (#)	coverage (%)	improvement (%)
kom–eng	2,153	95.26	2,051	655	54	25	630	46.30	1166.67
kom–fin	1,169	95.54	1,117	687	42	27	660	64.29	1571.43
kom–hun	1,063	95.29	1,013	699	152	35	664	23.03	436.84
kom–rus	1,155	92.54	1,069	672	465	223	449	47.96	96.56
chm–eng	4,883	98.83	4,826	1,670	347	53	1,617	15.27	465.99
chm–fin	3,578	98.57	3,527	1,903	443	213	1,690	48.08	381.49
chm–hun	2,589	98.29	2,545	1,633	34	12	1,621	35.29	4767.65
chm–rus	2,542	98.11	2,494	1,493	848	201	1,292	23.70	152.36
sme–eng	6,041	91.97	5,556	2,531	4,072	882	1,649	21.66	40.50
sme–fin	7,100	91.03	6,463	2,862	817	422	2,440	51.65	298.65
sme–hun	4,969	90.78	4,510	2,392	206	146	2,246	70.87	1090.29
sme–rus	4,373	95.40	4,172	2,034	306	237	1,797	77.45	587.25
udm–eng	2,087	99.14	2,069	751	32	15	736	46.88	2300.00
udm–fin	1,700	99.65	1,694	828	55	45	783	81.82	1423.64
udm–hun	1,204	99.50	1,198	729	128	69	660	53.91	515.62
udm–rus	1,226	98.78	1,211	578	643	247	331	38.41	51.48

Table 3: Results for the language pairs.

Erdmann, M., Nakayama, K., Hara, T., and Nishio, S. (2009). An Approach for Extracting Bilingual Terminology from Wikipedia. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 5(4):1–17.

Ferenczi, Zs., Mittelholcz, I., and Simon, E. (2018). Automatic Generation of Wiktionary Entries for Finno-Ugric Minority Languages. In *Proceedings of the 4th International Workshop for Computational Linguistics for Uralic Languages (IWCLUL 2018)*, page 39–50, Helsinki, Finland. Association for Computational Linguistics.

Fung, P. and Yee, L. Y. (1998). An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In *17th International Conference on Computational Linguistics*, pages 414–420, Stroudsburg. ACL.

Lewis, M. P. and Simons, G. F. (2010). Assessing endangerment: Expanding Fishman's GIDS. *Revue Roumaine de Linguistique*, 55(2):103–120.

Mohammadi, M. and Ghasem-Aghaee, N. (2010). Building Bilingual Parallel Corpora Based on Wikipedia. In *2nd International Conference on Computer Engineering and Applications*, pages 264–268.

Rapp, R. (1995). Identifying word translations in non-parallel texts. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 320–322, Stroudsburg. ACL.

Simon, E. and Mittelholcz, I. (2017). Evaluation of Dictionary Creating Methods for Under-Resourced Languages. In Kamil Ekštein et al., editors, *Text, Speech and Dialogue*, volume 10415 of *Lecture Notes in Artificial Intelligence*, pages 246–254, Prague, Czech Republic, August. Springer International Publishing.

Simon, E., Benyeda, I., Koczka, P., and Ludányi, Zs. (2015). Automatic creation of bilingual dictionaries for Finno-Ugric languages. In *1st International Workshop on Computational Linguistics for Uralic Languages*, Tromsø.

Tiedemann, J. (2009). News from OPUS – A collection of multilingual parallel corpora with tools and interfaces. In Nicolas Nicolov, et al., editors, *Recent Advances in Natural Language Processing V: Selected Papers from RANLP 2007*, pages 237–248. John Benjamins, Borovets.

Vulić, I. and Moens, M.-F. (2015). Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In *53rd Annual Meeting of the Association for Computational Linguistics*, pages 719–725, Stroudsburg. ACL.

Vulić, I., De Smet, W., and Moens, M.-F. (2011). Identifying word translations from comparable corpora using latent topic models. In *49th Annual Meeting of the Association for Computational Linguistics*, pages 479–484, Stroudsburg. ACL.

Dysarthric speech evaluation: automatic and perceptual approaches

Imed Laaridh[1,2], **Christine Meunier**[2], **Corinne Fredouille**[1]
[1]University of Avignon, CERI/LIA, France
[2]Aix Marseille Université, CNRS, LPL UMR 7309, 13100, Aix-en-Provence, France
imed.laaridh@alumni.univ-avignon.fr, christine.meunier@lpl-aix.fr, corinne.fredouille@univ-avignon.fr

Abstract

Perceptual evaluation is still the most common method in clinical practice for the diagnosis and monitoring of the condition progression of people suffering from dysarthria (or speech disorders more generally). Such evaluations are frequently described as non-trivial, subjective and highly time-consuming (depending on the evaluation level). Clinicians have, therefore, expressed their need for new objective evaluation tools more adapted to longitudinal studies or rehabilitation context.

We proposed earlier an automatic approach for the anomaly detection at the phone level for dysarthric speech. The system behavior was studied and validated on different corpora and speech styles. Nonetheless, the lack of annotated French dysarthric speech corpora has limited our capacity to analyze some aspects of its behavior, and notably,its severity (more anomalies detected automatically compared with human expert). To overcome this limitation, we proposed an original perceptual evaluation protocol applied to a limited set of decisions made by the automatic system, related to the presence of anomalies. This evaluation was carried out by a jury of 29 non-naive individuals. In addition to interesting information related to the system behavior, the evaluation protocol highlighted the difficulty for a human, even expert, to apprehend and detect deviations at the word level in dysarthric speech.

Keywords: Dysarthria, speech disorders, automatic speech processing, perceptual evaluation

1. Introduction

Dysarthria is a motor speech disorder that is a consequence of neurological damage located in the central or/and the peripheral nervous system. This may result in disturbances in any of the components involved in speech production and may be reflected by weakness, spasticity, incoordination, involuntary movements, or abnormal muscle tone depending on the location of the neurological damage (Duffy, 2005).

Perceptual evaluation by a set of listeners is the most common paradigm used to evaluate the characteristics and severity of impairment in speech pathologies. Furthermore, the current classification of dysarthria into six categories is based on a perceptual description of deviant speech dimensions. ((Darley et al., 1969b; Darley et al., 1969a; Darley et al., 1975). The clinical evaluation of patients is also based on several batteries of tests in which the production of dysarthric speakers is coded perceptually by clinicians. These batteries evaluate the vocal quality, phonetic realizations, prosody, respiration and intelligibility. The BECD (*Batterie d'Evaluation Clinique de la Dysarthrie*) (Auzou and Rolland-Monnoury, 2006) is the most commonly used test by clinicians for French speech. This test differentiates 35 items in order to characterize dysarthria. Consequently, the use of perception for the evaluation of dysarthric speech is frequent and well documented. And the clinicians evaluating the speech of patients are very well trained to the phonetic characteristics associated with the physiopathology of dysarthria. However, a frequent criticism to perceptual evaluation is the subjectivity of the listeners (both naive and expert).

In some previous work, the authors have proposed an automatic phone-based anomaly detection approach (Laaridh et al., 2015a). By detecting and localizing anomalies in speech production, this approach aims at enhancing the manual investigation of human experts and, at the same time, reducing the extent of their intervention by scrutinizing the speech signal. Indeed, this automatic process should permit treating a larger amount of speech production while guiding human experts to focus on specific parts of the speech, considered as atypical. This process is notably interesting for speech productions of people with mild to moderate dysarthria, for which speech impairment may be scattered along the speech signal. Moreover, this automatic detection and localization of abnormal acoustic phenomena can have applications in clinical practice. For example, the evaluation of dysarthria by clinicians could be partially helped by a visual display of abnormal phenomena localized in the speech production of dysarthric speakers, like a map. In the same manner, maps could be relevant to compare the speech productions of a dysarthric speaker over time, in clinical treatment or rehabilitation for instance. Finally, this automatic process could be extended to other kinds of speech disorders resulting in acoustic alterations in the speech signal, such as larynx or head and neck cancers.

In this paper, the authors investigate on the behavior of the system, and typically on its potential quality or shortcoming to over-detect anomalies compared to a human expert. The larger research question this work tries to handle could be also that of the relationship between the human perception of alterations in speech and their modeling by automatic speech processing systems. In this context, the objective of this work is to propose an original perceptual evaluation protocol, suitable for evaluating the performance of the automatic system. This evaluation protocol is based on the comparison of the output decisions yielded by the system relating to the presence of anomalies, to those of a jury composed of a large set of expert listeners (in order to minimize the effect of individual subjectivity) on a selection of speech sequences produced by a large number of dysarthric patients representing four different pathologies, and control

speakers.
The rest of this article is organized as follows. In section 2, the automatic anomaly detection approach and the motivations of this research work are presented. The experimental protocol and corpora used for the perceptual evaluation are presented in section 3. In section 4, the evaluation results are presented and discussed according to different aspects. Finally, section 5 provides a conclusion and directions for future work.

2. Motivations

2.1. Automatic anomaly detection approach

The automatic phone-based anomaly detection system relies on two steps: a text-constrained phone alignment to obtain the phone segmentation and a classification of speech segments into normal and abnormal phones (anomalies).
The automatic phone segmentation of the speech utterances into phones is carried out thanks to an automatic text-constrained phone alignment tool (Laaridh et al., 2015b). This tool takes as input the parameterization of the speech signal produced by a given speaker, gender-dependent acoustic models of French phones, the sequence of words pronounced by the speaker in each utterance and a phonetized phonologically-varied lexicon of words based on a set of 37 French phones. The sequence of words comes from a manual orthographic transcription performed by a human listener following some annotation rules. Then, a set of features considered as relevant for the anomaly detection task are extracted over each segment y_p associated with the phone p. The list of the features used can be found in (Laaridh et al., 2015a). These phone features are then fed into a 2-class automatic classification system based on Support Vector Machines (SVM) (Vapnik, 1995; Scholkopf and Smola, 2001). Figure 1 describes the automatic anomaly detection process.
In (Laaridh et al., 2015a), the system was evaluated on a very limited corpus of dysarthric speech (4 female and 4 male dysarthric patients, suffering from the same pathology, and 6 control speakers) annotated by one human expert. This annotation was made especially for the system development and evaluation, by labeling each phone as deviant or not from an acoustic point of view. On this corpus, the system obtained a quite high averaged recall measure[1] of 0.81 (0.72 on male patients and 0.89 on female patients) and a less convincing averaged precision measure[2] of 0.63 (0.61 on male patients and 0.65 on female patients). Still in this work, the automatic system was applied on a non annotated corpus, implying a larger number of speakers (118 dysarthric patients and normal speakers) and different pathologies like Parkinson's Disease, Amyotrophic Lateral Sclerosis, and Cerebellar Ataxia. Since no labeled data regarding anomalies was available, evaluation was carried out by observing the relationship between the amount of speech anomalies reported by the automatic system and the perceptual rates given by an expert jury on the global severity degree of dysarthria, the global degree of intelligibility and of articulation impairment, and finally, the speech rate of speakers. Analysis of the results pointed out very interesting behavior of the automatic system, which exhibits very relevant correlations with the majority of the perceptual criteria (e.g. between 0.8 and 0.9 for almost all of the pathologies for the global severity degree). In another work (Laaridh et al., 2016), the application of the automatic anomaly detection on read and spontaneous speech still highlighted the interest of such an approach.

[1]The ratio between the number of phones correctly detected as anomalies by the automatic approach and the number of zones labeled as abnormal in the reference.

[2]The ratio between the number of phones correctly detected as anomalies by the automatic approach and the total number of anomalies reported by the automatic processing (truly or falsely).

2.2. Proposed protocol features

The aim of the work reported in this paper is to cope with the lack of annotated corpora suitable for evaluating the automatic detection of anomalies in speech produced by patients suffering from speech disorders, compared with normal speech. Generally, the annotation of corpora is long, costly and time-consuming. In our context, difficulties increase by the fact that the automatic detection of anomalies is carried out at the phone level. Previous unpublished work we did demonstrated that the perceptual evaluation of the presence of anomalies in speech production by human at the phone level is a very complex task, leading to very heterogeneous decisions, even while involving a large number of listeners. Based on these observations, we propose an original perceptual evaluation protocol of the outputs of the automatic system. The task of listeners in this protocol is still to determine the presence of speech deviance (anomalies), in terms of articulatory realization.
The first feature of this protocol is to transpose the decision of the automatic system, initially at the phone level, to the word level, to facilitate the perceptual evaluation done by humans. In this way, each monosyllabic word including, at least, one phone detected as an anomaly by the automatic system is considered as abnormal. In parallel, the presence of two phones, at least, detected as an anomaly in a polysyllabic word makes it abnormal.
The second feature of this protocol is the set of speech sequences used for the perceptual evaluation task. Due to the cost of such tasks mentioned above, the entire corpora automatically annotated by the system can not be used. The concentration level and cognitive effort required for each participant for the evaluation task has also to be taken into account. For these different reasons, the set of speech sequences has to be limited in size to make the task feasible and efficient while relevant for the assessment of the quality of the automatic system decisions. First, records presenting low signal quality, noise or other artifacts are excluded from the study. Then, on the basis of the automatic annotation, speech sequences are selected on the expertise of two human annotators (distinct from the jury used later). Each speech sequence contains from one to several words targeted for the perceptual evaluation. For example, in the sequence "*il **mange** tout seul bien **tristement***" (*he eats very sadly alone*), the words "***mange***" (*eats*) and "***tristement***" (*sadly*) are targeted for the evaluation; the other words of the sequence were considered to be normally produced by the system and both annotators.

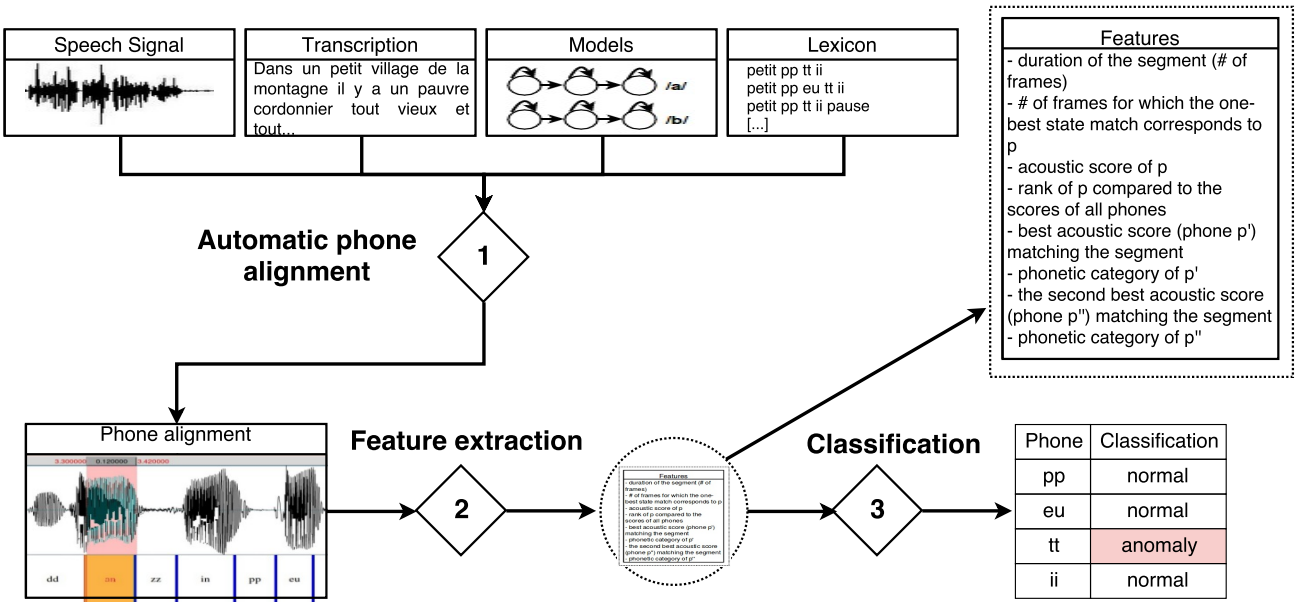


Figure 1: Automatic approach for a phone-level anomaly detection.

Speech sequences are chosen to fit one of the following 4 predefined categories, regarding uniquely the target word(s) (as noted above, the rest of the words in the speech sequences is considered as normal, independently of the categories) :

- 12.5% of the test sequences are referred to as "obvious segments". Here, both annotators agree with the system annotation considering the target word(s) as abnormal. This category is rather limited in size, compared to the others since the authors were more interested by the potentially wrong behavior of the automatic system;
- 37.5% are referred to as "ambiguous segments". Here, the human annotators disagree with each other and are not able to decide whether the automatic annotation of the target word(s) as abnormal is correct or not;
- 25% are referred to as "false negatives". Here, both annotators consider that **the system fails to detect the presence of a true anomaly on the target word(s)**;
- 25% are referred to as "false positives". Here, both annotators consider that the system falsely labels the target word(s) as abnormal.

Other factors shape the set of the speech sequences. First of all, efforts have been concentrated on selecting speech produced by the largest number of patients, and representing the four pathologies available on our corpora. Second, efforts are made to balance the selected sequences and targeted words in order to vary their nature (grammatical, and lexical words), their length (long, and short words) and their position in the sequence (start, middle, and end).
To respond to these different constraints, a total of 98 speech sequences produced by 40 speakers, included 33 dysarthric patients and 7 healthy control speakers, are finally selected for the perceptual evaluation task.

The last feature of the protocol relies on the choice of the participant listeners and their degree of expertise to evaluate the presence or not of abnormal words in the speech sequences. Let's recall that this perceptual evaluation protocol aims at evaluating the quality of the outputs of an automatic system, considered itself as an "expert" - its goal is to bring some objective "expertise" to clinicians or phoneticians in their analysis of speech disorders. It seems natural to demand that listeners, qualified in evaluating such speech disorders, participate in this evaluation protocol. A jury of expert listeners are, therefore, selected.

3. Experimental protocol

3.1. Corpora

All the selected speech sequences are extracted from French read speech recordings of the fairy tale "*Tic Tac*" (The elves and the shoemaker). In total, 98 sequence produced by 40 speakers from dysarthric speech corpora *VML* and *TypALoc* (Meunier et al., 2016) are selected. Four pathologies are represented in these corpora : Parkinson's Disease (PD), Cerebellar Ataxia (CA), Amyotrophic Lateral Sclerosis (ALS), and lysosomal diseases (LYS).
Table 1 details the number of patients and sequences for each pathology and their dysarthric class. The selected segments were extracted from the recordings using Praat (Boersma and Weenink,) and artificial silences of 400 and 200ms were added to each at the beginning and the end respectively in order to avoid abrupt signal cuts for the perceptual evaluation process.

3.2. Jury and experimental design

The selected jury includes 29 experts aged between 22 and 58 (average age of 33). They all have French as their mother tongue and have no prior audition or learning disorders. The jury is composed of 10 speech therapists, 18 final-year speech therapy students and 1 Ear, Nose and Throat (ENT) specialist, and speech pathologist.

Population	Corpora	Dysarthria class	# of speakers	# of sequences
Control speakers	*TypALoc*	-	7	15
Parkinson's disease	*TypALoc*	Hypokinetic	6	15
Cerebellar ataxia	*TypALoc*	Ataxic	8	22
Amyotrophic Lateral Sclerosis	*TypALoc*	Mixed	11	28
Lysosomal storage disease	*VML*	Mixed	8	18
Total	-	-	40	98

Table 1: Information related to the corpora used for the perceptual evaluation task including the different populations and dysarthria class - control speakers and patients suffering from Parkinson's disease, cerebellar ataxia, amyotrophic lateral sclerosis, and lysosomal diseases, the number of speakers and of speech sequences per population.

The proposed perceptual evaluation task is then computerized using the Perceval platform (Ghio et al., 2003). The evaluations last between 25 and 40 minutes and are performed in quite rooms as follows: (1) participants are presented with an instruction list on the screen explaining the test procedure; (2) an oral instruction indicating to focus only on articulatory realization and not to take prosodic or vocal aspects into account is then given to all participants; (3) a training phase of 4 sequences is proposed in order to get participants familiarized with the task and the use of Perceval platform; (4) when the evaluation starts, an orthographic transcription of the sequence appears on the screen. Under each word, the expert has to check one of two boxes to label the word as "**normal**" or "**abnormal**". Figure 2 shows an example screen shot of the experiment. The sequences are presented in a totally randomized order for each participant and can be played up to 3 times before making an evaluation.

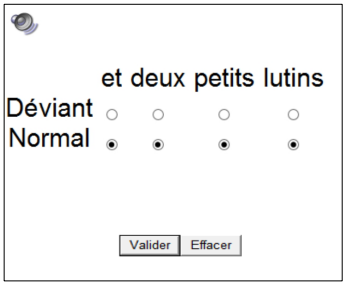


Figure 2: Screen shot from the Perceval platform used in the perceptual evaluation.

3.3. Evaluation methodology

To analyze the perceptual evaluation results, 3 System-Jury agreement rates are computed:

- $AG_targetAnomaly$ rate, measures the System-Jury agreement rate on the targeted abnormal words (target words automatically labeled as abnormal) for the "obvious segments", "ambiguous segments" and "false positives" categories. This rate measures the capacity of the automatic processing in detecting present abnormal zones and how much the jury agrees with it on the detected segments. The closer to 100 the rate is, the better the automatic system detects the abnormal zones;

- $AG_targetNormal$ rate, measures the System-Jury agreement rate on the targeted normal words (target words automatically labeled as normal) for the "false negatives" category. This rate reflects the system inability to detect potential present anomalies (according to the couple of annotators). The closer to 100 the rate is, the better the automatic approach is in distinguishing anomalies from normal words and not labeling them as abnormal;

- $AG_nonTargetNormal$ rate, measures the System-Jury agreement rate on the non target words (automatically labeled as normal and considered as such during the sequence selection by both the annotators) for all test categories ("obvious segments", "ambiguous segments", "false positives" and "false negatives"). This rate will measure the system precision and capacity to distinguish between normal and abnormal words. The closer to 100 the rate is, the better the automatic approach is in not labeling normal words as anomalies.

4. Results and discussions

4.1. Results according to test sequence categories

Figure 3 depicts the distribution of the different agreement rates when computed on each test category.

Considering the $AG_targetAnomaly$ measure, we observe a strong heterogeneity in the results depending on the test category, reaching 78%, 58%, and 13% for "obvious segments", "ambiguous segments" and "false positives" categories respectively. The high $AG_targetAnomaly$ rate on "obvious segments" confirms the capacity of the automatic approach to detect highly distorted segments. This capacity has been already highlighted in (Laaridh et al., 2015a) with 81% of phone-based anomalies annotated by an expert well detected by the system.

In contrast, the low $AG_targetAnomaly$ rate of 13% observed on "false positives" reveals the limitations of the proposed approach and its somehow approximate judgment when facing more subtle anomalies. This result calls for a more in-depth acoustical analysis of these segments in order to better comprehend the automatic system behavior and whether they could be related to acoustic distortions

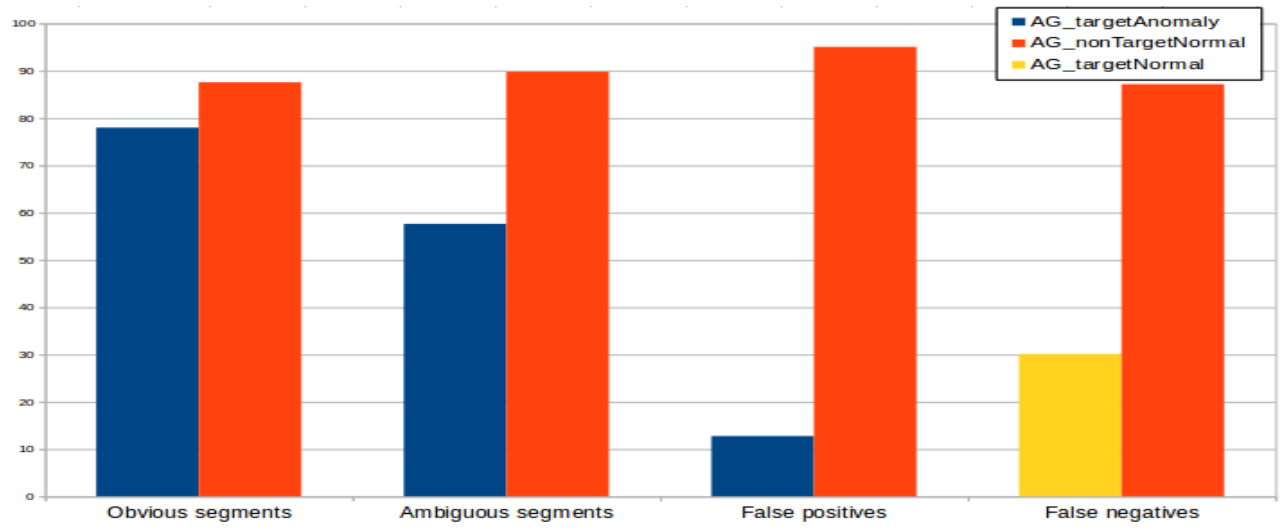


Figure 3: System-Jury agreement rates (%) per test sequence category.

	Obvious segments		Ambiguous segments		False negatives		False positives	
Population	AG_target Anomaly	AG_nonTarget normal	AG_target Anomaly	AG_nonTarget normal	AG_target Normal	AG_nonTarget normal	AG_target Anomaly	AG_nonTarget normal
CTRL	81.0	99.1	15.2	99.7	50.6	97.5	1.4	100.0
CA	71.3	92.3	59.8	86.4	24.9	86.6	9.8	94.6
PD	78.2	89.1	42.7	93.7	64.4	91.2	8.6	97.3
ALS	74.6	52.9	79.0	77.1	25.6	75.3	19.6	98.0
LYS	98.3	81.9	68.1	86.6	8.6	72.9	15.5	85.6

Table 2: System-Jury agreement rates (%) on automatically detected abnormal ($AG_targetAnomaly$) and normal ($AG_targetNormal$ and $AG_nonTargetnormal$) words per population and test sequence category.

of non-pathological nature (noise, breaths, etc.). Other hypotheses can be advanced to explain this behavior such as the presence of true anomalies not detected by the human experts in these segments or the presence of erroneous data in the system training corpora (errors in the human annotation used as reference).

Nonetheless, we support that this observation has to be balanced. Indeed, the System-Jury agreement rate on the non target normal words $AG_nonTargetNormal$, aggregated over all the test categories, reaches 88% and confirms that the behavior of the automatic approach is far from being arbitrary. Therefore, the system behavior on the "false positives" category being finally restrained to a small amount of speech segments, and judged as marginal, it would be more appropriate to describe the behavior of the system as more severe than the human experts.

Considering the two remaining categories, "ambiguous segments" and "false negatives", the System-Jury agreement rates confirm the difficulty and the non trivial nature of the perceptual evaluation of dysarthric speech task even when performed by experts. Here, more than half (58%) and nearly third (30%) of the jury decisions agree with the system on both categories respectively, which demonstrates a strong heterogeneity in the decisions given by the experts themselves.

4.2. Inter-population variability

Table 2 details the System-Jury agreement rates per population and test category. We can note that the best $AG_targetAnomaly$ rate is computed over LYS patients reaching 98.3% and 68.1% on "obvious segments" and "ambiguous segments" respectively. This behavior can be expected considering that this population is involved in the modeling of the abnormal phones in our system and is consistent with previous results in (Laaridh et al., 2015a). This does also highlight the importance of the training phase in our automatic approach and suggests that the use of more data associated with different pathologies and dysarthric classes should improve the system performance, already very promising given the results reported earlier.

Considering the other populations, notable differences are observed between the different pathologies. This is highly important considering that the instructions given to the jury explicitly restrict the evaluation task to the articulatory production of speakers. This behavior is particularly evident on ALS patients on whom the jury annotated the most anomalies compared to other populations and where the $AG_targetAnomaly$ rate reaches 19.6% on the "false positives" category. In contrast, an opposite behavior is observed on CTRL speakers and PD patients for whom an overall good quality of the speech is usually observed and the computed $AG_targetAnomaly$ rate over the "ambiguous segments" reaches 15.2% and 42.7% respectively.

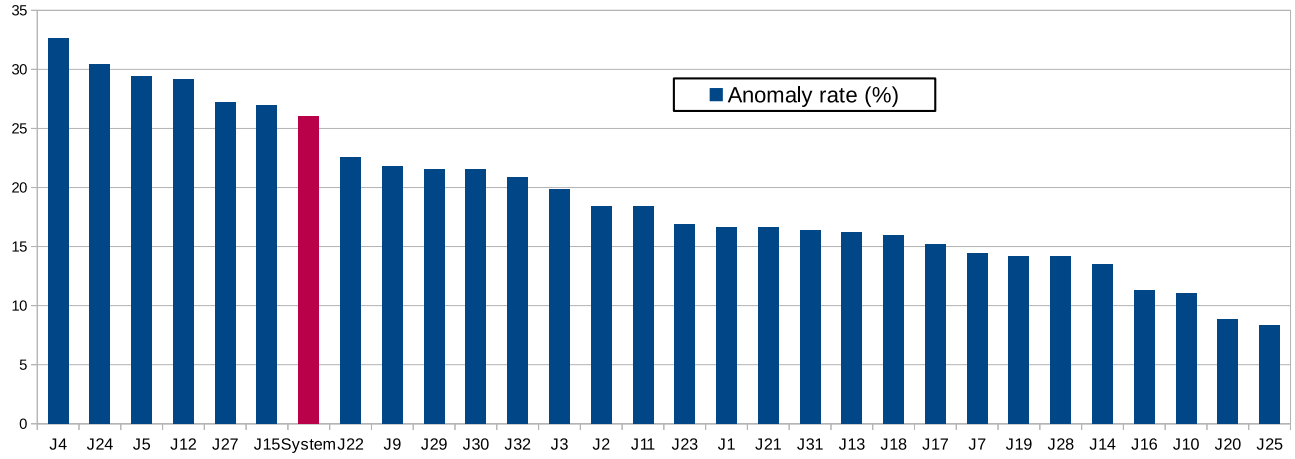


Figure 4: Anomaly rate (%) per jury member (perceptual, blue bars) and for the system (red bar).

4.3. Jury responses

Even though the experimental protocol proposed here aim at studying the behavior of an automatic anomaly detection approach, the computed results highlighted the difficulty for a human, even expert, to apprehend and detect deviations at the word level in dysarthric speech and revealed some evaluation tendencies depending on patients' pathologies and dysarthria severity. Figure 4 depicts the perceptual anomaly rate (%) per jury member.

This tends to show that listeners may be influenced by the contextual information. If speech sounds pathological (ALS patients for instance) then anomalies are more often detected in words. We should therefore ask if listeners are able to perform the same task as the system; the system is able to focus on short units to detect anomalies (phones, syllables, words) while subjects perform a contextual task to take their decision. The variability may be interpreted as a consequence of the difficulty of the task proposed to the jury. Indeed, listeners were asked to focus their attention on a single word which may be produced with or without an anomaly. This is not the way clinicians usually evaluate their patients. And this is also not the way we perceive speech. The process of speech perception requires a large context of speech in order to evaluate if it is distorted or not. The need to focus on a specific item is a very difficult task for listeners.

5. Conclusions

This paper investigates the results of a perceptual evaluation of the annotation performed by an automatic anomaly detection system on dysarthric speech. The results confirm the capacity and relevance of the automatic approach in detecting the presence of anomalies in dysarthric speech (high $AG_targetAnomaly$ rates on "obvious segments"). Moreover, and even on the more nuanced anomalies ("ambiguous segments"), the jury agrees 58% of the time with the automatic approach decisions.

However, the low $AG_targetAnomaly$ rate computed over the "false positives" category confirms the approach tendency to be more severe than the human experts. Considering the limits of the perceptual evaluation recognized in the literature(Zyski and Weisiger, 1987; Fex, 1992), we suggest that a more primitive question must be raised: should an automatic approach replicate the behavior of a human expert and what place should be envisaged for future investigations between supervised (relying on human annotations) and semi- or unsupervised approach ?

6. Acknowledgments

This work has been carried out thanks to the support of the BLRI Labex (ANR-11-LABEX-0036) and the A*MIDEX project (ANR-11-IDEX-0001-02) funded by the French government "Investissements d'Avenir" program managed by the ANR, and thanks to the French ANR projet Typaloc (ANR-12-BSH2-0003-03). We would also like to thank Laura Restivo and Laura Pianelli, who were in charge of the initial speech sequence annotation (the two annotators mentioned in the paper), of recruiting the expert jury and of organizing the perceptual evaluation-based experiments.

7. Bibliographical References

Auzou, P. and Rolland-Monnoury, V. (2006). *Batterie d'évaluation clinique de la dysarthrie*. Édition Ortho.

Boersma, P. and Weenink, D.). Praat: doing phonetics by computer. http://www.praat.org/.

Darley, F. L., Aronson, A. E., and Brown, J. R. (1969a). Clusters of deviant speech dimensions in the dysarthrias. *Journal of Speech and Hearing Research*, 12:462–496.

Darley, F. L., Aronson, A. E., and Brown, J. R. (1969b). Differential diagnostic patterns of dysarthria. *Journal of Speech and Hearing Research*, 12:246–269.

Darley, F. L., Aronson, A. E., and Brown, J. R. (1975). *Motor speech disorders*. W. B. Saunders and Co., Philadelphia.

Duffy, J. R. (2005). *Motor speech disorders: substrates, differential diagnosis and management*. Motsby- Yearbook, St Louis, 2nd edition.

Fex, S. (1992). Perceptual evaluation. *Journal of voice*, 6(2):155–158.

Ghio, A., André, C., Teston, B., and Cavé, C. (2003). Perceval: une station automatisée de tests de perception et d'évaluation auditive et visuelle. *Travaux interdisciplinaires du Laboratoire parole et langage d'Aix-en-Provence (TIPA)*, 22:115–133.

Laaridh, I., Fredouille, C., and Meunier, C. (2015a). Automatic detection of phone-based anomalies in dysarthric speech. *ACM Transactions on accessible computing*, 6(3):9:1–9:24, May.

Laaridh, I., Fredouille, C., and Meunier, C. (2015b). Automatic speech processing for dysarthria: A study of interpathology variability. In *Proceedings of Intl Congress of Phonetic Sciences (ICPHs'15)*, Glasgow.

Laaridh, I., Fredouille, C., and Meunier, C. (2016). Automatic anomaly detection for dysarthria across two speech styles: Read vs spontaneous speech. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz, Slovenia, may.

Meunier, C., Fougeron, C., Fredouille, C., Bigi, B., Crevier-Buchman, L., Delais-Roussarie, E., Georgeton, L., Ghio, A., Laaridh, I., Legou, T., Pillot-Loiseau, C., and Pouchoulin, G. (2016). The TYPALOC corpus: A collection of various dysarthric speech recordings in read and spontaneous styles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portoroz, Slovenia, May.

Scholkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Zyski, B. J. and Weisiger, B. E. (1987). Identification of dysarthria types based on perceptual analysis. *Journal of Communication Disorders*, 20(5):367–378.

Towards an Automatic Assessment of Crowdsourced Data for NLU

Patricia Braunger, Wolfgang Maier, Jan Wessling, Maria Schmidt
Daimler AG
Sindelfingen, Germany
{patricia.braunger, wolfgang.mw.maier, jan.wessling, maria.m.schmidt}@daimler.com

Abstract

Recent development of spoken dialog systems has moved away from a command-style input and aims at allowing a natural input style. Obtaining suitable data for training and testing such systems is a significant challenge. We investigate with which methods data elicited via crowdsourcing can be assessed with respect to its naturalness and usefulness. Since the criteria with which to assess usefulness depend on the application purpose of crowdsourced data we investigate various facets such as noisy data, naturalness and building natural language understanding (NLU) models. Our results show that valid data can be automatically identified with the help of a word based language model. A comparison of crowdsourced data and system usage data on lexical, syntactic and pragmatic level reveals detailed information on the differences between both data sets. However, we show that using crowdsourced data for training NLU services achieves similar results as system usage data.

Keywords: crowdsourcing, spoken dialog system, natural language understanding, training data

1. Introduction

A major goal of current spoken dialog systems (SDS) development is to obtain a more natural and human-like style of communication. Such a communication style presupposes an understanding of all utterances which are associated with a specific semantic meaning. Thanks to the advent of statistical data-driven approaches, recent systems have been able to interpret freely spoken user input. However, obtaining suitable data for their training and evaluation is a significant challenge.

In order to increase the amount of data for training and evaluation of natural language understanding (NLU), researchers are turning to crowdsourcing instead of collecting data from Wizard of Oz experiments or using handcrafted grammars (see, e.g., Callison-Burch and Dredze (2010)). While the benefits of crowdsourcing can include expedition, individuality of people, and low costs (see, e.g., Eskenazi et al. (2013)), it is often criticized for poor standards as it is difficult to control the quality of work when requesting complex tasks (Eskenazi et al., 2013). Since the community of crowd workers is acknowledged for being Internet savvy young adults, chiefly between ages 18 and 35, the findings are not representative for any target group. In addition, crowdsourced data intended to be used for improving SDS do not necessarily reflect real system usage. Therefore we investigate to which extent crowdsourced data are suitable for the development of human-like SDS. Specifically, we want to provide answers to the following questions:

- Do data elicited via crowdsourcing reflect real system usage and naturalness?
- Are data elicited via crowdsourcing suitable to encourage the development of high-quality NLU modules?

Thereby, we aim to propose methods with which to assess crowdsourced data for NLU. In order to establish a baseline of what requirements the utterances collected via crowdsourcing must fulfill, we conduct a study in which free user utterances are collected for an actual in-car SDS. By means of a comparative analysis with crowdsourced data, we first demonstrate to which extent real system usage utterances and crowdsourced utterances share properties in terms of naturalness and variety. Additionally, we address application-related questions that arise with making use of crowdsourcing for training and assessing NLU modules. We examine the impact of noisy data and we aim to answer the question how noisy data can be identified automatically.

The remainder of the paper is structured as follows: In section 2, we review previous literature which aims to evaluate crowdsourcing methods for different application purposes. Next, in section 3, we introduce our crowdsourced and system usage data and the design of our study. Section 4 presents and discusses the experiments and results. Lastly, section 5 serves as the coda of the article.

2. Related Work

In the past years, the speech processing communities have recognized that crowdsourcing is a promising solution to their strong need for data (Eskenazi et al., 2013). In the field of spoken dialog systems, crowdsourcing has been used for evaluation (Yang et al., 2010; Komarov et al., 2013) and acquiring training data for system components such as automatic speech recognition (ASR) (Rothwell et al., 2015; McGraw et al., 2010), natural language understanding (NLU) (Braunger et al., 2016; Wang et al., 2012; Misu, 2014), dialog management (Manuvinakurike et al., 2015; McGraw et al., 2010) and natural language generation (NLG) (Novikova et al., 2016; Mitchell et al., 2014).

The evaluation of the crowdsourced data sets differs depending on research questions and application purposes. The literature that evaluates crowdsourcing approaches for acquiring training and test data can described as follows.

Most of the research works compare different meaning representation modalities with which to elicit data for given meanings or tasks. Depending on the application of the acquired data, e.g., NLG or NLU, different metrics are used in order to find the best elicitation method among the proposed. Novikova et al. (2016), e.g., compare two meaning

representation modalities, namely text and pictures, with which to elicit NLG training data. The measures they use to assess the effect of the meaning representation method include time taken to collect data, average length of utterances, average number of sentences per utterance, semantic similarity, informativeness and naturalness. Naturalness is measured by human evaluation by asking whether the utterance could have been produced by a native speaker.

Braunger et al. (2016) evaluate whether pictures, semantic entities or textual descriptions are suitable for collecting natural language input for a given semantic form in an in-car SDS context. They compare the collected corpora in terms of semantic correctness and linguistic variance. They conclude that a text-based method is suitable for a myriad kinds of tasks, produces a high linguistic variance, and yields a high rate of usable data.

Yang Wang et al. (2012) compare three different crowdsourcing elicitation methods for the collection of utterances which correspond to a given semantic form, namely sentence-based, scenario-based and list-based method. They analyze the acquired data in terms of semantic correctness and the biases that the methods create. Their hypothesis is that if the method creates a bias the crowd workers follow the same slot order as presented. The authors found that where a natural ordering exists, it is captured. However, since slot ordering is only one of the criteria which can be affected by crowdsourcing methods, analyzing slot ordering is not enough to evaluate the utility of crowdsourced data in terms of naturalness. Other criteria that can be affected include wording, sentence types, politeness etc.

While the works described thus far use different criteria for what is natural, other authors evaluate their methods with help of real system-interaction data. Manuvinakurike and DeVault (2015), e.g., conduct their experiment each in the lab and online. They compare their crowdsourced dialog data set to a smaller lab-based data set. However, in terms of naturalness they do not compare the collected dialogs but rather subjective questionnaire ratings. As an example, a question related to naturalness was "I talked to my partner in the way I normally talk to another person".

Misu (2014) investigate crowdsourced ASR and NLU data for situated dialog systems. They collect information seeking queries that contain points-of-interests (POI) in the participants' surroundings such as "What is that blue building on the corner?". They compare the crowdsourced data to utterances generated by a handcrafted grammar in terms of similarity to data collected from users interacting with a situated dialog system in a moving car. With the help of BLEU score (Papineni et al., 2012) they show that crowdsourced data is closer to queries produced in real driving situations than manually created utterances. In addition, their evaluation with test set perplexity indicates that crowdsourced data improves the performance of language models compared to manually created utterances. Nevertheless, it is difficult to decide whether the scores they report are satisfactory for productive use.

For the development of human-like SDS, it is important that the data used for training and testing reflect real system usage. Within most crowdsourcing elicitation methods crowd workers have to fulfill imaginary tasks. Due to a lack of imagination and indirectness data collected via crowdsourcing might produce other kinds of utterances than data collected from real system usage. In order to approve the utility of crowdsourced data real system-interaction should be therefore utilized as a baseline. We aim to examine to which extent crowdsourced queries share linguistic properties with queries collected from real system usage and how well crowdsourced data perform compared to naturally spoken utterances. In addition, application-related questions such as the impact of noisy data have not been addressed thus far.

3. Data Collection Setup

One of our goals is to evaluate the usefulness of crowdsourced NLU data. Therefore, we evaluate our German data set acquired from crowdsourcing against data collected by an experimental study, in order to gain a baseline of what constitutes naturalness in user input. In the following, we explain the experimental setup and procedure of our investigation.

3.1. Crowdsourced Data

As crowdsourced data, we use a subset of the data we collected in a previous study (cf. Schmidt et al. (2015) and Braunger et al. (2016)). In a previous work, Braunger et al. (2016) compares three data elicitation methods which differ in how the tasks are presented to the participants, namely by pictures, semantic entities and text. Considering that a text-based method was found to suite best[1], we apply this method within the development process of an actual in-car SDS. For this reason, our experiments are based on utterances which are elicited via text descriptions following Braunger et al. (2016).

Our crowdsourced data is collected using the German crowdsourcing platform Clickworker[2]. First, crowd workers are asked to spontaneously formulate and record a voice command directed at solving an assigned problem, such as entering an address in the navigation application. Second, the crowd workers are asked to transcribe their utterances into a textual form[3]. The transcription has to be an exact match of their spoken utterances.

> "Imagine that you are travelling by car.
> For a little change you would like to switch to
> the radio station SWR3. What would you say?"

Figure 1: Textual Task Description (Task 1).

[1] The task presentation methods were compared in terms of linguistic variety, number of valid utterances and priming effects. Overall, the authors recommend making use of the text presentation method since it is a good compromise between a high rate of valid utterances, linguistic variety and the possibility of creating very specific tasks.

[2] http://www.clickworker.com/

[3] The analysis of the recordings themselves is beyond the scope of this paper. The quality of the audio data has been examined by Schmidt et al. (2015).

The tasks are presented to the crowd workers by means of textual descriptions of the situation in which they are in, as well as the actions they should perform. An example is given in Fig. 1.
With the help of the textual descriptions we requested speech input for seven tasks typically performed in a car:

1. Listen to radio station SWR3
2. Play Michael Jackson Greatest Hits
3. Next Shell gas station
4. Navigate to Stieglitzweg 23 in Berlin
5. Call Barack Obama on mobile phone
6. Set temperature to 23 degrees
7. Send text message to brother

For each of the seven tasks we collect 1,080 spoken and transcribed utterances. The 1,080 crowd workers are German native speakers. 40% of the crowd workers are female and 60% are male. 90% of the crowd workers are between the ages of 18 and 35.8% are between the ages of 36 and 55, and 2% are above the age of 55.

3.2. System Usage Data

As we want to examine to which extent utterances used for system interaction in a realistic situation share properties with artificial crowdsourced data, we use data from a previous study in which users had to freely speak to an actual in-car SDS within a Wizard of Oz (WOZ) experiment (Braunger et al., 2017). These WOZ data serve as gold standard data in the experiments below.
Among the tasks the participants have to solve are the same aforementioned seven tasks. In order to avoid bias, i.e. offering the participants verbal examples, the tasks are presented by pictures as opposed to textual descriptions. These pictures are first pre-tested with friendly users to evaluate if the desired situation was put in the user's mind. An example of the task description is given in Fig. 2.

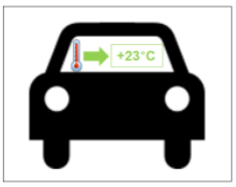


Figure 2: Graphical Task Description (Task 6).

The system behavior is simulated with the SUEDE tool (Klemmer et al., 2000) and designed such as available in a current Mercedes-Benz E-class. The participants are told that the system is able to understand any utterance in the given context. To activate the speech recognition engine, the participants have to speak the phrase "Hallo Auto" (eng. "Hello car"). After speaking their request the system directly activates the appropriate function or provides the requested information. For example, when given user input for Task 1, the radio program begins to play and the screen provides information for the current radio station.

Considering that we wish to discover how users naturally interact with a SDS while driving, we place the participants in a simulated driving situation. The car in which the participants are placed is situated in front of a canvas onto which the driving simulation is projected, as done by Hofmann et al. (2014). In the driving simulation, the participants drive behind another vehicle and they have to brake if and only if the preceding vehicle brakes. The setup is illustrated in Fig. 3.



Figure 3: Driving Simulation Setup.

The overall procedure of the experiment is as follows. First, the participants are shown the pictures which they have to interpret verbally. In order to prevent incorrect interpretations, the participants are offered assistance when necessary. Second, the participant is quickly acclimated to the driving simulation through a three minutes test drive. The instructor, who sits in the passenger seat, shows the pictures arbitrarily while the participant operates the vehicle. The tasks are permuted to avoid order effects. More details are described in Braunger et al. (2017).
Since 45 subjects participated in the study, 45 utterances per task are collected respectively (in total: 315 utterances). 46% of the participants are female and 54% are male. The average age is 39.5 years (standard deviation SD: 13.5). 27% of the participants are experienced in the use of voice-controlled devices while 74% have little to no experience with SDS.

4. Experiments and Results

As normalizing preprocessing steps, we automatically spell-check the data, standardize the spelling, lowercase it, and eliminate all punctuation. For further analysis we POS-tag and parse the data using *SpaCy*[4]. The POS tagger uses the Google Universal POS tag set of Petrov et al. (2012).

4.1. Noisy Data

When it comes to assess crowdsourced data many researchers examine rates of unusable data. Previous works report rates of unusable data between 6% and 30% (Wang et al., 2012; Schmidt et al., 2015; Braunger et al., 2016). Yang Wang et al. (2012) find incorrect slot values, missing or added slots and garbage utterances. In addition, Braunger et al. (2016) find technical problems and task misunderstandings that cause faulty data. We also expect our data con-

[4]https://github.com/explosion/spaCy

tains such mistakes. The experiments we have conducted confirm this assumption.
In order to create a baseline of valid utterances we first define "keywords" that have to be named. The keywords are defined such that they represent the minimal information which is required to fulfill the predefined tasks. As an example, at least the radio station name "SWR3" needs to be mentioned to fulfill task 1. The utterances are automatically annotated and manually corrected. This procedure leads to a rate of 4.6% unusable utterances within our crowdsourced data[5].
In order to automatically identify such erroneous utterances we propose an approach based on a maximum likelihood language model (LM) which is also applicable for more complex tasks. We train a word based trigram model since it performed best within our experiments[6]. As training data, we use the hundred most frequent utterances of the preprocessed crowdsourced data. Since we observe that the hundred most frequent utterances are repeated several times by different speakers we expect this data do not contain mistakes. Before we test the model on the whole crowdsourced data set we remove the stop words in both, training set and test set. The score of an utterance x is calculated such that the impact of the utterance length UL is normalized:

$$LMscore(x) = LaplaceSmoothing^{\frac{1}{UL}} \quad (1)$$

The results are filtered by an experimentally determined threshold value. All utterances, whose probabilities are more than 80% lower than the mean score, are classified as faulty. With the help of the LM we are able to identify 83% of the faulty utterances. The remaining utterances contain only 0.8% incorrect utterances which are not captured. Examples of the mistakes we identify are given below.

a) Wrong language (instead of German):
i tell my car to switch on to 23 degrees celsius.

b) Garbage utterance:
telefonieren mit amerika 001 amerikanischer präsident rufnummer suchen barack obama telefonnummer amerika barack obama.
"Call America 001 American president search telephone number Barack Obama phone number America Barack Obama."

c) Technical problem:
aufnahme lässt sich nicht abspielen.
"The sound file cannot be played back."

d) Technical problem:
23 grad wurde wegen technische aufnahmen nicht richtig aufgenommen.
"23 degrees was not correctly recorded due to technical issues."

e) Typing error:
swchlate radio swr3.
"Typing error *radio SWR3.*"

f) Wrong intent:
auto bitte öffne in meinen handy das sms schreiben und suche den kontakt meines bruders.
"Car, please open text messaging function on my mobile and search the entry of my brother."

g) Wrong intent:
auto bitte lass das lied the way you make me feel laufen.
"Car, please play the song The way you make me feel."

h) Missing slot:
spiel das album greatest hits.
"Play the album Greatest Hits."

i) Wrong slot value:
schreibe max mustermann eine sms.
"Write a text message to Max Mustermann."

j) Task misunderstanding:
ich könnte mir vorstellen dass es gut wäre wenn ich das lied summe oder singe wird es gefunden.
"I believe it would be good if the song I am singing can be found."

Even though 15% of the truly valid utterances are classified as faulty, the high precision score for the valid utterance class (99%) shows that the model is able to robustly identify truly valid utterances. Overall, the model achieves an accuracy rate of 85% and an F_1 score of 89%. With task 3 the model performs worst and with task 2 it performs best.

4.2. Linguistic Evaluation

In order to evaluate the usefulness of crowdsourcing we check the data set acquired by a previously evaluated crowdsourcing method against data acquired by an experimental study. Since our system usage data serves as a baseline for naturalness in user input, both data sets are examined and compared in terms of syntactic, lexical, and pragmatic criteria commonly used in literature. The criteria we examine also include those mentioned by literature for natural queries: politeness, full sentences, filler words and a higher number of words (Braunger et al., 2017).
Table 1 presents the lexical properties of both, crowdsourced queries and real system usage queries (referred to as "natural data"). A common measure of lexical diversity is the type-token ratio which is calculated by dividing the number of individual word types (lemmas) by the number of occurring word tokens. The standardized type-token ratio (STTR) (Johnson, 1944) is commonly used to normalize the impact of the size of different data sets. Table 1 shows that there are no significant differences between

[5]Note that the rates of usable data may vary depending on how valid utterances are defined. Following Schmidt et al. (2015) and Braunger et al. (2016) valid utterances must contain given entities in the form of proper names such as "SWR3" and additionally variable units such as "radio", "radio station" or "station". This constraint leads to a rate of 13% unusable utterances.

[6]Within this work several language models have been experimented with: probabilistic language models and recurrent neural network based language models. Each main kind was trained on word and on character basis. The maximum likelihood model trained on word based trigrams performed best concerning the task.

crowdsourced queries and real system usage queries.
The content-function word ratio is an indicator for lexical density. The ratio contains the proportion of content words (open class words) to the number of function words (closed class words). With 68.45% in crowdsourced queries and 60.9% in natural queries people seem to provide less information when speaking to an actual system.
The keyword-to-utterance-length ratio (KLR) is mentioned by Möller et al. (2008). "Keywords" represent obligatory contents to occur in each valid query. The ratio is calculated by the number of keywords in an utterance divided by the number of tokens in an utterance. The KLR computed for both, crowdsourced queries and natural queries, shows that people tend to get to the point when fulfilling the tasks via crowdsourcing.

Lexical criteria	Crowdsourced data	Natural data
STTR	34.93%	35.26%
Keyword-utterance length ratio	37.73%	30.22%
Content-function word ratio*	68.45%	60.90%

Table 1: Lexical properties of crowdsourced data and natural data. (*Differences are significant at $p<0.05$.)

The syntactic properties are presented in Table 2. A comparison of the utterance lengths shows that utterances towards the system are significantly longer than the crowdsourced utterances. Tree depth is used by Pinter et al. (2016) and Guy (2016) as an indicator for syntactic complexity. It is calculated by the number of edges in the longest path from the root node to a leaf. The mean tree depth of the system-directed utterances is 2.94 and of the crowdsourced utterances 2.33. That is, people tend to use more complex syntactic structures when talking to an actual system. In order to conclude the syntactic analysis we examine whether people used full sentences or syntactic incomplete structures. Therefore, we rely on Braunger et al. (2017) who defined the following sentence type categories: Interrogative, Declarative, Imperative, Infinitive and Verbless, whereas an infinite and a verbless construction do not count as full sentences. For annotation, we use a statistical model[7]. The results of the model are inspected and corrected manually. As Table 2 implies, 80% of the natural queries are full sentences but only 70% of the crowdsourced queries. In addition, natural queries contain a lot more declarative constructions such as "I would like to call Barack Obama on his mobile phone" whereas crowdsourced queries contain more imperative constructions.
Besides lexical and syntactic aspects we analyze our data in terms of pragmatic properties. Civility and filler words are mentioned by some authors being salient features of natural language compared to a command or keyword style of speaking to an SDS (Berg et al., 2010; Hofmann et al., 2012). As for politeness, we analyze the occurrences of the

[7]The statistical model bases on POS tag unigrams and bigrams and achieves an accuracy of 91% on similar crowdsourced data.

Syntactic criteria	Crowdsourced data	Natural data
Mean query length	5.72	6.76
Mean tree depth	2.33	2.94
Proportion of Interrogative	3%	7%
Proportion of Declarative*	5%	27%
Proportion of Imperative*	62%	46%
Proportion of Infinitive	19%	15%
Proportion of Verbless*	11%	5%

Table 2: Syntactic properties of crowdsourced data and natural data. (*Differences are significant at $p<0.05$.)

particle "please" and we additionally rely on the findings of Danescu-Niculescu-Mizil et al. (2013) who characterize politeness indicators (cf. Schmidt and Braunger (2018)). Out of the 14 strategies they mention as being polite, we examine the following:

- Counterfactual modal: *Could/Would* you
- Indicative modal: *Can/Will* you
- 1st person start: *I* search
- 1st person plural: Could *we* find

The proportion of utterances that contain these politeness markers is shown in Table 3. The proportion of polite queries in the natural data is nearly five times higher than in the crowdsourced data. Additionally, we compare the proportion of utterances containing filler words. Filler words do not contribute to the sentence meaning. The filler words we count include disfluencies such as "ähm" (eng. "uh") and modal particles according to Bross (2012). The modal particles we find include "doch", "einmal", "nochmal", "mal", "vielleicht", "denn", "eigentlich". Table 3 shows that filler words occur more often when people were speaking to an actual SDS.

Pragmatic criteria	Crowdsourced data	Natural data
Proportion of polite queries*	13.7%	60.9%
Proportion of queries that contain filler words	4.4%	9.0%

Table 3: Pragmatic properties of crowdsourced data and natural data. (*Differences are significant at $p<0.05$.)

Overall, we find quite striking differences between crowdsourced data and real system usage data. This arises the question whether crowdsourced data contribute to the goal of enabling a more human-like SDS interaction and thus whether crowdsourced data are as useful as real system usage data. In order to answer the question, we conduct an experimental evaluation which is described in the following section.

4.3. Application-Related Evaluation

Since NLU services are already widely used in both, academia and industry, we rely on NLU services to evalu-

ate the performance of our two data sets. We use Microsoft LUIS[8] and RASA[9] since Braun et al. (2017) find that those achieve the best results compared to other popular NLU services. The data sets were semi-automatically labeled by the authors.

First, the NLU services are trained on crowdsourcing data as well as on 60% of the system usage data. As crowdsourced data, we use the utterances which are classified as valid utterances (cf. section 4.1.). The labels created by the services are then compared against 40% of our system usage gold standard data. Fig. 4 shows that the average F_1 scores are very similar between crowdsourced training data and system usage training data. The models that are trained on crowdsourced data achieve very good results when tested on gold standard data.

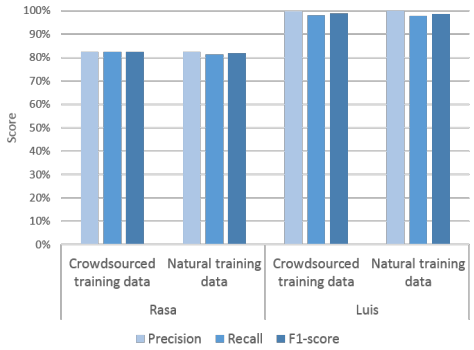


Figure 4: Average scores of intent classification and entity recognition: crowdsourced vs. natural data.

Since we identified that 4.6% of the crowdsourced data contain errors we additionally investigate the impact of such noise on performance results. In a further experiment the NLU services are therefore trained on both 100% of crowdsourced data and cleaned-up crowdsourced data (95.4%). The test set in that case consists of all gold standard utterances (100%). Fig. 5 displays the results. It is shown that the noisy data perform as well as the cleaned-up data. This can be due to the fact that the amount of training utterances is increased in this case. However, this result should be viewed with caution. Further experiments have to be conducted taking into account a more difficult task, e.g., more overlapping use cases.

5. Conclusion

We propose different methods of how to evaluate usefulness and naturalness of crowdsourced data. Therefore, we create a baseline by means of collecting real system usage data.

In order to automatically assess the quality of the crowdsourced data, we first train a word based language model on the hundred most frequent utterances and test the model on all crowdsourced utterances. We show that this method is able to successfully identify valid data. Since many works report high proportions of faulty utterances within crowdsourced data the amount of mistakes is an important criterion when it comes to assess crowdsourced data.

[8]https://www.luis.ai

[9]https://www.rasa.ai (Open source service).

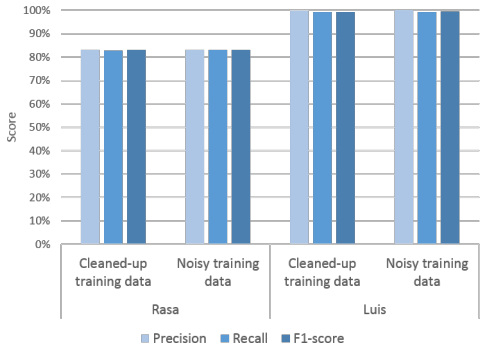


Figure 5: Average scores of intent classification and entity recognition: cleaned-up crowdsourced data vs. noisy crowdsourced data.

In order to assess the naturalness of the voice input elicited via crowdsourcing, we examine different linguistic criteria on a lexical, syntactic and pragmatic level. Our comparative analysis reveals the differences between crowdsourced data and system usage data. However, we show that when training NLU services on crowdsourced data the scores achieved are as good as system usage data, even when the test set contains faulty utterances. We conclude that for the purpose of training NLU services crowdsourced data is at least as suitable as system usage data.

6. Bibliographical References

Berg, M., Gröber, P., and Weicht, M. (2010). User study: Talking to computers. In *Proceedings of the Third Workshop on Inclusive eLearning*.

Braun, D., Mendez, A. H., Matthes, F., and Langen, M. (2017). Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*.

Braunger, P., Hofmann, H., Werner, S., and Schmidt, M. (2016). A comparative analysis of crowdsourced natural language corpora for spoken dialog systems. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia. European Language Resources Association (ELRA).

Braunger, P., Maier, W., Wessling, J., and Werner, S. (2017). Natural language input for in-car spoken dialog systems: How natural is natural? In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*.

Bross, F. (2012). German modal particles and the common ground. *Helikon. A Multidisciplinary Online Journal*, 2:182–209.

Callison-Burch, C. and Dredze, M. (2010). Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*.

Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J., and Potts, C. (2013). A computational approach to politeness with application to social factors.

In *Proceedings of the Conference of the Association for Computational Linguistics (ACL)*.

Eskenazi, M., Levow, G.-A., Meng, H., Parent, G., and Suendermann, D. (2013). *Crowdsourcing for Speech Processing: Application to Data Collection, Transcription and Assessment*. John Wiley & Sons.

Guy, I. (2016). Searching by talking: Analysis of voice queries on mobile web search. In *Proceedings of 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Hofmann, H., Ehrlich, U., Berton, A., and Minker, W. (2012). Speech interaction with the internet - a user study. In *Proceedings of the Eighth International Conference on Intelligent Environments*.

Hofmann, H., Hermanutz, M., Tobisch, V., Ehrlich, U., Berton, A., and Minker, W. (2014). Evaluation of in-car sds notification concepts for incoming proactive events. In *Proceedings of 5th International Workshop on Spoken Dialog Systems*.

Johnson, W. (1944). Studies in language behavior: I.a program of research. *Psychological Monographs*, 56:1–15.

Klemmer, S. R., Sinha, A. K., Chen, J., Landay, J. A., Aboobaker, N., and Wang, A. (2000). Suede: A wizard of oz prototyping tool of speech user interfaces. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology*.

Komarov, S., Reinecke, K., and Gajos, K. Z. (2013). Crowdsourcing performance evaluations of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

Manuvinakurike, R. and DeVault, D., (2015). *Natural Language Dialog Systems and Intelligent Assistants*, chapter Pair Me Up: A Web Framework for Crowd-Sourced Spoken Dialogue Collection, pages 189–201. SpringerLink.

Manuvinakurike, R., Paetzel, M., and DeVault, D. (2015). Reducing the cost of dialogue system training and evaluation with online, crowd-sourced dialogue data collection. In *Proceedings of the Workshop on Semantics and Pragmatics of Dialogue 2015*.

McGraw, I., Lee, C., Hetherington, L., Seneff, S., and Glass, J. R. (2010). Collecting voices from the cloud. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*.

Misu, T. (2014). Crowdsourcing for situated dialog systems in a moving car. In *Proceedings of Interspeech*.

Mitchell, M., Bohus, D., and Kamar, E. (2014). Crowdsourcing language generation templates for dialogue systems. In *Proceedings of the Eighth International Natural Language Generation Conference (INLG)*.

Möller, S., Gödde, F., and Wolters, M. (2008). Corpus aanalysis of spoken smart-home interactions with older users. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*.

Novikova, J., Lemon, O., and Rieser, V. (2016). Crowdsourcing nlg data: Pictures elicit better data. In *Proceedings of the 9th International Natural Language Generation Conference*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2012). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Petrov, S., Das, D., and McDonald, R. (2012). A universal part-of-speech tagset. In *Proceedings of 8th International Conference on Language Resources and Evaluation (LREC)*.

Pinter, Y., Reichart, R., and Szpektor, I. (2016). Syntactic parsing of web queries with question intent. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Rothwell, S., Elshenawy, A., Carter, S., Braga, D., Romani, F., Kennewick, M., and Kennewick, B. (2015). Controlling quality and handling fraud in large scale crowdsourcing speech data collections. In *Proceedings of Interspeech*.

Schmidt, M. and Braunger, P. (2018). Towards a speaking-style adaptive assistant for task-oriented applications. In *Proceedings of 29. Elektronische Sprachsignalverarbeitung (ESSV)*.

Schmidt, M., Müller, M., Wagner, M., Stüker, S., Waibel, A., Hofmann, H., and Werner, S. (2015). Evaluation of crowdsourced user input data for spoken dialog systems. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*.

Wang, W. Y., Bohus, D., Kamar, E., and Horvitz, E. (2012). Crowdsourcing the acquisition of natural language corpora: Methods and observations. In *Proceedings of the IEEE Workshop on Spoken Language Technologies (SLT)*.

Yang, Z., Li, B., Zhu, Y., King, I., Levow, G., and Meng, H. (2010). Collection of user judgements on spoken dialog system with crowdsourcing. In *Proceedings of the IEEE Workshop on Spoken Language Technology (SLT)*.

Visual Choice of Plausible Alternatives: An Evaluation of Image-based Commonsense Causal Reasoning

Jinyoung Yeo POSTECH
Gyeongbok Lee* Yonsei University
Gengyu Wang* Yonsei Unversity
Seungtaek Choi Yonsei Unversity
Hyunsouk Cho POSTECH
Reinald Kim Amplayo Yonsei University
Seung-won Hwang Yonsei University
{jinyeo, prory}@postech.edu {alias_n, posuer, hist0613, rktamplayo, seungwonh}@yonsei.ac.kr

Abstract

This paper proposes the task of Visual COPA (VCOPA). Given a premise image and two alternative images, the task is to identify the more plausible alternative with their commonsense causal context. The VCOPA task is designed as its desirable machine system needs a more detailed understanding of the image, commonsense knowledge, and complex causal reasoning than state-of-the-art AI techniques. For that, we generate an evaluation dataset containing 380 VCOPA questions and over 1K images with various topics, which is amenable to automatic evaluation, and present the performance of baseline reasoning approaches as initial benchmarks for future systems.

Keywords: Commonsense knowledge, Causality, Reasoning, Evaluation, Image understanding

1. Introduction

Commonsense causal reasoning is one of the fundamental research problems in Knowledge Representation & Reasoning (KR) domain. It aims at understanding the general causal dependency between common events or actions. Recent efforts for such understanding are focused on measuring the plausibility of one event statistically leading to another, and in particular, are competing on an evaluation set called *Choice of Plausible Alternatives* (COPA) (Roemmele et al., 2011), which is to select the more plausible alternative as a cause (or effect) of the premise as:

Example 1 *Premise:* A janitor is cleaning the floor. *What is cause?*
Alternative 1: There is a broken cup on the floor.
Alternative 2: There is a cup of coffee on the table.

For the purpose of this reasoning, the state-of-the-art, called CausalNet (Luo et al., 2016), harvests causality scores of cause-effect term pairs, *e.g.*, ('broken','clean'), by mining their causal patterns, *e.g.*, "If...broken ..., then... clean...", from an extremely large text corpus (10TB). As a result, it achieved a remarkable accuracy (70.2%) from COPA.

In the real world, the desirable reasoning ability should be required not only in KR domain but also in Computer Vision (CV) domain. Toward the optimal goal of human-level intelligence, CV researchers have actively studied on Visual/Video Question Answering (VQA) (Antol et al., 2015; Ye et al., 2017; Zhao et al., 2017), which is to understand textual questions and images and give correct textual answers by machine. However, such QA tasks are at an early stage of "reasoning", being limited to object-level reasoning, *e.g.*, "What is the man holding in his hand?".

Beyond object-level, the *event-level* visual reasoning is at intersection between the top of KR and CV capabilities, as the boundaries between the two domains are crumbling down due to the huge success of neural-based image-to-text or text-to-image converting techniques (Karpathy et al., 2014; Jiang et al., 2017; Vinyals et al., 2017). We argue these two domains are in complementary nature (Aditya, 2017; Aditya et al., 2015). For example, unlike existing learning of end-to-end signal matching (*e.g.*, image-to-object) in CV, commonsense and background knowledge in KR can help rectifying noise in visual inference. Also, unlike existing language-specific reasoning in KR, visual detection and captioning can help realizing more general reasoning scenarios.



Figure 1: An example of VCOPA question. If a premise image is effect, plausible alternatives image should be cause, and vice versa. Red mark indicates correct answers.

In this paper, we thus propose a new reasoning task and its evaluation dataset, called Visual COPA (VCOPA) as a variant of COPA, which covers the visual questions for commonsense causal reasoning. Specifically, Figure 1 illustrates an example of VCOPA question, which is converted from a textual question on COPA (Example 1) into its corresponding visual question with three images. Similar to COPA, given a premise image, the goal of the VCOPA task is to identify a more plausible alternative image. As a public dataset including over 1K images, VCOPA is amenable to automatic quantitative evaluation, making it possible to effectively track progress on this task.

As a baseline, we leverage CausalNet by reducing image-based questions to text-based question with the state-of-the-art neural image captioning technique (Vinyals et al., 2017). Although this approach cannot achieve significantly better performance when compared to the random baseline

*Authors in alphabetical order with equal contribution

of 50% accuracy, it can be a starting point on the task for the multiple communities such as KR and CV. As this guidance, VCOPA poses a rich set of challenges, many of which have been viewed as the holy grail of automatic image understanding and causal reasoning in general. However, it includes several components that the KR and CV communities have made significant progress on during the past few decades. Thus, we provide an attractive list of solution techniques accessible enough for the communities to start making progress on the VCOPA task.

2. VCOPA Dataset Collection

This section presents the Visual Choice of Plausible Alternatives (VCOPA) dataset by describing our process of collecting image (and text) questions.

2.1. Image Collection

The VCOPA task consists of 380 questions of commonsense causality with 1,140 images. The image question set was created using a specific collecting methodology that ensured breadth of topics on images and quality of the questions. We now explain the details.

Similar to COPA, the first major concern of the collecting methodology is the breadth of the image question set. Our approach is to identify question topics as inspired by COPA questions, which is already validated with a high degree of breadth, and then apply these topics to collecting premise and alternative images through our own creativity. This approach helps balance the generative and analytic aspects of this task, ensuring that the skewed topic interests of the image collectors are not over-represented in the question set, but still allowing for the creative design solutions that each of these questions required.

More specifically, as shown in Example 1 and Figure 1, we first try to directly convert the textual question to its semantically equivalent image question. As a result, we made 224 image questions, which can be compared with their corresponding 224 COPA textual questions. The rest of the 776 COPA questions is difficult to convert to images. Below is an example:

Example 2 *Premise:* The engine of the airplane was faulty. *What is effect?*
Alternative 1: The airplane crashed.
Alternative 2: The pilot made an error.

In the above example, although *Alternative 1* can be easily visualized to an image, *Alternative 2* is hard to visualize using a single image. In this case, we leverage our own creativity to make a new question, taking into account the topic inspired by COPA as possible as we can. To illustrate, Figure 2 recasts Example 2 into a different problem while capturing ideas about 'airplane'. This process made 276 image questions, which is not completely matched with COPA textual questions.

A challenging part of designing VCOPA questions is to establish the incorrect alternative for each question. This image is intended to be similar in form to the correct alternative image, and somewhat related to the premise image, but with no obvious causal connection, especially by eight



Figure 2: VCOPA question recasted from Example 2

main challenges, as we discuss later in Section 4. This design is intended to ensure that answering these questions requires both computer vision techniques and commonsense knowledge harvesting, and cannot be easily answered when using an individual technique. As a result, we made 500 questions, each of which has one premise image and two alternative images.

The second major concern of the collecting methodology is the quality of questions, that is, the strong agreement among human raters who were asked to answer each question. To validate the set, we enlisted the help of 10 volunteers, each of which validates the overall set of 500 questions. Agreement between raters was high (Cohen's K=0.942). In all, at least one volunteer answered 120 questions differently than was intended by the collector of the question. We strictly remove these 120 questions from the set, then the final set contained 380 questions, each of which has the perfect agreement among raters. Especially, 148 questions in this question set are interchangeable with COPA questions with the same meaning.

The order of the question set collected by the image collectors is randomized and the position of the correct alternative image is also randomized, ensuring that a random baseline would answer exactly 50% of the questions correctly.

2.2. Text Annotation

The VCOPA task focuses on causal reasoning with only images (and their metadata automatically generated by machine, *e.g.*, image captions) as input. Despite this fact, we aim at supporting other research scenarios, for example, mutlimodal questioning with image and human-generated text, *i.e.*, the combination of Figure 1 and Example 1. For this purpose, while the selected 148 questions are matched with their corresponding COPA questions, we generate the COPA-like text questions for the rest of questions. For example, the VCOPA question in Figure 2 is matched with a COPA-like question as follows:

Example 3 *Premise:* A plane is landing at the airport. *What is effect?*
Alternative 1: The plane is burning on the ground.
Alternative 2: The ship is burning on the ocean.

When generating text questions, we basically follow the authoring guidance in (Roemmele et al., 2011). For this task, six volunteers, mutually exclusive to the original image collectors, are asked to generate the texts, to exclude the subjectivity of intending causality context. Other than the supplementary files, the VCOPA dataset including visual and textual questions is accessible in our github site[1].

[1] https://github.com/antest1/VCOPA-Dataset

3. VCOPA Task Analysis

3.1. Reasoning Baselines

The VCOPA evaluation is designed so that a random baseline system, where one of the two alternative images is randomly chosen for each question, would perform at exactly 50%. In addition, we adopt a somewhat stronger baseline based on the state-of-the-art causal reasoning system in language domain, and investigate its reasoning performance. While we do not expect these baselines to be competitive with future sophisticated approaches in the CV and KR domains, successful systems must demonstrate improvements over these baseline results.

Our baseline approaches explore the simple idea that visual causality can be converted into textual causality by automatic image captioning techniques, as the causal inference achieved a high performance on COPA task. Accordingly, one would expect that causality between words in the captioning sentences captures image causality as well. Let L_p, L_{a_1}, and L_{a_2} be each automatic captioning sentence of p, a_1, and a_2, by a state-of-the-art system (Vinyals et al., 2017). Then, the more plausible alternative a^* can be identified as:

$$\begin{aligned} a^* &= \underset{a\in\{a_1,a_2\}}{\text{argmax}}\ \text{plausibility}(p,a) \\ &\approx \underset{a\in\{a_1,a_2\}}{\text{argmax}}\ \text{plausibility}(L_p,L_a) \\ &= \underset{a\in\{a_1,a_2\}}{\text{argmax}}\ \frac{1}{|L_p|+|L_a|}\sum_{t_i\in L_p}\sum_{t_j\in L_a} CS(t_i,t_j) \end{aligned} \tag{1}$$

where $CS(t_i, t_j)$ is a causality score between a cause term t_i and an effect term t_j extracted from CausalNet.

CausalNet (Luo et al., 2016) is a weighted and directed graph $G(L, E, W)$ with nodes (lemmatized English terms) $L = \{t_1, t_2, ...\}$ and edges (causal relations) E. The edge weights are captured by the function $W : E \rightarrow [0, 1]$. The weight $w_{i,j}$ associated with an edge (t_i, t_j) represents the causality score, denoted as $CS(t_i, t_j)$, of a cause t_i and an effect t_j. Causality scores depend on the number of occurrences that two terms t_i and t_j are in linguistic patterns known as causal cues (Chang and Choi, 2004) identifying precise cause/effect roles, *e.g.*, "If...t_i ..., then... t_j..." and "... t_j..., because...t_i ...". That is, as more occurrences of (t_i, t_j) in causal cues, its causality score is higher as:

$$W : w(t_i, t_j) = CS(t_i, t_j) \propto freq(t_i, t_j) \tag{2}$$

where $freq(t_i, t_j)$ is the frequency of observing the causal pair (t_i, t_j) from an English corpus. We omit the details of the list of causal cues and Eq. 2 and refer the readers to (Luo et al., 2016).

Despite building on a rather simple and shallow text analysis, by leveraging the scale and richness from a extremely large (10TB) text corpus, CausalNet achieves the state-of-the-arts accuracy on COPA tasks. The corpus contains 1.6B web pages, which result in 64,436 nodes in CausalNet.

3.2. Reasoning Results

Table 1 shows the reasoning performance on COPA and VCOPA evaluation. As reported in (Luo et al., 2016), although CausalNet formally achieved 70.2% accuracy on COPA evaluation, its accuracy on the overlapped set between COPA and VCOPA is 67.9%. Compared to this accuracy, replacing the manual annotation by the automatic caption gains much lower accuracy 52.7%.

Table 1: Reasoning performance on VCOPA evaluation

Dataset	Method	Accuracy
COPA ∩ VCOPA (148 questions)	Automatic Caption	52.7
	Manual Annotation	67.9
Test Set (190 questions)	Automatic Caption	54.2
	Manual Annotation	56.3
Dev Set (190 questions)	Automatic Caption	53.2
	Manual Annotation	55.8
Test + Dev (380 questions)	Automatic Caption	53.7
	Manual Annotation	56.1

In the overall VCOPA dataset, we compare using machine-generated captions with using human-generated annotations again. As a result, although using the captioning technique achieves a better accuracy than the random baseline, we find that using automatic captions cannot outperform the manual annotations. Note that as the VCOPA questions are designed to infer visual causality not textual causality, the average accuracy of their manual annotation is lower than that in COPA while the accuracy of automatic caption is consistent among all data divisions. Despite this trend, manual annotation is consistently better than automatic caption.

These results also suggests the limitation of our baseline approach, that is, using only textual information cannot achieve the high accuracy in VCOPA evaluation regardless of automatic and manual texts. We pose the challenges in VCOPA in the next section.

4. VCOPA Challenges

Figure 3 illustrates the list of challenges and their future work in VCOPA, which we enumerate as follows:

(a) Visual disambiguation. VCOPA dataset contains many image triples that are visually ambiguous. For example, in Figure 3(a), the two alternative images are blurred, however *Alternative 1* is more plausible than *Alternative 2* because the former blur corresponds to 'smog' while the latter corresponds to 'fog'. Due to the unambiguous nature of the ImageNet dataset (Deng et al., 2009), current object recognition systems do not consider visual ambiguity. In (Gella et al., 2016), they used multimodal embeddings by leveraging captions to disambiguate the visual senses. However, this would require very informative captions in order to be effective.

(b) Temporal disambiguation. The dataset also includes ambiguity on the chronological sense. For example, in Figure 3(b), both alternatives are strongly correlated to the premise. *Alternative 1* seems to happen before the premise, while *Alternative 2* happens after the premise. However, the question is asking for the effect, which means the latter alternative is the correct answer. This entails that simple correlation is not enough to find the solution. One possible solution is to use visual storytelling machines (Ferraro et al.,

2016) which describes images in sequence in order to determine if the current sequence makes sense.

(c) Fine-grained object recognition. Causal reasoning requires object recognition in the fine-grained level. For example, in Figure 3(c), an object recognition system should be able to recognize that the car in the premise image is a police car and the yellow tape in the first alternative is a police line in order to do reasoning. There are few systems which are able to detect distinct features of fine-grained objects by picking deeper filters (Zhang et al., 2016) and by localizing parts of the image (Wei et al., 2016).

(d) Event recognition. A good system should also be able to recognize events based on the identified objects. For example, in Figure 3(d), all images contain two objects (*e.g.*, a ball and a player), however all three images correspond to different events (*e.g.*, the premise image is a scene of a player kicking a ball). Recent works on event grounding (Cho et al., 2016) might help in recognizing these events.

(e) Inter-event relationship. Relationship between the events found on the pairs of images should also be considered. For example, in Figure 3(e), there is a relationship between the 'cleaning' event in the premise and the 'broken glass' event in the first alternative, even though they do not share the same objects.

(f) Event-sentiment relationship. Knowing the sentiment shown in the image can also help in causal reasoning. For example, in Figure 3(f), since the sentiment of the premise leans to negative, the second alternative, which has the same sentiment, is more plausible. Image sentiment analysis (You et al., 2015) is a growing sub-field of computer vision, and we expect solutions for VCOPA to also employ techniques used here.

(g) Inter-sentiment relationship. The relationship between the sentiments of two images is also important. Moreover, detecting more fine-grained *emotions* (Abdul-Mageed and Ungar, 2017), such as 'excited' and 'relieved', can help in causal reasoning. For example, in Figure 3(g), although all the images are showing smiling faces, fine-grained emotion detection tells us that the second alternative is 'scary', and therefore is incorrect.

(h) Commonsense knowledge. Beyond scene understanding at object, event, and sentiment levels, commonsense causal reasoning inherently requires commonsense knowledge, which is a non-visual dimension, such as "people prefer white wine to red wine with seafood", which entails that *Alternative 2* is the answer in Figure 3(h).

5. Conclusion

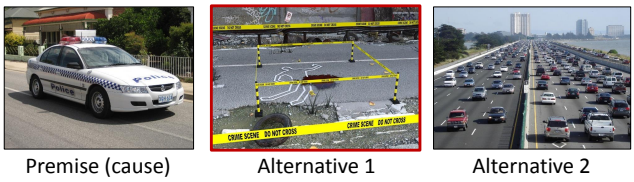
We introduce the task of visual commonsense causal reasoning with VCOPA evaluation dataset. Given a premise image and two alternatives as cause or effect, the task is to provide a more plausible answer with causality context. We provide the VCOPA dataset containing 380 questions of diverse variety on domains with over 1K images. We believe VCOPA has the distinctive advantage of pushing the frontiers on "multi-discipline" problems, while being amenable to automatic evaluation. A promising future work is to automatically harvest the VCOPA image triples to construct a large-scale image database for neural-based visual commonsense causal reasoning.



(a) Visual disambiguation



(b) Temporal disambiguation



(c) Fine-grained object recognition



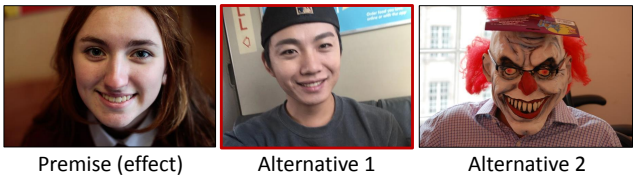
(d) Event recognition



(e) Inter-event relationship



(f) Event-sentiment relationship



(g) Inter-sentiment relationship



(h) Commonsense knowledge

Figure 3: Challenges in VCOPA

6. Acknowledgement

This work was supported by Microsoft Research, and Institute for Information communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2017-0-01778,Development of Explainable Human-level Deep Machine Learning Inference Framework). S. Hwang is a corresponding author.

7. References

Abdul-Mageed, M. and Ungar, L. (2017). Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *ACL*.

Aditya, S., Yang, Y., Baral, C., Fermuller, C., and Aloimonos, Y. (2015). Visual common-sense for scene understanding using perception, semantic parsing and reasoning. In *AAAI Symposium*.

Aditya, S. (2017). Explainable image understanding using vision and reasoning. In *AAAI*.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015). Vqa: Visual question answering. In *ICCV*.

Chang, D.-S. and Choi, K.-S. (2004). Causal relation extraction using cue phrase and lexical pair probabilities. In *ICON*.

Cho, H., Yeo, J., and Hwang, S.-W. (2016). Event grounding from multimodal social network fusion. In *ICDM*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *CVPR*.

Ferraro, F., Mostafazadeh, N., Misra, I., Agrawal, A., Devlin, J., Girshick, R., He, X., Kohli, P., Batra, D., Zitnick, C. L., et al. (2016). Visual storytelling. *arXiv preprint*.

Gella, S., Lapata, M., and Keller, F. (2016). Unsupervised visual sense disambiguation for verbs using multimodal embeddings. *arXiv preprint*.

Jiang, L., Kalantidis, Y., Cao, L., Farfade, S., Tang, J., and Hauptmann, A. G. (2017). Delving deep into personal photo and video search. In *WSDM*.

Karpathy, A., Joulin, A., and Li, F. F. F. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*.

Luo, Z., Sha, Y., Zhu, K. Q., Hwang, S.-w., and Wang, Z. (2016). Commonsense causal reasoning between short texts. In *KR*.

Roemmele, M., Bejan, C. A., and Gordon, A. S. (2011). Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Symposium*.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2017). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE TPAMI*.

Wei, X.-S., Xie, C.-W., and Wu, J. (2016). Mask-cnn: Localizing parts and selecting descriptors for fine-grained image recognition. *arXiv preprint*.

Ye, Y., Zhao, Z., Li, Y., Chen, L., Xiao, J., and Zhuang, Y. (2017). Video question answering via attribute-augmented attention network learning. *arXiv preprint*.

You, Q., Luo, J., Jin, H., and Yang, J. (2015). Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *AAAI*.

Zhang, X., Xiong, H., Zhou, W., Lin, W., and Tian, Q. (2016). Picking deep filter responses for fine-grained image recognition. In *CVPR*.

Zhao, Z., Yang, Q., Cai, D., He, X., and Zhuang, Y. (2017). Video question answering via hierarchical spatio-temporal attention networks. In *IJCAI*.

Is it worth it?
Budget-related evaluation metrics for model selection

Filip Klubička, Giancarlo D. Salton, John D. Kelleher
School of Computing
Dublin Institute of Technology
{filip.klubicka, giancarlo.salton}@mydit.ie
john.d.kelleher@dit.ie

Abstract

Projects that set out to create a linguistic resource often do so by using a machine learning model that pre-annotates or filters the content that goes through to a human annotator, before going into the final version of the resource. However, available budgets are often limited, and the amount of data that is available exceeds the amount of annotation that can be done. Thus, in order to optimize the benefit from the invested human work, we argue that the decision on which predictive model one should employ depends not only on generalized evaluation metrics, such as accuracy and F-score, but also on the gain metric. The rationale is that, the model with the highest F-score may not necessarily have the best separation and sequencing of predicted classes, thus leading to the investment of more time and/or money on annotating false positives, yielding zero improvement of the linguistic resource. We exemplify our point with a case study, using real data from a task of building a verb-noun idiom dictionary. We show that in our scenario, given the choice of three systems with varying F-scores, the system with the highest F-score does not yield the highest profits. In other words, we show that the cost-benefit trade off can be more favorable if a system with a lower F-score is employed.

Keywords: model evaluation, gain, budget, linguistic resource creation, idiom identification, idiom dictionary, F-score

1. Introduction

Building linguistic resources, such as corpora or dictionaries, can be very labor-intensive, requiring great amounts of work-hours and expert annotation. However, as pointed out by Ringger et al. (2008), fixed budgets constrain the amount of annotation that can go into the construction of linguistic resources. In many cases, the amount of available data far exceeds the time and money that is available for annotation, so one can only afford to label a subset of the data. Furthermore, for some linguistic resources only a particular (and sometimes rare) subset of the available data is relevant. Indeed, when the target linguistic phenomenon is relatively rare, the amount of time wasted filtering non-relevant data can be considerable. For example, consider the scenario of creating an idiom dictionary: in this context, only idiomatic phrases should be included in the dictionary and any time or effort expended on filtering non-idiomatic phrases is, essentially, an unnecessary cost to the project.

Given that annotators are paid either by the hour or, more often, by the number of annotations they produce (measured in words, sentences, phrases etc.), researchers are strongly motivated to reduce the annotation time, and thereby a project's annotation costs, without sacrificing the quality and coverage of the resources they are creating. To this end, the annotation process is often supplemented with predictive models that to an extent streamline, speed up and ultimately cheapen the cost of human intervention. In the scenario of creating a linguistic resource where only a subset of the potential data is relevant for inclusion (for example, a dictionary of idioms) these predictive models can be used to pre-filter the examples presented to an annotator, with the goal of maximizing the percentage of positive instances (e.g. idioms) that the annotator reviews, and minimizing the number of false positives (e.g. non-idioms) that the annotator must manually filter and discard.

In most projects focused on creating linguistic resources there exists a variety of different predictive (machine learning) models that could be used to pre-filter examples before human-annotation. Consequently, the question arises regarding how to choose the best model to aid in the annotation process. Traditionally, predictive models are evaluated based on established evaluation measures that reflect the accuracy of the model, using metrics such as F-score. Such measures are designed for evaluating predicting processes that include instance scoring and thresholding. An example of this would be deciding on the cutoff score between class labels in a binary prediction task (e.g. idiomatic and non-idiomatic phrases). However, in this paper we argue that, in contexts where annotation budgets are an important consideration and, hence, we can only afford to annotate a subset of our data, the sequence of the data becomes much more important than the application of the threshold. Thus, in such a constrained setting, selecting a pre-filtering prediction model based on a measure of accuracy which evaluates the interplay of the scoring and thresholding process on an entire test set may not be the optimal approach. Instead, we propose that a more suitable metric for model evaluation in this context is *gain*, and the associated measure of *cumulative gain*.

Unlike standard evaluation metrics that evaluate a prediction model's performance on how successfully it scores the examples and applies a threshold, the *gain* and *cumulative gain* measures provide insight into the performance of a model across different subsets of a test set. Furthermore, they are much more suitable for evaluating how the instances in the test set are sequenced. Hence, these measures provide insight that is useful for a number of different annotation scenarios. In particular there are three scenarios where we argue that *gain* and *cumulative gain* can be invaluable for the decision-making process:

1. We can have a fixed budget and can only afford to annotate a set amount of the data. Here we wish to decide which pre-filtering model is the best one to use.

2. We have a flexible budget and we can afford to annotate a variable amount of the relevant data, but we still wish to spend as little money as possible with comparable results, and we wish to decide which pre-filtering model is the best one to use.

3. We have already performed some annotation and we want to know if we should stop, or if it is worthwhile to annotate more, given the likely performance of the pre-filtering model over the next segment of data.

1.1. Related work

A number of researchers have already examined questions related to the costs associated with annotation of linguistic resources. Ringger et al. (2008) have done work on estimating the cost of corpus annotation as a step towards selecting which annotation environments are most appropriate for a given project. They perform an analysis of annotation costs on the task of correcting part of speech tags in an automatically annotated corpus. Based on these findings, they present a linear model for estimating the hourly cost of annotation for annotators of various skill levels, as well as a model for two granularities of annotation (sentence at a time and word at a time), thus providing informative guidelines for choosing an optimal annotation environment.

Similarly, Balamurali et al. (2012) present an economic model to asses the benefit accruing from the increase in project cost by performing annotation. They examine the relationship between the additional investment in annotation of WordNet senses and the subsequent increase in accuracy scores on the task of sentiment analysis. Instead of evaluating the predictive models conventionally, by comparing their accuracies, they compare expected profits, which are set up in terms of costs and expected returns. They make a comparison of approaches from different economic perspectives - namely which approach yields maximum expected profit, and which approach yields this profit the earliest (meaning less money can be spent overall). Their focus is to answer the question "Is the [subsequent] improvement in accuracy significant enough to justify the [additional] cost of annotation?", or in other words, they wonder "Should the extra cost of annotation be incurred for the task at all?" (Balamurali et al., 2012, p. 3090).

Their questions are similar to ours, but we examine a different setting and propose the *gain* metric as the answer to those questions. The remainder of our paper is thus organized as follows: in Section 2. we discuss standard machine learning model evaluation methods and provide a general motivation for the use of the *gain* measure for evaluating pre-filtering models in contexts where budgetary concerns are relevant; in Section 3. we introduce the *gain* measure; in Section 4. we illustrate the benefits of *gain* in the context of an annotation project using a case study (based on actual research data) of building an idiom dictionary; and finally we round up the paper with a conclusion.

2. Drawbacks with Traditional Model Evaluation Metrics for Annotation Pre-filtering Models

Within a machine learning context, the standard method for evaluating a predictive model is to first split a dataset into a *training set* and a *test set*. The model is induced by applying a machine learning algorithm to the training data. Once the model has been created it is then run on the test set and a measure of the performance of the model is calculated as a function of how often the predictions made by the model for the instances in the test set match the gold-standard labels for these instances.

There are a variety of different metrics that can be used to calculate the accuracy of a model on a test set. The simplest is metric is simply the raw accuracy of the model, calculated by dividing the count of test instances the model got correct by the total number of instances in the test set. Other measures of accuracy are designed to handle specific requirements or characteristics of a domain. For example, if a prediction model is being trained to discriminate between two outcomes (e.g. spam vs. ham email, healthy vs. unhealthy patients, idiomatic vs. non-idiomatic phrases) it may be that there is a particular outcome that we are interested in identifying instances of, either because of the cost of getting an instance of this class wrong or because instances of this class are rare. For example, in the health domain it is more important to identify patients who are suffering from a disease than to identify patients who do not have a disease[1]. In linguistics, it may be that we are interested in identifying instances of a rare linguistic phenomenon. In these contexts the outcome of particular interest is known as the *positive class* and there are a number of evaluation metrics designed to emphasize the ability of a model to correctly identify instances of this *positive class*. The *F-score* is an example of this type of evaluation metric[2]

The literature on the task of automatic type identification of idioms, more specifically verb and noun idiomatic combinations (VNIC), illustrates the use of these standard model evaluation metrics. Most of the work in this field either uses accuracy (used by Fazly et al. (2009)) or F-score (used by Muzny and Zettlemoyer (2013), Senaldi et al. (2016), Salton et al. (2017)) to compare model performance. These measures provide an appreciable sense of the reliability of a given model, which is why they are commonly used as evaluation metrics. However, because they focus on evaluation accuracy they are evaluating both the model's ability to score an instance appropriately and the threshold the model uses. We argue that for evaluating a model that will be deployed for pre-filtering in an annotation project this focus is not appropriate. In these contexts the best model is the model that can sequence instances correctly with a clear separation between positive instances (at the start of

[1] An error in predicting that a healthy patient is ill is likely to be corrected through follow up tests, whereas an error made in predicting an ill patient is healthy, resulting in the patient being discharged without further tests or treatment, can have disastrous consequences

[2] See (Kelleher et al., 2015) for an explanation of a range of evaluation metrics (including the F-score).

the list) and negative instances (at the end of the list). In situations where sequencing and class separation are important more important than thresholding, measures such as *gain* and *cumulative gain* become more useful to evaluate model suitability. In the following subsections we illustrate this distinction between classification accuracy and instance ranking using worked examples.

2.1. Worked example: Classification

We introduce our worked example in Table 1. Let us assume that we have a gold standard test set consisting of 6 phrases (presented in the first column), 3 of which are labeled as idiomatic (shaded green) and 3 of which are labeled as non-idiomatic (shaded red). We have trained two pre-filtering models (M1 and M2) that output a list of candidates ordered according to some sort of measure of confidence in the positive label being correct (be it a probability, or any other type of scaled measure). When using a traditional evaluation metric (accuracy, F-score, etc.) these confidence scores are converted to a class label by applying a threshold to the confidence scores. This cutoff point on our ordered list serves as a delimiter above which we consider all candidates to be labeled as positive (idiomatic), and below which we consider all candidates to be labeled negative (non-idiomatic) by our classifiers. Generally, different models will score instances differentally and will also have different thresholds. Consequently they will return different proportions of positive and negative predictions for a given test set. However, for the purposes of this discussion (and without loss of generality) we will assume that the interactions between the instance scoring and thresholding in each model is such that both models predict that two thirds of the test set are positive examples. Thus ∼60% of our data is labeled as positive by our model (shaded green), while 40% is labeled as negative (shaded red).

TestSet	S1		S2	
ID	**M1**	**M2**	**M1**	**M2**
1	1	1	3	5
2	5	2	6	4
3	4	6	4	3
4	2	5	5	1
5	6	3	2	6
6	3	4	1	2
Acc	0.5	0.5	0.17	0.5

Table 1: Two hypothetical prediction scenarios (S1 and S2) that illustrate the classification and evaluation process via thresholding an ordered list. Cells shaded green represent instances classified as positive, while cells shaded red represent instances instances classified as negative. Cells shaded in lighter colors represent false positives and false negatives, while the darker shaded cells represent true positives and true negatives.

Evaluating this output by comparing to the test set reveals the model's classification accuracy; given that accuracy is a statistic interested in evaluating prediction of both the positive and negative class, when performing the evaluation we divide the correct predictions by the total number of examples in the test set, rather than only the ones in the higher ranked group.

With this in mind, we present two scenarios: in the first scenario (S1), our first model (M1) and our second model (M2) end up with the same accuracy, meaning their performance is comparable - they both correctly predict the class of 3 out of 6 instances (the correctly labeled instances are shaded darker than the incorrectly labeled instances). Given that they have the same accuracy score, choosing a better model is a non issue. However, in the second scenario (S2), M1 has a lower accuracy than M2, meaning that M1 performs worse than M2. In this scenario, the obvious choice is the model with the higher accuracy - M2.

2.2. Worked example: Sequencing

However, when the ordered list returned by a pre-filtering model is used to order the candidates presented to a human annotator and, furthermore, the human annotator will not annotate the entire set of positive predictions (due to budget constraints), then the distribution of correct positive candidates within this ordered list, i.e. the sequence in which they are given to the annotator, becomes very important.

In other words, although two models that return the same number of correct predictions will be judged as identical using classification accuracy metrics, these models may still differ in terms of the distributions of true positive and true negative (i.e. prediction errors) instances: one model may group the true positive instances together near the top of the list and group the true negative instances near the bottom of the list, whereas the other model may intersperse the true negative instances with the true positive instances. For a model used to pre-filter data used in annotation, the model that bunches the true positive instances above the true negative instances is much more useful than the model that intersperses true positives and true negatives because the annotation process will likely be focused on the top portion of the list. This shift in focus is illustrated in Table 2, using the same example as before.

TestSet	S1		S2	
ID	**M1**	**M2**	**M1**	**M2**
1	1	1	3	5
2	5	2	6	4
3	4	6	4	3
4	2	5	5	1
5	6	3	2	6
6	3	4	1	2

Table 2: Two hypothetical prediction scenarios (S1 and S2) that illustrate the the difference of considering sequencing of true positive/true negative over classification accuracy. Instances of the positive class are shaded green, whereas the instances of the negative class are shaded red.

As we are not interested in predicting the negative class, we focus our interest only on the instances of the positive class (shaded green). In an ideal scenario, we would employ an annotator to go through the ordered list and pick out the true positive instances so we can add them to our dictionary.

If we can go over the whole list, the ordering is still not very important, as we will eventually get to all the positive instances.

However, as is often the case when building a linguistic resource, budgetary constraints mean that we cannot manually review all of the examples that a model returns as positive. Simulating the effect of these constraints on the test set requires a cut off point in the data. Thus, once budget comes into play, our list of candidates is shortened: if we can only afford to annotate ∼30% of our positively labeled data, six candidates drop out of each list, and we drop two thirds of the dataset (shaded light green and light red). This is illustrated in Table 3.

TestSet	S1		S2	
ID	**M1**	**M2**	**M1**	**M2**
1	1	1	3	5
2	5	2	6	4
3	4	6	4	3
4	2	5	5	1
5	6	3	2	6
6	3	4	1	2
TP	1	2	1	0

Table 3: Two hypothetical prediction scenarios (S1 and S2) that illustrate the advantage provided by considering the distribution of classes in the ordered list. Lighter-shaded cells represent data that cannot be annotated due to budget constraints. The bottom row shows total number of true positive (TP) instances included in the final resource.

At this point, if our goal is to get the largest amount of true positive examples for our resource, the choice between M1 and M2 becomes more complex. In S1, the most profitable choice becomes M2 (even though it has the same accuracy as M1 when judged traditionally), as it will yield 2 entries for our dictionary, whereas M1 would yield only 1 entry for the dictionary.[3] In S2, on the other hand, the most profitable choice is M1, as it would provide 1 entry for our dictionary, whereas M2 would provide 0 entries, even though its overall accuracy was almost thrice as high as that of M1.

This is only a provisional example, but it is important to keep in mind that these proportions do scale. If the model remains unchanged, the accuracy and distribution tendencies stay the same, no matter the size of the dataset. Because we know that for a given project budget we can have a human annotator review a limited number of instances, we are interested not so much in which model is the best at identifying positive instances across a full test-set, but rather which model will sequence its output in such a way as to return the largest number of true positive predictions within the top n instances ranked by the model as positive. Thus, accuracy is not an ideal measure in this situation, and relying exclusively on it can have very expensive consequences. In contrast, a measure that is very closely related to the size of the budget is *gain*.

[3]Note we are defining the concept of *profit* here as the ratio of resources invested (cost) and positive instances gained (benefit).

3. The gain measure

In Section 2. the concept of a *positive class* was introduced as a category of instances in which we are particularly interested. In contexts where such a positive class exists it is appropriate to use a model evaluation metric that emphasizes the performance of the model on the correctly identified instances belonging to this category. The F-score is an example of an evaluation metric which focuses on evaluating model performance on a positive class. The drawback with using the F-score to evaluate pre-filtering models in an annotation context is that it does not provide any insight into the distribution of true positive and true negative instances within the set of instances predicted to be positive by a model. *Gain*, and the associated concept of *cumulative gain*, also focus on a positive class but have the advantage of taking into account the separation of positive and negative instances:

> "The basic assumption behind gain is that if we were to rank the instances in a test set in descending order of the prediction scores assigned to them by a well-performing model, we would expect the majority of the positive instances to be toward the top of this ranking. The gain metric attempts to measure to what extent a set of predictions made by a model meets this assumption" (Kelleher et al., 2015, p. 433)

An important difference between traditional evaluation metrics and *gain* is that whereas traditional evaluation metrics return a single score for a model on a test set, using *gain* results in a number of *gain* scores for each model for a test set. In order to calculate the *gain* for a model, we first rank the instances in a test set according to the prediction score returned by the model for each instance. We then divide this ordered list into deciles (groups containing 10% of the dataset) and calculate the model's *gain* in each decile by dividing the true positive instances in each decile (based on the known target labels in the dataset) by the total number of positive instances in the test set:

$$gain(dec) = \frac{num\ positive\ test\ instances\ in\ decile\ dec}{num\ total\ positive\ test\ instances} \quad (1)$$

Hence, once we know what the *gain* is for each decile of our ordered list, with our budget dictating how many deciles we can annotate, we can easily identify at which point checking more candidates stops being profitable, or how much data we need to pass to our annotator to obtain the desired number of entries for our dictionary. Comparing these measures across several systems then proves to be a suitable evaluation measure. We can go back to Table 3 and apply the *gain* calculation to our hypothetical data. We are assuming that each row/data point is also one decile. Hence, we know that in S1 the first decile (D1) of both M1 and M2 has a *gain* of 0.5. However, the *gain* of M1 in D2 is 0, while the *gain* of M2 is, again, 0.5. Given that we can afford to only annotate 2 deciles, this makes M2 the clear winner.

3.1. Cumulative gain

A concept related to *gain* is *cumulative gain*, which can be defined as the proportion of positive instances in a data set that have already been identified up to a given decile. This means that as we go through the deciles, the value of *cumulative gain* rises until the last decile when it reaches its maximum. Plotting the *cumulative gain* values for each decile produces a *cumulative gain chart* (shown in Section 4., Figure 1), which allows us to understand how many of the positive instances in a complete test set we can expect to have identified at each decile of the dataset.

So in our example from Table 3, in the first decile D1 of S1-M1 the *cumulative gain* is equal to that same decile's *gain* - 0.5. But once we reach D4 (which also has a *gain* of 0.5), its *cumulative gain* will be 1, as by that point we will have encountered all the positive instances in the list.

This illustrates why we are particularly interested in *cumulative gain*, as it allows us to make the following consideration: for a given budget X we can annotate N deciles of our data. Knowing that model M has the highest *cumulative gain* at that point then tells us that we should use model M. Moving on from our hypothetical example, this reasoning can be applied to a number of real natural language processing tasks. We illustrate the application of the *gain* metric in the following section where we showcase its usefulness on a real dataset.

4. Case Study - Idiom Dictionary

As mentioned in Section 2., we examine the usefulness of *gain* on the task of building a dictionary of idioms. In our setting, we have several different predictive models at our disposal that perform type identification of potential idiomatic combinations of verbs and nouns. More specifically, our case study is based on the task of identifying potential verb and noun idiomatic combinations (VNIC). The output of each of our models provides an ordered list of potential candidate phrases for the dictionary, ranked according to a confidence measure. Hence, we do not actually classify the instances; we simply order all the instances presented to the prefiltering model and then pass them in sequence to an expert who checks whether the candidates are idiomatic or not.

We compare the performance of three models that perform VNIC type identification based on fixedness measures (Fazly et al., 2009). The particularities of the three models are irrelevant for this analysis, but it is important to note that the data featured here is actual experimental data stemming from real models.[4] Their global performance evaluations (precision, recall and F-measure) are presented in Table 4.

The table shows the models' evaluation results over a balanced test set. The metrics were calculated as the weighted average of the Precision, Recall and F-Scores on each class (VNICs and non-VNICs). All of the results are statistically significant according to Spearman's ranked correlation test (all $p < 0.05$).

[4]For more details on the background of the models, refer to Salton et al. (2017). The resulting dataset is published on GitHub as a freely available resource. It can be obtained through this link: `https://github.com/giancds/vnics_dataset`.

Model	Pr.	Rec.	F1
M1	0.82	0.73	0.70
M2	0.83	0.75	0.74
M3	0.83	0.78	0.77

Table 4: Results in terms of precision, recall and F-score, ordered by their F-scores, in the classification task over a balanced test set. The scores were calculated as the weighted average of Precision, Recall and F-Scores over each class (VNICs and non-VNICs).

As the table shows, there are variations between the F-scores: the first model (M1) has the lowest F-score, while the third model (M3) has the highest F-score, traditionally making M3 the clear choice. However, our annotation budget is constrained, so the question we ask is: which of the machines will provide the maximum benefit in terms of the number of correctly identified VNICs, given our budget? To answer this, we calculate *gain* and *cumulative gain*, as described in Subsection 3. Results of the calculations are presented in Figure 1

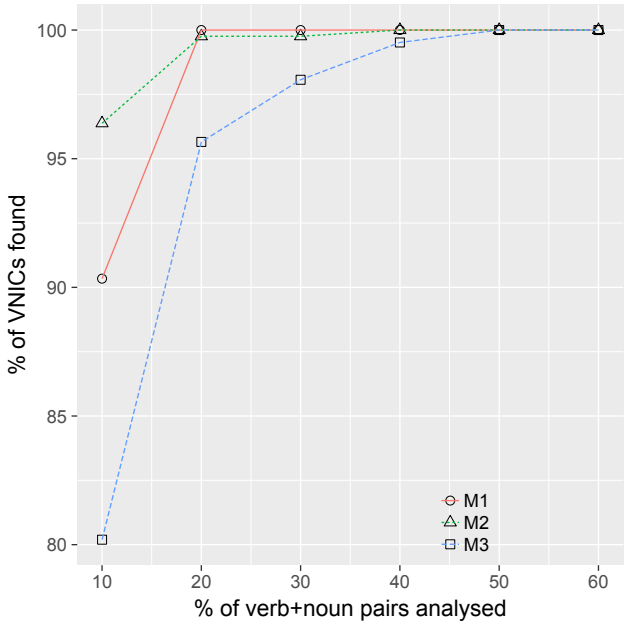


Figure 1: Cumulative gain chart for our three models' performance in the retrieval task.

The graphic relates the percentage of verb and noun pairs that were analyzed with the number of VNICs that were correctly identified by the models, essentially showing the *gain* calculations through each of the model's deciles. There are several perspectives from which we can make interpretations of this graph, depending on the variables of budget (i.e. number of deciles we can afford to annotate, which can be fixed or flexible) and the number of positive instances (i.e. VNICs, which is either 'as many as possible', or an actually targeted number, which is externally dictated). Consequently, we can find ourselves in one of the following situations:

(Situation 1) We have a fixed budget and can only afford to annotate a set amount of deciles, in hopes of getting as many VNICs as possible. We are looking for the model that can yield the maximum number of VNICs in the same amount of deciles. The choice of model for this situation then depends on the number of deciles we can annotate, and can even change depending on that number. For example, if we can afford to annotate only one decile, then Figure 1 tells us that M2 (painted green) is the preferred choice, as it will yield the most VNICs. However, if we can afford to annotate two deciles (or three, or four), then M1 (painted red) would give us more VNICs than M2, making it the preferred model. Finally, even though it has the highest F-score of the three models, M3 (painted blue) can really only be considered as an option if we can annotate five or more deciles.

(Situation 2) We have a flexible budget and we can afford to annotate a variable amount of deciles, but we still want to spend as little money as possible, yet find the maximum number of VNICs. In other words, we are looking for the model that can yield the maximum number of VNICs in the earliest decile. In our case study, that would be M1, which reaches the maximum positive instances in the second decile. In comparison, M2 reaches the maximum in the fourth decile, while M3 reaches it only in the fifth decile. This means that, if we were to choose M3, we would have to annotate 50% of the data to obtain all the candidates for our dictionary, while we would have to annotate only 20% of the data if we were to choose M1. It is interesting to note that ranking models with this in mind would show that M1>M2>M3, which is inverse to the models' ranking according to their F-scores. Thus, choosing a model based solely on the fact that it has the highest F-score would result in an unnecessary investment of time and money.

(Situation 3) We have performed some annotation and we have leftover funding - we wonder whether it is worth annotating some more of our data, e.g. another decile. Figure 1 can also help us in deciding whether it is worth spending more money on the human annotation at any given point. Say we have chosen M2 as our model and have already annotated three deciles of the data. We can consult the *cumulative gain chart* to see how many more VNICs we can expect if we decide to invest in annotating an extra decile of the data. For M1, we know we would get nothing back after the second decile, as we would already have all the available VNICs, so investing more money there would be pointless. However, if we were working with M3, it is obvious that annotating one further decile is very profitable at every step up until the fourth decile, where the *gain* starts to decline heavily. Such considerations could allow us to avoid unnecessary annotation and save money.

4.1. Cost model

There is an annotation cost associated with marking the candidate phrase as idiomatic or not. This cost is assumed to be fixed. The benefit is binary - if the human annotator deems a candidate to be idiomatic, we get a new entry in our dictionary; if it is not deemed idiomatic, there is no benefit, as the annotator was paid for considering the example, but this did not result in expanding the dictionary.

Similar to (Balamurali et al., 2012), we can apply some values to these variables and perform calculations of cost, in order to get a more palpable sense of the benefits of such considerations. To do this, we need to fix an annotation cost associated with annotating a VNIC candidate. Given that annotation costs vary for different tasks in different parts of the world, for the purpose of this illustration we fix the annotation cost at \$0.04 per candidate phrase. This value was chosen as it is the cost of a similar task (annotation of personal nouns) available on Amazon Mechanical Turk[5].

Using these values, we can apply the gain measure to perform cost calculations on our test set. We keep in mind that the gain scores scale to different sizes of datasets: if the model had a gain of x on the third decile of the test set, we can extended our annotation to cover the third decile of the project data as ordered by the model, we would be likely to have a similar gain on our larger dataset and ultimate linguistic resource. In other words, just like with gain, we can extrapolate from the cost calculations on our test set to estimate the cost of annotating a larger-scale unlabeled dataset.

We thus turn to our case study, in which our test set is our ordered list of 2,091 verb+noun pairs, of which 414 are VNICs. Using the aforementioned annotation costs, we know that the cost of a single annotator going through 1 decile (10% of data) would be \$8.3, and annotating the whole list would cost us \$83.64 (100% of data). Once we have these values ready, our cost calculations depend on what our goals are, as derived from the different situations described in the previous section.

If our goal is the one from Situation 1 – to get as many VNICs as possible on a fixed budget – we need to consider the budget and how many VNICs it can give us for each system. For illustrative purposes, we assume that our annotation budget is fixed at \$16. This means we can afford to annotate not more than two deciles. If we spend that money on M1 we will actually get all 414 VNICs. If we go with M2, we will get slightly less - 410 VNICs. If we opt for M3, however, we will get back 394 VNICs. This means that the difference between spending \$16 on M1 and M3 is a difference of 20 VNICs, or 5% fewer positive instances.

If our goal is to maximize the benefit while spending as little money as possible on a flexible budget (as in Situation 2), this means we want a model that will find all the true positives as early as possible. To annotate all the VNICS we would have to spend a minimum of \$16.56 in the ideal scenario where they are all found at the top of the ordered list. However, this is not the case with any of our three models. The costs of annotating all available VNICs in relation to each model are presented in Table 5. As we can see, comparing the cheapest model M1 with the most expensive model M3 yields a difference of 60%.

These values may seem small, but, as already stated in Section 2., these are calculations performed on a test set, and the proportions do scale to larger samples. As a consequence, in larger projects with vast amounts of data and numerous annotators, saving 60% of the annotation budget can turn out to be quite a considerable amount.

Finally, if we have already done some annotation, but our

[5] https://www.mturk.com

Model	Cost
M1	$16.73
M2	$33.46
M3	$41.82

Table 5: Cost of annotating the maximum amount of true positives for each model on the test set from our case study.

goal is to see if it is worth annotating some more data (as in Situation 3), we want to see how many more VNICs we will get if we annotate another decile. Let us assume that we have annotated only the first two deciles. This means we have already spent $16.6. If we look at M1 and M2, we see that spending another $8.3 (50% of the already spent budget) will actually give us 0 new VNICs, so we know that spending more resources on annotation would be fruitless. However, looking at M3's third decile, we see that, if we spent that additional $8.3, we would in fact get an additional 8 VNICs, which is a much more valuable cost-benefit trade-off than the previous two.

In addition, this final line of thinking does not exclusively apply to model comparisons. Even in a scenario where we only have a single predictive model at our disposal, rather than comparing different models' performances, we can still perform *gain* calculations to know when to stop annotating the output of just the one model. If M3 is our only available model, we may perform these calculations after each decile, and depending on how frugal we want to be, we could decide that the additional 4 VNICS obtained by annotation the 4th decile are not worth the additional investment of $8.3, thus saving the funding for other potential annotation tasks.

5. Conclusion

In this paper we emphasize budgetary concerns in the task of creating a linguistic resource that includes predictive modeling and subsequent human annotation of the output data. We demonstrate different ways of how using the measure of *gain* can help when faced with a choice between similarly performing models on a limited budget.

We show that examining the *gain* of a predictive model can help us answer several important budgetary questions. It helps us identify which model will yield the highest profit at the earliest time, or rather, which model will yield the highest number of positive annotations, given our budget. It can also inform us on how much we need to invest to achieve a desired number of entries in our dictionary, as well as tell us at which point annotating more candidates stops being profitable.

We demonstrate all of this on real experimental data, highlighting that sometimes a model that has considerably higher traditional evaluation scores can actually perform much worse in any of the above regards, ending up as more costly than a model with a lower evaluation score.

For the purpose of this paper we have illustrated our point on the example of building a dictionary of idioms as a linguistic resource, but we should note that the same considerations can be applied to the task of creating any other linguistic resource, or indeed many other NLP tasks, as long as the pipeline integrates a probabilistic model.

For example, in SemEval-2018[6], one of the shared tasks was hypernym discovery[7]. The goal was to build a system that would take a word or phrase as its input (e.g. folk rock) and then generate a list of the 10 most likely candidate hypernyms for that word (e.g. rock, genre, music, ...). Given that a word can have more than one correct hypernym, the goal of the task was not only to build a system with high global accuracy and F-scores, but also to build a system that can cleanly separate the positive and negative classes. Because the output of the task is constrained to the top 10 candidates, and those candidates are submitted as potential hypernyms, we would want the system to assign the highest probabilities to the true positives so that, ideally, all or most of the top 10 candidates are indeed hypernyms. Thus, if participants of the task were to build several systems, calculating *gain* will show which system would likely yield the highest evaluation scores, which could help inform their decision on which system to submit for evaluation.

Certainly, we are not claiming that performing *gain* calculations will invariably undermine the results of global F-score evaluations. We only wish to raise awareness of the fact that when it comes to model selection, external constraints (such as budget) can drastically change the perspective on the model's performance, which warrants more specialized consideration. Indeed, the usual evaluation metrics such as accuracy and F-score are a perfectly appropriate tool for an intrinsic comparison of models, but once they are taken out of the lab, the evaluation should be extrinsic, taking into account the whole pipeline of the task the models are applied to. This shift in context warrants a reconsideration of the appropriate evaluation metric, and *gain* might just be the answer.

Acknowledgements

This research was supported by the ADAPT Centre for Digital Content Technology which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund. Giancarlo D. Salton would like to thank Capes ("Coordenação de Aperfeiçoamento de Pessoal de Nível Superior") for his Science Without Borders scholarship, proc n. 9050-13-2.

Bibliographic References

Balamurali, A. R., Joshi, A., and Bhattacharyya, P. (2012). Cost and benefit of using wordnet senses for sentiment analysis. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3090–3097, Istanbul, Turkey, may.

Fazly, A., Cook, P., and Stevenson, S. (2009). Unsupervised type and token identification of idiomatic expressions. In *Computational Linguistics*, volume 35, pages 61–103.

[6] http://alt.qcri.org/semeval2018/

[7] https://competitions.codalab.org/competitions/17119

Kelleher, J. D., Namee, B. M., and D'Arcy, A. (2015). *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Word Examples and Case Studies*. MIT Press.

Muzny, G. and Zettlemoyer, L. S. (2013). Automatic idiom identification in wiktionary. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1417–1421.

Ringger, E., Carmen, M., Haertel, R., Seppi, K., Lonsdale, D., McClanahan, P., Carroll, J., and Ellison, N. (2008). Assessing the costs of machine-assisted corpus annotation through a user study. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 3318–3324, Marrakech, Morocco.

Salton, G. D., Ross, R. J., and Kelleher, J. D. (2017). Idiom type identification with smoothed lexical features and a maximum margin classifier. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pages 642–651, Varn, Bulgaria.

Senaldi, M. S. G., Lebani, G. E., and Lenci, A. (2016). Lexical variability and compositionality: Investigating idiomaticity with distributional semantic models. In *Proceedings of the 12th Workshop on Multiword Expressions, MWE@ACL 2016, Berlin, Germany, August 11, 2016.*, pages 21–31.

Automated Evaluation of Out-of-Context Errors

Patrick Huber, Jan Niehues, Alex Waibel
Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT)
uhejt@student.kit.edu, jan.niehues@kit.edu, alex.waibel@kit.edu

Abstract

We present a new approach to evaluate computational models for the task of text understanding by the means of out-of-context error detection. Through the novel design of our automated modification process, existing large-scale data sources can be adopted for a vast number of text understanding tasks. The data is thereby altered on a semantic level, allowing models to be tested against a challenging set of modified text passages that require to comprise a broader narrative discourse.

Our newly introduced task targets actual real-world problems of transcription and translation systems by inserting authentic out-of-context errors. The automated modification process is applied to the 2016 TEDTalk corpus. Entirely automating the process allows the adoption of complete datasets at low cost, facilitating supervised learning procedures and deeper networks to be trained and tested. To evaluate the quality of the modification algorithm a language model and a supervised binary classification model are trained and tested on the altered dataset. A human baseline evaluation is examined to compare the results with human performance. The outcome of the evaluation task indicates the difficulty to detect semantic errors for machine-learning algorithms and humans, showing that the errors cannot be identified when limited to a single sentence.

Keywords: Out-of-Context Error Recognition, Automatic Evaluation Dataset, Text Understanding, TEDTalk

1. Introduction

Machine learning strives to achieve universal application in solving arbitrary tasks rather than special functionalities and well-defined functions. This trend is reflected by the increasing importance of text understanding within the field of natural language processing (NLP).

Extracting the context of a narrative passage can thereby benefit a wide variety of applications, such as automatic speech recognition (ASR) and neural machine translation (NMT) tasks. Most state-of-the-art ASR and NMT systems are currently only processing one sentence at a time, solely basing the system's decisions on the local context (Bahdanau et al., 2014; Cho et al., 2014). This approach is reasonable, as it addresses the most general case. Nevertheless, considering a larger context can highly benefit the overall system's performance for text passages with contextual features.

Imagine an ASR system that is supposed to transcribe the sentence *There are great opportunities in the far-east* within a presentation on the global economy. Through an inarticulate pronunciation and a noisy signal at the conference, the ASR system transcribes the sentence as *There are great opportunities in the forest*, as the sentences are phonetically very similar. In this example, the language model of the ASR system will not be able to identify the out-of-context error by only considering the current sentence.

Therefore, developing a context component to enhance the text understanding is an important task to improve state-of-the-art systems.

While the number of published datasets regarding question-answering (QA) tasks steadily increases, for example based on children books (Hill et al., 2015) or simple reasoning (Weston et al., 2015), there are very limited resources covering the area of context-aware error detection based on a narrative.

For this reason, we designed a novel text understanding task, which is difficult to solve by only relying on the local context of a sentence, but becomes feasible when taking a broader discourse into account. We introduce a fully automated dataset extension procedure, designed to assess the performance of computational models to identify out-of-context errors. Through the fully automated modification of the data, the out-of-context errors vary within a complexity range. The diversity of the modification severity generates a broad scale to compare computational approaches.

To evaluate the performance of the modification procedure, we apply the process on the 2016 TEDTalk corpus (Cettolo et al., 2012) and test multiple baseline systems against the dataset.

Therefore, a standard sentence-based language model (LM) is evaluated on the corpus, reaching a F-score of 6.51%. To show the performance of supervised models, a binary classification network is set up and trained on the modified corpus achieving a F-score of 10.16% on the test set. To gain additional insight on the relative performance of the computational baseline models, we conduced a human baseline survey, showing the difficulty of the task for human subjects (see section 6. for more details).

2. Related Work

The LAMBADA dataset introduced by Paperno et al. (2016) is one of the first and most comprehensive data sources addressing the task of text understanding by the means of word prediction. Through a multi-staged filtering approach and extensive evaluation of each text passage, the corpus represents a high quality data source for context-aware word prediction tasks.

Nevertheless, generating these handcrafted data sources is very expensive and thus limited. For example, to be able to train supervised models, not only does the relatively small development and test sets need to be processed, but also the large set of training data.

Our newly developed alteration approach addresses this downside through the application of a fully automated modification procedure.
Even though the LAMBADA dataset and our newly introduced modification process are targeting the same overall task, the performed inference on the data varies. The LAMBADA inference task targets the continuation of the current text passage at a known position in the text by predicting the last word within the paragraph. In contrast to that, our newly introduced substitution process replaces context-relevant words within the text passage at arbitrary positions, increasing the complexity of the task.
Sennrich (2016) introduces a dataset with automatically inserted errors focusing on advanced computational models for NMT tasks.
The paper by Burlot and Yvon (2017) proposes an evaluation process for NMT models, assessing the morphological properties of a system. The process substitutes nouns, as well as other part-of-speech tokens with filtered and randomly chosen replacement words.

3. Task

The task introduced in this paper is designed to evaluate the performance of computational models for out-of-context error detections.
The fully automated modification process described in section 4. provides the ground truth for the task. The artificially inserted out-of-context tokens are uniformly distributed over the dataset, elevating the complexity of the task over fixed-position approaches. Compared to approaches with well-known target positions of substituted words w_{n} (e.g. at the end of a paragraph), the out-of-context word replacements within this task are at random positions w_{p}. The series of word-tokens is thereby described by the ordered sequence of words $W = (w_1, w_2, ..., w_{\mathrm{n}})$ with $w_{\mathrm{p}} \in W$.
The task to find the correct word w_{n} at a known position (in this case at the end of the paragraph of length N) can be described as a classification task with one class per word in the vocabulary m_{vocab}. The assumption to know the position of the replaced out-of-context word w_{p} does not hold, as the substitutions are randomly distributed. Instead, every word $w_{\mathrm{p}} \in W$ needs to be assessed against every other word $w_{\mathrm{q}} \in W$ in the sequence with $p \neq q$.
The presented task can therefore be interpreted as a binary sequence labeling problem defined by the input sequence $W = (w_1, w_2, ..., w_{\mathrm{n}})$ of length N representing the text passage and the output sequence $L_{\mathrm{OOC}} = (l_1, l_2, ..., l_{\mathrm{n}})$, also of length N, with $l_{\mathrm{i}} \in L_{\mathrm{OOC}}$ representing the label of the input element $w_{\mathrm{i}} \in W$. The labels L_{OOC} thereby separate the two classes $\{0, 1\}$, representing valid-context tokens (0) and out-of-context tokens (1).
The labeled text passage:

$W \cup L_{\mathrm{OOC}}$ = (*(We, 0) (have, 0), (in, 0) (higher, 0) (education, 0), (a, 0) (trillion, 0) (dollars, 0) (of, 0) (student, 0) (debt, 0) [...]. (We, 0) (have, 0) (a, 0) (lot, 0) (of, 0)* ***(shopping, 1)****. (Our, 0) (economy, 0) (grew, 0) [...] (on, 0) (the, 0) (back, 0) (of, 0) (consumers, 0) (massively, 0) (over, 0) (borrowing, 0).*)

from the presentation *The death of innovation, the end of growth* by Robert Gordon, held in February 2013, gives an example on how the binary sequence labeling L_{OOC} can be applied.
Our task definition does not provide any information about the position of the modifications, nor give any insight about the total number of out-of-context substitutions on the data. To identify the dataset replacements with a computational model, the narrative of the text passage needs to be employed. A context C is thereby defined as a coherent text passage of up to ten sentences containing multiple appearances of the same reoccurring noun n.

4. Data Modification

The goal of the dataset manipulation is to modify an existing database by artificially inserting out-of-context errors in text passages with especially strong contextual features. The substitution procedure is fully automated to enable the modification of entire large-scale datasets without the costly validation of the data by human subjects. A crucial task emerging through the automated processing is the reasonable replacement of context related words. The substitutions fulfill the following two requirements:
First of all, the modified dataset serves as the ground truth to test the train context-aware computational models against.
Secondly, through the out-of-context modifications on the training set, supervised models can be trained on the data.
This is achieved through a four-stage computational procedure:

(1) Dataset Filtering
To enhance the dataset quality, non-contextual parts of the data are removed, excluding self-contained short text passages with less than 200 words.

(2) Part-of-Speech Tagging
For the semantic out-of-context substitutions, only certain part-of-speech (POS) classes are taken into account. Thus, every token in the dataset needs to be assigned a POS category. As described by Paperno et al. (2016), context is especially critical to nouns, whereas other POS classes can often be inferred directly out of the local context of a sentence. Our substitution process therefore focuses on the replacement of nouns.

(3) Candidate Selection
To ensure that a text passage contains a sufficient context, the nouns determined by the POS-tagger are filtered for contextual coherence. A context is thereby assumed if the same noun appears multiple times within the same text passage. The last appearance of the noun in a context of ten consecutive nouns qualifies as a suitable replacement and is saved as a potential out-of-context substitution candidate.
Out of the list of potential replacement candidates, a predetermined number of tokens is randomly selected according to a uniform distribution.

(4) Appearance Window

For every selected token in the original dataset, a syntactically suited replacement token is determined. The tokens are thereby replaced by words within a defined appearance window. The appearance window approach is based on the assumption, that words with a similar word count on the original dataset are suitable replacements. The approach has multiple characteristics that enhance the quality of substitutions: (a) A similar word count of replacements avoids common words being substituted by rare words. (b) The replaced words are typically not related to the original word, as the overall word count generally not infers semantic affiliation.

Within the appearance window further filtering regarding the tense and grammatical number are executed to determine the most suitable substitution.
This process can be applied to arbitrary data sources with contextual features to train and test context-aware models on. To show the results of the modification process and evaluate the quality of the semantic replacement tokens, we apply the modification procedure on the 2016 TEDTalk corpus[1] and assess the semantic out-of-context dataset substitutions on supervised as well as unsupervised baseline models.

5. Baselines

To evaluate the performance of the modification process (section 4.) on the newly introduced task (section 3.), three baselines are designed. The baseline neural LM evaluates the quality of the modification pipeline for unsupervised models. A standard binary classification neural network assesses the performance of supervised models. A human baseline is additionally acquired to assess the human performance on the task.

(1) Language Model
The architecture of the baseline neural LM is derived from the original model proposed by Bengio et al. (2003), which employs a fundamental architecture for the neural network. The basic recurrent neural network (RNN) architecture is extended by state-of-the-art LSTM units to enhance the model's performance on the task, as proven more effective by Xie and Rastogi (2017) and Sundermeyer et al. (2012). The shallow design of the standard baseline LM to evaluate the quality of the modification process contains three computational layers.
The embedding layer is trained on the complete input of the neural network and encodes the sparse word representations into 256-dimensional real-valued word embedding vectors.
The 256-dimensional word embeddings are subsequently fed into the 512 LSTM units within the recurrent network layer.
To expose the outputs of the LSTM units, the output layer comprises of one computational unit per class. In order to normalize the output of the network, a softmax activation function is chosen as the computation of the output layer.
The model is trained with the Adam optimizer, a learning rate of 1e-3 and the cross entropy as the selected cost function. The size of the model is limited to a vocabulary size of 30,000 words and a maximum sequence length of 50 words.

(2) Binary Classification Model
The binary classification model is the supervised baseline model to evaluate the modification process against the defined task. In order to keep the approaches comparable, the architecture and hyper-parameters of the supervised baseline network are selected to be similar to the design of the unsupervised neural LM.
The main difference between the models is the output layer, as the binary classification model only distinguishes two classes $\{0, 1\}$. All other properties of the systems are identical.

(3) Human Baseline
To assess the human baseline performance on the newly introduced task, ten random sentences with one or more replacement tokens are selected from the modified dataset and presented to seven human participants in random order. The task description for the human baseline was to find the word(s) in the sentences that do not fit the context. With the inserted replacements being intentionally designed to be semantically rather that syntactical, the replacements are deliberately difficult to identify with only one sentence.
To be able to directly compare the human results to the two baseline neural networks, the computational models are also exclusively tested on the ten sentences (232 words) utilized for the human baseline evaluation.

6. Evaluation

The modification process (section 4.) is applied and evaluated on the English TEDTalk corpus released for the IWSLT 2016 (Cettolo et al., 2012). The original TEDTalk database contains transcripts of over 2,600 TEDTalks presented between 2007 and 2016. With a well-defined topic per TEDTalk and an average duration of 30 to 45 minutes per presentation, the TEDtalk corpus contains strong contextual features within long and coherent contexts. Furthermore, the TEDtalk dataset covers a wide variety of topics keeping the data mostly unbiased.
After applying the replacement procedure on the original TEDTalk dataset, over 25,000 contextual words are replaced by out-of-context tokens. The following example is taken from the modified TEDTalk corpus to illustrate the result of the substitution process.

Local context: *And that's the world we're coming into, in which we will increasingly see that our* ***marketers*** *are not fixed.*

Extended context: *Now notice, in a period which is dominated by a mono-polar world, you have fixed alliances – NATO, the Warsaw Pact. A fixed polarity of power means fixed alliances. But a multiple polarity of power means shifting and changing alliances. And that's the world we're*

[1] https://github.com/isl-mt/SemanticWordReplacement-LREC2018

coming into, in which we will increasingly see that our ***marketers*** *are not fixed.*

Ground truth: *marketers* → *alliances*

(Paddy Ashdown, December 2011 at TEDxBrussels)

The substituted words in the text passages shown above are difficult to identify by only relying on the local context of the sentence. However, taking the broader discourse of the extended context into account discloses which words are out of context.
As the modifications on the TEDTalk dataset are fully automated, not all replacements are equally challenging. Within the complexity range of modifications, the replacements vary from easy substitutions, which can eventually be identified by only assessing the local context, to difficult substitution, which are hard to infer for human subjects. Through the design of the processing pipeline, the number of these outliers is minimized.
Figure 1 supports this assertion by comparing the distribution of the original TEDTalk tokens with the replaced words using the computational baseline model described in section 5.. Thereby, every word in the vocabulary m_{vocab} is assessed at the replaced positions. The probabilities of the words are sorted and divided into four equal quarters representing the four classes on the x-axis. The distribution of the original TEDTalk tokens with 93.35% of the words within the top 25% of the vocabulary shows the effectiveness of the baseline LM on the original data. The distribution of the modified words, with 84.6% of the artificially replaced words within the highest quarter (75%-100%), shows that the replacements also fit well in the local context.

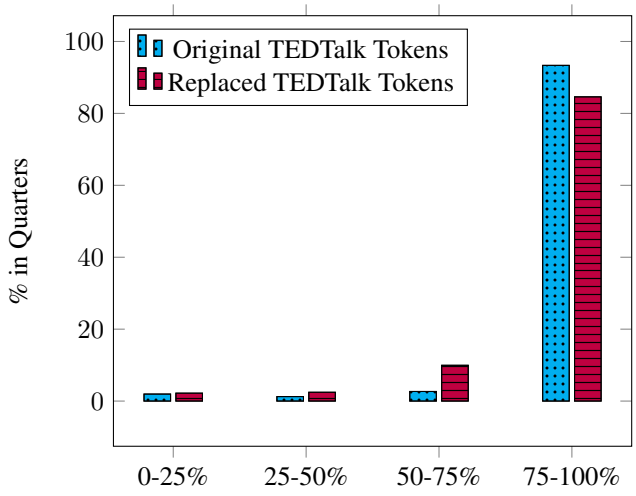


Figure 1: Substitution Quality of Modified Tokens

To further assess the quality of the replacements, we evaluate the results of the modification process by training and testing the baseline models described in section 5..
As the baseline LM is unsupervised, the unmodified TEDTalk dataset is used to train the computational baseline model. The performance is tested against the modified test set.
To compare the performance of the computational models, the F-score is assessed. As described in section 3., the absolute amount of replacements, as well as the positions of the out-of-context error tokens is not provided. Therefore, to assess the quality of the model, the probability of every word within the dataset D is calculated. The generated set of probabilities is subsequently sorted to classify the tokens with the lowest probability N_{ooc} as out-of-context tokens. With the dynamic threshold $t_{N_{\text{ooc}}} = \{1..N\}$ with $1 \leq N \leq |D|$ to separate the binary classes, the best F-score is retrieved.
For the unsupervised model, the perplexity measure is additionally applied to show the systems' performance on the general word prediction task.
The results are shown in table 1. The reported performance for the unsupervised neural LM on the test set is a perplexity value of 115 and a F-score of 6.51%.

Model	Perplex	F-Score	F-score*
Lang. Model	115	6.51%	8.65%
Bin. Class. Model	-	10.16%	13.43%
Human Survey	-	-	28.53%

Table 1: Out-of-Context Detection Rates on the Test-Set (F-score* refers to a randomly chosen subset of the test set to compare the computational and human baselines)

The second baseline model is the supervised binary classification model. This model particular benefits from the automated manipulation process, as the training and the testing require tagged data. In order to separate the two classes $\{0, 1\}$, the model learns from the labeled ground-truth data. As the positions of the replaced out-of-context tokens are not known, the separation between the classes is solely based on the score of the tokens. To divide the model's output into the two classes, it is sorted by the probability of the words and separated at the best threshold. The best result achieved by the model is a F-score of 10.16%.

As the third score, a human evaluation has been implemented[2]. This way, we can compare the computational results with the human performance.
The displayed F-score* in Figure 1 shows the direct comparison between the baseline LM, the baseline binary classification model and the results of the human survey on a randomly selected subset of the test data.
Both baseline models thereby achieve comparable results on the randomly chosen subset as to the score on the complete test set (8.65% for the LM and 13.43% for the binary classification model), indicating that the subset is representative. The human baseline evaluation shows a better result, reaching an F-score of 28.53%.
This outcome indicates that the presented task is not only

[2]https://github.com/isl-mt/SemanticWordReplacement-LREC2018

difficult to solve for computational models, but also for humans, when the context of a sentence is not available. If the user is able to see an extended context on the other hand, it is nearly always possible to find the error, as shown in some preliminary analysis.
With this result, we additionally show the integrity of the baseline models, performing on a reasonable level compared to the human performance.

7. Conclusion

This work presents a novel task to evaluate context-aware computational models. The task reproduces common errors of state-of-the-art transcription and translation systems that can be prevented by taking a broader discourse into account. The arbitrary positions of the inserted errors classify the task within the area of sequence labeling.
Our newly introduced substitution approach to modify existing datasets with out-of-context tokens represents the ground-truth for the assessment of the task. We show that the substituted tokens are difficult to infer by computational models that solely employ the local context. Thus, a context-aware approach needs to be developed to enhance the performance on the task.
The newly defined task is a comprehensive benchmark for context-aware models to train and test the ability of the system to abstract from the syntactical noise of the text passage and learn to focus on the semantic representation of the narrative. Through our automated modification pipeline, the necessary ground-truth generation is accessible and fast.
Therefore, the introduced combination of a real-world task definition and an automated processing pipeline represents a complete framework to test future models on the task of text understanding by the means of out-of-context error detection.

8. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3.

Burlot, F. and Yvon, F. (2017). Evaluating the morphological competence of machine translation systems. In *Proceedings of the Second Conference on Machine Translation*, pages 43–55.

Cettolo, M., Girardi, C., and Federico, M. (2012). Wit^3: Web inventory of transcribed and translated talks. pages 261–268, May.

Cho, K., van Merrienboer B, Bahdanau, D., and Bengio, Y., (2014). *On the properties of neural machine translation: Encoder-decoder approaches*.

Hill, F., Bordes, A., Chopra, S., and Weston, J. (2015). The goldilocks principle: Reading children's books with explicit memory representations. *CoRR*, abs/1511.02301.

Paperno, D., Kruszewski, G., Lazaridou, A., Pham, Q. N., Bernardi, R., Pezzelle, S., Baroni, M., Boleda, G., and Fernández, R. (2016). The LAMBADA dataset: Word prediction requiring a broad discourse context. *CoRR*, abs/1606.06031.

Sennrich, R. (2016). How grammatical is character-level neural machine translation? assessing MT quality with contrastive translation pairs. *CoRR*, abs/1612.04629.

Sundermeyer, M., Schlüter, R., and Ney, H. (2012). Lstm neural networks for language modeling. *Annual Conference of the International Speech Communication Association*, 13.

Weston, J., Bordes, A., Chopra, S., and Mikolov, T. (2015). Towards ai-complete question answering: A set of prerequisite toy tasks. *CoRR*, abs/1502.05698.

Xie, S. and Rastogi, R. (2017). Deep poetry: Word-level and character-level language models for shakespearean sonnet generation.

Matics Software Suite: New Tools for Evaluation and Data Exploration

Olivier Galibert, Guillaume Bernard, Agnes Delaborde, Sabrina Lecadre, Juliette Kahn
Laboratoire national de métrologie et d'essais
1 rue Gaston Boissier - 75015 Paris, France
firstname.lastname@lne.fr

Abstract

Matics is a free and open-source software suite for exploring annotated data and evaluation results. It proposes a dataframe data model allowing the intuitive exploration of data characteristics and evaluation results and provides support for graphing the values and running appropriate statistical tests. The tools already run on several Natural Language Processing tasks and standard annotation formats, and are under on-going development.

1. Introduction

The evaluation of data processing systems is a cornerstone for developers, researchers and users. The evaluation allows positioning a technology with regard to the competition, but also allows assessing the performance of the system in different contexts. Through quantified scores, it orientates the development or guides the user towards the most suitable product. As evidenced by the popularity of evaluation competitions such as the VarDial evaluation campaigns (Malmasi et al., 2016), the Interspeech challenges (Schuller et al., 2017), the CoNLL Shared Tasks (Oepen et al., 2017) or the Evalita evaluation campaign (Pierpaolo et al., 2017), there is a constant need for the rapidly-evolving Natural Language Processing (NLP) technologies to position themselves.

The evaluation of text, speech or multimedia data processing systems relies on large amounts of usually annotated data. Numerous works propose interfaces or frameworks to build, explore and visualize corpora of annotated data. ANVIL (Kipp, 2010) for instance proposes a well-conceived database-oriented annotation tool where the user can add temporally or spatially grounded elements. The annotated data can be exported to perform statistical analyses in several external systems. UAM (O'Donnell, 2008) emphasizes the project management aspects in a multi-layer text annotation task and offers dedicated statistical analysis tools. Headtalk/Handtalk (Knight et al., 2009) explores the annotation and visualization of multimodal corpora, for the purpose of building datasets suitable for statistic analysis.

Some propose a framework able to explore data in a specific context. (Schmitt et al., 2010) for instance is dialogue-oriented, and presents a multi-level interface including dialog selection from a database, display of the selected dialog, and application and evaluation of integrated prediction models for various characteristics (task completion, anger level, age and gender predictions).

All these systems are mainly dedicated to annotation tasks and/or to specific NLP applications. They offer data exploration features, and can either produce data formatted to perform an evaluation in an external system, or offer statistical analysis specialized for testing coherency or measuring advance on a specific NLP task.

The LNE (laboratoire national de métrologie et d'essais - French national metrology and testing laboratory) has conducted many evaluations of data-processing systems in projects such as Quaero (Galibert et al., 2011), ETAPE (Gravier et al., 2012), MAURDOR (Brunessaux et al., 2014), PEA-TRAD or REPERE (Giraudel et al., 2012; Galibert and Kahn, 2013). These evaluations concerned various NLP tasks and systems (speech recognition, speaker diarization, speaker identification, named entities recognition, optical character recognition, etc.), which implied dealing with different system output formats, annotation guides, and comparison metrics. A number of commonalities appeared through time in the process of such evaluations, in the pre-processing and exploration of the data and the computation and viewing of statistical scores, hence the need for a reusable and general framework to carry out the evaluations.

One aspect we are especially interested in is to be able to assess the representativity of the different sub-corpora created (train, development, test) and to identify factors of influence on the performance of the system. Such an analysis is usually done through a mix of independent evaluation tools, ad-hoc data extraction scripts and generic analysis engines (such as R), or dedicated to a specific NLP task (such as the NIST Scoring Toolkit SCTK (NIST, 2015) for speech recognition). This works perfectly fine for evaluations on specific applications, or on databases of average size; this becomes somewhat burdensome when performing large scale evaluations on a great panel of application types.

We thus decided to build a new tool to provide a unified response to our evaluation needs by first testing some data handling and UI prototype in a pre-project called LNE-Visu, presented in a demonstration at the French JEP-TAL-Recital joint conference in 2016 (Bernard et al., 2016).

Then, taking the results into account, we started an internal project to build the Matics software suite, to implement the vision we have of such an exploration interface. It integrates evaluation, exploration at varying granularity, graphical representations and statistical testing. All these aspects are presented in this paper.

2. Matics at a Glance

2.1. General Description

Matics comprises two interconnected softwares:

- DATOMATIC: It is designed for the importation and database indexation of corpora and files. The data can be made up of reference data (e. g. labeled by an expert) and hypothesis data (output of an NLP system, automatically labeled). Source data (i. e. unlabeled and/or unstructured) can also be included, such as plain text or audio. The data can be browsed through via search features, and visualized according to their types (text, video, audio and the related annotations). The software offers several descriptive statistics (signal duration, number of words/speakers/entities, file or language distribution...). Multi-criteria sub-selections on the corpora can be performed. The resulting corpora can be locally exported to be processed in Evalomatic.

- EVALOMATIC: Evalomatic works exclusively on Datomatic formatted databases. Evalomatic allows running evaluations, for example comparisons between reference and hypothesis data for speech transcription tasks. The reference and hypothesis data (as well as the evaluation results) are structured as *dataframes*, which allows performing several manipulations on the data for an evaluation at different levels of granularity. The software offers several standard comparison metrics (e. g. F-measure, Slot Error Rate SER), some of which specifically designed for NLP (e. g. Word Error Rate WER). Statistical functions are provided (e. g. t-tests or Anova). Data and results can be plotted on graphs (e. g. DET plot, bar chart).

Matics is an on-going work, initially developed to address our team's evaluation needs. The decision of publicly releasing it is motivated by our wish to contribute to a thriving development of NLP technologies, and artificially intelligent systems on the whole. In its earlier stages, the software suite presents some limitations: we do not guarantee it is fully bug-free, many features are left to add, and as of now the interface only offers French. Evaluation being our core activity, the development of Matics is one of our main priorities, and there are, and will be, constant updates.

2.2. Availability

The Matics suite is free and open-source. It can be downloaded at: https://www.lne.fr/logiciels/lne-matics.

2.3. Supported NLP Tasks

As of now, Matics allows performing evaluations on NLP systems for these tasks:

- Automatic Speech Recognition (ASR)
- Named Entity Recognition (NER)
- Tokenization
- Lemmatization
- Speaker verification

Note that Matics supports Latin and non Latin alphabet languages (Chinese, Arabic, Russian...).

2.4. Supported External Formats

Matics supports several standard structured formats, like XML (e. g. Transcriber) or the Tab Delimited Format of XTrans. It also supports annotation formats such as:

- The stm and ctm file formats (in the sclite() program developed by the NIST for the evaluation of speech recognizers);
- The CoNLL-X format (Tjong Kim Sang and De Meulder, 2003);
- MUC-7 (Chinchor and Robinson, 1997);
- QUAERO (Galibert et al., 2011).

As of now, unsupported formats need to be externally transformed into an supported format so as to be loadable in Datomatic, but supporting new formats for an already handled task requires a reasonable amount of effort.

2.5. Implemented Metrics

- ASR: WER (Word Error Rate); CER (Character Error Rate; NCE (Normalised Cross-Entropy)
- NER: SER (Slot Error Rate); ETER (Entity Tree Error Rate)
- Speaker verification: EER (Equal Error Rate); C_{det} (Cost of DETection); C_{llr} (Cost Log-Likelihood Ratio)
- General metrics: F-measure; Recall; Precision

These metrics cover the evaluation of NLP applications described hereinbefore. New metrics will be added along with the expansion of the NLP tasks list.

2.6. Statistical Functions

A toolbox of several standard statistical functions is available. The result of these functions can be used as new columns in the dataframe, meaning that they can be used as a test statistic in the evaluation.

- Descriptive statistics:
  - Gaussian statistics: mean, standard deviation, skewness, kurtosis
  - Distributional statistics: min, max, median, first and last quartile, first and last decile, mode
- Significance tests on paired experiments:
  - Gaussian : paired t-test
  - Non parametric: Wilcoxon
- Correlation tests:
  - Pearson linear correlation
  - Rank correlation (Kendall, Spearman)
- Anova

3. Matics Capabilities and Concepts

3.1. Data Management

3.1.1. Dataframe

The main underlying concept used in the Evalomatic interface is the *Dataframe*. It is a table roughly equivalent to a single SQL table or a R data frame. Each column has a two-part name: a group name and a column name. For instance the "speaker" group may have the columns "speaker.name", "speaker.gender" and "speaker.accent". Each column has a type that is built from four traits:

1. The column may contain *labels* or *values*: labels are names into categories (file name, speaker name, gender, turn id...) while values are actual values (time, score, word, text segment...).

2. The datatype of the column content can be string or numeric (integer for labels, floating-point for values).

3. The column can store the initial values/labels, or values computed from other columns through expressions.

4. (optional) The column can have a sub-type that tells the interface how to show or interpret the values. Currently defined sub-types are name, time, p-value (for statistical tests) and correlation (for correlation tests).

Non-expression columns actually store data. A stored value often span multiple lines. For instance, a speech transcription evaluation dataframe has one line per aligned word. In that dataframe, the turn start and end times span all the words of that turn. That spanned information is explicitly stored in the dataframe. In addition, some cells can be empty, which is a different status from zero or an empty string.

3.1.2. Granularity and Foldable Categories

A key capability of that dataframe structure is a variable granularity. Lines can be folded together, and columns optionally have a folding method, called reduce operation, which defines how the value for the folded lines is computed. A number of reduce operations are already available: min, max, mean, median, sum for numeric values, concatenation for string values. Expression columns either include a reduce operation, and then compute their value at the lowest possible granularity then apply the operation, or do not include a reduce operation and compute the value from the reduced values of the other columns.

To illustrate that capability, two examples can be given. Computing the WER in speech transcription is done by dividing the count of errors by the number of words in the reference. The WER is then an expression column without reduce operation which divides the value in the error column by the value in the reference words count column. These two source columns on the other hand have a "sum" reduce operation to accumulate the count of errors and words at the required granularity.

In contrast, computing the total speech time is done from the speech duration column which is an expression defined as turn end time minus turn start time, with a sum reduce operation. In that configuration the durations are computed at the turn level and summed together, giving the total turn time. The spans of the values in the start and end time columns are what sets the duration computation granularity.

3.2. Evaluation capabilities

As detailed in Sections 2.3. and 2.5., Matics can deal with the evaluation of several NLP tasks and implements the corresponding metrics. All the input formats are converted to reference or hypothesis dataframes which are then used to build an evaluation dataframe with a complete alignment of the texts.

The ASR evaluation subsystem, for example, is able to work on the word or character level, and take the case into account when requested to. It uses unicode for multilingual support.

The final evaluation dataframe contains the full alignment and the error counts per type, with computed columns added to provide WER/CER (Crossover Error Rate) and NCE (Normalized Cross Entropy) at any chosen granularity.

3.3. Statistical analysis capabilities

One aim of the interface is to give a fast access to statistical testing capabilities. The list of the currently available functions has been presented in Section 2.6.. A uniform, drag'n'drop based interface is proposed to select the data columns the testing applies to.

In the case of the standard descriptive statistics on a value, for example, the user selects a value column to compute the statistic on (for instance WER) and a label column for the granularity (for instance speech turn). They can also optionally select a factor column as a factor (for instance "System" – the NLP system of which we evaluate the output results) to compute a series of statistics instead of a global one. The computation of these statistics allows the user to summarize the distribution of the values and get an idea of how gaussian and symmetrical they are.

The second available analysis is a very common one: significance of a difference for paired values. The user selects a value (e. g. WER), a pairing/granularity (speech turn) and the factor to analyze (system) and the interface computes, for each system pair, the p-value, e. g. the probability that the WERs are in practice identical and the differences only randomness. It can use either a Student paired t-test if the user considers the values gaussian (which is rare), or a less powerful but more robust Wilcoxon paired-difference test otherwise.

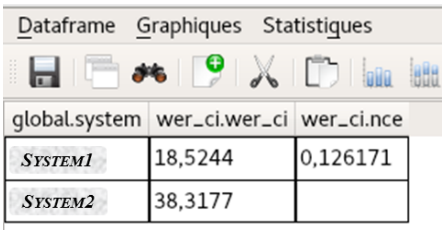
The third analysis is a correlation test between two value columns to, for instance, check whether the WER is correlated with the turn duration. The user selects the values to compare and the ganularity. The interface then computes three standard correlation values: Pearson's r (linear correlation), Kendall's τ and Spearman's ρ (rank correlations).

Finally a fourth analysis method is implemented: the Anova. It is used to measure the importance of different factors on a result, and measures how much of the variance can be explained by each factor. It should be available by the time the final papers are due.

4. User Interface

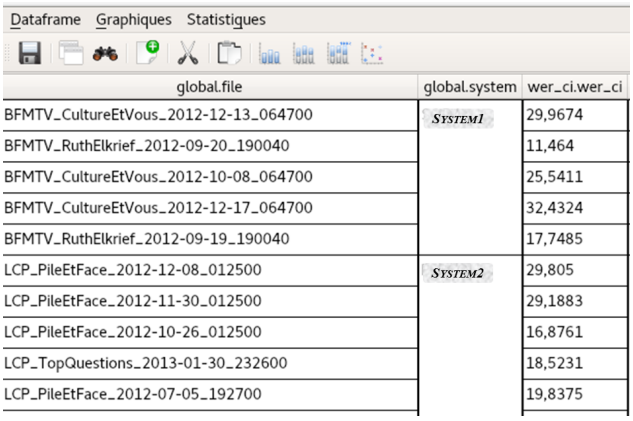
This section presents some views of the user interface offered by Matics.

4.1. Dataframe views



global.system	wer_ci.wer_ci	wer_ci.nce
SYSTEM1	18,5244	0,126171
SYSTEM2	38,3177	

Figure 1: Evaluation dataframe with every line folded but the system name. The WER is updated and represents the global score for all the files of the system.



global.file	global.system	wer_ci.wer_ci
BFMTV_CultureEtVous_2012-12-13_064700	SYSTEM1	29,9674
BFMTV_RuthElkrief_2012-09-20_190040		11,464
BFMTV_CultureEtVous_2012-10-08_064700		25,5411
BFMTV_CultureEtVous_2012-12-17_064700		32,4324
BFMTV_RuthElkrief_2012-09-19_190040		17,7485
LCP_PileEtFace_2012-12-08_012500	SYSTEM2	29,805
LCP_PileEtFace_2012-11-30_012500		29,1883
LCP_PileEtFace_2012-10-26_012500		16,8761
LCP_TopQuestions_2013-01-30_232600		18,5231
LCP_PileEtFace_2012-07-05_192700		19,8375

Figure 2: Evaluation dataframe for the comparison of two systems (names have been blurred out). Every line folded but the system and file name. wer_ci: WER case independent.

The main interaction is done with the dataframe. A dataframe has views on it, where each view has its own state. The user has control over which columns are visible and in which order they appear. The display granularity is implicitely controlled by the visible columns: consecutive lines with identical labels in all the label columns are collapsed. If every column is hidden except for the system name and the WER, then the per-system WER is visible, as can be seen in Figure 1. When the file name column is then shown, the per-file score then becomes visible, as in Figure 2. The dataframe can also be sorted on the columns, giving the possibility to get a per-speaker score in a dataframe originally generated with lines in time order.
Filtering is also possible, to view a subset of the lines. The active granularity and filtering is taken into account when doing a graph, while only the filtering is taken into account for statistical tests and the granularity is requested explicitely.

4.2. Data visualization

The interface gives the capability to link to source data (audio, video, etc.) and visualize the annotations present in a dataframe with an appropriate alignment to the original signal.
The audio display and listening is currently available. The interface allows listening to the signal at different levels: the whole signal, per speaker, per sentences, or per words. The segmentation follows the timestamps defined in the corresponding annotation file. The Figure 3 shows a screen capture of the interface.

4.3. Statistical Functions Selection

The selection form can be seen in Figure 4. The user can drag and drop between the column list at the bottom and the configurable fields in the middle.

4.4. Graphing

The other main capability of the interface is graphing data, to ease the visualization of data and results.

- Bar charts — The histogram graphic category can plot any value. The basic histogram allows graphing of one or more value columns with one or more label columns on the x axis. This allows counting the number of different labels in one column, using another for the x axis (for instance counting the speakers in each show) with optionally a third used to color subparts of the histogram (gender for instance). An optional gaussian curve can be overlaid.
- Scatterplots — The scatterplots can be created from two value columns with color and shape controlled from label columns. An example of scatterplot showing the lack of correlation between file speech duration in a file and the WER can be seen in Figure 5.
- Boxplots — Visualisation of the distribution of the data, through quartile and decile. A same graph can show the boxes for different factors (file, system...).
- Detection Error Tradeoff (DET) curves — For binary classification. The DET curves for several systems can be presented at once for visual comparison, with a visualization of EER and C_{det} decision thresholds. See an example Figure 6.

5. Conclusion

The Matics software suite offers a unified tool for the evaluation of NLP systems, through two independent tools: Datomatic and Evalomatic. Datomatic allows the manipulation, visualisation and sub-selection of hypothesis and reference corpora; evaluations can be conducted in Evalomatic, with metrics implemented for a range of NLP applications.
Developed by the LNE, specialized in the evaluation of NLP systems, Matics is free and open-source. While still in the development stage, the tool aims at providing a concrete and fully reusable solution for data exploration and evaluation. New features are expected to be implemented, and regular updates of the system will be offered according to the evolution of our evaluation activities.
For example, an expected upcoming feature is video synchronization with the annotation (for Datomatic). We are

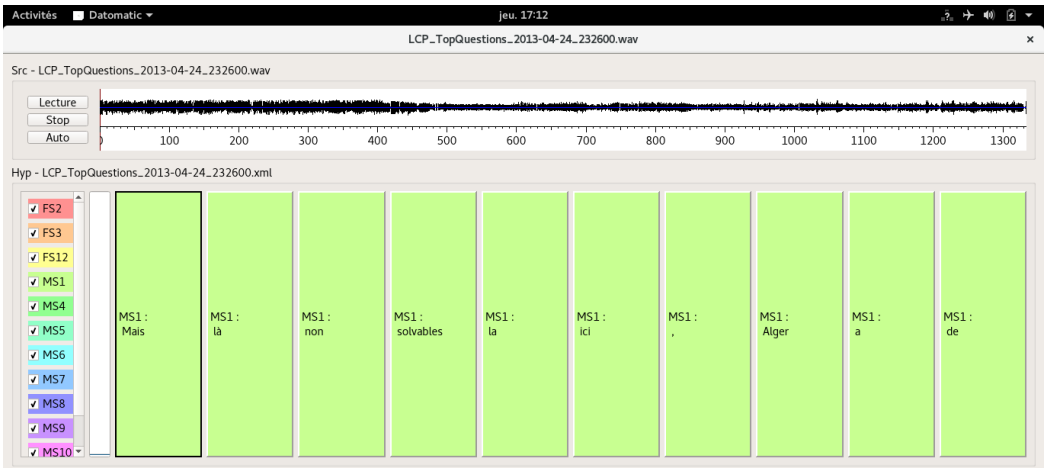
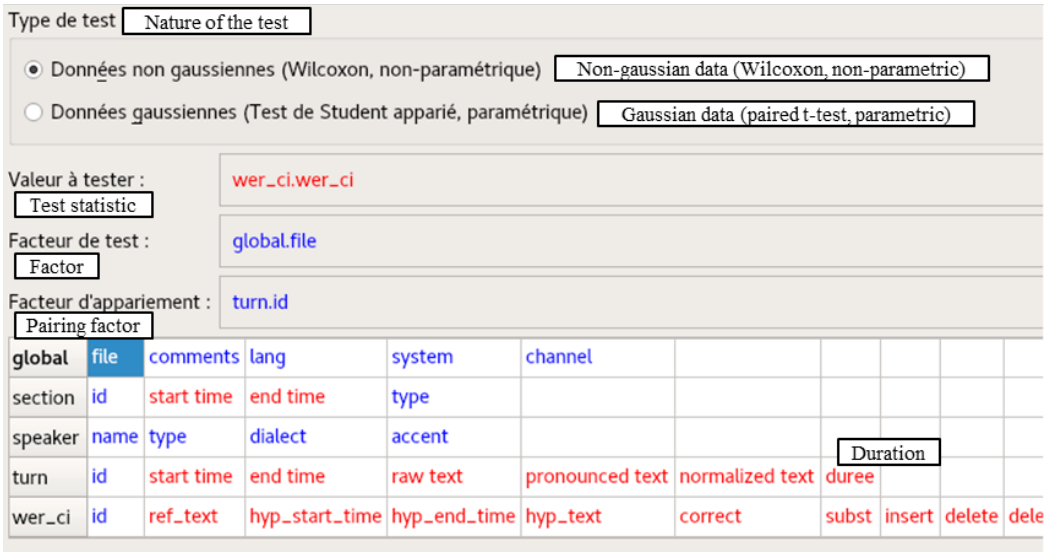


Figure 3: Audio signal and the associated transcription. A click on each token (one per rectangle in the center area) plays the corresponding audio segment.



global	file	comments	lang	system	channel					
section	id	start time	end time	type						
speaker	name	type	dialect	accent						
turn	id	start time	end time	raw text	pronounced text	normalized text	duree			
wer_ci	id	ref_text	hyp_start_time	hyp_end_time	hyp_text	correct	subst	insert	delete	dele

Figure 4: Statistical paired difference configuration interface. In blue (e. g. comments, lang, system): labels of the columns; in red (e. g. start time, end time): values of the columns. Labels in English have been added on the figure to translate the French items.

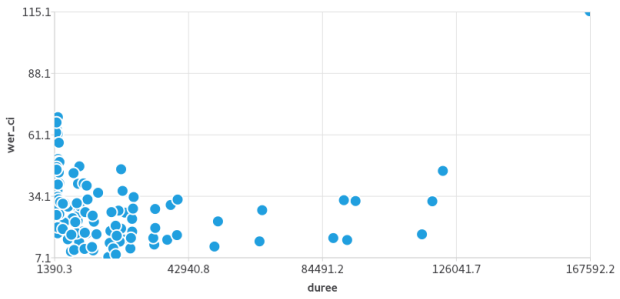


Figure 5: Scatterplot of WER vs. speech duration in a file

also concerned with a localization feature, to broaden out the system to the non-French speaking community. Although the interface vocabulary may be quite transparent to computer scientists and statisticians, that would be a strong requirement in terms of ergonomics. The localization process requires some modification at the core of the system that will be addressed soon.

A longer term perspective is to give the interface the capability to rewrite the different supported formats, and use that capability combined with statistical analysis possibilities to select representative subsets of data for train, development and test. This aspect, while quite out of the scope of evaluation, is also part of our mission of accompanying technology developers.

6. Bibliographical References

Bernard, G., Galibert, O., Rémi, R., Demeyer, S., and Kahn, J. (2016). LNE-Visu : une plateforme d'exploration et de visualisation de données d'évaluation (LNE-Visu: a platform for the exploration and display of evaluation data). In *Proceedings of the JEP-TALN-Recital joint conference*, 07.

Brunessaux, S., Giroux, P., Grilhères, B., Manta, M., Bodin, M., Choukri, K., Galibert, O., and Kahn, J.

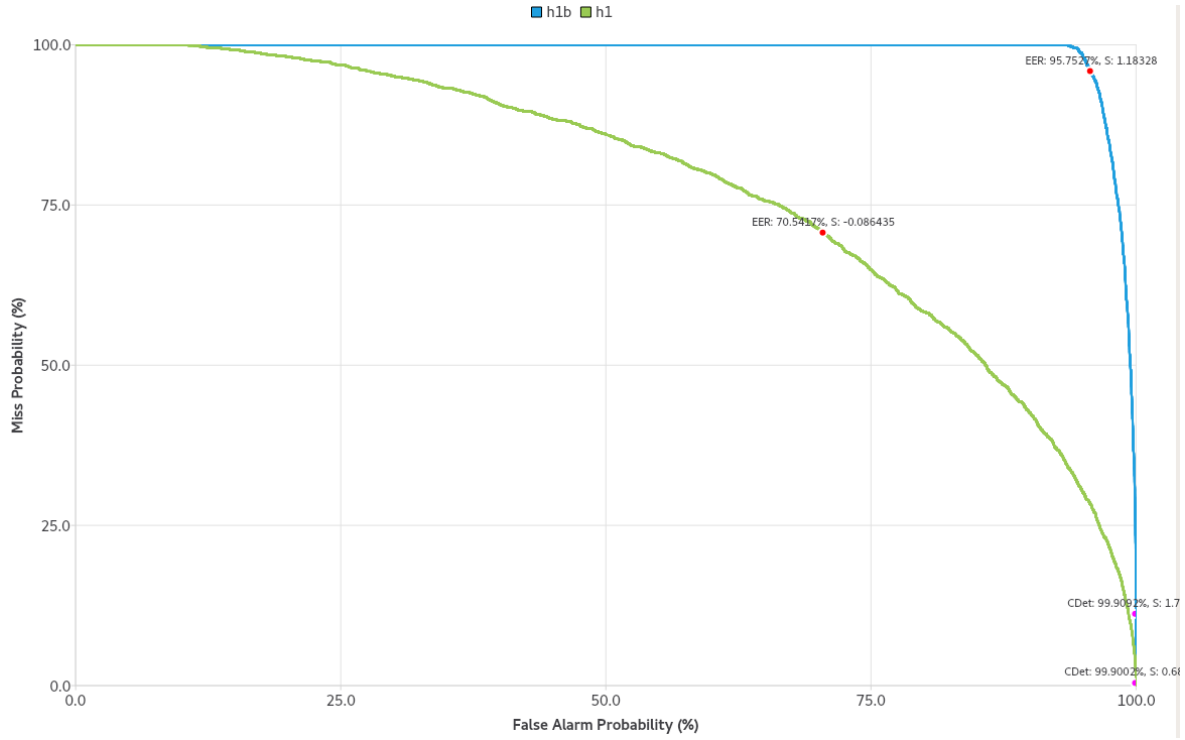


Figure 6: DET curves for two simulated systems (named h1b and h1) with EER and C_{det} decision thresholds.

(2014). The Maurdor Project: Improving Automatic Processing of Digital Documents. In *2014 11th IAPR International Workshop on Document Analysis Systems*, pages 349–354, April.

Chinchor, N. and Robinson, P. (1997). MUC-7 named entity task definition. In *Proceedings of the 7th Conference on Message Understanding*, volume 29.

Galibert, O. and Kahn, J. (2013). The First Official REPERE Evaluation. In *First Workshop on Speech, Language and Audio in Multimedia (SLAM'13)*.

Galibert, O., Rosset, S., Grouin, C., Zweigenbaum, P., and Quintard, L. (2011). Structured and Extended Named Entity Evaluation in Automatic Speech Transcriptions. In *Proc of IJCNLP*, Chiang Mai, Thailand, 9-11 novembre.

Giraudel, A., Carré, M., Mapelli, V., Kahn, J., Galibert, O., and Quintard, L. (2012). The REPERE Corpus : a multimodal corpus for person recognition. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Gravier, G., Adda, G., Paulson, N., Carré, M., Giraudel, A., and Galibert, O. (2012). The ETAPE corpus for the evaluation of speech-based TV content processing in the French language. In *LREC - Eighth international conference on Language Resources and Evaluation*, Turkey.

Kipp, M. (2010). Multimedia annotation, querying and analysis in ANVIL. *Multimedia information extraction*, 19.

Knight, D., Evans, D., Carter, R., and Adolphs, S. (2009). HeadTalk, HandTalk and the corpus: Towards a framework for multi-modal, multi-media corpus development. *Corpora*, 4(1):1–32.

Malmasi, S., Zampieri, M., Ljubešić, N., Nakov, P., Ali, A., and Tiedemann, J. (2016). Discriminating between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 1–14, Osaka, Japan, December.

NIST. (2015). NIST Multimodal Information Group - Tools. `https://www.nist.gov/itl/iad/mig/tools`. Accessed: 2018-02-21.

O'Donnell, M. (2008). The UAM CorpusTool: Software for corpus annotation and exploration. In *Proceedings of the XXVI Congreso de AESLA*, pages 1433–1447, 01.

Oepen, S., Øvrelid, L., Björne, J., Johansson, R., Lapponi, E., Ginter, F., and Velldal, E. (2017). The 2017 Shared Task on Extrinsic Parser Evaluation Towards a Reusable Community Infrastructure. *EPE 2017*, page 1.

Pierpaolo, B., Malvina, N., Patti, V., Rachele, S., and Francesco, C. (2017). EVALITA Goes Social: Tasks, Data, and Community at the 2016 Edition. *Italian Journal of Computational Linguistics*, 3(1):93–127.

Schmitt, A., Bertrand, G., Heinroth, T., Minker, W., and Liscombe, J. (2010). WITcHCRafT: A Workbench for Intelligent exploraTion of Human ComputeR conversaTions. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Schuller, B., Steidl, S., Batliner, A., Bergelson, E., Krajewski, J., Janott, C., Amatuni, A., Casillas, M., Seidl, A., Soderstrom, M., et al. (2017). The inter-

speech 2017 computational paralinguistics challenge: Addressee, cold & snoring. In *Computational Paralinguistics Challenge (ComParE), Interspeech 2017*, pages 3442–3446.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans et al., editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

MGAD: Multilingual Generation of Analogy Datasets

Mostafa Abdou[*⋆◇†], **Artur Kulmizev**[*⋆♣], **Vinit Ravishankar**[†⋆△]
[*]University of Groningen, [†]Charles University in Prague
[*]Center for Language and Cognition, [†]Institute of Formal and Applied Linguistics
[◇]m.abdou@student.rug.nl, [♣]a.kulmizev@student.rug.nl, [△]vinit.ravishankar@gmail.com

Abstract

We present a novel, minimally supervised method of generating word embedding evaluation datasets for a large number of languages. Our approach utilizes existing dependency treebanks and parsers in order to create language-specific syntactic analogy datasets that do not rely on translation or human annotation. As part of our work, we offer syntactic analogy datasets for three previously unexplored languages: Arabic, Hindi, and Russian. We further present an evaluation of three popular word embedding algorithms (Word2Vec, GloVe, LexVec) against these datasets and explore how the performance of each word embedding algorithm varies between several syntactic categories.

Keywords: word embeddings, evaluation, multilingual, natural language processing

1. Introduction

Since the emergence of dense word embeddings (Mikolov et al., 2013c), a sizable amount of work concerning the evaluation of their quality as linguistic representations has emerged. These evaluation methods can be divided into two classes: extrinsic and intrinsic. The former involves testing the performance of embeddings on various NLP tasks such as part of speech (POS) Tagging, Machine Translation, etc. (Schnabel et al., 2015). However, these methods are computationally expensive and task-specific, i.e. they solely demonstrate whether an embedding is suitable for one particular task and are therefore not suitable as a (quick) test of general quality. Intrinsic methods, on the other hand, are designed to test the degree to which a set of embeddings can model a certain linguistic property. This typically involves constructing human evaluated datasets to directly test syntactic or semantic relationships between words (Gladkova and Drozd, 2016). Embeddings are then typically evaluated by an aggregate score (e.g. a correlation coefficient) using of a set of query words and semantically related target words. This score serves as a measure of quality.

The majority of datasets created for intrinsic evaluation have focused on non-specific word relatedness (Bruni et al., 2014) or word similarity (Hill et al., 2014). More recently, the analogy-based task first proposed by Mikolov et al. (2013a) has gained popularity. This task involves retrieving a term by solving analogy questions of the form "a is to b as c is to X" using vector arithmetic (the most recognisable example of which is: *king - man + woman = queen*). However, the vast majority of analogy datasets have been constructed only for the English language, making it harder to evaluate whether the reported performance of popular embedding algorithms continue to hold when evaluated against previously untested languages. As such, this paper is concerned with generating multilingual syntactic analogy datasets for intrinsic evaluation of word embeddings. Specifically, we present a method of leveraging existing resources such as dependency parsed corpora to automatically generate the datasets. We argue that this method is advantageous in multilingual environments, where inflectional morphology can vary greatly between languages.

In Section 2., we briefly review the relevant literature in the domain. Section 3. provides more detail about what we are trying to achieve with our system, and why this is necessary. Section 4. describes our methodology in detail. Section 5. describes our evaluation parameters and how they are meaningful, whilst Section 6. describes our actual results, and demonstrates an analysis from a quantitative and qualitative perspective. Finally, we describe potential future extensions and improvements in Section 7. and conclude in Section 8.

2. Related Work

The Google analogy dataset (Mikolov et al., 2013b) is arguably the most widely adopted analogy dataset, comprising of 10,675 syntactic questions and 8,869 semantic questions (19,544 total). The former category consists of questions that aim to capture syntactic regularities that manifest in English (e.g. `good : better :: rough : ?`). Such questions were generated by POS-tagging a corpus of 267M words and extracting varying forms of adjectives, nouns, and verbs as represented by the Penn TreeBank (Marcus et al., 1993)(`JJ, JJR, JJS, NN, NNS,` etc.). The semantic questions portion of the dataset is largely comprised of `capital-country` pairs, which were manually chosen by a group of annotators. These pairs were then randomly combined with other such pairs in order to produce the entirety of the semantic portion.

Gladkova et al. (2016) expanded on the work of Mikolov et al. (2013b) in generating a balanced set of 99,200 analogy questions titled "The Bigger Analogy Test Set" (hereafter referred to as 'BATS'). These analogies were again separated into syntactic and semantic categories, each of which were further partitioned into inflectional/derivational and lexicographic/encyclopedic questions, respectively. In total, the project covered 40 different linguistic relations, each of which were represented by 2,480 questions. A distinguishing feature of BATS, outside of its breadth, was that it was designed to reduce homonymy, which occured often in

[*]Equal contribution. Listing order is random.

the Google analogy set. This was done by removing every word in the reference corpus that was attributed to more than one part of speech in the English WordNet. As such, pairs like *walk, walks*, both words of which could be attributed to nouns or verbs, were excluded.

The other task employed in the intrinsic evaluation of word embeddings is the word similarity/relatedness task. A commonly utilized dataset of this type is the WordSimilarity-353 Test Collection (Agirre et al., 2009). This dataset is separated into two parts which contain 153 and 200 word pairs, respectively. Each word pair is also coupled with human judgments of the two words' relatedness, represented by a 10-point Likert scale, with 0 representing completely unrelated words and 10 representing very related or identical words. Evaluating word embeddings against this dataset typically involves finding the cosine distance between both words in a word pair and generating the Spearman correlation between the human relatedness judgments and the cosine distance.

3. Motivation

Though each of the aforementioned datasets is generally suitable for the evaluation of English word embeddings, each of them fails to introduce a sustainable framework for generating similar datasets for other languages. For example, in order to recreate the WordSimilarity-353 dataset for any other language, it would be necessary to solicit the judgments of native-speakers of that language, which is a generally expensive task. The same can be said for the semantic portions of the Google and BATS datasets, the relations of which were produced by native English speakers and might not necessarily hold cross-linguistically. Furthermore, Google and BATS' reliance on the Penn Tree-Bank results in a failure to capture many linguistic features that do not meaningfully occur in English (case, animacy, etc.). Therefore, this makes a direct translation of either dataset not a particularly robust approach for the evaluation of non-English embeddings. In this work, then, we propose a method for the generation of analogy datasets which can be generalized for many languages. Our approach for generating syntactic analogies demonstrates that it is possible to create datasets cross-linguistically in a manner that is both low-resource and sensitive to the particularities of any given language. We test this method on three languages: Arabic, Russian, and Hindi. Our choice of languages is based on two factors. First, the dissimilarity of the languages is meant to demonstrate the cross-lingual robustness of the method. Secondly, each of this paper's authors is a native speaker of one of the languages, and so can serve as an annotator to manually check the quality of the generated datasets.

4. Methodology

4.1. Dependencies

Constituency grammars (Tesnière, 1965) have long been employed for formal representations of language grammars. However, though relatively recent, dependency grammars are an extension of this approach that offer several advantages to the former. In particular, dependency grammars' independence from a static word order allows a number of typologically varying languages to be properly represented. These advantages are reflected in the Universal Dependencies (UD) project (Nivre et al., 2016), which attempts to model dependency relations *cross-linguistically* based on a framework that holds true across all represented languages. UD treebanks typically use the 10-column CoNLL-U format for representing sentences and dependencies. Of interest to us, however, is the features column, which stores morphological information. Similar to the dependency annotations in the schema, these features are represented by cross-linguistic labels and the same format for every language, significantly simplifying querying different languages.

4.2. Templating

Though it would be ideal to have a fully automatic method of generating morphological analogies, this fails for several reasons. The most obvious of these is that morphological distinctions used across languages are far from universal, even within language families, let alone across. Another issue is the existence of morphological information that needs to be 'controlled' for; for instance, whilst BATS includes an example analogy of the English third-person singular to the infinitive, transferring this to Hindi would be non-trivial: several 'new' features that do not exist in English would need to be controlled for, such as gender or aspect; not fixing or registering these would result in multiple analogies.

To circumvent these issues, we design a simple Python script[1] that can parse 'templates' that define precisely what analogies need to be generated for a particular language. Table 1 is a (much truncated) example of one such template; the syntax for referencing the appropriate morphological features is similar to the syntax followed by UD, allowing someone with moderate familiarity with the schema to rapidly create their own templates.

The fact that UD treebanks typically store lemmas for every word results in two classes of analogies: 'core' analogies, that include the lemma, and 'composite' analogies between two non-lemmatic categories. In English, the former category would translate to something similar to `eat:eats` while the latter would be `eats:eaten`. The core analogies are trivial to generate using the lemma field; the latter, however, involve using the lemma as a link between the two word forms participating in the analogy. The analogies are further divided into `nominal` and `verbal` categories, representing noun-based and verb-based analogies, respectively. Table 2 displays the number of analogies generated, along with the distribution of their class and syntactic categories. The first row, **Type**, shows the number of 'core' and 'composite' analogies; **POS** shows the number of nominal and verbal analogies; and the last row shows the total number of analogies. Table 3 shows a selection of the generated analogy templates along with examples in each of the three languages.

4.3. Corpora

Whilst some treebanks such as Czech and Russian are large enough to provide a sufficient amount of lexical entries to

[1] All code, datasets, and templates are publicly available at: https://github.com/rutrastone/MGAD

NOUN|Number=Plur|Case=Nom	NOUN|Number=Sing|Case=Nom
NOUN|Number=Plur|Case=Acc	NOUN|Number=Sing|Case=Acc
VERB|Aspect=Perf|Gender=Masc|Number=Sing|VerbForm=Part	VERB|Case=Nom|VerbForm=Inf
VERB|VerbForm=Conv	VERB|Case=Nom|VerbForm=Inf
VERB|Case=Acc|VerbForm=Inf	VERB|Case=Nom|VerbForm=Inf

Table 1: Truncated example of a template.

	Russian	**Arabic**	**Hindi**
Type	$^{12.5}/_{35}$	$^{7.5}/_{12.5}$	$^{15}/_{12.5}$
POS	$^{25}/_{22.5}$	$^{2.5}/_{17.5}$	$^{10}/_{17.5}$
=	47.5	20	27.5

Table 2: Number of analogies (in 1,000s) generated per Type (core/compound), Part of Speech (noun/verb), and total number of analogies per language.

generate an analogy set, many are often too small and too domain-constrained to provide enough data. Therefore, we use a larger morphologically analysed corpus (Wikipedia dumps) to generate our data set. We use the MorphoDita (Straková et al., 2014) morphological analyser, trained on the manually annotated treebanks. The dependency parsing pipeline UDPipe (Straka and Straková, 2017) conveniently provides an easy-to-use wrapper for MorphoDita.

From these parsed corpora, we use our template extraction script to extract all relevant linguistic information. In generating analogies for Russian nouns, for example, we build the nominal part of our analogy set with all noun case combinations, with fixed number (eg. `+Nom+Sg:+Gen+Sg`, and with varying number and fixed case (eg. `+Nom+Pl:+Gen+Pl`. This has the effect of covering the entire breadth of cases and numbers that can possibly exist in Russian and is thus a robust evaluation of how these features are represented in any target embedding space.

An issue with this method is that languages with richer morphology would, undoubtedly, generate a much larger number of analogies than more morphologically simpler languages. We posit that this side-effect is a beneficial one, allowing the output dataset to fully assess the morphological breadth of inflectionally-rich languages on which the target embeddings are trained.

The second issue in generating our dataset is the question of word pair frequency. We follow the method used in BATS, in selecting the fifty most frequent pairs of each relation in our corpora. The frequency of a composite pair is, however, set to the minimum of the frequencies of its two constituent core pairs.

5. Evaluation

5.1. Word embeddings

In order to evaluate our datasets, we train word embeddings for each of the languages using three popular unsupervised methods: Skipgram with negative sampling (SGNS)(Mikolov et al., 2013c), GloVe (Pennington et al., 2014), and LexVec (Salle et al., 2016). These methods are a small but representative subset of the vast number of algorithms which have been proposed. However, as our focus is on evaluation datasets, we restrict our testing to these three methods and make no attempt to fine-tune the embeddings. All sets of embeddings are of dimension 300 and were trained with a window-size 5 for 5 epochs. SGNS and LexVec embeddings were trained with a learning rate of 0.05 and with 5 negative samples per training iteration, while GloVe embeddings were trained with the default learning rate of 0.025. All embeddings were trained with a minimum word count of 10 as the threshold for inclusion in the vocabulary. The Russian and Arabic embeddings were trained on their respective wikipedia dumps. However, HindMonoCorp 0.5 (Bojar et al., 2014) was used instead for Hindi, as the wikipedia corpus was significantly smaller than the others.

5.2. The Task

The task is to retrieve an answer to the question "a is to b as c is to X" as represented by hidden vector d, which is calculated as $argmax_{d \in V}(sim(d, c - a + b))$ where V is the vocabulary excluding vectors a, b, and c. We define similarity as the angular distance (cosine similarity) between vectors u and v:

$$similarity(w_1, w_2) = \frac{\vec{w_1} \cdot \vec{w_2}}{\|\vec{w_1}\|\|\vec{w_2}\|}$$

Though work by Levy and Goldberg (2014) and Linzen (2016) has shown that other functions may outperform cosine similarity on the analogy task, we nonetheless employ it in the interest of comparability with the majority of previous work.

6. Results

We report coverage[2] and accuracy over the full test datasets and separately for the nominal and verbal categories for each language per set of embeddings. A question is not covered if one or more of the words contained in it are not found in the embedding's vocabulary. Table 4 is a summary of the results.

6.1. Quantitative

Table 4 demonstrates that each embeddings model performs similarly across all three languages. This indicates the cross-linguistic stability of the datasets as a method of evaluation. Several things stand out in the results, however - particularly Hindi's excellent coverage. This is likely because the domain of our corpus is very similar to the domain of the dependency treebank, and also possibly significantly cleaner than Wikipedia as a corpus. Also, it is apparent that GloVe fares comparatively much worse for Russian

[2]Slightly different coverage over a language's dataset for different sets of embeddings is explained by the 'minimum word count' parameter being considered exclusive or inclusive (GloVe embeddings have a slightly larger vocabulary).

Category	Template Rule	Example	Gloss
Nominal	NOUN|Number=Plur|Case=Nom :	महीने महीना	month.PL month.SG
	NOUN|Number=Sing|Case=Nom	मामले मामला	issue.PL issue.SG
	NOUN|Number=Plur|Case=Nom|Definite=Ind :	مسئول مسؤولون	official.PL official.SG
	NOUN|Number=Sing|Case=Nom|Definite=Ind	مصدر مصادر	source.PL source.SG
	NOUN|Number=Sing|Case=Dat :	человеку человек	person.DAT person.NOM
	NOUN|Number=Sing|Case=Nom	жизни жизнь	life.DAT life.NOM
Verbal	VERB|Aspect=Perf|Gender=Masc|Number=Sing|VerbForm=Part :	दिखाया दिखाना	show.PERF.3MSG show.INF
	VERB|Case=Nom|VerbForm=Inf	कहा कहना	say.PERF.3MSG say.INF
	VERB|Aspect=Imp|Gender=Masc|Person=3|Number=Sing :	أراد يريد	want.IMPF.3MSG want.PERF.3MSG
	VERB|Aspect=Perf|Gender=Masc|Person=3|Number=Sing	توجه يتوجه	head_to.IMPF.3MSG head_to.PERF.3MSG
	VERB|Person=3|Number=Sing|Tense=Pres :	живет жить	live.PRES.3SG live.INF
	VERB|Aspect=Imp|VerbForm=Inf	стоит стоять	stand.PRES.3SG stand.INF

Table 3: Generated analogy templates and corresponding example analogies in Hindi, Arabic, and Russian

		word2vec		GloVe		LexVec	
		Coverage	**Accuracy**	**Coverage**	**Accuracy**	**Coverage**	**Accuracy**
Russian	**N**	79.09	25.41	81.78	17.91	79.09	22.23
	V	28.70	36.75	81.17	18.75	28.70	33.43
	=	55.23	28.20	81.50	18.30	55.23	25.09
Arabic	**N**	88.52	33.19	97.98	13.54	88.52	30.14
	V	56.30	24.69	98.67	34.41	56.30	22.93
	=	64.36	27.61	98.50	29.21	64.36	25.41
Hindi	**N**	98.02	33.14	99.98	62.85	98.02	29.45
	V	86.41	40.65	87.94	25.01	86.41	38.38
	=	90.62	37.69	92.32	39.91	90.62	34.87

Table 4: Word embedding performance on three generated sets. N, V and '=' indicate performance on nominal and verbal sections, and on a combination of the two

than it does for Arabic and Hindi. This could be related to the model's count-based implementation (as opposed to the prediction-based word2vec and LexVec), which may fail to represent low-occurring case inflection contexts in case-rich languages such as Russian.

6.2. Qualitative

Each of the three generated datasets was manually checked for correctness by one of the three authors who is a native speaker of the language. Whilst judging the correctness of our datasets is trivial, there is no simple method to judge their validity. The distinction here is that our choice of morphological forms for analogies is partly arbitrary, motivated solely by linguistic intuition of the language. There were also some areas where we were limited by the treebanks themselves: for instance, it was impossible to include inflectional adjective degrees (such as the comparative and superlative), since UD does not normalise lemmas across adjective degrees, making it impossible to "link" a positive adjective with its equivalent superlative form. For Arabic, case could not be included as a category due to both the Wikipedia corpus and the UD treebank not featuring diacritics which are used to distinguish the case of a noun.

6.3. Validity

In order to further ascertain the validity of our framework, we generate an English analogy dataset and use it to evaluate three sets of commonly utilized, publicly available pre-trained word embeddings: the Google News corpus embeddings (Mikolov et al., 2013a), Glove embeddings trained on Common Crawl (Pennington et al., 2014) (42B tokens), and LexVec embeddings trained on Common Crawl (58B tokens). We then compare the results to evaluation using the the syntactic half of the BATS dataset. As mentioned earlier, the syntactic portion of BATS includes two subcategories: inflectional and derivational analogies. The former class consists of structures such as regular plurals (`student:students`), infinitive:participle (`follow:following`) and participle:past (`following:followed`). The latter is is further divided into *stem change* and *no stem change* analogies: noun+less (`life:lifeless`) VS. verb+ation (`continue:continuation`). It is important to note that, though we attempted to replicate every relation used in the set of syntactic pairs in BATS, we were unable to generate adjective degree sets; as mentioned earlier, UD does not typically map different adjective degrees to the same lemma. We therefore generated solely the nominal and verbal sets, resulting in 17,500 pairs. The exact relations used are mentioned in the original paper (Straková et al., 2014); we do not replicate them here for brevity. The results (accuracy) are shown in Table 5 and they show a strong correlation between the results obtained by evaluating using the MGAD dataset and those obtained by using the syntactic half of BATS.

	word2vec	GloVe	LexVec
BATS (Syntactic)	68.41	63.17	68.41
MGAD	69.92	68.10	64.41

Table 5: Results (accuracy) of evaluating three sets of word embeddings using the syntactic half of BATS and the dataset generated using MGAD.

7. Future Work

7.1. Semantic relations

Our method only took syntactic relations into consideration when generating the datasets. Further work will explore resources like multilingual wordnets and Wiktionary in order to generate semantic datasets to serve as a complement to the syntactic ones we've generated here. Given the breadth of languages that are annotated in the latter source, we would like to explore the extent to which semantic relations like hypernymity and meronymity could be generated in an automatic or semi-supervised manner. Furthermore, it will be important to explore how previously-annotated English-language datasets could be translated in an efficient manner while retaining uniform syntactic relations between the terms in the analogy. Though we attempted to automatically translate the capital:country relations in the Google dataset, (e.g. case inconsistently was inevitable in every language we evaluated (Токио`(nom)`:Япония`(nom)` :: Париже`(prep)`:Франция`(nom)`; `Tokyo:Japan :: (in)Paris:France`).

7.2. Data sources

Our justification for using Universal Dependencies treebanks as our source of morphological data was obvious: the treebanks were cross-linguistic, and most were well-annotated morphologically. There are, however, several directions into which we could branch for more comprehensive resources, combined with our present approach. The Apertium project (Forcada et al., 2011) includes approximately 73 morphological analysers (with varying quality). These include several analysers for several minority languages that lack dependency treebanks, such as Kyrgyz (Washington et al., 2012), Marathi (Ravishankar and Tyers, 2017) and Sardinian (Tyers et al., 2017). The relative lack of standardization across Apertium morphological analysers, however, makes it significantly harder to partially automate than Universal Dependencies, and extraction would entail attaining some familiarity with a specific analyzer's structure.

8. Conclusion

In this work we presented a method for generating analogy datasets to be used for the evaluation of word embeddings for a large number of languages. Our method utilizes readily available resources (UD Treebanks and morphological analyses) and requires minimal human supervision. A quantitative evaluation reveals that our results reflect the expected performance of several popular word embedding models across the three represented languages, emphasizing the validity of our approach. Lastly, we release three syntactic analogy datasets for Russian, Arabic, and Hindi.

9. Acknowledgements

Work on this paper has been supported by the Erasmus Mundus Language and Communication Technologies grant. We thank Rik van Noord for his helpful comments.

10. Bibliographical References

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. Association for Computational Linguistics.

Bojar, O., Diatka, V., Rychlý, P., Straňák, P., Suchomel, V., Tamchyna, A., and Zeman, D. (2014). HindMonoCorp 0.5. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Bruni, E., Tran, N. K., and Baroni, M. (2014). Multimodal distributional semantics. 11.

Forcada, M. L., Ginestí-Rosell, M., Nordfalk, J., O'Regan, J., Ortiz-Rojas, S., Pérez-Ortiz, J. A., Sánchez-Martínez, F., Ramírez-Sánchez, G., and Tyers, F. M. (2011). Apertium: a free/open-source platform for rule-based machine translation. *Machine translation*, 25(2):127–144.

Gladkova, A. and Drozd, A. (2016). Intrinsic evaluations of word embeddings: What can we do better? In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 36–42.

Gladkova, A., Drozd, A., and Matsuoka, S. (2016). Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15.

Hill, F., Reichart, R., and Korhonen, A. (2014). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *CoRR*, abs/1408.3456.

Levy, O. and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185.

Linzen, T. (2016). Issues in evaluating semantic spaces using word analogies. *arXiv preprint arXiv:1606.07736*.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013b). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013c). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., and others. (2016). Univer-

sal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Ravishankar, V. and Tyers, F. M. (2017). Finite-state morphological analysis for marathi. In *Proceedings of the 13th International Conference on Finite State Methods and Natural Language Processing (FSMNLP 2017)*, pages 50–55.

Salle, A., Idiart, M., and Villavicencio, A. (2016). Matrix factorization using window sampling and negative sampling for improved word representations. *arXiv preprint arXiv:1606.00819*.

Schnabel, T., Labutov, I., Mimno, D. M., and Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In *EMNLP*, pages 298–307.

Straka, M. and Straková, J. (2017). Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August. Association for Computational Linguistics.

Straková, J., Straka, M., and Hajič, J. (2014). Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Baltimore, Maryland, June. Association for Computational Linguistics.

Tesnière, L. (1965). Eléments de syntaxe structurale.

Tyers, F. M., i Font, H. A., Fronteddu, G., and Martín-Mor, A. (2017). Rule-based machine translation for the italian–sardinian language pair. *The Prague Bulletin of Mathematical Linguistics*, 108(1):221–232.

Washington, J., Ipasov, M., and Tyers, F. (2012). A finite-state morphological transducer for kyrgyz. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

MIsA: Multilingual "IsA" Extraction from Corpora

Stefano Faralli[1], **Els Lefever**[2], **Simone Paolo Ponzetto**[1]
[1]Data and Web Science, University of Mannheim, University of Mannheim
{stefano, simone}@informatik.uni-mannheim.de
[2]LT3, Language and Translation Technology Team, Ghent University, Belgium
els.lefever@ugent.be

Abstract

In this paper we present Multilingual IsA (MIsA), which is a collection of hypernymy relations in five languages (i.e., English, Spanish, French, Italian and Dutch) extracted from the corresponding full Wikipedia corpus. For each language, we first established a set of existing (viz. found in literature) or newly defined lexico-syntactic patterns. Similarly to WebIsADb, the resulting resource contains hypernymy relations represented as "tuples", as well as additional information such as provenance (Gil and Groth, 2011), context of the extraction, etc. To measure the precision of the patterns, we performed a manual assessment of the quality of the extracted relations and an error analysis. In addition, we release the software developed for the extraction of the hypernym tuples.

Keywords: multilinguality, Hearst patterns, hypernym extraction, framework.

1. Introduction

Hypernymy relations represent the relationship between a generic term (hypernym) and a specific instance of it (hyponym). These relations play a key role for many Natural Language Processing (NLP) tasks, e.g. ontology learning, automatically building or extending knowledge bases, or word sense disambiguation and induction. In fact, hypernymy relations may provide the basis for the construction of more complex structures such as taxonomies, or be used as effective background knowledge for many word understanding applications.

In the past, many different methods have been developed for hypernym extraction, ranging from simple lexical patterns (Hearst, 1992; Oakes, 2005) to statistical and machine learning techniques (Dolan et al., 1993; Caraballo, 1999; Agirre et al., 2000; Ritter et al., 2009), to name a few.

Snow et al. (2004) first search sentences that contain two terms that are known to be in a taxonomic relation (term pairs are taken from WordNet (Miller et al., 1990)), then parse the sentences, and automatically learn patterns from the parse trees. Finally, they train a hypernym classifier based on these features. Lexico-syntactic patterns are generated for each sentence relating a term to its hypernym, and a dependency parser is used to represent them.

For the ontology learning task, Velardi et al. (2013) induce taxonomies from scratch by extracting hypernyms from a domain corpus and the Web. Definitional sentences such as *"lion is a dangerous animal"* (where *"animal"* is the hypernym of *"lion"*) are recognized by the Word Class Lattices classifier (Navigli and Velardi, 2010) trained on a large set of Wikipedia definitions.

Kozareva and Hovy (2010) induce a taxonomy using a particular kind of Hearst-like (Hearst, 1992) lexico-syntactic patterns, i.e. so-called Doubly Anchored Patterns (DAP). The hypernymy relation extraction consists of two phases. First, the authors bootstrap the terminology harvesting with DAP of the kind *"animals such as lions and *"*, so it is possible to discover new terms such as *"cats"*. Next, for each pair of terms in the discovered terminology, e.g. *("lions","cats")*, they automatically create a DAP^{-1} of the kind *"* such as lions and cats"* and discover new hypernyms (e.g. *"felines"*).

The above mentioned works focus on domain-specific hypernymy relations extraction and due to their need of domain constraints - a specific defined term in (Navigli and Velardi, 2010) or a seed pair (Kozareva and Hovy, 2010) - they can not be used to collect the whole set of hyponym-hypernym pairs from a large scale corpus such as the Web.

For Microsoft's *Probase* (Song et al., 2011), albeit not freely accessible, the authors used Hearst-like lexico-syntactic patterns to extract hypernymy relations from 1.68 billion web pages in Microsoft Bing's web corpus, instead of focusing on domain specific hypernymy relations. Probase's main purpose was to create a universal taxonomy containing more than 2.7 million concepts. To this end, the methods underlying Probase are able to extract approximately 25 million pairs.

The interest in the hypernymy extraction task is also illustrated by two shared tasks organised within the SemEval framework: TExEval (Taxonomy Extraction Evaluation) focused on finding hyponym-hypernym relations between a list of domain-specific English terms and subsequent taxonomy construction (Bordea et al., 2015), whereas TExEval-2 introduced a multilingual setting for this task, covering four different languages (English, Dutch, Italian and French) from domains as diverse as environment, food and science (Bordea et al., 2016).

Our MIsA is an extension of the WebIsADb framework (Seitner et al., 2016) - a publicly available database with more than 400 million English hypernymy relations extracted from the CommonCrawl web corpus - where:

1. we investigate and evaluate the performance of a collection of existing and new lexico-syntactic patterns for five languages of interest (i.e., English (EN), Spanish (ES), French (FR), Italian (IT), Dutch (NL));

2. we release a new standalone, language-independent and easy to adapt/configure extractor, which is ready to ex-

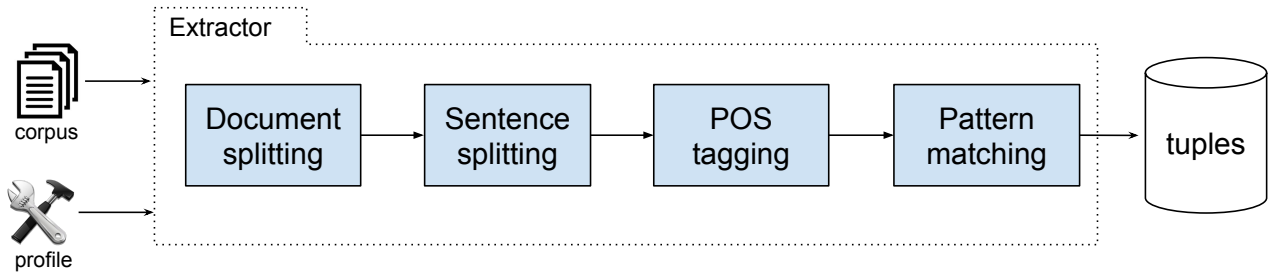


Figure 1: Pipeline for the extraction of language specific "IsA" tuples from a corpus.

Table 1: Statistics of the selected multilingual patterns.

	# patterns	excerpt
EN	74	NP_t, is a NP_c NP_c, such as NP_t NP_t, and any other NP_c
ES	33	NP_t, es un NP_c la NP_t es una de las NP_c NP_t, y otros NP_c
FR	60	NP_t, est un NP_c NP_c, comme par exemple NP_t NP_t, et autres NP_c
IT	78	NP_t, è un NP_c NP_t in confronto ad altre NP_c NP_t, e altri NP_c
NL	14	NP_t, is een NP_c NP_c is een soort van NP_t NP_t en veel andere NP_c

tract hypernymy relations from text corpora in five languages and which can be easily extended to work with other languages;

3. we release more than 32 million hypernymy relations extracted from the corresponding Wikipedia corpus for the five languages of interest.

Both datasets and tools can be downloaded from `http://web.informatik.uni-mannheim.de/misa/`

2. Multilingual patterns

As introduced in Section 1., we focused our efforts on five languages of interest (i.e., English, Spanish, French, Italian and Dutch). In Table 1 we show some statistics about our collections of Hearst-like (Hearst, 1992) lexico-syntactic patterns (the full list of selected patterns is available at `http://web.informatik.uni-mannheim.de/misa/`).

Lexico-syntactic patterns are defined as regular expressions where NP_t and NP_c are special expressions to match a noun-phrase for the definiendum and the hypernym respectively. In order to identify a noun phrase NP, similarly to WebIsADb (Seitner et al., 2016) we apply part-of-speech tagging to obtain the mapping of words to grammatical categories and identify a NP on a disjunctive/conjunctive sequence of accepted part-of-speech.

With this definition of NP we are able to intercept sequences of concepts. For example, given the English pattern "NP_c, such as NP_t" we are able to match sentences like "pure-bred dogs *such as* a bulldog or pug" (where NP_c = "pure-bred dogs" and NP_t = "bulldog"|"pug" is a sequence of concepts) and to produce multiple hypernymy relations from a single match (e.g., (bulldog, pure-bred dogs) and (pug, pure-bred dogs)).

Some patterns are directly selected or translated from literature works, such as: i) Ponzetto and Strube (2011), where *isa* patterns were used to induce a taxonomy from Wikipedia; ii) Orna-Montesinos (2011), where patterns for the term "building" were extracted on a set of specialized textbooks in the field of construction engineering; iii) Klaussner and Zhekova (2011) where the authors extract *IsA* relations from selected Wikipedia pages and iv) research describing lexico-syntactic patterns for languages other than English, such as Lefever et al. (2014) for Dutch, Séguéla (2001) for French and Galicia-Haro and Gelbukh (2014) and Ortega-Mendoza et al. (2007) for Spanish. The remaining are brand-new experimental patterns, whose selection is dictated mainly from the experience of experts in the field of NLP. In Section 5., we provide a manual assessment of the quality of the most productive patterns that were selected for the five considered languages.

3. Extraction Framework

In Figure 1 we show a diagram representing the pipeline of our extraction framework. Our pipeline input includes:

- a *corpus* in the form of a collection of flat text (UTF-8) documents written in a specific language of interest;
- a language *profile* including the following language-specific information:
  - *patterns:* the definition of lexical-syntactic patterns (as introduced in Section 2.) to be matched on the input corpus;
  - *abbreviations:* a list of abbreviations (e.g., the English abbreviation "Prof." for "Professor");
  - *pronouns:* lists of *demonstratives*, *personals*, *possessives* and *interrogative* pronouns;
  - *conjunctions:* to identify sequences of concepts (e.g., the English "or", "and").

Table 2: The total number of tuples extracted for each of the five selected languages.

	tuples	example
EN	23,386,043	. . . [*understory plants*]$_c$ such as [*bushes*]$_t$ and [*vines*]$_t$. . . . . . [*shrubs*]$_c$ including [*mountain laurel*]$_t$ and [*rhododendron*]$_t$. . . [*Mdawrush*]$_t$ is a [*municipality*]$_c$ . . .
ES	3,649,166	. . . [*Carnavalón*]$_t$, el cuales una [*ceremonia*]$_c$ acompañada de mùsica, . . . [*Paella*]$_t$ es una [*receta de cocina*]$_c$
FR	2,390,867	[*Saint-Claude*$_t$] est un [*village*]$_c$. . . [*Saint-Mars-sur-Colmont*$_t$] est une [*commune française*]$_c$. . . . . . [*Ponte alla Vittoria*$_t$] est un des [*ponts*]$_c$. . .
IT	774,964	La [*salumeria*]$_t$ è un [*negozio*]$_c$. . . [*Roma*]$_t$ è la [*capitale*]$_t$ della Repubblica Italiana,. . .
NL	1,844,644	[*Framboise Boon*]$_t$ is een [*fruitbier*]$_c$. . . [*vlees*]$_c$ zoals [*biefstuk*]$_t$, [*kip*]$_t$, [*varkenskarbonades*]$_t$, [*schapenvlees*]$_t$ en [*lamskoteletten*]$_t$. . .

where *abbreviations*, *pronouns* and *conjunctions* are devoted to language specific NP identification.

The extraction process is then divided into four main steps:

1. *Document splitting:* in the initial phase the corpus is structured as a collection of indexed documents and titles are collected to later identify all the extraction's provenances;
2. *Sentence splitting:* since the context of the extraction is a single sentence, we split each document in separated one-line sentences;
3. *POS tagging:* each sentence is processed with a POS tagger (we use the Stanford POS-tagger (Toutanova et al., 2003) for the EN, FR and ES corpora and TreeTagger (Schmid, 1994) for the IT and NL corpora) to allow identification of NPs in the next step;
4. *Pattern matching:* in this final step we find all matches between the lexico-syntactic patterns and the POS-tagged input sentence.

The output of our pipeline consists of a collection of tuples representing each pattern match, including (1) the pattern that is matched, (2) a pair of concept sequences (T, C) where T is the sequence of definiendum NPs and C is the sequence of hypernymy NPs, (3) the provenance (i.e., the corpus and document identifier), (4) the contextual matching sentence and (5) the POS-tag sequence for the sentence.

4. Multilingual Resource

We applied the pipeline described in Section 3. on five language specific Wikipedia dumps (latest available dumps accessed on 2 May 2017) with the corresponding language profiles including the patterns described in Section 2.. In Table 2 we show, for each language, the resulting number of generated tuples.

5. Evaluation

Similar to the WebIsADb, our aim is to provide with MIsA both a tool and a resource to favour investigations and applications with lexico-syntatic patterns. To this end, we provide in Section 5. an assessment of the precision of the extracted hypernymy pairs and a qualitative error analysis.

5.1. Manual assessment of precision

We show the results of our manual assessment of the quality of the extracted hypernymy relations for the five most productive patterns per language (30 patterns in total). For each pattern, a random sample of 100 extracted hypernym tuples was manually verified by the annotators, who assigned one of the following three labels to each matched hypernym pattern:

1. *Correct*: correctly extracted hypernym tuple.
2. *Partially correct*: the extracted hypernym tuple is not complete (missing hyponyms, part of the instance/class is missing, e.g. *Operation Little Switch was an exchange of sick and wounded prisoners* resulting in (Operation Little Switch, exchange)) or is too context-dependent or vague (e.g. *John Laurence is a friend* resulting in (John Laurence, friend)).
3. *Not Correct*: wrongly extracted hypernym tuple.

During the annotation process, the annotators also had access to the accompanying information as described in Section 3. Table 3 shows an example of the input the annotators were provided with, containing amongst others the hypernym tuple, the original input sentence containing the hypernym pair, the POS-tag sequence of the class and instance terms, etc.

We show in Table 4 the resulting estimated precision for the most productive patterns across the five languages of interest. As expected, when combining the correct and the partially correct matches, we observe similar pattern behaviors as in the WebIsADb (Seitner et al., 2016).

5.2. Error Analysis

The aim of this work is a robust hypernym extraction system, which can be easily deployed on very large (web) corpora. We implemented patterns for 5 languages, aiming for a high recall, sometimes at the cost of precision. As we make the code freely available to the community, researchers can easily adapt the code to add patterns or improve the precision of the current implementation.

Table 3: Example of the input provided for the manual labeling of the pattern quality.

Information	**example**
ID	Mastertapes.txt
Pattern	EN_p8a: NP_t, is a NP_c
NP_t	Mastertapes
NP_t POS	NN
NP_c	BBC Radio 4 programme
NP_c POS	NN NN CD NN
Input	Mastertapes is a BBC Radio 4 programme, presented by John Wilson, which discusses the making of significant rock albums.
POS	Mastertapes/NN, is/VBZ, a/DT, BBC/NN, Radio/NN, 4/CD, programme/NN, presented/VBN, by/IN, John/NN, Wilson,/NN, which/WDT, discusses/VBZ, the/DT, making/NN, of/IN, significant/JJ, rock/NN, albums/IN

Table 4: Number of matches and estimated precision for the most productive patterns across languages

	Pattern	**#matches**	**Precision**	**Partial match**
EN	NP_t, is a NP_c	1,855,931	48%	25%
	NP_c, such as NP_t	90,7430	50%	21%
	NP_c, including NP_t	894,625	34%	15%
	NP_t, was ((a)l(an)) NP_c	641,385	31%	25%
	examples of NP_c, are NP_t	501,122	6%	0%
ES	NP_t es una NP_c	1,413,307	59%	9%
	NP_t era una NP_c	311,044	34%	12%
	NP_t era un NP_c	311,010	29%	15%
	NP_t eran una NP_c	80,163	27%	25%
	NP_t eran un NP_c	79,891	20%	35%
FR	NP_t est un NP_c	600,613	52%	18%
	NP_t est une NP_c	444,456	38%	17%
	NP_t est ((un)l(une)) des NP_c	168,006	18%	13%
	NP_t nommé NP_c	104,938	10%	6%
	NP_t sont ((le)l(la)l(les)) NP_c	82,141	1%	1%
IT	NP_t è un NP_c	331,352	65%	24%
	NP_t è una NP_c	147,759	62%	24%
	NP_c in particolare NP_t	38,107	24%	18%
	NP_t (elo|(ed)|(oppure)) altri NP_c	22,029	59%	9%
	NP_t era un NP_c	21,987	48%	15%
NL	NP_t, is een NP_c	1,182,272	61%	13%
	NP_c, (zo)?als NP_t	634,383	10%	12%
	NP_c, (en|of) (veel)? ander(e)? NP_t	169,116	9%	3%
	NP_t, zijn een NP_c	24545	4%	14%
	NP_c, (inclusief|specifiek) NP_t	3,258	13%	12%

A manual analysis of the extracted hypernym tuples revealed several possibilities to improve the current implementation. One obvious way to increase the precision of the system is to add additional syntactic constraints to the regular expressions.
Two recurrent phenomena causing overgeneration of the patterns appeared to be:

1. hypernyms/hyponyms extracted from **prepositional phrases** (e.g. *Houston has had notable sports teams in its history, including Phi Slama Jama*, where (Phi Slama Jama, history) is extracted by the pattern "NP_c, including NP_t");

2. **wrong part-of-speech** tagging (e.g. *Abuse of positive leverage can also lead to coercion, including bribery and blackmail*, where *coercion* is tagged as a verb, which prevents the hypernym pattern "NP_c, including NP_t" from matching the input sentence).

Finally, it is also important to mention that the current implementation of the system does not take into account agreement between the hypernym and hyponym terms. As an example, we can cite the tuple (*Basilica San Marco*, *churches*), where the instance is a singular entity, whereas the class refers to a plural noun. A simple solution for these agreement problems could be the lemmatisation of all extracted hypernym and hyponym terms.

6. Conclusion

We presented *MIsA*, a multilingual collection of hypernymy relations extracted from five language specific Wikipedia dumps. The resource is created by means of language-specific sets of Hearst-like patterns that were collected from literature or dictated by experimental needs. In fact, the aims of this research are: i) to experiment with multilingual Hearst-like patterns on large corpora; ii) to provide an evaluation of both the quality and the limits of lexico-syntactic pattern approaches; iii) to release a versatile tool to let other researchers extend (with minimal effort) our case study to other languages or to different selections of language specific patterns. Future work will include the extension of the multilingual setting to larger corpora (e.g., Web-scale corpora) and to other languages.

Acknowledgements

This work was partially funded by the Deutsche Forschungsgemeinschaft within the JOIN-T project (research grant PO 1900/1-1).

7. Bibliographical References

Agirre, E., Ansa, O., Hovy, E. H., and Martínez, D. (2000). Enriching very large ontologies using the WWW. In *Proceedings of the ECAI Workshop on Ontology Learning*.

Bordea, G., Buitelaar, P., Faralli, S., and Navigli, R. (2015). Semeval-2015 task 17: Taxonomy extraction evaluation (texeval). In *Proceedings of SemEval-15*.

Bordea, G., Lefever, E., and Buitelaar, P. (2016). Semeval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *Proceedings of SemEval-16*.

Caraballo, S. A. (1999). Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of ACL-99*, pages 120–126.

Dolan, W., Vanderwende, L., and Richardson, S. D. (1993). Automatically deriving structured knowledge bases from on-line dictionaries. In *Proceedings of PACLING-93*, pages 5–14.

Galicia-Haro, S. and Gelbukh, A. (2014). Extraction of semantic relations from opinion reviews in spanish. In *Proceedings of MICAI-14*, pages 175–190.

Gil, Y. and Groth, P. T. (2011). Using provenance in the semantic web. *J. Web Sem.*, 9(2):147–148.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of COLING-92*, pages 539–545.

Klaussner, C. and Zhekova, D. (2011). Pattern-based ontology construction from selected wikipedia pages. In *Proceedings of the RANLP-11 SRW*.

Kozareva, Z. and Hovy, E. (2010). A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of EMNLP-10*, pages 1110–1118.

Lefever, E., Van de Kauter, M., and Hoste, V. (2014). Hypoterm detection of hypernym relations between domain-specific terms in Dutch and English. *Terminology*, 20(2):250–278.

Miller, G. A., Beckwith, R., Fellbaum, C. D., Gross, D., and Miller, K. (1990). WordNet: an online lexical database. *International Journal of Lexicography*, 3(4):235–244.

Navigli, R. and Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of ACL-10*, pages 1318–1327.

Oakes, M. P. (2005). Using Hearst's rules for the automatic acquisition of hyponyms for mining a pharmaceutical corpus. In *Proceedings of the RANLP Text Mining Workshop*, pages 63–67.

Orna-Montesinos, C. (2011). Words and patterns: lexicogrammatical patterns and semantic relations in domain-specific discourses. *Revista Alicantina de Estudios Ingleses*, 24:213–233.

Ortega-Mendoza, R., Villaseñor Pineda, L., and Montes-y Gómez, M. (2007). Using lexical patterns for extracting hyponyms from the web. In *Proceedings of MICAI-07*, pages 904–911.

Ponzetto, S. P. and Strube, M. (2011). Taxonomy induction based on a collaboratively built knowledge repository. *Artificial Intelligence*, 175(9-10):1737–1756.

Ritter, A., Soderland, S., and Etzioni, O. (2009). What is this, anyway: Automatic hypernym discovery. In *Proceedings of the 2009 AAAI Spring Symposium on Learning by Reading and Learning to Read*, pages 88–93.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49.

Séguéla, P. (2001). *Construction de modèles de connaissances par analyse linguistique de relations lexicales dans les documents techniques*. Ph.D. thesis, Université Toulouse III Paul Sabatier.

Seitner, J., Bizer, C., Eckert, K., Faralli, S., Meusel, R., Paulheim, H., and Ponzetto, S. P. (2016). A large database of hypernymy relations extracted from the web. In *Proceedings of LREC-16*.

Snow, R., Jurafsky, D., and Ng, A. Y. (2004). Learning syntactic patterns for automatic hypernym discovery. In *Proceedings of NIPS-04*, pages 1297–1304.

Song, Y., Wang, H., Wang, Z., Li, H., and Chen, W. (2011). Short text conceptualization using a probabilistic knowledgebase. In *Proceedings of IJCAI-11*, pages 2330–2336.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL-03*, pages 173–180.

Velardi, P., Faralli, S., and Navigli, R. (2013). OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction. *Computational Linguistics*, 39(3):665–707.

Biomedical term normalization of EHRs with UMLS

Naiara Perez-Miguel, Montse Cuadros, German Rigau
HSLT group at Vicomtech, IXA group at UPV/EHU
Donostia/San Sebastián, Spain
{nperez,mcuadros}@vicomtech.org, german.rigau@ehu.es

Abstract

This paper presents a novel prototype for biomedical term normalization of electronic health record excerpts with the Unified Medical Language System (UMLS) Metathesaurus, a large, multi-lingual compendium of biomedical and health-related terminologies. Despite the prototype being multilingual and cross-lingual by design, we first focus on processing clinical text in Spanish because there is no existing tool for this language and for this specific purpose. The tool is based on Apache Lucene™ to index the Metathesaurus and generate mapping candidates from input text. It uses the IXA pipeline for basic language processing and resolves lexical ambiguities with the UKB toolkit. It has been evaluated by measuring its agreement with MetaMap –a mature software to discover UMLS concepts in English texts– in two English-Spanish parallel corpora. In addition, we present a web-based interface for the tool.

Keywords: term normalization, UMLS, information extraction, biomedical text

1. Introduction

Biomedical text mining technologies are becoming a key tool for the efficient exploitation of information contained in unstructured data repositories, including scientific literature, Electronic Health Records (EHRs), patents, biobank metadata, clinical trials and social media. Natural Language Processing (NLP) and specifically Information Extraction (IE) tools, such as term normalization tools, can facilitate knowledge discovery, exchange, and reuse by finding relevant terms and semantic structure in those texts. This paper presents a preliminary application that enriches EHRs with links to the Unified Medical Language System (UMLS)[1], a multilingual repository of biomedical terminologies. The tool is multilingual and cross-lingual by design, but we first focus on Spanish EHR processing because there is no existing tool for this language and for this specific purpose. We propose a sequential pipeline that retrieves mapping candidates from an indexed UMLS Metathesaurus, uses the IXA pipeline (Agerri et al., 2014) for basic language processing and UKB (Agirre and Soroa, 2009) for word sense disambiguation (WSD). In addition to the pipeline itself, this paper also presents a demonstration interface for the tool that will be available on-line[2].

2. Related Work

Biomedical term normalization is a long-established research field in English-speaking countries where terminological resources and basic-processing tools for the biomedical domain and this language have been available for decades. Thus, there already exist several mature applications that are being effectively exploited for different purposes and by different organizations as of today. In what follows, we present some of the better-known applications.

MetaMap (Aronson, 2001; Aronson, 2006) enriches biomedical text with links to the UMLS Metathesaurus. It is "knowledge intensive" as it relies heavily on the SPECIALIST Lexicon, a large syntactic lexicon of biomedical and general English. Meystre and Haug (2005) evaluated MetaMap with 160 clinical documents of diverse nature (radiology reports, exam reports, and so on). MetaMap's results were compared to annotations by 8 physicians; the reported precision and recall for detecting a set of 80 diseases were 76% and 74%.

MedLEE (Friedman et al., 1994; Friedman, 2000) is one of the earliest English term mapping systems for the clinical domain, alongside MetaMap. It exploits several knowledge sources of their own. In Friedman et al. (1994), MedLEE is evaluated by measuring its precision and recall at detecting the presence of four diseases in a collection of health records; the results were 70% recall and 87% precision.

NCBO Annotator is a web service provided by the National Center for Biomedical Ontology (NCBO) that annotates textual data with terms from the UMLS and BioPortal ontologies. The details of how MGREP —the concept recognition tool— works are limited to the conference poster by Dai et al. (2008). Shah et al. (2009) experimented with the task of large-scale indexing of online biomedical resources: MetaMap recognized more concepts but with a lower precision than MGREP, and MGREP turned to be faster than MetaMap.

cTakes (Savova et al., 2010) is a comprehensive platform for performing many clinical information extraction tasks, including enriching text with terms from the UMLS Metathesaurus. cTakes does dictionary lookup to recognize and identify clinical entities. They report that mapping to the UMLS accuracy is high for exact span matches.

As for Spanish, there have been a few attempts to process clinical free text in this language. Next, we present some of these attempts that are relevant to the work presented in this paper.

GALEN (Carrero et al., 2008a; Carrero et al., 2008b) proposed a "Spanish MetaMap" that combines machine translation techniques with the use of MetaMap. Unfortunately, they did not apply this system to any task, so performance scores cannot be reported.

The system by **Castro et al. (2010)** aims at retrieving SNOMED CT® concepts based on an input phrase (SNOMED CT® is the most complete biomedical terminology, and it is included in the UMLS). Term normalization is done by querying an Apache Lucene™ index

[1] https://www.nlm.nih.gov/research/umls/

[2] http://demos-v2.vicomtech.org/umlsmapper/, user:vicomtech, password:umlsmapper

of SNOMED CT® and re-ranking the candidates with a function of their own. In order to evaluate the performance of this system, they obtained a set of 100 health records manually tagged by two specialists with "disruptions" or "procedures" concepts in SNOMED CT®. For the exact-matching assessment, they report an average precision of 39% and a recall of 0.65%. Partial matching increases precision to 71%, but recall is still 0.75%.

FreelingMed (Oronoz et al., 2013) uses the Freeling analyzer (Carreras et al., 2004) and extend its linguistic data with various knowledge sources including SNOMED CT®, a list of medical abbreviations (Yetano, 2003), Bot PLUS, and ICD-9. The actual task that the tool is meant to perform is term recognition, not term normalization. The system was assessed against a Gold Standard of 100 health records annotated with drug names, diseases and substances, counting as true positives approximate matches. The final result was 0.90 per the F-measure.

As can be seen, none of the tools presented offers a complete pipeline to perform biomedical term normalization in Spanish clinical text with the UMLS.

3. Pipeline Description

The overall architecture for the prototype is schematized in Figure 1. It consists of components executed in sequence, some of which use a knowledge base, our adaptation of the UMLS Metathesaurus. This section provides a description of the knowledge base and the overall workflow. We also report a first approximation for assessing the performance of the prototype.

3.1. The Knowledge Base

The knowledge base of the prototype has been derived from the 2016AA Full Release UMLS Metathesaurus. It gathers 196 terminology sources in 25 different languages, amounting to 3,250,226 concepts and 10,586,865 unique terms in total. For this prototype we focus on the subset of sources in Spanish, which consists of 451,297 concepts and 1,255,377 unique terms. Table 1 shows the amount of concepts and unique terms per source available in Spanish –7 out of 196 –, both in their English and Spanish versions. The table reveals that the Spanish versions have much less conceptual and lexical coverage.

To build the knowledge base for our prototype, we use specifically Metathesaurus terms that *a)* are in Spanish, *b)* do not belong to LOINC® [3], *c)* are shorter than 15 tokens, *d)* are not obsolete or suppressible, *e)* do not consist of a single character, *f)* do not consist of just numbers, and *g)* do not consist of only stopwords. We consider 303 common Spanish words except "no", "sin" and "con" (*no*, *without*, and *with*, respectively) because they may alter the polarity of expressions, which is essential to be processed in this domain (Ceusters et al., 2007). Applying these filters, we are left with 352,075 concepts and 546,309 unique terms.

The application proposed needs the knowledge base in three formats:

[3] LOINC® descriptors look typically like "especie de Thrichomonas:número areico:punto en el tiempo:sedimento urinario:cuantitativo:microscopia.de luz.campo de gran aumento", so they are not suited for the task at hand.

	English		Spanish	
	Concepts	**Terms**	**Concepts**	**Terms**
All sources	3,250,226	10,586,865	451,297	1,255,377
CPT®	39,152	61,923	2,720	2,484
ICPC	748	1,017	722	688
LOINC®	157,645	390,425	48,609	48,631
MedDRA	51,961	78,528	45,488	61,103
MeSH®	359,116	837,305	35,970	64,804
SCT®	357,448	1,115,865	306,539	746,600
WHO-ART	3,175	3,831	2,566	3,102

Table 1: UMLS 2016AA Full Release Metathesaurus counts for English and Spanish subsets of sources available in Spanish

The UMLS index. We use Apache Lucene™ in order to be able to make fast searches in our subset of the UMLS Metathesaurus. An index has been created where each entry represents a term of the subset and contains the following information: the term itself, a normalized version of the term, the concept identifier(s) it is related to, and its source(s). The normalized string is obtained after erasing spurious parenthetical content, punctuation, and stopwords. The list of the spurious parenthetical content has been curated manually after studying the Metathesaurus. As for the stopwords, they are the same 303 used to filter the UMLS Spanish subset.

The UKB Knowledge Graph. This graph contains all the relations in the 2016AA Metathesaurus whose origin and target concepts are both included in our UMLS subset. For each relation, it encodes the source and target concepts, the direction of the relation, and its type. Overall, the graph consists of 352,075 vertices and 8,381,482 edges. All the concepts indexed participate in one relation at least.

The UKB Dictionary. It maps the terms in our UMLS subset to their respective concept or concepts, in the case of those that are ambiguous.

3.2. Overview of the Workflow

Let us describe the proposed processing flow by means of an example; take the input text to be the following:

"acude por lesión grave en rodilla dcha"
[patient] comes due to serious injury in rt knee

First, the text received is analyzed in search of **abbreviations and acronyms**, which are expanded to their corresponding full expressions. The tool employed to identify abbreviation- or acronym-like elements in texts (Montoya, 2017) exploits a set of rules and a 2,312-item long list of abbreviation/acronym and corresponding expansions, curated after manual annotations by health care professionals. In our example, this step would produce

"acude por lesión grave en rodilla derecha"
[patient] comes due to serious injury in right knee

Next, the system does basic **linguistic processing** with the IXA pipeline (Agerri et al., 2014): tokenization, part-of-speech tagging, and constituent parsing. The linguistic information obtained serves as basis to perform **boundary**

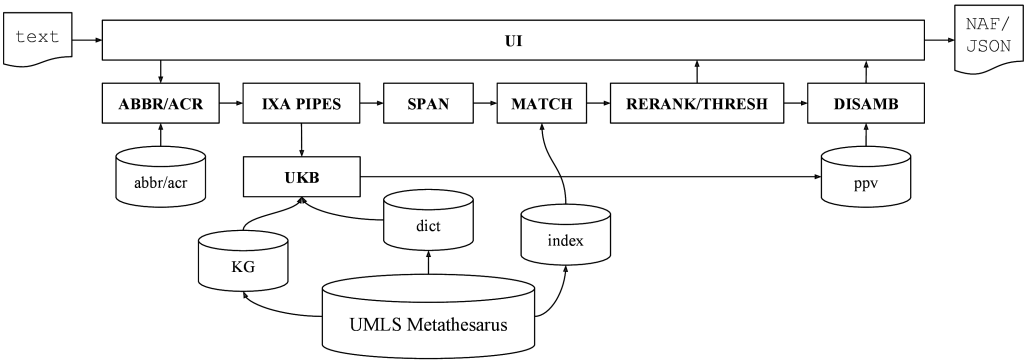


Figure 1: Architecture of the pipeline

detection, that is, to recognize in the text spans or sequences of tokens that are likely to be mapped to a medical concept. In order to maximize recall, we explore two methods: extracting n-grams of varying sizes, and extracting nominal phrases based on a simple set of rules that uses the linguistic information, allowing for discontinuous spans.
After extracting textual spans, the system attempts to find mapping candidates of the Metathesaurus terms indexed by lexical proximity. This is the role of the **matching** module. It queries the index with the spans, obtaining as a result of each query a collection of Metahesaurus terms, which are in turn related to one or more concepts and a relevance score.
The **reranking** module assigns new scores to the candidates using a function other than the one provided by Lucene. We explore two such functions: the one by Castro et al. (2010), and the one by Aronson (2001) implemented in MetaMap. Furthermore, a **threshold** can be applied to discard candidates with low scores.
Matching, reranking and thresholding are not done with all the spans detected; the **mapping candidate generation algorithm** prefers longer matches:

1. the system orders the spans by subsumption creating oriented trees as depicted in Figure 2;
2. then, it queries the index with the root of the tree and its direct children, reranks the results and applies a threshold;
3. if any of the children obtains a better result than their parent, then the results retrieved for the parent span are ruled out, and the algorithm is repeated recurrently for the children nodes;
4. if a parent has a result better than any of its children's, the results retrieved for the parent are accepted as candidates and the system does not attempt to map any of its descendants.

Following this algorithm, textual spans that overlap can be annotated with different concepts, but not spans that are nested within a bigger one.
At this point, a span can have zero, one or multiple mapping candidates. Then,

a) if no candidate is available, one must conclude that either the span in question was never a term in the first place, or that it is a term but does not have an explicit or convincing enough mapping to a UMLS Metathesaurus entry indexed;

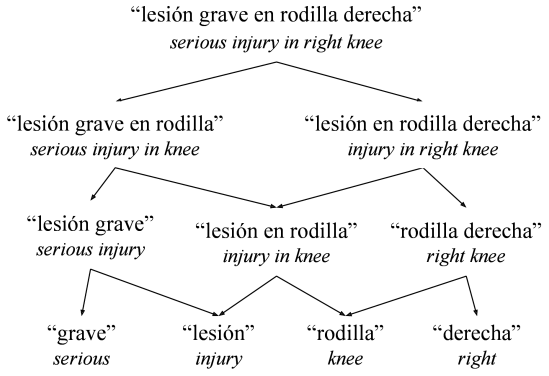


Figure 2: Oriented tree of detected spans

b) if one candidate is available, the system takes it as a final mapping for the span;

c) if more than one is available, the system takes as a final mapping the one scored highest; and

d) if more than one candidate become tied in first position, the system needs to carry out a **disambiguation** step in order to choose the correct mapping. This process is performed by the UKB module.

The algorithm behind UKB is *Personalized PageRank* (Haveliwala, 2002). Agirre et al. (2010) and Stevenson et al. (2012) prove that UMLS's conceptual graph can be used as a knowledge base for WSD. Here we implement a little variation of their approach. The context to initialize the Knowledge Graph consists of the tokens in the text; the system is able to provide this information as early as the basic linguistic processing is done. When the disambiguation module is required, it just needs to choose the mapping candidate with highest activation in the Personalized PageRank Vector.
The pipeline ends by gathering the final mappings and displaying them to the user.

3.3. Evaluation

At the moment there is no corpus available in Spanish annotated with UMLS concepts that can serve as Gold Standard to evaluate this application. For this reason, we propose the following evaluation framework as a first approximation to measure the performance of the tool proposed.

3.3.1. Design

Having created/obtained two English-Spanish parallel corpora of biomedical text, the English documents have been annotated with MetaMap and the Spanish ones with the prototype proposed; then, the agreement between the systems has been measured by means of Cohen's Kappa (Cohen, 1960). Crucially, MetaMap'a knowledge source has been reduced so that both systems can annotate only the same 352.075 concepts, in order to make the annotations comparable. MetaMap's mapping strategy has also been configured so that it prefers longer matches, as the prototype does.

Corpora. One of the corpora is a manually revised subset of the Scielo Corpus (Neves et al., 2016), resulting in 1,895 titles and abstracts of scientific literature. The other corpus consists of 10 EHR texts originally in Spanish and their English translations, plus 8 EHR texts originally in English and their translations to Spanish. Table 2 shows the sizes of the corpora.

	Scielo		EHR	
	es	**en**	**es**	**en**
# documents	1,895	1,895	18	18
# words	26,490	23,374	23,311	21,093

Table 2: Corpora used for evaluation

Metric. Cohen's Kappa k is defined as follows:

$$k = \frac{p_o - p_e}{1 - p_e} \quad (1)$$

where p_o is the proportion of units in which the annotators agree and p_e is the proportion of units for which agreement is expected by chance. The units are the 352.075 concepts in the index; MetaMap and our system agree only when both say that a given concept is present in the input document.

There is no universally accepted interpretation of Cohen's kappa as to what is considered high or low agreement. (Landis and Koch, 1977) proposed the following, which is widely cited, but has no evidential grounding:

$k < 0.00$	No agreement
$0.00 \leq k \leq 0.20$	Slight agreement
$0.21 \leq k \leq 0.40$	Fair agreement
$0.41 \leq k \leq 0.60$	Moderate agreement
$0.61 \leq k \leq 0.80$	Substantial agreement
$0.81 \leq k \leq 1.00$	Almost perfect agreement

Variables. The experiment has been carried out with the following prototype settings:

- Boundary detection: *ngram* or *phrase*.
- Re-ranking function: Lucene (*L*), Castro et al. (2010) (*C*), or (Aronson, 2001) (*A*); *L* is simply using the scores given by Lucene, that is, not re-ranking at all.
- Disambiguation: *UKB* or random disambiguation as baseline (*rand*).

The results reported can only be taken as hints for the differences in performance between the possible configurations of the modules. Therefore, a qualitative error and disagreement analysis has been carried out in an attempt to elucidate these issues.

	WSD	**score**	**ngram**	**phrase**
Scielo	rand	$L_{(.0)}$	0.323 ± 0.006	0.304 ± 0.006
		$A_{(.5)}$	0.331 ± 0.006	0.308 ± 0.006
		$C_{(.7)}$	0.398 ± 0.006	0.372 ± 0.006
	UKB	$L_{(.0)}$	0.343 ± 0.006	0.328 ± 0.005
		$A_{(.5)}$	0.349 ± 0.006	0.330 ± 0.006
		$C_{(.7)}$	$\mathbf{0.412 \pm 0.006}$	0.387 ± 0.006
EHR	rand	$L_{(.0)}$	0.286 ± 0.007	0.266 ± 0.007
		$A_{(.5)}$	0.330 ± 0.008	0.316 ± 0.008
		$C_{(.7)}$	0.403 ± 0.008	0.389 ± 0.008
	UKB	$L_{(.0)}$	0.321 ± 0.007	0.306 ± 0.007
		$A_{(.5)}$	0.365 ± 0.008	0.354 ± 0.008
		$C_{(.7)}$	$\mathbf{0.432 \pm 0.008}$	0.414 ± 0.008

Table 3: Agreement between MetaMap and the prototype

3.3.2. Results

Results show that our prototype can reach moderate agreement with MetaMap. They suggest that the scoring function proposed in Castro et al. (2010) makes the results of our prototype substantially more similar to the ones from MetaMap than the other two functions. Using n-grams to create textual spans yields always a slightly better agreement with MetaMap. Furthermore, agreement with MetaMap also improves when using UKB to perform disambiguation compared to the baseline proposed.

A manual analysis of the results has shown that the main source of **disagreement** is, of course, the fact that MetaMap and our application annotate different texts –parallel texts; furthermore, they use different sources of knowledge, in spite of the efforts to make them as similar as possible by limiting the knowledge base of MetaMap to contain only the concepts indexed for our system. To illustrate these facts, let us consider the following input:

en: "Should we rule out congenital anesplenia?"
es: "¿Debemos descartar una asplenia congénita?"

MetaMap and our best system (*UKB+C+ngram*) find mappings for these spans:

MetaMap: “rule”, “out”, “congenital”
Ours: “descartar”, “asplenia congénita”

To begin with, “rule out” is translated as “descartar” in Spanish. When MetaMap creates —in this case, incorrect— annotations for “rule” and “out”, it is impossible to produce the same annotations, since the Spanish “descartar” does not have the meaning of any of the two English words separately. We can also see that MetaMap does not recognize the concept “congenital anesplenia”. As it happens, MetaMap’s knowledge base contains “congenital asplenia” but not “congenital anesplenia”, and so it does not annotate it. Of course, problems like these occur in both directions.

As for the errors that our prototype commits, many **false positive errors** are produced due to the fact that the Metathesaurus does not capture all the possible meanings of the terms it contains; because candidates are scored simply by means of lexical similarity, the system will annotate a term that is similar enough to an entry in the Metathesaurus even if they denote different concepts. Let us illustrate the problem: the term “clavo” in Spanish has at least three meanings: *a)* clove (a spice), *b)* nail or rod (a metallic object), and *c)* corn of toe (a disease). However, the term “clavo” is only related to sense *a)* and *c)* in the Metathesaurus. This is not to say that sense *b)* is not represented in the Spanish subset, but that it is not represented as “clavo”. As a consequence, whenever an input text contains “clavo” (and it does not form a bigger concept with its surrounding words), it will be annotated as being a disease or a spice, even if it is neither of the two.

Another important source of false positives is the over-generation of spans: both n-gram-based and phrase-based detection generate incorrect spans that eventually can also be annotated. The n-gram strategy clearly generates spans that are not meant to form syntactic units, and thus neither intended meaning units. For example, in the text fragment “[...] arteria torácica en radiografía [...]” (*chest artery in x-ray*), the bigram [torácica, radiografía] would form a span that would, in turn, trigger mapping candidates consisting of concepts referring to chest x-ray, which is not actually mentioned in the text. Although the phrase-based strategy was meant to overcome this problem by leveraging syntactic information, the fact that it allows for discontinuous spans also produces over-generation sometimes, especially when coordination and/or enumeration are involved.

Regarding **false negatives**, there are two main reasons for our system to miss a biomedical concept: on the one hand, it can happen that the concept is not captured in the Metathesaurus at all; on the other hand, it could be that the concept is captured but not as expressed in the text, be it because it is misspelled, abbreviated in a way that the Metathesaurus does not contemplate, or formulated in any other non-standard way. That is, false negatives are caused by a poor lack of the Metathesaurus and the lexical variability of clinical narrative. MetaMap relies on a powerful tool to deal with variability –the SPECIALIST Lexicon; we do not address variability but for a closed list of abbreviations. As a consequence, our system is much more likely to produce this type of error, in any of its possible configurations. Additionally, phrase-based span detection is another source of false negatives, as it can miss noun phrases due to errors in the lower-level processing of the input texts: if it misses a noun phrase and the noun phrase happens to be a relevant term, the term is not annotated.

4. Demo

A web-based demonstrator has been developed to allow users to introduce a text of their choosing and visualize the mappings produced by the application in an interactive user interface. The client side of the demonstrator has been developed in Angular2[4]. It is a webservice that communicates with the application via HTTP. In order to enrich the demonstrator with information about the concepts that have been mapped, the demonstrator also communicates with an additional webservice that provides an API to query the UMLS Metathesaurus and the Semantic Network, which is a hierarchical classification of the concepts in the UMLS Metathesaurus, and a source of the Metathesaurus itself.

In the **home page**, users can introduce their text and configure the application. Users can also choose which semantic types of the Semantic Network of the UMLS they are interested in; the bottom part of the page contains the whole Semantic Network in the form of a tree that can be expanded and collapsed by the users in order to select the semantic types of the concepts to be used by the mapping procedure.

The result page is divided into three columns. An example is shown in Figure 3. The middle column contains the submitted text; annotations are marked in the text with different colors, depending on the semantic type of the concepts. On the left side is a list of the found concepts’ semantic types. By clicking on any of the semantic types, one can see below the actual concepts or annotations, represented by their preferred names. The example given in Figure 3 shows, for instance, that two signs or symptoms haven been found in the text (i.e. “tos” –*cough*– and “disnea” –*dyspnea*–). When the user clicks on one of the concept names, information about that concept appears on the right side of the page: preferred name, semantic types, a definition, and so on. Moreover, the user can also see hypernym and hyponym relations, and navigate through the concepts within this hierarchy. In the case of Figure 3, the user clicked on the concept “Asperguillus” —which is mentioned twice in the last paragraph of the processed text—. The figure shows that this concept, with identifier C0004034 in the UMLS, has 6 terms related to it in the Spanish extension of SNOMED-CT (SCTSPA) and one more in the Spanish translation of Medical Subject Headings (MSHSPA). It also shows, among other information, that “Aspergillus” is a “Ascomycota”, and that “Aspergillus clavatus”, “Aspergillus fumigatus” and “Aspergillus flavus” are all “Aspergillus”.

5. Conclusions

We have presented a prototype to perform biomedical term normalization in clinical texts with the UMLS Metathesaurus. The tool performs abbreviation/acronym expansion and WSD. Mapping candidate generation is done by

[4] `https://angular.io/`

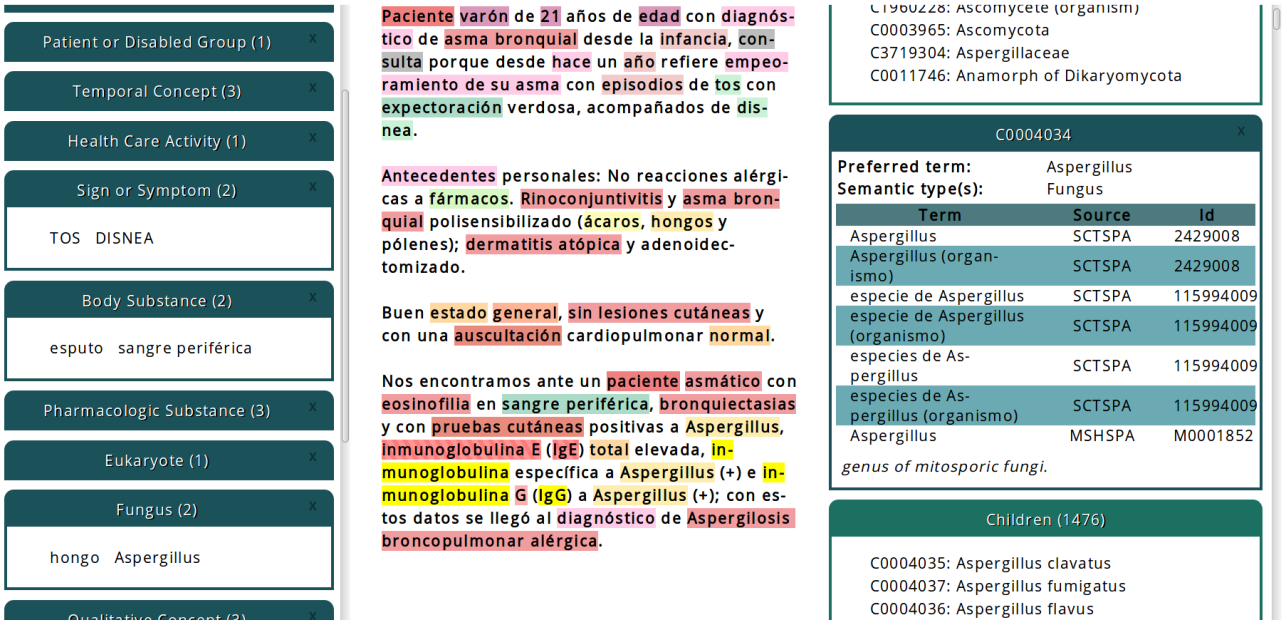


Figure 3: Results page of the demo website

querying an index of the Metathesaurus with spans of the input text. As a preliminary evaluation, agreement with MetaMap has been measured in two parallel corpora; our best system has reached moderate agreement with MetaMap. We have also presented a web-based user interface for the prototype. As future work, we plan to assess the tool with texts in languages other than Spanish. We must also address misspellings, morphological variants and synonyms of the terms covered in the UMLS. Furthermore, other evaluation frameworks for evaluation should be designed, in order to better understand the shortages that the current version of the prototype has and how the tool could be improved.

6. Acknowledgements

This work has been funded by the Department of Economic Development and Infrastructure of the Basque Government under the project BERBAOLA (KK-2017/00043), and by the Spanish Ministry of Economy and Competitiveness (MINECO/FEDER, UE) under the projects CROSSTEXT (TIN2015-72646-EXP) and TUNER (TIN2015-65308-C5-1-R).

7. Bibliographical References

Agerri, R., Bermudez, J., and Rigau, G. (2014). IXA pipeline: Efficient and Ready to Use Multilingual NLP tools. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 3823–3828. European Language Resources Association (ELRA).

Agirre, E. and Soroa, A. (2009). Personalizing PageRank for Word Sense Disambiguation. *Proceedings of the 12th Conference of the European Chapter of the ACL*, pages 33–41.

Agirre, E., Soroa, A., and Stevenson, M. (2010). Graph-based word sense disambiguation of biomedical documents. *Bioinformatics*, 26(22):2889–2896.

Aronson, A. R. (2001). Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program. In *Proceedings of the AMIA Symposium*, pages 17–21. American Medical Informatics Association.

Aronson, A. R. (2006). MetaMap: Mapping Text to the UMLS Metathesaurus. *Bethesda, MD: NLM, NIH, DHHS*.

Carreras, X., Chao, I., Padró, L., and Padró, M. (2004). Freeling: An Open-Source Suite of Language Analyzers. *Proceedings of the 4th Language Resources and Evaluation Conference (LREC 2004)*, 4:239–242.

Carrero, F. M., Cortizo, J. C., and Gómez, J. M. (2008a). Building a Spanish MMTx by Using Automatic Translation and Biomedical Ontologies. In *Intelligent Data Engineering and Automated Learning – IDEAL 2008*, pages 346–353. Springer.

Carrero, F. M., Cortizo, J. C., Gómez, J. M., and de Buenaga, M. (2008b). In the development of a Spanish Metamap. In *Proceedings of the 17th ACM conference on Information and knowledge mining - CIKM '08*, pages 1465–1466. ACM Press.

Castro, E., Iglesias, A., Martínez, P., and Castaño, L. (2010). Automatic Identification of Biomedical Concepts in Spanish Language Unstructured Clinical Texts. In *Proceedings of the 1st ACM International Health Informatics Symposium (IHI'10)*, pages 751–757. ACM.

Ceusters, W., Elkin, P., and Smith, B. (2007). Negative Findings in Electronic Health Records and Biomedical Ontologies: A Realist Approach. *International Journal of Medical Informatics*, 76(3):326–333.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Dai, M., Shah, N. H., Xuan, W., Musen, M. A., Watson, S. J., Athey, B. D., Meng, F., et al. (2008). An efficient solution for mapping free text to ontology terms. *AMIA Summit on Translational Bioinformatics*, 21.

Friedman, C., Alderson, P. O., Austin, J. H. M., Cimino, J. J., and Johnson, S. B. (1994). A general natural-language text processor for clinical radiology. *Journal of the American Medical Informatics Association*, 1(2):161–174.

Friedman, C. (2000). A broad-coverage natural language processing system. In *Proceedings of the AMIA Symposium*, pages 270–274. American Medical Informatics Association, American Medical Informatics Association.

Haveliwala, T. H. (2002). Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526. ACM.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Meystre, S. and Haug, P. J. (2005). Evaluation of Medical Problem Extraction from Electronic Clinical Documents Using MetaMap Transfer (MMTx). *Studies in Health Technology and Informatics*, 116:823–828.

Montoya, I. (2017). Análisis, normalización, enriquecimiento y codificación de historia clínica electrónica (HCE). Master's thesis, Konputazio Ingeniaritza eta Sistema Adimentsuak Unibertsitate Masterra, Euskal Herriko Unibertsitatea (UPV/EHU).

Neves, M., Yepes, A. J., and Névéol, A. (2016). The Scielo Corpus: a Parallel Corpus of Scientific Publications for Biomedicine. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2942–2948. European Language Resources Association (ELRA).

Oronoz, M., Casillas, A., Gojenola, K., and Pérez, A. (2013). Automatic Annotation of Medical Records in Spanish with Disease, Drug and Substance Names. In J. Ruiz-Shulcloper, editor, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 18th Iberoamerican Congress, CIARP 2013*, pages 536–543. Springer.

Savova, G. K., Masanz, J. J., Ogren, P. V., Zheng, J., Sohn, S., Kipper-Schuler, K. C., and Chute, C. G. (2010). Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–5013.

Shah, N. H., Bhatia, N., Jonquet, C., Rubin, D., Chiang, A. P., and Musen, M. A. (2009). Comparison of concept recognizers for building the open biomedical annotator. *BMC bioinformatics*, 10(9).

Stevenson, M., Agirre, E., and Soroa, A. (2012). Exploiting domain information for Word Sense Disambiguation of medical documents. *Journal of the American Medical Informatics Association*, 19(2):235–240.

Yetano, J. (2003). Diccionario de siglas médicas y otras abreviaturas, epónimos y términos médicos relacionados con la codificación de las altas hospitalarias.

Revisiting the Task of Scoring Open IE Relations

William Léchelle, Philippe Langlais
RALI - University of Montreal
{lechellw, felipe} @ iro.umontreal.ca

Abstract

Knowledge Base Completion infers missing facts from existing ones in knowledge bases. As recent Open Information Extraction systems allow us to extract ever larger (yet incomplete) open-domain Knowledge Bases from text, we seek to probabilistically extend the limited coverage we get from existing facts, to arbitrary queries about plausible information. We propose a simple baseline, based on language modeling and trained with off-the-shelf programs, which gives competitive results in the previously defined protocol for this task, and provides an independent source of signal to judge arbitrary fact plausibility. We reexamine this protocol, measure the (large) impact of the negative example generation procedure, which we find to run contrary to the belief put forward in previous work. We conduct a small manual evaluation, giving insights into the rudimentary automatic evaluation protocol, and analyse the shortcomings of our model.

Keywords: Open Information Extraction, Evaluation, Knowledge Base Completion

1. Introduction

Much recent effort has been put into building Knowledge Bases (KBs), either manually curated (Freebase (Bollacker et al., 2008), Cyc (Lenat, 1995)) or automatically produced (YAGO (Suchanek et al., 2007), Knowledge Vault (Dong et al., 2014)), ranging from logically consistent linked-data in OWL (SUMO (Pease et al., 2002)) to little-structured sets of textual relations extracted from text (NELL (Mitchell et al., 2015)) with Open IE systems (Reverb, Ollie (Mausam et al., 2012), ClausIE (Del Corro and Gemulla, 2013), Stanford Open IE (Angeli et al., 2015), CSD-IE (Bast and Haussmann, 2013)). However large they may be, typical KBs are largely incomplete, and many relevant facts are missing (West et al., 2014).

Because an exhaustive coverage of the information that ought to be part of the KB is a very desirable feature, KB completion (inference of missing facts from known ones) is a rapidly growing field (West et al., 2014; Nickel et al., 2015; Wang et al., 2015; Toutanova et al., 2016).

In the context of Open Information Extraction (OIE), our aim is to assign a score to an arbitrary unseen query fact[1], judging its *plausibility* as a member of the KB. This task is important for three reasons : first, it extends the coverage of the existing KB probabilistically to any query, greatly improving upon the closed-world assumption that facts not known to be true are false. Second, as the extraction of information in the open domain is a relatively noisy process, a confidence score helps detecting extraction errors, and makes for higher-quality automatically generated KBs. Last, adjusting the confidence threshold of extracted facts allows to tune as desired the trade-off between precision and recall of the extraction process.

The task has attracted little attention since it was introduced in (Angeli and Manning, 2013), most that we know in (Li et al., 2016). The authors propose to assign high KB-membership probability to facts that have *support facts* existing in the KB, which are similar to them (based on common phrases and word similarity).

We propose a new baseline for this task, in the form of a language model. Whereas the method of (Angeli and Manning, 2013) is fairly complicated to implement, and requires indexing the KB in various ways for intermediate computations, a trained language model is very compact, straightforward to implement and train, and fast to process requests at use time. We train the model on automatically extracted facts (including some noise) from the same corpus, i.e. the knowledge base, taken as a list of sentences. We experiment with language model features and show that a linear classifier gives good results at the task of recognising actual extracted facts, perhaps unsurprisingly.

We then go back over the experimental protocol proposed by (Angeli and Manning, 2013) and consider the way negative examples are automatically generated. We find that this procedure has a significant impact on the difficulty of the task. At last, we go back to the goal of improving existing extractions by picking out the noise. Instead of an automatically generated test set, we measure the ability of the models to identify the remaining wrong extractions in the RV-15M high-quality dataset (presented in section 4.1.).

2. Related work

2.1. Knowledge Base Completion

Much work in Knowledge Base Completion (KBC) has been done in recent years (Bordes et al., 2013; Riedel et al., 2013; García-Durán et al., 2015; Feng et al., 2016; Trouillon et al., 2016), on tasks very similar to ours, mostly focusing on Freebase, and other such large manually curated KBs (WordNet, NCI-PID (Schaefer et al., 2008), etc.).

The major difference between our approach and most KBC work is the predefined schema of the KB. The arguments of the relations curated in Freebase are mostly named entities, and the relations to be gathered were defined when building the KB. FB15k, a popular Freebase dataset, covers 1345 predicates, though only 401 have more than 100 occurences (Yang et al., 2014). NELL captures about 150 relations, and WordNet about 20. By contrast OIE seeks to extract all the relations expressed in text, resulting in hundreds of

[1] A "fact" is a relation phrase linking two or more argument phrases. The arguments need not be named entities.

thousands of relation predicates (even though many are synonyms). The RV-15M dataset used in this work has 660k distinct relation strings.
(Toutanova et al., 2015) embed surface textual patterns in the same vector space as the KB relations, which is similar to the implicit embedding of all predicates in the same vector space as we do. Yet their work only predicts relations based on the 237 predicates of the FB15k-237 dataset, whereas we predict confidence scores for all relations, including predicates never seen during training.

2.2. Angeli and Manning (2013)

In this work we reexplore in depth the task set up in (Angeli and Manning, 2013), and it is their work that is most similar to ours. They seek to probabilistically extend a KB to arbitrarily any query fact, in the sense that any candidate fact has a KB-membership probability (or confidence score). Indeed, this is a sensible way of considering a knowledge base system that must perform inferences.
To this aim, they compare a query fact to a set of candidate support facts from the KB. The candidate facts need to share 2 phrases (arguments or relation) with the query fact, and is allowed to differ by the third part. The query fact has a high KB-membership score if it is similar enough to its closest support facts (that is, the differing part is similar enough). The algorithm is as follows:

- Gather candidate facts that can support the query fact: candidates must share two of (argument 1, relation, argument 2) with the query, these phrases having at least the same head word. Stricter criteria are used as long as there is a sufficient number of candidates.
- Compute the distances between the query fact and each of its supporting facts, using 11 distance metrics, based on both distributional similarity and the WordNet thesaurus - cosine, Jaccard, etc.
- The highest similarities are used as weights in a linear classifier, whose job is to agreggate the similarity values across candidates and distance metrics.

2.3. OIE Systems Confidence

Contrarily to most OIE systems in which the confidence score is often an afterthought, Fader et al. (2011) went to great lengths to develop Reverb's confidence function (see their Section 4.2). They manually labeled the extractions from 1000 Web sentences as correct or incorrect, and trained a classifier using features about the *original sentence* to assign a confidence score to the extraction process. The confidence function of Ollie is based on the frequency of the syntactic pattern that was used to extract a given fact. ClausIE simply returns the confidence score of the underlying dependency parser, as its rules rely directly on it.
In contrast to our work, the confidence scores produced by OIE systems rely on the original sentence from which they were extracted, and on how well the extraction procedures could handle the input sentences. Our goal is to judge the quality of query facts as they stand, regardless of the sentences they come from. For instance, with the Chomsky sentence *Colorless green ideas sleep furiously* as input, past OIE evaluation would consider (*ideas, sleep, furiously*) to be a correct extractions, whereas the goal of our system is to reject any "fact" coming from that sentence on the grounds that they do not make sense.

3. Revisiting the Task Setup

Our end goal is to improve the OIE process by pruning out the erroneous or empty facts produced. We frame this as a classification problem, seeking to distinguish correct useful facts from ill-constructed or void statements.

3.1. Task Protocol

A KB is constructed by running an open information extractor over a textual corpus. Even though there is some noise, extracted facts are assumed to be correct, and samples of them are set aside from the KB to constitute positive examples of unseen facts for the classification task.
For negative examples, artificial facts are constructed by replacing one part of a genuine extraction with that of another. Let (a_1, r, a_2) and (a'_1, r', a'_2) be two genuine facts from the KB, then one negative example is picked between (a'_1, r, a_2), (a_1, r', a_2) and (a_1, r, a'_2). We will show in Section 4.3. that this choice is a very significant parameter of the experimental setup.
Classifiers are trained to discriminate the positive from the negative examples. The performance metric is the classification accuracy.

3.2. Approaches

ArgSim is a weak baseline for the task, measuring the cosine similarity between a_1 and a_2 in a fact. Arguments (effectively bag-of-words) are represented by the average of their individual words' embeddings. This performs well on ConceptNet, per (Li et al., 2016), but captures little information on OIE facts. This is both reported by Angeli and Manning (2013) and replicated in our experiments.
The full algorithm of (Angeli and Manning, 2013), presented in Section 2.2., is denoted **AM-system** in Table 1. We reimplemented the **count** and **cos** methods used in their evaluation, which are both simplified versions of their approach (the former coarse, and the latter close in principle to the full-fledged scoring function).
As has been noted by (Stanovsky et al., 2015), OIE output can be used as training material for other tasks such as text comprehension, word similarity and analogy. This is because OIE produces a distinctive intermediate representation of the sentence, from which complementary features (to that of dependency parse or lexical representations) can be extracted.
Moreover, in the confidence scoring function of Reverb, several features capture how completely the extraction covers the sentence's tokens. In short, the most typical correct extractions look like short declarative sentences, like (*Hudson*, *was born in*, *Hampstead*), or (*Hampstead*, *is a suburb of*, *London*). Then, it seems natural to train a language model on confidently extracted facts, and to expect from correct unseen extractions that they fit well and cause low perplexity. This implements the assumption that an unseen fact is plausible iff it resembles a short and natural-sounding sentence.

The most basic implementation of this idea (**LM-basic**) is to straightforwardly use the probability of concatenated ($a_1.r.a_2$) as its score. Next, we notice that given the way negative facts are constructed, all three argument and relation spans are probable as they stand (since they come from genuine extractions). What makes the fact incorrect is that the parts do not fit together. Therefore, we use as a score the (log) probability of the whole fact minus that of each individual part. We call this model **LM-junctions**. With $\{a_1, r, a_2\}$ the fact expressing that the relation r holds between the two arguments a_1 and a_2, and p the language model probability function : $\text{score}(\{a_1, r, a_2\}) = \log p(a_1.r.a_2) - \log p(a_1) - \log p(r) - \log p(a_2)$.

We trained the language models with KenLM (Heafield, 2011), using the knowledge base itself as a corpus, each fact being considered as a sentence. We trained 5-gram models (the default), doing as little parameter tuning as possible.

Further, we train a linear classifier based on linguistic modeling-based features (**LM-SVM**). We implemented the SVM with scikit-learn (Pedregosa et al., 2011)[2]. It uses 20 features such as the individual log-probabilities of the various parts, the log-probabilities of various bigrams and trigrams focused on the argument-relation junctions, and arithmetic operations over those values.

The classifier is trained on 10k genuine extractions as positive examples, and 10k artificially constructed ones as negative examples, counts picked to be the same as those in (Angeli and Manning, 2013).

4. Experiments

4.1. Dataset

We used Reverb-15M, a shared[3] dataset of high-quality binary assertions extracted by Reverb on the ClueWeb09 corpus. In order to obtain a high-precision dataset, its authors filtered the extractions by Reverb's confidence function (with a 0.9 threshold), stopwords, and frequency, along with certain syntactic criteria.[4] We used the normalized (lemmatized) version of the tuples. Taken as a text corpus, the RV-15M dataset is 98M tokens long.

Angeli and Manning (2013) used a similar set of extractions : the authors ran Reverb over ClueWeb09 themselves, filtered out extractions scoring under 0.5 per Reverb's confidence function, and retained the first billion extractions, which results in a KB of 500 million unique facts. Their dataset is thus larger, and noisier, than the one we used.

4.2. Classification

Table 1 shows the performance of our approach, along with the methods **count** and **cos** of (Angeli and Manning, 2013). The most elementary language modeling idea demonstrably captures some useful signal for the task. By focusing on the probability of the argument-relation junctions, the language model improves to near state-of-the-art performance.

	Reverb-15M (AUPRC)	Reverb-500M†
Random	50.0 (0.500)	50.0
ArgSim	53.0 (0.283)	52.6
AM-count	61.6 (0.399)	52.3
AM-cos	64.2 (0.462)	70.6
AM-system		74.2
LM-basic	65.4 (0.471)	
LM-junctions	73.6 (0.589)	
LM-SVM	76.3 (0.628)	

Table 1: Classification accuracy of the scoring methods on Reverb-15M, evaluated on the automatically-generated test set. Area under the precision-recall curve is indicated in parenthesis. † Results on Reverb-500M (a similar, larger and noisier dataset) published in (Angeli and Manning 2013), are reproduced for comparison.

By training a linear classifier on top of the language modeling features, we can gain 3 additional points in precision, surpassing previous state-of-art performance.

Examples of correct tuples scored highly by our model are (*Austin Airways, was an airline based in, Canada*) and (*Children, are welcome in, the Curriculum Lab*).

Reversely, (*A knight, can turn into, a serious problem*) and (*Hand, be in, The Los Angeles Times*) are examples of incorrect tuples scored highly. The first argument of the former was originally '*A bad sunburn*' and the relation of the latter was '*has also written for*'. Both facts are plausible and show the limits of the automatic evaluation procedure (*be in* is accidentaly close in meaning to *has also written for*, for a journalist and a newspaper).

Correct tuples scored poorly include (*A bounty Hunter, sent to kill ; Gust*) and (*Casey, also works on, pair-NIC*). Our model finds high perplexity in infrequent proper nouns. A Named Entity Recognition module would certainly improve the model, so that *<PERSON> also works on <MISC>* is seen at training time (or training data featuring the entities of interest if the KB focuses on a certain domain). Bad tuples correctly identified include (*81.45% of total wagers, can also avail the services of, bettors*) and (*A trellis, can also refer to, permission of the artist a structure*), in which relation and second argument formerly were '*returned to*' and '*a structure*' respectively.

4.3. Impact of Negative Examples Sampling Method

One important task parameter is the way negative examples are constructed. With (a_1, r, a_2) and (a_1', r', a_2') two genuine extractions, then one of (a_1', r, a_2), (a_1, r, a_2'), or (a_1, r', a_2) will be used as a negative example. We examine the impact of choosing one of the former two (changing an argument) versus picking the latter (changing the relation[5]). In Figure 1, we vary the fraction of relation changes over argument changes (a "knob" of the task setup), and measure the ensuing precision of systems.

[2]We shortly experimented with Logistic Regression and Random Forest classifiers, giving slightly inferior results. We also tried several available SVM kernels, the default RBF kernel giving slightly better results than the others.

[3]Available at `http://reverb.cs.washington.edu/`

[4]See `reverb.cs.washington.edu/README_data.txt`

[5]Or, from a different perspective, changing *both* arguments with respect to the other fact, since facts used for this construction are picked at random.

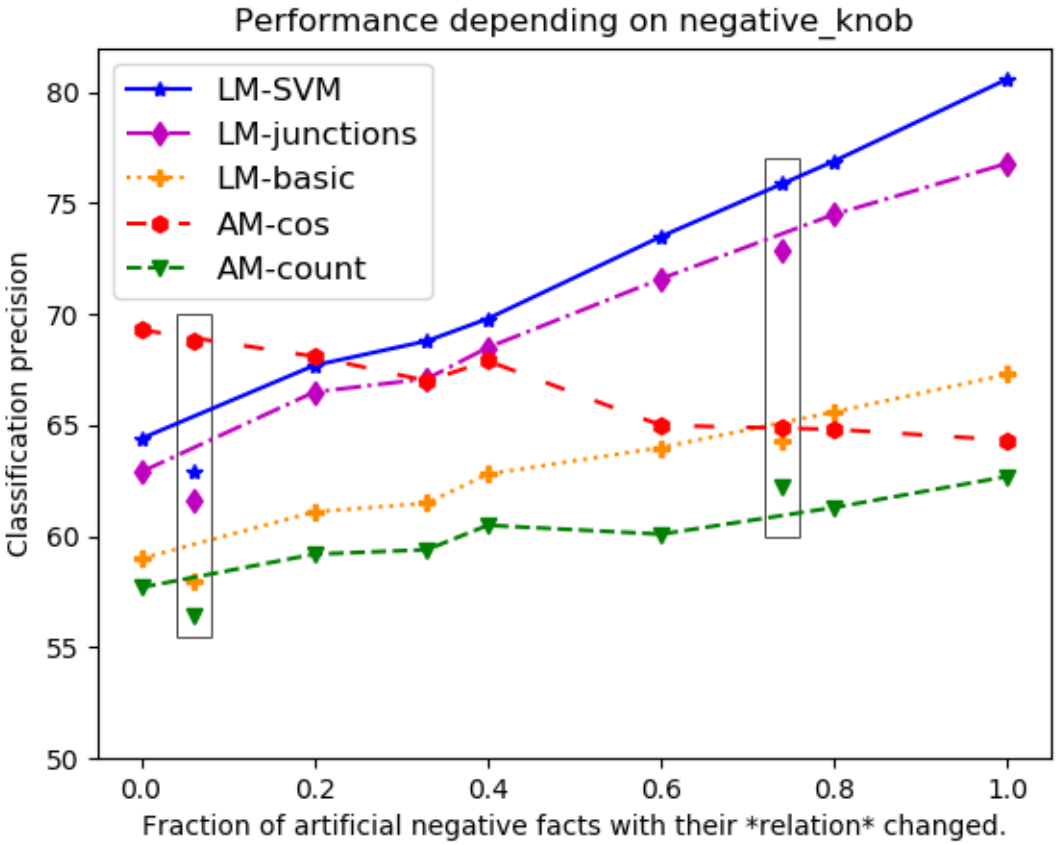


Figure 1: Performance depending on the proportion of negative examples where the *relation* was changed, rather than one of the arguments. 0.33 corresponds to picking at random, which we recommend. The method used by Angeli and Manning is not equivalent but corresponds to a value close to 0.75.

Angeli and Manning (2013) use a particular scheme : out of the 3 negative example candidates, they pick the one that has the largest cosine similarity with the original fact (a_1, r, a_2). This is with the stated purpose of training the classifier to discriminate between more similar examples, supposedly a better learning setup.

In practice, this means changing the relation in about 75% of cases, similarly to setting the knob at 0.75 in Figure 1 [6]. In practice, the fact most similar to the original tends to be the one with the swapped relation, because relational phrases are more often similar to each other than argument phrases. This in turn, we suppose, is because phrases are treated as bag of words and represented by their average word embedding, and relation phrases often share common verbs such as *be*, *make*, etc. and prepositions, whereas argument phrases are more distinct. Between $d(a_1, a'_1)$, $d(r, r')$ and $d(a_2, a'_2)$, $d(r, r')$ is the largest of the three in only 7% of cases (this choice of negative examples is the column of values at 0.07 on the x-axis).

Looking at Figure 1, we can see that **swapping the relation phrase makes the task easier** : except for AM-cos, all systems perform better, in a linear fashion. From a language modeling perspective, this is easy to explain : swapping the relation introduces two breaks inside the short sentence, where words may not fit together, instead of just one when an argument is replaced. When building artificial facts in this way, we recommend picking one of the 3 candidate negatives at random (i.e. setting the knob on 0.33). For more difficult learning tasks[7], further research could leverage other sources of information to produce better distractors as negative examples. This could be argument types, or using the fact that certain relations have only one correct value (e.g. *<person>, be born on, <date>*), so that differing values are known to be false.

4.4. Results on Manually Annotated Tuples

Our classification task up to now assumed all extracted facts to be correct, and all randomly-generated facts to be wrong. In practice this is not always the case, as some noise remains in the high-quality Reverb dataset, and some randomly assembled facts turn out to have interpretations that make them right, at least plausible. For instance (*a knight,*

[6] The difference is that our graph shows performance for a certain proportion of relation changes picked at random, whereas Angeli and Manning select a particular set of 75% relation changes. This is why points were made to be off the regression lines in the chart.

[7] For instance, distinguishing facts that are actually true in the world from facts that are false or sometimes false, or from facts that are merely plausible.

	Reverb-15M-manual
ArgSim	63.5
AM-count	47.7
AM-cos	57.5
LM-basic	52.6
LM-junctions	51.3

Table 2: Scores of unsupervised models on the manually annotated test set.

can turn into, a serious problem) would be true in medieval times, or in chess commentary.

Therefore, in an effort to come nearer the original task of recognizing impossible from plausible OIE extractions, we experiment with a small set of manually annotated facts. It is a delicate issue to decide what constitutes a correct extraction: see for instance Section 3.1 in (Stanovsky and Dagan, 2016). We annotated 430 tuples manually as:

- good extractions (45%) : capturing *some* information, and at least sometimes true. Some examples are (*Blackberry picking, is a great introduction to, foraging*), (*A new computer ; only costs around ; 500$*), and (*Blood pressure, is influenced by, dietary fibre*).

- unsure (30%) : typically only true in their original context which is lost, neither wrong nor useful in a vacuum. For instance (*Alfred, was against, garages*), (*Archaeology, answers this question with, confidence*), or (*Access ; is limited to ; official business*).

- incorrect extractions (25%) : nonsense or false, often due to upstream parser errors or noisy source text. E.g. (*5 Mts, Walk From, Wembley Stadium*), (*Atomic Kitten, released, the*) and (*A whole day ; set aside for ; literary pursuits*).

We annotate the tuples regardless of the sentences from which the facts were extracted : we label the facts as they stand and not the extraction process. The *unsure* labels (covering 25% of examples) were ignored and models had to classify correct and incorrect extractions. The negative examples were oversampled to balance the dataset.

The models used are the same as in Section 4.2., except that the SVM cannot be trained, as there is no training data (all facts are positive examples in the automatic task setup). Results are presented in Table 2. Among genuine extractions (all scored highly by Reverb's confidence function), current models have a hard time recognizing ill-formed or nonsensical extractions, and perform worse than on the automatically generated test set.

Some incorrect facts are scored highly because some particular pattern was often repeated over the web and systematically misinterpreted by the OIE system. One example is *Cross listed with <another class>.*, occuring frequently on certain university curriculum pages, from which Reverb extracts (*Cross, listed with,* e.g. *AIST2340*). This is correct from a language modeling perspective, even though it's a wrong fact.

4.5. Human performance

In an attempt to gauge the impact of false positives (genuine extractions by ReVerb that turn out to be erroneous) and false negatives (artificially assembled facts that turn out to be plausible), two human annotators, both NLP practitioners and including one author, manually performed the task on 200 facts. Half of them were not lemmatised. As in Table 1, the negative sampling method was that of (Angeli and Manning, 2013).

Both judges achieved just 80% accuracy at discriminating genuine and artificial facts, on both the lemmatised and unlemmatised versions of the task. Agreement was also 80%. Examples of highly ambiguous facts on which both annotators were mistaken include:

- (*Zaire ; will be maimed by ; betrayal*)
- (*Jean ; is a native of ; New York*)
- (*Peas ; here take advantage of ; ringtones*)
- (*Cooking school ; have changed a bit in ; the Los Angeles area*)

The first and third are genuine extractions (positive examples in the automated task) while the second and fourth were assembled at random (negative examples). Such facts constitute 10% of the test set.

Examples of facts on which annotators disagreed (one being mistaken) are:

- (*shimer college ; be establish in ; mt*)
- (*the north node ; will be in ; pisces*)
- (*Specific attention ; will be given to ; THE MAN*)
- (*Zoroastrianism ; is in even ; worse shape*)
- (*Kara ; is vice president of ; buying*)

Out of those five, only the third is artificial, and others are genuine extractions (the first two being lemmatised, as in the automatic evaluation). Such facts constitute 20% of the test set.

Overall, it is as if 60% of the automatically generated testset was reliably recognisable as genuine or artificial, humans performing no better than chance on the remaining 40% (hence the resulting 80% performance).

5. Conclusion

We revisit the task of judging the plausibility of a new candidate fact to extend a knowledge base, in the context of OIE — arbitrary relations between unrestricted noun phrases. Correctly assessing the validity of an unknown fact is highly valuable, both as a way to refine KBs built automatically, and to implicitly enhance finite stored knowledge for it to answer an order of magnitude more queries.

We propose a new baseline for this task, based on language modeling, which achieves state of the art performance. Indeed, archetypal correctly extracted information resembles short declarative sentences. We show that the way artificial negative examples are sampled has a large and robust impact on the difficulty of the task. We manually find genuine

yet incorrect extractions and show that while our system does capture some useful signal, picking up wrong extractions from a high quality dataset remains a challenging task. We examine the sort of facts that our model gets "right" despite the test set generation method being wrong, and the sort of facts on which it performs poorly. Future work extending the language model beyond off-the-shelf programs with named-entity recognition would improve its performance.

6. Acknowledgments

We would like to thank Fabrizio Gotti for his continuous help with annotation, alongside ever helpful comments. This work was supported by the Nuance Foundation.

7. Bibliographical References

Angeli, G. and Manning, C. D. (2013). Philosophers are mortal: Inferring the truth of unseen facts. In Julia Hockenmaier et al., editors, *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013*, pages 133–142. ACL.

Angeli, G., Premkumar, M. J., and Manning, C. D. (2015). Leveraging linguistic structure for open domain information extraction. In *ACL*.

Bast, H. and Haussmann, E. (2013). Open information extraction via contextual sentence decomposition. In *Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on*, pages 154–159. IEEE.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250, New York, NY, USA. ACM.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 2787–2795. Curran Associates, Inc.

Del Corro, L. and Gemulla, R. (2013). Clausie: Clause-based open information extraction. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 355–366, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Dong, X. L., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610. Evgeniy Gabrilovich Wilko Horn Ni Lao Kevin Murphy Thomas Strohmann Shaohua Sun Wei Zhang Geremy Heitz.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1535–1545, Stroudsburg, PA, USA. Association for Computational Linguistics.

Feng, J., Huang, M., Yang, Y., and Zhu, X. (2016). GAKE: graph aware knowledge embedding. In Nicoletta Calzolari, et al., editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 641–651. ACL.

García-Durán, A., Bordes, A., Usunier, N., and Grandvalet, Y. (2015). Combining two and three-way embeddings models for link prediction in knowledge bases. *CoRR*, abs/1506.00999.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, Edinburgh, UK, July. Association for Computational Linguistics.

Lenat, D. B. (1995). Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38, November.

Li, X., Taheri, A., Tu, L., and Gimpel, K. (2016). Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, pages 1445–1455. Association for Computational Linguistics.

Mausam, Schmitz, M., Bart, R., Soderland, S., and Etzioni, O. (2012). Open language learning for information extraction. In *Proceedings of Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CONLL)*.

Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B., Krishnamurthy, J., Lao, N., Mazaitis, K., Mohamed, T., Nakashole, N., Platanios, E., Ritter, A., Samadi, M., Settles, B., Wang, R., Wijaya, D., Gupta, A., Chen, X., Saparov, A., Greaves, M., and Welling, J. (2015). Never-ending learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*.

Nickel, M., Murphy, K., Tresp, V., and Gabrilovich, E. (2015). A review of relational machine learning for knowledge graphs: From multi-relational link prediction to automated knowledge graph construction. *CoRR*, abs/1503.00759.

Pease, A., Niles, I., and Li, J. (2002). The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Riedel, S., Yao, L., Marlin, B. M., and McCallum, A. (2013). Relation extraction with matrix factorization and universal schemas. In *Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguis-*

tics (HLT-NAACL '13).

Schaefer, C. F., Anthony, K., Krupa, S., Buchoff, J., Day, M., Hannay, T., and Buetow, K. H. (2008). Pid: the pathway interaction database. *Nucleic acids research*, 37(suppl_1):D674–D679.

Stanovsky, G. and Dagan, I. (2016). Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Austin, Texas, November. Association for Computational Linguistics.

Stanovsky, G., Dagan, I., and Mausam. (2015). Open ie as an intermediate structure for semantic tasks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 303–308, Beijing, China, July. Association for Computational Linguistics.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA. ACM.

Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., and Gamon, M. (2015). Representing text for joint embedding of text and knowledge bases. In Lluís Màrquez, et al., editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1499–1509. The Association for Computational Linguistics.

Toutanova, K., Lin, V., Yih, W., Poon, H., and Quirk, C. (2016). Compositional learning of embeddings for relation paths in knowledge base and text. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., and Bouchard, G. (2016). Complex embeddings for simple link prediction. *CoRR*, abs/1606.06357.

Wang, Q., Wang, B., and Guo, L. (2015). Knowledge base completion using embeddings and rules. In *IJCAI*, pages 1859–1866.

West, R., Gabrilovich, E., Murphy, K., Sun, S., Gupta, R., and Lin, D. (2014). Knowledge base completion via search-based question answering. In *WWW*.

Yang, B., Yih, W., He, X., Gao, J., and Deng, L. (2014). Embedding entities and relations for learning and inference in knowledge bases. *CoRR*, abs/1412.6575.

A supervised approach to taxonomy extraction using word embeddings

Rajdeep Sarkar[1], **John P. McCrae**[2], **Paul Buitelaar**[2]
[1]Indian Institute of Technology Kharagpur
[2]Insight Centre for Data Analytics, National University of Ireland Galway
rajdeep.sarkar@iitkgp.ac.in,
john@mccr.ae,
paul.buitelaar@insight-centre.org

Abstract

Large collections of texts are commonly generated by large organizations and making sense of these collections of texts is a significant challenge. One method for handling this is to organize the concepts into a hierarchical structure such that similar concepts can be discovered and easily browsed. This approach was the subject of a recent evaluation campaign, TExEval, however the results of this task showed that none of the systems consistently outperformed a relatively simple baseline.In order to solve this issue, we propose a new method that uses supervised learning to combine multiple features with a support vector machine classifier including the baseline features. We show that this outperforms the baseline and thus provides a stronger method for identifying taxonomic relations than previous methods.

Keywords: taxonomy extraction, word embeddings, supervised learning, terminology

1. Introduction

When confronted with a large collection of texts a key challenge is to organize these texts into a structure that makes it easy to discover the information that is behind these texts. For example, it is common that large organizations will collect many documents generated by their staff into a content management system, with only limited organization such as tags. One of the most widely used methodologies for organizing text collections is that of the taxonomic structure, where a set of topics (defined by a set of terms) are organized into a tree structure and individual documents are then clustered under these topics. This method, as exemplified in bibliothecal classification schemes such as the Dewey Decimal Classification scheme, has been widely employed by libraries to organize large collections of texts for over one hundred years. Such schemes are still widely employed by organizations such as the ACM[1] or the IEEE (IEEE, 2017) to organize contributions to major conferences and other research contributions.
The task of organizing topics into a hierarchy is called taxonomy extraction and consists of two key steps. Firstly, in the *term extraction* step the set of terms that are relevant can be extracted from a collection of texts and secondly, in the *taxonomy learning* step these terms are grouped into a hierarchical structure. The first step is well explored and recent strong results have been shown on this task (Astrakhantsev, 2016; Buitelaar et al., 2013) and as such we shall not focus on it in the course of this article. The second task is however much less well-explored and this is the focus of this article, and so we assume that the terms have already been identified by an approach such as those outlines above. The task of taxonomy extraction is closely related to tasks such as *hypernym detection* (Hearst, 1992) or *ontology learning* (Buitelaar et al., 2005), in which a structured representation of concepts should be learned. However, the task of taxonomy extraction does not have the formal nature that either of these tasks in that the terms only need to be loosely associated. For examples, taxonomies frequently place terms under broader concepts that do not match the strict requirements of ontological subsumption (Gangemi et al., 2002) that would be required from an ontology, e.g., grouping "Kalman Filter" is under "Filtering" in the IEEE taxonomy, where a type of filter is grouped under the concept of filtering. Moreover, often concepts are grouped in a way that does not conform to a strict hypernym or broader/narrower relationship, such as "Google" which is grouped under "Computer Networks" in the IEEE taxonomy. As such, this task is subtly yet importantly different to the other tasks.
Recently the TExEval task (Bordea et al., 2016) was created to provide an open forum for evaluating the performance of systems on this task. For this task, a simple method based on whether one term was a substring of another string was proposed as a baseline, while competitors in the shared task took many methods inspired from hypernym extraction and ontology learning. Surprisingly, the results of the task showed that none of the competitors consistently outperformed this very simple baseline. This underlines the highly distinct nature of this task, when compared to hypernym extraction and the need for novel approaches. In this context, we approach the problem as a supervised learning task, where individual features can be extracted from the label as well as word embeddings associated with these labels and these features can be weighted by means of a support vector machine (Cortes and Vapnik, 1995). We show that while none of theses features outperform the baseline of the TExEval baseline, the combination of these features with the baseline can provide a stronger result than the baseline.

2. Related Work

While taxonomy extraction is not simply the task of hypernym identification, this is one of the most important goals of such research and several methods going back to (Hearst, 1992) have been proposed for this task. A recent such system to use this is the TAXI system (Panchenko et

[1]http://www.acm.org/publications/class-2012

al., 2016), which combined simple string substring metrics with Hearst-like patterns learned from text, which are then constructed into a taxonomy using Tarjan's algorithm (Tarjan, 1972).

A different approach, by (Tan et al., 2016), used the *endocentricity* of a term, that is if a term contains another term, e.g., 'fish' in 'goldfish', whether this indicates a hypernym-like relationship. The QASSIT system (Cleuziou and Moreno, 2016) used the genetic algorithm in order to learn taxonomic relations, however performance across domains was poor.

BabelNet (Navigli and Ponzetto, 2012), a wide coverage dictionary, has been used both as a source of information about taxonomic relations (Maitra and Das, 2016) and also itself was constructed using automatic taxonomy learning (Navigli et al., 2011). However, the focus of this has been mostly on single words as would be found in a dictionary and less on the kind of multi-word terminology that can be used to describe specialist domains as in this paper.

Finally, there has been some work on the use of word embeddings to predict hypernym relations, such as in (Fu et al., 2014), where a linear project function from a word embedding to its hypernym was constructed. The possibility of combining this with syntactic patterns for hypernym discovery has also been investigated (Shwartz et al., 2016).

3. Methodology

Our methodology is based on the use of multiple features that can be extracted from the labels or an associated corpus. We then learn to combine these using a supervised learning approach. In this section we will present the methodology for the features we used.

3.1. Data

While, the TexEval task has recently given a baseline, by which performance on this task can be measured. There are some weaknesses with this baseline in particular that most of the sources are resources such as WordNet (Fellbaum, 1998), which are primarily databases of hypernyms. As such using these datasets does not distinguish the task well from that of hypernym detection. In fact, taxonomies are not intended to be strict hypernym graphs but may in fact contain other relations. However, we use this as the basis for our experiments as this dataset is established and has been used by other systems. In addition, to these datasets, we also make use of a large background corpus for our experiments. In this case, we use the English Wikipedia [2]. The Wikipedia dump file of English language which contains 2,976,828 number of articles with 1,940,620,295 unique terms. This is used particularly for the Jaccard similarity (see below) where we consider the articles that contain a given term. In addition, we use the Word2Vec vectors pre-trained on Google News [3] for calculating word similarity. As several of our methods use supervised learning, for these experiments we simply held-out one of the TExEval datasets as the test set and used the remaining datasets for training.

[2] http://en.wikipedia.org/

[3] https://code.google.com/archive/p/word2vec/

3.2. Term substring

Following the baseline of the TExEval task (Bordea et al., 2016), we use a baseline that counts a string if it is a substring of the other string. Following discussions with the TExEval organizers, we exactly implement this as in the task, that is we introduce a link in the taxonomy graph if either term is equal (ignoring case), or one term starts with another term and then a word boundary, or one term ends with another term (not considering word boundaries)[4].

3.3. Jaccard similarity

To extract relations, we can use Jaccard similarity between two terms to develop the taxonomy tree. To find the Jaccard similarity between two terms, we count the number of articles which contain both the terms and divide it by the number of articles which contain at least one of the two terms. Mathematically this can be defined as:

$$JS(A, B) = \frac{|W_{A \cap B}|}{|W_{A \cup B}|}$$

After calculating the scores for every possible pair of terms, we sort the terms on the basis of their Jaccard scores and then greedily select the pairs on the basis of their Jaccard similarity such that loops are not formed in the tree. We continue this process until all the terms are present in the final tree. The final tree obtained using this method will contain all the term nodes present in the dataset.

3.4. Word2Vec Min-Max

In this approach, we use word embeddings to extract similarity between two terms. Word2Vec (Mikolov et al., 2013) is a machine learning technique to obtain numerical representation of words in the form of word vectors. Using this method, we can obtain vector representation of each word present in the corpus. For each word in a given term, we obtain a vector of fixed size K (we choose $K = 100$). To obtain a vector representation of the entire term, we use the Min-Max combination technique of combining word vectors.

Min-Max technique: We form 2 vectors for the term. The first is known as the Min vector. It is constructed by taking the minimum value at each index position of its constituent words by comparing the values present in that index position of the vectors of its word constituents. The second vector is constructed in a similar manner by taking the maximum value at each index position.

The final vector of the term is obtained by concatenating the minimum vector with the maximum vector to obtain a vector of size $2K$ (200 in our case). The similarity between two terms is calculated by first normalizing the word vectors of both the terms and then taking the cosine similarity of the two vectors. The final tree is constructed in a similar manner as the tree constructed in Section 3.3.

3.5. Word2Vec averaged

In this method we again use word vectors to compute similarity between two terms. However, to construct the word vector of the term, we take the average of the word vectors

[4] This implementation exactly follows that of TExEval

Corpus	Number of terms
Environment (Eurovoc)	261
Food (WordNet)	1486
Food (EN)	1555
Science (Eurovoc)	125
Science (Wordnet)	370
Science (EN)	452

Table 1: The size of datasets used in these experiments

of its constituent words. We again find the similarity between two terms using cosine similarity of the two vectors And follow the greedy approach to construct the tree.

3.6. SVD classification

Let $\mathbf{V}$ be the matrix of vectors obtained using the methods of Section 3.4. and 3.5. We form the matrix by with each row representing the word vectors of the terms and let $\mathbf{B}$ be the matrix representing the relations between the terms. So, for any two terms a and b of the matrix $\mathbf{B}$,

$$R(a,b) = \begin{cases} 1 & \text{if } a \subset b \text{ or if } a \supset b \\ 0 & \text{otherwise} \end{cases}$$

Thus is we have $\mathbf{v}_i$, $\mathbf{v}_j$ as the vectors corresponding to the words w_i,w_j then we choose to estimate our similarity using a linear function with a matrix, $\mathbf{A}$, as follows:

$$R'(w_i, w_j) = \mathbf{v}_i^{\mathrm{T}} \mathbf{A} \mathbf{v}_j^{\mathrm{T}}$$

We find the matrix $\mathbf{A}$ that minimizes the Euclidean error between our training data across all word pairs, e.g.,

$$\text{error} = \sqrt{\sum_{i=1,\ldots,|v|} \sum_{i=1,\ldots,|v|} (R(w_i, w_j) - R'(w_i, w_j))^2}$$

We can rewrite this in terms of matrices:

$$\text{error} = ||\mathbf{B} - \mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}||_{\mathcal{F}}$$

We shall assume that $\mathbf{V}$ is a $m \times n$ network and that $m > n$, in other words that the number of words is greater than the length of the Word2Vec vectors. We can then assume that we can minimize the error by finding the point where

$$\frac{\partial \text{error}^2}{\partial \mathbf{A}} = 0$$

We derive this using identities proven in the Matrix Cookbook (Petersen et al., 2008), readers are recommended to refer to this text to better understand the derivation here:

$$\begin{aligned} &\frac{\partial ||\mathbf{B} - \mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}||_{\mathcal{F}}^2}{\partial \mathbf{A}} \\ &= \frac{\partial}{\partial \mathbf{A}} \mathrm{tr}(\mathbf{B} - \mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V})^{\mathrm{T}}(\mathbf{B} - \mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}) \\ &= \frac{\partial}{\partial \mathbf{A}} \mathrm{tr}[\mathbf{B}^{\mathrm{T}}\mathbf{B} - \mathbf{B}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V} - \\ &\mathbf{V}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{V}\mathbf{B} + \mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}] \\ &= 2\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}\mathbf{V}^{\mathrm{T}} - 2\mathbf{V}\mathbf{B}\mathbf{V}^{\mathrm{T}} \end{aligned}$$

Thus:

$$\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{A}\mathbf{V}\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{B}\mathbf{V}^{\mathrm{T}}$$

We can easily find a matrix $\mathbf{V}^{+}$ that satisfies [5]

$$\mathbf{V}^{+}\mathbf{V} = \mathbf{I}$$

In particular this can be achieved by using the singular value decomposition of $\mathbf{V}$ in order to find

$$\mathbf{V} = \mathbf{U}\mathbf{\Sigma}\mathbf{W}^{\mathrm{T}}$$

Such that Σ is a diagonal matrix and the following hold

$$\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$$

$$\mathbf{W}\mathbf{W}^{\mathrm{T}} = \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$$

Then

$$\mathbf{V}^{+} = \mathbf{W}\mathbf{\Sigma}^{-1}\mathbf{U}^{\mathrm{T}}$$

It is clear that a solution to find $\mathbf{A}$ is:

$$\mathbf{A} = (\mathbf{V}^{+})^{\mathrm{T}}\mathbf{B}\mathbf{V}^{+}$$

To obtain the $\mathbf{A}$ matrix, we used all but one of the taxonomies as training data and evaluated on the rest of the taxonomies. As such, our evaluation is cross-domain, which potentially further impacts our baseline results. After obtaining the $\mathbf{A}$ vector using the above method, The vectors $\mathbf{V}$ for the remaining datasets were obtained and then the $\mathbf{B}$ matrix for the terms was generated.

3.7. Using SVM to identify relations

Finally, we treat this task as a multi class classification problems, with the numerical representations of the relations as follow:

$$R(A,B) = \begin{cases} +1 & \text{if } A \subset B \\ -1 & \text{if } A \supset B \\ 0 & \text{otherwise} \end{cases}$$

We deploy a SVM classifier to classify the relations. The training data for the SVM was constructed holding one dataset out as above and we used different features to obtain our improved results. The features are as follow:

Baseline feature : Since the baseline method has a better recall and F-measure as compared to other methods, we choose this to be one of the features.

Word overlap : We also consider the number of words that are present in both the terms. We then divide the number of words overlapping by the number of words present in the term containing more number of terms.

Longest common subsequence : For the two terms in consideration, we find the length of the longest common subsequence present and divide it by the length of the longest term.

[5] This is the Moore-Penrose pseudo-inverse of $\mathbf{V}$

Features		Environment (Eurovoc)	Food (EN)	Science (EuroVoc)	Food (Wordnet)	Science (EN)	Science (Wordnet)
Baseline: String matching	Precision	0.500	0.536	0.576	0.501	0.668	0.783
	Recall	0.199	0.177	0.153	0.244	0.269	0.213
	F-measure	0.285	0.266	0.242	0.328	0.383	0.335
	F&M	0.000	0.000	0.006	0.002	0.004	0.000
Jaccard Similarity	Precision	0.038	0.023	0.095	0.037	0.053	0.081
	Recall	0.019	0.011	0.048	0.022	0.026	0.050
	F-measure	0.026	0.015	0.064	0.027	0.035	0.062
	F&M	0.000	0.000	0.000	0.022	0.000	0.050
Word2Vec Min-Max	Precision	0.100	0.016	0.127	0.053	0.085	0.109
	Recall	0.050	0.007	0.065	0.025	0.039	0.052
	F-measure	0.066	0.010	0.086	0.034	0.053	0.071
	F&M	0.000	0.000	0.000	0.000	0.000	0.000
Word2Vec Averaged	Precision	0.099	0.016	0.127	0.053	0.090	0.109
	Recall	0.050	0.007	0.065	0.025	0.041	0.052
	F-measure	0.066	0.010	0.086	0.034	0.056	0.071
	F&M	0.000	0.000	0.000	0.000	0.000	0.000
SVD Min-Max Classification	Precision	0.008	0.328	0.532	0.000	0.571	0.000
	Recall	0.007	0.112	0.202	0.000	0.069	0.000
	F-Measure	0.008	0.175	0.292	0.000	0.123	0.000
	F&M	0.001	0.104	0.000	0.000	0.011	0.000
SVD Averaged Classification	Precision	0.000	0.031	0.021	0.002	0.018	0.008
	Recall	0.000	0.011	0.008	0.001	0.002	0.005
	F-Measure	0.000	0.017	0.012	0.001	0.004	0.006
	F&M	0.000	0.000	0.000	0.000	0.000	0.014
SVM	Precision	0.341	0.683	0.391	0.515	0.474	0.463
	Recall	0.735	0.301	0.605	0.498	0.735	0.735
	F-Measure	0.577	0.418	0.475	0.506	0.577	0.568
	F&M	0.083	0.019	0.072	0.018	0.286	0.072

Table 2: Comparison of Unsupervised and Unsupervised Taxonomy Construction

SVD scores : We use the SVD scores from Section 3.6. between terms as one of the features for the SVM classifier since the tree obtained using this method had a higher F&M measure.

Word2Vec scores : We also use the word2vec scores as one of the features since word2vec scores between terms capture semantic similarity quite well.

Jaccard similarity scores : We use this as another feature since the this captures the co-occurrence of terms which gives an indication of their relatedness.

We used radial basis function(RBF) kernel and obtained the Cost(C) and Gamma(G) using grid search and ten fold cross validation.

We applied an ablation analysis, where we consider the effect of removing a single feature from the classifier and this analysis is presented in Table 3, where we present the average precision, recall and F-Measure over all results. We found that the there was a statistically significant improvement using McNemar's test at 5% (McNemar, 1947), by using all features except for the baseline and a non-significant improvement using the Word2Vec scores.

3.8. Taxonomy Construction

Once we have developed a function for each pair of terms in the taxonomy, which estimates whether it is likely to be a taxonomic relation and the direction of this relation, we need to extract a complete taxonomy. For these experiments we assume that the extracted structure should be a tree (acyclic graph) that contains all the topics that are candidates for this learning task. Initially, following earlier proposed solutions to the task (Navigli et al., 2011) we attempted to find a maximal spanning tree using Kruskal's algorithm (Kruskal, 1956). However we found that these results were disappointing and that a stronger and simpler approach was to greedily construct a tree by taking the highest scoring edges and adding them to the current graph, unless this would lead to a cycle. This process continues until all topics are included in the taxonomy.

4. Evaluation

We use the dataset used in the Taxonomy Extraction Evaluation (Bordea et al., 2016) which was a part of the International Workshop on Semantic Evaluation organized by ACL SIGLEX. The datasets consists of a set of terms from different corpora and the statistics of the datasets are given in Table 1.

	Precision	Recall	F-Measure
All features	0.676	0.478	0.429
- Baseline feature	0.550	0.478	0.462
- Word Overlap	0.694	0.471	0.416
- LCS	0.722	0.464	0.405
- Jaccard	0.682	0.473	0.425
- Word2Vec	0.693	0.479	0.429

Table 3: Ablation study of performance of features for SVM classification

We evaluated the taxonomy extraction procedure in two stages: firstly, we evaluated the results for the individual features using the greedy taxonomy construction procedure as described above. The results for this are given in Table 2, where we present the edge precision, recall and F-Measure based on the number of edges that match those in the taxonomy we are trying to create. In addition, we present the Fowlkes and Mallows metric (Fowlkes and Mallows, 1983) following the TExEval task as it "measures level by level how well a target taxonomy clusters similar nodes compared to a gold standard taxonomy" (Bordea et al., 2016).

In Table 2 the baseline features, Jaccard similarity (Section 3.3.), and Word2Vec similarities (Section 3.4. and 3.5.) are obtained in an unsupervised manner. The supervised results for the SVD and the combination of features with an SVM are also given in this table, where we held out the evaluation dataset and trained on all other datasets.

5. Conclusion

Taxonomy learning is a task that has up until now not achieved satisfactory results. In particular, research on terminology extraction using has not produced a higher F-Measure than for a very simple baseline of whether a substring is included in the term. While the TExEval results did show that these methods can improve in terms of the Modified Fowlkes and Mallows methods, this metric is very sensitive to the topology of the hierarchy, such as the depth of the network and the average number of children of a node. We present a new method that uses word embeddings in order to calculate the taxonomic relationship between two concepts in a directed manner. We compare this to the baseline used in the recent TExEval task, and show that while this metric itself does not outperform the baseline, it can easily be combined by means of a SVM classifier to produce a model that does outperform the baseline consistently in terms of F-Measure, proving a state-of-the-art result above the results of the TExEval task. While the improvements in F-Measure are significant and show the effectiveness of our methodology, we do not see the same improvements in the Fowlkes and Mallows score suggesting that the overall structure of our networks is not as good. This is likely due to the way the network has been constructed with a greedy algorithm and future work will look into the way that the network is constructed in order to produce a better overall taxonomy.

6. Bibliographical References

Astrakhantsev, N. (2016). ATR4S: Toolkit with state-of-the-art automatic terms recognition methods in scala. *arXiv preprint arXiv:1611.07804*.

Bordea, G., Lefever, E., and Buitelaar, P. (2016). SemEval-2016 task 13: Taxonomy extraction evaluation (TExEval-2). In *SemEval-2016*, pages 1081–1091. Association for Computational Linguistics.

Buitelaar, P., Cimiano, P., and Magnini, B. (2005). *Ontology learning from text: methods, evaluation and applications*. IOS press.

Buitelaar, P., Bordea, G., and Polajnar, T. (2013). Domain-independent term extraction through domain modelling. In *the 10th International Conference on Terminology and Artificial Intelligence (TIA 2013), Paris, France*.

Cleuziou, G. and Moreno, J. G. (2016). QASSIT at SemEval-2016 Task 13: On the integration of semantic vectors in pretopological spaces for lexical taxonomy acquisition. In *10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1315–1319.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Fellbaum, C. (1998). *WordNet*. John Wiley & Sons, Inc.

Fowlkes, E. B. and Mallows, C. L. (1983). A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383):553–569.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *Proceedings of the 2014 Conference of the Association for Computational Linguistics*, pages 1199–1209.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., and Schneider, L. (2002). Sweetening ontologies with DOLCE. *Knowledge engineering and knowledge management: Ontologies and the semantic Web*, pages 223–233.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

IEEE. (2017). 2017 IEEE taxonomy. Technical report, The Institute of Electrical and Electronics Engineers.

Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, (7):48–50.

Maitra, P. and Das, D. (2016). JUNLP at SemEval-2016 Task 13: A language independent approach for hypernym identification. In *10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1310–1314.

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The

automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Navigli, R., Velardi, P., and Faralli, S. (2011). A graph-based algorithm for inducing lexical taxonomies from scratch. In *IJCAI*, volume 2, page 2.

Panchenko, A., Faralli, S., Ruppert, E., Remus, S., Naets, H., Fairon, C., Ponzetto, S. P., and Biemann, C. (2016). TAXI at SemEval-2016 Task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. In *10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Petersen, K. B., Pedersen, M. S., et al. (2008). The matrix cookbook. *Technical University of Denmark*, 7(15):510.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. *arXiv preprint arXiv:1603.06076*.

Tan, L., Bond, F., and van Genabith, J. (2016). USAAR at SemEval-2016 Task 13: Hyponym endocentricity. In *10th International Workshop on Semantic Evaluation (SemEval-2016)*.

Tarjan, R. (1972). Depth-first search and linear graph algorithms. *SIAM journal on computing*, 1(2):146–160.

A Chinese Dataset with Negative Full Forms for General Abbreviation Prediction

Yi Zhang, Xu Sun
MOE Key Laboratory of Computational Linguistics, Peking University
School of Electronics Engineering and Computer Science, Peking University
zhangyi16, xusun@pku.edu.cn

Abstract

Abbreviation is a common phenomenon across languages, especially in Chinese. In most cases, if an expression can be abbreviated, its abbreviation is used more often than its fully expanded forms, since people tend to convey information in a most concise way. For various language processing tasks, abbreviation is an obstacle to improving the performance, as the textual form of an abbreviation does not express useful information, unless it's expanded to the full form. Abbreviation prediction means associating the fully expanded forms with their abbreviations. However, due to the deficiency in the abbreviation corpora, such a task is limited in current studies, especially considering general abbreviation prediction should also include those full form expressions that do not have general abbreviations, namely the negative full forms (NFFs). Corpora incorporating negative full forms for general abbreviation prediction are few in number. In order to promote the research in this area, we build a dataset for general Chinese abbreviation prediction, which needs a few preprocessing steps, and evaluate several different models on the built dataset. The dataset is available at `https://github.com/lancopku/Chinese-abbreviation-dataset`.

Keywords: Chinese abbreviation, negative full forms, conditional random field, long-short term memory

1. Introduction

Abbreviation processing mainly consists of three tasks, that is, abbreviation expansion, abbreviation recognition, and abbreviation prediction. Expanding the short form of an expression to its full form is called abbreviation expansion. Extracting the short form and full form pairs from the context is called abbreviation recognition. Abbreviation prediction refers to predicting the short form of an expression according to its full form. In this paper, we focus on the last task, i.e., abbreviation prediction. Abbreviation prediction plays an important role in various language processing tasks, because accurate abbreviation prediction will help improve performance. Sun et al. (2009) shows that better abbreviation prediction will help improve the performance of abbreviation recognition. Abbreviation prediction also benefits other tasks. For example, in an information retrieval (IR) system, a large number of the web pages contain only abbreviations. It will be helpful if we can estimate abbreviations of a query, because successful abbreviation prediction may improve the recall of IR systems as Sun et al. (2013a) showed. In addition, Yang et al. (2012) showed that Chinese abbreviation prediction can improve voice-based search quality.



Figure 1: Different cases of generating abbreviations

English abbreviations are usually formed as acronyms. Studies for English abbreviation proposed various heuristics for abbreviation prediction (Park and Byrd, 2001; Wren et al., 2002; Schwartz and Hearst, 2002). For example, use of initials, capital letters, syllable boundaries, stop words, etc. These studies performed well for English abbreviations. While Chinese abbreviations are quite different from English ones. Yang et al. (2012) showed that Chinese abbreviations are usually generated by three methods, reduction, elimination, and generalization. Characters are selected from the expanded full name to form the abbreviation. However, there are no general rules to convert a complete term into an abbreviation. As shown in Figure 1, an abbreviation may be generated using the first character and the last character. Sometimes, characters in the middle can be included while the last abbreviation takes the first two characters of the words. However, it is not necessary for Chinese abbreviations to take the first characters of words. They frequently take non-initial characters, like the last example in Figure 1. Chinese abbreviations are derived via a customary lexical process. Native speakers may associate a fully expanded term with its abbreviation by some intuition. But the process can not be adequately explained by any linguistic theory: Chang and Lai (2004) and Chang and Teng (2006).

Besides the irregularity of abbreviating phrases and terms, another main problem is caused by negative full forms. A word annotated with a negative full form means the word has no abbreviation at all. We usually recognize abbreviations or make abbreviation predictions in the text. Unfortunately, NFFs take up a large portion of Chinese words or phrases in the real world. With the strong noise, distinguishing the full forms with valid abbreviations is more difficult. This undoubtedly increases the difficulty of abbreviation prediction.

Many approaches have been proposed in the post studies. Sun et al. (2008) employed Support Vector Regression (SVR) for scoring abbreviation candidates. This method

outperforms the hidden Markov model (HMM) in abbreviation prediction. Yang et al. (2009) proposed to formulate abbreviation generation as a character tagging problem and the conditional random field (CRF) then can be used as the tagging model. Sun et al. (2009) combined latent variable model and global information to predict abbreviations. Zhang et al. (2016) used a recurrent neural networks to predict abbreviations for Chinese named entities.

However, most studies of abbreviation prediction focus on positive full form, which means a word has a valid abbreviation. Apparently, this implicit lab assumption is not practical. Nonetheless, we barely see studies that consider NFFs. One of the main reasons is the shortage of abbreviation prediction data with NFFs, which is one of the main issues this work tries to solve.

Apart from the annotation of a dataset with NFFs, we also conduct a few preprocessing steps to facilitate the usage of the dataset. Chinese does not insert spaces between words or word forms after morphological changes. Hence, most of the Chinese natural language processing methods assume a Chinese word segmenter is used in a preprocessing step to produce word-segmented Chinese sentences as inputs. There is no exception for abbreviation prediction. Given original texts, we should first recognize the boundaries of words. After segmentation, we annotate the part-of-speech information of phrases and terms, because the part-of-speech information can serve as features to help make abbreviation prediction.

This paper details how the dataset is created and evaluates some frequently used models on the abbreviation prediction task.

2. Dataset

2.1. Considerations

Commonness Our intention is to build a dataset with NFFs, so it can be widely used for general abbreviation prediction. This requires that dataset contains most frequently-used full forms regardless of whether or not the form has valid abbreviations. The data sources should be reliable and accredited. Thus, we extract long phrases and terms in popular Chinese natural language processing corpora, which include People's Daily corpora and SIGHAN word segmentation corpora.

Usability We also provide assisting information that is helpful for the abbreviation task in our dataset. Most existing methods treat abbreviation prediction as a sequence labeling problem. To make better tag predictions of characters, we usually need to extract some features.The word segmentation information and part-of-speech information are the most commonly used features. Unlike English, the smallest Chinese unit is a character rather than a word. There are no explicit boundaries between Chinese words. Since a full form usually can be segmented into several words and abbreviations often take characters from these words, segmentation information is most useful for abbreviation prediction. Another annotation is part-of-speech information. Many language processing tasks take part-of-speech information as features, including abbreviation prediction. The choice of characters which are used to form abbreviations may be related to their part-of-speech information. For example, if a full form composed of an adjective and a noun has an abbreviation, the abbreviation usually takes a character of the adjective and a character of the noun. The same issue applies to a full form composed of an adverb and a verb when the full form has an abbreviation.

Representativeness

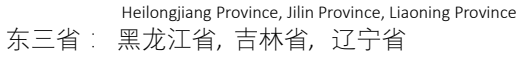


Figure 2: A special case for abbreviation.

As the dataset should be representative of the common construction of abbreviations, we do not include special and irregular abbreviations, which include words outside the full form. As shown in Figure 2, " 东三省 "represents three provinces of China. It is a special type of abbreviations, since an abbreviation could represent several different terms. Without some background knowledge, what the abbreviation stands for can not be understood. Sometimes the characters of the abbreviation are not taken from original characters of the full form and the sequence labeling method is no longer applicable for this case. This kind of "abbreviation" is more like a general name for some terms. We do not include these special abbreviations in our dataset.

2.2. Data Source

Our text is from People's Daily corpora and SIGHAN word segmentation corpora. We extract the long phrases and terms in the text. Then we classify the collected phrases and terms into two forms. One is the positive full form, which means the phrase or term has a valid abbreviation. Then its abbreviation is annotated. The other is the negative full form, which means the phrase or term does not have a general short form. Their abbreviations are "NULL". Samples of the data are shown in Figure 3.

As mentioned before, we annotate word segmentation information and part-of-speech information for every phrase or term. Word segmentation is a fundamental task in Chinese processing. Many practical Chinese processing applications rely on Chinese word segmentation. Part-of-speech information is often used as features for further prediction. Most methods formulate these tasks as a sequence labeling problem. Various models achieved good performance on these tasks and some open source tools have been published for use. We used ICTCLAS, one of the best Chinese Lexical analyzers, to label the segmentation and part-of-speech information.

2.3. Statistics

We build a dataset that is made up of phrases and terms. There are 10,786 full forms in this dataset, including 8,015 positive full forms and 2,661 negative full forms. The phrases contain noun phrases, verb phrases, organization names, location names, and so on. The distribution is shown in Table 2. For experiments, we randomly sampled 7,551 samples as the training set, 1078 samples as the development set and 2,157 samples as the testing set. We calculate the number of the words and characters (including

Positive / Negative Full Forms	Abbreviation
中国科学院 (Chinese Academy of Sciences)	中科院
二氯化碳酰 (Phosgene)	NULL
澳大利亚网球公开赛 (Australia Open Tennis Championships)	澳网
新加坡航空公司 (Singapore Airlines)	新航
独生子女家庭 (One-child family)	NULL
第一次世界大战 (The First World War)	一战
冠状动脉粥样硬化心脏病 (Coronary atherosclerotic heart disease)	冠心病
办公自动化 (office automation)	NULL

Figure 3: Samples of the collected data with NFFs. The "NULL" means no valid abbreviation.

Full Forms	
total entries	10,786
NFFs	2,661
total words	30,100
distinct words	8,293
total characters	60,877
distinct characters	2,557
average word length	5.644
Abbreviations	
total characters	23,077
distinct characters	1,687
average abbreviation length	2.140

Table 1: The statistics of the data. The results are count for full forms and abbreviations separately.

duplicates) in the data. We also count the number of distinct words and distinct characters. Then total characters of full forms divided by total entries is the average full form length. The average abbreviation length can be calculated in a similar way.

Category	Portion(%)
Noun Phrase	52.01%
Organization Name	26.84%
Verb Phrase	13.72%
Location Name	5.28%
Person Name	0.32%
Others	1.80%

Table 2: Distribution of the full forms in the data.

3. Models

3.1. CRF

Tsuruoka et al. (2005) formalized the process of abbreviation prediction as a sequence labeling problem. Each

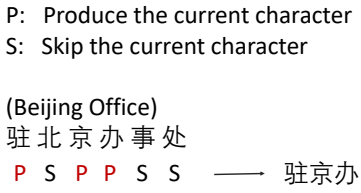


Figure 4: Chinese abbreviation generation as a sequential labeling problem.

character in the expanded form is tagged with a label, y $\in \{P, S\}$, where the label P produces the current character and the label S skips the current character. In Figure 4, the abbreviation is generated using the first character, skipping the flowing character and then using the subsequent two characters. Because our task is general abbreviation prediction, we add another label "N" to the tag set to label the characters in negative full forms. A number of recent studies have investigated the use of machine learning techniques. Traditional models like MEMM, peceptron and conditional random field perform well on such sequence labeling tasks. We use the well-known conditional random field (CRF) proposed by Lafferty et al. (2001) for sequential labeling.

We use features as follows:

- character feature : Input characters x_{i-1}, x_i and x_{i+1}
- character bi-gram : The character bigrams starting at $(i-2)\cdots i$.
- Numeral: Whether or not the x_i is a numeral.
- Organization name suffix: Whether or not the x_i is a suffix of traditional Chinese organization names.
- Location name suffix: Whether or not the x_i is a suffix of traditional Chinese location names.
- Word segmentation information: After the word segmentation step, whether or not the x_i is the beginning character of a word.
- Part-of-speech information: The part-of-speech tag information of x_i .

In our abbreviation prediction task, the input sequence $\boldsymbol{x}$ represents characters of a full form and output sequence $\boldsymbol{y}$ represents symbolic labels based on abbreviations. The probability is defined as follows:

$$P(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = \frac{exp[\boldsymbol{w}^T \boldsymbol{f}(\boldsymbol{y}, \boldsymbol{x})]}{\sum_{\forall \boldsymbol{y}'} exp[\boldsymbol{w}^T \boldsymbol{f}(\boldsymbol{y}', \boldsymbol{x})]} \quad (1)$$

where $\boldsymbol{w}$ is the weight vector and $\boldsymbol{f}$ is the mapping function.

Given a training set that consists of n labeled sequences $(\boldsymbol{x}^i, \boldsymbol{y}^i)$ for $i = 1 \cdots n$, the objective function is:

$$L(w) = \sum_{i=1}^{n} logP(\boldsymbol{y}^i|\boldsymbol{x}^i, \boldsymbol{w}) - R(\boldsymbol{w}) \quad (2)$$

where the second term is the L_2 regularizer.

Method	Discriminate Acc(%)	All-Acc(%)	Char-Acc(%)
Heuristic System	73.20	25.77	65.79
Perc	87.48	54.89	87.02
MEMM	86.97	50.16	85.92
CRF-ADF	87.80	56.69	87.20
BLSTM	91.47	56.98	81.73

Table 3: Results on comparing different methods on generalized abbreviations.

3.2. BLSTM

As mentioned above, traditional methods depend heavily on features which need to be designed elaborately. Nowadays, more and more research focuses on neural networks, such as recurrent neural network (RNN), convolutional neural network (CNN) and some variants of RNN. These neural network models can extract features automatically. In natural language processing, traditional RNNs usually take the previous state h_{t-1} and the embedding x_t as the t-th input to calculate current state h_t. Formally, we have

$$h_t = f(W \cdot x_t + V \cdot h_{t-1} + b_h) \tag{3}$$

where W and V are weight matrices, respectively. b_h is a bias term and f is a non-linear activation function.

In theory, RNN can keep a memory of previous information. However, it was difficult to train RNNs to capture longterm dependencies because the gradients tend to either vanish or explode. Therefore, some sophisticated variants of RNN were proposed. Long-short term memory units are proposed in Hochreiter and Schmidhuber (1997). This model introduces a gating mechanism, which controls the proportions of information to forget and to pass on to the next time step. Concretely, the LSTM-based recurrent neural network comprises four components: an input gate i_t, a forget gate f_t, an output gate o_t, and a memory cell c_t. LSTM memory cell is implemented as following:

$$\begin{aligned} f_t &= \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \\ i_t &= \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + bi) \\ \tilde{C_t} &= tanh(W_C \cdot x_t + U_C \cdot h_{t-1} + b_C) \\ C_t &= f_t \otimes C_{t-1} + i_t \otimes \tilde{C_t} \\ o_t &= \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \\ h_t &= o_t \otimes tanh(C_t) \end{aligned} \tag{4}$$

LSTM can solve the long-distance dependencies problem to some extent. However, the LSTM's hidden state h_t takes information only from the past, knowing nothing about the future. An elegant solution whose effectiveness has been proven by previous work (Dyer et al., 2015) is bi-directional LSTM(BLSTM). The basic idea is to present each sequence forwards and backwards to two separate hidden states to capture past and future information, respectively. Then the two hidden states are concatenated to form the final output. In this paper, we employ a bi-directional LSTM, which could capture the contextual information of the current input, to predict the abbreviations of full terms. Since we give a specific segmentation tag and a pos tag for every character, each segmentation tag and pos tag can be mapped to a real-valued vector by looking up their own embedding tables. These embeddings and character embeddings are all initialized randomly. At current time-step t, the character embedding, segmentation tag embedding and pos tag embedding are concatenated as the input x_t. Embeddings of segmentation tag and pos tag are both 20-dimensional. Character embedding is 50-dimensional. The hidden layer size of BLSTM is 200, 100 for forward LSTM and 100 for backward LSTM.

4. Evaluation

4.1. Evaluation Metrics

For evaluating abbreviation prediction quality, the systems are evaluated using the following two metrics:

- **Discriminate accuracy**: The discriminate accuracy checks the accuracy of discriminating positive and negative full forms, without comparing the generated abbreviations with the gold-standard abbreviations.
- **All-match accuracy (All-Acc)**: The number of correct outputs (i.e., label strings) generated by the system divided by the total number of full forms in the test set.
- **Character accuracy (Char-Acc)**: The number of correct labels (i.e., a classification on a character) generated by the system divided by the total number of characters in the test set.

4.2. Simple Heuristic Baseline System

The simple heuristic system means always choosing initial characters of words in the segmented full form. This is because the most natural abbreviating heuristic is to produce the first character of each word in the original full form. This is just the simplest baseline.

4.3. Evaluation

To study the performance of other machine learning models, we also implement other well known sequential labeling models, including maximum entropy Markov models (MEMMs) (McCallum et al., 2000) and averaged perceptrons (Perc) (Collins, 2002). Besides these traditional models, we also implement a bidirectional LSTM(BLSTM) to evaluate the performance of neural networks on this task.

The experimental results are shown in Table 3. In the table, the overall accuracy is most important and it means the final accuracy achieved by the systems in generalized abbreviation prediction with NFFs. For the completeness of experimental information, we also show the discriminate accuracy. The CRF model outperforms the MEMM

and averaged perceptron models. The CRF model achieves best overall character accuracy. BLSTM outperforms other models in both discriminate accuracy and all-match accuracy. However, training a neural network always needs a large amount of data. With a dataset that is not so large, the ability of a neural network may be limited.

5. Conlusions and Future Work

This paper proposes a novel abbreviation prediction dataset with NFFs. Different machine learning methods are evaluated on this general abbreviation task. LSTM shows competitive performance in this task. However, neural networks usually need large data for training. The related corpora are not sufficient and researches for general abbreviation prediction using neural networks are encouraged.

6. Acknowledgements

This work was supported in part by National Natural Science Foundation of China (No. 61673028), National High Technology Research and Development Program of China (863 Program, No. 2015AA015404), and an Okawa Research Grant (2016). Email correspondence to Xu Sun.

7. Bibliographical References

Chang, J.-S. and Lai, Y.-T. (2004). A preliminary study on probabilistic models for chinese abbreviations. In *Proceedings of the Third SIGHAN workshop on Chinese language learning*, pages 9–16.

Chang, J.-S. and Teng, W.-L. (2006). Mining atomic chinese abbreviations with a probabilistic single character recovery model. *Language Resources and Evaluation*, 40(3-4):367–374.

Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. *arXiv preprint arXiv:1505.08075*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

McCallum, A., Freitag, D., and Pereira, F. C. (2000). Maximum entropy markov models for information extraction and segmentation. In *Icml*, volume 17, pages 591–598.

Park, Y. and Byrd, R. J. (2001). Hybrid text mining for finding abbreviations and their definitions. In *Proceedings of the 2001 conference on empirical methods in natural language processing*, pages 126–133.

Schwartz, A. S. and Hearst, M. A. (2002). A simple algorithm for identifying abbreviation definitions in biomedical text. In *Biocomputing 2003*, pages 451–462. World Scientific.

Sun, X. and Wang, H. (2006). Chinese abbreviation identification using abbreviation-template features and context information. In *Computer Processing of Oriental Languages. Beyond the Orient: The Research Challenges Ahead, 21st International Conference, ICCPOL 2006, Singapore, December 17-19, 2006, Proceedings*, pages 245–255.

Sun, X., Wang, H.-F., and Wang, B. (2008). Predicting chinese abbreviations from definitions: An empirical learning approach using support vector regression. *Journal of Computer Science and Technology*, 23(4):602–611.

Sun, X., Okazaki, N., and Tsujii, J. (2009). Robust approach to abbreviating terms: A discriminative latent variable model with global information. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 905–913. Association for Computational Linguistics.

Sun, X., Li, W., Meng, F., and Wang, H. (2013a). Generalized abbreviation prediction with negative full forms and its application on improving chinese web search. In *IJCNLP*, pages 641–647.

Sun, X., Okazaki, N., Tsujii, J., and Wang, H. (2013b). Learning abbreviations from chinese and english terms by modeling non-local information. *ACM Trans. Asian Lang. Inf. Process.*, 12(2):5:1–5:17.

Tsuruoka, Y., Ananiadou, S., and Tsujii, J. (2005). A machine learning approach to acronym generation. In *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, pages 25–31. Association for Computational Linguistics.

Wren, J. D., Garner, H. R., et al. (2002). Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym-definition dictionaries. *Methods of information in medicine*, 41(5):426–434.

Yang, D., Pan, Y.-c., and Furui, S. (2009). Automatic chinese abbreviation generation using conditional random field. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 273–276. Association for Computational Linguistics.

Yang, D., Pan, Y.-C., and Furui, S. (2012). Vocabulary expansion through automatic abbreviation generation for chinese voice search. *Computer Speech & Language*, 26(5):321–335.

Zhang, L., Li, L., Wang, H., and Sun, X. (2014a). Predicting chinese abbreviations with minimum semantic unit and global constraints. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1405–1414.

Zhang, L., Wang, H., and Sun, X. (2014b). Coarse-grained candidate generation and fine-grained re-ranking for chinese abbreviation prediction. In *Proceedings of the 2014*

Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1881–1890.

Zhang, Q., Qian, J., Guo, Y., Zhou, Y., and Huang, X. (2016). Generating abbreviations for chinese named entities using recurrent neural network with dynamic dictionary. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 721–730.

Korean TimeBank Including Relative Temporal Information

Chae-Gyun Lim[†], Young-Seob Jeong[§], Ho-Jin Choi[†]
[†]School of Computing, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon 34141, South Korea
{rayote, hojinc}@kaist.ac.kr
[§]Department of BigData Engineering, Soonchunhyang University
22 Soonchunhyang-ro, Asan-si, Chungcheongnam-do 31538, South Korea
bytecell@sch.ac.kr

Abstract

Since most documents have temporal information that can be a basis for understanding the context, the importance of temporal information extraction researches is steadily growing. Although various attempts have been made to extract temporal information from researchers internationally, it is difficult to apply them to other languages because they are usually targeted at specific languages such as English. Several annotation languages and datasets had been proposed for the studies of temporal information extraction on Korean documents, however, the representation of relative temporal information is not enough to maintain it explicitly. In this paper, we propose a concept of relative temporal information and supplement a Korean annotation language to represent new relative expressions, and extend an annotated dataset, Korean TimeBank, through the revised language. We expect that it is possible to utilize potential features from the Korean corpus by clearly annotating relative temporal relationships and to use well-refined Korean TimeBank in future studies.

Keywords: Relative temporal information, Korean TimeML, Relative temporal relationship, Korean TimeBank

1. Introduction

Temporal information extraction is one of the important research fields in natural language processing, and it is necessary to understand the temporal context in real-world applications such like question answering or conversation system providing a good quality of service. As an international activity related to temporal information extraction research, there is a shared task called TempEval which is part of SemEval (UzZaman et al., 2012). According to the description of TempEval, the temporal information extraction consists of three separate processes as follows—*timex3*, *event*, and *tlink* extraction. TempEval has provided annotated datasets for performance comparison by adopting TimeML (Pustejovsky et al., 2003), which is one of well-known markup languages for annotating temporal information. However, since the dataset is basically constructed underlying English documents, it is not applicable to the studies of temporal information extraction in other languages. Preceding the construction of a corpus consisting of the documents in a particular language is very important because the linguistic characteristics inherent in the language have a significant impact to discover temporal relationships (Jeong, 2016).

For extracting temporal information in Korean, there are several previous researches. The Korean TimeML (Im et al., 2009), which is adopt morpheme-level stand-off annotation scheme and addressed some language-specific issues on Korean, was proposed. In 2015, the revised version of Korean TimeML was proposed to overcome several limitations such as applying the lunar calendar and character-level annotation scheme (Jeong et al., 2015). Moreover, they also published a corpus, namely Korean TimeBank, including a bunch of Korean documents annotated by the revised Korean TimeML.

In this paper, we propose a concept to find relative temporal information from Korean documents and complement the previous version of Korean TimeML to be able to clearly annotate that relative information. Additionally, we annotate many Korean documents to increase the size of the Korean TimeBank including the relative temporal information, and refine the annotated files in the corpus to improve the quality. Since the relative relationships between temporal entities are potentially useful information, we believe that this extended version of the Korean TimeBank will contribute to broad areas of research.

The rest of this paper is organized as follows. Section 2 explains a concept of relative temporal information and its annotation. Section 3 describes the extended version of Korean TimeBank in detail, and Section 4 concludes.

2. Relative Temporal Information

2.1. Concept of Annotating the Relative Temporal Information

The relative linkage between temporal entities—time expressions and events—is potentially important information when determining the temporal context. However, the *timex3* tag, which is treated as a target representation of annotation work in Korean TimeML, must store the exact value of date and time into '*value*' attribute. At this time, the date or time information is always converted to the absolute value (i.e., normalization) to store the *value* attribute even though that expression has a differential value from a specific reference date or time. For example, let us suppose there is a time expression 'a day ago' and its reference date is '2017-09-30'. We can obviously know that there are two *timex3* tags as follows.

```
<timex3 id="t1" value="2017-09-30"/>
<timex3 id="t2" text="a day ago"
value="2017-09-29"/>
```

Table 1: Examples of *relValue* attribute

Example in Korean	Meaning in English	relVaule
5년 4개월 후	after 5 years and 4 months	+P5Y4M
2주 전	2 weeks ago	-P2W
1시간 30분 25초 전	an hour, 30 minutes and 25 seconds ago	-PT1H30M25S
2년 10개월 15일 10시간 20분 30초 후	after 2 years 10 months 15 days 10 hours 20 minutes 30 seconds	+P2Y10M15DT10H20M30S
이번 달	this month	P0M

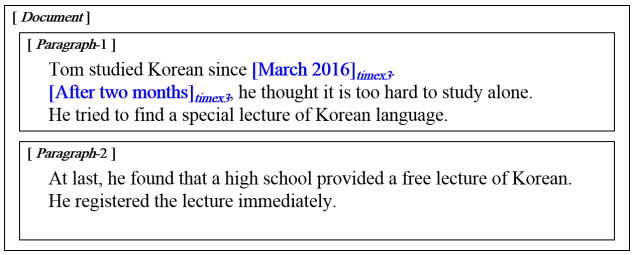


Figure 1: An example of two *timex3* entities and a relation of *RT2* instead of other types.

According to the existing Korean TimeML, the *timex3* tag 't2' has normalized value from the reference tag 't1', however, it could not maintain that the meaning of 'a day ago' is 'one day earlier than the reference date'. We need to prevent this kind of information loss by separating the relative value into additional attribute.

In addition, we divide the types of relative temporal relationships among entities into four levels as follows.

- *RT1 (In-Sentence)*: a relative temporal relation of two entities which are existed in a 'single sentence'.
- *RT2 (Between-Sentences)*: a relative temporal relation of two entities between 'two consecutive sentences' where an entity is existed in a sentence first and another is in the next sentence.
- *RT3 (In-Paragraph)*: a relative temporal relation of two entities in a 'paragraph' that consists of three or more sentences.
- *RT4 (In-Document)*: a relative temporal relation of two entities in a 'whole document' that consists of two or more paragraphs.

When a temporal relation is annotated by one of the four levels above, a relationship type having a small tolerance range has a higher priority than others. For instance, let us consider a given document including only two short paragraphs as shown in Figure 1. In the first paragraph of this example, there are three sentences and two *timex3* entities—'March 2016' and 'after two months'. They obviously have a temporal relationship over two consecutive sentences. Therefore, the relation should be *RT2* instead of *RT3* or *RT4* due to the relation's range.

2.2. Changes in the Annotation Language

We basically annotated the Korean temporal information according to the structure of the existing Korean TimeML (Jeong et al., 2015). To accurately represent the differential value of relative temporal information, we add a new attribute '*relValue*' to the *timex3* tag in this annotation language. Table 1 shows some examples of the *relValue* attribute. The *relValue* refers the ISO-8601 standard to express an amount of the difference—similar to the value of 'DURATION' type of the *timex3*. And there is a prefixed symbol either '+' or '-' depends on the direction of the relative relationships between temporal entities. In other words, the *relValue* should be start with '+' symbol if the meaning of text is later than the reference date, and vice versa. However, in a special case such as the last example in Table 1, there is no symbol to prefix because the meaning of text exactly pointed the specific date or time.

We believe that this work will contribute to reasoning of temporal relationship according to the relative temporal information. (Gennari and Vittorini, 2016) explained a reasoning system based on a service-oriented architecture (SOA), which is called SOA-based Qualitative Temporal Reasoner (SQTR). SQTR had applied knowledge representation techniques and tools to improve the performance of temporal reasoning on the annotated data. Similar to this work, in the perspective of knowledge representation, our work also can help to grasp relative relationships among entities.

3. Extended Korean TimeBank

Korean TimeBank v2.0 is an extended version of the previous Korean TimeBank (Jeong et al., 2016), which had been introduced on the last LREC 2016. In this version of dataset, we adopt new concept of relative temporal information by adding new *relValue* attribute of *timex3* tags and annotating relative relationships among them. Also, we continuously refined the annotated documents to improve the quality of corpus.

The statistics of Korean TimeBank v2.0 is summarized in Table 2. Compared with the previous version of Korean TimeBank, the total number of documents and sentences increased by about 121.9% and 52.7%. In addition, the number of *timex3*, *event*, *makeinstance*, *tlink* tags increased by about 35.7%, 52.3%, 51.9% and 23.8%, respectively. The annotation work for the corpus was performed by two annotators and one supervisor, and the supervisor mediated and led the annotators' consent when they had different opinions for an annotation result.

```
<?xml version="1.0" encoding="utf-8"?>
<doc id="부산대학교.txt" url="http://ko.wikipedia.org/wiki/%EB%B6%80%EC%82%B0%EB%8C%80%ED%95%99%EA%B5%90" category="대한민국의국립대학" date="2014-07-30T21:39">
  <contents>
    <sentence id="0">좌표: 북위 35° 14′ 2.33″ 동경 129° 4′ 45.52″</sentence>
    <sentence id="1">자료, 주한미대사관과 공동파트너십 체결에 의한 'Window on America'자료, 그 외 국제교육학프로그램자료, 독서치료프로그램자료 등 전문화된 자료를 별도 코너를
                    마련하여 비치하고 있다.</sentence>
          ⋮
    <sentence id="109">학생들은 당시 교수권 남용이라며 국가인권위원회에 진정을 냈고 최근 인권위가 대학 측에 최 교수에 대한 징계조치를 권고했다.</sentence>
  </contents>
  <timeAnnotation>
    <annotationInfo sentence_id="0">
      <text>좌표: 북위 35° 14′ 2.33″ 동경 129° 4′ 45.52″</text>
      <tag />
    </annotationInfo>
    <annotationInfo sentence_id="1">
      <text>자료, 주한미대사관과 공동파트너십 체결에 의한 'Window on America'자료, 그 외 국제교육학프로그램자료, 독서치료프로그램자료 등 전문화된 자료를 별도 코너를 마련하
          여 비치하고 있다.</text>
      <tag>
        <event id="TIME_S1_e0" begin="0" end="1" text="체결" class="OCCURRENCE" e_begin="3" e_end="3" />
        <event id="TIME_S1_e1" begin="0" end="2" text="마련하" class="OCCURRENCE" e_begin="17" e_end="17" />
        <event id="TIME_S1_e2" begin="0" end="2" text="비치하" class="OCCURRENCE" e_begin="18" e_end="18" />
        <makeinstance id="TIME_S1_ei0" eventID="TIME_S1_e0" POS="NOUN" tense="NONE" polarity="POS" />
        <makeinstance id="TIME_S1_ei1" eventID="TIME_S1_e1" POS="VERB" tense="NONE" polarity="POS" />
        <makeinstance id="TIME_S1_ei2" eventID="TIME_S1_e2" POS="VERB" tense="PRESENT" polarity="POS" />
      </tag>
    </annotationInfo>
          ⋮
    <annotationInfo sentence_id="109">
      <text>학생들은 당시 교수권 남용이라며 국가인권위원회에 진정을 냈고 최근 인권위가 대학 측에 최 교수에 대한 징계조치를 권고했다.</text>
      <tag>
        <timex3 id="TIME_S109_t0" type="DATE" begin="0" end="1" text="당시" e_begin="1" e_end="1" quant="NONE" relValue="P0Y" />
        <event id="TIME_S109_e0" begin="2" end="2" text="의" class="STATE" e_begin="3" e_end="3" />
        <event id="TIME_S109_e1" begin="0" end="1" text="진정" class="OCCURRENCE" e_begin="5" e_end="5" />
        <event id="TIME_S109_e2" begin="0" end="2" text="권고하" class="I_ACTION" e_begin="15" e_end="15" />
        <makeinstance id="TIME_S109_ei0" eventID="TIME_S109_e0" POS="VERB" tense="NONE" polarity="POS" />
        <makeinstance id="TIME_S109_ei1" eventID="TIME_S109_e1" POS="NOUN" tense="PAST" polarity="POS" />
        <makeinstance id="TIME_S109_ei2" eventID="TIME_S109_e2" POS="VERB" tense="PAST" polarity="POS" />
        <tlink id="TIME_tl200" timeID="TIME_S108_t0" relatedToTime="TIME_S109_t0" relType="IDENTITY" comment="RT2 (Between-Sentences)" />
        <tlink id="TIME_tl201" timeID="TIME_S108_t0" relatedToTime="TIME_S109_t0" relType="IDENTITY" comment="RT2 (Between-Sentences)" />
        <tlink id="TIME_tl202" timeID="TIME_S109_t0" relatedToEventInstance="TIME_S109_ei1" relType="INCLUDES" />
      </tag>
    </annotationInfo>
  </timeAnnotation>
</doc>
```

Figure 2: An example of annotated document in the Korean TimeBank v2.0.

In this version of dataset, there are 184 *timex3* tags which are specified the *relValue* attribute. And there are 333 *tlink* tags which are directly connected to those *timex3* tags including *relValue*. Table 3 summarizes the statistics of relative temporal relations in the Korean TimeBank.

As an example of annotation work, Figure 2 shows a part of a sample annotated document in the Korean TimeBank. We used stand-off scheme that annotated results of a document write into a separated file by XML format. Figure 3 shows the XML schema to store the annotated document of Korean TimeBank. Each block means an XML node where the header text is the name of node and the list of items is attributes of the node. An arrowed line means a connection from a parent node to its child node. Small text nearby the arrow, either '1' or '*' symbol, is a cardinality of the node where '1' means a single child node must be appeared and '*' means any number of child nodes is allowed. Since we create an annotated document for a corresponding document individually, only one *doc* node is existed as a root node. For each sentence of the given document, *sentence* node is created and it will be a child of *contents* node. Also, *annotationInfo* nodes are stored corresponding to the sentences. At the bottom of this diagram, there are four types of tags we used to—i.e., *timex3*, *event*, *makeinstance*, and *tlink*. The temporal information is stored in the annotation document by these four kinds of tag nodes.

Table 2: Summary of the Korean TimeBank v2.0

Name	Count
# of documents	2,393
# of sentences	6,189
# of empty sentences (no tags)	112
# of words	78,327
Avg. # of words per sentence	12.65584
# of morps	188,687
Avg. # of morps per sentence	30.48748
# of *timex3* tags	3,462
# of *event* tags	17,543
# of *makeinstance* tags	17,583
# of *tlink* tags	4,933

Table 3: Relative Temporal Relations in the Korean TimeBank v2.0

Relation Type	Count	Proportion
RT1 (In-Sentence)	242	72.67%
RT2 (Between-Sentences)	34	10.21%
RT3 (In-Paragraph)	12	3.60%
RT4 (In-Document)	45	13.51%
Total	333	100%

4. Conclusion

In this paper, we supplemented the annotation language to reflect relative temporal information and presented an extended dataset—Korean TimeBank v2.0. Not only the number of documents and sentences in the corpus has significantly increased, but the concept of relative temporal information has also been annotated additionally, so we expect this Korean TimeBank to be useful in various applications of temporal information extraction.

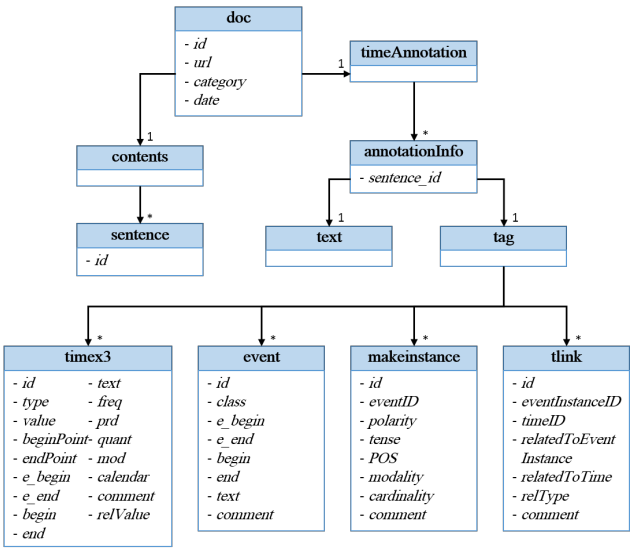


Figure 3: XML schema of annotated document.

5. Acknowledgements

This work was supported by the Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP) (No.2013-0-00131, Development of Knowledge Evolutionary WiseQA Platform Technology for Human Knowledge Augmented Services; and No.2017-0-00868, Development of Conversational Solution for Intelligent Chat Services Based on Pragmatic and Context Analysis of Dialogues).

6. Bibliographical References

Gennari, R. and Vittorini, P. (2016). Qualitative temporal reasoning can improve on temporal annotation quality: How and why. *Applied Artificial Intelligence*, 30(7):690–719.

Im, S., You, H., Jang, H., Nam, S., and Shin, H. (2009). Ktimeml: specification of temporal and event expressions in korean text. In *Proceedings of the 7th Workshop on Asian Language Resources*, pages 115–122. Association for Computational Linguistics.

Jeong, Y.-S., Kim, Z. M., Do, H.-W., Lim, C.-G., and Choi, H.-J. (2015). Temporal information extraction from korean texts. In *Proceedings of the 19th SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, pages 279–288.

Jeong, Y.-S., Joo, W.-T., Do, H.-W., Lim, C.-G., Choi, K.-S., and Choi, H.-J. (2016). Korean timeml and korean timebank. In *Proceedings of the 10th edition of the Language Resources and Evaluation Conference (LREC)*, pages 356–359.

Jeong, Y.-S. (2016). *Temporal Information Extraction from Korean Texts*. Doctoral thesis, School of Computing, Korea Advanced Institute of Science Technology (KAIST).

Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003). Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

UzZaman, N., Llorens, H., Allen, J., Derczynski, L., Verhagen, M., and Pustejovsky, J. (2012). Tempeval-3: Evaluating events, time expressions, and temporal relations. *arXiv preprint arXiv:1206.5333*.

Mining Biomedical Publications With The LAPPS Grid

Nancy Ide*, Keith Suderman*, and Jin-Dong Kim**
*Department of Computer Science, Vassar College
Poughkeepsie, New York USA
{ide, suderman}@cs.vassar.edu
*Database Center for Life Science (DBCLS), Univ. of Tokyo Kashiwa-no-ha
Wakashiba, Kashiwa-shi, Chiba, Japan
jdkim@dbcls.rois.ac.jp

Abstract

It is widely recognized that the ability to exploit Natural Language Processing (NLP) text mining strategies has the potential to increase productivity and innovation in the sciences by orders of magnitude, by enabling scientists to pull information from research articles in scientific disciplines such as genomics and biomedicine. The Language Applications (LAPPS) Grid is an infrastructure for rapid development of natural language processing applications (NLP) that provides an ideal platform to support mining scientific literature. Its Galaxy interface and the interoperability among tools together provide an intuitive and easy-to-use platform, and users can experiment with and exploit NLP tools and resources without the need to determine which are suited to a particular task, and without the need for significant computer expertise. The LAPPS Grid has collaborated with the developers of PubAnnotation to integrate the services and resources provided by each in order to greatly enhance the user's ability to annotate scientific publications and share the results. This poster/demo shows how the LAPPS Grid can facilitate mining scientific publications, including identification and extraction of relevant entities, relations, and events; iterative manual correction and evaluation of automatically-produced annotations, and customization of supporting resources to accommodate specific domains.

Keywords: biomedical text mining, LAPPS Grid, PubAnnotation, Open Advancement evaluation, reproducibility

1. Introduction

Keeping up with the ever-expanding flow of data and publications is untenable and poses a fundamental bottleneck to scientific progress. The global research community generates approximately 2.5 million new scholarly papers per year (in English only); a new research paper is published every 12 seconds. Current search technologies typically find many relevant documents, but they do not extract and organize the information content of these documents or suggest new scientific hypotheses based on this organized content.

It is widely recognized that the ability to exploit Natural Language Processing (NLP) text mining strategies has the potential to increase productivity and innovation in the sciences by orders of magnitude, by enabling scientists to pull information from research articles in scientific disciplines such as genomics and biomedicine. The application of NLP techniques can also lead to hypotheses and discoveries for which there is "hidden" (not explicitly stated) evidence in the research literature and enable linking extracted information to form new facts or new hypotheses to be explored further. These methods enable scientists to rapidly identify publications relevant to their own research as well as make scientific discoveries by scouring hundreds of research papers for associations and connections (such as between drugs and side effects, or genes and disease pathways) that humans reading each paper individually might not notice.

Up to now, the use of NLP technologies has required considerable skill in the field. However, recent development of environments for constructing customizable NLP applications has opened the door for scientists to exploit NLP technologies for discovering and mining information from massive bodies of scientific publications such as those found in PubMed, PLoS, Web of Science, etc.

The Language Applications (LAPPS) Grid[1] (Ide et al., 2014) provides an infrastructure for rapid development of natural language processing applications (NLP) by providing access to a wide range of tools and making them both syntactically and semantically interoperable. The LAPPS Grid uses the Galaxy platform[2] (Giardine et al., 2005), originally developed for use by genomics researchers with little computational expertise, as its workflow engine. The Galaxy interface and the interoperability among tools together provide an intuitive and easy-to-use platform that enables users to experiment with and exploit NLP tools and resources without the need to determine which are suited to a particular task, and without the need for significant computer expertise.

We demonstrate how the LAPPS Grid can be used to rapidly and easily develop out-of-the-box workflows for information and relation extraction and adapt them to data for specific disciplines, for example by providing means to rapidly bootstrap custom dictionaries and gazetteers. We also show how users can employ a cycle of automatic annotation and manual correction to create more robust annotations, and exploit state-of-the-art evaluation services to determine the optimal tool and resource configuration for a given task. Finally, we demonstrate the use of the LAPPS Grid access major scientific publications databases stored in the cloud as well as materials and facilities available through the PubAnnotation portal[3], and query them with Apache Solr for data discovery and mining.

[1] http://www.lappsgrid.org
[2] http://galaxyproject.org
[3] http://pubannotation.org

2. LAPPS Grid Overview

The US National Science Foundation SI[2]-funded Language Applications (LAPPS) Grid[4] (Ide et al., 2014) was originally developed to facilitate rapid development of Natural Language Processing (NLP) applications. From the outset, it was designed to enable sophisticated analyses while hiding the complexities associated with the underlying infrastructure, with an eye toward serving the needs of researchers and students who may not have substantial computational expertise. The LAPPS Grid provides seamless access to a wide range of NLP tools, including popular public tools such as StanfordNLP, OpenNLP, NLTK, LingPipe, as well as tools and modules available in GATE and various UIMA platforms; machine learning facilities; and a state-of-the-art Open Advancement (OA) evaluation system developed at Carnegie Mellon University and used in the development of IBM's Jeopardy-winning Watson. It also provides access to several mainstream resources, including holdings of the Linguistic Data Consortium (LDC). Most crucially, the LAPPS Grid provides for using all of these tools and resources *interoperably* in a seamless "plug-and-play" workflow environment, thereby eliminating the effort required to harmonize input and output formats to use a set of tools together. The LAPPS Grid is also flexible and extensible, as tools and datasources are routinely added to the LAPPS Grid as required by various researchers and projects.

The LAPPS Grid is open source (Apache 2.0 license) and free for use by anyone, and can be run from the web, on a user's laptop or desktop, in the cloud, or as a self-contained docker image when it is necessary to protect sensitive data or no network connection is available. We have also developed means to provide secure access to licensed data and software where necessary from within the LAPPS Grid, as well as to allow for user authentication and identification through identity providers (e.g., InCommon[5]) that provide a secure and privacy-preserving trust fabric for their members.

The LAPPS Grid provides cloud-based computation via the NSF-funded Extreme Science and Engineering Discovery Environment (XSEDE)[6] and the associated Jetstream[7] cloud environment. These resources allow users to create virtual machines configured as specialized versions of the LAPPS Grid on the remote resource. If necessary, access can be given to specified domain servers holding secure data.

2.1. Galaxy workflow engine

The LAPPS Grid's adaptation of the Galaxy workflow engine provides an intuitive and easy-to-use interface and data management system. The Galaxy project[8] started in 2005 to create a system enabling biologists without informatics expertise to perform computational analysis through the web. It has since been widely adopted within the life sciences community.

[4]http://www.lappsgrid.org

[5]http://incommon.org

[6]https://www.xsede.org

[7]https://jetstream-cloud.org

[8]http://galaxyproject.org

Galaxy is an open-source application[9] that includes tool integration and history capabilities together with a workflow system for building automated multi-step analyses, a visualization framework including visual analysis capabilities, and facilities for sharing and publishing analyses (Goecks et al., 2010; Afgan et al., 2016). It is accessed through a graphical interface where data inputs and computational steps are selected from dynamic menus, and results are displayed in plots and summaries that encourage interactive workflows and the exploration of hypotheses.

With funding from the National Science Foundation[10], we are working with the Galaxy development team in order to adapt the system to our domain and apply Galaxy's powerful analytic and visualization software to information extracted from texts using the LAPPS Grid without leaving the platform. The ultimate goal of this collaboration is to both enhance the capabilities required to support NLP application development and contribute to the expansion of Galaxy to domains outside the life sciences, which is a current goal of the Galaxy project.

2.2. Comparison to existing platforms for biomedical text analysis

The two most well-known platforms that currently support scientific literature mining are the UIMA-based U-Compare (Kano et al., 2008) and a more recently developed platform named Argo (Rak et al., 2012). Both of these systems allow the user to assemble modular pipelines and perform evaluations against a gold standard. U-Compare is plagued by instabilities of platform interoperability, permissions, and the like, and typically requires the intercession of a specialized software engineer. Argo attempts to ameliorate some of these problems by providing a web-based interface to the underlying UIMA-based system, but suffers from many of the same problems as U-Compare and is seemingly unsupported at this time. Frameworks that support general text mining, e.g., the General Architecture for Text Engineering (GATE) (Cunningham et al., 2011), provide "local interoperability" for tools available from within the framework, but there is no interoperability with tools or components available from outside the framework that the user might wish to use. In contrast, the LAPPS Grid provides interoperable access to tools in various UIMA systems as well as tools from GATE, which can be pipelined within the LAPPS Grid without the need for I/O format conversion.[11].

2.3. Access to Resources

Access to publication resources in the biomedical domain is problematic for several reasons:

1. Repositories vary in the types and format of their contents, some containing abstracts while others contain full text articles. Many provide only access to query

[9]Distributed under the terms of permissive Academic Free License: http://getgalaxy.org

[10]US NSF grant ABI 1661497

[11]The LAPPS Grid currently contains several GATE tools and makes them interoperable with all other tools in the framework, including several UIMA-based tools from DKPro.

results and not the full text itself. For those that do deliver the full text, many provide them in PDF format, which requires sophisticated conversion to enable text mining, while others deliver XML, JSON, and a variety of other formats.

2. Repositories are spread out all over the web, making them difficult to find and even harder to use together.

3. While some repositories are freely open, others require subscriptions or licenses.

An instance of the LAPPS Grid tailored to mining biomedical publications is currently maintained on the JetStream[12] cloud environment.[13] The instance currently includes full-text PubMed data used in several BioNLP and SemEval shared tasks, some with several layers of annotation. In addition, the LAPPS Grid has collaborated with the developers of PubAnnotation[14] to integrate the services and resources provided by each in order to greatly enhance the user's ability to annotate scientific publications and share the results. PubAnnotation is a repository of text annotations applied to biomedical publications, all of which are aligned to the canonical text in either PubMed or PubMed Central, thus linking all PubAnnotation annotations to each other through the canonical texts. Annotations are accessible and searchable through standard web protocols such as the REST API. Through the collaboration with PubAnnotation we have enabled access to holdings of PubMed (ca. 12 million abstracts) and PubMed Central (over 11 thousand full-text documents) by creating a "PubAnnotation datasource" in the LAPPS Grid. Users also have access to the annotations in the PubAnnotation repository that are linked to the texts.

Note that all text resources in the LAPPS Grid are rendered in the JSON-LD-based LAPPS Interchange Format (LIF) (Verhagen et al., 2015), which means that they are consumable by all tools in the LAPPS Grid.

We aim to provide access to as many full text articles as possible from within the LAPPS Grid as direct datasources, thus providing a "one stop" location for accessing publications available from otherwise scattered locations. The articles are converted to our internal JSON-LD format for delivery within the Grid so that researchers need not be concerned with issues of format conversion. For the purposes of mining scientific publications, we are interested in repositories that deliver full text, such as the following: PubMed Central (PMC)[15] (open section), PubMed Central Canada[16], Open Science Repository[17], Public Library of Science (PLOS)[18], arXiv[19], and BioMed Central[20]. We also plan to provide access for users with appropriate credentials to data from repositories that require a subscription, such as the Web of Science[21], using the authentication procedures already in place for delivering data from the Linguistic Data Consortium (LDC) within the Grid.

The LAPPS Grid also integrates several existing lexicons, ontologies, and knowledge bases relevant to the fields in which our collaborators are working (e.g., UMLS[22], EntrezGene[23], MeSH[24], UniProt[25]) as datasources, thus eliminating the need to access each one from a different source. In addition, Galaxy itself provides access to a very wide range of datasources relevant to genomic analysis and gene sequencing, as well as multiple tools to convert and manipulate the data.

3. Tools for Biomedical Text Analysis

In addition to the wide range of general purpose NLP tools available in the web-based LAPPS Grid instance[26], the JetStream instance currently includes several entity recognizers for biomedical terminology, event annotators, relation extraction software, and dictionary-based entity recognizers that can use customized lexicons, as well as machine learning facilities and tools for identifying key terms, synonyms, acronyms, lexical variants, and the like.

Figure 1 shows a portion of the LAPPS Grid menu in Galaxy that includes biomedical annotation tools together with a visualization of protein annotation on a PubMed document, using the Brat visualization tool included in the Grid.

4. Open Advancement Evaluation

One of the most valuable components of the LAPPS Grid suite of services for publication mining is its OA evaluation capability. Most information systems consist of a number of processing units or components arranged in series, or workflow; OA enables the user to automatically and efficiently evaluate different workflow configurations and identify those that achieve the best results. Much current research focuses on experimentation with parameters of a single module while keeping modules and parameters elsewhere in the system frozen. For example, a typical, simple workflow for biomedical text mining will rely on algorithms, toolkits, and pre-trained models for basic NLP processes such as sentence segmentation, tokenization, part-of-speech tagging, and chunking, coupled with tools and resources specially suited to a given area of biomedical research that extract terms and entities, identify terms and acronyms, provide synonyms and lexical variants, etc. Improvement of the system's performance might focus on refining entity extraction, possibly applying different models and/or augmenting a gazetteer or lexicon, while the preliminary processes remain static. There is typically no knowledge of the degree to which the performance of any individual module contributes to overall performance, although

[12] https://jetstream-cloud.org

[13] This instance can be accessed at http://jetstream.lappsgrid.org.

[14] http://www.pubannotation.org

[15] https://www.ncbi.nlm.nih.gov/pmc/

[16] http://pubmedcentralcanada.ca/pmcc/

[17] http://www.open-science-repository.com

[18] https://www.plos.org

[19] https://arxiv.org/

[20] https://www.biomedcentral.com

[21] https://clarivate.com/products/web-of-science/

[22] https://www.nlm.nih.gov/research/umls/

[23] http://www.ncbi.nlm.nih.gov/gene

[24] https://www.nlm.nih.gov/mesh/

[25] http://www.uniprot.org

[26] http://galaxy.lappsgrid.org

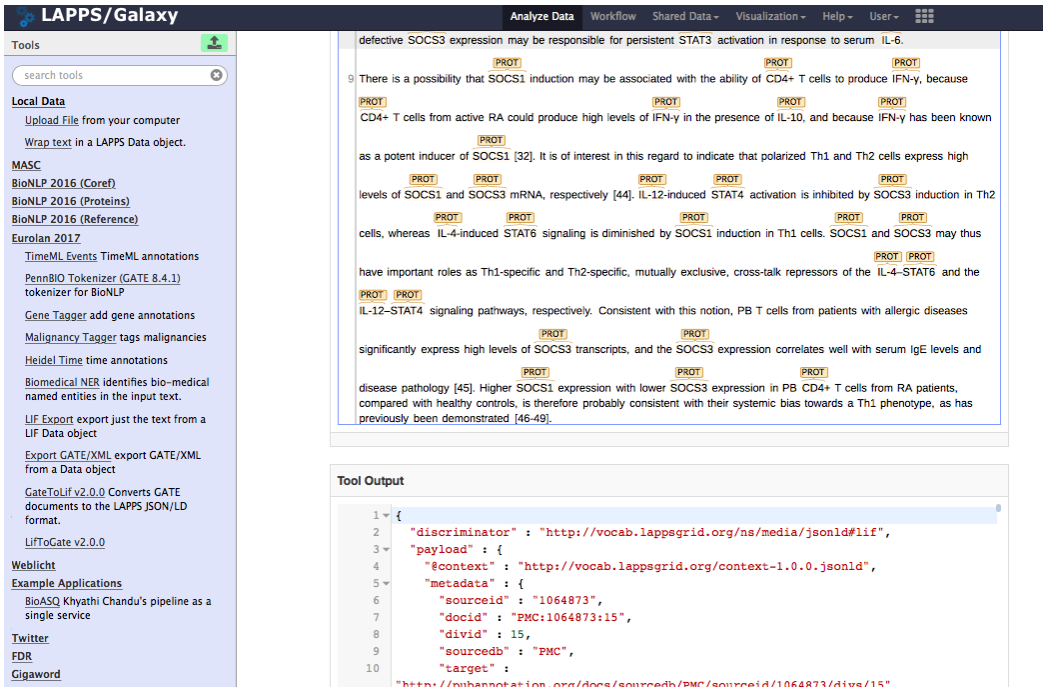


Figure 1: Some biomedical annotation tools and a visualization of protein annotation in the LAPPS Grid

earlier tools in the pipeline could in fact contribute significantly to poorer than expected results. The OA evaluation strategy overcomes this by exposing the performance of individual modules along with overall performance; an easy-to-use framework for building and testing different tool configurations such as the LAPPS Grid makes it easy to examine alternative tool chains and determine the optimal configuration.

As an example, (Yang et al., 2013) applied the OA approach to implement biomedical information systems for question answering tasks from the TREC Genomics. The OA framework automatically evaluated different system configurations and identified those that achieved better results than prior published results. The study found some simple and relatively unexplored contributors to improved performance including, for example, leveraging various sources varied for synonym expansion and acronym expansion, and altering weights for concept terms and verbs (concept terms favored higher weights and verbs favored moderate weights). Although automatic search through alternative pipelines is not yet implemented in the LAPPS Grid[27], even manual iteration over different configurations based on detailed information about each module's contribution can both improve results and reveal the impacts of processing components that are typically overlooked, due to the LAPPS Grid's wide range of modules for various tasks and the ease of workflow construction and modification.

5. Integration of Annotation Facilities

PubAnnotation includes TextAE, a powerful and easy-to-use Javascript app for text annotation and visualization. In our collaboration with PubAnnotation, we have enabled LAPPS Grid users to invoke TextAE from within the Grid, and we have similarly enabled PubAnnotation users to access and apply LAPPS Grid tools from within the PubAnnotation environment. Reciprocal access between PubAnnotation and the LAPPS Grid means that users can easily apply automatic annotation tools and subsequently manually correct annotations, in an iterative "human-in-the-loop" process of refinement. This is especially useful for the creation of training data for machine learning, especially iterative, semi-supervised approaches such as active learning. This, coupled with the Open Advancement evaluation facilities, provides a powerful environment for rapid development of high-quality automatic annotation procedures.

The LAPPS Grid provides means for users to easily register and thus share annotations in the PubAnnotation registry. As noted above, annotations registered in PubAnnotation are aligned with the canonical text and all other annotations applied to the same data. Conversely, annotations from the PubAnnotation registry, or annotations created by users from within the PubAnnotation platform using the TextAE editor, can be imported into the LAPPS Grid for further automatic processing, for example, application of tools that use these annotations in order to produce additional annotation layers.

Data and annotations in PubAnnotation are stored in a JSON format[28]; LAPPS Grid services use the Grid's JSON-LD format (LIF). For communication between the two platforms, we automatically convert between the two formats so that interoperability is seamless from the point of view of the user. Conversion has dictated some minor modifications to the PubAnnotation format, including the addition of metadata that is required by LAPPS Grid services, but is

[27] Search over multiple configuration spaces in the LAPPS Grid OA component is currently under development.

[28] http://www.pubannotation.org/docs/annotation-format/

otherwise relatively straightforward.

6. Support for Replicability and Reproducibility of Results and Data Sharing

There is an ongoing concern in the biomedical community concerning the transparency and reproducibility of published research (Ioannidis, 2005; Iqbal et al., 2016). Our adaptation of the Galaxy workflow system fosters replicability and reproducibility for biomedical text mining studies by providing the following capabilities[29]:

- automatic recording of inputs, tools, parameters and settings used for each step in an analysis in a publicly viewable history, thereby ensuring that each result can be exactly reproduced and reviewed later;
- provisions for sharing datasets, histories, and workflows via web links, with progressive levels of sharing including the ability to publish in a public repository;
- ability to create custom web-based documents to communicate about an entire experiment, which takes a step towards the next generation of online publications that would include both full paper and all supporting materials.

In addition to enabling other users to replicate an experiment, the individual user can develop a rich, organized catalog of reusable workflows rather than starting from scratch each time or trying to navigate a collection of *ad hoc* analysis scripts. Similarly, it is possible to repeatedly apply a command history on different data. Once an analysis is done, the record eliminates ambiguity as to which result used which settings and provides critical information for follow-up analysis.

Within the LAPPS Grid and its community of users, sharing of newly created resources (lexicons, etc.) for use by others from within the Grid, as well as for the purposes of replicability and reproducibility, is strongly encouraged. We anticipate that Grid users will increasingly create new resources of this kind, especially as we develop better facilities for domain adaptation (see Section 9.).

7. Support for Data Privacy

Privacy constraints often make it necessary to protect biomedical and clinical data from exposure to network access. In addition, research activity can be sensitive or proprietary and require protection from outside access. To address this need, a docker image containing a self-enclosed instance of the LAPPS Grid can be installed locally on a user's machine or server and used to access and process local data, thus disabling access via the internet. A local docker instance can also be used when a network connection is not available for any reason.

In addition to ensuring privacy, creation of a docker image containing a particular instance of the LAPPS Grid can provide absolute replicability for results, by encapsulating specific versions of tools, parameters, workflows, and data used in an experiment; any person wishing to replicate the original results or apply the methods to new data can easily do so without attempting to recreate the original environment, and with assurance that that environment is exactly as reported.

8. Handling Large-scale Data

An issue to be dealt with in text mining of large publication databases is the ability to handle high-throughput data at scale. The LAPPS Grid uses XSEDE, a heterogeneous high performance computing (HPC) environment for research, and the associated Jetstream cloud environment, for large-scale analyses and storage. The Grid will benefit tremendously in the near future from the recently-funded NSF ABI project (NSF ABI 1661497) that is linking Galaxy and XSEDE in order to provide HPC processing capabilities and massive storage through the Galaxy platform. To enable fast query of publication databases, we plan to create full text indexes with Apache Solr[30] of open scientific publication databases (PubMed, PLoS, Web of Science, etc.) for distributed indexing and load-balanced querying. The indexes will be stored in the cloud on Jetstream. We will also regularly (e.g., weekly) regenerate the Solr indexes so that the most recent material is available.

9. Future Work: Support for Domain Adaptation

Different domains and individual researchers have specific goals and knowledge interests, but it is all too often the case that an available system doesn't target the information of specific interest and is therefore not useful. We see development of methods for domain adaptation for text mining as a critical, and *as yet unaddressed*, need for mining scientific publications.

Scientific publications share commonalities of structure and style, but across domains and areas of specialization they differ most drastically in the use of highly specialized vocabularies and terminology, which may comprise as much as 12% of overall document vocabulary. Domain adaptation for scientific publications therefore necessarily focuses on handling previously unseen vocabulary and terms. Various vocabularies, ontologies, and knowledge bases exist in the field, but these resources cover only a fraction of the vocabulary, especially for specialized subdomains. Furthermore, the constant introduction of new terms and short forms or abbreviations makes vocabulary adaptation for scientific publications an ongoing activity.

Domain adaptation for scientific text mining therefore involves the ability to modify and extend existing lexicons and other supporting resources. To do this, the scientist must be able to examine results, identify unrecognized terms and false positives, retrain a recognition module using the new information, and run the workflow again. Thus domain adaptation involves an iterative cycle of performance improvement (active learning).

[29] See (Goecks et al., 2010) for a comprehensive overview of Galaxy's sharing and publication capabilities, and (Sandve et al., 2013) for further discussion.

[30] http://lucene.apache.org/solr/

In Section 5. we briefly outlined how a user might exploit within the LAPPS Grid/PubAnnotation integration to facilitate the development of training data; a similar iterative process can be applied to development of supporting resources such as lexicons and term banks. We see domain adaptation as an increasingly important need for the future development of publication mining capabilities; by providing facilities to support this activity, as well as supporting and encouraging sharing of resources and methods, we hope to move the field forward.

10. Conclusion

Support for biomedical text mining is now a major focus for LAPPS Grid development. In addition to the facilities and collaborations with the Galaxy and PubAnnotation projects described above, we have just begun working on mining and summarizing clinical reports in a collaboration with the US Center for Disease Control (CDC) and Food and Drug Association (FDA), which will extend our work to a broader range of text types and applications.

At this time, tools in the LAPPS Grid focus on English. However, in a project funded by the A. K. Mellon Foundation, we are establishing an interoperable "bridge" with the CLARIN WebLicht framework (Dima et al., 2012) hosted by the CLARIN-D Center in Tübingen, Germany[31], which will give LAPPS Grid users access to a vast range of NLP tools and data available from WebLicht and CLARIN Centers throughout Europe. Access to tools for multi-lingual analyses will extend LAPPS Grid capabilities to multiple languages. In addition, because the LAPPS Grid is federated with seven other grids in the Federated Grid of Language Services (FGLS) (Ishida et al., 2014), including the Language Grid housed at Kyoto University[32], users will have interoperable access to atomic and composite web services for Asian languages available from any of these federated grids.

As a final note, the LAPPS Grid, Galaxy, and PubAnnotation are all open source projects that invite contributions from developers and users. In particular, the LAPPS Grid seeks the contribution of tools and resources for biomedical text mining to augment the current facilities. Thus we hope to build up a truly useful platform that is available to everyone, whatever their goals.

11. Acknowledgements

This work is supported by National Science Foundation grants NSF-ACI 1147944, NSF-ACI 1147912, and NSF ABI 1661497.

12. Bibliographical References

Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Cech, M., Chilton, J., Clements, D., Coraor, N., Eberhard, C., Grüning, B. A., Guerler, A., Hillman-Jackson, J., Kuster, G. V., Rasche, E., Soranzo, N., Turaga, N., Taylor, J., Nekrutenko, A., and Goecks, J. (2016). The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. *Nucleic Acids Research*, 44(Webserver-Issue):W3–W10.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M. A., Saggion, H., Petrak, J., Li, Y., and Peters, W. (2011). *Text Processing with GATE (Version 6)*.

Dima, E., Hinrichs, E., Hinrichs, M., Kislev, A., Trippel, T., and Zastrow, T. (2012). Integration of weblicht into the clarin infrastructure. In *Proceedings of the Joint CLARIN-D/DARIAH Workshop at Digital Humanities Conference 2012: Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts*, pages 17–23, Hamburg, Germany.

Giardine, B., Riemer, C., Hardison, R. C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., Miller, W., Kent, W. J., and Nekrutenko, A. (2005). Galaxy: A platform for interactive large-scale genome analysis. *Genome Research*, 15(10):1451–55.

Goecks, J., Nekrutenko, A., and Taylor, J. (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11:R86.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., Wang, D., Suderman, K., Verhagen, M., and Wright, J. (2014). The language application grid. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLOS Medicine*, 2(8).

Iqbal, S. A., Wallach, J. D., Khoury, M. J., Schully, S. D., and Ioannidis, J. P. A. (2016). Reproducible Research Practices and Transparency across the Biomedical Literature. *PLOS Biology*, 14(1).

Ishida, T., Murakami, Y., Lin, D., Nakaguchi, T., and Otani, M. (2014). Open Language Grid–Towards a Global Language Service Infrastructure. In *The Third ASE International Conference on Social Informatics (SocialInformatics 2014)*, Cambridge, Massachusetts, USA.

Kano, Y., Nguyen, N., Sætre, R., Yoshida, K., Fukamachi, K., Miyao, Y., Tsuruoka, Y., Ananiadou, S., and Tsujii, J. (2008). Towards Data And Goal Oriented Analysis: Tool Inter-Operability And Combinatorial Comparison. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 859–864, Hyderabad, India, January.

Rak, R., Rowley, A., Black, W., and Ananiadou, S. (2012). Argo: An integrative, interactive, text mining-based workbench supporting curation. *Database*, 2012:bas010.

Sandve, G. K., Nekrutenko, A., Taylor, J., and Hovig, E. (2013). Ten Simple Rules for Reproducible Computational Research. *PLoS Computational Biology*, 9(10).

Verhagen, M., Suderman, K., Wang, D., Ide, N., Shi, C., Wright, J., and Pustejovsky, J. (2015). The LAPPS Interchange Format. In *Revised Selected Papers of the Second International Workshop on Worldwide Language Service Infrastructure WLSI (2015)*, pages 33–47, Kyoto,

[31] This project is described in a separate submission to LREC 2018.

[32] http://langrid.org

Japan. Springer International Publishing.

Yang, Z., Garduno, E., Fang, Y., Maiberg, A., McCormack, C., and Nyberg, E. (2013). Building Optimal Information Systems Automatically: Configuration Space Exploration for Biomedical Information Systems. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 1421–1430, New York, NY, USA. ACM.

An Initial Test Collection for Ranked Retrieval of SMS Conversations

Rashmi Sankepally, Douglas W. Oard
iSchool and UMIACS
University of Maryland
College Park, MD USA 20742
rashmi@umd.edu, oard@umd.edu

Abstract

This paper describes a test collection for evaluating systems that search English SMS (Short Message Service) conversations. The collection is built from about 120,000 text messages. Topic development involved identifying typical types of information needs, then generating topics of each type for which relevant content might be found in the collection. Relevance judgments were then made for groups of messages that were most highly ranked by one or more of several ranked retrieval systems. The resulting TREC style test collection can be used to compare some alternative retrieval system designs.

Keywords: ranked retrieval, test collection, relevance assessment, retrieval effectiveness

1. Introduction

With the ubiquity of smartphone usage, many people have turned to using short message services (SMS) to send quick messages that don't require an immediate response. As with other types of "conversational text" (e.g., email, discussion forums, online reviews), personal SMS text message archives can be valuable information sources in their own right, both for the people who contributed to their creation and for others (e.g., historians or researchers) who may subsequently gain authorized access to such collections.

Test collections have been built for a number of information retrieval tasks, both in shared-task venues such as the Text Retrieval Conference (TREC) and for more focused development. We are, however, not aware of any prior work on characterizing the effectiveness of searching SMS content. As part of the DARPA Broad Operational Language Translation (BOLT) program, the Linguistic Data Consortium (LDC) released some SMS corpora with the principal goal of supporting research on machine translation for informally written content (Song and others, 2014). In this paper, we describe our development of an information retrieval test collection based on those LDC SMS corpora.

The paper is organized as follows: the next section reviews some related work, the data collection is described in section 3, section 4 describes topics used in the test collection, section 5 gives the relevance assessment procedure and we conclude in section 6.

2. Related Work

Although formal dissemination-oriented content such as news stories and scientific papers have been the focus of much information retrieval research, there has also been some work on development of test collections for more informal user-generated content in which there is potentially some interaction among the content creators. Among these are the CLEF Cross-Language Speech Retrieval Track focused on interviews (in which interviewer and interviewee co-construct the interview, passages from which are to be found) (Pecina and others, 2007), the TREC Microblog track (focused on Twitter content, some of which is directed between specific users) (Lin and others, 2014), the TREC Legal Track (focused on email content, some of which is easily threaded into conversations) (Grossman and others, 2011), and the the TREC Blog Track (some of which involves comments on other blog posts) (Macdonald et al., 2009). Although differing in some details, each proved amenable to a fairly conventional approach to test collection construction involving topic design, runs from a diverse set of systems, some way of sampling documents for relevance judgments, and some rank-based or set-based evaluation measures. We are, however, aware of only one shared task evaluation involving SMS messages: The FIRE SMS-Based FAQ Retrieval task. In that task, queries were posed using SMS, and a preexisting set of answers to Frequently Asked Questions (FAQ) provided the "document" set to be searched (Contractor and others, 2013). Our goal in this paper is to switch the focus from searching using SMS queries to searching SMS content itself. We studied the process adopted by LDC to create open-domain queries as part of the same DARPA BOLT program (Griffitt and Strassel, 2016). Those queries were developed for discussion forum posts and intended to be cross-lingual to some extent. We used a modified version of that process to create our queries to search the SMS content as described in section 4..

3. Selecting the Messages

The SMS collection that we have used was assembled by the LDC for the DARPA BOLT program and released in three phases (Song and others, 2014)[1]. As released, the LDC corpora contained both English SMS messages and English text chat logs that were contributed for research use (in exchange for compensation) by individuals. Contributors were offered the opportunity to redact content that they did not wish to have distributed, and LDC reviewers also examined each message for content for which distribution would not be appropriate. Redactions are marked with sequences of hash characters ("#"). The vast majority of the messages are from SMS, so we used only the SMS mes-

[1] LDC catalog: LDC2013E49, LDC2013E63, LDC2013E84

sages as the basis for our test collection. Because the machine translation systems for which the corpora were originally designed are sometimes designed to process entire documents, the LDC grouped the SMS messages between each pair of participants into time-ordered sets that can be thought of as "conversations" (although of course in practice some such sets actually include discussions of multiple topics). Redactions proved to be rare, affecting only 202 conversations, and being limited to part or all of a single message in 151 of those conversations, so we retained conversations that included redacted messages.

This process resulted in 8,282 conversations. From these, we removed the 55 that contained only a single message and the 40 with the greatest number of messages. This resulted in 8,187 conversations for our test collection, each containing between 2 and 303 consecutive SMS messages between a pair of correspondents. For most (6,439) of these conversations, the time span between the first and last messages is no more than 24 hours. Together, the 8,187 conversations contain 121,114 messages (for a mean conversation length of 14.8 messages); only 184 of the 8,187 conversations contain 100 or more SMS messages.

For convenience, we used the LDC conversations as the unit of annotation for relevance judgments. In other words, our evaluation asks whether we are able to find a conversation that the searcher might wish to see. We also used the conversations as retrieval units when pooling system results for relevance judgment, although we additionally conducted post-hoc experiments with smaller indexing units, thus preferring conversations that exhibit a temporal concentration of content on the topic being searched for.

4. Topics and Queries

The most challenging aspect of creating our test collection was to develop topics that we believed reasonably represent what real users might actually search for in an SMS message archive. As a starting point, we looked to two prior observational studies for other types of conversational content. Looking first to the nature of the content, Naaman et al. report that the most common types of content users share on Twitter are opinions or complaints, self reports, random thoughts and facts (Naaman et al., 2010). Looking next to what people might ask about, Oard clustered questions from an existing question answering test collection for searching Web discussion forums, identifying four categories: "open" opinion oriented questions that do not suggest a perspective, focused opinion-oriented questions that ask for opinions on a specific aspect of a topic, experiential questions and knowledge-oriented questions (Oard, 2012).

To see which categories might be useful for an SMS collection, we examined part of our collection (about 100 conversations). After this we settled on four topic types: *opinion* (seeking personal opinions), *behavior* (seeking to learn what people do), *experience* (seeking insights from someone with experience) and *knowledge* (seeking to learn something that someone knows). We then described these question types in general terms to a colleague who had not seen the collection and asked them to craft some plausible topics of each type. We additionally looked for inspiration to topic descriptions from TREC Microblog tracks (2011, 2012, 2013) and TREC Robust tracks (2004, 2005).

```
<top lang="en" type="opinion">
<num> 034 </num>
<title> farmers markets </title>
<desc>
What do people think about farmers'
markets?
</desc>
<narr>
Farmers' markets feature a retail
market where food items  are sold
directly by farmers to consumers.
To be relevant, conversations would
contain people expressing their
opinions on farmers' markets.
</narr>
</top>
```

Figure 1: An example topic.

Table 1: Topics types across the two assessment phases along with example titles for each type.

Type	Phase1	Phase2	Total	Example Title
opinion	3	4	7	new Xbox release
behavior	4	3	7	disobeying rules
experience	7	9	16	living with parents
knowledge	0	1	1	Philly bars
Total	14	17	31	

We formalized each topic in a TREC-like format, as illustrated in Figure 1. Title (T) contains a few words that we expect a user might type as an initial search; Description (D) is a fully formed question styled after a question answering task; and Narrative (N) is intended to guide relevance assessment. A total of 62 topics were developed in this way (see `https://github.com/rashmisankepally/SMSTestCollection/blob/master/topics.txt`). We indexed the collection using Indri [2] with conversations as the unit of retrieval, and then checked to see if we could find at least some relevant content for each topic. This triage process yielded 36 potentially useful topics, of which we actually used the 31 for which our assessors ultimately found one or more relevant conversations. Table 1 shows the final number of topics by type for the two phases of relevance assessment as described in section 5. below.

5. Relevance Assessment

We hired 3 assessors to perform relevance judgments. Because none had prior relevance judgment experience, we divided the task into two phases. After some initial training, each assessor judged 7 topics for relevance, 3 of which

[2] `https://www.lemurproject.org/indri/`

Table 2: Inter-assessor agreement (kappa).

Assessor Pair	Phase 1	Phase 2
A:B	0.090	0.328
B:C	0.100	0.203
C:A	0.184	0.498

were common to all assessors. Phase 1 thus produced judgments for 15 different topics.

Pools of conversations to be judged for each topic were created by selecting the top 50 conversations from three diverse ranked retrieval systems: (1) language model (LM): Indri's (Strohman and others, 2005) default Language Modeling with $\mu = 1000$, (2) query expansion (QE): the pseudo relevance feedback model in Indri with 20 documents and 30 terms used for query expansion, and (3) BM25: Indri's implementation of BM25 term weights, with default parameters. Each of these was run three times (once each with T, TD, TDN queries, each created by concatenating all words in the indicated fields).

Assessors were asked to give a score for each topic-conversation pair that they were presented. Conversations were formatted for display as shown in Figure 2. Assessors were asked to base their decision regarding relevance on whether the information need was addressed by any part of the conversation. They had four options for assessment: HREL (worthy of being a top result); REL (somewhat relevant content); NON (no useful information about the topic); and JUNK (no useful information for any purpose). For scoring, we collapsed the NON and JUNK categories into a single "non-relevant" category. This resulted in ternary graded judgments that (for computing measures such as kappa and Average Precision) we further binarized by additionally collapsing HREL and REL into a single "relevant" category.

We evaluated assessor agreement for the 3 common topics (described below) in phase 1 and used those results to guide a conversation among the assessors with the goal of achieving a greater agreement in phase 2. In the second phase, each assessor judged relevance for 9 additional topics, 3 of which were again common. This produced judgments for the remaining 21 topics. Depth 50 pooling was again used, this time with 12 runs (the same 9 runs as phase 1, plus 3 runs (T, TD, TDN) for a system built using word2vec (Mikolov and others, 2013)). Our word2vec system was a query expansion model using 100-dimension word embedding vectors that was added to increase the diversity of the pools. Word2vec's clustered bag of words (CBAG) model was used, with context set to 10 and number of iterations set to 5.

Over the total of 31 topics that each had at least one relevant document, there are a total of 214 relevant conversations in the pools (119 for 14 topics in phase 1, 95 for 17 topics in phase 2) for an average of 6.9 relevant conversations per topic.

5.1. Assessor Agreement and Comparing System Rankings

In phase 1, the pools for topic numbers 015, 017 and 020 were judged by all three assessors; together those judgment pools contained 438 conversations on which we could compute Cohen's kappa, a chance-corrected agreement measure. In phase 2, the pools for topics 023, 024 and 032 were judged by all three assessors; in this case kappa was computed over 421 conversations. As Table 2 shows, annotator agreement improved markedly in phase 2. Some possible reasons for the lower agreement in phase 1 were: long conversations, abrupt topic shifts in a conversation, and differing interpretations of a topic. That last factor was perhaps exacerbated by cultural factors, as all the three assessors were from India or China, while the SMS messages had been collected in the USA. Post hoc analysis found that length did not measurably affect inter-annotator agreement.

```
<conversation id="SMS_ENG_20110xx.x">

 m0000 - A:[2011-02-09 19:03:03]
         Where you at.what you doing?
 m0001 - B:[2011-02-09 19:04:04]
         I'm in NYC to see a show for
         the evening!
 m0002 - B:[2011-02-09 19:19:15]
         I have an extra ticket if
         you wanna join :)
```

Figure 2: Conversation format shown to assessors.

The kappa values in phase 2 are more typical of agreement statistics for relevance judgment results in other settings (Voorhees, 2000). Ultimately however, what we care about is whether the resulting judgments can be used to determine which retrieval systems are better. To explore this, we examined the degree to which judgments from different assessors during the second phase would produce the same preference order among systems when those systems were evaluated by Average Precision, a widely used ranked retrieval measure. Of the three topics for which judgments were obtained for all assessors, only topic 024 proved to be suitable for this analysis. Assessor B judged none of the sampled conversations as relevant to Topic 032, thus precluding computation of any system rankings for that topic using Assessor B's judgments. For Topic 023, each assessor found only two relevant conversations. While it is numerically possible to compute Average Precision with only two relevant items, the use of the inverse rank in the Average Precision computation introduces substantial quantization noise when so few relevant items are known to exist. For topic 024, by contrast, no assessor judged fewer than 5 conversations to be relevant (specifically, Assessor A found 12, Assessor B found 5, and Assessor C found 16 relevant conversations). Five relevant items suffice to compute Average Precision with only minimal quantization noise effects.

Figure 3 plots a comparison for topic 024, showing Average Precision (AP) scores for 15 systems (the 12 that contributed to the pools, plus 3 passage retrieval runs—for the same three query lengths—constructed using 60-

Assessor pair	τ
A-B	0.45
B-C	0.30
C-A	0.16

Table 3: Rank correlation for Topic 024.

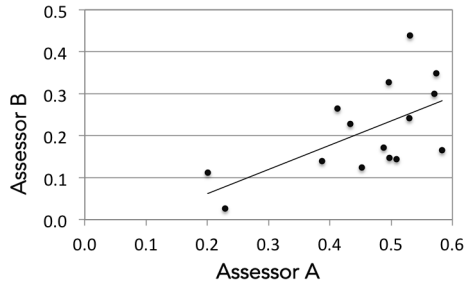


Figure 3: Average Precision Correlation for Topic 024.

word sliding windows with 45-word overlap and Indri's language model, with each document assigned the score of its highest-scoring passage) computed using relevance judgments from different assessors. Although Assessor B's scores tend to be somewhat lower than Assessor A's (perhaps because Assessor A judged more conversations to be relevant, thus setting up an easier search problem), higher AP scores computed with Assessor B's judgments are clearly predictive of higher AP scores computed with Assessor A's judgments.

We can summarize the degree of consistency of system rankings using Kendall's τ, a rank correlation number whose values range between -1 and 1. (with 1 indicating identical rankings, 0 indicating completely random swaps that would be characteristic of unrelated rankings, and -1 indicating complete reversal). As Table 3 shows, assessors A and B show reasonable consistency in system ranking. Another thing that we can observe from Table 3 is that systems ranked using Assessor B's judgments agree more strongly with the judgments of Assessors A and C than do Assessors A and C agree with each other. Thus we have some indication that choosing Assessor B's judgments would be reasonable for the six topics that had been judged by all three assessors. That's what we have done to create the final relevance judgment file in the released test collection.

6. Using the Collection

Table 4 shows phase 2 results for the systems that contributed to the pools, and for T, TD, and TDN runs with a fifth system (LM-60-45), the best of a set of passage retrieval systems we built for post hoc experiments in which our goal was to assess the effectiveness of more focused retrieval techniques. For LM-60-45 we use a sliding window to form 60-word passages, with 45-word overlap between adjacent passages. We score each passage using the Indri LM, and each conversation is assigned the maximum LM score across its passages. The best of the systems that we tried (LM-60-45 with TDN queries) achieved MAP above 0.4 (indicating that near the top of the ranked list more than 40% of the conversations were relevant, on average) and

Table 4: Phase 2 results.

Query	System	MAP	nDCG
TDN	LM-60-45	0.421	0.594
TD	LM-60-45	0.414	0.587
T	LM-60-45	0.412	0.575
T	QE	0.379	0.550
T	BM25	0.367	0.534
T	LM	0.352	0.523
TDN	LM	0.350	0.516
T	word2vec	0.343	0.506
TD	LM	0.338	0.500
TD	QE	0.325	0.499
TD	word2vec	0.304	0.460
TD	BM25	0.239	0.442
TDN	QE	0.219	0.405
TDN	word2vec	0.191	0.305
TDN	BM25	0.130	0.307

nDCG near 0.4 (indicating that on average the best system ranks documents better than half as well as the best possible ranking ,which would rank all the HREL ahead of all the REL, which are in turn ahead of all the other conversations). These results suggest that a retrieval system built with the best of these methods should be usable for typical interactive search tasks. The correlation between these 15 systems across the two measures is nearly perfect; the only reversal is between the two lowest-ranked systems. Results for phase 1 topics were broadly comparable, ranking the smaller number of systems that we tested in phase 1 consistently with those in phase 2, with MAP ranging between 0.268 and 0.463 and nDCG range in between 0.505 and 0.659.

Although our principal focus has been on development of the test collection, two results seem worthy of note. First, our best passage retrieval model did substantially better than any other approach. Pooled assessment can create some degree of bias against post hoc assessment of new systems, so this result suggests that passage retrieval may be a useful approach for this task. Second, longer (TDN) queries yielded better results with passage retrieval, whereas shorter (T) queries yielded better results for every other approach. This comports with our intuition that additional context can be provided by longer queries or by longer "documents," thus suggesting that when short queries are all that the user provides we might prefer longer passage lengths (up to entire conversations).

7. Conclusion

We now have a test collection for evaluating information retrieval systems designed for SMS conversations. Topics, relevance judgments and the list of SMS conversation IDs used in the collection can be obtained from `https://github.com/rashmisankepally/SMSTestCollection`. Although the collection contains only 31 topics, a number generally considered too few for reliable statistical significance tests (Sanderson and Zobel, 2005), we have already

been able to make some useful observations. Mean Average Precision results for phase 2 topics ranged between 0.130 and 0.421, values consistent with those observed on typical TREC collections, showing that the systems that contributed to the pools performed well enough to find a substantial number of relevant documents. Since we completed this work, the LDC has released larger SMS collections, so leveraging what we have learned to create larger test collections, both with more messages and with more topics, would be a natural next step. In addition to judgments on conversations, we also collected (but have not yet analyzed) message-level relevance judgments. Analyzing agreement on those message-level judgments, and using those judgments for more fine-grained evaluation, would be worth doing.

8. Bibliographical References

Contractor, D. et al. (2013). Text retrieval using SMS queries: Datasets and overview of fire 2011 track on SMS-based FAQ retrieval. In *Multilingual Information Access in South Asian Languages*, pages 86–99. Springer.

Griffitt, K. and Strassel, S. (2016). The query of everything: Developing open-domain, natural-language queries for BOLT information retrieval. In *LREC*.

Grossman, M. R. et al. (2011). Overview of the TREC 2011 legal track. In *Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, USA, November 15-18, 2011*.

Lin, J. et al. (2014). Overview of the TREC-2014 microblog track (notebook draft). In *Proceedings of TREC*.

Macdonald, C., Ounis, I., and Soboroff, I. (2009). Overview of the TREC 2009 blog track. In *Proceedings of The Eighteenth Text REtrieval Conference, TREC 2009, Gaithersburg, Maryland, USA, November 17-20, 2009*.

Mikolov, T. et al. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Naaman, M., Boase, J., and Lai, C.-H. (2010). Is it really about me?: message content in social awareness streams. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 189–192. ACM.

Oard, D. (2012). Answering questions from conversations. In *COLING Question Answering for Complex Domains Workshop*. Powerpoint slides.

Pecina, P. et al. (2007). Overview of the CLEF-2007 cross-language speech retrieval track. In *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, pages 674–686.

Sanderson, M. and Zobel, J. (2005). Information retrieval system evaluation: Effort, sensitivity, and reliability. In *SIGIR*, pages 162–169.

Song, Zhiyi and others. (2014). *Collecting Natural SMS and Chat Conversations in Multiple Languages: The BOLT Phase 2 Corpus.*

Strohman, T. et al. (2005). Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, volume 2, pages 2–6.

Voorhees, E. (2000). Variations in relevance judgments and the measurement of retrieval effectiveness. *IP&M*, 36(5):697–716.

FrNewsLink : a corpus linking TV Broadcast News Segments and Press Articles

Nathalie Camelin[1], Géraldine Damnati[2], Abdessalam Bouchekif[1,3], Anais Landeau[1], Delphine Charlet[2], Yannick Estève[1]

(1) LIUM, Maine University, France – {firstname.surname}@univ-lemans.fr

(2) Orange Labs, Lannion, France – {firstname.surname}@orange.com

(3) LSE, EPITA, France – {firstname.surname}@epita.fr

Abstract

In this article, we describe *FrNewsLink*, a corpus allowing to address several applicative tasks that we make publicly available. It gathers several resources from TV Broadcast News (TVBN) shows and press articles such as automatic transcription of TVBN shows, text extracted from on-line press articles, manual annotations for topic segmentation of TVBN shows and linking information between topic segments and press articles. The *FrNewsLink* corpus is based on 112 (TVBN) shows recorded during two periods in 2014 and 2015. Concomitantly, a set of 24,7k press articles has been gathered. Beyond topic segmentation, this corpus allows to study semantic similarity and multimedia News linking.

Keywords: content linking, semantic textual similarity, topic segmentation

1. Introduction

Several corpora are dedicated to evaluation in the domain of topic segmention, semantic similarity or resource linking (textual or multi-modal). An overview of such corpora is proposed in Section 2. contextualizing our proposed corpus. We describe in this article the corpus *FrNewsLink* that can be downloaded on the LIUM website[1] in its first version. During two different periods of time: 7 consecutive days in February 2014 and 2 consecutive days in January 2015, several TVBN shows from 8 different French channels have been recorded and web press articles appearing on the Google News homepage have been downloaded. As a consequence, *FrNewsLink* resources (whose elements are described in Section 3. and detailed statistics are given in Section 5.) deal with various fields such as politics, sports, cinema, *etc.* and some events that are very dynamic during a given day. Then, a manual annotation process (described in Section 4.) has been performed in order to obtain: 1- topic segmentation annotations and 2- linking annotations between topic segments and press articles. Therefore, this corpus is very useful for many tasks such as topic segmentation, topic titling, video linking, semantic similarity, events follow-up and topic modeling (grouping documents addressing similar topics). Section 6. presents several tasks that can be addressed with our corpus, along with evaluations of theses tasks performed on it.

2. Related work

2.1. Corpora for topic segmentation

The C99 corpus designed by (Choi, 2000) has been widely used for evaluating topic segmentation of written text. It is an artificial corpus composed of 7000 segments randomly selected from articles of the Brown corpus. These segments are grouped into 700 documents where each document is the concatenation of 10 segments. A segment is composed of the first n sentences of the original article.

The *ICSI* corpus (Shriberg et al., 2004) (Shriberg et al., 2000) contains 75 documents transcribed automatically from meeting records (approximately one hour each). For each conversation turn the speaker, start time, end time, and word content are marked. This corpus has been exploited in several works as (Eisenstein and Barzilay, 2008) and (Galley et al., 2003).

The *TDT* (Topic Detection and Tracking) corpus has become a standard for topic segmentation. This corpus contains English, Arabic and Chinese (Mandarin) documents. The corpus with its different versions (from TDT1 to TDT5) is used to evaluate many topic segmentation systems as (Rosenberg and Hirschberg, 2006) and (Xie et al., 2010). It is important to note that several works dedicated to topic segmentation use their own corpus (Malioutov and Barzilay, 2006), (Eisenstein and Barzilay, 2008). In (Malioutov and Barzilay, 2006), the authors have created their corpus from physics course recordings. In (Eisenstein and Barzilay, 2008), the authors have made available a medical book in which each section is considered as a new topic segment. In the domain of French TVBN topic segmentation (Guinaudeau, 2011) created a corpus of 57 news programs broadcasted in February and March 2007 on the French television channel France 2. It contained 1180 topic boundaries. As will be developed in section 5.1. the particularity of our corpus lies in the diversity of TVBN shows sources and formats.

2.2. Corpora for semantic textual similarity

Semantic textual similarity measures the meaning similarity of texts, beyond lexical similarity. It has been treated in several evaluation campaigns in SemEval in the recent years (Cer et al., 2017). Available corpora are mainly in English, even though some have been introduced in Spanish and Arabic. In the semantic textual similarity task of SemEval, similarity is measured between short sentences, according to a scale of 5 levels, ranging from "no relation at all" to "paraphrase". Another task of the SemEval campaign deals with semantic similarity in the context of Community Questions Answering (Nakov et al., 2017). In particular: one subtask addresses "question-question" similarity, where the questions are questions posted on an english-

[1] https://lium.univ-lemans.fr/frnewslink/

speaking forum (Qatar Living Forum) which deals with any aspect of daily life for foreign people in Qatar. In the corpus, for each original question, a set of 10 questions retrieved by a classical information retrieval system are to be ranked, according to their semantic similarity to the original question. Manual reference for semantic similarity is on a 3-level scale (Perfect Match, Relevant, Irrelevant).
To the best of our knowledge, there isn't any freely available corpus in French, annotated in semantic similarity.

2.3. Multi-modal linking

Multi-modal linking is a domain where semantic similarities are searched among multi-modal documents and/or across modalities. Video hyperlinking is a task in multimedia evaluation campains such as MediaEval (Eskevich et al., 2014) or TRECVid (see *e.g.* (Bois et al., 2017b)). The objective here is to be able to link an anchor (a piece of a BBC program which has been selected by experts as a segment of interest) to other segments defined as targets, that can be extracted from 2,700 hours of programs. This task, similarly to textual semantic similarity tasks, refers to homogeneous data: the objective is to link a fragment to another fragment from the same source. Some other works attempt to link heterogeneous sources but from an alignment perspective (*e.g.* books and movies (Zhu et al., 2015) or video lectures and scientific papers (Mougard et al., 2015)).
In the News domain there has been several studies about linking press articles with other information sources. (Aker et al., 2015) explore linking press articles and comments on the AT corpus (Das et al., 2014) which has been built from article of The Guardian. Linking press articles and Tweets have also been studied (Guo et al., 2013). Closer to our purpose is the work of (Bois et al., 2017a) who attempt to build graph representations for News browsing. The authors have collected over a 3 week period (May 20−Jun 8, 2015) a corpus of documents in French including press articles, videos (*e.g.* daily news from France 2, political news), and radio podcasts (*e.g.* news programs from France Inter, political interviews from RMC). This corpus is not distributed so far. The FrNewsLink corpus allows addressing several multi-modal linking tasks, with heterogeneous data from various sources and of various length.

3. The FrNewsLink corpus

The *FrNewsLink* corpus is composed of several resources extracted from TVBN shows and web press articles collected during the same period.

3.1. TV Broadcast News data

The *FrNewsLink* corpus contains 86 TVBN shows recorded during one week from the 10^{th} to the 16^{th} of February 2014 and 26 TVBN shows recorded on 26^{th} and 27^{th} January 2015. The first epoch of the corpus is referred to as TV_W07_14 (7^{th} week of 2014) and the second as TV_W05_15. Contrarily to many other TVBN corpora, the particularity of this corpus lies in the large variety of formats : 14 different shows from 8 different channels (Arte, D8, Euronews, France2, France3, M6, NT1, TF1) at various times of the day. Our original objective was to develop automatic structuration approaches that are robust to format variations. Hence the corpus contains both traditional shows (with a succession of reports introduced by one anchorman) and more modern setups (with two anchormen or no anchorman at all, with debates on stage, with short pieces of news, *etc.*). Section 5. provides detailed statistics on the TVBN part of the corpus.
Due to legal issues, we cannot distribute the videos corresponding to these shows, but we propose to make available the automatic transcription as well as the automatic speaker diarization results. The automatic transcription is performed with a variant of the LIUM ASR system described in (Tomashenko et al., 2016) in its chain-TDNN version with discriminative learning (Peddinti et al., 2015). Speaker diarization results are provided by the LIUM_SpkDiarization system (Meignier and Merlin, 2010). On similar corpora, such systems observed a performance between 9% and 13% of WER and between 11% and 14% of DER.

3.2. Press articles

During the same periods of time, we have extracted articles from various press websites, using the Google News homepage for selection. Google News articles are clustered and presented by news topic. For each topic cluster, one article is highlighted as the main article and other ones are shown as related. We have chosen to download the Google News homepage every hour. Each referenced article becomes an entry of our database which is composed of an id number, the article's title, the date and time of the record, the link to the referenced article and the id number of the main topic article. This extraction has been performed during seven consecutive days from the 10^{th} to the 16^{th} of February 2014, leading to the Press_W07_14 sub-corpus, and on the 26^{th} and 27^{th} of January 2015, leading to the Press_W05_15 sub-corpus).
Only the article content itself is relevant while the remaining (*e.g.* navigation menu, reader comments, pictures, *etc.*) is uninformative and has to be discarded. To that purpose, we have used the Boilerpipe (Kohlschütter et al., 2010) library that provides quite accurate algorithms to detect and remove all the surplus of a web page.

4. Annotation process

4.1. Reference annotation for topic segments

The task consists in placing boundaries between topic segments. The definition of a topic can be subject to various interpretations. In our work, we consider a topic as a precise information, which takes place at a given moment and place. For instance in the case of several consecutive reports on sport results, there will be as many topic segments as addressed sport disciplines. Within the context of the crisis in Greece, if several consecutive reports concern several consequences of the crisis, they will be split into consecutive topic segments. For greater clarity, and in order to facilitate the other annotation tasks, a small textual description has been inserted describing the content of each segment. Note that the first and last topic segments of a show have been annotated with a particular tag when they correspond to the titles presentation or the summary. Actually for topic

segmentation evaluation they are usually discarded. Moreover, in some BN, the anchor gives a small description in the middle of the show of the topics that will be dealt with later. These segments are also particularly tagged during the annotation phase. One human annotator has proceed to the whole annotation : segmentations, summaries and discarded segments. Time spent on this first annotation phase is about 8 months (about 1.1k hours).

4.2. Reference annotation for linking

Originally our objective was to define a titling strategy where a given topic segment from a TVBN show could be assigned a title from a set of candidate press article titles (Bouchekif et al., 2016). Hence the protocol for manually annotating the ground truth for this task was defined as follows. Given a reference topic segment, and the set of article titles collected during the same day, the annotation process consists in specifying for each candidate title if *(i)* a title is suitable for the segment, if *(ii)* a title could be suitable for the segment but appears to be too specific or too generic or if *(iii)* it does not reflect the segment. Indirectly, this three scale annotation allows to assess to which extent a press article and a TVBN topic segment can be linked. The 3-level scale is analogous to the scale used for the question-question textual similarity task described in section 2.2..

In order to facilitate the annotation process, we have defined a strategy to reduce the amount of potential titles to evaluate: only the titles of articles of the day having at least two content words in common with the news segment are proposed to the annotator. Hence, on average the annotator had to evaluate 127 titles by TVBN topic segment. If the annotation has been performed with a scale of three possible values, we consider in the rest of the article that a TVBN segment and a press article can be linked if the title satisfies condition *(i)* (the title is suitable for the segment).

The same human annotator as the one of the first annotation phase has linked press articles and segments according to the 3 values. Time spent on this second annotation phase is about less than 2 months (about 0.25k hours).

5. Corpus statistics

5.1. TV Broadcast News data

Table 1 describes our two sub-corpora in terms of number of topic boundaries, number of segments and average segment duration. In several studies we have observed that segment duration can have a significant impact on downstream modules performances. When analyzing the duration distribution, it appears that they can be easily divided into two sets, where the threshold between *short* and *long* segments is set to $30s$. Table 2 describes the two sub-corpora in terms of segment duration and type. Note that the longest segment in the TV_W07_14 is an exception and corresponds to a long report on a particular society subject inserted in the Sunday midday show.

Corpus	TV_W07_14	TV_W05_15
Duration	23.3h	9.9h
# boundaries	895	271
# segments	997	297
# *long* segments	761	227
# *short* segments	236	70
av. segment duration	105.1	120.5
(min, max)	(5.2, 1145.4)	(5.1, 655.5)

Table 1: TVBN corpus description

5.2. Press articles

The data collection process described in section 3.2. results into a database of 22141 entries after suppressing duplicates and erroneous exports (dead urls or irrelevant contents): 17324 articles for PRESS_W07_14 and 4821 for PRESS_W05_15. On average around 2460 press articles are available for each day. When reducing to the first article of each Goggle News cluster, this figure goes down to 590 candidates for linking on average each day. The full corpus can be used to train vectorial representations as in (Bouchekif et al., 2015) where we show that using semantic relations derived from word embeddings could help for the topic segmentation task. To this purpose table 3 provides figures in terms of number of words in the overall press articles sub-corpora.

5.3. Linking statistics

As a result of the linking annotation process, TVBN topic segments can be separated into two sets. MATCH (M) contains segments that can be linked to at least one press article. NOMATCH ($\overline{M}$) contains segments that have no corresponding article in the candidate list. Table 4 provides detailed description of the cross-modal linking annotation for the TV_W07_14 sub-corpus.

Interestingly, short TVBN topic segments are more likely to be linked with a press article. Actually they usually correspond to a brief recall of news such as sport results, international events, *etc.* that are also referred to in press articles. Longer topic segments on TVBN are more likely to correspond to a particular seasonal report or a general society question illustrated in a particular place, that is not necessarily relevant for a press article or not necessarily put forward by the Google News algorithms.

Beyond $segment \leftrightarrow article$ linking, other linking tasks can be derived. Namely it is possible to derive $segment \leftrightarrow segment$ linking by identifying segments that share similar articles in their list of linked articles. Conversely we can derive $article \leftrightarrow article$ linking by looking for articles that are linked to a same TVBN topic segment. In our TV_W07_14 sub-corpus, out the 658 segments of the MATCH set, 489 segments (74%) can be linked to at least one other segment and among them 275 can be linked to more than 5 other articles. Conversely for the same period, out of the 4129 candidate articles for linking, 1605 articles (39%) have been linked to at least one TVBN topic segment. Among them, 469 articles can be linked to more than five other articles. These figures indicate that our corpus would be well suited to study graph-linking of a set of news fragments.

6. A variety of addressable tasks

This corpus enables to develop and evaluate several tasks such as thematic segmentation and semantic-based content

show	average segment duration (min; max)		
	TV_W07_14	TV_W05_15	Type
Arte_LeJournal	123.6 (19.7 ; 341.4)	158.9 (14.0; 452.1)	T
D8_LeJournal	170.0 (39.9 ; 258.2)	-	T
Euronews_LeJournal	68.8 (15.5 ; 159.2)	67.6 (19.0 ; 314.4)	M
France2_7Heures	62.7 (9.5 ; 290.0)	59.8 (13.7 ; 152.4)	M
France2_8Heures	56.5 (9.6 ; 137.4)	61.6 (12.8 ; 151.6)	T
France2_13Heures	147.4 (8.7 ; 447.3)	190.3 (10.8 ; 560.7)	T
France2_20Heures	126.2 (9.9 ; 447.5)	176.5 (11.7 ; 655.6)	T
France3_12/13Heures	84.9 (11.4 ; 274.2)	118.4 (15.2; 320.9)	T
France3 _19/20Heures	101.2 (13.8 ; 258.2)	132.7 (18.0; 315.2)	T
M6_12h45	81.7 (8.8 ; 232.9)	105.9 (17.3 ; 300.0)	T
M6_19h45	92.0 (20.7 ; 373.7)	117.6 (18.8 ; 360.0)	T
TF1_13Heures	126.5 (5 .2 ; 1145.4)	113.1 (5.1 ; 265.2)	T
TF1_20Heures	121.8 (15.3 ; 451.7)	131.3 (14.1 ; 348.9)	T
NT1_LeJournal	54.6 (15.9 ; 107.0)	68.2 (21.0 ; 83.2)	M

Table 2: Description of TVBN shows in terms of segment duration and type (**T** for Traditional and **M** for Modern)

Corpus	PRESS_W07_14	PRESS_W05_15
# articles	17324	4821
# words	378.6K	109.8K

Table 3: Press articles corpus description

Corpus	MATCH (M)	NOMATCH ($\overline{M}$)
# segments	658	339
# *long* segments	467	294
# *short* segments	191	45

Table 4: TV_W07_14 sub-corpus description in terms of linking

linking. In this section we provide a few results on different tasks that can be considered as baseline for further developments.

6.1. Topic Segmentation

The *FrNewsLink* corpus has been extensively used to develop and evaluate topic segmentation (Bouchekif et al., 2015). The approach was inspired by TextTiling (Hearst, 1997), a sliding window based algorithm, with several major improvements, mainly in the representation of segments and in the similarity measure between 2 consecutive windows. The similarity measure is computed between vectors encoding not only the words in the window but also the speakers who uttered the words. What's more, the similarity score embeds a semantic relation matrix in order to take into account words which are semantically related. The semantic relation matrix is based on word similarity, where words are represented by their embeddings, estimated on the set of the news articles of the same day, through word2vec tools (Mikolov et al., 2013).

When evaluating topic segmentation as a boundary detection task, with a tolerance margin of 10s around the actual boundary, this system yields a F-measure of 76.3% decomposing into 73.6% recall and 79.1% precision on the TV_W07_14 sub-corpus. More recently (Bouchekif et al., 2017) introduced a new metric in order to evaluate topic segmentation as a segment retrieval task and showed that this model has a retrieval score of 66.3% in terms of number of segments and a retrieval score of 75.1% in terms of segment duration.

6.2. Content linking by textual semantic similarity

For each topic segment S from the speech-data, a list $Accept(S)$ (possibly empty) of news texts, which are semantically similar, is provided.

Let M be the set of speech segments which have non-empty $Accept(S)$ and $\bar{M}$ be the set of speech segments without any corresponding article.

Note that, if one restricts to the news text ranked in first position, and the semantic similarity score between this text and the topic segment is above a certain threshold, the title of this news text can be considered as an acceptable title for the topic segment. This was the paradigm of the topic titling task proposed in (Bouchekif et al., 2016)

6.2.1. Ranking by semantic textual similarity

Content linking can be seen as a ranking task: for a given topic segment, rank all the news texts of the same day, in order to retrieve the news texts that are semantically similar to the topic segment. The semantically similar news texts ($Accept(S)$) must be ranked before the other ones. Such ranking can be evaluated in the usual way in Information Retrieval with Mean Average Precision (MAP) or Mean Reciprocal Rank (MRR). We perform experiments on textual similarity scoring. Speech segments and news articles are represented as weighted bags of selected lemmas. The selected lemmas are names, adjectives and non-auxiliary verbs, and the weights are $Okapi - BM25$ weights (Jones et al., 2000). For speech segment, Okapi computation is performed considering each topic segment in a given TVBN show as a document, and the whole TVBN show as the collection. For news article, Okapi computation is

performed considering all the articles of the same day as the collection. Ranking of news articles is done, based on the cosine similarity score between weighted bags of selected lemmas of the given speech segment and the article. For each speech segment, the similarity score is computed with all the articles of the same day (on average 590 articles). We compute a MAP@10, *i.e.* a MAP evaluation restricted to the list of the 10 first ranked news articles. In this task, the focus is made on ranking the articles according to their similarity score, and not on the thresholding of this score. Thus, the MAP evaluation is restricted to the set of 658 segments in M (subset for which $Accept(S)$ is non-empty). The obtained MAP@10 in such configuration is 83.7%.

6.2.2. Titling

In (Bouchekif et al., 2016), it is proposed to assign to a speech segment the title of the news article of the same day, which is the most similar to the speech segment and whose similarity is above a given threshold. As all speech segments in the corpus don't have a corresponding article, 3 kinds of errors can occur. A substitution (Sub) occurs when a segment S of M is assigned a title that doesn't belong to $Accept(S)$, a false rejection (FR) occurs when a segment S of M is not assigned any title and a false alarm (FA) occurs when a segment S from $\bar{M}$ is assigned a title. The Titling Error Rate (TER) is defined as follows :

$$TER = \frac{\#Sub + \#FR + \#FA}{|M| + |\bar{M}|} \quad (1)$$

One can also adopt the evaluation framework proposed for answer triggering in (Yang et al., 2015): the goal is to detect whether a speech segment has a corresponding article in the news corpus, and return the best corresponding article if there exists such one. Thus, the task can be evaluated as the detection of correctly titled speech-segments, with conventional metrics for detection evaluation: precision, recall and f-measure.

In (Bouchekif et al., 2016), on the set of 997 speech segments of TV_W07_14, the best results were obtained when computing a cosine similarity score between weighted bags of selected lemmas of the speech segment and of the news article (as explained in section 6.2.1.). The optimal threshold on this score gives, for reference topic segments, a TER of 11.8%, with $Sub = 3.9\%$, $FA = 3.8\%$ and $FR = 1.1\%$, which translates, for the detection of correctly titled segments, in $Re = 87.8\%$, and $Pr = 88.2\%$. Thus, for manually segmented topic segments, the results, with a simple textual similarity performs pretty well, but there is still room for improvement. (Bouchekif et al., 2016) extends the evaluation to the case of automatically segmented segments, with a global segmentation and titling error rate, where the titling error is computed only if the segmentation reaches a sufficient level of quality.

7. Conclusion

This article has presented a new resource that enables to study topic segmentation and semantic similarity through linking tasks (including cross-modality linking). Along several consecutive days, it gathers TVBN shows and press articles. Topic segments from the TVBN shows are linked with press articles of the same day whenever associations are possible. Further annotations could be added to precisely specify the links between segments and articles. Furthermore linking was restricted to articles and segments produced during the same day but studying the evolution of a topic across days could also be annotated and studied on the basis of the *FrNewsLink* corpus.

8. Bibliographical References

Aker, A., Kurtic, E., Hepple, M., Gaizauskas, R., and Di Fabbrizio, G. (2015). Comment-to-article linking in the online news domain. In *Proceedings of the SIGDIAL 2015 Conference*, pages 245–249. ACL.

Bois, R., Gravier, G., Jamet, É., Morin, E., Robert, M., and Sébillot, P. (2017a). Linking multimedia content for efficient news browsing. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, ICMR 2017, Bucharest, Romania, June 6-9, 2017*, pages 301–307.

Bois, R., Vukotić, V., Simon, A.-R., Sicre, R., Raymond, C., Sébillot, P., and Gravier, G., (2017b). *Exploiting Multimodality in Video Hyperlinking to Improve Target Diversity*, pages 185–197. Springer International Publishing, Cham.

Bouchekif, A., Damnati, G., Estève, Y., Charlet, D., and Camelin, N. (2015). Diachronic semantic cohesion for topic segmentation of TV broadcast news. In *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, pages 2932–2936.

Bouchekif, A., Damnati, G., Charlet, D., Camelin, N., and Estève, Y. (2016). Title assignment for automatic topic segments in TV broadcast news. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, March 20-25, 2016*, pages 6100–6104.

Bouchekif, A., Charlet, D., Damnati, G., Camelin, N., and Estève, Y. (2017). Evaluating automatic topic segmentation as a segment retrieval task. In *Interspeech 2017*, pages 2924–2928.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Choi, F. Y. Y. (2000). Advances in domain independent linear text segmentation. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, NAACL 2000, pages 26–33.

Das, M. K., Bansal, T., and Bhattacharyya, C. (2014). Going beyond corr-lda for detecting specific comments on news & blogs. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 483–492. ACM.

Eisenstein, J. and Barzilay, R. (2008). Bayesian unsupervised topic segmentation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, October.

Eskevich, M., Aly, R., Racca, D., Ordelman, R., Chen, S.,

and Jones, G. J. (2014). The search and hyperlinking task at mediaeval 2014.

Galley, M., McKeown, K., Fosler-Lussier, E., and Jing, H. (2003). Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 562–569.

Guinaudeau, C. (2011). *Structuration automatique de flux télévisuels*. Ph.D. thesis, Institut National des Sciences Appliquées de Rennes, France, December.

Guo, W., Li, H., Ji, H., and Diab, M. T. (2013). Linking tweets to news: A framework to enrich short text data in social media. In *ACL (1)*, pages 239–249.

Hearst, M. A. (1997). Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

Jones, K. S., Walker, S., and Robertson, S. E. (2000). A probabilistic model of information retrieval: development and comparative experiments: Part 1. *Information processing & management*, 36(6):779–808.

Kohlschütter, C., Fankhauser, P., and Nejdl, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, USA*, pages 441–450.

Malioutov, I. and Barzilay, R. (2006). Minimum cut model for spoken lecture segmentation. In *Proceedings of 44th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, Sydney, Australia, July.

Meignier, S. and Merlin, T. (2010). Lium spkdiarization: an open source toolkit for diarization. In *CMU SPUD Workshop*, volume 2010.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*.

Mougard, H., Riou, M., de la Higuera, C., Quiniou, S., and Aubert, O. (2015). The paper or the video: Why choose? In *Proceedings of the 24th International Conference on World Wide Web*, pages 1019–1022. ACM.

Nakov, P., Hoogeveen, D., Màrquez, L., Moschitti, A., Mubarak, H., Baldwin, T., and Verspoor, K. (2017). Semeval-2017 task 3: Community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 27–48.

Peddinti, V., Povey, D., and Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts. In *INTERSPEECH*, pages 3214–3218.

Rosenberg, A. and Hirschberg, J. (2006). Story segmentation of broadcast news in english, mandarin and arabic. In *Proceedings of the Human Language Technology Conference of the NAAC*, pages 125–128.

Shriberg, E., Stolcke, A., Hakkani-Tür, D. Z., and Tür, G. (2000). Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, 32(1-2):127–154.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The icsi meeting recorder dialog act $mrda$ corpus. Technical report.

Tomashenko, N., Vythelingum, K., Rousseau, A., and Estève, Y. (2016). Lium asr systems for the 2016 multi-genre broadcast arabic challenge. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pages 285–291. IEEE.

Xie, L., Yang, Y., Liu, Z.-Q., Feng, W., and Liu, Z. (2010). Integrating acoustic and lexical features in topic segmentation of chinese broadcast news using maximum entropy approach. In *Audio Language and Image Processing (ICALIP)*, pages 407–413, Nov.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, pages 2013–2018.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

PyRATA, Python Rule-based feAture sTructure Analysis

Nicolas Hernandez and Amir Hazem
Laboratoire des Sciences du Numérique de Nantes (LS2N)
Université de Nantes, 44322 Nantes Cedex 3, France
Nicolas.Hernandez@univ-nantes.fr

Abstract

In this paper, we present a new Python 3 module named *PyRATA*, which stands for "*P*ython *R*ule-based fe*A*ture s*T*ructure *A*nalysis". The module is released under the Apache V2 license. It aims at supporting rules-based analysis on structured data. PyRATA offers a language expressiveness which covers the functionalities of all the concurrent modules and more. Designed to be intuitive, the pattern syntax and the engine API follow existing standard definitions; Respectively Perl regular expression syntax and Python re module API. Based on simple native Python data structures (i.e. sequence of feature sets), PyRATA is able to deal with various kinds of data (textual or not) at various levels, such as a list of words, a list of sentences, a list of posts in a forum thread, a list of events in a calendar... This specificity makes it free from any (linguistic) process.

Keywords: rules-based analysis, semantic annotations, regular expression, information extraction, Python 3

1. Introduction

Rules-based approaches must not be set in opposition to machine-learning-based (ML) approaches. The former provide the advantages of quickly generating some self-explanatory results with a few rules, even with little expert knowledge and no training data. But they have the pitfalls of becoming difficult to maintain with the number and the complexity of rules increasing. Cleverly trained, ML models are capable of generalizing i.e. they present the ability to perform well on new unseen data and offers so a larger recall performance. But their major drawbacks are to depend on large quantity of training data which often results from a costly manual annotation task. In addition, their decisions are harder to explain.

Rules are currently unpopular in Natural Language Processing (NLP) research community, but there are still good reasons to use them: 1) Since they do not require training data, they are often a good first cut to explore the data and define more precisely a problem; 2) In some languages, some problems can be done deterministically with rules pretty well; 3) To prototype a model by using rules and use it to generate training data for a ML system; 4) To define features (feature engineering) and let the ML component learn how to combine them; 5) To augment ML models with rules in pre- or post-processing stages to achieve an algorithm of 100 % performance (Manning, 2011). The rules are so used to tune input/output and handle some specific unwanted system behaviors.

To the best of our knowledge, the NLP community benefits from two software solutions[1] which allow to define patterns of annotations with some additional actions to perform on the matched annotations, namely *GATE JAPE*[2] (Cunningham et al., 1999) and *UIMA RUTA*[3] (Kluegl et al., 2016). Apart from learning the rule languages, the use of these options is not straightforward since both of them are part of a global text analysis framework, and consequently assume a basic understanding of the framework concepts as well as some technical skills (e.g. Eclipse workbench). Accidentally the programmer would have to develop in Java to integrate them in their own solution.

Python users do not benefit from the same advanced tools. At least they have some modules to formulate search patterns and extract resulting matches namely the *Python nltk chunk* module (Bird, 2006), the *clips pattern.search* (De Smedt and Daelemans, 2012) and the *spaCy* module.

In this paper, we present a new Python 3 module named *PyRATA*, which stands for "*P*ython *R*ules-based fe*A*ture s*T*ructure *A*nalysis". The module is released under the Apache V2 license. It aims at supporting rules-based analysis on structured data. PyRATA offers a language expressiveness which covers the functionalities of all the concurrent modules and more. Designed to be intuitive, the pattern syntax and the engine API follow existing standard definitions; Respectively Perl regular expression syntax and Python re module API. Using a simple native Python data structure (i.e. sequence of feature set) allows it to deal with various kinds of data (textual or not) at various levels, such as a list of words, a list of sentences, a list of posts of a forum thread, a list of events of a calendar... This specificity makes it free from any (linguistic) process.

Section 2. describes the PyRATA pattern language, the module API and its implementation. And Section 3. discusses PyRATA with respect to the other various Python alternatives.

2. PyRATA module

Regular expressions (RE) are traditionally known to define patterns of possible sequences of characters, which are in turn used by search algorithms on strings for actually finding matching sequences. In Natural Language Processing (NLP), RE are the essence of the rules-based approach for text analysis.

[1] We did not consider here environments such as Nooj (Silberztein, 2005) http://www.nooj4nlp.net or Unitex (Paumier et al., 2009) http://unitexgramlab.org which are deeply rooted in linguistic analysis.

[2] https://gate.ac.uk/sale/tao/splitch8.html, Java 8, GNU

[3] https://uima.apache.org/ruta.html, Java 8, Apache v2

```
1 >>> import nltk
  >>> data = 'Chuck Norris is cooler than Dolph Lundgren.'
3 >>> pyrata_data = [{'raw':word, 'pos':pos, 'lem':nltk.WordNetLemmatizer().lemmatize(word.
    lower())} for (word, pos) in nltk.pos_tag(nltk.word_tokenize(data))]
```

Figure 1: Generating a PyRATA data structure from a String (here used to store a simple sentence) using Python and the nltk module.

```
1 >>> pyrata_data
  [{'raw': 'Chuck', 'pos': 'NNP', 'lem': 'chuck'},
3  {'raw': 'Norris', 'pos': 'NNP', 'lem': 'norris'},
   {'raw': 'is', 'pos': 'VBZ', 'lem': 'is'},
5  {'raw': 'cooler', 'pos': 'JJR', 'lem': 'cooler'},
   {'raw': 'than', 'pos': 'IN', 'lem': 'than'},
7  {'raw': 'Dolph', 'pos': 'NNP', 'lem': 'dolph'},
   {'raw': 'Lundgren', 'pos': 'NNP', 'lem': 'lundgren'},
9  {'raw': '.', 'pos': '.', 'lem': '.'}]
```

Figure 2: Example of a PyRATA Data Structure resulting from the process described in Figure 1.

2.1. The data structure

A string can be seen as somehow a list of character tokens. But a character string is a poor data structure while in many cases the matter of interest are linguistic phenomena at various text levels (e.g. words, sentences, ...).

The main innovative feature of PyRATA is may be to deal with *lists* of *associative arrays* (i.e. `list` of `dicts` in the Python jargon). The dict data structure is a set of values indexed by unique keys (i.e. a name-value feature set).

Figure 2 illustrates the data structure[4] the PyRATA engine takes in input. Line 3 Figure 1 shows how to generate it in one simple instruction line in Python; Here thanks to the nltk module. The process takes the `String` sentence in line 2, then performs a tokenization, pos tagging and lemmatization, and stores the result in a `list` where each value is a `dict` which represents the various forms of a word of the given sentence. The names, `raw`, `pos`, and `lem`, are freely chosen to mean the feature keys.

2.2. Pattern syntax

The pattern syntax used by PyRATA is a subset of the PERL regular expression syntax. The subset is also common to the POSIX extended syntax. This subset suffices to describe all regular languages. In some aspects, the syntax looks like the language used to define queries in Corpus Query Processor (CQP) of the IMS Open Corpus Workbench (CWB) (Christ, 1994)[5].

A *pattern* can be seen as a sequence of elements separated by whitespace characters. In its minimal form, an *element* specifies a *constraint* on the value of a feature that a data token should satisfy. For a given feature name, you can specify its required exact value (e.g. `raw="Chuck"`), a regular expression definition of its value (e.g. `pos~"JJ.?"`), a list of possible values –a lexicon– (e.g. `lemma@"POSITIVE"`) or if the value correspond to a IOB[6] tag and should match a sequence consequently (e.g. `chunk-"NP"`).

More complex elements are quantified elements, element classes, groups of elements, alternatives or combinations of these various types. A *quantified element* allows to specify *optional* elements (`?`), elements which should occur *at least one* (+), or *zero or more* (⋆). An *element class* aims at specifying a logical combination of minimal constraints on the data token features. The combination is delimited by squared brackets (`[]`) and the constraints are combined with usual logical operators namely *parenthesis* (`()`) and logical connectors such as *and* (&), *or* (|) and *not* (!). A *group of elements* is delimited by parenthesis (`()`) and can be used to refer to and process subsequently subparts of a match. An *alternative* defines expected sets of data token candidates at a specific point of the data stream. The *wildcard element* can be set by a dot character.

Line 1 and 2 of Figure 3 show some concurrent patterns to define noun phrases chunks. They illustrate the use of equal and regular expression constraints as well as element classes and quantifiers. After importing the PyRATA module (line 3), line 4 shows the search instruction of the latter pattern over the previously defined data structure. Figure 4 shows the result as a list of `Match` objects, each one made of the matched sequence and its offsets.

2.3. Finding and editing operations

The API is developed to be familiar for whom who develops with the *python re module*[7] API. The module defines several common functions such as `search`, `findall`, or `finditer`. The functions are also available for compiled regular expressions. They generally take at least two arguments including the pattern to recognize and the data to explore (e.g. `re.search(pattern, data)` or `compiledPattern.search(data)`). Depending

[4] In Python, squared brackets delimit list while curly brackets mark dicts. The feature of a dict are separated by a comma and the feature name and its value are separated by a colon.

[5] `cwb.sourceforge.net`

[6] I for Inside, B for Begin and O for Other

[7] `https://docs.python.org/3/library/re.html`, Python 3, PSF

```
>>> pyrata_np_pattern = 'pos="DT"? [pos="JJ"|pos="NN"]* [pos="NN"|pos="NNS"|pos="NNP"]+'
>>> pyrata_np_pattern = 'pos="DT"? pos~"JJ|NN"* pos~"NN.?"+'
>>> import pyrata.re as pyrata_re
>>> found_noun_phrases = pyrata_re.finditer(pyrata_np_pattern , pyrata_data)
```

Figure 3: Recognizing simple Noun Phrases patterns with PyRATA (taking the PyRATA data structure of Figure 1 as input).

```
>>> found_noun_phrases
<pyrata.re MatchesList object; matcheslist="[
  <pyrata.re Match object; groups=[[[{'raw': 'Chuck', 'pos': 'NNP', 'lem': 'chuck'}, {'raw
    ': 'Norris', 'pos': 'NNP', 'lem': 'norris'}]], 0, 2]]>,
  <pyrata.re Match object; groups=[[[{'raw': 'Dolph', 'pos': 'NNP', 'lem': 'dolph'}, {'raw
    ': 'Lundgren', 'pos': 'NNP', 'lem': 'lundgren'}]], 5, 7]]>
]">
```

Figure 4: Display of a PyRATA `MatchesList` object resulting from the `finditer` operation in Figure 3.

on the selected function, the result can be a featured object which stands for the first match of a given pattern, all the non-overlapping matches of a pattern, or an iterator yielding match objects over all non-overlapping matches of a pattern. Figure 3 and 4 illustrate the `finditer` method returning an iterator yielding match objects.

In addition to the exploration methods, the module offers methods to edit the data structure either by substitution (`sub`), update (`update`) or extension (`extend`) of the data feature structures. These methods are very common in NLP annotation tasks.

Figure 2 shows that the nltk lemmatizer did not assign the correct lemma to the verb *to be*. Figure 5 illustrates how to easily fix this hassle thanks to the `update` method. Figure 6 illustrates the annotation method `extend` to add new features to token in order to create IOB tags corresponding to a noun phrase (NP) chunk. It uses the NP pattern defined in Figure 3. Since the resulting data structure is also a PyRATA data structure, such methods can be used to simulate transducers and rewriting rules.

Line 1 Figure 7 defines two lexicons while Line 2 both illustrates the combination in one pattern of the lexicons and IOB-chunk constraints, as well as the use of a sequence group. PyRATA assigns a group id to the chunks. Group 0 corresponds to the whole match.

More examples are available in the user documentation[8] and the demo directory of PyRATA.

2.4. Implementation

Since version `v0.4`, PyRATA has been based on a simple implementation[9] of Thompson's algorithm which aims at converting Regular Expressions (RE) to Non-deterministic Finite Automata (NFA) and simulate them in a multiple-state approach which offers a linear time efficiency in $\mathcal{O}(n)$ (Thompson, 1968).

The following description is based on the explanations of Russ Cox[10]. Every regular expression has an equivalent NFA. The NFA is built up from partial NFAs for each subexpression, with a different construction for each operator. The partial NFAs have no matching states: instead they have one or more dangling arrows, pointing to nothing. The construction process will finish by connecting these arrows to a matching state.

Running an NFA using some data as input requires tracking the current states that the NFA is in, and the next set of states that the NFA will be in, after processing the current token. The pattern is matched if the NFA reaches a final state.

For a regular expression of length m run on text of length n, the Thompson NFA requires $\mathcal{O}(mn)$ time. Since we only need to scan the pattern string once in order to build the corresponding NFA, the time efficiency for this step is linear $\mathcal{O}(n)$.

3. Comparison to alternatives

This section deals with the differences between the various Python alternatives to perform regular expressions over annotations.

Table 1 summarizes the main functionalities of the alternatives Python modules. Data structure dependency is set to dependent if the module works on an internal data structure. Concerning *clips.pattern* there was no successful straightforward `2to3` conversion.

The Python *nltk chunker*[11] module (Bird, 2006) offers to define chunk and chink grammars on word POS tags consisting of rules (expressed in regular-expression style) that indicate how sentences should be chunked (e.g. `"NP: {<DT>?<JJ>*<NN>}"`). By respecting the (Python) data structure, the engine can be applied on vari-

[8] `https://github.com/nicolashernandez/PyRATA/blob/master/docs/user-guide.rst`

[9] Gui Guan, "A Beautiful Linear Time Python Regex Matcher via NFA", August 19, 2014, `https://www.guiguan.net/a-beautiful-linear-time-python-regex-matcher-via-nfa/`, free to use and modify in any way.

[10] Russ Cox, "Regular Expression Matching Can Be Simple And Fast (but is slow in Java, Perl, PHP, Python, Ruby, ...)', January, 2007, `https://swtch.com/~rsc/regexp/regexp1.html`, Last consulted on September 30th, 2017

[11] `http://www.nltk.org/book/ch07.html` and `http://www.nltk.org/_modules/nltk/chunk/regexp.html`, Python 3, Apache v2

```
>>> updated_pyrata_data = pyrata_re.update ('[lem="is" | lem="are" | lem="wa"]', {'lem':'
    be'}, pyrata_data)
[{'raw': 'Chuck', 'pos': 'NNP', 'lem': 'chuck'},
 {'raw': 'Norris', 'pos': 'NNP', 'lem': 'norris'},
 {'raw': 'is', 'pos': 'VBZ', 'lem': 'be'},
 {'raw': 'cooler', 'pos': 'JJR', 'lem': 'cooler'},
 {'raw': 'than', 'pos': 'IN', 'lem': 'than'},
 {'raw': 'Dolph', 'pos': 'NNP', 'lem': 'dolph'},
 {'raw': 'Lundgren', 'pos': 'NNP', 'lem': 'lundgren'}]
```

Figure 5: Updating (edit operation) a PyRATA data structure by modifying a feature value. The PyRATA data structure of Figure 1 is taken as input.

```
>>> extended_pyrata_data = pyrata_re.extend (pyrata_np_pattern, {'chunk':'NP'},
    updated_pyrata_data, iob=True)
[{'raw': 'Chuck', 'chunk': 'B-NP', 'pos': 'NNP', 'lem': 'chuck'},
 {'raw': 'Norris', 'chunk': 'I-NP', 'pos': 'NNP', 'lem': 'norris'},
 {'raw': 'is', 'pos': 'VBZ', 'lem': 'be'},
 {'raw': 'cooler', 'pos': 'JJR', 'lem': 'cooler'},
 {'raw': 'than', 'pos': 'IN', 'lem': 'than'},
 {'raw': 'Dolph', 'chunk': 'B-NP', 'pos': 'NNP', 'lem': 'dolph'},
 {'raw': 'Lundgren', 'chunk': 'I-NP', 'pos': 'NNP', 'lem': 'lundgren'}]
```

Figure 6: Extending (edit operation) a PyRATA data structure by adding a new feature. The PyRATA data structure of Figure 1 is taken as input.

```
>>> lexicons = {'POSITIVE':['cooler', 'stronger'], 'NEGATIVE':['poor', 'worst']}
>>> pyrata_re.finditer('chunk-"NP" (lem="be" [raw@"POSITIVE"  & !raw="than"]* raw="than")
    chunk-"NP"' , extended_pyrata_data, lexicons = lexicons).group().group(2)
[{'raw': 'is', 'pos': 'VBZ', 'lem': 'be'}, {'raw': 'cooler', 'pos': 'JJR', 'lem': 'cooler'
    }, {'raw': 'than', 'pos': 'IN', 'lem': 'than'}]
```

Figure 7: Defining groups and lexicons in PyRATA syntax.

ous types of information (one at a time) as well as be cascaded.

The *CLIPS pattern.search*[12] module (De Smedt and Daelemans, 2012) has a pattern matching system similar to regular expressions, that can be used to search a string by syntax (word POS and chunk tags) or by lexical (word surface, lemmata and semantical categories) constraints (e.g. "`DT? JJ?+ NN`"). *CLIPS pattern.search* is actually the best choice in terms of pattern language expressiveness. Unfortunately, the language and the engine are not well separated from the clips internal data structure which prevents from using it with external processing.

The *spaCy*[13] module features a rule-matching engine that operates over tokens, similar to regular expressions. The rules can refer to token annotations and flags, and matches support callbacks to accept, modify and/or act on the match. spaCy is written in Python and Cython. Despite its astonishing performance, the use of the spaCy rule-matching engine is not easy out of the box. Indeed, it is strongly attached to the spaCy internal data structure and without entering the code, it cannot go further than addressing simple chunk extractions. In addition, the matches cannot serve as input of new patterns.

Table 2 gives the time performance of searching similar patterns on each module. Due to the expressiveness differences, only a minimal pattern expression can be used to compare the modules. This is the role played by the first line dedicated to each module in the table. This minimal noun phrase definition assumes that an NP is ending by a noun which can be preceded by an optional determiner and zero or more adjectives in this order. Figure 8 depicts the results for this four systems. Other NP definition variants allow to consider various adjective and noun forms as well as nouns as alternative to adjectives. These patterns illustrate the syntax of each module. To address the comparison, we used the Brown corpus as data (1,161,192 words) and measure the average time over 10 runs to process the n first words with n varying from 10,000 to 1,000,000. The experience was performed on a processor i7-4600U running at 2.10GHz. For *nltk chunker* and PyRATA we used the original Brown corpus word tokenization and the default nltk POS tagger in English. In order to use *clip.pattern* and *spaCy*, we joined the Brown words in a single string and performed the tokenization and the pos

[12] `https://www.clips.uantwerpen.be/pages/pattern-search` and `https://github.com/clips/pattern`, python 2.6, BSD-3

[13] `https://spacy.io/docs/usage/rule-based-matching`Â and `https://github.com/explosion/spaCy`, Python 3, MIT

	PyRATA	clips.pattern	nltk_chunk_regexp	spaCy
Python version	3	2	2 and 3	2 and 3
Token type	no restriction	word/chunk token	word/chunk token	word token
Data structure dependency	Python native	dependent	Python native	dependent
Feature type	no restriction	restricted to some syntax and semantic	no restriction but a single one at a time	some predefined, extensible with code
Pattern language	declarative	declarative	declarative	declarative + code
Wildcard	yes	yes	yes	
Quantifiers	yes	yes	yes	yes
Element alternative	yes	yes	yes	no
Group	yes	yes	no	no
Group alternative	yes	no	no	no
Match offsets	yes	no	no	yes
Cascade matcher	yes	no	yes	no
Find operations	search, match, findall	search, match, findall	findall	findall
Edit operations	sub, update, extend	no	extend	no

Table 1: Module functionalities comparison.

noun phrase patterns	10k	50k	100k	200k	300k	500k	750k	1,000k
CLIPS pattern.search								
`DT? JJ?+ NN+`	0.145	0.714	1.510	3.113	4.484	7.585	11.700	15.515
`DT? JJ|NN?+ NN|NNS`	0.212	1.008	2.110	4.675	6.328	10.825	16.712	22.224
`DT? JJ|NN?+ NN|NNS+`	0.220	1.033	2.170	4.493	6.538	11.069	17.929	22.879
nltk chunker								
`<DT>?<JJ>*<NN>+`	0.037	0.521	1.842	7.146	15.194	43.481	97.216	169.396
`<DT>?<JJ|NN>*<NN|NNS>`	0.043	0.652	2.386	9.574	20.130	58.745	127.489	228.397
`<DT>?<JJ|NN>*<NN.*>`	0.062	1.137	4.190	15.502	32.720	89.219	200.388	334.670
PyRATA								
`pos="DT"? pos="JJ"* pos="NN"+`	1.681	8.350	17.089	33.773	53.506	91.840	131.540	176.272
`pos="DT"? [pos="NN" | pos="JJ"]* [pos="NN" | pos="NNS"]`	2.132	10.662	21.413	42.731	65.062	114.885	167.478	222.817
`pos="DT"? [pos~ "NN|JJ"]* pos~"NN.*"`	1.854	9.695	18.872	40.288	58.423	101.974	141.782	189.163
spaCy								
`[{POS:"DET",'OP':"?"}, {POS:"ADJ",'OP':"*"},{POS:"NOUN",'OP':"+"}]`	0.002	0.004	0.008	0.020	0.0266	0.048	0.070	0.086

Table 2: Time performance (in seconds) for recognizing various Noun Phrases patterns on the first n Brown corpus words.

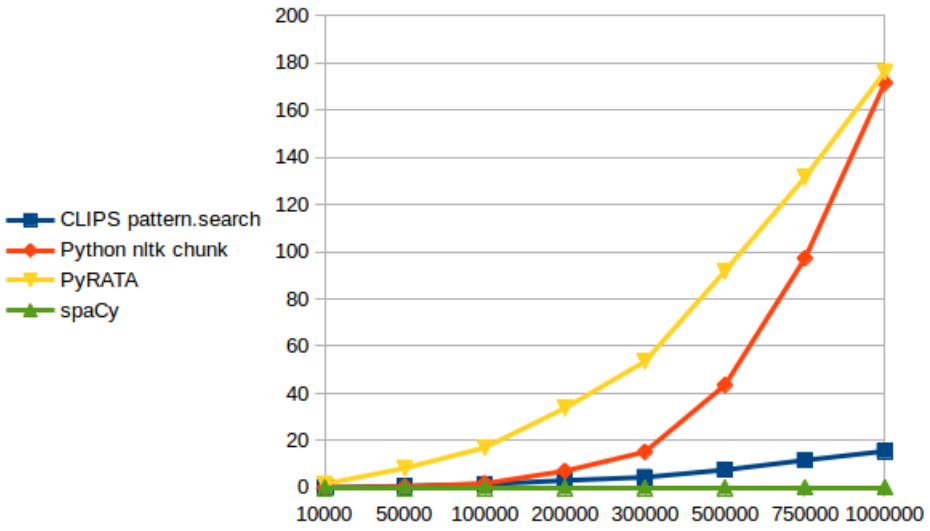


Figure 8: Selection of some system performances from Table 2. Only assumed identical patterns (first line of each module) are presented. Time (in seconds) vs quantity of processed words.

tagging processes offered by these modules. We used the `en_core_web_sm-1.2.0` model for *spaCy*. We could have hacked the *spaCy* tokenizer by making it segment only on the whitespaces and consequently format the Brown corpus so that *spaCy* would eventually obtain the same tokenization as the one already present in nltk. Since the resulting amount of tokens was pretty much the same we did not follow this path. As a result, we observe that there is nothing compare to the time performance of spaCy and right now we can say the same about the PyRATA language expressiveness. Nevertheless, if a few minutes to process one million words can largely be acceptable depending on the use cases it is a fact that the PyRATA holds the worse position in term of time performance. Surprisingly, we also observe that the *nltk chunker* time processing increases and converges toward the PyRATA one. We did not experiment with more data.

4. Conclusion

PyRATA is *fun and easy to use* to explore data for research or pedagogical motivations, define machine learning features, pre/post process machine learning based analysis, formulate expert knowledge in a declarative way.

Various directions exist to optimize further the engine code such as parallel programming, or caching the lists of states corresponding to effectively explored and possibly recurrent DFA (avoiding the cost of future computation repetitions). The language can also be extended like for instance the management of back-references.

Acknowledgments

The current work was supported by the ANR 2016 PASTEL[14]. The authors would like to thank the anonymous reviewers for their valuable comments.

5. Bibliographical References

Bird, S. (2006). Nltk: The natural language toolkit. In *Proceedings of the COLING/ACL on Interactive Presentation Sessions*, COLING-ACL '06, pages 69–72, Stroudsburg, PA, USA. Association for Computational Linguistics.

Christ, O. (1994). A modular and flexible architecture for an integrated corpus query system. In *Proceedings of COMPLEX'94: 3rd Conference on Computational Lexicography and Text Research*, pages 23–32, Budapest, Hungary. tt cmp-lg: tt 9408005.

Cunningham, H., Cunningham, H., and Tablan, V. (1999). Jape: a java annotation patterns engine.

De Smedt, T. and Daelemans, W. (2012). Pattern for python. *Journal of Machine Learning Research*, 13:2063–2067.

Kluegl, P., Toepfer, M., Beck, P.-D., Fette, G., and Puppe, F. (2016). Uima ruta: Rapid development of rule-based information extraction applications. *Natural Language Engineering*, 22:1–40, 1.

Manning, C. D., (2011). *Part-of-Speech Tagging from 97% to 100%: Is It Time for Some Linguistics?*, pages 171–189. Springer Berlin Heidelberg, Berlin, Heidelberg.

Paumier, S., Nakamura, T., and Voyatzi, S. (2009). UNITEX, a Corpus Processing System with Multi-Lingual Linguistic Resources. In *eLexicography in the 21st century: new challenges, new applications (eLEX'09)*, pages 173–175, October.

Silberztein, M. (2005). Nooj: A linguistic annotation system for corpus processing. In *Proceedings of HLT/EMNLP on Interactive Demonstrations*, HLT-Demo '05, pages 10–11, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thompson, K. (1968). Programming techniques: Regular expression search algorithm. *Commun. ACM*, 11(6):419–422, June.

[14] `http://www.agence-nationale-recherche.fr/?Projet=ANR-16-CE33-0007`

Towards Processing of the Oral History Interviews and Related Printed Documents

Zbyněk Zajíc, Lucie Skorkovská, Petr Neduchal, Pavel Ircing, Josef V. Psutka, Marek Hrúz, Aleš Pražák, Daniel Soutner, Jan Švec, Lukáš Bureš and Luděk Müller

University of West Bohemia, Faculty of Applied Sciences,
NTIS - New Technologies for the Information Society
and Department of Cybernetics
Univerzitní 8, 306 14 Plzeň, Czech Republic
{zzajic, lskorkov, neduchal, ircing, psutka_j, mhruz, aprazak, dsoutner, honzas, lbures, muller}@ntis.zcu.cz

Abstract

In this paper, we describe the initial stages of our project, the goal of which is to create an integrated archive of the recordings, scanned documents, and photographs that would be accessible online and would provide multifaceted search capabilities (spoken content, biographical information, relevant time period, etc.). The recordings contain retrospective interviews with the witnesses of the totalitarian regimes in Czechoslovakia, where the vocabulary used in such interviews consists of many archaic words and named entities that are now quite rare in everyday speech. The scanned documents consist of text materials and photographs mainly from the home archives of the interviewees or the archive of the State Security. These documents are usually typewritten or even handwritten and have really bad optical quality. In order to build an integrated archive, we will employ mainly methods of automatic speech recognition (ASR), automatic indexing and search in recognized recordings and, to a certain extent, also the optical character recognition (OCR). Other natural language processing techniques like topic detection are also planned to be used in the later stages of the project. This paper focuses on the processing of the speech data using ASR and the scanned typewritten documents with OCR and describes the initial experiments.

Keywords: historical sources processing, automatic speech recognition, optical character recognition, document processing

1. Introduction

The main objective of the project "System for permanent preservation of documentation and presentation of historical sources from the period of totalitarian regimes" is the research and development of software tools for archiving and providing access to the historical resources gathered within the documentary mission of the Institute for the Study of Totalitarian Regimes (USTR) [1]. This institute studies and impartially evaluates the two totalitarian periods of the history of the Czech Republic: the time of the Nazi occupation (1939-1945) and the time of Communist totalitarian power (1948-1989), examines the anti-democratic and criminal activity of state bodies, especially its security services, as well as other organizations based on its ideology. For that purpose, USTR secures and makes accessible to the public the documents related to those periods of suppressed freedom and converts acquired documents into the electronic form.

Within the documentation activities of the USTR, many documents and recordings are stored on the internal storage of the USTR and are made accessible for the researchers on DVDs or through a rudimentary digital storage services. Only a small fraction of the interviews contains the verbatim text transcription – in most of the cases, the interested scholars must manually sift through the whole recording in order to find any information they desire. Despite these imperfections, the historic resources gathered in this collection are being used by history experts and researchers from not only the Czech Republic but also from other European countries and USA[2].

The main goal of this project is to create an integrated archive of the recordings, searchable text documents, and photographs that would be accessible online and would provide multifaceted search capabilities (including the actual spoken content, name and other biographical information, the relevant time period, etc.). The archive created in such a way would make the work of the researchers more efficient and also would allow a wider scope of interested persons to access these historic resources.

In order to achieve this goal, the methods of the automatic speech recognition (ASR), automatic indexing, search in the stored index , optical character recognition (OCR) and other related techniques of natural language processing will be employed. The rest of the paper describes the first stages of the processing of the interviews with the witnesses of the totalitarian regimes using the ASR system. The witnesses have a specific vocabulary containing archaic words which make the recognition difficult. Further, the first experiment with the automatic processing of the scanned documents is presented, which is a challenging task in these circumstances since the source documents are old and of low quality.

2. The Data

During the years 2008-2015 at least 1000 hours of the audio recordings of the interviews with the witnesses of the totalitarian regimes in Czechoslovakia were created. The structured interview with the witness is ideally conducted

[1] https://www.ustrcr.cz/, http://old.ustrcr.cz/en

[2] http://old.ustrcr.cz/en/international-cooperation

over several sessions. The length of the interview is dependent on the experience of the interviewer and the narrating skills, health condition and life destinies of the interviewee. The word-by-word transcription is currently available for only about 160 audio recordings, the rest is annotated only by the identity of the interviewee.

Furthermore, at least 50 000 paper documents were scanned. These documents are relevant historical text materials and photographs mainly from the home archives of the interviewees or the archive of the State Security. Texts are usually typewritten or even handwritten with poor readability and their scans have really bad optical quality (see examples[3] in Figure 1).

All these data are currently not very well organized. They are just stored in a simple directory structure, each witness's data separated in one folder with subfolders containing audio/video interviews (typically of length in the order of hours, usually without any transcription whatsoever) and scanned relevant documents (photos, investigation files, etc.). The scanned materials almost completely lack the information about their content – some of them are in fact only long uninterrupted sequences of scanned pages from the archive of the State Security, without even the basic description such as which of the pages constitute the same document and where there is boundary between the documents. Therefore our goal is to process these data automatically. First, we plan to use ASR to transcribe and index the content of the audio data. Such automatic processing of the recordings will greatly increase the value of the archive since it will allow the word, phrase and phonetic search in the archive and, as such, will provide researchers a far better orientation in the recorded interviews without the need for costly manual transcription.

Second, the automatic processing of the scanned documents using image processing methods and subsequent natural language processing module will allow for faster and more efficient annotation of individual documents with metadata and will greatly simplify the possible assignment of similarity links among documents, which may be of different media nature (text, audio, or photographs). As a result, the researchers will be able to use a much wider context of the recording.

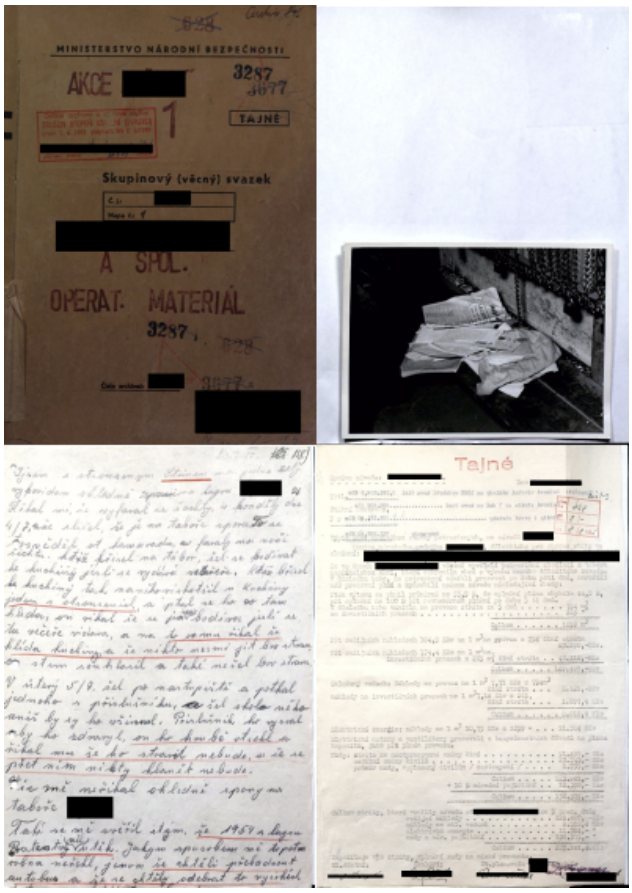


Figure 1: Different examples of the available documents in the database.

[3]All sensitive data have been blacked out from the example pictures before publishing in this paper.

3. Automatic Speech Recognition of Interviews

From the nature of the speech data, it is difficult to design an ASR system with sufficient accuracy. The speakers are usually elderly, their spontaneous speech is often heavily accented, and because of the nature of the stories, they are often very emotional. The speech quality in individual interviews is however very poor from the ASR point of view (many disfluencies and non-speech events as crying, laughter etc.). The speech was also often affected by using many colloquial words. For the above reasons, it is very difficult to find a suitable data source for training an acoustic model. Due to the great acoustic similarity with the MALACH project, we did the first experiment on the audio recordings with the ASR system designed for the MALACH corpus (Psutka et al., 2014). The MALACH contains 400 randomly selected speakers where only 15 minute segment was transcribed per each speaker. On the Czech part of the corpus, our last results reported in (Švec et al., 2017) achieved a word Accuracy (Acc) of 80.89

3.1. Acoustics models

We used the same acoustics model (AM) as in (Švec et al., 2017) in the experiment. That is, we have followed a typical Kaldi (Povey et al., 2011) training recipe (Kamper et al., 2016) for a Deep-Neural-Network-based AM training. This recipe supports layer-wise RBM pre-training, stochastic gradient descent training supported by GPUs and sequence-discriminative training optimizing sMBR criterion. We have applied the standard 6 layers topology (5 hidden layers, each with 2048 neurons) with a softmax layer. We have used features based on a standard 12-dimensional Cepstral Mean Normalized PLP coefficients with first and second derivatives. The model was trained on 84 hours of MALACH recordings.

3.2. Language Models

Since the vocabulary used in the recordings contains many archaic words and rare named entities, the choice of the training data for the Language Model (LM) is crucial for a good performance of the ASR system.

We have tested two LMs. The first one (denoted "MALACH+") was trained on the MALACH corpus com-

plemented by relevant text materials from USTR. This additional corpus consists of the few available transcriptions (excluding the ones that correspond to the test set interviews– see below) and other texts on the same topic (books, articles, etc.). In total, the training corpus contains 1.3M tokens and 70k different words.
The second LM was trained only on the available transcriptions of interviews used for testing of the system ("oracle" LM) to show the current performance upper bound of the ASR system. This model was trained using 10 interviews (a total of 60k tokens and 9k different words).

3.3. Results of ASR

As was mentioned previously, the tests were performed on 10 selected interviews, using our ASR system with the two different LM described in Section 3.2. (and the same AM trained on AMALACH corpus). The results can be seen in Table 1.

LM	Acc [%]
MALACH+	58.00
oracle	80.46

Table 1: Results in terms of word accuracy of the ASR system on a small test set (interviews with 10 witnesses) with two different LMs.

The poor result of our initial experiment is most probably caused by the specific language used in the interviews that is strongly connected with Nazi occupation (1939-1945) and the Communist totalitarian state (1948-1989). It seems that more data for the LM creation must be obtained – the easiest way is the manual transcription of the available interviews – in order to significantly improve the ASR performance. The second experiment with the oracle LM indicates the current capabilities of the acoustic model. However, the AM can be adapted using the available recordings to fit the actual acoustics conditions in the interviews.

4. Optical Character Recognition of the Scanned Typewritten Documents

One of the other subtasks of the project is a transformation of the (poor quality) scans of the typewritten documents into the searchable representation. Essentially, our aim is to convert the image data into the electronic text, i.e., perform the Optical Character Recognition (OCR). Note that such a text constitutes the same type of data as the transcripts obtained from ASR in the way described in Section 3. and thus it can be indexed and searched essentially in the same way.
We have decided to use the Google Tesseract OCR engine (Smith, 2007) and concentrate mainly on the appropriate pre-processing (and, in later stages, post-processing) of the image data. In the very first step of the procedure, we needed to automatically classify the available scans into one of the three classes: typewritten (or printed) document, handwritten document and picture. Our plan for the next stage of the project is to train a Convolutional Neural Network (CNN) for classification of each document into these classes. For training this classifier, we need enough reference data. For this purpose, we used an unsupervised clustering method based on CNN (Alexnet) (Krizhevsky et al., 2012) to obtain visual features that are clustered into three clusters using k-means with Euclidean distance. We identified the clusters representing each class by visual inspection. The clusters contained some false data and they need to be manually corrected. Only the documents containing mostly typewritten (printed) text will be processed in the consequent stages.
In order to improve the results of the OCR engine the preprocessing methods that would remove the noise and other artifacts in the documents were examined (see Section 4.1.).
Because the documents are stored in a single folder for each witness, without any information about the content in a long uninterrupted sequence of scans, we have proposed a method for the decomposition of a scanned folder – i.e. group of scans that belong to the particular witness – into the clusters of related documents and the creation of PDF files based on the clusters (see Section 4.2.).
The OCR transcription of these documents allows to search for important meta-information about documents such as its type, title or the mentioned persons which will be used for forming the structure of the final archive. In the first year of the project, we have performed several experiments mainly focused on the influence of the preprocessing methods for the OCR engine and the clustering of the related documents.

4.1. Preprocessing

During the preprocessing experiment, several methods and their combinations were used (Sonka et al., 2008). One of the most important preprocessing methods is a deskewing (estimation of a skew angle) algorithm. It searches for the rotation that has to be applied in order to get a document with no skew of the contained text. A method based on the Fourier Transform was proposed. This method searches for the skew angle as follows: An input image is transformed into a set of overlapping tiles of a constant size (219 x 248 pixels). Tiles are then sorted in an ascending order with respect to the average value of brightness in the tile. We assume that the tiles with the lowest average brightness value in pixels contain text. The Fast Fourier transform (FFT) is computed over the first eighth tiles in order to get the frequency domain of each tile. An angle is estimated for all frequency domain images. Finally, an average value is computed and the image is rotated by this value. During this process, the interpolation is applied which causes information loss. An example is shown in Figure 2.
The proposed method is promising (see Table 3, line "deskew") but it has worse results than the intern deskew algorithm in the Google Tesseract OCR (line "original RGB image"). Particular deskew algorithm used in Tesseract is unknown for us. Because of the small skew angle in our tested data (smaller than one degree), we deduced that Tesseract is able to process input image without deskewing algorithm – i.e. without information loss. In table 2 we tested our proposed deskew algorithm against the intern one in Tesseract on data with bigger skew angles. In this comparison, our algorithm has much better performance.
Further, we try to extend our preprocessing by binarization method (only with empirically set threshold) for cleaning

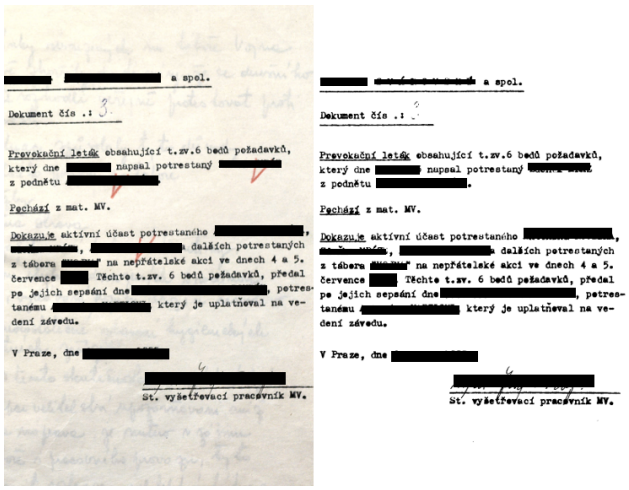


Figure 2: Example of the proposed deskewing approach. Left – the original image, Right – the deskewed and binarized image.

Skew angle approx	Tesseract	Deskew and Tesseract
-4	0.00	84.88
-3	67.13	83.20
-2	64.74	84.15
-1	82.22	86.10
1	85.60	87.25
2	82.77	83.87
3	82.63	84.02
4	78.30	84.76

Table 2: Results in terms of character accuracy of the OCR with our deskew algorithm and with an intern algorithm in Tesseract.

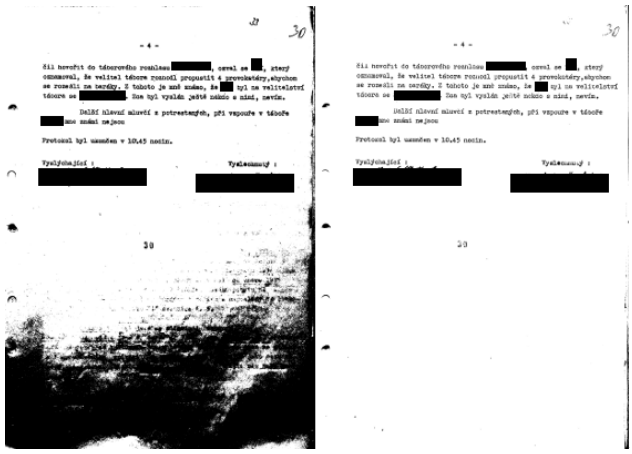


Figure 3: The difference between binarization approaches.

Method	Average accuracy [%]
original RGB image	79.290
deskew	73.378
binary	80.358
binary + deskew	81.838
RGB-binary	**83.753**
RGB-binary + deskew	82.270
bilateral	80.517
bilateral + RGB-binary	80.517
clahe	81.116
clahe + RGB-binary	79.572

Table 3: Results of the preprocessing methods (the average character accuracy of OCR) on the small document's dataset (25 annotated scans containing mostly text).

the noise around the text (line "binary" in Table 3). Next, we propose a different binarization algorithm where each component of an RGB image is binarized independently. The final image is composed of the binarized components as follows:

$$B_{rgb}(i,j) = \begin{cases} 0 & if\ B_r(i,j) = B_g(i,j) = B_b(i,j) = 0, \\ 1 & otherwise, \end{cases} \quad (1)$$

where $B_{rgb}(i,j)$ is a value of the binarized image at coordinates i,j. Values B_r, B_g and B_b respectively contain binary values of the image component at coordinates i,j.

The noise and image artifacts are reduced by this approach more effectively (line "RGB-binary" in Table 3). An example of the difference between the classic grayscale binarization (left) and our proposed method (right) is shown in Figure 3.

Several other methods and their combinations were tested during this experiment, particularly histogram equalization (Pizer et al., 1987) and image smoothing algorithms (Paris and Durand, 2009).

Histogram equalization is based on the method called Contrast Limited Adaptive Histogram Equalization (CLAHE) (Zuiderveld, 1994). This method usually increases the contrast in the image.

Bilateral filtering was used as a representative of the smoothing methods. A bilateral filter is a non-linear, edge-preserving and noise-reducing smoothing filter for images. The intensity value of each pixel in the image is replaced by a weighted average of the intensity values from the nearby pixels. This weight can be based on a Gaussian distribution. Crucially, the weights depend not only on the Euclidean distance of pixels but also on the radiometric differences. This preserves sharp edges by systematically looping through each pixel and adjusting the weights of the adjacent pixels accordingly (Paris and Durand, 2009).

The OCR score with all preprocessing methods was computed using the Levenshtein Distance metric (Yujian and Bo, 2007). It is a character-based evaluation metric that calculates the difference between the two string sequences. The resulting distance is a minimum value of edits - insertions, deletions, and substitutions - that has to be applied in order to transform one string into the other. Based on this distance, the score of the text accuracy can be calculated as follows

$$Acc = \left(1 - \frac{d}{\max\left(lenght(a), length(b)\right)}\right) \cdot 100[\%], \quad (2)$$

where d is the Levenshtein distance, $length(a), length(b)$ is the length of the input string a and b respectively.

4.2. Cutting of Consecutive Documents

Another task of the project is cutting off the consecutive documents. The goal of this task is to divide individual scans into groups that make up a single document in the document folder. The problem arises from the structure of the available data. All relevant documents for the particular witness are scanned and stored in one folder. Our goal is to preprocess all these scans in one folder and save all the documents as separate pdf – i.e. find the boundaries, where one document ends and the other one begins.

For this purpose, the part of data (1/3 of available data) was manually annotated using developed python application, see Figures 4 and 5. Application windows show two consecutive scans from document folder. The user has to make a decision whether they are connected to the single document or not. Annotated data are then used for training the K-Nearest Neighbors algorithm (KNN) classifier.

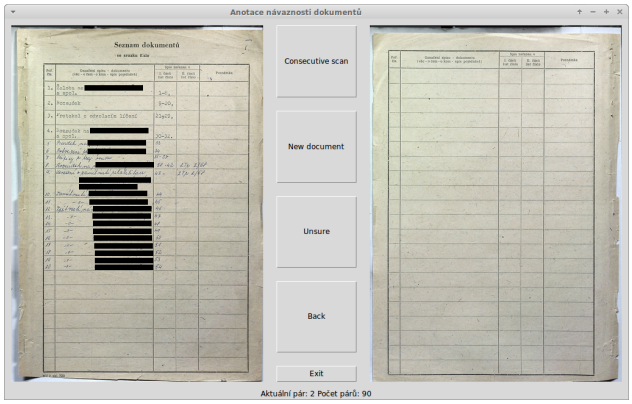


Figure 4: The window of annotation application with example of consecutive scans.

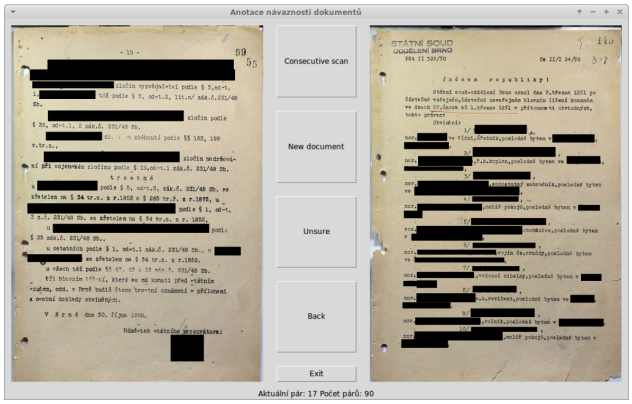


Figure 5: The window of annotation application with example of non-consecutive scans (each from different documents).

The feature vector is composed of the differences between the mean values of color components of a pair of consecutive scans in two color spaces (HSV and RGB). Thus the size of the vector is 6×1. This vector is then clustered using the KNN in order to decide about the consecutiveness of documents. Proposed approach exhibits promising results (evaluated only empirically) and seems to be a good starting point for the next experiments. In the future, we want to focus on extending this method by the OCR text-based decomposition and construction of the structural features of the documents. Also, an experiment with a more sophisticated classifier (e.g. Support Vector Machine, Neural Network) will be performed.

5. Conclusion

This paper described the goals of the project "System for permanent preservation of documentation and presentation of historical sources from the period of totalitarian regimes", the available dataset and the first results of the ASR system and the processing of the scanned documents. Based on these promising results, the subsequent research on the automatic processing of these data to allow the multifaceted search in the dataset is open. Our ASR system used for the transcription of the interviews shows its capabilities for such difficult recordings. For the OCR experiments on the documents, the results were improved by the proposed variant of the preprocessing, necessary for this low quality and high variety of the scanned documents. The proposed methods for the processing of all types of the available data show the first step to fulfill the goal of the project: to create the integrated archive of the recordings and documents that would provide the multifaceted search capabilities.

6. Acknowledgements

This research was supported by the Ministry of Culture Czech Republic, project No.DG16P02B048. Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

7. Bibliographical References

Kamper, H., Wang, W., and Livescu, K. (2016). Deep convolutional acoustic word embeddings using word-pair side information. In *ICASSP*, pages 4950–4954, oct.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *ImageNet Classification with Deep Convolutional Neural Networks*, pages 1–9.

Paris, S. and Durand, F. (2009). A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach. *International Journal of Computer Vision*, 81(1):24–52.

Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J. B., and Zuiderveld, K. (1987). Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlíček, P., Qian, Y., Schwarz, P., Silovský, J., Stemmer, G., and Veselý, K. (2011). The Kaldi Speech Recognition Toolkit. In *Workshop on Automatic Speech Recognition and Understanding*, pages IEEE Catalog No.: CFP11SRW–USB, Hawaii.

Smith, R. (2007). An Overview of the Tesseract OCR Engine. In *Document Analysis and Recognition*, pages 629–633, Parana.

Sonka, M., Hlavac, V., and Boyle, R. (2008). *Image processing, analysis, and machine vision second edition*. Springer, 3 edition.

Švec, J., Psutka, J. V., Šmídl, L., and Trmal, J. (2017). A relevance score estimation for spoken term detection based on RNN-generated pronunciation embeddings. In *Proceedings of Interspeech 2017*, pages 2934 – 2938, Stockholm.

Yujian, L. and Bo, L. (2007). A Normalized Levenshtein Distance Metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1091–1095.

Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. In P.S. Heckbert, editor, *Graphics Gems IV*, chapter VIII. Imag, pages 474–485. Academic Press Professional, Inc., San Diego.

8. Language Resource References

Psutka, Josef and Radová, Vlasta and Ircing, Pavel and Matoušek, Jindřich and Müller, Luděk. (2014). *SC-SFI MALACH Interviews and Transcripts Czech*. Linguistic Data Consortium, USC-SFI MALACH, 1.0, ISLRN 310-213-848-753-5.

Multi Modal Distance - An Approach to Stemma Generation with Weighting

Armin Hoenen
CEDIFOR
Goethe University Frankfurt
hoenen@em.uni-frankfurt.de

Abstract

Stemma generation can be understood as a task where an original manuscript M gets copied and copies – due to the manual mode of copying – vary from each other and from M. Copies $M_1, .., M_k$ which survive historical loss serve as input to a mapping process estimating a directed acyclic graph (tree) which is the most likely representation of their copy history. One can first tokenize and align the texts of $M_1, .., M_k$ and then produce a pairwise distance matrix between them. From this, one can finally derive a tree with various methods, for instance Neighbor-Joining (NJ) (Saitou and Nei, 1987). For computing those matrices, previous research has applied unweighted approaches to token similarity (implicitly interpreting each token pair as a binary observation: identical or different), see Mooney et al. (2003). The effects of weighting have then been investigated and Spencer et al. (2004b) found them to be small in their (not necessarily all) scenario(s). The present approach goes beyond the token level and instead of a binary comparison uses a distance model on the basis of psycholinguistically gained distance matrices of letters in three modalities: vision, audition and motorics. Results indicate that this type of weighting have positive effects on stemma generation.

Keywords: stemmatology, weighted distance, multi modal

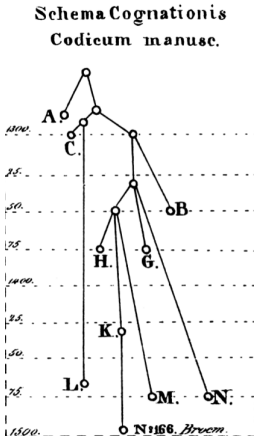


Figure 1: First modern stemma by Schlyter, 1827, from O'Hara (1996) with texts= nodes, copy processes= edges.

1. Introduction

Stemma generation is the process of determining the most likely tree[1] with manuscripts being represented by nodes in the usually directed acyclic graph[2] and edges representing copy processes or chains of such, see Figure 1. Since the late 1950ies computational methods have been applied to stemmatological tasks (Ellison, 1957) and in the last decade evaluation against benchmark datasets (or artificial traditions) has been conducted (Baret et al., 2004; Spencer et al., 2004a; Roos and Heikkilä, 2009; Hoenen, 2015a). These datasets have been generated by first giving one text (root) to volunteers to be handcopied (or dictated). Its copies have then been handcopied again and so forth. The true *vorlage*[3]-copy relations (edges in the true stemma) have been recorded by the authors, so that we know the entire true tree with all edges and the position of root. Those manuscript texts have been digitized and manually aligned for a *stemmatology challenge* (Roos and Heikkilä, 2009) where from a subset of manuscripts several teams attempted to reconstruct – manually or automatically – the true tree.[4] In this paper, these datasets are taken as basis to a new method which uses external data in the form of psycholinguistically generated letter and phoneme distance matrices in order to a) generate and evaluate stemmata and b) assess how large the influence of low level perceptual processes is. From the alignments of the artificial traditions, pairwise distance matrices of the single manuscripts (texts, nodes) are built. Each manuscript pair is compared tokenwise using some metric resulting in an overall distance. This metric can be described as weighted, where the external data serves for determining the weights. Concerning token comparison, philology describes a whole range of types of variation and their implications, see e.g. (Roelli and Macé, 2015; Andrews and Macé, 2013). Philologically motivated classification has been used for weighting token pairs upon distance computation. Categories such as "Word variant, changes meaning" or "Word change affecting rhyme" (Mooney et al., 2003, p.287) have been applied. A stemmatologically relevant distinction and driving force behind the will to weight variants is that between genealogically informative and accidental variation (Andrews and Macé, 2013). The implication is that some innovations in the text induced

[1] Insights and historical considerations on the tree as a preferred technical model for stemmatology and concurrent graph theoretical considerations are discussed for instance in Hoenen et al. (2017), Flight (1994) and sources therein.

[2] Note, that we target stemmata for closed traditions (Pasquali and Pieraccioni, 1952) that have no multiple originals, as is probable for orally transmitted epics (Lord, 1960). Hoenen (2017) has attempted to reconcile tree and network perspectives on stemmata.

[3] Vorlage is a German loan, which is used in philology to describe the model or the original of a copy.

[4] The webpage of the challenge is `https://www.cs.helsinki.fi/u/ttonteri/casc/`. Here, the challenge datasets can be obtained. For the current study the authors provided the full datasets additionally via direct contact.

by copying are idiosyncratic and hardly revertable, for instance when some non syntactically crucial word is accidentally left out: *this is a really big challenge → this is a big challenge* or when some content word gets replaced by one equally fitting into the context: *the clay dust shimmered → the day dust shimmered*. Such errors imply,[5] that the whole subbranch rooted by the manuscript having the new version at first will have it. In this way the innovation is genealogically informative. That is the information helps us locate the manuscript on the stemmatic tree, whereas other innovations could easily happen independently in different copy processes such as the introduction of punctuation at some point in time or some shift in definiteness *I heard the magpie → I heard a magpie*. Often variation can be multicausally explained and is analyzed on a case-by-case basis. One process which could be responsible for both kinds of innovations is the confusion of letters. In philological discussions on the complex processes which can lead to variation, the confusion of letters such as <cl> with <d> has been discussed early on, probably already in antiquity (Vanek, 2007, p.276). Reynolds and Wilson (2013, p.222) identify conscious and inadvertent processes as underlying such and other processes.

The current paper tries to determine how well the true stemmatic trees for artificial benchmark datasets in stemmatology (gold standards) can be approximated from external data on the confusion of letters. As Spencer et al. (2004b), we use manually provided alignments, derive pairwise distance matrices and then use the Neighbor-Joining (NJ) algorithm for stemma generation from the distance matrix. In computing the distance matrix of pairwise variant text distances, we compare each position of the alignments and implement three metrics, the simple binary (same or different variant?) Hamming distance (Hamming, 1950), the Levenshtein distance (Levenshtein, 1965) and the weighted Levenshtein distance.[6] For weighting, we do not consider philological classes of variation but distance matrices from psycholinguistic research on letter distances. These have been gained in experimental set-ups and do thus suffer less from a weighting bias introduced through subjectivity as mentioned in Spencer et al. (2004b). Comparing stemma generation with philologically inspired weighting against unweighted stemma generation (Hamming distance), Spencer et al. (2004b) found no crucial differences in the resulting stemmata for their data set but stated (p. 236) that 'different weightings could lead to completely different stemmata' concluding (p.238) that 'Determining appropriate weightings in these cases is an open problem'. The main aim of this paper is to assess a part of this problem through using external data for weighting.

[5]Terminologically, there are some slightly differing terms which imply similar things: variant, innovation, error, change, alteration. Since 'error' implies a knowledge of the correct form, the term can sometimes lead to controversies. Here, we use all terms quasi-interchangeably.

[6]Weights for transpositions are not immediately derivable from psycholinguistic letter confusion matrices. Additionally, there are long distance transpositions or transpositions of vowels of adjacent syllables which would require some additional linguistically carefully modelled distance. The java library debatty `info.debatty.java.stringsimilarity` was used for implementation of the weights.

2. Artificial Data Sets

For evaluation, we use three most used artificial datasets, called Parzival PRZ (English), Notre Besoin NB (French) and Heinrichi HR (Finnish) (Baret et al., 2004; Spencer et al., 2004a; Roos and Heikkilä, 2009) both in their entirety. A fourth[7] and fifth (Hoenen, 2015a)[8] are not focussed.[9] PRZ has 21 manuscripts and the alignment has 855 lines, NB features 13 manuscripts of 1035 lines and HR 64 manuscripts of 1208 lines.[10] From a machine learning perspective, these data sets are quite small and from a historical perspective, they may not represent but a tiny fraction of possible scenarios. Results are thus to be taken with utter caution. Nevertheless, these are the only data in the field for which an indisputable gold standard exists.

3. Method and Model

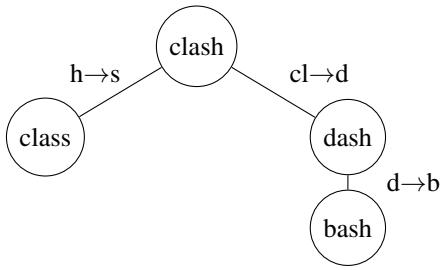
Of all pairwise manuscript comparisons, in large numbers of cases both manuscripts do not share an edge in the true stemma. Hence, those comparisons will include word pairs which stem from remotely distant manuscripts on the stemmatic tree. On each edge on the shortest path between them some event(s) may have happened with the implication that one is most often looking at variation reflecting more than one copy step and a back and forth of directionality. This is unfortunate but unavoidable if one doesn't know the true relations in advance. To illustrate, Figure 2 gives a small toy example of a tradition, where each manuscript contains only one word and where thus the comparison of the concurrent word pairs correspond to manuscript pair comparisons. Looking at Figure 2, when comparing and aligning words, all pairs are different in terms of a binary classification. Using the Levenshtein distance, token pair c gets the same distance as pair b, but only b corresponds to an edge. Counting differing alignment positions assigns the same distance to b and c as well. Only carefully chosen weights achieve an overall weighting that assigns the three lowest values to the pairs corresponding to edges and the largest to that with the longest path in the true stemma. Weights in the example are set intuitively to mimick confusability of the aligned letter units. Realistic confusability patterns of letters and phonemes have been researched

[7]`https://phylomemetic.wordpress.com/2015/02/12/artificial-textual-tradition-julies-caesar/` :last accessed on 04.02.2018

[8]Available under Tascfe at `https://www.texttechnologylab.org/applications/corpora/`.

[9]Both have been used scarcely in the literature, compare Robinson (2015) and the correct stemma available online for the first of them does not feature all node names. The second requires substantial additional modelling since it has a) multiple roots, b) is written in the Arabic writing system of Persian usage and c) because there are considerably less psycholinguistic resources for this constellation. Additionally the text is rather short. Experiments were conducted and results are briefly summarized below.

[10]We include the PRZ_loss challenge data set (17 ms) for comparison.



Index	comp	pos	Lev	MMD*	l(path)	EDGE
a	c-l-a-s-**h** c-l-a-s-**s**	1	1	0.4	1	T
b	**cl**-a-s-h **d** -a-s-h	1	2	0.2	1	T
c	**cl**-a-s-h **b** -a-s-h	1	2	0.8	2	
d	**cl**-a-s-**s** **d** -a-s-**h**	2	3	0.2+0.4 0.6	2	
e	**cl**-a-s-**s** **b** -a-s-**h**	2	3	0.8+0.4 1.2	3	
f	**d**-a-s-h **b** -a-s-h	1	1	0.3	1	T

Figure 2: An example stemma and all corresponding (text=) word pair comparisons. All word pairs are manually aligned, corresponding comparisons (column comp.) highlighted. The number of such comparisons or positions (column pos) is compared to the Levenshtein distance (lev) and a modally weighted version of it (*Multi Modal Distance, MMD, with one addend for each comparison). Path length (l(path)) between the nodes of a pair and whether this corresponds to an edge serve evaluation. Only MMD achieves an optimal ranking.

and can be inferred from psycholinguistic experiments, see next section, which brings external data into the model and which might help to avoid overfitting and subjectivity. Another question is how much these linguistically speaking low-level phenomena are responsible for the variation observed in copies.

3.1. Model

We operate with a number of observed (survived) manuscript variant texts $\mathbb{M}$, which are arranged in a provided token level alignment A. For each variant text pair $(M_i, M_j), i \neq j$, we sum the weighted Levenshtein distances of all words $M_{i,j_{1..k}}$ (implying different letter level alignments) according to the different weighting schemas of the modalities and then weight again each modality with a linear factor.

$$\begin{aligned} &\Delta(M_{i_k}, M_{j_k}) = \\ &\alpha \cdot wLev_{vis}(M_{i_k}, M_{j_k}) + \\ &\beta \cdot wLev_{ac}(M_{i_k}, M_{j_k}) + \\ &\gamma \cdot wLev_{mot}(M_{i_k}, M_{j_k}), i \neq j, \end{aligned} \tag{1}$$

where $wLev_{modality}$ is the weighted Levenshtein distance according to the values from modally determined (**vis**ual, **ac**oustic, **mot**oric) psycholinguistic letter distance matrices, M_{i_k} is the $k-$th token (alignment position, often word) of the $i-$th manuscript and α, β, γ are the respective weights for the modalities. The final distance of a variant text pair is then simply

$$\sum_{k=1}^{length(A)} \Delta(M_{i_k}, M_{j_k}). \tag{2}$$

In the even simpler conditions for comparison, the distance function Δ simply returns 1 (in case of difference) or 0 (in case of identity) of the elements in (M_{i_k}, M_{j_k}) or in the other condition $Lev(M_{i_k}, M_{j_k})$.

3.2. Modalities

Copying is a very complicated process and builds on many cognitive processes, compare Hoenen (2014), amongst others reading, retaining the read in memory and writing are involved. These make use of vision, probably acoustics (as far as retention in memory is involved) and motor innervation of the muscles responsible for the movements leading to writing. Among human modalities or senses, those three are assumed to be the decisive ones for the copy process.
Whilst human languages differ profoundly in a number of parameters, the basic receptory and cognitive apparatus is essentially the same for all humans. Consequently, basic confusability patterns should across time and language be roughly stable. Therefore, we believe one can use psycholinguistically derived confusability information for a weighting regardless of the time period or language from which the textual material may stem.

Vision and Reading In comparison to the other modalities, vision is not only the most important one, but witnesses by far the largest body of research on confusability of letters. In order to model the values of the visual modality, matrices of visual confusability have to be used. Müller and Weidemann (2011) have compared 55 papers from 1886 until 2011 that describe 74 experiments (the majority using psycholinguistic approaches (ca. 82%)) to establish letter discriminability matrices for the Latin alphabet. As many tables as were readily available from the supplied paper links have been extracted and it was ensured that they were labelled for

1. modality (visual, motoric, acoustical)
2. directionality (symmetric matrix?, $\Delta(< a >, < d >) = \Delta(< d >, < a >)$?)
3. letter set (upper case, lower case, numbers, mixed case)
4. polarity (similarity or distance)

However, some matrices or data reported in the papers were not used, since they either analysed irrelevant data (perception in pigeons (Blough, 1985), discrimination of the Braille alphabet, (Gilmore et al., 1979)), reported a poor predictive performance (Coffin, 1978), provided incomplete data (Uttal, 1969), featured very few observations (Banister, 1927) or were hardly extractable due to age or condition of the pdfs. We ended up with 27 matrices.
In order to make all matrices comparable, the values were normalized to a number between 0 and 1 by using the largest value as 1, if and only if the reported values were not already in that range explicitly representing percentages.

The values were transformed if necessary turning similarities into distances. Furthermore, distances were averaged if directional differences existed: $\Delta(a, b) = \Delta(b, a)$. This was primarily done since the direction of copy when comparing two manuscripts is not apriori known. All non observed letter combinations would receive the maximal distance. For numbers Keren and Baggen (1981) provided a table and for mixed case only Boles and Clifford (1989) reported confusability values. This gave 1 matrix for numbers and mixed case visual confusion and 6 matrices for lower case to lower case letters and 17 for upper case to upper case letters. We combined those and obtained and tested 102 combinations of visual uppercase, visual lowercase, visual mixed case, visual number confusabilities with acoustic and motoric confusion matrices. Matrices have been made available on GitHub.[11]

Acoustic Modality For acoustic confusion, the process of modal transition from and into the visual medium must be modelled as an additional step. Naturally, one could choose grapheme-to-phoneme (g2p) and p2g based approaches. However, since the aim of the present study is to analyse explicitly modally motivated errors, we alternatively do the following and leave g2p/p2g as an alternative for future research.

Cutler et al. (2004) provide phoneme-based confusability matrices. We use the ones for initial vowels and consonants discriminated by natives. In word initial position, the phonemes do usually not become subject of heavy coarticulation.[12] Additionally, there was a high canonical correlation between initial and final confusability values (vowels initial and final:0.99, consonants in onset and coda: 0.81). For the mapping of phoneme pair distances to graphemic units (GU), Van Berkel (2005)'s *basic*, *contextual* and *word specific* spellings were used for English.[13] For instance, the presumably confusable GU pairs potentially representing /aʊ/ and /aɪ/ constructed from this were: <ou>:<i>, <ou>:<y>, <ou>:<ie>, and <ow>:<i>, <ow>:<y>, and <ow>:<ie>. The same corresponding normalized distance value from the matrix of phoneme distances was assigned to each of them and used with the acoustically weighted Levenshtein distance. If one GU pair could represent multiple phoneme pairs, all of its values were averaged. For Finnish and French similar resources were used to obtain GUs (Lyytinen et al., 2013; Lehtonen, 2013; Wiik, 1965; International-Phonetic-Association, 1999; Guex and Pithon, 1975; Dryer and Haspelmath, 2013; O'Grady et al., 1997).

Those acoustic distances between phonemes for which Cutler et al. (2004) have provided no values have been estimated using the average of the values of all observed pairs, which had a similar distance. This distance was measured in terms of numbers and qualities (backness, height, roundedness) of edges in the vowel diagram or number and quality (place, manner and voice) of steps in the consonant table of the International Phonetic Alphabet (IPA). See Figure 3.2., here, to come from /a/ to /ɜ/ requires a shift in height (to reach ɛ) and then one in backness summing 2 steps. To estimate the value of a pair where at least one phoneme was not observed by Cutler et al. (2004), and where their distance was one shift in height and one in backness, we sum and average all distances for observed vowel pairs from the chart which have that very distance. Analogously, for consonants, for instance /n/ to /m/ has 4 place steps, /n/ to /d/ 4 place and 1 manner steps, /n/ to /t/ an additional voice step etc. For diphthongues with missing values, the distinction was made between such diphthongue pairs which shared at least one sound and such which did not and the concurrent observed averaged values were assigned.

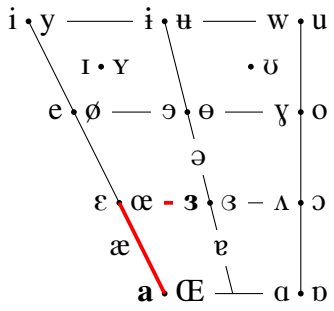


Figure 3: The vowel diagram from the IPA. Highlighted in red and bold is the path from /a/ to /ɜ/.

Motor Modality While biologically muscular neurology is well-understood, research focussing on letter production from a motor-perspective is comparatively rare. In Müller and Weidemann (2011), the only mentioned study focussing on letter production is Miozzo and Bastiani (2002), where production errors of one patient are reported, who suffered a brain damaging intoxication. They found letter substitutions to occur predominantly between letters with common strokes such as <b> and <p> and remarked that letter frequency, consonant-vowel status and letter gemination were affecting such errors. Their data was used to define all motoric weights and truly corresponds to handwriting. Non observed pairs received the maximum distance.[14]

4. Stemmatological Application of MMD

We used the artificial datasets to compute a pairwise distance matrix with the MMD (ignoring gaps in the alignment) using each combination of matrices (102 combinations). From the pairwise distances, we computed a stemma using NJ from the R package *ape* and then scored it using the so called Average Sign Distance (ASD),[15] an accuracy

[11] https://github.com/HoenenA/MultiModalDistance/. References to all in Müller and Weidemann (2011).

[12] Coarticulation is a linguistic phenomenon whereby some phonemes are influenced by previous or subsequent ones.

[13] Van Berkel (2005) analyses the English spelling system postulating for each phoneme a *basic spelling* which reflects the most frequent spelling for this phoneme, a *contextual spelling* representing a frequent but not the most frequent spelling and *word specific spellings*. Corresponding phonemes have been mapped from American to British English in the process.

[14] Values on $< a/e >$ were not used.

[15] While on the level of path comparison operating on distance, in terms of the overall manuscript comparison, the ASD is rather a similarity and referred to as Average Sign Similarity by other authors.

Trad.	Bin	Lev	MMD	RH09
NB	69.35	58.74	69.35	**77**
PRZ	72.11	67.58	72.11	
PRZ_loss	76.04	71.28	76.04	**87**
HR	72.71	72.9	74.32	

Table 1: Comparison of stemmatological evaluation results with ASD on the percentage of shared words (Bin), the Levenshtein distance (Lev) and the MMD (psycholinguistically weighted Levenshtein distance). RH09 gives the best achieved results of the 2009 study of Roos and Heikkilä.

Trad.	vis	ac	mot
NB	$-0.18(58)$	$\mathbf{0.56}(66)$	$-0.39(45)$
PRZ	$-0.43(69)$	$\mathbf{0.83}(72)$	$-0.4(68)$
PRZ_loss	$-0.6(67)$	$\mathbf{0.51}(76)$	$0.03(71)$
HR	$-0.12(73)$	$\mathbf{0.29}(74)$	$-0.17(73)$

Table 2: Pearson correlations between ASD values and modal parameters, strongest per row highlighted. In brackets, ASD value when modality was used exclusively (weighting factor set to 1 and all other weighting factors to 0, average over combinations of letter case confusion matrices).

value introduced by Roos and Heikkilä (2009) as:

$$u(A,B,C) = 1 - \frac{1}{2}|sign(d(A,B) - d(A,C)) - sign(d'(A,B) - d'(A,C))|$$

A, B and C are nodes present in both the true and the estimated stemma, $d(A,B)$ is the distance of the two nodes in the true stemma defined as the number of edges on the shortest path between them, $d'(A,B)$ the same distance for the estimated tree. $sign(d(A,B) - d(A,C))$ returns so to speak only the sign, discarding length, thus -1 if $d(A,B) < d(A,C)$, 1 in the opposite case and 0 if both are equal. The index equals 1 if both stemmata agree and 0 if they differ ($\frac{1}{2}$ if they partly agree, for details see the formula or (Roos and Heikkilä, 2009)) and is computed and turned into a proportion for all such triples. ASD is the to date most used evaluation metric for stemmata on the artificial data sets used for instance in (Lai and O'Sullivan, 2010; Roos and Zou, 2011; Hoenen, 2015a).[16]

4.1. Experiment and Results

For each of the artificial traditions, we tested 102 combinations of uppercase and lowercase confusion matrices and for each such combination, we tested 66 different parameter settings, including such where the weight for any one parameter was 1.0. In all, these were $6,732$ combinations per tradition, thus roughly $27,000$ results. Since this is far too much to be displayed in a simple table, we give results in several different ways. Table 1 contains the *best achieved results* of the MMD for each tradition (including the further not focussed loss scenario). Ranges between the best and worst results in the whole grid of $6,732$ configurations were roughly 24% ASD for NB with the worst result $45,5$ for PRZ (worst:67) and 26 for HR (worst:48).

As what regards the combinations of uppercase and lowercase matrices, it must be said that those transitions truly involving only uppercase letters were rare and those involving mixed case still very infrequent (in NB roughly 10% and for PRZ roughly 14%) in respect to lowercase to lowercase transitions. All matrix combinations (1 uppercase, 1 lowercase confusion matrix) witnessed parameter settings for which the respective best results were produced. Averages per matrix (over all parameter settings) produced roughly similar results and no matrix combination was an extreme outlier.

[16] A python and a C++ implementation are available from (Roos and Heikkilä, 2009) through the stemmatology challenge website.

As for the values of the modalities, we looked at the weighted average contributions of the parameters, that is for each modality: $\sum_{i=1}^{6732} \omega * ASD[i]$, where all 6,732 ASD values are in one array and each position of the array is conditioned by four parameters: the matrix combination, the weight for vision, acoustics and motorics and where ω is the corresponding weight for the modality under investigation.

Values were almost identical for the modalities and a desired significant difference was not visible. Looking only at those results, where one of the parameters had been set to 1, only for NB some significant pattern emerged: vision (contributing 39%) worked slightly better than audition (contributing 34%) while motorics performed a little worse (contributing 27%) and deviated from the mean significantly (t-test, significance level 0.01). Getting a deeper insight, while this was not possible for the matrix combinations due to the categorical character of these data points, for the weighting factor values additionally a Pearson correlation analysis with the ASD array could be conducted, which yielded more interesting results, see Table 2.

There is a strong positive correlation of the acoustic modality with the result for PRZ and NB and a weak one for HR. It must be said however that these correlations are to be understood as on the conjunction of parameter settings. This means that a larger negative value does not automatically mean that there is a bad effect of this modality but solely that in conjunction with at least one better performing modality, the contribution to the overall result was moderate. In other words, the reason why ASD values suffer if the negatively correlated modality gets stronger may be the result of the more effective ones getting weaker not necessarily because of a bad fit of the modality itself. This is corroborated by the values where the single modalities were used exclusively and by the fact that the best overall results were often only reached in settings where the modalities had been combined.

We conducted all the above analyses in the same way also for the so called TASCFE corpus (Hoenen, 2015a) which has some special characteristics such as being based on 4 different initial versions (entailing multiple roots or 4 clusters) and can be used as a testset to robustness of a stemma generating algorithm. Secondly and most importantly, TASCFE is written in Persian making all letter distance matrices from Müller and Weidemann (2011) unuseful. Not least because of the small length (the alignment

features only 137 positions) this data set is the most challenging and produced not unexpectedly the worst results. Best ASDs were roughly between 56 and 63 (still far from chance) on the 4 complete single subsets. For the MMD, visual confusion resulted in only 1 matrix which had been modelled based on the similarity of letter features in Wiley et al. (2016), motoric similarity came from the same source but used the reported stroke similarity between the letters. Acoustic to graphemic mappings were deducted from the International Phonetic Alphabet (International-Phonetic-Association, 1999).

5. Discussion

Results are generally negative in that they did not outperform the best of some previously reported values although they are in the range of many of the there presented approaches.[17] However, to a certain extent this was expectable, given that only a subset of the innovations found which occurred along the textual transmission are estimated to be the direct result of simple modal or multi-modal confusion on the token level. Consequently, we first looked at the data in more detail to find out how many of the deviations were possibly such captured by the MMD. We conducted a tentative and surely partly subjective classification to this end. In order to better be able to interpret this data, the analysis was confined to NB (French) and PRZ (English). Some of the confusions occurred many times in different copy processes not always of the same source manuscript. Some of them, in the French case, are presumably reverts of a dictation copy where a non-native speaker had misrecorded some silent endings. A large number of cases involved deletions or insertions of letters, which possibly in part explain why the results of the binary distance and the MMD are not differing for NB and PRZ. Those are not subject to the MMD weighting schema but will make token distance coincide with the Hamming value. Another reason can be that some distances in the matrices (for some matrices, the majority of distances) were so small that they could hardly make a big difference as compared to the Hamming distance.

Overall, we found roughly a third of the differences to be applicable to a visual (127 of 422) or acoustic (115) weighting. Motoric confusion was deemed possible for roughly a sixth (56) of the cases. In conjunction roughly half (206) of the variation was subject to MMD weighting.

As for the matrices, generally their performance was not extremely different with some interesting observations. Müller and Weidemann (2011) comparing 11 of the matrices found a mean correlation of 0.68 to the generated average matrix (p.30), which well aligns with our observation. The matrices were all qualitatively roughly similar but some would have a large range between smallest and largest values, some would give differing values for self similarity. In cases with no self similarity reported, we had assigned unity regardless of the magnitude of differences with and in the rest of the matrix. Overall, the matrix of (Geyer, 1977) performed best by a very small margin. (Courrieu and De Falco, 1989)'s matrix although on average best performer for English clearly performed worst for French. The data had been gained from the confusions of preschoolers and showed a presumably acoustically decoupled confusability component (<p> with <q>). Such relations might have influenced the distance matrix in a way as to overwrite some genealogical relations for French so that NJ, which is a greedy algorithm found some tree quite different from the others. Generally, the information from the differences which are not measurable by MMD may additionally crucially determine the schema of information reduction from distance matrix to stemma and thus obscure the fit.

Comparing the different metrics, interestingly, the Levenshtein distance was clearly outperformed for NB and PRZ by the binary distance despite Levenshteins ability to measure the degree of difference between two tokens. For HR however, this was not the case. Furthermore, for HR MMD was outperforming the binary distance. This result may be due to the writing systems of the languages involved. More specifically, Katz and Frost (1992) introduce the notion of *orthographic depth*. English and French in this sense are deep orthographies, that is their g2p and p2g relations contain many n:m relationships, whereas Finnish is a shallow system (Joshi and Aaron, 2013). Illustrating the difference between a deep (English) and a shallow (Finnish) orthography, it may suffice to look at the following two examples:
P2G: $/k/ \rightarrow \{<c>, <k>, <ck>\}^{EN}, \{<k>\}^{FIN}$
G2P: <a>$\rightarrow \{/ɑ/, /ɑ\text{:}/, /ɒ/, /æ/, /eɪ/\}^{EN}, \{/ɑ/\}^{FIN}$
The values from our confusion matrices in the MMD cover 1:1 letter confusion values. If now confusion took place also on some levels of graphemic units, these would not be captured by the visual and motoric confusion values, albeit by the acoustical ones. For instance the confusion of <their> and <there> could entail such a larger-unit-based confusion, where not one letter is cofused with one other letter. Acoustic distances as modelled however take into account such units since there is the above-described mapping between phonemes and graphemic units. In fact, the positive correlations of the acoustic weighting factors seem to support such an interpretation. Moreover, since Levenshtein may assign too large a value to confusions which involve n:m relations that correspond to just one confusion it may introduce noise, especially for deeper orthographies, so much so that its overall result becomes worse than the binary distance.[18] For the same reason, MMD is distinguishable from the binary distance for Finnish. In this vein, results all seem to be most consistent with an interpretation which suggests that the proposed method currently works best for texts written in languages with a shallow writing system (e.g. Latin). Confusion matrices for more com-

[17] Note, that our results on the binary distance combined with NJ achieved slightly worse results than those obtained in the challenge using the same data, metric and algorithm, for NB (theirs:76.2, ours:69.35), PRZ (theirs:81.5, ours:76.04) . This may be due to the fact that NJ is a greedy algorithm and different implementations and/or manuscript text orderings may output different trees.

[18] A similar explanation may hold for the observation of Spencer et al. (2004b) who found that subjective weights had made few difference. Here, weights might have accidentally obscured the genealogical information although the weighting, quite like Levenshtein may not have been unreasonable in itself.

plex orthogaphic units could be useful.[19] It also suggests that the level of graphemic units could be quite important in analysing confusion (on token level). This interpretation would be consistent also with the Persian data, where taking into account abstract letter identities[20] produced some better values than the graphemic distances.

However, utter caution must be taken since the data sets are by all means small and not representative of historical data as such. Their size entails a grave danger of overfitting, which is why using methods of machine learning to optimize the weights may be dangerous and surely much more effective on larger data sets. Additionally, our model of an interplay of the modalities is not the only possibility and ideally each position of a manuscript would require some different weighting input or an entirely different model (for instance if not modal confusion on the token level but contextual priming effects paired with some degree of visual similarity cause miscopying (Hoenen, 2015b)). There are confusions, where one single modality is to be held responsible. When Spencer et al. (2004a) mention the exapmle of <cl> and <d>, it is unlikely that the reason for the confusion lie in any other modality than vision. Thus modelling each modality separately and summing them, apart from having neurological correlates, is not unreasonable but the presented approach is surely just a first step to investigate a complex and data sparse object.

6. Conclusion

We presented an approach to weighted stemma generation from pairwise manuscript text distance matrices. In the approach, external data in the form of psycholinguistically generated letter and phoneme distance matrices in the visual, acoustic and motoric modalities was used to model weights for a weighted version of the Levenshtein distance. We tested and evaluated the approach producing stemmata from manuscript pair distances of three artificial data sets with known ground truth. Results were not outperforming the best results reported in Roos and Heikkilä (2009), but in all cases were better than many other approaches. Which external input matrix to choose was found not to be crucial in our setting and all combinations of matrices performed very similarly. Regarding the contribution of the single modalities, acoustics as modelled performed very well, but best results were often only achieved when the modalities were combined in a weighting schema. We additionally found that most likely orthographic depth was the reason why MMD outperformed the binary distance only for Finnish and why the unweighted Levenshtein distance was outperformed by the binary distance for the French tradition NB and the English tradition PRZ. The main contribution of the paper is thus in corroborating an argument in the discourse. That argument is that weighting beyond the word level may make sense, but weights must be carefully elicited and theoretically grounded for instance using psycholinguistically derived confusability matrices. Approaches to weighting which are not confined to the comparison of manuscript and token pairs, but which take into account additional distributional information of each variant, such as the one presented by Roelli and Bachmann (2010) could improve results of weighted approaches in another vein, which a quantitative assessment for instance against the benchmark datasets could reveal. We conclude with a word of caution, that all results have been obtained on relatively small data sets.

[19] To this end, some experiments with OCR error data on n:m confusions showed positive effects.

[20] An abstract letter identity is a cognitive entity which connects different elements of a writing system behaving in the same way, for instance the lowercase 'incarnation' and the uppercase 'incarnation' of a letter $\{a, A\}$.

7. Acknowledgements

The author would like to thank Prof. Dr. Alexander Mehler, who proposed the design of formula (1) and made many other helpful suggestions and provided infrastructure.

8. Bibliographical References

Andrews, T. L. and Macé, C. (2013). Beyond the tree of texts: Building an empirical model of scribal variation through graph analysis of texts and stemmas. *Literary and Linguistic Computing*, 28(4):504–521.

Banister, H. (1927). Block capital letters as tests of visual acuity. *The British journal of ophthalmology*, 11(2):49.

Baret, P., Macé, C., and P. Robinson, e. (2004). Testing methods on an artificially created textual tradition. In *Linguistica Computationale XXIV-XXV*, volume XXIV-XXV, pages 255–281, Pisa-Roma. Instituti Editoriali e Poligrafici Internationali.

Blough, D. (1985). Discrimination of letters and random dot patterns by pigeons and humans. *Journal of Experimental Psychology: Animal Behavior Processes*, 11(2):261–280.

Boles, D. B. and Clifford, J. E. (1989). An upper- and lower case alphabetic similarity matrix, with derived generation similarity values. *Behavior Research Methods,Instruments, & Computers*, 21:597–586.

Coffin, S. (1978). Spatial frequency analysis of block letters does not predict experimental confusions. *Perception & Psychophysics*, 23(1):69–74.

Courrieu, P. and De Falco, S. (1989). Segmental vs. dynamic analysis of letter shape by preschool children. *Cahiers de psychologie cognitive*, 9(2):189–198.

Cutler, A., W., A., Smits, R., and Cooper, N. (2004). Patterns of English phoneme confusions by native and non-native listeners. *Journal of the Acoustical Society of America*, 116:3668 —-3678.

Matthew S. Dryer et al., editors. (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Ellison, J. W. (1957). *The use of electronic computers in the study of the Greek New Testament text*. Harvard University.

Flight, C. (1994). A complete theoretical framework for stemmatic analysis. *Manuscripta*, 38(2):95–115.

Geyer, L. (1977). Recognition and confusion of the lowercase alphabet. *Perception & Psychophysics*, 22(5):487–490.

Gilmore, G., Hersh, H., Caramazza, A., and Griffin, J. (1979). Multidimensional letter similarity derived

from recognition errors. *Perception & Psychophysics*, 25(5):425–431.

Guex, A. and Pithon, M. (1975). *Manuel de phonétique française*. Ecole de français moderne de l'Université, Lausanne.

Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell Labs Technical Journal*, 29(2):147–160.

Hoenen, A., Eger, S., and Gehrke, R. (2017). How many stemmata with root degree k? In *Proceedings of the 15th Meeting on the Mathematics of Language*, pages 11–21.

Hoenen, A. (2014). Simulation of scribal letter substitution. In T. L Andrews et al., editors, *lectio 1*, pages 119–139. Brepols, Turnhout.

Hoenen, A. (2015a). Das artifizielle Manuskriptkorpus TASCFE. In *DHd 2015 - Von Daten zu Erkenntnissen - Book of abstracts*. DHd.

Hoenen, A. (2015b). Simulating misreading. In *Proceedings of the 20th International Conference on Applications of Natural Language to Information Systems (NLDB)*.

Hoenen, A. (2017). Beyond the tree - a theoretical model of contamination and a software to generate multilingual stemmata. In *Book of Abstracts*, pages 155–159. AIUCD.

International-Phonetic-Association. (1999). *Handbook of the International Phonetic Association*. Cambridge University Press.

Joshi, R. M. and Aaron, P. (2013). *Handbook of Orthography and Literacy*. Routledge.

Katz, L. and Frost, R. (1992). The reading process is different for different orthographies: The orthographic depth hypothesis. *Haskins Laboratories Status Report on Speech Research*, SR-111:147–160.

Keren, G. and Baggen, S. (1981). Recognition models of alphanumeric characters. *Perception & Psychophysics*, pages 234–246.

Lai, P.-H. and O'Sullivan, J. A. (2010). Mdl hierarchical clustering with incomplete data. In *Information Theory and Applications Workshop (ITA), 2010*, pages 1–5. IEEE.

Lehtonen, A., (2013). *Handbook of Orthography and Literacy*, chapter Sources of Information Children Use in Learning to Spell: The Case of Finnish Geminates, pages 63–80. Routlegde.

Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848.

Lord, A. B. (1960). *The Singer of Tales*. Harvard University Press.

Lyytinen, H., Aro, M., Holopainen, L., Leiwo, M., Lyytinen, P., and Tolvanen, A., (2013). *Handbook of Orthography and Literacy*, chapter 4, pages 47–62. Routlegde.

Miozzo, M. and Bastiani, P. D. (2002). The organization of letter-form representations in written spelling: Evidence from acquired dysgraphia. *Brain and Language*, 80(3):366 – 392.

Mooney, L. R., Barbrook, A. C., Howe, C. J., and Spencer, M. (2003). Stemmatic analysis of lydgate's "kings of england" : a test case for the application of software developed for evolutionary biology to manuscript stemmatics. *Revue d'histoire des textes*, 31(2001):275–297.

Müller, S. and Weidemann, C. (2011). Alphabetic letter identification: Effects of perceivability, similarity, and bias. *Acta Psychologica*.

O'Grady, W., Dobrovolsky, M., and Katamba, F. (1997). *Contemporary Linguistics*. Longman, St. Martin's.

O'Hara, R. J. (1996). Trees of history in systematics and philology. *Memorie della Società Italiana di Scienze Naturali e del Museo Civico di Storia Naturale di Milano*, 27(1):81–88.

Pasquali, G. and Pieraccioni, D. (1952). *Storia della tradizione e critica del testo*. Le Monnier.

Reynolds, L. and Wilson, N. (2013). *Scribes and Scholars, A Guide to the Transmission of Greek & Roman literatures*. Oxford University Press.

Robinson, P. (2015). Four rules for the application of phylogenetics in the analysis of textual traditions. *Digital Scholarship in the Humanities*.

Roelli, P. and Bachmann, D. (2010). Towards Generating a Stemma of Complicated Manuscript Traditions: Petrus Alfonsi's Dialogus. *Revue d'histoire des textes*, 5(4):307–321.

Roelli, P. and Macé, C. (2015). Parvum lexicon stemmatologicum. a brief lexicon of stemmatology.

Roos, T. and Heikkilä, T. (2009). Evaluating methods for computer-assisted stemmatology using artificial benchmark data sets. *Literary and Linguistic Computing*, 24:417–433.

Roos, T. and Zou, Y. (2011). Analysis of textual variation by latent tree structures. In *11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14, 2011*, pages 567–576.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.

Spencer, M., Davidson, E. A., Barbrook, A. C., and Howe, C. J. (2004a). Phylogenetics of artificial manuscripts. *Journal of Theoretical Biology*, 227:503–511.

Spencer, M., Mooney, L., Barbrook, A., Bordalejo, B., Howe, C., and Robinson, P. (2004b). The effects of weighting kinds of variants. In P. van Reenen, et al., editors, *Studies in Stemmatology II*, pages 227–240. John Benjamins.

Uttal, W. R. (1969). Masking of alphabetic character recognition by dynamic visual noise (dvn). *Perception & Psychophysics*, 6(2):121–128.

Van Berkel, A., (2005). *Second Language Writing Systems*, chapter The Role of Phonological Strategy in Learning to Spell, pages 97–121. Multilingual Matters.

Vanek, K. (2007). *Ars corrigendi in der frühen Neuzeit: Studien zur Geschichte der Textkritik*, volume 4. Walter de Gruyter.

Wiik, K. (1965). *Finnish and English vowels*. Turun Yliopisto.

Wiley, R. W., Wilson, C., and Rapp, B. (2016). The effects of alphabet and expertise on letter perception. *Journal of Experimental Psychology: Human Perception and Performance*, 42(8):1186–1203.

A Corpus of Natural Multimodal Spatial Scene Descriptions

Ting Han, David Schlangen
Dialogue Systems Group // CITEC // Faculty of Linguistics and Literary Studies
Bielefeld University
firstname.lastname@uni-bielefeld.de

Abstract

We present a corpus of multimodal spatial descriptions, as commonly occurring in route giving tasks. Participants provided natural spatial scene descriptions with speech and abstract deictic/iconic hand gestures. The scenes were composed of simple geometric objects. While the language denotes object shape and visual properties (e.g., colour), the abstract deictic gestures "placed" objects in gesture space to denote spatial relations of objects. Only together with speech do these gestures receive defined meanings. Hence, the presented corpus goes beyond previous work on gestures in multimodal interfaces that either focusses on gestures with predefined meanings (multimodal commands) or provides hand motion data without accompanying speech. At the same time, the setting is more constrained than full human/human interaction, making the resulting data more amenable to computational analysis and more directly useable for learning natural computer interfaces. Our preliminary analysis results show that co-verbal deictic gestures in the corpus reflect spatial configurations of objects, and there are variations of gesture space and verbal descriptions. The provided verbal descriptions and hand motion data will enable modelling the interpretations of natural multimodal descriptions with machine learning methods, as well as other tasks such as generating natural multimodal spatial descriptions.

Keywords: Multimodal spatial descriptions, natural language, co-verbal gesture, abstract deictics

1. Introduction

When describing routes that are not visible in the situated environment, humans often accompany verbal descriptions with gestures to demonstrate relative spatial relations of landmarks or trajectories of the routes to follow (Emmorey et al., 2000; Alibali, 2005; Cassell et al., 2007). For example, when trying to help a person to locate a hotel not in current view, a description might be:

(1) You take the tram and get off at the stop "Schumacher street". Now here$_{[deictic]}$ is the tram station, here$_{[deictic]}$ is a fountain, if you walk$_{[iconic]}$ around it, you will see the hotel$_{[deictic]}$ on your left.

while the verbal description specifies the **landmarks** and **actions** (i.e., *tram station*, *walk around*), the deictic gestures encode the spatial layout of the landmarks with position information; the iconic gesture visualises the trajectory of the route. Only when combining speech and gestures together, it's possible to form a complete interpretation of the description, making it a challenging task even for human listeners (Schneider and Taylor, 1999).

In this paper, we present a corpus of multimodal spatial descriptions where hand gestures and speech are jointly used to describe spatial scenes. The corpus includes data collected from two experiments, a scene description experiment and a spatial description experiment. In the two experiments, participants received different instructions to perform the task and got different feedback signals when performing the task (see Section 3.1. and Section 4.1. for details). The former experiment focused on eliciting intuitive multimodal descriptions, while the latter experiment aimed to elicit spatial descriptions with human-computer interaction oriented instructions and constrained gesture space.

In the **scene description experiment**, we aimed to collect intuitive multimodal descriptions. Participants were given a spatial scene description task without instructions on how to perform the task. That is, they described intuitively, either only using speech or using both speech and gestures. The results show that participants often intuitively use gestures in such spatial descriptions and the deictic gestures reflect spatial layouts of landmarks. However, the varied gesture spaces and relatively limited tracking space of existing devices often make it difficult to track hand motion. Hence, we designed the **spatial description experiment** with a somewhat more constrained setup.

In the spatial description experiment, participants were told that they were describing to a computer program (WOz setting; Kelley (1983)). They were suggested to use gestures and restrain their hand gestures in the effective tracking area, so that the computer can "see" the gestures and understand the descriptions better. This setup resulted in a dataset with sufficient hand motion data, while none of the participants reported unnatural gestures due to the limited gesture space.

We have made the following resources of the corpus publicly available: scene information which were used to elicit the descriptions, the transcriptions of speech, recorded hand motion data, and annotations of deictic gestures and speech. (Han et al., 2018) modelled real-time understanding of spatial descriptions using the data of the spatial description experiment. The results show that incorporating hand gestures not only leads to more accurate interpretation of such descriptions, but also leads to earlier final correct interpretations.

2. Related work

Easily available video and audio recording devices have facilitated conversational/discourse level analysis of speech-gesture communications (Lücking et al., 2010; Quek et al., 2002; Schiel et al., 2002). Although these corpora provide natural multimodal communications and detailed annotations, it's a difficult research problem in itself to extract

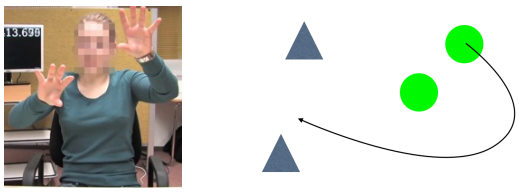


Figure 1: Providing a scene (*right*) description with speech and gestures (*left*). The arrow indicates the movement of the green ball.

3-D gesture features from these videos. Motion tracking sensors that recently have become readily available as well (e.g., Kinect[1] and Leap sensor[2]) make it possible to record large scale 3-D gesture datasets, such as (Tompson et al., 2014; Marin et al., 2014; Liu and Shao, 2013; Sadeghipour and Morency, 2011) and datasets mentioned in (Cheng et al., 2016); however, most of these existing datasets are collected for gesture classification tasks without accompanied speech. (Fotinea et al., 2016) presented a dataset of multimodal commands, where gestures and accompanied are both recorded. However, the gestures are with defined meanings that are independent of speech. In addition to previous datasets, we present a corpus composed of natural multimodal communications with high-resolution hand motion data, in which the meaning of gestures depends on accompanied speech.

3. The Scene Description Experiment

In this experiment, we aimed to collect intuitive scene descriptions. Participants were shown simple scenes (as shown in Figure 1) briefly and asked to describe the scenes from memory. There were no instructions on how to perform the task, hence participants described intuitively, either describing with speech (*mono-modal*) or with speech and gestures (*multi-modal*).

3.1. Task design

We designed a simple scene description task to elicit natural scene descriptions. Participants were asked to describe scenes composed of four simple objects and an arrow which indicates the movement of the object (as shown in Figure 1), intended to trigger deictic and iconic gestures.
We generated 50 such scenes. In each scene, the four objects are with two colours and two shapes. Object *colour*, *shape*, *size* and *position* were randomly selected when the scenes were generated. The arrows originate from one of the objects and point to somewhere near another object. To accurately describe the movement and the spatial configurations after the movement, participants will need to demonstrate the spatial layout with gestures as spatial configurations are difficult to convey with natural language.
To investigate the natural behaviours of such descriptions, participants were only asked to describe the spatial configurations of the objects and the movement indicated by the arrow. Gestures were not mentioned in the instruction.

To elicit accurate descriptions, participants were told that another person will watch the descriptions later and try to recreate the scenes. Describing accurately will make the re-creation task easier for the other person. For each description, the scenes were briefly (10 seconds) shown on a computer screen. After the scene disappeared, participants started to describe.
Each participant described for 20 minutes. In total, 15 participants (native German speakers; students from Bielefeld University) took part in the experiment.

3.2. Recording setup

We recorded audio and video with a HD camera. The hand motion was tracked with a Leap sensor, a portable device composed of two monochromatic cameras and three LED infrared sensors. The hand motion data was recorded with MINT Tools (Kousidis et al., 2013). Both videos and hand motion data were recorded with timestamps.
In the experiment, participants were seated in front of a table. Right across the table and in front of the participant is a HD camera to record audio and videos. A Leap sensor was placed on the table in front of the participant.
On the right side of the table is a monitor which displays the scenes. An experimenter was seated next to the participant to display the scenes. For each scene description, the experimenter clicked a button to show the scene for 10 seconds, then turned the screen to black. After that, participants started to describe. When the description ended, the experimenter advanced to the next scene.
The Leap sensor tracks hand movements and outputs data frames to represent hand motions as following:[3]

- **FrameID**: integer, a unique ID assigned to this data frame.
- **hand number**: integer, the number of tracked hands.
- **hand confidence**: float, ranging from 0 to 1. It indicates how well the internal hand model fits the observed data.
- **hand direction**: 3-D vector. The direction from the palm position toward the fingers.
- **hand sphere centre**: 3-D vector. The centre of a sphere fit to the curvature of this hand.
- **sphere radius**: float, the radius of a sphere fit to the curvature of this hand.
- **palm width**: float, the average width of the hand (not including fingers or thumb).
- **palm position**: 3-D vector, the centre position of the palm in millimetres from the Leap Motion Controller origin.
- **palm direction**: 3-D vector. The direction from the palm position toward the fingers.

[1] https://developer.microsoft.com/en-us/windows/kinect

[2] http://www.leapmotion.com

[3] For detailed descriptions of these features, please refer to the official SDK manualhttps://developer.leapmotion.com/documentation/python/index.html

- **palm velocity**: 3-D vector, the rate of change of the palm position in millimetres/second.
- **finger length**: float, the apparent length of a finger.
- **finger width**: float, the average width of a finger.
- **joint direction**: 3-D vector, the current pointing direction vector.
- **pinch strength**: float, the strength of a pinch pose between the thumb and the closest finger tip as a value in the range $[0, 1]$.
- **grab strength**: float, the strength of a grab hand pose as a value in the range $[0, 1]$. 0 when the hand is open. As a hand closes into a fist, the grab strength increases to 1.
- **finger type**: integer, the integer code representing the finger name. 0 for thumb, 1 for index, 2 for middle, 3 for ring, 4 for pinky.

3.3. Data processing

A sample description is shown as follows:

(2) a) Hier$_{[deixis]}$ ist ein graues Dreieck und hier$_{[deixis]}$ ist ein grüner Kreis hier$_{[deixis]}$ ist noch ein grüner Kreis und hier$_{[deixis]}$ ist noch ein graues Dreieck und von$_{[iconic_start]}$ dem oberen grünen Kreis geht rechts neben dem anderen grünen Kreis$_{[iconic_end]}$ zwischen den beiden Dreiecken nach links ein Pfeil.

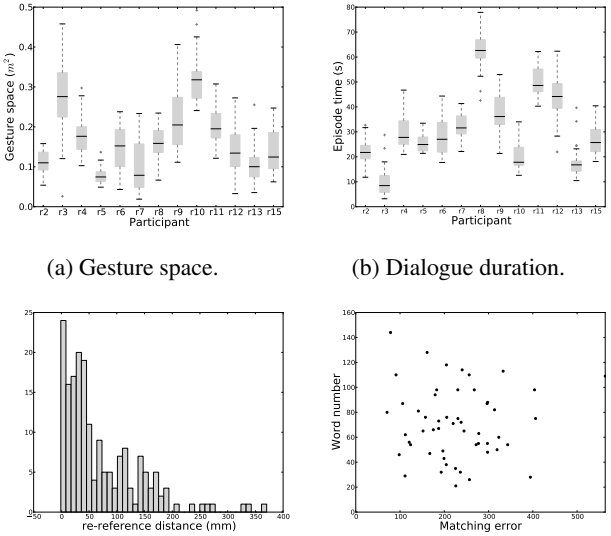
b) *Here$_{[deixis]}$ is a grey triangle and here$_{[deixis]}$ is a green circle here$_{[deixis]}$ is another green circle and here is another grey triangle and from$_{[iconic_start]}$ the upper green circle goes right next to the another green circle$_{[iconic_end]}$ between the two triangles to the left, the arrow.*

Transcription The audio was manually transcribed by native speakers. The transcriptions were temporally aligned with the audio and video recordings on the word-by-word level using an automatic forced alignment approach. We annotated each scene description with corresponding **scene ID** by watching the recordings in ELAN.[4] Each scene description was segmented into individual object descriptions which were annotated with corresponding **object ID**. For instance, the scene description in Example (2) was annotated as *Scene 15*, while the object description "*here$_{[deixis]}$ is a grey triangle*" was annotated as *object 1*.

Gesture annotation Conventionally, each deictic gesture is divided into several gesture phases: pre-stroke, stroke, stroke hold and retraction (Kendon, 1980). During the stroke hold phase, hands stay in the gesture space to indicate object positions, hence, it's the most informative phase. We manually annotated the stroke hold phase of each deictic gesture. The annotation was done by watching the video recordings and the described scenes using ELAN. Similar to natural language annotations, we labeled the *stroke hold* phases with the *object ID* of referential objects.

With the recorded timestamps, hand motion data was aligned with video recordings. Accordingly, the hand motion frames were labeled as *stroke hold* frames or *non-stroke hold* frames according to the timestamps.

As aforementioned, iconic gestures were also involved in the descriptions. For instance, in (2), while describing the movement of the grey triangle with utterance "from the upper green circle goes right next to another green circle", the participant drew a line in the gesture space to indicate the trajectory of the movement. We annotated the start and end iconic gestures with $[iconic_start]$ and $[iconic_end]$.



(a) Gesture space. (b) Dialogue duration.

(c) Histogram of re-referential distance. (d) Word number and description accuracy.

Figure 2: Preliminary analysis results.

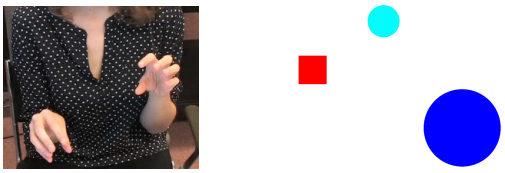
3.4. Preliminary analysis

Varied gesture spaces We calculated the maximal area that each participant's hands spanned during all their descriptions as their gesture space. As shown in Figure 2a, there are variations both within and between subjects in terms of the size of the gesture space which make hand motion tracking and gesture interpretation challenging tasks.

Referential accuracy We also analysed the re-reference accuracy. Figure 2c shows statistics of the reference distance between a deictic gesture and its original gesture. Among 185 re-reference points, 161 of them are with re-reference distance < 150 mm, while the maximum gesture space is $900 \times 671\, mm^2$.

Gesture accuracy of spatial configuration We used a shape matching method to compute the distance between the configurations of gestures and corresponding object positions. The distance was used as a measure of gesture accuracy and compared to the words spoken in each episode, as shown in Figure 2d. The result suggests that when people gesture less accurately, they tend to need more verbal effort to describe the scenes. We did linear regression to analyse the relationship between the number of words spoken in

[4] https://tla.mpi.nl/tools/tla-tools/elan



(a) (b)

Figure 3: Providing a spatial description with speech and gestures: *here is a red square, here is a light blue circle ...*

each episode and the corresponding gesture accuracy. The correlation coefficient is 0.523. It suggests that when people gesture less accurately, they tend to need more verbal effort to describe the scenes.
While the annotation work of iconic gestures is still going on, we leave it as future work to analyse statistics of iconic gestures.

4. The Spatial Description Experiment

In this experiment, we focused on collecting multimodal spatial descriptions with tracked hand motion data which enables the modelling of such multimodal behaviours with machine learning methods. Therefore, we simplified the description task by removing the iconic element (i.e., the arrow) from the scenes and suggested participants to describe with speech and gestures.

4.1. Task design

To elicit multimodal spatial descriptions, we simplified the description task. Instead of describing configurations of four objects and a movement, participants were asked to describe scenes only composed of two circles and one square, as shown in Figure 3. To further reduce the cognitive load of scene memorising, we displayed the scenes on the screen throughout descriptions.
We generated 100 such scenes. The colour and shape of each object were randomly selected when the scenes were generated. There were 6 colours and 2 shapes (*square, circle*). Each of them had the same chance to be assigned to an object. The size and position of each object was randomly generated. The object size ranges from 0.05 to 0.5 in ratio to the size of the scene image. The object positions were adjusted until none of the objects overlap with each other.
Participants were told that they will describe scenes to a *computer program*. The computer will try to understand the descriptions by listening to the verbal descriptions and watching their hand gestures. After each description, the computer displays a score on the screen which ranges from 1 (worst) to 5 (best) and indicates how well the computer understands the description. In reality, the score was from the experimenter who rates the descriptions according to the number of mentioned object attributes.

4.2. Recording setup

The technical recording setup is similar to previous experiment, except that the **hand types** (left or right) were also recorded with a new Leap SDK (SDK v2.3.1). We also placed a monitor in front of the participant to display their hand motions, encouraging them to gesture in the effective tracking area.
At the beginning of the experiment, the experimenter first introduced the task and all the recording devices to the participants, then demonstrated a description with speech and gestures. Participants were suggested to describe with speech and gestures and mention *shape, size, colour* and *relative positions* of the objects. They also had several minutes to play with the Leap sensor to get familiar with the effective tracking area of the sensor.
In total, 13 participants (native German speakers) took part in the experiment (None of them took part in the previous experiment). Each of them described for 20 minutes.

4.3. Data processing

The data was processed and annotated in the same way as previous experiment. A sample description is shown as follows:

(3) a) Hier$_{[deixis]}$ ist ein kleines Quadrat, in rot, hier$_{[deixis]}$ ist ein hellblauer kleiner Kreis und hier$_{[deixis]}$ ist ein blauer grosser Kreis.

b) *Here*$_{[deixis]}$ *is a small square, red, here*$_{[deixis]}$ *is a light blue small circle and here*$_{[deixis]}$ *is a blue big circle.*

Scene representation We represented each scene as a composition of three objects. Each object was represented with 4 attributes: *colour*, *shape*, *size* and *position*. For example, a real valued position coordinates can be represented as $x : 0.1, y : 0.2$. The position was further discreted into *top, middle, bottom* vertically and *left, middle, right* horizontally. So that with the scene ID and object ID in each multimodal description, corresponding object attributes can be retrieved to reconstruct the described scene. For example, the pink circle in Figure 1 is represented as following:

- SceneID: *scene 1*
  - Object ID: *object 1*
  - Colour: *red*
  - Shape: *circle*
  - Coordinates: {*x: 0.22, y: 0.54*}
  - Horizontal position: *left*
  - Vertical position: *middle*

4.4. Preliminary analysis

Varied verbal descriptions Although participants were suggested to encode *colour*, *shape*, *size* and *relative positions* of objects in the descriptions, they were allowed to form descriptions in their own way. The collected data also reflects varied verbal descriptions. For example, the same colour was described with various expressions. *Pink* was also described as *lila*. *Cyan* was sometimes referred as *light blue*. *Circles* were referred as *circle* or *ball*. The vocabulary size of the corpus is 291.

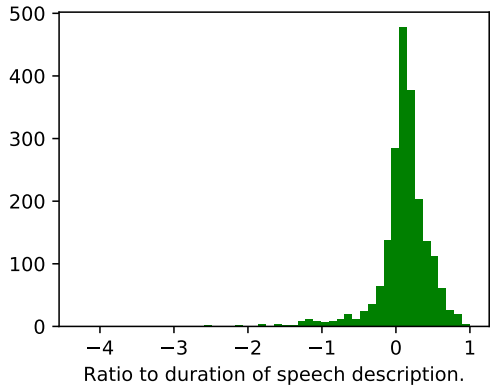


Figure 4: Temporal relations between speech and deictics.

Varied gestural behaviours From the data, we observed that when describing, sometimes participants use one hand each time to demonstrate the object position in the gesture space, hence, the listener needs to keep track of previous object positions to form a whole mental representation. Alternatively, some participants demonstrate with two hands in the gesture space to show relative positions. Among 830 description episodes, 637 descriptions (76.7%) involved the use of both hands; 193 (23.3%) with one hand. In both cases, the hand gestures convey spatial layout of the objects.

Temporal relations of speech and gestures Speech and co-verbal gestures are in parallel, and bear close temporal relations between each other (Ragsdale and Fry Silvia, 1982). We analysed the temporal relations of start timings between speech and gestures, as shown in Figure 4. Among 2074 speech-deictic ensembles, 24.5% deictics precede accompanied verbal description; 47.3% deictics occur in the first quarter of verbal descriptions. The parallel characteristics could benefit multimodal interpretation tasks on the incremental level (Han et al., 2018).

Indicating shape/size with deictics Deictic gestures have been extensively studied for positional information. However, humans often encode more than positional information while "pointing". In the collected data, we observed that, beside positional information, participants also encode shape and size information in gestures. For instance, some participants used different hand shapes when referring to circles and squares. Moreover, when mentioning objects with larger sizes, they tend to form larger hand spheres. This suggests that in future work, gesture interpretations should consider various dimensions of the information.

5. Availability

The hand motion data, anonymised transcriptions and annotations of the second experiment are publicly available[5] under the ODC Public Domain Dedication and Licence (PDDL).[6] To access the audio and video recordings, please contact the authors. Instructions on how to use the data are also available `https://tingh.github.io/resources/scene_description`.

[5] `https://pub.uni-bielefeld.de/data/2913177`

[6] `https://opendatacommons.org/licenses/pddl/1.0/`

6. Conclusion

We presented a corpus of multimodal descriptions, in which speech and gestures were used to describe spatial configurations of objects. We described the task designs, recording setups as well as the data annotation scheme. To investigate the usability of the corpus, we also provided preliminary analysis results concerning language, gesture behaviours and multimodal behaviours, then discussed possible use cases of the corpus such as modelling the interpretation of multimodal descriptions and generating multimodal behaviours.

7. Acknowledgments

This work was supported by the China Scholarship Council and the Cluster of Excellence Cognitive Interaction Technology CITEC (EXC 277) at Bielefeld University funded by the German Research Foundation (DFG). This work was also supported by the German Academic Exchange Service (DAAD) and sponsored by the German Federal Ministry of Education and Research (BMBF).

8. Bibliographical References

Alibali, M. (2005). Gesture in spatial cognition: Expressing, communicating, and thinking about spatial information. *Spatial Cognition and Computation*, 5(4):307–331.

Cassell, J., Kopp, S., Tepper, P., Ferriman, K., and Striegnitz, K. (2007). Trading spaces: How humans and humanoids use speech and gesture to give directions. *Conversational informatics*, pages 133–160.

Cheng, H., Yang, L., and Liu, Z. (2016). Survey on 3d hand gesture recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(9):1659–1673.

Emmorey, K., Tversky, B., and Taylor, H. a. (2000). Using space to describe space: Perspective in speech, sign, and gesture. *Spatial Cognition and Computation*, 2(3):157–180.

Fotinea, S.-E., Efthimiou, E., Koutsombogera, M., Dimou, A.-L., Goulas, T., and Vasilaki, K. (2016). Multimodal resources for human-robot communication modelling. In *LREC*.

Han, T., Kennington, C., and Schlangen, D. (2018). Placing Objects in Gesture Space: Toward Real-Time Understanding of Spatial Descriptions. In *Proceedings of the thirty-second AAAI conference on artificial intelligence (AAAI18)*. The association for the advancement of artificial intelligence.

Kelley, J. F. (1983). An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '83, pages 193–196, New York, NY, USA. ACM.

Kendon, A. (1980). Gesticulation and speech: two aspects of the process of utterance. *The Relationship of Verbal and Nonverbal Communication*, 25:207–227.

Kousidis, S., Pfeiffer, T., and Schlangen, D. (2013). MINT . tools : Tools and Adaptors Supporting Acquisition , Annotation and Analysis of Multimodal Corpora. In *Proceedings of Interspeech 2013*, pages 2649–2653, Lyon, France. ISCA.

Liu, L. and Shao, L. (2013). Learning discriminative representations from rgb-d video data. In *IJCAI*, volume 4, page 8.

Lücking, A., Bergmann, K., Hahn, F., Kopp, S., and Rieser, H. (2010). The bielefeld speech and gesture alignment corpus (saga). In *LREC 2010 workshop: Multimodal corpora–advances in capturing, coding and analyzing multimodality*.

Marin, G., Dominio, F., and Zanuttigh, P. (2014). Hand gesture recognition with leap motion and kinect devices. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 1565–1569. IEEE.

Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K. E., and Ansari, R. (2002). Multimodal human discourse: gesture and speech. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 9(3):171–193.

Ragsdale, J. D. and Fry Silvia, C. (1982). Distribution of kinesic hesitation phenomena in spontaneous speech. *Language and Speech*, 25(2):185–190.

Sadeghipour, A. and Morency, L.-P. (2011). 3D Iconic Gesture Dataset.

Schiel, F., Steininger, S., and Türk, U. (2002). The smartkom multimodal corpus at bas. In *LREC*.

Schneider, L. F. and Taylor, H. a. (1999). How do you get there from here? Mental representations of route descriptions. *Applied Cognitive Psychology*, 13(September 1998):415–441.

Tompson, J., Stein, M., Lecun, Y., and Perlin, K. (2014). Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics*, 33, August.

The Effects of Unimodal Representation Choices on Multimodal Learning

Fernando Tadao Ito, Helena de Medeiros Caseli, Jander Moreira
Universidade Federal de São Carlos
Rod. Washington Luís Km 235, Postal Box 676, CEP: 13565-905, São Carlos-SP
f.tadaoito@gmail.com, helenacaseli@dc.ufscar.br, jander@dc.ufscar.br

Abstract

Multimodal representations are distributed vectors that map multiple modes of information to a single mathematical space, where distances between instances delineate their similarity. In most cases, using a single unimodal representation technique is sufficient for each mode in the creation of multimodal spaces. In this paper, we investigate how different unimodal representations can be combined, and argue that the way they are combined can affect the performance, representation accuracy and classification metrics of other multimodal methods. In the experiments present in this paper, we used a dataset composed of images and text descriptions of products that have been extracted from an e-commerce site in Brazil. From this dataset, we tested our hypothesis in common classification problems to evaluate how multimodal representations can differ according to their component unimodal representation methods. For this domain, we selected eight methods of unimodal representation: LSI, LDA, Word2Vec, GloVe for text; SIFT, SURF, ORB and VGG19 for images. Multimodal representations were built by a multimodal deep autoencoder and a bidirectional deep neural network.

Keywords: distributed representations, multimodal representation, multimodal autoencoders, classification

1. Introduction

In the real world, multiple modes of information are gathered to create knowledge in a way humans can understand. From structured language, we can express abstract ideas in a standardized fashion; while visualization can help us to detect and observe objects and intention. However, in mathematical terms, each kind of data has different intrinsic statistical features that cannot be compared in a trivial way.

To analyze these features, we use distributed representation models to map real-world unimodal data to mathematical vectors with reduced dimensionality (Van Gelder, 1992). This idea is based on the distributional hypothesis (which states that "items with similar distributions have similar meanings") to discover non-trivial patterns and relationships between data instances (Harris, 1954).

The challenge of melding different data domains into a single conceptual space is the goal of **multimodal representations**. A multimodal representation model is a function that maps objects that have multiple representations to a single vectorial space, combining the semantic implications that are expressed by distance similarities in unimodal representations (Atrey et al., 2010). Using a multimodal model, we can correlate words in a textual description of an object to the visual features of an image representing it, learning its "concept". For example, a multimodal model could take as input the image of a sofa and return words that have similar meaning such as "chair" and "armchair", or take one of these words as input and output a related image. The hypothesis of a multimodal model is that it can find features that are invariant to information modality, intrinsic to the concept, and can be subsequently used in any machine learning task (e.g. e-commerce product classification).

In this paper, we explore unimodal and multimodal representations and measure their performance in a classification task using real-world data. Our hypothesis is that multimodal representations are directly influenced by the underlying unimodal representations used in its creation. Previous works on the area have already used multimodal techniques to increase accuracy in multimedia classification tasks (Ngiam et al., 2011). In our work, we want to expand these previous works by assessing new combinations of representations.

2. Related Works

Early multimodal methods focused on representation fusions, either by combining representations before classification (*feature level* fusion) or by combining the results of classifications performed in single-mode representations in another analysis (*decision level* fusion) (Atrey et al., 2010). Feature level fusion has been used in multimodal pedestrian tracking using multiple detection algorithms (Yang et al., 2005) and traffic surveillance with multiple video sources (Wang et al., 2003). Decision level fusion can be compared to ensemble learning (Dietterich, 2000), combining different algorithms and data sources to create a more accurate representation.

One of the first multimodal model-based representation approaches is the one described by Ngiam et al. (2011), applying Restricted Boltzmann Machines (RBM) (Salakhutdinov and Hinton, 2009) and autoencoders (Bourlard and Kamp, 1988) for video-audio joint representations. The unimodal representations are first used to pre-train the input layers of the multimodal autoencoder network via RBM unsupervised training. The initial weights are then used in the complete autoencoder network trained using single-modality data and multi-modality data. In that paper, the grayscale image is cropped in a specific region of interest (the mouth), rescaled into a 60×80 pixel matrix and then translated to a reduced dimensionality space (32 dimensions) by Principal Component Analysis (PCA) whitening (Friedman, 1987). The audio was represented by spectrogram signals with temporal derivatives, resulting in a 483 dimension vector further reduced (100 dimensions) by PCA whitening. Using digit images from the MNIST dataset (LeCun and Cortes, 2010) and noisy variants as a second mode, multimodal features performed better on an image classification task when compared to single-mode feature

representations.

Expanding on the multimodal approach introduced above, Andrew et al. (2013) use two different autoencoders and align the projections generated by maximizing the canonical correlation between the features. This model assumes that the projection of a vectorial space into the other is sufficient to portrait the features shared by two different modalities. This model was not tested in a classification task, only the correlation of the generated features was taken into account. The databases used (MNIST and XRMB (Westbury, 1994)) did not undergo any preprocessing or representation methods. This model creates correlated projections of data that share information between modalities without needing both modes to be calculated.

Wang et al. (2016) present an overview of multimodal techniques, including the ones described by Ngiam et al. (2011) and Andrew et al. (2013), expand on their objective functions and propose new architectures fusing these objective functions. In particular, the Deep Canonical Correlated Autoencoder (DCCAE) uses both objectives from previous works simultaneously to increase correlation between obtained features and fidelity to original data. This architecture is then tested and compared with others in unsupervised classification tasks in noisy MNIST and XRMB datasets: accuracy was measured by how the representations were clustered together, and multimodal representations had the best results.

Vukotić et al. (2016) present a new approach to multimodal fusion using paired crossmodal neural networks (BiDNN). Two neural networks are created, each mapping one modality to another directly, with the weights of hidden layers of both networks tied. The result of this training is a central layer in both networks that maps any given modality to a shared representation space between the two given modalities. This architecture is compared to others using metrics obtained from analyzing the MediaEval 2014 dataset (Eskevich et al., 2014) using audio transcripts and video segments to link a video to a specific concept (anchor), obtaining superior results in this classification task. The transcripts were represented with Word2Vec embeddings, and the video was transcribed to human visual concepts.

3. Methodology

In this section, we describe the representations used for comparison in our experiments. Four methods of each modality were chosen. For texts, we used: (1) Latent Semantic Indexing (Landauer et al., 1998) and (2) Latent Dirichlet Analysis (Blei et al., 2003) for topical representation, and (3) Word2Vec (Mikolov et al., 2013) and (4) Global Vectors for word representation (Pennington et al., 2014). For images, we applied: (1) Scale-Invariant Feature Transform (Lowe, 2004), (2) Speeded-Up Robust Features (Bay et al., 2008), (3) Oriented FAST and Rotated BRIEF (Rublee et al., 2011) for Bag-of-Visual-Words (Yang et al., 2007) vector generation, and (4) neural features obtained from the VGG19 pre-trained network (Simonyan and Zisserman, 2014).

3.1. Textual Representation

Textual distributed representation is based on the aforementioned distributional hypothesis (Harris, 1954) which states that similar statistical distributions denotes semantic similarity between two items. In the textual domain, this means that words that appear in the same context have similar semantic meaning.

In this paper, we review methods of textual representation from two ways: one by **topic representation**, obtaining vectors that encode the pertinence of words in sentences if they appear together in a *corpus*, and one by **word embeddings**, creating word representations based on their occurrence context and combining them in a single document vector. Thus, the methods for textual representation investigated in this paper are:

- Latent Semantic Indexing (LSI) (Landauer et al., 1998): This method uses the term-document matrix that encodes the frequency of each term by document and its eigenvalues and eigenvectors to decompose this data and find a new representation. This is accomplished by Singular Value Decomposition, selecting only the highest eigenvalues and their correspondent eigenvectors to recompose the term-document matrix, leading to a reduced topic-document matrix.

- Latent Dirichlet Analysis (LDA) (Blei et al., 2003): Similar to LSI, this method codifies a topical distribution of words using a term-document matrix. But, instead of matrix operations to simplify the original data, it uses probability and parameter estimations to find word-topic and topic-document distributions.

- Word2Vec (W2V) (Mikolov et al., 2013): This neural model uses a shallow neural network to create distributed representations based on the context of each word. This model can be trained in two ways: training the model to find the context of a given word (Skipgram) or find the central word of a given context (CBoW). Either way, the model codifies a representation of a word in its hidden layer. The simplest way of codifying a document vector from its words is to add all present vectors and divide them by the number of words in it, as the semantic information is kept on this vector combination. This document vector performs poorly in large texts, as it loses semantic information shared between contexts the same way as the standard Bag-of-Words representation does. Another way of codification of document vectors that circumvents the aforementioned limitation is Doc2Vec (Le and Mikolov, 2014), adding a document identification token to each context and then calculating its embedding.

- Global Vectors for Word Representation (GloVe) (Pennington et al., 2014): This statistical model uses a term co-ocurrence matrix and ratios of co-ocurrence to find word vectors with distances between words relative to their co-ocurrence ratios. In the example given by the proposing paper, the ratio of $\mathrm{P}(solid|ice)/\mathrm{P}(solid|steam)$ is much higher than 1,

meaning that "solid" is more semantically related to "ice" than "steam". The objective function of this model reflects this ratio in the distances between these word vectors.

3.2. Visual Representation

A digital image is represented by a matrix of pixels (tuples of numbers with intensities of particular color channels). Albeit great for visualization, this matrix often has highly correlated neighboring pixels, creating redundant data. To extract meaningful mathematical information from an image, we must first find regions of interest that uniquely define it, and map these to a vector space where they can be compared.

This paper uses two different ways to generate features from an image: **hand-crafted descriptors**, which are predefined ways to find regions of interest and encode them into vectors; and **neural features**, automatically extracted and selected by an ImageNet pre-trained neural network. Thus, the methods for visual representation investigated in this paper are:

- Scale-Invariant Feature Transform (SIFT) (Lowe, 2004): This method extracts features that are invariant to scale, illumination and rotation. It is composed of four main steps: (1) keypoint extraction, via Differences of Gaussians in different scales; (2) keypoint localization, to refine and filter extracted keypoints; (3) orientation assignment, for each keypoint to achieve rotation invariance; (4) keypoint description, using the histogram of gradients in the neighborhood of the keypoint to encode a 128-position vector.

- Speeded-Up Robust Features (SURF) (Bay et al., 2008): As an extension of the SIFT method, SURF follows the same steps of SIFT applying different mathematical methods. For keypoint extraction and localization, the Differences of Gaussians are replaced by Box Filters and Hessian Matrix determinants; for orientation assignment and keypoint description, SURF uses Haar Wavelet responses around the keypoint. SURF achieves similar results to SIFT, generating smaller vectors (64-positions) with improved speed.

- Oriented FAST and Rotated BRIEF (ORB) (Rublee et al., 2011): This method is a fusion of two descriptors: FAST (Rosten and Drummond, 2006) and BRIEF (Calonder et al., 2010). ORB uses the FAST keypoint extraction method, achieves rotation invariance by analyzing the weighted centroid of intensities around each keypoint, and then combines this orientation with the BRIEF descriptor by prior rotation of the pixels in the described keypoint neighborhood.

- VGG19 classes (Simonyan and Zisserman, 2014): This method uses the classes detected from the VGG19 model pre-trained using the images from ImageNet, creating a probability vector of 1000 positions, probabilities of specific objects in a scene.

3.3. Multimodal Representation

To combine multiple data representations in a simple way, we can use feature concatenation prior to classification (Wang et al., 2003), create ensembles of single-mode classifiers (Radová and Psutka, 1997), align features of single-mode representations by similarity measures (Frome et al., 2013) or co-learn single-modality representations using another modality as basis (Information Resources Management Association, 2012, Chapter 28). But these representations do not encode a real fusion between two modalities in a **single mathematical vector space**, as they only add information to an existing space or combine inferences from separate spaces in a shallow way.

In this paper, we use both early-stage feature fusion with concatenation (Wang et al., 2003) and model-based feature fusion, creating a single vector space that maps both modalities to a single vectorial space. In this paper, we will use a simple multimodal autoencoder framework with single-mode pre-training, in a similar architecture to the one proposed in Ngiam et al. (2011):

1. Two deep autoencoders are pre-trained with single modalities until convergence;

2. The weights are then ported to a multimodal deep autoencoder with one (or more) shared hidden layers that will codify our multimodal representation;

3. Training will have instances that will try to reconstruct two modalities from one, reconstruct one modality from two, and reconstruct one modality based on the other.

The architecture described in Vukotić et al. (2016) (BiDNN) will also be used to compare results between the multimodal approaches. The simple autoencoder described above was also compared in Vukotić et al. (2016), and we replicate this experiment with our dataset.

4. Experiments

The experiments described in this paper were executed in a multimodal dataset composed of 6,400 textual descriptions of e-commerce electronic products paired with their respective images. Not all the textual descriptions have a related image. The original downloaded images have $(55 \times 55 \times 3)$ pixels each. Each product is assigned to a class automatically extracted by web crawlers.

The 10 classes in this dataset are:

- **Automotive** – items related to automobile sound systems, tires and related electronic gadgets;

- **House and Electronics** – products related to kitchen appliances and house maintenance;

- **Games** – video-game related products such as consoles, joysticks and games;

- **Hardware** – computer components such as processors, GPUs and coolers;

- **Computing** – stand-alone products that are related to computers such as routers, no-breaks and power cables;
- **Stationary** – office products such as organizing boxes and tacks;
- **Peripherals** – computer peripherals such as headphones, keyboards and mouses;
- **Used** – an amalgam of all the other products in used conditions and user-made descriptions;
- **Smartphones** – smartphones and its accessories;
- **Telephony** – landlines, cables and radios.

Single-modality and multi-modality experiments have been executed with five simple classification algorithms:

- Support Vector Machine with both stochastic gradient descent (SGD) and logistic regression learning (SVC);
- Binary Support Vector Machine (one-versus-all approach) with both stochastic gradient descent (SGD-B) and logistic regression learning (SVC-B);
- Multilayer Dense Neural Network (MLP) with 256, 128 and 64 neurons on each layer, respectively;

All representations except the VGG19-derived ones were generated in a 128 dimensional space. Word2Vec and GloVe word representations were calculated for all words in a document and averaged to create a 128-dimension vector. To generate Word2Vec and GloVe representations, we used the tokenized texts without rare (less than 2 occurrences on the corpus) or large (more than 50 letters) tokens. To generate LSI and LDA word representations, each text was first converted to Bag-of-Words and these count values were used to generate *term frequency–inverse document frequency* (TFIDF) (Wu and Salton, 1981) values. Each image had its SURF, SIFT and ORB descriptors extracted, and a Bag-of-Visual-Words was generated by a k-means clustering algorithm, with $k = 128$. Images were passed through the VGG19 network and a vector with 1000 positions was generated for each image. The scripts were written in Python, with CUDA acceleration (Nickolls et al., 2008) and Tensorflow (Abadi et al., 2015) in the neural network training phases. The classification experiment used 10-fold cross-validation.

4.1. Baselines

Firstly, we tested single-modality classification and early-stage feature fusion, concatenating the vectors to generate 256-dimension vectors (or 1128-dimension vectors in the VGG19 modality) for combined representations.

The combinations were made with two different textual representations (text×text), two different image representations (image×image) or one representation of each (image×text). Although simple, this approach improved classification accuracy and recall on some multimodal combinations, with the best results depicted in bold on Table 1. These results show that textual classification outperforms its image counterpart by a large amount in this domain.

Modalities	Representation	SGD	SVC	MLP	SGD-B	SVC-B
Text	GloVe	0.71	0.74	0.80	0.70	0.74
	W2V	0.70	0.74	0.80	0.71	0.73
	LDA	0.62	0.68	0.75	0.65	0.68
	LSI	0.84	0.85	**0.87**	0.83	**0.87**
Image	SIFT	0.39	0.38	0.40	0.38	0.40
	SURF	0.36	0.38	0.36	0.36	0.38
	ORB	0.29	0.32	0.30	0.29	0.29
	VGG19	0.38	0.44	**0.57**	0.40	0.45
Image×Image	SIFT+SURF	0.45	0.42	0.43	0.41	0.41
	SIFT+ORB	0.41	0.40	0.41	0.41	0.37
	SIFT+VGG19	0.47	0.50	**0.53**	0.45	0.49
	SURF+ORB	0.38	0.38	0.39	0.38	0.36
	SURF+VGG19	0.46	0.49	0.52	0.48	0.48
	ORB+VGG19	0.45	0.45	0.49	0.46	0.46
Text×Text	GloVe+W2V	0.76	0.79	0.83	0.76	0.81
	GloVe+LDA	0.76	0.78	0.81	0.76	0.79
	GloVe+LSI	0.83	0.85	0.84	0.82	0.86
	W2V+LDA	0.75	0.78	0.82	0.75	0.78
	W2V+LSI	0.86	**0.88**	0.87	0.85	0.86
	LDA+LSI	0.84	0.86	0.84	0.85	0.87
Image×Text	SIFT+W2V	0.72	0.72	0.74	0.71	0.75
	SIFT+GloVe	0.74	0.71	0.80	0.71	0.72
	SIFT+LDA	0.66	0.68	0.69	0.66	0.68
	SIFT+LSI	0.84	0.84	0.84	0.83	**0.86**
	SURF+W2V	0.70	0.73	0.75	0.71	0.72
	SURF+GloVe	0.68	0.71	0.80	0.72	0.71
	SURF+LDA	0.65	0.66	0.67	0.65	0.65
	SURF+LSI	0.84	0.84	0.83	0.83	0.85
	ORB+W2V	0.68	0.71	0.73	0.71	0.69
	ORB+GloVe	0.70	0.72	0.76	0.71	0.69
	ORB+LDA	0.61	0.66	0.65	0.61	0.66
	ORB+LSI	0.83	0.84	0.81	0.82	0.85
	VGG19+W2V	0.73	0.74	0.78	0.72	0.74
	VGG19+GloVe	0.70	0.74	0.81	0.69	0.71
	VGG19+LDA	0.68	0.72	0.75	0.70	0.72
	VGG19+LSI	0.80	0.83	0.83	0.81	0.84

Table 1: Unimodal and early-stage multimodal fusion experiments. Metric used is F-Score. Best scores for each modality are highlighted in bold.

Modalities	Representation	MMAE	BIDNN
Image×Image	SIFT+SURF	0.38	0.37
	SIFT+ORB	0.37	0.36
	SIFT+VGG19	**0.39**	0.44
	SURF+ORB	0.35	0.34
	SURF+VGG19	**0.39**	**0.46**
	ORB+VGG19	0.32	0.43
Text×Text	GloVe+W2V	0.73	0.79
	GloVe+LDA	0.60	0.76
	GloVe+LSI	0.50	0.82
	W2V+LDA	**0.74**	0.82
	W2V+LSI	**0.74**	**0.84**
	LDA+LSI	0.61	0.83
Image×Text	SIFT+W2V	0.66	0.75
	SIFT+GloVe	0.52	0.61
	SIFT+LDA	0.60	0.66
	SIFT+LSI	0.32	0.74
	SURF+W2V	0.68	0.72
	SURF+GloVe	0.49	0.61
	SURF+LDA	0.58	0.63
	SURF+LSI	0.38	0.77
	ORB+W2V	**0.70**	0.72
	ORB+GloVe	0.44	0.54
	ORB+LDA	0.61	0.62
	ORB+LSI	0.39	0.73
	VGG19+W2V	0.67	0.73
	VGG19+GloVe	0.51	0.54
	VGG19+LDA	0.65	0.67
	VGG19+LSI	0.37	**0.79**

Table 2: Multimodal fusion experiments. Metric used is F-Score on a Multilayer Perceptron model. Best scores between modalities and algorithms are highlighted in bold.

4.2. Our approach

After finding our baselines, we tested our proposed approach by using multimodal features generated by a Deep Multimodal Autoencoder (MMAE) (Ngiam et al., 2011)

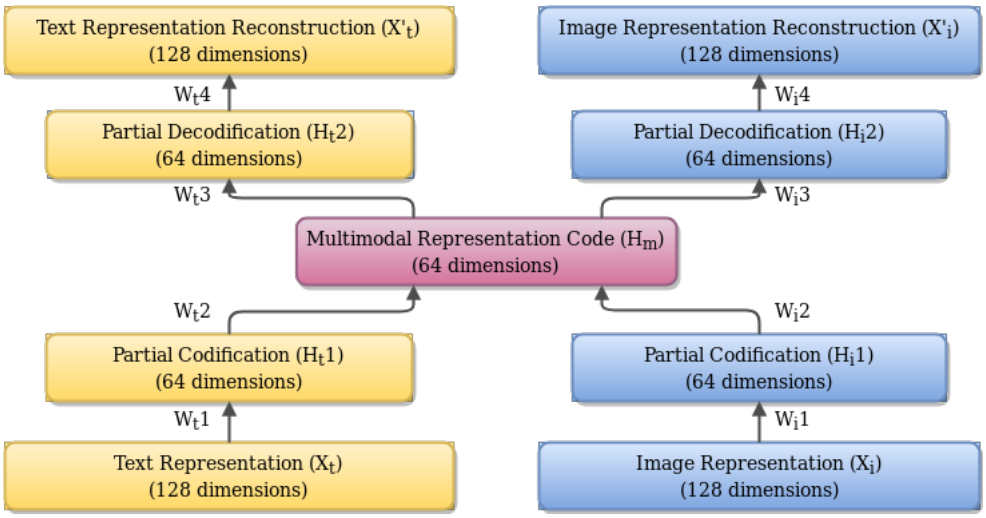


Figure 1: Deep Multimodal (MMAE) architecture used in the experiments.

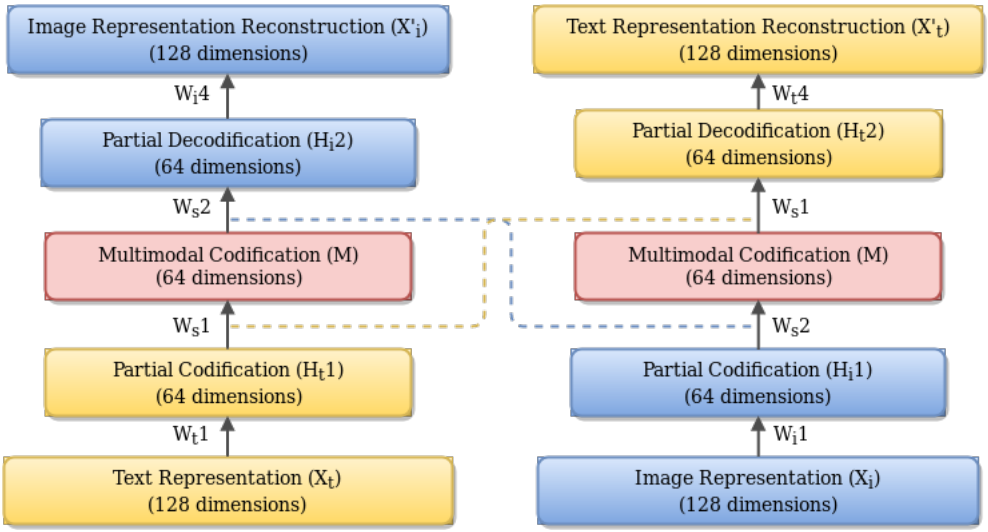


Figure 2: Bidirectional Deep Autoencoder (BIDNN), with shared weights.

and a Bidirectional Neural Network (BIDNN) (Vukotić et al., 2016) with tied weights, as illustrated in Figures 1 and 2.

These features have 64 dimensions, fusing text and image representations. Both models were trained in batches of 256 instances for 1000 epochs, optimized using ADADELTA (Zeiler, 2012) with binary cross-entropy loss. The results of these tests are depicted in Table 2. Only the Multilayer Perceptron classification model results are shown in this table, as the other models achieved poor F-Scores (below 0.4).

The results show that the Bidirectional Neural Network is consistently better in this task, corroborating the experiments in (Vukotić et al., 2016). But, when compared to the previous experiment, lower F-Scores were obtained in the neural multimodal features: the best results obtained with these features were 0.74 (Multimodal Autoencoder) and 0.84 (Bidirectional Neural Network), lower than the results obtained by early-fusion of textual features (0.87 using W2V+LSI) and comparable to the results of unimodal features (GloVe, W2V and LSI had results higher or equal than 0.80).

5. Discussion

As can be noticed by the values in Table 1, LSI seems to be the best text representation model with 0.87 F-Score with MLP and SVC-B. It is also the best one to combine with image representation models since its performance with SIFT (SIFT+LSI), SURF (SURF+LSI) and ORB (ORB+LSI) was really close to the best values achieved when only text representations were used. We believe that this LSI's best performance is related to the fact that most descriptions of e-commerce products are mainly composed of highly specialized keyword groups (*eg.*, "smartphone" and "mobile" in the "smartphone" class, and "DPI" and "anti-ghosting" in "peripherals"), thus topical representations can adequately summarize concepts in this domain. And, as these products rarely share the same visual concepts of oth-

ers in their class (*eg.*,"smartphones" had images of smartphones, smartphone cables and accessories), visual descriptors are not efficient in this task. From these early-stage multimodal results we can conclude that text (at least, LSI) can help image in the classification scenario tested in our experiments.

Regarding our proposed neural multimodal approach, we could not significantly improve classification accuracy using multimodal features obtained via neural means (MMAE and BIDNN). In order to visualize the similarity between representations, we generated comparison images for selected products [1]. The four most similar products (top-4) to a target image of a printer are depicted in Figures 3 (using only neural image features), 4 (using only textual features) and 5 (using multimodal neural features), from a special receipt printer. In Figure 3, image representations could approximate other printers and rectangular objects to its immediate vectorial vicinity. In Figure 4, using textual representations, the two closest product vectors had no relation whatsoever to printing: one represents a repair toolkit for a CPU cooler and the other is a motion sensor. In Figure 5, using multimodal features, the four closest products are all printers, and the first two are receipt printers just as the target product is.

Analyzing product similarity, we can see that some products can be grouped more accurately using multimodal features (as seen in Figure 5). The information gain from multimodal features was not relevant for classification in our corpus since there is too much noise in certain categories (*eg.*, in Figure 4 in which a printer, a cable and a sensor were grouped together in the same category). However, we think that multimodal features can be used in other tasks involving similarity such as e-commerce search queries: improving comparison speed and quality using these features to navigate monolithic databases can increase site performance.

As there are few linguistic resources for the Portuguese language in this area, we intend to make this set a starting point for other works. The dataset can be obtained on request, and the code will be released in a public repository after the publication of this paper.

6. Acknowledgements

This research is part of the MMeaning project, supported by São Paulo Research Foundation (FAPESP), grant #2016/13002-0, and was also partly funded by the Brazilian Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

7. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Andrew, G., Arora, R., Bilmes, J., and Livescu, K. (2013). Deep Canonical Correlation Analysis. Proceedings of the 30th International Conference on Machine Learning (ICML-13), 28:1247–1255.

Atrey, P. K., Hossain, M. A., El Saddik, A., and Kankanhalli, M. S. (2010). Multimodal fusion for multimedia analysis: A survey. Multimedia Systems, 16(6):345–379.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding, 110(3):346–359.

Blei, D. M., Edu, B. B., Ng, A. Y., Edu, A. S., Jordan, M. I., and Edu, J. B. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research, 3:993–1022.

Bourlard, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. Biological Cybernetics, 59(4):291–294, Sep.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). BRIEF: Binary robust independent elementary features. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 6314 LNCS, pages 778–792.

Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. Multiple Classifier Systems, 1857:1–15.

Eskevich, M., Aly, R., Racca, D. N., Ordelman, R., Chen, S., and Jones, G. J. (2014). The search and hyperlinking task at MediaEval 2014. In CEUR Workshop Proceedings, volume 1263.

Friedman, J. H. (1987). Exploratory projection pursuit. Journal of the American Statistical Association, 82(397):249–266.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al. (2013). Devise: A deep visual-semantic embedding model. In Advances in neural information processing systems, pages 2121–2129.

Harris, Z. S. (1954). Distributional structure. Word, 10(2-3):146–162.

Information Resources Management Association. (2012). Data mining: Concepts, methodologies, tools, and applications.

Landauer, T. K., Folt, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. Discourse processes, 25(2):259–284.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In Eric P. Xing et al., editors, Proceedings of the 31st International Conference on Machine Learning, volume 32 of Proceedings of Machine Learning Research, pages 1188–1196, Bejing, China, 22–24 Jun. PMLR.

LeCun, Y. and Cortes, C. (2010). MNIST handwritten digit database. AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist.

Lowe, D. G. (2004). Distinctive image features from scale-

[1]The representations generated in the previous experiments were used to measure the vector distance between these selected products and all others in their vector space.



Figure 3: Top-4 similar products using VGG19-generated features.



Figure 4: Top-4 similar products using LSI-generated features.

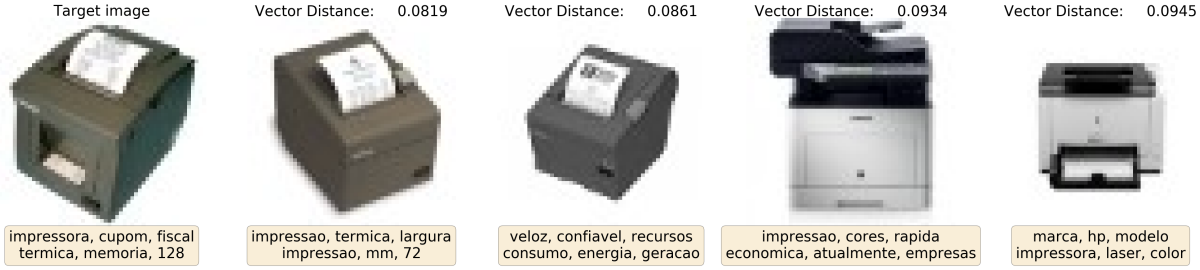


Figure 5: Top-4 similar products using Bidirectional Neural Network features.

invariant keypoints. International Journal of Computer Vision, 60(2):91–110.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, pages 1–12.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). Multimodal Deep Learning. Proceedings of The 28th International Conference on Machine Learning (ICML), pages 689–696.

Nickolls, J., Buck, I., Garland, M., and Skadron, K. (2008). Scalable parallel programming with CUDA. AMC Queue, 6(April):40–53.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543.

Radová, V. and Psutka, J. (1997). An approach to speaker identification using multiple classifiers. In Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on, volume 2, pages 1135–1138. IEEE.

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 3951 LNCS, pages 430–443.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision, pages 2564–2571.

Salakhutdinov, R. and Hinton, G. (2009). Deep belief networks. Scholarpedia, pages 4–5.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556.

Van Gelder, T. (1992). Defining 'distributed representation'. Connection Science, 4(3/4):175.

Vukotić, V., Raymond, C., and Gravier, G. (2016). Bidirectional joint representation learning with symmetrical deep neural networks for multimodal and cross-modal applications. In Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, ICMR '16, pages 343–346, New York, NY, USA. ACM.

Wang, J., Kankanhalli, M. S., Yan, W., and Jain, R. (2003). Experiential sampling for video surveillance. In First ACM SIGMM international workshop on Video surveillance, pages 77–86. ACM.

Wang, W., Arora, R., Livescu, K., and Bilmes, J. (2016). On Deep Multi-View Representation Learning: Objectives and Optimization. Arxiv Preprint, XX(XX):XX.

Westbury, J. (1994). X-ray Microbeam Speech Production Database User's Handbook: Version 1.0 (June 1994). Waisman Center on Mental Retardation & Human Development.

Wu, H. and Salton, G. (1981). A comparison of search term weighting: term relevance vs. inverse document frequency. In ACM SIGIR Forum, volume 16, pages 30–39.

Yang, M.-T., Wang, S.-C., and Lin, Y.-Y. (2005). A multimodal fusion system for people detection and tracking. International journal of imaging systems and technology, 15(2):131–142.

Yang, J., Jiang, Y.-G., Hauptmann, A. G., and Ngo, C.-W. (2007). Evaluating bag-of-visual-words representations in scene classification. In Proceedings of the international workshop on Workshop on multimedia information retrieval - MIR '07, page 197.

Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. CoRR, abs/1212.5701.

An Evaluation Framework for Multimodal Interaction

Nikhil Krishnaswamy and James Pustejovsky
Brandeis University
Waltham, MA, USA
{nkrishna, jamesp}@brandeis.edu

Abstract

In this paper we present a framework for evaluating interactions between a human user and an embodied virtual agent that communicates using natural language, gesture, and by executing actions in a shared context created through a visual simulation interface. These interactions take place in real time and demonstrate collaboration between a human and a computer on object interaction and manipulation. Our framework leverages the semantics of language and gesture to assess the level of mutual understanding during the interaction and the ease with which the two agents communicate. We present initial results from trials involving construction tasks in a blocks world scenario and discuss extensions of the evaluation framework to more naturalistic and robust interactions.

Keywords: multimodal, evaluation, semantics, objects, events, language, gesture, simulation, action, vision

1. Introduction

As the field of computational linguistics evolves and more sophisticated natural language systems become integrated with everyday use, naive users will come to expect their interactions to approximate what they are familiar with when communicating with another human, multimodally. With increased interest in multimodal interaction comes a need to evaluate the performance of a multimodal system on all levels with which it engages the user. Such evaluation should be modality-agnostic and assess the success of communication between human and computer, based on the semantics of objects, events, and actions situated within the shared context created by the human-computer interaction.

We use the modeling language VoxML (Pustejovsky and Krishnaswamy, 2016) as the platform for modeling the aforementioned objects, events, and actions, and use the VoxML-based simulation implementation VoxSim to create the environment in which a multimodal interaction involving natural language and gesture takes place. This allows us to exercise VoxML object and event semantics to assess conditions on the success or failure of the interaction.

2. Multimodal Interaction

A wealth of prior work exists on the role of gestural information in human-computer interaction. "Put-that-there" (Bolt, 1980) included deixis for disambiguation, and inspired a community surrounding multimodal integration (Dumas et al., 2009; Kennington et al., 2013; Turk, 2014).

As speech and gesture are processed partially independently (Quek et al., 2002), using both modalities complementarily increases human working memory and decreases cognitive load (Dumas et al., 2009). Visual information has been shown to be particularly useful in establishing mutual understanding that enables further communication (Clark and Wilkes-Gibbs, 1986; Clark and Brennan, 1991; Dillenbourg and Traum, 2006; Eisenstein et al., 2008a; Eisenstein et al., 2008b). We will hereafter refer to this type of shared understanding as "common ground," which can be expressed in multiple modalities.

Coordination between humans using non-verbal communication (cf. Cassell (2000), Cassell et al. (2000)) can be adapted to the HCI domain, particularly in the context of shared visual workspaces (Fussell et al., 2000; Kraut et al., 2003; Fussell et al., 2004; Gergle et al., 2004). Allowing for shared gaze has been shown to increase performance in spatial tasks in shared collaborations (Brennan et al., 2008), and the co-involvement of gaze and speech have also been studied in interaction with robots and avatars (Mehlmann et al., 2014; Skantze et al., 2014; Andrist et al., 2017).

In the context of shared physical tasks in a common workspace, shared perception creates the context for the conversation between interlocutors (Lascarides and Stone, 2006; Lascarides and Stone, 2009b; Clair et al., 2010; Matuszek et al., 2014), and it is this shared space that gives many gestures, such as pointing, their meaning (Krishnaswamy and Pustejovsky, 2016a). Dynamic computation of discourse (Asher and Lascarides, 2003) becomes more complex with multiple modalities but embodied actions (such as coverbal gestures) fortunately do not seem to violate coherence relations (Lascarides and Stone, 2009a).

Prior work on multimodal evaluation also includes evaluation of gestural usage, although in this case *gesture* often refers to interfaces with multimodal displays, such as those on mobile devices (Oviatt, 2003; Lemmelä et al., 2008; Johnston, 2009). Evaluation of embodied virtual agents is often focused on the agent's "personality" or non-verbal actions, to help overcome the "uncanny valley" effect (Krämer et al., 2007). However, recent developments in multimodal technology and robotics provide resources on formally evaluating the success of multimodal grounding operations (e.g., Declerck et al. (2010), Hough and Schlangen (2016), Zarrieß and Schlangen (2017)), or of interactive systems (e.g., Fotinea et al. (2016)).

Many of the newest methods rely on datasets gathered using high-end technology and complex experimental setups, including motion capture, multiple depth cameras, range-finding sensors, or geometrically-calibrated accelerometry (systems rarely rely on all of these as that would be prohibitive). Our evaluation scheme is intended to be situation-agnostic and relies solely on logging the time and nature of interactions between interlocutors, conditioning on semantic elements during post-processing. In addition, using a suite of gesture-recognition software running on Titan X/Xp GPUs, the experimental setup we use relies only on

a single depth camera, a tablet computer and any machine capable of running the Unity-based VoxSim virtual world. With access to the GPUs over the internet, all required components can be minimally carried in a backpack and deployed anywhere with a fast internet connection. Coupled with a streamlined evaluation scheme, this allows data to be collected in a variety of different situations and conditions. We are concerned with evaluating a system for its effectiveness in creating mutual understanding between the human and virtual agents. An effective multimodal system should therefore support multimodal commands and shared perception, and approximate peer-to-peer conversations. We propose a semantically-informed evaluation scheme and a sample scenario for evaluation, with the expectation that a lightweight scheme for evaluating lightweight systems should scale to domain-agnostic interactions.

2.1. Gestures

Visual gesture recognition has long been a challenge for real-time systems (Jaimes and Sebe, 2007; Rieser and Poesio, 2009; Gebre et al., 2012; Madeo et al., 2016). In our demonstration system, we use Microsoft Kinect depth sensing (Zhang, 2012) and ResNet-style deep convolutional neural networks (DCNNs) (He et al., 2016) implemented in TensorFlow (Abadi et al., 2016). The system is capable of recognizing 35 independent gestures, chosen for their frequent occurrence in a prior elicitation study on human subjects (Wang et al., 2017a; Wang et al., 2017b). Seven of these are currently used in the sample blocks world task:

1. Engage. Begins the task when the human approaches the avatar, and ends it when they step back.
2. Positive acknowledge. A head nod or a "thumbs up." Used to signal agreement with avatar's choice or answer a question affirmatively.
3. Negative acknowledge. A head shake, "thumbs down," or palm-forward "stop" sign. Signals disagreement with a choice or negative response to a question.
4. Point. Deixis includes the direction of the hand and/or arm motion: one or two of front, back, left, right, up, or down. Indicates a region or object(s) in that region.
5. Grab. A "claw," mimicking grabbing an object. Tells the avatar to grasp an indicated object.
6. Carry. Moving the arm in a direction while the hand is in the grab position. "Carry up" can be thought of as pick up, while "carry down" is equivalent to put down.
7. Push. A flat hand moving in the direction of the open palm. Like "carry," but without the up and down directions. A beckoning motion signals the avatar to push an object toward the human.

Each gesture is assigned a compositional, underspecified semantics. We treat gestures as a special case of the VoxML entity type PROGRAM. Figure 1 shows an example.

Each gesture is linked to a VoxML verbal PROGRAM (e.g., the gesture in Figure 1 would also link to the verb [[PUSH]]). Each gesture and associated programs are distinguishable based on minimal pairs of features (e.g., [[PUSH]] in this vocabulary requires that fingers be pointed forward whereas [[CARRY]] keeps the fingers curved). This allows an evaluation scheme to correlate specific successes

$$\left[\begin{array}{ll} \textbf{push} & \\ \text{LEX} = & \left[\begin{array}{l} \text{PRED} = \textbf{push} \\ \text{TYPE} = \textbf{gesture} \\ \text{SEM} = \textbf{push} \end{array}\right] \\ \text{TYPE} = & \left[\begin{array}{ll} \text{HEAD} = \textbf{process} & \\ \text{ARGS} = & \left[\begin{array}{l} \text{A}_1 = \textbf{x:agent} \\ \text{A}_2 = \textbf{y:hand} \\ \text{A}_3 = \textbf{z:finger+} \\ \text{A}_4 = \textbf{v:vector} \end{array}\right] \\ \text{BODY} = & \left[\begin{array}{l} \text{E}_1 = [while([open(y) \wedge fwd(z)], \\ \qquad move(x, y, align(v))] \end{array}\right] \end{array}\right] \end{array}\right]$$

Figure 1: Sample gesture voxeme: [[PUSH]]

or failures within an interaction with the features surrounding a gesture that occurred at the same time. An example enabled by evaluation might be discovering that straight-fingered gestures are more easily interpretable than curved-fingered gestures, or that gestures without direction are less ambiguous than gestures with it.

2.2. VoxSim

The virtual avatar and blocks world are implemented with VoxSim, a semantically-informed reasoning system (Krishnaswamy and Pustejovsky, 2016b) that allows the avatar to react to gestural events with both actions and words. VoxSim is built on the platform created by VoxML "voxemes," or semantic visual objects, and therefore allows the direct interpretation of gestures mapped through dynamic semantics. A sample voxeme for a gesture is given above in Section 2.1. The system also accepts speech input for simple directions, answers to yes/no questions, and object disambiguation by attributive adjective. Further information about voxeme properties is laid out in Pustejovsky and Krishnaswamy (2016).

2.3. Scenario

The sample interaction is adopted from functionality presented in Krishnaswamy et al. (2017). In this scenario, a human and an avatar in the VoxSim environment must collaborate to complete a simple construction task using virtual blocks that are manipulated by the avatar. The human has a plan or goal configuration that they must instruct the avatar to reach using a combination of gestures and natural language instructions. The avatar in turn communicates through gestures and natural language output to request clarification of ambiguous instructions or present its interpretation of the human's commands. The human may indicate (point to) blocks and instruct the avatar to slide and move them relative to other blocks or relative to regions of the virtual table. The human must also respond to the avatar's questions, when the avatar perceives an ambiguity in the human's instructions.

3. Hallmarks of Communication

As our goal in developing multimodal interactions is to achieve naturalistic communication, we must first examine what we mean by and desire out of an interaction such as that illustrated in Section 2.3.

We take the view that a "meaningful" interaction with a computer system should model certain aspects of a similar interaction between two humans. Namely, it is one where each interlocutor has something "interesting" to say, and one that enables them to work together to achieve common

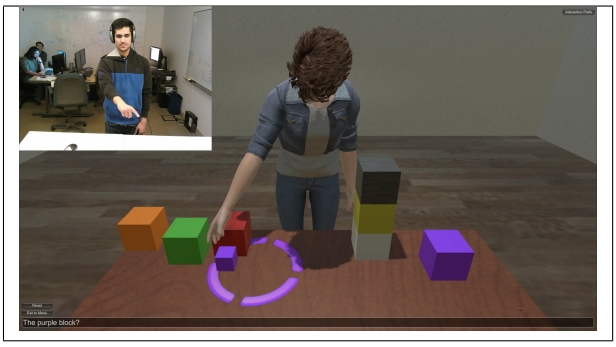


Figure 2: Example interaction setup, showing human superimposed in upper left on avatar and VoxSim

goals and build off each other's contributions, thereby conveying the impression to the user that the computer system is experiencing the same events. We therefore build the evaluation scheme off of the following qualitative metrics:

1. Interaction has mechanisms to move the conversation forward (Asher and Gillies, 2003; Johnston, 2009)
2. System makes appropriate use of multiple modalities (Arbib and Rizzolatti, 1996; Arbib, 2008)
3. Each interlocutor can steer the course of the interaction (Hobbs and Evans, 1980)
4. Both parties can clearly reference items in the interaction based on their respective frames of reference (Ligozat, 1993; Zimmermann and Freksa, 1996; Wooldridge and Lomuscio, 1999)
5. Both parties can demonstrate knowledge of the changing situation (Ziemke and Sharkey, 2001)

Many of these metrics are subjective, so we approach evaluation from a semantics-centered perspective, and use distinct semantic properties of specific elements in the interaction to determine what about the interaction enabled or hindered "shared understanding."

4. Evaluation

A robust multimodal evaluation scheme should be able to be applied to a human-computer interaction on a novel system and return a result representative of the system's *coverage* of the total possible interactions within the system's domain (e.g., construction tasks in a blocks world).
To this end, a user must be truly naive, having very little to no knowledge of exactly what the system understands. This way, during evaluation, the user has their own definition of objects, actions, and events, and the system has its own. By interacting with the system, the human should be able to learn its ways of acting within the task domain, what it knows, and what it does not.

4.1. Logging

The interaction consists of "moves" taken by each participant, which are logged live during the interaction. For instance, in our sample scenario, we log:

- Gesture received by computer (i.e., made by human)
- Word received by computer (i.e., spoken by human)
- Gesture made by computer
- Action taken by computer
- Utterance made by computer

Each of these have VoxML semantics assigned to their content.
An interlocutor's understanding throughout the task can be inferred from the logged data by the time and number of moves taken to successfully communicate an instruction (e.g., successfully indicating a distinct object, successfully indicating an action to be taken, or any combination of the previous) or understanding of an instruction (e.g, acknowledging receipt of an instruction, asking a clarifying question, or executing an interpreted action). Longer time between steps indicates more time needed for the human to think (either for interpretation or planning), or for the avatar to generate an interpretation of the human's input, and more moves in the course of completing a single instruction indicates difficulty in communicating intended meaning.
Having identified some proxy measures for the respective understanding of the human and the computer in the interaction, the task then becomes quantifying how much the two accord.

4.2. Data collection

We ran preliminary trials on the gesture interaction system heretofore described, with 20 individual human subjects, all computer science graduate students without prior knowledge of the avatar's conceptual vocabulary. All subjects were presented with a scene with six uniquely-colored, identically-sized blocks and told to collaborate with the avatar to build a six-block, three-stepped staircase. They were told the computer was capable of understanding gestures and speech but were not given specific words or gestures to use. We then logged each interaction from start to finish, defined as the point at which the human and computer had created a three-stepped staircase to the satisfaction of the human user.

4.3. Post-processing

Consider the following output snippet from a sample trial. Beginning with the block setup in Figure 2, the human attempts to instruct the computer to select the purple block and slide it next to the yellow block. Moves are numbered, `G` indicates gestures, `S` indicates spoken words, `A` indicates actions taken, and `P` indicates a continuously-updated moving coordinate interpreted from the human user's pointing and situated in the virtual world, as indicated by the purple circle in Figure 2. Each update in continuous pointing is interpreted as a separate move and multiple updates are elided in the example below for space reasons. Prefixed `H` indicates a human move while prefixed `A` indicates an avatar move. The description and interpretation of each gesture is laid out in Krishnaswamy et al. (2017) and above in Section 2.1. Time format is given as seconds after trial start.

```
1 HG engage start 1.145281
2 AS "Hello." 1.145281
3 HP r,-0.25,-0.87 4.889832
...
73 HP r,-0.08,11.69 8.552608
74 HG right point high,-0.02,5.45
  8.588802
75 AS "Are you pointing here?" 8.588802
```

```
76 AG look_at(<0.03367592; 0.743948;
  1.559844>) 8.588802
77 AG point(rHand,<0.03367592; 0.743948;
  1.559844>) 8.588802
78 HP r,-0.02,5.45 8.660162
...
126 HP r,-0.14,-0.62 11.97283
127 HS NO 12.03008
128 AS "Sorry, I don't know what you
  mean." 12.03008
129 HP r,-0.10,-0.41 12.07262
...
188 HP r,0.96,-17.45 16.93849
189 HS PURPLE 16.97243
190 AS "OK, go on." 16.97243
191 AG reach(block7) 16.97243
192 HP r,1.06,-19.16 17.06419
193 HP r,-0.25,-0.39 17.32132
194 HP l,-0.20,1.66 18.01301
195 HP l,-0.20,1.66 18.13922
196 HP l,-0.20,1.66 18.20552
197 HG push left high 19.28887
198 HG push left stop 20.07251
199 AS "Do you want me to push this
  this way?" 20.07251
200 AG LARM_PUSH_RIGHT 20.07251
201 HG posack high 22.76486
202 AS "Yes?" 22.76486
203 AG RARM_THUMBS_UP 22.76486
204 AG HEAD_NOD 22.76486
205 HS YES 25.16582
206 AS "Should I push the purple block
  right of the yellow block?" 25.16582
207 HS YES 26.53876
208 AS "OK." 26.53876
209 AA slide(block7,left(block1))
  26.53876
```

Table 2: Sample trial log

As a proxy for the human's understanding of an avatar move, we simply take the time elapsed between the *first* in a block of avatar moves uninterrupted by a human move, and the human response that follows. The avatar's move can be gesture, action, utterance, or ensemble of multiple types, and multiple moves in a block often happen simultaneously (e.g., moves 199 and 200 here). Here, we can see that between moves 190 and 192, *0.092* seconds elapsed, whereas between 202 and 205, *2.401* seconds elapsed. These differences should reflect the clarity or expressiveness of the avatar's move. We call this the human's time to *begin response*. The avatar's utterance at 190 simply prompts the human to move to their next planned step, so the human can proceed without having to spend time interpreting it. The utterance at 202 is a question asking the human to confirm a move, which requires the human to process the preceding discourse and infer some of the computer's intent in order to respond properly, possibly accounting for the longer response time.

Often, the human may make (or the gesture recognition may see) gestures that, in the current context, the avatar has no interpretation for, and thus the human makes multiple moves before the avatar responds. These circumstances are also captured by measuring the time between the first in an uninterrupted block of moves by the human, and the first response by the avatar thereafter. Between steps 3 and 75, the human points around for *3.699* seconds before landing on a particular spot that the avatar asks to confirm. Later, *3.008* seconds pass between the human's moves beginning at 192 (responding to the avatar's utterance at 190) and the avatar's response to the subsequent content at 199, a length which may indicate difficulty moving the conversation forward. The system may be misinterpreting the gestures received or the human may be making gestures the system does not recognize. By contrast, when the human succeeds in producing contentful gestures or speech interpretable in context, the avatar is able to respond immediately as the input is processed (cf., moves 1-2, 201-202, 205-206) and move the conversation forward. We call this the avatar's time to *recognize content*. The human may have trouble communicating something contentful at the beginning, but by the time context is established through deixis and object disambiguation, the avatar is able to advance the interaction by providing a possible interpretation of the human's *push* instruction. These distinctions, if consistent across multiple trials, show areas where the communication between the interlocutors flows quickly or more slowly.

Response times may be charted against the semantic features of the moves that prompted the relevant response. As VoxML structures are componential, the distribution of response times can be plotted as a probability density over the magnitude function of preceding moves that contain a given semantic feature. For instance, the response time to a *push* instruction, where the fingers must be pointed forward (as in Figure 1), can be compared to response times to a *carry* instruction where the fingers must be curved. Response times can be divided with quantiles with a q selected for the desired granularity, and comparable moves should be those that occur in similar semantic contexts, that is, $[m_{j-n}..m_{j+n}]$ where m_j is the move in question, examined in a window of size $2n + 1$. Thus $\mathbf{P}(t_i|m_{j-n}..m_j..m_{j+n})$ represents the probability that a response time t falls in an interval i given a move and surrounding context. Individual moves can be replaced by a VoxML semantic feature of the move. Higher P for lower i indicates a higher likelihood of understanding being shared through the move or semantic feature at m_j.

5. Preliminary Results

Here we present a selection of the some of the most interesting and illustrative results drawn from the pilot user studies. In each chart, the X-axis shows the quantile in which reaction times fall relative to all reaction times per agent for any move, and the Y-axis shows the probability that the reaction time to the move in question falls within that quantile (on a 0%-100% scale). q = 5 in these plots.

Figure 3 shows the distribution of times taken for the avatar to recognize the verbal and gestural realizations of positive acknowledgment and negative acknowledgement, respectively. The distributions within modalities track each other roughly, but the avatar tends to take more time to recognize spoken "yes" than "no," which may be because the human takes more time to communicate a spoken positive acknowledgment than a negative one. Meanwhile, the

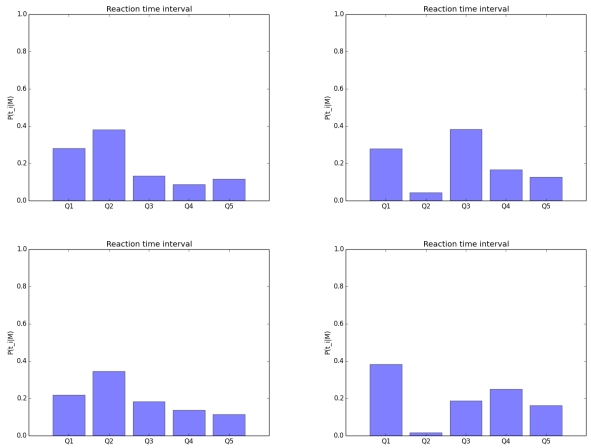


Figure 3: $\mathbf{P}(t_i|M)$, for avatar time to recognize positive/negative acknowledgment gesture (left top/bottom) vs. word "yes"/"no" (right top/bottom)

avatar appears to have a slightly quicker reaction time, in most cases, to positive acknowledgment through gesture than negative acknowledgment.

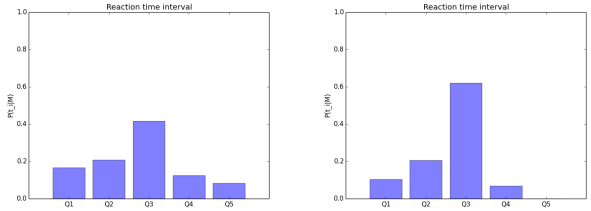


Figure 4: $\mathbf{P}(t_i|M)$ for human time to begin response to [[PUSH]] gesture (left) vs. [[CARRY]] gesture (right)

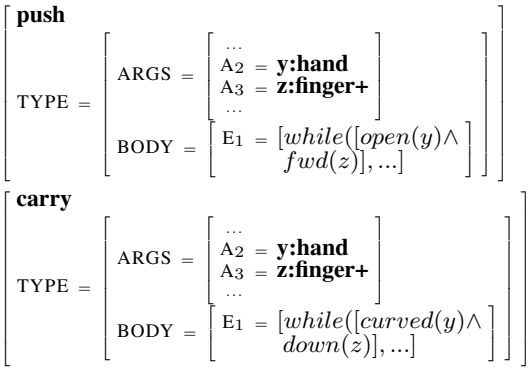


Figure 5: Abbreviated gesture voxeme type structures: [[PUSH]] vs. [[CARRY]]

Figure 4 shows the distribution of times taken for the human to respond to the avatar making a [[PUSH]] or [[CARRY]] gesture. The distributions are roughly equivalent, favoring the mid-range, but [[PUSH]] is almost twice as likely as [[CARRY]] to have a "very quick" (first interval) response time and middle-quintile response times are accordingly lower. The minimal distinction between [[PUSH]] and [[CARRY]] is the orientation and curvature of the hand and fingers, as shown in Figure 5, and so we can surmise that gestures with curved fingers might be harder for the human to interpret when compared to gestures with straight fingers (this evidences a conjecture made in Section 2.1.).

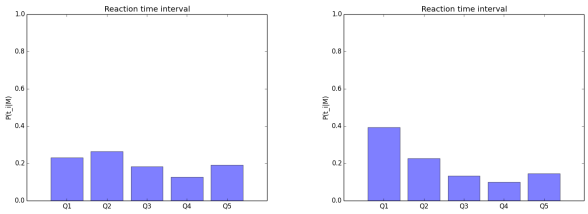


Figure 6: $\mathbf{P}(t_i|M)$ for avatar time to recognize [[POINT]] gesture with the left hand (left) vs. right hand (right)

Figure 6 shows the distribution of times taken for the avatar to recognize a pointing gesture with the left and right hand, respectively. The distributions roughly track each other, but the avatar is notably quicker to recognize right-hand pointing than left-hand pointing. Semantically, these gestures are the same, with only the orientation of the extended finger relative to the hand flipped. A possible explanation is that the gesture recognition displays greater variance in detecting the coordinates denoted by left-hand pointing, which should be accounted for in the recognition model.

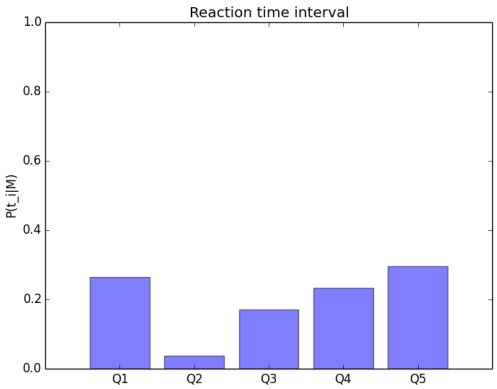


Figure 7: $\mathbf{P}(t_i|M)$ for human time to begin response to avatar asking a question

Figure 7 shows the distribution of times taken for the human to respond to the avatar's question. In most cases, the human answers quickly, or takes a long time to answer. This may be caused by the human not hearing or realizing that a question has been asked, and since the system does not repeat itself, some time can pass before the human realizes the move to make to move the conversation forward.
As developers, inferences like the above provide useful information for improving our example system.

6. Evaluation Variants

The evaluation scheme presented is deliberately high-level, in order to infer largely qualitative information from quantitative metrics. It represents a kind of minimal "base case" of conditions and parameters for evaluation of a multimodal human-computer system, many of which are easily varied to test different aspects of the interaction. As such, it is purposefully designed to be extensible to allow for different system types, interactive modalities, and scenarios.
Inference based on probabilistic distributions of course risks bias in the inferencing or due to missing information. Longer response times may occasionally be due to the trial subject getting distracted or some lag in the system rather

than any loss of understanding. Therefore we would like to present a set of variant and optional metrics to account for some of these potential omissions and test different aspects of the system under evaluation.

6.1. Additional and Optional Metrics

One simple additional parameter to log is flagging those utterances that clearly indicate confusion, such as the avatar saying "I don't understand," or the human saying "never mind," or abruptly disengaging mid-interaction. This can easily be evaluated to determine the probability of obvious confusion given a semantic context.

Similarly, affirmative and negative responses can be incorporated as a heuristic in determining when a block of moves should be initiated and terminated in evaluation. In Table 2, for instance, at move 75 the avatar asks the user if they are pointing at a particular location, and that question is answered with "No." That entire span in the interaction indicates communication of the pointing concept from the human to the avatar, but misinterpretation of the intent. If we were to treat that entire span as a block when calculating reaction time, rather than the raw human vs. avatar moves, we could glean additional information about the accuracy of pointing in the scenario.

6.2. Variant Conditions

In the outline above, we begin trials having given the user little to no information about the computer's vocabulary or capability. This allows evaluation to test the coverage of the system, but not necessarily how well the human adapts their behavior to the system. Therefore trials might also be conducted after the user has been presented with written descriptions of possible gestures (as in Section 2.1.), or after watching a video recorded from a successfully completed interaction, which would allow them to see gestures and known vocabulary in use. Both these conditions could allow evaluation of how well the interaction functions within the known constraints on the vocabulary, and can test if training the user reduces error.

6.3. User Feedback and "Ground Truth"

User reaction to the trial may be assessed by surveying them immediately following the interaction. For instance, they might rate certain moves by the avatar from clear to confusing, giving a qualitative rating of specific circumstances. Additionally, users might provide live feedback during the trial by a "talkback mode," wherein they could provide inputs such as the following:

- What user expects avatar to do following their move
- Whether an avatar question is reasonable or not
- Whether an avatar response is situationally inappropriate, incomplete, redundant, etc.

How can the overall success or failure of an interaction be assessed from the perspective of the computer system? At the beginning of the interaction, the target pattern might be fed into a planner that determines an optimal solution of moves to make on the part of both parties in order to build the target pattern (i.e., a ground truth solution), and at the completion of the interaction, the actually executed interaction is compared to the optimal solution. This can be assessed using simple metrics, like edit distance.

6.4. Scenario Variants and User Modeling

Blocks world tasks can serve as a proxy for situations requiring a collaborative interaction in a controlled environment, but do risk missing information about what a user knows about other types of complex objects versus what a computer knows. Introducing non-block objects adds other parameters that can be conditioned against, such as how the interlocutors interpret each others' behavior with convex objects vs. non-convex objects, or round objects vs. flat ones. Would a human, for example, more readily ask an agent to roll a ball than roll a block, due to knowledge that balls *afford* being rolled (Gibson, 1977; Gibson, 1979)? This information can then be incorporated into the agent's model of what its interlocutor knows about the vocabulary of available concepts. As the virtual agent becomes more and more certain that the human knows certain concepts or prefers certain moves, it may more readily execute them, or could even plan for expected user behavior.

7. Conclusion

We have proposed an evaluation scheme to assess the coverage of multimodal interaction systems and provided an outline of its use evaluating a sample interaction in a system that uses linguistic, gestural, and visual modalities. The example system exploits many advantages of virtual embodiment (Kiela et al., 2016), but consistent evaluation is required to test areas where the system needs improvement, and the framework outlined above can provide this information without very complicated algorithms to process the logged data. It uses simple metrics and processing based on object and event semantics. These properties are agnostic to the precise modalities used in a given interaction, and so the evaluation scheme accommodates measurement of various phenomena through the course of a human-computer interaction in a multimodal system. We have presented preliminary results from naive users run through the sample system, which show how we can use simple metrics to assess the ease or difficulty with which specific features communicate information. We believe this type of evaluation will be useful for developing user models and helping researchers assess the gaps in novel computational interaction systems in a variety of modalities, scenarios, and interaction types.

8. Acknowledgements

The authors would like to thank the reviewers for their helpful comments. We would also like to thank Robyn Kozierok, Brad Goodman, Lynette Hirschman, and their colleagues at the MITRE Corporation for developing the "hallmarks of communication" rubric. We would like to thank our colleagues at Colorado State University and the University of Florida for developing the gesture recognition systems: Prof. Bruce Draper, Prof. Jaime Ruiz, Prof. Ross Beveridge, Pradyumna Narayana, Isaac Wang, Rahul Bangar, Dhruva Patil, Gururaj Mulay, Jason Yu, and Jesse Smith; and our Brandeis University colleagues, Tuan Do and Kyeongmin Rim, for their work on VoxSim. This work is supported by a contract with the US Defense Advanced

Research Projects Agency (DARPA), Contract W911NF-15-C-0238. Approved for Public Release, Distribution Unlimited. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

9. Bibliographical References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). TensorFlow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI). Savannah, Georgia, USA*.

Andrist, S., Gleicher, M., and Mutlu, B. (2017). Looking Coordinated: Bidirectional Gaze Mechanisms for Collaborative Interaction with Virtual Characters. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 2571–2582, New York, NY, USA. ACM.

Arbib, M. and Rizzolatti, G. (1996). Neural expectations: A possible evolutionary path from manual skills to language. *Communication and Cognition*, 29:393–424.

Arbib, M. A. (2008). From grasp to language: embodied concepts and the challenge of abstraction. *Journal of Physiology-Paris*, 102(1):4–20.

Asher, N. and Gillies, A. (2003). Common ground, corrections, and coordination. *Argumentation*, 17(4):481–512.

Asher, N. and Lascarides, A. (2003). *Logics of conversation*. Cambridge University Press.

Bolt, R. A. (1980). *"Put-that-there": Voice and gesture at the graphics interface*, volume 14. ACM.

Brennan, S. E., Chen, X., Dickinson, C. A., Neider, M. B., and Zelinsky, G. J. (2008). Coordinating cognition: The costs and benefits of shared gaze during collaborative search. *Cognition*, 106(3):1465–1477, March.

Cassell, J., Stone, M., and Yan, H. (2000). Coordination and context-dependence in the generation of embodied conversation. In *Proceedings of the first international conference on Natural language generation-Volume 14*, pages 171–178. Association for Computational Linguistics.

Cassell, J. (2000). *Embodied conversational agents*. MIT press.

Clair, A. S., Mead, R., Matarić, M. J., et al. (2010). Monitoring and guiding user attention and intention in human-robot interaction. In *ICRA-ICAIR Workshop, Anchorage, AK, USA*, volume 1025.

Clark, H. H. and Brennan, S. E. (1991). Grounding in communication. In Lauren Resnick, et al., editors, *Perspectives on Socially Shared Cognition*, pages 13–1991. American Psychological Association.

Clark, H. H. and Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, 22(1):1–39, February.

Thierry Declerck, et al., editors. (2010). *Semantic Multimedia: 5th International Conference on Semantic and Digital Media Technologies, SAMT 2010. Saarbrücken, Germany, December 1-3, 2010. Revised Selected Papers*.

Dillenbourg, P. and Traum, D. (2006). Sharing solutions: Persistence and grounding in multimodal collaborative problem solving. *The Journal of the Learning Sciences*, 15(1):121–151.

Dumas, B., Lalanne, D., and Oviatt, S. (2009). Multimodal interfaces: A survey of principles, models and frameworks. *Human machine interaction*, pages 3–26.

Eisenstein, J., Barzilay, R., and Davis, R. (2008a). Discourse topic and gestural form. In *AAAI*, pages 836–841.

Eisenstein, J., Barzilay, R., and Davis, R. (2008b). Gesture salience as a hidden variable for coreference resolution and keyframe extraction. *Journal of Artificial Intelligence Research*, 31:353–398.

Fotinea, S.-E., Efthimiou, E., Koutsombogera, M., Dimou, A.-L., Goulas, T., and Vasilaki, K. (2016). Multimodal resources for human-robot communication modelling. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May. European Language Resources Association (ELRA).

Fussell, S. R., Kraut, R. E., and Siegel, J. (2000). Coordination of Communication: Effects of Shared Visual Context on Collaborative Work. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, CSCW '00, pages 21–30, New York, NY, USA. ACM.

Fussell, S. R., Setlock, L. D., Yang, J., Ou, J., Mauer, E., and Kramer, A. D. I. (2004). Gestures over Video Streams to Support Remote Collaboration on Physical Tasks. *Hum.-Comput. Interact.*, 19(3):273–309, September.

Gebre, B. G., Wittenburg, P., and Lenkiewicz, P. (2012). Towards automatic gesture stroke detection. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

Gergle, D., Kraut, R. E., and Fussell, S. R. (2004). Action As Language in a Shared Visual Space. In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, CSCW '04, pages 487–496, New York, NY, USA. ACM.

Gibson, J. J. (1977). The theory of affordances. *Perceiving, Acting, and Knowing: Toward an ecological psychology*, pages 67–82.

Gibson, J. J. (1979). *The Ecology Approach to Visual Perception: Classic Edition*. Psychology Press.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.

Hobbs, J. R. and Evans, D. A. (1980). Conversation as planned behavior. *Cognitive Science*, 4(4):349–377.

Hough, J. and Schlangen, D. (2016). Investigating fluidity for human-robot interaction with real-time, real-world grounding strategies.

Jaimes, A. and Sebe, N. (2007). Multimodal human–computer interaction: A survey. *Computer vision and image understanding*, 108(1):116–134.

Johnston, M. (2009). Building multimodal applications with EMMA. In *Proceedings of the 2009 interna-*

tional conference on Multimodal interfaces, pages 47–54. ACM.

Kennington, C., Kousidis, S., and Schlangen, D. (2013). Interpreting situated dialogue utterances: an update model that uses speech, gaze, and gesture information. *Proceedings of SIGdial 2013*.

Kiela, D., Bulat, L., Vero, A. L., and Clark, S. (2016). Virtual embodiment: A scalable long-term strategy for artificial intelligence research. *arXiv preprint arXiv:1610.07432*.

Krämer, N. C., Simons, N., and Kopp, S. (2007). The effects of an embodied conversational agent's nonverbal behavior on user's evaluation and behavioral mimicry. In *International Workshop on Intelligent Virtual Agents*, pages 238–251. Springer.

Kraut, R. E., Fussell, S. R., and Siegel, J. (2003). Visual Information As a Conversational Resource in Collaborative Physical Tasks. *Hum.-Comput. Interact.*, 18(1):13–49, June.

Krishnaswamy, N. and Pustejovsky, J. (2016a). Multimodal semantic simulations of linguistically underspecified motion events. In *Spatial Cognition X: International Conference on Spatial Cognition*. Springer.

Krishnaswamy, N. and Pustejovsky, J. (2016b). VoxSim: A visual platform for modeling motion language. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. ACL.

Krishnaswamy, N., Narayana, P., Wang, I., Rim, K., Bangar, R., Patil, D., Mulay, G., Ruiz, J., Beveridge, R., Draper, B., and Pustejovsky, J. (2017). Communicating and acting: Understanding gesture in simulation semantics. In *12th International Workshop on Computational Semantics*.

Lascarides, A. and Stone, M. (2006). Formal semantics for iconic gesture. In *Proceedings of the 10th Workshop on the Semantics and Pragmatics of Dialogue (BRANDIAL)*, pages 64–71.

Lascarides, A. and Stone, M. (2009a). Discourse coherence and gesture interpretation. *Gesture*, 9(2):147–180.

Lascarides, A. and Stone, M. (2009b). A formal semantic analysis of gesture. *Journal of Semantics*, page ffp004.

Lemmelä, S., Vetek, A., Mäkelä, K., and Trendafilov, D. (2008). Designing and evaluating multimodal interaction for mobile contexts. In *Proceedings of the 10th international conference on Multimodal interfaces*, pages 265–272. ACM.

Ligozat, G. F. (1993). Qualitative triangulation for spatial reasoning. In *European Conference on Spatial Information Theory*, pages 54–68. Springer.

Madeo, R. C. B., Peres, S. M., and de Moraes Lima, C. A. (2016). Gesture phase segmentation using support vector machines. *Expert Systems with Applications*, 56:100–115.

Matuszek, C., Bo, L., Zettlemoyer, L., and Fox, D. (2014). Learning from unscripted deictic gesture and language for human-robot interactions. In *AAAI*, pages 2556–2563.

Mehlmann, G., Häring, M., Janowski, K., Baur, T., Gebhard, P., and André, E. (2014). Exploring a Model of Gaze for Grounding in Multimodal HRI. In *Proceedings of the 16th International Conference on Multimodal Interaction*, ICMI '14, pages 247–254, New York, NY, USA. ACM.

Oviatt, S. (2003). User-centered modeling and evaluation of multimodal interfaces. *Proceedings of the IEEE*, 91(9):1457–1468.

Pustejovsky, J. and Krishnaswamy, N. (2016). VoxML: A visualization modeling language. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May. European Language Resources Association (ELRA).

Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K. E., and Ansari, R. (2002). Multimodal human discourse: gesture and speech. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 9(3):171–193.

Rieser, H. and Poesio, M. (2009). Interactive gesture in dialogue: a PTT Model.

Skantze, G., Hjalmarsson, A., and Oertel, C. (2014). Turn-taking, feedback and joint attention in situated human-robot interaction. *Speech Communication*, 65:50–66, November.

Turk, M. (2014). Multimodal interaction: A review. *Pattern Recognition Letters*, 36:189–195.

Wang, I., Narayana, P., Patil, D., Mulay, G., Bangar, R., Draper, B., Beveridge, R., and Ruiz, J. (2017a). EGGNOG: A continuous, multi-modal data set of naturally occurring gestures with ground truth labels. In *To appear in the Proceedings of the 12th IEEE International Conference on Automatic Face & Gesture Recognition*.

Wang, I., Narayana, P., Patil, D., Mulay, G., Bangar, R., Draper, B., Beveridge, R., and Ruiz, J. (2017b). Exploring the use of gesture in collaborative tasks. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, pages 2990–2997, New York, NY, USA. ACM.

Wooldridge, M. and Lomuscio, A. (1999). Reasoning about visibility, perception, and knowledge. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 1–12. Springer.

Zarrieß, S. and Schlangen, D. (2017). Deriving continous grounded meaning representations from referentially structured multimodal contexts. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 970–976.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE MulitMedia*, 19:4–10.

Ziemke, T. and Sharkey, N. E. (2001). A stroll through the worlds of robots and animals: Applying jakob von uexkull's theory of meaning to adaptive robots and artificial life. *SEMIOTICA-LA HAYE THEN BERLIN-*, 134(1/4):701–746.

Zimmermann, K. and Freksa, C. (1996). Qualitative spatial reasoning using orientation, distance, and path knowledge. *Applied intelligence*, 6(1):49–58.

The WAW Corpus: The First Corpus of Interpreted Speeches and their Translations for English and Arabic

Ahmed Abdelali, Irina Temnikova, Samy Hedaya, Stephan Vogel
Qatar Computing Research Institute & Translation and Interpreting Institute
Hamad Bin Khalifa University, Doha, Qatar
{aabdelali, itemnikova, shedaya, svogel}@hbku.edu.qa

Abstract

This article presents the WAW Corpus, an interpreting corpus for English/Arabic, which can be used for teaching interpreters, studying the characteristics of interpreters' work, as well as to train machine translation systems. The corpus contains recordings of lectures and speeches from international conferences, their interpretations, the transcripts of the original speeches and of their interpretations, as well as human translations of both kinds of transcripts into the opposite language of the language pair. The article presents the corpus curation, statistics, assessment, as well as a case study of the corpus use.

Keywords: corpus, interpreting strategies, annotation, translation, Arabic

1. Introduction

The Arabic Language Technologies research group at the Qatar Computing Research Institute[1] is developing Natural Language Processing (NLP) tools to support Arabic, including an Automatic Speech Recognition (ASR) system (Khurana and Ali, 2016), and a Speech-to-Text (S2T) Machine Translation (MT) system (Dalvi et al., 2017). Despite the advances in automatic MT, such systems still lag behind interpreters in accuracy and fluency. One of the planned applications of our S2T translation system is to support the educational domain and provide translations of lectures. As simultaneous interpretation is different from translation, we wanted to better understand the strategies interpreters apply to hopefully implement some of these strategies to improve our system. A first step for this was to build a corpus of interpreted lectures, which can be used to analyze how interpreters deal with the challenges in simultaneous interpretation. Although a number of Arabic corpora exist (Zaghouani, 2017; Wray and Ali, 2015; Ali et al., 2016a; Ali et al., 2016b; Ali et al., 2017), to the extent of our knowledge there are no publicly available interpreting corpora for Arabic. We therefore collected a corpus of lectures by recording talks given at conferences, together with their interpretations done by professional interpreters. After giving a short survey of the related work, we present the WAW Corpus and details about the collection and curation process. Next, we provide a quantitative and qualitative assessment of the collected data. We also present a pilot/case study of the use of the corpus for extracting interpretation strategies used by interpreters.

2. Related Work

In (Al-Khanji et al., 2000) interpreting strategies in Arabic have been studied, but no re-usable corpus was released. There are also Arabic speech corpora used for MT[2] (Kumar et al., 2014; Zaidan and Callison-Burch, 2014), but they do not include human interpretation of the original speech (only translated speech transcripts are provided). We are not aware of any other publicly available interpreting corpora for Arabic, whereas they exist for Italian, Spanish, English, French, Dutch (Bendazzoli and Sandrelli, 2005; Falbo, 2012), Brazilian Portuguese and German (House et al., 2012), Japanese and Chinese (Tohyama and Matsubara, 2006; Hu and Qing, 2009). Differing from the existing Arabic speech corpora, the WAW corpus contains recordings of the original speakers, the recordings of the interpreters, the transcripts of both recordings, and the translations of all transcripts.

3. WAW Corpus

The WAW corpus comprises recordings from three international conferences, which took place in Qatar: **W**ISE 2013 (World Innovation Summit for Education)[3], **A**RC'14 (Qatar Foundation's Annual Research and Development Conference[4], and **W**ISH 2013 (World Innovation Summit for Health)[5]. The speeches and discussions were mostly in English, some in Arabic.

Professional interpreters were hired by the event organizers to provide simultaneous interpretation from English into Modern Standard Arabic or vice-versa. To the best of our knowledge the interpreters were all native speakers of Arabic, and most talks were interpreted from English into Arabic, so into the native language of the interpreter. As all interpreters were high-level professionals, their level of English was at least at an advanced level. All speakers and interpreters signed a release form to transfer the ownership of their talks to the conferences organizers (Qatar Foundation) and to give permission for their speech to be recorded and used for scientific purposes.

Table 1 shows the corpus' topics composition with number of files per conference, classified per broad areas (for ARC'14, which is a wide interdisciplinary conference) and topics (for WISE 2013 and WISH 2013, as they are already specialized in the broad areas of Education and Health).

[1] http://www.qcri.org
[2] https://catalog.ldc.upenn.edu/LDC97T19.
[3] http://www.wise-qatar.org
[4] http://www.qf-arc.org
[5] http://www.wish-qatar.org

Conference	Number of files
ARC'14	71
Topics: Energy and Environment(31), Computing and Information Technology(21), Social Sciences, Arts and Humanities(12), Health(7)	
WISE 2013	33
Topics: General(16), Massive Open Online Courses(5), World-Wide Education(4), Web Literacy(4), Education Without Teachers(2), Primary School Teachers(2)	
WISH 2013	22
Topics: Big Data(4), Health and Ethics: End of Life(4), Road Traffic Injury(3), Patient Engagement(3), Innovation of Health Care(3), General(3), Mental health(2)	

Table 1: WAW Corpus number of files per conferences and topics.

3.1. Original Recordings

The original audio files where recorded with two audio channels captured from the audio stream – Arabic and English – using Zoom H4n Handy Recorders hooked into the audio systems of the conference room. The recordings were stored using the WAV format with 24-bit encoding and a sampling rate of 48khz. The stereo recordings were then split per language and per session. Each session presents a speech, a lecture, or a discussion. For consistency, questions and answers segments were separated from the original lecture as an independent session. This resulted in a total of 521 sessions with an average length of 13.56 min, and a total of 119 hours. A subset of 252 sessions (62h52m) among the fully parallel (i.e recordings were completed for both languages with English speaker and Arabic interpretation) have been used for this study. A total of 12 interpreters delivered the live interpretation in these events.

3.2. Audio Transcription

Once the audio collection was completed and separated by session and language, we developed guidelines for transcription to be carried by a professional agency. The guidelines include[6]. :

- Tagging Named Entities (Persons, Locations, Organizations)
- Capturing Non-lexical noises, i.e. Breathing, Laugh, Applause, Music.
- Capturing speech acts such as Repetitions, Hesitations, Interjections, False Starts and Corrections.
- How to limit segment size.

Figure 1 shows excerpts from the transcripts of an English audio and its Arabic interpretation with various annotations for NEs, speech acts and timing information.
The aim of the additional annotation was to capture the maximum features that allow to exploit the verbatim transcriptions for automatic recognition and processing of speech – for example, speaker information can be used for speech diarisation – while the resulting parallel data can be used in building automatic machine translation.

```
514
00:31:56,266 --> 00:31:58,862
well so that is the way I think you know [NE:PER Danny Weitzner]
515
00:31:58,862 --> 00:32:01,986
[REP who is] [HES] [NOISE] is was [REP in] [FALSE DC]
[REP in] in [NE:LOC Washington DC] for a while
516
00:32:01,986 --> 00:32:05,072
[HES] [FALSE for the] working [FALSE with the] [REP for the]
for the government.
517
00:32:05,072 --> 00:32:06,605
and he is back at [NE:ORG CSAIL] now.
518
00:32:06,605 --> 00:32:09,785
he is a lawyer [NOISE] [REP and] and a principal investigator
at [NE:ORG CSAIL].
------------------------------
160
00:31:59,489 --> 00:32:12,985
[تردد] [علم: شخص داني وايتتر] [تردد] موجود في [علم: مكان وشنطن دي سي] بالنسبة
[تردد] والآن عاد إلى [علم: مكان سيئل] أو [علم: مكان سياتل] [تردد] وهو محامي.
```

Figure 1: Speaker's (English) and corresponding interpreter's (Arabic) original transcripts with various annotations and time information.

3.3. Translation of Transcripts

The transcripts of both the original audio and the interpretations were translated into Arabic/English in accordance to which language the original speech was given in.
The goal for transcripts translation was to provide ground for comparison between the original text and the output of the interpreter. Because of this, the transcripts were translated to be able to perform experiments comparing translation features with interpretation features. Another reason was that we wanted to be able to run comparison of the original speaker's transcript and the interpreter in the same language. This could be achieved cross-lingually but the ideal scenario would be to have the material in the same language.
The translations were done by a professional translation agency[7]. The translation was guided by rules to ensure the quality and the style. Some of the translation rules include:

- The translation should be faithful to the original text in terms of meaning, cultural assumptions, and style, while preserving grammaticality, fluency, and naturalness.
- The translator is expected to maintain the same speaking style or register as the source. For example, if the source is polite, the translation should maintain the politeness. If the source is excited or angry, the translation should convey the same tone.

[6]Complete transcription guidelines are available at http://alt.qcri.org/resources/wawcorpus

[7]Transcription and translation were done by different professional agencies.

- The translation is expected to contain the exact meaning conveyed in the source text, and it should neither add, nor delete information. For instance, if the original text uses "Bush" to refer to the former US President, the translation should not be rendered as "President Bush, George W. Bush", etc.

The structure of the resulting corpus is shown in Figure 2, where for each speech and its corresponding interpretation, there are transcriptions and translations of the transcriptions into the other language.
The corpus contains additional meta data, including the language of the original speech and its interpretation, information regarding which interpreter interpreted which speech, the length (in seconds and in words) of the speech and of its interpretation[8].

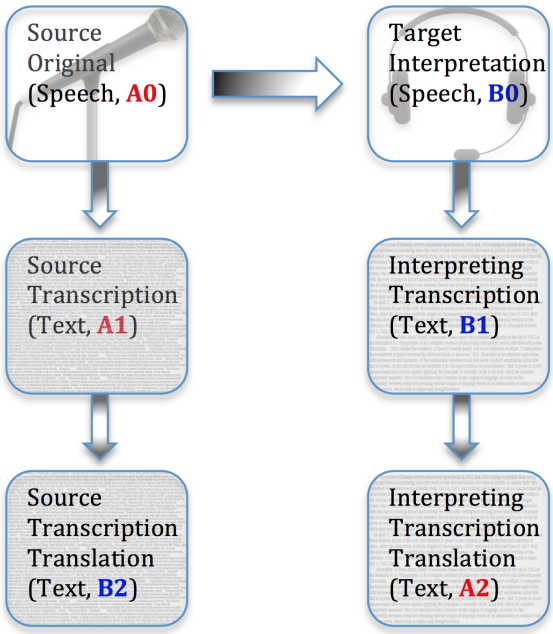


Figure 2: WAW Corpus structure. For each audio and its corresponding interpretation (A0, B0) there are transcripts (A1, B1) and translations (B2, A2).

4. Corpus Content Assessment

The transcriptions of the original audios and their respective interpretations were carried out independently. This resulted in *differences in time segmentation* and different number and length of segments in the original speakers' transcripts versus the interpreters' ones. In Figure 1 we see an example where the English part has been split by the transcriber into 5 short segments whereas the transcriber of the Arabic recording combined it into 1 segment.
We can also see a *difference in the number of speaker's words versus the ones generated by the interpreter*. This difference is due to the *language pair* and to the *strategies applied by the interpreter* (see Sections 4.1. and 5.1.). A detailed summary of corpus statistics is provided in Table 2.

[8]http://alt.qcri.org/resources/wawcorpus/releases/WAW-Readme.txt.

	English	Arabic
N. of files	126	126
Total Time	31:26:24	31:26:24
N. of Segments	26,572	9,413
N. of Words	286,024	156,814
N. of Words Translation	222,025	194,927

Table 2: WAW Corpus size in total number of files, recorded time, number of lines/segments, and number of words.

The simultaneous interpreter is *usually lagging a few seconds behind the speaker*. In the example in Figure 1 the interpreter started with 3 seconds delay and also ended with a delay of approximately 3 seconds. This décalage has been investigated in a number of studies, e.g. (Kroll and De Groot, 2009).

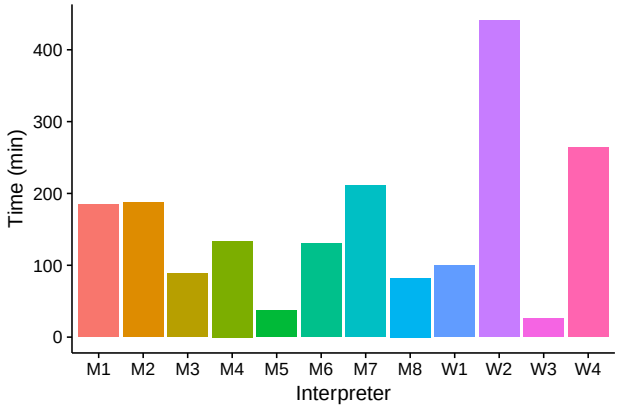


Figure 3: Interpreters' total time in the corpus (Mx: Male, Wx: Female)

In the following, we present an overall inspection of the contents of the corpus, in order to get an idea of the quality and the nature of its data. To achieve that, we used two measurements: a *lexical measurement of token ratios* and a *semantically oriented measurement of the distribution of "Named Entity" (NE) tags* used in the corpus and compared them cross-linguistically between the speaker and the interpreter.

4.1. Token Ratios

Arabic is an agglutinative language, in which several morphemes get fused together to compose a single Arabic word. As noted in corpora studies, the ratio between Arabic to English in typical written text is around 0.7 (Salameh et al., 2011). This is what we do also observe in the WAW corpus when we compare the number of tokens of the Arabic translations (B2) compared with the transcript of the original speaker's speech (A1, in English). However, when we compare the number of word tokens in the transcripts of the interpreters (B1) with the number of tokens in the translations of these transcripts (A2), we see a much smaller ratio. Figure 4 shows these ratios across documents grouped by interpreters. On average less text is produced by the interpreters compared to the original speakers and also com-

pared to the translators. In addition, we observe a wider distribution of ratios for the interpretations than for the translations. This reflects the challenging aspect of the interpretation work and the cognitive load the interpreter has to deal with while carrying the task of interpretation.

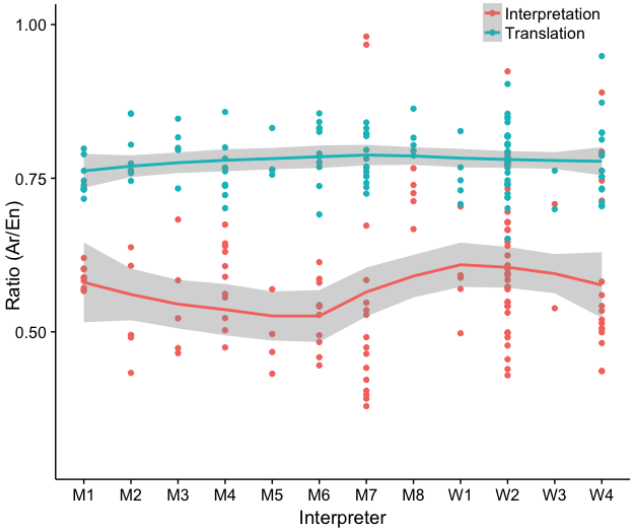


Figure 4: Comparison of the interpretation/original speaker and translation/original speaker ratios.

4.2. Distribution of Annotated Named Entities

Measuring the interpretation accuracy is not a straightforward task (Kahane, 2000; Zwischenberger and Pöchhacker, 2010). To measure it, we selected the "Named Entities" tag as an indicator or a feature to compare them cross transcripts. The expectation is that in the ideal scenario the interpretation should contain the same (or most of the) named entities that are mentioned in the original transcript. We realize that this is not a precise evaluation as the interpreter or the speaker could use pronominal or other types of references, which might not be captured while annotating the data. This approach nonetheless could still be helpful in the initial assessment. Figure 5 shows the distribution of the NE tag ratios between the original and interpreter transcripts. Most of the medians are around 0.8, which indicates that the interpreters were able to reproduce 80% of the NEs mentioned by the speaker. Interpreter M2 is a clear outlier, generating on average only 30% of the named entities. It will require more in-depth analysis to see if this is the result of a specific strategy, e.g. substituting names by pronouns, or just the indication of a sub-par interpretation.

5. Case Study: Interpreting Strategies Annotation

As a use case of the WAW interpretation corpus, we present a study of manually annotating the interpreting strategies in the corpus. The aim of this study was to reveal which strategies interpreters from English into Arabic use, and how often. The hope is that this might eventually provide some indications if our speech-to-text automatic translation system (Dalvi et al., 2017) could benefit from implementing some of these human interpreting strategies.

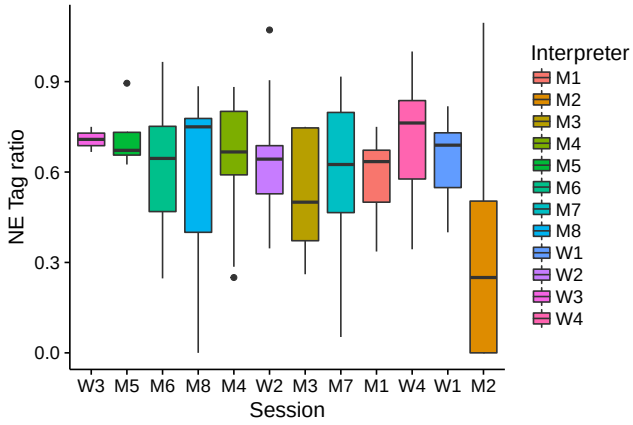


Figure 5: Ratios of NEs between the interpretation transcripts and the original speaker's transcripts (Mx: Male, Wx: Female)

The study was also motivated by our previous observation on the differences in the word ratios between interpretations and translations and the discrepancy in number of named entities tags.

This study is a follow-up of (Temnikova et al., 2017), where we conducted a preliminary annotation of the WAW corpus for interpreting strategies by analyzing a small sample of 7500 words (English+Arabic) from the transcripts of 4 sessions, with 2 female interpreters, including W2 (the most productive interpreter in our corpus), and 2 talks for each of these interpreters. For the study reported in the current paper we expanded the sample to 8 sessions, done by 4 interpreters, 2 women (W2, W4) and 2 men (M1, M7), adding up to 16,955 words in English and 9,477 words in Arabic. We selected these specific interpreters based on the transcript ratios against the original speakers, picking one with a high ratio and one with a low ratio for both male and female interpreters. As something new, added to our previous preliminary study, we also combined female and male interpreters, in order to further investigate whether there is any gender difference.

Finally, we also revised our annotation guidelines by adding more categories. A professional translator – native speaker of Arabic and fluent in English – with expertise in translation strategies annotation, was recruited to carry these annotation tasks.

5.1. Interpreting Strategies Annotation Guidelines

Our annotation guidelines are based on the state-of-the-art work on interpreting strategies (Roderick, 2002; Shlesinger, 1998; Bartłomiejczyk, 2006; Kalina, 1998; Al-Khanji et al., 2000), and on practical observations of the WAW corpus (Temnikova et al., 2017). Our updated annotation guidelines are available online[9].

Our annotation categories can be grouped into five major groups (as defined by our annotator):

1. **Summarizing**: The interpreter combines two clauses

[9] http://alt.qcri.org/resources/wawcorpus/releases/WAWCorpusSegmentationandAlignmentGuidelines.docx

into one or summarizes a single but longer clause.

2. **Self-correction**: The interpreter usually uses "أو" (or) or repetition to alter a lexical choice or correct a mispronounced word. Also, the interpreter may correct or modify the part of speech.

3. **Omissions**: The interpreter omits words which were in the source language text. We have a further subdivision into 3 categories of omissions (further information can be consulted in our guidelines).

4. **Additions**: The interpreter adds additional information, which is not found in the source text. We have 5 varieties of additions, which can be seen in our guidelines.

5. **Transliterations**: The interpreter uses the source language word in the target language. E.g. using "الموكس" for MOOCS.

For these categories, the expert annotator was asked to both provide a classification category of an interpreting strategy, and to provide an evaluation whether this "change" was *acceptable* or *not acceptable* (in order to further identify interpreters' errors versus successful strategies). (For more details and examples, please see our annotation guidelines.)

5.2. Analysis of the Results

The analysis of the results is based on these five aforementioned categories (see Figure 6). As it can be seen in the figure, corroborating the results of our previous study, the omissions are the highest number of interpretation strategies used, followed by additions. The ratio of omissions correlates with the Arabic to English word ratios with a coefficient of 0.89%. This is typically a strong correlation, but the other strategies show loose correlation with the Arabic to English word ratios (the coefficient varies between -0.12% and 0.38%).

On the other hand, we know that the different strategies affect the text of the transcript in a different way. "Additions" expand the text by adding newer content that was not in the original version of the text. This impacts negatively the ratio. In our scenario for interpreting from English into Arabic, the ratio "English/Arabic" tokens gets smaller when the denominator gets bigger (when more text is added to the Arabic interpretation). For this reason, we thought of looking at the strategies in combination. The different categories of strategies impact on the resulting transcript as a whole rather than individual. The new formalism we obtained was:

$$Combined = Addition - Omission + Self\ Correction - Summarizing \quad (1)$$

The new combined metric in Eq. 1 modeled better the impact on text that the various strategies employed by the interpreters, had. The correlation ratio between *Combined* and *Ratio (En/Ar)* is *0.92%*. This better explains the drop in the text ratio and how closely it is related to the number of strategies employed.

From another point of view, we did not notice any gender-related differences (at least for this small sample).

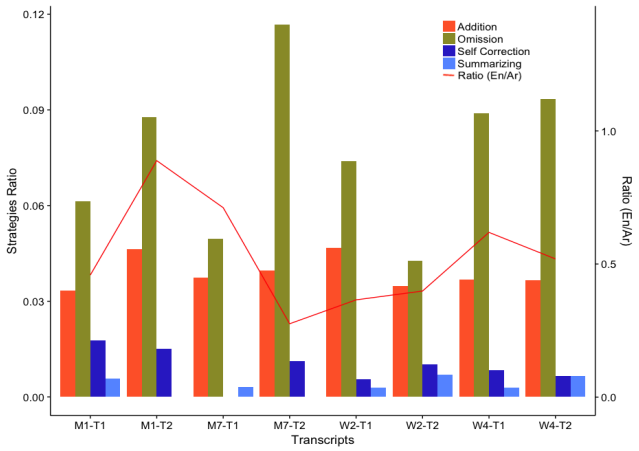


Figure 6: Distribution of Strategies normalized by the transcript length for each interpreter versus the ratio of Arabic tokens over English.

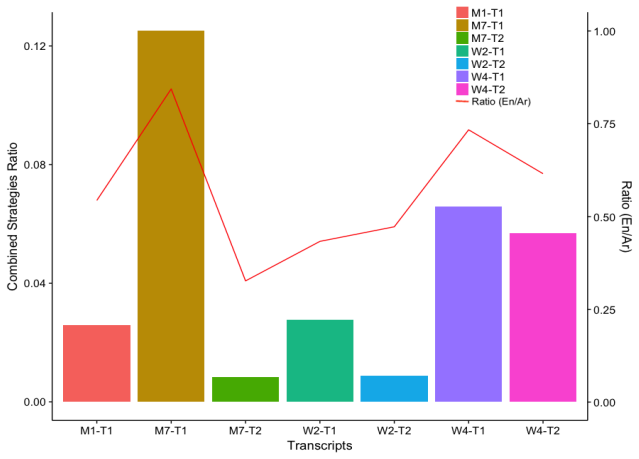


Figure 7: Combined Strategies normalized by the transcript length versus the ratio of Arabic over English.

6. Conclusions

This article presents our (a first) corpus of conference speeches, interpreted, transcribed, and translated, for the English-Arabic language pair. We provide some statistics and assessment of the corpus, as well as a case study involving interpreting strategies annotation by an expert. The findings from the annotations explain the differences between translated and interpreted material. The amount of omitted and summarized material skew the normal ratio between original documents and their translation. We would like to further exploit these findings and employ them for Machine Translation and S2T system where time is more critical. As future work, we plan to continue collecting more annotations at this level of granularity (omissions, additions, etc.), as well as to deepen the analysis into more detailed strategies. Further, we plan to automatize the annotation at the omissions and additions level, as well as to train our in-house speech-to-text machine translation system (Dalvi et al., 2017) on the basis of our findings.

References

Al-Khanji, R., El-Shiyab, S., and Hussein, R. (2000). On the use of compensatory strategies in simultaneous interpretation. *Meta: Journal des traducteurs/Meta: Translators' Journal*, 45(3):548–557.

Ali, A., Bell, P., Glass, J., Messaoui, Y., Mubarak, H., Renals, S., and Zhang, Y. (2016a). The MGB-2 Challenge: Arabic multi-dialect broadcast media recognition. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*.

Ali, A., Dehak, N., Cardinal, P., Khurana, S., Yella, S. H., Glass, J., Bell, P., and Renals, S. (2016b). Automatic dialect detection in Arabic broadcast speech. In *Interspeech, San Francisco, USA*, pages 2934–2938.

Ali, A., Vogel, S., and Renals, S. (2017). Speech recognition challenge in the wild: Arabic MGB-3. In *Automatic Speech Recognition and Understanding (ASRU), 2017 IEEE Workshop on*. IEEE.

Bartłomiejczyk, M. (2006). Strategies of simultaneous interpreting and directionality. *Interpreting*, 8(2):149–174.

Bendazzoli, C. and Sandrelli, A. (2005). An approach to corpus-based interpreting studies: Developing EPIC (european parliament interpreting corpus). In *MuTra 2005–Challenges of Multidimensional Translation: Conference Proceedings*, pages 1–12.

Dalvi, F., Zhang, Y., Khurana, S., Durrani, N., Sajjad, H., Abdelali, A., Mubarak, H., Ali, A., and Vogel, S. (2017). QCRI live speech translation system. *EACL 2017*, page 61.

Falbo, C. (2012). CorIT (Italian Television Interpreting Corpus): Classification criteria. *Breaking Ground in Corpus-based Interpreting Studies, Bern et al., Peter Lang*, pages 157–185.

House, J., Meyer, B., and Schmidt, T. (2012). CoSi-A corpus of consecutive and simultaneous interpreting. *Multilingual corpora and multilingual corpus analysis*, 14:295.

Hu, K. and Qing, T. (2009). Hanying huiyi kouyi zhong yupian yiyi xianhjua jiqi dongyin yanjiu–yixiang jiyu pingxing yuliaoku de yanjiu [A corpus-based study of explicitation of textual meaning in chinese-english conference interpreting]. *PLA Foreign Studies University Journal*, 4:67–73.

Kahane, E. (2000). Thoughts on the quality of interpretation. *aiic.net May 13, 2000. Accessed September 30, 2017. http://aiic.net/p/197*.

Kalina, S. (1998). *Strategische Prozesse beim Dolmetschen: Theoretische Grundlagen, empirische Fallstudien, didaktische Konsequenzen*, volume 18. G. Narr.

Khurana, S. and Ali, A. (2016). QCRI advanced transcription system (QATS) for the Arabic multi-dialect broadcast media recognition: MGB-2 challenge. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*.

Kroll, J. F. and De Groot, A. M. (2009). *Handbook of bilingualism: Psycholinguistic approaches*. Oxford University Press.

Kumar, G., Cao, Y., Cotterell, R., Callison-Burch, C., Povey, D., and Khudanpur, S. (2014). Translations of the CALLHOME Egyptian Arabic corpus for conversational speech translation. In *Proceedings of International Workshop on Spoken Language Translation (IWSLT)*.

Roderick, J. (2002). *Conference Interpreting Explained (Translation Practices Explained)*. St. Jerome Publishing.

Salameh, M., Zantout, R., and Mansour, N. (2011). Improving the accuracy of English-Arabic statistical sentence alignment. In *Int. Arab J. Inf. Technol.*, volume 8, pages 171–177.

Shlesinger, M. (1998). Corpus-based interpreting studies as an offshoot of corpus-based translation studies. *Meta: Journal des traducteurs/Meta: Translators' Journal*, 43(4):486–493.

Temnikova, I., Abdelali, A., Hedaya, S., Vogel, S., and Al Daher, A. (2017). Interpreting strategies annotation in the WAW corpus. In *The First Workshop on Human-Informed Translation and Interpreting Technology (HiT-IT), co-located with RANLP 2017. September 7 2017, Varna, Bulgaria.*

Tohyama, H. and Matsubara, S. (2006). Collection of simultaneous interpreting patterns by using bilingual spoken monologue corpus. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*.

Wray, S. and Ali, A. (2015). Crowdsource a little to label a lot: Labeling a Speech Corpus of Dialectal Arabic. In *Interspeech*. (in press).

Zaghouani, W. (2017). Critical survey of the freely available Arabic corpora. *arXiv preprint arXiv:1702.07835*.

Zaidan, O. F. and Callison-Burch, C. (2014). Arabic dialect identification. *Computational Linguistics*, 40(1):171–202.

Zwischenberger, C. and Pöchhacker, F. (2010). Survey on quality and role: Conference interpreters expectations and self-perceptions. *Communicate! AIIC Webzine*.

Polish Corpus of Annotated Descriptions of Images

Alina Wróblewska
Institute of Computer Science, Polish Academy of Sciences
Warsaw, Poland
alina@ipipan.waw.pl

Abstract

The paper presents a new dataset of image descriptions in Polish. The descriptions are morphosyntactically analysed and the pairs of these descriptions are annotated in terms of semantic relatedness and entailment. All annotations are provided by human annotators with strong linguistic background. The dataset can be used for evaluation of various systems integrating language and vision. It is applicable for evaluation of systems designed to image generation based on provided descriptions (text-to-image generation) or to caption generation based on images (image-to-text generation). Furthermore, as selected images are split into thematic groups, the dataset is also useful for validating image classification approaches.

Keywords: language and vision, evaluation dataset, annotated image descriptions, Polish

1. Introduction

Language and vision are two essential modalities that enable interpersonal communication. As broadly understood communication is an important area of artificial intelligence, AI researchers attach importance to image processing, natural language processing, and integration of these two fields. Vision-to-language approaches consist mostly in mapping images to sentences, e.g. Hodosh et al. (2013), or generating image descriptions, e.g. Xu et al. (2015), Karpathy and Fei-Fei (2017). For example, Karpathy and Fei-Fei (2017) propose a model that generates natural language descriptions of whole images or their regions. Based on datasets of images and their sentence descriptions, the model learns about the inter-modal correspondences between language and vision. Regarding language-to-vision approaches, research and scientific experiments are conducted in the areas of text-based image retrieval, e.g. Rasiwasia et al. (2010), or text-based image generation, e.g. Denton et al. (2015). For example, the paper by Denton et al. (2015) describes a "generative parametric model capable of producing high quality samples of natural images". In order to evaluate many of these approaches, the high-quality caption-image datasets are necessary.

This paper presents a new Polish dataset of annotated image descriptions – AIDe[1] (**A**nnotated **I**mage **De**scriptions). The dataset consists of 2K natural language descriptions of 1K images. The dataset is probably too small for training a sophisticated language–vision system. For training purposes, the dataset should be expanded to the greatest possible extent. However, in the age of intensive research on multilingual NLP, e.g. Faruqui and Dyer (2014), it seems to make sense to build even small but high-quality evaluation resources.

The presented dataset can be used for evaluation of various systems integrating language and vision. It is applicable for evaluation of systems designed to generation of images based on provided descriptions (text-to-image generation) or to generation of captions based on images (image-to-text generation). Furthermore, as elected images are split into thematic groups, the dataset is also useful for validating image classification approaches.

The procedure of selecting, describing, and splitting images into thematic groups is described in Section 2. The image descriptions are morphosyntactically annotated (see Section 3.) and the pairs of these descriptions are annotated in terms of semantic relatedness and entailment (see Section 4.). All annotations are provided by human annotators with strong linguistic background.

2. Dataset

2.1. Image Classification

The first step of building the dataset consists in selecting 1K images from the Flickr8k dataset (Hodosh et al., 2013). The selected images are arbitrarily split into 46 thematic groups.[2]

As the classes do not link to any of standard ontologies and each image is classified into only one group, even if some of them could be classified into multiple classes, we decided to reclassify the images according to the heuristics based on WordNet hyperonym hierarchy[3] (Fellbaum, 1998).

[1] http://zil.ipipan.waw.pl/Scwad/AIDe

[2] The images are assigned to the following thematic groups: **people** (kids, different_people), **animals** (dogs, birds, different_animals), and sport and leisure activities: **water activities** (fishing, swimming, surfing, kayaking, boating_or_sailing), **winter activities** (skiing, snowboarding, sledding, ice-skating), **driving and riding** (driving, motorbike_riding, quad-bike_riding, biking, non-motor_vehicle_riding, horse_riding, (inline)roller-skating, skateboarding), **playing** (jumping, jumping_to_water, jumping_on_trampoline, swinging, sliding_down, dancing), **team games** (basketball, football, volleyball, baseball_or_rugby, hockey), **individual activities** (individual_sports, martial_arts, climbing, mountain_hiking, running_or_jogging), **unclassified activities** (flying, photographing, telephoning, kissing, musical_instruments, eating, resting, sunbathing).

The numbers of images within individual thematic groups vary from 6 images in the volleyball and telephoning groups to 94 images in the different_people group. The second largest groups are children and dogs with 50 images each.

[3] We use English categories from WordNet in our dataset, however these classes could be straightforwardly map onto plWordNet (Rudnicka et al., 2012).

We distinguish 3 class types: events (**Event**), entities (**Entity**), i.e. participants and artefacts, and general location (**Out-In**), i.e. outside vs. inside (the building). Each image has to be assigned to at least one class within each of three class groups:

- **Event** classes (24):[4]

sing (6)	swing (21)	game (47)
smoke (6)	lie (22)	run (68)
kiss (8)	climb (26)	walk (82)
fish (16)	fly (28)	play (95)
consume (18)	swim (33)	sit (104)
carry (19)	watch (34)	ride (125)
photograph (19)	go (40)	jump (127)
dance (20)	skate (45)	stand (186)

- **Entity** classes (6):[5]

food (23)	vehicle (166)
instrument (49)	animal (172)
artifact (141)	person (876)

- **Out**side-**In**side classes (2):[6]

inside (211) outside (845)

[4]The number in brackets corresponds to the number of images classified into the particular class. The game class is further divided into 8 subclasses (**Event-hypon**ym):

cricket (2)	ice hockey (5)	soccer (9)
rugby (2)	football (7)	basketball (10)
field hockey (4)	baseball (8)	

[5]Some of the classes within **Entity** group are further divided into the following subclasses (**Entity-hypon**ym):

fish (2)	roller skate (18)
insect (2)	sledge (18)
wheelchair (2)	snowboard (18)
aquatic mammal (4)	bicycle (20)
racket (4)	horse (20)
equipment (5)	motorcycle (20)
reptile (5)	skateboard (20)
weapon (5)	ski (20)
scooter (6)	slide (20)
cat (7)	surfboard (20)
percussion instrument (7)	boat (21)
aircraft (8)	hoofed mammal (22)
ball (8)	car (23)
mammal (8)	guitar (25)
bike (9)	bird (41)
string (9)	dog (71)
trampoline (9)	athlete (82)
kayak (10)	woman (283)
skate (10)	child (323)
wheeled vehicle (14)	man (441)
wind (15)	

[6]**Inside-Outside** classes are sometimes specified with the following subclasses:

stairs (9)	court (19)	street (53)
ring (10)	park (26)	track (101)
apartment (12)	stadium (26)	water (104)
ice rink (16)	pool (28)	field (118)
shore (16)	beach (36)	
playground (17)	mountain (49)	

The classes of **Event** type correspond mostly to the sentence predicates. The generalised **Entities** correlate, in turn, with predicate arguments. Finally, the classes of **Out-In** group can be equated with the location adverbials.

Statistics One image can be classified into more than one class within a thematic group (see Table 1). Especially in the group of Entities, the images are assigned to more than one class (see 2-fold classification).

	1-fold	2-fold	3-fold
Event	808	189	3
Entity	582	409	9
Outside_**In**side	944	56	0

Table 1: Classification statistics: 1-fold, 2-fold, and 3-fold classifications correspond to processes of assigning an image to one, two, or three classes (within one class type), respectively.



Figure 1: An example image from https://www.flickr.com/photos/floridatania/1057089366.

Example The example image (see Figure 1) is classified as follows:

- Event: jump
- Entity: person (subclass of the person class: child)
- Location: outside (subclass: pool).

2.2. Image Descriptions

The chosen images are presented to two authors who, independently of each other, formulate their descriptions based on a short instruction. The authors are instructed to write one single sentence (with a sentence predicate) describing the entities, events and scenes depicted in a displayed image. They should not describe an imaginable context or an interpretation of what may lie behind the scene in the picture. If some details in the picture are not obvious, they should not be described either. Finally, the descriptions should contain Polish diacritics and proper punctuation.

The final set of image descriptions consists of 2K sentences, i.e. two sentences for each image. The descriptions of the same image are not doublets.

Statistics The authors write similarly long sentences. The average length of sentences written by the A author is 12.42 tokens. The B author's description length is 12.49 tokens per sentence on average.

In order to estimate textual similarity between sentences in each pair we apply two measures: Monge-Elkan similarity measure (Monge and Elkan, 1996) and tokens sort ratio based on approximate string matching (Wagner and Fisher, 1974). Monge-Elkan distance is a hybrid measure for computing similarity between two strings of multiple tokens using an internal measure (e.g. Jaro-Winkler or Levenshtein) to estimate similarity between individual tokens. Measured with Monge-Elkan distance, textual similarity between description pairs A and B is 0.811 (average)[7] and between pairs B and A is 0.809 (average)[8].

Approximate string matching token sort splits two strings into tokens, sorts the tokens, and estimates the similarity of the sorted strings. Textual similarity between desription pairs is 0.595 (average)[9] measured with approximate string matching token sort ratio.

Example The image in Figure 1 is described as follows in our dataset:

A. *Chłopiec skacze do basenu z wysokiej trampoliny.*
(Eng. 'A boy is jumping into the pool off a high diving board.')

B. *Chłopiec w niebieskim ubraniu skacze do basenu z trampoliny.*
(Eng. 'A boy in blue clothes is jumping into the pool off a diving board.')

3. Morphosyntactic Annotations

The similar datasets, e.g. for English, typically consist of images combined with multiple captions, e.g. Rashtchian et al. (2010),[10] Hodosh et al. (2013),[11] Lin et al. (2014).[12]

The morphosyntactic annotations of English captions are usually not provided, because English is a resource-rich language and there are plenty of high-quality English NLP tools. Polish, in turn, is a resource-poor language which still suffers from the lack of high-quality NLP tools. In order to avoid propagation of tagging or parsing errors[13] to the evaluation of language-vision systems, we provide morphosyntactic annotations of our Polish image descriptions. Furthermore, with annotated data, it is possible to verify the impact of tagging and/or parsing on the overall quality of language-vision systems.

3.1. Annotation Procedure

Each description is tokenised and morphologically analysed with Morfeusz (Woliński, 2014) and tagged and lemmatised with Concraft (Waszczuk, 2012). The sentences are then parsed with MaltParser (Nivre, 2009) and Mate parser (Bohnet, 2010) trained[14] on Polish Dependency Bank (Wróblewska, 2014).

Two dependency trees and accompanying morphosyntactic annotations (lemmas, part-of-speech tags, morphological features) are manually verified and possibly corrected by two linguists. Finally, two verified dependency trees are unified by the third linguist who is the most experienced in Polish linguistics.

3.2. Dependency Tree Formats

The dependency trees and accompanying morphosyntactic annotations are stored in column-based CoNLL format (Nivre et al., 2007). As the format of Universal Dependencies (UD)[15] becomes more and more common, the dependency trees are also automatically converted into corresponding UD trees.

token	lemma	POS	morph
Chłopiec	CHŁOPIEC	subst	sg|nom|m1
skacze	SKAKAĆ	fin	sg|ter|imperf
do	DO	prep	gen
basenu	BASEN	subst	sg|gen|m3
z	Z	prep	gen|nwok
wysokiej	WYSOKI	adj	sg|gen|f|pos
trampoliny	TRAMPOLINA	subst	sg|gen|f

Figure 2: The morphosyntactic analysis of *Chłopiec skacze do basenu z wysokiej trampoliny.* (Eng. 'A boy is jumping into the pool off a high diving board.').

Example Taking the A caption as an example (see Section 2.2.), its morphosyntactic analysis with part-of-speech tags and morphological features, and its dependency tree are in Figure 2[16] and Figure 3, respectively. The UD-formatted example is in Figures 4 and 5.

[7] Standard deviation of the average similarity scores estimated with Monge-Elkan measure: 0.086.

[8] Standard deviation of the average similarity scores estimated with Monge-Elkan measure: 0.085.

[9] Standard deviation of the average similarity scores estimated with approximate string matching: 0.114.

[10] http://vision.cs.uiuc.edu/pascal-sentences

[11] http://nlp.cs.illinois.edu/HockenmaierGroup/8k-pictures.html

[12] http://cocodataset.org

[13] Some statistics about the quality of Polish NLP tools are collected on http://clip.ipipan.waw.pl/benchmarks.

[14] http://zil.ipipan.waw.pl/PDB/PDBparser

[15] http://universaldependencies.org

[16] Explanation of grammatical classes and categories:
subst – substantive

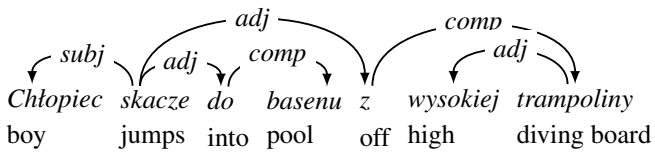


Figure 3: The dependency tree of *Chłopiec skacze do basenu z wysokiej trampoliny.* (Eng. 'A boy is jumping into the pool off a high diving board.').

token	UD-POS	UD-feature
Chłopiec	NOUN	Animacy=Hum,Case=Nom, Gender=Masc,Number=Sing
skacze	VERB	Aspect=Imp,Mood=Ind, Number=Sing,Person=3, Tense=Pres,VerbForm=Fin
do	ADP	AdpType=Prep,Case=Gen
basenu	NOUN	Animacy=Inan,Case=Gen, Gender=Masc,Number=Sing
z	ADP	AdpType=Voc,Case=Gen, Variant=Short
wysokiej	ADJ	Case=Gen,Degree=Pos, Gender=Fem,Number=Sing
trampoliny	NOUN	Case=Gen,Gender=Fem, Number=Sing

Figure 4: The UD-formatted morphosyntactic analysis of *Chłopiec skacze do basenu z wysokiej trampoliny.* (Eng. 'A boy is jumping into the pool off a high diving board.').

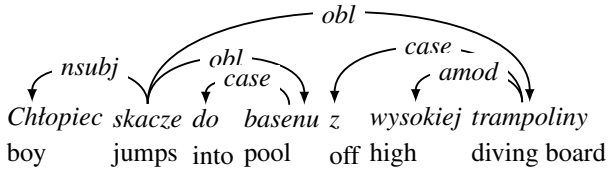


Figure 5: The UD-formatted dependency tree of *Chłopiec skacze do basenu z wysokiej trampoliny.* (Eng. 'A boy is jumping into the pool off a high diving board.').

prep – preposition
fin – finite verb
adj – adjective
sg – singular number
nom – nominative case
gen – genitive case
m1 – human masculine (virile) gender
m3 – inanimate masculine gender
f – feminine gender
ter – third person
imperf – imperfect aspect
nwok – non-vocalic
pos – positive degree (of adjectives)

4. Semantic Annotations

In order to test a system of image generation based on two descriptions, information whether these two descriptions are semantically related, or whether the meaning of one description entails the meaning of the other one, seems to be relevant. Therefore, each sentence pair is human-annotated for relatedness in meaning and entailment. The semantic annotations are derived from Polish CDSCorpus[17] (Wróblewska and Krasnowska-Kieraś, 2017).

4.1. Semantic Relatedness

The relatedness score corresponds to the degree of semantic relatedness between two sentences and is calculated as the average of six human ratings collected for this sentence pair on the 6-point Likert scale (0 to 5). This score indicates the extent to which the meanings of two sentences are related. The score 5 indicates very related descriptions and the score 0, in turn, indicates unrelated descriptions. The scores 1–4 denote that the pair consists of more or less related descriptions. The degree of semantic relatedness is not equivalent to the degree of semantic similarity. Semantic similarity is only a special case of semantic relatedness, semantic relatedness is thus a more general term than the other one.

Statistics Table 2 aggregates the occurrences of 6 possible relatedness scores, calculated as the mean of all 6 individual annotations, rounded to an integer.

relatedness	# of pairs
0	0
1	11
2	99
3	418
4	440
5	32

Table 2: Relatedness scores rounded to integers (total: 1K pairs).

Example The captions A and B (see Example in Section 2.2.) are annotated as quite related (scored 4) in our dataset.

4.2. Entailment Relations

The entailment relation between two descriptions of the same image is labelled with *entailment* or *neutral*.[18] The description pairs are annotated for entailment in both directions (i.e. bidirectional entailment annotations), because an entailment relation between two sentences must not be symmetric. The final entailment label is actually a pair of two labels:

[17] http://zil.ipipan.waw.pl/Scwad/CDSCorpus

[18] There is also the label *contradiction*, but it is not present in the dataset. This is in line with our assumption that two sentences describing one image should not be contradictory.

- *entailment+neutral* points to one-way entailment,[19]
- *entailment+entailment* points to equivalence (two-way entailment),
- *neutral+neutral* points to no entailment.

The label assigned by the majority of 3 human annotators is selected as the valid entailment label.

Statistics Table 3 shows the number of the particular entailment labels in the corpus.

entailment	# of pairs
entailment+entailment	40
entailment+neutral	293
neutral+neutral	667

Table 3: Entailment labels (total: 1K pairs).

Example The descriptions A and B (see Example in Section 2.2.) are labelled *neutral* in both entailment directions.

5. Multilingual Variant of the Dataset

The decision to choose images from Flickr8k was well thought out. Based on Flickr8k dataset and AIDe dataset it is possible to compile a new Polish-English multi-parallel corpus, which could be used for validating e.g. machine translation systems.
In our dataset, we apply original image IDs of Flickr8k. In order to build the multi-parallel corpus, for each image ID, e.g. 1057089366_ca83da0877.jpg,[20] the following sentences should be selected:

- two Polish sentences from AIDe (see A-PL and B-PL in the list in Example),
- five English captions from Flickr8k (see 0-EN to 4-EN in Example)[21].

As Polish is a resource-poor language, each new resource is valuable, even if it is a by-product like in this case.

Example An excerpt of a possible Polish-English multi-parallel corpus:

A-PL Chłopiec skacze do basenu z wysokiej trampoliny.

B-PL Chłopiec w niebieskim ubraniu skacze do basenu z trampoliny.

0-EN A boy descends off the end of a high diving board.

1-EN A child jumps off a high diving board into the pool.

2-EN A kid jumps off the diving board and into the swimming pool below.

3-EN A little kid is jumping off a high dive at the pool.

4-EN The boy is jumping off a high diving board into the pool.

6. Conclusions

The aim of this paper was to present AIDe – Polish dataset of annotated image descriptions. The descriptions were morphosyntactically annotated, i.e. tokens were assigned part-of-speech tags and morphological features, and sentences were represented as dependency trees (also UD-formatted trees). Apart from morphosyntactic annotations, we also provided semantic annotations of description pairs. Image description pairs were annotated with semantic relatedness scores and bidirectional entailment labels. All annotations were provided by human annotators with strong linguistic background.
We decided to augment raw image descriptions with morphosyntactic annotations, in order to provide a dataset which is designed to evaluate the correspondence between language and vision undisturbed by errors at the lower language processing stages. Furthermore, pre-annotated data enable verification of the impact of tagging and/or parsing on the overall quality of language-vision systems.
We are not aware of availability of other image-caption datasets which are annotated with semantic relatedness scores and entailment labels. Hence, our dataset enables to check whether additional semantic information is useful e.g. in image generation based on multiple sentences.
The dataset is small and without increasing its size it is insufficient for training purposes. This high-quality resource is rather intended for evaluation of systems integrating language and vision (text-to-image or image-to-text generation), for image classification, as selected images are split into thematic groups based on WordNet, or even for evaluation of machine translation systems, if corresponding captions are extracted from Flickr8k dataset.

7. Acknowledgements

The research presented in this paper was founded by SONATA 8 grant no 2014/15/D/HS2/03486 from the National Science Centre Poland and by the Polish Ministry of Science and Higher Education as part of the investment in the CLARIN-PL research infrastructure.
The author would like to thank the anonymous reviewers for their careful reading of the paper and their valuable comments and suggestions.

[19] While the actual corpus labels are ordered in the sense that there is a difference between e.g. *entailment+neutral* and *neutral+entailment* (the entailment occurs in different directions), we treat all labels as unordered for the purpose of this summary (e.g. *entailment+neutral* covers *neutral+entailment* as well, representing the same type of relation between two sentences).

[20] This is the ID of the image in Figure 1.

[21] The original IDs of English captions are:
1057089366_ca83da0877.jpg#0
1057089366_ca83da0877.jpg#1
1057089366_ca83da0877.jpg#2
1057089366_ca83da0877.jpg#3
1057089366_ca83da0877.jpg#4

8. Bibliographical References

Bohnet, B. (2010). Very High Accuracy and Fast Dependency Parsing is not a Contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING 2010, pages 89–97.

Denton, E. L., Chintala, S., Szlam, A., and Fergus, R. (2015). Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In C. Cortes, et al., editors, *Advances in Neural Information Processing Systems 28*, pages 1486–1494. Curran Associates, Inc.

Faruqui, M. and Dyer, C. (2014). Improving Vector Space Word Representations Using Multilingual Correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471.

Christiane Fellbaum, editor. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing Image Description As a Ranking Task: Data, Models and Evaluation Metrics. *Journal of Artificial Intelligence Research*, 47(1):853–899.

Karpathy, A. and Fei-Fei, L. (2017). Deep Visual-Semantic Alignments for Generating Image Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):664–676.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In David Fleet, et al., editors, *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, pages 740–755. Springer International Publishing.

Monge, A. E. and Elkan, C. P. (1996). The field matching problem: Algorithms and applications. In *Proceedings of The Second International Conference on Knowledge Discovery and Data Mining*, pages 267–270.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 Shared Task on Dependency Parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932.

Nivre, J. (2009). Non-Projective Dependency Parsing in Expected Linear Time. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, volume 1, pages 351–359.

Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. (2010). Collecting Image Annotations Using Amazon's Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 139–147. Association for Computational Linguistics.

Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G. R., Levy, R., and Vasconcelos, N. (2010). A New Approach to Cross-modal Multimedia Retrieval. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 251–260.

Rudnicka, E., Maziarz, M., Piasecki, M., and Szpakowicz, S. (2012). A strategy of mapping Polish Wordnet onto Princeton Wordnet. In *Proceedings of COLING 2012: Posters*, pages 1039–1048, Mumbai, India, December. The COLING 2012 Organizing Committee.

Wagner, R. A. and Fisher, M. J. (1974). The String-to-String Correction Problem. *Journal of the Association for Computing Machinery*, 21(1):168–173.

Waszczuk, J. (2012). Harnessing the CRF complexity with domain-specific constraints. The case of morphosyntactic tagging of a highly inflected language. In *Proceedings of COLING 2012: Technical Papers*, pages 2789–2804.

Woliński, M. (2014). Morfeusz reloaded. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 1106–1111. ELRA.

Wróblewska, A. and Krasnowska-Kieraś, K. (2017). Polish Evaluation Dataset for Compositional Distributional Semantics Models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 784–792, Vancouver, Canada. Association for Computational Linguistics.

Wróblewska, A. (2014). *Polish Dependency Parser Trained on an Automatically Induced Dependency Bank*. Ph.D. dissertation, Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In David Blei et al., editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2048–2057.

Action Verb Corpus

Stephanie Gross[1], Matthias Hirschmanner[2], Brigitte Krenn[1], Friedrich Neubarth[1], Michael Zillich[2]
Austrian Research Institute for Artificial Intelligence[1], Technical University Vienna (ACIN)[2]
Freyung 6, Gußhausstrasse 27-29
1010 Vienna, 1040 Vienna
Austria
{stephanie.gross, brigitte.krenn, friedrich.neubarth}@ofai.at, {matthias.hirschmanner,michael.zillich}@tuwien.ac.at

Abstract

The Action Verb Corpus comprises multimodal data of 12 humans conducting in total 390 simple actions (TAKE, PUT, and PUSH). Recorded are audio, video and motion data while participants perform an action and describe what they do. The dataset is annotated with the following information: orthographic transcriptions of utterances, part-of-speech tags, lemmata, information which object is currently moved, information whether a hand touches an object, information whether an object touches the ground/table. Transcription, and information whether an object is in contact with a hand and which object moves where to were manually annotated, the rest was automatically annotated and manually corrected. In addition to the dataset, we present an algorithm for the challenging task of segmenting the stream of words into utterances, segmenting the visual input into a series of actions, and then aligning visual action information and speech. This kind of modality rich data is particularly important for crossmodal and cross-situational word-object and word-action learning in human-robot interactions, and is comparable to parent-toddler communication in early stages of child language acquisition.

Keywords: multimodality, corpus, language modelling

1. Motivation and Related Work

In the following, the multimodal Action Verb Corpus (AVC) is presented, comprising visual and linguistic (German) information related to instances of TAKE, PUT, and PUSH actions. The corpus consists of 140 instances of TAKE/PUT actions each, and 110 instances of PUSH actions). It is geared to robotic action word learning inspired by early word learning of infants. The corpus focuses on modality rich input for crossmodal, cross-situational learning of word-object and word-action mappings. The speciality of AVC is that people perform an action and verbalize what they are doing, leading to rich multisensory data. This is specific to task-oriented communication as well as to parent-toddler communication where the parents' speech is often connected to what the infants see or do (Suanda et al., 2016).

The corpus is small compared to, for instance, the Kinetics video dataset (Kay et al., 2017) comprising more than 300,000 videos of complex actions, or CLEVR (Johnson et al., 2016) comprising a total of 100,000 images of object configurations (locations, object shape, size, color, texture) and approximately a million questions related to those images. While the latter two are designed for training and testing of deep learning models for human action classification such as 'brushing hair', 'riding a bike', etc. (Kinetics), or for diagnosing visual question-answering systems (CLEVR), the Action Verb Corpus focuses on the multimodal representation of instances of TAKE, PUT, and PUSH actions, recorded from adults in order to reflect what a robot will be exposed to when being trained by humans.

In this respect, AVC is complementary to existing child-specific datasets (Nilsson Björkenstam and Wirén, 2013; MacWhinney, 2000). The former collected a longitudinal corpus of parent-child interactions including transcriptions of child-directed speech, child vocalizations, and annotations of gestures, eye gaze, and object-related actions by both parent and child. The CHILDES corpus (MacWhinney, 2000) is a large repository of first language acquisition data in different languages (amongst others German and English), mainly transcriptions of parent and child utterances. Some of the data contain video and audio recordings. Extra-linguistic cues in language learning and the role of social interaction can be investigated based on these data. However, the interactions do not contain the same tasks conducted by different child-adult dyads.

(Gaspers et al., 2014) present a corpus where caregivers/instructors verbally describe learning tasks. Their instructions are - similar to ours - based on visual input. The corpus has information on the wrist position included, so that a robot is able to follow the hand with its gaze. The AVC, in addition, provides information on the arm trajectories and finger joints, and information which object is held by the speaker and where it is moved to.

Three examples for human-robot interaction are the Home Tour Corpus (Green et al., 2006), the Vernissage corpus (Jayagopi et al., 2013), and the Rolland corpus (Anastasiou, 2012). In the first one, users show an environment (a single room, or a whole floor) in a WoZ-Setting to a robot. Recorded were speech, gestures, and gaze. In the second one, a robot induces interactive behavior with and between humans by explaining paintings in a room and then quizzing the participants. In the third one, a user was asked to carry out a set of simple tasks with a powered wheelchair. In the WoZ Setup, the user interacted with the wheelchair via language and gestures. For an overview on annotation tools and schemes, see (Abuczki and Ghazaleh, 2013) or (Tenbrink et al., 2013).

In general, current multimodal corpora typically include language, objects in the visual field, eye gaze, and pointing gestures. An exception is the JCT corpus (Foster et al., 2008) which also includes the currently moved object in object manipulation tasks. Other than in the AVC, the actions

were conducted on a screen and not in a real-world setting. These corpora are partially suitable to evaluate multimodal computational models for object-word learning. However, they are not sufficient for crossmodal action learning.

2. Setup for Collecting Data

2.1. Task Scenarios

The overall goal of the data collection activity was to gather multimodal data of basic actions such as TAKE (Ge: *nehmen*), PUT (Ge: *stellen/legen*) and PUSH (Ge: *schieben*). The participants were instructed to perform actions with three objects positioned on a table – a bottle, a can and a box – and verbalize what they are doing. Task 1 was to take an object from the table and put it back on the table at a specific position and use the action verbs *nehmen, legen/stellen* (En: take, put) when verbalizing one's action. Task 2 is comparable to Task 1 but with a focus on verbally specifying the spatial location where the object is put. In Task 3, the user had to push objects to certain locations. The actual actions the user had to perform were displayed in the Virtual Reality headset (Figure 3, right). Sample configurations for Tasks 1 and 2 are shown in Figure 1. In total, there were 4 configurations each depicting a start situation, an intermediate and an end situation. Possible verbalizations for the tasks depicted in Figure 1 would be: T1 – *ich nehme die Flasche und stelle sie neben die Schachtel* ('I take the bottle and put it next to the box'), T2 – *ich nehme die Flasche stelle sie rechts neben die Dose* ('I take the bottle and put it on the right side of the can'), whereby the picture on the left shows the start configuration, the one at the right (2) shows the end configuration, and the one in the middle (1) shows the intermediate configuration where the moved object is not on the table. As regards Task 3, the users were presented with a starting configuration and 10 intermediate configurations indicating which object had to be moved to which position, see Figure 2 for 3 sample moves. Possible verbalizations would be: *ich schiebe die Schachtel rechts neben die Flasche* ('I push the box to the right side of the bottle') (3), *ich schiebe die Flasche zwischen die Schachtel und die Dose* ('I push the bottle between the box and the can') (4), *ich schiebe die Flasche links neben die Dose* ('I push the bottle to the left of the can') (5). While in the TAKE/PUT actions of T1 and T2 the moved object leaves the table (intermediate configuration Figure 1), the pushed object in T3 stays on the table while changing position (configurations 3, 4 and 5 in Figure 2).

2.2. Technical Setup

The user sits (or stands) in front of the table and is wearing an Oculus Rift DK2 Virtual-Reality (VR) headset[1] mounted with the Leap Motion sensor[2] for hand tracking, see Figure 3, left. A camera (Microsoft Kinect) is positioned opposite of the user directed at the table for object tracking. The user performs different actions defined by visual instructions and verbally describes what he/she is doing. Two microphones record the user's description of the performed action. The data is recorded in such a way that it resembles

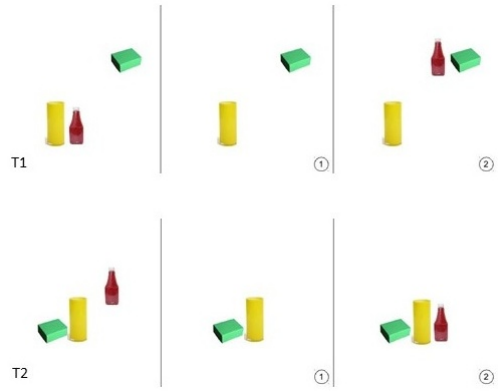


Figure 1: Sample instructions for tasks T1 and T2.



Figure 2: Sample instructions for task T3.

what a robot sees and hears. Letting the human see through the robot's eyes forces the subjects to perform more pronounced movements within a restricted field of vision. This facilitates processing of the input by the artificial system.
The Leap Motion is a stereo infrared camera which is specialized on hand tracking. The Software Development Kit (SDK) provides detailed information on the position of the various joints of the user's arm down to the finger segments. We use the Leap Motion mounted on a VR headset to have the best available tracking performance.



Figure 3: Experimental setup (left) and view through Oculus including instructions (right).

The Oculus Rift DK2 is worn by the user and provides the user's head pose. It is needed to transfer the tracking data of the Leap Motion to a fixed coordinate system. The instructions for the current task are displayed in the Oculus Rift above the camera images of the Leap Motion. This way, the users are able to look at the instructions without moving their heads as it would be the case if they had to look at printout versions of the instructions. Additionally, the setup forces the user to direct the Leap Motion at his/her hands to see what he/she is doing. This behavior is necessary for satisfying hand tracking performance.
For object tracking, the RGB as well as depth data of the

[1] https://www.oculus.com/dk2/
[2] https://www.leapmotion.com

Kinect camera is recorded as a ROS[3] bag on a separate machine running Ubuntu. The models of the objects were created beforehand with the RTM-Toolbox from the V4R library[4]. The objects were positioned on a turntable and a Kinect camera recorded a sequence of RGB-D images. The software tool uses a Kanade-Lucas-Tomasi feature tracker with FAST features to create an object model as described in (Prankl et al., 2015). The features for creating the model are saved with the model and used for object tracking.

We apply the object tracker from the V4R Library on the recorded data from the user experiments[5]. It utilizes the Perspective-n-Point algorithm to calculate the 6-Degrees-of-Freedom pose of the objects from a monocular image stream. The offline tracking enables the best possible results because the object tracker can be tuned for a specific recording. Besides the position and orientation of the object, two Boolean variables are saved: object is in contact with the table and object is in contact with a hand. The former is set automatically depending on the object's position, the latter is currently manually annotated. We use the following information to process the raw data from the object tracker: If the object is in contact with the user's hand, the object position is saved as received from the object tracker. If the object is not in contact with the hand, we calculate an average object pose because we assume that the object does not move. This assumption is correct in our experimental setup, as long as the user does not push an object with a different object. The weighted arithmetic mean is used whereby the weight is the confidence received from the object tracker. Tracked object positions with a low confidence or further away than 10 cm from the average position are discarded to omit wrong positives. However, we count the number of outliers with a high confidence. If this number exceeds a certain threshold, the current average position is discarded. This way, the jittering of the raw object position is cancelled out and occlusions do not impair tracking performance if the object was successfully tracked before.

The recorded data from the different systems (hand and object tracking) are transformed to a common coordinate frame (fixed in the user's head) and temporally aligned in a post-processing step. This alignment is done manually.

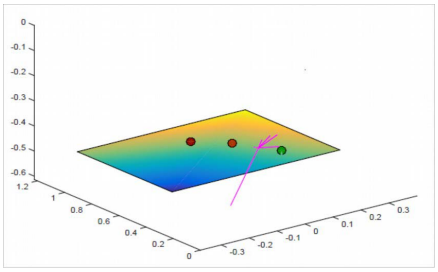


Figure 4: Visualization of manipulating hand and objects.

[3] http://www.ros.org

[4] http://www.acin.tuwien.ac.at/forschung/v4r/software-tools/rtm/

[5] http://www.acin.tuwien.ac.at/forschung/v4r/software-tools/v4r-library/

3. Dataset and Annotations

The corpus, at the time of writing, comprises 140 instances of TAKE/PUT actions each and 110 instances of PUSH actions from 20 recordings of Task 1, 15 recordings of Task 2 and 11 recordings of Task 3. See Table 1 for an overview. Overall, 12 persons conducted the tasks.

Task	Number of Recordings	Number of Actions per Recording
T1	20	4 TAKE/PUT-actions
T2	15	4 TAKE/PUT-actions
T3	11	10 PUSH-actions

Table 1: Composition of corpus.

3.1. Representation of Information

Apart from the raw data from hand and object tracking, the audio recordings, and grey-scale videos from the user perspective, each instance of a task is represented by:

1. the merged output of the hand and the object tracker, including per frame the 3D positions of the joints in the elbow, wrist, and knuckles of the instructor's hands as well as the object poses and their reliability estimate calculated by the object tracker;
2. an animation of the merged hand and object tracking, see Figure 4;
3. manual orthographic transcriptions of speech: one is close to what is actually spoken, the other one is normalized with respect to standard NLP tools such as taggers, parsers, stemmers, etc.;
4. part-of-speech tags, automatically generated with the Tree-Tagger (Schmid, 1995) and manually corrected;
5. lemmata, automatically generated with the Tree-Tagger and manually corrected;
6. the information which object is moved and where it is moved to (manually annotated);
7. the information whether the left or right hand touches a particular object (manually annotated);
8. the information whether a particular object touches the ground/table (automatically identified by the object tracker and manually corrected);
9. segmentation of the stream of words into chunks using heuristics such as long pauses (min. 0.5 sec) or connectives (e.g. *und* – 'and');
10. segmentation of the visual input into situations comprising actions: in our specific dataset, a situation is construed each time a hand touches an object and the object moves. If the object is lifted from the ground, the situation consists of the two actions TAKE and PUT, otherwise it is a PUSH action.

While the information described in 1. is provided as a CSV file per recording, the information described in 3. to 10. (henceforth the annotation tiers) is represented as Elan[6] file (.eaf) and CSV exported from Elan. All annotation tiers are synchronized with the real-time animation of the hand and object tracking and with the speech stream. Praat[7] is used

[6] https://tla.mpi.nl/tools/tla-tools/elan/

[7] http://www.fon.hum.uva.nl/praat/

for transcribing the utterances.[8]

3.2. Alignment of Visual Action and Speech

For tasks such as crossmodal learning of word-object and word-action references, the chunks obtained from the transcribed speech as described in 9. above, and the actions retrieved from the visual representation of situations (see 10. above) need to be aligned. Each situation – a multimodal perceptual frame – comprises one or two events. An event consists of an action (with all involved objects) and an utterance that verbally describes what happens. In order to obtain these events, information from different modalities has to be processed in order to identify actions, utterances and correct alignment between them.

For the given dataset, each action ideally is described by exactly one utterance (made up of one or several chunks), see for instance the following example where the speech sequence *ich schiebe die Dose* ('I push the can') is followed by a (relatively long) pause of 2.082 seconds and then followed by the sequence *neben die Schachtel* ('next to the box'). These two chunks need to be aggregated into one utterance and aligned to the corresponding action. Alignment proceeds in two steps: first the right boundary is identified from the set of chunks, in a second step the algorithm works backwards in order to determine the left boundary. When a situation consists of two actions, the string of selected chunks has to be split into two utterances, otherwise the string of chunks directly maps to a single utterance.

In order to identify the rightmost speech chunk that is to be aligned to the current action(s), the algorithm proceeds left to right on the list of chunks. The endpoint of the situation (plus some latency, here set to 0.7 sec) marks the preliminary cutoff point (1), see Figure 5. Chunks that start later than that can be connected to the preceding chunk, if the pause between the previous and the currently examined chunk does not exceed a given threshold of 1.4 sec (2). In addition to that, the chunk must not overlap with the next situation (3) and must not start with a connective ('and') since this would hint to the begin of a new utterance (4).

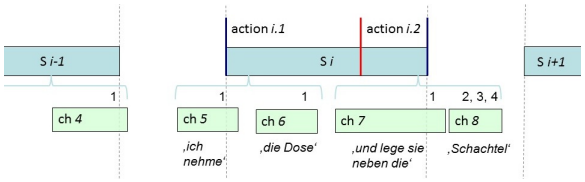


Figure 5: Step 1: Alignment of utterance and action.

In the second step, starting from the last chunk, preceding chunks are merged until either the chunk ends long before the situation starts, or it is already aligned to a preceding situation. The threshold for the difference between endpoint of speech chunk and begin of situation is set to 7 sec, equivalent to temporal bounds in 'working memory'(Miller, 1956). Only if there are two actions within a given situation ('take' together with 'put') – implying that there should be two utterances describing theses actions – for the second utterance the threshold is set to 1.4 sec. In that case, the algorithm is also attentive to continuation words ('and'). When it hits one, it would regard this chunk as the begin of the second utterance. See Figure 6.

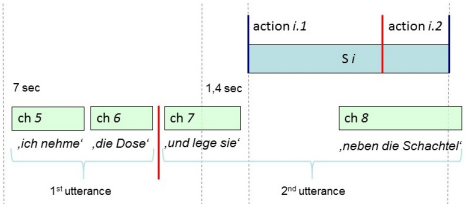


Figure 6: Step 2: Alignment of utterance and action.

This procedure presupposes that speakers tend to start the description of an action before executing it. This presupposition is supported by the empirical facts. The accuracy of the algorithm is high (92% on a subset of the AVC and 86% on Dataset 1 of the MMTD corpus[9]), and it also guarantees that explanations and comments by the test persons that are temporally outside the given bounds and do not pertain to the actions described, are not aligned to actions.

4. Conclusion and Future Work

So far, the presented corpus provides a solid basis for crossmodal and cross-situational word-object and word-action learning, inspired by early language acquisition in children. In addition, we presented an algorithm for segmenting actions and utterances, and aligning the two. The three actions TAKE, PUT, and PUSH can be successfully segmented by the presented heuristics, however, there are limitations as the algorithm is fine-tuned to the given dataset. As we plan to extend the number and complexity of actions, we expect the list of relevant features to segment those actions to be extended. In addition, some aspects of visual actions are currently manually annotated (whether an object is moving and whether a hand touches an object). Work is underway in order to recognize this information automatically. An open issue is also the granularity of actions (e.g., whether the GRASP action preceding a PUSH action should be modeled as part of the PUSH action, or as a separate action) and the effect this has on crossmodal and cross-situational action learning in robots.

5. Acknowledgements

This research is supported by the Vienna Science and Technology Fund (WWTF), project RALLI – Robotic Action-Language Learning through Interaction (ICT15-045) and the CHIST-ERA project ATLANTIS (2287-N35). The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry for Science, Research and Economy.

[8]The dataset (comprising the information described in 1. to 10.) can be downloaded from `http://www.ofai.at/research/interact/avc.html`.

[9]`http://www.ofai.at/research/interact/MMTD.html`

6. Bibliographical References

Abuczki, Á. and Ghazaleh, E. B. (2013). An overview of multimodal corpora, annotation tools and schemes. *Argumentum*, 9:86–98.

Anastasiou, D. (2012). A speech and gesture spatial corpus in assisted living. In *LREC*, pages 2351–2354.

Foster, M. E., Bard, E. G., Guhe, M., Hill, R. L., Oberlander, J., and Knoll, A. (2008). The roles of haptic-ostensive referring expressions in cooperative, task-based human-robot dialogue. In *Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction*, pages 295–302. ACM.

Gaspers, J., Panzner, M., Lemme, A., Cimiano, P., Rohlfing, K., and Wrede, S. (2014). A multimodal corpus for the evaluation of computational models for (grounded) language acquisition. In *Proceedings of EACL Workshop on Cognitive Aspects of Computational Language Learning*, pages 30–37.

Green, A., Hüttenrauch, H., Topp, E. A., and Eklundh, K. S. (2006). Developing a contextualized multimodal corpus for human-robot interaction. In *Proceedings of the 5th International Conference on Language Resources and Evaluation LREC*. Citeseer.

Jayagopi, D. B., Sheiki, S., Klotz, D., Wienke, J., Odobez, J.-M., Wrede, S., Khalidov, V., Nyugen, L., Wrede, B., and Gatica-Perez, D. (2013). The vernissage corpus: A conversational human-robot-interaction dataset. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 149–150. IEEE Press.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. B. (2016). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. *CoRR*, abs/1612.06890.

Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., and Zisserman, A. (2017). The kinetics human action video dataset. *CoRR*, abs/1705.06950.

MacWhinney, B. (2000). The childes project: Tools for analyzing talk: Volume i: Transcription format and programs, volume ii: The database. *Computational Linguistics*, 26(4):657–657.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological review*, 63(2):81.

Nilsson Björkenstam, K. and Wirén, M. (2013). Multimodal annotation of parent-child interaction in a free-play setting. In *Thirteenth International Conference on Intelligent Virtual Agents (IVA 2013)*.

Prankl, J., Aldoma, A., Svejda, A., and Vincze, M. (2015). Rgb-d object modelling for object recognition and tracking. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 96–103. IEEE.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to german. In *In Proceedings of the ACL SIGDAT-Workshop*. Citeseer.

Suanda, S. H., Smith, L. B., and Yu, C. (2016). The multisensory nature of verbal discourse in parent–toddler interactions. *Developmental Neuropsychology*, 41(5-8):324–341.

Tenbrink, T., Eberhard, K., Shi, H., Kuebler, S., and Scheutz, M. (2013). Annotation of negotiation processes in joint-action dialogues. *Dialogue & Discourse*, 4(2):185–214.

EMO&LY (EMOtion and AnomaLY) : A New Corpus for Anomaly Detection in an Audiovisual Stream with Emotional Context

Cédric Fayet[1,2], **Arnaud Delhay**[1,2], **Damien Lolive**[1,2], **Pierre-François Marteau**[1,3]
[1]IRISA - EXPRESSION Team, Lannion&Vannes, France
[2]Université de Rennes 1, Rennes, France
[3]Université de Bretagne Sud, Vannes, France
{cedric.fayet, arnaud.delhay, damien.lolive, pierre-francois.marteau}@irisa.fr

Abstract

This paper presents a new corpus, called EMOLY (EMOtion and AnomaLY), composed of speech and facial video records of subjects that contains controlled anomalies. As far as we know, to study the problem of anomaly detection in discourse by using machine learning classification techniques, no such corpus exists or is available to the community. In EMOLY, each subject is recorded three times in a recording studio, by filming his/her face and recording his/her voice with a HiFi microphone. Anomalies in discourse are induced or acted. At this time, about 8,65 hours of usable audiovisual recording on which we have tested classical classification techniques (GMM or One Class-SVM plus threshold classifier) are available. Results confirm the usability of the anomaly induction mechanism to produce anomalies in discourse and also the usability of the corpus to improve detection techniques.

Keywords: Audiovisual corpus, Acted emotion, Reading task, Anomaly detection, Unbalanced data

1. Introduction

According to (Chandola et al., 2009), "an anomaly is defined as a pattern that does not conform to an expected normal behavior". Anomaly detection systems have been applied to various domains: intrusion detection (Axelsson, 2000), fraud detection (Abdallah et al., 2016), sensor networks (Park et al., 2010), flight safety monitoring (Li et al., 2011) or video surveillance (Ko, 2008).

In the field of human abnormal behavior detection, the focus is put on tasks like crowd modeling, violence detection (Mehran et al., 2009; Gu et al., 2014), human activity detection (Chaquet et al., 2013). In the corpora used or presented in those works, speech and/or face of the subject are usually not available. On the other hand, speech corpora available in the scope of anomaly detection are focused on disease, stress (Hansen et al., 1997) and detection of depression. For instance, the NKI-CCRT corpus (Clapham et al., 2012) has been built to study speech intelligibility before and after cancer. In (Giraud et al., 2013), a corpus containing multimodal expressions of stress during a public speaking task is presented. While this corpus is not dedicated to anomaly detection, it has the advantage to be multimodal, notably including speech and face of the participant. Such a corpus enables to study jointly facial expressions and speech.

When it comes to emotions, some corpora are also available like the Belfast Naturalistic Database (Douglas-Cowie et al., 2003) containing records of people discussing emotive subjects or the EmoTV1 corpus (Abrilian et al., 2005) containing TV interviews bearing naturalistic non-acted emotionally marked data.

All those corpora have been designed and collected to answer to specific scientific questions. As far as we know, no emotional audiovisual corpus exists containing controlled, acted or natural anomalies to address specifically the anomaly detection question.

Consequently, in this paper, we present a new and complementary multimedia corpus called EMOLY which contains human-centered anomalies. The corpus is composed of 41 subjects who read a tale three times. During the reading, we induce a reaction of the subject which can be seen as an abnormal behavior given the context. The corpus gathers various anomalous reactions in terms of intensity, modality (speech or facial expression) or emotion.

The remainder of the paper is structured as follows. Section 2. details how the corpus is designed while section 4. describes the corpus content. Finally, first experiments using the corpus are presented in section 5.

2. Corpus Design

In its first version, the corpus is constituted of 123 records from 41 participants who read three time the French version of the tale "the handless maiden" ("La jeune fille sans main") from Brothers Grimm.

2.1. Participants

Participants are either Master level students from ENSSAT (engineering school affiliated to University of Rennes 1), Licence level students from Lannion IUT or members of the lab (colleagues, PhD students). 41 participants have been recorded including 11 females and 30 males. Participants have an average age of 22 ± 2 years old. Over the 41 participants, 37 are native french speaker and four are non-native french speaker. 12 participants have had opportunities (theatre, representation, ...) to improve their communication skills during their life. Each participant has taken part to three different recording sessions. Participants are not supposed to know that we study anomalous reactions. The task is presented as the constitution of an emotional corpus.

2.2. Recording Process

To get participants accustomed to the task, the tale is sent before the first recording to each participant and the recordings are done during two separate sessions. A printed version is also available in the recording booth to give the possibility to each participant to read it one more time before

the recording session. For all three recordings, the directions given to the subjects remain the same: we ask each participant to read the tale to the camera as if she/he was telling the tale to a child. The purpose is to get expressive speech and facial expressions from them.
The first recording session is anomaly free and is only used to make the speaker more comfortable with the reading task in front of a camera. Before the recording, each participant completed a form which includes identity, personality assessment and a confidential agreement. During a second session, two recordings are carried out. In the first one, we introduce an anomaly, as explained before, to cause a reaction from the speaker. In the second one, we ask the speaker to act the anomaly of his choice. One week delay is kept between the two sessions and the participants do not know what is going to happen during the second session.
To sum up, for each participant, we obtain three recordings of the reading of a slideshow:

1. Introductory record: This is the first time that a participant discovers the slides and reads them in the context of the recording.

2. Induced record: For a particular slide, we use an anomaly induction to trigger a reaction.

3. Acted record: We ask the participant to act an anomaly for a particular slide. No guideline is given by default for this record. In case the participant asks for directions, we give some basic examples.

At the end of the second session, each participant filled in a broadcast agreement and signed a confidential agreement between them and the lab. This agreement is supposed to avoid any revelation of the true purpose of the corpus to the others participants.

2.3. Identity and Personality Assessments

The corpus is focused on humans who read a tale. Reaction(s) to the anomaly inductor is expected to be related to the speaker. If we want to study his/her reaction, we need to gather information about his personality, his speaking abilities and some standard information. In this purpose, the subject needs to fill in two assessment forms at the beginning of the first session.
The first one, called 'Identity assessment', contains questions relative to the genre, age, profession, social origin, geographical origin (birth country, and actual residence with time spent in each of them). We also asked them to evaluate their own oral abilities by answering to several questions about past training, previous jobs or tasks related to oral presentation and speaking abilities. We finally asked them for an auto-evaluation of their oral abilities on a scale from one to six (one is the lowest).
The second called 'Personality assessment' contains questions that can provide a good estimation of the personality of the subject. We choose to use the Big Five model to describe subject personality. The Big Five model has been proposed to describe personality through five personality traits: openness, conscientiousness, extraversion, agreeableness and neuroticism (John and Srivastava, 1999). To evaluate their personality traits each subject completes the BFI-Fr Questionnaire (Plaisant et al., 2010) adapted to french speakers from which a score (with a range between 1 (lowest) to 5 (highest)) is computed for each Big Five's scale. The figure 1 shows the overall distribution of each score by using box-and-whisker plots. The box extends from the lower to upper quartile values of the data, with a line at the median.

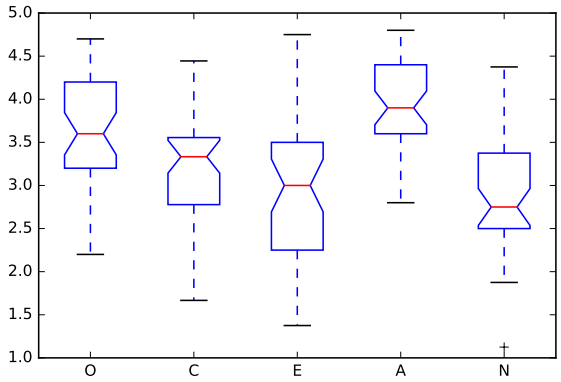


Figure 1: Distribution of the five personality traits (Openness (O), Conscientiousness (C), Extraversion (E), Agreableness (A), and Neuroticism (N)) for the 41 subjects.

2.4. Stimuli

The tale text is manually split into 11 parts, each one is composed of an emotionally consistent part of the story. The different parts are then presented to the subjects as a slide show. Each slide contains the text to read and an illustration in the background (Figure 2). In the normal situation (no anomaly induction), we added a background image related to the textual content to help each participant to act emotionally the content of the slide.
During the second recording, we want to provoke a reaction from the reader (a behavioural anomaly). The reaction needs to be considered as unexpected by someone who watches the record without seeing the slides. To complete this purpose we use an anomaly induction (detailed in section 2.5.).

2.5. Anomaly Induction

Anomaly induction is implemented to produce a variation in the protocol to which the subject is accustomed to. We define three types of anomaly inductions :

- Image : Changing the background with the image of a baby with the head of "Mister Bean", and, introducing some transparency to the text box to increase the visibility of the background.

- Animate : Changing the background with an animated image of the experimenter doing a grimace, and, introducing some transparency to the text box to increase the visibility of the background.

- Text : Rotating the text box multiples time to left and to the right.

Based on these three types of anomalies, we can generate different anomaly inductions, by choosing the slide on

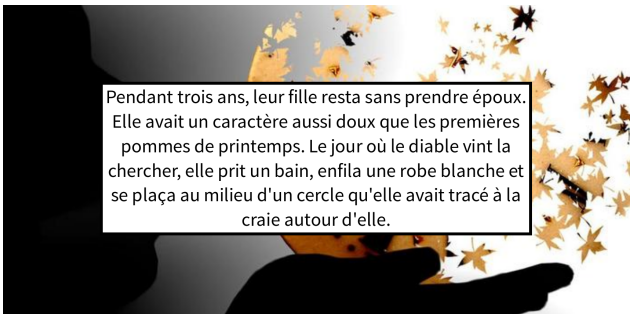


Figure 2: Example of a slide presented to the subjects. This is the sixth slide and it contains a piece of text and an illustration.

which it will be applied and delayed his effect on the slide a certain number of times. An anomaly induction is characterized by his type, the slide in which it will be triggered, and a time delay.

2.6. Hardware

The records have been performed in a recording studio which is composed of two parts: a recording booth and a supervision room. The recording booth is isolated from outside with acoustic insulation materials and a double door. The recording and supervision rooms are separated with soundproofed double glazing so as the supervisor can observe what is happening in the recording room. A green screen was placed in the background of the room from the camera point of view. The video of the face is captured with a webcam (Logitech HD Pro Webcam C920 Webcam). Speech is recorded using a high fidelity headset microphone (DPA 4066-F Omnidirectional Headset Microphone) to ensure high quality in capturing the voice.

3. Annotation

During the second record, with the purpose to induce an anomalous behavior to the participant, we use an anomaly induction described in section 2.5.. And during the third record, we asked to the participant to act an abnormal reaction during the reading of a pre-selected slide. This does not guarantee that the participant will react to the anomaly induction. During the third recording, it is also possible that the speaker either forget to act or act a too weak reaction. Otherwise the reactions can take different forms and intensity. In order to capture the presence and the diversity of anomalies in the recordings, we put an annotation tool in place to get a first human analysis by several annotators.

3.1. Annotation Tool

The annotation tool is a full web service incorporated into the team evaluation web service called PercepEval. The tool has two major screens:

- Timeline annotation. This screen contains three main elements: a video of a subject reading a slide, the slide itself, and an interactive timeline. The interactive timeline gives the possibility to create an annotation by selecting a time range in the video. The number of annotations per video is not constrained. Moreover, two annotation labels can share the same time range.
- Label annotation. To move to the next video, each annotation needs to be filled with a label. This screen contains two lists: one for choosing a label to describe the specific observation that handle the creation of the annotation; and another list, to give an intensity to this observation.

3.2. Annotation Protocol

Before accessing to the first sample, an annotator has access to a page which explains how to use the tool, the context of the record, and what to annotate. The annotator knows that the subject is reading a slide-show of a tale. But he only has access to the slide to annotate. He is asked to annotate reactions that are unexpected during the reading of the slide, by considering voice, facial expression, emotional reaction, and the realization of the task. The slide presented to the annotator does not contain any information about the way that anomaly is produced (acted or induced), and if ever it contains any anomaly.

3.3. Available Labels

The annotator can choose the level in the label hierarchy he wants, to best qualify his observation. The label hierarchy is given below.

- Audio:
  - speech:
    - prosody: rhythm, accent, intonation, intensity;
    - dysfluency: repetition, stuttering;
    - paraverbal: different type of laughing, respiration;
    - different type of silence;
    - timber;
  - others: noise, outdoor event ...
- Emotion: contains sixteen emotions which correspond to the extended list of basic emotions proposed in (Ekman, 2005).
- Video:
  - face: we use the FACS system (Ekman and Friesen, 1978) as label, but, we create a hierarchical representation by regrouping all the Action Units (AU) corresponding to a specific face part into a node (lips, eye, eyebrows, cheek, eyelid, nose, chin, neck, jaw, tongue, neck, glabella);
  - head movement;
  - eye movement;
  - physiology; blushing;
  - others: agitated subject, outdoor event ...
- Task: the reader's words are not link with the text contained in the slide, missing a sentence, sentence is not read at the right time/order

Each annotation is associated with an intensity using 4 values: slight, marked or pronounced, severe, maximum.

		Slide								
		10		9		8		7		
	Time (s)	0	3	0	3	0	3	0	3	
Method	Image	2	2	2	2	0	0	0	0	8
	Text	4	5	5	4	1	0	1	0	19
	Animate	3	1	2	4	2	0	1	1	14
	Total	9	8	9	10	3	0	2	1	41
		17		19		3		3		

Table 1: Number of anomaly inductions used by the experimenter during the building of the corpus. Each anomaly induction is identified by its type, slide and time.

4. Dataset Content

4.1. Anomaly Types

In our corpus, we have induced and acted anomalies. The first type, *induced* anomaly, corresponds to the reaction of the subject during the second recording that may happen when we triggered an event during the reading of the slide. Table 1 contains the number of different anomaly induction methods used during the creation of the corpus.
The second type, *acted* anomaly, corresponds to an abnormal behavior acted by the subject during the third recording. Before the beginning of the third recording, the experimenter asks to the subject to act an abnormal behavior during the reading of a specific slide (the slide is chosen by the experimenter). In total, the slides 7, 8, 9 and 10 have been used, respectively, 3, 3, 18 and 18 times. One subject has acted an abnormal behavior for two different slides, and has been counted twice.

4.2. Annotators

The eleven annotators are members of the EXPRESSION research team at IRISA. The team focuses on studying human language data conveyed by different media: gesture, speech and text. Three annotators have completed each more than twenty annotations, one have completed ten annotations and the others have completed less than six annotations. Each sample has been seen at least by one annotator.

4.3. Annotation of Anomalies

Table 2 presents the number of annotations for induced (by considering each type of anomaly inductor) and for acted samples. Annotations have been grouped into 6 categories: paraverbal, verbal (by splitting the audio node into this two categories), face (by using the node face available in the video node), emotion and task (by using their respective node), and a "nothing" category when we found no annotation for a sample.
First, we notice that annotators have detected nothing unusual for 14 samples, while an anomaly induction was triggered during the reading of the slide. In details, those 14 samples correspond to 5 samples with inductors of the "Image" type, 2 with the "Text" type, and 7 in the case of acted anomaly.
"Image" inductor has been used for eight participants (Table 1). For five recordings out of eight (Table 2), the annotators have noticed no abnormal reaction. This lack of subject's reaction has been noticed by the experimenter during the recording. Consequently, this inductor type has been used less than the others.

	Induced			**Acted**
	Image	**Animate**	**Text**	
Verbal	1 (12.5%) 1.0 1.0 ±0.0	4 (28%) 7 1.75 ±0.83	17 (89%) 1 1 ±0	14 (33%) 24 1.71 ±0.95
Paraverbal	1 (12.5%) 3 1.33 ±0.47	7 (50%) 12 1.72 ±1.16	16 (84%) 58 1.78 ±1.13	28 (69%) 42 1.5 ±0.78
Face	1(12.5%) 1 1 ±0	8 (57%) 15 1.875 ±1.05	16 (84%) 28 1.78 ±1.14	32 (76%) 53 1.65 ±0.87
Emotion	0 0 0	9 (64%) 13 1.44 ±0.68	14 (73%) 25 1.78 ±0.94	12 (28.57%) 17 1.41 ±0.64
Task	1 (12.5%) 3 3 ±0	1 (7%) 1 1 ±0	4 (21%) 4 1 ±0	17 (40.47%) 27 1.58 ±1.23
Nothing	5 (62.5%) 5 1	0 0 0	2 (10%) 2 1	7 (16%) 7 1

Table 2: Number of annotated cues identified by the annotator for the abnormal slide (acted and induced). The anomalous slides are grouped into columns by the type of their anomaly inductor. The rows correspond to label categories. In a cell, the first line corresponds to the number of samples which have at least one annotation and the percentage that it represents compared to all the samples that share the same type of anomaly inductor. The second line corresponds to the total number of annotations that used this category. And the third line corresponds to the mean (and std. dev.) of the numbers of annotations of this category per sample annotated in this same category (*i.e.* line2/line1).

For "Text" and "Animate" induced anomalies, the annotators have tagged abnormal phenomenon by choosing "Verbal&Paraverbal", "Face", and "Emotion" labels for more than half of the sample set. In this case, observable phenomena show that the inductive event had measurably impacted on this three channels. In the case of acted anomaly, we can see that the importance of the "Emotion" category decreases, and the importance of the "Task" category increases. That can be explained by the variety of acted anomalies and by the fact that subjects choose to act an anomaly by doing a variation in their voice or facial expressions, more than doing a variation in the emotion channel. Our interpretation is that completely acting an emotion is more challenging than just doing a variation in the voice or in the facial expression.
By analyzing the annotation, we find that, in general, annotators have done a precise description of their observations. They have chosen leaf labels, instead of choosing preferably general labels at the top of the hierarchy. And they have precisely selected the time range in which the phenomena occur.

4.4. Anomaly Example

Examples of speech signals are given on Figure 3. Speech signals correspond to the same text: "*Implorant le pardon de sa fille, il se mit à aiguiser sa hache*" (Imploring his daughter's forgiveness, he began to sharpen his axe.). The

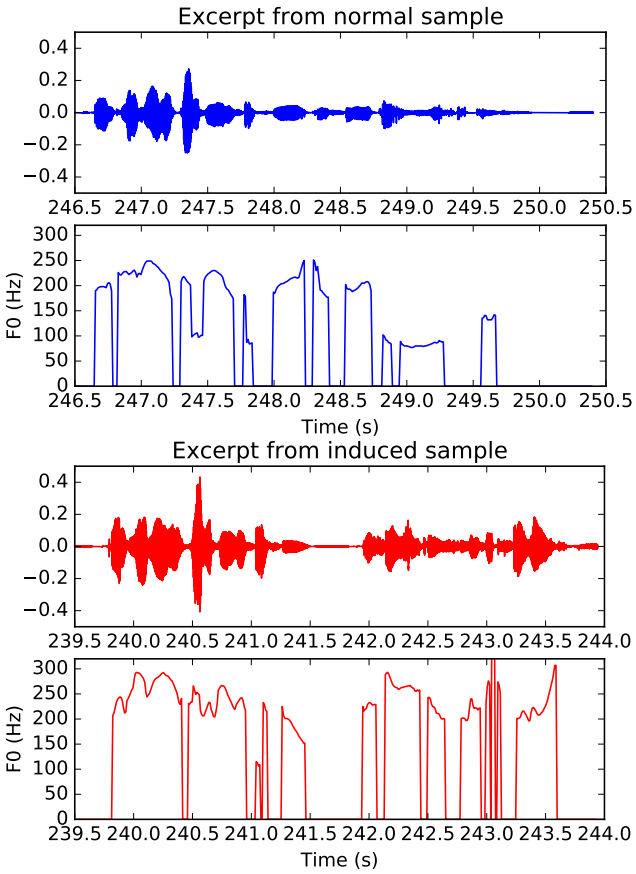


Figure 3: Speech analysis for the reading of the same sentence during the normal session (top, blue curve) and the induced session (bottom, red curve).



Figure 4: Face reaction at the beginning of the slide containing the anomaly induction for the normal session (left) and the induced session (right).

bottom part of Figure 3 contains the speech signal and the F0 contour corresponding to the induced anomaly sample. At the top of Figure 3, the speech signal and the F0 contour are from the same slide but extracted from the normal recording. The induction process on this example makes the speaker laugh while speaking. This explains the big acoustic differences between the two samples.

Figure 4 shows the face of a subject who reacts to an anomaly inductor (right) and the face of the subject for the same sequence during the normal session (left). It clearly shows a smiling expression when the subject discovers the anomaly induction.

4.5. Available Materials

The corpus is composed of three records of 41 subjects. The recording is done through a web-interface using web RTC (chrome implementation) to capture and to align the audio-video signals. The codecs used are VP9 (with a variational fps) for the video encoding and Opus codec (with an audio sampling at 48kHz) for the audio encoding.
After the recording, we check manually the quality of the records. We have used *ffmpeg* to correct mis-alignments and fixed the sampling rate of the video file at 25 fps (with a resolution up to 1080p), and the audio sampling at 44kHz. Finally, the corpus contains 1353 samples, including 83 samples *i.e.* 6% of the total population, where the experimenter either tries to induce an anomalous reaction, or asks to act an abnormal behavior. By considering only the samples that received at least one annotation, we get 68 samples *i.e.* 4.4% of the total population. The samples that have been proposed to the annotation process are either those where an anomaly inductor has been triggered or those for which the subject was supposed to act an anomalous behavior.
Audio, video features and meta-data are available from the team website[1].

5. First Experiment

A priliminary experiment has been conducted using the anomaly detection framework described in (Fayet et al., 2017). This framework is a 2-step pipeline: first an unsupervised classifier which assigns an anomaly score to each sample, and then a threshold classification is used to label each sample as normal or anomalous.
As mentioned in section 4.5., the reading of a slide corresponds to a sample. Each reading either with the induced or acted reaction is considered to belong to the anomalous class and others samples are assigned to the normal class.
Samples are represented by using only audio-based features. Those features are 24 prosodic features, based on low level features like pitch, first two formants, energy and duration of voiced/unvoiced segments with a sliding analysis window size of 40 ms and a step of 10 ms. They are extracted by using the PRAAT software (Boersma and Weenink, 2016). From these low-level features, we derive the final feature vectors that summarize their time evolution by computing the mean, maximum, minimum and entropy for each of them.
We compare here a Gaussian Mixture Model (GMM) and a OneClass-SVM (OCSVM). Hyper-parameters of the models are tuned by using the BIC score for GMM (Steele and Raftery, 2010) and an unsupervised score classifier for OCSVM (Caliński and Harabasz, 1974). For the GMM, the number of components is setup to 10 and each component has its own diagonal covariance matrix. For the OCSVM, we chose the RBF kernel and found, on the training data, ν equals to 0.3 and γ equals to 0.0001.
Results in Table 3 seem to indicate that a GMM and OCSVM approches are able to separate the normal samples from anomalous ones; with an advantage to the GMM

[1] https://www-expression.irisa.fr/results-and-resources/corpus/emoly/

	GMM	OC-SVM
Acted	0.771 $\pm$0.052	0.730 $\pm$0.069
Induced	0.677 $\pm$0.064	0.624 $\pm$0.097
Acted+Induced	0.717 $\pm$0.047	0.683 $\pm$0.056

Table 3: Mean area under ROC curve (ROC-AUC score) and standard deviations.

approach. By comparing the results for acted anomalies to induced ones, we can notice that acted anomaly are better detected through our anomaly chain than induced ones. The reason could be that reactions caused by induced anomalies are more subtle and nuanced, sometimes under control of the speaker, and then more difficult to detect than acted behaviors.

Moreover, by considering the percentage of "Face" labels in Table 2, it seems that the facial expressions, hence the video signals, contain some useful information about the anomalous reactions. So, by using features extracted from the video signal, we expect an improvement in our results.

6. Conclusion and Future Work

In this paper we have presented a new corpus usable to study anomalous behaviors during expressive interactions using both speech and facial expressions. This first version of the corpus contains various kind of anomalies and is composed of records from 41 subjects (11 females and 30 males), totalling about 8.65 hours of records. More information on the corpus is available on the team website[2].

One major objective of this work was to collect and test the reactions of subjects to anomaly induction and also check the ability to automatically detect those anomalies. First experiments seem to show that unsupervised classifiers are usable to separate anomalous samples from normal ones. We confirmed that the protocol is able to induce anomalous reactions from the subject and also that a subject with none or small guidelines is able to act one.

As a next step, it will be interesting to get more annotators and to expand the annotation to all the sample of the corpus. It could enable a comparison between the performance of the annotators and the automatic detection.

7. Acknowledgements

This research has been financially supported by the French Ministry of Defense - Direction Générale pour l'Armement and the région Bretagne (ARED) under the MAVOFA project.

8. Bibliographical References

Abdallah, A., Maarof, M. A., and Zainal, A. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications*, 68:90–113.

Abrilian, S., Devillers, L., Buisine, S., and Martin, J.-C. (2005). EmoTV1: Annotation of real-life emotions for the specification of multimodal affective interfaces. In *Proceedings of the 11th International Conference on Human-Computer Interaction*.

Axelsson, S. (2000). Intrusion detection systems: A survey and taxonomy. Technical report, Chalmers University of Technology, Göteborg, Sweden.

Boersma, P. and Weenink, D. (2016). PRAAT: doing phonetics by computer, December.

Caliński, T. and Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15.

Chaquet, J. M., Carmona, E. J., and Fernández-Caballero, A. (2013). A survey of video datasets for human action and activity recognition. *Computer Vision and Image Understanding*, 117:633–659.

Clapham, R. P., van der Molen, L., van Son, R., van den Brekel, M. W., Hilgers, F. J., et al. (2012). NKI-CCRT Corpus-Speech Intelligibility Before and After Advanced Head and Neck Cancer Treated with Concomitant Chemoradiotherapy. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC)*, volume 4, pages 3350–3355.

Douglas-Cowie, E., Campbell, N., Cowie, R., and Roach, P. (2003). Emotional speech: Towards a new generation of databases. *Speech communication*, 40(1):33–60.

Ekman, P. and Friesen, W. V. (1978). *Manual for the facial action coding system*. Consulting Psychologists Press.

Ekman, P., (2005). *Basic Emotions*, pages 45–60. John Wiley & Sons, Ltd.

Fayet, C., Delhay, A., Lolive, D., and Marteau, P.-F. (2017). Big Five vs. Prosodic Features as Cues to Detect Abnormality in SSPNET-Personality Corpus. In *Proc. Interspeech 2017*, pages 3281–3285. Springer.

Giraud, T., Soury, M., Hua, J., Delaborde, A., Tahon, M., Jauregui, D. A. G., Eyharabide, V., Filaire, E., Le Scanff, C., Devillers, L., et al. (2013). Multimodal expressions of stress during a public speaking task: Collection, annotation and global analyses. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 417–422. IEEE.

Gu, X., Cui, J., and Zhu, Q. (2014). Abnormal crowd behavior detection by using the particle entropy. *Optik-International Journal for Light and Electron Optics*, 125(14):3428–3433.

Hansen, J. H., Bou-Ghazale, S. E., Sarikaya, R., and Pellom, B. (1997). Getting started with SUSAS: a speech under simulated and actual stress database. In *Eurospeech*, volume 97, pages 1743–46.

John, O. P. and Srivastava, S., (1999). *The Big Five trait taxonomy: History, measurement, and theoretical perspectives*, volume 2, pages 102–138. Guilford.

Ko, T. (2008). A survey on behavior analysis in video surveillance for homeland security applications. In *Applied Imagery Pattern Recognition Workshop, 2008. AIPR'08. 37th IEEE*, pages 1–8. IEEE.

Li, L., Gariel, M., Hansman, R. J., and Palacios, R. (2011). Anomaly detection in onboard-recorded flight data using cluster analysis. In *IEEE/AIAA 30th Digital Avionics Systems Conference*, pages 4A4–1–4A4–11. IEEE.

[2]https://www-expression.irisa.fr/results-and-resources/corpus/emoly/

Mehran, R., Oyama, A., and Shah, M. (2009). Abnormal crowd behavior detection using social force model. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 935–942. IEEE.

Park, K., Lin, Y., Metsis, V., Le, Z., and Makedon, F. (2010). Abnormal human behavioral pattern detection in assisted living environments. In *Proc. of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments*, page 9.

Plaisant, O., Courtois, R., Réveillère, C., Mendelsohn, G., and John, O. (2010). Validation par analyse factorielle du big five inventory français (bfi-fr). analyse convergente avec le neo-pi-r. In *Annales Médico-psychologiques, revue psychiatrique*, volume 168, pages 97–106. Elsevier.

Steele, R. J. and Raftery, A. E. (2010). Performance of Bayesian model selection criteria for Gaussian mixture models. *Frontiers of Statistical Decision Making and Bayesian Analysis*, 2:113–130.

Development of an Annotated Multimodal Dataset for the Investigation of Classification and Summarisation of Presentations using High-Level Paralinguistic Features

Keith Curtis[⋆], **Nick Campbell**[†], **Gareth J. F. Jones**[⋆],
ADAPT Centre[⋆†]
School of Computing[⋆], School of Computer Science & Statistics[†]
Dublin City University[⋆], Trinity College Dublin[†]
Dublin, Ireland
Keith.Curtis@dcu.ie, nick@tcd.ie, Gareth.Jones@dcu.ie

Abstract

Expanding online archives of presentation recordings provide potentially valuable resources for learning and research. However, the huge volume of data that is becoming available means that users have difficulty locating material which will be of most value to them. Conventional summarisation methods making use of text-based features derived from transcripts of spoken material can provide mechanisms to rapidly locate topically interesting material by reducing the amount of material that must be auditioned. However, these text-based methods take no account of the multimodal high-level paralinguistic features which form part of an audio-visual presentation, and can provide valuable indicators of the most interesting material within a presentation. We describe the development of a multimodal video dataset, recorded at an international conference, designed to support the exploration of automatic extraction of paralinguistic features and summarisation based on these features. The dataset is comprised of parallel recordings of the presenter and the audience for 31 conference presentations. We describe the process of performing manual annotation of high-level paralinguistic features for speaker ratings, audience engagement, speaker emphasis, and audience comprehension of these recordings. Used in combination these annotations enable research into the automatic classification of high-level paralinguistic features and their use in video summarisation.

Keywords: Data Collection, Annotation, Classification

1. Introduction

Online archives of presentations provide valuable sources of material for study and research. However, such is the amount of this content available in many settings it can be difficult for users to efficiently access material which is most likely to be of interest to them. Search of this content based on the words spoken enables potentially relevant material to be identified based on the topic being presented. However, text-based analysis does not support location of the most important or emphasised material on a particular topic as indicated by the behaviour of the speaker or the audience. Use of high-level paralinguistic features which form part of the presentation offers the potential to identify the most significant topically relevant material within a presentation. Research into the automatic identification of such paralinguistic features and their utilisation in guiding users to relevant material by the use of applications such as summarisation requires the development of suitable corpora to support this work.

Such a dataset must include the relevant audio visual content with suitable manual annotations of the features to be extracted. We describe the construction of such a corpus designed to support this research. Our corpus consists of recordings of paper presentations at the Speech Prosody 7 conference, an academic conference held in Dublin, Ireland in May 2014. The recorded contents include audio-visual content of the presenter, but also parallel recordings of the audience to each presentation.

To investigate the automatic identification of paralinguistic features, these recordings were manually labelled with regions of emphasis by the speaker, ratings of the effectiveness of the speaker, audience engagement with the presentation and their comprehension of its content.

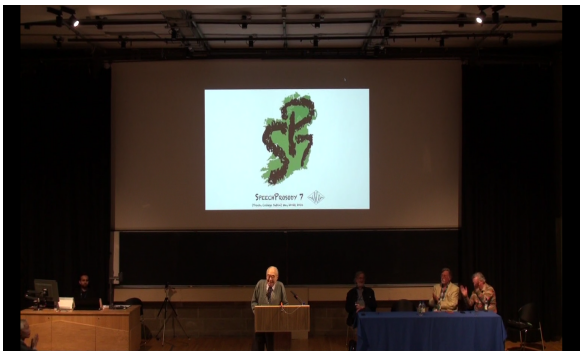


Figure 1: Full camera view of the stage

The remainder of this paper gives further details of the recorded data collection and the processes of its annotation with paralinguistic features.

2. Data Collection

The Speech Prosody 7 conference included a range of presentation types including keynotes, oral presentation of full papers and poster presentation. Our collection consisted of 31 full paper presentations. These were recorded in high quality with fully synchronised recordings of the audience to each presentation. Recordings have a full view of the stage, including the slides used for the presentation. In addition to this, a pdf version of the slides used in each presentation was archived. A total of three fixed cameras



Figure 2: Audience View

were used to record each presentation. Two cameras were fixed within the gallery facing the speaker at the approximate mid-point of the seating structure. One camera was set to record the overall wide-angle view of the whole stage, including the presenter, slides and the surrounding stage area, as seen in Figure 1. The other camera zoomed-in to record the presentation slides in order to provide a back-up to those provided to us by the presenters. The third camera was set up just behind and slightly to the side of the presenter in order to record the audience during each presentation, demonstrated in Figure 2.

After gaining ethical clearance from the host university, all presenters at the conference were asked to give permission for the recording of their presentation(s). Also, all attendees to the conference were asked to give their approval for the recording of the audience to academic presentations.

Recordings were made using three SONY HDR-XR500 cameras. Video was recorded in 1080p at 29.97 fps with an H.264 codec. Audio was recorded in Dolby Digital 48kHz, 16 bit stereo at 256 kbps. The recording standard used was AVCHD.

Presentation recordings were later synchronised by matching presenter recordings and audience recordings using the acoustic footprint. The recorded audio was then synchronised to the video recordings. Recordings were later processed by *Spoken Data* [1], to create full automatic speech transcripts and keywords for each presentation provided. Speech transcripts and keywords were provided by external processing by *Super Lectures* [2].

Videos were processed with an H.264 codec in mp4 format. These processed videos had a frame rate of 25 fps and a bit-rate of 768 kbps. Audio recordings were processed with MPEG-4 AAC audio codec with a sample rate of 44100 Hz and an audio bit-rate of 86 kbps.

The content of the presentations total 520 minutes of conference video, with an average presentation length of a little over 17 minutes. This gave a total of 1040 minutes of video for the presenter and audience recordings requiring human annotation for each paralinguistic concept studied.

3. Human Annotation

Human annotations of the recordings were made to create gold standard labels of the paralinguistic features. The annotated features were: emphasis, speaker ratings, audience engagement and audience comprehension. These features were expected to provide useful information in terms of identifying regions of the presentations which will be of most interest to users of the videos. Annotators were required to watch the content using a specially developed web-based annotation tool and to estimate levels of engagement and comprehension, or in the case of spoken emphasis, to estimate whether or not the content is emphasised. Presentation videos were uploaded to *YouTube*, from which video segments were embedded into the annotation tool.

Annotators for speaker ratings, audience engagement and emphasis tasks were recruited from a pool of research students, support staff and research engineers across Dublin City University and Trinity College Dublin. A total of 40 annotators were equally balanced between native English speakers and non-native speakers. Some, but not all annotators, had prior experience of working with spoken content. Annotation records from each annotator were captured which later allowed us to analyse ratings made by individual annotators. Annotators for comprehension were recruited using a popular crowdsourcing website, further details are given within the description of the comprehension annotation procedure in Section 3.3.

3.1. Speaker Ratings & Audience Engagement

Our annotations made use of the scheme for annotation of conversational engagement provided in (Bednarik et al., 2012). This labels conversational engagement over 6 levels of engagement for window size of 15 seconds. Since the annotation requirements for our dataset differ from their work in that we are marking general audience engagement as they follow an ongoing presentation, we designed our annotation scheme in a similar manner, though using different levels of engagement and window sizes due to the differences in the annotation task.

Our objective was to obtain gold-standard labels for speaker ratings and audience engagement levels. In order to determine suitable lengths of content for annotation we performed a pre-study with a small number of subjects. In this, the subjects were asked to watch a selection of video segments, ranging from 10 seconds up to 50 seconds. Participants were asked to select the best segment length based on time taken to make judgements of engagement levels within the audience, whilst avoiding segments that were too long and thus allowing too much change to occur in engagement levels. If too much change occurs in engagement levels during an annotation segment, estimating the level of engagement will be more inconsistent and less meaningful. The 30-second video segments were selected as the best based on the results of the pre-study. Audience and presenter video segments were of the same length and times, in order to match speaker ratings and audience engagement levels for corresponding video segments.

We asked each annotator to watch 30 second video segments, selected at random from the collection, and to estimate the audience engagement level for this video segment based on an ordinal scale from 1 to 4. Prior to performing the engagement rating, participants were provided with example labelled video segments from each of the 4 engage-

[1] https://spokendata.com/

[2] https://superlectures.com/



Figure 3: Presenter Close-up view

ment levels. Annotators were also requested to provide an estimate of the attendance level at each talk, i.e. how full was the auditorium, estimated on a scale from 1 to 5.

Following this annotation task, participants were asked to watch 30 second video segments of the speaker, selected at random, and to rate the speaker according to their level of agreement with the following statement 'This is a good speaker who is able to capture the attention of the audience and bring the presentation to life.' Annotators were asked to base their judgements on both acoustic and visual stimuli. Human judgements were provided on an ordinal scale from 1 to 8, with 1 being the weakest level of agreement with the given statement and 8 being the strongest level of agreement. Even numbered ordinal scales were chosen for this evaluation in order to force annotators into making a decision rather than selecting the middle option. Views of the presentation slides were excluded from the annotators view in order to ensure that human judgements were based solely on the strengths of the speaker and not on the content.

Each video segment was annotated once, thus further steps were taken to eliminate bias in annotation as explained below. For annotation we followed the assumption, as observed from watching the dataset, that audience engagement levels do not vary much over a short period of time. As each segment was annotated just once for this concept, a number of steps were taken to prepare the data in order to reduce annotator bias and to smooth the annotations, since it was observed that individual annotators can have a tendency to annotate on the high side, or on the low side, when performing annotation tasks.

Outlier Removal The first step involved the removal of obvious outliers from the dataset. Outliers were defined as labels which did not match well with nearby segment annotations. An example would be where a sequence of video segments received an engagement rating of 4, followed by a segment receiving an engagement rating of 1. Outliers were removed to be re-annotated by different annotators from the pool.

Normalisation The next step involved the normalisation of labels to account for annotator bias. This was achieved by analysing ratings for each annotator and applying either a lowering or raising of annotator labels to bring each annotator's ratings in line with other annotations. This was necessary since some annotators were found to have an annotation range from 2 to 5 while others were found to have a range from 3 to 7. By analysing annotations we were able to match these up and lower annotations which were on the high side or increase annotation ratings which were found to be on the low side.

Time Windowing The next step involved time windowing. This was performed in order to smooth annotations and reduce the effect of annotator bias. Video segments were aligned into time windows each 90 seconds in length, i.e. combining three consecutive segments and in steps of 30 seconds. In order to find the label for each 90 second time window, we took the mean of labels for each video segment within that time window. This resulted in annotations for three sequential video segments being combined and averaged.

3.2. Emphasis

The next task was to obtain human annotations for intentionally or unintentionally emphasised speech in our presentations. For annotation of these emphasised parts of audio-visual presentation, we first asked human annotators to watch two five minute clips from an audio-visual presentation and to mark areas of the video where they perceived the presenter to be applying emphasis either intentionally or unintentionally. In order to obtain gold-standard annotations for audience engagement at a fine-grained level, annotators were also asked to watch two 5-minute clips from the audience to different presentations and to estimate audience engagement levels for 6 second video clips. This was to enable investigation of potential correlations between emphasised speech and audience engagement.

Annotation of emphasis over the audio stream only was reported in (Kennedy and Ellis, 2003), where annotators listened to 22 minutes of speech audio and marked utterances which they considered to be emphasised. Annotators were provided with speech transcripts while listening to the audio stream, and marked emphasised points on the transcripts for all sentences they considered emphasised. Since the annotation required in our task differed from this earlier work, in that we are studying emphasis over the audio and visual streams, which had not been studied before, we showed the visual stream to the annotators as they listened to the speech. We considered there to be no need to give annotators the full transcripts as context should already be available to annotators from the visual stream.

A total of 10 annotators were recruited for our speech emphasis annotation task, and were paid 5 euro each after completion of tasks. First, 5 annotators labelled emphasis for two five-minute presentation segments, and engagement levels of the audience for another two five-minute presentation segments. The other five annotators labelled audience engagement for the first two videos and emphasis for the final two videos to ensure that each of the four video segments chosen for this task was labelled for emphasis by 5 annotators and the audience for engagement by five different annotators. This ensured that no annotator labelled audience engagement for the same video that they had already annotated for emphasis. This was to prevent bias during annotation of engagement levels which may occur if the presenter was already aware that they may have labelled for

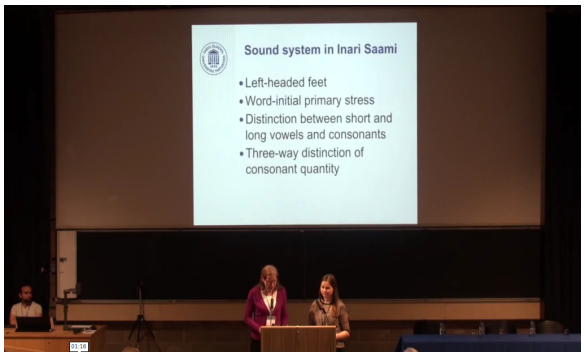


Figure 4: Two Presenters jointly present a talk

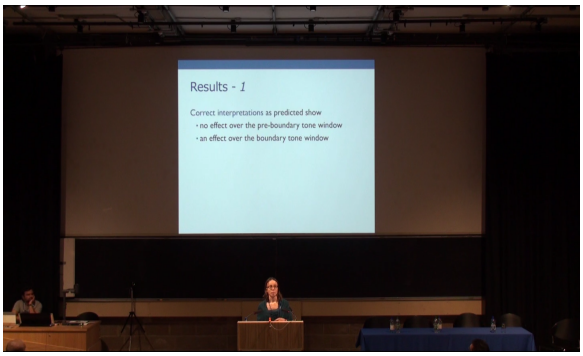


Figure 5: Annotators View

emphasis at a particular point in time.

Following this initial annotation for emphasised speech, it was clear that high levels of disagreement existed between annotators. We considered this to be due to the high level of subjectivity on just what constitutes emphasised speech, and meant that the training of a standard classifier on this data to classify emphasised speech was not practical. Upon further study of areas of agreed emphasis between annotators, a set of potential conditions to satisfy emphasis were constructed and an algorithm developed to identify all possible areas of potentially emphasised speech. All potential areas of emphasis identified by this algorithm were later judged by three human annotators. These sections of the speech data were marked as either emphasised or not emphasised. The mean intra-class correlation between these judgements was calculated as 0.5818, giving us a good level of inter annotator agreement between judges.

3.3. Comprehension

No study of audience comprehension has been reported previously, meaning no existing annotation schemes for tasks of this nature were available to us. Thus, we designed our own annotation scheme based on the information available to prospective annotators.

In order for our dataset to be able to support the study of the concept of audience comprehension within the presentations, we needed to create gold-standard labels for comprehension levels over the dataset. For this task each of the presentations in the dataset was divided into between 4 and 7 contiguous video segments. Each video segment was between 2 and 4 minutes in length. This gave a total of 172 video segments requiring manual annotation.

Annotators were asked to watch each video segment, in order, and to provide a short, written summary of the presentation segment. The purpose of these written summaries was to have them think about the content first before providing their comprehension estimate and also to provide for a method to ensure quality of annotations. Following this, annotators were asked to provide an estimate of how comprehensible they considered the material to be on an ordinal scale from 1 to 8. An even numbered scale was chosen in order to encourage the annotators to make a definite decision on comprehension level rather than choosing a middle, neutral option.

An alternative to using video in this way might be to have required audience members from the presentation to provide information of their perception of the comprehensibility its when attending it. In the absence of this information we consider our annotation method the best approach available.

As stated at the beginning of this section, the comprehension annotators were recruited from a popular crowdsourcing website. Annotators were paid an average rate of 7.50 euro per hour. Recruited human annotators all had English as their first language and all had at least some level of university level education. Annotators each watched contiguous audio-visual segments from one full academic presentation. Each video segment was annotated by at least three annotators and the final gold label was calculated from the average of the three annotations. A total of 93 paid annotators were recruited. The quality of their work was checked before payment was made by studying their provided text summaries and comparing them with their estimated levels of comprehension. For example, if an annotator was unable to provide an accurate text summary of the presentation they had just watched, then it is unlikely they could have a high level of comprehension for that segment. Thus, any high level of comprehension reported by the annotator for that segment could not be taken as reliable.

To calculate the level of inter-annotator agreement for this task we calculated the intra-class correlation model 1, ICC(1,1), over all annotations, which assumes that the annotators rating different subjects are different, being subsets of a larger set of annotators, and chosen at random (Shrout and Fleiss, 1979). The intra-class correlation was calculated using the online ICC calculator available at *Chinese University of Hong Kong* [3]. The mean ICC(1,1) score was found to be 0.6034, which considering the subjectivity of the task at hand we regard as a good level of agreement between judges.

4. Concluding Remarks

In this paper we have described the creation of a multimodal dataset of academic presentations from an international conference and its annotation for the investigation of high-level paralinguistic features to support access to regions most likely to be of interest to a user of this content. While we are unable to release our collection to the wider

[3] http://department.obg.cuhk.edu.hk/researchsupport/IntraClass_correlation.asp

research community, the methods described can be applied to other collections.

The annotation methods described in this paper have proven effective in our work, as shown in the following publications emanating from this work. We demonstrated effective classification of audience engagement in (Curtis et al., 2015). We showed annotation for the identification of emphasised regions of speech to be effective in (Curtis et al., 2017a). We demonstrated classification of audience comprehension during academic presentations to be effective in (Curtis et al., 2016). Finally, we demonstrated the usefulness of all these methods for the summarisation of academic presentations (Curtis et al., 2017b; Curtis et al., 2018), which highly depends on all of the features discussed in this paper.

5. Acknowledgements

This research is supported by Science Foundation Ireland through the CNGL Programme (Grant 12/CE/I2267) in the ADAPT Centre (www.adaptcentre.ie) at Dublin City University. We would like to thank all of our annotators for participating in this study.

6. Bibliographical References

Bednarik, R., Eivazi, S., and Hradis, M. (2012). Gaze and conversational engagement in multiparty video conversation: an annotation scheme and classification of high and low levels of engagement. In *Proceedings of the 4th workshop on eye gaze in intelligent human machine interaction*, page 10. ACM.

Curtis, K., Jones, G. J., and Campbell, N. (2015). Effects of good speaking techniques on audience engagement. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 35–42. ACM.

Curtis, K., Jones, G. J., and Campbell, N. (2016). Speaker impact on audience comprehension for academic presentations. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 129–136. ACM.

Curtis, K., Jones, G. J., and Campbell, N. (2017a). Identification of emphasised regions in audio-visual presentations. In *Proceedings of the 4th European and 7th Nordic Symposium on Multimodal Communication (MMSYM 2016), Copenhagen, 29-30 September 2016*, pages 37–42. Linköping University Electronic Press, Linköpings universitet.

Curtis, K., Jones, G. J., and Campbell, N. (2017b). Utilising high-level features in summarisation of academic presentations. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pages 315–321. ACM.

Curtis, K., Jones, G. J. F., and Campbell, N. (2018). Summarising academic presentations using linguistic and paralinguistic features. In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 2: HUCAPP*, pages 64–73. INSTICC, SciTePress.

Kennedy, L. S. and Ellis, D. P. (2003). Pitch-based emphasis detection for characterization of meeting recordings. In *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, pages 243–248. IEEE.

Shrout, P. E. and Fleiss, J. L. (1979). Intraclass correlations: uses in assessing rater reliability. *Psychological bulletin*, 86(2):420.

BKTreebank: Building a Vietnamese Dependency Treebank

Kiem-Hieu Nguyen
School of information and communication technology,
Hanoi university of science and technology,
1 Dai Co Viet, Bach Khoa, Hai Ba Trung, Hanoi, Vietnam
hieunk@soict.hust.edu.vn

Abstract

Dependency treebank is an important resource in any language. In this paper, we present our work on building BKTreebank, a dependency treebank for Vietnamese. Important points on designing POS tagset, dependency relations, and annotation guidelines are discussed. We describe experiments on POS tagging and dependency parsing on the treebank. Experimental results show that the treebank is a useful resource for Vietnamese language processing.

Keywords: treebank, dependency parsing, POS tagging, word segmentation, Vietnamese, less-resourced language

1. Introduction

Dependency treebank is important for data-driven dependency parsing. However, building a dependency treebank is complicated and expensive.

Dependency treebanks have been available in English and many languages. VnDT (Nguyen et al., 2014) is a Vietnamese dependency treebank which was automatically converted from tree bracketing in VietTreebank (VTB) (Nguyen et al., 2009; Nguyen et al., 2015).

In this work, we present the building of a dependency treebank for Vietnamese[1]. Our treebank was manually annotated by annotators. Its annotation guidelines substantially differ from VTB. Our contributions are two-fold:

- A manual dependency treebank for Vietnamese.
- Experiments on POS tagging and dependency parsing based on the treebank.

The paper is organized as follows: Section 2. briefly introduces related work on building treebanks for Vietnamese and dependency treebanks for other languages. Section 3. highlights important points of annotation guidelines. Section 4. describes in brief the annotation process. Section 5. is dedicated to evaluations and discussions on automatic POS tagging and dependency parsing results. The paper is concluded in Section 6.

2. Related Work

2.1. Treebanks for Vietnamese

VTB was the pioneer treebank for Vietnamese. It has been developed from 2006-2010. It contains manual annotations on about 40K sentences for word segmentation, 10K sentences for POS tagging, and 10K sentences for bracketing.

VnDT contains dependency annotations which were automatically converted from bracketing annotations in VTB. State-of-the-art performance on VnDT is 80.7% and 73.5% on UAS and LAS, respectively (Nguyen et al., 2016a).

Recently, a new treebank for Vietnamese has been developed (Nguyen et al., 2016b; Nguyen et al., 2017). It consists of 40K sentences annotated with word segmentation, POS tagging, and bracketing. While generally agreeing on word segmentation and bracketing, they propose a POS tagset and POS tagging guidelines which focus more on word-class transformation, particularly between verbs and other word-classes. This issue is important as Vietnamese is an analytic language. Unfortunately, their treebank has not been publicly available for research community yet.

2.2. Dependency treebank for other languages

One of the most notable dependency treebanks for English was developed by Stanford NLP group (De Marneffe and Manning, 2008). The Stanford treebank is automatically converted from PeenTreebank phrase structures (Marneffe et al., 2006). Similar approaches were used to build dependency treebanks in other languages such as French, Korean, and Croatian (Candito et al., 2010; Choi et al., 2012; Berovic et al., 2012). Other treebanks are built manually for languages such as Norwegian (Solberg et al., 2014).

The Universal Dependencies is inherited from Penn POS tagset and Stanford typed dependency representation, and has been expanded to many languages (Marneffe et al., 2014; Nivre et al., 2016).

3. Annotation Guidelines

3.1. POS tagging guidelines

Our POS tagset relies on Penn tagset (Santorini, 1990) with the following adaptation to Vietnamese (see Table 2 for the full tagset):

- As Vietnamese is an analytic language, we omit tags related to plurality, tense, and superlative in Penn tagset.
- *CL* is used for noun classifiers. In Vietnamese, a countable noun could be accompanied by a classifier when we want to indicate quantity or simply to emphasize. For example, 'tấm' is a classifier' in "Anh ta giành được hai tấm huy chương vàng" (He won two gold medals); 'chiếc' is a classifier in "Chiếc xe này khá đắt" (This car is quite expensive). In (Nguyen et al., 2016b), the authors also dedicate two tags *Nc* and *Ncs* for noun classifiers. Similar phenomena could be found in other languages such as Korean (Kim and Yang, 2006).

[1] For information on using BKTreebank, please visit `http://is.hust.edu.vn/~hieunk/bktreebank/`

- *PFN* is used for prefix nominalizers. Many nominal expressions in Vietnamese are formed by a leading nominalizer and a verb or an adjective (see Table 1 for examples). In (Nguyen et al., 2016b), there are also POS tags mentioning word-class transformation including VA (Verb-Adjective), VN (Verb-Noun), and NA (Noun-Adjective) but it is not clear from the paper how the tags are designed.

- *NML* is used for phrasal nominalizers. In Vietnamese, a special word such as 'việc' is used as a clausal adverbial marker for a clausal component. For instance, in "Việc xử lý chất thải công nghiệp cần được làm ngay" (The processing of industry garbage needs to be done immediately), 'việc' is the marker for the clausal subject.

- *VA* is used for adjectival verb. In Vietnamese, when the predicate is an adjective, there is no copula verb *to be*. It is hence tagged as an adjectival verb. In the sentence "Tình hình tương đối khả quan" (The situation is[2] quite positive), 'khả quan' is predicate and is tagged as VA.

- *AV* stand for verbal adjective. When a verb modifies a noun, it is tagged as an verbal adjective (e.g. biển/NN quảng_cáo/AV (advertising board)).

- *TO* is used to tagged 'để', which has similar meaning as "in order to" in English.

Prefix nominalizer	Word	Expression
niềm	vui	niềm vui (happiness)
sự	hi sinh	sự hi sinh (sacrifice)
niềm	tin	niềm tin (belief)

Table 1: Examples of prefix nominalizer in Vietnamese.

3.2. Dependency parsing guidelines

Our dependency relations relies on Stanford dependencies (De Marneffe and Manning, 2008) (Table 3). We add two relations for norminalization:

- *case:pfn* is used for nominalizing modifier between a headword as a nominalizer and a verb or an adjective (see examples in Table 1).

- *mark:relcl* is used for phrasal adverbial modifier between a headword as the predicate of the clause and a marker such as 'việc'.

Guidelines for other relations are similar to Stanford dependencies with some modifications. For instance,

- *aux* is also used for relationship between a verb and a tense auxiliary (e.g. thực hiện/VB - aux - đang/MD in "đang thực hiện" (be executing)).

[2]Note that there is no *to be* in the sentence in Vietnamese due to *zero copula*.

POS tag	Description
CD	Cardinal number
DT	Determiner
MD	Modal
NN	Noun
NNP	Proper noun
NML*	Phrasal nominalizer
PFN*	Prefix nominalizer
PRP	Personal pronoun
RB	Adverb
VB	Verb
VA*	Adjectival verb
IN	Preposition
JJ	Adjective
AV*	Verbal adjective
PUNCT	Punctuation
CC	Coordinating conjunction
WDT	Wh-determiner
WP	Wh-pronoun
WRB	Wh-adverb
CL*	Noun classifier
TO	'để' (in order to)
UH	Interjection
FW	Foreign word

Table 2: Our POS tagset (* Tag specific for Vietnamese).

- *det* is also used for relationship between a noun and its plural marker. Here, we tag a plural marker as a determiner (e.g. trường hợp/NN - det - những/DT in "những trường hợp" (cases)).

4. Annotation Process

The raw corpus was collected from Dantri[3], a general-domain online news agency.

Texts were first segmented by UETSegmenter (Nguyen and Le, 2016). Sentences longer than 50 words were removed. Three annotators produced manual POS tagging and dependency parsing using the annotation tool BRAT (Stenetorp et al., 2012).

We decided to annotate POS tagging and dependency in parallel because the two tasks are complimentary to each other. After being explained the annotation guidelines, the annotators were first asked to separately annotate the same small sample dataset. After finishing the sample dataset, they discussed differences and agreed on final decisions.

After being trained, each annotator were asked to annotate separate documents. They discussed with each other when dealing with confusing cases. Every week, the annotators together reviewed and discussed a random annotated document. In the final round, a forth annotator reviewed all annotations and discussed with the annotators in the previous round when necessary to make final decisions.

After removing invalid parsed sentences, our treebank contains 6909 manually annotated sentences on POS tagging and dependency parsing with the average speed of 7 min/sentence.

[3]http://dantri.vn

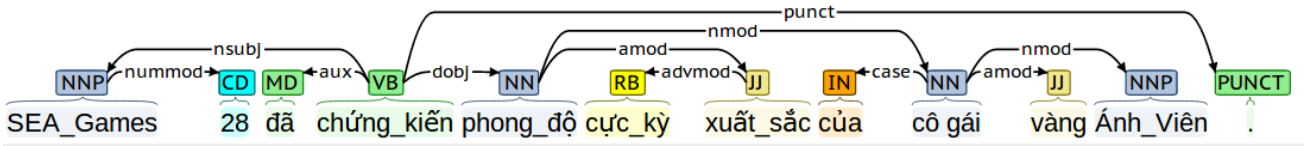


Figure 1: An annotation example in BRAT (SEA Games 28 witnesses an excellent performance from Golden Girl Anh Vien).

Relation	Description
nsubj	Nominal subject
nsubjpass	Passive nominal subject
dobj	Direct object
iobj	Indirect object
csubj	Clausal subject
csubjpass	Passive clausal subject
ccomp	Clausal component
xcomp	Open clausal component
advcl	Adverbial clause modifier
advmod	Adverbial modifier
aux	Auxiliary
cop	Copula
mark	Marker
mark:relcl*	Phrasal nominalizer
nmod	Nominal modifier
appos	Appositional modifier
nummod	Numeric modifier
acl	Adjectival clause
amod	Adjectival modifier
det	Determiner
case:pfn*	Prefix nominalizer
case	Case marking
conj	Conjunct
cc	Coordinating conjunction
punct	Punctuation
dep	Unspecified dependency

Table 3: Our dependency relations (* Relation specific for Vietnamese)

Figure 1 illustrates an annotation example using BRAT. Segmented texts are put into BRAT. Syllables of the same word are connected by '_'. POS tags are labeled for each words.

5. Annotation Evaluations

5.1. Inter annotator agreement

After finishing annotation, the three annotators were asked again to separately annotate the same small dataset to measure Inter-Annotator-Agreement (IAA). Averaged *kappa* is 94.5, 85.2, and 80.4 for POS tagging, unlabeled dependency parsing, and labeled dependency parsing, respectively. Note that IAA was measured for separate annotations of the three annotators without revising of the forth one. Such agreement shows good coherence between different annotators.

5.2. Initial results on POS tagging and dependency parsing

The treebank was divided into a training set of 5639 sentences and a test set of 1270 sentences for learning and testing POS tagging and dependency parsing.

We built a vanilla POS tagging model using CRFSuite[4] implementation of first-order Conditional Random Fields with default hyper-parameters. We used a straightforward feature set as described in Table 4. Our lexicon was built by merging the lexicon of VietTreebank (Nguyen et al., 2006) with frequent tags in our corpus considering important differences in tagging guidelines. Only (word, tag) pairs that were tagged more than three times in the corpus were considered and were reviewed before adding to the lexicon.

Feature set
w[-2], w[-1], w[0], w[1], w[2]
candidate tags
is_head_capitalized
is_all_capitalized
is_numeric

Table 4: Feature set for learning POS tagger with CRF

Tag	P	R	F
NN	92.4	93.6	93.0
IN	89.0	95.0	91.9
MD	97.6	98.3	98.0
VB	89.6	91.1	90.3
VA	58.2	41.6	48.6
CD	89.1	97.7	93.2
RB	84.2	87.0	85.5
CL	85.3	71.1	77.6
AV	59.4	42.3	49.4
PUNCT	99.9	100.0	99.9
JJ	85.9	66.9	75.2
NNP	91.9	94.5	93.2
DT	97.1	94.6	95.8
PFN	73.9	86.7	79.8
CC	92.5	96.3	94.4
PRP	90.7	88.6	89.7
Overall accuracy: 90.7			

Table 5: POS performance by tag

[4] http://www.chokkan.org/software/crfsuite/

We used the transition-based MaltParser (Nivre et al., 2007) with default algorithm and feature set[5] to built a vanilla dependency parser.

Relation	UAS	LAS
ROOT	80.4	80.4
acl	63.4	63.4
advcl	64.7	39.5
advmod	86.8	86.2
amod	89.9	89.4
aux	98.4	97.4
auxpass	98.8	92.8
case	97.5	97.5
case:pfn	100.0	100.0
cc	84.9	84.9
ccomp	77.9	46.8
cl	100.0	100.0
conj	59.9	48.9
cop	95.4	94.2
csubj	75.0	63.9
dep	72.2	72.2
det	97.1	97.1
dobj	92.4	89.2
mark	93.1	93.1
mark:relcl	100.0	100.0
neg	89.6	85.6
nmod	78.2	74.3
nsubj	86.2	79.6
nsubjpass	93.5	75.8
nummod	91.6	89.5
punct	73.9	73.7
xcomp	79.9	70.9
Overall	**84.4**	**81.4**

Table 6: Dependency parsing performance by relation

5.3. Discussions

As shown in Table 5, performance of POS tagging on nouns is similar to averaged performance. Verbs are more difficult to tag as they are ambiguous, not only with nouns and adjectives, but also with verbal adjective (modifiers). Automatic tagging of verbal adjective modifiers is very challenging as such modifiers are not infectional, and in some cases it requires knowledge at syntactic level. They are usually mistakenly tagged as a predicate verb. Verbal adjectives are also difficult because of zero-copula phenomenon.

Dependency parsing performance is promising as shown in Table 6. Parsing at phrase-level is accurate except for nominal modifiers perhaps due to confusing usage of directional and temporal adverbial nouns and prepositions in Vietnamese. On the other hand, parsing at clause-level is poor. There are plenty rooms for improvement on such long-distance dependencies.

6. Conclusion

In this paper, we present the building of a dependency treebank for Vietnamese. Our work is based on previous works on treebanks for Vietnamese and dependency treebanks for other languages. Although current size of the corpus is limited, initial experimental results on POS tagging and dependency parsing is promising.

In the future, we are going to expand BKTreebank with a bootstrapping approach using automatic tagger and parser learned from the dataset. We are going to investigate several approaches to POS tagging and dependency parsing for Vietnamese, including the joint learning approach.

[5] `http://www.maltparser.org/userguide.html`

7. Acknowledgements

This project has been partially funded by VCCorp via collaboration with Data science laboratory, School of information and communication technology, Hanoi university of science and technology. We would like to thank Vu Xuan Luong for enthusiastic discussions on VietTreebank.

8. Bibliographical References

Berovic, D., Agic, Z., and Tadic, M. (2012). Croatian dependency treebank: Recent development and initial experiments. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Candito, M., Crabbe, B., and Denis, P. (2010). Statistical french dependency parsing: Treebank conversion and first results. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Choi, D., Park, J., and Choi, K.-S. (2012). Korean treebank transformation for parser training. In *Proceedings of the ACL 2012 Joint Workshop on Statistical Parsing and Semantic Processing of Morphologically Rich Languages*, pages 78–88, Jeju, Republic of Korea, July 12. Association for Computational Linguistics.

De Marneffe, M.-C. and Manning, C. D. (2008). The stanford typed dependencies representation. In *Coling 2008: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, CrossParser '08, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kim, J.-B. and Yang, J., (2006). *Processing Korean Numeral Classifier Constructions in a Typed Feature Structure Grammar*, pages 103–110. Springer Berlin Heidelberg, Berlin, Heidelberg.

Marneffe, M., Maccartney, B., and Manning, C. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC-2006)*, Genoa, Italy, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L06-1260.

Marneffe, M.-C. D., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal stanford dependencies: a cross-linguistic typology. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on*

Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Nguyen, T. P. and Le, A. C. (2016). A hybrid approach to vietnamese word segmentation. In *2016 IEEE RIVF International Conference on Computing Communication Technologies, Research, Innovation, and Vision for the Future (RIVF)*, pages 114–119, Nov.

Nguyen, T. M. H., Romary, L., Rossignol, M., and Vu, X. L. (2006). A lexicon for vietnamese language processing. *Language Resources and Evaluation*, 40(3/4):291–309.

Nguyen, P. T., Vu, X. L., Nguyen, T. M. H., Nguyen, V. H., and Le, H. P. (2009). Building a large syntactically-annotated corpus of vietnamese. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 182–185, Suntec, Singapore, August. Association for Computational Linguistics.

Nguyen, D. Q., Nguyen, D. Q., Pham, S. B., Nguyen, P.-T., and Le Nguyen, M., (2014). *From Treebank Conversion to Automatic Dependency Parsing for Vietnamese*, pages 196–207. Springer International Publishing, Cham.

Nguyen, P.-T., Le, A.-C., Ho, T.-B., and Nguyen, V.-H. (2015). Vietnamese treebank construction and entropy-based error detection. *Lang. Resour. Eval.*, 49(3):487–519, September.

Nguyen, D. Q., Dras, M., and Johnson, M. (2016a). An empirical study for vietnamese dependency parsing. In *Proceedings of the Australasian Language Technology Association Workshop 2016*, pages 143–149, Melbourne, Australia, December.

Nguyen, Q., Miyao, Y., Le, H., and Nguyen, N. (2016b). Challenges and solutions for consistent annotation of vietnamese treebank. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Nguyen, Q. T., Miyao, Y., Le, H. T. T., and Nguyen, N. T. H. (2017). Ensuring annotation consistency and accuracy for vietnamese treebank. *Language Resources and Evaluation*, Jul.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kubler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Santorini, B. (1990). Part-Of-Speech tagging guidelines for the Penn Treebank project (3rd revision, 2nd printing). Technical report, Department of Linguistics, University of Pennsylvania, Philadelphia, PA, USA.

Solberg, P. E., Skjarholt, A., Ovrelid, L., Hagen, K., and Johannessen, J. B. (2014). The norwegian dependency treebank. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France, April. Association for Computational Linguistics.

GeCoTagger: Annotation of German Verb Complements with Conditional Random Fields

Monica Fürbacher, Roman Schneider
Institute for the German Language (IDS)
R5 6-16, 68161 Mannheim, Germany
{fuerbacher, schneider}@ids-mannheim.de

Abstract

Complement phrases are essential for constructing well-formed sentences in German. Identifying verb complements and categorizing complement classes is challenging even for linguists who are specialized in the field of verb valency. Against this background, we introduce an ML-based algorithm which is able to identify and classify complement phrases of any German verb in any written sentence context. We use a large training set consisting of example sentences from a valency dictionary, enriched with POS tagging, and the ML-based technique of Conditional Random Fields (CRF) to generate the classification models.

Keywords: Grammar and Syntax, Verb Valency, Machine Learning Methods

1. Introduction

Verb complements are indispensable for constructing a correct grammatical sentence in German. The appropriate usage of complements is one fundamental skill for language learners, so the concept of verb complements - or verb valency - is not only an established field of linguistic research, but also often used for didactical purposes. The popular valency dictionary VALBU (=*Valenzwörterbuch deutscher Verben*) (Schumacher et al., 2004) and its expanded online counterpart E-VALBU (=*Elektronisches Valenzwörterbuch deutscher Verben*) (Kubczak, 2009) support both linguists and language learners by providing detailed descriptions of nearly 700 German verbs with more than 3,000 reading variants. Besides other linguistically motivated information, the dictionaries contain authentical example sentences, extracted from DeReKo ((Kupietz et al., 2010)), with a manually added fine-grained markup of verb complement classes.

Unfortunately, compiled dictionaries are naturally limited and cannot cover all possible sentences of a living language and even not the range of all existing verbs. Filling this gap manually seems to be an unpromising task, because it consumes much time and is error-prone. An automatic classification of complements for each verb in any sentence would solve this sophisticated problem. Though we see a remarkable increase of machine learning (ML) tools for natural language processing, we do not know of any empirical approach for the automatic classification of verb complements.

For the development of our ML-based classification algorithm, we compile a corpus of 28,649 example sentences provided by the XML representation of E-VALBU (IDS-Mannheim, 2010) (Müller-Spitzer and Schneider, 2009). The corpus will then be POS-tagged and lemmatized. With this data set, we will train ML-models with different parameters, based on Conditional Random Fields (CRF). The result will not only indicate whether complements can be identified at all by our algorithm, but also prove whether the complements will be correctly classified.

2. Complement classification

The following section briefly addresses theoretical knowledge for the task of classifying verb complements. It covers some linguistic basics regarding complements and gives necessary background information for developing a classification algorithm based on machine learning.

2.1. Linguistic background

A grammatically correct German sentence consists of three main components: the verbal complex, at least one verb complement and the facultative supplements. Those are also called primary components (Zifonun et al., 1997). The verbal complex can consist of a main verb, an auxiliary verb or a modal verb. Each verb determines the number of obligatory complements which are also referred to as actants (Tesnière and Engel, 1980). This characteristic of the verb is called verb valency (Bussmann, 2008) (Schumacher, 1996); for a comparison of English and German see (Fischer, 1997).

Depending on the verb, none or up to four obligatory complements can be required. Thus, there are five different verb valency classes. Moreover, verbs require specific kinds of complements. For German, there exist eight complement classes: the subject, genitive, dative, accusative, adverbial, prepositional, predicative and verbal complement class. However, a sentence does not only contain obligatory complement phrases, but also facultative ones. The latter are often called supplements. While the obligatory complements are dependent on the verb, the supplement's occurrence is usually independent of the verb (Engel and Schumacher, 1978).

This possibility of a complement being facultative leads to some problems. For example, there are bivalent verbs which can occur in a monovalent way like the German verb *essen* (*to eat*) which is declared as bivalent, even if it can occur with only one actant, as demonstrated in the following example (1).

(1a) $[\text{Ich}]_{\text{Ksub}}$ esse.
I eat.

(1b) [Ich]$_{Ksub}$ esse [einen Apfel]$_{Kakk}$.
I eat an apple.

2.2. Machine learning

For solving the task of complement classification, the ML-based technique of Conditional Random Fields (CRF) is used. A CRF is an undirected graphical model which was introduced by Lafferty et al. (2001). It is defined as a linear-chain CRF with the random input variable *x* over observation sequences and the random output variable *y* over label sequences. Based on the fundamental theorem of the random fields, the applied formula for this joint distribution is given in (2).

In this, f_k and g_k are the binary feature functions while θ contains two parameters which are estimated from the training set and by using the improved iterative scaling algorithm. Those feature functions are defined by the means of the transitions between the observation sequences and the states or label sequences.

Furthermore, a graph *G* of the label sequence *Y* is defined as a linear chain whose cliques consist of edges *E=(i, i+1)* and nodes or vertices *V=(1,2, ... m)*. Whereas the edges focus on the transition of the observation sequence and the previous and current labels, the vertices creates the features for current label and the corresponding observation sequence.

(2) $$p_\theta(x|y) \\ exp(\sum_{e\in E,k} \lambda_k f_k(e, y|_{e'} x) \\ + \sum_{v\in V,k} \mu_k g_k(v, y|_{v'} x))$$

CRF is proven for many applications like POS-tagging (Lafferty et al., 2001) (Patel and Gali, 2008), named-entity-recognition (McCallum and Li, 2003) (Watrin et al., 2014), shallow parsing (Sha and Pereira, 2003) or sentence boundary detection (Liu et al., 2005), to name just a few.

3. Algorithm for complement classification

3.1. Data Set

For the dataset construction, we use example sentences from the E-VALBU corpus. E-VALBU is a freely available electronic valency dictionary with valency information for 677 German verbs. The verb selection is based on the vocabulary used for the certification of German as a foreign language by the Goethe Institute. Among other linguistically motivated data, E-VALBU contains information about obligatory and facultative complements by providing (mostly corpus-based) example sentences for each verb (see table 1). We extract these sentences and augment all words with part-of-speech (POS) annotations and lemmatizations. For this task, we choose the TreeTagger tool (Schmid, 2009) with the german-utf-8 tagset, which uses the Stuttgart-Tübingen-Tagset (STTS).

The dataset is then split up into a training file and a test file. The training file contains 80% of the data set, while the test file includes 20%. These values are pitted against the number of sentences (28,659) contained in the data set.

Complement Class	Example Sentences
Subject	**Ich** halte seinen Vorschlag für sehr vernünftig. **Die neue Bluse** steht dir gut.
Accusative	Sie liebt **diesen Mann**. **Ihre Hilfe** wird er annehmen müssen.
Dative	Ich konnte **seinen Worten** nicht immer folgen. Paul hat **mir** ein Buch geschenkt.
Genitive	Er versicherte den Präsidenten **seiner Freundschaft**. Der Zeuge hat sich **seiner** erinnert.
Adverbal	Das Inhaltsverzeichnis steht **am Anfang des Buchs**. Sie fährt **nach Heidelberg**. Die Sitzung beginnt **um drei Uhr**.
Prepositional	Ich denke **an dich**. Das liegt **an dir**. Er hält nichts **von diesem Vorschlag**.
Predicative	Das Wetter ist **schön**. Mein Vater ist **Arzt**. Man nannte ihn **einen Idioten**. Wir hielten ihn für **originell**.
Verbal	Er bedeutet ihr **zu kommen**. Das Medikament beginnt **zu wirken**. Die Untertanen finden, **dass die Steuern gesenkt werden müssen**.

Table 1: German example sentences from E-VALBU.

3.2. Training algorithm

For the model training with Conditional Random Fields, the open source software tool CRF++ is chosen, cf. (Kudo, 2005 2013). The tool's training algorithm is based on a limited-memory BFGS (LM-BFGS).

3.2.1. The feature template

As a first step, a so-called feature template file is created. The template file describes relations between the tokens and therefore determines the number of features. A template file does not only consist of one template but of many templates. This means that every line in a template file is a template by itself and consists of macros. A macro is specified like %x[i,j], whereas *i* is the row and *j* is the column. Table 2 illustrates, how to determine a macro for the sentence *Ich kaufe dir ein Buch.* (*I buy you a book.*) with its respective POS-tags and lemmata. In each row in the first column there is the word itself, the second column presents the POS-tag and in the third column there is the lemma of the original word. Thus, a first macro for a unigram template U1:%x[-2,1] is the POS-tag *PPER* while the second macro U2:%x[-1,1] is *VVFIN* and so on. The relation between these two neighboring tokens can be described in an own macro as U3:%x[-2,1]%x[-1,1]. Each of these macros represents one feature. Almost all tokens in that template file need to be specified as a macro. The only tokens which are not defined are the output tags.

3.2.2. Parameter settings

However, considering too many features overloads the model, so that CRF++ crashes without generating one. To avoid this problem, the parameter settings have to be modified, e.g. the cut-off threshold for the features, the hyperparameter, the shrinking size, the maximum number of

i \ j	0	1	2
-2	Ich	PPER	ich
-1	kaufe	VVFIN	kaufen
0	dir	PRF	du
1	ein	ART	einen
2	Buch	NN	Buch
3	.	$.	.

Table 2: Specifying macros.

iterations during a generating process, the parameter for the termination criterion or the number of used threads. For testing purposes, we generate sixteen disctinct models: (1) Parsedf2, (2) Parsedf2c2, (3) Parsedf3, (4) ParsProc, (5) ParsProcc2, (6) ParsProcc3, (7) ParsProcf2, (8) ParsProcf2c2, (9) Proc, (10) Procc2, (11) Procc3, (12) Procc4, (13) Procc5, (14) Procf2, (15) Procf2c2, (16) Procf3 (please refer to (Fürbacher, 2015) for a detailed description of the models).

For each model, different parameter settings are tested, as shown below. It has to be noted that the number of iterations will not be limited and therefore the parameter -m is not modified. Since the default settings for the shrinking size as well as for the termination criterion work best, these parameters are not changed, too. In addition, four threads are used for all generated models. The only parameters that are changed are the cut-off threshold for the features and the cost-value.

The 16th model utilizes the fewest number of features of all with only 881,352 features. Models number four to six use the most features: 10,771,240. Taking a look at the number of iterations, model number four requires only 250 iterations to be generated. The second, third and eighth models need to iterate most often with more than 400 times. Model number 16 is generated in the shortest time, more precisely within 288.08 seconds. The longest generation time is required for the third model with 826.23 seconds. Nevertheless, all models completed during 5 to15 minutes.

4. Results

The classification result of the models is a binary one, so that the F-measure can be used as a criteria for the model quality. The F-measure is calculated to clarify the following questions:

1. How well can the model distinguish whether a word is a complement or not?
2. How well can the model classify a complement into the correct complement class?

4.1. Results for complement identification

This F-measure also predicates how well the model can identify a word being no complement.

The model with the highest recall, precision as well as the highest F-measure is the first one which was trained with an unreviewed training file. With a recall of 81.3% and a precision of 92.7% it reaches an F-measure of 86.6%. Second best is the 10th model with 79.3%. It also reaches the second best F-measure with 82.4%, even if the precision is 85.7%. The third-best model is the 13th one which has a recall of 79%, a precision of 85.5% and an F-measure of 82.3%.

Furthermore, the cost-value affects the performance of the models. The model result gets worse by increasing the value up to 3.0. Interestingly, by raising the cost-value up to 4.0 and 5.0, the result gets better. However, not all models are optimized by the cost-value. For example, the result of the second model, for which the cost-value is raised up to 2.0, performs significantly worse than the first model with unmodified cost-value. With 77.9%, the recall of the second model is about 3.4 percentage points worse than the first one. Also, the precision reaches only 66.2%, which means it is about 26.5 percentage points lower than the first model. This leads naturally to an F-measure of 69.2%. This implies that the optimal cost-value depends on the training file and thus has to be ascertained separately for each one.

4.2. Results for complement classification

The results of the best models for the task of classifying the complements in their correct complement class are shown in table 3. We state a wide range of recall (.029 - .778) and precision (.1 - .798).

The F-measure shows how well the models can distinguish between certain complement classes. As a consequence, it also indicates which complement classes are difficult to identify.

4.2.1. Subject complement class

The overall results of the recalls for the task of classifying subject complements range from 59.5% to 77.8%, the precision of the models from 67.2% to 79.8%. This leads to an F-measure between 63.1% and 78.7%.

The 11th model has not only the best F-measure of 78.8%, but also the best recall of 77.8% and a precision of 79.8%. Model number nine is the second best with an F-measure of 78.7%, a recall of 77.8% and a precision of 79.6%. The third best one is the 13th model with an F-measure of 78.6%, a recall of 77.6% and a precision of 79.6%.

4.2.2. Accusative complement class

When classifying accusative complements, the recall results range between 45.0% and 64.2%. The lowest precision is 62.0% and the highest value is 66.8%. Thus, the yielded F-measure lies between 52.1% and 64.6%.

The best model is the 12th one. It has a recall of 63.7%, a precision of 65.5% and an F-measure of 64.6%. Second best is model number 10 with a recall of 64.2%, a precision of 64.4% and an F-measure of 64.3%. With a recall of 62.9%, a precision of 65.2% and an F-measure of 64.0%, the 11th model is the third-best one. They all have in common that they comprise all features, and also their cost-value is raised.

4.2.3. Dative complement class

The results of the recall reach from 22.2% to 59.3% for the task of classifying dative complements. The precision results range between 55.0% and 87.2%, and the F-measures vary between 31.6% and 62.7%.

The best model - number (6) - yields an F-measure of 62.7% with a recall of 59.3% and a precision of 66.7%.

For this one, all available features are used and only the cost-value was raised up to 3.0. For the second and third best models, only those features are involved which occur at least twice. The 14th model reaches the second best results with a recall of 41.6%, a precision of 85.1% and an F-measure of 55.9%. The third-best model for classifying dative complement is model number 15, which yields a recall of 43.3%, a precision of 86.5% and an F-measure of 57.7%. Besides the feature parameter, the cost-value is also modified for this model.

4.2.4. Genitive complement class

For the task of classifying genitive complements, the models one to eight fail to detect any genitive.

From the eight models that succesfully identify genitive complements, the recall ranges from 25.3% to 31.3%, while the value for the precision varies between 32.2% and 38.7%. The lowest F-measure is 29.8% and the highest one is 32.7%.

4.2.5. Adverbial complement class

The highest recall result for the task of classifying adverbial complements is 37.8%, while the lowest value is 24.2%. The precision ranges from 41.5% to 51.2%. This leads to an F-measure between 30.8% and 41.2%.

The best model overall is the 14th one, while models 13 and 15 are second and third best. Interestingly, not all features are used by creating the best model, and also the cost-value is not changed. Only while training the 13th model, the cost-value is raised up to 5.0. This implies that in order to create a well-working model for classifying adverbial complements, either not all features should be involved or the cost-value should increase considerably.

4.2.6. Prepositional complement class

The recall results for the classification of prepositional complements range from 33.9% to 63.5%, while the results for the precision range from 53.8% to 60.2%. The resulting F-measures vary between 42.7% and 60.9%.

The 6th model yields the highest F-measure with 60.9% by a recall of 61.6% and a precision of 60.2%. The 11th model shows a higher recall (63.5%) than the 6th one, but also a lower precision (58.3%). The 12th model has the same F-measure as the 11th one, but a recall of 63.1% and a precision of 58.7%. These three models are trained including all available features, but with an increased cost-value of 3.0 respectively 4.0.

4.2.7. Predicative complement class

For the task of classifying predicative complements, the lowest recall result is 32.3%, while the highest one is 39.6%. The worst result for the precision is 65.3% and the best is 81.9%. Hence, the F-measure varies between 38.0% and 51.7%.

With a recall of 38.8%, a precision of 75.5% and an F-measure of 51.7%, model number five is the best one in classifying predicative complements. Second best is model number six, which has a recall of 38.2%, a precision of 76.2% and an F-measure of 50.9%. The 4th model is the third-best with a recall of 36.5%, a precision of 81.9% and an F-measure of 50.5%. These models share the fact of using all available features. Moreover, the cost-value is raised up to 2.0 for the best model and up to 3.0 for the second best.

Complement Class	Recall	Precision	F-score
Subject	.778	.798	.788
Accusative	.637	.655	.646
Dative	.593	.667	.627
Genitive	.283	.387	.327
Adverbal	.341	.512	.409
Prepositional	.616	.602	.609
Predicative	.388	.755	.517
Verbal	.029	.100	.056

Table 3: Best results for each complement class.

4.2.8. Verbal complement class

Due to the fact that only six models are able to classify verbal complement phrases, this task seems to be one of the most difficult ones. Moreover, the six models that classified verbal complements at all yield bad results. The poorest recall is 1.4%, the best one only 2.9%. However, the precision for one model is 1.0% while the other five models have a precision of 100%. This leads to an F-measure between 4.1% and 5.6%.

The two best performing models include all available features and have an increased cost-value of 2.0 respectively 3.0. The third-best model is generated using only features which occur at least twice, but has also a raised cost-value of 2.0.

To sum up, it comes out that our complement classifying algorithms perform well in principle, but would very probably benefit from more training data for specific complement classes.

5. Discussion

All models yield reliable results for the task of classifying whether a word within a freely entered natural language sentence is a complement or not. However, when the models should distinguish between the given complement classes, the F-measure decreases. The corresponding results clearly suggest that the training set should contain more example sentences for the underrepresented verbal complements and genitive complements.

Since the manual tagging of these complement classes is both time consuming and indispensable, the further extension of our gold standard training corpus will probably be beneficial for the linguistic community. We are firmly convinced that this effort will result in even better recall/precision values for the automatic assignment of yet underrepresented complement classes.

6. Concluding Remarks

We presented a well-working ML-based algorithm for the identification of verb complements within any German sentence as well as for the annotating of complement classes. For that purpose, a specific data set was created, with POS-tagged and lemmatized example sentences for different complement types. Applying Conditional Random Fields



Figure 1: *GeCoTagger* Output

with the CRF++ toolkit, various models were generated with respect to feature-dependent parameter settings (see also (Fürbacher, 2015)).

We demonstrated that even relatively small amounts of natural language data – the valency dictionary used for our training runs contains about 700 verb lemmata with an example corpus of only some thousand annotated sentences – can constitute a sound basis for machine learning, given that they contain reliable, scientifically grounded, fine-grained information. Fur the future, we think of enhancing our application with a user feedback function, so that complement classifications that are manually evaluated as correct or wrong would contribute to further improvements of the classification model.

Furthermore, our mostly positive evaluation results led to the development of a web interface prototype, which is called *GeCoTagger* (=*German Complement Tagger*). It will be freely available for the linguistic community within the GRAMMIS information system (IDS-Mannheim, 2018) (Schneider and Schwinn, 2014) and allows users to enter natural language sentences, and to receive an analysis of its verb complements.

Figure 1 presents an online example for the classification output of the sentence *Drei Affen schenken dir eine Banane.* (*Three monkeys give you a banana.*), where each word of the sentence is coloured according to its complement class in E-VALBU. Since the user input is always pre-processed by Treetagger, POS-tags and lemmata can be added easily to the output.

7. Bibliographical References

Bussmann, H. (2008). *Lexikon der Sprachwissenschaft*. Kröner, Stuttgart.

Engel, U. and Schumacher, H. (1978). *Kleines Valenzlexikon deutscher Verben*. Number 31 in Forschungsbericht des Instituts für deutsche Sprache. Gunter Narr Verlag, Tübingen.

Fischer, K. (1997). *German-English Verb Valency: A Contrastive Analysis*. Tübinger Beiträge zur Linguistik. Narr.

Fürbacher, M. (2015). *Developing of an ML-Based Algorithm for Typing of Complement Phrases*. Master Thesis, University of Trier, Computational Linguistics.

IDS-Mannheim. (2010). *Das elektronische Valenzwörterbuch deutscher Verben*. http://www.ids-mannheim.de/e-valbu/.

IDS-Mannheim. (2018). *Grammis - Grammatisches Informationssystem*. https://grammis.ids-mannheim.de, DOI: 10.14618/grammis.

Kubczak, J. (2009). Hier wird Ihnen geholfen! Das elektronische Valenzwörterbuch deutscher Verben. In *Sprachreport 04/2009*, pages 17–23.

Kudo, T. (2005 – 2013). *CRF++: Yet Another CRF Tool Kit*. Version 0.58, https://taku910.github.io/crfpp.

Kupietz, M., Belica, C., Keibel, H., and Witt, A. (2010). The German reference corpus DeReKo: A primordial sample for linguistic research. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Lafferty, J. D., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, pages 282–289.

Liu, Y., Stolcke, A., Shriberg, E., and Harper, M. (2005). Using conditional random fields for sentence boundary detection in speech. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 451–458.

McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*, pages 188–191.

Müller-Spitzer, C. and Schneider, R. (2009). Ein XML-basiertes Datenbanksystem für digitale Wörterbücher - Ein Werkstattbericht aus dem Institut für Deutsche Sprache. *it-Information Technology*, 51(4):197 – 206.

Patel, C. and Gali, K. (2008). Part-of-speech tagging for gujarati using conditional random. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*, pages 117–122.

Schmid, H. (2009). *TreeTagger - a part-of-speech tagger for many languages*. http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.

Schneider, R. and Schwinn, H. (2014). Hypertext, Wissensnetz und Datenbank: Die Web-Informationssysteme grammis und Progr@mm. In Franz Josef Berens et al., editors, *Ansichten und Einsichten. 50 Jahre Institut für Deutsche Sprache*, pages 337–346. IDS, Mannheim. http://nbn-resolving.de/urn:nbn:de:bsz:mh39-24719.

Schumacher, H., Kubczak, J., Schmidt, R., and de Ruiter, V. (2004). *VALBU - Valenzwörterbuch deutscher Verben*. Number 31 in Studien zur Deutschen Sprache. Narr, Tübingen.

Schumacher, H. (1996). Satzbaupläne und Belegungsregeln im Valenzwörterbuch deutscher Verben. In *Lucien Tesnière - syntaxe structurale et opérations mentales: Akten des deutsch-französischen Kolloquiums anläßlich der 100. Wiederkehr seines Geburtstages, Strasbourg 1993*, pages 281–294. Niemeyer, Tübingen.

Sha, F. and Pereira, F. (2003). Shallow parsing with conditional random fields. In *Proceedings of HLT-NAACL 2003*, pages 134–141.

Tesnière, L. and Engel, U. (1980). *Grundzüge der strukturalen Syntax*. Klett-Cotta, Stuttgart.

Watrin, P., de Viron, L., Lebailly, D., Constant, M., and Weiser, S. (2014). Named entity recognition for German using conditional random fields and linguistic resources. In *Proceedings of the 12th KONVENS (Konferenz zur Verarbeitung natürlicher Sprache)*, pages 153–156.

Zifonun, G., Hoffmann, L., and Strecker, B. (1997). *Grammatik der deutschen Sprache: Bd. 1-3*. de Gruyter, Berlin.

AET: Web-based Adjective Exploration Tool for German

Tatiana Bladier, Esther Seyffarth, Oliver Hellwig, Wiebke Petersen
Heinrich Heine University of Düsseldorf, Germany
bladier@phil.hhu.de, seyffarth@phil.hhu.de, hellwig7@gmx.de, petersen@phil.hhu.de

Abstract

We present a new web-based corpus query tool, the Adjective Exploration Tool (AET), which enables research on the modificational behavior of German adjectives and adverbs. The tool can also be transferred to other languages and modification phenomena. The underlying database is derived from a corpus of German print media texts, which we annnotated with dependency parses and several morphological, lexical, and statistical properties of the tokens. We extracted pairs of adjectives and adverbs (*modifiers*) as well as the tokens modified by them (*modifiees*) from the corpus and stored them in a way that makes the modifier-modifiee pairs easily searchable. With AET, linguists from different research areas can access corpus samples using an intuitive query language and user-friendly web interface. AET has been developed as a part of a collaborative research project that focuses on the compositional interaction of attributive adjectives with nouns and the interplay of events and adverbial modifiers. The tool is easy to extend and update and is free to use online without registration: http://aet.phil.hhu.de

Keywords: Corpus-query tool, adjectives, adjective modification, German, syntactic dependencies

1. Introduction

Current research on semantic compositionality, relatedness, clustering and distributional properties of adjectives and adverbials in German (Eroms, 2011; Hartung and Frank, 2011; Dalmas et al., 2015; Petersen and Hellwig, 2016) requires access to corpus tools which enable linguists to answer complex questions about the behavior of the adjectives in their context. These questions may, for instance, relate to the syntactic relationships or morphological properties of the adjectives at hand. Currently available online corpus query tools for German include the Stuttgarter IMS WORKBENCH tool (Evert and Hardie, 2011), COSMAS II (Kupietz and Keibel, 2009) and DWDS (Klein and Geyken, 2010). These and other resources provide data that is annotated on several linguistic levels, but do not lend themselves specifically to research on adjectives. In particular, we are not aware of any freely accessible corpus query tools that offer the possibility to easily filter the corpora according to complex syntactic or morphological (e. g. derivational) criteria.

In this paper, we present a user-friendly, web-based corpus query tool designed for linguists researching different aspects of usage patterns of adjectives and adverbs in German. The underlying corpus of our Adjective Exploration Tool (AET) is based on a subset of the corpora from the Workshop on Machine Translation 2014 (WMT14) shared tasks (Bojar et al., 2014)[1] and contains texts from German print media. In addition to the corpus text itself, the AET database also contains data on syntactic dependencies, word co-occurrences, morphological properties of the occurring tokens, and frequency statistics for the adjectives and adverbs. We store dependency-parsed sentences in a relational MySQL database, capturing direct modification[2] relations between the *modifiers* (i.e., adjectives or adverbs) and the *modifiees* (i.e., nouns, verbs, or adjectives modified by adjectives or adverbs). AET currently contains over 28 million tokens in roughly 8 million sentences[3], from which we extracted and stored around 13 million adjective-modifiee pairs.

Users of AET can query the underlying database using a web interface and a query language. This lets them define morphological, syntactic, lexical, character-based, or statistical criteria to retrieve all samples of particular modification or co-occurence patterns, together with the sentences in which they occur in the corpus. The results from such a query can be downloaded as a `.csv` (comma-separated values) file to enable offline work and further processing of the retrieved items.

AET was originally designed in a research project concerned with modeling the compositional interaction of attributive adjectives with nouns[4] and a project concerned with the interplay of events and adverbial modifiers[5]. However, the structure of the AET database makes it easy to extend the tool to different research areas and languages or to add more corpora. AET is available freely without registration at http://aet.phil.hhu.de.

1.1. Why AET?

In order to analyze the modification patterns of German adjectives and adverbs, it does not suffice to look for simple surface-level co-occurrences in a corpus. While adjectives and the tokens they modify may occur directly adjacent to each other in German sentences, they are also often found far apart and in different orders, as shown in examples (1) to (3). This is why information on syntactic dependencies is necessary in order to identify the modification behavior independently of the order or relative distance in which the participants of that modification are observed.

[1] The file is available at http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2013.de.shuffled.gz.

[2] In this paper, we use the term *modification* as an umbrella term for the syntactic relationships in which adjectives or adverbs are involved, including *attribution* and *predication*. For more discussion on this, see (McNally and Boleda, 2004).

[3] When processing the original corpus contents, we removed the sentences that did not contain any modifier-modifiee pairs, since these sentences are not of interest in the context of AET.

[4] http://www.sfb991.uni-duesseldorf.de/en/c10/

[5] http://www.sfb991.uni-duesseldorf.de/en/b09/

(1) Der Kuchen **schmeckt$_{V}$** **gut$_{ADJ}$**.
the cake tastes$_{V}$ good$_{ADJ}$

(2) Er sagt, dass der Kuchen **gut$_{ADJ}$** **schmeckt$_{V}$**.
he says that the cake good$_{ADJ}$ tastes$_{V}$

(3) Der Kuchen **schmeckt$_{V}$** wegen der verwendeten
the cake tastes$_{V}$ due_to the used
Zuckerart **gut$_{ADJ}$**.
sugar_type good$_{ADJ}$

In all three examples above, the adjective *gut* (Engl. *good*) modifies the verb *schmeckt* (Engl. *tastes*). The search language provided by AET can be used to retrieve all modification pairs that include the lemma *gut* as the modifier, or the lemma *schmeckt* as the modifiee; but it can also answer other types of questions, for example:

- Which denominal adjectives occur in the corpus?
- Which adjectives never occur in inflected forms?
- Which adjectives with a particular suffix occur at least 50 times in the corpus?

Specific questions like these can only be answered with a corpus query tool if that tool enables user-defined filtering with regard to several different kinds of linguistic information. AET provides this possibility in a fast and user-friendly way.

1.2. Related Work

In recent years, several corpus query tools have been made available with the aim of studying the collocations of certain lemmas, the most popular of which are COSMAS II (Kupietz and Keibel, 2009), DWDS (Klein and Geyken, 2010), and the IMS OPEN CORPUS WORKBENCH (CWB) (Evert and Hardie, 2011). In this section we will briefly compare AET with the mentioned tools.

COSMAS II and DWDS contain large amounts of data for German from different genres and centuries, annotated with several tagsets. While these tools can display a number of different morphological or syntactic observations about tokens in the corpora along with the search results, they do not provide an easy way for the user to include these facts in the search query.

CWB is a corpus analysis architecture developed at the University of Stuttgart. It can be used in combination with any corpus and uses CQL (Corpus Query Language) for user queries. CWB allows users to search for words of a certain part of speech which occur in a context of certain window size or within a single syntactic constituent. However, it cannot be used to query the data with regard to morphological information on the derivation processes of lemmata (i.e., deverbal, denominal, deadjectival, etc.) or to the possible syntactic positions of the adjectives, and CQL queries quickly become long and complex.

Since none of the tools described above provide the possibility to query for the properties we have mentioned, the example questions given in Section 1.1. cannot be answered using the already existing tools. AET aims at filling this gap in the available resources.

2. Data Sources and Processing

The AET corpus is derived from a German subset of the WMT14 corpus (Bojar et al., 2014) that contains samples from German online newspapers. We used the MATE parser (Björkelund et al., 2010) to annotate the corpus with syntactic dependencies, part-of-speech (POS) tags, and lemma identification. MATE was chosen because it is among the best-performing tools for German with respect to accuracy (Choi et al., 2015). Figure 1 shows an example sentence parsed with MATE[6].

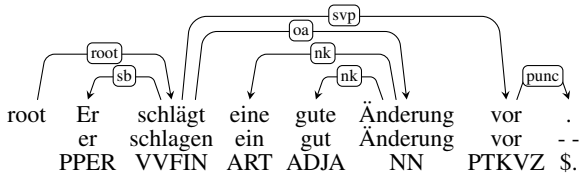


Figure 1: Dependency tree visualizing the output of MATE

We added morphological data for the tokens in the corpus from the two databases CELEX (Baayen et al., 1995) and DERIVBASE (Zeller et al., 2013) to enhance the morphological output from MATE. These databases contain detailed information on the affixation and derivation of the lemmata and word forms; CELEX additionally contains frequency counts[7]. Since CELEX provides no options to analyse out-of-vocabulary words, we use DERIVBASE as a fallback solution for those cases.

2.1. Qualitative Evaluation of the Input Data

The architecture of the database behind AET was designed to store data output from different dependency parsers. This makes the recognition of modification pairs dependent on the parser performance.

Although the MATE parser is known to be one of the best state-of-the-art parsers for German, we encountered several cases of erroneous analyses provided by it. The main source of incorrect parsing analyses are sentences in which several adjectives modifying the same token are interrupted by punctuation marks or conjunctions. Figure 2 shows a case of an erroneous analysis resulting from the punctuation mark between the adjectives in the adjective chain. Long chains of adjectives with different modification relations (for example, with both adverbial and adjectival use of adjectives) also lead to erroneous parses.

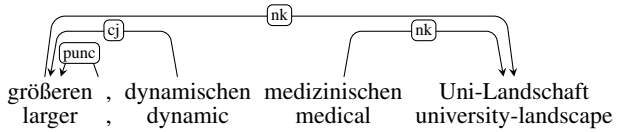


Figure 2: Erroneously recognized dependencies in MATE

[6]Morphological analyses and semantic role labeling are part of the MATE output, but are not shown here.

[7]The frequencies provided in CELEX are those observed in the Mannheim corpus of the Institut für deutsche Sprache. For more information, see (Gulikers et al., 1995).

We also encountered cases in which dependency relations provided by MATE systematically do not correspond to the correct modification relations. Among such cases are predicative uses of adjectives ("*They found the cat well-fed*") and the cases in which an adjective follows after a copula verb ("*The cat is well-fed*"). In both cases MATE recognizes the dependency relation between the verb and the adjective, but does not recognize the modification relation between the noun and the adjective. Elimination of such systematic errors requires improving the performance of the parser, a task which goes beyond the scope of the AET tool.

3. How AET Works

The main goal of AET is to let users retrieve specific types of corpus samples without needing any programming skills. A MySQL database stores the processed corpus contents, and a web interface developed with CakePHP lets the user enter queries in an easy-to-understand query language that handles the interaction with the database. A query translation module handles the generation of SQL expressions based on the user input. The following subsections describe the architecture of the database, the query language and its translation to SQL, and how the tool can be used to retrieve modifier-modifiee pairs according to user-specified criteria.

3.1. Relational Database

We created a MySQL database to store the processed data from the original corpus. The database contains information on individual tokens and lemmata in the corpus (such as morphology, word type, gradation, frequency, etc.), as well as information on the adjective-modifiee pairs that are found in the sentences. Table 1 shows the searchable fields stored for the adjective *vertretbaren* (Engl. *justifiable*).

Searchable field	Value
word form	vertretbaren
lemma	vertretbar
word type	adjective
derivation type	deverbal
derivation scheme	Vx
prefix	ver
suffix	bar
derivation tree	(((ver)[V|.V],(tret)[V])[V], (bar)[A|V.])[A]
previous derivation step	vertret+bar
composition	non-compositum
gradation	positive
frequency in AET	62
number	plural
case	dative
gender	feminine
never inflected	no
occurs sentence-final	yes

Table 1: Searchable fields in AET for the word form *vertretbaren*. The user can search for either the word form or the lemma.

Table 2 shows the searchable fields in the database for adjective-modifiee pairs, shown by the example pair *benötigten Mittel* (Engl. *necessary funds*).

Searchable field	Value
modifier word form	benötigten Mittel
modifiee word form	Mittel
modifier lemma	benötigt
modifiee lemma	Mittel
modifier POS tag	ADJA
modifiee POS tag	NN
POS category pair	AN
pair frequency	56
precedence	yes

Table 2: Searchable information for the adjective-modifiee pair *benötigten Mittel*. The user can search for either surface form or lemma pairs.

Our intuitive query language enables users to interact with this (relatively complex) database structure. We have optimized the database structure in order to avoid long waiting times while the results are being collected from the server.

3.2. The Query Language of AET

One of the main criteria guided the design of AET was that the tool should be easy to use, particularly for researchers with no background in programming or computer science. The user input is parsed and turned into the corresponding SQL expression by the query translation module, which consists of a script written in Python 3.6. The relations between the search terms of the query language on the one hand and the database structure on the other hand are defined in an easy-to-edit configuration file in `.yaml` format. Therefore, search terms can be added or changed by the administrator by simply editing that configuration file.
Some of the available search terms are presented in Table 3 to give an overview of the types of queries that can be formed. Since the intended function of AET is to retrieve information about modifier-modifiee pairs, the results of each search query are always sorted with respect to pairs. As an example, consider the query in (4). It will return all modifier-modifiee pairs from the database that contain a modifier that is some form of the lemma *essbar* (Engl. *edible*).

(4) `modifier_lemma(essbar)`

After the query is submitted, the query translation component generates the corresponding SQL query for the database search. The resulting expression is this:

```
SELECT DISTINCT amtokenpair.id
FROM token_pairs amtokenpair
JOIN lemmas l1
ON amtokenpair.mer_lemma_id = l1.id
WHERE l1.lemma = 'essbar';
```

The Boolean operators AND, OR, and NOT can be used to build complex queries using the search terms that are available. Brackets can be used to indicate operator precedence between subqueries; where they are absent, natural precedence is assumed. The query in (5) is an example of the way search terms can be combined to build more specific search queries. It will retrieve all pairs from the database which

Search term	Explanation
`modifier_lemma(blau)`	Single-word modifier lemma to look for, e.g. *blau*.
`modifiee_lemma(Haus)`	Single-word modifiee lemma to look for, e.g. *Haus*.
`modifier_wordform(rotem)`	Single-word modifier word form to look for, e.g. *blaues*.
`modifiee_wordform(Kreuzen)`	Single-word modifiee word form to look for, e.g. *Kreuzen*.
`modifiee_pos(v)`	Search for modifiees of the specified part of speech, e.g. *verb*.
`modifier_lemma_starts(ab)`	(Character) prefix of a modifier lemma to look for.
`modifier_wordform_starts(ver)`	(Character) prefix of a modifier word form to look for.
`modifiee_wordform_starts(ent)`	(Character) prefix of a modifiee word form to look for.
`modifier_lemma_ends(lich)`	(Character) suffix of a modifier lemma to look for.
`modifier_wordform_ends(ende)`	(Character) suffix of a modifier word form to look for.
`modifier_derivationtype(deverbal)`	Derivation type of the modifier as analysed by the parser.
`modifier_never_inflects(true)`	Only show pairs in which the modifier is never inflected.
`pair_type_frequency_greater(100)`	Frequency filter for the pair types.

Table 3: Selection of search terms available in AET (for the full list, see documentation on website)

fulfill the constraint of containing both a non-deverbal modifier and a modifiee that is a form of the lemma *Mann*:

(5) `(NOT modifier_derivationtype(deverbal)) AND modifiee_lemma(Mann)`

Many search terms, including those listed in Table 3, have several possible spellings, in order to make it easier to remember the terms. For instance, the alternative spellings for the search term `modifiee_wordform_starts(ver)` include `mee_wf_starts(ver)` and `mee_wordform_starts(ver)`. A more detailed documentation of the query language is available on the AET website.

3.3. Searching With AET

Figure 3 shows a screenshot of the AET query input field, including a sample query. The tab labeled `Documentation` lists all available search terms when clicked. The `Submit` button sends the query to the server, which handles the translation to SQL and the collection of results from the database.

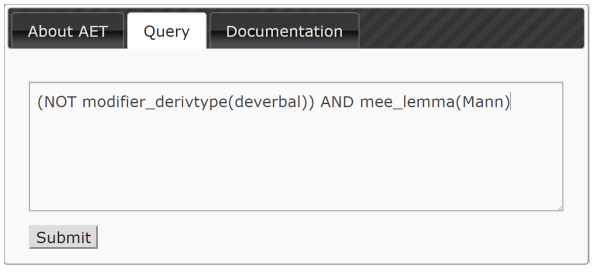


Figure 3: Query input field

Figure 4 shows an extract of the results as displayed on the website. The results are grouped by the modifier and modifiee lemmata that occur in each pair. Clicking on a lemma pair expands that section to show all sentences in the corpus that contain the given lemmata.
Clicking the `Export results` button lets the user download a `.csv` file that contains all results from the query and can be used for offline work and further processing steps.

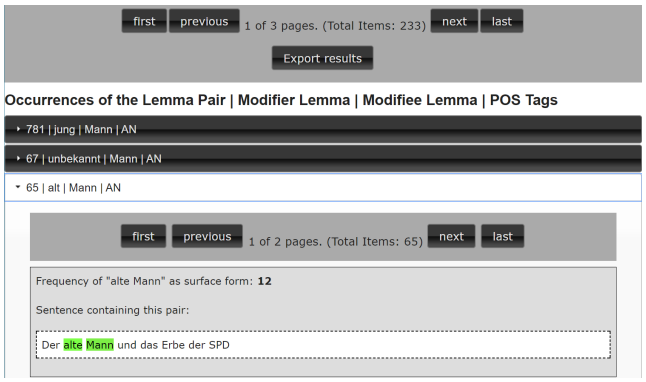


Figure 4: Representation of the search results

4. Conclusion and Future Work

AET was developed to aid linguistic researchers who are interested in the behavior and co-occurrence patterns of German adjectives, adverbs, and the words they modify. The current version of the tool provides a number of ways to filter and extract modifier-modifiee pairs from the corpus. We now outline some directions for further development.
One major aspect of the behavior of adjectives with regard to the adjective ordering has not been touched upon yet by our representation – namely, the analysis of *adjective chains*. Adjective chains are sequences of two or more adjectives modifying the same token in the sentence or each other (Dye et al., 2017). For an example, see Figure 5. Regarding the inclusion of such chains in AET, the central challenge is to find an appropriate mode of presentation for chains of different structures, lengths, and argument orders. We have explored the option of extending the database by adding more corpora in order to increase the diversity of the

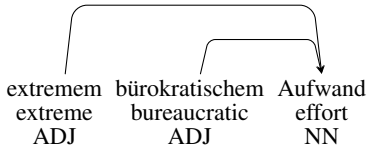


Figure 5: Example of an adjective chain

genres covered by the tool. The structure of AET makes this easy in theory, with the main practical limitations being the availability of large corpus files and the execution time of individual queries as the database grows.

We are also interested in creating a version of the tool that lets the user search for modifier-modifiee pairs in other languages. As long as inflected languages are being processed, our database structure is independent of the language of the corpus and can be reused with minimal changes to the architecture.

The tool we have developed is not restricted to adjectives and adverbs. It can also easily be used to store and query other modifications, such as nominal compounds or preposition collocations.

Regarding the translation of the user input to an SQL expression, it is simple to update only the parts of the configuration file that define the search terms which are directly affected by the changes when tables are being added or restructured; there is no need to modify the scripts which parse the user input and manage the SQL translation.

Since parsing tools for natural languages are based on different algorithms, it may be advantageous to use one or more additional parsing systems for processing the corpora in AET and to compare the parser outputs in order to decrease the number of erroneous syntactic analyses.

The methods and structures we used when designing AET were chosen specifically to enable quick and effortless extensions, updates and changes. The tool in its current state is already well suited for linguistic corpus research concerning adjective modification and co-occurence patterns of adjectives and adverbs in German. Many functionalities could still be added to AET, and the design of the tool makes it easy to do so.

Acknowledgements

This work was carried out as a part of the DFG Collaborative Research Centre 991 "The Structure of Representations in Language, Cognition, and Science" in Düsseldorf, Germany. The authors of the paper would like to thank Kai Koch for the implementation of the user interface prototype as well as Sebastian Löbner, Ekaterina Gabrovska, and Curtis Anderson for their valuable advice on the semantics of adjectives and adverbs. We would also like to thank three anonymous reviewers for their suggestions and constructive comments.

5. Bibliographical References

Baayen, R. H., Piepenbrock, R., and Gulikers, L. (1995). *The CELEX Lexical Database*. Linguistic Data Consortium, Univer- sity of Pennsylvania, Philadelphia, Pennsylvania.

Björkelund, A., Bohnet, B., Hafdell, L., and Nugues, P. (2010). A high-performance syntactic and semantic dependency parser. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, COLING '10, pages 33–36, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., Soricut, R., Specia, L., and Tamchyna, A. (2014). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Choi, J. D., Tetreault, J. R., and Stent, A. (2015). It depends: Dependency parser comparison using a web-based evaluation tool. In *ACL (1)*, pages 387–396. The Association for Computer Linguistics.

Dalmas, M., Dobrovol'skij, D., Goldhahn, D., and Quasthoff, U. (2015). Evaluation with adjectives. towards a corpus-based approach to synonymy. *Lili - Zeitschrift fur Literaturwissenschaft und Linguistik*, 45(177):12–29.

Dye, M., Milin, P., Futrell, R., and Ramscar, M. (2017). Cute little puppies and nice cold beers: An information theoretic analysis of prenominal adjectives. CogSci 2017, 39th Annual Meeting of the Cognitive Science Society.

Eroms, H.-W. (2011). Attributive adjective clusters [Attributive Adjektivcluster]. *Deutsche Sprache*, 39(2):113–136.

Evert, S. and Hardie, A. (2011). Twenty-first century corpus workbench: Updating a query architecture for the new millennium.

Gulikers, L., Rattink, G., and Piepenbrock, R. (1995). German linguistic guide.

Hartung, M. and Frank, A. (2011). Exploring supervised LDA models for assigning attributes to adjective-noun phrases. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP'11, pages 540–551, Stroudsburg, PA, USA. Association for Computational Linguistics.

Klein, W. and Geyken, A. (2010). Das digitale Wörterbuch der deutschen Sprache (DWDS). In *Lexicographica: International annual for lexicography*, pages 79–96. De Gruyter.

Kupietz, M. and Keibel, H. (2009). The Mannheim German Reference Corpus (DeReKo) as a basis for empirical linguistic research. *Working papers in corpus-based linguistics and language education*, 3:53–59.

McNally, L. and Boleda, G. (2004). Relational adjectives as properties of kinds. *Empirical Issues in Formal Syntax and Semantics*, 5:179–196.

Petersen, W. and Hellwig, O. (2016). Exploring the value space of attributes: Unsupervised bidirectional clustering of adjectives in German. In *COLING*, pages 2839–2848. ACL.

Zeller, B., Šnajder, J., and Padó, S. (2013). DErivBase: Inducing and evaluating a derivational morphology resource for German. In *Proceedings of ACL 2013*, pages 1201–1211, Sofia, Bulgaria.

ZAP: An Open-Source Multilingual Annotation Projection Framework

Alan Akbik and **Roland Vollgraf**
Zalando Research
Mühlenstraße 25, 10243 Berlin
{firstname.lastname}@zalando.de

Abstract

Previous work leveraged *annotation projection* as a convenient method to automatically generate linguistic resources such as treebanks or propbanks for new languages. This approach automatically transfers linguistic annotation from a resource-rich source language (SL) to translations in a target language (TL). However, to the best of our knowledge, no publicly available framework for this approach currently exists, limiting researchers' ability to reproduce and compare experiments. In this paper, we present ZAP, the first open-source framework for annotation projection in parallel corpora. Our framework is Java-based and includes methods for preprocessing corpora, computing word-alignments between sentence pairs, transferring different layers of linguistic annotation, and visualization. The framework was designed for ease-of-use with lightweight APIs. We give an overview of ZAP and illustrate its usage.

The framework is available on github at https://github.com/zalandoresearch/zap

Keywords: Annotation projection, corpora, multilingual data

1. Introduction

Linguistically annotated corpora, such as *treebanks* (Marcus et al., 1993) or *propbanks* (Palmer et al., 2005), are a crucial driver of progress in natural language processing research. As a cost-effective alternative to manual annotation, previous work explored the use of *annotation projection* (Yarowsky et al., 2001) in parallel corpora to automatically create linguistically annotated corpora for new languages. This approach requires only a parallel corpus consisting of sentences in a resource-rich source language (SL) and their translations in a target language (TL), as well as existing parsers for the SL. It leverages the hypothesis that translated sentences share a degree of syntactic and, in particular, semantic parallelism (Padó and Lapata, 2009), thus allowing us to automatically transfer linguistic annotations from SL to TL. Refer to Figure 1 for an illustration of this approach, and Section 2 for more details.

Annotation projection resources. Previous works leveraged annotation projection for various layers of syntactic and semantic annotation (Yarowsky et al., 2001; Hwa et al., 2005; Van der Plas et al., 2011; Akbik et al., 2015). However, to the best of our knowledge, no publicly available framework exists for this approach. This limits the ability of researchers to quickly set up experiments, discuss and compare approaches against previous work, and analyze the viability of annotation projection for a specific type of linguistic annotation or language pair.

ZAP framework. For these reasons, we present ZAP, the first open-source framework for annotation projection. The framework contains a number of components callable through lightweight APIs:

1. A set of preprocessing tools that wrap popular NLP libraries to facilitate parsing of sentences in all supported languages.

2. An alignment module for word-aligning translated sentence pairs in a parallel corpus.

3. An annotation projection module for transferring different types of linguistic annotation between word-aligned sentences.

4. A visualization module for rendering word-aligned sentence pairs and projected annotations, enabling researchers to execute and visually inspect annotation projection for specific sentence pairs.

In this extended abstract, we first give a brief overview of annotation projection and related work. We then introduce the ZAP framework and present a walkthrough of core API calls and functionality. Finally, we discuss extensibility of the framework and future work.

2. Annotation Projection

We first briefly illustrate the principle of annotation projection using the sentence pair in Figure 1. Here, the source language is English and the target language is German. Our goal is to automatically generate linguistic annotation for the German sentence. In the first step (Figure 1.a), we use syntactic and semantic parsers to predict part-of-speech (PoS) tags, dependencies and semantic roles for the English sentence. We also perform word alignment to link each English word to its German translation.

We then successively transfer linguistic annotation along these word alignments to the German sentence. We begin with word-level PoS tags (see Figure 1.b). For instance, we transfer the NOUN tags from the English words *cat* and *cheese* onto the German translations *Katze* and *Käse*, thereby marking them up as nouns. We then also project dependency arcs and labels, as well as semantic roles (see Figure 1.c), thus learning that *Katze* (like the *cat* in the SL) is a syntactic subject that takes the semantic role of CONSUMER in the target sentence. Thus, the German sentence is automatically annotated with multiple layers of linguistic annotation.

Previous work. Previous work used annotation projection for a wide range of annotations, including PoS tags

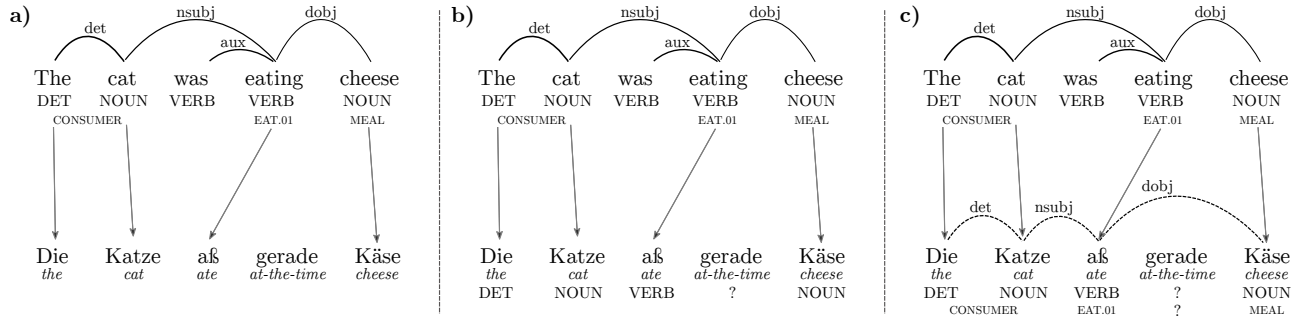


Figure 1: Stepwise example of annotation projection for an English-German sentence pair. In a), SL parsers are used to predict syntax and semantic roles for the English sentence, and words are aligned to their German translations. In b) PoS tags are transferred along alignments onto German words. In c) dependency arcs and semantic roles are transferred along word alignments, thus automatically labeling the German sentence with dependencies, PoS tags and semantic roles. Note that the unaligned German word *gerade* remains unannotated.

(Yarowsky et al., 2001), syntactic chunks (Yarowsky and Ngai, 2001), dependency trees (Hwa et al., 2005), word senses (Bentivogli and Pianta, 2005), named entities (Mayhew et al., 2017) and semantic roles (Padó and Lapata, 2009; Akbik et al., 2015).

However, as the example in Figure 1 shows, annotation projection may not always produce fully annotated target language sentences. Previous work found annotation projection to be sensitive to factors such as the quality of translations, accuracy of word alignments and parsers (Akbik et al., 2015), and subject to errors stemming from translational divergences (Dorr, 1994). For these reasons, previous works devised a number of strategies to address such issues. These include identifying and filtering suboptimally aligned sentence pairs from the parallel corpus (Ni et al., 2017), blocking the projection of specific annotations (Van der Plas et al., 2011), guiding projection using cross-lingual statistics (Täckström et al., 2012), using semi-supervised learning to fill annotation gaps in the target language corpus (Akbik et al., 2015) and leveraging crowdsourcing to curate projections (Wang et al., 2017).

With the release of ZAP, we aim to assist researchers investigate and further improve such annotation projection strategies for various layers of linguistic annotation, with the hope of eventually generating new linguistic resources for low-resource languages that approach the quality of expert annotation.

3. The ZAP Framework

The ZAP framework is a Java-based open source project available on github[1]. It provides both a set of lightweight APIs to programmatically design annotation projection experiments, as well as a simple UI for exploratory analysis of projection approaches. In the following, we illustrate the core usage of the framework.

3.1. Packaging and Distribution

The project is managed using the *maven* build automation tool, giving researchers two simple ways to start using the framework. The first is to clone the github repository and build the project locally by calling `mvn install` in its root folder, which will execute unit tests and install the project into the local maven repository. This is the recommended option for researchers interested in fully understanding and extending the APIs for their experiments. The second option is to include the project as a dependency into a Java project. This is accomplished simply by adding the following dependency to the project's POM.XML:

```
<dependency>
 <groupId>org.zalando.research</groupId>
 <artifactId>zap</artifactId>
 <version>1.0</version>
</dependency>
```

We recommend the latter option for most researchers, allowing them to quickly get started with building their own projection experiments.

3.2. Parser Wrappers

The first component of ZAP are wrappers around popular NLP tools, namely STANFORD CORENLP (Manning et al., 2014) (for PoS-tagging, named entity recognition and dependency parsing), the ANNA lemmatizers (Bohnet, 2010) (for lemmatization) and the MATE toolkit (Björkelund et al., 2009) (for semantic role labeling). These tools were selected for their open source availability and their maven packaging. This design choice ensures that users do not need to install any third party parsers or tools to get started – maven downloads all required dependencies automatically.

Parsing a sentence. After adding ZAP to a project, users can parse any sentence with two lines of code:

```
PipelineWrapper pipeline = new
    PipelineWrapper(Language.ENGLISH);

Sentence parse = pipeline.parse("The
    cat was eating cheese.");
```

The first line initializes the parser wrapper for the selected language, the second parses the sentence into an object that contains the full syntactic-semantic parse of the sentence.

[1] https://github.com/zalandoresearch/zap

This object provides methods for accessing annotations and rendering itself in *CoNLL-U*[2] format.

Different languages. In order to parse a sentence in a different language, the wrapper only needs to be initialized with the appropriate language enum, as follows:

```
PipelineWrapper pipeline = new
    PipelineWrapper(Language.GERMAN);

Sentence parse = pipeline.parse("Die
    Katze aß gerade Käse.");
```

At time of writing, ZAP wraps tools for parsing of English, German, French, Spanish and Chinese. We expect more languages added to this list in the near future.

Gold-annotated corpora. It is also possible to create SENTENCE objects from existing gold-annotated treebanks in lieu of using parsers. For this purpose, we include classes for reading corpora in CoNLL-U and CoNLL-X format.

3.3. Heuristic Word Alignment

A prerequisite for annotation projection is to word-align parallel sentences. A first step is to create a BISENTENCE object that contains a source sentence and its target language translation. The source sentence is typically parsed (as described above), while the target sentence is initialized without annotation:

```
Sentence sourceSentence =
    pipeline.parse("The cat was eating
    cheese.");

Sentence targetSentence =
    Sentence.fromTokenized("Die Katze
    aß gerade Käse.");

BiSentence biSentence = new
    BiSentence(sourceSentence,
    targetSentence);
```

The ZAP framework offers several ways to add word alignments to a BISENTENCE object. One way is to read externally computed word alignments in the "Pharaoh format", as produced by most popular word-alignment tools including FASTALIGN (Dyer et al., 2013) and BERKELEYALIGNER (DeNero and Liang, 2007). As an alternative option, ZAP also offers a heuristic word alignment module that uses pre-computed word translation probabilities computed over large parallel corpora (Tiedemann, 2012). The latter option has the advantage of not requiring external tools. It can be called by instantiating the HEURISTICALIGNER.

```
HeuristicAligner aligner =
    HeuristicAligner
  .getInstance(Language.GERMAN);

biSentence.align(aligner);
```

[2] `http://universaldependencies.org/format.html` lists a specification of the format.

```
1   Die     _   DET   DT   _   2   det
2   Katze   _   NOUN  NN   _   3   nsubj
3   aß      _   VERB  VBG  _   0   root
4   gerade  _   _     _    _   0   _
5   Käse    _   NOUN  NN   _   3   dobj
6   .       _   _     _    _   0   _
```

Figure 2: German example sentence with projected annotations rendered in CoNLL-U format (first 8 columns).

3.4. Annotation Projection

Once a word-aligned BISENTENCE is produced, a simple call of the ANNOTATIONTRANSFER object suffices to project all word-level (PoS-tags), span-level (named entities), tree-level (dependencies) and proposition semantic levels of annotation from the source sentence onto the target sentence:

```
new AnnotationTransfer()
    .transfer(biSentence);
```

In ZAP, annotation is projected following standard practices. Word-level annotation is projected using direct transfer (Van der Plas et al., 2011), i.e. directly following word alignments. Annotations that span several words, such as semantic roles, are projected onto aligned target language constituents that are identified following the procedure introduced in Padó and Lapata (2009). Next to the above-illustrated method call which transfers all linguistic annotation, users may also choose only a subset of annotation types to be projected. For instance, a user may only be interested in projecting named entities, while using existing target language parsers to identify syntactic structure.

Annotation projection produces a target language SENTENCE object with added linguistic annotation. This object can then be saved in CoNLL-U format, for instance for verification or for use in training of target language parsers. Refer to Figure 2 for an illustration.

3.5. Visualization

As discussed above, previous works found that the viability of annotation projection depends both on the type of annotation being projected as well as the specific language pair in question. For this reason, previous work often conducted careful qualitative analyses of error sources and defined strategies to address these issues. To assist such analysis, we previously presented a demonstration of a web-based UI – called THE PROJECTOR (Akbik and Vollgraf, 2017) – that visualizes sentence pairs, word alignments and various layers of linguistic annotation. We include the visualization capabilities of THE PROJECTOR into ZAP.

To launch the interactive UI, a simple method call suffices:

```
int port = 9000;
TheProjectorUI.instance()
    .startServerAtPort(port);
```

This launches the web UI locally at the specified port. For more information on the functionality of THE PROJECTOR, refer to Akbik and Vollgraf (2017). A screenshot of the

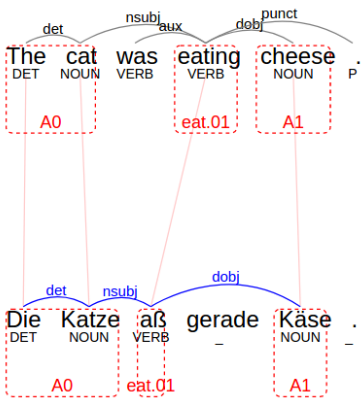


Figure 3: ZAP's visualization of the running example.

example sentence pair as rendered in the UI is illustrated in Figure 3.

4. Summary and Outlook: Extending ZAP

To facilitate experimentation with different strategies, we designed simple interfaces for ZAP that allow researchers to extend the framework. It is our hope that the open source nature of the project will encourage more research into annotation projection, eventually leading to automatically generated linguistic resources that approach the quality of expert annotation. Our current work focuses on extending the framework to support more languages and types of annotation, as well as adding heuristic methods for addressing translational divergences.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no 732328 ("FashionBrain").

5. Bibliographical References

Akbik, A. and Vollgraf, R. (2017). The projector: An interactive annotation projection visualization tool. In *EMNLP 2017, Conference on Empirical Methods on Natural Language Processing*, pages 43–48.

Akbik, A., Chiticariu, L., Danilevsky, M., Li, Y., Vaithyanathan, S., and Zhu, H. (2015). Generating high quality proposition banks for multilingual semantic role labeling. In *ACL 2015, 53rd Annual Meeting of the Association for Computational Linguistics*, pages 397–407.

Bentivogli, L. and Pianta, E. (2005). Exploiting parallel texts in the creation of multilingual semantically annotated resources: the multisemcor corpus. *Natural Language Engineering*, 11(3):247–261.

Björkelund, A., Hafdell, L., and Nugues, P. (2009). Multilingual semantic role labeling. In *CoNLL 2009, 13th Conference on Computational Natural Language Learning: Shared Task*, pages 43–48.

Bohnet, B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In *COLING 2010, 23rd International Conference on Computational Linguistics*, pages 89–97.

DeNero, J. and Liang, P. (2007). The berkeley aligner. `http://code.google.com/p/berkeleyaligner/`.

Dorr, B. J. (1994). Machine translation divergences: A formal description and proposed solution. *Computational Linguistics*, 20(4):597–633.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. In *NAACL 2013, North American Chapter of the Association for Computational Linguistics*, pages 644–648.

Hwa, R., Resnik, P., Weinberg, A., Cabezas, C., and Kolak, O. (2005). Bootstrapping parsers via syntactic projection across parallel texts. *Natural language engineering*, 11(03):311–325.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL 2014, 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Mayhew, S., Tsai, C.-T., and Roth, D. (2017). Cheap translation for cross-lingual named entity recognition. In *EMNLP 2017, Conference on Empirical Methods in Natural Language Processing*, pages 2526–2535.

Ni, J., Dinu, G., and Florian, R. (2017). Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. *CoRR*, abs/1707.02483.

Padó, S. and Lapata, M. (2009). Cross-lingual annotation projection for semantic roles. *Journal of Artificial Intelligence Research*, 36(1):307–340.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Täckström, O., McDonald, R., and Uszkoreit, J. (2012). Cross-lingual word clusters for direct transfer of linguistic structure. In *NAACL 2012, North American Chapter of the Association for Computational Linguistics*, pages 477–487.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus. In *LREC 2012, 8th International Conference on Language Resources and Evaluation*, pages 2214–2218.

Van der Plas, L., Merlo, P., and Henderson, J. (2011). Scaling up automatic cross-lingual semantic role annotation. In *ACL 2011, 49th Annual Meeting of the Association for Computational Linguistics*, pages 299–304.

Wang, C., Akbik, A., Chiticariu, L., Li, Y., Xia, F., and Xu, A. (2017). CROWD-IN-THE-LOOP: A hybrid approach for annotating semantic roles. In *EMNLP 2017, Conference on Empirical Methods in Natural Language Processing*, pages 1914–1923.

Yarowsky, D. and Ngai, G. (2001). Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *NAACL 2001, North American Chapter of the Association for Computational Linguistics*, pages 1–8.

Yarowsky, D., Ngai, G., and Wicentowski, R. (2001). Inducing multilingual text analysis tools via robust projection across aligned corpora. In *HLT 2001, 1st International Conference on Human Language Technology Research*, pages 1–8.

PALMYRA: A Platform Independent Dependency Annotation Tool for Morphologically Rich Languages

Talha Javed, Nizar Habash, Dima Taji
Computational Approaches to Modeling Language Lab
New York University Abu Dhabi
{talha.javed, nizar.habash, dima.taji}@nyu.edu

Abstract

We present PALMYRA, a platform independent graphical dependency tree visualization and editing software. PALMYRA has been specifically designed to support the complexities of syntactic annotation of morphologically rich languages, especially regarding easy change of morphological tokenization through edits, additions, deletions, splits and merges of words. PALMYRA uses an intuitive drag-and-drop interface for editing tree structures, and provides pop-up boxes and keyboard shortcuts for part-of-speech and link label tagging.

Keywords: Annotation, Interfaces, Syntax, Morphologically Rich Languages

1. Introduction

Producing high-quality natural language syntactic annotation is expensive. Well-known large-scale syntactic annotation projects, such as the Penn Treebank (Marcus et al., 1993) and the Prague Dependency Treebank (Böhmová et al., 2003), relied on expert linguists to produce carefully annotated data. This process is rather costly, and as a result, such annotation projects have been undertaken for only a handful of important languages.

Efficient annotation tools play an important role in lowering treebank development costs and enabling the creation of larger, higher quality treebanks. The typical approach is to automatically create syntactic annotations, which are then manually corrected. In this scenario, a goal of the annotation tool is to lower the annotation burden on the annotators as much as possible. For morphologically rich languages (MRLs), such as Arabic and Hebrew, syntactic annotation includes morphological ambiguity resolution as well as tokenization adjustment. The change of tokenization affects the total token count of a sentence and requires adding or deleting tree tokens. For example, the word وجدها *wjdhA*[1] may be automatically analyzed as وجد+ها *wjd*/VERB+*hA*/PRON 'he found her', but needs to be corrected to و+جد+ها *w*/CONJ+*jd*/NOUN+*hA*/PRON[2] 'and her grandfather' (or vice versa). Since an automatic parser may select an incorrect analysis and tokenization that will need to be corrected, an interface that facilitates such corrections during syntactic annotation would be most helpful.

PALMYRA is an annotation tool intended to help with the annotation of MRLs in general. However, we focus on Arabic here as a representative language for the type of challenges we are interested in addressing. We are primarily interested in dependency treebanking and thus the discussion of related work will be focused on dependency representations.

[1]Arabic transliteration is presented in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007).

[2]For more information on Arabic morphology and natural language processing, see (Habash, 2010).

2. Design Aims

The following are the design aims for the PALMYRA editor interface. **First**, we want the editor to provide an easy and direct way to modify the form of a word: by changing it, splitting it, or merging it with other words. **Second**, we want the editor to allow easy switching between syntactic annotation and morphological tokenization to allow the annotators to make these decisions jointly and avoid error propagation. **Third**, we want the editor to be platform independent, and require no installation effort. **Fourth**, we want the editor to be language-independent and easily configurable. **Finally**, we want the editor to be open source so it can be easily extended for other projects without restrictions.

3. Related Work

Much work has been done on improving dependency parsing and thus many tools for annotation have been created. The most famous and commonly used tool is TrEd. BRAT and WebAnno are web-based tools for text annotation, designed mainly for collaboration. EasyTree is another lightweight tool designed to be run in browsers and consists of only a front-end. These tools have some great features but also lack key features which do not make them completely suitable for annotating MRLs. In particular, changing the tokenization of a word is not easy to do in these systems. We discuss these tools next to highlight their advantages and shortcomings.

TrEd (Pajas and Štěpánek, 2008) is a graph visualization and manipulation program written in Perl. It has been used as an annotation tool for several treebank projects, including the Prague Dependency Treebank (Böhmová et al., 2003), Prague Arabic Dependency Treebank (Hajic et al., 2004) and Columbia Arabic Treebank (CATiB) (Habash and Roth, 2009). It supports macros to automate frequently repeated operations and has a substantial number of features. It can be unintuitive at times and difficult to learn; thus, it may not be a good choice for less experienced annotators. A notable limitation of TrEd is that it is a standalone application and thus cannot be run through a web-browser. PALMYRA is designed to run in web browsers,

making it simple to use and appropriate for web-based annotation tasks. TrEd does not provide any simple option for word tokenization.

BRAT (Stenetorp et al., 2012) is a web-based annotation tool with a focus on collaborative annotation. BRAT is designed in particular for structured annotation, where the notes are not freeform text but have a fixed form that can be automatically processed and interpreted by a computer. The way BRAT displays text, on a single line, makes following the dependency arcs somewhat difficult, and thus, this tool is probably more appropriate for other tasks, such as marking events and named entities. Furthermore, BRAT is slow when processing documents of more than 100 sentences and has limited support for different file formats. It also does not allow for web-based configuration of tag sets.

WebAnno (Yimam et al., 2013) is also another general-purpose web-based annotation tool mainly meant for distributed teams. To visualize the text and annotations, it uses the JavaScript based annotation visualization from BRAT, thus facing the same display issues. WebAnno supports type specification through the import/export of tag sets. Similar to TrEd, WebAnno and BRAT have no support for word tokenization.

EasyTree (Little and Tratz, 2016) is a light-weight tool designed for annotating dependency trees in browsers. It is limited to a front-end only and does not provide any interface to integrate parsers for pre-annotation. EasyTree has multiple intuitive features, such as color-coded part-of-speech (POS) indicators and optional translation displays. EasyTree allows the customization of POS tags but does not maintain sentence order of nodes. It has no functionality for splitting or merging of words. PALMYRA's design is highly influenced by EasyTree and uses some of its code.

4. PALMYRA Design Specifications

Design and Implementation PALMYRA was written entirely in JavaScript, CSS, and HTML. As such it can run in modern web browsers and is platform independent and requires no special installation process.[3] For the interactive graphical display, PALMYRA leverages D3, a popular open source data visualization library written in JavaScript. PALMYRA utilizes UTF-8 encoding, which enables it to work with characters from different languages. PALMYRA is built on top of the base code of EasyTree (Little and Tratz, 2016) and makes use of its drag-and-drop functionality. Figure 2 shows the main Palmyra interface.

Input Files For input, PALMYRA supports a basic dependency file format containing five columns; ID, word, POS, parent, and relation. PALMYRA input files may contain a number of independent trees, which can be browsed through. Editing is performed on one tree at a time. Figure 1 shows the five columns used to represent the dependency tree shown in Figures 2 and 4. The columns, in order, correspond to (i) a unique word index, (ii) the word, (iii) the CATiB POS tag (Habash and Roth, 2009), (iv) the word's parent, (v) the relation (link) label. The words in this tree

[3]Try PALMYRA online at `https://camel.abudhabi.nyu.edu/palmyra/`.

ID	Word	POS	Parent	Relation
1	و+	PRT	3	MOD
2	لم	PRT	3	MOD
3	يتم	VRB	0	---
4	اعتقال	NOM	3	SBJ
5	أحد	NOM	4	IDF
6	ب+	PRT	3	MOD
7	حسب	NOM	6	OBJ
8	الشرطة	NOM	7	IDF
9	.	PNX	3	MOD

Figure 1: The tree for the sentence ولم يتم اعتقال أحد بحسب الشرطة *wlm ytm AςtqAl ÂHd bHsb AlšrTħ* 'and no one was arrested according to the police."

are tokenized according to the Penn Arabic Treebank tokenization scheme (Maamouri et al., 2004), but other Arabic tokenization schemes can be used just the same (Habash, 2010). The format shown is produced by CamelParser (Shahrour et al., 2016).

PALMYRA also accepts sentences for input when annotating from scratch. Every input line is considered as a separate tree. It tokenizes a sentence on spaces, and assumes all nodes are siblings under the root with default POS tag and link labels.

Output Files PALMYRA has two output formats: the dependency format discussed above, and PNG image.

Configuration File PALMYRA takes as input an optional configuration file, which specifies the various configuration and annotation options for the tool. The config files consist of key-value pairs, each specifying a property of the editor. The most important use of the config file is to specify the options for POS tags, link labels, and keyboard shortcuts.

5. Tree Node Editing

Dependency Tree Display Figure 2 presents a screenshot of a dependency tree in PALMYRA. The words are shown at the bottom in a straight horizontal line, and the tree nodes are aligned horizontally directly above the corresponding words, and vertically according to the structure of the tree. PALMYRA leverages D3-hierarchy module which includes layout algorithms for visualizing hierarchical data. The POS of a word is displayed next to it, and the relation link is shown half-way on the edge connecting two tree nodes. The direction of display of the tree can be changed to be left-to-right or right-to-left.

Drag-and-Drop and Zoom Tree editing in PALMYRA is straightforward; users simply click on word nodes and move them around using drag-and-drop. When the user begins dragging a node, red circles appear around the remaining nodes. These 'drop zones' indicate where the node may be re-attached. Figure 2 displays the tree during editing, when the node الشرطة is being dragged. When the dragged node is eventually dropped onto a drop zone, a link representing a syntactic dependency is created between the two nodes with the dropped node as the child. The re-attached

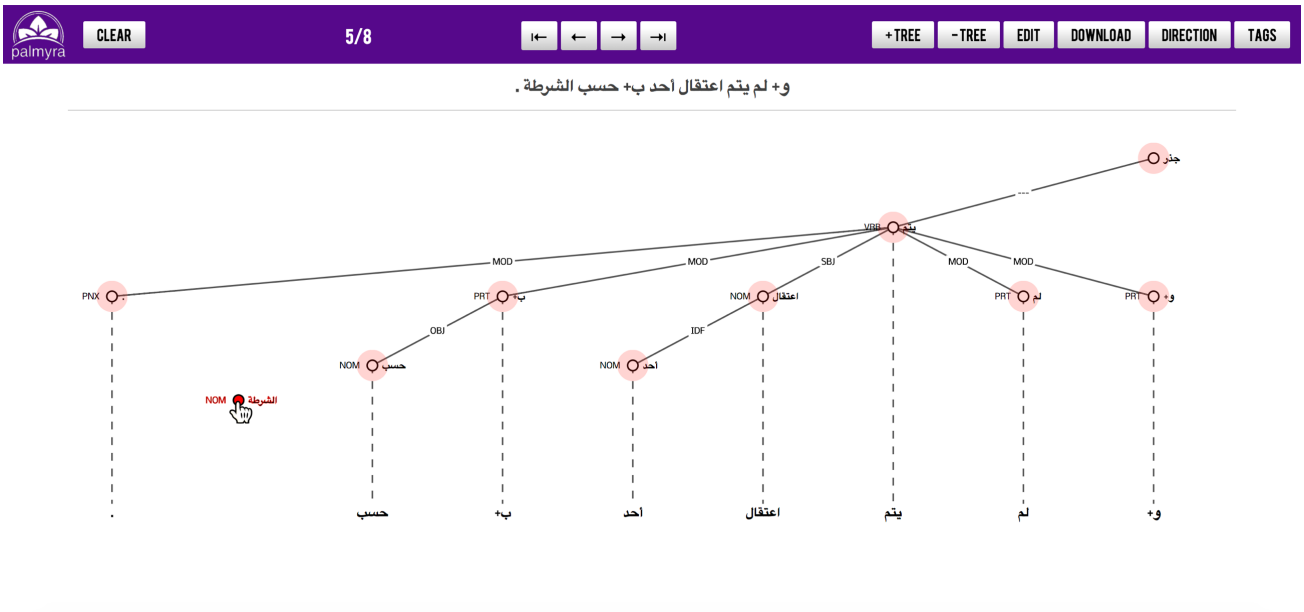


Figure 2: PALMYRA Interface while editing a tree using the drag-and-drop functionality

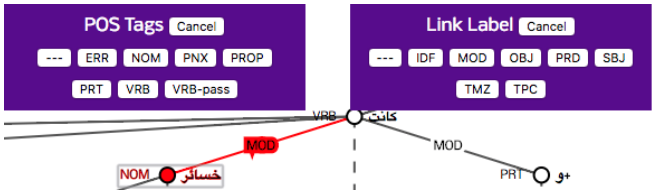


Figure 3: POS Tag and Link Label Selection

node is inserted such that it and its new siblings remain sorted according to the original word order of the sentence. The tool also includes tree zoom-in and zoom-out options.

Keyboard Tree Node Navigation Tree node selection can be done by directly clicking on the node of interest. But PALMYRA also allows the user to move from one node to another using keyboard arrows. Node-highlighting provides feedback on the selected node and facilitates fast directed movement within the tree: to go up or down a parent connection or jump to siblings' subtrees using keyboard arrows.

Part-of-Speech Tag and Link Label Clicking on the "TAGS" button opens two sub-menus with the available options for POS tags and link labels (Figure 3). The user must select a node before being able to change the POS tag or link label. A node can be selected by either clicking on the node or navigating to it through keyboard arrow keys. Once selected, the user can change the POS tag or link label either through keyboard shortcuts or by clicking on the displayed buttons in the TAGS sub-menus using a mouse.

Adding and Deleting Trees The user can also add a new tree to the file using the "+ TREE" button. The tree starts with only a root, and the user can use the word insertion functionality (discussed in the next section) to add nodes to the tree. The "- TREE" button can be used to delete a tree.

6. Sentence Token Editing

The sentence token editing mode is toggled when clicking the "EDIT" mode button in the upper interface bar. Within this mode, the annotator can delete, insert, change, split or merge tree nodes. Figure 4 displays the tree in sentence token editing mode.

Deletion In the "EDIT" mode, PALMYRA displays a red "x" sign under nodes, which can be clicked to delete nodes. If the deleted node has children, they are all promoted to be the children of the deleted node's parent.

Insertion Similarly, a green "+" sign is shown between nodes to provide the ability to add a new node in the respective position. Inserted nodes are linked to the root and given the word form New/جديد.

Substitution and Word Splitting Clicking the word in edit mode causes a word edit pop-up to appear. The word can be rewritten for a simple substitution. A word can also be split into multiple words that will then be represented as different nodes in the tree. This is done by simply adding spaces in the middle of the word where the splits should take place.

Merge Words Merging neighboring words can be accomplished by clicking on the arrows to the right or left of a word. The node for whom the merge arrows are clicked is deleted and its name (word) is added to the name (word) of the respective neighbor node (on left or right given the clicked arrow). The original node loses its POS tag and link

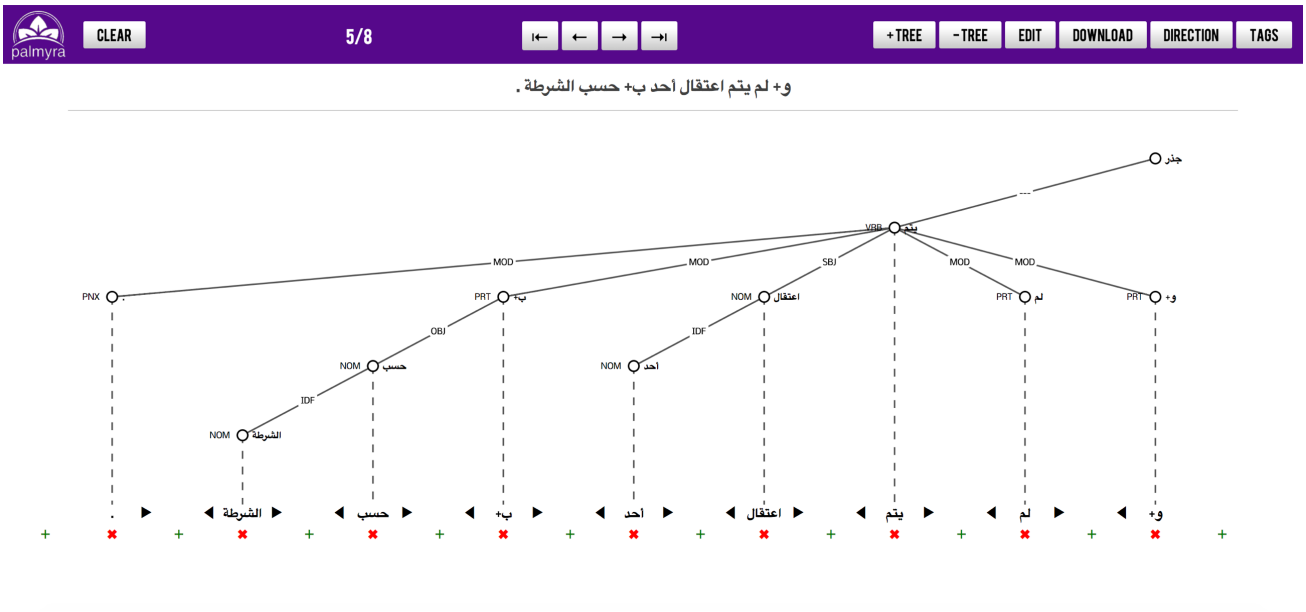


Figure 4: PALMYRA Interface in sentence token editing mode

label, and its children are assigned to the node it merged with. At the end, only one node remains with its word being the concatenation of the words of the two original nodes.

7. Conclusion and Future Work

In this paper, we described PALMYRA, a platform independent graphical dependency tree visualization and editing software. PALMYRA has been specifically designed to support the complexities of syntactic annotation of morphologically rich languages, especially regarding easy change of morphological tokenization. Being built entirely using standard web technologies, PALMYRA runs on all major web browsers and is ideal for online annotation efforts, such as crowdsourcing efforts.

In the future, we will be using PALMYRA heavily as part of a treebanking annotation project. We plan to continue enhancing PALMYRA's capabilities. In particular, we are interested in linking it to a state-of-the-art Arabic parser (Shahrour et al., 2016) to support online syntactic analysis.

8. Bibliographical References

Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2003). The Prague dependency treebank. In *Treebanks*, pages 103–127. Springer.

Habash, N. and Roth, R. M. (2009). Catib: The Columbia Arabic treebank. In *Proceedings of the ACL-IJCNLP*, Stroudsburg, PA, USA.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic Transliteration. In A. van den Bosch et al., editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Hajic, J., Smrz, O., Zemánek, P., Šnaidauf, J., and Beška, E. (2004). Prague Arabic dependency treebank: Development in data and tools. In *Proc. of the NEMLAR Intern. Conf. on Arabic Language Resources and Tools*, pages 110–117.

Little, A. and Tratz, S. (2016). Easytree: A graphical tool for dependency tree annotation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, 2016.*

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330.

Pajas, P. and Štěpánek, J. (2008). Recent advances in a feature-rich framework for treebank annotation. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 673–680, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shahrour, A., Khalifa, S., Taji, D., and Habash, N. (2016). Camelparser: A system for Arabic syntactic analysis and morphological disambiguation. In *COLING (Demos)*, pages 228–232.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: A web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 102–107, Stroudsburg, PA, USA.

Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). Webanno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Sofia, Bulgaria.

A Web-based System for Crowd-in-the-Loop Dependency Treebanking

Stephen Tratz*, **Nhien Phan**†
*U.S. Army Research Laboratory, Adelphi, Maryland, USA
†University of Maryland, College Park, Maryland, USA
stephen.c.tratz.civ@mail.mil, nphan12@terpmail.umd.edu

Abstract

Treebanks exist for many different languages, but they are often quite limited in terms of size, genre, and topic coverage. It is difficult to expand these treebanks or to develop new ones in part because manual annotation is time-consuming and expensive. Human-in-the-loop methods that leverage machine learning algorithms during the annotation process are one set of techniques that could be employed to accelerate annotation of large numbers of sentences. Additionally, crowdsourcing could be used to hire a large number of annotators at relatively low cost. Currently, there are few treebanking tools available that support either human-in-the-loop methods or crowdsourcing. To address this, we introduce CROWDTREE, a web-based interactive tool for editing dependency trees. In addition to the visual frontend, the system has a Java servlet that can train a parsing model during the annotation process. This parsing model can then be applied to sentences as they are requested by annotators so that, instead of annotating sentences from scratch, annotators need only to edit the model's predictions, potentially resulting in significant time savings. Multiple annotators can work simultaneously, and the system is even designed to be compatible with Mechanical Turk. Thus, CROWDTREE supports not simply human-in-the-loop treebanking, but crowd-in-the-loop treebanking.

Keywords: annotation, dependency trees, visualization, GUI, crowdsourcing, treebanking

1. Introduction

Although treebanks exist for a number of languages, they are often quite small in size. Even some of the largest and most useful treebanks are restricted to only a handful of sources, genres, and/or topic areas—the venerable Penn Treebank (Marcus et al., 1993) being a prime example of this. Unfortunately, manual annotation is expensive and slow, which impedes both the expansion of existing treebanks and the creation of new ones. For example, The Prague Dependency Treebank, which contains over 1 million syntactically connected words, cost approximately $600,000 to create (Böhmová et al., 2003). One method for accelerating the treebanking process is to utilize human-in-the-loop[1] methods that leverage machine learning algorithms during annotation. Such algorithms could pre-parse the data, highlight attachment decisions of greater importance, and/or suggest sentences for annotation that are most likely to prove useful as training examples. Another useful method is crowdsourcing, which taps into a large (and typically inexpensive) worker pool. Together, these complementary techniques hold tremendous promise for reducing the time and cost of creating large datasets for natural language processing, including treebanks. In order to facilitate faster and less expensive dependency tree annotation, this paper presents CROWDTREE, which is, to the best of our knowledge, the first web-based treebanking tool designed specifically for human-in-the-loop, and even crowd-in-the-loop, tree annotation.

The system consists of two primary components, an interactive graphical frontend, shown in Figure 1, and a Java servlet. The frontend, which enables users to visually construct/edit dependency trees using drag-and-drop, is based on the EASYTREE dependency tree editor (Little and Tratz, 2016) but has a variety of improvements, including an improved layout and various bug fixes. It is written entirely in HTML, CSS, and JavaScript, and, thus, works with modern browsers without the need for special plugins. The frontend communicates with the second component, the Java servlet, which, in addition to recording the annotated data, has the option to constantly train a transition-style parsing model (McDonald and Nivre, 2007) throughout the annotation process from the work that is submitted to the server. This model can parse a sentence as it is provided to an annotator, which promises to reduce the number of edits required to construct correct dependency trees. CROWDTREE has a Mechanical Turk-compatible mode, which opens new possibilities for large scale treebanking as well as new opportunities for research into the use of large numbers of non-expert treebankers.

The remainder of this paper provides a more detailed description of the nature and capabilities of our tree annotation system (Section 2), describes related tools and research efforts (Section 3), summarizes our contributions (Section 4), and details some of our planned experimental work and feature additions (Section 5). We are planning to release our system open source in the near future via the U.S. Army Research Laboratory's GitHub site or similar repository.

2. System Description

CROWDTREE has a client-server architecture with two components: a graphical web-based interface that annotators use to create dependency trees and a Java servlet for storing user annotations and running machine learning processes.

2.1. User Interface

The user interface, shown in Figure 1, is written using standard web technologies—HTML, CSS, and JavaScript. The interactive Support Vector Graphics-based dependency tree widget is implemented using the popular D3.JS[2] data

[1] By 'human-in-the-loop', we mean 'involving both human interaction and machine learning'.

[2] https://d3js.org/

visualization and manipulation JavaScript library. Anno-

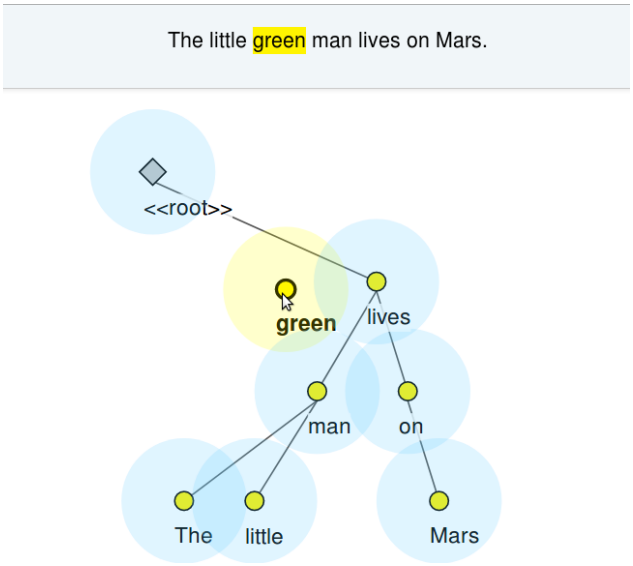


Figure 1: In CROWDTREE, blue "drop zones" highlight possible attachment sites when dragging word nodes.

tators alter the dependency tree structure by clicking on word nodes and dragging them to their appropriate attachment sites. When the dragging process begins, circular "drop zones" appear around the potential attachment sites, as shown in Figure 1. When the dragged node is dropped, the layout of the tree adjusts in a smooth, animated fashion. The code is derived from the EASYTREE dependency tree editor (Little and Tratz, 2016) but the visual layout is modeled after the layout used in the tree editor TRED (Pajas and Štěpánek, 2008), with the words maintaining their original left-to-right order on the horizontal axis. One of the challenges with the EASYTREE layout is that the horizontal ordering of word nodes is not maintained across the entire sentence, and, hence, a noun phrase like *the little brown flower pot* would have the last two words *brown* and *flower* appear to the right of the word *pot* instead of to its left.

Both labeled and unlabeled dependency annotation are supported by the system, which can be configured to either hide or show the dependency labels and part-of-speech tags of the word nodes. Plugging in one's own dependency labels and part-of-speech tags can be accomplished easily with just a text editor.

CROWDTREE includes pan and zoom capabilities to enable users to easily work with very large trees. Zooming in and out is accomplished using the mouse scrollwheel, and the user can reposition the entire tree by clicking on the background and dragging the mouse in the desired direction.

2.2. Java Servlet

The Java servlet, in addition to handling some important input and output functions, is in charge of running machine learning processes. It currently has the option of training either a SWAP parser (Nivre, 2009) or an EASYFIRST style parsing model (Goldberg and Elhadad, 2010) with support for non-projectivity via swapping (Tratz and Hovy, 2011). The servlet runs a model training thread that continuously iterates over the annotations that have been received; thus, the model tends to improve as annotation proceeds. Whenever an annotator requests a sentence from the servlet, the parser processes the sentence using its current model weights and sends the resulting parse to the annotator for correction. The same parsing model is shared across all annotators, which enables them to benefit from each other's work. This promises to accelerate the entire annotation process by minimizing the number of edits annotators need to make in order to construct correct parse structures. [3]

2.3. Input and Output

One of the significant limitations of the EASYTREE dependency tree editor, which our system is derived from, is that the default method for getting data into and out of the editor is to copy and paste an individual sentence, make any needed changes, and then save out a file containing the annotation. This process is, of course, rather tedious, not to mention disruptive to the annotator's concentration. With our tree editor, the locations of the input and output files are specified in a system configuration file. The sentences contained in the input file are loaded by the Java servlet and are sent, as requested, to the web client via HTTP/HTTPS in JavaScript Object Notation (JSON). The annotations produced by the annotator(s) are then returned to the servlet in a similar fashion. When the servlet receives annotated trees, it writes them out to the file specified in the configuration and adds them to the pool of training instances that the server-side parser learns from.

CROWDTREE includes Java classes for reading files in different input formats, including plain text, CoNLL-U, and our own format. To support as many formats as possible, it is also possible to implement a custom reader Java class and plug it in. The frontend is capable of displaying both left-to-right and right-to-left scripts, and UTF-8 encoding is used throughout the system, so it can support a wide variety of different languages.

2.4. Mechanical Turk Mode

One of the most notable features of CROWDTREE is that it was designed from the beginning to be able to run in connection with the Amazon Mechanical Turk crowdsourcing platform. This is accomplished via Mechanical Turk's *ExternalQuestion* Human Intelligence Tasks (HITs). HITs are the basic unit of work on Mechanical Turk; essentially, they are questions to be answered, with a monetary award attached. In the case of *ExternalQuestion* HITs, the requester provides Amazon with the URL of a web site located externally to the main Mechanical Turk website. As shown in Figure 2, when the workers on Mechanical Turk view and work on the HITs, they see a web page hosted on the main Mechanical Turk website with an internal frame (an HTML *iframe*) that points to the external website hosting the CROWDTREE system. When the worker submits his/her work, it is sent both to the external server, where it can be

[3] Of course, submission of erroneous annotations will likely degrade the performance of the model. Mechanisms for mitigating this negative effect may be necessary, depending on the use case.

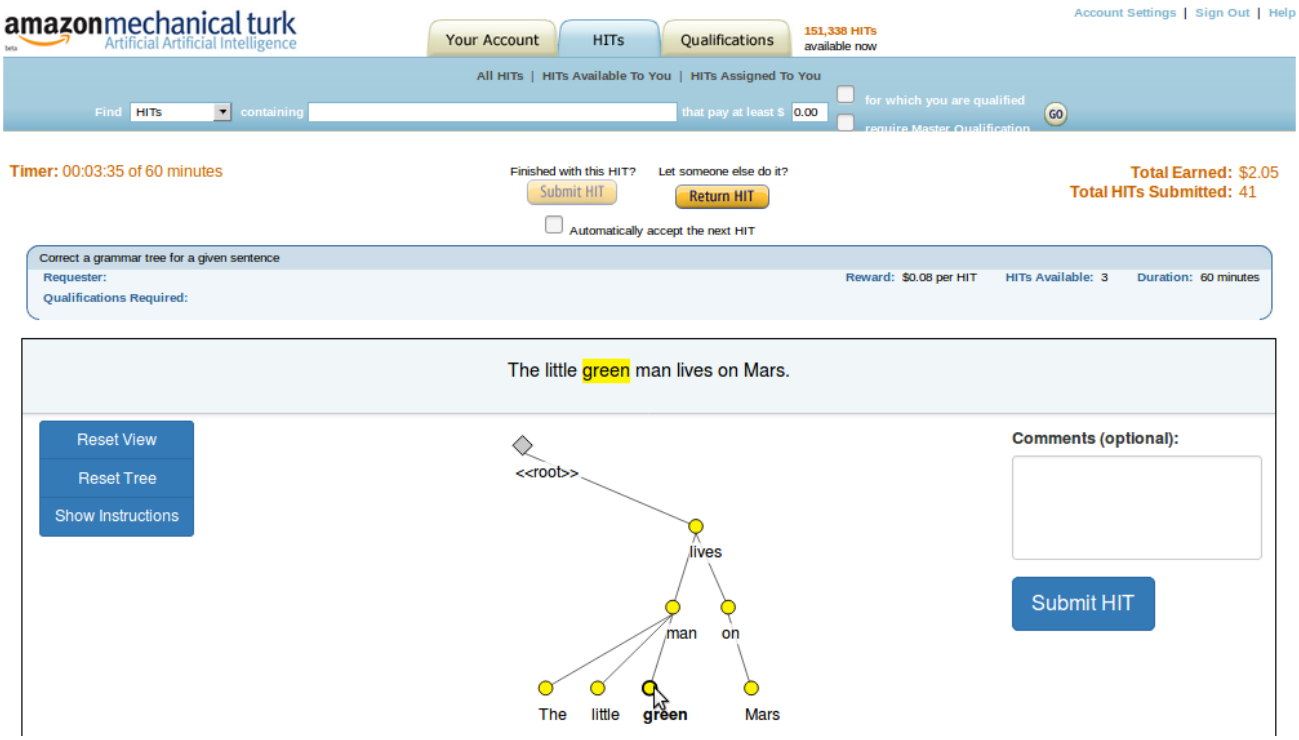


Figure 2: Screenshot of CROWDTREE working in the Mechanical Turk sandbox.

accessed by the parsing model training thread (or any machine learning processes implemented in the future), and to the Mechanical Turk website so that the worker can proceed to the next HIT, have his/her work approved, and receive payment.

2.5. Standalone Mode

As mentioned above, CROWDTREE was designed with Mechanical Turk in mind. It is possible, however, for annotators to connect directly to the server hosting the Java servlet instead. In this use case, annotators work on each sentence in the order that they appear in the input file. In the future, we would like to add more sophisticated methods for assigning particular sentences to specific annotators as well as tools that would help supervisors review annotators' work.

3. Related Work

3.1. Treebanking Tools

Perhaps the best known tool for treebanking is the tree editor TRED (Pajas and Štěpánek, 2008). TRED is a programmable graphical user interface for editing and viewing parse trees, including both constituent and dependency trees, and has been used for several treebanking projects, including the Prague Arabic Dependency Treebank (Hajič et al., 2004). It has a wide array of functionality—more than our system—but can be difficult to set up and learn. One of TRED's greatest limitations is that it is a standalone application written in Perl. In contrast, CROWDTREE is designed to run in modern web browsers, making it comparatively simple to deploy, especially for geographically distributed annotators. Some aspects of the visual layout and styling of CROWDTREE were inspired by TRED.

Another web-based tool that can be used for treebanking is WEBANNO (Yimam et al., 2013; Yimam et al., 2014; de Castilho et al., 2016). It builds upon the BRAT interface (Stenetorp et al., 2012) and supports multiple layers of annotation across a variety of linguistic annotation tasks. Additionally, WEBANNO is designed to support annotation across geographically distributed sites and has an integrated machine learning model for suggesting span annotations. It may, however, be more appropriate for semantic labeling tasks, such as word sense disambiguation (WSD) and semantic role labeling (SRL), than for syntactic annotation; although compact, we hypothesize that its horizontal word layout, an example of which is depicted in Figure 3, is slower for treebanking and more exhausting to annotators because they have to follow curved arcs with their eyes and pay attention to the arrowheads at the ends in order to understand the directionality of the dependencies.

Three other dependency tree editors with horizontal layouts similar to WEBANNO but designed with a more focused range of functionality are ARBORATOR (Gerdes, 2013), UD ANNOTATRIX (Tyers et al., 2018), and DGANNOTATOR[4].

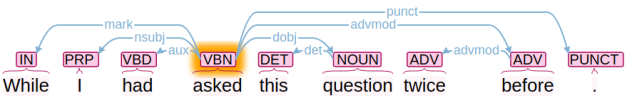


Figure 3: Screenshot showing the type of horizontal layout used in BRAT, WEBANNO, and most other dependency annotation tools.

[4]http://medialab.di.unipi.it/Project/QA/Parser/DgAnnotator/

3.2. Crowdsourcing Parsing

To date, there have been very few efforts to crowdsource parsing and no efforts, to the best of our knowledge, to do so directly with a full parse tree editor like CROWDTREE outside small classroom studies like that of Gerdes (2013).

In one notable ongoing effort, researchers have built and deployed a Game With a Purpose (GWAP) (Von Ahn, 2006) called ZOMBILINGO in order to crowdsource a French treebank (Fort et al., 2014; Guillaume et al., 2016; Fort et al., 2017). ZOMBILINGO participants work on one attachment decision at a time, using the metaphor that the governing word is devouring the child just as zombies seek out brains. Participants must complete a training phase for each dependency relation they work on and can not work on the relations deemed more difficult until they have demonstrated some skill on less challenging dependency relations. ZOMBILINGO relies entirely on volunteers and is a standalone website[5] not designed for integration with Mechanical Turk or similar crowdsourcing platforms.

Another notable effort is that of He et al. (2016), who, in a first step toward human-in-the-loop parsing, crowdsource individual attachment annotations by asking annotators multiple choice questions automatically generated using parse trees produced by an existing parser. For example, to resolve the attachment of a relative clause with a root verb of *sells*, the question posed to the annotators would be something like, "What *sells* something?", and the available answers would be the list of noun phrases that appear prior to the relative clause. He et al. are able to achieve some modest gains, improving by 0.2 F1 on their in-domain corpus and 0.6 F1 on their out-of-domain corpus.

3.3. Crowdsourcing combined with Machine Learning

Even though using crowdsourcing to collect data for processing with machine learning algorithms has been widespread for a number of years, it is still relatively rare to run machine learning algorithms at the same time that the crowd is actively annotating. This is especially true in the field of natural language processing (NLP). One notable exception to this rule is the work of Laws et al. (2011), who ran active learning processes while crowdsourcing Named Entity Recognition (NER) and sentiment detection tasks on Mechanical Turk. To the best of our knowledge, no one has utilized machine learning in any similar fashion while crowdsourcing parse trees.

4. Conclusion

In this paper, we introduce CROWDTREE, an interactive graphical editor for dependency tree annotation. Based upon standard web technologies, the system is compatible with modern browsers and, thus, easy to deploy to geographically distributed annotators. Our system's integrated back end parsing model makes it suitable for human-in-the-loop or even crowd-in-the-loop annotation, where the integrated parsing model learns as the annotation proceeds so that, ideally, the amount of effort required per sentence decreases as time goes on. CROWDTREE supports left-to-right and right-to-left scripts, labeled and unlabeled dependency annotation, and can run in conjunction with Mechanical Turk (via *ExternalQuestion* HITs) or in standalone mode. To the best of our knowledge, it is the first dependency tree editor designed for use with a large online crowdsourcing platform; thus, we expect it to prove useful to the wider computational linguistics research community and intend to release the code open source in the near future, likely as a repository on the U.S. Army Research Laboratory's GitHub site [6].

5. Future Work

We built CROWDTREE to explore the feasibility of crowdsourcing dependency parse annotation and to demonstrate the value of utilizing human-in-the-loop machine learning during the parse annotation process. Accordingly, we will be posting HITs on Mechanical Turk with a variety of parameterizations and publishing our findings. We hope to show that annotators without extensive linguistic training (e.g., Turkers) can quickly, cheaply, and accurately construct dependency trees and that integrated machine learning greatly benefits the overall process.

Furthermore, we intend to expand the system's machine learning capabilities by adding support for one or more active learning schemes. One obvious option would be to host multiple parsing models on the server and to select sentences for annotation based upon the level of disagreement between the parsers. We are considering the possibility of training one (or perhaps more) models per annotator during the annotation process, although implementing this in such a way as to avoid overtaxing available memory and computational resources may be a challenge.

A number of other extensions could be made to improve the tool's versatility in order to support a greater range of treebanking projects. We envision supporting empty nodes in order to handle traces and other phenomena, and we would like to support multiple syntactic word nodes for a given surface token, which would be valuable functionality for languages that make frequent use of clitics, such as Arabic. Finally, we hope to eventually expand the capabilities of CROWDTREE to support use cases beyond dependency parsing, such as part-of-speech tagging, semantic role labeling, or even language learning.

Acknowledgements

We wish to thank the anonymous reviewers for their comments and the Army Educational Outreach Program (AEOP) for the College Qualified Leaders (CQL) undergraduate internship program.

References

Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2003). The Prague Dependency Treebank. In Anne Abeillé, editor, *Treebanks: Building and Using Parsed Corpora*, pages 103–127. Springer Netherlands.

de Castilho, R. E., Mujdricza-Maydt, E., Yimam, S. M., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C.

[5] https://zombilingo.org/

[6] https://github.com/usarmyresearchlab/

(2016). A Web-Based Tool for the Integrated Annotation of Semantic and Syntactic Structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 76–84.

Fort, K., Guillaume, B., and Chastant, H. (2014). Creating Zombilingo, a Game With A Purpose for dependency syntax annotation. In *Proceedings of the First International Workshop on Gamification for Information Retrieval*, pages 2–6.

Fort, K., Guillaume, B., and Lefebvre, N. (2017). Who wants to play Zombie? A survey of the players on ZOMBILINGO. In *Proceedings of Games4NLP: Using Games and Gamification for Natural Language Processing*.

Gerdes, K. (2013). Collaborative dependency annotation. In *Proceedings of the Second International Conference on Dependency Linguistics (DepLing 2013)*, pages 88–97.

Goldberg, Y. and Elhadad, M. (2010). An Efficient Algorithm for Easy-First Non-Directional Dependency Parsing. In *Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2010)*.

Guillaume, B., Fort, K., and Lefebvre, N. (2016). Crowdsourcing Complex Language Resources: Playing to Annotate Dependency Syntax. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 3041–3052.

Hajič, J., Smrz, O., Zemánek, P., Šnaidauf, J., and Beška, E. (2004). Prague Arabic Dependency Treebank: Development in Data and Tools. In *Proceedings of the NEMLAR International Conference on Arabic Language Resources and Tools*, pages 110–117.

He, L., Michael, J., Lewis, M., and Zettlemoyer, L. (2016). Human-in-the-Loop Parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 2337–2342.

Laws, F., Scheible, C., and Schütze, H. (2011). Active Learning with Amazon Mechanical Turk. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 1546–1556.

Little, A. and Tratz, S. (2016). EasyTree: A Graphical Tool for Dependency Tree Annotation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):330.

McDonald, R. and Nivre, J. (2007). Characterizing the Errors of Data-Driven Dependency Parsing Models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 122–131.

Nivre, J. (2009). Non-projective dependency parsing in expected linear time. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL-IJCNLP 2009)*, pages 351–359.

Pajas, P. and Štěpánek, J. (2008). Recent Advances in a Feature-Rich Framework for Treebank Annotation. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008)*, pages 673–680.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). brat: a Web-based Tool for NLP-Assisted Text Annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2012)*.

Tratz, S. and Hovy, E. (2011). A Fast, Accurate, Non-Projective, Semantically-Enriched Parser. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1257–1268.

Tyers, F. M., Sheyanova, M., and Washington, J. (2018). UD Annotatrix: An Annotation Tool for Universal Dependencies. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories (TLT16)*, pages 10–17.

Von Ahn, L. (2006). Games with a Purpose. *Computer*, 39(6):92–94.

Yimam, S. M., Gurevych, I., Eckart de Castilho, R., and Biemann, C. (2013). WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013): System Demonstrations*, pages 1–6.

Yimam, S. M., Biemann, C., de Castilho, R. E., and Gurevych, I. (2014). Automatic Annotation Suggestions and Custom Annotation Layers in WebAnno. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014): System Demonstrations*, pages 91–96.

Building Universal Dependency Treebanks in Korean

Jayeol Chun[1], Na-Rae Han[2], Jena D. Hwang[3], Jinho D. Choi[1]
Emory University[1], University of Pittsburgh[2], IHMC[3]
Atlanta GA 30322[1], Pittsburgh, PA 15260[2], Ocala, FL 32502[3]
che.yeol.chun@emory.edu, naraehan@pitt.edu, jhwang@ihmc.us, jinho.choi@emory.edu

Abstract

This paper presents three treebanks in Korean that consist of dependency trees derived from existing treebanks, the Google UD Treebank, the Penn Korean Treebank, and the KAIST Treebank, and pseudo-annotated by the latest guidelines from the Universal Dependencies (UD) project. The Korean portion of the Google UD Treebank is re-tokenized to match the morpheme-level annotation suggested by the other corpora, and systematically assessed for errors. Phrase structure trees in the Penn Korean Treebank and the KAIST Treebank are automatically converted into dependency trees using head-finding rules and linguistic heuristics. Additionally, part-of-speech tags in all treebanks are converted into the UD tagset. A total of 38K+ dependency trees are generated that comprise a coherent set of dependency relations for over a half million tokens. To the best of our knowledge, this is the first time that these Korean corpora are analyzed together and transformed into dependency trees following the latest UD guidelines, version 2.

Keywords: universal, dependency, conversion, korean, treebank

1. Introduction

The Universal Dependencies (UD) project has brought on an increasing momentum to the research community for finding morphological patterns and syntactic relations appropriate to multiple languages (Zeman et al., 2017). The UD project has facilitated collaborative work among several organizations for 70+ languages, and inspired computational linguists to further analyze both resource-rich and -poor languages by suggesting universal guidelines that help them create and augment treebanks in different languages. The UD project has also promoted research on cross-lingual learning that explores the possibility of adapting statistical parsing models from one language to another (McDonald et al., 2013).

Several treebanks had been introduced for Korean, all of which comprised annotation of morphemes and phrase structure trees (Choi et al., 1994; Han et al., 2002; Hong, 2009), each following its own set of guidelines. Phrase structure trees in these treebanks had been converted into dependency trees using head-finding rules and linguistically-motivated heuristics, and used to evaluate Korean dependency parsing performance (Choi and Palmer, 2011; Choi, 2013). The previous efforts did not, however, focus on the compatibility among dependency trees converted from different corpora, resulting in the generation of a distinct set of dependency relations for each treebank.

This paper presents three dependency treebanks in Korean, derived from existing corpora and pseudo-annotated by the latest UD guidelines, version 2. The motivation behind this study is to make a comprehensive analysis between these corpora and convert phrase structure trees across different treebanks into dependency trees with consistent relations, providing a large corpus of compatible dependency trees. The contributions of this work are as follows:

- The Google UD Korean Treebank is manually assessed and systematically corrected (Section 3.).
- Phrase structure trees in both the Penn Korean Treebank and the KAIST Treebank are converted into dependency trees using the UD guidelines (Sections 4. and 5.).
- Corpus analytics are provided that include statistics of the new dependency treebanks, and remaining issues with the current annotation (Section 6.).

To the best of our knowledge, this is the first time that these Korean corpora are analyzed together and transformed into dependency trees following the latest UD guidelines.

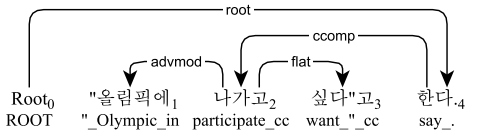
2. Related Work

Petrov et al. (2012) introduced the universal part-of-speech tagset and provided a mapping from 25 different treebank tagsets to this universal set. They showed that parsing performance using the universal part-of-speech tagset was comparable to the one using the original tagsets. McDonald et al. (2013) presented the universal dependency annotation and provided pseudo and manually annotated dependency treebanks for 6 languages. They showed promising results for cross-lingual parsing and initiated the effort for developing universally acceptable grammars. The official UD project started with a group of 10 languages (Nivre et al., 2015) and has expanded to over 70 languages. Recently, this project organized the CoNLL'17 shared task on multilingual parsing, involving over 40 languages (Zeman et al., 2017).

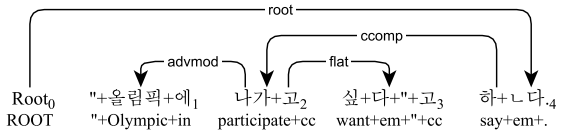
In addition, the Sejong Treebank, consisting of phrase structures trees for 60K sentences from 6 different genres of text released by Hong (2009), were converted into dependency trees by Choi and Palmer (2011). Despite of its large size, the Sejong Treebank is excluded from this work due to the license restriction. Hani corpora (Park, 2017) is also an effort annotated under UD guidelines; however, published exposition of this work has not yet been made available.

3. Google UD Korean Treebank

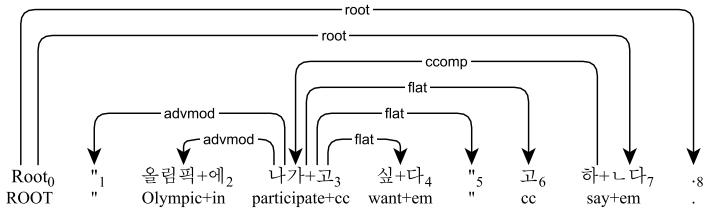
McDonald et al. (2013) provided the Google UD Treebanks comprising 6K sentences scraped from weblogs and newswire, annotated under the universal dependency guidelines for 6 languages including Korean. Because these treebanks were annotated before the official UD project started, the guidelines under which the Korean treebank was created



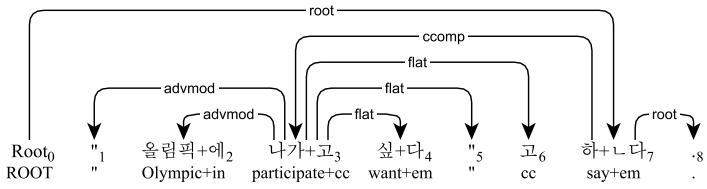
(a) A sample dependency tree from the original GKT.



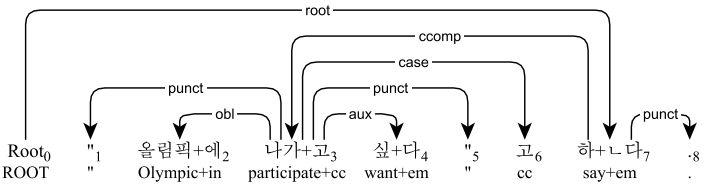
(b) After morphological analysis (Section 3.1.).



(c) After tokenization (Section 3.2.).



(d) After head ID remapping (Section 3.4.).



(e) After dependency relabeling (Section 3.5.).

Figure 1: Step-by-step illustration of our correction procedure of GKT (`cc`: coordinating conjunction, `em`: ending marker).

differed significantly from that of the version 2 of the UD (UDv2). The Google UD Korean Treebank (GKT) was automatically converted to follow the UDv2 guidelines, and distributed as a part of the CoNLL'17 shared task datasets. We perform a manual check over GKT to determine whether or not this automatic conversion generated sound dependency relations and carry out systematic correction.

3.1. Morphological Analysis

Korean is an agglutinative language with highly productive verbal and nominal suffixation, and limited prefixation. Without morphological analysis, then, any system that solely relies on surface forms must contend with the sparsity issue. As McDonald et al. (2013) points out, the automatic tokenization carried out for the original GKT was generally too coarse-grained; the suffixes or particles were left in with the tokens, indicating the necessity for future improvements through manual revision and annotation.

To help remedy this problem, we augment GKT with automatic morphological analysis obtained by the KOMA tagger, a general-purpose morphological analyzer for Korean (Lee and Rim, 2009) that produces the morpheme tagset defined by the Sejong Treebank (Hong, 2009). Figure 1(b) shows the morphological analysis of the original sentence in 1(a). The full morphological analysis is included for each token as the last column in our dataset.

3.2. Proper Tokenization

The tokenization in GKT does not split out the inflectional and derivational particle as separate tokens, nor are the punctuations tokenized. While a complete retokenization of particles in GKT is beyond the scope of this study, since improper tokenization of punctuation can lead to inappropriate dependency relations, we tackle the tokenization of symbols and punctuation marks for the proper configuration of the dependency relations. The morphological analysis from the KOMA tagger enables us to recognize symbols as well as particles so that they are split into separate tokens in our corpus. This is exemplified in Figure 1(c), where the two double quotes found in the 1st and 3rd tokens and the period in 4th token, are retokenized. Dependency labels for these new tokens are inferred from their morpheme tags. Over 9K tokens with embedded punctuation are revised, resulting in 3K additional tokens.

3.3. Part-of-speech Tags Relabeling

Once properly tokenized, measures are taken to assign appropriate parts-of-speech (POS) tags to separated tokens based on their morphemes. Note that the original GKT provides two POS tags for each token (columns 4 and 5), first of which is UDv2 compliant. Our relabeling focuses on replacing the first set of POS tag, and for the sake of consistency with other corpora, the secondary POS column is removed from our corpus.

3.4. Head ID Remapping

With tokenization and POS assignment complete, the head IDs of the separated tokens are redirected. In general, the word inherits the original head ID while the punctuation points to the previous token (i.e., token from which the punctuation was split) as seen in token 8 in Figure 1(d).

An exception is made for quotations or parenthetical phrases. Based on the observation that in general a quotation forms a sentence, a quotation (marked by quotation marks (e.g., “ ”) and seen in the 1st and 3rd token in Figure 1(b)) will feature its own sub-dependency tree where only its root will link to an element outside of the quotation. Therefore, the root of the sub-dependency tree is located by finding the link from within the quotation to an outside element. Punctuation points to the head of the quotation, as seen with 1st and 5th tokens in the Figure 1(d).

In the case of parenthetical expressions involving (), <>, [], ‘’ and ≪≫, we found that in the vast majority of cases, the elements within the parenthetical symbols were supplementary phrases describing a preceding token. This being so, the head of the parenthetical phrase is assigned to the rightmost element[1]. When the parenthetical expression forms a single token with the preceding word as seen in Figure 2, the token preceding the parenthetical expression inherits the original head ID and becomes the head of the root of the parenthetical expression. If there are any case particles attached to the right of the parenthetical (see token 6 in the same figure), then the case markings are also made dependent on the token preceding the parenthetical expression.

[1]Note that Korean is a head-last language.

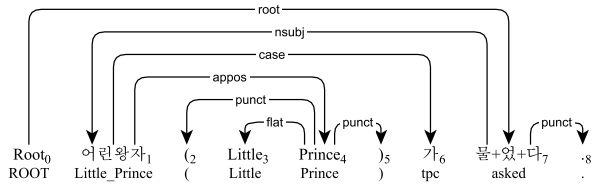


Figure 2: Example dependency tree with a parenthetical expression (tpc: topic marker).

3.5. Dependency Relabeling

Since the CoNLL'17 shared task, UDv2 has undergone changes that were not reflected in GKT. Thus, we apply morpheme-level rules to GKT and relabel all dependency relations to reflect the latest updates in UDv2. In Figure 1(e), the 2nd and 3rd tokens translate to *Olympics+in* and *participate*, respectively. Previous UDv2 considers *Olympics+in* an adverbial modifier (advmod) of *participate*, which is relabeled as an oblique (obl) in our corpus, as specified in the newest version of UDv2.

3.6. Lexical Correction

We manually assess the entire GKT for spelling errors. Social media is one of the main sources for GKT, which include a disproportionately large number of misspellings. Some are common incorrect spellings (e.g., 왠만하면 → 웬만하면) or deliberate non-standard forms known as ‘netspeak’ (e.g., 시른 → 싫은), while the rest are simple errors. Additionally, the HTML entity symbols are replaced with corresponding lexical symbols (e.g, & → &). The corrected spellings, 146 tokens in total, are provided in the lemma column.

4. Penn Korean Treebank

Han et al. (2006) created the Penn Korean Treebank (PKT) consisting of manual annotation of morphemes and phrase structure trees for 15K sentences from newswire in Korean. PKT is the only Korean treebank including annotation of empty categories, which enables to generate non-projective dependencies. The previous version of PKT (Han et al., 2002), which included phrase structure trees for 5k sentences from a military corpus—known as the Virginia corpus, is excluded from our conversion due to the lack of generality in its source, the military domain.

4.1. Empty Categories

Empty categories denote nominal units that point to the location of their antecedent syntactic elements found elsewhere in the sentence. In dependency structure, they serve to capture long-distance dependencies at the cost of introducing non-projective dependencies in the resultant tree. PKT features four empty categories exemplified in Figure 3: (1) trace *T* seen on line 3, (2) dropped subject *pro* seen on line 1, (3) empty operator *op* seen on line 0, and (4) ellipsis *?* seen on line 7.

4.1.1. Trace

An argument that precedes its subject leaves in its place a trace *T*. Given a terminal node that represents a trace like (NP-OBJ *T*-1) in line 3 in Figure 3, we find its

antecedent, (WHNP-1*op*) in line 1. Then we reorder the sentence in such a manner that the subtree with the non-terminal node as a root is extracted out of its position and inserted in place of the trace node, resulting in Figure 4.

```
0: (S (NP-SBJ (S (WHNP-1 *op*)
1:                (S (NP-SBJ *pro*)
2:          (VP (NP-ADV 어제/NNC)
3:              (VP (NP-OBJ *T*-1)
4:               사/VV+은/EAN))))
5:        (NP 아이폰/NPR+은/PAU))
6:    (ADJP (NP-COMP 어디/NPN+에/PAD)
7:          (VJ *?*))
8: ?/SFN)
```

Figure 3: Examples of 4 types of empty categories: *op*, *pro*, *T*, *?*.

```
0: (S (NP-SBJ (S (S (NP-SBJ  *pro*)
1:          (VP (NP-ADV 어제/NNC)
2:              (VP (NP-OBJ (WHNP-1 *op*))
3:                  사/VV+은/EAN))))
4:       (NP 아이폰/NPR+은/PAU))
5:    (ADJP (NP-COMP 어디/NPN+에/PAD)
6:          (VJ *?*))
7: ?/SFN)
```

Figure 4: The example in Figure 3 after trace mapping.

4.1.2. Empty Assignment and Empty Operator

Dropped arguments are represented by *pro* and relative clauses are represented by *op*. No explicit steps are taken to reorder sentence structures with these empty categories.

4.1.3. Ellipsis

Elided elements are indicated with *?* in PKT, which can result from a dropped predicate in a matrix clause (Figure 3) or when two clauses are coordinated with an implicitly shared predicate (Figure 5). In the first case, resolving the predicate will involve contextual information and therefore is outside of our project's scope. In the second scenario, mapping ellipsis must be performed intra-sententially: the first step is locating the predicate that has been 'deleted', and point to it as a head. PKT however does not provide an index that links the ellipsis token and its antecedent like it does with empty operators, presumably due to the fact that not all ellipses have in-sentence antecedents. To remedy this, we represent this relationship as a fixed conjunct, as seen with the 3rd and the 7th token in Figure 5. The relationship is established through simple heuristics of matching constituency tags at phrasal and morpheme level as well as functions tags if they exist.

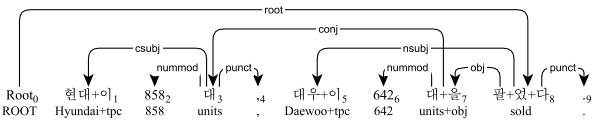


Figure 5: Example dependency tree with Ellipsis (obj: objective case particle).

4.2. Coordination

Following the guideline of Choi and Palmer (2011), each conjunct points to its right sibling as its head so that the rightmost conjunct becomes the head of the phrase. Because PKT does not offer the conjunctive function tag, our conversion discovers coordination structure by applying a set of heuristics[2]. An example of the coordination structure is shown in Figure 4.2., where 호박 (*pumpkin*) is the head of its left sibling 양파와 (*Onion+tpc*), and 오이가 (*Cucumber+tpc*) is made the head of the entire noun phrase involving the coordinated structure.

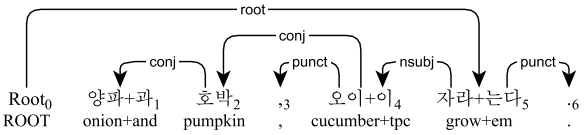


Figure 6: Sample PKT dependency tree with coordination.

4.3. Part-of-speech Tags

The POS tags are manually mapped from PKT to UDv2;[3] for the most part, this mapping is categorical. One exception is DAN, determiner-adnominal, which encompasses two semantically distinct subgroups: (1)demonstrative prenominals (e.g., 이 (*this*), 그 (*the*), 저 (*it*)) and (2)attribute adjectives that lack predicative counterparts (e.g., 새 (*new*), 헌 (*old*)). The former is mapped to DET (determiner); the latter to ADJ (adjective). Additionally, we identify nominal and verbal particles whose function are to encode conjunction and assigned them to the appropriate UDv2 POS tags. PCJ (conjunctive post-position) is singled out and assigned to CCONJ (conjunction), while the remaining post-position categories (PCA, PAD, PAU) are mapped to ADP (adposition). The ECS (coordinate, subordinate, adverbial) verbal endings require additional attention to context: they are categorized as CCONJ when they are considered coordinating verbs or verb phrases, and as SCONJ when considered coordinating clauses. All remaining verbal endings are categorized as PART (particle) along with copula (CO) and suffixes (X*).

4.4. Dependency Relations

The establishment of dependency relations starts with handling empty categories, discussed in Section 4.1. Then each node is assigned its head with head-percolation rules based on Table 1. The dependency relationship between the node and its head is inferred by investigating the function tags, phrasal tags and morphemes.

5. Kaist Treebank

Choi et al. (1994) created the KAIST Treebank (KTB) containing phrase structure trees for 31K sentences from various sources including literature, newswire, and academic manuscripts. Trees in this corpus were converted into dependency trees and used as a part of the shared task on parsing

[2]A simpler version of the heuristics used for PKT is exemplified by Algorithm 1, that is, coordination heuristics for KAIST.

[3]The mappings between the POS tagsets from PKT, KTB, and UDv2 can be found from our project site.

morphologically rich languages (Choi, 2013). Unlike PKB, KTB does not include empty categories and function tags, which renders the dependency conversion more challenging.

5.1. Coordination

Coordination in KTB is discovered and handled by Algorithm 1, which calls Algorithm 2 to check whether a given phrase or a sentence contains a coordination. However, the lack of empty categories in KTB, and hence the lack of representation of verb ellipsis, is the most notable difference between the two corpora. As it was for PKT, the rightmost conjunct becomes the head of the coordination.

Algorithm 1: find_coordination(C, R)

Input : A constituent C; the headrule R of C
$child$, $head \leftarrow$ `null`, `null`;
$children \leftarrow C$'s children list;
$type \leftarrow$ contains_coordination(C, $children$);
switch $type$ **do**
  case *0* **do**
    return $false$;
  case *1* **do**
    foreach $c \in children$ **do**
      if $child$ = `null` **then**
        $child$, $head \leftarrow c$, c;
      else
        if *c is* `sp` **then**
          c.set_head($child$, `punct`);
        else if *c ends with* `jcj` **then**
          $child$.set_head(c, `conj`);
          $child$, $head \leftarrow c$, c;
        else if *c is* `maj` **then**
          c.set_head(*C's right sibling*, `cc`);
        else
          $child$.set_head(c, `conj`);
          $child$, $head \leftarrow c$, c;
  case *2* **do**
    foreach *c in children* **do**
      if *child is null* **then**
        $child$, $head \leftarrow c$, c;
      else
        $child$.set_head(c, `conj`);
        $child$, $head \leftarrow c$, c;
if $type > 0$ **then**
  C.update_head(head);

5.2. Part-of-speech Tags

Similarly to PKT, the KTB POS tag mapping, for the most part, is categorical; exhibiting many-to-one mappings from KTB to UDv2. In some cases, KTB and UDv2 take a different slice through the semantics of what these tags represent. For example, while the KTB's case particles generally map to the UDv2's adpositions (`ADP`), the conjunctive case particles (`jcj`) in KTB functionally align with the UDv2's conjunctions (`CONJ`). Much like PKT, the ending particles (`x*`) in KTB are analyzed on the basis of semantic context: adveribial derivational suffixes (`xsa`) signal assignments to the UDv2's adverbs (`ADV`), while the rest of the ending particles in KTB are considered `PART` in UDv2.

Algorithm 2: contains_coordination(C, N)

Input : A constituent C;
  An ordered list N of child constituents of C
Output : The conjunct-flag, either 0, 1, 2 or 3
if C *is* `NP` **then**
  foreach c *in* N **do**
    if c is `maj` or `sp` **then return** 1
    if c ends with `ecc` or `jcj` **then return** 1
if C *is* `VP` *or* `ADJP` **then**
  foreach c *in* `N` **do**
    if c ends with `ecc` **then return** 2
return 0;

5.3. Dependency Relations

KTB dependency conversion follows the procedure outlined for PKT where the head of nodes is located with head-percolation rules based on Table 2.

While the dependency label inference benefits from the rich morphological analysis of KTB, the small number of phrasal tags and the absence of function tags has led to complications such as mapping of noun phrases ending with *jxt* to `dislocated`. Similarly to PKT, where *-SBJ* function tag denotes a subject node, KTB offers three morpheme tags for the same purpose: *jcs*, *jcc*, and *jxt*. However, while *jcs* and *jcc* roughly correspond to `nsubj` and `csubj`, `jxt` suggests that the phrase is the topic of the phrase or clause, but offers nothing informative in distinguishing whether it is in fact a subject (which it frequently is) and, if so, whether it is a clausal or nominative subject. Although UDv2 offers `dislocated` for topical elements ubiquitous in languages like Korean and Japanese, KTB offers no systematic way of distinguishing `dislocated` from its subject counterparts in `nsubj` or `csubj`.

Phrase	D	Headrules
S	r	VP;ADJP;S;NP;ADVP;*
VP	r	VP;ADJP;VV|VJ;CV;LV;V*;NP;S;*
NP	r	N*;S;N*;VP;ADJP|ADVP;*
DANP	r	DANP|DAN;VP;*
ADVP	r	ADVP;ADV;-ADV;VP;NP;S;*
ADJP	r	ADJP;VJ;LV;*
ADCP	r	ADC;VP;NP|S;*
ADV	r	VJ;NNC;*
VX	r	V*; NNX;*
VV	r	VV;NNC;VJ;*
VJ	r	VJ;NNC;*
PRN	r	NPR;N*|NP|VP|S|ADJP|ADVP;*
CV	r	VV;*
LV	r	VV;J;*
INTJ	r	INTJ;IJ;VP;*
LST	r	NNU;*
X	r	*

Table 1: Headrules for PKT. **Phrase** lists all phrasal tags in PKT. **D** denotes the search direction, `r` denotes searching for rightmost constituent, `*` denotes any tag headed by what follows, and `|` denotes logical `or`. Each **Headrule** gives higher precedence to the left tag on the list.

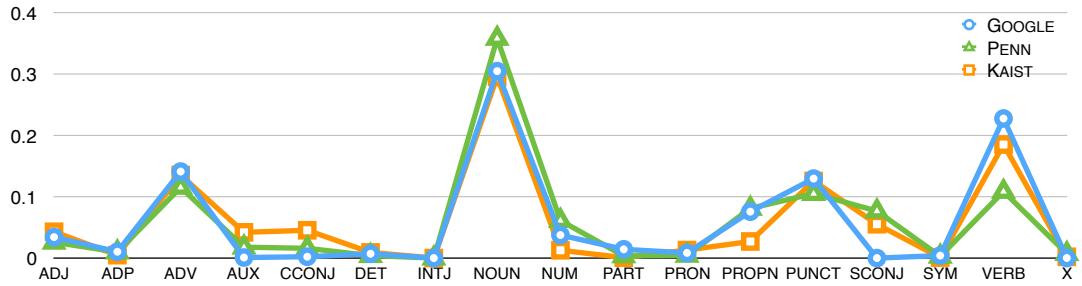


Figure 7: Distributions of part-of-speech tags for all three treebanks.

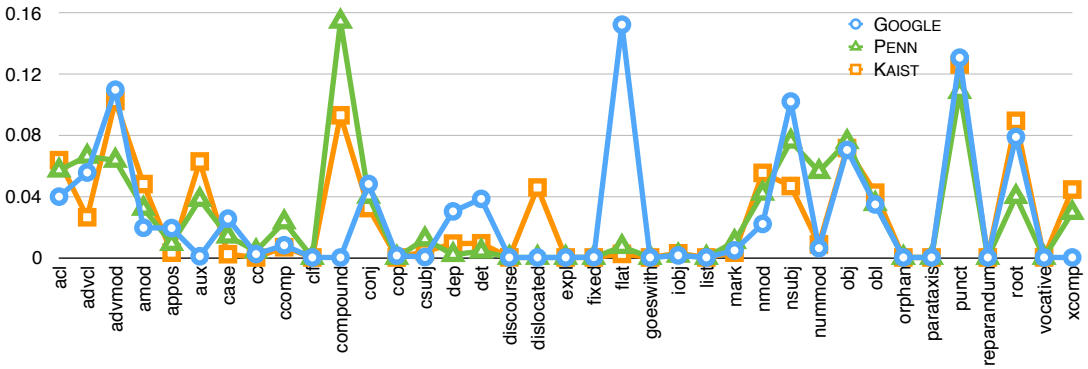


Figure 8: Distributions of the dependency labels for all three treebanks.

Phrase	D	Headrules
S	r	VP;ADJP;S;NP;ADVP;*
VP	r	pv*|pa*|n*|VP|NP;ADJP;S;*
NP	r	n*|f|NP|S|pv*|VP|pa*|ADJP;ADVP|MODP;*
ADJP	r	ADJP|pa*|n*;ADVP;VP;NP;S;*
ADVP	r	ADVP;VP;ma*;NP;S;*
AUXP	r	AUXP;NP;p*;n*;px;*
MODP	r	mm*;VP;ADJP;NP;*
IP	r	ii;p*;n*;ADVP;m*;*
X	r	*

Table 2: Headrules for KTB (see Table 1 for tabular details).

6. Corpus Analytics

6.1. Statistics of the New Dependency Treebanks

At approximately 26 dependency nodes per sentence, PKT includes on average the longest and complex sentences among the three corpora. This is likely reflective of the news domain PKT represents. KTB is by far the largest corpus in this study with its sentence complexity comparable to that of GKT at approximately 12 dependency nodes per sentence.

Number	GKT	PKT	KTB	Total
Sentences	6,339	5,010	27,363	38,712
Nodes	80,392	132,041	350,090	562,523

The frequencies of the POS tags in the three corpora are shown in Table 3. The three corpora shared NOUN, VERB, ADV and PUNCT as the top parts-of-speech (Figure 7). Beyond these four, no other POS reaches double-digit %, and the relative rankings start to diverge. In both PKT and GKT, PROPN (proper noun) is the fifth-highest ranking POS, while it is seen ranking much lower in KAIST, which instead has ADJ (adjective) taking the spot. NUM (number) is prominent in PKT which is likely a reflection of its news domain. Absence of the SCONJ in GKT is due to the tokenization that does not analyze particles as separate tokens. Notably, AUX (auxiliary)[4] and PART (particle)[5], which were entirely lacking in the original GKT , were partially introduced into the revised GKT as the result of tokenization of symbols and punctuation marks as discussed in Section 3.2..

The frequencies of dependency labels in the three corpora are shown in Table 4. The distributions of the dependency labels display intriguing trends across all treebanks (Figure 8). PKT and KTB appear consistent except in `compound`, `nummod`, `dislocated` and `nsubj`. As briefly mentioned, `compound` and `nummod` are likely domain-specific particularities. As for `dislocated` and `nsubj`, the discussion of 5.3. likely explains the discrepancy. GKT's abundant annotation of `flat` is a remnant of coarse tokenization that led to embedded tokens labeled `flat` as a whole.

6.2. Discussion

GKT While a number of salient errors has been handled in this work, our analysis show that there are a number of remaining issues with GKT that we strongly recommend be addressed in a future release of the data. The errors include structural problems, incorrect argument attachment, and in-

[4]All verbs were uniformly categorized as VERB in the original GKT. Given that auxiliary verb is a well-established category in Korean grammar, we find this a rather puzzling design decision.

[5]Particles were not tokenized in the original GKT.

Tag	Description	GKT	PKT	KTB
ADJ	Adjective	2,760	3,431	14,223
ADP	Adposition	1,791	1,251	1,498
ADV	Adverb	11,361	15,174	49,204
AUX	Auxiliary	74	2,263	12,906
CCONJ	Coordinating Conjunction	223	2,453	19,368
DET	Determiner	573	685	4,824
INTJ	Interjection	3	0	56
NOUN	Noun	32,345	46,866	105,193
NUM	Numeral	847	7,931	4,848
PART	Particle	31	464	268
PRON	Pronoun	682	857	7,712
PROPN	Proper Noun	490	12,257	12,366
PUNCT	Punctuation	10,440	13,428	38,925
SCONJ	Subordinating Conjunction	0	9,780	18,466
SYM	Symbol	328	376	260
VERB	Verb	18,431	13,855	59,273
X	Other	13	970	700
Total		80,392	132,041	350,090

Table 3: Frequencies of part-of-speech tags in the final resulting corpora.

correct dependency labelling.[6] Additionally, GKT shows a (mostly) consistent tendency to go with a head-first analysis in cases of conjunction (i.e., *talking* is the direct dependent of *reading* for conjunction *talking and reading*) and noun-noun compounds[7], both of which represent inconsistent treatments of a verb-final language.

Additionally, the GKT currently contains duplicates in the dataset, many of which are fairly complex sentences. Out of the 195 duplicates present in the data (out of total 6,339 sentence tokens), 113 duplicates appear verbatim in both the training and test sets (represents over 11% of the test data) and 28 duplicates cross over training and development sets (represents 3% of the development set), which indicates a flawed data sampling process.

PKT and KTB The conversion and error-analysis for PKT has undergone various iterations and the UDv2 compliant PKT data is now complete. PKT has been praised for its strong annotation consistency; that coupled with well-publicized documentation has enabled a quick and reliable implementation of the targeted conversion strategies.

KTB, our newest converted treebank, is near completion, however, there are still a few lingering issues that require attention. One issue that often came up was the treatment of grammaticalized multi-word expressions such as -ㄹ 것 이다 (*-l kesita*) and -ㄹ 수 있다 (*-l swu issta*). On the face of it, they involve dependent nouns 것 (*kes*, 'thing') and 수 (*swu*, 'way') respectively to literal translations of '... will be a thing' and 'there is a way to ...'. On the whole, however, they are grammaticalized forms that encode future/irrealis and epistemic modality, respectively: PKT acknowledges this and marks them as multi-word auxiliaries in annotation which facilitated our conversion process. In KTB, these forms had to be individually and lexically targeted to ensure parallel treatment. The Google Treebank, however, does not make such provision; as a matter of fact, it lacks AUX as a POS category altogether, which means this corpus remains disparate on this issue. This illustrates difficulty in achieving uniformity across multiple corpus resources by way of automatic and semi-automatic conversion.

[6] We suspect these errors were present in the original annotation of the corpus and propagated to the current distribution of CoNLL'17 shared task data.

[7] This is true even in a noun-noun compound where one of the noun explicitly case marked such as "**샐러드 바-를** 먹을 수 있다" (tr. **salad bar-obj** can eat), where *salad* is assigned the head even though *bar* is marked with the accusative case.

7. Conclusion

We present the manual assessment and revision process for the GKT, and the phrase-structure to UD conversion of Penn Korean and KAIST treebanks, discussing some of the statistics and the current issues relating the three presented treebanks. To the best of our knowledge, this is the first time that these three Korean corpora are converted together into dependency trees following the latest UD guidelines, resulting in a total of 38K+ dependency trees.

It is our expectation that the compilation of these treebanks will help facilitate further research in dependency parsing in Korean, where the lack of training data has remained an obstacle. Furthermore, we expect that the conversion methodologies described in this paper will serve as helpful resources to those wishing to carry out phrase-structure to dependency conversion for other corpora.

Future directions include further enhancements to the quality of treebanks established in this study and the development of parsers based on this dataset to aid further research in Korean NLP. All our resources including source codes and links to the corpora are provided at: https://github.com/emorynlp/ud-korean.

Tag	Description	GKT	PKT	KTB
acl	Clausal Modifier of Noun	3,198	1,488	21,468
advcl	Adverbial Clause Modifier	4,515	11,636	20,487
advmod	Adverbial Modifier	8,810	2,964	19,102
amod	Adjectival Modifier	1,566	1,595	16,584
appos	Appositional Modifier	1,544	1,182	1,059
aux	Auxiliary	64	4,807	18,935
case	Case Marking	1,624	1,548	1,343
cc	Coordinating Conjunction	223	785	5,234
ccomp	Clausal Complement	651	9,858	15,655
clf	Classifier	0	0	1
compound	Compound	0	28,908	24,696
conj	Conjunct	3,863	9,960	20,774
cop	Copula	102	418	303
csubj	Clausal Subject	21	8,014	1,202
dep	Unspecified Dependency	2,437	609	3,019
det	Determiner	3,077	685	4,824
discourse	Discourse Element	0	0	47
dislocated	Dislocated Elements	0	0	20,964
expl	Expletive	0	0	0
fixed	Fixed Multiword Expression	13	528	3,186
flat	Flat Multiword Expression	12,252	18	803
goeswith	Goes With	0	0	0
iobj	Indirect Object	108	222	967
list	List	0	0	0
mark	Marker	372	1,003	799
nmod	Nominal Modifier	1,761	5,555	22,045
nsubj	Nominal Subject	8,290	4,012	17,444
nummod	Numeric Modifier	489	154	3,295
obj	Object	5,801	9,823	23,605
obl	Oblique Nominal	2,784	3,357	11,577
orphan	Orphan	0	0	0
parataxis	Parataxis	0	0	0
punct	Punctuation	10,494	13,073	39,016
reparandum	Overridden Disfluency	0	0	0
root	Root	6,332	5,036	27,363
vocative	Vocative	0	0	15
xcomp	Open Clausal Complement	1	4,803	4,278
Total		80,392	132,041	350,090

Table 4: Frequencies of dependency labels in the final resulting corpora.

8. Bibliographical References

Choi, J. D. and Palmer, M. (2011). Statistical Dependency Parsing in Korean: From Corpus Generation To Automatic Parsing. In *Proceedings of IWPT workshop on Statistical Parsing of Morphologically Rich Languages*, SPMRL'11, pages 1–11.

Choi, K.-S., Han, Y. S., Han, Y. G., and Kwon, O. W. (1994). KAIST Tree Bank Project for Korean: Present and Future Development. In *In Proceedings of the International Workshop on Sharable Natural Language Resources*, pages 7–14.

Choi, J. D. (2013). Preparing Korean Data for the Shared Task on Parsing Morphologically Rich Languages. Technical Report 1309.1649, ArXiv.

Han, C.-H., Han, N.-R., Ko, E.-S., Palmer, M., and Yi, H. (2002). Penn Korean Treebank: Development and Evaluation. In *In Proceedings of the 16th Pacific Asia Conference on Language, Information and Computation*, PACLIC'02.

Han, N.-R., Ryu, S., Chae, S.-H., yun Yang, S., Lee, S., and Palmer, M. (2006). Korean Treebank Annotations Version 2.0. `https://catalog.ldc.upenn.edu/LDC2006T09`.

Hong, Y. (2009). 21st Century Sejong Project Results and Tasks (21세기 세종 계획 사업 성과 및 과제). In *New Korean Life (새국어생활)*. National Institute of Korean Language.

Lee, D.-G. and Rim, H.-C. (2009). Probabilistic Modeling of Korean Morphology. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(5):945–955, July.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N., and Lee, J. (2013). Universal Dependency Annotation for Multilingual Parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, ACL'13, pages 92–97.

Nivre, J., Bosco, C., Choi, J., de Marneffe, M.-C., Dozat, T., Farkas, R., Foster, J., Ginter, F., Goldberg, Y., Hajič, J., Kanerva, J., Laippala, V., Lenci, A., Lynn, T., Manning, C., McDonald, R., Missilä, A., Montemagni, S., Petrov, S., Pyysalo, S., Silveira, N., Simi, M., Smith, A., Tsarfaty, R., Vincze, V., and Zeman, D. (2015). Universal Dependencies 1.0. LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague.

Park, J. (2017). Universal dependencies for korean: Hani (ver1.0) [data set].

Petrov, S., Das, D., and McDonald, R. (2012). A Universal Part-of-Speech Tagset. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, LREC'12, pages 2089–2096.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, c., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, CoNLL'17, pages 1–19.

Moving TIGER beyond Sentence-Level

Agnieszka Falenska, Kerstin Eckart, Jonas Kuhn
Institute for Natural Language Processing
University of Stuttgart
{falenska, eckartkn, jonas}@ims.uni-stuttgart.de

Abstract

We present TIGER 2.2-doc – a new set of annotations for the German TIGER corpus. The set moves the corpus to a document level. It includes a full mapping of sentences to documents, as well as additional sentence-level and document-level annotations. The sentence-level annotations refer to the role of a sentence in the document. They introduce structure to the TIGER documents by separating headers and meta-level information from article content. Document-level annotations recover information which has been neglected in the intermediate releases of the TIGER corpus, such as document categories and publication dates of the articles. Additionally, we introduce new document-level annotations: authors and their gender. We describe the process of corpus annotation, show statistics of the obtained data and present baseline experiments for lemmatization, part-of-speech and morphological tagging, and dependency parsing. Finally, we present two example use cases: sentence boundary detection and authorship attribution. These use cases take the data from TIGER into account and illustrate the usefulness of the new annotation layers from TIGER 2.2-doc.

Keywords: Corpus Annotation, Treebank, Document Structure

1. Introduction

The TIGER corpus (Brants et al., 2004) is an important treebank for German. It has been frequently employed over the years: for several shared tasks on multilingual dependency parsing (Buchholz and Marsi, 2006) and parsing morphologically rich languages (Seddah et al., 2013; Seddah et al., 2014); as part of the HamleDT compilation (Zeman et al., 2012); and as training data for processing pipelines for variety of tasks, including coreference and bridging resolution (Björkelund et al., 2014), theoretical linguistics (Haselbach et al., 2012), and building morphological word-embeddings (Cotterell and Schütze, 2015).

TIGER consists of roughly 900,000 words or 50,000 sentences from German newspaper articles and its focus is on token and sentence level: For the first release of the TIGER treebank, the corpus was semi-automatically annotated and corrected with part-of-speech tags and constituent structures. The following versions enhanced the corpus with respect to the number of sentences and introduced morphological and lemma annotations. Additionally, parts of the corpus were released as dependency treebanks, e.g. a dependency version of TIGER 2.2 (Seeker and Kuhn, 2012).

During the early development of the corpus parts of the information about document boundaries were lost. The current distribution 2.2 does not contain any grouping of sentences into documents, i.e., the corpus consists of one long sequence of sentences. Therefore, it is not possible to annotate this dataset with any document-level information, such as anaphora and coreference relations, or discourse trees.[1] However, as opposed to datasets where contiguous documents are not available due to technical design or copyright reasons (Faaß and Eckart, 2013; Schäfer, 2015), TIGER does contain full documents and the only information lost is a mapping between sentences and documents to which they belong.

The purpose of this paper is to overcome shortcomings of the current TIGER distribution and to enable application of document-level tasks to it. We present TIGER 2.2-doc[2], a new set of annotations that divides the sentences of the corpus into documents, i.e., newspaper articles, and realigns articles with their categories and approximate publication dates using original sources. We also semi-automatically enrich documents with structure annotations (differentiating meta-level information from article's real content), authors, and authors' genders. Additionally, we release a new split into training, development and test sets.[3]

A corpus containing gold-standard syntactic annotations and document boundaries serves as a thorough platform for further annotations and applications, such as the frequently used German treebank TüBa-D/Z (Telljohann et al., 2015). We believe that adding new annotations to TIGER and introducing TIGER 2.2-doc will make the corpus an interesting resource for NLP researchers working on a variety of tasks. The wide range of annotations – from sentence boundaries, through syntax, to authors and categories – makes it possible to answer new questions. For example, document categories can be used to examine which types of articles are the most difficult to parse. Similarly, having gold syntax trees and gender of authors allows to analyze if some syntactic constructions are used more often by women than men. In this paper we present baseline experiments on lemmatization, part-of-speech (POS) tagging, morphological analysis, and dependency parsing for the new training/development split of TIGER. We also show pilot experiments on sentence boundary detection and authorship attribution in TIGER.

[1] Cf. guidelines like Riester and Baumann (2017) and Riester et al. (to appear) which have been applied to German text, taking documents into account.

[2] Persistent identifier (PID): `http://hdl.handle.net/11022/1007-0000-0000-8E50-6`

[3] In TIGER 2.2 one additional consequence of the sentence-level focus is that the typical training, development, and test set split does not take document borders into account and contains sentences from the same documents in different sets.

2. (Re-)introducing TIGER Documents

TIGER is provided in several representation formats, the set of which differs among the releases. In the Negra Export format (Brants, 1997), some metadata was encoded in a section called ORIGIN. An entry in this section consists of a numeric identifier, which we call the **ORIGINID**, information on an approximate publication time and, partially, information on article categories (news, feuilleton, etc.). According to the dates from the metadata, the corpus can be split into articles from years 1992, 1995 and 1997.

A sentence is related to the metadata via the ORIGINID. We consider a set of consecutive sentences annotated with the same ORIGINID an article and would like our documents to contain exactly one article each. However, the information in the corpus releases was not sufficient to comprehensively identify the document boundaries: (1) For the 1992 part, the metadata shows a different granularity by subsuming all articles under one ORIGINID, so no document boundaries could be extracted from the metadata for this part. (2) Part of the mapping between sentences and ORIGINID got lost during processing, such that several blocks of sentences were assigned a fall-back ORIGINID 0, which does not contain any information. (3) Some presumed articles contain more than one or only parts of articles, maybe due to extraction errors when the articles were selected from the newspaper data.

To introduce a full sentence to document mapping we used additional resources and some old work files from the creation phase of the corpus. Since the additional information usually referred only to one of the corpus parts (1992, 1995 or 1997), we treated each part separately. However, some common guidelines were applied:

1. TIGER 2.1 sentence numbers and segments are kept;
2. ORIGINIDs are kept to align the metadata;
3. When applying additional resources with article markup their boundaries should be reflected in the annotation as long as they do not conflict with the boundaries introduced by the ORIGINIDs.

TIGER 2.2-doc thus introduces **DOCUMENTIDs** X_Y where X contains an ORIGINID and Y is an addition to mark several documents within one ORIGINID.[4]

For the 1992 part we automatically extracted article borders from ECI/MC1 (European Corpus Initiative, 1994). After minimal manual post-processing all documents were read by a native speaker who also compared the proposed boundaries to the ECI/MC1 data.

For the 1995 part we applied an old work file. The file contained a full mapping of the ORIGINIDs but different sentence numbers and sentence segmentation. For each range with ORIGINID 0 the ORIGINIDs from the work file were automatically mapped when the sentences were identical and unique. Afterwards all documents were read by native speakers. Taking the work file into account, they assigned and corrected DOCUMENTIDs where necessary.

For the 1997 part we applied an old work file and a part of the DeReKo (Institut für Deutsche Sprache, nd). Again, identical and unique sentences with ORIGINID 0 were automatically mapped to the work file. Afterwards all documents were read by native speakers. Taking the work file and DeReKo into account, they assigned and corrected documentIDs where necessary. A range of sentences with ORIGINID 0 remained, for which only the metadata information was available. The missing DOCUMENTIDs were assigned manually based on the number of ORIGINIDs left for this part and the respective article categories.

During the process of (re-)introducing document boundaries, the annotators found special cases in the corpus. Some of these can be explained as processing artifacts, such as where a sentence from one article was copied accidentally into another article, while others reflect peculiarities of the newspaper, such as collection articles, where a set of independent newsflashes is combined under one heading and consequently marked with the same ORIGINID in the corpus. Collection articles however influence document-level applications such as coreference annotations, because one document consists of several small parts which usually do not share any referents. Based on Guideline 3 mentioned above, these collection articles were handled according to their markup in the external resources and treated either as one or as several documents. Respective comments are released with the annotation of TIGER 2.2-doc to let users decide to e.g. exclude some article types.

3. Further Annotations

TIGER 2.2-doc includes a mapping from sentences to 2,263 documents, with on average 22 sentences per document. The documents are split into training/development/test sets with 1,863/200/200 documents. An example document with 21 sentences is shown in Figure 1. Clearly the document in its original published form contained an internal structure – the first sentence was a title, followed by a subtitle, meta information about the author, and place and date of release. We introduce the structure to the documents and annotate them on sentence and document level. However, we exclude from this annotation process 100 collection documents, i.e., documents consisting of more than one article, since they have their own structure.

3.1. Sentence-level Annotations

We introduce a new sentence-level layer of annotations with three possible values (see Figure 1 for examples): (i) HEADER for all sentences which act as titles (in this version we do not distinguish titles from subtitles); (ii) META for all sentences which contain meta-level document information, as author, date of release of the article, notes for the reader; (iii) BODY for the actual content of the article.

We annotated all sentences with HEADER, META, or BODY semi-automatically. First all sentences were annotated automatically with a rule-based system. The system checked if a sentence ends with a sentence-final punctuation mark (. ? ; or !), is capitalized, contains verb, or contains one of manually selected key words, e.g., "Von" (eng. *by*), "Kommentar" (eng. *(editorial) comment)*, or "Seite" (eng. *page*). Then the annotations were checked manually by a native speaker and corrected. Approximately 1% of sentences received a wrong annotation during the automatic process, in most cases HEADER was incorrectly annotated as BODY.

[4] Both parts are four digit values with leading zeros, i.e. 0001_0005 is the fifth article found under ORIGINID 1.

1	SPD-Spitze stimmt Bosnien-Einsatz zu *Head of SPD [Social Democratic Party of Germany] agrees to Bosnia mission*	HEADER
2	Parteitag soll Engagement deutscher Soldaten “ ohne Kampfauftrag ” beschließen *Party convention is to give consent on German troops engaging in "non-combat" mission*	HEADER
3	Von Helmut Lölhöffel *By Helmut Lölhöffel*	META
4	MANNHEIM , 13. November .	META
5	Die SPD ist bereit , dem Einsatz von Bundeswehr-Transport- und Aufklärungsflugzeugen bei der Umsetzung eines Bosnien-Friedensplanes zuzustimmen . *The SPD is willing to go along with having German Federal Defense Forces operate transport and reconnaissance aircraft in implementing a road map for peace for Bosnia.*	BODY
6	Damit folgt die Partei der sozialdemokratischen Bundestagsfraktion , die dies schon vorher beschlossen hatte . *By doing so, the party follows the social democratic members of German parliament, who have already agreed on this.*	BODY
...	...	
21	Leitartikel auf Seite 3 *Editorial on page 3*	META

Figure 1: An example document (0834_0001) and its translation to English (translation is not a part of the treebank). Document-level annotations for this document are: category – NAC, author – Helmut Lölhöffel, author's gender – male, approximate date of publication – 1995-11-14. The publication date may differ from the one in the article ("13. November" v. "1995-11-14") because it is approximate and refers to the week in which the article was published.

In the final version 6% of all sentences is annotated with META and 9% with HEADER.

3.2. Document-level Annotations

Document-level annotations are categories, approximate dates of publication, and authors. The first two types were recovered from the metadata of the original TIGER sources, while authors were annotated semi-automatically (see details below). The distribution of document-level annotations across all documents is presented in Figure 2. Approximate publication dates are available for all documents, categories for 2,016 of them, and authors for 541. For 506 documents all three types of annotations are available.

Categories The document categories were recovered from existing sources – see Table 1 for details. For six big categories (news, economy, world news, feuilleton, politics, and scientific topics) both the category ID and meaning of it were recovered. 1,749 documents received one of these categories. For additional 267 documents (under ”Various categories” in the Table 1) only the category ID was restored. For 247 documents category is unknown.

Category		#documents	Meaning
Nachrichten	NAC	689	*News*
Wirtschaft	WIR	601	*Economy*
Other		267	*Various categories*
–		247	*Unknown category*
Aus Aller Welt	AAW	177	*World news*
Feuilleton	FEU	148	*Feuilleton*
Politik	POL	106	*Politics*
Wissenschaft	WIS	28	*Scientific topics*

Table 1: Document categories.

Publication Dates The publication dates available in the metadata are approximate publication dates for the articles, probably referring to the week in which an article was published. By means of the ORIGINID this information is available for all documents.

Authors Information about the author of the document comes from the article's content – see for example Sentence 3 in Figure 1. We used sentence-level annotations to collect two types of meta annotations: the author of the article and the gender of the author. First we found all sentences with META annotations. Then we used gold POS tags to filter phrases with NE tags. These phrases were manually checked and the correct authors were collected. Finally we used morphological features to mark genders of the authors. For example, token level annotations for Sentence 3 from Figure 1 are: Von/`APPR/-` Helmut/`NE/case=nom|num=sg|gender=masc` Lölhöffel/`NE/case=dat|num=sg|gender=masc`. We used POS tags to filter "Helmut Lölhöffel" as a possible author and values of the gender morphological feature to set the gender of the document's author to "male".
The information about authors is available for 541 TIGER documents: 8 of the articles were written by more than one author, all the other 533 articles were written by 243 different authors, among which 87 wrote more than one article. We present the distribution of authors in Figure 3.

4. Baseline Experiments

The TIGER corpus has been widely employed to evaluate methods predicting token-level annotations, as POS tags or dependency trees (Buchholz and Marsi, 2006; Seddah et al., 2014, among others). These token-level annotations do not differ between TIGER 2.2 and TIGER 2.2-doc. However, TIGER 2.2-doc is released with a new split into training, development, and test sets. To enable future comparisons, we present baseline results for lemmatization, part-of-speech tagging, morphological analysis, and dependency parsing for the new split.

Methodology and Tools We predict POS tags and morphological features with the state-of-the-art morphological

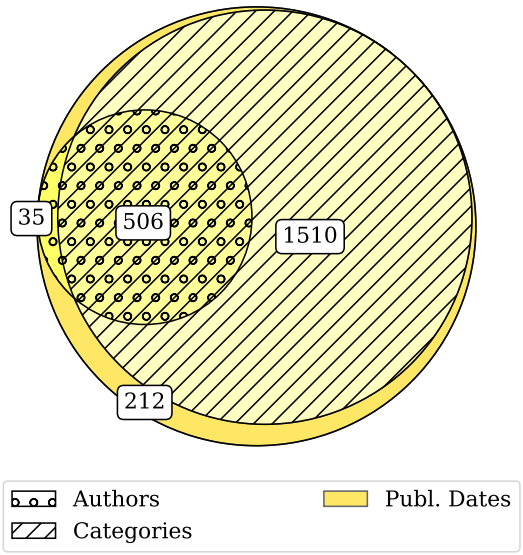


Figure 2: Number of documents annotated with different document-level annotations. Publication dates are available for all 2,263 documents. 2,016 of them (1510 + 506) are annotated with categories and 541 (506 + 35) with authors. For 506 documents all three types of annotations are available.

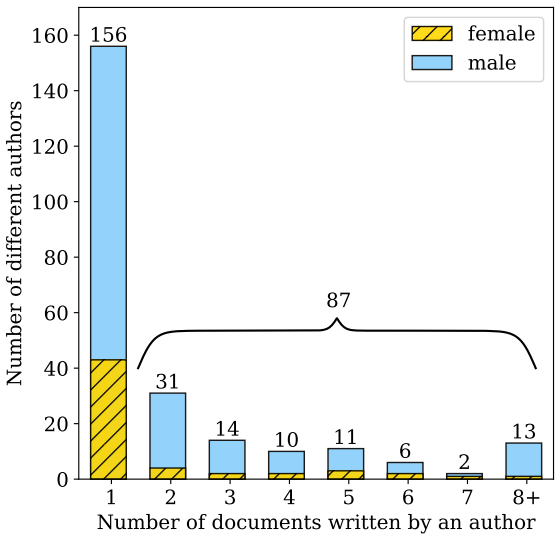


Figure 3: Distribution of authors of TIGER articles and gender of the authors. 8 articles which were written by more than one author are not included in the plot. Remaining 533 articles were written by 243 different authors, among which 87 wrote more than one article.

CRF-based tagger (Müller et al., 2013). We use the mate-tools for lemmatization.[5] For dependency parsing we apply a transition-based beam search parser by Björkelund and Nivre (2015) which uses the ArcStandard system extended with a Swap transition (Nivre, 2009). We train the parser on 10-fold jackknifed POS and morphological tags.

Results We provide results for the new development and test sets in Table 2.

	Lemma F1	POS F1	Morph F1	Parsing LAS
Dev	97.96	97.78	90.68	91.27
Test	97.91	97.86	90.63	92.53

Table 2: Baseline results for lemmatization, part-of-speech tagging, morphological analysis, and dependency parsing on the development and test sets.

5. Use cases

To illustrate the usefulness of the new annotations, we present two use cases: sentence boundary detection and authorship attribution. Both tasks take document borders into account and could not have been applied on TIGER before introducing the new annotation layers. While gold-standard sentence boundaries were available in TIGER 2.2, TIGER 2.2-doc introduces the distinction between sentence boundaries and document boundaries. The new information allows to train and evaluate sentence segmenters only on meaningful instances of sentence borders, and not the ones at the end of documents.

[5] https://code.google.com/archive/p/mate-tools/

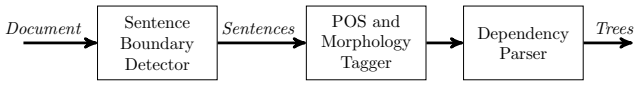
5.1. Case Study: Sentence Boundary Detection

The logical structure of a document describes of what units (e.g., titles, enumerations, tables) the document consists. Such information is useful for automatic document processing tasks, such as information extraction or summarization. Document structure reconstruction is the task of automatically recovering the logical structure of documents, for example from the results of an OCR process.

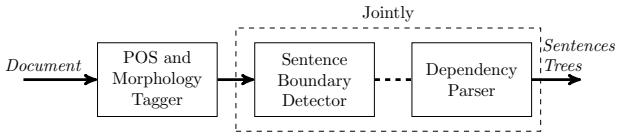
The original newspaper versions of TIGER articles were structured and contained visually distinguishable titles, subtitles, author's footnotes, etc. In the current version some of this information was reconstructed and marked with sentence level annotations, making TIGER 2.2-doc a possible experimental field for document structure reconstruction. In this paper we present pilot experiments for the first step of automatic document structure reconstruction: sentence boundary detection. We leave the next steps (e.g., prediction of sentence-level annotations) for future work.

Methodology Sentence boundary detection is often regarded as a solved task, at least in the domain of well-edited texts. Therefore, typical sentence boundary detectors focus on orthographic clues, as punctuation and capitalization marks. However, when moving to non-standard texts basic assumptions about punctuation and capitalization may be violated, thus rendering sentence segmentation more challenging task (Evang et al., 2013).

We have previously shown (Björkelund et al., 2016) that for texts where typical orthographic features are not present better sentence segmentation can be achieved by re-ordering the standard NLP pipeline (e.g., pipeline that applies lemmatization, part-of-speech tagging, and/or parsing). In the standard approach, sentence segmentation is regarded as the easiest task and is performed first (see Figure 4a). Alternatively, we trained a POS tagger on whole documents, applied it as the first tool in the pipeline, and



(a) standard pipeline



(b) re-ordered pipeline

Figure 4: Schematic view of pipelines applied in the experiments.

	Segm. F1	POS F1	Morph F1	Parsing LAS
Standard pipeline				
UDPIPE	82.34	97.66	90.45	87.42
MARMOT	89.21	97.74	90.58	88.47
Re-ordered pipeline				
JOINT	93.26	97.73	90.19	88.53
JOINT-REPARSED				88.92

Table 3: Results on the development set for two pipelines and three sentence segmenters.

performed sentence segmentation and parsing jointly (see Figure 4b). We showed that syntax is useful for sentence segmentation and that the joint method detects sentence boundaries better.

TIGER is built from newspaper articles which are usually considered well-edited texts and common sentence segmenters should perform well for them. However, TIGER documents also contain meta-level sentences and headers which are not straightforward to segment. We perform pilot experiments for sentence segmentation in TIGER, apply state-of-the art tools and investigate if the joint method of Björkelund et al. (2016) is a better choice for this dataset.

Tools We apply state-of-the-art sentence segmenters to establish baseline results:

UDPIPE (Straka et al., 2016): is a toolset performing, i.a., sentence segmentation, POS tagging, and dependency parsing. From all the functionalities of UDPIPE we employ only the sentence segmenter. It uses a single-layer bidirectional GRU network. The segmenter works on character level, i.e., for each character in text it predicts if a sentence ends after it. We use UDPIPE as a baseline following the recent CoNLL Shared Task 2017 (Zeman et al., 2017).

MARMOT: is the best performing baseline from Björkelund et al. (2016). It augments POS tags with information if a token starts a new sentence. Then, it trains the sequence labeler of Müller et al. (2013) on whole documents annotated with the augmented tags. The method applied to new documents jointly predicts sentence boundaries and POS tags.

For POS tagging, morphological analysis and parsing we apply the methods from Section 4 – namely CRF-based tagger by Müller et al. (2013) and transition-based parser by Björkelund and Nivre (2015).

For the re-ordered pipeline we train the same POS tagger but on whole documents instead of sentences. We employ the aforementioned parser extended to predict sentence boundaries (Björkelund et al., 2016) (referred to as **JOINT**). It predicts sentence boundaries and dependency trees jointly. For both pipelines we train parsers on 10-fold jackknifed POS and morphological tags. We calculate the results with the evaluation script from the CoNLL Shared Task 2017 (Zeman et al., 2017).

Results The results of the experiments are presented in Table 3 in the first column. We find that MARMOT outperforms the UDPIPE baseline by a big margin of almost 7 points. JOINT surpasses both baselines – by 4 points for MARMOT and almost 11 points for UDPIPE, making it a better choice for the first step of document structure reconstruction in the TIGER dataset.[6]

We also investigate how the next pipeline steps are influenced by different sentence segmentation results. POS tagging and morphological analysis are almost not influenced by the selection of the pipeline, i.e., regardless if they are applied to sentences or documents their results are very similar. When comparing parsing results we see that sentence boundary detection errors propagate through the pipeline. The final parser loses one point when applied to sentences predicted by UDPIPE instead of MARMOT. Interestingly, syntactic features in the JOINT method help the sentence boundary detection but not the other way around, i.e., its parsing result outperforms both baselines only slightly. To assess the influence of error propagation through the pipeline we follow Björkelund et al. (2016) and parse the sentences once again (denoted **JOINT-REPARSED**). In this scenario JOINT serves only as a sentence segmenter and parsing is performed separately. We find that the better sentence segmentation translates to better parsing results and JOINT-REPARSED outperforms both MARMOT and UDPIPE.

5.2. Case Study: Authorship Attribution

Authorship attribution is the task of determining the author of a document. TIGER is built from short newspaper texts which mostly belong to categories related to news. It is an interesting question if it is possible to track documents' authors in this domain. And if yes, if it is due to stylometric differences or content of the documents (e.g. authors tending to write documents on the same topics). We present pilot experiments for two authorship attributions tasks: predicting the author of the text and gender of the author.

Methodology and Tools In both of the tasks we use only sentences not labeled as META, as they contain implicit information on authors (names, surnames and cities). For gender prediction we use all 533 documents with author id (for a detailed distribution of documents and their authors

[6]We also applied the two described pipelines to an out-of-domain test suite (Seeker and Kuhn, 2014). Results showed that JOINT trained on the new TIGER 2.2-doc has an advantage over standard pipeline also for other diverse datasets – for example dvd player manuals.

see Figure 3). For author attribution we use only authors who wrote at least two documents (87 authors and 377 documents). We use two classification methods:

DELTA: the Burrows's Delta (Burrows, 2002) is the most established method for capturing stylometric differences. We calculate DELTA and perform classification with the R stylo package (Eder et al., 2013) with default parameters. To establish if the method captures stylometric or content differences we follow Schulz et al. (2016) and calculate DELTA in two ways: on all the words in the documents (DELTA–STYLE) and on content words (DELTA–CONTENT). As content words we select nouns, verbs and adjectives and filter them by gold-standard POS tags.

VERETAL: we copy the experimental setup from Verhoeven et al. (2016). We use LinearSVC from sklearn[7] with default parameters and use unigrams and bigrams of words and trigrams and tetragrams of characters as features.

We compare the two methods to two control baselines: weighted random baseline (**WRB**), which predicts a stratified random class, and majority baseline (**MAJ**), which predicts the most frequent class. We evaluate the accuracy with 10-fold cross-validation.

Results Table 4 presents results of the experiments. In the task of gender prediction documents are assigned only two classes and the methods achieve higher accuracy. Among all the methods, WRB is the worst and is able to correctly predict authors' gender only for 69.03% of the documents. DELTA slightly outperforms WRB, especially when using content words. This might be an indicator that the "author signal" in the TIGER documents is weak and classification methods capture more content than stylometric differences. Generally, DELTA is not able to outperform the simple MAJ baseline, which achieves accuracy of 79.91%. But VERETAL, a classifier with a richer set of features, proves to be the best method and outperforms MAJ by a margin of 2.43 points.

Results for author prediction are generally lower than for gender prediction because the task is harder – it involves assigning one of 87 classes. In this case DELTA is able to outperform both control baselines. Interestingly, this time style features give a small boost over the content ones. Again, VERETAL surpasses all the other methods with a big margin of more than 32 points.

	Author prediction	Gender prediction
WRB	2.71	69.03
MAJ	2.99	79.91
DELTA–STYLE	12.16	71.08
DELTA–CONTENT	9.75	72.97
VERETAL	44.41	82.34

Table 4: Accuracy (10-fold cross-validated) for predicting author (87 authors, 377 documents) and gender (female/male, 533 documents).

[7] scikit-learn.org/

6. Conclusion

We presented TIGER 2.2-doc, a new set of annotations for the frequently used German corpus TIGER. Prior releases of TIGER already contained token and sentence boundaries. TIGER 2.2-doc adds explicit document borders based on the newspaper articles, and a suggested split into training, development and test sets. We presented new document- and sentence-level annotations which broaden the range of possible applications of TIGER. We showed two example use cases which employ the new annotations: sentence boundary detection (as a part of document structure reconstruction) and authorship attribution. TIGER 2.2-doc is available by means of a persistent identifier.

7. Acknowledgements

We would like to thank Amir Zeldes and Florian Zipser for discussions on the (re-)introduction of the document borders and for help with finding the old work files, and our student annotators for patiently mapping and reading lots of old news. This work was supported by the German Federal Ministry of Education and Research (BMBF) via CLARIN-D and the German Research Foundation (DFG) via SFB 732, projects D8 and INF.

8. Bibliographical References

Björkelund, A. and Nivre, J. (2015). Non-deterministic oracles for unrestricted non-projective transition-based dependency parsing. In *Proceedings of the 14th International Conference on Parsing Technologies*, pages 76–86, Bilbao, Spain. Association for Computational Linguistics.

Björkelund, A., Eckart, K., Riester, A., Schauffler, N., and Schweitzer, K. (2014). The extended DIRNDL corpus as a resource for coreference and bridging resolution. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 3222–3228, Reykjavik, Iceland. European Language Resources Association (ELRA).

Björkelund, A., Faleńska, A., Seeker, W., and Kuhn, J. (2016). How to train dependency parsers with inexact search for joint sentence boundary detection and parsing of entire documents. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1924–1934, Berlin, Germany. Association for Computational Linguistics.

Brants, S., Dipper, S., Eisenberg, P., Hansen-Schirra, S., König, E., Lezius, W., Rohrer, C., Smith, G., and Uszkoreit, H. (2004). TIGER: Linguistic interpretation of a German corpus. *Research on Language and Computation*, 2(4):597–620.

Brants, T. (1997). The NEGRA export format. CLAUS Report 98, Universität des Saarlandes, Computerlinguistik, Saarbrücken, Germany.

Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 149–164, New York City. Association for Computational Linguistics.

Burrows, J. (2002). âĂŸDeltaâĂŹ: a measure of stylistic difference and a guide to likely authorship. *Literary and linguistic computing*, 17(3):267–287.

Cotterell, R. and Schütze, H. (2015). Morphological word-embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1287–1292, Denver, Colorado. Association for Computational Linguistics.

Eder, M., Kestemont, M., and Rybicki, J. (2013). Stylometry with R: a suite of tools. In *Digital Humanities 2013: Conference Abstracts*, pages 487–489. University of Nebraska, Lincoln Lincoln, NE.

Evang, K., Basile, V., Chrupała, G., and Bos, J. (2013). Elephant: Sequence labeling for word and sentence segmentation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1426. Association for Computational Linguistics.

Faaß, G. and Eckart, K. (2013). SdeWaC – a corpus of parsable sentences from the web. In Iryna Gurevych, et al., editors, *Language Processing and Knowledge in the Web*, volume 8105 of *Lecture Notes in Computer Science*, pages 61–68. Springer Berlin Heidelberg.

Haselbach, B., Eckart, K., Seeker, W., Eberle, K., and Heid, U. (2012). Approximating theoretical linguistics classification in real data: the case of German "nach" particle verbs. In *Proceedings of COLING 2012*, pages 1113–1128. The COLING 2012 Organizing Committee.

Müller, T., Schmid, H., and Schütze, H. (2013). Efficient Higher-Order CRFs for Morphological Tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA. Association for Computational Linguistics.

Nivre, J. (2009). Non-projective dependency parsing in expected linear time. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 351–359, Suntec, Singapore. Association for Computational Linguistics.

Riester, A. and Baumann, S. (2017). *The RefLex Scheme – Annotation Guidelines*, volume 14 of *SinSpeC. Working Papers of the SFB 732*. University of Stuttgart.

Riester, A., Brunetti, L., and Kuthy, K. D. (to appear). Annotation guidelines for questions under discussion and information structure. In Katharina Haude Evangelia Adamou et al., editors, *Information Structure in Lesser-Described Languages*. Benjamins, Amsterdam.

Schäfer, R. (2015). Processing and querying large web corpora with the COW14 architecture. In Piotr Bański, et al., editors, *Proceedings of Challenges in the Management of Large Corpora 3 (CMLC-3)*, Lancaster. UCREL, IDS.

Schulz, S., Kuhn, J., and Reiter, N. (2016). Authorship attribution of mediaeval German text: Style and contents in Apollonius von Tyrland. In *Digital Humanities 2016: Conference Abstracts*, pages 883–885, KrakÃşw, Poland.

Seddah, D., Tsarfaty, R., Kübler, S., Candito, M., Choi, J. D., Farkas, R., Foster, J., Goenaga, I., Gojenola Galletebeitia, K., Goldberg, Y., Green, S., Habash, N., Kuhlmann, M., Maier, W., Nivre, J., Przepiórkowski, A., Roth, R., Seeker, W., Versley, Y., Vincze, V., Woliński, M., Wróblewska, A., and de la Clergerie, E. V. (2013). Overview of the SPMRL 2013 shared task: A cross-framework evaluation of parsing morphologically rich languages.

Seddah, D., Kübler, S., and Tsarfaty, R. (2014). Introducing the SPMRL 2014 shared task on parsing morphologically-rich languages.

Seeker, W. and Kuhn, J. (2012). Making ellipses explicit in dependency conversion for a German treebank. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012)*, pages 3132–3139, Istanbul, Turkey. European Language Resources Association (ELRA).

Seeker, W. and Kuhn, J. (2014). An out-of-domain test suite for dependency parsing of German. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 4066–4073, Reykjavik, Iceland. European Language Resources Association (ELRA).

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4290–4297, PortoroÅ¿, Slovenia. European Language Resources Association (ELRA).

Telljohann, H., Hinrichs, E. W., Kübler, S., Zinsmeister, H., and Beck, K. (2015). Stylebook for the Tübingen treebank of written German (TüBa-D/Z). In *Seminar fur Sprachwissenschaft, Universität Tübingen, Tübingen, Germany*.

Verhoeven, B., Daelemans, W., and Plank, B. (2016). TwiSty: A multilingual twitter stylometry corpus for gender and personality profiling. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1632–1637, PortoroÅ¿, Slovenia. European Language Resources Association (ELRA).

Zeman, D., Mareček, D., Popel, M., Ramasamy, L., Štěpánek, J., Žabokrtský, Z., and Hajič, J. (2012). HamleDT: To Parse or Not to Parse? In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, pages 2735–2741, Istanbul, Turkey. European Language Resources Association (ELRA).

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Mis-

silä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, c., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.

9. Language Resource References

European Corpus Initiative. (1994). *European Corpus Initiative Multilingual Corpus 1 (ECI/MC1)*. Linguistic Data Consortium, ELRA, 1.0, ISLRN 511-168-567-582-5.

Institut für Deutsche Sprache. (n.d.). *Deutsches Referenzkorpus / Archiv der Korpora geschriebener Gegenwartssprache*. Institut für Deutsche Sprache.

Spanish HPSG Treebank based on the AnCora Corpus

Luis Chiruzzo, Dina Wonsever
Universidad de la República
Montevideo, Uruguay
{luischir, wonsever}@fing.edu.uy

Abstract

This paper describes a corpus of HPSG annotated trees for Spanish that contains morphosyntactic information, annotations for semantic roles, clitic pronouns and relative clauses. The corpus is based on the Spanish AnCora corpus, which contains trees for 17,000 sentences comprising half a million words, and it has CFG style annotations. The corpus is stored in two different formats: An XML dialect that is the direct serialization of the typed feature structure trees, and an HTML format that is suitable for visualizing the trees in a browser.

Keywords: HSPG, Spanish, AnCora, corpus

1. Introduction

We present the construction of a Spanish HPSG treebank based on the Spanish corpus AnCora (Taulé et al., 2008). This is part of an ongoing project to build a statistical Spanish HPSG parser.

An earlier version of this corpus is described in (Chiruzzo and Wonsever, 2016). In this previous work we used a series of hand-crafted rules to process all constituents in AnCora in order to find the heads and binarize all the phrases, identifying when the constituents acted as specifier, complement or modifier. In this first step, clitics and relative clauses had been identified but not properly handled.

In this work we finished the transformation of the corpus by incorporating an analysis for clitics and relative clauses into the feature structure and by annotating those constructions in the corpus, as well as producing a way of visualizing the HPSG trees.

2. Background

HPSG grammars are rich grammars (Pollard and Sag, 1994) that are able to represent both syntactic and semantic information in the same parse tree. The nodes in the trees and the rules to combine them are defined as typed feature structures (Carpenter, 1992) with a unification operation. The leaves of the parse tree are the words of a sentence, and the HPSG feature structure generally includes morphosyntactic information such as part of speech, agreement features, and syntactic valence features.

The English Resource Grammar (Flickinger, 1999) is an implementation of the HPSG principles for English, built into the LinGO Grammar Matrix (Bender et al., 2002), a framework for building unification grammars. There also exists a Spanish HPSG grammar built over the same principles: the Spanish Resource Grammar (SRG) (Marimon, 2010a). Both grammars are hand crafted paying particular attention to the linguistic details of the theory and the correctness of the modeled sentences. This means the trees obtained using these grammars are very rich, but on the other hand sentences that are not perfectly written (as is the case for general text extracted from the web) would not be even partially parsed. In this work we have the final aim of building a statistical parser from scratch based on the statistics of a properly annotated corpus, so we aim to improve the robustness of the parsing process for sentences that not necessarily are well written.

We base our approach in the work done for the parser Enju (Matsuzaki et al., 2007), which is a statistical HPSG parser for English created through the conversion of the Penn Treebank corpus (Marcus et al., 1993) into a HPSG suitable format (Miyao et al., 2005). The result is a fast high coverage parser for English that returns syntactic trees as well as argument structure information. We tried to follow a similar approach by transforming the Spanish corpus AnCora (Taulé et al., 2008) from its original context free like annotations to an HPSG compatible format.

The AnCora corpus contains about half a million words of news text in Spanish (there is also a version in Catalan of the same size), annotated in a context free grammar style enriched with attributes such as the grammatical function of constituents and the predicate-argument structure annotated in PropBank style (Kingsbury and Palmer, 2002).

There exists another HPSG corpus for Spanish based on a subset of AnCora that uses the Spanish Resource Grammar called the Tibidabo Treebank (Marimon, 2010b). In this corpus only sentences up to 40 words were considered, and the original AnCora annotations were not used. Instead, the sentences were parsed using the SRG and the appropriate analysis was selected from the set of resulting trees. A related corpus called the IULA Spanish LSP Treebank (Marimon et al., 2012) was originally annotated using a HPSG scheme, but the available version of the corpus contains dependency trees annotations instead of HPSG. This corpus contains around 40,000 sentences, generally shorter than the sentences in AnCora: 80% of the sentences have 20 words or less. In our case, we aimed to leverage the existing annotations of the AnCora corpus and try to use all its information to build our corpus, adding some missing information when necessary.

3. Description of the grammar

Our grammar is largely based on the one described in (Sag et al., 1999) for English, though some adaptations had to be done in order to port it to Spanish, and also because the information that could be extracted from AnCora sometimes was not enough to complete all aspects of the theory. We

include morphological and syntactic information into our grammar, but the main simplification is our treatment of semantics: In our version of the grammar, we include features for representing the argument structure of the verbs (annotated as PropBank style semantic roles) as semantic features. The original grammar uses a more complex approach to semantics based on Minimal Recursion Semantics (Copestake et al., 2005), but our approach is easier to extract from the information that is readily available in the AnCora corpus. Also, this approach could serve as a base for transforming the semantic representation into some other formats like Abstract Meaning Representation (Banarescu et al., 2012) which, in spite of not being the semantic perspective traditionally developed in HPSG, it still offers interesting insights for some semantic tasks (for example: text entailment and paraphrasing).
Figure 1 shows the lexical entry for a generic word, indicating all the possible features. A particular word might instantiate only the features that it defines, e.g. verbs in Spanish do not have a defined value for gender, so it would not be shown in the lexical entry (see section 3.3. for an example).

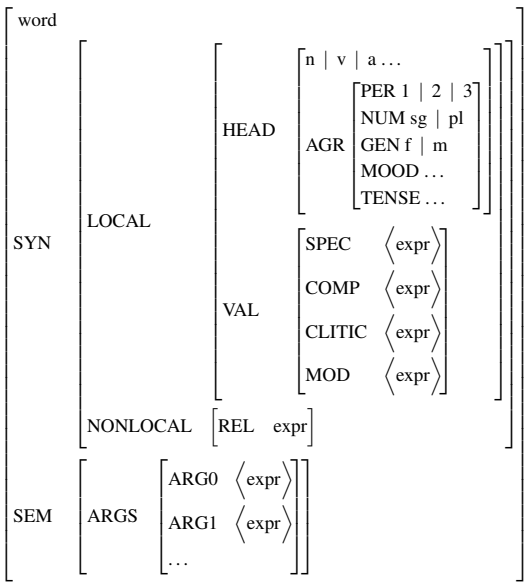


Figure 1: Feature structure for a lexical entry

3.1. Grammar rules

Generally HPSG grammars have very few rules and most of the combinatorial constraints are encoded in the features associated to each word or lexical entry. That is why HPSG grammars are generally said to be highly lexicalized grammars. Every rule must clearly mark which of its daughters is the head of the phrase, as the phrase will inherit the value for many of the head features. In our case, the grammar contains only thirteen rules:

- Two rules for applying a complement to the left or to the right of a head: `head_comp` and `comp_head`
- Two rules for applying a modifier to the left or to the right of a head: `head_mod` and `mod_head`
- Two rules for applying a specifier to the left or to the right of a head: `head_spec` and `spec_head`
- One special unary rule for representing the Spanish null subjects, which could be seen as a special case of an empty lexical entry: `empty_spr`
- Two rules for binarizing chains of coordinations: `coord_left` and `coord_right`
- One rule for applying a clitic to the left of a head: `clitic_head`
- One rule for joining a noun with a relative clause that modifies it: `head_rel`
- Two rules for applying a punctuation symbol to the left or to the right of a head: `head_punct` and `punct_head`

For example, consider the schematic definition of the specifier rule `spec_head` as shown in figure 2, that applies a specifier to the left of a head. This rule is used to analyze both a determinant as specifier of a noun phrase and a noun phrase as specifier of a sentence. On the right hand side there is an expression (the specifier) followed by another expression (the head) which requires a specifier. On the left hand side, the result is a phrase whose feature `HEAD` (which carries part of speech and agreement information) is coindexed with the same feature of the head expression. The `SPEC` feature of the head is coindexed with the specifier, but the value of that feature is removed from the resulting phrase on the left hand side.

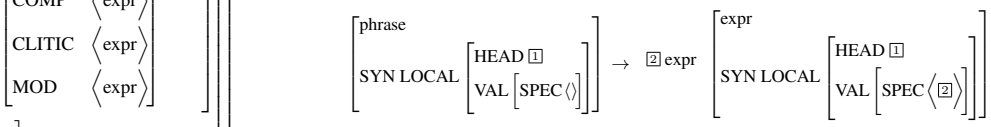


Figure 2: Schematic definition of the `spec_head` rule

Notice that the agreement principle defined in HPSG (Sag et al., 1999) is expressed in this rule by coindexing the `HEAD` features of the parent and the head daughter, and consequently the `AGR` features of both structures will also be coindexed.
Generally the head defines further constraints for the specifier. These constraints are not defined in the rule, but in the lexical entry corresponding to the head (see for example figure 3).
There is also another rule in our grammar that allows to combine a specifier with a head, the `head_spec` rule that takes a specifier on the right. Although Spanish is generally regarded as a SVO language, there exist many cases where it is more common to find the subject located after the verb. One example of this is the sentence *"llegó el tren"* / "the train arrived". We decided to create two different rules for allowing the subject to be applied both to the left and to the right of the verb, as well as two rules for applying the complement to the left or to the right. The existence of these rules allows many possible analysis for the sentences, so it is important that a parser takes into account the probability of applying the rules in each context.

3.2. General transformation of the corpus

The initial transformation of the AnCora corpus into a HPSG compatible format is described in (Chiruzzo and Wonsever, 2016). As AnCora contains rather flat context free grammar structures and a lot of variability between rules, we used a top down process to break all complex structures and separate them into simple units called *elementary trees* (head surrounded by arguments and modifiers). Then a bottom up process used a series of hand written rules to decide which of the elements inside a phrase was the head, and what rules to select in order to binarize the rest of the phrase. Special care had to be taken for the transformation of verb phrases, which include auxiliary and modal verb constructions, because their analysis in AnCora was different than the rest of the phrases (see section 3.5.). The process achieved an average 95.3 % precision for head detection (98.7 % without considerind coordinations) and 92.5 % average precision for argument detection.

However, this work left out the analysis of some interesting phenomena. Particularly the presence of clitics in the corpus was marked but not properly analyzed, as well as the use of relative clauses as modifiers of noun phrases. In this work we provide an appropriate analysis for these structures.

3.3. Modeling clitics in Spanish

Clitics are pronouns that can occur in different positions in verb phrases, for instance before a verb, and they could either take the place of an argument or coexist together with the argument in the phrase. Consider the sentence *"Juan le dará un regalo a María"* / "John will give a present to Mary". The lexical entry for the verb *"dará"* / "will give" as used in this sentence is shown in figure 3. Notice that in this case the clitic (*"le"*) corresponds to the indirect object (*"a María"*) which is also present in the sentence. This phenomenon is very common in Spanish and it is known as *clitic doubling*, presenting additional modeling complexity (Pineda and Meza, 2005). In our feature structure this is represented by setting both values as `ARG2` in the argument structure, which corresponds to the beneficiary semantic role. If either the clitic or the explicit argument are present, then the semantic argument will point to that expression, if both are present then the list associated to the semantic argument will contain both expressions.

3.4. Relative clauses as modifiers

In this work we focus on one kind of long distance dependency that is very common in Spanish: the use of a relative clause as a modifier of a noun, where the noun at the same time is acting as an argument of the verb in the clause. Generally the noun acts as the subject (e.g. *"el perro que me mordió"* / "the dog that bit me") or direct object (e.g. *"el libro que leí"* / "the book I read"), but it could act as any argument. Unlike English, in Spanish it is mandatory that these clauses are introduced by a subordinating relative expression that always contains a relative pronoun, such as *"que"* / "that" or *"a quien"* / "to whom".

Consider for example the sentence *"los cultivos que contienen almidón"* / "the crops that contain starch". The analysis according to our grammar is shown in figure 4. The

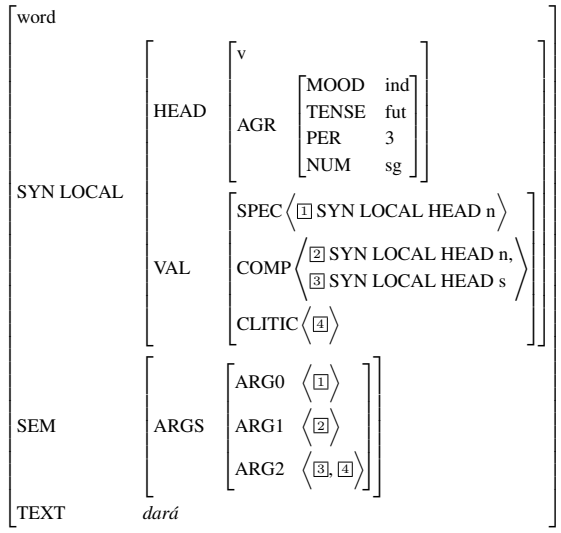


Figure 3: Feature structure for ditransitive verb *"dará"*, future indicative third person singular form of the verb *"to give"*

verb of the relative clause is transitive, but its corresponding subject (*"cultivos"*) is not readily available. Instead, the relative pronoun *"que"* takes the place of the subject, but keeps a non-local feature `REL` that points to the noun it stands for. The rule `head_rel` is used to unify a non-saturated `NONLOCAL.REL` feature to the appropriate expression it should be bound to, the resulting phrase cancels the value of the `NONLOCAL.REL` feature.

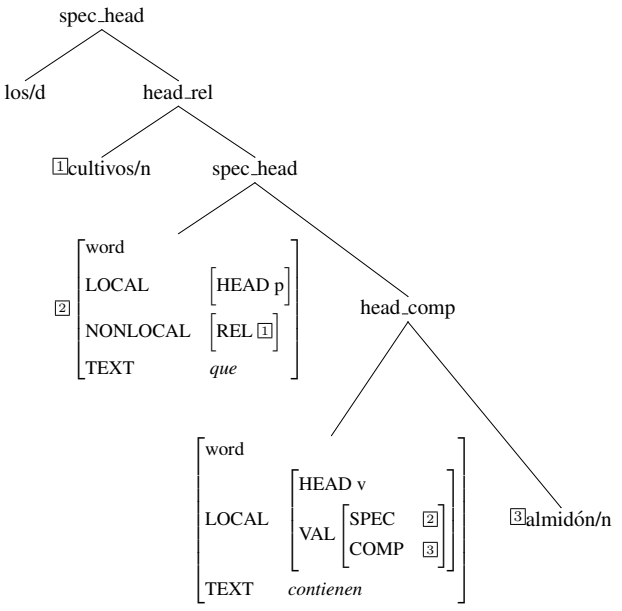


Figure 4: Simplified analysis for *"los cultivos que contienen almidón"* / "the crops that contain starch"

3.5. Verb phrases

Consider a sentence like *"ellos pueden hacer pasta"* / "they can make pasta", it contains the verbal periphrasis *"pueden hacer"* / "can make", plus a subject and a complement. A standard HPSG analysis for this phrase would first apply

the complement *"pasta"* to the verb *"hacer"*, then this verb would be applied as a complement to the verb *"pueden"* and finally the subject *"ellos"* would be applied to the resulting head. However, constructions of this type are analyzed differently in AnCora: the verbal periphrasis is considered a unit, so *"pueden hacer"* becomes a phrase that should expect a complement and a subject. We decided to keep this behavior from AnCora because it simplifies the analysis of displaced constituents.
The result of applying the complement *"hacer"* to the modal verb *"pueden"* is shown in figure 5. It is a phrase that combines the features of both daughters:

- It expects a noun phrase as specifier, which is coindexed with the specifier of both verbs.
- It expects the complement required by *hacer*.
- The agreement features are copied from the modal verb *puede*, because the subject must agree with the syntactic head.

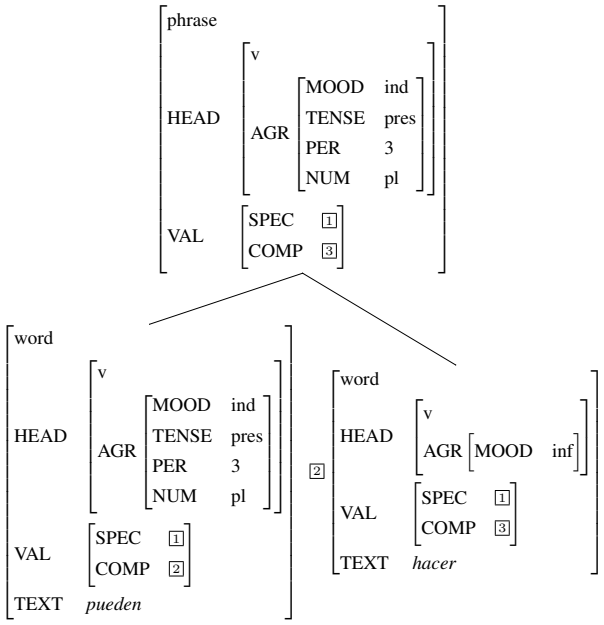


Figure 5: Simplified analysis for verb phrase *"pueden hacer"* / "can make"

In general verb phrases involving modal verbs or auxiliary verbs (e.g. *"haber visto"* / "have seen") in AnCora are analyzed like units and they are transformed in the same way in our corpus. Notice that in these cases the valence principle of HPSG is not applied in the same way, as the COMP feature of the non-head daughter (instead of the one from the head) is percolated to the mother. This is a variant of the standard HPSG complement rule that applies to a class of verbs such as modals and auxiliaries. One way of interpreting these structures is to consider that in these cases the syntactic head and the semantic head of the structure are different, and the main semantic content is provided by the semantic head. This can be extended to structures with more than two verbs, for example *"pueden querer traer"* / "might want to bring". In this structure we could consider that the main semantic content is in the verb *"traer"* / "to bring", so the arguments expected by this verb are percolated, through the successive application of the rules, to the base of the structure.

4. Composition of the corpus

The corpus has approximately half a million words in 17,000 sentences. The number of words for each part of speech is shown in table 1, while the number of times each rule is applied is shown in table 2.

POS	Instances	Unique
noun (n)	121089	22339
verb (v)	61688	11611
adjective (a)	35936	8228
adverb (r)	18951	1047
determinant (d)	76125	164
pronoun (p)	22690	183
preposition (s)	79897	342
interjection (i)	99	47
conjunction (c)	27062	125
date (w)	2731	986
number (z)	5362	3326
punctuation (f)	65538	28
Total	517168	48426

Table 1: Number of unique words and instances by part of speech

Rule	Instances
spec_head	111192
head_spec	6477
empty_spr	10776
head_comp	153473
comp_head	5777
head_mod	91913
mod_head	27873
head_punct	38709
punct_head	22848
coord_left	18226
coord_right	18226
clitic_head	8070
head_rel	7782

Table 2: Number of times each rule is applied in the corpus

The corpus is stored in two different formats[1]: XML and HTML. The XML format is a direct serialization of the typed feature structure trees into an XML dialect, we call this the TFSML format. Figure 6 shows an example of a lexical entry from the corpus in TFSML format. The HTML format is suitable for visualizing the trees in a browser. Currently we are in the process of converting the sentences into the format used by LKB (Copestake et al., 1999), a widely used grammar development environment.

5. Conclusion

We built a Spanish HPSG annotated corpus based on the AnCora Spanish corpus, containing the analysis of 17,000 sentences and half a million words. The parse trees contain syntactic and morphological information, semantic role

[1] A sample of the corpus can be found at:
www.fing.edu.uy/inco/grupos/pln/recursos/spanish_hpsg/index.html

```
<tfs id="v22119" text="analizado" type="word">
  <feature name="SYN">
    <feature name="LOCAL">
      <feature name="HEAD">
        <tfs type="v">
          <feature name="AGR">
            <feature name="GEN">
              <tfs type="m"/>
            </feature>
            <feature name="NUM">
              <tfs type="s"/>
            </feature>
            <feature name="MOOD">
              <tfs type="par"/>
            </feature>
          </feature>
        </tfs>
      </feature>
      <feature name="VAL">
        <feature name="SPEC" pointer="p8085"/>
        <feature name="COMP" pointer="sn54284"/>
      </feature>
    </feature>
  </feature>
  <feature name="SEM">
    <feature name="ARG0" pointer="p8085"/>
    <feature name="ARG1" pointer="sn54284"/>
  </feature>
</tfs>
```

Figure 6: Representation of the past participle *"analizado"* / "analyzed" in TFSML format.

labels annotated in PropBank style both for explicit arguments and clitic pronouns, and the analysis of relative clauses that act as modifiers.

This is part of an ongoing project for building a statistical HPSG parser for Spanish. Some work has already been done in this direction, for example we carried some baseline parsing experiments using the data of this corpus, and we also trained a supertagger over an earlier version of the corpus that is able to classify complex verbs, nouns and adjectives using the categories of our corpus (Chiruzzo and Wonsever, 2015). The next step in this process will be to train a full deep parser for Spanish using this corpus.

6. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2012). Abstract meaning representation (amr) 1.0 specification. In *Parsing on Freebase from Question-Answer Pairs. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Seattle: ACL*, pages 1533–1544.

Bender, E. M., Flickinger, D., and Oepen, S. (2002). The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In *Proceedings of the 2002 workshop on Grammar engineering and evaluation-Volume 15*, pages 1–7. Association for Computational Linguistics.

Carpenter, B. (1992). *The logic of typed feature structures*. Cambridge University Press.

Chiruzzo, L. and Wonsever, D. (2015). Supertagging for a statistical hpsg parser for spanish. In *International Conference on Statistical Language and Speech Processing*, pages 18–26. Springer.

Chiruzzo, L. and Wonsever, D. (2016). Transforming the ancora corpus to hpsg. In *HeadLex16, Joint 2016 Conference on Head-driven Phrase Structure Grammar and Lexical Functional Grammar*.

Copestake, A., Carroll, J., Malouf, R., and Oepen, S. (1999). The (new) lkb system. *Center for the Study of Language and Information, Stanford University*.

Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. (2005). Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Miyao, Y., Ninomiya, T., and Tsujii, J. (2005). Corpus-oriented grammar development for acquiring a head-driven phrase structure grammar from the penn treebank. In *Natural Language Processing–IJCNLP 2004*, pages 684–693. Springer.

Pineda, L. and Meza, I. (2005). The spanish pronominal clitic system. *Procesamiento del lenguaje natural*, 34:67–103.

Pollard, C. and Sag, I. A. (1994). *Head-driven phrase structure grammar*. University of Chicago Press.

Sag, I. A., Wasow, T., Bender, E. M., and Sag, I. A. (1999). *Syntactic theory: A formal introduction*, volume 92. Center for the Study of Language and Information Stanford, CA.

7. Language Resource References

Flickinger, D. (1999). The english resource grammar.

Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *LREC*. Citeseer.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Marimon, M., Fisas, B., Bel, N., Vivaldi, J., Torner, S., Lorente, M., Vázquez, S., and Villegas, M. (2012). The iula treebank. In *Lrec*, pages 1920–1926.

Marimon, M. (2010a). The spanish resource grammar. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC*, pages 17–23, Valletta, Malta.

Marimon, M. (2010b). The tibidabo treebank. *Procesamiento del lenguaje natural, 2010, vol. 45, num. 1, p. 113-119*.

Matsuzaki, T., Miyao, Y., and Tsujii, J. (2007). Efficient hpsg parsing with supertagging and cfg-filtering. In *IJCAI*, pages 1671–1676.

Taulé, M., Martí, M. A., and Recasens, M. (2008). Ancora: Multilevel annotated corpora for catalan and spanish. In *LREC*.

Universal Dependencies for Amharic

Binyam Ephrem Seyoum[1], Yusuke Miyao[2], Baye Yimam Mekonnen[3]
Addis Ababa University[1,3], National Institute of Informatics[2]
P.O.Box 1176, Addis Ababa[1,3], 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430[2]
binyam.ephrem@aau.edu.et[1], yusuke@nii.ac.jp[2], baye.yimam@aau.edu.et[3]

Abstract

In this paper, we describe the process of creating an Amharic Dependency Treebank, which is the first attempt to introduce Universal Dependencies (UD) into Amharic. Amharic is a morphologically-rich and less-resourced language within the Semitic language family. In Amharic, an orthographic word may be bundled with information other than morphology. There are some clitics attached to major lexical categories with grammatical functions. We first explain the segmentation of clitics, which is problematic to retrieve from the orthographic word due to morpheme co-occurrence restriction, assimilation and ambiguity of the clitics. Then, we describe the annotation processes for POS tagging, morphological information and dependency relations. Based on this, we have created a Treebank of 1,096 sentences.

Keywords: Treebank, Universal Dependencies, Amharic

1. Introduction

In recent years, different language processing applications demand state-of-the-art parsers. Question answering, machine translation, information summarization and similar applications require high-quality parsers. In order to train or develop an efficient parser, it has become an established practice to create a Treebank, linguistically annotated corpus which includes, in most cases, morphological and syntactic annotations. Treebanks play an important role to the research in parsing natural languages. They can also be used in testing linguistic theories and scrutinize corpus-based language analysis. Furthermore, treebanks are essential resources for building and testing data-driven tools such as POS taggers and morphological analyzers where they serve as a gold standard for these tools.

Treebanks have been developed for well-resourced languages in different frameworks such as Phrase Structure, HPSG, and Dependency. However, there are no treebanks for Amharic in any form. In this study, an attempt will be done to create treebanks for Amharic. Apart from developing this resource, the research contributes to the general problem of parsing Morphologically-rich Languages (MRL). In such languages, a dependency relation exists not only between the orthographic words (space-delimited tokens) but also relations within a word itself (Goldberg, Elhadad, and Gurion, 2009). Because of this, clitics attached to orthographic words need to be segmented for proper syntactic analysis. However, automatic segmentation of the prefix and the suffix clitics from the orthographic word in Amharic is problematic due to morpheme co-occurrence restriction, assimilation and ambiguity of the clitics (cf. Section 3 and 4). In this paper, first we discuss clitic segmentation then we describe the creation of the treebanks which are annotated for POS tag, morphological information and dependency relation.

2. Background

Universal Dependencies (UD) project is a collaborative effort to ensure consistent annotations across many languages. This project has benefited from earlier efforts including universal annotation of Google Universal part-of-speech tags (Petrov, Das, and Mcdonald, 2012), morphosyntactic features (Zeman, 2008; Zeman et al., 2012) and Stanford Dependencies (de Marneffe et al., 2014; de Marneffe and Manning, 2008). The objective of UD, as stated in Nivre (2015) is to encourage multilingual parser improvement, cross-lingual learning, and parsing research from a language typology point of view. Even if UD proposes consistent ways of annotations across languages, it does not compromise the unique features of each language. The framework allows language-specific features to be included in annotations. In this paper, we discuss the language-specific features for Amharic.

UD (v2.0) was released on March 01, 2017, with 70 treebanks representing 50 languages (Nivre et al., 2017). All treebanks were annotated with POS tags, morphological features and syntactic relations. Most of them were automatic conversions from one version of treebanks to UD treebanks with manual corrections at some level. The number of sentences were ranging from 600 to 90,000. It also includes some low-resourced languages with a small number of sentences. This demonstrated how low-resourced languages could be benefited from the experience of other languages and contributed to the wider research community. This is also true for Amharic as well. The project encourages more languages to come into the picture.

3. Issues in Amharic Word Segmentation

An orthographic word in Amharic, though, it is delimited by white space, leaves boundaries of lexical or syntactic units unclear. This is because it combines some syntactic words into one compact string of letters. A given orthographic word may attach one or more function words and inflectional morphemes beside the root form. As in Arabic and Hebrew, function words such as prepositions, conjunctions and articles are attached to other content words. This makes an orthographic word in such language function as a phrase, a clause or a sentence. Currently, it has become a trend in Semitic languages to separate function words or clitics as tokens for further linguistic

analysis. For example, አልሰጠሁትም። /ʔalɨsət't'əhutɨmm/ “I did not give (it) (to) him.” is written as an orthographic word but it is a full-fledged sentence. This orthographic word encompasses syntactic elements with four parts-of-speech; particle, verb, and two pronominal suffixes. It also expresses three syntactic functions: predicate, subject and direct object.

A syntactic analysis in UD is based on the lexicalist view which says grammatical relations are expressed among syntactic words. It is indicated that practical computational models gain from this approach (de Marneffe et al., 2014). Following this, UD suggests segmentation of function words from content words (Nivre et al., 2016). For example, the above Amharic orthographic word, አልሰጠሁትም። could be segmented into five syntactic forms: አል, ሰጠ, ሁ, ት, ም. However, clitic segmentation is not an easy task in Amharic.

Amharic writing system is said to be ‘syllabic’. Most clitics are vowel forms or at least they begin with a vowel. Since Amharic phonology constrains sequences of two vowels, most clitics undergo phonological changes. The change is also exhibited in the written form where clitics are attached to their host. For proper segmentation, then, we need to recover the hidden form before we segment it. For example, the word ባንድ /band/ "in one”, can be segmented into the preposition በ /bə/ “in” and the numeral አንድ /ʔand/ “one”. However, if we simply segment the first character “ባ” /ba/, the remaining form, ንድ /nd/ will not have meaning.

In addition, the written form in Amharic might lose some grammatical morphemes due to morpheme co-occurrence restriction. For instance, there are some verbs like ተገኘ /təgəɲɲə/ “be found” which can give a sense of passive (which is marked by ተ- /tə-/). When the passive form is used in jussive constructions, ይገኝ /jɨgəɲɲ/ “let it be found”, the passive marker ተ- /tə-/ gets assimilated to the stem initial consonant. Further, the jussive form can serve as input for the imperfective form እሚገኝ /ʔɨmmigəɲɲ/ “that which will be found”. Note that in such imperfective forms, the passive marker ተ- /tə-/ assimilates to the initial consonant of the stem form and the subject marker ይ- /jɨ/ of the jussive form assimilates to the imperfective marker final consonant. The same is true in the case of relative clause, የሚገኝ /jəmmigəɲɲ/ “that which can be found”, where the passive marker ተ- /tə-/ and the subject marker ይ- /jɨ-/ are assimilated and the imperfective marker እም- /ʔɨmm-/ is reduced to -ም /-mm/ only. The process of assimilation and reduction of forms make segmentation of orthographic forms difficult.

Furthermore, some clitic forms can be part of the word without being segmented. In such cases, clitics need context for segmentation; otherwise, they are ambiguous. For example, ከሱ /kəssu/, can mean ‘from him’ or ‘he(honorific)/they lost weight’. It can be segmented into the preposition ከ /kə/ and the pronoun እሱ /ʔɨssu/ for the former meaning but not segmented for the latter meaning.

Segmentation of some clitics may cause other affixes or morphological elements to be separated as well. For instance, we consider the definite marker as a clitic. Unlike Arabic and Hebrew, the definite and the case marker in Amharic are suffixes. When a definite noun appears in an object position, it is marked for the accusative case and the marker follows the definite marker. Thus, segmenting the definite marker has an effect on the status of the case marker that behaves as a clitic. In both Arabic and Hebrew, case markers are treated as morphological features whereas, in Amharic, they are independent syntactic elements. Thus, we have ‘case’ relations rather than morphological features.

When a noun, in Amharic, is modified by an adjective or by other modifiers, the definite marker is attached to one of the modifiers only. In Arabic and Hebrew, such instance is treated as agreement phenomena within the noun phrase. However, in Amharic noun phrase, the definite marker is attached to one of the non-head elements. It could be considered as a phrasal element which can be added to the entire phrase. In our analysis, we treat definiteness at a syntactic level or dependency relation between the noun and the definite marker. The following examples demonstrate our points.

1. መጽሐፉን ሰጠው።
   məs'haf-**u-n** sət't'-ə-w
   book-DEF-ACC give.PRF.-3SGM-3SGM
   “He gave him the book.”
2. ትልቁን መጽሐፍ ሰጠው።
   tɨlk'-**u-n** məs'haf sət't'-ə-w
   big-DEF-ACC book give.PRF.-3SGM-3SGM
   “He gave him the big book.”
3. ጥቁሩን ትልቅ መጽሐፍ ሰጠው።
   t'ɨk'ur-**u-n** tɨlk' məs'haf sət't'-ə-w
   black-DEF-ACC big book give.PRF.-3SGM-3SGM
   “He gave him the big black book.”

In the above examples, the definite marker (-u) and the case marker (-n) are attached to the head noun in (1), but to the adjective in (2) and (3). When the noun phrase expands both markers are attached to the left most element. The noun phrases in (2) and (3) get their definite features from other elements within the phrase. That is why we consider these features as phrasal elements. However, in the segmentation task, since both definite and case markers co-occur, we segment them separately.

Morphemes to be considered as clitics are listed in Binyam, Miyao, and Baye (2016). Following this, we developed a manually segmented data of 2, 300 sentences or 50,520 tokens out of which we selected only 1000 sentences, 12, 039 tokens for the manual annotation of POS tagging, morphological information, and dependency relations.

4. Parts of speech annotation

There have been some works on POS tagging in Amharic (Gamback B., 2012; Martha, Solomon, and Besacier, 2011; Binyam, 2010; Gambäck, Olsson, Argaw, and Asker, 2009; Sisay, 2005). However, the work of Demeke and Getachew (2006), known as the Walta Information Center corpus (WIC), has received much attention among Amharic NLP researchers and has been used for different applications. They propose a 31 tag-set for the manual annotation of a news corpus of 210,000 tokens. The tag-set is based on orthographic words. As a result, they propose a compound tag-set for those words which attach preposi-

tion and/or conjunctions. Since these elements are attached to different lexical categories like nouns, verbs, adjectives, etc, the number of tag-sets has increased. This in return has an effect on the efficiency of automatic taggers trained on the corpus, developed following the proposed tag-set. A recent work by Rychlý and Suchomel (2016) reports an average accuracy of 87.4% of a TreeTagger that is trained and evaluated on WIC.

Besides expected inconsistencies in WIC, which is a manual annotation, such a tag-set has an impact on the performance of an automatic tagger. One impact is, though, they claim to do the task of POS tag, it is beyond the scope of POS tagging. They are trying to give tag-sets for various syntactic constructions, (phrases, clauses and sentences) in addition to a syntactic word. On the other hand, Amharic is a less-resourced and morphologically-rich language where problems of OOV and ambiguities are major bottlenecks. Considering orthographic words for tagging task makes the problems more complex. This is because we are trying to learn several syntactic constructions represented in the orthographic words from a limited corpus.

The other impact is that we miss some information or become confused as the orthography leads to loss of some syntactic information. For instance, in WIC corpus, a separate tag is proposed for relative verbs (VREL). When verbs attach a preposition they are tagged as VP (which means a verb with a preposition). However, when relative verbs attach a preposition, for instance, the relative marker gets deleted due to morpheme co-occurrence restrictions in the language. It is confusing for annotators which tag to use from the orthographic information in such cases. We noted inconsistencies in the tagging of such words in WIC. Some annotators consider the internal structure of a word and tagged them as VREL even if there is a preposition, while others use VP, which contradicts with other similar VP tagged structures. Furthermore, such constructions are also tagged as adjectives (ADJ), considering their modification function in a noun phrase.

In WIC tag sets, it is only the preposition and conjunction that are identified as elements that can be attached to other lexical categories. According to the guideline these elements are attached to nouns, verbs, pronouns, adjective and numerals. However, some adverbs (for instance, ዛሬ /zare/ ‘today’) can attach a preposition and/or conjunction. In addition, the guideline suggests some lexical categories to have sub-classes. Specifically, nouns (verbal noun - VN), verbs (auxiliary - AUX, relative verb - VREL) and numerals (cardinal – NUMCR and ordinal - NUMOR) which have sub-categories with the respective specific tags. However, when these sub-categories attach a preposition or a conjunction, their distinction from the other respective categories cannot be distinguished. This is because the compound tag-sets are used for all categories. For instance, the guideline suggests that a VP tag is used for any verb including auxiliary and relative verbs attaching a preposition. Thus, an auxiliary, other verbs, and relative verbs with a preposition have similar tags as VP. Consequently, an expression tagged as VP following their tag-sets, will have different syntactic structures, i.e. it can be an auxiliary with a preposition or it is a verb or a relative verb with a preposition but tagged similarly. Therefore, the distinction they want to capture by the tags of the sub-categories will not be used when such forms attach a preposition.

The above mentioned problems occur due to the fact that a word is defined as any form that is delimited by a white space. We suggest that for languages like Amharic, clitics should be segmented before tagging and the units for tagging should be syntactic words rather than orthographic words.

When adopting UD, we need to give language- specific information regarding the POS tag-set relevant to Amharic. We need also to provide specific tag-set for some clitics which may as well appear independently. For instance, prepositions and conjunctions can be written separately. For such clitics, we may use the existing tag-sets. However, there are some clitics that need a new tag-set which are result of clitic segmentation.

UD POS	**Amharic tag-set**	**examples**
ADJ	ADJ	ትልቅ “big ”
ADP	ADP	ከ “from”
ADV	ADV	በጣም “very”
AUX	AUX	ኣል “verb to be”
CCONJ	CCONJ	ግን “but”
DET	DET	ይህ “this”
INJ	INJ	ሆ “oh”
NOUN	NOUN	በግ “sheep”
PART	ACC	ን “accusative case”
	NEG	አለ_ሴት “without a woman”
	RLP	የ_መጣ "who came”
	IRLP	አም_ይ_መጣ "who will come”
	NCM	አል_መጣ_ም "He didn’t come”
PRON	PRON	አንተ "you”
	OBJC	ነገር_ኩ_ኣት "I told her”
	SUBJC	ሄድ_ኧ “he went”
	POSM	ቤት_ኡ“my house”
PROPN	PROPN	ካሳ “Kassa”
PUNCT	PUNCT	። "period/fulstop”
SCONJ	SCONJ	ስለ "because”
SYM	SYM	€፣£፣$
VERB	VERB	በላ “eat”
X	X	other

Table1: UD POS tag and Amharic-Specific tag-sets

As can be noted from Table 1, we expand both the particles and the pronouns to handle some clitics that may not have proper tagging after segmentation. Tagging these clitics separately has two advantages. First, segmentation reduces word forms. Due to the morphological structure of the language, word-forms in Amharic are very large. The word-forms even increase with different clitics. Second, it helps to represent syntactic relations between clitics and their host. There is syntactic relation for instance between a preposition and a noun. In the above table, we indicate the mapping between UD tag and Amharic-Specific tag. It is possible to convert Amharic-Specific tags into corresponding UD tags.

5. Morphological annotation

The UD annotation schema defines a set of 21 morphological features across languages. These include Case, Person, Number, Voice and Mood. However, in contrast to the POS tag, the language specification allows treebanks to introduce morphological features that are not included in this universal inventory. This suggests that morphological features can be drawn from the extended compilation of morphological features of other languages (Zeman, 2008).

As we have shown in Section 3 above, due to clitic segmentation some morphological features like the case and the agreement markers are treated as separate forms. Following this decision, case and person features are handled at the syntactic level. Table 2 summarizes the morphological features used in Amharic treebank annotation.

Category	Features	Tag	Description
Nominal	Gender	Mas	Masculine
		Fem	Feminine
		Com	Common gender
	Number	Sing	Singular
		Plur	Plural
		Coll	Collective
Verbal	Verb Form	Conv	Converb
		Inf	Infinite
		Vnoun	Verbal noun
	voice	Pass	Passive
		Mid	Middle
		Rcp	Reciprocal
		Cas	Causative
	Tense	NPas	Future/Present
		Past	Past
	Aspect	Imp	Imperfect
		Perf	Perfect
		Prog	Progressive
		Presp	Prospective
	Polarity	Neg	Negative
		Pos	Affirmative

Table 2: Morphological Features

6. Syntactic annotation

Syntactic dependency types for Amharic are defined in order to be as consistent as possible with the principle of UD. In table 3 below, we provide some samples of typical dependency relations in Amharic. However, the dependency relations for Amharic needs some language-specific information.

One language-specific can be the relation between the subject and/or object clitics and the verb. UD requires the use of the expletive (*expl)* relation for cases of true clitic doubling. In Amharic, the lexical nominal and the clitic may appear in a clause or in a sentence. The nominal will be given the grammatical role of *nsubj*, *obj*, etc., while the clitics will be treated as a pronominal copy of the nominal and will get the role of *expl*. However, when the nominal is dropped, the clitic will get the grammatical roles of *nsubj* or *obj*. Such analysis helps us to handle the case of pro-drop in Amharic. For example, the expression "አለቀሰ" and "እሱ አለቀሰ" are equivalent and can mean "He cried." The structural difference can be captured using an *expl* relation as indicated in Figure 1.

Relation	Construction in Amharic	→direction
nsubj	ልጁ መጣ።	nsubj(መጣ, ልጁ)
obl	ለልጁ ሰጠው።	obl(ሰጥ, ልጅ)
iobj	ደብተሩን ለልጁ ሰጠው።	iobj(ሰጥ,ደብተር)
csubj	የተናግረችው ትርጉም ይሰጣል።	csubj (ይሰጣል, ተናገረችው)
nmod	የእኛ ሀገር	nmod(ሀገር,እኛ)
amod	ትልቁ ልጅ	amod(ልጅ,ትልቅ)
admod	የት መሄድ ትፈልጋለህ ?	advmod (መሄድ, የት)
mark	የመጣው ልጅ	mark(መጣ, የ)
aux	ልጁ ሄዷል።	aux(ሄድ, ኣል)
cop	ልጁ ጎበዝ ነው።	cop(ጎበዝ,ነው)
det	ልጁ (ልጅ_ኡ)	det(ልጅ, ኡ)
acl	የመጣው ልጅ	acl(ልጅ, መጣ)
advcl	ካወቃችሁ ለእስተማሪው ንገሩት።	advcl(እወቅ,ንገር)
ccomp	ጫማውን ልጠግነው እችላለሁ ብሎ ነበር።	ccomp (ብል, ልጠግነው)
expl	ካሳ ለእልማዝ ነገር_ኧ_ኣት	expl(ነገር, ኧ)

Table 3: Some of the dependencies for Amharic

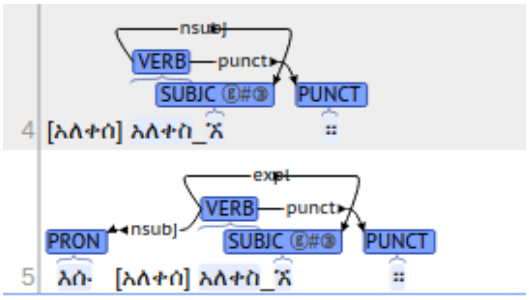


Figure 1: Amharic pro-drop

Another issue is the treatment of converbs. Amharic converbs have features like verbal form, adverbial, non- finite and subordinate. In addition, they modify the verb phrase and uniformly lack specification for most verbal grammatical features like tense, aspect, etc. Thus, we consider them as non-main verbs and the final verb as a main verb. Functionally, converbs may have three functions: serial, consecutive, and co-extensive (Meyer, 2011). They are serial, when they express a chain of actions that constitute one activity and that is concluded by the final verb. They are consecutive when the con-verb expresses an action that takes place earlier than the following verb. They are co-extensive when the action of the con-verb (stative) occurs simultaneously or when they make up one verbal meaning (Desalegn, 2016). We suggest an adverbial modifier (*advmod*) to be used in relation to co-extensive functions. In a structure of subordination, that means both serial and consecutive, we propose to use a sub-relation of

compounds, *compound:svc* (a compound with serial verb construction). Figure 2 and 3 demonstrate this point.

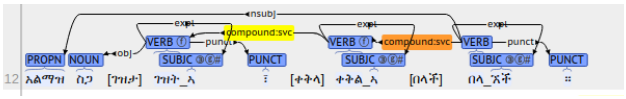


Figure 2: Con-verb with serial construction

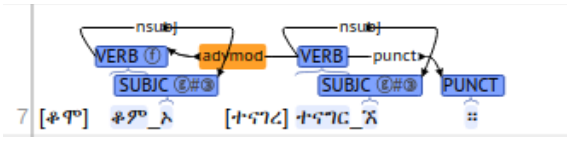


Figure 3: Con-verb with coexistive function

A further language-specific information which is important for Amharic is the treatment of light-verbs in Amharic. Light verb forms do not have a category by themselves. They get their category from their second member. For instance: in construction *sɨbbɨr + allə* – "broken", it is a compound verb as the second member is a verb, where as in *sɨbbɨr + at* - "brokenness", it is a compound noun. The light verb construction is constructed from a light verb and the existential verb. Since, the light verb is semantically null, it cannot be the head. Thus, in such a case, we have decided the copula to be treated as a main verb and the head of the phrase. The relation between the light verb and the main verb is labeled as a compound. Figure 4 shows how light verbs are treated in our UD.

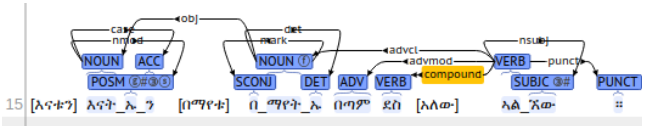


Figure 4: Light-verbs

Copulas in UD are treated as dependent of a lexical predicate (de Marneffe et al., 2014). In Amharic they are used to carry TAM information. However, because we decided to segment subject clitics, the clitics will have syntactic relation with the main verb. The treatment of copula construction is demonstrated in figure 5.

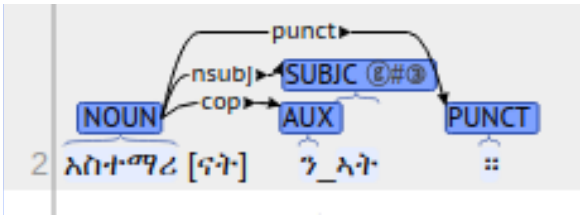


Figure 5: Copula constructions

We have noted that the segmentation of clitics helps us to handle some syntactic relations within a phrase. As we discussed above (cf. Sections 3), there are some clitics that can be attached only once to any one of the constituents within a noun phrase. The phrase gets its features from those clitics attached to the non-head elements. However, in our analysis, the syntactic relations hold between the head and the clitics. The examples (1) to (3) in Sections 3 can be annotated as depicted in figure 6.

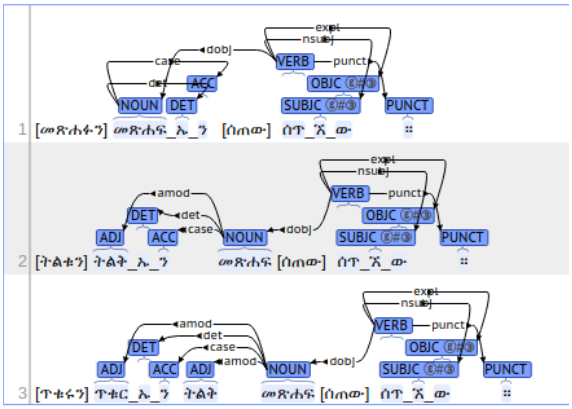


Figure 6: Definite and case marking within NP

7. Corpus and annotation consistencies

In most works of Amharic corpora, data are collected from electronic media, especially from the news media. However, such sources are produced without proper text editing tools like a spell or grammar checker. As a result, errors occur every now and then that require manual editing.

In addition, the Amharic writing system is not standardized. Some people tend to write in phonemic form (the abstract form or what one intends to say) and some tend to write in the phonetic form (what is actually uttered). As a result, there are different forms for a given word in the corpus collected from most electronic media. This makes it difficult to develop a language model. Thus, we focus on collecting sentences from grammar books as they try to cover a variety of grammatical constructions and are consistent with the way the language is written. We have also included other sources like fictions, biographies, religious texts and news.

After sentences were collected from such sources, they were manually corrected for spelling errors. Before the annotation, however, words with clitics were manually segmented. We measured the annotation agreement in the segmentation task. This task has two components. The first task is identifying those words which bear clitics or considered to be complex words. The second task is proper segmentation of the clitics. We measured the agreement using the Kappa measure for both components of the segmentation task. For the first task, we have calculated the number of times that both annotators agree to consider a word as complex, the number of times both agree to exclude, the number of times only annotator one wanted to include a word as complex, and the number of times only annotator two wanted to include a word as complex. Based on this we got the Kappa value of 0.862, which is interpreted as almost perfect agreement.

For the second task in segmentation, we considered those words identified as complex by both annotators. We calculated the number of times that both provide identical segmentation, the number of times that only the first annotator adds more segmentation and the number of times that only the second annotator adds more segmentation. Based on this, we got a Kappa of 0.585

which is considered to be a moderate agreement. Although the agreement was not bad, we asked other linguists to validate their segmentation after the annotators segmented the data, as their input was very important for the further process. Based on the linguists recommendation, we corrected the segmentation data to make it ready for the annotation process.

In the annotation stage, words were annotated for POS, morphological information and syntactic relations which were done manually. Two annotators were trained based on the guideline developed for this purpose. After a series of trainings and updating the guideline, we measured the annotation consistencies for POS tagging and dependency relations using a sample sentence. In order to calculate the Kappa measure for POS agreement, we used a confusion matrix for each tag used in the manual annotation. Accordingly, we got a kappa measure of 0.622, which means there is a substantial agreement between the annotators.

We also measure the annotation consistency for dependency relation. In doing so, we developed a confusion matrix for each dependency relation we used in the annotation. According to the Kappa measure we got, 0.488, there is a moderate agreement between the manual annotators. In order to increase the reliability of the corpus, we have also verified the annotations with two linguists' after the manual annotations were done.

The UD corpus is composed of 1,096 sentences and it contains 8,025 tokens, clitics are not counted as tokens. The data will be released in the upcoming UD version, v2.2[1].

8. Conclusion

We have presented the process of creating Amharic treebanks following the UD annotation scheme. Adopting UD to Amharic needs some kind of decisions regarding the tokens or syntactic words. We have mentioned problems related to clitic segmentation and indicated that Amharic orthographic words may not only bear morphological information but also carry other function elements of syntactic relations. Due to morpheme co-occurrence restrictions, phonological assimilations, and ambiguities, it is difficult to recover syntactic elements from orthographic words. Thus, we suggest that MRL like Amharic segmentation or tokenization of the orthographic word should be the fist step for proper syntactic analysis. For future work, we have a plan to increase the size of segmentation data so that we can develop a machine learning model. In addition, we have a plan to expand the size of the treebank.

Acknowledgements

This project is partially funded by NORHED project called Linguistic Capacity Building-Tools for inclusive development of Ethiopia[2]. We would like thank the University of Oslo for its financial support. We also would like to extend our sincere gratitude to the anonymous reviewers for their valuable feedback which greatly improved this article.

[1]http://universaldependencies.org/

[2]http://www.hf.uio.no/iln/english/research/projects/linguistic-capacity-building-tools-for-the-inclu/

9. Bibliographical References

Binyam, E., Miyao, Y., and Baye, Y. (2016). Morpho-syntactically Annotated Amharic Treebank. In *Proceedings of Corpus Linguistics Fest (CLiF 2016), June 6-10, 2016* (pp. 48–57).

Binyam, G. G. (2010). *Part of Speech Tagging for Amharic*. University of Wolverhampton.

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal Stanford Dependencies: A cross-linguistic typology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC14)* (pp. 4585–4592).

de Marneffe, M.-C., and Manning, C. D. (2008). Stanford typed dependencies manual. In *Proceedings of COLING 2008 Workshop on Cross-framework and Cross-domain Parser Evaluation*.

Demeke, G. a., and Getachew, M. (2006). *Manual annotation of Amharic news items with part-of-speech tags and its challenges*. *Ethiopian Languages Research Center Working Papers* (Vol. 2).

Desalegn, A. (2016). The Inceptive Construction and Associated Topics in Amharic and Related Languages. Stockholm University.

Gambäck, B., Olsson, F., Argaw, A. A., and Asker, L. (2009). Methods for Amharic part-of-speech tagging. *Proceedings of the First Workshop on Language Technologies for African Languages*.

Gamback B. (2012). Tagging and Verifying an Amharic News Corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation.* (pp. 79–84). European Language Resources Association.

Goldberg, Y., Elhadad, M., and Gurion, B. (2009). Hebrew Dependency Parsing: Initial Results. In *Proceedings of the 11th IWPT09* (pp. 129–133). Paris. Retrieved from http://www.aclweb.org/anthology/W09-3819

Martha, Y. T., Solomon, T. A., and Besacier, L. (2011). Part-of-Speech Tagging for Under-Resourced and Morphologically Rich Languages – The Case of Amharic. In *Conference on Human Language Technology for Development* (pp. 50–55). Alexandria, Egypt.

Meyer, R. (2011). The Converb in Amharic. *Journal of Semitic Studies*, (1980), 165–192.

Nivre, J. (2015). Towards a Universal Grammar for Natural Language Processing. In A. Gelbukh (Ed.), *Proceeding of the 16th International Conference of Computational Linguistics and Intelligent Text Processing, CICLing 2015* (Vol. 9041, pp. 3–16). Springer International Publishing Switzerland.

Nivre, J., Agić, Ž., Ahrenberg, L., Aranzabe, M. J., Asahara, M., Atutxa, A., … Zhu, H. (2017). Universal Dependencies 2.0. *Http://universaldependencies.org/*.

Nivre, J., de Marneffe, M.-C., Ginter, F., Bullet, M., Ginter, F., Goldberg, Y., … Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (pp. 1659–1666).

Petrov, S., Das, D., and Mcdonald, R. (2012). A Universal Part-of-Speech Tagset. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2012* (pp. 2089–2096).

Rychlý, P., and Suchomel, V. (2016). Annotated Amharic Corpora. In *19th International Conference on Text, Speech and Dialogue (TSD 2016)* (Vol. LNAI 9924, pp. 295–302). Brno, Czech Republic: Springer International Publishing Switzerland.

Sisay, F. A. (2005). Part of Speech tagging for Amharic using Conditional Random Fields. In *Proceedings of the ACL workshop on computational approaches to semitic languages* (pp. 47–54). Association for Computational Linguistics.

Zeman, D. (2008). Reusable Tagset Conversion Using Tagset Drivers. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)* (pp. 213–218).

Zeman, D., Mareček, D., Popel, M., Ramasamy, L., Štěpánek, J., Žabokrtský, Z., and Hajič, J. (2012). HamleDT: To Parse or Not to Parse? In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)* (pp. 2735–2741).

A Parser for LTAG and Frame Semantics

David Arps, Simon Petitjean
Heinrich-Heine-Universität Düsseldorf
david.arps@uni-duesseldorf.de, petitjean@phil.uni-duesseldorf.de

Abstract

Since the idea of combining Lexicalized Tree Adjoining Grammars (LTAG) and frame semantics was proposed (Kallmeyer and Osswald, 2013), a set of resources of this type has been created. These grammars are composed of pairs of elementary trees and frames, where syntactic and semantic arguments are linked using unification variables. This allows to build semantic representations when parsing, by composing the frames according to the combination of the elementary trees. However, the lack of a parser using such grammars makes it complicated to check whether these resources are correct implementations of the theory or not. The development of larger resources, that is to say large-coverage grammars, is also conditioned by the existence of such a parser. In this paper, we present our solution to this problem, namely an extension of the TuLiPA parser with frame semantics. We also present the frameworks used to build the resources used by the parser: the theoretical framework, composed of LTAG and frame semantics, and the software framework, XMG2.

Keywords: Parsing, Grammar, Syntax, Semantics, Tools, Systems, Applications.

1. Introduction

The development of linguistic resources such as precision grammars and lexicons is a complex and time consuming task. Even though the challenges offered by these tasks often come from the size of the resources to develop, the difficulty can be increased by the lack of tools processing this type of data and allowing to test it. This is the case of grammars using both the formalism of Lexicalized Tree Adjoining Grammars and frames as semantic representations. For example, the description of the syntax-semantics interface in (Zinova, 2017) uses a workaround (simulating the lexical insertion while creating the grammar, and not during parsing), as no parser for such a type of resource was available at the time. This paper introduces a parser which uses grammars of this type, based on an existing parser for LTAG: TuLiPA.

In Section 2., we provide a short explanation of the framework and show how it interacts with semantic parsing. In Section 3., we summarize the compact description of our resources in a metagrammar. In Section 4., we summarize the architecture of the TuLiPA parser that we are using. In Section 5., we introduce our new extension of TuLiPA to handle frame semantics, and give an example of use of this parser in Section 6..

2. Semantic Parsing with LTAG

2.1. Tree Adjoining Grammars

A Tree Adjoining Grammar consists of a set of elementary trees, from which larger trees are built using substitution and adjunction. Nodes in elementary trees have terminal or non-terminal labels. In a lexicalized TAG, every elementary tree has at least one terminal node. In a fully derived tree, all leaf nodes have terminal labels and all internal nodes have non-terminal labels. Elementary trees are either initial trees or auxiliary trees. The latter have one leaf node marked as a foot node by an asterisk. The foot node and the root node of the auxiliary tree have the same label. All trees may have leaf nodes that are marked as substitution nodes by an arrow. A derivation starts with an initial tree.

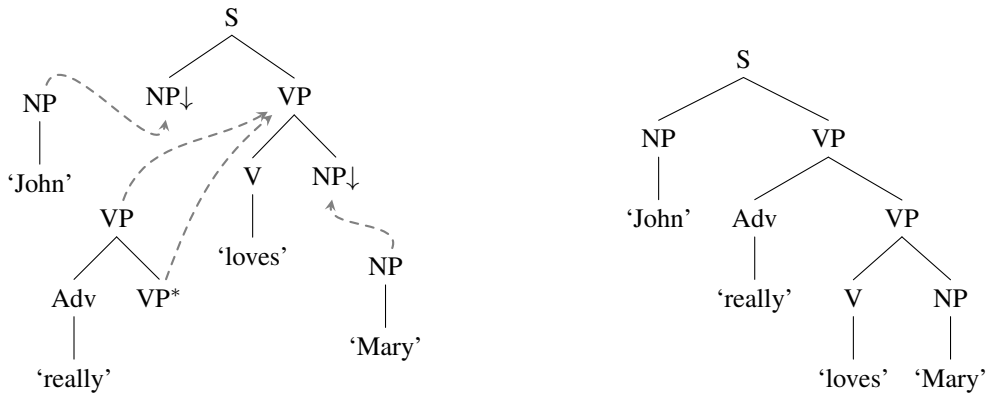
Two operations are used to combine trees during a derivation. In substitution, the root node of an elementary tree replaces a non-terminal leaf node of another tree, where both nodes have the same label. For adjunction, an auxiliary tree is adjoined to a target node in another tree. The root node of the auxiliary tree replaces the target node, and the part of the target tree below the target node is attached to the foot node in the auxiliary tree. An example is shown in Figure 1. In1a, the elementary trees for 'John' and 'Mary' are substituted at the subject and object argument slots in the initial tree for 'loves'. The adverbial modifier 'really' is adjoined at the VP node. The resulting derived tree is shown in 1b.

Elementary trees represent the constructional meaning of their lexical items in that they have substitution nodes for all arguments of the anchor. Recursive phenomena like modification are modelled by adjunction, which leads to long-distance dependencies in derived trees that were local in one elementary tree (Joshi and Schabes, 1997). Thus, all elementary syntactic structures can carry semantic information locally. Morphosyntactic information and links to the semantic representation are stored in feature structures at the nodes of elementary trees (Vijay-Shanker and Joshi, 1988). The acceptance of a derivation is determined by unification of those feature structures.

2.2. Frames and semantic parsing

In the syntax-semantics interface for LTAG proposed by (Kallmeyer and Osswald, 2013), semantic frames are represented as base-labelled, typed feature structures. They can be understood as a straightforward representation of the semantic and conceptual knowledge about a situation, while having good computational properties as their composition relies on the unification of attribute-value structures. LTAG elementary trees are paired with frames by using unification variables, as shown in Figure 2.

Here, the elementary tree for 'loves' is paired with a frame of type *love*, which has two attributes: an actor and a theme. This frame, labelled by e, is the semantic representation associated to the verb, therefore the variable e is shared with the feature structure of the syntactic node VP. The variables associated to the actor and the theme in the frame are shared with the two NP nodes of the syntactic tree (where they are values of the attribute I). The elementary trees for 'John'



(a) Elementary trees

(b) Derived tree

Figure 1: A derivation in LTAG.

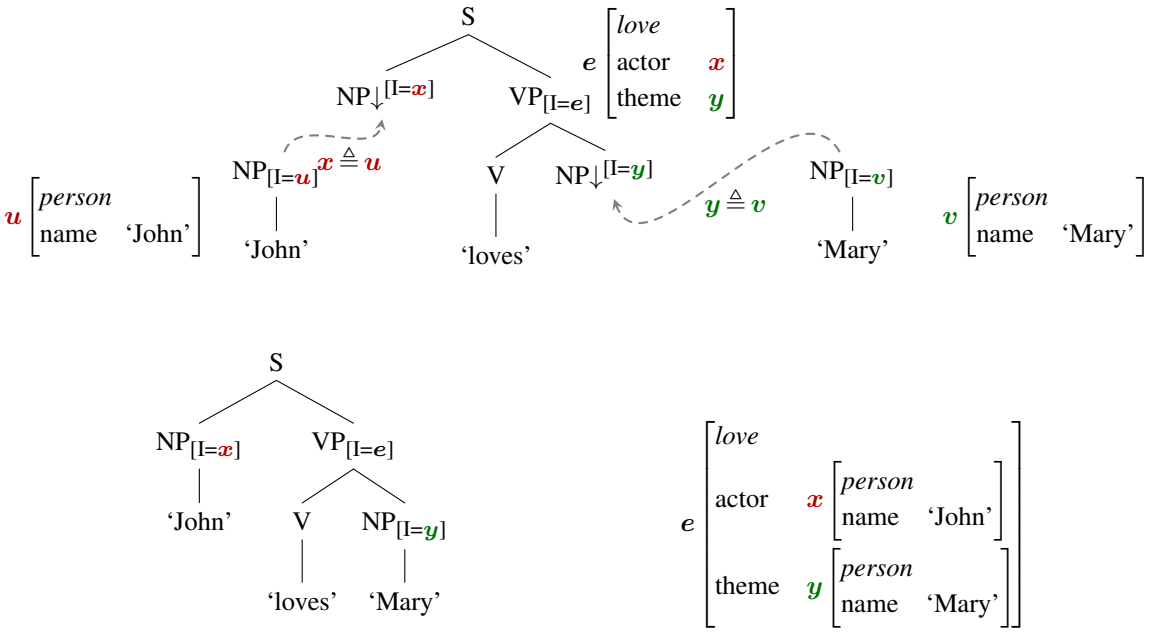


Figure 2: An example of pairing of LTAG trees and frames and the result of their combination.

and 'Mary' are both paired with frames of type *person*, where only the value of the attribute name differs, respectively labeled by the variables u and v.

During parsing, as syntactic trees are combined (by adjunction or substitution), the semantic representations are also combined. The unification of variables in the feature structures associated to the nodes triggers the unification of variables in the frames. In our example, as the substitution of the subject NP takes place (combining the elementary trees of 'love' and 'John'), the respective values associated to the attribute I in the feature structures are unified. This results in the unification of the variables x and u, which makes the frame for John become the actor of the event 'love'. The same happens when the tree for 'Mary' is substituted at the object NP node of the 'love' tree: y and v unify to let the frame for 'Mary' become the value of the theme attribute in the frame e.

3. XMG and XMG2

For the electronic description of such resources, the approach proposed by (Kallmeyer and Osswald, 2013) is to use a metagrammar. The latter makes the development and maintenance of grammars easier by allowing abstractions. This is especially useful in LTAG grammars, as they show a lot of structural redundancy. XMG, for eXtensible MetaGrammar (Crabbé et al., 2013), permits the generation of grammars from fully declarative specifications (called metagrammars) which are based on logic programming and constraints. The name XMG stands both for the description language and the compiler for this language, that is to say the tool that will create a grammar from a metagrammatical description. Its newer evolution XMG2 (Petitjean et al., 2016) allows for more flexibility regarding the type of linguistic data to describe, for instance frames which were not initially supported. XMG2 is not a more complex descrip-

tion language and compiler, but a tool allowing to create new compilers of the kind of XMG. It offers a collection of different languages and compilers (automatically generated) dedicated to different description tasks. This set of description tools allows us to generate all the resources that we need in this paper using a single framework.
XMG comes with a system of dimensions which allows to separate the levels of linguistic description (here syntax and semantics, but also lexicon). The ⟨**syn**⟩ dimension allows to describe trees by using dominance and precedence constraints between syntactic nodes, while the ⟨**frame**⟩ dimension allows the description of typed feature structures, as well as type hierarchies (Lichte and Petitjean, 2015). The ⟨**lemma**⟩ and the ⟨**morph**⟩ dimensions can be used to generate a lexicon. The architecture of the lexicon that we use for our parsing task will be described in the Section 6., together with examples of XMG2 code using the four previsously mentioned dimensions.
An XMG2 compiler takes as input a metagrammar, and produces the lexicon of all the structures described in it. In other words, it converts a compact representation of a resource into the resource itself. In our case, every entry of the generated grammar is a pair of an unanchored tree and a typed feature structure. Grammars generated with XMG2 can naturally be used for tasks such as generation or parsing, provided that the adapted tool exists. The next section introduces TuLiPA, which is one of these tools.

4. TuLiPA

Mildly context-sensitive grammar formalisms like TAG have been shown to capture complex natural language phenomena, such as cross-serial dependencies, while being parsable in polynomial time. TuLiPA ("Tübingen Linguistic Parsing Architecture" (Kallmeyer et al., 2008)) is a parser for several mildly context-sensitive formalisms, including LTAG. The LTAG grammars used by the original TuLiPA version can feature semantic information using predicate semantics as in the syntax-semantics interfaces of Gardent and Kallmeyer (2003) and Kallmeyer and Romero (2008).
The lexical information processed by TuLiPA is 2-layered. The morphological lexicon maps inflected tokens to their lemma, storing morphological information in a feature structure. The lemmas are stored with semantic information and the tree families to which the lemmas can be anchored. Before parsing, trees are anchored, i.e. for every word of the input sentence, possible elementary trees are selected where the semantic information on the lemma of the word matches with the information on the anchor node of the tree.
Therefore, the grammar used by TuLiPA, similarly to the XTAG grammar (XTAG Research Group, 2001), is composed of three elements: a lexicon of unanchored elementary trees (tree templates), a lexicon of lemmas and a lexicon of fully inflected forms. This is another level of factorization (in addition to the metagrammatical one presented in Section 3.) which helps reducing the size of the resource.
Currently, there are two parsing modes for parsing LTAG with semantic frames available in TuLiPA. As of (Kallmeyer et al., 2008), the input grammar is converted

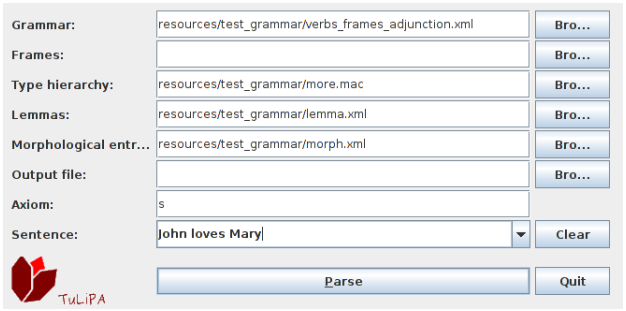


Figure 3: The user interface of TuLiPA

to a simple Range Concatenation Grammar (RCG, Boullier (1999)), which is used for parsing the input sentence, The TAG derivation structures are extracted from the RCG parsing results. Because this algorithm was designed to handle multi-component TAG (which is an extension of TAG), the conversion to RCG performed poorly on large-scale LTAG grammars. We included the implementation of a CYK parser by Thomas Schoenemann, based on the deduction rules given in (Kallmeyer, 2010). The parsing results, namely derivation trees, derived trees, derivation steps and possibly semantic representations, can be viewed in a graphical user interface or exported as XML files.
In the next section, we will describe how we extended TuLiPA to be able to process LTAG grammars including frame semantics.

5. A TuLiPA Extension for Frame Semantics

There are several motivations for the choice of TuLiPA as a starting point for our parser: with TuLiPA, we already have an open source LTAG parser, with graphical user interface, and multiplatform (as it is written in Java). As explained in the previous section, TuLiPA can already process semantic descriptions expressed as predicates.
Our extension, released like TuLiPA as an executable jar[1], accumulates typed feature structures instead of predicates during the parsing. The main changes are the following:

1. The grammars used are now composed of tree-frame pairs as presented in Section 2.. They are generated by XMG2, using the ⟨**syn**⟩ and the ⟨**frame**⟩ dimensions.

2. A type hierarchy (also produced by XMG2) must also be given to the parser to process the unification of typed feature structures.

3. The lexicon of trees and frames can be given separately. If this option is chosen, the tree-frame pairs are built during parsing (according to information provided by the lexicon).

Figure 3 shows the user interface of the parser where all the input data is given.
In the next section, we will go through all the steps which are necessary to create toy resources and recreate a parse example.

[1] https://github.com/spetitjean/TuLiPA-frames

```
1  class commonnoun
2  declare ?NP ?N ?X0 ?X1
3  {
4    <syn>{
5      node ?NP [cat=np, i=?X0];
6      node ?N (mark=anchor) [cat=n, i=?X1];
7      ?NP -> ?N
8    };
9    <frame>{
10     ?X0[pizza]
11   }
12 }
```

Figure 4: XMG2 description for a proper noun elementary tree and its frame.

```
1  class commonnoun
2  declare ?NP ?N ?X0 ?X1
3  {
4    <syn>{
5      node ?NP [cat=np, i=?X0];
6      node ?N (mark=anchor) [cat=n, i=?X1];
7      ?NP -> ?N
8    };
9      <iface>{[i=?X0]}
10 }
```

Figure 5: XMG2 description for a proper noun elementary tree.

6. Example of Parsing

To recreate the analysis of 'John eats pizza' similar to the one given in Figure 2, we will first see how to create the inputs needed by TuLiPA. The source code is available online, see 1. All the input files (tree templates, frames, lemma lexicon and morphological lexicon) are XML files produced by XMG2. The use of a single framework is a new feature: While the grammar and the frames were always produced by XMG, the morphological and lemma files used to be generated by a tool called lexConverter. The extensibility of XMG2 made it possible to create new compilers to generate these lexicons, using a consistent syntax and the same modular approach. In this work, we will use three different XMG2 compilers called `synframe`, `lex` and `mph`, which are accessible by the same command (`xmg compile`).

6.1. The grammar

The first resource to build is a LTAG grammar composed of at least 3 trees (transitive verb for 'eat', proper noun for 'John' and common noun for 'pizza') paired with their corresponding frames. These structures are described in the metagrammar and compiled using XMG2.

The XMG2 code describing a common noun tree and its frame can be as in Figure 4. Our architecture uses the notion of family, which are sets of trees which allow the same lexical anchors. In our example, the family `commonnoun` is composed of a unique tree, and the only lexical item compatible with this family is `pizza`.

```
1  class FramePizza
2  declare ?X0
3  {
4    <frame>{
5      ?X0[pizza]
6    };
7    <iface>{
8      [i=?X0]
9    }
10 }
```

Figure 6: XMG2 description for a single frame, to be paired with a syntactic tree during parsing.

On the first line, **class** `commonnoun` means that we define an XMG2 abstraction, which is in fact the tree-frame pair as it will be in the grammar. `?NP ?N ?X0 ?X1` are unification variables used in the class. The tags **syn** and **frame** separate the syntactic and the semantic descriptions. In the syntactic one, two nodes (`?NP` and `?N`) are created with feature structures (`?NP` has category `np`, etc.). The node `?N` is marked as an anchor node. Finally, the constraint on line 7 means that XMG2 will only generate trees where the NP node has the N node as daughter.

On the semantic side, we create a typed feature structure of type `pizza`, labeled by the variable `?X0`. However, pairing the frame for a pizza with the TAG tree for a common noun is not a very natural solution. We will now split this entry in two different lexicons: a purely syntactic lexicon of unanchored elementary trees and a purely semantic lexicon of frames. The syntactic tree `commonnoun` will be associated to the semantic frame `pizza` only when this token is read by the parser. The lemma lexicon will take care of this binding, as explained in the following subsection. This allows to associate different frames to one elementary tree, giving more flexibility.

In our case, we can split syntax and semantics as shown in Figures 5 and 6. The linking between the syntactic node and the semantic frame is this time done through the interface (**iface**). This dimension contains only a feature structure, which allows to share information between other dimensions. During parsing, when the elementary tree and the frame are paired, the two interfaces are unified, resulting in the unification of the two variables named `?X1` (which were independent until now, as every structure has its own namespace). The two files (syntactic and semantic) are compiled using the `synframe` compiler to produce the two lexicons.

6.2. The lemmas

The second step is to create a lexicon of lemmas containing at least three entries (one transitive verb and two nouns). The entry for pizza is as shown in Figure 7.

Here, we specify the name of the lemma (`entry`), its syntactic category `cat` and the family of trees which can be anchored by it (`fam`), which is the one we created in the previous XMG2 syntactic class, namely `commonnoun`. The value of the `sem` feature indicates which frame this lemma should be associated to. The class `LemmaPizza` is part of

```
class LemmaPizza
{
  <lemma> {
    entry <- "pizza";
    sem   <- FramePizza;
    cat   <- n;
    fam   <- commonnoun
   }
}
```

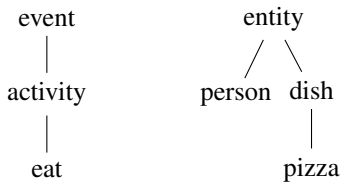
Figure 7: XMG2 description for the lemma 'pizza'.

```
class MorphPizza
{
  <morpho> {
    morph <- "pizza";
    lemma <- "pizza";
    cat   <- n;
    num   <- sg
   }
}
```

Figure 8: XMG2 description for the morphological entry for 'pizza'.

```
frame-types = {event, activity, eat,
        entity, person, dish, pizza}
frame-constraints = {
        activity -> event,
        entity event -> -,
        eat -> activity,
        person -> entity,
        dish -> entity,
        dish person -> -,
        pizza -> dish }
```

(a) Implementation in XMG2



(b) Graphical representation

Figure 9: The implementation and graphical representation of the type hierarchy used in 'John eats pizza'

a new XMG2 file, which must be compiled with the `lex` compiler.

6.3. The inflected forms

Finally, we must write a lexicon of at least three inflected forms ('John' for the lemma 'john', etc.). The entry for 'pizza' (one of the inflected forms of the previously defined lemma, along with 'pizzas' for example) is shown in the code of Figure 8. This entry associates the inflected form 'pizza' to the lemma `pizza`, with additional features, here only a syntactic category and a number. The XMG2 file containing this class amongst others can be compiled with the `mph` compiler to produce the lexicon of inflected forms.

6.4. The type hierarchy

The implementation of the type hierarchy in Figure 9a follows Lichte and Petitjean (2015). In the code, line 1 and 2 define the elementary types that are used in the hierarchy. From line 3 on, constraints are declared on these elementary types. Constraints in the form of line 4 express type subsumption: every `activity` is also an `event`. Constraints as in line 5 express that the unification of elementary types fails: When a minimal model of the type hierarchy is computed, `entity` and `event` do not have a common subtype. An example is provided in the next section. It is also possible to express constraints on the attributes of frames with a certain type. For a more detailled description, see (Lichte and Petitjean, 2015). The graphical representation in Figure 9b has to be read top-down, with the most general types on top.

6.5. The parse result

The result of the parse of the sentence 'John eats pizza', as shown by the graphical interface of the parser, is given in Figure 10.

On the left, we see the list of successful parses (only one), the set of elementary trees which were used to derive the selected parse, and the derivation steps. The derived tree is shown at the top right, with the semantic representation at the bottom.

The parse result is similar to the one which we gave in Figure 2. The semantic representation consists of one frame: It has type `activity-eat-event` and two attributes, the `actor` and `target` of the verb. Their values are shared with the syntax: The actor is of type `person-entity` and is unified with the interface feature in the subject-`np`-node through the variable `?B0`. The target is of type `pizza-dish-entity` and corresponds to the object-`np`-node, unified by the variable `?A0`. These unifications are triggered by substitution, and the insertion of 'john' into the frame is triggered by lexical anchoring. Note that the type `dish`, which is not present in the `eat` frame nor in the `pizza` frame, is inferred from the type hierarchy (as every frame of type `pizza` must also have type `dish`). The interface feature of the subject-`n`-node give an idea of this. Unification via adjunction works in the same way.

The types of the feature structures are here conjunctive types and get modified by the constraints expressed in the metagrammar. For instance, one constraint in our metagrammar specifies that all structures of type `person` also have type `entity`, hence the conjunctive types `[person-entity]`.

7. Conclusion

In this paper, we presented our parser for LTAG and frame semantics. The parser is an extension of TuLiPA and is consequently based on the same architecture, meaning that the grammar is separated into several levels: a lexicon of unanchored elementary trees paired with frames (provided

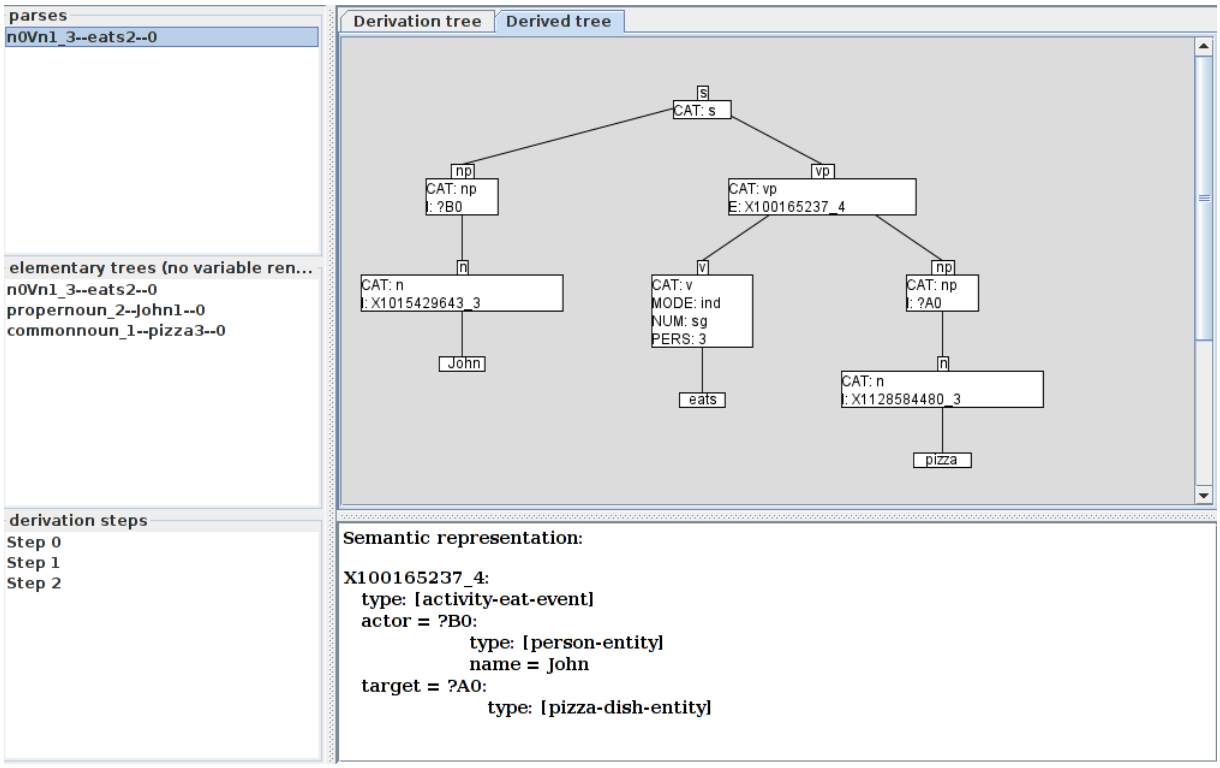


Figure 10: Result of the parse of 'John eats pizza'

together or separately), one of lemmas and one of inflected forms. The composition of frames, implemented following Kallmeyer and Osswald (2013), happens as elementary trees are combined, triggered by the unification of linked variables. In Section 6., we showed how to create the different resources to parse a simple example, and also that the analysis done by our tool for this example was the expected one. The availability of such a tool will make the use of the existing LTAG grammars including frame based semantics possible, and ease the creation of new ones.

As a next step, in order to improve parsing efficiency when using large grammar resources, we consider implementing grammar compression using subtree sharing and compression of these subtrees into minimal Finite State Automata, as described in (Waszczuk et al., 2016)

We are also working on a parser for Role and Reference Grammar (RRG, Van Valin (2005)), following the formalization proposed in (Kallmeyer and Osswald, 2017). RRG is a grammar theory based on flat constituent structures and focusing on semantic and pragmatic aspects. We can of course imagine that grammars based on RRG and frame semantics will be created, with a framework similar to the one presented in this paper, and tested with our tool.

8. Acknowledgements

The research presented here has been supported by the TreeGraSP Project funded by an ERC Consolidator Grant. We would like to thank three anonymous reviewers for their valuable comments on earlier drafts of this article.

9. Bibliographical References

Boullier, P. (1999). On TAG and Multicomponent TAG Parsing. Research Report RR-3668, INRIA.

Crabbé, B., Duchier, D., Gardent, C., Le Roux, J., and Parmentier, Y. (2013). XMG : eXtensible MetaGrammar. *Computational Linguistics*, 39(3):1–66.

Gardent, C. and Kallmeyer, L. (2003). Semantic construction in feature-based tree adjoining grammar. In *10th conference of the European Chapter of the Association for Computational Linguistics*.

Joshi, A. K. and Schabes, Y. (1997). Tree-Adjoining Grammars. In G. Rozenberg et al., editors, *Handbook of Formal Languages*, volume 3, pages 69–124. Springer, Berlin, New York.

Kallmeyer, L. and Osswald, R. (2013). Syntax-driven semantic frame composition in lexicalized tree adjoining grammars. *Journal of Language Modelling*, 1(2):267–330.

Kallmeyer, L. and Osswald, R. (2017). Combining predicate-argument structure and operator projection: Clause structure in role and reference grammar. In *Proceedings of the 13th International Workshop on Tree Adjoining Grammars and Related Formalisms*, pages 61–70. Association for Computational Linguistics.

Kallmeyer, L. and Romero, M. (2008). Scope and Situation Binding in LTAG using Semantic Unification. *Research on Language and Computation*, 6(1):3–52.

Kallmeyer, L., Lichte, T., Maier, W., Parmentier, Y., Dellert, J., and Evang, K. (2008). Tulipa: Towards a multi-formalism parsing environment for grammar engineering. In *Coling 2008: Proceedings of the workshop*

on Grammar Engineering Across Frameworks, pages 1–8, Manchester, England, August.

Kallmeyer, L. (2010). *Parsing Beyond Context-Free Grammars*. Springer Publishing Company, Incorporated, 1st edition.

Lichte, T. and Petitjean, S. (2015). Implementing semantic frames as typed feature structures with XMG. *Journal of Language Modelling*, 3(1):185–228.

Petitjean, S., Duchier, D., and Parmentier, Y. (2016). XMG 2: Describing Description Languages. In *Logical Aspects of Computational Linguistics. Celebrating 20 Years of LACL (1996–2016) 9th International Conference, LACL 2016, Nancy, France, December 5-7, 2016, Proceedings 9*, pages 255–272. Springer Berlin Heidelberg.

Van Valin, Jr., R. D. (2005). *Exploring the Syntax-Semantics Interface*. Cambridge University Press.

Vijay-Shanker, K. and Joshi, A. K. (1988). Feature structures based tree adjoining grammars. In *Proceedings of the 12th Conference on Computational Linguistics - Volume 2*, COLING '88, pages 714–719, Stroudsburg, PA, USA. Association for Computational Linguistics.

Waszczuk, J., Savary, A., and Parmentier, Y. (2016). Enhancing practical TAG parsing efficiency by capturing redundancy. In *21st International Conference on Implementation and Application of Automata (CIAA 2016)*, Séoul, South Korea, July.

XTAG Research Group. (2001). A Lexicalized Tree Adjoining Grammar for English. Technical Report IRCS-01-03, IRCS, University of Pennsylvania.

Zinova, J. (2017). *Russian Verbal Prefixation: A Frame Semantic Analysis*. Heinrich-Heine-Universität Düsseldorf dissertation.

Multilingual Dependency Parsing for Low-Resource Languages: Case Studies on North Saami and Komi-Zyrian

KyungTae Lim, Niko Partanen, Thierry Poibeau
LATTICE (CNRS & École normale supérieure / PSL & Université Sorbonne nouvelle Paris 3 / USPC)
1, rue Maurice Arnoux, 92120 Montrouge, France
kyungtae.lim@ens.fr, nikotapiopartanen@gmail.com, thierry.poibeau@ens.fr

Abstract

The paper presents a method for parsing low-resource languages with very small training corpora using multilingual word embeddings and annotated corpora of larger languages. The study demonstrates that specific language combinations enable improved dependency parsing when compared to previous work, allowing for wider reuse of pre-existing resources when parsing low-resource languages. The study also explores the question of whether contemporary contact languages or genetically related languages would be the most fruitful starting point for multilingual parsing scenarios.

Keywords: dependency parsing, word embeddings, Uralic languages

1. Introduction

Developing systems for low-resource languages is a crucial issue for Natural Language Processing (NLP). Most NLP systems are built using supervised learning techniques (Weiss et al., 2015; Straka et al., 2016; Ballesteros et al., 2016). These systems require a large amount of annotated data and are thus targeted toward specific languages for which this kind of data exists. Unfortunately, producing enough annotated data is known to be time- and resource-consuming, which means that annotated data, especially of the type required for parsing, is lacking for most languages. To take a recent example, the 2017 CoNLL Shared Task concerned around 50 languages, roughly all of the languages for which enough syntactically annotated data is available in the Universal Dependency format. This was probably (by far) the most ambitious parsing challenge ever undertaken with regard to language diversity, but the figure of 50 languages should be viewed relative to the 6,000 languages in the world: even if one includes only those languages for which written data is available, the 50 languages targeted at CoNLL 2017 cover only a fraction of all the world's languages.

When it comes to parsing, the supervised, monolingual approach based on syntactically annotated corpora has long been the most common one. However, thanks to recent developments involving feature-representation methods (a.k.a. word embeddings) and neural network models, it is now possible to develop accurate multilingual models, too. The multilingual approach has yielded encouraging results for both low- (Guo et al., 2015) and high-resource languages (Ammar et al., 2016a). Generally speaking, the multilingual approach can be implemented in two ways. The first involves projecting annotations available for a high-resource language onto a low-resource language using a parallel corpus, while the second aims at producing a cross-lingual transfer model that can work for several languages.

Guo et al. (2016) and Ammar et al. (2016a) have conducted multilingual parsing studies for Indo-European languages using the model transfer approach. They demonstrated that a multilingual model can yield better results than monolingual models for different European languages. However, their approach relied on the existence of a massive parallel corpus, as their experiment was based on Europarl.[1] Thus the problem of low-resource languages remains unaddressed, especially in cases when no parallel corpus is available. In this paper, we propose a simple but powerful method for creating a dependency parsing model when no annotated corpus or parallel corpus is available for training. Our approach requires only a small bilingual dictionary and the manual annotation of a handful of sentences. We assume that the performance one can obtain with this approach depends largely on the set of languages used to train the model. This is why we have developed several models using genetically related and non-related languages, so as to gain a better understanding of the limitations or possibilities of model transfer across different language families.

In this study, we are working with languages that, until very recently, did not or still do not have a Universal Dependency corpus: North Saami[2] and Komi-Zyrian, henceforth Komi.[3] (In this paper, we use ISO 639-3 codes to refer to the languages in tables.) This alone does not make them low-resource languages, but they are still poorly equipped with regard to NLP tools. Saami language technology has been in active development for a longer time in the Giellatekno project of the University of Tromsø,[4], but in the case of Komi, resources have only very recently begun to emerge. Work has also been done on these languages within the framework of language documentation; however, with the exception of a few individual projects (Blokland et al., 2015; Gerstenberger et al., 2016), this field has generally not paid much attention to the use of NLP tools. The same scenario applies to many other low-resource languages. On the other hand, the amount of written data in smaller languages may have grown rapidly in last years thanks to an increased online presence and various digitalization projects.

[1] www.statmt.org/europarl
[2] glottolog.org/resource/languoid/id/nort2671
[3] glottolog.org/resource/languoid/id/komi1268
[4] giellatekno.uit.no

This makes it possible to build word embeddings, as the only resource needed for these is a sufficiently large amount of text.

A major contribution of this work also lies in the new resources that have been developed as a result of it. We have created a new UD-type corpus for Komi[5] (version 0.1 used in this study), as well as bilingual dictionaries, multilingual word embeddings for Komi and Saami, and a multilingual parser, all of which are freely available in public repositories.[6] [7]

The structure of the paper is as follows. We first provide an overview of our approach (Section 2) before detailing the multilingual resource representation used (Section 3) and our parsing model (Section 4). We then describe a series of experiments aiming at validating the approach (Section 5) before presenting an analysis of our results along with some thoughts for future work (Section 6).

2. Approach

The languages used for training the models have been selected with the assumption that genetically related languages or contemporary contact languages, at least in certain scenarios, share structural similarities with the low-resource languages in question. Our aim is to test whether it is possible to use the annotated data from larger languages as part of the training data, exploiting this structural similarity.

Our work with Northern Saami was initially motivated by it being one of the surprise languages in the CoNLL 2017 Shared Task. After the Shared Task ended, the larger training data became available in the dev-branch of the UD_North_Sami GitHub repository in the Universal Dependencies project[8], which enabled us to carry out broader testing. With Komi-Zyrian, the situation is essentially the same as it was with the North Saami before the recent release of the new data: we have 85 manually annotated examples we can use to train and test the parsing result. The training and testing set was created manually for Komi for the purpose of this study, but we are currently expanding it. This Komi-Zyrian treebank will be included in Universal Dependencies project during 2018.

Currently, only English has been used as a control language with no direct contact or genetic relation with Komi or Saami. Our expectation is that the match rate between Russian and Komi-Zyrian should be exceptionally high, as there has been such a long history of contact between these languages, leading to a variety of morphosyntactic changes in Komi (Leinonen, 2006, p. 241). North Saami has a complex contact relationship with Finnish, but in addition to this, it is also a closely genetically related language (Aikio, 2012, p. 67–69). From this point of view, one could expect Finnish to perform well in parsing both Komi and North Saami, although the similarities between these languages have not been studied in great detail from the perspective of syntax and dependency structures. Other types of experiments have also been conducted using this approach, for example, by using a Komi-Zyrian–Russian multilingual model to parse data that contains both languages in the form of code-switching: in these tests, the parser has been shown to be able to analyse language-specific constructions when they occur within same utterance (Partanen et al., 2018).

[5] github.com/langdoc/UD_Komi-Zyrian

[6] github.com/jujbob/multilingual-bist-parser

[7] github.com/jujbob/multilingual-models

[8] github.com/UniversalDependencies/UD_North_Sami

3. Multilingual Lexical representation

3.1. Preparation of Language Resources

The bilingual lexicons were taken from the Giellatekno infrastructure (Giellatekno, 2017), as these are rather large and have been released with a GNU GPLv3 license. As some portions of this data are derived from printed dictionaries, not all entries are directly usable for our purposes. Multi-word translations and entries containing question marks have been removed. Table 1 shows the sizes of the dictionaries used.

Bilingual pairs	Bi-dictionary	Bi-embedding
Finnish–Komi	12,879	2.3GB
Finnish–North Saami	12,398	2.4GB
Komi–English	8,746	7.5GB
North Saami–Finnish	10,541	2.4GB
Russian–Komi	12,354	5.7GB

Table 1: Dictionary sizes and size of bilingual word embeddings generated by each dictionary.

We have used the pretrained Finnish and Russian FastText word embeddings published by Facebook in May 2017 (Bojanowski et al., 2016). Since the Komi and Saami Wikipedias are relatively small, we have also trained larger word embeddings using FastText. For Komi, we have used Public Domain books digitalized in the Fenno-Ugrica collection (`https://fennougrica.kansalliskirjasto.fi/`) and proofread by The Finno-Ugric Laboratory for Support of the Electronic Representation of Regional Languages in Syktyvkar (`http://komikyv.org/`). For North Saami, we have used the SIKOR North Saami free corpus (`http://hdl.handle.net/11509/100`), which has been published with a CC-BY 3.0 license.

3.2. Projection of Multiple Word Embeddings onto a Single Space

In the previous section, we described how we obtained and trained monolingual embeddings for each language, but each of those embeddings is trained in its own vector space. In order to transform the different embeddings into one single bilingual word embedding (encoded through a single vector space model), we apply the linear transformation method proposed by Artetxe et al. (2016). According to comparisons presented in Artetxe et al. (2017, p. 457), the size of dictionaries we used is well above what is needed to carry out the mapping task using this method.

The method is as follows. Let target language X and source language Y be the word embedding matrix trained by two different languages. And let $D=\{(x_i,y_i)\}_{i=1}^{m}$ (*where* $x_i \in X$, $y_i \in Y$) be a bilingual dictionary consisting of word-embedding vector pairs. Our goal is to find a transformation matrix W such that xW approximates y. This is done by

minimizing the sum of squared errors, following Mikolov et al. (2013):

$$\arg\min_{W} \sum_{i=1}^{m} \|x_i W - y_i\|^2 \quad (1)$$

However, the application of a linear transformation to an embedding without constraints may cause a degradation of performance since the computation of mapping may break monolingual invariance. Artetxe et al. (2016) proposed an orthogonal mapping method that lets W be an orthogonal matrix and makes it possible to avoid a degradation of performance.

We followed this method in order to map two different embeddings and produce bilingual word embeddings with language-specific prefixes preceding each surface forms. For example, in case of a English-French pair, the projected embedding can have "eng:dog" and "fra:chien". To train word embeddings related to more than two languages, we first selected a standard source language S and then mapped it with each target language $T=(t_1,t_2,t_i ..)$. Based on the trained parameter $W_T = (W_{t1}, W_{t2}, W_{ti} ..)$, we can thus build a multilingual word embedding by multiplying the parameter W_{ti} and all the surface forms included in t_i.

4. Cross-Lingual Dependency Parsing Model

Traditionally, many parsers have applied linear supervised learning models with hand-crafted feature functions. The feature function takes features for classifying head-modifier and relations among tokens (i.e. "word forms and POS tags from first and second tokens on top of the stack") (Kiperwasser and Goldberg (2016)). Parsers also require many templates in order to make a decision about the relations between tokens. However, extracting the proper features and templates manually is a difficult and time-consuming job. In order to address the limitation of manual work, Chen and Manning (2014) proposed using non-linear classifiers with a neural network model. This method encodes lexical (words) and non-lexical (POS tags) features as vectors and then concatenates the features of each token to feed the non-linear classifiers. This has two advantages: On the one hand, non-linear classifiers show better performance than linear models in identifying relations between tokens, and on the other hand, the use of a neural network with concatenated features alleviates the need for manual work because the neural model, especially in Recurrent Neural Networks (RNNs), has access to tokens and features computed previously for a given sentence.

Our basic feature representation approach is based on Chen and Manning (2014), with the exception of the method used for pretrained multilingual embeddings. In order to take into account lexical resources during parsing, we have extended the multilingual graph-based parser based on the bidirectional LSTM feature representations proposed by Lim and Poibeau (2017).

4.1. Bidirectional LSTM Feature Representations

Recent advances in NLP have been possible largely due to innovative feature representations that provide an accurate overview of word relations inside the sentence (Cho, 2015; Huang et al., 2015). A good example is the BIST-parser proposed by Kiperwasser and Goldberg (2016), which is based on bidirectional LSTM learning. BiLSTM is a powerful learning model for sequential data because it consists of two LSTM layers, a *Forward* layer that reads the sentence from left to right, and another that reads it from right to left. For example, given a sentence $t = (t_1,t_2,...,t_n)$, in which the symbol $\circ$ denotes a concatenation operation, the BiLSTM function can be represented as: $\text{BiLSTM}(t_{1:n}, i) = \text{LSTM}_{Forward}(t_{1:i}) \circ \text{LSTM}_{Backward}(t_{i:n})$.

4.2. Cross-Lingual Feature Representations

In their paper, Lim and Poibeau (2017) proposed a system for multilingual parsing based on bidirectional LSTM feature representations. They followed most of the approach used for the BIST-parser (see Kiperwasser and Goldberg (2016) and the previous section) and transformed it into a multi-source trainable model with multilingual word embeddings and language hot-encoding. This system obtained an LAS F1 score of 70.93% in the 2017 CoNLL Shared Task (rank 5/33). For this paper, we have extended the parser using the multilingual word embeddings proposed in Section 3, and have adapted the same token representation methods and dimensions proposed by Lim and Poibeau (2017).

Token Representation. Given an input sentence $t = (t_1,t_2,...,t_n)$, a word form w, a corresponding POS tag p, pretrained word embedding xw and language hot-encoding l, the i^{th} token is defined as: $t_i = e(w_i) \circ e(p_i) \circ e(xw_i) \circ e(l_i)$, where e denotes the embedding vector of each feature and $e(xw_i)$ is the pretrained word embedding introduced in Section 3. Additionally, we added a language hot-encoding vector composed of 0 and 1 for each language as proposed earlier (Naseem et al., 2012; Ammar et al., 2016a). Compared with monolingual parsers, most use $e(w_i)$ and $e(p_i)$, with additional features such as distance between head node and language-specific lexical features included in the UD corpus. Note that t_i feeds into $\text{BiLSTM}(t_{1:n}\ i)$ in order to store the *Forward* and *Backward* contexts from the LSTM.

4.3. Parsing Model

There are two mainstream approaches to parsing, one being the transition-based model (Nivre, 2004) and the other the graph-based model (McDonald et al., 2005b). For this study, we chose the graph-based approach based on the BIST-parser since graph-based approaches seem to show better performance for parsing UD-type corpora (Dozat et al., 2017). From the features and tokens stored in the BiLSTM layer, the BIST-parser computes a candidate tree for each head word and modifier, after which scores attached to the different candidate trees are computed using the multilayer perceptron (MLP), which is a basic neural network model that can be used as a scoring function. Finally, the system finds the best dependency parsing trees based on the sum of the subtrees. For further information on the graph-based and arc-factored model used in the BIST-parser, please see Taskar et al. (2005) and McDonald et al. (2005a).

Case	Training corpus	LAS	UAS
1	sme (20)	32.96	46.85
2	eng (12,217)	32.72	50.44
3	fin (12,543)	40.74	54.24
4	sme (20) + eng (12,217)	46.54	61.61
5	**sme (20) + fin (12,543)**	**51.54**	**63.06**

Table 2: Labeled attachment scores (LAS) and unlabeled attachment scores (UAS) for Northern Saami (sme)

5. Experiment

We conducted a series of experiments on Saami and Komi. For Saami, we tested different language combinations for the cross-lingual model. All the experiments were carried out using 20 training sentences in Saami, as was the case for the 2017 CoNLL Shared Task, which means these results can be compared to the ones in the official CoNLL evaluation. For Komi, no annotated corpus was available, but we designed ten different experiments, again exploring different language combinations for the cross-lingual model. The experiments for Komi are representative of an extremely low-resource scenario, which is quite common, meaning the approach can be reused for a wide variety of other languages.

Training Corpus. We used corpora available in the Universal Dependency 2.0 format (Nivre et al., 2017) to train and test all models except for Komi. Since there is no UD 2.0 Komi corpus, we used 10 sentences for training and 75 sentences for the testing set (the corpus was designed specifically for this study). Following previous works by Guo et al. (2015) and Zhang and Barzilay (2015), we used gold POS sets for training and testing for Komi. For Saami, however, the teams in the CoNLL Shared Task used preprocessed POS tagging sets processed by UDpipe. In order to maintain the same conditions as in the Shared Task, we also used the preprocessed POS tagging sets to compare with others in Table 3.

Training Conditions. Since we wanted to explore low-resource scenarios (even in the case of Saami, for which larger data now exists), we assumed that there had been no development data for parameter tuning, and we restricted all training experiments to run just one epoch with the training corpus without early stopping (i.e. the 5_{th}-low of column titled "Case" in Table 2 runs only 12,563 iterations). Similar restrictions have also been suggested by Ammar et al. (2016b), who proposed running one epoch, and by Guo et al. (2016), who proposed restricting iterations to 20,000 for low-resource scenarios. However, when training a model with multi-source training data, the size of training corpora for low-resource languages is comparably smaller than for high-resource languages. Following the previous work of Guo et al. (2016), we iterated 20 times more for low-resource training data than for high-resource. In Table 2 and Table 4, the sizes of the training sets used are provided in brackets.

Comparison with the CoNLL Shared Task. We used the same training environments in these experiments as for the CoNLL Shared Task (the same training sets and no development set). Moreover, in order to guarantee similar experimental conditions, we allowed the training models to run just one epoch, following Guo et al. (2016). Table 3 reports the official 2017 CoNLL results. C2L2 (Cornell Univ.) obtained the best performance for Saami with a delexicalized transfer approach (using a Finnish training corpus and a corpus of 20 Saami sentences as a development set for parameter tuning without lexicalized features). IMS (Stuttgart) used a delexicalized transfer approach with a very large training corpus based on 40 different training corpora in UD, obtaining the second-best result. Compared with the result of the Shared Task, it seems that our approach (lexicalized cross-lingual transfer parsing with resources from relevant languages) can be effective for parsing low-resource languages. However, additional language features and the application of the ensemble mechanism also seem to be very important. This is due to the fact that the team C2L2 outperform others with more than 6.4 percent of LAS score based on character embeddings as additional features and an ensemble mechanism composed of three different dependency parsers.

Team	LAS	UAS
C2L2 (Ithaca)	48.96	58.85
IMS (Stuttgart)	40.67	51.56
HIT-SCIR (Harbin)	38.91	52.51
This work	**42.50**	**54.94**

Table 3: The highest results of this experiment compared with top 3 results for Saami from the CoNLL 2017 Shared Task.

Case	Training corpus	LAS	UAS
1	kpv (10)	22.33	51.78
2	eng (12,217)	44.47	59.29
3	rus (3,850)	53.85	71.29
4	fin (12,543)	48.22	66.98
5	kpv (10) + eng (12,217)	50.47	66.23
6	kpv (10) + rus (3,850)	53.1	69.98
7	kpv (10) + fin (3,850)	53.66	71.29
8	kpv (10) + fin (12,543)	55.16	**73.73**
9	kpv (10) + eng (12,217) + fin (12,543)	52.5	68.57
10	**kpv (10) + rus (3,850) + fin (12,543)**	**56.66**	71.86

Table 4: Labeled attachment scores (LAS) and unlabeled attachment score (UAS) for Komi (kpv). We doesn't conduct training for "kpv + eng + rus" language combination because of unrealistic training scenario (It takes more than 40GB memory for training)

6. Analysis

All the experiments we conducted using Finnish for training obtained better results than other language combinations (i.e. English for Saami and Russian and English for Komi). This means that transferring knowledge from genetically related languages is, at least in our case, a very efficient method for parsing. This is true in the case of Saami and Finnish, but also in the case of Finnish and Komi, which are arguably more distantly related to one another than Finnish and Saami. A contact language can also be of significant help in improving the results with a low-resource language, as can be seen in the case of Russian and Komi.

Even in cases where the LAS scores (i.e. labels of the syntactic dependencies) were not significantly different, there was often greater variation in the UAS scores (unlabeled dependencies). This means that although the actual labels may not have been assigned correctly, the basic relations were found and the root was correctly recognized. It is also worth noting that the highest UAS scores were obtained with the Komi–Finnish pair, although the Komi–Russian–Finnish multilingual model gave the best overall result.

Our results show that adding even a small number of target-language example sentences into a parser that uses bilingual word embeddings can improve the result significantly. The size of the available training corpus is very important, and the quality of the bilingual dictionaries used to align word embeddings is also crucial.

7. Conclusion

In this paper, we have presented a multilingual approach to parsing that is effective for languages with few resources and no syntactically annotated corpora available for training. We have shown that our multilingual models provide better results than monolingual ones. Adding training material from other languages usually did not decrease the parsing result. It should, however, be noted that the relative size of the different corpora used for training seems to be relevant, since using corpora that are too imbalanced may weaken the result. More detailed analysis of the results, beyond the LAS and UAS scores, is most likely needed in order to determine the exact influences of different language pairs or combinations. It remains a question for further study whether the improvements observed here are actually attributable to the genetic relationship between the language, or if the same result could be obtained by simply selecting languages that are otherwise typologically similar.

Acknowledgments

We want to thank the LREC reviewers for their useful comments. This work has been developed in the framework of the LAKME project funded by a grant from Paris Sciences et Lettres (IDEX PSL reference ANR-10-IDEX-0001-02). Thierry Poibeau is partially supported by a RGNF-CNRS (grant between the LATTICE-CNRS Laboratory and the Russian State University for the Humanities in Moscow). Kyungtae Lim is partially supported by the ERA-NET Atlantis project. Niko Partanen's work has been carried out at the LATTICE laboratory within the project 'Language Documentation meets Language Technology: the Next Step in the Description of Komi', funded by the Kone Foundation. Thanks to Alexandra Kellner for proofreading the paper.

8. Bibliographical References

Aikio, A. (2012). An essay on Saami ethnolinguistic prehistory. In Riho Grünthal et al., editors, *A Linguistic Map of Prehistoric Northern Europe*, pages 63–117. Société Finno-Ougrienne.

Ammar, W., Mulcaire, G., Ballesteros, M., Dyer, C., and Smith, N. A. (2016a). One parser, many languages. *CoRR*, abs/1602.01595.

Ammar, W., Mulcaire, G., Tsvetkov, Y., Lample, G., Dyer, C., and Smith, N. A. (2016b). Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925*.

Artetxe, M., Labaka, G., and Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *EMNLP*, pages 2289–2294.

Artetxe, M., Labaka, G., and Agirre, E. (2017). Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462.

Ballesteros, M., Goldberg, Y., Dyer, C., and Smith, N. A. (2016). Training with exploration improves a greedy stack lstm parser. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2005–2010, Austin, Texas, November. Association for Computational Linguistics.

Blokland, R., Fedina, M., Gerstenberger, C., Partanen, N., Rießler, M., and Wilbur, J. (2015). Language documentation meets language technology. In *Septentrio Conference Series*, number 2, pages 8–18.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *EMNLP*, pages 740–750.

Cho, K. (2015). Natural language understanding with distributed representation. *arXiv preprint arXiv:1511.07916*.

Dozat, T., Qi, P., and Manning, C. D. (2017). Stanford's graph-based neural dependency parser at the conll 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada, August. Association for Computational Linguistics.

Gerstenberger, C., Partanen, N., Rießler, M., and Wilbur, J. (2016). Utilizing language technology in the documentation of endangered Uralic languages. *Northern European Journal of Language Technology*, 4:29–47.

Guo, J., Che, W., Yarowsky, D., Wang, H., and Liu, T. (2015). Cross-lingual dependency parsing based on distributed representations. In *Proceedings of ACL*.

Guo, J., Che, W., Yarowsky, D., Wang, H., and Liu, T. (2016). A representation learning framework for multi-source transfer parsing. In *AAAI*, pages 2734–2740.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Kiperwasser, E. and Goldberg, Y. (2016). Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL*, 4:313–327.

Leinonen, M. (2006). The russification of Komi. Number 27 in Slavica Helsingiensia, pages 234–245. Helsinki University Press.

Lim, K. and Poibeau, T. (2017). A system for multilingual dependency parsing based on bidirectional lstm feature representations. In *Proceedings of the CoNLL 2017*

Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pages 63–70, Vancouver, Canada, August. Association for Computational Linguistics.

McDonald, R., Crammer, K., and Pereira, F. (2005a). Online large-margin training of dependency parsers. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 91–98. Association for Computational Linguistics.

McDonald, R., Pereira, F., Ribarov, K., and Hajič, J. (2005b). Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 523–530, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013). Exploiting similarities among languages for machine translation.

Naseem, T., Barzilay, R., and Globerson, A. (2012). Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 629–637. Association for Computational Linguistics.

Nivre, J., Agić, Ž., Ahrenberg, L., et al. (2017). Universal dependencies 2.0, CoNLL 2017 shared task development and test data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Nivre, J. (2004). Incrementality in deterministic dependency parsing. In *Proceedings of the Workshop on Incremental Parsing: Bringing Engineering and Cognition Together*, pages 50–57. Association for Computational Linguistics.

Partanen, N., Lim, K., Rießler, M., and Poibeau, T. (2018). Dependency parsing of code-switching data with cross-lingual feature representations. In Tommi A. Pirinen, et al., editors, *Proceedings of the 4th International Workshop on Computational Linguistics for Uralic languages*, ACL Anthology, pages 1–17. Association for Computational Linguistics.

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia. European Language Resources Association.

Taskar, B., Chatalbashev, V., Koller, D., and Guestrin, C. (2005). Learning structured prediction models: A large margin approach. In *Proceedings of the 22nd international conference on Machine learning*, pages 896–903. ACM.

Weiss, D., Alberti, C., Collins, M., and Petrov, S. (2015). Structured training for neural network transition-based parsing. *CoRR*, abs/1506.06158.

Zhang, Y. and Barzilay, R. (2015). Hierarchical low-rank tensors for multilingual transfer parsing. Association for Computational Linguistics.

9. Language Resource References

Giellatekno. (2017). *The Northern Sami-Finnish-Northern Sami Dictionaries of Neahttadigisánit*. Available online, SVN revision r157659, ISLRN 452-785-459-365-4.

FonBund: A Library for Combining Cross-lingual Phonological Segment Data

Alexander Gutkin, Martin Jansche, Tatiana Merkulova
Google
London, United Kingdom
{agutkin, mjansche, merkulova}@google.com

Abstract

We present an open-source library (FonBund) that provides a way of mapping sequences of arbitrary phonetic segments in International Phonetic Alphabet (IPA) into multiple articulatory feature representations. The library interfaces with several existing linguistic typology resources providing phonological segment inventories and their corresponding articulatory feature systems. Our first goal was to facilitate the derivation of articulatory features without giving a special preference to any particular phonological segment inventory provided by freely available linguistic typology resources. The second goal was to build a very light-weight library that can be easily modified to support new phonological segment inventories. In order to support IPA segments that do not occur in the freely available resources, the library provides a simple configuration language for performing segment rewrites and adding custom segments with the corresponding feature structures. In addition to introducing the library and the corresponding linguistic resources, we also describe some of the practical uses of this library (multilingual speech synthesis) in the hope that this software will help facilitate multilingual speech research.

Keywords: phonology, phonetic segments, software

1. Introduction

Speech and language technology is currently only available for a tiny fraction of the world's languages. There has been a growing awareness of the importance of addressing this disparity, especially in recent years. One of the outcomes of this realization is the appearance of several linguistic typology resources (Forkel, 2014) that aim to organize the world's languages according to their structural and functional features (O'Horan et al., 2016). One typical example is URIEL (Littell et al., 2017), a resource (and the corresponding software) that collates various features from various existing databases (Hammarström et al., 2015; Dryer and Haspelmath, 2013; Moran et al., 2014) that describe languages in terms of their phonological, lexical, morphosyntactic, phylogenetic and geographic distribution properties.

Linguistic typology resources have been used in several ways to address the data scarcity problem for under-resourced languages. The first approach is multilingual joint learning where one hopes that training multiple languages jointly will help with pooling the resources across languages and boost the performance on under-resourced language. Another popular approach is language transfer, where a resource-rich language is used to improve the performance of a resource-scarce language via model and data transfer (O'Horan et al., 2016). These approaches were successfully used in several recent speech and language tasks such as grapheme-to-phoneme conversion (Deri and Knight, 2016; Peters et al., 2017), multilingual language modeling (Tsvetkov et al., 2016), text-to-speech (Tsvetkov, 2016), predicting missing language representation features (Malaviya et al., 2017) and name tagging (Zhang et al., 2017), among others (O'Horan et al., 2016).

The focus of our work is on the linguistic resources that offer typology features describing the phonological structure of the world's languages. Such resources are extremely useful in multilingual speech research. Consider a multilingual joint training approach to text-to-speech. In this scenario the training set contains diverse corpora from many sources representing many languages and dialects following different phonological transcription conventions. In order to train an acoustic model on such data, each phoneme inventory ideally needs to be transformed into a uniform canonical representation. In our work we use a representation based on the International Phonetic Alphabet (2015), or IPA, which is also used by all phonological segment inventories described in this study.

The conversion process may be quite involved because it requires linguistic expertise for constructing the mappings into IPA for languages employing custom representations. Additional difficulties arise when these mappings disagree due to differences between transcribers, diverging transcription conventions, or the lack of native speakers to guide the design. For example, a decision to represent many Nepali diphthongs as atomic members of the phoneme inventory may not be the most optimal choice. This process can be facilitated by the use of linguistic typology resources. The PHOIBLE (Moran et al., 2014) database, for example, can provide guidance on which IPA segments are more likely out of a list of candidates for the mapping. PanPhon (Mortensen et al., 2016) can help establish whether the candidate constitutes a well-formed IPA segment.

An even bigger issue we have encountered is that for certain under-resourced languages it may be difficult to establish a faithful phoneme inventory due to the lack of linguistic resources and/or expertise. In such case, a linguistic typology resource, such as PHOIBLE, may help to establish the initial phoneme inventory for the language (at the time of writing PHOIBLE supports 2,155 phoneme inventories for 1672 distinct languages).

Once the multilingual corpus is transformed into a uniform representation, the next important step is to decide on a representation of phonological segment structure in terms of articulatory features and to derive this structure from the IPA segments provided by the multilingual corpus. In

this paper we present an open-source library called *FonBund*[1] that was developed to facilitate this step. FonBund wraps phonological segment inventories (and their corresponding unique feature systems). At present, PHOIBLE, PanPhon and PhonClassCounts (Dediu and Moisik, 2015) databases are supported and tested but the library is flexible enough to support other representations. Our design goal was to make the library agnostic to a particular choice of phoneme inventory because several phoneme inventories for any given language may be devised based on different linguistic sources. The library provides a simple interface that rewrites any sequence of IPA segments into the desired articulatory feature representation (or combinations thereof, if multiple representations are requested) that can be used as discrete features in machine learning algorithms. This paper is organized as follows: A brief overview of phonological segment databases that FonBund currently supports is provided in Section 2. An overview of the core library design is given in Section 3. One of the possible applications of this library, namely speech synthesis for languages not in the training data, is described in Section 4. Paper is concluded in Section 5.

2. Phonological Segment Inventories

This section briefly describes the three databases and the respective feature systems that our library currently supports. Our primary focus is on the global segment inventories that contain a list of all the unique phonetic segments (or *segment types*) encountered for all the languages along with their corresponding articulatory feature representations.

2.1. PHOIBLE

PHOIBLE (Moran, 2012) is a freely available database containing cross-linguistic phonological data compiled from many linguistic sources. The online 2014 edition (Moran et al., 2014) includes 2155 phoneme inventories with 2160 segment types found in 1672 distinct languages. We primarily investigated the current PHOIBLE online segment inventory[2], but also looked at a slightly older version from the CLLD collection[3].

According to its documentation (Moran et al., 2014), the feature system in PHOIBLE is "loosely based on Hayes (2009) and Moisik and Esling (2011), but goes beyond both of these sources to be descriptively adequate cross-linguistically" and is likely to change as new languages are added. Overall the feature system consists of 37 "binary" features (such as [±labiodental] and [±spreadGlottis]) that for the simple segments take the ternary values: present $(+)$, absent $(-)$ and not applicable (0). For complex segments, such as diphthongs, tuples of the above values are used. For example, the value of a vowel feature [±syllabic] for diphthong /ɛw/ is a pair $(+, -)$.

2.2. PanPhon

PanPhon is resource consisting of a database that relates over 5,000 IPA segments (simple and complex) to their definitions in terms of about 23 articulatory features and a Python package to manipulate the segments and their feature representations (Mortensen et al., 2016). Unlike PHOIBLE, which documents the actual snapshot of contemporary phonological knowledge of the world's languages from the standpoint of linguistic theory, PanPhon's mission is to develop a methodologically solid resource to facilitate research in NLP. One of the nice features of PanPhon is its great flexibility, which is achieved as follows: The resource contains a core set of approximately 146 core segments represented in IPA and their corresponding features. This core set is then extended by application of rules written in a user-editable YAML syntax (Ben-Kiki et al., 2009). The rules describe the application of diacritics and modifiers, the feature specifications that provide the necessary context for the modification and articulatory feature changes required if the diacritic or modifier is applied. Over 5,000 segments are compiled from the core set using the above procedure[4]. This set can be easily extended further to cover non-trivial segments by writing new rules.

Similar to PHOIBLE, a ternary system is used to represent each of the (evolving set of) 23 articulatory features loosely based on well-established phonological classes: *major* ([±syllable], [±sonorant], [±consonantal], [±continuant]), *laryngeal* ([±voice], [±spread glottis], [±constricted glottis]), *major place* ([±anterior], [±coronal], [±labial], [±velaric], [±distributed]), *minor place* ([±high], [±low], [±back]), *manner* ([±nasal], [±lateral], [±delayed release], [±strident]) and *minor manner* ([±round] [±tense], [±long]).

2.3. PhonClassCounts

Dediu and Moisik (2016) note that segment-level databases, such as PHOIBLE, cannot be used directly for generalizations over classes of segments that share theoretically interesting features, such as "retroflex stops". They introduce a method for defining a set of "atomic" (more phonetic) features that help deriving interesting sets of classes generalizing over the existing segment inventories. We denote the resulting resource and the corresponding software that they released (Dediu and Moisik, 2015) as *PhonClassCounts*.

Of particular interest to us is the *Fonetikode* feature system (Dediu and Moisik, 2016) provided by PhonClassCounts resource. Inspired by IPA, Fonetikode is a feature system consisting of 13 phonetically inspired multivalued features. For example, the [initiation] feature can take values from the set (pulmonic egressive, glottal ingressive, glottal egressive, velaric ingressive). An encoding of the PHOIBLE segment inventory using the Fonetikode representation is available as part of the PhonClassCounts resource[5].

The Fonetikode encoding of the segment database collected and curated by Merritt Ruhlen and released by Creanza et al. (2015) is also available as part of PhonClassCounts but has not been investigated in this work because it has sig-

[1] https://github.com/googlei18n/language-resources/tree/fonbund/fonbund

[2] https://github.com/phoible/dev/tree/master/raw-data/FEATURES

[3] https://github.com/clld/phoible/data

[4] https://github.com/dmort27/panphon/blob/master/panphon/data/ipa_all.csv

[5] https://github.com/ddediu/phon-class-counts/blob/master/input/phoible_Features_Fonetikode.csv

nificantly smaller coverage than PHOIBLE and serves a different purpose (the corpus indicates availability of 728 phonemes in 2028 distinct languages).

3. Overview of Library Design

FonBund parses a stream of phonetic segments (in IPA) and outputs articulatory features collected from several phonemic databases into a single output file in protocol buffer format (Google, 2008). The message defined by the protocol buffer (`DistinctiveFeatures`[6]) contains a list of articulatory features for a given input IPA segment. The size of articulatory feature list depends on the number of phonological segment databases configured. The message format supports both binary (PHOIBLE, PanPhon) and multivalued (Fonetikode) features. For example, if both PHOIBLE and PanPhon representations for a segment /t̪/ are requested, the resulting articulatory feature list will consists of 60 features (37 for PHOIBLE and 23 for PanPhon).
This unified format can be consumed directly or easily transformed for use by machine-learning frameworks such as TensorFlow (Abadi et al., 2016). A schematic representation of FonBund's operation on a possible broad phonetic transcription of the Danish word *mørk* (/m œɐ̯ g/) is shown in Figure 1. The algorithm produces three distinct articulatory feature representations for each of the three input segments.

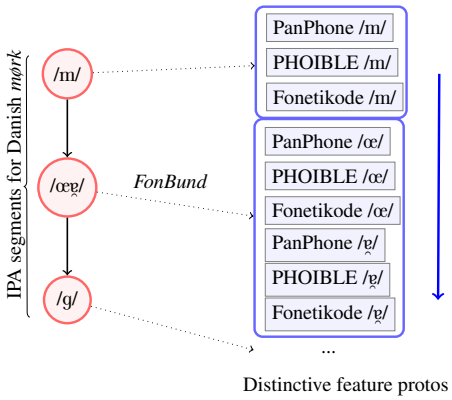


Figure 1: FonBund flow for a simple input.

Some segments have complex structure and may not necessarily have a one-to-one mapping to the segments in the databases. This applies to classes of segments such as diphthongs and in general to any segments describing non-trivial multiple articulations. We currently do not provide an algorithm for automatic decomposition of segments such as /œɐ̯/. Instead, FonBund requires such segments to be decomposed prior to calling the library, so the above segment should be rewritten using the provided delimiter + as /œ+ɐ̯/. The library still treats such a decomposition as a single segment returning articulatory feature representations for each of the components in decomposition.
The library is implemented in Python and affords significant flexibility in setting up additional phonological segment inventories. We are using the Bazel build system (Google, 2017) to configure the phonetic inventory

[6] https://github.com/googlei18n/language-resources/blob/fonbund/fonbund/distinctive_features.proto

Code	Description	Language Family	Train	Test
bn-BD	Bangla (Bangladesh)	Indo-Aryan	✓	×
bn-IN	West Bengali (India)	Indo-Aryan	✓	✓
en-IN	Indian English	Germanic	×	✓
hi-IN	Hindi (India)	Indo-Aryan	✓	×
ml-IN	Malayalam (India)	Dravidian	✓	×
mr-IN	Marathi (India)	Indo-Aryan	×	✓
si-LK	Sinhala (Sri-Lanka)	Indo-Aryan	✓	×
ta-IN	Tamil (India)	Dravidian	×	✓
te-IN	Telugu (India)	Dravidian	×	✓

Table 1: Languages for training and testing.

Configuration	Segment ID	PHOIBLE	PanPhon	PhonClassCounts
B	✓	×	×	×
PH	×	✓	×	×
PP	×	×	✓	×
PC	×	×	×	✓
B+PH	✓	✓	×	×
B+PP	✓	×	✓	×
B+PC	✓	×	×	✓
B+PH+PP	✓	✓	✓	×
B+PH+PP+PC	✓	✓	✓	✓

Table 2: Input features for acoustic models.

databases (in comma or whitespace-separated format) as remote build resources. In addition, we maintain a configuration file in protocol buffer format that describes, for each database, the necessary information on how to parse it: the database-specific basic segment normalization details, the type of the feature system (binary versus multivalued), the number of features and so on. The parsing logic is implemented by the `SegmentRepositoryReader` interface. In addition, we provide simple utilities for displaying the raw contents of supported databases (`show_segments.py`) and for converting broad phonetic transcriptions to articulatory feature representations using any combination of supported databases (`features_for_segments.py`).

4. Experiments

In what follows we describe one of the obvious applications of the FonBund library: multilingual text-to-speech synthesis. We are particularly interested in synthesizing speech for languages that are not encountered in the training data. The main goal of the experiments is to answer the question whether articulatory features derived from cross-lingual phonological segment databases can boost the performance of a multilingual text-to-speech system by providing richer structure than plain phonetic segment identities when training the multiple languages jointly. The second question is which representation out of the three databases currently supported by FonBund is more suitable for our application.

4.1. Experimental Setup

The multilingual corpus consists of nine speech databases of South Asian languages (English, Hindi, Malayalam, Marathi, Sinhala, Tamil, Telugu, and Indian and Bangladeshi dialects of Bengali) from both the Indo-Aryan and the Dravidian language family, shown in Table 1, where a language is identified by its BCP-47 language and region tag (Phillips and Davis, 2009). The region tags help us distinguish the Bengali dialect spoken in India from the Ben-

Code	B	PH	PP	PC	B+PH	B+PP	B+PC	B+PH+PP	B+PH+PP+PC
bn-IN	3.40±0.09	3.16±0.11	3.28±0.11	3.22±0.11	3.37±0.12	3.30±0.09	3.24±0.12	3.12±0.09	3.49±0.12
en-IN	2.93±0.11	**3.50±0.09**	3.39±0.12	3.18±0.11	3.42±0.10	3.36±0.10	3.24±0.12	3.45±0.11	3.42±0.13
mr-IN	3.35±0.11	3.43±0.10	3.39±0.11	3.36±0.09	3.27±0.12	3.39±0.11	3.27±0.09	3.38±0.12	3.31±0.09
ta-IN	2.08±0.09	2.56±0.08	2.62±0.08	2.67±0.09	2.53±0.07	2.58±0.09	2.68±0.08	**2.68±0.07**	2.66±0.09
te-IN	3.16±0.13	3.70±0.12	3.41±0.11	3.82±0.10	3.52±0.11	3.42±0.12	**3.85±0.11**	3.62±0.12	3.80±0.10

Table 3: Subjective Mean Opinion Scores (MOS) (along with 95% confidence intervals) for languages synthesized with various acoustic model configurations. Best scores are underlined. Statistically significant improvements shown in bold.

gali of Bangladesh. Each database (apart from Hindi) contains recordings from multiple speakers and genders.

Given the multilingual corpus we generate nine different training data configurations. Each configuration corresponds to a particular type and combination of the input features and is shown in Table 2. The baseline (B) corresponds to the input features consisting solely of phonetic segment identities (e.g., /ɛw/). The remaining eight configurations correspond to either replacing the segment identity features with the articulatory features from one of the segment databases or using the segment identity features in conjunction with articulatory feature combinations from multiple segment inventories. For example, the features in configuration B+PH+PP consist of segment identities and articulatory features from PHOIBLE and PanPhon. No other input features apart from the ones described are used, in order to keep the experiment pure.

For each of the nine configuration we trained an LSTM-RNN acoustic model, the details of which are described in Gutkin and Sproat (2017). Each model was evaluated on five South Asian languages from Table 1. Out of five languages tested, Marathi, Tamil and Telugu are completely unseen during training. Indian English is less challenging since some of our Hindi database contains English prompts. Finally, West Bengali is an in-domain language for this test. The motivation behind selecting this particular group of South Asian languages is to investigate how the presence or absence of articulatory features from various sources affects the synthesis of languages for which we have no training data (Tamil, Telugu and Marathi) vs. the languages for which some data is available (Bengali and Indian English).

Each configuration was evaluated using subjective Mean Opinion Score (MOS) listening tests. For each test we used 100 sentences not included in the training data for evaluation. Each rater was a native speaker of the language and was asked to evaluate a maximum of 100 stimuli. Each item was required to have at least 8 ratings. The raters used headphones. After listening to a stimulus, the raters were asked to rate the naturalness of the stimulus on a 5-point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). Each participant had one minute to rate each stimulus. The rater pool for each language included at least 8 raters. For each language, all configurations were evaluated in a single experiment.

4.2. Evaluation Results and Discussion

Table 3 shows the results of subjective listening tests for five languages where, for each language, nine acoustic model configurations described in the previous section were tested. Each mean opinion score is shown along with the corresponding confidence interval statistics at 95% confidence level (Wonnacott and Wonnacott, 1990) computed using the recommendations in ITU-T P.1401 (2012). The highest scores are underlined. The best configurations which exhibit no overlap in confidence intervals are deemed statistically significant and shown in bold.

For all five languages, using some combination of articulatory features results in improvements over the baseline configuration. For three languages (Indian English, Tamil and Telugu) these improvements are large, while for West Bengali and Marathi the improvements are not statistically significant. We hypothesize that for these two languages, the slightly disappointing results are not due to the use of cross-lingual segment repositories per se, but rather due to the suboptimal design of our phoneme inventories.

It is interesting to note that for Indian English and Marathi one can safely replace the segment identity features with the articulatory features derived from PHOIBLE, while improving upon the baseline. As can be seen from Table 3, this result cannot be replicated for PanPhon or PhonClassCounts inventories. In addition, we note that there is no clear "winning" feature representation out of PHOIBLE (PH), PanPhon (PP) and PhonClassCounts (PC). Combining them individually with the segment identity features leads to big improvements for Telugu (B+PC), while more complex combinations strongly improve Tamil (B+PH+PP).

5. Conclusion and Future Work

This paper introduced an open-source library for mapping sequences of arbitrary IPA segments to multiple articulatory feature representations currently based on three popular cross-language phonological databases. The library is flexible and can be extended to support additional phonological databases. Applying the library to the domain of multilingual text-to-speech synthesis confirms the hypothesis that articulatory features derived from cross-language databases are very useful and in certain situations can replace the original phonological segment identity features altogether.

While at present the library is restricted to phonological information, we are planning to extend it to other representations, such as the morphosyntactic representation offered by WALS (Dryer and Haspelmath, 2013) and phylogenetic and geographical representations from Glottolog (Hammarström et al., 2015). We are also planning to apply the library to other speech and language tasks.

6. Acknowledgments

The authors would like to thank Rob Clark, Richard Sproat and the anonymous reviewer for helpful suggestions.

7. Bibliographical References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). TensorFlow: A System for Large-Scale Machine Learning. In *OSDI*, volume 16, pages 265–283.

Ben-Kiki, O., Evans, C., and Ingerson, B. (2009). *YAML Ain't Markup Language (YAML™) Version 1.2*. http://www.yaml.org/spec/1.2/spec.html.

Creanza, N., Ruhlen, M., Pemberton, T. J., Rosenberg, N. A., Feldman, M. W., and Ramachandran, S. (2015). A comparison of worldwide phonemic and genetic variation in human populations. *Proc. of the National Academy of Sciences*, 112(5):1265–1272.

Dediu, D. and Moisik, S. (2015). *Features (and feature counts) from PHOIBLE and Ruhlen's databases*. Max Planck Institute for Psycholinguistics, Nijmegen, Netherlands, December. https://github.com/ddediu/phon-class-counts/.

Dediu, D. and Moisik, S. (2016). Defining and Counting Phonological Classes in Cross-linguistic Segment Databases. In *LREC 2016: 10th International Conference on Language Resources and Evaluation*, pages 1955–1962, Slovenia, May. European Language Resources Association (ELRA).

Deri, A. and Knight, K. (2016). Grapheme-to-Phoneme Models for (Almost) Any Language. In *Proc. ACL 2016: 54th Annual Meeting of the Association for Computational Linguistics*, pages 399–408, Germany, August.

Dryer, M. S. and Haspelmath, M. (2013). WALS online. *Leipzig: Max Planck Institute for Evolutionary Anthropology*. http://wals.info/.

Forkel, R. (2014). The Cross-Linguistic Linked Data project. In *3rd Workshop on Linked Data in Linguistics: Multilingual Knowledge Resources and Natural Language Processing*, pages 61–66, Iceland, May.

Google. (2008). Protocol Buffers. Google's Data Interchange Format. https://developers.google.com/protocol-buffers/.

Google. (2017). Bazel: A fast, scalable, multi-language and extensible build system. https://bazel.build/.

Gutkin, A. and Sproat, R. (2017). Areal and Phylogenetic Features for Multilingual Speech Synthesis. In *Proc. of Interspeech 2017*, pages 2078–2082, Sweden, August.

Hammarström, H., Forkel, R., Haspelmath, M., and Bank, S. (2015). Glottolog. *Leipzig: Max Planck Institute for Evolutionary Anthropology*. http://glottolog.org.

Hayes, B. (2009). *Introductory Phonology*. Blackwell Textbooks in Linguistics. Wiley-Blackwell, Oxford.

IPA. (2015). International Phonetic Alphabet. Technical report, International Phonetic Association. https://www.internationalphoneticassociation.org/content/ipa-chart.

ITU-T P.1401. (2012). Methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models. *International Telecommunication Union*, July.

Littell, P., Mortensen, D., Lin, K., Kairis, K., Turner, C., and Levin, L. (2017). URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proc. EACL 2017*, pages 8–14, Spain, April. European Chapter of the Association for Computational Linguistics.

Malaviya, C., Neubig, G., and Littell, P. (2017). Learning Language Representations for Typology Prediction. In *Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2529–2535, Denmark.

Moisik, S. and Esling, J. H. (2011). The "whole larynx" approach to laryngeal features. *Proc. of ICPhS: International Congress of Phonetic Sciences*, pages 1406–1409, August.

Moran, S., McCloy, D., and Wright, R. (2014). *PHOIBLE Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. http://phoible.org/.

Moran, S. (2012). Using Linked Data to Create a Typological Knowledge Base. In Christian Chiarcos, et al., editors, *Linked Data in Linguistics*, pages 129–138. Springer.

Mortensen, D., Littell, P., Bharadwaj, A., Goyal, K., Dyer, C., and Levin, L. (2016). PanPhon: A Resource for Mapping IPA Segments to Articulatory Feature Vectors. In *Proc. COLING 2016: 26th International Conference on Computational Linguistics*, pages 3475–3484, Japan, December. https://github.com/dmort27/panphon/.

O'Horan, H., Berzak, Y., Vulić, I., Reichart, R., and Korhonen, A. (2016). Survey on the Use of Typological Information in Natural Language Processing. In *Proc. of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 1297–1308, Japan.

Peters, B., Dehdari, J., and van Genabith, J. (2017). Massively Multilingual Neural Grapheme-to-Phoneme Conversion. In *Proc. of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 19–26, Denmark.

Phillips, A. and Davis, M. (2009). BCP 47 - Tags for Identifying Languages. *IETF Trust*.

Tsvetkov, Y., Sitaram, S., Faruqui, M., Lample, G., Littell, P., Mortensen, D., Black, A. W., Levin, L., and Dyer, C. (2016). Polyglot Neural Language Models: A Case Study in Cross-Lingual Phonetic Representation Learning. In *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1357–1366, San Diego, California.

Tsvetkov, Y. (2016). *Linguistic Knowledge in Data-Driven Natural Language Processing*. Ph.D. thesis, Georgia Institute of Technology.

Wonnacott, T. H. and Wonnacott, R. J. (1990). *Introductory Statistics*, volume 5. Wiley New York.

Zhang, B., Lu, D., Pan, X., Lin, Y., Abudukelimu, H., Ji, H., and Knight, K. (2017). Embracing Non-Traditional Linguistic Resources for Low-resource Language Name Tagging. In *Proc. of the 8th International Joint Conference on Natural Language Processing*, volume 1, pages 362–372, Taiwan.

Voice Builder: A Tool for Building Text-To-Speech Voices

Pasindu De Silva[2], Theeraphol Wattanavekin[1], Tang Hao[1], Knot Pipatsrisawat[1]
[1]Google,[2]Google (on contract from Teledirect Pte Ltd)
{pasindu,twattanavekin,tanghao,thammaknot}@google.com

Abstract

We describe an opensource text-to-speech (TTS) voice building tool that focuses on simplicity, flexibility, and collaboration. Our tool allows anyone with basic computer skills to run voice training experiments and listen to the resulting synthesized voice. We hope that this tool will reduce the barrier for creating new voices and accelerate TTS research, by making experimentation faster and interdisciplinary collaboration easier. We believe that our tool can help improve TTS research, especially for low-resourced languages, where more experimentations are often needed to get the most out of the limited language resources.

Keywords: text-to-speech, voice building, opensource

1. Introduction

There is growing demand for text-to-speech (TTS) voices in various industries as the world moves towards technologies that can interact with users more naturally. Many industries including gaming, film, appliances, and accessibility (Szarkowska, 2011) now have vested interest in creating TTS voices. As a result, there is a lot of needs to produce a variety of good quality voices in different languages. The TTS problem is challenging especially for low-resourced languages, because of the lack of data. Smaller text and audio corpora imply that more experiments might need to be done in order to get the most out of the available resources. Tackling this problem for low-resourced languages usually requires close collaboration between computer scientists, who understand the algorithms and know how to utilize the tools, and linguists, who understand linguistic phenomena and can help provide linguistic resources required to improve the voices.

From our experience building voices for low-resourced languages, we have observed many pain points that prevent these researchers from conducting quick experiments and iterating on their existing baseline voices. These pain points include (i) the complexity of setting up initial voice building infrastructure, which often requires fairly strong technical background to setup and maintain (ii) the heterogeneous nature of different tools, which requires a non-trivial amount of effort to *stitch* the required pieces together to form a working pipeline for TTS research.

In this paper, we propose *Voice Builder*, a text-to-speech voice building tool, which comes with a simple web-based user-friendly interface that enables researchers to quickly build and listen to prototype voices and to iteratively improve them with ease.

2. Related Work

There are many opensource TTS systems available with underlying algorithms ranging from classification regression trees in Festival (Taylor et al., 1998; Black and Taylor, 1997), hidden Markov models in HTS (Zen et al., 2007), to neural networks in Merlin (Wu et al., 2016). Some of these tools depend on components of their predecessors. For example, Merlin requires HTS or Festival to handle the linguistic frontend processing. Some systems, such as MaryTTS (Schröder et al., 2011), try to provide more end-to-end solutions for building various components, such as recording and transcription, required for a complete TTS system. Other tools focus only on certain parts of the TTS pipeline. For example, FreeTTS (W. Walker, 2009) allows users to synthesize audio from TTS models produced by other tools, while Sparrowhawk (Google, 2018a; Ebden and Sproat, 2015) is an opensource text-normalization system.

3. Voice Builder and the Problems It Addresses

Each of the tools mentioned above has its own strengths based on the approach it uses. When experimenting with new data for a language, we often wish for a quick way to build initial voices from different approaches in order to establish a good baseline. In such cases, the quality of the voice can be quickly assessed even without all the bells and whistles that would be needed in a production system. However, connecting the aforementioned tools together to form a working pipeline and/or setting up multiple algorithms for TTS research on the same platform is a daunting task. Users often end up spending too much time setting up and maintaining end-to-end system, rather than focusing on the real research.

Voice Builder is a TTS voice building tool that was designed to allow users to quickly build and listen to initial voices. It consists of a web frontend that allows users to easily build and listen to synthesized voices, regardless of their technical ability. Voice Builder by itself is not a TTS engine, but, rather, allows arbitrary engines (such as Festival, Merlin, MaryTTS, etc) to be plugged in and used for training voice models. Voice Builder will serve as the user interface and workflow manager that connects all the necessary steps together. Once setup, users need to fill out a simple web form and click a button to initiate a voice model training process. Voice Builder focuses on only the following 3 necessary pieces of language resources to start training a voice model: audio-transcription pairs, lexicon, and phonology. Once the job finishes, users can visit Voice Builder UI to listen to audio synthesized from arbitrary

text that the users can input. Figure 1 gives a high-level overview of Voice Builder.

3.1. Making Voice Building More Accessible

Building a new TTS voice has always been a fairly technical task. There is a lot of data processing to be done and the TTS tools available often require no-nonsense technical knowledge to harness. This technical barrier makes it difficult for the general public or language enthusiasts to contribute. After all, we cannot expect computer scientists to speak all languages, especially the long-tailed ones. Therefore, collaboration with native speakers or linguists is required in order to build voices for these languages. Voice Builder addresses this problem by providing a dead simple user interface (UI) to allow anyone with just basic computer knowledge to initiate a voice model training experiment. Figure 2 shows a portion of Voice Builder's graphical user interface for starting model training. As long as the necessary data is provided in a web form, the users can start voice building by just clicking a button; no further skill is needed. Then the user just needs to wait for the job to finish. Once model training is finished, the user can come back to the same UI to enter some text and listen to the synthesized voice. This simple process makes it easier for anyone to get involved in building voices. Moreover, Voice Builder's web UI allows collaborators across the world to work on the same project. By listening to the resulting voice, a non-technical native speaker might already be able to provide valuable feedbacks to more technical team members to further improve the voice.

3.2. Easier Experimentation

When working on TTS for low-resourced languages, one often needs to run many experiments on different data sets and to try on different training parameters. When dealing with many (often parallel) experiments, it is important to keep track of the settings and environment of each experiment for later diagnosis. Handling this manually is a potentially error-prone process; there could be many different files to keep track of. Voice Builder solves this problem by managing and isolating each experiment into its own sandbox. The configuration and data used to initiate each run is stored in a location dedicated to that experiment. These data are immutable for later analysis. Moreover, different TTS engines (or even different versions of the same engine) can be modularly added to the system, making it easy to trace back to earlier environments, if needed.

3.3. Support for Multiple Algorithms

The ability to experiment with many voice building algorithms is an important part of achieving high quality TTS voices for low-resourced languages. As a result, Voice Builder was designed with this in mind. Users will be able to easily plug in additional voice training engines into our system. We utilize Docker (Merkel, 2014) containers to encapsulate these engines. Therefore, integrating an additional engine only amounts to creating a Docker image and a configuration file.

3.4. Zero-ops Cloud Services

Creating a web-based service typically requires a substantial effort to manage and maintain. This is not the case for Voice Builder, because we chose to utilize Google Cloud Platform services for our various components. The web frontend is hosted by Google Compute Engine (Google, 2018f). All training data and configuration for each experiment are stored in the designated location on Google Cloud Storage (Google, 2018e). TTS engines and synthesizers are hosted in containers via Docker images. Lastly, we use Google Cloud Functions (Google, 2018c) to connect different stages of the pipeline together. This means that the users will get the security and reliability benefits without having to maintain any server themselves.

3.5. Custom Pre-processing

Different research groups may have different formats for various data files, depending on the tools used to obtain the data. Voice Builder was designed with this use case in mind and allows a custom data pre-processor to be used. When the user initiates a voice training experiment in the UI, a *create voice* request is sent to a (customizable) API endpoint to initiate pre-processing. The system expects the output processed data to be put in the designated job folder on Google Cloud Storage. Once all the required files are present in the folder, the system will automatically trigger the rest of the pipeline.

4. System Architecture

In this section, we explain the underlying architecture in more details. Figure 3 shows the system diagram of Voice Builder. We will now explain the components in this system from the perspectives of model training and voice synthesis.

4.1. Voice Model Training

When a user starts a voice model training process, they go through the following flow:

1. User enters information about locations of training data on the web frontend. The necessarily language resources are (i) audio-transcription pairs (ii) lexicon (iii) phonology definition. In the frontend, the user can also select the TTS engine to use and adjust the training parameters. A training specification containing all the parameters is created. The frontend also displays the status of each training job.

2. The training configuration is sent over to an optional pre-processing unit. This unit is responsible for processing and outputting all the necessary files to the cloud storage location specified in the configuration. If this unit is not present, all training data is gathered directly into the cloud storage location by the frontend.

3. Once all the required files are present, a cloud function will automatically trigger the training process by spawning a TTS engine in a container. The engine spawned is determined by the parameters in the configuration.

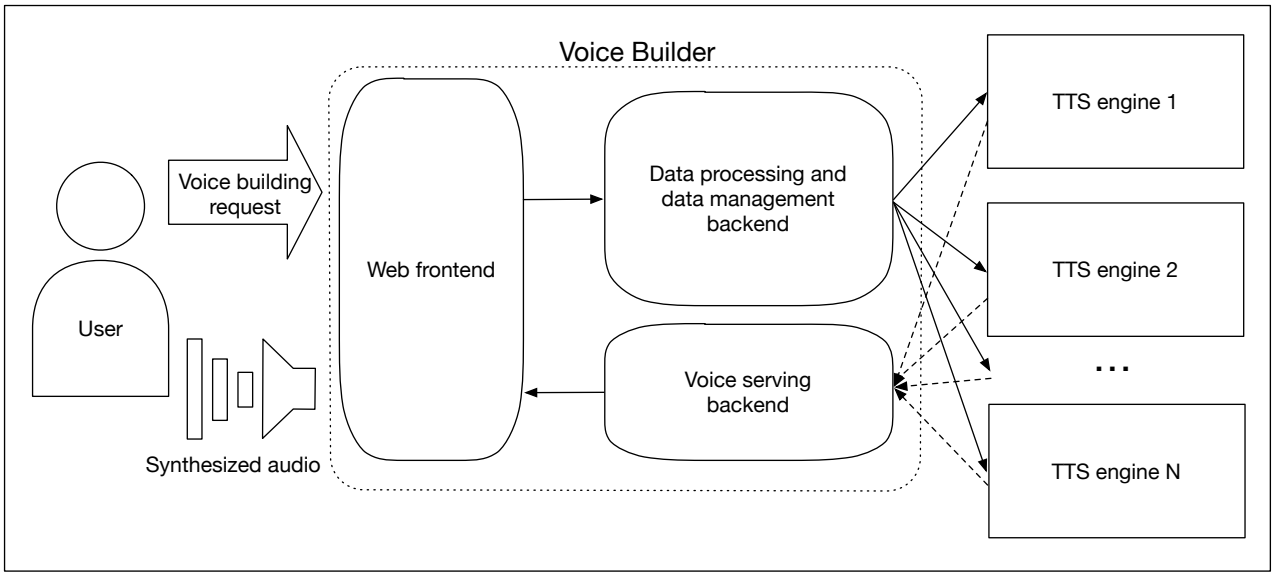


Figure 1: An overview diagram of Voice Builder.

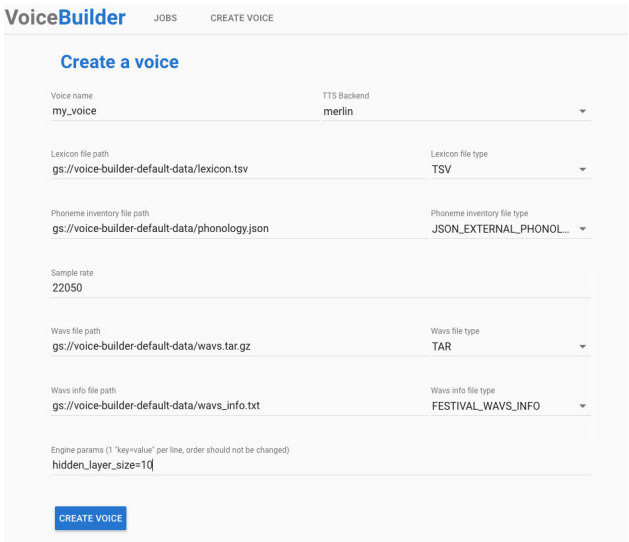


Figure 2: A screenshot of Voice Builder's graphical user interface for training a new voice.

4. The specified training engine is created in a container and initiates training based on the given specifications. Once the process finishes, the output model is saved at the designated cloud storage location.

5. Once the model is present, a cloud function will trigger a process to deploy a voice synthesis server using a container. This server is now ready to serve voice synthesis requests.

4.2. Voice Synthesis

When a user wants to listen to the voice synthesized by a trained model, they go through the following flow:

1. The user visits the frontend page that corresponds to the finished training job. On that page, the user can enter arbitrary text to be synthesized.

2. The request is sent to the voice synthesis server associated with that job.

3. Audio is synthesized and returned to be played on the frontend.

Notice that we utilize many Google Cloud Services that are publicly available in our system. The reasons for using cloud services, as opposed to building own servers, are as follow:

- **Encapsulation of different training engines.** Container technology allows us to be flexible and extensible in our support of various model training engines. It enables us to easily wrap the engines and integrate them with other parts of our system. Google Cloud Platform (GCP) (Google, 2018d) has various container solutions, two of which are used by our system:

    1. *Genomics Pipeline* (Google, 2018g) is used for voice model training. It provides an easy way to run and monitor any command. Input and output files are copied from and to Google Cloud Storage and are fully managed by the pipeline. It also has a simple template to help users plug in any model training engine easily.

    2. *Containers on Compute Engine* (Google, 2018b) is used to serve our frontend as well as the synthesis servers. Containers make our components platform-agnostic, which means that they can be run anywhere. Users can download a model to run locally on their machines or send to others to run using the same container image. In most cases, the same container images used for model training can also be used here. These characteristics make our system more portable, easier to deploy and maintain at scale.

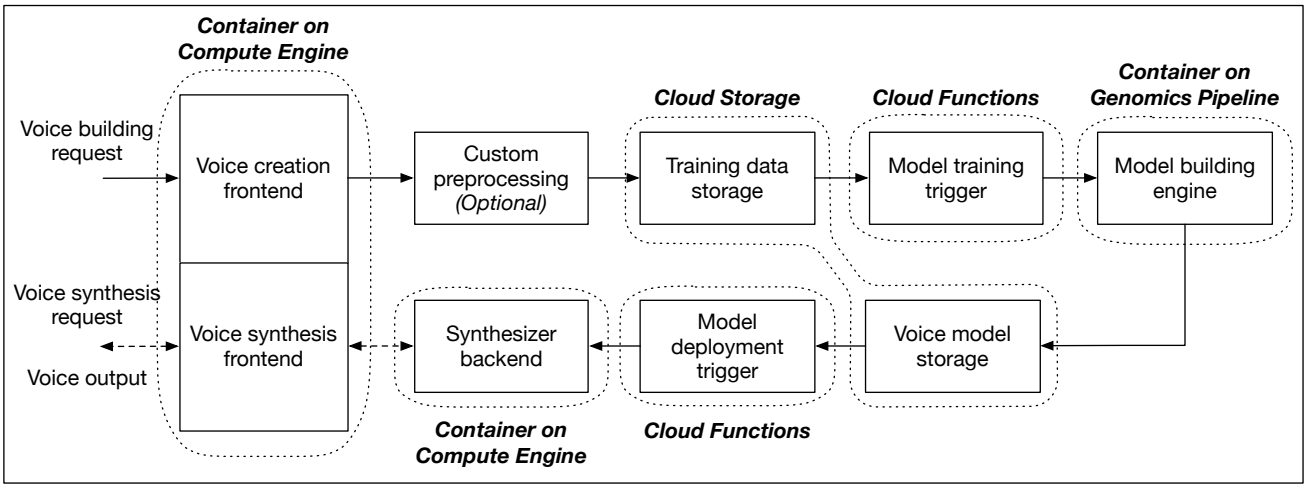


Figure 3: System diagram of Voice Builder. Each block corresponds to a system component. Each dotted bubble indicates a cloud service used for the enclosed component. Voice model training flow is indicated by solid arrows, while voice synthesis flow is indicated by dashed arrows.

- **Simpler connective layer.** Creating a business logic pipeline that stitches all components together is not a trivial task. Google Cloud Platform provides smooth integration between different products via Cloud Functions. Instead of managing our own event messages, Cloud Functions give us a powerful native triggering mechanism. For example, Cloud Functions can respond to change notification from Google Cloud Storage by automatically deploying a voice model after all the required files have been placed in the designated location on Google Cloud Storage.
- **Public availability and ease of use.** All GCP products in our system are available for the public. Setting up our system requires virtually no other additional software, making the installation process easy.

These cloud services help make our system flexible, reliable, and easy to set up.

5. Available Resources

We have opensourced the Voice Builder source code under the Apache 2.0 license. The code is available at https://github.com/google/voice-builder with setup instructions and some examples. In this release, we provide integrations with Festival and Merlin as a starting point for users.

6. Future Work

In the future, we would like to make Voice Builder even more useful for researchers to evaluate voices and perform diagnosis. Our hope is to make evaluation and diagnosis tools available right in the web frontend. For example, the frontend could allow users to synthesize a batch of sentences from a pair of models and present them for side-by-side listening test. For diagnosis, we hope to be able to provide basic alignment information used during training so that linguists could identify problematic data easily.

7. Conclusions

In this paper, we present a tool for creating TTS voices. The goals of this work is to make creating new TTS voices simpler, more flexible, and accessible by everyone. We designed the system to be user-friendly by providing an easy-to-use web frontend. Voice Builder allows for easy integration with publicly available TTS engines. Moreover, we utilize cloud services to run our system, thus, making our tool globally accessible to collaborators without requiring them to maintain their own servers. We believe that this tool could help move TTS research forward, especially for low-resourced languages, by facilitating experimentation and encouraging collaboration among researchers, no matter where they are in the world.

8. Bibliographical References

Black, A. W. and Taylor, P. A. (1997). The Festival Speech Synthesis System: System Documentation. Technical Report HCRC/TR-83, Human Communciation Research Centre, University of Edinburgh, Scotland, UK. Available at http://www.cstr.ed.ac.uk/projects/festival.html.

Ebden, P. and Sproat, R. (2015). The kestrel tts text normalization system. *Natural Language Engineering*, 21(3):333–353.

Google. (2018a). https://github.com/google/sparrowhawk. Accessed: 2018-02-19.

Google. (2018b). Containers on compute engine. https://cloud.google.com/compute/docs/containers/. Accessed: 2018-02-12.

Google. (2018c). Google cloud functions. https://cloud.google.com/functions/. Accessed: 2018-02-12.

Google. (2018d). Google cloud platform. https://cloud.google.com/. Accessed: 2018-02-12.

Google. (2018e). Google cloud storage. https://cloud.google.com/storage/. Accessed: 2018-02-12.

Google. (2018f). Google compute engine. https://cloud.google.com/compute/. Accessed: 2018-02-12.

Google. (2018g). Google genomics pipeline. https://cloud.google.com/genomics/reference/rest/v1alpha2/pipelines. Accessed: 2018-02-12.

Merkel, D. (2014). Docker: Lightweight Linux Containers for Consistent Development and Deployment. *Linux J.*, 2014(239), March.

Schröder, M., Oliva, M. C., Pammi, S., and Steiner, I. (2011). Open source voice creation toolkit for the MARY TTS Platform. In *Proceedings of Interspeech 2011*. ISCA, August.

Szarkowska, A. (2011). Text-to-speech Audio Description: Towards Wider Availability of AD. 15:142–163, January.

Taylor, P. A., Black, A., and Caley, R. (1998). The Architecture of the Festival Speech Synthesis System. In *The Third ESCA Workshop in Speech Synthesis*, pages 147–151, Jenolan Caves, Australia.

W. Walker, P. Lamere, P. K. (2009). A speech synthesizer written entirely in the Java(TM) programming language. https://freetts.sourceforge.io/, March. Accessed: 2018-02-12.

Wu, Z., Watts, O., and King, S., (2016). *Merlin: An Open Source Neural Network Speech Synthesis System*, pages 218–223. ISCA, September.

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A. W., and Tokuda, K. (2007). The HMM-based speech synthesis system (HTS) version 2.0. In *SSW*.

Sudachi: a Japanese Tokenizer for Business

Kazuma Takaoka[†], Sorami Hisamoto[†], Noriko Kawahara[†],
Miho Sakamoto[†], Yoshitaka Uchida[†], Yuji Matsumoto[‡]
[†]Works Applications
[‡]Nara Institute of Science and Technology
{takaoka_k hisamoto_s, kawahara_n, sakamoto_mi, uchida_yo}@worksap.co.jp, matsu@is.naist.jp

Abstract

Tokenization, or morphological analysis, is a fundamental and important technology for processing a Japanese text, especially for industrial applications. However, we often face many obstacles, such as the inconsistency of token unit in different resources, notation variations, discontinued maintenance of the resources, and various issues with the existing tokenizer implementations. In order to improve this situation, we develop a tokenizer called *Sudachi* and its accompanying dictionary with features such as multi-granular output and normalization of notation variations. In addition to this, we continuously maintain our software and language resources in long-term as a part of the company business. We release the resulting tokenizer software and language resources freely available to the public as an open source software. You can access them at https://github.com/WorksApplications/Sudachi.

Keywords: Tokenization, Morphological Analysis, Segmentation, Part-of-Speech Tagging, Lemmatization, Open Source Software

1. Introduction

Unlike whitespace separation between words for English text, Japanese text does not contain explicit word boundary information. We need unobvious methods to recognize words within a text. For Japanese text, the process equivalent to "tokenization" in other languages is often called "morphological analysis". It consists of three subprocesses, namely segmentation, part-of-speech (POS) tagging, and lemmatization (In the following we use the word "tokenization" to indicate these three processes).

The definition of a token in Japanese is not trivial; People develop various systems and resources, each with different kinds of the standard. Lack of token unit compatibility is one of the critical problems of Japanese language resources. When we look at the applications of Japanese text processing in various business scenes, applying parsing or more advanced language process is not common. It is typical to just conduct tokenization and use its post-processed output. For many companies tokenization is a fundamental and important technology for text processing. However, when increasing number of companies are demanding Japanese text processing recently, we are lacking freely available and useful resources for tokenization.

In order to improve this situation, we develop a new Japanese tokenizer and dictionary for business use. We make them available to the public as an open source software (OSS).

2. Previous Work

2.1. Japanese Tokenizers

When conducting Japanese tokenization for business applications, in the majority of cases *MeCab*[1] (Kudo et al., 2004) or *Kuromoji*[2] (the re-implementation of MeCab) are used.

MeCab can process text at excellent speed, however, its functions are limited to segmentation, POS tagging, and lemmatization; Users need to pre-/post-process the text by themselves. It is common to conduct text formatting, sentence segmentation, and character normalization as pre-processing. Typical post-processing includes simple chunking (e.g., for numeric expressions) and filtering by POS tags. Each user performs these processes on their own, therefore we tend to reinvent the wheel, or conduct such processes in inefficient ways.

There are two versions of Kuromoji, the standalone tool and the one integrated into a search library Apache Lucene[3]. The former version has issues similar to MeCab, and for the latter, although it provides some pre-/post-processing functions as part of the search system, we can not use them outside Lucene.

2.2. Language Resources

For the systems such as MeCab and Kuromoji, the language models are independent of the system, in form of dictionaries. The user may select a resource for tokenization from publicly available choices.

IPADIC (Asahara and Matsumoto, 2003) is the most widely used resource for Japanese tokenization; However it has not been updated since 15 years ago, therefore the dictionary lacks new words and the bug fixes have not been applied.

NAIST Japanese Dictionary[4], a dictionary developed based on IPADIC, aimed to solve the license issues of IPADIC, as those issues make it difficult to use the resource for OSS purposes. However it is currently not widely used, as the dictionary lacks some essential vocabularies, and IPADIC license issues have been solved subsequently.

UniDic[5] (Den et al., 2007; Kouno and Ogiso, 2015) (National Institute for Japanese Language and Linguistics, 2017) is a project to develop a Japanese electronic dictionary with uniformity and identity. The outcome is used

[1] http://taku910.github.io/mecab/
[2] https://www.atilika.com/ja/kuromoji/
[3] https://lucene.apache.org/
[4] https://ja.osdn.net/projects/naist-jdic/
[5] http://pj.ninjal.ac.jp/corpus_center/unidic/

for building Corpus of Spontaneous Japanese (CSJ)[6] and Balanced Corpus of Contemporary Written Japanese (BCCWJ)[7] (Maekawa, 2008). The project also offers a dictionary for conducting tokenization using MeCab. To emphasize the reproducibility of annotation, it adopts shorter token units in order for the annotators to process text unambiguously. To ensure the reproducibility the segmentation rules are defined as operating procedures, therefore it may get annotated in unintuitive fashions. It is shown effective for search purposes (Takahashi and Sassano, 2016), however, it is not suitable for syntactic or semantic analysis (National Institute for Japanese Language and Linguistics, 2017). The number of language resources derived from BCCWJ is growing, therefore UniDic is adopted more in the academic fields.

NEologd[8] (Sato et al., 2016; Sato et al., 2017) is a resource released as an add-on for other dictionaries such as IPADIC or UniDic. It consists of new words that are not included in those dictionaries. These new words are extracted from many language resources on the Web automatically or semi-automatically, and it is frequently updated (currently twice a week). This dictionary contains vocabularies in longer unit compare to other resources; We may say that these vocabularies are more of named entities (NEs) instead of words. If we apply these vocabularies directly to the search systems, we tend to get missing search results, as the shorter tokens that constitute the NEs are not indexed. The authors claim that the user can get shorter tokens by recursively tokenizing these NEs. The recursive tokenization can be erroneous, however, the authors do not discuss the negative impact thereof.

3. Japanese Tokenizer for Business

Given the situation we described in the previous section, we aim to develop a tokenizer and dictionary for business use. In more detail, by "for business use" we mean the following:

- Available for a variety of applications and purposes
- Sufficient accuracy
- Sufficient coverage
- Long-term continuous maintenance

We release our software and language resources freely available to the public (Works Applications, 2018).

Table 1 shows the comparisons between the language resources for Japanese tokenization.

3.1. Multi-granular Tokenization

How to define the granularity of the token unit in Japanese tokenization has long been discussed. However, the suitable unit differs for each application. For example, the search systems need shorter units in order to ensure high recall rate, whereas we want longer units for semantic analysis in order to recognize the entities. Hence we manually annotate the constituting shorter units within the longer units, in order to provide tokens of different granularity for each application purpose.

We define three types of units:

- **Short**: Compatible with UniDic
- **Middle**: Similar to the "words" in general sense
- **NE**: Named entities

For each unit, we manually annotate its word structure constituted by the shorter units.

To be precise, Named Entity Recognition (NER), detecting NEs in a text, is not a subject of tokenization. However to conduct accurate NER we need some kind of lexical information, and it is more efficient to register these NEs to the dictionary and process them together at the tokenization step.

Figure 1 shows an example of multi-granular tokenization.



Figure 1: An example of multi-granular tokenization. For short unit the input is tokenized into the short parts, whereas for NE unit the result refers to the name of an existing museum.

3.2. Normalization

The Japanese language has a complicated written form system, and it does not have a rigorous orthography. This makes the notation variation (表記揺れ) a severe problem for processing Japanese text. It is essential to solve this issue in order to process text available in the real world. Table 2 shows different types of notation variation.

To deal with this problem, we manually add normalized forms for the vocabularies in our dictionary.

3.3. Continuous Maintenance

For the languages, new words appear as time passes, and the usage of existing words may change as well. Therefore it is essential to maintain the dictionary continuously.

In Japan, the maintenance of the resources developed by national universities and institutes is often discontinued after some time. On the other hand, we continuously maintain our resources in long-term as a part of the company business. NEologd also claims to continue the maintenance, however, it is created automatically or semi-automatically, therefore the part of the resource have poor quality, and it cannot deal with normalization of notation variations.

3.4. Software for Tokenization

We release a tokenizer software called *Sudachi*[9] as an OSS in order to use the language resources we have developed.

[6] http://pj.ninjal.ac.jp/corpus_center/csj/

[7] http://pj.ninjal.ac.jp/corpus_center/bccwj/

[8] https://github.com/neologd

[9] https://github.com/WorksApplications/Sudachi

	IPADIC	UniDic	NEologd	Sudachi
Multi-granularity	No	No	No	**Yes**
Named Entity	Some	Some	**Yes**	**Yes**
Normalized Form	**Yes**	**Yes**	No	**Yes**
Continuous Maintenance	No	No	**Yes**	**Yes**
Manual Check	**Yes**	**Yes**	No	**Yes**

Table 1: Comparisons between the resources for Japanese tokenization.

Type	Example
Kana Suffix Variation (送り違い)	打込む / 打ち込む
Character Type Variation (字種)	かつ丼 / カツ丼
Glyph Variation (異体字)	附属 / 付属
Misspelling (表記誤り)	シミュレーション / シュミレーション

Table 2: Types of notation variation in Japanese. Words with different notations, indicating the same meaning.

This tokenizer has functions such as tokenization in different granularity, and the normalization of notation variations.

We also design the pre-/post-processes to be plugins of the tokenizer, aiming to aggregate the knowledge of various users that were previously scattered in individual user's environment. We implement the plugins and release them to the public, so that anyone can easily conduct Japanese tokenization without having a detailed knowledge of the task.

The original version is implemented in Java. We also release the Python version called *SudachiPy*[10]. In addition to the tokenizer itself, we also develop and release a plugin[11] for Elasticsearch[12], an open source search engine.

4. Current Status

As we described in subsection 2.2., the number of language resources derived from BCCWJ is growing; Therefore we emphasize the compatibility to UniDic in order to make use of those resources. We hence develop our resources based on UniDic's tokenization dictionary, add middle and NE unit vocabularies from NEologd and other resources, then adjust the word structures and normalized forms.

4.1. Short Unit: Revising UniDic

UniDic is designed to emphasize the reproducibility of annotation, to have segmentation in a good order. This makes the annotation consistent regardless of who conducted it, however, it sometimes has unintuitive segmentation and this causes issues for practical usages.

For example, UniDic has the rule to decide segmentation according to its origin (語種) as Japanese, Chinese or Western; For Japanese origin words, it considers the word base and the suffix together as a unit, whereas for Chinese origin words they are segmented into 2 units. Table 3 shows a segmentation example where the behaviors are different even with the same suffix.

We manually modify these unintuitive segmentations for the practical purposes.

[10]https://github.com/WorksApplications/SudachiPy

[11]https://github.com/WorksApplications/elasticsearch-sudachi

[12]https://www.elastic.co/jp/products/elasticsearch

UniDic Segment	Origin (語種)
使用 / 料 (Charge)	Chinese (漢語)
為替料 (Exchange Fee)	Japanese (和語)

Table 3: An example of segmentation difference between the tokens with the same suffix in UniDic. They have the same suffix "料", however, the resulting segments are different becasuse of their origins.

4.2. NE Unit: Revising NEologd

The words in NEologd are collected automatically or semi-automatically. Therefore we screen the resource to exclude entities that are included by unclear reason, or unnecessary for our purposes. For example, the date expressions are sometimes included in NEologd, however, we would like to exclude and handle them in pre-/post-processing steps instead. We then manually add the word structure and normalized form. We also fix *kana* information if necessary, as it is automatically estimated in NEologd and it occasionally is inaccurate.

4.3. Middle Unit: Selection Policies

Middle unit is the unit that is most close to the "words" the users would expect, and it will be useful for different purposes. However, it is difficult to comprehensively define this unit.

Currently, we are investigating the definition for each category separately. If the token is a noun, we define the unit to include its prefix and suffix. On the other hand, if it is a verb we include up to the compound verb to be a middle unit.

4.4. Vocabulary Size

We currently selected 2.6 million tokens, and 1.4 millions of them are accurately given normalized forms, POS, and *kana* information. Among them, 0.8 million tokens have word structure information.

5. Future Work

The language of Computer-mediated Communication (CMC) such as in e-mails, blogs, and social networks is called Uchi-kotoba (打ち言葉, typing language). It has different characteristics compared to written and spoken languages. Uchi-kotoba has unique spellings, simplifications,

and phase vocabularies. We expect to have an even larger amount of such text in the future, and the emphasis on processing such text will increase.

It is difficult to minimize the negative effect when incorporating the NEs that may result in erroneous analysis. Transient NEs such as the titles of the movies, TV shows or the names of the political parties[13] may become the cause of erroneous analysis. We need to devise methods in order to minimize such errors, for example, by removing them after a certain period of time when the frequency of appearance has decreased.

Documents handled in the business scenes may contain noises. These noises may result from typos while inputting text, or failures of speech recognition or optical character recognition. We would like to achieve a robust tokenization that minimizes the negative impact to the surroundings of these noise sections. It is also important to develop a tokenizer with error correction features for such noises.

There are situations which we have not yet decided how to handle in our formulation. One situation is when a word has ambiguous word structure. For example, a word "暴力団員" (Yakuza, a gang member) can be formed of either "暴力 / 団員" (violence / organization member) or "暴力団 / 員" (gang / member). We want both structures in applications. Another situation is when it is not obvious how to annotate a word. For example, the compound of "輸出" (export) and "輸入" (import) is "輸出入" (export and import), where the common prefix "輸" is shared and appears only once.

6. Conclusion

In this paper, we presented a Japanese tokenizer Sudachi and its dictionary, aiming to improve the current situation of the Japanese tokenization task especially for the business application purposes.

We first described the importance of tokenization for Japanese text processing in business applications.

We then showed the problems of the current tokenization tools and language resources. Pre-/post-processing are the important parts of text processing, however, the existing tokenizers do not have these processes built into the tool, or not easily usable for the users' needs. This tends to force the users to reinvent the wheel or conduct such processes in inefficient ways. Existing language resources for Japanese tokenization are often not maintained continuously, or the token unit of the resource may not be suitable for the users' purposes.

We aim to solve such problems by developing a high-quality dictionary with multi-granular token information for different purposes, normalized form information for notation variations, long-term continuous maintenance as a part of the company business, and an accompanying tokenizer with pre-/post-processes as plugins. We make our tokenizer and language resources freely available to the public as an OSS.

We explained the current status and issues of the project.

Lastly, we mentioned various topics that need to be dealt to improve tokenization quality.

[13]Political parties in Japan are often formed for a short-term, and they are occasionally titled using common names.

7. Acknowledgments

We thank Dr. Masayuki Asahara of National Institute for Japanese Language and Linguistics and Toshinori Sato of LINE Corporation for the discussions. We also thank the Sudachi users and the anonymous reviewers for the comments.

8. Bibliographical References

Asahara, M. and Matsumoto, Y., (2003). *ipadic version 2.7.0 ユーザーズマニュアル (IPADIC version 2.7.0 Users Manual, in Japanese)*. Nara Institute of Science and Technology.

Den, Y., Ogiso, T., and Ogura, H. (2007). The development of an electronic dictionary for morphological analysis and its application to japanese corpus linguistics (in japanese). *Japanese linguistics*, 22:101–123.

Kouno, T. and Ogiso, T. (2015). Improving an electronic dictionary for morphological analysis of japanese: Use of historical period information. In *Proc. of The 9th International Conference of Asian Association of Lexicography (ASIALEX)*, pages 441–449.

Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). Applying conditional random fields to japanese morphological analysis. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 230–237.

Maekawa, K. (2008). Balanced corpus of contemporary written japanese. In *Proc. of The 6th Workshop on Asian Language Resources (ALR), The 3rd International Joint Conference on Natural Language Processing (IJCNLP)*, pages 101–102.

National Institute for Japanese Language and Linguistics, (2017). *UniDic とは? (What is UniDic?, in Japanese)*. `http://unidic.ninjal.ac.jp/about_unidic`.

Sato, T., Hashimoto, T., and Okumura, M. (2016). Operation of a word segmentation dictionary generation system called neologd (in japanese). In *Information Processing Society of Japan, Special Interest Group on Natural Language Processing (IPSJ-SIGNL)*, pages NL–229–15.

Sato, T., Hashimoto, T., and Okumura, M. (2017). Implementation of a word segmentation dictionary called mecab-ipadic-neologd and study on how to use it effectively for information retrieval (in japanese). In *Proc. of the Twenty-three Annual Meeting of the Association for Natural Language Processing*, pages NLP2017–B6–1.

Takahashi, F. and Sassano, M. (2016). 情報検索のための単語分割一貫性の定量的評価 (quantitative evaluation of word segmentation consistency for information retrieval, in japanese). In *Proc. of the Twenty-two Annual Meeting of the Association for Natural Language Processing*, pages 949–952.

9. Language Resource References

National Institute for Japanese Language and Linguistics. (2017). *UniDic*. 2.1.2, ISLRN 114-759-406-461-7.

Works Applications. (2018). *Sudachi Dictionary*. 0.1.0, ISLRN 811-243-395-600-3.

Chemical Compounds Knowledge Visualization with Natural Language Processing and Linked Data

Kazunari Tanaka[1,2] Tomoya Iwakura[1,2] Yusuke Koyanagi[1,2] Noriko Ikeda[1] Hiroyuki Shindo[2,3] Yuji Matsumoto[2,3]
FUJITSU LABORATORIES LTD.[1],
RIKEN Center for Advanced Intelligence Project[2],
NARA INSTITUTE of SCIENCE and TECHNOLOGY[3]
{tanaka.kazunari, iwakura.tomoya, koyanagi.yusuke, nona}@jp.fujitsu.com,
{shindo, matsu}@is.naist.jp

Abstract
This paper proposes a visualization system for chemical compounds. New chemical compounds are being produced by every moment and registration of chemical compounds to databases strongly depends on human labor. Our system uses Natural Language Processing technologies for extracting information of chemical compounds from text and for storing the extracted results as Linked Data (LD). By combining the extracted results with LD-based existing chemical compound knowledge, our system provides visualization of chemical compound information such as integrated view of several databases and chemical compounds that have similar structures.

Keywords: Information extraction, chemical compounds

1. INTRODUCTION

Knowledge of chemical compounds has great value for developing new materials, new drugs, and so on. Therefore, databases of chemical compounds are being created. For example, CAS [1], one of the largest databases, includes information on over 100 million chemical compounds. However, the creation of such databases strongly depends on manual labor since chemical compounds are being produced at every moment. In addition, the database creation mainly focuses on English text. Therefore, in other words, chemical compound information other than English is not good enough to be available. For example, although Japan has one of the largest chemical industries and has large chemical compound information written in Japanese text documents, such information is not exploited well so far.

We propose a visualization system based on chemical compound extraction results with Japanese Natural Language Processing and structured databases represented as Linked Data (LD).

Figure 1 shows an overview of our system. First, chemical compound names in text are recognized. Then, aliases of chemical compound names are identified. The extraction results and existing chemical compound databases are represented as LD. By combining these LD-based chemical compound knowledge, our system provides different views of chemical compounds.

2. Chemical Compound Information Extraction

Unknown chemical compounds usually first appear in unstructured text such as scientific papers and patents. In order to extract chemical compound names from text, we use a Japanese Named Entity (NE) recognition method [2]. An NE recognizer is trained with a distant supervision [3] using a dictionary of chemical compound names as a lexical resource. The dictionary is compiled from Nikkaji [4].

The dictionary is used to recognize chemical compound names in text. The chemical compound names in text recognized with the dictionary are used as positive samples for training. Negative samples are documents that would be

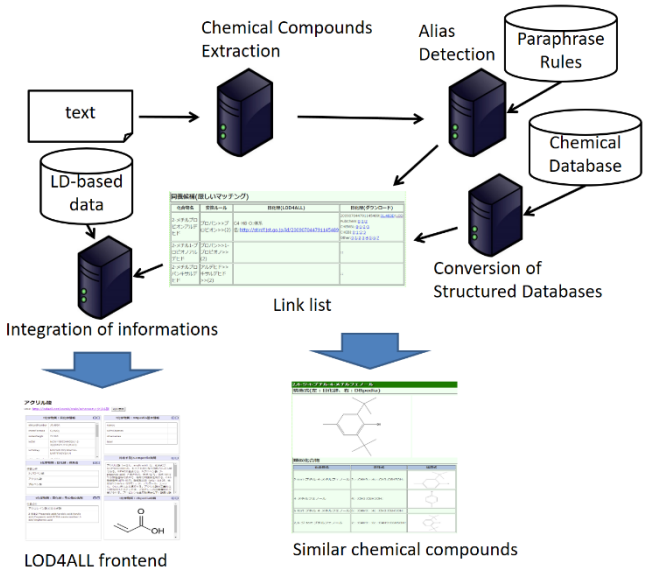


Figure 1: An overview of our system.

irreverent with chemical domains such as politics and sports. Then, the automatically created training data was used for training an NE recognizer.

By using a chemical compound name recognizer based on an NE recognition method, we expect to obtain new chemical compound names from text with contextual information like “boiling point” and “fusing points” or, prominent words consisting of chemical compounds like “acid” and “methylphenol”.

In addition, we extract usage of chemical compounds from text by a rule-base method. Some of the examples are chemical compounds used as a plasticizer and a surfactant. Another example is usage of chemical compounds as replacement candidates. For example, ‘Dioctyl phthalate’ is used as an alternative of ‘Diisononyl phthalate’ for a plasticizer.

3. Identification of Aliases of Chemical Compound Names

Aliases of chemical compound names are possibly used instead of their full spellings [5]. For example, information of chemical compounds is included as different names in different databases and text such as Nikkaji [4], PubChem [6], ChEMBL [7], CPCat [9], DBPedia [11], patents and scientific papers.

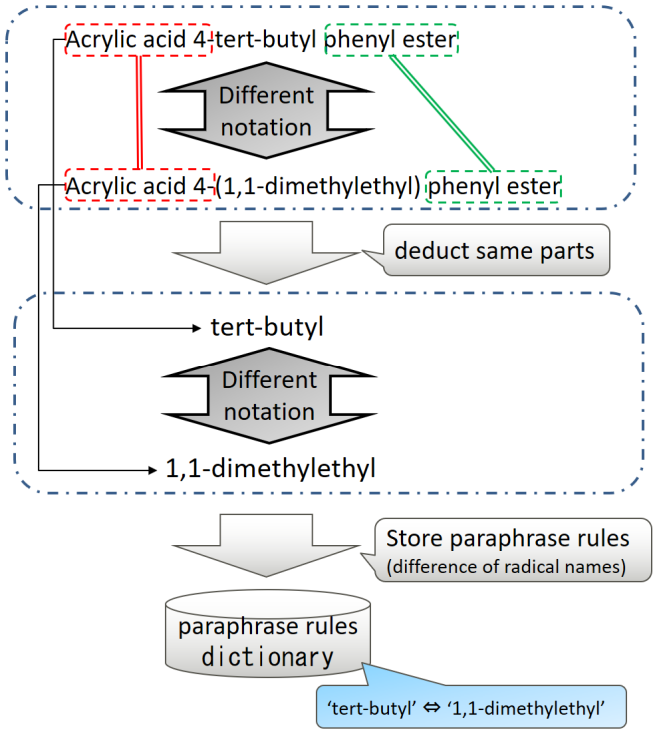


Figure 2: An extraction of a paraphrase rule.

To identify aliases of chemical compounds, we compiled paraphrasing rules from Nikkaji [4]. In order to obtain paraphrasing rules, we split chemical compound names into constituents of structures based on IUPAC nomenclature [5]. Then we extract paraphrase rules by comparing parts of the structure of a chemical compound name with its alias.

We describe a method to get paraphrase rules by using 'Acrylic acid 4 - tert - butylphenyl ester' and its alias 'Acrylic acid 4 - (1,1 - dimethylethyl)phenyl ester'.

Figure 2 shows an example of an extraction of a paraphrase rule from the above two chemical compound names. At first, we split 'Acrylic acid 4 - tert - butylphenyl ester' into 'Acrylic acid', 'tert - butyl' and 'phenyl ester'. In a similar way, 'Acrylic acid 4 - (1,1 - dimethylethyl)phenyl ester' is split into 'Acrylic acid', '(1,1 - dimethylethyl)' and 'phenyl ester'.

Next, we deduct the common parts of the both chemical compound names. Finally, we extract the pair of 'tert - butyl' and '1,1 - dimethylethyl' as a paraphrase rule. We use a dictionary to split chemical compound names into parts of structures.

We expect that a possibility to access useful information would increase by using these rules.

For example, '2 - (p - Tolyl)ethanol' has not been recorded on Nikkaji, However, '2 - (4 - Methylphenyl)ethanol' that is created by paraphrase rule has been recorded.



Figure 4: Integration different data by LD-based information.

4. Visualization of Chemical Compounds

4.1 Highlighting Chemical Compounds

Many chemical patents include sequences of chemical compound names. It is difficult for us to recognize differences between chemical compounds in patent documents. Therefore, a high skill and a large amount of time are required to readers. In order to alleviate such problem, chemical compound names are highlighted on the screen in our system as shown in Figure 3. In addition, each chemical compound has links to the visualization of the following sections. Therefore, users can access additional information by clicking the links in a current text.

4.2 Integrated Data Representation

Figure 4 is an example of an integrated visualization of LD-based knowledge of acrylic acid. The visualization includes information of databases such as figures of chemical compounds and information extracted from text as described in Section 2.

The visualization is realized as follows. At first, we extract chemical compound names from documents and create different notations by using paraphrase rules. The processed information is embedded as links in text. Then, if a link is clicked, a search is conducted with extracted names and created notations with paraphrasing. Finally, with the search results, an integrated data representation as shown in Figure 4 is generated.

【0074】
中でも、1，1－ビス（4－ヒドロキシフェニル）シクロペンタン≪16≫、1，1－ビス（3－メチル－4－ヒドロキシフェニル）シクロペンタン≪53≫、1，1－ビス（4－ヒドロキシフェニル）シクロヘキサン≪15≫、1，1－ビス（4－ヒドロキシフェニル）－3，3，5－トリメチルシクロヘキサン≪20≫、2，2－ビス（4－ヒドロキシフェニル）アダマンタン≪18≫、2，2－ビス（3－メチル－4－ヒドロキシフェニル）アダマンタン≪62≫、1，3－ビス（4－ヒドロキシフェニル）アダマンタン≪19≫、1，3－ビス（3－メチル－4－ヒドロキシフェニル）アダマンタン≪63≫、1，1－ビス（3－メチル－4－ヒドロキシフェニル）シクロヘキサン≪52≫、1，1－ビス（4－ヒドロキシフェニル）シクロドデカン≪17≫、1，1－ビス（3－メチル－4－ヒドロキシフェニル）シクロドデカン≪76≫が溶解性に優れるPC共重合体を与えるという点で好ましい。

Figure 3: An example of many chemical compounds written in a patent document.

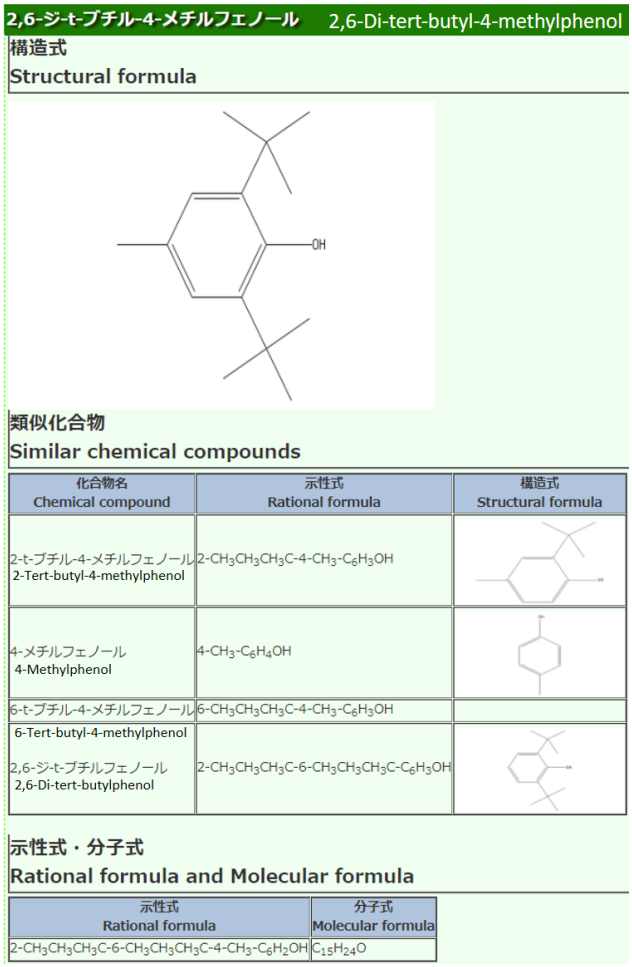


Figure 5: An example of Visualization of Relevant Chemical Compounds.

4.3 Relevant Chemical Compounds

If we can identify the same chemical compounds with different notations by using paraphrasing rules, we can get information from different databases that register the same chemical compounds with different notations. However, the new chemical compounds that have not been registered in databases cannot be found.

In order to help users to understand new chemical compounds, we present additional information about chemical compounds that have similar structures with a given chemical compound.

Figure 5 depicts a set of similar chemical compounds of “2,6-Di-tert-butyl-4-methylphenol”. This chemical compounds are recorded in the database and overall structures are displayed. In addition, chemical compounds that have the same parts of structures are displayed as similar chemical compounds.

Table 1: Evaluation of effectiveness of paraprasing

Number of chemical compound names	36
recorded on Nikkaji	12
covered by using paraphrase rule	4
similar chemical compounds	2

By displaying similar chemical compounds, users can infer the structure of a given chemical compound even if it is not registered in databases.

Chemical compounds consist of substituents, which are constituents of structures. Therefore, we can know the structures of chemical compounds by splitting a chemical compound name into substituent names. Furthermore, we can create similar chemical compounds of a given chemical compound by deducting some substituents from the given chemical compound name. Chemical compound names created by deducting substituents of a chemical compound may be recorded in chemical databases. As a result, we can get clues about the structure information of a new chemical compound that have not been registered in databases.

4.4 Analysis of Chemical Compound Names

Figure 6 is an example of a table representation. Eleven chemical compounds extracted from Figure 3 are listed. A combination of some substituents and some cores creates these variations.

We analyze a hierarchically connection of constituents of structures. In addition, we convert relationships between cores and substituents of chemical compounds into a table representation.

This example shows that there are two patterns in their chemical compounds. And they contain one exception. By displaying difference of chemical compounds, users can realize an overview of them.

5. Evaluation

We evaluated our paraphrasing method, which is one of the key components of our system, with 36 chemical compounds written in the patent document (P2014-263456, paragraph [0017]). The evaluation was done whether the paraphrase contributed to discovery of the same and relevant chemical compounds in databases.

Table 1 shows the experimental result. Twelve chemical compounds out of 36 were recorded in Nikkaji. Four chemical compounds out of 24 were converted into recorded notations by paraphrase rules. Two chemical compounds out of the remaining got similar chemical compounds.

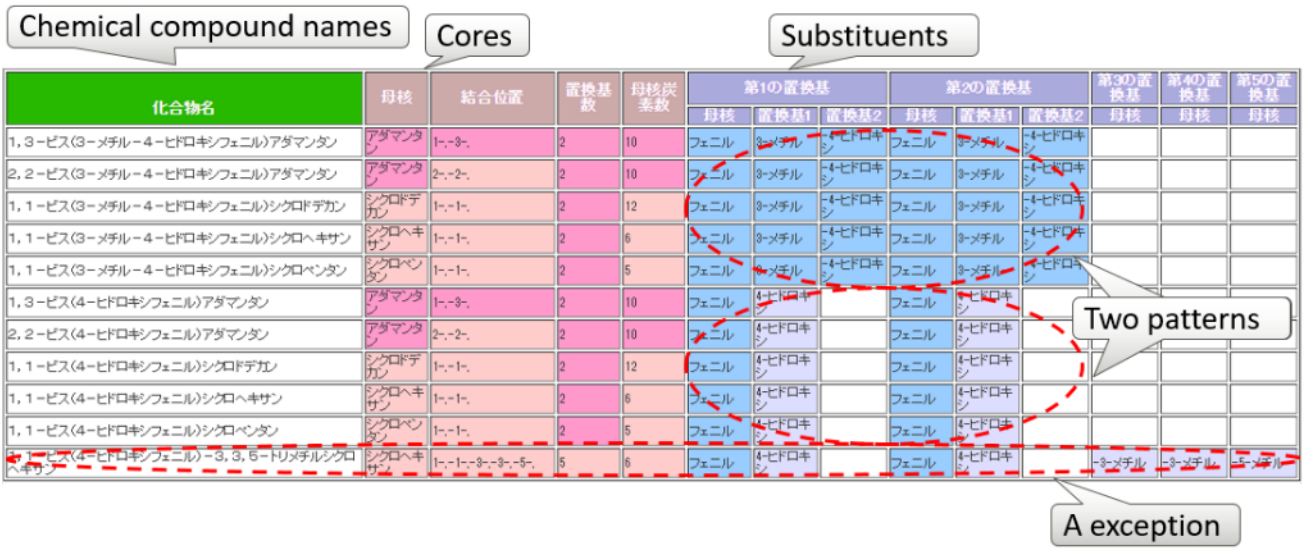


Figure 6: An example of table of chemical compounds.

From these results, we see difficulty to cover all chemical compounds by only databases. "4,4'-dihydroxy-3,3'-dimethylphenyl ether" and "4,4'-dihydroxy-3,3'-dimethyldiphenylsulfide" were not covered. However, the constituents of structures, which are 'hydroxy', 'methyl', 'diphenyl ether' and 'diphenylsulfide', had been recorded in chemical databases. In this paper, we created only similar chemical compounds, created by deducting one substituent from the chemical compounds. Therefore, there may be many relevant chemical compounds that we can not find.

On the other hand, even if similar chemical compounds become different from original chemical compounds, users are still difficult to estimate overall structures. Therefore, we provide chemical formulas (Rational formula, Molecular formula) like Figure 5 to estimate overall structure.

6. Conclusion

This paper proposes a visualization system for chemical compound information extracted from Japanese texts and chemical compound databases. This system enables users to get information of chemical compounds not only from existing databases but also from text.

In the future work, we would like to extend our system to languages other than Japanese.

REFERENCES

1. CAS web page: https://www.cas.org/index
2. Tomoya Iwakura. A Named Entity Recognition Method using Rules Acquired from Unlabeled Data. Proc. of RANLP'11. Pp. 170—177.
3. Mintz, Mike and Bills, Steven and Snow, Rion and Jurafsky, Dan. Distant Supervision for Relation Extraction without Labeled Data. Proc. Of ACL'09. pp. 1003—1011. 2009.
4. Nikkaji: http://dbarchive.biosciencedbc.jp/jp/nikkaji/desc.html
5. IUPAC: https://iupac.org/what-we-do/nomenclature/
6. The PubChem Project: https://pubchem.ncbi.nlm.nih.gov/
7. ChEMBL: https://www.ebi.ac.uk/chembl/
8. ChemSpider: http://www.chemspider.com/
9. Chemical and Product Categories (CPCat): https://www.epa.gov/chemical-research/chemical-and-product-categories-cpcat
10. LOD4ALL frontend: http://lod4all.net/frontend/
11. DBpedia: http://wiki.dbpedia.org/

Using Discourse Information for Education with a Spanish-Chinese Parallel Corpus

Shuyuan Cao[▲*], **Harritxu Gete**[*]
[▲]Universitat Pompeu Fabra (UPF)
[*]University of Basque Country (UPV/EHU)
shuyuan.cao@hotmail, harritxu.gete@gmail.com

Abstract

Nowadays, with the fruitful achievements in Natural Language Processing (NLP) studies, the concern of using NLP technologies for education has called much attention. As two of the most spoken languages in the world, Spanish and Chinese occupy important positions in both NLP studies and bilingual education. In this paper, we present a Spanish-Chinese parallel corpus with annotated discourse information that aims to serve for bilingual language education. The theoretical framework of this work is Rhetorical Structure Theory (RST). The corpus is composed of 100 Spanish-Chinese parallel texts, and all the discourse markers (DM) have been annotated to form the education source. With pedagogical aim, we also present two programs that generate automatic exercises for both Spanish and Chinese students using our corpus. The reliability of this work has been evaluated using Kappa coefficient.

Keywords: discourse analysis, education, corpus, bilingual language learning

1. Introduction

Using natural language processing (NLP) for educational applications starts from the early history (Burstein, 2009). Different NLP studies make a great advance in different educational areas, for instance, translation studies, text retrieval, text mining or speech recognition.

Among different NLP studies, the emphasis on the idea that discourse information may be useful for Natural Language Processing (NLP) has become increasingly popular. Discourse analysis is an unsolved problem in this field, although discourse information is crucial for many NLP tasks (Zhou et al., 2014).

As two of the most spoken languages in the world, Spanish and Chinese occupy important positions in NLP development. Due to the great linguistic distance that between this pair of languages, the number of differences in their discourse structure is also considerable. The following examples show some of the discourse differences between Spanish and Chinese.

Ex.1:

1.1 Sp: Aunque aún no contamos con resultados, intuimos que el modelo será más amplio que el del sintagma nominal.

[Aunque aún no contamos con resultados,]$Unit_1$ [intuimos que el modelo será más amplio que el del sintagma nominal.]$Unit_2$

[DM[1] still no get still no get results,]$Unit_1$ [we consider that the model will more extensive than the sentence group nominal[2].]$Unit_2$

1.2 Sp: Intuimos que el modelo será más amplio que el del sintagma nominal, aunque aún no contamos con resultados.

[Intuimos que el modelo será más amplio que el del sintagma nominal,]$Unit_1$ [aunque aún no contamos con resultados.]$Unit_2$

[We consider that the model will more extensive than the sentence group nominal.]$Unit_1$ [DM still no get still no get results]$Unit_2$

1.3 Ch: 尽管还没有取得最终结果，但是我们认为该模型已囊括了语段模型涉及的内容。

[尽管还没有取得最终结果，]$Unit_1$ [但是我们认为该模型已囊括了语段模型涉及的内容。]$Unit_2$

[DM1 still no get results,]$Unit_1$ [DM2 we consider that the model contains the sentence group nominal.]$Unit_2$

1.4 Eng: Although we haven't got the results yet, we consider that the model will be more extensive than the nominal sentence group.

In Example 1, we can see that the Spanish passage and the Chinese one have a similar discourse structure. Both passages start with a discourse marker in the first unit. However, the discourse markers are used differently to show the same meaning in both languages. In Chinese, it is mandatory to include two DMs: the first one is "*jinguan*" (尽管), and it is located at the beginning of the first unit, and the other marker is "*danshi*" (但是), which is placed at the beginning of the second unit. These two discourse markers are equivalent to the English discourse marker 'although'. By contrast, in Spanish, just one DM "*aunque*" is needed to express the same meaning. Although in 1.1 it is being used at the beginning of the first unit, as we can see in 1.2, the order of the discourse units in this Spanish passage can be changed and it makes sense syntactically, so the DM can appear both at the beginning of the first or the second unit. By contrast, the order cannot be changed in the Chinese passage, because neither syntactically nor grammatically makes sense.

Ex.2:

2.1 Sp: Si optas por un aprendizaje lo más parecido posible a la inmersión, y necesitas mejorar tu nivel de español rápidamente, los cursos intensivos son una buena opción.

[1] DM means discourse marker. We will give the specific definition of discourse marker in the methodology section.
[2] In this work, for all the presented examples, we will give an English literature translation for each example.

[Si optas por un aprendizaje lo más parecido posible a la inmersión,]$Unit_1$ [y necesitas mejorar tu nivel de español rápidamente,]$Unit_2$ [los cursos intensivos son una buena opción.]$Unit_2$

[DM1 you opt for a learning as more similar possible to immersion,]$Unit_1$ [DM2 you need to improve your level of Spanish quickly,]$Unit_2$ [the courses intensive are a good option.]$Unit_3$

2.2 Ch: 若您希望进行全面集中的语言学习，或者您希望短时间内提高您的语言水平，紧凑课程是一个很好的选择。

[若您希望进行全面集中的语言学习，]$Unit_1$ [或者您希望短时间内提高您的语言水平，]$Unit_2$ [紧凑课程是一个很好的选择。]$Unit_3$

[DM1 you want to focus on completely intensive of language learning,]$Unit_1$ [DM2 you wish in short time to improve your language level,]$Unit_2$ [the intensive courses are a good option.]$Unit_3$

2.3 Eng: If you want to focus on language intensive learning or if you want to improve your language skills in a short time, a compact program is a good choice.

In Example 2, we can see that there are two DMs in the Spanish passage, one is "*si*" ('if' in English) and another one is "*y*" ('and' in English). The DM "*y*" represents a LIST relation between the first unit and the second unit. Meanwhile, the DM "*si*" connects the first two units ($Unit_{1-2}$) and the third unit ($Unit_3$) as a CONDITION relation. In the Chinese passage, there are also two discourse markers. One of them is "*ruo*" (若), which means 'if' in English; the other one is "*huozhe*" (或者) and corresponds to 'or' in English. The DM "*huozhe*" (或者) represents a DISJUNCTION relation between the first and second unit. Same as the Spanish passage, The DM "*ruo*" (若) also gives a CONDITION relation between the first two units ($Unit_{1-2}$) and the third unit ($Unit_3$).

Although this is a parallel example, we can see that the discourse relation between the first unit and the second unit are different in the Spanish and Chinese passage. In the Spanish passage, the relation within the first two units is LIST while in the Chinese passage the first two units hold a DISJUNCTION relation. This is because of the translation strategy. The discourse relations are different, but the main idea in both passages is the same.

These two examples show some of the differences in the discourse structure of Spanish and Chinese. Since this pair of languages have considerable differences in this aspect, a comparative discourse analysis between Spanish and Chinese is essential for language learning. Therefore, this work aims to give a discourse analysis with a Spanish-Chinese parallel corpus. This analysis could be beneficial for Spanish-Chinese language learning education, from discourse level point of view. In addition, with educational purposes, we have also developed two programs for automatic exercise generation. The generated exercises can be used by Spanish and Chinese language learning students in order to practice the usage of DMs in these languages.

In the second section, we introduce the theoretical framework of the work. In the third section, we talk about the state of the art. In the fourth section, we present the methodology. Firstly, we explain how we create and annotate the corpus and then, we explicate the methodology used for the automatic exercise generation. In the fifth section, we describe the evaluation method and we show the obtained results. Finally, in the last section, we explain our conclusions and the possible future work.

2. Theoretical Framework

Different theories and approaches have been applied to discourse analysis, the most used are: a) The Rhetorical Structure Theory (RST) (Mann and Thompson, 1988), b) the Penn Discourse Tree-Bank (PDTB) (Miltsakaki et al., 2004; Prasad et al., 2004) and c) the Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003).

RST addresses both hierarchical and relational aspects of text structures for discourse analysis. Elementary Discourse Units (EDUs) (Marcu, 2000) and coherence relations are established in RST. Relations are recursive in RST and are defined to hold between EDUs; the EDUs can be Nuclei or Satellites. Satellites offer additional information about nuclei. The EDUs can be linked among them holding a nucleus-satellite (e.g. CAUSE, JUSTIFY, EVIDENCE, CONCESSION) function or a multinuclear (e.g. CONJUNCTION, LIST, SEQUENCE) function and as relations are recursive, all the discourse units of the text have a function in a treelike structure, if and only if the text is coherent.

PDTB is a large-scale annotation project and extends the work of the Penn Treebank (PTB) (Marcus, Santorini and Marcinkiewicz, 1993) and Propbank (Kingsbury and Palmer, 2002). Grounded on a lexicalized approach to discourse, the discourse connectives and their arguments have been annotated in PDTB (for instance, contingency, temporal, expansion). Sense labels for each relation with a hierarchical classification scheme. The senses annotation works detect the polysemy of connectives and make the PDTB helpful for sense disambiguation tasks (Miltsakaki et al., 2005).

SDRT explores the logical form between discourse interpretation and discourse coherence by using rhetorical relations to model the semantics/pragmatics of a text. In SDRT, the logic of information content is used to represent the logical forms of discourse and the glue logic is being applied to construct logic forms. The logic forms represent the syntax and dynamic semantics of a language. SDRT can model the complexity of the interaction between semantics and pragmatics under some discourse relations (e.g. CAUSE, EXPLANATION, CONTRAST).

RST has been selected as the theoretical framework of this work. Comparing to PDTB and SDRT, RST focuses on the hierarchical structure of a whole text, where discourse relations can be annotated within a sentence (intra-sentence style) and between sentences (inter-sentence style). The intra-sentence annotation and

inter-sentence annotation styles help to inform how discourse elements are being expressed in a language and translation strategies (if there are) can be detected in different levels of an RS-tree (da Cunha and Iruskieta, 2010; Iruskieta, da Cunha and Taboada 2015).

3. State of the art

3.1 RST Corpus and Education

Studies addressing RST for education have been applied to different language pairs. For instance, for English and Chinese, by annotating Chinese students' and native speakers' compositions of the same topic under RST, Zhang (2010) describes and compares the rhetorical structure diagrams of these compositions from the perspective of amount, frequency and distribution of each relation to help teachers to explore the deficiencies of Chinese students' compositions. By using news texts on *China Daily* and *The New York Times*, Fang (2008) explores the discourse features of English that expressed by Chinese native speakers by means of RST, and the study helps English foreign language learners acquire a better understanding of Chinese style English. In order to help Chinese students' argumentative writing in English, Li and Liao (2015) take RST as the theoretical framework to explore the different features with 60 English essays written by Chinese students. Beside of the English-Chinese language pair, there is one work focuses on the language education between Chinese and Korean and takes RST as its framework. Liang and Yang (2016) use the spoken data of Korean students and Chinese native speakers to reveal the differences in their use of causal and transitional markers, and analyse the typical errors under RST. Finally, they give some suggestions for Korean-Chinese speaking teaching.

Regarding the language pair Spanish-Chinese, few works exist and can be useful for Spanish-Chinese education. Yao (2008) uses film dialogues to elaborate an annotated corpus, and compares the Chinese and Spanish discourse markers in order to give some suggestions for teaching and learning Spanish and Chinese. In this work, Yao does not use a particularly detailed framework and only offers a comparative analysis of Spanish and Chinese discourse markers, followed by his conclusions. Taking different newspapers and books as the research corpus, Chien (2012) compares the Spanish and Chinese conditional discourse markers to give some conclusions of the conditional discourse marker for foreign language teaching between Spanish and Chinese. Wang (2013) uses Pedro Almodóvar's films *La mala educación* and *Volver* as the corpus to analyze how the subtitled Spanish discourse markers can be translated into Chinese, so as to make a guideline for translation education between the language pair. However, none of these works use RST as its theoretical framework. The only work for the Spanish-Chinese education under RST is the work of Cao, da Cunha and Bel (2015). They explore the different Chinese translations of the Spanish DM "*aunque*" ('however/but') in the UN corpus to indicate how to translate the Spanish DM "*aunque*" into Chinese during the foreign language study. Yet, this work only focuses on the discourse structure of the single sentences instead of the whole texts.

3.2 Exercise Generation and Multi Choice Question

Regarding the exercise generation aspect, some successful studies have been applied to education filed by using different approaches. For example, under statistical ranking module, address the challenge of automatically generating questions from reading materials for educational practise and assessment; Heilman and Smith (2010) give a rule-based system to rank the output to give the *wh*- question. Under the situation module, Chen, Aist and Mostow (2009) test the generality of their question generation approach by extending the approach to informational text. Moreover, discourse information has also been used in their study. Concept map is another approach can be used for question generation. Olney, Graesser and Person (2012) erase the gap between psychological theories of question asking and computational models of question generation by computing conceptual graphs.

To our knowledge, our work is the first one that focuses on the discourse structure of the whole texts under RST for Spanish-Chinese language education, and contains the exercise generation function.

4. Methodology

We carry out different steps for this study and the following subsections details our methodology.

4.1 Research Corpus

Cao, da Cunha and Iruskieta (2017) indicate that there is not an adequate Spanish-Chinese parallel corpus for discourse analysis under RST; therefore, we construct a new Spanish-Chinese parallel corpus. We have determined the main characteristics that the texts should include. These characteristics are the following: (a) Texts with an equal translation process. This means texts originally written in Spanish and translated into Chinese by natives or vice versa. (b) Texts with different sizes: texts between 90 and 1,500 words. This means that they are texts with a complex discourse structure. (c) Specialized texts. This also means that they can have a complex discourse structure. (d) Texts from different domains (to obtain a heterogeneous corpus). (e) Texts from different genres (to obtain a heterogeneous corpus). (f) Texts from different sources (to obtain a heterogeneous corpus). (d) Texts from different authors (to avoid bias).

Secondly, we have searched for texts with these characteristics in different sources. To obtain a high translation quality and various rhetorical structures (that is, coherence structure) in our corpus, we decided to use Spanish texts and their translations into Chinese, done by Chinese translators. In order to confirm that all the texts fulfilled this translation process, it was necessary to

contact with the people in charge of the organizations that had been published the source documents and their translations. Due to the limitation of the available sources and the specific characteristics that we have determined, the amounts of texts that correspond with the required translation process are few. Finally, 50 Spanish texts and their parallel Chinese texts have been selected for our study. The longest text includes 1,201 words and the shortest text contains 91 words.

The original sources of these texts are: (a) International Conference about Terminology (1997), (b) Shanghai Miguel Cervantes Library, (c) Chamber of Commerce and Investment of China in Spain, (d) Spain Embassy in Beijing, (e) Spain-China Council Foundation, (f) Confucius Institute Foundation in Barcelona, (g) Beijing Cervantes Institute and (h) Granada Confucius Institute. Moreover, in order to guarantee the representativeness of our corpus, we have selected different types of texts from several domains. The genres of the texts are four: (a) abstracts of research papers, (b) news, (c) advertisements and (d) announcements.

4.2 Segmentation Annotation

In this work, we first segment the whole corpus with RSTTool (O'Donell, 2000) manually. We adopt the segmentation criteria by Cao et al. (2017), which can be applied to both Spanish and Chinese. The segmentation criteria are the following:

• Paragraphs and line breaks. A line break will be taken as an independent EDU to segment the titles (and subtitles).

• Sentences and periods. A period will be taken as an independent EDU.

• Question mark and exclamation mark. Both marks are signals of a sentence boundary.

• Other EDUs should have an adjunct verb phrase. This is a basic segmentation criterion and all other following segmentation criteria should follow this rule.

• DM, comma and adjunct verb phrase. If there is a DM at the beginning of the sentence and the sentence is being divided into two parts by a comma. In addition, if a DM is after a comma and the EDU has a verb, we also segment.

• Semicolon plus adjunct verb phrase.

• Parenthetical and dash. Only when the parented unit does not modify noun neither adjective, it is an independent segment; if within the parenthetical unit there are coordinated parts, we also segment the coordinated parts.

• Coordination and ellipsis with verbs. Coordinated clauses with verbs represent the independent EDUs, including where the subject is eliminated in the following EDUs.

For those non-EDU segmentation criteria are the following:

• Relative, modifying and appositive clauses. Relative clauses, clauses that modifies a noun or adjective, or appositive clauses are not considered as EDUs.

• Reported speech. Reported speech cannot be considered as an independent EDU.

• Truncated EDUs. For the cases of truncated EDUs, we use the non-relation label of Same-unit (Carlson, Marcu and Okurowski, 2003).

The corpus is accessible to the academic community. More detailed information of the corpus can be consulted at: `http://ixa2.si.ehu.es/rst/zh/`, including the segmentation examples of each text in the corpus.

4.3 Discourse Markers Annotation

Schiffrin (2001: 54) indicates: "Discourse markers (DMs) involve linguistic items that in cognitive, expressive, social and textual domains." Also, Portolés (2001) explains that DMs are invariable linguistic units that depend on the following aspects: (a) distinct morpho-syntactic properties, (b) semantics and pragmatics and (c) inferences made in the communication. Eckle-Kohler, Kluge and Gurevych (2015) give a more specific definition of DMs from the textual level that DMs are used to signal discourse relations in a text segment. In our study, we follow the definition of Eckle-Kohler, Kluge and Gurevych (2015), which we think is more appropriate for our study. Because the study analyses the language pair Spanish-Chinese from discourse level.

We have categorized different types of DMs as following show[3]:

➢ N-S type

• Antithesis

Nuclear: The author favors the idea.
Satellite: The author disfavors the idea.
Spanish DM(s): aunque; por el contrario; sino
Chinese DM(s): 但是

• Cause

Nuclear: A situation.
Satellite: Another situation that causes that one.
Spanish DM(s): como; debido a; ya que
Chinese DM(s): 因为; 由于

• Circumstance

Nuclear: The text shows the ideas or the events that occur in the interpretive text.
Satellite: An interpretive context of situation or time.
Spanish DM(s): cuando
Chinese DM(s): 作为; 如同

• Concession

Nuclear: A situation confirmed by the author.
Satellite: Another situation inconsistent but also affirmed by the author.
Spanish DM(s): pero; sino que; si bien
Chinese DM(s): 尽管; 然而

• Condition

Nuclear: Action or situation whose occurrence results

[3] We have annotated all the DMs in the research corpus, for each text, we have annotated the DMs within a sentence and between the sentences. Due to the translation strategies, not all the discourse relations contain the DMs for both languages. We use "/" to indicate the cases that donot have the DMs in the corpus.

from the occurrence of the conditioning situation.
Satellite: A condition situation.
Spanish DM(s): si
Chinese DM(s): 若; 如果

- Elaboration

Nuclear: The basic information.
Satellite: Additional information of the basic information.
Spanish DM(s): además; además de;
Chinese DM(s): 此外; 另外

- Evidence

Nuclear: A claim.
Satellite: Information that increases the reader's belief in the claim.
Spanish DM(s): de acuerdo a; de acuerdo con; de ahí; tal y como
Chinese DM(s): 比如

- Interpretation

Nuclear: A situation.
Satellite: An interpretation of the situation.
Spanish DM(s): en concreto
Chinese DM(s): /

- Purpose

Nuclear: An intended situation
Satellite: The intent behind the situation.
Spanish DM(s): a fin de; con afán de; con la movilidad; con el objetivo de; con este fin; con tal fin; de manera que; para; para ello
Chinese DM(s): 以便; 旨在; 为了

- Restatement

Nuclear: A situation.
Satellite: A re-expression of the situation.
Spanish DM(s): es decir
Chinese DM(s): 即

- Result

Nuclear: A situation.
Satellite: Another situation which is caused by that one.
Spanish DM(s): en consecuencia; de manera que; por consiguiente
Chinese DM(s): 于是; 因此

- Summary

Nuclear: A text.
Satellite: Summary of the text.
Spanish DM(s): en resumen
Chinese DM(s): 总之; 总而言之

➢ N-N type

- Conjunction

Nuclear: A situation or an action.
Nuclear: Another situation or another action that happens at the same time.
Spanish DM(s): al mismo tiempo
Chinese DM(s): 同时; 与此同时

- Contrast

Nuclear: One alternate.
Nuclear: The other alternate.
Spanish DM(s): por el contrario
Chinese DM(s): 而; 相反

- Disjunction

Nuclear: An alternative.
Nuclear: Another alternative.
Spanish DM(s): o
Chinese DM(s): 或; 或是; 或者; 亦或

- List

Nuclear: An item.
Nuclear: The next item
Spanish DM(s): e; ni; y; no solo; por un lado; por otro lado; sino también; tanto como
Chinese DM(s): 并; 并且; 和; 一方面..另一方面; 及; 以及; 还; 不仅(仅); 也; 既不…也不; 同样也

- Sequence

Nuclear: An item.
Nuclear: A next item.
Spanish DM(s): a continuación; antes de; en primer lugar; en tercer lugar; por último; seguidamente; tras
Chinese DM(s): 首先; 接下来; 紧接着

4.4 Exercise Elaboration

4.4.1. Exercise for L2 Spanish Learner

The exercises to practice Spanish language consist of different texts with some blanks within them. After each text, there are multi-choice answers for the user, who can choose between several DMs. These exercises have been generated automatically by removing the annotated DMs from the texts of the corpus. The distractors are DMs that can be used in the same context as the correct answer. Apart from the automatic generation of the exercises, the system can also grade the answers of the user and it gives the correct ones.

We have used the Python programming language to generate the texts automatically by removing the annotated DMs.

First of all, we have annotated all the Spanish DMs in the research corpus. Secondly, we have made a program to generate the Spanish exercises one by one automatically (an exercise is created from each text). The following steps have been carried out to make the program: (a) We have elaborated a list of the DMs we want to remove. This list is created with all the annotated DMs that appear in the texts we are using to develop the program. (b) With a simple program developed with Python, we remove the DMs following this rules: some of the DMs of the list have to be removed always, other ones only if they appear at the beginning of the sentence and finally, there are some DMs that we have to be removed only if in the same sentence appears another specific word (this is the case of the composed DMs). (c) Finally, to select the possible answers of the exercises, we have created 7 groups of DMs depending on their discourse meaning. When a DM is removed, the distractors of the exercise are selected from those in the same group. However, within each group, the DMs are grouped if it is almost impossible to distinguish between them. In this case, one cannot be used as a distractor of the other.

Thirdly, we have made another program to grade the answers.

4.4.2. Exercise for L2 Chinese Learner

Similarly, we have made a small program to take out all the discourse markers in the Chinese texts. However, the exercise design is different from Spanish language exercises. In this case, instead of giving different options for each blank, the system gives all the erased DMs at the end of the text, as choices for the Chinese language student.

Firstly, we have annotated all the DMs in each Chinese text and then, we have designed a simple program to erase from the texts all the DMs in the list.

The reason to make two different designs for Spanish and Chinese texts is because, although the texts are parallel, comparing with the Spanish texts, the Chinese texts are more difficult to understand because of the different meanings but the same word (including some annotated DMs[4]). Therefore, we consider that, for a Chinese text, it is better to remove all the DMs and mix the correct answers to let the users to choose so that they can understand the text better by filling the DMs.

5. Evaluation

In this work, we use the Cohen's Kappa coefficient to evaluate the correctness of our automatic exercise generation program. Previous works use Kappa to measure the annotation agreement for RST studies (Iruskieta, Diaz de Ilarraza and Lersundi, 2015; Cao et al., 2017), Kappa gives the agreement of annotation as follows:

$$K = \frac{P(A) - P(E)}{1 - P(E)}$$

where P(A) represents the actual observed agreement, and P(E) represents chance agreement.

We have developed our program using only a part of the annotated corpus (60% of the corpus, 30 Spanish texts and their parallel Chinese texts). Then, the program has been applied to the rest of the corpus in order to measure its accuracy, so the Kappa coefficient evaluates the correctness of our program deleting the DMs in those texts.

Table 1 shows the K results of the 20 tested Spanish texts and their parallel Chinese texts[5].

Text Name	Spanish	Chinese
TERM18	0.963	0.885
TERM19	0.878	0.857
TERM23	0.975	0.877
TERM25	0.866	0.657
TERM30	0.950	0.746
TERM31	0.971	0.891
TERM34	0.914	0.861
BMCS3	1	0.795
BMCS5	1	0.962
CCICE1	0.797	0.538
CCICE4	0.931	0.921
EEP3	1	0.873
EEP4	0.912	0.955
FICB3	1	0.907
FICB4	0.886	0.662
FCEC2	0.905	0.866
ICP2	0.927	0.806
ICP6	0.963	0.897
ICP7	1	0.822
ICEG1	0.973	0.907

Table 1: Program accuracy of the 40 tested Spanish-Chinese parallel texts

From table 1, we can see that our program works quite well for all the Spanish texts, among the 20 tested Spanish texts, 5 of them have 100% accuracy. Other texts maintain the accuracy from 0.86 to 1 except the text CCICE1 (0.797). After analysing the outputs, we find that the common limitation for the Spanish texts is that, not all the annotated Spanish DMs have been erased. Here we give the text CCICE1 as an example. In this short text, two Spanish DMs have removed (*y* ['and']; *así* ['thus']), while one DM (*por el contrario* ['in contrast']) is not.

For the Chinese texts, we can see that, the lowest results of Chinese texts fall on TERM25 (0.657), TERM30 (0.746), BMCS3 (0.795), CCICE1 (0.538) and FICB4 (0.662). We give a qualitative analysis for the texts that contain low results and we find some common limitations for these texts, here we give CCICE1 as the example. In this text, the Chinese character "*wei*" (为) appear 7 times, however, none of the them can be considered as the DM, whose discourse meaning is 'aim for'. Based on the short text content, the character "*wei*" (为) means 'as'. Another limitation related with this character appears in the text FICB4. The Chinese phrase "*zuowei*" (做为) ('as' in English) contains the annotated DM "*wei*" (为), however, in this case, together with the character "*zuo*" (做) ('make / to do' in English), "*wei*" (为) cannot be considered as a DM.

The sequence of the phrases in Chinese also brings us some limitations during the test process. Among the annotated Chinese DMs, one of them is "*zhizai*" (旨在), whose meaning is 'to do something' or 'aims to do something' in English. In the text TERM25, the phrase "*zhuzhi*" (主旨) ('main purpose') that ends with "*zhi*" (旨) is next to the phrase "zaiyu" (在于) ('lie in') who starts with "*zai*" (在)[6]. Since there is no space between Chinese characters in a text, hence, our programming considers "*zhi*" (旨) and "*zai*" (在) as a DM.

In conclusion, these are the limitations of our program in

[4] For example, the Chinese DM "*wei*" (为) means 'aim for' in English, but it also means 'as', 'to help' in Chinese.

[5] As section 4.1 shows, the research corpus has different sources, therefore, the number of selected texts for test based on the percentage of each part in the research corpus, and the appearances of annotated DMs in each text.

[6] The original content in TERM25 is "*zhuzhi zaiyu*" (主旨//在于), and as we have indicated, "*zhizai*" (旨在) is a DM.

the Chinese subcorpus:

• A character could have different meanings depending on the text content, so that sometimes it is a DM and sometimes not, but our system cannot understand the text content.

• Some DMs are composed of two Chinese characters, however, our programming just annotates one character. For instance, in the case of "*yiji*" (以及), the first character "*ji*" (及)[7] is removed from the output. Other similar cases exist.

• The possible phrase sequences can cause the characters combine as an annotated DM, for example, the case of "*zhuzhi zaiyu*" (主旨//在于) that we explained before.

• Some Chinese DMs are single characters, but they can convert to a new phrase together with another different character, under this case, we cannot consider this character as a DM anymore.

6. Conclusion and Future Work

In this work, we have presented the first RST Spanish-Chinese parallel corpus that can be used for language exercise with multichoices for each text.

In this work, we have presented the first RST Spanish-Chinese parallel corpus, which can be used for the important task of discourse analysis. Experts have annotated it manually so that DMs are indicated and classified.

With educational aim, we have used the corpus to create a program that automatically generates language exercises with multiple choices for each text. These exercises are useful to learn the usage of discourse markers for Spanish and Chinese language students.

Despite the simplicity of our program, we get very good result for the Spanish subcorpus. In the case of the Spanish exercises, our system can also grade the users' answers automatically. While for Chinese subcorpus we get some limitations, it can also give good L2 Chinese language exercises.

For future work, we will annotate more cases for Chinese subcorpus to get better results, and we will also make our program to be able to grade the Chinese language exercises.

7. Acknowledgements

We give our acknowledgements to the IXA group of the University of Basque Country, for the construction of the corpus webpage. Especially, we want to thank Itziar Aldabe, for her insightful ideas on this work.

8. References

Asher, N., and Alex, L. (2003). *Logics of conversation*. Cambridge: Cambridge University Press.

Burstein, J. (2009). Opportunities for Natural Language Processing in Education. 2009. *Lecture Notes in Computer Science*, 6-27

Cao, S. Y., da Cunha, I., and Bel, N. (2015). An analysis of the Concession relation based on the discourse marker *aunque* in a Spanish-Chinese parallel corpus. *Procesamiento del Lenguaje Natural*, 56: 81-88.

Cao, S. Y., da Cunha, I., and Iruskieta, M. (2017). Toward the Elaboration of a Spanish-Chinese Parallel Annotated Corpus. *EpiC Series in Language and Linguistics*, 2: 315-324.

Cao, S. Y., Xue, N. W., da Cunha, I., Iruskieta, M., and Wang, C. (2017). Discourse Segmentation for Building a RST Chinese Treebank. In *Proceedings of 6th Workshop "Recent Advances in RST and Related Formalisms"*, 73-81.

Chen W., Aist G., and Mostow J. (2009). Generating Questions Automatically from Informational Text. In *The 2nd Workshop on Question Generation, volume 1 of AIED 2009 Workshops Proceedings*, 17-24.

Chien, Y. S. (2012). Análisis contrastivo de los marcadores condicionales del español y del chino. *MarcoELE: Revista de Didáctica Español Lengua Extranjera (14)*.

da Cunha, I., and Iruskieta, M. (2010). Comparing rhetorical structures of different languages: The influence of translation strategies. *Discourse Studies*, 12(5): 563-598.

Eckle-Kohler, J., Kluge, R., and Gurevych, I. (2015). On the Role of Discourse Markers for Discriming Claims and Premises in Argumentative Discourse. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP' 2015)*, 2236-2242.

Fang, Y. (2008). *Zhongguo yingyu yupian tezheng yanjiu: zhongmei shiguxing xinwenwenben duibi fenxi* (中国英语语篇特征研究：中美事故性新闻文本对比分析, *[*Discourse of China English: Rhetorical Relations in News Texts in *China Daily* and *The New York Times]*). Master thesis. Shanghai: Donghua University.

Heilman M., and Smith N. A. (2010). Good Question! Statistical Ranking for Question Generation. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL (NAACL' 2010)*, 609-617.

Iruskieta, M., da Cunha, I., and Taboada, M. (2015). A Qualitative Comparison Method for Rhetorical Structures: Identifying different discourse structures in multilingual corpora. *Language resources and evaluation*, 49(2): 263-309.

Iruskieta, M., Diaz de Ilarraza, A., and Lersundi, M. (2015). Establishing criteria for RST-based discourse segmentation and annotation for texts in Basque. *Corpus Linguistics and Linguistic Theory*, 11(2) : 303-334.

Kingsbury, P., and Palmer M. (2002). From Treebank to PropBank. In *Proceddings of International Conference on Language Resources and Evaluation (LREC' 2002)*, 1989-1993.

Li, H. K., and Liao, Z. H. (2015). A Rhetorical Structure Theory Based Approach to Texture Coherence in

[7] "*yiji*" (以及) and "*ji*" (及) are two annotated Chinese DMs in our work, both of them represent a LIST discourse relation. The meaning of the two DMs are same, is 'and' in English.

Chinese EFL Learners' Argumentative Writing. *Overseas English*, 16: 201-204.

Liang, S. S., and Yang, Z. L. (2016). *Hanguo xuesheng kouyu duochong yinguozhuanzhe yupian shiyong qingkuang fenxi* (韩国学生口语多重因果转折语篇使用情况分析，*[*A study of Multiple Causal and Transitional Discourse in Korean Students' Spoken Chinese*]*), *Chinese Teaching in the world*, 30(3): 356-367.

Mann, W. C., and Thompson, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text&Talk*, 8(3): 243-281.

Marcu, D. (2000). The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3): 395-448.

Marcus, M., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2): 313-330.

Miltsakaki E., Prasad R., Joshi A., and Webber B. (2004). The Penn Discourse Treebank. In *Proceedings of 4th International Conference on Language Resources and Evaluation (LREC' 2004)*, 2237-2240.

O'Donnell, M. (2000). RSTTool 2.4 – A Markup Tool For Rhetorical Structure Theory. In *Proceedings of First International Conference on Natural Language Generation*, 253-256.

Olney A. M., and Graesser A. C., and Person N. K. (2012). Question Generation from Concept Maps. *Dialogue and Discourse*, 3(2) : 75-99.

Pórtoles, J. (2001). *Marcadores del discursivo*. 4th edition. Barcelona: Ariel.

Schiffrin, D. (2001). Discourse markers: language, meaning, and context. *The handbook of discourse analysis*, 1: 54-75.

Wang, Y. C. 2013. *Los marcadores conversacionales en el subtitulado del español al chino: análisis de La mala educación y Volver de Pedro Almodóvar*. PhD thesis. Barcelona: Universitat Autònoma de Barcelona.

Yao, J. M. (2008). *Estudio comparativo de los marcadores del discurso en español y en chino a través de diálogos cinematográficos*. PhD thesis. Valladolid: Universidad de Valladolid.

Zhang, Y. (2010). *Zhongguo daxuesheng he yingmei liuxuesheng tongti yingwenzuowen de xiucijiegou fenxi yu bijiao* (中国大学生和英美留学生同题英文作文的修辞结构分析与比较，*[*An Analysis of Compositions by Chinese Students and Native Speakers Based on Rhetorical Structure Theory*]*). Master thesis. Hangzhou: Zhejiang University.

Zhou, L. J., Wei, Z. Y., and Wong, K. F. (2014). The CUHK Discourse TreeBank for Chinese: Annotating Explicit Discourse Connectives for the Chinese TreeBank. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC' 2014)*, 942-949.

A 2nd Longitudinal Corpus for Children's Writing with Enhanced Output for Specific Spelling Patterns

Kay Berkling
Cooperative State University
Karlsruhe, Germany
berkling@dhbw-karlsruhe.de

Abstract

This paper describes the collection of three longitudinal Corpora of German school children's weekly writing in German, called H2 (H1 is available via LDC and contains some of the same students' writing 2 years previously), E2 (E1 is not public), and ERK1. The texts were written within the normal classroom setting. Texts of children whose parents signed the permission to donate the texts to science were collected and transcribed. The corpus consists of the elicitation techniques, an overview of the data collected and the transcriptions of the texts both with and without spelling errors, aligned on a word by word basis. In addition, the hand-written texts were scanned in. The corpus is available for research via Linguistic Data Consortium (LDC). When using this Corpus, researchers are strongly encouraged to make additional annotations and improvements and return it to the public domain via LDC, especially since this effort was unfunded.

Keywords: Orthography, Acquisition, Corpora, Elementary School, Writing, Digitization

1. Introduction

Reading and Spelling are key skills acquired by children during their first four years of school. Unfortunately, it is not clearly understood how and why performance for standardized test populations may degrade or improve. They may be due to teaching style or any other factor. Useful studies might look at how writing changes in longitudinal studies or as a function of particular training programs, thereby lending insight into quality of school books or teaching philosophies. However, very little of this kind of validation is done on a larger scale or open to comparative research with open corpora.

Two comparative standardized exams highlight that we have a serious problem that needs addressing without delay. The IQB study looks at reading, mathematics and spelling ability and has been performed in 2015 (9th grade), 2016 (4th grade) and will look at science in 2018. It represents a regional addition to the international PISA test and looks at skills across the different German states. Over the last few years, results for Baden-Württemberg have fallen dramatically (Stanat et al., 2016).

Another study, VERA in 2017 (Blank and Schult, 2017), clearly demonstrates that student competence in orthography is much worse than expected. Instead of the predicted 35%, an actual 64% of students place in the lowest 2 out of 5 competency levels in orthography. The skill of reading is less dramatic with around 40% of students expected and actually measured at the lower 2 competency levels. However, both skills show scandalous results when looking at children who are not speaking German in their homes (an estimated 20% of the total number of students): Students tested into the lower 2 levels contain 70% (instead of 34% for German speakers) in reading and 80% (instead of 60% for German speakers) in orthography.

Since we started collecting corpora to support research in the area of orthography acquisition, the problem has become more acute. Therefore, the need for data and research in this area is increasingly dire. The corpus presented in this paper is another significant contribution in this area and remains one of the rare resources that are available publicly. More background information to this problem is already presented in the introduction of the precursor paper at LREC 2016 (Berkling, 2016), which introduced the H1 Corpus (Linguistic Data Consortium, 2016).

This paper describes the corpus that has been collected since LREC 2016. It consists of three sets from different schools, H2, E2 and ERK1. The number indicates the nth collection. While E1 is not public, H1 is published in the above paper and available through LDC. H2 contains a few of the children in Grade 4 who wrote in H1 in Grade 2. The mapping is available with the corpus. E2 and ERK1 both participated in a program to train orthography called Phontasia (Berkling and Pflaumer, 2014). One of the classes from ERK1 who participated in the training scored 32% compared to 64% national average into the lower 2 competency levels, thereby dramatically outperforming other classrooms. More on the difficulty of studying orthography acquisition and the impact of interventions on the present data set can be found in (Berkling, 2017).

2. Data Collection

This new corpus extends the previous collection. It is larger, contains longer sessions of 16 weeks for older children (4th and 6th grade) and second participants as well as the original 9-week session for new classes of 2nd and 3rd graders. In addition, some of the corpus subjects have been exposed to an orthographic training that may have an influence on the writing ability. As in the last corpus, there are formal pre- and post-tests that are the same for all participants that are comparative across all classrooms.

This section describes the collected data and the data transcription and annotation methods. All data are collected in elementary schools and one Hauptschule (academically lowest of three school forms available to children after triage in fourth grade) in the state of Baden Würtemberg,



Figure 1: Picture for text elicitation in Week 3 for older children or those participating in study for second time.

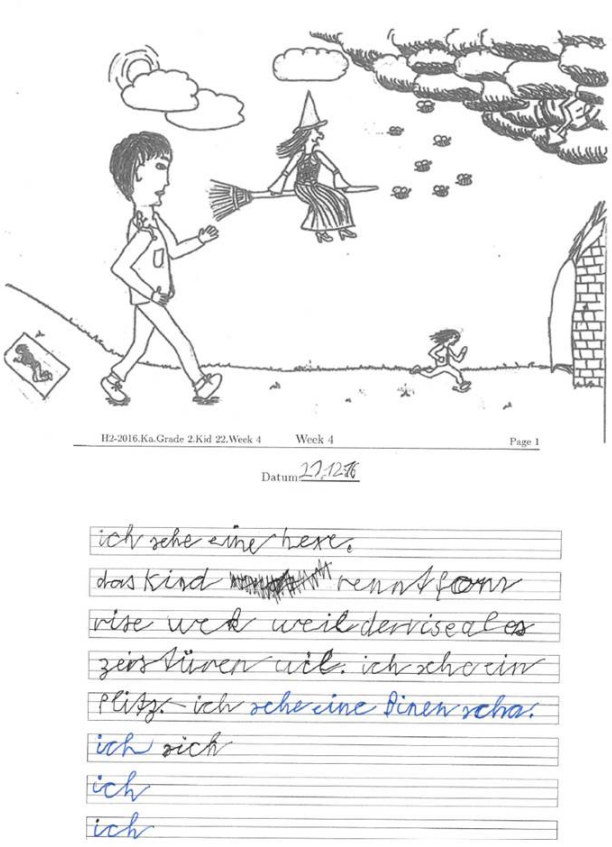


Figure 2: Picture for text elicitation in Week 4 for younger children.

Germany. The texts are digitized and transcribed by hand at the Cooperative State University of Karlsruhe during the 2016/2017 school year. A total of 13 classrooms participated distributed across the corpora as depicted in Table 2., indicating which classes were part of an intervention similar to (Berkling et al., 2015). The resulting total of 173 children and 2117 texts are listed in detail in Table 1, disregarding G6 and VKL, since they have not yet been transcribed and may enter the public corpus at a later time. VKL is a preparatory classroom of refugee children, learning German.

Out of these 173 children, meta data is available for 166, 100 of these are multilingual. Every week one text was written, resulting in a total of 2117 texts. The word count is given in Table 3.

Table 1: Distribution of Texts by Classrooms. (.5 indicates that only half of the pre- or post-test is available (either the dictation or the word elicitation by picture. Details are documented with the Corpus.

		H2	ERK	E2	Total Texts
2	A	171	0	0	171
	B	226	0	0	226
	C				0
3	A	111	167	277	555
	B	200	173	0	373
	C	137.5	0	0	137.5
4	A	305	0	0	305
	B	97	0	0	97
	C	253	0	0	253
	Kids	114	35	24	173
				Total Texts	2117.5

2.1. Text Elicitation

Texts were written within regular class settings, the instructions given to the teacher are included in the corpus. The pictures that are used for text elicitation are designed to enhance the output with respect to important spelling error categories, namely the marking of short vowels with a silent consonant letter and the correct spelling of the long vowel <ie>. This is motivated by previous work showing these to be critical error categories that are both frequent and persistent until the upper grades of high school (Berkling and Lavalley, 2015). The pictures for the 9-week session are mostly the same as in the H1 corpus. Only Week 1 picture was changed because the old picture was not effective, though the topic and words that went into the drawing are similar to the H1 study. Therefore, all week writings are comparable between this corpus and the H1 corpus. Children had at least 15 minutes time to write the texts. They were asked to write a picture description or a story. If unable to write a text, they were asked to list the things they see on the pictures. An example of such an output is de-

School Grade	2	3	4	6	VKL	Phontasia
ERK1		2				yes
E2		1				yes
H2	2	3	3	1	1	no

Table 2: Table listing the number of classrooms by corpus and grade level. Note that Grades 6 and VKL are still in the process of being transcribed and will not appear in further tables.

Table 3: Size of Corpus by Words

School	Grade	Word-Type	Word-Token
E2	G2	0	0
	G3	2420	17692
	G4	0	0
H2	G2	1985	15239
	G3	3866	34699
	G4	3914	32067
ERK1	G2	0	0
	G3	2823	18924
	G4	0	0
Total		8619	118621

picted in Figure 1 for a sixth grade example or Figure 2 for a second grade example text.

2.2. Pre- and Post-tests

Pre- and post tests contain the same writing material across all writers and were administered as an anchor with respect to the orthographic skills measured through the spontaneous texts based on picture elicitation. The pre- and post tests were given at the beginning and end of the data collection. The words were split between dictation and picture naming to account for dictation bias. The words are chosen to have a high frequency in 2-syllable and 1-syllable words containing <ie> and short vowel marking with silent following consonant letter as well as a few higher level concepts and exceptions and complex onsets (”Hühner”). The list deviates slightly from those in the H1 Corpus and is shown in Table 4.

2.3. Meta Data

Meta data was collected for every text in the database.

- school week of collection
- school type
- age
- gender
- grade / classroom
- language spoken at home
- school materials used (always Jojo in elementary school)

Category		Wordlist
Pretest	Dictation	gehen. Der Wind weht. Schreck. steht. tief. drehen. dreht. Schuhe. Hühner. heulen. Beule. Farbe. lernt. Er wirft. Schrecken. Sie spielt. Er trinkt Wasser. Decke. Pfütze. Pfanne.
	Pictures	Bohne, Huhn, Butter, Drachen, Wiege, Ziege, Locke, Horn, Feder, Messer, Rahmen, Zügel, Teller, Zwiebel, Pfütze, Rose, schlafen, Säbel, Seife, Straße, Treppe, Tunnel, Geige, Brunnen, Wurm, Robbe
Posttest	Dictation	Er steht. Sie ruhen. Sie schnarchen. Speck. Sie ermahnen. Sie lahmen. Hitze. Er sitzt. jucken. Ich weiß. Kuss. Es brennt. Flamme. Schleife. Sie streiten. Er tritt. leider. beide. morgen. Sie werfen.
	Pictures	Herz, Betten, Bügel, Brief, Flügel, Drachen, Hase, Kamm, Parkplatz, Reifen, Ritter, Pudel, Schnuller, Sack, Wasser, Spaten, Mücke, Zahn, Schnecke, Sahne, Sternschnuppe, Sonne, Waffel, Decke, Brunnen, Leiter, Hose, Biene, Treppe, bohren

Table 4: Wordlist for Pre- and Post-test. Words are elicited via pictures or dictation.

3C	pre	w3	w4	w5	w6	w7	w8	w9	w10	w11	post	w13
1	x	x	x	x	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x	x	x	x	x

Table 5: Texts collected by class, ID, and week

ID	Grade	f/m	Age	Languages
H2.KA.G2.2	G2	m	8	g
E2.KA.G2.4	G2	m	9	e ar g

Table 6: Meta-data for Corpus. Author ID, classroom, gender, age at time of writing, and language biography (ar=aramaic, al=albanian, k=kurdish, g=german, e=english, i=italian, t=turkish).

As was done in the H1 Corpus, statistics about the data are released with the data itself. There are files containing the school week for which children wrote texts, the list of texts submitted by class, week and child (including absences) as exemplified in Table 5, and the meta data (see excerpt in Table 6). In addition, the packages (including templates and pictures) as well as instructions given to the teachers are available.

2.4. Anonymization

Texts were submitted in anonymized fashion. A few mistakes by children were corrected.

3. Transcription

The obtained texts were digitized in two forms: the original text, including all errors (achieved) and the intended (target) text, where all spelling errors have been removed. Annotations are needed at this level to distinguish the words that should not be analyzed for spelling errors such as names or foreign words. All annotations, as listed in Table 7 are added to both the target and achieved text to maintain a word by word match between the two texts, see also (Berkling and Lavalley, 2015; Lavalley et al., 2015). In order to prepare for sentence-level analysis, syntax errors have been annotated by marking substitutions, deletions and insertions at word level. In such cases, the used word is analyzed for spelling and the correct word is used for sentence structure analysis. The annotation conventions used in the transcription are listed in a Table 7 at both word and sentence level.

Letter- and Word-Level Annotations:	
*	unreadable letter
a_b	a and b should have been written separately
a§b	a and b should have been joined
a=b	missing hyphen
a∼b	wrongly placed hyphen
a−−b	denotes split of word at end of line (not hyphen)
a{n}	n repetitions of word a
a{F}	Foreign word defined by non-German graphemes foreign grapheme-phoneme correspondence
a{G}	grammatical errors not to be analyzed for spelling
a{N}	Names, not analysed with the spell tagger
Sentence Level Annotations	
[§ fW]	an unknown deletion
[§ b]	a known deletion b
[a §]	an insertion a
[a b]	substitution of a for b a is corrected on target side Achieved: [seinne ihre] Target: [seine ihre]
[a b_c]	best guess of word boundary
[a_b c]	kanicht = ka[n nn_n]icht
[a *]	some combinations of letters make up word a the real word can not be identified.
a can include conventions from word-level annotations For example: [rtchen**gdsdfg *] [rtchen**gdsdfg *] or [a{G} b] Numbers (1,2,..): kept as numbers. Words with exaggerated spelling: [Leeeeooooooon Leon].	

Table 7: Conventions for annotation of transcriptions as relevant to automatic spelling annotation.

The use of transcription convention is shown in the following example. We have printed the sequence of texts by a single child in third grade chosen at random. Notice how the text changes, both length of the text and complexity of the sentence seems to increase by inspection. The corpus therefore gives a rich basis for study of text development. Note, that there was no feedback on the writing given by the teacher. This may indicate that writers improve simply through writing. These are research questions that still need to be answered. The example is taken from school ERK, looking at "a_" the achieved transcription (including spelling errors) of child 10, in Grade 3 (G3) and classroom A (KA).

School: ERK1, Child: a_G3.KA.10

Week 1:
Die Puppe sitzt auf dem Stul . Ein kleiner Junge rennt weg . 2{N} Fische schwimmen im Aqauruim{F} . Die Birnen stehen in einem Kasten . Ein Schüpfer , Löffel und Schüssel liegen auf einem Regal . Ein Junge bindet seine Schue .

Week 2:
Neben dem Fenster steht ein Bett . Zwei Bilder hängen an der Wand . Neben dem Bett steht ein Besen . Ein Sessel steht neben der Wand . Eine Puppe liegt auf dem Boden . Zwei Federn kleben an der Wand . Eine Blumme steht auf dem Schrank . Die Bienne fliegt im Zimmer . Auf dem boden liegen Spielzeuge . Im Zimmer ist es unordentlich . Die Giskanne steht neben der Blumme . Das Buch liegt auf dem Boden . Ein Bild liegt auf dem Boden .

Week 3:
Die Babysind in der Schule . Eine Robe feiert sein{G} Geburtstag mit seinen Freunden . Zwei Roben Frühstucken auf_einem Eisblok . Zwei Robenkinder spielen mit einem Ball . In der ferne schwimmt ein Boot . Eine Robe sucht futter . Eine Robe unt erichtet Gymmnastik .

Week 4:
Das Hexengewiter An einem schönem{G} Morgen spielten zwei Kinder auf der Wiese . Plötzlich began es zu Donnern und eine Hexe fliegte{G} auf seinem{G} Besen her . Sie hatte auf seinem{G} Kleid ein Spinnennetz , ein{G} Hut und eine gresliche Nase . Die Kinder sind schnel in den Tunnel gerant unn kuckten ängstlich auf die Hexe . Die Hexe sagte : " Es solle ein Gewitter geben ! " Dan erlöste sie sie sich in luft [§ .] es gab ein Gewitter [§ ,] aber es ging schnel vorbei und niemand hat die Hexe gesehen .

Week 5:
Sommerpiknik An einem herlichen Sommartag spielten drei Kinder auf der Wiese . Zwei Hasen sprangen rum und Biennen_sumten herum . Ein Mädchen mallte ein Bild mit wunderschönen Farben , das Mädchen heist Lena{N} .

Week 6:
Der Landriese In einem fernen Land [§ ,] des ganz ungewönlich ist [. ,] Ist alles riesengroß ! Biennen wie Bären , Raben wie Girafen und Menschen wie Risen ! Die Hare aus Bäumen [§ .] Das schöne Bild Es gibt einen{G} ungewonliches Land [§ ,] es heist Landwunder und in diesem Land ist alles riesengroß . Biennen zum vergleich wie Bären !

Week 7:
Eilein zu Haus Dima{N} und Julia{N} ganz gewonliche Kinder . Julia{N} ist drei und Dima{N} ist fümf . Aber einmal ist was ungewonliches pasirt [§ .] Dima{N} hatte ein Zauberbuch gelesen und dann wurde das Kinderzimmer umgeschütelt und sie waren in einer neuen Welt . Julia{N} hatte ein wunderschönes kleid [§ ,] ein blaues [§ ,] wie sie es gewünscht hatte . Drausen stand eine Ziege und sie hatte Julia{N} gefragt [§ ,] ob sie was essen durfte von den leckeren Blättern [§ .] " Ja " [§ ,] sagte das Madchen und plötzlich war alles vorbei [§ ,] weil sie Getreumt hatte . Aber später wuste sie [§ ,] das in echt pasiert ist . Weil Dima{N} hatte seinen Traum auch erzält [§ ,] es war der gleiche [§ .]

Week 8:
Im Freizeitspark An einem heisem Mittag [, §] wollten die Kinder aus der Familie Kutarie{N} in den Freizeitspark gehen . Papa und Mama stimten zu . Die Tiere Rubo{N} Hase , Bello{N} Hund und Pierie{N} Ziege dürfen auch mitkommen . Da waren sie also im Freizeitspark [§ .] niemand war da [§ ,] aber die Familie war so groß alles besetzten . Karie{N} Baby{F} spielte mit Mama im sandkasten , Sändie{N} giest die Blumen , Tom{N} und Julia{N} Tenis , Max{N} rutst auf der Rutsche [§ ,] Karina{N} malte ein schönes Bield und Toni{N} wartete mit Pierie{N} auf Papa [§ ,] der auf dem Klo war . Und Abents dachte Baby{F} Karie{N} [§ ,] ob sie Morgen wider in den Freizeitspark fahren können .

Week 9:
Die Piraten-Insel Heute sind die Piraten auf der Piraten Insel , sie haben viel Arbeit . Die junge Merjungfrau Erielia{N} wird den schatz im Wasser suchen . Grade zeigt der Piratenschef{F} Ahoj{N} wo_der Schatz liegt . Der andere hat Fische geangelt . Arengut{N} hat Pause . Die Kinder Larend{N} und Rud{N} waren in der Schule . Und die Restlichen tragen Schätze , Tiere essen und trinken an Bord . Abents konnten endlich los_fahren und Erielia{N} konnte mitschwimmen . Alle waren frölich , und des war die Geschichte von der Pirateninsel . Ende

4. Data Exploration

The following results exemplify the kind of work that can be accomplished on the children's text corpus. Since our work concentrates on orthographic development, we were able to use the automatic error tagger on all the texts within minutes in order to explore the possibility of classroom diagnostics or long term development of particular orthographic skills. This automated process and its performance is described further in (Berkling and Lavalley, 2017).

Figure 4 depicts the mean change in text length across several classrooms. A better depiction is given in Figures 7 and 8 and shows the distribution as well as the change. $E2$

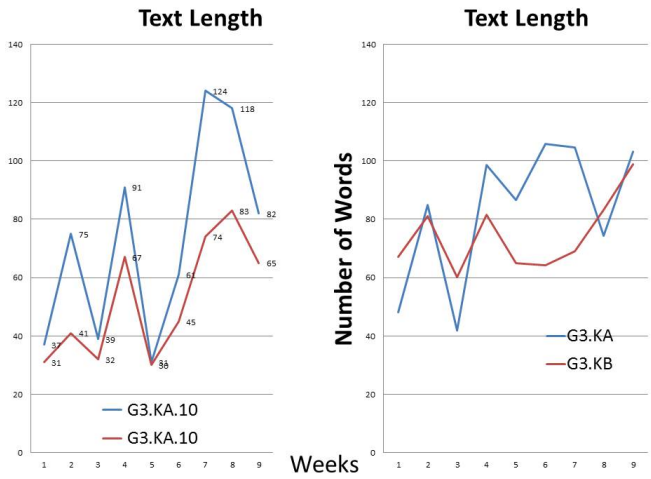


Figure 3: Change in lengths of stories, comparing Kid ERK1.a_G3.KA.10 (blue = word count, red = vocabulary) across the nine weeks (see also example in Section 3.) with two different classroom average lengths.

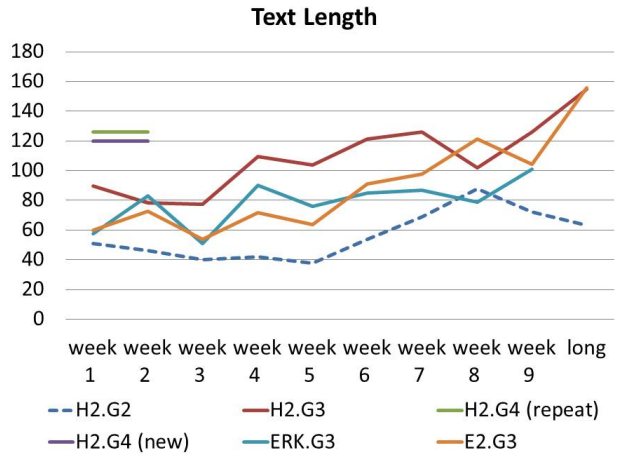


Figure 4: Change in average lengths of stories across some different classrooms.

and ERK have used Phontasia (Berkling and Pflaumer, 2014). This game teaches German Phonics skills to kids by starting with simple word patterns and leveling up to more difficult patterns, focusing on German specific orthographic patterns that traditionally cause problems for children given some of the current elementary school teaching methods that suppose that one phoneme corresponds to one grapheme when transcribing speech into text. The game asks children to search their memory for words with particular patterns that they are then required to enter into the game to proceed. This may support vocabulary growth, especially, when kids work in teams to come up with further words for particular patterns. Whether Phontasia contributes in supporting text writing is one of the research questions that can be addressed with this corpus.

Another interesting question is text length and quality. Figure 4 shows that even though there is no explicit feedback given to the children, the average text length in all

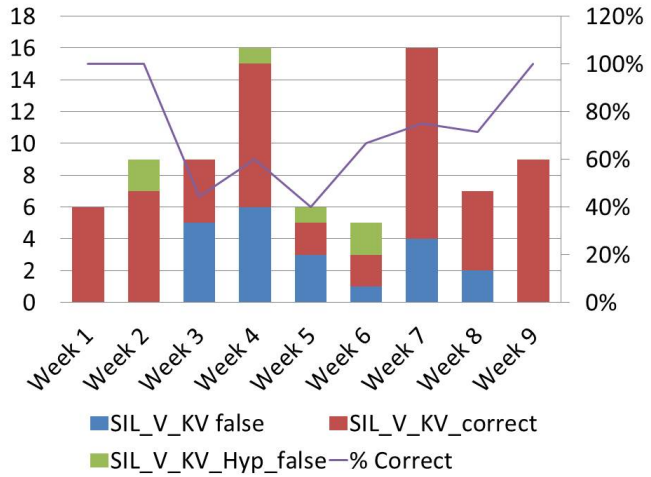


Figure 5: Development of one particular spelling error over the weeks, namely short vowel marking, relating number of items written correctly vs. incorrectly, compared to hyper compensation and% correct across the weeks. This shows how non-linear acquisition is.

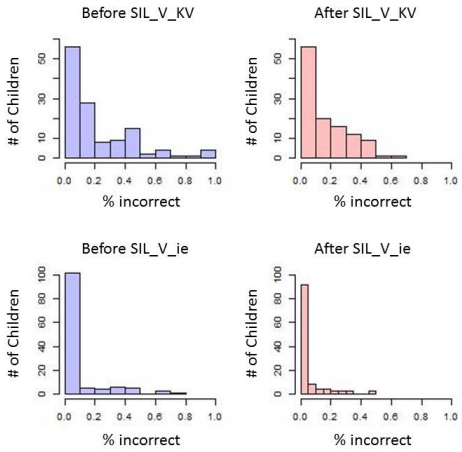


Figure 6: Development of spelling errors (correct transcription of long sounding <ie> and SIL_V_KV across all children in ERK.G3.

classrooms increases. The student in the example of Section 3. can be analyzed regarding one of the most important spelling errors for kids, here (and in related publications) it is called SIL_V_KV (see also (Lavalley et al., 2015) for more details on spelling error categories and statistics across large corpora). This category refers to the doubling of consonant letters to modulate the length of the preceding vowel , ie. "rate" (guess) vs. "Ratte" (rat). In doubling the <t>, the vowel <a> is pronounced short. This trivial concept is not mastered by many kids. Interestingly it is also not explicitly taught in the first years of orthographic acquisition. The example shows the non-linear development for orthographic transcription. This child has also had the Phonatsia intervention. Figure 5 depicts the proportion of correctly (red) vs. incorrectly (blue) spelled items for this error category, along with any hyper-compensation (green). The lines shows the % correct value for this spelling pattern. Though the student seems to be getting worse in the beginning, the acquisition pattern is actually quite complex. Even as the texts get longer, the spelling errors seem to be decreasing for this student. Mastery of this highly frequent spelling pattern is a non-linear process with most children exhibiting multiple U-shaped learning patterns. Looking at the non-linearity it becomes highly questionable to compare students to each other at a particular point of examination without taking the personal acquisition process into account.

Looking at the classroom of all children, it can be seen in Figure 6 that the distribution of errors comparing the first week and the pre-test to the last week and the post-test has improved.

5. Conclusions

We have provided a digitized transcription for a publicly available data set of student writings. The data are available via the Linguistic Data Consortium (H2, ERK1, E2 Children's Text; LDC (Linguistic Data Consortium, 2018)).

There is no report on inter-annotator agreements. Similar to the H1 Corpus, the goal of this unfunded work was to publish the resource and it's transcription. Improvements to the transcription are highly welcome.

Understanding written L1 language acquisition is a prerequisite to diagnosis and supporting tools. Even in 2017, very little work exists joining those three areas of study and using speech and text processing technology for automatic analysis of large amounts of data. With more know-how in this area, the field of personalized training for children can grow. The clear need for this kind of work is evident in the negative development of children's skills in vocabulary size, orthography, reading, and the sciences, which depend on the ability to be able to read and write well.

6. Acknowledgements

The existence of this data is due to a number of dedicated teachers, a large number of children who write for us, and the parents who donate these texts to help understand the process of learning so that we can improve our knowledge of orthographic acquisition. Special thanks go to the transcribers who have worked hard on this large dataset. This work is a grass-roots effort that was not funded. Thanks also go to DHBW-Förderverein for supporting parts of the transcribers' work.

7. Language Resource References

Linguistic Data Consortium. (2016). H1 Children's Text Corpus. https://catalog.ldc.upenn.edu/LDC2016T01. Available: 2016-04.

Linguistic Data Consortium. (2018). H2, E2, ERK1 Children's Text Corpus. https://catalog.ldc.upenn.edu/LDC2018T05. Available: 2018-04.

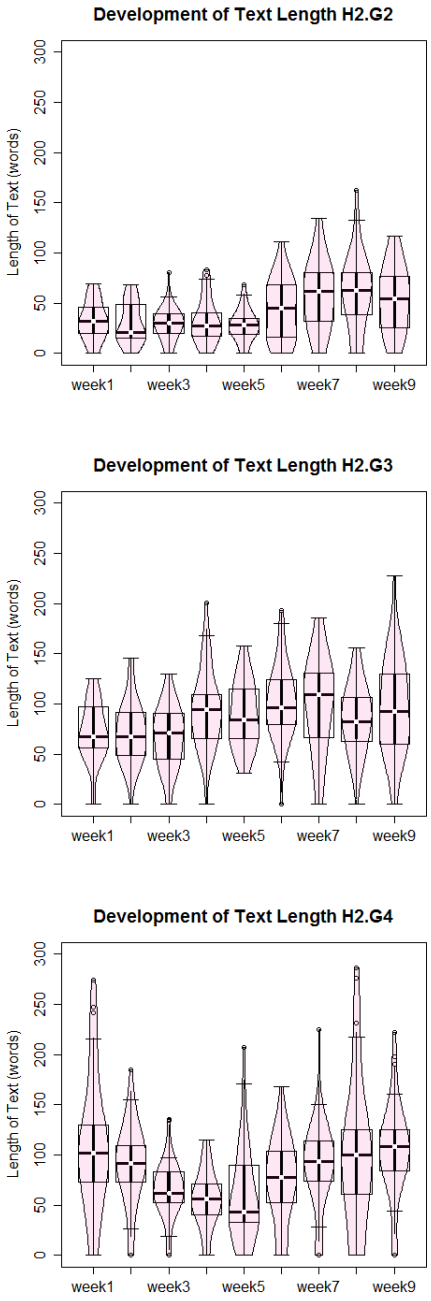


Figure 7: Longitudinal change in lengths of stories shown as distributions over entire population (without intervention).

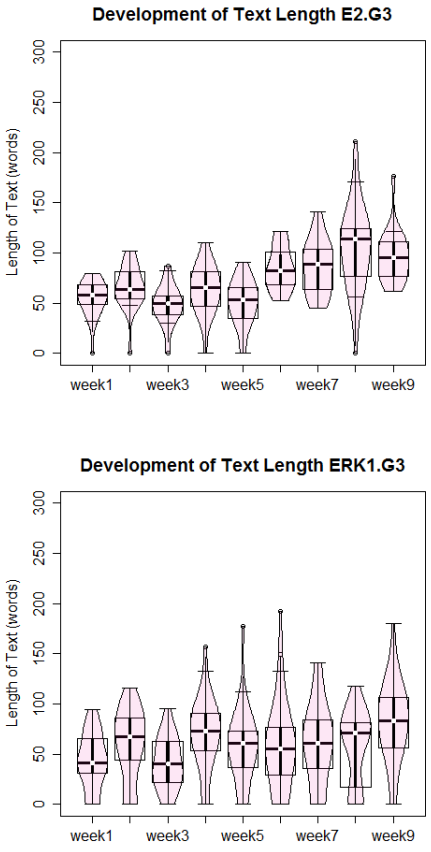


Figure 8: Longitudinal change in lengths of stories shown as distributions over entire population (with intervention).

8. Bibliographical References

Berkling, K. and Lavalley, R. (2015). WISE – A Web-Interface for Spelling Error Recognition in German: A Description and Evaluation of the Underlying Algorithm. In German Society for Computational Linguistics and Language Technology, editor, *GSCL*.

Berkling, K. and Lavalley, R. (2017). Automatic orthographic error tagging and classification for german texts. *Computer Speech Language*.

Berkling, K. and Pflaumer, N. (2014). Phontasia - a Phonics Trainer for German Spelling in Primary Education. In CHILD-SIG Interspeech, editor, *Workshop on Child Computer and Interaction*, pages 33–38.

Berkling, K., Pflaumer, N., and Lavalley, R. (2015). German phonics game using speech synthesis - a longitudinal study about the effect on orthography skills. In Stefan Steidl, et al., editors, *ISCA International Workshop on Speech and Language Technology in Education, SLaTE 2015, Leipzig, Germany, September 4-5, 2015*, pages 167–172. ISCA.

Berkling, K. (2016). Corpus for children's writing with enhanced output for specific spelling patterns (2nd and 3rd grade). In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA).

Berkling, K. (2017). Phontasia: A phonics game for german and its effect on orthographic skills–first corpus explorations. In *Proc. WOCCI 2017: 6th International Workshop on Child Computer Interaction*, pages 7–16.

Blank, S. and Schult, J. (2017). Bericht VERA 3.

Lavalley, R., Berkling, K., and Stüker, S. (2015). Preparing Children's Writing Database for Automated Processing. In *Workshop on L1 Teaching, Learning and Technology (L1TLT)*, Leipzig, Germany.

Stanat, P., Böhme, K., and Schipolowski, S. (2016). IQB-Bildungstrend 2015: Sprachliche Kompetenzen am Ende der 9. Jahrgangsstufe im zweiten Ländervergleich.

Development of a Mobile Observation Support System for Students: FishWatchr Mini

Masaya Yamaguchi, Masanori Kitamura, and Naomi Yanagida
National Institute for Japanese Language and Linguistics, Nanzan University, Hitotsubashi University
10-2 Midori-cho, Tachikawa City, Tokyo, Japan; 18 Yamazato-cho, Showa-ku, Nagoya, Japan;
2-1 Naka, Kunitachi City, Tokyo, Japan,
masaya@ninjal.ac.jp, mkita@ic.nanzan-u.ac.jp, n.yanagida@r.hit-u.ac.jp

Abstract

This paper presents a system called FishWatchr Mini (FWM), which supports students in observing and reflecting on educational activities such as presentation practices. Although video annotation systems are supposed to support such activities, especially reflection, they have been used by researchers and teachers outside the classroom, rather than by students inside the classroom. To resolve problems with introducing video annotation systems in the classroom, we propose three solutions: (a) to facilitate preparation of devices, FWM is implemented as a Web application on students' smartphones; (b) to facilitate students' learning to use the system, FWM is designed to automate as many operations as possible, apart from annotation; and (c) FWM allows students to examine annotation data through reflection, by providing functions such as visualization. The results of applying FWM to presentation practices in a university validated the effectiveness of FWM in terms of its ease of use and applicability of annotation results. Since annotated video data could also provide language resources for teachers, it is anticipated that they will contribute to improving their classes in the future.

Keywords: video annotation tool, real-time annotation, reflection support

1. Introduction

Educational activities such as presentation practices, discussion exercises, and trial lessons for trainee teachers have been conducted in university classrooms. These activities include cooperative phases (Barkley et al., 2005; Morimoto and Otsuka, 2012): Together, students observe their practices and reflect on them after the observations.

Thus far, many video annotation tools have been developed to observe such activities (Rich and Hannafin, 2009; Yousef et al., 2014) because recorded video and sound are useful tools, particularly for reflecting on practice. Most of these tools, however, have been designed for researchers to make complex annotations (Brugman and Russel, 2004) or for teachers to share videos and comments as web-based systems (Kong et al., 2009; Colasante, 2011). These tools are not suited to students' real-time annotation and to group reflection in the classroom.

In this paper, we developed a system called FishWatchr Mini (FWM), which supports students' observation and group reflection in the classroom. To validate the effectiveness of FWM, we presented the results of the application to presentation practices in a university setting.

We believe that one of the reasons that video annotation systems have not been introduced effectively in the classroom is the stringent condition of the system operation: many students, as observers, make real-time simultaneous annotations and the results of those annotations are then used in the group reflection phase.

FWM is designed to address the following three problems related to the introduction of video annotation systems in educational activities in the classroom.

The first problem is preparing classroom devices for systems and settings. To introduce annotation systems into the classroom, dedicated devices (Nakajima, 2008) such as Clicker are required, or students need to work in a computer lab.

FWM is designed to work on students' mobile devices and is implemented as a Web application, meaning there is no need to install it on each device. Teachers can define the settings for their practices on an FWM server and then easily apply them to all students' devices.

The second problem is students' burdens of learning how to use these systems. It can result in reduced lesson time and distract from observations. Accordingly, FWM is designed to automate as many operations (e.g., saving annotations, loading the settings) as possible, apart from annotation. Moreover, the user interface for annotation is made as simple as possible.

The third problem is the lack of options available for students to make use of their annotation results. During the group reflection phase, students reflect on their practices based on all their annotation results and videos. FWM includes functions that enable students to visualize the annotation data merging results of all students and to synchronize the annotation data with the video.

In this paper, to demonstrate how the solutions for the three problems are addressed in the design of FWM, we explain how FWM can be used to observe and reflect on the educational activities in a model of presentation practice.

Based on this design, FWM is implemented as a mobile system for observing educational activities in real-time, and reflecting immediately after the activities. FWM works as a real-time video annotation system because the annotation data are premised to be synchronized with recorded video; although FWM itself does not serve to edit, play, and record video data.

To further explore the topic, this paper is structured as follows. First, in Section 2, we will describe the FWM design to resolve the three problems outlined above. Second, the implementation of FWM is then shown in Section 3. Third, Section 4 presents the results of a presentation practice in

a university to validate the effectiveness of FWM. Finally, the conclusion and opportunities for future study are considered in Section 5.

2. Design of FishWatchr Mini

2.1. Model Activity

To understand the FWM design, we will describe the presentation practice in a university as a model activity for application of FWM.

The procedure for the practice is conducted as follows. FWM could be used in Steps 2b and 3 by students and in Step 4 by teachers.

Step 1 Students are divided into groups of three to four students. The number of students as observers is assumed to be more than 10 since a smaller number of observers may not notice important scenes that should be shared in the group reflection phase.

Step 2a Each group does their presentation practice in front of other students. A teacher shoots videos for their practices.

Step 2b Students as observers make real-time annotations to scenes that they considered important. The evaluation points in observing presentations are given by the teacher before the observation (refer Section 4).

Step 3 Having complete the practice session, each group then reflects on their own practice based on all the annotation data and video data.

Step 4 The teacher gives feedback to the students based on all the annotation results and video data.

One of the characteristics of this model activity is that the procedure is the same as conventional presentation practices. In particular, the required time for the observation in Step 2b is the same because students observe the practices in real-time and use the video data following Step 2b. This makes it easier for teachers to introduce video annotation systems in the classroom.

On the other hand, there could be procedures where each group conducts their presentation practice individually, recording themselves (Step 2a') and then observes the other groups' practices through video playback (Step 2b'). The observation condition in Step 2b' is more effective than the real-time observation in Step 2b because students can observe their practices at an individual pace and as many times as they wish. The observation takes more time, however, compared to the one described above. Moreover, teachers are at risk of failing to record observations and students may not be able to conduct their observations through playback of the video data.

As described above, this model activity is concerned with facilitating the system's introduction in the classroom, allowing students to use the video data after exercises. On this basis, FWM is designed as a mobile system that allow users to conduct real-time annotation (observation), enabling them to synchronize annotation data with video data.

2.2. Solutions to the Three Problems

Here, we propose solutions to the three problems described in Section 1.

2.2.1. Preparation of Devices

FWM is designed to run on students' smartphones. According to an investigation conducted by the Japanese Ministry of Internal Affairs and Communications(2017), 96.8% of people in their 20s connect to the Internet on their smartphones. Therefore, it is feasible to use smartphones as devices for FWM in the classroom, in particular if a small number of auxiliary smartphones can be secured.

FWM is implemented as a Web application because students can use FWM through Web browsers without installing the application on their smartphones. Users can save specific settings, which are defined with a "SessionID" for each practice, to an FWM server through an FWM client. The settings can be applied to all FWM clients[1], using the SessionID. A SessionID can be set to an specific FWM client using a query parameter in the FWM server's URL. If the teacher instructs students to open the following URL, the SessionID "practice2017" are set to their FWMs, and the settings for the SessionID are then loaded on their FWMs automatically.

http://csd.ninjal.ac.jp/f/m.html&config=practice2017

2.2.2. Burden of Learning

Students' burdens of learning how to use FWM are lessened by automating as many of operations as possible apart from annotation. More specifically, these include loading specific settings for practice, saving annotation results to the FWM server, and so on.

For annotation, the operations are limited to pressing "annotation buttons," which can be defined by the teacher. One annotation has four attributes: annotation time, annotator's name, user-defined attribute 1, and user-defined attribute 2. The values for user-defined attributes 1 and 2 are assigned to the annotation buttons. Each user-defined attribute can have up to 8 values. In Figure 2, two attributes, namely presenters' name and evaluation for presentations, are defined with six buttons. The rest of the attributes, specifically annotation time and annotator's name, are automatically filled when the values of two user-defined attributes are fixed by pressing the annotation buttons.

2.2.3. Means of Using Annotation Results

In Steps 3 and 4 of the model activity, students and teachers must be able to examine annotation results made by all students and their video data. FWM provides users with three functions to facilitate this examination.

Visualizing annotation data FWM has two types of visualization methods. The first is to summarize annotation data as a histogram of the user-defined attributes. The histograms help students become aware of the characteristics used in their own presentation.

The second type of visualization is to display a histogram of annotation data in a time series. Using the histogram, each student can ascertain how the other students evaluated

[1] FWM means an FWM client, if not otherwise specified.

her/his part of the presentation, and find scenes of educational value where many students have made annotations.

Synchronizing annotation data with video data FWM provides a dedicated function for annotating a start time for the video shooting. When the teacher begins to take a shot of the practice at the same time as a button for the function is pressed, the annotation data for that SessionID is saved to the FWM server.
The start time information is used for displaying elapsed times of videos on time-series histograms. Although FWM does not include a media player, users can play a scene corresponding to a specific point on a time-series histogram, with external media players referring to the elapsed time.

Exporting annotation data to other programs FWM can export all annotation data by a SessionID including the start time information. If the annotation and video data are imported by FishWatchr (Yamaguchi, in press)[2], which is a video annotation system that we have developed, FishWatchr allows its users to edit or visualize annotations and to easily play scenes that correspond to the annotations.

3. Implementation of FishWatchr Mini

3.1. System Structure

Figure 1 illustrates the system structure of FWM. Since FWM is a Web application, users can access a URL of an FWM server (a Web server) through a Web browser. FWM clients are written in JavaScript and HTML5. Scripts on FWM servers are written in PHP.
The main functions of FWM clients are to provide a user interface for annotation and to visualize annotation data. When users finish making their annotations, the annotation results are saved to the FWM server using SessionIDs. When users employ the visualization function, the FWM client will request that the server send all annotation data with a SessionID and display histograms that are used in the reflection phase, based on the data.
FWM servers manage annotation data using SessionIDs. A set of annotation data for each SessionID is stored for 180 days on the server. By specifying a SessionID, users can download annotation data as a ZIP-compressed file, which includes all annotation data related to that SessionID. Note that anybody who knows the SessionID can download the annotation data for that SessionID.
The video camera does not need to be controlled by FWM clients or a FWM server because the annotation data is synchronized with the video data, based on the time information related to these data. Therefore, generic video cameras such as video cameras for home use and smartphones can be used to record the video.
FWM clients themselves have no function to play and edit video data. If FWM is used in the reflection phase: Step 3 and 4, users need an another media player on which to play the video data, referring to the time information displayed on time-series histograms, as described in Section 3.3.. Another option is to use video annotation systems such as FishWatchr, which can import the annotation data of FWM and the video data, and then synchronize them.

[2]https://github.com/himawari-san/FishWatchr

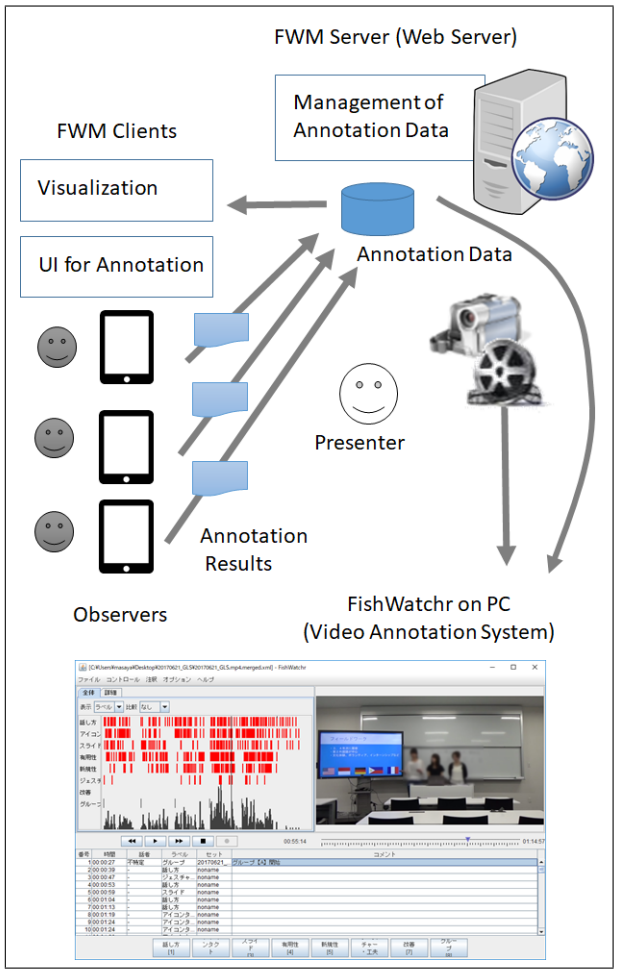


Figure 1: System Structure of FWM

3.2. Annotation

The annotation screen of FWM is shown in Figure 2. The four annotation buttons in the left-hand column and three buttons in the right-hand column are for speaker attributes and evaluation attributes respectively. Button settings can be configured for each SessionID on FWM.
One annotation is recorded after the two user-defined attributes are fixed by pressing the buttons. The two most recent annotation results are listed below the annotation buttons and can be deleted by pressing x-mark buttons beside the annotation results, when necessary.
When the "Finish" button in Figure 2 is pressed, the annotation phase is complete and the annotation data are stored automatically in the FWM server (in the FWM client optionally). Students can thus make annotations using these simple operations.

3.3. Visualization

FWM provides the opportunity for its users to visualize annotation data in two viewing styles. Since the visualization does not depend on video data, users can use this function immediately after practices.
Figure 3 includes two examples of a time-series histogram. The upper figure draws a histogram of practices in five groups. The lower figure is focused on the practice of Group 4 by changing the time interval (60 to 15 sec) and the

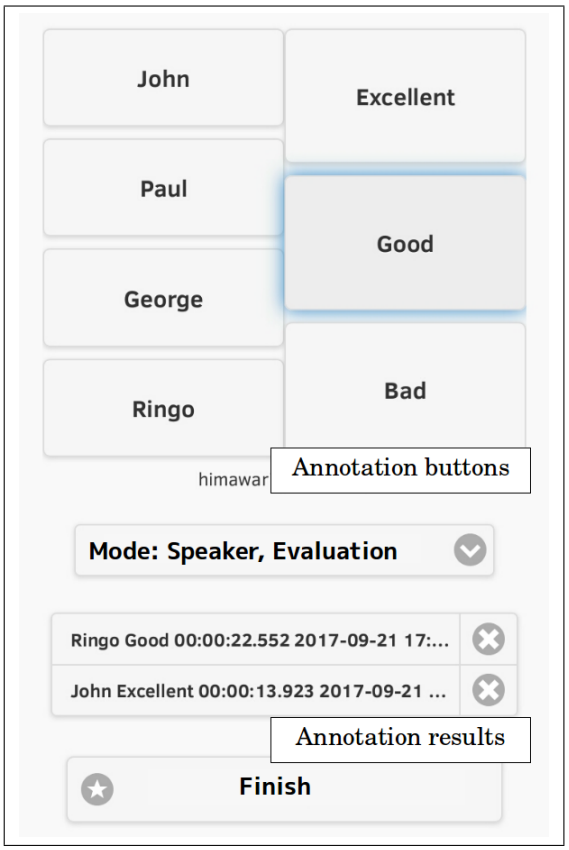


Figure 2: Annotation Screen of FWM

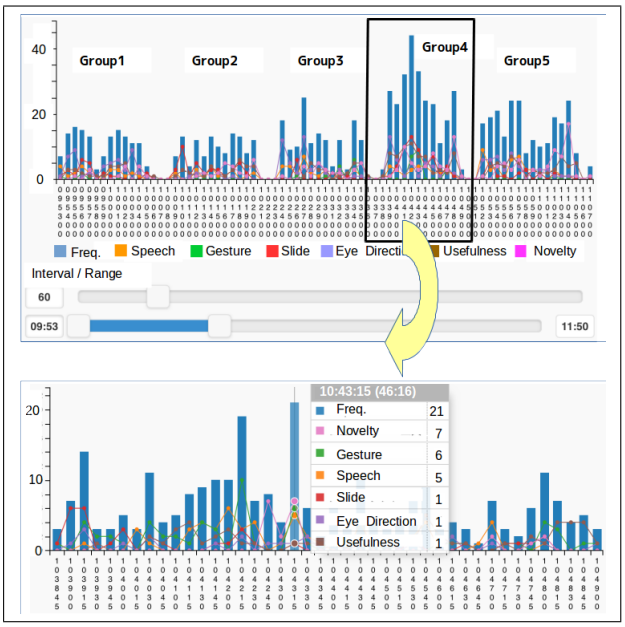


Figure 3: An Example of a Time-Series Histogram

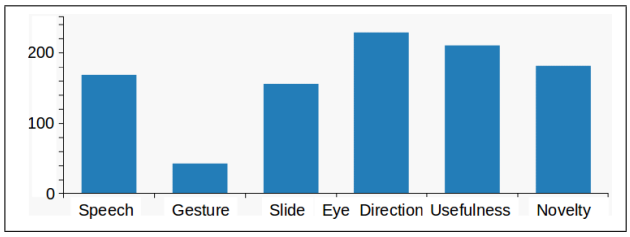


Figure 4: An Example of a Summary Histogram

time range (09:53-11:50 to 10:38-10:49) using the slides. The wide range view allows students to compare their own group with other groups. The narrow range view helps students examine a specific scene. When a bar of histograms is clicked, the detailed information is shown in a pop-up window that includes the annotated time (10:43:15), the elapsed time (46:16) of the video data, and the breakdown of user-defined attributes of annotations (in this figure, values of "evaluation" attribute are shown).

Figure 4 is an example of a summary histogram. This histogram draws annotation results based on the "evaluation" attribute. Since the time range can be changed as with time-series histograms, users can compare a summary histogram of all groups with a summary histogram of a specific group. Summary histograms can also be shown through specifying a certain observer, typically the observer himself/herself. These histograms help students as observers to compare other observers' annotation results. Figure 5 demonstrates an example of an observer-specified summary histogram of "evaluation" attributes. The left and right bars of each attribute refer to the number of annotations by the specified observer and the average number of annotations by observers respectively. Figure 5 shows that the specified observer made more annotations for speech and gesture, but less annotations for eye direction and usefulness compared to the other observers.

3.4. Data Coordination with Other Equipment

As illustrated in Figure 1, FWM works with various equipment. This section describes the data coordination between FWM, video cameras, and the video annotation system FishWatchr.

3.4.1. Video Camera

To synchronize annotation data with video data, the time information of annotation data and the start time information of recording the video are used. Moreover the start time information could be annotated with FWM clients. Devices where FWM clients work are therefore required to be set to the same time.

In our exercises, all devices, such as students' smartphones

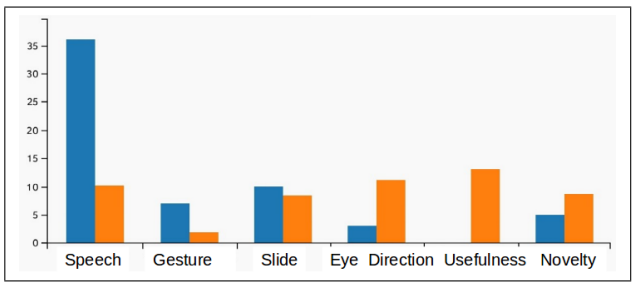


Figure 5: An Example of an Observer-Specified Summary Histogram

and PCs, were configured to use network clocks and teachers instructed students to compare them to the time provided by the Japan standard time site[3].

3.4.2. FishWatchr

With regard to importing to FishWatchr, all that users have to do is to collect downloaded annotation data files (XML format) and the video file in a folder, and to then drag&drop the folder to FishWatchr.

Two methods are provided to synchronize annotation data and video data. One method is to annotate a start time for the video shooting. If a teacher does the annotation, students can import these data without using special operations for synchronization.

The other method is to set the start time information manually during the import process. A way of determining the start time is to shoot a precise clock at the beginning. Even if the first method is used, the clock shooting is recommended in case users fail to annotate the start time.

Annotation data files seems to be simple to use with other software since they are available as tab-delimited text or as simple XML files. The following example is a "comment" element in XML-formatted annotation data file, which describes one annotation.

```
<comment date="19x8-10-12 21:22:30.232"
         commenter="GeorgeM."
         target="John"
         label="Excellent" />
```

4. Introduction to Presentation Practice

4.1. Overview and Conditions

To confirm the effectiveness of FWM, we introduced FWM to presentation practices in a university based on the model activity, apart from in Step 4 where the teacher used neither FWM nor FW.

It took 90 minutes (one class period) to complete Steps 1-2b and 90 minutes for Steps 3-4. Steps 3-4 were conducted one week after Step 1-2b. FWM and FW were used in Steps 2b and Step 3 respectively. The total training time required for students to use FW and FWM was approximately 30 minutes.

In Step 1, fourteen students were divided into five groups. Each group had 10 minutes to complete the practice with a theme that they had set.

The teacher defined six evaluation points: speech, gesture, slide, eye direction, usefulness, and novelty. These points are assigned to the values of a user-defined attribute of FWM. In this activity, the speaker's name was not used as a user-defined attribute because only one student was responsible for one part of the presentation and spoke at a time. Students pressed the annotation buttons when they found good relevant scenes based on the six evaluation points.

In Step 3, each student imported the annotation data, which was exported from an FWM server, and the video data to FW and then reflected on the annotated scenes in which he/she or members of his/her group presented. The teacher distributed the annotation data to students and the video data on USB flash drives. The total data size was about 1.4GB.

4.2. Results

All the practices were completed without any significant problems. A faster distribution method for annotation data and video data may be required, however, if the interval between Step 2b and Step 3 is shorter, for example if the class element of Step 3 is complete immediately after that of Step 2b.

During the practices, students made 838 annotations. The median, maximum, and minimum values of the annotations are 52, 150, and 22 respectively. The distributions of the annotations are shown in Figure 3 and 4. Table 1 displays the results of the questionnaires conducted after all the practices had been completed, which use a 5-point scale[4]. The scores in the table represent the mean scores.

Question	Score
1. Could you observe practices with FWM smoothly?	4.29
2. Were the evaluation points sufficient for evaluating the presentations?	4.36
3. Was it helpful for you to know when your group's practice was evaluated positively in correspondence with the video?	4.21
4. Were you helped to find problems in your presentation by reflecting on scenes where other members of your group were evaluated positively?	4.00

Table 1: Results of Questionnaires

Here, we evaluated this study based on the following two points of view.

Ease of use of FWM Despite the 30-minute training time, we did not observe any students facing difficulties in using FWM while they were observing the presentations. This ease of use is further supported by the scores in Question 1 (4.29) and by the high minimum number of annotations (22).

Applicability of Annotation Results The result of Question 3 in Table 1 shows that annotations against students' own presentations were helpful in that they could recognize what kinds of actions were evaluated positively. From the result of Question 4, it is clear that students became aware of problems in their own presentation by observing others' good practice.

These results validated that FWM and FW worked well in the presentation practices.

5. Conclusion

In this paper, we developed an observation support system for students called FWM. FWM was designed to provide the following three solutions for introducing annotation systems effectively in the classroom: (a) FWM was

[3] https://www.nict.go.jp/JST/JST5.html

[4] Five is the best.

implemented as a Web application run on students' smartphones; (b) many operations, apart from annotation, were automated to lessen Students' burdens of learning how to use FWM; and (c) three functions, namely visualization, synchronization, and exportation, were provided so that the annotation results could be used in the reflection phase. The results of applying FWM to the presentation practices proved the effectiveness of FWM in terms of its ease of use and the applicability of annotation results.

The annotated video data collected through our model activity can provide a helpful language resource, not only for students, but also for teachers. Future study will examine how we can improve subsequent presentation practices and its instruction by using annotated video data.

6. Acknowledgments

I would like to thank Professor Atsushi Mori for his helpful suggestions. This work was supported by JSPS KAKENHI Grant Number JP17K01105.

7. Bibliographical References

Barkley, E. F., Cross, K. P., and Major, C. H. (2005). *Collaborative Learning Techniques: A handbook for college faculty*. John Wiley & Sons, Inc.

Brugman, H. and Russel, A. (2004). Annotating multimedia/multi-modal resources with ELAN. In *Proceedings of LREC 2004*.

Colasante, M. (2011). Using video annotation to reflect on and evaluate physical education pre-service teaching practice. *Australasian Journal of Educational Technology*, 27(1):66–88.

Kong, S. C., Shroff, R. H., and Hung, H. K. (2009). A web enabled video system for self reflection by student teachers using a guiding framework. *Australasian Journal of Educational Technology*, 25(4):544–558.

Ministry of Internal Affairs and Communications, Japan. (2017). Investigations of usage time and information activities of information-communication media. http://www.soumu.go.jp/main_content/000492875.pdf.

Morimoto, I. and Otsuka, H. (2012). *Educational Research and Development at a Workshop-based Education Program Fostering Students' Self-directing Communication Abilities*. Nakanishiya Publishing.

Nakajima, T. (2008). Supporting for teaching improvement by integrating video recording with real-time feedback through response analyzers. *Japan Journal of Educational Technology*, 32(2):169–179.

Rich, P. and Hannafin, M. (2009). Video annotation tools: Technologies to scaffold, structure, and transform teacher reflection. *Journal of teacher education*, 60(1):52–67.

Yamaguchi, M. (in press). A video annotation system for learners to observe educational activities. In Motoko Ueyama et al., editors, *Digital Resources for Learning Japanese*. Bononia University Press, Bologna.

Yousef, A. M. F., Chatti, M. A., and Schroeder, U. (2014). The state of video-based learning: A review and future perspectives. *International Journal on Advances in Life Sciences*, 6(3):122–135.

The AnnCor CHILDES Treebank

Jan Odijk,Alexis Dimitriadis,Martijn van der Klis,Marjo van Koppen,Meie Otten,Remco van der Veen
Utrecht University
Trans 10 3512 JK Utrecht
{j.odijk, a.dimitriadis, m.h.vanderklis, j.m.vankoppen, m.otten1, r.p.vanderveen}@uu.nl

Abstract

This paper (1) presents the first partially manually verified treebank for Dutch CHILDES corpora, the AnnCor CHILDES Treebank; (2) argues explicitly that it is useful to assign adult grammar syntactic structures to utterances of children who are still in the process of acquiring the language; (3) argues that human annotation and automatic checks on this annotation must go hand in hand; (4) argues that explicit annotation guidelines and conventions must be developed and adhered to and emphasises consistency of the annotations as an important desirable property for annotations. It also describes the tools used for annotation and automated checks on edited syntactic structures, as well as extensions to an existing treebank query application (GrETEL) and the multiple formats in which the resources will be made available.

Keywords: treebank, Dutch, CHILDES, GrETEL, treebank querying

1. Introduction

We describe the approach to the development of the AnnCor CHILDES Treebank, a treebank for Dutch CHILDES corpora. The whole treebank has been automatically generated by the Alpino parser, but a part of this corpus has also been manually checked and, where needed, corrected. The AnnCor treebank is being created in the Utrecht University AnnCor project, which we describe in section 2..[1] We describe the syntactic annotation in this treebank in section 3.: we discuss some methodological issues with regard to the annotation (section 3.1.), the cleaning of CHILDES utterances (section 3.2.), the tool we use to inspect and edit syntactic structures (section 3.3.), annotation conventions we developed (section 3.4.), and checks on edited syntactic structures (section 3.5.). We have extended an existing treebank query application, and the extensions are described in section 4.. In section 5. we describe the formats in which the resources will be made available. In section 6. we discuss related work, and we end with conclusions and plans and suggestions for future work in section 7..

This paper (1) presents the AnnCor CHILDES Treebank, the first partially manually verified treebank for Dutch CHILDES corpora; (2) argues explicitly that it is useful to assign adult grammar syntactic structures to utterances of children who are still in the process of acquiring the language; (3) argues that human annotation and automatic checks on this annotation must go hand in hand to combine the human's intelligence and the software's rigor; (4) argues that explicit annotation guidelines and conventions must be developed and adhered to. Requiring consistency of the annotations is important and should be an essential ingredient of such annotation conventions.

[1] This paper contains many hyperlinks hidden under terms and acronyms. The presence of a hyperlink is visible in digital versions of the paper but may be badly visible or invisible in printed versions of the paper.

2. The AnnCor Project

The AnnCor project[2] is an Utrecht University internal research infrastructure project that aims to create linguistically annotated corpora for the Dutch language and to enhance and extend an existing treebank query application in order to query the annotated corpora. Various types of corpora are being annotated, and various types of annotations are added. The corpora include learner corpora (texts produced by pupils at primary school), news corpora, narrative corpora, and language acquisition corpora (in particular, natural spoken interactions between parents and children). Annotations include annotations for learners' errors and their corrections, discourse annotations, and full syntactic structures. In this paper we focus on the creation of treebanks (i.e. text corpora in which each utterance is assigned a syntactic structure) for language acquisition data, in particular the Dutch CHILDES corpora (MacWhinney, 2000).[3] The CHILDES corpora contain annotated orthographic transcriptions of the interaction between multiple speakers, usually a target child and other participants (e.g the child's mother).

According to (Sagae et al., 2007), 'linguistic annotation of the corpora provides researchers with better means for exploring the development of grammatical constructions and their usage'. The research described in (Odijk, 2015; Odijk, 2016a) illustrates this for the study of the acquisition of particular syntactic modification and complementation phenomena using the Dutch CHILDES corpora. It is clear from these papers that such research cannot be done properly and efficiently without treebanks for these corpora. This can be illustrated with a simple example. If one wants to study the behavior of Dutch words such as *heel*, *erg* and *zeer* (all meaning 'very'), it is not sufficient to search for these strings: one will miss inflected variants such as *hele* and *erge* and find irrelevant occurrences of these words in different uses and meanings (such as *zeer* meaning 'pain'

[2] https://anncor.sites.uu.nl/.

[3] Accessible via http://childes.talkbank.org/access/Dutch/.

or 'painful'). Significantly better search results are obtained when one can search in a treebank containing these words. Ambiguity is not restricted to these words: very many words are highly ambiguous, in particular the words that linguists are most interested in. Many of these ambiguities are resolved in treebanks. The AnnCor project aims to create exactly such treebanks, which, together with query applications, will become an integrated part of the Dutch part of the CLARIN research infrastructure (Odijk, 2016b; Odijk and van Hessen, 2017). These treebanks and the associated search and analysis applications can then contribute to an acceleration of language acquisition research and to a larger empirical basis for theories or hypotheses, thus providing a basis for carrying out groundbreaking research in which old questions can be investigated in new ways and new questions can be raised and investigated for the first time.

3. Syntactic Annotation

The syntactic annotation is added using the Alpino parser (Bouma et al., 2001). Since Alpino has been developed for written adult language such as newspapers, it is not surprising that it creates many wrong parses. The problem is twofold: CHILDES contains transcriptions of spoken utterances from a dialogue, and many of them are uttered by children that are still in the process of acquiring the language. The fully automatically generated parses can be inspected and queried in the PaQu application (Odijk et al., 2017).[4] In the AnnCor project we create a manually verified subcorpus, with a targeted size of 20% of the Dutch CHILDES *Van Kampen* subcorpus, sampled in a representative manner by selecting a contiguous subpart of each session of the subcorpus.[5] In addition, we manually verified and, if needed, corrected the parse trees for which it was very likely that they contain errors, as determined on the basis of a variety of heuristics for identifying potential errors. For example, syntactic structures containing nodes labeled as *dp* (discourse part) are often the result of the Alpino robustness module which is used when Alpino is not able to make a single full parse using its normal rules. This may be caused by fragmentary or ungrammatical input, or by parser errors.[6] Another example concerns coordinate structures, which often contain errors. All together we target to have 35% of the corpus manually verified.

3.1. Methodological Considerations

The CHILDES corpora contain orthographic transcriptions of the interaction between a target child and other participants. The latter are usually a parent, caretaker, or investigator, but may include others, among which other children. Parsing the utterances of adult speakers with a parser for a grammar which is supposed to reflect the competence of adult native speakers is unproblematic from a methodological point of view. However, it is not obvious that it makes sense to parse the children's utterances with a parser for a grammar which is supposed to reflect the competence of adult native speakers. Of course, we do not know what internal grammar the children have (and we have no model for it), and this internal grammar changes over time. In fact, one of the goals of creating the AnnCor Treebank is to enable researchers to gain knowledge about the internal grammar of children. The parses of children's utterances must not be seen as claims about the syntactic structure assigned by the children's grammar but as a classification of children's utterances and their parts in terms of the (best implemented model that we have for the) grammar for adults. We assume that children converge on a grammar structured like the grammar for adults and actually almost identical to it. Therefore it makes sense to classify children's utterances in terms of the adults' grammar, so that we can compare children's and adults' utterances.

Children often make utterances that are not well-formed according to the grammar for adults. Often these utterances appear to reflect a different internal grammar or performance factors typical for children (such as a more limited memory, less developed pronunciation abilities, etc). Such utterances pose problems for the creation of the treebank. Automatically assigned syntactic structures are most likely incorrect, and require manual correction. It is not always a priori evident what these corrections should look like. Therefore explicit conventions and guidelines have to be developed on what syntactic structure should be assigned to such utterances. These conventions and guidelines must be set up in such a way that they are maximally useful for research into language acquisition by children. We discuss some of these annotation conventions and guidelines in section 3.4..

3.2. Cleaning

CHILDES utterances are enriched with all kinds of annotations. An extensive description of these annotations can be found in (MacWhinney, 2015a). Many of these annotations are in-line annotations. Some examples are given in (1):[7]

(1) Example in-line annotations in CHILDES CHAT files:

a. < ik wi > [//] ik wil xxx bekertje doen.
< I wan > [//] I want xxx cup-DIM do
'I want to do the little cup'

b. < doe maar even > [/] doe maar even op tafel.
< put PRT PRT > [/] put PRT PRT on table
'Just put on the table'

c. knor knor [=! pig sound] , ik heb honger.
oink oink [=! pig sound] , I have hunger
'Oink oink, I am hungry'

These examples illustrate annotations for retracing ([//])and repetition ([/]), both with scope over the preceding part be-

[4] http://portal.clarin.nl/node/4182.

[5] The Van Kampen subcorpus contains 109k utterances with 526k tokens distributed over two target children, with monthly recordings for these children from their first through their fifth year.

[6] Gertjan van Noord, personal communication.

[7] The sources are indicated by the session name (e.g. Sarah35) followed by the utterance number (e.g. 224), starting counting at 1. Utterances Sarah35.015, Sarah35.023 and Sarah35.224 from the *Van Kampen* corpus.

tween angled brackets, for unintelligible material (xxx) and for paralinguistic material ([=! ...]).[8]
The Alpino parser cannot deal with these annotations. A cleaning programme has been developed to remove the annotations and send a cleaned utterance to the Alpino parser.[9]
The cleaned variants of the utterances in (1) are:

(2) Example cleaned utterances:
   a. ik wil xxx bekertje doen.
   b. doe maar even op tafel.
   c. knor knor , ik heb honger.

It is not always obvious how this cleaning should be done, and we have experimented with several variants, for example in an earlier variant we removed the xxx markings. However, this often led to clearly undesirable parses. Alpino analyses xxx as an unknown word, and assigns it a part of speech depending on the context (most often: noun), which is often correct or at least plausible. Even when it is wrong, the overall parse is generally easier to correct when xxx is present than when it is absent.
The cleaning program is available on GitHub[10] and has been integrated in the GrETEL upgrade described in section 4..

3.3. Editing syntactic structures

We use TrEd 2.0[11] as the editor for inspecting and correcting the syntactic structures (Pajas and Štěpánek, 2004). There already existed a TrEd extension[12] for syntactic structures in the format generated by Alpino, but it was only compatible with an older version of TrEd. We updated it to work with the current TrEd version, and created an 'extension repository'[13] to allow it to be installed from the TrEd 2.0 plug-in API. The editor provides an intuitively appealing graphical interface for inspecting and manipulating the syntactic structures, and is a desktop application, which works on multiple platforms (Linux, Windows, MacOS and several UNIX-based systems). Our experience is that most web-based interfaces are inferior to desktop interfaces, because of the requirement to be on-line, often unpredictable scrolling and cursor behavior and more limited options for keyboard short cuts. Therefore we did not employ a web-based tree editor. In addition, earlier projects that created treebanks for the Dutch language (in particular, the LASSY project (van Noord et al., 2013)) used TrEd so we could benefit from the experience with working with this tool gained there. Furthermore, the main platform our annotators work on is Windows, which restricts the options. At the time we had to select a treebank editor (early 2015), there were not many alternatives to TrEd. See the Exmeralda Linguistic Annotation Wiki for an overview of annotation tools and related matters.
MPI's Synpathy could not be installed, and there is no support or further development. The EXMARaLDA Sextant `http://exmaralda.org/en/sextant-en/` tool (Wörner, 2009) was not available in a stable version.[14] *WebAnno* Version 2 (Yimam et al., 2014) is an annotation tool that is web-based and this version appears not particularly suited for annotating syntactic structures, though its successor version 3 (Eckart de Castilho et al., 2016) might be worth investigating further. *Atomic* is a new annotation editor (a desktop application) but claimed by the developers not to be stable yet (Druskat et al., 2014). The @nnotate tool appears to offer the required functionality but does not work on the Windows platform. Arborator deals with dependency relations in CONNL format only and just visualises edits in a textual CONLL file. The FLAT tool (van Gompel et al., 2017) was not ready for annotation of syntactic structures in 2015 and is a web application.

3.4. Annotation Conventions

Utterances from spoken language can contain many performance phenomena and errors for which it is not obvious how they should be analysed syntactically. The utterances used by the children contain many phenomena that are not part of the adult language. In addition, as in any annotated corpus, many phenomena can be analysed in multiple ways, none of which can be considered better than any other on purely linguistic grounds. It is important to analyse each construction in a consistent and uniform manner, so that it can be easily automatically identified and distinguished from other constructions in a treebank query application when the data are used in research. For this reason, it is important to develop and adhere to annotation conventions and guidelines.
For utterances made by adults we adhere to the annotation guidelines developed in the LASSY (van Noord et al., 2011) and Spoken Dutch Corpus Projects (Hoekstra et al., 2003), wherever applicable. For phenomena not covered there, we developed new annotation conventions. We will illustrate these with some examples.
As is well known, spoken language often contains incoherently structured utterances, with rephrasings, unfinished sentences, or just mispronunciations, all of which prove difficult for the Alpino parser to handle. Note that these problems are not limited to child speech, but also frequently occur in adult spoken language, which is often produced on the fly. In example (3), such phenomena are illustrated:[15]

[8] The glosses and translations given for these examples are not included in the CHILDES databases.

[9] Of course, removing these annotations makes it impossible to do research on these phenomena in combination with the syntactic structures. Therefore, we will translate these annotations into metadata in a new version of the cleaning program. See section 7..

[10] `https://github.com/JanOdijk/chamd`.

[11] TrEd is an abbreviation for Tree Editor. See `https://ufal.mff.cuni.cz/tred/`.

[12] `https://bitbucket.org/alpino/alpino`.

[13] `http://languagelink.let.uu.nl/~alexis/tred/extensions/alpino/`.

[14] We quote from the website: 'The development of the tool and the respective components is a "work in progres". This means: it is not guaranteed that the functions displayed in the software, or described elsewhere, will work. Some parts of this software and the schemas and models that underlie them, may change anytime. Therefore, no guarantee regarding the integrity of the data that will be edited with this tool, can be given.'

[15] LAURA28.264

(3) eh , wangetjes eh eve een keer , eve
um , cheek.DIM.PL um briefly a time , briefly
olie op wangetjes.
oil on cheek.DIM.PL
'Um, cheeks, um, probably once, briefly oil on cheeks.'

Since the structure in this sentence is chaotic, featuring filled pauses (*eh* 'um') which interrupt the flow of words, Alpino is unable to properly analyse the utterance. The lack of a verb to head the sentence makes it impossible to discover any reasonable analysis. Alpino also cannot distinguish the false start from the core part. We analyse the first part (until *een keer*) as a false start, which is filled with a sequence of fragments. The second part holds all the meaning, even on its own. We analyse such utterances as consisting of two parts. The meaningful part becomes the NUCL, or the nucleus, whereas the false start becomes the SAT, or the satellite.
An example of utterances that are not part of the adult language involves determiners. In acquiring adult language it is imminent that children eventually learn that some nouns combine with the article *de* 'the-UTR'[16] and others with the article *het* 'the-NEUT'. However, in early stages of acquisition children often do not use the different determiners correctly. Example (4)[17] shows such an utterance:[18]

(4) in de bad , he
in the-UTR bath , PRT
'in bath, right?'

Though it is clear that in this example *de bad* should be analysed as a noun phrase despite the gender mismatch, Alpino cannot do this. Since many researchers will want to know about determiner use by children acquiring language, it is important to manually correct such cases.
The following examples appear to contain a finite verb form (*lees* and *kocht*, respectively) where a participle is expected:[19]

(5) a. ik heb niet lees
I have not read-PRES
'I have not read'
b. Ik heb bolletjes kocht
I have roll-DIM-PL buy-PAST
'I have bought little rolls'

It is not a priori clear how such examples should be analysed: the child might be producing forms that do not conform to the adult language due to syntactic reasons, morphological reasons or phonological reasons. Each of these causes would imply a different analysis, but only after an intensive investigation of each phenomenon can one decide among them . In constructing the treebank we do not take a stand as to how such examples should be analysed, but we do treat each of them in a uniform way, so that each can be easily and automatically identified by researchers using a treebank query application. The examples in (5) are analysed in the treebank as participial verbal complements (*vc/ppart*) that contain a finite verb.[20]
A lot of utterances consist of an infinitive or perfect participle and its complements. Such a construction is not part of the adult language as a main clause. As one might expect, Alpino analyses them incorrectly. It occurs both with an overt subject and without. Without an overt subject it can occur as an infinitival or participial complement to other verbs in the adult language.

(6) a. ikke pap eten (Laura09.527)
I-emph porridge eat-INF
'I eat porridge'
b. en die maken (Laura13.042)
and that make-INF
'and make that one'
c. die weggelopen (Sarah10.024)
that away-walk-VD
'that one (has) run away'

In the AnnCor corpus we analyse such examples uniformly as infinitival (or participial) main clauses.
A spoken language effect that is difficult to capture by Alpino involves contractions. An example of such a contraction can be found in example (7):[21]

(7) Nee, das een andere, van de uiln !
No, that's a different, from the owls !
'No, that is a different one, from the owls!'

This example actually illustrates multiple problems. In spoken language the two words *dat is* 'that is' are sometimes pronounced as a contracted form *das*. It is orthographically incorrect to write this contraction as *das*. Therefore, Alpino cannot deal with it. A second problem is that the word *das* is a correct word of Dutch, so Alpino tries to assign a structure to the sentence in which this word is analysed as a noun (meaning 'tie' or 'badger'). In addition, the transcription in the CHILDES corpus violates the CHILDES transcription rules. In accordance with these rules (MacWhinney, 2015a, 47), this example of contraction should have been transcribed as *da(t) (i)s*. With such a transcription and the cleaning program (see section 3.2.) there would have been no problem for Alpino. We correct this by splitting up the contraction, though for such cases it would have been preferable to adapt the transcription and have the corrected transcription reparsed by Alpino.
The AnnCor documentation (Otten et al., 2018) describes these and many other annotation conventions in detail.

[16] UTR = uter, i.e. non-neuter, and NEUT = neuter.
[17] LAURA28.201
[18] The fact that this utterance is not a full sentence is in itself not problematic for Alpino.
[19] Utterances Laura09.527 and Laura13.042 from the *Van Kampen* Corpus.
[20] Assigning a structure to such utterances differing from what Alpino assigns to them reduces the options for example-based search, which requires parsing by Alpino (see section 4.), so such examples will have to be searched by writing XPath queries or by adapting XPath queries generated by example-based querying.
[21] LAURA70.71

3.5. Checks on the Annotation

In general, about 10% of each batch of manually corrected sentences were checked by a second annotator. At the beginning of the project, and when a new annotator started, all output was double-checked.

We carried out an initial small experiment to determine interannotator agreement for an independently annotated sample, by comparing all combinations of the annotations of two annotators, resulting in an average F-score of 0.86 (6 annotator pairs, random sample of 100 utterances). We used F-score, since it is not clear how a metric such as Cohen's κ or related scores can be applied (since there is not really a fixed number of categories for classification). Though the F-score of .86 is reassuring, a test with a larger sample is desirable and planned.

The annotators have excellent knowledge of the language and its syntactic structures and this enables them to make such corrections. However, they are human and therefore will very likely make errors due to lack of attention, oversights, etc. This is especially so because the annotations can be very complex and a lot of small details have to be attended to. We try to avoid such errors as much as possible in a number of ways. First, the TrEd tool avoids potential errors by providing fixed dropdown lists for fixed ranges of values (part of speech codes, morpho-syntactic features, etc.). Second, we developed a new tool, called the AnnCor Check Engine (ACE), which checks properties of the syntactic structures. We provide some examples of such checks:

- certain grammatical relations can occur only once in a local tree (i.e., a parent node and its children nodes). For example, the grammatical relation *su* (for subject) can occur only once. If an annotator accidentally violated this constraint while editing a tree, ACE issues an error message.
- In earlier Dutch treebank creation projects (Spoken Dutch Corpus (Oostdijk et al., 2002) and LASSY (van Noord et al., 2013)), it was decided that syntactic structures should not contain unary branching nodes, and the Alpino output follows this policy. Thus in a sentence such as *he swims* the pronoun *he* is not dominated by an NP node (which would lead to unary branching) but is immediately dominated by the clausal node (and it bears the grammatical relation *su* for subject). ACE warns against violations against this ban on unary branching.
- The syntactic structures contain some redundancies. For example, past participle and infinitival complements have different labels: *ppart* and *inf* respectively. But a verb heading such a phrase has morphosyntactic features specifying whether it is a participle (*wvorm=vd*) or an infinitive (*wvorm=inf*). A phrase labeled *ppart* should contain a head verb that is a participle, and a phrase labeled *inf* should contain a head verb that is an infinitive. ACE warns against violations of this rule.
- We collected statistics on local trees from the Spoken Dutch Corpus and LASSY treebanks in a database. We assigned a score to each local tree, in principle equal to its frequency. If a syntactic structure contains a local tree configuration with a score equal to 0, ACE issues an error message. If the local tree configuration has a score below a certain threshold,[22] ACE issues a warning. Of course, results obtained in the past do not give guarantees for the future. For this reason, we set up a tuning phase in which we manually added legal configurations that happened not to occur in the earlier treebanks but did occur in the AnnCor CHILDES treebank. We also adapted the scores for well-formed configurations that occurred in the earlier treebanks with a frequency below the set threshold.

The tool currently checks for 34 different potential errors and its local tree configuration database contains several thousand legal local configurations. The tool may occasionally give incorrect error or warning messages. In such a case the annotators can mark this error or warning message as incorrect for this instance. This ensures that the message will not be issued for this instance again when the syntactic structure is checked in a later stage or by a different annotator. The functionality to mark certain phenomena as exceptions is surely needed when a syntactic structure has to be assigned to an utterance that deviates from what is allowed in adult Dutch, such as in the examples in (5), in which a finite verb heads an infinitival or participial complement.

4. Treebank Querying

In addition to creating the treebank, the Anncor project developed an application that can be used to explore the corpus.

We extended the existing treebank query application *GrETEL* (Version 3) developed in Leuven (Augustinus et al., 2012),[23] adding the possibility to upload one's own corpora and associated metadata, as well as functionality to analyse and filter on data and metadata in *GrETEL* Version 4.[24]

GrETEL is a web application that allows researchers to search in Dutch treebanks and to perform a limited analysis of the search results. It has a very user-friendly example-based interface, but also allows queries in the XML query language XPath.

The corpus upload functionality allows users to upload an archived collection of plain-text files. The software will tokenise and parse these files using the Alpino parser, and import them into the XML database BaseX (Grün, 2010) for querying with GrETEL. Users can specify their corpus as private (only searchable for them) or publicly available. Next to plain text input, input in the CHAT format is possible as well. In this case, the software uses, inter alia, the cleaning algorithm described in section 3.2.. One can also upload a treebank, i.e. a corpus in which each utterances has been assigned a syntactic structure that is compatible with the Alpino output format. Such syntactic structures can have been generated fully automatically, or be the result of manual annotation. Work is currently ongoing to

[22] set, after some experimentation, to 10.

[23] `http://gretel.ccl.kuleuven.be/gretel3/index.php`.

[24] `http://gretel.hum.uu.nl/gretel4/`.

provide a wider range of input formats (in particular, FoLiA (van Gompel and Reynaert, 2013) and TEI).[25]

For adding metadata to corpora, we use a format defined during the development of PaQu, which allows users to add metadata in the running text (see `http://zardoz.service.rug.nl:8067/info.html#cormeta` for details). The software reads in the metadata and will create faceted search in GrETEL to allow users to both analyse and filter their search results. Users can change the facets to their liking, e.g. to use a range filter instead of checkboxes for numeric metadata. After finding a result set of interest, this set can be further analysed in an analysis interface. This interface enables the creation of pivot tables and graphs, which allows rapid insight into the data. The result set can also be exported to a tab-separated value text format to allow further analysis in other tools.

We will illustrate GrETEL 4 by the example that we also used in (Odijk et al., 2018), but now applied to the CHILDES subcorpus *Van Kampen Sarah*. We are interested in utterances by young children containing three bare[26] verbs. GrETEL offers query by example (QBE) functionality, in which a sentence containing the desired construction is entered, parsed and used as the basis for a query. We use this function with the example from (8), in which the three bare verbs are in boldface:

(8) hij **zal** dat **willen doen**
he will that want do
'he will want to do that'

This example sentence is now parsed by Alpino, and it parse result is shown. In order to have this example turned into a query we have to specify that the subject *hij* is not relevant for the construction we are interested in (the example contains it because Dutch sentences of this type must have a subject). Neither is the direct object *dat* (the example contains it because the verb *doen* is a transitive verb). The verbs must of course be included, but not these specific verbs: any verb that can occur in this construction will do. We therefore only require that they are verbs. The example is a main clause, but we also want to find examples in subordinate clauses. Therefore we specify that the properties of the top node of the parse tree containing the selected elements (*smain*, i.e. main clause) should be ignored. GrETEL offers a graphical interface to make such selections, which is illustrated in Figure 1.

This selection results in the XPath query (9):

(9)
```
//node[@cat and
       node[@rel="hd" and @pt="ww"] and
       node[@cat="inf" and @rel="vc" and
            node[@pt="ww" and @rel="hd"] and
            node[@rel="vc" and @cat="inf" and
                 node[@rel="hd" and @pt="ww"]]]]
```

which can be represented graphically as the query tree (10):

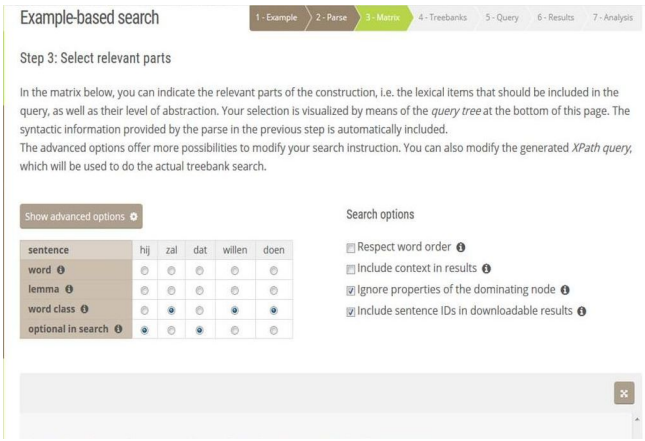
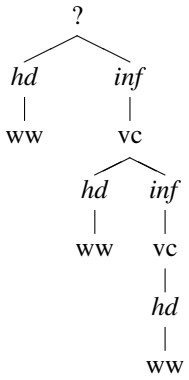


Figure 1: Selection of the construction elements.

(10)


We run the query on the *Van Kampen Sarah* subcorpus[27] (*vksarah* in Gretel) and get 147 hits. We can now analyse the search results in terms of data (the elements that match nodes in the query tree) and metadata such as speaker, age, role, etc.). We illustrate the analysis page in Figure 2, which specifies the frequency of the most superordinate verbs used by Sarah. The application allows many aspects of the results to be counted and tabulated. For example, 18 utterances have Sarah (code SAR) as speaker, 129 are by the mother Jacqueline (code JAC). The child uses the construction already at the age of 28 months, but it occurs only sporadically until month 46, after which its frequency increases. 14 of the utterances by the child contain triples of verbs that are not in the input provided by the mother (in this, admittedly small, sample), and of the ones that do occur in the mother's input only one belongs to the triples frequently used by the mother (*zullen gaan doen* 'will go do'). These findings are consistent with the findings for the CHILDES subcorpus *Van Kampen Laura* reported in (Odijk et al., 2018), though Sarah starts using the construction earlier than Laura.

We refer to (Odijk et al., 2018) for many more details on the analysis options offered by GrETEL 4.

5. Updated CHAT files

We will make the parsed data available as downloadable files and as part of the GrETEL application. We will also provide the treebank in the CHAT format, in the MOR and

[25] `http://www.tei-c.org/`.

[26] i.e without *te*, cf. English *to*

[27] This corpus contains 44,869 utterances

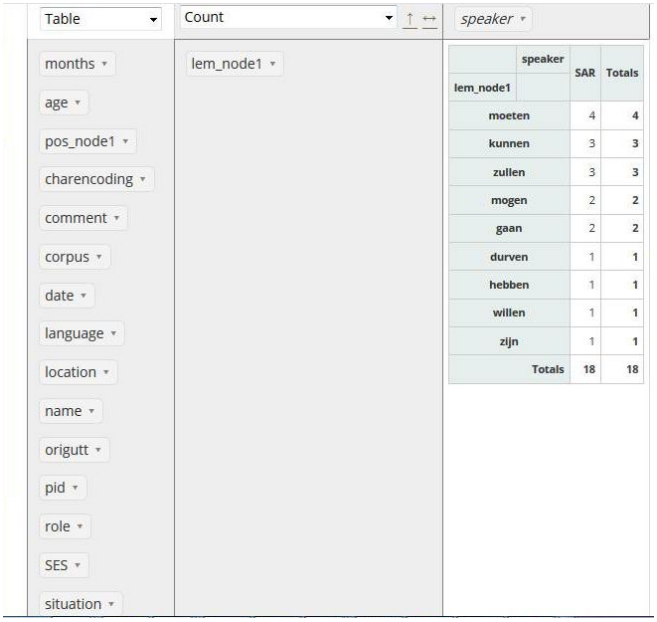


Figure 2: Screenshot of the analysis page: frequency of the superordinate verb used by Sarah.

GRA tiers (MacWhinney, 2017). The MOR tier contains lemma, morphological and morpho-syntactic information for each word occurrence, and the GRA tier contains syntactic dependency relations between word occurrences. The representation in these tiers will be integrated in upgrades of the Dutch CHILDES corpora, so that these data can also be analysed with standard CHILDES tools such as *CLAN* (MacWhinney, 2015b) or CHILDES-recommended query and visualisation tools such as ANNIS (Krause and Zeldes, 2016).

To that end, we have created a tool to convert D-COI style tags as used in Alpino into MOR-style tags, a matter which is not completely trivial because the concepts behind the two different tagging methods differ radically: basically, D-COI tags represent a morpho-syntactic characterisation that abstracts from their concrete realisation with morphs, while MOR-style tags represent abstract characterisations of sequences of allomorphs. We plan to report on this in a separate paper (Odijk et al., in preparation).

6. Related Work

To our knowledge, (Sagae et al., 2001) was the first to parse utterances in CHILDES corpora, for English. They parsed the child-directed (adult) utterances only. These already form a challenge because they are transcriptions of 'casual and conversational' speech, 'differing significantly from written natural language'.

(Sagae et al., 2007) proposed an annotation scheme for representing syntactic information as grammatical relations in CHILDES data largely based on (Sagae et al., 2004), a manually curated gold-standard corpus of 65,000 words annotated according to this scheme, and a parser (called *MEGRASP*) that was trained on the annotated corpus and produces highly accurate grammatical relations for both child and adult utterances, for English . We have not developed a parser specific to the CHILDES corpus but started from an existing parser developed for adult language, but the treebank resulting from our project can of course be used to train a child language parser.

(Pearl and Sprouse, 2013a) and (Pearl and Sprouse, 2013b) describe the creation of the CHILDES Treebank for the child-directed speech in various English CHILDES subcorpora in order to investigate the types of learning biases that are necessary to learn these constraints from the input, with the goal of determining whether any innate domain-specific biases are necessary.

(Laakso, 2005) reports on attempts to parse English CHILDES corpora automatically with a variety of rule-based and statistical parsers, showing that each of them has poor performance though the statistical parsers were slightly more successful.

(Gretz et al., 2015) describes a novel annotation scheme of dependency relations reflecting constructions of child and child-directed Hebrew utterances. A subset of the corpus was annotated with dependency relations according to this scheme, and was used to train two parsers (MaltParser and MEGRASP) with which the rest of the data were parsed.

(Dredze et al., 2007) describe results of their research on adaptation in the 2007 CoNLL Shared Task on Domain Adaptation, which involved, inter alia, CHILDES data. Their error analysis for this task suggests that a primary source of error is differences in annotation guidelines between treebanks, which clearly indicates that consistency of annotation is crucial for the usefulness of treebanks, both for humans as research material and for machine learning based software.

7. Conclusions and Future Work

We have described the approach to the development of the AnnCor CHILDES Treebank for Dutch. The treebank is still under development (we aim to make it available through the CLARIN research infrastructure in October 2018), but some results are already available, e.g. the extensions in Version 4 of the GrETEL query application, and the automatically generated parses have been made available in the PaQu application by our Groningen colleagues.

There are a number of aspects that we would like to work on in the future: (1) create options to adapt the transcription and process the adapted transcription (see section 3.4.); (2) changing a wrong syntactic parse into a correct one can be quite difficult if the wrong parse differs significantly from the correct one. The Alpino parses allows directives in the input string to guide the parsing process (so-called 'bracketed input').[28] Though we occasionally already use this feature, we would like to make it an integrated feature of the Alpino extension to the TrEd editor; (3) The CHILDES annotations and information in tiers related to an utterance such as in example (1) are currently ignored by the cleaning program. However, one might convert them into a format that can be used by the GrETEL query engine to extend the query options for searching for metadata and information on these other tiers. This requires at least an extension of the metadata notation used by GrETEL (e.g., to specify the

[28] See `https://www.let.rug.nl/vannoord/alp/Alpino/AlpinoUserGuide.html`.

span for which the annotation holds, as in example (1)), and probably also extensions in the query and analysis components.

8. Acknowledgements

This work was financed by the Utrecht University internal AnnCor research infrastructure project and by the CLARIAH-CORE project funded by the Dutch National Science Foundation (NWO).

9. Bibliographical References

Augustinus, L., Vandeghinste, V., and Eynde, F. V. (2012). Example-based treebank querying. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Bouma, G., van Noord, G., and Malouf, R. (2001). Alpino: Wide-coverage computational analysis of Dutch. *Language and Computers*, 37(1):45–59.

Dredze, M., Blitzer, J., Talukdar, P. P., Ganchev, K., Graca, J., and Pereira, F. C. (2007). Frustratingly hard domain adaptation for dependency parsing. In *EMNLP-CoNLL*, pages 1051–1055.

Druskat, S., Bierkandt, L., Gast, V., Rzymski, C., and Zipser, F. (2014). Atomic: an open-source software platform for multi-level corpus annotation. In *Proceedings of the 12th edition of the KONVENS conference*, volume 1, Hildesheim, Germany.

Eckart de Castilho, R., Mújdricza-Maydt, É., Yimam, S., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C. (2016). A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the LT4DH workshop at COLING*, pages 76–84, Osaka, Japan, dec.

Gretz, S., Itai, A., Macwhinney, B., Nir, B., and Wintner, S. (2015). Parsing Hebrew CHILDES transcripts. *Lang. Resour. Eval.*, 49(1):107–145, March.

Grün, C. (2010). *Storing and querying large XML instances*. Ph.D. thesis, University of Konstanz, Konstanz, Germany.

Hoekstra, H., Moortgat, M., Renmans, B., Schouppe, M., Schuurman, I., and van der Wouden, T. (2003). CGN syntactische annotatie. CGN report, Utrecht University, Utrecht, the Netherlands.

Krause, T. and Zeldes, A. (2016). ANNIS3: A new architecture for generic corpus query and visualization. *Digital Scholarship in the Humanities*, 31. http://dsh.oxfordjournals.org/content/31/1/118.

Laakso, A. (2005). On parsing CHILDES. Submitted to Midwest Computational Linguistics Colloquium (MCLC) 4/10/2005.

MacWhinney, B. (2000). *The CHILDES Project: Tools for Analyzing Talk*. Lawrence Erlbaum Associates, Mahwah, NJ, 3 edition.

MacWhinney, B. (2015a). Tools for analyzing talk, electronic edition, part 1: The CHAT transcription format. Technical report, Carnegie Mellon University, Pittsburg, PA, April27. http://childes.psy.cmu.edu/manuals/CHAT.pdf.

MacWhinney, B. (2015b). Tools for analyzing talk, electronic edition, part 2: The CLAN programs. Technical report, Carnegie Mellon University, Pittsburg, PA, February23. http://childes.psy.cmu.edu/manuals/CLAN.pdf.

MacWhinney, B. (2017). Tools for analyzing talk, electronic edition, part 3: Morphosyntactic analysis. Technical report, Carnegie Mellon University, Pittsburg, PA, March16. http://childes.psy.cmu.edu/manuals/MOR.pdf.

Jan Odijk et al., editors. (2017). *CLARIN in the Low Countries*. Ubiquity Press, London, UK. DOI: http://dx.doi.org/10.5334/bbi. License: CC-BY 4.0.

Odijk, J., van Noord, G., Kleiweg, P., and Tjong Kim Sang, E. (2017). The parse and query (PaQu) application. In Jan Odijk et al., editors, *CLARIN in the Low Countries*, chapter 23, pages 281–297. Ubiquity, London, UK. DOI: http://dx.doi.org/10.5334/bbi.23. License: CC-BY 4.0.

Odijk, J., van der Klis, M., and Spoel, S. (2018). Extensions to the GrETEL treebank query application. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories (TLT16)*, pages 46–55, Prague, Czech Republic, January 23-24. http://aclweb.org/anthology/W/W17/W17-7608.pdf.

Odijk, J., van der Veen, R., and Otten, M. (in preparation). Mapping D-COI tags to MOR-style tags. AnnCor technical report, Utrecht University.

Odijk, J. (2015). Linguistic research with PaQu. *Computational Linguistics in the Netherlands Journal*, 5:3–14, December.

Odijk, J. (2016a). A Use case for Linguistic Research on Dutch with CLARIN. In Koenraad De Smedt, editor, *Selected Papers from the CLARIN Annual Conference 2015, October 14-16, 2015, Wroclaw, Poland*, number 123 in Linköping Electronic Conference Proceedings, pages 45–61, Linköping, Sweden. CLARIN, Linköping University Electronic Press. http://www.ep.liu.se/ecp/article.asp?issue=123&article=004, http://dspace.library.uu.nl/handle/1874/339492.

Odijk, J. (2016b). Linguistic research using CLARIN. *Lingua*, 178:1 – 4. Linguistic Research in the CLARIN Infrastructure,http://dspace.library.uu.nl/handle/1874/339377.

Oostdijk, N., Goedertier, W., Eynde, F. V., Boves, L., Martens, J., Moortgat, M., and Baayen, H. (2002). Experiences from the Spoken Dutch Corpus project. In M. González Rodriguez et al., editors, *Proceedings of the third International Conference on Language Resources and Evaluation (LREC-2002)*, pages 340–347. ELRA, Las Palmas.

Otten, M., van der Veen, R., van Engeland, J., Meertens, E., and Wijsman, S. (2018). Anncor documentation. Utrecht University AnnCor Project Document, January.

Pajas, P. and Štěpánek, J. (2004). Recent advances in a feature-rich framework for treebank annotation. In *Proceedings of the 22nd International Conference on Com-*

putational Linguistics, pages 673–680, Manchester, UK.

Pearl, L. and Sprouse, J. (2013a). Computational models of acquisition for islands. In J. Sprouse et al., editors, *Experimental Syntax and Islands Effects*, pages 109–113. Cambridge University Press, Cambridge, UK.

Pearl, L. and Sprouse, J. (2013b). Syntactic islands and learning biases: Combining experimental syntax and computational modeling to investigate the language acquisition problem. *Language Acquisition*, 20:23–68.

Sagae, K., Lavie, A., and MacWhinney, B. (2001). Parsing the CHILDES database: Methodology and lessons learned. In *Proceedings of the Seventh International Workshop on Parsing Technologies (IWPT-2001), 17-19 October 2001, Beijing, China*. Tsinghua University Press.

Sagae, K., Lavie, A., , and MacWhinney, B. (2004). Adding syntactic annotations to transcripts of parent-child dialogs. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1815–1818, Lisbon, Portugal. ELRA, ELRA.

Sagae, K., Davis, E., Lavie, A., MacWhinney, B., and Wintner, S. (2007). High-accuracy annotation and parsing of CHILDES transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, CACLA '07, pages 25–32, Stroudsburg, PA, USA. Association for Computational Linguistics.

van Gompel, M. and Reynaert, M. (2013). FoLiA: A practical XML format for linguistic annotation - a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal*, 3:63–81, 12/2013.

van Gompel, M., van der Sloot, K., Reynaert, M., and van den Bosch, A. (2017). FoLiA in practice: The infrastructure of a linguistic annotation format. In Jan Odijk et al., editors, *CLARIN in the Low Countries*, chapter 6, pages 71–81. Ubiquity, London, UK. DOI: `http://dx.doi.org/10.5334/bbi.6`. License: CC-BY 4.0.

van Noord, G., Schuurman, I., and Bouma, G. (2011). Lassy syntactische annotatie (revision 19455). Lassy report, RU Groningen, Groningen, September 7. `https://www.let.rug.nl/vannoord/Lassy/sa-man_lassy.pdf`.

van Noord, G., Bouma, G., Van Eynde, F., de Kok, D., van der Linde, J., Schuurman, I., Tjong Kim Sang, E., and Vandeghinste, V. (2013). Large scale syntactic annotation of written Dutch: Lassy. In Peter Spyns et al., editors, *Essential Speech and Language Technology for Dutch*, Theory and Applications of Natural Language Processing, pages 147–164. Springer Berlin Heidelberg.

Wörner, K. (2009). *Werkzeuge zur flachen und hierarchischen Annotation von Transkriptionen gesprochener Sprache*. Phd, University of Bielefeld, Bielefeld. urn:nbn:de:hbz:361-16696, `https://pub.uni-bielefeld.de/publication/2301935`.

Yimam, S. M., Biemann, C., Eckart de Castilho, R., and Gurevych, I. (2014). Automatic annotation suggestions and custom annotation layers in WebAnno. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 91–96, Baltimore, Maryland, June. Association for Computational Linguistics.

BabyCloud, a Technological Platform for Parents and Researchers

Xuân-Nga Cao, Cyrille Dakhlia, Patricia Del Carmen,
Mohamed-Amine Jaouani, Malik Ould-Arbi, Emmanuel Dupoux
Laboratoire de Sciences Cognitives et Psycholinguistique,
EHESS / Ecole Normale Supérieure / PSL Research University / CNRS / INRIA
Paris, France
ngafrance@gmail.com, dakhliacyrille@gmail.com, pat_delcarmen@hotmail.com,
jaouani.mohamed.amine@gmail.com, m.ouldarbi@gmail.com, emmanuel.dupoux@gmail.com

Abstract

In this paper, we present *BabyCloud*, a platform for capturing, storing and analyzing day-long audio recordings and photographs of children's linguistic environments, for the purpose of studying infant's cognitive and linguistic development and interactions with the environment. The proposed platform connects two communities of users: families and academics, with strong innovation potential for each type of users. For families, the platform offers a novel functionality: the ability for parents to follow the development of their child on a daily basis through language and cognitive metrics (growth curves in number of words, verbal complexity, social skills, etc). For academic research, the platform provides a novel means for studying language and cognitive development at an unprecedented scale and level of detail. They will submit algorithms to the secure server which will only output anonymized aggregate statistics. Ultimately, *BabyCloud* aims at creating an ecosystem of third parties (public and private research labs...) gravitating around developmental data, entirely controlled by the party whose data originate from, i.e. families.

Keywords: language acquisition, cognitive development, linguistic development, day-long recordings, child speech corpus, child activity corpus

1. Introduction

During their first years, infants develop both physically and cognitively at an amazing speed: many parents are eager to know more about this general process and about how well their child is doing. Recent progress in infant speech database collection (Xu et al., 2008; Roy, 2009; VanDam et al., 2016; Casillas et al., 2017; Warlaumont et al., 2017) and computational modeling of developmental processes (Martin et al., 2016; Ludusan et al., 2015; Carbajal et al., 2016a; Carbajal et al., 2016b; Ludusan et al., 2017) open up possibilities to measure and model the progress of children using data collected in their natural environment. In other words, it becomes technically possible to offer parents scientifically grounded analytics and tools to explore and document their child's progress.

In this paper, we introduce *BabyCloud*, a platform for capturing, storing and analyzing day-long audio and photo recordings of children's linguistic environments. The platform is structured around 4 components: (a) *Baby Logger* which collects high definition pictures and high-quality recordings to be later decoded by automatic speech recognizers, (b) *Baby Dock* which transfers the data to a secure cloud server, recharges the recorder's batteries, (c) *Baby SmartBox*, a secure cloud database which stores and indexes raw data and extracts metadata and developmental analytics (child's linguistic landmarks, linguistic and social interactions clips and analyses) using machine learning algorithms and (d) *Baby Explorer*, which consists itself in two sub-components: a mobile application which allows families to manage and search their data, and an API accessible to scientists, will let them query summary statistics of the data and remotely run algorithms (no access to raw data).

2. Related Work

One of the more comprehensive resource for child language acquisition research is the CHILDES database (MacWhinney, 2000), which, though a tremendous source of data for language studies, presents some shortcomings. The corpora were primarily constructed to document the child's productions, and only secondarily their input; as a result, many recordings are done in the laboratory setting, and are not entirely representative of his or her activities and input in their natural environment. In addition, the majority of the corpora contain only orthographic transcriptions, many of the audio recording were done for human transcription and the quality is not ideal for automatic speech processing.

An ambitious project for gathering day-long home audio recordings was initiated at MIT with the Human Speechome Project (Vosoughi and Roy, 2012). It consisted in a continuous recording of all of the child's environment (audio and video), some of which has been transcribed. However, due to the private nature of the data, it is not accessible to outside researchers.

HOMEBANK (VanDam et al., 2016) created by the DARCLE[1] group, is another repository for day-long family audio recordings and addresses some of the limitations encountered with the CHILDES database or the Human Speechome Project. It gathers large datasets of audio (and sometimes video) recordings of the infant in his or her natural environment. It addresses the privacy problem by a legally binding protocol of nondisclosure agreement signed by the researchers, their institutions and the repository. This only partially addresses the problem however, given

[1] Day-long Audio Recording of Children's Linguistic Environments

once the agreement is signed, the data can be copied onto the researcher's computer, with no possibility to control what becomes of the data. This is a problem as data protection laws differ across countries (for example, the General Data Protection Regulation law will be enforced in May 2018, aiming at protecting and securing all European residents' personal data, while the US Data Privacy regulation ensures only online consumer's data security and privacy is under FTC authority since 2000).

Another technical issue for the automatic processing of large quantities of speech data is the audio quality of the initial recordings. One recorder developed with this objective in mind is the LENA[2](Xu et al., 2008; Oller, D.K, 2011). The device has been miniaturized and engineered to maximize the quality of the recording, while worn by children of various ages. The LENA system also provides an analysis package using machine learning technology to automatically segment the recordings in audio chunks labelled with broad classes (target child, adult male near, background noise, etc). One main limitation of this tool is that the data and processing tools are proprietary and only optimized for English households, making it difficult to adapt to other languages or projects. In addition, it offers no search tools for the parents.

3. The *BabyCloud* Platform

We developed *BabyCloud*, an innovative open-source platform that strives to connect two communities of users: families and academic researchers. It stands out from existing models with its commitment to protect, first and foremost, the collected data and to give the full control and ownership to the party whose data originate from: the families.

Many parents who contribute to science by donating their infant's data and their time also care about their child's development and his/her prospects at school. To address this, the platform will enable the development of novel services such as the ability to follow the development of their child on a regular basis through language and cognitive analytics (growth curves in number of words, verbal complexity, social skills, etc.) even before he/she attends school. In addition, they will have complete control of their data usage, with the option to open up null/partial/full data access to researchers and/or third parties.

For academic research, the platform provides a novel instrument for studying language and cognitive development, potentially, at an unprecedented scale and detail level. Unlike existing data repositories, our platform ensures simultaneously total privacy protection and openness: the data are encrypted and never leave the secure server; however, researchers are able to submit algorithms to the server to analyze the data and only anonymized aggregate statistics will be allowed to get out of the server (see figure 1 for the *BabyCloud* workflow).

[2]https://www.lena.org/

3.1. The *Baby Logger* Component

The *Baby Logger* is a light, ergonomic and wearable recorder. It has been designed to combine efficiency and convenience for daily life usage. The device records sounds thanks to an array of eight high-quality microphones, captures high resolution pictures and data from a three-axis accelerometer. It is protected by an open source hardware licence, and can be reconfigured for particular projects (see figure 2 for more details).

3.1.1. Operating Modes

The operating modes of the device optimize ease of use and respect of privacy. Two versions of the device have been designed, one for the parents and the other for the child. They have the same features, but the one worn by the parents stops recording and goes into hibernation mode when they go far away from the child. This proximity detection is possible thanks to a radio-frequency identification (RFID) chip affixed on the parents' recorder whereas the child's one only possesses a tag. This setup gives more privacy to the parents and protects the child from a permanent wave exposure. In addition to this automatic setting, a manual privacy mode can be selected, with the parents temporarily stopping the recording on both devices. This situation may occur when the conversation contain sensitive data or other guests are present and don't want to be recorded. The camera will also be equipped with a flap to hide the lens if the family doesn't want the *Baby Logger* to take pictures. The decision to add the camera is to have pictures from the child's perspective, which can shed light on his/her home environment and development.

The user interface is very simple with only two buttons, one for turning on/off the device and the other, easily accessible for selecting the privacy mode. Two LEDs indicate each one of the following status: privacy vs recording mode, and a warning for docking station operation (low battery or full memory). The *Baby Logger* is equipped with a large internal memory to allow a maximum of 24 hours of continuous sound recordings and a stream of images every 10 seconds. Less dense acquisition settings are available, notably, random or periodic 2 to 10 minutes sample. The battery has been designed to last for at least 24 hours under the maximum recording density.

3.1.2. Choice of Technology

The choice of the electronic components has been made to optimize data quality and power usage. Our recorder is built around a high performance microcontroller, equipped with a Cortex-M7 ARM microprocessor providing enhanced performance and optimized power consumption, representing the most relevant solution for IoT and multimedia systems. Audio acquisition is performed with an array of 8 MEMS (Micro ElectroMechanical Systems) microphones, a new generation of digital microphones directly built on semiconductors and integrating digital circuits (ASIC). MEMS are an attractive solution because of their small size and weight, allowing systems to be miniaturized. The camera is a 5 Mega Pixels CMOS digital camera. It has a Digital Signal Processor (DSP) and

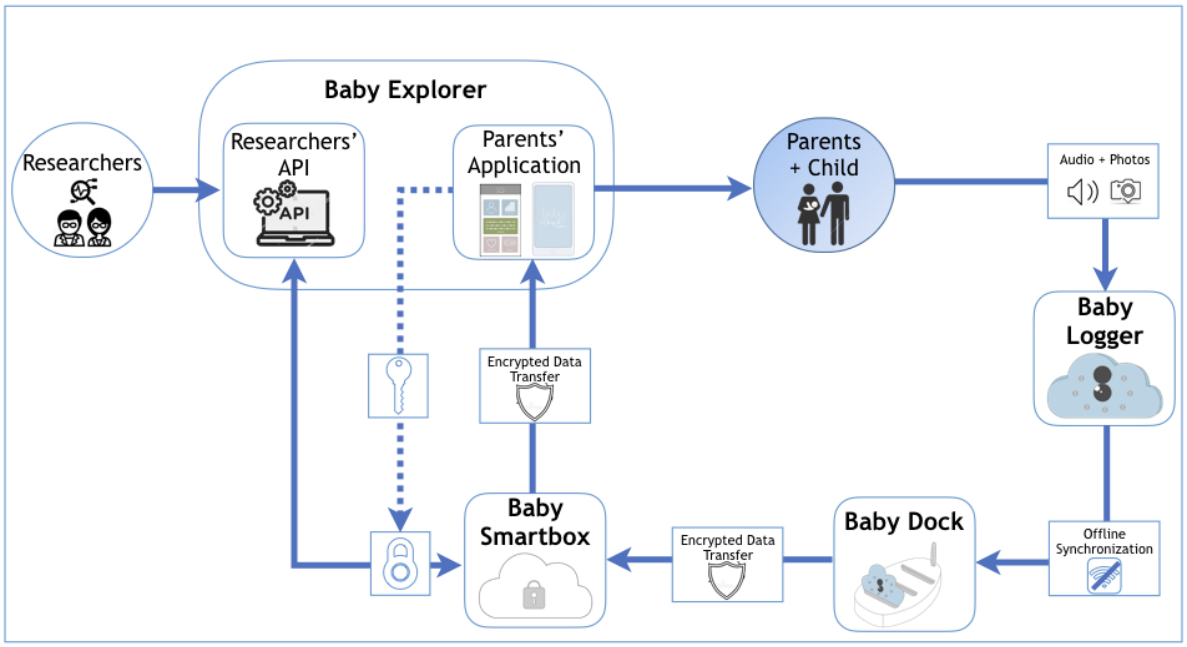


Figure 1: *BabyCloud* Workflow.

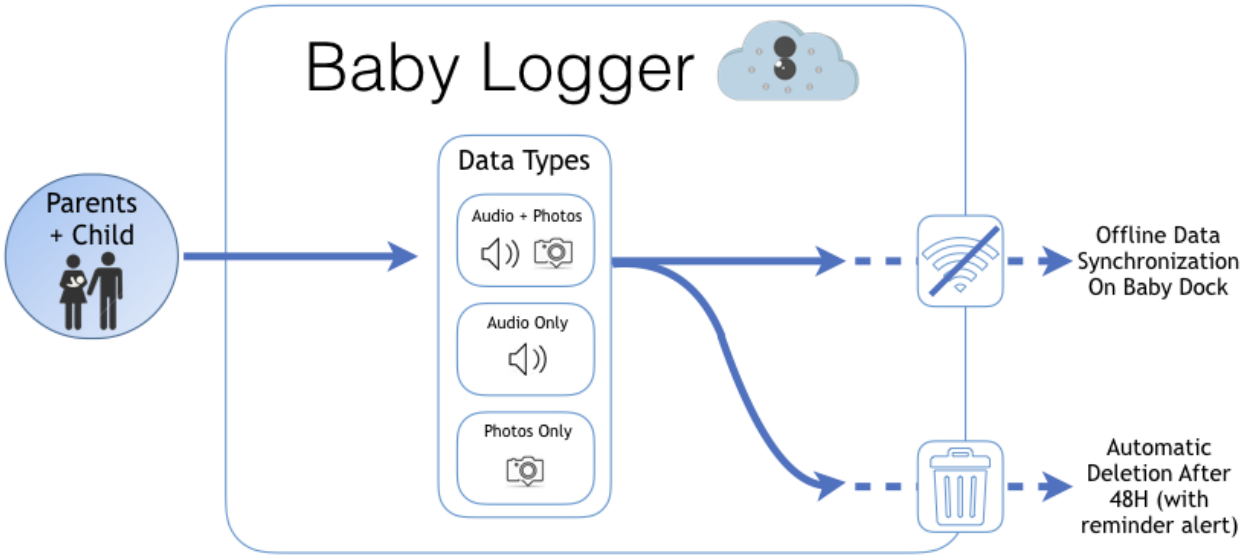


Figure 2: *Baby Logger* Architecture.

presents a control interface allowing customization of the camera settings.

3.2. The *Baby Dock* Component

While the device is recording, no data is being transferred to save on battery life and to protect the child from wave exposure. The wireless data synchronization is performed only when the battery is being recharged onto the *Baby Dock* (see figure 3 for more details).

The *Baby Dock* represents the interface between the wearable recorder and the secure cloud. A Raspberry Pi 3 (our preferred solution but any computer with a Wifi module working as a host for a local network can be used) is connected to the Internet thanks to an ethernet wire which ensures the link with the home network. A three-phase cyclical software will run on the *Baby Dock*:

- Recorder synchronization Phase: During this stage, the connection to the Internet is disabled to ensure the security of the raw data being transferred from the recorders to the *Baby Dock*. The transfer is done instead via the local Wifi network setup between the Life Logger and the Dock. Simultaneously the Life Logger's batteries are recharged through inductive charging. At the end of synchronization, the data is removed

from the *Baby Logger*'s memory.

- Data Filtering Plugin: Signal processing algorithms are applied offline to the data in order to separate speech from background noise (source separation, speech enhancement) and segment the recordings according to two classes (speech, non-speech). This significantly reduces the amount of data transferred to the cloud. We will use open-source sytems (Jin and Schultz, 2004; Rudzicz, 2013) which will be tuned to the baby logger data and customized to fit into the baby dock.
- Cloud synchronization Phase: The last phase of the cycle ensures the transfer of the encrypted data to the secure cloud server via an ethernet connection and their removal from the Dock.

3.3. The *Baby Smartbox* Component

The *Baby Smartbox* stores the collected raw data, runs advanced signal processing and machine learning algorithms to automatically add meta-data (speaker diarization, activity detection, estimation of the child's vocal maturity, etc), and provides a database server. The server is protected to provide access only to authorized users via a login authentification process. It interfaces with the Baby Explorer via a RESTful API developed within the Python Flask framework (see figure 4 for more details).

This component will have a core secure database management system selectively granting access to data to different classes of users (parents, researchers). In addition, it will have a set of machine learning algorithms (plugins) which will automatically data annotations at various linguistic levels. We are currently focusing on replicating some of the annotation layers of the LENA system, using state-of-the-art, open source and retrainable software. These include:

- Segmentation broad class speaker ID. This requires algorithms that perform Speech or Voice Activity Detection: they segments speech from background noise (Uhle and Bäckström, 2017). Broad class speaker ID classifies the resulting segments into a small number of categories, like target child, other child, males, females.
- Speaker diarization. This task is more difficult than the above, because it requires to classify utterances according to an unknown number of speakers. The task is made easier if the speakers can be 'enrolled' in advance, i.e. that a few minutes are manually annotated. The current performance of speaker diarization systems based on I-vectors is dependant on utterance length. Typically, performance is not very good with short utterances (Kanagasundaram et al., 2016).

Other plugins will be added as they become available in open source format. In particular, we are working in collaboration with the ACLEW[3] consortium. This is funded by a transatlantic research initiative (Digging Into Data project[4]) aiming at creating a common annotation scheme for diverse (culturally and linguistically) infant datasets, and at building tools to semi-automatically analyze those datasets. The plug-ins currently under development are:

- Syllabic segmentation. Syllables are acoustically saillant events in speech and can be efficiently detected (Räsänen et al., 2018). In turn these can be used as proxy for utterance length.
- Vocal maturity. Infant produce several stages of vocalization at different developmental stages (vegetative vocalizations, crying, laughter, canonical babbling, varigated babbling, etc.). Given their rather stable shape across languages, it seems feasible, using standard machine learning tools (markov models, SVM or random forest models) to construct a retunable classifier that will enable to produce developmental statistics based on these categories.
- Child-directed vs. adult-directed speech. Parents in many culture address their infants in specific ways (Fernald and Kuhl, 1987), and child directed input seems to be a better predicted for language acquisition than total input (Weisleder and Fernald, 2013). Being able to automatically annotate this difference in register could help predictive models of language development.

As these plugins are developed and open sourced, they will be incorporated into the platform. We construct all of these modules to be trainable. Which means that the automatic annotation can be retuned to specific recordings, provided the parents gave authorization for a subset of the data to be annotated manually. In that case, the annotated data is split into a training set and a test set to evaluate the reliability of the machine annotation. Similarly, manual annotations are required to adapt any new plugin to the particular data collected by the baby logger. All of the automatic annotations are considered derived data and added to the raw data. They are used to derived analytics, such as daily vocal activity, linguistic complexity (mean utterance length), and social responsiveness indexes (turn taking), which will be accessible to the parent for their child and to the scientist as aggregate statistics (see section 3.4.2).

3.4. The *Baby Explorer* Component

The *Baby Explorer* is designed to cater to 2 types of users: parents and researchers.

3.4.1. The *Baby Explorer*: parents' interface

It is a hybrid mobile/web application developed within the Ionic framework using the following technologies: Typescript (Javascript) and Angular 4 for data processing and actions linked to screen/hardware events, HTML5 and CSS3 for graphic user interfaces, Cordova for hardware access to use mobile native functionalities, and PostgreSQL for databases.

[3] Analyzing Child Language Experience around the World: https://sites.google.com/view/aclewdid/home

[4] https://diggingintodata.org/

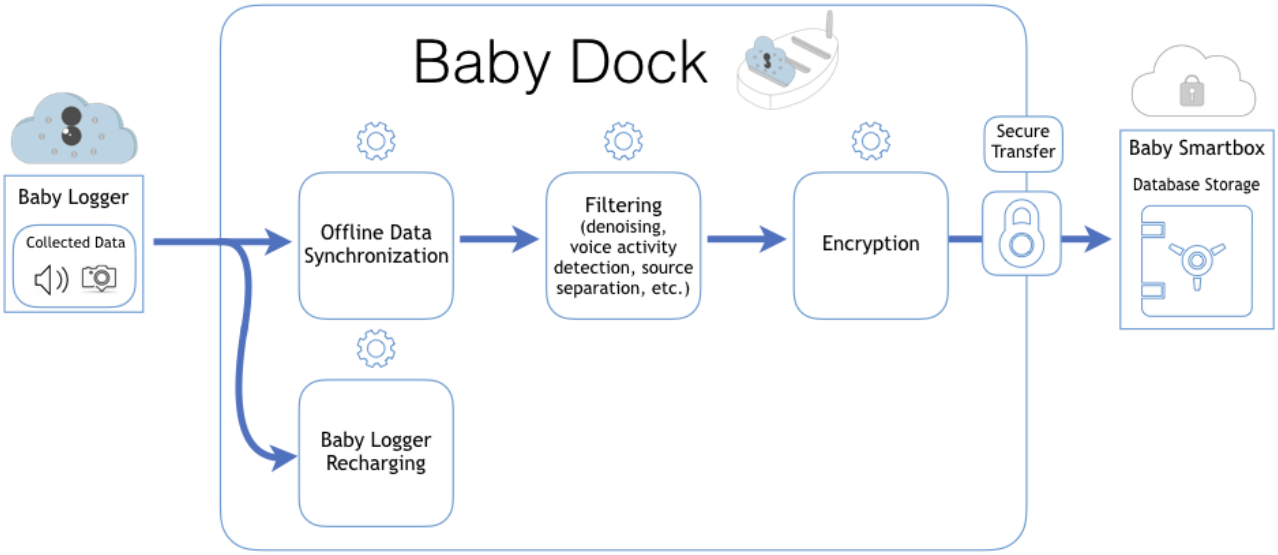


Figure 3: *Baby Dock* Architecture.

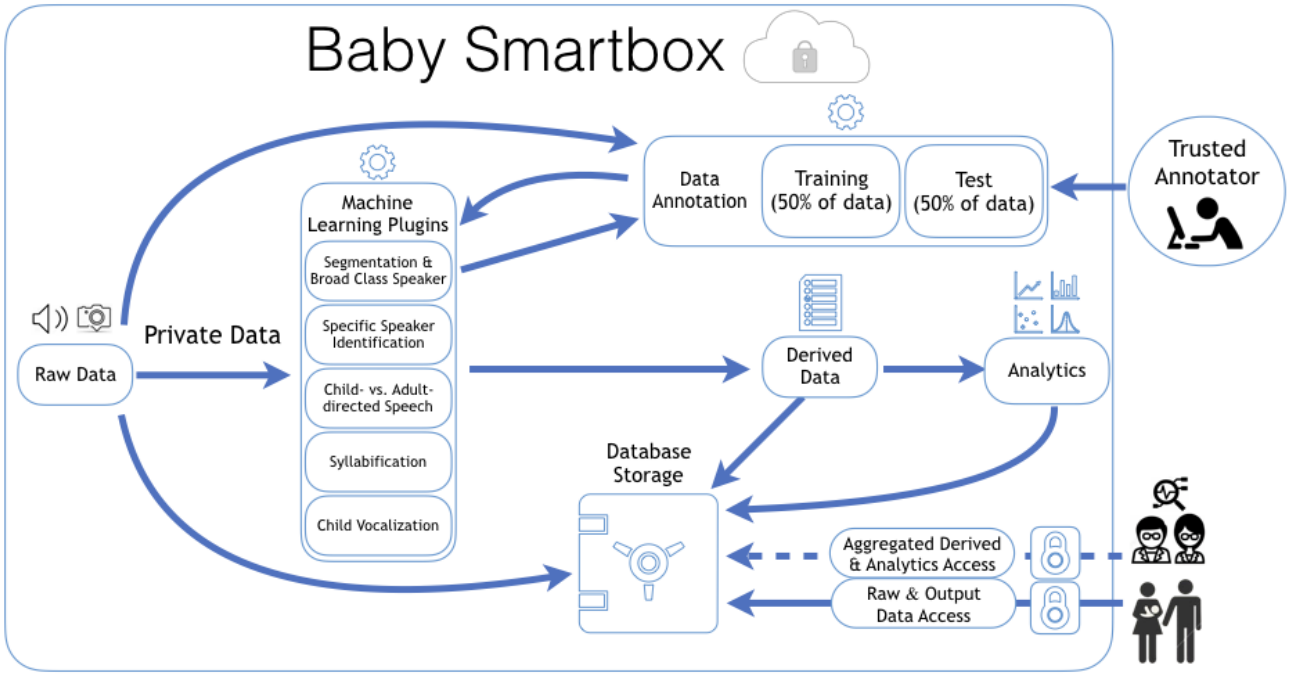


Figure 4: *Baby Smartbox* Architecture with example machine learning plugins.

The objective is to actively engage families with their dataset by offering them an attractive tool with several functionalities: (a) browse their child's recordings using time and/or activity filters, (b) track their child's language and cognitive development with the provided analytics and statistics, (c) manage the data access for researchers (see figure 5 for more details).

To ensure that parents fully understand the extent of their rights with regards to data protection, an online consent form must be signed when parents log in for the first time onto the *Baby Explorer*. Unlike traditional terms of agreement clauses, this one is aimed at protecting the user's rights and privacy and making sure that the user understands the terms of contract he/she is agreeing on. This last step is verified by an online quiz while the user is going through the consent form. Since parents own the data, they have the option to update, erase, remove them or ask for their portability from the secured cloud, using the interface. In the same way, the child, once he/she turns 13 years old (based on Children's Online Privacy Protection Rule aka "COPPA"), will be required to express his/her free consent

to continue or stop his/her participation to the project.

- Data View: The data view interface helps families browse through their infant's data via a customized and ergonomic time-line, allowing to simultaneously access audio recordings and their associated pictures. On this timeline parents can control the audio and image channels (play, stop, rewind, re-scale, etc), and can filter the data by activity. They can bookmark their favorite recordings or pictures, share them with relatives or friends and set privacy settings on their data with regards to their use for research. The pictures taken from the child's device will inform parents and researchers on his/her perception of his local environment (people he/she is looking at, location, activities such as playing, eating, napping...).
- Dashboard: The dashboard displays an overview of the metrics about the child's linguistic development. It allows parents to follow his/her cognitive development on a daily basis through analytics and statistics (growth curves in number of words, verbal complexity, social skills, etc.).
- Access Authorization: This interface allows families to see all the requests made by researchers to access their data. Parents are able to read information about the enquirers such as their name, contact information, status, summaries of their projects and collaborations, publications, etc. Acceptance from the parents is a pre-requisite for scientists to work on the data. Parents have the option to define presets to automatically accept/reject/block certain types of requests (educational research, medical research, etc.). They can also grant/revoke access on a project by project basis.

3.4.2. The *Baby Explorer*: researchers' interface

This API (Application Programming Interface) offers a bundle of functionalities and tools dedicated to researchers. They will have at their disposal new and large dataset with fine layers of annotations, ready for analysis and obeying the regulations on data protection. Beforehand, all researchers will be required to pre-register their studies and get approved by an ethics committee. Once their algorithms have been tested with public data on the virtual machine sandbox at their disposal, scientists can send access requests to parents to run these algorithms on their meta-data. When the access is granted, they have to use the validated virtual machine, designed to make sure that the raw data cannot be retrieved. The only output retrieved by the researchers is anonymized statistics of the data. In a later part of the project, we will add the possibility for scientists to request algorithmic access to the raw data in order to add new layers of meta-data (see figure 6 for more details).

4. Completed Work

At the time of writing, we have completed the following steps: we have a working prototype of the Baby Logger, of the Baby Dock, of the Secure Cloud Database and of the Baby Explorer parent's interface. The algorithms for speech analysis are being developed in parallel by the ACLEW team (see section 3.3 above). We have submitted the project description to our local ethics committee. It has been reviewed favorably and will be approved after revisions. The proposal has also been submitted for a formal authorization to the National Committee for data protection. It is currently under review. The miniaturization and baby friendly design of the Life Logger is underway, and the scientist's interface under study.

We also conducted a short pilot with six families to apprehend the parents' needs and expectations. These families were selected from our babylab's database which comprises more than 1000 volunteer families. The infant wore a T-shirt with a pocket hiding the audio recorder (such as the LENA or a USB key recorder). The recording happened at home for one/two day(s). We asked for the parents' feedback afterwards about the experiment. All were in favor of pursuing the study but they expressed reservations on the recording duration (once or twice a week instead of continuous recordings), were interested in a more practical and fit-all recorder. There were mixed reactions on the camera function (some parents wanted to disable it completely). We annotated (with the parents' consent) a small sample of the collected data in order to integrate the analytics into the application. A second interview with the families will be scheduled to show the results on the application and have their feedback on the user experience to improve on the functionalities of the platform.

Once the protocols are validated by the data protection and ethics committees, we plan to scale up the project in terms of number of families (20 families) and duration (one year or more). We aim to achieve a functional platform with agile software/hardware development through rapid iteration cycles. During these cycles of improvement, we will pay particular attention to user experience and data protection.

5. Expected Impact and Conclusion

In this paper, we presented an innovative platform aimed at reinforcing collaboration between parents and researchers. The benefits of such a platform for the community users and for society are diverse.

The benefits for families are twofold. The first one is health related: the language and cognitive analytics could help parents spot potential developmental delays and trigger early medical intervention. The platform does not propose medically validated diagnostic tools but parents may use the provided data and consult speech and language therapists to get further assistance and medical advice. A secondary benefit is related to the quality of life. Our platform will provide cloud storage space and indexing services related to the child: automatically extracted pictures and audio snippets. In addition, parents will be able to include other types of information (potentially,

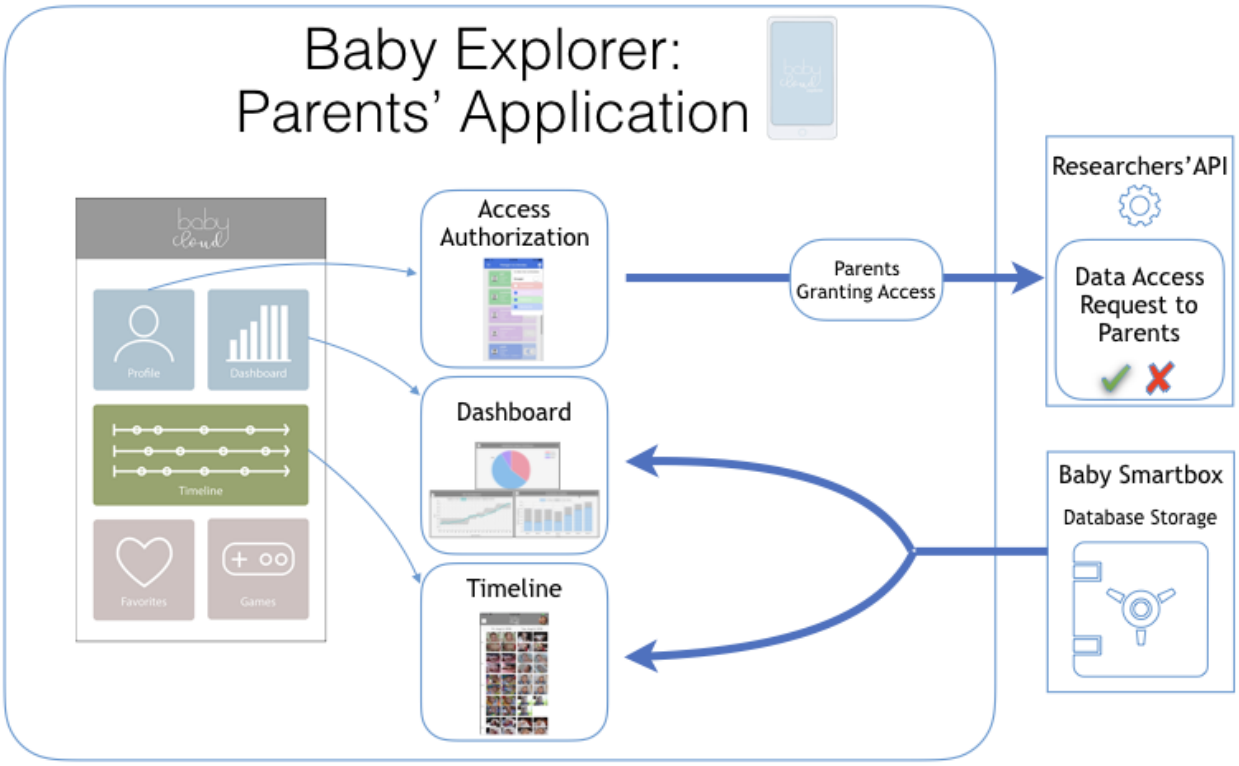


Figure 5: *Baby Explorer* Architecture: Parent's Application.

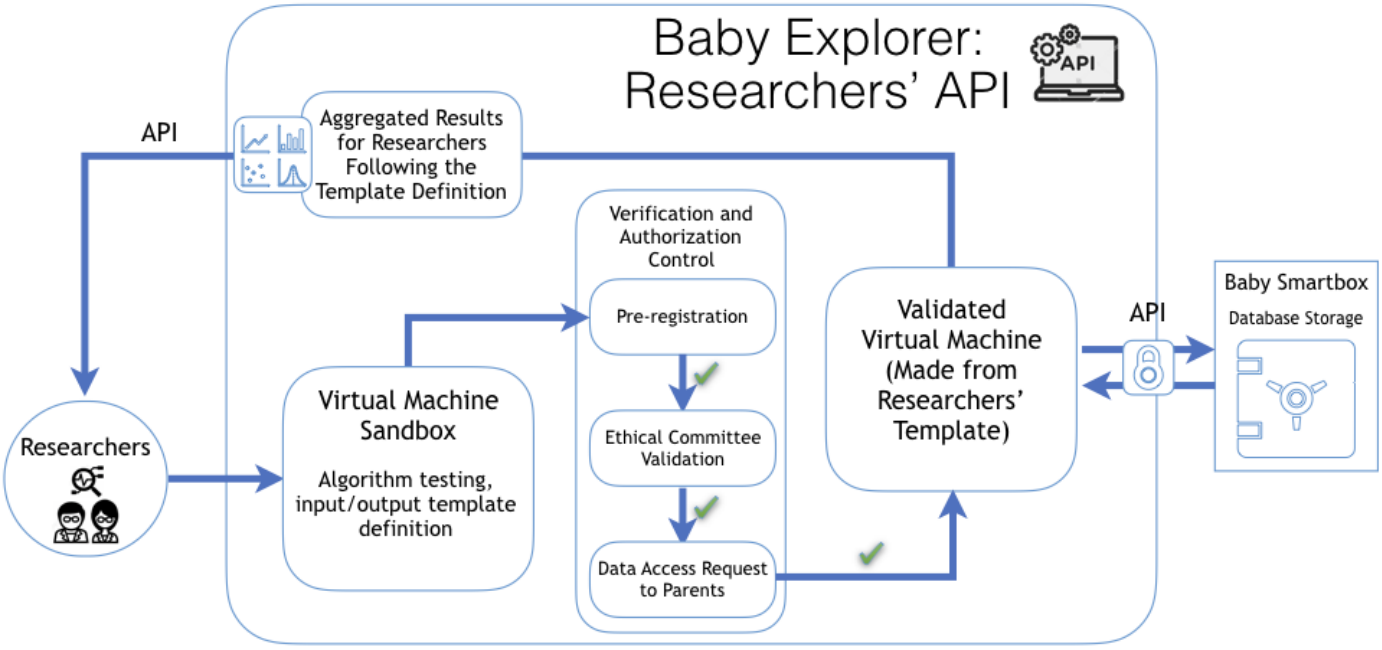


Figure 6: *Baby Explorer* Architecture: Researcher's API.

school and medical records), which security level will be guaranteed. The whole portfolio will constitute a digital asset for the child, which can be permanently accessible as a memory of early childhood.

For academia, large recordings of infant development will allow researchers to establish new quantitative and predictive models at a level that was not possible before.

These models will feed back into better analytics and in a long term, detect early signs of potential cognitive delays. An additional benefit will be the involvement of parents into science (citizen science), who will help collect data and perhaps, volunteer annotations.

In a later phase, we plan to include a third community to interact with the families and researchers: private companies. For private IT companies, personal data is a valuable source for big data analytics and building business strategies based on the customer's preferences. The platform will offer access to large and rich amount of data but which will be protective of families' privacy. Under this strict privacy preserving protocol, companies would be able to offer families new tools and analytics. Vice-versa, families may open up, through smart contracts, partial aspects of their data to companies for specific purposes. These smart contracts will remain under the lab's supervision to ensure that the family's privacy and data protection are safeguarded.

Finally, society may benefit from the platform in the area of developmental pathologies. Developmental disorders are typically diagnosed too late (i.e., at a time when the child is already lagging behind in school), resulting in mental strains and heavy financial costs for families and also for society. Tools that will raise awareness about these disorders and their impact may help alleviate those costs.

6. Bibliographical References

Carbajal, J., Dawud, A., Thiollière, R., and Dupoux, E. (2016a). The 'language filter' hypothesis: Modeling language separation in infants using i-vectors. In *EPIROB 2016*, pages 195–201.

Carbajal, J., Fér, R., and Dupoux, E. (2016b). Modeling language discrimination in infants using i-vector representations. In *The 38th Annual Conference of the Cognitive Science Society*, pages 889–896.

Casillas, M., Bergelson, E., Warlaumont, A. S., Cristia, A., Soderstrom, M., VanDam, M., and Sloetjes, H. (2017). A new workflow for semi-automatized annotations: Tests with long-form naturalistic recordings of childrens language environments. In *Proc. Interspeech 2017*, pages 2098–2102.

Fernald, A. and Kuhl, P. (1987). Acoustic determinants of infant preference for motherese speech. *Infant behavior and development*, 10(3):279–293.

Jin, Q. and Schultz, T. (2004). Speaker segmentation and clustering in meetings. In *8th International Conference on Spoken Language Processing*, page 597–600.

Kanagasundaram, A., Dean, D., Sridharan, S., and Fookes, C. (2016). Dnn based speaker recognition on short utterances. *arXiv preprint arXiv:1610.03190*.

Ludusan, B., Cristia, A., Martin, A., Mazuka, R., and Dupoux, E. (2015). Learnability of prosodic boundaries: Is infant-directed speech easier? *Journal of the Acoustical Society of America*, 26(3):341–347.

Ludusan, B., Mazuka, R., Bernard, M., Cristia, A., and Dupoux, E. (2017). The role of prosody and speech register in word segmentation: A computational modelling perspective. In *ACL 2017*.

MacWhinney, B. (2000). *The CHILDES Project: Tools for analyzing talk*. Mahwah, NJ: Lawrence Erlbaum Associates, 3rd edition.

Martin, A., Schatz, T., Versteegh, M., Miyazawa, K., Mazuka, R., Dupoux, E., and Cristia, A. (2016). Mothers speak less clearly to infants: A comprehensive test of the hyperarticulation hypothesis. *Psychological Science*, 140(2):1239–1250.

Oller, D.K. (2011). Lena: automated analysis algorithms and segmentation detail: how to interpret and not overinterpret the lena labelings. In *LENA Users Conference*, Denver, CO.

Räsänen, O., Doyle, G., and Frank, M. C. (2018). Prelinguistic segmentation of speech into syllable-like units. *Cognition*, 171:130–150.

Roy, D. (2009). New horizons in the study of child language acquisition. In *Proceedings of Interspeech*.

Rudzicz, F. (2013). Adjusting dysarthric speech signals to be more intelligible. *Computer Speech Language*, 27:1163–1177, 09.

Uhle, C. and Bäckström, T. (2017). Voice activity detection. In *Speech Coding*, pages 185–203. Springer.

VanDam, M., Warlaumont, A. S., Bergelson, E., Cristia, A., Soderstrom, M., De Palma, P., and MacWhinney, B. (2016). Homebank, an online repository of daylong child-centered audio recordings. *Seminars in Speech and Language*, 37:128–142.

Vosoughi, S. and Roy, D. (2012). An automatic child-directed speech detector for the study of child language development. In *Interspeech 2012*, Portland, Oregon.

Warlaumont, A. S., VanDam, M., Bergelson, E., and Cristia, A. (2017). Homebank: A repository for long-form real-world audio recordings of children. In *Proc. Interspeech 2017*, pages 815–816.

Weisleder, A. and Fernald, A. (2013). Talking to Children Matters: Early Language Experience Strengthens Processing and Builds Vocabulary. *Psychological Science*, 24(11):2143–2152, November.

Xu, D., Yapanel, U., Gray, S., and Baer, C. (2008). The lena language environment analysis system: the interpretive time segments (its) file. Technical Report No. LTR-04-2, LENA Foundation Technical Report.

7. Acknowledgements

Our research was funded by the European Research Council (ERC-2011-AdG 295810 BOOTPHON). It was also supported by the Agence Nationale pour la Recherche (ANR-10-IDEX-0001-02 PSL and ANR-10-LABX-0087 IEC) and the ENS Fondation (chaire Almerys).

Infant Word Comprehension-to-Production Index Applied to Investigation of Noun Learning Predominance Using Cross-lingual CDI database

Yasuhiro Minami[1], Tessei Kobayashi[2] and Yuko Okumura[2]
The University of Electro-Communications[1], NTT[2]
1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan[1], 2-4 Hikaridai Seika-cho Soraku-gun, Kyoto, 619-0237, Japan[2]
minami.yasuhiro@is.uec.ac.jp[1], (kobayashi.tessei, okumura.yuko)@lab.ntt.co.jp[2]

Abstract

This paper defines a measure called the comprehension-to-production (C2P) index to investigate whether nouns have predominance over verbs in children's word learning that identifies a partial word learning period from comprehension to production. We applied the C2P index to noun predominance using cross-lingual child communicative development inventory databases and confirmed that it indicates noun predominance in word learning, suggesting that the process between a word's comprehension and its production is a significant factor of predominance in noun learning by children.

Keywords: vocabulary learning, word acquisition, cross-lingual CDI database

1. Introduction

At around one year of age, infants produce their first words and rapidly start acquiring more of them. Almost every child follows this tendency. However, the content of a child's first few words depends on the individual. These variances might reflect different culture, language, and individual environments. In the children word learning process, identifying what parts of speech children learn first is important; this result would be a step toward resolving the children's word acquiring mechanism and would provide clues for engineering solutions for teaching words to computers. Gentner argued that nouns should predominate verbs since they appear more often than verbs in the early development stages of children (Gentner, 1979; Gentner et al., 2001).

There are two hypotheses for this predominance: the existence of a word continuum and constraints (biases). Gentner explained the former hypothesis using a division of dominance continuum. She assumed a word continuum in abstract space that varies from cognitive to linguistic dominance and introduced two assumptions, natural partitions and relational relativity, to explain that children learn words in the cognitive dominance region earlier than words in the linguistic dominance region. Since the natural partitions insist that concrete objects and entities are easier to individuate in the world, children easily acquire nouns. Relational relativity insists that a verb's meaning is not isolated and that it depends on the surrounding words.

Maguire et al. explained similar reasons for noun predominance using the shape, individuation, concreteness, and imageability (SICI) continuum (Maguire et al., 2006). These factors are assigned to the one dimensional abstract space axis (from left to right). The instances of words are arranged on the axis where the far left instance has an easy shape, simple individuation, high concreteness, and high imageability. Considering the meaning of nouns and verbs, we can allocate nouns to the left hand side and verbs to the right hand side. The SICI continuum has distributions of nouns and verbs that describe their individual difficulty differences.

However, no direct evidence has connected the concrete measure of word difficulty to abstract spaces because no index, other than the occurrence rate of the part of speech, has expressed word difficulty. Since the occurrence rate is strongly affected by such environments as culture and parent input, measuring word difficulty is not appropriate. This assumption, that noun acquisition predominates verb acquisition, remains controversial (Benedict, 1979; Tardif, 1996).

This paper defines a concrete measure to evaluate noun predominance in word learning. We apply this measure to evaluate noun predominance using a cross-lingual child word development database. We also discuss the reason underlying noun predominance.

2. Crosslingual CDI databases

The MacA rthur Communicative Development Inventories (CDI) (Dale et al., 1996), which are based on parent reports, are used to check when children comprehend and produce a particular word. Mothers of children of a certain age or less complete the Words and Gestures (WG) CDI form, and mothers of children over a particular age by months fill out the Words and Sentences (WS) form. WG has two columns that verify whether the toddler understands or understands/says a particular word. Checking the "understand" column means that the child completely comprehends the word. Checking the "understand/say" column means that the child has produced the word. WS has only one column to verify whether a child can say a particular word. Checking this column means that the child has produced that word.

Even though CDI was originally develped for English (Dale et al., 1996), it has recently been adapted to other languages to provide research resources, and cross-lingual

CDI databases are now available. The next sub-section describes the databases we used.

2.1 American, Spanish and Danish CDI databases

We use American (Dale et al., 1996), Mexican Spanish (Dale et al., 1996), Danish (Madsen, 2008) database in the Lex 2005 CDI database (Jørgensen, et al., 2010)

American inventory for WG has 396 words. The comprehension and production norms are calculated from inventory result for children from 8 to 16 month ages. American inventory for WS has 680 words for children from 16 to 30 month ages. Mexican Spanish inventory for WS has 427 words. The comprehension and production norms are calculated from inventory result for children from 8 to 18 month ages. Mexican Spanish inventory for WS has 681 words for children from 16 to 30 month ages.

2.2 Japanese CDI database

We collected cross-sectional data from 1,852 mothers living in Nara, Osaka, and Kyoto with 10~32 month-old children, and these women performed the Japanese version of CDI at our laboratories over about six years from April, 2006. The Japanese inventory for WG has 448 words for 10 to 22 months. WS has 711words for 20 to 32 month-old children..

Table 1. Number of children in WS and WG for each language.

Language	Number of children	
	WS	WG
American	1461	1089
Mexican Spanish	778	1094
Danish	3714	2398
Japanese	1506	346
Croatian	250	377
French (Quebec)	827	537
Italian	753	648
Korean	156	40
Latvian	500	183
Norwegian	9304	2926
Slovak	1065	657
Turkish	2422	1115

2.3 Croatian, French_(Quebec), Italian, Korean, Latvian, Norwegian, Slovak and Turkish databases

CDI has also been applied to other languages. The following databases were extracted from Wordbank's available databases (Frank et al., 2016): Croatian (Kovacevic, Babic, Brozovic, 1996), French (Quebec) (Trudeau, Sutton, 2011), Italian (Caselli, Casadio, Bates, 1999), Korean, Latvian, and Norwegian (Simonsen, Kristoffersen, Bleses, Wehberg, Jørgensen, 2014), and Slovak and Turkish (Ay, 2009). The data were downloaded on 9/17/17.2.4 Number of children for each language database

The number of children for each language is shown in Table 1.

3. Fitting the acquisition curves by logistic functions

By monthly categorizing the obtained CDI data, we calculated the acquisition rates of children who comprehend and produce a particular word every month. Here the rates, which were calculated from insufficient data, were treated as missing values. Then we modeled these curves using logistic functions with respect to age in days:

$$f(x) = \frac{ae^{cx+b}}{1+e^{cx+b}}, \quad (1)$$

$$f'(x) = \frac{a'e^{c'x+b'}}{1+e^{c'x+b'}}, \quad (2)$$

where f and f' are the logistic functions for word comprehension and word production. First, Eq. (1) is calculated by the nonlinear least mean square method to fit the rates of infants who produce a word. Here we introduce a constraint where the probability must be one or less. To satisfy this constraint, if a is bigger than 1.0, it is set to 1.0, and then b and c are recalculated by the nonlinear least mean square method.

Next we calculated Eq. 2. However, we found that for some words, the comprehension acquisition rates were not appropriately calculated since we didn't have as much data for these words as for the word production data. This lack of data was caused by the difficulty that mothers verify the their children's vocabulary after their children comprehend a large number of words. Thus we set constraint $f(x) \leq f'(x)$ under which a', b', and c' should be obtained. However, simultaneously obtaining a', b', c', a, b, and c under the constraint is mathematically difficult. Therefore instead of that complicated constraint, we introduce a simple constraint, $a \leq a'$, in the following optimizing method:

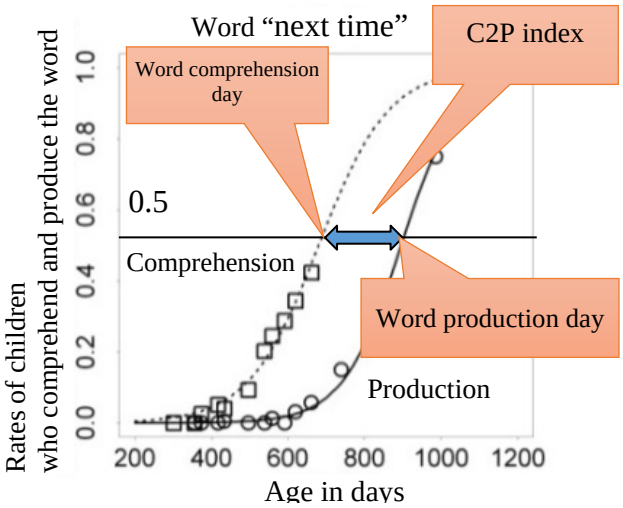


Fig. 1 Logistic functions fitting production acquisition and comprehension rates of kondo (next time) with constraint and calculating C2P index .

To estimate correctly the production rate, we introduce a constraint where the word comprehension rate should exceed the word production rate. a', b' and c' are obtained by the non-linear least mean square method ; if a>a', a' is fixed to a, and then b' and c' are recalculated by non linear least mean square method. If a' > 1.0, a' is fixed to 1.0, and then b and c are recalculated.

Fig. 1 shows an example of estimated logistic functions for the acquisition rates of kondo (next time). These approximations work well for the acquisition and word production rates. An example of this situation is shown in Fig. 1, where the dotted line is the comprehension rate curve. The solid line in Fig. 1 shows the obtained production rate.

4. Word comprehension-to-production index

The ratio of the parts of speech produced within a certain period sequence was previously used to investigate which part of speech predominates. One problem with such ratios is that since they can only be obtained for parts of speech, they cannot determine which word is difficult by a word-by-word examination. Because of this, the rate is strongly affected by such environments as culture and parent input

To evaluate word difficulty in such a word-by-word fashion, we used the word comprehension day, the word production day, and the period between them. We call this period the comprehension-to-production (C2P) index. Although some might think that it only evaluates a partial learning process of children, it is important to subdivide the learning process and investigate each part of it to precisely understand the entire learning process. To calculate this index, we first define the word comprehension and production days as when 50% of the children respectively comprehend and produce a particular word. These days were determined by approximating the word comprehension and production rate curves by two logistic functions, setting the functions to 0.5, and solving them by the Newton method. Fig. 1 shows an example of the C2P index for kondo (next time).

5. Investigation of noun predominance for word acquisition days and periods

We used the American database to investigate which indexes are good measures to evaluate noun predominance: the comprehension days, the production days, or the C2P indexes. To classify the words into nouns and verbs, we used Caselli's part-of-speech classification (Caselli et al.) and calculated the comprehension days, the production days, and the C2P indexes for the words except those whose C2P indexes couldn't be calculated.

Fig. 2 shows the noun and verb distributions for the comprehension days for the American database. The average number of comprehension days for verbs was 480, and for nouns it was 498, showing a small difference in the number of days between them ($p < 0.2$). In terms of the comprehension days, nouns do not predominate verbs. Fig. 3 shows the noun and verb distributions for the American database for the word production days. The average for verbs was 731 days, and for nouns it was 681 days, showing significant differences in the number of days ($p < 0.001$). Nouns predominate verbs in terms of word production days.

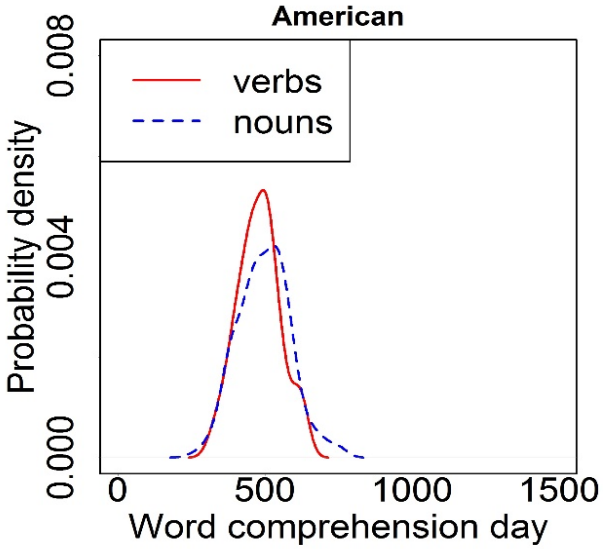


Fig. 2 American word distributions of nouns vs. verbs with respect to word comprehension days .

Considering the process of word acquisition, the word production day can be divided into two processes: the comprehension process and the process for the C2P index. From Fig. 2 we confirmed that word comprehension days do not contribute to noun predominance, suggesting that the C2P index period primarily contributes to noun predominance.

Fig. 4 shows the noun and verb distributions for the American database for the C2P index. The average C2P index of verbs was 251 days, and for nouns it was 183 days, showing significant differences in the number of days ($p < 0.001$). This result shows that in the periods between the comprehension and production days, nouns

strongly predominate verbs. The process from word comprehension to word production strongly affects noun predominance in word-learning.

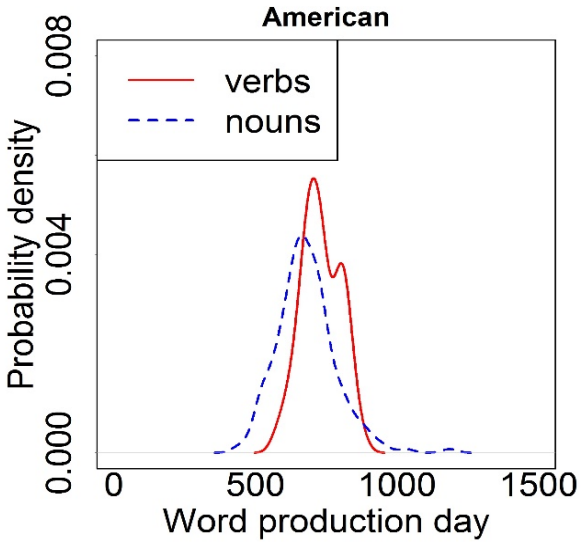


Fig. 3 American word distributions of nouns vs. verbs with respect to word production days .

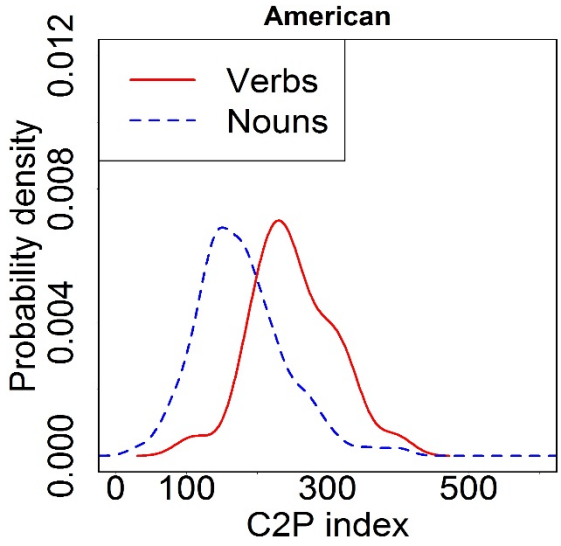


Fig. 4 American word distributions of nouns vs. verbs with respect to C2P index .

Fig. 5, Fig. 6 and Fig.7 show the noun and verb distributions for the Danish database for the comprehension day, the production day, and the C2P index. These results resemble those of the American database and also show that the process from word comprehension to word production strongly affects noun predominance in word-learning.

We calculated the comprehension days, the production days, and the C2P indexes for all of these languages to investigate the generality of this result using the cross-lingual database described in Section 2. Tables 1, 2, and 3 show the comprehension days, the production days, and the C2P indexes of nouns and verbs for the target languages. We also evaluated the value differences between nouns and verbs using a t-test. The results show noun predominance in the C2P indexes among all the languages except for Slovak and Turkish.

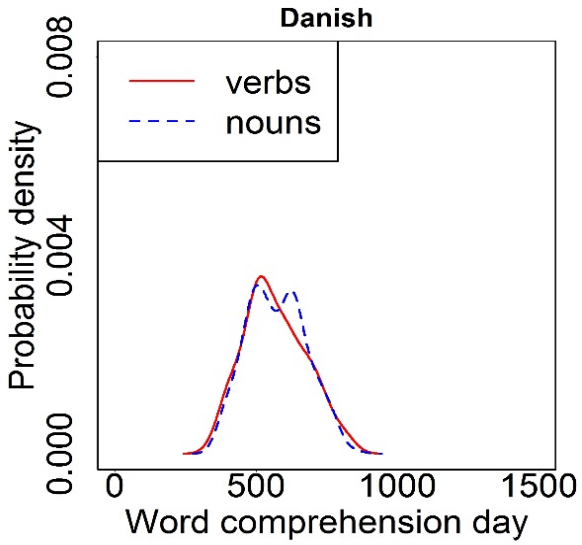


Fig. 5 Danish word distributions of nouns vs. verbs with respect to word comprehension days .

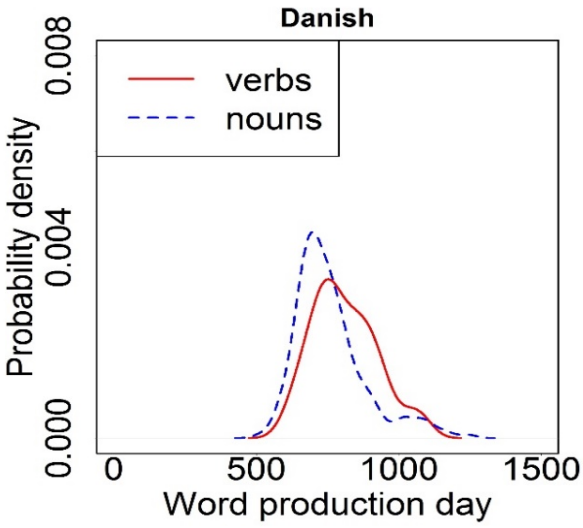


Fig. 6 Danish word distributions of nouns vs. verbs with respect to word production days .

This result suggests that the process between a word's comprehension and its production is a significant factor of predominance in noun learning by children and strongly supports the Gentner relational relativity hypothesis, because, to produce a verb, the surrounding words must be understood. The C2P index is a good measure to evaluate this factor because we confirmed that it evaluates the word acquisition difficulty of individual verbs and indicates that investigation of the C2P index of verbs might reveal the infant acquisition process of syntax.

Table 4 shows that C2P is also a good measure to examine language characteristics.

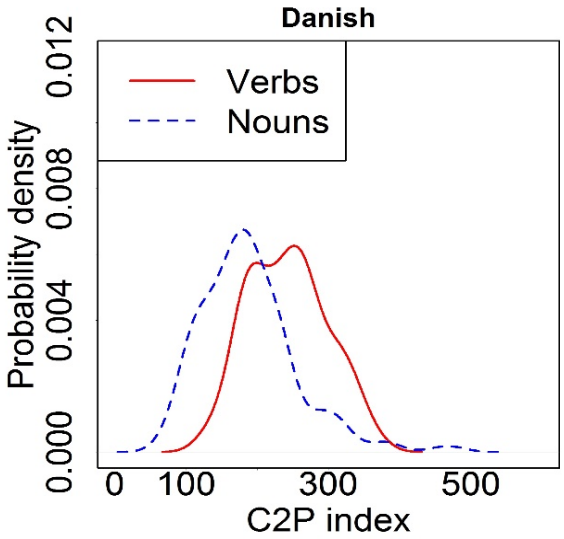


Fig. 7 Danish word distributions of nouns vs. verbs with respect to C2P index

6. Conclusion

This paper defines a comprehension-to-production (C2P) index that measures the difficulty of a partial word learning process from comprehension to production to investigate whether nouns have pre

dominance over verbs in word learning by children. We evaluated C2P indexes for cross-languages using cross-lingual CDI databases and confirmed that they indicate noun predominance in word learning and that the process between a word's comprehension and its production is a significant factor of predominance in noun learning by children. The experiment in this paper strongly supports the Gentner relational relativity hypothesis and argues that C2P is also a good measure to examine verb and language characteristics.

Table 2. Word comprehension days of nouns and verbs for target languages and t-test results.

Language	Word comprehension days		P-value
	Nouns	Verbs	
American	498	480	0.13
Danish	572	568	0.79
Mexican Spanish	509	483	0.12
Japanese	584	557	0.009
Croatian	439	453	0.35
French_(Quebec)	461	507	0.047
Italian	539	469	0.009
Korean	480	482	0.94
Latvian	425	431	0.83
Norwegian	497	528	0.24
Slovak	416	397	0.46
Turkish	468	469	0.95

Table 3. Word production days of nouns and verbs for target languages and t-test results.

Language	Word production days		P-value
	Nouns	Verbs	
American	681	731	4.8e-05
Danish	760	810	0.007
Mexican Spanish	736	817	4.0e-07
Japanese	773	806	0.002
Croatian	694	760	0.001
French_(Quebec)	659	770	6.0e-05
Italian	739	775	0.056
Korean	699	756	0.01
Latvian	697	775	0.003
Norwegian	688	756	0.018
Slovak	708	682	0.33
Turkish	744	761	0.53

Table 4. C2P indexes of nouns and verbs for target languages and t-test results.

Language	C2P index (days)		P-value
	Nouns	Verbs	
American	183	251	6.0e-10
Danish	188	242	2.8e-08
Mexican Spanish	227	334	4.0e-16
Japanese	188	249	5.7-11
Croatian	255	307	8.5e-05
French_(Quebec)	198	263	3.9e-06
Italian	200	306	4.0e-10
Korean	218	273	0.0024
Latvian	272	345	2.2e-05
Norwegian	191	227	0.002
Slovak	291	284	0.70
Turkish	276	291	0.18

7. Acknowledgements

A part of this study was supported by a Grant-in-Aid for Scientific Research (B) 17H02190.

8. Bibliographical References

Benedict, H. (1979). Early lexical development: Comprehension and production. Journal of child language, 6(2), 183-200.

Caselli, M. C., Bates, E., Casadio, P., Fenson, J., Fenson, L., Sanderl, L., et al. (1995). A cross-linguistic study of early lexical development. Cognitive Development, 10(2), 159-199.

Gentner, D. (1982). Why nouns are learned before verbs: Linguistic relativity versus natural partitioning. Language, 2(1), 301-334.

Gentner, D., & Boroditsky, L. (2001). Individuation, relativity, and early word learning. Language acquisition and conceptual development, 3, 215-256.

Maguire, M. J., Hirsh-Pasek, K., & Golinkoff, R. M. (2006). A Unified Theory ofWord Learning: Putting Verb Acquisition in Context. In Action meets word: How children learn verbs, 364-391.

Tardif, T. (1996). Nouns are not always learned before verbs: Evidence from Mandarin speakers' early vocabularies. Developmental Psychology, 32(3), 492.

9. Language Resource References

Ay, S. (2009). Essays on Turkish linguistics: proceedings of the 14th International Conference on Turkish Linguistics, August 6-8, 2008 (Vol. 79): Otto Harrassowitz Verlag.

Bleses, D., Vach, W., Slott, M., Wehberg, S., Thomsen, P., Madsen, T. O., et al. (2008). The Danish Communicative Developmental Inventories: validity and main developmental trends. Journal of child language, 35(3), 651-669.

Caselli, C., Casadio, P., & Bates, E. (1999). A comparison of the transition from first words to grammar in English and Italian. Journal of Child Language, 26(01), 69-111.

Dale, P. S., & Fenson, L. (1996). Lexical development norms for young children. Behavior Research Methods, Instruments, & Computers, 28(1), 125-127.

Frank, M. C., Braginsky, M., Yurovsky, D., & Marchman, V. A. (2017). Wordbank: An open repository for developmental vocabulary data. Journal of child language, 44(3), 677-694.

Jørgensen, R. N., Dale, P. S., Bleses, D., & Fenson, L. (2010). CLEX: A cross-linguistic lexical norms database*. Journal of Child Language, 37(2), 419.

Kovacevic, M., Babic, Z., & Brozovic, B. (1996). A Croatian language parent report study: Lexical and grammatical development. Paper presented at the Seventh International Congress for the Study of Child Language, Istanbul, Turkey.

Simonsen, H. G., Kristoffersen, K. E., Bleses, D., Wehberg, S., & Jørgensen, R. N. (2014). The Norwegian Communicative Development Inventories: Reliability, main developmental trends and gender differences. First Language, 34(1), 3-23.

Trudeau, N., & Sutton, A. (2011). Expressive vocabulary and early grammar of 16-to 30-month-old children acquiring Quebec French. First language, 31(4), 480-507.

Building a TOCFL Learner Corpus for Chinese Grammatical Error Diagnosis

Lung-Hao Lee[1,2], Yuen-Hsien Tseng[2], Li-Ping Chang[3]
[1]MOST Joint Research Center for AI Technology and All Vista Healthcare, National Taiwan University
[2]Graduate Institute of Library and Information Studies, National Taiwan Normal University
[3]Mandarin Training Center, National Taiwan Normal University
[1]No. 1, Sec. 4, Roosevelt Rd., Taipei 10617, Taiwan
[2,3]No. 162, Sec. 1, Heping East Rd., Taipei 10610, Taiwan
{lhlee, samtseng, lchang}@ntnu.edu.tw

Abstract

This study describes the construction of a TOCFL learner corpus and its usage for Chinese grammatical error diagnosis. We collected essays from the Test Of Chinese as a Foreign Language (TOCFL) and annotated grammatical errors using hierarchical tagging sets. Two kinds of error classifications were used simultaneously to tag grammatical errors. The first capital letter of each error tags denotes the coarse-grained surface differences, while the subsequent lowercase letters denote the fine-grained linguistic categories. A total of 33,835 grammatical errors in 2,837 essays and their corresponding corrections were manually annotated. We then used the Standard Generalized Markup Language to format learner texts and annotations along with learners' accompanying metadata. Parts of the TOCFL learner corpus have been provided for shared tasks on Chinese grammatical error diagnosis. We also investigated systems participating in the shared tasks to better understand current achievements and challenges. The datasets are publicly available to facilitate further research. To our best knowledge, this is the first annotated learner corpus of traditional Chinese, and the entire learner corpus will be publicly released.

Keywords: computer-assisted language learning, second language acquisition, grammatical error diagnosis, interlanguage analysis

1. Introduction

Annotating learners' inappropriate usage of written language is an important task for learner corpus research (Díaz-Negrillo and Fernández-Domínguez, 2006; Tono, 2003). From a linguistic perspective, annotated learner corpora are valuable resources for research in second language acquisition (Swanson and Charniak, 2013), foreign language teaching (Wang and Seneff, 2007), and contrastive interlanguage analysis (Granger, 2015). In engineering, such language resources can be used to develop natural language processing techniques for educational applications, such as automatic essay scoring (Yannakoudakis et al., 2011), assessment report generation (Sakaguchi et al., 2013), and native language identification (Malmasi and Dras, 2015).

Automated grammatical error detection and correction are important research directions and a number of competitions have been organized to encourage innovation (Leacock et al., 2014). For example, Helping Our Own (HOO) is a series of shared tasks for correcting errors in non-native English texts (Dale and Kilgarriff, 2011, Dale et al., 2012). The CoNLL 2013/2014 shared tasks aimed to correct grammatical errors for learners of English as a foreign language (Ng et al., 2013; 2014). The NLPTEA workshops had hosted a series of shared tasks for Chinese grammatical error diagnosis (Yu et al., 2014; Lee et al., 2015b; Lee et al., 2016b). Recently, the IJCNLP 2017 shared task 1 has focused on Chinese grammatical error diagnosis (Rao et al., 2017). All these competitions require annotated learner corpora for system development and evaluation.

To our best knowledge, only two previous studies have manually annotated learner corpora for Chinese as a foreign language (CFL). One is the HSK Dynamic Composition Corpus constructed by the Beijing Language and Culture University (Cui and Zhang, 2011; Zhang and Cui, 2013); the other is the Jinan Chinese Learner Corpus (Wang et al., 2015). The target language of these two studies is simplified Chinese, and no traditional Chinese learner corpus is available for public research. This pilot study thus aims to build such a learner corpus of traditional Chinese to expand research resources, especially for the study of linguistic differences or similarities among Chinese learners around the world.

This study annotated grammatical errors in texts collected from learner essays written as part of the Test Of Chinese as Foreign Language (TOCFL). The TOCFL learner corpus contained 2,837 essays written by learners originating from a total of 46 different mother-tongue languages. Chinese native speakers were trained to annotate these essays using hierarchical error tags, and 33,835 inappropriate grammatical usages were identified. We then used the Standard Generalized Markup Language to format annotated errors and their correct usages. Some of the annotated sentences were used for shared tasks hosted by the NLPTEA workshops. We also investigated systems participating in the shared tasks using various approaches for automated Chinese error diagnosis.

The rest of this paper is organized as follows. Section 2 briefly reviews existing learner corpora from around the world. Section 3 describes the process of TOCFL learner corpus annotation. Section 4 presents the annotation results. Section 5 investigates the shared tasks for Chinese grammatical error diagnosis based on the TOCFL learner corpus. Conclusions are finally drawn in Section 6.

2. Related Work

The Longman Learner Corpus is the first to collect essays and exam scripts written by learners of English (Gillard and Gadsby, 1998). The International Corpus of Learner English (ICLE) consists of argumentative essays written by advanced English learners from different native language backgrounds (Granger, 2003). The Cambridge Learner Corpus (CLC) is established to assist English Language Teaching/Training (ELT) publishers to create various learning aids including dictionaries and ELT course books (Nicholls, 2003). The NUS Corpus of Learner English (NUCLE) is annotated for the development and evaluation

of grammatical error correction systems (Dahlmeier et al., 2013).

Additional learner corpora exist for European languages. The Lund CEFLE is a leaner corpus of texts in French produced by adolescent Swedish learners of French (Granfeldt et al., 2006). The Error-Annotated German Learner Corpus (EAGLE) is a corpus of beginning learners with grammatical error annotation (Boyd, 2010). The ASK corpus is a learner corpus of Norwegian as a second language that contains essays collected from language tests (Tenfjord et al., 2006). The CzeSL corpus is a learner corpus of Czech that has been annotated using multi-layer error types (Hana et al., 2010). The Hungarian Learner Corpus is composed of student journals annotated for learner errors using tagging sets from different linguistic categories, including phonology, morphology and syntax (Dickinson and Ledbetter, 2012).

Recently, learner corpora have been established for Asian languages. The HSK Dynamic Composition Corpus contains simplified Chinese essays written by learners of Chinese, annotated with different error types (Cui and Zhang, 2011; Zhang and Cui, 2013). The Jinan Chinese Learner Corpus is a collection of texts produced for educational applications (Wang et al., 2015). Lang-8 contains a Japanese learner corpus extracted from a language learning and exchange social network service (Mizumoto et al., 2011). Linguistic properties of Korean particle errors have been outlined and annotated in collected learner writings (Lee et al., 2012).

The present study follows the research trend of worldwide learner corpora construction to build the TOCFL learner corpus. To the best of our knowledge, this is the first traditional Chinese learner corpus to be publicly available for research.

3. Annotation

The Steering Committee for the Test Of Proficiency-Huaya (SC-TOP) aims to develop and promote the Test Of Chinese as a Foreign Language (TOCFL) to assess the proficiency of CFL learners. The TOCFL writing test references the proficiency levels of the Common European Framework of Reference (CEFR) (Little, 2006). The testing principle is task orientation, which evaluates learners' ability to express their thoughts in the context of real-world situations. There are 4 available levels of the TOCFL writing test, which are described as follows.

(1) *Waystage level* (A2): test takers have to write a note and describe a story after looking at four pictures.
(2) *Threshold level* (B1): test takers are asked to write relatively detailed personal letters that describe their experiences and feelings about events they have encountered.
(3) *Vantage level* (B2): test takers are asked to write a functional letter highlighting definite purposes or to develop an argument to express personal opinions of specific events.
(4) *Proficiency level* (C1): test takers are asked to write an essay or report that gives reasons in support or against a particular point of view or explains provided figures and tables.

Test takers choose an exam level based on their current perceived level of Chinese proficiency. The proficiency-level evaluation is based on the appropriateness of the test takers' responses to situational tasks, compositional structure and completeness, syntax correctness, and the use of a suitably wide range of appropriate vocabulary. Each evaluation is conducted by at least two Chinese teachers and is then scored on a 0-5 point scale, where a score of 3 is a passing grade.

We collected learners' essays that had been given a score of 3 or above. Lower-scoring essays were ignored because of the difficulty in interpreting the learners' intended meaning and to annotate possible errors. In addition to learners' written texts for the TOCFL test, we collected all accompanying metadata including the corresponding CEFR level, evaluated score, and learner's native language. The TOCFL writing test is computer-based, and Chinese-typing ability is a requirement for all test takers. Spelling errors of Chinese characters frequently arise from confusion among multiple-character words that are phonologically and visually similar but semantically distinct. Spelling errors were corrected in our collected learners' written texts.

We then annotated grammatical errors to analyze inappropriate linguistic usages. Hierarchical tag sets were used in annotating grammatical errors (Lee et al., 2016a). Table 1 shows two kinds of error classifications used to simultaneously to tag grammatical errors. The first capital letter denotes the coarse-grained surface differences, while the subsequent lowercase letters denote the find-grained linguistic category. The coarse-grained error types originate from comparing erroneous sentences with the correct usages. There are 4 error types: missing, redundant, incorrect selection, and word ordering errors of linguistic components (also called PADS error types, denoting errors of Permutation, Addition, Deletion, and Selection). The fine-grained error types focus on representing linguistic concepts. A total of 36 error types were distributed into word-level errors (16 cases), grammatical function-level errors (11 cases), sentence pattern-level errors (7 cases), and mixture errors (2 cases).

Native Chinese-speaking annotators were trained to follow our annotation guidelines for the error-tagging task. There were cases in which our annotators knew something was wrong, but were unable to select the appropriate annotation. In such cases, the annotators discussed the errors amongst themselves to seek agreement on the appropriate tag. A tagging editor was used to help annotators to insert error tags and rewrite the contextually correct usage for the learner corpus (Lee et al., 2014). This editor provides an error analysis function to further assist annotators in easily finding incorrect or inconsistent tagging instances during the annotation process. Some example annotations are given in Table 2. In sentence 1(a), there is a missing object "他" (he), so it was annotated using a tag [Mobj]. The correction with English translation is shown in 1(b). In sentence 2(a), the tag [Rdet] represents that "第" (the) is a redundant word that should be deleted. The word "兩" (two) in sentence 3(a) is an incorrect numeral (denoting as the tag [Snum]), where the contextually correct one should be "二" (two/second). Sentence 4(a) has a word ordering error. The phrase "七點鐘" (seven o'clock) had been tagged as [Wtime], that means this phrase should precede the word "起床" (get up).

After annotating grammatical errors and providing their correct usages, we used the Standard Generalized Markup Language (SGML; ISO 8879:1986) to format the learner texts into the following four parts.

(1) *Essay*: unique identification number, writing style, article title, the obtained score, and writing date.
(2) *Learner*: learner's mother-tongue language and his/her corresponding language proficiency of CEFR.
(3) *Text*: learners' original written texts.
(4) *Mistake*: paragraph in which a grammatical error occurs along with the positions of the starting and ending character, in which each character or punctuation mark has a counting position value of 1.

Target Modification Taxonomy	
Missing (*M*), Redundancy (*R*), Incorrect Selection (*S*), Word Ordering Error (*W*)	
Linguistic Category Classification	
Word-level	action verb (*v*), auxiliary (*aux*), stative verb (*vs*), noun (*n*), pronoun (*pron*), conjunction (*conj*), preposition (*p*), numeral (*num*), demonstrative (*det*), measure word (*cl*), sentential particle (*sp*), aspectual particle (*asp*), adverb (*adv*), structural particle (*de*), question word (*que*), plural suffix (*plural*)
Grammatical Function-level	subject (*sub*), object (*obj*), noun phrase (*np*), verb phrase (vp), preposition phrase (*pp*), modifier (*mod*), time expression (*time*), place expression (*loc*), transitivity (*tran*), separable structure (*vo*), [numeral /determiner+measure] phrase (*dm*),
Sentence Pattern-level	complex noun clause (*rel*), 把 sentence (*ba*), 被 sentence (*bei*), 讓 sentence (*rang*), 是 sentence (*shi*), 有 sentence (*you*), other patterns (*pattern*)
Mixture	formation (*form*), ambiguity of syntactic or meaning (*sentence*)

Table 1: Error Tags for grammatical error annotations.

1(a) * 他請我教 [*Mobj*] 日文
1(b) 他請我教他日文
(He asked me to teach him Japanese.)

2(a) * 我會在第 [*Rdet*] 一樓等你
2(b) 我會在一樓等你
(I will wait for you on the first floor.)

3(a) * 傑克是一個兩 [*Snum*] 年級的高中生
3(b) 傑克是一個二年級的高中生
(Jack is a second-year senior high school student.)

4(a) * 他平常起床七點鐘 [*Wtime*]
4(b) 他平常七點鐘起床
(He usually gets up at seven o'clock.)

Table 2: Example annotations and their corrections.

Figure 1 shows an example of our built TOCFL learner corpus. This essay was given an identification number "0612", and was written in "2009年5月" ('May 2009') by a "B1"-level learner with "韓語" ('Korean') as his/her L1. The topic of this "記敘文" ('narrative-style') essay is "最難忘的購物經驗" ('My memorable shopping experience'), and was given a score of "4". A number of errors were annotated. For example, the first error occurred in the first paragraph starting and ending at position 105, meaning that a word "是" ('is') is missing between "有時候" ('sometimes') and "比" ('than'). It was annotated using an error tag "Mshi", in which the first capital letter denotes the coarse-grained surface difference, while the subsequent lowercase letters denote the fine-grained linguistic category. In the fourth paragraph, "那時候" ('that time') was put in a wrong position resulting in a "Wtime" error tag denoting a time phrase in the wrong order and the word "了" ('le') was annotated as "Rasp" denoting a redundant aspectual particle that should be removed. The annotation can yield the correct sentence "記得那時候讀書讀得不太好" ('Recalling that I didn't study well at that time').

```
<ESSAY id="0612" style="記敘文" title="最難忘的購物經驗" score="4" date="2009年5月">
<LEARNER>
<L1>韓語</L1>
<CEFL>B1</CEFL>
</LEARNER>
<TEXT>
<P>
…… 在星巴克讀書，雖然需要付錢，可是我覺得有時候比圖書館好多的地方。
</P>
<P>
我喝咖啡的時候，常喝美式咖啡。……
</P>
<P>
有一天，我去星巴克，……
</P>
<P>
…… 那時候記得讀書讀得不太好了。以後也不常去那裏的星巴克了。
</P>
</TEXT>
<MISTAKE paragraph="1" start_off="105" end_off="105">
<TYPE>Mshi</TYPE>
<CORRECTION>是</CORRECTION>
</MISTAKE>
……
<MISTAKE paragraph="4" start_off="33" end_off="37">
<TYPE>Wtime</TYPE>
<CORRECTION>記得那時候</CORRECTION>
</MISTAKE>
<MISTAKE paragraph="4" start_off="45" end_off="45">
<TYPE>Rasp</TYPE>
<CORRECTION>null</CORRECTION>
</MISTAKE>
</ESSAY>
```

Figure 1: An essay in our TOCFL learner corpus.

4. Results

Table 3 provides descriptive statistics for our TOCFL learner corpus. The B1 level occupies near a half of the corpus. In total, about one million characters were collected and annotated covering 62 different essay titles.

CEFR	#Title	#Essay	#Char	Ratio%
A2	21	850	131,684	29.96%
B1	24	1,388	540,286	48.93%
B2	14	503	280,239	17.73%
C1	3	96	50,079	3.38%
Total	62	2,837	1,002,288	100%

Table 3: Statistics of TOCFL learner corpus.

Figure 2 shows the distribution of the learners' mother-tongue languages, with the top 10 languages accounting for 88%, and another 36 languages accounting for the remaining 12%. Slightly more than one fourth of the sample spoke Japanese as their first language, followed by English, Vietnamese, Korean, and Indonesian.

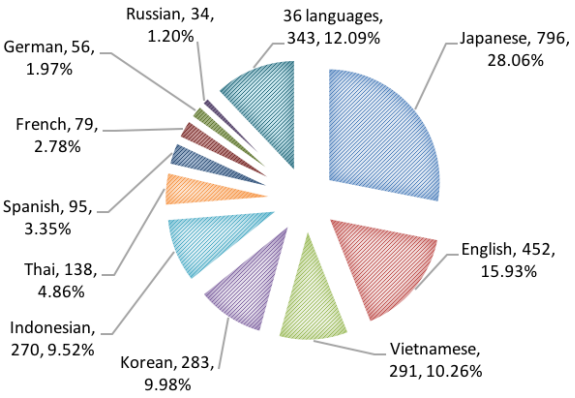


Figure 2: Distribution of learners' first languages.

Figure 3 shows the distribution of the 33,835 total errors. The most common error type was incorrect selection of linguistic components (13,278 cases or 39% of the total), followed by missing errors (12,155 cases), redundancy (6,066 cases) and word ordering errors (2,336 cases). Figure 4 further shows the histogram of the top 10 error tags (accounting for about 47% of all errors) among the total 124 distinct error tags. The most common errors were related to the incorrect selection of verbs (Sv) and nouns (Sn). Half of these errors are categorized as missing word-level linguistic components.

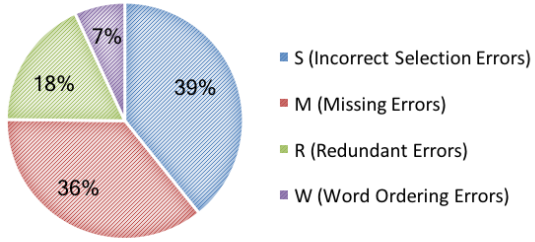


Figure 3: Distribution of all error tags.

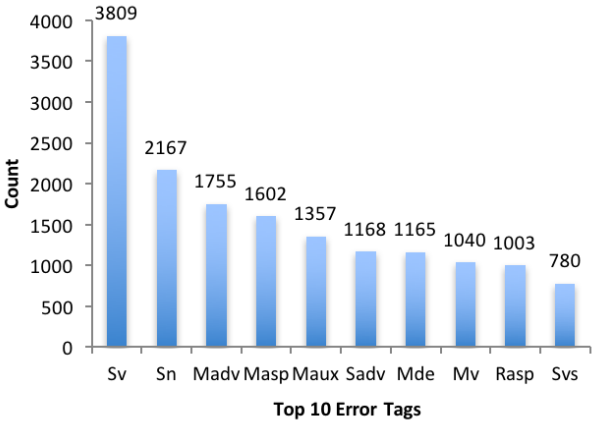


Figure 4: Distribution of the top 10 error tags.

We also developed and implemented a special-purpose retrieval system for the TOCFL learner corpus, which is available online at http://tocfl.itc.ntnu.edu.tw, to facilitate interlanguage analysis for second language acquisition (Lee et al., 2015a). Take the '讓' (*rang4* 'to make') sentence for example, we can choose the main error type S and the sub-type *rang*. Figure 5 shows the search results. We found that learners usually confuse '讓' (*rang4* 'to make') with '把' (*ba3* 'disposal marker'), '對' (*dui4* 'to someone'), and '給' (*gei3* 'to give'). It is difficult to check each sentence individually to find all erroneous and confused usages and this retrieval system effectively reduces required analysis time. Moreover, we can limit the search results for specific words, such as '把' (*ba3* 'disposal marker'), which will benefit observation and analysis. In addition to the filtering function, we can also select specific learners' attributes such as the learners' mother tongue or their L2 proficiency, thus increasing the ease and efficiency of interlanguage analysis.



Figure 5: Search results of rang4 sentences.

5. Shared Tasks

A part of our annotated TOCFL learner corpus was used to organize a series of shared tasks for Chinese grammatical error diagnosis hosted by NLPTEA workshops. These tasks seek to develop NLP techniques to automatically identify grammatical errors in Chinese sentences. In the NLPTEA 2014 workshop (Yu et al., 2014), the developed system is used to check a sentence for grammatical errors and, if found, identify the error type. In the NLPTEA 2015 workshop (Lee et al., 2015b), the system was required to additionally indicate the range of occurring errors. In the

NLPTEA 2016 workshop (Lee et al., 2016b), the task is basically the same, except that the target sentence may contain more than one error, and the HSK Dynamic Composition Corpus was also included for the task.

In addition to the provided data sets, participating systems were allowed to use the other language resources which should be found in the system description reports. We briefly describe them as follows.

For the NLPTEA 2014 shared task, the KUAS&NTNU system used manually constructed and automatically generated rules to identify grammatical errors (Chang et al., 2014). The UDS system designed an n-gram frequency-based approach to detect grammatical errors (Zampieri and Tan, 2014). The NTOU system defined several features to train SVM classifiers for error detection (Lin and Chan, 2014). The NCYU system adopted word segmentation and part-of-speech tagging techniques to identify missing and redundant error types (Yeh et al., 2014). To compensate for data insufficiency for supervised machine learning, the TMU system extracted a Chinese learner corpus from the Lang-8 website, and used it as a parallel corpus for phrase-based statistical machine translation for grammatical error identification (Zhao et al. 2014).

For the NLPTEA 2015 shared task, The HITSZ system presented an ensemble learning based method to detect and identify grammatical errors (Xiang et al. 2015). The SCAU system adopted a hybrid model by integrating rule-based and n-gram statistical methods for grammatical error diagnosis (Wu et al., 2015b). The CYUT team built an error diagnosis system based on the Conditional Random Fields (CRF) (Wu et al., 2015a). The NTOU system proposed two sentence likelihood functions based on frequencies of Google n-grams to diagnose grammatical errors (Lin, & Chen, 2015). The NCYU system also used statistical word and part-of-speech patterns based CRFs to detect grammatical errors (Yeh et al., 2015). The TMU examined corpus augmentation and explored syntax-based and hierarchical phrase-based translation models for use in this task (Zhao et al. 2015).

For the NLPTEA 2016 shared task, the ANO system and CYUT-III system diagnosed grammatical errors based on the CRF (Chen et al., 2016a; Liu et al., 2016) along with word order sensitive embedding approaches (Chou et al., 2016). The NTOU system generated and scored correction candidates for grammatical error diagnosis (Chen et al., 2016b). The HIT system adopted long short-term memory (LSTM) neural networks to identify grammatical errors (Zheng et al., 2016). The PKU system presented a model based on bidirectional LSTM (Huang, & Wang, 2016). The NCYU system proposed the structure of the recurrent neural network using LSTM to detect grammatical errors (Yeh et al., 2016). The YUN-HPCC system built single word embeddings based convolutional neural networks and LSTM neural networks for this task (Yang et al., 2016).

In terms of performance, a good system should have a high F1 score and a low false positive rate. Overall, none of the participating systems provided satisfactory results when measuring different metrics (i.e. False Positive Rate, Accuracy, Precision, Recall and F1), indicating the difficulty of developing systems for effective grammatical error diagnosis, especially in the context of CFL.

Recently, neural network-based deep learning techniques have shown promising results in identifying Chinese grammatical errors. However, a large amount of training data is needed to train and fine-tune the parameters of these complex networks. Another challenge is raised by the biased distribution of error types in the training instances reflecting real-world errors caused by CFL learners, emphasizing the importance of annotated learner corpora in tackling this research problem.

Shared tasks can be used to meaningfully compare the performance of various techniques using the same data sets and evaluation metrics. All evaluations encourage the proposal of unorthodox and innovative approaches which could lead to breakthroughs. Following each shared task, all gold standard data sets and evaluation tools are made publicly available for research purposes as follows:

- NLPTEA 2014 shared task: http://ir.itc.ntnu.edu.tw/lre/nlptea14cfl.htm
- NLPTEA 2015 shared task: http://ir.itc.ntnu.edu.tw/lre/nlptea15cged.htm
- NLPTEA 2016 shared task: http://ir.itc.ntnu.edu.tw/lre/nlptea16cged.htm

6. Conclusions

This study presents the construction of a TOCFL learner corpus based on 2837 learner essays annotated using hierarchical tagging sets. Our error tags are used to simultaneously represent two kinds of grammatical error classifications. The first capital letter of each error tags denotes the coarse-grained surface differences, while the subsequent lowercase letters denote the fine-grained linguistic categories. Native Chinese-speaking annotators were trained to follow our annotation guidelines for the error-tagging task, and identified 33,835 grammatical errors. We then used SGML to format the annotations and their corresponding corrections along with learners' accompanying metadata. Parts of the annotated TOCFL learner corpus have been used to organize shared tasks for Chinese grammatical error identification. We also investigate participating systems to better understand the current capabilities and challenges. The shared-task datasets are publicly available to facilitate further research. We plan to release the entire learner corpus in fully annotated SGML format in the hopes that this resource can facilitate future development in related research areas.

7. Acknowledgements

This study was partially supported by the Ministry of Science and Technology (MOST), Taiwan under the grant: MOST 103-2221-E-003-013-MY3, MOST 105-2221-E-003-020-MY2, MOST 106-2221-E-003-030-MY2, and MOST 107-2634-F-002-019-. We also thank all the annotators for their efforts in the error-tagging task.

8. Bibliographical References

Boyd, A. (2010). EGALE: an error-annotated corpus of beginning learner German. Proceedings of 7th International Conference on Language Resources and Evaluation (LREC'10), pages 1-6.

Chang, T.-H., Sung, Y.-T., Hong, J.-F. and Chang, J.-I (2014). KNGED: a tool for grammatical error diagnosis of Chinese sentences. Proceedings of the 1st Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 48-55.

Chen, P.-L., Wu, S.-H., Chen, L.-P. and Yang, P.-C. (2016a). CYUT-III system at Chinese grammatical error diagnosis task. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 63-72.

Chen, S.-H., Tsai, Y.-L. and Lin C.-J. (2016b). Generating and scoring correction candidates in Chinese grammatical error diagnosis. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 131-139.

Chou, W.-C., Lin, C.-K., Liao, Y.-F. and Wang, Y.-R. (2016). Word order sensitive embedding features/conditional random field-based Chinese grammatical error detection. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 73-81.

Cui, X. and Zhang, B. (2011). The principles for building the "International Corpus of Learner Corpus". *Applied Linguistics*, 2011(2):100-108.

Dahlmeier, D., Ng. H. T. and Wu, S. M. (2013). Building a large annotated corpus of learner English: the NUS corpus of learner English. Proceedings of the 8th Workshop on the Innovative Use of NLP for Building Educational Applications (BEA'13), pages 22-31.

Dale, R. and Kilgarriff, A. (2011). Helping our own: The HOO 2011 pilot shared task. Proceedings of the 13th European Workshop on Natural Language Generation (NLG'11), pages 1-8.

Dale, R., Anisimoff, I. and Narroway, G. (2012). HOO 2012: A report on the preposition and determiner error correction shared task. Proceedings of the 7th Workshop on the Innovative Use of NLP for Building Educational Applications (BEA'12), pages 54-62.

Díaz-Negrillo, A. and Fernández-Domínguez, J. (2006). Error tagging systems for learner corpora. *Revista española de lingüística aplicada (RESLA)*, 19:83-102.

Dickinson, M. and Ledbetter. S. (2012). Annotating errors in a Hungarian learner corpus. Proceedings of 8th International Conference on Language Resources and Evaluation (LREC'12), pages 1659-1644.

Gillard, P. and Gadsby, A. (1998). Using a learners' corpus in compiling ELT dictionaries. In S. Granger, & G. Leech (Eds.), *Learner English on Computer*. Addison Wesley Longman, London, pages 159-171.

Granfeldt, J., Nugues, P., Ågren, M., Thulin, J., Persson, E. and Schlyter, S. (2006). CEFLE and direct profil: a new computer learner corpus in French L2 and a system for grammatical profiling. Proceedings of 5th International Conference on Language Resources and Evaluation (LREC'06), pages 565-570.

Granger, S. (2003). The international corpus of learner English: a new resource for foreign language learning and teaching and second language acquisition research. *TESOL*, 37(3):538-546.

Granger, S. (2015). Contrastive interlanguage analysis: a reappraisal. *International Journal of Learner Corpus Research*, 1(1):25-74.

Hana, J., Rosen, A., Škodová, S. and Štindlová, B. (2010). Error-tagged learner corpus of Czech. Proceedings of the 4th Linguistic Annotation Workshop (LAW'10), pages 11-19.

Huang, S. and Wang, H. (2016). Bi-LSTM neural networks for Chinese grammatical error diagnosis. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 148-154.

Leacock, C., Chodorow, M., Gamon, M. and Tetreault, J. (2014). Automated grammatical error detection for language learners, 2nd edition, Morgan & Claypool Publishers.

Lee, L.-H., Chang, L.-P., Liao, B.-S., Cheng, W.-L. and Tseng, Y.-H. (2015a). A retrieval system for interlanguage analysis. Proceedings of 23rd International Conference on Computers in Education (ICCE'15), pages 599-601.

Lee, L.-H., Chang, L.-P. and Tseng, Y.-H. (2016a). Developing learner corpus annotation for Chinese grammatical errors. Proceedings of the 20th International Conference on Asian Language Processing (IALP'16), pages 254-257.

Lee, L.-H., Lee, K.-C., Chang, L.-P., Tseng, Y.-H., Yu, L.-C. and Chen, H.-H. (2014). A tagging editor for learner corpus annotation and error analysis. Proceedings of 22nd International Conference on Computers in Education (ICCE'14), pages 806-808.

Lee, L.-H., Rao, G., Yu, L.-C., Xun, E., Zhang, B. and Chang, L.-P. (2016b). Overview of the NLP-TEA 2016 shared task for Chinese grammatical error diagnosis. Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 40-48.

Lee, L.-H., Yu, L.-C. and Chang, L.-P. (2015b). Overview of the NLP-TEA 2015 shared task for Chinese grammatical error diagnosis. Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 1-6.

Lee, S.-H., Dickinson M. and Israel, R. (2012). Developing learner corpus annotation for Korean particle errors. Proceedings of the 6th Linguistic Annotation Workshop (LAW'12), pages 129-133.

Lin, C.-J. and Chan, S.-H. (2014). Description of NTOU Chinese grammar checker in CFL 2014. Proceedings of the 1st Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 75-78.

Lin, C.-J. and Chen, S.-H. (2015). NTOU Chinese grammar checker for CGED shared task. In Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 15-19.

Little, D. (2006). The Common European Framework of Reference for languages: content, purpose, origin, reception, and impact. *Language Teaching*, 39(3):167-190.

Liu, Y., Han, Y., Zhuo, L. and Zan, H. (2016). Automatic grammatical error detection for Chinese based on conditional random field. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 57-62.

Malmasi, S. and Dras, M. (2015). Large-scale native language identification with cross-corpus evaluation. Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'15), pages 1403-1409.

Mizumoto, T., Komachi, M., Nagata, M. and Matsumoto, Y. (2011). Mining revision log of language learning SNS for automated Japanese error correction of second language learners. Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP'11), pages 147-155.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H. and Bryant, C. (2014). The CoNLL-2014 shared task on grammatical error correction. Proceedings of the

18th Conference on Computational Natural Language Learning (CoNLL'14), Shared Task, page 1-14.

Ng, H. T., Wu, S. M., Wu, Y., Hadiwinoto, C. and Tetreault, J. (2013). The CoNLL-2013 shared task on grammatical error correction. Proceedings of the 17th Conference on Computational Natural Language Learning (CoNLL'13), Shared Task, page 1-12.

Nicholls, D. (2003). The Cambridge Learner Corpus – error coding and analysis for lexicography and ELT. Proceedings of the Corpus Linguistics 2003 Conference (CL'03), pages 572-581.

Rao, G., Zhang, B. and Xun, E. (2017). IJCNLP-2017 Task 1: Chinese Grammatical Error Diagnosis. Proceedings of the 8th International Joint Conference on Natural Language Processing (IJCNLP'17), Shared Tasks, pages 1-8.

Sakaguchi, K., Arase, Y. and Komachi, M. (2013). Discriminative approach to fill-in-the-blank quiz generation for language learners. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL'13), pages 238-242.

Swanson, B. and Charniak, E. (2013). Extracting the native language signal for second language acquisition. Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'13), pages 85-94.

Tenfjord, K., Meurer, P. and Hofland, K. (2006). The ASK Corpus – a language learner corpus of Norwegian as a second language. Proceedings of 5th International Conference on Language Resources and Evaluation (LREC'06), pages 1821-1824.

Tono, T. (2003). Learner corpora: design, development and applications. Proceedings of the Corpus Linguistics 2003 Conference (CL'03), pages 800-809.

Wang, C. and Seneff, S. (2007). Automatic assessment of student translations for foreign language tutoring. Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'07), pages 468-475.

Wang, M., Malmasi, S. and Huang, M. (2015). The Jinan Chinese learner corpus. Proceedings of the 10th Workshop on the Innovative Use of NLP for Building Educational Applications (BEA'15), pages 118-123.

Wu, S.-H., Chen, P.-L., Chen, L.-P., Yang, P.-C. and Yang, R.-D. (2015a). Chinese grammatical error diagnosis by conditional random fields. Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 7-14.

Wu, X., Huang, P., Wang, J., Guo, Q., Xu, Y. and Chen, C. (2015b). Chinese grammatical error diagnosis system based on hybrid model. Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 117-125.

Xiang, Y., Wang, X., Han, W. and Hong, Q. (2015). Chinese grammatical error diagnosis using ensemble learning. Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15). pages 99-104.

Yannakoudakis, H., Briscoe, T. and Medlock, B. (2011). A new dataset and method for automatically grading ESOL texts. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL'11), pages 180-189.

Yang, J., Peng, B., Wang, J., Zhang, J. and Zhang, X. (2016). Chinese grammatical error diagnosis using single word embedding. In Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 155-161.

Yeh, J.-F., Hsu, T.-W. and Yeh, C.-K. (2016). Grammatical error detection based on machine learning for Mandarin as second language learning. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 140-147.

Yeh, J.-F., Lu, Y.-Y., Lee, C.-H., Yu, Y.-H. and Chen, Y.-T. (2014). Detecting grammatical error in Chinese sentence for foreign. Proceedings of the 1st Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 62-68.

Yeh, J.-F., Yeh, C.-K., Yu, K.-H., Li, Y.-T. and Tsai, W.-L. (2015). Conditional random field-based grammatical error detection for Chinese as second language. In Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 105-110.

Yu, L.-C., Lee, L.-H. and Chang, L.-P. (2014). Overview of grammatical error diagnosis for learning Chinese as a foreign language. Proceedings of the 1st Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 42-47.

Zampieri, M. and Tan, L. (2014). Grammatical error detection with limited training data: the case of Chinese. Proceedings of the 1st Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 69-74.

Zhang, B. and Cui, X. (2013). Design concepts of the construction and research of the inter-language corpus of Chinese from global learners. *Language Teaching and Linguistic Study*, 2013(5):27-34.

Zhao, Y., Komachi, M. and Ishikawa, H. (2014). Extracting a Chinese learner corpus from the Web: grammatical error correction for learning Chinese as a foreign language with statistical machine translation. Proceedings of the 1st Natural Language Processing Techniques for Educational Applications (NLPTEA'14), pages 56-61.

Zhao, Y., Komachi, M. and Ishikawa, H. (2015). Improving Chinese grammatical error correction with corpus augmentation and hierarchical phrase-based statistical machine translation. Proceedings of the 2nd Natural Language Processing Techniques for Educational Applications (NLPTEA'15), pages 111-116.

Zheng, B., Che, W., Guo, J. and Liu, T. (2016). Chinese grammatical error diagnosis with long short-term memory networks. Proceedings of the 3rd Natural Language Processing Techniques for Educational Applications (NLPTEA'16), pages 49-56.

MIAPARLE: Online training to improve discrimination of stress contrasts

Jean-Philippe Goldman, Sandra Schwab
University of Geneva
jeanphilippegoldman@gmail.com, sandra.schwab@unige.ch

Abstract

MIAPARLE is a public web application that is designed to offer a range of CAPT (computer-aided pronunciation teaching) tools for L2 learners. Besides helping language learners to reduce their foreign-accentedness, the goal of the platform is to test these tools, gather feedback and improve them according to their educational impact. In this article, we describe one particular training tool that focuses on stress perception. This tool is particularly useful for speakers whose L1 is a fixed-stress language, such as French. These speakers have difficulties perceiving and discriminating stress contrasts. To help them with this so-called stress 'deafness', the methodology used in the training is based on successive questions in which a visual pattern is associated with the sound of a lexical item. After successively completing their pre-tests, training and post-tests, the participants are given their improvement score. The performance of the training is evaluated by comparing the learner's results at the pre- and post-test stages. Various methodological parameters, such as the number of training items or the number of visual patterns are tested in parallel in order to quantify their teaching efficiency, and to optimise the overall teaching impact.

Keywords: CAPT, pronunciation training, prosody, stress deafness

1. Introduction

Second language learners tend to imprint the prosody of their mother language onto the second language (L2) (e.g., Barquero et al., 2014; Muñoz, 2010). This prosodic cross-language transfer is often combined with segmental transfers, which can lead to the presence of a foreign accent. A foreign accent can not only hamper communication between learners and natives (like in Anderson, Munro & Derwing 1988), but it can also affect the credibility of learners and how they are evaluated by others; this can sometimes lead to social discrimination (e.g., Dailey et al., 2005, Lev-Ari & Keysar, 2010; , Purnell et al. 1999). Moreover, despite the crucial role of prosody in speech flow segmentation (e.g., Cutler 1984; Cutler & Butterfield 1992 for English; Bagou 2002 for French), it is rarely taught in language courses, even in foreign-language pronunciation courses.

In the framework of *computer-assisted pronunciation teaching*, the goal of this project is to develop a methodology which improves the prosodic skills of L2 learners in order to reduce cues of foreign accent. After many years of research (Neri et al. 2002, Busà 2008, Chun 2012), there has recently been a growing interest in CAPT applications (e.g. for French[1], Spanish[2], English[3],and Norwegian[4]). However, there have been few applications that focus specifically on prosodic features (among others,, Vakil & Trouvain, 2015 for German stress; Sztahó et al, 2014, for Hungarian intonation, stress and speech rhythm; Niebuhr et al. 2017, for visualization methods of prosody).

This paper focuses on the tool used to assist in discrimination of stress contrasts. Although this method is designed for learners with any first language (L1), training on stress contrasts is mostly useful for speakers with a fixed lexical stress native language (such as French, Czech, or Hungarian). For example, French-speaking listeners tend to experience difficulties in perceiving stress contrasts in a second language such as Spanish. These difficulties are the basis of the *stress deafness* hypothesis (e.g., Dupoux et al., 1997.). According to this hypothesis, the degree of *stress deafness* is related to the stress properties of the L1; more specifically, to the nature of lexical stress (free or fixed). In a free-stress language such as Spanish or English, lexical stress has a distinctive function, since it distinguishes segmentally identical words, such as in Spanish *<u>nú</u>mero* (['numeɾo], English *(the) number*) and *nu<u>me</u>ro* ([nu'meɾo], English *I number*)[5] . As a consequence, speakers of a free-stress language need to encode stress position in their mental representation of the words. On the other hand, the position of stress in a fixed-stress language such as French is not variable, and thus not contrastive. Consequently, the stress information does not need to be stored in the lexical representation. The *stress deafness* hypothesis claims that speakers of fixed-stress languages have difficulties in perceiving stress contrasts in free-stress L2s since they are not able, or at least not trained, to encode stress information in their mental lexicon (e.g., Dupoux et al., 1997). Nevertheless, as showed by our recent research (Schwab & Dellwo, 2016b), French learners are able to

[1] http://www.coelang.tufs.ac.jp/mt/fr-swiss/dmod/index_en.html
[2] http://www.enterate.unam.mx/Articulos/2007/abril/sarahi.html
[3] http://www.carnegiespeech.com/products/nativeaccent.php
[4] https://orgesuniversitetet.no/prosjekt/computer-assisted-listening-and-speaking

[5] The underlined syllable in these examples, and in the rest of the paper, corresponds to the stressed syllable.

improve their stress detection ability after a 4-hour training session. Although the current methodology focuses on isolated words only, our goal is to extend it to larger groups of words and continuous speech flow. Based on its promising results, our aim is that our tool for training stress-contrast discrimination will give L2 learners the bootstrapping required to start encoding stress information in their mental lexicon.

2. MIAPARLE

MIAPARLE is a web application that hosts various tools and activities dedicated to pronunciation training. It is based on PyBossa, a micro-tasking crowdsourcing platform. PyBossa is developed and maintained by SciFabric, runs on flask micro-framework and is well designed to be responsive on smaller screens such as those in tablets and smartphones.

In this paper, we describe the training tool for learning stress contrasts in Spanish. The application presents teaching material such as audio samples of lexical items pronounced by native speakers and corresponding clip arts. The whole procedure is organized as follows: the participants' initial performance is evaluated with 2 different pre-tests, the participants then follow one of two possible full training pathways (A or B), and eventually take post-tests in order to quantify their improvement level in the perception of lexical stress.

2.1 Pre- and post-tests

The pre- and post-tests are used to evaluate the training and comprise similar tasks. Two types of exercises are available: Localisation and Odd-One-Out, as described below. Each exercise has 15 items in the pre-tests (so the participants go through 30 items in total during two pre-tests). During the post-test, each activity shows the same 15 items and 15 additional items, in order to test the generalisation of the training. Thus the learner is shown 60 items during the post-tests.

	Pre-test	Training	Post-test
	localisation (15)	A or B	localisation (15+15)
	odd-one-out (15)		odd-one-out (15+15)
Total	30	104 or 90	60

Table 1. In parenthesis, the number of items for pre-test, training and post-test

2.1.1 Localisation of a given stress pattern

In this exercise, learners have to localise a given stress pattern. They hear trials of different words produced by a native Spanish female speaker. Each word is associated with a representative drawing. They have to answer to the following question: "Which word has stress on the X syllable". For example, as shown in Figure 1, the learners hear the words "*río*" (river), "*mesa*" (table) and "*champú*" (shampoo) and have to indicate which word has stress on the final syllable (i.e., "dernière syllabe " in French). The learners indicate their response by clicking on the drawing corresponding to the word with the given stress pattern. Before answering, they have the opportunity to listen to the words as many times as they want. They do not receive any feedback after each trial, but their score is displayed at the end of the exercise.

The task can target the final, penultimate or antepenultimate syllable. The difficulty of the task increases during the exercise when words produced by another (male) speaker are introduced, and by increasing the number of words presented in the trials (2, 3, 4 words) or/and the number of syllables in the words (2, 3, and 4 syllables).

Fifteen trials were used in the pre-test. In addition to these 15 trials, we used 15 new trials in the post-test in order to evaluate the generalisation of the training method to new items.



Figure 1: Localisation of a given stress pattern. Question "Click on the word which has stress on the final syllable"

2.1.2 Odd-One-Out

In this exercise, the learners perform an Odd-One-Out task (Schwab & Dellwo, 2016a). They hear a trial of three segmentally identical stimuli (e.g., *numero*). Among them, two stimuli present the same stress pattern (e.g., stress on the penultimate syllable) and one (i.e., the *odd*) presents a different stress pattern (e.g., stress on the final syllable). After hearing each trial, learners have to indicate which of the three elements is the *odd* one by clicking on the corresponding option on the screen (Mot 1, Mot 2, Mot 3; in English Word 1, Word 2, Word 3).

We used triplets of trisyllabic CV.CV.CV and CVC.CV.CV Spanish words that differ with respect to the stressed syllable. Each triplet is composed of a proparoxytone word (i.e., stress on the antepenultimate syllable; in trisyllabic words, it means that the first syllable is stressed, e.g., *número*), a paroxytone word (i.e., penultimate syllable stressed word, e.g., *numero*) and an

oxytone word (i.e., final syllable stressed word, e.g., *numeró*). Two native speakers of Spanish (one female and one male) produced the six words twice. The trials presented to the learners are composed of two words with the same stress pattern (e.g., *número*) and of one word with a different stress pattern (e.g., *numero*). The word with the stress accentual pattern is the odd element.

All combinations of stress contrasts are tested (i.e., proparoxytone target word paired with paroxytone words or with oxytone words; paroxytone target word paired with proparoxytone words or with oxytone words; oxytone target word paired with proparoxytone words or with paroxytone words). The position of the odd word within the trial is assigned randomly. The difficulty of the task increases in the second part of the exercise when words produced by the male speaker are introduced. Fifteen trials were also used in the pre-test for this task. In addition to these 15 trials, we used 15 new trials in the post-test for the same reason as in 2.1.1.

2.2 Training

The training *per se* is the principal, and therefore the most time-consuming, activity in the whole process. We present two versions of the training here. The first version is an adapted replication of the training used in Schwab & Dellwo (2016b), while the second version has a more explicit methodology with respect to prosodic patterns.

2.2.1 Version A

In this training phase, learners perform a shape/word matching task. They hear a word and 4 of 6 possible shapes appear on the screen (see Figure 2). Learners have to click on the shape which they think corresponds to the word they hear. After giving their response they receive feedback: they hear the word again and the correct shape is indicated with a green frame. If they had clicked on an incorrect shape, it is indicated with a red frame. This feedback enables the learners to learn the correspondence between the words and the shapes.

Two triplets of trisyllabic Spanish words are used in this training: *cáscara* (shell), *cascara* (en. that he cracked), *cascará* (he will crack) and *módulo* (module), *modulo* (I modulate), *moduló* (he modulated); these are associated to 6 the shapes as shown in Figure 2.

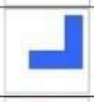
Shape	Word
	cáscara
	cascara
	cascará
	módulo
	modulo
	modulo

Figure 2: Shapes and words used in the training.

As can be observed, each triplet is composed of a proparoxytone (i.e., *cáscara* and *módulo*), a paroxytone (i.e., *cascara* and *modulo*) and an oxytone word (i.e., *cascará* and *modulo*). The six words are produced by another female native Spanish speaker (i.e., different to the one who recorded the stimuli in the pre-tests).

The training is divided into three blocks. Each word (e.g., cáscara) is presented six times per block (6 words x 6 times = 36 times per block). Among the four shapes that appear on the screen, only one corresponds to the word and the three others are distractors. Among the three distractors, one corresponds to a word with a different stress pattern (e.g., *cascara*) and the two other shapes correspond to two words from the other triplet (e.g., *módulo* and *modulo*). The position of the shapes on the screen is assigned randomly. The order of presentation of the trials is determined semi-randomly in such a way that no more than two same stress patterns and no more than two members of each triplet follow each other.

2.2.2 Version B

In version B of the training phase, learners perform a shape/stress pattern matching task. They hear a word and three shapes appear on the screen (see Figure 3). The shape iconically represents a trisyllabic word with stress on one of the three syllables (see Figure 3). The participants have to click on the shape they think corresponds to the stress pattern they hear. After giving their response, they receive feedback: they hear the word again and the correct shape is indicated with a green frame. If they had clicked on an incorrect shape, it is indicated with a red frame. The feedback enables the learners to learn the correspondence between the stress patterns and the shapes.

Ten triplets of trisyllabic Spanish words are used in this training pathway. Each triplet is composed of a proparoxytone (e.g., *cálculo*), a paroxytone (e.g., *calculo*) and an oxytone word (e.g., *calculó*). The 30 words are produced by a female native Spanish speaker (the same who produced the stimuli in version A). The three stress patterns are associated to three shapes, as shown in Figure 3.

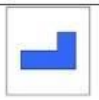
Shape	Stress pattern
	Proparoxytone (e.g., cálculo)
	Paroxytone (e.g., calculo)
	Oxytone (e.g., calculo)

Figure 3: Shapes and stress patterns used in the training version B.

Similarly to version A, the training is divided into three blocks. Each word (e.g., *cálculo*) is presented once per block. Among the three shapes that appear on the screen, only one corresponds to the target stress pattern and the two others correspond to the other two stress patterns. The position of the shapes on the screen is assigned randomly. The order of presentation of the trials is determined semi-randomly in such a way that no more than two same stress patterns, and no more than two members of each triplet, follow each other.

3. Preliminary results

The platform was launched in mid-September 2017. Six first-testers went through the whole training process (pre-tests, training and post-tests) and gave us some useful qualitative feedback in respect of user experience.

The platform uses an A/B testing mechanism, so that every other registered participant is led alternatively to version A or version B of the training. For version A, the mean scores (i.e correct identification of the appropriate shape) increased from 56% at the pre-test to 72% at the post-test, i.e., an improvement of 16%. For version B, the mean scores increased from 66% to 74%, i.e., an improvement of 8%. In both versions, we could notice that the best improvement was as high as 20%.

We also noticed a ceiling effect at 80% for the post-test scores, which limited the degree of possible improvement for speakers with a higher ability level in L2 Spanish. Finally, we estimated that the total time that the participants spent on the whole training was 30 minutes.

4. Conclusion

With the implementation of the web platform, we now have a framework which allows us to test several methodological configurations in parallel and compare their teaching impact. For example, we can address the question of implicit vs explicit teaching (as seen in Schwab 2016b) in order to test the effectiveness of explicit instructions. In the explicit training, explicit instruction and explanations about Spanish lexical stress contrasts would be given to the participants, whereas no mention to Spanish lexical stress would be made in the implicit training.

Our two main short-term tasks are 1) to continue taking the qualitative feedback of our testers into account in order to refine the didactic instructions and the general user experience, and 2) to deploy the platform to a larger number of participants in order to consolidate our preliminary results. We are also considering expanding this platform for any L1 by localising it in other languages. Further we also aim to expand the L2 material in other free-lexical-stress languages such as English and German, as well as tone languages such as Mandarin. Finally, another set of exercises that focus on production - rather than perception - is currently under development and will allow participants to speak and have their intonation automatically compared with reference stimuli.

5. Acknowledgements

This project is supported by the INNOGAP proof-of-principle fund from UNITEC, University of Geneva.

6. References

Anderson-Hsieh, J. & Koehler, K. (1988). The Effect of Foreign Accent and Speaking Rate on Native Speaker Comprehension. *Language Learning*, 38, 561–613.

Bagou, O., Fougeron, C. & Frauenfelder, U. H. (2002). Contribution of prosody to the segmentation and storage of "words" in the acquisition of a new mini-language. *Speech Prosody 2002*, Aix-en-Provence, France, 159-162.

Barquero, Mª A., Racine, I., Baqué, L. & Schwab, S. (2014). La estructuración acentual: estudio comparativo en la interlengua español-francés. Caso de la lectura. In: Y. Congosto Martín, M. L. Montero Curiel & A. Salvador Plans (Eds.), *Fonética experimental, educación superior e investigación*. (Vol. II, pp. 9-28). Madrid: Arco/Libros.

Busà, M. G. (2008). New perspectives in teaching pronunciation. In *From DIDACTAS to ECOLINGUA. An ongoing research project on translation and corpus*

linguistics. Trieste: Università degli Studi di Trieste, 165-182.

Chun, D. M. (2012). Computer-Assisted Pronunciation Teaching. In C. A. Chapelle (Ed.), *The encyclopedia of applied linguistics*. Oxford: Wiley-Blackwell. doi:10.1002/9781405198431.wbeal0172

Cutler, A. & Butterfield, S. (1992). Rhythmic cues to speech segmentation: Evidence from juncture misperception. *Journal of Memory and Language*, 31, 218-236.

Cutler, A. (1984). Stress and accent in language production and understanding. In: D. Gibbon & H. Richter (eds.) *Intonation, Accent and Rhythm: Studies in Discourse Phonology*. Berlin: Walter de Gruyter.

Dailey, R. M., Giles, H. & Jansma, L. L (2005). Language attitudes in an Anglo-Hispanic context: the role of the linguistic landscape. *Language and Communication*, 25, 27–38.

Dupoux, E., Pallier, C., Sebastián, N. & Mehler, J. (1997). A destressing 'deafness' in French? Journal of Memory and Language, 36(3), 406-421.

Lev-Ari, S. & Keysar, B. (2010). Why don't we believe non-native speakers? The influence of accent on credibility. *Journal of Experimental Social Psychology*, 46, 1093-1096.

Muñoz Garcia, M. (2010). *La perception et la production de l'accent lexical de l'espagnol par des francophones: aspects phonétiques et psycholinguistes*. Thèse de doctorat, U. Toulouse 2/Universitat Autònoma de Barcelona.

Munro M. J. & Derwing T. M. (1995). Foreign accent, comprehensibility, and intelligibility in the speech of second language learners. *Language Learning*, 45, 73–97.

Neri, A., Cucchiarini, C., Strik, H., Boves, L. (2002). The pedagogy-technology interface in Computer-Assisted Pronunciation Training, *Computer-Assisted Language Learning* 15, 5: 441-467.

Niebuhr, O. (2017), Comparing visualization techniques for learning second language prosody. First results. *International Journal of Learner Corpus Research* 3:2 (2017), 250–277.

Purnell, T, Idsardi W. & Baugh J. (1999). Perceptual and phonetic experiments on American English dialect identification. *Journal of Language and Social Psychology*, 18, 10-30.

Sztahó, D., Kiss, G. Czap, L. & Vicsi, L. (2014). A Computer-Assisted Prosody Pronunciation Teaching System. Fourth Workshop on Child Computer Interaction (WOCCI 2014), Singapore, September 19, 2014.

Schwab, S. & Dellwo, V. (2016a). The use of the Odd-One-Out task in the study of the perception of lexical stress in Spanish by German-speaking listeners. *Proceedings of the 6th International Conference on Speech Prosody*, Boston, May, 31 – June, 3, 2016.

Schwab, S. & Dellwo, V. (2016b). Explicit versus implicit training methods in the learning of Spanish accentual contrasts by French- and German-speaking listeners. *Second Language Prosody Workshop*, November, 18-19, 2016, Graz, Austria.

Vakil, A. S. & Trouvain, J. (2015). Automatic classification of lexical stress errors for German CAPT. SLaTE 2015, Leipzig, September 4–5, 2015.

ESCRITO – An NLP-Enhanced Educational Scoring Toolkit

Torsten Zesch and Andrea Horbach
Language Technology Lab, Department of Computer Science and Applied Cognitive Science
University of Duisburg-Essen, Germany
{torsten.zesch, andrea.horbach}@uni-due.de

Abstract

We propose ESCRITO, a toolkit for scoring student writings using NLP techniques that addresses two main user groups: teachers and NLP researchers. Teachers can use a high-level API in the teacher mode to assemble scoring pipelines easily. NLP researchers can use the developer mode to access a low-level API, which not only makes available a number of pre-implemented components, but also allows the user to integrate their own readers, preprocessing components, or feature extractors. In this way, the toolkit provides a ready-made testbed for applying the latest developments from NLP areas like text similarity, paraphrase detection, textual entailment, and argument mining within the highly challenging task of educational scoring and feedback. At the same time, it allows teachers to apply cutting-edge technology in the classroom.

Keywords: automatic assessment, text classification und software toolkit

1. Introduction

Scoring student writings is a core task for teachers, which requires a lot of manual effort. Using assisted or automated scoring might have a tremendous impact on the quality of teaching, as it potentially shifts the focus from tedious assessment tasks to communicating knowledge. In the case of free-form student writings, like essays or answers to factual questions, correctly assessing a response is still a challenging task. There are typically many ways how a correct response can be expressed, and the variety of responses is increased even more by orthographic and grammatical deviations, which occur frequently in student writings. Therefore, automatically scoring student writings usually involves various other tasks central to NLP such as spell checking, grammatical error correction, POS tagging, paraphrase recognition, textual entailment, or argument mining.

In recent years, attempts have been made to link free-form scoring to some of the above-mentioned fields. For example, the SemEval 2013 Student Response Analysis Task (Dzikovska et al., 2013) combined short-answer scoring with recognizing textual entailment. More fundamental research along such lines would be desirable. In addition to basic research, more practical experimentation in the classroom is needed as well. These goals require the involvement of two disjunct groups of people: *NLP researchers* (who develop scoring methods) and *teachers* (who bring the methods to the classroom). However, existing implementations are often very specific to certain use cases and datasets. That makes it difficult for teachers without a technical background to use them out-of-the-box, let alone apply them to new data. At the same time, the application field is highly complex, which discourages NLP researchers from testing the latest developments in this field.

We thus present ESCRITO, the Educational SCoRIng TOolkit, a toolbox for free-text scoring based on natural language processing and machine learning, which caters to both user groups. ESCRITO has two main goals: (i) to enable teachers to quickly build free-text scoring systems and apply them in real-life scenarios, and (ii) to provide an application testbed for the integration and evaluation of NLP algorithms. Teachers can access ESCRITO using a high-level API that allows them to specify and execute scoring pipelines on their own data following best-practices in the field. Research scientists will find a low-level API, which allows them to access, customize, and extend all relevant aspects of automatic scoring including preprocessing, feature extraction, and machine learning setup.

We ensure reproducibility of results through detailed automated documentation of experimental setups. We also have designed ESCRITO to be as language-independent as possible. It has been successfully applied to data in various languages. All parts of ESCRITO have already been used in research projects concerning essay scoring (Zesch et al., 2015b; Horbach et al., 2017c), spellchecking on learner data (Horbach et al., 2017a), clustering (Zesch et al., 2015a), and neural short-answer scoring (Riordan et al., 2017). This shows the wide applicability of the framework and that state-of-the-art approaches can be easily modeled within the framework.

Related Work To the best of our knowledge, there are no other publicly available general-purpose scoring frameworks addressing either programmers or practitioners. Proprietary systems such as e-rater (Attali and Burstein, 2004) can only be used commercially and as a sort of black-box. While a number of scoring implementations are publicly available, such as Neural Essay Assessor (Taghipour and Ng, 2016), an essay scoring system for Swedish (Östling et al., 2013), or clustering-based scoring Zesch et al. (2015a), these implementations are typically centered around a specific dataset and method, and not straight-forward to extend or apply to new data. Equally, there are approaches for supporting teachers with free-text answers in MOOCs. An example is a plugin for the learning management system Moodle which sorts answers by their similarity to a reference answer (Pado and Kiefer, 2015).

2. Educational Scoring Toolkit

Educational free-text scoring is often tackled as a classical supervised learning task with the goal to assign a label to some piece of text written by a learner in response

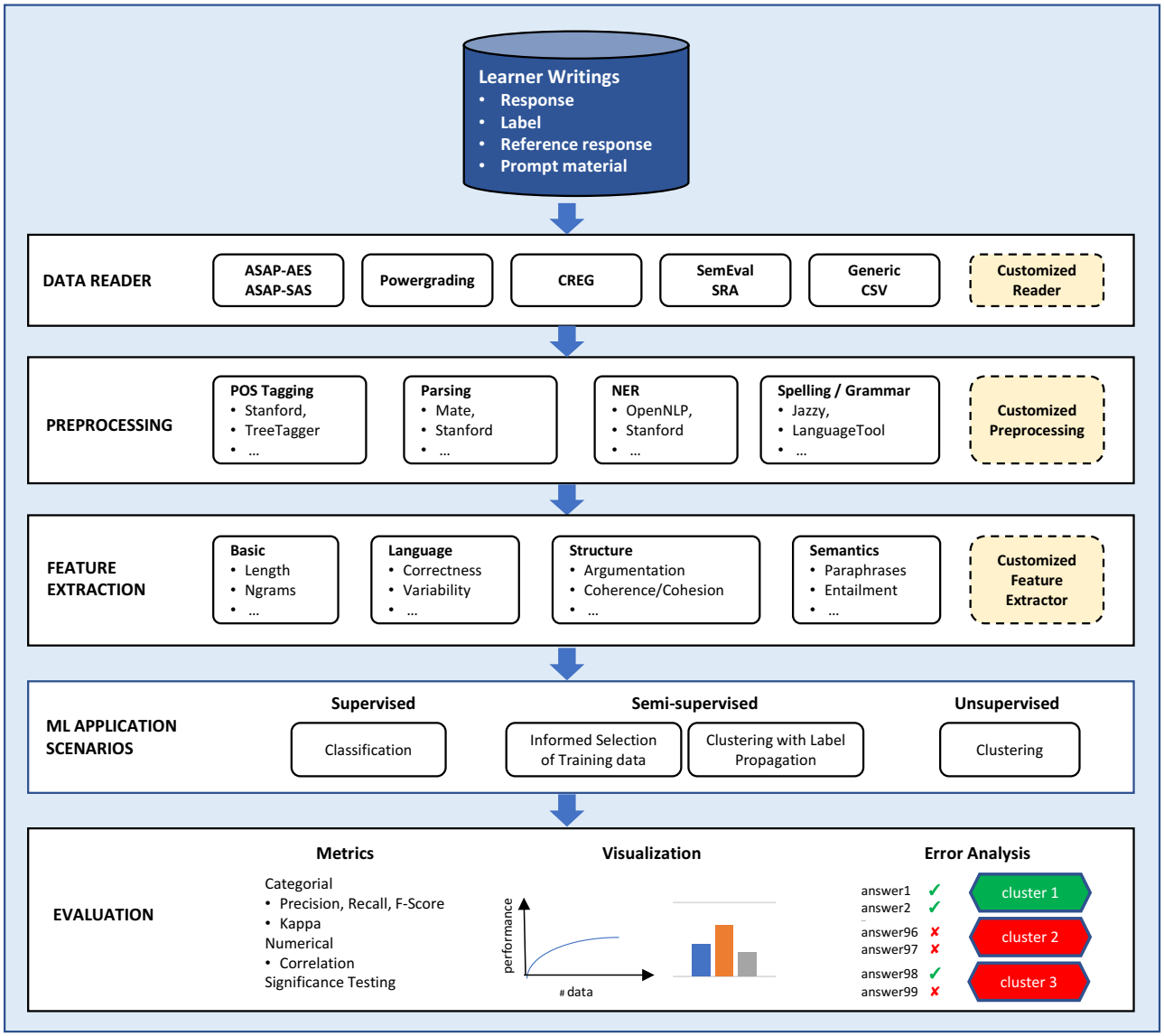


Figure 1: Overview of the ESCRITO architecture.

to a prompt. This text can either be an answer to a so-called short-answer question, asking for answers consisting of only a few words or a few sentences, or it can be a longer text, such as an essay consisting of several hundred words. Figure 1 shows examples for the range of free-text exercises and the variety of possible learner answers for individual prompts.

The labels assigned may be numeric as well as categorical (correct/incorrect or more fine-grained diagnostic labels). Their value can either be based on content alone (content-scoring) or on content and form (typically in holistic scores for essay scoring). Typically, the prompt in response to which answers are given is available. Predefined reference answers, i.e. sample solutions, are available in some cases. These additional materials can also be leveraged in automatic scoring; for example, to compare whether an answer to be scored is similar to the reference answer or to identify domain-specific vocabulary useful for spell-checking.

In order to model this complex setup, we build ESCRITO on top of DKPro TC (Daxenberger et al., 2014), an UIMA-based open-source framework that provides easy access to various algorithms for supervised text classification and extensive parameter documentation enabling reproducible research. ESCRITO extends DKPro TC with respect to the specific needs of educational scoring applications: it offers easy access to existing educational datasets, various preprocessing options, state-of-the-art scoring features, evaluation and visualizations for common scoring scenarios, as well as options to integrate new data and to customize existing or to add new preprocessing components and feature extractors. Figure 1 gives an overview of the system's architecture.

2.1. Easy Access to Existing Datasets

Datasets often come each in their own format, such that data preparation can be tedious and time-consuming. We provide pre-implemented readers for state-of-the-art datasets, such as ASAP-AES[1] for essay scoring and ASAP-

[1] https://www.kaggle.com/c/asap-aes

PHRASE-LENGTH LEARNER ANSWERS FOR CONTENT SCORING (POWERGRADING - PROMPT 1)
QUESTION: What is one right or freedom from the First Amendment of the U.S. Constitution?

LEARNER ANSWERS:

- **correct:** freedom of speech
- **correct:** free speech
- **correct:** freedom to talk freely
- **correct:** freedome of religeon
- **incorrect:** the right to bear arms
- **incorrect:** life

PARAGRAPH-LENGTH LEARNER ANSWERS FOR CONTENT SCORING (ASAP-2 - PROMPT 1)
QUESTION: After reading the groups procedure, describe what additional information you would need in order to replicate the experiment. Make sure to include at least three pieces of information.

LEARNER ANSWERS:

- **3 points:** Some additional information you will need are the material. You also need to know the size of the contaneir to measure how the acid rain effected it. You need to know how much vineager is used for each sample. Another thing that would help is to know how big the sample stones are by measureing the best possible way.
- **1 point:** After reading the expirement, I realized that the additional information you need to replicate the expireiment is one, the amant of vinegar you poured in each container, two, label the containers before you start yar expirement and three, write a conclusion to make sure yar results are accurate.
- **0 points:** The student should list what rock is better and what rock is the worse in the procedure.

TEXT-LENGTH LEARNER ANSWER FOR SCORING OF PERSUASIVE ESSAYS (ASAP-1 - PROMPT 1)
INSTRUCTION: Write a letter to your local newspaper in which you state your opinion on the effects computers have on people. Persuade the readers to agree with you.

LEARNER ANSWER:

- **6 points:** Dear, @ORGANIZATION1 I think the effects that computers do on people are really positive. Computers can be used for all sorts of things. Examples like finding things out about history. People that changed the world and other information. Computers give the power for children to learn. For example, their are lots of websites that offer online tutoring or good ways to help you pass school. Other positive way is online dateing sites. You can meet new people and is a good way to make life even better. Popular sites like @CAPS1, @CAPS2, @CAPS3, and so on make a good way to keep in touch with friends from your past, or even make new ones. But the most that I think thats the best in my opinion is going to school online. Once your done with colloge and you are a nuse, for an example you can get a higher degree like a registered nurse then being a @ORGANIZATION2. I think computers has a positive effect on people.

Table 1: Examples for free-text tasks asking for answers of very different complexity.

SAS[2] for short-answer scoring, the Powergrading short-answer dataset (Basu et al., 2013), the dataset by Mohler and Mihalcea (2009), the Student Response Analysis (SRA) dataset (Dzikovska et al., 2013), and the German CREG corpus (Meurers et al., 2011) to facilitate this task. Furthermore, we also support generic educational datasets, consisting of at least a set of learner answers, each with an ID and a score. We provide generic readers for such datasets in line-based formats (such as CSV) which novice users may use to integrate their own data without having to write their own reader. These readers support different properties of the data, such as categorical and numerical scoring labels. They also support datasets where a prompt text is available and both datasets without reference answers as well as datasets where one or several reference answers are provided and used for comparison with the learner answer. We additionally provide interfaces for expert users to easily integrate new dataset readers.

2.2. Preprocessing

For preprocessing, ESCRITO integrates all tools available through DKPro Core (Eckart de Castilho and Gurevych, 2014), which provides access to a large number of interchangeable and interoperable components in several languages. In the high-level mode, the system automatically checks which preprocessing components are needed for a selected language and feature setting. It assembles a standard preprocessing pipeline according to best practices, so that a teacher does not have to worry about the implementation details of linguistic analysis. In the expert mode, the

[2] https://www.kaggle.com/c/asap-sas

user can select from the complete range of DKPro Core preprocessing components and assemble a pipeline according to their needs while they get a warning if a feature extractor requires a preprocessing step that is not included by the user.

In ESCRITO, we add preprocessing components according to the special requirements of the task. Learner answers typically show a much higher orthographic variability than standard language. Spelling errors are typically ignored in content-scoring while their absence or presence is a useful feature for form-based scoring. Therefore, learner answers can optionally be normalized before feature extraction or language errors are simply annotated to be consumed later by a feature extractor that extracts the number and type of errors in an answer as features. Therefore, we add spell-checking components based on spellcheckers like Jazzy[3]. Our method (Horbach et al., 2017b) automatically extends their lexicon using prompt-specific material, e.g. the words in the prompt the answer refers to. Thus, lexical items that a learner likely referred to when creating their answers appear in the dictionary of a spell-checker even if they come from a very specific domain and would not appear in standard spelling dictionaries. Furthermore, we implement spell-checking mechanisms that prefer domain-vocabulary over non-domain-vocabulary when automatically correcting spelling errors.

In addition to spell-checking, we provide optional marking of elements in a learner answer as being copied from the question in the prompt. Consider the example question *Where was Peter born?* where the rheme-only answer *in Berlin* and the full answer *Peter was born in Berlin* convey the same content. Marking this material provides the option to ignore it later in some feature extractors.

Even more task-specific preprocessing components can easily be plugged in by wrapping them as UIMA annotators.

2.3. State-of-the-art Scoring Features

ESCRITO provides a wide variety of state-of-the-art features from both essay and content scoring as well as means for easily integrating newly developed ones.

N-gram features are known to be a powerful feature group determining the content of an answer. We extend the n-gram feature extractors provided through DKPro TC by integrating different normalization techniques and determining n-grams not only on the basis of words and characters, but also based on dependency triples. **Length features**, such as number of tokens or sentences are also known to have a highly predictive power, especially when answers are written under a time limit.

Several feature groups target the language of an answer in terms of complexity and correctness: **Linguistic Complexity** is especially important for essay scoring. We measure linguistic complexity through variance on the lexical level (type-token-ratio), on the syntactic level (distribution of POS tags, average and maximal depth of parse trees, number and type of subordinate clauses), and via a number of readability measures from DKPro TC readability. We extract **language correctness** features about the nature and frequency of different language errors identified by our own spell-checking methods as well as grammar and stylistic errors found by LanguageTool.[4]

We target the structure of an essay through **Coherence and Cohesion features** which measure the usage of connectives in an answer, as well as the content overlap between adjacent sentences. We also provide features on the **argumentative structure** of an essay, e.g. through the number and distribution of claims and citations in a text. Especially in short-answer scoring, prompts often include a target answer and answers are typically correct if they are similar to or entail the target answer. Based on DKPro Similarity (Bär et al., 2013), we provide **textual similarity** feature extractors that measure similarity between the learner and the target answer on the surface level (token overlap and string similarity measures), on the syntactic level (overlap of dependency triples), or the semantic level (e.g. using concept alignment).

In case that there are several possible reference answers for a learner answer, we offer different ways of combining the evidence from these reference answers by either taking the maximum, minimum, average, or all feature values produced when comparing a learner answer to the individual target answers. These differences can, for example, be important when handling both similarity scores as well as distances. New features can easily be implemented and integrated by using interfaces for features either based on the learner answer text alone or its comparison with a prompt text or reference answer.

Integration of Deep Learning Deep learning methods became widely used in various NLP areas including educational free-text scoring (Taghipour and Ng, 2016; Riordan et al., 2017) and are often a very powerful alternative to traditional shallow learning methods. DKPro TC has been extended (Horsmann and Zesch, 2018) to also provide interfaces to widely used deep learning frameworks including Keras (Chollet and others, 2015), DeepLearning4J [5], and Dynet (Neubig et al., 2017), while ensuring reproducibility and easy preprocessing through DKPro TC. We integrate this extension to make sure deep learning methods can be used in ESCRITO.

2.4. Machine Learning Scenarios

We specify commonly used experimental setups that allow for both supervised and unsupervised machine learning scenarios according to the needs of the two user groups.

From an NLP researcher's perspective, the supervised case with labeled train and test data is certainly the most common one. We provide setups for both cross-validation and train-test setups with the option to choose from different machine learning tools as provided by Weka (Hall et al., 2009) and wrapped through DKPro TC.

Additionally, learning curve evaluations come in handy when one wants to assess how many training instances are needed until no further improvement can be reached with more data. ESCRITO implements learning curves which

[3] https://github.com/reckart/jazzy

[4] https://www.languagetool.org/

[5] http://deeplearning4j.org

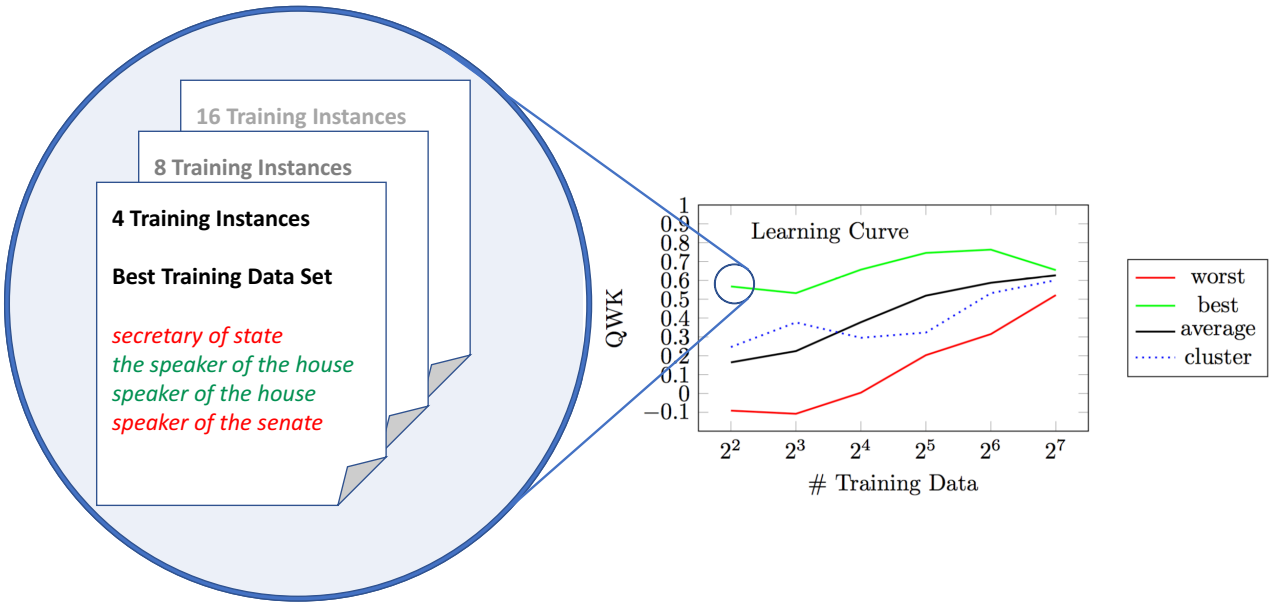


Figure 2: An example for the evaluation of a learning curve experiment.

simulate that only a limited number of training data is available and provides best, worst, and average learning curves across large numbers of randomly selected items.

Real-life scoring scenarios often fall somewhere in between the supervised and the unsupervised case: some part of the data is typically unlabeled and should be labeled by the tool (and afterwards potentially re-checked by a teacher). Therefore, we assume that scenarios where a model is to be trained on existing labeled training data and applied on new unlabeled test-data will occur frequently; for example, when the same or similar prompts have already been used and scored in an earlier exam and are now administered to a new cohort of students. In such a case, we can provide evaluations on the training data using cross-validation to allow a performance estimation and present classification results on the unlabeled test data with additional confidence scores. This allows a teacher to re-visit the automatically scored data manually, while concentrating on the uncertain cases and checking with higher preference.

When a teacher wants to score data from a new domain with no existing labeled training examples, two options are possible. First, in a completely unsupervised scenario, items are clustered according to the similarity between answers, such that clusters are formed that contain similar answers. A teacher can then inspect clusters and assign scoring labels either to whole clusters or to individual members of a cluster. In this way, they can save annotation time and effort and are at the same time informed about common misconceptions in the student answers (Basu et al., 2013). Second, a teacher might want to label some, but not all their data, train a classifier and then re-label the complete dataset (or only the so-far unlabeled data). In this scenario, we provide methods to select the items to be labeled in an informed way, for example by selecting items so that they cover as much of the feature space as possible. As an alternative, we provide methods from *active learning*, where items are dynamically selected in a way that a machine learner profits most from them, such that human annotation effort is reduced (Horbach and Palmer, 2016).

2.5. Evaluation and Visualization

We report frequently used evaluation metrics depending on the type of labels used in a dataset: accuracy, quadratically and linearly weighted kappa, precision, recall, F-measure as well as correlation scores such as Pearson(Pearson, 1895) and Spearman(Spearman, 1904). To facilitate error analysis, incorrectly classified items (i.e. false positives and false negatives for a class) are written to separate files and can easily be inspected. Besides plain text result files, ESCRITO also writes tables and figures in LaTeX formatting or images. Additionally, we provide mechanisms to compare different experiments and perform significance testing on them.

In the unsupervised case, scoring results per answer are presented to the user for manual inspection where the user can customize the data to be ordered, e.g. by scoring confidence of the system, by the assigned class, or clustered based on item similarity so that similar items can be reviewed together.

Figure 2 shows as an example part of the evaluation of a learning curve experiment. The user gets output in the form of a chart showing the performance of the best, worst and average training data configurations resulting from a large number of random samples of training data as well as the performance of selecting training data based on cluster centroids. The user can inspect for every point on the curve

which particular item selection lead to that result.

3. Example Use Cases

We will now have a closer look how the toolkit can be used, which we exemplify based on two example users.

A **textual entailment researcher** who has developed a new, promising algorithm might want to evaluate it as part of a educational scoring pipeline. With ESCRITO, she can easily assemble a baseline system that is evaluated on multiple datasets and then compare the results against an augmented system, that uses entailment as an additional feature. She can even easily get access to the instances on which the two system configurations differed in order to examine the cases in which the entailment-based system outperformed (or underperformed) the baseline.

An *educational science researcher* wants wants to know whether automatic scoring in a particular test discriminates against non-native speaking students, for example because they might use a wording not so frequently used in the training data. To do so, he can easily train and cross-validate scoring models for his test with a set of annotated answers. By inspecting the automatic scores in comparison to the human annotations (for example in the form of files with lists of false postive and false negative answers), he can easily determine whether answers from non-native speakers are more likely to be misclassified by a certain algorithmic setup.

A **teacher**, who wants to score student writings in Italian as a consistency check in addition to his own manual scoring, can use on of the pre-configured setups to train a model and apply it to a new cohort. The data only needs to be formatted in the default format of one response per line with the label separated by a tab. ESCRITO automatically selects the default preprocessing pipeline for Italian and the teacher can easily inspect the resulting classifications. If he decides that it would be better to ignore spelling errors in the scoring, he can configure that using a high-level configuration API.

4. Summary

We presented ESCRITO, a toolkit for scoring of free-text answers in the educational domain. We support two user groups for educational NLP applications, teachers and NLP researchers, through both a high-level plug-and-play version and an easily extendable low-level API. We do so by providing baseline methods and setups for a number of common datasets, which are easily extendable through new datasets, preprocessing methods, or features.

ESCRITO enables *reproducible* research by carefully logging the experimental configuration and allowing to publish complete experimental setups including all preprocessing steps. ESCRITO directly addresses *multi-linguality* by providing preprocessing and feature extraction for a wide range of languages as well as a language-independent core of scoring functionality. We hope that ESCRITO will foster research on applying cutting-edge NLP technologies in educational applications as well as practical experimentation using free-text scoring in real-life scenarios.

5. Acknowledgments

This work has been funded by the German Federal Ministry of Education and Research under grant no. FKZ 01PL16075.

6. Bibliographical References

Attali, Y. and Burstein, J. (2004). Automated essay scoring with e-rater® v. 2.0. *ETS Research Report Series*, 2004(2).

Bär, D., Zesch, T., and Gurevych, I. (2013). DKPro Similarity: An Open Source Framework for Text Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 121–126, Sofia, Bulgaria, August. Association for Computational Linguistics.

Basu, S., Jacobs, C., and Vanderwende, L. (2013). Powergrading: a Clustering Approach to Amplify Human Effort for Short Answer Grading. *Transactions of the Association for Computational Linguistics (TACL)*, 1:391–402.

Chollet, F. et al. (2015). Keras. `https://github.com/fchollet/keras`.

Daxenberger, J., Ferschke, O., Gurevych, I., and Zesch, T. (2014). DKPro TC: A Java-based Framework for Supervised Learning Experiments on Textual Data. In *Proceedings of ACL: System Demonstrations*, pages 61–66, Baltimore, Maryland. Association for Computational Linguistics.

Dzikovska, M. O., Nielsen, R., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., and Dang., H. T. (2013). SemEval-2013 Task 7: The Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge. **SEM 2013: The First Joint Conference on Lexical and Computational Semantics*.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: An Update. 11:10–18.

Horbach, A. and Palmer, A. (2016). Investigating active learning for short-answer scoring. In *Proceedings of BEA 2016*, pages 629–637.

Horbach, A., Ding, Y., and Zesch, T. (2017a). The influence of spelling errors on content scoring performance. In *Proceedings of the forth Workshop on NLP Techniques for Educational Applications (NLPTEA 2017)*.

Horbach, A., Ding, Y., and Zesch, T. (2017b). The Influence of Spelling Errors on Content Scoring Performance. In *Proceedings of 4th Workshop on NLP Techniques for Educational Applications (NLPTEA 2017) at IJCNLP*.

Horbach, A., Scholten-Akoun, D., Ding, Y., and Zesch, T. (2017c). Fine-grained essay scoring of a complex writing task for native speakers. In *Proceedings of the Building Educational Applications Workshop at EMNLP*, page to appear, Copenhagen, Denmark.

Horsmann, T. and Zesch, T. (2018). DeepTC – An Extension of DKPro Text Classification for Fostering Reproducibility of Deep Learning Experiments. In *Under review at LREC 2018*.

Meurers, D., Ziai, R., Ott, N., and Kopp, J. (2011). Evaluating Answers to Reading Comprehension Questions in Context: Results for German and the Role of Information Structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9, Edinburgh, Scottland, UK. Association for Computational Linguistics.

Mohler, M. and Mihalcea, R. (2009). Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 567–575, Stroudsburg, PA, USA. Association for Computational Linguistics.

Neubig, G., Dyer, C., Goldberg, Y., Matthews, A., Ammar, W., Anastasopoulos, A., Ballesteros, M., Chiang, D., Clothiaux, D., Cohn, T., Duh, K., Faruqui, M., Gan, C., Garrette, D., Ji, Y., Kong, L., Kuncoro, A., Kumar, G., Malaviya, C., Michel, P., Oda, Y., Richardson, M., Saphra, N., Swayamdipta, S., and Yin, P. (2017). DyNet: The Dynamic Neural Network Toolkit - Technical Report.

Östling, R., Smolentzov, A., Tyrefors Hinnerich, B., and Höglin, E. (2013). Automated essay scoring for swedish. In *The 8th Workshop on Innovative Use of NLP for Building Educational Applications, Atlanta, GA, USA, June 13, 2013*, pages 42–47. Association for Computational Linguistics.

Pado, U. and Kiefer, C. (2015). Short Answer Grading: When Sorting Helps and When it Doesnt. In Linköpings universitet Linköping University Electronic Press, editor, *Proceedings of the 4th workshop on NLP for Computer Assisted Language Learning, NODALIDA 2015*, Linköping Electronic Conference Proceedings, pages 42–50, Wilna, Mai. LiU Electronic Press and ACL Anthology.

Pearson, K. (1895). Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242.

Riordan, B., Horbach, A., Cahill, A., Zesch, T., and Lee, C. M. (2017). Investigating neural architectures for short answer scoring. In *Proceedings of the Building Educational Applications Workshop at EMNLP*, page to appear, Copenhagen, Denmark.

Spearman, C. (1904). The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101.

Taghipour, K. and Ng, H. T. (2016). A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1882–1891.

Zesch, T., Heilman, M., and Cahill, A. (2015a). Reducing annotation efforts in supervised short answer scoring. In *Proceedings of the Building Educational Applications Workshop at NAACL*.

Zesch, T., Wojatzki, M., and Scholten-Akoun, D. (2015b). Task-independent features for automated essay grading. In *Proceedings of the Building Educational Applications Workshop at NAACL*.

A Leveled Reading Corpus of Modern Standard Arabic

Muhamed Al Khalil,[*] Hind Saddiki,[*†] Nizar Habash,[*] Latifa Alfalasi[‡]
[*]New York University Abu Dhabi, UAE
[†]Mohammed V University in Rabat, Morocco
[‡]Ministry of Education, UAE
{muhamed.alkhalil, hind.saddiki, nizar.habash}@nyu.edu, latifa.alfalasi@moe.gov.ae

Abstract

We present a reading corpus in Modern Standard Arabic to enrich the sparse collection of resources that can be leveraged for educational applications. The corpus consists of textbook material from the curriculum of the United Arab Emirates, spanning all 12 grades (1.4 million tokens) and a collection of 129 unabridged works of fiction (5.6 million tokens) all annotated with reading levels from *Grade 1* to *Post-secondary*. We examine reading progression in terms of lexical coverage, and compare the two sub-corpora (curricular, fiction) to others from clearly established genres (news, legal/diplomatic) to measure representation of their respective genres.

Keywords: Arabic, Corpus, Leveled Reading, Curriculum, Fiction

1. Introduction

Corpora are built for a wide range of purposes such as modeling language use for linguistics research, instructional material for educators, or training data for natural language processing (NLP) applications. Continued efforts in creating such resources are instrumental in furthering research for all application domains of NLP, namely, parsing and part-of-speech (POS) tagging, speech recognition, machine translation, document classification, etc.

Work in NLP for Modern Standard Arabic (MSA) is gaining momentum as more resources and tools are developed (Habash, 2010). Corpus data for MSA has been mostly sourced from the news genre (Zaghouani, 2014), while there are far fewer specialized resources, such as corpora for educational applications (Zaghouani et al., 2014; Alfaifi et al., 2013). As a particular type of educational resource, a level-annotated reading corpus can be leveraged for a multitude of applications: text simplification, automatic readability assessment, computer-assisted language learning, data-driven pedagogy, text genre and register profiling, and so on. Building a corpus of this nature contributes to the variety of resources at our disposal, allowing for research in Arabic NLP to progress in new directions.

In this paper, we present a reading corpus in MSA collected from textbooks of the United Arab Emirates (UAE) curriculum and a collection of 129 unabridged works of fiction. The curriculum texts are labeled with levels from grade 1 to 12 and the fiction texts are at a *Post-secondary* level, i.e., adult-level reading that is accessible to someone after achieving 12th grade reading proficiency. This corpus was created in the context of a project on the Simplification of Arabic Masterpieces for Extensive Reading (SAMER) intended to simplify works of Arabic fiction to a level that is more accessible for school-aged readers (Al Khalil et al., 2017).

The paper is organized as follows. Section 2 presents related work in corpus creation; Section 3 describes the corpus collection and annotation; We analyze the data in Section 4 before stating our conclusions and future work.

2. Background and Related Work

The multi-faceted complexity of MSA makes it a challenging language to tackle in NLP. There is the issue of morphological complexity due to its wide inflectional range and rich composition of clitics (Habash, 2010). Then, there is the challenge of resolving ambiguity due to its writing system with optional diacritics. While it is common to see fully diacritized texts for children, older readers are expected to resolve ambiguity from experience and context in readings where diacritics are often partial or omitted.

Corpora in Arabic have predominantly been collected from news data to serve as general purpose text for NLP applications (Habash, 2010; Zaghouani, 2014). In recent years, the various dialects of Arabic began receiving more attention (Shoufan and Al-Ameri, 2015; Khalifa et al., 2016; Jarrar et al., 2016). Specialized corpora have also been released for various NLP applications such as machine translation (Ziemski et al., 2016), plagiarism detection (Bensalem et al., 2013), sentiment analysis (Abdul-Mageed and Diab, 2012), and error correction (Alfaifi et al., 2013; Zaghouani et al., 2014) to name a few. High-resource languages, on the other hand, have enjoyed a wider variety of specialized corpora, including data for pedagogical and educational applications (Pravec, 2002; Braun et al., 2006; Laufer and Ravenhorst-Kalovski, 2010). Also, recently reignited interest in text readability assessment as a computational task has encouraged more work in the creation of curricular and pedagogical corpora (Collins-Thompson, 2014; François, 2014; Volodina et al., 2014; Zalmout et al., 2016).

Budding research in computational readability for MSA has led to the creation of leveled corpora from curriculum texts. For instance, a corpus of 150 texts from the Saudi Arabian (KSA) curriculum labeled with [easy, intermediate, difficult] (Al-Khalifa and Al-Ajlan, 2010), and a corpus of 1196 texts totaling 400K words from the Jordanian curriculum (Al Tamimi et al., 2014). To the best of our knowledge, a corpus at the scale of the curricular data collected in our work (1.4M tokens) has yet to be released.

Grade 2	الطّالِبَةُ النَّجيبَةُ مَيْثاءُ طالِبَةٌ نَشيطَةٌ، تُطيعُ والِدَيْها، وَتُحافِظُ عَلى صَلاتِها، تَصْحو مِنْ نَوْمِها مُبَكِّرَةً، تَتَناوَلُ فَطورَها، وَتُنَظِّفُ أَسْنانَها، وَتَرْتَدي مَلابِسَ الْمَدْرَسَةِ، تَلْتَقي مَعَ زَميلاتِها مُبْتَسِمَةً، تَجْلِسُ في صَفِّها بِهُدوءٍ وَانْتِباهٍ. *Maitha is a clever hard-working student. She listens to her parents, and keeps her prayers. She wakes up early, eats her breakfast, brushes her teeth, and puts on her school uniform. She greets her classmates with a smile, and sits quietly and attentively in her class.*
Grade 7	شاعَ شِعْرُ الحِكْمَةِ في الأَدَبِ العَرَبِيِّ، وَهُوَ يُوَضِّحُ القِيَمَ وَالمَبادِئَ وَالأَخْلاقَ وَالأَوامِرَ الإِلَهِيَّةَ، وَيُفْصِحُ عَنْ تَجارِبَ وَخِبْراتٍ سابِقَةٍ تُنْقَلُ عَبْرَ الأَجْيالِ، وَتَحْكي قِصَصًا نَتَعَلَّمُ مِنْها المَواعِظَ وَالْعِبَرَ، وَيَرِدُ هذا الشِّعْرُ في بَيتٍ أَوْ أَبْياتٍ أَوْ في قَصائِدَ كامِلَةٍ. *Poetry of wisdom became prevalent in Arabic literature. It is a kind of poetry that clarifies divine commandments, morals, principals, and values. It also discloses and transmits past experiences across generations, telling stories from which we learn lessons and wisdom. This poetry can come in the form of one line, a few lines, or a whole poem.*
Grade 10	وقد حدث يومًا وأنا مدرس في المدرسة الخديوية أن دخلت غرفة الصف فألفيت على مكتبي كل أدوات الرياضة مرصوصة على نحو لا شك أنه متعمد، وكان تلاميذي لا يجهلون كرهي للرياضة، وكنت أنا لا أكتمهم أني أعد نفسي جاهلاً بها، وكان غرضهم من رصّ هذه الأدوات أن يعابثوني عسى أن أثير الضجة التي يشتهونها ولا يفوزون مني بها، ولكني لم أفعل بل اكتفيت بأن دعوت الفراش فحمل هذه الأدوات ووضعها في مكانها ثم بدأت الدرس. *One day when I was teaching at the Khedive School I entered the classroom and found all the mathematics tools lined up purposefully in a pattern. My students were not ignorant of my hate of mathematics, and I never concealed to them that I considered myself ignorant in the field. Their goal was to jest with me so that I make the big fuss they desire but never attain. And I did not; I only called the janitor who carried the tools and put them back in their place; then I started the lesson.*
Novel	ثم أنا أمضي إلى هذه النافذة، فلا أكاد أفتحها حتى تمتلئ نفسي روعةً وجلالًا لهذه الأشجار النائمة، وهذه الأزهار المتأرجة، وهذه الأطيار التي تحلم في ثنايا الغصون، وكل هذا لي ملك خالص لا يشاركني فيه أحد، ولا يزاحمني عليه أحد، أستطيع أن أعبث به إن شئت، ومتى شئت، وكيف شئت، لا يسألني أحد عما أفعل! *Then I went to this window, and no sooner had I opened it than my soul filled up with majestic awe of these slumbering trees, these fragrant flowers, and these birds dreaming in the nooks of branches. This is all mine, I share it with no one, and no one crowds me for it. I can toy with it if I wish, whenever I wish, however I wish, and I answer to no one about it.*

Figure 1: Samples of reading text from different levels of the corpus

3. Corpus Description

In this section, we discuss the variety observed in the corpus with illustrative examples. We then document the data collection and processing efforts, and present descriptive statistics and details of the text annotations.

3.1. Text Varieties in the Corpus

This corpus consists of two sub-corpora: a diverse body of texts combining the full UAE curriculum, and a body of fiction texts derived from the Hindawi collection. A curricular sub-corpus, especially one covering different subjects, includes almost all kinds of texts: expository, transactional, procedural, argumentative, informative, narrative, literary, scientific, etc. A fiction-based corpus provides a special register of the language, and has been used to study both general linguistic features and more specific stylistic features (Biber, 2011). The key difference between the two bodies of texts is that while the curricular sub-corpus is focused on information delivery and educational growth assessment, the second is occupied with the literary aesthetic and is thus pleasantly blasé about teaching and learning. Between the two, however, one can capture the full spectrum of written language phenomena that a school-educated Arabic-speaker would experience, allowing the corpus to qualify as a general corpus (McEnery et al., 2006).

Illustrative Examples To give samples of the texts included in each level, we chose four short pieces that best reflect the nature and variety of those texts. For the first three pieces, each piece comes from a grade that tends to be midrange in the grades of that level; with the fourth piece coming from, perhaps, the best well-known novel in that literary collection. The first textual piece comes from the 2nd grade and it describes a person and her daily habits. It is fully diacritized. The text is – as is expected in this introductory level – direct, concrete, and less complex. It is generally one-dimensional comprised mainly of short declarative sentences. The second piece comes from the 7th grade and it describes a genre of poetry in Arabic. It is also fully diacritized. It is expository, conceptual, and meta-lingual (using language about language). It is more complex in terms of both vocabulary and sentence structure and length. The third piece comes from the 10th grade and it is excerpted from a memoir. It is not diacritized. It is story-like told in the first person. Its style is narrative made of several complex sentences and expressions. The fourth piece comes from a well-known novel in the Hindawi collection, *The Call of the Curlew* by Taha Hussein.[1] It is not diacritized. It is an introspective musing by the omnipresent narrator. It is made of run-on complex sentences with more abstract vocabulary. It has a clear literary style, typically found in fiction: mixing the concrete with the poetic to produce a pleasant emotive sense.

3.2. Data Gathering and Extraction

Curriculum The curriculum textbooks were obtained as `InDesign`[2] files spanning 12 grades (Elementary Grade 1 to Secondary Grade 12) and three subjects (Arabic lan-

[1] Accessible at `http://www.hindawi.org/books/13052715/`

[2] Adobe InDesign desktop publishing software `http://www.adobe.com/products/indesign.html`

guage, social studies, Islamic studies). We converted each InDesign file into an intermediary HTML format then into raw UTF-8 text format. The curriculum files were obtained from the UAE Ministry of Education.[3]

Fiction We collected 129 works of fiction available in the public domain from the online catalog of the Hindawi Foundation.[4] We downloaded the individual e-book files in `.epub`[5] format and converted them to an intermediary HTML format then into raw UTF-8 text format.

3.3. Building the Corpus

For the curricular sub-corpus, all data pertaining to a given grade is labeled with its corresponding grade level going from primary grade level 1 to secondary (high school) grade level 12. Additional annotation for subject (Arabic Language, Social Studies, Islamic Studies), term (1st, 2nd, sometimes 3rd) and unit number (each unit is marked in the textbook's table of contents as a set of lessons under a theme with specific learning objectives).

Books in the fiction sub-corpus are all labeled at the *Post-secondary* level indicating they are accessible to readers having achieved reading proficiency of the full 12-grade curriculum. Each book has a unique ID tied to its meta-information (author and title) as well as manually annotated year of copyright and publication.

We annotated each token in the corpus with morphological information including lemma, POS using the MADAMIRA tool for morphological disambiguation (Pasha et al., 2014). We expect a drop in accuracy on this genre of text given that MADAMIRA has been trained on news data. An in-house evaluation on an example of literary fiction text[6] shows a drop of 4% absolute in word analysis performance for choice of lemma and POS. While lower than on news text, the performance is still at a high 92%.

Table 1 presents summary statistics on all the collected text, differentiating the curricular and fiction sub-corpora. The *Sentences* represent complete lines of text. Words counts in the text are reported by whitespace-based *tokens* (including punctuation and numbers as separate words). To get a sense of lexical richness, we also compute unique tokens, i.e., *types*, and unique *lemmas* for the word forms occurring in the text.

The learner's vocabulary after completing *Grade 12* education reaches 22K distinct lemmas (closer to 18K when proper nouns, punctuation and digits are excluded). When compared to English, Nation (2013) estimates a learner to require a vocabulary of 15K to 20K words in order to optimally read and comprehend text with no obstruction from unknown vocabulary. However, we bear in mind that vocabulary is not the only indicator of level. One must take into account how common or specialized the vocabulary is, semantic fields, discourse, style, and so on to fully assess reading level beyond word frequency.

Grade Level	Sentences	Tokens	Types	Lemmas
1	10,860	57,409	9,193	4,391
2	8,580	65,014	10,142	4,390
3	10,966	87,460	13,692	5,531
4	11,597	108,946	18,291	7,059
5	8,833	86,096	15,727	6,453
6	9,710	108,557	19,862	7,937
7	12,112	116,176	21,489	8,466
8	11,619	118,288	21,092	8,175
9	13,176	172,175	25,547	9,850
10	11,518	171,340	27,003	10,196
11	12,253	157,453	27,827	10,364
12	10,812	165,791	31,323	11,732
Curriculum (All)	132,036	1,414,705	89,446	22,143
Fiction (avg. per book)	1,279	43,367	10,584	4,719
Fiction (All)	165,005	5,594,310	261,920	44,498

Table 1: Summary statistics for the leveled reading corpus

4. Quantitative Corpus Analysis

We describe a preliminary exploration of the corpus by conducting two studies: lexical coverage progression over the curriculum as a measure of the grade-leveling scheme's validity, and a similarity comparison with other well-known corpora in the news genre (Gigaword (Parker et al., 2011)) and the legal/diplomatic genre (UN Corpus (Ziemski et al., 2016)) to establish curricular and fiction texts as distinct genres.

All studies in Section 4 are performed on *content* tokens only. In other words, we exclude punctuation and digits (non-content tokens) from our calculations, which make up 18% and 15% of all tokens in the curricular and fiction sub-corpora, respectively. We also discount any content words not in the MADAMIRA vocabulary database, i.e., out-of-vocabulary tokens, which amount to 0.96% of all content tokens in the curricular sub-corpus and 2.2% of all content tokens in the fiction sub-corpus.

Level	Lexical Coverage
1	n/a
2	93.6%
3	95.3%
4	96.1%
5	97.2%
6	97.3%
7	97.6%
8	98.6%
9	98.1%
10	98.5%
11	98.5%
12	99.4%
Post-secondary	97.1%

Table 2: Lexical coverage in levels 1 to 12; Average lexical coverage per book in the post-secondary level

[3]The corpus obtained from the UAE Ministry of Education pertained to the curriculum applied between 2014 and 2016. The current curriculum was designed with a richer selection of literary and informational readings. We look forward to analyzing the current curriculum as part of ongoing collaboration with the UAE Ministry of Education.

[4]On 06/29/2017 from `http://www.hindawi.org/`

[5]`http://idpf.org/epub`

[6]Chapter 1 of Ibrahim Alkatib, by Ibrahim Al-Mazini (1889-1949).

4.1. Lexical Coverage

We examine whether the grade-leveling scheme is a valid indication of reading level by measuring lexical coverage. Lexical coverage is defined as follows: a word list is said to provide lexical coverage of 80% of a given text if 80% of all word tokens in said text occur in that word list. When reading a text, the amount of vocabulary familiar to the reader influences comprehension, which raises the question of lexical threshold, i.e., the minimum rate of lexical coverage for reading comprehension. Studies on lexical thresholds for reading set a lexical coverage of 95% as the minimum threshold for *adequate comprehension*[7] and lexical coverage of 98% as the threshold for optimal (unassisted) comprehension. See (Nation, 2006; Laufer and Ravenhorst-Kalovski, 2010) for further details.

Steps for the curricular sub-corpus lexical coverage:

- Selecting a target **Grade$_i$**
- Computing familiar vocabulary from all previous grades [1,i-1] as a list of unique lemmas
- Calculating the total count of tokens in **Grade$_i$** corresponding to lemmas that exist in the list of familiar vocabulary
- Reporting the lexical coverage as the ratio of tokens matching the list over total token count for the target **Grade$_i$**

Steps for the fiction sub-corpus lexical coverage:

- Selecting a target **Book$_i$**
- Computing familiar vocabulary from all curricular grades [1,12] as a list of unique lemmas
- Calculating the total count of tokens in **Book$_i$** corresponding to lemmas that exist in the list of familiar vocabulary
- Computing the lexical coverage as the ratio of tokens matching the list over total token count for the target **Book$_i$**
- Reporting the lexical coverage as the average of all lexical coverage ratios computed for the 129 books in the fiction sub-corpus[8]

Table 2 presents the results of the study carried out according to the steps described for both sub-corpora. We point out that no lexical coverage is reported for Grade 1. Although vocabulary acquisition does occur prior to Grade 1, our curricular sub-corpus lacks data for the Kindergarten level. We rely on the 95% minimum and 98% optimal thresholds for English as a ballpark estimate, being fully aware that these threshold numbers may vary for MSA and our target readership. We observe a clear progression across the curricular levels and a lexical coverage ratio indicating that the 95% minimum threshold is consistently met while the optimal threshold of 98% is reached starting the 8th Grade, at which time learners are expected to have acquired a much richer vocabulary. The post-secondary lexical coverage of 97.1% suggests that vocabulary acquired from readings in a 12-grade curriculum allows for adequate reading and understanding of a work of fiction.

[7]Usually measured by testing and scoring readers with comprehension questions (Nation, 2006).

[8]Averaging per book is more representative of the lexical coverage required for reading any work of fiction at a post-secondary level.

Gigaword	65.5%		
Curriculum	76.7%	71.0%	
UN	57.3%	68.5%	64.4%
	Fiction	**Gigaword**	**Curriculam**

Table 3: Dice Similarity (1) between corpora of different genres

4.2. Genre Similarity and Difference

A similarity comparison of our corpus with other established corpora in the news genre (Gigaword (Parker et al., 2011)) and the legal/diplomatic genre (UN Corpus (Ziemski et al., 2016)) can approximate difference in genre, which could potentially establish this corpus as representative of the curricular genre.
We use the Dice Coefficient (1) to compute similarity between pairs of corpora. Given that the curricular sub-corpus is the smallest in size with 1.4M tokens, for comparison we use randomly sampled subsets of nearly 1.4M tokens for each of Gigaword, UN and the Fiction sup-corpus. The similarity is calculated on unique lemma sets *A* and *B* for each comparison pair.

$$Dice = \frac{2 \cdot |A \cap B|}{|A| + |B|} \quad (1)$$

We report the results of pairwise Dice similarity comparisons for the four corpora in Table 3. The UN corpus using specialized legal/diplomatic language behaves as expected, being the least similar to other genres. It presents with the lowest similarity score of 57.3% in the UN-Fiction comparison, given that legal or administrative language is quite different from literary writing. We note with interest the Gigaword-Fiction 65.5% similarity. This comparison of two corpora from clearly distinct genres (news and literary texts) gives us a better sense of what 65% similarity or rather 35% difference means between two clearly established genres. The 23%, 29% and 36% respective difference in Curriculum (-Fiction, -Gigaword, -UN) comparisons could indicate sufficient distance between the curricular corpus and the others for it to be representative of its own curricular/educational genre.

5. Conclusion and Future Work

We presented a corpus for reading in MSA that was collected from curricular texts (1.4M tokens) and works of fiction (5.6M tokens). The corpus was annotated with reading levels per grade for the curricular sub-corpus and a post-secondary level for the collection of novels in the fiction sub-corpus. We assessed the validity of a grade-leveling scheme using progression of lexical coverage over the curriculum. A similarity comparison with other established corpora in the news genre, and the legal/diplomatic genre

could potentially establish this corpus as representative of the curricular or educational genre.

In the future, we plan to use this corpus in modeling levels of reading proficiency to simplify works of fiction in the context of the SAMER project. We also plan on annotating portions of the corpus with morphological and syntactic information. It is our intent to work on releasing this data in full-text format and/or as an n-gram frequency dataset to be exploited in training NLP tools for any number of educational applications.

6. Acknowledgements

The work on this project is funded by a New York University Abu Dhabi Research Enhancement Fund grant. We would like to express our thanks to the UAE Ministry of Education for providing us with the curriculum materials, which are essential for this research project. We also thank Bassel Musfi for annotating the fiction sub-corpus with copyright and publication year information.

7. References

Abdul-Mageed, M. and Diab, M. T. (2012). Awatif: A multi-genre corpus for modern standard Arabic subjectivity and sentiment analysis. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, pages 3907–3914.

Al-Khalifa, H. S. and Al-Ajlan, A. A. (2010). Automatic readability measurements of the Arabic text: An exploratory study. *Arabian Journal for Science and Engineering*, 35(2 C):103–124.

Al Khalil, M., Habash, N., and Saddiki, H. (2017). Simplification of Arabic masterpieces for extensive reading: A project overview. *Procedia Computer Science*, 117:192–198.

Al Tamimi, A. K., Jaradat, M., Al-Jarrah, N., and Ghanem, S. (2014). Aari: automatic Arabic readability index. *Int. Arab J. Inf. Technol.*, 11(4):370–378.

Alfaifi, A., Atwell, E., and Abuhakema, G. (2013). Error annotation of the Arabic learner corpus. In *Language Processing and Knowledge in the Web*, pages 14–22. Springer.

Bensalem, I., Rosso, P., and Chikhi, S. (2013). A new corpus for the evaluation of Arabic intrinsic plagiarism detection. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 53–58. Springer.

Biber, D. (2011). Corpus linguistics and the study of literature: Back to the future? *Scientific Study of Literature*, 1(1):15–23.

Braun, S., Kohn, K., et al. (2006). Corpus technology and language pedagogy: New resources, new tools, new methods. English corpus linguistics vol 3.

Collins-Thompson, K. (2014). Computational assessment of text readability: A survey of current and future research. *ITL-International Journal of Applied Linguistics*, 165(2):97–135.

François, T. (2014). An analysis of a French as a foreign language corpus for readability assessment. In *Proceedings of the third workshop on NLP for computer-assisted language learning at SLTC 2014, Uppsala University*, number 107. Linköping University Electronic Press.

Habash, N. Y. (2010). *Introduction to Arabic natural language processing*, volume 3. Morgan & Claypool Publishers.

Jarrar, M., Habash, N., Alrimawi, F., Akra, D., and Zalmout, N. (2016). Curras: an annotated corpus for the Palestinian Arabic dialect. *Language Resources and Evaluation*, pages 1–31.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016). A Large Scale Corpus of Gulf Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Laufer, B. and Ravenhorst-Kalovski, G. C. (2010). Lexical threshold revisited: Lexical text coverage, learners' vocabulary size and reading comprehension. *Reading in a foreign language*, 22(1):15.

McEnery, T., Xiao, R., and Tono, Y. (2006). *Corpus-based language studies: An advanced resource book*. Taylor & Francis.

Nation, I. (2006). How large a vocabulary is needed for reading and listening? *Canadian Modern Language Review*, 63(1):59–82.

Nation, I. S. (2013). *Learning Vocabulary in Another Language Google eBook*. Cambridge University Press.

Parker, R., Graff, D., Chen, K., Kong, J., and Maeda, K. (2011). Arabic Gigaword Fifth Edition. LDC catalog number No. LDC2011T11, ISBN 1-58563-595-2.

Pasha, A., Al-Badrashiny, M., Diab, M., El Kholy, A., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. M. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *LREC 2014, Reykjavik, Iceland*.

Pravec, N. A. (2002). Survey of learner corpora. *ICAME journal*, 26(1):8–14.

Shoufan, A. and Al-Ameri, S. (2015). Natural language processing for dialectical Arabic: A survey. In *ANLP Workshop 2015*, page 36.

Volodina, E., Pilán, I., Borin, L., and Tiedemann, T. L. (2014). A flexible language learning platform based on language resources and web services. In *LREC*, pages 3973–3978.

Zaghouani, W., Mohit, B., Habash, N., Obeid, O., Tomeh, N., Rozovskaya, A., Farra, N., Alkuhlani, S., and Oflazer, K. (2014). Large scale Arabic error annotation: Guidelines and framework. In *LREC 2014, Reykjavik, Iceland*.

Zaghouani, W. (2014). Critical survey of the freely available Arabic corpora. In *Proceedings of the Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools, LREC*, pages 1–8.

Zalmout, N., Saddiki, H., and Habash, N. (2016). Analysis of foreign language teaching methods: An automatic readability approach. *NLPTEA 2016*, page 122.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The united nations parallel corpus v1. 0. In *LREC*.

Developing New Linguistic Resources and Tools for the Galician Language

Rodrigo Agerri [*], Xavier Gómez Guinovart [**], German Rigau [*], Miguel Anxo Solla Portela [**]
[*] IXA NLP Group, University of the Basque Country UPV/EHU | [**] TALG Research Group, University of Vigo
rodrigo.agerri@ehu.eus, xgg@uvigo.es, german.rigau@ehu.eus, miguelsolla@uvigo.es

Abstract

In this paper we describe the work towards developing new resources and Natural Language Processing (NLP) tools for the Galician language. First, a new corpus, manually revised, for POS tagging and lemmatization is described. Second, we present a new manually annotated corpus for Named Entity tagging for Galician. Third, we train and develop new NLP tools for Galician, including the first publicly available Galician statistical modules for lemmatization and Named Entity Recognition, and new modules for POS tagging, Wikification and Named Entity Disambiguation. Finally, we also present two new Web demo applications to easily test the new set of tools online.

Keywords: Galician, Less-resourced languages, Language Resources, Linguistic Tools

1. Introduction

We present new developments on linguistic resources and Natural Language Processing tools for the Galician language. There are previous works addressing the creation of resources to allow the automatic processing of the Galician language, including the Galician WordNet (Gómez Guinovart and Solla Portela, 2017), the Galician SemCor (Solla Portela and Gómez Guinovart, 2017), works on terminology (Solla Portela and Gómez Guinovart, 2015) or annotation of large corpora (Gómez Guinovart and López Fernández, 2009). Furthermore, there are also publicly available Natural Language Processing tools for the Galician language, notably Freeling (Padró and Stanilovsky, 2012) and Linguakit (Gamallo and Garcia, 2017). However, it is remarkable the lack of linguistic processors that are available for other less-resourced languages, such as statistical tools for lemmatization or Named Entity Recognition.

In this work we aim to contribute to the development of NLP resources for Galician by providing a new manually revised corpus for POS tagging and lemmatization, and a new manually annotated corpus for Named Entity Recognition. This would allow us to train previous unavailable statistical tools for the processing of Galician texts. As a result of our effort, seven new linguistic processors for Galician are presented: a rule-based tokenizer and statistical tools for POS tagging, lemmatization, Named Entity Recognition, Named Entity Disambiguation, Wikification and graph-based Word Sense Disambiguation. In particular, this paper presents the first open source statistical lemmatizer and Named Entity tagger for the Galician language.

2. Corpora and Lexical Resources

In this paper the main work on corpora and lexical resources was undertaken in order to create new resources to train a statistical POS tagger and lemmatizer, and a new Named Entity Recognition and Classification tagger. The basis for the corpora required is the CTG Galician Technical Corpus[1] (TALG Research Group, 2016). The CTG corpus contains around 18 million words from various domains:

- GALEX Galician legal texts (∼3M words).
- XIGA Galician texts on computing and telecommunications (∼3M words).
- AUGA Galician texts on ecology and environmental sciences (∼3M words).
- ACHEGA Galician economy texts (∼2M words).
- SOGAL Galician texts on sociology and social sciences (∼3M words).
- MEDIGAL Galician texts on medicine (∼4M words).

In order to provide the required resources to train a POS tagger and lemmatizer for Galician, we manually revised a subset of the CTG corpus containing 2,852,472 tokens, 105,986 sentences and 938 texts (from scientific-technical communications, academic works and news from ecology and environmental sciences, and legal texts). Apart from the annotated corpora, two dictionaries to perform dictionary-based lemmatization and multiword recognition were built from several sources:

- Dicionario da Real Academia Galega[2].
- Vocabulario ortográfico da lingua galega (VOLGa)[3].
- Hunspell Spellchecker for Galician[4].
- Galician dictionary distributed by Apertium[5].
- Galician dictionary distributed by Freeling[6].

[1] http://sli.uvigo.gal/CTG/

[2] http://academia.gal/dicionario/

[3] http://www.realacademiagalega.org/recursos-volg

[4] https://github.com/meixome/hunspell-gl

[5] http://sourceforge.net/projects/apertium/

[6] http://nlp.lsi.upc.edu/freeling/

- Other textual and lexical resources developed by the TALG research group.

The collection of part-of-speech tags used in the CTG subset corpus and dictionaries are based on the CTAG tagset developed by the TALG Group (Gómez Guinovart and López Fernández, 2009). The subset used for training the POS tagger contains around 200 morphological tags and 23K different lemmas. For training and testing, we created three splits following the Penn Treebank as model. Thus, we took the first 950K tokens for training, the following 150K for development and the next 150K for test.
Finally, a new corpus for Named Entity Recognition in Galician (TALG Research Group, 2018) was manually annotated on a subsection of the CTG corpus consisting of 202,334 tokens in 8,137 sentences (from the news and ecology and environmental sciences domains). The CoNLL guidelines for annotation were followed (Tjong Kim Sang and De Meulder, 2003). This resulted in an inventory of 4 named entity classes distributed as follows: 1,293 persons (PER), 3,183 organizations (ORG), 2,616 locations (LOC) and 1,375 miscellaneous entities (MISC). From this corpus 162K tokens are used for training and 41K for test.

3. NLP Tools

In order to develop new linguistic processors using the resources described in the previous section, we decided to try the IXA pipes tools[7] (Agerri et al., 2014). The aim of IXA pipes is to provide a modular set of ready to use Natural Language Processing (NLP) tools. Apart from being easy to train and deploy, they are also a good fit for our current aim of providing new tools for Galician because every module but the tokenizer is machine learning based. In fact, IXA pipes tries to use the same approach across NLP tasks in order to create robust processors both across domains and languages. This strategy has proven to be very successful for Named Entity Recognition and Classification (NER) (Agerri and Rigau, 2016) and Opinion Target Extraction (OTE) (San Vicente et al., 2015) benchmarks, both in out-of-domain and in-domain evaluations.

3.1. Semi-supervised approach

IXA pipes learns supervised models based on the Perceptron algorithm (Collins, 2002). To avoid duplication of efforts, IXA pipes uses the Apache OpenNLP project implementation[8] customized with its own features. By design, IXA pipes tools aim to establish a simple and shallow feature set, avoiding any linguistic motivated features, with the objective of removing any reliance on costly extra gold annotations apart from the target task (POS, lemmas, NER) and/or cascading errors if automatic language processors are used. IXA pipes modules consist of: (i) Local, shallow features based mostly on orthographic, word shape and n-gram features plus their context; and (ii) three types of simple clustering features, based on unigram matching. Specifically, IXA pipes implements, on top of the local features, a combination of three word representation features: (i) Brown (Brown et al., 1992) clusters, taking the 4th, 8th, 12th and 20th node in the path; (ii) Clark (Clark, 2003) clusters and, (iii) Word2vec (Mikolov et al., 2013) clusters, based on K-means applied over the extracted word vectors using the skip-gram algorithm. The implementation of the clustering features looks for the cluster class of the incoming token in one or more of the clustering lexicons induced following the three methods listed above. If found, then we add the class as a feature. The Brown clusters only apply to the token related features, which are duplicated. The word representation features are *combined* and *stacked* from features induced over different data sources. For this work the new Galician POS tagger, lemmatizer and NER tagger are based on this design.
Furthermore, IXA pipes are extended with third-party tools for those type of annotations not developed within its toolchain. Most notably, it includes integration of wikification and Named Entity Disambiguation (NED) via DBPedia Spotlight (Mendes et al., 2011) as well as graph-based Word Sense Disambiguation (Agirre et al., 2014).
Summarizing, the new set of NLP tools for Galician implemented within IXA pipes consists of the following modules:

- ixa-pipe-tok: A rule-based tokenizer and sentence segmenter.
- ixa-pipe-pos: A statistical lemmatizer and POS tagger (ixa-pipe-pos); ixa-pipe-pos is complemented by the dictionaries described in the previous section for dictionary-based lemmatization and multiword detection. For efficiency, these dictionaries are deployed as finite state automata based on Morfologik[9].
- ixa-pipe-nerc: A state of the art NER tagger.
- ixa-pipe-wikify: Wikification tool based on DBpedia Spotlight.
- ixa-pipe-ned: A NED tool based on DBpedia Spotlight. The NED uses the entities spotted by ixa-pipe-nerc as input to perform the disambiguation.
- ukb-naf: UKB graph-based Word Sense Disambiguation.

3.2. Named Entity Disambiguation and Wikification

Galician language already had a previous version of the DBpedia Spotlight[10] that was implemented together with the official server of the Galician DBpedia (Solla Portela and Gómez Guinovart, 2016), but it used the Lucene version (Mendes et al., 2011). A new, better performing, generative model (Daiber et al., 2013) for DBpedia Spotlight has been created by configuring the Galician language in model-quickstarter[11] in order to handle Wikification and Named Entity Disambiguation with IXA pipes, using the modules already available for other languages.

[7] http://ixa2.si.ehu.es/ixa-pipes
[8] http://opennlp.apache.org/
[9] https://github.com/morfologik/
[10] http://sli.uvigo.gal/dbpedia/spotlight/
[11] https://github.com/dbpedia-spotlight/model-quickstarter

3.3. Web demo

The full new set NLP tools provided by the IXA pipes for the Galician language can be easily tested through the *Lingaliza* Web application[12] developed by the TALG Research Group of the University of Vigo. Furthermore, *DContado* [13], the user-oriented version of Lingaliza, is designed to be used in the field of Digital Humanities for Research in Humanities and Social Sciences under the auspices of the European infrastructure CLARIN[14] (Bel et al., 2016).

4. Experimental Results

In this section we will report on results for the POS tagger and Named Entity Recognition taggers trained on the CTG corpora as described in section 2. The aim of this section is to give us a first idea of the performance of these tools for Galician. In order to train our systems with the semi-supervised approach described in section 3.1., first the three types of word representations needed to be induced from large unlabelled data. The first obvious candidate was to use the Galician Wikipedia. However, as shown by previous experiments using this approach (Agerri and Rigau, 2016), it is convenient for best performance to provide word representation features induced from different data sources. In order to do that we compiled a *large corpus* from various domains by crawling data from the following Web data sources:

- Asociación para a Defensa Ecolóxica de Galiza (ADEGA) (http://adega.gal).
- The Galician Political Party, *Bloque Nacionalista Galego* (BNG) (http://www.bng.gal/).
- Galician newspaper *Praza Pública* (http://praza.gal/).
- Galician weekly and newspaper *Sermos Galiza* (http://www.sermosgaliza.gal/).
- Galician government official website: *Xunta de Galiza* (http://www.xunta.gal).

Roughly speaking, the text used from the Galician wikipedia to train the clusters contained 31M words, whereas the *large corpus* compiled contained around 20M words (see (Agerri and Rigau, 2016) for details on the clusters training process). For POS tagging we choose our best feature configurations on the development set whereas for NERC we did our development via 5-fold cross validation. The NER results reported in Table 1 confirm the behaviour of *ixa-pipe-nerc* previously observed in (Agerri and Rigau, 2016) for other languages. The local features are improved substantially by the clustering features. At the same time, the combination of those features provide the best results.

Table 2 reports on the results of POS tagging. As it is known from previous approaches using distributional semantic features for POS tagging, the gains obtained from using word representations for this task as not as large as those obtained for NER. At least if we look at word accuracy (WA) only. However, by looking at the sentence accuracy (SA) and unknown accuracy (UA) scores it can be seen the clear improvements in performance from the local features to the models using clustering features on top of the local ones. The UA scores in particular are quite interesting as it show that, despite the negligible improvements in terms of WA, the models containing clustering features are much more robust to tag unseen words.

Features	P	R	F1
Local	79.95	81.93	80.93
Local + BL2000	82.34	83.65	82.99
Local + CW300	82.32	83.21	82.76
Local + W2VW100	81.83	84.22	83.01
Local + BL2000 + CW300	83.85	84.54	84.19

Table 1: NER results. BL2000: Brown 2000 classes from Large corpus; CW300: Clark 300 classes from Wikipedia, and W2V100: Word2vec 100 classes from Wikipedia.

Features	SA	UA	WA
Local	72.08	80.09	98.31
Local + BW1000	70.83	81.14	98.24
Local + CW300	74.41	83.94	98.50
Local + W2VW100	71.93	81.45	98.30
Local + CW300 + CL400	75.22	85.35	98.54

Table 2: POS results. SA: Sentence Accuracy; UA: unknown accuracy; WA: word accuracy.

5. Related Work

Previous work on NLP for Galician have been centred around Freeling (Padró and Stanilovsky, 2012) and Linguakit (Gamallo and Garcia, 2017). Freeling provides wide support for several languages, however, it does not yet include machine learning based lemmatization or Named Entity Recognition and Classification for Galician. Current support includes tokenization, POS tagging, and rule-based lemmatization and detection (no classification) of named entities.

With respect to Linguakit, it provides applications for Galician such as a verb conjugator, a POS tagger, dependency parser, Named Entity tagger, sentiment analyzer, a keyword extractor and a summarization module. Lemmatization, Named Entity tagging and parsing are performed by language independent rule-based modules.

6. Conclusions and future work

In this paper we presented a new set of linguistic resources and NLP tools for the Galician language. In particular, we would like to highlight the contribution of two new manually revised and annotated corpora for Galician POS tagging, lemmatization and Named Entity Recognition (NER). Furthermore, this paper has presented a number of novel statistical NLP tools for the Galician language, including a lemmatizer, a NER tagger, and a wikification and NED module. Additionally, the reported results indicate that the

[12] http://sli.uvigo.gal/lingaliza/
[13] http://sli.uvigo.gal/dcontado/
[14] http://clarin.eu

performance obtained is similar to performance of those tasks for other languages which is promising start (Agerri and Rigau, 2016), although more experiments remain to be done.

With respect to future work, we believe that establishing a clear benchmarking between the different tools for Galician NLP would be very interesting. In this way, and apart from providing an objective comparison, it would be possible to learn what are the strengths and weaknesses of each toolchain currently available for Galician. Additionally, we would like to keep extending the coverage of NLP tools for Galician. For example, IXA pipes provides modules for statistical chunking and constituent parsing, which have been already deployed for Basque, English and Spanish.

The code for every tool presented in this paper is already available for public use through the IXA pipes website. In the same way, every trained model[15] and associated linguistic resources are freely available[16]. Furthermore, these tools and models can be tested using the two Web implementations developed by the TALG Research Group of the University of Vigo.

7. Acknowledgements

This research has been carried out thanks to the project TUNER (TIN2015-65308-C5-1-R) supported by the Ministry of Economy and Competitiveness of the Spanish Government and the European Fund for Regional Development (MINECO/FEDER).

8. Bibliographical References

Agerri, R. and Rigau, G. (2016). Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

Agerri, R., Bermudez, J., and Rigau, G. (2014). IXA pipeline: Efficient and ready to use multilingual NLP tools. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2014)*.

Agirre, E., López de Lacalle, O., and Soroa, A. (2014). Random walks for knowledge-based word sense disambiguation. *Comput. Linguist.*, 40(1):57–84, March.

Bel, N., González-Blanco, E., and Iruskieta, M. (2016). CLARIN centro-k-español. *Procesamiento del Lenguaje Natural*, 57:151–154.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Clark, A. (2003). Combining distributional and morphological information for part of speech induction. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 59–66. Association for Computational Linguistics.

Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8.

[15] http://ixa2.si.ehu.es/ixa-pipes/

[16] http://sli.uvigo.gal/download/SLI_Galician_Corpora/

Daiber, J., Jakob, M., Hokamp, C., and Mendes, P. N. (2013). Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

Gamallo, P. and Garcia, M. (2017). Linguakit: a multilingual tool for linguistic analysis and information extraction. *Linguamática*, 9(1):19–28.

Gómez Guinovart, X. and López Fernández, S. (2009). Anotación morfosintáctica do Corpus Técnico do Galego. *Linguamática*, 1(1):61–71.

Gómez Guinovart, X. and Solla Portela, M. A. (2017). Building the galician wordnet: methods and applications. *Language Resources and Evaluation*.

Mendes, P. N., Jakob, M., Garcia-Silva, A., and Bizer, C. (2011). Dbpedia spotlight: Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems (I-Semantics)*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2473–2479, Istanbul, Turkey, May. European Language Resources Association (ELRA).

San Vicente, I., Saralegi, X., and Agerri, R. (2015). Elixa: A modular and flexible absa platform. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 748–752. Association for Computational Linguistics.

Solla Portela, M. A. and Gómez Guinovart, X. (2015). Termonet: Construcción de terminologías a partir de WordNet y corpus especializados. *Procesamiento del Lenguaje Natural*, 55:165–168.

Solla Portela, M. A. and Gómez Guinovart, X. (2016). Dbpedia del gallego: recursos y aplicaciones en procesamiento del lenguaje. *Procesamiento del Lenguaje Natural*, 57:139–142.

Solla Portela, M. A. and Gómez Guinovart, X. (2017). Diseño y elaboración del corpus SemCor del gallego anotado semánticamente con wordnet 3.0. *Procesamiento del Lenguaje Natural*, 59:137–140.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147.

9. Language Resource References

TALG Research Group. (2016). *CTG Corpus [Galician Technical Corpus]*. University of Vigo, SLI resources, 1.0, ISLRN 437-045-879-366-6.

TALG Research Group. (2018). *SLI NERC Galician Gold CoNLL*. University of Vigo, SLI resources, 1.0, ISLRN 435-026-256-395-4.

Modeling Northern Haida Verb Morphology

Jordan Lachler[1], Lene Antonsen[2], Trond Trosterud[2], Sjur N. Moshagen[2], Antti Arppe[1]
[1]University of Alberta
[2]UIT Arctic Univerity of Norway
Email: {lachler, arppe}@ualberta.ca
{lene.antonsen, trond.trosterud, sjur.n.moshagen}@uit.no

Abstract

This paper describes the development of a computational model of the morphology of Northern Haida based on finite state machines (FSMs), with a focus on verbs. Northern Haida is highly endangered, and a member of the isolate Haida macrolanguage, spoken in British Columbia and Alaska. Northern Haida is a highly-inflecting language whose verbal morphology relies largely on suffixes, with a limited number of prefixes. The suffixes trigger morphophonological changes in the stem, participate in blocking, and exhibit variable ordering in certain constructions. The computational model of Northern Haida verb morphology is capable of handling these complex affixation patterns and the morphophonological alternations that they engender. In this paper, we describe the challenges we encountered and the solutions we propose, while contextualizing the endeavour in the description, documentation and revitalization of First Nations Languages in Canada.

Keywords: less-resourced/endangered languages, language modeling, morphology

1. Introduction

While the study of the Indigenous languages of North America has been a mainstay of descriptive linguistics over the past century or more, these languages have received much less attention in the computational linguistic literature. This has begun to change in recent years, e.g. through work initiated on Algonquian languages such as Plains Cree (Snoek et. al., 2014; Harrigan et. al., 2017), East Cree (Arppe et al., 2017a), Odawa (Bowers et al. 2017), as well as Dene languages such as Tsuut'ina (Arppe et al. 2017b). This work has come about through computational linguists, field linguists and Indigenous communities working together to create tools and applications, which can support the continued use and learning of these often threatened languages (Arppe et al., 2016). In this paper, we provide a description of a computational model of the verbal morphology of one highly endangered North American Indigenous language, Northern Haida, which was created in furtherance of these goals.

2. Background

2.1. Northern Haida

Northern Haida, known as *Xaad Kíl*, is one of two members of the macrolanguage Haida (Simons and Fennig, 2017). Speakers of Northern Haida live in a small number of communities on the Haida Gwaii archipelago off the coast of British Columbia, and on the islands of far southeastern Alaska. Haida is an isolate, with no known linguistic relatives. Today, there are fewer than a dozen fluent speakers of Northern Haida remaining, all of them over the age of 70, but also an active and growing cohort of younger learners of the language.

2.2. Morphological Structure of Verbs

Most verbs in Northern Haida are entirely suffixing in a relatively neatly agglutinative manner. They inflect for a range of grammatical categories including Tense, Aspect, Mood, Evidentiality, and Polarity among others, but not for the person and number values of their subject or object. The only exception to this is the Third Person Plural suffix *-'wa-*, which is used to indicate that there is a definite third person plural participant in the clause.

In total, there are fewer than 20 distinct suffixes, but they can combine to give nearly 200 basic paradigmatic forms for each verb. While the pattern of suffixes is quite agglutinative in some respects, there are several complications which are worthy of note.

To begin with, the suffixes frequently trigger morphophonemic changes in the verb stem to which they attach (and vice versa). For instance, the Present tense suffix has three forms, *-ng*, *-ang* and *-gang*. The form *-ang* occurs only after verb stems that end in either *-s* or *-d*. The addition of the Present suffix to verb stems of this type causes the last vowel of the stem to change to *-ii-*, as with the verb *k'agáangad* "oversleep" and its Present tense form *k'agáangiidang*.

Moreover, while the order of suffixes is not variable, it is also not entirely consistent. For example, the Negative suffix {*-'ang-*} and the Habitual aspect suffix {*-gang-*} occur in different orders, depending on what other suffixes are present. In the Present tense, the Habitual comes first, followed by the Negative: *gatáa-gang-'ang-gang* (eat-HAB-NEG-PRES) "never eats". However, in the Reported Past, the order is reversed,

with the Negative coming first and the Habitual coming second: *gatáa-'ang-gaang-aa-n* (eat-NEG-HAB-REP-PAST) "never used to eat".

A further complication arises from certain logical suffix combinations which are blocked. The Habitual aspect suffix and the Future tense suffix do not occur together, even though they are not semantically incompatible. The Habitual aspect suffix also does not occur adjacent to the Past tense suffix. Instead, the single Customary tense-aspect suffix {*-giinii*} is used: *gatáa-giinii* (eat-CUST) "used to eat". Note that this only applies when those two suffixes would be adjacent to one another; when other suffixes intervene (as in the example of *gatáa'anggaangaan* above), then both suffixes can and do occur.

While the large majority of Northern Haida verbs are solely suffixing, as described above, a significant minority have an obligatory prefix slot as well. This slot is filled by a Classifier prefix, which gives further semantic specification about the absolutive argument in the clause. For instance, the verb stem CLSF+*gang* "hold something of CLSF-shape" require a monosyllabic prefix which indicates what type of object is behind held. Typical forms include: *dlagáng* "hold something animate", *tlagáng* "hold something thin and flat", *skáagang* "hold something small and round", *k'íigang* "hold something large", and *chagáng* "hold a bag of something". The number of possible Classifiers varies by verb, from just a few, to a couple dozen, to several hundred. These verbs are then suffixally inflected in the same way as other verbs.

A final complication in the conjugation of Northern Haida verbs is its robust system of Auxiliaries. Auxiliaries occur immediately following the main verb. The main verb appears in one of several Construct states, depending on the identity of the following Auxiliary. The inflections, then, manifest as suffixes on the Auxiliary. For example, the main verb *k'ajúu* "sing" can be followed the Auxiliary *áwyaa* "very, really". When these combine, *k'ajúu* occurs in its Construct form *k'ajáaw*, and the inflectional suffixes occur on the Auxiliary *áwyaa*: *k'ajáaw áwyaa-gan* (sing:CONST very-PAST) "really sang". There are around 40 of these Auxiliaries, which can sometimes stack up after the main verb. The result is a multi-part verb form, written orthographically as two or more words, but bearing a single inflection, e.g.: *káayd déed gudáa-ng* (leave right.away want.to-PRES) "wants to leave right away". There are around 160 attested Auxiliary strings, each of which can in principle be inflected for the full range of nearly 200 endings found on main verbs. Thus, including the possible Auxiliaries, each verb phrase can in principle be inflected for approximately 32,000 distinct forms.

3. Computational Modeling

3.1. Framework

As the computational formalism for implementing our model for Northern Haida morphology, we make use of Finite-state machines (FSMs) (cf. e.g. Beesley & Karttunen, 2003) which have become one standard way for computationally modeling the morphological structure of words in natural languages. There are currently several implementations of FSM compilers, e.g. *xfst* (Beesley & Karttunen 2003), *foma* (Hulden 2009) and HFST (Lindén et al. 2011), of which the first is available for non-commercial research use and the second and third are open source resources. The key advantages of FSMs are many, but most crucially they are designed for rule-based definition of paradigms, which does not require large corpora from which to learn such rules. This is helpful in the case of endangered languages, such as Northern Haida, for which such corpora are usually lacking.

Furthermore, FSMs allow for easy integration with other software applications, for instance as spell-checking modules within word-processors, morphologically “intelligent” electronic dictionaries, and “intelligent” computer-aided language-learning applications. Here, we make use of the *Giella* infrastructure, developed by the Giellatekno and Divvun research teams at the University of Tromsø (Trosterud 2006; Moshagen et al. 2013), which provides ready-made solutions for the integration of an FSM-based computational model as part of such end-user applications.

3.2. Design considerations and choices

In designing a finite-state computational model, one has to decide whether to model morphophonological alternations at stem+affix junctures by (1) dividing stems into subtypes which are each associated with their own inflectional affix sets that can simply be glued onto the stem, or by (2) modeling such morphophonological alternations using context-based rewrite rules. Furthermore, one has to decide the extent to which one treats affix sequences by splitting these into their constituent morphemes, each associated with one morphosyntactic feature, or rather treats affixes as unanalyzed chunks which are associated with multiple morphosyntactic features (Arppe et al., 2018; Arppe et al. 2017a, 2017b). The more one splits affix sequences, the more one may need to develop and test rules for dealing with morphophonological alternations at these morpheme junctures, whereas in the case of chunking such alternations are precomposed within the chunk. In contrast, the more one uses chunks, the more one has to enumerate chunks based on the number of relevant inflectional subtypes.

While the chunking strategy is not parsimonious and compact in terms of linguistic description, in our experience it results in FST source code which is

nevertheless structurally quite flat and easily comprehensible for scholars who are not specialists in the language in question. Importantly, current finite-state compilers, e.g. *xfst*, HFST, or *foma* (Beesley and Karttunen 2003; Lindén et al. 2011; Hulden 2009), implement a minimization procedure on the finite-state model, so that recurring realizations of string-final character sequences and associated morphological features are systematically identified and merged, resulting, in the end, in a relatively compact model. On the other hand, if some aspect of the chunked morpheme sequences needs to be changed, with the chunking strategy these have to be implemented in multiple locations.

For the Northern Haida model, we decided to (1) split the pre-stem morphemes (the classifiers), as there are very few morphophonological phenomena and these are very regular; (2) entirely chunk the post-stem suffix morphemes, associating the chunks with multiple morphological feature tags; and (3) make maximal use of inflectional subtypes using technical stems and post-stem technical suffix chunks, an approach which we call *maximal chunking*. Thus, we will require only a few morphophonological rules for the stem-suffix morpheme juncture. These morphophonological rules are implemented using the TWOLC formalism within the FST framework. As to the rest, the LEXC formalism in the FST framework is used to define the concatenation of stems and affixal morphophology (with morpheme sequences treated chunks). There are currently just under 10,000 verb stems in the lexicon (Enrico, 2005; Lachler, 2010).

We now discuss the motivations for these design choices (which are similar to, and indeed the original inspiration for those adopted for East Cree, cf. Arppe et al. 2017b) from the perspective of documentary linguists working on an endangered language, and their practical implementation. The morphophonemic changes that the endings trigger on the verb stems are limited to the rhyme of the final syllable of the stem. As such, verb stems are grouped into sub-lexica by the shape of their final rhyme. They are listed with all but their final rhymes in the lexicon, which we term the *technical stem*. The technical stem is then augmented with various final rhymes to produce an inflectable stem.

For example, many verbs have a lexical stem that ends in a short *-a-* (e.g. *skyáana* "be awake"). Before certain endings, the final *-a-* remains short, while before other endings it lengthens to *-aa-*. As such, these verbs are listed in the lexicon with their technical stem which does not include this variable-length final vowel.

```
skyáan CLASS-A "be awake" ;
```

The technical stem then directs to the appropriate continuation lexicon (in this case, CLASS-A, named after the form of the underlying rhyme). Here, two stems are created, one with a short *-a-* and another with a long *-aa-*.

```
LEXICON CLASS-A
 :a CLASS-A-STEM-1 ;
 :aa CLASS-A-STEM-2 ;
```

These two stems then each direct to different continuation lexica, one containing all of the endings which condition a final short *-a-* in the stem (such as in the Present tense form *skyáanang*), and the other containing all of the endings which condition a final long *-aa-* in the stem (such as in the Future tense form *skyáanaasaang*).

```
LEXICON CLASS-A-STEM-1
+V+PRES:ng # ;
LEXICON CLASS-A-STEM-2
+V+FUT:saang # ;
```

We next illustrate how our maximal chunking model works in practice. For example, the inflected form *skyáana'ang'ugan* "were not awake" would be composed from the CLASS-A-STEM-1 form *skyáana* and the following ending or "chunk":

```
LEXICON CLASS-A-STEM-1
+V+NEG+3PL+DIR+PAST:'áng'ugan # ;
```

This ending carries the grammatical features of negative polarity (+NEG), a third person plural participant in the clause (+3PL), direct evidentiary knowledge (+DIR) and past tense (+PAST).

The addition of this ending yields the form *skyáana'áng'ugan*, with two accented syllables, which is incorrect. This triggers a TWOLC rule which deletes any accent after the first accent in the word, producing the correct form *skyáana'ang'ugan*.

While it is straightforward to map those four grammatical tags onto the three component suffixes of that ending (*-'ang-* Negative, *-'u-* 3rd Person Plural, and *-gan* Direct Past), we took the maximal chunking approach in order to minimize the development time required to create the morphological model. Although all of the complications described above in Section 2.2 can be easily handled within LEXC and TWOLC using a non-chunking approach, significant time is required for the field linguist to learn all of the formalisms required to handle morphophonemic alternations, morpheme blocking, and any context-dependent morpheme orderings. By adopting a chunking approach, the morphological model more closely mirrors the field linguist's original documentation of the language. This, in turn, streamlines the development and testing of the FSM, allowing the team to move more rapidly towards the creation of practical tools and applications.

4. Evaluation

For evaluation, we created complete inflectional paradigms for each of the inflectional subtypes, consisting of pairing of the realized surface forms and the underlying morphosyntactic analyses. Due to our maximal chunking approach and the comprehensiveness of the lexical database we have at our disposal, we have been able to reach 100% accuracy for both word-form analysis and generation.

5. Conclusion

The development of computational models for morphologically complex endangered languages can often be slowed due to the difficulty of adapting existing language descriptions into the appropriate computational formalisms. The approach offered here for Northern Haida seeks to minimize any such delay by simplifying the formal machinery required to produce correct results, even where this may run afoul of descriptive parsimony. This should allow more descriptive linguists to develop computational models of the language they have documented, and to move more rapidly to the stage where useful tools and applications can be deployed in the community.

6. Acknowledgements

This work has been undertaken in the Alberta Language Technology Lab (ALTLAb: URL: http://altlab.artsrn.ualberta.ca), University of Alberta, as part of a more general research program to computationally model North American Indigenous languages (Arppe et al. 2016), and it has been funded by the Social Sciences and Humanities Research Council of Canada (SSHRC) through a Partnership Development Grant (890-2013-0047), a Connections Outreach Grant (611-2016-0207), and a KIAS Research Cluster Grant 2015-2018 (University of Alberta).

Bibliographical References

Arppe, A., Cox, C., Hulden, M., Lachler, J., Moshagen, S. N., Silfverberg, M. and Trosterud, T. (2017). Computational Modeling of Verbs in Dene Languages: The Case of Tsuut'ina. *Proceedings of the 2016 Dene Languages Conference*, "Red Books" series, Alaska Native Language Center, University of Alaska, Fairbanks.

Arppe, A., Junker, M.-O. Harvey, C., and Valentine, J. R. (2018). Algonquian verb paradigms. a case for systematicity and consistency. *Papers of the Algonquian Conference 47 (PAC47),* 1-22.

Arppe, A., Junker, M-O, and Torkornoo, D. (2017). Converting a comprehensive lexical database into a computational model: The case of East Cree verb inflection. *Proceedings of the 2nd Workshop on Computational Methods for Endangered Languages (ComputEL-2)*, 43-47, University of Hawai'i, Manoa, 6-7 March 2017. ACL Anthology.

Arppe, A., Lachler, J., Trosterud, T., Antonsen, L., and Moshagen, S. N. (2016). Basic Language Resource Kits for Endangered Languages: A Case Study of Plains Cree. In: Soria, C., Pretorius, L., Declerck, T. Mariani, J., Scannell, K., and Wandl-Vogt, E. (eds). Proceedings of the LREC 2016 Workshop *CCURL 2016 – Collaboration and Computing for Under-Resourced Languages – Towards an Alliance for Digital Language Diversity*, (Portorož, Slovenia, 23 May 2016), 1-8.

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications, Stanford, CA.

Bowers, D., Arppe, A., Lachler, J., Moshagen, S. N. and Trosterud, T. (2017). A Morphological Parser for Odawa. *Proceedings of the 2nd Workshop on Computational Methods for Endangered Languages (ComputEL-2)*, 1-9, University of Hawai'i, Manoa, 6-7 March 2017. ACL Anthology.

Enrico, J. (2005). *Haida Dictionary*. University of Alaska-Fairbanks and Sealaska Heritage Institute.

Harrigan, A., Schmirler, K., Arppe, A., Antonsen, L. Moshagen, S. N., Trosterud, T. and Wolvengrey, A. (accepted pending revisions 22-Jan-2017; revised version submitted 29-Sept-2017). Learning from the Computational Modeling of Plains Cree Verbs. *Morphology,* 27(4), 565–598.

Hulden, M. (2009). Foma: a finite-state compiler and library. In *Proceedings of EACL*, pp. 29–32, Athens. Association for Computational Linguistics.

Lachler, J. (2010). *Alaskan Haida Dictionary*. Sealaska Heritage Institute.

Lindén, K., E. Axelson, S. Hardwick, M. Silfverberg, and Pirinen, T. (2011). HFST – Framework for Compiling and Applying Morphologies. *Proceedings of Second International Workshop on Systems and Frameworks for Computational Morphology (SFCM)*, 67–85.

Moshagen, S. N., Pirinen, T. and Trosterud, T. (2013). Building an open-source development infrastructure for language technology projects. *Proceedings of the 19th Nordic Conference of Computational Linguistics* (NODALIDA 2013). Linköping Electronic Conference Proceedings #85: 343-352.

Simons, Gary F. and Charles D. Fennig (eds.) (2017). *Ethnologue: Languages of the World*, Twentieth edition. Dallas, Texas: SIL International. Online version: http://www.ethnologue.com.

Snoek, C., Thunder, D., Lõo, K., Arppe, A., Lachler, J., Moshagen, S., and Trosterud, T. (2014). Modeling the Noun Morphology of Plains Cree. In *Proceedings of ComputEL: Workshop on the use of computational methods in the study of endangered languages*, 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, Maryland, 26 June 2014, 34-42.

Trosterud, T. (2006). Grammatically based language technology for minority languages. In *Lesser known languages of South Asia*, 293–316. Mouton de Gruyter, The Hague.

Low-resource Post Processing of Noisy OCR Output for Historical Corpus Digitisation

Caitlin Richter, Matthew Wickes, Deniz Beser, Mitch Marcus
University of Pennsylvania
Departments of Linguistics and Computer and Information Science, 3330 Walnut Street, Philadelphia PA 19104
{ricca, wickesm, dbeser}@seas.upenn.edu, mitch@cis.upenn.edu

Abstract

We develop a post-processing system to efficiently correct errors from noisy optical character recognition (OCR) in a 2.7 million word Faroese corpus. 7.6% of the words in the original OCR text contain an error; fully manual correction would take thousands of hours due to the size of the corpus. Instead, our post-processing method applied to the Faroese corpus is projected to reduce the word error rate to 1.3% with around 65 hours of human annotator work. The foundation for generating corrected text is an HMM that learns patterns of OCR error and decodes noisy OCR character sequences into hypothesised correct language. A dictionary augments the HMM by contributing additional language knowledge, and a human annotator provides judgements in a small subset of cases that are identified as otherwise most prone to inaccurate output. An interactive workstation facilitates quick and accurate input for annotation. The entire toolkit is written in Python and is being made available for use in other low-resource languages where standard OCR technology falls short of desirable text quality. Supplementary analyses explore the impact of variable language resource availability and annotator time limitations on the end quality achievable with our toolkit.

Keywords: Faroese, OCR, low resource corpus development, human-assisted post processing

1. Introduction

1.1. Motivation

We provide a toolkit to improve unsatisfactory text quality in a newly digitised Faroese corpus (Føroyamálsdeildin, nd). The 2.7 million word corpus contains historical interviews of unique linguistic and cultural value, which would provide valuable material for research in a variety of disciplines if it were readily accessible (e.g. Heycock and Wallenberg, 2013; Galbraith, 2016). The interviews were recorded on tape up to 50 years ago, transcribed by typewriter, and later scanned (Figure 1) before being input to unknown commercial optical character recognition (OCR) software for text extraction.

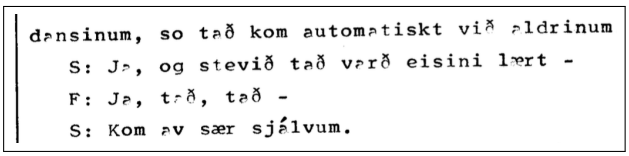

dansinum, so tað kom automatiskt við aldrinum
S: Ja, og stevið tað varð eisini lært -
F: Ja, tað, tað -
S: Kom av sær sjálvum.

Figure 1: Partial scanned typewriter page of Faroese interview transcription.

This process significantly corrupted the text with incorrect recognition. A lack of enough language-specific data for good Faroese language models in the initial OCR process had a large role in damaging the text quality; many of the errors produce tokens with distinctly un-Faroese letter sequences, or even numbers and punctuation in the middle of words (like OCR error form *r!ey^ur* for *deyður*). Close to 8% of tokens (words) contain at least one error, suggesting a rough expectation for on average every sentence to contain one incorrectly recognised (non-)word. This error rate severely degrades the usability of the corpus for NLP, computational linguistic research, or any interest that relies on basic text search to identify relevant documents within the 2.7 million word corpus.

Faroese is spoken by only about 50,000 people on the Faroe Islands, located between Norway, Scotland, and Iceland (Thráinsson et al., 2012). It is a Germanic language whose nearest relative is Icelandic; the two are not normally mutually intelligible in speech to unpractised listeners, and Faroese is considerably more different (certainly not mutually intelligible) from its next closest relatives, the larger and better-resourced Mainland Scandinavian languages Danish, Norwegian, and Swedish. However, many native speakers of Faroese do speak Danish as a second language, in current times often used for university education or employment in Denmark (while Faroese is the language of home, daily life, school, etc. on the Faroe Islands). This reflects past Danish rule of the Faroe Islands; of particular note to present purposes, this rule included a period of around 300 years in which the Faroese language was generally banned from official and written contexts (Thráinsson et al., 2012). The modern orthography used to transcribe the corpus we correct was developed in the late 1800s, after the language ban was lifted, by linguist V.U. Hammershaimb; it incorporates etymological reconstruction and morphophonemic (rather than phonetic) representation, which has the effect that it is not inherently biased to resembling the pronunciation of particular dialects. In this corpus, while lexical items will vary according to speakers' dialects, the spelling itself can be considered a stable standardised system. The Faroese alphabet officially has 29 letters, but in practise a few more are found - variant conventions like **ø** and **ö** which are intended as the same letter in this context, and foreign borrowings like **å** and **c**. We suspect, based on certain qualitative patterns in recognition, that the OCR system that produced the Faroese corpus text may have had Danish as its primary language, with the Faroese alphabet traded in.

1.2. Approach

We developed a suite of post-processing tools that a corpus can be fed through, obtaining adequate machine-readable text despite the excessive error rate from standard OCR. The toolkit is general and we are distributing it for use in any corpora where conventional OCR lacking sufficient language-specific resources is inadequate.

Some human annotation seems necessary in the digitisation of the Faroese corpus. In the first place, automated OCR is unsatisfactory; then, we find also that building an entirely automated post-processing system, although it improves the text, still leaves far too much error. However, incorporating a moderate investment of annotator time during post-processing leaves an acceptable token error rate below 1.5% in this low-resource setting.

Automated tools handle most of the corpus without human intervention, but also identify scenarios where automated accuracy tends to be lowest. In these difficult cases, the automatic tools restrict the problem to a forced choice of at most 3 potential corrections for an error, which are then sent to a human annotator for adjudication. Our projections estimate a final token error rate of 1.3% with around 65 hours of annotator work to process the entire Faroese corpus.

The only alternative in order to produce machine-readable text of acceptable quality would be manual human work. A human editor moving quickly can correct around 1,000 words of text per hour, for an expectation of 2,700 hours of work to process the Faroese corpus while still being prone to some degree of mistakes and inattention leading to a comparable token error rate in the corpus. As an alternative to correction, the entire corpus could be transcribed from scans by typists, skipping the OCR entirely; this too would take substantial time for 2.7 million words, and the cost of hiring professional transcribers is often prohibitive for academic or community owners of language resources being digitised. Our toolkit is designed to achieve similar final corpus quality at much lower annotator cost.

1.3. Contributions

The Faroese corpus whose noisy OCR we correct represents a general problem in the dissemination of physical documents in low resource language varieties. A basic pipeline is that a physical document is scanned to a digital image, the digital image has machine-readable text extracted by OCR, and then the text is available for distribution. OCR is considered a solved problem for modern type in large-population languages, but standard technologies perform poorly in low resource language varieties, or when source documents are physically degraded by time, environment, or show visual variation due to the original printing method (Afli et al., 2016). Digitised resources could be distributed as scanned images rather than machine readable text, and communities may still access the heritage and culture represented in the resources. However, text facilitates better distribution due to reduced file size, and large corpora are not easily navigable for any purpose without searchable text. Text is of course also essential for NLP, computational linguistics, and other research uses of corpora. Therefore, we consider machine-readable text a highly important element for any digitised text corpus. Our toolkit becomes a post-processing stage in the resource dissemination pipeline, improving error-filled OCR output before the text is distributed. The toolkit's code, in Python, is being made available for general use.

2. Overview of Toolkit

A suite of five components and their relations are introduced briefly here, before individual presentation in §3.

Character decoding. The core generation of OCR corrections uses a character HMM to decode noisy input text into k-best candidates for correct Faroese.

Alignment. Word-aligned parallel texts match noisy OCR with its gold standard correction. Aligned texts are used to train the character HMM and for development/evaluation.

Dictionary. A dictionary helps filter out spurious (non-word) character sequences generated by HMM decoding.

Heuristic Decisions. A central decision module draws on the output of character decoding and the language dictionary to select output forms or determine when to seek human adjudication between k-best candidates.

Interface. The human annotator is presented with a potential OCR error and up to three candidate corrections in an interactive workstation, and inputs their selected correction.

3. System Description

3.1. Text Alignment

Aligned texts (Table 1) contain word pairs used to train models of OCR character error. Aligned parallel text like this is also required for system tuning and evaluation.

Correct	Original
tøðini	tøðini
v**ó**rðu	v**ú**rðu
borin	borin
út	út
á	**§**
bøin	bøin
í	**i**

Table 1: Sample of aligned text, including OCR errors (underlined).

Corrected versions of several Faroese corpus texts were available as they had been previously produced and used for linguistics research. Aligning a noisy OCR document to its corrected version is not trivial, because the two texts normally have different token counts, different word and sentence segmentations, and in some cases different line divisions. However, both should contain corresponding elements in the same order. We used the Needleman-Wunsch sequence alignment algorithm (familiar also as 'dynamic time warping' when applied to speech) on pairs of noisy and corrected Faroese text files to find optimal alignments (Kleinberg and Tardos, 2006).This dynamic programming algorithm creates a grid of character (mis)matches between two files, and uses these to iteratively calculate the best word-level alignment minimising overall mismatch. A fixed window for mismatch length reduces runtime.

3.2. Character Decoding

To generate corrections for tokens in noisy OCR text, we use a hidden Markov model to decode the original OCR into hypothesised correct Faroese output. The decoding process takes a token from the input file and identifies the sequence of actual Faroese characters most likely to have been recognised by OCR as that token, which may or may not be identical to the original character sequence. We use a modified version of the Viterbi algorithm to find the top-k most probable character sequences, instead of the standard single best character sequence. This modified algorithm uses beam search, maintaining a list of the k-best most probable sequence possibilities at each time slice within the decoding process.

The states of the HMM are the characters present in the corrected training data: the Faroese alphabet, numbers, and punctuation. The HMM output symbols are the slightly larger set of characters present in the original noisy OCR, such as đ alongside Faroese ð. Emission parameters of the HMM are learned from parallel training data, while initial and transition probabilities are learned from corrected texts. Laplace smoothing enables generalisation to character transitions and OCR errors not seen in the training data.

Initial Error	7.8%
Final Error	**5.4%**
Introduced Error	0.8%
Tokens Successfully Corrected	3.2%

Table 2: Operation of HMM alone on dev set. All quantities are expressed by percentages of corpus tokens (words).

The HMM is trained on 170k words of parallel Faroese data, and decoding uses beam width k=4. This HMM alone is not able to effectively correct OCR errors. Table 2 illustrates projected performance for always selecting the most probable (Viterbi) decoding on a test set of 70k words held out from its training data. This poor performance motivates our strategy of overgeneration (k-best decoding) before taking additional steps to refine the OCR correction.

The simple HMM topology that we use is capable of correcting single character errors, but the noisy OCR on the Faroese corpus also produced errors of multiple characters read as a single character and single characters read as multiple characters. These errors are generally rare, but two types occur with higher frequency: 'll' read as 'H' and 'rn' read as 'm'. These two error types are present in 0.18% of the tokens in the parallel Faroese data.

Correction of these errors is done after computing the k-best decodings of the original token, and is corpus-specific. If none of the k-best decodings are present in the dictionary then the token is checked for whether it satisfies the conditions for any of the character mis-segmentation errors known to be frequent in the corpus.

1. If the original token contains an 'H' in a non-initial position, then any occurrences of 'H' are replaced by 'll' and the k-best decodings are then found for the new token. The k-best among the original decodings and the new decodings are then selected. Only non-initial 'H' is replaced because no words beginning with 'll' are present in the Faroese dictionary.

2. If the original token contains an 'm' then variant tokens are created substituting 'rn' in place of 'm'. Every occurrence of 'm' is either replaced by 'rn' or left unchanged. The k-best decodings are computed for each of these variants and then the k-best among those and the original decodings are selected.

If a token satisfies conditions for both errors then both processes are applied.

The two strategies applied here, complete replacement in (1) and variable mixing in (2), can be generalised to other OCR errors of this nature in corpora where such errors are more prevalent.

3.3. Dictionary

The Faroese dictionary contains around 450k types. It was created originally for spell checking (J. S. Andersen, 2001), and lightly augmented for our corpus (for example, common abbreviations like *Fx* 'speaker' and *Fxx* 'speaker 2'). Around 5% of tokens in the corrected text are still not present in the dictionary.

This dictionary acts as a filter on the k-best decodings, when at least one candidate is in the dictionary and at least one is not. In these cases, a candidate that is in the dictionary may be a better choice than one that is not, even if this is counter to their decoding probabilities. For example, the 4-best HMM decodings for the token *stðani* are [stóani stðani stúani síðani]; of these, the fourth *síðani* is correct and is present in the dictionary, while none of the other three generated forms are members of the dictionary. Similarly, the two best decodings for *slektaðu:* with the token ending in a colon are [slektaðu: slektaður]. Here *slektaður* is correct, and it is in the dictionary while *slektaðu* (punctuation stripped for dictionary check) is not. Therefore, if some but not all generated candidates are found in the dictionary, non-members can be removed from the candidate set. The dictionary is also used to assess whether to replace original tokens with any non-identical candidate at all: if the original token is already in the dictionary, then it is less likely to be an error in need of correction.

3.4. Central Decisions

OCR correction is ultimately managed by a module that combines all available information with heuristic handling strategies, and either selects the best word decoding or solicits human judgement.

The information available to the central decision module includes an original token, four candidate decodings of it (one of which may be identical to the original), probabilities for each decoding under the HMM, and binary dictionary membership checking. Decisions about which form to use as output, or whether to ask a human annotator to adjudicate between a set of candidates, must derive from this information.

The current decision procedure is in broad summary:

1. Construct an initial ranked candidate list, from the original token and *k*-best list: [ORIG, K1, K2, K3, K4]

2. Check basic attributes of list members, such as whether they are members of the dictionary and whether they are identical with the original input form.

3. Based on token and candidate attributes, handle the correction by either making no change, changing to the highest-ranked non-identical candidate present in the dictionary, or seeking human annotator judgement.

The decision procedure applies invariantly whether the best decoded candidate (K1) is different or identical to the input, that is, whether the character model estimates that the original OCR contains an error or not.

This basic outline is implemented through a set of heuristics that allow different classes of tokens to be handled differently, increasing correction accuracy and maximising efficiency for the annotator by better targeting their efforts. Tokens that are being corrected can be classified based on four binary attributes visible to the system at correction time: whether the top-ranked generated candidate K1 is identical to the input token; whether the input token is in the dictionary; whether the best generated candidate K1 is in the dictionary; and whether *any* generated candidates are in the dictionary. Due to frequent logical dependencies between the attribute values, 9 classes are created. Each class can receive an independent decision for how to best handle its tokens, with these heuristic decisions optimised on a dev set: possible choices for each are to select the original token, the top generated candidate, the top candidate appearing in the dictionary (if present), or human annotation. The decision for how to handle each class is determined by oracle evaluation run on a dev set of parallel original/corrected (gold standard) text. This shows what the optimal decision on the dev set would have been, as well as what percentage of the corpus falls into each of the nine classes so that annotator time can be allocated to make the largest possible impact while minimising workload. If the dev set reasonably represents the corpus as a whole, then the optimal heuristic settings for the dev set should lead to accurate and efficient corrections when used to correct the entire corpus.

3.4.1. Fit to Faroese corpus

We used four-fold cross validation to set heuristic options (based on 10,000 words parallel text) and test the projected performance of these settings when applied to other text (another 10,000 words). The parallel text used for evaluation is 500-word excerpts of 40 texts from the corpus, randomly sampled with restriction of no more than one text per interviewee to enforce wide coverage of vocabulary, year interviewed/transcribed, and individual typewriters. The corrected versions of the parallel texts were finalised with reference to scans of their original typewritten copies to produce a gold standard for precise evaluation.

In four-fold cross validation, there was no variation in which heuristic settings were found to be optimal for different random 10,000-word segments of the dev data, so we treat this as a single correction system. The settings found for Faroese correction can be summarised across the 9 cases in the following decision procedure:

1. If the top-ranked generated candidate K1 is in the dictionary, use K1 (which may be identical to the input).

2. Else, if the original input is in the dictionary, use that input.

3. Else, if some candidate(s) K2-4 are in the dictionary, use the highest-ranked dictionary member.

   [3.1.] However, consider sending the token to an annotator if K1 is identical to the input and K1 is overwhelmingly more probable than K2 under the character model.

4. Else, when nothing is in the dictionary, send to an annotator.

   [4.1.] However, if K1 is identical to the input, consider using the input form without sending to an annotator if the difference in probability between K1 and K2 is large enough.

That these particular settings emerged from the dev set provides some insight on both the corpus and the more general operation of the system we are using. Step 1, using K1 if it is in the dictionary, applies even when the original input token differs from K1 and is also in the dictionary. This introduces some error, but implements far more changes that turn out to be real corrections (such as changing *Olavur* to *Ólavur*). Step 1 is the only case (besides human annotation) in which an original token that is present in the dictionary may end up changed - if K1 is not in the dictionary but any K2-4 is, the original input token that is in the dictionary will be used for output. This reflects tensions between confidence in the character HMM and the original text, with the dictionary 'endorsing' both; evidently, the OCR quality is just bad enough that the character model's top choice is a better bet in this situation - but only if it is the character model's top choice, and not K2-4.

The human annotator's role is mainly to adjudicate error-prone cases between the character model, which has confidence in a best candidate decoding, and the dictionary, which (sometimes due to dictionary incompleteness, and sometimes due to real OCR error) shows no confidence in the original OCR token (Step 3.1) or in any other candidate as well (Step 4). In both Step 3.1 and 4.1, we find that the character probability difference between the top candidate and the next-best candidate, when the top candidate is identical in form to the original token, is a fairly good signal for whether the *correct* form of that token is also identical with its original form. The precise value used is given in Equation 1, the ratio of this difference to the probability of K1.

Equation 1. $$\frac{P(K1)-P(K2)}{P(K1)}$$

It turns out that the value of this K1-K2 difference ratio tends to approach 1 when K1 is identical to the original token and to the gold standard correct token.

In Step 3.1, the character model endorses the original form of the token, while the dictionary selects some other lower-ranked candidate K2-4 because the original form/K1 is not

in the dictionary. In the majority of these cases the lower-ranked dictionary member is correct; however, there are some tokens for which the original form is correct and switching to a candidate K2-4 will lead to highly undesirable introduced error in the corpus. Close to 3% of the corpus falls into condition 3.1, so it is not feasible to pass all such tokens to a human annotator. Instead, a high K1-K2 difference ratio (threshold >0.99, set on 70k dev set; threshold referred to as TH3.1 in Table 5 below) flags most of the tokens for which K1 is correct - and flags few enough false positives (for which K2-4 is correct) that annotation of all flagged tokens is still practical - so that introduced error is minimised with efficient human effort.

In Step 4 when absolutely nothing is in the dictionary, if the top candidate K1 differs from the original token then there is very little good information for the correction system to work with so all such tokens are sent to the human annotator. These are only about 0.6% of tokens in the corpus, and they can be a good source of progressive augmentation to the dictionary. For cases falling under 4.1 where K1 does have the same form as the original input token, the K1-K2 difference ratio can provide a useful signal saving human effort. As in Step 3.1, this ratio tends towards 1 when K1/input form is correct; therefore, if it is sufficiently high (threshold >0.95, set on 70k dev set; referred to as TH4.1 in Table 5) then the token can bypass the annotator and be left as its original form/K1.

Although this section has presented corpus-specific settings for the Faroese typewritten data we are correcting, the basic heuristic method is widely adaptable, subdividing the correction problem into token classes from observable attributes and estimating optimal output settings for each class from a dev set of parallel text.

3.5. Interface

Annotator judgements are collected in a simple text interface.

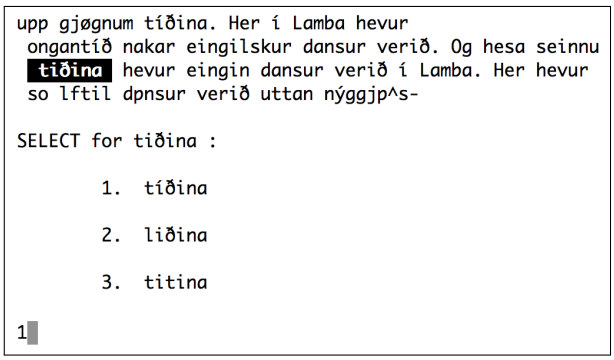


Figure 2: Interface for annotation.

The interface presents the target word highlighted in context, facilitating a quick and accurate decision. For each target word, up to three candidate corrections are offered; any of them can be chosen by number. The annotator can also type a single-character control code to keep the original word. If the original word is an error but none of the suggested candidates are correct, the annotator can type custom input. Font, size, and colours are easily controlled by users' general terminal interface settings.

Trials of this interface while producing parallel training/test data (further post-corrected by hand for gold standard) yield an estimated annotation speed of 4 seconds per target word, or 900 annotations per hour.

4. Results

We first present the primary evaluation of our correction system as developed in §3.4.1; then, three supplementary evaluations illustrate variation in performance as a function of annotator time, amount of HMM character model training data, and amount of heuristic-setting dev set data.

4.1. Main system

Applying the full correction procedure in four-fold cross validation with heuristics set on 10,000 words and tested on another 10,000 words of parallel text yields the projections for final corpus quality and human annotator cost shown in Table 3.

Initial Error	7.6%
Final Error	**1.3%**
Annotator Task	2.2%
Introduced Error	0.4%
Tokens Successfully Corrected	6.6%

Table 3: Current toolkit operation including annotation. All quantities are expressed by percentages of corpus tokens (words).

In this evaluation table, the Initial Error is the percent of tokens incorrect (different from gold standard) in the original input OCR, while Final Error is the total percent incorrect in the output. Annotator Task is the percent of the corpus that the annotator must give a judgement for. Introduced Error is the percentage of tokens in the corpus for which our automated system erroneously changes a form that was actually correct (matched the gold standard) in the original input. Tokens Successfully Corrected is the percentage of tokens that are real errors in the original input but have been corrected to match the gold standard, either by the automated system or by the human annotator.

Given the 2.2% annotator task and our estimate of 4 seconds per annotator judgement, the 2.7 million word Faroese corpus therefore requires about 65 annotator hours for full correction using the current version of our toolkit, compared with an estimated 2700 hours for manual editing without the toolkit.

Initial Error	7.6%
Final Error	**2.4%**
Introduced Error	0.6%
Tokens Successfully Corrected	5.7%

Table 4: Operation of character model with dictionary and heuristic handling but no human annotation. All quantities are expressed by percentages of corpus tokens (words).

Even without human annotation, applying dictionary-informed heuristic conditional filtering to the *k*-best candidates performs much better than the plain dictionary-free HMM (final error 5.4%, see Table 2); Table 4 shows projections for this combination in direct comparison to Table 3.

4.2. Ablation analysis

Table 5 systematically illustrates the tradeoffs between annotator expense and corpus quality, for this Faroese corpus. The progression starts from the best fully automated annotator-free system (the same as in Table 4) as Stage 0, and builds at each stage by replacing the automated decision strategy of one more of the system's logically separable heuristic cases with a human annotator until reaching full human annotation in Stage 11. The ordering of stages in the ablation is somewhat arbitrary; we have attempted to give some illustrations of sensible choices at different annotation budgets, as well as pass through the system we actually use. Our chosen system (that of Table 3) is bolded; this reflects our target of around 2% annotator task.

The heuristic cases defining the ablation steps are those described in §3.4, created from logical combinations of four binary-valued attributes: E, whether the top-ranked generated candidate's form is Equal to the original input token; O, whether the Original token is in the dictionary; B, whether the Best generated candidate is in the dictionary; and K, whether any generated candidates K1-4 are in the dictionary. These attributes are listed under Switched in Table 5 to specify which heuristic has been switched to human annotation at each stage; their true/false values are coded +/-. Table 5 has 11 separable steps progressing from Stage 0, rather than only the 9 heuristic cases described in §3.4, because in our best-performing implementation two of the cases are further split by a threshold to conditionally send only some tokens to a human annotator. These split cases (using thresholds TH3.1 and TH4.1 in §3.4.1 above) are marked in the table alongside the EOBK attribute coding when a set of tokens falling over or under their threshold is switched to human annotation.

Stage	Switched	Annotated	Final error
Stage 0		0%	2.41%
Stage 1	E+O-B-K+>TH3.1	0.42%	2.15%
Stage 2	E-O-B-K-	1.13%	1.68%
Stage 3	E+O-B-K-<TH4.1	**2.23%**	**1.34%**
Stage 4	E-O+B+K+	2.50%	1.27%
Stage 5	E-O-B-K+	2.90%	1.18%
Stage 6	E+O-B-K+<TH3.1	5.77%	0.79%
Stage 7	E+O-B-K->TH4.1	7.46%	0.57%
Stage 8	E-O-B+K+	10.27%	0.36%
Stage 9	E-O+B-K+	10.49%	0.35%
Stage 10	E-O+B-K-	10.55%	0.34%
Stage 11	E+O+B+K+	100%	0%

Table 5: Ablation analysis for amount of human annotation. All quantities are expressed by percentages of corpus tokens (words). The Annotated column lists how much of the corpus must be annotated by a human. The Switched column indicates which heuristic case has just been switched to human annotation at the current stage.

This exact ablation curve is specific to the unique corpus and initial heuristic settings it comes from, but it clearly illustrates a general principle for combining human annotation with an automated system: an annotator is effectively used in cases where they can annotate a manageable chunk of the data and have a beneficial impact with most of those annotations. Our system's automated generation of candidate corrections and heuristic case-based handling for them creates a structure that allows for an efficient use of the annotator's effort, and flexibility in decision-making depending on the annotation budget and corpus quality requirements.

4.3. Reduced character model training data

As described in §3.2, the character decoder that generates output forms is trained on 170,000 words of parallel text. Since this much parallel text could be expensive to produce if it is not already available independently, Table 6 shows the consequences of using character models trained on significantly less parallel training data, to observe the effect on system quality and usability. The first row is reproduced from Table 3, while results for the other two character decoders are reported under identical heuristic settings.

Training Data	Corrected	Added Error	Annotated	Final Error
170k	6.6%	0.4%	2.2%	1.3%
100k	6.6%	0.5%	2.5%	1.4%
50k	6.3%	0.4%	2.6%	1.6%

Table 6: Performance when reducing character decoder training data: percent of true errors that are successfully corrected; percent of error introduced to originally correct tokens; percent of tokens judged by human annotator; total remaining error after processing and annotation.

Overall the character decoding component of our system appears fairly robust to major reduction in training data. Although some negative effect of the reduction is certainly evident, especially in the training set of 50,000 words, performance seems to degrade slowly.

The training size of 100k words appears to have a little more introduced error, and a higher workload for the annotator, when tested with the same heuristic settings that we found to be best when using full-size training data. Some of the increase in annotator task could be because the thresholds that control which tokens are annotated were set on a rather different dev set, and may not be well calibrated for this character model's probabilities. It would be possible to recalibrate them to reduce the annotation task, in exchange for raising the final error rate. However, another part of the increased annotator task is simply that more tokens overall fall into the annotator-directed heuristic cases. This is essentially because when this slightly degraded character decoder now fails to generate the right form (that is usually in the dictionary), it often instead generates forms that are not in the dictionary, and it is mainly cases lacking dictionary membership that fall to the annotator (§3.4.1). The summary effect of the 100k training set is perhaps to shift the system's entire annotation ablation curve (Table 5) to

a slightly more expensive place: very good quality is still achievable with a bit more annotator time, or if annotator time is strictly limited then quality will be slightly worse than if using the 170k training set.
With 50k words of training data, there is a reduced ability to correct some of the errors in the corpus, and the annotator task has increased further due to unsuccessful generated candidates that are not dictionary members. With the most expensive annotation requirement and yet the worst result (final error), training on only 50k words clearly hurts system performance. However, it does still significantly improve the corpus without extremely unreasonable annotator effort, reducing error from an initial rate of 7.6% (Table 3), and therefore may be acceptable to some users.

4.4. Reduced heuristic dev set data

In §4.1 we used dev sets of 10,000 words of parallel text to set optimal decisions for the heuristic cases. Here, we explore the outcome of using smaller dev sets to set heuristic decisions, while still testing these decisions on 10,000 words to obtain reliable comparisons to the test sets of §4.1. Table 7 shows the projected outcomes of using heuristics derived from a variety of smaller dev sets.

Dev Size (words)	Annotated	Final Error
8,000	2.2%	1.3%
5,000	2.2%	1.3%
3,000	1.8%	1.5%
2,000	2.5%	1.6%

Table 7: Impact of smaller dev sets on annotator workload and final error.

Table 7 illustrates sample results only, giving an example from one dev set at each of the listed sizes; this shows what kind of outcome is possible but does not necessarily reflect average expected performance for dev sets of that size.
For the dev sets of 8,000 and 5,000 words, the optimal heuristics were identical to those found in §4.1. Therefore, there is no change in projected results. This gives further assurance that our current system is based on adequate dev data, and may be reasonably expected to generalise well to the rest of the corpus when it is corrected. This result also indicates that if the amount of parallel text were very limited, the dev set could be fairly small, under 10,000 words for this corpus.
With a 3,000-word dev set, there is one difference in the apparently optimal settings; human annotation is not used in one case, because the somewhat sparse information in this dev data was insufficient to clearly identify all good opportunities to take advantage of annotation. The result of these settings misses the mark a bit - it does not fully use our intended annotator budget of 2%, and so has a worse final error rate than might otherwise have been achieved - but it is still probably a decent heuristic setting at a different target point on an annotation/error tradeoff curve (Table 5).
Using 2,000 words, 4 of the 9 heuristic cases have different settings than in the original system. The damage to results is not catastrophic; changes to low-frequency cases (among those most affected by sparse data) have limited impact. Still, with an increase in both annotator effort and final error, the lack of sufficient data in the 2,000-word dev set is clearly harmful. Using a statistically misleading (too small to be representative) dev set has a risk of setting misfit parameters that waste annotator effort without achieving quality worth the effort. The estimates of how much of the corpus will be annotated also become less accurate when they are estimated on smaller dev sets, resulting in logistical trouble from more annotation burden than expected, or suboptimal corpus quality if not using the full annotation budget.

5. Discussion

5.1. Previous Work

There is limited precedent for the problem of post-processing OCR output to a high quality from physically degraded input in low-resource languages. Doush and Al-Trad (2016) use both Microsoft Word and Google's spelling suggestions to moderately improve OCR error in Arabic. Afli et al. (2016) use machine translation to repair French OCR; the translation system is trained on 90 million words of parallel text that is all hand corrected by 2-3 annotators, and achieves 1.7% word error rate. Kolak and Resnik (2002; 2005) use HMM-based post-processing like ours that is applicable to low resource languages, but human annotation is not incorporated and error rates remain too high for our goals with the Faroese corpus. More generally, there are a variety of noisy channel model analogues to our scenario, in OCR itself (Natarajan et al., 1999; Natarajan et al., 2009; Shlien and Kubotal, 1986), transliteration (Jiampojamarn et al., 2007), machine translation of closely related languages (Pourdamghani and Knight, 2017), multilingual part of speech tagging (Duong et al., 2014), spell checking (Li et al., 2012), and speech recognition (Gales and Young, 2008), among others. These applications all tend to differ from ours in that they use much more training data, or are satisfied with higher output error rates in tasks ranging from much harder than ours to rather similar.
Our toolkit is situated in the paradigm of human-assisted NLP. Combining statistical NLP with human judgement, supervision, or correction is common in preparation of small but valuable high-quality resources for theoretical linguistics research, such as annotating the Icelandic corpus of Rögnvaldsson et al. (2012). The methodology used by Ingason et al. (2014) shows the current state of Faroese NLP for historical corpora; they used a bilingual (Icelandic) approach to NLP resource development, together with time-consuming manual correction to parse a Faroese corpus of 50,000 words. With a larger corpus and simpler task of OCR correction, we have placed relatively greater emphasis on reducing human annotator time. An alternative human-assisted system developed for correcting OCR error in Yupik (Schwartz and Chen, 2017) may be preferable to our method in some use cases, although it would not be able to effectively correct our Faroese corpus. The method of Schwartz and Chen (2017) takes advantage of systematic restrictions on Yupik syllable structure to identify words containing OCR error (violating legal syllable shapes), and flag them for annotator attention. This does not require a dictionary or parallel training text; therefore, unlike our

method, it could be used in extremely low-resource settings with no expected performance loss. However, the approach of Schwartz and Chen (2017) critically depends on the typology of the language it is applied to, and whether that language has such a restricted set of permitted syllable structures (detectable from the orthography) that most OCR errors will violate them. Faroese, unlike Yupik, permits a fairly large number of syllable shapes; many OCR errors (such as *nógu* which should be *nógv*) create locally reasonable sequences that happen to not result in real whole words of Faroese, so that a dictionary provides a more useful signal here. Additionally, the system for Yupik does not propose candidate corrections for the errors it identifies, so the human annotator must address all of them. This would require several times more annotator effort than we intended for the Faroese corpus: 7.6% of tokens assuming perfect error detection, with more effort per token than expressing a judgement in one keystroke. The language resources (parallel text and dictionary) that our method requires enable it to generate corrections, and implement many of them with no human attention, to use annotator time efficiently in large corpora.

5.2. General Application for OCR Correction

Any use of our toolkit for OCR post-processing must find a balance between three types of resources: parallel training text for character error models and heuristic estimation, a dictionary or wordlist to help filter decoding output, and human annotation to adjudicate between the character model and dictionary. The Faroese corpus we correct has a good amount of parallel data for training and development, and a large high-quality dictionary; our goal was to bring the corpus to a very low error rate with only a little annotator effort. Either of our major language resources may not be accessible for other corpora our toolkit is used with, the initial error rate of corpora will vary, and corpus developers' ultimate goals and priorities may differ from ours. We intend for our illustrative analyses to provide prospective users with at least a first glimpse at how applicable our tools might be for their problem, given their particular available language resources, amount of annotator time, and size of corpus the annotator's time must scale to.

The supplementary evaluations exploring how robust the toolkit is to artificial limitations in resource availability showed that, although we had around 250k words of Faroese parallel text available for training and development (produced by human editors correcting portions of the Faroese corpus for various purposes over a period of years), it is still possible to substantially reduce OCR error in the corpus with under half or even one quarter this amount of parallel text. Performance is clearly best when more training data is available, but when it must be limited (if no corrected parallel text already exists), our toolkit can still be expected to improve the condition of a corpus with a fairly reasonable annotator workload.

The Faroese dictionary is from an open-source spell checking project; although Faroese is spoken by a small population, it is a living language whose speakers have one of the highest rates of computer/internet usage globally. The strategy of §3.4 relies heavily on a dictionary, and while this enjoys evident success with the large dictionary we use for Table 3, it may be a vulnerability if correction quality degrades rapidly with reduced dictionary coverage. We recognise that many extremely low-resource languages will not have comparable dictionaries readily available. Alternate sources may include scraping Wikipedia or other websites, smaller wordlists created for any purpose, and the set of words in the training data. Requirements for useful dictionary size will also vary by language typology. In cases of relatively low dictionary coverage, we would suggest an iterative approach, building a larger dictionary with annotator-approved forms as corpus correction proceeds. Our method appears generally well suited to a bootstrapping approach, to progressively build up both the dictionary and a larger bank of parallel training/development text when needed: the initial cost to start up can be reduced because the method has at least reasonable performance with relatively lower amounts of input resources, and the subideal system can then be used to fairly quickly produce more corrected (parallel) text and augment the dictionary, creating the resources necessary to more accurately and efficiently correct large corpora.

5.3. Aligner Uses

Independently of our toolkit as a whole, the aligner can be broadly useful. It produces quality word-aligned data from parallel source texts where token count, tokenisation, line breaks, etc. may not agree. Texts produced by human editing will often not align straightforwardly with their sources; on the other hand, constraining editors to strictly preserve the token structure of input files in their corrections would increase their workload while degrading the natural language quality of the edited text.

Our aligner resolves this tension, and it may be helpful in a variety of noisy channel model NLP tasks (e.g. transliteration (Jiampojamarn et al., 2007)), or any setting with parallel text where at least one side has been produced by a human editor. Because the aligner algorithm is not dependent on external resources like a dictionary or human annotator, it can be applied to new corpora with confidence.

6. Acknowledgements

We thank Caroline Heycock, Tony Kroch, and the faculty of Føroyamálsdeildin for their support in the development of Faroese corpus resources; and Marine Carpuat and Sharon Goldwater for their technical insight. This work was supported by NSF grant DGE-1321851, and by DARPA Lorelei under Cooperative Agreement No. HR0011-15-2-0023.

7. Bibliographical References

Afli, H., Barrault, L., and Schwenk, H. (2016). OCR Error Correction Using Statistical Machine Translation. *International Journal of. Computational Linguistics and Applications*, 7(1):175–191.

Doush, I. A. and Al-Trad, A. M. (2016). Improving post-processing optical character recognition documents with arabic language using spelling error detection and correction. *International Journal of Reasoning-based Intelligent Systems*, 8(3-4):91–103.

Duong, L., Cohn, T., Verspoor, K., Bird, S., and Cook, P. (2014). What Can We Get From 1000 Tokens? A Case Study of Multilingual POS Tagging For Resource-Poor Languages. In *Proceedings of EMNLP*, pages 886–897. Association for Computational Linguistics.

Galbraith, D. (2016). Faroese ballad meter: a constraint-based account. In *NordMetrik: Versification, Metrics in Practice*, 5/25/16 - 5/27/16. University of Helsinki.

Gales, M. and Young, S. (2008). The application of hidden markov models in speech recognition. *Foundations and trends in signal processing*, 1(3):195–304.

Heycock, C. and Wallenberg, J. (2013). How variational acquisition drives syntactic change: The loss of verb movement in Scandinavian. *Journal of Comparative Germanic Linguistics*, 16(127).

Ingason, A. K., Loftsson, H., Rögnvaldsson, E., Sigurðsson, E. F., and Wallenberg, J. C. (2014). Rapid Deployment of Phrase Structure Parsing for Related Languages: A Case Study of Insular Scandinavian. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 91–95, Reykjavik, Iceland, May. European Language Resource Association (ELRA).

Jiampojamarn, S., Kondrak, G., and Sherif, T. (2007). Applying many-to-many alignments and hidden markov models to letter-to-phoneme conversion. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 372–379, Rochester, New York, April. Association for Computational Linguistics.

Kleinberg, J. and Tardos, É., (2006). *Algorithm design*, chapter 6.6 Dynamic programming: Sequence alignment. Pearson Education.

Kolak, O. and Resnik, P. (2002). OCR error correction using a noisy channel model. In *Proceedings of the second international conference on Human Language Technology Research*, pages 257–262. Morgan Kaufmann Publishers Inc.

Kolak, O. and Resnik, P. (2005). OCR post-processing for low density languages. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 867–874. Association for Computational Linguistics.

Li, Y., Duan, H., and Zhai, C. (2012). A generalized hidden markov model with discriminative training for query spelling correction. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 611–620. ACM.

Natarajan, P., Bazzi, I., Lu, Z., Makhoul, J., and Scwhartz, R. (1999). Robust OCR of degraded documents. In *Document Analysis and Recognition, 1999. ICDAR'99. Proceedings of the Fifth International Conference on*, pages 357–361. IEEE.

Natarajan, P., MacRostie, E., and Decerbo, M. (2009). The BBN byblos Hindi OCR system. In *Guide to OCR for Indic Scripts*, pages 173–180. Springer.

Pourdamghani, N. and Knight, K. (2017). Deciphering related languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2503–2508.

Rögnvaldsson, E., Ingason, A. K., Sigurðsson, E. F., and Wallenberg, J. (2012). The Icelandic Parsed Historical Corpus (IcePaHC). In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1977–1984, Istanbul, Turkey, May. European Language Resource Association (ELRA).

Schwartz, L. and Chen, E. (2017). Liinnaqumalghiit: A web-based tool for addressing orthographic transparency in St. Lawrence Island/Central Siberian Yupik. *Language Documentation & Conservation*, 11:275–288.

Shlien, S. and Kubotal, K. (1986). Optical character recognition of touching characters. In *Graphics Interface 1986*, pages 390–395. Canadian Information Processing Society.

Thráinsson, H., Petersen, H. P., Jacobsen, J. í. L., and Hansen, Z. S. (2012). *Faroese: An overview and reference grammar*. Føroya Fróðskaparfelag, Tórshavn.

8. Language Resource References

Føroyamálsdeildin. (n.d.). *Bandasavnið*. Fróðskaparsetur Føroya ('The tape archive, of the Faroese Language faculty, University of the Faroe Islands'), https://setur.fo/gransking/soevn/bandasavn/.

J. S. Andersen. (2001). *Føroyska orðalistin til rættlestur*. The Faroese Linux User Group, http://fo.speling.org.

Introducing the CLARIN Knowledge Centre for Linguistic Diversity and Language Documentation

Hanna Hedeland, Timm Lehmberg, Felix Rau, Sophie Salffner, Mandana Seyfeddinipur, Andreas Witt
HZSK/Universität Hamburg, INEL/Universität Hamburg, IfL/Universität zu Köln,
ELAR&SWLI/SOAS University of London, IfDH/Universität zu Köln
{hanna.hedeland, timm.lehmberg}@uni-hamburg.de
{f.rau, andreas.witt}@uni-koeln.de
elararchive@soas.ac.uk

Abstract

The European digital research infrastructure CLARIN (Common Language Resources and Technology Infrastructure) is building a Knowledge Sharing Infrastructure (KSI) to ensure that existing knowledge and expertise is easily available both for the CLARIN community and for the humanities research communities for which CLARIN is being developed. Within the Knowledge Sharing Infrastructure, so called Knowledge Centres comprise one or more physical institutions with particular expertise in certain areas and are committed to providing their expertise in the form of reliable knowledge-sharing services. In this paper, we present the ninth K Centre – the CLARIN Knowledge Centre for Linguistic Diversity and Language Documentation (CKLD) – and the expertise and services provided by the member institutions at the Universities of London (ELAR/SWLI), Cologne (DCH/IfDH/IfL) and Hamburg (HZSK/INEL). The centre offers information on current best practices, available resources and tools, and gives advice on technological and methodological matters for researchers working within relevant fields.

Keywords: CLARIN Knowledge Sharing Infrastructure, linguistic diversity, language documentation

1. Introduction

The European digital research infrastructure CLARIN (Common Language Resources and Technology Infrastructure)[1] is not restricted to interconnected technical centres and service units; CLARIN is also actively building a Knowledge Sharing Infrastructure (KSI) to ensure that existing knowledge and expertise in individual centres all across Europe (and beyond) is easily available both for the CLARIN community and for the humanities research communities for which CLARIN is being developed. Within the Knowledge Sharing Infrastructure[2], so called Knowledge Centres[3], or K Centres, play an important role. The Knowledge Centres comprise one or more physical institutions with particular expertise in certain areas and are committed to providing their expertise in the form of reliable knowledge-sharing services.

In this paper, we present the ninth K Centre: CLARIN Knowledge Centre for Linguistic Diversity and Language Documentation (CKLD)[4], certified in September 2017. The CKLD is a virtual distributed centre comprising institutions at the Universities of London (ELAR/SWLI), Cologne (DCH/IfDH/IfL) and Hamburg (HZSK/INEL). The mission of the CKLD is to establish a single interface to reliably provide the expertise on various aspects of the thematic focus brought in by the founding members. Section 3. thus describes the areas of competence and specific services of the individual partners.

The knowledge-sharing services provided by the CKLD are the topic of section 4.: The center offers information on current best practices, available resources and tools, and gives advice in technological and methodological matters for researchers working within relevant fields. As a member of the European digital research infrastructure CLARIN, the aim of the CKLD is also to allow for a better integration of these research communities and the resources and tools traditionally used into existing and emerging digital research infrastructures. This also includes further coordinated work on relevant best practices.

Language documentation and research in linguistic diversity share a joint focus on less-widely studied languages. These languages and their varieties are often minority languages and frequently endangered. Further aspects relevant to most of these linguistic settings are multilingualism and language contact, since usually more than one (non-standard) language or variety interact with at least one other. The main objective of language documentation is an annotated multimodal audio-visual corpus as a comprehensive record of the linguistic practices characteristic of a given speech community. Research in linguistic diversity on the other hand comprises descriptive studies and analyses of individual under-resourced languages and comparative typological linguistics that takes into account the world-wide diversity of human languages.

While research questions and methods may differ, both disciplines share many methods, tools, and challenges. The documentation and analysis of smaller, lesser known and endangered languages around the world and related research – typically in a field-work setting – requires technical knowledge and specialized skills. The required expertise comprises recording of speech events, metadata man-

[1] https://www.clarin.eu/

[2] https://www.clarin.eu/content/knowledge-sharing

[3] https://www.clarin.eu/content/knowledge-centres

[4] http://ckld.uni-koeln.de

agement, data handling, (semi)-automatic analysis, documentary and typological corpus and database compilation, archiving of results, among others. Typological research in language diversity often requires additional layers of annotations and the application of search and clustering algorithms to typological corpora and databases.

2. Building a Distributed Centre

Building a sustainable distributed support and knowledge infrastructure for linguistic diversity and language documentation is an organisational and academic challenge. Linguistic diversity research and language documentation are by definition global endeavours and at the same time highly localized. Methodologically, it includes the classical issues of linguistic fieldwork as well as the application of the full range of language-related technologies from the areas of speech technology and digital humanities in well-equipped academic settings and remote field sites. The CKLD approaches this challenge by combining the expertise of language typologists, field linguists, sociolinguists, computer linguists, computer scientists, data curators as well as language archivists from institutions in several geographic locations into a single digital institution. The geographical distribution and variation in object languages and research paradigms of research projects is often an obstacle when it comes to creating synergies and common solutions to similar problems. It is therefore more than desirable that joint initiatives with a technical/infrastructural focus bring together researchers from different disciplines under one subject.

3. Participating Institutions

The CKLD is a joint endeavour of three main founding members located at SOAS University of London (ELAR/SWLI), the University of Hamburg (HZSK/INEL), and the University of Cologne (DCH/IfDH/IfL). Since the founding of the centre, the newly established Leibniz-Centre General Linguistics (ZAS) at the Humboldt University in Berlin has recently joined as a full member, and further partners with suitable profiles are likely to become a part of the centre in the future.

3.1. London (ELAR/SWLI)

The Endangered Languages Archive (ELAR) and the SOAS World Languages Institute (SWLI) provide training in language documentation, data collection, data management and archiving. Training has been a core activity of ELAR and SWLI since their inception in 2002 and 2016 respectively.

The Endangered Languages Archive (ELAR) is a digital repository specialising in preserving and publishing endangered language documentation materials from around the world. ELAR currently holds multimedia collections of endangered languages worldwide (more than 420 languages), with regional strongholds in Africa, Middle East, Asia, Australia, Oceania and Meso- and South America. The collections can be browsed and accessed through the ELAR online catalogue[5]. All materials are digital, free to access, openly available (after free registration) and have a worldwide coverage. ELAR is seen as a reputable and reliable repository and as one of the leading language and culture archives worldwide, being approached by more and more external projects (e.g. American NSF- funded projects, EU projects) within and beyond linguistics as a host repository now that data depositing is seen as an integral part of good practice research. ELAR has a small team of highly specialised archivists, data scientists, audiovisual experts, and linguists, pioneering in the areas of user interface, managed access and accessibility. The archive was originally funded by Arcadia and is since 2014 part of the National Research Library of SOAS University of London.

Starting out 100 years ago with teaching languages of Asia, SOAS now combines language scholarship and disciplinary expertise with a regional focus. It has the largest concentration in Europe of academic staff concerned with Africa, Asia and the Middle East and offers unparalleled expertise in a wide range of non-European languages, e.g. Somali, Khmer or Syriac. Building on SOAS's tradition of non-European language teaching and research, the SOAS World Languages Institute (SWLI) was created in 2016, bring together a wide range of affiliated scholars and research students from across SOAS to make the expertise of its members more widely available to the wider world. A further focus of the Institute is the digital humanities. The institute supports researchers in planning projects with digital components and aims to strengthen SOAS scholars' expertise in the methods of the digital humanities. The SWLI runs various events throughout the year, including talks, seminars, workshops, conferences and round table discussions, as well as book launches and film screenings. It provides bespoke training programmes related to languages and language-based research. The specialist courses can be tailor-made for individuals and groups of different sizes from a range of different backgrounds – including the public and private sector, business, charities, government and NGOs – and include topic such as Language Documentation: theory and methods, annotation software for audio and video data, video recording for scientific analysis, or data management. The SWLI partners with several institutions and projects around the world, as for instance Docip (a Swiss Foundation based in Geneva and Brussels which documents and provides access to records of the UN representation of indigenous peoples from all around the world before international organisations) and Memrise (a language learning company based in East London). At the moment SWLI is part of the UK CLARIN-Consortium and aims at being a C- and possibly a B-Centre.

3.2. Hamburg (HZSK/INEL)

The members of the CKLD within the University of Hamburg are the Hamburg Centre for Language Corpora (Hamburger Zentrum fur Sprachkorpora, HZSK), which is a certified CLARIN B Centre with a thematic focus on spoken and multilingual corpora, and the long-term (18-year) project INEL (Grammatical Descriptions, Corpora, and Language Technology for Indigenous Northern Eurasian Languages), which builds on the existing HZSK infrastructure while creating and making available more than ten cor-

[5] https://elar.soas.ac.uk/

pora for endangered and/or lesser documented languages from the Uralic and Altaic language families.

The HZSK was launched in 2011 to create a sustainable institutional framework for the software, corpora and expertise developed within the research data managing project of the Special Research Centre on Multilingualism in Hamburg during twelve years of funding.

The software suite EXMARaLDA (Schmidt and Wörner, 2014) for spoken corpora now developed jointly by the HZSK and the Institute for the German Language (IDS)[6] includes a transcription and annotation editor (Partitur-Editor), a corpus and metadata management tool (Coma) and a corpus search and analysis tool (EXAKT). The CLARIN compatible EXMARaLDA framework allows for import and export of most common transcription formats and for multimedia visualisation of transcription data in various layouts. Parts of the framework are also integrated as web services into the Data Seal of Approval (DSA)[7] certified HZSK Repository[8](Jettka and Stein, 2014), through which the corpora hosted at the HZSK are made available to the academic community.

The corpora at the HZSK (Hedeland et al., 2014) traditionally cover various aspects of individual and societal multilingualism, but also document lesser known regional varieties, such as Porteño Spanish or Faroese Danish, or endangered languages such as Nganasan (Wagner-Nagy and Szeverényi, 2015). For the transparent and efficient curation of corpora, several customized workflows have been developed and implemented at the HZSK using the open source distributed version control system Git[9] and the open source project management software Redmine[10].

The expertise gathered at the HZSK is provided as consulting, support and training on best practices, standards and tools for researchers and research projects working with corpora. The HZSK also developed and implemented the CLARIN-D Helpdesk(Lehmberg, 2015), which is now managed by the HZSK and used to coordinate user support for CLARIN-D centres and CLARIN services.

The INEL project (Hedeland et al., 2016), which started in February 2016, is organized jointly in the framework of the Academies' program of the Federal Republic of Germany by the Union of the German Academies of Sciences and Humanities and the University of Hamburg and institutionally located at the Institute for Finno-ugric/Uralic Languages (IFUU).

Since INEL is making use of the HZSK software and infrastructure, several components are being adapted and extended to meet the needs of the linguistic subprojects, in which richly annotated corpora based on existing archival material and partly supplementing new recordings are being created and made available to the academic community. The extensions include interoperability with commonly used tools such as FLEx[11] and the definition of streamlined collaborative Git-based workflows for the curation of archival materials and the publication of corpora of glossed data with rich sociolinguistic metadata. The INEL project also provides specific expertise on fieldwork and curation of archival materials for indigenous languages.

The synergetic alliance of the HZSK as an infrastructure unit and INEL as a long term project with their particular expertise in a broad spectrum of aspects in linguistic diversity and language documentation enables both of them to provide solid support for the linguistic community in these overlapping areas.

3.3. Cologne (DCH/IfDH/IfL)

The Department of Linguistics (IfL), the Data Center for the Humanities (DCH), and the Digital Humanities Department (IfDH) of the University of Cologne are partners in CKLD. The Department of Linguistic is a center for research and teaching in the areas of language documentation, methodologies of language documentation, and language diversity (Himmelmann, 2008; Himmelmann, 2012). It has been active in the documentation of endangered languages since the very beginning of the field in the early 1990s (Himmelmann, 1998). Since then, numerous research projects with a focus on language documentation were and are based in Cologne. Language documentation and language diversity are the core of the linguistics curriculum in Cologne. Most of the teaching and research staff of the general linguistics department are working on either linguistic diversity or language documentation or both and the department offers several courses related to theory and methodology of language documentation and linguistic diversity research every year. The Data Centre for the Humanities (DCH) is a facility of the faculty of arts and humanities. The DCH provides methodological and technical support to research projects and provides services in the areas of data management, curation and archival storage (Sahle and Kronenwett, 2013). DCH and IfL have developed software for language documentation, archiving[12] and analysis. (Blumtritt et al., 2013) The Language Archive Cologne[13] is a joint endeavour of the linguistics department and the DCH. Together with the linguistics department and other partners, the DCH is extending the Language Archive Cologne in the KA$\hat{3}$ project (Cologne Center Analysis and Archiving of Audio-Visual Data)[14] funded by the German Ministry of Research (BMBF). The newly established Digital Humanities Department features one chair explicitly dedicated to digital linguistics. The Digital Humanities Department will also contribute expertise and staff to the K-Centre.

4. Services

Building and running a distributed knowledge centre for linguistic diversity and language documentation requires a flexible and reliable support and knowledge-sharing infrastructure. The services offered by the knowledge centre consist of a web presence providing information on the relevant

[6] http://www.ids-mannheim.de

[7] https://www.datasealofapproval.org/

[8] https://corpora.uni-hamburg.de/repository

[9] https://git-scm.com/

[10] www.redmine.org/

[11] https://software.sil.org/fieldworks/

[12] http://cmdi-maker.uni-koeln.de

[13] https://lac.uni-koeln.de/en/

[14] http://dch.phil-fak.uni-koeln.de/ka3.html

topics, a helpdesk allowing researcher to interact directly with the centre, as well as training courses offered on a regular basis.

4.1. The Website as a Portal to the Centre

The CKLD website[15] operated by the University of Cologne is the main portal to the Knowledge Centre. The site offers a description of the centre and a list and explanation of its services. Besides providing a collection of informational material and further relevant links to software tools, language resources, manuals, and documentation of best practices, the website is also the entrance point into the interaction with the CKLD Helpdesk.

4.2. Support through the CKLD Helpdesk

The CKLD Helpdesk is integrated into the CLARIN-D Helpdesk operated by the HZSK in Hamburg. The use of a ticketing system (OTRS)[16] and proven support workflows allows for efficient handling of incoming inquiries. For each question, an automatic confirmation mail is sent to the user. Within two working days, questions will either be answered directly by the first-line support at the HZSK, or, if needed, forwarded to experts within the CKLD while keeping the user updated on the status of the inquiry. The incoming inquiries also provide the CKLD with valuable information on which topics need to be covered more in detail on the website or in dedicated trainings. The CKLD Helpdesk provides information and guidance to researchers in the preparation and during the execution of language documentation and other field-work based linguistic research projects as well as typological research. This includes questions relating to equipment, digital tools, methods, where to find data and information, whom to contact for specialist information on particular regions or language families.

4.3. Teaching and Training Modules

The participating institutions offer courses in language documentation, corpus creation, data analysis and research data management. ELAR and SWLI are one of the leading providers in trainings in theory and method in modern language documentation. The HZSK regularly offers trainings for the EXMARaLDA software or project specific trainings on research data management. With the distributed expertise, the CKLD can offer various training modules on topics such as research data management, publication and sustainability, including ethical and legal issues, but also hands-on trainings that provide researchers and students with the practical knowledge required to actually create high quality language resources. The partners coordinate their training activity and communicate their activities via the CKLD website.

5. Outlook

The newly established Knowledge Centre CKLD aims to act as a contact point for all researchers in the fields of linguistic diversity and language documentation. While working to provide researchers and students with relevant information, support and trainings, the CKLD is also a cooperation between centres in different geographical locations, which also on the level of object languages focus on different geographical areas respectively, but which can align their work on best practices for the creation and publication of language resources to jointly work towards better interoperability and integration into existing and emerging international digital research infrastructures.

[15] http://ckld.uni-koeln.de

[16] https://www.otrs.com/

6. Bibliographical References

Blumtritt, J., Bouda, P., and Rau, F. (2013). Poio API and GraF-XML: A radical stand-off approach in language documentation and language typology. In *Balisage: The Markup Conference 2013, Montréal, Canada, August 6 - 9, 2013. In Proceedings of Balisage: The Markup Conference 2013*.

Hedeland, H., Lehmberg, T., Schmidt, T., and Wörner, K. (2014). Multilingual corpora at the hamburg centre for language corpora. In Michael Haugh, et al., editors, *Best Practices for Speech Corpora in Linguistic Research*, pages pp. 208–224. Cambridge Scholars Publishing.

Hedeland, H., Lehmberg, T., and Wagner-Nagy, B. (2016). Digitale workflows in langzeitprojekten am beispiel einer infrastruktur zur dokumentation indigener nordeurpäischer sprachen (inel). In *DHd 2016: Modellierung - Vernetzung - Visualisierung, (Leipzig, 7-12 March, 2016)*, pages 152–155, Leipizg. Digital Humanities im deutschprachigen Raum e.V.

Himmelmann, N. P. (1998). Documentary and descriptive linguistics. *Linguistics*, 36(1):161–196.

Himmelmann, N. P. (2008). Reproduction and Preservation of Linguistic Knowledge: Linguistics' Response to Language Endangerment. *Annual Review of Anthropology*, 37(1):337–350.

Himmelmann, N. P. (2012). Linguistic Data Types and the Interface between Language Documentation and Description. *Language Documentation & Conservation*, (6):187–207.

Jettka, D. and Stein, D. (2014). The HZSK repository: Implementation, features, and use cases of a repository for spoken language corpora. *D-Lib Magazine*, 20(9/10).

Lehmberg, T. (2015). Wissenstransfer und Wissensressourcen: Support und Helpdesk in den Digital Humanities. In *Forschungsdaten in den Geisteswissenschaften (FORGE) 2015, (Hamburg, 5-18 September, 2015)*, pages 25–27, Hamburg.

Sahle, P. and Kronenwett, S. (2013). Jenseits der Daten. *Libreas*, (23).

Schmidt, T. and Wörner, K. (2014). Exmaralda. In Ulrike Gut Jacques Durand et al., editors, *Handbook on Corpus Phonology*, pages 402–419. Oxford University Press.

Wagner-Nagy, B. and Szeverényi, S. (2015). Linguistically annotated spoken nganasan corpus. *Tomsk Journal of Linguistics and Anthropology*, 2:25–33.

Low Resource Methods for Medieval Document Sections Analysis

Petra Galuščáková[ab] and Lucie Neužilová[c]
[a]UMIACS, University of Maryland
College Park, MD, USA
petra@umiacs.umd.edu
[c]Charles University, Faculty of Arts, Department of Auxiliary Sciences of History
Prague, Czech Republic

Abstract

This paper describes a small but unique digitized collection of medieval Latin charters. This collection consists of 57 charters of 7 types illustrating various purposes of issuance by the Royal Chancellery. Sections in these documents were manually annotated for deeper analysis of the structure of issued charters. This paper also describes two baseline methods for an automatic and semi-automatic analysis and detection of sections of diplomatic documents. The first method is based on an information retrieval paradigm, and the second one is an adaptation of Hidden Markov Models. Both methods were proposed to work with respect to a small amount of available train data. Even though these methods were specifically proposed to work with medieval Latin charters, they can be applied to any documents with partially repetitive character.

Keywords: medieval documents, document sections detection, information retrieval, Latin

1. Introduction

Medieval charters issued by the Royal Chancellery were composed with respect to their typical structure (Guyotjeannin et al., 1993). They often consisted of repetitive sections and phrases, what is frequent in diplomatic language. Type and section ordering depends on the document type, which is defined by its purpose, type of the issuer and period of issuance. The purpose of this work is to 1) create a digital collection of medieval diplomatic charters, 2) provide manual annotations of a common structure of these charters, and 3) propose general methods for automatic detection of sections of these charters. The proposed methods should be trained and tested on this specific collection. As the number of diplomatic charters created using a given structure is often very limited and the size of the available train set is thus restricted, we aim at the methods which use a small amount of manually annotated documents for the training. We expect that providing an annotated collection of medieval charters and tools for automatic or semi-automatic recognition of document sections and their classification will significantly facilitate reading and processing of charters. It will allow further deeper analysis of individual diplomatic sections of charters and provide tools which will enable history researchers to think about relations and classification of material instead of spending time with slow and monotonous work and thus speeds up the whole research process.

1.1. Medieval Manuscript Collections

Lately, there is a growing effort towards digitization of medieval manuscripts. A number of such collections are provided as online digital libraries, e.g. Manuscriptorium[1], Digital Scriptorium[2], Manuscripta.at[3], Syriaca[4], E-codices[5], Saramusik[6] and Monasterium[7]. Manuscriptorium is a project of Czech National library which provides access to manuscripts, incunabula, early printed books, maps and charters. Digital Scriptorium is a consortium of American libraries and museums which provides free online access to pre-modern manuscripts. Some of the archives are aimed at specific types of manuscripts, e.g. E-codices provides access to Swiss manuscripts, Syriaca provides and access to manuscripts in Syriac language and Saramusik provides access to Arabic music manuscripts. Closest to our interest is Monasterium which collects archives of medieval and early modern charters. Monasterium provides access to almost 200 collections and more than half a million of documents. Growing effort in the area of processing charters is also reflected by the creation of Charters Encoding Initiative (CEI) (Vogeler, 2010). This coding scheme evolved from a TEI encoding scheme (Burnard and Rahtz, 2013) which is standard for the representation of texts in digital form and it is widely used for encoding historical documents. CEI extension was proposed for charter encoding and it is broadly used in the Monasterium collections.

1.2. Document Sections Analysis

This paper presents technical, linguistic and diplomatic problems of automatic analysis of the structure of medieval charters. According to our best knowledge, similar research had not been conducted before. However, our research is connected to research on automatic detection of general document structure, for example, automatic analysis of the discourse structure of medical abstracts (Lin et al., 2006).

[b]Previously Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics

[1]http://www.manuscriptorium.com
[2]http://www.digital-scriptorium.org
[3]http://manuscripta.at
[4]http://www.syriaca.org/
[5]http://www.e-codices.unifr.ch/en
[6]http://www.saramusik.org
[7]http://monasterium.net

Hidden Markov Models and Support Vector Machines were used to detect the introduction, methods, results and conclusions sections. In contrast to our case, train data for this domain can be relatively easy to acquire and supervised methods can be well applied. Document structure can also be detected using various semantic-based segmentation methods and lexical chain-based algorithms.

The two most often used algorithms for semantic segmentation are *TextTiling* (Hearst, 1997) and *C99* (Choi, 2000). Both measure segment similarity by calculating the cosine distance between neighbouring segments. *C99* calculates the similarity between all sentence pairs using the cosine measure and identifies regions with high intra-similarities. *TextTiling* calculates the similarity for adjacent segments of predefined size and points with the lowest values are designated as boundaries. Lexical chain-based algorithms detect boundaries based on the fact that the number of lexically related words within one segment is typically higher than the number of related words between adjacent paragraphs. Repetition of the lexical items can be detected easily and this approach may be improved by using synonyms and subordinates. Morris and Hirst (1988) determine lexically close words from thesaurus, Nguyen et al. (2011) further utilize word collocations, Mohri et al. (2010) calculate co-occurrence statistics and Kozima (1993) estimates similarities for pairs of words and uses them to find a sequence of lexical cohesiveness. Ponte and Croft (1997) propose a method for detection of small segments which share few common words using Local Content Analysis. In contrast to these approaches, our approach is supervised, and in addition to segmentation of the text, we are also interested in detection of the section type.

2. Medieval Charters Collection

Documents created in the era of John the Blind, King of Bohemia (1310–1346) and Count of Luxembourg are used in this work. We work only with charters written in Latin due to its much more consistent orthography than vernacular languages. Medieval charters also have a high level of uniformity. The presence and absence of charter sections and the level of their expression are frequently in correspondence with the type of the charter and with its historical importance.

Fealty deeds of John the Blind (von Estgen et al., 2009) are mainly used in this work. Fealty deeds are representative, typically contain all common sections and were not influenced by the literary creativity of the author. They are also very common in the High Middle Age. The advantage of this particular charter edition is also available transcription rules and uniformed orthography. The charters from printed edition were scanned, automatically recognized by a scanner built-in OCR system and manually corrected. Manual corrections were especially necessary as the applied OCR system did not support Latin, though the quality of the scanned documents was reasonable. Finally, sections in these charters were manually marked using CEI encoding scheme. All annotations were done by a single annotator who was a doctoral student of medieval history with a knowledge of Latin.

In most of the cases, we followed the transcription rules of the mentioned edition, except some simple rules, as interchanging *u* and *v*, which used to form the same grapheme in medieval Latin. These rules helped us to unify different word variants and achieve higher data consistency. The list of applied rules is given in Table 1.

Original	Replaced
ae	e
oe	e
y	ii
j	i
ci + vowel	ti + vowel
uu + vowel	w + vowel
vowel + u	vowel + v
u + vowel	v + vowel

Table 1: A list of applied transcription rules.

The collection consists of 57 manually annotated documents in total. These documents were divided into train, held-out and test set. Apart from fealty deeds (*lenni slib*), the collection also contains a limited number of acquaintances (*kvitance*), debt reliefs (*zprosteni dluhu*), donation (*donace*), reformation (*polepseni*) and request (*zadost*)) charters. Some documents were not able to be well categorized. Collection statistics are tabulated in Table 2. All these types of documents are on the same level in the hierarchy of importance of documents issued by the Royal Chancellery. This diversity of the document types allows us to test our methods on a different type of charters.

The annotated collection was published together with section detection framework[8] and is available under Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

2.1. Charter Sections

Sections and phrases of diplomatic charters often have repetitive character, and similar phrases and sections occur in similar document types. However, this kind of stability does not involve always using the same words and word phrases. Meaning and the role of the charter is often unified, but it was on the author how to express this meaning. Especially in the Royal Chancellery, the forms were relatively stabilized. For different authors, chancellery locality and era, it is characteristic to use different forms and phrases. Thanks to automatic recognition of these sections, we can study more about the process of creating of charters and about the chancellery personnel.

Some of these sections and phrases were composed very practically using only a few necessary words. For example, *intitulatio*, where the issuer introduces himself, *publicatio*, where the issuer expresses his intention to issue the charter, or *corroboratio*, where the issuer announces means of the sealing. Other sections and phrases leave more space for creativity and literary art of the author. *Dispositio*, the main

[8] http://ufal.mff.cuni.cz/Medieval-Charter-Sections-Corpus

Collection Set	Fealty deeds	Acquaintance	Debt relief	Donation	Reformation	Request	None	Total
Train	15	1	0	0	1	0	6	23
Held-out	3	0	0	0	0	1	0	4
Test	18	6	3	3	0	0	0	30

Table 2: Document type statistics

section of the charter, expresses the primary meaning of the charter as a legal document which codifies a legal act of the issuer. *Narratio*, which express issuer's intention to do something, is frequently formulated even more freely and is thus more complicated to detect automatically.

These differences between sections are fundamental for our work. Since *publicatio*, *intitulatio* and *corroboratio* typically contain only a few words and have unified meaning, they can be easily recognized by information retrieval-based methods. However, *dispositio* and *narratio* are almost never composed in the same way and they are thus much harder to recognize. As the ordering of the sections is also often standardized, positional information in the text can be expected to be helpful for recognition of these sections.

Another problem is that not all the sections are contained in each charter. Presence and absence of the sections and phrases are in connection with a type of charter. Typically, all mentioned sections are present only in the most important charters. Documents in our collection, contain all the necessary phrases like *intitulatio*, *publicatio*, and *dispositio*. More freely formulated phrases such as *narratio* are present only in some of them.

Presence of these section types in our data is displayed in Table 3. We also show an average length of each section type in terms of words, which is often in correspondence with quality of automatic detection of the specific section.

2.2. Additional Metadata

Apart from manually marked sections, the documents in the train set also contain manually marked named entities. Documents contain 68 personal names (*persName*), 54 roles or positions in society, (*rolename*), 43 place names (*placename*), 66 single letter characters (*c*), 59 measure labels (*measure*), 23 date labels (*date*), and 4 expansion of an abbreviation (*expan*). All documents in all sets also contain short manually crafted abstract in Czech.

3. Charter Sections Detection

We experimented with two methods for automatic detection of charter sections. The first method is based on information retrieval and the second one uses Hidden Markov models. Both methods were adapted to be able to be trained using a small amount of data.

To further reduce language variance, we also experimented with lemmatized forms of words. Lemmatized word forms were created using the online version of the Lemlat lemmatizer (Ruffolo et al., 2017) for Latin. In addition to this, we also trained our models with and without manually marked named entities.

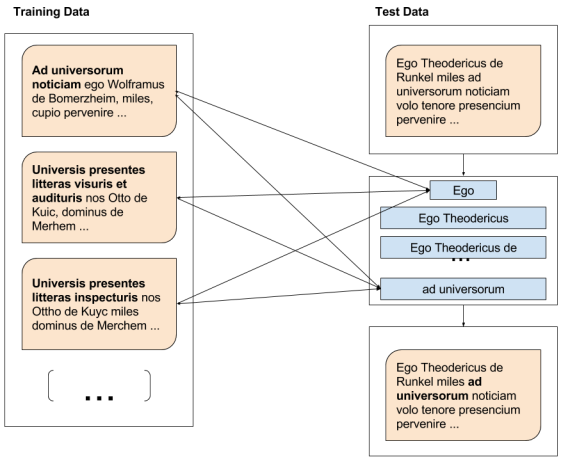


Figure 1: An overview of the information retrieval-based method. The scheme presents searching for the *intitulatio* section. Three examples of the *intitulatio* section are displayed in the train data. These are then compared with all the possible sections in the test data document. The section in the test document, which is the most similar to one of the sections in the train data is then marked as the *intitulatio* section.

3.1. Section Detection Approaches

Thanks to the reduced vocabulary of some sections, **information retrieval**-based approach is expected to work well on them. For each pre-defined section, we find a sequence of words in the test document which is the most similar to any phrase manually marked as this pre-defined section in the train data. A similarity is calculated between each sequence of words in the test document and each marked phrase in all train documents. Then, a sequence of words with a maximal cosine similarity is selected to belong to the pre-defined section (the *Cosine* method). We also used the TF IDF score in addition to the cosine similarity (the *TF IDF* method). Employing the cosine similarity and TF IDF also enables us to detect the most similar section in the annotated document collection, which can be very helpful for the history researchers. To further improve the precision, we also calculate a sum of distances between the sequence of words from the test document and three most similar phrases manually marked to belong to the pre-defined section (the *Cosine Max 3* method). Information retrieval-based method is explained in Figure 1.

Information retrieval-based approach is supposed to work exceptionally well for short and well-defined sections, but it cannot be expected to work well on more freely formulated sections. Therefore, we also use **Hidden Markov Models** (HMM), which are supposed also to include infor-

Section Phrase	Train	Held-out	Test	Avg. Length [words]
Corroboratio	23	2	29	15.1
Datatio	22	3	30	7.8
Dispositio	20	3	30	94.7
Inscriptio	15	4	5	3.7
Intitulatio	22	4	30	4.1
Narratio	4	1	2	20.7
Publicatio	21	3	30	4.4

Table 3: Statistics of manually marked section phrases.

Method	Named Entities	Lemmatized	Precision	Recall	F-score
Cosine	No	No	0.92	0.19	0.31
Cosine	No	Yes	0.93	0.20	0.33
Cosine	Yes	No	0.86	0.27	**0.41**
Cosine Max 3	No	No	**0.94**	0.17	0.29
TF IDF	No	No	0.88	0.23	0.36
TF IDF	No	Yes	0.83	0.22	0.35
TF IDF	Yes	No	0.83	0.25	0.38
HMM	No	No	0.67	0.27	0.38
HMM	No	Yes	0.65	0.20	0.31
HMM	Yes	No	0.52	**0.30**	0.38

Table 4: Comparison of information retrieval and HMM setting trained with and without marked named entities and with and without lemmatization. If named entities are set to "No", then all manually marked entities were removed from the train data. Best results are in **bold** type.

Train Documents Type	Test Documents Type	Precision	Recall	F-score
All	All	0.92	0.19	0.31
All	Fealty deeds	0.91	**0.22**	**0.35**
All	Acquaintance	**0.96**	0.14	0.24
Fealty deeds	All	0.91	0.20	0.32
Fealty deeds	Fealty deeds	0.91	0.21	0.34
Fealty deeds	Acquaintance	**0.96**	0.15	0.25

Table 5: Comparison of the cosine similarity method trained and tested on different document types. Best results are in **bold** type.

mation about sections ordering into the decision. Hidden states in the proposed HMM are formed by individual section types. These hidden states then generate charter text. As the amount of train data is small and the output probabilities for individual words can be skewed, cosine distance is used instead of the output probability. This distance is calculated between continuous word sequence which ends in the current state and the closest phrase manually assigned to this state (i.e. section type) in the train data. More precisely, all words in the word sequence need to be assigned to the same state as the current one. Finally, each word has an assigned section type. As the sections created in this way can be non-continuous and several word sequences can be marked to belong to a single section type, we assign only the continuous word sequence with the highest cosine similarity score to each section type.

4. Results and Discussion

Results of tested approaches are tabulated in Table 4 and Table 5. All methods are evaluated using precision, recall and F-score.

The highest precision scores are achieved using information retrieval-based method with cosine measure. The overall highest F-score of 0.41 is obtained for cosine measure when it is trained with all available named entities and no lemmatization is employed. The overall highest precision is acquired when this measure is calculated as a sum of cosine distances over three most similar phrases. We confirm that information retrieval-based methods perform especially well on more standardized section types. For example, the precision of retrieval *intitulatio* is 1, but the returned sequences are very short and they often contain only a few words.

As predicted, HMM-based method achieve better results for less formally standardized sections. It produces longer phrases and thus also achieves higher recall. The recall is highest when all named entities are used and no lemmatization is used. Though, the HMM-based method does not perform well when a particular section is missing from the test document.

To test retrieval of specific charter type, we compare retrieval trained on full train set and the *Fealty deeds* documents only. We test retrieval on full dataset, *Fealty deeds* documents and *Acquaintance* documents as we only have maximally three documents from other charter types in the test set. These results show that the differences between different document types are small and thus confirm the uniformity of our collection.

5. Conclusion and Future Work

In the paper, we presented the digitized and annotated collection of medieval Latin charters. We provided a detailed inspection of sections of such documents with respect to different issuance properties. In addition to manual annotations of the sections available in our collection, we also provided manually crafted named entities and Czech abstracts.

We also presented methods for detection of sections of documents and applied them to our collection. The highest precision was achieved when we used word sequences with the highest sum of cosine similarity scores calculated over three most similar phrases assigned to particular section type in the train data. The highest recall was achieved for the HMM-based algorithm.

We confirm that used methods can substantially reduce the effort needed by historians to process medieval charters, what enables them to work faster and thus to analyze larger amounts of data. As the methods were initially been not proposed to replace manual work completely but to allow researchers to work in a semi-automatic way, achieved precision is especially encouraging, as it enables reliable detection of probable word sequences assigned to sections. Researchers can then easily manually correct these results, if needed, by including additional words.

However, we would still further like to improve relatively low recall and thus enlarge the length of detected sections. We believe that better post-processing of the HMM algorithm can improve this. Moreover, we also plan to combine both methods to be able to make an advantage of the high precision achieved by information retrieval- based methods on short sections and recall achieved by the HMM-based methods on more freely formulated sections.

6. Acknowledgements

This research is supported by the Czech Science Foundation, grant number P103/12/G084 and NSF grant number 1618695.

7. Bibliographical References

Burnard, L. and Rahtz, S. (2013). A Complete Schema Definition Language for the Text Encoding Initiative. Presented at XML London 2013.

Choi, F. Y. Y. (2000). Advances in Domain Independent Linear Text Segmentation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, NAACL 2000*, pages 26–33, Seattle, WA, USA.

Guyotjeannin, O., Pycke, J., and Tock, B. (1993). *Diplomatique Médiévale*. L'atelier du médiéviste. Brepols.

Hearst, M. A. (1997). TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics*, 23(1):33–64.

Kozima, H. (1993). Text Segmentation Based On Similarity Between Words. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics ACL '93*, pages 286–288, Columbus, Ohio.

Lin, J., Karakos, D., Demner-Fushman, D., and Khudanpur, S. (2006). Generative Content Models for Structural Analysis of Medical Abstracts. In *Proceedings of the Workshop on Linking Natural Language Processing and Biology: Towards Deeper Biological Literature Analysis*, pages 65–72, New York City, New York.

Mohri, M., Moreno, P., and Weinstein, E. (2010). Discriminative Topic Segmentation of Text and Speech. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 533–540, Chia Laguna Resort, Sardinia, Italy.

Morris, J. and Hirst, G. (1988). Lexical Cohesion, the Thesaurus, and the Structure of Text. Technical report, Computer Systems Research Institute, University of Toronto, Toronto, Canada.

Nguyen, V. C., Nguyen, L. M., and Shimazu, A. (2011). Improving Text Segmentation with Non-systematic Semantic Relation. In *Computational Linguistics and Intelligent Text Processing: 12th International Conference, CICLing 2011, Part I*, pages 304–315, Tokyo, Japan.

Ponte, J. M. and Croft, W. B. (1997). Text Segmentation by Topic. In *Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries ECDL '97*, volume 1324 of *LNCS*, pages 113–125, Pisa, Italy.

Ruffolo, P., Passarotti, M., Budassi, M., and Litta, E. (2017). The Lemlat 3.0 Package for Morphological Analysis of Latin. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, number 133, pages 24–31.

Vogeler, G. (2010). Charters Encoding Initiative Overview. In *Digital Proceedings of the Lawrence J. Schoenberg Symposium on Manuscript Studies in the Digital Age*, volume 2.

von Estgen, A., Pauly, M., Pettiau, H., and Schroeder, J. (2009). *Urkunden- und Quellenbuch zur Geschichte der altluxemburgischen Territorien IX. Die Urkunden Graf Johanns des Blinden (1310–1346). Teil 2. Die Urkunden aus den Archives Générales du Royaume Brüssel*. Luxemburg.

SB-CH: A Swiss German Corpus with Sentiment Annotations

Ralf Grubenmann, Don Tuggener, Pius von Däniken, Jan Deriu, Mark Cieliebak
Zurich University of Applied Sciences, Winterthur, Switzerland
SpinningBytes AG, Küsnacht, Switzerland
rg@spinningbytes.com, tuge@zhaw.ch, piusvd@gmail.com, deri@zhaw.ch, mc@spinningbytes.com

Abstract

We present the SB-CH corpus, a novel Swiss German corpus with annotations for sentiment analysis. It consists of more than 200,000 phrases (approx. 1 Mio tokens) from Facebook comments and online chats. Additionally, we provide sentiment annotations for almost 2000 Swiss German phrases. We describe the methodologies used in the collection and annotation of the data, and provide the first baseline results for Swiss German sentiment analysis.

Keywords: Swiss German, Sentiment Analysis, Emotion Corpus, Language Resources

1. Introduction

Swiss German denotes a collection of Alemannic dialects spoken in the German-speaking part of Switzerland. Although not one of the official languages, it is spoken on a daily basis by an estimated 4,5 million speakers, i.e. by more than three fifths of the Swiss population (Coray and Bartels, 2017). Swiss German is almost exclusively used for private communications between native speakers, while Swiss Standard German, which closely resembles the written German in Germany, is used for official and public communications. Due to its non-official status and the phonetic differences between the dialects, there is no standardized spelling for Swiss German, and it has been hardly used in written form (Baumgartner, 2003). Through the advent of social media and messaging systems, Swiss German has increasingly expanded to written form (Stark et al., 2015, e.g.). However, resources for written Swiss German are still sparse, and it can be considered a low-resourced language. Combined with the fluent nature of the (mostly) oral language, it is non-trivial to find a large collection of written dialect. Furthermore, there exists no Swiss German corpus for Sentiment Analysis.

Contributions In this paper, we present the following[1]:

- A new corpus *SB-CH* composed of more than 200,000 Swiss German phrases[2] and approx. 1 Mio tokens.
- *Sentiment annotations* for 1843 phrases with labels positive, negative, or neutral.
- *Baselines* for Swiss German sentiment classification.

2. Related Work

Language Resources for Swiss German At present, we are aware of the following resources for Swiss German:

- Swiss SMS Corpus (Stark et al., 2015): a collection of SMS which were sent by the Swiss public in 2009/2010. It contains 10,708 SMS in Swiss German, together with demographic information about the author.
- NOAH's Corpus of Swiss German Dialects (Hollenstein and Aepli, 2015): a compilation of 7,453 Swiss German texts from various genres, collected between 2010-2014. Text sources include the Alemannic Wikipedia, blog posts, novels by Viktor Schobinger, newspaper articles from "Blick am Abend", and the Swatch Annual Business Report. The texts were manually annotated with part-of-speech tags for 106,987 tokens.
- Sprachatlas der Schweiz (Baumgartner, 2003): a documentation of regional differences in Swiss German dialects, based on data collected between 1939-1957. It contains more than 1500 maps, depicting dialect varieties in Switzerland.
- Kleiner Sprachatlas der Deutschen Schweiz (Christen and Renn, 2010): an excerpt of the "Sprachatlas der Schweiz" with 120 maps and explanations.
- ArchiMob corpus (Samardžić et al., 2016): this corpus contains 300 interviews in Swiss German about the Second World War. Of these, 34 recordings were manually transcribed into "Schwyzertütschi Dialäktschrift" (Dieth, 1986), resulting in 528,381 tokens.

To complement these resources with a large-scale corpus, we have compiled a new corpus of written Swiss German, called SB-CH. This corpus is composed of 203,242 Swiss German phrases and 981,247 tokens. The texts were retrieved from Facebook comments and chat messages.

Sentiment Analysis Automatic sentiment analysis is a fundamental research area in natural language processing (NLP) and serves as a flagship task for other classification problems. Generally, the goal of sentiment analysis is to classify a text into one of the classes positive, negative, or neutral. Sentiment analysis has gained the interest of both academia and industry due to its important applications in e.g. social media monitoring or customer care. Various initiatives exist in the scientific community, such as shared

[1]The corpus including the annotations is available here: `https://www.spinningbytes.com/resources/swissgermansentiment`

[2]The term *phrases* is used to describe a unit of the dataset, which is typically a sentence or a short paragraph of text.

tasks at SemEval (Nakov et al., 2013) or TREC (Ounis et al., 2008). At present, most solutions for sentiment analysis incorporate supervised machine learning algorithms such as Support Vector Machines (Jaggi et al., 2014) or Convolutional Neural Networks (Deriu et al., 2016). In order to develop these systems, manually annotated training data is required. Various sentiment corpora exist for e.g. English (Nakov et al., 2013), German (Cieliebak et al., 2017), French (Bosco et al., 2016), or Italian (Barbieri et al., 2016). However, we are not aware of any sentiment corpora for Swiss German. For this reason, we enriched the SB-CH corpus with sentiment annotations for 1843 distinct phrases. In this paper, we describe the annotation process and measure the agreement among annotators along with key metrics of the corpus. Furthermore, we present results of baseline sentiment classifiers trained on our dataset. To the best of our knowledge, SB-CH is the largest collection of Swiss German documents at present, and the first one that contains sentiment annotations.

3. SB-CH Corpus Overview

The corpus consists of two parts: the full corpus of collected Swiss German texts and the subsection containing the phrases annotated with sentiment. This section details the properties of the full corpus, whereas the next section focuses on the annotated phrases.

3.1. Sources

To compile the corpus, we crawled two messaging platforms.

Facebook page "Schwiizerdütsch" The Facebook page "Schwiizerdütsch" (Swiss German)[3] posts about current events, news, and tradition in Switzerland. The page was crawled for comments on posts, most of which are written in Swiss German by native speakers. The comments were extracted using the official Facebook API, and no private posts or profiles were included. The comments were stored without user information and sanitized of user mentions. In total, 70,904 comments were crawled covering a time period from 2010 to 2017.

Chatmania The online chat platform "Chatmania"[4] was crawled for chat messages in public chatrooms. Since the platform is based on the Internet Relay Chat (IRC) protocol, messages could be obtained by simply joining the IRC channels and logging all messages. The messages were then filtered from the logs and cleaned of usernames and other possibly identifying information.

3.2. Corpus Statistics

SB-CH contains 203,242 Swiss German phrases with 981,247 tokens. Figure 1 shows the distribution of phrase lengths over the entire corpus, and Table 1 provides statistical data per source type.

The average phrase length in SB-CH is 28.6 characters, which is a lot shorter compared to other Swiss German corpora (e.g. 88.6 for NOAH corpus and 117.5 for Swiss SMS Corpus). This is probably due to the fact that SB-CH is built from social media posts and from turns in online chats, which can be quite short.

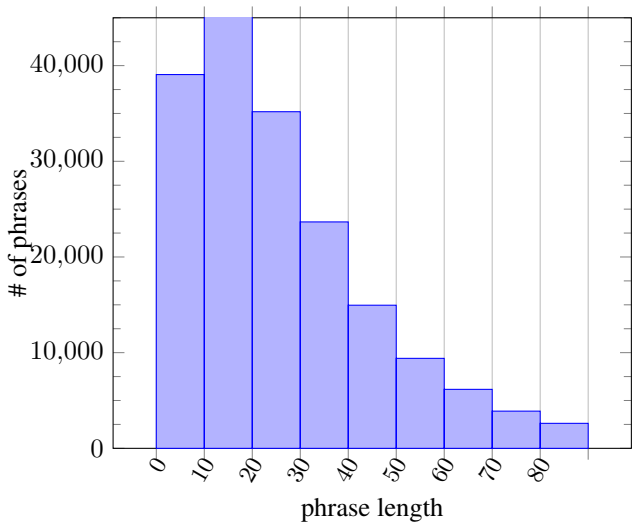


Figure 1: Histogram of phrase lengths in SB-CH by number of characters

Source	Phrases	Tokens	Characters per Phrase
Facebook	87,892	424,185	31.8
Chatmania	115,350	557,062	28.1
Total	**203,242**	**981,247**	
Corpora included for sentiment annotation			
NOAH corpus	7,453	106,987	88.6
Swiss SMS Corpus	10,708	217,940	117.5

Table 1: Corpus composition

4. Sentiment Annotation

In this section, we focus on the ongoing sentiment annotation of SB-CH. We describe the annotation scheme, the annotation guidelines, the annotation process, and provide statistics of the resulting annotations.

4.1. Sources

Two existing corpora were included alongside our newly created corpus SB-CH to get a more varied pool of phrases for annotation with sentiment polarity. These corpora are the NOAH Corpus (Hollenstein and Aepli, 2015) and the Swiss SMS Corpus (Stark et al., 2015). Random samples were then drawn from the pool and presented to annotators.

4.2. Annotation Scheme

The corpus sampled for sentiment annotation mainly consists of social media messages (chats, micro-blogging posts, SMS) and is thus similar to the corpus used in the SemEval tasks on sentiment detection (Nakov et al., 2013)

[3] https://www.facebook.com/Schwiizerduetsch/

[4] http://www.chatmania.ch/

which is compiled from Twitter and SMS messages in English. Following the SemEval scheme, we asked annotators to annotate message level polarity.[5] In addition to the standard labels for positive, negative and neutral (POS, NEG, NEUT), we used the label NA (i.e. *not applicable*) for phrases that were unintelligible due to e.g. errors in sentence splitting, not Swiss German, or too short ($<$ 3 words). Furthermore, the UNSURE label was introduced for phrases that contain both positive and negative sentiments, or phrases that depend on context for disambiguation w.r.t. polarity. The UNSURE label was also used for messages containing irony and sarcasm. The distinction between NA and UNSURE was introduced to separate noisy data from messages that might be interesting for future work (e.g. irony detection).

4.3. Annotation Process

Annotations were performed by 5 annotators using a custom web-based annotation platform. The group of annotators was composed of scientific assistants employed by the Zurich University of Applied Sciences (ZHAW) and employees of SpinningBytes AG. The annotators were from the cantons[6] of Aargau, Schaffhausen, Bern, St. Gallen and Zurich, giving a good distributions of speakers of different dialect groups.

A small set of phrases, picked following a uniform random distribution, was first annotated and studied to help develop the annotation guidelines. This was followed by two iterations of annotations, with annotators labeling randomly picked samples of the corpus in the first iteration, followed by the second iteration where the annotators labelled phrases previously annotated by a different annotator to ensure multiple annotations per phrase.

4.4. Annotation Guidelines

The texts in the corpus were randomly sampled from the heterogeneous sources of the corpus and presented to the annotators without context. Thus, we applied a guideline that focuses on text-level polarity, abstracting away from the (opaque) writer's intent. That is, annotators were asked to judge the sentiment of the phrases mainly by the polarity of the vocabulary encountered in them and regardless of pragmatics, where possible. For example:

- *Guete Morge!* is a common Swiss German greeting, *Good Morning*, and contains the word for *good*, which has a clearly positive polarity, so the text is judged as positive.
- *Lüg nid!* is an imperative, meaning *Don't lie!*, which contains the clearly negative word *lying*, so it is judged as negative.

The latter example could also appear as a flirtatious response in an online chat and be intended to carry an enticing, positive polarity. However, without the accompanying dialogue context, this cannot be inferred, and thus the phrase is labelled as negative based on the word *lying*.

[5]However, we did not ask annotators to annotate individual words or word spans within the messages with polarity.

[6]Member states of the Swiss Confederation

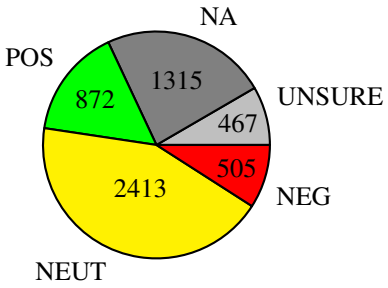


Figure 2: Distribution of labels in annotations

Questions were judged as neutral by default, since question polarity depends highly on pragmatics and intentions of the questioner cannot be determined reliably without it.
The distinction between subjectively or objectively positive/negative was of lesser importance. *"Ich mues is gfängnis"* (*"I have to go to prison"*) is negative for the writer, but can also be judged to be objectively positive (*a criminal has been caught*). But the word *prison* has a negative polarity, so the phrase is judged as negative.
While including salutations and greetings in the annotations is less common, our motivation to annotate them was to be able to capture polar information of important Swiss German adjectives (e.g. *Guet*) that are helpful in classification. This was also the driving idea behind putting a general focus on the lexical items in the annotation guidelines. Ideally, our annotations can be used to bootstrap a classifier for other domains, such as customer support messages in Swiss German in the future.

4.5. Annotation Statistics and Inter-Annotator Agreement

The statistics of the annotations are summarized in figure 2. The chart shows the number of phrases annotated with each label so far. The annotations show a distribution skewed towards the *neutral* label, similar to a related corpus for sentiment analysis comprised of Tweets in Standard German (Cieliebak et al., 2017), but also encountered in an English Twitter corpus for sentiment (Nakov et al., 2013).
On average, each phrase has been annotated by 2.45 annotators, and more than 80% of the phrases have been labelled by more than one annotator.
To calculate inter-annotator agreement, we applied Krippendorff's α metric (Krippendorff, 2007). We obtained an α of 0.42 over all annotations, and an α of 0.75 if only POS, NEG and NEUT annotations were regarded. We attribute this comparably high agreement to the relatively straightforward annotation guidelines and the fact that annotators did not have to select spans in the phrases, but only had to label the phrase-level polarity. Since UNSURE and NA describe inherent uncertainty regarding the annotations, the low agreement when including these classes is to be expected.

4.6. Sentiment Classification Baselines

In the following, we provide baseline sentiment polarity results for SB-CH using a traditional Support Vector Machine model (Cortes and Vapnik, 1995) and fastText (Joulin et al.,

2016), a model based on a shallow neural network. As features for the SVM approach, we use the TF-IDF vectors of the unigrams in the phrases to predict their polarity. Since our data is highly skewed, we apply the class imbalance to weight the samples by inverse frequency of their labels. For fastText, we tried different parameter settings and settled for the best performing one, i.e. reducing the number of dimensions of the embeddings and increasing the number of epochs.[7]

For our experiments, we harmonized the annotations per phrase by majority vote (and removed the phrases with a tie) to create a gold standard. We then sampled 20% of the phrases using stratified sampling to create a test set and used the remaining 80% of the phrases as training data. We trained and evaluated the models using two sets of labels. The first run made use of only the POS/NEG/NEUT labels as is common in the sentiment analysis literature. Additionally, we used all labelled phrases to simulate a more realistic setting where a system has to classify all phrases encountered in a stream of messages, as in e.g. social media streams.

Common labels

	Precision	Recall	F1-score
NEG	0.35	0.28	0.31
NEUT	0.75	0.86	0.80
POS	0.61	0.39	0.48
avg	0.68	0.70	0.68

All labels

	Precision	Recall	F1-score
NA	0.78	0.79	0.79
UNSURE	0.00	0.00	0.00
NEG	0.28	0.23	0.25
NEUT	0.63	0.66	0.64
POS	0.45	0.48	0.47
avg	0.63	0.64	0.63

Table 2: Results of SVM baseline

The results of the baseline models are presented in Tables 2 and 3. We report Precision, Recall, and F1-Score for each of the labels and weighted average F1. We see that both the SVM and fastText achieve similar performance. For the common labels, both models achieve rather low F1-Scores for the POS and NEG labels. This is not surprising due to the small size of our data at the current stage, and we believe results will improve once more phrases have been annotated. Unsurprisingly, including all labels in the classification task leads to a drop in performance for both models regarding the POS, NEG, and NEUT labels. The models perform quite well in labeling the NA class, but struggle with the UNSURE class. This is due to the lower amount of examples for the UNSURE class and the fact that the NA class subsumes noisy data that is often not Swiss German and thus consists of a different lexical items than the other classes.

The performance of the baselines at this stage of the annotation process meets our expectations and indicates that the annotation effort is going in the right direction.

[7] flags: -minCount 0 -dim 50 -epoch 50

Common labels

	Precision	Recall	F1-score
NEG	0.39	0.23	0.29
NEUT	0.74	0.90	0.81
POS	0.59	0.34	0.43
avg	0.67	0.70	0.67

All labels

	Precision	Recall	F1-score
NA	0.73	0.79	0.76
UNSURE	1.00	0.10	0.18
NEG	0.25	0.10	0.14
NEUT	0.62	0.75	0.68
POS	0.47	0.30	0.37
avg	0.63	0.65	0.62

Table 3: Results of fastText baseline

5. Conclusion

We presented, to the best of our knowledge, the largest collection of Swiss German texts, and the first annotated corpus of sentiment polarity for Swiss German. We achieved a high inter-annotator agreement of 0.75 (Krippendorff α) for our ongoing manual sentiment annotation, and we created baselines with reasonable F1-Scores for automatic sentiment prediction.

We expect that these resources will enable other researchers to address new and interesting research questions for Swiss German. One such question that we would like to tackle next is how resources and technologies for Standard German, which are available, can be utilized to improve solutions for Swiss German. If successful, such technology transfers could be applied to other low-resourced languages.

6. Acknowledgements

Zurich University of Applied Sciences (ZHAW) and SpinningBytes AG are collaborating in a joint research project to develop state-of-the-art solutions for text analytics tasks in several European languages. This research has been funded by Commission for Technology and Innovation (CTI) project no. 18832.1 PFES-ES.

7. References

Barbieri, F., Basile, V., Croce, D., Nissim, M., Novielli, N., and Patti, V. (2016). Overview of the EVALITA 2016 sentiment polarity classification task. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*.

Baumgartner, H. (2003). *Sprachatlas der deutschen Schweiz*, volume 9. Francke.

Bosco, C., Lai, M., Patti, V., and Virone, D. (2016). Tweeting and being ironic in the debate about a po-

litical reform: the French annotated corpus Twitter-Mariagepourtous. In *LREC*.

Christen, H. and Renn, M. (2010). *Kleiner Sprachatlas der deutschen Schweiz*. Huber.

Cieliebak, M., Deriu, J., Egger, D., and Uzdilli, F. (2017). A Twitter corpus and benchmark resources for German sentiment analysis. In *Proceedings of the 4th International Workshop on Natural Language Processing for Social Media (SocialNLP 2017)*.

Coray, R. and Bartels, L. (2017). *Schweizerdeutsch und Hochdeutsch in der Schweiz*, volume BFS-Nummer 1762-1700. Bundesamt für Statistik, Schweizerische Eidgenossenschaft (BFS).

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297, Sep.

Deriu, J., Gonzenbach, M., Uzdilli, F., Lucchi, A., De Luca, V., and Jaggi, M. (2016). SwissCheese at SemEval-2016 task 4: Sentiment classification using an ensemble of convolutional neural networks with distant supervision. In *SemEval@ NAACL-HLT*, pages 1124–1128.

Dieth, E. (1986). *Schwyzertütschi Dialäktschrift. Dieth-Schreibung*. Verlag Sauerländer.

Hollenstein, N. and Aepli, N. (2015). A resource for natural language processing of Swiss German dialects. In *GSCL*, pages 108–109.

Jaggi, M., Uzdilli, F., and Cieliebak, M. (2014). Swiss-Chocolate: Sentiment detection using sparse SVMs and part-of-speech n-grams. In *SemEval@ COLING*, pages 601–604.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Krippendorff, K. (2007). Computing Krippendorff's alpha reliability. *Departmental papers (ASC)*, page 43.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in Twitter. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 312–320.

Ounis, I., Macdonald, C., and Soboroff, I. (2008). On the TREC blog track. In *ICWSM*.

Samardžić, T., Scherrer, Y., and Glaser, E. (2016). Archimob - a corpus of spoken Swiss German. In Nicoletta Calzolari, et al., editors, *Language Resources and Evaluation (LREC 2016)*, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), pages 4061–4066. s.n., May.

Stark, E., Ueberwasser, S., and Ruef, B. (2015). Swiss SMS corpus. www.sms4science.ch.

Universal Dependencies for Ainu

Hajime Senuma†‡, Akiko Aizawa‡†
†University of Tokyo, ‡National Institute of Informatics
{senuma, aizawa}@nii.ac.jp

Abstract

This paper reports an on-going effort to create a dependency tree bank for the Ainu language in the scheme of Universal Dependencies (UD). The task is crucial both language-internally (language revitalization) and language-externally (providing sources for new features and insights to UD). Since the language shows many of the representative phenomena of a type of languages called polysynthetic languages, an annotation schema to Ainu can be used as a basis to extend the current specification of UD. Our language resource comprises an annotation guideline, dependency bank based on UD, and a mini-lexicon. Although the size of the dependency bank will be small and contain only around 10,000 word tokens, it can serve as a base annotation for the next step. Our mini-lexicon is encoded under the W3C OntoLex specification with UD and UniMorph (UM) features with the system-friendly JSON-LD format and thus bearable to future extensions. We also provide a brief description of dependency relations and local features used in the bank such as pronominal cross-indexing and alienability.

1. Introduction

The project of Universal Dependencies (UD) (Nivre et al., 2016) marks a milestone in the history of natural language processing (NLP), as it unifies syntactic annotation schemes across languages/corpora and enables cross-lingual processing. At the CoNLL Shared Task 2017 for UD, 33 teams participated in the task, proving that the community is quickly thriving (Zeman et al., 2017).

We are currently working on the UD annotation scheme and corpus of Ainu, a language spoken by the Ainu people, an ethnic minority in Japan. A part of the work was presented elsewhere (Senuma and Aizawa, 2017). The task aims to tackle two purposes:

1. to promote the language revitalization of Ainu, as it is a highly endangered language but has a rich amount of the records of oral literature, and

2. to serve as a basis to test if universal specifications in NLP such as UD and UniMorph (UM) (Sylak-Glassman, 2016) can encode the world's languages.

The first issue is urgent because Ethnologue classifies the languages as nearly extinct (Lewis et al., 2016), and scholars have been alerting the status of language usages (DeChicchis, 1995; Sato, 2012). To mitigate the situation, Bugaeva (2011) created a freely-accessible Ainu dictionary for daily conversation based on the old dictionary published in 1898 by Kotora Jimbō and Shōzaburo Kanazawa. Our work will be in line of these movements.

Whereas the first motivation is language-internal, the second motivation is external, as it contributes to our deep understanding of natural languages. Being a member of polysynthetic languages (Baker, 1996), the language exhibits many peculiar properties from the view point of the speakers of major languages. Complex verb formation rules called noun incorporation (Mithun, 1984) may be the hallmark of such properties. Other properties (some of which will be discussed later) include pluractionality, possessed case, alienability, and multiple pronominal markers. These "peculiar" properties are, however, actually very common in world's languages, if we look at typological data such as WALS Online (Dryer and Haspelmath, 2013). Through the examination of annotating Ainu, it may be possible to be used to extend the current specifications of NLP including UD.

2. Dataset and System Description

We are currently working on the annotation of a collection of traditional Ainu songs *Ainu Shin'yōshū*, transcribed by Yukie Chiri into latin scripts, including Japanese translations by herself (Chiri, 1923). We also consulted the work of (Kirikae, 2003), which retranscribed Yukie Chiri's writing into one close to modern orthographic system and appended a lexicon for all words appeared in the text. The work consists of thirteen mythological songs in the style of *kamuyyukar* type poetry, typical in the Ainu oral literature. Since the work is the most famous Ainu literature, distributing resources to read it in the open format will be useful for language revitalization. Furthermore, Ainu poetry is usually told by *atomte itak* "Adorned Speech", a variant of Ainu which has more polysynthetic nature than *yayan itak* "Common Speech", so it serves as a good resource to expand the inventory of UD and UM.

At this stage, the annotation process has been mainly done by the first author alone, since it is experimental and requires the quality of linguistic efforts rather than the quantity of human power. Although the annotation process is behind schedule, but we plan to release the data set until camera-ready. We estimate the size of the resulting corpus will be around 10,000. It is very small in comparison with other corpora, but still it is enough to create a sound documentation for annotating the language.

Although Ainu did not have orthographic systems in ancient times, two writing systems have been used in these 200 years: one based on latin scripts and one based on Japanese katakana. Latin-based one reflects the phonology of Ainu more accurately, while katakana-based system is more friendly for elder people and young children in Japan. the Foundation of Research and Promotion of Ainu Culture (FRPAC) provides instructions for both systems.

Since the original Ainu text was written in the outdated version of latin-based orthography, we retranscribed it into a

latin-based modern orthographic system , almost equivalent to FRPAC. We, however, adopted two orthography rules not seen in FRPAC, following Tamura (2000):

1. irregular accent positions are marked by an acute, as in *húre*, and
2. irregular positions of glottal stops are marked by apostrophe, as in *yay'eyukar*.

Our data set contains three items:

1. UD annotation guideline, in the form of Markdown texts, the official format for UD documentation,
2. dependency tree bank formatted in JSON, and
3. mini-lexicon formatted in JSON-LD, following the W3C OntoLex/lemon inventory.

3. Data Format

This section briefly discusses the design of data format and system used in our annotation.

3.1. Dependency bank

Our dependency bank is written in JSON format. It has a similar style to sd-parse, the official UD annotation format for documentation, although it is still JSON, maintaining easiness to be used in systems.
By using a converter, we translate data into CoNLL-U, the official format of UD. We also developed a dependency bank viewer for our dataset (Figure 1). A tooltip pops up to show the contents of our lexicon by pointing to a word that the reader does not know. The viewer will serve as a handy tool to learn the Ainu language.

3.2. Lexicon

Each entry of our lexicon contains basic information such as word forms, pronunciations, and the concise definitions of its senses. It also contains bibliographic references to three dictionaries: Kirikae (2003; Nakagawa (1995; Tamura (1996). It is formatted in JSON. For example:

```
{
  "@context":
    "...",
  "@id": "san_1",
  "@type": "Word",
  "canonicalForm": {
    "latn": "san",
    "kana": "",
    "ipa": "san"
  },
  "lexicalForm": [
    {
      "latn": "sap",
      "kana": "",
      "ipa": "sap",
      "feature": [
        "um:intr",
        "x:pluract"
      ]
    }
  ],
  "pos": "verb",
  "feature": ["um:intr"],
  "sense": [
    {
      "@id": "san_1_1",
      "reference": {
        "prefLabel":
          "to go downstream"
      },
      "usage": "...",
      "bibliography": [
        {
          "bib:key": "bib:Kirikae2003",
          "bib:loc": "p.˜378"
        },
        {
          "bibkey": "bib:Nakagawa1995",
          "bib:loc": "p.˜203"
        },
        {
          "bibkey": "bib:Tamura1996",
          "bib:loc": "pp.˜602-603"
        }
      ]
    },
    {
      "@id": "san_1_2",
      "reference": {
        "prefLabel": "to go toward ..."
      },
      "bibliography": [
        {
          "bib:key": "bib:Tamura1996",
          "bib:loc": "pp.˜602-603"
        }
      ]
    }
  ]
}
```

In reality it is in W3C JSON-LD (Sporny et al., 2017), the specification for linking data. By using a context file it is possible to expand each element into the RDF triples, enabling ontological data exchange. For example, in our context file, there are mappings like the following.

```
"ontolex": "http://www.w3.org/ns/lemon/ontolex#",
"latn": {
  "@id": "ontolex:writtenRep",
  "@language": "ain-Latn"
},
```

Then we can convert it into expanded triples as

```
_:b0
http://www.w3.org/ns/lemon/ontolex#writtenRep
"san"@ain-latn .
```

This way we can link our lexicon to the vocabulary of OntoLex/Lemon (W3C Ontology-Lexicon Community, 2017), bridging gaps.
Our lexicon has features associated to the inventory of UM (Sylak-Glassman, 2016; Cotterell et al., 2017) which also allows morphosyntactic descriptions.
Although not seen in the above example, our lexicon can also contain word compositions and synonym/antonym relations. Through OntoLex it may be possible to connect to WordNet in the future.

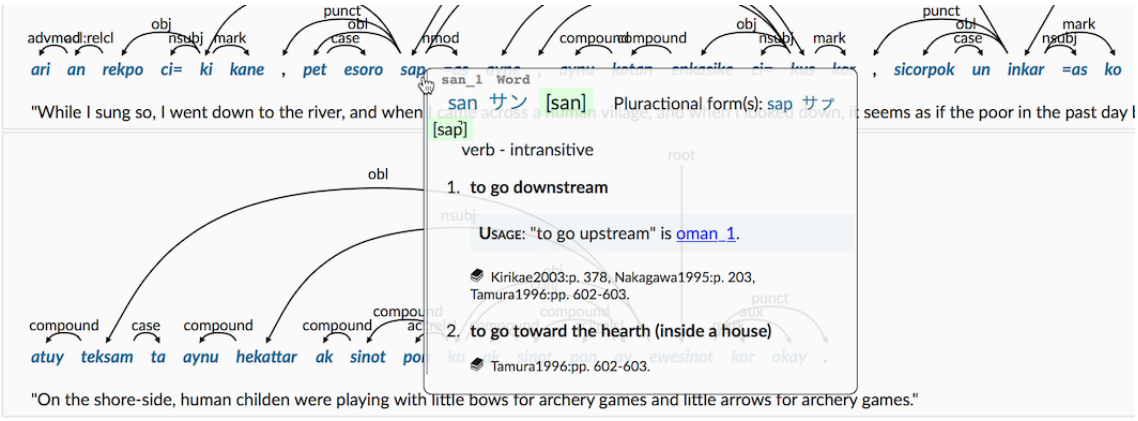


Figure 1: Screen shot of the dependency bank viewer with linking to the lexicon

4. Linguistic Properties

Ainu is abundant for linguistic phenomena not seen in major languages. In this extended abstract we mention two of such properties.

In this section, the style of interlinear glossings are roughly based on the Leipzig Glossing Rules (Comrie et al., 2008).

4.1. Pronominal cross-indexing

The first one is related to pronominal agreement (or pronominal *indexing* in the functionalist terminology (Croft, 2003)). The Ainu language has two sets of pronominal markers: clitics (such as *e*=, "you") and pronouns (such as *eani* "you"). Nevertheless, in almost all cases, pronouns do not occur in texts, and only clitics are realized.

horkew *e=* *ne*
wolf 2= COP

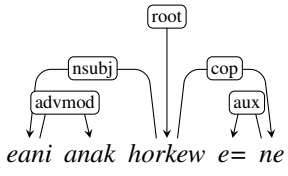
"You are a wolf."

However, in some cases (especially if we want to emphasize who involved a topic), pronouns can be optionally used, although clitics are obligatory.

eani *anak* *horkew* *e=* *ne*
2SG.S INT wolf 2SG.S= COP

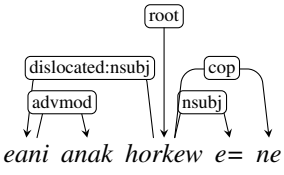
"(As for you), you a wolf."

This kind of phenomena prohibits us to annotate even for simple declarative sentences, because we do not know which tokens should be counted as subject. In previous work (Senuma and Aizawa, 2017), we annotated this phenomenon in the following strategy.



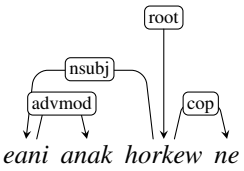
The problem of this approach is that the pronominal clitic *e*= is treated as an auxiliary, rather than an argument. This approach contradicts to formalists and functionalists in orthodox linguistics. Some formalists in the Chomskyan tradition claim that clitics (such as *e*=) are true arguments to predicates in some polysynthetic languages and pronouns are mere adjuncts (Jelinek, 1984; Baker, 1996). On the other hand, functionalists reject the idea of arguments/adjuncts dichotomy in the first place; Haspelmath (2013) called pronouns/nominals as *cominals* and claimed that in these *cross-indexing* constructions both are real arguments . At any rate, a clitic *e*= must be treated as an argument, for cross-lingual comparisons, although at the same time we should also adhere to UD's approach that there should be no more than one `nsubj` (nominal subject).

A solution suggested by the UD community is the usage of `dislocated`, a relation originally used to encode dislocated pronouns, commonly seen in informal French. Descriptively, the approach is also justified by the fact that in an Ainu grammar published in 1936, Mashiho Chiri "likens the use of the Ainu personal pronouns to those of Latin and French, and contends that the following expressions are parallel in their use of the over pronoun (Ainu *kuani* 'I', Latin *ego*, and French *moi*)" (Shibatani, 1990, p. 30). However, unlike the inherent informality of French, Ainu systematically uses this system. According to Haspelmath (2013, p. 8), French dislocation is not counted as conominals, because pronominal indexes (such as *e*= in Ainu) and conominals must be in the same narrow phrases, while French dislocation occurs outside these phrases. We thus introduce language-specific features `dislocated:nsubj`, `dislocated:obj`, and `dislocated:iobj` to indicate that they are conominals with valid status as subjects and objects, without violating UD's restriction that a predicate must not have more than one argument per role.



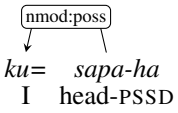
In some dialects of Ainu, for informal speech, it is re-

ported that pronominal clitics were dropped, possibly due to the influence of Japanese (Izutsu, 2006). Since UD has the realization-first approach, in these cases conominals are promoted to common arguments, as in:



4.2. Alienability and possessed case

In most major languages, syntactic information is marked on its dependent, e.g., "my head" (rather than "I head-have"). The Ainu language, however, marks it on the head (*possessum* (Croft, 2003) or *possessee*), as in *ku=sapa-ha* "I=head-PSSD" in some circumstances.

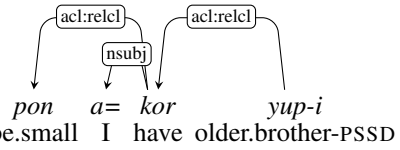


"my head"

Such phenomenon is called the *possessed case* (Sylak-Glassman, 2016). In reality, out of 236 languages recorded in the WALS Online, the number of languages with dependent-marking is 98 (42%) and that with head-marking possession is 78 (33%), and therefore it is by no means a minor construction.

The UD corpus for a Uralic language Hungarian, being an exceptional language in Europe which has head-marking possession, utilizes UD's "layered feature" (e.g., `Person[psor]=3` "it is possessed by 3rd person") to annotate possessed cases. Ainu possessees do not, however, inflect on number/person, and its meaning is only realized on a clitic such as *ku=*, thereby prohibiting the usage of these systems. We thus borrowed PSSD (possessed case) from the UniMorph inventory, and used a language-specific feature `Case=Pssd`.

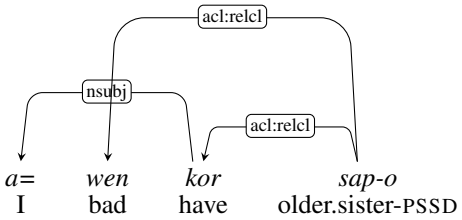
In addition, Ainu has another possessive construction with a relative clause (the phrase was taken from Shibatani (1990, p. 44)).



"my young older brother" (lit. "older brother that is young and that I=have")

In the above case, the form *a=kor yupi* ("older brother I have") is optional, and in common speech the form *a=yupihi* is preferred. But not all nouns have possessed forms. Only two classes of nouns called inalienable nouns (such as body parts and kins) and locative nouns (such as *or* "place" and *enka* "higher place (above)") have possessed cases. If nouns are alienable, relative clause based constructions are mandatory. We use language-specific lexical features `Alienability=Naln` (inalienable) and `Alienability=Aln` (alienable), borrowed from the inventory of UniMorph, and `Locativity=Yes` to annotate these words.

It is interesting to see that the pronominal clitic can reside in anywhere in the relative-clause construction, exhibiting crossing dependencies in some cases (the phrase was taken from Shibatani (1990, p. 44), too).



"my dear older sister" (lit. "older sister I=that is bad and (...) have")

5. Conclusion

In this paper, we presented an attempt to construct UD bank and lexicon for Ainu, including dataset description, data format description, and explanations for some linguistic properties. Compared with previous work published from the UD Workshop, we created a system and viewer to annotate Ainu more easily, and a lexicon which can be also used as a bridge to the community of morphology and the community of ontology/lexicon. We furthermore refined our annotation scheme so that it meets the standards of both linguistic typology and UD. We plan to publish the dataset in permissible open licenses such as CC-BY and MIT in the near future.

6. Acknowledgements

This work was supported by CREST, Japan Science and Technology Agency. We are also grateful to the following researchers for valuable information and suggestions: William Croft, Teresa Lynn, Hiroshi Nakagawa (Chiba University), Joakim Nivre, Sebastian Schuster, and Francis Tyers.

7. Bibliographical References

Baker, M. C. (1996). *The Polysynthesis Parameter*. Oxford University Press.

Bugaeva, A. (2011). Internet Applications for Endangered Languages: A Talking Dictionary of Ainu. *Waseda Institute for Advanced Study Research Bulletin*, 3:73–81.

Chiri, Y. (1923). *Ainu Shin'yōshū [Collection of Ainu Kamuyyukar]*. Kyodo Kenkyusha.

Comrie, B., Haspelmath, M., and Bickel, B. (2008). The Leipzig Glossing Rules. Technical report, Max Planck Institute for Evolutionary Anthropology and University of Leipzig.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., Xia, P., Faruqui, M., Kübler, S., Yarowsky, D., Eisner, J., and Hulden, M. (2017). CoNLL-SIGMORPHON 2017 Shared Task: Universal

Morphological Reinflection in 52 Languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

Croft, W. (2003). *Typology and Universals, 2nd ed.* Cambridge University Press.

DeChicchis, J. (1995). The current state of the Ainu language. *Journal of Multilingual and Multicultural Development*, 16(1-2):103–124, January.

Dryer, M. S. and Haspelmath, M. (2013). WALS Online. Technical report, Max Planck Institute for Evolutionary Anthropology.

Haspelmath, M. (2013). Argument indexing: a conceptual framework for the syntactic status of bound person forms. In Dik Bakker et al., editors, *Languages Across Boundaries: Studies in Memory of Anna Siewierska*, pages 197–226. Walter de Gruyter.

Izutsu, K. (2006). Ainugo no Hinshi Bunrui Saikō: Iwayuru Ninshō Daimeishi o Megutte [Ainu Parts of Speech Revisited: with Special Reference to So-called Personal Pronouns]. *Journal of Hokkaido University of Education: Humanities and Social Sciences*, 56(2):13–27.

Jelinek, E. (1984). Empty Categories, Case, and Configurationality. *Natural Language & Linguistic Theory*, 2(1):39–76, January.

Kirikae, H. (2003). *Ainu Shin'yōshu Jiten [Lexicon to Ainu Shin'yōshū]*. Daigaku Shorin.

Lewis, M. P., Simons, G. F., and Fennig, C. D. (2016). *Ethnologue: Languages of the World, Nineteenth edition*. SIL International.

Mithun, M. (1984). The Evolution of Noun Incorporation. *Language*, 60(4):847–894.

Nakagawa, H. (1995). *Ainugo Chitose Hōgen Jiten [The AinuJapanese Dictionary: Chitose Dialect]*. Sōfūkan.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666.

Sato, T. (2012). Ainugo no Genjō to Fukkō [The Present Situation of the Ainu Language and Its Revitalization]. *Gengo Kenkyu*, 142:29–44.

Senuma, H. and Aizawa, A. (2017). Toward Universal Dependencies for Ainu. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*.

Shibatani, M. (1990). *The Languages of Japan*. Cambridge University Press.

Sporny, M., Longley, D., Kellogg, G., Lanthaler, M., and Lindström, N. (2017). JSON-LD 1.1: A JSON-based Serialization for Linked Data, Draft Community Group Report 12. Technical report, W3C.

Sylak-Glassman, J. (2016). The Composition and Use of the Universal Morphological Feature Schema (UniMorph Schema), Working Draft, v. 2. Technical report.

Tamura, S. (1996). *Ainugo Saru Hōgen Jiten [The Ainu-Japanese Dictionary: Saru Dialect]*. Sōfūkan.

Tamura, S. (2000). *The Ainu Language*. ICHEL Linguistic Studies. Sanseidō.

W3C Ontology-Lexicon Community. (2017). Lexicon Model for Ontologies. Technical report.

Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinková, S., Hlaváčová, J., Kettnerová, V., Nka Urešová, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., De Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., De Paiva, V., Droganova, K., Alonso, H. M., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonça, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. *CoNLL*, (1):1–19.

Signbank: Software to Support Web Based Dictionaries of Sign Language

Steve Cassidy*, Onno Crasborn†, Henri Nieminen‡, Wessel Stoop†, Micha Hulsbosch†, Susan Even†, Erwin Komen†, Trevor Johnston*

*Macquarie University, Sydney, Australia,
{steve.cassidy, trevor.johnston}@mq.edu.au;

†Centre for Language Studies, Radboud University;
PO Box 9103, 6500 HD Nijmegen, The Netherlands;
{o.crasborn, w.stoop, m.hulsbosch, e.komen, s.even}@let.ru.nl;

‡University of Jyväskylä , Jyväskylä, Finland
henri.i.nieminen@student.jyu.fi

Abstract

Signbank is a web application that was originally built to support the Auslan Signbank on-line web dictionary, it was an Open Source re-implementation of an earlier version of that site. The application provides a framework for the development of a rich lexical database of sign language augmented with video samples of signs. As an Open Source project, the original Signbank has formed the basis of a number of new sign language dictionaries and corpora including those for British Sign Language, Sign Language of the Netherlands and Finnish Sign Language. Versions are under development for American Sign Language and Flemish Sign Language. This paper describes the overall architecture of the Signbank system and its representation of lexical entries and associated entities.

Keywords: sign language, lexical database, dictionary, web application

1. Introduction

Auslan Signbank [1] is an on-line dictionary for Australian Sign Language (Auslan) and is the latest in a long series of lexical resources for Auslan developed by Trevor Johnson and others. Beginning with a Filemaker Pro database (Johnston, 2001), the Auslan lexical database has been developed as a resource to inform the study of Auslan and the annotation of a corpus of Auslan video recordings (Johnston, 2008).

The original web-based version of Auslan Signbank was implemented commercially using Microsoft SQL Server which incurred a significant ongoing cost and meant that any changes to the data had to be made via contract with the developers. The current version of Auslan Signbank is an open-source re-implementation of the original website, with features added to allow updates to the database by researchers. Once this was implemented, the web based Auslan Signbank became the primary version of the dataset, removing the need to maintain the Filemaker Pro database.

The various Signbank websites support a range of users interested in sign language. In some cases, a public facing dictionary of sign language is presented with a search facility allowing users to find sign videos matching keywords. Registered users can also provide feedback to the Signbank editors about missing signs or errors in the existing sign entries. In this way, Signbank acts as a resource for the deaf community to help build a shared dictionary of the language.

Another class of users are researchers who are interested in using Signbank as a lexical resource in their research or in contributing to the rich lexical models in the dictionary. These researchers are able to see a much richer set of data than the public view and can use an advanced search facility over the data. Some researchers provide detailed feedback on the signs to the editors and can help in the construction of sign entries. A further set of users are the editors of Signbank who can create new signs and upload videos associated with them. Overall, this forms a rich community of sign language users collaborating to build a resource to support their language.

Since the Signbank software is an open-source project (distributed under a 3-clause BSD licence), it can be repurposed and extended by other projects. The software now forms the basis of three additional dictionaries with more in development. The goal of this paper is to describe the architecture of the Signbank software and the changes that have been made by the other projects. While there have been publications on the language resources themselves in the past, this is the first publication about the software system that underlies these resources.

2. The Signbank Application

The Signbank application is written in Python using the Django web application framework (Django Software Foundation, 2017). Django is a modern, widely used framework that provides a high level of support for building web applications based on complex data models. The choice of implementation environment was based on the widespread use of this environment and the consequent support for hosting these applications and the ease of finding developers with the appropriate expertise to maintain and extend the software. Django is actively maintained as an Open Source project and has frequent security updates that help to ensure that the Signbank application can be deployed safely on the web.

Django has many useful features built-in, including an object-relational mapper (ORM), extensible admin inter-

[1] http://auslan.org.au/

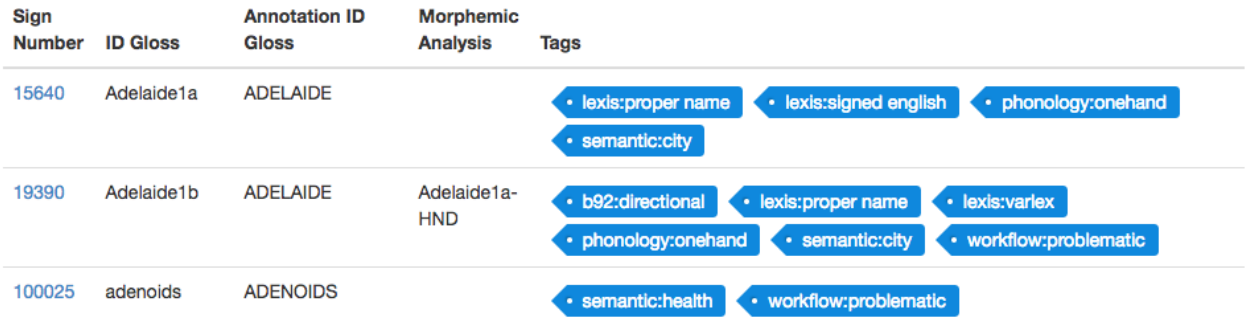


Figure 1: A screenshot from the sign search interface in auslan.org.au showing various fields associated with the Gloss and a number of Tags.

face, static files management, and internationalisation and localisation framework allowing translation of the application into multiple different languages. There are also a wide range of third-party packages for Django that can be used to provide parts of the Signbank service - for example, user management and web page design support. The ability to combine these packages with the core Signbank dictionary applications saves a lot of effort in deploying a complete web application.

2.1. Lexical Model

At the core of the Signbank dictionary is a lexical model that supports various levels of description of the signs in the language.

In the earlier FileMaker version of the dictionary, each sign was stored as an entry in a single table with a very large number of properties. This structure had been developed from the earlier CDROM dictionary and was added to as new descriptive properties were developed for the dictionary. The current model was developed as a normalised version of this table to allow, for example, an arbitrary number of keywords and definitions to be associated with a sign.

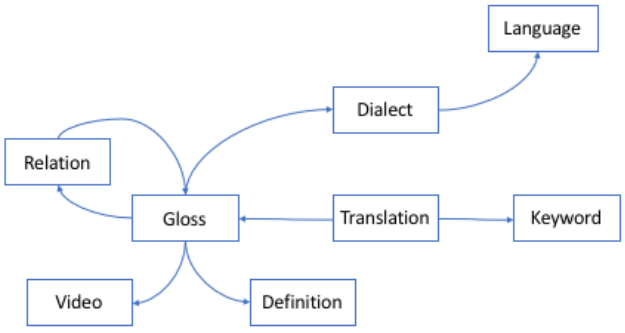


Figure 2: The entities in the Signbank data model.

The central entity in the data model is the sign, represented by the *Gloss* model; the model name reflects the central use of the dictionary to inform the annotation of sign language corpora, each lexical entry is associated with a textual gloss that is used in annotation. Each lexical entry has a unique identifier, the *IDGloss* which can be used to refer to this entry. A second *Annotation IDGloss* is the gloss used in annotation and may be shared by one or more lexical entries that differ in form in only minor or insignificant ways.

A Gloss has a number of descriptive properties that are used to describe the sign from a morphological and phonological perspective (Johnston, 2001).

Each Gloss can have an associated *Sign Number* that is optionally used to define the ordering of the entries in the dictionary. If the sign number is assigned to every sign, then the user can traverse the dictionary in sign number order to find signs that are near each other in the dictionary.

Relationships between signs can be represented using the *Relation* model to represent a named link between two gloss entries. New relation names can be defined as part of the dictionary. Common relations are *synonym*, *homophone* and *variant*.

As an example, in Fig. 1 two signs are shown with the Annotation IDGloss *ADELAIDE*, each with a different IDGloss identifier. In this case these are *variants* of a single sign (and will be linked with the variant relation).

A Gloss represents a single entry in the dictionary but may correspond to one of a number of alternate *senses* of a sign. In this case the same surface form (sign) has a number of meanings (homophony) and each is recorded as a separate Gloss entry. Each sense has a different sense number and senses are related to each other via the *homophone* relation.

To provide a search facility for the online dictionary based on words, each gloss can have one or more *Keywords* associated with it via the *Translation* model, in English or any other spoken language with an orthography. These keywords are intended to be the most common English translations for each sign. Each sign can also have one or more *Definitions* associated with it. Again, these definitions are written language definitions of the meaning of the sign for presentation in the dictionary view. Definitions can be recorded with different roles, for example *as a noun*, *as a verb or adjective* or *as a question*. In general, the definition can hold any text that should be associated with the sign, so it has also been used for lexicographic notes which would not be shown on the public view.

Each Gloss can have an associated *Video* instance that contains a video showing the sign. The Video model is a simple wrapper around a stored video file, but supports versioning of videos uploaded to the dictionary; new versions of videos can be uploaded but the old versions are maintained

for reference.

To allow further descriptive properties to the dictionary, Signbank supports adding *Tags* to Gloss entries (see examples in Fig. 1). Tags can be arbitrary text strings but in the application, a pre-defined set of tags is defined to give a controlled vocabulary that can be used in a particular instance of Signbank. Tags can be used for different purposes, for example, Auslan Signbank uses tags to define semantic categories such as health, education and sports related signs, but also some phonological and morphological properties. Tags can also be used for dictionary workflow purposes (as in NGT and FinSL Signbank), for example tagging signs that need to have videos recorded or reviewed, or whose phonological description needs double-checking.

2.2. Public and Researcher Views

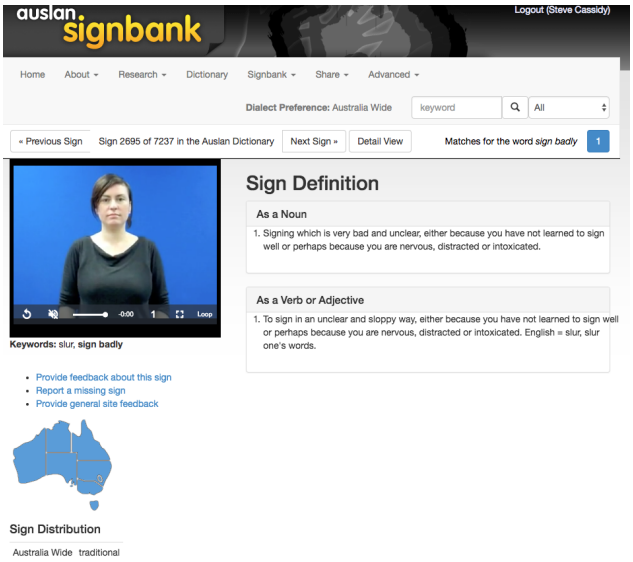


Figure 3: The Auslan Signbank public view for a sign.

The application provides two views of the dictionary; a detailed view for sign language researchers and a more restricted view for the general public.

The public view provides a keyword search facility that allows users to find signs based on their written translations. Each sign is presented as a single page with the sign video featured prominently (Fig. 3). Sign definitions and dialect information is presented on these pages. If sign numbers are used in the dictionary, users can navigate backwards and forwards in the dictionary. If more than one sign matches a keyword, the user can navigate between these matches.

Logged in users can also provide various kinds of feedback on the entries in the dictionary. Feedback can be from the general public, reporting errors or omissions from the dictionary, or from interpreters or researchers providing feedback as part of a review process. Feedback is visible to the editors and publishers of the dictionary.

The researcher view (Fig. 4) is available only to certain users assigned to different user groups (e.g. editors, researchers, interpreters) with various levels of access to browse and update the dictionary. The researcher can view displays the full set of data associated with each sign, including phonological and morphological descriptors, definitions and relations to other signs. If users have permission, they can edit the entry for a sign and upload new video clips.

Researchers also have access to a rich search facility where signs can be located via any aspect of their description. This includes search by keyword, IDGloss, tags and textual search within definitions and notes.

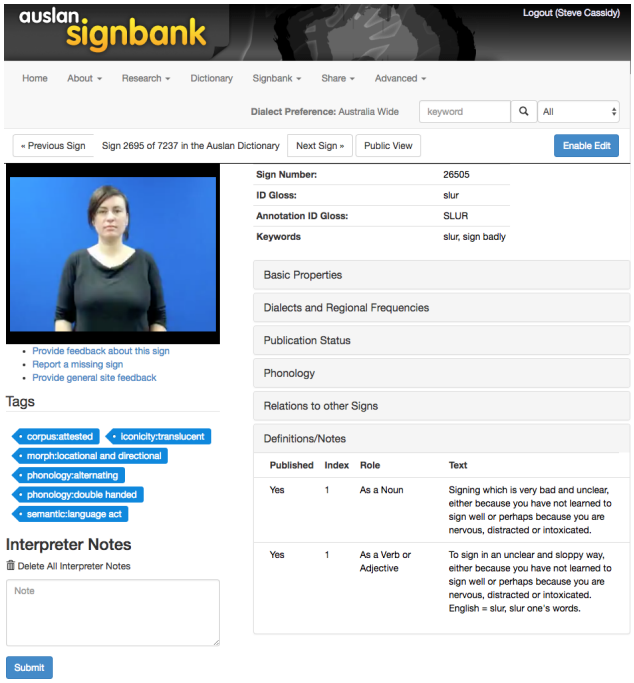


Figure 4: The Auslan Signbank detail (researcher) view for a sign.

3. Open Source Development

As previously mentioned, the release of the Signbank software as open source has allowed other research groups to implement Signbank systems for other languages. These projects have run independently following their own research goals and so have diverged somewhat in the features that each has implemented. Unfortunately, this has meant that there are now three distinct versions of the software with different feature sets; however, the three groups are now planning to collaborate where possible and have established a shared organisation on Github[2] to facilitate this. This section gives an overview of the major new features that have been implemented by the two new projects.

3.1. Datasets

FinSL Signbank introduced and and NGT sigbank later adopted the concept of a *Dataset* as a container for a collection of *Glosses* Fig. 5. This among other things allows a single Signbank instance to be used to store lexicons for multiple languages at once. Now several research groups can work under the same Signbank seamlessly, permissions are controlled per Dataset in order to allow user access to

[2] https://github.com/signbank

the glosses of each corpus lexicon. Datasets are also used to control the publicity of the data, Datasets can be set to be public or private, and the glosses can be marked published in order to show them in the public version of the site.
A Dataset can be associated with a *SignLangauge*, and with multiple written *Language*s in order to select the translation equivalent languages of the glosses of a Dataset.

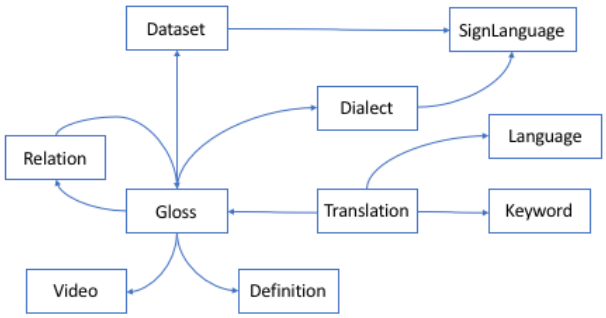


Figure 5: Signbank lexical model extended with Datasets.

This is an important step towards the production of a *Global Signbank* that would be capable of handling a wide range of sign languages in a single lexical database.

3.2. ELAN Export

NGT Signbank implemented and FinSL Signbank later adopted facilities for exporting of lexical information from Signbank to the ELAN annotation tool (Max Planck Institute for Psycholinguistics, 2017) as an external controlled vocabulary (ECV). This can make annotating corpus videos faster and more reliable as annotators are able to easily refer directly to lexical entries from Signbank as they work in the ELAN environment.
On top of that, NGT Signbank has implemented an application programming interface (API) for accessing additional gloss data and gloss media. This API can now be used as a so-called 'lexicon service' in ELAN (from version 5.0.0b on). It lets the user view gloss videos from Signbank in the ELAN annotation panel and display a subset of the fields from Signbank in the Lexicon tab in ELAN. The functionality is explained in more detail in a screen-cast (Crasborn, 2017) and in the ELAN manual.

3.3. Morphology in NGT and ASL Signbank

A number of additional functionalities have been created for NGT Signbank, most of which are also made available in ASL Signbank. These have led to an enriched model of morphology that is represented in Figure 6.
In addition to accounting for sequential compounding by referring to sequences of other glosses, the model accounts for 'simultaneous morphology': bound morphemes like handshapes or locations that have their own form-meaning pairing and that recur in various signs. The phonology section has been extended not just with a number of additional fields at the gloss level (see section 4.), but also with a separate specification of handshapes into their constituent features in terms of the phonological model for NGT (**?**).
Some further linguistic issues related to NGT Signbank are discussed elsewhere (Crasborn et al., 2016).

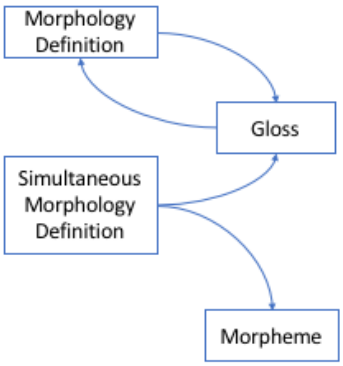


Figure 6: The NGT Signbank model of Morphology.

3.4. Interface changes in NGT

At the level of the user interface, a Variant View has been added, which is a detail view for a sign that lists any relations to other signs: signs with the same Annotation ID Gloss before the suffix (e.g. DOG-A, DOG-B, and DOG-C), which are usually regional variants, and as such, synonyms across regions; signs with a semantic relation (e.g. synonym, homonym, hypernym) specified as a relation to another Annotation ID Gloss; and finally, minimal pairs (which are based on a single field difference within a subset of all the phonology fields). This Variant View facilitates navigation between signs in the database, and helps improve the quality of the phonological description. To further promote the latter, a special warning in the Detail View of a sign highlights forms that have the same phonological description but are not explicitly marked as homophones, and vice versa, signs listed as homophones but which do not have the same phonological description.

3.5. Extensions for FinSL

Development of FinSL Signbank started in May 2015 based on the NGT Signbank. The aim was to make it possible to translate the interface into multiple languages, allow hosting of several corpus lexicons in the same Signbank, and make it possible to export Glosses from Signbank into ELAN (Salonen et al., 2016).
The current version of FinSL Signbank, as of February 2018, has many differences to Auslan Signbank. For instance, the user interface has been made translatable using Django's internalisation and localisation features, currently the interface is translated in Finnish and English. Due to the need for having translation equivalents in several written languages, the translation model holds information about the language. This addition makes it possible to have *Translation* equivalents in multiple *Language*s for one sign, not just in English. The way translations are represented can also be modified, translation equivalents can be grouped according to a set of rules in order to make it possible to have more dictionary like way of representation of translation equivalents.
The public version of the Signbank has been completely remade for FinSL Signbank, with focus on allowing users to access the videos with less effort. The search results list

shows the videos and the videos can be directly viewed on the search page, while the search hits are highlighted on the page.

It is possible to have multiple videos per Gloss, so the users are free to determine their conventions and are not limited to only one video per Gloss. To speed up the annotation process, a feature was added to allow capturing videos directly in the browser via webcam. With this feature annotators can quickly record a video for a new gloss, making it possible for other annotators to recognize the form of the sign from the video.

Other features currently only present in FinSL Signbank include a commenting feature and notifications. Since the introduction of commenting, the users have started organizing the workflow of annotation via Signbank. The commenting is strictly for users using the detailed view. The Auslan Signbank Feedback application also provides the ability to comment, in a slightly different manner. Notifications were added to make it possible to mention other users in comments so that they will surely see them. A separate view for searching *Relations* has been implemented, and relations and reverse relations are also shown on gloss pages. To help with tracking changes to Glosses it is possible to view the changes in an easy to read format directly on the *Gloss* page. Tags are utilized in many places, to make it easier to add the relevant tags to each object, a feature was added to control which tags can be used for which types of objects.

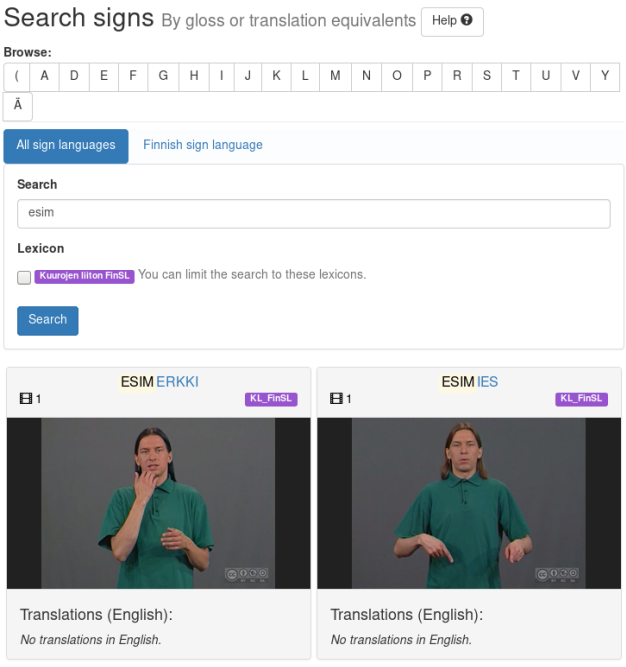


Figure 7: Public search of FinSL Signbank.

3.6. Video Definitions in Auslan

A recent addition to the Auslan Signbank is the ability to associate videos with the definitions of signs. This relatively minor change allows for the first time a sign language native dictionary to be produced in Signbank. Sign definitions can be created and published in the native language of the

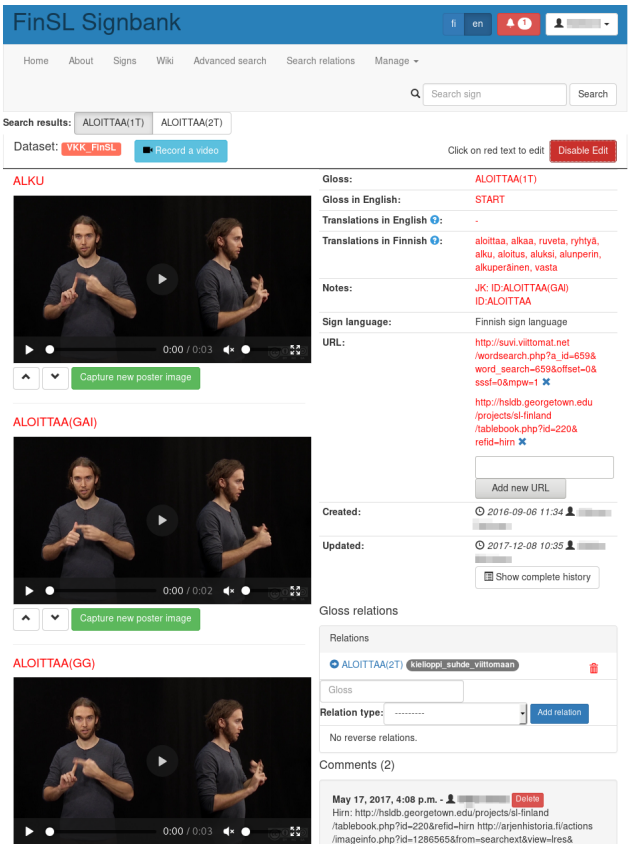


Figure 8: Detailed view of FinSL Signbank with edit mode on.

dictionary, rather than in a written language. Work is now underway to create video definitions for Auslan signs.

4. Areas of divergence in linguistic descriptions

The differing goals of the various research groups that have given rise to the different Signbanks have resulted in some differences in the linguistic description of signs, in particular in relation to the phonological form of a sign. In addition, there are multiple phonological models in the linguistic literature that have resulted in differences in how the phonological form of signs is described (see e.g. (Corina and Sandler, 1993). Where, for the NGT Signbank, one of the aims is to enable research on the phonological system of sign languages, for the Auslan, FinSL, and ASL Signbanks, the central goal is to provide a short phonological characterisation of a sign that enables dictionary users to search for signs by major parameters like handshape and location. These differences inevitably lead to big differences in how many fields are needed to specify the phonological form. In order to make data sets compatible, ongoing work is trying to identify exactly what the overlap between the different ways of describing the form a sign is, and where perspectives differ.

5. Collective Development

Development of the Signbank software has been divergent so far because of the requirements of funding to produce

specific features in different versions of Signbank. This has resulted in three incompatible code-bases with significant overlap and the need to 'port' features already in place in one of the other projects. Over the past year, we have been discussing how the projects might collaborate more closely to ensure that effort can be shared between them rather than duplicated. This has led in the first instance to the establishment of a shared Github organisation[3] that hosts all of the project's source code.

One step towards a more collaborative development model has been the work done on the Auslan Signbank project to modularise the Signbank application. While the original version was a Django project consisting of a number of sub-modules, these were closely linked and inter-connected such that all were needed to build the application. The most recent version of the Auslan Signbank consists of separate applications for the dictionary, video handling and user feedback. Each of these is independent and can be developed and tested on their own. One early goal is to have the three Signbank project share at least the video handling module so that they can take advantage of each other's work in this area - for example the work done by FinSL on in-browser video recording.

One of the biggest areas of difference, as described above, is the representation of linguistic descriptions in each system due to the divergent goals and theoretical positions of the linguists driving these projects. From a software perspective, the challenge is to develop models that would allow any of these forms of description within the same framework. This is challenging in particular because all of the funding for development of this software comes from one or other group. However, the developers are now engaging in discussions about how the projects might collaborate more closely and the increased level of communication can only help this effort.

6. Conclusion

This paper has presented the Signbank software that manages online dictionaries of sign language. Starting from a single system for Auslan, the software now supports a range of languages and has three active development groups. The three groups have diverged somewhat partly based on the linguistic drivers behind their different projects. Future collaboration will hopefully see more shared components being developed to enhance this significant software project.

7. Acknowledgements

Onno Crasborn's contributions to this paper have been supported by grants 360.70.500 and 277-70-014 from the Netherlands Organisation for Scientific Research (NWO) and NIH grant R01 DC 013578.

8. Bibliographical References

Corina, D. and Sandler, W. (1993). On the nature of phonological structure in sign language. *Phonology*, 10:165–207.

Crasborn, O., Bank, R., Zwitserlood, I., van der Kooij, E., Schüller, A., Ormel, E., Nauta, E., Zuilen, M. v., Winsum, F. v., and Ros, J. (2016). Linking lexical and corpus data for sign languages: Ngt signbank and the corpus ngt. In *Workshop Proceedings: 7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining/Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), ISBN 978-2-9517408-9-1*. European Language Resources Association (ELRA).

Crasborn, O. (2017). Innovations in ELAN: linking to a sign language lexicon, Oct. 1, 2017. `https://vimeo.com/225093547`.

Django Software Foundation. (2017). Django framework. `https://djangoproject.com`.

Johnston, T. (2001). The lexical database of Auslan (Australian sign language). *Sign Language and Linguistics*, 4(1/2):145–170.

Johnston, T. (2008). Corpus linguistics and signed languages: no lemmata, no corpus. In O. Crasborn, et al., editors, *5th Workshop on the Representation and Processing of Signed Languages: Construction and Exploitation of Sign Language Corpora*, pages 82–87. ELRA.

Max Planck Institute for Psycholinguistics. (2017). ELAN. `https://tla.mpi.nl/tools/tla-tools/elan/`.

Salonen, J., Takkinen, R., Puupponen, A., Nieminen, H., and Pippuri, O. (2016). Creating Corpora of Finland's Sign Languages. In *Workshop Proceedings: 7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining/Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), ISBN 978-2-9517408-9-1*. European Language Resources Association (ELRA).

9. Language Resource References

Auslan Signbank, `http://www.auslan.org.au/`
BSL Sigbank, `http://bslsignbank.ucl.ac.uk/`
NGT Signbank, `https://signbank.science.ru.nl`
ASL Signbank, `https://aslsignbank.haskins.yale.edu`
FinSL Signbank, `https://signbank.csc.fi`

[3] `https://github.com/signbank/`

J-MeDic: A Japanese Disease Name Dictionary based on Real Clinical Usage

Kaoru Ito, Hiroyuki Nagai, Taro Okahisa, Shoko Wakamiya, Tomohide Iwao, Eiji Aramaki

Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
{kito, hironagai, taro-o, wakamiya, iwao, aramaki}@is.naist.jp

Abstract

Medical texts such as electronic health records are necessary for medical AI development. Nevertheless, it is difficult to use data directly because medical texts are written mostly in natural language, requiring natural language processing (NLP) for medical texts. To boost the fundamental accuracy of Medical NLP, a high coverage dictionary is required, especially one that fills the gap separating standard medical names and real clinical words. This study developed a Japanese disease name dictionary called "J-MeDic" to fill this gap. The names that comprise the dictionary were collected from approximately 45,000 manually annotated real clinical case reports. We allocated the standard disease code (ICD-10) to them with manual, semi-automatic, or automatic methods, in accordance with its frequency. The J-MeDic covers 7,683 concepts (in ICD-10) and 51,784 written forms. Among the names covered by J-MeDic, 55.3% (6,391/11,562) were covered by SDNs; 44.7% (5,171/11,562) were covered by names added from the CR corpus. Among them, 8.4% (436/5,171) were basically coded by humans), and 91.6% (4,735/5,171) were basically coded automatically. We investigated the coverage of this resource using discharge summaries from a hospital; 66.2% of the names are matched with the entries, revealing the practical feasibility of our dictionary.

Keywords: Medical NLP, Case reports, Discharge summary, Named entity, ICD-10

1. Introduction

Medical data are fundamentally important resources for the development of medical AI and information extraction tools. Among various data, Electronic Health Records (EHR) are a promising resource because they include detailed information about a patient and diagnosis processes. Nevertheless, it is difficult to extract information from EHR because several expressions refer to the same concept. Orthographical variations present particular difficulty, especially for the Japanese language, in which characters of at least five kinds are used: Hiragana, Katakana, Kanji, Latin alphabet, and Arabic Numerals. In addition to orthographic difficulties, variations of expression for the same concept delivered by other reasons such as abbreviations are included in medical texts produced at clinics, hospitals, and other medical institutes. These variations present obstacles that are encountered when developing medical AI or information extraction tools, although several studies have been undertaken to solve these and related difficulties by developing or assisting automatic coding systems (Fabry et al., 2003; Yamada et al., 2010; Bouchet et al., 1998).

Actually, several medical resources exist for non-Japanese languages, such as the International Classification of Diseases (ICD; World Health Organization, 2004) [1], Medical Subject Headings (MeSH; Lipscomb, 2000), and Systematized Nomenclature of Medicine Clinical Terms (SNOMED-C[2]; Benson, 2012). SNOMED-CT, the largest, includes approximately 308,000 concepts and 777,000 expressions, and officially supports English and Spanish. Also, projects are translating SNOMED-CT into other languages (Abdoune et al., 2011; Zhu et al., 2012).

Currently, medical language resources for Japanese language are smaller than those for other major languages such as English. The ICD10-based Standard Disease-Code Master (SDCM; Hatano and Ohe, 2003) [3] is the most widely used resource in Japan. The current version of SDCM covers approximately 24,000 disease names. Each name has a corresponding ICD-10 code.

This study was conducted to solve such a problem, developing a dictionary of disease names that appears in medical texts. First, we collected over 45,000 medical case reports from the Japanese Society of Internal Medicine. After we annotated disease expressions in case reports automatically using a named entity recognition (NER) tool to reduce the related work, 13 annotators amended them manually. Next, we split the disease list into three sub-lists: high-frequency, middle-frequency, and low-frequency parts. For the high-frequency part, three human coders manually allocated codes. The middle-frequency part was divided into three subparts: and each of the three coders coded each subpart. For the low-frequency part, we automatically added codes using a classifier trained with the high-frequency part.

Characteristics of the dictionary we developed, the Japanese Medical Dictionary (J-MeDic), are explained below.

- Entries were collected by the Japanese Society of International Medicine and were validated using data obtained from the University of Tokyo Hospital.
- Wide varieties of the expression for an illness are included. The average number of variants for one concept (ICD-10) is 6.74.
- Each entry has information about the corresponding ICD-10. Via ICD-10, Japanese names can be translated into various languages in which ICD-10 is available.

[1] http://www.who.int/classifications/icd/en/

[2] http://www.snomed.org/snomed-ct/

[3] http://www.dis.h.u-tokyo.ac.jp/byomei/index.html

2. Materials

To construct J-MeDic, we used the following three materials for different purposes: ICD-10 and ICD-10-based Standard Disease Code Master for the basis of classification; CR corpus (case reports) for extracting new vocabularies that represent disease names; and HDS corpus (hospital discharge summaries obtained from the University of Tokyo Hospital) for validation of J-MeDic. Both corpora consist of electrical data.

2.1 ICD-10 and ICD10-based Standard Disease Code Master

ICD-10 is the diagnostic classification standard for clinical and research purposes. It is also the international standard for reporting diseases and health conditions. It classifies diseases, disorders, injuries, and other related health conditions (hereinafter, "diseases") in a comprehensive and hierarchical fashion. The ICD code comprises an alphabet and 2–4 digits. The first character of an ICD code is an alphabet called an *axis*, which refers to the kind of disease followed by digits referring to a detailed site. For example, in "C341", the first character "C" refers to "malignant neoplasms". The following two digits "34" refer to "malignant neoplasm of bronchus and lung" together with "C". The last digit "1" means "upper lobe". To match Japanese language expressions with ICD-10 code, we used SDCM, which provides an interface on the website for retrieval of ICD code from natural language and vice versa.

2.2 CR Corpus

To collect names for diseases, we used the CR corpus, an annotated corpus of 44,761 case reports. The Japanese Society of International Medicine provided the reports. After annotating the disease names in the corpus, we extracted them to collect entry candidates for J-MeDic.

2.3 HDS Corpus

The HDS corpus holds discharge summaries from 291,641 patients hospitalized at the University of Tokyo Hospital, Japan, between 2004 and 2016. These summaries, which include brief descriptions of diagnoses, clinical outcomes, comorbidities and observations on admission, and post-admission clinical course, were used for J-MeDic validation: We counted the frequency of the names included in J-MeDic to confirm the extent to which J-MeDic covers the names for diseases of other real clinical texts.

3. Methods

3.1 Structure of J-MeDic

A record in J-MeDic has the following fields:

Name: The expression collected from the corpus, presumed to be used as dictionary entries

ICD code: The ICD code allocated to the name

Standard disease name: The standardized disease name (SDN, disease names collected and standardized in SDCM) allocated to the name

Reliability level: Level of reliability (S, A, B, C)

Kana: Pronunciation of the name, written in Hiragana characters

Site and symptom type: The site at which the disease happens and the type of symptom

Frequency JSIM: Frequency of the name in the data from the Japanese Society of Internal Medicine (i.e. CR corpus)

Frequency UTH: Frequency of the name in the data from the University of Tokyo Hospital (HDS corpus)

For example, Table 1 presents the record of the entry "糖尿病" (diabetes).

Field	Value
Name	糖尿病 (diabetes)
ICD code	E14
Standard disease name	糖尿病 (diabetes)
Reliability level	S
Kana	とうにょうびょう
Site and symptom type	region=膵臓/type=その他 (region=pancreas/type=other)
Frequency JSIM	61,572
Frequency UTH	5,645

Table 1: Record of *diabetes*. English translation added (enclosed in parentheses).

3.2 Entire Process of J-MeDic Construction

J-MeDic was constructed with the following steps.

1. CR corpus annotation
2. Extracting disease names from annotated CR corpus and ICD coding
3. Reliability assessment of the coding
4. Merger with SDNs
5. Evaluation of representativeness

In step 1, we first annotated the words that represent diseases in the corpus automatically; then we manually modified it (Section 3.3). In step 2, candidate entries for J-Medic were extracted from the annotated CR corpus. Then the coders allocated ICD-10 codes (and also the SDN that the allocated ICD accompanies) to these names (Section 3.4). After finding the reliability of the ICD code(s) of each entry (step 3, Section 3.5), SDNs were merged with them (step 4, Section 3.6). Finally, we verified the representativeness of J-Medic using of HDS corpus (step 5, Section 3.7).

3.3 CR Corpus Annotation

To reduce manual annotation work, we automatically annotated disease names that appear in the CR corpus using MedEX/J, which is a tool for analyzing Japanese clinical text. It supports identification and extraction of text strings of potential disease names. Based on conditional random fields, the module learns disease name labels from a disease name annotated corpus. Details of this module are available (Aramaki et al., 2017).
Subsequently, 13 annotators including non-medical workers modified the CR corpus. To construct a resource

for disease names, we set the following criteria and annotated the corpus.

(I) To avoid complexity caused by reference to a disease by split words, the syntactic category of the target was limited to (compound) nouns.

(II) To extract as many candidate disease names as possible, the lexical units were labeled if suspected.

Of those, (I) is important in cases where a disease is referred by a noun, a verb, or other peripheral words. The following three expressions refer to almost identical situations (coded as N289 in ICD-10 system), but which are grammatically different:

(a) **腎機能低下**が見られた (***renal function degeneracy*** *was observed*)

(b) **腎機能**が**低下**していた (***renal function*** *had* ***degenerated***)

(c) **腎機能**が高度に**障害されて**いた (***renal function*** *had severely* ***degenerated***)

In (a), the disease is referred to by a single compound noun (*renal function degeneracy*), whereas the disease is referred by separate words in (b) and (c) (*renal function* and *degenerated in* both cases). In the latter cases, it is difficult to delineate the exact boundary because of its syntactic complexity. Therefore, we limited the syntactic unit of the annotation target.

In addition, (II) is important to maximize the number of disease names that were not yet collected from other language resources. In other words, it is important to expand the vocabulary in J-MeDic. Although technical terms should be standardized, disease names in real case reports are expressed in abbreviated form or in a slightly modified way. In the step of manual annotation, we collected such variations to the greatest extent possible. Then inappropriate variations were excluded from the coding step.

3.4 Coding Procedures

3.4.1 Coders

In this study, three coders coded the data. All the coders had work experience as health care staff.

3.4.2 Collection of Disease names and Manual Coding

Three coders coded high-frequency and middle-frequency parts before automatic coding for the low-frequency part. For the high-frequency part, all coders coded the entire part, discussing it if needed. Each of the three coders coded different subparts for the middle-frequency part. Figure 1 presents a coding process summary.

From the annotated CR corpus, we extracted all the disease names appearing in the corpus. Then, the coders coded disease names in ICD-10. First, we searched the exact matches of the SDNs Master for all the disease names. If a disease name had an exact match, then it was allocated the corresponding ICD of the SDN.

Next, for names that had not been coded in the prior step, the coders searched the exact match of the transliteral variations. For example, a person name "Wegener" appears as it is (i.e. in Latin alphabet) or as various transliteration into Japanese characters such as "ウェゲナー" and "ウェジナー". Therefore, to code "Wegener 肉芽腫" (Wegener's granulomatosis), the coders sought an exact match of "ウェゲナー肉芽腫" and "ウェジナー肉芽腫".

After searching orthographical variants, the coders searched partial matches of the remainder of disease names to avoid extra modifiers. The coders tried queries that are created by omitting modifiers in the name. For example, "LQT2 型 QT 延長症候群" (LQT2 type long QT syndrome) does not match any SDN. In this case, the deletion of "LQT2 type" allows matching of "long QT syndrome". It is coded as I490 (Ventricular fibrillation and flutter). Furthermore, guessing from that "LQT2" refers to a kind of gene, "LQT2 type long QT syndrome" can be categorized as a subcategory of "inherited long QT syndrome", which is listed in corresponding standardized disease name to I490. Therefore, "LQT2 type long QT syndrome" was coded as I490, and standardized as "inherited long QT syndrome".

When multiple ICD codes correspond to a name, we allocated up to two codes. If more than two possible codes were found, then the name was excluded from the targets of ICD coding, and was coded as "-1". In case reports, multiple nouns that represent a disease often appear together to form a compound noun. For example, "脂肪肝合併 2 型糖尿病" (type 2 diabetes complicated with fatty liver) is divisible into "type 2 diabetes" and "fatty liver". Therefore, we allocated both codes to this name. There were also other cases in which multiple codes are allocated to a name: (i) the concept represented by the name is too vague and (ii) the interpretation of the name differs from coder to coder.

If no matched SDN was found after these steps explained

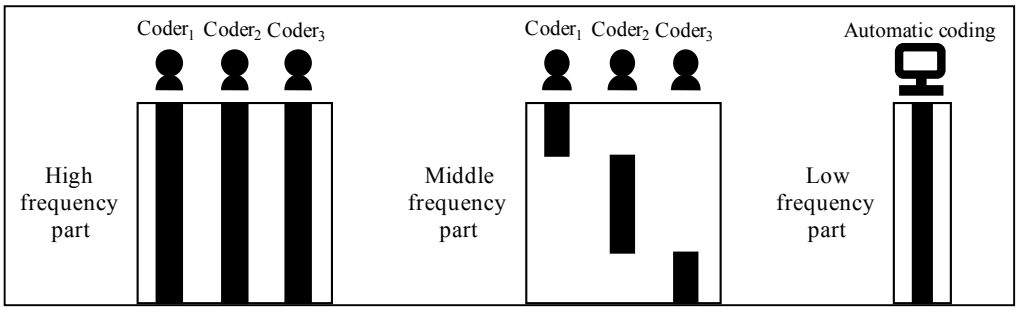


Figure 1: Summary of the coding. A black band represents that the data were coded by the worker indicated above.

above, then the name was coded as "-1".

3.4.3 Comparison and Discussion

Among extracted disease names from the corpus, all the three coders coded the names that appeared more than 29 times (hereinafter, *high-frequency names*). The names appeared more than 3 and fewer than 30 times (hereinafter, *low-frequency names*) divided into three parts and each part was coded by a coder. When a coder found the name difficult to code, the three coders discussed the coding.

The results of coding for frequent names sometimes varied among coders. The final codes for frequent names were decided using the following criteria.

I. When all coders judged a name as not a target, the name was coded as "-1".
II. When all coders allocated the same ICD code to a name, the code was adopted as the final version.
III. When coders allocated a name to different codes, they discussed it and chose a code.

3.5 Reliability Assessment

Based on the coding results, we decided the reliability level of the names depending on how the code was decided. The levels were defined as explained below.

S: Matched with a SDN

A: Three coders allocated the code

B: Coded by one coder, or two coders if discussed

C: Automatically coded, pended, or excluded from the target

3.6 Merger with SDNs

To expand J-MeDic, we merged SDNs with disease names extracted from CR corpus. All the SDN were included into J-MeDic with reliability level "S". Regarding disease names that are extracted from HDS corpus and which are not in annotated CR corpus nor SDN. Disease names collected from HDS corpus were coded automatically.

3.7 Evaluation of Representativeness

To evaluate the representativeness of the entries in J-MeDic, we calculated the coverage of the disease names that appear more than nine times in HDS corpus. The disease names on HDS corpus were extracted using the natural language processing module MedEX/J.

4. Results

4.1 Overview: The J-MeDic size

The J-MeDic covers 7,683 ICD-10 concepts (when a disease name was allocated two ICD codes, the pair was counted as one concept) and 51,784 written forms. Among the written forms, 25,365 (49.0%) forms were not contained in SDCM.

4.2 Matching with SDN

From CR corpus, 30,923 disease names were extracted. Among them, 5,558 names were matched exactly with a SDN and coded as the corresponding ICD-10 of the SDN.

4.3 High-frequency Names

Except for exact-matched disease names with SDN, 804 high-frequency names were found. Table 2 shows the result of coding for high-frequency names by the three coders.

Coding category	# of names (n = 804)
Pended	165
Same code allocated	467
Differently coded	172

Table 2: Primary result of the coding on high-frequency names

We also calculated the inter-rater agreement among the disease names that are not excluded from the target (Table 3). Each coder is represented by c_i.

coder pair	ratio (%)
c_1, c_2, and c_3 allocated the same code	73.1 (467/639)
Only c_1 and c_2 allocated the same code	7.5 (48/639)
Only c_2 and c_3 allocated the same code	7.8 (50/639)
Only c_2 and c_3 allocated the same code	7.5 (48/639)

Table 3: Agreement between coders

4.4 Middle-frequency Names

Except for exact-matched disease names with SDN, 5,319 middle-frequency names were found. Table 4 presents results of coding on low-frequency names.

Coding category	# of names (n = 5,319)
Coded by one coder	4,710
Decided after discussion	3
Pended	606

Table 4: Result of the coding on high-frequency names

4.5 Low-frequency Names

After annotating high-frequency and middle-frequency names, low-frequency names were annotated automatically using backward matching. For each low-frequency name, we found the longest backward match with the high-frequency and middle-frequency names. Then we allocated its ICD (or ICDs, if it has two codes) to the name. Considering the morphological structure of disease names (i.e. most of the head of a compound noun occupies the latter part), we used backward matching.

4.6 Reliability level

Table 5 shows the counts of the reliability level.

Reliability level	**# of names (n = 51,784)**
S	26,419
A	528
B	4,808
C	20,029

Table 5: Reliability level

4.7 Coverage

In the HDS corpus, 17,469 disease names appeared more than nine times. J-MeDic covers 66.2% (11,562/17,469) of these names. Among the names covered by J-MeDic, 55.3% (6,391/11,562) were covered by SDNs; 44.7% (5,171/11,562) were covered by names added from the CR corpus. Among them, 8.4% (436/5,171) were entries

with reliability level A or B (i.e. basically coded by humans), and 91.6% (4,735/5,171) were entries with reliability level C (i.e. basically coded automatically).

5. Discussion

5.1 Extension of the Resource

As described in Section 5.1, J-MeDic contains 51,784 new written forms; 49.0% of those were newly incorporated. However, 44.7% of the disease names that are covered by J-MeDic were newly incorporated written forms. This result can be regarded as indicating that J-MeDic increased the number of the disease names included in a language resource by about 90%.

However, J-MeDic also has limitations. Among newly incorporated disease names that appeared in the HDS corpus, only 8.4% of the names were reliability level A or B, although 21.0% (5,336/25,365) of the disease names in J-MeDic are labeled as reliability level A or B. Because this ratio can differ depending on the corpus, it does not mean directly that the coverage of disease names with reliable ICD code in J-MeDic is low. Therefore, J-MeDic mainly contributed to extension of the entry because disease names of reliability level C are useful to search disease names, although their ICD codes are not reliable. At the same time, J-MeDic can partly be used to detect the particular diseases listed in ICD-10 written in various forms.

5.2 Problem Stemming from ICD

Some difficulties arise stemming from the system of ICD-10. First, because the criteria of the classification in ICD-10 were not clear, coders sometimes had difficulties to search or to identify the ICD code that correspond to a disease name, especially in the Japanese version. For example, N40 (前立腺症, Hyperplasia of prostate) and N429 (前立腺障害, Disorder of prostate, unspecified) respectively correspond to similar disease names, but have different codes. Moreover, because the ICD code for a particular body part sometimes does not exist, the coders had to guess. Furthermore, some orthographical variations caused search difficulties.

5.3 Difference between Coders

Some limitations arose from different opinions among coders. One cause is that coders differently decided if a disease name is a target or not. For example, 転移 (*metastasis*) can be associated with 転移性腫瘍 (C80, Malignant neoplasm, without specification of site), and also 止血困難 (*difficulty in hemostasis*) can be associated with 出血 (R58, Haemorrhage, not elsewhere classified). However, these expressions are a clue to guessing the disease, but not the disease itself. We excluded such expressions from the target.

Another cause is that coders assigned some different codes to disease names. In such cases, the code was generally chosen by majority, but important minority opinions were considered and accepted sometimes. As Section 4.4 showed, up to two codes were allocated to one disease name when selecting only one was difficult.

6. Conclusion and Future Work

We developed J-MeDic, designed for automatic information extraction from medical texts. The newly incorporated words were collected from case reports to improve the coverage of orthographical variations appearing in unstructured texts. We believe that J-MeDic is useful in various fields: Not only can it be used to develop medical AI; it can also help medical workers to write documents using standardized medical terms.

Although we have extended language resources for disease names, numerous names labeled as reliability level D in J-MeDic were coded automatically and were not verified. In future work, further investigations will be necessary to improve the dictionary reliability.

7. Acknowledgements

This work was partly supported by JSPS KAKENHI (JP16H06395, JP16H06399), and by AMED (Grant Number: JP17lk1010019), Japan.

8. Bibliographical References

Abdoune, H., Merabti, T., Darmoni, S. J., and Joubert, M. (2011). Assisting the translation of the CORE subset of SNOMED CT into French. In *MIE* (Vol. 169, pp. 819-823).

Aramaki, E., Yano, K., and Wakamiya, S. (2017). MedEX/J: A One-scan simple and fast NLP tool for Japanese clinical texts. MEDINFO 2017: eHealth-enabled Health.

Benson, T. (2012). SNOMED CT Concept Model. In *Principles of Health Interoperability HL7 and SNOMED* (pp. 253-266). Springer London.

Bouchet, C., Bodenreider, O., and Kohler, F. (1998). Integration of the analytical and alphabetical ICD10 in a coding help system. Proposal of a theoretical model for the ICD representation. *Medinfo.* 9(1):176–179.

Fabry, P., Baud, R., Ruch, P., Le Beux, P., and Lovis, C. (2003). A frame-based representation of ICD-10. *Studies in Health Technology and Informatics*, 95:433–438.

Hatano, K. and Ohe, K. (2003). Information Retrieval System for Japanese Standard Disease-Code Master Using XML Web Service. *AMIA Annual Symposium Proceedings*, 859.

Lipscomb, C. E. (2000). Medical Subject Headings (MeSH). *Bulletin of the Medical Library Association*, 88(3):265–266.

World Health Organization. (2004). *International statistical classification of diseases and related health problems* (Vol. 1). World Health Organization.

Yamada, E., Aramaki, E., Imai, T., and Ohe, K. (2010). Internal structure of a disease name and its application for ICD coding. *Studies in Health Technology and Informatics*, 160(2):1010–1014.

Zhu, Y., Pan, H., Zhou, L., Zhao, W., Chen, A., Andersen, U., Pan, S., Tian, L., and Lei, J. (2012). Translation and Localization of SNOMED CT in China: A pilot study. *Artificial Intelligence in Medicine*, 54(2):147–149.

Building a List of Synonymous Words and Phrases of Japanese Compound Verbs

Kyoko Kanzaki and Hitoshi Isahara
Toyohashi University of Technology
1-1 Hibarigaoka Tenpaku-cho, Toyohashi, Aichi 441-8580
kanzaki@imc.tut.ac.jp, isahara@tut.jp

Abstract

We started to construct a database of synonymous expressions of Japanese "Verb + Verb" compounds semi-automatically. Japanese is known to be rich in compound verbs consisting of two verbs joined together. However, we did not have a comprehensive Japanese compound lexicon. Recently a Japanese compound verb lexicon was constructed by the National Institute for Japanese Language and Linguistics(NINJAL)(2013-15). Though it has meanings, example sentences, syntactic patterns and actual sentences from the corpus that they possess, it has no information on relationships with another words, such as synonymous words and phrases. We automatically extracted synonymous expressions of compound verbs from corpus which is "five hundred million Japanese texts gathered from the web" produced by Kawahara et.al. (2006) by using word2vec and cosine similarity and find suitable clusters which correspond to meanings of the compound verbs by using k-means++ and PCA. The automatic extraction from corpus helps humans find not only typical synonyms but also unexpected synonymous words and phrases. Then we manually compile the list of synonymous expressions of Japanese compound verbs by assessing the result and also link it to the "Compound Verb Lexicon" published by NINJAL.

Keywords: database, Japanese compound verbs, synonymous words and phrases

1. Introduction

In this work, we deal with Japanese verbs and in particular, those of the compound variety. Japanese "Verb+Verb" compounds frequently appear in daily communication which is related to human actions. In terms of morphology, Japanese compounds involving verbs and other predicates are productive and widespread in both syntactic and lexical domains. We have started to construct a database of synonymous expressions of Japanese "Verb + Verb" compounds semi-automatically.

Recently the Japanese compound verb lexicon was constructed by the National Institute for Japanese Language and Linguistics (NINJAL) (2013-15). It has meanings, example sentences, syntactic patterns and actual sentences from the corpus that they possess.

However, it has no relationship information with another words, such as synonymous words and phrases. We try to detect them automatically as much as possible in order to help humans find not only typical synonyms but also unexpected synonymous words and phrases and manually compile a lexicon of synonymous expressions of Japanese compound verbs. We conducted three actions, 1) extract synonymous expressions (words and phrases) from corpus by using word2vec and cosine similarity measure, 2) classify synonymous expressions into several clusters by using the clustering method k-means++ (Arthur and Vassilvitskii 2009) and 3) find a suitable number of clusters corresponding to the meanings of each compound verb by PCA and compile a list of synonymous expressions and their semantic classes by assessing them manually.

2. Compound verbs that we treat

Japanese compound verbs consist of two verbs joined together. The morphological form is a combination of the first verb in an adnominal form and the second verb coming after it, as in *hikari* (adnominal form)-*kagayaku* (give.off.light & shine) 'shine like the sun', *nage* (adnominal form)-*ireru* (throw & put.in) 'throw in'.

These compound verbs are divided into two types in terms of syntactic and morphological analysis ; syntactic compound verbs and lexical compound verbs (Kageyama 1993). Kageyama(1993) says that syntactic compound verbs are easily recognizable and interpretable due to some characteristics, that is, a limitation of the variety of second verbs, no restriction on the first verbs and so on. We exclude the syntactic compound verbs and treat only lexical compounds which tightly combine two verbs as one word and also not productive than syntactic compound verbs

3. Related Researches

So far, NLP domain researches on complexed verbal meaning have treated multi word expressions in order to distinguish a literal meaning with the metaphoric meaning, but their purposes are word sense disambiguation or thegeneration of compounding words(Sag et.al.2002; Katz and Giesbrecht2006; Hashimoto and Kawahara 2008 and so on). In Japanese, Uchiyama and Ishizaki (2003), and Uchiyama and Baldwin (2004) investigated the ambiguities of compound verbs and tried to discover the rules for generation, but the number of compound verbs that they treated was insufficient. As a research on predicative verbs, which was not limited to compound verbs, Izumi et.al (2013) proposed the recognition of semantically equivalent predicate phrases by using definitions in a dictionary and several thesauri as features of verbs. They showed a good result in their experiment, however, in our research, compound verbs that we deal with are not always registered on those lexicons. We have to explore possible methods to find similar expressions of words from corpus.

4. Data

We use "five hundred million Japanese texts gathered from the web" produced by Kawahara et.al. (2006) as corpus for extracting synonymous words and phrases. For compound verbs, we treated compound verbs registered in the "Compound Verb Lexicon (CVL)" created by the National Institute for Japanese Language and Linguistics (NINJAL). The total number of compound verbs in this lexicon is 2700, and each compound verb has meanings, syntactic patterns and example sentences. We also utilized it for an assessment.

5. Extraction of synonym candidates of each Compound Verb

We utilized word2vec (Mikolov, 2013), one of the deep learning methods, for the vectorization of words. The learning model of word2vec that we used is CBOW (Contiguous Bag of Words) and the range of window is five words. We estimate w(t), a word located in position "t" in a sentence, due to two words each before and after w(t) (that is, w(t-2), w(t-1), w(t+1), and w(t+2)).
In our experiment, we vectorized all of the 2700 compound verbs with word2vec by using the five hundred million Japanese web corpus and extracted synonyms of 2700 compound verbs based on this vectorization.

Step (1): preprocessing

In the first trial, we simply put the output of the morphological analyzer JUMAN into word2vec, however, the result was unsatisfactory. Consequently, we decided to utilize syntactic information for the input data. Syntactic information means the case relations between verbs and nouns extracted from web corpus by KNP parser. The sets of a noun, a verb and a case marker consist of the input data for word2vec.
We needed to treat phrasal expressions as "verbs", because some compound verbs can be paraphrased into phrases. We concatenated modification relations between verbs and adverbial words and made units which we treated as "verbs" (e.g. correctly / understand → "correctly understand").

Step (2): vectorization and similarity

We performed then vectorization of all verbs and nouns in the web corpus by using word2vec and explored the semantic distance between verbs (including verbal phrases) by cosine similarity. For each compound verb, the verb and verbal phrases were arranged in descending order from the highest score.

Step (3): list creation

For each compound verb in the CVL, 2000 similar expressions were chosen in order of the highest score of cosine similarity. Here, the lists of synonymous expressions for each compound verb were created. However, in this list, the polysemy of compound verbs was not taken into account. That is, the synonymous expressions of compound verbs were stored together without distinction of polysemous meaning in this list.

6. Getting Clusters for a Compound verb

In order to identify each of the polysemous meanings we classify synonymous expressions for each compound verb by using the clustering method k-means++ and Principal Component Analysis (PCA) (Pearson 1901).
Our procedure is shown in detail in Figure 1 below.

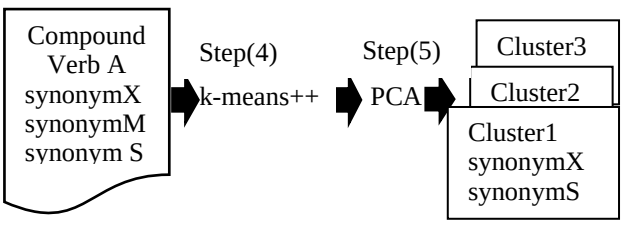


Figure1: the process of obtaining polysemous clusters

Step(4):choosing 100 synonymous expressions

In order to select a suitable clustering method, we compared the results obtained by hierarchical clustering and k-means++. For the hierarchical clustering, we adopted Ward's method. k-means++ is a partial optimization clustering of k-means developed by Arthur and Vassilvitskii (2007), and its initialization method is improved. Comparing the results, these methods both had advantages and disadvantages, and the plausibility of classification of synonymous expressions is similar, however, we chose k-means++ in this work because it found some unexpected expressions.
At the beginning, each compound verb has 2000 synonymous expressions extracted from the web corpus. We needed to narrow down the number of expressions in order to detect the plausible synonymous expressions. Our idea is to decrease the number of expressions in a step-by-step approach by iterating the k-means++. The actual procedure is described below.
1. Firstly, we set 64 clusters for the k-means++. In this stage, 2000 expressions are classified into 64 clusters.
2. For each cluster, we extract 10 expressions with the highest similarity values in the list. In this stage, we narrow down to 640 expressions (64 clusters * 10 expressions).
3. We iterate the same process as step2 for the 640 expressions. In this stage, we set 10 clusters. As a result, we obtained 100 synonym expressions classified into10 clusters (10 clusters * 10 expressions).
This data is then used as the input data for Principal Component Analysis (PCA).

Step (5): Clustering by PCA

We settled the 100 synonymous expressions classified into 10 clusters for each compound verb. However, the number of senses of a compound verb differs from each other .We tried to detect the appropriate numbers of senses of each compound verb by using PCA. We implemented PCA with 100 expressions for each compound verb. We manually found clusters from the result by PCA. In the scatter diagram we extracted clusters clearly separated from the groups of unrelated expressions that were gathered together.

7. Evaluation for 40 compound verbs

In order to predict how many suitable synonyms and clusters semi-automatically obtained by our method, we evaluate our results manually. For 40 compound verbs which are the most frequent compound verbs in our corpus, four examinees evaluated the suitability of the synonymous expressions classified in each cluster.
We evaluated the expressions for each cluster by comparing them to sense descriptions of the compound verb in CVL. As a result, 59% of extracted words are evaluated as synonyms.
Furthermore, we evaluated the suitability of clusters created by our method. We compared the clusters to sense descriptions of the compound verb in CVL. As a result, 65% of extracted clusters are evaluated as representing the proper meaning of the compound verb.
For example, "*Furikaeru* (*Furu*-+*kaeru*)" has a single meaning like "look behind with twisting body" in CVL. Our method could extract another meaning, i.e. "think back on the previous episode."

In terms of recall, the total number of meanings of 40 compound verbs registered in CVL is 64. Among them 14 meanings could not be obtained by our methods (22%). These 14 meanings are included in 13 compound verbs.
Our method could not extract proper clusters for two verbs among 40 verbs, i.e. "*toi-awaseru* (make an inquiry)" and "*sashi-dasu* (holdout)". Most of the candidates of synonymous expressions we extracted for these compound verbs were unsuitable.

8. Manually Making a List of Synomymous Expressons from the Result of PCA

From the result of PCA, we manually classify synonymous expressions into clusters, i.e. drawing circles in Fig. 2.

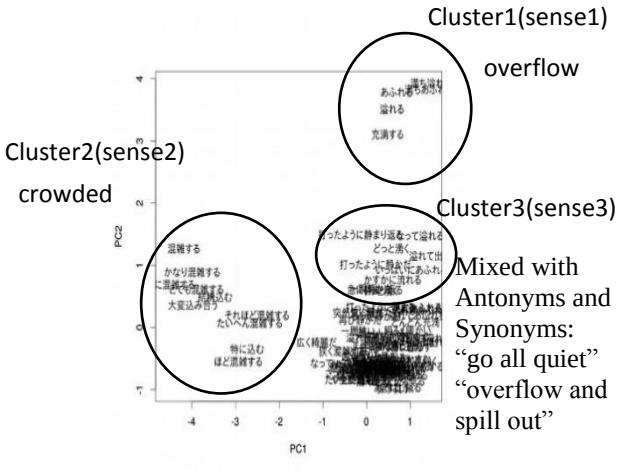


Fig.2. A distribution of synonymou expressions of あふれかえる (*afure-kaeru*) derived from PCA

The list of clusters extracted for あふれかえる (*afure-kaeru*) is shown as follows.

[Cluster 1]

Meaning : overflow
Synonymous expressions :
あふれる (*afureru*, overflow), 満ち溢れる (*michi-afureru*, bubble), 充満する(*juumansuru*, be full of) ...

[Cluster2]

Meaning : crowded
Synonymous expressions :
混雑する(*konzatsusuru*, crowded), かなり込み合う (*kanari komi-au*, very crowded), 特に込む(*tokuni komu*, extreamly congested), ...

[Cluster3]

Meaning : go all quiet
Synonymous expressions :
水を打ったように静かになる (*mizu-wo uttayouni shizukani naru*, become a complete silence as if someone can hear a pin drop.),
かすかに流れる *(kasukani nagareru*, flow faintly)
いっぱいにあふれる (*ippaini afureru*, full of and overflow),
あふれて出る (*afurete deru*, overflow and spill out) ..

Expressions in Cluster3 are a mixture of anotonymous and synonymous expressions. There are antonymous expressions like 水を打ったように静かになる (*mizu-wo uttayouni shizukaninaru*, hush fell over) and かすかに流れる (*kasukani nagareru*,, flow faintly) and also synonymous expressions like いっぱいにあふれる (*ippaini afureru*, full of and overflow) and あふれて出る (*afurete deru*, overflow and spill out). That is, the expressions in Cluster 3 are not separated clearly. When we finally make a list of synonymous expressions, even if the synonymous expressions are classified into an unsuitable cluster, i.e. Cluster 3 in this case, we do not ignore such expressions but move them to a suitable cluster. In this case, we pick up the synonymous expressions いっぱいにあふれる (*ippaini afureru*, full of and overflow) and あふれて出る (*afurete deru*, overflow and spill out) from Cluster3 and move them to a feasible cluster, in this case, Cluster1, manually.
The list for あふれかえる (*afurekaeru*) that we finally obtain is :

[Cluster 1]
Meaning : overflow
Synonymous expressions :
あふれる (*afureru*, overflow), 満ち溢れる (*michi-afureru*, bubble), 充満する (*juumansuru*, be full of), いっぱいにあふれる (*ippaini afureru*, full of and overflow), あふれて出る(*afurete deru*, overflow and spill out)

[Cluster2]
Meaning : crowded
Synonymous expressions :
混雑する(*konzatsusuru*, crowded), かなり込み合う (*kanari komi-au*,very crowded),特に込む (*tokuni komu*, extremely congested), ...

[Cluster3]
Meaning : go all queit
Synonymous expressions :
水を打ったように静かになる (*mizu-wo uttayouni shizukani naru*, become a complete silence as if someone can hear a pin drop), かすかに流れる (*kasukani nagareru*, flow faintly)

9. Compound Verb Lexicon compiled by NINJAL

The list that we are now compiling will be linked to CVL in order to cover meanings that we fail to find and also extend the CVL.
Compound Verb Lexicon (CVL) was constructed by the National Institute for Japanese Language and Linguistics (NINJAL) (2013-15). It says that comprising over 2,700 verb-verb compound verbs commonly used in contemporary Japanese, the Compound Verb Lexicon is designed to provide both researchers in linguistics and foreign learners of Japanese with useful information on their grammatical, semantic, and other linguistic features. In addition to Japanese representations, it offers English, Chinese, and Korean translations for the semantic definitions and example sentences.
As an example, the description of あふれかえる (*afurekaeru*) in CVL is as follows.

Meaning:

Japanese: 場所の収容力以上に、いっぱいである.
English: (Of a place) To be full beyond capacity.
Chinese : 远超过场所可容纳的量。［意译：爆满］
遠超過場所可容納的量。［意譯：爆滿］

Korea : 장소의 수용력 이상으로 가득차 있다.

Example :

駅前にはタクシー待ちの人たちがあふれ返っていた.
Ekimae-ni-wa takushiimachi-no hitotachi-ga afurekaette ita.
English : There were huge crowds of people waiting for taxis in front of the station.
Chinese :车站前挤满了等出租车的人
車站前擠滿了等計程車的人。
Korea : 역앞에는 택시를 기다리는 사람들이 넘쳐났다.

Syntactic pattern :

N1-*ni* N1-*ga afurekaeru*
N1-*ga* N2-*de afurekaeru*

10. Link our List of Synonymous Expressions to CVL

In our list, each cluster (cluster1, cluster2,..) corresponds to one of the meanings of each compound verb, therefore, in our list あふれかえる (*afurekaeru*) has two meanings (cluster1 and 2) and one antonymous meaning (cluster3). In CVL, this compound verb has one meaning, "(Of a place) to be full beyond capacity". CVL would be bundling cluster1 and cluster2 in our list by the same core meaning. However, in our result, synonym expressions in cluster 1 and those in cluster2 are clearly divided because they are used in a different context by figurative meaning. Actually, "people are overflowing at the station" is acceptable in Japanese, however, "water was crowded" is not acceptable. In our list, we keep two meanings, cluster1 (overflow) and cluster2 (crowded) and link cluster1 to the meaning "(Of a place) to be full beyond capacity" of あふれかえる (*afurekaeru*) in CVL and, on the other hand, add Cluster2 as a new meaning.

11. Future work

We try to compile a list of synonymous expressions for about 2700 compound verbs registered in CVL and link to CVL so that we can find what kind of paraphrases compound verbs have.

If we try to adopt our method to the construction of a large scale lexicon, it will consume significant time and human effort because the final step described in section 8 is manually conducted. In fact, it takes two people about one month to fully evaluate the result. For this task, the number of Japanese " Verb + Verb" compounds that we call "lexical compound verbs", not "syntactic compound verbs", is limited and they are not productive. Japanese VV lexical compound verbs are said to be about 3000 words, therefore our method can adopt to our task. For future work, although our result is interesting and is not bad, we intend to try another automatic method for the final clustering step instead of PCA.

Japanese compound verbs are often tough obstacles for beginning learners of Japanese to work through. By linking synonym expressions to CVL, we would like to contribute to linguistic, NLP, and Japanese language education.

12. Bibliographical References

David Arthur, Sergei Vassilvitskii. 2007. k-means++: The Advantages of Careful Seeding. In Proceedings of the 18th annual ACM-SIAM symposium on Discrete algorithms, 1027-1035.

Katz Graham, and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties, Association for Computational Linguistics, 12-19.

Chikara Hashimoto and Daisuke Kawahara. 2008. Construction of an idiom corpus and its application to idiom identification based on WSD in corporating idiom-specific features. In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing(EMNLP 2008). 992-1001.

Tomoko, Izumi., Tomohide Shibata, Kuniko Saito, Yoshihiro Matsuo and Sadao Kurohashi. 2013. *Recognizing Semantically Equivalent Predicate Phrases based on Several Linguist Clues*. Journal of Natural Language Processing, 539-561.

Taro Kageyama(1993), *Bunpō to Gokeisei* [Grammar and Word Formation], Tokyo: Hituzi Syobo

Daisuke Kawahara and Sadao Kurohashi. 2006. A Fully-lexicalized Probabilistic Model for Japanese syntactic and Case Structure Analysys. In Proceedings of Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLT-NAACL 2006), NY, USA, 176-183.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In proceedings of 27th Annual Conference on Neural Information Processing Systems,3111–3119.

Ivan A.Sag, Timothy Baldwin, Francis Bond, Ann Copestake and Dan Flickenger. 2002. Multiword Expressions: A pain in the Neck for NLP. In CICLing '02 Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing, 1-15.

Kiyoko Uchiyama and Shun Ishizaki. 2003.The Method on the Semantic Analysis for disambiguation of compound verbs. In proceedings of the 9th annual conference of Natural Language Processing,163-166.

Kiyoko Uchiyama, Timothy Baldwin,.2004. Automatic Disambiguation of Compound Verbs by Machine Learning. In proceedings of the 10th annual conference of Natural Language Processing, 741-744.

13. Language Resource References

National Institute for Japanese Language and Linguistics. Compound Verb Lexicon.(2013-2015) http://vvlexicon.ninjal.ac.jp/en/

Acknowledgements

This work was supported by JSPS KAKENHI（Grant-in-Aid for Scientific Research (C)） Grant Number JP 16K02727.

Evaluating EcoLexiCAT: a Terminology-Enhanced CAT Tool

Pilar León-Araúz, Arianne Reimerink
University of Granada
Department of Translation and Interpreting, c/Buensuceso, 11 18002 Granada (Spain)
{pleon, arianne}@ugr.es

Abstract

EcoLexiCAT is a web-based tool for the terminology-enhanced translation of specialized environmental texts for the language combination English-Spanish-English. It uses the open source version of the web-based CAT tool MateCat and enriches a source text with information from: (1) EcoLexicon, a multimodal and multilingual terminological knowledge base on the environment (Faber et al., 2014; Faber et al., 2016); (2) BabelNet, an automatically constructed multilingual encyclopedic dictionary and semantic network (Navigli & Ponzetto, 2012); (3) Sketch Engine, the well-known corpus query system (Kilgarriff et al., 2004); (4) IATE, the multilingual glossary of the European Commission; and (4) other external resources (i.e. Wikipedia, Collins, Wordreference, Linguee, etc.) that can also be customized by the user. The tool was built with the aim of integrating terminology management – often considered complex and time-consuming – in the translation workflow of a CAT tool. In this paper, EcoLexiCAT is described along the procedure with which it was evaluated and the results of the evaluation.

Keywords: computer-assisted translation; terminology management; specialized translation

1. Introduction

Currently, computer-assisted tools (CAT) do not seamlessly integrate terminology management – often considered complex and time-consuming – into the translation workflow. Furthermore, most terminological resources do not take into account the real search behavior of translators (Tudhope et al., 2006; Durán Muñoz, 2012: 78). Most terminological modules in CAT tools do not go beyond a simple glossary of source and target terms and access to corpora is rarely, if ever, provided. This leads to an inevitable loss of translation quality and a waste of precious time.

To fill this void, we developed EcoLexiCAT (León-Araúz et al., 2017), a terminology-enhanced CAT tool that provides easy access to domain-specific terminological knowledge in context. This application integrates different features of the professional translation workflow in a stand-alone interface where a source text is interactively enriched with terminological information (i.e. definitions, translations, images, compound terms, corpus access, etc.) from different external resources: (1) EcoLexicon, a multimodal and multilingual terminological knowledge base (TKB) on the environment (Faber et al., 2014: Faber et al., 2016); (2) BabelNet, an automatically constructed multilingual encyclopedic dictionary and semantic network (Navigli and Ponzetto, 2012); (3) Sketch Engine, the well-known corpus query system (Kilgarriff et al., 2004); (4) IATE, the multilingual glossary of the European Commission; and (5) other external resources (i.e. Wikipedia, Collins, Wordreference, Linguee, etc.) that can also be customized by the user.

The motivations behind the integration of these resources lie in the needs and expectations of translators regarding terminology management. According to Durán Muñoz (2012: 82), translators consider that the most important ISO fields (ISO 12630: 1999) in the microstructure of terminological resources are the following: clear and concrete definitions, equivalents, derivatives and compounds, domain specification, examples, phraseological information, definition in both languages for bilingual resources and abbreviations and acronyms.

Likewise, translators believe that terminological resources should be able to (1) permit exportability and/or importability in different formats; (2) include more pragmatic information about usage and tricky translations (old usage, false friends, specific usage in a domain or region, etc.); (3) offer links to other resources to improve or increase results; (4) improve search options; and (5) provide examples taken from real texts. Quite surprisingly, although the translators in this study (Durán Muñoz, 2012) did not show much interest in having access to corpora, they did highlight the need for more phraseological information, pragmatic information and examples taken from real texts. Even though this information can be extracted from corpora, translators were probably reticent to use them because it can take a long time if the right query methods are not provided.

EcoLexiCAT takes into account all of the above and includes the essential fields mentioned, links to other resources and improved search options for corpus analysis that provide the necessary pragmatic information and real text examples. Furthermore, it is a single-platform web-based CAT environment that has the capabilities of importing and exporting different file types and formats. Therefore, the next logical step is to evaluate the functionalities and performance of the tool based on the experience of prospective users in order to assess whether it meets the expectations of professional translators.

The remainder of this paper is organized as follows. Section 2 describes the web-based open source CAT tool MateCat on which EcoLexiCAT is based as well as the external resources used for terminology enhancement. Section 3 provides a description of EcoLexiCAT. Section 4 shows the procedure with which EcoLexiCAT was evaluated and the results of the evaluation. Finally, Section 5 presents the conclusions and future research.

2. EcoLexiCAT sources

2.1 MateCat

MateCat, acronym of Machine Translation Enhanced Computer Assisted Translation, was originally aimed at improving the integration of machine translation (MT)

and human translation (Federico et al., 2014: 129). This application is not only an industrial tool but also an open source platform[1]. The fact that it has an open-source version as well as its high level of flexibility made it a suitable option for the development of EcoLexiCAT. In addition, the features and operation of MateCat are basically the same as those found in most CAT tools used nowadays, such as a text editor that divides the text to be translated in source and target segments and saves them along with their translation in a translation memory (TM). Moreover, it supports 59 different types of source files.

MateCat runs as a web server and communicates with other services through open APIs. It allows communication with pre-existing TMs and the collaborative TM MyMemory, terminological databases, concordance searches within the TMs and machine translation (MT) engines, from which the MT provider, also named MyMemory (a combination of Google Translate and Microsoft Translator), is freely available. Therefore, professional translators will not need to invest much time in learning how to use the tool and will benefit from the interoperability of CAT-related formats (TBX for glossaries, XLIFF for bilingual files, TMX for TMs, etc.). This enables them to use the resources generated during the translation process in other similar tools and reuse pre-existing resources (i.e. glossaries, bilingual files and TMs) in EcoLexiCAT.

2.2 EcoLexicon

EcoLexicon[2] is a multilingual and multimodal terminological knowledge base on environmental science (Faber, León-Araúz & Reimerink 2014; 2016). It is the practical application of Frame-based Terminology (FBT; Faber et al., 2011; Faber, 2012, 2015), a theory of specialized knowledge representation that uses certain aspects of Frame Semantics (Fillmore, 1982; Fillmore & Atkins, 1992) to structure specialized domains and create non-language-specific representations.

EcoLexicon currently has 3,601 concepts and 20,211 terms in Spanish, English, German, French, Modern Greek, and Russian. Regarding the languages included in EcoLexiCAT, EcoLexicon currently contains 5,290 terms in English and 4,898 terms in Spanish. This terminological resource is conceived for language and domain experts as well as for the general public. It targets users such as translators, technical writers, and environmental experts who need to understand specialized environmental concepts with a view to writing and/or translating specialized and semi-specialized texts. The resource contains definitions, semantic networks, equivalents, images, phraseological information and access to corpus information

2.3 BabelNet and Babelfy

The multilingual encyclopedic dictionary and semantic network BabelNet[3] was created by integrating the lexicographic and encyclopedic knowledge contained in WordNet and Wikipedia (Navigli & Ponzetto, 2012: 218). It connects concepts and named entities in a network of semantic relations, made up of about 14 million entries, called Babel synsets. Each Babel synset represents a given meaning and contains all the synonyms expressing that meaning in a range of different languages. Wikipedia and WordNet are integrated through automatic mapping and by filling in lexical gaps in resource-poor languages with MT.

BabelNet is an enormous information resource that can be accessed through an open API, and was considered to be a valuable addition to EcoLexiCAT in those cases where EcoLexicon, a manually-built resource, did not include sufficient information or information regarding general language issues or for texts that combine environmental issues with other domains of expertise. Furthermore, the BabelNet researchers created their own algorithm, called Babelfy, for the disambiguation of polysemic words when found in the context of a particular text (Moro, Raganato & Navigli, 2014; Moro, Cecconi & Navigli, 2014). In EcoLexiCAT, the source text is disambiguated through Babelfy before matching the terms with BabelNet.

2.4 Sketch Engine

Sketch Engine (Kilgarrif et al., 2004) is an online corpus query system with a very efficient search engine and a statistical component for enhanced precision. It contains over 300 corpora in over 60 languages and allows end users to create their own corpora as well. One very interesting module is information extraction through word sketches. Word sketches are summaries of collocational information of a search term, where the term is analyzed according to the verbs, modifiers and other usual constructions that accompany it in real texts. Word sketches are created through sketch grammars that launch specific queries to a corpus. End users can create their own grammars for word sketches and therefore adapt the tool to their specific needs.

Sketch Engine also hosts a set of freely available open corpora that can be queried with full Sketch Engine functionalities with no need of having a subscription. This made a perfect option for EcoLexiCAT, since the English EcoLexicon Corpus was uploaded as an open corpus[4] and can be accessed from EcoLexiCAT through its API.

2.5 IATE

IATE, Inter-Active Terminology for Europe, is the UE's inter-institutional terminology database that has been used in the EU institutions and agencies since 2004, enhancing standardization and promoting an official EU terminology. It has around 8.4 million terms. IATE cannot be accessed through APIs but can be downloaded[5]. Therefore, we downloaded the set of English and Spanish terms and stored them in a database to interact with EcoLexiCAT as a fourth external resource.

3. EcoLexiCAT: a terminology-enhanced CAT tool

When users start a new project in EcoLexiCAT they first access the project settings interface where they can do the following: (1) name the project; (2) choose directionality (so far, English-Spanish or Spanish-English); (3) select a

[1] https://www.matecat.com/open-source/
[2] ecolexicon.ugr.es
[3] babelnet.org
[4] https://the.sketchengine.co.uk/open/
[5] http://iate.europa.eu/tbxPageDownload.do

particular domain within the environment – these are in consonance with the domains according to which EcoLexicon is organized and are included in this first step as a way to classify projects and TMs for later reuse; (4) choose between general and patent segmentation rules, for the source text to be segmented accordingly; (5) optionally add an MT provider for post-editing – MyMemory is freely available, but others (e.g. Moses, DeepLingo, IP Translator) can also be added if users have an account with them; (6) optionally add users' own TMs and/or glossaries – otherwise a collective TM stored in the system will be used; and (7) upload the source text. These steps, except for (3), are default options in MateCat.

Once the source text is processed and converted into a bilingual format (XLIFF), users can access the main interface (Figure 1), which is divided into two main sections. The left-hand section is where the four external resources (i.e. EcoLexicon, BabelNet/Babelfy, Sketch Engine and IATE) provide the terminological enhancement of the translation process. The right-hand section is where the target text is produced, an editor where the source text appears split into different segments.

In the right upper part of the editor, users may download the target or the source text in their original format, and export the bilingual file in SDLXLIFF (SDL Trados Studio's native format) or the whole project in OmegaT's native format, another desktop open source CAT tool. This, together with the possibility of downloading the TM and the glossary created during the project, ensures the interoperability of different formats across different CAT tools, an issue that professional translators must often deal with.

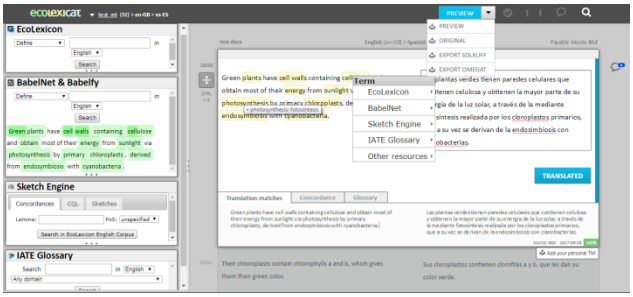


Figure 1: User's interface of EcoLexiCAT.

Figure 2 shows a segment within the editor, which offers the usual editing features of any CAT tool, and how the terms from EcoLexicon are recognized (highlighted in yellow). Users can also split or merge segments, copy the source text in the target segment, benefit from a QA (quality assurance) system that detects missing spaces or tags, create on-the-fly glossary entries, search for concordances within the TM and get suggestions from previously stored segments in the TM or, if added, from an MT engine. Once a segment is confirmed, it is stored in the users' TM or in the collaborative TM, from which other users can benefit. This converts the tool into a collaborative environment. Of course, translators working under confidentiality agreements should always choose their own TM, which will prevent their segments from getting stored in the collaborative TM.

However, the difference between an ordinary CAT tool and EcoLexiCAT is that EcoLexiCAT is a terminology-enhanced translation tool. This means that the editor interacts with external terminological resources that can assist the translator during the different phases of the translation workflow. First of all, the source segment is enriched with information from EcoLexicon. As shown in Figure 2, all matching terms are highlighted in yellow, and users can interact with them in three ways: (1) if they hover the mouse over them, all possible translations (equivalent terms and synonyms) are displayed in an emerging box; (2) if they click on any of them, the EcoLexicon box of the left-hand side shows both the translations and the definition; and (3) if they right-click on any of them, a scroll-down menu gives access to all the different options provided by each of the resources of the left-hand section, sending ready-made queries based on the selected term.

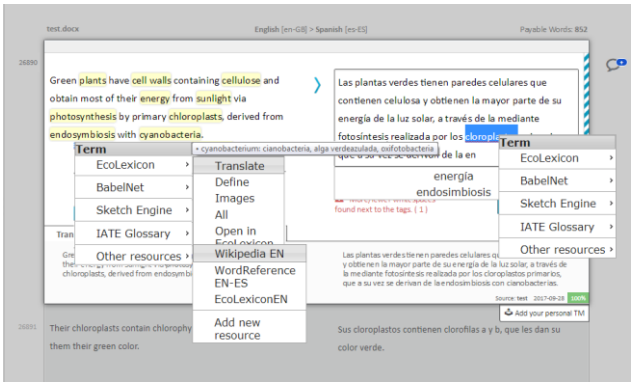


Figure 2: EcoLexiCAT editor.

In the case of EcoLexicon, these options correspond to the data categories in the TKB that usually serve for text comprehension: translations, synonyms, definitions, and images. Also from this menu, a new tab can be opened in the browser to access the EcoLexicon TKB for a more detailed analysis of the conceptual networks.

In the BabelNet & Babelfy box the source text, previously disambiguated by the Babelfy algorithm, is also matched against the BabelNet network. This enables the system to propose statistically relevant candidate translations, which is a significant advantage taking into account that BabelNet covers any specialized or general domain and ambiguity can be frequently encountered.

All matched terms are highlighted in green and behave in the same manner as the terms in the source segment with regard to EcoLexicon (hovering the mouse, clicking and right-clicking). The options available from BabelNet, after right-clicking, correspond to the data categories that have been considered most interesting for translators: definitions, translations, compound words and images. Also, from the definitions option, a new tab can be opened in the browser to access the semantic networks in BabelNet.

The Sketch Engine box can be used to select a term from both the source and target segments and analyze its behavior in the EcoLexicon corpus. So far, only the EcoLexicon English Corpus is hosted in Sketch Engine Open Corpora. The EcoLexicon Spanish Corpus is still in the compilation phase. The corpus can be queried through basic or CQL queries as well as through word sketches. Word sketches are based on the default sketch grammars offered by Sketch Engine but also on a set of customized sketch grammars specifically built for the extraction of semantic relations (León-Araúz et al., 2016).

Corpus information can be very useful during the text production phase (e.g. searching for modifiers or verbs that collocate with a particular noun, looking for synonyms or frequent syntactic structures, etc.). However, corpora can also help translators to understand how concepts interrelate with each other within the domain (hence, the customized semantic word sketches).

For this reason, corpus queries are enabled from both source and target segments. The output of the queries can be opened in a new tab that sends users to the website of Sketch Engine Open Corpora for a more detailed analysis. In this way, they can use all the functionalities of the tool (e.g. Context, Word list, Thesaurus, Sketch Diff, etc.) and make more specific queries filtered by the features according to which the corpus is tagged (i.e. year, genre, contextual domain, user type and linguistic variant).

The IATE box gives access to the information that the downloadable dump provides (equivalents and domains). The queries can be limited by domain and the system delivers all possible equivalents in both directionalities.

In turn, the target segment is enriched with a predictive typing feature. As soon as users start typing a word that has been matched as the translation of one of the terms in the source segment, all possible translations are shown in a drop-down list. In addition, as in the source segment, users can right-click on any term they type in the target segment and send queries to the four resources in the opposite language directionality.

Finally, from the scroll-down menu that can be accessed from both source and target segments, a last option is provided to interact with a customized list of other external resources that can be expanded by the user. This appears as an emerging window based on the URLs of the resources that users may usually consult, such as WordReference, Wikipedia, Linguee, etc. This works as the SDL Trados Studio plug-in Web Lookup or the MemoQ web search feature.

4. Evaluating EcoLexiCAT

4.1 Experimental setup

EcoLexiCAT was evaluated by comparing the behavior and products of two subject groups, one using EcoLexiCAT and the other one acting as a control group and using MateCat. Both groups were made up of students from the Master's Degree in Professional Translation of the Faculty of Translation and Interpreting of the University of Granada (Spain).

Prior to the translation task, participants of both groups were asked to fill out a brief questionnaire in order to collect data about their professional/training background, their expectations of terminological resources and CAT tools and their habits regarding the use of dictionaries, corpora, terminological resources, etc. when confronted with a translation assignment.

Then, the group using EcoLexiCAT was made acquainted with the tool, so that subjects did not waste their time getting used to its specific functionalities. Both groups were already familiar with MateCat. For this experiment, the collaborative memory option nor machine translation were used.

Afterwards, subjects were presented with a translation task consisting of two short specialized translation assignments, one English-Spanish and the other Spanish-English, of extracts of scientific papers on the topic of coastal engineering, a domain widely covered in EcoLexicon. The reason for having chosen both directionalities is, firstly, to see if behavior and results change according to directionality; secondly, because the only corpus available so far is the EcoLexicon English Corpus and usage examples are usually demanded during the text production phase.

Students were required to deliver publishable texts in two hours. Therefore, the length of each source text was under 200 words (EN-SP 194 and SP-EN 168 words). Other features of the source texts were high term density, syntactically complex sentences and collocational specificities that called for both a deep understanding of domain knowledge and expression. Students were thus confronted with several challenges both during the comprehension and production phases of the translation workflow.

Both groups were asked to note down all the problems encountered and the resources that helped them solve each problem. The EcoLexiCAT group was allowed to use resources outside EcoLexiCAT only if they did not find the answer within the tool.

Finally, after finishing the assignments, EcoLexiCAT users filled out another questionnaire on the tool's usability, functionality and efficiency, three main parameters established by the ISO 9128 standard for software product evaluation. They were also asked to highlight the problems encountered and propose possible improvements.

The purpose of this evaluation was two-fold. We intended to assess user satisfaction but also user performance. The first parameter was assessed based on the answers given by the EcoLexiCAT user group in the last questionnaire, whereas the second parameter was assessed by comparing the time employed and the average quality of the target texts delivered by both groups. Quality assessment was based on a scale where both translation and linguistic errors and wise choices are accounted for. The editing log of EcoLexiCAT and MateCat was used to see how long students took to translate each text.

4.2 Results and discussion

A total of 19 students, aged from 22 to 37, were included in the evaluation, 10 in the EcoLexiCAT group and 9 in the control group. All subjects except for one were native speakers of Spanish, 11 subjects have English as their first foreign language and 5 as their second foreign language. One subject is a native speaker of both English and Spanish and 2 did not include English as one of their official working languages during their undergraduate degree, but did have sufficient proficiency. The large majority has a translation degree (84%), the others have degrees in modern languages or related areas. Only four subjects mentioned previous professional translation experience. The different characteristics were evenly divided over both experimental groups.

4.2.1 User expectations

In the first questionnaire, the participants were asked to classify the following features in CAT tools as essential, desirable or unnecessary: access to MT engines, access to corpora, interoperable file formats, access to terminological resources, access to terminological resources defined by users, and QA and revise options. The results in Figure 3 show that, according to the subjects, the most important features are format interoperability, terminological resources and QA revise options. Access to corpora was regarded essential and desirable in the same proportion, whereas access to MT engines was deemed only desirable. This might be due to the fact that post-editing of MT is still not widely accepted by the translators' community. MT engines, access to corpora and user-defined terminological resources were the only features that were unnecessary according to a few participants.

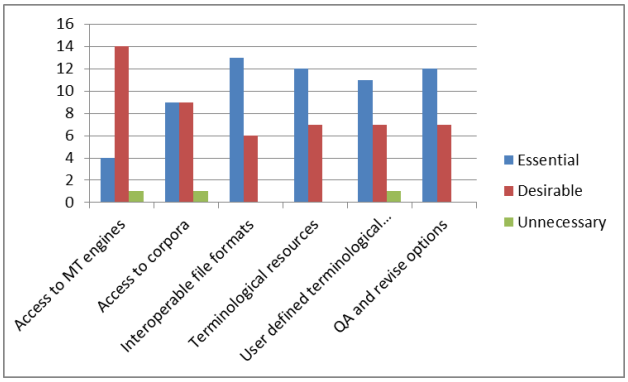


Figure 3: Users's expectations about CAT tools.

When asked about other features not included in the above list, most students could not identify any other feature that they would consider relevant in CAT tools. Exceptions were image editors and customizable QA rules.

Participants were also asked to do the same with a set of data categories usually included in terminological resources. The data categories were: definitions, translations, synonyms and variants, context and usage examples, conceptual relations, register, images, phraseological and collocational information, etymology, pronunciation, compounds and derivatives, part of speech, pragmatic information on term usage, and access to corpora.

The results in Figure 4 show that definitions, translations, context and usage examples, and access to corpora are the most relevant data categories. Among desirable categories, conceptual relations, register, images and compounds and derivatives stand out. Etymology and pronunciation are the categories most often regarded as unnecessary.

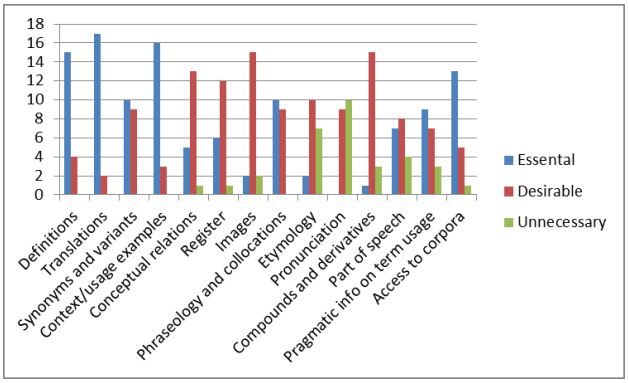


Figure 4: Users's expectations about terminological resources.

When asked about other features not included in the above list, most students could not identify any other feature that they would deem relevant in terminological resources. Exceptions were specialized reference works and term use frequency. Among the resources that students use the most for their translation assignments, the following stand out: Wordreference, Linguee, Reverso Context, IATE, Merriam-Webster, Oxford dictionaries, Collins, esTenTen and enTenTen corpora in Sketch Engine, the BNC, CREA, the web as a corpus, Pons and Termium Plus.

The analysis of subjects' answers indicates that EcoLexiCAT meets most of users' needs and expectations, but it also points to how to improve the tool and even EcoLexicon. For instance, currently there is a phraseology module under construction in EcoLexicon that will undoubtedly be linked to EcoLexiCAT in the future.

4.2.2 User performance

All target texts were evaluated by one reviser to ensure the same criteria were applied in all cases. To assess the quality of the target texts of both groups, ten translation problems were identified for both the English-Spanish and the Spanish-English assignment. The problems identified were based on those that the students mentioned repeatedly and on the reviser's expertise in the text type and domain. Depending on how well the students solved these problems they could obtain up to 10 translation points. On the other hand, the language errors in both Spanish and English were deducted from a maximum grade of 10. The final grade was then the average between the translation points obtained and the linguistic quality of the target text (see Table 1).

For example, one translation problem of the English-Spanish assignment was finding the correct terminological equivalent in Spanish for the different types of current (longshore, tidal and rip current). Another problem was understanding the exact location of a groyne in "perpendicular or slightly oblique to the shoreline

extending into the surf zone (generally slightly beyond the low water line)". An example of a translation problem in the Spanish-English assignment was understanding that "bocana" and "desembocadura" are synonyms and can both be translated as "river mouth".

The EcoLexiCAT group outperformed the control group in both directionalities, although only slightly in the SP-EN assignment. The average quality of the target texts of both groups was not very high. This is understandable because most subjects of both groups did not have any professional translation experience or previous knowledge about the environmental domain. The results are promising, though, as EcoLexiCAT helps to obtain a better target text in less time.

	EcoLexiCAT EN-SP	Control EN-SP	EcoLexiCAT SP-EN	Control SP-EN
	6.8	6.1	6.9	7.1
	4.5	3.9	7.5	6
	8.4	7.6	8.1	6.6
	7.4	3.9	6.4	5
	8.0	4	7.1	4.6
	7.8	4.8	5.4	6.5
	7.4	6.3	6.1	7.3
	6.5	6.5	4.5	6.6
	7.6	6.1	7.6	6.1
	5.3		4.5	
Average	6.9	5.5	6.4	6.2

Table 1: User's interface of EcoLexiCAT.

It is also interesting to point out that the control group used very similar resources to solve the translation problems to those included in EcoLexiCAT: EcoLexicon, BabelNet, Wordreference, IATE, Linguee, and Wikipedia.

In terms of the time invested, in both directionalities the EcoLexiCAT group outperformed the control group. Suprisingly, the EcoLexiCAT group took longer in the Spanish-English assignment than in the English-Spanish one, whereas the control group took longer in the Spanish-English assignment, which is striking because even though it was a shorter source text, the assignment implied translating into a non-mother tongue of most of the participants.

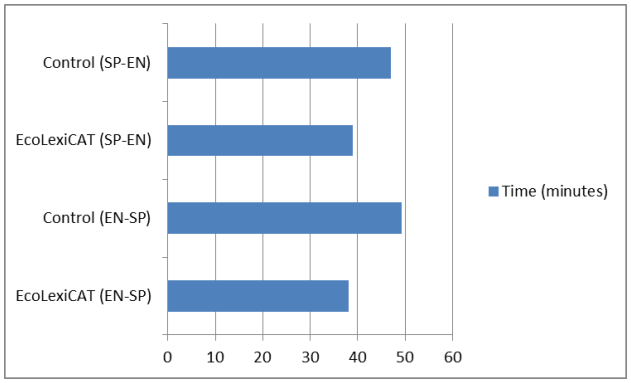


Figure 5: Time invested during SP-EN and EN-SP assignments by the EcoLexiCAT and control groups.

4.2.3 User satisfaction

Generally speaking, the subjects belonging to the EcoLexiCAT group believed that the tool is very useful (60%) or useful (40%). No subjects answered "not very useful" or "useless" when asked about the general usefulness of the tool for the translation of environmental texts.

The parameters of functionality, usability and efficiency were evaluated based on the rating of different items in a 1-to-5 scale, where 1 was the lowest and 5 the highest rate.

Regarding functionality (Figure 6), the subjects were asked whether the tool contained suitable features for: (1) the translation of environmental texts (80% answered 4 and 20% 5); (2) the comprehension phase of an environmental text (80% answered 4 and 20% 3); and (3) the production of an environmental text (50% answered 4, 40% 3 and 10% 2). This implies that EcoLexiCAT is more comprehension-oriented and that future improvements should head for the assistance in production-oriented tasks.

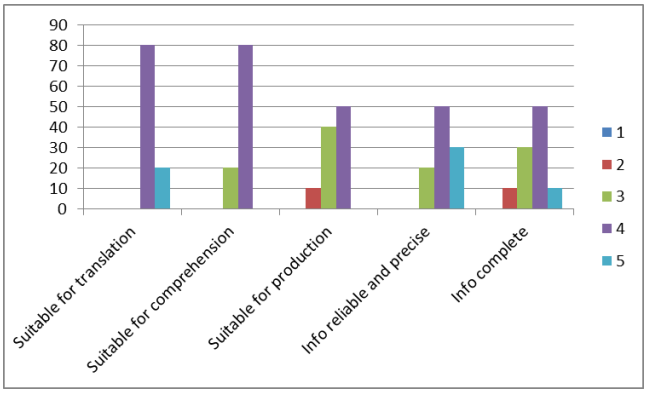


Figure 6: Functionality of EcoLexiCAT.

After that, they were consulted on the type of information provided and the usefulness of external resources. They were asked whether the information provided was: (1) reliable and precise (50% answered 4, 30% 5 and 20% 3); and (2) complete (50% answered 4, 30% 3, 10% 2 and 5).

When asked to rate the usefulness of external resources during their assignments (Figure 7), EcoLexicon, Linguee and Sketch Engine seem to be the best rated.

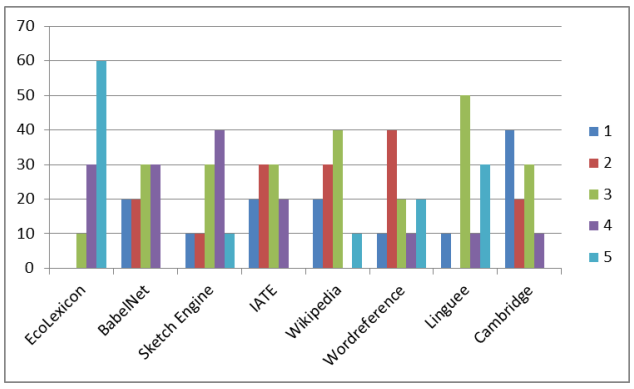


Figure 7: Usefulness of external resources.

As for usability (Figure 8), subjects were asked whether EcoLexiCAT: (1) was intuitive and easy to use (60% answered 4 and 40% 5); (2) had a functional design (50% answered 4, 40% 5 and 10% 3; and (3) provided an adequate interaction with the layout (40% answered 4; 30% 3, 20% 5 and 10% 2).

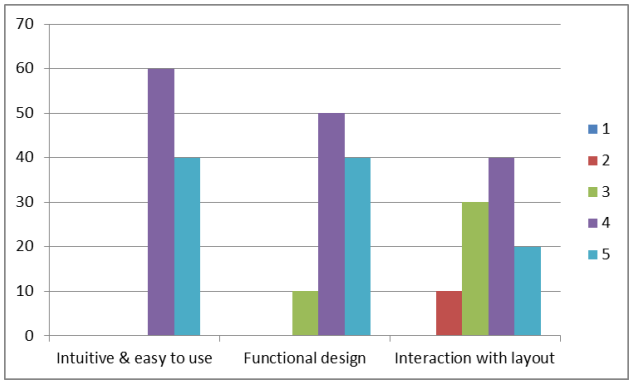


Figure 8: Usability of EcoLexiCAT.

Finally, efficiency (Figure 9) was assessed based on whether the information was loaded at the right speed (60% answered 4 and 40% 5) and fluency: (1) user interaction with the editor (50% answered 4, 20% 5, 20% 3 and 10% 2); (2) interaction of the editor with external resources (50% answered 4, 30% 3 and 20% 5); (3) user interaction with external resources (50% answered 3, 40% 4 and 10% 5).

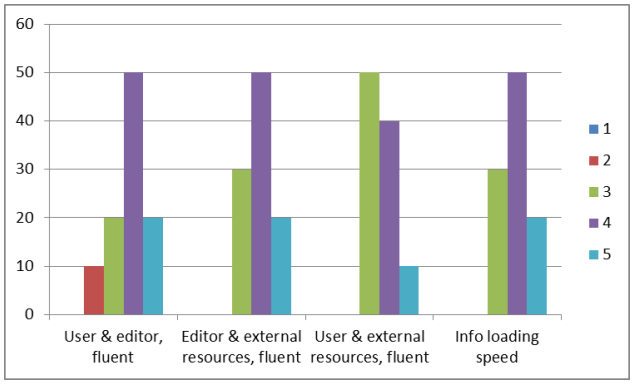


Figure 9: Efficiency of EcoLexiCAT.

The three parameters point to a favorable evaluation of EcoLexiCAT, although efficiency seems to be the one to be first improved in the future.

5. Conclusions

Although it is difficult to generalize de results obtained from the experiment described above, they do indicate that integrating terminology management in the translation workflow in a stand-alone interface improves the quality of the translation and reduces the time spent on the task. EcoLexiCAT users considered the tool to be useful and were satisfied with its functionality, usability and, although slightly less, efficiency.

In the near future, improvements will be made to the tool to make the interaction between the user and the tool and its external sources more fluent.

As the results also indicate that EcoLexiCAT was considered more useful for the comprehension phase than for the production phase, we plan to add other sources to improve on the latter. For example, Reverso Context will be added to the external sources. In addition, we are currently designing and implementing a phraseology module for EcoLexicon, which will be linked to EcoLexiCAT and will undoubtedly be very helpful for the production phase of the translation workflow.

Furthermore, EcoLexicon is currently being linked to other encyclopedic (i.e. DBpedia) and environmental resources (i.e. GEMET, AGROVOC) by means of Linked Data. Once the TKB is fully integrated into the Linguistic Linked Open Data, EcoLexiCAT will also benefit from reliably disambiguated encyclopedic and specialized term entries.

We also plan to carry out further evaluation experiments with professional translators and the machine translation option.

Finally, when all new features are included in the tool, EcoLexiCAT will be made freely available for any user interested in translating English or Spanish environmental texts. Users will only need to register and indicate their educational background, translation experience and the purpose for which they will be using the tool. This will help us analyze user profiles and behaviour when interacting with the tool. Moreover, it will allow us to classify the resources generated (i.e. TMs), which can be used as a parallel corpus, thus enriching both the tool and the EcoLexicon Corpus.

6. Acknowledgements

The authors would like to thank the master students who participated in the evaluation of EcoLexiCAT.
This research was carried out as part of projects FF2014-52740-P, Cognitive and Neurological Bases for Terminology-enhanced Translation (CONTENT), and FFI2017-89127-P, Translation-oriented Terminology Tools for Environmental Texts (TOTEM), funded by the Spanish Ministry of Economy and Competitiveness.

7. Bibliographical References

Durán Muñoz, I. (2010). Specialized lexicographical resources: a survey of translators' needs. In S. Granger & M. Paquot (eds) (2010). *eLexicography in the 21st century: New Challenges, new applications. Proceedings of ELEX2009*. Cahiers du Cental. Vol. 7. Louvain-La-Neuve: Presses Universitaires de Louvain, pp. 55–66.

Faber, P., León-Araúz, P. & Reimerink, A. (2011). Knowledge representation in EcoLexicon. In N. Talaván, E. Martín Monje & F. Palazón (eds.) *Technological Innovation in the Teaching and Processing of LSPs: Proceedings of TISLID*, 10. Madrid: Universidad Nacional de Educación a Distancia, pp 367–385.

Faber, P. (ed.) (2012). *A Cognitive Linguistics View of Terminology and Specialized Language*. Berlin/New York: Mouton de Gruyter.

Faber, P. (2015) Frames as a framework for terminology. In H. J. Kockaert & F. Steurs (eds.) *Handbook of*

Terminology, 1. John Benjamins Publishing Company, pp. 14–33.

Faber, P., León-Araúz, P. & Reimerink, A. (2014). Representing environmental knowledge in EcoLexicon. In *Languages for Specific Purposes in the Digital Era. Educational Linguistics*, 19. Springer, pp 267–301.

Faber, P., León-Araúz, P. & Reimerink, A. (2016) EcoLexicon: new features and challenges. In I. Kernerman, I. Kosem Trojina, S. Krek, & L. Trap-Jensen (eds.) *GLOBALEX 2016: Lexicographic Resources for Human Language Technology in conjunction with the 10th edition of the Language Resources and Evaluation Conference*, Portorož, pp. 73–80.

Fillmore, C. J. (1982). Frame Semantics. In The Linguistic Society of Korea (ed.) *Linguistics in the Morning Calm*. Seoul: Hanshin, pp. 111–137.

Fillmore, C. J. & Atkins, B. T. S. (1992). Toward a Frame-based Lexicon: The Semantics of RISK and Its Neighbors. In A. Lehrer & E. Kittay (eds.) *Frames, Fields and Contrasts: New Essays in Semantic and Lexical Organization*. Hillsdale NJ: Erlbaum, pp. 75–102.

Kilgarriff, A, Rychly, P., Smrz, P. & Tugwell, D. (2004). The Sketch Engine. In *Proceedings of the 11th EURALEX International Congress*. Lorient: EURALEX, pp. 105–116.

León-Araúz, P., Reimerink, A. & Faber, P. (2017) EcoLexiCAT: a Terminology-enhanced Translation Tool for Texts on the Environment. In *Electronic lexicography in the 21st century. Proceedings of eLex 2017 conference*, edited by Kosem, I., Kallas, J., Tiberius, C., Krek, S., Jakubíček, M. & Baisa, V., pages 321-341. Brno: Lexical Computing CZ s.r.o.

León-Araúz, P., San Martín, A. & Faber, P. (2016) Pattern-based Word Sketches for the Extraction of Semantic Relations. In *Proceedings of the 5th International Workshop on Computational Terminology (Computerm2016)*, pages 73-82. Osaka, Japan: COLING 2016

Navigli, R. & Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193, pp. 217–250.

Tudhope D., Koch T. & Heery R. (2006). Terminology Services and Technology: JISC state of the art review. Available at: http://www.ukoln.ac.uk/terminology/JISC-review2006.html

8. Language Resource References

LexiCon Research Group. (2017). EcoLexicon. Available at http/:/ecolexicon.ugr.es.

LexiCon Research Group. (2017). EcoLexicon English Corpus. Available at https://the.sketchengine.co.uk/open/

A Danish FrameNet Lexicon and an Annotated Corpus Used for Training and Evaluating a Semantic Frame Classifier

Bolette S. Pedersen[1], Sanni Nimb[2], Anders Søgaard[3], Mareike Hartmann[4], Sussi Olsen[5]
University of Copenhagen [1,3,4,5] & The Danish Society for Language and Literature[2]
Njalsgade 136, DK-2300 Copenhagen S[1,5,] Christians Brygge 1, DK-1219 Copenhagen K[2], Universitetsparken 1, DK-2200 Copenhagen,[3,4]
bspedersen@hum.ku.dk, sn@dsl.dk, soegaard@di.ku.dk, hartmann@di.ku.dk, saolsen@hum.ku.dk

Abstract

In this paper, we present an approach to efficiently compile a Danish FrameNet based on the Danish Thesaurus, focusing in particular on cognition and communication frames. The Danish FrameNet uses the frame and role inventory of the English FrameNet. We present the corresponding corpus annotations of frames and roles and show how our corpus can be used for training and evaluating a semantic frame classifier for cognition and communication frames. We also present results of cross-language transfer of a model trained on the English FrameNet. Our approach is significantly faster than building a lexicon from scratch, and we show that it is feasible to annotate Danish with frames developed for English, and finally, that frame annotations – even if limited in size at the current stage – are useful for automatic frame classification.

Keywords: Danish FrameNet, concept dictionary, frame-annotated corpus, low-resourced languages, semantic frame classifier

1. Danish as an under-resourced language

The META-NET white papers, which discussed the most urgent risks and chances of the European languages in the digital age, illustrated that several of our languages are severely under-resourced for the ongoing and coming digital revolution; Danish being no exception (cf. Pedersen et al. 2012).

To this end, several players in the Danish language and language technology community have in recent years focused on methods for building language technology resources and tools that employ *both* existing Danish lexical data *and* language transfer from better resourced



Figure 1: A wordnet (DanNet), a framenet and a semantically annotated corpus (SemDax) expanded from two dictionaries via common sense ids

languages (cf. Pedersen et al. 2009, Nimb et al. 2017, Johannsen et al. 2015, Levy et al. 2017). In order to enable this combination of methods, quite a lot of effort has been put into relating the resources to international standards (see for instance Martinez et al. 2016).

Most recently, effort has been put into compiling a Danish Berkeley style (Ruppenhofer et al. 2016) frame lexicon (BFN) by extracting semantic data from The Danish Thesaurus (DT) and The Danish Dictionary (DDO), (cf. Nimb et al. 2017). The FrameNet is one of several LT resources being built from a common sense id inventory first established with The Danish Dictionary and further employed in The Danish Thesaurus. Figure 1 illustrates the complex of interrelated resources, including also a Danish WordNet, DanNet, and a semantically annotated corpus, SemDaX.

In this paper we focus on the evaluation of our method of using linked data to compile new lexical resources to be used for semantic annotation and processing. At the current state, the lexicon contains 5,300 verbs (80 % of the verb lemmas in DDO) and 6,490 verbal nouns represented in 33,930 different expressions[1]. These are given either in the form of just the lemma itself, or in terms of a collocation from DDO, or an infinitive phrase with grammatical elements expressed as pronouns (based on the DDO valency patterns), or a multiword expression from DDO, all assigned a frame value from Berkeley FrameNet. The words and expressions represent 1/7 of the senses in DDO. Each verb lemma has an average of 3.3 frames. The noun lemmas have half the number of frames per lemma, namely 1.7. In total, 671 different frame values from Berkeley FrameNet have been applied, and among the most used ones are the ones describing acts from the semantic areas of motion, emotion, communication and cognition.

The paper is organized as follows. Below we sketch out how we, in order to test the strength of the lexical working method, started by compiling a pilot frame lexicon based on only two selected semantic domains in existing lexica (Section 2). Section 3 describes how we used the resulting set of frames to annotate selected

[1] The Danish frame lexicon is now freely available at https://github.com/dsldk/dansk-frame-net. It will be presented in more detail at The International FrameNet Workshop 2018, Multilingual FrameNets and Constructions at LREC 2018 (Nimb, submitted for review).

corpus examples within the same two domains, and finally we present in Section 4 how we used the frame and role annotated corpus data for training and evaluating a semantic frame classifier.

2. From a Thesaurus to a FrameNet Lexicon

The Danish Frame Lexicon is being built by exploiting the thematic divisions of DT and the fact that each subdivision includes groups of semantically closely related words (see Figure 2). In the pilot project we focus on groups of verbs, including idiomatic multiword units (typically phrasal verbs) and in the case of act groups, also deverbal nouns which in a semantic content correspond quite well to the ontological groupings of acts and events in FrameNet.

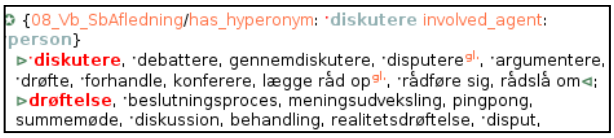


Figure 2: Synonyms and near synonyms of the verb *diskutere* ('to discuss') in DT, including verbal nouns and annotated with type 08 (acts) and semantic relations.

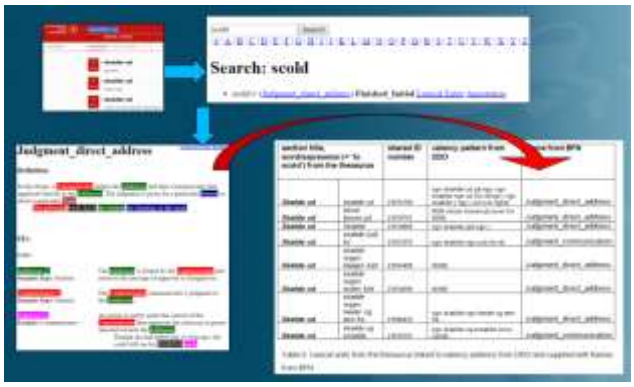


Figure 3: Mapping Berkeley frames onto thematically ordered verb groups from DT, in this case synonyms and near synonyms of *skælde ud* 'to scold'.

Having detected a group of closely related verbs and verbal nouns and furthermore supplied the verbs with valency patterns from DDO via the common id numbers, the editor would search for an appropriate frame in BFN and assign this to the particular group (shown in Figure 3). In this way, a relative large framenet lexicon is being compiled with relatively little effort[2].

Not surprisingly, however, not all groups were equally easy to assign frames to since the relation between DT and BFN was obviously not one-to-one in all semantic areas. A classic example is the discrepancy related to antonymous word senses such as remembering and forgetting which are covered by one frame (seen as the same scenario) in BFN but situated in two different groups in the thesaurus which has antonymy as an important criteria for section division. All such examples obviously required careful adjustment by the editors.

Some situations proved to have a higher degree of lexicalization in Danish than in English, for instance we found no frame which covered the Danish word *hemmeligholde* with the sense 'to refrain from telling'/'to keep as a secret'. The fact that the compilation of Berkeley FrameNet is still in progress and that all areas are not covered yet, probably also caused a lack of frames in some cases. Approx. 10 % of Berkeley FrameNet's ~1000 frames came into play when covering the two semantic areas in Danish, and the number of frames was more or less evenly distributed between the two.

3. Annotating Frames and Roles for Communication and Cognition

Behind the idea of a framenet lies not only the identification of a particular semantic frame for a particular verb or deverbal noun sense, but also the identification of the semantic roles/frame elements that are activated with a particular frame.

We rely on two assumptions:

- that our frame lexicon will ease annotation considerably since a very limited set of possible frames for a given word is presented to the annotator via the annotation tool, and
- that BFN frames for English can be more or less directly transferred to Danish; in other words, that the same frame elements or semantic roles can be identified in a Danish textual context with a particular frame. (A similar approach is taken for most other framenets being built for a number of languages, cf. Heppin & Gronostaj 2012, 2014 for Swedish, Candito et al. 2014 for French, Ohara 2014 for Japanese).

In order to test this approach, we annotated 440 sentences from the corpus with their corresponding frames and frame elements. The sentences from SemDax cover a variety of text types such as blog, chat, forum, magazine, Parliament debates (written down by professionals), and newswire, of which the latter constitutes almost half of the corpus.

In order to easily access examples which would evoke frames relating to communication and cognition, we took advantage of the coarse sense annotations available in SemDax corpus (Pedersen et al 2016) and extracted all sentences annotated with either cognition and communication events (or both)[3]. This extraction also enabled us to prove whether a frame lexicon based on thesaurus vocabulary was actually extensive enough.

We used an open source, browser-based framenet annotation tool (https://github.com/andersjo/framenet-annotation). For each verb or verbal noun relating to

[2] The project is supported by the Danish Research Council and by The Carlsberg Foundation.

[3] In SemDax all verbs, nouns and adjectives are annotated with so-called supersenses, cf. Martinez et al. (2016).

cognition or communication, the annotator has access to a small set of relevant frames, depending on the previous semantic annotation of the word. Once a frame is chosen, the annotator can assign the frame elements pertaining to this frame to the other words in the sentence by writing the word's position in the role boxes, see Figure 4.

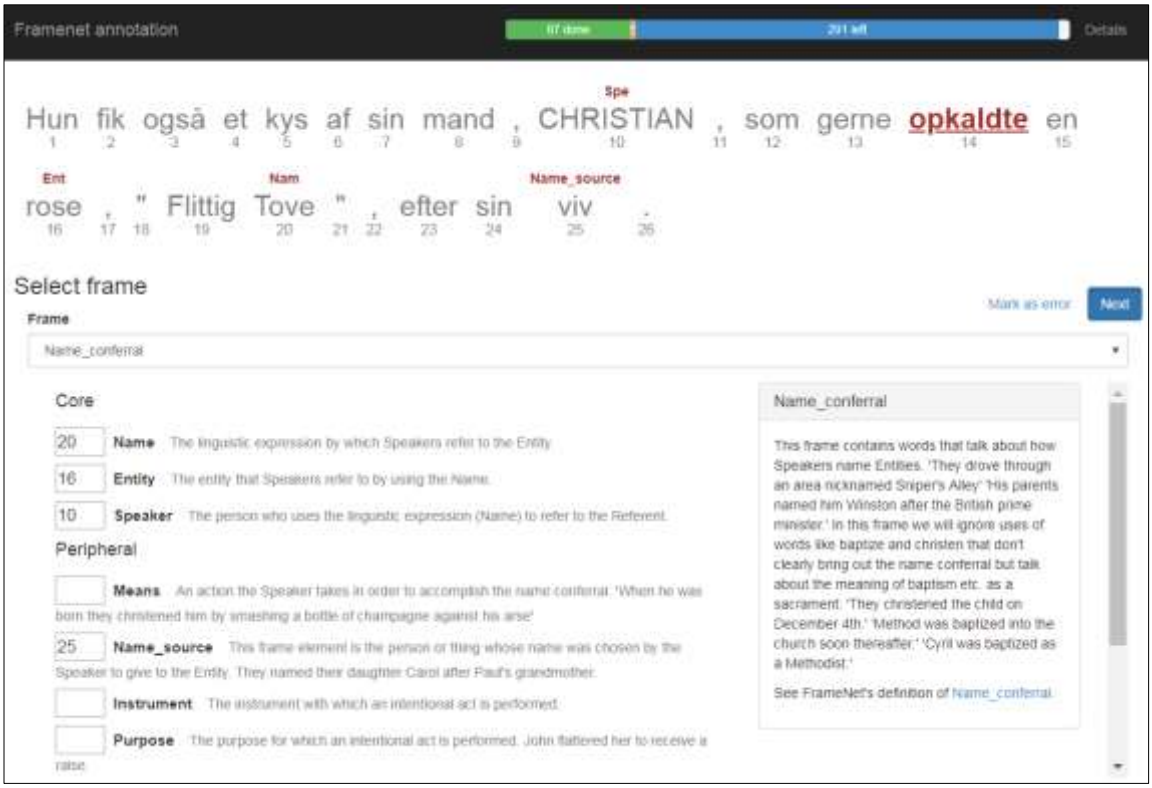


Figure 4: The FrameNet annotation tool shows the sentence with the verb to be annotated in red, and the assigned frame elements in red above the words. Below is the box for choosing the frame and boxes for assigning words to the frame elements.

Several observations were made during annotation:

- Semantic roles were generally straight-forward to detect and corresponded well to the ones suggested by the frame, however, in hardly any of the examples would we find all roles present; leaving for many of the roles to be implicit in the text.
- The annotators would sometimes disagree upon which precise frame to choose, often depending on whether to interpret the act in a concrete way or not (would the purpose of the act be more important to annotate than the concrete way of carrying it out, for example).
- The lexicon facilitated to a high degree the annotation task since the annotator only had to consider a very restricted set of frames in each case.

However, for some verbs, frames were missing because the specific sense had not been foreseen in the frame lexicon based on the DT vocabulary. The largest part of these were ad hoc (figurative) senses not to be included in the frame lexicon (nor in the dictionaries), but there were also cases which led us to expand the lexicon, e.g. cognition verbs with communication senses in corpus but not included in the thesaurus chapters on communication (for good reasons since the sense depends completely on a very specific discourse context).

The 440 frame annotated sentences were validated and in all cases supplied with the most appropriate frame before we used the corpus for training and evaluation.

4. Training and Evaluation of a Multilingual Semantic Frame Classifier

In order to demonstrate the value of our resource in terms of training and evaluating NLP models, we used the 440 frame-annotated sentences of the SemDax corpus to train and evaluate a semantic frame classifier. Since the corpus is relatively small and only contains a subset of the frames contained in the Danish FrameNet Lexicon, we experiment with cross-lingual frame label prediction in order to exploit more data. We reduce the frame-semantic parsing problem to a sentence classification problem, i.e. we train a model that predicts one frame label per sentence. The top frames subsume a total of 6 distinct frames, making this a 6-way multinomial classification problem.

Inspired by recent semantic parsing models, e.g., Zhou and Xu (2015), we use a set of binary deep, bi-directional Long Short Term Memory (LSTM) networks to predict frame labels. Each network predicts a single label, and we evaluate each network individually by computing sentence-level F1-scores.

In the following, we report implementation details and results for our experiments in the frame prediction task. We ran experiments in three setups.

- First, we trained the model on the Danish data using 10-fold cross validation and predicted Danish test data.
- Second, we did a cross-lingual experiment, where we trained on English FrameNet example sentences annotated with the equivalents of the frame labels in the Danish corpus, and predicted the Danish test data. This setting is unsupervised in that we did not use any Danish training data.
- The third setting was meant to provide a reference point for comparing the performance of the Danish model. Here, we trained on English frames and predicted English test data. As a baseline, we used a model that randomly assigns labels to the test data.

In all experiments, we represented sentences by pre-trained cross-lingual word embeddings. The embeddings are 40-dimensional, computed on 59 languages using MultiCCA.[4]

Our results for the six most frequent frames[5] are presented in Table 1. For all experiments, the LSTM hyper-parameters are tuned on English development data in a supervised set-up. The best hyper-parameter values that were used in the final experiments is a single hidden layer with a dimensionality of 20, the optimization algorithm was Adam with initial learning rate 0.001, and we used tanh activation functions and softmax for the output layer. The maximum sequence length during training is set to 20.

English-English

	Ours	*Random*
Statement	0.81	0.37
Opinion	0.39	0.06
Telling	0.49	0.05
Text_creation	0.60	0.02
Becoming_aware	0.57	0.05
Certainty	0.47	0.07

Danish-Danish

	Ours	*Random*
Statement	0.66	0.25
Opinion	0.69	0.15
Telling	0.52	0.11
Text_creation	0.86	0.09
Becoming_aware	0.43	0.07
Certainty	0.54	0.09

English-Danish

	Ours	*Random*
Statement	0.31	0.20
Opinion	0.16	0.13
Telling	0.13	0.12
Text_creation	0.08	0.05
Becoming_aware	0.06	0.05
Certainty	0.08	0.05

Table 1: Supervised and unsupervised F1-scores for the 6 most frequent frames.

As mentioned, we rely only on the Danish Framenet annotated corpus, not on the frame lexicon as such. Das et al. (2010), for example, use heuristics to detect frame triggers and pick the appropriate frame using a classifier that only considers the frames licensed by the (English) Framenet. Li et al. (2012), working on POS tagging, use a tag dictionary to constrain the output space; the Danish Framenet could be used in a similar way for frame-semantic parsing. The cross-lingual parsing performance would probably improve a bit from using such constraints. On the other hand, learning to associate trigger words and frames is the easiest part of the frame detection problem; disambiguation is the hardest. In other words, if we assume that we can solve the disambiguation problem with more data, we should be able to trivially learn which frames are adequate for each verb. Also, we are already limiting the search space by considering only a subset of the total set of frames.

As can be seen from Table 1, our F1-scores are much lower when relying exclusively on cross-lingual signals (English-Danish). This shows that only part of the signal from the English data transfers. The fact that we perform better than the random baseline across all frames, shows that transfer is possible, as also suggested by Johannsen et al. (2015). However, the gap between Danish-Danish and English-Danish shows the value of annotating data in low-resource languages – and, in particular, the need for scaling up the annotated resource. Since performance correlates with support in the data – i.e., our model performs better on frequent frames – it may be beneficial to consider active learning as a strategy to efficiently annotate more data (Martínez et al., 2015).

5. Concluding Remarks

Building language resources for LT is cumbersome and expensive, and relying on existing lexicographical resources is often a challenging and not always straight forward business, in particular if – at the same time – you want to conform to international standards.

Our experiments with the compilation of a Danish framenet show that – with the given high-quality background resources, DDO and DT – it is actually feasible to build a framenet on top of an existing resource and to start from the lexicon part and move onwards to the corpus annotation. Further, adapting the role inventory from BFN to Danish proves unproblematic, with a few exceptions.
In our training experiments we have shown that language transfer of semantic frame information is possible, but that improvements are considerable when

[4] http://128.2.220.95/multilingual/data/

[5] Due to data scarcity, only the six most frequent frames were used for training and evaluation in this experiment.

combining language transferred data with annotated data of the specific target language, in this case Danish.

6. Bibliographical References

Candito, M., Amsili, P., Barque L., Benamara, F., De Chalendar, G., Djemaa, M., Haas, P., Huyghe, R., Yannick Mathieu, Y., Muller, P., Sagot, B. & Vieu, L. (2014). Developing a French FrameNet: Methodology and First Results. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14),* Reykjavik, Iceland.

Das, D., N. Scneider, D. Chen & N. Schmith. (2010). Probabilistic Frame-Semantic Parsing. In *Proceedings opf ACL 2010*.

Heppin, K. & M. Gronostaj. (2012). The Rocky Road towards a Swedish FrameNet - Creating SweFN. In: *Proceedings of the Eight International Conference on Language Resources and* Evaluation. Istanbul, Turkey

Johannsen, A., Alonso, H. M., Søgaard, A. (2015). Any-langøøuage frame-semantic parsing. Empirical Methods in *Natural Language Processing* 2015. Lisbon, Portugal.

Li, J. V. Graca, B. Taskar. (2012). Wiki-ly Supervised Part-of-Speech Tagging. In: *Proceedings of ACL 2012.*

Martínez Alonso, H., Johannsen A, Olsen S, Nimb S and Pedersen BS (2016). An empirically grounded expansion of the supersense inventory. *Proceedings of the 8th Global Wordnet Conference*, Bucharest, Romania.

Martínez Alonso, H. M., Plank, B., Johannsen, A., Søgaard, A. (2015). Active learning for sense annotations. *The 20th Nordic Conference on Computational Linguistics*. Vilnius, Lithuania.

Nimb, S. , A. Braasch, S. Olsen, B. S. Pedersen, A. Søgaard (2017). From thesaurus to framenet. In: *Proceedings of eLex 2017*, Leiden.

Nimb, S., Lorentzen, H., Theilgaard, L., Troelsgård, T. (2014 a). *Den Danske Begrebsordbog*, Det Danske Sprog- og Litteraturselskab, Copenhagen, Denmark.

Nimb, S., Lorentzen H., Trap-Jensen, L.(2014 b). The Danish Thesaurus: Problems and Perspectives In: Abel A., Vettori C. & Ralli N. (eds.). Proceedings of the XVI EURALEX International Congress: The User in Focus. 15-19 July 2014. Bolzano/Bozen 2014: EURAC Research, pp. 191-199.

Nimb, S. , A. Braasch, S. Olsen, B. S. Pedersen, A. Søgaard (2017). From thesaurus to framenet. In: *Proceedings of eLex 2017*, Leiden.

Nimb, S. (Submitted for review). The Danish Frame Lexicon : Method and Lexical Coverage. In *Proceedings from The International FrameNet Workshop 2018 at LREC/Japan: Multilingual FrameNets and Constructicons*.

Ohara, K. H. (2014). Relating Frames and Constructions in Japanese FrameNet. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14),* Reykjavik, Iceland pp. 2474-2477.

Levy, Omer; Søgaard, Anders; Goldberg, Yoav. 2017. A strong baseline for learning cross-lingual word embeddings from sentence alignments. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL).* Valencia, Spain.

Pedersen, B. A. Braasch, A. Johanssen, H. Martínes Alonso, S.Nimb, S. Olsen, A. Søgaard, N. Sørensen (2016). The SemDaX Corpus - sense annotations with scalable sense inventories. In *Proceedings of the 10th edition of the Language Resources and Evaluation* Conference, Portorož, Slovenia.

Pedersen, B.S, J. Wedekind, S. Kirchmeier-Andersen, S. Nimb, J.E. Rasmussen, L.B. Larsen, S. Bøhm-Andersen, H.Erdman Thomsen, P. J. Henrichsen,J. O. Kjærum, P. Revsbech, S.Hoffensetz-Andresen, B. Maegaard (2012). *The Danish Language in the Digital Age - Det danske sprog i den digitale tidsalder* META-NET White Paper Series, Springer Verlag.

Pedersen, B.S, S. Nimb, J. Asmussen, N. Sørensen, L. Trap-Jensen, H. Lorentzen (2009). DanNet – the challenge of compiling a WordNet for Danish by reusing a monolingual dictionary. *Language Resources and Evaluation, Computational Linguistics Series, pp.269-299.*

Ruppenhofer, J., Ellsworth, M., Petruck, M. R. L., Johnson, C. R., Baker, C. F., Scheffczyk, J. (2016). *FrameNet II: Extended Theory and Practice* (Revised November 1, 2016.) https://framenet.icsi.berkeley.edu/fndrupal/the_book.

Zhou, J., Xu, W. (2015). End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics. Beijing, China.*

7. Language Resource References

The Danish Dictionary (DDO) (2008-) : online dictionary, ordnet.dk/ddo, Det Danske Sprog- og Litteraturselskab, Copenhagen, Denmark.

The Danish Frame Lexicon (2018): available at https://github.com/dsldk/dansk-frame-net.

The Danish Thesaurus (DT): Nimb, S., Lorentzen, H., Troelsgård T., Theilgaard, L., Trap-Jensen, L. (2014a): *Den Danske Begrebsordbog*, Society for Danish Language and Literature, Copenhagen, Denmark.

SLIDE – a Sentiment Lexicon of Common Idioms

Charles Jochim, Francesca Bonin, Roy Bar-Haim, Noam Slonim
IBM Research
{charlesj,fbonin}@ie.ibm.com, {roybar,noams}@il.ibm.com

Abstract

Idiomatic expressions are problematic for most sentiment analysis approaches, which rely on words as the basic linguistic unit. Compositional solutions for phrase sentiment are not able to handle idioms correctly because their sentiment is not derived from the sentiment of the individual words. Previous work has explored the importance of idioms for sentiment analysis, but has not addressed the breadth of idiomatic expressions in English. In this paper we present an approach for collecting sentiment annotation of idiomatic multiword expressions using crowdsourcing. We collect 10 annotations for each idiom and the aggregated label is shown to have good agreement with expert annotations. We describe the resulting publicly available lexicon and how it captures sentiment strength and ambiguity. The *Sentiment Lexicon of IDiomatic Expressions (SLIDE)* is much larger than previous idiom lexicons. The lexicon includes 5,000 frequently occurring idioms, as estimated from a large English corpus. The idioms were selected from Wiktionary, and over 40% of them were labeled as sentiment-bearing.

Keywords: Idiom, Lexicon, Sentiment Analysis

1. Introduction

Multiword expressions (MWE) are a key challenge in Natural Language Processing (Sag et al., 2002). Among MWEs, *idioms* are often defined as non-compositional multiword expressions, the meaning of which cannot be deduced from the literal meaning of constituent words (Nunberg et al., 1994).

Sentiment analysis systems typically consider words as the basic sentiment units. Word sentiments are either learned from the training data or looked up in a sentiment lexicon. Text sentiment is then derived by means of aggregation over word sentiments, often with some treatment of compositional phenomena such as valence shifters (Polanyi and Zaenen, 2004) and mixed sentiment (Kiritchenko and Mohammad, 2016). Other approaches are based on bottom-up sentiment composition, starting at the word level and computing the sentiment of each phrase based on the semantics and sentiment of its daughter phrases, according to the syntactic structure of the sentence (Moilanen and Pulman, 2007; Nakagawa et al., 2010; Socher et al., 2013).

Due to their non-compositionality, idioms are often not handled correctly by current sentiment analysis systems (Balahur et al., 2010). Word-level sentiment analysis would miss the positive sentiment in *two thumbs up*, and on the other hand, we might incorrectly assign positive sentiment to *as well as*, because of the positive sentiment of *well*. Similarly, we would like to know that it is not good if something *bites the dust*, while we would be happy to hear that our handing of idioms was *dead on*. Ignoring idioms overlooks an important signal of the sentiment of the text, as figurative and idiomatic language often directs sentence polarity (Rentoumi et al., 2012). For the above reasons, idioms have begun to receive some attention in recent sentiment analysis literature (Williams et al., 2015; Liu et al., 2017). Yet, robust treatment of idioms is hindered by their limited coverage in current sentiment datasets and lexicons.

In this work we introduce *SLIDE (Sentiment Lexicon of IDiomatic Expressions)*, a new resource for sentiment analysis, created via crowdsourcing. Our lexicon is an order of magnitude larger than previous idiom sentiment lexicons and focuses specifically on the most frequently used idioms.[1] In creating this resource, we are somewhat agnostic to the exact definition of *idiom*. We are more generally interested in sentiment analysis that can handle MWEs. In this paper, we have initially focused on idioms because they are the most problematic for sentiment analysis, being strictly non-compositional.

In the rest of the paper, we first describe the crowdsourcing-based idiom annotation process and its quality assessment (Section 2). We then provide a description of the resulting lexicon (Section 3). Section 4 describes an auxiliary annotation step aimed at identifying frequent idioms that have non-idiomatic meaning and neutral sentiment in most contexts. Section 5 covers the previous work related to idioms, sentiment lexicons, and sentiment analysis. Finally, we conclude and discuss future work in Section 6.

2. Lexicon Creation

2.1. Idiom Selection

We start with the list of all idioms available in Wiktionary,[2] which resulted in 8,772 idioms. To narrow down the list of idioms we send for annotation, we take the following steps: 1) We remove some special Wiktionary links that start with `Appendix:` or `Citation:` (only 22 in total); 2) We remove all unigrams, which we do not consider to be idioms here; however, hyphenated words are not removed; 3) Finally, we remove all idioms comprised only of stopwords (e.g., *about to, and so on*). This results in a list of 8,637 idiomatic expressions.

The resulting list contains many idioms that are infrequently used or are less popular in current usage. To ease the annotation effort, we have chosen to select frequently used idioms, which would be the most useful for sentiment analysis tools. To determine the most frequent idioms we

[1] SLIDE is available at `http://www.research.ibm.com/haifa/dept/vst/debating_data.shtml`

[2] `https://en.wiktionary.org/w/index.php?title=Category:English_idioms`

use a proprietary corpus of news articles and other publications coming from thousands of sources. The corpus includes about 200 million documents. We count the idiom's frequency in this corpus and then select the most frequent 5,000 for annotation.

Many idioms have personal pronouns or possessive adjectives and they are listed as idioms using *one*, *anyone*, or *someone*, and *one's*, *anyone's*, or *someone's*. For example we have the idioms *keep someone posted* and *on one's feet* in our lexicon. To count the frequency of these idioms we expand the generic pronoun with the various pronouns for person, number, and gender that we may encounter in the corpus. Specifically, in the case of possessive adjectives, we replace occurrences of *one's*, *anyone's*, or *someone's* with *my, your, his, her, our*, and *their*. For the remaining pronouns (*one*, *anyone*, and *someone*), if they follow a verb they are treated as object pronouns and replaced with *me, you, him, her, us*, and *them*, otherwise they are treated as subject pronouns and replaced with *I, you, he, she, we*, and *they*. The frequencies for each of the pronoun-substituted idioms are summed for the frequency of the original idiom. After the frequency is collected for all idioms, we select the top 5,000 most frequent idioms to be manually annotated for sentiment. This threshold covers the idioms that occur in at least 1,266 documents in our corpus.

2.2. Idiom Annotation

We use the CrowdFlower platform for crowdsourcing annotation.[3] CrowdFlower is an online labor market where workers are paid to complete tasks uploaded to the platform. The system requires the submission of a task via a web interface.

We first ran several exploratory pilots for the annotation task to refine the guidelines and settings for the annotation. Following CrowdFlower conventions, the annotation guidelines are succinct. Annotators are provided with the following overview of the task:

> *In this job, you will be shown expressions and their Wiktionary definitions and you should indicate if the expression has a positive, negative, or neutral sentiment in most contexts in which it is found. You should consider the most common meaning of the expression, which may or may not be idiomatic. Please look at the definitions provided to help guide your decision.*

And we suggest the following steps for performing the annotation:

1. Read the expression and definitions and imagine different contexts in which it can be found.
2. Decide if the expression tends to contribute positively, negatively, or neutrally to the sentiment of these contexts.
3. Select positive, negative, or neutral accordingly (if the content is inappropriate or vulgar, select only the Inappropriate/Vulgar box).

[3] https://www.crowdflower.com/

Some expressions can have several meanings, which may be more or less idiomatic. For example *rip up* is defined as (i) "to destroy by ripping" (less idiomatic), and also (ii) "to move quickly or violently" (more idiomatic). Our instructions ask the annotator to label the most commonly found sentiment of the expression, whether it be idiomatic or not. Annotating the most frequent sentiment of the expression is due to practical considerations, as we assume that the users of our lexicon will not perform sense disambiguation, which is a very challenging task for idioms. A number of idioms still have similar sentiment across their different meanings; so even in cases where the expression's meaning may be ambiguous, the sentiment is not. For example, the more literal and more idiomatic senses of *fall apart* are used to express negative sentiment.

Crowdsourced annotation has been shown to be an effective and reliable source of labeled data when a sufficiently large number of non-expert annotators are employed (Snow et al., 2008; Nowak and Rüger, 2010).[4] We collected 10 annotations for each idiom for greater confidence in our annotation. The CrowdFlower platform includes mechanisms to ensure consistent annotation by including random test questions with known labels to be sure the annotator understands the task and is annotating in good faith. If an annotator falls below a predefined threshold of accuracy on test questions, then the annotator is removed from the task and their annotations are discarded. This further preserves the quality of the annotation. CrowdFlower has three levels of annotation expertise and our annotation task used Level 2. We selected annotators from the United States, Canada, United Kingdom, Ireland, and Australia in an effort to have as native fluency as possible.

We found that even native speakers may be less familiar with some idioms. In pilot annotation we included links to Wiktionary definitions for when annotators might be in doubt, but we did not present the definition to the annotators. CrowdFlower annotators were likely reticent to look up the definition and even as native speakers could easily have overlooked some expression's meaning when many idioms have multiple meanings or senses.[5] We resolved this by embedding the Wiktionary page directly in the CrowdFlower annotation (see Figure 1). This encourages the annotators to check the definition and makes the resulting annotation more consistent.

To confirm the quality of our annotation we measure Fleiss' kappa (Fleiss, 1971) between an expert annotator and the most voted label from the crowdsourced annotation (Table 1). We took a random sample of 400 idioms from the 5,000 annotated and had them annotated by an in-house expert annotator. The kappa score for this sample is 0.55. We expect some disagreement in the annotation because there is inherent subjectivity in sentiment assessment, in addition to the ambiguity of some of the idioms. When presented out of context, the assessment of an idiom's sentiment sometimes depends on the annotator's own biases and beliefs.

[4] Snow et al. (2008) suggest at least four annotators to reach expert level annotation; Nowak and Rüger (2010) found a majority vote of nine non-expert annotations matched expert annotation.

[5] 36% of the idioms we collect from Wiktionary have more than one meaning.

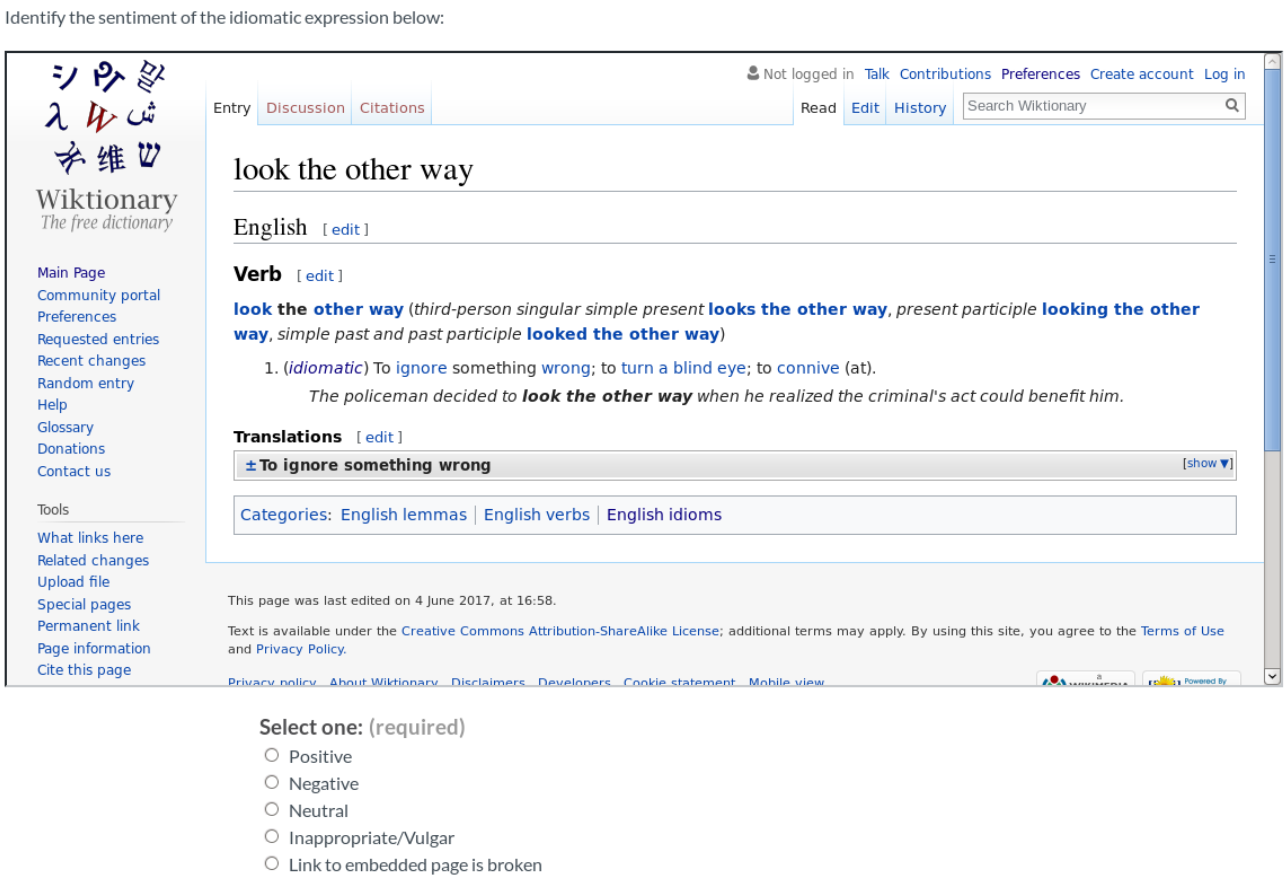


Figure 1: Annotation example of Wiktionary idioms in CrowdFlower.

	All	≥ 60%	≥ 80%
Kappa	0.55	0.61	0.74
Sample size	400	356	244

Table 1: Fleiss' kappa agreement between expert and crowd. Latter columns refer to percent agreement in crowd.

For example, the idiom *live on the edge* was assessed as positive by two annotators, as negative by three annotators, and as neutral by five annotators.

Users of our lexicon may also choose to filter out lower-argreement idioms. As expected, higher agreement among crowd annotators leads to higher agreement with the expert annotator. If we only consider idioms where at least 60% of the crowd agreed (90% of the lexicon), then on this sample of 356 we have Fleiss' kappa of 0.61. Likewise with 80% crowd agreement (60% of the lexicon) we have kappa of 0.74 over a sample of 244 idioms.

3. Resource Description

The result of our annotation task is a lexicon of 5,000 idioms with at least 10 annotations. The lexicon we release includes the percentage of *positive, negative, neutral*, and *inappropriate* annotations so that future users can decide the degree of polarity that they would like to include, e.g., only the most positive and negative idioms or also idioms with weaker sentiment.

The lexicon includes a sentiment label along with the distribution of sentiment annotations. Our labels are assigned by taking the label with the greatest number of votes from the crowdsourced annotation. In the case of ties between *positive* (or *negative*) and *neutral*, the label is *positive* (resp. *negative*). In the rare cases of ties between *positive* and *negative*, we use the *neutral* label. The resulting lexicon has 946 *positive* idioms, 1,108 *negative*, 2,945 *neutral*, and 1 *inappropriate* (see Table 2). Table 2 includes additional columns that capture the lexicon's makeup. By looking at all idioms with more than 20% *positive* or *negative* annotation, we get a sense of the number of idioms with weak sentiment, i.e., over 2/3 have at least weak polarity. One may also consider using a smaller lexicon with stronger sentiment. Using 60% or 80% agreement from CrowdFlower we reduce the number of idioms with weak or ambiguous sentiment (see the kappa values in Table 1), which results in smaller *positive* and *negative* lexicons.

	Label	≥ 20%	≥ 60%	≥ 80%
Positive	946	1717	745	426
Negative	1108	1819	917	517
Neutral	2945	4252	2842	2021
Inapprop.	1	29	1	0

Table 2: Lexicon statistics. Percentage columns refer to distribution of total annotation, e.g., 1717 idioms are *at least* 20% positive.

As stated in the guidelines, we want to capture the senti-

ment of the expression's common usage. However, we are also interested in the ambiguity that is reflected in the answer distribution. Williams et al. (2015) include an option for directly labeling an idiom as ambiguous (which is ultimately combined with *neutral*). Our approach allows us to handle ambiguous idioms by considering their most frequent sentiment, while the approach of Williams et al. only assigns sentiment to unambiguous idioms. We opted for only options of *positive, negative*, and *neutral*.[6] Our hope was that the distribution of annotator answers would capture the ambiguity of the idioms and the resulting label distribution could represent that ambiguity. For example, *make the cut* is labeled 80% positive and 20% neutral. It is defined in Wiktionary as "to succeed at something or meet a requirement; to be chosen out of a field of candidates or possibilities," which is largely positive, however the ambiguity might come from usage where *making the cut* is not excelling but sufficient. This contrasts *at the ready*[7] which is largely neutral with very weak positive sentiment (80% neutral vs 20% positive). These are only two examples and there are others with ambiguity that is more debatable. However, we think this highlights the advantage of having *positive, negative*, and *neutral* annotation so that we may have more insights into the ambiguity, e.g., mixed sentiment with equal percentages for *positive* and *negative*, or weak sentiment with a low percentage of *positive* (resp. *negative*) but for the most part *neutral*. In this respect our lexicon is similar to SentiWordNet (Esuli and Sebastiani, 2006; Baccianella et al., 2010), which has scores for positivity, negativity, and objectivity.

In Table 3 we have listed some examples from our lexicon that are potentially problematic when handling words alone. The list is broken down into frequency bins, and for each bin, the idioms' frequency is above the bin's threshold, but lower than the previous bin's threshold. We can see that, while the sentiment of the 500 most frequent idioms is obviously important for sentiment analysis, there is also important idiom sentiment in the long tail of less frequent idioms. This motivated our choice in drawing idioms from a much larger pool of idioms than previous work.

Positive	Negative	Neutral
document frequency $>$ 160,900 (10% of lexicon)		
make a difference	behind bars	as well as
bounce back	under fire	on the table
on one's feet	in the red	keep an eye on
document frequency $>$ 11,430 (50% of lexicon)		
in shape	red flag	on the clock
over the moon	in the hole	outside the box
breath of fresh air	wide of the mark	change of heart
document frequency $>$ 1,266 (entire lexicon)		
on cloud nine	up a tree	dead ringer
have a ball	on thin ice	scratch the surface
bury the hatchet	booby prize	birthday suit

Table 3: Frequent idioms by label. For each bin, the idioms' frequency is above the bin's threshold, but lower than the previous bin's threshold.

4. Context-Based Post-Filtering

As previously mentioned, some phrases in our lexicon have both idiomatic and non-idiomatic interpretations, which may differ in polarity. The guidelines ask the annotators to choose the sentiment most commonly associated with the phrase, considering both idiomatic and non-idiomatic meanings. However, showing the annotators the idiomatic definitions from Wiktionary created bias towards these idiomatic senses. For example, the phrase *make it*, which has the idiomatic meaning of achieving one's goals, was given positive sentiment in our lexicon, although in most contexts it has neutral polarity. This may introduce many errors when applying the lexicon to sentiment analysis tasks.

We observed that such situations are much more common for highly frequent phrases, for which polarity errors also have the greatest impact. We therefore apply an additional filtering step to the 300 most frequent idioms in our lexicon with *positive* or *negative* labels. For each of these expressions, we query the corpus mentioned in Section 2.1 and retrieve ten different contexts containing the expression. We give the expressions and contexts to a group of in-house annotators to determine if the expression is *positive*, *negative*, or *neutral* in a sample of actual contexts. For each of the 300 expressions there are ten contexts and we require five annotations for each context. To determine if the idiom should be filtered, we first aggregate the annotations by taking the label with the majority vote per context (if there is no majority, the label is *neutral*). Then we check the number of context labels per expression. If the majority of contexts have been labeled *positive* or *negative*, we mark the expression to keep, otherwise we mark it to filter out. This annotation marks 103 n-grams that can be filtered out, leaving 197 as idioms with sentiment in the majority of contexts in which they were found. Some examples of filtered phrases are *do in*, *play games* and *make it*. We provide this filtering as an additional layer of annotation, rather than discarding filtered phrases from the lexicon.

5. Related Work

There is a wealth of literature on sentiment analysis ((Liu, 2012)) and idioms ((Nunberg et al., 1994)). Here we focus on work related to building sentiment lexicons and the importance of handling idioms in sentiment classification.

Available sentiment lexicons do not handle idiomatic expressions and focus almost entirely on unigrams. Manually curated lexicons such as the Harvard General Inquirer (Stone et al., 1966) or MPQA (Wilson et al., 2005) have hyphenated words but no idioms or MWEs. The lexicons created by early automatic approaches (Turney and Littman, 2003; Hu and Liu, 2004) deal with words but not longer n-grams. Approaches using WordNet (Miller, 1995), like those of Esuli and Sebastiani (2006) or Blair-Goldensohn et al. (2008), will include MWEs but WordNet has low coverage of idioms in our lexicon. Other graph-based approaches using distributionally similar n-grams (Velikovich et al., 2010) can return sentiment for MWEs, but the approach is sensitive to parameter tuning and there has been

[6] The *inappropriate* option was for filtering out idioms with swear words and was rarely used or needed.

[7] Wiktionary: "ready; in a state of preparation or waiting; in position or anticipation."

no evaluation of the quality of the MWE sentiment. Recently, Williams et al. (2015) released a sentiment lexicon with 580 idioms, but the selection of idioms focused on emotional idioms, some of which are not very frequent (e.g., they showed that more than a quarter were not found in the British National Corpus). To address the lack of large-scale idiom sentiment lexicon, we manually annotated 5,000 of the most frequently used idioms, which is still feasible using crowdsourcing and avoids potential pitfalls of automatic lexicon creation.

After analysis from Balahur et al. (2010) showing the prevalence of idiom errors in sentiment classification, and some success by Xie and Wang (2014) using idioms for sentiment classification in Chinese, Williams et al. (2015) further investigated the role of idiomatic expressions in sentiment analysis. They showed that the use of sentiment annotated idiomatic expressions as features can improve the results of sentiment analysis of sentences.

Liu et al. (2017) has also recently shown that consideration of idioms can improve sentiment classification. They propose two models to address the more and less compositional idioms discussed in (Nunberg et al., 1994): one treating idioms as a fixed phrase and a second that considers morphology to account for the possible syntactic variation in the idioms. Incorporating external knowledge on idiom sentiment is likely to further improve the performance of such approaches, in particular on sentences containing idioms that did not appear in the training data.

6. Conclusion

In this paper we have motivated the need for better handling of idioms in sentiment analysis and presented a large sentiment lexicon of idiomatic expressions for this purpose. We make the final lexicon available and hope that it will be useful in improving sentiment classification and opinion mining. In future work, we plan on expanding the lexicon by utilizing other sources of idioms, as well as covering additional types of sentiment-bearing MWEs other than idioms. For example, a negative health condition such as *high blood pressure* should get negative sentiment. We then plan on using this lexicon for experiments not only in sentiment classification but other related text classification tasks that would benefit from distinguishing idioms and their sentiment, e.g., stance classification.

7. Bibliographical References

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., Goot, E. v. d., Halkia, M., Pouliquen, B., and Belyaeva, J. (2010). Sentiment analysis in the news. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Languages Resources Association (ELRA).

Blair-Goldensohn, S., Neylon, T., Hannan, K., Reis, G. A., Mcdonald, R., and Reynar, J. (2008). Building a sentiment summarizer for local service reviews. In *NLP in the Information Explosion Era*.

Esuli, A. and Sebastiani, F. (2006). SENTIWORDNET: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06)*, pages 417–422.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Kiritchenko, S. and Mohammad, S. M. (2016). Sentiment composition of words with opposing polarities. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1102–1108, San Diego, California, June. Association for Computational Linguistics.

Liu, P., Qian, K., Qiu, X., and Huang, X. (2017). Idiom-aware compositional distributed semantics. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1215–1224, Copenhagen, Denmark, September. Association for Computational Linguistics.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Miller, G. A. (1995). WordNet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Moilanen, K. and Pulman, S. (2007). Sentiment composition. In *Proceedings of RANLP 2007*, Borovets, Bulgaria.

Nakagawa, T., Inui, K., and Kurohashi, S. (2010). Dependency tree-based sentiment classification using crfs with hidden variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 786–794, Los Angeles, California, June. Association for Computational Linguistics.

Nowak, S. and Rüger, S. (2010). How reliable are annotations via crowdsourcing? a study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the international conference on Multimedia information retrieval - MIR '10*, page 557.

Nunberg, G., Sag, I. A., and Wasow, T. (1994). Idioms. In Stephen Everson, editor, *Language*, pages 491–538. Cambridge University Press.

Polanyi, L. and Zaenen, A. (2004). Contextual valence shifters. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text*.

Rentoumi, V., Vouros, G. A., Karkaletsis, V., and Moser, A. (2012). Investigating metaphorical language in sentiment analysis: A sense-to-sentiment perspective. *ACM Trans. Speech Lang. Process.*, 9(3):6:1–6:31, November.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing: Third International Conference, CICLing 2002 Mexico City, Mexico, February 17–23, 2002 Proceedings*, pages 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Y. (2008). Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA. Association for Computational Linguistics.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October. Association for Computational Linguistics.

Stone, P. J., Dunphry, D. C., Smith, M. S., and Ogilvie, D. M. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press.

Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346, October.

Velikovich, L., Blair-Goldensohn, S., Hannan, K., and McDonald, R. (2010). The viability of web-derived polarity lexicons. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 777–785, Los Angeles, California, June. Association for Computational Linguistics.

Williams, L., Bannister, C., Arribas-Ayllon, M., Preece, A., and Spasić, I. (2015). The role of idioms in sentiment analysis. *Expert Systems with Applications*, 42(21):7375 – 7385.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Xie, S.-x. and Wang, T. (2014). Construction of unsupervised sentiment classifier on idioms resources. *Journal of Central South University*, 21(4):1376–1384, Apr.

PronouncUR: An Urdu Pronunciation Lexicon Generator

Haris Bin Zia[1], Agha Ali Raza[1], Awais Athar[2]
[1]Information Technology University, 6th Floor, Arfa Software Technology Park, Ferozepur Road, Lahore, Pakistan
[2]EMBL-EBI, Wellcome Genome Campus, Hinxton, Cambridgeshire, CB10 1SD, UK
{haris.zia, agha.ali.raza}@itu.edu.pk
awais@ebi.ac.uk

Abstract

State-of-the-art speech recognition systems rely heavily on three basic components: an acoustic model, a pronunciation lexicon and a language model. To build these components, a researcher needs linguistic as well as technical expertise, which is a barrier in low-resource domains. Techniques to construct these three components without having expert domain knowledge are in great demand. Urdu, despite having millions of speakers all over the world, is a low-resource language in terms of standard publically available linguistic resources. In this paper, we present a grapheme-to-phoneme conversion tool for Urdu that generates a pronunciation lexicon in a form suitable for use with speech recognition systems from a list of Urdu words. The tool predicts the pronunciation of words using a LSTM-based model trained on a handcrafted expert lexicon of around *39,000* words and shows an accuracy of *64%* upon internal evaluation. For external evaluation on a speech recognition task, we obtain a word error rate comparable to one achieved using a fully handcrafted expert lexicon.

Keywords: Pronunciation Lexicon, Pronunciation Modeling, Lexicon Learning, Speech Recognition, Urdu

1. Introduction

Automatic Speech Recognition (ASR) for resource scarce languages has been an active research area in the past few years (Sherwani, 2009; Qiao, 2010; Chan, 2012). Modern speech recognition systems usually require three resources: transcribed speech for acoustic modeling, a large text data for language modeling and a pronunciation lexicon that maps words to sub-word units known as phonemes. Pronunciation lexicon acts as a link connecting language model with the acoustic model.

While it is comparatively easy to gather transcribed speech waveforms and large text datasets, developing a pronunciation dictionary is quite expensive and requires tremendous amount of manual effort and linguistic expertise. Therefore, development of a pronunciation lexicon is the bottleneck when building ASR systems for low-resource languages. Techniques to reduce the need of expert knowledge in design and development of pronunciation lexicons are in great demand.

We are interested in developing a pronunciation lexicon generation tool for Urdu which is an Indo-Aryan language spoken widely with over 100 million speakers[1]. Urdu is official language of Pakistan. Its writing system is *Segmental* and more specifically *Abjad* i.e. only consonants are marked while vowels (diacritics) are optional. Urdu follows Arabic script written from right to left. A sentence written in Urdu along with its English translation is given below:

اردو پاکستان کی قومی زبان ہے ۔
Urdu is the national language of Pakistan.

Automatic Speech Recognition (ASR) research for Urdu exhibits number of challenges which are discussed in detail in subsequent sections. Despite being spoken by millions of speakers all over the world, Urdu is low-resource in terms of standard publically available linguistic resources.

To our best knowledge, our Urdu pronunciation lexicon generation tool is the first tool of its kind that makes it easier for researchers to work on Urdu speech recognition systems without prior linguistic knowledge.

The remainder of the paper is structured as follows. Section 2 reviews similar kind of work for different world languages. We then present Urdu orthography and Urdu phonetic inventory in Section 3. Section 4 briefly discusses challenges in Urdu pronunciation modeling. We present our tool in Section 5 and conclude in Section 6.

2. Literature Review

There exists a range of research focusing on lexical resources or tools available for different world languages for pronunciation modeling in speech recognition tasks.

- CMUdict[2] (Carnegie Mellon pronunciation dictionary) is an open-source pronunciation dictionary for North American English that contains over 134,000 words and their pronunciations (Weide, 1998). There is also a lexicon generation tool[3] available that uses CMUdict.

- Tan et al. (2009) proposed a rule based grapheme-to-phoneme tool generating a pronunciation dictionary for Malay language. Their trained ASR on read speech corpus, using tool generated pronunciation dictionary achieved a word error rate (WER) of 16.5%.

- A Bengali pronunciation dictionary[4] was developed under Google Internationalization Project[5] (Gutkin et al., 2016). The dictionary contains around 65,000 words that were manually transcribed into their phonemic representation by a team of five linguists.

[1] https://www.ethnologue.com/language/urd

[2] https://github.com/cmusphinx/cmudict

[3] http://www.speech.cs.cmu.edu/tools/lextool.html

[4] https://github.com/googlei18n/language-resources/blob/master/bn/data/lexicon.tsv

[5] https://developers.google.com/international/

- Pronunciation lexicons were developed for Amharic, Swahili and Wolof languages under LFFA Project[6] and were made available publically[7] (Gauthier et al., 2016).

- Mandarin Chinese Phonetic Segmentation and Tone is a publically[8] available corpus of 7,849 Mandarin Chinese utterances and their phonetic segmentation. The corpus can be used for pronunciation modeling of Mandarin Chinese.

- Arabic Speech Recognition Pronunciation Dictionary is a publically[9] available pronunciation dictionary for Modern Standard Arabic (MSA) that contains 526,000 words and two million pronunciations.

- Masmoudi et al. (2014) presented Tunisian Arabic Phonetic Dictionary based on a set of phonetic rules and manually tagged lexicon of exceptions (for words that do not follow phonetic rules).

- Egyptian Colloquial Arabic Lexicon is a publically[10] available pronunciation dictionary of Egyptian Colloquial Arabic (ECA), it contains 51,202 words and their pronunciation.

- The Georgetown dictionary of Iraqi-Arabic is a modern, up-to-date, publically[11] available dialectal Arabic language resource that can be used for pronunciation modeling of Iraqi-Arabic. It contains 17,500 Iraqi-Arabic entries along with their IPA pronunciations.

- Bonaventura et al. (1998) presented a letter-to-phone conversion system for Spanish that can be used to supply phonetic transcriptions to a speech recognizer.

- Mendonça et al. (2014) proposed a hybrid approach based on manual transcription rules and machine learning algorithms to build a machine readable pronunciation dictionary for Brazilian Portuguese. The dictionary as well as algorithms used to build pronunciation dictionary were made publically[12] available.

Pronunciation dictionaries developed under GlobalPhone Project (Schultz, 2014) are also available for research and commercial purposes in 20 different languages - German, French, Russian, Korean, Turkish, Chinese and Thai to name a few.

3. Urdu Language

3.1 Orthography

Urdu is written in Arabic script in a cursive format (Nastaliq style) from right to left using an extended Arabic character set. The character set includes 37 basic and 4 secondary letters, 7 diacritics, punctuation marks and special symbols (Hussain & Afzal, 2001; Afzal & Hussain, 2001; Hussain, 2004) (see Appendix A).

3.2 Phonetics

Urdu has a very rich phonetic inventory[13], combination of Urdu letters and diacritics realizes 44 consonants (28 non-aspirated & 16 aspirated), 7 long vowels, 7 nasalized long vowels, 3 half long vowels, 3 short vowels and 3 nasalized short vowels (Saleem et al., 2002; Hussain, 2007; Hussain, 2004). Since speech recognition systems require the representation of sounds using some phonemic notation such as IPA[14] or SAMPA[15] etc., we have used CISAMPA (Case Insensitive Speech Assessment Methods Phonetic Alphabet) proposed by Raza et al. (2010) to represent Urdu phonemes (see Appendix B).

4. Challenges in Urdu Pronunciation Modeling

Pronunciation modeling for Urdu exhibits a number of challenges:

Dialects: Due to large user base and variety of speakers, there are variations in dialect leading to large variations in pronunciation and phonetics.

Script: In Urdu, diacritics serve to inform reader of the short vowels accompanying each written consonant, but commonly used Urdu script generally does not contain diacritics. Speakers can distinguish the words through context and experience but some constructions may still be ambiguous, for instance, the word اس can mean either 'this' (اِس) or 'that' (اُس), their respective IPA representation being /ɪs/ or /ʊs/ respectively.

Morphology: Urdu is a morphologically rich language, combinations of affixes and stems results into large vocabulary of words.

Dual Behavior: Three Urdu characters show dual behavior i.e. both consonantal and vocalic, based on their position of occurrence (Hussain, 2004).

5. PronouncUR

We have developed PronouncUR, an Urdu grapheme-to-phoneme tool based on a model (c.f. Section 5.2) that can generate a pronunciation lexicon in a form suitable for use with speech recognition systems from a list of Urdu words. PronouncUR is freely available online[16].

5.1 Lexicon

To train our model we have developed a lexicon of approximately 46K words. Lexicon has been tagged by trained transcription experts, carefully considering the letter-to-sound rules for Urdu proposed by Hussain (2004).

[6] http://alffa.imag.fr/
[7] https://github.com/besacier/ALFFA_PUBLIC
[8] https://catalog.ldc.upenn.edu/LDC2015S05
[9] https://catalog.ldc.upenn.edu/LDC2017L01
[10] https://catalog.ldc.upenn.edu/LDC99L22
[11] http://press.georgetown.edu/book/languages/georgetown-dictionary-iraqi-arabic
[12] https://github.com/gustavoauma/aeiouado_g2p
[13] http://www.cle.org.pk/Downloads/ling_resources/phoneticinventory/UrduPhoneticInventory.pdf
[14] https://www.internationalphoneticassociation.org/
[15] http://www.phon.ucl.ac.uk/home/sampa/
[16] http://lextool.csalt.itu.edu.pk

The format of the training lexicon is very straight forward. Each line consists of one word form and its pronunciation. Word forms and their pronunciations are separated by tab. A small portion of the training lexicon is given in Table 1.

فولاد	F O L A_A D_D
علامات	A L A_A M A_A T_D
جائیداد	D_Z A_A I_I D_D A_A D_D
لڑکیوں	L A R_R K I J O_O_N
درویشی	D_D A R V A_Y S_H I_I
الجھاؤ	U L D_Z_H A_A O_O
رکوا	R U K V A_A
ایران	I_I R A_A N
خریدی	X A R I_I D_D I_I
اُفات	A_A F A_A T_D
فریاد	F A R J A_A D_D
عراقی	I R A_A Q I_I

Table 1: Training Lexicon

Out of 67 phonemes available in Urdu phonetic inventory (see Appendix B), our training lexicon currently caters for 64 phonemes, while the work is in progress to include 3 nasalized short vowels. Phonemes M_H and J_H occur very rarely in Urdu and thus have only one entry each in the training lexicon, for the rest of the phonemes the frequency of occurrence is given in Table 2.

#	Phoneme	Frequency	#	Phoneme	Frequency
1	A	30947	32	Q	2080
2	A_A	27170	33	X	1641
3	R	18386	34	R_R	1562
4	N	15139	35	A_Y_N	1386
5	I_I	13920	36	N_G	1297
6	I	13683	37	A_A_N	1060
7	L	10909	38	K_H	1035
8	M	10538	39	O	928
9	S	10522	40	G_G	800
10	T_D	10075	41	T_S_H	711
11	K	8470	42	B_H	690
12	A_Y	7562	43	I_I_N	660
13	B	7147	44	D_Z_H	571
14	U	6540	45	D_D_H	555
15	T	6024	46	T_D_H	531
16	D_D	5913	47	T_H	495
17	Z	4940	48	P_H	435
18	H	4771	49	G_H	424
19	O_O	4766	50	A_E_H	375
20	P	4742	51	U_U_N	332
21	V	4144	52	R_R_H	225
22	O_O_N	4128	53	D_H	194
23	J	3963	54	O_O_H	70
24	U_U	3581	55	Z_Z	52
25	A_E	3440	56	A_E_N	43
26	S_H	3423	57	Y	36
27	D_Z	3331	58	A_Y_H	33
28	G	3275	59	N_H	12
29	F	3233	60	L_H	8
30	D	2762	61	R_H	8
31	T_S	2491	62	O_N	4

Table 2: Frequency Distribution of Phonemes in Training Lexicon

5.2 G2P Model

The grapheme-to-phoneme (G2P) is the task of translating input sequence of graphemes (letters) to output sequence of phonemes.

Graphemes	ب	◌َ	ن
Phonemes	B	A	N

Table 3: An example of grapheme-to-phoneme translation

Given the success of sequence-to-sequence learning (Sutskever et al., 2014) and power of LSTM for sequence modeling (Hochreiter et al., 1997), we choose LSTM for grapheme-to-phoneme conversion as proposed by Yao et al. (2015). We used open-source G2P toolkit[17] to train our G2P model with 2 LSTM layers and 512 hidden units in each layer.

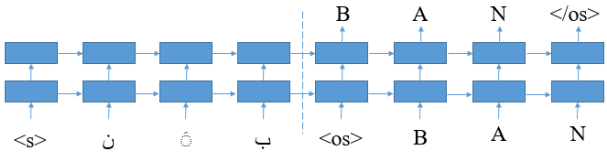


Figure 1: An encoder-decoder LSTM with two layers.

Figure 1 shows a sample of the model where the encoder LSTM is on the left of dotted line while decoder on the right. The encoder reads a time-reversed sequence "<s> ن ◌َ ب" and produces the last hidden layer activation to initialize the decoder. The decoder reads "<os> B A N" as the past phoneme prediction sequence and uses "B A N </os>" as the output sequence to generate. <s> denotes input sequence beginning while <os> and </os> denotes output sequence beginning and ending respectively.

5.3 Performance Evaluation

We split our handcrafted lexicon in 85% training set, 5% validation and 10% test set. Intrinsic evaluation on unseen test set our G2P model achieved word error rate (WER) of 36%. The same G2P model trained on CMUdict has WER of 28.61% (Yao et al., 2015). The low word error rate of CMUdict can be attributed to its large size. Another reason for our comparatively higher WER may be that only about 11% of the words in our corpus have diacritics. As a result, a good performance would require overcoming the problem of automatic diacritization which gets harder while processing a list of isolated words without any context.

To perform extrinsic evaluation of the performance of lexicon tool on speech recognition task, we trained a Hidden Markov Model (HMM) based speech recognition system on phonetically rich Urdu speech corpus[18] (Raza et al., 2009) and spontaneous speech corpus (Raza et al., 2010) using CMUSphinx[19] speech recognition toolkit. The combined data from both corpora contains 3,974 utterances spanning over 179 minutes of speech, out of which 157 minutes (3,174 utterances) were used for training and 22 minutes (800 utterances) for testing. A tri-

[17] https://github.com/cmusphinx/g2p-seq2seq
[18] http://csalt.itu.edu.pk/PRUSCorpus/index.html
[19] https://cmusphinx.github.io/

gram language model using the training data transcripts was applied during decoding. By using lexicon generated through lexicon tool, we obtained a word error rate (~19%) that approaches the rate achieved using a fully handcrafted expert lexicon. We used the same train/test split as used by Raza et al. (2010) and thus results are directly comparable.

6. Conclusion and Future Work

We presented an online pronunciation lexicon generation tool for Urdu that can be used to generate pronunciation lexicon to be used with speech recognition systems. Experimental results showed that pronunciation lexicon generated through lexicon tool behaves as good as handcrafted expert lexicon in speech recognition tasks.

As a future direction, we will look into the ways to decrease the WER of lexicon tool e.g. increase diacritic coverage in training lexicon, increase size of training lexicon, add support for nasalized short vowels and increase the coverage of rarely occurring phonemes.

7. Acknowledgements

We would like to thank Atique-ur-Rehman for providing us with cloud hosting and Murtaza Azam Khan for his help with frontend.

8. Bibliographical References

Afzal, M., & Hussain, S. (2001). Urdu computing standards: development of Urdu Zabta Takhti (UZT) 1.01. In Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International (pp. 216-222). IEEE.

Aminzadeh, A. R., & Shen, W. (2008, December). Low-resource speech translation of Urdu to English using semi-supervised part-of-speech tagging and transliteration. In Spoken Language Technology Workshop, 2008. SLT 2008. IEEE (pp. 265-268). IEEE.

Bonaventura, P., Giuliani, F., Garrido, J. M., & Ortin, I. (1998, August). Grapheme-to-phoneme transcription rules for Spanish, with application to automatic speech recognition and synthesis. In Proceedings of the Workshop on Partially Automated Techniques for Transcribing Naturally Occurring Continuous Speech (pp. 33-39). Association for Computational Linguistics.

Chan, H. Y., & Rosenfeld, R. (2012, March). Discriminative pronunciation learning for speech recognition for resource scarce languages. In Proceedings of the 2nd ACM Symposium on Computing for Development (p. 12). ACM.

Gutkin, A., Ha, L., Jansche, M., Pipatsrisawat, K., & Sproat, R. (2016, May). TTS for Low Resource Languages: A Bangla Synthesizer. In LREC.

Gauthier, E., Besacier, L., Voisin, S., Melese, M., & Elingui, U. P. (2016, May). Collecting resources in sub-saharan african languages for automatic speech recognition: a case study of wolof. In 10th Language Resources and Evaluation Conference (LREC 2016).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Hussain, S., & Afzal, M. (2001). Urdu computing standards: Urdu zabta takhti (uzt) 1.01. In Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International (pp. 223-228). IEEE.

Hussain, S. (2004, August). Letter-to-sound conversion for Urdu text-to-speech system. In Proceedings of the workshop on computational approaches to Arabic script-based languages (pp. 74-79). Association for Computational Linguistics.

Hussain, S. (2007). Phonetic correlates of lexical stress in Urdu (Doctoral dissertation, UMI Ann Arbor).

Masmoudi, A., Khmekhem, M. E., Esteve, Y., Belguith, L. H., & Habash, N. (2014, May). A Corpus and Phonetic Dictionary for Tunisian Arabic Speech Recognition. In LREC (pp. 306-310).

Mendonça, G., & Aluisio, S. (2014). Using a hybrid approach to build a pronunciation dictionary for Brazilian Portuguese. In Fifteenth Annual Conference of the International Speech Communication Association.

Qiao, F., Sherwani, J., & Rosenfeld, R. (2010, December). Small-vocabulary speech recognition for resource-scarce languages. In Proceedings of the First ACM Symposium on Computing for Development (p. 3). ACM.

Raza, A. A., Hussain, S., Sarfraz, H., Ullah, I., & Sarfraz, Z. (2009, August). Design and development of phonetically rich Urdu speech corpus. In Speech Database and Assessments, 2009 Oriental COCOSDA International Conference on (pp. 38-43). IEEE.

Raza, A. A., Hussain, S., Sarfraz, H., Ullah, I., & Sarfraz, Z. (2010). An ASR system for spontaneous Urdu speech. The Proc. of Oriental COCOSDA, 24-25.

Saleem, A. M., Kabir, H. A. S. A. N., Riaz, M. K., Rafique, M. M., Khalid, N. A. U. M. A. N., & Shahid, S. R. (2002). Urdu consonantal and vocalic sounds. CRULP Annual Student Report.

Sherwani, J. (2009). Speech interfaces for information access by low literate users (Doctoral dissertation, Carnegie Mellon University).

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104-3112).

Schultz, T., & Schlippe, T. (2014, May). GlobalPhone: Pronunciation Dictionaries in 20 Languages. In LREC (pp. 337-341).

Tan, T. P., & Ranaivo-Malançon, B. (2009). Malay grapheme to phoneme tool for automatic speech recognition. In Proc. Workshop of Malaysia and Indonesia Language Engineering (MALINDO) 2009.

Weide, R. L. (1998). The CMU pronouncing dictionary. URL: http://www.speech.cs.cmu.edu/cgibin/cmudict.

Yao, K., & Zweig, G. (2015). Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. arXiv preprint arXiv:1506.00196.

9. Language Resource References

Ali, Ahmed. Arabic Speech Recognition Pronunciation Dictionary LDC2017L01. Web Download. Philadelphia: Linguistic Data Consortium, 2017.

Kilany, Hanaa, et al. Egyptian Colloquial Arabic Lexicon LDC99L22. Web Download. Philadelphia: Linguistic Data Consortium, 1997.

Yuan, Jiahong, Neville Ryant, and Mark Liberman. Mandarin Chinese Phonetic Segmentation and Tone LDC2015S05. Web Download. Philadelphia: Linguistic Data Consortium, 2015.

Appendix A

چ	ج	ث	ٹ	ت	پ	ب	ا
ز	ڑ	ر	ذ	ڈ	د	خ	ح
ع	ظ	ط	ض	ص	ش	س	ڑ
ن	م	ل	گ	ک	ق	ف	غ
			ے	ی	ء	ہ	و

Table A1: Basic Urdu Letters

ھ	ۃ	ں	آ

Table A2: Secondary Urdu Letters

ً	ْ	ٰ	ّ	ُ	ِ	َ

Table A3: Urdu Diacritics

Appendix B

Sr. No.	Urdu Letter	IPA	CISAMPA
Consonants			
1	پ	p	P
2	پھ	p^h	P_H
3	ب	b	B
4	بھ	b^h	B_H
5	م	m	M
6	مھ	m^h	M_H
7	ت،ط	$\underline{t}$	T_D
8	تھ	$\underline{t}^h$	T_D_H
9	د	$\underline{d}$	D_D
10	دھ	$\underline{d}^h$	D_D_H
11	ٹ	t	T
12	ٹھ	t^h	T_H
13	ڈ	d	D
14	ڈھ	d^h	D_H
15	ن	n	N
16	نھ	n^h	N_H
17	ک	k	K
18	کھ	k^h	K_H
19	گ	g	G
20	گھ	g^h	G_H
21	ن in نک،نکھ،نگ،نگھ	ŋ	N_G
22	ق	q	Q
23	ع	ʔ	Y
24	ف	f	F
25	و	v	V
26	س	s	S
27	ظ،ض،ز،ذ	z	Z
28	ش	ʃ	S_H
29	ژ	ʒ	Z_Z
30	خ	x	X
31	غ	ɣ	G_G
32	ہ،ح	h	H
33	ل	l	L
34	لھ	l^h	L_H
35	ر	r	R
36	رھ	r^h	R_H
37	ڑ	ɽ	R_R
38	ڑھ	$ɽ^h$	R_R_H
39	ی	j	J
40	یھ	j^h	J_H
41	چ	ʧ	T_S
42	چھ	$ʧ^h$	T_S_H
43	ج	ʤ	D_Z
44	جھ	$ʤ^h$	D_Z_H
Vowels			
45	ُ و	u:	U_U
46	و	o:	O_O
47	َ و	ɔ:	O
48	آ،ا	a:	A_A
49	ی	i:	I_I
50	ے	e:	A_Y
51	َ ے	æ:	A_E
52	ُ وں	ũ:	U_U_N
53	وں	õ:	O_O_N
54	َ وں	ɔ̃:	O_N
55	آں،اں	ã:	A_A_N
56	ِ یں	ĩ:	I_I_N
57	یں	ẽ:	A_Y_N
58	َ یں	æ̃:	A_E_N
59	ہ ِ	e·	A_Y_H
60	ہ َ	æ·	A_E_H
61	ہ ُ	o·	O_O_H
62	ِ	ɪ	I
63	ُ	ʊ	U
64	ء،َ	ə	A
65	ں ِ	ɪ̃	I_N
66	ں ُ	ʊ̃	U_N
67	ں َ	ə̃	A_N

Table B1: Urdu Letters with IPA and CISAMPA

SimLex-999 for Polish

Małgorzata Marciniak[1], **Agnieszka Mykowiecka**[1,2], **Piotr Rychlik**[1]
Institute of Computer Science, Polish Academy of Sciences[1]; Polish-Japanese Academy of Information Technology[2]
Jana Kazimierza 5, Warsaw, Poland; Koszykowa 86, Warsaw, Poland
{mm, agn, rychlik}@ipipan.waw.pl

Abstract

The paper addresses the Polish version of SimLex-999 which we extended to contain not only measurement of similarity but also relatedness. The data was translated by three independent linguists; discrepancies in translation were resolved by a fourth person. The agreement rates between the translators were counted and an analysis of problems was performed. Then, pairs of words were rated by other annotators on a scale of 0–10 for similarity and relatedness of words. Finally, we compared the human annotations with the distributional semantics models of Polish based on lemmas and forms. We compared our work with the results reported for other languages.

Keywords: similarity, synonymy, Polish, distributional models evaluation

1. Introduction

Distributional semantics methods are commonly used in various linguistic tasks. So creating resources to evaluate them is extremely important. The best known of these, (Gabrilovich, 2017) (Finkelstein et al., 2002), consists of 353 word pairs together with their similarity scores. Human annotators evaluated the similarity of word pairs understood as synonymy, relatedness and association of words. Another well-known resource developed for the same purpose is SimLex-999 (Hill, 2017) (Hill et al., 2015). It was created to test similarity, but here it was understood as synonymy and quasi-synonymy, which seems to be a hypernym/hyponym relation or co-hyponymy. It consists of 999 word pairs rated for this interpretation of similarity. The authors clearly excluded relatedness and association from the similarity relation. Moreover, the resource provides information on part-of-speech, abstract vs. concrete concepts, and independent measures of relatedness of pairs of words for English. Both the above resources were translated into German, Italian and Russian (Leviant and Reichart, 2015) and are available as Multilingual WS353 and Multilingual SimLex999 (Leviant and Reichart, 2017), respectively. [1]
Many authors (e.g. (Faruqui et al., 2016), (Chiu et al., 2016)) point out the disadvantages of evaluation for distributional semantics methods based on isolated tests which are not connected to a whole processing system. For languages with less developed linguistic infrastructure, such as Polish, it is important to provide resources for intrinsic evaluation too. Therefore, we decided to translate SimLex-999. As it would be interesting to compare results for synonymy and relatedness on the same resource, we rated both types of similarity relation.

2. Translation

The translation of SimLex-999 into Polish was done by three linguists, native speakers of Polish with good knowledge of English, according to the instruction published together with Multilingual SimLex999. The final translation was agreed by a fourth person with a computational linguistics background. She only resolved cases where differences in the three translations occurred, for 585 pairs in fact. As the Multilingual SimLex999 resource includes translation into Russian, a Slavic language with similar linguistic phenomena and grammar as Polish, we assumed that the most important issues were taken into account. Additionally, we wanted the translators to pay special attention to the following issues, mentioned by Leviant and Reichart in the guidelines.

- As the dataset was intended to test language models, each word should be translated into one word. This caused several problems as many English one-word terms are translated into two words in Polish. 'Sunrise' and 'sunset' in Polish are 'wschód słońca' 'zachód słońca' (shortened by all three annotators to *wschód* and *zachód*, which are polysemic and can also mean e.g. directions of the world). Moreover, many Polish verbs consist of a verb form and the reflexive pronoun *się* 'self' (in Russian a reflexive pronoun is attached to a verb, if it exists). For some verbs, this pronoun is obligatory, e.g. *śmiać się* 'laugh', while for others the existence of the reflexive pronoun changes the meaning of the verb, e.g. *słuchać* 'listen' and *słuchać się* 'obey'. The translators were ask to avoid verbs composed with the reflexive pronoun if possible.

- In Polish, as in Russian, adjectives have different forms for masculine (*stary* 'old'), feminine (*stara* 'old') and neutral (*stare* 'old') genders. In translation, we use masculine forms as dictionaries use them as lemmas.

- The translators should prefer interpretations which make the words in a pair more related, e.g. in the pair of words 'body' and 'chest', for 'chest', the translator chose the interpretation *klatka piersiowa* (shortened to *pierś*) instead of *skrzynia*.

Unfortunately, an accurate translation of an English word into one word in Polish does not always exist. As in all

[1]The list of English words was copied from the original SimLex-999 set but new similarity values were assigned.

repetitions:	exact	reverse order
English	0	1
Italian	4	5
Russian	14	15
German	1	4

Table 1: Pair duplicates in Multilingual SimLex999 files

	T1/T2	T2/T3	T1/T3
common pairs	497	534	535
common words	1316	1374	1410

Table 2: Inter translator agreement

SimLex-999 translations only single words are used, we asked the translators to find a similarly related pair of words which is as close as possible to the original one. For example, 'groom' and 'bride' is in Polish: *pan młody* and *panna młoda*, and might be transformed to: *narzeczony* 'fiance' and *narzeczona* 'fiancée'. There were only a few such situations.

The original SimLex-999 list contains some pairs with similar words, e.g. *bad–awful, bad–terrible*. This can result in creation of two or even more identical pairs after translation. To eliminate this effect, we checked the final agreed data for repetitions, and choose alternative translations for such 22 repeated pairs. In Table 1 we show how may pairs are repeated in other Multilingual SimLex999 data.

Table 2 shows the agreement of the translators of SimLex-999 into Polish. The obtained results are worse than those reported in (Leviant and Reichart, 2015) for Russian, German and Italian, where two translators were involved in each task. The authors reported the following numbers of differently translated 1998 words: Russian 353 words, 17.7%; Italian 196 words, 9.8%; and German 396 words, 19.8%. For Polish, the best inter translator agreement is for T1 and T3 translators; they translated differently 588 words, 29.4%.

One of the reasons for discrepancies in translation are different preferences when choosing one of synonyms, e.g. in the pair *happy–chearful*, the first word was translated identically into *szczęśliwy* while the second one was either *wesoły* or *radosny*, which are near synonyms, or *pogodny*, which is a little more distant but still quite close in meaning. Sometimes, when an English pair consisted of near synonyms, one Polish word was chosen as a translation of either the first or the second word from the pair, e.g. for the pair *weird–strange* a Polish word *dziwny* was an equivalent for both *weird* and *strange*. Another source of differences is the lack of an instruction concerning perfective and imperfective verbs, e.g. *kupić* and *kupować* 'buy', which translators used inconsistently. Polish allows for diminutive forms of nouns (and even adjectives), which was another source of different translations. For example, *wuj* and *wujek* both mean 'uncle', while the first one is more official. Unfortunately, no guidelines were given for spelling. Several English words which are in common use in Polish are written differently in Polish, but the English version is more popular. For example, 'gin' can be written *dżin* or *gin*. Cultural differences meant that translators were looking for the best equivalent out of several possibilities. For instance, the differences in English and Polish education systems meant that the word 'college' was translated into three different words *uczelnia*, *uniwersytet* and *kolegż*. The best translation was the two word term *szkoła wyższa*, but, as multi-word units were excluded, we decided to use the first proposal.

3. Annotation

SimLex-999 contains information on the extent to which two words that make up a pair are similar to each other. The similarity coefficient is the average from many (approx. 50) human (Mechanical Turk) annotations. The similarity was understood here as the semantic equivalence; thus, the words that are synonyms are the most similar to each other. The annotation instruction was not very elaborate and contained two main postulates:

- words that are related are not necessary similar, e.g. *car - tire*,
- "it is perfectly reasonable to use one's intuition ..., especially when you are asked to rate word pairs that you think are not similar at all".

To retain compatibility with the original data set, we gave our annotators the same instruction. However, our annotators were linguists and computer scientists. We got three annotation from our translators who were also instructed to judge the similarity of the Polish words, but each of them only evaluated their own translation. The unified translation results – the MSimLex999_Polish dataset – were annotated by another 7 annotators who did not get the original dataset, only the Polish word pairs. These 7 annotators assigned similarity and relatedness scores to each word pair (two integers between 0 and 10). In this case, there were no formal annotation guidelines. The annotators were instructed to annotate all types of relatedness. In addition to synonymy, they also took into account hyponymy, hyperonymy, cohyponymy, antonymy, and other relations between objects or concepts that might be implied by different situations or contexts in which these objects or concepts appear in.

The final similarity and relatedness scores are average values of all annotations. To test whether this solution is plausible, we check the correlation between all the annotators' scores (pairwise). For similarity judgment the smallest Spearman's correlation coefficient (ρ) was equal to 0.52, while the highest value was equal to 0.71. To see whether the notion of the words from the source language influences the results, we compared average values for those annotators who were translators (AVR_tr) with the average obtained for all other annotators (AVG_nontr), as well as to the final average of all annotations (AVG_all). The results given in Table 3 show that differences were visible, but the final set is equally close to the judgments proposed by both groups of annotators. The average correlation between translators was a little higher (0.67) in comparison to the other group (0.63), but this might have been caused by the smaller number of translators.

	AVG_tr	AVG_nontr	AVG_all
AVG_tr	1	0.82	0.94
AVG_nontr	0.82	1	0.95
AVG_all	0.94	0.95	1

Table 3: Similarity annotation correlation between annotators who were translators and those who were not

The annotations of relatedness were a little more diverse than in the case of similarity. The highest correlation value (ρ) was equal only to 0.68, while the lowest value of the correlation was nearly the same as for the similarity – 0.53. The average correlation of the annotators was 0.59 while the average correlation with the final average ratings was 0.8.

3.1. Correlation of the Annotation with Other Languages

We compared our similarity annotation with the original SimLex-999 annotations and with those obtained by (Leviant and Reichart, 2015) while translating the data into other languages. The Spearman's coefficient for the sequences of all the pairs' similarity judgments is given in Table 4. The agreement with English data is relatively high, with greater agreement observed with respect to the original SimLex-999 annotations. The weakest agreement is observed with the German data (0.74).

dataset	correlation
SimLex-999	0.856
MSimLex999_English	0.816
MSimLex999_Russian	0.793
MSimLex999_German	0.736
MSimLex999_Italian	0.795

Table 4: Cross language similarity agreement (ρ)

The relatedness scores were compared to the association measures calculated for the SimLex-999 by University of South Florida Free Association Norms (Nelson et al., 1998) obtained directly from the SimLex-999 data. The Spearman's ρ in this case is lower (0.54). It is lower even for the correlation with the Polish similarity scores, which is equal to 0.67. The different assumptions made while assigning association scores are already visible when comparing the averages scores from these two sets. In the Free Association Norms, SimLex-999 pairs got on average association of 0.75, while in our annotations the average relatedness score is equal to 5.95. For example, the English pair *new-fresh* has the association value 1.98, while the Polish pair – 8.57, similarly the English pair *sharp-dull* has a score of 1.46, while the Polish equivalent *ostry-tępy* – 7.43.

4. Correlation of the Annotation with Polish Distributed Models

We checked the manually annotated pairs of words against several distributional models of Polish. For this purpose we prepared 16 300-dimensional models based either on forms and lemmas from the combined set of two corpora — National Corpus of Polish (Przepiórkowski, A. et al., 2012) and Polish Wikipedia. All our models were trained with the gensim tool (Řehůřek and Sojka, 2010) using two neural network architectures: Continuous Bag of Words (CBOW) and Skip-Gram (SG), and two algorithms: hierarchical softmax and negative sampling. Most of the models are described in (Mykowiecka et al., 2017a) and are available from (Mykowiecka et al., 2017b). In our experiments we also used two publicly available 100-dimensional CBOW and SG models with negative sampling trained on data from Polish Wikipedia (Rogalski and Szczepaniak, 2016a) available from (Rogalski and Szczepaniak, 2016b) – models 17 and 18 in Table 5. Table 5 contains a list of tested models.

1	f3c-hs	forms, cbow, hierarchical softmax
2	f3c-hs50	forms, cbow, hierarchical softmax, freq. above 50
4	f3c-ns1	forms, cbow, negative sampling, window 1
5	f3c-ns2	forms, cbow, negative sampling, window 2
6	f3c-ns50	forms, cbow, negative sampling, freq. above 50
7	f3s-ns	forms, skip gram, negative sampling
8	f3s-hs	forms, skip gram, hierarchical softmax
9	f3s-hs50	forms, skip gram, hierarchical softmax freq. above 50
10	f3s-ns50	forms, skip gram, negative sampling, freq. above 50
11	l3c-hs	lemmas, cbow, hierarchical softmax
12	l3c-ns	lemmas, cbow, negative sampling
13	l3c-ns1	lemmas, cbow, negative sampling, window 1
14	l3c-ns2	lemmas, cbow, negative sampling, window 2
15	l3s-ns	lemmas, skip gram, negative sampling
16	l3s-hs	lemmas, skip gram
17	plc	forms, cbow, negative sampling,
18	pls	forms, skip gram, negative sampling

Table 5: List of models. Models 1-16 are based on NKJP and Wikipedia and have 300 features. When it is not explicitly stated, the size of context window is 5. Models 17-18 are based on Wikipedia only, have 100 features and context window of the size 5.

The results of the comparison of the obtained scores with the cosine similarities of word embeddings from different models are shown in Figure 1 and 2. The results show that the correlation of the vector similarity with the manually assigned similarity scores are equally good, or even better than the correlation with the relatedness score, for the models which are based on forms and use the CBOW approach. For the form based models which use the skip gram approach and for all lemma based models, the correlation with the relatedness scores is significantly higher than with the similarity scores. Generally, lemma based models show greater correlation with both similarity and relatedness scores. This confirms the intuition that vector similarities correspond to various types of relations and not only similarity, although it was not true for all the tested models. The best model for relatedness scores is the skip gram lemma based model with negative sampling, while for the similarity, the best model is the lemma based CBOW model with negative sampling and the size of window equal

to 1. However, both models are also nearly the best in the other task. Limiting the size of the window has a limited and inconsistent influence on form based CBOW models, while it improved the correlation with the similarity scores and decreased the correlation with the relatedness scores for lemma based models. Thus, for lemma based models our conclusions are consistent with (Chiu et al., 2016).

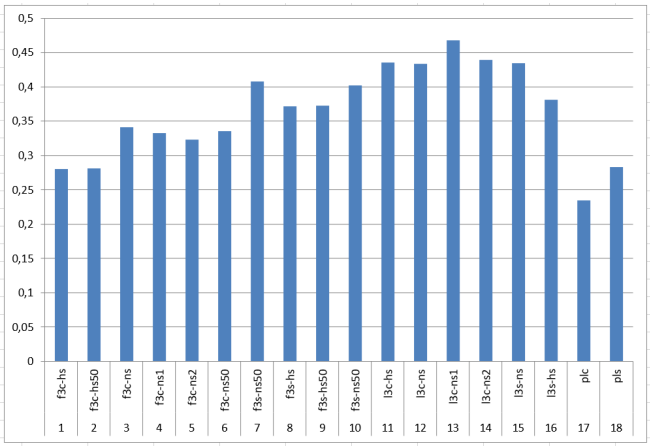


Figure 1: Correlation of MSimLex999_Polish similarities with the cosine similarity of word embeddings for different distributed models

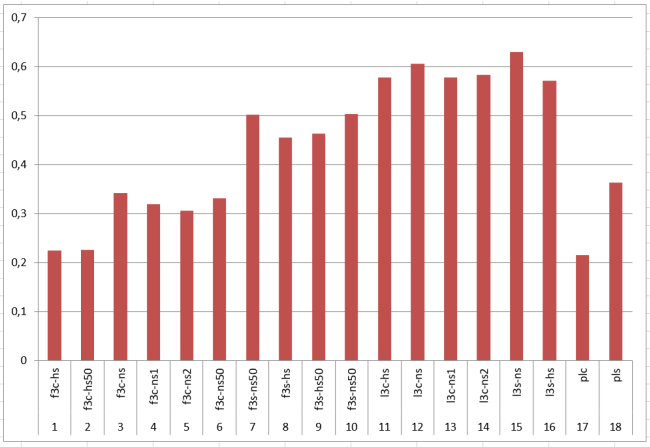


Figure 2: Correlation of MSimLex999_Polish relatedness with the cosine similarity of word embeddings for different distributed models

5. Conclusions

SimLex-999 is a resource which is frequently used as a reference set for evaluating various NLP solutions. Elaborating its Polish version may thus help in making comparisons of results of specific tasks obtained for Polish and other languages. The comparison of the manually obtained list of word similarities with the word embeddings shows that vectors obtained for a smaller data set with the smaller number of features performed worse than the larger models calculated on the bigger corpus. The best model for Polish (trained on NKJP and Wikipedia, 300 features) has the Spearman's ρ correlation with the manually annotated data equal to 0.47; while the model trained on Wikipedia, 100 features has a 0.28 correlation). The best correlation reported in (Leviant and Reichart, 2015) (a model trained on Wikipedia; 400 features) is for German — 0.34. The results obtained for the lemma based models confirmed the intuition that the vector similarity is more likely to resemble relatedness than the similarity of words, but the results for form based models are not so clear.

The relatedness judgment turned out to be more problematic. We obtained values which are different from the association values included in University of South Florida Free Association Norms and which do not show high correlation with word embeddings similarity. There may be different reasons for this and further analysis is needed.

The data are available at (Institute of Computer Science, 2017).

6. Acknowledgments

This work was supported by the Polish National Science Centre project 2014/15/B/ST6/05186 and partially as a part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

7. Bibliographical References

Chiu, B., Korhonen, A., and Pyysalo, S. (2016). Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 1–6. Association for Computational Linguistics.

Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 30–35. Association for Computational Linguistics.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, January.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695, December.

Leviant, I. and Reichart, R. (2015). Judgment language matters: Multilingual vector space models for judgment language aware lexical semantics. *CoRR*, abs/1508.00106.

Mykowiecka, A., Marciniak, M., and Rychlik, P. (2017a). Testing word embeddings for Polish. *Cognitive Studies*, 17.

Przepiórkowski, A., et al., editors. (2012). *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New*

Challenges for NLP Frameworks, pages 45–50, Valletta, Malta, May. ELRA. http://is.muni.cz/publication/884893/en.

Rogalski, M. and Szczepaniak, P. S. (2016a). Word embeddings for the Polish language. In Leszek Rutkowski, et al., editors, *Artificial Intelligence and Soft Computing - 15th International Conference, ICAISC 2016, Zakopane, Poland, June 12-16, 2016, Proceedings, Part I*, volume 9692 of *Lecture Notes in Computer Science*, pages 126–135. Springer.

8. Language Resource References

Evgeniy Gabrilovich. (2017). *The WordSimilarity-353 Test Collection*. http://www.cs.technion.ac.il/~gabr/resources/data/wordsim353/wordsim353.html.

Felix Hill. (2017). *SimLex-999*. https://www.cl.cam.ac.uk/~fh295/simlex.html.

Institute of Computer Science, Polish Academy of Sciences. (2017). *MSimLex999_Polish*. http://zil.ipipan.waw.pl/CoDeS/.

Ira Leviant and Roi Reichart. (2017). *Multilingual SimLex999 and WordSim353*. http://www.leviants.com/ira.leviant/MultilingualVSMdata.html.

Agnieszka Mykowiecka and Małgorzata Marciniak and Piotr Rychlik. (2017b). *Word embeddings for Polish*. http://dsmodels.nlp.ipipan.waw.pl/.

D. L. Nelson and C. L. McEvoy and T. A. Schreiber. (1998). *The University of South Florida word association, rhyme, and word fragment norms*. http://www.usf.edu/FreeAssociation/.

Marek Rogalski and Piotr S. Szczepaniak. (2016b). *pl-embeddings*. http://publications.it.p.lodz.pl/2016/word_embeddings/.

Finely Tuned, 2 Billion Token Based Word Embeddings for Portuguese

João Rodrigues and António Branco
University of Lisbon
NLX-Natural Language and Speech Group, Department of Informatics
Faculdade de Ciências
Campo Grande, 1749-016 Lisboa, Portugal
{joao.rodrigues, antonio.branco}@di.fc.ul.pt

Abstract

A distributional semantics model — also known as word embeddings — is a major asset for any language as the research results reported in the literature have consistently shown that it is instrumental to improve the performance of a wide range of applications and processing tasks for that language. In this paper, we describe the development of an advanced distributional model for Portuguese, with the largest vocabulary and the best evaluation scores published so far. This model was made possible by resorting to new languages resources we recently developed: to a much larger training corpus than before and to a more sophisticated evaluation supported by new and more fine-grained evaluation tasks and data sets. We also indicate how the new language resource reported on here is being distributed and where it can be obtained for free under a most permissive license.

Keywords: distributional semantics, word embeddings, word analogy, lexical similarity, conceptual categorization, Portuguese.

1. Introduction

Distributional semantics assumes that the frequency of contexts in which expressions occur helps to capture important syntactic and semantic properties of these expressions (Garvin, 1962). Exploring distributional models has led to advances in a range of natural language processing tasks, from dialog systems (Chen et al., 2014) to question answering (Bordes et al., 2014), among many others.

A distributional model is a major language resource for any language as it is instrumental to enhance the performance of many applications and processing tasks for that language. Recent work concerned with distributional models for Portuguese contributed with the creation and public release of a free distributional model for this language that resorts to a considerably large corpus, for the training, and to a standard analogy test set, for the evaluation and tuning of this model (Rodrigues et al., 2016).

Since then that corpus has been greatly expanded by our group and is now a very large data set, reaching more than 2 billion tokens.

More recently, finer tuning of distributional models for Portuguese became possible with the creation and distribution of a wide range of data sets by our group that support the evaluation of lexical similarity and conceptual categorization tasks, and that are comparable in size and domain to mainstream datasets with the same purpose for other languages (Querido et al., 2017).

The aim of the present paper is to report on the development, and its free distribution, of a more advanced distributional model for Portuguese, with the largest vocabulary and the best evaluation scores published so far, made possible by a much larger training corpus than before and by a more sophisticated evaluation supported by new and more fine-grained evaluation tasks and data sets.

In Section 2., we discuss previous work in the literature related to the construction and application of distributional models of Portuguese. The methods that were used for the development of the new distributional model are presented in Section 3.. Section 4. describes the data sets used for this purpose, and Section 5. the experiments undertaken to fine-tune the model. We present and discuss the outcome of this development in Section 6.. The paper closes with Section 7., indicating how the resources developed are distributed and can be obtained, and with Section 8. with final remarks.

2. Related Work

Recently, there has been an increased interest in the distributional semantics for Portuguese to support a number of natural language processing tasks, where distributional models tend to be seen as an additional instrumental resource that may help to improve the performance of these tasks. But taking the distributional as our core subject of research, this can alternatively be envisaged under a different perspective. As the development of distributional models is a research topic in itself, applying different models — developed under different design options — in different tasks offers the test bed for the evaluation of these models and of their level of fitness in terms of capturing the essential linguistic information they are meant to encode.

Extrinsic evaluation of these models consists in testing how differences in the design of distributional models induce possible improvements in complex applications or systems where they are embedded. Depending on the level of contribution to the final performance of these systems, it may thus happen that improvements on the models may induce limited to none improvement to those systems. That is why it is important to have also *intrinsic* evaluation, which is meant to give an indication of the appropriateness of the models irrespectively of their eventual impact to larger systems where they may happen to become embedded.

Recent research on the distributional semantics of Portuguese has focused mainly on its eventual instrumental contribution to a number of language processing tasks, and thus directed more to the extrinsic evaluation of distributional models. Examples of this increased interest include named entity recognition (Hermann and Blunsom, 2014),

document classification (dos Santos et al., 2015), language modeling (Ling et al., 2015), topic mining (Machado et al., 2017) and paraphrase detection (Hartmann et al., 2017). Part-of-speech tagging represent the largest share of this interest, with the works of (Al-Rfou et al., 2013), dos Santos and Zadrozny (2014), Fonseca et al. (2015), Fonseca and Aluísio (2016) and Hartmann et al. (2017).

In what concerns the intrinsic evaluation of distributional models, the *word analogy* task is the test bed of choice. The goal is to complete an analogy with a fourth word, for which only the other three words are given. Having, for example, the following analogy, "Berlin is for Germany as Lisbon is for Portugal", and hiding one of the four key terms of the analogy, for example "Portugal", the successful completion of the task will consist of correctly indicating that "Portugal" is the missing term.

This task was one of the first used for the intrinsic evaluation of distributional semantics spaces as its resolution is considered to be tapping in some crucial way on the underlying syntactic and semantic knowledge encoded in the semantic vectors.

An evaluation data set for the word analogy task in Portuguese, LX-4WAnalogies, was first made available by our group, namely Rodrigues et al. (2016), who also developed and distributed the first publicly available word embeddings for Portuguese, LX-DSemVectors, developed on a 1.7 billion token training corpus and tunned on this LX-4WAnalogies data set. This distributional model LX-DSemVectors was obtained with the Skip-Gram neural network from word2vec (Mikolov et al., 2013) and achieves 37.7% accuracy in the word analogy task.

The evaluation data set LX-4WAnalogies was subsequently used by Hartmann et al. (2017) to fine tune another distributional model for Portuguese. This model was developed with Glove (Pennington et al., 2014) on a 1.4 billion token corpus, achieving 46.2% accuracy in the word analogy task.

The intrinsic evaluation of distributional models has been further pursued in the literature, and further tasks have been proposed to complement the assessment supported by the word analogy task. The respective evaluation data sets have been development, most prominently for the English language.

In our group, we have recently pioneered also the development of these types of data sets for the intrinsic evaluation of distributional models (Querido et al., 2017), which we now briefly introduce, together with a description of the respective task.

The task of *lexical similarity* consists of indicating a score, in some predefined scale, that represents the level of semantic similarity between two words that are entered as input. For example, the word "ice-cream" has a weak to none semantic similarity with the word "whale", but a strong similarity to words such as "gelato" or "sorbet", and should be annotated a higher score than the first pair.

The LX-WordSim-353, LX-SimLex-999, and LX-Rare Word Similarity data sets were developed in our group. they contain pairs of Portuguese words together with scores representing their similarity, assigned by human annotators.

The task of *conceptual categorization* consists in clustering a set of words into categories taking into account the semantic relations across those words. For example, given the words "apple", "orange", "sun" and "moon", the task would be to cluster the first two words in one group, corresponding to a fruit related category, and the last two words in a second group, corresponding to an astronomy related category.

The LX-ESSLLI 2008, LX-Battig and the LX-AP datasets were also developed in our group for Portuguese. They permit to evaluate the performance of a resolver for conceptual categorization that consists in clustering the semantic vectors of the words in a predefined number of categories.

Some of these evaluation data sets were already put to use by Oliveira (2017), who evaluated the word embeddings also developed in our group, LX-DSemVectors, under these new tasks by running them over the the LX-SimLex-999, LX-Rare Words Similarity and LX-WordSim-353 datasets.

Saleiro et al. (2017) have also used data sets for word similarity to evaluate word embeddings they trained over a corpus of Tweets. This author also used the Portuguese word embeddings for Portuguese from the Facebook team (Bojanowski et al., 2016), with no evaluation however reported in the respective paper, and evaluated them on the LX-SimLex-999, LX-Rare Words Similarity and LX-WordSim-353 datasets, obtaining 0.34ρ, 0.34ρ and 0.43ρ, respectively — these scores are worse than our best model reported in the present paper, in Section 6..

Against this background, our objectives are twofold: on the one hand, to take advantage of the larger training data set that we developed (described in detail in Section 4.) to develop an enhanced version of our distributional model, which expectedly would ensure a better capture of the linguistic properties of Portuguese words given it can be trained on a substantially larger corpus; on the other hand, and in confluence with the first objective, to take advantage of the additional data sets for intrinsic evaluation we have just distributed in order to support the development a finer tuned distributional model that more appropriately capture the linguistic properties of the Portuguese words.

3. Methods

For the training of the word embeddings for Portuguese, we resorted to the Skip-Gram model, consisting of a shallow neural network that we implemented using the Gensim framework (Řehůřek and Sojka, 2010).

The training of the Skip-Gram model consists on the learning of a distributional semantics space by predicting the words in the neighborhood for the target word of interest, using a sliding window over the words in a given corpus.

With the model eventually trained, each word is mapped to a specific vector in the resulting semantic space. By scoring the distance between the vectors corresponding to two given words along with some metric, typically the cosine distance, syntactic and semantic properties and relations of these words may be captured and support natural processing tasks and applications.

A Skip-Gram model typically supports tasks with better performance when it is trained with larger data sets. For the gathering of the largest possible data set, we used the largest raw corpora previously gathered for Portuguese and publicly described in (Rodrigues et al., 2016), and expanded it

with new data massively web-crawled mostly from newspapers.

For the evaluation and tuning of the resulting distributional semantics space, we resorted to a range of evaluation data sets. These include the analogy test set LX-4WAnalogies already developed and used by (Rodrigues et al., 2016), but also several more, namely the datasets for Portuguese recently made available by Querido et al. (2017): LX-WordSim-353, LX-SimLex-999, LX-Rare Word Similarity, LX-ESSLLI 2008, LX-Battig and the LX-AP. These evaluation datasets were obtained by translating into Portuguese similar ones existing for English, which have been translated also into other languages (Hassan and Mihalcea, 2009) (Joubarne and Inkpen, 2011) (Camacho-Collados et al., 2015) (Freitas et al., 2016) (Cinková, 2016). Given the latter are mainstream datasets, and that these datasets for Portuguese are the most prominent ones existing for this language, this allows thus to develop word embeddings that are mainstream for Portuguese and are comparable with those in English provided large enough data is available for their training.

In the next Section 4., we present in more detail the datasets we used.

4. Datasets

To the best of our knowledge, the largest raw text corpus ever gathered for the Portuguese language, with 1.7 billion tokens, and publicly described in a publication is presented in (Rodrigues et al., 2016). To this data set, we added now over 500 million tokens, corresponding to almost 25 million new sentences, increasing the vocabulary from 873,909 to 1,172,295 items.

These new sentences were mainly gathered in a year time span resorting to the continued crawling of newspaper articles, and the outcome of this crawling was subject to cleaning and tokenization. Table 1 describes the sources of these texts and their respective size.

Once a distributional semantics model is trained with the method described in the previous Section 3., its evaluation can then be accomplished with the support of the evaluation data sets. Next, we provide more information about these datasets, summarized in Table 2.

These corpora can be grouped into three major categories, as summarized in Table 2, namely the ones containing the so-called word analogies, those with pairs with lexical similarity scores and the ones with sets of words supporting conceptual categorization. They are intended to support the evaluation of an equal number of types of natural language processing tasks.

As described above, in the word analogy task, the goal is to complete an analogy with a fourth word, for which only the other three words are given, the accuracy is measured by counting the percentage of given correct answers. The LX-4WAnalogies data set is used to evaluate the performance of resolvers of this word analogy tasks, scored in terms of accuracy. The LX-4WAnalogies contains 17,558 entries of word analogies, each analogy entry containing a set of four words, making a total of 70,232 tokens.

The task of lexical similarity consists of indicating a score representing the level of semantic similarity between two words that are entered as input. The LX-WordSim-353, LX-SimLex-999, and LX-Rare Word Similarity data sets contain pairs of words together with scores representing their similarity, assigned by human annotators.

The LX-WordSim-353 is the smallest of the three similarity data sets, it contains 352 entries, each with two tokens. It follows in size the LX-SimLex-999 with 999 entries and the largest similarity data set, the LX-Rare Word Sim. containing 2,034 entries, both data sets have approximately two tokens for each entry.

They are used to evaluate the resolvers of the lexical similarity task, where the cosine distance of the vectors was measured and compared with the original human score, and the Spearman's rank correlation coefficient metric is calculated over those individual scores.

The Spearman's rank correlation coefficient metric measures the correlation between two rank-ordered scales, in this case, between the resolvers evaluation and the original human score for semantic similarity.

Corpus	Tokens	Sentences	Cat.
Libreoffice	1,456	995	Term.
MSTerminology	38,820	13,030	Term.
Semanario	45,686	2,148	News
JmMadeira	229,197	10,280	News
WikipediaIT	673,932	24,723	Wiki
Abola	838,439	29,260	News
TSF	1,432,332	63,609	News
RegiaoDeLeiria	1,528,711	97,802	News
TMadeira	3,015,973	153,949	News
JornalDoFundao	3,285,362	131,274	News
Sabado	4,527,134	209,848	News
ORegional	4,652,416	203,243	News
OJogo	4,822,878	702,578	News
Euronews	5,355,977	130,672	News
SicNoticias	6,088,051	188,562	News
Lux	7,499,680	358,975	News
IOnline	8,037,427	332,573	News
Sol	8,288,925	323,939	News
Visão	8,987,410	274,193	News
Resistir.info	13,576,530	297,266	News
DN	14,664,275	565,424	News
TVI24	15,332,649	609,173	News
OMirante	17,683,489	705,868	News
CorreioManha	19,171,780	930,494	News
Ocasiao	23,105,092	1,158,281	Sales
OInterior	24,775,869	793,259	News
UniLeipzig	24,933,538	1,000,000	News
JNegócios	27,705,401	791,931	News
Destak	28,137,748	1,148,997	News
DinheiroVivo	37,108,769	1,975,316	News
JornalDiario	43,838,564	1,924,182	News
Expresso	47,272,332	1,893,705	News
Zwame	54,238,235	3,449,424	Forum
JRC-acquis	75,911,681	3,684,145	Law
Total	536,805,758	24,179,118	

Table 1: Novel Portuguese corpora used for the training of word embeddings

Data set	Type	Evaluation Metric	# Entries	Tokens (avg.)
LX-4WAnalogies	Analogy	Accuracy	17,558	70,232 (4)
LX-WordSim-353			352	715 (2)
LX-SimLex-999	Similarity	Spearman's rank	999	2,051 (2.05)
LX-Rare Word Sim.			2,034	4,342 (2.13)
LX-ESSLLI 2008			44 (6 clusters)	44 (1)
LX-Battig	Categorization	Purity	82 (10 clusters)	87 (1.06)
LX-AP			402 (21 clusters)	442 (1.09)

Table 2: Evaluation datasets by type with corresponding evaluation metric

The task of conceptual categorization consists in clustering words into categories. Resorting to the LX-ESSLLI 2008, LX-Battig and the LX-AP datasets, it is possible to evaluate the performance of a resolver for conceptual categorization. The LX-ESSLI 2008 is the smallest of the three conceptual categorization data sets, it contains 44 entries for 6 categories which results in approximately 7 entries per category, each entry is a single word/token. The LX-Battig contains 82 entries for 10 categories, approximately 8 entries per category and 1 word/token per entry. The largest of the categorization data sets, the LX-AP, contains 402 entries for 21 categories, approximately 19 entries per category.
Categorization consists in clustering the semantic vectors of the words in a predefined number of categories. The k-means clustering method is applied and resolvers are scored with the purity metric. Purity measures the accuracy obtained by assigning each cluster to the class which is most frequent in the cluster (Christopher et al., 2008).

5. Experiments

In order to develop the word embeddings for Portuguese, we experimented with different arrangements in terms of the training data set used, taking into account the best practices that have been reported in the literature. We undertook four major experiments where we resorted to the best parameterization setup of the Skip-gram models for Portuguese empirically determined in (Rodrigues et al., 2016), which optimizes the performance of the word analogy task. This corresponds to a Skip-gram parameterization that consists of a vector dimension of 400 units, a 10-word window, a learning-rate of 0.025, using 15 negative samples and a total of 5 epochs.
These four experiments (exp1 to exp4) are described below. Their results are presented in the next Section 3.

- First (exp1), we trained a model using the novel 500+ million token data set we collected, described in section 4.. This experience aims at getting an initial assessment as to whether these new data are cleaned and fit enough to train the word embeddings, by observing whether the performance results on this task are in line with the respective results previously obtained by Rodrigues et al. (2016), even tough with a data set of a different size and composition.

- Second (exp2) we trained a model using the 1+ billion token sub-data set of the whole 1.7 million token dataset that was previously gathered in Rodrigues et al. (2016). This is a replication of a previous experiment in (Rodrigues et al., 2016), only with one difference from it, namely the use of a new and improved version of Gensim (from 0.13.1 to 2.1). The goal here is to make sure that this change is not induce any degradation of the previous level of performance of the best model developed in that previous work.

- Third (exp3), we trained a model on the data set that results from the merging of the two datasets used in exp 1 and exp 2. The aim is to assess whether more data support better performance of the analogy resolver.

- Fourth (exp4), a model was trained with the largest possible data set, which contains the whole 1.7 billion token dataset from Rodrigues et al. (2016) plus the 0.5 billion novel dataset we gathered and describe in the present paper. The goal is to make use of the largest amount of data in order to observe if the models gain from it in accuracy.

Given the large training data, the training of the distributional semantics space is an intensive computational and time-consuming task. The time required for the most intensive task was approximately 220 hours (around 9 days). This value was obtained using 15 dedicated CPUs[1].
In addition to the above four experiments, we carried out three more experiments, which use the same training data as the one used in exp3.

- In a fifth experiment (exp5), the number of training epochs was increased from 5 to 30 in order to assess its impact on the accuracy, given that with a higher number of training epochs the distributional semantic space will be fitter and impact the resolver.

- A sixth experiment (exp6) was undertaken by increasing the vector dimension from 400 to 500 units in order to assess if the vector dimension is enough for the amount of data or a higher data representation is needed.

- In a seventh experiment (exp7), we increased the number of training epochs from 5 to 15 and the negative samples from 15 to 30 in order to assess if repeated negative samples may improve the tasks not related to word analogy.

[1] Intel Core Processor (Haswell, no TSX) @ 2.50GHz model.

Data set	exp1	exp2	exp3	exp4	exp5	exp6	exp7
LX-4WAnalogies	33.5%	43.4%	45.8%	**47.1%**	46.5%	45.9%	45.8%
LX-WordSim-353	0.4614ρ	0.5089ρ	0.5002ρ	**0.5146**ρ	0.4923ρ	0.4937ρ	0.4869ρ
LX-SimLex-999	0.3190ρ	0.3265ρ	0.3341ρ	**0.3502**ρ	0.3239ρ	0.3351ρ	0.3303ρ
LX-Rare Word Sim.	0.3196ρ	0.3325ρ	0.3520ρ	**0.3618**ρ	0.3457ρ	0.3526ρ	0.3501ρ
LX-ESSLLI 2008	0.7045	0.6364	0.6818	0.5909	0.6364	0.5455	**0.7045**
LX-Battig	0.6235	0.6941	**0.8589**	0.8000	0.7294	0.7059	0.7294
LX-AP	0.5297	0.5845	**0.6575**	0.6438	0.5982	0.6301	0.6187

Table 3: Results from the seven experiments (columns) evaluated over the word analogy, semantic similarity and conceptual categorization tasks (rows), with best score in bold.

6. Results and Discussion

In this Section, we discuss the results obtained with the different experiments described in Section 5., which are summarized in Table 3.

Given that the new additional data set originates mainly from the crawling of newspapers, we proceeded with the first experiment (exp 1) in order to assess if this data set (0.5 billion tokens) is suitable for the training of a distributional semantics space.

That happens to be the case as the score obtained for exp1 (33.5% accuracy) is in-line with the best score of the experiments reported in (Rodrigues et al., 2016) (37.7%), for the word analogy task, the only evaluation data set available then.

In the second experiment (exp 2), this state-of-the-art model from Rodrigues et al. (2016) is reproduced with that data set used by then (1 billion tokens) but with the improved version of the Gensim framework.

No degradation of the previous performance was observed, with exp 2 bringing even a better result (43.4%) than the one obtained before (37.7%), thus indicating that, when used, this new version of Gensim is responsible for the improvement of the output model.

In the third experiment (exp 3), a new model was trained on the data set used for the state-of-the-art model from (Rodrigues et al., 2016) merged with the newly collected data set.

Comparing that new model against the models trained with each one of these two data sets separately (exp1 and exp2), a higher accuracy is obtained for almost all evaluation data sets except one, namely LX-WordSim-353 and even in this case with an inferior score by a very thin margin (0.5002ρ against 0.5089ρ). This confirms our hypothesis that the merged data sets (1.5 billion tokens) would support the development of a more appropriate distributional model.

The fourth experiment (exp 4) also confirms the hypothesis that by using the largest amount of data (2.2 billion tokens), one obtains the best scores in the word analogies task and in the semantic similarity tasks. This model scored a 47.1% accuracy with the LX-4WAnalogies evaluation data set, and 0.5146ρ, 0.3502ρ and 0.3618ρ, respectively with LX-WordSim-353, LX-SimLex-999 and LX-Rare Word Sim data sets.

It is worth noting that the scores obtained for the conceptual categorization tasks are lower than the ones obtained with some models trained on smaller data sets.

The fifth, sixth and seventh experiment (exp 5, 6 and 7) used the same training data set as exp 3 (1.5 billion).

By increasing the number of training epochs, in exp 5 only one task got an improvement over exp3, namely the word analogy with 46.5% accuracy, against 45.8% in exp 3. This appears to indicate that the increase of the number of epochs is not bringing a clear advantage to the output model.

In exp 6, by increasing the vector dimension, there was a residual improvement over exp 3 and only on the LX-4WAnalogies (delta of 0.1%), the LX-SimLex-999 (delta of 0.001ρ) and LX-Rare Word Sim (delta of 0.0006ρ). No clear improvements were thus gained by increasing the vector dimension.

Regarding exp 7, where the negative sampling was increased, the results are identical to the word analogy task and worst for all of the data sets in the semantic similarity tasks, when compared to exp 3. The model obtains an improvement in the conceptual categorization task for the LX-ESSLLI data set, matching the best results obtained in exp 1 with a 0.7045 score. The two other conceptual categorization data sets got a worse result than exp 3 though.

The attempts of re-tuning the parameters with exp 5, 6 and 7 yielded no substantive improvements. This seems to indicate that the parameters tuned in (Rodrigues et al., 2016) when using 1 billion tokens by then still deliver pretty good results.

Distributional models for Portuguese previously published in the literature were evaluated only against the word analogy task, with the LX-4WAnalogies data set, scoring 37.7% and 46.2%, respectively by Rodrigues et al. (2016) and Hartmann et al. (2017). Our initial results with the new 2.2 billion token data set under this task, with 47.1% in exp 4, indicate that the model described here is the one with the best score for this task.

The use of larger training data clearly indicates that, as expected, it is a most important factor for the improvement of the scores of the analogy and semantic similarity data sets.

7. Resources distributed

All the distributional semantics models whose development is reported in the present paper are distributed at http://github.com/nlx-group under a Attribution 4.0 International (CC BY 4.0) license. These models are termed as **LX-DSemVectors 2.2b** and users of these language resources should refer to them by citing both the present paper and (Rodrigues et al., 2016).

8. Final remarks

In this paper, we presented an advanced distributional model (aka word embeddings) for Portuguese, LX-DSemVectors 2.2b, with the largest vocabulary and the best intrinsic evaluation scores published so far.

This model permits to score 47.1% accuracy in the word analogy task, with the LX-4WAnalogies data set; 0.5146ρ, 0.3502ρ and 0.3618ρ in the lexical similarity task, respectively with the evaluation data sets LX-WordSim-353, LX-SimLex-999, and LX-Rare Word Similarity; and a purity of 0.5909, 0.8000 and 0.6438 in the conceptual categorization task, respectively with the LX-ESSLLI 2008, LX-Battig and the LX-AP evaluation data sets.

This model is distributed for free under a most permissive license.

The reproducibility of the results reported in this paper can be verified by using this model and obtaining the evaluation data sets, also available for free and under a most permissive license.

Acknowledgment

The present research was partly supported by the Infrastructure for the Science and Technology of the Portuguese Language (CLARIN Língua Portuguesa), by the National Infrastructure for Distributed Computing (INCD) of Portugal, and by the ANI/3279/2016 grant.

9. Bibliographical References

Bordes, A., Chopra, S., and Weston, J. (2014). Question answering with subgraph embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chen, Y.-N., Wang, W. Y., and Rudnicky, A. I. (2014). Leveraging frame semantics and distributional semantics for unsupervised semantic slot induction in spoken dialogue systems. In *Proceedings of the Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 584–589. IEEE.

Christopher, D. M., Prabhakar, R., and Hinrich, S. (2008). Introduction to information retrieval. *An Introduction To Information Retrieval*, 151:177.

dos Santos, C. and Zadrozny, B. (2014). Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1818–1826.

dos Santos, C., Guimaraes, V., Niterói, R., and de Janeiro, R. (2015). Boosting named entity recognition with neural character embeddings. In *Proceedings of the NEWS 2015 The Fifth Named Entities Workshop*, page 25.

Fonseca, E. R. and Aluísio, S. M. (2016). Improving POS tagging across portuguese variants with word embeddings. In *Proceedings of the International Conference on Computational Processing of the Portuguese Language*, pages 227–232. Springer.

Fonseca, E. R., Rosa, J. L. G., and Aluísio, S. M. (2015). Evaluating word embeddings and a revised corpus for part-of-speech tagging in portuguese. *Journal of the Brazilian Computer Society*, 21(1):2.

Garvin, P. L. (1962). Computer participation in linguistic research. *Language*, 38(4):385–389.

Hartmann, N., Fonseca, E., Shulby, C., Treviso, M., Rodrigues, J., and Aluisio, S. (2017). Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *arXiv preprint arXiv:1708.06025*.

Hermann, K. and Blunsom, P. (2014). Multilingual distributed representations without word alignment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ling, W., Luís, T., Marujo, L., Astudillo, R. F., Amir, S., Dyer, C., Black, A. W., and Trancoso, I. (2015). Finding function in form: Compositional character models for open vocabulary word representation. *arXiv preprint arXiv:1508.02096*.

Machado, M. T., Temporal, J. C. A. N., Pardo, T. A. S., and Ruiz, E. E. S. (2017). Mineração de tópicos e aspectos em microblogs sobre dengue, chikungunya, zika e microcefalia. In *Proceedings of the XXXVII Congresso da Sociedade Brasileira de Computação - 17º WIM - Workshop de Informática Médica*.

Oliveira, H. G. (2017). Unsupervised approaches for computing word similarity in portuguese. In *Proceedings of the Portuguese Conference on Artificial Intelligence*. Springer.

Querido, A., Carvalho, R., Rodrigues, J., Garcia, M., Silva, J., Correia, C., Rendeiro, N., Pereira, R., Campos, M., and Branco, A. (2017). LX-LR4DistSemEval: a collection of language resources for the evaluation of distributional semantic models of portuguese. In *Proceedings of the XXXII Encontro Nacional da Associação Portuguesa de Linguística (ENAPL)*.

Rodrigues, J., Branco, A., Neale, S., and Silva, J. (2016). LX-DSemVectors: Distributional semantics models for portuguese. In *Proceedings of the International Conference on Computational Processing of the Portuguese Language*, pages 259–270. Springer.

Saleiro, P., Sarmento, L., Rodrigues, E. M., Soares, C., and Oliveira, E. (2017). Learning word embeddings from the portuguese twitter stream: A study of some practical aspects. In *Proceedings of the Portuguese Conference on Artificial Intelligence*. Springer.

10. Language Resource References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. *CoNLL-2013*, page 183.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2015). A framework for the construction of monolingual and cross-lingual word similarity datasets. In *Proceedings of the ACL Workshop on Evaluating Vector Space Representations for NLP*, pages 1–7.

Cinková, S. (2016). Wordsim353 for czech. In *Proceedings of the International Conference on Text, Speech, and Dialogue*, pages 190–197. Springer.

Freitas, A., Barzegar, S., Sales, J. E., Handschuh, S., and Davis, B. (2016). Semantic relatedness for all (lan-

guages): A comparative analysis of multilingual semantic relatedness using machine translation. In *Proceedings of the Knowledge Engineering and Knowledge Management: 20th International Conference, EKAW 2016*, pages 212–222. Springer.

Hartmann, N., Fonseca, E., Shulby, C., Treviso, M., Rodrigues, J., and Aluisio, S. (2017). Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *arXiv preprint arXiv:1708.06025*.

Hassan, S. and Mihalcea, R. (2009). Cross-lingual semantic relatedness using encyclopedic knowledge. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1201. Association for Computational Linguistics.

Joubarne, C. and Inkpen, D. (2011). Comparison of semantic similarity for different languages using the google n-gram corpus and second-order co-occurrence measures. In *Proceedings of the Canadian Conference on Artificial Intelligence*, pages 216–221. Springer.

Mikolov, T., Yih, W., and Zweig, G. (2013). Linguistic regularities in continuous space word representations. In Lucy Vanderwende, et al., editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751. The Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Querido, A., Carvalho, R., Rodrigues, J., Garcia, M., Silva, J., Correia, C., Rendeiro, N., Pereira, R., Campos, M., and Branco, A. (2017). LX-LR4DistSemEval: a collection of language resources for the evaluation of distributional semantic models of portuguese. In *Proceedings of the XXXII Encontro Nacional da Associação Portuguesa de Linguística (ENAPL)*.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

Rodrigues, J., Branco, A., Neale, S., and Silva, J. (2016). LX-DSemVectors: Distributional semantics models for portuguese. In *Proceedings of the International Conference on Computational Processing of the Portuguese Language*, pages 259–270. Springer.

Teanga: A Linked Data based platform for Natural Language Processing

Housam Ziad, John P. McCrae, Paul Buitelaar
Insight Centre for Data Analytics, National University of Ireland, Galway
[firstname.lastname]@insight-centre.org

Abstract

In this paper, we describe Teanga, a linked data based platform for natural language processing (NLP). Teanga enables the use of many NLP services from a single interface, whether the need was to use a single service or multiple services in a pipeline. Teanga focuses on the problem of NLP services interoperability by using linked data to define the types of services input and output. Teanga's strengths include being easy to install and run, easy to use, able to run multiple NLP tasks from one interface and helping users to build a pipeline of tasks through a graphical user interface.

Keywords: Natural Language Processing, Linked Data, NLP Architecture, NLP Framework

1. Introduction

Natural language processing (NLP) tasks typically consist of many individual components that need to be used together in order to solve real-world problems. However, it is frequently the case that these components are developed independently and thus far integration of these services is far from trivial. The installation of these services can act as a significant barrier to entry for NLP developers and even once developed these pipelines can be opaque and brittle. These issues are of course endemic to software development and until recently could only be solved by integrating all components within a single development model, for example, such as integrating all NLP tools using the Python language as has been done by NLTK(Bird, 2006). An alternative model has arisen in the form of Web services that provide integration between multiple components through clear and defined protocols such as REST. However, Web services have generally not been adopted by researchers or industry, in part due to the fact that the remote nature of the computation can lead to issues with the availability of services (as external services are often down) and the speed of these services (as sending requests to servers creates significant bottlenecks).

In this paper, we propose a new platform called Teanga[1], which aims to achieve the best of both worlds. We use Web services combined by means of novel linked data standards, in particular, JSON-LD, to provide interoperability between services without the need to have particular programming or framework. We also use containerization technology, in particular, Docker, to ensure that these services and the pipelines that are generated by these tools are highly portable and can easily be used at scale. Furthermore, Teanga has an easy to use UI that allows users to visualise their pipelines as well as the progress (or failure) of each service individually.

2. NLP Services

NLP services cover a broad spectrum of tasks, including term extraction, taxonomy extraction, machine translation, sentiment analysis, suggestion mining, automatic summarisation, entity recognition and text classification, which, along with many others, are used widely in many application domains. For example, using entity recognition to extract the names of diseases or medicines from a healthcare database, or using sentiment analysis to extract emotions from social media websites for the purpose of social studies.

NLP services frequently suffer from one or more of the following problems:

- Services are often very focused on a single task, such as part of speech tagging, which is not clearly of use to the end user.
- Many NLP services are still in an early technological readiness level, so there isn't a full application built for them, and/or they don't provide a Graphical User Interface (GUI).
- Many services can't be installed easily, have required dependencies or programming libraries, do not run on all platforms, or lack sufficient documentation.

As a proposed solution to these problems, we present Teanga, an open-source integrated NLP framework. Teanga can apply many NLP tasks either one by one or multiple tasks at once by chaining user-selected services as a pipeline to reduce the amount of manual work required by researchers, in integrating these tools or manually copying results between services.

Teanga will enable researchers to input text in one interface, create a pipeline of required tasks, click a single button, and get all the required results in JSON-LD[2]. We have chosen JSON-LD as the main output format for Teanga as: a) it is easy to use with all programming languages and environments, especially for Web browsers b) provides deep semantics based on RDF and other Semantic Web technologies, c) introduces few overheads to the encoding of the data.

3. Related Work

In the domain of NLP architecture, multiple frameworks, toolkits, and suites have been created, and each of them

[1] 'teanga' ['tʲaŋgə] means 'language' in Irish

[2] https://json-ld.org/

uses a different approach to creating interoperability among all their services, and, by that, reduce the amount of manual work needed to process data. Among these are the LAPPS Grid (Ide et al., 2015) and its Galaxy front-end (Ide et al., 2016), GernEdiT: A Graphical Tool for GermaNet Development (Henrich and Hinrichs, 2010), Language Grid: An Infrastructure for Intercultural Collaboration (Ishida, 2006), and Unstructured Information Management Architecture (UIMA) (Ferrucci and Lally, 2004).

Some problem with the applications of other platforms is that some of them only run on a desktop machine or rely on a platform-specific program, e.g. Eclipse plugins.
For example, in the case of UIMA, it's only a middleware architecture to be taken into account while developing a new NLP tool. For example, it doesn't provide the user with an interface to process data. UIMA also is like GATE when it comes to the complexity of installing and setting up the environment to be used in the development process. Its intended for an expert who is developing an NLP tool and wants to include it in the UIMA environment.

The other platforms do not sufficiently consider user-experience, as user-experience problems can be seen in two parts. The first is installing and running them for the first time, which, for all of them, requires a high level of expertise in a specific environment or programming language, and for some of them, is a time-consuming process. The other part can be seen in the user interface, as some of them don't include a graphical user interface, and users need to run commands from the terminal. Others like GATE created an interface but, even to its developers, (Cunningham, 2002): "The visual interface is complex and somewhat non-standard.". While in the case of LAPPS, their interface seems to be hard to be used by an unskilled user.
A common issue of all of these platforms is the fact that developers have to follow specific standards or guidelines while developing their services before they can be added to the framework to guarantee the interoperability of the platform. While a recent project, OpenMinTeD[3], is working on the standardisation of tools for NLP, these proposals have yet to bear fruit. With Teanga, developers of already existing services can add their services to the platform only by including a configuration file in the container.

4. Linked Data

4.1. Introduction

While data in human-readable formats such as HTML is fully comprehensive for humans, it's still a problem for machines to understand and analyse that data. In the age of big data, where data is expanding exponentially every day, data wrangling still takes up a significant part of the time in the development of an NLP application.
A solution to this as proposed by (Berners-Lee, 2006) gives four key requirements for data to be considered linked data:

- Use URIs to identify all types of data items, for example, if we have a dataset of papers, we would use a unique URI for each paper.
- Make the URIs accessible globally by using HTTP URIs. This way, people can look up the identifiers over the Internet.
- If someone accessed one of the above mentioned URIs, provide useful structured information in RDF.
- Provide links among different data items by including RDF links that point to other URIs; this would help the discovery of related information.

The W3C has proposed a number of standards to help in the creation of linked data, in particular, the Resource Description Framework (RDF), which provides the standard for linked data resources on the Web. In addition, a number of models have been proposed for detailing the semantics of data described in RDF, in particular, the RDF Schema Model (Brickley and Guha, 2000), which allows for inductive reasoning on properties and the Web Ontology Language (McGuinness et al., 2004, OWL), which allows for more sophisticated reasoning about data using description logic.

4.2. Linked Data in Teanga

The use of Linked Data and Semantic Web technologies in applications delivers structured information, which can be used and queried by a flexible and an extensible way to get a better understanding of the data. In particular, the platform exploits the Semantic Web model of data types, to describe the possible format of input and output to services. Since Teanga is designed to deal with any NLP service, and since we can't predict all possible datatypes that may be used by NLP services, we use Semantic Web technologies to define the data types that pass through the services. By doing this, we can let the machine running Teanga understand what data it is processing and how to handle moving it around all the services in a pipeline. The use of Semantic Web technologies will help Teanga, as a platform, to understand the data input and output for each of the services added to the system, and will contribute to creating data interoperability among services to create clear and straightforward pipelines when the user needs to use them.
In particular, there have been a number of models for the representation of linguistic structures used in natural language processing as linked data. The major type of data handled by Teanga is corpus data, and there are a number of models for stand-off annotation of corpora data that have been developed including the NLP Interchange Format (Hellmann et al., 2013) and the Open Annotation format (Sanderson et al., 2013). In addition, more detailed linguistic models such as POWLA (Chiarcos, 2012) as well as specific models such as for parse trees (Chiarcos and Fäth, 2017). In addition, we rely on common models for linguistic categories such as those proposed by ISOcat (Windhouwer and Wright, 2012), now maintained by the CLARIN Concept Registry (Schuurman et al., 2016), or open repositories such as LexInfo (Cimiano et al., 2011)

[3] http://openminted.eu

and OLiA (Chiarcos and Sukhareva, 2015). Finally, we can also use models for representing lexical information on the Web, in particular, the Lexicon Model for Ontologies (McCrae et al., 2012; Cimiano et al., 2016).
As an example, for a machine translation service, we would find that both the source language and target language share the same type because both are referring to a natural language. In this case, we can use an existing type such as `Language` from the LexVo Ontology (de Melo, 2015) and require that values are given as one of the known inputs to this service. We can use JSON-LD aliases to simplify this creating a mapping between the string, e.g., `"en"`, and the URL, e.g., `http://www.lexvo.org/page/iso639-3/eng`). Moreover, for other datatypes such as strings, we can reuse other standards such as XML Schema to define basic datatypes (such as `xsd:string`) or using custom datatypes that can be defined using the OWL vocabulary.
This can be used to join services, for example, if we have a sentiment analysis service that accepts multilingual input and /or output, this service would have an input to enter the text, and an option to select the text language. In this case, the language in the sentiment analysis is of the same type as the languages in the machine translation. If we want to pass data from the sentiment analysis to the machine translation, we can have something as shown in Figure 1.

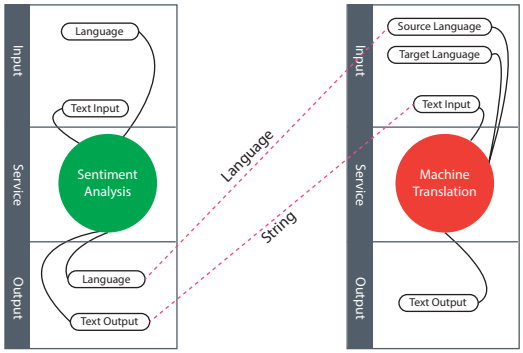


Figure 1: Showing how services share the same types, and how to connect a simple pipeline.

As many of the datatypes used by NLP services are basic or common values such as plain text or language, linked data methods can help the machine understand what type this piece of data is and where to connect that data once we have a pipeline of services. Furthermore, as each of these types is mapped to a URL it is possible to find extra information about it, such as a description, by dereferencing the URL and to provide restrictions, backed by description logic to detect inconsistent combinations of services.

5. Technologies

A key enabling factor for the Teanga platform is the use of the best technologies that exist, in order to enable the service to work efficiently at scale. In particular, we make use of the following open-source tools:

1. Easy-to-use interface by using the Bootstrap[4] library.
2. Stability and maintenance of the Web framework by using the AngularJS[5] library to build the frontend.
3. Using the NodeJS[6] library to run the server and the backend parts.
4. MongoDB[7] is used for data storage, as it uses a JSON-like data structure, which corresponds to our use of JSON-LD files.
5. Using Docker[8] as containerization technology so that the user can download and run Teanga in a simple process of only one step.
6. JSON-LD files for input and output, and to create an interoperable model among the services.

We believe that these technologies will create a highly-performing platform, with an easy-to-use interface, and extremely easy to install and use by third-party users.

6. Design and Implementation

6.1. Introduction

Teanga is designed to host as many NLP services as the users need, for that it should be able to interact with all services and create interoperability among them, especially in the case of running multiple tasks in a pipeline.
Based on that, we had to create a systematic method that helps service developers to add their services to the platform with minimal need to modify their code and architecture to make it compatible with Teanga beforehand. This method must provide three outcomes at once by letting the service provider add a configuration file to the corresponding directory in the platform, with outcomes:

1. Defining the input parameters and types of the service, to make it clear for the interoperability model to understand the service input.
2. Defining the output parameters and types of the service, which makes the service return a compatible output with the platform.
3. By combining the two items above, generating the corresponding user-interface (UI) elements to the user for better interaction with the services.

By using a built-in system to generate the UI, the user or the service provider does not have to write specific code for each service, but rather, by using a format that is widely accepted among Semantic Web applications, the system would create the UI.

6.2. Parameters and Data Types

Any application that interacts with the user would have relevant specific fields in a form or a wizard to collect the data from the user, which can be processed later. These forms and wizards have to interact with the user to collect

[4] `http://getbootstrap.com/`
[5] `https://angularjs.org/`
[6] `https://nodejs.org/`
[7] `https://www.mongodb.com`
[8] `https://www.docker.com`

all corresponding data systematically that the system needs for accurate results.

After analysing many NLP services and considering other cases where we needed to build interfaces for services and applications, we found that we need to build a system that uses semantic structure, JSON-LD files in our case, to generate a user interface automatically and to create interoperability with other services in the platform.

For this, we made our system build an interface depending on the JSON-LD file that describes the service. In this file, we describe the service in details, and then we move to each input and output parameter in it and describe it using linked data technologies.

6.3. Adding a Service to Teanga

Since Teanga is a platform that accepts adding new services to the system, we implemented a simple way to add or remove a service. The method is to add JSON-LD files to the system that represent the service description schema and the context of service components.

The Teanga's file hierarchy contains:

- Ontology: this directory contains the Teanga ontology file, which holds the Teanga ontology that describes all the properties for the services.
- Schema: this directory holds the JSON-LD files that describe the services, one for each service.
- Webapp: This directory has the web applications files and directories, such as the directory named assets, which holds all the static files like images and styles.

To add a new service, we simply need to add a description file for it in the "schema" directory, and to remove the service, we just remove the JSON-LD description file from the directory. As modifying the platform for each service takes some effort, we also scan all loaded Docker images for a scheme directory and load all valid files automatically.

6.4. Containerization Technology using Docker

We're implementing containerization technology in Teanga using Docker to help users overcome the problem of installing many dependencies to run a program. We chose Docker for this task because Docker provides high portability due to a lightweight virtualization with almost no overhead, and that enables running multiple containers on a single machine and adding a layer of abstraction for Teanga. In this case, Teanga would be isolated from other applications on the server, which guarantees performance for the platform and its components inside their containerization environment.
The idea is that we're preparing Teanga inside a downloadable Docker image that holds the Teanga web application and all the libraries, dependencies, and databases elements that it needs to run. This means that the user will only need to download one image that contains everything related to Teanga. By installing a Docker client on their servers, and running the image file, they would have Teanga up and running inside a Docker container by just one step. This is what makes Teanga very easy to install and run. Currently, the Teanga platform is an individual image and the services are in other images; the users can pull them and run them on their servers. It is expected that any third party can easily extend Teanga by simply creating a Docker image and providing an appropriate service description as described above and this image can be published at DockerHub[9] or internal repositories, ensuring that a new service can be added by simply giving the name of its container.

6.5. Error Control

Even perfectly written software can fail due to configuration errors, and most of the software developed by NLP researchers is not developed to industry standards. As such it is a vital goal of the platform to handle failures within services gracefully and show these errors clearly to the user so they may be properly debugged. Teanga can handle the following errors:

- If a service returns an error message, Teanga will display the error message to the user contained inside the results tab.
- If a service fails or has a server error, which usually stops the service and causes it to crash and display default servers messages, Teanga can contain that and return a corresponding message.
- If a service crashed and it returns blank data, Teanga would display an error message that the service is returning an empty message.

We're still working on improving error handling with our continuous experiments with adding services.

6.6. The Interface

We will use an example to describe how the interface works: extracting suggestions from a Spanish text, using a suggestion mining application developed for English.
To run a task in Teanga, we start by entering the text as shown in Figure 2.

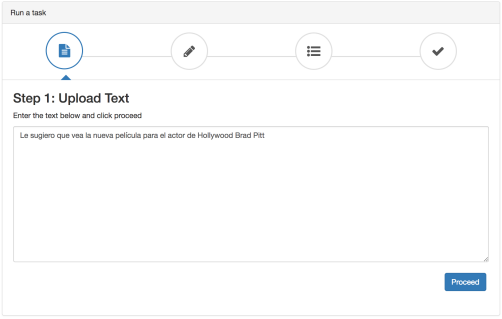


Figure 2: Uploading the text

In Figure 2, we can see that the input text is Spanish, and we need to translate that to English to be ready for the next service, which is suggestion mining.

[9] https://hub.docker.com/

Then, we drag the required services to the service area as in Figure 3. Notice the bottom area, which shows all the available services in the platform represented as a bubble. This means that the system contains three service description files.

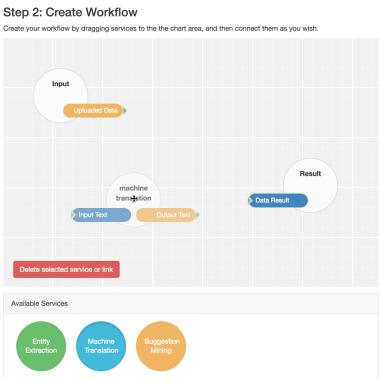


Figure 3: Placing a service in the service area.

Connect the services as shown in in Figure 4. Connecting the service can be done by dragging the points using the mouse.

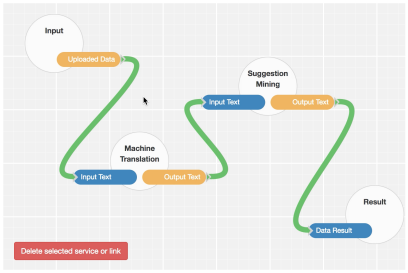


Figure 4: the final workflow graph

Some services do not only have input and output parameters, but they have options as well, e.g. machine translation has options for source and target language. In this step, we select the corresponding options for some of the services as in Figure 5 by selecting the service using the mouse, and then the options sidebar will show up automatically.

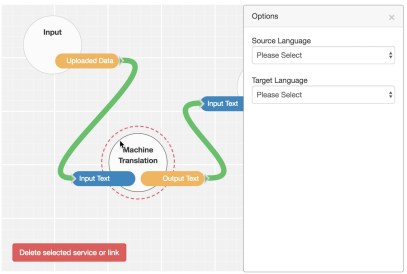


Figure 5: the options panel

And finally proceed to the results page, Figure 6.



Figure 6: The results

6.7. Smart Features in The Interface

As a part of enhancing the user experience, we added two features to Teanga as follows:

1. The predefined tasks, which cover the common experiments that are used in NLP research, we put in a list to choose from, which the system will use to create the whole workflow, saving multiple steps for the user. For example, when the user selects suggestion mining on multilingual text, the system will place the machine translation and the suggestion mining services in the graph, connect them, and then the user only has to choose the languages for the machine translation service.

2. The ability to save and load a saved workflow, in case the user needs to rerun the same experiment in future on a different data set. They can upload the data, and just load the workflow in one click.

7. Testing

Since Teanga is a specialised software, we tested by asking NLP researchers to apply it to their own data. We asked three NLP researchers, two employed at a university and one at a large multinational technology firm, with one of the researchers holding a PhD in NLP.
All the testers agreed that Teanga is well designed and easy to navigate. Yet, they provided 20 comments, of which 13 were easily fixed, while some were kept for future work. The proposed features include:

- Automatic source language detection.
- Downloading the results in JSON-LD files.
- Adding related information to the results page, e.g. how long the experiment took.
- Adding the ability to upload data files to the system instead of direct text entry. These data files can be in different formats.

In addition, some of the comments made us rethink our approach, as in:

- Implementing better ways to choose options for services.
- Changing the view in the last step for the JSON-LD output and adding the ability to dismiss some of the results.
- Enabling the user to choose how to display results.

8. Conclusion

Previous NLP platforms had multiple issues related to their architecture (ability only to apply one task at a time), user experience (no graphical user interface), or interoperability. As a solution for these problems, we developed Teanga, a Linked Data based NLP framework, which aims to enable easily constructed, flexible and high-performance NLP pipelines. We use Linked Data technologies, including JSON-LD files, to define the types of the data that pass through services to deliver structured information that can be used and queried in a flexible and an extensible way and to get a better understanding of the data. We further use a containerization technology (Docker) to ensure that the platform can be easily distributed and installed by end users.

9. Bibliographical References

Berners-Lee, T. (2006). Linked Data - Design Issues. Blog post at www.w3.org.

Bird, S. (2006). NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Brickley, D. and Guha, R. V. (2000). Resource Description Framework (RDF) Schema Specification 1.0: W3C Candidate Recommendation 27 March 2000. W3C Recommendation, World Wide Web Consortium.

Chiarcos, C. and Fäth, C. (2017). CoNLL-RDF: Linked corpora done in an NLP-friendly way. In *International Conference on Language, Data and Knowledge*, pages 74–88. Springer.

Chiarcos, C. and Sukhareva, M. (2015). OLiA–ontologies of linguistic annotation. *Semantic Web*, 6(4):379–386.

Chiarcos, C. (2012). POWLA: Modeling linguistic corpora in OWL/DL. In *Proceedings of the Extended Semantic Web Conference (ESWC)*, pages 225–239.

Cimiano, P., Buitelaar, P., McCrae, J., and Sintek., M. (2011). LexInfo: A declarative model for the lexicon-ontology interface. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(1):29–51.

Cimiano, P., McCrae, J. P., and Buitelaar, P. (2016). Lexicon Model for Ontologies: Community Report. Technical report.

Cunningham, H. (2002). GATE, a general architecture for text engineering. *Computers and the Humanities*, 36(2):223–254.

de Melo, G. (2015). Lexvo. org: Language-related information for the linguistic linked data cloud. *Semantic Web*, 6(4):393–400.

Ferrucci, D. and Lally, A. (2004). UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *Proceedings of the International Semantic Web Conference (ISWC)*, pages 98–113.

Henrich, V. and Hinrichs, E. (2010). GernEdiT: A Graphical Tool for GermaNet Development. In *Proceedings of the ACL 2010 System Demonstrations*, pages 19–24. Association for Computational Linguistics.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., DiPersio, D., Shi, C., Suderman, K., Verhagen, M., Wang, D., and Wright, J. (2015). The Language Application Grid. In *International Workshop on Worldwide Language Service Infrastructure*, pages 51–70. Springer.

Ide, N., Suderman, K., Pustejovsky, J., Verhagen, M., Cieri, C., and Nyberg, E. (2016). The Language Application Grid and Galaxy. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 457–462, Paris, France, may. European Language Resources Association (ELRA).

Ishida, T. (2006). Language Grid: An Infrastructure for Intercultural Collaboration. In *Applications and the Internet, 2006. SAINT 2006. International Symposium on*, pages 1–5. IEEE.

McCrae, J., de Cea, G. A., Buitelaar, P., Cimiano, P., Declerck, T., Gómez-Pérez, A., Gracia, J., Hollink, L., Montiel-Ponsoda, E., Spohr, D., and Wunner, T. (2012). Interchanging lexical resources on the Semantic Web. *Language Resources and Evaluation*, 46(6):701–709.

McGuinness, D. L., Van Harmelen, F., et al. (2004). OWL Web Ontology Language Overview. W3C recommendation, World Wide Web Consortium.

Sanderson, R., Ciccarese, P., Van de Sompel, H., Bradshaw, S., Brickley, D., Castro, L. J. G., Clark, T., Cole, T., Desenne, P., Gerber, A., et al. (2013). Open Annotation data model. W3c community draft.

Schuurman, I., Windhouwer, M., Ohren, O., and Daniel, Z. (2016). CLARIN concept registry: the new semantic registry. In *Selected Papers from the CLARIN Annual Conference 2015*, pages 62–70.

Windhouwer, M. and Wright, S. E. (2012). Linking to linguistic data categories in ISOcat. In *Linked Data in Linguistics*, pages 99–107. Springer.

Automatic and Manual Web Annotations in an Infrastructure to handle Fake News and other Online Media Phenomena

Georg Rehm, Julian Moreno-Schneider, Peter Bourgonje
DFKI GmbH, Alt-Moabit 91c, 10559 Berlin, Germany
{georg.rehm, julian.moreno_schneider, peter.bourgonje}@dfki.de

Abstract

Online media are ubiquitous and consumed by billions of people globally. Recently, however, several phenomena regarding online media have emerged that pose a severe threat to media consumption and reception as well as to the potential of manipulating opinions and, thus, (re)actions, on a large scale. Lumped together under the label "fake news", these phenomena comprise, among others, maliciously manipulated content, bad journalism, parodies, satire, propaganda and several other types of false news; related phenomena are the often cited filter bubble (echo chamber) effect and the amount of abusive language used online. In an earlier paper we describe an architectural and technological approach to empower users to handle these online media phenomena. In this article we provide the first approach of a metadata scheme to enable, eventually, the standardised annotation of these phenomena in online media. We also show an initial version of a tool that enables the creation, visualisation and exploitation of such annotations.

Keywords: Tools, Systems, Applications; Linked Data; Semantic Web

1. Introduction

The amount of online news content is constantly growing. In addition to typical online news outlets, news are more and more consumed through social media channels. While about five or ten years ago only a fraction of the global population got their news online, nowadays online media and social networks are ubiquitous and used by a significant part of the population. By now we have an incredibly high amount of internet users who consume most or all of their news online. While there is still no clear consensus regarding the question whether the outcome of the 2016 US presidential election (Allcott and Gentzkow, 2017) has been manipulated through highly focused social media ads and large bot nets, we, nonetheless, now live in an age in which successful manipulations of online content (news, advertisements, likes, clicks, viral campaigns etc.) can have significant consequences in the real world. It is, however, important to state that there are different types of "fake news" – *false news* is a more appropriate term – and they all come with their own specific intentions and purposes (Rehm, 2018). Often the impact of false news can be harmless (April Fools jokes) or maybe less harmless but still limited in scope (pupils spreading false rumours to bully their peers).

Many online news outlets apply the same journalistic principles that have been in use for newspapers for decades, especially factchecking. However, there are also websites whose primary mission is not to provide high-quality journalistic reporting but, basically, any type of content as long as it produces as many clicks as possible, generating revenue through online advertisements. At first glance, online articles of this second type look just like normal journalistic content. In addition, there are other types of content whose design also mimicks news websites, for example, satire pages. This optical uniformity coupled with the fact that many users only scan the headlines, immediately taking them for fact, makes the World Wide Web and its users susceptible for manipulations and deceptions. This is why online users need to be equipped with additional tools and technologies, they need to be empowered to handle modern online media phenomena, most importantly by helping them to assess the quality of a piece of content, its accuracy, trustworthiness and reputation. Checking the facts and assessing the trustworthiness of online content are increasingly left to the reader. For this purpose, distributed automatic but also semi-automatic approaches can be applied.

In order to empower users to handle the online phenomena mentioned above, in an adequate way, several different approaches could be realised. A consensus seems to emerge that fully automatic means are most likely insufficient properly to address the issue, i. e., we need to combine automatic tools and the wisdom of the crowd through manual or, rather, intellectual assessments in the form of annotations that users attach to a piece of content. Annotations can be made by human users but also by automatic filters, classifiers and watchdogs. In a follow-up step, users can be informed about any issues that the automatic tools or human peers have with the content (manipulated, satirical, imposter content, etc.). In this article we focus on the first steps towards the definition of an annotation scheme to be used both by humans and by machines so that the needed metadata can be added to arbitrary pieces of online content in the form of annotations. We also demonstrate the current version of a browser plugin that allows the creation, visualisation and, eventually, exploitation of the different annotations.

The remainder of this article is structured as follows. First, Section 2. describes related work, while Section 3. briefly sketches the overall infrastructural concept. Section 4. provides an initial draft of a false news annotation schema. Section 5. illustrates the annotation tool. Finally, Section 6. concludes the article.

2. Related Work

Despite the recent increase in research in this area, an effective technological antidote against false news is yet to be found (Rehm, 2018; Rubin et al., 2015). One common denominator of all related work is that they address specific aspects of the broad set of content phenomena. The EU

project Pheme (Derczynski and Bontcheva, 2014), for example, focused on modeling, identifying, and verifying online rumours (Srivastava et al., 2017), which they call "phemes" (internet memes with added truthfulness or deception), as they spread across media, languages, and social networks. (Conroy et al., 2015) looks into different veracity assessment methods emerging from two major categories, i. e., linguistic cues (through machine learning) and network analysis. Martinez-Alvarez (2017) notes in this context: "Fake news is a too general and too vague problem to address directly.", which is why he is splitting it up into smaller, more approachable problems: fact checking, source credibility and trust, news bias and misleading headlines.

Factchecking is a key characteristic of high-quality news content and journalism in general. A large number of factchecking initiatives is active all over the world (Mantzarlis, 2017) but they mostly rely on human expertise and, thus, do not scale (Martinez-Alvarez, 2017; Dale, 2017). The small number of automated fact checking initiatives is fragmented, unreliable and not efficient (Babakar, Mevan and Moy, Will, 2016).

Identifying bias in online articles is another fundamental challenge. Watanabe (2017) analyses the influence of the Russian government on ITAR-TASS during the Ukraine crisis using the state-owned news agency, while Yeo et al. (2017) studied the effect of uncivil comments in online news articles in order to avoid bias interpretations. apply neural networks to identify polarity in news. Valdeón (2017) examines the impact of bias introduced in translations.

Clickbait is often subsumed under the label of "fake news" (Bourgonje et al., 2017). Wei and Wan (2017) identifies misleading headlines – supposed to generate clicks – using class sequential rules to exploit structure information in ambiguous headlines. The BuzzFeed Marketing Challenge (Cowley, 2017) encourages the creation, publication and promotion of an article for generating 1,000 article views in one week. An overview of clickbait analysis approaches was published by Chen and Rubin (2017). Important related characteristics are also the dissemination (Maheshwari, 2016) and spreading (Giglietto et al., 2016) false news exhibit.

Satirical articles are in stark contrast to false news whose objective often is, in the severe cases, to misinform and to manipulate. Rubin et al. (2016) show that online satire often mimics the format and style of journalistic reporting. In addition to false news, other online phenomena need to be taken into account such as abusive language (Nobata et al., 2016; Bourgonje et al., 2018) and hatespeech (Warner and Hirschberg, 2012; Djuric et al., 2015; Schmidt and Wiegand, 2017).

A large number of industrial approaches are focusing on detecting fake news. In early 2017, Facebook announced that they collaborate with the factchecking group Correctiv[1] in Germany (Reuters, 2017). Fakeblok[2] is a Chrome plugin that aims to sanitize the Facebook newsfeed from fake news sites using a curated, factchecked and monitored list of links curated by a group of independent media professionals. Another relevant tool is Fakenews Dataset Analysis, a machine learning system that analyses online news and provides a user-friendly visualisation.[3] Hoaxy[4] visualises how reported claims – and checks of those claims – spread online through social networks. Facebook, Google and seventeen French news organisations joined forces to combat fake news through an initiative called CrossCheck, which uses tools such as CrowdTangle or Spike.[5] There are also several startups in this space like, for example, Factmata, who use NLP and IR algorithms.[6] Among the factchecking initiatives are FactCheck.org,[7] PolitiFact[8] and Fact Checker.[9] These initiatives rely on the claim of Baker (2017): "Human-led fact-checking is the most obvious (and longstanding) weapon against misinformation."

3. Infrastructure Concept

In (Rehm, 2018) we define an infrastructure for the handling and processing of fake news and related phenomena. Here, we give a brief overview of the envisioned system (Figure 1).

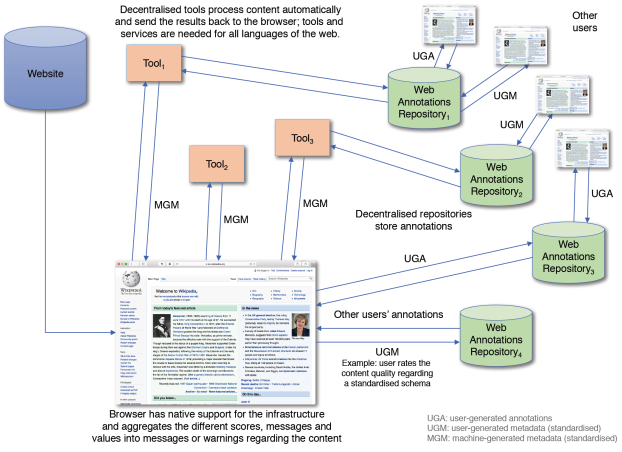


Figure 1: Simplified architecture of the infrastructure

The infrastructure is to be natively embedded into the architecture of the World Wide Web itself. It should rely on standards and be endorsed and supported not only by all browser vendors but also by all content and media providers. The services must be unobtrusive and cooperative, their recommendations and warnings must be clearly understandable. Several pieces are already in place: Web Annotations, standardised by W3C in early 2017 (Sanderson et al., 2017a; Sanderson et al., 2017b; Sanderson, 2017), enable users to annotate arbitrary pieces of web content, creating an independent layer on top of the regular web. They are *the* natural mechanism to enable users to work with content and to include feedback and assessments. Their content can be automatically mined but there are still limitations. Content providers need to enable Web Annotations. Federated

[1] https://correctiv.org
[2] https://fakeblok.com
[3] https://github.com/melphi/fakenews-analysis
[4] http://hoaxy.iuni.iu.edu
[5] https://crosscheck.firstdraftnews.com
[6] http://factmata.com
[7] http://factcheck.org
[8] http://www.politifact.com
[9] https://www.washingtonpost.com/news/fact-checker/

sets of annotation stores are not yet foreseen, neither are native controls in browsers that provide aggregated feedback. Browsers should, ideally, enable free-text annotations and simple flagging of problematic content, e. g., "content pretends to be factual but is of dubious quality". Annotations could be aggregated and presented to new readers to provide guidance and indicate issues. Automatic tools and services can also make use of Web Annotations, e. g., simple classifiers (e. g., regarding abusive language), or sophisticated NLU components that attempt to fact-check statements against knowledge bases. The results can be made available as globally accessible Web Annotations. Also needed is an agreed upon metadata schema (Babakar, Mevan and Moy, Will, 2016) to be used in manual or automatic annotation scenarios. Its complexity should be as little as possible so that key content characteristics can be adequately captured and described by humans or machines. W3C published standards to represent the provenance of objects (Groth and Moreau, 2013; Belhajjame et al., 2013a; Belhajjame et al., 2013b). An alternative approach is Schema.org's ClaimReview markup.[10] Furthermore, the architectural setup must be federated and decentralised to prevent abuse. Annotations must be stored in decentral repositories. These will soon also include more complex data, information and knowledge that tools and services will make use of, e. g., for fact checking. Crowd-sourced knowledge graphs such as Wikidata or DBpedia will continue to grow, the same is true for semantic databases, usually available as Linked Open Data. Already now we can foresee more sophisticated methods of validating and fact-checking content using systems that make use of knowledge graphs, e. g., through entity recognition and linking, relation and event extraction. Finally, we need to be able to aggregate manual and automatic annotations.

4. Annotation Approach and Schema

One of the next steps towards a first prototype is the definition of an annotation schema so that online content can be marked up, both by humans and by machines. We work with an ontology composed of a set of classes and relations that allow automatically processing the annotated data. As the infrastructure is meant to be natively embedded into the web technology stack, we work with the W3C standard for Web Annotations (Sanderson et al., 2017a), which is ideally suited to address the phenomena discussed in this article. The annotation schema includes properties and classes taken from the Provenance Ontology (Prov-O) (Belhajjame et al., 2013a), created for the annotation of the provenance of annotations.

The infrastructure needs to be able to process the following three different types of annotations (Rehm, 2018).

Machine-Generated Metadata (MGM) are automatically generated by a specific service that analyses and annotates a piece of text or multimedia content (image, video, etc.) accordingly, e. g., by assigning a respective score for a given content dimension such as political bias or veracity.

User-Generated Metadata (UGM) are manually annotated by a user through an interface. The user manually assigns a set of predefined tags or scores to express an opinion about the content with the help of a controlled vocabulary, for example, "content is not factual and intentionally misleading".

User-Generated Annotations (UGA) are free text annotations added by a user, i. e., essentially a natural language comment regarding a piece of content.

The annotation schema is defined in an experimental ontology, FANE (Fake News Ontology), which makes use of the Web Annotation standard combined with relevant existing ontologies, such as the Prov-O ontology for provenance information. At the current stage of the implementation the main goal of this experimental schema is to illustrate the overall approach and to demonstrate technical feasiblity. The ontology is not meant to be complete or all-encompassing, for example, currently we are studying the inclusion/mapping of Schema.org's ClaimReview.

In these ontologies we already have all the necessary mechanisms to make annotations in texts or multimedia content in the Web. Therefore, we only need a formal definition of how to annotate the different types of fake news and the values (degree of membership in a fake news type) associated with each of these annotations. Therefore, FANE defines several additional classes and relations (cf. Figure 2).

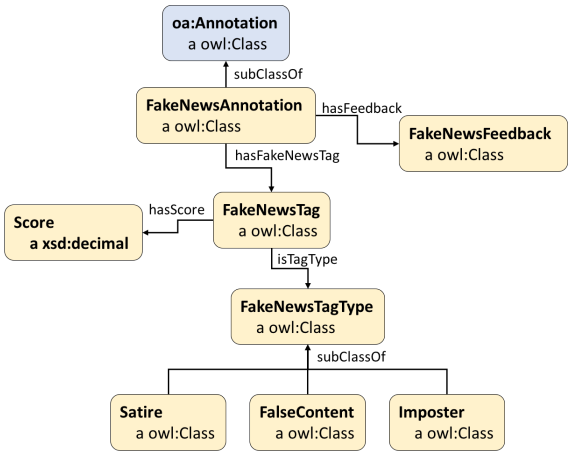


Figure 2: Classes and relations defined in the experimental Fake News Ontology

To simplify the notation, from now on we summarise the URLs, replacing *http://persistence.dfki.de/ontologies/fane-core#* with *"fane:"*. The different classes and properties defined in the ontology are summarised in Table 1.

There are currently seven classes in the ontology (see Table 2). The three relations are described in Table 3.

These classes and relations allow the annotation ("FakeNewsAnnotation") of arbitrary web content in order to tag it ("FakeNewsTag") as different types of false news. The three currently implemented different types ("Satire", "Imposter" and "FalseContent") are defined through the "FakeNewsTagTypes" class allowing the assignment of multiple false news types to the same annotation and are meant to be illustrative examples only.

"FakeNewsFeedback" is a class and mechanism included to allow users to provide feedback on such annotations, which is meant to be a solution for the misuse of such annotations, i. e., a "false news" annotation that has the objective of discrediting real news can be encountered through the feedback

[10] https://schema.org/ClaimReview

Name	Definition
FakeNewsAnnotation	A piece of content annotated by a human user or by an automatic process, on a general level, as false news (can be either UGA, MGM or UGM)
FakeNewsTag	Defines FakeNewsTag associated with the annotation
FakeNewsTagType	Defines the type of the FakeNewsTag
FakeNewsFeedback	Defines feedback about fake news annotations provided by users
Satire	Type Satire
FalseContent	Type False Content
Imposter	Type Imposter
...	*Any additional types of false news*
hasFakeNewsTag	References the FakeNewsTag associated with a false news annotation
isTagType	References the false news type associated with the content
hasScore	Specifies a score associated with the annotation (either MGM or UGM)

Table 1: Classes and properties defined in the Fake News Ontology

of others users affirming that the news is real and that the annotation is false (or vice versa, of course).
In order further to explain the schema, we are going to exemplify it together with an annotated news example (cf. Listing 1) selected from the satirical website The Onion.[11]
The example combines annotations from different ontologies (defined in the namespace part) allowing the use of web annotations and provenance information in a simple way. The main element, "**ex:anno1**", is a web annotation ("**oa:Annotation**") and also a false news annotation ("**fane:FakeNewsAnnotation**"). This annotation is associated with:

- the content defined in "**ex:target1**", which is at the same time associated with a source ("<http://goo.gl/pD9gVE>"), a selector defining the concrete part of text that was annotated and a person who generated the content ("**ex:person2**")
- the person who created the annotation ("**ex:person1**")
- the annotation itself in "**ex:body1**" containing value, format and language
- a creation date ("2017-09-28T16:48:00Z")
- the annotation activity through it was generated "**ex:annotationActivity**"
- the fake news annotations tags "**ex:fnTag1**" and "**ex:fnTag2**"

The fake news information is annotated in "**ex:fnTag1**" and "**ex:fnTag2**", which are fake news tags associated with a fake news tag type ("**fane:isTagType**") and a decimal score or ranking ("**fane:hasScore**"). In this example, the first tag is a "**fane:Satire**" tag, while the second is a "**fane:Imposter**" tag.
Regarding the provenance of the content and annotations, "**ex:person1**" and "**ex:person2**" are physical persons ("**foaf:Person**" and "**prov:Agent**") belonging to companies ("**ex:dfki_gmbh**" and "**ex:the_onion**") who generated content (text) and annotations.

```
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dc:   <http://purl.org/dc/elements/1.1/> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix ex:   <http://example.org/> .
@prefix fane: <http://persistence.dfki.de/ontologies/fane#> .

ex:anno1 a oa:Annotation,prov:Entity,fane:FakeNewsAnnotation ;
  oa:hasBody ex:body1 ;
  oa:hasTarget ex:target1 ;
  oa:motivatedBy oa:describing ;
  dcterms:created "2017-09-28T16:48:00Z" ;
  prov:wasGeneratedBy ex:annotationActivity ;
  prov:wasAttributedTo ex:person1 ;
  fane:hasFakeNewsTag ex:fnTag1, ex:fnTag2 .

ex:body1 a oa:TextualBody ;
  rdf:value "This affirmation is completely false." ;
  dc:format "text/plain" ; dc:language "en" .

ex:target1 a prov:Entity ;
  oa:hasSource <http://goo.gl/pD9gVE> ;
  oa:hasSelector ex:target1_selector ;
  prov:wasAttributedTo :person2 .

ex:target1_selector a oa:TextPositionSelector ;
  oa:start 257 ; oa:end 303 ;
  oa:exact "Obamacare is collapsing under its own weight," .

ex:fnTag1 a fane:FakeNewsTag ;
  fane:hasScore 0.9^^xsd:decimal ;
  fane:isTagType ex:fnTagType1 .

ex:fnTagType1 a fane:FakeNewsTagType, fane:Satire .

ex:fnTag2 a fane:FakeNewsTag ;
  fane:hasScore 0.7^^xsd:decimal ;
  fane:isTagType ex:fnTagType2 .

ex:fnTagType2 a fane:FakeNewsTagType, fane:Imposter .

ex:person1 a foaf:Person, prov:Agent ;
  foaf:givenName "Julian" ;
  prov:actedOnBehalfOf ex:dfki_gmbh .

ex:dfki_gmbh a foaf:Organization, prov:Agent ;
  foaf:name "DFKI GmbH" .

ex:annotationActivity a prov:Activity ;
  prov:wasAssociatedWith ex:person1 ;
  prov:startedAtTime "2011-07-14T01:01:01Z"^^xsd:dateTime ;
  prov:used ex:target1 ;
  prov:endedAtTime "2011-07-14T02:02:02Z"^^xsd:dateTime .

ex:person2 a foaf:Person, prov:Agent ;
  foaf:givenName "Article author" ;
  prov:actedOnBehalfOf ex:the_onion .

ex:the_onion a foaf:Organization, prov:Agent ;
  foaf:name "The Onion" .
```

Listing 1: Content annotation example

5. Current Prototype

As described above and, in detail, in Rehm (2018), our goal is to use a set of decentralised automatic tools and services (that add MGM to content) in tandem with information added by users (UGM, UGA). The respective annotations are stored in decentralised Web Annotation repositories. Whenever a user retrieves online content to be rendered in

[11] http://www.theonion.com

Name	URI	Label	SubClass of
FakeNewsAnnotation	fane:FakeNewsAnnotation	Fake News Annotation	http://www.w3.org/ns/oa#Annotation
FakeNewsTag	fane:FakeNewsTag	Fake News Tag	
FakeNewsTagType	fane:FakeNewsTagType	Fake News Tag Type	
FakeNewsFeedback	fane:FakeNewsFeedback	Fake News Feedback	
Satire	fane:Satire	Satire	fane:FakeNewsTagType
FalseContent	fane:FalseContent	False Content	fane:FakeNewsTagType
Imposter	fane:Imposter	Imposter	fane:FakeNewsTagType

Table 2: Classes in the Fake News Ontology

Name	URI	Label	Domain	Range
hasFakeNewsTag	fane:hasFakeNewsTag	has FN Tag	fane:FakeNewsAnnotation	fane:FakeNewsTag
isTageType	fane:isTagType	is Tag Type	fane:FakeNewsTag	fane:FakeNewsTagType
hasScore	fane:hasScore	has Score	fane:FakeNewsTag	xsd:decimal

Table 3: Relations in the Fake News Ontology

the browser, the browser then retrieves the available information about the content (MGM, UGM, UGA) from the currently configured repositories, aggregates them into easily consumable values and displays these values to the user, for example, through a traffic light metaphor or through a set of reputation and confidence scores.

The prototype relies on Web Annotations, which are not yet natively supported by all browsers. As soon as there is native support in all browsers, the solution we propose will develop its full potential, i. e., users will be able automatically to get clear signals and recommendations with regard to the content they are currently seeing in their browsers – for example, whether to trust it or to take it with a grain of salt. We are currently developing a prototype of such a browser feature in the form of a plugin, which offers the possibility of adding UGA and UGM to content. The implementation is mostly a technical challenge; the process of annotating web resources does not rely only on the graphical interface, but also on servers in the backend that enable automatic content processing and the generation of annotations (text classification, dealing with author/source information, storing the resulting annotations, etc.).

Most of these features are present in the Web Annotation infrastructure provided by Hypothes.is.[12] The Hypothes.is ecosystem not only allows the annotation of online content, but also the annotation of comments and also annotations made by users (UGM, UGA) or machines (MGM). The Hypothes.is tools are currently tailored to Chrome but as soon as Web Annotations are natively available in all browsers, the solution we propose will be universally available without the cumbersome installation of needed plugins.

Our current prototype consists of three main components: the client, the server and the web extension (GUI). For now we use the Hypothes.is infrastructure as our client/server architecture. On the interface level we are making the necessary modifications so that users can not only add and visualise annotations based on the annotation schema (Section 4.), but also provide feedback. The interface (cf. Figure 3) is an adapted version of the Hypothes.is extension, through which users can automatically annotate web content, including numeric scores for different types of false news (Rehm, 2018).

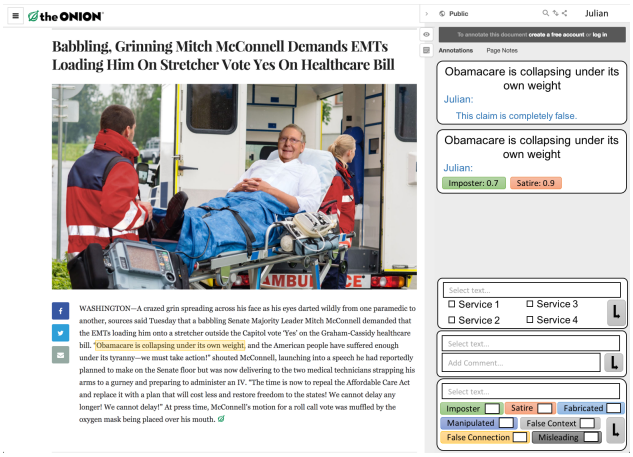


Figure 3: Annotated online content

As regards automatic classification services (MGM), we have conducted several experiments, especially for rumours, for clickbait content and for abusive language and hate-speech (Srivastava et al., 2017; Bourgonje et al., 2018; Bourgonje et al., 2017). We are currently working on attaching these experimental services to the backend so that respective analysis results are automatically shown to the user.

6. Conclusions and Future Work

There is currently a lot of interest in the wider Computational Linguistics, Language Technology and AI community on the general topic of fake news and related online content phenomena as can be seen by the high number of dedicated workshops, e. g., Fake News Challenge,[13] Abusive

[12] https://web.hypothes.is

[13] http://www.fakenewschallenge.org

Language Workshop,[14] Computational Fake News Analysis,[15] the SemEval task on Rumour Evaluation[16] and the Clickbait Challenge,[17] among others. This interest shows that there is a real need and high demand for solutions. The infrastructural approach suggested in (Rehm, 2018), further extended with practical next steps towards an annotation schema and prototype client application in this paper, is an attempt at providing an umbrella application scenario that is both practical, universally applicable in the real world by relying on W3C standards and actually usable and flexible enough to take on board many different types of decentralised classifiers and automatic services.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable comments and suggestions.
The project "Digitale Kuratierungstechnologien" (DKT) is supported by the German Federal Ministry of Education and Research (BMBF), "Unternehmen Region", instrument Wachstumskern-Potenzial (no. 03WKP45). More information: http://www.digitale-kuratierung.de.

7. Bibliographical References

Allcott, H. and Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36, May.

Babakar, Mevan and Moy, Will. (2016). The State of Automated Factchecking – How to make factchecking dramatically more effective with technology we have now. https://fullfact.org/media/uploads/full_fact-the_state_of_automated_factchecking_aug_2016.pdf.

Baker, T. (2017). Fake news: What is it, and how can we tackle it? http://www.nesta.org.uk/blog/fake-news-what-it-and-how-can-we-tackle-it.

Belhajjame, K., Cheney, J., Corsark, D., Garijo, D., Soiland-Reyes, S., Zednik, S., and Zhao, J. (2013a). PROV-O: The PROV Ontology. W3C Recommendation, World Wide Web Consortium (W3C), April. https://www.w3.org/TR/2013/REC-prov-o-20130430/.

Belhajjame, K., Deus, H., Garijo, D., Klyne, G., Missier, P., Soiland-Reyes, S., and Zednik, S. (2013b). PROV Model Primer. W3C Working Group Note, World Wide Web Consortium (W3C), April. https://www.w3.org/TR/2013/NOTE-prov-primer-20130430/.

Bourgonje, P., Schneider, J. M., and Rehm, G. (2017). From Clickbait to Fake News Detection: An Approach based on Detecting the Stance of Headlines to Articles. In Octavian Popescu et al., editors, *Proceedings of the Second Workshop on Natural Language Processing meets Journalism – EMNLP 2017 Workshop (NLPMJ 2017)*, pages 84–89, Copenhagen, Denmark, September. 7. September.

Bourgonje, P., Schneider, J. M., and Rehm, G. (2018). Automatic Classification of Abusive Language and Personal Attacks in Various Forms of Online Communication. In Georg Rehm et al., editors, *Language Technologies for the Challenges of the Digital Age: 27th International Conference, GSCL 2017, Berlin, Germany, September 13-14, 2017, Proceedings*, number 10713 in Lecture Notes in Artificial Intelligence (LNAI), pages 180–191, Cham, Switzerland, January. Gesellschaft für Sprachtechnologie und Computerlinguistik e.V., Springer. 13/14 September 2017.

Cavalcanti, R. D., Lima, P. M., De Gregorio, M., and Menasche, D. S. (2017). Evaluating weightless neural networks for bias identification on news. In *Networking, Sensing and Control (ICNSC), 2017 IEEE 14th International Conference on*, pages 257–262. IEEE.

Chen, Y. and Rubin, V. L. (2017). Perceptions of clickbait: A q-methodology approach. In *Proceedings of the 45th Annual Conference of The Canadian Association for Information Science/L'Association canadienne des sciences de l'information (CAIS/ACSI2017), Ryerson University, Toronto, May 31-June 2, 2017.*

Conroy, N. J., Rubin, V. L., and Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, ASIST '15, pages 82:1–82:4, Silver Springs, MD, USA. American Society for Information Science.

Cowley, S. W. (2017). The buzzfeed marketing challenge: An integrative social media experience. *Marketing Education Review*, 27(2):109–114.

Dale, R. (2017). NLP in a post-truth world. *Natural Language Engineering*, 23(2):319–324.

Derczynski, L. and Bontcheva, K. (2014). Pheme: Veracity in digital social networks. In Iván Cantador, et al., editors, *Posters, Demos, Late-breaking Results and Workshop Proceedings of the 22nd Conference on User Modeling, Adaptation, and Personalization co-located with the 22nd Conference on User Modeling, Adaptation, and Personalization (UMAP2014), Aalborg, Denmark, July 7-11, 2014.*, volume 1181 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., and Bhamidipati, N. (2015). Hate speech detection with comment embeddings. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, pages 29–30, New York, NY, USA. ACM.

Giglietto, F., Iannelli, L., Rossi, L., and Valeriani, A. (2016). In *Convegno annuale dell'Associazione Italiana di Comunicazione Politica (AssoComPol 2016), Urbino, Dec 15-17, 2016.*

Groth, P. and Moreau, L. (2013). PROV-Overview: An Overview of the PROV Family of Documents. W3C Working Group Note, World Wide Web Consortium (W3C), April. https://www.w3.org/TR/prov-overview/.

Maheshwari, S. (2016). How Fake News Goes Viral: A Case Study. https://www.nytimes.com/2016/11/20/business/media/how-fake-news-spreads.html?mcubz=0.

Mantzarlis, A. (2017). There are now 114 fact-checking initiatives in 47 countries. https://www.poynter.org/news/there-are-now-114-fact-checking-initiatives-47-countries.

[14] https://sites.google.com/site/abusivelanguageworkshop2017

[15] http://www.sobigdata.eu/computational-fake-news-analysis-practical-workshop

[16] http://alt.qcri.org/semeval2017/task8/

[17] http://www.clickbait-challenge.org

Martinez-Alvarez, M. (2017). How can machine learning and ai help solving the fake news problem? https://miguelmalvarez.com/2017/03/23/how-can-machinelearning-and-ai-help-solving-the-fake-news-problem/.

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., and Chang, Y. (2016). Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 145–153, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Rehm, G. (2018). An Infrastructure for Empowering Internet Users to handle Fake News and other Online Media Phenomena. In Georg Rehm et al., editors, *Language Technologies for the Challenges of the Digital Age: 27th International Conference, GSCL 2017, Berlin, Germany, September 13-14, 2017, Proceedings*, number 10713 in Lecture Notes in Artificial Intelligence (LNAI), pages 216–231, Cham, Switzerland, January. Gesellschaft für Sprachtechnologie und Computerlinguistik e.V., Springer. 13/14 September 2017.

Reuters. (2017). Facebook germany announces tools to tackle fake news ahead of election. http://www.abc.net.au/news/2017-01-16/facebook-germany-says-it-will-start-tackling-fake-news-in-weeks/8184436.

Rubin, V. L., Chen, Y., and Conroy, N. J. (2015). Deception detection for news: Three types of fakes. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, ASIST '15, pages 83:1–83:4, Silver Springs, MD, USA. American Society for Information Science.

Rubin, V., Conroy, N., Chen, Y., and Cornwell, S. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*, pages 7–17.

Sanderson, R., Ciccarese, P., and Young, B. (2017a). Web Annotation Data Model. W3C Recommendation, World Wide Web Consortium (W3C), February. https://www.w3.org/TR/2017/REC-annotation-model-20170223/.

Sanderson, R., Ciccarese, P., and Young, B. (2017b). Web Annotation Vocabulary. W3C Recommendation, World Wide Web Consortium (W3C), February. https://www.w3.org/TR/2017/REC-annotation-vocab-20170223/.

Sanderson, R. (2017). Web Annotation Protocol. W3C Recommendation, World Wide Web Consortium (W3C), February. https://www.w3.org/TR/2017/REC-annotation-protocol-20170223/.

Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10.

Srivastava, A., Rehm, G., and Schneider, J. M. (2017). DFKI-DKT at SemEval-2017 Task 8: Rumour Detection and Classification Using Cascading Heuristics. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 477–481, Vancouver, Canada, August. Association for Computational Linguistics.

Valdeón, R. A. (2017). Political and sexist bias in news translation: Two case studies. *Trans. Revista de Traductología*, (11):231–243.

Warner, W. and Hirschberg, J. (2012). Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, LSM '12, pages 19–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Watanabe, K. (2017). Measuring news bias: Russia's official news agency itar-tass'coverage of the ukraine crisis. *European Journal of Communication*, 32(3):224–241.

Wei, W. and Wan, X. (2017). Learning to identify ambiguous and misleading news headlines. *arXiv preprint arXiv:1705.06031*.

Yeo, S. K., Su, L. Y.-F., Scheufele, D. A., Brossard, D., Xenos, M. A., and Corley, E. A. (2017). The effect of comment moderation on perceived bias in science news. *Information, Communication & Society*, pages 1–18.

The LODeXporter: Flexible Generation of Linked Open Data Triples from NLP Frameworks for Automatic Knowledge Base Construction

René Witte and Bahar Sateli
Semantic Software Lab
Concordia University, Montréal, QC, Canada
http://www.semanticsoftware.info

Abstract

We present LODeXporter, a novel approach for exporting Natural Language Processing (NLP) results to a graph-based knowledge base, following Linked Open Data (LOD) principles. The rules for transforming NLP entities into Resource Description Framework (RDF) triples are described in a custom mapping language, which is defined in RDF Schema (RDFS) itself, providing a separation of concerns between NLP pipeline engineering and knowledge base engineering. LODeXporter is available as an open source component for the GATE *(General Architecture for Text Engineering)* framework.

Keywords: Linked Open Data, Automatic Knowledge Base Construction, Semantic Web

1. Introduction

Knowledge bases in standardized graph-based formats have become increasingly important in modern applications, such as personal intelligent assistants. Due to the large amount of formalized knowledge required, automatic knowledge base (KB) construction is a crucial step that typically spans multiple information sources (such as structured databases, sensors, and documents). A significant amount of knowledge resides in natural language text and can be automatically extracted through existing or custom-made Natural Language Processing (NLP) components and pipelines. Yet, so far no generic solution existed for exporting the results of an NLP pipeline into a knowledge base.

Our contribution is a novel technique for converting the results of an NLP pipeline into an LOD-compliant KB, based on the Resource Description Format (RDF) (Cyganiak et al., 2014) and RDF Schema (RDFS) (Brickley and Guha, 2014) W3C standards. The concrete mapping of NLP results to entities in the knowledge base is defined through a mapping language, which is itself written in RDFS. This provides high flexibility, as the same NLP pipeline can drive the generation of triples in different KBs with different vocabularies. It also relieves the NLP engineer from dealing with the technical details of generating correct Uniform Resource Identifiers (URIs), thereby providing an agile solution for bringing NLP results onto the Linked Open Data (LOD) cloud (Heath and Bizer, 2011).

LODeXporter has been implemented based on the *General Architecture for Text Engineering* (GATE) framework (Cunningham et al., 2013) and is available as open source (LGPLv3) on GitHub.[1]

2. Foundations

In this section, we provide brief background information on the foundations of our approach.

Linked Data. Linked Open Data (LOD) (Heath and Bizer, 2011; Wood et al., 2014) is the practice of using semantic web technologies, like RDF, to publish structured, open datasets that can be interlinked and shared automatically between machines. All data within the LOD cloud have unique URIs. When dereferenced, the URIs provide useful, machine-readable information about that entity, as well as links to other related entities on the web of data.

Automatic Knowledge Base Construction. As semantic knowledge bases, triplestores are especially well suited for connecting previously isolated knowledge sources (so-called "information silos"). Towards this end, various solutions have been developed for converting existing sources to a knowledge base, e.g., RDB2RDF for creating RDF data from a relational database.[2] Natural language documents are an abundant source of rich semantic knowledge. One of the best known LOD datasets, *DBpedia*,[3] is a large knowledge base that is automatically generated from the *Wikipedia* (Bizer et al., 2009), with several billions of in-bound links from other open datasets. Similarly, custom solutions exist that convert the results of a specific NLP pipeline into a pre-defined knowledge base.

However, so far there existed no generic solution for populating a knowledge base with the results from a text mining pipeline. While it is always possible to develop custom export strategies, this is not a practicable solution in a modern agile data science workflow, where it often becomes necessary to experiment with different knowledge bases and representation vocabularies; typical examples are shared tasks, where the expected format of the knowledge base is prescribed and existing NLP pipelines need to be adapted to a challenge-specific format.

3. Design of LODeXporter

We now describe our design decisions behind LODeXporter in detail.

[1] LODeXporter on GitHub, https://github.com/SemanticSoftwareLab/TextMining-LODeXporter

[2] Relational Databases to RDF (RDB2RDF), https://www.w3.org/2001/sw/wiki/RDB2RDF

[3] DBpedia, http://wiki.dbpedia.org/

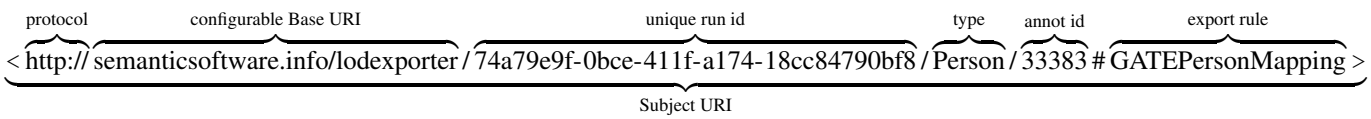


Figure 1: Anatomy of a LODeXporter-generated Subject URI

3.1. Requirements

Our high-level vision is the design of a new NLP component that can be added to any existing NLP pipeline, thereby providing functionality to export the text analysis results in linked data format. We derived a number of detailed requirements from this vision:

Scalability (R1). For the population of large knowledge bases, it must be possible to scale out the generation of triples, i.e., running pipelines on multiple (cloud) instances and storing the resulting triples in a networked triplestore. The time required to export the annotations as triples must scale linearly with the size of the documents and the number of triples to be exported.

Separation of Concerns (R2). Apart from adding a new component to an analysis pipeline, no further changes must be required on the NLP side. Thus, we separate the work of a language engineer, developing the NLP pipeline (e.g., to find domain-specific entities) from the work of a knowledge base engineer, who defines the structure of a concrete KB.

Configurability (R3). The export must be dynamically configurable, so that the same NLP pipeline can drive the generation of different triples in the same, or multiple different, knowledge bases. In particular, the vocabularies used in the mapping process must be easily changeable, to support experiments with different knowledge bases and different application scenarios.

LOD Best Practices (R4). The solution must conform to the relevant W3C standards on Linked (Open) Data. This includes the recommended format for the generated triples as well as the support of standard protocols for communicating with triplestores in a product-independent fashion.

3.2. Mapping NLP Annotations to Triples

A core idea of our approach is that the concrete mapping, from NLP results to knowledge base, is not part of the NLP pipeline itself: Rather, it is externalized in form of a set of *mapping rules* that are encoded in the knowledge base itself. In other words, only by dynamically connecting an NLP pipeline to a knowledge base is the concrete mapping effected: The same pipeline can so be used to export different results, using different mapping rules, to multiple knowledge bases. This is a key feature to enable agile data science workflows, where experiments with different formats can now be easily and transparently set up.

The mapping rules themselves are defined in form of RDF triples as well, based on our mapping vocabulary. We provide mappings for common NLP result types, in particular annotations, annotation features, and annotation relations. Additionally, a number of meta-features about the NLP pipeline itself can be defined for storage in a KB.

URI Generation. One of the core features of LODeXporter is the generation of URIs for LOD triples from text **(R4)**. In designing the URI generation scheme, we had to strike a balance between *(a)* conforming to LOD best practices; *(b)* taking into account the NLP source of the URIs; and their *(c)* usability, in particular for querying knowledge bases that mix NLP-generated knowledge with other sources. Figure 1 shows an example for a subject URI that was generated for a single *Person* instance in a document. Conforming to LOD best practices, URIs are HTTP-resolvable. A user-definable run-time parameter specifies the *base URI* (e.g., for a public SPARQL endpoint), which should always be a domain 'owned' by the user. This is followed by a *unique run id*, which ensures that each new run of LODeXporter on the same text generates new URIs. Generally, exposing implementation details in URIs is discouraged (Wood et al., 2014). However, re-running NLP pipelines (and therefore re-generating triples) is an extremely common occurrence in language engineering – for example, after improving a component, a machine learning model, or experimenting with different parameters. Hence, we decided that the triples from different runs of an NLP pipeline must be able to peacefully co-exist in a KB and be easily distinguishable, which is achieved with the generation of this *run id*. Next, the semantic (annotation) *type*, as detected by the NLP pipeline, is encoded (here "Person"), followed by its unique *annotation id* within a document. Finally, the mapping *export rule* name is added, as it is possible to export the same annotation (with the same type) multiple times, using different export rules (e.g., with different vocabularies).

3.3. Mapping Language

The central idea of our approach to automatic KB construction from NLP results is to *externalize* the knowledge about the export process, so that the NLP pipeline can remain unchanged (except for adding the LODeXporter component). This provides the required separation of concerns **(R2)** and makes it possible to easily reuse existing NLP pipelines for different knowledge bases. The export rules – how NLP results are mapped to triples – are themselves described in form of RDF triples using our new *mapping language*. Thus, the configuration how a specific knowledge base has to be populated with triples can be read from the same knowledge base, which provides for an introspective NLP export **(R3)**.

In detail, our mapping language provides constructs for mapping entities (subjects), features (properties and objects), as well as meta-information about the export (e.g., text offsets of entities or the name of the pipeline that generated an entity). A concrete mapping configuration is a set of rules (see Figure 2c for an example). These rules are read by LODeXporter, typically at the end of a pipeline, and triples

are then generated according to these rules. Hence, the exact same pipeline can be used to generate different LOD triples, e.g., when facts need to be expressed with different LOD vocabularies for different applications.

Mapping Entities. For each entity that needs to be exported, a corresponding mapping rule specifies the NLP type and its output RDF type. For example, a Person entity detected in a text can be mapped to a KB using the FOAF vocabulary.[4] To export all instances of a GATE annotation of type Person, detected in a document, to foaf:Person in a KB, this rule (in RDF/XML):[5]

```
<rdf:Description rdf:about="GATEPersonMapping">
  <rdf:type rdf:resource="map:Mapping"/>
  <map:baseURI rdf:resource=
      "http://semanticsoftware.info/lodexporter/"/>
  <map:type rdf:resource="foaf:Person"/>
  <map:GATEtype>Person</map:GATEtype>
</rdf:Description>
```

results in the generation of a triple with (1) a subject URI, as shown in Figure 1; (2) a property defining the rdf:type of that triple, and (3) the object URI with the defined type:

```
<SubjectURI>
    <http://www.w3.org/1999/02/22−rdf−syntax−ns#type>
    <http://xmlns.com/foaf/0.1/Person>
```

This mapping approach makes our solution extremely flexible: For example, to change the generation of triples to use the Person Core Vocabulary,[6] instead of FOAF, only a change of the mapping rule is required, which is read at export-time. Thus, the designer of a KB is free to experiment with multiple different knowledge representation approaches, without requiring any reconfiguration on the NLP side.

Mapping Features. Most entities are further described in an NLP process with additional features, e.g., the detected *gender* for a name. We can map any existing feature of an entity (one subject URI) to different properties and objects, with the same configuration approach. For example, to map a feature *gender* for a Person, we would define an additional mapping rule to transform it into foaf:gender:

```
<rdf:Description rdf:about="GATEFeatureMapping">
  <rdf:type rdf:resource="map:Mapping"/>
  <map:type rdf:resource="foaf:gender"/>
  <map:GATEfeature>gender</map:GATEfeature>
</rdf:Description>
```

This rule is then included in the first rule above with an additional triple:

```
<rdf:Description rdf:about="GATEPersonMapping">
  ...
  <map:hasMapping rdf:resource="GATEFeatureMapping"/>
</rdf:Description>
```

resulting in another output triple with the same subject URI:

```
<SubjectURI> <http://xmlns.com/foaf/0.1/gender> "male"
```

Additionally, we provide for the export of several meta-information of an annotation, such as the start- and end-offsets of the entity in a document, as well as the underlying string representation (surface form).

[4] Friend-of-a-Friend (FOAF), http://xmlns.com/foaf/spec/

[5] The prefix `map:` refers to our mapping language.

[6] Person Core Vocabulary, https://www.w3.org/ns/person

Mapping Relations. Additionally, we can export some pre-defined relations between entities into triples. In particular, we can map a *contains* relationship (based on text offsets) to any LOD vocabulary. An example is shown in Figure 2c, where we define that any *RhetoricalEntity* containing a *DBpediaNE* results in an additional triple with the two subject triples, linked by pubo:containsNE (see Figure 2b).

Finally, we provide for exporting a number of meta-information of an export process as relations, including: the name of the NLP pipeline that generated the annotations, creation timestamp, document name, corpus name, among others. These are important when the traceability of triples in a KB to their provenance is a concern.

4. Implementation

LODeXporter has been implemented in Java as an NLP component for the GATE *(General Architecture for Text Engineering)* framework (Cunningham et al., 2013). GATE is a mature, open source, component-based framework, which integrates with a number of well-known NLP libraries, such as Stanford CoreNLP, LingPipe, OpenNLP, and UIMA. This ensures LODeXporter can be used with a large variety of existing NLP applications.

The RDF(S) triples generated by LODeXporter follow the W3C standards and are not tied to any specific implementation. Internally, LODeXporter relies on the Apache Jena[7] libraries for generating the RDF graph. Results can be exported in two different modes **(R1)**.

Direct KB Export Mode: The first option is to directly connect the NLP framework with a triplestore. Running the NLP pipeline with LODeXporter will then directly populate the KB with the triples generated from a text. Here, we currently support the Apache TDB storage layer.[8]

File-Based Export Mode: The second option is to export the triples of a document into a file, based on the W3C standard N-Quads format.[9] These triples can then be imported into any triplestore. A typical application scenario in a distributed (e.g., cluster or cloud-based) environment is to upload the generated triples to a triplestore via the W3C SPARQL 1.1 Graph Store HTTP Protocol.[10] The files generated by LODeXporter can be used directly with an HTTP POST operation; we tested this extensively with the Apache Jena Fuseki SPARQL server[11] and its included *s-post* script.

The choice between the two export formats depends on the intended application scenario: In cases where a knowledge base is created in batch-mode, based on an existing corpus, the first solution avoids web protocol overhead, but currently

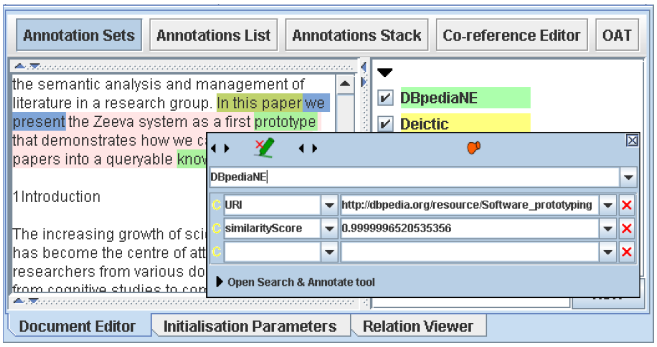
[7] Apache Jena, https://jena.apache.org/

[8] Apache TDB, https://jena.apache.org/documentation/tdb/index.html

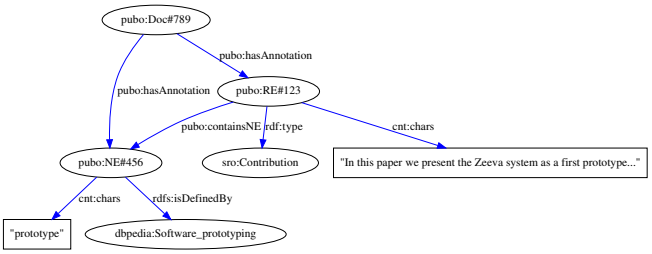
[9] W3C N-Quads, http://www.w3.org/TR/n-quads/

[10] SPARQL 1.1 Graph Store HTTP Protocol, http://www.w3.org/TR/sparql11-http-rdf-update/

[11] Apache Jena Fuseki, https://jena.apache.org/documentation/fuseki2/



(a) NLP annotations in GATE Developer



(b) LODeXporter output

```
@prefix map: <http://lod.semanticsoftware.info/mapping/mapping#> .
@prefix pubo: <http://lod.semanticsoftware.info/pubo/pubo#> .
@prefix rdf: <http://www.w3.org/1999/02/22−rdf−syntax−ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf−schema#> .
@prefix cnt: <http://www.w3.org/2011/content#> .
@prefix sro: <http://salt.semanticauthoring.org/ontologies/sro#> .
@prefix ex: <http://example.com/> .

### Annotation Mapping ###
ex:GATERhetoricalEntity a map:Mapping ;
        map:type        sro:RhetoricalElement ;
        map:GATEtype    "RhetoricalEntity" ;
        map:hasMapping  ex:GATEContentMapping .

ex:GATEDBpediaNE a map:Mapping ;
        map:type        pubo:LinkedNamedEntity ;
        map:GATEtype    "DBpediaNE" ;
        map:hasMapping  ex:GATEContentMapping ;
        map:hasMapping  ex:GATELODRefFeatureMapping .

### Feature Mapping ###
ex:GATEContentMapping a map:Mapping ;
        map:type        cnt:chars ;
        map:GATEattribute "content" .
ex:LODRefFeatureMapping a map:Mapping ;
        map:type        rdfs:isDefinedBy ;
        map:GATEfeature "URI" .

### Relation Mapping ###
ex:RE_NE_RelationMapping a map:Mapping ;
        map:type        pubo:containsNE ;
        map:domain      ex:GATERhetoricalEntity ;
        map:range       ex:GATEDBpediaNE ;
        map:GATEattribute "contains"
```

(c) LODeXporter mapping rules example

Figure 2: Example RDF graph (b) generated by LODeXporter from NLP annotations (a) based on the mapping rules shown on the right (c) for a paper in a Semantic Publishing application

only works with TDB-based triple stores and is limited to parallelization within a single JVM (using the "GATE Cloud Parallelizer"[12] tool). The second approach is suitable for continuously updating a knowledge base, e.g., based on a stream of incoming documents, and can be easily distributed in a cloud infrastructure. It is also not limited to a specific knowledge base implementation, but requires an accessible SPARQL over HTTP interface.

5. Application

Figure 2 shows an example from a real-world application scenario in *Semantic Publishing* (Berners-Lee and Hendler, 2001), which aims at enriching scientific publications with machine-readable information in order to explicitly mark up experiments, data, and rhetorical elements in their raw text. While a number of RDFS and OWL ontologies are now available, like SALT (Groza et al., 2007) and the Semantic Publishing And Referencing (SPAR) ontology family (Peroni et al., 2012), a serious bottleneck to their adoption is the vast amount of existing publications, which would be too time-consuming to manually annotate.

Hence, NLP techniques play a crucial role in enabling Semantic Publishing workflows and applications. In (Sateli and Witte, 2015), we show how scientific literature can be transformed into a queryable knowledge base, through an NLP pipeline that detects rhetorical entities and domain concepts in their full-text (Figure 2). The input to our pipeline was a set of 136 open access computer science articles. We processed the full-text of each article to extract various rhetorical entities (e.g., claim or contribution sentences) and further linked each domain concept in the document to its corresponding resource on the LOD cloud. We used the LODeXporter component in our text mining pipeline to transform the detected entities (added to the document in form of GATE annotations) to RDF triples, based on the mapping rules shown in Figure 2c. The complete knowledge base contains 1,086,051 triples.[13]

A few example triples from the populated knowledge base are presented in Figure 3a, describing a contribution sentence found in a document, as well as a concept (named entity) mentioned in its text. Here, we can see the triple generated for the document and the PUBO vocabularies used to attach the triples describing the rhetorical and named entities to the document. Based on this schema, the knowledge base can then be queried, e.g., to find all documents with their contribution sentences and mentioned topics by using a SPARQL query, like the one shown in Figure 3b. The query shown here retrieves all documents that have an annotation of type `sro:Contribution` (sentence), as well as the named entities that are contained within the sentence boundary, returning *(a)* their surface form (as they appeared in the document), and *(*b) their corresponding resource from the DBpedia ontology. An example entry from the result set is illustrated in Figure 3c. Here, the sentence contains the term "ASR", which was grounded to the entry for 'Automatic Speech Recognition' in the DBpedia ontology, demonstrat-

[12] The GATE Cloud Paralleliser (GCP), https://gate.ac.uk/gcp/

[13] Available in N-Quads format at http://www.semanticsoftware.info/semantic-scientific-literature-peerj-2015-supplements

```
<http://example.com/Corpora/PeerJ-CompSci/cs-8.xml>
 <http://lod.semanticsoftware.info/pubo/pubo#hasAnnotation>
 <http://semanticsoftware.info/lodexporter/9182bfec-a88f-4b61-8bde-0bd6a670449b/RhetoricalEntity/97528#map:GATERhetoricalEntity>,
 <http://semanticsoftware.info/lodexporter/9182bfec-a88f-4b61-8bde-0bd6a670449b/DBpediaNE/114836#map:GATEDBpediaNE> .

<http://semanticsoftware.info/lodexporter/9182bfec-a88f-4b61-8bde-0bd6a670449b/RhetoricalEntity/97528#map:GATERhetoricalEntity>
 a <http://salt.semanticauthoring.org/ontologies/sro#RhetoricalElement> ,
   <http://salt.semanticauthoring.org/ontologies/sro#Contribution> ;
 <http://lod.semanticsoftware.info/pubo/pubo#containsNE>
   <http://semanticsoftware.info/lodexporter/9182bfec-a88f-4b61-8bde-0bd6a670449b/DBpediaNE/114836#map:GATEDBpediaNE> ;
 <http://www.w3.org/2011/content#chars>
  "We evaluated the system performance using this ASR implementation."^^<http://www.w3.org/2001/XMLSchema#string> .

<http://semanticsoftware.info/lodexporter/9182bfec-a88f-4b61-8bde-0bd6a670449b/DBpediaNE/114836#map:GATEDBpediaNE>
 a <http://lod.semanticsoftware.info/pubo/pubo#LinkedNamedEntity> ;
 <http://www.w3.org/2000/01/rdf-schema#isDefinedBy> <http://dbpedia.org/resource/Speech_recognition> ;
 <http://www.w3.org/2011/content#chars>"ASR"^^<http://www.w3.org/2001/XMLSchema#string> .
```

(a) Examples triples in N-Quads format from the knowledge base

```
PREFIX pubo: <http://lod.semanticsoftware.info/pubo/pubo#>
PREFIX sro: <http://salt.semanticauthoring.org/ontologies/sro#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cnt: <http://www.w3.org/2011/content#>

SELECT DISTINCT ?document ?sentence ?topic ?uri WHERE {
      ?document pubo:hasAnnotation ?rhetoricalEntity .
      ? rhetoricalEntity  cnt:chars  ?sentence .
      ? rhetoricalEntity  rdf:type  sro:Contribution  .
      ? rhetoricalEntity  pubo:containsNE ?namedEntity.
      ?namedEntity rdfs:isDefinedBy ?uri .
      ?namedEntity cnt:chars ?topic .
} ORDER BY ?paper
```

(b) A SPARQL query to find contribution sentences and their domain concepts

Document	Sentence	Topic	URI
cs-8.xml	"We evaluated the system performance using this ASR implementation."	"ASR"	dbpedia:Speech_recognition

(c) One example result from the knowledge base

Figure 3: Example triples (a) generated by LODeXporter in a semantic publishing pipeline, the corresponding SPARQL query (b) on the knowledge base, and the resulting table (c) on the bottom.

ing the power of connecting the NLP results in a knowledge base with external open datasets.

6. Related Work

We previously presented *OwlExporter* (Witte et al., 2010) at LREC 2010, an NLP component for ontology population from text. In contrast with LODeXporter, OwlExporter is focused on the population of Web Ontology Language (OWL) ontologies,[14] with Description Logic (DL) reasoning as their primary application. In some respect, LODeXporter is our completely re-designed approach for knowledge export from NLP pipelines, based on the experience we gained from building numerous knowledge-intensive applications. In particular, with LODeXporter we focus on scenarios that were less common when we first designed OwlExporter, more than 15 years ago: Large-scale knowledge bases, generated in distributed cloud-based environments, linked data datasets and applications (based on the LOD initiative started in 2007), shared open vocabularies, as well as the proliferation of AI applications, such as intelligent assistants, where NLP techniques form just one of numerous knowledge sources and algorithms. While OwlExporter relied on special NLP annotations driving the ontology population, with LODeXporter we completely externalized the mapping rules, so that the same NLP pipeline can now generate different triples, based on different vocabularies, when connected to various knowledge bases.

NLP2RDF[15] is another approach for converting NLP tool output to RDF (Hellmann et al., 2013). However, the goals of NLP2RDF are largely orthogonal to the ones from LODeXporter: NLP2RDF focuses on the representation of *NLP data* (such as, words, sentences, strings), with the primary goal of providing an exchange format between different NLP tools. All knowledge is represented through its own NIF (NLP Interchange Format) vocabulary.[16] In contrast, LODeXporter focuses on the representation of *domain data* (e.g., biological entities, company intelligence, financial data), which is represented in domain-specific vocabularies. The resulting knowledge bases are primarily meant for creating LOD datasets for publication and/or building intelligent applications on top of them. Hence, although LODeXporter and NLP2RDF both generate RDF from NLP frameworks, they are two quite distinct solutions for different application use cases.

[14] Web Ontology Language (OWL), https://www.w3.org/OWL/

[15] NLP2RDF, see https://site.nlp2rdf.org/

[16] NIF 2.0 Core Ontology, see http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core

7. Conclusion

We presented LODeXporter, a novel approach for the generation of linked open data (LOD) compliant triples from text analysis pipelines. LODeXporter provides for a separation of concerns, as the NLP pipeline (developed by a language engineer) is strictly separated from the export of the NLP results into RDF(S) triples (designed by a knowledge engineer). We achieve this separation through a new mapping language, which is defined itself in RDFS, and defines the rules for converting NLP annotations and features into *(subject, property, object)* triples. This approach supports agile data science workflows (Sateli and Witte, 2016), where existing NLP pipelines can be easily connected with different web vocabularies, for example, to facilitate submissions to shared tasks within competitions.

LODeXporter enables NLP framework users to easily generate knowledge bases in an LOD-compliant format, which can then be shared on the linked open data (LOD) cloud. It also facilitates the development of complex AI applications, such as intelligent assistants, which typically rely on numerous knowledge sources, including text analysis results.

8. Acknowledgements

This work was partially funded by a Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant (DG).

9. Bibliographical References

Berners-Lee, T. and Hendler, J. (2001). Publishing on the semantic web. *Nature*, 410(6832):1023–1024.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). DBpedia—A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165.

Brickley, D. and Guha, R. (2014). RDF Schema 1.1. W3C Recommendation, https://www.w3.org/TR/rdf-schema/.

Cunningham, H., Tablan, V., Roberts, A., and Bontcheva, K. (2013). Getting more out of biomedical documents with GATE's full lifecycle open source text analytics. *PLoS computational biology*, 9(2):e1002854.

Cyganiak, R., Wood, D., and Lanthaler, M. (2014). RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation, https://www.w3.org/TR/rdf11-concepts/.

Groza, T., Handschuh, S., Möller, K., and Decker, S. (2007). SALT – Semantically Annotated LaTeX for Scientific Publications. In *The Semantic Web: Research and Applications*, LNCS, pages 518–532. Springer.

Heath, T. and Bizer, C. (2011). *Linked Data: Evolving the Web into a Global Data Space*. Synthesis lectures on the semantic web: theory and technology. Morgan & Claypool Publishers.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP Using Linked Data. In *Proceedings of the 12th International Semantic Web Conference - Part II*, ISWC '13, pages 98–113. Springer-Verlag New York, Inc.

Peroni, S., Shotton, D., and Vitali, F. (2012). Scholarly publishing and linked data: describing roles, statuses, temporal and contextual extents. In *Proceedings of the 8th International Conference on Semantic Systems*, pages 9–16. ACM.

Sateli, B. and Witte, R. (2015). Semantic representation of scientific literature: bringing claims, contributions and named entities onto the Linked Open Data cloud. *PeerJ Computer Science*, 1(e37), 12/2015.

Sateli, B. and Witte, R. (2016). From Papers to Triples: An Open Source Workflow for Semantic Publishing Experiments. In *Semantics, Analytics, Visualisation: Enhancing Scholarly Data (SAVE-SD 2016)*, Montréal, QC, Canada, 04/2016. Springer.

Witte, R., Khamis, N., and Rilling, J. (2010). Flexible Ontology Population from Text: The OwlExporter. In *International Conference on Language Resources and Evaluation (LREC)*, pages 3845–3850, Valletta, Malta, May 19–21. ELRA.

Wood, D., Zaidman, M., Ruth, L., and Hausenblas, M. (2014). *Linked Data: Structured data on the Web*. Manning Publications Co.

LiDo RDF: From a Relational Database to a Linked Data Graph of Linguistic Terms and Bibliographic Data

Bettina Klimek[1,2,a], Robert Schädlich[1,2,a], Dustin Kröger[2,b], Edwin Knese[2,b], Benedikt Elßmann[2,b]
[1]AKSW/KILT Research Group, InfAI and University of Leipzig, Germany
[2]Faculty of Mathematics and Computer Science, University of Leipzig, Germany
[a]{klimek, schaedlich}@informatik.uni-leipzig.de, [b]{dk33cuku, ek41cyvi, be57xocu}@studserv.uni-leipzig.de

Abstract

Forty years ago the linguist Dr. Christian Lehmann developed a framework for documenting linguistic terms, concepts and bibliographic data that resulted in the LiDo Terminological and Bibliographical Database (LiDo TBD). Since 2006 students and linguistic researchers benefit from the data by looking it up on the Web. Even though, the LiDo TBD is implemented as a relational database, its underlying framework aims at yielding a terminological network containing data nodes that are connected via specific relation edges in order to create an interrelated data graph. Now, with the emergence of Semantic Web technologies we were able to implement this pioneering work by converting the LiDo TBD relational database into a Linked Data graph. In this paper we present and describe the creation of the LiDo RDF dataset and introduce the LiDo RDF project. The goals of this project are to enable the direct use and reuse of the data both for the scientific research community and machine processing alike as well as to enable a valuable enrichment of already existing linguistic terminological and bibliographic data by including LiDo RDF in the LLOD cloud.

Keywords: Lido Terminological and Bibliographical Database, linguistic concepts, linguistic terminology, linguistic bibliography, Linguistic Linked Open Data (LLOD)

1. Introduction

The clarification of terminological problems is a prerequisite for methodologically and scientifically sound linguistic research (Lehmann, 1996). Resources providing and defining linguistic terminology are, therefore, of inestimable value because they practically assist the researcher in finding, understanding and reusing linguistic terms in the context of his work. Such a resource constitutes the *LiDo Terminological and Bibliographical Database*[1] (Lehmann, 1996; Lehmann, 1976). Having started in year 1976, Christian Lehmann[2] contributed his knowledge as a general and comparative linguist by manually collecting more than 4500 unique linguistic concepts, 15000 terms, 20000 books and 1200 journals. He structured this still ongoing data compilation effort within a relational database that interrelates concept, term and bibliographical data which resulted in the LiDo website that is available since 2006. While this browsable Web interface is usable as a look-up resource, it provides no means that enable a citation of the terms and concepts by Christian Lehmann because the single entries are not rendered with specified URLs which are usually required for citing Web resources.

Subsequently, the LiDo RDF project emerged in order to enable the direct citation of the terminological data for researchers but also to empower machine processing. This is achieved by converting the relational database that is the source of the LiDo website (cf. footnote 1) into an RDF dataset[3]. The choice of the RDF format is motivated by the advantages of Linked Data in general, i.e. the interoperability and integration of the LiDo data into the Linguistic Linked Open Data (LLOD) Cloud[4] which allows a direct reuse and interconnection to other (terminological and/or bibliographical) linguistic data resources.

The remainder of this paper is structured as follows. An overview of the scope and aims of the LiDo RDF project is given in Section 2. A delimitation of the LiDo RDF dataset in comparison to the existing related work is outlined in Section 3. Section 4. explains how the LiDo RDF dataset has been created. This includes the description of the source data (Section 4.1.), the presentation of the created ontology (Section 4.2.) and also an illustration of the used SQL to RDF mapping tool (Section 4.3.). This is followed by an overview of the resulted LiDo RDF dataset in Section 5. and an investigation of the quality of this data in Section 6. The paper concludes in Section 7. by giving a prospect of the future work.

2. The LiDo RDF Project

The LiDo RDF project evolved out of the initial aim to provide referenceable term and concept resources for the *LiDo Terminological and Bibliographical Database* (in short *LiDo TBD*). Therefore, the data that is available on the LiDo TBD website has been converted into the LiDo RDF dataset following the creation procedure as described in Section 4. While the resulting dataset (cf. Section 5.) constitutes the main effort of the LiDo RDF project, the following goals are pursued in addition to the mere dataset conversion task and are to a large extent achievable due to the underlying Linked Data format of LiDo RDF:

1. **Data evolution**: Because the LiDo TBD is still edited and updated by Christian Lehmann, the LiDo RDF dataset strives to evolve accordingly. Therefore, a new version of LiDo RDF will be generated from LiDo

[1] http://linguistik.uni-regensburg.de:8080/lido/Lido

[2] https://www.christianlehmann.eu/

[3] The authors thank Christian Lehmann for enabling this dataset creation by providing them with his database and allowing that the LiDo RDF dataset can be published and reused under an open license.

[4] http://linguistic-lod.org/llod-cloud

TBD twice a year. The previous versions will be kept available for download.

2. **Data reuse for human users:** RDF datasets are generally not easily to be read by humans. In order to keep the lookup of the data as user friendly as possible, a browsable search interface similar to the LiDo TBD has been created based on LiDo RDF. It differs, however, in that every entry can be now cited with a unique URL.

3. **Data exploration:** For users who are more familiar with the RDF data format, a navigation through the data graph via the resource links will be available in a Linked Open Data view of LiDo RDF. Moreover, further insights into the data can be obtained by querying the data using the provided SPARQL endpoint.

4. **External data enrichment:** The LiDo RDF dataset is one among other datasets for linguistic terminology and bibliographic resources. Therefore, it is desirable to interrelate the LiDo RDF data with other similar resources. This can be achieved by integrating LiDo RDF into the LLOD Cloud and interlinking it with already existing terminological or bibliographical resources.

All in all, the LiDo RDF project aims at preserving the original LiDo TBD data and providing the means to enable the reuse of the data for humans and for machine processing alike. In contrast to the LiDo TBD which is only indirectly available as a dataset to search and view via a web interface, LiDo RDF additionally provides the actual data that can be cited in ongoing research but also reused, shared and interlinked by the linguistic research community. Furthermore, it can be directly integrated into applications that need to consume or process the data. The dataset versions and all the features described will be available from `http://lidordf.aksw.org/`. More technical details are contained in this Github repository: `https://github.com/AKSW/lido2rdf`.

3. Related Work

With regard to the models that are present for representing terminological data in linguistics as a Linked Data graph, the OntoLex-Lemon model[5] (McCrae et al., 2017) and the Ontology for Linguistic Terminology (OnLiT) (Klimek et al., 2017) have to be mentioned. OntoLex-Lemon is specified for representing lexical language data. This poses several difficulties which led to the conclusion that this vocabulary is not suitable as a modelling basis for the LiDo TBD into RDF. Most importantly what is defined as a *term* in LiDo TBD does not apply to the notion of *lexical entry* in OntoLex-Lemon. Also the concepts included in LiDo TBD cannot be understood as *lexical sense*, which would be the corresponding equivalent in OntoLex-Lemon. What is more, LiDo TBD contains a set of relations for which appropriate object and datatype properties do not exist in the OntoLex-Lemon vocabulary. It turned out that the modelling of the terminological data in LiDo TBD requires a more specific vocabulary. Such a vocabulary is OnLiT, that has been created for the purpose to model LiDo TBD in RDF. How it has been eventually used is explained in detail in Section 4.2.

Contentwise the LiDo RDF dataset is concerned with three different kinds of data: 1) linguistic concepts, 2) linguistic terms and 3) linguistic bibliography. Various Linked Data datasets exist that contain data of one or two of these domains. In the following, exemplary datasets are mentioned.

The General Ontology for Linguistic Description (GOLD)[6] (Farrar, 2010) provides a taxonomy of nearly 600 linguistic concepts for descriptive linguistics with corresponding term data encoded only as strings in RDF labels and not as independent resources.

The Bibliography of Linguistic Literature Thesaurus (BLL Thesaurus) (Chiarcos et al., 2016) includes around 7400 linguistic terms. However, only a part of the term data is semantically defined via links to corresponding OLiA[7] concepts. In this case, a comprehensive interlinking to LiDo RDF could provide appropriate concepts and their definitions for other yet unspecified term resources in the BLL Thesaurus.

The Grammis online resource[8] contains both linguistic concepts and terms specifically for German grammar (Suchowolec et al., 2017).

In the context of the Semantic Web and the existing LLOD Cloud, efforts emerged that aim at interlinking the aforementioned and other existing datasets in order to provide more comprehensive terminological and bibliographic data that is relevant for the linguistic sciences. Such an endeavour, for instance, is the BLL – Linguistic Linked Open Data Edition[9] (Chiarcos et al., 2016) that interlinks the BLL Thesaurus with the Bibliography of Linguistic Literature[10].

However, the framework that Christian Lehmann has built for the LiDo TBD pursued the same goal already forty years ago. In this respect, to the best of our knowledge, LiDo RDF is unique because it provides term, concept and bibliographic data —that is semantically more specifically and consistently interrelated than it is in existing datasets — joined into one single and multilingual dataset. Therefore, we believe, the ongoing task of interlinking existing resources about linguistic terminology would highly benefit from considering or at least discussing the reuse of the LiDo ontology (i.e. mainly OnLiT) in order to arrive at a more coherent and semantically richer Linked Data graph as a terminological basis for linguistic research.

4. Dataset Creation

4.1. LiDo TBD Source Data

The LiDo TBD is only available for lookup purposes as it is present on the website `http://linguistik.`

[5] `https://www.w3.org/community/ontolex/wiki/Final_Model_Specification`

[6] `http://linguistics-ontology.org/gold-2010.owl`

[7] `http://acoli.cs.uni-frankfurt.de/resources/olia`

[8] `http://www.ids-mannheim.de/grammis/`

[9] `http://data.linguistik.de/bll/index.html`

[10] `http://www.blldb-online.de`

uni-regensburg.de:8080/lido/Lido. In order to create LiDo RDF we used the Microsoft Access database that is the editing basis of Christian Lehmann. From this we generated a PostgreSQL database that served as the source data for the data conversion[11]. The PostgreSQL database is very similar to the Microsoft Access database, however, PostgreSQL was the required format for the mapping tool that was used for converting the tabular data to RDF data. The source data mainly consists of tables that contain entities and attributes describing them, e.g. a "term" table or a "book" table[12]. Additionally, the source data contains tables that interrelate the entities of other tables, e.g. concept entities with book entities or concept entities with term entities. The large part of the data centers around concepts, terms and bibliographic elements such as books and journals. For concept entities attributes like "delimitation", "analytic procedure" and "phenomenology" have been entered in addition to rather common attributes such as "definition" and "example". The attributes that describe the term entities are "abbreviation", "etymology" and "language". The latter is provided for every term, predominantly in English, German, Spanish and Portuguese. With regard to the bibliographic entities typical attributes like "author", "title", ect. exist but also the "text sort" is stated which facilitates the identification of dictionaries, questionnaires and more.
Furthermore, 18 dedicated relations are established that interrelate term entities, concept entities and concept entities with term entities. Out of these, 14 constitute coordinating and subordinating relations which interconnect concept entities. Very general hierarchical or meronymic relations like "is a (kind of)" or "is part of", but also more specific relations such as "is result of", "manifests" or "is operator of" contribute to the creation of a relational network which is already quite close to the semantic RDF graph structure. In summary, the data basis consisted of 11 tables containing entities with attributes and 7 tables that interrelated these entities by cross reference. Metadata such as the date of the last edit for an entry or the logbook table were not part of the RDF conversion.

4.2. LiDo Ontology

Every Linked Data dataset needs to be formally described by a specified model, i.e. its underlying vocabulary or ontology. In compliance to the best practices and the given Semantic Web standards, we reused existing vocabularies and extended them where necessary in order to create the LiDo ontology[13] which is the basis for the LiDo RDF dataset. While the modelling decisions for the LiDo ontology will be explained in what follows, it is recommended to refer to Figure 1 which indirectly also exemplifies the usage of the vocabulary.
The LiDo TBD adheres to the statement that "proper terminology is concerned with the relationship between concepts, and between them and their designations, rather than with designations alone or with the objects they represent"[14] and, thus, concomitantly distinguishes and interrelates linguistic concept and term resources. While the former are defined as language-independent mental objects (i.e. units of meaning) the latter are defined as language-specific linguistic objects. Consequently, within the LiDo TBD source data **a linguistic concept is unique and associated with a linguistic term that in turn hypostatizes the concept**. Within the LiDo TBD source data this has been realized by identifiers and unique labels, which are in many cases Latin expressions chosen by Christian Lehmann, to represent the concept entities within the tabular data. Similarly, the term entries are also represented by identifiers which are interrelated with their language specific expressions. All semantic interrelations occur between concept entities which are associated with the respective term entities. All of this has been already modelled within the Ontology for Linguistic Terminology (OnLiT)[15]. OnLiT was created in previous work mainly for the purpose of creating LiDo RDF and emerged from the same LiDo TBD data source (Klimek et al., 2017)[16]. Since this vocabulary contains the modelling of concept and term resources as well as their established interrelations according to the source data, it is included and reused as an OWL import within the LiDo ontology.
The other part of the data containing the bibliographic data could be modelled by reusing the Bibliographic Ontology (D'Arcus and Giasson, 2011)[17], because it contained the required classes and object properties for representing the tabular source data entities, e.g. "book", "journal", "author" and "publication year".
While the majority of the LiDo TBD could be represented by importing OnLiT and using a part of the Bibo ontology, a set of tables containing data such as languages, areas or text sorts which were interrelated with the "term" or "book" tables remained. In order to cover this data as well, we decided to create new classes and object properties to model this relational data within the LiDo ontology. The object property lido:hasBibRef, for instance, needed to be introduced to account for the interrelation of concept resources with bibliographic resources. Moreover, with regard to the Lido TBD tables "languages" and "areas", it has to be mentioned that Linked Data resources already exist, e.g. datasets for geographical and language data. However, we did not reuse these because it requires a large amount of manual mapping effort to retrieve the ca. 1100 language entities and ca. 160 area entities. Instead, the classes lido:Language, lido:Area and lido:Textsort, which were not included within the Bibo vocabulary, have been newly created and populated with the respective individuals from the tables. While the more accurate mapping task remains open as future work the source data is at its current state modelled exhaustively with the LiDo ontol-

[11] Neither this PostgreSQL database nor the underlying MS Access file are or will be publicly available. The authors have been, however, generously granted access for the undertaking of the LiDo RDF dataset creation.

[12] The source data is comprehensively described in (Lehmann, 1996), which is recommended for further reading.

[13] http://lidordf.aksw.org/ontology/

[14] http://www.computing.surrey.ac.uk/ai/pointer/report/section1.html

[15] http://lido.linguistic-lod.org/onlit.rdf

[16] Please consult this reference for more details and examples of the concept and term representation in the source data.

[17] http://purl.org/ontology/bibo/

ogy.

4.3. SQL to RDF Mapping

The actual data of the LiDo RDF dataset was created automatically by using the presented LiDo ontology and the Sparqlify tool[18] (Stadler et al., 2015) which enables a reliable and scalable transformation of relational data to RDF. Since the PostgreSQL source data and the desired RDF format differ fundamentally from each other, a manual mapping between both formats is required. Due to the supported Sparqlification Mapping Language (SML) that is integrated in Sparqlify we could easily express the necessary PostgreSQL to RDF mappings with SML[19]. As a result, the Sparqlify tool provides as output file the converted RDF data in N-Triples, a plain text serialization format for RDF graphs. In order to implement several post-processing possibilities, e.g. the set up of a Linked Open Data navigation view or the addition of links to other datasets, the output file has been loaded into a triplestore. Thereby we could also provide the SPARQL endpoint[20] for the Lido RDF dataset which enables a direct querying of the data for more detailed insights. Additionally, SPARQL queries are used to display LiDo RDF data dynamically while navigating through the human-readable search interface[21] of the LiDo RDF data. Because the search interface is based on the results of the queries for the current RDF concept and term resources, their URIs can be now used to cite the original LiDo TBD in scientific works.

5. LiDo RDF Dataset

5.1. LiDo as Linked Data Graph

By using the LiDo ontology all the data that is browsable within the LiDo TBD website could be converted into the LiDo RDF dataset. The namespace for the data is `http://lidordf.aksw.org/resource/` and its prefix is `lido`. Figure 1 illustrates a part of the data graph focusing especially on the semantically rich interrelation of linguistic concepts. The examples of the `lido:Book_13757`, `onlit:Concept_509` and `onlit:Term_511` instances show that the identifiers of the LiDo TBD source data have been reused to populate the `bibo:Book`, `onlit:Concept`, `onlit:Term` classes. The given designations for the concept identifiers (e.g. "denominatio") and the language-specific expressions corresponding to the term identifiers (e.g. "denomination" in English) in the tabular source data have been modelled by using the `rdfs:label` object property. The labels of all `onlit:Concept` instances in the LiDo dataset correspond to the list of entries that can be found under "Unique Designation" on the LiDo TBD website and are mostly suitable Latin expressions chosen by Christian Lehmann which makes it easier to refer to the concepts (in place of using the identifiers). The labels of `onlit:Term` instances, since encoding the concepts in different languages, are provided with a language tag.

In the following an overview will be given about the data that is created in LiDo RDF for the three main data types and how they are interconnected within the graph:

Bibliographic data: The main class within the LiDo ontology is the reused `bibo:Document` class, that contains the book and journal entries from the LiDo TBD source data. All book and journal resources are further specified for typical information about bibliographic entities, such as author, title, publisher, publication date but also for their text sort (not shown in Figure 1). What is remarkable, however, is that bibliographic works that can be consulted for more information about a certain linguistic concept are interlinked with respective concept resources, e.g. in Figure 1 the concept `lido:Concept_509` ('denominatio') is associated with the bibliographic reference `lido:Book_13757` ("Knobloch, Clemens and Schaeder, Burkhard (eds.) 1996, Nomination - fachsprachlich und gemeinsprachlich. Opladen: Westdeutscher Verlag (Sprachwissenschaft).").

Term data: Term instances can have additional etymological information or a given abbreviation (which are not shown in Figure 1). Furthermore, some terms are interrelated. I.e. it is explicitly stated whether a term is an abstract or concrete noun of another term, e.g. the German term *Semantik* is the abstract noun of the German adjective term *semantisch*. Regarding the interrelation of term and concept instances, LiDo RDF implemented the two possibilities of stating that a term is either the standard or non-standard expression of a concept according to the LiDo TBD as it is exemplified in Figure 1). To the whole dataset applies a one-to-one correspondence between term and concept resources. This meets the self-imposed requirement by Christian Lehmann for representing terminological data in order to ensure a disambiguated traceability of terms (and to which the LiDo TBD source data also applies). By that the LiDo RDF dataset represents a language-independent approach that enables the integration of multilingual terminological networks by defining a term in relation to the linguistic concept it encodes.

Concept data: Consequently, not the term but the concept instances are specified with a definition. For these also examples and information on the analytic procedure, delimitation and history as well as the phenomenology of a concept resource can be given, which is exemplified in Figure2. All this information is stated in plain text for every single concept resource. In addition to that, the meaningful interrelation of concepts constitutes an added value for defining a linguistic concept within the broad domain of linguistics. I.e. in LiDo RDF are no loose concepts. Every concept has at least one direct relation to another concept. The underlying basis for these interrelations builds an hierarchical structure between the concepts that is created due to the 18 subordinating and coordinating relations and emerged from the concept-concept relations that are contained in the LiDo TBD source data. Figure 1 shows some of these re-

[18] `https://github.com/AKSW/Sparqlify`

[19] The created mapping file can be found here: `https://github.com/AKSW/lido2rdf/blob/master/SPARQLIFY%20SQL2RDF%20mapping/mapping.sml`

[20] To be found here: `http://lidordf.aksw.org/sparql/`.

[21] To be accessed here: `http://lidordf.aksw.org/glossary/`.

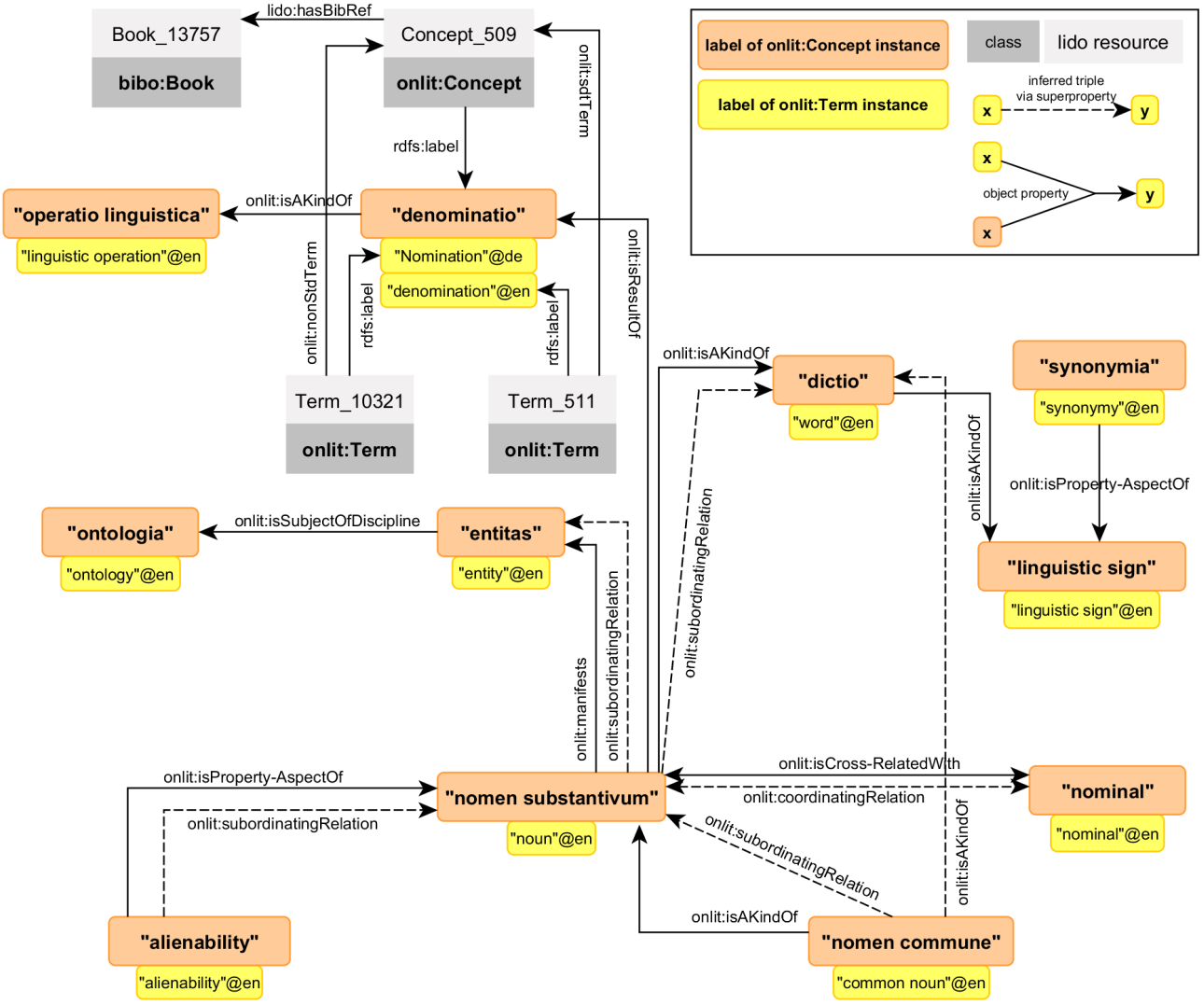


Figure 1: Example excerpt of the LiDo RDF data graph.

lations that hold between various concept instances[22] and have been modelled as OnLiT object properties. They are the cause for the emergence of the relational network of linguistic terms in the LiDo TBD source data and for the linked data graph structure of LiDo RDF respectively. To the best of our knowledge this manageable and semantically standardized set of relations is unique for a dataset of linguistic terminology because these more specific relations can be used to disambiguate the sub- and coordinating relations that are normally only formally defined and, hence, confound semantically different relations. As a result, the LiDo RDF data graph yields more informative insights into a linguistic concept, because the meaning of a linguistic concept is, next to a formulated definition, also defined by its semantic interrelation to other linguistic concepts.

Finally, with LiDo RDF an openly accessible version of the LiDo TBD dataset is available for further reuse. The underlying graph structure of the data does not only satisfy the requirements of representing the relational network of the LiDo TBD source data, but also enables the gain of deeper insights into the dataset by means of inference and reasoning. I.e. obtaining all concepts that are interlinked to one bibliographic resource, or extracting the whole hierarchy tree for a concept at once is now possible but also incomplete concept entries (e.g. with missing definitions) for future editing of the dataset can be easily retrieved. The generated LiDo RDF dataset is finally completed after it has passed a data quality check-up.

5.2. LiDo within the LLOD Cloud

While the LiDo TBD in its current state as a searchable Web interface is isolated in terms of data reusability, the availability of the LiDo RDF dataset entails the possibility to interrelate it with other existing datasets containing terminological and/or bibliographic data for the domain of linguistics. Such an interconnection of datasets including similar content is desirable because it provides the research community with an overview of various data sources that

[22] Note that for better readability the labels of the `onlit:Concept` instances and their corresponding English `onlit:Term` instances are displayed in place of their respective instances.

semantically refer and, consequently, also enrich each other by pointing to similar data entries that can be also consulted for further information. For linguistic data in the Semantic Web landscape such an environment provides the LLOD Cloud. In order to enrich the LiDo RDF dataset with external data sources and simultaneously provide the possibility for other dataset creators to point to linguistic concepts or terms within LiDo RDF, an integration of the data into the LLOD Cloud is considered a worthwhile undertaking.

Therefore, a manual set of 50 links adhering to the requirements of publishing datasets within the LLOD Cloud[23] has been created as a starting point to add LiDo RDF to the already existing collection of linguistic data within the Cloud. For the linking the Bibliography of Linguistic Literature (BLL) Thesaurus[24] (Chiarcos et al., 2016) has been chosen to be a suitable dataset because it also contains terminological and bibliographic data. In this case the linguistic concepts between LiDo RDF and the BLL Thesaurus have been mapped by using the object property `skos:exactMatch`. The linking for the linguistic concept 'circumfix', for instance, is as follows:

```
<http://lidordf.aksw.org/resource/
Concept_290>

skos:exactMatch

<http://data.linguistik.de/
bll/bll-thesaurus#bll-133083225>.
```

It has to be noted that linguistic concepts in LiDo RDF are represented as OWL individuals, while they are represented as OWL classes and are also of the type `skos:Concept` within the BLL Thesaurus. A valid statement is, however, created because the skos object property `skos:exactMatch` entails and, therefore, automatically creates a type assertion for the `lido:Concept_290` instance to be also of the `rdf:type skos:Concept`, which is necessary to yield a valid statement. The 50 created links have been added to the LiDo RDF dataset and further include, for example, the linguistic concepts 'dative', 'number', 'subject' and 'juxtaposition'.

The identification of matching linguistic concepts required a detailed study of both datasets. I.e. in order for a LiDo RDF and a BLL Thesaurus concept to be considered as an exact match, the following two requirements had to be fulfilled: 1) The English `skos:prefLabel` of the `bllt:bll-133083225` class and the label of the standard English term instance that corresponds to the `lido:Concept_290` instance had to be identical; 2) the textual definition of the LiDo concept instance and the textual comment defining the OLiA[25] class that corresponds to BLL Thesaurus class had to convey a close to equivalent content. Even though an entire linking by using also less strict mapping properties, such as `skos:broader` or `skos:narrrower` seems to be promising, the manual linking just explained showed that this task will be time-consuming and requires human judgment about the similarity of two concepts. An automated linking process due to the amount of data is, hence, favourable but should also implement some kind of quality assurance.

In conclusion, the contribution of LiDo RDF to the LLOD Cloud shall rise awareness of the dataset itself and will ideally result in collaborative work with the creators of similar existing or future datasets concerning the realization of a more extensive interlinking of terminological or bibliographic data that benefits the whole linguistic research community.

5.3. LiDo RDF Web Interface

As has been already mentioned, one of the main goals for creating the LiDo RDF dataset was to enable the citation of the term and concept data. Since not all users are, however, familiar with navigating through a Linked Data graph and using the resource URIs for citation purposes, a web interface that is intuitive and easy to browse had to be provided as well. In order to create such an interface, the URIs of the resources that are contained within LiDo RDF could be reused, as has been stated earlier in Section 4.3. The interface is accessible under the URL `http://lidordf.aksw.org/glossary/` which retrieves the respective resource identifier from LiDo RDF whenever a specific term or concept entry is selected[26]. The screenshot of the English term entry *quantifier* in Figure 2 illustrates the layout of the web interface. Only the data that is transformed from the LiDo TBD source data into RDF can be searched, while additional links to external datasets are only accessible via the SPARQL endpoint or the Linked Data view of Lido RDF.

All in all the interface is similar to the LiDo TBD website. The main difference consists in the arrangement of the term, concept and bibliographic data boxes. While the left side of the term data box in principle shows the semantic relations between linguistic concept resources within the dataset, these are displayed, however, by using their corresponding term expressions in the selected language. The right side of the term data box then shows the corresponding expressions of the chosen concept in other languages. Further, the bibliographic references that correspond to a concept can be selected as another view within the concept data box while the term data above stays visible. However, a detailed search interface for the bibliographic resources has not been implemented. Finally, the novelty and only additional feature of the LiDo RDF based web interface are the two "cite" buttons, which provide a pre-formulated reference for a selected term or concept entry that can be directly copied from the pop-up window (cf. Figure 3).

[23] `https://wiki.okfn.org/Working_Groups/Linguistics/How_to_contribute`

[24] `https://old.datahub.io/dataset/bll-thesaurus`

[25] Please cf. to (Chiarcos et al., 2016) for more details of the usage of the OLiA Ontologies within the BLL Thesaurus development.

[26] Note that the difference between for example `http://lidordf.aksw.org/resource/Term_1651` and `http://lidordf.aksw.org/glossary/Term_1651` lies only in the way the LiDo RDF data is displayed, i.e. as Linked Data view or within the web interface.

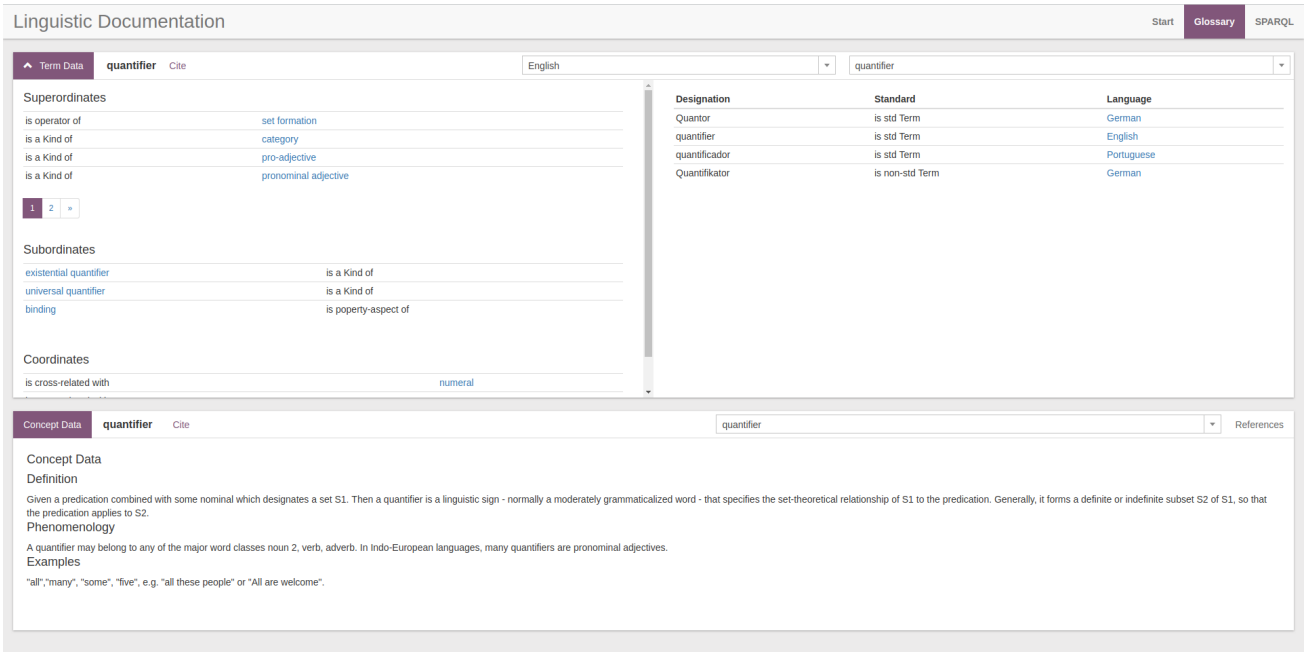


Figure 2: Screenshot of the LiDo concept 'quantifier' as displayed within the provided web interface.

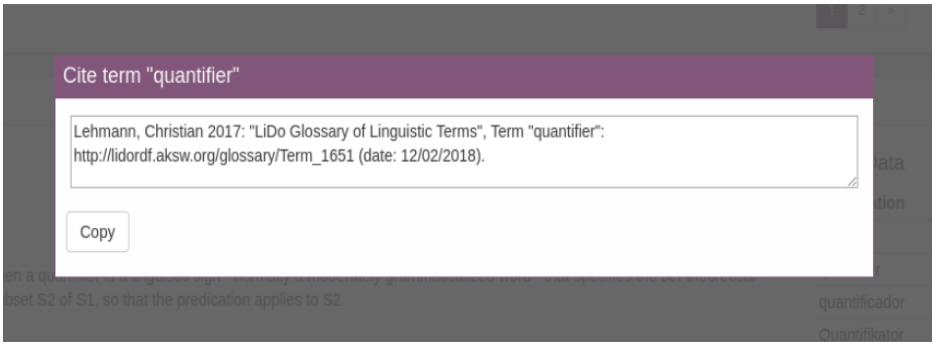


Figure 3: Screenshot of the citation function for the English term *quantifier* provided within the web interface.

The different layout presents just another option next to the existing LiDo TBD website. The design and features are still under development and might change if the user feedback demands refinements or other additional functionalities.

6. Data Quality

Automatically generated data is often subject to data quality issues. In order to preserve the high quality of the thoroughly compiled LiDo TBD, we conducted two different kinds of quality checks. The first constitutes a syntax validation, that ensures that the data is free of formal errors and can be processed by machines without any problems. The second check concerns the completeness of the data and assures that no data has been lost during the transformation process. Therefore, specific SQL and SPARQL queries have been created that count and compare the entries of both datasets. This validation process gives an immediate feedback about passing or not passing entries[27]. The completeness of the generated LiDo RDF data is then verified when all queries in the LiDo PostgreSQL source dataset and the LiDo RDF dataset produce the same output results. Only after both data quality checks are successfully applied the LiDo RDF dataset and its future versions will be published.

7. Conclusion and Future Work

In this paper we described LiDo RDF, i.e. the Linked Data version of the LiDo TBD. The primary goal of enabling the citation of single term and concept entries of the LiDo TBD could be achieved by providing a similar search interface that reuses the LiDo RDF resource URIs in a slightly altered way. Following the aim to contribute to the provision of this linguistic resource we also presented the LiDo RDF project that maintains the human-readable search interface as well as the SPARQL endpoint.

[27] See the results at `http://lidordf.aksw.org/validation/`.

As a result, the LiDo TBD is not an isolated dataset on the Web anymore, but reusable for human users and machine processing alike. Furthermore, the underlying modelling framework of LiDo TBD and LiDo RDF that represents linguistic terminology as an interrelation of concepts and corresponding terms by means of a consistent and manageable set of relations can serve as a starting point for the discussion of standards for modelling linguistic terminology as Linked Data in the future. This is especially interesting with regard to already existing standard vocabularies like OntoLex-Lemon for lexical language data, for which another module for specifically representing terminological data might be developed.
Finally, the LiDo RDF dataset could be integrated into the LLOD Cloud and is, thus, visible to the broader linguistic Linked Data research community. By that, we hope to initiate the collaboration with the creators of other datasets, such as the BLL Thesaurus or Grammis, in order to conduct a high quality and far-reaching interlinking task. This will consequently benefit linguistic research in general by contributing to a large and interconnected knowledge graph for linguistic terms, concepts and bibliographic data.

Acknowledgment This paper‘s research activities were funded by grants from the H2020 EU projects ALIGNED (GA-644055) and the Smart Data Web BMWi project (GA-01MD15010B).

8. Bibliographical References

Chiarcos, C., Fäth, C., Renner-Westermann, H., Abromeit, F., and Dimitrova, V. (2016). Lin|gu|is|tik: Building the linguist's pathway to bibliographies, libraries, language resources and linked open data. In *LREC*.

D'Arcus, B. and Giasson, F. (2011). *Bibliographic ontology specification. Specification document, 4 November 2009.* Retrieved August 10 (2009).

Farrar, S. (2010). *General Ontology for Linguistic Description (GOLD).* Department of Linguistics (The LINGUIST List), Indiana University.

Klimek, B., McCrae, J. P., Lehmann, C., Chiarcos, C., and Hellmann, S. (2017). Onlit: An ontology for linguistic terminology. In *Language, Data, and Knowledge: First International Conference, LDK 2017, Galway, Ireland, June 19-20, 2017, Proceedings*, volume 10318, pages 42–57. Springer.

Lehmann, C. (1976). *Um sistema de documentação para a lingüística*. Pontifícia Universidade Católica do RGS, Instituto de Letras e Artes.

Lehmann, C. (1996). Linguistische Terminologie als relationales Netz. In *Nomination—fachsprachlich und gemeinsprachlich*, pages 215–267. Springer.

McCrae, J. P., Bosque-Gil, J., Gracia, J., Buitelaar, P., and Cimiano, P. (2017). The ontolex-lemon model: Development and applications. In *Proceedings of eLex 2017 conference*.

Stadler, C., Unbehauen, J., Westphal, P., Sherif, M. A., and Lehmann, J. (2015). Simplified rdb2rdf mapping. *Proceedings of the 8th Workshop on Linked Data on the Web (LDOW2015), Florence, Italy*.

Suchowolec, K., Lang, C., Schneider, R., and Schwinn, H. (2017). Shifting complexity from text to data model. adding machine-oriented features to a human-oriented terminology resource. In *Language, Data, and Knowledge: First International Conference, LDK 2017, Galway, Ireland, June 19-20, 2017, Proceedings*, volume 10318. Springer.

Towards a Linked Open Data Edition of Sumerian Corpora

Christian Chiarcos*, Émilie Pagé-Perron◇, Ilya Khait*, Niko Schenk*, Lucas Reckling◇
*Goethe University Frankfurt, Germany, ◇University of Toronto, Canada
{chiarcos, khait, schenk}@informatik.uni-frankfurt.de, {e.page.perron, lucas.reckling}@mail.utoronto.ca

Abstract

Linguistic Linked Open Data (LLOD) is a flourishing line of research in the language resource community, so far mostly adopted for selected aspects of linguistics, natural language processing and the semantic web, as well as for practical applications in localization and lexicography. Yet, computational philology seems to be somewhat decoupled from the recent progress in this area: even though LOD as a concept is gaining significant popularity in Digital Humanities, existing LLOD standards and vocabularies are not widely used in this community, and philological resources are underrepresented in the LLOD cloud diagram (`http://linguistic-lod.org/llod-cloud`).

In this paper, we present an application of Linguistic Linked Open Data in Assyriology. We describe the LLOD edition of a linguistically annotated corpus of Sumerian, as well as its linking with lexical resources, repositories of annotation terminology, and the museum collections in which the artifacts bearing these texts are kept. The chosen corpus is the Electronic Text Corpus of Sumerian Royal Inscriptions, a well curated and linguistically annotated archive of Sumerian text, in preparation for the creating and linking of other corpora of cuneiform texts, such as the corpus of Ur III administrative and legal Sumerian texts, as part of the Machine Translation and Automated Analysis of Cuneiform Languages project (`https://cdli-gh.github.io/mtaac/`).

Keywords: Linked Open Data, Linguistic Linked Open Data, Sumerian, RDF, Linked Dictionaries.

1. Background

The Sumerian language is an agglutinative isolate that was written using the cuneiform script[1] in ancient Iraq; it is the first recognized written language. Assyriologists have long been painstakingly transcribing cuneiform texts for their research. These transliterations are generally published on paper, and to a lesser extent collected in electronic archives as part of perhaps a dozen projects. Unfortunately, these digital initiatives do not share the same encoding, and the computational toolset available for processing these data is limited.

As a collaboration between specialists in Assyriology, computer science and computational linguistics at the Goethe University Frankfurt, Germany, the University of California, Los Angeles (UCLA), the University of Toronto, Canada, and the Cuneiform Digital Library Initiative (CDLI)[2], our recently funded project "Machine Translation and Automated Analysis of Cuneiform Languages" (MTAAC)[3] aims to tackle natural language processing challenges presented by these ancient Mesopotamian languages (Pagé-Perron et al., 2017).

MTAAC is developing a methodology and a set of state-of-the-art NLP components geared to the processing of cuneiform text. The homogenized, annotated, and translated texts, accompanied by extracted information and prepared using this pipeline, will be made available both to designated audiences and machines to facilitate the study of the language, culture, history, economy and politics of the ancient Near East. Beyond applying statistical and neural techniques, linked data formalisms and open vocabularies will be employed to facilitate the reusability of these data, thereby contributing to interoperability, in particular with other philological portals,[4] and also to encourage research reproducibility. Additionally, because the field of Assyriology suffers from a lack of shared standards, we expect that our linked data approach will provide a serious opportunity for data integration both within the field and beyond it.

The terminological and technological foundations described in this paper represent the basis for the future publication of cuneiform corpus data and their annotation, developed within the MTAAC project. In particular, this includes the major administrative and legal Sumerian corpus of the Ur III period (2100-2000 B.C.) and the deployment of the infrastructure to enable linking all cuneiform text in the encompassing Cuneiform Digital Library Initiative (CDLI), which curates the largest corpus of digitized cuneiform artifacts.

According to Chiarcos et al. (2013), the primary objectives of linking language resources is to foster:

- Representation: a flexible representation format for research data (corpora, dictionaries, extracted information) and metadata (vocabularies);
- Interoperability: common RDF models can easily be integrated;
- Federation: data from multiple sources can be combined effortlessly;
- Ecosystem: tools for RDF and linked data are widely available under open-source licenses;

[1] The cuneiform script is formed by impressing a sharpened reed stylus into fresh clay, creating wedge-like impressions to form distinct signs. It was employed from ca. 3500 BC to the end of the first millennium BC to write texts in Sumerian, the Semitic language Akkadian, and a number of other languages spoken in the region.

[2] `https://cdli.ucla.edu`.

[3] `https://cdli-gh.github.io/mtaac`.

[4] E.g. Syriac `http://syriaca.org`, Hebrew `http://hebrew-terms.huji.ac.il/`, Indo-European, and Caucasian languages, `http://titus.fkidg1.uni-frankfurt.de/`.

– Expressivity: existing vocabularies help express linguistic resources;
– Semantics: common links express what is meant;
– Dynamicity: web data can be continuously improved.

Developing an (L)LOD edition for Sumerian and linking representative language resources includes the application of the following ontologies:

– lemon/ontolex[5] for lexical data;
– CIDOC/CRM[6] for object metadata;
– lexvo[7] for language identification;
– Pleiades[8] for geographical information;
– OLiA[9] for linguistic annotations.

While these are established *de facto* standards in the field, editing principles for philological *corpora* are only now emerging, with different alternative vocabularies (POWLA, NIF, TELIX)[10] currently being discussed. Consequently, we focus on this aspect. Our proof-of-concept relies on the morphologically annotated Electronic Text Corpus of Sumerian Royal Inscriptions (ETCSRI) (Zólyomi et al., 2008) and describes the application of CoNLL-RDF that serves as LOD representation within the CDLI as part of the MTAAC project.

So far, only two projects currently provide open *annotated* cuneiform text. The first is the Open Richly Annotated Cuneiform Corpus (ORACC),[11] a portal hosting sub-projects with local glossaries. Most projects focus on sources in the Akkadian language.[12] Not all ORACC projects are annotated, but the platform offers a standalone lemmatizer which provides an interface to call a server service for the semi-automated annotation of lexical information. The second is the Electronic Text Corpus of Sumerian Literature (ETCSL) (Black et al., 1998–2006), which presents most known Sumerian literary compositions. A handful of other projects offer digital access to unannotated text.[13]

Linked Open Data has previously been applied to the humanities, including linguistics, NLP and other language sciences (Chiarcos, C. et al., 2012). Beyond prosopography and gazetteers on the one hand (e.g., Pelagios, perio.do), and early efforts to create addressable units for passages in texts on the other (CTS, Canonical Data Services), applications of (L)LOD to computational philology are rare, and indeed absent from the field of cuneiform studies. There are, however, projects that touch upon the classification of artifacts; for instance the Modref project (Tchienehom, 2017)[14] employs CIDOC-CRM, as is customary for the classification of museum artifacts, and connects three different collections, including the CDLI. Similarly, the British Museum provides a CIDOC-CRM-based SPARQL end point[15] that encompasses almost 22% of all CDLI artifact entries. By following explicit links within such repositories, Linked Data technology allows us to query disparate artifacts across different collections. In addition, SPARQL 1.1 federation, as described further below, allows us to access these metadata repositories remotely and to link CDLI data with them.

Two pioneering experiments on the application of *ontologies* to Sumerian are to be noted: Jaworski (2008a) presented an ontology-based approach to the semantic parsing of a domain-specific subset of Ur III administrative texts from the CDLI with the goal of tracing patterns of transfer of cattle between individuals and institutions. While he has been successful in identifying several thousand transactions, this approach is limited to a highly restricted domain. Neither the annotations nor the parser are available, but they are well documented in Jaworski (2008b). Another experiment was concerned with annotating the ETSCL corpus mentioned above with an ontology of literary concepts. The mORSuL ontology was developed to attach CIDOC-CRM to Ontomedia (Nurmikko, 2014; Nurmikko-Fuller, 2015);[16] however, this has only reached the status of a case study. No data are available from either of these applications, nor are their data being linked with the original corpora or other resources.

While these experiments show the potential interest that the scientific community would have in Sumerian corpus data being published in accordance with Semantic Web principles, neither of them actually aim to provide *Linked* (Open) Data as an end product. By bringing together corpus data, lexical data, linguistic annotations and object metadata, the MTAAC project is thus breaking new ground for the field of Assyriology, as well as computational philology in general.

2. Neo-Sumerian (Ur III) corpora

2.1. Ur III Data in CDLI

The Cuneiform Digital Library Initiative (CDLI) collects and makes available on the web metadata and, to a lesser extent, transliterations, transcriptions and translations, of all artifacts bearing cuneiform inscriptions. The project is based on the efforts of an international group of language specialists, museum curators and historians of science. World collections hold approximately 550,000 objects, and the CDLI has catalogued some 334,000 of them. Forty percent of these texts are written in Sumerian, of

[5] http://lemon-model.net/,https://www.w3.org/community/ontolex/wiki/Final_Model_Specification.

[6] http://www.cidoc-crm.org/.

[7] http://www.lexvo.org/.

[8] https://pleiades.stoa.org/.

[9] http://www.acoli.informatik.uni-frankfurt.de/resources/olia/.

[10] https://sourceforge.net/projects/powla/, http://aksw.org/Projects/NIF.html, http://ontorule-project.eu/telix.

[11] http://oracc.museum.upenn.edu.

[12] Akkadian is a Semitic language that was written using the cuneiform script. It was used from the second half of the 3rd millennium up to the end of the 1st millennium BC.

[13] Among others the CDLI, the Database of Neo-Sumerian Texts (BDTNS), a database of texts dating to the Ur III period http://bdts.filol.csic.es/, and Archibab, specializing in the Old Babylonian period (ca. 1900-1600 BC) http://www.archibab.fr.

[14] http://triplestore.modyco.fr:8080/ModRef.

[15] https://collection.britishmuseum.org/sparql.

[16] http://www.contextus.net/ontomedia.

which 2/3 were produced during the Ur III period, which refers to a dynasty of the end of the 22nd, and the whole of the 21st centuries BC.
In Ur III times, the (Neo-)Sumerian language dominated the cultural sphere: Sumerian texts were mass-produced in this era. However, this was apparently accompanied by its gradual decline as the common spoken language. Nevertheless, Sumerian remained the prevalent language not only of literature and royal texts, but also of legal, administrative, and economic documents.

Genre	Texts	/Total	Translit.	/T. total	Translated	/Translit.
Admin.	97675	96.98%	67697	96.23%	1582	2.34%
Royal	1529	1.52%	1468	2.09%	264	17.98%
Letter	744	.74%	700	1.00%	12	1.71%
Legal	451	.45%	383	.54%	9	2.35%
Other	319	.31%	100	.24%	13	34.32%
Total	100718		70348		1880	2.67%

Table 1: Ur III material in the CDLI

The Sumerian Ur III corpus available on CDLI is summarized in Table 1. The category "other" encompasses the literary, school, lexical, prayer, votive, scientific, other, uncertain, and fake genres. Although most literary and royal texts are not accompanied by a translation, the composite texts to which they are related are themselves often translated. The translations available can be viewed for all composite witnesses in the CDLI score pages, at `https://cdli.ucla.edu/tools/scores/partitur-index.html`.
As evident from the table, most of the data are of an administrative nature, but thus far untranslated. Therefore, we focus on this genre but aim to develop tools and resources to link with CDLI data in general.

2.2. Morphological Annotation in ETSCRI

The Electronic Text Corpus of Sumerian Royal Inscriptions (ETCSRI) is a sub-project of ORACC assembling all Sumerian royal inscriptions, compiled, verified and annotated by Zólyomi et al. (2008). ETCSRI is fully translated, lemmatized and morphologically annotated, and it provides transliterations and translations. Additionnaly, glossaries based on the project, which include named entities, are available for consultation.
The original texts on which this study is based are ancient inscriptions in the Sumerian language, written on diverse artifacts commemorating actions and dedications of higher elites that lived in ancient Iraq between 2900 and 1600 BC. Many of these come from the Ur III period, but they cover the history of the Sumerian language. All texts provide cross-references with the CDLI.
In addition to ETCSRI, the Electronic Text Corpus of Sumerian Literature (ETCSL)[17] also provides lemmatized, morphologically annotated and translated text in Sumerian, albeit following different (and, in parts, dated) transliteration principles. This corpus comprises a variety of literary compositions which were written down from the Ur III period onwards, mostly in the Old Babylonian period.
For our proof-of-concept, we chose the ETCSRI corpus as a basis for our efforts principally because of the reusability of the annotations. Not only does it have a substantial overlap with CDLI Ur III data, but its transliterations are up-to-date and the morphological annotations are based on a good morphological model of Sumerian. Moreover, the data are open and released to the public domain.[18]

2.3. Beyond Royal Inscriptions

The MTAAC project aims to complement the existing annotated and translated corpora of literature and royal inscriptions with a corpus of Ur III data in general, in particular its administrative texts. Due to the amount of data in question, only portions of it, however, can be manually annotated or translated. One objective of the MTAAC project is thus to provide automated analyses for this purpose.
Morphological annotations are based on ETSCRI (Zólyomi et al., 2008). ETSCRI-style part-of-speech annotations include named entity classification, for which we build on earlier efforts towards semi-automated entity annotation (Liu et al., 2015, SNER).[19]
Ur III administrative data are comparably easy in terms of morphosyntax, since morphology is often not expressed in writing (though the information may be inferred from the context). However, this also means that morphology is somewhat uninformative, so that effective querying and searching of these data requires structural analysis. We thus retrieve relational information in addition to morphosyntax as found in ETCSRI. This extends earlier work on (semantic) parsing by Jaworski (2008a) in that we do not rely on domain-specific rules, rather we employ state-of-the-art machine learning techniques. At the moment, we are evaluating the suitability of the Universal Dependency (UD)[20] schema and UD-based annotation projection for these kind of data, possibly to be augmented with an additional layer of semantics (Peterson et al., 2014): since administrative texts are not exclusively composed of grammatical sentences but also often comprise lists, semantic role labeling (SRL) annotation is considered crucial for this genre. The SRL inventory will be based on Hayes (2000) and Jaworski (2008b), yet grounded in the English PropBank.
Since further details of the annotation process will be presented elsewhere, we focus here on infrastructural measures.

3. Towards Linked Data

3.1. Corpus Representation

The **(Canonical-)ASCII Transliteration Format** ((C-)ATF) is a text encoding format developed by CDLI and the Electronic Pennsylvania Sumerian Dictionary (ePSD)[21], a text encoding scheme for cuneiform transcriptions which was designed as a human-friendly archival format to complement the usage of machine-oriented XML formats for annotations (Koslova and Damerow, 2003). Basically, it is a data entry and storage format. It first encodes the

[17] `http://etcsl.orinst.ox.ac.uk`.

[18] See ORACC `http://oracc.museum.upenn.edu/doc/opendata/index.html` "Open Data". Until recently, the data were available only under a Creative Commons Share alike license but the release of the data in JSON format was done under the public domain.

[19] `https://wwunlp.github.io/sner/`.

[20] `http://universaldependencies.org/`.

[21] `http://psd.museum.upenn.edu/`.

CDLI text-ID along with its designation, some feature tags to encode language and medium type, and a line-by-line numbered transliteration, augmented with interlinear annotations such as normalization, translation, and comments on structure and content. It also uses specific conventions to annotate structure. Transliteration lines are restricted to the ASCII character range.

See, as an example, one of the exemplars of a royal inscription of king Amar-Suen[22]:

```
&P226657 = RIME 3/2.01.03.01, ex.  07
#atf:  lang sux
@object brick
@surface a
1.  {d}amar-{d}suen
2.  nibru{ki}-a
3.  {d}en-lil2-le
4.  mu pa3-da
5.  sag-us2
6.  e2 {d}en-lil2-ka
7.  nita kal-ga
8.  lugal uri5{ki}-ma
9.  lugal an ub-da limmu2-ba
@surface b
1.  {d}amar-{d}suen
2.  nibru{ki}-a
3.  {d}en-lil2-le
4.  mu pa3-da
5.  sag-us2
6.  e2 {d}en-lil2-ka
7.  nita kal-ga
8.  lugal uri5{ki}-ma
9.  lugal an ub-da limmu2-ba
```

The format was subsequently extended in ORACC to provide support for additional annotation layers. ORACC-ATF[23] uses Unicode characters in the transliteration lines. Additionally, annotations are stored in comment lines in between lines of text.[24]

The ETCSRI edition of the Sumerian royal inscription above and its morphological annotation are available from ORACC in XHTML and JSON formats [25]. The ATF and XML versions are available only to privileged users.

ORACC uses comment lines to store more information about the text, such as lemma information, but it is impossible to add another layer of annotation, such as syntax, for instance, into the ATF format. Aligning with the initial philosophy of the C-ATF format, we do not intend to extend the specification but instead we will supplement it with community standards, in this case, the CoNLL TSV format. Due to the specifics of our data, we define our own inventory of columns for information storage, although we convert this CDLI-CoNLL format to fully-fledged CoNLL-U for further processing.[26] In the case of the proof-of-concept presented here, the original text and annotations are extracted from XHTML and converted to CoNLL. As for the full MTAAC project, text and annotations are stored as follows: C-ATF file for the textual and text structure data, and CDLI-CoNLL files for the annotation. A validation script to check the alignment between both files is under preparation.

These data formats are primary *annotation* formats; they do not provide data structures that can be easily queried or transversed using off-the-shelf technology. As such, for the sake of a web publication, we thus provide an RDF converter which is built on the CoNLL2RDF tool[27], whose output serves as the basis for linking the corpus. We will, in a subsequent step, make the corpus available in other machine friendly formats such as (RDF-) XML and JSON.

3.2. CoNLL-RDF

While the development of vocabularies for lexical data has progressed significantly and was recently aggregated in the lemon community standard (see below), the representation of linguistically annotated text is a more heterogeneous area, with highly generic models for *richly annotated corpora* on the one hand (Chiarcos, 2012), and problem-specific models on the other, such as *NLP web services* (Hellmann et al., 2013) or semantic annotation (Sanderson et al., 2017). All these data models can be serialized in different RDF formats – which are, however, generally verbose and not intended for human consumption nor direct (string-based) manipulation.

Finally, CoNLL-RDF (Chiarcos and Fäth, 2017) fills in this gap by providing a middle ground that accounts for the needs of NLP specialists: easy to read, easy to parse, and close to conventional representations. Important is the format's *potential for LLOD integration*: it is directly processable using Semantic Web technology, thereby facilitating interoperability, interpretability, linkability, queryability, transformability, database support, and integration with web technologies. In addition, CoNLL-RDF complements its data model with layout conventions (and a formatter) to facilitate the easy low-level access to the CoNLL TSV format.

An example of RDF rendering of an ETCSRI excerpt[28] is

[22] https://cdli.ucla.edu/P226657.

[23] On the differences between the ATF dialects see http://oracc.museum.upenn.edu/doc/help/editinginatf/cdliatf/index.html.

[24] http://oracc.museum.upenn.edu/doc/help/editinginatf/primer/structuretutorial/index.html, http://oracc.museum.upenn.edu/doc/help/lemmatising/primer/.

[25] http://oracc.museum.upenn.edu/etcsri/Q000981.

[26] This is a characteristic of most members of the CoNLL format family; tool support for CoNLL thus typically involves routines for reordering, merging or dropping columns.

Using the CoNLL format has the additional advantage that annotated corpora stored in other formats can easily be converted to a standard format without requiring extensions of a proprietary archival representation. This was especially helpful to facilitate working with the ETCSRI Corpus which is internally stored both as ATF and TEI/XML but published only as XHTML and JSON; their ORACC-ATF files are not open and might not contain all of the annotations since the ORACC website gathers data both from the ATF and the glossary to generate the final rendering of the text output.

[27] https://github.com/acoli-repo/conll-rdf.

[28] The example uses a fragment from a short votive inscription by Ur-Namma, the first ruler of the Ur III dynasty (ETCSRI's Ur-Namma 2, line 3), see

```
@prefix : <http://oracc.museum.upenn.edu/etcsri/Q000935#> .
@prefix conll: <http://ufal.mff.cuni.cz/conll2009-st/task-description.html#> .
@prefix rdf: <http://www.w3.Org/1999/02/22-rdf-syntax-ns#> .
@prefix terms: <http://purl.org/acoli/open-ie/> .
@prefix nif: <http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#> .
@prefix rdfs: <http://www.w3.Org/2000/01/rdf-schema#> .

:s2_0 nif:nextSentence :s3_0 .

:s3_0 a nif:Sentence .

:s3_1 a nif:Word; conll:WORD "lu2";
                  terms:lemma <http://psd.museum.upenn.edu/epsd/epsd/e3356>;   conll:BASE "lu2";
                  conll:CF "lu";   conll:EPOS "n"; conll:FORM "lu2";
                  conll:GW "person";
                  conll:HEAD :s3_0; conll:ID "1"; conll:LANG "sux"; conll:MORPH "N1=lu";
                  conll:MORPH2 "N1=stem"; conll:NORM "lu"; conll:POS "N"; conll:SENSE "person";
                  nif:nextWord :s3_2 .

:s3_2 a nif:Word; conll:WORD "e2";
                  terms:lemma <http://psd.museum.upenn.edu/epsd/epsd/ell66>;   conll:BASE "e2 ";
                  conll:CF "e";   conll:EPOS "n"; conll:FORM "e2";
                  conll:GW "house";
                  conll:HEAD :s3_0; conll:ID "2"; conll:LANG "sux"; conll:MORPH "N1=e";
                  conll:MORPH2 "N1=STEM"; conll:NORM "e"; conll:POS "N"; conll:SENSE "house, temple";
                  nif:nextWord :s3_3 .

:s3_3 a nif:Word; conll:WORD "{d}nanna";
                                                                   conll:BASE "{d}nanna";
                  conll:CF "Nanna"; conll:EPOS "DN"; conll:FORM "{d}nanna\\gen\\abs";
                  conll:GW "1";
                  conll:HEAD :s3_0; conll:ID "3"; conll:LANG "sux"; conll:MORPH "N1=Nanna.N5=ak.N5=Ø";
                  conll:MORPH2 "N1=name.N5=gen.N5=abs"; conll:NORM "Nanna.ak.Ø"; conll:POS "DN"; conll:SENSE "1" .
```

Figure 1: Illustration of a CoNLL-RDF representation of an ETCSRI excerpt. (One word per line format expanded for better legibility.)

provided in Figure 1. The first line in a sentence refers to the sentence URI and defines it as a `nif:Sentence`. The second line holds the first content word and defines it as a `nif:Word`, followed by its `conll:WORD`, other annotations in alphabetical order of their properties, concluding with a `nif:next` statement pointing to the next word in the sentence (if available). The relation between words and sentences is established via `conll:HEAD`. `conll:WORD`, `conll:HEAD`, etc., are properties that express the annotations found in the respective columns in the original CoNLL file. Turtle provides different triple separators: "." separates independent triples, ";" separates one from the next that shares the same (omitted) subject, "," enumerates multiple objects for one property of the same subject. In CoNLL-RDF, all of these are written in one line, so that the CoNLL convention of WPL annotations is respected; different properties are separated using ";", different values enumerated with ",". Finally, the sentence is concluded with a `nif:nextSentence` statement (if more sentences follow).

Figure 2 illustrates the overall conversion workflow including different data formats that were involved up to this point, i.e. from raw (ATF) text to (CoNLL)RDF-compliant corpus representations. A final step involves in particular the linking of external knowledge sources, such as annotations, lexical data, and meta data, which we describe in the following section.

3.3. Annotation Pipeline

Linking the annotations is the last task in our pipeline and it is done automatically. The MTAAC project comprises two major steps, the semi-automated annotation of our gold corpus and the automated annotation performed based on this gold corpus. The first step starts with the validation of the ATF text data, which are then morphologically pre-annotated using a dictionary-based tool that is fueled by a database of forms and their associated morphological analyses[29]. A human annotator then verifies the morphological annotations and advances the text in the pipeline. The text is then pre-annotated using a rule-based syntax annotation. The syntactic annotations are generated using our RDF-

`http://oracc.museum.upenn.edu/etcsri/Q000935`. Note that the fragment under consideration, translated as "the man who (built) the temple of Nanna", is in fact part of a longer sentence that we have shortened for the sake of illustration.

[29] See our morphology pre-annotation tool here: `https://github.com/cdli-gh/morphology-pre-annotation-tool`.

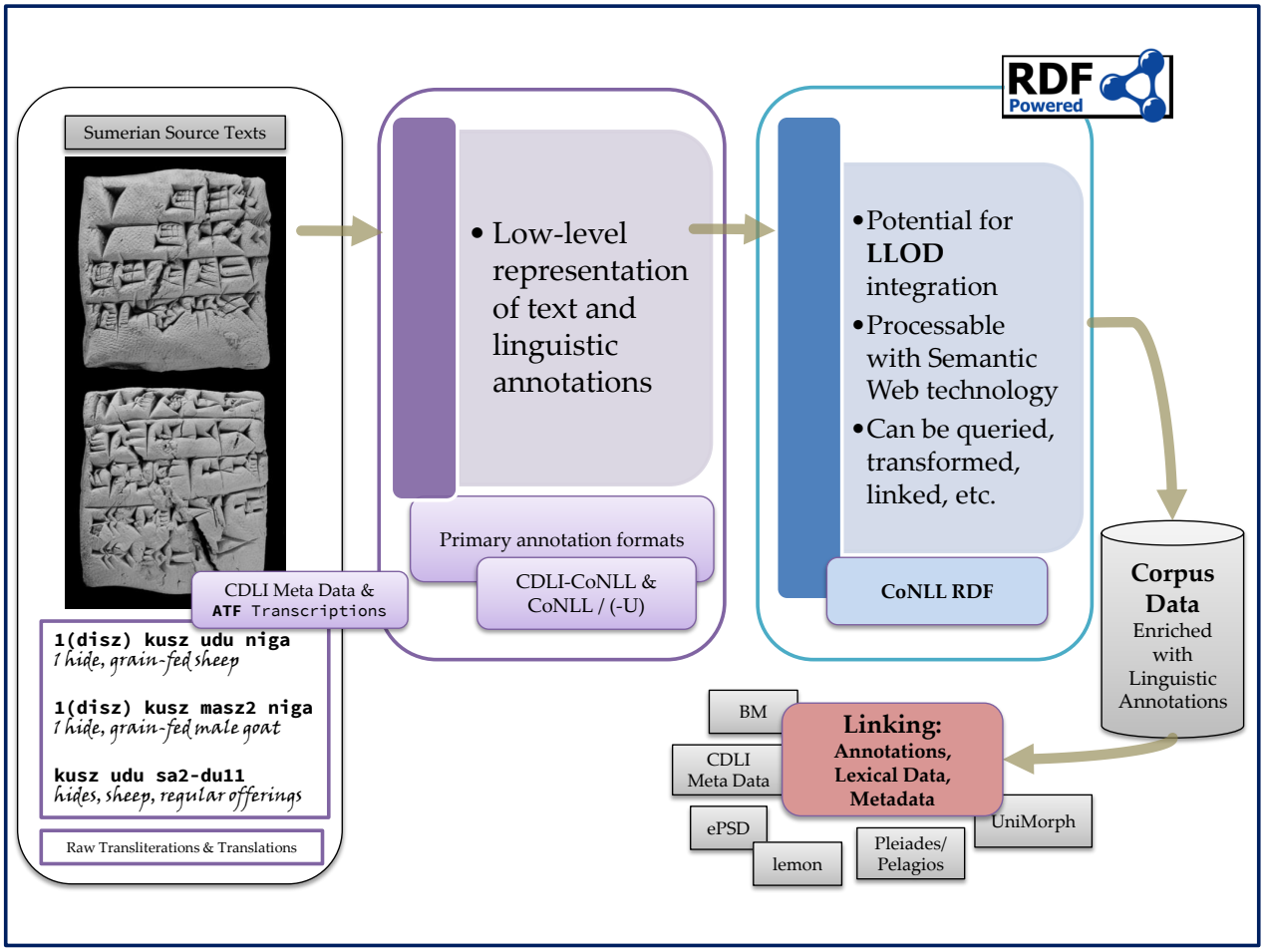


Figure 2: Workflow illustration for the LOD edition of Sumerian texts

based pre-annotation tool[30] and then manually adjusted, when needed. This pipeline is more thoroughly described and evaluated in our LDL2018 paper[31].

The annotated texts produced in this semi-automated pipeline are in the exact same format at those exiting the automated pipeline, thus our linking process applies to both, while it also handles the ETCSRI data we use for our proof-of-concept.

3.4. Linking Annotations

In the case of morphology, and as part of the proof-of-concept we have developed, we map the existing ETCSRI morphological annotation scheme[32] of which we use the "MORPH 2" tag, with the Universal Morphological Feature Schema (UniMorph) specifications[33], though a Turtle RDF mapping of the ORACC:ETCSRI morphological tags inventory with UniMorph[34]. UniMorph is able to define morphological features in language-independent terms, allowing for facile translation between languages employing the schema (Sylak-Glassman, 2016, 3); this effectively brings Sumerian for the first time into the language corpora that are linked by their linguistic annotations, making it now available for richer cross-linguistic research.

Some challenges have emerged in the mapping, since Sumerian is a language isolate. The first challenge is the verb modality expressed through a series of prefixes. Both for the LLOD mapping using UniMorph and for Universal Dependencies mapping when converting our home CoNLL format to ConLL-U Feats field, these prefixes cannot all find an adequate home. We are currently preparing proposals to include adequate tags in both schemes. The second hurdle is the Sumerian enclitic copula, which also has no equivalent analysis in UniMorph. But overall, the impressive flexibility of UniMorph made it possible to combine tags to account for the exact meaning of certain morphemes, for example in the case of the locative morphemes, that we represent with "IN+ESS", "ON+ESS" and "APUD+ESS".

We use CoNLL-RDF as a working format to leverage the capabilities of SPARQL for syntactic annotation. Linking to the syntactic data is made possible through CDLI-CoNLL, the main format used in our corpusto store annotations. For this purpose, we provide and consult an OWL representation of the CDLI annotation scheme and its link-

[30] Seehttps://github.com/cdli-gh/mtaac_work/tree/master/parse

[31] http://ldl2018.linguistic-lod.org/.

[32] vhttp://oracc.museum.upenn.edu/etcsri/parsing/index.html

[33] http://unimorph.org/.

[34] https://github.com/cdli-gh/mtaac_work/blob/master/lod/annotations/um-link.ttl.

ing with UD POS, feature and dependency labels.

3.5. Linking Lexical Data

The ePSD is the only comprehensive and available digital resource for Sumerian vocabulary. We converted the ePSD data to an index of deep links, expressed as a lemon dictionary. Our pipeline consults (or constructs, if it is not found) a lemon/ontolex compliant index for the ePSD, whose URIs are provided to the linking. Because of the structure of the ePSD, links point to guide word entries. While we prepare the MTAAC Ur III research corpus, additional local lexical resources for the Sumerian language will be provided.

3.6. Linking Metadata

Cuneiform text is inscribed on objects about which specialists gather a set of metadata that is useful when integrated into a method for text analysis. Information such as provenience, period, and size of the artifact are examples of such characteristics. Other information such as the condition of the object, the museum in which it is kept, and the publications mentioning the text in question are all helpful for the discovery and study of the artifact.

The CDLI catalogue data live in a MySQL database and is exported daily in CSV format. For this exercise we used the CSV data package. Fields of interest are: museum no (here BM: British Museum), id_text (CDLI object id), composite, height, thickness, width, material, and finally period. The composite number regroups witnesses of a composition that was copied on different artifacts.

Our modeling approach follows that of the British Museum: the composite field translates to `?composite a crm:E34_Inscription` and id_text to `?object URI ? object crm:P65_shows_visual_item ?composite` [35] and lastly, the museum_no maps with `owl:sameAs` which is a resolved museum number.

Using this model, we convert the data to RDF with the csv2rdf tool supplemented with embedded custom turtle templates that we prepared for the occasion.[36] We link to external metadata repositories: the Modref project[37] and the British Museum.[38] The ModRef project's goal is to "move heterogeneous data into triplestores also called data warehouses or collections of RDF files in order to improve the sharing, exchange and discovery of new knowledge" (Tchienehom, 2017). Their model formalizes three different collections in a coherent model: the CDLI catalogue, the ObjMythArcheo database,[39] a corpus of archaeological objects related to mythological iconography, and BiblioNum, a DL about France in the 20th century. Because text is so much more meaningful with its context, linking catalog information of the artifacts on which the texts are inscribed greatly enriches our linking model for Assyriologists, who need this information to understand the texts as well as to compare these artifacts with other classes of artifacts that possess similar characteristics, such as provenience, period, size, and the collection in which they are kept. Cuneiform objects are for the most part studied only in the field of Assyriology. Making them available in the semantic web increases the possibility of including them in larger-scale studies, thus overcoming this limitation in the scope of research.

4. Summary and Outlook

In this paper, we described the (L)LOD edition and linking of corpora for Assyriology. We apply our model to ETCSRI as a proof-of-concept for future application to the Neo-Sumerian (Ur III) administrative corpus targeted by the MTAAC project. We have successfully integrated these diverse and distributed knowledge sources (linguistic and non-linguistic):

- CDLI (local resource; CoNLL-RDF plus CIDOC-CRM)
- ORACC:ETSCRI (by conversion; CoNLL-RDF)
- ePSD (by conversion and links to HTML; lemon)
- ModRef & BM (by federation; CIDOC-CRM)

With this experiment we also demonstrated the applicability and usefulness of (L)LOD standards to Assyriology. Other vocabularies such as Pleiades, Snap dragon and perio.do, among others, can be added analogously. The annotation data on Sumerian morphology and syntax will be produced in the future by the (semi-)automatic annotation pipeline under development, which we see as a crucial step towards an (L)LOD edition of cuneiform corpora. Lastly, this proof-of-concept is now our tested and refined template for the infrastructure that will be integrated into the CDLI as part of the MTAAC project. As such, we welcome feedback to further strengthen our model.

Although creating new linguistic data and tools to manipulate this data should improve the research outcomes for Assyriologists, we realize that knowledge circulation is directly dependent on access, classification and discoverability. As such, the linking of linguistic and other resources has been built in as an essential part of the MTAAC project. We also share our linking workflow under publicly `https://github.com/cdli-gh/mtaac_work/tree/master/lod`.

Acknowledgements

The project Machine Translation and Automated Analysis of Cuneiform Languages is generously funded by the German Research Foundation, the Canadian Social Sciences and Humanities Research Council, and the American National Endowment for the Humanities through the T-AP Digging into Data Challenge.[40]

Our appreciation goes to Heather D. Baker and Robert K. Englund for their insights and suggestions.

[35] From the CRM documentation, the superproperty crm:P128_carries would suit too, see `http://www.cidoc-crm.org/html/5.0.4/cidoc-crm.html#P128` hence, it must be a physical thing, in our case a `E84_Information_Carrier` However, if no `?composite` is found, then a separate `crm:E34_Inscription` must be created.

[36] `http://clarkparsia.github.io/csv2rdf/`.

[37] `http://modref-labexpassespresent.huma-num.fr`.

[38] `https://collection.britishmuseum.org/sparql`.

[39] `http://www.limc-france.fr` and `http://medaillesetantiques.bnf.fr`.

[40] `https://diggingintodata.org/`.

5. Bibliographical References

Black, J. A., Cunningham, G., Eberling, G., Flückiger-Hawker, J., Robson, E., Taylor, J., and Zólyomi, G. (1998–2006). The Electronic Text Corpus of Sumerian Literature. `http://etcsl.orinst.ox.ac.uk`.

Chiarcos, C. and Fäth, C. (2017). CoNLL-RDF: Linked corpora done in an NLP-friendly way. In *International Conference on Language, Data and Knowledge*, pages 74–88. Springer.

Chiarcos, C., McCrae, J., Cimiano, P., and Fellbaum, C. (2013). Towards open data for linguistics: Linguistic linked data. In Alessandro Oltramari, et al., editors, *New Trends of Research in Ontologies and Lexical Resources: Ideas, Projects, Systems*, pages 7–25. Springer, Berlin, Heidelberg.

Chiarcos, C., et al., editors. (2012). *Linked Data in Linguistics*. Springer, Berlin, Heidelberg.

Chiarcos, C. (2012). POWLA: Modeling linguistic corpora in OWL/DL. In *Extended Semantic Web Conference*, pages 225–239. Springer.

Hayes, J. L. (2000). *A Manual of Sumerian Grammar and Texts. Second Revised and Expanded Edition*. Number 5 in ARTANES. Undena Publications, Malibu.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *International semantic web conference*, pages 98–113. Springer.

Jaworski, W. (2008a). Contents modelling of neo-Sumerian Ur III economic text corpus. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 369–376, Stroudsburg, PA. Association for Computational Linguistics.

Jaworski, W. (2008b). *Ontology-Based Knowledge Discovery from Documents in Natural Language*. Ph.D. thesis, University of Warsaw.

Koslova, N. and Damerow, P. (2003). From cuneiform archives to digital libraries: The Hermitage Museum joins the Cuneiform Digital Library Initiative. In *Proceedings of the 5th Russian Conference on Digital Libraries RCDL 2003*, pages 1–8. St.-Petersburg, Russia.

Liu, Y., Burkhart, C., Hearne, J., and Luo, L. (2015). Enhancing Sumerian lemmatization by unsupervised named-entity recognition. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1446–1451. Association for Computational Linguistics.

Nurmikko-Fuller, T. (2015). *Telling ancient tales to modern machines: ontological representation of Sumerian literary narratives*. Ph.D. thesis, University of Southampton.

Nurmikko, T. (2014). Assessing the suitability of existing OWL ontologies for the representation of narrative structures in Sumerian literature. *ISAW Papers*, 7(1):1–9.

Pagé-Perron, É., Sukhareva, M., Khait, I., and Chiarcos, C. (2017). Machine translation and automated analysis of the Sumerian language. In *Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, ACL*, pages 10–16. Association for Computational Linguistics.

Peterson, D., Palmer, M., and Wu, S. (2014). Focusing annotation for semantic role labeling. In *LREC*, pages 4467–4471.

Sanderson, R., Ciccarese, P., and Young, B. (2017). Web Annotation data model. Technical report, W3C Recommendation.

Sylak-Glassman, J. (2016). The composition and use of the universal morphological feature schema (UniMorph schema). Technical report, Department of Computer Science, Johns Hopkins University.

Tchienehom, P. (2017). ModRef Project: from creation to exploitation of CIDOC-CRM triplestores. In *The Fifth International Conference on Building and Exploring Web Based Environments (WEB 2017)*.

Zólyomi, G., Tanos, B., and Sövegjártó, S. (2008). The Electronic Text Corpus of Sumerian Royal Inscriptions. `http://oracc.museum.upenn.edu/etcsri/introduction/index.html`.

A Bird's-eye View of Language Processing Projects at the Romanian Academy

Dan Tufiș[1], Dan Cristea[2,3]
[1]Research Institute for Artificial Intelligence "Mihai Drăgănescu" of the Romanian Academy,
[2]Institute for Computer Science of the Romanian Academy
[3] "Alexandru Ioan Cuza" University of Iaşi
[1]13Calea 13 Septembrie, Bucharest, Romania
[2]2 T. Codrescu St, 700481, Iaşi, Romania
[3]11 Carol I Blv, 700506, Iaşi, Romania

[1]tufis@racai.ro, [2]dcristea@info.uaic.ro

Abstract

This article gives a general overview of five AI language-related projects that address contemporary Romanian language, in both textual and speech form, language related applications, as well as collections of old historic documents and medical archives. Namely, these projects deal with: the creation of a contemporary Romanian language text and speech corpus, resources and technologies for developing human-machine interfaces in spoken Romanian, digitization and transcription of old Romanian language documents drafted in Cyrillic into the modern Latin alphabet, digitization of the oldest archive of diabetes medical records and dialogue systems with personal robots and autonomous vehicles. The technologies involved for attaining the objectives range from image processing (intelligent character recognition for hand-writing and old Romanian documents) to natural language and speech processing techniques (corpus compiling and documentation, multi-level processing, transliteration of different old scripts into modern Romanian, command language processing, various levels of speech-text alignments, ASR, TTS, keyword spotting, etc.). Some of these projects are approaching the end, others have just started and others are about to start. All the reported projects are national ones, less documented than the international projects we are/were engaged in, and involve large teams of experts and master/PhD students from computer science, mathematics, linguistics, philology and library sciences.

Keywords: contemporary language reference corpus, speech corpus, natural language dialogue systems, intelligent character recognition, medical records archive

1. Introduction

Language technology is a discipline needed in more and more areas. The computational linguistics traditional domain has been extended towards dealing with vast and diversified quantities of linguistic data, both written and spoken. This extension requires putting together expertise from natural language processing (NLP), Internet-of-Things (IoT), robotics, image processing, cognitive computing, High Performance Computing (HPC), semantic web, social media and many others. In this article we will give a brief account on several projects at the Romanian Academy, their prospects and outline the challenges posed by each of them. The common characteristic of these projects, apart from being all conducted by teams affiliated to different institutes of the Romanian Academy, is the use of a solid background of language resources, tools and models, acquired along the last two decades and permanently enhanced. As such, we argue that a constant accumulation of human expertise, language resources and software utilities can bring about a level that allows for the development of extremely complex projects. Without this know-how, data and processing platforms, these enterprises would have not been feasible.

2. CoRoLa

The CoRoLa project is a priority program of the Romanian Academy and is developed by two institutes of the Romanian Academy: "Mihai Drăgănescu" Research Institute on Artificial Intelligence in Bucharest (RACAI) and the Institute of Computer Science of the Iaşi branch of the Romanian Academy (IIT), and voluntarily contributed by linguists from "Al. Philippide" Institute of Romanian Philology in Iaşi and many master and PhD students from "Alexandru Ioan Cuza" University of Iași, University "Politehnica" of Bucharest and the Bucharest University.

The project is concerned with the construction and maintenance of the reference corpus of the contemporary written and spoken Romanian as well as with the infrastructure to openly use it.

The corpus covers 5 large domains (arts & culture, society, science, nature and *others*[1]) which are further refined into 71 sub-domains - see (Barbu MItitelu *et al.*, 2018) for more details.

Currently, CoRoLa contains over 1,200,000,000 tokens of written texts and almost 152 hours of pre-processed speech recordings and may be queried at *corola.racai.ro* via three web interfaces (see Figure 1, 2 and 3 in the annex): KorAP (the entire text corpus is searchable), NLP-CQP (only a selection of about 35% from the entire text corpus is searchable) and OCQP (the oral corpus).

All the CoRoLa data is IPR cleared based on individual written agreements concluded with the data providers (IPR holders). The written texts are fully pre-processed (sentence split, tokenized, lemmatized and morpho-syntactically tagged). A small part of the corpus represents a core of a Reference Treebank dependency parsed (RoRefTrees) (approx. 10,000 sentences) and hand validated (Barbu-Mititelu *et al.*, 2016). In (Barbu-Mititelu *et al.*, 2016) there is a description of the previous RACAI

[1] "others" is a category for all documents that could not be definitely classified into the named categories.

dependency parser, based on MaltParser (Nivre *et al.*, 2007), trained on RoRefTrees. Recently, a new parser, SSPR (Semantics-driven Syntactic Parser (for Romanian)) has been developed. It is a feed-forward neural network ensemble parser developed for Romanian. The network architecture contains two hidden layers, the first with 100 neurons, and the second with 10 neurons. The activation function for both hidden layers is *tanh*. The parser is actually a classifier that combines the parsing decisions of three (the number is arbitrary) different parsers, namely MaltParser, RGB (Lei et al., 2014) and MATE (Bohnet, 2010) and using information from additional lexical, morpho-syntactic and semantic features builds the most likely dependency parse (see (Ion et al, 2018) for further details). SSPR performs better than any of the three individual parsers, with average scores of 83% LAS (labeled attachment score) and 88.44% UAS (unlabeled attachment score). Thus, SSPR is in the same accuracy class with the top 5 performers at the CONLL 2017 dependency parsing shared task for Romanian. This statement is backed up by the fact that the training, development and test data used for comparison were identical (RoRefTrees).

SSPR will be used to parse the entire content of the textual part of CoRoLa. The current textual content (which will continue to grow) is, quantitatively, well above the project initial promise (500 million tokens).

Recently, we constructed various sets of word embeddings from the CoRoLa corpus (Păiș and Tufiș, 2018), using different vector sizes (100, 200, 300, 400, 500 and 600) and frequency thresholds (1, 5, 10, 20, 50). The pre-trained CoRoLa word embeddings, analogy application and data sets may be downloaded from http://89.38.230.23/word_embeddings/. The user may play around with associations such as "însurat" (En. *married*, term used only for men) - "bărbat" (En. *man*) + 'femeie" (En. *woman*) and get "măritată" (En. *married*, term used only for women).

The situation is not equally good for the pre-processed spoken data, although the quantity of the collected raw speech data exceeds the promised 300 hours. The reason is that several sets of spoken material are not transcribed and some are not IPR-cleared yet (mainly children stories). The transcribed spoken data is pre-processed (sentence split, tokenized, lemmatized and morpho-syntactically tagged) and aligned with the speech data at sentence, word and phoneme levels. The interface allows a user to search for a word and hear its pronunciation or to listen to the entire sentence where it occurs.

The CoRoLa language data is accompanied by detailed meta-data information (CMDI compliant). The initial corpus management system was IMS Open Corpus Workbench, an open source medium (CWB, http://cwb.sourceforge.net/) but lately, based on a collaboration with IDS-Mannheim, we commuted on their KorAP (Banski et al., 2014) environment which appears to be rather insensitive (response time-wise) to the volume of the corpus. For the time being, KorAP does not offer facilities for querying the speech data, so we continue to use our own interface. However, as it will become available, all CoRoLa data (text and speech) will be searchable in a single unified environment (KorAP).

This collaboration, financed by the Humboldt foundation, joints forces from IDS Mannheim, University of Bucharest, IIT-Iași and RACAI-Bucharest in the DRuKoLA project (Cosma et al., 2016). There are further developments on this project, hopefully towards a pan-European initiative – EuReCo (Kupietz *et al.*, 2017*).*

The first phase of CoRoLa project (2014 – 2017) ended with an official public launching with presentations and demos, enjoying a high interest from the academic community, as well as from the media. The Romanian Academy decided that the CoRoLa project will be further continued and enhanced (with the same status of a priority program) within the next phase (2018-2020). The enhancement of the CoRoLa corpus will be pursued in parallel with the DRuKoLA project and, possibly, with EuReCo partnership.

3. Human-Machine Interfaces in Spoken Romanian

To compensate the shortage of the IPR-cleared speech data, we launched this year a national project involving the major local players in speech processing: University POLITEHNICA of Bucharest, Technical University of Cluj-Napoca, "A.I. Cuza" University of Iaşi and RACAI-Bucharest. All partners have developed small but clean corpora of transcribed pre-processed speech data (few hundred hours of speech recordings) and one objective of this project is to harmonize these speech data collections into a single larger common speech corpus that will contain at least 500 hours of fully processed speech recordings and which will be documented with the appropriate unified meta-data. The recordings represent, in general, read texts, but also multiple participants interviews and a few recorded hours or spontaneous speech. This new speech corpus will be included into CoRoLa. Other objectives are the development of an accurate, speaker independent, Automatic Speech Recognition (ASR) system (which we plan to use for transcribing the bulk of not yet transcribed records) and a high quality Text to Speech (TTS) system capable to generate expressive (emotional) speech, based on subjectivity marked-up text (Tufiș and Ștefănescu, 2012). This project aims at producing portable and interoperable much better ASR and TTS systems than the existing ones (Cucu et al. 2014), (Stan et all. 2013), (Boroș and Dumitrescu, 2015). The ASR and TTS systems will be integrated into a platform for trainable, generic and situated (script-based) dialogues.

This work is supported by a grant of the Ministry of Research and Innovation CCCDI-UEFISCDI, project cod PN-III-P1-1.2-PCCDI-2017-0818 within PNCDI III.

In parallel with this project we are working, together with experts from the Military Academy, on an experimental-demonstrative project (to be finished in 2018) for real time detection of keywords (Romanian language) in telephone conversations (*http://heimdall.racai.ro/*).

Keyword Spotting (KWS) is a technology that enables the detection of word occurrences within spoken language

from audio or audio/video streams. While it shares many aspects in terms of audio processing with speech recognition and transcription, it is fundamentally different from any restricted (grammar based) or language model (LM) based speech transcription system. The main difference between these technologies is that KWS does not require any language modeling and, provided with automatic grapheme-to-phoneme (G2P) capabilities, KWS is not in any way affected by out-of-vocabulary (OOV) words.

The acoustic models will be constructed based on the oral corpus in CoRoLa and an additional corpus of telephone recordings developed within this project. The data is annotated for speaker's gender (male-female), age interval (18-35/35-60) and regional accent (Bucovina, Moldova, Oltenia, Muntenia).

Though KWS is developed as a system for security applications, it is not restricted to this field. With minor adjustments the system can be used for call-center monitoring, business analytics, etc.

The experimental model will be a KWS software integrated with a VoIP laboratory network. The system will be able to detect keywords in live conversations in real-time. The proposed system starts from a proof of concept KWS developed by RACAI, and improves it on the following areas: acoustic models, enlarged corpora for training the acoustic models and real time performance.

This work is supported by a grant of the Ministry of Research and Innovation CCCDI-UEFISCDI, project code PN-III-P2-2.1-PED-2016-1within PNCDI III.

4. Speaking with Your Personal Robot

This is a complex project (ROBIN), user centered, aiming to develop software and services for interaction with assistive robots and autonomous vehicles. The project consortium, besides RACAI, includes experts in robotics and digital signal processing from the Polytechnics University of Bucharest and the Institute of Mathematics of the Romanian Academy, well known specialists in IoT technology from the University of Bucharest and Technical University of Cluj-Napoca and experts in designing autonomous vehicles from University "Dunărea de Jos" of Galați. ROBIN is an ensemble of five collaborating sub-projects, combining advanced techniques and technologies from AI, human-robot interaction, pervasive and Cloud computing, each of them having specific objectives. Language interaction is one of the challenges, as no robots can be talked to now in Romanian, neither any interaction, with appropriately equipped cars, may be conducted in Romanian.

The human-robot dialogues are situated ones (close-world based) and our involvement will be supported by previous results in the recently finished project ANVSIB (Boroș and Dumitrescu, 2017) on intelligent buildings equipped with IoT devices controlling the home appliances (TV set, heating system, drapes and interior blinds, main locks, etc.).

The situated dialogue component of ANVSIB project, enhanced with the new facilities to be developed in the previously mentioned generic dialogue platform will be adapted for this project. However, the major challenge is that the spoken commands will not be sent via a smart phone (as in ANVSIB project), but addressed directly to the robot, thus being affected by more surrounding noise.

The sub-project concerned with the development of intelligent software modules for hands-off driving, automated driving and a prototype of electric semi-autonomous vehicle benefits of a prototype DACIA car offered by an industrial partner (PRIME Motors Industry). Within this sub-project, our team will be responsible for the development of a module for voice interaction (Romanian language) between the driver and the car.

This work is supported by a 33 month grant of the Ministry of Research and Innovation PCCDI-UEFISCDI, project code PN-III-P1-1.2-PCCDI-2017-734 within PNCDI III.

5. Intelligent Character Recognition in Medical Records Processing

Automatic handwriting recognition is a major preoccupation for commercial companies and there are already several products available in this area: Quick Draw With Google[2], an interactive drawing recognition tool recently launched by Google, Google Handwriting Input in 82 Languages[3] and Microsoft Handwriting Input for Windows[4] to name just a few of them. These OCR systems are in general agnostic on the document structure and content they process.

We faced the need for Intelligent Character Recognition (ICR) a few years ago, but recently a precise requirement for it emerged: the *National Institute of Diabetes, Nutrition and Metabolic Diseases "Prof. N.C. Paulescu"* (NIDNMD) is the owner of the world oldest medical records in diabetes, still active in a city (Bucharest), from 1941 until today (Ionescu-Tîrgoviște, 2001). The archive is paper-based, containing more than 220,000 medical records. The documents are structured, according to three types of forms, but the filling in of the blanks is handwritten. Most fields which medical experts are interested in are numerical ones (i.e. age, height, weight, diabetes type, blood glucose, etc.). Knowing what to expect from the document structure makes the handwritten recognition task a little bit easier to accomplish and enables automatic validation of data consistency.

Thus, we engaged, together with the medical experts from NIDNMD, into transforming the paper-based archive into a digital searchable archive. This project, called InsyderPal, is still under the last phase of evaluation, within the national program PNIII (*Fundamental and Frontier Research-PN-III-P4-IDPCCF-2016*), meant to support multi-, inter- and trans-disciplinary research.

[2] https://quickdraw.withgoogle.com/

[3] https://research.googleblog.com/2015/04/google-handwriting-input-in-82.html

[4] https://www.howtogeek.com/297443/how-to-use-handwriting-input-on-windows-10/

Provided it is funded, the project is expected to have also a standardization effect for the digital recording of the diabetes data in the specialized medical institutions, all over the country.

RACAI has previously worked with sequence based processing algorithms. The ICR tool, recently implemented at RACAI and which will be further developed in this project, is based on recurrent neural networks with a connectionist temporal classification layer (Graves and Schmidhuber, 2009). Our proposed ICR system architecture is composed of 3 main tools, two already prototyped: (a) a segmentation tool for text lines extraction using vertical projection-profile (Likforman-Sulem et al., 2007) (b) a classification and document structure understanding (DSU) tool, and (c) the ICR recognition tool itself. While the first (a) and later mentioned (c) tools already exist, the document understanding tool will be later developed in the project. Its primary purpose is to support the process of automatic classification and detection structures within documents and to enable the creation of data sanity checks (e.g. the overall cholesterol must be equal with the sum of HDL and LDL cholesterols). Depending on the output of the DSU system, ICR will be performed either by CNNs whenever the input is numerical and easily separable or by using bidirectional LSTMs when we are dealing with continuous handwritten data.

To train the ICR tool, a collection of learning examples has to be compiled. An example is a triple formed by an image segment (extracted by means of the segmentation tool mentioned above), a label identifying the region in the scanned image of a record from where the segment was extracted (according to the templates mentioned at step b) above) and the correct result of its recognition (the truth). We estimate that for a high precision ICR process, the training set should include around 100,000 examples. The training data creation is an interactive process, requiring a lot of human involvement (especially in defining the truth for an image segment, but not only).

The project also has as an objective the implementation of an application allowing the professionals to update and exploit the digitalized archive as well as to extract relevant information (in Romanian language) from specialized literature.

There are several language resources backing-up the information extraction, including a heavily annotated and carefully validated domain specific BioRo corpus (currently 561,978 sentences, with almost 10 million tokens, MSD tagged, lemmatized, NERC marked-up), a thematic term dictionary (about 8,000 entries) and a collection of pre-processing tools (tokenizer, tagger, lemmatizer, NER) trained for this specific medical language (Mitrofan and Tufiș, 2016), (Mitrofan, 2017), (Mitrofan and Tufiș, 2018). All these resources will be significantly extended, both quantitatively and qualitatively.

6. Turning Old Romanian Documents Written with Cyrillic Alphabets into a Latin Character Set Electronic Corpus

An even more challenging project requiring ICR is "CyRo", a typical Digital Humanities project, the objective of which is to create a prototype processing chain, turning old Romanian documents written with Cyrillic characters into editable documents and then transliterating them into Latin alphabet. This is also a project proposal, still under the last phase of evaluation, within the national program *Fundamental and Frontier Research-PN-III-P4-IDPCCF-2016*.

The consortium of this project is a large one with experts in language, image processing and deep learning technology, from the Institute of Computer Science of the Romanian Academy in Iași, "A.I. Cuza" University of Iași, the University of Bucharest and the Research Institute for Artificial Intelligence of the Romanian Academy in Bucharest, experts in old and rare manuscripts from the Library of the Romanian Academy and experts in diachronic Romanian and Slavonic languages from "Al. Philippide" Philology Institute of the Romanian Academy in Iași, the Faculty of Letters of the University of Bucharest and the Faculty of Letters of "A. I. Cuza" University of Iași.

While the Cyrillic into Latin text transliteration is an almost error-free process (Cojocaru *et al.*, 2016), the main difficulties are in transforming scanned old documents into editable texts. Cojocaru and her colleagues (2016) reported very good results on implementing an integrated processing flow meant to OCR and transliterate in the Latin alphabet recent Romanian prints (1951-1989) written with Cyrillic characters in the former Moldavian Soviet Socialist Republic.

Yet, the CyRo project faces more difficulties: three different Cyrillic alphabets used during the targeted period, plus two types of handwritten documents: the so-called semi-uncial handwriting and regular cursive handwriting. In semi-uncial handwriting each drawn character imitates a printed character and is separated from the neighboring ones. Therefore this type of handwriting is supposed to be easier to process than cursive writing, which includes ligatures, overwriting, the use of multiple phonetic values for some letters, where the designation of proper names in many cases appears with initial lowercase, where abbreviations and diacritics without phonetic values in Romanian are frequently used, where there are combinations between abbreviations and overwriting, inconstant denotation of numerals, and sometimes spaces separating words are missing (*scriptio continua*).

Another important group of difficulties (Cristea et al. 2012), (Simionescu et al., 2012) is triggered by the diachronic changes of the Romanian language (phonetic, morphological, lexical, syntactic). Although, for realistic promises, we explicitly address only documents printed and with semi-uncial handwriting, we will also aim to do extensive experiments on cursive handwriting, but offering also the expert the possibility to make corrections on the final interpreted document.

The ICR technology created in CyRo will facilitate the development of a quasi-exhaustive diachronic corpus of the Romanian language, which will be used in applications and studies supposed to have a high impact on the study of the Romanian language, as well as, more generally, on the family of Romance languages. Moreover, the new methods and tools laid down in CyRo are expected to inspire a whole generation of interpretative instruments adapted to Slavic languages, for the automatic interpretation of old documents written in Cyrillic.

7. Conclusions

Some of the projects, approaching the end, mentioned in this article are described in larger details in dedicated papers (Barbu Mititelu *et al.*, 2018; Mitrofan and Tufiș, 2018). We tried to offer a global view of the relevant projects we and our colleagues carry on in the area of AI, NLP and DH domains. Unlike many of our previous projects, the current ones are nation-wide, bringing together more R&D groups and aiming at more ambitious targets. Some results will be transferred to the industry for further development and industrialization and their distribution will be restricted. However, most outcomes of the mentioned projects will be freely accessible to the interested public. Several datasets (lexicons, treebank, text and speech corpora, word embeddings) created for training different machine learning modules and tools (TTL, SSPR, MLPLA) are already available and others (ASR, TTS, KWS, ICR) will be made public (ensuring anonymisation, whenever necessary, of the personal information, e.g. in the processed medical records and the speech recordings), as soon as they are ready.

8. Acknowledgements

The people who are involved in the above mentioned projects are too many to be listed here, but we want to express our gratitude to all of them (data providers, researchers, PhD and MSc students), for their enthusiastic past, present and future engagement and hard work.

We also thank to the funding bodies that made these endeavors possible: Romanian Academy, Romanian Ministry of Education and Research, UEFISCDI (Executive Agency for Higher Education, Research, Development and Innovation Funding), Humboldt Foundation and Institute for German Language IDS-Mannheim.

9. Bibliographical References

Bański, P., Diewald, N., Hanl, M., Kupietz, M., Witt, A. (2014): Access Control by Query Rewriting. The Case of KorAP. In *Proceedings of the Ninth Conference on International Language Resources and Evaluation (LREC'14)*. Reykjavik: European Language Resources Association (ELRA), pp. 3817-3822.

Barbu Mititelu, V., Ion, R., Simionescu, R., Irimia, E., Perez C.-A. (2016) The Romanian Treebank Annotated According to Universal Dependencies. In Proceedings of The Tenth International Conference on Natural Language Processing (HrTAL2016), Dubrovnik, Croatia, 29 September – 1 October 2016

Barbu Mititelu, V., Tufiș, D., Irimia, E. (2018): The Reference Corpus of the Contemporary Romanian Language (CoRoLa). Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaky, Japan.

Bohnet, B. 2010. Very High Accuracy and Fast Dependency Parsing is not a Contradiction. The 23rd International Conference on Computational Linguistics (COLING 2010), Beijing, China.

Boroș, T. and Dumitrescu, S. D. (2017). A "small-data"-driven approach to dialogue systems for natural language human computer interaction. In *The 9th Conference on Speech Technology and Human-Computer Dialogue (SPED 2017)*. Bucharest, Romania

Boroș, T., and Dumitrescu, S. D. (2015). Robust deep-learning models for text-to-speech synthesis support on embedded devices. In *Proceedings of the 7th International Conference on Management of Computational and Collective Intelligence in Digital EcoSystems*, pp. 98-102, ACM.

Boroș, T., Ion, R., & Tufiș, D. (2013). Large tagset labelling using Feed Forward Neural Networks. Case study on Romanian Language. In *Proceedings of ACL*, Sofia, pp. 692-700.

Cojocaru, S., Colesnicov, A., Malahov, L. (2017). Digitization of Old Romanian Texts Printed in the Cyrillic Script. In *Proceedings of DATECH 2017*, 1-2 June, Göttingen

Cosma, R., Cristea, D., Kupietz, M., Tufiș, D., Witt, A. (2016). DRuKoLA – Towards Contrastive German-Romanian Research based on Comparable Corpora. In Bański, P., Barbaresi, A., Biber, H., Breiteneder, E., Clematide, S., Kupietz, M., Lüngen, H., and A. Witt (eds.) The 4th Workshop on Challenges in the Management of Large Corpora. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož / Paris: ELRA, pp. 28-32.

Cristea, D., Simionescu, R., & Haja, G. (2012). Reconstructing the Diachronic Morphology of Romanian from Dictionary Citations. In *LREC* (pp. 923-927).

Cucu, H., Buzo, A., Besacier, L., & Burileanu, C. (2014). SMT-based ASR domain adaptation methods for under-resourced languages: Application to Romanian. *Speech Communication*, *56*, 195-212.

Dumitrescu, Ș. D., Boroș, T. and Tufiș, D. (2017). RACAI's natural language processing pipeline for Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. ACL Vancouver, Canada, pp. 174-181.

Graves, A., & Schmidhuber, J. (2009). Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in neural information processing systems* (pp. 545-552).

Ion, R., Irimia, E., Barbu Mititelu, Verginica. (2018). Ensemble Romanian Dependency Parsing with Neural Networks. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaky, Japan.

Ionescu-Tîrgoviște, C., *Diabetul in Romania* (*Analele Academice Diabetologice)*,(2001) Briliant, Bucuresti, 2001.

Kupietz, M., Andreas Witt, A., Bański, P., Tufiș, D.,

Cristea, D., Váradi, T. (2017). EuReCo - Joining Forces for a European Reference Corpus as a sustainable base for cross-linguistic research, In *Proceedings of the 5th* CMLC, Birmingham, pp.15-20.

Lei, T., Xin, Y., Zhang, Y., Barzilay, R., Jaakkola, T. (2014). Low-Rank Tensors for Scoring Dependency Structures. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (ACL 2014), June 23-25, Baltimore, Maryland, USA.

Likforman-Sulem, L., Zahour, A., Taconet, B. (2007). Text line segmentation of historical documents: a survey. In *International Journal of Document Analysis and Recognition*, Vol 9(2-4), pp.123-138.

Mitrofan, M. (2017). Bootstrapping a Romanian Corpus for Medical Named Entity Recognition. In *Proceedings of RANLP,* Varna, Bulgaria, pp. 501–509.

Mitrofan M., and Tufiș, M.(2016). Building and evaluating the Romanian medical corpus. In Proceedings of the 12th International Conference ”Linguistic Resources and tools for processing the Romanian language”, pp. 29–36.

Mitrofan M., and Tufiș, D. (2018). BioRo: The Biomedical Corpus for the Romanian Language. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaky, Japan.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., Marsi, E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering* 13(2), pp. 95-135.

Păiș, V., and Tufiș, D. (2018). Computing Distributed Representations of Words using the CoRoLa Corpus. Proceedings of the Romanian Academy, Series A, vol 19. No.1/2018

Simionescu, R., Cristea, D., & Haja, G. (2012). Inferring diachronic morphology using the Romanian Thesaurus Dictionary. In *Proceedings of the 8th International Conference “CONSILR”, April, Bucharest* (pp. 85-92).

Stan, A., Watts, O., Mamiya, Y., Giurgiu, M., Clark, R. A., Yamagishi, J., & King, S. (2013, August). TUNDRA: a multilingual corpus of found data for TTS research created with light supervision. In *INTERSPEECH* (pp. 2331-2335).

Tufiș, D., & Ștefănescu, D. (2012). Experiments with a differential semantics annotation for WordNet 3.0. *Decision Support Systems*, *53*(4), 695-703.

Tufiș D., Barbu Mititelu V., Irimia E., Dumitrescu Ș. D., Boroș T. (2016). The IPR-cleared Corpus of Contemporary Written and Spoken Romanian Language. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (LREC 2016), European Language Resources Association (ELRA), ISBN: 978-2-9517408-9-1, pp. 2516-2521, Portorož, Slovenia

ANNEXES

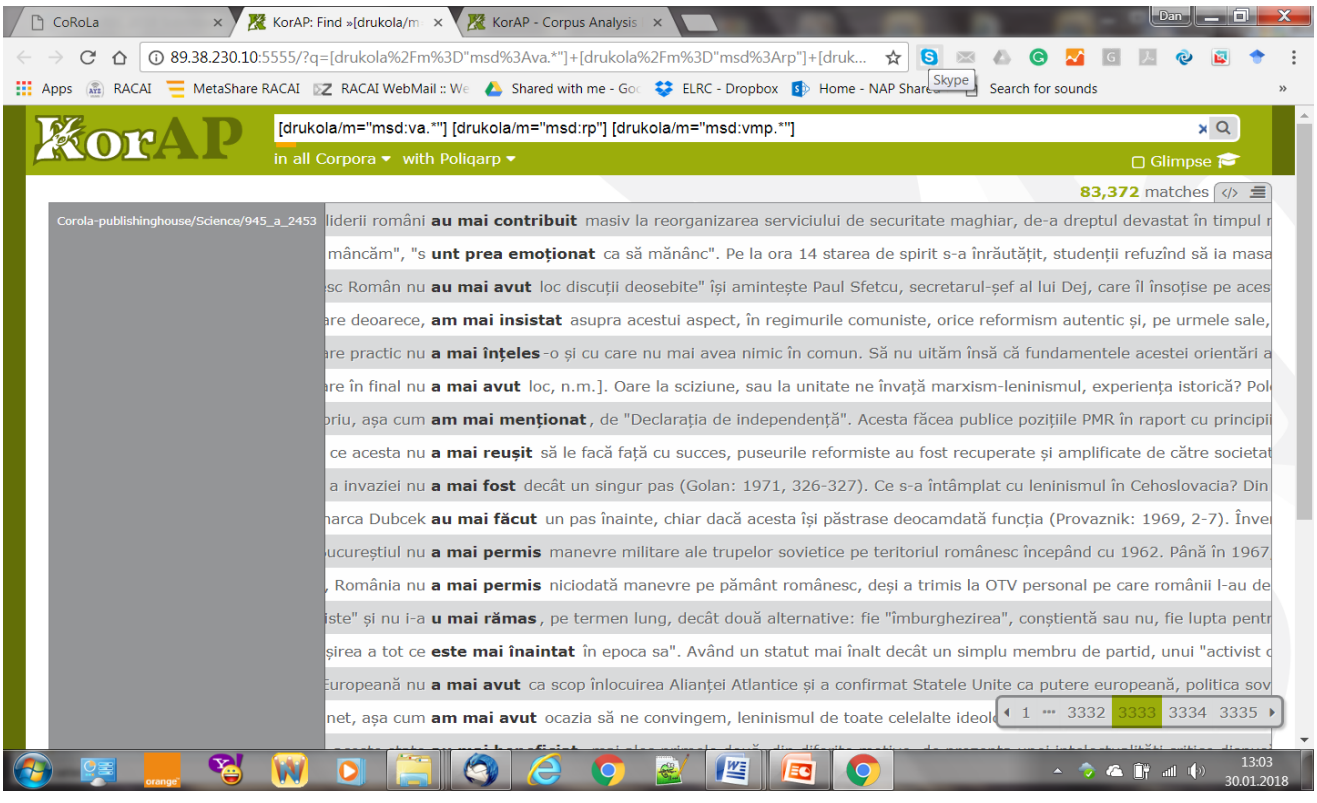


Figure 1: KoRaP Search Interface for CoRoLa
Query (Poliqarp): an auxiliary verb followed by an adverb followed by a participle

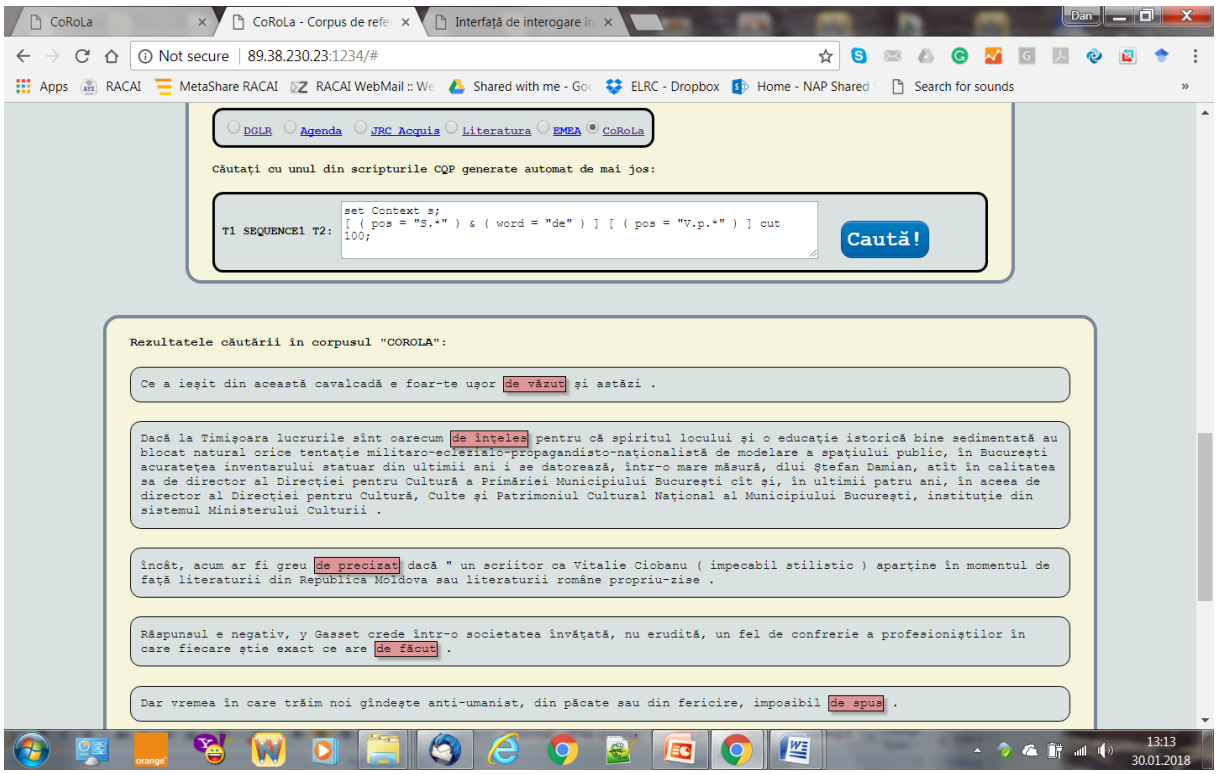


Figure 2: NLP2CQP Search Interface for CoRoLa
Query (NL language): „100 de fraze în care prepoziția "de" este urmată imediat de un verb la participiu = 100 sentences in which the preposition "de" is immediately followed by a participle verb"

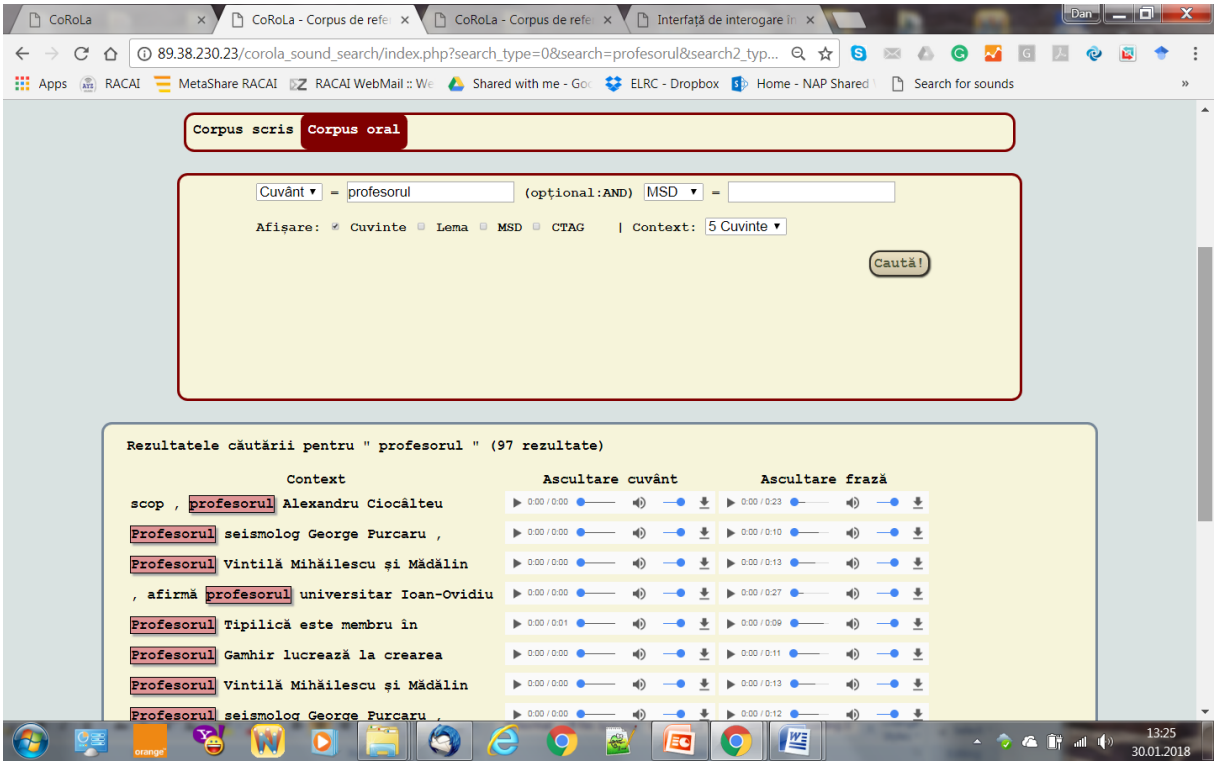


Figure 3: OCQP Search Interface for CoRoLa
Query: oral fragments containing the wordform „profesorul" (the professor)

PMKI: an European Commission action for the interoperability, maintainability and sustainability of Language Resources

Peter Schmitz, Enrico Francesconi*, Najeh Hajlaoui, Brahim Batouche
Publications Office of the European Union, Luxembourg
* Publications Office of the European Union, Luxembourg and ITTIG-CNR, Florence, Italy
{Peter.Schmitz, Enrico.Francesconi}@publications.europa.eu
{Najeh.Hajlaoui, Brahim.Batouche}@ext.publications.europa.eu

Abstract

The paper presents the Public Multilingual Knowledge Management Infrastructure (PMKI) action launched by the European Commission (EC) to promote the Digital Single Market in the European Union (EU). PMKI aims to share maintainable and sustainable Language Resources making them interoperable in order to support language technology industry, and public administrations, with multilingual tools able to improve cross border accessibility of digital services. The paper focuses on the main feature (interoperability) that represents the specificity of PMKI platform distinguishing it from other existing frameworks. In particular it aims to create a set of tools and facilities, based on Semantic Web technologies, to establish semantic interoperability between multilingual lexicons. Such task requires to harmonize in general multilingual language resources using standardised representation with respect to a defined core data model under an adequate architecture. A comparative study among the main data models for representing lexicons and recommendations for the PMKI service was required as well. Moreover, synergies with other programs of the EU institutions, as far as systems interoperability and Machine Translation (MT) solutions, are foreseen. For instance some interactions are foreseen between PMKI and MT service provided by the EC but also with other NLP applications.

Keywords: Language Resources, Interoperability, Maintainability, Sustainability.

1. Introduction

Language barriers in the EU make the European market fragmented and decrease its economic potential. The EU institutions aim to overcome language obstacles and increase cross-border e-commerce by building open multilingual tools and features free of charge. For this reason the European Commission, through the ISA[2] program[1], launched a pilot project on creating a public multilingual knowledge management infrastructure aimed to support e-commerce solutions in a multilingual environment. By creating interoperable multilingual classifications and terminologies, easily reusable by small and medium-sized enterprises (SMEs) and public administrations, the project aims to support services like machine translation, localisation and multilingual search. SMEs are currently at a disadvantage compared to big companies due to the high cost of providing multilingual services. In this respect PMKI aims to create a set of tools and facilities, based on semantic Web technologies, aimed to support enterprises, in particular the language technology industry, with multilingual tools in order to improve cross border accessibility of digital services and e-commerce solutions.

In practical terms, overcoming language barriers on the Web means creating multilingual lexicons (as vocabularies, thesauri, taxonomies, semantic networks), establishing links between concepts, as well as using them to support the accessibility of services and goods offered through the Internet.

This paper is organized as follows: in Section 2 the main objectives of PMKI are discussed; in Section 3 the foreseen interoperability solutions are illustrated; finally in Section 4 some conclusions are reported.

[1] ISA[2]: Interoperability solutions for public administrations, businesses and citizens (https://ec.europa.eu/isa2/home_en)

2. PMKI

The objective of PMKI is to implement a proof-of-concept infrastructure to expose and to harmonize internal (European Union institutional) and external multilingual lexicons aligning them in order to facilitate interoperability. Additionally the project aims to create a governance structure to extend systematically the infrastructure by the integration of supplementary public multilingual taxonomies/terminologies.

PMKI is a pilot project to check the feasibility and to prepare a road map to convert such proof of concept into a public service.

The need to have a public and multilingual platform that can play the role of a hub to collect and to share language resources in standardised formats is essential to guarantee semantic interoperability of digital services. For instance, such platform is missing in CEF.AT[2], while it would provide an advantage for the development of machine translation systems. In particular it can provide alignments of domain specific terminologies for developing specific-domain translation systems (tender terminology, medical terminology, etc.).

A platform like PMKI may represent a one-stop-shop harmonized multilingual lexicons repository at European level.

Complementary to the European Language Resource Coordination (ELRC[3]) action, which aims at identifying

[2] https://ec.europa.eu/digital-single-market/en/automated-translation

[3] ELRC: the European Language Resource Coordination action launched by the European Commission as part of the CEF.AT platform activities, to identify and gather language data across all 30 European countries participating in the CEF programme. This will be followed up by actions concerning the setting up of a repository of language resources for CEF AT and further

and gathering language and translation data, PMKI platform aims first to harmonize multilingual language resources making them interoperable, then to integrate supplementary public multilingual taxonomies/terminologies in a standardized representation. That is why we need first to define 1) a sophisticated standard representation that will be used with respect to 2) a defined core data model (in case with extensions) under 3) an adequate architecture.
These three requirements were respectively analysed and detailed in three first analysis phases of the project:

- Analysis of existing relevant standards for the representation of lexicons that will be made available on the PMKI platform;
- PMKI core data model and extensions (based on the standard representation that was recommended as result of the previous analysis);
- Analysis of available platforms for managing lexicons.

Interoperability is one of the main features of the PMKI platform; such platform will provide support to develop multilingual tools such as machine translation, localization, search etc. For instance for the machine translation tool, interoperable translation data is a factor of success to improve the quality mainly for under-resourced languages.

3. Interoperability

Interoperability between language resources is essential in order to facilitate the access to European service by overcoming the language barriers. In PMKI we started to apply such interoperability by aligning language resources.

3.1 Experimental approach

The alignment idea is based on the work of Bartoloni and Francesconi (2010). They proposed a framework based on the definition of an isomorphism between linguistic resources alignment problem and the Information Retrieval (IR) problem.

To have a linguistic resource amenable for computation, we first organize the linguistic resource according to a specific standard. Then in order to create a linguistic resource alignment system we follow the following processing steps that (Figure 1):



Figure 1: Steps of alignment with supervised learning.

1. Give a representation of the semantics of linguistic resources amenable for computation.
2. Compute similarities between source and target resources to create dataset for learning.
3. Validate the matching of linguistic resource with user validator to build supervised learning dataset.
4. Launch the learning process to create an alignment system.

As EuroVoc[4] is the kernel of PMKI platform, we use it as source dataset to be aligned with new target dataset. All resources are indexed by their URI. The storage of new resource follows SKOS, LEMON, or ONTOLEX standards.
The diversification of linguistic resources is a necessary condition to have a good coverage of the learning phase, able to deal with different written/pronounced word and multi-word expressions. This diversification problem is one of the sensitive aspects in PMKI.

We started collecting linguistic datasets, each one represented as a graph of triples. Each dataset is represented as one of the three standards mentioned above, each standard has some useful properties for alignment. We selected the value of those properties using SPARQL query and through the SPARQL endpoint of the related triple store. Therefore, those selecting property values are used to instantiate the resources (according to Section 3.3). We used information retrieval tool (Lucene[5] as java API) to create and to represent the index of terms for our alignment approach.

3.2 Formalization of alignments

Our matching methods are string-based, language-based, and constraint-based techniques. From these three techniques, the string-based matching technique allows to represent the input format for the alignment. The language-based technique is useful to extract useful words for the string-based technique. Therefore we use an hybridization of matching techniques where language-based matching is used before the string-based one.

3.3 Linguistic resource formalization

Mapping between linguistic resources aims at matching terms semantics rather than searching for lexical equivalences. In traditional linguistic resources descriptors and non-descriptors are represented by different terms (such as `lemon:writtenRep`, `lemon:usage`, `skos:prefLabel` and `skos:altLabel`) expressing the same meaning. More precisely, each meaning is expressed by one or more terms in the same language (e.g. *pollution, contamination, discharge of pollutants*), as well as in different languages. Moreover each term can have more than one sense, i.e. it can express more than one concept. Therefore to effectively map linguistic resource, term (simple or complex) semantics has to be captured and represented.

In Information retrieval (IR) a query is usually constructed as a context (set of keywords) able to better represent the semantics of the users' information needs. Similarly in linguistic resource mapping (LM) the semantics of a term (simple or complex) is conveyed not

---

data collection and awareness actions in the context of calls for tenders and calls for proposals for which the selection procedure is still ongoing. More information can be found here: http://www.lr-coordination.eu/

[4] http://eurovoc.europa.eu/4067
[5] https://lucene.apache.org/

only by the keywords, but also by the context in which the term is used as well as by the relations with other terms. In the LM problem, we aimed at identifying logical views and related framework for linguistic resource concepts representation able to better capture the semantics of terms in source and target linguistic resource, as well as to measure their conceptual similarity.

Bartoloni and Francesconi (2010) proposed to represent the semantics of a linguistic resource by a vector *d* of binary entries composed by the term itself, relevant terms in its definition, in the alternative labels, as well as terms of directly related linguistic resource (`skos:broader`, `skos:narrower`, `skos:related` concepts).

A vocabulary of normalized terms from target linguistic resource is constructed, where "normalization" in this context means string pre-processing, in particular word stemming and stop-words elimination. Being *T* the dimension of such vocabulary, both source and target linguistic resource concepts are represented in a vector space of *T*-dimension ($d=[x_1,x_2,...x_T]$); the entry x_i gives information on the presence/absence of the corresponding x^{th} vocabulary term among the terms characterizing the linguistic resource *d*. In (Figure 2) a binary vector representation of EuroVoc concept is sketched.

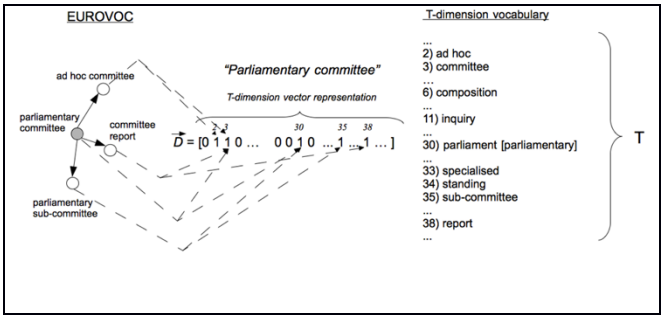


Figure 2: T-dimension vectorial representation of a thesaural description *d* ***d***

3.4 Formal characteristic of linguistic resource mapping

Such kind of characteristics allow us to propose a definition of the schema-based Linguistic-resource Mapping (LM) problem as an Information Retrieval (IR) problem: the aim is to find concepts in a target linguistic resource, which match with the semantics of a given source concept. The isomorphism between LM and IR can be established once we consider a source language resource as a query of the IR problem, and a target language resource as a document (Baeza-Yates, 1999).

Therefore the LM problem can be viewed and formalized, as an IR problem composed of a 4-uple $LM = [D, Q, F, R(q,d)]$ $LM=D,Q,F,Rq,d$ where:

1. D $D$is a set of the possible representations (logical views) of a concept in the target linguistic resource (i.e. a document to be retrieved in the IR problem);
2. Q $Q$is the set of the possible representations (logical views) of a concept in the source linguistic resource (a query in the IRIR problem);
3. F $F$is the framework of resource representation in source and target linguistic resource (in our case the vectorial space of T-dimension endowed with metrics);
4. $R(q,d)$ Rq,d is a ranking function, which associates a real number with (q,d) where $q \in Q$ and $d \in D$ giving an order of relevance to the resource in the target concept with respect to the one of the source.

In this framework the implementation of a linguistic resource mapping procedure is represented by the instantiation of the four previous components.

3.5 Evaluation of similarity

The similarity is inversely proportional to the distance between binary vectors representing the linguistic resources. Therefore, the function has the input parameters $q$and d which are the vectors detailed previously. Having represented the semantics of linguistic resource as a binary vector, their similarity can be measured as the related binary vectors correlation, quantified, for instance, as the cosine of the angle between them.

$Sim(q,d)= q \times d / |q| * |d|$

Where $|q|$ and $|d|$ are respectively the norms of the vectors representing concepts in the source and in the target linguistic resource. In our case, a source concept will belong to EuroVoc data set, and the target of the linguistic resource will belong to any linguistic dataset to be aligned. The size of vectors *q* and *d* corresponds to the size of the list of dictionary words; each word has a position in the vector representing a concept. A binary value in the vector is equal to 1 if the word of that position is founded in the description of the concept; otherwise it is equal to 0 as shown in figure 2.

3.6 Supervised Learning

Having established a proper similarity measure between concepts, a criterion able to predict matching concepts has to be defined. In Francesconi et al (2008) a criterion was implemented by defining a heuristic threshold over a similarity measure: if the similarity between linguistic resource is over a threshold, a `skos:exactMatch` relation is established. Such strategy, anyway, usually suffers from generalization capabilities out of the matching examples used to tune the heuristics. Generalization capabilities for a prediction strategy can be introduced by adopting machine learning techniques able to learn a predictive function from a training set of matching relations. In Bartoloni and Francesconi (2010) such predictive function is obtained by a Support Vector Machine (SVM)[6] trained to classify a pair of descriptors into two classes *{Match (+1), no-Match (-1)}*.

[6] In order to select the best technique, other machine learning algorithms will be compared to SVM such as *Multi-Layer Perceptron* (MLP).

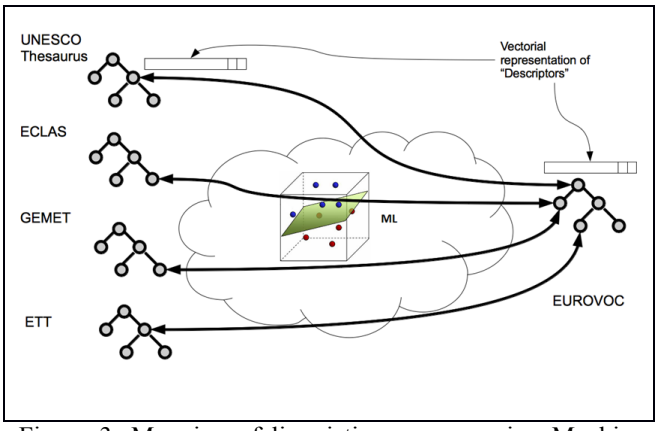


Figure 3: Mapping of linguistic resource using Machine Learning (ML)

A training set for the SVM linguistic resource matching predictor is composed by training examples described by vectors of features deemed representative for descriptors conceptual matching: in particular the i^{th}example is represented by a feature vector $\vec{\varphi}_i$ associated to a pair (d_l,q) of a source and target linguistic resource respectively, including:

- the similarity measure $sim(d_l,q)$, computed according to the cosine function;
- the logical view of the target descriptor d_l

together with a relevance judgment $y=\{+1,-1\}$ for that target descriptor (d_l) on that source descriptor (q) that is either matching (+1) or non-matching concept (-1). Therefore a generic i^{th} training example describing a pair of linguistic resource descriptors and related relevance judgement is

$$\varphi_i=<< sim\,(d_l,q),\, d_l>,y>$$

On the basis of such training set[7], the goal is to build a classifier (a separating surface) which is able to distinguish between matching and non-matching descriptors. The classifier provides also the distance of the examples from the separating surface, giving a measure of the prediction confidence, thus allowing a ranking among candidate target descriptors. The best ranked descriptor is finally chosen as the predicted matching concept.

4. Conclusion and perspectives

This paper presents the preliminary results obtained within the PMKI project for implementing interoperability solution of lexical resources. In particular the main Semantic Web standards available in literature for representing lexicons have been identified and their characteristics were analysed. For the ability of describing lexical components in different languages using LEMON, the related concepts and their mapping relations using SKOS, the Ontolex standard resulted as the preferred model to be adopted as reference for the PMKI platform.

In this work, we formalized the alignment problem as information retrieval problem that can be treated using supervised machine learning techniques. Different data tests are required to train the machine learning algorithm. Our approach can be summarised into three points: 1) Knowledge representation of linguistic standards (SKOS, LEMON and ONTOLEX), 2) Processing framework and steps of alignment and 3) Formalization of linguistic resource, dataset of learning, similarity functions, supervised learning.

The next phase of the project will provide an evaluation of mapping algorithms and propose a technical infrastructure for the implementation and maintenance of lexical resources and their interoperability.

5. Acknowledgements

This work has been drafted under the 2016.16 action[8], which is supported by the ISA² Programme[9]. ISA² is a EUR 131 million programme of the European Commission, supporting the modernisation of public administrations in Europe through the development of eGovernment solutions. More than 20 solutions are already available[10], with more to come soon. All solutions are open source and available free of charge to any interested public administration in Europe.

6. Bibliographical References

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). Modern Information Retrieval. *Addison Wesley*.

Bartoloni, G. and Francesconi, E. (2010). Sharing Knowledge by Conceptual Mapping: the case of EU Thesaural Interoperability. In Proceedings of the *JURIX Conference*, pp. 17-26, ISBN 978-1-60750-681-2, IOS Press, 2010, DOI 10.3233/978-1-60750-682-9-17.

Bosque-Gil & al., (2015) Applying the OntoLex Model to a Multilingual Terminological Resource, *ESWC* 2015

Francesconi, E. Faro, S. and Marinai, E. (2008). Thesauri alignment for eu egovernment services: a methodological framework. In Proceedings of the *JURIX 2008 Conference*, pp. 73-77, IOS Press.

Euzenat, J. Shvaiko, P. (1998). Ontology Matching, Springer-Verlag, Berlin Heidelberg (DE), 2013xviii+511pp. (incl. bibl., index, exercises and solutions), 98 figures, 20 tables, 646 references, ACM Classification: H.3, H.4, I.2, F.4.

Miles, A. Matthews, B. Wilson. M, and Brickley. D. (2005) SKOS Core: Simple knowledge organisation for the Web, DCMI.

Villegas, M. & Bel, N. (2015). PAROLE/SIMPLE 'lemon' ontology and lexicons. *Semantic Web*, 6, 363-369.

[7] The training set is built by human experts from a set of linguistic resources of interest for the European institutions.

[8] https://ec.europa.eu/isa2/actions/overcoming-language-barriers_en

[9] https://ec.europa.eu/isa2/

[10] https://ec.europa.eu/isa2/solutions_en

The Abkhaz National Corpus

Paul Meurer
University of Bergen
Bergen, Norway
paul.meurer@uib.no

Abstract

In this paper, we present the Abkhaz National Corpus, a comprehensive and open, grammatically annotated text corpus which makes the Abkhaz language accessible to scientific investigations from various perspectives (linguistics, literary studies, history, political and social sciences etc.). The corpus also serves as a means for the long-term preservation of Abkhaz language documents in digital form, and as a pedagogical tool for language learning. It now comprises more than 10 million words and is continuously being extended. Abkhaz is a lesser-resourced language; prior to this work virtually no computational resources for the language were available. As a member of the West-Caucasian language family, which is characterized by an extremely rich, polysynthetic morphological structure, Abkhaz poses serious challenges to morphosyntactic analysis, the main problem being the high degree of morphological ambiguity. We show how these challenges can be met, and what we plan to further enhance the performance of the analyser.

Keywords: Abkhaz National Corpus, lesser-resourced languages, polysynthetic languages, parsing of rich morphology

1. Motivation, sociolinguistic situation

Abkhaz, a West-Caucasian language, is spoken in Abkhazia by approx. 100,000 people, and by a considerable number of people in the Abkhazian diaspora in Turkey, Syria, Jordan, Adzharia (Georgia), Russia and elsewhere, with exact numbers unknown.

Abkhaz has a rather young literary tradition, reaching back no further than to the end of the 19th century, when Russian linguists devised a script for the language. In the 20th century, a vivid literary production developed, and the language was taught in schools and used as a medium of mainstream communication.

However, already in the 19th century, when Abkhazia became part of the Russian Empire and the muslim Abkhazians were deported to Turkey, the Abkhazians were in the minority. Because of the Soviet and Georgian rule, and the continuous influx of Georgians and other non-Abkhaz population, Abkhaz came, in the course of the 20th century, even more in the position of a minority language, with many ethnic Abkhazians giving up their language in favour of Russian. In 1989, before the Abkhaz-Georgian war, only 17.8% of the population of the Autonomous Republic Abkhazia were ethnic Abkhazians.[1]

Because of the almost total exodus of ethnic Georgians from Abkhazia during the Abkhaz-Georgian war, the Abkhazians are now again the majority population in the region (50,7% according to the 2011 census). However, Russian, which is the second official language in the self-declared Abkhaz state, has remained the dominant language of mainstream communication.

Since the Abkhaz de-facto independence, there is a growing awareness in the Abkhaz public of the need to improve the viability of the language. There are also political aspirations to make Abkhaz more widely used in public, and even a law ("Law on the official language of the Republic Abkhazia" from 2006) that tries to enforce the use of Abkhaz in the state administration and in public life, with the long-term aim to have Abkhaz replace Russian as the dominant language.

But the implementation of these measures, overzealous as they appear from the outset, is severely hampered by the lack of qualified teachers and adequate or even basic printed or electronic teaching material.

On this background, the Abkhaz National Corpus (AbNC)[2] project has the following aims:

- To establish a linguistically annotated Abkhaz language corpus of reasonable size
- To develop freely available computational resources (electronic dictionaries and lexicons, parsers and more) for the language
- To give learners of the language a tool that provides electronic support for reading texts online
- To serve as a long-time storage repository for Abkhaz language texts, dictionaries and other resources in electronic form (as a CLARIN[3] resource)

2. Previous work

The project I am reporting on is the first computational work being done on Abkhaz, and indeed on any West-Caucasian language. The only corpus resource besides the AbNC is a small corpus of 2 million words without grammatical annotation.[4] There are plans to create an annotated Circassian/Adyghe corpus (Arkhangelskij and Lander, 2016), but this work seems still to be in a preparatory phase.

3. The project

The AbNC project is in the fortunate situation that there exists a large online collection of freely available Abkhaz

[1] See https://en.wikipedia.org/wiki/Demographics_of_Abkhazia for these numbers.

[2] http://clarino.uib.no/abnc

[3] https://www.clarin.eu

[4] http://baltoslav.eu/apsua

texts.[5] The maintainers of the site are enthusiasts who put a lot of effort into collecting and scanning Abkhaz books and other documents, with the aim to build a comprehensive archive of Abkhaz text resources. In addition to scanned texts, they also make available digitally-born PDF files from Abkhaz publishing houses, and it is mainly these digital PDF texts (originating from 2010 or later, but often constituting reprints of older editions) the AbNC project draws upon. In addition, internet news sites were harvested, some important texts were scanned and OCRed in the project; and some texts came directly from the authors.

The texts belong to a variety of genres, with novels, traditional texts and news texts in the majority. Table (1) shows approximate sizes for the genres represented in the corpus. The sizes are in 1000 tokens, where tokens comprise both words and punctuation.

(1)

Genre	*Size*
fiction	
traditional (fairy tales, Nart saga etc.)	1,000
religious (New Testament)	200
novels and other prose	6,700
poetry	400
drama	300
non-fiction	
journalism and news	1,250
political	20
textbook	140

The texts were thoroughly preprocessed and annotated with rough structural markup adhering to TEI P5 standards that captures section headings, paragraphs, page breaks, footnotes, lines of poetry and the structure of drama texts. In addition, necessary metadata was included in the TEI header. The result is a very clean corpus with texts that can be read nicely in a web browser.

The project was able to agree with the Abkhaz partners on an open license that poses very few restrictions on the use of the texts in the confines of the project. In technical terms, the license is a CLARIN PUB license (CLARIN_PUB-BY-NC-ND).

4. Typological features of Abkhaz

Phonetically, Abkhaz is characterized by a large number of consonant phonemes (between 56 and 65, depending on the dialect), whereas there are few vowel phonemes. Only *a* and *ə* are phonemic, whereas *e, i, o* and *u* can appear in loan words and as phonetic realizations of *a* and *ə* in certain contexts. The nominal morphology is rather simple; there is no case marking, but the language exhibits noun-noun and noun-adjective compounding. Postpositions can be suffixed to the noun or adjective. Remarkably, cardinal numbers can be prefixed to the noun. The verbal morphology can be characterized as agglutinative and polysynthetic, with a huge number of verbal prefixes and suffixes. The position of the affixes in the verb is rigid, they fit into given slots of a general template.

[5] http://apsnyteka.org

Morphemes are generally monosyllabic (*Ca, Cə*), with a high degree of homonymy. There is however little suppletion, few irregularities and few phonological processes. Abkhaz has free and dynamic word stress which is not coded in the orthography.

5. The analyser

Since there is no disambiguated training data available, and because of the extraordinarily rich morphology of the language, we did not consider a statistical approach to morphological tagging at the outset. Instead, we developed a finite-state morphological analyser using the FST platform (Beesley and Karttunen, 2003), and Constraint Grammar rules are used for disambiguation. We are however planning to train a statistical disambiguator that operates on the partially disambiguated output of the CG parser once a sufficiently large gold standard corpus has been created.

5.1. The lexicon

The lexicon of the analyser is based on a large bilingual (Abkhaz-Russian) dictionary (Kaslandzia, 2005), which we had to convert into a suitable machine-readable format in a semi-automatic process. All lexicon entries of the dictionary are marked for word stress, which is distinctive in Abkhaz. For the nouns, plural forms are given. The verb entries, with the verbal noun as lemma form, contain information on transitivity; static and dynamic verbs are distinguished, and for each entry, a sample inflected form is given, in addition to translations and example phrases. Unfortunately, there are many typographic and other errors in the dictionary entries, and the important transitivity information is not very reliable. We have corrected these errors manually, and often we had to consult other dictionaries (Yanigasawa, 2010; Š'aḳrəl and Kondž'aria, 1986) and Abkhaz native speakers.

To be useful for the analyser, the verb lemma entries had to be segmented correctly into preverb, stem and affix morphemes, and to those segments the correct morphological features had to be assigned. The segmentation candidates were generated by an auxiliary finite state transducer that is basically the subset of the full finite-state analyser restricted to verbal nouns. In general, there are many segmentation candidates, and we did choose the correct analyses manually.

As an example, the annotation of the correct segmentation of the verb 'аитцанаршəшəара' *áiçanarš^{w}š^{w}ara* 'to be shaken (in a car etc.)' is *á-ai/R-ça/A-na/S-r:š^{w}š^{w}a-ra*, with markers for reciprocal (*/R*), local preverb taking an object (*/A*), inanimate agent (*/S*), and causative (*r:*) before the root. The prefix *á-* is the generic article and the suffix *-ra* is the verbal noun marker, both of which are always present.

A specific problem is to properly segment the stem of intransitive verbs. A verbal stem is either simple (consisting of the verbal root only), or it is composed of a (simple or complex) preverb and the root. For transitive verbs, the boundary between preverb and root is obvious from the inflected form, because the agent affix (normally *-i-, -l-* or *-r-* in the sample forms) is placed immediately before the root. Intransitive verbs however have no agent affix; here, the boundary is only visible in comparatively rare negated

aorist forms, where negation is marked by the negation infix *-m-* placed between preverb and root. When such a form was not contained in the example phrases, we had to find evidence for the correct segmentation by a corpus search or from informants.
The analyser lexicon comprises 6,750 verb stems, 16,000 nouns, 3,000 adjectives and 1,350 adverbs.

5.2. The feature set

The morphological analyser (Meurer, 2011) takes as input an orthographic surface form in the Abkhaz Cyrillic script and returns, for each found reading, a lemma form (containing stress information), and a bag of morpho-syntactic features.
The feature set is quite rich, it comprises more than 350 features. There are 160 features for inflected verb forms alone. This high number is due to the polysynthetic nature of the Abkhaz verb; there are affixes not only for the subject, but also for direct, indirect and locative objects in the verb, there are applicative markers with associated person affixes, as well as adverbial, conjunctional, question and clitic elements. Person affixes include relative and reciproque markers. The verb form in Example (2) will be analysed as the lemma form 'аҽамадара' *a-č·á·ma·da-ra* with feature set (*V Dyn Tr NonFin Pres Neg Refl:Rel Reln:Pot RO:Rel LO:3Pl QWho Excess*).

(2) зҽеызызрымамдатцәода
z.čə-zə.z-rə-ma-m-da-c^{w}a.wa-da
Rel.Refl-Rel.Pot-3Pl-PV-Neg-ROOT-Exc-Pres-Who?
'who cannot trust them too much?', lit.: 'who is it who cannot betrust whose head to them in excess?'

5.3. Ambiguity

Since nominal and verbal stems and affixes often consist of only one simple (CV) syllable, there is considerable homonomy among those morphemes. Although morphemes of equal syllable structure may differ in their accent status, this information is not (or in some cases only indirectly) available in the orthographic word form, since stress is not marked in present-day orthography.
As a consequence, most word forms are highly ambiguous; often there are many ways to segment a word form into preverb(s), stem, and affixes, and those stems and affixes may by themselves be ambiguous. In addition, lemma forms may differ solely in their stress pattern.
For example, the word 'илаба' *ilaba* (with its most frequent reading 'his stick') gets 42 analyses, consisting of inflected and predicative forms of *a-labá* N 'stick', *a-lába* N 'male dog', *abá* N 'textile', *abá* A 'dry', *á-la·ba-ra* V 'suck up', *a-ba-rá* V 'see'.

5.4. Disambiguation

Abkhaz poses serious challenges to disambiguation due to the virtual absence of dependent marking (e.g., no case marking), the weak significance of word order, and the extreme prevalence of homonymy.
To keep ambiguity not higher than necessary, we tried as much as possible to avoid overgeneration in the morphological analyser. Obviously, the prefixes and suffixes that can fill the slots of the verb template cannot combine freely. The negation marker *-m-*, for example, can only occur in one position at a time, although there is a prefix and a suffix slot position for negation. The position of the marker depends on a combination of tense, finiteness, and whether the form is dynamic or static. Relational prefixes are restricted to non-finite tense suffixes. Other restrictions depend on syntactic and semantic properties of a verb; e.g., agent prefixes are obviously restricted to transitive verbs only. The relevant grammatical literature (e.g., (Spruit, 1986; Šaduri, 2006; Hewitt, 2010), to mention a few important works) is however in many cases vague about those combinatorial restrictions, and we had to work out specific rules to be used in the implementation of the morphology.
One such case is the possibility of relative markers in a verb form. A general rule restricts relative markers to non-absolutive non-finite forms. However, a corpus search reveals that the situation regarding absolutives is more complicated:
In Abkhaz, absolutives are used in serial verb constructions, where the absolutive and the verb following it are part of the same event. This can be seen in (3), where the verb 'амазаара' *á-ma-zaa-ra* 'have' initiates a serial construction.

(3) Ача иманы иааит.
a-č'á$_i$ *ø$_i$-i$_j$-ma-nɔ́* *i$_j$-áai-ṭ*
the-bread (it)-it-have.Abs it-come.Aor.Fin
'It came with (lit. having) the bread.'

However, if the main verb of the serial construction is a non-finite relativized verb like in (4), its subject relative marker (*-i-*) is taken up by the absolutive (in the form of the indirect object relative marker *-z-*). Thus, we get an absolutive form with a relative marker.

(4) ача зманы иааиз
a-č'á$_i$ *ø$_i$-z$_j$-ma-nɔ́* *i$_j$-áai-z*
the-bread (it)-Rel-have.Abs Rel-come.Aor.NonFin
амашьына
a-maš'ɔ́na
the-car
'the car which came with the bread'

In devising the Constraint Grammar rules, several strategies were used to disambiguate homonymous forms:

- Highly unlikely (albeit possible) analyses of frequent words were ruled out.
- Syntactic rules were used where possible.
- To disambiguate further, semantic information in the form of selectional restrictions was used.
- As a heuristics, analyses that match a dictionary entry were preferred.

5.5. Treatment of predicate phrases

In Abkhaz, stative verbs can be formed from nouns and adjectives, but also from two-element noun phrases, with predicative meaning: 'X is NP'. This is shown in (5), where the brackets indicate the NP that is surrounded by verbal morphological material.

(5) Дысԥыза бзиоуп.
də-[s-y^{w}ə́za bzío]-up.
he-[my-friend good]-be.Stat.Pres.Fin
'He is my good friend.'

Аҟәа шқалақь бзиоу
Áq^{w}a ø-š-[kalak' bzío]-u
Sukhum (it)-how-[city good]-be.Stat.Pres.NonFin
'what a good city Sukhum is'

The close analogy to true static verbs is apparent from (6).

(6) Дгылоуп.
d-[gə́lo]-up.
he-[stand]-Stat.Pres.Fin
'He is standing.'

ишгылоу
i-š-[gə́lo]-u
it-how-[stand]-Stat.Pres.NonFin
'how it is standing'

The question arises whether to analyse such constructions as verbs, or to keep the part of speech of the underlying noun or noun phrase components. Treating a verbal form derived from a complex NP as a verb would entail that the construction would have to be represented as a single token containing whitespace. This would be unfortunate in the context of a searchable corpus, for several reasons. First, it would not be clear how the lemma form of such a complex static verb should look like – verbal nouns cannot be derived from two-element NPs. And even if a solution to this problem were found, it would be complicated and unintuitive to search for the components of the NP stem in such constructions.
Therefore, the two components of the predicate NP are treated as two separate tokens, each being annotated with the part of speech of the underlying component. In addition, the components are annotated with a *Pred* feature, indicating that they are parts of a predicative static verb construction, and with a *LHP* (left half predicate) and *RHP* (right half predicate) feature, respectively, showing that they are the left and right parts of a full predicate construction. They also get the features that other affixes of both components might contribute. Such an analysis might not be ideal, but it is satisfactory in the context of a searchable corpus. The analysis of the phrase 'шқалақь бзиоу' *ø-š-[kalak' bzío]-u* would thus look like this:

(7) *á-kalak'* Noun NH Sg Pred LHP How S:3 S:Ad
a-bzía Adj Sg Pred RHP NonFin Pres

This analysis is also extended to static verbs derived from single nouns and adjectives.

5.6. Performance

At the present stage of development, it is premature to give exact precision and recall figures for the analyser. For the corpus as a whole, 93.6% of the tokens and 74.0% of the types are recognized by the analyser. A non-formal evaluation of the parser on a section of a novel chapter (medium complexity) indicates that more than 85% of the tokens are assigned the correct part of speech. Precision for all features will obviously be somewhat lower.

6. The corpus

The Abkhaz National Corpus is available in the corpus management platform Corpuscle (Meurer, 2012), which offers advanced search capabilities and a user-friendly web interface. To make the corpus accessible to Abkhazians, who are often uncomfortable with English, the tool was localized to Russian and to Abkhaz. All Abkhaz text, which by default is shown in the modern Abkhaz Cyrillic alphabet, can also be viewed in scientific transliteration, which makes it easier for non-specialist linguists to use the corpus.

6.1. Features for language learners

In addition to being a corpus-linguistic resource and tool, the corpus also serves as a digital library and a pedagogical tool for language learning.
From the catalogue page, where all texts are listed by author, title, creation date and genre, the user can select a text for reading. On the reading page, one page of the text is shown, which in most cases corresponds to a page of the printed edition. The user can maneuvre to the next or previous page, or select a page or a book section from a drop-down menu. The current reading position can be stored as a bookmark.
Most importantly, by clicking on a word in the text, the user is shown grammatical information about that word, including lemma form and morphosyntactic features, and the lemma can be looked up in the integrated Abkhaz-Russian Dictionary (Kaslandzia, 2005) (see Fig. 1).

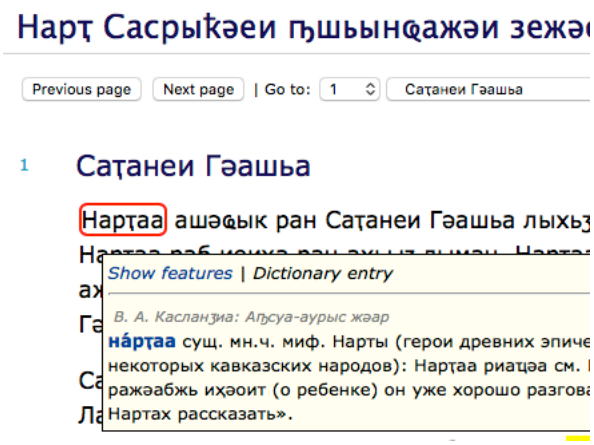


Figure 1: Dictionary lookup.

7. Conclusions and future work

Abkhaz, a polysynthetic language, exhibits a high degree of morphological complexity and word form homonymy. We have outlined how a morphosyntactic analyser can be implemented that adresses these complexities. Although much

remains to be done to improve precision and recall of the analyser, it is already in its present state a useful tool for annotating a text corpus of medium size, the Abkhaz National Corpus. When the feature set has stabilized, a gold corpus will be manually annotated, so that exact performance figures can be calculated. The gold corpus will also be used to train a statistical disambiguator that will be used on top of the Constraint Grammar-based disambiguator.
The corpus itself is continually growing, and the long-term aim is to include all important texts that have been written in Abkhaz.

8. Acknowledgments

The Abkhaz National Corpus project started in 2015. Project partners are the Goethe University Frankfurt, Institute of Empirical Linguistics, the Centre for Civil Integration and Inter-Ethnic Relations (CCIIR), Tbilisi, and the organization Business Women of Abkhazia, Sukhum.
The project is funded by the United States Agency for International Development (USAID).
We would like to thank the Abkhaz project partners for their valuable help in resolving questions about Abkhaz morphology and the lexicon.

9. Bibliographical References

Arkhangelskij, T. A. and Lander, Y. A. (2016). Developing a polysynthetic language corpus: problems and solutions. In *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2016"*, Moscow.

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications, Stanford.

Hewitt, G. (2010). *Abkhaz. A Comprehensive Self-Tutor*. LINCOM, München.

Kaslandzia, V. (2005). *Apsua-aurəs žʷar (Abkhaz-Russian Dictionary)*. Abkhazian State Academy, Sukhum.

Meurer, P. (2011). A finite state approach to Abkhaz morphology and stress. In *Lecture Notes in Computer Science 2011, Volume 6618*, pages 271–282.

Meurer, P. (2012). Corpuscle: a new corpus management platform for annotated corpora. In *Exploring newspaper language: using the web to create and investigate a large corpus of modern Norwegian*, pages 31–49. John Benjamins Publishing Company.

Spruit, A. (1986). *Abkhaz studies. Doctoral dissertation*. Leiden.

Yanigasawa, T. (2010). *Analytic Dictionary of Abkhaz*. Hituzi Syobo Publishing, Tokyo.

Šaduri, I. (2006). *Morfologija abxazskogo jazyka*. Tbilisi.

Š'aḳrəl, K. S. and Kondž'aria, V. H. (1986). *Apsua bəzšʷ a ažʷar (Dictionary of the Abkhaz language)*. Alašara, Sukhum.

Collecting Language Resources from Public Administrations in the Nordic and Baltic Countries

Andrejs Vasiļjevs, Rihards Kalniņš, Roberts Rozis, Aivars Bērziņš
Tilde
Vienibas gatve 75a, LV1004, Riga, Latvia
{andrejs, rihards.kalnins, roberts.rozis, aivars.berzins}@tilde.com

Abstract

This paper presents Tilde's work on collecting language resources from government institutions and other public administrations in the Nordic and Baltic countries. We introduce the activities and results of the European Language Resources Coordination (ELRC) action in this region, provide a synopsis of ELRC workshops held in all countries of the region, identify potential holders and donors of language data suitable for improving machine translation systems, and describe the language resources collected so far. We also describe several national projects and initiatives on sharing of language data accumulated in the public sector and creation of new language resources from this data. Opportunities and challenges in consolidating language data from the public sector are discussed, and related actions and regulatory initiatives are proposed.

Keywords: language resources, national/international projects, parallel corpora, ELRC, machine translation, CEF eTranslation

1. Introduction

Several large-scale projects and initiatives have been undertaken in this century to collect language resources (LR) and create LR repositories and infrastructures on a pan-European scale.

A family of closely related projects co-funded by EC FP7 and ICT-PSP programmes (T4ME, CESAR, META-NORD, META-NET4U) succeeded by establishing the META-NET network of excellence, comprised of 60 research centres in 34 countries (Rehm&Uszkoreit, 2011), and creating META-SHARE[1], an online repository of language resources distributed and automatically synchronized across 28 hosting nodes across Europe (Piperidis, 2012). The META-NORD project contributed to these developments by coordinating language resource collection and setting up META-SHARE repositories in all 8 Nordic and Baltic countries (Vasiļjevs et al., 2011).

Nordic and Baltic countries are part of the CLARIN infrastructure[2] (Wittenburg et al., 2010), which supports the sharing, use, and sustainability of language data and tools for research in the humanities and social sciences (Varadi et al., 2008). National consortiums are established and funded by the Nordic governments, such as CLARINO[3] in Norway, FIN-CLARIN[4] in Finland and CLARIN-DK[5] in Denmark.

The breath and usability of web data is demonstrated by the OPUS corpora[6] (Tiedemann, 2012), in which parallel data is extracted and aligned from numerous web sources, covering from formal (e.g., legal texts) and highly technical (e.g., user interface strings, medicine descriptions) to informal (e.g., movie subtitles, conversational phrases) language.

Consolidation activities have also targeted specific types of language resources. For instance, the EuroTermBank project supported by the EC eContent programme has consolidated heterogeneous multilingual terminology resources in a distributed online termbank (Vasiļjevs et al., 2008).

Still, most of the LRs in these repositories are created by research institutions, or the translation departments of large international organisations (such as the European Commission or United Nations), and have resulted from research projects. At the same time, thousands of translation units are created every day by companies and public sector institutions. Acquisition of commercial data can be very costly (e.g., monetary of reciprocal data exchange at TAUS Data Repository[7]).

Data in the public sector, at least in theory, should be much more available thanks to the Public Sector Information Directive adopted in EU and EEA countries (European Parliament, 2003). The directive basically stipulates that data created by taxpayer's money should be made freely available for any use, including commercial, with only limited exceptions for privacy or confidentiality protection.

In this paper we present our work on collecting language resources from government institutions and other public administrations in the Nordic and Baltic countries.

We introduce the activities and results of the European Language Resources Coordination (ELRC) action in this region, provide a synopsis of ELRC workshops held in all countries of the region, identify potential holders and donors of language data suitable for improving machine translation (MT) systems, and describe the language resources collected so far.

We also describe several national projects and initiatives on sharing of language data accumulated in the public sector and creation of new language resources from this data. Opportunities and challenges in consolidating language data from the public sector are discussed, and related actions and regulatory initiatives are proposed.

[1] http://www.meta-share.org/
[2] http://www.clarin.eu/
[3] http://clarin.w.uib.no/
[4] http://kitwiki.csc.fi/twiki/bin/view/FinCLARIN
[5] http://info.clarin.dk/en/
[6] http://opus.nlpl.eu/
[7] https://www.tausdata.org/

Work-shop	Location	Organizers	Partici-pants
Denmark workshop, Mar-7-2016	European Environment Agency in Copenhagen	Danish Language Committee, Tilde	71
Estonia workshop, Feb-11-2016	EC Representation Office in Tallinn	Tilde, Estonian Language Resource Centre Office of the EC Representation in Estonia	66
Finland workshop, Feb-19-2016	University of Helsinki	University of Helsinki, Tilde	34
Iceland workshop, Nov-11-2016	Safnahúsið (the Culture House) in Reykjavik	Vigdís Finnbogadóttir Institute Tilde	36
Latvia workshop, Oct-6-2015	EC Representation Office in Riga	Tilde, Culture Information Systems Centre, Office of the EC Representation in Latvia	59
Lithuania workshop, Feb-24-2016	Lithuanian Government Building in Vilnius	State Commission of the Lithuanian Language, Office of the EC Representation in Lithuania, Office of the Government of the Republic of Lithuania, Tilde IT	141
Sweden workshop, Mar-10-2016	Europahuset in Stockholm	Language Council of Sweden, Språkbanken, EC Representation in Sweden, Tilde	47
Norway workshop, Jun-8-2016	Difi's course and conference venue in Oslo	Difi Agency, University of Bergen Tilde	53

Table 1 : ELRC Workshops in Nordic and Baltic countries

2. ELRC in the Nordic and Baltic countries

The aim of the European Language Resource Coordination action is to identify and gather language and translation data relevant to national public services, administrations, and governmental institutions across all 30 European countries participating in the Connecting Europe Facility (CEF) programme[8], i.e. EU Member States, Norway and Iceland (Lösch et al., 2018). This data is used to improve the quality of automated translation systems provided by the European Commission's CEF eTranslation service[9]. In return, CEF eTranslation makes MT services available to public service providers and administrations to support them in their interactions with citizens across language barriers.

The ELRC consortium includes DFKI[10], ELDA[11], ILSP/Athena RC[12] and Tilde[13]. It operates under service contracts SMART 2014/1074 and SMART 2015/1091 with the European Commission. Tilde is responsible for coordination of ELRC activities in Nordic and Baltic countries presented in this paper. ELRC activities in CEF countries are supported by one technological representative (Technology National Anchor Point) and one representative from the public services administration (Public Services National Anchor Point). ELRC National Anchor Points in the Nordic and Baltic region are highly respected representatives from academic sector (University of Helsinki, Vigdís Finnbogadóttir Institute, University of Iceland, IMCS at the University of Latvia), national language policy institutions (Danish Language Committee, State Commission of the Lithuanian Language, Swedish Language Council), government agencies (Estonian Ministry of Education and Research, Finnish Prime Minister's Office, Latvian Culture Information Systems Centre, Norwegian Agency Difi), and language resource centres (Estonian Language Resource Centre, National Library of Norway, Språkbanken at the University of Gothenburg). A full list of National Anchor Points is available on the ELRC online platform[14].

3. Findings from the ELRC Workshops

The first task of ELRC was to inform and engage the public administrations. For this task we organized a series of local workshops in all Nordic and Baltic countries, with the support of national experts (see Table 1). The goals of the workshops were to raise awareness about the importance of language data, understand the needs of national public sector administrations with regard to automated translation, jointly identify relevant sources of multilingual language resources, and discuss technical and legal issues involved in the use of data for automated translations.

Although having the same objectives and similar structure, the workshops reflected differences and particularities in various areas – national policies in the field of language resources and language technologies, openness of the public sector to share linguistic data, awareness of applicability of machine translation and other language technology tools, etc.

The Latvia workshop included a presentation of government activities in creating the national machine translation platform Hugo.lv, as part of Latvia's e-Government service infrastructure (see Section 6.1).

The Latvian workshop also showed that the largest translation volumes are accumulated at the State Language Center (formerly the Translation and Terminology Center), which has a well-established translation process and large translation memories accumulated through the usage of the SDL Trados computer-assisted translation (CAT) tool. This agency also intends to consolidate Latvian terminology data into a national terminology database. We see this agency as a major collaboration partner and provider of valuable parallel data.

The Estonia workshop raised a discussion on the applicability of state-of-the-art machine translation systems for such complex, agglutinative languages as Estonian and Finnish. Several participants expressed concerns that current systems like Google Translate produce translations that are not suitable for practical applications or post-editing. This emphasized the need for much larger volume of parallel data to train MT systems of

[8] https://ec.europa.eu/inea/en/connecting-europe-facility
[9] https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation
[10] http://dfki.de/en
[11] http://elda.org/en
[12] http:// www.ilsp.gr/en
[13] http://tilde.com
[14] http://lr-coordination.eu/anchor-points

significantly better quality. Public sector participants were interested in contributing to the development of Estonian MT with their language data, making it available not only for the EC, but also for Estonian researchers and developers through the Estonian Language Resource Center.

The Finland workshop explored the complexities of translating into the Finnish language. Several activities such as FIN-CLARIN have successfully mobilized Finland's language technology community to meet this challenge. The workshop demonstrated the acute needs of the public sector in higher quality MT. The Finnish public administration has a pressing need to translate information into both Swedish and Finnish, as well as into major European languages in order to reach the multilingual population. Therefore, the public sector (e.g., Office of the Prime Minister, Municipality of Helsinki, Ministry of Justice, Tax Administration) is actively following the developments in CEF Automated Translation and would be potentially very interested in integrating the platform into public services.

The Lithuania workshop had the largest number of participants (141) and prominent keynote speeches from government representatives, which demonstrated the efficiency of national level coordination by the Lithuanian State Language Commission. A particularly valuable source of translation data was identified in Seimas (the Lithuanian Parliament). Its Legislation Editing and Translation Unit is tasked with the translation, editing, signing, and publication of legislative acts submitted to the Seimas, legislative acts adopted by the Seimas, documents of the Board of the Seimas, and decrees of the President of the Seimas. The 12-member department has collected Translation Memories primarily in Lithuanian, English, and Russian.

The Denmark workshop was a very well-attended event, with nearly 70 participants representing a wide cross-section of public institutions, research organisations, and businesses in Denmark.

The program featured presentations from the ELRC consortium, governmental, translation, and language technology sectors. Both the audience and the speakers had a positive attitude towards the CEF platform and showed interest in how the Danish public administration can provide language resources to the European Commission in order to develop CEF AT. This interest was also compounded by the presence of members of the private sector, particularly representatives from IBM Denmark and translation companies.

The workshop also stimulated discussion on the need to set up national level coordination of public administrations on LRs and application of MT.

The Sweden workshop had a strong showing from actual translators and editors at the Swedish public administration, particularly English language translators at the Ministry of Foreign Affairs. Thanks to the participation of these translators, many of whom are foreign nationals living and working in Sweden, the workshop was able to generate discussion around an actual use case of MT in the public sector, as well as identify multiple sources of translation memories from the Swedish public administration.

The Norway workshop was one of the most successful seminars organized during the ELRC action to date. Norway has a diverse linguistic landscape, due to the co-existence of several official languages: Norwegian variations Bokmål and Nynorsk, Sami, and Kven. Therefore, language resources exist in each of these distinct languages – though, of course, in varying degrees of volume and coverage. As Norwegians are keenly aware of this disparity, they are also aware of the challenges faced by developing machine translation technologies for the public sector. For example, should engines be developed for all official languages, or just one?

To answer these and other questions, the public administration in Norway has already commissioned a large-scale study of the possible costs of collecting sufficient language resources in each of the official languages, as well as the foreseen benefits of utilizing machine translation in the everyday work of the public sector (Oslo Economics, 2016).

The conclusions reached were that national programmes in Norway should work hand-in-hand with the ELRC action to generate the largest possible volumes of language resources for raising the quality of Norwegian MT provided by CEF eTranslation. Adjusting machine translation for the CEF Online Dispute Resolution (ODR) service[15] was identified as a particular priority that requires data that represents more informal language used by users in customer complaints. The workshop stimulated national activities to complement work of ELRC on collecting data related to ODR. The workshop also identified several sources of language resources from the Ministry of Foreign Affairs, which has been collecting Translation Memories generated by translators working in the SDL Trados Studio CAT tool for many years.

The Iceland workshop generated a good deal of interest among a wide range of participants from both the public and private sectors. Interest in language technology in Iceland is certainly driven by the relatively small size of the language community (with just 330 000 native speakers) and the high degree of national pride in keeping the language vital.

The Icelandic workshop was also candid about the limits of developing high-quality machine translation services for a language with such a small amount of input data. Therefore, the participants at the workshop were strongly supportive of mobilizing the public administration to gather as much language resources as possible for developing CEF eTranslation for Icelandic.

Detailed reports on the workshops as well as presentations and video recordings are available on the ELRC online platform[16].

4. Language Resource Collection

To facilitate LR collection, the ELRC consortium created the online LR repository ELRC-SHARE[17], based on the META-SHARE open source distributive. The ELRC-SHARE data model is an extension of the META-SHARE schema (Gavrilidou, 2012) with added fields to support LR management.

ELRC dissemination activities have already spurred the interest of numerous public administrations to assess and

[15] http://ec.europa.eu/consumers/odr

[16] http://lr-coordination.eu/events

[17] https://www.elrc-share.eu

share their language data. Some of them have directly uploaded their data on the ELRC-SHARE repository. Others consulted with ELRC experts or National Anchor Points (NAPs) and provided their data through them.
By the end of the first phase of the ELRC action, 65 language resources were provided from all countries of the region (see Table 2). Most LRs (39) are parallel corpora, and only 6 are monolingual corpora. This is because ELRC has prioritized parallel data as the scarcest and most necessary resource to improve the quality of MT.
Instead of providing language data, several institutions pointed to their websites with parallel multilingual content. ELRC partners then crawled these sites to extract parallel data from it, process, align and build a parallel corpus from the data collected.
The collected data is first provided to the EC for use in training their MT systems. After an assessment by the EC, these language resources will also be distributed on the META-SHARE platform and the European Open Data Portal.

Type of Text	Corpus	Lexical Resource
Danish	2	2
Estonian	7	1
Finnish	11	4
Icelandic	1	1
Latvian	9	1
Lithuanian	4	1
Norwegian	2	1
Swedish	9	9
Total	**45**	**20**

Table 2 : LRs collected and processed in the first phase of ELRC action by language and type

Among all the data collected and donated in the project there were some outstanding resources in terms of their volume. The main difference with them is that the content of these resources was created in a managed way: as part of a centralized translation workflow or as part of a document or terminology content management system. This clearly illustrates the importance and yield from managed workflows in the area of language resource processing. Also, most of those resources were donated to the ELRC action as part of direct communication on the part of NAPs with the resource holder, a relationship built during ELRC events and follow-up communication. The following summarize some of the most outstanding resources:

- Translations of Lithuanian legislation from Seimas of the Republic of Lithuania – the entire translation memory of Lithuanian-English translations exported from the translation server of Lithuanian Seimas – a total of over 130 000 translation units donated to the ELRC action by the Legal Acts Editing and Translation Unit of the Document Department at the Seimas of the Republic of Lithuania.
- Corpus of the Translations of Estonian legislation with 47 255 translation units, donated for the ELRC by the Ministry of Justice of the Republic of Estonia.
- Translation memories from the Ministry of Foreign Affairs containing Norwegian Bokmål and Nynorsk translations of Acquis Communautaire with 733 081 translation units licensed under CC-BY license. This resource is hosted by the Norwegian Language Bank (Språkbanken).
- Icelandic Termbank by The Árni Magnússon Institute for Icelandic Studies containing 103 753 term entries in 41 term collections in various domains, licensed under CC-BY-SA license.

5. Intellectual Property Rights

The ELRC consortium takes particular care about the clearance of Intellectual Property Rights (IPR) for all resources collected or received from donors. To do so, the Consortium follows a set of guidelines established by ELRC.
The basic principles behind this workflow consist of (1) checking whether the data under consideration are available under an Open License; otherwise (2) see whether they are under PSI scope, or (3) if they need to be negotiated. If neither step 1 nor step 2 suffice, the Consortium contacted the data owners to negotiate and agree upon usage conditions.

6. National Projects

Besides the ELRC action, which was initiated and funded by the EC, there are several other activities on a national level that contribute to the collection of language resources from public administrations.

6.1 HUGO.LV in Latvia

The collection of parallel data is a part of the work on the machine translation platform for the e-Government infrastructure of Latvia. The platform, called HUGO.LV and developed by Tilde, includes machine translation systems for Latvian, English, and Russian tailored for the requirements of various e-Government services (e.g., the state e-services portal Latvia.lv). As part of the platform's development efforts, a large corpus of parallel and monolingual data was collected.
The project has raised awareness in the state administration about the need to manage translation service tenders in a way that not only fulfils the direct goal of acquiring translations, but also requires translation memories to be returned in order to build an open corpus of public sector translations, which, in turn, can be further reused on newer builds of the Hugo.lv service. To simplify the data workflow, a LR data upload facility has been created in HUGO.LV. Within this facility, registered state officials can upload parallel content in any format: translation memories (.tmx .sdlxliff, etc.), text files (TXT, DOC, DOCX, etc), PDF, or JPEG. This ensures that language technicians can transform the content to reusable corpora, to be used in the training of HUGO.LV MT systems, as well as to be distributed on open data portals. Figure 1 contains a screenshot of the parallel data upload page.

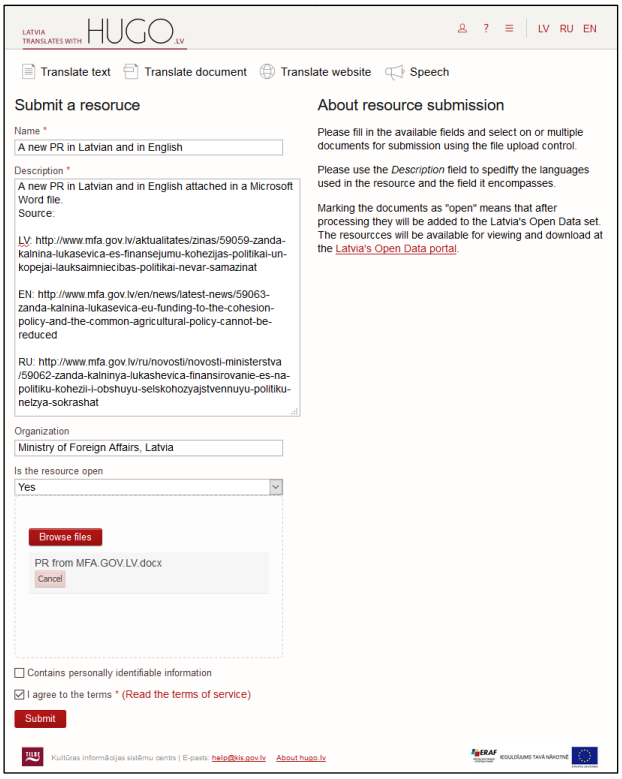


Figure 1: HUGO.LV submission page for parallel textual data

The government institution in charge of HUGO.LV – Culture Information Systems Centre (CISC) – has several cooperation partners. The Latvian State Language Centre – *Valsts valodas centrs* (*VVC)* – performs official translation work of national legislation, international conventions and agreements, EU legal documents between Latvian and English as well as maintains official terminology for the translation of legal texts. VVC is committed to donate TMs with at least 250 000 translation segments and 100 000 terminology entries for the benefit of the second phase of the HUGO.lv project. The Latvian Court Administration, in turn, is anticipated to provide 2000 anonymized judicial decisions and other parallel documents. The Information Society Committee, led by the Prime Minister, has tasked CISC with compiling a full list of Translation Memories available in the Latvian public sector, to be submitted by the end of 2018.

CISC has agreed to cooperate with ELRC and contribute to its data collection activities.

6.2 Data Creation for ODR in Norway

The Norwegian Consumer Centre (Forbruker Europa) contributes to ELRC by creating and providing data that will support Norwegian in the ODR service. The project is focused on the production of translated texts from consumer enquiries. The first batch of 4400 translated segments are already submitted to ELRC and EC to train ODR-specific CEF eTranslation engine.

6.3 Estonian Open Parallel Corpus

The National Programme for Estonian Language Technology supports the development of Estonian-related language resources and technologies. In 2016, the Estonian Open Parallel Corpus project collected, aligned and published into the EOPC corpus as many as 413 000 English-Estonian and 155 000 Estonian-Russian translation units from websites of Estonian public sector and EU institutions. In 2017, the EOPC project collected a parallel corpus of 226 631 translation units from Estonian to English, French, German and Russian languages from public parallel web sites. The resulting corpora is distributed on the META-SHARE[18] repository.

6.4 Government Translation and Language Services in Finland

The Finnish government has begun an initiative to improve data management in their country[19]. The Translation and Language Services Division (TLD), established in 2015 as part of the Government Administration Department, provides translation, language, and terminology services to all ministries in the Finnish government. Among the duties of TLD is the management of a Translation Memory system and internal termbases, as well as maintenance of the government's online term bank VALTER[20].

TLD has also created a system for the ordering and management of translation requests, called SHAKE. The system manages language resources by storing all documents sent for translation automatically on a network drive, accessible to the entire TLD (all translations, XLIFF files, and background material are stored in the same place). In addition, SHAKE allows users to search for and retrieve earlier translations via automatic archiving and enables the establishment of common practices for the use of TMs and term banks, including via server-based TMs with set attributes that enable the retrieval of "theme-based" translation memories for the use of external service providers.

In effect, the Finnish government is successfully implementing in the public sector a language resource management process that help to manage TMs, leveraging them for use in translation projects, as well as maintain a high degree of order in the management and storage of data.

7. PSI Directive in Practice

The ELRC action shows that public sector institutions are very interested in using MT and they have ample resources to contribute. However, there are multiple barriers that must be overcome to make that happen.

Often translation data is not organized within an institution. Employees of government institutions refer to their website as the ultimate data source. The drawbacks of this approach are that just a small fraction of the data is readily available; it must yet be cleaned and aligned both on a document- and sentence-level.

The statements of the PSI directive are not yet adapted effectively. Often the data is there, but motivation is missing to go the last mile to actually share the data. It takes pleading, clarifying, and sending official letters from EC or ELRC consortium for someone in the institution to take the responsibility to give an internal order to donate the data. Instead, we anticipate proactive sharing of data online so

[18] http://metashare.tilde.com/repository/search/?q=estonian+open+parallel+corpus

[19] http://www.lr-coordination.eu/sites/default/files/Belgium/2017/Brussels_3rd_ELRC_Conference/Taru%20Virtanen_Case%20Study%20Finland.pdf

[20] http://www.valter.fi

that anyone who has interest may find and download the data that they need.

There is a need to raise awareness that written texts produced in public sector is valuable public data, which may be reused for language technology research and development; therefore it must be organized, saved, and made available publicly whenever possible. This may require a common approach on all levels – starting from submitting tenders for translation services; introducing clauses in contracts with translation service providers for submitting TMs as part of delivered data; and the need for common state-wide infrastructure for managing the textual data – monolingual documents, translated documents, TMs, source files prior to their publishing or conversion to PDF.

There is a limited understanding within institutions about their language data assets. Government officials being addressed do not know what data the institution has, and often see data collection activities as an additional burden. Surprisingly, in some cases it turns out that it takes very little effort to export translation memories and to upload them on ELRC-SHARE.

ELRC is actively promoting best practice examples such as data contribution by the Parliament of the Republic of Lithuania or HUGO.LV project in Latvia to encourage other institutions and national governments to follow.

8. Conclusions

The results of the ELRC action have shown that public administrations have valuable language data in their possession. More than 500 participants in ELRC workshops and numerous data contributions demonstrate that public administrations in the region can be effectively engaged in language resource identification and collection. The workshops also showed that the public administrations in Northern Europe have a pressing need for integrating machine translation into public services, as the region is highly multilingual.

At the same time, the languages of the region are extremely complex, with relatively small volumes of available data. However, by continuing to mobilize stakeholders in the region, the ELRC action shows how to identify and gather valuable language resources for improving the quality of MT services for Northern European languages.

65 new Baltic/Nordic language resources were collected in the first phase of the ELRC action. In the second phase by the end of 2017, 72 additional resources were collected or donated by public administrations and are in processing to produce LRs ready for use in MT training.

Acknowledgments

The research leading to these results has received funding from the research project "Competence Centre of Information and Communication Technologies" of EU Structural funds, contract No. 1.2.1.1/16/A/007 signed between IT Competence Centre and Central Finance and Contracting Agency, Research No. 2.2. “Prototype of a Software and Hardware Platform for Integration of Machine Translation in Corporate Infrastructure".

Bibliographical References

Gavrilidou, M., Labropoulou, P., Desypri, E., Piperidis, S., Papageorgiou, H., Monachini, M., Frontini, F., Declerck, T., Francopoulo, G., Arranz, V. and Mapelli, V. (2012). The META-SHARE Metadata Schema for the Description of Language Resources. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012), Turkey.

European Parliament (2003). Directive 2003/98/EC of the European Parliament and of the Council of 17 November 2003 on the re-use of public sector information, Official Journal of the European Union, L 345, 31.12.2003, p. 90–96

Lösch, A., Mapelli, V., Piperidis, S., Vasiļjevs, A., Smal, L., Declerck, T.; Schnur, E., Choukri, K., and van Genabith, J. (2018). European Language Resource Coordination: Collecting Language Resources for Public Sector Information Management. In Proceedings of the 11th Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA)

Oslo Economics (2016). Kartlegging av behovet for automatisk oversettelse i statlig sector, *Utarbeidet for Kommunal- og moderniseringsdepartementet OE-rapport 2016-15*

Piperidis, S. (2012). The META-SHARE Language Resources Sharing Infrastructure: Principles, Challenges, Solutions. Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, pages 36-42.

Rehm G. and Uszkoreit H. (2011). Multilingual Europe: A challenge for language tech. *MultiLingual*, 22(3):5152, April/May 2011.

Wittenburg, P., Bel, N., Borin, L., Budin, G., Calzolari, N., Hajicova, E., Koskenniemi, K., Lemnitzer, L., Maegaard, B., Piasecki, M. and Pierrel, J.M. (2010). Resource and service centres as the backbone for a sustainable service infrastructure. In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010) (pp. 5-9).

Skadiņa, I., Vasiljevs, A., Borin, L., De Smedt, K., Linden, K. and Rognvaldsson, E. (2011). META-NORD: Towards Sharing of Language Resources in Nordic and Baltic Countries. Proceedings of Workshop on Language Resources, Technology and Services in the Sharing Paradigm, Chiang Mai, Thailand, pp. 107-114.

Tiedemann J. (2012). Parallel data, tools and interfaces in OPUS. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Váradi, T., Krauwer, S., Wittenburg P., Wynne, M., Koskenniemi, K. (2008). CLARIN: common language resources and technology infrastructure. Proceedings of the Sixth International Language Resources and Evaluation Conference (LREC 2008).

Vasiļjevs, A., Pedersen, B. S., De Smedt, K., Borin, L., Skadiņa, I. (2011). META-NORD: Baltic and Nordic Branch of the European Open Linguistic Infrastructure. Proceedings of the NODALIDA 2011 workshop on Visibility and Availability of LT Resources, Riga, Latvia, pp. 18-22.

Vasiljevs, A., Rirdance, S. and Liedskalnins. A. (2008). EuroTermBank: Towards Greater Interoperability of Dispersed Multilingual Terminology Data. Proceedings

of the First International Conference on Global Interoperability for Language Resources (ICGL 2008), Hong Kong

Language Resource References

The Árni Magnússon Institute for Icelandic Studies. (2016). The Icelandic Term Bank, accessible online at http://www.malfong.is/index.php?lang=en&pg=idord

Nasjonalbiblioteket Språkbanken. (2016). Translation memories from The Ministry of Foreign Affairs, dowloadable at http://www.nb.no/sprakbanken/show?serial=oai%3Anb.no%3Asbr-36&lang=en

Tilde. (2016). Estonian Open Parallel Corpus, https://www.keeletehnoloogia.ee/en/projects-2011-2017/estonian-open-parallel-corpus-1, distributed via META-SHARE

The Prime Minister's Office Finland. (2016). Hallituskausi 2011–2015, distributed via The Prime Minister's Office

The Prime Minister's Office Finland. (2015). Hallituskausi 2007–2011, The Prime Minister's Office

LIDIOMS: A Multilingual Linked Idioms Data Set

Diego Moussallem[1,2], Mohamed Ahmed Sherif[2],Diego Esteves[1] Marcos Zampieri[3], Axel-Cyrille Ngonga Ngomo[2]
[1]Faculty of Mathematics and Computer Science - University of Leipzig, Germany
[2]Data Science Group - University of Paderborn, Germany
[3]Research Group in Computational Linguistics - University of Wolverhampton, United Kingdom
lastname@informatik.uni-leipzig.de

Abstract

In this paper, we describe the LIDIOMS data set, a multilingual RDF representation of idioms currently containing five languages: English, German, Italian, Portuguese, and Russian. The data set is intended to support natural language processing applications by providing links between idioms across languages. The underlying data was crawled and integrated from various sources. To ensure the quality of the crawled data, all idioms were evaluated by at least two native speakers. Herein, we present the model devised for structuring the data. We also provide the details of linking LIDIOMS to well-known multilingual data sets such as BabelNet. The resulting data set complies with best practices according to Linguistic Linked Open Data Community.

Keywords: multilingual, idioms, translation

1. Introduction

Recently, the Linguistic Linked Open Data (LLOD)[1] movement has gained significant momentum. According to McCrae et al. (2016), a large number of linguistic data sets have been extracted from various sources and been represented as Linked Data (LD). This new movement was motivated by the novel capabilities of the LD paradigm pertaining to transforming, sharing, and linking linguistic data on the Web (Chiarcos et al., 2012). Resources such as dictionaries and knowledge bases are essential in the development of Natural Language Processing (NLP) systems. However, most of these resources are still bilingual on the LLOD. Thus, becoming worthwhile to develop multilingual knowledge bases by reusing these bilingual contents. Multilingualism is important not only for sharing information across Web but also for learning new concepts from other cultures.

There are many data sets and linguistic resources available at LLOD, however, most of them do not contain much information about Multiword Expressions (MWE). MWE are known to constitute a difficult problem on a number of NLP tasks such as machine translation, language generation, and sentiment analysis/opinion mining. There are different types of MWE, according to Nunberg et al. (1994), MWE are categorized as phrase verbs, compounds, fixed expression, semi-fixed expressions, idioms, slang, and others. This work focuses on idioms, a particular type of MWE.

Most idioms are culture-bound and their senses come from particular concepts of everyday life to a given culture. By definition, idioms are a sequence of words whose meaning cannot be derived from the meaning of words that constitute them (Nunberg et al., 1994). Idioms are generally classified as non-compositional. One of the direct consequences of non-compositionality is the impossibility of translating this kind of word group literally (Nunberg et al., 1994) posing challenges to human translators and to machine translation systems.

In this paper, we propose LIDIOMS, a multilingual linked data set of idioms in five languages. In LIDIOMS, we do not distinguish between idioms sub-categories and thus work on idioms in general by providing lexical and semantic knowledge on a multilingual basis. The selected languages are English, German, Italian, Portuguese, and Russian. This choice of languages intends to show the possibility of correct translations among idioms independent of their language family, syntax or culture. Additionally, one of the goals of LIDIOMS is to support further investigations of similarity among idioms from different languages.

In the following, we begin by presenting the related work (Section 2.) and the data sources that we used for the extraction (Section 3.). In Section 4., we give an overview of the model that underlies our data set. Section 5. depicts the creation process that led to the publication of our data set. In Section 6., we present our approach to link LIDIOMS internally and externally. Then, we present usage scenarios for our data set in Section 7.. Subsequently, we discuss LLOD quality in subsubsection 7.4.1. and we conclude the paper and provide avenues for future work in Section 8..

2. Related Work

A large number of ontologies have been developed to represent natural language data as LD on the Web of Data. In this context, the well-known ontology *lemon* (McCrae et al., 2012) was originally developed to model lexical data in mono or multilingual way. Subsequently, a significant amount of effort has been invested in order to improve the support of multilingual contents. To this end, other modules have been extended from *lemon* for representing multilingual data including (Gracia et al., 2014), which extends some of the *lemon* properties describing relationships among translations.

Recently, multilingual data sets have been created such as DBnary (Sérasset, 2012), which was released with the

[1] http://linguistics.okfn.org/

main purpose of describing translations among lexical entries. Another resource that describes multilingual content is BabelNet (Navigli and Ponzetto, 2010), which integrates knowledge from various lexical resources, such as WordNet (Miller, 1995). Additionally, BabelNet has adopted the *lemon* structure for representing lexical entries (Ehrmann et al., 2014). Although these resources are linked lexical multilingual data sets, they contain a limited number of idioms described correctly along with their respective translations across languages. This lack of information about MWE and idioms is due to the missing appropriate ontologies and vocabularies for handling this phenomena properly. Despite *Lexinfo* ontology (Cimiano et al., 2011) contains a certain property just for representing idioms, there are no appropriate classes to reuse this information. Fortunately, the W3C Ontology Lexica Community Group[2] has created an extension of *lemon* called *Ontolex*[3] in order to not only address this lack of information but also to describe more appropriately linguistic terms (Bosque-Gil et al., 2015). Thus, enabling LIDIOMS to represent a particular type of linguistic unit, that is to say idioms. In the following, we present the data set creation process in more detail.

Additionally, a number of multilingual data sets have been published as Linked Open Data (LOD) in the last years. The well-known knowledge base of *DBpedia* (Lehmann et al., 2015) is one of first multilingual knowledge bases extracted from *Wikipedia*[4]. Recently, the *Semantic Quran* data set has published translations of the Quran in 43 different languages as linked data (Sherif and Ngonga Ngomo, 2015). *xLiD-Lexica* (Zhang et al., 2014) is a cross-lingual linked data lexica which is constructed by exploiting all language versions of Wikipedia. *Terminesp* (Bosque-Gil et al., 2015) is another multilingual resource for terms along with their definitions in various languages.

3. Data Sources

In this section, we list the data sources from which LIDIOMS originates, where we describe the data collection process of each data source. In addition, we discuss how we ensure the quality of the collected data.

3.1. Data sets

We collected a set of MWE from the online lexical resources: (1) *Phrase finder*, (2) *Memrise*, (3) *Collins* and (4) *Oxford* dictionaries[5]. *Phrase finder* is an online dictionary about idiomatic expressions created by *Gary Martins* (Martin, 2007) in 1997 for supporting his post-graduate research in computational linguistics. *Memrise* is an online course about idiomatic expressions for achieving a native speaker level. *Collins* and *Oxford* provide high quality lexical resources. Therefore, we use them to guarantee the quality of the idioms definitions and also for gathering some additional idioms. *Memrise* and *Oxford* provided idioms in English, German, Italian, and Russian languages, while the idioms in *Phrase finder* and *Collins* are in English. The Portuguese idioms were initially gathered from *Wikipedia* Portuguese page[5] but because of the limited number of the available Portuguese idioms in Wikipedia, we asked four native speakers (one from Portugal and the other three from Brazil) to add more Portuguese idioms.

For the sake of clarity pertaining to the copyrights to use the data, *Memrise* and *Collins* granted us a full permission while the others data providers have a free licence policy when to use the data for research purposes.

3.2. Data Collection process

Using a custom web crawler, we collected the MWE from these aforementioned on-line data sources. Each of the crawled resources has specific pages about each MWE, which ease the configuration of our crawler. Note that, all data sources are bilingual but not necessarily including English as one of the involved languages. For instance, *Oxford* has idiomatic expressions from Italian to Portuguese. We also noticed that most on-line dictionaries does not correctly categorize MWE. For example, in some cases the meaning of MWE can be deduced from the meaning of their components (e.g. *"by the book"*) while in other cases this is not possible (e.g. *"out of the blue"*). Therefore, MWE which can be represented by the meaning of their components should not be into the same category as the others with pragmatic meanings (i.e. non-compositional idioms). Collecting the right idioms was a hard task due to the lack of MWE categorization. Thus, we carried out the idiom collection manually where we discarded all the entries that were semantically equivalent to their lexical definitions which means to be *not* non-compositional. We dubbed this process *pragmatically-based selection*. The pragmatically-based selection identified only 50% of the MWE retrieved by our crawler as idioms. For instance, the idiom mentioned before *"by the book"* means "to follow the rules as demand". The meaning of "book" is "a stuff which contains information, rules, descriptions, and it can be a manual". Therefore, this MWE is deductible from the meaning of each of its components, the meaning gets "to follow the book's writing". Therefore, it is not considered an idiom, in contrast of the idiom *"out of the blue"* which means "an event that occurs unexpectedly", the meaning of "blue" is "color" then no relationship exists between "blue" or "out of" with "unexpected happening".

Moreover, considering that the meaning of idioms may vary according to the geographical location where they are used (Martin, 2007). For example, American idioms which come from The United States of America differ from the British idioms which come from United Kingdom. We consider the location of idioms as an important characteristic to be included in LIDIOMS.

3.3. Data Evaluation

To ensure the quality of the retrieved data, we asked two native speakers and one linguist (per language) to evaluate the extracted idioms and their respective definitions in English. For evaluating an idiom, each native speaker separately evaluated the idioms' definition. Idioms with accepted definitions by both evaluators are accepted. Also,

[2] https://www.w3.org/community/ontolex/

[3] https://www.w3.org/community/ontolex/wiki/Final_Model_Specification

[4] https://www.wikipedia.org/

[5] All repositories web pages http://faturl.com/repositories/?open

idioms with idioms' definitions marked as wrong by both evaluators were discarded. In case a mismatch evaluation happens, the idiom was judged by the linguist. This procedure resulted in a manually checked data set containing a large number of idioms as shown in Table 1. The *Collection* column shows the number of all MWE retrieved by our web crawler. The *Filter* column shows the number of idioms retrieved based on our pragmatically-based selection, a step which recognizes only idioms among MWE (see Section 3.2.). The *Total* column presents the resulting number of idioms after the manual review process made by the natives and the linguist.

Language	Collection	Filter	Total
English	1230	600	291
Portuguese	600	215	114
Italian	500	284	175
German	400	245	130
Russian	220	150	105

Table 1: Number of idioms retrieved by step

4. Semantic Representation Model

The representation model of LIDIOMS aims at describing idioms correctly as a sub-type of MWE together with their translations and geographical usage area. For this purpose, LIDIOMS data set is based on Ontolex model. We chose the *Ontolex* model because it contains the necessary classes to represent MWE and its translations properly. *Ontolex* also reuses the well-known *Lexinfo* ontology which has an essential term type called `lexinfo:idiom` for representing idioms as one type of MWE.

We used the core Ontolex's classes to model LIDIOMS, where (1) we use the class `ontolex:LexicalEntry` for representing a lexical entry (i.e. a word, a multi-word expression or an affix), (2) the sub-class `ontolex:MultiwordExpression` is used to specify a lexical entry as a multi-word expression, (3) the `ontolex:LexicalConcept` class suits perfectly for representing idioms meaning as its formal definition comprises of "to be a mental abstraction, concept or a thought that can be described by a given collection of senses". (4) the `ontolex:LexicalSense` class for lexical sense of an idiom. (5) the `ontolex:Form` class describes the written and alternative forms of the entries and (6) `ontolex:Lexicon` class is used for representing a collection of lexical entries.

For translations, *Ontolex* uses the *vartrans* module which connects `ontolex:LexicalSense` instances among themselves through `vartrans:Translation` class. The `vartrans:Translation` uses the property `vartrans:category` for describing translations and also representing variations of these translations across entries in the same[6] or different languages. The `vartrans` module was inspired by (Gracia et al., 2014) and we also reuse one of its translation categories called `trcat:culturalEquivalent` which represents a translation between two entries that are not semantically but pragmatically equivalent. Note that a cultural translation of an idiom is not a literal translation, rather it represents the specific cultural semantics of that idiom.

For the geographical area of idioms, we use the `lexvo:usedIn` class from the *Lexvo* Ontology (de Melo, 2015). The geographical area of an idiom is of great importance because the meaning of an idiom can vary in the same language depending on where it is used (diatopic variation). For instance, the Portuguese idiom *"amarrar o burro"*(its literal translation: "tie the donkey") means "to relax" in Portugal while in Brazil it means "to advise someone about future problems from one action". Furthermore, this idiom has also more meanings even within Brazil, for example, "to be angry when someone does not allow you to do something" that is typical for children. In addition, some idioms are not understood in all countries even sharing the same language. For instance, the Portuguese idiom *"comprei um mamao"* (eng: "buy a lemon") is used in Brazil but not in Portugal.

In Figure 1, we present a complete example of a translation of two idioms from Portuguese ("custa os olhos da cara") to English ("arm and a leg") using *vartrans* class along with the others descriptions modeled by *Ontolex* in LIDIOMS.

In order to represent the names of the languages in a unified way, we publish LIDIOMS based on the best practices of the International Organization of Standardization (ISO). Given the fact that Brazilian Portuguese does not have an ISO resource, we chose to use the Brazilian Portuguese DBpedia resource[7] for substituting that missing ISO.

5. RDF Generation

The original idioms were crawled in heterogeneous formats such as CSV, XML, and HTML. To convert the idiom data into RDF, we used *OpenRefine*[8] together with its RDF extension. The model underlying the RDF conversion relies on the group of patterns to generate linguistic resources as LD recommended by the Best Practices for Multilingual Linked Open Data (BPMLOD) W3C community group[9]. In spite of our work being multilingual, we followed the patterns for bilingual dictionaries[10]. We were able to use bilingual patterns because we use English as pivot language given that all the target translations are in English. Thus, the multilingual translations were found by inference relying on the reflexivity property of the `vartrans:target`. For more details about LIDIOMS see Table 2 and visit LIDIOMS GitLab repository[11].

6. Linking

In this section, we describe how we link idioms in LIDIOMS internally (i.e. within the data set) and externally (i.e. with other data sets).

[6] Same entry from a given language with different meanings

[7] `http://dbpedia.org/page/Brazilian_Portuguese`

[8] `http://openrefine.org/`

[9] `https://www.w3.org/community/bpmlod/`

[10] `https://www.w3.org/2015/09/bpmlod-reports/bilingual-dictionaries/`

[11] `https://github.com/dice-group/LIdioms`

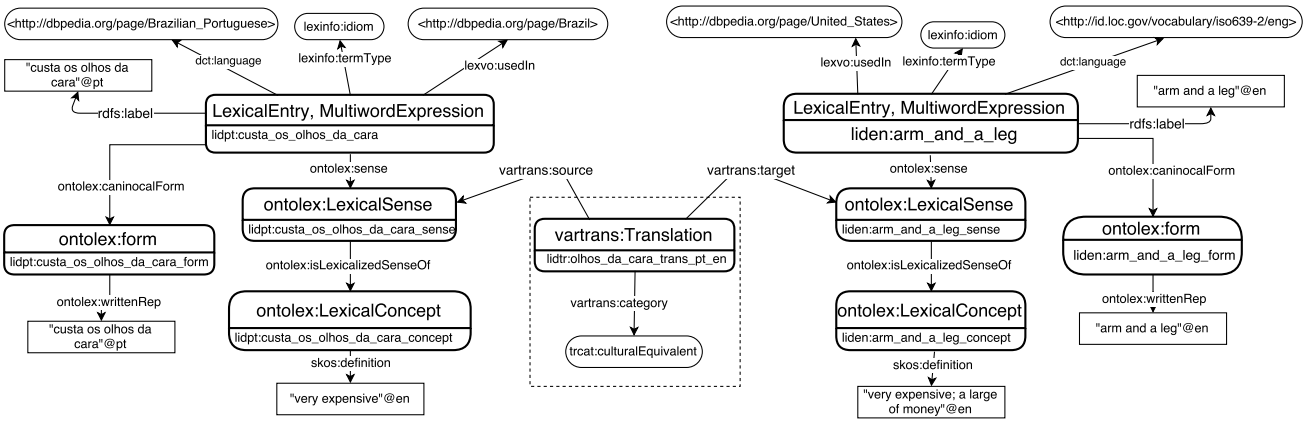


Figure 1: RDF representation of translation of two idioms from Portuguese ("Custa os olhos da cara") to English ("arm and a leg") by entries modeled with the LIDIOMS model.

Name	LIDIOMS
Example	`http://lid.aksw.org/en/kill_two_birds_with_one_stone`
Dump	`http://lid.tabsolucoes.com/dataset/dump-1-0.tar.gz`
Sparql	`http://lid.aksw.org/sparql`
Repository	`https://datahub.io/dataset/lidioms`
Ver. Date	20.04.2017
Ver.No	1.0
License	CC BY-NC-SA 3.0

Table 2: Technical Details LIDIOMS.

6.1. Internal linking

While most of the definitions of the retrieved idioms were in English (87%), only in a few cases the definition was in another language. We then decided to provide the definitions of all idioms in English regardless of the idioms' original language. The other 13% of idioms which had the definitions in another language were translated by a native speaker to English. Therefore, the English definitions became our pivot language, i.e. the idioms' English definitions were used as bridge for the internal linking process across languages. For instance, the *"when pigs fly"* English idiom has the definition "something that will never happen". In Portuguese, the idiom *"nem que a vaca tussa"* has exactly the same lexical definition, but its literal translation would be "nor the cow cough". Still, it is valid to decide to link these two idioms internally based on their definitions. Figure 2 illustrates the main idea underlying this work, i.e. the provision of indirect translations (represented by dotted line) of idioms through a pivot language.

Note that, some idioms have multiple idiomatic equivalents in other languages while others have none. However, some idioms have definitions with almost equivalent syntactic structures while the semantics of the definitions are very different. For instance, the English idiom *"Once in a blue moon"* means *"something that happens rarely"* and another English idiom *"When pigs fly"* means *"something that will never happen"*. This kind of phenomena is likely to decrease the quality of an automatic linking process, because current link discovery frameworks (Nentwig et al., 2015) only support syntax-based string similarities. Given the lack of support of semantic-based string similarity functions, the internal linking was carried out *manually* by the authors and a cross-validation among the natives and linguists were done on this manual internal linking.

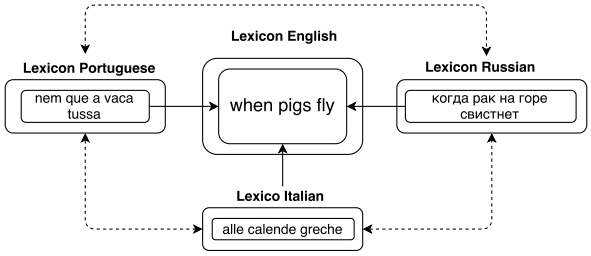


Figure 2: An indirect translation excerpt

The Table 3 shows the number of direct and indirect translations found for the selected idioms per language.

	EN	PT	IT	DE	RU	Total
Idioms	291	114	175	130	105	815
Translations	192	79	73	62	82	488

Table 3: Number of idioms and Translations

6.2. External linking

Linking LIDIOMS to other external resources is based on the string similarities between LIDIOMS's resources and the other data sets' resources. The current version of LIDIOMS is linked to two other data sets in order to ensure reusability and integrability.

The first data set we linked to LIDIOMS is *DBnary*. We used the algorithms provided in LIMES (Ngomo, 2012; Sherif and Ngonga Ngomo, 2015) framework which are time-efficient to carry out the DBnary linking tasks. The linking was through `rdfs:label` property using the `trigram` similarity with acceptance threshold 0.85.

Languages	LIDIOMS	BabelNet Retrieval	BabelNet Accepted	BabelNet Precision	DBnary Retrieval	DBnary Accepted	DBnary Precision
English	291	600	195	0.325	362	323	0.892
Portuguese	114	23	9	0.391	26	4	0.153
Italian	175	52	33	0.634	4	4	1.0
German	130	27	8	0.296	45	45	1.0
Russian	105	48	16	0.333	0	0	0
Total	**815**	**750**	**261**		**437**	**384**	

Table 4: Number of links and precision values obtained between LIDIOMS and other data sets.

The second data set we linked with LIDIOMS is *BabelNet*. The BabelNet linking process was carried out using the BabelNet API[12] to retrieve senses and definitions. While linking, we noticed that *BabelNet* do not correctly type idioms (more details see Section 7.4.1.). We thus linked to BabelNet manually by comparing our `skos:definition` property with the `bn-lemon:definition` property of the BabelNet resources. This task was performed by the same group of linguists previously requested.

6.3. Linking Quality

In this section, we show and discuss the linking statistics of LIDIOMS with BabelNet and DBnary.

Table 4 presents the number of links per resource and language in the LIDIOMS data set. Note that all the links were evaluated manually. The *Retrieval* columns show the number of total idioms collected from a given data set and the *Accepted* columns present the number of idioms which were matched exactly as an idiom. We also present the precision achieved by the aforementioned link specifications.

DBnary has presented a good precision in general. Its lower score only comes from Portuguese and Russian as these languages are a bit exploited by linguistic resources in terms of MWE thus containing only few idioms. *DBnary* follows the best practice of publishing linked data which means without any typos in labels (e.g `rdfs:label`) in contrast of *BabelNet* (see [13]. This problem contributes for the lower precision score of *BabelNet* because its API does not handle it instead of LIMES.

7. Use Cases

In this section, we outline selected application scenarios for our data set. Listing 1, Listing 2 and Listing 3 illustrate different facets of how LIDIOMS can support translation use cases. LIDIOMS contains a significant number of instances of concepts, places and translations. Thus, multilingual idioms along with their definitions concerning about a specific information can be easily retrieved from our data set. Moreover, the aligned multilingual representation allows searching for idioms with the same meaning across different languages.

7.1. Gathering idioms by definitions

The first use case for our data set is exploratory in nature. Machine translation agents are commonly in need of expressions that have a certain meaning. Using a simple SPARQL query over LIDIOMS enables these potential agents to easily find idioms which contain a keyword of choice. For example, Listing 1 shows a SPARQL query for retrieving English, Italian and Russian idioms which contains the verb *"to deceive"* in their definitions.

```
SELECT ?label ?definition
WHERE {
      ?idiom rdfs:label ?label.
      ?idiom ontolex:sense ?sense.
      ?sense ontolex:isLexicalizedSenseOf ?concept.
      ?concept skos:definition ?definition.
FILTER(bif:contains(?definition, "deceive")) .
FILTER( lang(?label) = "it" || lang(?label) = "en" ||
     lang(?label) = "rus" ).}
```

Listing 1: Idioms definitions that contains the same verb in (i) English (ii) Italian and (iii) Russian.

7.2. Idioms usage per area

LIDIOMS provides information about the place of usage of each idiom. For instance, the idiom *"it's raining cats and dogs"* has English as its language property and comes from England. By being aware of the place of origin of an idiom, translators are now empowered to translate an idiom to the right idiom for a given target group. Listing 2 shows a SPARQL query which retrieves all idioms from England.

```
SELECT ?idiom ?label
WHERE {
  ?idiom rdfs:label ?label;
         lexvo:usedIn dbr:England .
}
```

Listing 2: All idioms coming from England.

7.3. Translating across languages

Another important use of LIDIOMS is to retrieve *indirect translations*. By indirect translation we mean a translation which is based on another translation. Nevertheless, the power of RDF representation of LIDIOMS enable the induction of indirect translations through the English translations. For example, the SPARQL query in Listing 3 first finds the English translation of the German idiom *"Zwei Fliegen mit einer Klappe schlagen"*, then it retrieves Russian idioms with equivalent English translations.

[12] `http://babelnet.org/guide`

[13] `http://babelnet.org/rdf/page/once_in_a_blue_moon_r_EN`

```
SELECT ?idiom
WHERE {
  ?i rdfs:label "zwei fliegen mit einer klappe schlagen"
       @de;
     ontolex:sense ?sense.
  ?trans vartrans:source ?sense;
         vartrans:target ?target.
  ?transind vartrans:target ?target;
            vartrans:source ?source.
  ?lex ontolex:sense ?source;
       rdfs:label ?idiom.
 FILTER( lang(?idiom) = "rus" ).
}
```

Listing 3: Indirect translation.

7.4. Third-party uses: Retrieving More Information through Links

LIDIOMS is linked to other data sets, from which we are able to retrieve additional idiom-related information. For example, Listing 4 shows a SPARQL query for retrieving a given part-of-speech tag of the English idiom *"out of the blue"* from the same resource exists in *DBnary*.

```
SELECT ?pos
   WHERE {
     ?idiom rdfs:label "out of the blue"@en;
            owl:sameAs ?ext_idiom.
     SERVICE <http://kaiko.getalp.org/sparql> {
       SELECT ?ext_idiom ?pos
       WHERE{
         ?ext_idiom dbnary:partOfSpeech ?pos
       }
   }
}
```

Listing 4: Retrieving data from different resources.

7.4.1. Discussion

A main limitation in the currently available data sets in LLOD is the lack of proper categorization of MWE. For example, neither *BabelNet* nor *DBnary* have specific MWE types. For instance, in *BabelNet*, some idioms were not typed as lexical entries, we were capable of finding exact matches of many idioms which are included in LIDIOMS but the matches were from other classes such as a film, a book or music album (e.g., "head over heels" is the label of a film[14]). In order to alleviate this problem, we also tried to filter the idioms by *bn-lemon:synsetType* in BabelNet, however, incorrect types avoided us to link them easily. For example, the idiom "The Goose That Laid the Golden Eggs" is typed as "Named Entity" (see http://babelnet.org/-rdf/page/s03200922n), but it should be a concept. Additionally, Listing 5 shows an example resource from BabelNet.

In Listing 5, the idiom *"arm and a leg"* is represented as a *noun* while it should be firstly represented as a MWE or more precisely as an idiom. This lack of accurate categorization of MWE makes linking data sources such as LIDIOMS with other resources very difficult. In particular, using declarative link discovery frameworks for computing similarities among MWE without the right classification information becomes a slow task which leads to links with a low level of precision.

[14] http://babelnet.org/rdf/page/s03412613n

```
bn:arm_and_a_leg_n_EN
  a lemon:LexicalEntry ;
  rdfs:label "arm_and_a_leg"@en ;
  lemon:canonicalForm <http://babelnet.org/rdf/
       arm_and_a_leg_n_EN/canonicalForm> ;
  lemon:language "EN" ;
  lemon:sense <http://babelnet.org/rdf/arm_and_a_leg_EN/
       s13676929n> ;
  lexinfo:partOfSpeech lexinfo:noun .
```

Listing 5: Fragment of a *BabelNet* resource.

Furthermore, this incomplete categorization exists also in other data sets such as *DBpedia* and *DBnary*. We thus regard LIDIOMS as a first effort towards a better LLOD, where MWEs (especially idioms) are represented as such. We envision that this better representation will lead to qualitative linked-data driven NLP systems, including but not limited to better Machine Translation (MT) applications.

8. Summary

In this paper, we described LIDIOMS, a multilingual Resource Description Framework (RDF) data set containing idioms represented in five languages. The data set fills an important gap on MWE processing and it can be used as a resource in NLP pipelines. The current version of LIDIOMS contains $13,889$ triples modeling 815 concepts with 488 translations (115 indirect translations) coming from 7 different sources and linked to 645 external resources. LIDIOMS connects idioms from different languages that have semantically equivalent definitions. To ensure interoperability with other data sets on the LLOD, LIDIOMS is linked to BabelNet and DBnary.

8.1. Future Work

We are currently working to extend the coverage of LIDIOMS so that researchers and developers who work on languages not currently present in the data set can benefit from it. Future versions of LIDIOMS will include idioms from other languages such as Arabic, Chinese, Korean, Czech, Finnish, and French. Moreover, to handle diatopic language variation, the current languages of LIDIOMS are being updated including more fine-grained locations (e.g., cities) as geographical area of use for idioms with more than one meaning even sharing the same country and language. Finally, we plan to improve the automation of the process of internal as well as external linking of idioms by implementing an approach for semantically linking idioms' definitions.

Acknowledgements

We would like to thank all native speakers which contributed with this work. Specifically *Maria Sukhareva* from the *Olia* Project[15], *Will Hanley*[16], *Chiara Pace* and *Devayani Bhave*. Special thanks to *John McCrae*, *Jorge Gracia* and *Gilles Sérasset* for their constructive discussions about modelling and best practices to create our data set.

[15] http://acoli.cs.uni-frankfurt.de/resources/olia/

[16] http://history.fsu.edu/People/Faculty-by-Name/Will-Hanley

This work has been supported by the H2020 project HOBBIT (GA no. 688227) and supported by the Brazilian National Council for Scientific and Technological Development (CNPq) (no. 206971/2014-1). This research has also been supported by the German Federal Ministry of Transport and Digital Infrastructure (BMVI) in the projects LIMBO (no. 19F2029I), OPAL (no. 19F2028A) and GEISER (no. 01MD16014E) as well as by the BMBF project SOLIDE (no. 13N14456).

Bibliographical References

Bosque-Gil, J., Gracia, J., Aguado-de Cea, G., and Montiel-Ponsoda, E. (2015). Applying the ontolex model to a multilingual terminological resource. In *The Semantic Web: ESWC 2015 Satellite Events*, pages 283–294. Springer.

Chiarcos, C., Nordhoff, S., and Hellmann, S. (2012). *Linked Data in Linguistics*. Springer.

Cimiano, P., Buitelaar, P., McCrae, J., and Sintek, M. (2011). Lexinfo: A declarative model for the lexicon-ontology interface. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(1):29–51.

de Melo, G. (2015). Lexvo. org: Language-related information for the linguistic linked data cloud. *Semantic Web*, 6(4):393–400.

Ehrmann, M., Cecconi, F., Vannella, D., McCrae, J., Cimiano, P., and Navigli, R. (2014). Representing multilingual data as linked data: the case of BabelNet 2.0. In *LREC*, pages 401–408.

Gracia, J., Montiel-Ponsoda, E., Vila-Suero, D., and Aguado-de Cea, G. (2014). Enabling language resources to expose translations as linked data on the web. In *LREC*, pages 409–413.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Martin, G. (2007). The phrase finder. *Phrases, sayings and idioms*, 5.

McCrae, J., Aguado-de Cea, G., Buitelaar, P., Cimiano, P., Declerck, T., Gómez-Pérez, A., Gracia, J., Hollink, L., Montiel-Ponsoda, E., Spohr, D., et al. (2012). Interchanging lexical resources on the semantic web. *Language Resources and Evaluation*, 46(4):701–719.

McCrae, J., Chiarcos, C., Bond, F., Cimiano, P., Declerck, T., de Melo, G., Gracia, J., Hellmann, S., Klimek, B., Moran, S., et al. (2016). The open linguistics working group: Developing the linguistic linked open data cloud. In *LREC*.

Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Navigli, R. and Ponzetto, S. P. (2010). BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 216–225. Association for Computational Linguistics.

Nentwig, M., Hartung, M., Ngonga Ngomo, A.-C., and Rahm, E. (2015). A survey of current link discovery frameworks. *Semantic Web*, pages 1–18.

Ngomo, A.-C. N. (2012). On link discovery using a hybrid approach. *Journal on Data Semantics*, 1(4):203–217.

Nunberg, G., Sag, I. A., and Wasow, T. (1994). Idioms. *Language*, pages 491–538.

Sérasset, G. (2012). Dbnary: Wiktionary as a lemon-based multilingual lexical resource in RDF. *Semantic Web Journal-Special issue on Multilingual Linked Open Data*.

Sherif, M. A. and Ngonga Ngomo, A.-C. (2015). Semantic Quran. *Semantic Web*, 6(4):339–345.

Zhang, L., Färber, M., and Rettinger, A. (2014). xlid-lexica: Cross-lingual linked data lexica. In *LREC*, pages 2101–2105.

Annotating Modality Expressions and Event Factuality for a Japanese Chess Commentary Corpus

Suguru Matsuyoshi[1], **Hirotaka Kameko**[2], **Yugo Murawaki**[3], **Shinsuke Mori**[4]
[1]Graduate School of Informatics and Engineering, The University of Electro-Communications
[2]Graduate School of Engineering, The University of Tokyo
[3]Graduate School of Informatics, Kyoto University
[4]Academic Center for Computing and Media Studies, Kyoto University
[1]Chofugaoka, Chofu, Tokyo, Japan [2]Hongo, Bunkyo-ku, Tokyo, Japan
[3,4]Yoshida-honmachi, Sakyo-ku, Kyoto, Japan
[1]matuyosi@uec.ac.jp, [2]kameko@logos.t.u-tokyo.ac.jp, [3]murawaki@i.kyoto-u.ac.jp, [4]forest@i.kyoto-u.ac.jp

Abstract

In recent years, there has been a surge of interest in the natural language processing related to the real world, such as symbol grounding, language generation, and nonlinguistic data search by natural language queries. We argue that *shogi* (Japanese chess) commentaries, which are accompanied by game states, are an interesting testbed for these tasks. A commentator refers not only to the current board state but to past and future moves, and yet such references can be grounded in the game tree, possibly with the help of modern game-tree search algorithms. For this reason, we previously collected *shogi* commentaries together with board states and have been developing a game commentary generator. In this paper, we augment the corpus with manual annotation of modality expressions and event factuality. The annotated corpus includes 1,622 modality expressions, 5,014 event class tags and 3,092 factuality tags. It can be used to train a computer to identify words and phrases that signal factuality and to determine events with the said factuality, paving the way for grounding possible and counterfactual states.

Keywords: game commentary, modality, factuality, symbol grounding

1. Introduction

The field of natural language processing (NLP) has experienced a resurgence of interest in the *symbol grounding* problem (Harnad, 1990). An increasing number of datasets available align natural language expressions with real world objects in the form of images and videos (Hashimoto et al., 2014; Mori et al., 2014b; Ferraro et al., 2015), and systems built on top of such datasets typically perform image/video-to-text generation (Ushiku et al., 2011; Yang et al., 2011).

These systems are, however, more akin to great apes than to humans in the sense that "[t]heir lives are lived entirely in the present" (Donald, 1991). While there is no evidence that nonhuman animals communicate past episodes and planned future events, human language is abundant with them (Szagun, 1978). It is even suggested that the faculty of language has a close connection to the ability to image the future (Suddendorf and Corballis, 1997).

Here we argue that commentaries for an extensive-form game (e.g., chess) are an ideal testbed for developing truly intelligent systems. Specifically, we focus on *shogi* (Japanese chess). Unlike typical image captions, human comments on *shogi* games are full of references to past and future moves as we will see in Subsection 2.2. Yet, thanks to the well-definedness of the world, many of such references can be grounded in a game tree. Although ambiguities inherent in natural language remain a challenging problem, modern game-tree search algorithms (Tsuruoka et al., 2002) help distinguish realistic states from unrealistic ones. For this reason, we have collected *shogi* commentaries together with the corresponding board states and have been developing a game commentary generator (Mori et al., 2016; Kameko et al., 2015).

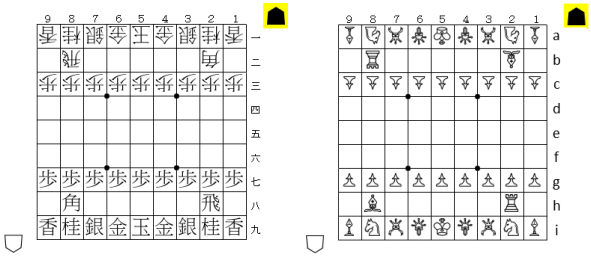


Figure 1: Starting setup of *shogi* (left: normal depiction, right: chess-like depiction).

A human commentator usually expresses to *shogi* fans the degree of confidence on the future move he is describing, which reflects his evaluation of the game-tree. Since confidence is expressed through a wide variety of words and phrases, which we call *modality expressions*, identifying such expressions and binding the factual statuses to events are necessary steps toward automatic generation of human-like commentaries.

To this end, we annotate a *shogi* corpus for a commentator's confidence in this work. Specifically, we adopt a three-layer annotation scheme: modality expression, event class and factuality. The first layer specifies constituents of a sentence construction conveying a certain level of confidence. The second layer extracts event mentions that can be mapped into a game tree. The last layer denotes confidence level of each event mention.

2. Game and Commentary

2.1. *Shogi*

Shogi is a two-player board game similar to chess as illus-

Tag	Meaning
Tu	Turn
Po	Position
Pi	Piece
Ps	Piece specifier
Mc	Move compliment
Mn	Move name
Me	Move evaluation
St	Strategy
Ca	Castle
Ev	Evaluation: entire
Ee	Evaluation: part
Re	Region
Ph	Phase
Pa	Piece attribute
Pq	Piece quantity
Hu	Human
Ti	Time
Ac	Player action
Ap	Piece action
Ao	Other action
Ot	Other notion

Table 1: *Shogi*-specific named entity tags.

trated in Fig 1. The goal of each player is to capture the opponent's king. Each player moves one piece alternately. The biggest difference from chess is that a player can drop a captured piece back onto the board instead of making a move. For detailed explanation of *shogi*, please refer to (Leggett, 2009).

2.2. *Shogi* Commentary

Professional players and writers give commentaries of professional matches for *shogi* fans. Mori et al. (2016) collected *shogi* commentaries with board states described in *Shogi* Forsyth-Edwards notation (SFEN). The corpus consists of 6,523 matches including 744,327 sentences and 11,083,669 words. Mori et al. (2016) first segmented nine matches, or 2,041 sentences, automatically with a text analyzer KyTea[1] and then corrected word boundaries manually. Finally they manually added to the sentences *shogi*-specific named entities (NEs) tags defined in Table 1. We call the annotated corpus the *Shogi* Game Commentary Corpus (SGC corpus).

In the SGC corpus, we observed that the following contents were mainly expressed with recollection of the past and imagination of the future:

- Reason behind moves.
  この桂馬打ちは飛車取りを狙ったものだ (This drop of knight aims at capturing the opponent's rook.)
  角の仇ですね (The move must be a revenge for his bishop.)
- Prediction of the next moves and strategies.
  美濃囲いが有効だ (Black will build *Mino* castle for making it better.)
  ゴキゲン中飛車との予測は外れた (I expected that white used cheerful central rook strategy, but he did not.)
- Mentioning future directions.
  △１四香とすれば決戦 (White's Lx1d would shift the phase to the end game.)

[1] http://www.phontron.com/kytea/

3. Japanese Modality Expressions

We augment the SGC corpus with annotations of modality expressions and event factuality. Table 2 shows a sample text annotated according to our annotation scheme. In this section, we describe the first layer of modality expressions. *Modality expressions* are words and multi-word expressions which impose a proposition in a sentence on propositional modality and event modality (Palmer, 2001). Here are some examples of Japanese modality expressions (EV denotes an event mention and ME denotes a modality expression):

EX 1 後手は歩を成り捨て$_{EV}$た$_{ME}$ (White sacrifice$_{EV}$-ed$_{ME}$ the pawn in the oppenent's zone.)

EX 2 この試合では居飛車を採用$_{EV}$ するかもしれない$_{ME}$ (White may$_{ME}$ use$_{EV}$ rook stay strategy in the game.)

EX 3 おそらく$_{ME}$ △１四香が良好$_{EV}$ (Probably$_{ME}$ white's Lx1d will be good$_{EV}$.)

EX 4 ▲８五桂の跳ね$_{EV}$ 出しを防い$_{ME}$ だ (The move prevented$_{ME}$ black's Nx8e attack$_{EV}$.)

EX 5 ここで△１四歩と受け$_{EV}$ れば$_{ME}$ 先手はつらい (If$_{ME}$ white chooses$_{EV}$ Px1d, black will have a hard time.)

In EX 1, "た (PAST)" is an auxiliary verb for past tense and indicates that the event is a fact. In EX 2, "かもしれない (may)" is a multi-word expression functioning as an auxiliary verb and expressing the low possibility of the event. In EX 3, "おそらく (probably)" is an adverb which suggests that the event is probable. "防い (prevent)" is a verb used as a modality expression while in EX 4 it is used a main verb and can be seen as an event mention. This counterfactive verb makes it explicit that the event "跳ね (attack)" did not happen. In EX 5, "ば (if)" is a conjunctive particle that leads to a conditional construction. Factuality of event mentions in the hypothetical construction is absolutely uncertain.

In the first layer of our annotation scheme, we mark up modality expressions as described above.

3.1. POS

There have been several studies of detecting Japanese modality expressions (Suzuki et al., 2012; Izumi et al., 2013) (Kamioka et al., 2015). However, they focus on auxiliary verbs and functional multi-word expressions. Modal adverbs and conjunctive particles are largely out of scope of these studies. By contrast, we target all types of modality expressions for mark-up regardless of their parts-of-speech.

Layer																										
String	先手	は	美濃	囲い	が	崩れ	て	い	る	の	で	、	飛車	交換	は	後手	の	得	に	な	り	そう	だ	。		
	black	TOP	*Mino*	castle	NOM	break	have PP			because			rook	change	TOP	white	's	good	to	be		be likely to			
POS	N	P	N	N	P	V	P	V	Sf	P	Aux	Pnc	N	N	P	N	P	N	P	V	Sf	Adj	Aux	Pnc		
NE	Tu-B	O	Ca-B	Ca-I	O	Ao-B	O	O	O	O	O	O	Mn-B	Mn-I	O	Tu-B	O	Ee-B	O	Ao-B	O	O	O	O		
Modality	O	O	O	O	O	MEn-B	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	MEa-B	O	O		
Event class				EVe		EVe		EVf						EVi						EVe						
Factuality				FNc		FPc														FPr						

Table 2: Five annotation layers for a commentary string with a well-formatted board state.

3.2. Tags

We define the following two groups of tags: factuality and time.

3.2.1. Factuality Group

The following five tags concern possibility and polarity.

MEy suggests the target event is factual.

銀の捕獲$_{EV}$ に成功$_{MEy}$ (Black succeeded$_{MEy}$ in capturing$_{EV}$ the silver general.)

MEa suggests the target event is possibly factual.

先手はこのあと居飛車に組む$_{EV}$可能性が高い$_{MEa}$

(Black is likely to$_{MEa}$ use$_{EV}$ static rook strategy.)

ME0 suspends mentioning possibility of the target event.

その間に先手玉に迫る$_{EV}$ 手段があるかどうか$_{ME0}$

(Is there$_{ME0}$ a way to attack$_{EV}$ black's king before the castle is completed?)

MEm suggests the target event is possibly counterfactual.

後手の飛車もあまり$_{MEm}$利い$_{EV}$ ていない (White's rook has$_{EV}$ little$_{MEm}$ effect on it.)

MEn suggests the target event is counterfactual.

△ 9 四歩とはし$_{EV}$なかっ$_{MEn}$ た (He did not$_{MEn}$ take$_{EV}$ Px9d.)

3.2.2. Time Group

We introduce the following three tags from the viewpoint of timeline regardless of polarity.

MEp suggests the target event was in the past.

ここで銀交換$_{EV}$ に応じた$_{MEp}$ (They change$_{EV}$-ed$_{MEp}$ each other's silver general.)

MEf suggests the target event will or will not happen in the future.

先手は将来$_{MEf}$ 的に右辺に玉を囲う$_{EV}$ ことになる (White will eventually$_{MEf}$ besiege$_{EV}$ the opponent's king at the right side of the board.)

MEh suggests the target event is hypothesized or has a condition.

ここで△ 1 四歩と受け$_{EV}$ れば$_{MEh}$ 先手はつらい (If$_{MEh}$ white chooses$_{EV}$ Px1d, black will have a hard time.)

4. Event Classes and Factuality

In this section, we describe the second and third layers of our annotation scheme. In the second layer, we classify event mentions into event classes. The objective of the classification is to distinguish from the others event mentions that can be mapped into a game tree. In the third layer, we mark up the factual statuses of events.

4.1. Event

A sentence conveying information contains not only propositions, but also modality, polarity and the writer's attitude. A proposition is expressed as an event mention in a sentence. Following TimeML (Sauri et al., 2006), we consider *events* a cover term for situations that happen or occur. Propositions describing states or circumstances are also considered as events.

4.2. Event Classes

As mentioned in the FactBank annotation guidelines (Sauri, 2008), it is not preferable to target all event mentions in a text for factuality mark-up. This is because a writer's attitude such as wish, command, permission, question and hypothesis is expressed in a text. An event mention surrounded by such an attitude expression is not suitable for factuality mark-up because the factuality of the event is absolutely uncertain. We believe that these events should be separated from events that are free from a writer's attitude. For this reason, we classify event mentions into several event classes from the viewpoint of attitude in advance of factuality mark-up.

Automatic extraction of event mentions in Japanese text is complicated by the existence of grammaticalized verbs and adjectives. Japanese text includes a non-negligible number of these tokens. We mark up grammaticalized tokens in advance of factuality mark-up.

We define the following two groups of class tags: attitude and grammaticalization.

4.2.1. Attitude Group

The following five classes of tags reflect a writer's attitude.

EVa With wish, request, command or obligation.

多くのファンに楽し$_{EVa}$ んでもらいたい (I wish many fans enjoy$_{EVa}$ this game.)

EVq With interrogative.

先手は桂を取る$_{EVq}$ か (Will black attack$_{EVq}$ white's knight?)

EVi Imaginary event, i.e., in conditional construction or not instantiated.

ここで△１四歩と受け$_{\text{EVi}}$ れば先手はつらい$_{\text{EVi}}$

(If white chooses$_{\text{EVi}}$ Px1d, black will have$_{\text{EVi}}$ a hard time.)

EVp With granting a permission.

サイン会だけの参加$_{\text{EVp}}$ も可能です (You can participate$_{\text{EVp}}$ only in an autograph session.)

EVs Simile or metaphor.

３四の銀をあざ笑$_{\text{EVs}}$ うかのように玉を進行させる

(The king is proceeding as if it ridiculed$_{\text{EVs}}$ the silver general of Sx3d.)

4.2.2. Grammaticalization Group

We introduce the following three classes of tags from the viewpoint of lexical/grammatical usage.

EVe Event mention whose core is a lexical predicate with some arguments.

後手は歩を成り捨て$_{\text{EVe}}$ た (White sacrifice$_{\text{EVe}}$-ed the pawn in the oppenent's zone.)

EVc Constituent of a named entity or a verb phrase whose role is a modifier. While it is not a normal event mention, it can be interpreted as an event mention if required in some tasks such as recognizing textual entailment and information extraction.

遊び$_{\text{EVc}}$ 駒を竜にぶつけた (He made an idle$_{\text{EVc}}$ piece invade a path of the opponent's promoted rook.)

EVf Grammaticalized function word or a constituent of a functional multi-word expression.

封じ手につい$_{\text{EVf}}$ て検討が進んでいる (Examination of [= について (*lit.* be attached$_{\text{EVf}}$ to)] the next move has been progressing.)

4.3. Factuality Tags

In the third layer of our annotation scheme, we mark up a factuality tag for an event mention whose class is EVe in the previous phase.

As factuality tags, we use factuality values proposed in FactBank (Sauri, 2008). However, the value Uu is the sole exception. It is not adopted in our scheme because a predicate which would have Uu has been annotated with an EVq or EVi tag in the previous phase. Consequently, we use the following six tags for factuality mark-up. (P stands for the propositon mentioned in a target event. The corresponding values in FactBank are shown in brackets):

FPc Certainly the case that P. <CT+>

後手は歩を成り捨て$_{\text{FPc}}$ た (White sacrifice$_{\text{FPc}}$-ed the pawn in the oppenent's zone.)

FPr Probably the case that P. <PR+>

おそらく△１四香が良好$_{\text{FPr}}$ (Probably white's Lx1d will be good$_{\text{FPr}}$.)

Tag	Freq.	Ratio
MEy-B	49	3%
MEa-B	224	14%
ME0-B	158	10%
MEm-B	21	1%
MEn-B	269	16%
MEp-B	692	43%
MEf-B	59	4%
MEh-B	150	9%
Total	1,622	100%
EVa	39	1%
EVq	111	2%
EVi	707	14%
EVp	7	0%
EVs	4	0%
EVe	3,092	62%
EVc	293	6%
EVf	761	15%
Total	5,014	100%
FPc	2,646	86%
FPr	233	7%
FPs	35	1%
FNc	140	5%
FNr	34	1%
FNs	4	0%
Total	3,092	100%

Table 3: Frequency and ratio of each tag of the three-layer annotation.

FPs Possibly the case that P. <PS+>

この試合では居飛車を採用$_{\text{FPs}}$ するかもしれない

(White may use$_{\text{FPs}}$ rook stay strategy in the game.)

FNc Certainly not the case that P. <CT−>

角交換$_{\text{FNc}}$ には応じなかった (He was not tempted to change$_{\text{FNc}}$ each other's bishop.)

FNr Probably not the case that P. <PR−>

穴熊に組む$_{\text{FNr}}$ つもりはないだろう (I don't think white will use$_{\text{FNr}}$ *anaguma* castle.)

FNs Possibly not the case that P. <PS−>

△９四歩は指し$_{\text{FNs}}$ づらいかもしれません (Maybe it is hard to take$_{\text{FNs}}$ Px9d.)

5. Annotated Corpus

We explain our annotation process and show some statistics on our annotated corpus. We also report preliminary experiments on modality expression detection and event factuality analysis using it.

5.1. Annotation Process

In annotation process, we annotated the SGC corpus with modality expressions, event classes and factuality in this order. Because a modality expression can be made of more than one word, we adopt the IOB2 tagging style (Sang and Meulder, 2003) for our modality expression layer. The

Tag	Freq.	Precision	Recall	F-measure
MEy	21	0.00	0.00	0.00
MEa	103	0.63	0.68	0.65
ME0	105	0.63	0.60	0.62
MEm	7	1.00	0.14	0.25
MEn	140	0.75	0.71	0.73
MEp	370	0.91	0.92	0.92
MEf	32	0.92	0.34	0.50
MEh	66	0.69	0.47	0.56

Table 4: Results of modality expression detection.

Tag	Freq.	Precision	Recall	F-measure
FPc	1,349	0.64	0.63	0.64
FPr	116	0.34	0.14	0.20
FPs	21	0.00	0.00	0.00
FNc	67	0.50	0.15	0.23
FNr	16	0.00	0.00	0.00
FNs	3	0.00	0.00	0.00

Table 5: Results of event factuality analysis.

annotation was performed by a single annotator. To prepare concise guidelines for annotating our tags for other annotators and examine inter-annotator agreement are future work.

5.2. Some Statistics

Table 3 shows the frequency and ratio of each tag in our three-layer annotation. Our corpus contains 1,622 modality expressions and 3,092 lexical event mentions (EVe). Among them 2,646 are labeled with FPc (factual), while 140 are labeled with FNc (counterfactual). The remaining are judged as being in hedge contexts.

5.3. Experiments

To give a glimpse of how difficult the tasks of modality expression detection and event factuality analysis are, we ran on the corpus a simple tagger with a basic set of features. As described in Subsection 2.2., the SGC corpus consists of nine matches. We used the latest match[2] among them as test data and the others as training data in experiments. The numbers of the sentences in the training and the test data are 968 and 1,072, respectively.

For both of modality expression detection and event factuality analysis, we adopt sequential labeling of tags including the O ("not the case") tag. In the experiments, we used PWNER[3] (Sasada et al., 2015) as a tool for sequential labeling. PWNER is a named entity recognizer, which is composed of two modules. The first module enumerates IOB2 tags with confidences for each word independently from the tags for other words. It uses character strings and character types (Chinese character, *hiragana*, Arabic numeral, and so on) of words surrounding the target word as features. The second module searches for the best tag sequence by using conditional random fields referring to the tag-confidence pairs.

Tables 4 and 5 show the results of modality expression detection and event factuality analysis, respectively. In the tables, "Freq." indicates the number of a target tag in the test data. From Table 4, we can see that the F-measures of the modality expression tags whose frequencies are more than 100 are high although the training data is small in size. We expect that increasing the size of the corpus can lead to higher performance of modality expression detection for *shogi* commentaries. In the event factuality analysis task, we obtained the F-measure of 0.64 for the FPc (factual) tag while the F-measures for the other tags are low, as shown in Table 5. This suggests that event factuality analysis for *shogi* commentaries is a difficult task, although increasing the size of the corpus might be effective. Given the obvious dependency of event factuality analysis on modality expressions, we expect to improve performance of event factuality analysis by incorporating the results of modality expression detection as features.

5.4. Availability

We plan to distribute our corpus, except textdata,[4] in our website `http://www.ar.media.kyoto-u.ac.jp/data/game/home-e.html`. For detailed explanations, readers may visit it.

6. Application

The most important applications of our corpus are text analysis such as modality expression detection and factuality analysis as employed in Subsection 5.3. Below we discuss several future research directions.

6.1. Factuality Analysis

Every sentence in our corpus has the corresponding board state (the state history if necessary) and NE labels. They can assist in event factuality analysis. Automatic simultaneous tagging of NE and factuality might work well.

6.2. Automatic Commentary Generation

With our corpus, we can improve automatic commentary generation (Kaneko, 2012; Kameko et al., 2015). The previous work proposed a two-step approach where identification of characteristic words for the given game state is followed by language model-based generation. With consideration for event factuality in addition to characteristic words, the generator is expected to choose appropriate modality expressions. We can try generation using automatically generated templates (Reiter, 1995; Mori et al., 2014a) or deep learning with NEs in place of dialog acts (Wen et al., 2015).

[2] This is a title match called "*Meijinsen*," which causes longer text than a usual match.

[3] `http://www.ar.media.kyoto-u.ac.jp/tool/PWNER/home.html`

[4] The game records and the commentary sentences are distributed in the website: `http://www.meijinsen.jp` (in Japanese) for a fee. We provide a helper script to download the records and the text at `https://github.com/hkmk/shogi-comment-tools`.

6.3. Game State Retrieval

We can also create a system for game state search by natural language queries. A previous study proposed search for game states by piece positions (Ganguly et al., 2014). NE recognition enables a user to search by strategy names and move evaluation (Ushiku et al., 2017). Factuality analysis allows for fine grained retrieval of game states, for example, the one in which the given strategy was judged good by a commentator, and the one where the strategy was mentioned as an option but was not adopted by the player.

6.4. Symbol Grounding

One of the most interesting research directions is symbol grounding. While symbol grounding of a content word to the world is a straight concept, grounding of a modality expression such as "must" and "may" to images, videos, many other forms of media is an open question. An application of modal logic (Kripke, 1963) to *shogi* game tree as a set of possible worlds can be a solution to grounding of some modality expressions.

We believe that there are many other novel applications including bilingual lexicon acquisition for function words and modality expressions based on symbol grounding (Kiela et al., 2015).

7. Conclusion

In this paper, we described our annotation scheme of modality expressions, event classes, and factuality. We annotated the text of a *shogi* commentary corpus with these tags. The annotated corpus includes 1,622 modality expressions, 5,014 event class tags and 3,092 factuality tags. As illustrated in the example sentences at Subsection 2.2., human language, which is different from great ape communication, is abundant with recollection of the past and imagination of the future. Our work leads to the tasks of automatically identifying modality expressions and binding the factual statuses to events in a text. They are necessary steps toward automatic analysis and generation of human commentaries.

The most interesting characteristics of our corpus is that every commentary is associated with a game state (real world). This will enable NLP and AI researchers to tackle various new problems such as commentary generation, intelligent game state search, and symbol grounding.

8. Bibliographical References

Donald, M. (1991). *Origins of the Modern Mind: Three Stages in the Evolution of Culture and Cognition*. Harvard University Press.

Ferraro, F., Mostafazadeh, N., Huang, T.-H., Vanderwende, L., Devlin, J., Galley, M., and Mitchell, M. (2015). A survey of current datasets for vision and language research. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 207–213.

Ganguly, D., Leveling, J., and Jones, G. J. (2014). Retrieval of similar chess positions. In *Proceedings of the 37th annual international ACM SIGIR conference*, pages 687–696. ACM.

Harnad, S. (1990). The symbol grounding problem. *Physica D*, 42:335–346.

Hashimoto, A., Sasada, T., Yamakata, Y., Mori, S., and Minoh, M. (2014). KUSK dataset: Toward a direct understanding of recipe text and human cooking activity. In *Proceedings of the SixthInternational Workshop on Cooking and Eating Activities*.

Izumi, T., Imamura, K., Asami, T., Saito, K., Kikui, G., and Sato, S. (2013). Normalizing complex functional expressions in Japanese predicates: Linguistically-directed rule-based paraphrasing and its application. *ACM Transactions on Asian Language Information Processing*, 12(3):1–20.

Kameko, H., Mori, S., and Tsuruoka, Y. (2015). Learning a game commentary generator with grounded move expressions. In *Proceedings of the 2015 IEEE Conference on Computational Intelligence and Games*.

Kaneko, T. (2012). Real time commentary system for shogi. In *First Workshop on Games and NLP*.

Kiela, D., Vulić, I., and Clark, S. (2015). Visual bilingual lexicon induction with transferred convnet features. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 148–158.

Kripke, S. A. (1963). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16:83–94.

Leggett, T. (2009). *Japanese chess : the game of shogi*. Tuttle Publishing.

Mori, S., Maeta, H., Sasada, T., Yoshino, K., Hashimoto, A., Funatomi, T., and Yamakata, Y. (2014a). Flowgraph2text: Automatic sentence skeleton compilation for procedural text generation. In *Proceedings of the the Eighth International Conference on Natural Language Generation*, pages 118–122.

Mori, S., Maeta, H., Yamakata, Y., and Sasada, T. (2014b). Flow graph corpus from recipe texts. In *Proceedings of the Nineth International Conference on Language Resources and Evaluation*, pages 2370–2377.

Mori, S., Richardson, J., Ushiku, A., Sasada, T., Kameko, H., and Tsuruoka, Y. (2016). A japanese chess commentary corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*.

Palmer, F. (2001). *Mood and Modality Second edition*. Cambridge University Press.

Reiter, E. (1995). Nlg vs. templates. In *Proceedings of the the Fifth European Workshop on Natural Language Generation*, pages 147–151.

Sang, E. F. T. K. and Meulder, F. D. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Computational Natural Language Learning*, pages 142–147.

Sasada, T., Mori, S., Kawahara, T., and Yamakata, Y. (2015). Named entity recognizer trainable from partially annotated data. In *Proceedings of the Eleventh International Conference Pacific Association for Computational Linguistics*.

Suddendorf, T. and Corballis, M. C. (1997). Mental time travel and the evolution of the human mind. *Genetic, So-*

cial, and General Psychology Monographs, 123(2):133–167.

Suzuki, T., Abe, Y., Toyota, I., Utsuro, T., Matsuyoshi, S., and Tsuchiya, M. (2012). Detecting Japanese compound functional expressions using canonical/derivational relation. In *International Conference on Language Resources and Evaluation*.

Szagun, G. (1978). On the frequency of use of tenses in english and german children's spontaneous speech. *Child Development*, 49(3):898–901.

Tsuruoka, Y., Yokoyama, D., and Chikayama, T. (2002). Game-tree search algorithm based on realization probability. *ICGA Journal*, 25(3):145–152.

Ushiku, Y., Harada, T., and Kuniyoshi, Y. (2011). Automatic sentence generation from images. In *Proceedings of the 19th Annual ACM International Conference on Multimedia*, pages 1533–1536.

Ushiku, A., Mori, S., Kameko, H., and Tsuruoka, Y. (2017). Game state retrieval with keyword queries. In *annual international ACM SIGIR conference*.

Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. (2015). Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 207–213.

Yang, Y., Teo, C. L., III, H. D., and Aloimonos, Y. (2011). Corpus-guided sentence generation of natural images. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*.

9. Language Resource References

Kamioka, Y., Narita, K., Mizuno, J., Kanno, M., and Inui, K. (2015). Semantic annotation of japanese functional expressions and its impact on factuality analysis. In *Proceedings of The 9th Linguistic Annotation Workshop*, pages 52–61.

Sauri, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., , and Pustejovsky, J., (2006). *TimeML Annotation Guidelines Version 1.2.1*. `https://catalog.ldc.upenn.edu/docs/LDC2006T08/timeml_annguide_1.2.1.pdf`.

Sauri, R., (2008). *FactBank 1.0 Annotation Guidelines*. `https://catalog.ldc.upenn.edu/docs/LDC2009T23/annotationGuidelines.pdf`.

Annotating Chinese Light Verb Constructions according to PARSEME guidelines

Menghan Jiang, Natalia Klyueva, Hongzhi Xu and Chu-Ren Huang
The Hong Kong Polytechnic University
University of Pennsylvania
menghan.jiang, natalia.klyueva, churen.huang@polyu.edu.hk
xh@cis.upenn.edu

Abstract

In this paper we present a preliminary study on application of PARSEME guidelines of annotating mutiword expressions to Chinese language. We focus on one specific category - light verb constructions (LVCs). We make use of an existing resource containing Chinese light verbs and examine whether this resource fulfill the requirements of the guidelines. We make a preliminary annotation of a Chinese UD treebank in two steps: first automatically identifying potential light verbs and then manually assigning the corresponding nouns or correcting false positives.

Keywords: verbal multiword expressions, light verb constructions, Chinese, corpus linguistics

1. Introduction

Multiword Expressions (MWEs) present a challenge in a bunch of areas of Natural Language Processing like Machine Translation or Information Extraction. They are idiosyncratic in their nature - the meaning of the whole can not be derived from the meaning of its parts. The exact definition of MWEs varies across different linguistic theories and can not be necessarily universal for all languages.

PARSEME (Savary et al., 2017) is a European project which aims at processing multiword expressions from different perspectives and for various languages. Universal guidelines were created to annotate verbal multiword expressions distinguishing several subtypes unique for a language: idioms, LVCs, verb-particle constructions, inherently reflexive verbs and others (miscellaneous category). Corpora in 18 languages have been annotated which result in a multilingual resource (Savary et al., 2017). One of the long-term plans of the project is to extend the set of (mostly European) languages to Asian languages, including Chinese Mandarin. Chinese linguistic tradition is different from the European which presents a challenge when trying to apply the guidelines created under the European project to Chinese. In this paper we make a pilot study of one particular type of verbal MWEs - LVCs and see how the existing resources for Chinese can be adjusted to the PARSEME annotation schema.

We consider two possible corpora for annotation: the Sinica Balanced Corpus of Modern Chinese(Huang et al., 2000) and Universal Dependencies (Nivre et al., 2017). Finally, we have chosen Chinese Universal Dependencies treebank[1] (wiki data) because it features syntactic annotation in dependency-based format required by the shared task.Unlike other languages such as Czech (Bejček et al., 2017), there is no resources we can use to generate MWEs automatically for Chinese. But we don't do the annotation from scratch as well. We use a list of Chinese light verbs to pre-process the data, and then manual work is performed to assign the corresponding nouns of the MWEs and exclude the false ones.

This paper is structured as follows. In Section 2 we describe which types of verbal MWEs can occur in Chinese, in Section 3 we give more extensive information on Chinese LVCs, and in Section 4 we explore how the annotation guidelines created under the PARSEME shared task can be applied to the already existing resource of Chinese LVCs examining the tests. Section 5 presents the statistical information of the annotation result.

2. Verbal Multiword Expressions in Chinese

Following the concept of VMWEs (Verbal Multiword Expressions) that was set in PRASEME project, we can explore two categories of VMWEs in Chinese: verbal idioms (ID),such as 吃醋 *eat vinegar* 'to be jealous' and light-verb constructions (LVC).

In addition to light verb construction, Chinese **Verb+Obj1+Obj2** construction can also be considered as VMWE. The VO compound (e.g., 帮忙 *bangmang* 'do favor'/入籍 *ruji* 'naturalize') can be used transitively taking an external object, such as 入籍中国 *ruji zhongguo* 'naturalize China'. Furthermore, the external object Obj2 can also be placed between Verb and Obj1 (i.e. Verb+Obj2+Obj1), such as 入中国籍 *ru zhongguo ji* 'to naturalize China'.

In this work we will concentrate on LVCs.

3. Light Verbs in Chinese

In modern Chinese, Light verbs are generally defined as the semantically bleached verbs in the sense that the predicative content mainly comes from its taken complement (Zhu, 1985) while the light verb itself may only serve as a syntactic operator, without containing any eventive information. For example, for the construction 加以讨论 *jiayi taolun* 'to discuss', the predicative information all comes from the complement 讨论 *taolun* 'discuss' while the light verb 加以 *jiayi* 'inflict' adds no semantic to the LVC. The most typically used Chinese light verbs are 进行/加以/做/搞/从事 *jinxing/jiayi/zuo/gao/congshi*'proceed/inflict/do/do/engage'.

[1] http://universaldependencies.org/zh/overview/introduction.html

Chinese Light verbs pose a challenge in both traditional linguistics and computational linguistics because of its syntactic and semantic versatility and its unique distribution different from regular verbs with higher semantic content and selectional restrictions. (Butt, 2010) examines Chinese light verbs in the paper, but deals with directional complements and aspectual markers only, without mentioning any of the more typical usage of light verbs in Chinese as dealt with in literature on LVC in Chinese (e.g., (Zhu, 1985)). Moreover, the Propbank (Xue and Palmer, 2005) also treats light verbs as verb with zero argument. However, the issues is, these previous work cannot be directly transferred to the PARSEME framework, hence we further refer to (Lin et al., 2014). In (Lin et al., 2014), the authors specifically dealt with annotation of light verbs in Chinese, as well as automatic classification of different Chinese light verbs, therefore is the directly relevant resource we can make reference to.

4. Adjusting PARSEME guidelines to the database of Light Verbs

The annotation guidelines of the PARSEME shared task[2] define light verb constructions with several key characteristics. The first one is that LVCs are formed by a verb and a (single or compound) noun, which either directly depends on verb (and possibly contains a case marker or a postposition) (e.g., *give a lecture*), or is introduced by a preposition (e.g., *come into bloom*). The second one is the (single or compound) noun is predicative, often referring to an event (e.g. *to make a decision*) or a state (e.g. *to have fear*). The third characteristic is that the verb is "light", in the sense that it contributes to the meaning of the whole only by bearing morphology: person, number, tense, mood, as well as morphological aspect (perfective/imperfective) in some languages. It may be "light" either per se, or when used in the specific context of the noun.

Based on these three main characteristics, the annotation guideline further proposes five specific tests to help determine whether a MWE is LVC or not. In this section, we are going to discuss how these tests can be applied to the determination of Chinese light verbs. A list of Chinese light verbs has been compiled and collected from previous studies (Hu and Fan, 1995), (Diao, 2004), it contains 34 verbs in total. We also include the 5 light verbs mentioned in (Lin et al., 2014). Each of them is put through the tests to decide whether it is a light verb or not. By doing this, we can also examine to what extent the annotation guidelines apply in Chinese.

4.1. Test 1

Test 1 examines whether the taken complement of the light verb (i.e. the n) is a predicative noun or not. For example, *'pay a visit'*, the *visit* represents an event with two arguments (the visitor and the visitee) while the *cake* in *make a cake* represents a simple noun, without any eventive information. Therefore the former one can continue to the next test while the latter one is excluded. In terms of Chinese data, this standard can also be used to make the distinction. Among all the 34 verbs, 26 (e.g., 进行/加以/予以/开展 *jinxing/jiayi/yuyi/kaizhan* 'proceed/inflict/give/carry out') of them pass test 1 while the other 8 verbs (有/做/搞/干/处　置 *you/zuo/gao/gan/chuzhi* 'proceed/inflict/give/carry out') are ambiguous and need to rely on the context for further decision (i.e. they can take both eventive complements and simple nouns). For example, the light verb 进行 *jinxing* 'proceed' and 加以 *jiayi* 'inflict' can only take eventive noun (either deverbal noun as in 加以研究 *jiayi yanjiu* 'conduct research' or event noun as in 进行比赛 *jinxing bisai* 'to have a competition') under all contexts. While the light verb 做 *zuo* 'do' and 搞 *gao* 'do' can have both light verb (e.g., 做治理 *zuo zhili* 'provide governance'/搞竞赛 *gao jingsai* 'to have competition') and non-light verb usage (e.g., 做蛋糕 *zuo dangao* 'make a cake'/搞形式主义 *gao xingshizhuyi* 'do formalism').

4.2. Test 2

Test 2 is to investigate whether the noun is used in one of its original sense. Examples like 'pay a visit' can pass the test since the noun is literally understood while 'kitten' in 'have kittens' is not used in one of its normal senses. With respect to Chinese light verbs, all of the 34 verbs in our wordlist can pass the test.

4.3. Test 3

Test 3 is to check whether a light verb bears morphology (tense, mood etc.) and adds no semantic that is not already present in a noun, other than pointing to which semantic role is played by verb's subject with respect to noun's predicate. For constructions like 'take a walk', 'make a decision' and 'perform a check', the light verb 'take/make/perform' add no meaning to the whole construction, while 'start' in 'to start a walk' does add an aspectual meaning to the noun. For Chinese light verb, we use a simple test to examine this property. That is, if the light verb can be omitted without changing the proposition/semantic meaning of the construction, we consider the verb itself adds no semantic and is semantically bleached. For 10 out of 34 verbs, this light semantics of the verb is usual (i.e. the verb is used as a pure syntactic operator under different contexts, like 进行 *jinxing* 'proceed', 加以 *jiayi* 'inflict', 开展 *kaizhan* 'carry out', 作出 *zuochu* 'make'). For example, the function of 进行/加以研究 *jinxing/jiayi yan jiu* 'conduct research' is the same as 研究 *yanjiu* 'research'.

For another 10 verbs, they fail in this test in the sense that they do contribute to the construction and cannot be omitted. For example, 禁得起/禁不起诱惑 *jindeqi/jinbuqu youhuo* 'be/not be able to stand temptation' is certainly different from 诱惑 *youhuo* 'temp'. 遭到批评 *zaodao piping* 'be criticized' is also different from 批评 *piping* 'criticize' in meaning, in the sense that the former constructions contains the passive reading. However, for the other 14 verbs, this light verb semantics happens in the context of the particular noun. For example, constructions like 做点心 *zuo dianxin* 'make dessert' and 对环境做整治 *dui huanjing zuo zhengzhi* 'to renovate the environment' can

[2] http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.0/?page=040_Annotation_process_-_decision_tree

both be found in corpus. In the construction 做整治 *zuo zhengzhi* 'to renovate', 整治 *zhengzhi* 'renovate' as a deverbal noun contains the eventive information. 做 *zuo* 'do' in this case can be omitted without change proposition of the construction (e.g., 对环境(做) 整治 *dui huanjing (zuo) zhengzhi* 'environment'), although sometimes the whole construction/sentence needs to be re-written to ensure the grammaticality. In contrast, 点心 *dianxin* 'dessert' is a common NP which refers to a concrete entity and does not contain any predicative information. Therefore the eventive information all comes from the verb 做 *zuo* 'do'. In this case, 做 *zuo* 'do' in 做点心 *zuo dianxin* 'make dessert' may represent a series of actions including 'stir' 'blend' 'knead' 'bake' and so on. Hence it cannot be omitted in the construction.

4.4. Test 4

Test 4 is a syntactic test, aiming at testing whether a NP in which verb's subject becomes a noun's dependent evoke the same event. For example, 'Paul had a walk' and 'Paul's walk' both refer to the same walking event, while 'Paul made a good impression' and 'Paul's impression on his wife' refer to different semantics. In terms of Chinese data, the remaining 24 verbs all pass this test. For example, 政府进行改革 *zhengfu jinxing gaige* 'government carry out reform' and 政府的改革 *zhengfu de gaige* 'government's reform' refer to the same 'reform' activity.

4.5. Test 5

Test 5 is focused on the noun's prohibited argument, aiming at examining whether the (single or compound) noun, in the presence of a verb, prohibit at least one syntactic argument a which it normally licensed in the absence of a verb (except when a is in the whole–part relation with verb's subject). In English, for example, *The Queen paid a visit to the Prime Minister* + *a visit of the Lady to the Prime Minister* →**The Queen paid a visit of the Lady to the Prime Minister*. In other words, the visitor cannot be a modifier of visit. To be specific, the noun *visit* takes two semantic arguments, the visitor and the visited entity, as in 'the visit of the Queen to the Prime Minister'. When used in to *pay a visit*, the semantic argument, *visitor*, is realized as the subject of *to pay* (*The Queen paid a visit to the Prime Minister*), and cannot be realized at the same time within the NP headed by *visit (*The Queen paid a visit of the Lady to the Prime Minister)*.In contrast, *Paul transmitted the advice to his sister* + *Peter's advice* →*Paul transmitted Peter's advice to his sister*. The advice can be complemented by its author. Therefore this one is excluded from light verb construction. For Chinese light verbs, all the remaining 24 verbs can pass the tests. The semantic argument of n cannot be realized as its syntactic dependent, since it is already realized as verb's syntactic dependent instead (usually verb's subject). For example, 上海市进行对税收制度的改革 *shanghai shi jinxing dui shuishouzhidu de gaige* Shanghai proceed for tax system DE reform 'Shanghai carry out reform on tax system' + 财政局的改革 *caizhengju de gaige* Bureau of Finance DE reform Bureau of Finance's reform = *上海市进行对税收制度的财政局的改革* *shanghaishi jinxing dui shuishouzhidu de caizhengju de gaige*Shanghai proceed for tax system DE Bureau of Finance DE reform. The reformer cannot be a modifier of the 改革 *gaige* 'reform'.

It can be summarized as the annotation guidelines are efficient and effective in underlying Chinese data. In terms of the determination of Chinese light verbs, only a small part of the verbs can be considered as light verb per se, while most of them rely on the context. Only under certain context as well as with the co-occurrence of certain nouns, these verbs can be considered as light verb.

5. Tagging Light Verbs in a corpus

Provided the selection of potential candidate of light verbs described above, we use this list to pre-tag the UD Chinese corpus, and manually checked and corrected it afterwards. The UD Chinese corpus is in two parts, namely train set and development set. We finally got 836 hits of light verbs in the train set and 108 hits in the development set. Among them, the most frequent light verb is 有 *you* 'have', which gets 483 and 65 hits in the train set and development set respectively. The second and third frequent light verbs are 进行 *jinxing* 'proceed' and 做 *zuo* 'do'. On the other hand, there are 11 light verbs that didn't appear in the corpus. There are 8 light verbs only appear in the train set.

The pre-annotated data as described above were manually checked as follows:

- If the verb was not LVC, it was substituted by '_'
- If a verb was true LVC, the tag was left and the respective noun was assigned with a tag 'cont' - continuation.

In Table 1 below we summarize the changes to the file (comparing tags before the annotation and after)

corpus	lvc-auto	lvc-corrected	cont-inserted
train	836	176	184
dev	108	13	13

Table 1: Summary for a number of automatically assigned tags (lvc-auto), lvc tags left after the manual correction (lvc-corrected), and the number of nouns tagged as a part of LVC (cont-inserted)

The initial format of data will contain only four attributes: form, lemma, POS tag and an LVC tag. Following is an example of an annotated sentence:

```
此後    此後    NOUN    _
，      ，      PUNCT   _
廣東    廣東    PROPN   _
的      的      PART    _
動蕩    動蕩    ADJ     _
局面    局面    NOUN    _
得到    得到    VERB    lvc
基本    基本    NOUN    _
扼止    扼止    NOUN    cont
。      。      PUNCT   _
```

6. Conclusion and future work

In this paper, we demonstrated a pilot study on annotating light verbs in Chinese Universal Dependencies treebank adjusted to PARSEME annotation guidelines. First, a corpus was tagged with a list of potential light verbs, and then it was manually checked and the respective noun were tagged as noun components of LVCs. For the two sets of UD data – training and dev – 189 LVC instances were annotated in total in about 4000 sentence corpus. This corresponds to the LVC-per-sentence ratio of some other corpora annotated under PARSEME. Out future work will include revising the annotation of LVC and including more light verbs as well as other VMWE constructions. When the whole data is ready we can plan to train a classifier to automatically tag Chinese text with LVCs/VMWEs.

7. Bibliographical References

Bejček, E., Hajič, J., Straňák, P., and Urešová, Z. (2017). Extracting verbal multiword data from rich treebank annotation. In *Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories (TLT 15)*, pages 13–24. Indiana University, Bloomington, Indiana University, Bloomington.

Butt, M. (2010). The light verb jungle: still hacking away. *Complex predicates in cross-linguistic perspective*, pages 48–78.

Diao, Y. B. (2004). Research on delexical verb in modern chinese. *Henan University Press*.

Hu, Y. S. and Fan, X. (1995). Research on verbs. *Dalian: Liaoning Normal University Press*.

Huang, C.-R., Chen, F.-Y., Chen, K.-J., Gao, Z.-m., and Chen, K.-Y. (2000). Sinica treebank: Design criteria, annotation guidelines, and on-line interface. In *Proceedings of the Second Workshop on Chinese Language Processing: Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 12*, CLPW '00, pages 29–37, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lin, J., Xu, H., JIANG, M., and Huang, C.-R. (2014). Annotation and classification of light verbs and light verb variations in mandarin chinese. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 75–82, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (2017). The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, Valencia, Spain.

Xue, N. and Palmer, M. (2005). Automatic semantic role labeling for chinese verbs. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, IJCAI'05, pages 1160–1165, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Zhu, D. X. (1985). Dummy verbs and nv in modern chinese. *Journal of Peking University (Humanities and Social Sciences)*, pages 1–6.

8. Language Resource References

Nivre, Joakim and Agić, Željko and Ahrenberg, Lars and Aranzabe, Maria Jesus and Asahara, Masayuki and Atutxa, Aitziber and Ballesteros, Miguel and Bauer, John and Bengoetxea, Kepa and Bhat, Riyaz Ahmad and Bick, Eckhard and Bosco, Cristina and Bouma, Gosse and Bowman, Sam and Candito, Marie and Cebiroğlu Eryiğit, Gülşen and Celano, Giuseppe G. A. and Chalub, Fabricio and Choi, Jinho and Çöltekin, Çağrı and Connor, Miriam and Davidson, Elizabeth and de Marneffe, Marie-Catherine and de Paiva, Valeria and Diaz de Ilarraza, Arantza and Dobrovoljc, Kaja and Dozat, Timothy and Droganova, Kira and Dwivedi, Puneet and Eli, Marhaba and Erjavec, Tomaž and Farkas, Richárd and Foster, Jennifer and Freitas, Cláudia and Gajdošová, Katarína and Galbraith, Daniel and Garcia, Marcos and Ginter, Filip and Goenaga, Iakes and Gojenola, Koldo and Gökırmak, Memduh and Goldberg, Yoav and Gómez Guinovart, Xavier and Gonzáles Saavedra, Berta and Grioni, Matias and Grūzītis, Normunds and Guillaume, Bruno and Habash, Nizar and Hajič, Jan and Hà My, Linh and Haug, Dag and Hladká, Barbora and Hohle, Petter and Ion, Radu and Irimia, Elena and Johannsen, Anders and Jørgensen, Fredrik and Kaşıkara, Hüner and Kanayama, Hiroshi and Kanerva, Jenna and Kotsyba, Natalia and Krek, Simon and Laippala, Veronika and Lê Hong, Phuong and Lenci, Alessandro and Ljubešić, Nikola and Lyashevskaya, Olga and Lynn, Teresa and Makazhanov, Aibek and Manning, Christopher and Mărănduc, Cătălina and Mareček, David and Martínez Alonso, Héctor and Martins, André and Mašek, Jan and Matsumoto, Yuji and McDonald, Ryan and Missilä, Anna and Mititelu, Verginica and Miyao, Yusuke and Montemagni, Simonetta and More, Amir and Mori, Shunsuke and Moskalevskyi, Bohdan and Muischnek, Kadri and Mustafina, Nina and Müürisep, Kaili and Nguyen, Luong and Nguyen Thi Minh, Huyen and Nurmi, Hanna and Ojala, Stina and Osenova, Petya and Øvrelid, Lilja and Pascual, Elena and Passarotti, Marco and Perez, Cenel-Augusto and Perrier, Guy and Petrov, Slav and Piitulainen, Jussi and Plank, Barbara and Popel, Martin and Pretkalniņa, Lauma and Prokopidis, Prokopis and Puolakainen, Tiina and Pyysalo, Sampo and Rademaker, Alexandre and Ramasamy, Loganathan and Real, Livy and Rituma, Laura and Rosa, Rudolf and Saleh, Shadi and Sanguinetti, Manuela and Saulīte, Baiba and Schuster, Sebastian and Seddah, Djamé and Seeker, Wolfgang and Seraji, Mojgan and Shakurova, Lena and Shen, Mo and Sichinava, Dmitry and Silveira, Natalia and Simi, Maria and Simionescu, Radu and Simkó, Katalin and Šimková, Mária and Simov, Kiril and Smith, Aaron and Suhr, Alane and Sulubacak, Umut and Szántó, Zsolt and Taji, Dima and Tanaka, Takaaki and Tsarfaty, Reut and Tyers, Francis and Uematsu, Sumire and Uria, Larraitz and van Noord, Gertjan and Varga, Viktor and Vincze, Veronika and Washington, Jonathan North and Žabokrtský, Zdeněk and Zeldes, Amir and Zeman, Daniel and Zhu, Hanzhi. (2017). *Universal Dependencies 2.0*.

Savary, Agata and Ramisch, Carlos and Cordeiro, Silvio

Ricardo and Sangati, Federico and Vincze, Veronika and QasemiZadeh, Behrang and Candito, Marie and Cap, Fabienne and Giouli, Voula and Stoyanova, Ivelina and Doucet, Antoine and Adalı, Kübra and Barbu Mititelu, Verginica and Bejček, Eduard and El Maarouf, Ismail and Eryiğit, Gülşen and Galea, Luke and Ha-Cohen Kerner, Yaakov and Liebeskind, Chaya and Monti, Johanna and Parra Escartín, Carla and Kovalevskaitė, Jolanta and Krek, Simon and van der Plas, Lonneke and Aceta, Cristina and Aduriz, Itziar and Antoine, Jean-Yves and Attard, Greta and Azzopardi, Kirsty and Boizou, Loic and Bonnici, Janice and Boz, Mert and Bumbulienė, Ieva and Busuttil, Jael and Caruso, Valeria and Cherchi, Manuela and Constant, Matthieu and Czerepowicka, Monika and De Santis, Anna and Dimitrova, Tsvetana and Dinç, Tutkum and Elyovich, Hevi and Fabri, Ray and Farrugia, Alison and Findlay, Jamie and Fotopoulou, Aggeliki and Foufi, Vassiliki and Galea, Sara Anne and Gantar, Polona and Gatt, Albert and Gatt, Anabelle and Herrero, Carlos and Iñurrieta, Uxoa and Jagfeld, Glorianna and Hnátková, Milena and Ionescu, Mihaela and Klyueva, Natalia and Koeva, Svetla and Kovács, Viktória and Kuzman, Taja and Leseva, Svetlozara and Louisou, Sevi and Lynn, Teresa and Malka, Ruth and Martínez Alonso, Héctor and McCrae, John and de Medeiros Caseli, Helena and Miral, Ayşenur and Muscat, Amanda and Nivre, Joakim and Oakes, Michael and Onofrei, Mihaela and Parmentier, Yannick and Pasquer, Caroline and Pia di Buono, Maria and Priego Sanchez, Belem and Raffone, Annalisa and Ramisch, Renata and Rimkutė, Erika and Rizea, Monica-Mihaela and Simkó, Katalin and Spagnol, Michael and Stefanova, Valentina and Stymne, Sara and Sulubacak, Umut and Tabone, Nicole and Tanti, Marc and Todorova, Maria and Urešová, Zdenka and Villavicencio, Aline and Zilio, Leonardo. (2017). *Annotated corpora and tools of the PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions (edition 1.0)*.

Using English Baits to Catch Serbian Multi-Word Terminology

Cvetana Krstev, Branislava Šandrih, Ranka Stanković, Miljana Mladenović
Faculty of {Philology, Philology, Mining and Geology} University of Belgrade, College for Preschool Teachers
{Studentski trg 3, Studentski trg 3, Djušina 7} Belgrade, Bujanovac, Serbia,
cvetana@matf.bg.ac.rs, branislava.sandrih@fil.bg.ac.rs, ranka.stankovic@rgf.bg.ac.rs, ml.miljana@gmail.com

Abstract

In this paper we present the first results in bilingual terminology extraction. The hypothesis of our approach is that if for a source language domain terminology exists as well as a domain aligned corpus for a source and a target language, then it is possible to extract the terminology for a target language. Our approach relies on several resources and tools: aligned domain texts, domain terminology for a source language, a terminology extractor for a target language, and a tool for word and chunk alignment. In this first experiment a source language is English, a target language is Serbian, a domain is Library and Information Science for which a bilingual terminological dictionary exists. Our term extractor is based on e-dictionaries and shallow parsing, and for word alignment we use GIZA++. At the end of procedure we included a supervised binary classifier that decides whether an extracted term is a valid domain term. The classifier was evaluated in a 5-fold cross validation setting on a slightly unbalanced dataset, maintaining average F-score of 89%. After conducting the experiment our system extracted 846 different Serbian domain phrases, containing 515 Serbian phrases that were not present in the existing domain terminology.

Keywords: aligned texts, word alignment, terminology extraction, electronic dictionaries, morphological inflection

1. Motivation

Terminology is rapidly developing in many research and technological fields. It is very difficult to produce and maintain up-to-date terminology resources, especially for languages for which terminology in many fields is transferred and adapted from other languages. Such is the case for Serbian for which terminological resources in many domains, if existing, tend to be obsolete. Purely manual production of terminological resources is not the solution due to rapid changes both in research fields and corresponding terminology.

The work presented in this paper is motivated by our belief that Natural Language Processing (NLP) resources, methods and tools can help in the development of terminology in the Serbian language. Our work relies on the following presuppositions:

1. Serbian terminology is today transferred mainly from English because English terminology is better developed for many scientific and technological domains than Serbian (in the past from French and German). In (Ananiadou et al., 2012) lexical resources for English obtained grades 4.5–6 for all seven criteria, availability rated as excellent (the highest grade 6). To the contrary, the similar survey for Serbian (Vitas et al., 2012) showed that lexical resources are much less developed – they were rated 1–2.5.
2. Terminology consists mainly of Multi-Word Terms (MWT) (data presented in Subsection 4.2. corroborate this claim).[1]
3. A large portion of MWT terms in Serbian has a limited number of syntactic structures. Namely, 98% of all nominal MWEs in the Serbian general e-dictionary of MWEs has one of 13 different structures (having 2, 3 or 4 components) (Stankovic et al., 2016).

Under these presuppositions we are formulating the following hypothesis:[2]

> On the basis of the bi-lingual, aligned, domain-specific textual resources, the terminological list in the source language and the system for the extraction of terminology-specific nominal phrases (MWT) in the target language it is possible to compile the bilingual aligned terminological list.

2. Related Work

In recent years extraction of bilingual MWTs, and MWEs in general, from bilingual aligned corpora has been exploited by many researchers. Although most of them rely on automatic word alignment they differ both in resources and techniques used and in purpose for which they are compiled. In several cases the bilingual MWE lists are produced in order to improve statistical machine translation (Bouamor et al., 2012; Tsvetkov and Wintner, 2010) or to help developing certain lexical resources in the target language on the basis of the existence of such resource in the source language (e.g. used for the Slovenian WordNet (Vintar and Fišer, 2008)). In some cases, no lexical resources are used (Bouamor et al., 2012), while others rely on the existence of some bilingual lexicon (Tsvetkov and Wintner, 2010). MWEs are identified (in a source or a target language) in various ways: some authors use morphosyntactic patterns on lemmatized and POS-tagged texts

[1] Multiword expressions (MWE) are lexical units composed of more than one word, which are syntactically, semantically, pragmatically, and/or statistically idiosyncratic (Baldwin and Kim, 2010). MWTs are domain-specific MWEs.

[2] In this paper we will call 'source' language a well-resourced language (English), and 'target' language a less-resourced language (Serbian).

(Bouamor et al., 2012; Vintar and Fišer, 2008), while others perform full semantic parsing (Moirón and Tiedemann, 2006). For (Tsvetkov and Wintner, 2010) automatic word alignment is the main source of information for identifying MWEs.

In our case, we are relying on the existence of the lexical resource (terminology) in the source language. In order to align it with MWT in the target language we use neither full parsing (as not available for Serbian) nor we lemmatize and POS-tag text. Instead we use shallow parsing relying on extensive morphological e-dictionaries of Serbian (Cvetana Krstev, Duško Vitas, 2015) that not only helps to identify terminology precisely, but also enables production of correct MWT lemmas and consequently all its inflected forms (which is crucial for various applications, from searching to machine translation).

3. The Design of the System

The System consists of several components developed in C# and Python and interconnected to work in a pipeline. It relies on existing monolingual extraction of MWEs for Serbian implemented in LeXimir (Stankovic et al., 2016) and GIZA++ word alignment, while all other components are newly developed. The overall design of our system (Figure1) is as follows:

1. Input:
   - A sentence-aligned domain-specific corpus involving a source and a target language. We will denote an entry in this corpus with $S(text.align) \leftrightarrow T(text.align)$;
   - A list of terms from the same domain in a source language (both single-word terms (SWTs) and MWTs). We will denote an entry in this list with $S(term.list)$;
   - A list of MWTs extracted from the target part of the aligned corpus having some expected syntactic structure. We will denote an entry from this list with $T(term.extract)$.
2. Processing:
   - Aligning bilingual chunks (possible translation equivalents) from the aligned corpus. We will denote aligned chunks with $S(align.chunk) \leftrightarrow T(align.chunk)$;
   - Filtering the chunks to those in which a source part of a chunk matches a term from a list of domain terms in a source language: $S(align.chunk) \sim S(term.list)$, where symbol $\sim$ denotes the relation "match" (that is for our experiment defined in Subsection 4.5.);
   - Filtering once more previously filtered chunks to those in which a target part of a chunk matches a term from a list of extracted MWTs in a target language: $T(align.chunk) \sim T(term.extract)$;
3. The result: the list of filtered chunks that pass a certain threshold linked to matching source and target terms: $S(term.list) \leftrightarrow T(term.extract)$, where $(S(term.list) \sim S(align.chunk)) \wedge (T(term.extract) \sim T(align.chunk)) \wedge (S(align.chunk) \leftrightarrow T(align.chunk))$.

In order to test our approach and determine a threshold we have used the existing bi-lingual terminology resource in order to establish:

- How many aligned chunks after two-pass filtering have a target part that matches a target term in the bi-lingual resource: $(S(term.list) \leftrightarrow T(term.list)) \wedge (T(align.chunk) \sim T(term.list))$ (correspondence between terms in the bi-lingual terminology resource confirmed);
- How many aligned chunks after the first filtering have a target part that matches a term from the list of extracted terms in a target language and does not match a target term in the bi-lingual resource: $(T(align.chunk) \sim T(term.extract)) \wedge (T(align.chunk) \not\sim T(term.list))$ (a potentially new bi-lingual terminological terms);
- How many of aligned chunks obtained in the previous step contain a sound terminology in their target part (the new bi-lingual terminological pairs established).

On the basis of these results we developed a binary supervised classifier with aim to predict whether extracted terms belong to a domain terminology.

4. The Set-Up of the Experiment

In this section we will present resources that we used as an input for our experiment (subsections 4.1.– 4.4.), and the tools used in the processing steps (Subsection 4.5.). For this experiment we are using as an input:

- For the domain of Library and Information Science we have developed the English/Serbian textual resource containing 14,710 aligned sentences;
- Already mentioned Dictionary of Library and Information Science (English/Serbian pairs);
- The rule-based system for the extraction and lemmatization of potential terminological nominal phrases;
- Bilingual Serbian/English list of inflected word forms and MWE pairs derived from bilingual dictionaries and morphological (inflected) dictionaries for Serbian and English;

4.1. Aligned/parallel corpus

The English/Serbian textual resource was derived from the journal for Digital Humanities *Infotheca*[3] that is published biannually in Open Access. 12 issues with 84 papers were aligned at sentence level resulting in 14,710 alignment

[3] infoteka.bg.ac.rs/index.php/en

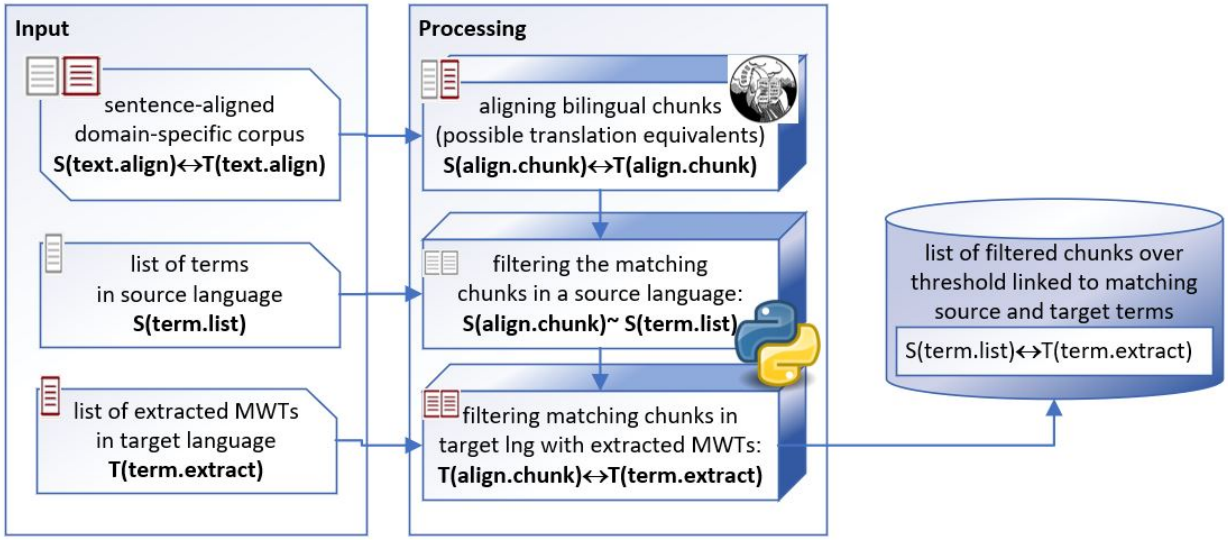


Figure 1: The overall design of the system for terminology extraction using monolingual and bilingual resources.

pairs. (Stanković et al., 2017); (Stanković et al., 2014).[4] The Serbian part has 301,818 simple word forms (41,153 different), while English part has 335,965 simple word forms (21,272 different). This means that in the Serbian part word forms repeat approximately 7 times, while in English they repeat 15 times. The major reason for this difference is the high inflection characteristic to Slavic languages producing many different forms for each lemma.

4.2. Dictionary of Library and Information Science

The development of the Dictionary of Librarianship: English-Serbian and Serbian-English (in this text referred to as 'Dictionary') (Ljiljana Kovačević, 2014) has started in 2001 at the National Library of Serbia, with the aim of presenting the librarianship terminology on different media (Kovačević et al., 2004).

This resource was first used on aligned texts in query expansion (Stanković et al., 2012); the Excel format of the dictionary was at that time transformed into a relational database. The version of the Dictionary that we used for our experiment has 12,592 different Serbian terms (9,376, 74% MWT), 11,857 different English terms (8,575, 72% MWT), and 17,872 distinct pairs.[5] Among distinct pairs in 10,574 cases both terms were MWT (60%), while in 1,923 cases a Serbian MWT had a single-word term equivalent in English (11%), and in 1,070 cases an English MWT had a single-word term equivalent in Serbian (6%). Both terms in a pair were SWTs in 4,305 cases (24%). Among Serbian SWTs, 1,378 are components of some MWTs, while the same occurs for 1,245 English SWTs. All important features of the original dictionary were preserved: the relations between translational equivalents, the synonymous relations within each specific language, and, important for Serbian, relations between Ekavian and Ijekavian pronunciation variants. For the research presented in this paper, we used only Ekavian variant (because texts were in this pronunciation) and only those English/Serbian translation pairs where at least one term in a pair is an MWT.

4.3. The rule-based system for the terminology extraction

The approach to terminology extraction for Serbian based on e-dictionaries and local grammars described in (Stankovic et al., 2016) was improved with additional syntactic patterns in order to cover as many terminology structures as possible (Ranka Stankovic, Cvetana Krstev, 2016). In this research, 20 syntactic structures grouped in 12 classes[6] for extraction purposes were used, which can extract the most frequent syntactic structures identified by an analysis of several Serbian terminological dictionaries and Serbian e-dictionary of MWEs.

This system was applied to the Serbian part of the aligned text (presented in 4.1.) and the results of its work are presented in Table 1.[7] For each class the syntactic structure it recognizes is output as well as the number of extracted forms and (word by word) lemmas.[8] The total number of

[4]Available for searching at *Biblisha* site `jerteh.rs/biblisha/Default.aspx`

[5]The version on the Web contains 40.000 entries (appr. 14.000 in Serbian, 12.400 in English and 14.000 in German) `http://rbi.nb.rs/en/home.html`

[6]One class can group MWTs with various syntactic structures (recognized by different graphs, or finite-state automata); all MWTs in one class have the same number and characteristics of components that inflect.

[7]A – adjective, N – noun, *g* – the genitive case, *i* – the instrumental case, Prep – preposition, *pc* – the case that agrees with the preceding preposition, x – a word separator or a MWU component that does not inflect; S_2 – two component MWU; S_3 – three component MWU; S_4 – four component MWU.

[8]One of the strongest features of this system is that it performs lemmatization of MWUs; however, for the purpose of this experiment, in order to ensure flexibility, we have done simple-word lemmatization. A proper lemmatization is left for the final phase. The difference between these two types of lemmatization can be illustrated with the following example: a simple-word

ID	Inflectional class	Syntactic pattern	Description	Forms	Lemmas
1	AxN	AxN	Both components inflect and agree in gender, number and case	18,249	13,948
2	2xN	2xN	1st component does not inflect	123	99
3	N2x	$NxN_{g(i)}$	1st component inflects; the second is always in gen/inst	7,724	6,528
4	NxN	NxN	Both components inflect and agree in case and number	3,775	3,522
5	N4x	$NxA_{g(i)}xN_{g(i)}$ $NxPrepxN_{pc}$	1st component inflects; the second and third are in gen/inst 1st component inflects; 3rd agrees in case with a preposition	9,647	8,909
6	AxN2x	AxN2x	1st component inflects and agrees with a structure 3.	2,662	2,480
7	AxAxN	AxAxN	All 3 components inflect and agree in gender, number and case	2,421	2,260
8	2xAxN	2xAxN	1st component does not inflect; 2nd and 3rd are structure 1	137	113
9	N6x	$Nx(S_3)_{g(i)}$ $NxPrep(S_2)_{pc}$	1st inflects; 2nd, 3rd and 4th are a structure 5–8 in gen/instr 1st component inflects; 3rd and 4th are a structure 1–4 in a case that agrees with a preposition	2,452	2,376
10	AxN4x	AxN4x	The 1st component inflects and agrees with a structure 5.	3,160	3,082
11	AxN6x	AxN6x	1st inflects and agrees with a structure 5–8 in gen/instr	1,263	1,252
12	N8x	$Nx(S_4)_{g(i)}$ $NxPrep(S_3)_{pc}$ $(S_2)xPrep(S_2)_{pc}$	1st inflects; remaining 4 are structure 8, 9 in gen/instr 1st inflects; last three are a structure 5–8 in a case that agrees with a preposition First two are structure 1-4; last two are a structure 1–4 in a case that agrees with a preposition	1,135	1,113

Table 1: Term candidates per inflectional classes

recognized forms is 52,748, counting same forms recognized by different finite-state automata (distinct 49,552). The total number of lemmatized forms is 45,682 (distinct 42,638).

4.4. Bilingual list of inflected word forms

We explored different ways of utilizing existing lexical resources to improve the quality of statistical machine alignment. In order to do that we have augmented the set of aligned sentences with inflected forms (English/Serbian). We have used two bilingual lexical resources. (a) Serbian Wordnet (SWN) (Cvetana Krstev, 2013) that is aligned to the Princeton WordNet (PWN)[9] and (b) a bilingual list containing general lexica with 10,551 English/Serbian entries. The production of the bilingual list of inflected forms was done in several steps:

1. First we compiled the parallel list from SWN and PWN containing 75,766 aligned English/Serbian literals. This list was merged with existing bilingual list yielding the new list of 86,317 entries.

2. To each Serbian noun, verb or adjective from the list compiled in the previous step we assigned its inflected forms obtained from the Serbian morphological dictionaries (Krstev, 2008). These inflected forms have various grammatical codes assigned to them that were used in the step 4.

3. We performed the similar procedure for English nouns, verbs and adjectives from the bilingual list. In order to obtain inflected forms with grammatical categories we used the English morphological dictionary from the Unitex distribution[10] and the MULTEX-East English lexicon.[11] Grammatical codes from these two sources were harmonized.

4. In the final step we aligned Serbian and English inflected word forms by using corresponding grammatical codes and harmonizing them as best as possible. In many cases one English word form was aligned with several inflected forms in Serbian. For example, the English noun *board* (in singular) is related to Serbian: *tabla*, *table*, *tabli*, *tablu*, *tablo*, *tablom*, while its plural form *boards* is related to *table*, *tabli*, *tablama*. MWEs from the bilingual list obtained in the first step were connected in the same way: for instance, *blotting paper* ↔ *upijajućeg papira*, *upijajućega papira*, *upijajućem papiru*, . . . and *blotting papers*↔ *upijajuće papire*, *upijajući papiri*, *upijajućih papira*, . . .

At the end of this procedure we obtained the bilingual list of inflected forms having 372,432 entries.

4.5. Alignment of Chunks

In order to acquire a list of aligned bilingual chunks several steps have to be performed. Our dataset included 14,710 aligned sentences, containing general lexica with 10,551 English/Serbian entries, parallel list from SWN and PWN containing 75,766 aligned English/Serbian literals and aligned Serbian and English inflected word forms having 372,432 entries (all described in previous subsections).

lemma of a multi-word form *bibliotečko-informacionom delatnošću* is *bibliotečki-informacioni delatnost*, while a correct lemma is *bibliotečko-informaciona delatnost* 'library and information activities'.

[9]Serbian WordNet can be browsed at `http://sm.jerteh.rs/`.

[10]Unitex/GramLab, a lexical-based corpus processing suite `http://unitexgramlab.org/`

[11]`https://www.clarin.si/repository/xmlui/handle/11356/1041`

First, our data had to be preprocessed by tokenization, truecasing and cleaning. In the next step the 3-gram translation model was built using KenLM (Heafield, 2011) followed by the training of the translation model. For the purpose of word-alignment, phrase extraction, phrase scoring and creating lexicalised reordering tables we used GIZA++,[12] (Och and Ney, 2003), together with *grow-diag-final* symmetrization heuristic (Koehn et al., 2003). Resulted phrase-table contained 1,672,362 potential translation equivalents. In order to discard as many as possible of those aligned pairs that are not exact translation of each other, two filtering steps were done. Each pair of aligned chunks from this list also contains information about inverse and direct phrase translation probability.[13] First we kept only those aligned chunks that have at least one of these probabilities greater than 0.85, simultaneously performing punctuation elimination and discarding those that consisted of punctuation and numbers only. This reduced the list to 982,598 aligned chunks.

For the second step we provided Bag-of-Words (BoW) representation for the English terms from the Dictionary (described at subsection 4.2.) and removed stop words from the list, so the list is mainly populated with domain words. Then we lemmatized each token from BoW using Natural Language Toolkit (nltk) Python library and its WordNet interface.[14] The same simple-word lemmatization was applied to the English parts of the aligned chunks. English parts of the aligned chunks that do not have at least one lemmatized content word present in the lemmatized BoW Dictionary representation were eliminated. Along with original and lemmatized form of chunks and their counts, we also kept information about translation order of words from the original phrase-table. This information helped us to make a backup of Serbian SWTs that translate as English SWTs. We eliminated these pairs, but also stored them in a separate file, so we can use them while obtaining final results (described throughly in Section 5., step 6). Final list contained 491,990 translation pair candidates.

We decided to enrich corpus with additional parallel lists (described in Subsection 4.4.) since we observed certain improvement in evaluations of translation quality. First we splitted corpus of aligned sentences into three disjoint parts: training (80%), development (10%) and test set (10%). BLEU score (Papineni et al., 2002) was obtained for the three different 3-gram language models. First model was trained only on the training set, tuned and tested, which obtained BLEU of 24.78. Second model was trained on the training set extended with the bilingual list containing the general lexica list and the parallel list from SWN and PWN. BLEU score for the test set increased to 24.93. Third model was extended with the list of inflected forms as in the real experiment setting and BLEU score increased to 26.21.

[12]Statistical Machine Translation toolkit can be found at `https://github.com/moses-smt/giza-pp`

[13]How phrase translation probabilities are determined can be read in more details at `http://www.statmt.org/moses/?n=FactoredTraining.ScorePhrases`

[14]More about Natural Language Toolkit for Python and its WordNet interface can be found at `http://www.nltk.org/howto/wordnet.html`

Before continuing to the next processing step, we defined "match" relation between chunks as follows: Let a chunk be represented as an unordered set of distinct words contained in it after stop words removal. Two chunks match if they have the same set representation of this kind.

5. Results and Evaluation

1. The number of distinct[15] aligned chunks after two-pass filtering (the English part matches some term in the Dictionary – $S(align.chunk) \sim S(term.dict)$ – and the Serbian part is a multi-word chunk, that is, it contains at least 2 content words) was 11,678.

2. In the next step the additional filtering was done – $(T(align.chunk) \sim T(term.list)) \wedge (T(term.list) \leftrightarrow S(term.dict)) \wedge (S(term.dict) \sim S(align.chunk))$, as a result 425 different MWTs from the Serbian Dictionary were matched with the Serbian part of the aligned chunk.

3. The aligned chunks from Step 1 were filtered with the additional condition $T(align.chunk) \sim T(term.extract)$. 2,266 chunks were obtained, 2,120 of them matching different extracted MWTs.

4. The aligned chunks from the step 2 were filtered with the additional condition $(T(align.chunk) \sim T(term.extract))$. 326 different Serbian MWTs were both matched with the Dictionary and extracted by the tool.

5. The aligned chunks from step 3 were filtered with the additional condition $(T(align.chunk) \not\sim T(term.list))$. 1,935 Serbian MWTs were extracted by our term extractor (they were not in the Dictionary; they, however, may be synonymous to some term already in the Dictionary due to the condition in step 1).

6. Among results obtained in the previous step there was a number of only partially correct pairs. Namely, some mostly simple-word English terms were aligned with Serbian MWTs that contain as a component the translation of English terms. An example of such situation is: the term LIBRARY translates as BIBLIOTEKA, but this Dictionary term is different from the Serbian chunk and Serbian extracted term (e.g. BIBLIOTEKA $\neq$ PARTNER BIBLIOTEKA 'participating library') and it was therefore kept in step 5. Most of these pairs were expelled in this step, reducing to 1,018 pairs at the end (see also Table 3.

There were 452 (44.4%) Serbian MWTs that have English Dictionary SWT as a translation, and 566 (55.6%) which have respective English Dictionary translation as an MWT. There were 575 extracted MWTs with frequency 1, 158 extracted MWTs with frequency 2 and 285 extracted MWTs

[15]Phrase table often contains several similar entries of the same phrase. For example, *at the digital library*, *for digital library*, *because digital library* and *of the digital library* would represent four different entries within phrase table. We observed these as one phrase, in the manner of the previously defined "match" relation.

with frequency greater or equal to three. For the sake of getting insight in the exact number of different new Serbian MWTs extracted during the last step, we asked professional from the librarianship domain to perform manual annotation of the extracted phrases. After manual validation, 515 extracted Serbian MWTs were evaluated as good translations of the paired English Dictionary phrases.

The examples illustrating this process are given in Table 3. MWTs extracted by the extractor that are equal to Serbian dictionary entries are determined in step 4 while those that are new are determined in step 6. New MWTs that represent only partial translations are deleted after step 5. Note that Serbian terms presented in this table are simple-word lemmatized, as explained in Subsection 4.3.. Correct MWE lemmatization is performed as a separate task.

At this moment we did not consider MWTs not recognized by our term extractor (condition $(T(align.chunk) \not\sim T(term.list))$ applied to the results of step 3 filtering), because they are mixed with false omissions due to enrichment of the aligned corpus with bilingual inflected dictionary (see Subsection 4.4.).

In order to make our system able to automatically decide whether an extracted term is a valid domain term, we trained a supervised binary classifier. All samples from the step 4 were considered good translations (331 samples) and this set was expanded with samples manually annotated as good translations during step 6 (515, total 846 positive samples). The remaining pairs from the step 6 were labeled as bad translations (negative class, 503 samples).

In the preprocessing step, we extracted 43 text features (also referred as "linguistic" features in (Ebert, 2017; Repar and Pollak, 2017)) from original (GIZA_SRP_ORIG) and lemmatized (GIZA_SRP_LEMM) form of Serbian chunk obtained from GIZA++, corresponding extracted Serbian term (SRP_EXTRACTED) and from the English part of the aligned chunk (GIZA_ENG_LEMM) and its Dictionary entry (ENG_DICT). These features are: 1) total number of words in Serbian and English chunks, extracted term and English Dictionary term (*_WC), 2) extracted term frequency (*_FREQ), 3) count of chunks in text (*_COUNT), 4) count of present diacritics in Serbian terms (*_DIACRITICS), 5) number of characters in English and Serbian terms (*_LEN), 6) ratio of diacritics count and length (*_DIACRITICS_LEN_RATIO) and 7) ratio of lengths of two different terms (*_LEN_RATIO). After eliminating high correlating or constant features, the final dataset contained 28 features.[16]

The performance of the classifier was evaluated in the 5-fold cross validation setting using the following metrics: accuracy (Acc), precision (P), recall (R) and F-score (F_1). After several different classifiers evaluation, Gradient Boost model (Friedman, 2001) implementation from the scikit-learn toolkit for Machine Learning for Python (Pedregosa et al., 2011) turned out to have the best performance on this dataset. Gradient boosting is an iterative technique that combines a set of weak learners and delivers improved prediction accuracy. The instances predicted correctly are given a lower weight and the ones misclassified are weighted higher, until best instance weights are found. The performance of our classifier per each fold k is displayed in Table 2.

k	1	2	3	4	5	Avg
Acc	0.844	0.848	0.870	0.852	0.888	0.861
R	0.906	0.899	0.912	0.865	0.896	0.896
P	0.856	0.864	0.897	0.896	0.908	0.884
F_1	0.881	0.881	0.905	0.880	0.902	0.890

Table 2: Gradient Boost classifier evaluation

Ten features with highest influence on the classification outcome are displayed in Figure 2.

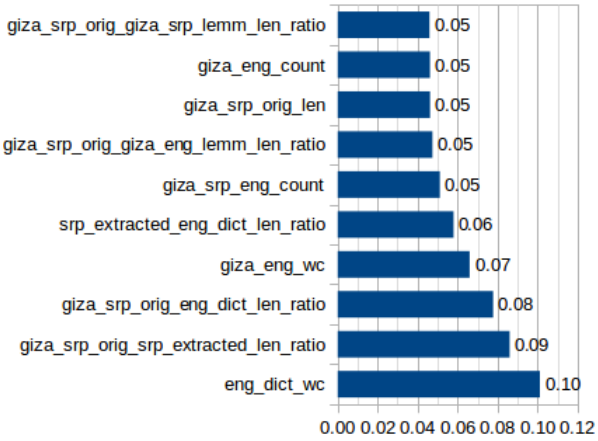


Figure 2: Influence of features on predictions

6. Conclusion

With this new experiment we wanted to achieve two goals: (a) to automatically evaluate extracted nominal phrases in the target language terminology by aligning it with the established terminology in the source language; (b) to build a classifier that would evaluate as positive automatically caught Serbian terminology using English baits. As an additional result we enriched the Dictionary of Library and Information Sciences with 515 synonyms in the Serbian part. Another by-product is the bilingual Serbian/English list of inflected word forms and MWE pairs derived from bilingual dictionaries and morphological dictionaries.

We will apply the same approach to other domains – mining, electro-distribution and management – since aligned domain corpora have already been prepared. At the same time the presented system will be improved with the user friendly interface for presentation of the results. Our intention is also to revise and further improve the relation "match" between aligned chunks and lexical resources, and possibly to introduce numeric values for the assessment rate. Needless to say, the enrichment of sentence-aligned domain-specific corpora is the long-term activity.

[16]The whole set of extracted features and the classifier itself are available on `https://github.com/Branislava/domain_terminology_extraction`.

Serbian Giza Chunk	Serbian Giza Lemmatized	Serbian Extracted	Serbian Dictionary	English Dictionary	Result sets	MWT
bibliotečkih udruženja sa	bibliotečki udruženje	bibliotečki udruženje	biblioteka društvo	association library	1, 3, 5, 6	yes
bibliotekarskim društvima	bibliotekarski društvo u	bibliotekarski društvo	biblioteka društvo	association library	1, 3, 5, 6	yes
bibliotečka asocijacija	nije bibliotečki asocijacija	bibliotečki asocijacija	biblioteka društvo	association library	1, 3, 5, 6	yes
udruženja i asocijacija poput biblioteke	udruženje i asocijacija poput biblioteka		biblioteka društvo	association library	1	
bibliotekarskog društva	bibliotekarski društvo	bibliotekarski društvo	bibliotekar	librarian	1, 3, 5	
bibliotečko informacionog sitema	bibliotečki informacioni sitema na	bibliotečki informacioni sistem	bibliotečki informacioni sistem	library information system	1, 2, 3, 4	
biblioteka medijateka	biblioteka medijateka		bibliotečki informacioni sistem	library information system	1	
drugim bibliotečko informacionim sistemima da	drugi bibliotečki informacioni sistem		bibliotečki informacioni sistem	library information system	1	
i informacioni sistemi u bibliotekama	informacioni sistem u biblioteka	informacioni sistem u biblioteka	bibliotečki informacioni sistem	library information system	1, 3, 5, 6	yes
koriste ovaj bibliotečko informacioni sistem	koji koriste ovaj bibliotečki informacioni sistem		bibliotečki informacioni sistem	library information system	1	
bibliotečke građe koju	bibliotečki građa koji	bibliotečki građa	bibliotečki građa	library material	1, 2, 3, 4	
biblioteka i bibliotečke građe	biblioteka i bibliotečki građa		bibliotečki građa	library material	1	
građe u bibliotekama	građa u biblioteka	građa u biblioteka	bibliotečki građa	library material	1, 3, 5, 6	yes
građu sve su biblioteke pod	građa sav su biblioteka		bibliotečki građa	library material	1	
bibliotečkog materijala	bibliotečki materijal i	bibliotečki materijal	bibliotečki građa	library material	1, 3, 5, 6	yes
građe iz drugih biblioteka a	građa iz drugi biblioteka		bibliotečki građa	library material	1	
bibliotečke standarde	bibliotečki standard	bibliotečki standard	bibliotečki građa	library material	1, 3, 5, 6	no
u pitanju bibliotečki materijal	pitanje bibliotečki materijal za		bibliotečki građa	library material	1	
biblioteka partnera	biblioteka partner i	biblioteka partner	biblioteka učesnica	participating library	1, 3, 5, 6	yes
biblioteke učesnice	biblioteka učesnica i	biblioteka učesnica	biblioteka učesnica	participating library	1, 2, 3, 4	yes
biblioteke učesnice i bibliotečki	biblioteka učesnica i bibliotečki		biblioteka učesnica	participating library	1	
su partneri biblioteke	partner biblioteka	biblioteka partner	biblioteka	library	1, 3, 5	
istraživanje iz oblasti bibliotekarstva i	istraživanje iz oblast bibliotekarstvo	istraživanje iz oblast bibliotekarstvo	naučni biblioteka	research library	1, 3, 5, 6	no
na sve naučne biblioteke	sav naučni biblioteka u		naučni biblioteka	research library	1	
of research libraries	of research libraries		naučni biblioteka	research library	1	
naučnoistraživačkih biblioteka	naučnoistraživački biblioteka u	naučnoistraživački biblioteka	naučni biblioteka	research library	1, 3, 5, 6	yes
naučne biblioteke ako su bile	naučni biblioteka ako su bile		naučni biblioteka	research library	1	
naučne biblioteke ako	naučni biblioteka u	naučni biblioteka	naučni biblioteka	research library	1, 2, 3, 4	
biblioteke ka istraživanju	biblioteka ka istraživanje		naučni biblioteka	research library	1	
iz istraživačkih biblioteka	istraživački biblioteka	istraživački biblioteka	naučni biblioteka	research library	1, 3, 5, 6	yes
knjiškomu moljcu	knjiški moljac	knjiški moljac	knjižni crv	bookworm	1, 3, 5, 6	yes
su razrešavanje višeznačnosti	razrešavanje višeznačnost i	razrešavanje višeznačnost	objašnjenje	disambiguation	1, 3, 5, 6	yes
za npr razrešavanje	npr razrešavanje		objašnjenje	disambiguation	1	
otklanjanje višeznačnosti su	otklanjanje višeznačnost	otklanjanje višeznačnost	objašnjenje	disambiguation	1, 3, 5, 6	yes
a razrešavanje	a razrešavanje		objašnjenje	disambiguation	1	
postoji povratna sprega	da postoji povratan sprega		povratan vez	feedback	1	
i povratne informacije	povratan informacija za	povratan informacija	povratan vez	feedback	1, 3, 5, 6	yes

Table 3: The examples illustrating the steps of terminology alignment

Acknowledgment

This research was partially supported by Serbian Ministry of Education and Science under the grants #III 47003 and 178006.

7. Bibliographical References

Ananiadou, S., McNaught, J., and Thompson, P. (2012). *The English Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

Baldwin, T. and Kim, S. N. (2010). Multiword expressions. *Handbook of natural language processing*, 2:267–292.

Bouamor, D., Semmar, N., and Zweigenbaum, P. (2012). Identifying bilingual multi-word expressions for statistical machine translation. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Ebert, S. (2017). *Artificial Neural Network Methods Applied to Sentiment Analysis*. Ph.D. thesis, Ludwig-Maximilians-Universität München.

Friedman, J. H. (2001). Greedy Function Approximation: a Gradient Boosting Machine. *Annals of statistics*, pages 1189–1232.

Heafield, K. (2011). KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical Phrase-based Translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.

Kovačević, L., Injac, V., and Begenišić, D. (2004). *Bibliotekarski terminološki rečnik: englesko-srpski, srpsko-engleski*. Narodna biblioteka Srbije.

Krstev, C. (2008). *Processing of Serbian. Automata, Texts and Electronic Dictionaries*. Faculty of Philology of the University of Belgrade.

Moirón, B. V. and Tiedemann, J. (2006). Identifying idiomatic expressions using automatic word-alignment. In *Proceedings of the EACL 2006 Workshop on Multi-word expressions in a multilingual context*, pages 33–40.

Och, F. J. and Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational linguistics*, 29(1):19–51.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Repar, A. and Pollak, S. (2017). Good Examples for Terminology Databases in Translation Industry. In *eLex 2017: eLex 2017: The 5^{th} biennial conference on electronic lexicography, Netherlands, 19-21 September 2017*, pages 651–661.

Stanković, R., Krstev, C., Obradović, I., Trtovac, A., and Utvić, M. (2012). A tool for enhanced search of multilingual digital libraries of e-journals. In Nicoletta Calzolari, et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Stankovic, R., Krstev, C., Obradovic, I., Lazic, B., and Trtovac, A. (2016). Rule-based automatic multi-word term extraction and lemmatization. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Stanković, R., Krstev, C., sko Vitas, D., Vulović, N., and Kitanović, O., (2017). *Keyword-Based Search on Bilingual Digital Libraries*, pages 112–123. Springer International Publishing, Cham.

Tsvetkov, Y. and Wintner, S. (2010). Extraction of multi-word expressions from small parallel corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1256–1264, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vintar, Š. and Fišer, D. (2008). Harvesting multi-word expressions from parallel corpora. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Vitas, D., Popović, L., Krstev, C., Obradović, I., zetić, G. P.-L., and Stanojević, M. (2012). *Srpski jezik u digitalnom dobu – The Serbian Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

8. Language Resource References

Cvetana Krstev, Duško Vitas. (2015). *Serbian Morphological Dictionary - SMD*. University of Belgrade, HLT Group and Jerteh, Lexical resource, 2.0.

Cvetana Krstev. (2013). *Serbian WordNet*. University of Belgrade, HLT Group and Jerteh, Lexical database, 2.0.

Ljiljana Kovačević, Dr Dobrila Begenišić, Vesna Injac-Malbaša. (2014). *Dictionary of Library and Information Sciences*. National Library of Serbia: Scientific Research Department, Lexical database, 3.0.

Ranka Stankovic, Cvetana Krstev. (2016). *LeXimir*. University of Belgrade, HLT Group, Software Toolkit, 2.0.

Ranka Stanković and Cvetana Krstev and Nikola Vulović and Biljana Lazić. (2014). *Biblisha*. University of Belgrade, HLT Group, Bilingual digital library, 2.0.

Construction of Large-scale English Verbal Multiword Expression Annotated Corpus

Akihiko Kato, Hiroyuki Shindo, Yuji Matsumoto
Graduate School of Information and Science
Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara, 630-0192, Japan
{kato.akihiko.ju6, shindo, matsu} @is.naist.jp

Abstract

Multiword expressions (MWEs) consist of groups of tokens, which should be treated as a single syntactic or semantic unit. In this work, we focus on verbal MWEs (VMWEs), whose accurate recognition is challenging because they could be discontinuous (e.g., **take .. off**). Since previous English VMWE annotations are relatively small-scale in terms of VMWE occurrences and types, we conduct large-scale annotations of VMWEs on the Wall Street Journal portion of English Ontonotes by a combination of automatic annotations and crowdsourcing. Concretely, we first construct a VMWE dictionary based on the English-language Wiktionary. After that, we collect possible VMWE occurrences in Ontonotes and filter candidates with the help of gold dependency trees, then we formalize VMWE annotations as a multiword sense disambiguation problem to exploit crowdsourcing. As a result, we annotate 7,833 VMWE instances belonging to various categories, such as phrasal verbs, light verb constructions, and semi-fixed VMWEs. We hope this large-scale VMWE-annotated resource helps to develop models for MWE recognition and dependency parsing that are aware of English MWEs. Our resource is publicly available.

Keywords: Multiword expressions, Phrasal verbs, Crowdsourcing

1. Introduction

Multiword expressions (MWEs) consist of groups of tokens, which should be treated as a single syntactic or semantic unit. MWEs are also known as "idiosyncratic interpretations that cross word boundaries" (Sag et al., 2002).

In this paper, we focus on verbal MWEs (VMWEs) among various types of MWEs, such as compound nouns and compound function words. An accurate recognition of VMWEs is challenging because VMWEs could be discontinuous (e.g., **take .. off**). We show the main categories of VMWEs in Table 1.

While dependency parsing and MWE recognition could be solved independently, dependency structures in that each MWE is a syntactic unit are preferable to word-based dependency structures for downstream NLP tasks, such as semantic parsing. Because MWE recognition could help syntactic parsing (Nivre and Nilsson, 2004; Eryiğit et al., 2011), several works tackle MWE-aware dependency parsing in French (Candito and Constant, 2014; Nasr et al., 2015). They use French Treebank (Abeillé et al., 2003) because of its explicit MWE annotations.

Regarding English MWEs, Schneider et al. (2014) constructs an MWE-annotated corpus based on English Web Treebank (Bies et al., 2012). However, the number of VMWE occurrences (1,444) and types (1,155) in their corpus is relatively small-scale.

In this work, we conduct full-scale VMWE annotations on the Wall Street Journal (WSJ) portion of English Ontonotes (Pradhan et al., 2007), which results in 7,833 VMWE occurrences and 1,608 types. Concretely, we construct a VMWE dictionary based on the English-language Wiktionary [1]. Based on this dictionary, we collect possible VMWE occurrences from Ontonotes and filter candidates with the help of gold dependency trees. To exploit crowdsourcing, we formalize VMWE annotations as a multiword sense disambiguation problem. This resource will enable the development of large-scale English MWE recognition and MWE-aware parsing models.

Our resource is publicly available at https://github.com/naist-cl-parsing/Verbal-MWE-annotations.

2. Corpus Construction

2.1. Candidate Extraction

First, we construct a VMWE dictionary by extracting multiword verbs from English Wiktionary [2]. We exclude auxiliary verbs and MWEs consisting of be-verbs and non-verbal components (e.g., **be above**, **be with**). As a result, we get 8,369 VMWE types.

Second, we extract possible VMWE occurrences in 37,015 sentences of the WSJ portion of Ontonotes Release 5.0 (LDC2013T19) by using the above VMWE dictionary. We allow each VMWE to include gaps (e.g., **take .. off**), consider inflections of verbs and a variability of placeholders in semi-fixed MWEs (e.g., someone, something, one's and

Category	Examples
Verb-particle constructions	pick up, take over
Prepositional verbs	look for, base on
Light verb constructions	make a decision, take a look
Verb-noun(-preposition)	take care (of)
Semi-fixed VMWEs	make one's way

Table 1: Main categories of VMWEs.

[1] https://en.wiktionary.org

[2] We select multiword entries that have "English_verbs" as categories.

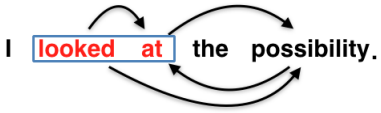


Figure 1: Dependency trees with function-head (above) and content-head (below). We omit edges common in both trees. The box corresponds to a VMWE ("look at"). To filter a possible VMWE as a subtree of a dependency tree, a function-head scheme is preferable to a content-head scheme.

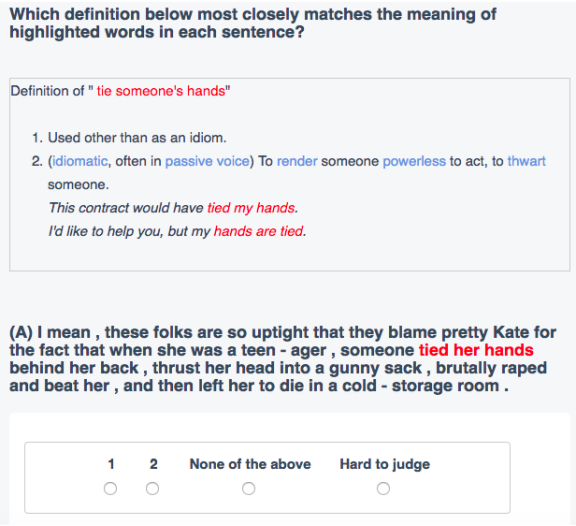


Figure 2: A screenshot of a web interface for VMWE annotations on CrowdFlower.

oneself). We exclude candidates that do not include any verbs by using gold part-of-speech information. Also, we filter out candidates that have other verbs or punctuation marks within the gaps.

Because most of the VMWEs are syntactically regular, we filter a VMWE whose components form a subtree in a Stanford basic dependency tree (Marneffe and Manning, 2008), which is converted from a phrase structure tree given in Ontonotes. We exploit Stanford basic dependency because its function-head scheme is suitable for filtering positive occurrences of VMWEs, that have a frequent POS pattern, "V IN". In many cases, a noun phrase follows this type of MWEs. Therefore, in a content-head scheme like Universal Dependency (McDonald et al., 2013), a verb of this MWE governs a head of the noun phrase, that is, such MWE does not form a subtree (Figure 1). On the contrary, such MWE corresponds to a subtree in a function-head scheme.

Regarding phrasal verbs (PVs), we perform an additional filtering. In this work, we construct a VMWE-annotated corpus by extending Komai et al. (2015)'s corpus, because they have partially performed annotations of PVs in Ontonotes. For PVs that are not covered by their dictionary, we adopt the following methods: (1) We classify PVs as verb-particle constructions (VPCs) or prepositional verbs (Baldwin et al., 2009). (2) We examine a label of a dependency edge from a verb to a particle. For

	# of constituent tokens				
	2	3	4	≥ 5	Total
VMWE instances	7,067	597	138	31	7,833
VMWE types	1,235	270	80	23	1,608

Table 2: Corpus statistics. We show VMWE instances and types by the number of constituent word tokens.

# of gaps	0	1	2
VMWE instances	6,855	968	10

Table 3: VMWE instances by the number of gaps.

VPCs, we regard a candidate as a positive VMWE occurrence iff the dependency label is "prt". For prepositional verbs, if the dependency label is "prep", and there is no gap between the verb and the particle, we regard this candidate as a positive VMWE occurrence. This is subject to rules proposed by Komai et al. (2015). Otherwise, we conduct crowdsourced annotations.

2.2. Large-scale Annotations of VMWEs by Crowdsourcing

Based on the above filtering, we conduct large-scale VMWE annotations on the WSJ portion of English Ontonotes by crowdsourcing using a web interface shown in Figure 2. To exploit crowdsourcing, we formalize VMWE annotations as a multiword sense disambiguation problem. Annotators read a sentence in which components of a possible VMWE are highlighted. They are also given possible definitions of the VMWE, extracted from the English part of Wiktionary. For each VMWE, we provide one literal sense and non-literal senses [3]. Based on this, they are asked to determine which definition most closely matches the meaning of highlighted words in the sentence. During annotations, workers are allowed to answer that the meaning of highlighted words is not in the given senses ("None of the above"), or they are not certain of the multiword sense ("Hard to judge").

We collect crowdsourced annotations of VMWEs by using CrowdFlower [4]. We set the following requirements: (1) Annotators belong to *Level 3* contributors, who are regarded as the smallest group of most experienced, highest accuracy contributors on CrowdFlower. (2) Annotators live in countries with English as an official language. (3) Annotators achieve a success rate higher than 70 % in answering test questions, to which we give gold answers. To facilitate annotations, we provide workers with an interface to show multiple sentences (less than 6) that include possible occurrences of the same VMWE. We collect three judgments for each of 2,135 possible VMWE occurrences. Data collection costs $1,016 USD in total.

To determine whether each VMWE candidate is positive or not, we adopt the following criteria:

1. If all judgments correspond to the same sense, we

[3] If a definition of a literal sense is omitted in Wiktionary, we add a choice corresponding to it ("Used other than as an idiom").

[4] https://www.crowdflower.com

POS pattern	Continuous	Discontinuous	Frequent MWEs
V_IN	3,071	260	base on : 142 look for : 86 focus on : 77 go to : 70 account for : 69
V_RP	2,081	229	set up : 62 take over : 49 point out : 47 turn out : 43 pick up : 39
V_RB	547	116	go back : 17 come back : 17 do well : 15 go down : 13 go ahead : 13
V_NN	280	167	take place : 41 do business : 27 take effect : 26 take steps : 24 have time : 22
V_DT_NN	114	45	take a look : 13 make a decision : 8 pave the way : 5 lay the groundwork : 5 turn a profit : 5
V_RP_IN	98	4	come up with : 20 make up for : 12 keep up with : 8 live up to : 7 add up to : 5
V_JJ	77	11	make sure : 14 go wrong : 8 go public : 6 keep quiet : 5 make much : 4
V_IN_NN	56	26	have in mind : 8 take into account : 7 set in motion : 5 sign into law : 5 take to heart : 4
V_V	47	32	be called : 34 be had : 5 have got : 4 make known : 4 let know : 4
V_PRP	77	0	make it : 16 have it : 10 buy it : 9 move it : 5 find oneself : 5
V_PRP$_NN	49	1	have one's way : 5 run one's course : 4 make one's way : 3 read someone's lips : 3 drag one's feet : 2
V_IN_IN	37	9	get out of : 12 come out of : 11 make out of : 8 grow out of : 4 get through to : 1
V_IN_DT_NN	33	11	put on the block : 5 come to an end : 5 grind to a halt : 3 jump on the bandwagon : 3 get into the act : 3
V_RB_IN	41	3	get back to : 6 shy away from : 5 cut back on : 4 walk away from : 4 come up with : 3
V_NN_IN	32	6	take advantage of : 21 take care of : 6 keep tabs on : 3 get wind of : 2 take issue with : 1
MD_V	17	6	will do : 23
V_DT_JJ_NN	17	0	go a long way : 7 look the other way : 4 learn the hard way : 2 take a back seat : 1 fight a losing battle : 1
V_DT_NN_IN	14	1	keep a lid on : 3 keep an eye on : 2 put the brakes on : 2 put the blame on : 1 put a damper on : 1
RB_V	5	7	never mind : 4 clear cut : 4 second guess : 2 reverse engineer : 1 short circuit : 1
V_RP_PRP$_NN	8	4	make up one's mind : 7 pull in one's horns : 2 roll up one's sleeves : 1 clean up one's act : 1 hold up one's end : 1

Table 4: VMWE statistics by POS patterns (for patterns occurring 10 or more times).

adopt it (67.1 %). If the sense is not literal, we regard this candidate as a VMWE.

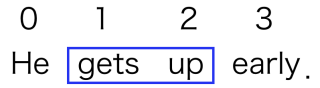
2. If any judgment does not correspond to a literal sense, we regard the candidate as a positive occurrence of the VMWE (9.0 %).

3. Otherwise, we manually select a definition most closely matching the meaning of the VMWE candidate in the sentence. If the definition corresponds to one of non-literal senses, we regard this candidate as a VMWE (23.8 %).

2.3. Resolution of Inclusions and Overlaps

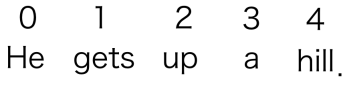
Finally, we check inclusions and overlaps between annotations by us and those by (Komai et al., 2015), which results in 159 inclusions and 40 overlaps. Regarding inclusions, we adopt the broader MWE-spans. For instance, given two MWE occurrences corresponding to "come at" and "come at a price" in that a span of the latter includes a span of the former, we leave only the latter one. Concerning overlaps, we merge overlapped MWE-spans if we can get a new VMWE that is in both of the following dictionaries: Cambridge Dictionary [5] and The Free Dictionary [6]. For instance, we get an occurrence of "take over the reins" by merging occurrences of "take the reins" and "take over". Also, we resolve pseudo overlaps originating from false annotations. As a result, we reduce the number of overlaps to 11 instances, which correspond to essential overlaps, such as "look back" and "look .. on .. as" in the following sentence: "He may be able to **look back on** this election **as** the

[5] http://dictionary.cambridge.org
[6] http://idioms.thefreedictionary.com



Indices of a VMWE : (1,2)

(a) A positive instance (non-literal usage)



(b) A negative instance (literal usage)

Figure 3: Positive and negative instances of a VMWE (**get up**).

high-water mark of far-left opposition.".

2.4. Corpus Statistics

As a result of annotations, we get 1,608 VMWE types and 7,833 instances in Ontonotes. We show VMWE frequencies by the number of constituent word tokens (Table 2) and by the number of gaps (Table 3). Moreover, frequent POS patterns are shown in Table 4, in which you can see various kinds of VMWE, such as phrasal verbs (PVs), light verb constructions (LVCs), and semi-fixed MWEs. The top 3 POS patterns ("V_IN", "V_RP", and "V_RB") correspond to PVs. Each of those includes a fair number of discontinuous instances.

Our corpus annotations are represented as token indices of components of VMWEs. By using them, we can classify potential VMWEs in our corpus as positive and negative instances (Figure 3).

3. Related Work

We introduce several MWE-annotated corpora. First, French Treebank (Abeillé et al., 2003) is often used as a dataset for French MWE-aware dependency parsing (Candito and Constant, 2014) because of its explicit MWE annotations. It consists of phrase structure trees, augmented with morphological information and functional annotations of verbal dependents. Second, Vincze (2012) provides an English-Hungarian parallel corpus annotated for LVCs, which belong to VMWEs. Their corpus contains 703 LVCs in Hungarian and 727 in English based on 14,261 sentence alignment units, taken from economic-legal texts and literature. Recently, PARSEME organized a shared task on automatic identification of verbal MWEs (Savary et al., 2017). They provide annotation guidelines and annotated corpora of 5.5 million tokens and 60,000 VMWE annotations for 18 languages. Note that their corpora do not support English in edition 1.0.

Regarding English MWEs, Shigeto et al. (2013) first constructs an MWE dictionary by extracting functional MWEs [7] from the English-language Wiktionary, and classifies their occurrences in Ontonotes into either MWE or literal usage. Kato et al. (2016) and Kato et al. (2017) integrates annotations of these functional MWEs and named entities (NEs) [8] into phrase structures by establishing MWEs as subtrees. They exploit this dataset for experiments on English MWE-aware dependency parsing.

4. Conclusion

In this work, we conduct large-scale annotations of English VMWEs in the Wall Street Journal portion of Ontonotes. Based on a VMWE dictionary extracted from English Wiktionary, we collect possible VMWE occurrences in Ontonotes, and filter candidates with the help of gold dependency trees. To take advantage of crowdsourcing, we formalize annotations of VMWEs as a multiword sense disambiguation problem. Our future work could involve the followings:

1. We plan to integrate our VMWE annotations into annotations for functional MWEs and named entities in Ontonotes by Kato et al. (2016) and Kato et al. (2017). This will help to develop models for MWE recognition and dependency parsing that are aware of various kinds of English MWEs.

2. We get VMWE occurrences in Ontonotes for only 1,608 out of 8,369 types in our VMWE dictionary. Therefore, we plan to explore VMWE occurrences on a larger corpus, such as the Annotated English Gigaword treebank [9].

5. Acknowledgements

This work was partly supported by JSPS KAKENHI Grant Number 15K16053, 26240035 and JST CREST Grant Number JPMJCR1513, Japan. We are grateful to members of Computational Linguistics Laboratory at NAIST, and anonymous reviewers for valuable feedback. Regarding the preparation of a web interface for crowdsourcing, I am particularly grateful for the assistance given by Michael Wentao Li.

6. Bibliographical References

Abeillé, A., Clément, L., and Toussenel, F., (2003). *Building a Treebank for French*, pages 165–187. Springer Netherlands, Dordrecht.

Baldwin, T., Kordoni, V., and Villavicencio, A. (2009). Prepositions in applications: A survey and introduction to the special issue. *Computational Linguistics*, 35(2):119–149.

Bies, A., Mott, J., Warner, C., and Kulick., S. (2012). English web treebank. *Technical Report LDC2012T13, Linguistic Data Consortium, Philadelphia, Pennsylvania, USA.*

Candito, M. and Constant, M. (2014). Strategies for contiguous multiword expression analysis and dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 743–753. Association for Computational Linguistics.

[7] By functional MWEs, we mean MWEs that function either as prepositions, conjunctions, determiners, pronouns, or adverbs.

[8] The NE annotations are given by Ontonotes.

[9] http://catalog.ldc.upenn.edu/LDC2012T21

Eryiğit, G., Ilbay, T., and Can, O. A. (2011). Multiword expressions in statistical dependency parsing. In *Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages*, pages 45–55, Dublin, Ireland, October. Association for Computational Linguistics.

Kato, A., Shindo, H., and Matsumoto, Y. (2016). Construction of an english dependency corpus incorporating compound function words. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Kato, A., Shindo, H., and Matsumoto, Y. (2017). English multiword expression-aware dependency parsing including named entities. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 427–432, Vancouver, Canada, July. Association for Computational Linguistics.

Komai, M., Shindo, H., and Matsumoto, Y. (2015). An efficient annotation for phrasal verbs using dependency information. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation: Posters*, pages 125–131.

Marneffe, M.-C. and Manning, D. C., (2008). *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, chapter The Stanford Typed Dependencies Representation, pages 1–8. Coling 2008 Organizing Committee.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N., and Lee, J. (2013). Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97. Association for Computational Linguistics.

Nasr, A., Ramisch, C., Deulofeu, J., and Valli, A. (2015). Joint dependency parsing and multiword expression tokenization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1116–1126, Beijing, China, July. Association for Computational Linguistics.

Nivre, J. and Nilsson, J. (2004). Multiword units in syntactic parsing. In *Workshop on Methodologies and Evaluation of Multiword Units in Real-World Applications*, pages 39–46, Lisbon, Portugal.

Pradhan, S. S., Hovy, E. H., Marcus, M. P., Palmer, M., Ramshaw, L. A., and Weischedel, R. M. (2007). Ontonotes: A unified relational semantic representation. In *Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007), September 17-19, 2007, Irvine, California, USA*, pages 517–526.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A. A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. In *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing, pages 1–15, London, UK, UK. Springer-Verlag.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (2017). The parseme shared task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47, Valencia, Spain, April. Association for Computational Linguistics.

Schneider, N., Onuffer, S., Kazour, N., Danchik, E., Mordowanec, M. T., Conrad, H., and Smith, N. A. (2014). Comprehensive annotation of multiword expressions in a social web corpus. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 455–461, Reykjavik, Iceland, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1433.

Shigeto, Y., Azuma, A., Hisamoto, S., Kondo, S., Kouse, T., Sakaguchi, K., Yoshimoto, A., Yung, F., and Matsumoto, Y., (2013). *Proceedings of the 9th Workshop on Multiword Expressions*, chapter Construction of English MWE Dictionary and its Application to POS Tagging, pages 139–144. Association for Computational Linguistics.

Vincze, V. (2012). Light verb constructions in the szegedparalellfx english-hungarian parallel corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

Konbitzul: an MWE-specific database for Spanish-Basque

Uxoa Iñurrieta, *Itziar Aduriz, Arantza Díaz de Ilarraza, Gorka Labaka, Kepa Sarasola
IXA NLP group, University of the Basque Country
*University of Barcelona
usoa.inurrieta@ehu.eus, itziar.aduriz@ub.edu, {a.diazdeilarraza, gorka.labaka, kepa.sarasola}@ehu.eus

Abstract

This paper presents Konbitzul, an online database of verb+noun MWEs in Spanish and Basque. It collects a list of MWEs with their translations, as well as linguistic information which is NLP-applicable: it helps to identify occurrences of MWEs in multiple morphosyntactic variants, and it is also useful for improving translation quality in rule-based MT. In addition to this, its user-friendly interface makes it possible to simply search for MWEs along with translations, just as in any bilingual phraseological dictionary.

1. Introduction

Multiword Expressions (MWEs), also called Phraseological Units (PUs), are combinations of words which together express a single meaning (Sag et al., 2002). They often have irregular lexical-semantic and/or morphosyntactic features, and they are not always translated word-for-word (Examples 1-3). This means they cause challenges in various disciplines, such as Lexicography, Translation and Natural Language Processing (NLP).

(1) EN: *pull sb's leg*
ES: *tomar el pelo (a)* lit. *take sb's hair*
EU: *adarra jo* lit. *play the horn (to sb)*

(2) EN: *take steps*
ES: *dar pasos* lit. *give steps*
EU: *urratsak egin* lit. *steps do*

(3) EN: *take off*
ES: *alzar el vuelo* lit. *raise the flight*
EU: *aireratu* lit. *go-to-the-air*

Although interest in Phraseology has a longer history, studies on MWEs have multiplied considerably over the last two decades (Baldwin and Kim, 2010; Savary et al., 2015). Most of the work undertaken within the field of NLP focuses on MWE candidate extraction (Ramisch, 2015) – mainly for lexicographic purposes– or identification of MWE occurrences in corpora (Savary et al., 2017). However, some research has also been conducted into improving Machine Translation (MT) quality by enhancing MWE processing (Kordoni and Simova, 2014; Seretan, 2014). Meanwhile, a considerable amount of resources have been created for several languages, including MWE lists, lexicons and MWE-annotated treebanks (Losnegaard et al., 2016).

Concerning Basque phraseology, research has been done both to describe some linguistic phenomena and to develop NLP tools (Alegria et al., 2004; Gurrutxaga and Alegria, 2012), but researchers have had an almost exclusively monolingual perspective. Thus, our aim is, on the one hand, to analyse how MWEs are translated, and, on the other hand, to propose a method to improve their computational treatment in bilingual tools.

In this paper, we will present Konbitzul, a database of verb+noun MWEs in Spanish and Basque. As well as working as a bilingual phraseological dictionary, the database contains linguistic information which is useful for NLP-related tasks, notably for Parsing and MT.

We will start by introducing the database, including: the verb+noun MWEs collected (Section 2.1.), how linguistic information is included in the database (Section 2.2.), and how the interface is structured (Section 2.3.). We will then go on to explain what the database can be used for: as a helpful tool for MWE identification (Section 3.1.), or as a resource to improve MT quality (Section 3.2.). Finally, we will discuss some conclusions and ongoing and future work.

2. The database

Konbitzul is a database which can be publicly accessed online (Section 2.3.). It currently comprises 3,195 Spanish verb+noun MWEs (along with 7,132 translations) and 2,954 Basque noun+verb MWEs (along with 6,392 translations).

The MWEs in the database were gathered from two main sources: the Elhuyar Spanish-Basque and Basque-Spanish dictionaries[1] and the DiCE dictionary of Spanish collocations[2] (Vincze et al., 2011). However, the detabase being part of an ongoing project, additional sources will probably be used in the future, such as a list of Basque MWEs extracted from corpora by using Gurrutxaga *et al.*'s method (Gurrutxaga and Alegria, 2011). NLP-applicable linguistic information was added afterwards. As this was done in several phases, the amount of linguistic data provided varies from one MWE to another. More information about the analysis will be given in the following paragraphs.

2.1. Verb+Noun MWEs in Spanish and Basque

Whereas Spanish is a romance language, Basque is a non-indoeuropean language which does not belong to any known family. Their typological features are very different:

- Spanish is SVO-ordered, head-initial, fusional, and uses prepositions

[1] http://hiztegiak.elhuyar.eus
[2] www.dicesp.com

- Basque is canonically SOV-ordered[3], head-final, agglutinative, and uses postpositions

Thus, given that they are so dissimilar in such fundamental aspects, it is not surprising that both languages differ considerably in phraseology as well, as typological features directly affect the way in which languages combine words. The MWEs collected in Konbitzul are all made up of a verb and a noun. The Spanish ones can have a preposition and/or a determiner in-between (Example 4), and similarly, Basque noun phrases can have case markers or postpositions attached (Example 5).

(4) A. *tener afecto* (V+N)
lit. *have affection* 'have affection'
B. *hacer un favor* (V+D+N)
lit. *do a favour* 'do a favour'
C. *saber de memoria* (V+P+N)
lit. *know of memory* 'know by heart'
D. *dejar a un lado* (V+P+D+N)
lit. *leave to one side* 'leave aside'

(5) A. *denbora galdu* (N.abs+V)
lit. *time lose* 'waste time'
B. *sutan egon* (N.loc+V)
lit. *fire-in be* 'be very angry'
C. *aurrera egin* (N.alla+V)
lit. *front-to do* 'move forward'
D. *hutsetik hasi* (N.abl+V)
lit. *zero-from start* 'start from scratch'

In previous work, we showed that it is rare for a verb+noun MWE to be translated literally between Spanish and Basque (Example 6). As a matter of fact, out of the Spanish verb+noun combinations in a general bilingual dictionary, only 48.54% had a noun+verb translation in Basque, and only 10.58% were translated word-for-word.

(6) ES: *poner en libertad* (V+P+N)
lit. *put in liberty*
EU: *aske utzi* (Adv+V) / *askatu* (V)
lit. *free leave* / *(to) free*
EN: '(to) release'

As for Basque into Spanish (Example 7), the gap was even bigger: only 30.85% of the noun+verb combinations were translated by a verb and a noun, and only 8.64% of the translations were literal.

(7) EU: *zin egin* (N.abs+V)
lit. *oath do*
ES: *jurar* (V)
lit. *swear*
EN: 'swear'

[3]Note that, although Basque is classified as an SOV language, it is often said to be free-ordered, as word order can be freely altered for emphasis.

2.2. Methodology for analysing linguistic data

As we have already mentioned, most of the linguistic information in Konbitzul is analysed and structured so that it can later be used in NLP tools. The collection and analysis of the MWEs was done in five phases: during the first three, the annotation was mainly manual; the last two are the result of our attempt to automatize the previous manual work. We will now briefly explain the phases one by one.

Phase 1. All the entries consisting of a verb and a noun were gathered from the Elhuyar Spanish-Basque and Basque-Spanish dictionaries. Basic information about them was analysed semi-automatically: morphological structure, number and definiteness of the noun phrases (NPs), and whether the nouns and the verbs in both languages were regular translations or not. This information was used to make some preliminary estimations about the irregularities which occur when translating MWEs between Spanish and Basque (Inurrieta et al., in print).

Phase 2. After having looked at the frequencies of the MWEs analysed in Phase 1, the 150 most common combinations in Spanish were selected for more in-depth study, which would then be used for an identification experiment (Section 3.1.). The combinations were classified into lexical-semantic and morphosyntactic groups, and further morphosyntactic data was examined, such as: possible determiners inside the NPs, variations in number and definiteness, possibility of altering word order, etc. Detailed information about this can be found in (Inurrieta et al., 2016).

Phase 3. A Basque translation was manually given to each of the combinations analysed in Phase 2, and information about this translation was examined: lexical components, whether the number and/or definiteness needed changing between one language and the other, cases in which the translation was not made up of a noun and a verb, etc. The data obtained from this phase was later tested and evaluated in an MT system (Inurrieta et al., 2017).

Phase 4. Once having seen that the analysed information was helpful for MWE identification, the next step was to semi-automatize the linguistic analysis, so that our method could be useful on a bigger scale. We used both the list of Spanish verb+noun combinations from the Elhuyar Spanish-Basque dictionary and a new one obtained from the DiCE collocation dictionary (Vincze et al., 2011). Some data about the features analysed in Phase 2 was automatically extracted from both monolingual and parallel corpora, and this information was employed to group the MWEs according to fifteen morphosyntactic patterns: those never occurring with a determiner, those only used in the plural form, those where the pronominal form of the verb is especial, those which can be freely altered just like any other word combination, etc. We are now in the process of testing this information in MWE identification within parsing.

Phase 5. Parallel corpora were used to obtain translation candidates for the MWEs, by word and n-gram alignment. For each MWE, one of the translations was chosen as the most suitable for MT (usually the most common one requiring less grammatical changes when transferring it from the

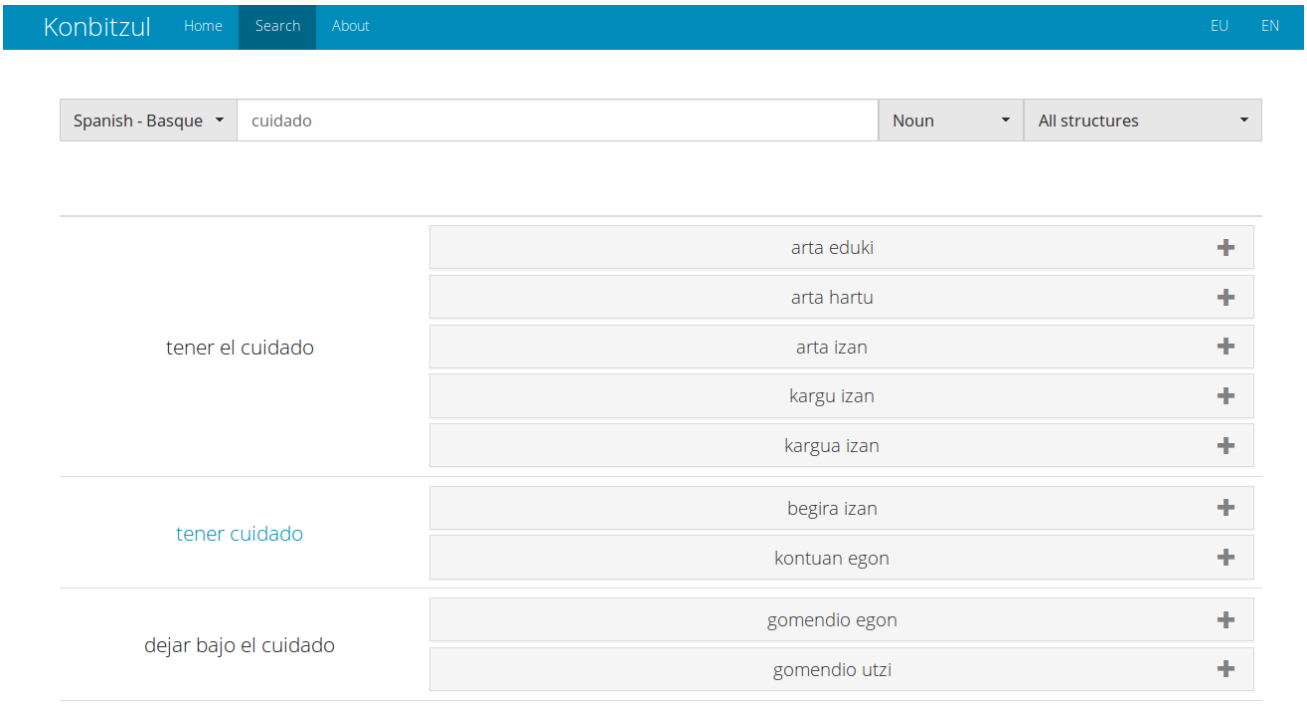


Figure 1: The Konbitzul database's interface. The noun *cuidado* (care) is searched, and three combinations are shown along with their possible translations: *tener el cuidado* (lit. have the care, 'be careful'), *tener cuidado* (lit. have care, 'be careful') and *dejar bajo el cuidado* (lit. leave under the care, 'leave in charge of').

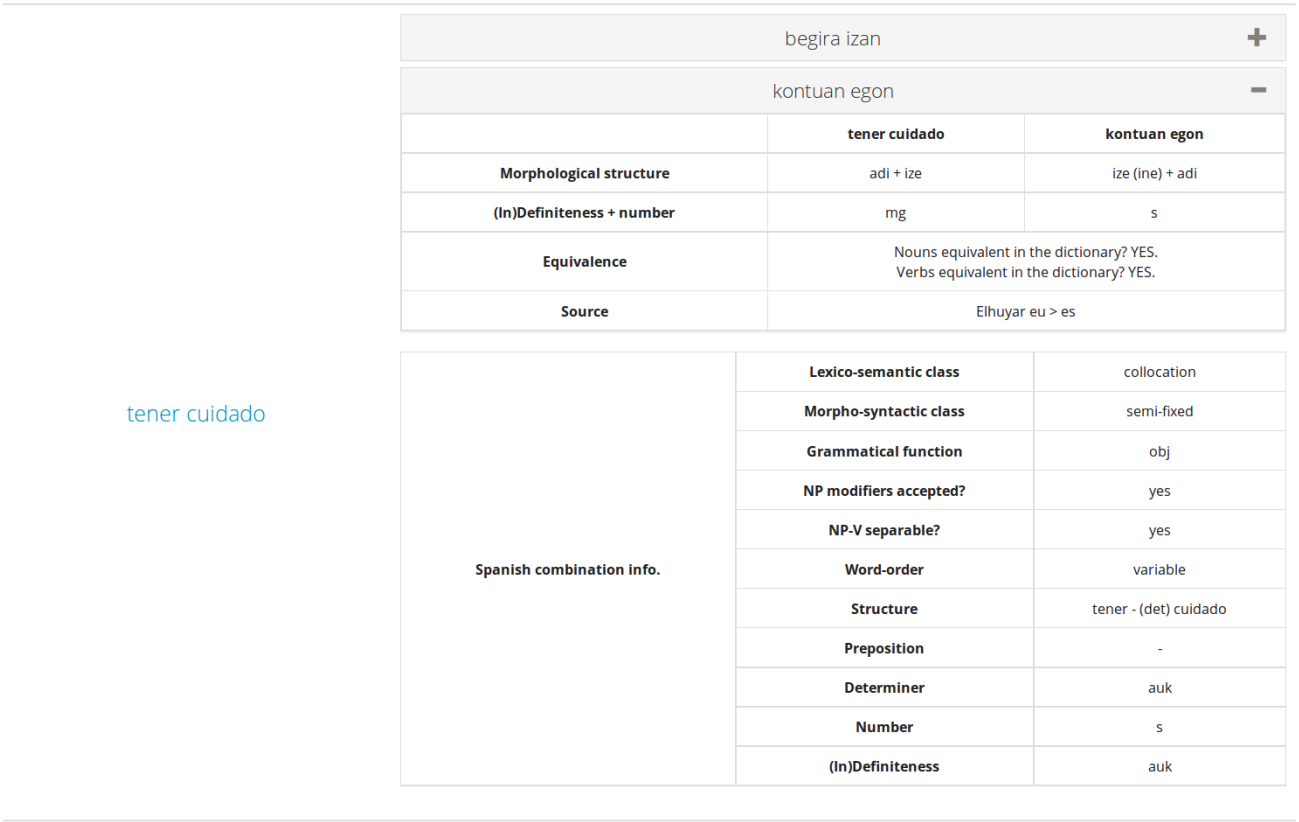


Figure 2: An example of how linguistic information is shown. Two tables are opened after clicking both on the entry *tener cuidado* and on the plus button besides the translation *kontuan egon* (lit. care-in be, 'be careful).

source to the target language). Then, lexical and grammatical information was added. This information is yet to be tested in MT.

2.3. The interface

The database can be publicly accessed at `http://ixa2.si.ehu.eus/konbitzul`. Combinations can be found by typing verb or noun lemmas or full combinations, and morphosyntactic structures can be filtered as well. The results matching the query are listed along with one or more possible translations (Figure 1).

By clicking on the plus button beside each translation, the basic information analysed in Phase 1 can be seen (top table in Figure 2). When the Spanish entry is in a different colour, it means that the combination was (either manually or semi-automatically) analysed in Phase 2 or 4; this information can be seen by clicking on the entry (bottom table in Figure 2). Finally, when one of the translations is also differently coloured and clickable, it means that this translation was marked as the most appropriate for MT (Phases 3 and 5).

3. Applications

Konbitzul was originally created as an NLP-applicable resource. However, with its user-friendly interface, it can simply be used as a phraseological dictionary as well.

In Sections 3.1. and 3.2., two past experiments will be explained, to show the potential impact of the analysed linguistic data on MWE identification and MT.

3.1. MWE identification

Concerning identification, one of the major problems of MWEs is their morphosyntactic variability (Example 8). The most straightforward means of identification is to try to match word sequences against dictionary entries; however, this method falls short in most cases, especially when it comes to verbal MWEs, which tend to have multiple morphosyntactic variants (Savary et al., 2017).

(8) *dar clase* lit. *give lecture*
dar una clase lit. *give one lecture*
dar clases lit. *give lectures*
la clase dada lit. *the lecture given*

In previous work (Inurrieta et al., 2016), the linguistic information in Konbitzul was used to help identify occurrences of a list of verb+noun MWEs in corpora. To be precise, the MWEs were the same ones studied during the second phase of the analysis presented in Section 2.2..

Two identification methods were compared: (A) that used by the Freeling parser (Padró and Stanilovsky, 2012), which only searches for non-separable occurrences of MWEs, and (B) a new one combining the linguistic data in Konbitzul with the chunking and dependency information provided by the parser. The results clearly showed that method B was considerably better, as it identified 28% more occurrences than method A, with a precision score as high as 98% (as opposed to 99%).

3.2. Machine Translation

Likewise, another experiment was undertaken to see whether the information in Konbitzul could improve MT quality. Matxin was used for this study, a rule-based system for Spanish-Basque (Mayor et al., 2011).

As with any rule-based system, Matxin works in three phases: analysis, transfer and generation. The data gathered from Konbitzul was added both to the analysis and transfer phases. Firstly, identification of MWEs was carried out as explained in Section 3.1., and then, lexical and grammatical information about the translation of each MWE (analysed in Phase 3 of Section 2.2.) was used.

The experiment resulted in an increase of 3% in BLEU score (Papineni et al., 2002). In addition, a manual evaluation by three experts was carried out in a controlled corpus, and it was concluded that the new translation was better than the old one in 78.6% of the cases (Inurrieta et al., 2017). Once again, this proves that the kind of linguistic information in the database is helpful for NLP purposes.

4. Conclusion

Konbitzul is an open-source online database of verb+noun MWEs in Spanish and Basque. It currently comprises 6,149 entries in all, which all have one or more translation and rich NLP-applicable linguistic information. Part was added manually, and the reminder is the result of a semi-automatic analysis.

Experiments have confirmed that the information in the database is helpful for NLP tools. Due to the large amount of MWEs requiring a non-regular translation, the database is of special interest for the area of MT, as well as being a useful resource to help identifying multiple morphosyntactic variants of MWEs in text.

As this is an ongoing project, the database is constantly being updated with further MWEs, translations and linguistic information. At the same time, new experiments are being undertaken both to semi-automatize the linguistic analysis and to test the automatic information in NLP tools.

5. Acknowledgements

Uxoa Iñurrieta's doctoral research is funded by the Spanish Ministry of Economy and Competitiveness (BES-2013-066372). This work was carried out in the context of the SKATeR (TIN2012-38584-C06-02) and TADEEP (TIN2015-70214-P) projects.

6. Bibliographical References

Alegria, I., Ansa, O., Artola, X., Ezeiza, N., Gojenola, K., and Urizar, R. (2004). Representation and treatment of multiword expressions in basque. In *Proceedings of the Workshop on Multiword Expressions: Integrating Processing*, pages 48–55. Association for Computational Linguistics.

Baldwin, T. and Kim, S. N. (2010). Multiword expressions. In *Handbook of Natural Language Processing, Second Edition*, pages 267–292. Chapman and Hall/CRC.

Gurrutxaga, A. and Alegria, I. (2011). Automatic extraction of NV expressions in Basque: basic issues on cooccurrence techniques. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pages 2–7. Association for Computational Linguistics.

Gurrutxaga, A. and Alegria, I. (2012). Measuring the compositionality of NV expressions in Basque by means of distributional similarity techniques. In *LREC*, pages 2389–2394.

Inurrieta, U., Díaz de Ilarraza, A., Labaka, G., Sarasola, K., Aduriz, I., and Carroll, J. A. (2016). Using linguistic data for English and Spanish verb-noun combination identification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): Technical Papers*, pages 857–867.

Inurrieta, U., Aduriz, I., Díaz de Ilarraza, A., Labaka, G., and Sarasola, K. (2017). Rule-based translation of Spanish verb+noun combinations into Basque. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 149–154.

Inurrieta, U., Aduriz, I., Díaz de Ilarraza, A., Labaka, G., and Sarasola, K. (in print). Analysing linguistic information about word combinations for a Spanish-Basque rule-based machine translation system. In Ruslan Mitkov, et al., editors, *Multiword Units in Machine Translation and Translation Technologies*, pages 41–59. John Benjamins Publishing Company.

Kordoni, V. and Simova, I. (2014). Multiword expressions in machine translation. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1208–1211.

Losnegaard, G., Sangati, F., Parra Escartín, C., Savary, A., Bargmann, S., and Monti, J. (2016). Parseme survey on MWE resources. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016), Paris, France*, pages 2299–2306. European Language Resources Association (ELRA).

Mayor, A., Alegría, I., Díaz de Ilarraza, A., Labaka, G., Lersundi, M., and Sarasola, K. (2011). Matxin, an open-source rule-based machine translation system for Basque. *Machine translation*, 25(1):53–82.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 2473–2479.

Papineni, K., Roukos, S., Ward, T., and Zhug, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Ramisch, C. (2015). *Multiword expressions acquisition*. Springer.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: a pain in the neck for NLP. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 1–15. Springer.

Savary, A., Sailer, M., Parmentier, Y., Rosner, M., Rosén, V., Przepiórkowski, A., Krstev, C., Vincze, V., Wójtowicz, B., Losnegaard Smørdal, G., et al. (2015). PARSEME: PARSing and Multiword Expressions within a European multilingual network. In *Proceedings of the 7th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC 2015)*.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., Qasemizadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., et al. (2017). The PARSEME Shared Task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47.

Seretan, V. (2014). On collocations and their interaction with parsing and translation. In *Informatics*, volume 1, pages 11–31. Multidisciplinary Digital Publishing Institute.

Vincze, O., Mosqueira, E., and Alonso Ramos, M. (2011). An online collocation dictionary of Spanish. In *Proceedings of the 5th International Conference on Meaning-Text Theory*, pages 275–286.

A Multilingual Test Collection for the Semantic Search of Entity Categories

Juliano Efson Sales[†], Siamak Barzegar[§], Wellington Franco[‡], Bernhard Bermeitinger[†], Tiago Cunha[¶], Brian Davis[§∓], André Freitas[†δ] and Siegfried Handschuh[†]

[†]Department of Computer Science and Mathematics - University of Passau, Germany
[§]Insight Centre for Data Analytics - National University of Ireland, Galway
[‡]Federal University of Ceará, Campus Crateús, Brazil
[¶]University of the International Integration of the Afro-Brazilian Lusophony, Brazil
[∓]Department of Computer Science - Maynooth University, Ireland
[δ]School of Computer Science - The University of Manchester, United Kingdom
{juliano-sales, bernhard.bermeitinger, siegfried.handschuh}@uni-passau.de
siamak.barzegar@insight-centre.org, wellington@crateus.ufc.br
tiagotmc@unilab.edu.br, brian.davis@mu.ie, andre.freitas@manchester.ac.uk

Abstract

Humans naturally organise and classify the world into sets and categories. These categories expressed in natural language are present in all data artefacts from structured to unstructured data and play a fundamental role as tags, dataset predicates or ontology attributes. A better understanding of the category syntactic structure and how to match them semantically is a fundamental problem in the computational linguistics domain. Despite the high popularity of *entity search*, entity categories have not been receiving equivalent attention. This paper aims to present the task of *semantic search of entity categories* by defining, developing and making publicly available a multilingual test collection comprehending English, Portuguese and German. The test collections were designed to meet the demands of the entity search community in providing *more representative* and *semantically complex* query sets. In addition, we also provide comparative baselines and a brief analysis of the results.

Keywords: semantic search, category search, paraphrasing, entity search, multilingual test collection

Lang.	***Category***
EN	American architectural styles
EN	French female artistic gymnasts
EN	Brazilian aerospace conglomerate
PT	Alfabetos derivados do latino
PT	Ordens e congregações religiosas católicas
PT	Primeiros-ministros de Portugal
DE	Ortsteil von Anröchte
DE	Soziale Bewegung als Thema
DE	Teilnehmer an Der Bachelor

Table 1: Examples of categories in English (EN), Portuguese (PT) and German (DE).

1. Introduction

Well-defined tasks and test collections are fundamental resources to allow reproducibility and comparability in information retrieval in general, and entity search in particular (Jones, 1981). As entities are the main target of a vast amount of search queries on the Web (Pound et al., 2010), over the years, a mature research community has emerged around the *entity search* domain.

Despite the considerable number of challenges and campaigns released in the field (Elbedweihy et al., 2015), in recent years, the entity search community has encouraged the development of new tasks. According to Balog and Neumayer (2013) three action points were signed as priorities:

1. (i) getting more representative information needs and favouring long queries over short ones;
2. (ii) limiting search to a smaller, fixed set of entity types (as opposed to arbitrary types of entities); and
3. (iii) using test collections that integrate both structured and unstructured information about entities.

In order to suit at least two of these priorities, this paper aims at presenting the task of *semantic search of entity categories* by defining, developing and making publicly available test collections, and providing a comparative analysis on the baseline results. In addition to English, the test collection comprehends two more languages to contribute to the semantic research community targeting Portuguese and German.

2. The Semantic Search of Entity Categories

Each type or attribute generally describes a singular characteristic of an entity. The nouns *president* and *monument*, the adjective *urban* and named entity *United States* are examples of entity types and attributes found in structured data. Users commonly refer to entities by combining a set of these characteristics to create richer descriptive categories, e.g. *President of the United States* and *Urban monument*. The following text excerpt shows a real example:

> "*Franklin Delano Roosevelt (January 30, 1882 - April 12, 1945), commonly known as FDR, was an American statesman and political leader who served as the 32nd* ***President of the United States****, from 1933 to 1945.*"[1]

President of the United States is a *natural language entity category* labelling the entity *Franklin Delano Roosevelt*.

[1]Extracted from `https://en.wikipedia.org/wiki/Theodore_Roosevelt`

Lang.	*Target entity category*	*Paraphrases*
EN	Irish Manuscripts	Writings from Ireland
EN	Dermatologic Drugs	Pharmaceuticals for the Skin
PT	Países e territórios de língua oficial inglesa	Comunidade anglofônica
PT	Grandes Mestres de xadrez	Enxadristas célebres
DE	Literatur über den Islam	Islamzentrische Bücher
DE	Station der Toronto Subway	Haltestelle der U-Bahn in Toronto

Table 2: Examples of paraphrases in English (EN), Portuguese (PT) and German (DE).

We define entity category as a concise and descriptive structured predicate described in natural language that combines one or more types/attributes of an entity.
In addition to their occurrence in unstructured data, entity categories are also available in the form of structured data. DBpedia (Auer et al., 2007) associates descriptive categories to entities representing in the form of Yago properties (Suchanek et al., 2007), which are available in all languages supported by Wikipedia. This information is extracted from the categories present in the Wikipedia articles which are freely described by the Wikipedia community.
Table 1 presents a list of entity categories extracted from the Yago/DBpedia dataset.
The use of natural language to create such a category allows users to express and potentially search for the same (or close) concept using different words. For example, a ***Brazilian aerospace conglomerate*** is also described as ***Brazilian Planemaker***[2] and categorised as ***Aircraft manufacturers of Brazil***[3]. Having the ability to match paraphrases of complex nominals, allows users to find other relevant categories for different lexical expressions. The current task is designed to evaluate the ability of a system to recognise *paraphrases of entity categories* written in English, Portuguese or German.

2.1. Query Types

According to Pound et al. (2010), in the context of *Entity Search*, there are five types of queries summarised as follows:

- **Entity query**: intending to find a particular entity. Expected results are entities corresponding to some disambiguation of the query entity.
- **Type query**: intending to find entities of a particular type or class. Expected results are entities that are instances of the specified type, or an identifier of the type itself.
- **Attribute query**: intending to find values of a particular attribute of an entity or type. Expected results are the values of an attribute specified in the query.
- **Relation query**: intending to find how two or more entities or types are related. Expected results are the one or more relationships among the query entities or types.
- **Other keyword query**: the query intent is described by some keywords that do not fit into any of the above categories. Expected results are resources providing relevant information.

Complex queries combining characteristics of *type queries* and *attribute queries* can benefit from the use of the natural language categories associated with the entities. The categories can provide a shortcut between the natural language query and the target entities. For example, the INEX-LD Challenge (Wang et al., 2012) defines natural language queries and lists associated entities. For the query "*bicycle sport races*" several relevant entities hold the Yago category *Cycling Competitions*. Intuitively we can assume that *bicycle sport races* and *cycling competitions* are equivalent paraphrases. Creating a mechanism able to pair them makes a shortcut between a natural language expression and a set of entities.
The semantic search of entity categories can be seen as a bridge between unstructured and structured data. In addition, they also suggest which kinds of types/attributes are relevant to create descriptive compositions from the user's point of view. For example, the combination of `dbo:occupation` and `dbo:birthPlace` generally creates *commonly used* descriptive categories (e.g. *French Poets*).

3. Test Collection

The test collections comprehend three knowledge bases of about 345,000 entity categories for English, 105,000 for Portuguese and 235,000 for German, which were created based on the set of DBpedia categories. From these sets, we chose a subset of 110 categories for each language to be part of the query sets. The categories in query sets were chose randomly and filtered later to ensure that they vary in size, number of place/demonym references, number of temporal expressions and different noun phrase components, in order to ensure a high semantic variety in the queries.
Examples of categories in the English query set are:

- categories having different sizes to represent different degrees of word compositionally. For example, *Prehistoric Canines* (two terms); *Victims of Helicopter Accidents or Incidents in the United States* (ten terms).
- categories containing references to places/demonym and temporal expressions. For example, *French Senators Of The Second Empire*; *Political Movements in Italy*.

[2]See the magazine article *Brazilian Planemaker Unveils Its Biggest Military Jet Yet* published by Business Insider.
[3]See the Wikipedia category *Aircraft manufacturers of Brazil*.

Language	*Inter-rater agreement*	*Recall*		*MRR*	
		Top 10	*Top 20*	*Top 10*	*Top 20*
English (EN)	0.9075	0.2806	0.2806	0.3096	0.3143
Portuguese (PT)	0.9489	0.3420	0.3907	0.4641	0.4665
German (DE)	0.9960	0.4487	0.4725	0.6760	0.6768

Table 3: Inter-rater agreement, recall and MRR results considering all relevant categories.

- categories containing different kinds of noun phase components, such as nouns, adjectives, and verbs. For example *Recurring Events Established in 1875*, *Magazines with Year of Establishment Missing*.

We grouped the query sets in blocks of 10 categories and asked volunteers fluent or native in English, Portuguese and German to suggest paraphrases for them. They were instructed to describe the same (or close) meaning using preferably different words and different syntactical structures. After that, we applied a curation process conducted by two researchers to assess the quality of the paraphrases. In the end, we accepted a set of 220 paraphrases for English, 229 for Portuguese and 169 for German, which refer respectively to 98, 96 and 71 categories out of the initial set of 110.

Table 2 shows examples of entity categories and their associated paraphrases proposed by the volunteers in English, Portuguese and German. The task aims at retrieving the original category when querying using its paraphrase.

3.1. Evaluation and Relevance Judgements

The task is defined as an information retrieval problem:

- **Statement**: Let T be a set of entity categories, called target set, and (x, y) a pair of entity categories where $x \in T$, $y \notin T$ and y is a paraphrase of x, meaning that y is a semantic approximation of x, represented by $y \approx x$.
- **Search Procedure**: Let f be a semantic search procedure defined by $L = f(T, y)$ where $L = (T, \geq)$, which means L is an ordered set of entity categories.
- **Evaluation Procedure**: Let g be an evaluation procedure, defined by $r = g(L, x)$ where r is the ranking position of x in the list L.

As more than one target category can be relevant to a query, we also designed a second evaluation setting. In this new procedure, for each query, we retrieved the first 50 results using the method presented in Section 4. to be classified as one of the following judgements: *not relevant*, *relevant* and *highly relevant*, being the last class exclusively used to define the original target category that originated the paraphrase. Each result set was assessed by two fluent or native judges who were instructed to point as relevant those categories that he or she would be interested when searching for the given query. After the judgement process, more than 23,000 categories were analysed. Table 3 shows the inter-rater agreement for each language.

The task allows two evaluation settings. In the first, only the *highly relevant* category is accepted as a positive retrieval, assessing a system strictly by guided paraphrasing as formalised earlier. In the second, categories classified as *relevant* are also considered, allowing a broader evaluation.

Lang.	*Recall*		*MRR*	
	Top 10	*Top 20*	*Top 10*	*Top 20*
EN	0.2922	0.3287	0.1549	0.1575
PT	0.4736	0.5219	0.3151	0.3183
DE	0.7976	0.8392	0.5960	0.5988

Table 4: Recall and MRR results considering only the *highly relevant* categories.

4. Baselines and Results

To provide comparative baselines we implemented a method based on *distributional semantics* to cope with the search of entity categories, whose vectors were generated from unstructured text corpora.

Distributional semantics is based on the hypothesis that words co-occurring in similar contexts tend to have similar meaning (Harris, 1954). Distributional semantics provides representations of the meaning of words in a high-dimensional vector space, which is generated by analysing large-scale text corpora (Turney and Pantel, 2010). The simplification of the meaning representation model supports computation of semantic similarity between two terms by calculating the cosine similarity of their vectors. The baselines applied the Skip-gram (Mikolov et al., 2013) vector space model generated from the Wikipedia 2014 corpora. We pre-processed the corpora lower-casing and stemming each token using the Porter Algorithm (Porter, 1997) and generated the distributional model using the default parameters.

Our baseline is the *sum-algebraic-based method*, where entity categories are compared by an algebraic operation that sums up the vector's components using the resulting vector to calculate the cosine similarity. We developed the experiments with the support of Indra (a distributional semantics tool) (Sales et al., 2018b; Freitas et al., 2016).

The evaluation is applied in two scenarios. The first considers the Top-10 results of each execution and the second considers the Top-20. This assumption makes *precision* a redundant indicator since it can be derived from *recall*. So, the analysis measures recall and mean reciprocal rank (MRR). This methodology of evaluation follows the same strategy used in (Sales et al., 2016) and (Sales et al., 2018a).

Table 4 shows the recall and MRR considering only the *highly relevant* categories. This is the preferable experimental setting, since it evaluates the ability to identify the paraphrases proposed by the volunteers, ignoring any other potential relevant result.

Query Set	Size (# of Queries)
Existing Query Sets	
INEX-XER (Demartini et al., 2010)	55
TREC Entity (Balog et al., 2009)	17
SemSearch ES (Blanco et al., 2011; Halpin et al., 2010)	130
SemSearch LS (Blanco et al., 2011)	43
QALD-2 (Lopez et al., 2013)	140
INEX-LD (Wang et al., 2012)	100
Proposed Query Sets	
Entity Categories English (EN)	220
Entity Categories Portuguese (PT)	229
Entity Categories German (DE)	169

Table 5: Query sets and their sizes in number of queries.

The results show that the proposed baseline performs significantly different across languages. Considering the Top-20 scenario, while the algorithm retrieves about 32% of the English paraphrases, in the German dataset, this rate jumps to more than 82%. This occurs because many of the German paraphrases were described using words that share the same linguistic root of the original category. Furthermore, the English target dataset is significantly larger (345,000 against 235,000) which increases the probability of returning false-positive categories. For the Portuguese dataset, despite the paraphrases were constructed using a rich vocabulary, its target dataset is less than one-third of the English dataset (105,000), which put their results consistently between the other two languages.

Table 3 shows the evaluation considering all relevant categories. In this experiment, in addition to the original target category, we also consider as relevant those categories appointed by both judges simultaneously.

This new scenario of evaluation significantly differs from the previous, specially considering the German language. As we increase the set of relevant categories, the queries do not have the same linguistic roots as before, and so the recall decreases significantly. As both the English and Portuguese languages had already rich paraphrases, their recall did not experienced the same gap. The still higher retrieval score of the Portuguese language is again expressed for its smaller target dataset. Both scenarios of evaluation consistently points the German, Portuguese and English in better positions as shown by the MRR measure.

The dataset and the source code are respectively available at `https://rebrand.ly/cat-paraphases` and `https://rebrand.ly/cat-source-code`.

5. Related Work

Searching of entity categories was not exploited previously as a formatted task. Currently, the most related task is the entity search, which is commonly evaluated using the five query types generated in the context of challenges and campaigns.

Balog and Neumayer (2013) grouped those query sets, normalising their results to point to DBpedia instances.

The XML Entity ranking track (XER) discusses the standardisation of the evaluation procedures for entity retrieval and provides a large dataset sample in which the Wikipedia is used as an underlying collection (Demartini et al., 2010). The track explores two main tasks: Entity Ranking (ER) and Entity List Completion (LC), both of them using the Wikipedia 2009 XML data.

Blanco et al. (2011) present the entity search over Linked Data with keyword queries related to entities or their description. In comparison to TREC 2010 and INEX-XER tracks searches over structured data in RDF rather than unstructured data and XML as a data format, Respectively.

The goal of INEX-LD (Wang et al., 2012) was to investigate retrieval techniques over a combination of textual (wikipedia) and structured data (RDF). Lopez et al. (2013) aimed to present the faults and failures of question answering systems as interfaces to query linked data sources. The SemSearch ES challenge (Blanco et al., 2011; Halpin et al., 2010) in comparison searches over structured data in RDF rather than unstructured data and XML as the previous tasks.

Although these works deal with the entity searching studies, their focus rely on issues of evaluation and shortcomings of different interface to query systems and not aim the study on investigating entity categories. Table 5 shows the size of each query set, along with the size of the query sets proposed in our work.

Sales et al. (2016) proposed a compositional-distributional semantic model to search English entity categories whose syntactic structure plays a fundamental role in composing partial semantic relatedness scores. The method identifies the core concept behind the category and use it to guide the search. Their results, however, cannot be directly compared, since Sales et al. (2016) explores a different dataset.

6. Summary

Despite the high popularity of entity search, entity categories have not been receiving similar attention. In this paper, we shed some light on entitty category descriptors (complex nominals) by presenting the task of semantic search of entity categories and by making it publicly available. The test collections cover three different languages (English, Portuguese and German) and includes baselines. The test collections were designed to meet the demands of the entity search community in providing *more representative* and *long queries* and also by *integrating both structured and unstructured information about entities* (Balog and Neumayer, 2013).

The analysis of the semantic phenomena associated with interpreting natural language entity category allows a focused understanding of how humans define complex predicates and how systems can address complex compositions of predicates. Additionally, the ability to semantically interpret categories and to cope with its associated meaning variations plays a fundamental role in different semantic tasks including *entity search*, *schema-agnostic queries*, *question answering systems* and *text entailment* (Chen et al., 2016; Sales et al., 2016; Freitas, 2015).

However, this task has not been sufficiently individuated, becoming implicit in all of these tasks and not receiving the adequate focus. As a consequence, most discussion on how to cope with semantic variation of categories has been limited in the literature. The semantic search of entity categories can define a new bridge between unstructured and structured data.

7. Acknowledgment

This publication has emanated from research partially supported by the National Council for Scientific and Technological Development, Brazil (CNPq).

8. References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z., (2007). *DBpedia: A Nucleus for a Web of Open Data*, pages 722–735. Springer, Berlin, Heidelberg.

Balog, K. and Neumayer, R. (2013). A test collection for entity search in DBpedia. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 737–740, New York, USA. ACM.

Balog, K., de Vries, A. P., Serdyukov, P., Thomas, P., and Westerveld, T. (2009). Overview of the TREC 2009 entity track. In *TREC 2009 Working Notes*. NIST, November.

Blanco, R., Halpin, H., Herzig, D. M., Mika, P., Pound, J., Thompson, H. S., and Tran, D. T. (2011). Entity search evaluation over structured web data. In *Proceedings of the 1st International Workshop on Entity-Oriented Search*, Beijing, China, July.

Chen, J., Xiong, C., and Callan, J. (2016). An empirical study of learning to rank for entity search. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, pages 737–740, New York, NY, USA. ACM.

Demartini, G., Iofciu, T., and De Vries, A. P. (2010). Overview of the INEX 2009 entity ranking track. In *Proceedings of the Focused Retrieval and Evaluation, and 8th International Conference on Initiative for the Evaluation of XML Retrieval*, INEX'09, pages 254–264, Berlin, Heidelberg. Springer-Verlag.

Elbedweihy, K. M., Wrigley, S. N., Clough, P., and Ciravegna, F. (2015). An overview of semantic search evaluation initiatives. *Web Semantics: Science, Services and Agents on the World Wide Web*, 30(0).

Freitas, A., Barzegar, S., Sales, J. E., Handschuh, S., and Davis, B., (2016). *Semantic Relatedness for All (Languages): A Comparative Analysis of Multilingual Semantic Relatedness Using Machine Translation*, pages 212–222. Springer International Publishing, Cham.

Freitas, A. (2015). *Schema-agnositc queries over large-schema databases: a distributional semantics approach.* Ph.D. thesis, Digital Enterprise Research Institute (DERI), National University of Ireland, Galway.

Halpin, H., Herzig, D. M., Mika, P., Blanco, R., Pound, J., Thompson, H. S., and Tran, D. T. (2010). Evaluating ad-hoc object retrieval. In *Proceedings of the International Workshop on Evaluation of Semantic Technologies*, Shanghai, China, November.

Harris, Z. (1954). Distributional structure. *Word*, 10(23):146–162.

Karen Spärck Jones, editor. (1981). *Information Retrieval Experiment*. Butterworths.

Lopez, V., Unger, C., Cimiano, P., and Motta, E. (2013). Evaluating question answering over linked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 21:3 – 13. Special Issue on Evaluation of Semantic Technologies.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *ICLR Workshop Papers*.

Porter, M. F. (1997). Readings in information retrieval. chapter An Algorithm for Suffix Stripping, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Pound, J., Mika, P., and Zaragoza, H. (2010). Ad-hoc object retrieval in the web of data. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 771–780, New York, NY, USA. ACM.

Sales, J. E., Freitas, A., Davis, B., and Handschuh, S. (2016). A compositional-distributional semantic model for searching complex entity categories. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, *SEM@ACL 2016, Berlin, Germany, 11-12 August 2016*.

Sales, J. E., Freitas, A., and Handschuh, S. (2018a). An open vocabulary semantic parser for end-user programming using natural language. In *2018 IEEE Twelfth International Conference on Semantic Computing (ICSC)*, Jan.

Sales, J. E., Souza, L., Barzegar, S., Davis, B., Freitas, A., and Handschuh, S. (2018b). Indra: A word embedding and semantic relatedness server. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA. ACM.

Turney, P. D. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January.

Wang, Q., Kamps, J., Camps, G. R., Marx, M., Schuth, A., Theobald, M., Gurajada, S., and Mishra, A. (2012).

Overview of the INEX 2012 linked data track. In *CLEF 2012 Evaluation Labs and Workshop*, pages 1–13, Rome, Italy.

Towards the Inference of Semantic Relations in Complex Nominals: a Pilot Study

Melania Cabezas-García, Pilar León-Araúz
University of Granada
C/ Buensuceso, 11 18002 Granada (Spain)
{melaniacabezas, pleon}@ugr.es

Abstract

Complex nominals (CNs) (e.g. *wind turbine*) are very common in English specialized texts (Nakov, 2013). However, all too frequently they show similar external forms but encode different semantic relations because of noun packing. This paper describes the use of paraphrases that convey the conceptual content of English two-term CNs (Nakov and Hearst, 2006) in the domain of environmental science. The semantic analysis of CNs was complemented by the use of knowledge patterns (KPs), which are lexico-syntactic patterns that usually convey semantic relations in real texts (Meyer, 2001; Marshman, 2006). Furthermore, the constituents of CNs were semantically annotated with conceptual categories (e.g. *beach* [LANDFORM] *erosion* [PROCESS]) with a view to disambiguating the semantic relation between the constituents of the CN and developing a procedure to infer the semantic relations in these multi-word terms. The results showed that the combination of KPs and paraphrases is a helpful approach to the semantics of CNs. Accordingly, the conceptual annotation of the constituents of CNs revealed similar patterns in the formation of these complex terms, which can lead to the inference of concealed semantic relations.

Keywords: Complex Nominal, Semantic Relation, Terminology

1. Introduction

Complex nominals (CNs) (e.g. *wind power*) are very frequent in English specialized texts (Nakov, 2013). They are distinguished by their syntactic-semantic complexity, since at least two concepts are juxtaposed with no clear indication of the link between them (Rosario et al., 2002). This means that in CNs such as *air pollution* and *oil pollution*, which have the same external form (the head *pollution* combines with a noun modifier), different semantic relations can be established between their constituents (*has_patient* vs. *caused_by*, respectively) (Maguire et al., 2010). The root of this issue is noun packing, which can be addressed by analyzing the formation processes of CNs, involving predicate deletion (e.g. *power system*, instead of *a system produces power*) and predicate nominalization (e.g. *energy transfer*, instead of *energy is transferred*) (Levi, 1978). This paper describes the use of paraphrases conveying the conceptual content of English two-term CNs (Nakov and Hearst, 2006; Butnariu and Veale, 2008; Cabezas-García and Faber, 2017) in the specialized domain of environmental science. Verb paraphrases were used to access the concealed propositions in two-term CNs formed by predicate nominalization and verb deletion. Some of these paraphrases were based on the lexico-syntactic patterns that generally convey semantic relations in real texts (Meyer, 2001; Marshman, 2006). Our goal was to access the semantics of these multi-word terms (MWTs) in order to (i) disambiguate the semantic relation between the constituents of the CN; and (ii) develop a procedure to infer the semantic relations in these MWTs.

2. Complex Nominals and Meaning Access

Complex nominals (CNs) are expressions with a head noun preceded by a modifying element (i.e. a noun or an adjective) (Levi, 1978), e.g. *wind turbine*. CNs can be endocentric, when one term is the head and the other is its modifier (Nakov, 2013) (e.g. *power generation*). Alternatively, they can be exocentric, when the MWT is not a hyponym of one of its elements, and thus appears to lack a head (Bauer, 2008) (e.g. *fire rainbow*).

The semantic relation linking the constituents of CNs is usually implicit because of noun packing. This results in CNs with parallel external forms but different semantic relations, which sometimes can lead to interpretation difficulties. For instance, according to our world knowledge, the semantic relations underlying *fine sand* and *carbonate sand* are interpreted as an attribute of sand (its grain size, which is fine) and the composition of that type of sand (*carbonate*). Therefore, different methods of meaning access are usually employed. On the one hand, verb paraphrases (Nakov and Hearst, 2006) can be used (e.g. *beach erosion > the beach was eroded*). These paraphrases, which take the form of a predicate, its mandatory arguments, and its adjuncts (optional complements) (Tesnière, 1976), make the concealed propositions explicit and further specify the semantic relation in CNs. In that case it could be the non-hierarchical semantic relations *has_patient* or that of *erodes*, depending on the granularity of the semantic relation inventory. On the other hand, knowledge patterns (KPs) can also be used to identify semantic relations. KPs are the lexico-syntactic patterns that usually convey semantic relations in real texts (Meyer, 2001; Marshman, 2006) (e.g. erosion *takes place at/occurs/affects/threatens/impacts (on)* the beach; erosion *of/along/on/across* the beach; beach *impacted by* erosion). This paper combines the use of paraphrases and KPs with the semantic annotation of the constituents of CNs with conceptual categories (e.g. *beach* [LANDFORM] *erosion* [PROCESS]) to disambiguate the semantic relation between the constituents of the CN and develop a procedure to infer the semantic relations in these MWTs.

3. Materials

For the purposes of this study, we used an English corpus of specialized environmental texts compiled for the terminological knowledge base EcoLexicon (http://ecolexicon.ugr.es). It was composed mainly of articles and books, and comprised 67 million words on different environmental subdomains, such as Coastal Engineering, Meteorology, Geology, etc. Part of this

corpus is now available in Open Corpora (Sketch Engine)[1].

We selected CNs with a nominal or adjectival modifier that designated both entities (i.e. 56 hyponyms of *sand*) and processes (i.e. 57 hyponyms of *erosion*) in order to compare the semantic relations and conceptual categories that are characteristic of these different term types.

We identified relevant term candidates in TermoStat (http://termostat.ling.umontreal.ca/) (Drouin, 2003), namely two-term CNs that were hyponyms of *erosion* and *sand*. Then, Sketch Engine (https://www.sketchengine.co.uk/) (Kilgarriff et al., 2004) was used for term extraction, concordance analysis, paraphrase and KP search, and word sketch analysis. NooJ (http://www.nooj4nlp.net/pages/nooj.html) (Silberztein, 2003) was also employed for the semantic annotation of the constituents of CNs with conceptual categories, as explained below.

4. Corpus-based Semantic Analysis

The aim was to compare the semantic relations that are characteristic of CNs designating entities and processes. Not surprisingly, different semantic relations are established between the constituents of these types of terms. As will be shown, CNs that name a process (e.g. *sea erosion*) encode semantic relations such as *caused_by*, *has_patient*, *has_result*, etc., whereas CNs designating entities (e.g. *carbonate sand*) activate semantic relations such as *composed_of*, *has_origin*, *has_function*, etc. This is not surprising, since such constraints can be explained by the different semantic nature of entities and processes and their natural combinatorial potential (León-Araúz and Faber, 2010). However, what is not so obvious is the kind of constraints that may be inferred from the combination of each particular concept within the same CN. This suggests that conceptual categories play a role in the conceptualization of complex terms, which is directly linked to the notion of 'micro-context'. The head of a CN is considered to open slots that are filled by specific conceptual categories (Rosario et al., 2002; Maguire et al., 2010) that play a semantic role. Thus, the semantic category that is the head of the CN determines what can be done to it by means of the addition of modifiers that fill the slots opened by the head. Micro-contexts refer to this slot filling and are essential to the inference of semantic relations addressed in this study, because similar heads are considered to co-occur with related semantic categories and to evoke similar relations (Maguire et al., 2010).

Since the semantic nature of the head determines its combinatorial potential, CNs representing entities and processes must be differentiated as a first step in the development of a procedure of inference of the semantic relations in CNs. On the one hand, this pilot study focused on a set of CNs that were hyponyms of *erosion*, and thus represented processes, in order to ascertain the different semantic relations activated by the same head. These CNs were then classified according to the conceptual category of their modifiers, some of which are included in Table 1.

WATER BODY + PROCESS		
sea erosion marine erosion	fluvial erosion river erosion	stream erosion glacial erosion
PROCESS + PROCESS		
storm erosion subduction erosion	landslide erosion wind erosion	seepage erosion
LANDFORM + PROCESS		
dune erosion cliff erosion	bluff erosion slope erosion	mountain erosion delta erosion
PART OF LANDFORM + PROCESS		
toe erosion	bed erosion	berm erosion

Table 1: Example of the classification of the hyponyms of *erosion* based on the conceptual category of their modifiers.

On the other hand, an analysis of the conceptual categories combined to form the hyponyms of *sand* demonstrates that processes and entities usually combine with different conceptual categories as well as through different semantic relations. Table 2 shows the classification of some of the hyponyms of *sand* based on the conceptual category of their modifiers.

SIZE + MATERIAL	
fine sand medium sand	coarse sand
LANDFORM + MATERIAL	
beach sand dune sand	nearshore sand littoral sand
MATERIAL + MATERIAL	
carbonate sand silt sand	calcareous sand oil sand
PROCESS + MATERIAL	
nourishment sand construction sand	fill sand filter sand

Table 2: Example of the classification of the hyponyms of *sand* based on the conceptual category of their modifiers.

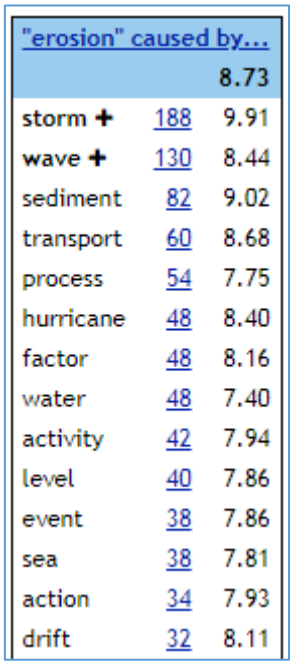
As stated earlier, although hyponymic CNs usually have the same external form, they can encode different semantic relations. At first sight, it might seem obvious that the relation held between landform-related concepts and material-related concepts, for instance, must be that of *located_at* (e.g. *beach sand* > sand *located_at* beach). However, with no world knowledge supporting our inferences, the elicitation of semantic relations may not be as straightforward. For instance, one should be knowledgeable in environmental issues in order to easily understand that sand can be used for filtering purposes in

[1] the.sketchengine.co.uk/open/

the domain of water treatment. Only then, the CN *filter sand* will thus evoke the *has_function* relation.

With a view to eliciting these relations, we performed different types of queries: (1) KP-based word sketches, (2) ws (word sketch) Corpus Query Language (CQL) queries, (3) free paraphrases, and (4) verb paraphrases.

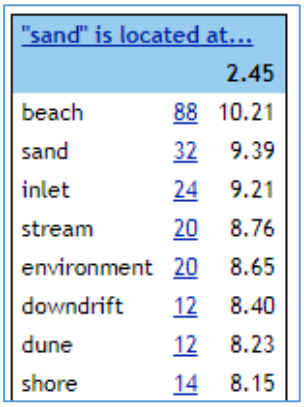
We first observed the KP-based word sketches generated by the head (*erosion* or *sand*) in Sketch Engine. These are automatic groupings of terms that specify the semantic relations between them, based on the application of KP-based sketch grammars (León-Araúz et al., 2016). Figure 1 shows an example of the main agents that cause erosion, which are usually the modifiers of its hyponymic CNs.



"erosion" caused by...		8.73
storm +	188	9.91
wave +	130	8.44
sediment	82	9.02
transport	60	8.68
process	54	7.75
hurricane	48	8.40
factor	48	8.16
water	48	7.40
activity	42	7.94
level	40	7.86
event	38	7.86
sea	38	7.81
action	34	7.93
drift	32	8.11

Figure 1: Sample of KP-based word sketches generated by *erosion*.

The automatic KP-based word sketches allowed the extraction of the internal semantic relations in 6 CNs out of the 57 hyponyms of *erosion*, namely the entities or processes being caused by erosion. As for the hyponyms of *sand*, the semantic relations in 6 out of the 56 CNs were elicited by means of these KP-based word sketches, namely the location of sand (see Figure 2), its origins, its composition, and its functions.



"sand" is located at...		2.45
beach	88	10.21
sand	32	9.39
inlet	24	9.21
stream	20	8.76
environment	20	8.65
downdrift	12	8.40
dune	12	8.23
shore	14	8.15

Figure 2: Sample of KP-based word sketches generated by *sand*.

These low figures can be explained by the fact that KP-based grammars for relations like *has_patient*, which are prevalent in the CNs in our study, have not yet been implemented in the sketch grammars. Furthermore, these word sketches do not include adjectives, which appear in 8 of the hyponyms of *erosion* and 26 of the hyponyms of *sand*. In addition, word sketches are a mere statistic summary of term combinations, i.e. they do not show all the terms actually annotated as a KP-based word sketch in the corpus.

For this reason, subsequent ws CQL queries were performed by combining the annotated word sketches with the specific components of each CN. This allowed us to find further knowledge-rich contexts, i.e. "a context indicating at least one item of domain knowledge that could be useful for conceptual analysis" (Meyer, 2001). Table 3 shows an example of a query that targets the sentences annotated as word sketches between *erosion* and *wind*, where ws means word sketch, "erosion-n" and "wind-n" are the terms that have been annotated as part of a word sketch in the corpus; and "\"%w\".*" means any relation defined in the KP-based sketch grammars. As can be observed, the semantic relation in this CN was found to be *caused_by*.

[ws("erosion-n","\"%w\" .*","wind-n ")]
Moreover, the **wind** *causes* more **erosion** when there are no plants and their roots to hold the soil in place.
It is not generally possible to do anything about the *causes* of the **erosion**, namely high tidal levels, **winds**, rain and wave action.
The land was devastated not only by acid deposition but also by the accumulation of toxic metals in the soil, the clearcutting of forested areas for fuel, and soil **erosion** *caused by* **wind**, water, and frost heave.

Table 3: Query for KPs between *erosion* and *wind*.

The same queries were performed to clarify the semantic relations in the CNs of *sand*. Table 4 illustrates a search for KPs between *sand* and *carbonate*, *oil* or *quartz*, which belong to the same category (MATERIAL). The semantic relation was found to be *made_of* since *sand* is usually composed of these materials.

[ws("sand-n","\"%w\" .*","carbonate-n|oil-n|quartz-n")]
The erosion of granitic mountains and the subsequent transport of the erosion products to the coastline by rivers have led to a very significant fraction (around 70 percent) of the beach **sand** being *composed of* **quartz**.
Everyone knew that, potentially, these **sands** *contained* almost unlimited amounts of **oil**, but they remained untouched until the price of oil rose above fifty dollars a barrel.
Most beach **sand** *contains* calcium **carbonate** (CaCO3) fragments from seashells.

Table 4: Query for KPs between *sand* and *carbonate*, *oil* or *quartz*.

The use of ws CQL queries allowed the extraction of the semantic relations in 14 more CNs of *erosion*, i.e. the relations in 20 CNs of *erosion* were accessed by means of KPs, either in the form of word sketches or ws CQL queries. As for the CNs of *sand*, the semantic relations in

6 more CNs were clarified by means of ws CQL queries. Therefore, the relations in 15 CNs of *sand* were elicited by means of KPs (word sketches or ws CQL queries).

CQL queries are more flexible and allowed the elicitation of the relations in CNs composed of adjectives by manually querying their nominal forms (e.g. *structural erosion* > *structure* + *erosion*). However, this is not always possible, because all adjectives do not have a nominal equivalence (e.g. *fine sand*). The relations extracted by means of CQL queries were *caused_by* and *has_patient*, for the hyponyms of *erosion*. As previously stated, although the *has_patient* relation is not yet implemented in the grammars, KRCs conveying this relation were elicited. This was due to the appearance of verbs such as *cause*, which do not only evoke the causative relation, but can also be part of other relations, as evidenced in the following concordance line: (…) *waves may attack the dune during storms, causing erosion and recession of the dune line*. As for the hyponyms of *sand*, the semantic relations elicited were *located_at*, *attribute_origin*, *made_of*, and *has_function*, which shows that the KP-based sketch grammars are so far more suited to the relations evoked by entities (e.g. location, composition, origin, etc.).

However, given that the KP-based sketch grammars need to be further developed in order to include more KPs and different relations (León-Araúz et al., 2016), these searches were complemented with the query for free paraphrases, i.e. specific CQL queries for co-occurrences of the CN constituents in a ±10 span, which allowed to obtain a general idea of the link between the constituents of the 57 CNs of *erosion* and the 56 CNs of *sand*. Even though the query for free paraphrases was entirely manual, offered more noise, and thus was more time-consuming than the previous steps, paraphrases were very useful for the elicitation of the semantic content concealed in CNs as a result of noun packing. Regarding the hyponyms of *erosion*, the semantic relations extracted by means of free paraphrases were *has_patient*, *has_result*, and *caused_by*, as shown in Table 5, which illustrates a query to extract words between *erosion* and *sea*, *river*, *stream* or *glacier*, and vice versa, in a span of 10 tokens.

(meet [lemma= "erosion"] [lemma="sea|river|stream|glacier"] -10 10) within <s/>
Glaciers *are powerful agents of* **erosion**, and are thought to have removed hundreds of feet (meters) from the continental surfaces during the last ice ages.
Guettard (1715), famous for his geological maps, believed the **sea** to *be the major agent in* land **erosion**, and that cliff coasts were the remnants of former extensive hill systems.
Sediment **erosion**, transport and deposition *by* **river**
Rivers *are the major agents of* water **erosion**.
Water, in the form of **streams** and **rivers**, changes mountains *by* **erosion**.
Streams and **rivers** *are significant agents of* **erosion**

Table 5: CQL query for paraphrases of *sea*, *river*, *stream* or *glacier* combined with *erosion*.

Alternatively, the relations elicited in the CNs of *sand* were *attribute_size*, *attribute_moisture*, *attribute_origin*, *located_at*, *patient_of*, *has_part*, and *has_function*, as shown in Table 6, which illustrates a query to extract words between *sand* and *nourishment*, *fill*, *filter* or *filtration*, and vice versa, in a span of 10 tokens.

(meet [lemma= "sand"] [lemma="nourishment|fill|filter|filtration"] -10 10) within <s/>
Characterize offshore sand sources to precisely identify the locations where suitable volumes of beach compatible **sand** exist so they may be *utilized for* beach **nourishment** purposes.
New sources of beach-quality **sand** need to be *readied for* beach **nourishment** following severe storm events and for long-term protection from rising sea level.
Adding sufficient **sand** *to* just **fill** the active profile of width at the rate of sea level rise.
It enables permeable **sands** to *function as* biocatalytic **filters**.
Subsequent **filtration** *using* either **sand**, anthracite, or GAC biofilters with EBCT up to 9 min.

Table 6: CQL query for paraphrases of *nourishment*, *fill*, *filter* or *filtration* combined with *sand*.

Thus, KPs and paraphrases turned out to be interdependent since paraphrases complemented the analysis of the semantic relations in CNs and, at the same time, they can be used to improve the grammars underlying the pattern-based word sketches.

Moreover, the word sketches of *erosion* and *sand* as subject or object of the proposition were analyzed in order to extract verb paraphrases (i.e. underlying verbs that further characterize the concealed semantic relation) of all of the CNs that complemented the previous queries. As can be seen in Tables 7 and 8, different verbs were extracted (i.e. *cause to erode*, *breach*, *destroy*, *cause damage*, *cause to degrade*, and *attack* for *dune erosion*; and *deliver*, *supply*, and *discharge* for *river sand*), all of which refine the semantic relation underlying the CN.

The same occurs in Estela (also Portugal), where dredging activities in the Cávado River and morphological changes in the river basin reduce littoral drift and *cause* the local **dunes** *to erode.*
When barrier beach **dunes** *are breached* by storm wave attack, the result may be the cutting of a new inlet.
Heavy storm events can *destroy* **dune** and beach nourishments by transporting the sediment to deeper water.
Nor is it intended to cover the well documented methods for the prevention or repair of *damage* to **dunes** *caused by* wind action, recreation or grazing.
This may *cause* the **dunes** *to degrade* resulting in loss of the protection provided by the natural dunes.
Waves approaching the beach during high water level may *attack* the **dune** during storms, *causing* erosion and recession of the dune line.

Table 7: Verb paraphrases of *dune erosion*.

Many of the major dune areas were originally formed several thousands of years ago from **sand** produced by the action of glaciers and *delivered to* the coast *by*

rivers in the last Ice Age.
Sand *supplied from* the **river** has mainly been transported by eastward longshore sand transport.
Fine **sand**, silt, and clay *transported by* the **river** and deposited on the floor of a sea or lake beyond the main body of a delta.
Large quantities of **sand** and especially silt and clay *are discharged by* the **river** and accumulate along the shore margins.

Table 8: Verb paraphrases of *river sand*.

After combining KPs and free and verb paraphrases, we observed that the noise and ambiguities produced by KPs and paraphrases were counteracted by combining both procedures. This was found to lead to the disambiguation of CNs. Furthermore, although paraphrases are eminently manual, they can be very valuable for the improvement and refinement of KP-based word sketches (León-Araúz et al., 2016), which can lead to further automation of the process.

5. Extrapolation to Other Complex Nominals

To ascertain whether the semantic relations usually encoded between certain conceptual categories are also reproduced in other CNs composed of terms belonging to the same categories, we extrapolated the semantic analysis of the hyponyms of *sand* and *erosion* to other CNs of the same type. For that purpose, a dictionary was created in NooJ to lemmatize and annotate the terms according to a previously defined set of conceptual categories. This set still needs to be refined (as will be shown) and so far entities have been classified into more fine-grained categories than processes. Then we performed queries based on the combination of most prototypical categories found in the CNs of *sand* and *erosion*: 1) LANDFORM + MATERIAL, WATER BODY + MATERIAL, MATERIAL + MATERIAL, PROCESS + MATERIAL, as found in the hyponyms of *sand*; and 2) LANDFORM + PROCESS, WATER BODY + PROCESS, PROCESS + PROCESS, as found in the hyponyms of *erosion*.

As for the first set of conceptual combinations (based on those found in the CNs of *sand*), 100% of CNs formed by the semantic patterns LANDFORM + MATERIAL (Table 9) and WATER BODY + MATERIAL (Table 10) shared the relation *has_location*.

<LANDFORM> ***location_of***	<MATERIAL> ***has_location***
basin beach bluff coast dune reef shore seabed	sediment
beach channel delta dune floodplain plain shore seafloor	deposit
seafloor reef cliff	rock
shore	gravel
delta	silt

Table 9: The semantic relation *has_location* in CNs formed by the semantic pattern LANDFORM + MATERIAL.

<WATER BODY> ***location_of***	<MATERIAL> ***has_location***
bay lagoon lake marsh swamp	deposit
lake pond river lagoon bay	sediment
lake bay	mud
river	gravel

Table 10: The semantic relation *has_location* in CNs formed by the semantic pattern WATER BODY + MATERIAL.

In the CNs formed by the semantic pattern MATERIAL + MATERIAL (Table 11), nearly all of them shared the relation *made_of*. Only one exception was found out of 75 other CNs analyzed.

<MATERIAL> ***makes_up***	<MATERIAL> ***made_of***
cement clay lime mud	mortar
clay granite	rock
clay coal gravel iron loess mud peat salt sludge	deposit
coral peat quartz	sediment

Table 11: The semantic relation *made_of* in CNs formed by the semantic pattern MATERIAL + MATERIAL.

The exception was *rock salt*, which is not a type of salt made of rock but a type of salt that forms with the appearance of a rock.

In the CNs formed by the semantic pattern PROCESS + MATERIAL (Table 12), 70% of them shared the relation *result_of* and 30% the relation of *has_function*. As will be shown, the type of process (indicated in small caps under the concepts) clearly determines the type of relation held. Therefore, if processes are classified in more specific categories, such as natural processes and artificial processes, the semantic pattern MATERIAL + PROCESS can be disambiguated in 100% of the CNs analyzed.

<PROCESS> ***has_result***	**<MATERIAL>** ***result_of***
accretion erosion metamorphism <NATURAL PROCESS>	sediment
avalanche corrosion flood mudsflow tsunami <NATURAL PROCESS>	deposit
aeration digestion waste <NATURAL PROCESS>	sludge
<PROCESS> ***function_of***	**<MATERIAL>** ***has_function***
compacting <ARTIFICIAL PROCESS>	sediment
composting <ARTIFICIAL PROCESS>	sludge
construction <ARTIFICIAL PROCESS>	aggregate
construction <ARTIFICIAL PROCESS>	rock
construction <ARTIFICIAL PROCESS>	mortar
construction <ARTIFICIAL PROCESS>	stone
construction <ARTIFICIAL PROCESS>	steel

Table 12: The semantic relations *result_of* and *has_function* in CNs formed by the semantic pattern PROCESS + MATERIAL.

As for the second set of conceptual combinations (based on those found in the CNs of *erosion*), approximately 90% of the CNs formed by the semantic pattern LANDFORM + PROCESS shared the relation *has_patient*. Once again, this figure was 100% when the PROCESS category was further refined to LOSS PROCESS and ADDITION PROCESS (see Table 13). However, there were some exceptions, such as *volcanic eruption*, which did not activate the relation *has_patient*, since volcano cannot be considered the patient of eruption but its agent.

<LANDFORM> ***patient_of***	**<PROCESS>** ***has_patient***
soil-aquifer	treatment <CHANGE>
alluvium	transport <MOVEMENT>
aquiclude beach	recharge <ADDITION>
beach	accretion <ADDITION>
beach dune shore cliff	recession <LOSS>
beach	replenishment <ADDITION>
beach dune inlet	stabilization <CHANGE>
canyon channel	flood <ADDITION>
channel	dredging <LOSS>
channel inlet	shoaling <CHANGE>
dune	construction <ACTIVITY>
dune	deflation <LOSS>
land	desertification <CHANGE>
land	subsidence <MOVEMENT>

Table 13: The semantic relation *has_patient* in CNs formed by the semantic pattern LANDFORM + PROCESS.

In the same manner, CNs formed by WATER BODY + PROCESS establish the pattern WATER BODY *causes*

PROCESS when the process entails an addition or movement (e.g. *lake flooding, river flooding, river deposition, runoff infiltration*, etc.). In contrast, when the process involves a change, the WATER BODY is *the_patient_of* the PROCESS (e.g. *lake acidification, lake eutrophication, river pollution*, etc.) (Table 14).

<WATER BODY> ***causes***	**<PROCESS>** ***caused_by***
lake	flooding <ADDITION>
river	flooding <ADDITION>
river	deposition <ADDITION>
river	transport <MOVEMENT>
runoff	infiltration <MOVEMENT>
<WATER BODY> ***patient_of***	**<PROCESS>** ***has_patient***
lake	acidification <CHANGE>
lake	eutrophication <CHANGE>
river	pollution <CHANGE>

Table 14: The semantic relations *caused_by* and *has_patient* in CNs formed by the semantic pattern WATER BODY + PROCESS.

Finally, in most cases the PROCESS + PROCESS pattern was found to encode the relation *causes* in weather- or water-related processes (e.g. *cyclone storm, hurricane storm, hurricane flooding, storm flooding, seepage erosion*, etc.). Nevertheless, exceptions were also present, mainly when there was human intervention in the process. Then, a PROCESS is usually *represented_by* another PROCESS (e.g. *flood simulation* or *erosion simulation*), or a PROCESS is the *patient_of* another PROCESS (e.g. *flood control*) (Table 15).

<PROCESS> ***causes***	**<PROCESS>** ***caused_by***
cyclone <WEATHER PHENOMENON>	storm <WEATHER PHENOMENON>
hurricane <WEATHER PHENOMENON>	storm <WEATHER PHENOMENON>
hurricane <WEATHER PHENOMENON>	flooding <WATER PHENOMENON>
storm <WEATHER PHENOMENON>	flooding <WATER PHENOMENON>
seepage <WATER PHENOMENON>	erosion <LOSS>
<PROCESS> ***represented_by***	**<PROCESS>** ***represents***
flood <WATER MOVEMENT>	simulation <HUMAN INTERVENTION>
erosion <LOSS>	simulation <HUMAN INTERVENTION>
<PROCESS> ***patient_of***	**<PROCESS>** ***has_patient***
flood <WATER MOVEMENT>	control <HUMAN INTERVENTION>

Table 15: The semantic relations *caused_by*, *represents*, and *has_patient* in CNs formed by the semantic pattern PROCESS + PROCESS.

The different semantic patterns of complex term formation highlight the fact that a fine-grained set of conceptual categories is essential in the inference of the semantic relations in CNs. Indeed, all examples in this section call for the recategorization of processes in deeper semantic levels.

6. Conclusions

CNs usually have similar external forms, although they encode different semantic relations. Our goal was to access the semantic relation linking the constituents of the CNs with a view to developing a procedure to infer the semantic relations in other similar CNs. For that purpose, we studied a set of 57 CN hyponyms of *erosion*, which represented environmental processes, and 56 CN hyponyms of *sand*, which designated environmental entities. After classifying the CNs based on the conceptual categories of the modifiers (e.g. LANDFORM, WATER BODY, etc.), we used KPs and paraphrases to semantically analyze the CNs. This analysis based on conceptual categories and semantic relations was extrapolated to other CNs formed by the same categories. It was observed that with a few exceptions, the semantic relations codified in the hyponyms of *erosion* and *sand* were parallel to those of the CNs formed by the same conceptual categories. The semantic analysis of CNs by means of KPs, paraphrases and conceptual categories was found to be a valuable starting point towards the inference of the concealed semantic relation in CNs.

This research can be applied to the inference of semantic relations in different domains and languages, as well as to the translation of CNs since there is a need to render terms into languages other than English. In future work, CNs formed by more than two constituents will be studied, since these longer structures give an insight into the combinatorial potential of semantic categories and their concealed semantics.

7. Acknowledgements

This research was carried out as part of project FF2014-52740-P, Cognitive and Neurological Bases for Terminology-enhanced Translation (CONTENT), funded by the Spanish Ministry of Economy and Competitiveness. Funding was also provided by an FPU grant given by the Spanish Ministry of Education to the first author.

8. Bibliographical References

Bauer, L. (2008). Les composés exocentriques de l'anglais. In D. Amiot (Ed.), *La composition dans une perspective typologique*. Arras: Artois Presses Université, pp. 35–47.

Butnariu, C., and Veale, T. (2008). A concept-centered approach to noun-compound interpretation. In Proceedings of the 22nd International Conference on Computational Linguistics. COLING '08, pages 81–88.

Cabezas-García, M. and Faber, P. (2017). Exploring the Semantics of Multi-word Terms by Means of Paraphrases. In M. A. Candel-Mora & C. Vargas Sierra (Eds.), *Temas actuales en terminología y estudios sobre el léxico*. Granada: Comares, pp. 193-217.

Drouin, P. (2003). Term extraction using non-technical corpora as a point of leverage, *Terminology*, 9(1): 99-117.

Kilgarriff, A., Rychly, P., Smrz, P., and Tugwell, D. (2004). The Sketch Engine. In G. Williams & S. Vessier (Eds.), *Proceedings of the Eleventh EURALEX International Congress*. Lorient: EURALEX, pp. 105–116.

León-Araúz, P., San Martín, A., and Faber, P. (2016). Pattern-based Word Sketches for the Extraction of Semantic Relations. In *Proceedings of the 5th International Workshop on Computational Terminology* (Computerm2016), pages 73-82. Osaka, Japan: COLING 2016.

León-Araúz, P. and Faber, P. (2010). Natural and contextual constraints for domain-specific relations. In V. Barbu Mititelu & V. Pekar (Eds.), *The Workshop Semantic Relations, Theory and Applications*, pp. 12-17.

Levi, J. (1978). The Syntax and Semantics of Complex Nominals. Academic Press, New York.

Maguire, P., Wisniewski, E. J., and Storms, G. (2010). A corpus study of semantic patterns in compounding. *Corpus Linguistics and Linguistic Theory*, 6(1): 49–73.

Marshman, E. (2006). Lexical Knowledge Patterns for Semi-automatic Extraction of Cause-effect and Association Relations from Medical Texts: A Comparative Study of English and French. PhD Thesis, Université de Montréal, Montréal.

Meyer, I. (2001). Extracting Knowledge-Rich Contexts for Terminography: A Conceptual and Methodological Framework. In D. Bourigault, C. Jacquemin, & M.-C. L'Homme (Eds.), *Recent Advances in Computational Terminology*. Amsterdam: John Benjamins, pp. 279-302.

Nakov, P. and Hearst, M. (2006). Using Verbs to Characterize Noun-Noun Relations. *Artificial Intelligence Methodology Systems and Applications*, 4183: 233-244.

Nakov, P. (2013). On the interpretation of noun compounds: Syntax, semantics, and entailment. *Natural Language Engineering*, 19(03): 291–330.

Rosario, B., Berkeley, U. C., Hearst, M. A., and Fillmore, C. (2002). The Descent of Hierarchy, and Selection in Relational Semantics. *ACL*: 247–254.

Silberztein M. (2003). NooJ Manual. Available at: www.nooj4nlp.net.

Tesnière, L. (1976). Eléments de syntaxe structurale. Klincksieck, Paris.

9. Language Resource References

LexiCon Research Group (2017). EcoLexicon.

Generation of a Spanish Artificial Collocation Error Corpus

Sara Rodríguez-Fernández[1], Roberto Carlini[1], Leo Wanner[1,2]
[1]NLP Group, Department of Information and Communication Technologies, Pompeu Fabra University
C/ Roc Boronat, 138, 08018 Barcelona (Spain)
[2]Catalan Institute for Research and Advanced Studies (ICREA)
sara.rodriguez.fernandez|roberto.carlini|leo.wanner@upf.edu

Abstract

Collocations such as *heavy rain* or *make* [*a*] *decision* are combinations of two elements where one (the *base*) is freely chosen, while the choice of the other (*collocate*) is restricted by the base. Research has consistently shown that collocations present difficulties even to the most advanced language learners, so that computational tools aimed at supporting them in the process of language learning can be of great value. However, in contrast to grammatical error detection and correction, collocation error marking and correction has not yet received the attention it deserves. This is unsurprising, considering the lack of existing collocation resources, in particular those that capture the different types of collocation errors, and the high cost of a manual creation of such resoures. In this paper, we present an algorithm for the automatic generation of an artificial collocation error corpus of American English learners of Spanish that includes 17 different types of collocation errors and that can be used for automatic detection and classification of collocation errors in the writings of Spanish language learners.

Keywords: artificial corpus, collocations, collocation errors, second language learning, computer assisted language learning

1. Introduction

Collocations, i.e., idiosyncratic word co-occurrences such as *ask* [*a*] *question*, *commit* [*a*] *murder*, *surmount* [*an*] *obstacle*, *faint suspicion*, *high expectation*, etc. are known to be one of the great challenges for language learners;[1] see, among others, Granger (1998), Lewis and Conzett (2000), Nesselhauf (2005) and Lesniewska (2006). According to Wible et al. (2003), "miscollocations" are the most frequent errors in the writings of students. Orol and Alonso Ramos (2013)'s study shows that the "collocation density" in learner corpora is nearly the same as in native corpora, while the collocation error rate in learner corpora is nearly 30% higher than in native corpora. Despite these palpable figures, collocation error identification and correction has not yet received in *Computer Assisted Language Learning* (CALL) the attention it deserves. Current collocation checkers focus mainly on collocation validation or identification of miscollocations (usually using mutual information- or distribution-based metrics) in the writings of learners and a display of lists of possible corrections, ordered in terms of the strength of their "collocationality" or similarity to the original miscollocation; see, e.g., (Chang et al., 2008; Liu et al., 2009; Wu et al., 2010; Ferraro et al., 2014). However, this is by far not sufficient. Ideally, learners should be given the same kind of feedback as given by language instructors when they mark students' essays: they use, as a rule, error type-specific symbols or acronyms during marking (e.g., 'SV' for "subject verb agreement", 'WO' for "wrong word order", 'WW' for "wrong word", etc.); see, e.g., (Nott, 2008). In other words, they classify the students' mistakes.

[1]In a collocation, one of the elements (the base) keeps the meaning it has in isolation, while the meaning of the other (the collocate) depends on the base. For instance, in *surmount* [*an*] *obstacle*, *obstacle* keeps its meaning, while the interpretation of the meaning of *surmount* depends on *obstacle*.

In order to be able to offer such advanced collocation checkers, sufficiently large collocation resources, and, in particular, learner corpora annotated with collocation error information, which could be used for training machine learning techniques, are needed. Unfortunately, in second language learning, corpora are usually too small. To remedy this bottleneck, artificial corpora have often been compiled in the context of automatic grammar error detection and correction; cf., e.g., Foster and Andersen (2009), Rozovskaya and Roth (2010), or Yuan and Felice (2013), among others. In our work, we adopt the same idea for automatic collocation error detection and correction. In what follows, we present an algorithm for the conversion of the Spanish GigaWord corpus into a collocation error corpus of American English learners of Spanish. As the blueprint of the error type occurrence and distribution, we use the Spanish learner corpus CEDEL2 (Lozano, 2009), which was annotated according to Alonso Ramos et al. (2010)'s three-dimensional fine-grained collocation error typology (see Section 2.). Section 3. presents the algorithm for the creation of the artificial corpus, and Section 4. provides a description of its characteristics. Finally, Section 5. concludes the paper.

2. Collocation Error Typology

Alonso Ramos et al. (2010) present a multidimensional collocation error typology, designed after carrying out an analysis of a fragment of a Spanish learner corpus, the *Corpus Escrito del Español L2* (CEDEL2) (Lozano, 2009). The first dimension, or *Location* dimension, describes where the error is produced, i.e., which element of the collocation is affected, namely the base or the collocate, or whether the error affects the collocation as a whole. The second, *Descriptive*, dimension accounts for the kind of error that has been produced (register, lexical or grammatical). Register errors capture context-inappropriate use of *per se* correct collocations. Lexical errors capture a mistake with re-

spect to one of the collocation elements (either wrong word or creation of a non-existing word) or the collocation as a whole (creation of an artificial single word instead of a collocation, creation of an artificial collocation, or use of a collocation with a different sense than intended). Grammatical errors concern the grammar of collocations (missing or superfluous determiner, wrong preposition, wrong subcategorization, etc.). Finally, the third dimension, the *Explanatory* dimension, models the cause of the errors, that is, whether they are caused by interlingual or intralingual reasons.

In previous works on error detection and correction, it has been common to divide errors according to the type of operation that needs to be carried out to make a particular error, that is a *substitution* operation, a *deletion* operation, or an *insertion* operation. Taking into account these operations in the context of learner collocation resources can be useful for developing more accurate strategies for error correction, and for providing better feedback to the learners. Given that the typology by Alonso Ramos et al. (2010) does not consider these operations, we have opted to include them and subdivide when possible the collocation error types into these three extra categories. We arrive, thus, at a fine-grained typology that takes into account, for each type of error: (1) the location of the error, i.e., base, collocate or collocation as a whole, (2) the error type that is produced, i.e., creation, government, order errors, etc., and (3) the type of operation that results in the particular error, i.e., substitution, deletion and insertion.[2] As a consequence, we obtain types of errors such as *Government Base Substitution*, where the preposition of the base is incorrectly chosen, *Pronoun Insertion*, where a reflexive pronoun is incorrectly inserted into the collocation, or *Collocate Creation*, where the collocate is an invented word, etc.

An analysis of the CEDEL2 corpus, annotated with collocation errors, reveals that some of the error types in the typology by Alonso Ramos et al. (2010) tend to occur very seldom. For this reason, we have opted to disregard them in our current work, arriving at the following classes of lexical and grammatical collocation errors (17 in total):

Lexical errors

• **SubB**. Erroneous choice of the base, as in **tener confidencia*, lit. 'have confidence [secret]'; corr.: *tener confianza*, lit. 'have confidence [trust]'.

• **SubC**. Erroneous choice of the collocate, as in **hacer una decisión*, lit. 'make a decision'; corr.: *tomar una decisión*, lit. 'take a decision'.

• **CrB**. Erroneous choice of a non-existing base, as in **hacer un llamo*, lit. 'make a *llamo* [non-existing word meaning *call*]'; corr.: *hacer una llamada*, lit. 'make a call'.

• **CrC**. Erroneous choice of a non-existing collocate, as in **serie televisual*, lit. '[non-existing word meaning *TV*] series'; corr.: *serie televisiva*, lit. 'TV series'.

Grammatical errors

• **DetD**. Erroneous omission of a determiner of the nominal base, as in **ir a _ escuela*, lit. 'go to school'; corr.: *ir a la escuela*, lit. 'go to the school'.

• **DetI**. Erroneous presence of a determiner of the nominal base, as in **hablar el inglés*, lit. 'speak the English'; corr.: *hablar inglés*, lit. 'speak English'.

• **GoBD**. Erroneous omission of a preposition governed by the base, as in **tener la oportunidad _ hacer algo*, lit. 'have the opportunity to do something'; corr.: *tener la oportunidad de hacer algo*, lit. 'tener la oportunidad of do something'.

• **GoBS**. Erroneous choice of the preposition governed by the base, as in **tener obligación a*, lit. 'have obligation to'; corr.: *tener obligación de*, lit. 'have the obligation of'.

• **GoCD**. Erroneous omission of a preposition governed by the collocate, as in **asistir _ una universidad*, lit. 'assist a university'; corr.: *assistir a una universidad*, lit. 'assist to a university'.

• **GoCI**. Erroneous presence of the preposition governed by the collocate, as in **perder a clientes*, lit. 'lose to clients'; corr.: *perder clientes*, lit. 'lose clients'.

• **GoCS**. Erroneous choice of the preposition governed by the collocate, as in **ir por tren*, lit. 'go by train'; corr.: *ir en tren*, lit. 'go in train'.

• **PrD**. Erroneous use of a non-reflexive form of the verbal collocate (omission of the reflexive pronoun), as in **el hielo _ descongela*, lit. 'the ice melts'; corr.: *el hielo se descongela*, lit. 'el hielo melts itself'.

• **PrI**. Erroneous use of the reflexive form of the verbal collocate (insertion of the reflexive pronoun), as in **odio que uno se siente*, lit. 'hatred that one feels themselves'; corr.: *odio que uno siente*, lit. 'hatred that one feels'.

• **NumB**. Erroneous number of the base, as in **dar bienvenidas*, lit. 'give welcomes'; corr.: *dar la bienvenida*, lit. 'give the welcome'.

• **NumD**. Erroneous number of the base determiner, as in **buena notas*, lit. 'good[sing] marks'; corr.: *buenas notas*, lit. 'good[pl] marks'.

• **Gen**. Erroneous gender, as in **aumentar las precios*, lit. 'raise the[fem] prices'; corr.: *aumentar los precios*, 'lit. 'raise the[masc] prices'.

• **Ord**. Erroneous word order, as in **educación buena*, lit. 'education good'; corr.: *buena educación*, lit. 'good education'.

3. Generation of an Artificial Collocation Error Corpus

This section focuses on the methodology for the generation of the artificial corpus. In our work, errors are generated and introduced probabilistically, based on the collocation error distribution of the CEDEL2 corpus. In what follows, we first present a statistical analysis of the CEDEL2 corpus and then provide a detailed description of the error generation algorithm. Afterwards, the resources that are used for the creation of the artificial corpus are outlined.

3.1. Analysis of the learner corpus CEDEL2

In order to obtain relevant information about the error distribution in the learner corpus, we start from CEDEL2, carrying out a statistical analysis of the errors present in this corpus. The error distribution is shown in Table 1.[3] The

[2] In our work, we also consider the *Explanatory* dimension, used as source of information for the automatic generation of the errors

[3] Currently, we only consider error types whose raw frequencies are equal or above 5

Error type	Frequency	%
SuC	470	32.41
Gen	116	8.00
GoCD	98	6.76
SuB	96	6.62
DetD	87	6.00
DetI	78	5.38
CrB	72	4.96
GoBS	48	3.31
GoCI	48	3.31
GoCS	45	3.10
Ord	38	2.62
NumB	33	2.27
GoBD	32	2.21
PrI	27	1.86
CrC	25	1.72
PrD	23	1.59
NumD	10	0.69

Table 1: Frequency of collocation errors in CEDEL2

Error type	Frequency	%
GoCI + SuC	11	0.76
PrD + SuC	10	0.69
GoCD + SuC	9	0.62
DetI + NumB	6	0.41
DetI + GoBS	5	0.34
PrI + SuC	5	0.34
Ord + SuC	5	0.34

Table 2: Multiple error types

second column of the table refers to the number of times that each type of error occurs in the corpus, and the third column shows the percentage of the corresponding error type with respect to the total number of collocation errors found in the corpus.

We observed that a collocation can be often affected by several errors at the same time, for instance, containing an error in the base and another in the collocate, such as in **jugar tenis*, 'to play tennis', corr. *jugar al tenis*, lit. 'to play to the tennis', where there is an omission of the base determiner *el* 'the', and an omission of the collocate preposition *a* 'to'.[4] In the current state of our work, these cases are treated as separate occurrences of the errors, and the decision whether to insert two errors in a collocation is taken randomly by the system.

Furthermore, we found that a base or a collocate can be affected by several errors.[5] This occurs less often, but is nonetheless a phenomenon that needs to be reflected in the artificial corpus. Table 2 shows all combinations whose raw frequencies are equal or above 5, and presents their frequencies and percentages with respect to the total number of collocation errors in CEDEL2.

In order to generate errors that simulate "real" errors produced by learners, it is not sufficient to copy the error distribution observed in a learner corpus; an analysis of the most

[4]In Spanish, when the preposition *a* 'to' is followed by the determiner *el* 'the', the contracted form *al* is used

[5]We include here cases where an error that affects the collocation as a whole, i.e., **Ord**, and an error affecting either the base or the collocate is produced in the same collocation

Correct	Incorrect	#	%
None	*a* 'at'	16	33.33
	con 'with'	14	29.17
	de 'of'	13	27.08
	en 'in'	2	4.17
	por 'by', 'for'	2	4.17
	para 'to', 'for'	1	2.08

Table 3: Frequently confused prepositions (GoCI)

Correct	Incorrect	#	%
a 'at'	None	83	84.69
en 'in'		8	8.16
de 'of'		3	3.06
con 'with'		2	2.04
por 'by', 'for'		1	1.02
sobre 'over'		1	1.02

Table 4: Frequently confused prepositions (GoCD)

frequently confused words is also needed for the cases in which errors are produced through word replacements. In our case, we perform this analysis only for government errors, since, on the one hand, in lexical errors the number of possible options is infinite and thus the usefulness for our work very limited and, on the other hand, the only type of grammatical error where the incorrect choice of a word is considered an error are government errors.[6] The statistics concerning the wrong use of prepositions are presented in Tables 3, 4, 5, 6 and 7.

3.2. Algorithm for Creation of the Error Corpus

The algorithm for the generation of the collocation error corpus passes through three main stages: (1) collocation extraction, (2) collocation classification, and (3) error generation and injection. Firstly, all the N–V, N–Adj and V–Adj dependencies that occur in the corpus where the errors are to be inserted, are retrieved and classified, according to their POS pattern, into three groups: N–V, N–Adj and V–Adj. A statistical check is performed to reject non-collocations: we choose the asymmetrical normalized *Pointwise Mutual Information* (PMI) by Carlini et al. (2014) and consider as collocations only those dependencies whose PMI is higher than 0. Collocations are stored with their prepositions, determiners and pronouns, along with relevant information that will be used at later stages, such as their position in the sentence, lemmas, POS-tags, morphological information, and their sentential context.

Secondly, collocations are classified according to the types of errors that they can contain. For instance, N–Adj collo-

[6]Recall that the incorrect choice of determiner and pronoun are not seen as collocation errors

Correct	Incorrect	#	%
de 'of'	None	27	84.37
en 'in'		3	9.37
para 'to', 'for'		1	3.12
sobre 'over'		1	3.12

Table 5: Frequently confused prepositions (GoBD)

Correct	Incorrect	#	%
en 'in'	*por* 'by', 'for'	16	69.56
	a 'at'	4	17.39
	de 'of'	2	8.69
	con 'with'	1	4.35
de 'of'	*en* 'in'	4	57.14
	a 'at'	3	42.86
para 'to', 'for'	*en* 'in'	1	100
a 'at'	*en* 'in'	7	77.78
	de 'of'	1	11.11
	por 'by', 'for'	1	11.11
por 'by', 'for'	*en* 'in'	2	66.66
	a 'at'	1	33.33
contra 'against'	*con* 'with'	1	100

Table 6: Frequently confused prepositions (GoCS)

Correct	Incorrect	#	%
en 'in'	*de* 'of'	2	50
	sobre 'over'	1	25
	**in* 'in'	1	25
de 'of'	*para* 'to', 'for'	9	42.86
	a 'at'	8	38.09
	en 'in'	2	9.52
	que 'that'	1	4.76
	como 'as'	1	4.76
para 'to', 'for'	*por* 'by', 'for'	2	50
	a 'at'	1	25
	de 'of'	1	25
a 'at'	*de* 'of'	6	85.71
	en 'in'	1	14.28
por 'by', 'for'	*para* 'to', 'for'	6	54.54
	a 'at'	3	27.27
	de 'of'	2	18.18
sobre 'over'	*de* 'of'	2	100

Table 7: Frequently confused prepositions (GoBS)

cations cannot be affected by pronoun errors, V–Adv collocations cannot contain gender errors, collocations that do not contain a determiner cannot be affected by a determiner omission error, etc. A list of candidates is thus created for each type of error.

Finally, errors are generated and inserted according to the error distribution presented in the CEDEL2 corpus. In each iteration, an error type is probabilistically chosen by the system; then a candidate from the list is taken, and an *error generator* produces an error, which is inserted into the sentence; otherwise, the candidate is ignored. In order to preserve the error distribution observed in the CEDEL2 corpus, the creation of the corpus ends when the number of candidates for any of the errors is equal to zero. The set of *Error Generators* that are used are presented below.

3.2.1. Error Generators

A total of six Generators is used to produce the 17 types of collocation errors that we target. 5 are developed for grammatical errors, and one generates all types of lexical errors.

1. Order Error Generator (OEG)

The OEG takes as input N–Adj and V–Adv collocations and swaps the order of the base and the collocate, generating order errors (**Ord**). In order to avoid the creation of uncontrolled grammatical errors, only collocations whose components appear in contiguous order are considered.

2. Gender Error Generator (GEG)

The GEG's role is to insert gender errors (**Gen**) into V–N and N–Adj collocations. In both types of collocations, gender errors are produced in the determiner of the base. In N–Adj collocations, the adjectival collocate is considered as a determiner itself, such that gender errors can be produced either in the base determiner or in the collocate. In the cases where a gender error can be inserted in both places, the GEG randomly chooses where to insert the error, i.e., in the determiner or in the adjective.

The GEG is made up of two main functions, one that changes the gender of the determiner, and one that changes the gender of the adjectival collocate. For determiners, the system first checks whether the input determiner is included in a list of irregular determiners, where both masculine and feminine forms are given. If so, the original determiner is replaced by its alternative form. Otherwise, common gender inflection rules are applied according to the determiner's last letters. For adjectives, a suffix map is used, where masculine suffixes are mapped to feminine ones, and vice versa. The system simply checks whether the adjective's last letters are included in the map, and replaces the original ending with the new one.

As a final step, the existence of the created form is guaranteed by checking its frequency in the reference corpus.

3. Number Error Generator (NEG)

The NEG inserts number errors into V–N and N–Adj collocations. As in the case of 'Gender' errors, 'Number' errors can be produced in the determiner or in the adjectival collocate. In contrast to 'Gender' errors, however, 'Number' errors can also affect the nominal base of the collocation. The NEG inserts, then, two types of errors: **NuBD** for errors produced in the determiner and adjectival collocate, and **NuBB** for errors produced in the base. In cases where the error can be inserted in more than one place, the NEG randomly chooses where to insert the error.

The NEG works as the GEG, i.e., two main functions are designed, one that deals with determiners and one that deals with adjectives and nouns. A list of irregular determiners together with number inflection rules is used for the former, while a suffix map is used for the latter.

4. Substitution Error Generator (SEG)

The SEG inserts replacement and deletion errors into N–V and N–Adj collocations. In the case of replacement errors, we only consider government replacement errors (**GoBS** and **GoCS**). The SEG takes as input collocations in which the target component (the base or the collocate) has a government preposition, and replaces it with another preposition, according to the statistics observed in the learner corpus.

Changing a preposition often results in an error, but in some occasions it can lead to a correct collocation that involves a change of meaning. In order to avoid the introduction of a false error, we developed an PMI-based association metric that calculates the association strength between the collo-

cate, the target preposition and the context of the collocation (a window of 2). Only when the contextual PMI of the original collocation is higher than the contextual PMI of the new collocation, is the error inserted.

Deletion errors can be produced in either prepositions, determiners or pronouns, giving rise to **GoBD**, **GoCD**, **DetD** and **PrD** errors. The mechanism of the SEG for deletion errors is the same as for replacement errors, the only difference being that while in replacement errors the replacement is a valid element, in deletion errors the replacement is void. Contextual PMI is also computed in deletion errors to check that the generated error is a true error.

5. Insertion Error Generator (IEG)

The IEG behaves as the SEG, with the difference that, in this case, none of the elements is changed nor removed, but rather a new element is inserted instead. The IEG generates government, determiner and pronoun insertion errors (**GoCI**,[7] **DetI** and **PrI**) in N–V and N–Adj collocations. As with the SEG, the IEG also uses contextual PMI scores to avoid the insertion of false errors.

The manner in which the element to be inserted is chosen depends on whether the target element is a preposition, a determiner or a pronoun. Prepositions are probabilistically chosen, according to the error statistics observed in the learner corpus, and inserted after the collocate. For determiners, the IEG inserts an indefinite article before the noun. Since neither the definite/indefinite confusion, nor the confusion of any determiner is considered as a collocation error in Alonso Ramos et al. (2010)'s typology, any determiner could be inserted in any case. For simplicity, we opted to always insert indefinite articles, choosing among the different forms depending on the noun number and gender. Finally, pronouns are inserted in two ways, following the rules of the Spanish grammar. For conjugated verbs, the correct pronoun that corresponds to the verb person and number is inserted before the verb. For infinitive forms, the reflexive pronoun *se* is added to the infinitive.

6. Lexical Error Generator (LEG)

The LEG inserts lexical substitution and creation errors in N–V, N–Adj and V–Adv collocations, in both the base and the collocate. The error types covered by the LEG are, therefore, **SuB**, **SuC**, **CrB** and **CrC**. The LEG finds or creates a replacement base or collocate and changes the original base or collocate by the replacement, an existing word in substitution errors, and a non-existing word in creation errors.[8]

Replacement words can be generated in different ways, i.e., (1) **transfer**, where the target base or collocate is translated into L1, (2) **affix change**, where a suffix (including gender inflection) is applied to the target element, (3) **transfer + affix change**, (4) **synonymy** (only for substitution errors), where the target element is replaced by one of its synonyms, and (5) **literal translation**, (exclusively for substitution collocation errors), where the base is translated into L1, and the verb that most often co-occurs with the base in the L1 is retrieved, translated into Spanish and used to replace the original verb. The choice of the method for generating the replacement is random. When unable to generate an error by means of the chosen option, the system selects another option until a valid replacement is found or until the options are finished.

3.3. Resources

The following resources have been used for the generation of the artificial corpus:

- **Base Corpus**. Spanish GigaWord corpus `https://catalog.ldc.upenn.edu/ldc2011t12`.
- **Learner corpus**. We use a learner corpus in order to obtain relevant information regarding the collocation errors that Spanish L2 learners make in their writings. As mentioned above, we use for this purpose CEDEL2. CEDEL2 is a Spanish L2 learner corpus (Lozano, 2009), which includes essays on different topics written by US learners of Spanish of different levels. Our working corpus is formed by 517 essays of levels ranging from pre-intermediate to advanced.
- **Reference corpus**. We use reference corpora (RC) to check word frequencies and co-occurrences. In particular, the algorithm makes use of two RCs, a Spanish RC and an English RC. The Spanish RC consists of 7 million sentences from newspaper material. For English, we use the British National Corpus (BNC), which contains 100 million words from texts of a variety of genres. In order to obtain syntactic dependency information, both corpora were processed with Bohnet (2010)'s dependency parser.
- **Spanish WordNet**. The algorithm also makes use of the Spanish WordNet, from the Multilingual Central Repository 3.0 (Gonzalez-Agirre et al., 2012) as a source of synonymy information. The NLTK library is used to access its contents.
- **Google Translate**. Google Translate is used as bidirectional translation engine, both to translate from Spanish to English, and from English to Spanish. Access to it is provided by the TextBlob Python library.
- **Morphological inflection tool**. Finally, the algorithm uses the morphological inflection tool by Faruqui et al. (2016). This tool allows for the generation of morphologically inflected forms of a word according to given morphological attributes. In our case, we use it for the generation of lexical errors, to inflect the words that are automatically created by the algorithm as replacement for bases and collocates.

4. The Artificially Generated Corpus

In order to check to what extent our artificial corpus simulates our learner corpus, we carried out an analysis of both of them. For this purpose, we took a sample of 50 sentences from each corpus and paid attention to three main aspects: (1) collocation errors, (2) non-collocation errors, and (3)

[7] In the CEDEL2 corpus the frequency of GoBI errors was rather small, so we opted for disregarding this type of error

[8] As in 'Gender' errors, the existence of the replacement words is checked in the RC.

sentence complexity. This is, on the one hand, because the analysis of the generated errors and their comparison to the "real" learners' errors is crucial for a qualitative evaluation of the resource. On the other hand, a comparison of the non-collocation errors and the sentence complexity between the "real" and the synthetic corpora might shed some extra light regarding the similarity of the two corpora. The analysis is presented below.

4.1. Collocation errors

A look at the generated errors points to some important conclusions, mainly that, even when some of the generated errors resemble indeed learners' errors, in some cases, the algorithm fails to generate errors correctly. Thus, firstly, not all the combinations in which errors are inserted are real collocations. Some are free combinations; cf., e.g., *representantes _ las islas* 'islands' representatives'; orig. *representantes de las islas* and *llenaba el plaza* 'filled the square'; orig. *llenaba la plaza*.

Secondly, the injection of an "error" does not always produce a collocation error but, rather, results in a correct collocation involving a change of meaning. For instance, in *la depresión nerviosa que le causó la muerte a su mujer*, lit. 'the nervous depression that caused the death to his wife'; orig. *la depresión nerviosa que le causó la muerte de su mujer* 'the nervous depression that caused the death of his wife'. In other cases, the injection of the "error" results in a change of determination, such as in *consumir una droga* 'to use a drug' lit. 'to consume a drug'; orig. *consumir _ droga* 'to use drugs' lit. 'to consume drug'.

Finally, the injection of an error may result in the generation of unexpected errors. For example, the substitution of *instrumento* 'instrument' by its synonym *herramienta* 'tool' in *es un herramienta que manejaremos*, lit. 'it is a tool that we will use'; orig. *es un instrumento que manejaremos*, lit. 'it is an instrument that we will use', produces a determiner error, since there is no agreement between the changed base *herramienta* and the determiner.

4.2. Non-collocation errors

This section summarizes our findings regarding the production of errors outside the context of collocations. In particular, we consider orthographical, grammatical, lexical, punctuation and discourse marking errors. Our base corpus (the GigaWord) is assumed to be well written, and thus to be free of any error, apart from those collocation errors that were automatically generated. A closer look at it reveals that it contains indeed only very few spelling and grammatical mistakes. Some spelling errors are present, although their proportion and variety is much smaller than in the CEDEL2 corpus: only an unaccented word and 4 typos have been found. The only type of grammatical error observed in the GigaWord sample are agreement errors. Lexical, punctuation and discourse marker errors have not been observed.

4.3. Sentence complexity

In order to measure the sentence complexity, we select several features that can approximate the level of sentence complexity. These features and the values obtained for the two samples are presented in Table 8. In order to obtain the POS and syntactic features, the samples have been processed with (Bohnet, 2010)'s dependency parser.

Feature	CEDEL2	GigaWord
Total words	1,301	2,021
Average sentence length	26.02	40.42
Sentence noun ratio	5.14	10.10
Sentence adjective ratio	1.54	5.18
Sentence verb ratio	3.62	3.50
Sentence adverb ratio	1.56	0.90
Sentence punctuation ratio	2.24	3.38
Sentence coordination ratio	1.10	1.10
Sentence subordination ratio	0.94	0.50
Sentence relativization ratio	0.72	0.66
Sentence passivization ratio	0.18	0.18
Sentence apposition ratio	0.08	0.66

Table 8: Syntactic complexity features in the GigaWord and CEDEL2 samples

As can be observed in Table 8, the values for some of the features, such as the coordination of passivization ratios are rather similar in both corpora. However, each corpus also shows its own morpho-syntactic profile. For instance, the apposition ratio is 8 times higher in the GigaWord corpus than in the L2 corpus. Nouns and adjectives are also significantly more common in the GigaWord corpus, as is the use of punctuation marks. On the contrary, learners tend to use more adverbs and subordinate clauses. As expected, sentence length is substantially shorter in L2 writings.

5. Conclusions and Future Work

We presented an algorithm for the automatic generation of a collocation error corpus of Spanish. The algorithm is able to insert 17 types of errors in error-free data. Such a resource can prove useful for the development of computational collocation tools designed to provide valuable feedback to language learners regarding the types of errors they make. For our experiments, we use the Spanish GigaWord as base corpus.

Using this algorithm, we generated an artificial collocation error corpus, showing that between the CEDEL2 learner corpus and the artificial corpus there still are some differences, which affect both collocation and non-collocation errors, and sentence complexity (in addition to differences in domain and style). All these differences are likely to imply that an algorithm trained on artificial data may not perform as well on L2 data as it may on the artificial data.

To validate the generated error corpus, we carried out some preliminary experiments on collocation error recognition and classification, using LSTMs, in which we achieved an average precision of 0.95 and an average recall of 0.67. As expected, performance falls when the evaluation is carried out on L2 data: when experiments are run on the CEDEL2 corpus, an average precision of 0.58 and recall of 0.39 is achieved.

References

Alonso Ramos, M., Wanner, L., Vincze, O., del Bosque, G. C., Veiga, N. V., Suárez, E. M., and González, S. P.

(2010). Towards a motivated annotation schema of collocation errors in learner corpora. In *LREC*.

Bohnet, B. (2010). Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd international conference on computational linguistics*, pages 89–97. Association for Computational Linguistics.

Carlini, R., Codina-Filba, J., and Wanner, L. (2014). Improving collocation correction by ranking suggestions using linguistic knowledge. In *Proceedings of the third workshop on NLP for computer-assisted language learning at SLTC 2014, Uppsala University*, number 107. Linköping University Electronic Press.

Chang, Y., Chang, J., Chen, H., and Liou, H. (2008). An Automatic Collocation Writing Assistant for Taiwanese EFL learners. A case of Corpus Based NLP Technology. *Computer Assisted Language Learning*, 21(3):283–299.

Faruqui, M., Tsvetkov, Y., Neubig, G., and Dyer, C. (2016). Morphological inflection generation using character sequence to sequence learning. In *Proceedings of NAACL-HLT*, pages 634–643.

Ferraro, G., Nazar, R., Alonso Ramos, M., and Wanner, L. (2014). Towards advanced collocation error correction in Spanish learner corpora. *Language Resources and Evaluation*, 48(1):45–64.

Foster, J. and Andersen, Ø. E. (2009). Generrate: generating errors for use in grammatical error detection. In *Proceedings of the fourth workshop on innovative use of nlp for building educational applications*, pages 82–90. Association for Computational Linguistics.

Gonzalez-Agirre, A., Laparra, E., and Rigau, G. (2012). Multilingual central repository version 3.0. In *LREC*, pages 2525–2529.

Granger, S. (1998). Prefabricated patterns in advanced EFL writing: Collocations and Formulae. In A. Cowie, editor, *Phraseology: Theory, Analysis and Applications*, pages 145–160. Oxford University Press, Oxford.

Lesniewska, J. (2006). Collocations and second language use. *Studia Lingüística Universitatis lagellonicae Cracoviensis*, 123:95–105.

Lewis, M. and Conzett, J. (2000). *Teaching Collocation. Further Developments in the Lexical Approach*. LTP, London.

Liu, A. L.-E., Wible, D., and Tsao, N.-L. (2009). Automated suggestions for miscollocations. In *Proceedings of the NAACL HLT Workshop on Innovative Use of NLP for Building Educational Applications*, pages 47–50, Boulder, CO.

Lozano, C. (2009). Cedel2: Corpus escrito del español l2.

Nesselhauf, N. (2005). *Collocations in a Learner Corpus*. Benjamins Academic Publishers, Amsterdam.

Nott, D. (2008). Marking studentsâ written work: principles and practice, available from: www.llas.ac.uk/resources/gpg/2956, accessed on 10.07.2016. In *Good Practice Guide*. Subject Centre for Languages, Linguistics and Area Studies, Southampton.

Orol, A. and Alonso Ramos, M. (2013). A Comparative Study of Collocations in a Native Corpus and a Learner Corpus of Spanish. *Procedia–Social and Behavioural Sciences*, 96:563–570.

Rozovskaya, A. and Roth, D. (2010). Training paradigms for correcting errors in grammar and usage. In *Human language technologies: The 2010 annual conference of the north american chapter of the association for computational linguistics*, pages 154–162. Association for Computational Linguistics.

Wible, D., Kuo, C.-H., Tsao, N.-L., Liu, A. L.-E., and Lin, H.-L. (2003). Bootstrapping in a language learning environment. *Journal of Computer Assisted Learning*, 19(4):90–102.

Wu, J.-C., Chang, Y.-C., Mitamura, T., and Chang, J. (2010). Automatic collocation suggestion in academic writing. In *Proceedings of the ACL Conference, Short paper track*, Uppsala.

Yuan, Z. and Felice, M. (2013). Constrained grammatical error correction using statistical machine translation. In *CoNLL Shared Task*, pages 52–61.

Improving a Neural-based Tagger for Multiword Expression Identification

Dušan Variš, Natalia Klyueva
Institute of Formal and Applied Linguistics, Charles University; The Hong Kong Polytechnic University
Malostranské náměstí 25, Prague; 11 Yuk Choi Rd, Hung Hom,
varis@ufal.mff.cuni.cz,
natalia.klyueva@polyu.edu.hk

Abstract

In this paper, we present a set of improvements introduced to MUMULS, a tagger for the automatic detection of verbal multiword expressions. Our tagger participated in the PARSEME shared task and it was the only one based on neural networks. We show that character-level embeddings can improve the performance, mainly by reducing the out-of-vocabulary rate. Furthermore, replacing the softmax layer in the decoder by a conditional random field classifier brings additional improvements. Finally, we compare different context-aware feature representations of input tokens using various encoder architectures. The experiments on Czech show that the combination of character-level embeddings using a convolutional network, self-attentive encoding layer over the word representations and an output conditional random field classifier yields the best empirical results.

Keywords: multiword expressions, machine learning, deep learning, conditional random field

1. Introduction

Multiword Expression (MWE) is a sequence of words for which the meaning of a whole sequence cannot be derived from the meaning of its components straightforwardly. MWEs are viewed by computational linguists as a "pain in the neck of NLP" due to their non-compositionality and irregularity that can cause problems in areas such as machine translation, terminology extraction, etc. Regarding this, MWEs are largely addressed in both the theoretical and applied research. Associative measures (calculating association between distinct words in MWE) are usually used for extracting MWEs (Ramisch, 2015; Kilgarriff et al., 2014). The identification of MWE in the text is thus a challenging task.

Within PARSEME's special MWE-related project[1], researchers from different countries created guidelines on how to define MWEs in the text and annotated corpora in 18 languages. The focus was on verbal MWEs which were categorized into five classes: idioms (ID), light verb constructions (LVC), inherently reflexive verbs (IReflV), verb-particle constructions (VPC), and other (OTH). The process of annotation was language-dependent. For instance, in Hungarian, VPCs are annotated and IReflVs are not, however, in Czech it is the other way around.

These data then served as the training data for systems participating in the shared task on automatic verbal multiword expression (VMWE) identification (Savary et al., 2017). In addition to the MWEs markings, morphosyntactic markings were provided in the corpora as well.

Seven systems based on various approaches and algorithms participated in the task. Two of them were based on conditional random field (CRF) (Maldonado et al., 2017; Boroş et al., 2017), the other was trained using dependency parsing (Simkó et al., 2017) and the winner was a transition-based system exploiting syntactic rules (Al Saied et al., 2017).

For some of the languages, our previous model based on the neural networks (NN) had comparable scores with other multilingual systems, yet it performed best only in one language – Romanian. Later comparison between our system and the winner (transition-based approach) revealed a large gap between the MWE-based scores, which focus on exact matches between the hypothetical and the gold MWEs and the token-based scores, which compare individual tokens from the MWEs. Our approach does not perceive MWEs as a whole, labeling each token individually, although, sometimes it can capture long distant dependencies between the MWE components.

In this paper, we describe our ongoing work on improving our tagger, MUMULS, using the current state-of-the-art sequence-to-sequence techniques applied in other NLP tasks, including different styles of embedding of the input tokens, creating a context-aware feature representation of the input sequence and generating of the target labels.

This paper is structured as follows. We describe the data preprocessing in Section 2. In Section 3., we describe the proposed improvements. In Section 4., we describe the experiments and analyze their results. We conclude our findings in Section 5.

2. Data Preparation

The training data were provided in two files per each language, one in the CoNLL-U format with the morphosyntactic annotations and the other in a specially created parseme TSV format with the respective annotations of VMWEs. From these two files, we extract word forms, lemmas, part-of-speech (POS) tags and target MWE labels and use them to train our models.

The MWE labels have a following format: *"mwe_id":"mwe_label"* (e.g. *1:ID*), where *"mwe_id"* is used to distinguish different MWEs within the same sentence. A single token can also belong to multiple MWEs having multiple labels separated by a semicolon (e.g. *1:ID;2:IReflV*).

Since it would be difficult to train a tagger using this specific label format, we preprocess the labels in a following way: the first token from a MWE receives the MWE label

[1] https://typo.uni-konstanz.de/parseme/

without the *"mwe_id"*[2] and the following tokens receive a special *CONT* label. During the postprocessing, we assign an unique *"mwe_id"* to each label except for *CONT* and replace each of the following *CONT* labels with the same *"mwe_id"*. If the first label encountered in the labeled sequence is *CONT*, we use the most frequent MWE label (based on the training data) to replace it.
Below is an excerpt from the training file for French with an idiomatic expression *mettre un terme* – 'finish' (the fourth and fifth column contain original and preprocessed labels respectively):

```
Il       il       PRON    _      _
met      mettre   VERB    1:ID   ID
un       un       DET     1      CONT
terme    terme    NOUN    1      CONT
à        à        ADP     _      _
sa       son      DET     _      _
carrière carrière NOUN    _      _
```

Clearly, our processing methods cannot handle several phenomena, for example, crossing MWEs or tokens that belong to multiple MWEs. For this reason, we used an "Oracle" tagger that only applied preprocessing and postprocessing on the gold labels in the test data and compared the produced output with the original labels. The results showed that using our processing methods can still produce a tagger with an f-measure score of 0.95 and higher. Therefore, we consider the suggested processing methods sufficient for this task.

3. System Description

Our MWE tagger is a sequence classifier that predicts the target labels using feature representations computed by a deep neural network (DNN). In the last few years, DNNs started achieving the state-of-the-art results in many NLP tasks including machine translation (Sutskever et al., 2014), natural language generation (Wen et al., 2015) and more importantly POS tagging (Pérez-Ortiz and Forcada, 2001) and named entity recognition (Lample et al., 2016). The system we submitted to the VMWE shared task [3] was implemented using the TensorFlow[4] open source library (Abadi et al., 2016). However, our current research-in-progress is implemented in the Neural Monkey[5] (Helcl and Libovický, 2017) framework for sequence modeling, because it enables easier prototyping and replication of the experiments.[6]
Figure 1 shows a general overview of the system architecture. It consists of three separate layers, the *embedding layer*, which assigns an embedding vector to each input token, the *encoder layer*, which transforms each embedding vector to a context-aware vector representation and the *classifier layer*, which assigns an output label to each token based on the context-aware representation. We describe each layer in more detail in the following sections.

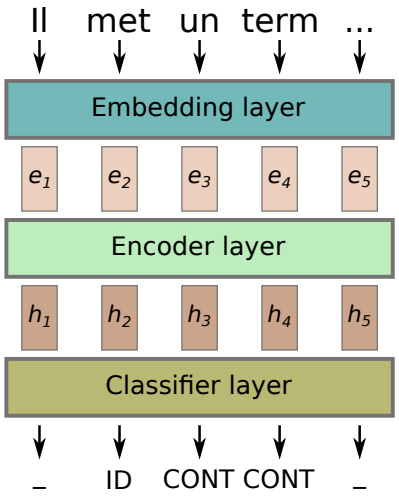


Figure 1: General overview of the MUMULS MWE tagger. e_i represents the embedding of the i-th word, h_i represents its context-aware representation.

3.1. Embedding Layer

The role of the embedding layer is to assign each word w_i in the input sequence $\mathbf{w} = (w_1, .., w_n)$ an embedding vector $e_i \in \mathbb{R}^d$ where *d* is the embedding size, creating a sequence representation $\mathbf{e} = (e_1, .., e_n)$. We can accomplish this in two ways: either by using an embedding lookup table or by computing the embedding using the embeddings of its characters. We call the former method *word-level embedding* and the latter a *character-level embedding*.
Figure 2 illustrates an embedding assignment using the embedding lookup table. Each word is mapped to an embedding based on its vocabulary index. OOV words are mapped to a special "UNK" embedding. The poor handling of OOV words and the size of the embedding lookup table are the main issues when using the word-level embeddings and can be eliminated to a certain degree by the character-level embeddings.

3.1.1. Character-level embeddings

Character-level embeddings are word representations created by combining the embeddings of the characters in the word. To capture dependencies between the characters in the word, we use either a recurrent neural network (RNN) or a convolutional neural network (CNN).
Figure 2 shows the process of creating the character-level embedding using the RNN. A sequence of embedded characters $\mathbf{ch} = (ch_1, .., ch_n)$, $ch_i \in \mathbb{R}^{d_{ch}}$, which is created using an embedding lookup table similar to the word-level embedding method, is fed to the bidirectional RNN (BiRNN) (Graves and Schmidhuber, 2005). The BiRNN creates a context-aware representation of each character $\mathbf{h} = (h_1, .., h_n)$,[7] $h_i \in \mathbb{R}^{d_h}$ in a recurrent fashion using

[2] In the case of multiple MWE labels, the token receives only the first one.

[3] See code https://github.com/natalink/mwe_sharedtask/tree/refactor

[4] www.tensorflow.org

[5] http://ufal.mff.cuni.cz/neuralmonkey

[6] The code used during experiments together with the experiment configurations is available at https://github.com/ufal/neuralmonkey/tree/lrec_mwe/neuralmonkey.

[7] Since we use BiRNN there are actually two states, h_i^R and h_i^L

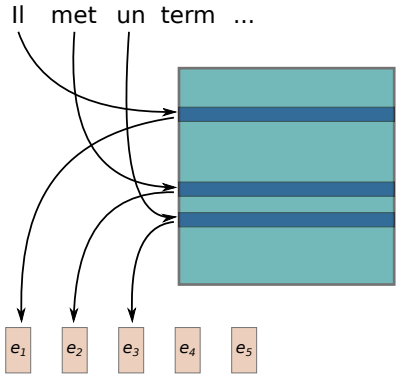


Figure 2: A word-level embedding example. Every input word is assigned an embedding depending on its vocabulary index.

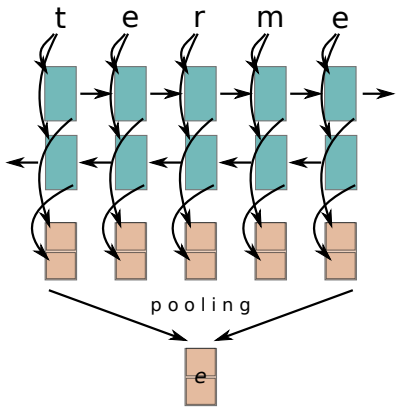


Figure 3: An illustration of the character-level RNN embedding. The outputs from each step of the BiRNN are concatenated and the whole output sequence is pooled to create the embedding of the word. The embeddings of the individual characters are omitted for simplicity.

the following formula:

$$h_i = f(h_{i-1}, ch_i) \quad (1)$$

The function $f(h, ch)$ is computed by a recurrent cell, usually the Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or the Gated Recurrent Unit (GRU) (Chung et al., 2014). After we get the context-aware representation of each character, we apply pooling (maximum or average) over the whole sequence **h** to get the embedding of the word.

Initially created and applied in image recognition task, CNNs became popular in the NLP tasks as well. The method relies on kernel sliding, however, the kernel slides over the sequence of embedded characters instead of image pixels. The Figure 4 shows the architecture of the CNN that can be used to create character-level embeddings. We apply a set of filters $F = [filt_1, .., filt_k]$ on the sequence of embedded characters. A filter $filt_j$ of width l takes an input $X \in \mathbb{R}^{l \times d_{ch}}$ and transforms it into a single output $Y \in \mathbb{R}^{d/|F|}$, where d is again the size of the output word embedding. We pad the sequence of characters to make

for each i, created by the forward and backward run respectively. The states are concatenated to create the representation h_i

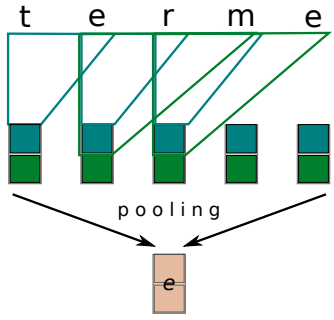


Figure 4: An illustration of the character-level convolutional embedding. The sequence of characters is padded to guarantee equal length of the filter outputs. We omit the embeddings of the individual characters for simplicity.

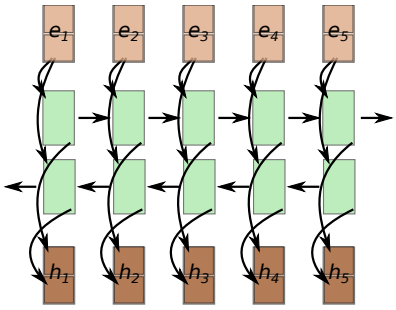


Figure 5: An illustration of the BiRNN encoder. Each embedding e_i is pasted to the RNN cell to produce a context-aware representation h_i.

each filter produce a sequence of the same length. These output sequences are then concatenated element-wise to produce a single sequence $\mathbf{h} = (h_1, ..h_n)$ and a we apply pooling to produce the embedding of the word.

To provide additional information for the tagger, we encode not only the word forms but also lemmas and POS tags that are available in the training data. For each token, we encode its word form, lemma and POS tag using a separate embedding lookup table in case of the word-level embedding and a separate character-level encoder in case of the character-level embedding. The resulting embeddings of the word form, lemma and POS tag are then concatenated to create the embedding of the token.

3.2. Encoder Layer

The purpose of the encoder layer is to take the embeddings of the words $\mathbf{e} = (e_1, .., e_n)$ provided by the embedding layer and transform them to a representation $\mathbf{h} = (h_1, .., h_n)$ where each embedding h_i contains additional information about its neighbors. We examine three methods of doing so: using a BiRNN, a deep convolutional network and a self-attentive network.

Figure 5 shows the architecture of the BiRNN encoder. The process is similar to the character-level BiRNN encoder: the input embeddings **e** are transformed to **h** using the following formula:

$$h_i = f(h_{i-1}, e_i) \quad (2)$$

The function f again represents the recurrent cell.

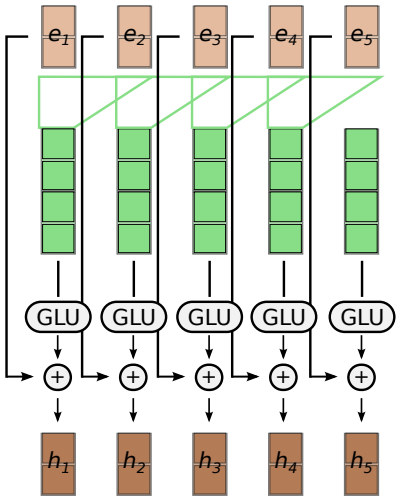


Figure 6: An illustration of the deep convolutional encoder with depth 1. The filter creates an itermediate embedding with double of the original embedding size and the GLU gating mechanism reduces the embedding back to the original size. Residual connections are applied to allow deep convolutions. The output of the layer can be used as input to another layer.

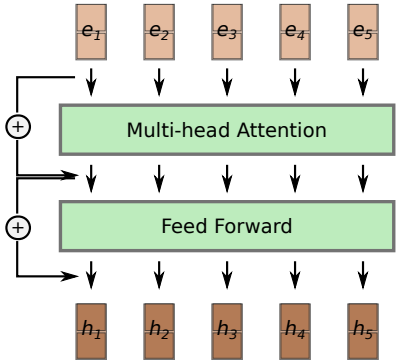


Figure 7: A simplified illustration of a single self-attentive layer consisting of a multi-head attention and feed forward sublayer. After each sublayer, residual connections and a layer normalization are applied. The layers can be stacked to create a deeper architecture.

Figure 6 describes the deep convolutional encoder architecture first used by a Facebook machine translation system (Gehring et al., 2017). In contrast to the RNN, convolutions do not provide explicit way to encode the position of a word in the input sequence. To counter this, we add additional positional information to the embeddings (Gehring et al., 2017; Vaswani et al., 2017). The position-aware embeddings are then transformed using a filter $F \in \mathbb{R}^{2d \times ld}$, where l is the width of the filter. The filter takes l input embeddings and transforms them into a single output embedding $h' \in \mathbb{R}^{2d}$. A Gated Linear Unit (GLU) (Dauphin et al., 2016), is applied on the h' as a gating mechanism to introduce non-linearity, producing $h \in \mathbb{R}^{d}$. We also add residual connections (He et al., 2016) to the output of the GLU to enable deep convolutions. These convolutional layers can be stacked one on top of another producing deeper representations. We use the output from the last layer as the input for the classifier layer.

	MWE-based F1	token-based F1
word-lvl	0.42	0.57
char-rnn-gru-avg	0.59	0.69
char-rnn-lstm-avg	0.59	0.69
char-rnn-gru-max	0.64	0.73
char-rnn-lstm-max	0.65	0.73
char-conv-5-avg	0.58	0.68
char-conv-6-avg	0.58	0.68
char-conv-5-max	**0.69**	**0.78**
char-conv-6-max	**0.69**	**0.78**

Table 1: Comparison between different embedding methods.

	MWE-based F1	token-based F1
birnn-softmax	0.69	0.78
birnn-crf	0.73	0.78
deep-convo-softmax	0.55	0.71
deep-convo-crf	0.59	0.72
self-att-softmax	0.70	0.78
self-att-crf	**0.74**	**0.79**

Table 2: Comparison between different encoder architectures and output classifiers.

The structure of the self-attentive encoder (Vaswani et al., 2017) is described in Figure 7. Similar to the convolutional encoder, the self-attentive encoder does not explicitly capture information about the position of the input in the sequence. Therefore, we use the position-aware embeddings again. We apply a multi-head attention mechanism on these embeddings (Vaswani et al., 2017) and a position-wise fully-connected layer. After each sublayer, a layer normalization (Ba et al., 2016) and a residual connections are added. Again, the process can be repeated by stacking multiple layers using the output of the previous layer as the input of the following one. We pass the output of the last layer to the classifier layer.

3.3. Classifier Layer

The classier layer takes the output representations $\mathbf{h} = (h_1, .., h_n)$ produced by the encoder layer and uses them to predict the target sequence. We compare two methods: a softmax classifier and a CRF classifier.

The softmax classifier first transforms each hidden representation h_i into a vector of logits $y_i \in \mathbb{R}^{|V|}$, where $|V|$ is the size of the target vocabulary. The logits y are then normalized using a softmax function creating a distribution over the target vocabulary. During training, we minimize the cross-entropy between the output distribution and the gold labels. During the inference, a label with the highest probability is selected.

The CRF classifier uses conditional random field to predict the whole output sequence instead of predicting each target label separately. This helps us to take into account dependencies between the predicted labels. Again, we first transform the hidden representation of each token into a vector of logits using a linear layer. During training, the

	System	MWE F1	token F1		System	MWE F1	token F1
BG	MUMULS-old	0.34	0.59	LT	MUMULS-old	0.00	0.00
	MUMULS	0.50	0.62		MUMULS	0.19	0.25
	PARSEME winner	0.61	0.66		PARSEME winner	0.28	0.25
CS	MUMULS-old	0.16	0.23	PT	MUMULS-old	0.44	0.60
	MUMULS	0.67	0.73		MUMULS	0.40	0.52
	PARSEME winner	0.71	0.73		PARSEME winner	0.67	0.71
DE	MUMULS-old	0.21	0.34	RO	MUMULS-old	0.77	0.84
	MUMULS	0.33	0.40		MUMULS	0.66	0.71
	PARSEME winner	0.41	0.45		PARSEME winner	0.77	0.84
FR	MUMULS-old	0.09	0.29	SL	MUMULS-old	0.31	0.45
	MUMULS	0.38	0.48		MUMULS	0.32	0.40
	PARSEME winner	0.51	0.61		PARSEME winner	0.43	0.47
IT	MUMULS-old	–	–	TR	MUMULS-old	0.34	0.45
	MUMULS	0.07	0.07		MUMULS	0.40	0.48
	PARSEME winner	0.40	0.44		PAESEME winner	0.55	0.55

Table 3: Comparison between our best systems configuration, old MUMULS system performance and the best submitted system across languages participating in the PARSEME Shared Task.

CRF classifer tries to minimize the negative log-likelihood of the gold target sequence $\mathbf{y} = (y_1, .., y_n)$ based on the following probability:

$$p(\mathbf{y}|\mathbf{h}; \theta) \sim epx(\sum_{i=0}^{n} s(y_i) + \sum_{i=1}^{n} s(y_i, y_{i-1})) \quad (3)$$

$s(y_i)$ is the score of the individual label based on its hidden representation h_i and $s(y_i, y_{i-1}$ is a transition score between the labels computed using transition parameters $\theta \in \mathbb{R}^{|V|x|V|}$, where $|V|$ denotes the size of the target label vocabulary. During the inference, the CRF classifier uses the Viterbi decoding algorithm (Forney, 1973) to output the sequence with the highest score.

4. Experiments

When we evaluated the suggested system configurations, we compared each layer configurations separately. We used the Czech dataset available for the PARSEME shared task. We used the MWE-based F1 measure metric to evaluate the performance of each system configuration. The metric is a standard F1 measure based on the precision and recall of the evaluated systems. For each MWE (represented as a set of word indices) in the reference, it searches for exact matches in the predicted MWEs. For comparison, we also used a fuzzy, token-based F1 measure which allows partial matches between the gold and predicted MWEs.

First we compared the variants of the embedding layer. We used the embedding size 100 for each word form, lemma and POS tag, resulting in a token embedding size 300. We fixed the encoder layer to a BiRNN with the hidden state size 300 and an LSTM recurrent cell. We only used the softmax classifier during this comparison. All experiments had a fixed dropout of 0.8. The word-level embedding method used a separate vocabularies for word forms, lemmas and POS tags. The character-level embeddings used the same character vocabulary for each factor. In the character-level RNN embedding, we set the size of the hidden state to the size of the input embedding. We compared the performance of both LSTM (char-rnn-lstm) and GRU (char-rnn-gru) cell. In the character-level CNN (char-conv), we tried sets of filters of lengths ranging from 2 to 5 and 2 to 6 respectively. In both character-level embedding methods, we compared both maximum (max) and average (avg) pooling.

Table 1 shows the individual performance of each embedding method. First, we can see that using the character-level embeddings brings significant improvement over the word-level embeddings. Second, the choice of the RNN cell seems to have little to no impact on the performance of the char-rnn embedding method. Finally, the results show that the char-conv embedding method yields the best results and that the maximum pooling method outperforms the average pooling.

Next, we compared the suggested encoder layer configurations. We used the convolutional character-level CNN with maximum pooling for embedding layer. The BiRNN on the encoder layer was identical to the one used during the embedding layer comparison. The deep convolutional encoder (deep-convo) had three layers, each having the filter width 3. The self-attention encoder (self-att) also had three layers, each having 10 attention heads and a feed-forward network with the hidden size 450. We chose the parameters so that each model had a comparable number of trainable variables. For each encoder, the size of the output hidden states was identical to the size of the input embeddings. We compared both the softmax and CRF classifier with each encoder.

Table 2 compares the performance of each encoder architecture and classifier method. The self-attention encoder achieved the best results being slightly better than the BiRNN encoder. The results also show that replacing the softmax layer with the CRF classifier consistently improves the performance.

Based on the results of the previous experiments we chose the following architecture to train the models for the other PARSEME Shared Task languages: convolutional character-level embedding network with filters of width 2 to 6 and maximum pooling layer, self-attention encoder layer and a CRF classifier. Table 3 shows a comparison

of the models with the results reported in the shared task. We can see, that our improvements were reflected in the MUMULS performance when compared with our old submission. However, we still were not able to beat the best submission. The lower performance of the improved MUMULS on the Romanian requires additional investigation in the future.

5. Conclusion

We described an ongoing work on improving MUMULS, a neural-based system for automatic identification of verbal multiword expressions. We compare several state-of-the-art architectures, experiment with different embedding methods and implement sequence model for label prediction using CRF. The results show that for most of the languages, our modifications bring additional boost in the system performance.

In the future, we plan to further investigate the possibilities of scaling the presented architectures and studying the model capacities with respect to the provided data. Special attention will be focused on investigating the decrease in performance for the Romanian because it is a language in which the old version of MUMULS yielded the best results.

6. Acknowledgements

The first author has been supported by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (projects LM2015071 and OP VVV VI CZ.02.1.01/0.0/0.0/16_013/0001781), by the Charles University SVV project number 260 453 and by the Meta-Net/T4ME Net project of the European Union (project FP7-ICT-2009-4-249119).

The second author has been supported by the postdoctoral fellowship grant of the Hong Kong Polytechnic University, project code G-YW2P.

7. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Al Saied, H., Constant, M., and Candito, M. (2017). The ATILF-LLF System for Parseme Shared Task: a Transition-based Verbal Multiword Expression Tagger. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 127–132, Valencia, Spain, April. Association for Computational Linguistics.

Ba, L. J., Kiros, R., and Hinton, G. E. (2016). Layer normalization. *CoRR*, abs/1607.06450.

Boroş, T., Pipa, S., Barbu Mititelu, V., and Tufiş, D. (2017). A data-driven approach to verbal multiword expression detection. PARSEME Shared Task system description paper. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 121–126, Valencia, Spain, April. Association for Computational Linguistics.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y., (2014). *Empirical evaluation of gated recurrent neural networks on sequence modeling*.

Dauphin, Y. N., Fan, A., Auli, M., and Grangier, D. (2016). Language modeling with gated convolutional networks. *arXiv preprint arXiv:1612.08083*.

Forney, G. D. (1973). The viterbi algorithm. *Proc. of the IEEE*, 61:268 – 278, March.

Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. (2017). Convolutional sequence to sequence learning.

Graves, A. and Schmidhuber, J. (2005). Framewise Phoneme Classification with Bidirectional LSTM and other Neural Network Architectures. *Neural Networks*, 18(5):602–610.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Helcl, J. and Libovický, J. (2017). Neural Monkey: An Open-source Tool for Sequence Learning. *The Prague Bulletin of Mathematical Linguistics*, 107:5–17.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovář, V., Michelfeit, J., Rychlý, P., and Suchomel, V. (2014). The sketch engine: ten years on. *Lexicography*, pages 7–36.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In Kevin Knight, et al., editors, *HLT-NAACL*, pages 260–270. The Association for Computational Linguistics.

Maldonado, A., Han, L., Moreau, E., Alsulaimani, A., Chowdhury, K. D., Vogel, C., and Liu, Q. (2017). Detection of Verbal Multi-Word Expressions via Conditional Random Fields with Syntactic Dependency Features and Semantic Re-Ranking. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 114–120, Valencia, Spain, April. Association for Computational Linguistics.

Pérez-Ortiz, J. A. and Forcada, M. L. (2001). Part-of-speech tagging with recurrent neural networks. In *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2001*, pages 1588–1592.

Ramisch, C. (2015). *Multiword Expressions Acquisition: A Generic and Open Framework*, volume XIV of *Theory and Applications of Natural Language Processing*. Springer.

Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., and Doucet, A. (2017). The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, Valencia, Spain.

Simkó, K. I., Kovács, V., and Vincze, V. (2017). USzeged: Identifying Verbal Multiword Expressions with POS Tagging and Parsing Techniques. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 48–53, Valencia, Spain, April. Association for Computational Linguistics.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need.

Wen, T.-H., Gasic, M., Mrksic, N., hao Su, P., Vandyke, D., and Young, S. J. (2015). Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In Lluís Màrquez, et al., editors, *EMNLP*, pages 1711–1721. The Association for Computational Linguistics.

Designing a Russian Idiom-Annotated Corpus

Katsiaryna Aharodnik, Anna Feldman, Jing Peng
CUNY, The Graduate Center, Montclair State University
New York, New York, Montclair, New Jersey, United States
kaharodnik@gradcenter.cuny, {feldmana,pengj}@montclair.edu

Abstract

This paper describes the development of an idiom-annotated corpus of Russian. The corpus is compiled from freely available resources online and contains texts of different genres. The idiom extraction, annotation procedure, and a pilot experiment using the new corpus are outlined in the paper. Considering the scarcity of publicly available Russian annotated corpora, the corpus is a much-needed resource that can be utilized for literary and linguistic studies, pedagogy as well as for various Natural Language Processing tasks.

Keywords: idioms, annotation, corpus, Russian.

1. Introduction

In recent years, there has been a growing interest in exploring the questions of automatic processing of semantic relationships and specifically those that are not trivial to define and disambiguate. Among these questions is the problem of automatic identification of figurative language within a large body of text. Largely, the problem lies in the ambiguous nature of idiomatic expressions and identifying the cues for idiom recognition. Some expressions can be interpreted either literally or idiomatically depending on the context in which they occur. Several approaches have been explored in finding a better solution to this problem (e.g., Fazly et al., 2009; Cook et al., 2007; Katz and Giesbrecht, 2006; Sporleder & Li, 2009; Li & Sporleder, 2010; Pradhan et al., 2017; Peng & Feldman, 2016(a, b); Peng et al., 2015; Peng et al., 2014, among others). Unfortunately, the corpora that could be used for training idiom classifiers are scarce, especially if one turns to languages other than English.

In this paper, we describe an idiom-annotated corpus for Russian. This corpus is a valuable language resource which can be used for various Natural Language Processing (NLP) tasks, such as automatic idiom recognition. Also, it can be utilized as a pedagogical tool for teaching the intricacies of the Russian language or as a corpus for linguistic investigations. Our corpus is available for research purposes https://github.com/kaharodnik/Ru_idioms. A pilot experiment using the idiom-annotated corpus is also described in the paper.

2. Motivation

Idioms lack a clear observable relation between the linguistic meaning and interpretation. Moreover, expressions can be ambiguous between idiomatic and literal interpretation depending on the context in which they occur (e.g., *sales hit the roof* vs. *the roof of the car*). Fazly et al.'s (2009) analysis of 60 idioms from the British National Corpus (BNC) has demonstrated that close to half of such expressions have a clear literal meaning; and of those with a literal meaning, on average around 40% of their usages are literal. Therefore, idioms present great challenges for many NLP applications, such as machine translation.

There has been substantial computational research on idioms, with an emphasis on English.

Previous approaches to idiom detection can be classified into two groups: 1) type-based extraction, i.e., detecting idioms at the type level; 2) token-based detection, i.e., detecting idioms in context. Type-based extraction relies on the idea that idiomatic expressions exhibit certain linguistic properties such as non-compositionality that can distinguish them from literal expressions (Sag et al., 2002; Fazly et al., 2009). While many idioms can be characterized by these properties, a number of idioms fall on the continuum from being compositional to being partly unanalyzable to completely non-compositional (Cook et al., 2007). Katz and Giesbrecht (2006), Birke and Sarkar (2006), Fazly et al. (2009), Sporleder and Li (2009), Li and Sporleder (2010), among others, emphasize that type-based approaches do not work on expressions that can be interpreted either idiomatically or literally depending on the context, and thus an approach that considers tokens in context is more appropriate for idiom recognition. Different token-based approaches have been proposed for more efficient ways of idiom identification. Some of them use topic-based representation (Peng et al. 2014); others utilize word embeddings (Peng et al., 2015, 2016; Pradhan et al., 2017). The above approaches rely on corpora annotated for both literal and idiomatic interpretations of expressions. The corpus proposed in this paper, besides its more general purpose, satisfies this requirement and thus is an important contribution to the community of researchers working on idiom detection in general and on Russian idioms in particular.

3. Corpus Description

Following the rationale for token-based approach, each corpus entry contains a target expression itself (idiomatic or literal) and two paragraphs of context. Thus, each entry is divided into three paragraphs: one paragraph preceding the paragraph with a target expression and the other following the paragraph with a target expression. Each target expression can be identified as both, idiomatic or literal, depending on the context. Each file of the corpus contains one entry. The examples of two corpus entries below show one-paragraph entries for literal (L) and idiomatic (I) interpretations of a target expression ***на чемоданах*** (*na čemodanah*) - *on suitcases.* Example 1, Literal:

Народ табором расположился ***на чемоданах*** *и баулах, расслабленно сидел, опустив руки, а кто-то доставал походную снедь, по палубе расползались ароматы жареных кур и копченой рыбы. У судна стали собираться крикливые чайки.*

In the above example, the target expression ***на чемоданах*** *(na čemodanah) on suitcases* is located in the second paragraph of the corpus entry. It can be interpreted literary *to sit on suitcases*. In the corpus entry below, the same target expression is interpreted idiomatically *to be packed and waiting, to be unsettled*. Generally, this idiom is similar to the English idiom *to live out of a suitcase.* Example 2, Idiomatic:

Шло время, но разрешения из ОВИРа не приходило. Афганская кампания ввода ограниченного контингента войск смешала все карты. Запах холодной войны проникал в самые отдаленные сферы жизни и прежде всего в государственную политику по так называемому тогда воссоединению семей. Единственная законная возможность уехать из страны Советов все более переходила в область мифов. Казалось, что выезд закрыт навсегда. Ждать всегда противно, а ждать разрешения на выезд противно вдвойне. Сколько времени можно жить ***на чемоданах****? Год, два, десять? Тем, кто работал сторожами и лифтерами, было вообще грустно: ни работы нормальной, ни перспектив.*

These examples demonstrate that an entry provides substantial context for each target expression in the corpus. The preceeding paragraph and the one following it are omitted in the examples.

To make the corpus balanced across written registers, it was compiled from texts of different genres: fiction and non-fiction, Wikipedia style text. The fiction sub-corpus was also split into two parts: Classical Russian Literature and Modern Russian Literature. The texts for this part were extracted from freely available online Russian library, Moshkov's library (http://lib.ru/). Classical literature texts were taken from Классика(Classical)/Проза(Prose). This part of corpus consists of Russian prose of late nineteenth-early twentieth century. Similarly, Modern literature sub-corpus consists of prose from Современная (Modern)/Проза (Prose) part of the library. In Modern Prose, the texts are written by a variety of Russian authors dating back to the second half of the twentieth and twenty-first centuries. The Wikipedia sub-corpus (Ru Wiki) was created from Russian Wikipedia freely available at http://linguatools.org/tools/corpora/wikipedia-monolingual-corpora/. In the corpus, the files were saved in folders according to genres, making it possible for researchers to conduct comparative analyses. Each text for Classical and Modern literature sub-corpora was saved in a separate file. The Ru Wiki sub-corpus was analyzed as a single XML file. Table 1 describes the total number of tokens used for idioms extraction for each part of the corpus.

Once the Russian corpus was compiled, the list of target expressions (idioms) of interest was created (see Section 4).

Corpus	# tokens
Classical Prose	111,725,751
Modern Prose	46,996,232
Ru Wiki	486,474,989

Table 1: Description of Sub-Corpora

4. Target Expressions

For the list of idioms, a Russian-English dictionary of idioms was used as a primary source (Lubensky, 2013). Initially, 150 idioms (target expressions) were included in the list. The rationale for choosing a certain target expression was that each expression could be interpreted as either idiomatic or literal depending on the context. Some idioms were not found in the source files and were excluded from the list. The final list consisted of 100 target expressions. This final list was used for compiling the actual annotated corpus.

The list of idioms included only multiword expressions (MWE). Each target expression consisted of more than one-word token, with their length ranging from two-word tokens, e.g., *длинный язык- long tongue*, to four-word tokens as in *с пеной у рта – with frothing at the mouth*. Syntactically, target expressions were not limited to a single structure. They could be separated into three groups: Noun Phrases (NP), Prepositional Phrases (PP), and Verb Phrases (VP) types of constructions. The PP type included Preposition + Noun, e.g., *без головы* (*without the head*), Preposition + Adjective/Attributive Pronoun + Noun, e.g., *на свою голову (on one's head)*, the NPs included Adjective/Possessive Pronoun + Noun e.g., *второй дом* (*second home*), and VP type included Verb + Preposition + Noun, e.g., *плыть по течению* (*to go with the flow*), and Verb + Noun, e.g., *поставить точку* (*to put a stop*). Table 2 provides a list of syntactic constructions with their counts. The list included idioms in their dictionary form, but each idiomatic expression was extracted from the compiled corpora in any form it appeared in files (conjugated forms for verbs or declined forms for adjectives and nouns).

4.1 Extracting Target Expressions

A target token is defined as a multiword expression that can be identified as either idiomatic or literal within the text. Each target expression was extracted with one preceding and one following paragraph from a source text file. Thus, one entry is defined as a three-paragraph text in one file.

Syntactic Construction	Russian	English	Count
Adj (Poss Pron) + Noun	Черный ворон	Black raven	33
Prep+Noun	Без головы	Without the head	82
Prep+Adj+Noun	На мою голову	On my head	78
Verb+(Prep)+Noun	Вцепить ся в глотку	To grab one's throat	50
Adv + Verb	Жирно будет	Too greasy (too much)	9
Noun + Short Adj	Концерт Окончен	The concert is over	4
Prep+Noun+Verb	Куда ветер дует	Where the wind blows	7

Table 2: Syntactic Constructions of Idiomatic Expressions

Each target expression was extracted following the steps below:

1. Convert the online text file to html format. This was done to preserve the html tags and use the tags for paragraph extraction.
2. Save each file as a plain text document with preserved html tags.
3. Extract each target expression (token) from each html document in a three-paragraph format, with the second paragraph containing a target expression.
4. Save each three-paragraph entry in a separate text file.

Overall, 100 tokens/target expressions were used to create the idiom-annotated corpus. The number of files in each sub-corpus varied depending on the amount of the idiomatic/literal expressions found in the sub-corpora.

4.2 Annotation

Once the expressions were extracted, each file was annotated manually by two Russian native speakers with overall high inter-annotator agreement (Kappa 0.81). Each target expression was assigned a tag Idiomatic (I) or Literal (L). Once the annotator made a decision about the tag, the three paragraph entries were saved in a text file format. In some cases, the resulting files did not have a required amount of paragraphs and were marked as a no paragraph label *_np* within a file name, e.g., *na_moyu_golovu_I_3_np.txt*. This could have happened for several reasons. Sometimes, preceding or following paragraphs could have been contaminated with tags without a sufficient amount of actual text. In these cases, the files were cleaned to include only intelligible text. In other cases, the target expressions were found in the first or last paragraph of a source file, hence they were missing the required amount of context. However, these files were not excluded from the corpus, since they can be still used for the analyses. The list of 10 most frequent target expressions extracted for the corpus is provided in Table 3. Table 3 also includes the counts of idiomatic and literal interpretations for each idiom. For each entry, an XML file was created with a label for an idiomatic expression within a file.

As the result, the idiom-annotated Russian corpus contained the three sub-corpora of files in plain text and XML formats with each target expression, three paragraph entries per file. The annotators' labels are assigned within XML files and are reflected in the folder names for plain text files. README files are also provided for each sub-corpora. Each README file lists the file directory for an idiomatic expression (File folder/File Name), the corresponding target expression in Russian, its translation in English, and the number of tokens (words and punctuation) prior to the first token of the idiomatic expression. The total counts for literal and idiomatic expressions extracted per sub-corpora are listed in Table 4.

#	Target	Gloss	Interpretation	I	L
1	s bleskom	with flying colors	brilliantly	246	78
2	na svoju golovu	on your own head	pain in the neck	185	58
3	na vysote	at the height	rise to the occasion	294	438
4	smotret' v glaza	look into the eyes	face (challenges)	48	83
5	čerez golovu	over the head	go over someone's head	100	316
6	na nožax	with the knives	to be at daggers drawn	53	43
7	po barabanu	on the drums	couldn't care less	86	25
8	vtoroj dom	second home	second home	14	40
9	vyše sebja	above oneself	beyond the possible	57	22
10	dlinnyj jazyk	long tongue	chatterbox	37	29

Table 3: Ten most frequent target expressions.

Sub-Corpus	# Literal Expressions	# Idiomatic Expressions	#Total files
Classical Literature	2,100	1,231	3,331
Modern Literature	612	803	1,415
Russian Wiki	315	386	701

Table 4: Literal and Idiomatic Total Counts per Sub-Corpora.

5. Idiom Detection Experiment

Below we report the results of a pilot idiom detection experiment for which we used the idiom-annotated corpus described in this paper. For this pilot experiment, we follow the hypotheses and the methodology described in Peng et al. (2018). The automatic idiom detection approach is based on two hypotheses: (1) words in a given text segment that are representatives of the local context are likely to associate strongly with a literal expression in the segment, in terms of projection of word vectors onto the vector representing the literal expression; (2) the context word distribution for a literal expression in word vector space will be different from the distribution for an idiomatic one (similarly to Firth, 1957; Katz and Giesbrecht, 2006).

5.1 Projection based on Local Context Representation

To address the first hypothesis, we propose to exploit recent advances in vector space representation to capture the difference between local contexts (Mikolov et al., 2013a; Mikolov et al., 2013b).

A word can be represented by a vector of fixed dimensionality q that best predicts its surrounding words in a sentence or a document (Mikolov et al., 2013a; Mikolov et al., 2013b). Given such a vector representation, our first proposal is the following. Let v and n be the vectors corresponding to the verb and noun in a target verb-noun construction, as in *blow whistle*, where $v \in \Re^q$ represents *blow* and $n \in \Re^q$ represents *whistle*. Let

$$\sigma_{vn} = v+n \in \Re^q.$$

Thus, σ_{vn} is the word vector that represents the composition of verb v and noun n, and in our example, the composition of *blow* and *whistle*. As indicated in (Mikolov et al., 2013b), word vectors obtained from deep learning neural net models exhibit linguistic regularities, such as additive compositionality. Therefore, σ_{vn} is justified to predict surrounding words of the composition of, say, *blow* and *whistle* in a literal context. Our hypothesis is that on average, the projection of v onto $\sigma_{blowwhistle}$, (i.e., $v \cdot \sigma_{blowwhistle}$, assuming that $\sigma_{blowwhistle}$ has unit length), where vs are context words in a literal usage, should be greater than $v \cdot \sigma_{blowwhistle}$, where vs are context words in an idiomatic usage.

For a given vocabulary of m words, represented by matrix

$$V = [v_1, v_2, \cdots, v_m] \in \Re^{q \times m},$$

We calculate the projection of each word v_i in the vocabulary onto σ_{vn}

$$P = V^t \sigma_{vn} \quad (1)$$

where $P \in \Re^m$, and t represents transpose. Here we assume that σ_{vn} is normalized to have unit length. Thus, $P_i = v_i^t \sigma_{vn}$ indicates how strongly word vector v_i is associated with σ_{vn}. This projection forms the basis for our proposed technique.

Let $D = \{d_1, d_2, \cdots, d_l\}$ be a set of l text segments (local contexts), each containing a target VNC (i.e., σ_{vn}). Instead of generating a term by document matrix, where each term is *tfidf* (product of term frequency and inverse document frequency), we compute a term by document matrix

$M_D \in \Re^{m \times l}$, where each term in the matrix is

$$p \cdot idf. \quad (2)$$

That is, the product of the projection of a word onto a target VNC and inverse document frequency. That is, the term frequency (tf) of a word is replaced by the projection of the word onto σ_{vn} (1). Note that if segment d_j does not contain word v_i, $M_D(i, j) = 0$, which is similar to *tf-idf* estimation. The motivation is that topical words are more likely to be well predicted by a literal VNC than by an idiomatic one. The assumption is that a word vector is learned in such a way that it best predicts its surrounding words in a sentence or a document (Mikolov et al., 2013a; Mikolov et al., 2013b). As a result, the words associated with a literal target will have larger projection onto a target σ_{vn}. On the other hand, the projections of words associated with an idiomatic target VNC onto σ_{vn} should have a smaller value.

We also propose a variant of $p \cdot idf$ representation. In this representation, each term is a product of p and typical *tf-idf*. That is,

$$p \cdot tf \cdot idf. \quad (3)$$

5.2 Local Context Distributions

Our second hypothesis states that words in a local context of a literal expression will have a different distribution from those in the context of an idiomatic one. We propose to capture local context distributions in terms of scatter matrices in a space spanned by word vectors (Mikolov et al., 2013a; Mikolov et al., 2013b).

Let $d = (w_1, w_2 \cdots, w_k) \in \Re^{q \times k}$

be a segment (document) of k words, where $w_i \in \Re^q$ are represented by a vectors (Mikolov et al., 2013a; Mikolov et al., 2013b). Assuming w_is have been centered, we compute the scatter matrix

$$\Sigma = d^t d, \quad (4)$$

where Σ represents the local context distribution for a given target VNC.

Given two distributions represented by two scatter matrices Σ_1 and Σ_2, a number of measures can be used to compute the distance between Σ_1 and Σ_2, such as Choernoff and Bhattacharyya distances (Fukunaga, 1990). Both measures require the knowledge of matrix determinant. We propose to measure the difference between Σ_1 and Σ_2 using matrix norms. We have experimented with the Frobenius norm and the spectral norm. The Frobenius norm evaluates the difference between Σ_1 and Σ_2 when they act on a standard basis. The spectral norm, on the other hand, evaluates the difference when they act on the direction of maximal variance over the whole space.

5.3 Methods

We carried out an empirical study evaluating the performance of the proposed techniques. The following methods are evaluated:

1. *p·id f*: compute term by document matrix from training data with proposed *p·id f* weighting (2).
   p · t f · id f: compute term by document matrix from training data with proposed p*tf-idf weighting (3).
2. $CoVAR_{Fro}$: proposed technique (4) described in Section 2.2, the distance between two matrices is computed using Frobenius norm.
3. $CoVAR_{Sp}$: proposed technique similar to $CoVAR_{Fro}$. However, the distance between two matrices is determined using the spectral norm.

For methods 3 and 4, we compute the literal and idiomatic scatter matrices from training data (4). For a test example, compute a scatter matrix according to (4), and calculate the distance between the test scatter matrix and training scatter matrices using the Frobenius norm for method 3, and the spectral norm for method 4.

5.4 Results

The results of the experiment suggest that for Russian our algorithm performs similarly to English, even considering the fact that Russian is a more morphologically complex language and has a relatively free word order. Specifically, the results demonstrate that one of our proposed methods - $CoVAR_{Fro}$ performs with highest average accuracy for precision and recall measures. The results are described in Table 5.

6. Corpus Importance

In this paper, we described the development of a Russian-language corpus annotated for idioms. This corpus is pivotal for a variety of NLP tasks such as idiom detection, as well as a useful resource for various linguistic analyses and pedagogical applications. The corpus contains only those expressions whose idiomatic or literal interpretation depends on context. The format of the corpus allows the user to easily search for idioms in context. In addition, unlike previous corpora annotated for idioms (e.g., Cook et al., 2008), this corpus contains expressions of various syntactic types.

Method	na svoju golovu get into trouble	na vysote to be at one's best	smotret' v glaza to face (a challenge)	Ave
Precision				
p·id f	0.75	0.49	0.40	0.55
p·t f ·id f	0.80	0.50	0.50	0.60
$CoVAR_{Fro}$	0.80	0.71	0.49	**0.67**
$CoVAR_{sp}$	0.78	0.64	0.54	0.65
Recall				
p·id f	0.73	0.83	0.40	0.65
p·t f ·id f	0.76	0.81	0.42	0.66
$CoVAR_{Fro}$	0.88	0.81	0.50	**0.73**
$CoVAR_{sp}$	0.76	0.76	0.50	0.67
Accuracy				
p·id f	0.63	0.64	0.57	0.61
p·t f ·id f	0.68	0.66	0.67	0.67
$CoVAR_{Fro}$	0.76	0.82	0.65	**0.74**
$CoVAR_{sp}$	0.68	0.77	0.68	0.71

Table 5: Average performance of competing methods on Russian idioms.

More generally, the described corpus facilitates research in the Russian language. Since the corpus contains sections from different time periods and genres, it is possible to investigate the usage of idioms in fiction vs. non-fiction or explore how figurative language changes over time. The variety of grammatical constructions provides insights into the syntactic nature of Russian idioms, especially those that can be productively used in either idiomatic or literal sense.

In this paper, we also reported the results of a pilot experiment using the corpus. The experiment demonstrates the feasibility of using the corpus for automated idiom identification approaches. We are planning to expand the size of the corpus in the future, by extracting more types of target expressions and adding other genres.

7. Acknowledgments

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-16-1-0261 and a National Science Foundation grant IIS-1319846.

8. References

Birke, J. and Sarkar, A. (2006). A Clustering Approach to the Nearly Unsupervised Recognition of Nonliteral Language. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL'06), pages 329–226, Trento, Italy.

Cook, P., Fazly, A., Stevenson, S. (2008). The VNC-tokens dataset. Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008*)*.

Fazly, A., Cook, P., Stevenson, S. (2009). Unsupervised Type and Token Identification of Idiomatic Expressions. *Computational Linguistics 35* (2009), 61-103

Katz, G., Giesbrecht, E. (2006). Automatic Identification of Non-compositional Multiword Expressions using Latent Semantic Analysis. Proceedings of the ACL/COLING-06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties. (2006) 12-19

Li, L., Sporleder, C. (2010). Using Gaussian Mixture Models to Detect Figurative Language in Context. In: *Proceedings of NAACL/HLT 2010*.

Lubensky, S. (2013). Russian-English Dictionary of Idioms. Yale University Press.

Peng, J., Aharodnik K., and Feldman A. (2018). A Distributional Semantics Model for Idiom Detection: The Case of English and Russian. In the Proceedings of the 10th International Conference on Agents and Artifificial Intelligence (Special Session on Natural Language Processing in Artificial Intelligence - NLPinAI 2018).

Peng, J. and Feldman, A. (2016). Experiments in Idiom Recognition. *Proceedings of the 26th International Conference on Computational Linguistics (COLING). Osaka, Japan*.

Peng, J. and Feldman, A. (2016). In God We Trust. All Others Must Bring Data. — W. Edwards Deming — Using word embeddings to recognize idioms. Proceedings of the 3rd Annual International Symposium on Information Management and Big Data — SIMBig, Cusco, Peru.

Peng, J., Feldman, A., and Jazmati, H. (2015). Classifying Idiomatic and Literal Expressions Using Vector Space Representations. Proceedings of the Recent Advances in Natural Language Processing (RANLP) conference 2015, Hissar, Bulgaria, September 2015

Peng, J., Feldman, A., and Vylomova, E. (2014). Classifying Idiomatic and Literal Expressions Using Topic Models and Intensity of Emotions. In Proceedings of the 2014 Empirical Methods for Natural Language Processing Conference (EMNLP).

Pradhan, M., Peng, J., Feldman, A., and Wright, B. (2017)**.** Idioms: Humans or machines, it's all about context. *In Proceedings of 18th International Conference on Computational Linguistics and Intelligent Text Processing. Budapest. Hungary*.

Sag, I.A., Baldwin, T., Bond, F., Copestake, A., Flickinger, D. (2002). Multiword expressions: A Pain in the Neck for NLP. Proceedings of the 3rd International Conference on Intelligence Text Processing and Computational Linguistics (CICLing 2002), Mexico City, Mexico (2002) 1–15

Sporleder, C., Li, L. (2009). Unsupervised Recognition of Literal and Non-literal Use of Idiomatic Expressions. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL '09), Morristown, NJ, USA, Association for Computational Linguistics (2009) 754-762

DeepTC – An Extension of DKPro Text Classification for Fostering Reproducibility of Deep Learning Experiments

Tobias Horsmann and Torsten Zesch
Language Technology Lab
Department of Computer Science and Applied Cognitive Science
University of Duisburg-Essen, Germany
{tobias.horsmann,torsten.zesch}@uni-due.de

Abstract

We present a deep learning extension for the multi-purpose text classification framework *DKPro Text Classification (DKPro TC)*. DKPro TC is a flexible framework for creating easily shareable and reproducible end-to-end NLP experiments involving machine learning. We provide an overview of the current state of DKPro TC, which does not allow integration of deep learning, and discuss the necessary conceptual extensions. These extensions are based on an analysis of common deep learning setups found in the literature to support all common text classification setups, i.e. single outcome, multi outcome, and sequence classification problems. Additionally to providing an end-to-end shareable environment for deep learning experiments, we provide convenience features that take care of repetitive steps, such as pre-processing, data vectorization and pruning of embeddings. By moving a large part of this boilerplate code into DKPro TC, the actual deep learning framework code improves in readability and lowers the amount of redundant source code considerably. As proof-of-concept, we integrate Keras, DyNet, and DeepLearning4J.

Keywords: Reproducibility, Deep Learning, DKPro TC, Keras, Dynet, DeepLearning4J

1. Motivation

Experiments based on deep neural networks pose huge challenges to reproducibility. An experiment consists not just of the actual neural network architecture, but also of a potentially large number of processing steps to prepare the data. Furthermore, countless network parameters exist, which can greatly affect a network's performance. Reproduction attempts, thus, lead to a high amount of time spent with constructing comparable processing setups. Even if the deep learning code is released, code that applies all preprocessing steps is often missing. Additional effort is often necessary to install and configure required third-party tools.

A potential solution to these reproducibility challenges is DKPro Text Classification (DKPro TC)[1] (Daxenberger et al., 2014). DKPro TC ensures that the same preprocessing is automatically applied to any (new) dataset and provides convenience services such as an automatic installation of third-party tools. DKPro TC experiments are end-to-end shareable, enabling a quick and easy execution of experiments by other researchers. However, until now, DKPro TC only supports shallow learning frameworks. In this work, we present a deep learning extension to DKPro TC called DeepTC. In addition to improved reproducibility, DeepTC also eases architecture analysis by moving boilerplate code for pruning word embeddings and vectorization into DeepTC. This leads to a considerably reduced amount of framework-specific deep learning code. As proof-of-concept, we integrate the deep learning frameworks Keras[2], Dynet (Neubig et al., 2017) and DeepLearning4J[3].

2. DKPro Text Classification (DKPro TC)

We start with an overview of the current state of DKPro TC and discuss how it helps NLP researchers with their daily work. DKPro TC is a Java-based open-source software framework build upon the UIMA architecture (Ferrucci and Lally, 2004) and the lightweight DKPro Lab framework (Eckart de Castilho and Gurevych, 2011) for parameter sweeping experiments. DKPro TC provides an intermediate software layer that harmonizes the use of various machine learning frameworks. The same experimental setup is easily executed with one or more classifiers, which enables a direct comparison of different implementations. The user defines feature extractors, which collect the information the classifier uses for training a model. DKPro TC transforms the extracted feature information into the data format required by the respective classifier. Hence, the user is completely shielded from the intrinsic data format details required by a certain implementation. Furthermore, DKPro TC allows running experiments as train/test or cross-validation setups and takes care of all data splitting operations, execution, and aggregation of results. Required pre-processing components are automatically downloaded and installed. In summary, DKPro TC allows sharing self-contained and executable experiments with other researchers.

As of version 0.9.0, DKPro TC supports: Weka (Hall et al., 2009), LibLinear (Fan et al., 2008), LibSvm (Chang and Lin, 2011), SvmHmm (Joachims, 2008), and CrfSuite (Okazaki, 2007) that cover the common machine learning tasks in NLP, i.e. single outcome, multi-outcome and sequence classification.

2.1. Design Goals

DKPro TC is designed around three design goals: (i) reproducibility, (ii) convenience, and (iii) applicability.

Reproducibility is achieved by using only software components that are released in public repositories such as Maven Central. This ensures that software remains available even if components are no longer maintained. Furthermore, all parametrization details of the experiment, e.g.

[1] https://github.com/dkpro/dkpro-tc.git
[2] https://keras.io
[3] https://deeplearning4j.org

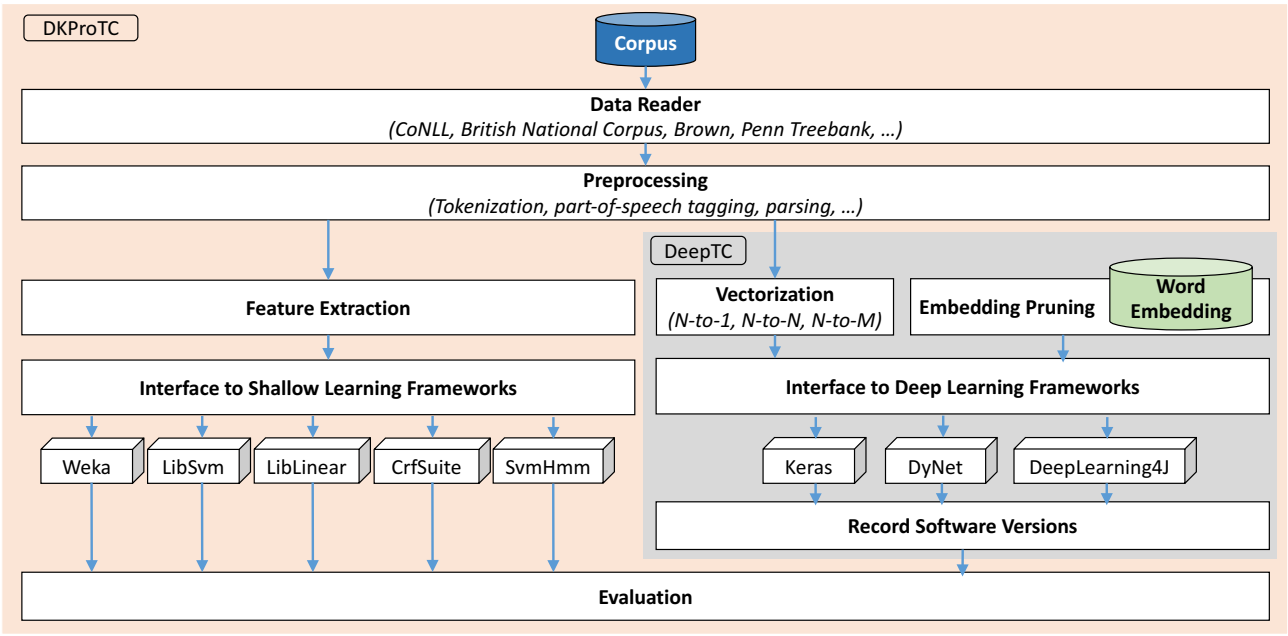


Figure 1: Processing Schema for Experiments in DKPro TC

classifier parametrization, features, and configuration of pre-processing tools, are automatically stored in a DKPro TC project for sharing the project right away.

Convenience is achieved by (i) easy-to-implement feature extractors with frequently needed ones being already pre-defined, and (ii) automatic installation of third-party components from public repositories. Additionally, DKPro TC integrates DKPro Core (Eckart de Castilho and Gurevych, 2014) and thus provides a rich source of tools such as tokenizers, part-of-speech taggers, or lemmatizers, which can be added in a plug-and-play fashion as processing component. These tools are automatically downloaded and installed as Maven artifacts. This provides a high degree of flexibility in terms of experimenting with various pre-processing tools and picking the best working one for a certain task. Of course, researchers can always implement their own UIMA processing components.

Applicability DKPro TC supports all common machine learning setups related to text classification tasks, i.e. single outcome (e.g. sentiment analysis), multi outcome or sequence classification (e.g. part-of-speech tagging), and regression (e.g. assessment of text reading difficulty).

2.2. Shallow Architecture

Figure 1 shows a conceptual overview of DKPro TC.

Reader The corpus data is read into DKPro TC by a reader component. Via DKPro Core dozens of common NLP formats are supported, for instance CoNLL, TEI, or Penn Treebank (Marcus et al., 1993).

Preprocessing In this step, an optional pre-processing can be applied, which might entail tasks such as tokenization or part-of-speech tagging.

Feature Extraction The feature extractors are applied to the data with access to information created during the pre-processing step. The extracted information is temporarily stored in an intermediate data format.

Interface to Shallow Learning Frameworks The feature information is transformed into the data format of the selected machine learning framework.

Evaluation If test data is provided, the trained model is applied to this dataset (after running through the same pre-processing and feature extraction as the train data). Many commonly used metrics such as accuracy, F-Score or Pearson correlation can be computed during evaluation. In case of cross-validation, aggregated results over all folds are automatically provided.

3. DeepTC – A Deep Learning Extension

Software focusing on the *shallow learning* paradigm is not easily extendible to support the *deep learning* paradigm. The conceptual differences between both paradigms make such an extension challenging, i.e. the shallow paradigm learns a model from a representation created from human defined features while the deep paradigm learns a suited representation by itself. Furthermore, a meaningful extension must not just work on a technical level, but also sustain the advantages of taking workload from the user. Consequently, we conducted an analysis of common deep learning setups in the literature to learn about the challenges to reproducibility and convenience. This led to the *DeepTC* extension shown in Figure 1.

3.1. Format

Many deep learning code releases assume a flat file format to demonstrate the usage of a new network architecture. The most common format is a whitespace or tabulator separation of text and labels. This format is quite popular and wide-spread as it allows a rather easy transformation of the textual data into an integer representation. Thus, one

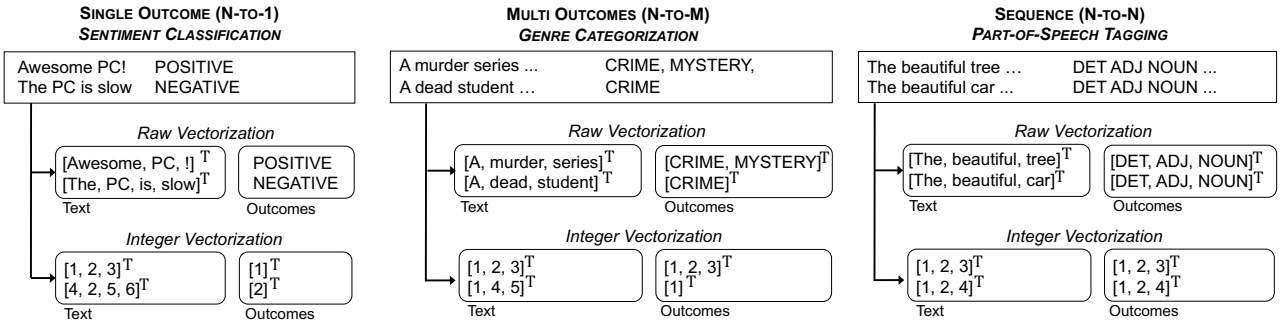


Figure 2: Vectorization N-to-1, N-to-M and N-to-N

must first transform his data into this flat file format before the code can be executed. In case of more complex data formats, such as XML, this leads to considerable additional effort. This challenge is solved in DKPro TC by the many data format readers included in DKPro Core. The seamless integration of DKPro Core allows it to read a large variety of data formats and access information such as lemmas, part-of-speech tags, etc. from DKPro TC. This enables a quick and easy exchange of corpora and data formats. Of course, own readers for highly specific data formats can be easily written, too.

3.2. Vectorization

All textual information has to be transformed into a numerical vector representation before it can be provided to the deep learning framework. This vectorization entails mapping words and labels to integer values. When applying a prototype to unlabelled plain text, the integer values have to be mapped back to their original label to obtain human interpretable results. This is a mandatory task that can be easily automatized. While the general task of vectorization appears straightforward, its details depend on the kind of classification task of which we distinguish the three variants shown in Figure 2:

Single Outcome (N-to-1): In this setting, a single outcome has to be predicted for a text document with N tokens. In classification the outcomes are labels, in case of regression they are numeric values. Use cases for single outcome classification are e.g. sentiment analysis or scoring the reading difficulty of a text (regression).

Multi-Outcome (N-to-M): For a text document with N tokens, M outcomes have to be predicted. For instance, categorization of books into genres, where a single book might have more than just one genre.

Sequence (N-to-N): For a text document with N tokens, an equal amount of N labels has to be predicted. The sequence in which the tokens occur is furthermore informative for predicting the labels. A prominent example is part-of-speech tagging.

Implementation The user is given control as to whether a vector is created with textual information (raw vectorization) or if the words have already been mapped to an integer representation (integer vectorization). Integer vectorization fits most setups and leads to further reduction of user-specific preprocessing code as the mapping process is done automatically by DeepTC. If the network architecture also considers sub-word information, e.g. character- or byte-level information, integer vectorization would be premature as the networks requires access to the actual word forms. For such cases, raw vectorization allows providing the actual words to the deep learning framework. As trade-off, the deep learning code has then to take care of mapping the raw data to an integer representation. This allows DeepTC to be flexible for more complex tasks, but still provide convenience features for common NLP setups.

3.3. Word Embeddings

It is common to use pre-trained word embeddings, which are often quite large with negative effects on the start-up time of experiments. As a consequence, embeddings are usually pruned to contain only words that occur in the vocabulary. Furthermore, in some tasks, words without pre-trained embedding are either dropped or vectors are randomly initialized instead.

Implementation We provide a processing step in which the word embedding is pruned to contain only the occurring vocabulary. The user is given control as to whether words missing in the embeddings are removed or shall be initialized with a random vector. In case no word embedding is provided, this step performs no operation.

3.4. Interface to Deep Learning Framework

The prepared data is provided to the deep learning framework. All necessary files are written to disk and the framework code is executed. The file locations are passed as parameters to the framework code. The framework code is expected to create a file at a specified location which contains the results of the execution.

Implementation An integration of third-party frameworks often leads to challenges how to interface with these frameworks. There might be breaks between programming environments, for instance operating Python frameworks from Java, but also breaks between the data representation in DeepTC and the data format that is expected by a framework. The break between programming environments, i.e. Java to Python to Java, are tackled by defining a protocol of data exchange. For each of the three defined classification tasks, i.e. single outcome, multi-outcome, and sequence classification, a data format is expected in which the framework code provides the predicted outcomes. This allows bridging to deep learning frameworks based on non-Java technologies. The break between data formats is solved by

the vectorization processing step which writes the data to disc. As non-Java frameworks work internally with their own data structures, the framework code then can read this data and wrap the vectorized data into the respective data format. This leads to a minimal amount of data conversion overhead that has to take place in the framework code. In case of Keras, for instance, which is based on Python, the vectorized data has to be transformed into the NumPy data type. This defines a simple way of interfacing between different data formats and deep learning environments.

3.5. Software Versions

An important challenge to reproducibility is keeping track of the software versions that are being used for running an experiment. Many deep learning frameworks are still under rapid development and, thus, change quickly with bugs being fixed and APIs being updated. If code is released, it is often not reported which software version was used.

For instance for Keras, which depends on a backend such as TensorFlow, we record not just the Keras version but also the version of the backend and the NumPy library as primary data structure. The software version that is recorded is highly dependent on the respective deep learning framework. This provides a basic software versioning record, which can be released with the experimental code.

4. Limitations

The rapid development of deep learning software creates practical limitations to reproducibility and convenience.

The convenience of automatically installing needed components is easily provided for Java/Maven-based software. For non-Java frameworks, this is not as easily possible and the task of installing software is delegated to the user. We would require a method to serialize the deep learning framework environment into a container that would allow deployment on a third-party computer, i.e. in the case of Keras, which would also entail the respective backend and their dependencies. A common strategy is to use virtualization software such as Docker (Merkel, 2014). A virtualization container is created that capsules a software environment, for instance Python with Keras installed, into a deployable container. At the moment, an automatically creation of such a virtualization container for an experiment is not supported by DeepTC, but one can run DeepTC within such a container if one is prepared beforehand.

A further challenge is to track the names and versions of all involved components the user has to install to reproduce an environment. While using a virtualization container as a black-box environment is convenient, it is often not clear which exact version of the required software is used in a certain setup. Furthermore, recording an entire system configuration setup would lead to an extremely long list of software components with some being more important than others to the reproducibility of results. As a compromise, we record the software versions of the key components, for instance Keras or NumPy, to create an overview of the used software versions to run an experiment.

A further limitation occurs if researchers work on unstable software versions. It is not uncommon that researchers compile their deep learning software from the latest version in a source-code repository to make certain features available, i.e. a bleeding edge version. One would have to record the exact hash-id of the source-code repository from which the software was built to enable reproducibility. Detecting such setups is beyond an automatic detection by DeepTC.

5. Proof of Concept

As proof of concept, we conduct a replication experiment with all three deep learning frameworks in parallel, and compare the results to using a shallow learning framework.[4] We attempt to reproduce the state-of-the-art results for Part-of-Speech (PoS) tagging. Our code is publicly available[5] and demonstrates the usage of DeepTC for deep learning experiments.

DeepTC configuration Figure 3 shows the code snippet for configuring DeepTC for our replication study. Configurations for other (deep learning) classification tasks follow the same structure.

The first lines define the UIMA data reader: one for the training data and one for the test data. As a suitable reader for the WSJ data is already provided by DKPro Core, we can simply us it as-is in our DeepTC experiment without any additional effort. After the readers, we define the so called *parameter space* which configures the experiment. Feature mode and learning mode define the nature of the learning task. In the case of PoS tagging, this is a sequence classification task where a single label is to be predicted for each word.[6] Python installation, pre-trained word embeddings, and user code point to locations in the user's file system. The Python path points to the installed Python version that shall be used to execute the framework code, i.e. the one for which the deep learning framework is installed. In the case of DL4J, which is Java, this parameter is not necessary. Word embeddings points to the file location of the pre-trained word embeddings, this parameter is optional and can be omitted if no pre-trained embeddings are necessary for an experiment. The variable `userCode` points to a file which contains the framework code. The seed value is passed through to the user code to initialize the random generator in the respective framework with the provided value[7]. Integer vectorization is set to `true` to enable the automatically mapping of words to integer values.

This parameter space is provided to a train-test experiment object (an alternative would be cross-validation), which is then executed by the underlying DKPro Lab environment. Changing the path of the user code and the machine learning adapter allows switching between the deep learning frameworks.

[4] All used frameworks are part of DKPro TC, which makes it easy to implement such comparison between multiple classifiers

[5] `https://github.com/Horsmann/LREC2018-DeepTC`

[6] Other configurations would allow, for instance, a multi-label classification on full documents rather than classifying single words in sentences. See Figure 2 which defines the different vectorization modes for different classification tasks.

[7] This ensures reproducibility by using a fixed seed value for initialization, which leads to the same random numbers being generated between several executions of the framework code

```
CollectionReaderDescription trainReader = createReaderDescription(
        PennTreebankChunkedReader.class,
        PennTreebankChunkedReader.PARAM_LANGUAGE, "en",
        PennTreebankChunkedReader.PARAM_SOURCE_LOCATION, train,
        PennTreebankChunkedReader.PARAM_PATTERNS, "**/*.pos");

CollectionReaderDescription testReader = createReaderDescription(
        PennTreebankChunkedReader.class,
        PennTreebankChunkedReader.PARAM_LANGUAGE, "en",
        PennTreebankChunkedReader.PARAM_SOURCE_LOCATION, test,
        PennTreebankChunkedReader.PARAM_PATTERNS, "**/*.pos");

Map<String, Object> dimReaders = new HashMap<String, Object>();
dimReaders.put(DIM_READER_TRAIN, trainReader);
dimReaders.put(DIM_READER_TEST, testReader);

ParameterSpace pSpace = new ParameterSpace(
        Dimension.createBundle("readers", dimReaders),
        Dimension.create(DIM_FEATURE_MODE, Constants.FM_SEQUENCE),
        Dimension.create(DIM_LEARNING_MODE, Constants.LM_SINGLE_LABEL),
        Dimension.create(DIM_PYTHON_INSTALLATION, python3),
        Dimension.create(DIM_SEED_VALUE, 12345),
        Dimension.create(DIM_PRETRAINED_EMBEDDINGS, embedding),
        Dimension.create(DIM_VECTORIZE_TO_INTEGER, true),
        Dimension.create(DIM_USER_CODE, userCode));

DeepLearningExperimentTrainTest exp =
        new DeepLearningExperimentTrainTest("MyExperiment", KerasAdapter.class);

exp.setParameterSpace(pSpace);
exp.setPreprocessing(createEngineDescription(SequenceOutcomeAnnotator.class));
exp.addReport(BatchTrainTestReport.class);

Lab.getInstance().run(exp);
```

Figure 3: Configuration of a DeepTC experiment

Experimental setup As training data, we use Wall-Street-Journal (WSJ) (Marcus et al., 1993) corpus sections 0-18 and test on sections 22-24, which is the usual evaluation data split of this corpus (Collins, 2002). For each deep learning framework, we implement a plain, bidirectional Long-Short-Term-Memory (LSTM) network (Hochreiter and Schmidhuber, 1997; Graves et al., 2005). As the purpose of the experiment is to show that we can execute arbitrary network code within DeepTC and not to obtain the highest possible results, we use 'default' parameter choices (as much as there is already something like a default in the field). Our bi-LSTM uses 100 hidden units, the output layer applies a softmax function, and we use cross-entropy as loss function during model training. We use the 64-dimensional Wikipedia word embeddings by Al-Rfou et al. (2013). We train 20 epochs with a learning rate of 0.1 using statistical gradient decent.

We compare the deep learning results to the results of a shallow learning PoS tagger that we also implement with DKPro TC. We use Conditional Random Field (CRF) (Lafferty et al., 2001) for implementing this shallow classifier, which is based on the implementation by Okazaki (2007) that is provided in DKPro TC. We use a minimalistic feature set of a tri-gram word window as local word context and provide the cluster-id of a the word in focus if it is contained in a Brown (Brown et al., 1992) word cluster, which was created from 100 million tokens of Twitter messages. The information obtained by Brown clustering is comparable to the information contained in the word embeddings that are used in the neural networks, i.e. both encode distributional knowledge. It is common to also use character ngrams, which we excluded in this case to sustain comparability to the neural network setup, which only use word-level information.

Results Table 1 shows that the different classifiers reach comparable results. Keras and DyNet reach the same result, which is not surprising as they both use Python and the NumPy library. The Java based Deeplearning4J gives slightly lower results. As we are using exactly the same setup and configuration, this is already a finding which could not have been easily achieved without DeepTC. Furthermore, the neural network results are competitive to the 96.5% by Brants (2000) and the 97.6% by Choi (2016).

DeepTC allowed us to avoid a large part of the repetitive work, and limited the manual effort for writing framework-specific code to defining the network architecture and few data-type wrapping method calls. Furthermore, the created experiments are immediately shareable with other re-

	Framework	Acc (%)
DeepTC	Deeplearning4j	95.8
	DyNet	96.4
	Keras	96.4
ShallowTC	CRF	94.2

Table 1: Accuracy on WSJ sections 22-24 using shallow and deep learning classifiers in DKPro TC

searchers to allow a quick and easy replication of our experiments.

6. Related Work

There are several software projects that aim at providing (shallow) machine learning tools over a common interface, e.g. ClearTK (Ogren et al., 2008), NLTK (Bird et al., 2009), Mallet (McCallum, 2002), Scikit-learn (Pedregosa et al., 2011), or Weka (Hall et al., 2009). These projects provide building blocks for creating text classification experiments, but still require a considerable amount of programming by the user. Most similar to DKPro TC are ClearTK and Weka. ClearTK is also UIMA-based and provides a similar middle-layer for defining feature extractors and shares many machine learning tools with DKPro TC. Weka provides many classifiers that work with a Weka-specific data format. An abstraction layer that extracts certain feature values from a dataset is not provided, and the user is responsible for compiling a file in the Weka data format. None of these projects intends to provide a self-contained environment.

There are many deep learning frameworks such as Tensorflow, Theano, DyNet, DeepLearning4J, Torch (Collobert et al., 2002), or Chainer (Tokui et al., 2015) to name just a few. Software such as Keras, Lasagne (Dieleman et al., 2015) or Fuel&Blocks (van Merriënboer et al., 2015) provide a simplified, building-block like interface to an underlying, low-level deep learning framework such as Theano. Data loading capabilities are included to some extent for instance for the well-known MNIST (Lecun et al., 1998) dataset with hand-written digits for image processing tasks. Furthermore, there are approaches to analyze what a neural network actually learns when applied to image and text processing tasks (Yosinski et al., 2015; Li et al., 2016). The deep learning software, thus, provides means to build prototypes quickly, but provides no means to ensure replicability by a third-party researcher. This means that all processing components must be manually provided and configured by the researcher who wants to run a certain prototype.

Hence, the DeepTC extension fills a gap in the software landscape, which will improve reproducibility of deep learning experiments.

7. Conclusion

We presented DeepTC, a deep learning extension of the NLP experiment framework DKPro TC. We discussed the current state of DKPro TC, which is limited to shallow learning frameworks and discussed the need for a software environment that also supports reproducibility for deep learning experiments. As frequent challenges to reproduction of deep learning experiments, we identified incomplete data preparation steps, embedding preparations tasks, and the vectorization of data into an integer representation. DeepTC takes care of those steps and allows to share a self-contained experiment to improve reproducibility. DKPro TC installs necessary pre-processing tools automatically and applies all processing steps to any dataset. Furthermore, by performing the data preparation inside DKPro TC, the high code duplication of typical deep learning code is avoided, which leads to a higher code readability of the actual network code. As proof of concept, we implemented support for three deep learning frameworks: Keras, DyNet, and DeepLearning4J. In a replication experiment, we showed that this setup allows to replicate state-of-the-art result and demonstrated the usage of DeepTC.

Acknowledgement

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media".

8. Bibliographical References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed Word Representations for Multilingual NLP. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. ACL.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

Brants, T. (2000). TnT: A Statistical Part-of-speech Tagger. In *Proceedings of the Conference on Applied Natural Language Processing*, pages 224–231, Seattle, Washington. Association for Computational Linguistics.

Brown, P. F., DeSouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-Based n-gram Models of Natural Language. *Computational Linguistics*, 18:467–479.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. 2(3):1–27.

Choi, J. D. (2016). Dynamic Feature Induction: The Last Gist to the State-of-the-Art. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 271–281, San Diego, California. Association for Computational Linguistics.

Collins, M. (2002). Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1–8, Philadelphia, USA. Association for Computational Linguistics.

Collobert, R., Bengio, S., and Mariéthoz, J. (2002). Torch: a modular machine learning software library. Technical Report IDIAP-RR 02-46, IDIAP.

Daxenberger, J., Ferschke, O., Gurevych, I., and Zesch, T. (2014). DKPro TC: A Java-based Framework for Supervised Learning Experiments on Textual Data. In

Proceedings of ACL: System Demonstrations, pages 61–66, Baltimore, Maryland. Association for Computational Linguistics.

Dieleman, S., Schlüter, J., Raffel, C., Olson, E., Sønderby, S. K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J. D., Heilman, M., de Almeida, D. M., McFee, B., Weideman, H., Takács, G., de Rivaz, P., Crall, J., Sanders, G., Rasul, K., Liu, C., French, G., and Degrave, J. (2015). Lasagne: First release. http://dx.doi.org/10.5281/zenodo.27878.

Eckart de Castilho, R. and Gurevych, I. (2011). A Lightweight Framework for Reproducible Parameter Sweeping in Information Retrieval. In *Proceedings of the Workshop on Data infrastructurEs for Supporting Information Retrieval Evaluation*, pages 7–10, New York, NY, USA.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). Liblinear: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874.

Ferrucci, D. and Lally, A. (2004). UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3-4):327–348.

Graves, A., Fernández, S., and Schmidhuber, J. (2005). Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition. In *Proceedings of the 15th International Conference on Artificial Neural Networks: Formal Models and Their Applications - Volume Part II*, pages 799–804, Warsaw, Poland. Springer-Verlag.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: An Update. 11:10–18.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Joachims, T. (2008). SvmHmm: Sequence Tagging with Structural Support Vector Machines. https://www.cs.cornell.edu/People/tj/svm_light/svm_hmm.html.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, San Francisco, CA, USA.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, J., Chen, X., Hovy, E., and Jurafsky, D. (2016). Visualizing and Understanding Neural Models in NLP. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 681–691, San Diego, California, June. Association for Computational Linguistics.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

McCallum, A. K. (2002). MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.

Merkel, D. (2014). Docker: Lightweight Linux Containers for Consistent Development and Deployment. *Linux Journal*, 2014(239), March.

Neubig, G., Dyer, C., Goldberg, Y., Matthews, A., Ammar, W., Anastasopoulos, A., Ballesteros, M., Chiang, D., Clothiaux, D., Cohn, T., Duh, K., Faruqui, M., Gan, C., Garrette, D., Ji, Y., Kong, L., Kuncoro, A., Kumar, G., Malaviya, C., Michel, P., Oda, Y., Richardson, M., Saphra, N., Swayamdipta, S., and Yin, P. (2017). DyNet: The Dynamic Neural Network Toolkit - Technical Report.

Ogren, P. V., Wetzler, P. G., and Bethard, S. J. (2008). ClearTK: A UIMA Toolkit for Statistical Natural Language Processing. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Okazaki, N. (2007). CRFsuite: A fast implementation of Conditional Random Fields. http://www.chokkan.org/software/crfsuite/.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *The Journal of Machine Learning Research*, 12.

Tokui, S., Oono, K., Hido, S., and Clayton, J. (2015). Chainer: a Next-Generation Open Source Framework for Deep Learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*.

van Merriënboer, B., Bahdanau, D., Dumoulin, V., Serdyuk, D., Warde-Farley, D., Chorowski, J., and Bengio, Y. (2015). Blocks and Fuel: Frameworks for deep learning. *CoRR*, abs/1506.00619. http://arxiv.org/abs/1506.00619.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., and Lipson, H. (2015). Understanding Neural Networks Through Deep Visualization. In *Deep Learning Workshop, International Conference on Machine Learning (ICML)*.

Improving Hate Speech Detection with Deep Learning Ensembles

Steven Zimmerman, Chris Fox, Udo Kruschwitz
University of Essex
Wivenhoe Park, Colchester CO4 3SQ, United Kingdom
{szimme, foxcj, udo}@essex.ac.uk

Abstract

Hate speech has become a major issue that is currently a hot topic in the domain of social media. Simultaneously, current proposed methods to address the issue raise concerns about censorship. Broadly speaking, our research focus is the area human rights, including the development of new methods to identify and better address discrimination while protecting freedom of expression. As neural network approaches are becoming state of the art for text classification problems, an ensemble method is adapted for usage with neural networks and is presented to better classify hate speech. Our method utilizes a publicly available embedding model, which is tested against a hate speech corpus from Twitter. To confirm robustness of our results, we additionally test against a popular sentiment dataset. Given our goal, we are pleased that our method has a nearly 5 point improvement in F-measure when compared to original work on a publicly available hate speech evaluation dataset. We also note difficulties encountered with reproducibility of deep learning methods and comparison of findings from other work. Based on our experience, more details are needed in published work reliant on deep learning methods, with additional evaluation information a consideration too. This information is provided to foster discussion within the research community for future work.

Keywords: Hate Speech, Reproducibility, Text Classification

1. Introduction and Motivation

Our research is focused on the development of better methods for protection of freedom of expression in the web domain and social media while simultaneously reducing illegal discrimination. Motivation is provided by the fundamental human rights (as outlined in articles 19 and 20 of (The United Nations, 1948) and (The United Nations General Assembly, 1966)) which simultaneously provide rights to freedom of expression and prevent censorship and illegal discrimination. Automated take down approaches potentially infringe upon rights to freedom of expression, such as when a text classifier incorrectly flags a page or post as something to be taken down. Hate speech classifiers are based on annotation methods that are very difficult to define, with questionable reliability (Ross et al., 2017). Even a manual take down approach, such as that used by Facebook, is a challenging task[1].

Censorship is a potential risk when addressing these issues with automated text classification methods, thus all options should be considered (Benesch, 2017). Actions to filter and block content (e.g. recently implemented laws in Germany and by platforms such as Twitter and Facebook) deemed to be hateful and / or threatening to the online community and society as whole have been taken, which is having negative consequences[2].

The goal of our work is to discover simple but effective methods to improve upon existing research in the area of hate speech classification. These methods will be useful in our broader research which tests mechanisms that provide users with feedback about their consumption of potentially hateful material, with the intent of changing their behavior through awareness as a possible alternative to regulation.

We include an initial investigation of existing methods for classification of abusive and hateful speech in the domain of Twitter. Additionally, we look into methods from the domain of sentiment analysis, as the classification task is similarly subjective and provides a larger body of research. Our contributions are as follows.

- Experimental results for a deep learning ensemble method that improves F-measure 2% over non-ensemble approaches and a nearly 5% increase over hand crafted methods from authors of a hate speech dataset.
- We provide recommendations for future work by the research community on text classification problems such as hate speech and suggestions for researchers using deep learning approaches. The recommendations are motivated by discovery of inconsistencies in evaluation methods and a lack of detail for methods used in previous research that was reviewed for our work.

In the following sections, we provide related background work, methods of implementations, results and analysis of findings.

2. Background

While lookup of hateful terms in a dictionary is one possible approach (Tulkens et al., 2016) to filter hateful content, such methods are deemed insufficient (Saleem et al., 2016). Text classification methods demonstrate much better results.

Ensemble models have shown promising results in many areas of machine learning and other fields as well (see (Molteni et al., 1996), an example from atmospheric sciences). Ensemble methods for text classification, such as

[1] See example of task: https://www.nytimes.com/interactive/2017/10/13/technology/facebook-hate-speech-quiz.html

[2] as demonstrated by the recently implemented law in Germany https://www.economist.com/news/europe/21734410-new-social-media-law-causing-disquiet-germany-silencing-hate-speech-cannot-define-it

stacking and bagging, are commonly used approaches (Aggarwal and Zhai, 2012; Xia et al., 2011). In the area of social media, simple but effective ensemble approaches have been used for sentiment classification of Tweets (Hagen et al., 2015). Most relevant to our experiments with neural networks and Twitter data are hybrid models (Badjatiya et al., 2017; Park and Fung, 2017) which combine outputs from different neural networks.

In recent years, efficient algorithms were produced (Mikolov et al., 2013b; Mikolov et al., 2013a; Pennington et al., 2014) that have allowed the use of word embeddings as features for neural networks and other algorithms (e.g. Logistic Regression). There are multiple pre-trained word embedding models available, trained in domains such as news articles (Mikolov et al., 2013b) and Twitter (Godin et al., 2015; Pennington et al., 2014). These unsupervised methods and models have produced significant improvements in downstream supervised NLP and text classification tasks.

These new approaches have allowed for significant improvements against previous SemEval[3] message level Twitter sentiment analysis test sets (Severyn and Moschitti, 2015; Stojanovski et al., 2015; Vosoughi et al., 2016; Yang and Eisenstein, 2017). Similar improvements have been shown (Badjatiya et al., 2017; Gambäck and Sikdar, 2017; Park and Fung, 2017) using the recently published hate speech datasets (Waseem and Hovy, 2016; Waseem, 2016)[4] and note two of the three methods mentioned fail to provide a direct comparison to original findings as test sets were split in a different manner. For all methods reviewed, limited information (if any) was provided regarding network weight initialization schemes, which our experiments demonstrate as important information for reproducibility purposes. Similar concerns regarding details of neural network configurations have recently been raised in the information retrieval community as well (Fuhr, 2017). Nonetheless, use of neural networks and embedding methods is worth exploration by NLP and text mining researchers, as the work of (Badjatiya et al., 2017; Gambäck and Sikdar, 2017; Park and Fung, 2017; Severyn and Moschitti, 2015; Stojanovski et al., 2015; Vosoughi et al., 2016; Yang and Eisenstein, 2017) are just some examples demonstrating strong improvements on previous work that made use of traditional features (e.g. n-grams, part of speech tags, etc.).

3. Methods

Due to challenges encountered with our own work when tuning and replicating previous work using neural networks, such as inconsistencies with weight initialization of networks, we decided to take a different approach. Knowing that neural networks are not guaranteed to find a global minimum (Goodfellow et al., 2016), coupled with difficulties of parameter tuning of networks and having limited computational resources to perform an extensive set of configurations, we recalled research in 2015 which produced robust results for Twitter sentiment classification utilizing a simple ensemble method (Hagen et al., 2015). In their work, logistic regression was used to produce 3 models based upon a diverse set of features. The probabilistic output for each sentiment classification (positive, negative or neutral) was summed and averaged, with the highest average chosen as the winning classification, which resulted in the best performing solution for the SemEval sentiment classification task in 2015. Similar success with these methods was found with different Twitter sentiment classification tasks by (Balikas and Amini, 2016; Sygkounas et al., 2016) and (Zimmerman and Kruschwitz, 2017). Based on previous successes with this method for classification tasks in Twitter, we hypothesize that similar ensemble methods with neural networks using different weight initializations could also produce improvements for the tasks of hate speech detection in Twitter.

The ensemble model is created in the following manner. First we take soft-max results from each underlying model and sum them together. Then we average the sum of soft-max results, by dividing by the number of models (10 total in our case). With the average soft-max score of all models, the class with highest average is chosen as winning class similar to methods in previous work (Hagen et al., 2015).

We evaluate our method on two Twitter classification datasets, abusive speech (Waseem and Hovy, 2016) and SemEval 2013 sentiment analysis (Nakov et al., 2013) dataset (Table 1). For the abusive speech dataset, we initially perform experiments on an 85/15 fixed random split on dataset to determine best parameters, then run final experiments in the same manner as (Waseem and Hovy, 2016) which evaluated results with 10-fold cross validation. This choice was made to allow for consistent comparison between evaluation scores for each run of our experiments. Additionally, we build ensemble models on the SemEval training and development sentiment test sets and evaluate against the SemEval 2013 test set. We performed this additional experiment to determine if ensemble methods were robust enough to improve results for a different classification task.

	Positive	**Negative**	**Neutral**	**Total**
SemEval Train	3632	1453	4564	9649
SemEval Test	572	338	729	1539
SemEval 2013	1568	599	1630	3797

	None	**Sexism**	**Racism**	**Total**
Abuse / Hate	11535	3378	1970	16883

Table 1: Summary of datasets, totals for each class

[3]SemEval is an annual shared task event where researchers compare methods on various semantic tasks, such as sentiment analysis, sarcasm detection and word sense disambiguation

[4]These are two frequently cited hate speech datasets, but only provide limited coverage in the domain of racism and sexism. We note the limited number of test sets currently available for this important task, and note that non-English datasets are further under-resourced

3.1. Experimental Setup

For our experiments we utilize Python neural network and machine learning libraries. Specifically, Scikit-Learn (Pedregosa et al., 2011) is used to create feature representations for input to machine learning algorithms. For the neural network model training, the Keras library (Chollet, 2015) with Tensorflow (Abadi et al., 2015) back-end was initially used, but switched to Theano (Theano Development Team, 2016) back-end due to a discovery that weight initialization cannot be reproduced, as functionality is currently not available with TensorFlow and Keras. We note that many authors do not publish libraries used in their work, however the lack of reproducibility of results with a Theano and Tensorflow back-end are one important example demonstrating why this information should be included.

Preprocessing Tweets - Prior to the embedding lookup, all Tweets were preprocessed in the same manner (i.e. tokenization and normalization of text) to the original texts used to create the embedding model. The raw Tweets are passed through a Tweet tokenizer[5] assumed to produce output similar to tokenizers used by (Godin et al., 2015) to create an embedding model. Additionally, all URLs, mentions and numbers were normalized to _URL_, _MENTION_ and _NUMBER_ respectively with the case of the Tweets left unchanged per original methods used for the embedding model (Godin et al., 2015).

Feature Extraction - A benefit of convolutional neural network (CNN) classifiers and word embeddings is the ability to consume sequential tokens through concatenation of token embeddings into a matrix(Goldberg, 2016), in contrast to n-gram features which lose the notion of position in a text (aside from immediate neighboring terms for bi/tri-grams). CNN classifiers, in theory, can consume variable length documents. In practice, the choice of software library may make the task of variable length document ingestion impossible. As Python Keras was used for experiments, we found it necessary to set the number of tokens into the CNN to a fixed length. It is noted that the mean number of tokens in our datasets was 17 and 22 for hate speech and sentiment respectively. A pre-experimental comparison of 30, 50 and 70 tokens as the window length showed 50 tokens having better performance. With this setting, only 5 Tweets for all datasets had tokens cutoff. Investigation of the best window length is a consideration for future work. Each Tweet is represented as a matrix $\mathbf{T} \subset \mathbb{R}^{m \times n}$, where m = length of embedding vector and n = maximum tokens taken from Tweet. In cases where tokens in Tweets are $<$ than n, dummy embedding vectors with zeros are used. For the embedding model used, a 50 token by length 400 embedding matrix is the output.

Machine Learning Classifier - For the CNN, we consider a very minimal network inspired by previous work (Kim, 2014). The convolution layer has a single 3 token window and 150 filters. Padding is set to 'same', thus the input and output of convolution layer match in length. The output of the convolution layer is fed into a global max-pooling layer for feature reduction. The max pooling layer feeds into a single hidden layer with 250 units. Glorot uniform distribution is used for weight initialization, which is the default for Keras, with fixed seed settings for reproducibility[6]. No regularization is used for the abusive speech dataset, however a dropout rate of 0.2 is applied after the max pooling layer for the SemEval dataset. Beyond pooling and dropout layer are the hidden (250 nodes with ReLu activation) and output (3 nodes with sigmoid activation). The weights are learned with a binary cross-entropy loss function and the adam optimizer.

Evaluation settings - For comparison of the SemEval and abusive speech datasets, we evaluate the configuration with 3 different seed weight initializations chosen arbitrarily. A pre-experiment investigation into parameters demonstrated that improvements in model accuracy generally leveled off around 10 epochs, with small gains and reductions in evaluation metrics for epochs beyond this value, thus we focused on 3 epoch settings (3, 5 and 10) not exceeding 10. Batch size had degrading effects on accuracy and time for model convergence as it was increased, notably beyond 100, with similar effects below 10. As such, we chose 4 batch size values within the range of 10 - 100 (10, 25, 50, 100). Resources were a limiting factor to perform a more detailed parameter search within these ranges.

We use the best settings (10 epochs and batch size 10) and run 10-fold cross validation on our method to allow direct comparison with the findings of (Waseem and Hovy, 2016) (see cross validation results of these settings in Tables 4 and 5). For comparison of findings on the SemEval dataset, we use the F-1 average score for positive and negative classifications as was done in the original competition.

4. Results

Results abusive speech test set - We review results for multiple ensemble models with variations in seed parameters, number of epochs and batch size. Table 2 provides a summary of results for the 85/15 split set. In all cases, the ensemble performs better when combining sub-models, with an average of 1.97% gain on F-1. Using the best epoch and batch size settings from the 85/15 split, we ran the ensemble with 10-fold cross validation to directly compare findings with (Waseem and Hovy, 2016). In Table 4 the flattened version of confusion matrices is provided for all 10 ensemble folds, which is useful for researchers that may wish to compare their work using different evaluation metrics (e.g. unweighted F-measure). Finally, Table 5 provides a direct comparison between the mean weighted macro F-1 measure for 10-fold model run with our ensemble method with the results from (Waseem and Hovy, 2016).

To confirm significance of findings, we produce 99% confidence intervals on each set of sub-models used to produce ensemble (10 sub-models for each ensemble) and find only 2 sub-models of all 100 sub-models performs above confidence. Thus, we conclude that with 99% confidence, our ensemble method will perform better than an individual model 98% of the time.

Results SemEval 2013 test set - Analysis and review of the results, in Table 3 further demonstrate the robustness

[5]Python NLTK Tweet Tokenizer was used

[6]Note to other researchers, at time of writing, fixed seed functionality is not available when using Tensorflow back-end

mean of sub-models					
	10	**25**	**50**	**100**	**Grand Total**
3	**75.98%**	75.71%	75.46%	75.53%	75.67%
5	75.11%	75.08%	75.00%	75.24%	75.11%
10	74.88%	74.61%	74.91%	75.01%	74.85%
Grand Total	75.32%	75.14%	75.12%	75.26%	75.21%

std deviation of sub-models					
	10	**25**	**50**	**100**	**Grand Total**
3	0.95%	1.26%	1.23%	1.19%	1.16%
5	1.16%	1.28%	**0.94%**	1.15%	1.13%
10	1.02%	1.31%	1.16%	0.98%	1.11%
Grand Total	1.04%	1.28%	1.11%	1.11%	1.14%

mean of ensembles					
	10	**25**	**50**	**100**	**Grand Total**
3	77.47%	77.29%	77.21%	76.85%	77.21%
5	77.61%	77.29%	76.79%	76.74%	77.11%
10	**77.83%**	77.39%	76.85%	76.88%	77.24%
Grand Total	77.63%	77.33%	76.95%	76.83%	77.18%

ensemble (average improvement) over sub-model mean					
	10	**25**	**50**	**100**	**Grand Total**
3	1.48%	1.58%	1.75%	1.32%	1.53%
5	2.49%	2.21%	1.79%	1.50%	2.00%
10	**2.95%**	2.78%	1.95%	1.87%	2.39%
Grand Total	2.31%	2.19%	1.83%	1.57%	1.97%

std deviation of ensembles					
	10	**25**	**50**	**100**	**Grand Total**
3	0.58%	0.28%	0.38%	0.27%	0.41%
5	0.40%	0.25%	0.27%	0.27%	0.46%
10	**0.12%**	0.65%	**0.12%**	0.42%	0.54%
Grand Total	0.39%	0.38%	0.31%	0.29%	0.46%

Table 2: Summary metrics for abusive speech ensembles and sub-models - Provided here are summary metrics (evaluation was based on average F-1 measure of positive and negative classifications) based on batch size and epochs, there were 3 ensembles produced (each with different weight initializations) for each batch size (10, 25, 50 and 100) and epoch (3, 5 and 10) setting, with best highlighted. The standard deviation of ensemble models is reduced from 0.94% for individual model approach to 0.12% for ensemble approach, signaling a strong reduction in variability. We also note a nearly 2 point gain in F-1 score when comparing the mean of all ensembles to mean of sub-models.

of our ensemble method of joining soft-max results from 10 sub-models to produce final classification, with similar improvements. When considering all sentiment models ensembles compared to individual models, there is an average of 1.84% gain on F-1. We note that our best ensemble model tied the results (F-1 of 71.91) of a computationally complex social network approach produced by (Yang and Eisenstein, 2017).

Impressively, the simple method, when run on both datasets produces an increase of nearly 2% on the evaluation metric. Furthermore, in evaluation of test sets we note the standard deviation is reduced by more than half for the ensemble method, signaling a strong reduction in variability.

The following questions provided guidance for our investigation and results. These were addressed with descriptive statistics and direct comparison. Brief summaries of findings are provided for each question.

- **RQ 1:** *Based on experience with weightings and inconsistent results, how much variability in evaluation metrics is found between models with different weight initializations?* Standard deviation is the chosen metric for variability, which is provided in Tables 2 and 5. Variability for individual model approach with best parameters is found to be +/- 0.94% of the median F-1 measure. For the ensemble approach, standard deviation is found to be +/- 0.12% of the median F-1 measure and also improves nearly 2% over best individual model.

- **RQ 2:** *Given a set of N models with varying weight initializations, can an ensemble of the N models produce better results by taking the average of their soft-*

mean of sub-models					
	10	**25**	**50**	**100**	**Grand Total**
3	68.44%	**68.65%**	68.39%	68.01%	68.37%
5	68.34%	67.90%	68.24%	68.03%	68.13%
10	66.41%	67.13%	67.16%	67.06%	66.94%
Grand Total	67.73%	67.89%	67.93%	67.70%	67.81%

std deviation of sub-models					
	10	**25**	**50**	**100**	**Grand Total**
3	2.72%	2.24%	2.17%	2.62%	2.44%
5	1.58%	1.61%	1.78%	2.05%	1.76%
10	1.87%	**1.38%**	1.73%	2.11%	1.77%
Grand Total	2.06%	1.74%	1.90%	2.26%	1.99%

mean of ensembles					
	10	**25**	**50**	**100**	**Grand Total**
3	70.34%	70.16%	69.34%	68.33%	69.54%
5	70.67%	69.74%	**70.36%**	69.46%	70.06%
10	69.17%	69.49%	69.67%	69.08%	69.35%
Grand Total	70.06%	69.79%	69.79%	68.96%	69.65%

ensemble (average improvement) over sub-model mean					
	10	**25**	**50**	**100**	**Grand Total**
3	1.90%	1.51%	0.95%	0.33%	1.17%
5	2.33%	1.84%	2.12%	1.42%	1.93%
10	**2.76%**	2.36%	2.51%	2.02%	2.41%
Grand Total	2.33%	1.90%	1.86%	1.26%	1.84%

std deviation of ensembles					
	10	**25**	**50**	**100**	**Grand Total**
3	0.22%	1.17%	**0.15%**	0.59%	1.01%
5	1.10%	0.55%	0.61%	0.62%	0.82%
10	0.50%	0.65%	0.31%	0.68%	0.54%
Grand Total	0.92%	0.78%	0.57%	0.74%	0.84%

Table 3: Summary metrics for ensembles and sub-models evaluated on the SemEval 2013 dataset - Provided here are summary metrics (evaluation was based on average F-1 measure of positive and negative classifications) based on batch size and epochs, there were 3 ensembles produced (each with different weight initializations) for each batch size (10, 25, 50 and 100) and epoch (3, 5 and 10) setting, with best and worst highlighted (best in **bold**). Overall the standard deviation of ensemble models is reduced to +/- 0.15%, a sharp reduction from standard deviation of individual models and a signal for reduction in variance. Similar to hate speech evaluations, we note a nearly 2 point gain in F-1 score when comparing the mean of all ensembles to mean of sub-models.

max predictions? We have set N = 10 in our experiment and are 99% confident that our ensemble approach will significantly improve F-1 scores 98% of the time compared to results from a single model.

- **RQ 3:** *With all model initialization parameters fixed, how do variations in batch size and number of epochs impact ensemble results?* We answer this question with relative improvements in F-1 scores between mean of individual models and mean of ensemble models. As shown in Table 2 and discussed in Section 4., the greatest improvements are made with smaller batch sizes and larger number of epochs. Variability, as measured by standard deviation, consistently reduces for all parameters.

- **RQ 4:** *How do methods compare with different classification tasks (e.g. Abusive speech vs. Sentiment)?* As outlined in results Section 4 and Table 3, the methods produce similar results when run on a sentiment analysis test set.

5. Discussion and Conclusions: Learnings from Experiments

We have demonstrated the usefulness of ensemble methods with a neural network configuration. We have shown that weight initialization methods are an important factor to consider in any research using deep learning. We demonstrated that a simple ensemble method for neural networks has statistically significant improvement over a single model. Furthermore, we have shown that individ-

True	None	None	None	Race	Race	Race	Sex	Sex	Sex
Pred	None	Race	Sex	None	Race	Sex	None	Race	Sex
Fold 1	1053	42	59	66	129	2	113	0	225
Fold 2	1054	52	48	48	148	1	122	1	215
Fold 3	1053	40	61	50	143	4	106	1	231
Fold 4	1066	29	59	58	136	3	106	0	232
Fold 5	1040	38	76	45	152	0	94	0	244
Fold 6	1032	46	75	49	147	1	106	4	228
Fold 7	1055	40	58	57	138	2	113	2	223
Fold 8	1055	37	61	55	142	0	122	1	215
Fold 9	1046	47	60	51	143	3	100	2	235
Fold 10	1064	44	45	50	147	0	113	1	223

Table 4: Confusion scores for all 10-fold ensembles (best in grey) on the (Waseem and Hovy, 2016) dataset. Gold standard and predicted classifications for the dataset are Sex = sexism, Race = racism and None = neither racism nor sexism.

mean of sub-models	75.65%
std deviation of sub-models	1.54%
mean of ensembles	78.62%
ensemble improvement on sub-model mean	2.96%
std deviation of ensembles	1.08%
best results from original author	73.93%
improvement on original work	4.69%

Table 5: Comparison of ensemble method on (Waseem and Hovy, 2016) dataset vs. results from original best method (Waseem and Hovy, 2016). Values are based upon F-1 Measure.

ual models have high variance when compared to the variance of ensemble models. Thus, one might place lower confidence in their model when an ensemble approach is not used. Also, in all trials, we find that ensemble models perform better on test sets compared to the mean of sub-models. The ensemble approach appears to leverage the high variance as an advantage for final classification via the simple method of averaging soft-max output.

5.1. Difficulties encountered

Several difficulties were encountered in our initial experiments due to weight initializations often not being reported by other authors coupled with the issue of a deep learning library lacking reproducibility due to seed setting. In our case, we had originally used Keras with a TensorFlow back-end. Post experimentation, we investigated this matter more and found that the issue with reproducibility of weight initializations is resolved with use of a Theano back-end. Nonetheless, this painful experience not only demonstrates the need to publish more details, it also can lead to better solutions, such as a more robust ensemble approach.

5.2. A request for future research

We note that in many papers reviewed for our work, researchers failed to publish their weight initialization methods. There are many choices available for weight initialization of a neural network and it is one of many important factors. Deep learning has many other considerations too, and the details provided in published work are frequently light in detail. When considering all of the parameters available (e.g. number of layers, embedding options, optimizers, weighting schemes, activation functions, libraries, etc.), neural networks can become very complex and therefore more details should be recorded for reproducibility. As our work demonstrates, seemingly innocuous values such as batch size, can have significant impacts on results. Filling in the missing details from published work is a time consuming task, which is best resolved through communication with original authors that may no longer be available due to various factors. As such, it may be worthwhile to make every effort to include all parameter choices, including weight initialization methods, in future work[7].
Additionally, a set of confusion matrices was provided in previous work on the abusive dataset (Gambäck and Sikdar, 2017). We have also provided confusion matrix results in Table 4. This information is useful for reproducibility, as you can compare many more evaluation metrics than the popular single aggregate measure F-1 macro weighted score. Reporting of confusion matrices opens the door to other metrics such as F-1 micro unweighted or F measure with different beta values. This information could easily be provided online, as publications often have space limitation, therefore it is worth consideration of a better approach.

5.3. Future work

Future work would consider evaluation of ensemble methods on additional test sets (e.g. SemEval 2014 and 2015 for example). Also, a comparison of different weighting schemes is likely useful to understand variations within this parameter. Beyond that, building models with different network configurations and embedding models are all considered to be natural next steps. Different approaches, such as LSTM networks based on character representations (as opposed to word embeddings) should be considered. Reproducing the promising results using LSTM and Gradient

[7]We made our code publicly available for other researchers, which can be found at `https://github.com/stevenzim/lrec-2018`.

Boosted Decision Trees (Badjatiya et al., 2017) on additional datasets is a worthwhile exercise too. Given knowledge that neural network performance improves as datasets become larger, it would be an interesting experiment to gain insight as to what amount of data is sufficient enough where ensemble methods do not provide a boost in performance. Therefore one possible next step for our work would be to try our methods on progressively larger datasets to empirically show that ensembles provide smaller improvements as training data increases.

6. Acknowledgments

This work was supported by the Economic and Social Research Council grant number ES/M010236/1.

7. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Aggarwal, C. C. and Zhai, C. (2012). A survey of text classification algorithms. *Mining text data*, pages 163–222.

Badjatiya, P., Gupta, S., Gupta, M., and Varma, V. (2017). Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760. International World Wide Web Conferences Steering Committee.

Balikas, G. and Amini, M.-R. (2016). Twise at semeval-2016 task 4: Twitter sentiment classification. *Proceedings of SemEval*, pages 85–91.

Benesch, S. (2017). Civil society puts a hand on the wheel: Diverse responses to harmful speech. *Harmful Speech Online*, page 31.

Chollet, F. (2015). Keras. `https://github.com/fchollet/keras`.

Fuhr, N. (2017). Some common mistakes in ir evaluation, and how they can be avoided. In *ACM SIGIR Forum*, volume 51, pages 32–41. ACM.

Gambäck, B. and Sikdar, U. K. (2017). Using convolutional neural networks to classify hate-speech. In *Proceedings of The First Workshop on Abusive Language Online*, pages 85–90. Association for Computational Linguistics.

Godin, F., Vandersmissen, B., De Neve, W., and Van de Walle, R. (2015). Multimedia lab @ acl w-nut ner shared task: Named entity recognition for twitter microposts using distributed word representations. In *Proceedings of the ACL 2015 Workshop on Noisy User-generated Text (ACL-IJCNLP)*, volume 2015, pages 146–153.

Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research(JAIR)*, 57:345–420.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Hagen, M., Potthast, M., Büchner, M., and Stein, B. (2015). Webis: An ensemble for twitter sentiment detection. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 582–589.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems(NIPS)*, pages 3111–3119.

Molteni, F., Buizza, R., Palmer, T. N., and Petroliagis, T. (1996). The ecmwf ensemble prediction system: Methodology and validation. *Quarterly journal of the royal meteorological society*, 122(529):73–119.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., and Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320.

Park, J. H. and Fung, P. (2017). One-step and two-step classification for abusive language detection on twitter. In *Proceedings of The First Workshop on Abusive Language Online*, pages 41–45. Association for Computational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Ross, B., Rist, M., Carbonell, G., Cabrera, B., Kurowsky, N., and Wojatzki, M. (2017). Measuring the reliability of hate speech annotations: The case of the european refugee crisis. *preprint arXiv:1701.08118*.

Saleem, H. M., Dillon, K. P., Benesch, S., and Ruths, D. (2016). A web of hate: Tackling hateful speech in online social spaces. In *Proceedings of the Workshop on Text Analytics for Cybersecurity and Online Safety (TA-COS 2016)*, LREC 2016, pages 1–10.

Severyn, A. and Moschitti, A. (2015). Twitter sentiment

analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 959–962. ACM.

Stojanovski, D., Strezoski, G., Madjarov, G., and Dimitrovski, I. (2015). Twitter sentiment analysis using deep convolutional neural network. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 726–737. Springer.

Sygkounas, E., Rizzo, G., and Troncy, R. (2016). Sentiment polarity detection from amazon reviews: an experimental study. In *Semantic Web Evaluation Challenge*, pages 108–120. Springer.

The United Nations General Assembly. (1966). International covenant on civil and political rights. *Treaty Series*, 999:171, December.

The United Nations. (1948). *Universal Declaration of Human Rights*. December.

Theano Development Team. (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May.

Tulkens, S., Hilte, L., Lodewyckx, E., Verhoeven, B., and Daelemans, W. (2016). A dictionary-based approach to racism detection in dutch social media. In *Proceedings of the Workshop on Text Analytics for Cybersecurity and Online Safety (TA-COS 2016)*, LREC 2016, pages 11–17.

Vosoughi, S., Vijayaraghavan, P., and Roy, D. (2016). Tweet2vec: Learning tweet embeddings using character-level cnn-lstm encoder-decoder. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 1041–1044. ACM.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of NAACL-HLT*, pages 88–93.

Waseem, Z. (2016). Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the 1st Workshop on Natural Language Processing and Computational Social Science*, pages 138–142.

Xia, R., Zong, C., and Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, 181(6):1138–1152.

Yang, Y. and Eisenstein, J. (2017). Overcoming language variation in sentiment analysis with social attention. *Transaction of the Association for Computational Linguistics (TACL)*.

Zimmerman, S. and Kruschwitz, U. (2017). Speaking of the weather: Detection of meteorological influences on sentiment within social media. In *Computer Science and Electronic Engineering (CEEC), 2017*, pages 1–6. IEEE.

Distributional Term Set Expansion

Amaru Cuba Gyllensten, Magnus Sahlgren
RISE SICS
Box 1263, SE-164 29 Kista
{amaru.cuba.gyllensten, magnus.sahlgren}@ri.se

Abstract

This paper is a short empirical study of the performance of centrality and classification based iterative term set expansion methods for distributional semantic models. Iterative term set expansion is an interactive process using distributional semantics models where a user labels terms as belonging to some sought after term set, and a system uses this labeling to supply the user with new, candidate, terms to label, trying to maximize the number of positive examples found. While centrality based methods have a long history in term set expansion (Sarmento et al., 2007; Pantel et al., 2009), we compare them to classification methods based on the the Simple Margin method, an Active Learning approach to classification using Support Vector Machines (Tong and Koller, 2002). Examining the performance of various centrality and classification based methods for a variety of distributional models over five different term sets, we can show that active learning based methods consistently outperform centrality based methods.

Keywords: Term Set Expansion, Lexicon Acquisition, Distributional Semantics, Word Embeddings, Active Learning

1. Introduction

One of the most commonly used resources in Natural Language Processing is the *term set*: a set of, optionally, labeled words. It is a standard approach to sentiment, subjectivity, and stance detection: compile lists of terms representing the categories in question, and then calculate the occurrence of these terms in data. A text is assigned to the category whose terms are most prevalent in the text. This approach – often referred to as *lexicon-based classification* – is simplistic, but surprisingly powerful, and often provides useful results in the absence of supervised classifiers (Eisenstein, 2017). Another closely related use case is topic monitoring in social media, in which case the frequency of topic-related terms over time can be used to gauge public interest in those topics.

The performance of such lexicon-based approaches obviously depends on the quality of the lexicon being used. A common approach is to use distributional models (word embeddings) to populate the lexicon on the basis of a small set of manually selected seed terms (e.g. "bad" and "subpar" as seed terms for negative sentiment, and "good" and "ace" as seed terms for positive sentiment). The seed terms are used as probes into the distributional model with the goal of finding other terms that are (distributionally) similar to the seed terms. Iterative term set expansion is the iterated, interactive, version of this procedure: An annotator defines an initial, incomplete, term set. This term set is fed to the term set expansion method, generating new candidate terms. The annotator labels these as belonging to, or not belonging to, the term set, which is updated accordingly. The new updated term set is then fed to the expansion method, and the process is repeated indefinitely. In this way, a small set of manually defined seed terms can be (semi-) automatically expanded into a potentially very large lexicon.

Expanding term sets using distributional models usually amounts to computing the similarity between all terms in the model and the seed terms, and then including the candidates that are most similar to the seed terms. This may seem like a well-defined process, but the quantification of similarity can be done in many different ways, and the choice of similarity function will have a significant impact on the quality of the resulting lexicon. To the best of our knowledge, there are no published comparisons between different ways of expanding term sets using distributional models, and consequently, we still lack a best practice for distributional term set expansion.

This paper aims to fill this void. In the following sections, we compare a number of standard approaches for iterative distributional term set expansion, with the aim of identifying a best practice for using distributional models to expand term sets. In doing so, we provide answers to the following questions: which methods are commonly used for term set expansion using distributional models? What are the performance differences between these methods? Is any of the methods more suitable to use for specific distributional models? And finally, is any method superior in general (and could consequently be described as a best practice)?

2. Distributional models

The quality of a distributionally-derived lexicon for classification purposes also depends on the choice of distributional model. We include the standard types of distributional models, which are detailed in the following sections, in our experiments.

2.1. (Weighted) Count models

The simplest distributional models are count-based models: for some notion of target and context items one counts the number of times the context item co-occurred with each target item. These models can be extended with weighting schemes to better fit the problem at hand. The most widely used and studied are variants on Pointwise Mutual Information (PMI), such as Positive PMI (PPMI), Smoothed PPMI, and Shifted PPMI (Levy et al., 2015).

2.2. Factorized count models

A common method to speed up usage and computation on distributional models is to factorize a (weighted) count matrix using truncated Singular Value Decomposition (SVD). Recently it has been shown that one can greatly improve

model performance by altering the singular values of the singular value decomposition, such as taking the square root of each singular value, or dropping them completely. In this study we have opted for the square root of the singular values, based on the results in (Levy et al., 2015).

2.3. Prediction models

The two prediction based models used are SkipGram with Negative Sampling (SGNS), and Continuous Bag Of Words (CBOW) (Mikolov et al., 2013). SGNS strives to predict whether an observation (consisting of a target word and a context word among the surrounding words) came from the data or was sampled from a distribution of negative examples. The objective of CBOW is the same, but instead of predicting each target-context pair, CBOW averages over all context items for the given observation.

2.4. Model choice

Ultimately, the models chosen were Factorized PPMI, Factorized Smoothed PPMI, SGNS, and CBOW, all with a window size of 2, and dimension 200. All models were trained on text data from the British National Corpus (Clear, 1993).

3. Iterative Term Set Expansion Methods

Iterative Term Set Expansion is the method of iteratively, with user input, expanding a term set. In this paper we have formalized it in the following way:
Given a labeling function $label$: Term $\rightarrow$ Label[1] (which would be a human annotator), an expansion method $expand$: [Term $\times$ Label] $\rightarrow$ [Term], and a set of already labeled terms L_t : [Term $\times$ Label], the labeled terms L_t are fed to the expansion method $expand$, which gives a set of new candidate terms to be labeled by the labeling function (or human annotator) $label$, resulting in a larger, and hopefully more informative, set of labeled terms L_{t+1}. If L_0 is the initial term set, L_i is the result after i expansion-labeling steps.[2]

$$L_{t+1} = L_t \cup \{(x, label(x)) | x \in expand(L_t)\} \quad (1)$$

Methods used to find candidate terms (the $expand$ method in Equation (1)) can be characterized as either *centrality* based or *classification* based. Centrality based methods work by constructing a representation of the term set within the distributional model. In essence constructing a synthetic, *central*, proxy term, whose neighborhood is taken to be representative of the whole term set. Centrality based methods have the advantage that one iteration of the term set expansion has complexity proportional to computing the central representation and performing a neighborhood query in the distributional model. Both of which are usually very quick operations with even more potential speed up if the centrality computation can be done in a streaming and/or parallel fashion. Classification based methods work instead by constructing a classifier based on the term set, i.e., a function from the distributional model's underlying space to some measure of belonging to the given term set. As such they are a superset of centrality based measures, where the measure of belonging to the term set is the similarity to its central representation.

3.1. Centrality based methods

The most intuitive centrality based method is the **centroid expansion method**: given a term set, its central representation is the average of all term vectors in the set:

$$\text{centroid}(T) = \bar{T} = \frac{1}{|T|} \sum_{t \in T} t. \quad (2)$$

Apart from being an intuitive and familiar notion of centrality, it also has the property that similarity to the centroid $\bar{T}$ is equivalent to the average similarity to terms in T, if similarity is an inner product on the vector space.
As a slight modification to the centroid method we introduce a simple **Signal-to-Noise ratio** centroid: the central representation of a term set is the average of all term vectors divided by empirical standard deviation:

$$\text{snr}(T) = \bar{T} \sqrt{\frac{1}{|T| - 1} \sum_{t \in T} (t - \bar{T})^2}. \quad (3)$$

The intuition is to scale down the importance of noisy dimensions and scale up the importance of dimensions where there is less noise.
Eigencentrality is a centrality measure usually associated with graphs, most famously used by Pagerank to rank importance of webpages (Page et al., 1998). Given an adjaceny matrix A, the eigencentrality is given by the eigenvector of A with the largest eigenvalue. Here, we compute the eigencentrality $W^T W$, where W is the Term $\times$ Feature matrix of the term set, and use the resulting vector as the central representation of the term set. This scales up the importance of central terms in the term set, and scales down the importance of peripheral terms.
To find new candidate terms to be labeled, we have chosen to return the unlabeled terms closest to the central representation of the positive examples in the term set:

$$expand_{center}(L) = \mathrm{k} \arg\max_{t \in \text{Vocab} \setminus L} sim(t, c)$$
$$\text{where } L^+ = \text{positive examples in } L$$
$$c = center(L^+)$$

3.2. Classification based methods

Classification methods are more general than centrality based methods, and where centrality based measures lack simple means of accounting for negative examples, negative examples are crucial for a classification based approach. In a classification setting, the challenge is not about

[1] $f : a$ denotes f has type a, $a \rightarrow b$ is the type of functions from a to b, $a \times b$ denotes the types of pairs of variables a and b, and $[a]$ denotes a finite set of as, Label is, in this work, always taken to be boolean (i.e. True if the term is in the term set and False otherwise), and Term is, rather sloppily, used to refer both to the actual term and its distributional representation.

[2] This can be expanded to the case where $expand$ returns a stream of terms to be labeled, and $label$ can decide to label, skip, or demand a new expansion with the recently labeled data, but that has been left out for simplicity's sake.

leveraging information from negative examples, but how to deal with the sparsity of labeled examples. One solution to this problem is Active Learning (Olsson, 2009).

Active learning is a subfield of machine learning that incorporates the selection and labeling of data into the learning framework. In this case, the active learner is used to train a classifier based on the labeled points, *and* to suggest new data points to label such that these new data points are as informative as possible for the active learner.

In this paper we have restricted ourselves to Support Vector Machines (SVMs), since these admit simple and efficient methods for active learning. We use **RBF kernels** and **Linear kernels** for the SVM. The motivation behind using Linear kernels is their simplicity, and the fact that all centrality based methods can be subsumed by linear classifiers[3] . The motivation behind using RBF kernels is, apart from their ubiquity, the fact that they capture the local influence of the supplied examples.

For both classification methods we have used the Simple Margin method to find new candidate terms. Simple Margin uses the structure of Support Vector Machines to select informative data points. This works by choosing the k words closest to the separating hyperplane as the next ones to be labeled:

$$expand_{margin}(L) = \underset{t \in \text{Vocab} \setminus L}{\text{k arg min}} |d(t)|$$
$$\text{where } d(t) = classify(L)(t)$$

The intuition here being that these are the data points the algorithm is the most unsure of, and whose *minimum* influence on the loss function is maximal (Tong and Koller, 2002). Note that this is not designed to maximize the number of positive examples we supply the labeler, but for the labeling of candidate terms to be as informative as possible for the underlying classifier.

4. Experimental setup

What we want to find out is how informative a tool such as this could be to a human annotator when building a term set. As such we are interested in how many positive examples the expansion method supplies the labeler with per iteration. This was evaluated against a number of predefined term sets: positive and negative sentiment term sets extracted from the AFINN word list (Nielsen, 2011) (an *affective word list*), the elements term set from Pantel (Pantel et al., 2009), a color term set extracted from Wikipedia, and an ingredient term set extracted from Wikibooks cookbook. These predefined term sets are used as proxies for a human annotator: Given a term set, we construct a random initial labeled sets with five terms taken from the term set, and five terms taken at random from terms not in the term set. When the iterative term set expansion queries the annotator for a label, this label is extracted directly from the predefined term set, i.e. with a labeling function defined as in Equation 4.

[3]Using the definition of centrality based methods we've used here, and assuming that notion of distance in the distributional space is an inner product, then the resulting measure of belonging of all centrality based methods are interchangeable with a linear classifier.

	Positive examples	**Negative examples**
L_0	responsive, perfects, popular, opportunity, comforting	acropolis, bogus, contestants, tartuffe, counter-themes

$expand(L_0)$:
agreeable, supportive, adaptable, attentive, conducive, non-threatening, open-minded, receptive, self-critical, sociable

L_1	responsive, perfects, popular, opportunity, comforting, agreeable supportive	acropolis, bogus, contestants, tartuffe, counter-themes, attentive, adaptable, conducive, non-threatening, open-minded, receptive, self-critical sociable

$expand(L_1)$:
encouragement, reassuring, support, instant, invaluable np, reassurance, salutary, snp, thatcher

⋮

Figure 1: Example of the iterative term set expansion process. Starting out with an initial labeled term set L_0 consisting of five positive and five negative examples of the sought after term set, we expand L_0 to get ten candidate terms ($expand(L_0)$). Of these ten candidate terms, the annotator labels "agreeable" and "supportive" as belonging to the sought after term set, and the labeled term set is updated accordingly. The procedure is then repeated with the updated term set L_1, yielding ten new candidates which the annotator labels, and so on, until a satisfactory term set has been constructed. In this particular case, the sought after term set is the AFINN POS term set, with the "annotator" being a simple lookup as described in Equation 4. The performance of the term set expansion method would be the average number of positive examples added to the labeled term set, in this case 2.5.

$$label_D(x) = \begin{cases} Positive & , x \in D \\ Negative & , x \notin D \end{cases} \quad (4)$$

Each combination of distributional model and expansion method was evaluated by running the term set expansion procedure for twenty steps, querying the "annotator" to label ten candidate terms at each step, for ten random initial labeled term sets[4]. The reported performance is the average number of positive examples among the candidate terms per iteration. An example of the first step of this procedure can be seen in Figure 1: The initial labeled term set L_0, sample from AFINN POS, is expanded, labeled, updated, and expanded again.

5. Results

Table 1 shows the average performance as described in the previous section, i.e. the average number of positive examples found per iteration. The results are displayed per tested term set, expansion method, and underlying distributional model used. It is evident from this table that, generally, Simple Margin using an RBF-kernel outperforms the other expansions methods. This is true for all term sets,

[4]The initial term sets were shared across models and methods.

	Ingredients			
	CBOW	PPMI	**SGNS**	SPMI
centroid expansion	1.69	2.35	1.17	2.37
eigencentrality	1.25	1.64	0.68	1.32
signal to noise	1.45	1.72	0.67	0.82
simple margin linear	0.36	2.31	1.68	1.58
simple margin rbf	1.15	2.46	**2.47**	1.49
	Colors			
	CBOW	PPMI	**SGNS**	SPMI
centroid expansion	1.53	1.92	0.38	1.47
eigencentrality	0.62	0.94	0.18	0.95
signal to noise	0.81	1.22	0.12	1.47
simple margin linear	0.67	2.84	2.57	2.27
simple margin rbf	3.20	2.95	**3.47**	2.58
	AFINN POS			
	CBOW	**PPMI**	SGNS	SPMI
centroid expansion	1.19	1.91	0.09	1.62
eigencentrality	0.53	0.98	0.05	0.96
signal to noise	0.96	1.78	0.08	1.53
simple margin linear	0.90	3.63	2.51	2.33
simple margin rbf	3.25	**4.27**	3.99	2.72
	AFINN NEG			
	CBOW	PPMI	**SGNS**	SPMI
centroid expansion	2.97	4.06	0.74	3.39
eigencentrality	1.27	2.09	0.33	1.94
signal to noise	2.35	3.32	0.21	3.03
simple margin linear	1.38	3.79	3.68	3.20
simple margin rbf	4.63	4.59	**4.78**	3.52
	Elements			
	CBOW	PPMI	SGNS	**SPMI**
centroid expansion	1.77	2.16	1.12	2.21
eigencentrality	1.22	1.74	0.74	1.67
signal to noise	0.79	2.09	0.52	2.08
simple margin linear	0.68	2.59	1.52	2.58
simple margin rbf	1.31	2.60	1.83	**2.65**

Table 1: Average number of positive examples found per iteration of the term set expansion method, based on ten random initialization with five positive and five negative examples. Simple Margin using an RBF kernel is consistently the best expansion method for all term sets.

and almost all combinations of term sets and distributional models tested.

It is also evident that centroid expansion clearly outperforms the other centrality based expansion methods, and in some instances, for some models, outperforms Simple Margin with a linear kernel.

6. Conclusion & discussion

As a best practice when using distributional methods for term set expansion, our results indicate that simple margin using an RBF-kernel is the best choice for all term sets, regardless of the distributional model used. Simple Margin with a linear kernel – which has both the advantage of being directly representable in the vector space, and being efficient to compute – also consistently performed well for all distributional models but CBOW.

strong, enjoyed, excited, excellent, tremendously, thanks, marvellous, rich, **disappointed**, uplifting, fun, enjoy, interesting, enjoying, healthy, terrific, lovely, ambitious, fantastic, enjoyable, **worried**, interested, **sorry**, improved, wonderfully, powerful, **upset**, successful, relieved, amazing,

Figure 2: Top 30 unlabeled candidates for AFINN POS using PPMI and Simple Margin with an RBF kernel after an expansion procedure as described in the section 4. The underlined words are those in the predefined term set, and the bold words are words that we deemed erroneous

It should be noted that, apart from providing the labeler with candidate terms, the simple margin methods also provides a classifier based on the labeled set. This could be used to quickly expand the term sets – without supervision, but with some uncertainty – to include all terms the classifier would consider positive examples. An example of this can be seen in Figure 2.

Both SGNS and CBOW stands out: SGNS, while outperforming most other models when using the RBF method, performed terribly in conjunction with centrality based methods. For CBOW, there is a significant loss of performance when using simple margin with a linear kernel, a phenomena not observed for the other distributional models. This could indicate that the distributional representations produced by SGNS are locally noisy but globally coherent, and representations produced by CBOW are locally coherent, but globally noisy.

It should also be noted that the training data used for the distributional models (BNC) is a comparably small, balanced, corpus. Results would be different for different sizes and kinds of corpora.

7. References

Clear, J. H. (1993). The digital word. chapter The British National Corpus, pages 163–187. MIT Press, Cambridge, MA, USA.

Eisenstein, J. (2017). Unsupervised learning for lexicon-based classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3188–3194.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119.

Nielsen, F. Å. (2011). A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*.

Olsson, F. (2009). A literature survey of active machine learning in the context of natural language processing. Technical Report T2009:06.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The pagerank citation ranking. In *Proceedings of the 7th*

International World Wide Web Conference, pages 161–172, Brisbane, Australia.

Pantel, P., Crestan, E., Borkovsky, A., Popescu, A.-M., and Vyas, V. (2009). Web-scale distributional similarity and entity set expansion. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 938–947. Association for Computational Linguistics.

Sarmento, L., Jijkuon, V., de Rijke, M., and Oliveira, E. (2007). More like these: growing entity classes from seeds. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 959–962. ACM.

Tong, S. and Koller, D. (2002). Support vector machine active learning with applications to text classification. *The Journal of Machine Learning Research*, 2:45–66.

Can Domain Adaptation be Handled as Analogies?

Núria Bel, Joel Pocostales
Universitat Pompeu Fabra
Roc Boronat, 138, 08018 Barcelona, Spain
nuria.bel@upf.edu; joel.pocostales@upf.edu

Abstract

Aspect identification in user generated texts by supervised text classification might suffer degradation in performance when changing to other domains than the one used for training. For referring to aspects such as quality, price or customer services the vocabulary might differ and affect performance. In this paper, we present an experiment to validate a method to handle domain shifts when there is no available labeled data to retrain. The system is based on the offset method as used for solving word analogy problems in vector semantic models such as word embedding. Despite of the fact that the offset method indeed found relevant analogues in the new domain for the classifier initial selected features, the classifiers did not deliver the expected results. The analysis showed that a number of words were found as analogues for many different initial features. This phenomenon was already described in the literature as 'default words' or 'hubs'. However, our data showed that it cannot be explained in terms of word frequency or distance to the question word, as suggested.

Keywords: Aspect identification, domain adaptation, document classification

1. Introduction

Machine Learning in general, and classifiers in particular, might suffer degradation in performance when the data to handle belongs to a different domain than the data used for training. Domain adaptation addresses the problem of moving from a source distribution for which we have labelled training data to a target distribution for which we have no labels. Domain adaptation might be crucial for identifying aspects in Aspect Based Sentiment Analysis (ABSA). Note that while for polarity identification, it is likely that a common vocabulary is shared among different domains (e.g. good, bad), for identifying aspects such as quality, design, or support, different domains might exhibit different vocabulary. For instance, quality for a laptop would be described in terms of 'performance': fast, powerful, etc., while for a restaurant would be described in terms of 'taste': delicious, tasty, etc.

In this paper, we present an experiment designed to validate a method to handle domain shifts when there is no available labelled data in a new target domain for an aspect classifier to be retrained. Our approach was to leverage the use of vector space models for semantics such as the one provided by word embeddings (WE, Mikolov et al. 2013). We experimented with using the offset method, as used for solving word analogy problems, to tackle domain adaptation.

In a WE model, we found the examples like in (1), where responses provided by the offset method are in bold.

(1) laptop : shop :: bread : **bakery**

laptop : shop :: beer : **brewery**

laptop : shop :: medicine: **pharmacy**

Thus, we produced lists of analogue words to support domain adaptation of a system for classifying sentences. The task was classifying user-generated texts as according to the aspect (or attribute) of the product (or entity) the user talks about in the text, as described in Aspect Based Sentiment Analysis, ABSA 2014 (Pontiki et al. 2014), Subtask 1, Slot 1: *Aspect Category Detection*.

For our classification experiment, we worked with the following attributes as classes: Design, Price, Quality and Support. While Price was expected not to be very affected by a change of domain, the others would be more affected because differences in vocabularies.

A SVM classifier per aspect, using a bag of words as sentence representations, was trained with user-generated comments on the domain of laptops and tested on the domain of restaurants. In Table 1, some examples and their intended labelling give a hint about the complexity of the task, given the length of the texts. Note that a text can get more than one label.

	LAPTOPS	RESTAURANTS
DESIGN	Lightweight and the screen is beautiful!	The music is great, and the lighthearted atmosphere will lifts you spirits.
SUPPORT	But no one could tell me when my part would be shipped nor could they tell me where to buy it ON THEIR WEBSITE!!!	We waited for an hour to be seated.
QUALITY	The image is great, and the sound is excellent.	The coffee was good even by xxx[1] standards and the food was outstanding.
PRICE	It's a steal when considering the specs and performance as well.	Good Food, Great Service, Average Prices.

Table 1. Selected examples and intended labels.

Despite the fact that the offset method found some relevant words in the new domain, the results of our evaluation experiment showed no improvement with respect to the baseline.

[1] Brands are anonymized

The remainder of the paper is organized as follows. In section 2, we present a review of related research, in section 3, we describe the methodology followed for the experiment; in section 4, the results are presented. In section 5, the results and the error analysis are discussed and, finally, conclusions are presented in section 7.

2. Related work

Aspect identification was one of the subtasks of Aspect Based Sentiment Analysis (ABSA) in SEMEVAL 2014 (Pontiki et al. 2014)[2]. The goal was to identify product aspects mentioned in user-generated reviews, for instance if a customer was talking about the quality, price or service of a restaurant. Most teams that participated at SemEval ABSA used SVM classifiers and lexical data as features to represent sentences in different implementations of the bag of words (BoW) approach.

The NRC-Canada system (Kiritchenko et al. 2014), which achieved the best scores (88.57 % F1 and 82.92 % accuracy), used SVMs with features based on various types of n-grams and other lexical information learned from the Yelp dataset. Other systems equipped their SVMs with features that were a linear combination of BoW and WordNet seeds (Castellucci et al. 2014). They used aspect terms extracted using a domain lexicon derived from WordNet and a set of classification features created with the help of deep linguistic processing techniques (Pekar et al., 2014), or they only used BoW features (Nandan et al., 2014). Similarly, Brun et al. (2014) used BoW features and information provided by a syntactic parser to train a logistic regression model that assigned to each sentence the probabilities of belonging to each category. Other teams used the MaxEnt model to build classifiers, where only a BoW was used (Zhang et al., 2014) or they used BoW and *Tf-idf* selected features (Brychcin et al., 2014). Liu and Meng (2014) developed a category classifier with the MaxEnt model with the occurrence counts of unigrams and bigrams words of each sentence as features. Other participating teams only employed WordNet similarities to group the aspect terms into categories by comparing the detected aspect terms either against a term (or a group of terms) representative of the target categories (García Pablos et al. 2014) or against all categories themselves (Bornebusch et al. 2014). Veselovská and Tamchyna (2014) simply looked up the aspects' hyperonyms in WordNet. This approach, however, had many limitations and the systems that used it were ranked in the last positions. And finally, the SNAP system (Schulze et al. 2014) proposed a hybrid approach that combined a component based on similarities between WordNet synsets of aspect terms and categories and a machine learning component, essentially a BoW model that employed multinomial Naive Bayes classifier in a one-vs-all setup. Basically, all the systems made extensive use of lexical data and this creates serious problems when changing the domain.

As for domain adaptation methods, there are a number of different algorithms developed for compensating the degradation in performance. Daumé III (2007) and Blitzer et al. (2006) assumed the availability of some labelled examples in the new domain, and most of the methods proposed after these initial works still require some labelled data of the new domain to retrain, which, in practice, are not available. Daumé III (2007) proposed an approach for supervised adaptation by changing the selected features for ones relevant to the new domain and re-training the classifiers with an augmented list of features.

Our method, explained in next section, proposed to augment the initial list of features by projecting them into the new domain. We formulated the problem as analogy questions.

Word analogy questions have been used to demonstrate that vector space representations consistently encode linguistic regularities (Mikolov et al. 2013, Levy et al., 2014, Linzen, 2016, among others). These linguistic relations are referred as "syntactic", including morphological relations such as verbal base forms and gerund forms, or "semantic" involving world knowledge such as currencies in different countries. Our task was closer to find semantic relations, as we intended to find words expressing specialization of taxonomic relations, for instance finding parts-of or properties-of. To our knowledge, this is the first time that word analogy method is applied to a domain shift problem.

3. Methodology

Our proposal worked upon a basic text classification approach that, as we have seen in section 2, used a supervised classifier with features extracted from the corpus represented as a bag of words (BoW). A classifier was trained for every class or aspect, as listed in table 1, and testing was done as one-vs.-all setup. Note that a text can get more than one label.

3.1 Reduced BoW feature selection

The BoW representation of texts has been successfully used for document classification. However, for short text classification, this approach delivers very sparse vectors, which are not useful for classification purposes. Different techniques have been devised for vector dimensionality reduction, among these, the ones based on statistical feature selection according to an observed training dataset. In our experiment, we used Adjusted Mutual Information, AMI (Vinh et al. 2009), and chi-squared test to select the words for representing sentences. While AMI, and in general Mutual Information based measures, are known to be useful to identify relevant features, they are biased towards infrequent words. To compensate this bias, we combined it with chi-squared selected ones. Thus, our system first ranks the best candidates in two separated lists, each using a different measure. Then, the two lists are joined into a new one by summing the AMI and chi-squared scores[3]. For instance, if a word is ranked 3rd by AMI and 5th by chi-squared, in the joined list it will be the 8th. A single BoW of 600 features was used for all the aspect classifiers.

For the classifiers, we trained SMO classifiers (as implemented by Weka, Hall et al., 2009). Texts were processed as follows. First, they were cleaned eliminating urls, hashtags, and rare characters. Second, texts were

[2] ABSA 2016 presented an Out-of-Domain track only for French, but no participants registered for it.

[3] In case of tie, results are ordered alphabetically.

tokenized and lemmatized using Freeling 4.0 (Padró & Stanilovsky, 2012). Stop words were eliminated before assessing the combined AMI+chi-squared rank explained before. Note that brand names were also ignored and were not selected for the BoW if recognized. Once the list of selected words is obtained, another module read texts and converted them into 600 dimension binary vectors.

3.2 Mapping features to the new domain

The domain adaptation experiment was based on this reduced BoW. For each feature in it, analogous words in the new domain were found by applying the vector space-based offset method in the following way.

(2) laptop : [each feature] :: meal : X

Then, when converting the new domain sentences into a vector, the occurrence of either the initial feature or the found X was considered a positive feature. In this way, no retraining of the classifiers would be necessary, and the classifiers would have to perform well in both domains.

Note that for aspect identification, to retrieve a related word, although not exactly a corresponding analogue word, should be enough as the goal is to take into account words that refer to a particular aspect of a product. It could be different for polarity analysis where it is not the same to observe 'good' than 'bad'. But for aspect identification both, even if antonymous, refer to quality, for instance.

For computing the offset, we used the 3COSMUL method as proposed by Levy et al. (2014). 3COSMUL was demonstrated to better balance the different aspects of similarity to prevent that similarity aspects in different scales can be more predominant in the calculation. The list of analogues proposed by 3COSMUL, which comes from all the corpus vocabulary, was filtered by discarding stop-words and forms not found in a spelling dictionary. Therefore, the list of features used for the out-of-domain classification experiment included the initial ones and the features that were ranked first by 3COSMUL that were actual words (preventing, for instance, forms such as *tablespoonful*) and were not prepositions, pronouns, etc.

To create the vector space model to extract WE, a ten window `word2vec` Skip-Gram (Mikolov et al., 2013) with negative sampling model was trained with the following corpora: a Wikipedia dump[4] and the training initial domain datasets totalling 636M words. Other parameters were: algorithm SGNS, 300 dimensions, context window = 10, subsampling $t=10^{-4}$, context distribution smoothing = 0.75, and 15 iterations.

3.3 Evaluation Datasets

We used the ABSA 2016 (Pontiki et al. 2016) datasets for English on *laptops* and *restaurants*. ABSA 2016 proposed a closed list of aspect or attribute labels for each product or entity. Our aim was using the classifier trained for a particular domain (laptops) to classify texts of another domain (restaurants) and therefore a common set of labels was needed. Moreover, ABSA entities are very fine-grained: the restaurant corpus included six entity labels (i.e. restaurant, food, drinks, ambience, service, location) and the laptop corpus included 22 (i.e. laptop, display, keyboard, mouse, motherboard, cpu, fans_cooling, ports, memory, power_supply optical_drives, battery, graphics, hard_disk, multimedia_devices, hardware, software, OS, warranty, shipping, support, company). Therefore, for the experiments reported here, only the laptop and restaurant entities were considered. As for attributes, we used DESIGN, PRICE, QUALITY and SUPPORT (Vázquez et al. 2014, Bel et al. 2017) as labels for general aspect identification. ABSA entities and attributes were automatically mapped to these labels as follows:

- LAPTOP DESIGN_FEATURES and RESTAURANT STYLE attribute labels (used for all the texts that include a reference about specific features such as size, color, presentation, styling, ambience, etc.) were directly relabeled as a single DESIGN label.
- LAPTOP PRICE and RESTAURANT PRICES, for texts that comment on prices of goods or services, were relabeled as PRICE.
- QUALITY attributes, for texts that refer to the quality, performance or positive and negative characteristics of a product or service that affects user experience were used as our QUALITY label.
- Both LAPTOP SUPPORT, for pre- and after-sales customer support, repair services and staff, and RESTAURANT SERVICE, for opinions focusing on the service in general, staff's attitude and professionalism, etc., were merged into a common SUPPORT label.

As already mentioned, the laptop corpus was used for training (but a small held out dataset for testing) and the restaurant corpus for testing the out-domain scenario. Table 2 shows the distribution of the datasets used for the experiment.

	Training	Testing	
Aspect		ID-HO	OD
DESIGN	344	105	57
QUALITY	378	87	238
SUPPORT	144	33	144
PRICE	136	25	44
NONE	2601	323	103
TOTAL	3603	573	587

Table 2. Size of the datasets in sentences. ID-HO for In-Domain Held Out and OD for Out Domain. NONE label is for texts with other labels, ignored in this experiment, or with no labels

4. Results

Table 3 shows the results of the classifiers, in a one-vs-all scenario, for testing with in-domain data as well as out-domain data and out of domain data represented with

[4] Snapshots of 19-03-2016

vectors created with the initial and analogue feature list. Analogues found by the offset method were added to the initial selected feature list when converting sentences into vectors, as explained in section 3.2.

	In Domain			Out Domain			Analogues		
	P	R	F1	P	R	F1	P	R	F1
DESIGN	0.27	0.56	0.36	0.10	0.19	0.12	0.08	0.28	0.12
QUALITY	0.30	0.67	0.42	0.49	0.22	0.30	0.39	0.20	0.26
SUPPORT	0.19	0.42	0.26	0.68	0.51	0.58	0.46	0.60	0.52
PRICE	0.25	0.84	0.39	0.40	0.38	0.39	0.34	0.36	0.35

Table 3: Results of the classification experiment

As expected, the performance of the classifiers was affected by the change of domain. For three of the four categories, the most relevant impact was a noticeable decrease in recall, because of the lexical differences in these two domains. As expected, the category PRICE was not much affected, in general terms, because of the little change of vocabulary, as already mentioned, original features are also used. Surprisingly, the SUPPORT category performed better in the out-domain scenario, but it might be because this category is more clearly expressed in the restaurant data. Compare restaurant examples as in (2) with laptop examples as in (3), for instance (in bold, words in the initial list of selected features).

(2) We **never** had to **wait** more than 5 **minutes**.

The **service ranges** from mediocre to offensive.

(3) It **took** 3 **days** to **make** an appointment at the local **store**.

I could not **believe** they did not **consider** the **battery** as defective so I **went** to the **store** myself and **asked** for a manager.

As for the out-domain dataset using sentence representation that took into account the suggested analogous words, the results did not show the expected improvements in recall, nor a consistent improvement with respect to the out-domain simple test. We discuss these results in the next section.

5. Discussion

Despite of the fact that the offset method indeed found relevant words on the new domain, as shown in Table 4, the classification results did not show the expected improvement. We performed an error analysis addressing two questions: to what extent the analogy questions indeed retrieved words related to the new domain and therefore could be informative, and to what extent found analogues were good features for the classifier.

5.1 Are selected words analogues?

It has already been said that, for the aspect classification task, we expected to find a method for mapping, in a loose way, specialized words from one domain to another. Basically, we expected that there would be some words pairs like the ones in (1) that would map. The intuition was that the analogy method could find related words that without being real analogues, nevertheless, could be useful for aspect classification. There would be lexical relations such as 'part-of'' that would map laptop components to meal components, or cases like *driver-baguette*, *keyboard-accompaniment*, and others shown in table 4. For other words that were not particularly related to any of the domains, we expected that the method would select near synonyms, like the examples in table 5, which are also actual mappings in our experiment.

Laptop	Meal
hour	mealtime
delivery	take-out
quality	palatability
problem	undernourishment
computer	food
outlet	grab-and-go
nightmare	ravenous
absolutely	scrumptious
store	grocery
ergonomics	dietetics
house	tavern
keyboard	accompaniment
premium	all-you-can-eat
book	cookbook
shop	bakery
player	appetizer
driver	baguette
headphone	grill
wire	sirloin
chat	mealtime
office	restaurant
bar	restaurant

Table 4: Examples of analogues proposed for domain words

Laptop	Meal
travel	trip
trip	journey
dislike	disgust
area	region
probably	likely
start	begin
begin	start
build	construct

Table 5: Example of analogues proposed for general domain words

Laptop	Meal
surf	barbecue
company	beverages
email	dinner
adaptor	fatty
box	oatcake
edit	paella
print	ragout

Table 6: Example of analogues in the new domain but with a non-obvious relation to the feature

After manual evaluation, we observed that indeed most of the selected analogues belong to the new domain, although sometimes the relation between the feature and the

analogue was not obvious, as the selected examples in table 6 shows.

A more quantitative evaluation of the results to assess the mapping to the new domain was not possible because of the impact of the hubness problem that we describe in the next section.

5.2 Are analogues good classification features?

In order to evaluate how good the selected words were for classifying the restaurant texts, we run the feature selection method, described in section 3, with the restaurant corpus and we compared the resulting list of 423 selected features with the list of features resulting of the laptop corpus and with the list of proposed analogues.

Laptop domain feature list and the restaurant one shared 201 words, what supports our decision of keeping features selected for the laptop domain also when classifying the new domain.

The list of proposed analogues and the list of selected features for the restaurant corpus shared 48 types, that is, 48 unique words out of the list of 423 features. Some words were suggested as analogues for many different features. Figure 1 plots the number of repetitions for the 58 unique words that were suggested. For instance, *food* was suggested as analogue for 46 different features, *high-carbohydrate* for 40 and *eat* for 81, while *dinner*, *grill*, or *sirloin* were selected for two each.

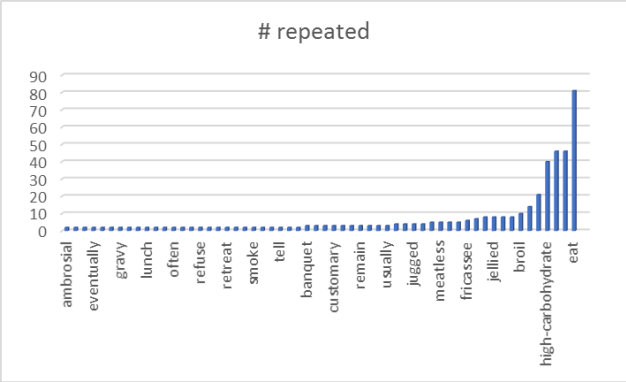


Figure 1: Number of repetitions of the 58 analogues that were suggested for a total of 423 features

For our classifiers, these repetitions caused the classifier vectors to be totally inconsistent and obviously impacted their accuracy rate.

This repetition phenomenon has already been discussed in the literature: Levy et al. (2014) call them "default words" and Dinu et al. (2015), "hubs". Dinu et al. (2015) found it in the task of English into Italian bilingual lexicon induction. When inducing a function from one language vector space to another language vector space, it was already found that neighborhoods surrounding mapped vectors contained many items that were, in their terms, "universal" neighbors which were called "hubs". Dinu et al. noted that the hubness problem was exacerbated when there was a mapping from one original space to a target space.

Levy et al. (2014) found it in the analogy task and defined the problem as "one central representative word is provided as an answer to many questions of the same type" and, in this work, it was observed both for explicit and embedding word representations, accounting for the 39% of the errors[5]. In our case, default words represented 43% of the errors. Seven default words (*broil*, *multi-course*, *hearty*, *high-carbohydrate*, *food*, *non-halal* and *eat*) were repeated more than ten times. However, in our results, in addition to clear central representative words (like *food* and *eat*) there were also words that can hardly be considered representative, even although there was a certain degree of similarity for the features that got the same analogue. In table 7, some examples of these cases are shown. Note that for *hearty* most of the analogues seem to be related to 'emotional' concepts, for *appetizer* to images, although for *broil* a common characteristic is not obvious.

hearty	broil	appetizer
little	battery	graphic
charm	processor	card
reputation	windows	pro
fan	install	video
well	add	download
nicely	charger	player
satisfy	pc	beats
hope	heat	bonus
disappointment	resolve	
definitely	reinstall	
astonish		
appreciate		
surprise		
enjoy		
truly		
disappoint		
impress		

Table 7: Examples of features that got the same analogues

Linzen (2016), who revised the consistent encoding of lexical semantic relations in vector semantic spaces, found that for most of the cases, the offset method, or more precisely the use of cosine distance for assessing it, tends to retrieve the word which is closest to the query word, in our case *meal*. Furthermore, Schnabel et al. (2015) observed that there is a strong correlation between the frequency of a word and its position in a ranking of nearest neighbors.

However, as table 8 shows, not all the hub words were among the 11 closest words to meal (cf. 'other hubs') and

[5] Levy et al. (2014) define default behavior error when the same incorrect answer is returned for a particular relation 10 or more times.

some of the closest words were never selected (i.e. *breakfast*). As for frequency, although some are very frequent[6] (i.e. *food*) others are rather infrequent words (i.e. *multicourse*). Note that *meal* itself, with a relative frequency of 42.8 per million, is in the same frequency range than *eat*, which is the most suggested analogue. However, it still needs further investigation to find what are the conditions that make other hubs to appear.

Closest words to *meal* (descending)	RF	#	Other hubs	RF	#
lunch	9.60	2	broil	0.18	10
multicourse	0.01	14	hearty	0.60	21
soup	7.30	8	high-carbohydrate	0.03	40
food	132.20	46			
dinner	17.80	2			
eat	45.30	81			
breakfast	11.60	0			
bread	13.30	0			
snack	4.40	0			
three-course	0.06	2			
non-halal	0.02	46			

Table 8: List of closest words to 'meal' in descending order. RF stands for relative frequency in percentage per million words, and # for number of repetitions.

6. Conclusion

In this paper, we have presented the results and analysis of an experiment that approached domain adaptation as a search for analogues of the selected features of a reduced BoW used to train a SVM classifier. The benefit of such approach would be that the classifier could be used for a new domain without retraining it with a new domain labelled dataset. The results have shown that 3COSMUL (Levy et al., 2014), used in a vector space created with word2vec (Mikolov et al. 2013), found analogues which were relevant for the new domain in a task of aspect identification. However, the phenomenon known as 'default' words or 'hubs', in which some few analogues are selected for many original question words, makes the resulting vectors a bad input for the classifiers. After an error analysis, we found that, contrary to what has been published before, those analogues which are proposed for many features are, although words of the new domain, either very infrequent words, and not generic words (Levy et al., 2014), or words that were not among the closest to the query words (Linzen, 2016). Currently, there is a growing interest in understanding the characteristics of the WE vector space, and how operations like vector offset for finding analogues actually work (Gittens et al. 2017). In future work, we will explore in our data the hints provided by these works to seek a method to improve the list or to filter it such that it allows to create an easy and cheap method for domain adaptation.

[6] Frequency, expressed in percentage per million, was assessed at the English Wikipedia 2014 Corpus of 1.3 billion words using SketchEngine. https://www.sketchengine.co.uk/

7. Acknowledgements

This work was supported by the Spanish TUNER project TIN2015-65308-C5-5-R (MINECO/FEDER, UE).

8. Bibliographical References

Bel, N.; J. Diz-Pico, M. Marimon, J. Pocostales (2017) Classifying short texts for a Social Media monitoring system, Procesamiento del Lenguaje Natural, Revista nº 59, 57-64.

Blitzer, J.; McDonald, R.; and Pereira, F. 2006. Domain adaptation with structural correspondence learning. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP '06). Association for Computational Linguistics, Stroudsburg, PA, USA, 120-128.

Bornebusch, F.; G. Cancino; M. Diepenbeck; R. Drechsler; S. Djomkam; A. Nzeungang Fanseu; M. Jalali; M. Michael; J. Mohsen; M. Nitze; C. Plump; M. Soeken; M. Tchambo; and H. Ziegler. 2014. "itac: Aspect based sentiment analysis using sentiment trees and dictionaries". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, pages 351–355.

Brun, C.; D. N. Popa; and C. Roux. 2014. "Xrce: Hybrid classification for aspect-based sentiment analysis". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 838–842.

Brychcín, T.; M. Konkol, and J. Steinberger. 2014. "Uwb: Machine learning approach to aspect-based sentiment analysis". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 817–822.

Castellucci, G.; S. Filice; D. Croce; and R. Basili, R. 2014. "Unitor: Aspect based sentiment analysis with structured learning". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 761–767.

Daume III, H. 2007. Frustratingly easy domain adaptation. In Conference of the Association for Computational Linguistics (ACL), Prague, Czech Republic.

Dinu, G.; A. Lazaridou, and M. Baroni, 2015. Improving zero-shot learning by mitigating the hubness problem. In Proceedings of ICLR Workshop Track, San Diego, CA.

García Pablos, A.; M. Cuadros, and G. Rigau. 2014. "V3: Unsupervised generation of domain aspect terms for aspect based sentiment analysis". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 833–837.

Gittens, A., Achlioptas, D., & Mahoney, M. W. (2017). Skip-Gram – Zipf + Uniform = Vector Additivity. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (pp. 69–76)

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. 2009. The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.

Kiritchenko, S., X. Zhu, C. Cherry, and S. Mohammad. 2014. "Nrc-canada-2014: Detecting aspects and sentiment in customer reviews". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 437–442.

Lazaridou, A., Dinu, G., & Baroni, M. (2015). Hubness and pollution: Delving into crossspace mapping for zero-shot learning. In Proceedings of the 2015 Conference Association for Computational Linguistics, (ACL) Stroudsburg, PA, USA,, pp. 270-280.

Levy, O., Goldberg, Y. and Ramat-Gan, I. 2014. Linguistic regularities in sparse and explicit word representations. CoNLL-2014.

Linzen. T. (2016). Issues in evaluating semantic spaces using word analogies. In Proceedings of the First Workshop on Evaluating Vector Space Representations for NLP, 13–18

Liu, P. and H. Meng. 2014. "Seemgo: Conditional random fields labeling and maximum entropy classification for aspect based sentiment analysis". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 527–531.

Mikolov, T., Chen, K.; Corrado, M. and Dean, J. 2013. Efficient Estimation of Word Representations in Vector Space. Proceedings of Workshop at ICLR, 2013.

Nandan, N.; D. Dahlmeier, A. Vij, and N. Malhotra. 2014. "Sapri: A constrained and supervised approach for aspect-based sentiment analysis," in Proceedings of the 8th International Workshop on Semantic Eva-luation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 517–521.

Padró, Ll. and E. Stanilovsky. 2012. FreeLing 3.0: Towards Wider Multi-linguality. In Proceedings of the Language Resources and Evaluation Conference (LREC 2012) ELRA.

Pekar, V.; N. Afzal, and B. Bohnet. 2014. "Ubham: Lexical resources and dependency parsing for aspect-based sentiment analysis," in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 683–687.

Pontiki, M.; D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar. (2014). SemEval 2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014) at COLING 2014, pages 27-35. Dublin, Ireland.

Pontiki, M.; D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, Q. Bing, O. De Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. Jiménez-Zafra, G. Eryigit (2016) "SemEval-2016 Task 5: Aspect Based Sentiment Analysis,"Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval '16, San Diego, California, June 16-17, 2016.",19–30,2016, Association for Computational Linguistics.

Schulze Wettendorf, C., R. Jegan, A. Körner, J. Zerche, N. Plotnikova, J. Moreth, T. Schertl, V. Obermeyer, V. Streil, T. Willacker, and S. Evert. 2014. "Snap: A multi-stage xml-pipeline for aspect based sentiment analysis". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 578–584.

Vázquez, Silvia; Muñoz-García, Óscar; Campanella, Inés; Poch, Marc; Fisas, Beatriz; Bel, Nuria; Andreu, Gloria (2014). "A classification of user-generated content into consumer decision journey stages, Neural Networks,58, 68-81.

Veselovská K. and A. Tamchyna. 2014. "U´ fal: Using hand-crafted rules in aspect based sentiment analysis on parsed data". In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, August 2014, pages 694–698.

Vinh, N. X., Epps, J., and Bailey, J. (2009). Information theoretic measures for clustering comparison: is a correction for chance necessary? In Proc. the 26th International Conference on Machine Learning (ICML'09), p.1073- 1080. ACM.

Zhang, F.; Z. Zhang, and M. Lan. 2014. "Ecnu: A combination method and multiple features for aspect extraction and sentiment polarity classification," in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland: ACL and Dublin City University, pages 252–258.

Author Profiling from Facebook Corpora

Fernando Chiu Hsieh, Rafael Felipe Sandroni Dias, Ivandré Paraboni
University of São Paulo, School of Arts, Sciences and Humanities
São Paulo, Brazil
fernando.hsieh@usp.br , rafaelsandronidias@gmail.com, ivandre@usp.br

Abstract

Author profiling - the computational task of prediction author's demographics from text - has been a popular research topic in the NLP field, and also the focus of a number of prominent shared tasks. Author profiling is a problem of growing importance, with applications in forensics, security, sales and many others. In recent years, text available from social networks has become a primary source for computational models of author profiling, but existing studies are still largely focused on age and gender prediction, and are in many cases limited to the use of English text. Other languages, and other author profiling tasks, remain somewhat less popular. As a means to further this issue, in this work we present initial results of a number of author profiling tasks from a Facebook corpus in the Brazilian Portuguese language. As in previous studies, our own work will focus on both standard gender and age prediction tasks but, in addition to these, we will also address two less usual author profiling tasks, namely, predicting an author's degree of religiosity and IT background status. The tasks are modelled by making use of different knowledge sources, and results of alternative approaches are discussed.

Keywords: Author profiling, Document Classification, Corpus-based approaches

1. Introduction

Author profiling - the computational task of prediction author's demographics from text - has been a popular research topic in the NLP field, and also the focus of a number of prominent shared tasks (Pardo et al., 2017). Author profiling is a problem of growing importance, with applications in forensics, security, sales and many others (Rangel et al., 2015).

In recent years, text available from social networks has become a primary source of data for computational author profiling. However, existing studies of this kind are often focused on age and gender prediction, and are in many cases limited to the use of English text and a few others. Other languages, and other author profiling tasks, by contrast, remain somewhat less popular.

In this work we address the issue of computational author profiling from a Facebook corpus in the Brazilian Portuguese language. In doing so, we address standard age and gender prediction tasks, and also two less-known alternatives: predicting an author's degree of religiosity and his/her IT background status. Religiosity prediction may be seen as an extension of personality recognition (Mairesse et al., 2007), and degrees of religiosity are indeed known to correlate with certain personality traits in the Big Five model (de Andrade, 2008). IT background status prediction, on the other hand, may be seen as a means to group authors into (e.g., professional) communities.

In order to predict age, gender, religiosity and IT background status, we propose a number of author profile models based on standard document classification methods (from bag of words to word embeddings) and using different knowledge sources (from purely textual to psycholinguistic features). In doing so, our goal is to determine which methods are best suited for the four tasks at hand based on Facebook text in the Brazilian Portuguese language.

The rest of this paper is structured as follows. A number of existing author profiling models are reviewed in Section 2. Our own approach and corpus are described in Section 3. Evaluation and its results are described in Section 4. Final remarks are presented in Section 6.

2. Background

Studies on gender and age prediction are popular in NLP and related fields, and have in many cases been carried out in the light of the PAN-CLEF Shared task series (Pardo et al., 2017). Outside the scope of shared tasks, however, a direct comparison among existing methods may be complicated by the fact that different studies may rely on different problem definitions (e.g., regression versus classification), different datasets, languages (in most cases, English), and evaluation metrics (e.g., accuracy, F-measure etc.). These difficulties notwithstanding, in what follows we briefly discuss a number of recent studies of this kind.

The work in (Nguyen et al., 2014) presents a comparative study between an automated system and an experiment with human subjects to predict the gender and age of Twitter users, and discusses the limitations of current computational approaches to gender and age prediction from text. The study highlights the difference between social and biological identities, showing that more than 10% of the Twitter users do not employ language that would normally be associated with their biological sex, and that older Twitter users are often perceived to be younger. The study is based on Dutch text translated into English, and compares a linear regression model for age prediction and logistic regression for gender prediction with guesses provided by human subjects. A model based on majority vote reaches an accuracy of 0.84, a result that is arguably similar to existing automatic classification systems on English Twitter data.

The work in (Sap et al., 2014) presents predictive lexica for age and gender using regression and classification models from language usage in social networks text with associated demographic labels. The weighted lexica obtained average

prediction accuracy of 0.83 (age) and 0.82 (gender) for the English language.

The work in (Álvarez-Carmona et al., 2015) proposes gender and age recognition models that combine second order features (already employed in their own previous work at PAN 2013 and 2014) with latent semantic analysis (LSA). The resulting model is the overall winner of the PAN-2015 shared task, ranking among the top three systems in the three languages of the task (English, Dutch and Spanish).

The work in (op Vollenbroek et al., 2016) makes use of a SVM model trained on English, Dutch and Spanish Twitter data for gender and age prediction from unknown, non-Twitter text. Given the goal of building a cross-genre model for these tasks, the work avoids the use of language-specific features, focusing instead on n-gram counts, capitalisation, punctuation, word and sentence length, out-of-vocabulary words, vocabulary richness, function words, part-of-speech and emoticons. The model also includes second-order features representing relative values of some of these absolute measures. The resulting model in (op Vollenbroek et al., 2016) was the overall winner of the PAN-2016 cross-genre author profiling task (Rosso et al., 2016), obtaining 0.53 joint accuracy (i.e., age and gender prediction put together) when combining results from English, Spanish and Dutch texts. However, the authors pointed out that their present (cross-genre) results were considerably lower than those obtained in previous (single-genre) author profiling tasks at the PAN-CLEF series.

The work in (Basile et al., 2017) presents a model called N-GrAM for gender prediction from Twitter text in English, Spanish, Arabic and European Portuguese. The model makes use of a linear SVM model with word unigrams and character 3- to 5-grams as features. Interestingly, language- and domain-dependent features such as POS tags and Twitter handles were found to decrease overall accuracy. The proposed model was the best-performing participant in the PAN-CLEF-2017 shared task (Pardo et al., 2017), with 0.83 average accuracy when considering the four languages combined.

Finally, the work in (Guimarães et al., 2017) presents a method for binary age classification from Twitter, and it is one of the few studies to address this task using text in Brazilian Portuguese. The model makes use of Twitter-related features such as punctuation, text length, sharing status, and topic. The study compares a number of classification methods, including MLP, DCNN, Random Forest and SVMs. Among these, best results were obtained by using DCNN, with 0.94 average F-measure.

3. Current work

We designed an experiment to compare a number of document classification models applied to four prediction tasks from a Brazilian Portuguese Facebook corpus. These tasks are described individually in the following sections.

3.1. Author profiling tasks

In the present work we consider both standard gender and age-bracket prediction, and two less usual author profiling tasks, namely, predicting author's degree of religiosity and IT background status. The focus on religiosity and IT background is mainly motivated by our choice of corpus (cf. next section), in which both kinds of information are readily available. Both issues however constitute potentially relevant research questions on their own right.

Age bracket prediction was modelled as a three-class problem (18-20, 23-25, and 28-61 years-old). As in (Rangel et al., 2015), instances in the intermediate age brackets (i.e., 20-23 and 25-28) were disregarded in order to minimise the possible mismatch between a Facebook user age and the actual time of the publication.

Gender and *IT-background status* prediction were modelled as binary classification tasks (male / female and yes / no).

Religiosity is represented by self-reported scores ranging from 1 (not religious at all) to 5 (highly religious), and the corresponding prediction task was modelled as a three-class problem (1-2 degrees, 3 degrees, and 4-5 degrees) so as to obtain nearly-balanced groups[1].

3.2. Computational models

In what follows we will consider five author profiling models. First, given that author profiling may be seen as a document classification task, we will consider models based on both word counts (bag-of-words) and TF-IDF counts. Second, since char n-gram models have been popular in the related task of author prediction (Tschuggnall et al., 2017) we will also investigate their use in our present tasks. Third, given that other author profiling tasks - in particular, personality prediction as in (Mairesse et al., 2007) - may benefit from the use of psycholinguistic lexical information, we will make use of LIWC (Pennebaker et al., 2001) and related features as well. Finally, as in many recent NLP tasks, we will also consider the use of distributed word representations (Mikolov et al., 2013). These models are detailed below.

We will consider three standard text models that do not rely on external knowledge sources: the *TF-IDF* model, which consists of TF-IDF counts for the 3k most frequent terms; the *BoW*, which is a Bag-of-words model based on word counts, also keeping the 3k most frequent words; and the *Char* model, which is a standard 3-5 character n-gram model.

In addition to that, we will also consider a lexical model conveying 68 psycholinguistic features obtained from two external knowledge sources: the LIWC psycholinguistic dictionary (Pennebaker et al., 2001) and additional MRC-like (Coltheart, 1981) psycholinguistic properties. This model is presently labelled as *LIWC+P*.

From the Brazilian Portuguese of the LIWC dictionary (Filho et al., 2013), we computed 64 features representing psycholinguistic word categories (e.g., anger, family, wealth etc.). Each feature represents the number of words found in the corresponding category divided by the total number of words in the document. Moreover, from the psycholinguistic properties database in (dos Santos et al.,

[1] The choice for this scale is similar to (de Andrade, 2008). For a more thorough account of religiosity - including, e.g., the distinction between organisational, non-organisational and subjective religiosity - see (Koenig and Bussing, 2010).

2017a), we computed four features representing concreteness, imageability, subjective frequency and age of acquisition for Brazilian Portuguese text. Each of these features represents the average score of all document words in the corresponding category.

Finally, we will also consider a model based on word embeddings hereby called *Word2Vec*. This consists of the average word vectors obtained from a skip-gram-600 model (Mikolov et al., 2013) using window size=5, min_count=10, built from a 50-million Twitter corpus.

All the above models were built using logistic regression. Language-specific resources in *LIWC+P* and *Word2Vec* models (i.e., psycholinguistic dictionaries and Twitter data) concern the Brazilian Portuguese language only.

3.3. Data

In our experiment we make use of a portion of the *b5* corpus of texts and accompanying author demographics for the Brazilian Portuguese language (Ramos et al., 2018). The corpus has been applied to a number of NLP/NLG tasks ranging from personality recognition (dos Santos et al., 2017b; Silva and Paraboni, 2018) to personality-dependent content selection (Paraboni et al., 2017) and lexical choice (Lan and Paraboni, 2018).

All models were built from the 2.2 million word *b5-post* subcorpus of Facebook status updates. This dataset conveys up to 1,000 status updates from 1019 users of Brazilian Portuguese Facebook. The data is partially labelled with age, gender, degrees of religiosity and IT status information[2].

Gender, age and IT-background information were generally obtained from Facebook and, in some cases, provided by some of the participants of the *b5-post* data collection task upon request. Degrees of religiosity were provided by a small subset of participants upon request.

Table 1 summarises the number of documents (i.e., user's Facebook time lines) to be classified in each of the four prediction tasks.

Task	Documents
Age bracket	516
Gender	1018
Religiosity	440
IT background	814

Table 1: Data size for each classification task

Modelling age prediction as a classification task (as opposed to, e.g., a regression problem) requires deciding how to define age brackets. Our present choice is influenced by the concentration of individuals in their 20's (including a large proportion of undergraduate students) in our data. This distribution is illustrated in Figure 1.

4. Evaluation

After stop words removal using NLTK[3], the five models - *TF-IDF*, *Bow*, *Char*, *LIWC+P* and *Word2Vec* - were trained from the entire dataset for each classification problem. Results from 10-fold cross validation logistic regression are summarised as follows: age bracket in Table 2, gender in Table 3, religiosity in Table 4 and IT background status in Table 5.

Class	N	TF-IDF	BoW	Char	LIWC+P	Word2Vec
18-20	182	**0.60**	0.57	0.56	0.51	0.58
23-25	189	**0.56**	0.48	0.48	0.40	0.48
28-61	145	**0.61**	0.55	0.54	0.45	0.59

Table 2: Age bracket F-measure results

Class	N	TF-IDF	BoW	Char	LIWC+P	Word2Vec
female	578	**0.90**	0.86	0.84	0.80	0.88
male	440	**0.86**	0.81	0.79	0.72	0.85

Table 3: Gender F-measure results

Class	N	TF-IDF	BoW	Char	LIWC+P	Word2Vec
1-2	217	**0.67**	0.60	0.55	0.61	0.59
3	96	0.17	0.19	0.18	0.22	**0.26**
4-5	127	**0.52**	0.40	0.38	0.41	0.43

Table 4: Religiosity F-measure results

5. Discussion

As expected for a small corpus of this kind, overall results vary across dataset size and class definition. Best results are observed in the case of the two larger binary classes, namely gender classification and IT background status. Age bracket classification was moderately accurate, and degrees of religiosity classification had the lowest accuracy of all, with particularly low results for the sparse intermediate (degree 3) class.

Regarding the computational methods under consideration, we notice that the *TF-IDF* model generally outperforms the alternatives, with *Word2Vec* as a second best (particularly in the case of gender identification). By contrast, the lexical *LIWC+P* method generally produced the lowest results of all, which seems to suggest that psycholinguistic knowledge may be more suitable for personality and sentiment recognition than for the present author profiling tasks.

6. Final remarks

We have presented initial results of an author profiling task from Facebook text for the Brazilian Portuguese language. Our work has focused on both standard gender and age prediction, and less usual tasks of religiosity and IT background status prediction. We have compared a number of logistic regression models, and overall best results were observed when using TF-IDF counts. As future work, we intend to build higher-order n-gram models of Portuguese (Pereira and Paraboni, 2007; de Novais and Paraboni, 2012) and larger word embedding representations to further the use of deep learning methods applied to the current tasks.

[2]The latter is a result of having a significant proportion of Computer Science students among the target participants.

[3]http://www.nltk.org/

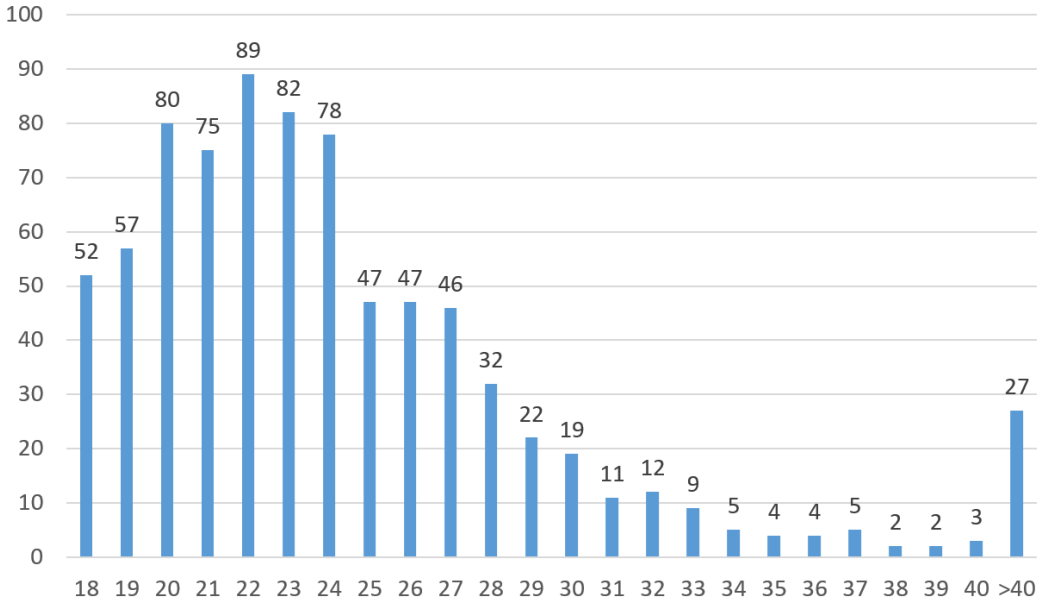


Figure 1: Age distribution in the b5-post corpus.

Class	N	TF-IDF	BoW	Char	LIWC+P	Word2Vec
no	491	**0.80**	0.76	0.73	0.75	0.73
yes	323	**0.64**	0.60	0.59	0.55	0.59

Table 5: IT background F-measure results

7. Acknowledgements

This work has been supported by the Brazilian Ministry of Education PET programme, and by grant # 2016/14223-0, São Paulo Research Foundation (FAPESP).

8. Bibliographical References

Álvarez-Carmona, M., López-Monroy, A., y Gómez, M. M., Villaseñor-Pineda, L., and Escalante, H. (2015). INAOE's participation at PAN'15: Author Profiling task. In *CLEF 2015*.

Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., and Nissim, M. (2017). N-GrAM: New groningen author-profiling model. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum*, Dublin.

Coltheart, M. (1981). The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology*, 33(4):497–505.

de Andrade, J. M. (2008). *Evidências de validade do inventário dos cinco grandes fatores de personalidade para o Brasil*. Ph.D. thesis, Universidade de Brasília.

de Novais, E. M. and Paraboni, I. (2012). Portuguese text generation using factored language models. *Journal of the Brazilian Computer Society*, 19(2):135–146.

dos Santos, L. B., Duran, M. S., Hartmann, N. S., Junior, A. C., Paetzold, G. H., and Aluísio, S. M. (2017a). A lightweight regression method to infer psycholinguistic properties for Brazilian Portuguese. In *TSD-2017*.

dos Santos, V. G., Paraboni, I., and Silva, B. B. C. (2017b). Big five personality recognition from multiple text genres. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 29–37, Prague, Czech Republic. Springer-Verlag.

Filho, P. P. B., Aluísio, S. M., and Pardo, T. (2013). An evaluation of the Brazilian Portuguese LIWC dictionary for sentiment analysis. In *9th Brazilian Symposium in Information and Human Language Technology - STIL*, pages 215–219, Fortaleza, Brazil.

Guimarães, R. G., Rosa, R. L., de Gaetano, D., Rodríguez, D. Z., and Bressan, G. (2017). Age groups classification in social network using deep learning. *IEEE Access*, 5:10805–10816.

Koenig, H. G. and Bussing, A. (2010). The duke university religion index (DUREL): a five-item measure for use in epidemiological studies. *Religions*, 1(1):78–85.

Lan, A. G. J. and Paraboni, I. (2018). Definite description lexical choice: taking speaker's personality into account. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

Mairesse, F., Walker, M., Mehl, M., and Moore, R. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research (JAIR)*, 30:457–500.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Nguyen, D.-P., Trieschnigg, R. B., Dogruoz, A. S., Gravel, R., Theune, M., Meder, T., and de Jong, F. M. (2014). Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment. In *Proceedings of COLING-2014*, pages 1950–1961. Association

for Computational Linguistics.

op Vollenbroek, M. B., Carlotto, T., Kreutz, T., Medvedeva, M., Pool, C., Bjerva, J., Haagsma, H., and Nissim, M. (2016). GronUP: Groningen user profiling: Notebook for PAN at CLEF 2016. In *CEUR Workshop Proceedings*, Netherlands.

Paraboni, I., Monteiro, D. S., and Lan, A. G. J. (2017). Personality-dependent referring expression generation. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 20–28, Prague, Czech Republic. Springer-Verlag.

Pardo, F. M. R., Rosso, P., Potthast, M., and Stein, B. (2017). Overview of the 5th author profiling task at PAN 2017: Gender and language variety identification in Twitter. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum*, Dublin.

Pennebaker, J. W., Francis, M. E., and Booth, R. J. (2001). *Inquiry and Word Count: LIWC*. Lawrence Erlbaum, Mahwah, NJ.

Pereira, D. B. and Paraboni, I. (2007). A language modelling tool for statistical NLP. In *5th Workshop on Information and Human Language Technology (TIL-2007). Anais do XXVII Congresso da SBC*, pages 1679–1688, Rio de Janeiro. Sociedade Brasileira de Computação.

Ramos, R. M. S., Neto, G. B. S., Silva, B. B. C., Monteiro, D. S., Paraboni, I., and Dias, R. F. S. (2018). Building a corpus for personality-dependent natural language understanding and generation. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

Rangel, F., Celli, F., Rosso, P., Potthast, M., Stein, B., and Daelemans, W. (2015). Overview of the 3rd Author Profiling Task at PAN 2015. In *CLEF 2015 Evaluation Labs and Workshop*.

Rosso, P., Rangel, F., Potthast, M., Stamatatos, E., Tschuggnall, M., and Stein, B. (2016). Overview of PAN'16—New Challenges for Authorship Analysis: Cross-genre Profiling, Clustering, Diarization, and Obfuscation. In *7th International Conference of the CLEF Initiative (CLEF 16)*, Berlin Heidelberg New York. Springer.

Sap, M., Park, G., Eichstaedt, J., Kern, M., Stillwell, D., Kosinski, M., Ungar, L., and Schwartz, H. (2014). Developing age and gender predictive lexica over social media. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1146–1151, Doha, Qatar. Association for Computational Linguistics.

Silva, B. B. C. and Paraboni, I. (2018). Learning personality traits from Facebook text. *IEEE Latin America (to appear)*.

Tschuggnall, M., Stamatatos, E., Verhoeven, B., Daelemans, W., Specht, G., Stein, B., and Potthast, M. (2017). Overview of the 5th author profiling task at PAN 2017: Style breach detection and author clustering. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum*, Dublin.

Semantic Relatedness of Wikipedia Concepts – Benchmark Data and a Working Solution

Liat Ein Dor, Alon Halfon, Yoav Kantor, Ran Levy, Yosi Mass, Ruty Rinott, Eyal Shnarch, Noam Slonim
IBM Haifa Reseacrh Lab
Haifa University, Mount Carmel, Haifa, HA 31905, Israel
{liate,alonhal,yoavka,ranl,yosimass,rutyr,eyals,noams}@il.ibm.com

Abstract

Wikipedia is a very popular source of encyclopedic knowledge which provides highly reliable articles in a variety of domains. This richness and popularity created a strong motivation among NLP researchers to develop relatedness measures between Wikipedia concepts. In this paper, we introduce WORD (Wikipedia Oriented Relatedness Dataset), a new type of concept relatedness dataset, composed of 19,276 pairs of Wikipedia concepts. This is the first human annotated dataset of Wikipedia concepts, whose purpose is twofold. On the one hand, it can serve as a benchmark for evaluating concept-relatedness methods. On the other hand, it can be used as supervised data for developing new models for concept relatedness prediction. Among the advantages of this dataset compared to its term-relatedness counterparts, are its built-in disambiguation solution, and its richness with meaningful multi-word terms. Based on this benchmark we develop a new tool, named WORT (Wikipedia Oriented Relatedness Tool), for measuring the level of relatedness between pairs of concepts. We show that the relatedness predictions of WORT outperform state of the art methods.

Keywords: Concept Relatedness, Relatedness function, Dataset

1. Introduction

Wikipedia is the leading open encyclopedia with very good coverage on diverse topics. Many terms encountered in human written texts are in fact Wikipedia concepts. As a result, multiple NLP tasks, such as entity linking and document clustering, can benefit from quantitatively measuring the level of relatedness between such concepts. A variety of methods have been proposed for measuring relatedness between Wikipedia concepts, e.g. (Witten and Milne, 2008; Strube and Ponzetto, 2006; Gabrilovich and Markovitch, 2007; Sherkat and Milios, 2017). Evaluating these methods, and developing new ones naturally requires high quality large benchmark data. Here, we introduce a new dataset, named WORD, composed of $19,276$ Wikipedia concept pairs, manually annotated to determine their level of relatedness.

We exploit these data to assess several types of state of the art semantic-relatedness tools, including word and document similarity functions (Mikolov et al., 2013; Gabrilovich and Markovitch, 2007), and link based methods (Witten and Milne, 2008; Ceccarelli et al., 2013), for the task of concept relatedness. Moreover, we suggest several new utilities, that are explicitly designed for this task, such as PMI (Church and Hanks, 1990; Bullinaria and Levy, 2007) measured between *concepts* based on their statistical co-occurrence in the entire Wikipedia corpus. We further exploit the data for supervised learning of concept-relatedness function, and use our novel utilities along with known state-of-the-art semantic-relatedness methods as features in a Linear Regression (LR) model. The resultant concept-relatedness tool, termed henceforth WORT, clearly outperforms each individual feature.

Finally, to demonstrate the versatility of WORD, we suggest a mechanism for automatically generating a disambiguated term-relatedness dataset from a Wikipedia concept-relatedness dataset. This mechanism, which does not involve additional human annotation, yields a new term-relatedness dataset containing $37,309$ pairs. This dataset, WORD, and the annotation guidelines are all available in `http://www.research.ibm.com/haifa/dept/vst/debating_data.shtml`.

2. Related Work

Most existing semantic relatedness datasets are composed of pairs of *words* (e.g. (Finkelstein et al., 2002; Hill et al., 2014)). Nevertheless, dataets with other element types, such as multi-word terms (Levy et al., 2015b), knowledge base concepts (Ceccarelli et al., 2013) and documents (Lee and Welsh, 2005) are also available.

From all term-relatedness datasets, TR9856 (Levy et al., 2015b) is probably the most similar to the current work, mainly due to its large size, and its generation process. The major difference between the two is in the type of elements composing them, Wikipedia concepts vs. terms, and thus in the relatedness task underlying their scores. Although part of the terms in TR9856 can be linked to concepts, the concepts are not available in the data, and were not provided to the annotators.

Another related work (Ceccarelli et al., 2013) uses links within Wikipedia documents to create a ranking dataset, under the assumption that a concept which is actually mentioned in a document, is more related to other concepts in the same document, than the other, false-positive candidates. Their work differs from ours in several aspects. Their interest in the relatedness task is directly linked to the task of entity disambiguation. Consequently, their goal is learning to rank, which affects their definition of related terms. However, for a variety of NLP tasks, measuring the actual relatedness between concepts, rather than ranking their relatedness, is of interest. We thus rely on human annotators to obtain an actual relatedness score between pairs of

concepts, instead of ranking pairs of concepts based on the aforementioned heuristics over Wikipedia.

3. Benchmark Generation

The benchmark generation process is composed of several stages.

3.1. Concept Pairs Selection

The objective of the Concept Pairs Selection stage is to create a balanced population of related and unrelated concept pairs. Clearly, a set of randomly selected pairs is expected to contain a low fraction of related pairs. To overcome this we follow (Levy et al., 2015b) who suggested a similar procedure for n-grams, and hypothesize that concepts which are over represented within a given article, tend to be related to one another. Thus, we first apply the TagMe (Ferragina and Scaiella, 2010) wikification tool, to identify concepts mentioned in the article, and then use the Hypergeometric (HG) test to create a Concept Lexicon (CL) per article, composed of all concepts with a HG p-value ≤ 0.05 after Bonferroni correction.

We selected 123 Wikipedia articles covering various topics like "affirmative action", "vegetarianism" and "atheism", and used the above mentioned procedure to generate 123 CLs. From each CL we created 160 concept pairs by randomly selecting 160 concepts from the CL, and pairing 40 of them with the article concept, and the rest 120 with another 120 concepts selected at random from the CL. For too small CLs, the maximal possible number of pairs was selected. In any case, we avoided selecting pairs of identical concepts.

3.2. Annotation

The above process yielded a total of $19,649$ concept pairs that were manually annotated via CrowdFlower, each by 10 annotators. The annotators were presented with a pair of URIs of Wikipedia articles, and were asked to mark them as "related", if they believe there is an immediate associative connection between them, and as "unrelated" otherwise. The annotators were further instructed to consider antonyms as related.

3.3. Post Processing

In order to increase the reliability of our dataset, we applied the following post processing procedure. We analyzed the inter-annotator agreement using Cohen's kappa coefficient, and filtered out annotators whose average kappa with the other annotators was smaller than 0.25. Annotators that did not have at least 50 common pairs with at least three other annotators were filtered out. The concept pairs that had less than 8 judgments after the annotators' filtering process, were removed from the benchmark.

3.4. Dataset Statistics

After filtering out some annotators we were left with 247 distinct annotators with average pairwise Cohen's kappa coefficient of 0.57. Removing concept pairs that were left with < 8 annotations, we ended up with a total of $19,276$ annotated concept pairs covering $10,871$ unique concepts. The relatedness score of each concept pair was computed by averaging over the binary answers of the annotators, yielding a relatedness score in the range $[0, 1]$. We refer to these data as WORD, standing for Wikipedia Oriented Relatedness Dataset.

WORD is enriched with related concepts, where more than 50% of the pairs have a positive score, and more than 15% have a score > 0.5. WORD is also enriched with multiword terms (MWTs). 66% of the concepts are MWTs, among which 65% are bigrams. WORD is split into training and test set, where $2/3$ of the topic articles, are in the training set, and the rest are in the test set. The information provided in the data includes the concept URIs and titles; the title of the article from which they were selected; whether they belong to the train or test set; and their relatedness annotation score.

3.5. Advantages Compared to Existing Datasets

Most existing term-relatedness datsets suffer from several issues (Batchkarov et al., 2016; Chiu et al., 2016; Faruqui et al., 2016) for which WORD provides at least partial solutions. To the best of our knowledge, this is the largest dataset of its type, whose number of pairs is large enough to yield high quality learning and evaluation results. Unlike most existing datasets, it is split into a training and a test set, providing a well defined framework for developing supervised relatedness methods, and for comparing between different relatedness functions. Furthermore, we use a cross validation setting, named LOTO, whose advantages are described in Section 4.. Moreover, as a concept-relatedness dataset, it is enriched with meaningful MWTs, unlike most standard term-relatedness datasets that include mostly unigrams, or noisy n-grams. Furthermore, the entities in WORD are disambiguated, since their meaning is provided by their Wikipedia article. As a result, the agreement within the annotators is rather high compared to the typical agreement in term-relatedness tasks. Finally, models that learn from WORD can utilize the large amount of structure data and side information of Wikipedia for improving relatedness predictions.

3.6. Term Relatedness Dataset Generation

We suggest a method for automatically generating a term-relatedness dataset from Wikipedia concept relatedness dataset. Our method is based on redirects and commonness of Wikipedia links, which are used to find, for each concept, a disambiguated and reliable equivalence set of n-grams associated with it. Then, given a concept pair $\{c_1, c_2\}$, these sets are used to create all possible n-gram pairs, and are assigned the score of $\{c_1, c_2\}$. This procedure is repeated for all the pairs in WORD, and results in a term-relatedness dataset of 37,309 pairs.

4. Wikipedia Oriented Relatedness Tool (WORT)

WORT is a tool for measuring relatedness between pairs of concepts. It receives two concepts as input, and returns a value in the range $[0, 1]$ indicating their predicted level of relatedness. The tool is based on a Linear Regression algorithm, which outperformed other models, including Nearest Neighbors. For model and feature selection we used a

Leave-One-Topic-Out (LOTO) cross validation setting on the training data, dividing the pairs into folds based on the article topic they originate from. This ensures that pairs included in the training set fold have little overlap with pairs associated with the left-out topic, hence avoiding a biased estimate of the model performance.

4.1. Model Features

WORT has 16 features which we classify into four categories, based on the type of information they rely on: *word-level* distribution, *concept-level* distribution, Wikipedia meta-data and article-text. The word-level distribution features treat a concept as the set of words composing its article title. Specifically, the relatedness between concepts is computed as follows. Let $r(x, y)$ be a word level relatedness measure between words x and y. We define the word-to-concept relatedness between a word x and a concept B to be $max_i r(x, y_i)$, where $\{y1, .., yn\}$ are the words of the article title of concept B. The relatedness between concept A and concept B $CC(A, B)$ is the average word-to-concept relatedness between the words in A and the concept B. The concept level relatedness between A and B is the average over the asymmetric functions $CC(A, B)$ and $CC(B, A)$. The word-level distribution features of the model are cosine similarity between Word2Vec(W2V) representation (Mikolov et al., 2013) and first order positive point mutual information (Church and Hanks, 1990), denoted by PMI1. Notice that word-level second order positive point mutual information (Bullinaria and Levy, 2007) was also computed, but was not included in the model since it did not contribute to its performance, probably due to its similarity to W2V (Levy et al., 2015a). The concept-level distribution features are novel relatedness methods, that adapt the word level measures to the level of concepts. These methods use concepts as basic units in the distribution computations. For examples, all appearances of the terms "U.S.", "U.S.A", "United States" etc. are treated as appearances of the concept "United States". This unification results in richer statistics about the associated terms. WORT includes the concept-level versions of PMI1, PMI2, and their normalized versions (Bouma, 2009), obtained based on applying TagMe (Ferragina and Scaiella, 2010) to the entire Wikipedia May 2017 dump. The Wikipedia meta-data features, rely on information such as the category and link structure. A thorough list of features from this group is provided in (Ceccarelli et al., 2013), most of which are included in WORT. Finally, WORT contains two *article-text* features, ESA (Gabrilovich and Markovitch, 2007), and the cosine similarity between the tf-idf vectors of the two articles (TFIDF-CS). Other features, such as the Jensen Shannon divergence between the bag-of-words representation of two articles, were omitted from the final model due to their high computational time and their negligible contribution to the overall performance. The full list of the 16 WORT features and their description appears in Table 1. We use $in(e)$ and $out(e)$ to denote the number of incoming and outgoing links of concept e.

word-level distribution features	
PMI1	word level first order positive PMI (Church and Hanks, 1990)
W2V	cosine similarity between the Word2Vec representations of the two concepts (Mikolov et al., 2013)
concept-level distribution features	
NPMI1_C	normalized concept-level first order positive PMI1
PMI1_C	concept level normalized PMI1
PMI2_C	cosine similarity between the PMI1 values of each concept with the 10000 most frequent concepts
Wikipedia meta-data features	
INLINK-PMI	normalized PMI between the in links of the concepts (Ceccarelli et al., 2013)
MW	co-citation based similarity (Witten and Milne, 2008)
COND-PROB	$\frac{1}{2}(\frac{in(a)\cap in(b)}{in(b)} + \frac{in(a)\cap in(b)}{in(a)})$
J-IN	Jaccard similarity: $\frac{in(a)\cap in(b)}{in(a)\cup in(b)}$
J-OUT	Jaccard similarity considering the outgoing links
J-IN-OUT	Jaccard similarity considering the union of incoming and outgoing links
FRIEND	1 if a links to b and b links to a,
DIRECT-LINK	1 if $b \in in(a) \& a \in in(b)$, 0 otherwise
CATEGORY	1 if the two concepts share a Wikipedia category, 0 otherwise
article-text features	
TFIDF-CS	Cosine similarity between the tf-idf vectors of the two concept articles
ESA	Cosine similarity between ESA representation of the concept aricles

Table 1: Full list of WORT features

5. Results

We used the test set of WORD to evaluate the performance of the different features in comparison to WORT. We computed both Spearman and Pearson correlations, to receive a wider picture of the relative strengths and weaknesses of the different features. The strongest features, according to Pearson correlation, are presented in Figure 1. As shown in the figure, each of the four feature categories mentioned in 4.1. has a representative within the five leading features. The strongest feature is the *text-based* feature, TFIDF-CS, which is followed by the *word-level distribution* features, PMI1 and W2V. Next, the novel concept-level normalized positive PMI1 (NPMI1_C) leads the *concept-level distri-*

bution features, while INLINK-PMI, which computes the PMI between the incoming links of the two concepts (Ceccarelli et al., 2013), leads the *Wikipedia meta-data* features. To examine the statistical significance of the obtained feature ranking, we used the MRDS method (Rastogi et al., 2015), which given a pair of features, computes the significance of the difference between their prediction capabilities, based on the difference between their observed correlations, the correlation between them, and the size of the data. Applying this test to the pairs within the top ranked features, the advantage of PMI1 over W2V and INLINK-PMI over PMI1_C was shown to be insignificant. The relative ranking of the other feature pairs was statistically significant, with p-value < 0.05. We further applied MRDS to check the advantage of the highest ranked feature from each category, over the other members of its category. The advantage of PMI1 over W2V was already shown to be insignificant. However, the advantage in the three other feature categories was significant, with p-value < 0.05.

Sorting the features by their $Spearman$ correlation results in a slightly different feature hierarchy. The three leading features remain unchanged, but the concept-level features are replaced in the top list with link-structure features. This result implies that the link based features are stronger in concept ranking then in prediction of relatedness level, where the opposite is true for the *concept-level distribution* features.

WORT improvement of Spearman and Pearson correlation compared to the best feature (TFIDF-CS), is 16% and 13% respectively. This improvement is significant for both Spearman and Pearson correlation with MRDS p-value $< 10^{-308}$.

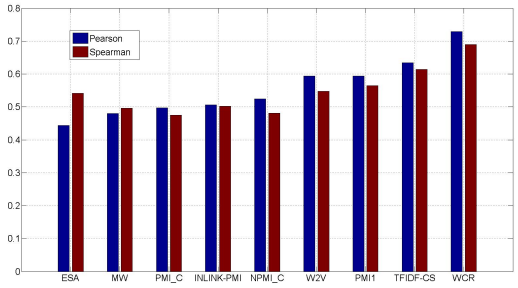


Figure 1: Pearson (blue) and Spearman (red) correlations of WORT (right most bars), and of the highest ranked WORT features according to Pearson correlation. The feature names appear below the bars.

6. Discussion and Future Work

We propose a novel type of semantic relatedness dataset, composed of pairs of Wikipedia concepts manually scored by human annotators. We display its usefulness for evaluating different families of relatedness functions, and study their relative performance. To demonstrate the versatility of WORD, we introduce a mechanism for automatically converting it into an even larger disambiguated term-relatedness dataset. Furthermore, by using the documents of the concept articles, WORD can serve also as a document relatedness dataset which is dramatically larger than the existing dataset (Lee and Welsh, 2005). From the supervised learning point of view, the performance of our new tool, WORT, which despite its simplicity, yields a relatedness function that significantly improves state of the art methods, demonstrates the high synergistic potential that lies in the diversity of the participating features. More complex models can be learned for further improving the relatedness function. For example, one can adopt the approach used by (Mueller and Thyagarajan, 2016) for metric learning of sentences, and use Siamese networks for learning concept representations and a metric between them. Finally, the concept pairs identified as related in WORD probably manifest various types of relatedness, including topical relations, functional relations, and so forth. More work is needed to characterize these relations types and their relative frequency in the data; different prediction methods could then be developed, specializing in predicting specific types of relations between concept pairs.

7. Acknowledgements

The authors would like to thank Vered Shwartz for her contribution to the preliminary stages of this work.

8. Bibliographical References

Batchkarov, M., Kober, T., Reffin, J., Weeds, J., and Weir, D. (2016). A critique of word similarity as a method for evaluating distributional semantic models. *ACL 2016*, page 7.

Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. In *Proceedings of the Biennial GSCL Conference*, volume 156.

Bullinaria, J. A. and Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior research methods*, 39(3):510–526.

Ceccarelli, D., Lucchese, C., Orlando, S., Perego, R., and Trani, S. (2013). Learning relatedness measures for entity linking. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 139–148. ACM.

Chiu, B., Korhonen, A., and Pyysalo, S. (2016). Intrinsic evaluation of word vectors fails to predict extrinsic performance. *ACL 2016*, page 1.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. *arXiv preprint arXiv:1605.02276*.

Ferragina, P. and Scaiella, U. (2010). Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *CIKM*.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Gabrilovich, E. and Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611.

Hill, F., Reichart, R., and Korhonen, A. (2014). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *arXiv preprint arXiv:1408.3456*.

Lee, M. D. and Welsh, M. (2005). An empirical evaluation of models of text document similarity. In *In CogSci2005*, pages 1254–1259. Erlbaum.

Levy, O., Goldberg, Y., and Dagan, I. (2015a). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Levy, R., Ein-Dor, L., Hummel, S., Rinott, R., and Slonim, N. (2015b). Tr9856: A multi-word term relatedness benchmark. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 419–424, Beijing, China, July. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mueller, J. and Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *AAAI*, pages 2786–2792.

Rastogi, P., Van Durme, B., and Arora, R. (2015). Multi-view lsa: Representation learning via generalized cca. In *HLT-NAACL*, pages 556–566.

Sherkat, E. and Milios, E. (2017). Vector embedding of wikipedia concepts and entities. *arXiv preprint arXiv:1702.03470*.

Strube, M. and Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, volume 6, pages 1419–1424.

Witten, I. and Milne, D. (2008). An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, Chicago, USA*, pages 25–30.

Experiments with Convolutional Neural Networks for Multi-Label Authorship Attribution

Dainis Boumber[*], **Yifan Zhang**[*], **Arjun Mukherjee**
Department of Computer Science, University of Houston
Philip Guthrie Hoffman Hall, 3551 Cullen Blvd., Room 501, Houston, TX 77204-3010, USA
{dboumber, yzhang114}@uh.edu, arjun@cs.uh.edu

Abstract

We explore the use of Convolutional Neural Networks (CNNs) for multi-label Authorship Attribution (AA) problems and propose a CNN specifically designed for such tasks. By averaging the author probability distributions at sentence level for the longer documents and treating smaller documents as sentences, our multi-label design adapts to single-label datasets and various document sizes, retaining the capabilities of a traditional CNN. As a part of this work, we also create and make available to the public a multi-label Authorship Attribution dataset (MLPA-400), consisting of 400 scientific publications by 20 authors from the field of Machine Learning. Proposed Multi-label CNN is evaluated against a large number of algorithms on MLPA-400 and PAN-2012, a traditional single-label AA benchmark dataset. Experimental results demonstrate that our method outperforms several state-of-the-art models on the proposed task.

Keywords: multi-label authorship attribution, convolutional neural networks, datasets

1. Introduction

Authorship Attribution uses textual features to distinguish between texts written by different authors (Stamatatos, 2009). A typical AA problem is a single-label text-categorization task: given a set of candidate authors for whom text samples of undisputed authorship are available, a text is assigned to one candidate author (Sebastiani and Ricerche, 2002; Stamatatos, 2009). Less common yet equally important is the case that involves identifying multiple co-authors of a document. This is the problem of multi-label AA and as we will see it is significantly harder than classical AA. We briefly review classical AA and then introduce our work in multi-label AA.

Classification algorithms utilizing lexical, semantic, syntactic, stylistic, and character n-gram features have been explored by Graham et al. (2005), Gamon (2004), Sapkota et al. (2015), and Shrestha et al. (2017). Qian et al. (2014) proposed a tri-training method to solve AA under limited training data per author. It extended standard co-training using three views: lexical, character and syntactic and was shown to have better generalization performance. This method assumes that a large set of unlabeled documents authored by the same given closed set of authors is available. Sapkota et al. (2016) leveraged Domain Adaptation in an AA scenario where articles on different topics may be written by the same author and labeled training data is limited. The method introduced was a modification of Structural Correspondence Learning (Blitzer et al., 2006) and requires a large set of unlabeled documents pertaining to the target domain and written by the same authors. Seroussi et al. (2012) used latent topic features to improve attribution. Although useful, it requires a large text collection per author. AA via text distortion has been used on traditional PAN corpora (Stamatatos, 2017). AA with a large number of authors and limited training data has been studied by Luyckx and Daelemans (2008). A lazy memory-based learner based on k-NN was shown to work well with combinations of features. Ruder et al. (2016) used character level and hybrid multi-channel Deep Neural Networks for large scale multi-author AA to a great degree of success; however, their work was concerned with classical single-label scenarios only.

We propose a CNN for multi-label AA tasks. Our design treats a document as a set of sentences where each one has multiple labels. Individual authorship of continuous sections is taken into account together with the possibility of co-authors influencing each other's style or editing passages written by others. We name this strategy *collaborative section attribution*. Multi-label CNN utilizes depth-wise convolutions for the separate processing of the two input channels, capturing information unique to each, which helps filters activate on more relevant inputs. We conduct a series of experiments using our model, the recently successful version of CNN by Kim (2014) and a large number of baselines. Proposed Multi-label CNN outperforms the competition by a significant margin on multi-label data (MLPA-400) and matches or defeats relevant baselines on single-label tasks (PAN 2012).

For evaluation, we consider a realistic problem of multi-label AA in the realm of scientific publications by creating a publicly available dataset consisting of 400 Machine Learning papers, Machine Learning Papers' Authorship 400 (MLPA-400)[1]. To the best of our knowledge, multi-label AA of scientific publications has not received a lot of attention. It deserves more attention because automatic resolution of authorship issues in papers can have a variety of downstream applications in intellectual property managements, citation analysis, archival systems, and author disambiguation. The task is challenging: papers have many authors whose writing style can evolve or influenced

[*]The first two authors contributed equally to the work but are named alphabetically.

[1]https://github.com/dainis-boumber/AA_CNN/wiki/MLPA-400-Dataset

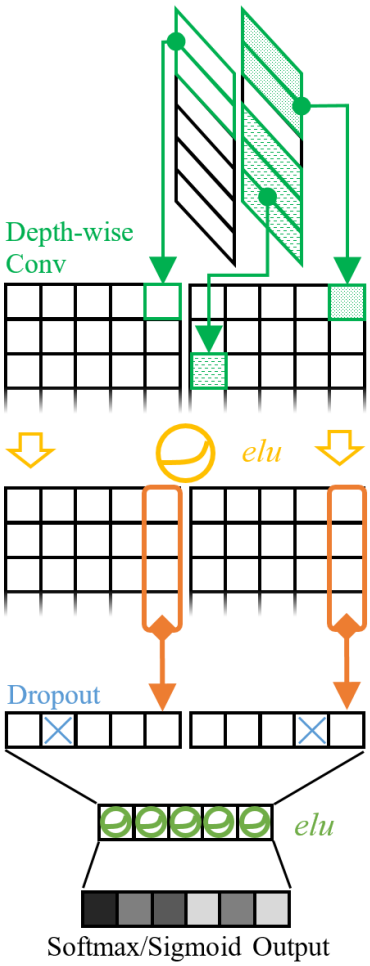


Figure 1: The architecture of proposed multi-label CNN.

by colleagues, they contain direct quotes from other works, authors' contribution to the paper in terms of the amount of text written is unknown; the number of papers and authors is large.

The contribution of our work is threefold: a CNN that employs collaborative section attribution and separate channels in depth-wise convolutions, a novel real-world multi-label MLPA-400 corpus from top cited ML authors for an AA scenario, a thorough performance evaluation of the proposed algorithm with relevant baselines on the new MLPA-400 dataset and PAN-2012.

2. Network Hierarchy

At a high level, our design is a multilayer CNN that either computes a probability distribution for an entire document (single-label problems) or an average of probability distributions over individual sentences of a document (multi-label problems). Attributing multi-author work sentence by sentence intuitively makes sense, because co-authors typically write different sections of the paper.

2.1. CNN Architecture

Let the document to be classified be $\boldsymbol{D}$. Each $\boldsymbol{D}$ consists of $|S|$ number of sentences and each sentence consists of $|T|$ words. We classify each $\boldsymbol{S}$ and take the mean to obtain predicted label of $\boldsymbol{D}$. To allow the network to generate consistent dimensions, we pad sentences to the same length $|T| = 128$. Documents are padded to same number of sentences $|S| = 128$. The ground truth label associated with each document is denoted using vector $\boldsymbol{y}$. Then $|\boldsymbol{y}|$ is equal to the cardinality of the set of possible authors. An element of $\boldsymbol{y}$ is marked 1 if the corresponding person is one of the authors and 0 otherwise.

To get an edge over Kim (2014), we use fixed word2vec (Mikolov et al., 2013) and Glove (Pennington et al., 2014) embeddings, leveraging Glove's superior general performance (Pennington et al., 2014) and word2vec's advantage when it comes to rare words or symbols (Shazeer et al., 2016). Words in $\boldsymbol{D}$ are mapped into pre-trained word2vec and Glove embedding space by replacing each word with a corresponding row in the embedding matrix $\boldsymbol{X}_{word2vec}$ and $\boldsymbol{X}_{glove}$, where both have $\boldsymbol{X} \in \mathbb{R}^{|V| \times n}$, assuming a total vocabulary count of $|V|$ and embedding dimension n. Hence replacing all word features in a sentence with one of the embedding matrices will transform the sentence into a matrix $\boldsymbol{S} \in \mathbb{R}^{|T| \times n}$.

To capture information that is distinctive to each embedding space, we help filters activate on relevant inputs by using depth-wise two-dimensional convolutions which process input from word2vec and Glove channels separately. In spirit with Kim (2014), multiple filters of multiple sizes h are used to extract features from $\boldsymbol{S}$. Unlike Kim, we use 100 filters for each window size $h \in \{1, 2, 3, 4, 5\}$ to let smaller filters pick up on simple stylistic features present, such as words unique to one author. Applying a filter $\boldsymbol{f} \in \mathbb{R}^{h \times n}$ on one of the input channels at word window i to $(i + h - 1)$ amounts to

$$c_i^f = elu\left(\boldsymbol{f} \cdot \boldsymbol{S}_{[i:i+h-1]} + b^f\right) \tag{1}$$

where b^f is the bias term corresponding to filter f and $elu(x)$ is the *exponential linear unit* (ELU). ELUs have negative values which pushes the mean of the activations closer to zero, resulting in faster training and lower variance. The positive part of these functions is the identity; their derivative is one and not contractive, thus the vanishing gradient problem is alleviated (Clevert et al., 2015).

We denote the number of filter sizes as $|\boldsymbol{h}|$ and the number of filters of each size as q. For $|\boldsymbol{h}| = 5$ and $q = 100$, we would have a total of 500 filters generating 500 features at each word location. By concatenating feature values generated by each filter into a vector, a total of 500 vectors $\boldsymbol{C}^f$s are generated for each sentence in each embedding channel.

$$\boldsymbol{C}^f = \left[c_1^f, c_2^f, \ldots c_{|T|-h+1}^f\right] \tag{2}$$

All vectors are batch-normalized (BN) to reduce overfitting (Ioffe and Szegedy, 2015).

Each $\boldsymbol{C}^f$ generated by the convolutional layer is max-pooled to keep only the largest value out of all the values across a sentence generated by one filter:

$$\hat{c}^f = max\left\{\boldsymbol{C}^f\right\} \tag{3}$$

All values resulting from max-pooling in both embedding channels are concatenated into a vector $\boldsymbol{o} \in \mathbb{R}^{|\boldsymbol{h}| \cdot q \cdot 2}$. Dropout (Srivastava et al., 2014) at the rate of 0.5 is then applied to $\boldsymbol{o}$ resulting in a vector $\hat{\boldsymbol{o}}$.

2.2. Predictions

Very short documents are unlikely to have more than one author. If the average document in the training corpus is shorter than the maximum sentence length $|T| = 128$, $\hat{\boldsymbol{o}}$ is passed to a softmax output layer for single-label classification. Otherwise, we pass $\hat{\boldsymbol{o}}$ to a fully connected sigmoid output layer of size $|\boldsymbol{y}|$, with one element per candidate author. For single-label tasks, *softmax cross entropy loss* was used. For multi-label tasks the loss used is *sigmoid cross-entropy*. We average resulting values for each sentence to determine the final document level result. The code can be obtained at `https://github.com/dainis-boumber/AA_CNN`

3. Datasets

3.1. MLPA-400

3.1.1. Considerations

Many approaches to creating a suitable corpus exist. For example, papers can be chosen across domains. However, even within one domain the stylistic differences between venues are significant enough to make individual style hard to detect. A random sample of authors can be taken, but the number of multi-labeled documents would be few. Another possibility is taking the transitive closure of the set of co-authors and extracting at least k papers per author. However, creation of such a dataset for any reasonable k results in a very large transitive set.

3.1.2. Design

Using Google Scholar as a source, we created a list of top 20 authors in Machine Learning, ranked by the number of citations. We ensured a reasonable number of papers had an overlap of authors (i.e., we also included papers that were jointly authored by the set of authors). For each author, 20 papers were downloaded for a total of 400 publications for the entire dataset. Each work is assigned 20 binary labels. The labels indicate which of the authors contributed to the paper's creation. 100 papers out of 400 have more than one author from the 20 listed. The number of authors ranged from 1 to 3 and the average was 1.2925. The text was extracted from the PDF files using pdfminer (Hinyama, 2017) and pre-processed. The title, authorship information, and bibliography fields were removed from each paper to ensure the classifier abides by the rules of blind review instead of simply using author list while learning authorship. Formulas, table and figure captions were retained as they may contain valuable author specific style and topic information. The dataset is available at `https://github.com/dainis-boumber/AA_CNN/wiki/MLPA-400-Dataset`

3.2. PAN-2012

For classical AA, we use the PAN-2012 (Juola, 2012) corpus and report performances on its 3 tasks: A, C and I. Their training sets consist of 2 documents per author. The test sets have 1 text per author, except Task A which has 2 texts from 3 authors, 800 to 6060 words each. Task C has 8 authors; the texts are larger, up to 13000 words long. Task I has 14 authors, with documents ranging from approximately 40,000 to 170,000 words. Further details on this data is available in (Juola, 2012).

4. Experiments

4.1. Baselines

Our method was tested against a wide array of baselines. We used n-grams with n ranging from 1 to 5 words and 1 to 8 characters. We experimented with TFIDF, hashing, count vectorization, binary bow model and doc2vec (Le and Mikolov, 2014). The resulting document vectors were used as inputs to the baseline classifiers: Support Vector Machine (SVM) (Cortes and Vapnik, 1995), Gaussian and Multinomial Naive Bayes (GNB and NB), Decision Tree with AdaBoost, Gradient Boosting, Random Forest, K Nearest Neighbors, Multilayer Perceptron (MLP) (Rumelhart et al., 1986), and Logistic Regression. We varied the hyper-parameters of the algorithms in order to achieve the best result (we vary kernels, the margin and the penalty between L_1, L_2, and $L_1 + L_2$ for hinge loss SVM with SGD; breadth and number of layers for MLP, etc.). The baselines were implemented using the scikit-learn library (Pedregosa et al., 2011).

We implemented CNN-non-static, a sentence classification approach proposed by Kim (2014) that has recently been successful in text classification. It initializes embedding layer with pre-trained word2vec vectors (Mikolov et al., 2013) and optimizes the embedding layer together with the rest of network's parameters during training. The CNN consists of one convolutional layer that samples from the input using multiple window sizes, a max over time pooling layer, and a fully connected output layer. Dropout (Srivastava et al., 2014) and L_2 are used for regularization.

The MLPA-400 problem is a multi-label task and cannot be directly solved by most classifiers. We used One vs. Rest Classification approach, fitting one binary classifier per class. We then associated a set of positive examples for a given class and a set of negative examples which represent all the other classes present within the training folds. No class balancing was performed to retain the natural class distribution in the data. As One vs Rest Classification scheme requires a separate model for each author, it is not feasible to use it with CNN-non-static on MLPA-400. Instead, we augmented CNN-non-static with the multi-label modification described in section 2.2. We evaluated an almost exhaustive combination of models with a total of 16 classifiers and 4 vectorizers that employ n-grams, with the maximum n equal to 5 for words and 8 for characters. This resulted in 16x4x5 + 16x4x8 or 832 baselines. We tested the CNN non-static and used doc2vec embeddings for each classifier, giving 33 more for a total of 865. After varying the hyper-parameters for those algorithms that allowed it, the total number of models was 1685.

For PAN-2012, the top performing teams results' in the AA challenge are also considered beyond the 1685 baseline models we trained.

4.2. Metrics

For the MLPA-400 data we use accuracy, micro and macro F1, Jaccard index and Hamming loss. In PAN-2012 exper-

Baseline	F1 macro	F1 micro	Jaccard index	Hamming loss	Accuracy
multi-label CNN (word2vec+glove)	0.736	0.744	**0.713**	0.031	**65.3%**
CNN-non-static (Kim, 2014)	0.685	0.695	0.664	0.037	60.8%
K Nearest Neighbors (binary 3-gram)	0.666	0.737	0.633	0.033	52.0%
Perceptron (sgd, l^2, binary 3-gram)	**0.748**	**0.751**	0.591	**0.030**	51.0%
SVM (sgd, squared hinge loss, l^2, binary 1-gram)	0.681	0.690	0.516	0.034	45.3%
MLP (doc2vec)	0.569	0.640	0.461	0.041	40.0%

Table 1: ML Papers evaluation results.

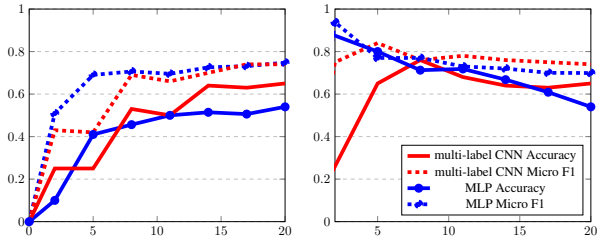


Figure 2: Varying the number of papers (left) and authors (right) on accuracy and micro-F1.

iments, we follow the instructions provided by the creators of the challenge and focus only on accuracy.

Exact label accuracy for multi-label problems is a particularly unforgiving metric: incorrectly attributing a single author from what can be a long list of co-authors results in the paper being marked as incorrectly classified. For this reason we also report Hamming loss and Jaccard index (also known as similarity coefficient), which are common metrics for multi-label tasks. Hamming loss is the fraction of labels that are incorrectly predicted. Jaccard index is the size of the intersection of the ground truths and the predictions divided by the size of their union.

5. Results and Discussion

5.1. MLPA-400

On this dataset, our approach is significantly more accurate, as seen in Table 1 yielding a confidence of $p = 0.0828$ against the next top competitor CNN non-static using a paired t-test across five-fold cross validation. Next, binary vectorization dominates all simple character and word vectorization approaches (not shown in Table 1 as they fall below top-500). Character n-grams perform poorly.

5.1.1. Effect of the number of Documents/Author

To discover the correlation between the amount of training data and the performance of our multi-label CNN, we explore the effect of the number of papers per author on accuracy and micro F1. The latter was chosen because it takes into account class and label balance. The # of papers per author was set to 2, then increased to 5, 8, 11, 14, 17 and 20. We compare multi-label CNN and MLP in Figure 2 (left). As the amount of training data (# of papers per author) available increases, so does the algorithms' ability to generalize. The improvement becomes minimal past 11 papers, but continues to increase. Because overfitting is reduced with the increased amount of training data (Brain and Webb, 1999), increasing the size of the dataset can benefit CNNs and other powerful models, allowing for deeper architectures that may discover new style features.

5.1.2. Effect of the number of Authors

To determine the effect of the number of authors on prediction accuracy, the number of papers was fixed at 20 per authors and the number of authors was varied between 2, 5, 8, 11, 14, 17 and 20. In Figure 2 (right), performance grows with the amount of training data, then decreases as the problem gets difficult due to the number of labels. Similar situations were also observed in Luyckx and Daelemans (2008) who recommend enriching the feature space by using combinations of features and employing a lazy learner.

5.2. PAN-2012

Task A: Our method, most baselines, and most competitors such as Sapkota and Solorio (2012) tied at 100% accuracy[2].

Task C: Our approach results in 100% classification accuracy. Only three of the challenge participants attain the same level of success (Grozea and Popescu, 2012; Sapkota and Solorio, 2012; Giraud and Artières, 2012). CNN-non-static performs very poorly, correctly classifying only 3 documents. SVM and NB achieve 87.5%, with other baselines falling far behind.

Task I: Kim-non-static and our method tie the state-of-the-art (Grozea and Popescu, 2012; Sapkota and Solorio, 2012; Tanguy et al., 2012) with the accuracy of 92.86%. SVM and NB score 85.71% and are on par with most of the contestants. Remaining baselines fall short.

5.3. Sensitivity of CNN Parameters

We found that using L_2 regularization improved training speed at the expense of accuracy. Additional convolutional layers failed to produce any effect.

6. Conclusion

This paper presented a CNN architecture designed to address multi-label Authorship Attribution problems. To test our design in non-traditional AA environment and alleviate the lack of relevant corpora, we created and made available to the public MLPA-400 — a dataset consisting of publications from well-known researchers. Experimental results show our method significantly outperforming the competition in a multi-label scenario and matching or surpassing state-of-the-art on traditional AA tasks.

Acknowldgments This work is supported in part by NSF 1527364.

[2]PAN-2012 results: http://pan.webis.de/clef12/pan12-web/pan12-authorship-attribution-evaluation-results.xlsx

7. Bibliographical References

Blitzer, J., McDonald, R., and Pereira, F. (2006). Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 120–128, Stroudsburg, PA, USA. Association for Computational Linguistics.

Brain, D. and Webb, G. I. (1999). On the effect of data set size on bias and variance in classification learning. In D. Richards, et al., editors, *Proceedings of the Fourth Australian Knowledge Acquisition Workshop (AKAW '99)*, pages 117–128, Sydney. The University of New South Wales.

Clevert, D., Unterthiner, T., and Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). *CoRR*, abs/1511.07289.

Cortes, C. and Vapnik, V. (1995). Support vector networks. *Machine Learning*, 20:273–297.

Gamon, M. (2004). Linguistic correlates of style: authorship classification with deep linguistic analysis features. In *COLING*.

Giraud, F.-M. and Artières, T. (2012). . In Pamela Forner, et al., editors, *CLEF 2012 Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy*, September.

Graham, N., Hirst, G., and Marthi, B. (2005). Segmenting documents by stylistic character. *Natural Language Engineering*, 11(4):397–415.

Grozea, C. and Popescu, M. (2012). Encoplot - Tuned for High Recall (also proposing a new plagiarism detection score). In Pamela Forner, et al., editors, *CLEF 2012 Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy*, September.

Hinyama, Y. (2017). Pdfminer: Python pdf parser and analyzer. `http://www.unixuser.org/~euske/python/pdfminer/`.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167.

Juola, P. (2012). An overview of the traditional authorship attribution subtask. In Pamela Forner, et al., editors, *CLEF (Online Working Notes/Labs/Workshop)*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Luyckx, K. and Daelemans, W. (2008). Authorship attribution and verification with many authors and limited data. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 513–520, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Qian, T., Liu, B., Chen, L., and Peng, Z. (2014). Tri-training for authorship attribution with limited training data. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 345–351, Baltimore, Maryland, June. Association for Computational Linguistics.

Ruder, S., Ghaffari, P., and Breslin, J. G. (2016). Character-level and multi-channel convolutional neural networks for large-scale authorship attribution. *CoRR*, abs/1609.06686.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chapter Learning Internal Representations by Error Propagation, pages 318–362. MIT Press, Cambridge, MA, USA.

Sapkota, U. and Solorio, T. (2012). Sub-Profiling by Linguistic Dimensions to Solve the Authorship Attribution Task—Notebook for PAN at CLEF 2012. In Pamela Forner, et al., editors, *CLEF 2012 Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy*, September.

Sapkota, U., Bethard, S., y Gomez, M. M., and Solorio, T. (2015). Not all character n-grams are created equal: A study in authorship attribution. In Rada Mihalcea, et al., editors, *HLT-NAACL*, pages 93–102. The Association for Computational Linguistics.

Sapkota, U., Solorio, T., Gomez, M. M., and Bethard, S. (2016). Domain adaptation for authorship attribution: Improved structural correspondence learning. In *ACL (1)*. The Association for Computer Linguistics.

Sebastiani, F. and Ricerche, C. N. D. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34:1–47.

Seroussi, Y., Bohnert, F., and Zukerman, I. (2012). Authorship attribution with author-aware topic models. In *ACL (2)*, pages 264–269. The Association for Computer Linguistics.

Shazeer, N., Doherty, R., Evans, C., and Waterson, C. (2016). Swivel: Improving embeddings by noticing what's missing. *CoRR*, abs/1602.02215.

Shrestha, P., Sierra, S., Gonzalez, F., Montes, M., Rosso, P., and Solorio, T. (2017). Convolutional neural networks for authorship attribution of short texts. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 669–674. Association for Computational Linguistics.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Stamatatos, E. (2009). A survey of modern authorship at-

tribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556.

Stamatatos, E. (2017). Authorship attribution using text distortion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1138–1149, Valencia, Spain, April. Association for Computational Linguistics.

Tanguy, L., Sajous, F., Calderone, B., and Hathout, N. (2012). Authorship Attribution: Using Rich Linguistic Features when Training Data is Scarce. In *PAN Lab at CLEF*, pages –, Rome, Italy, September.

A Fast and Accurate Vietnamese Word Segmenter

Dat Quoc Nguyen[1], **Dai Quoc Nguyen**[2], **Thanh Vu**[3], **Mark Dras**[4], **Mark Johnson**[4]
[1]The University of Melbourne, Australia; [2]Deakin University, Australia;
[3]Newcastle University, United Kingdom; [4]Macquarie University, Australia
dqnguyen@unimelb.edu.au, dai.nguyen@deakin.edu.au,
thanh.vu@newcastle.ac.uk, {mark.dras, mark.johnson}@mq.edu.au

Abstract

We propose a novel approach to Vietnamese word segmentation. Our approach is based on the Single Classification Ripple Down Rules methodology (Compton and Jansen, 1990), where rules are stored in an exception structure and new rules are only added to correct segmentation errors given by existing rules. Experimental results on the benchmark Vietnamese treebank show that our approach outperforms previous state-of-the-art approaches JVnSegmenter, vnTokenizer, DongDu and UETsegmenter in terms of both accuracy and performance speed. Our code is open-source and available at: https://github.com/datquocnguyen/RDRsegmenter.

Keywords: Vietnamese, Word segmentation, Single classification ripple down rules

1. Introduction

Word segmentation is referred to as an important first step for Vietnamese NLP tasks (Dien et al., 2001; Ha, 2003; Duc Cong et al., 2016). Unlike English, white space is a weak indicator of word boundaries in Vietnamese because when written, it is also used to separate syllables that constitute words. For example, a written text "thuế thu nhập cá nhân" (individual$_{\text{cá_nhân}}$ income$_{\text{thu_nhập}}$ tax$_{\text{thuế}}$) consisting of 5 syllables forms a two-word phrase "thuế_thu_nhập cá_nhân."[1] More specifically, about 85% of Vietnamese word types are composed of at least two syllables and 80%+ of syllable types are words by themselves (Thang et al., 2008; Le et al., 2008), thus creating challenges in Vietnamese word segmentation (Nguyen et al., 2012).

Many approaches are proposed for the Vietnamese word segmentation task. Le et al. (2008), Pham et al. (2009) and Tran et al. (2012) applied the maximum matching strategy (NanYuan and YanBin, 1991) to generate all possible segmentations for each input sentence; then to select the best segmentation, Le et al. (2008) and Tran et al. (2012) used n-gram language models while Pham et al. (2009) employed part-of-speech (POS) information from an external POS tagger. In addition, Nguyen et al. (2006), Dinh and Vu (2006) and Tran et al. (2010) considered this segmentation task as a sequence labeling task, using either a linear-chain CRF, SVM or MaxEnt model to assign each syllable a segmentation tag such as B (Begin of a word) or I (Inside of a word). Another promising approach is joint word segmentation and POS tagging (Takahashi and Yamamoto, 2016; Nguyen et al., 2017b), which assigns a combined segmentation and POS tag to each syllable. Furthermore, Luu and Kazuhide (2012), Liu and Lin (2014) and Nguyen and Le (2016) proposed methods based on pointwise prediction (Neubig and Mori, 2010), where a binary classifier is trained to identify whether or not there is a word boundary between two syllables.

In this paper, we propose a novel method to Vietnamese word segmentation. Our method automatically constructs a Single Classification Ripple Down Rules (*SCRDR*) tree (Compton and Jansen, 1990) to correct wrong segmentations given by a longest matching-based word segmenter. On the benchmark Vietnamese treebank (Nguyen et al., 2009), experimental results show that our method obtains better accuracy and performance speed than the previous state-of-the-art methods JVnSegmenter (Nguyen et al., 2006), vnTokenizer (Le et al., 2008), DongDu (Luu and Kazuhide, 2012) and UETsegmenter (Nguyen and Le, 2016).

2. SCRDR methodology

This section gives a brief introduction of the SCRDR methodology (Compton and Jansen, 1988; Compton and Jansen, 1990; Richards, 2009). A SCRDR tree is a binary tree with only two unique types of edges "except" and "if-not", where every node is associated with a *rule* in a form of "*if* **condition** *then* **conclusion**." To ensure that the tree always produces a conclusion, the rule at its root (default) node has a trivial condition which is always satisfied.

[1]In the traditional underscore-based representation in the Vietnamese word segmentation task (Nguyen et al., 2009), white space is only used to separate words while underscore is used to separate syllables inside a word.

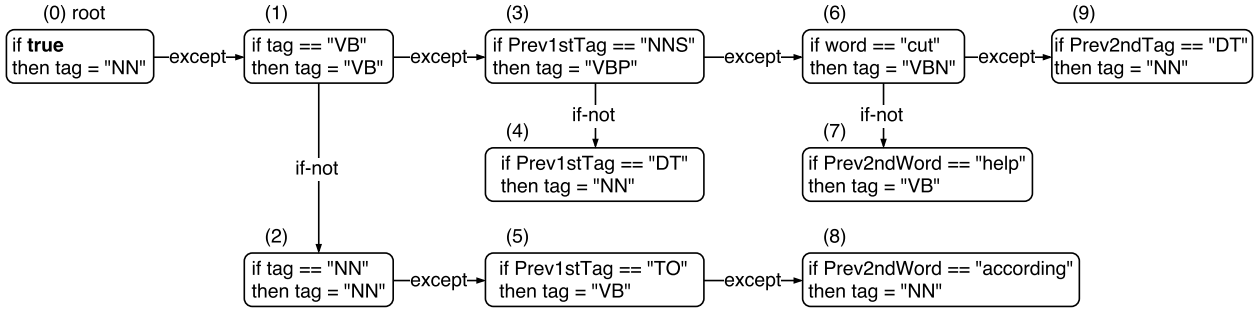


Figure 1: An illustration of a SCRDR tree for POS tagging. This figure is adapted from Nguyen et al. (2016).

Each case to be evaluated starts at the root node and ripples down as follows: (i) If the case satisfies the condition of a current node's rule, the case is then passed on to the current node's "except" child if this "except" child exists. (ii) Otherwise, if the case does not satisfy the condition, it is then passed on to the current node's "if-not" child. So, the conclusion returned by the tree is the conclusion of the last satisfied rule in the evaluation path to a leaf node.

For example, Figure 1 illustrates a SCRDR tree for POS tagging. Let us consider a concrete case "*as/IN investors/NNS anticipate/VB a/DT recovery/NN*" where "*anticipate*" and "*VB*" is the current considered pair of word and its initial POS tag. Because this case satisfies the conditions of the rules at nodes (0), (1) and (3), it is passed on to node (6) using the "except" edge. The case does not satisfy the condition of the rule at node (6), thus it is passed on to node (7) using the "if-not" edge. Finally, the case does not satisfy the condition of the rule at the leaf node (7). So, the rule at node (3)—the last satisfied rule in the the evaluation path (0)-(1)-(3)-(6)-(7)—concludes "*VBP*" should be the POS tag of the word "*anticipate*" instead of the initial POS tag "*VB*."

To correct a wrong conclusion returned for a given case, a new node containing a new *exception* rule may be attached to the last node in the evaluation path. If the last node's rule is the last satisfied rule given the case, the new node is added as its child with the "except" edge; otherwise, the new node is attached with the "if-not" edge.

SCRDR has been successfully applied in NLP tasks for temporal relation extraction (Pham and Hoffmann, 2006), word lemmatization (Plisson et al., 2008), POS tagging (Xu and Hoffmann, 2010; Nguyen et al., 2011b; Nguyen et al., 2014; Nguyen et al., 2016), named entity recognition (Nguyen and Pham, 2012) and question answering (Nguyen et al., 2011a; Nguyen et al., 2013; Nguyen et al., 2017a). The works by Plisson et al. (2008), Nguyen et al. (2011b), Nguyen et al. (2014) and Nguyen et al. (2016) build the tree automatically, while others manually construct the tree.

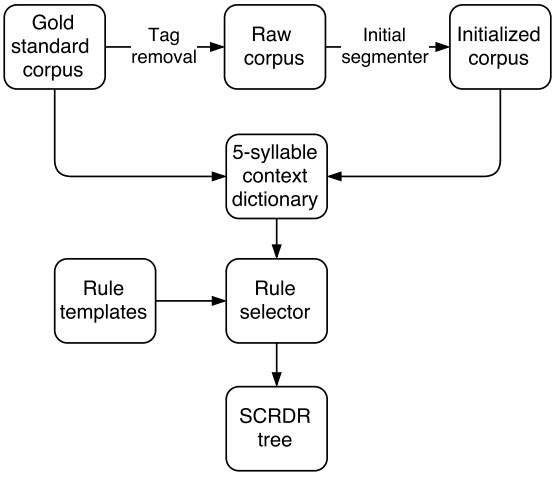


Figure 2: Diagram of our approach.

3. Our approach

This section describes our new error-driven approach to automatically construct a SCRDR tree to correct wrong segmentations produced by an initial word segmenter.

Following Nguyen et al. (2006) and Tran et al. (2010), we also formalize the word segmentation problem as a sequence labeling task. In particular, each syllable is labeled by either segmentation tag B (Begin of a word) or I (Inside of a word). As a result, our approach can be viewed as an extension to word segmentation of the automatic SCRDR approach for POS tagging (Nguyen et al., 2014; Nguyen et al., 2016). Our learning diagram is described in Figure 2.

We start with an underscore-based *gold standard training corpus* consisting of manually word-segmented sentences, e.g. "thuế_thu_nhập cá_nhân" (individual$_{\text{cá_nhân}}$ income$_{\text{thu_nhập}}$ tax$_{\text{thuế}}$) and transform this corpus into a BI-formed representation (e.g. "thuế/B thu/I nhập/I cá/B nhân/I"). We then extract syllables to construct the *raw corpus* (which does not have B and I segmentation tags, and would look like "thuế thu nhập cá nhân").

Tuple as key	Value	
(“”, “”, “”, “”, **thuế**, **B**, thu, B, nhập, I)	B	√
(“”, “”, thuế, B, **thu**, **B**, nhập, I, cá, B)	I	X
(thuế, B, thu, B, **nhập**, **I**, cá, B, nhân, I)	I	√
(thu, B, nhập, I, **cá**, **B**, nhân, I, “”, “”)	B	√
(nhập, I, cá, B, **nhân**, **I**, “”, “”, “”, “”)	I	√

Table 1: Examples of key-value pairs in the 5-syllable context dictionary $\mathcal{D}$ when comparing the BI-formed gold standard corpus “thuế/B thu/I nhập/I cá/B nhân/I” and the BI-formed initialized corpus “thuế/B thu/B nhập/I cá/B nhân/I.” Here, **“”** denotes an empty element in tuples. √ and X represent the correct and incorrect initial segmentations, respectively.

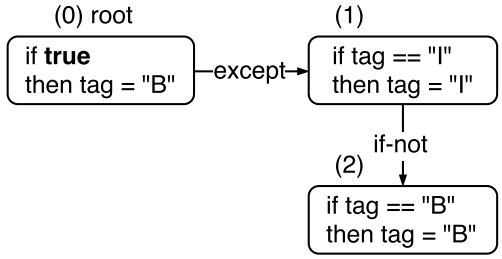


Figure 3: SCRDR tree initialization.

We apply an *initial segmenter* on the input raw corpus to get the output BI-formed *initialized corpus*. For example, given the input raw text “thuế thu nhập cá nhân”, the initial segmenter returns the output BI-formed initialized text “thuế/B thu/B nhập/I cá/B nhân/I.” The initial segmenter in our approach is based on the longest matching strategy (Poowarawan, 1986), using a Vietnamese lexicon from Le et al. (2008).

We then compare the BI-formed gold standard corpus and the BI-formed initialized corpus to generate a *5-syllable context dictionary* $\mathcal{D}$ where each key-value pair consists of a 5-syllable window tuple as key and a gold standard tag as value. Here, each tuple captures a 5-syllable window context of a current syllable and its initial segmentation tag B/I in a format of (Previous-2nd-syllable, Previous-2nd-tag, Previous-1st-syllable, Previous-1st-tag, syllable, tag, Next-1st-syllable, Next-1st-tag, Next-2nd-syllable, Next-2nd-tag) from the initialized corpus,[2] while the gold standard tag is the corresponding segmentation tag of the current syllable in the gold standard corpus. So, a wrong segmentation is when the initial segmentation tag is different from the gold standard tag, as shown in the second row in Table 1.

Based on the 5-syllable context dictionary $\mathcal{D}$, the *rule selector* selects the most suitable rules to construct the *SCRDR tree*. Concrete rules are generated based on *rule templates*. Table 2 presents short descriptions of the rule templates. The SCRDR tree is initialized with a default rule—the rule at the root node—and its two exception rules, as shown in Figure 3. Our learning process to automatically add new exception rules to the SCRDR tree is as follows:

syllable	s_{-2}, s_{-1}, s_0, s_{+1}, s_{+2}
	(s_{-2}, s_0), (s_{-1}, s_0), (s_{-1}, s_{+1}), (s_0, s_{+1}) (s_0, s_{+2})
	(s_{-2}, s_{-1}, s_0), (s_{-1}, s_0, s_{+1}), (s_0, s_{+1}, s_{+2})
tag	t_{-2}, t_{-1}, t_0, t_{+1}, t_{+2}
	(t_{-2}, t_{-1}), (t_{-1}, t_{+1}), (t_{+1}, t_{+2})
syllable & tag	(t_{-1}, s_0), (s_0, t_{+1}), (t_{-1}, s_0, t_{+1}), (t_{-2}, t_{-1}, s_0) (s_0, t_{+1}, t_{+2})

Table 2: Short descriptions of our rule templates. “s” refers to syllable and “t” refers to B/I segmentation label while subscripts -2, -1, 0, 1, 2 denote indices. For example, (s_{-1}, s_{+1}) represents the rule template “IF Previous-1st-syllable == **tuple.Previous-1st-syllable** && Next-1st-syllable == **tuple.Next-1st-syllable** THEN tag = **gold-standard-tag**”, where elements in **bold** are replaced by concrete values from tuple and gold tag pairs in the 5-syllable context dictionary $\mathcal{D}$. Given (s_{-1}, s_{+1}) and the second row in Table 1, we have a concrete rule “IF Previous-1st-syllable == **thuế** && Next-1st-syllable == **nhập** THEN tag = **I**.”

- Let us consider a node N in the tree. We define a subset $\mathcal{T}_N$ of the context dictionary $\mathcal{D}$ such that the rule at N is the last satisfied rule in the evaluation path for every tuple in $\mathcal{T}_N$ but N returns a wrong segmentation tag. For example, given node (2) in Figure 3 and $\mathcal{D}$ in Table 1, $\mathcal{T}_{(2)}$ would contain a pair of the tuple (**“”**, **“”**, thuế, B, **thu**, **B**, nhập, I cá, B) and gold segmentation tag **I** from the second row in Table 1. A new node containing a new exception rule must be added to the current tree to correct the errors given by N.[3]

- The new exception rule is selected from all concrete rules, in which these concrete rules are generated by applying the rule templates to all tuples in $\mathcal{T}_N$. The selected rule must satisfy following constraints: (1) If N is not one of the first three nodes in Figure 3, then the selected rule's condition must not be satisfied by every tuple for which N already returns a correct segmentation tag. (2) The selected rule is associated with the highest

[2]Syllables in each tuple are all converted into a lowercase form.

[3]See the second last paragraph in Section 2. for how to attach a new node to an existing SCRDR tree.

value of the subtraction $a - b$. Here a is the number of tuples in $\mathcal{T}_N$ in which each tuple not only satisfies the rule's condition but also gets a correct segmentation tag given by the rule's conclusion, while b is the number of tuples in $\mathcal{T}_N$ in which each tuple also satisfies the rule's condition but gets a wrong segmentation tag given by the rule's conclusion. (3) The value $a - b$ must be not smaller than a given threshold.

- This process is repeated until at any node it cannot select a new exception rule satisfying constraints above.

With the learned SCRDR tree, we perform word segmentation on unsegmented text as follows: The initial segmenter takes the input unsegmented text to generate a BI-formed initialized text. Next, by sliding a 5-syllable window from left to right, a tuple is generated for each syllable in the initialized text; then the learned SCRDR tree takes the input tuple to return a final segmentation tag to the corresponding syllable. Finally, the output of this labeling process is converted to the traditional underscore-based representation.

4. Experiments

4.1. Experimental setup

Following Nguyen and Le (2016), we conduct experiments and compare the performance of our approach—which we call RDRsegmenter—with published results of other state-of-the-art approaches on the benchmark Vietnamese treebank (Nguyen et al., 2009). The training set consists of 75k manually word-segmented sentences (about 23 words per sentence in average).[4] The test set consists of 2120 sentences (about 31 words per sentence) in 10 files from *800001.seg* to *800010.seg*.[5] We use F_1 score as the main evaluation metric to measure the performance of word segmentation.

Note that to determine the threshold in our RDRsegmenter, we sampled a development set of 5k sentences from the full training set and used the remaining 70k sentences for training. We found an optimal threshold value at 2 producing the highest F_1 score on the development set. Then we learned a SCRDR tree from the full training set with the optimal threshold, resulting in 1447 rules in total.

[4]The data, officially released in 2013, is provided for research or educational purpose by the national project VLSP on Vietnamese language and speech processing.

[5]The test set was originally released for evaluation in the POS tagging shared task at the VLSP 2013 workshop.

Approach	Precision	Recall	F_1
vnTokenizer	96.98	97.69	97.33
JVnSegmenter-Maxent	96.60	97.40	97.00
JVnSegmenter-CRFs	96.63	97.49	97.06
DongDu	96.35	97.46	96.90
UETsegmenter	**97.51**	98.23	97.87
Our **RDRsegmenter**	97.46	**98.35**	**97.90**

Table 3: Vietnamese word segmentation results (in %). The results of vnTokenizer, JVnSegmenter and DongDu are reported in Nguyen and Le (2016).

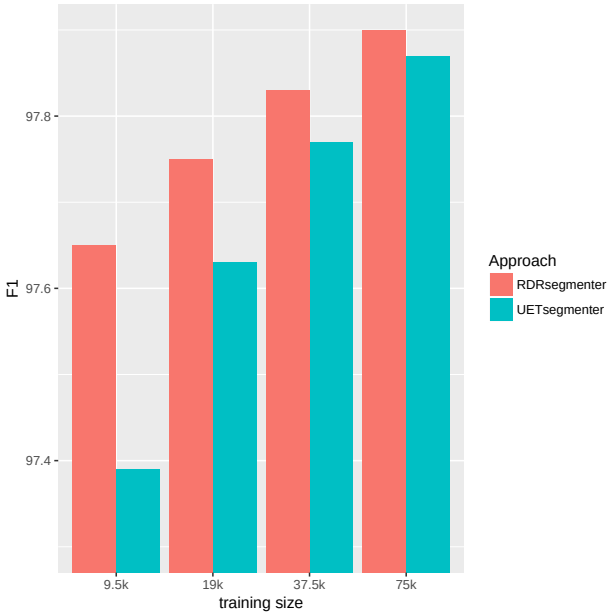


Figure 4: F_1 scores (in %) when varying the training size at 9.5k, 19k, 37.5k and full 75k sentences.

4.2. Main results

Table 3 compares the Vietnamese word segmentation results of our RDRsegmenter with results reported in prior work, using the same experimental setup.

Table 3 shows that RDRsegmenter obtains the highest F_1 score. In particular, RDRSegmenter obtains 0.5+% higher F_1 than vnTokenizer (Le et al., 2008) though both approaches use the same lexicon for initial segmentation. In terms of a sequence labeling task, RDRSegmenter outperforms JVnSegmenter (Nguyen et al., 2006) with 0.8+% improvement. Compared with the pointwise prediction approaches DongDu (Luu and Kazuhide, 2012) and UETsegmenter (Nguyen and Le, 2016), RDRsegmenter does significantly better than DongDu and somewhat better than UETsegmenter. In Figure 4, we show F_1 scores of RDRsegmenter and UETsegmenter at different training sizes, showing that RDRsegmenter clearly improves performance in a smaller dataset scenario.

It is worth noting that on a personal computer of In-

tel Core i7 2.2 GHz, our RDRsegmenter processes at a speed of 62k words per second in a single threaded implementation, which is 1.3 times faster than UETsegmenter.[6] In addition, Nguyen and Le (2016) showed that UETsegmenter is faster than vnTokenizer, JVnSegmenter and DongDu.[7] So RDRsegmenter is also faster than vnTokenizer, JVnSegmenter and DongDu.

5. Conclusion

In this paper, we have proposed a new error-driven method to automatically construct a Single Classification Ripple Down Rules tree for Vietnamese word segmentation. Experiments on the benchmark Vietnamese treebank show that our method obtains better accuracy and speed than previous approaches. Our code is available at: `https://github.com/datquocnguyen/RDRsegmenter`.

Note that excluding the language-specific initial segmenter, our method generally can be viewed as a language independent approach. Here, a Vietnamese syllable is analogous to a character in other languages such as Chinese and Japanese. So we will adapt our method to those languages in future work.

6. Acknowledgments

This research was partially supported by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project DP160102156). This research was also partially supported by NICTA, funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program. This research was done while the first author was at Macquarie University.

7. Bibliographical References

Compton, P. and Jansen, B. (1988). Knowledge in Context: A Strategy for Expert System Maintenance. In *Proceedings of the 2nd Australian Joint Artificial Intelligence Conference*, pages 292–306.

Compton, P. and Jansen, R. (1990). A Philosophical Basis for Knowledge Acquisition. *Knowledge Aquisition*, 2(3):241–257.

Dien, D., Kiem, H., and Toan, N. V. (2001). Vietnamese Word Segmentation. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*, pages 749–756.

Dinh, D. and Vu, T. (2006). A Maximum Entropy Approach for Vietnamese Word Segmentation. In *Proceedings of the 2006 International Conference on Research, Innovation and Vision for the Future*, pages 248–253.

Duc Cong, S. N., Hung Ngo, Q., and Jiamthapthaksin, R. (2016). State-of-the-art Vietnamese word segmentation. In *Proceedings of the 2nd International Conference on Science in Information Technology*, pages 119–124.

Ha, L. A. (2003). A method for word segmentation in Vietnamese. In *Proceedings of Corpus Linguistics*.

Le, H. P., Nguyen, T. M. H., Roussanaly, A., and Ho, T. V. (2008). A hybrid approach to word segmentation of Vietnamese texts. In *Proceedings of the 2nd International Conference on Language and Automata Theory and Applications*, pages 240–249.

Liu, W. and Lin, L. (2014). Probabilistic Ensemble Learning for Vietnamese Word Segmentation. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 931–934.

Luu, T. A. and Kazuhide, Y. (2012). Ứng dụng phương pháp Pointwise vào bài toán tách từ cho tiếng Việt. https://github.com/rockkhuya/DongDu.

NanYuan, L. and YanBin, Z. (1991). A Chinese word segmentation model and a Chinese word segmentation system PC-CWSS. *Journal of Chinese Language and Computing*, 1(1).

Neubig, G. and Mori, S. (2010). Word-based Partial Annotation for Efficient Corpus Construction. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, pages 2723–2727.

Nguyen, T.-P. and Le, A.-C. (2016). A Hybrid Approach to Vietnamese Word Segmentation. In *Proceedings of the 2016 IEEE RIVF International Conference on Computing and Communication Technologies: Research, Innovation, and Vision for the Future*, pages 114–119.

Nguyen, D. B. and Pham, S. B. (2012). Ripple Down Rules for Vietnamese Named Entity Recognition. In *Proceedings of the 4th international conference on Computational Collective Intelligence: Technologies and Applications*, pages 354–363.

[6] We repeated the segmentation process on the test set 100 times, and then computed the averaged speed. Note that model loading time was not taken into account, in which RDRsegmenter took 50 miniseconds while UETsegmenter took 10 seconds.

[7] Evaluated on a computer of Intel Core i5-3337U 1.80GHz, Nguyen and Le (2016) showed that JVnSegmenter, vnTokenizer, DongDu and UETsegmenter obtained performance speeds at 1k, 5k, 17k and 33k words per second, respectively. All of them are implemented in Java except DongDu which is in C++. Our RDRsegmenter is also implemented in Java.

Nguyen, C.-T., Nguyen, T.-K., Phan, X.-H., Nguyen, L.-M., and Ha, Q.-T. (2006). Vietnamese Word Segmentation with CRFs and SVMs: An Investigation. In *Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation*, pages 215–222.

Nguyen, P. T., Vu, X. L., Nguyen, T. M. H., Nguyen, V. H., and Le, H. P. (2009). Building a Large Syntactically-Annotated Corpus of Vietnamese. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 182–185.

Nguyen, D. Q., Nguyen, D. Q., and Pham, S. B. (2011a). Systematic Knowledge Acquisition for Question Analysis. In *Proceedings of the 8th International Conference on Recent Advances in Natural Language Processing*, pages 406–412.

Nguyen, D. Q., Nguyen, D. Q., Pham, S. B., and Pham, D. D. (2011b). Ripple Down Rules for Part-of-Speech Tagging. In *Proceedings of the 12th International Conference on Intelligent Text Processing and Computational Linguistics - Volume Part I*, pages 190–201.

Nguyen, Q. T., Nguyen, N. L., and Miyao, Y. (2012). Comparing Different Criteria for Vietnamese Word Segmentation. In *Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing*, pages 53–68.

Nguyen, D. Q., Nguyen, D. Q., and Pham, S. B. (2013). KbQAS: A Knowledge-based QA System. In *Proceedings of the 12th International Semantic Web Conference (Posters & Demonstrations Track)*, pages 109–112.

Nguyen, D. Q., Nguyen, D. Q., Pham, D. D., and Pham, S. B. (2014). RDRPOSTagger: A Ripple Down Rules-based Part-Of-Speech Tagger. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 17–20.

Nguyen, D. Q., Nguyen, D. Q., Pham, D. D., and Pham, S. B. (2016). A Robust Transformation-Based Learning Approach Using Ripple Down Rules for Part-of-Speech Tagging. *AI Communications*, 29(3):409–422.

Nguyen, D. Q., Nguyen, D. Q., and Pham, S. B. (2017a). Ripple Down Rules for Question Answering. *Semantic Web*, 8(4):511–532.

Nguyen, D. Q., Vu, T., Nguyen, D. Q., Dras, M., and Johnson, M. (2017b). From Word Segmentation to POS Tagging for Vietnamese. In *Proceedings of the Australasian Language Technology Association Workshop 2017*, pages 108–113.

Pham, S. B. and Hoffmann, A. (2006). Efficient Knowledge Acquisition for Extracting Temporal Relations. In *Proceedings of the 17th European Conference on Artificial Intelligence*, pages 521–525.

Pham, D. D., Tran, G. B., and Pham, S. B. (2009). A Hybrid Approach to Vietnamese Word Segmentation using Part of Speech tags. In *Proceedings of the 2009 International Conference on Knowledge and Systems Engineering*, pages 154–161.

Plisson, J., Lavrač, N., Mladenić, D., and Erjavec, T. (2008). Ripple Down Rule Learning for Automated Word Lemmatisation. *AI Communications*, 21(1):15–26.

Poowarawan, Y. (1986). Dictionary-based Thai Syllable Separation. In *Proceedings of the Ninth Electronics Engineering Conference*, pages 409–418.

Richards, D. (2009). Two Decades of Ripple Down Rules Research. *Knowledge Engineering Review*, 24(2):159–184.

Takahashi, K. and Yamamoto, K. (2016). Fundamental tools and resource are available for Vietnamese analysis. In *Proceedings of the 2016 International Conference on Asian Language Processing*, pages 246–249.

Thang, D. Q., Phuong, L. H., Huyen, N. T. M., Tu, N. C., Rossignol, M., and Luong, V. X. (2008). Word segmentation of Vietnamese texts : a comparison of approaches. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, pages 1933–1936.

Tran, T. O., Le, A. C., and Ha, Q. T. (2010). Improving Vietnamese Word Segmentation and POS Tagging using MEM with Various Kinds of Resources. *Journal of Natural Language Processing*, 17(3):41–60.

Tran, N. A., Dao, T. T., and Nguyen, P. T. (2012). An effective context-based method for Vietnamese-word segmentation. In *Proceedings of the 1st International Workshop on Vietnamese Language and Speech Processing*, pages 34–40.

Xu, H. and Hoffmann, A. (2010). RDRCE: Combining Machine Learning and Knowledge Acquisition. In *Proceedings of the 11th International Workshop on Knowledge Management and Acquisition for Smart Systems and Services*, pages 165–179.

Finite-state morphological analysis for Gagauz

Sevilay Bayatli,⋆ Güllü Karanfil,† Memduh Gökırmak,‡ Francis M. Tyers♣
⋆ Faculty of Electrical and Computer Engineering, School of Science and Engineering, Altınbaş University
† Anadolu BİL Meslek Yüksekokulu, Uygulamalı Rusça ve Çevirmenlik, İstanbul Aydin University
‡ Department of Computer Engineering, School of Engineering, İstanbul Teknik Üniversitesi
♣ School of Linguistics, Faculty of the Humanities, Higher School of Economics
sevilaybayatli@gmail.com, gullukaranfil@hotmail.com, memduhg@gmail.com, ftyers@hse.ru

Abstract

This paper describes a finite-state approach to morphological analysis and generation of Gagauz, a Turkic language spoken in the Republic of Moldova. Finite-state approaches are commonly used in morphological modelling, but one of the novelties of our approach is that we explicitly handle orthographic errors and variance, in addition to loan words. The resulting model has a reasonable coverage (above 90%) over a range of freely-available corpora.

Keywords: morphology, Gagauz, finite-state transducer

1. Introduction

Gagauz is a Turkic language of the Oghuz group spoken by approximately 140,920 worldwide and 105,000 in the Republic of Moldova where it is co-official with Moldovan[1] in the autonomous region of Gagauzia. Most speakers are bilingual in Russian and education takes place almost entirely in Russian. This leads to a number of issues with attempting to create a wide-coverage morphological model, as text *in the wild* is often less orthographically accurate than in languages which are widely taught. There is a further issue that keyboards may not be easily available, and so the choice might be between a Turkish keyboard which does not include ţ and a Moldovan keyboard which does not include ı, ö, ü.

The paper is laid out as follows, in Section 2. we describe give a brief typological description of Gagauz, in Section 3., in Section 4. we describe the evaluation and results then in Sections 5. and 6. we give some thoughts on further work and conclusions.

2. Gagauz

To our knowledge to date there has been no computational linguistic work on Gagauz. There are two available grammars, one in Russian (Покровская, 1964) and one in Turkish (Özkan, 1996). The former describes the literary language, where the latter takes a comparative approach, comparing Gagauz with Turkish. There have also been a number of works on Gagauz lexicography, for example Каранфил (2009).

As with other Turkic languages, Gagauz is an SOV language. It has no gender, plural marking and seven cases (nominative, genitive, dative, accusative, locative, ablative and instrumental). Vowel harmony is applied in affixes. Inflectional marking is used to indicate possession and also for agreement within subordinate clauses, *işlediņi gördüm* 'I saw that you were working' (lit. 'I saw your working').

[1] Moldovan is one of the official names for Romanian in the Republic of Moldova. The discussion of if Romanian and Moldovan are separate languages is a sociopolitical one and so for the purposes of this article we consider the names to be equivalent.



Figure 1: Location of the Gagauz speaking area (gag) within the Black Sea region, relative to other Oghuz (Turkish – tur, and Azerbaijani – azb and azj) and Kypchak (Urum – uum, Crimean Tatar – crh, Karachay-Balkar – krc, Nogay – nog, Kumyk – kum and Kazakh – kaz) languages.

Subordination is done principally with verbal affixes which may have multiple possible syntactic readings.

There are many derivational processes, some being very productive, such as the *-k{I}* morpheme which creates attributives from locatives (1a) and substantives from genitives (1b).

(1) a. *Komrat-ta-kı sport liţeyi aç-ıl-dı*
Komrat-LOC-ATTR sport lycée-3SG open-PASS-PAST
'The sports lycée in Komrat was opened.'

b. *ana-m-ın-kı-nı gör-dü-m*
mother-1SG-GEN-SUBST-ACC see-PAST-1SG
'I saw my mother's (one)'

Gagauz shows the usual variation regarding voicing/devoicing of initial consonants in the Oghuz branch of the Turkic family, for example *taa/daha* 'more'. Gagauz also does not exhibit the Turkish characteristic of final consonant devoicing, e.g. the word arab becomes arap in Turkish but does not undergo this change in Gagauz.

Gagauz makes a distinction between the sounds *e* [e] and *ä* [æ], like Azerbaijani and unlike Turkish. Turkish tends to avoid long vowels, specifically in native morphology. Where there would be a voiced velar approximant or a glide in Turkish, in Gagauz the consonant is often not present and there is instead a long vowel. This is not limited to morphology, and some of the lexicon also exhibits this difference, as can be seen in the same example of Turkish *daha* and Gagauz *taa* 'more'.

Like many other languages post-Soviet nations, certain writing conventions have been adopted from Russian, such as a long dash – between the subject and predicate of a declarative sentence with a zero copula. *Arab dili – angisindä laf eder arab halkı* 'The Arabic language – that which the Arab people speak.'

3. Methodology

Development broadly follows the methodology described in Washington et al. (2014), using the Helsinki Finite-State Toolkit, HFST (Lindén, 2009). This toolkit supports the `lexc` formalism for building lexicons and the `twol` for defining phonological constraints. It also supports weighted finite-state transducers. The system is composed of a lexical transducer implemented in the `lexc` formalism, and three phonological/orthographical transducers implemented in the `twol` formalism.

3.1. Lexicon and morphotactics

The lexical transducer, which maps between lexical forms and morphotactic forms (see the example for *insannarına* 'to the people of' in Table 1), was created in the following way: The lexicon was developed completely by hand based on a frequency list. The nominal morphotactics was based on that in Washington et al. (2014), while the verbal morphotactics was created from scratch based on the grammars by Покровская (1964) and Özkan (1996) and corpus investigation. There is a freely-available morphological analyser for Turkish by Çöltekin (2010), but we decided against reusing the verbal morphotactics, as despite Gagauz and Turkish being closely related, there are a number of differences in the verbal morphotactics that would make it difficult to transfer (for example, Gagauz does not have the progressive form that is found in Turkish).

The lexical transducer consists of 5,211 lexemes and 101 continuation classes (sets of suffixes).

3.2. Text encoding

Gagauz uses two letters common with Moldovan, ş /ʃ/ and ţ /ts/. In Moldovan these are normatively spelt as ș and ț using a comma as opposed to a cedilla. In Gagauz text both encodings are found and so we implemented a `twol` file to allow both variants.

3.3. Morphophonology

The morphophonological component is implemented using two-level morphology, `twol` (Koskenniemi, 1984); a total of 24 rules are applied to the lexical forms (see §3.1.) in order to produce surface forms.

For example, in the passive the archiphoneme is -{i}{л}[2], the {л} changes to *n* if it is preceeded by an *-l-*. For example the underlying form of *bulunduk* 'we were found' is `bul-{i}{л}-{D}{I}-k`. The `twol` rule in (2) implements this constraint.

```
(2) "{л} to n in passive following l"
    %{л%}:n <=> :l/:0* %{i%}: _ ;
```

Another rule (3) implements the vowel harmony of the `{I}` archiphoneme:

```
(3) "Vowel harmony for archiphoneme {I}"
    %{I%}:Vy <=> :Vx [:Cns|LowerCns:]+/:0* _ ;
      except
        :Vx [ LowerCns: ] %>: %{s%}: _ %{n%}: ;
      where Vx in ( a â ä e ê o ö u ü ı i i )
            Vy in ( ı ı i i ı u ü u ü ı i i )
      matched ;
```

This rule has an exception for cases of stem consonant elision, for example, the word *topraana* 'to its land' has the underlying form of `topra{k}-{s}{I}{n}-{й}{A}`, where the `{s}` surfaces as null because of the preceeding `{k}`, the `{k}` surfaces as null because it ends up between two vowels, *a* and `{I}` and finally the `{I}` is constrained to assimilate with the previous vowel (carried out by another rule).

3.4. Orthographic errors

Orthographic errors are modelled both in the lexicon and in the phonological rules. In the lexicon errors in stems, for example **iyilik* instead of *iilik* 'goodness', are marked with a comment `Err/Orth`. For typographical, orthographical and phonological errors, such as **içinde* instead of *içindä* 'in the inside of', a separate `twol` file is used for modelling errors. In this error-model `twol` file, we relax the constraints to allow for possible errors, for example `{I}` can surface as {ı, i, u, ü}.

3.5. Compilation

We first compile the lexicon without the orthographic errors to produce the transducer L_n, we then compile the two-level normative rule file to produce the set of rule transducers P, we compose L_n with P to produce the transducer which contains the normative surface forms, T_n. We then compile the full lexicon L_f and compose it with the two-level error rule file E, creating the transducer with all the possible strings T_f (this includes orthographic errors). After that, we subtract the strings in T_f that are in T_n and append a tag to each string indicating orthographic error, `<err_orth>`, and call this the error transducer T_e. The final transducer is the union of T_n and T_e where each pair of erroneous surface form and analysis has a tag indicating it is an error at the end of the analysis.

4. Results

We calculate the *naïve coverage* for the analyser over a number of available corpora: Gagauz Wikipedia, a collection of

[2]We chose Cyrillic {л} 'l' to represent this as Latin 'l' was used for the plural morpheme.

Lexical form		Morphotactic form
`insan<n><pl><px3sp><dat>`	↔	`insan>{L}{A}r>{s}{I}{n}>{й}{A}`

Table 1: Morphotactic representation of the surface form *insannarına* 'to the people of'. The symbols within '{' and '}' characters are archiphonemes which may appear in the surface as a number of different characters, for example `{L}` may appear as *l* or *n*, `{I}` may appear as any high vowel: *ı*, *i*, *u*, or *ü*. The symbols within '<' and '>' characters are grammatical tags, for example `<px3sp>` is the third-person possessive suffix.

Corpus	Genre	Tokens	Coverage (%)	
			GAG	TUR
Ana Sözü	News	525,483	90.2	62.1
Wikipedia	Encyclopaedic	163,403	90.8	64.3
NT	Religion	330,431	90.0	66.2

Table 2: Coverage of the morphological analyser over a range of corpora. The column GAG refers to the coverage using our implementation, while TUR is the coverage after running the TRMorph morphological analyser. As can be seen, despite being closely-related, differences amount to a substantial difference in coverage between the two analysers.

texts from the news site *Ana Sözü*[3] and the New Testament (NT) in Gagauz. Table 2 presents these results. Naïve coverage refers to the percentage of surface forms in a given corpora that receive at least one morphological analysis. Forms counted by this measure may have other analyses which are not delivered by the transducer. We also give the results for running the corpora through TRMorph (Çöltekin, 2010), a freely-available analyser for Turkish, which can be considered a kind of baseline — indicating the performance that could be achieved simply by running a mature analyser for a closely-related language.

In order to evaluate the analyser on a deeper level, We selected 1,000 tokens at random from a list of unique tokens produced from the concatenation of all the corpora. We gave these tokens in 60 characters of context each to a speaker of Gagauz along with the following questionnaire:

- **Is the word Gagauz?** By this we mean could the word be seen in texts in Gagauz, it may be a word with Turkic roots, like *üüredici* 'teacher', or it may be a Russian/international word like *komerţiya* 'commerce' or a word from any number of other languages such as Arabic, e.g. *ceza* 'fine'. The important question is not, "is this a native Gagauz word?" but "could this word be seen in Gagauz texts". A counter example would be *attempt*, an English word which would not appear in Gagauz texts. Another example would be *html* which is not a Gagauz word but some kind of code. Please answer 'NO' if the word is a concatenation of two words caused by a tokenisation error for example *sevincimnänAteş*.

- **Is the word correctly spelt?** By this we mean, is the word spelt correctly according to Gagauz orthography, this includes using the correct diacritics and special characters e.g. *intergraţiya* 'integration' not **intergratsiya* and *lääzım* 'necessary' not **laazim* or **laazım*. Other examples of typographical errors might be adopting Turkish orthographical rules like **Türkiye'nin* instead of Gagauz *Türkiyenin*. Please pay special attention to vowel harmony.

- **What is the part of speech?** Please give the part of speech of the word, choose from: Noun, Adjective, Verb, Adverb, Other.

- **What is the lemma?** Please give the lemma of the word. This is the dictionary form of the word, for example the lemma of *kitaplar* 'books' is *kitap* 'book'.

They then filled out the answers to the questionnaire in a spreadsheet, a sample of which can be found in Table 3.

Out of the tokens, 90 of them were not Gagauz words, and 99 were misspellings. This gave us 910 tokens to evaluate the analyser.

Table 4 shows the coverage of each of the parts of speech according to the test corpus. Note that unlike the naïve coverage, this is a coverage of a random set of tokens and does not take frequency into account. Even so we can see that most of the unknown words come from the open classes (adjectives, nouns and proper nouns).

Figure 2 gives an example of output from the transducer.

5. Future work

As Gagauz is syntactically very close to Turkish, we would like to try cross-lingual methods to morphological disambiguation and dependency parsing. There is an existing treebank of Turkish (Sulubacak et al., 2016) in the format of the Universal Dependencies project and this would be ideal to train a statistical disambiguator. In addition we would also like to explore machine translation between Turkish and Gagauz using the Apertium platform (Forcada et al., 2011). It is worth noting that there are a number of problems in the phonological rules that we are intending to fix. We would also like to expand the lexicon.

6. Concluding remarks

We have presented the first computational model of Gagauz morphology. The transducer has good coverage of a range of available corpora and can handle a range of issues relating to orthography and encoding that are found when dealing with Gagauz. The transducer has the potential to be of great use to the language community as a spellchecker as well as being a key part of other language processing tools. The code is

[3] The web pages was scraped from the home page at `http://www.anasozu.md` in HTML and the text was extracted using a custom Python script

ID	Word	Gagauz?	Spelling?	POS	Lemma	Context
5	payedelecklär	YES	PAYEDILECEKLÄR	FİİL	PAYET-	... dı. Onnar üç uurda proektlara **payedelecklär**. ...
71	ministrulara	NO	–	–	–	... ne premyer-ministruya, ne da **ministrulara** hiç bir soruş ta ...
109	ordenını	YES	ORDENİNİ	İSİM	ORDEN	... likasının "Ordinul de Onoare" **ordenını**. ...
145	süveriz	YES	SÜÜYERİZ	FİİL	SÜÜ-	... nardan çektiimiz için bunnarı **süveriz**. ...
181	İnstitunda	YES	İNSTİTUTUNDA	İSİM	İNSTİTUT	... uzıcescu" adına İncäzanaatlar **İnstitunda** (1972-1992) kulturo ...
…	…	…	…	…	…	…

Table 3: An sample of five lines from the evaluation questionnaire. Note that the Moldovan word *ministru* 'minister' appears here with Gagauz morphology *ministrulara* 'to the ministers'. The annotator has decided that this is a mistake as the normative Gagauz word would be *ministr* 'minister'. Although this would appear to go against our definition of 'is the word Gagauz?' (e.g. appears with Gagauz morphology in a normal Gagauz text) we deferred in all cases to the judgement of the annotator.

```
^Kendi/kendi<det><ref>$
^insannıı/insannık<n><px3sp><nom>$
^için/için<post>$
^hem/hem<cnjcoo>$
^becerikli/becerikli<adj>$
^çalışmakları/çalış<v><tv><ger><pl><px3sp><nom>$
^için/için<post>$
^“/“<lquot>$
^Komrat/Komrat<np><top><nom>$
^kasabasının/kasaba<n><px3sp><gen>$
^şannı/şannı<adj>$
^vatandaşı/vatandaş<n><px3sp><nom>$
^”/”<rquot>$
^adını/ad<n><px3sp><acc><acc>$
^taşıyêr/taşı<v><tv><pres><p3><sg>$
^./.<sent>$
```

Figure 2: Example output of the morphological analyser, the analyses have been disambiguated in context in order to fit on the page. The sentence is *Kendi insannıı için hem becerikli çalışmakları için "Komrat kasabasının şannı vatandaşı" adını taşıyêr.* 'She holds the title "honoured citizen of the town of Komrat" for her skilled work for her people.'

Category	Known	Unknown	Coverage (%)
Punctuation	1	0	100.0
Conjunction	1	0	100.0
Particle	1	0	100.0
Numeral	7	1	87.5
Adverb	10	2	83.3
Verb	164	49	77.0
Pronoun	16	9	64.0
Noun	296	200	59.8
Proper noun	58	68	46.0
Adjective	11	15	42.3
Abbreviation	0	1	0.0
Total:	565	345	61.0

Table 4: Coverage of part of speech categories in the randomly selected test set.

available under a free/open-source licence.[4]

7. Acknowledgements

We would like to thank Alla Büük and Kämil Gagauz for their help as native speakers.

[4] https://svn.code.sf.net/p/apertium/svn/incubator/apertium-gag

References

Çöltekin, Ç. (2010). A Freely Available Morphological Analyzer for Turkish. In: *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC2010)*, pp. 820–827.

Forcada, M. L., Ginestí-Rosell, M., Nordfalk, J., O'Regan, J., Ortiz-Rojas, S., Pérez-Ortiz, J. A., Sánchez-Martínez, F., Ramírez-Sánchez, G., and Tyers, F. M. (2011). Apertium: a free/open-source platform for rule-based machine translation. In: *Machine Translation* 25.2, pp. 127–144.

Koskenniemi, K. (1984). A General Computational Model for Word-form Recognition and Production. In: *Proceedings of the 10th International Conference on Computational Linguistics and 22Nd Annual Meeting on Association for Computational Linguistics*. ACL '84. Stanford, California: Association for Computational Linguistics, pp. 178–181. DOI: 10.3115/980491.980529. URL: https://doi.org/10.3115/980491.980529.

Lindén, K. (2009). Guessers for Finite-State Transducer Lexicons. In: *Computational Linguistics and Intelligent Text Processing 10th International Conference, CICLing 2009*, pp. 158–169.

Özkan, N. (1996). Gagavuz Türkçesi. Ankara: Ankara.

Sulubacak, U., Gökırmak, M., Tyers, F., Çöltekin, Ç., Nivre, J., and Eryiğit, G. (2016). Universal Dependencies for

Turkish. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan, pp. 3444–3454.

Washington, J. N., Salimzyanov, I., and Tyers, F. M. (2014). Finite-state morphological transducers for three Kypchak languages. In: *Proceedings of the 9th Conference on Language Resources and Evaluation, LREC2014*.

Каранфил, Г. (2009). Гагаузская лексикология. Комрат.

Покровская, Л. А. (1964). Грамматика гагаузкого язка: Фонетика и морфология. Moscow: Издательство «Наука».

Albanian Part-of-Speech Tagging: Gold Standard and Evaluation

Besim Kabashi, Thomas Proisl
Friedrich-Alexander-Universität Erlangen-Nürnberg
Professur für Korpuslinguistik
Bismarckstr. 6, 91054 Erlangen, Germany
{besim.kabashi, thomas.proisl}@fau.de

Abstract

In this paper, we present a gold standard corpus for Albanian part-of-speech tagging and perform evaluation experiments with different statistical taggers. The corpus consists of more than 31,000 tokens and has been manually annotated with a medium-sized tagset that can adequately represent the syntagmatic aspects of the language. We provide mappings from the full tagset to both the original Google Universal Part-of-Speech Tags and the variant used in the Universal Dependencies project. We perform experiments with different taggers on the full tagset as well as on the coarser tagsets and achieve accuracies of up to 95.10%.

Keywords: Albanian, part-of-speech tagging, gold standard

1. Introduction

Albanian is an Indo-European language that is spoken by approximately 5.4 million people. Within the Indo-European family of languages, Albanian constitutes a subgroup of its own; it is a distinct branch on the same level as, for example, the Hellenic, Romance, Slavic or Germanic subgroups. The language has a diverse vocabulary with many loan words due to language contact with Greek, Latin/Italian, Slavic languages and Turkish.

Albanian has a rich morphological system and a relatively free word order, similar to, for example, German. Particularly challenging, from a pos tagging point of view, are the many multi-word units. An interesting and frequent phenomenon are multi-word units that have articles or particles as their first part. These combinations are borderline cases at the lexis-grammar interface. They are certainly more grammatical in nature and written as two separate graphical tokens. Here is one example from standard Albanian: *i mirë* "(the) good one" (masc.) vs. *e mirë* "(the) good one" (fem.). Presence or absence of the preceding article or particle can change the meaning of a word or its part of speech.

Albanian is one of the Balkan languages with the least resources available. In particular, there is no part-of-speech tagged corpus available,[1] let alone one that covers the multi-word phenomena mentioned above. This paper is a step to improve that situation. In the next section, we briefly discuss previous work in the areas of part-of-speech tagging and morphological analysis. In Section 3., we present a new manually annotated gold standard corpus for part-of-speech tagging. The corpus is annotated with a medium-sized tagset that can adequately represent the multi-word phenomena mentioned above. To improve interoperability with existing multilingual tools and resources, we provide mappings to the popular Universal Part-of-Speech Tags, both the Google and the Universal Dependencies variant (Section 4.). In Section 5., we perform evaluation experiments with different part-of-speech taggers on the full tagset and on the coarser UPOS tagsets.

[1]The Albanian National Corpus (`http://web-corpora.net/AlbanianCorpus/search/`), which consists of roughly 16.7 million tokens, is only annotated with morphological analyses that have not been disambiguated (and only for words known to the morphological analyzer). The AlCo corpus (Kabashi, 2017), which consists of roughly 100 million tokens, is part-of-speech tagged, but is not yet publicly available.

2. Related Work

2.1. Part-of-Speech Tagging

Hasanaj (2012) presents two tagsets: A basic set that consists of 16 tags and a large set that consists of 326 tags. In the basic tagset, there are ten tags for the traditional parts of speech, three tags for delimiters, two for special cases (short forms of pronouns) and one for articles. The large tagset encodes the major word-classes and additional features like number or case. (Hasanaj, 2012) also attempts an evaluation of the two tagsets using a maximum entropy tagger and a perceptron tagger. However, his gold standard corpora consist only of 263 tokens for the basic tagset and 641 tokens for the large tagset.

Kadriu (2013) uses a tagset of 22 tags that refines the ten traditional parts of speech in some places and adds tags for feminine and masculine nouns, impersonal, reflexive, transitive and intransitive verbs, personal and possessive pronouns, determiners, exclamations, indeclinables and indefinite elements. Her tagging system is implemented on top of the NLTK unigram and regular expression taggers and uses a simple stemming algorithm to deal with unknown words. Her evaluation is based on 30 news articles from three domains. The number of tokens in the gold standard corpus is not mentioned in the paper.

The tagsets by Hasanaj (2012) and Kadriu (2013) are either very small or extremely large and do not cover the interactions between words and their preceding articles or particles described in Section 1. In our own previous work (Kabashi and Proisl, 2016), we introduce a medium-sized tagset for Albanian that consists of 67 tags and that aims to adequately represent the morphosyntactic properties of the Albanian language. In particular, combinations of preposed articles or particles with words of other word-classes are treated in a linguistically sensible way.

2.2. Morphological Analysis

There are a number of tools for the morphological analysis of Albanian that provide very detailed analyses but no disambiguation and no mapping to a medium-sized tagset.
Trommer and Kallulli (2004) present a morphological analyzer that seems to cover the main inflection types of Albanian. Its output format follows the EAGLE guidelines standard (Leech and Wilson, 1999) in that the tags consist of sets of attribute-value pairs. The tags come in three varieties: Detailed, abbreviated and collapsed. We can gather from the paper that the tagset distinguishes between 17 broad word-class labels, seven of which are reserved for subtypes of pronouns, one for the preposed article, one for sentence equivalents, one for the participle form of the verb, and the rest for the traditional parts of speech. Trommer and Kallulli (2004) evaluate their morphological analyzer against a gold standard corpus consisting of 1,000 tokens.
The goal of the tool described in Piton et al. (2007) and Piton and Lagji (2008) is to cover the inflection of Albanian. For words forms that can have them, their analysis also allows for preposed articles. It is not clear how many morphological tags they have or on how many major word-classes the tagset is based, though it seems to cover at least the ten traditional parts of speech. Their papers do not include an evaluation.
UniParser (Arkhangelskiy et al., 2012), the morphological analyzer used in the Albanian National Corpus, supports a variety of languages, e. g. Albanian, Greek, Kalmyk, Lezgian and Ossetic. There seems to be no published information about the Albanian model.
The morphological analyzer and generator by Kabashi (2015) extends the traditional parts of speech with additional tags for things like abbreviations or punctuation marks. For some word-classes, e. g. pronouns, more fine-grained subtypes are specified and are given their own tags. The system can also handle the preposed articles or particles that can occur with some word-classes. The coverage of the morphological analyzer is evaluated against word lists that each comprise more than 100,000 entries.

3. Gold Standard

3.1. The Tagset

The tagset used to annotate the corpus is a revised version of our earlier draft (Kabashi and Proisl, 2016) that has been used in the corpus described by Kabashi (2017). Traditional grammars like Newmark et al. (1982), Buchholz and Fiedler (1987) or Demiraj et al. (1995) give ten parts of speech for Albanian: Nouns, verbs, adjectives, adverbs, pronouns, prepositions, conjunctions, numerals, particles and interjections. While we follow this general division, the tagset allows for more fine-grained distinctions within the traditional word-classes and has additional tags for things like interjections, articles, pronominal clitics, punctuation, etc. that do not fit into the traditional word-classes.
In the course of annotating the corpus, we came across phenomena that could not be adequately analyzed with the old version of the tagset. Therefore, we refined and extended it by introducing ten novel tags. We also renamed some of the existing tags to have a more consistent and transparent naming scheme. In particular, we make the following changes:

- We introduce four new verb tags: VImpv (imperative form of verb), VImpvCl (imperative with infixed clitic), VSubj (verb with subjunctive particle) and VSubjPass (verb with subjunctive and passive particles).
- We introduce three new tags for pronouns: PPersSF (short form of personal pronoun), PIndefA (indefinite pronoun preceded by article) and PIntÇ (interrogative pronoun *Ç/ç*).
- For multipart conjunctions, we distinguish between coordinating (ConjC1, ConjC2) and subordinating (ConjS1, ConjS2).
- We introduce two new particle tags: PtMod (modal particle), PtPass (passive particle) and PtQM (question particle *mos*).
- Several tags were renamed to make them more consistent with the rest of the tagset. Now, all tags start with the major word-class (N, V, Adj, etc.) and are easier to interpret, e. g. PPAdjPPArt vs. AdjPA for a preposed adjective with preceding article or RelPPPArt vs. PRelA for a relative pronoun with preceding article.

The complete tagset is shown in Table 1.

3.2. The Corpus

We manually annotated a sample of 2,020 sentences (31,584 tokens) with the part-of-speech tagset described above. Half of the sentences have been randomly selected from a large collection of texts consisting of literary works, news and science articles and web texts. Selection of the other half has been guided by the wish to include as wide a variety of linguistic phenomena as possible in the gold standard corpus. Therefore, these sentences were manually selected and contain instances of rarer morphosyntactic phenomena. Consequently, the resulting corpus is definitely less homogeneous than a collection of whole documents and has a higher type/token-ratio.

The data was annotated by two native speakers who are trained linguists. For creating the gold standard, all controversial cases were discussed with a non-native speaker of the language who is a trained linguist. Table 2 gives agreement scores between the annotators and the gold standard. For this purpose, we also created two additional versions of the corpus where we mapped the manual annotations to two flavors of Universal Part-of-Speech Tags (cf. Section 4.).

The major differences between the annotators are visualized in Fig. 1. For this visualization, we map the tags to the coarser tagset used in the Universal Dependencies project (cf. the next section). As we can see, the two largest areas of disagreement between the annotators are within-class choices for pronouns and verbs. The annotators also disagree fairly often on whether a word is a particle, coordinating or subordinating conjunction. Fig. 2 gives a detailed visualization of the differences within the pronoun class. The major source of disagreement is the choice between PCl, i. e. pronominal clitics, and PClSubj, i. e. amalgamations of subjunctive particle and pronominal clitic. Fig. 3 visualizes the differences within the verb class. Major sources of disagreement are the choices between V and VPass, V and VSubjCl, and VPass and VRefl.

#	Tag	Name	Example
1	**N**	Noun	*hënë*
2	**N**A	Noun preceded by article	*e **hënë***
3	**N**Hg	Het. noun	*art* (sg. m.) vs. *arte* (pl. f.)
4	**N**E	Name	*Peja/Drini/Joni*
5	**V**	Verb (finite forms)	*tha*
6	**V**Part	Participle (non-finite forms)	*thënë*
7	**V**Cl	V. w. clitic	*i **tha***
8	**V**Impv	Imperative form	*Prit!/Fol!*
9	**V**ImpvCl	Imperative w. clitic	*Tregomëni!*
10	**V**Pass	V. w. pass. part. *u*	*u **tha***
11	**V**PassCl	V. w. pass. part. and clitic	*ua **tha***
12	**V**Subj	V. w. subj. particle	*të **thotë***
13	**V**SubjCl	V. w. pass. part. and cl.	*ta **tha***
14	**V**SubjPass	V. w. subj. a. pass. part.	*t'u **thotë***
15	**V**Aux	Auxiliar verb	*kam*
16	**V**Mod	Modal verb	*mund*
17	**V**Recp	Reciprocal verb	*njihen*
18	**V**Refl	Reflexive verb	*lahem*
19	**Adj**	Adjective	*djali **trim***
20	**Adj**A	Adj. preceded by article	*djali i **mirë***
21	**Adj**P	Preposed adj.	***trimi** djalë*
22	**Adj**PA	Prep. adj. prec. by art.	*i **miri** djalë*
23	**Adj**N	Noninflected adjective	*blu/neto*
24	**Adv**	Adverb	*mirë*
25	**Adv**Pt	Adv. prec. by part.	*së shpejti*
26	**Adv**MP	Multipart adverb	*kohë pas kohe*
27	**P**Pers	Personal pronoun	*ti*
28	**P**PersSF	Pers. pron. (short form)	*ta/të/to*
29	**P**Dem	Demonstrative pronoun	*ky/këta*
30	**P**DemA	PDem preceded by article	*i tillë*
31	**P**Poss	Possessive pronoun	*im*
32	**P**PossA	PPoss preceded by article	*i tij/të vetën*
33	**P**Int	Interrogative pronoun	*kush*
34	**P**IntA	PInt preceded by article	*i kujt/i cilit*
35	**P**IntÇ	Interrogative pronoun *Ç/ç*	***ç'**libër?*
36	**P**Rel	Relative pronoun	*që*
37	**P**RelA	PRel preceded by article	*i cili*
38	**P**Indef	Indefinite pronoun	*dikush*
39	**P**IndefA	PIndef preceded by article	*të tjerëve*
40	**P**Refl	Reflexive pronoun	*me **vete***
41	**Prep**	Preposition	*me/pa/nga/për*
42	**Conj**C	Coordinating conjunction	*dhe*
43	**Conj**S	Subordinating conjunction	*që*
44	**Conj**C1	First part of coord. conj.	***edhe** … edhe*
45	**Conj**C2	Second part of coord. conj.	*edhe … **edhe***
46	**Conj**S1	First part of subord. conj.	***le që** … por*
47	**Conj**S2	Second part of subord. conj.	*le që … **por***
48	**Num**C	Cardinal number	***dy** fitore*
49	**Num**O	Ordinal number	*fitorja e **dytë***
50	**Pt**	Particle	*ja*
51	**Pt**Comp	Comparative particle	***më** i mirë*
52	**Pt**Cond	Conditional particle *në/po*	*në/po …*
53	**Pt**Fut	Future particle	*do*
54	**Pt**Ger	Gerundive particle *duke*	***duke** ecur*
55	**Pt**Inf	Infinitive particle *për*	*për*
56	**Pt**Jus	Jussive particle *le*	*le*
57	**Pt**Mod	Modal particle	*mund*
58	**Pt**Neg	Negation particle	*nuk/mos/jo*
59	**Pt**NegD	Negation particle *dot*	*s'/nuk … **dot***
60	**Pt**NegS	Negation particle *s'*	***s'** punon*
61	**Pt**Pass	Passive particle *u*	*u*
62	**Pt**PassCl	Pass. part. with clitic	*iu*
63	**Pt**Priv	Privative particle *pa*	***pa** punuar*
64	**Pt**Prog	Progressive particle *po*	***po** lexon*
65	**Pt**Proh	Prohibitive particle	*mos*
66	**Pt**QA	Question particle *A/a*	***A** punon?*
67	**Pt**QM	Question particle *mos*	***Mos** iku?*
68	**Pt**Subj	Subjunctive particle *të*	*të*
69	**Intj**	Interjection	*o/hm/uh/ii*
70	**Art**	Article	*i/e/të/së*
71	**PCl**	Pronominal clitic	*i*
72	**PCl**2	2nd part of pron. clitic	*e* [in: *na e*]
73	**PCl**Subj	Subj. particle and PCl	*ta* [i. e. *të+e*]
74	**Abbr**	Abbreviation	*d.m.th./etj.*
75	**FW**	Foreign word/Non-Alban.	*web*
76	**Punct**	Punctuation (sent.-ending)	*. ? !*
77	**Punct**2	Punctuation (not sent.-end.)	*, : ; - –*
78	**NLE**	Non-linguistic element	*· § % …*
79	**EM**	Emoticon	*:-)*

Table 1: Tagset.

	full tagset	UD UPOS	Google UPOS
Ann1 vs. Ann2	90.63	92.83	94.07
Ann1 vs. Gold	91.89	93.67	94.72
Ann2 vs. Gold	98.64	99.12	99.33

Table 2: Agreement between annotators and gold standard for the full tagset and for versions mapped to the coarser tagsets. Values are accuracy percentages.

4. Mapping to Universal Part-of-Speech Tags

For some applications and use-cases, it is useful to have a more coarse-grained set of part-of-speech tags that makes a broad distinction between word-classes but abstracts away from most of the additional properties encoded in the tagset described above. Rather than invent a new coarse-grained set of labels, we adopt the popular Universal Part-of-Speech Tags that have become a quasi standard. There are two flavors of Universal Part-of-Speech Tags:

- Google UPOS, the original Google Universal Part-of-Speech Tagset by Petrov et al. (2012) that consists of 12 tags[2] and

[2] https://github.com/slavpetrov/universal-pos-tags

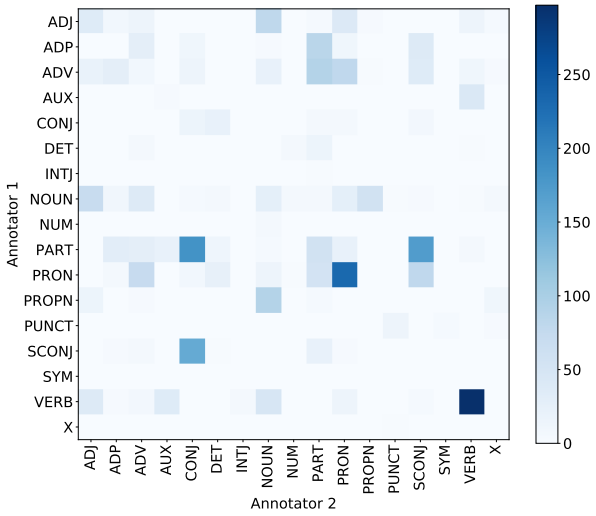


Figure 1: Tagging differences between the annotators. For clarity of presentation, we map the tags to UD UPOS tags, i. e. entries on the diagonal indicate differences between tags that get mapped to the same UD UPOS tag.

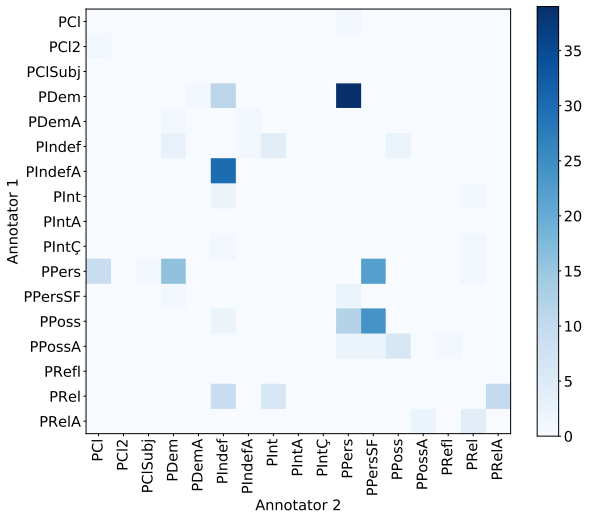


Figure 2: Tagging differences between the annotators (pronominal tags only).

- UD UPOS, the refined version used in the treebanks of the Universal Dependencies project (Nivre et al., 2016) that consists of 17 tags.[3]

In Table 3, we provide a mapping from our tagset to both varieties of the Universal Part-of-Speech Tagset.

The mapping is straightforward. All of our tags can be seen as refinements of the Universal Part-of-Speech Tagset – a deliberate design choice.

5. Evaluation

For our evaluation, we perform tagging experiments using the following part-of-speech taggers:

[3] http://universaldependencies.org/

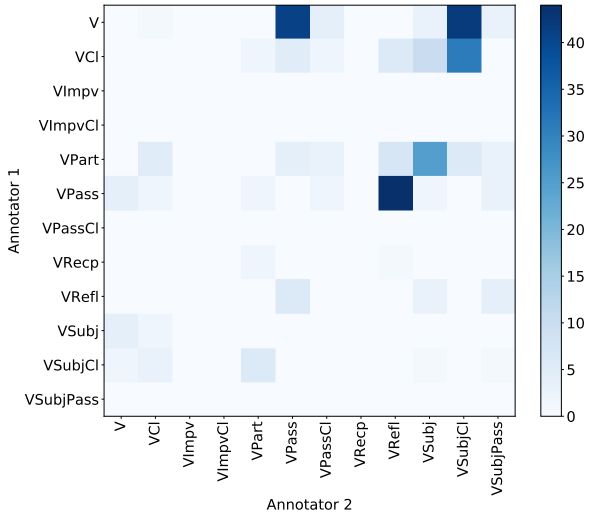


Figure 3: Tagging differences between the annotators (verbal tags only).

- The HMM-based HunPos tagger (Halácsy et al., 2007),[4] an open source reimplementation of Brants' TnT tagger (Brants, 2000).
- The maximum entropy tagger from the Apache OpenNLP project[5].
- TreeTagger (Schmid, 1994; Schmid, 1995),[6] a tagger based on decision trees.
- SoMeWeTa (Proisl, 2018),[7] a perceptron-based tagger that can make use of external resources. We provide it with Brown clusters (Brown et al., 1992) extracted from 82 million tokens of Albanian text.
- The Stanford POS Tagger (Toutanova et al., 2003),[8] a maximum entropy tagger that uses cyclic dependency networks.

We evaluate tagging accuracy in five scenarios using ten-fold cross-validation on our gold standard corpus. The five scenarios are:

1. Training and testing using our full tagset,
2. training and testing using UD UPOS,
3. training and testing using Google UPOS,
4. training using our full tagset and mapping the output to UD UPOS for testing and
5. training using our full tagset and mapping the output to Google UPOS for testing.

The results are shown in Table 4. At first glance, they seem to be rather modest. Using the full tagset, the best tagger achieves 91.00% accuracy – a good six points less than the state of the art for languages like English, French or German. However, we have to take into consideration that our corpus is orders of magnitude smaller than for example the Wall Street Journal part of the Penn Treebank and that it is at the

[4] https://code.google.com/archive/p/hunpos/

[5] https://opennlp.apache.org/

[6] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[7] https://github.com/tsproisl/SoMeWeTa

[8] https://nlp.stanford.edu/software/tagger.html

Google	UD	tags
ADJ	ADJ	Adj, AdjA, AdjP, AdjPA, AdjN, NumO
ADP	ADP	Prep
ADV	ADV	Adv, AdvPt, AdvMP
CONJ	CONJ SCONJ	ConjC, ConjC1, ConjC2 ConjS, ConjS1, ConjS2
DET	DET	Art
NOUN	NOUN PROPN	N, NA, NHg NE
NUM	NUM	NumC
PRON	PRON	PPers, PPersSF, PDem, PDemA, PPoss, PPossA, PInt, PIntA, PIntÇ, PRel, PRelA, PIndef, PIndefA, PRefl, PCl, PCl2, PClSubj
PRT	PART	Pt, PtComp, PtCond, PtFut, PtGer, PtInf, PtJus, PtMod, PtNeg, PtNegD, PtNegS, PtPass, PtPassCl, PtPriv, PtProg, PtProh, PtQA, PtQM, PtSubj
VERB	AUX VERB	VAux, VMod V, VPart, VCl, VImpv, VImpvCl, VPass, VPassCl, VSubj, VSubjCl, VSubjPass, VRecp, VRefl
X	INTJ SYM X	Intj EM, NLE Abbr, FW
.	PUNCT	Punct, Punct2

Table 3: Mapping table from our tagset to Universal Part-of-Speech Tags (both the Google and the Universal Dependencies variant).

same time much more heterogeneous. In addition, Albanian has a much richer morphological system than English and our tagset is more than 50% larger than that of the Penn Treebank.

Training the taggers using the full tagset and mapping the predicted tags to one of the UPOS tagsets eliminates within-class errors produced by the taggers and leads to much better results with up to 94.68% accuracy. For almost all taggers, this setting works better than training directly on one of the coarser tagsets, i. e. the taggers benefit from a more fine-grained internal representation. The exception to this rule is SoMeWeTa, the only tagger provided with additional external knowledge in the form of Brown clusters, which achieves better results of up to 95.10% when trained directly on the coarser tagsets.

The main sources of errors for SoMeWeTa on the full tagset are visualized in Figure 4. The largest group of errors is confusion between different verb tags. Next are confusions between different noun tags (mostly between N and NHg) and, to a lesser extent between verb and noun tags and between different tags for pronouns. The most frequent misclassifications for the 20 tags with the most errors are shown in Table 5. The single most difficult distinction for the tagger is that between N and NHg, i. e. between nouns and heterogeneous nouns. This is not surprising given that

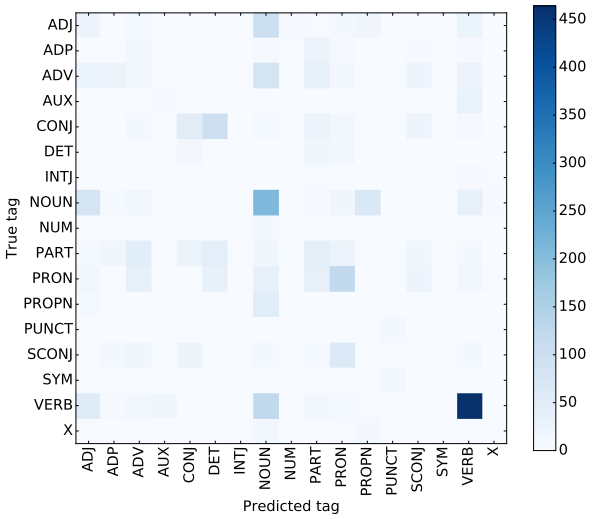


Figure 4: Errors made by SoMeWeTa on the full tagset. For clarity of presentation, we map the tags to UD UPOS tags, i. e. errors on the diagonal indicate misclassifications between tags that get mapped to the same UD UPOS tag.

this is a lexical distinction that cannot be deduced from context. To be able to correctly distinguish between the two, the tagger must have seen the word during training or must be provided with a corresponding lexicon. Other frequent misclassifications include noun (N) vs. proper noun (NE), noun (N) vs. adjective (ADJ) and coordinating conjunction (ConjC) vs. article (Art).

6. Conclusion and Future Work

Up to now, there has been no manually tagged corpus of substantial size for Albanian. The gold standard presented in this paper is by far the largest manually tagged corpus for Albanian and consists of 2,020 sentences (31,584 tokens) annotated with a medium-sized part-of-speech tagset designed with a focus on the syntagmatic aspects of the language, especially multi-word units involving articles or particles. While the corpus is still too small to achieve state-of-the-art tagging accuracies comparable with those for better-resourced languages, the evaluation experiments show very promising results similar to what could be expected from an English corpus of similar size. For the coarser tagsets, we achieve accuracies of up to 95.10%. We are also optimistic that the results could be further improved by providing the taggers with an additional lexicon. Such a lexicon could be derived from one of the existing morphological analyzers.

7. Bibliographical References

Arkhangelskiy, T., Belyaev, O., and Vydrin, A. (2012). The creation of large-scale annotated corpora of minority languages using UniParser and the EANC platform. In *Proceedings of COLING 2012: Posters*, pages 83–92. The COLING 2012 Organizing Committee.

Brants, T. (2000). TnT – A statistical part-of-speech tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP-2000)*, pages 224–231.

Brown, P. F., Pietra, V. J. D., de Souza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of

tagger	full tagset	UD UPOS	Google UPOS	mapped to UD UPOS	mapped to Google UPOS
HunPos	88.73 ±0.79	92.44 ±1.18	93.63 ±1.18	92.72 ±0.86	94.04 ±0.78
SoMeWeTa	**91.00** ±0.83	**94.32** ±1.40	**95.10** ±1.38	**93.99** ±0.97	**94.68** ±0.97
TreeTagger	88.14 ±1.12	91.15 ±1.69	92.15 ±1.87	92.24 ±1.20	93.43 ±1.17
OpenNLP	87.50 ±1.23	89.87 ±1.39	91.24 ±1.35	90.64 ±1.35	91.98 ±1.20
Stanford Tagger	85.96 ±1.26	89.72 ±1.28	91.37 ±1.24	89.92 ±1.59	91.36 ±1.49

Table 4: Evaluation results (mean accuracy percentages ±2 standard deviations).

tag	freq	err	most frequent confusions
N	5470	233	NE (63), NHg (45), Adj (35), V (25)
Adv	965	198	N (67), ConjS (21), Prep (20), Pt (20)
V	1562	178	N (62), VRefl (32), VCl (22), VSubj (18)
ConjC	1234	172	Art (98), ConjS (19), Pt (16), PCl (10)
NHg	493	155	N (141), NE (3), VRefl (3), Adj (2)
VRefl	401	150	V (56), VPass (35), VSubj (19), N (11)
ConjS	656	133	PRel (56), ConjC (20), Adv (18), N (10)
Pt	292	131	Adv (25), ConjC (20), ConjS (15)
Adj	904	112	N (69), NE (10), V (8), AdjA (8)
VCl	486	82	V (14), N (13), AdjA (13), VRefl (11)
VPass	142	82	VRefl (57), V (14), VPart (3), PtPass (2)
VSubj	376	70	VPart (15), AdjA (14), VSubjCl (9)
PIndef	308	69	Adv (20), N (19), PIndefA (8), PInt (4)
PCl	694	68	Art (29), PtPass (14), PtSubj (9)
NE	712	63	N (54), Adj (3), AdjA (3), PCl (1)
Prep	2292	63	Adv (14), PtInf (10), PtComp (10), V (5)
NA	124	57	AdjA (28), N (16), PIndefA (5), VCl (2)
AdjA	1118	54	N (17), NA (7), VCl (6), NE (5)
VPart	526	52	N (20), AdjA (8), VSubjCl (8), VCl (3)
Art	2996	46	PtSubj (16), PCl (11), ConjC (11)

Table 5: The 20 tags that SoMeWeTa most frequently misclassified.

natural language. *Computational Linguistics*, 18(4):467–479.

Buchholz, O. and Fiedler, W. (1987). *Albanische Grammatik*. VEB Verlag Enzyklopädie, Leipzig.

Demiraj, S., Agalliu, F., Agoni, E., Dhrimo, A., Hysa, E., Lafe, E., and Likaj, E. (1995). *Morfologjia*, volume 1 of *Gramatika e Gjuhes Shqipe*. Akademia e Shkencave e Republikës së Shqipërisë, Tiranë.

Halácsy, P., Kornai, A., and Oravecz, C. (2007). HunPos: An open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 209–212, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hasanaj, B. (2012). *A Part of Speech Tagging Model for Albanian*. Lambert Academic Publishing, Saarbrücken.

Kabashi, B. and Proisl, T. (2016). A proposal for a part-of-speech tagset for the Albanian language. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4305–4310, Paris, France. European Language Resources Association.

Kabashi, B. (2015). *Automatische Verarbeitung der Morphologie des Albanischen*. FAU University Press, Erlangen.

Kabashi, B. (2017). AlCo – një korpus tekstesh i gjuhës shqipe me njëqind milionë fjalë. In *Seminari XXXVI Ndërkombëtar për Gjuhën, Letërsinë dhe Kulturën Shqiptare*. Universiteti i Prishtinës, Prishtinë.

Kadriu, A. (2013). NLTK tagger for Albanian using iterative approach. In *Proceedings of the 35th International Conference on Information Technology Interfaces (ITI)*.

Leech, G. and Wilson, A. (1999). Standards for tagsets. In Hans van Halteren, editor, *Syntactic Wordclass Tagging*, pages 55–80. Kluwer Academic Publishers, Dordrecht.

Newmark, L., Hubbard, P., and Prifti, P. (1982). *Standard Albanian – A Reference Grammar for Students*. Stanford University Press, Stanford, CA.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666, Paris, France. European Language Resources Association.

Petrov, S., Das, D., and McDonald, R. T. (2012). A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012), Istanbul, Turkey, May 23-25, 2012*, pages 2089–2096.

Piton, O. and Lagji, K. (2008). Morphological study of Albanian words, and processing with NooJ. In *Proceedings of the 2007 International NooJ Conference*, pages 189–205. Cambridge Scholars Publishing.

Piton, O., Lagji, K., and Përnaska, R. (2007). Electronic dictionaries and transducers for automatic processing of the Albanian language. In *Proceedings of the 12th International Conference on Applications of Natural Language to Information Systems (NLDB 2007)*, pages 407–413. Springer, Berlin, Heidelberg.

Proisl, T. (2018). SoMeWeTa: A part-of-speech tagger for German social media and web texts. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki. European Language Resources Association.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to German. In *Proceedings of the EACL SIGDAT-Workshop*, pages 47–50, Dublin.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In Marti A. Hearst et al., editors,

Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003).

Trommer, J. and Kallulli, D. (2004). A morphological analyzer for standard Albanian. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1271–1274. European Language Resources Association.

8. Language Resource References

Péter Halácsy and András Kornai and Csaba Oravecz. (2007). *HunPos*. `https://code.google.com/archive/p/hunpos/`.

Thomas Proisl. (2017). *SoMeWeTa*. `https://github.com/tsproisl/SoMeWeTa`.

Apache OpenNLP project team. (2004). *Apache OpenNLP*. `https://opennlp.apache.org/`.

Helmut Schmid. (1994). *TreeTagger*. `http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`.

Kristina Toutanova and Dan Klein and Christopher Manning and William Morgan and Anna Rafferty and Michel Galley and John Bauer. (2004). *Stanford Log-linear Part-Of-Speech Tagger*. `https://nlp.stanford.edu/software/tagger.html`.

Morphology Injection for English-Malayalam Statistical Machine Translation

Sreelekha S, Pushpak Bhattacharyya
IIT Bombay, Mumbai, India
{sreelekha, pb} @cse.iitb.ac.in

Abstract

Statistical Machine Translation (SMT) approaches fails to handle the rich morphology when translating into morphologically rich language. This is due to the data sparsity, which is the missing of the morphologically inflected forms of words from the parallel corpus. We investigated a method to generate these unseen morphological forms. In this paper, we analyze the morphological complexity of a morphologically rich Indian language Malayalam when translating from English. Being a highly agglutinative language, it is very difficult to generate the various morphological inflected forms for Malayalam. We study both the factor based models and the phrase based models and the problem of data sparseness. We propose a simple and effective solution based on enriching the parallel corpus with generated morphological forms. We verify this approach with various experiments on English-Malayalam SMT. We observes that the morphology injection method improves the quality of the translation. We have analyzed the experimental results both in terms of automatic and subjective evaluations.

Keywords: Morphology Injection, Statistical Machine Translation, English-Malayalam Machine Translation

1. Introduction

Malayalam, Telugu, Kannada and Tamil are being most prominent out of 17 languages in the southern Indian family of Dravidian languages; about 95 per cent of the South Indian population speaks one of these four languages. Out of these over 38 million people is speaking Malayalam primarily in the state of Kerala. Throughout its gradual evolution of the present day Malayalam, the influence of Sanskrit is evident in the alphabet, phonology and vocabulary and to a lesser extent in the morphology of Malayalam. Malayalam is a highly agglutinative and inflectionally rich language with a free word order. The semantic and syntactic relations between the verbs and other constituents in a sentence are represented by the case endings of the words. V*aachakam*, which denotes the matter, action and quality, and *dyootakam*, which denotes the relationships, are the two types of "*Sabdam*", a combination of sounds with a meaning. *Naamam* (noun), *kriyaa* (verb) and *bheedakam* (modifier) are the three types of *Vaachakam*. *gati* (preposition), *ghatakam* (conjunction) and *vyaakseepakam* (interjection) are the three types of *dyootakam* (Varma, 2000). Malayalam has a strong postpositional inflections with highly agglutinative suffixes (Namboodiri, 1998). These inflections carry information about tense, mood and aspect for verbs and cases (accusative, dative, etc.), gender, number, person information for nouns.

Most approaches to Statistical Machine Translation, i.e., phrase based models (Koehn, Och and Marcu, 2003), syntax based models (Yamada and Knight 2001) do not allow incorporation of any linguistic information in the translation process. The introduction of factored models (Koehn and Hoang, 2007) provided this missing linguistic touch to the statistical machine translation.

Factored models (Koehn and Hoang, 2007) treat each word in the corpus as vector of tokens. Each token can be any linguistic information about the word which leads to its inflection on the target side. Hence, factored models are preferred over phrase based models (Koehn, Och and Marcu, 2003) while translating from morphologically poor language to morphologically richer language. There were many attempts to improve the quality of SMT systems such as; using Monolingually-Derived Paraphrases(Marton et al., 2009), Using Related Resource-Rich languages (Nakov and Ng, 2012), (Minkov et. al., 2007) . In this paper, we study SMT models and the problem of sparseness and morphological complexity in the context of translation to a highly agglutinative, morphologically rich language Malayalam from English. There are many ongoing attempts to develop MT systems for Indian languages (Antony, 2013; Bharathi et. al., 1996; Kunchukuttan et al., 2014; Nair et. al., 2012; Sreelekha et al., 2013; Sreelekha et al., 2015; Sreelekha et al., 2015; Sreelekha et al., 2016; Sreelekha et al., 2018) using both rule based and statistical approaches. Even though there were many attempts to develop Machine Translation systems between English and Malayalam, the complexity of morphology, especially the word compounding phenomena and the various derivation morphology forms makes the translation quality worse. In this paper we propose a simple and effective solution to handle the morphological complexity which is based on enriching the input with various morphological forms of words. The flow of the paper is as follows: Section 2 describes Morphological Phenomena in Malayalam; Section 3 describes Morphology Generation technique; Section 4 describes the experimental discussion and section 5 describes the conclusion.

2. Morphological Phenomena in Malayalam

Malayalam poses many morphological variations due to the Inflections, Derivations and Word compounding features. Noun or verb is attached to suffixes to generate words of the same category in inflectional morphology. On the other hand noun or verb with a suffix attached to it generates a word of new category in Derivational morphology. New words are formed by combining a noun and a noun, noun and adjective, verb and noun, adverb and verb, adjective and noun and in some cases all the words of an entire sentence to reflect the semantics of the sentence in word compounding (Jurafsky 2003).

2.1 Nouns, Pronouns and Verbs Characteristics

GNP, the Gender (Masculine, feminine and neuter), Number (singular or plural) and Person (person, second person or third person) information is used for the processing of nouns and pronouns. For proper nouns and abstract nouns there is no plural. Nominative, accusative, sociative, dative, instrumental, genitive and locative are the seven cases in Malayalam. Inflections of verbs can be generated by considering tense, mood and aspect. There are two main groups of verb classification: *kaaritam*, simple verbbases with '*ക്കു*' *(kku)* (eg.*ചിരിക്കുക (chirikkuka/ to laugh)*; *akaaritam*, simple verb-bases which do not contain '*ക്കു*'*(kku) (e.g. ചാടുക (chaaduka/ to jump)*). Past finite, present finite, future finite, negative past, negative present, negative future, or infinitive are the Tense forms. There are direct imperatives (singular and plural), indirect imperatives and negative imperatives. Participles can be verb participle (positive and negative), conditional participle (positive and negative), concessive participle (positive and negative), relative participle (past, present, future, and negative). Aspect can be habitual, trial, completive, durative, reflexive, or perfective. Mood can be expressed as possibility (positive and negative), obligatory (positive and negative), inceptive, ability (past, present, future), or causative (past, present, future). Past tense markers in Malayalam is based on the verb base (*kaaritam* or *akaaritam*) and on the phoneme quality of the last character in the root such as palatal, labial, *vyanjana* or *chil* (Varma, 2000).

2.2 Derivational Morphology for nouns

Adjectives, adverbs and verbs can be derived from nouns by adding proper suffixes. Modifiers are qualifiers and of three types: *naamavisheshanam* (adjective), when it modifies a noun; *kriyaavisheshanam* (adverb), when it modifies an adverb; *bhedakavisheshana* (modifier of modifier), when it modifies a modifier. The modifiers can be pure modifiers (such as, determinative adjectives, superlative adjectives, interrogative adjectives, temporal adverbs, special adverbs and adverbs of manner) or those derived from nouns and verbs. The Table 1 shows the inflectional and derivational morphology for nouns commonly found in Malayalam.

2.3 Derivational morphology for Verbs

A new category of word is generated by attaching noun or verb with a suffix. The derivations considered are:

i) participles (verbal participle, conditional participle, concessive participle and relative participle.

For example, *കഴിഞ്ഞ (kazhinja* / over-relative participle), *കഴിഞ്ഞാൽ* (*kazhinjaal* / if over-conditional participle).)

ii) infinitives (The suffix taken by infinitives is "*aan*". For example, *വരു (vaRz/ come) + ആൻ (aan / to) = വരാൻ (vaRaan / to come)).*

Table 2 shows the inflection generation forms for a verb "varuka".

		type	Suffix replacement	Rule(end chatr)	Example	
					root	Inflected form
Inflections	Plural		/kal	i	/kutti	/kuttikal
	Case	Nominative	No suffix	-	/krishnan	/krishnan
		Accusative	/e	/n	/krishnan	/krishnane,
			/ine	/u	/indu	/induvine
		Dative	/odu	/i	/kavi	/kaviyodu
			/inodu	/u	/tanu	/tanuvinodu
		Sociative	/ikku	/i	/rathi	/rathikku
			/inu	/u	/indu	/induvinu
		Instrumental	/aal	/i	/tadi	/tadiyaal
			/inal	/u	/indu	/induvinal
		Genitive	/inte	/a	/kavita	/kavitayude
			/ude	/n	/mohan	/mohante
Derivation	adjective	Quality	/aaya	/n	/nallavan	/nallavanaya
			/ulla	/i	/bhangi	/bhangiyulla
		Place	/ile	/m	/maram	/marathile
	Adverb	Manner	/aayi	/i	/bhangi	/bhangiyayi
		Direction	/ekkz	/u	/kadz	/kattilekkz

Table 1 : Inflection generation rules for Noun

2.4 The complexity in Word compounding

Type		Imperative forms	Suffix	Example Inflection
Tense		Past	/um	/varum
		Present	/unnu	/vannu
Aspect	Perfect	Present	/ittundu	/vannittundu
			/irikkunnu	/vannirikkunnu
		Past	/ikkazhinjirikkunnu	/vannethikazhinjirikkunnu
			/ittittundayirunnu	/vannittundayirunnu
		Future	/irikkum	/vannirikkum
			/ittundakum	/vannittundakum
	continuous	Present	/kontirunnu	/vannukontirunnu
		Past	/iriikukkayayirunnu	/vannukontirikkukayayirunnu
	Imperfective		/unnundu	/varunnundu
			/ukayanau	/varukayanu
	Ingressive		/aarayi	/vararayi
	Other Auxiliaries		/poyi	/vannu poyi
			/kalanju	/vannukalanju
Mood	optative		/atte	/varatte
	Intentional		/aam	/varaam
	Debititive		/anam	/varanam
			/Etheeru	/vannetheeru
	Debititive (-ve)		/anta	/varanta
			/ikkooda	/varakkoda
			/aanpadilla	/varanpadilla
	Ability		/aam +dative subject	/varaam
			/aankazhinju	/varankazhinju
			/aansadhichu	/varan sadhichu
			/aanothu	/varanothu
	Permission (+ve)		/am	/varam
			/ate	/varatte
			/oloo	/vannoloo
	Permission (-ve)		/aruthu	/vararuthu
	Degree of certainity		/Ekkaam	vannekkam
			/umayirikkam	varumayirikkam
	Authority for assertion		/athre	/varumathre
			/ennukettu	/varumennukettu

Table 2: Inflection generation forms for a verb "varuka"

Word compounding (*sandhi*) is called the sound changes when two words or suffixes join and it is common in Malayalam. Consider an example to understand it's complexity, were five words are joined as a single word;

Malayalam: *ഞാനിന്നൊരാളെപ്പറ്റിച്ചു*

{njaaninnoraleppattichu}

English: *I fooled one person today.*

Panini has classified the word compounding according to the position in which the compounding occurs, such as:

- word_medial (padamadhyam) occurs between a stem and a suffix;
- word final (padaanta) occurs between two words;
- hybrid (ubhaya), both word medial and word final involves.

Malayalam compounding rules are also classified as:

1) Vowel sandhi*: വാഴ (vazha/ rain)+അല്ല (alla/not)= വാഴയല്ല (vazhayalla)*

2) Vowel- consonant sandhi: *താമര (thaamara/lotus)+ കുളം (kuLam /pond)= താമരക്കുളം (thaamarakkulam / lotus pond)*

3) Consonant -consonant sandhi:- *വേനൽ (venal*/summer) + *അവധി* (*avadhi*/leave)= *വേനലവധി* (*venalavadhi / summer leave).*

Word 1st end	Word-2nd beg	Substitution	Example
/am	Vowel(v2)	(1,1,/ma+ss(v2))	varam+alla = varamalla
/am	Vowel(v2),word class=casemarker	(1,1,/tha+ss(v2))	varam+e= varathe
/N	/da, /tha, /na	(1, 1, /nta, NNa resp.)	TaN+taar=tanN Taar
/am	/ka,/cha,/da, /tha, /pa	/nka, /ncha, /nda, /nta, /mpa, /nga	varam+kal = varangal
/N, n, L, I, r	Vowel(v2)	(-, 1, /Na, /na, /La, /la, /ra	Aval + il = avalil

Table 3: Substitution (Aadesa Sandhi)

Word-1stend	Word-2ndbeg	Substitution	Example
Vowel	Vowel	(-,1,ya+ss(v2))	tara+odu = tarayodu
/U, /uu	Vowel	(-,1,/va+ss(v2))	rhitu+aayi=rhituvayi
Word-1=/a, /I, /e	Vowel	(-, 1, va, ss(v2)	e+ ir = ivar
/a	consonant	(-, -, ss(aa))	kala+mELa = kalaamELa

Table 4: Addition Rules (Aagama Sandhi)

Word-1st -end	Word-2nd begining	Substitution	Example
/z	Vowel	(1,1,ss(v2))	veedz+il = veettil
Word-1= /alla, /illa	Vowel	(-,1,ss(v2))	alla+ennz=allennz
Word-1=/oru	Vowel	(1,1, ss(v2)	oru+aal = oraal
Word-1=/aayi, /pOyi	Vowel	(-,1, s(v2))	poyi+ennu = poyennu
Word-1=/u,/um, Word-1 category = verb	Vowel	(1,1, ss(v2))	Pokum + illa = pokilla

Keralapanini (Varma, 2000) has classified the compounding rules based on the changes occur during the compounding as:

- *lOpa sandhi* (elison), one of the sounds is lost;
- *aagama sandhi* (addition), new sound is added;
- dvitva sandhi (germination or reduplication), one of the sounds geminates;
- *aadEsa sandhi* (displacement or substitution), one of the sounds is displaced by another sound.

Table 5: Elision Rules (Lopa Sandhi)

The rules which we have created for compounding in each category are given in Tables 3, 4, 5 and 6; Table 3 shows the Substitution Rules (AAdesa Sandhi), Table 4 shows the Addition Rules (Aagama Sandhi), Table 5 shows the Elision Rules (Lopa Sandhi). The morphophonemic changes at the boundary depend on the ending vowel or consonant, the category of the first or the second word and the beginning vowel or consonant of the second word.

2.5 Sanskrit compounding

Word 1stend	Word-2nd -beg	Substitution	Example	Sandhi
/a	/a, /aa	(-1,ss(aa))	Padya+ avasanam = padyaavasanam	Deergha Sandhi
/i	/I, /ii	(1,1,ss(ii))	Kavi +iisvaran = kaviisvaran	
/u	/u, /uu	(1, 1,ss(/uu))	Guru+upakaram = guruupakaram	
/A	/i, /u	(,1,ss(ee/oo))	Sara+upadesam = saaroopdaesam	Guna Sandhi
/aa	/E	(1,1,ss(/ya))	Sada+eevam =sadaivam	Vridhi Sandhi
/i	Vowels except / e	(1,1,ss(/ya))	Athi+aavasyam = Athyavasyam	
/u	/a, /aa	(1,1,ss(/va))	Uru+aagamanam = gurvagamanam	YaNa Sandhi

Table 6: Sanskrit Compounding Rules

Malayalam and Sanskrit shares many of the compounding rules since Malayalam is derived from Sanskrit. Compounding in Sanskrit are:

- vowel compounding *(swarasandhi), joining of* two vowels;
- consonant compounding *(vyanjanasandhi),* consonants join.

D*eerghasandhi, guNasandhi, vridhisandhi and yaNsandhi are the further classifications of Vowel Sandhi*. Table 6 shows the Sanskrit compounding rules.

3. Morphology Generation Technique

The SMT systems face the problem of data sparsity; the data does not have enough inflectional forms when translating from a morphologically poor language to a morphologically rich language. Another case is that data sparseness arises only when using factored models. To handle this, we need to generate all combinations of the factors used. We have used a Morphology injection method that generates various morphological forms of noun and verb entities by classifying them and augments the training data with newly generated morphological forms of nouns. The basic algorithm of the Morphology injection method can be described as below:

1. Find out the noun/verb entity pairs (Eng-Mal)
2. Categorize Malayalam nouns/verbs into classes
3. Generate new morphological forms of the nouns using the rules
4. Augment the training data with the generated inflected forms

We have created rules for handling the inflections in noun and verb. Moreover, for handling the word compounding phenomena in Malayalam, we have created rules for elision, substitution, addition and for Sanskrit compounding and are presented in Tables 1, 2, 3, 4, 5 and 6. Then we have generated the respective inflected forms using the created rules with the help of a parallel dictionary of root words between English and Malayalam.

3.1 General Factored model for handling morphology

Factored translation models allow additional annotation at the word level by considering word as a vector of tokens. Factored translation models can be seen as the combination of several components (language model, reordering model, translation steps, and generation steps). These components define one or more feature functions that are combined in a log linear model [Koehn and Hoang, 2007]:

$$p(e|f) = \frac{1}{Z}\sum_{i=1}^{n} \lambda_i h_i(e,f) \quad (1)$$

From equation (1), each h_i is a feature function for a component of the translation, the λ_i values are weights for the feature functions and Z is the normalization constant.

Figure 1 shows a general factored model approach for translation from a morphologically poor language to a morphologically rich language. On the source side we have: Surface word, root word, and set of factors *S* that affect the inflection of the word on the target side. On the target side, we have: Surface word, root word, and suffix (can be any inflection). The model has the following mapping steps:

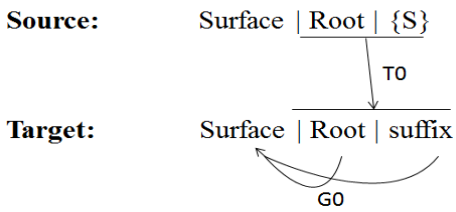


Figure 1: Factored model setup to handle inflections

- **Translation step (T0):** Maps source root word and factors in *S* to target root word and target suffix

  Generation step (G0): Maps target root word and suffix to the target surface word. Note that the words which do not take inflections have *null* as values for the factors in *S*.

Figure 2 shows the factored model setup to handle nominal inflections in Malayalam.

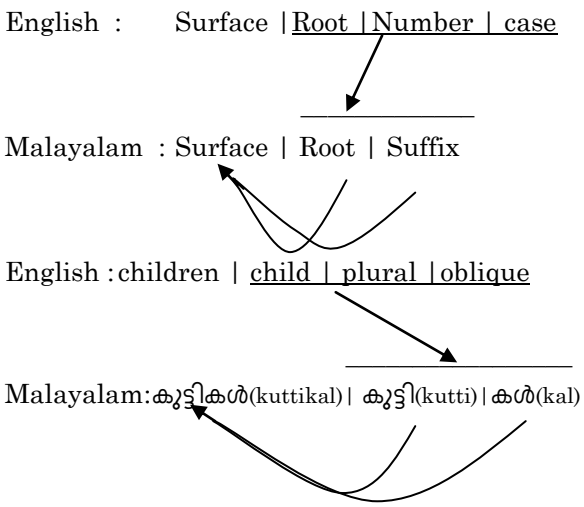


Figure 2: Factored model setup to handle nominal inflections

4. Experimental Discussion

We performed the experiments on ILCI (Indian Languages Corpora Initiative) English-Malayalam dataset. Domain of the corpus is health and tourism. We used 46,000 sentence pairs for training and 3000 sentence pairs for testing. The inflected-form dictionary was created using the Malayalam word lexicon. It consisted of 50,000 noun forms and 150,000 verb forms of Malayalam. The generated verb and noun forms have been validated manually over a period of 6 months with an English-Malayalam bilingual expert who is having a Master's degree in Malayalam Literature. Table 7 shows the statistics of the corpus used for training, testing, tuning and the generated word-form dictionary. *Moses*[1] toolkit was used for training and decoding. Language model was trained on the target side corpus with *IRSTLM*[2].

Sl. No	Corpus Source	Training Corpus	Corpus Size [Parallel Sentences]
1	ILCI	Health	23000
2	ILCI	Tourism	23000
Total			46000

Sl. No	Corpus Source ILCI	Tuning Corpus (MERT) size	Testing Corpus Size [Parallel Sentences]
1	Tourism	500	1500
2	Health	500	1500

Generated inflected form dictionary	Parallel DictionarySize
Noun	1,00,000
Verb	1,50,000
Total	2,50,000

Table 7: Statistics of the corpus used

For our experiments, we compared the translation output of the following systems:

- Phrase-based (unfactored) model (**Phrase**);
- Basic factored model for solving noun and verb morphology (**Fact**);
- Phrase-based model trained on the corpus used for *Phr* augmented with the word form dictionary for solving noun and verb morphology (**Phrase-Morph**);
- Factored model trained on the corpus used for *Fact* augmented with the word form dictionary for solving noun and verb morphology (**Fact-Morph**).

[1] http://www.statmt.org/moses/
[2] https://hlt.fbk.eu/technologies/irstlm-irst-languagemodeling-toolkit

With the help of syntactic and morphological tools, we extract the number and case of the English nouns and number, person, tense, aspect and modality of the English verbs. We have followed both the automatic evaluation (BLEU score (Papineni et al., 2002)) and subjective evaluation procedure with the help of linguistic experts as described in (Sreelekha et.al.(2013) for evaluating the systems. Table 8 shows the experimental results in terms of BLEU score evaluation and Table 9 shows the experimental results in terms of subjective evaluation (Fluency and Adequacy). For evaluation, we randomly chosen 250 translation outputs from each system were manually given adequacy and fluency scores. The scores were given on the scale of 1 to 5 going from worst to best, respectively. The BLEU score and subjective evaluations shows promising improvements in terms of the improvement of translation quality for both the Phrase and Factor based models.

Morph Problem	Model	Adequacy	Fluency
		En-Ml	En-Ml
Noun	Fact	28%	35%
	Fact-Morph	56%	67.32%
Verb	Fact	31.43%	45.09%
	Fact-Morph	58.89%	71.23%
Noun & Verb	Fact	35.67%	46.02%
	Fact-Morph	51.36%	65.34%
Noun	Phrase	26.87%	36.21%
	Phrase-Morph	50.56%	64.67%
Verb	Phrase	27.87%	37.12%
	Phrase-Morph	52.56%	64.87%
Noun & Verb	Phrase	33.87%	36.12%
	Phrase-Morph	57.56%	68.12%

Table 9: Subjective Evaluation Results-Morphology Injection

Morph Problem	Model	BLEU Score	
		Without Tuning	With Tuning
		En-Ml	En-Ml
Noun	Fact	26.17	28.23
	Fact-Morph	32.42	33.45
Verb	Fact	26.54	28.82
	Fact-Morph	36.54	38.30
Noun & Verb	Fact	24.01	26.08
	Fact-Morph	31.56	32.65
Noun	Phrase	26.78	29.01
	Phrase-Morph	31.30	33.12
Verb	Phrase	26.98	29.17
	Phrase-Morph	32.41	35.56
Noun & Verb	Phrase	27.51	29.92
	Phrase-Morph	35.03	37.73

Table 8: BLEU score Evaluation of Morphology Injection

5. Conclusion

SMT approaches suffer due to the morphological complexity when translating into a morphologically rich language. We solve this problem by enriching the original data with the missing morphological forms of words in Malayalam. Morphology injection performs very well and improves the translation quality. We observe huge improvement in BLEU score, adequacy and fluency of the translation outputs. We observe up to 38.30 improvements in BLEU score, up to 58.89% improvement in adequacy and up to 71.23% improvement in fluency. This method is more effective when used with factored models than the phrase-based models. A possible future work is to generalize the approach of morphology generation and verify the effectiveness of morphology injection on more morphologically complex languages.

6. Acknowledgments

The authors would like to thank Dr. Suman Mary Idicula and Dr. Latha R. Nair of CUSAT for their helps during this work. This work is funded by Department of Science and Technology, Govt. of India under Women Scientist Scheme-WOS-A with the project code-SR/WOS-A/ET-1075/2014.

7. Bibliographical References

Antony P. J. 2013. *Machine Translation Approaches and Survey for Indian Languages*, The Association for Computational Linguistics and Chinese Language Processing, Vol. 18, No. 1, March 2013, pp. 47-78

Bharathi A. and Sangal, V.R. (1996) *Natural Language Processing – A Paninian Perspective*. New Delhi: PHI.

Jurafsky D., Martin H.M., "Speech and natural language processing", Prentice Hall, 2003.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2007. *Statistical phrase-based translation*. NACAL on Human Language Technology- Volume 1. ACL.

Koehn, Philipp and Hieu Hoang. 2007. *Factored Translation Models*. In Proceedings of the 2007 Joint Conference on EMNLP and CoNLL, pages 868–876.

Kunchukuttan, A., Mishra, A., Chatterjee, R., Shah, R., and Bhattacharyya, P. (2014) *Shata-Anuvadak: Tackling Multiway Translation of Indian Languages*, LREC 2014, Rekjyavik, Iceland, 26-31.

Marton, Y., Callison-Burch, C. and Resnik, P. (2009) *Improved Statistical Machine Translation Using Monolingually-derived Paraphrases*, EMNLP, Volume 1- Pages 381- 390.

Minkov, E., Toutanova, K., and Suzuki, H. 2007. Generating complex morphology for machine translation. ACL 07, pages 128–135.

Nair, Latha R. and Peter S. David 2012. *Machine Translation Systems for Indian Languages*,

International Journal of Computer Applications (0975 – 8887), Volume 39– No.1, February 2012.

Nakov, P. I. and Ng, H. T. (2012). Improving Statistical Mahcine Translation for a Resource-Poor Language Using Related Resource-Rich Languages, Journal of AI Research, Volume 44, pages 179-222.

Namboodiri, E.V.N. (1998) *Kerala Bhasha Vyakaranam*. Kottayam: DC Books., Kerala, India.

Och, F. J. and Ney, H.. *A Systematic Comparison of Various Statistical Alignment Models.* Computational Linguistics, 2003.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. (2002). *BLEU: a Method for Automatic Evaluation of Machine Translation*, ACL Philadelphia, pp. 311-318.

Sreelekha, Dabre, R., Bhattacharyya, P. (2013). *Comparison of SMT and RBMT, The Requirement of Hybridization for Marathi–Hindi MT,* ICON.

Sreelekha. S, Bhattacharyya, P., Malathi.D. (2018). "Statistical vs. Rule Based; A Case Study on Indian Language Perspective", Springer Journel of Advances in Intelligent Systems and Computing Vol. 632, DOI 10.1007/978-981-10-5520-1_60.

Sreelekha, Bhattacharyya, P., Malathi, D. (2015). *A Case Study on English- Malayalam Machine Translation*, iDravidian Proceedings.

Sreelekha, Dungarwal, P., Bhattacharyya, P., Malathi, D. (2015). *Solving Data Spasity by Morphology Injection in Factored SMT*, ACL-Anthology, International conference on NLP - ICON 2015.

Sreelekha, Bhattacharyya, P.. (2016). *Lexical Resources to enrich English Malayalam Machine Translation*, LREC .

Varma, A.R.R.R. (2000) *Kerala Paaniniiyam*. Kottayam: DC Books, kerala, India.

Yamada, Kenji. And Knight, Kevin. 2001. *A Syntax-based Statistical Translation Model,* ACL.

The Morpho-syntactic Annotation of Animacy for a Dependency Parser

Mohammed Attia, Vitaly Nikolaev, Ali Elkahky
Google Inc.
New York, USA
{attia,vitalyn,alielkahky}@google.com

Abstract

In this paper we present the annotation scheme and parser results of the animacy feature in Russian and Arabic, two morphologically-rich languages, in the spirit of the universal dependency framework (McDonald et al., 2013; de Marneffe et al., 2014). We explain the animacy hierarchies in both languages and make the case for the existence of five animacy types. We train a morphological analyzer on the annotated data and the results show a prediction f-measure for animacy of 95.39% for Russian and 92.71% for Arabic. We also use animacy along with other morphological tags as features to train a dependency parser, and the results show a slight improvement gained from animacy. We compare the impact of animacy on improving the dependency parser to other features found in nouns, namely, 'gender', 'number', and 'case'. To our knowledge this is the first contrastive study of the impact of morphological features on the accuracy of a transition parser. A portion of our data (1,000 sentences for Arabic and Russian each, along with other languages) annotated according to the scheme described in this paper is made publicly available (`https://lindat.mff.cuni.cz/repository/xmlui/handle/11234/1-1983`) as part of the CoNLL 2017 Shared Task on Multilingual Parsing (Zeman et al., 2017).

Keywords: animacy, Arabic, Russian, Dependency Parsing

1. Introduction

The explicit encoding and identification of the animacy of entities in data has been reported to improve NLP tasks such as syntactic function disambiguation (Øvrelid, 2004; Lamers, 2007), anaphora resolution (Orasan and Evans, 2007a; Liang and Wu, 2004; Singh et al., 2014), syntactic parsing (Marton et al., 2011; Ambati et al., 2010; Nivre et al., 2008; Bharati et al., 2008), and disambiguation accuracy in boosting fluency in text generation (Bloem and Bouma, 2013).

In human cognition research, animacy is considered as one of the first characterizations made by human beings in their infancy and the last distinction lost in adults with Alzheimer's disease (Szewczyk and Schriefers, 2010). This is probably why computational representation of languages (particularly where animacy plays a morpho-syntactic function, e.g. Russian, Arabic, and Hindi) needs to provide adequate description and annotation of this feature. Contrary to previously prevalent perception that animacy is merely a semantic feature, recent years have seen increasing research showing animacy as permeating all levels of linguistic representations: morphology, syntax, semantics, and discourse (Folli and Harley, 2008; Ramchand, 2008; Ritter, 2014; Szewczyk and Schriefers, 2010). Moreover, animacy annotation for some languages is important for both downstream tasks as well as generation.

Animacy is a linguistic property that impacts and relates to a number of other linguistic phenomena, such as case marking, agreement, topicality, argument realization, and structural preferences. It has received formal linguistic investigation in the works of various researchers, most notably (Silverstein, 1976; Dik, 1997; Aissen, 2003; Martin et al., 2005; Dahl, 2008; Bloem and Bouma, 2013; Eckhoff, 2015; Karsdorp et al., 2015).

Various projects have targeted animacy annotation, most remarkably was the benchmark initiative of (Zaenen et al., 2004) where animacy annotation was included in two projects: the Paraphrase project and Possessive Alternation. The GNOME project (Poesio, 2004) aimed to create a corpus to study aspects of discourse and included an animacy taxonomy. The Dutch Cornetto lexical-semantic database (Martin et al., 2005) includes animacy annotation using hierarchical division of categories and subcategories. Orasan and Evans (2007a) described animacy annotation meant for anaphora resolution in English. Thuilier and Danlos (2012) annotated a French corpus for animacy and verb semantic classes. Jena et al. (2013) reported on work to enrich an already available treebank for Hindi with animacy information. Alkuhlani and Habash (2011), Elghamry et al. (2008), and Diab et al. (2014) reported on annotating Arabic data for gender, number and rationality (a hyponym of animacy).

1.1. Animacy Hierarchies

Different hierarchies and annotation schemes have been developed for animacy with different levels of granularity, but the common denominator in these hierarchies is the three level taxonomy first proposed by (Silverstein, 1976):

human > animate > inanimate

More fine-grained details are added to this core hierarchy by classifying 'human' according to person (1st, 2nd, and 3rd person) (Dik, 1997), breaking down 'inanimate' into 'concrete' and 'abstract' (Martin et al., 2005), classifying animates into 'organizations', 'animals', 'intelligent machines' and 'vehicles' (Zaenen et al., 2004) and making a special class for 'inanimate forces' (Dik, 1997) or 'natural forces' (DeLancey, 1981). This classification is assumed to emanate from the speakers' view of objects in the universe where humans are considered as more interesting and valuable than animals, and animals more so than things (Ransom, 1977).

For languages that employ animacy in their morpho-

syntactic paradigm, the exact meaning of animacy is also a source of variation from one language to the other. This is clearly exemplified by the comparison between animacy in Arabic and Russian. Russian makes distinction based on the lower scale of Silverstein's animacy hierarchy (animate vs. inanimate), while Arabic, makes the distinction on the higher scale, rational vs. irrational, which is the equivalent of human vs. non-human. In effect, the binary opposition in the animacy space in the two languages is carved out differently. The paradigms of animacy in Russian and Arabic are illustrated in Figure 1 where animacy in Russian is represented by the solid lines and in Arabic by the dotted ones.

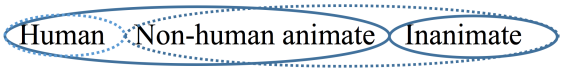


Figure 1: Comparing the notions of Animacy in Russian and Arabic

It is to be noted that animacy as a morpho-syntactic feature does not have to align with biological or ontological animacy. In biology, animacy is used to refer to living things, or anything that has a form of life, including animals and plants. Animals by contrast have the power of locomotion, which separates them from plants. Linguistic animacy is also not equivalent to the notion of 'living things', but is similar to the biological class of 'animal', yet modified by augmenting two additional features: sentience and volition (Jena et al., 2013), that are entities with intrinsic feeling and free will. This allows linguistic animacy to optionally exclude germs, viruses, and lower animates that are seen as collective entities, like cattle and insects. Moreover, what separates humans from non-humans is the feature of 'rationality'. This allows languages to optionally treat entities from outside homo sapiens, like devils and angels, as human.

We also note that animacy as a morpho-syntactic feature, does not have a universal definition, as there are exceptions and a certain element of arbitrariness in the assignment of this feature in different languages. In a survey of animacy in Penobscot, Quinn (2001) points out some unusual examples of nouns treated as animates, including nouns denoting fluid containers: kettle, pot and cup.

1.2. Animacy Types

There is a considerable amount of controversy regarding whether animacy is inside or outside of the "narrow syntax" (Folli and Harley, 2008; Ramchand, 2008; Ritter, 2014; Szewczyk and Schriefers, 2010), or in other terms whether animacy is a semantic/ontological category or a morpho-syntactic feature. We assume that the resolution to this question is to distinguish between the various types of animacy depending on how it is manifested across languages. (Wiltschko and Ritter, 2014) argued for the existence of two types of animacy: morphological animacy and high (semantic) animacy. We take this distinction further by hypothesizing three more types of animacy: anaphoric animacy, syntactic animacy and discourse animacy, adding up to a total of five types of animacy as explained below. These types are not dichotomous, and the existence of animacy at one level, usually has relevance to animacy in the other levels.

Morphological animacy is a property of lexical entries and can additionally be featured in the inflection or derivation of words where lexemes at a certain animacy cutoff limit receive specific inflectional or derivational paradigms. In some languages animacy is a determinant of inflectional paradigms, like plural suffixes in Blackfoot, where animate nouns take the plural suffix -iksi while inanimates take the suffix -stsi (Wiltschko and Ritter, 2014), the plural suffix in the Sistani dialect of Persian (Shariphia et al., 2014) where -hā is used for non-human and ān for human nouns, and Arabic sound (regular) masculine plural suffix -uwna which is used exclusively with rational (human) entities. The derivational paradigm is exemplified by the Arabic collective nouns for lower animates (جراد (jarAd) – *'locusts'*, ذباب (*ubAb) – *'flies'*, نمل (namol) – *'ant'*) and lower inanimates (برتقال (burotuqAl) – *'oranges'*, تفاح (tuf AH) – *'apple'*, and عنب (Einab) – *'grapes'*) where, contrary to other derivational tenancies, the singular is derived from the plural form.

Syntactic animacy can show in the grammar of a languages as strict rules and constraints or merely preferences. As a set of constraints, animacy is featured in the differential case assignment or gender-number agreement which is delineated by animacy, as detailed further in Section 2.. Languages which show inherent or explicit morphological animacy are expected to have syntactic animacy employed in one form or the other. Swart et al. (2008) points out that nouns that lie in the borderline between animate and inanimate may behave syntactically in both ways, reflecting the different aspects of their semantics (Jena et al., 2013). This is clearly exemplified by collective human nouns denoting organizations (such as tribe, team, police) which can be treated syntactically with alternate animacy agreement depending on whether the usage is literal or metonymical. Ambiguous words, like 'monster', and figurative language, like 'star' can also be considered on the borderline of the animacy divides, allowing them to optionally attach to different animacy classes.

As a set of preferences, animacy is assumed to affect the choice of the genitive type in English, where animate nouns tend to attract the *s*-genitive and inanimate nouns the *of*-genitive (Hinrichs and Szmrecsanyi, 2007; Altenberg, 1982).

Anaphoric (or pronominal) animacy is manifested in the animacy-based opposition in the pronoun system (Orasan and Evans, 2007b). English is a good example of this type of animacy. In English the pronouns 'he', 'she', 'it', 'who', and 'which' are demarcated along animacy (and gender) borderlines, and hence animacy is a property of referents, not of the morphology of the lexical items themselves. This feature lies on the border between felicity and grammaticality, depending on whether you see a phrase like 'The book and his reader' as ungrammatical or just unacceptable.

Semantic animacy is responsible for placing selectional restrictions on argument positions (Dahl and Fraurud,

1996). While the first three types of animacy are language-specific, semantic animacy is universal across all languages. It is the encyclopedic knowledge which determines, for instance, that the subject of the verb 'read' must be a human at a certain age. This feature determines sentence felicity, rather than grammaticality.

Discourse animacy is related to the effect of animacy (as an accessibility scale) on topicality and the salience of entities in texts. It is suggested that animacy plays an important role in determining entity prominence and how likely they are to be pronominalized (Poesio, 2004; McGill, 2009; Prat-Sala and Branigan, 2002).

The five types of animacy illustrate just how profoundly animacy permeates human language. This lends support to Dahl (2000)'s claim that animacy is "so pervasive in the grammars of human languages that it tends to be taken for granted and become invisible," and to Øvrelid (2004) 's statement that there is a "strong correlation between the animacy dimension and other linguistic dimensions." Nonetheless, the encoding of animacy in morphology and syntax is directly expressed in much fewer languages than the other features.

Our focus in this paper is on animacy as a purely morpho-syntactic phenomenon (i.e. the first two types above) where animacy applies to a certain, predefined cutoff limit. For example in Arabic and Russian, animacy has an immediate and tangible effect on the morphology and syntax of the two languages.

2. Animacy in Russian and Arabic Grammar

Despite the apparent differences in linguistic perception and dividing lines of animacy in Russian and Arabic (animate vs. inanimate and rational vs. irrational respectively) and despite the differences in the way animacy is featured in the syntax of both languages (differential object marking and differential gender-number agreement), we find that the two languages converge along many routes in the morphological and syntactic representation of animacy.

In morphology, the two languages meet along several demarcating lines when deciding the cutoff limits in animacy hierarchy. For example, groups of people are treated as inanimate in both languages(ru: армия (armiya) – ar: جيش (jayo$) – *'army'*; ru: народ (narod) – ar: شعب ($aEob) – *'nation'*). Groups of animals (ru: стадо (stado) – ar: قطيع (qaTiyE) – *'herd'*) and insects (ru: муравьи (muravyi) – ar: نمل (namol) – *'ants'*) are inanimate in Russian, and they receive inanimate morphology in Arabic by taking collective noun marking).

In syntax, both languages impose structural constraints based on animacy distinctions. In Arabic when a noun is both plural and irrational, it receives agreement equivalent to feminine gender and singular number, as shown by the following example.

الأولاد يلعبون (Al->awoladu yaloEabuwna.) 'The boys play.**pl.masc**'

القطط تلعب (Al-qiTaTu taloEabu.) 'The cats play.**sg.fem**'

In Russian animate nouns have identical inflection marking in genitive and accusative forms, while inanimates have identical inflection marking in nominative and accusative forms, as shown by the following example.

Я увидел мальчика. (Ya uvidel mal'chika.) 'I saw the boy.**gen**[1].'

Я читаю журнал. (Ya chitayu jurnal.) 'I am reading a journal.**nom**.'

3. Data Description

Over the past 25 years, the Linguistic Data Consortium (LDC) has been the main provider of annotated data and annotation standards. However, there are two main concerns with the LDC data (Marcus et al., 1995; Maamouri et al., 2003; Lander, 2005) that poses somes questions about its reliability and relevance to contemporary language. Firstly, most of the collected data is over two decades old now, which makes it less up-to-date. Interestingly, the last two decades have particularly witnessed many socio-political changes, information revolution, and many technological innovations leading to the coinage of many new terms and concepts. Secondly the LDC data is mainly focused on news edited texts, significantly limiting the robustness and scalability of parsing systems. The emergence of the social media and user-generated data have remarkably contributed to the surfacing of new or previously-ignored linguistic phenomena such as code-switching, informal texts and dialects, substandard spelling and grammatical constructions and the use of hashtags, repeated characters for emphasis, electronic addresses and emoticons.

As our annotation is eventually meant for aiding Information Retrieval, we focus on three document genres, namely, news, Wikipedia, and web documents. For the time being we do not include social media posts (e.g. from Facebook, Twitter or Google+) and we do not mine data from other genres, such as literature, religion, tourism, educational materials, technical manuals, spoken dialog, etc.

We annotated freshly collected data with roughly one-third from news articles (covering politics, sports, entertainment, business, health, sci-tech, arts), one-third from Wikipedia articles and one-third from web articles (including blogs, forums, reviews). We also try as much as possible to maintain the balance between sub-domains. This is to ensure that the data we collect is fresh, heterogeneous, application-oriented and representative.

Given the limited bandwidth of annotation time and cost, we also wanted to achieve the largest possible coverage with the smallest possible amount of sentences. Therefore we apply random sampling only for the initial 30% of the data. For the subsequent 70% we apply a 'word frequency-based sampling' approach which favors sentences with lexical items not seen in the initial set.

In this current work we annotate 10,466 sentences (189,029) for Russian and 9,717 sentences (399,774 to-

[1]We show the distinction between animate and inanimate paradigm by marking accusative case endings with the case label of the syncretic inflection. So for animate nouns accusative case is marked with **gen**, and for inanimate nouns the accusative case is marked with **nom**.

kens) for Arabic, with the average sentence length of 18.06 for Russian and 41.14 for Arabic. In the extracted data, we observed that only 0.39% of Russian sentence exceeded the 70 token limit while in Arabic (which exhibits the phenomenon of run-on sentences) we have 27.44% of the data going above that limit. Heuristically we decided to exclude all sentences exceeding 70 tokens.

4. Animacy Annotation Guidelines

Despite of the similarities between Russian and Arabic in the general perspective of animacy and its manifestation on linguistic representation, the two languages still have paradigmatic differences related to the cutoff limit within this feature (animate and inanimate vs. rational and irrational). This is why the annotation guidelines are developed independently to reflect the nature and idiosyncratic needs of each language.

4.1. Animacy in Russian Treebank Annotation

The Russian annotation scheme postulates that the category of animacy refers to the grammatical representation of nouns; it is a grammatical class and does not always align with the biological classification of living and nonliving entities. It affects the accusative case marking in singular forms of masculine and neuter nouns and the accusative case marking in plural forms of all nouns. It is also annotated on adjectival modifiers of these nouns.

Animate nouns of all three genders (feminine, masculine, neuter) have the same marking in the genitive and the accusative cases of the plural form. Likewise, all inanimate nouns regardless of gender have the same marking in the nominative and the accusative cases of the plural form. Unlike feminine nouns, masculine and neuter nouns show the same syncretic pattern across animacy also in the singular paradigm: animate nouns share inflections in the genitive and the accusative case and inanimate nouns share inflections in the nominative and the accusative cases. Animacy for feminine nouns and plural-only nouns are defined based on this pattern:

- Nominative_Singular: лампа “lamp”
- Nominative_Plural: лампы “lamps”
- Genitive_Plural: ламп “lamps”
- Accusative_Plural: лампы “lamps”= Nominative_Plural (inanimate)
- Nominative_Singular: мышь “mouse”
- Nominative_Plural: мыши “mice”
- Genitive_Plural: мышей “mice”
- Accusative_Plural: мышей “mice” = Genitive_Plural (animate)

The majority of neuter nouns is inanimate in Russian. There are several classes of exceptions, such as nouns referring to living creatures with the augmentative suffix -ище (-ische) like котище (kotische) – *‘big cat’*, чудовище (chudovische) – *‘monster’*, страшилище (strashilische) – *‘ogre’*, etc. For non-masculine nouns, we identify the pattern of syncretism in plural paradigms to determine the animacy. If the accusative plural is identical to the genitive plural, we annotate the nouns as animate even in singular forms where this morpho-syntactic distinction is not observed; likewise, if the accusative plural is identical to the nominative plural, we annotate inanimate.

4.2. Animacy in Arabic Treebank Annotation

The general principle of animacy annotation in Arabic is that all nouns (NNs) and proper nouns (NNPs) need to be tagged as either “rat” (rational), “irrat” (irrational) or “unps_a” (unspecified). Rationality mostly correlated with the state of humanness of the entity the noun denotes. The tag unsps_a is reserved for quantifiers (when used nominally), as they do not have intrinsic rationality and can be used variably to refer to rational or irrational objects, e.g. البعض Al-baEoD “some”, الأكثر Al->akovar “most”.

Two of the problem areas when annotating animacy in Arabic are pluralia tantum and metaphors and homonyms, and they are detailed below.

a) Pluralia Tantum

The pluralia tantum are special case plurals which break away from the productive methods of forming plurals from singular nouns. They refer to groups of people, animals or objects, and sometimes they are treated as units which can themselves be assigned plural inflection. They are of two main categories:

Group nouns. Plurals that can receive plural inflection, such as: جماعة jamAEap “group”, فريق fariyq “team” (rationality: irrat).

Fixed plurals. These are plurals not derived from the singular forms, e.g. نساء nisA’ “women”, ناس nAs “people” (rationality: rat).

Mass nouns. These plurals that do not have singular forms, e.g. رمل ramol “sand”, تراب turAb “dust”, ضباب DabAb “cloud” (rationality: irrat).

Collective nouns. In this class, the lemma has a plural meaning, and the singular is derived from the plural by adding a morpheme (taa marboutah) in the end, such as: بقر baqar “cows”, ذباب *ubAb “flies”, تفاح tuf AH “apples” (rationality: irrat).

b) Similes and Metaphors

Similes denote likeness between rational and irrational entities, while metaphors use an irrational entity to denote a rational one. In the former case, the human entity should be tagged as rational and the non-human entity as irrational. In the latter case, metaphors are treated as rational entities, as in the following example.

حضر الحفل العديد من النجوم HaDara Al-Hafola Al-Eadiydu mina Al-nujuwmi NN/rat

‘Many stars NN/rat attended the party.’

5. Inter-Annotator Agreement Results

We computed inter-annotator agreement (IAA) for animacy (among other features) over a random set of 500 sentences. The purpose of this process is to determine if annotators require more training and/or if the guidelines need to be adjusted. The IAA corpus is representative of all document types and is annotated by different annotators in our teams, using the exact same workflow and guidelines. The IAA scores for Russian and Arabic are presented in Table 1.

Note that PoS is part-of-speech tagging accuracy, and LAS is Labeled Attachment Score (Buchholz and Marsi, 2006). The table shows that LAS scores are in the low 90s while PoS scores are in the high 90s. While aimacy IAA scores are above 90% for both Russian and Arabic, Russian scores are 5% higher, which indicate more error analysis is needed for Arabic.

Metric	Russian	Arabic
PoS	98.07	96.70
LAS	91.28	88.97
Per Token morphology accuracy	94.34	94.33
Animacy: F-measure	98.46	93.52
Number of annotators	5	5
Sentences annotated	500	454
Workflow	2-way	2-way

Table 1: IAA scores for Russian and Arabic

	Russian	Arabic
Total Sentences	3,985	3,745
total tokens	79,730	145,725
avg tokens per sentence	20.01	38.91
tokens with anim/rat	41,337	45,378
tokens with anim/rat %	51.85%	31.14%

Table 2: Percentage of tokens with animacy annotations

We also extract interesting statistics from the annotated data (full corpus). Table 2 shows the prevalence of the animacy feature in the annotated data. We also observe that the ratio of tokens annotated with animacy is different between Arabic (31.14%) and Russian (51.85%). This is due to the fact that the Arabic corpus has animacy annotations only for common and proper nouns, whereas the Russian corpus has animacy annotation on common and proper nouns, adjectives, participles, determiners, numerals, and several types of pronouns.

	NN-ru	NN-ar	NNP-ru	NNP-ar
anim/rat	13.82	9.93	44.27	45.55
inanim/irrat	86.18	90.07	55.73	53.59

Table 3: Animacy in NN and NNP

Table 3 demonstrates correlation between animate/rational and inanimate/irrational values in common (NN) and proper (NNP) nouns. Animacy in NNs and NNPs have a very similar distribution across Arabic and Russian where inanimate/irrational values are assigned to the majority of NNs, while we see a more balanced distribution of animacy in NNPs.

In regards to the correlation between gender and animacy, all genders across both languages show (Table 4) a general tendency towards a higher ratio of inanimate/irrational values associated with fem. This is contrasted by a more balanced distribution in masculine nouns: Arabic 20.27% rational vs. 79.73% irrational; Russian 27.25% animate vs. 72.75% inanimate. Almost all Russian nouns with the neuter gender are inanimate.

	Russian %	Arabic %
fem anim/rat	7.57	5.81
fem inan/irrat	92.43	94.19
masc anim/rat	27.25	20.27
masc inan/irrat	72.75	79.73
neut anim	0.29	-
neut inan	99.71	-

Table 4: Gender to animacy correlation

In Table 5 we observe correlations between dependency function and animacy values in Russian and Arabic. For example, there is a general tendency for the animate/rational entities to occupy the subject position, and for the inananimate/irrational to occupy the position of an object of preposition. In Russian, 16.50% of animate nouns occupy the nsbuj position, compared to 7.09% of inanimate nouns. Similarly in Arabic, 18.77% of rational nouns are nsubj, compared to 9.16% of irrational nouns. Genitive modifiers (gmod) in both languages appear in the top three positions regardless of the animacy value.

6. Dependency Grammar

The dependency syntactic representation is a simple description of the grammatical relationships in a sentence. It represents a sentence as a set of binary asymmetrical dependency relations between each pair of tokens or constituents within the sentence. The dependencies merely express dominance relations between a governor (also known the head) and a dependent (also known as the child).

Dependency Grammar is usually viewed in contrast to Constituency Grammar (Chomsky, 1964) which represents sentences as phrase structure trees with a root node branching into non-terminal, terminal, and leaf nodes. It also uses empty nodes and traces for dropped elements. The advantage of Dependency Grammar is that it simplifies the representation by having one node for each token (or syntactic unit) and the arc is labelled with syntactic functions rather than phrase types. It does not employ the concept of empty nodes, making sure that the number of nodes corresponds to the number of tokens.

The origin of Dependency Grammars is usually traced back to the French grammarian Lucien Tesnière who is considered as the father of the theory (Osborne et al., 2015). Dependency Grammar subsequently evolved with contributions from a number of scholars most notably among them are Hays (Hays, 1964), Robinson (Robinson, 1970) and Mel'čuk (Mel'čuk, 1988). Computational implementation within the Dependency Grammars framework has been realized in the creation of dependency treebanks, such as the Prague Dependency Treebank (Hajič et al., 2001), the Stanford Dependencies (De Marneffe and Manning, 2008) and Universal Dependencies (Nivre et al., 2016; McDonald et al., 2013), and the development of dependency parsers,

ru inan %		ru anim %		ar irrat %		ar rat %	
23.23	amod	16.50	nsubj	36.19	pobj	22.52	nn
22.80	pobj	12.75	gmod	27.52	gmod	18.77	nsubj
14.25	gmod	11.83	amod	9.16	nsubj	16.32	gmod
7.09	nsubj	11.13	nn	8.21	conj	13.43	pobj
6.12	conj	9.19	conj	6.89	dobj	12.83	appos
5.10	dobj	8.85	appos	1.91	ROOT	7.03	conj
4.39	appos	8.01	pobj	1.41	appos	3.08	dobj
2.68	ROOT	4.40	ROOT	1.09	nn	1.13	attr
2.17	tmod	3.20	iobj	0.99	nsubjpass	0.89	ROOT

Table 5: Animacy and syntactic dependency

such as the Stanford parser (Chen and Manning, 2014), the inductive dependency parser (Nivre et al., 2004) and the MaltParser (Nivre et al., 2007).
Within Dependency Grammar, the dependency relations can be represented both in relational format and in graph format. In the relational format, the representation is a triple which shows a relation between a pair of words. The head of the dependency relation is given as the first argument and the dependent as the second. We represent this relation as follows:

```
relation(head, dependent)
```

For example, the sentence "he slept" can be represented as:

```
nsubj(slept, he)
```

which means "the nominal subject of 'slept' is 'he"', and the verb is the head of the pronoun.
Similarly, in the graph representation the dependency arc point from the head category to the dependent category, and the relation (or grammatical function) is represented as a label on the arc as shown in Figure 2.

- head-dependent relations (directed arcs)
- functional categories (arc labels)

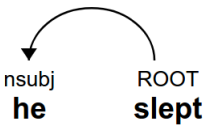


Figure 2: Sample Dependency Graphs

7. Experimental Setup and Evaluation Results

For the prediction of animcy, in addition to the other morphological features, we train a linear-SVM classifier using features based on a window of the current word and word clusters of three words to the left and to the right, suffix for length 1, 2 and 3 for the current word, its left and right words, prefixes of lengths 1, 2 and 3 for the current word and the set of all known morphological attributes for the current word paired with the observed POS tags in the training data.
In our experiments we use an arc-eager transition based dependency parser (Nivre, 2003) with a model trained using a linear SVM architecture similar to the one in Yamada and Matsumoto (2003). When experimenting with morphological features, we add the morphological attributes for both stack top and buffer top tokens.

7.1. Morphological Analysis

Apart from the morphological feature of animacy, our data is annotated for number, gender, case, aspect, mood, person, tense and voice, in addition to the Arabic-specific feature of definiteness and the Russian-specific features of number-antecedent and gender-antecedent.
The prediction results for animacy (as shown in Table 6) is 95.39% for Russian and 92.71% for Arabic which is slightly below the average (95.62% for Russian and 93.12% for Arabic).

	Russian %	**Arabic %**
animacy	95.39	92.71
number	97.60	96.37
gender	95.63	94.16
case	90.82	84.29
Average	95.62	93.12

Table 6: Evaluation results of morphological analysis

Tables 7 and 8 show the confusion matrices for 'animate' and 'inanimate' in Russian and 'rational' and 'irrational' for Arabic.

	Off-Diag	**anim**	**inan**
Off-Diagonal	197	268	589
anim	445	2,642	418
inan	396	226	14,906

Table 7: Confusion matrix for animacy in Russian

Table 9 shows the distribution of the sample space. It shows that in Russian and Arabic 'animate' and 'rational' are the minority classes (they occupy only 16% and 18%, less than one fifth of the sample space), and this correlates with the

	Off-Diag	irrat	rat
Off-Diagonal	1030	1117	417
irrat	1,023	18,101	276
rat	681	425	3,581

Table 8: Confusion matrix for animacy in Arabic

high error rates for these classes (14.42% and 15.98% respectively). This shows that one possibility to improve the results is to treat the imbalance in the data, probably using active learning.

Feature	Sample Total	Sample Ratio	Error Rate
ru/anim	3,087	16.79%	14.42%
ru/inan	15,302	83.21%	2.59%
ar/rat	4,262	18.22%	15.98%
ar/irrat	19,124	81.78%	5.35%

Table 9: Animacy Sample Space and Error Rate

Setup	Ru %	Loss	Ar %	Loss %
All features	81.76	-	80.05	-
No animacy	81.01	0.75	79.95	0.10
No case	79.78	1.98	79.71	0.34
No gender	81.82	-0.06	79.91	0.14
No number	81.66	0.10	80.08	-0.03
No anim/case	79.79	1.97	79.78	0.27
No anim/gend	81.42	0.34	79.91	0.14
No anim/num	81.20	0.56	79.96	0.09
No anim/case num/gend	79.14	2.62	79.74	0.31

Table 10: Evaluation results of dependency parsing

7.2. Dependency Parsing

We conducted a number of experiment using different morphological tags as features for the transition parser to evaluate the effectiveness of animacy in improving the parser, and to compare animacy to other features found in nouns, independently and as a group or pairs.

Table 10 compares the parsing results using all morphological features to the situation where only one, a couple or a group of morphological features are removed. The results show that 'animacy' has a moderately positive impact on Russian and a slightly positive impact on Arabic. Compared to the other morphological features, 'animacy' is the second strongest morphological feature in Russian, while it ranks third among the four morphological features in Arabic.,

Table 10 also shows that morphological tags are helpful as features to the dependency parser, although the impact is much higher in Russian than in Arabic. Moreover, it is to be noted that the predicted tags (not the gold ones) are used as features in the transition parser, and improving the morphological prediction accuracy would likely improve the dependency accuracy.

8. Conclusion

In this paper we have described the motivation for the annotation of animacy in a syntactic dependency treebank. We explained our annotation scheme for Russian and Arabic, two morphologically-rich, remotely related languages which have different views to the animacy scale. We have presented the results and statistics of the annotation of the animacy feature in our dependency treebanks for Russian and Arabic.

The annotated data is used to train a morphological analyzer and the results show a prediction f-measure of 95.39% and 92.71% for animacy in Russian and Arabic respectively. We also show that animacy along with other morphological tags can boost the performance of a dependency parser, and we assume that work on improving the morphological accuracy prediction can lead to further improvement on the parser.

9. Bibliographical References

Aissen, J. (2003). Differential object marking: Iconicity vs. economy. In *Natural Language & Linguistic Theory, 21(3)*, pages 435–483.

Alkuhlani, S. and Habash, N. (2011). A corpus for modeling morpho-syntactic agreement in arabic: Gender, number and rationality. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: short papers*, pages 357–362, Portland, Oregon.

Altenberg, B. (1982). *The Genitive v. the Of-Construction. A Study of Syntactic Variation in 17th Century English*. CWK Gleerup.

Ambati, B. R., Husain, S., Nivre, J., and Sangal, R. (2010). On the role of morphosyntactic features in hindi dependency parsing. In *In Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 94–102. Association for Computational Linguistics.

Bharati, A., Husain, S., Ambati, B., Jain, S., Sharma, D., and Sangal, R. (2008). Two semantic features make all the difference in parsing accuracy. In *Proc. of ICON 8*.

Bloem, J. and Bouma, G. (2013). Automatic animacy classification for dutch. In *Computational Linguistics in the Netherlands Journal 3*, pages 82–10.

Chen, D. and Manning, C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 740–750.

Chomsky, N. (1964). Aspects of the theory of syntax. Technical report, Massachusetts Institute of Technology.

Dahl, Ö. and Fraurud, K. (1996). Animacy in grammar and discourse. *Pragmatics and Beyond New Series*, pages 47–64.

Dahl, Ö. (2000). Animacy and the notion of semantic gender. In *In Gender in grammar and cognition: I: approaches to gender, ed. by Barbara Unterbeck, Matti Rissanen, Terttu Nevalainen and Mirja Saari*, pages 99–115. Berlin: de Gruyter.

Dahl, Ö. (2008). Animacy and egophoricity: Grammar, ontology and phylogeny. *Lingua*, 118:141–150.

De Marneffe, M.-C. and Manning, C. D. (2008). The stanford typed dependencies representation. In *COLING 2008: Proceedings of the Workshop on Cross-framework and Cross-domain Parser Evaluation*, pages 1–8. Association for Computational Linguistics.

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. (2014). Universal stanford dependencies: A cross-linguistic typology. In *Proceedings of LREC-14*.

DeLancey, S. (1981). An interpretation of split ergativity and related patterns. pages 626–657. JSTOR.

Diab, M., Al-Badrashiny, M., Aminian, M., Attia, M., Elfardy, H., Habash, N., and Hawwari, A. (2014). Tharwa: A large scale dialectal arabic - standard arabic - english lexicon. In *The 9th edition of the Language Resources and Evaluation (LREC) Conference*, Reykjavik, Iceland.

Dik, S. C. (1997). The theory of functional grammar. In *Ed. Kees Hengeveld*. Berlin and New York: Mouton de Gruyter.

Eckhoff, H. M. (2015). Animacy and differential object marking in old church slavonic. *Russian Linguist*, 39.

Elghamry, K., Al-Sabbagh, R., and El-Zeiny, N. (2008). Cue-based bootstrapping of arabic semantic features. In *JADT 2008: 9es Journées internationales d'Analyse statistique des Données Textuelles*.

Folli, R. and Harley, H. (2008). Teleology and animacy in external arguments.

Hajič, J., Vidová-Hladká, B., and Pajas, P. (2001). The prague dependency treebank: Annotation structure and support. In *Proceedings of the IRCS Workshop on Linguistic Databases*, pages 105–114.

Hays, D. G. (1964). Dependency theory: A formalism and some observations. *Language*, 40(4):511–525.

Hinrichs, L. and Szmrecsanyi, B. (2007). Recent changes in the function and frequency of standard english genitive constructions: A multivariate analysis of tagged corpora. *English Language & Linguistics*, 11(3):437–474.

Jena, I., Bhat, R. A., Jain, S., and Sharma, D. M. (2013). Animacy annotation in the hindi treebank. In *Proceedings of the 7th Linguistic Annotation Workshop & Interoperability with Discourse*, pages 159–167, Sofia, Bulgaria.

Karsdorp, F., Meulen, van der; Meder, M., Bosch, T., and van den, A. (2015). Animacy detection in stories. In *In Finlayson, M., Miller, B., Lieto, A. (ed.), Proceedings of the 6th Workshop on Computational Models of Narrative (CMN-2015)*, pages 82–97.

Lamers, M. J. (2007). Verb type, animacy and definiteness in grammatical function disambiguation. *Linguistics in the Netherlands*, 24:125–137.

Lander, T. (2005). Cslu: 22 languages corpus. In *LDC2005S26*, Philadelphia. Linguistic Data Consortium.

Liang, T. and Wu, D.-S. (2004). Automatic pronominal anaphora resolution in english texts. In *Computational Linguistics and Chinese Language*, pages 21–40.

Maamouri, M., Bies, A., Jin, H., and Buckwalter, T. (2003). Part 1 v 2.0. In *LDC2003T06*, Philadelphia. Linguistic Data Consortium.

Marcus, M., Santorini, B., and Marcinkiewicz, M. A. (1995). Treebank-2. In *LDC95T7*, Philadelphia. Linguistic Data Consortium.

Martin, W., Maks, I., Bopp, S., and Groot, M. (2005). Rbn-documentatie. In *Report, TST Centrale*.

Marton, Y., Habash, N., and Rambow, O. (2011). Improving arabic dependency parsing with form-based and functional morphological features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 1586–1596, Portland, Oregon.

McDonald, R. T., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K. B., Petrov, S., Zhang, H., Täckström, O., et al. (2013). Universal dependency annotation for multilingual parsing. In *The Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 92–97, Sofia, Bulgaria.

McGill, S. (2009). Gender and person agreement in cicipu discourse. In *PhD thesis*, School of Oriental and African Studies.

Mel'čuk, I. A. (1988). *Dependency syntax: theory and practice*. SUNY press.

Nivre, J., Hall, J., and Nilsson, J. (2004). Memory-based dependency parsing. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Nivre, J., Boguslavsky, I. M., and Iomdin, L. L. (2008). Parsing the syntagrus treebank of russian. In *Proceedings of the 22nd International Conference on Computational Linguistics*. Association for Computational Linguistics.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal dependencies v1: A multilingual treebank collection. In *International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666, Portorož, Slovenia.

Nivre, J. (2003). An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT)*.

Orasan, C. and Evans, R. J. (2007a). Np animacy identification for anaphora resolution. In *Journal for Artificial Intelligence Research*, volume 29, pages 79–103.

Orasan, C. and Evans, R. J. (2007b). Np animacy identification for anaphora resolution. *Journal of Artificial Intelligence Research*, 29:79–103.

Osborne, T., Maxwell, D., and Center, U. C. (2015). A historical overview of the status of function words in dependency grammar. In *DepLing*, pages 241–250.

Øvrelid, L. (2004). Disambiguation of syntactic functions in norwegian: modeling variation in word order interpretations conditioned by animacy and definiteness. In

Proceedings of the 20th Scandinavian Conference of Linguistics, Helsinki: University of Helsinki.

Poesio, M. (2004). Discourse annotation and semantic annotation in the gnome corpus. In *The ACL 2004 Workshop on Discourse Annotation*, pages 72–79, Barcelona, Spain.

Prat-Sala, M. and Branigan, H. P. (2002). Discourse constraints on syntactic processing in language production: A cross-linguistic study in english and spanish. *Journal of Memory and Language*, 42:168–182.

Quinn, C. (2001). A preliminary survey of animacy categories in penobscot. In *Papers of the 32nd. Algonquian Conference*, pages 395–426.

Ramchand, G. (2008). In *Verb Meaning and the Lexicon: A First Phase Syntax*. Cambridge University Press.

Ransom, E. N. (1977). Definiteness, animacy, and np ordering. In *Proceedings of the 3rd Annual Meeting of the Berkeley Linguistics Society*, pages 418–429.

Ritter, E. (2014). Featuring animacy. In *Nordlyd 41.1:, special issue on Features edited by Martin Krämer, Sandra-Iulia. Ronai and Peter Svenonius. University of Tromsø*, pages 103–124. The Arctic University of Norway.

Robinson, J. J. (1970). Dependency structures and transformational rules. *Language*, pages 259–285.

Shariphia, S., Mirib, M., and Heidarypur, M. (2014). Investigation of relation of animacy and number in sistani dialect. *International Journal of Education and Research*, 2.

Silverstein, M. (1976). Hierarchy of features and ergativity. In *Dixon, R.M.W. (Ed.), Grammatical Categories in Australian Languages*, pages 112–171, Australian Institute of Aboriginal Studies, Canberra.

Singh, S., Lakhmani, P., Mathur, P., and Morwal, S. (2014). Analysis of anaphora resolution system for english language. *International Journal on Information Theory (IJIT)*, 3.

Swart, P. D., Lamers, M., and Lestrade, S. (2008). Animacy, argument structure, and argument encoding. *Lingua*, 118:131–140.

Szewczyk, J. M. and Schriefers, H. (2010). Is animacy special? erp correlates of semantic violations and animacy violations in sentence processing. *Brain Research*, 1368:208–221.

Thuilier, J. and Danlos, L. (2012). Semantic annotation of french corpora: animacy and verb semantic classes. In *LREC 2012*, pages 1533–1537. European Language Resources Association (ELRA).

Wiltschko, M. and Ritter, E. (2014). Animating the narrow syntax. In *Talk given at GLOW 37*.

Yamada, H. and Matsumoto, Y. (2003). Statistical dependency analysis with support vector machines. In *Proceedings of the 8th International Workshop on Parsing Technologies*, pages 195–206, Nancy, France.

Zaenen, A., Carletta, J., Garretson, G., Bresnan, J., Koontz-Garboden, A., Nikitina, T., Catherine O'Connor, M., and Wasow, T. (2004). Animacy encoding in english: Why and how. In *Proceedings of the Workshop on Discourse Annotation*.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinková, S., jr, J. H., Hlavácová, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., Droganova, K., Alonso, H. M., Çağrı Çöltekin, Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada.

MADARi: A Web Interface for Joint Arabic Morphological Annotation and Spelling Correction

Ossama Obeid, Salam Khalifa, Nizar Habash,
Houda Bouamor,† Wajdi Zaghouani,* Kemal Oflazer†
Computational Approaches to Modeling Language Lab, New York University Abu Dhabi, UAE
† Carnegie Mellon University in Qatar, Qatar
* Hamad Bin Khalifa University, Qatar
{oobeid, salamkhalifa, nizar.habash}@nyu.edu,
hbouamor@qatar.cmu.edu, wzaghouani@hbku.edu.qa, ko@cs.cmu.edu

Abstract

In this paper, we introduce MADARi, a joint morphological annotation and spelling correction system for texts in Standard and Dialectal Arabic. The MADARi framework provides intuitive interfaces for annotating text and managing the annotation process of a large number of sizable documents. Morphological annotation includes indicating, for a word, in context, its baseword, clitics, part-of-speech, lemma, gloss, and dialect identification. MADARi has a suite of utilities to help with annotator productivity. For example, annotators are provided with pre-computed analyses to assist them in their task and reduce the amount of work needed to complete it. MADARi also allows annotators to query a morphological analyzer for a list of possible analyses in multiple dialects or look up previously submitted analyses. The MADARi management interface enables a lead annotator to easily manage and organize the whole annotation process remotely and concurrently. We describe the motivation, design and implementation of this interface; and we present details from a user study working with this system.

Keywords: Arabic, Morphology, Spelling Correction, Annotation

1. Introduction

Annotated corpora are vital for research in natural language processing (NLP). These resources provide the necessary training and evaluation data to build automatic annotation systems, and benchmark them. The task of human manual annotation, however, is rather difficult and tedious and several annotation interface tools have been created to assist in such effort. These tools tend to be specialized to optimize for specific tasks such as spelling correction, part-of-speech (POS) tagging, named-entity tagging, syntactic annotation, etc. Certain languages bring additional challenges to the annotation task. Compared with English, Arabic annotation introduces a need for diacritization of the diacritic-optional orthography, frequent clitic segmentation, and a richer POS tagset.

In this paper, we focus on designing and implementing a tool targeting Arabic dialect morphological annotation. Standard Arabic morphology is quite rich (Habash, 2010), but Arabic dialects introduce more complexity than Standard Arabic in that the input text has noisy orthography. For example, the word ويابوهاالخليج *wyAbwhAAlxlyj*[1] involves two spelling errors[2] (a word merge and character replacement) which can be corrected as وجابوها الخليج *wjAbwhA Alxlyj* 'and they brought it to the Gulf'. Furthermore, the first of the two corrected words includes two clitics that when segmented produce the form: و+ جابوا +ها *w+ jAbwA +hA* 'and+ they-brought +it'.

[1]Transliterations are in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007).

[2]Since Arabic dialects do not have a standard orthography, *spelling correction* here means to conventionalize as per the CODA standard (Habash et al., 2018).

Previous work on Arabic morphology annotation interfaces focused either on the problem of manual annotations for POS tagging (Maamouri et al., 2014), or diacritization (Obeid et al., 2016), or spelling correction (Obeid et al., 2013). In this paper we present a tool that allows doing all of these tasks together, eliminating the possibility of error propagation from one annotation level to another. Our tool is named MADARi[3] after the project under which it was created: Multi-Arabic Dialect Annotations and Resources (Bouamor et al., 2018).

The remainder of this paper is structured as follows: we present work related to this effort in Section 2. In Section 3., we discuss the design and architecture of our annotation framework. In Section 4. and Section 5., we discuss the annotation and management interfaces, respectively. We finally describe a user study of working with MADARi in Section 6.

2. Related Work

Several annotation tools and interfaces were proposed for many languages and to achieve various annotation tasks. Some are general purpose annotation tools, such as BRAT (Stenetorp et al., 2012) and WebAnno (Yimam et al., 2013). Task-specific annotation tools for post-editing and error correction include the work of Aziz et al. (2012), Stymne (2011), Llitjós and Carbonell (2004), and Dickinson and Ledbetter (2012).

For Arabic, there are several existing annotation tools, however, they are designed to handle specific NLP tasks and are not easy to adapt to our project. Examples include tools for semantic annotation such as the work of Saleh and Al-Khalifa (2009) and El-ghobashy et al. (2014),

[3]مداري *madAriy* means 'my orbit' in Arabic.

and the work on dialect annotation by Benajiba and Diab (2010) and Diab et al. (2010). Attia et al. (2009) built a morphological annotation tool. Recently, Al-Twairesh et al. (2016) introduced MADAD, a general-purpose online collaborative annotation tool for readability assessments project in Arabic. In the COLABA initiative (Diab et al., 2010), the authors built tools and resources to process Arabic social media data such as blogs, discussion forums, and chats. Javed et al. (2018) presented an online interface for joint syntactic annotation and morphological tokenization for Arabic.

In general, many of these existing tools are not designed to handle the peculiarities of dialectal Arabic. They neither provide facilities for managing thousands of documents nor permit the distribution of tasks to tens of annotators, including managing inter-annotator agreement (IAA) tasks. Our interface borrows ideas from three other existing annotation tools: DIWAN, QAWI, and MANDIAC. Here we describe each of these tools and how they have influenced the design of our system.

DIWAN is an annotation tool for Arabic dialectal texts (Al-Shargi and Rambow, 2015). It provides annotators with a set of tools for reducing duplicate effort including the use of morphological analyzers to pre-compute analyses, and the ability to apply analyses to multiple occurrences simultaneously. However, it requires installation on a Windows machine and the user interface is not very friendly to newcomers.

QAWI (the QALB Annotation Web Interface) was used for token-based text editing to create raw and and text corrected parallel data for automatic text correction tasks (Obeid et al., 2013; Zaghouani et al., 2014; Zaghouani et al., 2015; Zaghouani et al., 2016). It supported the exact recording of all modifications performed by the annotator which previous tools did not. We utilize this token-based editing system for minor text corrections that transform text of a given dialect into the appropriate CODA orthography (Habash et al., 2018).

MANDIAC utilized the token-based editor used in QAWI to perform text diacritization tasks (Obeid et al., 2016). More importantly, it introduced a flexible hybrid data storage system that allows for adding new features to the annotation front-end with little to no modifications to the back-end. MADARi utilizes this design to provide the same utility.

3. MADARi Design

The MADARi interface is designed to be used by human annotators to create a morphologically annotated corpus of Arabic text. The text we work with comes from social media and is highly dialectal (Bouamor et al., 2018; Khalifa et al., 2018) and has numerous spelling errors. The annotators will carefully correct the spelling of the words and also annotate their morphology. The in-context morphology annotation includes tokenization, POS tagging, lemmatization and English glossing.

3.1. Desiderata

In order to manage and process the annotation of the large scale dialectal Arabic corpus, we needed to create a tool to streamline the annotation process. The desiderata for developing the MADARi annotation tool include the following:

- The tool must have very minimal requirements on the annotators.
- The tool must allow off-site data management of documents to allow annotation leaders to assign and grade documents from anywhere in the world and to allow hiring annotators anywhere in the world.
- The tool must allow easily customizable POS tag sets by annotation leads.
- The tool must allow easy access to other user annotations of similar texts.
- The tool must allow for easy navigation between spelling changes and morphological disambiguation.

3.2. Design and Architecture

The design of our interface borrows heavily from the design of MANDIAC (Obeid et al., 2016). In particular, we utilized the client-server architecture, as well as the flexible hybrid SQL/JSON storage system used by MANDIAC. This allows us to easily extend our annotation interface with minor changes, if any, to the back-end. Our system stores documents one sentence per row, unlike MANDIAC which stores one document per row. This modification allows the annotation interface to handle larger file sizes without affecting its performance by only overwriting the JSON of the modified sentences and not that of the entire document. Like, DIWAN and MANDIAC, we also utilize MADAMIRA (Pasha et al., 2014), a morphological analyzer and disambiguator for Arabic to pre-compute analyses.

4. Annotation Interface

The annotation interface (illustrated in Figures 1 to 4) is where annotators perform the annotation tasks assigned to them. Here we describe the different components and utilities provided this interface.

4.1. Task Overview

When starting an annotation session, annotators are first shown the "Task Overview" screen (Figure 1). Here annotators can see information on the size of the current task and their progress so far (Figure 1a). The sentence list can be filtered to contain sentences matching a desired search term using the filter bar (Figure 1b). The list of sentences in the current task is also displayed with validated tokens color-coded green (Figure 1c). Clicking on any word in the sentence list will open the annotation interface (Figure 2) at that word.

4.2. Word Analysis

The essential component of our interface is the morphological analysis screen (Figure 2). The original text is provided for reference at the top of the panel (Figure 2a). Figure 2b displays the updated form of the words, and allows selecting a word to annotate. The currently selected word is colored blue; and validated words are colored green.

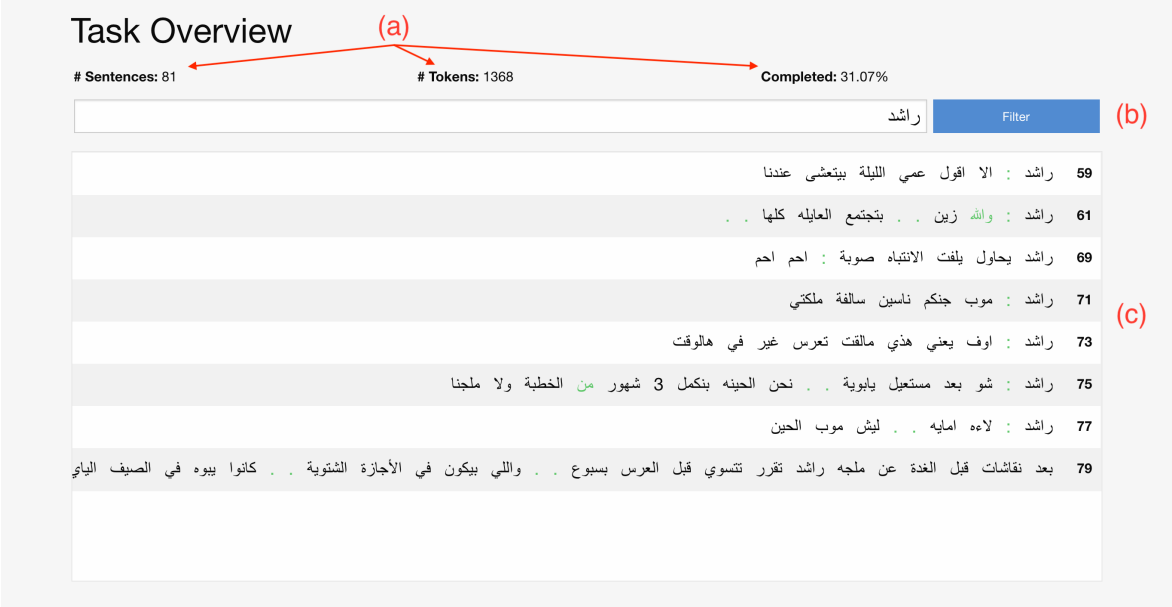


Figure 1: MADARi Task Overview screen

Figure 2c is the heart of the annotation interface, where annotators manually disambiguate the morphological analysis. Disambiguation includes morphologically tokenizing each word into proclitics, baseword, and enclitics (Figure 2c, first row from right to left, respectively). Each of these are assigned a POS tag as well as a morphological feature[4] where applicable (Figure 2c, second row). Annotators also assign the lemma, gloss, and dialect for each word (Figure 2c, third row, second to fourth fields from the left respectively). For the convenience of the annotators, we provide pre-computed values for each field using MADAMIRA's morphological analysis. Each word has a validated field (Figure 2c, third row, right-most field) to indicate that the annotator has fully analyzed it and is confident with their analysis.

Generally, the final form of a word is a concatenation of the proclitics, baseword, and enclitics. However, there are certain cases where that is not true because some orthographic rewrite rules must apply (Habash et al., 2018). Using the example in the introduction, وجابوها *wjAbwhA* should be tokenized by annotators into و+ جابوا +ها *w+ jAbwA +hA*. However, when displaying the detokenized token, the system should show وجابوها *wjAbwhA* and not وجابواها *wjAbwAhA*. MADARi has built-in rewrite rules for trivial detokenization cases but we also allow annotators to manually edit the detokenized form manually as needed (Figure 2c, third row, left-most field).

4.3. Text Editing

Annotators can freely alternate between morphological analysis and spelling modification of the words in the sentence. This gives them the freedom to make joint decisions on spelling and morphology and avoid error propagation. Sentence edits can be made by going to the "Edit Sentence" view (Figure 3). In the "Edit Sentence" view, only the word tokens of the sentence are shown, each with a left and right arrow button surrounding them (Figure 3a). Clicking on one of these arrows merges that token with the one on the left or right respectively. Double clicking on a token displays the "Edit Token" pop-up (Figure 3a). In this pop-up, an annotator can edit a word or split it into multiple tokens by inserting spaces between the letters.

4.4. Utilities

We have added a number of utility features to make the annotation process easier and more efficient for annotators. Basic utilities include undo and redo buttons (Figure 2h), switching between English and Arabic POS tags (Figure 2f). Annotators can jump to the next or previous sentence, go to the "Task Overview" screen, or exit the task in the navigation bar (Figure 2e). All functions in the navigation bar automatically save any changes made by the annotator. Furthermore, annotators can see what document and sentence they are currently annotating as well as the whether there are any unsaved changes in the task status bar (Figure 2g).

We also allow annotators to update multiple instances of a word with the same orthography together. In the "Contexts" panel, annotators are shown a list of all occurrences of a word within the current document in context (Figure 4a). They can then select each context they would like to update by clicking on the check box on the left of each instance. Finally, annotators click on the "Apply to Selected" button (Figure 4a) to apply the analysis of the current word to all the selected instances.

Additionally, we provide annotators with a search utility to look up previously submitted analyses as well as query MADAMIRA for out-of-context analyses in different dialects and apply a chosen analysis in real-time using the "Analysis Search" panel (Figure 2d, Figure 4b). Annotators type in a word to query in the search bar. Clicking

[4]We use the CAMEL POS tag set and features defined by Khalifa et al. (2018).

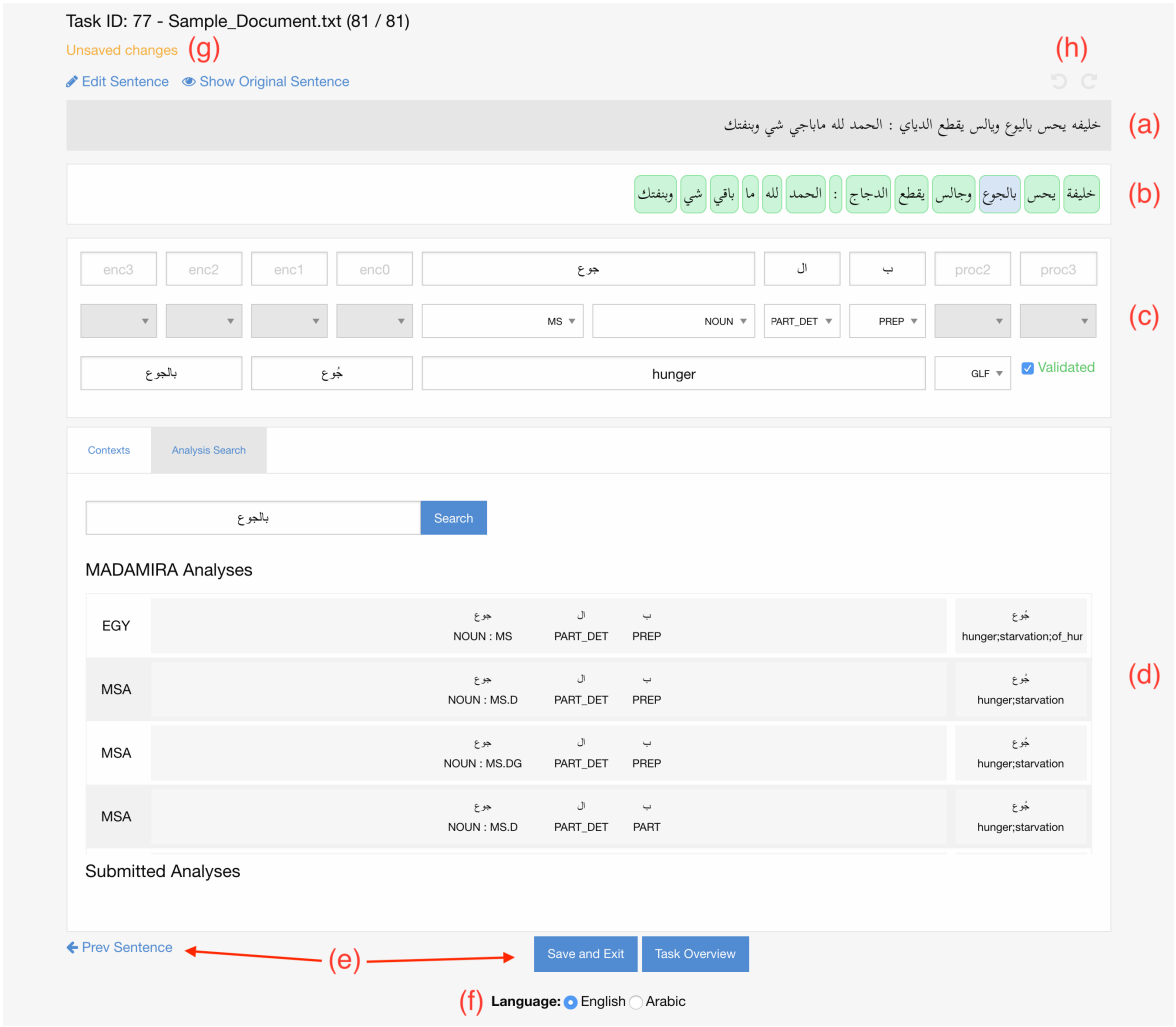
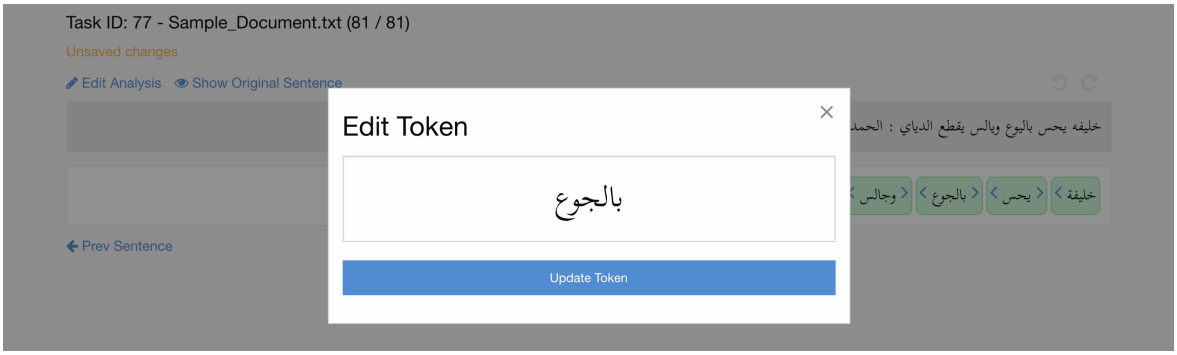


Figure 2: Full view of the MADARi annotation interface

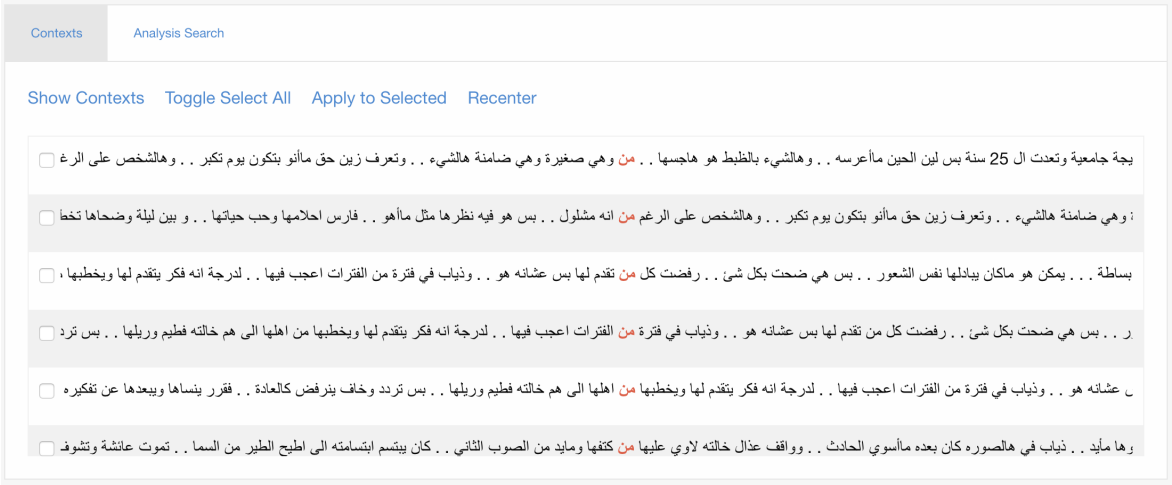


(a) Token merge and split view

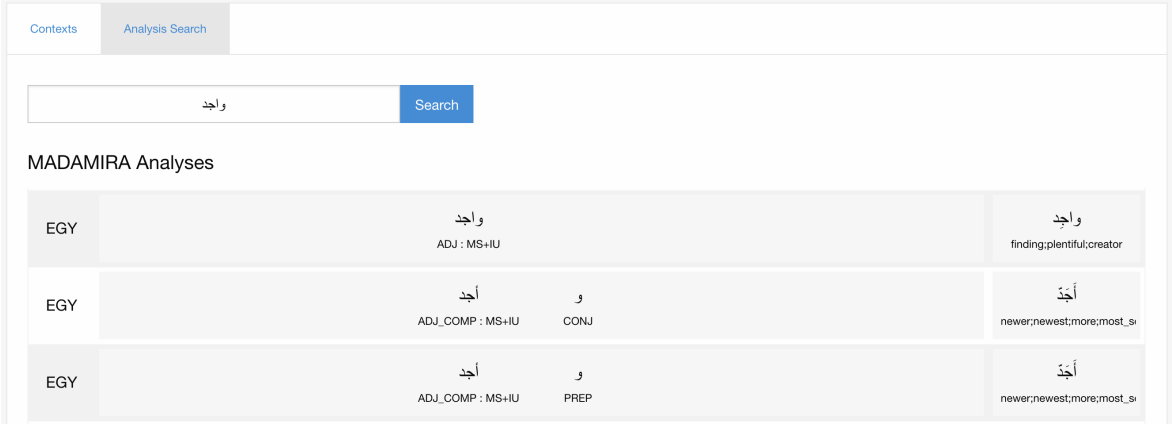


(b) Token edit pop-up

Figure 3: Edit sentence view



(a) Contexts panel view



(b) Analysis search panel.

Figure 4: Contexts and analysis search view

"Search" retrieves a list of out-of-context analyses from MADAMIRA and a list of previously submitted analyses of the search term. Double clicking on a listed analysis applies it to the current word if the current word can be tokenized in such a way that its clitics match those of the selected analysis. For example بالليل *bAllyl* 'during the night' and بالنهار *bAlnhAr* 'during the day' have the same proclitics ب+ال *b+Al+* and no enclitics, thus they have matching clitics.

5. Management Interface

The Annotation Management Interface enables the lead annotator to easily manage and organize the whole annotation process remotely and concurrently. The management interface contains: (a) a user management tool for creating new annotator accounts and viewing annotator progress; (b) a document management tool for uploading new documents, assigning them for annotation, and viewing submitted annotations; (c) a monitoring tool for viewing overall annotation progress; (d) a data repository and annotation export feature; and (e) a utility for importing pre-annotated documents, overriding MADAMIRA's analyses.

6. User Study

Our tool is being used as part of an ongoing annotation project on Gulf Arabic (Khalifa et al., 2018). In this paper, we describe the experience of one annotator who has done annotations in different settings previously. The annotator morphologically disambiguated 80 sentences totaling in 1,355 raw tokens of Gulf Arabic text.

The annotator preferred, based on her experience, to convert the orthography of the text to CODA first, which made the disambiguation task more efficient.

It took about 52 minutes to complete this task (corresponding to a rate of 1,563 words/hour). The annotator made a few minor fixes later on, which is an advantage of our tool to minimize error propagation. The total number of words that were changed from the raw tokens to CODA was 288 (21%). Changes were mostly spelling adjustments and the rest is word splitting (44 cases or 15% of all changes) but no merges. The final word count is 1,398 words.

Following the CODA conversion, the annotator worked on tokenization, POS tagging, lemmatization and English glossing. This more complex task took around 6 hours (at a rate of 277 words/hour). This makes the cumulative time

spent to finish the spelling adjustment and the full disambiguation tasks for this set of data about 7 hours (at a rate of 200 words/hour).

Since the tool provides initial guesses for all the annotation components, the annotator was able to use many of the valid decisions as is, and modify them in other cases. In the event of a word split, the tool currently removes the raw word predictions, but the analysis search utility allows fast access to alternatives to select from.

We compared the final tokenization, POS tag and lemma choices to the ones suggested by the tool on the CODA version of the text. We found that the tool gave correct suggestions 74% of the time on tokenization, 69% of the time on baseword POS tags and 70% of the time on lemmas.

The annotator indicated that their favorite utilities were the ability to annotate multiple tokens of the same type in different contexts simultaneously, and the ability to use the *Analysis Search* box to annotate multiple fields simultaneously.

7. Conclusion and Future Work

We presented an overview of our web-based annotation framework for joint morphological annotation and spelling correction of Arabic. We plan to release the tool and make it freely available to the research community so it can be used in other related annotation tasks. In the future, we will continue extending the tool to support different dialects and genres of Arabic.

Acknowledgments

This publication was made possible by grant NPRP7-290-1-047 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

Bibliographical References

Al-Shargi, F. and Rambow, O. (2015). DIWAN: A Dialectal Word Annotation Tool for Arabic. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, page 49.

Al-Twairesh, N., Al-Dayel, A., Al-Khalifa, H., Al-Yahya, M., Alageel, S., Abanmy, N., and Al-Shenaifi, N. (2016). Madad: A readability annotation tool for arabic text. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Attia, M., Rashwan, M. A., and Al-Badrashiny, M. A. (2009). Fassieh, a Semi-automatic Visual Interactive Tool for Morphological, PoS-Tags, Phonetic, and Semantic Annotation of Arabic Text Corpora. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(5):916–925.

Aziz, W., de Sousa, S. C. M., and Specia, L. (2012). PET: a tool for post-editing and assessing machine translation. In *Proceedings of the LREC'2012*.

Benajiba, Y. and Diab, M. (2010). A Web Application for Dialectal Arabic Text Annotation. In *Proceedings of the LREC Workshop for Language Resources (LRS) and Human Language Technologies (HLT) for Semitic Languages: Status, Updates, and Prospects*.

Bouamor, H., Habash, N., Salameh, M., Zaghouani, W., Rambow, O., Abdulrahim, D., Obeid, O., Khalifa, S., Eryani, F., Erdmann, A., and Oflazer, K. (2018). The MADAR Arabic Dialect Corpus and Lexicon. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *LREC Workshop on Semitic Language Processing*.

Dickinson, M. and Ledbetter, S. (2012). Annotating Errors in a Hungarian Learner Corpus. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2012)*.

El-ghobashy, A. N., Attiya, G. M., and Kelash, H. M. (2014). A Proposed Framework for Arabic Semantic Annotation Tool. 3(1).

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic transliteration. In Abdelhadi Soudi, et al., editors, *Arabic Computational Morphology*, volume 38 of *Text, Speech and Language Technology*, chapter 2, pages 15–22. Springer.

Habash, N., Khalifa, S., Eryani, F., Rambow, O., Abdulrahim, D., Erdmann, A., Faraj, R., Zaghouani, W., Bouamor, H., Zalmout, N., Hassan, S., shargi, F. A., Alkhereyf, S., Abdulkareem, B., Eskander, R., Salameh, M., and Saddiki, H. (2018). Unified Guidelines and Resources for Arabic Dialect Orthography. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Javed, T., Habash, N., and Taji, D. (2018). Palmyra: A platform independent dependency annotation tool for morphologically rich languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, may.

Khalifa, S., Habash, N., Eryani, F., Obeid, O., Abdulrahim, D., and Kaabi, M. A. (2018). A Morphologically Annotated Corpus of Emirati Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Llitjós, A. F. and Carbonell, J. G. (2004). The Translation Correction Tool: English-Spanish User Studies. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*.

Maamouri, M., Bies, A., Kulick, S., Ciul, M., Habash, N., and Eskander, R. (2014). Developing an Egyptian Arabic Treebank: Impact of Dialectal Morphology on Annotation and Tool Development. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Obeid, O., Zaghouani, W., Mohit, B., Habash, N., Oflazer,

K., and Tomeh, N. (2013). A Web-based Annotation Framework For Large-Scale Text Correction. In *The Companion Volume of the Proceedings of IJCNLP 2013: System Demonstrations*, pages 1–4, Nagoya, Japan.

Obeid, O., Bouamor, H., Zaghouani, W., Ghoneim, M., Hawwari, A., Alqahtani, S., Diab, M., and Oflazer, K. (2016). MANDIAC: A Web-based Annotation System For Manual Arabic Diacritization. In *The 2nd Workshop on Arabic Corpora and Processing Tools 2016 Theme: Social Media*, page 16.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Saleh, L. M. B. and Al-Khalifa, H. S. (2009). AraTation: an Arabic Semantic Annotation Tool. In *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services*, pages 447–451. ACM.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: A Web-based Tool for NLP-assisted Text Annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, Stroudsburg, PA, USA.

Stymne, S. (2011). Blast: a Tool for Error Analysis of Machine Translation Output. In *Proceedings of the ACL'2011: Systems Demonstrations*, pages 56–61.

Yimam, S. M., Gurevych, I., de Castilho, R. E., and Biemann, C. (2013). WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *ACL (Conference System Demonstrations)*, pages 1–6. The Association for Computer Linguistics.

Zaghouani, W., Mohit, B., Habash, N., Obeid, O., Tomeh, N., Rozovskaya, A., Farra, N., Alkuhlani, S., and Oflazer, K. (2014). Large Scale Arabic Error Annotation: Guidelines and Framework. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland.

Zaghouani, W., Habash, N., Bouamor, H., Rozovskaya, A., Mohit, B., Heider, A., and Oflazer, K. (2015). Correction annotation for non-native Arabic texts: Guidelines and corpus. *Proceedings of The 9th Linguistic Annotation Workshop*, pages 129–139.

Zaghouani, W., Habash, N., Obeid, O., Mohit, B., and Oflazer, K. (2016). Building an Arabic Machine Translation Post-Edited Corpus: Guidelines and Annotation. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

A Morphological Analyzer for St. Lawrence Island / Central Siberian Yupik

Emily Chen, Lane Schwartz
Department of Linguistics
University of Illinois Urbana-Champaign, Champaign, Illinois
{echen41, lanes}@illinois.edu

Abstract

St. Lawrence Island / Central Siberian Yupik is an endangered language, indigenous to St. Lawrence Island in Alaska and the Chukotka Peninsula of Russia, that exhibits pervasive agglutinative and polysynthetic properties. This paper discusses an implementation of a finite-state morphological analyzer for Yupik that was developed in accordance with the grammatical standards and phenomena documented in Jacobson's 2001 reference grammar for Yupik. The analyzer was written in *foma*, an open source framework for constructing finite-state grammars of morphology. The approach presented here cyclically interweaves morphology and phonology to account for the language's intricate morphophonological system, an approach that may be applicable to typologically similar languages. The morphological analyzer has been designed to serve as foundational resource that will eventually underpin a suite of computational tools for Yupik to assist in the process of linguistic documentation and revitalization.

Keywords: computational morphology, finite-state grammar, morphological analyzer

1. Introduction

We introduce in this paper an implementation of a morphological analyzer for St. Lawrence Island / Central Siberian Yupik (ISO 639-3: ess), an agglutinative, polysynthetic language of the Inuit-Yupik language family.[1] This analyzer is implemented in *foma* (Hulden, 2009b), and represents a faithful adaptation of the grammatical and morphophonological rules documented in *A Practical Grammar of the St. Lawrence Island / Siberian Yupik Eskimo Language* (Jacobson, 2001). We intend for the morphological analyzer to be a foundational resource that underpins a suite of computational tools to be shared with the Yupik community for purposes of language preservation and revitalization.

2. Language Description

Yupik is the westernmost variety of the Inuit-Yupik language family (Krauss et al., 2010). Most of the 2400–2500 Yupik people reside in two villages on St. Lawrence Island, Alaska and two villages on the Chukotka Peninsula of far eastern Russia (Krupnik and Chlenov, 2013). Out of that population, fewer than 1000 are estimated to be L1 Yupik speakers (Koonooka, 2005; Schwalbe, 2017).

The Yupik phonemic inventory comprises 31 consonants and 4 vowels, /ə/, /i/, /a/, /u/ which can be lengthened (with the exception of /ə/) for a total of seven vocalic phonemes. Of the 31 consonants, there are 8 pairs of continuants and 4 pairs of nasals where each pair differs only in voicing. In the standard Latin-based orthography, this difference is marked in 5 of the 8 continuant pairs and in all of the nasal pairs through *graphemic doubling*, that is, *l* and *ll*, *r* and *rr*, *g* and *gg*, *gh* and *ghh*, *ghw* and *ghhw*, *m* and *mm*, *n* and *nn*, *ng* and *ngng*, *ngw* and *ngngw*. In each pair, the doubled grapheme represents the voiceless phoneme.

2.1. Morphology

Like all languages in the Inuit-Yupik family, the morphology of Yupik is remarkably generative. Yupik nouns and verbs are principally responsible for the most morphologically complex words in the language, and permit up to seven derivational morphemes or *postbases* as they are referred to in the literature (de Reuse, 1994, p.53). Noun and verb roots are considered *bases*, although the term *base* is used more generally to apply to any uninflected form that is available for further affixation (de Reuse, 1994, p.24). That is to say, a *base* + *postbase* unit may also be referred to as a *base*. There is only one attested prefix in Yupik, which can be applied only to demonstratives; all other affixation is suffixing (Jacobson, 2001, p.109). Thus, the underlying structure of most Yupik words is *(noun/verb) base + zero or more derivational postbases + inflectional postbase* (*+ optional enclitic*), where enclitics are associated with nouns only and simply affix in word-final position.

As far as previous literature has shown, derivational postbases in Yupik can only be one of four types (Jacobson, 2001), and may or may not constitute a closed class:

Nouns subsequently inflect for person, number, and possession based on the case of the noun base, while verbs inflect for person and number based on the mood of the verb base. Person and number are expressed within a single morpheme, suggesting that Yupik also possesses some fusional properties. Other root bases that can be inflected include demonstratives, numerals, and personal pronouns, while Yupik "adjectives" manifest as verb bases, such as **kavite-** (*to be red*).

2.2. Morphophonology

While some Yupik postbases directly affix to bases, such as the verb-elaborating postbase **+sug** (*to want one to V*) which yields **kavitesug-** (*to want one to be red*), most postbases trigger a series of morphophonological changes at the immediate left-adjacent base-postbase boundary. In Jacobson (2001), each morphophonological process is systematically documented and assigned a unique symbol, such as + which designates straightforward affixation of postbase to base. While the morphonological processes called upon by each postbase are not predictable, the symbols repre-

[1] The term *Yupik* will be used henceforth to refer to the St. Lawrence Island / Central Siberian Yupik variety.

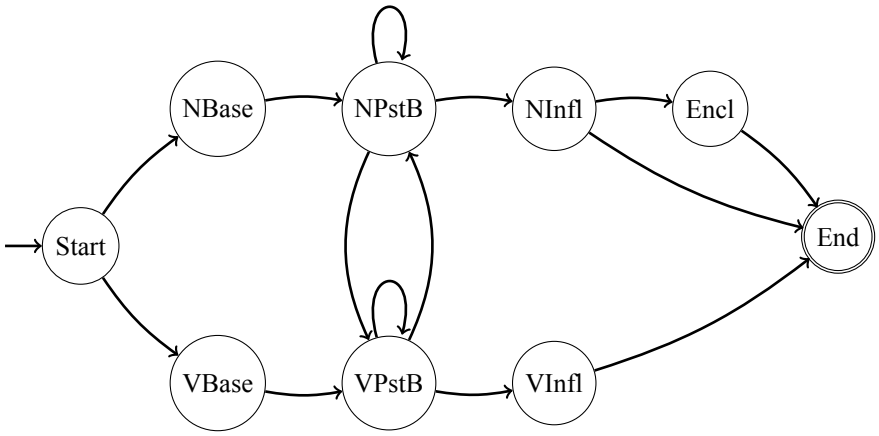


Figure 1: Finite-state network that illustrates the general design of the Yupik analyzer with the exception of some root bases and continuation classes, such as demonstratives and numerals. Of the ones shown, `NBase` and `VBase` refer to the noun base and verb base lexicons, while `NPstB` and `VPstB` refer to the respective postbase lexicons. The transition arcs between the postbase lexicons account for nominalizing and verbalizing postbases, while the `NInfl` and `VInfl` continuation classes handle case and mood inflection, and `Encl` handles enclitics.

senting each process are somewhat lexicalized, and hint at the process being represented (see Table 1). In all examples we use the standard listing order and notation used in the Yupik grammar (Jacobson, 2001) and the Yupik dictionary (Badten et al., 2008) as well as Leipzig glossing conventions. For the most part, the ordering of the morphophonological symbols listed with each postbase reflects the order in which the morphophonological processes apply, as in **@**$\sim_{\mathbf{f}}$**–ragkiigh** (*to V quickly*). Modification of base-final *te*, which is represented by the symbol **@**, occurs before base-final *e* is dropped via $\sim_{\mathbf{f}}$. Otherwise, the ordering of the symbols is arbitrary, since the morphophonological processes being represented are mutually exclusive, as in $\sim_{\mathbf{f}}$ and **–**, where the latter drops base-final consonants. As a base cannot end in a consonant and an *-e*, the only necessary symbol ordering in the postbase **@**$\sim_{\mathbf{f}}$**–raghkiigh** is between **@** and $\sim_{\mathbf{f}}$. Any allomorphy in the postbase, however, is always handled first.

Following these conventions, the subsequent example illustrates the full derivation of the Yupik word **aghnaaguq**[2].

(1) aghnaaguq
aghnagh- -∼:(ng)u- -∼$_{f}$(g/t)u- -q
woman- -to.be.N- -INTR.IND- -3SG
'*She is a woman*'

The verbalizing postbase **∼:(ng)u**, with its morphophonological processes underlined, affixes first:

1. **(ng)** If the base ends in a vowel, affix *ng*
2. **∼** If *e* appears in base-final or penultimate (semi-final) position, drop *e*
3. **:** If *gh* appears between two vowels that can be lengthened, drop *gh*

Following this first iteration, the resulting intermediate form is **aghnau-** (*to be a woman*). Yupik phonology forbids unlike vowel clusters, however, so *u* assimilates to *a* in a process known as *vowel dominance* to yield **aghnaa-**.

The affixation of the postbase **∼$_{\mathbf{f}}$(g/t)u** marks the valence and verb mood as intransitive indicative:

1. **(g/t)** If the base ends in a vowel or a consonant, affix allomorphs *g* or *t* respectively
2. **∼$_{\mathbf{f}}$** If base ends in final *e*, drop *e*

Finally the person/number marker **q** completes the surface form. Thus, from the string *aghnagh-∼:(ng)u-∼$_{f}$(g/t)u-q*, we derive **aghnaaguq**.

2.3. Derivation of a More Intricate Word

Here, we derive a Yupik word of greater complexity, **pagunghalighnaqaqa** (*I am going to put crowberries in…*), that consists of two derivational postbases, and four unique morphophonological processes contained within those postbases. The underlying gloss and source sentence are given in Example 2, while a full listing of the standard morphophonological symbols is also provided in Table 1 as a reference.

The verbalizing postbase **–ligh**, with its single morphophonological process underlined, affixes first to the base **pagunghagh***.

1. **–** If the base ends in a consonant, including strong *gh* (see Table 1), drop the consonant

Following this iteration, the first intermediate form is **pagunghaligh-** (*to put crowberries in*).

Affixation of the verb-elaborating postbase **@**$\sim_{\mathbf{f}}$**naqe** is next:

1. **@** For this postbase, if the base ends in *te*, drop *te*
2. **∼$_{\mathbf{f}}$** If the base ends in final *e*, drop *e*

The second intermediate form is then **pagunghalighnaqe-** (*to be going to put crowberries in*).

[2]Note that we deviate from Leipzig glossing conventions in the following instances: N = *Noun*, V = *Verb*, POSS = *Possessor*, UNPD = *Unpossessed Noun*, ABL_MOD = *Ablative-Modalis Case*, OPT = *Optative Mood*

(2) Naagpek sagnegha

naagpek	sagnegha
naa-–gpek-	-saghnegh∼:(ng)a-
mother-2SG.POSS	bowl-3SG.POSS

pagunghalighnaqaqa

pagunghagh*-	-–ligh-	-@∼$_f$naqe-	-∼(g)a-	-qa
crowberry-	-to.put.N.in-	-to.be.going.to.V-	-TRANS.IND-	-1SG.3PL

'*I am going to put crowberries in…*' (Jacobson, 2001)

Symbol	Description
∼	Drops *e* in penultimate (semi-final) position or *e* in base-final position and *hops* it *e-Hopping* is the process by which a full vowel in the first syllable of the base is lengthened as a result of dropping semi-final or final-*e*, so termed because it is as if the *e* has "hopped" into the first syllable and assimilated. *e-Hopping* will not occur if doing so would result in a three-consonant cluster within the word or a two-consonant cluster at the beginning (Jacobson, 2001).
∼$_f$	Drops final-*e* and hops it
∼$_{sf}$	Drops semi-final *e* and hops it
-w	Drops weak final consonants, that is, *gh* that is explicitly marked without an *. Strong *gh* is denoted as *gh**
:	Drops uvulars that appear between single vowels
–	Drops final consonants
—	Drops final consonants and preceding vowel
@	Indicates some degree of modification to base-final *te*, the degree of which is unpredictable and dependent on the postbase
+	Adds ending as presented This symbol is excluded from the Yupik analyzer and is implicitly assumed when no other morphophonological symbols are present.

Table 1: List of all documented morphophonological processes in Yupik and their lexicalized symbols

The affixation of the inflectional postbase ∼**(g)a** marks the valence and verb mood as transitive indicative:

1. **(g)** If the base ends in a double vowel, affix *g*
2. ∼ If *e* appears in base-final or penultimate (semi-final) position, drop *e*

The last intermediate form, **pagunghalighnaqa-**, is then followed by affixation of the person/number marker **-qa**, which completes the derivation.

3. Finite State Morphology: *foma* and *lexc*

A finite state transducer is the ideal mechanism for computationally modeling morphology, since it maps a bidirectional relation between two sets of strings, that is, an underlying form and a surface form (Beesley and Karttunen, 2003).

We implemented our morphological analyzer in *foma* (Hulden, 2009b), which is responsible for the composition of string-transforming rules to derive the surface string from the underlying string and vice versa. We encode the lexicon in the *lexc* format used by *foma*. Each lexical item is associated with a *continuation class*, and is encoded with an underlying form and an intermediary form that passes through the transducer in *foma* to generate the surface form.

Also included within *lexc* are the definitions of multicharacter symbols, typically glossing abbreviations such as [N] and [V], as well as any multi-graphemic phonemes (§ 2.).

4. Implementation

The Yupik analyzer is strictly implemented within the *lexc* and *foma* languages, although lexical transducers for sister languages to Yupik such as Iñupiaq have incorporated other resources such as XML databases (Bills et al., 2010), and an analyzer for Inuktitut developed by the Institute for Information Technology of the National Research Council of Canada was implemented in Java (Institute for Information Technology, 2012). Limiting the programming of the analyzer to *foma* was sufficient for our purposes, however, as there are a number of APIs to bridge the completed transducer with external utilities such as a spell-checker (Hulden, 2009a).

4.1. *lexc* File

The Jacobson (2001) reference grammar contains approximately 600 root bases and 80 derivational morphemes, in addition to the extensive inflectional morphology for noun case and verb mood. From this, the *lexc* file was crafted by hand to ensure that each lexicon was followed by the proper continuation class to generate a comprehensive set of permissible underlying strings (Figure 1).

The underlying forms and intermediate forms involved in the derivation of **aghnaaguq** are displayed in Figure 2. The noun base **aghnagh** is selected from the `NounBase` root lexicon, and proceeds to the `NounPostbase` continuation class, where it concatenates with the underlying form of the verbalizing postbase, **∼:(ng)u[N→V]**. The string derived thus far then continues to the `VInfl` continuation class, and is concatenated with the underlying gloss of the inflectional ending, [V][INTR][IND][3SG]. Eventually, the underlying forms of the postbase and inflectional ending, which are included for clarity and readability, are rewritten as the intermediate forms, **∼:(ng)u** and **∼$_f$(g/t)uq** respectively, for processing in *foma*.

```
LEXICON NounBase
aghnagh NounPostbase;                !woman

 LEXICON NounPostbase
–ghhagh[N→N]:–ghhagh NounPostbase; !dear N
∼%:(ng)u[N→V]:∼%:(ng)u VerbInfl;  !to be N

LEXICON VerbInfl
[V][Ind][3Sg]:∼f(g/t)uq #;
```

Figure 2: Sample *lexc* file that traces the underlying strings of '*aghnaaguq*' and '*aghnaghaaguq*' (see Examples 1 and 3). Exclamation points mark the beginning of a comment, and the percent sign is an escape character.

The noun-elaborating postbase **–ghhagh** is likewise included in the NounPostbase lexicon, where the morphophonological symbol – drops all base-final consonants. Its continuation class is the NounPostbase lexicon itself, and this form of self-reference permits the recursive attachment of derivational postbases to yield strings such as the one show below:

(3) aghnaghhaaguq
aghnagh- -–ghhagh- -∼:(ng)u- -∼f(g/t)u- -q
woman- -dear.N- -to.be.N- -INTR.IND- -3SG
'*She is a dear woman*'

4.1.1. Flag Diacritics

The natural valency of a Yupik verb dictates the form of the inflectional ending, since the ending marking intransitivity differs from the ending marking transitivity with respect to the morpheme and morphophonological processes. For instance, the indicative mood postbase for intransitives is **∼f(g/t)u**, while the analogous ending for transitives is **∼(g)a**.

Foma includes a feature called *flag diacritics* to handle long-distance dependencies in words, effectively constraining the morphemes that may co-occur. Each flag diacritic has the form @FLAGTYPE.FEATURE.VALUE@, where *feature* and *value* refer to arbitrary strings set by the programmer (Hulden, 2011). A flagtype value of P indicates that the *feature* should be set to *value*, while a flagtype value of R requires the *feature* to already be set to *value*. We use the following flag diacritics:

1. @P.VALENCE.INTR@
2. @P.VALENCE.TRNS@
3. @R.VALENCE.INTR@
4. @R.VALENCE.TRNS@

Strings that contain morphemes with mismatched diacritics are discarded, as demonstrated in the expanded *lexc* file presented in Fig. 3. For instance, transitive verb **ungipaate** (*to tell*) has its VALENCE feature set to TRNS via its continuation class, VerbTrns. It may optionally receive a verbal postbase, or directly proceed to inflection where its VALENCE feature is checked in the VerbInfl continuation class. A mismatch in flag diacritics then prevents the concatenation of **ungipaate** with the intransitive inflectional ending. Similar circumstances arise for intransitive verb

```
LEXICON NounBase
aghnagh NounPostbase;              !woman

LEXICON VerbBase
nagate VerbTrns;    !to listen
ungipaate VerbTrns; !to tell
nagate VerbIntr;    !to listen
umughqaa VerbIntr;  !to have sleep paraly-
sis

LEXICON NounPostbase
–ghhagh[N→N]:–ghhagh NounPostbase;  !dear N
∼%:(ng)u[N→V]:∼%:(ng)u VerbIntr; !to be N

LEXICON VerbTrns
@P.VALENCE.TRNS@ VerbPostbase;

LEXICON VerbIntr
@P.VALENCE.INTR@ VerbPostbase;

LEXICON VerbPostbase
0:0 VerbInfl;
@lleqe[V→V]:@lleqe VerbPostbase; !will V

LEXICON VerbInfl
@R.VALENCE.TRANS@ VerbInflTrns;
@R.VALENCE.INTR@  VerbInflIntr;

LEXICON VerbInflTrns
[V][Ind][3Sg][3Sg]:∼(g)aa #;

LEXICON VerbInflIntr
[V][Ind][3Sg]:∼f(g/t)uq #;
```

Figure 3: Sample *lexc* file that is expanded to include the flag diacritic continuation classes to oversee verb valence constraints.

umughqaa (*to have sleep paralysis*), while ambitransitive verb **nagate** (*to listen*) recognizes both endings.

4.2. *foma* File

The morphophonological processes responsible for transforming strings are individually implemented in the *foma* file as rules that trigger the pertinent transformation under specific environmental conditions. A subset of these rules are shown in Figure 4. Rules take the form $A \rightarrow B \,||\, \Gamma_\Delta$, where A is rewritten as B in the context of Γ and Δ. In the Yupik analyzer, A typically refers to the substring that is rewritten as B in the morphophonological context determined by Γ and Δ, where Δ may refer to the morphophonological symbol. As such, the *foma* file consists of all contextual rewrite rules defined according to the morphophonological processes they represent, concluding with a single rules cascade that composes the rewrite rules together in the order that the morphophonological processes occur. In accordance with the discussion in § 2.2., the contextual rewrite rule modifying base-final *te* appears earlier in the cascade relative to the rule that drops base-final *e*. The full finite-state grammar represents the composition of the *lexc* lexi-

```
read lexc xample.lexc
⋮
define ResolveAllomorphy
  "(ng)" -> ng || V _ .o.
  "(ng)" -> 0 || C _ .o.
  "(g/t)" -> g || V _ .o.
  "(g/t) -> t || C _ .o.
  "(ng)" -> 0 .o.
  "(g/t)" -> 0;
⋮
define UvularDropping
  "gh" -> 0 || [V - e] _ ":" [V - e] .o.
  ":" -> 0;
⋮
define Grammar [
  Lexicon .o.
  ResolveAllomorphy .o.
  SemiAndFinalE .o.
  UvularDropping .o.
  FinalE .o.
  VowelDominance ];
```

Figure 4: Sample *foma* file that contains implementation of some morphophonological rules, composed together with the '.o.' operator to form a finite-state grammar.

con with these contextual rewrite rules.

Yupik morphophonology requires that these rewrite rules be completely applied at each successive morpheme boundary in sequence. As such, the grammar shown in Figure 4 is insufficient to correctly process most words. When processing the underlying form from Example 1, the grammar in Figure 4 incorrectly yields **aghnaauq** as the surface form, instead of **aghnaaguq**. In this example, the morphophonological symbols *(ng)* and *(g/t)* in the underlying string *aghnagh-∼:(ng)u-∼$_f$(g/t)u-q* simultaneously delete to no effect, since neither appear in the contexts specified in the transducer. The premature deletion of *(g/t)* falsely yields **aghnaauq**. Although the rules in Figure 4 are composed in the correct order, the fact that the rules are applied to all morpheme boundaries simultaneously results in an incorrect derivation.

To remedy this, we configured the transducer to resolve the morphophonological processes of each successive morpheme boundary in its entirety before considering the morphophonological processes of the next boundary. This was accomplished by having the original single cascade of morphophonological processes iterate eight times in anticipation of seven potential derivational postbases (§ 2.1.) and subsequent inflection. Each character string used in *foma* was further categorized as either an alphabetic character (*Alph*) or a morphophonological symbol (*MPSymbols*), and a morpheme boundary marker ^ was introduced at every juncture of an alphabetic character and a morphophonological symbol. The required context for each rule was modified such that the rule applies only at the leftmost unprocessed morpheme boundary. At the end of each iteration, the leftmost morpheme boundary marker, and the associated morphophonological symbols, are permitted to delete, thus setting up the requisite context for the next iteration. Figure 5 illustrates these changes.

```
read lexc xample.lexc

define Alph [ "*" | a |…| y ];
define MPSymbols [ (g/t) | (ng) |":"|…];
define MBndry "^";
define WBndry [ .#. ];

define InsertMBndry
  [..] -> MBndry || Alphabet _ MPSymbols;
define CleanupMBndry
  MBndry -> 0 || WBndry Alph+ _;
⋮
define ResolveAllomorphy
  "(ng)" -> "ng" || V MBndry _ .o.
  "(ng)" -> 0 || C MBndry _ .o.
  "(g/t)" -> g || V MBndry _ .o.
  "(g/t) -> t || C MBndry _ .o.
  "(ng)" -> 0,
  "(g/t)" -> 0 || WBndry Alph+ MBndry _;
⋮
define UvularDropping
  "gh" -> 0 || [V - e] _ MBndry MPSymbols*
":" [V - e] .o.
  ":" -> 0 || WBndry Alph+ MBndry _ ;
⋮
define Grammar [
  Lexicon .o.
  InsertMBndry .o.
  !! ITERATION 1 !!
  ResolveAllomorphy .o.
  SemiAndFinalE .o.
  UvularDropping .o.
  FinalE .o.
  VowelDominance .o.
  CleanupMBndry .o.
    ⋮
  !! ITERATION 8 !!
  ResolveAllomorphy .o.
  SemiAndFinalE .o.
  UvularDropping .o.
  FinalE .o.
  VowelDominance .o.
  CleanupMBndry ];
```

Figure 5: Sample *foma* file that correctly derives the surface string '*aghnaaguq*', where '+' regex operator denotes one or more of the preceding string.

The derivation of the string **aghnaaguq** now models the iterative process first described in § 2.2. From the underlying string *aghnagh^∼:(ng)u^∼$_f$(g/t)uq*, the first postbase applies in its entirety which allows the leftmost morpheme boundary marker to delete, but retains the morphophonological symbol *(g/t)* until application of the second and final

(4) Piyukuvek qergesek qelpeghtikek

piyukuvek	qergesek	qelpeghtikek
piyug[V][Intr][Cond][2Sg]	**qergese[N][Abs][Unpd][Du]**	**qelpeghte[V][Trns][Opt][PRS][2Sg][3Du]**
piyug-@∼$_{sf}$–(g)k(u)vek	qergese-∼$_{sf}$-w:(e)-k	qelpeghte-∼$_{f}$(i)kek
person.walking.in.distance-INTR.COND.2SG	window-UNPD.ABS-2DU	to.open-TRANS.PRS.OPT.2SG.3DU

'*Open the window(s) if you (take a) walk*' (Jacobson, 2001)

(5) Tukuqa neghsameng gaaghaquq

tukuqa	neghsameng	gaaghaquq
tukugh[N][Abs][1SgPoss][SgPosd]	**neghsagh[N][Abl_Mod][Unpd][Sg]**	**gaagh∼(g$_1$)aqe[V→V][V][Intr][Ind][3Sg]**
tukugh-∼–ke	neghsagh-∼$_{f}$-wmeng	gaagh-∼(g$_1$)aqe-∼$_{f}$(g/t)u-q
host-1SG.POSS	seal-UNPD.ABL_MOD.SG	to.cook-to.be.Ving-INTR.IND-3SG

'*My host is cooking seal*' (Jacobson, 2001)

(6) Mangteghaghllangllaghyugtukut
mangteghagh–-ghllag[N→N]–-ngllagh[N→V]-@$_1$ ∼$_{f}$yug[V→V][V][Intr][Ind][1Pl]
mangteghagh–-ghllag–-ngllagh-@∼$_{f}$yug-∼$_{f}$(g/t)u-kut
house-huge.N-to.build.N-to.want.to.V-INTR.IND-1PL
'*We want to build a huge house*' (Jacobson, 2001)

	Precision	Recall	F-Measure
Types	97.81	97.11	97.46
Tokens	98.20	97.61	97.90

Table 2: Reported precision, recall, and f-measure values of the Yupik analyzer when evaluated against the end-of-chapter translation exercises of the reference grammar.

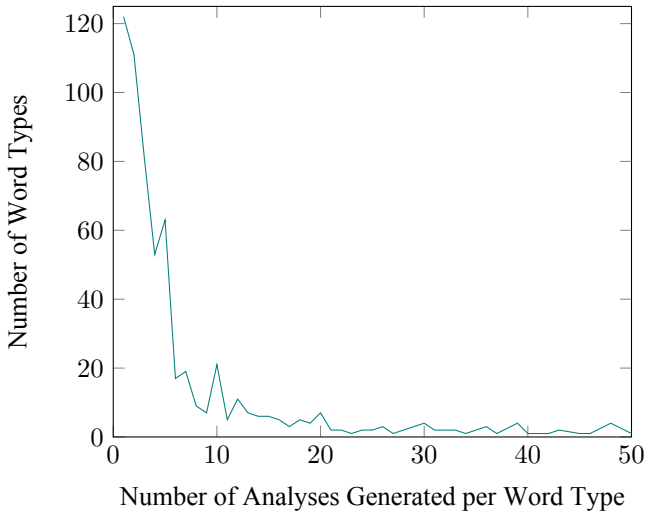


Figure 6: Of the Yupik word types in the Jacobson (2001) end-of-chapter exercises for which the analyzer returns a result, the analyzer returns five or fewer analyses for 67%. There are 30 Yupik word types from these exercises for which the analyzer returns more than 50 analyses.

postbase.

5. Evaluation

While the Jacobson (2001) reference grammar was instrumental to the implementation of the analyzer, the grammar nonethless was somewhat lexically impoverished, listing no more than 600 noun and verb bases and 80 postbases of the approximately 8000 nouns and verbs and 600 postbases documented in the Yupik dictionary. In order to more precisely evaluate the efficacy of the analyzer, the remaining lexical entries were added to the *lexc* file via a semi-automated process that organized the dictionary entries by part of speech, that is, as either a noun, verb, particle, or one of the four derivational postbases. Placement into one of these classes was determined by exploiting patterns in the dictionary definitions, for instance, the definition of a nominalizing postbase typically contained the phrase "one who…" or "one that…", while the definition of a verb-elaborating postbase contained "to…V". In all, some 4000 noun bases, 4000 verb bases, 600 postbases, and 500 particles were newly integrated into the analyzer's lexicon. As a result, when evaluated against the end-of-chapter translation exercises provided in the reference grammar, the morphological analyzer performed reasonably well, successfully parsing several hundreds of words of varying morphological complexity (see Examples 4–6).

Numerically, the end-of-chapter exercises consisted of 281 Yupik sentences to be translated by the reader, summing to 796 individual tokens, of which 658 were unique. The analyzer could not derive parses for 19 tokens, while gold standard translations provided by a native Yupik speaker suggested that all of the analyses returned for 14 of the 658 types were incorrect.

Precision, recall, and f-measure values were then calculated for types versus tokens and are reported in Table 2.

While these values attest to the coverage of the analyzer, its efficiency and accessibility was further assessed by examining its output for all parsed words.

In particular, the average number of analyses generated per word was 20, while the median was only five, a discrepancy that suggested some words in the end-of-chapter exercises were generating an unusually high number of analyses, thus distorting the average. To qualify these observances, a script was implemented to identify outliers, and found that the word with the greatest number of analyses, **laalighfiknaqaqa**, had *6823* analyses followed by **laalighfikiikut** with 1074. Although these values suggest that the quality of output of the morphological analyzer was at times substandard, instances of acute overgeneration were rela-

(7) Laalighfikiikut
***laaligh-—i[N→V]-@$_1$ ~:(u)n[V→N]-–ghte[N→V]-@$_1$ ~$_f$vik[V→N]-—i[N→V][V][Intr][Ind][1Pl]**

51	tukfighinaluni
54	inimakanga
54	qikmilguyugtunga
59	lliiki
62	seghleghunii
63	ungipaatinga
67	laalightaqngavnga
72	qavaghniinga
76	tagitiki
81	mangteghaanituq
86	atightughaqsin
92	aghveliighsiin
92	naliita
98	kiyaghhneghinniluki
125	aakaqa
136	neghyaqunaan
137	pagunghalighnaqaqa
141	ungipaataanga
143	guunnaqaqa
185	atuqnaqegkeni
200	esghaataqukut
215	liilleqii
270	nallukaqa
315	atuqnaqaa
364	akitutaqukung
383	alikaqa
414	aghvightesugiinkut
461	llinaqaqa
1074	laalighfikiikut
6823	laalighfiknaqaqa

Table 3: Yupik word types from the Jacobson (2001) end-of-chapter exercises for which the analyzer returns more than 50 analyses, along with the number of analyses returned for each such word type.

tively rare (see Figure 6 and Table 3). Of the 658 unique types in the corpus, 136 types or approximately 20.64% generated more than ten analyses, while 77 types or approximately 11.68% generated more than 20 analyses. Concerning more severe cases, only 16 types or approximately 2.43% generated more than 100 analyses.

6. Ongoing Work

As a result of these evaluations, the most pressing issue at present concerns curtailing the instances of severe overgeneration, and adapting the analyzer to generate output that is minimal but correct, and amenable to incorporation into a Yupik language spell-checker and pedagogical materials for students. Fortunately, as demonstrated by the **laalighfiknaqaqa** and **laalighfikiikut** case examples, there seems to exist a pattern among those words that are most susceptible to overgeneration, and identifying these patterns may assist in paring down the number of analyses generated per word.

For instance, one proposed nonsense analysis of the word **laalighfikiikut** is given in Example 7, where the verbalizing postbase **—i[N→V]** (*to make N*) materializes twice, resulting in semantic infelicity. Resolving this could be as simple as programming a `Filter` function that filters out any undesired permutations of strings, although the feasibility of such a solution would require a better understanding of the scope to which this overgeneration occurs.

It is likewise critical that the analyzer eventually be evaluated against texts other than the reference grammar, to ensure completeness of the dictionary, and of the morphophonological rules presented in the grammar. We have already identified several gaps in the documentation concerning certain Yupik linguistic phenomena, including aspects of the demonstrative and numeral systems. In particular, the reference grammar presents demonstrative inflection as a series of eleven "paradigmatic sets", and within each set, demonstratives take on an additional six base forms that dictate the inflected form (Jacobson, 2001, p.110). It is unclear however, how these sets and base forms were determined, and to what degree they differ from one another with respect to inflection. Concerning Yupik numerals, the reference grammar says little beyond the fact that they may serve as appositives to other nouns, incorporate into verbs via verbalizing postbases, and may also inflect for the specific *ablative-modalis* case (Jacobson, 2001, p.112), raising the issue of whether there are limitations to the suffixing patterns of numerals. Since understanding these systems may require a considerable amount of fieldwork and elicitation, these systems are, at the moment, simply hard-coded into the *lexc* file. A more elegant and desired solution would designate each of these grammatical elements as individual `Root` lexicons with the relevant continuation classes that reflect the element's inflection patterns.

Lastly, it is our intention to eventually extend this morphological analyzer to parse Yupik written in the Cyrillic script as well. We imagine that the morphophonological rules cascade should remain relatively the same, although the graphotactic constraints differ – for instance, the Cyrillic Yupik orthography differentiates rounded consonants from their unrounded counterparts by means of the Cyrillic grapheme 'ÿ'. Any difference in language varieties should also be accounted for, although these are believed to be relatively minor (Krauss, 1975).

7. Conclusion

The postbase attachment process presented herein for Yupik may have implications for morphophonological theory, in that it lends credence to the idea that derivational morphology must be performed cyclically in some languages in order to derive the proper surface form (§ 2.2.).

In discussing the implementation of the Yupik morphological analyzer, however, we have presented a design technique to handle this phenomenon, and anticipate that usage of the Foma toolkit in this way may be adapted to lan-

guages that require such a processing pattern, for instance, other languages in the Inuit-Yupik language family, which are the most typologically similar to Yupik. Nevertheless, while much work remains in this respect and in regards to efficacy evaluation, our implementation incorporates all the lexical items and morphophonological rules of Yupik that have been documented and described to date.

8. Bibliographical References

Badten, L. W., Kaneshiro, V. O., Oovi, M., and Koonooka, C. (2008). *St. Lawrence Island / Siberian Yupik Eskimo Dictionary*. Alaska Native Language Center, University of Alaska Fairbanks.

Beesley, K. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications, Stanford, CA.

Bills, A., Levin, L. S., Kaplan, L. D., and MacLean, E. A. (2010). Finite state morphology for Iñupiaq. *7th SaLTMil Workshop on Creation and Use of Basic Lexical Resources for Less-Resourced Languages*, pages 19–26.

de Reuse, W. J. (1994). *Siberian Yupik Eskimo — The Language and Its Contacts with Chukchi*. Studies in Indigenous Languages of the Americas. University of Utah Press, Salt Lake City, Utah.

Hulden, M. (2009a). Fast approximate string matching with finite automata. *Procesamiento del Lenguaje Natural*, 43:57–64.

Hulden, M. (2009b). Foma: a finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 29–32. Association for Computational Linguistics.

Hulden, M. (2011). Morphological analysis tutorial: A self-contained tutorial for building morphological analyzers. `https://fomafst.github.io/morphtut.html`. Accessed: 2017-09-19.

Institute for Information Technology. (2012). The UQAILAUT Project: Inuktitut Morphological Analyzer. http://inuktitutcomputing.ca/Uqailaut/info.php.

Jacobson, S. A. (2001). *A Practical Grammar of the St. Lawrence Island / Siberian Yupik Eskimo Language, Preliminary Edition*. Alaska Native Language Center, Fairbanks, Alaska, 2nd edition.

Koonooka, (Petuwaq), C. (2005). Yupik language instruction in Gambell (St. Lawrence Island, Alaska). *Études/Inuit/Studies*, 29(1/2):251–266.

Krauss, M., Holton, G., Kerr, J., and West, C. T. (2010). Indigenous peoples and languages of Alaska. ANLC Identifier G961K2010.

Krauss, M. (1975). St. Lawrence Island Eskimo phonology and orthography. *Linguistics: An International Review*, 13(152):39–72, January.

Krupnik, I. and Chlenov, M. (2013). *Yupik Transitions — Change and Survival at Bering Strait, 1900-1960*. University of Alaska Press, Fairbanks, Alaska.

Schwalbe, D. M. (2017). Sustaining linguistic continuity in the Beringia: Examining language shift and comparing ideas of sustainability in two Arctic communities. *Anthropologica*, 59(1):28–43.

Universal Morphologies for the Caucasus region

Christian Chiarcos Kathrin Donandt Maxim Ionov
Monika Rind-Pawlowski Hasmik Sargsian Jesse Wichers Schreur
Frank Abromeit Christian Fäth
{chiarcos|donandt|ionov|abromeit|faeth}@informatik.uni-frankfurt.de
{sargsian|wichersschreur}@em.uni-frankfurt.de
{rind-pawlowski}@lingua.uni-frankfurt.de
Goethe University Frankfurt, Germany

Abstract

The Caucasus region is famed for its rich and diverse arrays of languages and language families, often challenging European-centered views established in traditional linguistics. In this paper, we describe ongoing efforts to improve the coverage of Universal Morphologies for languages of the Caucasus region. The Universal Morphologies (UniMorph) are a recent community project aiming to complement the Universal Dependencies which focus on morphosyntax and syntax. We describe the development of UniMorph resources for Nakh-Daghestanian and Kartvelian languages as a well as for Classical Armenian, we discuss challenges that the complex morphology of these and related languages poses to the current design of UniMorph, and suggest possibilities to improve the applicability of UniMorph for languages of the Caucasus region in particular and for low resource languages in general. We also criticize the UniMorph TSV format for its limited expressiveness, and suggest to complement the existing UniMorph workflow with support for additional source formats on grounds of Linked Open Data technology.

Keywords: morphology, Caucasus, UniMorph

1. Background

The Universal Morphology project (Sylak-Glassman et al., 2015b, UniMorph)[1] is a recent community effort aiming to complement the Universal Dependencies (Nivre and others, 2015, UD),[2] which focus on syntax, with coverage of morphology. We describe the development of UniMorph resources for languages of the Caucasus region, known for its rich and diverse arrays of languages and language families, and often posing challenges to European-centered views established in traditional linguistics. In particular, we focus on Nakh-Daghestanian (North-East Caucasian) and Kartvelian (South Caucasian) languages, as well as on Classical Armenian, and discuss challenges that these and related languages pose to the current design of UniMorph.

A practical challenge for linguists working with dictionary data consists of linking it with text data. Corpus-based research thus requires computational models of the morphology of the languages under consideration, i.e., lemmatization, at least. But also for low-resource languages (for which few or small amounts of corpus data exist or have to be collected), an explicit treatment of morphology is necessary for the study of language contact, especially if morphologically rich languages are involved (as in the Caucasus area): Neither inherited words nor loan words are transferred between language(stage)s in their base form only. Accordingly, the computational handling of complex morphological processes and features are important for grasping interrelations of Caucasian languages.

The over 100 languages spoken in the Caucasus are grouped into several language families, out of which three are indigeneous, i.e., Caucasian in a strict sense: Kartvelian or South Caucasian, Abkhazo-Adyghean or (North-)West Caucasian and Nakh-Daghestanian or (North-)East Caucasian. A fourth language family with roots in the Caucasus, Hurro-Urartian, is known only from epigraphic records and assumed to be extinct for more than 2000 years.

With respect to morphosyntax, certain typological traits are frequently encountered in Caucasian languages: (Klimov, 1994)[3]: (1) use of agglutination, with a varying degree of inflective elements, (2) verbocentric sentence structure and complex verbal morphology, often including agreement with multiple syntactic arguments, (3) features of ergative, where the subject argument of intransitive verbs receives the same morphological case as the object of transitive verbs (absolutive case), whereas the transitive subject receives ergative case[4], and (4) in Nakh-Daghestanian languages: rich case systems, with up to more than 40 morphological cases. In addition, all living languages in the Caucasus are low-resource (except for Georgian and Armenian which have considerable amounts of written literature), and many exhibit traces of intense language contact with Iranian, Armenian, Georgian, Turkic, Arabic and/or Russian (reflecting shifting patterns of political dominance in the last 2,500 years).

2. Universal and language-specific morphology

Following the success of the Universal Dependencies as a growing community project, a similar effort for the de-

[1] http://unimorph.github.io/

[2] http://universaldependencies.org/

[3] These characteristics do not apply to Armenian, which is an Indo-European language, albeit 'as Caucasian as an Indo-European language could possibly become' (Gippert, p.c., May 2017).

[4] In addition, active-inverse structures can be found in several Caucasus languages, as manifested, for example, in the Kartvelian 'narrative' case (which is, however, often referred to as 'ergative' in Western linguistics).

velopment of cross-linguistic features for inflectional morphology has been initiated: Universal Morphology. Both projects aim to develop features and categories which are *cross-linguistically applicable* (not necessarily universal in the sense of any notion of 'universal grammar'). As such, the UniMorph annotation schema "allow[s] any given overt, affixal (non-root) inflectional morpheme in any language to be given a precise, language-independent definition ... [by means] of a set of features that represent semantic "atoms" that are never decomposed into more finely differentiated meanings in any natural language" (Sylak-Glassman et al., 2015b, p.674).

2.1. UniMorph inventories

The UniMorph data format is a list of tab-separated values for one word per line, with columns for the word form, the lemma and morphological features; it is thus roughly comparable to the CoNLL format as previously used for, e.g., syntactically annotated corpora of Classical Armenian (Haug and Jøhndal, 2008).[5] The primary data structure of UniMorph is an unordered set of semicolon-separated, unqualified features. Figure 1 shows an example of a conventional gloss of the Megrelian word *kešerxvaduk* 'I will meet you' together with its UniMorph representation.

UniMorph resources are rarely original resources, but rather extracted from existing material,[6] such as Wiktionary (Kirov et al., 2016, first-generation UniMorph inventories) and other dictionaries, bootstrapped from morpheme inventories or corpora (as described here), or generated by rule-based morphologies. However, this conversion-based approach means that the segmentation and annotation principles of the underlying resource tend to be preserved.

In general, UniMorph follows a word-based approach to morphology where inflected forms are organized in paradigms, but their internal structure left unanalyzed. In language documentation, however, a morpheme-based approach prevails, i.e., words are segmented into morphemes which are annotated with the linguistic features that they encode. This can lead to vastly different analyses: In morpheme-based annotation, a number of language-specific features are inflectional morphemes that *contribute* to the indication of morphosemantic features rather than to unambiguously *indicate* them. As such, two Megrelian morphemes in Fig. 4b conspire with other TAM markers to indicate tense, aspect and mood (resp., valency). Which morphosemantic (UniMorph) category these morphemes resolve into, remains, however, unspecified as it cannot be automatically deducted from the original resource. We thus describe their function by means of language-specific labels, here LGSPEC6 and LGSPEC7.

[5] https://proiel.github.io/

[6] The primary reason is that UniMorph morpheme inventories are actually rather uninformative, as the format does not permit to provide translations, examples or metadata, e.g., regarding the source of a particular form. Any serious effort to create morpheme inventories in the context of language documentation or philology thus requires an extended format, from which UniMorph TSV files are then to be extracted.

ma	**si**	**kešerxvaduk**					
ma	si	ke-	še-	r-	xvad	-u	-k
1SG	2SG	AFF	PV	O2	meet	TM	S1/2SG

```
xvad kešerxvaduk AFF;V;LGSPEC4;ARGDA2S;
ARGNO1S;LGSPEC6
```

Figure 1: Megrelian *(ma si) kešerxvaduk* 'I will meet you' as conventional interlinear glossed text (above) and in UniMorph LEMMA - FORM - FEATS representation (below)

2.2. Caucasian languages in UniMorph

Already during the design of the UniMorph guidelines (Sylak-Glassman, 2016), Nakh-Dagestanian languages have been taken into consideration for some phenomena, e.g., with respect to the 'universal' gender features NAKH1, ..., NAKH8 for Nakh-Dagestanian noun classes (Sylak-Glassman, 2016, p.27). Selected features of Abkhazo-Adyghean (on argument marking, p.12-13; on interrogativity, p.29), and Kartvelian (on evidentiality, p.25) have been mentioned, too. Beyond this, languages from the Caucasus area are not discussed in relation to the UniMorph schema and the UniMorph repositories comprise datasets for only Modern Georgian and Modern East Armenian. The datasets provided as result of our efforts thus constitute a major increase in coverage of languages of the Caucasus area. We created morphologically annotated datasets in the UniMorph data format for Megrelian (Kartvelian), Khinalug (Nakh-Daghestanian) and Classical Armenian (Indo-European). Additional data on Batsbi (Nakh-Daghestanian) is in preparation.

2.3. Language specific features

In addition to universal features, UniMorph conventions permit language-specific features to be represented by LGSPEC, followed by a numerical index. Although a separate file that defines those markers can be provided, limiting LGSPEC markers to numerical labels impedes the readability of this data, as the Megrelian example in Fig. 1 illustrates.

For languages with a greater number of language-specific features, this convention for the nomenclature of language-specific features may become problematic, as they are likely to be confused and errors in LGSPEC assignment cannot be easily spotted. A more transparent solution would thus be to allow extended LGSPEC labels with human-readable and established abbreviations as those used in conventional glossing. While difficulties in the choice of labels can be resolved relatively easily,[7] a more severe issue exists with respect to another characteristic of UniMorph, the requirement that features are both unordered and unqualified. Furthermore, when analyzing specific languages, descriptive linguists try to use terminology that fits the grammati-

[7] Resolving such difficulties requires a consensus in the UniMorph community to improve their labeling system or, alternatively, to develop and to document (language-specific) conventions how to deal with conflicting terminologies.

cal phenomenon under question best. Very often, the names and labels that are used are not particularly well suited for cross-linguistic comparison. For example the Kartvelian so-called 'thematic marker' (TM in Fig. 1), is a conventional label given to a morpheme that shows up in certain tense/aspect stems and not in others, but cannot be linked to a specific grammatical function.[8] Consequently, linguists would feel the need to use UniMorph's LGSPEC feature abundantly, to a point where cross-linguistical comparison (be it computational or not) would be impossible. Indeed, it is understood among many linguists that using a single label for two similar grammatical categories in two different languages can be misleading (Haspelmath, 2007). For example, the dative case in Kartvelian can function as the subject, direct object, and indirect object of a clause, which is fundamentally different from its prototypical function (in, say, Latin). The label 'Ergative case' is used in Megrelian to describe a morpheme that not only marks transitive objects, but also intransitive ones. Again, this use contrasts with its prototypical use.

If we follow this line of thought *in absurdum*, all grammatical features would be best translated into UniMorph's LGSPEC, making the procedure pointless. A solution might be found in Linked Data technology (see Section 5.1.). If grammatical features, once translated into UniMorph terminology (in which language-specific details of certain grammatical categories inevitably get lost), would still retain their link to their corresponding language-specific links in the original resource, the procedure would be less lossy, and when necessary, the original data would be easily retraceable.

3. On nominal inflection

3.1. Complex patterns of case marking

In nominal morphology, several instances of the same feature can be overtly realized and need to be distinguished. We discuss this for the double marking of case, which may arise, for example, for languages that provide morphological marking for the inherent case of a noun (reflecting the syntactic status of the noun), and the head case (reflecting the syntactic status of its head). At the moment, instances of such double-coding in nominal morphology are not covered by UniMorph.

Suffixaufnahme: The phenomenon of *Suffixaufnahme* was originally described for Old Georgian and Hurro-Urartian, but has also been documented for **Megrelian** (Boeder, 1995, p.194):[9]

(1) *gi-∅-a-ntχū-d-esə* *k'ata-sə* *te*
PV-O3-LOC-fall-IPF-S3PL.PST people-DAT this
χenc'əpe-ši *ɣəmala-ši-sə*
king-GEN dominion-GEN-DAT

'They attacked the people of this king's dominion.'

Here, *dominion* is not only marked for its inherent case (genitive), but also expresses the (dative) case of its head (*people*). With features regarded as a set (and thus, order-insensitive) of unqualified features, as defined in UniMorph, this information can only be preserved if inherent (genitive) case and head (dative) case are distinguished by different features.

Case attraction: A similar differentiation between inherent case and head case can be found in **Classical Armenian**, although without double-coding (Hübschmann, 1906, p.478-480):

(2) a. *i* *knoǰ-ê* *t'agawor-i-n*
by wife-ABL.SG king-GEN.SG-DEF

b. *i* *knoǰ-ê* *t'agawor-ē-n*
by wife-ABL.SG king-ABL.SG-DEF

'by the wife of the king'

Although Classical Armenian does not mark inherent and head case simultaneously, the regular (inherent) genitive case marking (2a) can be replaced by the morphological case of its head, especially for ablative (2b) or instrumental (Plank, 2014, p.20-21). For the annotation of corpus data, it would be important to distinguish inherent and head case, as they have an impact on syntactic parsing. Beyond the future alignment with the Universal Dependencies, this does not directly concern UniMorph, because the overtly realized case uses the same morphemes for, say, ablative, regardless of whether it indicates inherent or head case. Some languages do, however, provide separate sets of morphemes for both functions, and in these circumstances, it would be important to distinguish inherent case morphology from agreement-based case morphology.

Case combination: Another source of multiple case marking is case combination as found in **Khinalug**: Two cases suffixes can be combined in order to complement their functions. For example, when the ablative in *-(i)lli* attaches to nominal stems directly, it expresses the general ablative meaning, e.g. *mɯda-lli* mountain-ABL 'from the mountain'. However, it can also attach to three other cases. The case in -χ expresses both apudessive and approximative. When combining with the ablative, this leads to the expression of a movement 'away from near sth.', e.g.:

(3) a. *t'u* *quš*
REMT.REF1.ABV bird.ABS
mɯda-χ *učmuškui-'o-mä.*
mountain-APUD/APPRX fly.PRS-ABV-DECL

'The bird up there is flying towards the mountain.'

b. *t'u* *quš*
REMT.REF1.ABV bird.ABS
mɯda-χ-illi
mountain-APUD/APPRX-ABL
učmuškui-'o-mä.
fly.PRS-ABV-DECL

'The bird up there is flying away from near that mountain.'

[8] The label 'present/future stem formant' is misleading since it also shows up in the imperfective past.

[9] This example is likely to be a loan translation from Old Georgian.

The situation is more complex with the possessive-locative in *-š*. Among several other functions, this case attaches to recipient of the verb 'give', when the item is given away for temporary possession only (otherwise the recipient is dative-marked).[10] The ablative-marked form in *-š-illi* marks the former possessor with verbs of the meaning 'take, buy': It combines the meaning of temporary possession (*-š*) and a movement away from the possessor (*-(i)lli*).

(4) a. *Äħmäd-iš* *vaz* *läk'-šä-mä.*
Ahmad-POSLOC knife.ABS give-PST-DECL
'(S/he) gave a knife to Ahmad.'

b. *Äħmäd-iš-illi* *vaz*
Ahmad-POSLOC-ABL knife.ABS
t^{h}enʒuqχ-šä-mä.
buy-PST-DECL
'(S/he) bought a knife from Ahmad.'

Contact-induced double marking: While *-š-illi* in the examples above can be explained by the composition of morphological functions, younger speakers of **Khinalug** prefer to attach *-(i)lli* to *-š* whenever the construction is expressed by an ablative in Azerbaijani – apparently due to the influence of the dominant language. This can be observed for partitive, material indications, and topics of a conversation (even though the construction without ablative is still considered grammatically correct):

(5) *dä* *vaz* *ura-š(-illi)*
ABS.PROX/PHOR knife.ABS iron-POSLOC-ABL
k^{h}ui-qo-mə.
make.PST-FH/BEL-DECL
'This knife was made from iron.'

A similar combination of ablative and comparative case does not seem to be supported by existing case combinations in Khinalug, but triggered by the Azerbaijani use of the ablative for the marking of the object of comparison, since a functional difference between the comparative case marker *-q'* and the comparative-ablative case marking *-q'-illi* cannot be detected at all.

(6) *pši* *hilam-iq'(-illi)* *čiχi*
horse.ABS donkey-COMP-ABL big
qo-mä.
COP.FH/BEL-DECL
'A horse is bigger than a donkey. '

Double case marking in pronominal morphology: Patterns of multiple case marking can also be found in the **inflection of pronouns** in many languages, including well-studied European languages. The German possessive demonstrative pronoun *deren* (roughly, 'their') carries double case marking: In *mit deren Männern* 'with their men', the demonstrative expresses agreement with the head noun *Männer* (DAT;PL) as expected from German adjective (cf. *mit vielen Männern* 'with many men') and article (*mit den Männern* 'with the men') inflection. At the same time, however, *deren* is an extension of an inflected demonstrative, itself, namely from *der* (GEN;SG;F or GEN;PL) – as can be seen from its masculine/neuter counterpart *dessen* (roughly 'his' or 'its', from *des*, GEN;SG;M or GEN;SG;N). Here, the demonstrative carries double inflection: the inherent case, person and number of its antecedent, plus case, person and number of its syntactic head.

Case stacking: While in European languages, this phenomenon is restricted to function words, and therefore of limited relevance to UniMorph, multiple case marking of *full* nouns has also been documented outside the Caucasus, e.g., in Sumerian case stacking:[11]

(7) *ama* *^{D}ba-u$_2$* *e$_2$-tar-sir$_2$-sir$_2$-ta* ...
[mother Bau [E-tarsirsir]$_{ABL}$]$_{ERG}$ ...
'Mother Bau from E-tarsirsir (granted well-being to Gudea).'

Here, the last noun of a noun phrase carries the agreement information of all embedded nouns; the ergative case of *mother Bau* is thus expressed on the last noun of the noun phrase, which itself stands in ablative case, thus resulting in double case marking on *E-tarsirsir*.
Multiple case marking is not limited to two cases. Hurrian *eġli=ve=NE=ve=NA=až=(v)a* 'of the saviour', lit. 'of the one of the salvation', exhibits double genitive, augmented with dative agreement with its head (Wegner, 1995, p.144-145), and, similarly, Sumerian *bi$_3$-lu$_5$-da ud-bi-ta* 'customs of former times' is literally 'ritual of from-the-day', i.e., 'day.ABL.GEN.ABS'.[12]
Such examples can be taken as instances of **recursive inflectional morphology** (Kracht, 2003), which the current design of UniMorph as a flat, unordered set of unqualified features, however, cannot express.

Locatives Multiple case marking is discussed in the UniMorph schema only with reference to locatives: For these, Sylak-Glassman (2016, p.18) follows the analysis of Radkevich (2010, p.5), who suggests the following universal template for the arrangement of local cases:

Noun.Lemma-Stem.Extender-Place-Distal-Motion-Aspect

Non-locative cases are thus expected to occur as stem extenders, and he even mentions Nakh-Dagestanian languages as an example for the use of ergative marking at this point. This description, however, leads to the impression that multiple case marking only occurs in a constellation

[10] Other functions include to mark the addressee of the verb *li* 'say', the topic of a conversation with other *verba dicendi*, it may also function as a partitive and mark the material something is made of, and it marks the subject of abilitative predicates.

[11] http://oracc.museum.upenn.edu/etcsri/Q001547, iii.2-5

[12] http://oracc.museum.upenn.edu/etcsri/Q001124, vii.26-27

where one inherent, non-locative case is applied together with some locative marking. If that would be true, multiple cases could always be resolved unambiguously, in that any non-locative case is by definition the inherent case. However, the examples given above involve several non-locative cases or even a reversal of this pattern, so that an explicit mechanism to express multiple case marking is required.

3.2. A ranking mechanism

As a conservative extension of UniMorph, we suggest to introduce numerical indices to 'non-default' agreement features in nominal morphology. Default agreement would be the inherent case of a noun or pronoun: In a possessive construction like 'people-DAT of this king's-GEN dominion-GEN-DAT' (see the Megrelian example above), the *dominion* would thus receive GEN as a mark of its inherent case. Case features that indicate the agreement of the immediate head would then receive index 1, thus marking *dominion* with GEN;DAT-1 because of its dative agreement with *people*. To express an agreement with the head of the head, the index increases accordingly: Hurrian *eġli=ve=NE=ve=NA=až=(v)a* would thus be glossed as N;GEN;GEN-1;DAT-2, etc.

In preparation for the syntactic annotation of corpora of Classical Armenian and the example given above, we can now also distinguish between *t'agawor-i-n* (N;GEN;SG;DEF) and *t'agawor-ē-n* (N;ABL-1;SG-1;DEF), or between the latter and the same word expressing inherent case (N;ABL;SG;DEF).

In order to gloss the Khinalug double case marked forms appropriately, we suggest to indicate the first case suffix as usual, and add a number to any further following case suffix starting with 1: The tokens in question (in examples (3b), (4b), and (6)) would thus be glossed as follows: *mɯda-χ-illi* N;APUD/APPRX;**ABL-1**, *Äħmädiš-illi* PROPN;LGSPEC.POSLOC;**ABL-1**, and *hilam-iq'(-illi)* N;COMPV(;**ABL-1**).

In this way, neither the current UniMorph design with its non-consideration of order needs to be abandoned nor do we lose information anymore.

This allows to keep existing annotation for nouns intact, as these are currently annotated for their inherent case only. This approach also aligns very well with the impression that multiple case marking is somewhat exceptional, so that 'basic' annotations focus on inherent case.

4. On verbal inflection

In the example in Fig. 1, the Megrelian verb shows head marking for both its syntactic arguments, a first person subject and a second person object. UniMorph allows us to distinguish both clearly, by forming compound features of argument case and argument features, e.g., ARGNO1S for the nominative subject as being first person singular, and ARGDA2S for the dative object as being second person singular (Sylak-Glassman, 2016, p. 13). We discuss problems of argument identification by morphological case for the example of Kartvelian, and suggest an alternative.

4.1. Complex patterns of argument marking

As Fig. 1 shows, verbs in Kartvelian languages can specify grammatical features for *multiple* arguments, so that the agreement information about one (e.g., subject) argument must be clearly distinguished from agreement information about another (e.g., object). This is not specific to Caucasian languages, but does also occur, for example, in Basque, and has been addressed in UniMorph before. The UniMorph solution to the problem is to form compound features, which has the drawback that the UniMorph schema is partially redundant; the following statements are equivalent:

```
has     have     V;3;SG;IND
has     have     V;ARGNO3SG;IND
```

In practice, this problem does not occur, because ARG features are provided for languages where verbs are marked for their arguments. Nevertheless, UniMorph was designed with the intention to project morphological annotations between languages (Yarowsky et al., 2001; Sylak-Glassman et al., 2015a). It is not clear, however, to what extent compound features such as `ARGNO3SG` can be put in *any* relation with the 'regular' features `3;SG` unless additional (language-specific!) assumptions about case morphology and its relation with subjecthood are taken into consideration.

In particular, this is a problem for Kartvelian. It is an established convention for most languages to identify arguments in terms of their grammatical roles ('subject' and 'object'). In Georgian (as well as in Megrelian and other Kartvelian languages), however, the linking between grammatical roles and morphological cases is relatively complex, and the same role can by expressed by different cases, depending on the tense/aspect of the verb, while at the same time, this argument can be marked on the verb by a single element, regardless of the case of the (pro)nominal it refers to.

(8) a. *bavšv-eb-s* *da-∅-malav-s*
child-PL-DAT PV-O3-hide-S3SG

'S/he will hide the children.'

b. *bavšv-eb-i* *da-∅-mala*
child-PL-NOM PV-O3-hide.AOR.S3SG

'S/he hid the children.'

Hence, in the Georgian sentences (8a) and (8b), the 3rd person object is marked by the absence of a prefix (i.e. the 1st and 2nd persons would have been marked). In the future tense, the object constituent receives the dative case, while in the aorist tense it receives the nominative case. Referring to grammatical roles here would be less confusing and more in line with established research.

When populating UniMorph inventories from existing glossed corpora, however, we need to keep in mind that tense may be indicated by *multiple* morphemes which do not have a clear interpretation in a morpheme-based annotation (as for the Megrelian example in Fig. 1). Accordingly, only a language expert can implement a direct mapping between existing morpheme-based annotations (which

refer to grammatical roles) and cases of the corresponding arguments (whose identification depends on identifying the tense feature) – which may create a barrier for creating UniMorph resources.[13]

4.2. A ranking, again

In extension of the ranking-based modeling of multiple case marking, it is possible to generalize over both the case-based and the grammatical-role-based encoding of arguments as well as over compound and regular features for arguments in different languages.

At least since Dowty (1991) and Grosz et al. (1995), the importance of aligned hierarchies of grammatical and semantic roles has been recognized in different communicative functions, and established as such in both linguistics and natural language processing: According to Dowty, a ranking of semantic roles (AGENT > PATIENT > ...) is underlying the assignment of grammatical roles; according to Grosz et al., a ranking of grammatical (or semantic) roles is taken to reflect and to establish discourse salience (Subject > Object > ...) which is closely tied with pronominalization. By extension of this approach, highly salient discourse referents can be expressed by $\emptyset$ pronouns or by verbal inflection, alone, thereby establishing a grammaticalization path from pronouns to verbal inflection (Ariel, 2000). In summary, a ranking of grammatical (semantic) roles is almost universally upheld, and a close relation with verbal morphology is assumed, at least.

Following the UniMorph approach to render grammatical roles with morphological cases, it would thus seem possible to provide a language-specific ranking of morphological cases that represent these cases – or the underlying grammatical roles. Such a ranking, now, can be expressed by numerical indices, as well, with the top-ranked element being assigned the empty index. With a ranking of nominative (subject) over dative (object) over other cases, we can thus develop an alternative representation of Megrelian *kešerxvaduk*, i.e.,

V;...;1;SG;2-1;SG-1
instead of V;...;ARGNO1S;ARGDA2S

This approach elegantly overcomes the asymmetry between compound and individual features, it establishes a principled approach to deal with the assignment of multiple instances of the same feature to (different arguments/heads of) a single word in both the nominal and the verbal domain, and it can be formulated without a priori restrictions to certain grammatical features. UniMorph will thus gain in scalability. Moreover, it eliminates alternative encoding strategies for the same phenomenon, and it even facilitates comparability across languages , as the Megrelian features now overlap with, say, those of its Russian translation *vstrechu*: V;...;1;SG (assuming nominative > other cases). Again, this extension has little impact on most existing UniMorph resources, as ARG features are used for Basque only, so far. For other languages, one would formally need to define a ranking, but default rankings can be posited, too. For accusative languages, the corresponding default ranking would be

nominative > accusative > other

For languages with ergative alignment, the default ranking would be

ergative > absolutive/nominative > dative > other

This hierarchy is suggested to reflect the fact that the ergative case is almost exclusively used for subjects, constituents marked with nominative/absolutive can by subjects or direct objects, and the dative case can be used to mark subjects and both direct and indirect objects. Thus, the hierarchies for both accussative and ergative languages correspond to the hierarchy for grammatical roles:

subject > object > other

5. On the UniMorph format

So far, we discussed possible extensions of the current UniMorph schema that arise from our work on Caucasian and other low-resource languages. However, also the UniMorph file format may represent a hurdle for its application beyond NLP. Although its minimalistic design establishes a high level of interoperability, it seriously limits the usability of UniMorph data sets for linguistic research – and their acceptability for linguists. Therefore, we suggest a workflow to derive the current TSV format from more expressive formalisms that are closer to current practices in language contact studies, language documentation and linguistic research in general.

5.1. Beyond Tab-Separated Values

In a field of research where Interlinear Glossed Text and an elaborate toolchain for its generation and processing (including Toolbox[14] and FLEx[15]) is the state of the art, converting carefully constructed, high-quality morpheme inventories to an incomplete and less interpretable representation poses a problem regarding acceptability and dissemination. The UniMorph format must not be understood as a full-fledged representation formalism, but rather as an interchange format between rich and high-quality language resources on the one hand and morphological generators on the other hand, as developed, e.g., in the context of the SIGMORPHON shared tasks. Moreover, the UniMorph repositories are very likely to get out of sync with the underlying resource, as they are maintained in Github repositories structured according to the same conventions: This means that the source data and its UniMorph extract are maintained at different locations. This maintenance aspect is not that complicated for high-resource languages, as their morphological description is unlikely to evolve greatly in the immediate future. In the context of low-resource languages, however, efforts in language documentation frequently lead

[13] We would like to point out that the current Georgian UniMorph data does *not* include compound features for multiple argument marking, probably for precisely this reason. Furthermore, the Basque UniMorph data (that *does* employ compound features) deviates from the schema by partly using grammatical role labeling (instead of case labeling).

[14] https://software.sil.org/fieldworks/

[15] https://software.sil.org/toolbox/

to novel insights into the inventory and the function of inflectional morphemes in a language.

As an alternative to the current UniMorph publication model, we propose a formalism and a workflow that allows to embed UniMorph linkings into existing resources, in particular, if these are provided in XML, CSV, JSON, or RDF. The current TSV format can then be retrieved from various types of source data, and the UniMorph repositories can be populated with morpheme inventories in their native representation, avoiding information loss and forks between different versions of the same resource.

```
:sl_3052 a ontolex:LexicalEntry;                  # 1: explicit data structures
    ontolex:canonicalForm
    [ ontolex:writtenRep "xvad" ];                # 2: lemma
    ontolex:otherForm :sl_3052_15.                # 3: link with form(s)
:sl_3052_15 a ontolex:Form;                       # 4: explicit data structures
    ontolex:writtenRep "kešerxvaduk"@xmf;         # 5: word form
    unimorph:hasFeature unimorph:V, unimorph:SG.  # 6: ontology linking, samples
```

Figure 2: Megrelian lemon/RDF sample in Turtle

5.2. Linking UniMorph

Our solution builds on modelling language resources, resp. the linking between them, on the basis of **Linked Data** formalisms. The Linked Data paradigm (Berners-Lee, 2006) postulates rules for the publication and representation of Web resourceswhich facilitate information integration, and thus, interoperability. Data should be represented by means of W3C standards, such as RDF (Resource Description Framework). RDF provides a generic data model based on labeled directed graphs, which can be serialized in different formats. Information is represented by *triples* which consist of a *predicate* (a relation, i.e., a labeled edge) that connects a *subject* (a so-called "RDF resource", i.e., a labeled node) with its *object* (another RDF resource, or a literal, e.g., a string). The RDF resources are represented by URIs, making them unambiguous in the web of data, allowing resources hosted at different locations to refer to each other and thus creating a network of data collections with densely interwoven elements (Chiarcos et al., 2013).

Linked Data has been successfully applied to convert and link language resources (Chiarcos et al., 2012), leading to the emergence of the Linguistic Linked Open Data (LLOD) cloud (Chiarcos et al., 2013; McCrae et al., 2016),[16] a set of linked open language resources for all fields of linguistics, digital philology, natural language processing, the localization industry and the Semantic Web community, tied together by shared vocabularies, the use of reference knowledge bases and links between each other. As such, the lemon/ontolex vocabulary (Cimiano et al., 2016) developed as a community standard for machine-readable dictionaries in the cloud, and its extension to morpheme inventories is currently being discussed. In the context of UniMorph and language documentation, recent proposals to develop vocabularies for Interlinear Glossed Text (Chiarcos et al., 2017) and TSV-based corpus formats (Chiarcos and Fäth, 2017, CoNLL-RDF) are to be mentioned. On the basis of CoNLL-RDF, we developed a tool for the LLOD conversion of the UniMorph format as part of our LLODifier library[17]. We envision a future infrastructure for UniMorph where different source formats are mediated by RDF representations and associated SPARQL scripts. These scripts can then be used to derive the TSV format as currently in use, or – alternatively – can be digested directly (and losslessly) by downstream applications. As an example, we developed converters from FLEx to FLEx RDF (applied to Megrelian), from CoNLL to CoNLL-RDF (applied to Classical Armenian), from ELAN[18] to ELAN RDF (independently from Caucasus studies applied to Old High German), and from TSV to lemon/RDF (applicable to every existing UniMorph dataset).

A key benefit of representing language resources in RDF is that individual items within a resource are identifiable by means of a URI so statements about them can be added easily, e.g. explicit links with an ontology. We provide an OWL2/DL formalization of the UniMorph schema,[19] designed as an Annotation Model in the OLiA architecture (Chiarcos and Sukhareva, 2015). It is thus possible to provide declarative links between individual items in a morphological inventory and the UniMorph ontology:[20] Fig. 2 shows a fragment from the lemon edition of the Megrelian UniMorph inventory, with the original UniMorph features transposed into explicit links to well-defined entities in the web of data (where data consumers can look up the definition, relation to other features, etc.).[21]

5.3. Beyond RDF

RDF technology does, however, not require the source data to be RDF. RDFa (Adida et al., 2015), for example, permits to add typed links to XML documents, which can then be parsed into other RDF serializations. Alternatively, explicit RDF conversion instructions can be attached to XML documents using GRDDL (Connolly, 2007). Similarly, tabular data (as in the current UniMorph format) does not require an explicit conversion: CSV2RDF (Tandy et al., 2015) is a W3C recommendation that allows the *direct* interpretation of tabular data as RDF – and thus enables its linking with, say, the UniMorph ontology. For other formats designated converters are provided, for example, as part of the LLODifier library.

Such conversions from various source formats merely require (a) an indication of their original format (TSV, RDF, XML – for W3C-supported formats), resp., the converter (for other formats), and (b) one SPARQL Update script per source format to guarantee conformancy with common specifications, e.g., to resolve feature abbreviations into links against the UniMorph ontology. The latter, however,

[16] http://linguistic-lod.org/

[17] https://github.com/acoli-repo/LLODifier/tree/master/unimorph.

[18] https://tla.mpi.nl/tools/tla-tools/elan/

[19] http://purl.org/olia/owl/experimental/unimorph

[20] Such links can be auto-generated from abbreviations, cf. https://github.com/acoli-repo/LLODifier/blob/master/unimorph/link-and-load-FEATS.sparql.

[21] This linking mechanism can also be used to map from an existing annotation scheme into UniMorph, as currently implemented, e.g., from the PROIEL schema to Classical Armenian.

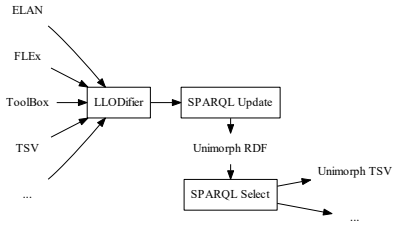


Figure 3: Suggested pipeline for converting data to UniMorph RDF

are optional, and invariant for each type of source format.

5.4. Back to TSV

With RDF data or an RDF interpretation of source data in place, a TSV file can then be automatically generated using a SPARQL SELECT statement, e.g., from a lemon RDF file:

```
SELECT ?word ?lemma ?feats
WHERE {
   ?form a ontolex:Form; writtenRep ?word.
   ?lexEnt ontolex:otherForm ?form;
           ontolex:canonicalForm/ontolex:writtenRep
           ?lemma.
    {
    SELECT ?word (GROUP_CONCAT(?feat; separator=";")
    AS ?feats)
    WHERE {
        ?word unimorph:hasFeature/unimorph:hasLabel
        ?feat
    } GROUP BY ?word
    }
}
```

The output format of this query is a table and its TSV serialization can be directly fed into existing UniMorph-based tools.

The use of RDF for data conversion and SPARQL for data transformation and querying thus facilitates the development of a technical infrastructure for UniMorph which allows the community to grow beyond the limitations imposed by the crippling TSV format it is currently based on, an achievement which would be most welcome to linguists, researchers and NLP engineers working on low-resource languages.

6. Summary and outlook

We describe the creation of UniMorph resources for languages in the Caucasus region, including Megrelian, Khinalug, and Classical Armenian, which are published under an open license via our UniMorph fork[22] and which are to be integrated with the main UniMorph infrastructure (in case our suggested modifications meet community approval), thereby increasing the coverage of languages from the Caucasus area in UniMorph.

We discussed a number of peculiarities of these languages and potential conceptual difficulties in the application of the UniMorph scheme to them and other languages. As a result, we suggest the following extensions to the UniMorph schema:

- human-readable labels for LGSPEC features, e.g., LGSPEC-TM instead of LGSPEC4 for Megrelian,
- a ranking-based numerical scheme to represent iterative features in nominal inflection,
- a ranking-based numerical scheme to encode multiple arguments of polyvalent verbs in head-marking languages, and
- the postulation of a default ranking for verbal arguments, as well as the possibility to posit language-specific rankings.

In addition, we discuss the UniMorph TSV format and criticize its limited expressiveness which creates a gap between its uses in NLP and potential users of UniMorph technology or providers of UniMorph data in linguistics. We thus suggest to complement the existing UniMorph workflow with support for additional source formats on grounds of Linked Open Data technology. For this purpose, we provide converters for UniMorph TSV, FLEx, ELAN and other formats to RDF,[23] a SPARQL query for the generation of UniMorph TSV out of RDF and an RDF/OWL edition of the UniMorph schema that we provide as part of the Ontologies of Linguistic Annotation.[24]

The combination of these resources allows us to derive Unimorph TSV files from various source formats, and our UniMorph fork provides not only TSV files, but also Makefiles and associated resources. For the future, however, one may consider to follow a streamlined approach and develop a uniform UniMorph representation in RDF, which can be derived from resource-specific RDF representations and mediate between these and the current UniMorph TSV representation as illustrated in Fig. 3.

One key advantage of a future RDF vocabulary of UniMorph data in comparison to TSV data would be that additional data can be added as needed, without affecting its processability. In particular, it may preserve *any* information from the original, resource-specific RDF – just cleanly separated in a distinct namespace. Such a UniMorph vocabulary could build, for example, on existing community standards such as lemon (Cimiano et al., 2016), as illustrated in Fig. 2.

7. Acknowledgements

This research was conducted in the context of the research group 'Linked Open Dictionaries' (LiODi, 2015-2020, Goethe University Frankfurt, Germany). LiODi is funded by the Federal Ministry of Education and Research and its activities are centered around developing innovative methodologies for utilizing dictionaries and corpora for language contact studies, with a particular focus on applications of Linguistic Linked Open Data. LiODi consists of researchers from applied computational linguistics and empirical linguistics, with a focus on language contact in the Caucasus area.

8. Bibliographical references

Adida, B., Birbeck, M., McCarron, S., and Herman, I. (2015). RDFa Core 1.1 - Third Edition. Syntax and

[22] https://github.com/acoli-repo/unimorph

[23] https://github.com/acoli-repo/LLODifier

[24] http://purl.org/olia/owl/experimental/unimorph

processing rules for embedding RDF through attributes. Technical report, W3C Recommendation.

Ariel, M. (2000). The Development of Person Agreement Markers: from Pronouns to Higher Accessibility Markers. In Michael Barlow et al., editors, *Usage-based models of language*, pages 197–260. CSIL Publications, Stanford, CA.

Berners-Lee, T. (2006). Design issues: Linked data. Technical report.

Boeder, W. (1995). Suffixaufnahme in Kartvelian. In Frans Plank, editor, *Double Case: Agreement by Suffixaufnahme*, pages 151–215. Oxford University Press, Oxford.

Chiarcos, C. and Fäth, C. (2017). CoNLL-RDF: Linked Corpora Done in an NLP-Friendly Way. In Jorge Gracia, et al., editors, *Language, Data, and Knowledge*, pages 74–88, Cham. Springer.

Chiarcos, C. and Sukhareva, M. (2015). OLiA – Ontologies of Linguistic Annotation. *Semantic Web*, 6(4):379–386.

Christian Chiarcos, et al., editors. (2012). *Linked Data in Linguistics - Representing and Connecting Language Data and Language Metadata*. Springer, Berlin, Heidelberg.

Chiarcos, C., Moran, S., Mendes, P. N., Nordhoff, S., and Littauer, R. (2013). Building a Linked Open Data cloud of linguistic resources: Motivations and developments. In *The People's Web Meets NLP. Theory and Applications of Natural Language Processing*, pages 315–348. Springer, Berlin, Heidelberg.

Chiarcos, C., Ionov, M., Rind-Pawlowski, M., Fäth, C., Wichers Schreur, J., and Nevskaya, I. (2017). LLODifying Linguistic Glosses. In *International Conference on Language, Data and Knowledge*, pages 89–103, Cham. Springer.

Cimiano, P., McCrae, J., and Buitelaar, P. (2016). Lexicon Model for Ontologies. Technical report, W3C Community Report.

Connolly, D. (2007). Gleaning Resource Descriptions from Dialects of Languages (GRDDL). Technical report, W3C Recommendation.

Dowty, D. (1991). Thematic Proto-Roles and Argument Selection. *Language*, 67(3):547–619.

Grosz, B. J., Weinstein, S., and Joshi, A. K. (1995). Centering: A Framework for Modeling the Local Coherence of Discourse. *Computational linguistics*, 21(2):203–225.

Haspelmath, M. (2007). Pre-established categories don't exist: Consequences for language description and typology. *Linguistic Typology*, 11(1):119–132.

Hübschmann, H. (1906). Armeniaca. *Indogermanische Forschungen*, 19(1):457–480.

Kirov, C., Sylak-Glassman, J., Que, R., and Yarowsky, D. (2016). Very-large Scale Parsing and Normalization of Wiktionary Morphological Paradigms. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris. European Language Resources Association (ELRA).

Klimov, G. A. (1994). *Einführung in die kaukasische Sprachwissenschaft*. Buske Helmut Verlag GmbH.

Kracht, M. (2003). Against the feature bundle theory of case. In Ellen Brandner et al., editors, *New Perspectives on Case Theory*, pages 165–190. CSLI Publications, Stanford, CA.

McCrae, J. P., Chiarcos, C., Bond, F., Cimiano, P., Declerck, T., de Melo, G., Gracia, J., Hellmann, S., Klimek, B., Moran, S., et al. (2016). The Open Linguistics Working Group: Developing the Linguistic Linked Open Data Cloud. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2435–2441, Paris. European Language Resources Association (ELRA).

Nivre, J. et al. (2015). *Universal Dependencies 1.2*. LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague.

Plank, F. (2014). *Double case: Agreement by Suffixaufnahme*. Oxford University Press.

Radkevich, N. (2010). On Location: The Structure of Case and Adpositions. Ph.D. thesis, University of Connecticut, Storrs, CT.

Sylak-Glassman, J., Kirov, C., Post, M., Que, R., and Yarowsky, D. (2015a). A Universal Feature Schema for Rich Morphological Annotation and Fine-Grained Cross-Lingual Part-of-Speech Tagging. In Cerstin Mahlow et al., editors, *Systems and Frameworks for Computational Morphology*, pages 72–93, Cham. Springer.

Sylak-Glassman, J., Kirov, C., Yarowsky, D., and Que, R. (2015b). A Language-Independent Feature Schema for Inflectional Morphology. In Chengqing Zong et al., editors, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 674–680, Beijing. Association for Computational Linguistics.

Sylak-Glassman, J. (2016). The Composition and Use of the Universal Morphological Feature Schema (Unimorph Schema). Technical report, Department of Computer Science, Johns Hopkins University. working draft, v.2.

Tandy, J., Herman, I., and Kellogg, G. (2015). Generating RDF from Tabular Data on the Web. Technical report, W3C Recommendation.

Wegner, I. (1995). Suffixaufnahme in Hurrian: Normal Cases and Special Cases. In Frans Plank, editor, *Double Case: Agreement by Suffixaufnahme*, page 136–147. Oxford University Press, Oxford.

Yarowsky, D., Ngai, G., and Wicentowski, R. (2001). Inducing Multilingual Text Analysis Tools via Robust Projection across Aligned Corpora. In *Proceedings of the First International Conference on Human Language Technology (HTL)*, pages 1–8, Stroudsburg, PA. Association for Computational Linguistics.

9. Language Resource References

Haug, D. and Jøhndal, M. (2008). Creating a Parallel Treebank of the Old Indo-European Bible Translations.

In Caroline Sporleder et al., editors, *Proceedings of the Second Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2008)*. Marrakech.

EMTC: Multilabel Corpus in Movie Domain for Emotion Analysis in Conversational Text

Phan Duc-Anh and Yuji Matsumoto
Computational Linguistics Laboratory, Graduate school of Information Science
Nara Institute of Science and Technology, Japan
{phan.duc_anh.oq3, matsu}@is.naist.jp

Abstract

It is proved that in text-based communication such as sms, messengers applications, misinterpretation of partner's emotions are pretty common. In order to tackle this problem, we propose a new multilabel corpus named Emotional Movie Transcript Corpus (EMTC). Unlike most of the existing emotion corpora that are collected from Twitters and use hashtags labels, our corpus includes conversations from movie with more than 2.1 millions utterances which are partly annotated by ourselves and independent annotators. To our intuition, conversations from movies are closer to real-life settings and emotionally richer. We believe that a corpus like EMTC will greatly benefit the development and evaluation of emotion analysis systems and improve their ability to express and interpret emotions in text-based communication.

Keywords: emotion corpus, movie transcript, text-based communication, emotion analysis, multilabel

1. Introduction

In the recent years, we experience rapid development of online communication with the help of mediated devices such as smart phones, tablets, computer. People use talk-over-internet, video conferencing features for both daily and business tasks. Text based methods like emails or text messengers are still very convenient and indispensable to us because of its unique advantages: they do not require intermediate responses and can be used for the sake of record-keeping. However, it is already proved that in any online communication methods, users experience more difficulties in interpreting and conveying emotions than face-to-face communication due to the limitation in communication modality. Furthermore, text-based methods are where difficulties are encountered the most (Kruger et al., 2005; Arimoto and Okanoya, 2016). Therefore, we targeted the effort to build an emotion analysis system focusing on text data. The starting point is to develop an emotional corpus that has conversational texts and is as close to real-life communication as possible.

Existing emotional text corpora are often collected from micro-blog platforms using multiclass scheme - *one emotion per example* (Liew et al., 2016). Most of them are automatically annotated by extracting hashtags rather than by human judgements (Dini and Bittar, 2016; Li et al., 2016). While the text data from micro-blog platforms like Twitters are very convenient and easy to collect, the fact that they are limited in the number of characters (140 for a tweet) differs themselves from daily conversation text and therefore, have limited use in a real-life settings. On the other hand, multiclass scheme has its own limitation. One input is only associated to one emotion. However, it is observed in some research (Liew et al., 2016) that multilabel scheme *with no limitation in the number of emotions per example* is a better and more natural way of annotating emotion labels.

We describe our efforts to construct and annotate partly the Emotional Movie Transcript Corpus (EMTC). Most of the corpus are unsupervised data. We annotated by ourselves 10,000 utterances and use them for training. Finally, the testing data, which include 1000 utterances, are annotated by 5 independent annotators. To our understanding, EMTC is the only emotional corpus that is annotated using multilabel scheme and has conversational text instead of short text like tweets or news headlines (Strapparava and Mihalcea, 2007; Mohammad, 2012b). Moreover, EMTC provide the annotators with movie clips instead of just text to help them give better annotation. Our contributions are summarized as follow:

- We explain the multilabel annotating scheme following Plutchik's theory of emotions (Plutchik, 2001). We then later conclude that our annotating scheme provide much better inter-annotators agreement score than other corpora.
- We present and describe the characteristics of the conversational corpus and the statistics of the annotated data.
- We conduct supervised machine learning experiments to evaluate the emotion classification using our corpus and the word-embedding extracted from it.

2. Related Works

There have been numerous works on building emotion corpora. The first notable work is the ISEAR dataset (Scherer and Wallbott, 1994) which has more than 7,000 responses from participants. The participants are asked to describe the situation where they experience some certain emotions. In our work, we use this dataset to extract collocation features for the manual feature extration step described in the below section. Another corpus is the Semeval-2007 task 14: Affective text (Strapparava and Mihalcea, 2007) which consists of 1,250 news headlines with six Ekman's emotion labels. More recent works are (Mohammad, 2012b; Liew et al., 2016; Dini and Bittar, 2016) where they collect data from micro-blog platforms and automatically annotate

them using hashtags with or without human revision afterwards. The limitation with those corpora is that they only consist of short, independent pieces of text and undoubtedly not close to real-life conversation.

As a matter of fact, modeling emotions in a conversation is indeed a difficult but rewarding task with a wide range of applications. A good system should consider every word in the conversation, the grammatical structure and syntactic variables such as negations, embedded sentences, and type of sentence (question, exclamation, command, or statement), the general context of the conversation, each and every utterances in the conversation - especially when what is said in the previous utterance can have an impact on the emotions of the later one (Collier, 2014). Maybe, because of this complicated nature of the problem, there is a lack of emotional conversation corpus.

Another problem with the existing corpora is the annotating scheme: many works limit the emotion labels to a small number (Mohammad, 2012b; Wang et al., 2015) or only allow annotators to label one emotion per utterance (Yang et al., 2007; Hasegawa et al., 2013). As pointed out in many psychology research (Plutchik, 2001; Russell, 2003), emotions are not mutually exclusive. In fact, in many cases, people may experience a mixture of various emotions at the same time (Choe et al., 2013). Therefore, the corpus for any emotion analysis task should be multilabel. Limiting the number of emotion labels may narrow down the problem but can cause troubles for the annotators to provide correct judgement when the emotions in an example are sophisticated or expressed implicitly.

In our work, we employ Plutchik's theory of emotions and extend the set of labels to a total of 48 labels to provide more freedom to the annotators. The extension and Plutchik's theory will be explained in more details in Section 3., where we present the construction of the corpus. Section 4. investigates the characteristics of our newly built corpus and Section 5. discusses the experiments and evaluation of the corpus. Lastly, Section 6. gives conclusions and future work.

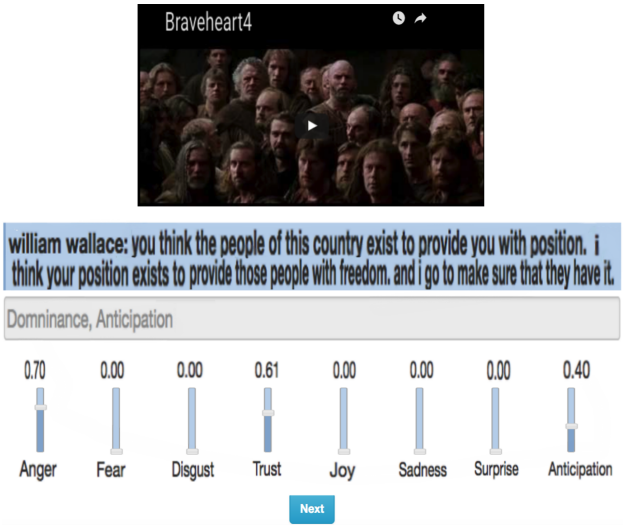
3. Methodology

3.1. Imdb quotes dataset

In order to mimic real-life conversation settings, we rely on the Imdb datasets [1], in particularly, the movie quotes dataset. This dataset includes in total 2,107,863 utterances (turns in conversation) out of 117,425 movies and tv series of all genres such as: thrillers, action, romantic, etc. To our assumption, movies conversation should be close to real-life settings and emotionally rich. We can also easily eliminate the low inter-annotators agreement score problem that is often encountered in other corpus (Strapparava and Mihalcea, 2007; Dini and Bittar, 2016) by providing them the clips from the movies in addition to the transcripts (Figure 1a).

At first, we would also want to measure the judgement of emotion intensity from the annotators, hence the bar measurement. However, the collected numbers are very diverse and unreliable. This is due to disagreement among the annotators and their different interpretation of the emotion intensity during the annotation sessions.

There is also a concern that different culture will give different interpretation of emotion expressions in those provided clips. However, nowadays, as people from different cultures are more exposed to and have more opportunity to watch American movies, they steadily learn how to interpret the emotions from other cultures better (Hareli et al., 2015; Lim, 2016).



(a) UI of the annotating website. Users can choose the appropriate emotions by adjusting the confidence bars or by typing the emotions or dyads into the text box. The dyads are then decomposed automatically into primary emotions and the bars are readjusted

robert the bruce: my hate will die with you.	{"Anger":"0.37","Disgust":"0.40"}
princess isabelle: the king desires peace.	{"Trust":"0.22"}
william wallace: longshanks desires peace?	{"Anger":"0.26","Disgust":"0.34","Surprise":"0.36"...
william wallace: go back to england and tell them ...	{"Anger":"0.52","Disgust":"0.38","Trust":"0.34"}

(b) Examples of annotated transcripts from movie: Brave Heart (1995) -

Figure 1: Annotating scheme of the testing data. Each utterance is annotated with primary emotions

3.2. Plutchik's theory of emotions

The reason for most research to limit the number of emotion categories is to have a better inter-annotators agreement score. The more categories are allowed, the lower the score it becomes. However, by limiting the number of categories, they also limit the freedom of the annotators to give accurate judgements because emotions are sophisticated and the basic emotions can hardly cover all the cases. In our work, we found out a way to avoid this trade-off: Plutchik's theory of emotions.

According to Plutchik, there are eight primary emotions grouped on a positive or negative basis: joy versus sadness; anger versus fear; trust versus disgust; and surprise versus anticipation. Some emotions are similar to the primary ones but different in intensity (Table 1). Some pri-

[1] The datasets are available from http://www.imdb.com/interfaces

mary emotions can be mixed to form more complex emotions 2. Implementing the theory, we allow the annotators to use the full 48 emotion labels from the two tables, the system will automatically decompose the annotated labels into primary emotions later (Figure 1).

Intense emotion	Ecstasy	Admiration	Terror	Amazement
Mild emotion	Serenity	Acceptance	Apprehension	Distraction
Primary emotion	Joy	Trust	Fear	Surprise
Primary opposite	Sadness	Disgust	Anger	Anticipation
Mild opposite	Pensiveness	Boredom	Annoyance	Interest
Intense opposite	Grief	Loathing	Rage	Vigilance

Table 1: Emotions and the opposite of them

Human feelings	**Parent Emotions**	**Opposite feelings**	**Parent Emotions**
Optimism	Anticipation, Joy	Disapproval	Surprise, Sadness
Hope	Anticipation, Trust	Unbelief	Surprise, Disgust
Anxiety	Anticipation, Fear	Outrage	Surprise, Anger
Love	Joy, Trust	Remorse	Sadness, Disgust
Guilt	Joy, Fear	Envy	Sadness, Anger
Delight	Joy, Surprise	Pessimism	Sadness, Anticipation
Submission	Trust, Fear	Contempt	Disgust, Anger
Curiosity	Trust, Surprise	Cynicism	Disgust, Anticipation
Sentimentality	Trust, Sadness	Morbidness	Disgust, Joy
Awe	Fear, Surprise	Aggressiveness	Anger, Anticipation
Despair	Fear, Sadness	Pride	Anger, Joy
Shame	Fear, Disgust	Dominance	Anger, Trust

Table 2: Dyads - Combinations of emotions: two primary emotions can blend together to form another complex one

3.3. Annotation Scheme

To produce labeled data, annotators are asked to watch the corresponding movies with subtitles (Figure 1a) and follow the annotation scheme shown below:

- One utterance may hold zero, one or more emotions at the same time. In case an utterance holds no emotion, it should be annotated with "None." The intensity of emotions is also considered in the labeling phrase (Figure 1).
- The annotators can choose appropriate emotion labels from the list of 48 emotions in tables 1 and 2. The system will decompose the dyads into primary emotions automatically.
- The annotators need to assign the whole utterance which may have two or more sentences with a set of all emotions expressed inside it. There may be cases where conflict emotions according to Plutchik's theory to appear simultaneously in the same utterance as in the last example of the subfigure 1b.

4. Characteristics, the inter-annotators agreement and the word-embedding of the corpus

This corpus includes in total 2,107,863 utterances with 26 millions words, 181,276 of which are unique terms. As mentioned in the above sections, we can only annotate the corpus partly. There are 10,000 utterances that are annotated by the authors ourselves as the training data. The average emotion labels per utterances are 1.68. The testing data are reviewed by 5 independent annotators to form a gold standard data (with majority rule) with 1,000 utterances. The reason for two different datasets was because the annotation sessions are expensive and time consuming. We have to provide the annotators with clips cutting from real movies and match the text from the corpus to the correct scenes in the full movies.

The average labels per utterances are 1.41. We report the inter-annotators agreement score of our testing data in the Table 3 where the performance of each annotator is compared to the gold standard data as ground-truth.

Emotion class	**Accuracy**
Anger	0.72
Fear	0.673
Disgust	0.624
Trust	0.65
Joy	0.606
Sadness	0.584
Surprise	0.575
Anticipation	0.491
Average accuracy (by class)	0.615
Average accuracy (by annotator)	0.43
Average F1 (by annotator)	0.626
Total No. utterances	1,000

Table 3: Inter-annotator Agreement score with gold standard data as ground-truth.

From on the table, it can be concluded that: Our corpus, even when being annotated using multilabel scheme, yields better agreement score to the multiclass - Twitter Emotion Corpus (Mohammad, 2012b) (Average F1-score is 43.7).

We believe that our choice of using a movie corpus and providing movie clips to support the annotation process plays an important factor here.

4.1. Word-embedding of the corpus

Word-embedding is the vector multi-dimensional representations of every words in the corpus. It can be a simple yet effective input features for many machine learning methods. In this research, we follow this approach and create the embedding with 100 dimensions using word2vec (Řehůřek and Sojka, 2010). Table 4 shows the top 5 most similar terms to the primary emotion words and some of the dyads.

Interesting points can be observed from the table: 1) We notice some of the dyads appear in the top similar list of the parent emotions, which - to some extend - validate Plutchik's theory. 2) Most of the top similar terms are quite on point and reasonable. 3) Opposite emotions and dyads sometime appear together, it maybe interesting to investigate the correlation among labels to see if the same phenomenon occurs.

Emotions	Top similar
Anger	Rage, Pain, Hatred, Guilt, Grief
Fear	Hatred, Darkness, Despair, Grief, Desire
Trust	Betray, Confuse, Respect, Underestimate, Threaten
Disgust	Horrified, Grunts, Sobs, Laughs, Startled
Joy	Beautiful, Eternal, Passion, Happiness, Sadness
Sadness	Sorrow, Loneliness, Emptiness, Despair, Joy
Surprise	Invitation, Party, Disappointed, Gifts, Shock
Anticipation	Exhaustion, Horror, Discomfort, Unending, Awakening
Love	Hate, Wonderful, Sweet, Beautiful, Charming
Curiosity	Beliefs, Ignorance, Guilt, Memories, Thrive
Aggressiveness	Unstable, Inferior, Increasing, Dominant, Destructive
Pride	Wealth, Dignity, Wisdom, Courage, Freedom

Table 4: Top similar words to primary emotions and some dyads

5. Experiments and Evaluation

5.1. Evaluation of extracted word-embedding

In order to test the practicality of our extracted word-embedding, we run an experiment on our corpus comparing two approach: One uses manual feature selection and Wordnet-Affect (Strapparava et al., 2004) and another uses the word-embedding for automatic feature extractions.

5.1.1. Feature selection approach

Most research agree that emotion words and phrases are the most obvious clue to identify emotions (Mohammad, 2012a; Strapparava and Mihalcea, 2007). Human have developed language to fit their needs of expressing ideas and feelings. Therefore, when describing our emotion, we tend to use some specific words. By picking up on these words, we have a general idea about the emotional direction of the examined text. In this approach, our first set of features is the basic emotion tendency: to express how an input relates to the 8 basic emotions. Wordnet-Affect is employed to interpret the emotion tendency of each and every words in an utterance. If one emotion exists in one word of the input then the corresponding tendency feature will be set to 1 and 0 otherwise.

However, solely relying on text would cause problem for emotion detection system. We should also consider the effect of negation words and phrases. Simply by putting a negation word, we reverse the emotion state of the text. The sentence: *You are not bad at all!* indicate a strong feeling of approval instead of the usual negative feelings from the word *bad*. Moreover, the context of the input also provide valuable information, especially in conversations. Therefore, we then define the second set of features which includes all similar traits. In the end, we have a list of manual selected features as follow:

1. The sum vector of the current input which suggest the *local tendency*.
2. The sum vector of all the utterances in the lexicon that appear in the conversation which provides the *context of the conversation*.
3. The sum vector of the previous utterance in the conversation which also provides the *context of previous exchange* (of what triggered the current emotion).
4. The *polarity* (negative/ positive) score of the sentence.
5. *Features* such as: length, is_it_a_question, is_it_an_exclamatory_sentence, is_there_negation_word.
6. Colocation features: we mine the ISEAR () dataset for phrases that are often appear in a specific emotional situation. If the input include these phrases, we set the binary flag of the corresponding features to 1.

The structure of the network is shown in figure 2: input layer of manually selected features, two hidden layers, a threshold multi-label output layer.

5.1.2. Word-embedding Network: text to vector

We consider a bag-of-features approach to transform the raw input text into vectors form. Therefore, for a piece of text, its representation is the sum vector of all lexical items inside. Because our goal is to predict the emotional labels for each utterance in a conversation, we also have to vectorize the previous utterance and the entire conversation to capture the contextual information . As a result, the vector representation of an utterance is a 300-dimensional-vector

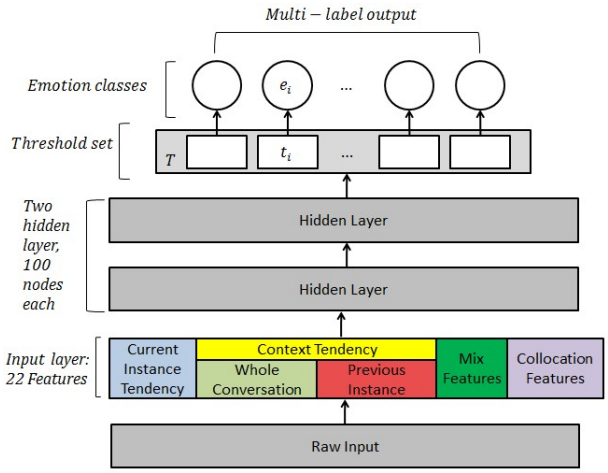


Figure 2: Structure of the manual feature selection network (MFSNet)

concatenated product of the utterance itself and the above-mentioned contextual information. This representation is then fed to the input layer of the neural network in the below figure 3.

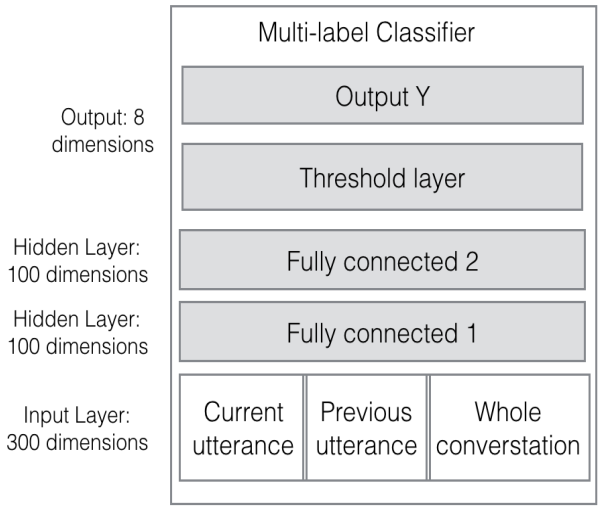


Figure 3: Structure of the word-embedding network

The two networks are both experimented on our annotated corpus. We report the performance of each network and make comparison of our methods to the another method and corpus in the next section.

5.2. Evaluation of the corpus

To evaluate, we use the two previous mentioned networks: manual feature selection network (MFSnet) and word-embeddings network (WENet). We evaluate the result with the gold standard test data using two major measurements in multi-label learning: hamming score (or accuracy in multilabel classification), multilabel F1-score (Table 5). The most important baseline is the average agreement score of the 5 human annotators on our corpus. We also want to compare our corpus to the existing Twitter Emotion Corpus (TEC) (Mohammad, 2012b) for their agreement score and their system's performance. The Twitter Emotion Corpus has tweets with emotion word hashtags. Similar to our work of creating word-embeddings, TEC was used to create the NRC Hashtag Emotion Lexicon (Mohammad, 2012a).

Corpus	Baselines	Hamming score	F1-score
EMTC	**Human annotators**	**43.2**	**62.6**
	WENet	39.1	53.9
	MSFNet	35.1	41.4
TEC	Human annotators	—	43.7
	Binary Classifiers	—	42.2

Table 5: Corpus evaluation

As we can observe, our simple system are performing worse than human annotators by a considerable margin. However, in comparison with other corpora 's Inter-annotators agreement F1-score such as Twitter Emotion Corpus (the score is 43.7), we see the potential of the corpus: It is reliable and the performance of emotion analysis system on it is great. The result is especially significant when our corpus is multilabeled and consists of conversational data which are much more complicated and practical. Both of our networks benefit from the corpus and have good F1-score. While MFSNet is slightly behind Binary Classifiers of TEC, WENet outperforms both. This result suggests that automatic feature extraction using word-embedding is better than manual selected features. We believe that because our embedding are built from the corpus, it has captured the relation between emotional words in the corpus better than general domain lexicon like Wordnet-Affect.

6. Conclusion

In this paper, we present our Emotion Movie Transcript Corpus developed from Imdb quotes dataset. EMTC consists of conversational text extracted from movies and as a result, is close to real-life settings and very practical for emotion analysis tasks. The corpus is partly annotated using our multilabel scheme and the annotators are provided with corresponding movie clips to ensure the reliability of the inter-annotators score of the corpus. We also conduct experiments on two networks: MFSNet that uses manual feature selection and WENet that uses Word-embedding to extract the bag-of-features from the input for supervised learning. The statistics and experimental results show that our extracted word-embedding and the corpus are reliable and even a very simple supervised method like WENet can perform fairly well using only bag-of-features from the embedding.

We would like to investigate the correlation among annotated labels and expand the size of testing data of our corpus using the same annotating scheme in the future. After that, we would focus on building an emotion lexicon from the word-embedding extracted from EMTC.

7. Acknowledgements

This research was supported by JST CREST Grant Number JPMJCR1513, Japan. We deeply appreciate the support of

our colleagues from Computational Linguistics Lab and all the annotators participated in this research for the valuable effort and patience.

8. Bibliographical References

Arimoto, Y. and Okanoya, K. (2016). Comparison of emotional understanding in modality-controlled environments using multimodal online emotional communication corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Choe, W., Chun, H.-S., Noh, J., Lee, S.-D., and Zhang, B.-T. (2013). Estimating multiple evoked emotions from videos. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 35.

Collier, G. (2014). *Emotional expression*. Psychology Press.

Dini, L. and Bittar, A. (2016). Emotion analysis on twitter: The hidden challenge. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Hareli, S., Kafetsios, K., and Hess, U. (2015). A cross-cultural study on emotion expression and the learning of social norms. *Frontiers in psychology*, 6:1501.

Hasegawa, T., Kaji, N., Yoshinaga, N., and Toyoda, M. (2013). Predicting and eliciting addressee's emotion in online dialogue. In *ACL (1)*, pages 964–972.

Kruger, J., Epley, N., Parker, J., and Ng, Z.-W. (2005). Egocentrism over e-mail: Can we communicate as well as we think? *Journal of personality and social psychology*, 89(6):925.

Li, M., Long, Y., Qin, L., and Li, W. (2016). Emotion corpus construction based on selection from hashtags. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Liew, J. S. Y., Turtle, H. R., and Liddy, E. D. (2016). Emotweet-28: A fine-grained emotion corpus for sentiment analysis. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Lim, N. (2016). Cultural differences in emotion: differences in emotional arousal level between the east and the west. *Integrative medicine research*, 5(2):105–109.

Mohammad, S. (2012a). Portable features for classifying emotional text. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 587–591. Association for Computational Linguistics.

Mohammad, S. M. (2012b). # emotional tweets. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 246–255. Association for Computational Linguistics.

Plutchik, R. (2001). The nature of emotions human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA. `http://is.muni.cz/publication/884893/en`.

Russell, J. A. (2003). Core affect and the psychological construction of emotion. *Psychological review*, 110(1):145.

Scherer, K. R. and Wallbott, H. G. (1994). Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310.

Strapparava, C. and Mihalcea, R. (2007). Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 70–74. Association for Computational Linguistics.

Strapparava, C., Valitutti, A., et al. (2004). Wordnet affect: an affective extension of wordnet. Citeseer.

Wang, Z., Lee, S., Li, S., and Zhou, G. (2015). Emotion detection in code-switching texts via bilingual and sentimental information. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 763–768, Beijing, China, July. Association for Computational Linguistics.

Yang, C., Lin, K. H.-Y., and Chen, H.-H. (2007). Emotion classification using web blog corpora. In *Web Intelligence, IEEE/WIC/ACM International Conference on*, pages 275–278. IEEE.

Complex and Precise Movie and Book Annotations in French Language for Aspect Based Sentiment Analysis

Stefania Pecore, Jeanne Villaneau
IRISA-D6 Media et Interactions
Université de Bretagne Sud, France
{stefania.pecore,jeanne.villaneau}@univ-ubs.fr

Abstract

Aspect Based Sentiment Analysis (ABSA) aims at collecting detailed opinion information according to products and their features, via the recognition of targets of the opinions in text. Though some annotated data have been produced in challenges as SemEval, resources are still scarce, especially for languages other than English. We are interested in enhancing today's mostly statistical text classification with the use of linguistics tools, in order to better define and analyze what has been written. The work presented in this paper focuses on two French datasets of movies and books online reviews. In reviews, text length is much higher compared to a tweet, giving us the opportunity to work on a challenging and linguistically interesting dataset. Moreover, movies and books are products that make classifying opinions into aspects quite complex. This article provides an analysis of the particularities of the two domains during the process of collecting and annotating data, a precise annotation scheme for each domain, examples and statistics issued from the annotation phase, and some perspectives on our future work.

Keywords: French Aspect Based Sentiment Analysis, opinion mining, user reviews

1. Introduction

The Web makes it relatively easy to collect a large amount of textual data. Researchers, as well as enterprises, politicians and governments are very interested in analyzing them, especially to learn more about people's opinions. This is due to many reasons: to improve services and products, to know the public opinion, to avoid terrorism acts. For researchers, opinion analysis is a multifaceted problem, with different names according to the studied aspect: subjectivity analysis (Wiebe et al., 2004), opinion mining (Pang and Lee, 2008), sentiment extraction (Das and Chen, 2007). While the subject is mature, as proved by the many published surveys (Pang and Lee, 2008; Liu, 2012), there is still room for improvement, demonstrated by the interest for the yearly NLP conferences and workshops as SemEval or Wassa(Mohammad and Bravo-Marquez, 2017) and by the challenges that opinion analysis still offers (Breck and Cardie, 2016; Mohammad, 2016).

Aspect Based Sentiment Analysis (ABSA) aims at "determining the orientation of sentiment expressed on each aspect" (Liu, 2012). ABSA was introduced as a shared task for the first time in SemEval-2014 (Pontiki et al., 2014), with datasets in English language for two domains: laptops and restaurants. In SemEval-2015, the task was repeated and extended, adding the hotel domain with a dataset of whole reviews and not just isolated sentences (Pontiki et al., 2015). In SemEval-2016 (Pontiki et al., 2016), new and multilingual datasets were provided: restaurant reviews in six languages (English, French, Dutch, Russian, Spanish and Turkish), hotel reviews in Arabic, consumer electronics reviews in three languages (English, Dutch and Chinese), telecom reviews in Turkish and museum reviews in French.

In this paper we propose an annotated French dataset for different domains compared with SemEval ones: movies and books. We have chosen to create a corpus of reviews and not tweets because we are interested in combining statistical and linguistical approaches, therefore we need to work on texts longer than tweets. As far as we know, there are few works in French for these domains (Hamdan et al., 2016), as a consequence it is not so easy to find this kind of data. Using ABSA is a difficult task on such reviews, because opinion is expressed in complex and various forms. Unlike other kind of reviews, limited in total amount of usable characters, these reviews are non-predictable in terms of length, and may carry opinions about other products related to the reviewed one, which are used as comparison. They can also merge in a same paragraph user's opinion and description of the evaluated product.

The paper describes our corpus (see Section 2.) and the specific form of annotation we have chosen in order to cope with comparisons and language complexity (Section 3.). Finally, we briefly explain how our future work will exploit these annotated data (Section 4.).

2. Corpus

2.1. Collecting text reviews

In our first experiments, the corpus was only composed by a collection of 450 books reviews and 450 movies reviews from the French Sentiment Corpus (FSC) produced between 2009-2013 by Vincent and Winterstein (2013). According to that paper, FSC was not originally used for Aspect-Based Sentiment Analysis, but to study opinion polarities. We decided to extend the corpus by adding more recent reviews - the NC Corpus, created between 2016 and 2017. The final corpus is the result of the union of these two corpora, both with equal dimensions. In total it is composed by 1800 reviews, i.e 4113 sentences for the books domain and 5222 for the movies domain. The corpus comes from two websites: Amazon.fr for books, and Allocine.fr for movies. By taking into account word and line counts of similar projects, it is possible to state that the corpus used in this project is suitable for being used in the context of Aspect-Based Sentiment Analysis tasks. Compared to other corpora in the field, the one used in this project has

a higher-than-average number of sentences. Some examples are (Hamdan et al., 2016) with 200 books reviews in French, (Alvarez-Lopez et al., 2017) with 2977 sentences from English books reviews, (Thet et al., 2010) with 1000 sentences from English movies reviews, (Sorgente et al., 2014) with 2648 sentences from movies reviews in Italian.

The corpus deals with books and movies using three types of rating: 1 for extremely negative reviews, 3 for the moderate ones, and 5 for the extremely positive ones. We decided to include 3 stars rating reviews because we noticed that, in such reviews, reviewers do not express a strong opinion: they are therefore induced to justify their balanced opinion to precise what they consider to be the negative and positive aspects of the book or movie.

The corpus provides the title of the reviewed product, to detect if the movie or book analyzed is the same of the title, or if it is another one used as term of comparison.

While making NC corpus, we noticed that collecting negative books reviews was not so easy, because there was an overwhelmingly higher amount of positive reviews compared with the negative ones. We suppose that, unlike people reviewing movies, people who don't like the book that they are reading prefer to close it and not to write a review - unless they have read others author's works and they want to criticize the author's choices compared to the previous ones.

Concerning the linguistics point of view we noticed that, more than in movies reviews, the book genre constitutes a linguistics sub-domain of books, i.e. user's style changes based on the genre of the book that is reviewed. For this reason we chose, when possible, to collect reviews of different types of books, in order to have a more complete annotation. In effect we noticed two phenomena: books' reviewers seem to make fewer misspellings than movies' reviewers; in addition to this, it seems that there is a difference between the diversity of words used in movies and books reviews, even when they share the same topic. The difference is to be found in the fact that, in movies reviews, users do an evaluation by just labeling the movies as good or bad (following a description of the several aspects of the product). In books reviews, instead, users try also to develop a critique that uses a variety of context-specific words.

In Table 1 a comparison between a movie review and a book review sharing the same topic: the consequences of immigration in France and the presence of Islam.

We can notice that the movie review shows several **misspellings** (**déçut* [for déçu - disappointed], **ou* [for où - where], **vraimant — *vraiament* [for vraiment - really], **kritic* [for critique - critics], **parceuq'lle* [for parce qu'elle - because she]), **slang** (*kiffé* [love],**Pourkoi* [for Pourquoi - why],**je vois pas* [for je ne vois pas - I don't see (the reasons why...)], **ke* [for que - what]) and a **general vocabulary** to evaluate a movie and not specially designed for the topic (*un beau film, j'ai aimé les images magnifiques, les décors somptueux* (...) *y avait la musique vraimant belle* [a good film, I loved the wonderful images, the gorgeous sets (...) the music was really good]). On the other hand, in books reviews sometimes it is harder to find misspellings. Moreover, it is not just an evaluation about the strengths and weaknesses of the product (*L'habilité de l'écrivain* nous permet de voir la société dans laquelle nous vivons [*the writer's talent* lets us perceive the society in which we live]). The reviewer, in fact, develops a critique about the topic of the book using more context-related expressions (*excès individualistes* [individualistic excess], *facilité par lâcheté*[facilitated by cowardice], *remords historiques*[historical remorse], *repentances incessantes* [continuous repentance], *être chassé de ses terres* [to be chased out of his lands], *politique d'immigration* [immigration policy], *pamphlet de la haine* [pamphlet of hatred], *paix sociale* [social peace], *viols, vols, trafics et agressions* [rape, robbery, traffic and assault])

Movies	Ben moi j'ai vu le film aujourd'hui à 16h à Torcy (...) Eh ben j'ai pas été ***déçut** (...) C'est juste un beau film, j'ai aimé les images magnifiques, les décors somptueux, rien que pour ça déjà j'ai trop **kiffé**. (...) ***Pourkoi** cette hostilité alors qu'il est bien filmé, ***je vois pas ou** il s'est senti trahi??? bon voilà, c'est que mon avis, pas une ***kritic** mytho. Ey puis y avait la musique ***vraimant** belle. j'ai passé un bon moment et faut que j'y retourne avec ma mère ***parceuq'lle** voulait pas y aller a cause des polémiques. Moi je trouve que ça parle ***vraiament** de ce ***ke** se passe dans les quartiers.
Books	L'habilité de l'écrivain nous permet de voir la société dans laquelle nous vivons, avec ses **excès individualistes** (...). N'importe quel peuple qui baisse les bras et cède à la **facilité par lâcheté** (...), **remords historiques** et **repentances incessantes** finit par **être chassé de ses terres**. Cela n'a rien de surprenant en soi (...) seulement il y a une sorte de couvercle posé sur l'Islam en France, parce qu'il est le résultat d'une **politique d'immigration** (...). Ce livre n'est pas un **pamphlet de la haine** (...) Ceux qui y voient une incitation à la haine sont les mêmes qui, au nom de la **paix sociale**, acceptent les milliers de **viols**, **vols**, **trafics** et **agressions** que subissent depuis trop longtemps les Français(es) honnêtes.

Table 1: Two examples of movies and books reviews showing some differences concerning writing style and misspellings.

2.2. Corpus statistics

Each part of the corpora (divided per rating) contains 150 reviews. Table 2 gives the number of words for each of them. The total number of words is 169,333 distributed over 9335 sentences.

Rating	Rat. 1	Rat. 3	Rat. 5	Years
FSC movie	11621	10682	11319	2009-13
NC movie	19974	31668	11339	2016-17
FSC book	14498	11259	8816	2009-13
NC book	14847	9732	13578	2016-17
Total	60940	63341	45052	

Table 2: Number of words of the corpora.

Statistical data (see Table 3) show that the number of words of the reviews is very variable (some reviews have more than 2000 words). The great σ indicates a great dispersion of the values and the difference between *mean* and *median* is the clue of an asymetrical distribution.

	Mean	σ	median	min	max
Movies	103.02	139.64	60	1	2094
Books	81.20	118.74	48	12	2052

Table 3: Statistic data related to the number of words of the reviews.

The histogram showing the number of words of movies and books reviews (see Figure 1) indicate that most of the reviews are composed by less than 200 words. A box-plot reveals that, for the movies domain, reviews with more than 254 words are to be considered outliers, and the average amount of words per review is 127.5. On the other hand, for the books domain, reviews with more than 181 words are to be considered outliers, and the average amount of words per review is 96.5.

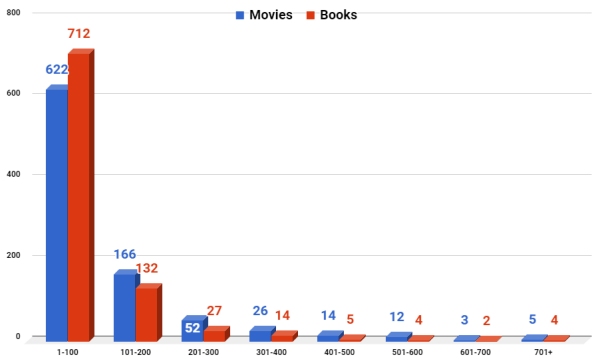


Figure 1: Number of words of reviews: histograms.

Another difference between the two parts of the corpus is that, even if movies reviews are generally longer than books reviews, they are less dense. In other words, we noticed that book reviewers are more creative in language expression than movie reviewers, as it is possible to see from the table 4 that shows the ratio between the total number of words (*token*) and the unique words (*types*) on the different parts of the corpus divided per type and rating. Movies' users tend to write longer reviews when they are giving a three-star review to a product (M3 has 43289 tokens, compared with negative reviews - 31595 tokens, and positive - 22658 tokens).

FSC corpus, despite the domain, shows a number of tokens lower than the new one (exception made by the FSC B3 reviews). We suppose it is due to the increasing comfort for the user in being critical compared with the past. In fact, it is common to find very long reviews reposted from blog articles.

Movies	Token	Type	Ratio
m1	31595	6624	20.9%
m3	43289	8244	19%
m5	22658	5443	24%
Books	**Token**	**Type**	**Ratio**
b1	29345	7492	**25.5%**
b3	21002	4916	**23.4%**
b5	22394	5895	**26.3%**

Table 4: Density of the text: differences between corpora and domain.

3. Annotation

The objective of the annotation is to provide a both generic and precise tool, which makes it possible to:

1. study (or automatically learn) the vocabulary used to express a negative or positive opinion, generally or in a given aspect;
2. know indications of the aspect related to a given positive or negative sentiment.

At the same time, we wanted the annotation to offer a generic aspect classification, which can be applied to various kinds of books or movies.

3.1. Annotation scheme and guidelines

We tried to create annotation schemes suitable for all type of movies or books, regardless of genre. We did not provide an annotation scheme specially designed for e-books. This is due to two reasons: first, we did not find many big differences between a book review and an e-book review; second our corpus does not have e-books reviews. The annotation scheme is composed by aspects and attributes (see Table 5 and 6). The annotation scheme for the book domain is composed by 5 aspects and 19 attributes. The movie one is composed by 7 aspects and 28 attributes. These classifications may be viewed as very precise, but grouping classes is easy, depending on the expected use of the corpus.

Aspects	**Attributes**
General Feeling	_
Text	*General, Subject, Style, Characters, Pace/Narration, Interest/Accuracy, Translation/Adaptation, Readability*
Illustration	*General, Interest/Accuracy, Graphic quality*
Author	*General, Text Author, Translator, Illustration Author,*
Form	*General, Bookbinding, Typography, Inner structure, Distribution*

Table 5: Aspects and Attributes for books reviews.

Aspects	Attributes
General Feeling	_
Direction	*General, Director, Point of view, Direction of actors, Shooting, Sound recording*
Acting	*General, Actor, Stuntman*
Visual	*General, Sets, Costumes, Special FX*
Sound	*General, Music, Sound Effect, Songwriter*
Script	*General, Subject, Plot, Dialogues, Characters, Pace/Narration, Remake/Adaptation/Reboot*
Distribution	*General, Type of Data Storage, Original Version/French Version*

Table 6: Aspects and Attributes for movies reviews.

3.2. Opinion annotation

Each opinion expression is annotated in three steps.

1. The first step is to select a group of contiguous words that indicate a positive or negative opinion. Opinion is evaluated by an ordinal value: -1 or -2 for a negative sentiment, according to its intensity; 1 or 2 if the sentiment is positive.

2. The second step is to detect the entity to which the opinion has to be reported. This entity is not always expressed, especially if it is the movie or book that is evaluated. When it is expressed, it is most of the time a name or a nominal group. Since including coreference resolution is beyond the subject of this work, pronouns are not selected as entities. Whenever opinion expression refers to a pronoun, the entity is reported of its previous closest reference. If the entity is detected, a relation is created, which joins opinion expression with entity phrase.

3. In the third step, an aspect and an attribute are chosen in the annotation scheme.

The many forms of opinion expressivity are related to various relations between entity, opinion word and aspect, as pointed out by the following examples.

- In the very simple phrase *"Un bon film" [a good movie]*, *"bon" [good]* indicates a positive sentiment (value: 1), the entity is *film [movie]* and the aspect *General Feeling* is chosen because of the entity.

- In *"c'est un navet" [it's a rubbishy movie]* the word *"navet" [rubbishy movie]* indicates a very negative sentiment (value: -2) and refers to the entity at the same time. So, the aspect, given by the entity, is *General Feeling*.

- In the phrase *"Le style est très agréable" [the style is very pleasant]*, extracted from the book corpus, the phrase *"very pleasant"* indicates a very positive sentiment (value: 2). The entity is *"style"*. It indicates that the category is *Text* with *Style* attribute.

- In the same corpus, the phrase *"le livre est bien mal écrit" [the book is very badly written]*, the phrase related to the sentiment is *"bien mal écrit" [very badly written]* (value: -2), the entity is *"livre" [book]*. The aspect is *Text* with *Style* attribute, because of the verb *écrire [to write]*.

- In a very negative book review, *"la bobo au style frelaté" [the boo-boo with degenerated style]*, the word *degenerated* refers to a very negative opinion (-2). It can be reported to the entity *Style* and classified in *Text#Style*). Because of the reference to the style, one can say that *bobo* refers to the author; like in *"un navet"*, *"la bobo"* expresses in a single word the entity and the opinion of the reviewer.

Previous examples, though being very simple, show how entities, opinion phrases and context have to be combined, to determine the aspect to which they have to be reported. The complexity of expression in the corpus makes it difficult to allocate aspects only to entities, as it is classically done, for example in SemEval 2016 annotation (Apidianaki et al., 2016).

3.3. Entities related to other products

Some phrases indicate a positive or negative sentiment related to another book or movie, most of the time to be compared with the reviewed one. For example, in
"rien à voir avec le seigneur des anneaux, carrément passionnant" [(this film has) nothing to do with the Lord of the rings, (that is) downright fascinating], (1)
the phrase *"downright fascinating"* indicates a very positive opinion, but it is applied to another movie: *"the Lord of the rings"*. On the contrary, the full phrase indicates a negative feeling about the movie.
Such comparisons are frequent and can be a problem for an automatic opinion detection; that is why we wanted the possibility for the annotation to report them precisely. To cope with the problem, the annotation of the entities indicates whether they are or not related to the evaluated product, a product of the same series, another product, etc. So, in the previous example (1), the very positive phrase *"downright fascinating"* is reported to *"the Lord of the rings"* classified as an entity which refers to *another product*. The phrase *"(this film has) nothing to do with the Lord of the rings"* is annotated as a negative opinion, reported to an entity which refers to the evaluated product.

3.4. Annotation Process

The annotation has been done by two experts: a native speaker and a non-native speaker. After the choice of the annotation form and the redaction of the guidelines, experiments have been conducted to estimate inter-annotator agreement. The most difficult task was the selection of the phrases related to an opinion, with particular attention to the determination of their scope. As per word selection, Cohen's κ was equal to 0.71, an acceptable result given the difficulty and subjectivity of the task. However, to improve the reliability of the corpus, we decided to perform a cross-reading of the annotations between the two annotators.

The annotation was performed via Glozz software (Widlocher and Mathet, 2012). Glozz is a multi-purpose text annotation tool, which comes with a fully WYSIWYG interface. It makes it possible to create units, defined as contiguous span of texts and relations between them. Annotations may be exported in several file formats and especially as *SQL data*.

3.5. Annotation Results

We annotated 5001 opinion phrases on movies (M1, M3, M5) and 3274 on books (B1, B3, B5). Annotations on negative reviews outnumber annotations on positive reviews, with circa 1992 annotations on M5 and B5 corpora against around 2899 on M1 and B1 corpora.

Figure 3 shows how annotations are distributed between the main classes. Nearly half of annotations are classified as *General_Feeling* in both corpora.

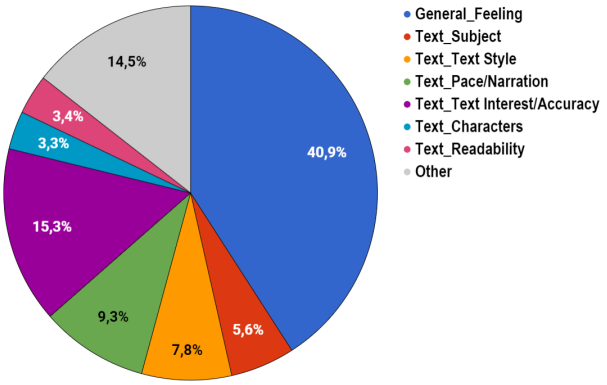


Figure 2: Books reviews: annotations collected on specific aspects.

In the book corpus, the most important six classes related to specific aspects are *Text#Interest and Accuracy*, *Text#Pace-Narration*, *Text#Style*, *Text#Characters*, *Text#Readability* and *Text#Subject*. All of them are related to the aspect *Text* and they collect 44.7% of the annotations not classified as *General_Feeling*. In the very wide variety of the assessed books, the textual aspect represents therefore a very large majority, with great importance given to the interest in the content.

In the movie corpus, the most important five classes related to specific aspects are *Acting#General*, *Script#Plot*, *Direction#General*, *Script#Pace/Narration* and *Script#General*. They collect 47.6% of the annotations not classified as *General_Feeling*. Apart from *Distribution*, all the aspects collect a significant number of annotations: the cinema is a multi-modal media, which combines sound and images to tell a story played by actors.

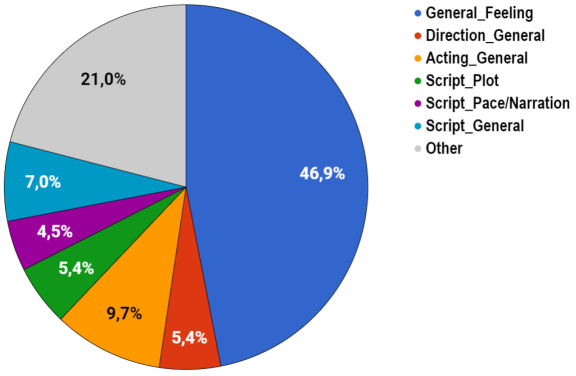


Figure 3: Movies reviews: annotations collected on specific aspects.

6392 entities have been selected; 5693 of them are related to the book or movie on which the review was written; this means that around 5% of the opinion expressions are related to another book or movie: a little more than 3.8% for the movie corpus and slightly less than 6.9% for the book corpus. Since most of them are very different from the general opinion of the reviewed book or movie, it is interesting to detect them.

	No. Relations	No. Aspects	No. Entities
Movies	5001	6235	3903
Books	3274	4787	2489

Table 7: Number of relations, aspects and entities annotated.

4. Perspectives and conclusion

The goal of the annotation is to give to the research community a dataset composed of annotated French opinion expressions about movies and books. The dataset has been created using the Aspect-Based Sentiment Analysis method, which is, based on our experience, the best fitted method to analyze these domains. Our future work will use our annotations on a system merging linguistics analysis and statistics. We will use a shallow parser and an opinion lexicon that takes advantage of the dataset to track complex expressions.

5. Bibliographical References

Alvarez-Lopez, T., Fernandez-Gavilanes, M., Costa-Montenegro, E., Juncal-Martiez, J., Garcia-Mendez, S., and Bellot, P. (2017). A book reviews dataset for aspect based sentiment analysis. In *8th Language Technology Conference*.

Apidianaki, M., Tannier, X., and Richart, C. (2016). Datasets for aspect-based sentiment analysis in french. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Breck, E. and Cardie, C., (2016). *Oxford Handbook of Computational Linguistics.*, chapter Opinion mining and sentiment analysis. Oxford University press, 2nd edition. *to appear*.

Das, S. R. and Chen, M. Y. (2007). Yahoo! for amazon: Sentiment extraction from small talk on the web. *Management Science*, 53(9):1375–1388.

Hamdan, H., Bellot, P., and Bechet, F. (2016). Sentiment analysis in scholarly book reviews. In *arXiv preprint arXiv:1610.03106*.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan and Claypool Publishers.

Mohammad, S. M. and Bravo-Marquez, F. (2017). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mohammad, S. M. (2016). Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In Herb Meiselman, editor, *Emotion Measurement*. Elsevier.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135.

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., and Androutsopoulos, I. (2015). Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado, June. Association for Computational Linguistics.

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., et al. (2016). Semeval-2016 task 5: Aspect based sentiment analysis. In *ProWorkshop on Semantic Evaluation (SemEval-2016)*, pages 19–30. Association for Computational Linguistics.

Sorgente, A., Vettigli, G., and Mele, F. (2014). An italian corpus for aspect based sentiment analysis of movie reviews. In *CLICIT*.

Thet, T. T., Na, J.-C., and Khoo, C. S. (2010). Aspect-based sentiment analysis of movie reviews on discussion boards. *J. Inf. Sci.*, 36(6):823–848, December.

Vincent, M. and Winterstein, G. (2013). Construction et exploitation d'un corpus français pour l'analyse de sentiment. In *TALN-RÉCITAL 2013*, Les Sables d' Olonne, France.

Widlocher, A. and Mathet, Y. (2012). The glozz platform: A corpus annotation and mining tool. In *Proceedings of the 2012 ACM symposium on Document engineering*, pages 171–180.

Wiebe, J., Wilson, T., Bruce, R., Bell, M., and Martin, M. (2004). Learning subjective language. *Comput. Linguist.*, 30(3):277–308, September.

Lingmotif-lex: a Wide-coverage, State-of-the-art Lexicon for Sentiment Analysis

Antonio Moreno-Ortiz, Chantal Pérez-Hernández
University of Málaga
Spain
{amo,mph}@uma.es

Abstract

We present *Lingmotif-lex*, a new, wide-coverage, domain-neutral lexicon for sentiment analysis in English. We describe the creation process of this resource, its assumptions, format, and valence system. Unlike most sentiment lexicons currently available, *Lingmotif-lex* places strong emphasis on multi-word expressions, and has been manually curated to be as accurate, unambiguous, and comprehensive as possible. Also unlike existing available resources, *Lingmotif-lex* comprises a comprehensive set of contextual valence shifters (CVS) that account for valence modification by context. Formal evaluation is provided by testing it on two publicly available sentiment analysis datasets, and comparing it with other English sentiment lexicons available, which we adapted to make this comparison as fair as possible. We show how *Lingmotif-lex* achieves significantly better performance than these lexicons across both datasets.

Keywords: sentiment analysis, opinion mining, lexicons, language resources

1. Introduction

Sentiment Analysis has received increased attention in the last decade as a subtask of Natural Language Processing. The task can be roughly summarized as the classification of entire documents or parts of it (sentences or text segments that denote a certain aspect about a certain entity). Most efforts focus on the classification of domain-specific documents, usually short texts such as product reviews and tweets, although other more sophisticated tasks are becoming common, such as emotion and intensity detection. Traditionally, corpus-based and lexicon-based approaches have been distinguished in terms of the general system architecture. Strictly speaking, the former use a training corpus to extract textual cues found in each of the tagged documents, whereas the latter employ a sentiment lexicon where sentiment-carrying words are stored.

In practice, however, NLP practitioners combine methodologies from both approaches, for example, (Riloff et al., 2006). Generally speaking, lexicon-based approaches are preferred for sentence-level classification (Andreevskaia and Bergler, 2007), whereas corpus-based, statistical approaches are preferred for document-level classification.

The use of sentiment dictionaries is a widespread methodology, since it makes sense that the presence of certain sentiment-carrying words determine the polarity of the text in which they appear. WordNet (Fellbaum, 1998) has been a recurrent source of lexical information (Kim and Hovy, 2004; Hu and Liu, 2004; Andreevskaia and Bergler, 2006) either directly as a source of lexical information or for sentiment lexicon construction. Other common lexicons used in English sentiment analysis research include The General Inquirer (Stone and Hunt, 1963), MPQA (Wilson et al., 2005), and Bing Liu's Opinion Lexicon (Hu and Liu, 2004).

Yet other researchers have used a combination of existing lexicons or created their own (Hatzivassiloglou and McKeown, 1997; Turney, 2002). The use of lexicons has sometimes been straightforward, where the mere presence of a sentiment word determines a given polarity. However, negation and intensification can alter the valence or polarity of that word.[1] Modification of sentiment in context has also been widely recognized and dealt with by some researchers (Kennedy and Inkpen, 2006; Polanyi and Zaenen, 2006; Choi and Cardie, 2008; Taboada et al., 2011).

One disadvantage of relying solely on a sentiment lexicon is that different domains may greatly alter the valence of words, a fact well recognized in the literature (Aue and Gamon, 2005; Pang and Lee, 2008; Choi et al., 2009). A number of solutions have been proposed to these, mostly using ad hoc dictionaries, sometimes created automatically from a domain-specific corpus (Tai and Kao, 2013; Lu et al., 2011).

Lingmotif-lex has been embedded in the Lingmotif application (Moreno-Ortiz, 2017a), a fully user-focused, lexicon-based sentiment analysis system, since its availability in 2016. The Lingmotif application has been showcased at relevant conferences (Moreno-Ortiz, 2017b) and used as the main tool in a number of sentiment analysis shared tasks (Moreno-Ortiz, 2017c; Moreno-Ortiz and Pérez-Hernández, 2017). It currently supports English and Spanish; other languages (French and Italian) will be subsequently added and released.

The experience gained from participating in these sentiment and emotion classification tasks, as well as the input received from the application's users since its release, has served us to identify issues in the lexical resources, improve them, and refine them. The evaluation results shown in section 5. have provided us with the confidence to release a high-quality resource that can readily be used for real-world NLP tasks. Our intention, however, is to keep releasing improved versions of the resources.

Our Lingmotif sentiment analysis system is entirely lexicon-based. It implements a shifters system, which we describe in section 4. below, but it does not yet have entity or aspect management capabilities. This means that the quality of its results are entirely dependent on the quality

[1] The terms *valence* and *polarity* are used inconsistently in the literature. We use *polarity* to refer to the binary distinction positive/negative sentiment, and *valence* to a value of intensity on a scale.

of its lexical resources, that is, Lingmotif-lex, which is the object of our evaluation in this paper.

2. Creation Process

Lingmotif-lex is the result of many years of computational lexicography work and testing on many different sentiment analysis tasks. We started by merging available sentiment dictionaries. Specifically, we merged items from The Harvard General Inquirer (Stone and Hunt, 1963), MPQA (Wilson et al., 2005), and Bing Liu's Opinion Lexicon (Hu and Liu, 2004), which we reduced to a common format that included simply the lemma or form and its associated polarity. The resulting lexicon was then expanded semiautomatically by using a thesaurus and derivational generation rules. Since we decided to use a graded valence system instead of binary polarity, all items were manually ranked on a -5 to 5 scale by a team of trained annotators, using consensus and corpus linguistics techniques.[2]

The above-mentioned original resources, however, are characterized by their lack of attention to multiword expressions (MWEs). Multiword expressions not only denote polarity very often, but they also serve as good disambiguation resource in terms of polarity (Moreno-Ortiz et al., 2013), and are therefore key to successful lexicon-based sentiment analysis. In order to provide good coverage for MWEs, we used a number non SA-specific lexical resources, including common idioms from Wiktionary, which we tagged manually for valence. Ultimately, Lingmotif's lexicons are the result of intensive lexicographical work.

The obtained lexicon was further enhanced by correcting and adding lexical entries manually. A desktop application, Lingmotif[3] (Moreno-Ortiz, 2017a; Moreno-Ortiz, 2017b), was developed that enabled our team of annotators to easily analyze texts to identify errors and omissions. Since our aim was to create a domain-neutral sentiment lexicon, we chose a varied sample of texts for this development stage, with the aim of creating a resource that could be used on any sentiment analysis task, regardless of genre or topic.

To account for the domain specificity issue, the Lingmotif application allows the use of *plugin lexicons*. This is a flexible mechanism that allows users to develop and optionally apply a domain-specific lexicon for particular topics or domains. Lexical information contained in plugin lexicons overrides that in Lingmotif's core lexicon. When a plugin lexicon is selected for analysis, the plugin lexicon is searched first. If a word or phrase is found there, the core lexicon will not be searched for that item, and its information in the plugin lexicon will be used. Thus, plugin lexicons can be used to provide domain-specific sentiment items, but also to override polarity assignment in the core lexicon, for whatever reasons.

Plugin lexicons use exactly the same format as the core lexicon. In order to import a plugin lexicon, it must first be created as a UTF-8 encoded CSV file, which is then imported. Updating a plugin lexicon simply involves modifying the source CSV file and importing it again. Any number of plugin lexicons can be created in Lingmotif, but only one can be used for a given analysis.

Lingmotif-lex, along with a number of compatible plugin lexicons is ongoing work.

3. Format and Valence System

A *Lingmotif-lex* language set consists of three components: the lexicon itself, the context rules, which account for context modification of sentiment. A code library was developed to facilitate interfacing matching an input against the lexical resources. Table 1 summarizes the number of items contained in the current version of *Lingmotif-lex* for English.[4]

Item	**Count**
Single words (forms)	28,000
Multiwords (forms)	38,570
Emojis	130
Context rules (shifters)	700
Total	**67,400**

Table 1: Number of entries in *Lingmotif-lex*

The lexicon is stored in a plain text, tab-separated file encoded in UTF-8, which allows us to include Emojis just like any other lexical entry. Each lexicon entry consists of four data fields, which we describe in table 2.

Data field	**Example/List**
Word form	well-trained, looked_down_upon
Part of speech	[ALL, NN, JJ, VB, RB, UH, IN]
Polarity	[POS, NEG, NEU]
Intensity	[1, 2, 3]

Table 2: Lexicon format

In the past, polarity and intensity were expressed as a simple negative or positive integer. However, the current system allows us to express cases where polarity is not well defined or highly context-dependent, but where the presence of intensity is unquestionable, which is a very common situation. Words and expressions such as "wild", "wicked", "sick", or "oh, my god" are clear examples. Thus, even if this format presents some more processing difficulty, it gains in expressive power, and allows users to apply their own disambiguation techniques if required.

The valence system in the release version, based on a 3-point intensity scale is also different from the scale used in previous versions used in the Lingmotif application, and described elsewhere (**?**), where a more fine-grained, 5-point scale was used. We found this to be useful for some cases, such as graded adjective series, but harder to define in many other scenarios. The current coarser 3-point scale is more intuitive for annotators (low, mid, high intensity), and just as useful for practical purposes.

All entries are lower case, and contain no blanks. The following are examples of *Lingmotif-lex* single-word entries:

```
kind JJ POS 2
```

[2]This scale was kept for some time, but the final version was reduced to a 3-point intensity scale. The valence system is described in section 3..

[3]The Lingmotif application is available for download at http://tecnolengua.uma.es/lingmotif

[4]These figures might be different in the final release version.

```
corrupt ALL NEG 2
curious ALL NEU 1
```

Single-word entries are pos-tagged as `ALL` by default, meaning that the word, functioning as any part of speech, has that valence. When a word takes the valence only when functioning as a particular part of speech, the part of speech is given (using the Penn Treebank tag set). Part-of-speech data is used by the context rule matching system.

Multiwords are always pos-tagged, and otherwise treated just like single words, the only difference being signaled by the presence of underscores separating the individual words that make them up. English phrasal verbs are also treated as MWEs. Multiword expressions may consist of up to six words, some of which may be placeholders, marked by "0". For example, the following set of entries:

```
brought_into_harmony VB POS 2
brought_0_into_harmony VB POS 2
brought_0_0_into_harmony VB POS 2
brought_0_0_0_into_harmony VB POS 2
```

would match the strings "brought into harmony", "brought them into harmony", "brought the situation into harmony", and "brought the guests back into harmony". This system allows to express and effectively match the vast majority of multiword expressions. Lingmotif generates such sets of entries from a more user-friendly format with the form

```
<bring>_3_into_harmony VB POS 2
```

where the word in angled brackets means it is a lemma, so all its forms hould be generated during import, and the integer in braces specifies the number of words that may occur between both part of the multiword expression, which generates the forms shown above.

Conceptually, however, there is an important difference between single and multi words in *Lingmotif-lex*. Most single words (except polarity-ambiguous ones) have a non-neutral polarity tag, and invariably have an intensity greater than 0, the assumption being that any single word not in the lexicon is, generally, not a sentiment word. On the other hand, the lexicon includes multiwords which may not be sentiment-carrying but contain individual words that are. The rationale behind this is that, were they not included in the lexicon, those individual words would be matched and (wrongly) identified as sentiment items (for example the word "kill" in "kill time". Further explanations and examples are provided in Moreno-Ortiz et al. (2013).

4. Sentiment shifters

Real-world NLP tasks involving sentiment analysis have often relied on sentiment lexicons, but less so on a wide-coverage system that accounts for contextual modification of the lexical items. Context can greatly modify the sentiment of, fundamentally, any sentiment word, either by inversion, such as in negation, where the overall polarity is inverted; or by intensification and downtoning by means of, for example, a quantifier. This means that the sheer presence of a sentiment word does not determine the polarity or valence assigned to that sentiment word in the lexicon. A shifters system may help accounting for contextual modification of sentiment, and is particularly interesting for fine-grained sentiment analysis, such as aspect-based SA, where the text's overall classification/score is not enough (Yu et al., 2016). However, such systems have rarely been fully implemented.

Lingmotif-lex does include a with a wide-coverage shifters system, which we have implemented by means of context rules, basically a template matching system on the input text. We use a similar linguistic approach to Polanyi and Zaenen (2006), Kennedy and Inkpen (2006), or Taboada et al. (2011). Our context rules work by specifying words or phrases that can appear in the immediate vicinity of the identified sentiment word. Our set of context rules has been compiled through extensive corpus-based work and have been tested on several sentiment analysis datasets. Currently, *Lingmotif-lex* contains over 700 such rules.

Table 3 shows the data structure used by *Lingmotif-lex*'s shifters system.

Data field	**Example/Value list**
Part of speech	[NN, JJ, VB, RB]
Polarity	[+, -]
Shifter form	not, tremendous*, pretty_much
Location	[L, R, LR]
Span	[1, 2, 3, 4, 5]
Result	[INT*n*, DOW*n*, VAL*n*, INV0]

Table 3: Context rules format

As can be deduced from the example shifters in table 3, they can be either a literal string or contain the asterisk as a wildcard (e.g., "tremendous*" matches both the adjective and the adverb "tremendously"). Shifters can also be multiword expressions, in which case they must be included in the lexicon. The *location* data field expresses the position of the shifter with reference to the sentiment word, and *span* defines the maximum number of words from it. The *result* values (INT*n*, DOW*n*, VAL*n*, INV0) define both the type of shifting produced (intensification, downtoning, value, inversion) and the degree of the shift (as expressed by the integer *n*).

Table 4 provides examples of all types of sentiment shifters according to the effect they produce on the resulting text segment.

Shift type	**Example context rule**
Intensification	`JJ +- highly L 1 INT2` `VB +- literal* LR 2 INT2`
Downtoning	`JJ +- sort_of L 1 DOW1` `NN +- limited L 1 DOW1`
Inversion	`NN +- decreas* LR 3 INV0` `JJ +- never L 2 INV0`
Final value	`VB + try_to L 1 VAL0` `JJ +- would_have L 3 VAL0`

Table 4: Context rules examples

The set of rules is in a separate plain text, CSV file, and a Python library provides easy access to it, allowing users to programmatically choose to individually apply inversion, intensification, downtoning, or no rules at all, in which case the original valence of the lexical items will be returned. The library returns the list of matched *sentiment items* and

their corresponding valences. Sentiment items, after applying context rules, may be a single word form, a sequence of word forms matching a multiword expression, or a sequence of word forms matching a single or multiword item and the strings that make up the context rule.

5. Evaluation methodology

We evaluate *Lingmotif-lex* in terms of performance on a typical sentiment classification task, using two publicly available sentiment analysis datasets, and compare it with five other well-known available English sentiment lexicons. Comparing sentiment lexicons is not easy, since different resources have different formats and data. For example, they may or may not include part of speech information, and, if this is present, different tagsets may be used. Their entries may be word forms or lemmas, and the valence system may also be different, from simple polarity to fine-grained data.

5.1. Datasets

After many years of research in sentiment analysis, the types of documents that have been dealt with in the field are varied. However, two types stand out among the rest: user reviews of products or services, and microblogging short texts, particularly Twitter data.

For our performance evaluation we chose two highly representative datasets of these two types. For the first, we used the classic movie reviews dataset from Pang and Lee (2004)'s seminal paper. This database has been widely used in the literature both for training and testing algorithms and resources. It is made up of 1,000 positive and 1,000 negative movie reviews tagged for polarity (pos/neg).[5] For Twitter data, we chose a dataset specifically designed for testing sentiment analysis techniques on microblogging texts, STS-Gold (Saif et al., 2013), which contains 2,034 tweets tagged for polarity (1,402 negative, 632 positive).

Present-day SA shared tasks commonly aim to classify texts into finer categories, including the neutral category and degrees of intensity, but we think simpler datasets, such as these two, provide a solid base on which to test sentiment lexicons, since there is less chance of annotation issues.

5.2. Sentiment lexicons

We compared the performance of *Lingmotif-lex* with five other well-known sentiment lexicons:

1. The General Inquirer (Stone and Hunt, 1963). This is one of the oldest sentiment lexicons publicly available. It is based on work in cognitive psychology and content analysis. This resource offers syntactic, semantic, and pragmatic information to part-of-speech tagged words, with 1915 positive and 2291 negative words. Lexical items for "yes" and "no" (in the sense of refusal) are grouped in separate categories and further semantic dimensions, such as strength or active/passive orientation, are also included[6].

2. Bing Liu Opinion Lexicon (Hu and Liu, 2004). A compilation of about 6,800 words drawn from product reviews, originally labeled using a bootstrapping method using WordNet adjective synsets and their antonyms (Hu and Liu, 2004). It contains 2006 positive and 4783 negative words[7].

3. MPQA (Multi-Perspective Question Answering) Subjectivity Lexicon (Wilson et al., 2005). This resource contains 2,718 positive and 4,912 negative words drawn from a combination of sources, including the General Inquirer lists, the output of the system created by Hatzivassiloglou and McKeown (1997) and a bootstrapped list of subjective clues (Riloff and Wiebe, 2003), hand-labeled for sentiment. The lexicon also includes labels for reliability (strongly subjective or weakly subjective) and four polarity tags: positive, negative, both and neutral. The majority of words are marked as having either positive (33.1 percent) or negative (59.7 percent) polarity, whereas only a small number of clues (0.3 percent) are marked as having both positive and negative, and 6.9 percent of the clues in the lexicon are marked as neutral[8].

4. SentiWordNet 3.0 (Baccianella et al., 2010). a lexical resource explicitly devised for supporting sentiment classification and opinion mining applications (Baccianella et al., 2010). It is the result of automatically annotating all WordNet synsets according to their degrees of positivity, negativity, and neutrality. The current version is 3.0 (based on WordNet 3.0) and differs from previous versions in the algorithm used for annotation, which now refines the scores by using a random-walk step in addition to the initial semi-supervised learning step[9].

5. Sentiment140 Lexicon (Mohammad et al., 2013). Created from a collection of 1.6 million tweets that contain positive and negative emoticons. This is the only lexicon in this list that contains multiwords, including 62,468 unigrams, 677,698 bigrams, and 480,010 pairs tagged as either positive or negative[10].

We made every effort to adapt all resources to a common usable format, although, inevitably, some information was lost in some adaptations. For example, some lexicons, such as MPQA, list lemmas rather than forms (implicitly, in this case), so we generated all forms for each lemma, which surely generates some forms that do not exhibit the same polarity.

5.3. Training and testing procedure

All lexicons were evaluated using the same metrics (precision, recall, and F1 score) obtained with the same training and testing procedure. Each dataset was split using a

[5] The dataset is available at http://www.cs.cornell.edu/people/pabo/movie-review-data/

[6] Available at http://www.wjh.harvard.edu/ inquirer/.

[7] Available at http://www.cs.uic.edu/ liub/FBS/opinion-lexicon-English.rar

[8] Available at http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/.

[9] Available at http://sentiwordnet.isti.cnr.it/.

[10] Available at http://saifmohammad.com/WebPages/lexicons.html.

widely-used approach to sentiment classification at the document level: 80 percent for training and 20 percent for testing. Sentiment words present in each of the lexicons were searched in the texts, and a positive and negative score was calculated for each text as the sum of the valences found in the lexicon. When the lexicon simply contained polarity, a valence of *1* was added to that polarity score; when the lexicon included a more fine-grained valence specification, that number was used. The classifier was trained exclusively on these two features: positive score and negative score, calculated as the accumulated sum of the valences of the matching items, as specified in each of the lexicons.

As for the classifier itself, we used a "traditional" support vector machine algorithm, specifically the SVC implementation in the Python-based scikit-learn (Pedregosa et al., 2011) machine learning toolkit, with the default RBF kernel and parameters for all lexicons.

This methodology minimally guarantees that the best results would be obtained by the best lexicons, since all other conditions where kept equal in all cases, the lexicon being the only variable. It is of course debatable whether a larger and/or more varied sample should be used, but we think these two datasets are fairly representative of typical sentiment analysis tasks.

6. Evaluation results

Tables 5, 6, 7, 8, 9, and 10 show the performance results we obtained for each of the lexicons on both datasets. It is interesting to see how all lexicons exhibit significant differences for each of the classes (positive, negative). Negativity consistently obtains high precision but low recall, whereas the positive class is just the opposite (better recall, poorer precision). This is surely due to the fact that negativity is expressed using more sophisticated linguistic resources (irony, sarcasm, understatements, etc.), and, therefore, not so easily identified. The difference in performance among classes is particularly evident in some lexicons, such as SentiWordNet (8) and Sentiment 140 (9).

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.81	0.64	0.71	1278
pos	0.54	0.74	0.62	722
avg/total	0.71	0.68	0.68	2000
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.96	0.73	0.83	1842
pos	0.21	0.69	0.32	192
avg/total	0.89	0.73	0.78	2034

Table 5: Evaluation results - General Inquirer

Tables 11 and 12 summarize the results we obtained for the movie reviews dataset and the STS-Gold dataset, respectively, for each of the lexicons.

As these results show, Lingmotif-lex obtains a higher score on both datasets, followed by Bing Liu's lexicon, which it improves by 0.3 and 0.4 respectively, a significant difference given the tight scores. This consistency of results is desirable, since it is a sign of reliability and predictability,

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.78	0.70	0.73	1115
pos	0.66	0.75	0.70	885
avg/total	0.73	0.72	0.72	2000
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.87	0.84	0.85	144
pos	0.63	0.69	0.66	585
avg/total	0.80	0.80	0.80	2034

Table 6: Evaluation results - Bing Liu

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.75	0.69	0.72	1092
pos	0.66	0.73	0.69	908
avg/total	0.71	0.71	0.71	2000
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.90	0.77	0.83	1628
pos	0.41	0.64	0.50	406
avg/total	0.80	0.75	0.76	2034

Table 7: Evaluation results - MPQA

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.97	0.71	0.82	1901
pos	0.13	0.63	0.22	133
avg/total	0.91	0.71	0.78	2034
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.97	0.71	0.82	1901
pos	0.13	0.63	0.22	133
avg/total	0.91	0.71	0.78	2034

Table 8: Evaluation results - SentiWordNet

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.73	0.64	0.68	1149
pos	0.58	0.69	0.63	851
avg/total	0.67	0.66	0.66	2000
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.94	0.76	0.84	1733
pos	0.34	0.71	0.46	301
avg/total	0.85	0.75	0.78	2034

Table 9: Evaluation results - Sentiment 140

something that may not be said of the other lexicons, especially MPQA, which ranked third on the movie reviews dataset, but last on the STS-Gold dataset.

7. Conclusions

In this paper we have presented *Lingmotif-lex*, a wide-coverage, non-domain-specific lexicon for sentiment analysis in English. We have described the creation process of

Movie reviews				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.81	0.72	0.76	1124
pos	0.68	0.78	0.73	876
avg/total	0.75	0.74	0.75	2000
STS-Gold				
Class	**Precision**	**Recall**	**F1**	**Support**
neg	0.93	0.85	0.89	1547
pos	0.62	0.81	0.70	487
avg/total	0.86	0.84	0.84	2034

Table 10: Evaluation results - Lingmotif-lex

Lexicon	**Precision**	**Recall**	**F1 Score**
Lingmotif-lex	0.75	0.74	0.75
Bing Liu	0.73	0.72	0.72
MPQA	0.71	0.71	0.71
General Inquirer	0.71	0.68	0.68
Sentiment-140	0.67	0.66	0.66
SentiWordNet 3.0	0.64	0.64	0.64

Table 11: Evaluation results - Movie reviews dataset

Lexicon	**Precision**	**Recall**	**F1 Score**
Lingmotif-lex	0.86	0.84	0.84
Bing Liu	0.80	0.80	0.80
Sentiment-140	0.85	0.75	0.78
SentiWordNet	0.91	0.71	0.78
General Inquirer	0.89	0.73	0.78
MPQA	0.80	0.75	0.76

Table 12: Evaluation results - STS-Gold dataset

this resource, embedded in the Lingmotif sentiment analysis system, and the way its format and valence system has evolved over the years. Three main features characterize *Lingmotif-lex* vis-à-vis other available sentiment lexicons: careful manual curation and testing on many different texts and datasets, the strong emphasis placed on multi-word expressions, and the inclusion of a valence shifters system, that accounts for valence modification by context.

We have also provided a formal evaluation of our lexicon by testing its performance on a typical sentiment classification task of two publicly available sentiment analysis datasets: the classic movie reviews used in Pang and Lee's groundbreaking 2004 paper and the STS-Gold collection of tweets. Our evaluation results show that *Lingmotif-lex* achieves significantly better performance than the other five lexicons evaluated across both datasets. Further testing, using other datasets, might be considered, but these results already offer enough evidence as to the high quality and usefulness of our resource.

8. Acknowledgements

This research was supported by Spain's MINECO through the funding of project Lingmotif2 (FFI2016-78141-P).

9. Bibliographical References

Andreevskaia, A. and Bergler, S. (2006). Mining wordnet for fuzzy sentiment: Sentiment tag extraction from wordnet glosses. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 209–216.

Andreevskaia, A. and Bergler, S. (2007). CLaC and CLaC-NB: Knowledge-based and corpus-based approaches to sentiment tagging. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 117–120, Stroudsburg, PA, USA. Association for Computational Linguistics.

Aue, A. and Gamon, M. (2005). Customizing sentiment classifiers to new domains: A case study. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, Borovets, Bulgaria.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 2200–2204, Valletta, Malta.

Choi, Y. and Cardie, C. (2008). Learning with compositional semantics as structural inference for subsentential sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 793–801, Stroudsburg, PA, USA. Association for Computational Linguistics.

Choi, Y., Kim, Y., and Myaeng, S.-H. (2009). Domain-specific sentiment analysis using contextual feature generation. In *Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 37–44, Hong Kong, China. ACM.

Christiane Fellbaum, editor. (1998). *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May.

Hatzivassiloglou, V. and McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 174–181, Madrid, Spain. Association for Computational Linguistics.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, Seattle, WA, USA. ACM.

Kennedy, A. and Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125.

Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1367, Geneva, Switzerland. Association for Computational Linguistics.

Lu, Y., Castellanos, M., Dayal, U., and Zhai, C. (2011). Automatic construction of a context-aware sentiment lexicon: An optimization approach. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 347–356, New York, NY, USA. ACM.

Mohammad, S. M., Kiritchenko, S., and Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the Seventh In-*

ternational Workshop on Semantic Evaluation Exercises (SemEval-2013), Atlanta, Georgia, USA, June.

Moreno-Ortiz, A. and Pérez-Hernández, C. (2017). Tecnolengua Lingmotif at TASS 2017: Spanish Twitter dataset classification combining wide-coverage lexical resources and text features. In *TASS 2017: Workshop on Semantic Analysis at SEPLN*, pages 35–42, Murcia, Spain. SEPLN.

Moreno-Ortiz, A., Pérez-Hernández, C., and Del-Olmo, M. (2013). Managing multiword expressions in a lexicon-based sentiment analysis system for spanish. In *Proceedings of the 9th Workshop on Multiword Expressions MWE 2013*, pages 1–10, Atlanta, Georgia, USA. The Association for Computational Linguistics.

Moreno-Ortiz, A. (2017a). Lingmotif: A user-focused sentiment analysis tool. *Procesamiento del Lenguaje Natural*, 58(0):133–140, March.

Moreno-Ortiz, A. (2017b). Lingmotif: Sentiment analysis for the digital humanities. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 73–76, Valencia, Spain, April. Association for Computational Linguistics.

Moreno-Ortiz, A. (2017c). Tecnolengua lingmotif at EmoInt-2017: A lexicon-based approach. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 225–232, Copenhagen, Denmark, September. Association for Computational Linguistics.

Pang, B. and Lee, L. (2004). A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 271–278, Barcelona, Spain. Association for Computational Linguistics.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830, November.

Polanyi, L. and Zaenen, A. (2006). Contextual valence shifters. In James G. Shanahan, et al., editors, *Computing Attitude and Affect in Text: Theory and Applications*, pages 1–10. Springer Netherlands, Dordrecht.

Riloff, E. and Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 105–112, Stroudsburg, PA, USA. Association for Computational Linguistics. ACM ID: 1119369.

Riloff, E., Patwardhan, S., and Wiebe, J. (2006). Feature subsumption for opinion analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 440–448, Stroudsburg, PA, USA. Association for Computational Linguistics. ACM ID: 1610137.

Saif, H., Fernández, M., He, Y., and Alani, H. (2013). Evaluation datasets for Twitter sentiment analysis: a survey and a new dataset, the STS-Gold. In *1st Interantional Workshop on Emotion and Sentiment in Social and Expressive Media: Approaches and Perspectives from AI (ESSEM 2013)*, volume 3, Turin, Italy.

Stone, P. J. and Hunt, E. B. (1963). A computer approach to content analysis: Studies using the general inquirer system. In *Proceedings of the May 21-23, 1963, Spring Joint Computer Conference*, AFIPS '63 (Spring), pages 241–256, New York, NY, USA. ACM.

Taboada, M., Brooks, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

Tai, Y.-J. and Kao, H.-Y. (2013). Automatic domain-specific sentiment lexicon generation with label propagation. In *Proceedings of International Conference on Information Integration and Web-based Applications & Services*, IIWAS '13, pages 53:53–53:62, New York, NY, USA. ACM.

Turney, P. D. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 417–424, Philadelphia, USA.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yu, H., Shang, J., Hsu, M., Castellanos, M., and Han, J. (2016). Data-driven contextual valence shifter quantification for multi-theme sentiment analysis. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, pages 939–948, New York, NY, USA. ACM.

A Japanese Corpus for Analyzing Customer Loyalty Information

Yiou Wang, Tahara Takuji
Fuji Xerox Co., Ltd
6-1, Minatomirai, Nishi-ku, Yokohama-shi, Kanagawa 220-8668, Japan
{Yiou.Wang, Tahara.Takuji} @fujixerox.co.jp

Abstract

Today customers voice attitudes, opinions and their experience about some brands, companies, products or services through center calls, web reviews or SNS and analyzing them is an important task. On the other hand, customer loyalty has long been a topic of high interest in both academia and industry. Therefore, it is attractive to consider exploiting customer loyalty information by analyzing the voice of customer. However, although many previous studies focused on analyzing attitudes, opinions, sentiments of the text data, no work has been conducted from the perspective of customer loyalty, which is reflected by a combination of customer attitudes and behavior. In this work, we present JCLIC, Japanese Customer Loyalty Information Corpus, which is a corpus for analyzing customer loyalty information. For each review we have annotated detailed customer loyalty information which contains: *loyalty degree* that reflects loyalty level of the customer, *loyalty expression* that expresses the customer loyalty, *loyalty type* that indicates the category to which loyalty expression belongs., *reason expression* that expresses why the customer have such loyalty degree, and *reason type* that indicates the category to which reason expression belongs.. We describe our annotation scheme and annotation process, present results of an agreement study and give some statistics about the corpus we have annotated.

Keywords: Customer loyalty, Voice of customer, Opinion mining

1. Introduction

Customers voice attitudes, opinions and their experience about some brands, companies, products or services. Today voice of the customer (VOC), capturing a view of customer's behaviors, needs, and feedbacks, can be obtained through center calls, emails, questionnaire, web reviews or SNS (Subramaniam et al., 2012; Aguwa et al., 2012; Choi et al., 2013; Saeed et al., 2013). Such unstructured textual information contains valuable information for marketing analysis. On the other hand, customer loyalty has long been a topic of high interest in both academia and industry (Griffin, 2002; Asuncion et al., 2004; Scriosteanu and Popescu, 2010), and is critical for business profitability. Therefore, it is attractive to consider exploiting customer loyalty by analyzing the voice of customer.

To automatically find or track the attitudes, feelings and evaluations in texts, opinion mining and sentiment analysis have been extensively studied from different perspectives (Kushal et al., 2003; Pang and Lee, 2008; Nakagawa et al., 2008; Wang et al., 2012). There are also several corpus for opinion mining, such as MPQA Opinion Corpus version 3.0 (Deng and Wiebe, 2015), JDPA Corpus (Kessler et al., 2010) and Chinese evaluation information corpus (Wang et al., 2012). All the existing works are related to the affective and evaluative dimensions and focus on understanding customer satisfaction. However, customer loyalty is not customer satisfaction. Satisfaction is a necessary but not sufficient criterion for customer loyalty (Griffin, 2002; Ganiyu et al., 2012). We know that "very satisfied" to "satisfied" customers sometimes switch to competitors. Costumer loyalty is multi-dimensional and comprise three dimensions namely, the emotive tendency towards the brand/product/service, the evaluative tendency towards the brand/product/service, the behavioral tendency towards the brand/product/service. To our knowledge, no work has been conducted from the perspective of customer loyalty, which is reflected by a combination of customer attitudes and behaviors. There is also no corpora in this direction.

In this work we present a dedicated gold standard corpus of customer loyalty information. In order to investigate and understand how is customer loyalty and what drives loyalty in a particular market, we analyze the customer loyalty at a fine-grained level and define a 5-tuple that consists of (1) loyalty degree, (2) loyalty expression, (3) loyalty type, (4) reason expression, and (5) reason type as the basic unit of customer loyalty information. We annotate the texts of several domains (1500 reviews in 15 topics) in order to train a robust system and analyze a wide variety of customer loyalty information. To our knowledge, our corpus is the first resource of customer loyalty analysis in natural language field to date.

In the following, we will present our annotation scheme and annotation process. We will present results of an agreement study and give some statistics about the data we have annotated.

2. Customer Loyalty Information

Customer Loyalty is both an attitudinal and behavioral tendency to favor one brand over all others, whether due to satisfaction with the product or service, its convenience or performance, or simply familiarity and comfort with the brand (Griffin, 2002). Customer loyalty encourages consumers to shop more consistently, spend a greater share of wallet, and feel positive about a shopping experience, helping attract consumers to familiar brands in the face of a competitive environment.

The general purpose of this study is to identify, describe and analyze elements that have an impact on understanding customer loyalty or disloyalty. We need to investigate how is the customer loyalty degrees and types. The behavioral and attitudinal tendency of a customer is not consistent in some cases. Some customers have high evaluation and attitude but they are not repeat patronage, while some customers shop more consistently even though they are very

satisfied with the product. We need to distinguish such different loyalty types. Furthermore the factors which lead to loyalty or disloyalty need to be uncovered and understood before designing and implementing the strategies for customer retention and loyalty. Therefore we analyze the customer loyalty at a fine-grained level and define a 5-tuple that consists of (1) loyalty degree , (2) loyalty expression, (3) loyalty type, (4) reason expression, and (5) reason type as the basic unit of customer loyalty information. Each item is defined as follows and the details of the items will be described in Section 3 .

- **Loyalty degree** is the level of loyalty or disloyalty.
- **Loyalty expression** is a span of text that describes the loyalty of a customer. It can be a single word, a multi-word expression, or a sentence.
- **Loyalty type** indicates the category to which loyalty expression belongs. We define the loyalty types into 7 categories and each type has a polarity information, positive (+), neutral (0) or negative (-). which will be described in Section 3 .
- **Reason expression** is a span of text that describes the factors which lead to loyalty and disloyalty.
- **Reason type** indicates the category to which reason expression belongs. We define the reason types into 8 categories and each type has a polarity information, positive(+), neutral(0) or negative(-). which will be described in Section 3 .

3. Annotation Schema

In order to obtain consistent annotations, we provided annotators with a more detailed annotation guidelines. Here we describe the annotation schema in general.

3.1. Loyalty Degree

To understand customer loyalty one must recognize there are different degrees and types of loyalty. In order to investigate how is the customer loyalty degree, for each review, one loyalty degree is annotated.
Dick and Basu (1994) brought out the idea of relative attitudes while defining various forms of loyalty depicted below. They described loyalty as the strength of the relationship between a customer's relative attitude and repeat patronage and four dimensions had been identified: true loyalty, latent loyalty, spurious loyalty and no loyalty (Table1).

		Repeat Patronage	
		high	low
Relative Attitude	High	True Loyalty	Latent Loyalty
	low	Spurious Loyalty	No Loyalty

Table 1: Various forms of loyalty

Inspired by their idea, the loyalty degree is divided into 5 levels and is described in Table2. We annotate the loyalty degree by taking the loyal expression and loyalty types which will be described in Section 3 2 into consideration. Level 2 is the same as true loyalty in Table1, which means to show loyalty in both behavioral and attitudinal dimensions. Level 1 include both spurious Loyalty and latent Loyalty in Table1. We extended the no loyalty and disloyalty into 3 levels in order to found the important differences between loyal and disloyal customers by text mining.

Level	Description
2	True or premium loyalty
1	Loyal customer or latent loyal customer
0	Neutral customer or No loyalty
-1	Latent churn customer
-2	Disloyal customer or churn customer

Table 2: The loyalty degree

3.2. Loyalty Expression and Loyalty Types

Customer loyalty is the outcome of several factors comprising affective, behavioral and attitudinal dimensions. To clarify the scope of customer loyalty that we address in this study, we classify loyalty expressions into several categories. Such categorization is also helpful for further use of loyalty information.
Customer loyalty is reflected by a combination of attitudes and behaviors. The attitudes include how strong the psychological commitment or attachment is to the brand/service/product and emotional or feeling reactions in customer experiences. The behaviors include repeat purchasing, purchasing more and different products or services from the same company, and recommending the company to others. We define the loyalty types which include both behavioral and attitudinal aspects. In order to investigate the latent churn related information, the switching information are also annotated. The loyalty types are described in Table3.

Large Category	**Loyalty Types**
Attitude	Attachment
	Emotion
Behavior	Experience
	Repurchase
	Recommendation
Switching	Switching cost
	Competitor

Table 3: Types of loyalty expression

For each review, multiple loyalty expressions and the corresponding loyalty types should be annotated. Each type has a polarity information, positive (+), neutral (0) or negative (-). Positive is the loyal attitudes or behaviors, negative is the disloyal attitudes or behaviors. Here we give some examples to explain the loyalty expressions, the loyalty types and polarities. We use underline to show review topics and boldface for loyalty expression.

- Attachment +/0/-: an expression that expresses the condition of being attached to a product or a service.

e.g. Product A is **my favorite one**. (Attachment+)

- Emotion +/0/-: an expression that expresses customer feelings or emotions.
  e.g. I **felt anger at** the clerk's rudeness in that shoe store. (Emotion-)
- Experience +/0/-: an expression that describes the experience of shopping or utilizing, such as how long or how often a customer has used a service .
  e.g. I **have been a member of this gym for several years** . (Experience+)
- Repurchase +/0/-: an expression that express repeat purchasing, purchasing more and different products from the same company.
  e.g. I will **buy other skin-care products** of this brand . (Repurchase+)
- Recommendation +/0/-: an expression that describes how customers rave about a product and promote it to their friends.
  e.g. I **don't recommend** this hotel at all. (Recommendation-)
- Switching cost +/0/-: an expression that shows the economic, psychological and physical costs to switch to the competitors.
  e.g. **It costs a lot to switch to** other mobile company. (Switching cost+)
- Competitor +/0/-: an expression that expresses the comparison with the competitor's product or temptation of the competitors.
  e.g. **The competitor's product is more convenient than** this healthcare wearable watch. (Competitor-)

Note that for some cases, our loyalty polarity may differ from the sentiment polarity. For examples, *The competitor's product is more convenient than this healthcare wearable watch*. Although the entire sentence is not negative, the polarity of loyalty expression, which implies the risk of churn, is negative. From this point of view, we consider the polarity in the connection with our topics and loyalty.

3.3. Reason Expression and Types

McKinsey & Company found(Griffin, 2002) :

- customers who have major problems but do not complain about them have a repurchase intention rate of about 9%
- those who do complain, regardless of the outcome, have a repurchase rate of about 19%
- customers who have their complain resolved have a repurchase intention rate of 54%
- customers who have complaints quickly resolved have a repurchase intention rate of 82%

Therefore in order to develop and sustain loyalty among the customers, it is important to find out what drives loyalty or disloyalty in a particular market. The factors which lead to loyalty need to be uncovered and understood. For each review, multiple reason expressions and the corresponding reason types should be annotated. Each type has a polarity information, positive (+), neutral (0) or negative (-). Positive is the loyal factors, negative is the disloyal factors. The reason types is described in Table4.

Large Category	**Reason Types**
The product or service	Quality
	Function
	Price
Relating factors	Convenience
	Staff
	Related services
Other	External influences
	Others

Table 4: Reason types for reason expression

Here we give some examples to explain the reason expressions and the reason types. We use underline to show review topics and boldface for reason expression.

- Quality +/0/-: an expression that expresses the factors related to the quality.
  e.g. Camera X **broke just three days after I bought it**. (Quality-)
- Function +/0/-: an expression that is related to the function of a product or substantial service menu.
  e.g. Product A **can also perform a specific medical function**. (Function+)
- Price +/0/-: an expression that express price related factors.
  e.g. **It is too expensive** to go to the restaurant frequently. (Price-)
- Convenience +/0/-: an expression that expresses convenience, such as the location of the store
  e.g. The restaurant is **only five minutes away from my home**. (Convenience+)
- Staff +/0/-: an expression that describes the factors related to the attitude, skill level of staffs.
  e.g.The service desk staff in the gym **was very unfriendly** . (Staff-)
- Related services +/0/-: an expression that shows the reasons about aftercare, delivery and so on.
  e.g. **It took too much time for the delivery** since my order was placed in the web store. (Related services-)
- External influences +/0/-: an expression that expresses the influence by SNS or friends .
  e.g. I book this hotel **because of the review from Tripadvisor** . (External influences+)
- Others +/0/-: an expression that is loyalty factors which are not included in the above types.
  e.g. **Other users of the fitness club were too annoying** (Others-)

4. Corpus Construction

4.1. Data Collection

To train a robust system and analyze a wide variety of customer loyalty information, we constructed an customer loyalty information corpus which consists of Japanese reviews extracted from web pages of wide range of topics and

Mobile virtual network operator, Fitness club, Restaurant, English language school, Travel agents, Streaming services, Insurance services, Cloud services, Loan services, Beauty salon, Glasses, Wearable sensors, Laptops, Hospitals and dental clinics, Customer to Customer Websites, Airlines,Hair dryer, Watch, Hotels, Amazon Prime.

Table 5: Topics of Japanese loyalty information corpus

styles. We chose 20 topics which relate to products or services we use in daily life. The topics are shown in Table 5 (We only finished the annotation of the previous 15 topics). The followings are the steps for the corpus data collection:
(1) Use the topic as the keyword and search documents using a Web search engine.
(2) Collect HTML files of web pages from the retrieval results for each topic. Specifically, the pages in the retrieval results from forum sites, review sites are collected. We attempted to collected the reviews from different web paged. In this way, the corpus can cover different writing styles and reflect more diverse perspectives.
(3) Randomly choose candidate reviews that are related to topic keywords from the above files. For each topic, we randomly collected 100 reviews, which may be in one sentences or in several sentences.

4.2. Annotation Procedure

The followings are the steps for the construction of Japanese Customer Loyalty Information Corpus:

- Step1: Trained annotators checked whether the review contained any loyalty expressions or not. If the review contained loyalty expressions, the annotator annotated the text spans of the loyalty expressions, the types and the polarities of the loyalty expressions.
- Step2: The annotators checked whether the review contained any reason expressions or not. If the review contained reason expressions, the annotator annotated the text spans of the reason expressions, the types and the polarities of the reason expressions.
- Step3: Trained annotators judged the loyalty level of a whole review and assigned a level label from a predefined set.

In some cases, one review may contain multiple loyalty expressions or multiple reason expressions, and then multiple loyalty information sets must be annotated. That is to say, all the customer loyalty information was annotated for each review. It took 360 man-hours to construct the corpus in 15 topics.
Table6. shows one example of annotation result of the following review.
The original review: *My daughter is going to this English school twice a week. After she studied in the school for half a year, her Katakana English became to the natural accent. The teachers always find a way to make the class funny and interesting. Going to the English school is very enjoyable for her. I think this school is the first choice for all beginners.*

Loyalty expression	Loyalty type
is going to this English school twice a week	Experience+
Going to the English school is very enjoyable	Emotion+
this school is the first choice for all beginners.	Recommendation+
Reason expression	**Reason type**
her Katakana English became to the natural accent	Quality+
The teachers always find a way to make the class funny and interesting.	Staff+
Loyalty degree	
2	

Table 6: One example of annotation result

5. Corpus Analysis

5.1. Statistics

We have finished the annotation of 1500 reviews in 15 topics, including 7,780 sentence. Statistics about our complete corpus can be found in Table7.

Topics	15
Review	1,500
Loyalty reviews	831
Disloyalty reviews	570
Neutral reviews	99
Sentence	7,780
Loyalty expression	3,922
Reason expression	4,882

Table 7: Statistics of the corpus

As for some detailed statistics, the information of loyal expressions and types is shown in Table8 and the information of reason expressions and types is shown in Table9. The result shows that emotion type is the most in loyalty expressions. The number of loyalty expressions which show attitudinal tendency is 2,232, while the number of loyalty expressions which show behavioral tendency is 1,222. The remaining 468 loyalty expressions are about switching information. The loyalty expressions reflected by customers' attitudes covers the majority. As for the reason expressions, the reasons related to "staff" are the main factor and the following is the quality.

5.2. Inter-Annotator Agreement

We first let both annotators independently annotate the same 2 reviews in English school and beauty salon topics, on which we compute the inter-annotator agreement. After that, each of the annotators annotated their own set of 13 reviews. We take one of the annotator's annotation as a

Loyalty Types	Numbers
Attachment	406
Emotion	1,826
Experience	693
Repurchase	308
Recommendation	221
Switching cost	43
Competitor	425

Table 8: Statistics of loyalty expressions and types

Reason Types	Numbers
Quality	1,141
Function	936
Price	504
Convenience	294
Staff	1,326
Related services	340
External influences	120
Others	221

Table 9: Statistics of reason expressions and types

gold standard and the other annotator's annotation as a system output, and then use the following measures to evaluate inter-annotator agreement:

- **Recall**: ratio of correctly extracted loyalty/reason expressions to the number of expressions in the gold standard.
- **Precision**: ratio of correctly extracted loyalty/reason expressions to the number of expressions in system output.
- **F-measure**: harmonic mean of recall and precision.
- **Accuracy**: ratio of the number of correct system output to the number in the gold standard.

For the annotation of text spans where each annotator individually picks some words from the review instead of assigning a label from a predefined set, we follow Breck et al. (2007) and use the following partial match criteria to calculate the recalls, precisions and F-measures of the loyalty/reason expressions. The partial match is defined as follows:

- **Partial match**: extracted expression is regarded as correct if it overlaps the gold standard's one.

The inter-annotator agreement is shown in Table10. With the only exception of reason type accuracy of beauty salon, for the annotation of assigning a label from a predefined set, such as loyalty degree, loyalty type and reason type, we have achieved a reliable accuracy, ranging from 80% to 90%. For the annotation of extract some words from the review, the F-measures are ranging from 75% to 80%. The detailed results from the agreement study will be used to refine the annotation guidelines.

	English School	Beauty Salon
Loyalty Degree Accuracy	0.85	0.96
Loyalty Expression Recall	0.66	0.67
Loyalty Expression Precision	0.85	0.86
Loyalty Expression F-measure	0.74	0.75
Loyalty Type Accuracy	0.85	0.85
Reason Expression Recall	0.63	0.93
Reason Expression Precision	0.90	0.70
Reason Expression F-measure	0.74	0.80
Reason Type Accuracy	0.83	0.57

Table 10: Inter-annotator agreement

6. Conclusion

In this work, we presented a dedicated gold standard corpus of customer loyalty information in Japanese. For each review we have annotated detailed information about the customer loyalty. We have described our annotation process and given an overview of our annotation guidelines. In total, we have annotated 7780 sentences from reviews in 15 topics. We will extend our corpus and build supervised models for determining the loyalty level, recognizing loyalty expressions and reason expressions and classifying the loyalty types and reason types in further.

7. Bibliographical References

Aguwa, C. C., Monplaisir, L., and Turgut, O. (2012). Voice of the customer: Customer satisfaction ratio based analysis. *Expert Systems with Applications*, 39:10112–10119.

Asuncion, B., D.J.Martin, and Quintana, A. (2004). Model of customer loyalty in the retail banking market. *European Journal of Marketing*, 38:253–275.

Breck, E., Choi, Y., and Cardie, C. (2007). Identifying expressions of opinion in context. In *Proceedings-IJCAI-2007*, pages 2683–2688.

Choi, C., J. Lee, G. P., Na, J., and Wan-SupCho. (2013). Voice of customer analysis for internet shopping malls. *International Journal of Smart Home*, 7:291–304.

Deng, L. and Wiebe, J. (2015). Mpqa 3.0: Entity/event-level sentiment corpus. In *2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT'15)*, Colorado, USA.

Dick, A. S.and Basu, K. (1994). Customer loyalty: Toward an integrated conceptual framework. *Journal of the Academy of Marketing Science*, 22:99–113.

Ganiyu, R. A., Uche, I. I., and Elizabeth, A. O. (2012). Is customer satisfaction an indicator of customer loyalty? *Australian Journal of Business and Management Research*, 2(7):14–20.

Griffin, J. (2002). *Customer Loyalty: How to Earn It, How to Keep It*. Jossey-Bass, 2nd edition.

Kessler, J. S., Eckert, M., Clark, L., and Nicolov, N. (2010). The 2010 ICWSM JDPA sentment corpus for the automotive domain. In *4th International AAAI Conference on Weblogs and Social Media Data Workshop Challenge (ICWSM-DWC'10)*.

Kushal, D., Lawrence, S., and Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and se-

mantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*.

Nakagawa, T., Kawada, T., Inui, K., and Kurohashi, S. (2008). Extracting subjective and objective evaluative expressions from the web. In *Proceedings of the 2nd International Symposium on Universal Communication*, pages 251–258.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

Saeed, R., Lodhi, R. N., Munir, J., Riaz, S., Dustgeer, F., and Sami, A. (2013). The impact of voice of customer on new product development. *World Applied Sciences Journal*, 24:1255–1260.

Scriosteanu, A. and Popescu, D. (2010). Customer loyalty - the major goal of customer relationship management. *Annals of University of Craiova - Economic Sciences Series*, 3:294–300.

Subramaniam, L. V., Faruquie, T. A., Ikbal, S., Godbole, S., and Mohania, M. K. (2012). Business intelligence from voice of customer. In *Proceedings of IEEE International Conference on Data Engineering (ICDE'09)*, Shanghai, China, March. IEEE.

Wang, Y., Junichi, Kazama, Kawada, T., and Torisawa, K. (2012). Chinese evaluative information analysis. In *Proceedings of the 24rd International Conference on Computational Linguistics(COLING'12)*, pages 2773–2788.

FooTweets: A Bilingual Parallel Corpus of World Cup Tweets

Henny Sluyter-Gäthje[1], Pintu Lohar[1], Haithem Afli[1-2] and Andy Way[1]
[1]ADAPT Centre
School of Computing
Dublin City University
Dublin, Ireland
{FirstName.LastName}@adaptcentre.ie
[2] Cork Institute of Technology
haithem.afli@cit.ie

Abstract

The way information spreads through society has changed significantly over the past decade with the advent of online social networking. Twitter, one of the most widely used social networking websites, is known as the real-time, public microblogging network where news breaks first. Most users love it for its iconic 140-character limitation and unfiltered feed that show them news and opinions in the form of tweets. Tweets are usually multilingual in nature and of varying quality. However, machine translation (MT) of twitter data is a challenging task especially due to the following two reasons: (i) tweets are informal in nature (i.e., violates linguistic norms), and (ii) parallel resource for twitter data is scarcely available on the Internet. In this paper, we develop *FooTweets*, a first parallel corpus of tweets for English–German language pair. We extract 4,000 English tweets from the FIFA 2014 world cup and manually translate them into German with a special focus on the informal nature of the tweets. In addition to this, we also annotate sentiment scores between 0 and 1 to all the tweets depending upon the degree of sentiment associated with them. This data has recently been used to build sentiment translation engines and an extensive evaluation revealed that such a resource is very useful in machine translation of user generated content.

Keywords: tweets, parallel data, sentiment translation

1. Introduction

Due to the continuously developing Internet technology, there are countless digital media and social networking sites, all of which have a unique characteristic and purpose. Social media has evolved from being cyber world geek buzz to a massive platform for businesses, entrepreneurs, professionals and organizations that seek greater recognition and identification at a very economical price. However, business information sharing is not the only aspect of web services, e.g. people from all over the world with different cultural backgrounds stay connected and communicate via widely used social networking websites such as Twitter, Instagram, Facebook etc. Twitter is an extremely useful social networking tool for different event, business and news organisation that want to reach out to people (and are ready for a reply). Recently, Twitter has gained massive popularity and the number of Twitter users has increased significantly during the last few years.

However, Twitter users are often encouraged to write informal texts due to the 140-character limitation.[1] They follow many other users who tweet in different languages so, the tweets are multilingual in nature and often need to be translated from a specific (source-) language to the (target-) language of choice. In addition, tweets include spelling errors, hashtags, user handles, retweets, short forms etc. As a result, translation of such noisy texts becomes a difficult task. To the best of our knowledge, bilingual parallel corpora of tweets are hardly available on the Internet. The development of such corpus is therefore extremely important for MT of such noisy user-generated content (UGC). In this work, we extract 4,000 English tweets from the FIFA World Cup 2014 and apply the following steps in order to build the first bilingual parallel tweet corpus for the English–German language pair: (i) we translate all the English tweets into German with some translation guidelines (discussed in detail in Section 3), and (ii) each tweet is assigned a sentiment score between 0 and 1 depending upon the degree of emotion associated with it.

The remainder of this paper is organised as follows. Section 2 provides a brief literature survey of this field. In Section 3, we discuss some translation guidelines followed during the corpus development. The sentiment-annotation procedure is explained in Section 4. In Section 5, we briefly discuss the usefulness of this corpus with an example of our recent work on sentiment translation system using a the sentiment classification approach (Lohar et al. (2017)). Finally, we conclude and point out some possible future work in Section 6.

2. Related work

Parallel data for Twitter is scarcely available on the Internet. One of the available corpora is "microtopia", a parallel corpus of microblogs (Ling et al. (2014)). Recently, TweetMT (Vicente et al. (2016)) has been introduced as a parallel corpus of tweets in four language pairs that combine five languages (Spanish from/to Basque, Catalan, Galician and Portuguese). Ling et al. (2013) present a framework to crawl parallel data from microblogs in order to find parallel resources from single posts, with translations of the same sentence in two languages. Hajjem et al. (2013) create an Arabic–French comparable corpus, the first comparable corpus collected from Twitter. Despite this apparent lack of data, some work has been carried out in the area of tweet translation. Kaufmann and Kalita (2010) combine a statistical machine translation (SMT) system with a preprocessor and successfully remove the majority of noise from a tweet, which results in increasing its readability in the target lan-

[1] recently expanded to 280

guage. The work in Gotti et al. (2013) reports experimental results obtained from translating Twitter feeds published by agencies and organizations, using an SMT system. They mine parallel web pages linked from the URLs contained in English–French pairs of tweets in order to create the tuning and training material. Jiang et al. (2012) propose strategies to handle shortforms, acronyms, typos, punctuation errors, non-dictionary slang, wordplay, censor avoidance and emoticons.

3. Translation guidelines

This section describes the main guidelines we followed during the manual translation process. As Tweets may contain shortforms, typos, wordplays etc., all of which are often deliberately introduced especially due to the character limitation. Such characteristics of tweets pose challenges in translations into another language. We therefore place an emphasis on the following three strategies while translating the tweets: (i) informal to informal translation, (ii) informal to formal translation, and (iii) sentiment preservation. Following sections describe each of the main guidelines in detail.

3.1. Informal to informal translation

The tweets often contain informal texts such as short forms, stylistic effects etc. For example, the English tweet "GOAAAAL ♡ ♥ ♡ ♥" implies that the Twitter user expresses a positive emotion and deliberately introduces a stylistic effect (that is, the repetition of "A" in the word "GOAL") while writing the tweet. We place a strong emphasis on such behaviour by the users and translate the tweets accordingly into the target language, the result of which is essentially informal in nature as well. The above example tweet is therefore translated into German as "TOOOOR ♡ ♥ ♡ ♥" in order to retain the same degree of sentiment in the target language.

3.2. Informal to formal translation

As mentioned earlier in Section 3., the Twitter users are encouraged to use short forms at word or phrase level in order to fit all the contents within the specified characters limitation. Accordingly, most of the time they intentionally make acronyms for a group of words or a phrase. For example, nowadays it has become more popular to write *lol* instead of writing *laughed out loud*. In a similar vein, the phrase "going to" is often shortened to *gonna*. In addition to this, Twitter users often create short forms of individual words by omitting one or more characters from them. For example, the word "*you*" is contracted to *u* by removing the letters "*y*" and "*o*" so that it sounds almost the same as the original word. Such behaviour challenges in the translation process. It is therefore necessary to scan through the tweets carefully and identify such *noisy* content. Once these elements are found, we translate them with special attention so that their translations are formal. Therefore, the informal "u" is translated as formal "dir" in German. Such *informal-to-formal* translation is definitely useful in building MT engines in order to make the translation of tweets easier, which is otherwise a difficult task.

3.3. Sentiment preservation

As many of the tweets convey a certain degree of sentiment, they draw our special attention during the translation process maintaining the original sentiment. In addition, the deliberate stylistic effects applied on the tweets encourage us to perform the translation accordingly. For example, the tweet "YEEEEEESSSS!!!" contains a higher level of positive sentiment than the tweet "YES!!!". Considering this phenomenon, we translate the tweets based not only on their literal meaning but also the way they are expressed. The above example, therefore, can be translated as "JAAAAAAA!!!" into German.

4. Sentiment score annotation

Once the translation is performed on all the 4,000 English tweets, we manually assign sentiment scores (from 0 to 1)to each of them. However, as our intention was to categorise the tweets into three different classes namely *negative*, *neutral* and *positive*, we categorise them using the following criteria: (i) *negative*, if sentiment score < 0.5, (ii) *neutral*, if sentiment score $= 0.5$ and (iii) *positive*, if sentiment score > 0.5 .

With the above criteria for categorisation, it is perfectly valid to assign any sentiment score from 0 to less than 0.5 for the negative-sentimented tweets as it does not affect the sentiment class as long as the score remains in this range. In a similar manner, the tweets that convey positive sentiment can be assigned any score that is greater than 0.5 but is less than or equal to 1. With this consideration, we decided to use the following three different values for sentiment scores to make this task easier: (i) 0.3 for negative tweets, (ii) 0.5 for neutral tweets, and (iii) 0.7 for positive tweets.

Note that this categorisation technique is valid only if there are only these three sentiment classes. It may be required to decide on other values if we include other sentiment classes such as *strong negative*, *strong positive* etc. However it is not required in this case as our main focus is not on the exact sentiment score but only on these three sentiment classes.

Table 1 shows some example translations along with the sentiment scores assigned to them. There are three different values for sentiment scores (0.3, 0.5 and 0.7) that depend upon the sentiment class the tweet belongs to. Example 1 in this table is a negative-sentimented tweet and hence is assigned a score of 0.3 according to our criteria of sentiment annotation. We can see that "I am" is shortened as "Im" but it is translated as "bin ich" in German, which is essentially informal-to-formal translation. Similarly in example 2 (positive-sentimented tweet), the phrase "going to" is informally written as "gonna". Note that the first segment is "not gonna lie..." where the pronoun "I" is missing but it is obvious from the context. This informal segment is translated into "ich werde nicht lügen" which is formal in German. Subsequently the word "next" is contracted to "nxt" (with a character omission) but translated as "nächstes", a formal word in German. However, sometimes users write almost-formal texts in their tweets. For example, the item number 4 in Table 1 is an example where all the words are written correctly. This tweet is grammatically correct except that the verb "is" is missing (i.e., it should be "Luis

Ex.	English tweet	German translation	Sentiment score
1	now that Neymar cant play, Im so nervous for Brazil	jetzt, da Neymar nicht spielen kann, bin ich so aufgeregt wegen Brasilien	0.3
2	not gonna lie... Germanys national anthem is awesome.	ich werde nicht lügen... Deutschlands Nationalhymne ist genial.	0.7
3	sorry Brazil.... dont host a worldcup nxt time	tut mir leid Brasilien.... veranstalte nächstes Mal keine Weltmeisterschaft	0.3
4	Luis Suárez suspended for nine matches and banned for four months from any football-related activity	Luis Suárez für neun Spiele gesperrt und vier Monate von jeder Fussballtätigkeit ausgeschlossen	0.3
5	shame on Messi.. wack Argentine team.. no clear cut chance created at all...	Schande über Messi.. Schwaches argentinisches Team.. überhaupt keine klaren Chancen erspielt...	0.3
6	just making sure its still there!!!	Stelle nur sicher, dass es immer noch da ist!!!	0.5
7	Yesssss!!!!!!!!!!!! Golazo!!!!!!!	Jaaaaa!!!!!!!!!!!! Golazo!!!!!	0.7

Table 1: Tweets and their translations along with the sentiment scores

Suárez *is* suspended..."). It can be observed that the example 6 is a tweet that can be considered as having neutral sentiment as it is very difficult to associate either negative or positive sentiment with it. Finally, example 7 contains stylistic effect as the word "Yes" is deliberately written as "Yesssss" in order to express the positive emotion of the user. As expected, this word is translated as "Jaaaaa" into German to retain its positivity during the translation process. This is an example of *informal-to-informal* translation which we specially consider for sentiment preservation. Upon completing the translation of all the tweets, we obtain a distribution of the three sentiment classes. Table 2 shows the distribution of negative, neutral and positive tweet pairs.

Negative	Neutral	Positive	Total
1,019	1,408	1,573	4,000

Table 2: Data distribution

5. Usefulness of the corpus

According to the best of our knowledge, the data we developed in this work is the first ever published parallel Twitter corpus for English–German language pair. Although it is a very small-sized corpus having only $4,000$ tweet pairs, it can play a significant role in building MT engines as it contains different levels of informal parallel texts. It is, therefore, expected that the MT models built from it are capable of translating many informal texts although not everything, as it is extremely difficult to cover all types of variations of a word. In addition, annotating sentiment scores to all the tweets opens up a number of opportunities in future for sentiment analysis of the tweets as well. This corpus is useful in translating tweets and at the same time preserving the sentiment during translation by building a suite of sentiment translation engines (Lohar et al. (2017)). In the following section, we briefly discuss this work in order to highlight the importance of our corpus.

5.1. Tweet translation and sentiment preservation

The work in Lohar et al. (2017) presents a sentiment translation system based on a sentiment classification approach. The idea is to divide the English–German parallel twitter corpus into three different parts based on the following characteristics of the tweets: (i) negative corpus with sentiment score ≤ 0.4, (ii) neutral corpus with sentiment score $\simeq 0.5$, and (iii) positive corpus with sentiment score ≥ 0.6 . Afterwards, three different translation models are built from each of the above parallel corpus set and referred to as negative, neutral and positive translation model, respectively. The objective to translate the tweets using sentiment-specific translation models and at the same time preserve the sentiment during translation. They translate German tweets into English in order to be able to use the sentiment analysis tool in English especially designed for tweets (Afli et al. (2017)).

5.1.1. Experiments

As the corpus is small in size, a very small subset of 50 tweets per sentiment (negative, neutral and positive) is held out for tuning and testing purposes in order to maintain as large an amount as possible for training purpose. The remaining $3,700$ tweet pairs are considered as the training data and are similarly divided into negative, neutral and positive tweet pairs. The translation models are built using the Moses SMT tool (Koehn et al. (2007)) using Giza++ (Och and Ney (2003)) for word and phrase alignment. Afterwards, the models are tuned using minimum error rate training (Och (2003)). The additional resources used are English–German parallel Flickr data[2] and "News-Commentary (News)" data[3] in order to build larger MT engines. The evaluation process consists of two different types of measurements: (i) MT quality and (ii) sentiment preservation. For MT evaluation, the automatic evaluation metrics BLEU (Papineni et al. (2002)), METEOR (Denkowski and Lavie (2014)) and TER (Snover et al.

[2] http://www.statmt.org/wmt16/multimodal-task.html\#task1

[3] http://data.statmt.org/wmt16/translation-task/training-parallel-nc-v11.tgz

Translation model	Sent_Clas	BLEU	METEOR	TER	Sent_Pres
Twitter	✓	48.2	59.4	34.2	72.66%
Twitter (Baseline)	×	50.3	60.9	31.9	66.66%
Twitter + Flickr + News	✓	50.3	62.3	31.0	**75.33%**
Twitter + Flickr + News	×	**52.0**	**63.4**	**30.1**	73.33%
Twitter (wrong MT engine)	✓	46.9	57.9	35.4	47.33%

Table 3: Experimental evaluation with data concatenation

Reference	**Sentiment translation system**	**Baseline**
Bosnia and Herzegovina really got f*** over man	Bosnia and Herzegowina eliminated echt demolished	Bosnia and Herzegovina were a abgezogen
when USA lost, but were still moving onto the next round	even if USA today we in the next round	could usa loses the next round
Brazil 5 WorldCup championship Argentine 2 WorldCup championship so Ill go with Brazil	Brazil 5 time world champion Argentina 2 time world champion so Im for Brazil	Brazil 5 time world champions Argentina 2 time world champions so for Brazil

Table 4: Examples where sentiment is altered by the Baseline system

Reference	**Right MT engine**	**Wrong MT engine**
little break on the #WorldCup for an amazing #Wimbledon final!	small Pause from the #WorldCup for a amazing #Wimbledon final!	kleine Pause of the #WorldCup for a erstaunliches #Wimbledon final!
yes !!!!!	yes !!!!!	so !!!!!
a bit boring...	a little boring ...	some was ...

Table 5: Comparison between sentiment polarities using the right and wrong MT engine

(2006)) are used. In order to measure the sentiment preservation, the fraction of all of the source-language tweets in the test set that remain under the same sentiment class after translation is calculated.

In addition, the authors also performed a random test by translating the (i) negative tweets by the positive model, (ii) neutral tweets by the negative model, and (iii) positive tweets by the neutral model. The aim was to arbitrarily choose one of any of the model-selection combinations so that the tweets with a specific sentiment class is translated by the translation model with a different sentiment, in order to see the effects on translation quality and sentiment preservation. The resultant translation system was termed as the "Wrong MT engine" whereas the "Right MT engine" was their sentiment translation system.

5.1.2. Results

The results are summarised in Table 3. It can be observed that when only the Twitter data is used, better BLEU, METEOR and TER scores are obtained without using the sentiment classification ("Sent_Clas") approach ('Twitter (Baseline)'). In contrast, the sentiment preservation score ("Sent_Pres") is higher when using the sentiment classification (72.66%) method whereas switching it off causes the score to be reduced to 66.66%. The best BLEU, METEOR and TER scores of 52.0, 63.4 and 30.1, respectively, are obtained with the concatenation of additional Flickr and News data. The sentiment classification approach still manages to increase the sentiment better in this case too (from 73.33% to 75.33%). The last row in this table shows that the wrong MT engines produce the lowest scores both in terms of MT quality and sentiment preservation.

Table 4 shows how the sentiment translation system is capable of preserving the sentiment in the target language whereas the Baseline alters the sentiment during translation. Finally, Table 5 highlights the fact that the sentiment polarity is changed by using the wrong MT engines. This is a very interesting result which suggests that it is essential to translate a specific text by using the translation system that is built from the data whose sentiment matches the input text.

6. Conclusion and Future Work

In this work, we developed the first corpus (*FooTweets*) of English–German parallel tweets. We followed some translation guidelines that are very important for translating such noisy texts. In addition to this, we manually annotated the sentiment scores for all the 4,000 tweets in order to facilitate the task of sentiment analysis. Initially we restricted the sentiment classes to only negative, neutral and positive. However, in future, it can easily be extended with some other sentiment classes such as strong negative, strong positive etc. Although these processes require a significant amount of time, in future, we would like to increase the size of our corpus as it will help improve the quality of the Twitter translation engines. We have made this corpus publicly available for access(*FooTweets*[4]). We hope that this parallel resource will be helpful for the researchers who are interested in the area of MT and sentiment translation systems. It may also open up a number of opportunities for future

[4] Available at: `https://github.com/HAfli/FooTweets_Corpus`

work for other natural language processing tasks related to UGC.

Acknowledgments

The ADAPT Centre for Digital Content Technology at Dublin City University is funded under the Science Foundation Ireland Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

Haithem Afli, Sorcha McGuire, and Andy Way. Sentiment translation for low resourced languages: Experiments on irish general election tweets. In *18th International Conference on Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary, 2017.

Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, USA, 2014.

Fabrizio Gotti, Philippe Langlais, and Atefeh Farzindar. Translating government agencies' tweet feeds: Specificities, problems and (a few) solutions. In *Proceedings of the Workshop on Language Analysis in Social Media*, pages 80–89, 2013.

Malek Hajjem, Maroua Trabelsi, and Chiraz Latiri. Building comparable corpora from social networks. In *Proceedings of the 7th Workshop on Building and Using Comparable Corpora*, pages 11–16, Reykjavik, Iceland, 2013.

Jie Jiang, Andy Way, and Rejwanul Haque. Translating user-generated content in the social networking space. In *Proceedings of the Tenth Biennial Conference of the Association for Machine Translation in the Americas*, pages 1–9, San Diego, USA, 2012.

Max Kaufmann and Jugal Kalita. Syntactic normalization of Twitter messages. In *Proceedings of the 8th International Conference on Natural Language Processing*, pages 149–158, Kharagpur, India, 2010.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180, Prague, Czech Republic, 2007.

Wang Ling, Guang Xiang, Chris Dyer, Alan Black, and Isabel Trancoso. Microblogs as parallel corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 176–186, Sofia, Bulgaria, 2013.

Wang Ling, Luis Marujo, Chris Dyer, Alan Black, and Isabel Trancoso. Crowdsourcing high-quality parallel data extraction from twitter. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 426–436, Baltimore, USA, 2014.

Pintu Lohar, Haithem Afli, and Andy Way. Maintaining sentiment polarity in translation of user-generated content. *The Prague Bulletin of Mathematical Linguistics*, 108(1):73–84, 2017.

Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, 2003.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51, March 2003. ISSN 0891-2017.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, 2002.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231, Massachusetts, USA, 2006.

Iñaki San Vicente, Iñaki Alegria, Cristina España-Bonet, Pablo Gamallo, Hugo Gonçalo Oliveira, Eva Martínez Garcia, Antonio Toral, Arkaitz Zubiaga, and Nora Aranberri. Tweetmt: A parallel microblog corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 2936–2941, Portorož, Slovenia, 2016.

The SSIX Corpora: Three Gold Standard Corpora for Sentiment Analysis in English, Spanish and German Financial Microblogs

Thomas Gaillat, Manel Zarrouk, André Freitas, Brian Davis
Insight Centre for Data Analytics NUIG, Insight Centre for Data Analytics NUIG, University of Passau, Maynooth University
Galway Ireland, Galway Ireland, Passau Germany, Maynooth Ireland
{thomas.gaillat, manel.zarrouk}@insight-centre.org
andre.freitas@uni-passau.de
brian.davis@mu.ie

Abstract

This paper introduces the three SSIX corpora for sentiment analysis. These corpora address the need to provide annotated data for supervised learning methods. They focus on stock-market related messages extracted from two financial microblog platforms, i.e., StockTwits and Twitter. In total they include 2,886 messages with opinion targets. These messages are provided with polarity annotation set on a continuous scale by three or four experts in each language. The annotation information identifies the targets with a sentiment score. The annotation process includes manual annotation verified and consolidated by financial experts. The creation of the annotated corpora took into account principled sampling strategies as well as inter-annotator agreement before consolidation in order to maximize data quality.

Keywords: Sentiment Analysis, Opinion, Corpus, Finance, Stock-market, Microblogs, Polarity Annotation

1. Introduction

The objective of this paper is to report on the creation of three Gold Standard (GS) corpora in the domain of Sentiment Analysis (SA) related to financial microblogs. The purpose is to create three annotated corpora which support the quantification of the polarity of an opinion with regard to target named entities (e.g. a stock or a company). The produced gold standards represent sentiment values in a fine-grained fashion: (i) by identifying the polarity in relation to specific target entities and (ii) by using a continuous polarity scale. The corpora are in English, German and Spanish.

Today's ubiquity of social media services in people's lives has lead to the daily posting of vast amounts of data from users in many languages. Many of these postings bear a common characteristic that is of holding judgment. Compiling opinions has become a major stake for organisations and people seeking to have a clear view of opinions regarding specific entities of interest. In a highly competitive and volatile domain such as finance, acquiring an accurate vision of operators' opinions at scale may give a competitive edge in sell-and-buy decisions. Microblogs posted on social media such as Twitter [1] or StockTwits[2] are a central data source to understand the market perception.

Research in the area of Sentiment Analysis (SA) can help address this need but it faces two kinds of issues. Firstly, SA is domain dependent. Much effort has been spent in the field to assign sentiments to various types of texts. Initially focused on product reviews the research has moved to other domains and other types of texts such as microblogs. SA results show that different topics yield differences in SA accuracy (Liu, 2012, p. 36). While SA models for product reviews may already have achieved high classification accuracy, other types of texts might still present specific linguistic analysis and modeling challenges. In this respect, financial microblogs display a number of linguistic properties that are domain specific and that makes their analysis complex. Texts are short and include many references to concepts that have a specific terminology. Consequently, SA methods must be adapted to the syntactic and semantic features of financial microblogs. The second kind of issue for SA is that of multilingualism. Many SA methods rely on surpervised machine learning strategies. As such they require annotated data sets for model training in the target languages. Few annotated resources exist in German and Spanish in the domain of microblogs and even less in the financial domain.

To the best of our knowledge, there is no resource available publicly that is dedicated to such synthetic messages with multiple target entities and languages. To bridge the gap, we built three Gold Standard corpora dedicated to financial microblogs. The corpora are in three different languages (English, Spanish and German) and sizes. After extraction, the data were manually annotated by several independent expert annotators. Annotation comes in the form of a score showing positive or negative gradual degrees of opinions about entities included in sentences. Possible exploitations of the corpora include tasks such as construction and evaluation of SA models and linguistic analysis of the investment domain.

The rest of this paper is organized as follows. In Section 2., we cover existing literature in annotated corpora for SA and finance. Section 3. details the data collection method put in place to ensure representative samples. Section 4. describes the annotation process and quality analysis results are presented in Section 5.. Finally, we conclude in Section 6..

2. Related work

A large body of literature exists in the domain of Sentiment Analysis. Since our problem was to identify multilingual

[1] https://twitter.com/
[2] https://stocktwits.com/

financial resources for use in supervised learning methods, we narrow our review to annotated corpora of the financial domain. Work on financial corpora has a long history, starting with the Penn Treebank corpus (Marcus et al., 1993) which includes Wall Street Journal articles in American English. More recent work includes corpora targeting various types of texts and different languages.

A number of relevant financial corpora exists in English. (O'Hare et al., 2009) analysed financial blogs. To do so, they developed a corpus of financial blogs. The annotation scheme targets 500 specific companies and is applied at document and paragraph level, not sentence level. Annotators successively used a 3 and a 2 point polarity scale of sentiment. An annotation tool was used by seven trained raters who annotated 979 documents. Inter-rater agreement shows a 0.712 Kappa for the three point scale and perfect agreement for the binary scale. The annotated corpus made up of 1,691 document-level annotations is subsequently used in a sentiment polarity classification approach. In (Malo et al., 2013), the authors trained classifiers to conduct sentence-level analysis of financial news sentiments (not microblogs). They used a human-annotated phrasebank with close to 5,000 sentences collected across a number of financial news sources. The annotation scheme is designed with a 3 point scale polarity. Three trained raters annotated the corpus and pairwise Kappa values ranged from 0.611 to 0.886. Majority vote is used to consolidate the Gold Standard. (Takala et al., 2014) focused on annotating a corpus of newswire texts categorised in ten different topics in economics. They addressed the need to have sentence level annotation to determine the aggregated sentiment for a given topic at document-level. In total, 297 texts were annotated both at document and sentence levels, including a three point scale polarity. Pairwise agreement Kappa for documents ranges from 0.412 to 0.897 and from 0.682 to 0.756 for sentence level.

In other languages, several corpora have been published. The TASS Spanish corpus focuses on microblog posts from Twitter (Román et al., 2015). It is composed of 68,017 messages in ten topics such as politics, entertainment, literature and soccer. Polarity is indicated at text and entity levels. No inter-annotator agreement results are given. The Emotiblog corpus (Boldrini et al., 2009) includes a 30,000 word Spanish component. It includes message blogs that focus on topical news such as the Kyoto protocol but not on financial topics. The fine-grained annotation scheme includes a mix of syntactic and semantic features. Average inter-annotator agreement across all categories is 0.68. In German, there are two projects of interest. Scholtz et al (Scholz et al., 2012) published a corpus of news-related texts. It includes 15,089 sentences and includes 3-level polarity scale (Cohen's Kappa = 0.88). Momtazi (Momtazi, 2012) published another German corpus. It focuses on blog messages and includes 500 short texts from social media. Messages are annotated at document level with four polarity scores. Fleiss' Kappa scores range from 0.5 to 0.65.

This short review indicates that microblog annotated corpora focused on the financial sector are scarce. In addition, very few of these resources provide polarity information at entity level. The SSIX GS corpus addresses this lack by providing polarity annotation, on a continuous scale, for financial entities mentioned in microblogs.

3. Data collection

The gathering process of the data requires the actual collection from data sources and the extraction of the data with filters to eliminate irrelevant contents such as spams and non-reliable sources.

3.1. Data source

The corpora are specialised in the domain of stock-market investments. They are intended to represent financial microblog messages in each language. A number of platforms such as Twitter and Stocktwits provide messages published by their users who are investors with a specific interest in following stock prices. They share stock trend analysis and relevant events. Our sampling frame is a collection of messages from the StockTwits and Twitter platforms for their focus on financial entities called *cashtags*, e.g., *$AAPL* for Apple's stock. The texts are classified according to the *cashtag* categories that refer to companies or stocks. The messages are in English, Spanish and German.

StockTwits and Twitter are social media platforms. StockTwits was used as a source for the English corpus. The reason is that it provided an official firehose access or on-demand historical English data specialised in the financial domain. The period of collection was between October 2011 and June 2015. The Spanish and German corpora were sourced from Twitter for several weeks between October and November 2016. They underwent a less sophisticated sampling process due to limited access on demand. The data were then passed through further content filtering operations (to exclude spam and unreliable authors) which are described below.

3.2. Extraction process

We explain the sampling process from the data sources. We then describe how the data was filtered to obtain relevant messages.

3.2.1. Sampling the data source

The extraction process was conducted in two stages. Firstly, samples were collected in each language by applying a stratified strategy. The purpose was to ensure representativeness in the corpora (Biber, 1993). We first categorised messages according to their time stamps and their *cashtags*. We then randomly sampled messages from all categories to make sure that each sample included a balanced representation of messages according to time and entities.

3.2.2. Filtering the data

After sampling was applied, a number of filtering rules were identified to eliminate non-desirable messages in each corpus. Such messages contained spams and irrelevant content. To detect irrelevant content we used the following criteria: *Number of messages per author*, *number of cashtags mentioned*, *number of message followers*. Once detected the corresponding messages were eliminated and three language-specific corpora were prepared for manual annotation.

4. Annotation

After extracting messages from the data sources, we conducted a manual annotation of the corpora.

4.1. Description of the Annotated Corpora

The corpus is divided into three corpora that match the language of the messages. Table 1 shows the properties of each corpus. The corpora include the messages and their

Corpus Language	Number of annotated messages	Number of unique entities annotated	Number of tokens
English	1,336	805	16,567
Spanish	766	182	11,482
German	784	101	13,660

Table 1: The English, Spanish and German corpora

score annotations. In each message the targeted entity is assigned a score. If a message included several entities, each entity was rated independently, resulting in identical or different scores depending on the context. The following is a Spanish message with its annotation including sentiment polarity and its target:

- `ID`: 1,069
- `Source ID`: 272
- `Text`: $BAC reportó ganancias e ingresos mejores que los proyectados, impulsados por el trading de bonos #stocks (Translation: $BAC reported higher than planned profits and revenues, which were pushed up by trading bonds #stocks)
- `Target`: $bac
- `Average score`: 0.62
- `Final score` : 0.412

The *$BAC* message target appears in the text and is assigned a positive score. The average score represents the average of the scores assigned by the annotators. The final score represents the consolidated score. The ID is the unique identifier within the corpus and the source ID is the message identifier of the microblog platform.
In terms of textual characteristics, financial microblogs demonstrate domain-related and linguistic specificities. Much domain terminology is used to describe assets, events and intentions, e.g., *calls*, *puts* indicating buy or sell options. The syntax is also specific as parataxis appears to be predominant. The following excerpts from the English GS (JSON format) show simple messages in which strong and even weak punctuation or symbols are used as a separators for clauses.


```
{
"Unique":  "14908$TZA",
"Text":  "Buying $TZA calls / $TNA puts
tomorrow",
"Average Score":  0.841,
"Type":  "stocktwits",
"source_id":  5350721
},
{
"Unique":  "14975$SPY",
"Text":  "$SPY Yes buy everything up,
World is fine",
"Average Score":  0.829,
"Type":  "stocktwits",
"source_id":  6058084
}
```


4.2. Applying the Annotation Scheme

The three corpora underwent a twofold process: annotation and consolidation.

4.2.1. Annotation

Firstly, several expert annotators annotated the messages by applying a score to each financial entity found in each message. Scoring involved assigning values along a continuous range of [-1;1]. This range covers seven categories that were detailed in specific guidelines for annotators. On a scale of 1 to 7, the annotators were asked to annotate how bearish (negative) or bullish (positive) the opinion holder was regarding a specific financial entity mentioned in a given message. The annotations spanned from clearly negative (*selling intention*) to clearly positive (*buying intention*). The intermediary categories showed different degrees on the scale, and 0 was kept for purely informative messages in nature. The annotators were instructed not to overuse neutral scores if possible.

4.2.2. Consolidation

The second part of the annotation process consisted in consolidating the different scores assigned to each message by the annotators. It implied the systematic flagging of inconsistent scores for the same message (e.g., averaging positive and negative values). We applied two filtering rules. The first one consisted in setting a minimum number of required annotation scores (set to 3) per entity in order to proceed with computing their average. The second rule conditioned the calculation to only taking scores within a specific deviation of one another. If a score was over the deviation, it was not included in the consolidated score. These two rules ensured a degree of control over extreme and isolated scores and prevented them from being automatically included in the final set. Finally, the messages that conflicted with the two rules were flagged and reviewed by an expert for final score assignment. During consolidation, the selection of relevant text spans corresponding to scores was abandoned due to the lack of agreement between annotators.

5. Quality tests and results

Different measures were applied at different stages in the process to evaluate the quality of the corpora. First, we applied a binomial test (Baroni and Evert, 2009) on the StockTwits English corpus to measure the quality of the sampling method regarding representativeness. We could not apply this to the Spanish and German corpora as, unlike StockTwits, we could not access the full population for the time

period due to the sampling restrictions of the Twitter API. Results showed that a low number of tokens were proportionally different between the initial data set and the corpora (see Table 2).

Total number of tokens (%)	72,263 (100%)
Number of tokens significantly different	1,330 (1.8%)
Number of tokens not significantly different	70,933 (98.2%)

Table 2: Number of tokens that are different and similar between the initial data source and the English corpus (p-value < 0.05)

Following Artstein and Poesio (2008), inter-annotator agreement was also measured with Fleiss' Kappa and Krippendorff's Alpha. The annotations were split into 2 classes (negative and positive). Results for each corpus are presented in Table 3.

Languages	English	Spanish	German
Raters	4	3	3
Krippendorff's Alpha	0.61	0.504	0.594
Fleiss' Kappa	0.69	0.547	0.703

Table 3: Inter-annotator agreement for the three corpora

The rates between corpora are not identical; especially with the Spanish corpus. These differences suggest that, in spite of identical annotation guidelines, Spanish-speaking annotators deviated from their English and German-speaking counterparts. This may indicate variations in interpreting the degrees of sentiments in the examples given in the English guidelines.

Overall, agreement results are comparable to the state-of-the-art annotation experiments mentioned in Section 2.. Interpreting the values is a matter of discussion in itself as there is not any definite scale that expresses exact agreement levels, in particular under a continuous scale setting. For a recent interpretation on consensus analysis the reader is referred to (Artstein and Poesio, 2008).

6. Conclusion

In this paper, we have presented the compilation and annotation of three gold standard corpora for financial microblogs. We followed a twofold process. Firstly, we extracted the data from the Stocktwits and Twitter microblog services. The messages were filtered to ensure the elimination of irrelevant messages such as spams and bot-authored texts. Secondly, we created the annotated corpora by applying a sentiment polarity scale at entity-level in three different languages, i.e., English, Spanish and German. These corpora were annotated by following strict guidelines by several experts in the financial domain.

The corpus data structures follows the TSV data-exchange format, the English corpus is available online[3] and the Spanish and German corpora are available on request. Future work will involve the integration of more languages in the corpus.

7. Acknowledgements

We would like to thank all the people involved in the creation of the Gold Standard. This work is funded by the SSIX Horizon 2020 project (Grant agreement No 645425)



Co-financed by the European Union
Connecting Europe Facility

8. Bibliographical References

Artstein, R. and Poesio, M. (2008). Inter-coder Agreement for Computational Linguistics. *Comput. Linguist.*, 34(4):555–596, December.

Baroni, M. and Evert, S. (2009). Statistical methods for corpus exploitation. In Anke Lüdeling et al., editors, *Corpus Linguistics An International Handbook*, volume 2, pages 777–803. De Gruyter, Berlin, Boston.

Biber, D. (1993). Representativeness in Corpus Design. *Literary and Linguistic Computing*, 8(4):243–257.

Boldrini, E., Balahur Dobrescu, A., Martínez-Barco, P., and Montoyo, A. (2009). EmotiBlog: a fine-grained model for emotion detection in non-traditional textual genres. In *Proceedings of the 1st Workshop on Opinion Mining and Sentiment Analysis*, pages 22–31, Seville, Spain. WOMSA.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, San Rafael, Calif., May.

Malo, P., Sinha, A., Takala, P., Ahlgren, O., and Lappalainen, I. (2013). Learning the Roles of Directional Expressions and Domain Concepts in Financial News Analysis. In *Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops*, ICDMW '13, pages 945–954, Washington, DC, USA. IEEE Computer Society.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Momtazi, S. (2012). Fine-grained German Sentiment Analysis on Social Media. In *Proceedings of the Eight International Conference on Language Resources and Evaluation*, Istanbul, Turkey. European Language Resources Association (ELRA).

O'Hare, N., Davy, M., Bermingham, A., Ferguson, P., Sheridan, P., Gurrin, C., and Smeaton, A. F. (2009). Topic-dependent Sentiment Analysis of Financial Blogs. In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*, TSA '09, pages 9–16, New York, NY, USA. ACM.

[3] https://ssix-project.eu/knowledgebase/gold-standards/

Román, J. V., Cámara, E. M., Morera, J. G., and Zafra, S. M. J. (2015). TASS 2014 - The Challenge of Aspect-based Sentiment Analysis. *Procesamiento del Lenguaje Natural*, 54(0):61–68.

Scholz, T., Conrad, S., and Hillekamps, L. (2012). Opinion Mining on a German Corpus of a Media Response Analysis. In *Text, Speech and Dialogue*, Lecture Notes in Computer Science, pages 39–46. Springer, Berlin, Heidelberg, September.

Takala, P., Malo, P., Sinha, A., and Ahlgren, O. (2014). Gold-standard for Topic-specific Sentiment Analysis of Economic Texts. In *International Conference on Language Resources and Evaluation (LREC)*, pages 2152–2157, Reykjavik, Iceland.

Sarcasm Target Identification: Dataset and An Introductory Approach

Aditya Joshi[2,3,4], **Pranav Goel**[1], **Pushpak Bhattacharyya**[2], **Mark J Carman**[3]
[1]IIT (BHU) Varanasi, India, [2]IIT Bombay, India, [3]Monash University, Australia
[4]IITB-Monash Research Academy, India
adityaj@cse.iitb.ac.in, pranav.goel.cse14@iitbhu.ac.in, pb@cse.iitb.ac.in, mark.carman@monash.edu

Abstract

Past work in computational sarcasm deals primarily with sarcasm detection. In this paper, we introduce a novel, related problem: sarcasm target identification (*i.e.*, extracting the target of ridicule in a sarcastic sentence). As a benchmark, we introduce a new dataset for the task. This dataset is manually annotated for the sarcasm target in book snippets and tweets based on our formulation of the task. We then introduce an automatic approach for sarcasm target identification. It is based on a combination of two types of extractors: one based on rules, and another consisting of a statistical classifier. Our introductory approach establishes the viability of sarcasm target identification, and will serve as a baseline for future work.

Keywords: Sentiment Analysis, Computational Sarcasm, Aspect Extraction

1. Introduction

Sarcasm is a form of verbal irony that is intended to express contempt or ridicule (Source: The Free Dictionary). While several approaches have been reported for sarcasm detection (Rajadesingan et al., 2015; Joshi et al., 2015; Tsur et al., 2010; González-Ibánez et al., 2011), no past work, to the best of our knowledge, has attempted to identify a crucial component of sarcasm: the target of ridicule (Campbell and Katz, 2012). This is important because the sentiment of the sarcastic text needs to be attributed to this target of ridicule. Towards this motivation, **we introduce 'sarcasm target identification'**: the task of extracting the target of ridicule (*i.e.*, *sarcasm target*) of a sarcastic text. The input is a sarcastic text while the output is either (a) a subset of words in the sentence that point to the sarcasm target, or (b) a fall-back label 'Outside'[1]. In this paper, we **present a manually labeled dataset** consisting of text from two domains: tweets and book snippets. We also **report an automatic approach that takes as input a sarcastic text and returns its sarcasm target**. Our hybrid approach combines a rule-based extractor (that implements a set of rules) and a statistical extractor (that uses a word-level classifier for every word in the sentence, to predict if the word will constitute the sarcasm target).

Since this is the first work in sarcasm target detection, no past work exists to be used as a baseline. Hence, we devise two baselines to validate the strength of our work. The first is a simple, intuitive baseline to show if our approach (which is computationally more intensive than this simple baseline) holds value. In absence of past work, using simple and obvious techniques to solve a problem have been considered as baselines in sentiment analysis (Tan et al., 2011; Pang and Lee, 2005). As the second baseline, we use a technique reported for sentiment/opinion target identification since sentiment target identification appears to be related to sarcasm target identification, on the surface.

Our manually labeled datasets are available for download at: `https://github.com/Pranav-Goel/Sarcasm-Target-Detection`. Each unit consists of a piece of text (either book snippet or tweet) with the annotation as the sarcasm target where the sarcasm target is a subset of words in the text or the fall-back label 'Outside'. In addition to this, our hybrid approach for sarcasm target identification will serve as a baseline for future work.

Sarcasm target identification **can benefit natural language generation and sentiment analysis systems**. Being able to recognize the entity towards which the negative sentiment was intended, a natural language generation system will have more context to generate a response. Similarly, a sentiment analysis system will be able to attribute the negative sentiment in a sarcastic text towards the correct aspect of a product or the appropriate entity.

2. Related work

Computational sarcasm primarily focuses on sarcasm detection: classification of a text as sarcastic or non-sarcastic. Joshi et al. (2016a) present a survey of sarcasm detection approaches. They observe three trends in sarcasm detection: semi-supervised extraction of sarcastic patterns, use of hashtag-based supervision, and use of contextual information for sarcasm detection (Tsur et al., 2010; Davidov et al., 2010; Joshi et al., 2015). However, to the best of our knowledge, no past work aims to identify phrases in a sarcastic sentence that indicate the target of ridicule in the sentence.

Related to sarcasm target identification is sentiment target identification. Sentiment target identification deals with identifying the entity towards which sentiment is expressed in a sentence. Qiu et al. (2011) present an approach to extract opinion words and targets collectively from a dataset. Aspect identification for sentiment has also been studied. This deals with extracting aspects of an entity (for example, color, weight, battery in case of a cell phone). Probabilistic topic models have been commonly used for the same. Titov et al. (2008) present a probabilistic topic model that jointly estimates sentiment and aspect in order to achieve sentiment summarization. Lu et al. (2011) perform multi-aspect sentiment analysis using a topic model.

[1]This label is necessary because the sarcasm target may not be present as a word, as discussed in Section 2.

Example	Target
Love when you don't have two minutes to send me a quick text.	you
Don't you just love it when Microsoft tells you that you're spelling your own name wrong.	Microsoft
I love being ignored.	being ignored
He is as good at coding as Tiger Woods is at avoiding controversy.	He, Tiger Woods
Oh, and I suppose the apple ate the cheese.	Outside

Table 1: Examples of sarcasm targets

Several other topic model-based approaches to aspect extraction have been reported (Mukherjee and Liu, 2012). To the best of our knowledge, ours is the first work that deals with sarcasm target identification.

3. Formulation

Sarcasm is a well-known challenge to sentiment analysis (Pang et al., 2008). Consider the sarcastic sentence '*My cell phone has an awesome battery that lasts 20 minutes*'. This sentence mocks the battery of the cell phone. Aspect-based sentiment analysis deals with identifying sentiment expressed towards different aspects or dimensions of an entity. Therefore, aspect-based sentiment analysis needs to identify that the sentence expresses a negative sentiment towards the aspect '*battery*'. With sarcasm target identification, we hope to enable aspect-based sentiment analysis to attribute the negative sentiment to the correct target aspect in the case of a sarcastic text.

We define the **sarcasm target** as the entity or situation being ridiculed in a sarcastic text. In the case of '*Can't wait to go to class today*', the word '*class*' is the sarcasm target. We make two (fair) assumptions here. (a) Every sarcastic text has at least one sarcasm target. This holds true by definition of sarcasm. Also, (b) The notion of sarcasm target is applicable for sarcastic texts only. A non-sarcastic text does not have a sarcasm target. With these assumptions, we define **sarcasm target identification** as follows. Given a sarcastic text, sarcasm target identification is the task of extracting the subset of words that indicate the target of ridicule. However, in some cases, the target of ridicule may not be present among the words. In such a case, a fall-back label 'Outside' is expected. Examples of some sarcasm targets are given in Table 1.

Some challenges of sarcasm target identification are:

- **Presence of multiple candidate phrases**: Consider the sentence '*This phone heats up so much that I strongly recommend chefs around the world to use it as a cook-top*'. In this sentence, the words '*chefs*', '*cook-top*' and '*phone*' are candidate phrases. However, only the '*phone*' is being ridiculed in this sentence.
- **Multiple sarcasm targets**: A sentence like '*You are as good at coding as he is at cooking*' ridicules both '*you*' and '*he*', and hence, both are sarcasm targets.
- **Absence of a sarcasm target word (the 'Outside' case)**: Consider something bad happens in the beginning of the day and one says, '*What a great way to start off the day!*'. No specific word in the sentence is the sarcasm target. The target here is the situation that started off the day. We refer to such cases as the 'Outside' cases.

4. Dataset

4.1. Collection of sarcastic text

We experiment with two datasets: book snippets and tweets. The dataset of book snippets is a sarcasm-labeled dataset by (Joshi et al., 2016b). From this dataset, 224 book snippets marked as sarcastic are used. The second dataset is dataset of tweets, given by (Riloff et al., 2013). 506 sarcastic tweets from this dataset are used. These book snippets and tweets are manually annotated with the sarcasm target. The statistics of the two datasets are shown in Table 2. The average length of a sarcasm target is 1.6 words in the case of book snippets and 2.08 words in the case of tweets. The last two rows in the table point to an interesting observation. In both the datasets, the average polarity strength[2] of sarcasm target is lower than polarity strength of rest of the sentence. This shows that the sarcasm target is likely to be more neutral than sentiment-bearing.

4.2. Annotation

The annotation is manually carried out by three annotators. The annotators hold more than 5 years of linguistic annotation experience each for sentiment analysis, word sense disambiguation and related tasks. The annotators first discuss the notion of sarcasm target. They are then given the following guiding question for their annotation:

'*The given text is sarcastic. Which words in this text indicate the target of sarcasm that the author making fun of? If you cannot locate specific words, mark them as 'Outside"*'.

The target could be an entity or a phrase referring to the situation, and the annotators were told to prefer specific entities over situations when possible.

Every textual unit in both the datasets is labeled for sarcasm target, and the label comprises of either a subset of words in the tweet or the fall-back label 'Outside'. Note that a subset of words means that the word or phrase is taken *exactly as it is* from the sarcastic text being marked for the target - no change in the case, punctuation or wording occurs when the corresponding labels are given. In the case of a phrase, any punctuation at the boundary of the phrase is not included in the sarcasm target/label. In the case of multiple targets, all the targets are annotated.

[2]Polarity strength is the sum of polarities of words. We use a sentiment word-list (McAuley and Leskovec, 2013) to get the strength values

	Snippets	Tweets
Count	224	506
Average #words	28.47	13.06
Vocabulary	1710	1458
Total words	6377	6610
Average length of sarcasm target	1.6	2.08
Average polarity of sarcasm target	0.0087	0.035
Average polarity of portion apart from sarcasm target	0.027	0.53

Table 2: Statistics of our datasets; 'Snippets': Book Snippets

4.3. Inter-annotator agreement

To understand the challenge posed by sarcasm target annotation, we conduct an additional experiment. Two annotators, different from the original set of expert annotators who produced the gold labels, are given the same task of annotation. The annotators are undergraduate students of computer science with English as the primary language of instruction throughout their academic tenure. Following the question as stated above, the new annotators annotate a subset of 50 tweets and 50 book snippets separately from the original annotation and separately from each other.

Since this is a phrase extraction task and not a label assignment task, we use match two proportion-based metrics to quantify the inter-annotator agreement. The first metric is exact match. This metric is defined as the percentage of texts for which the pair of annotations are exactly the same. in the case of the book snippets dataset, the exact match is 28% while in the case of tweets, it is 38%. The higher value in the case of tweets is likely to be due to the more direct nature of ridicule in tweets leading to clearer and more specific targets. The second metric is partial match. This metric is defined as the percentage of texts for which the pair of annotations show a partial overlap. The partial match in the case of both the book snippets and tweets datasets is 66%. There was no overlap in annotations for about one third of the dataset for both book snippets and tweets. The values of exact and partial match hint at the task presenting a reasonable challenge to humans.

5. An Introductory Approach

5.1. Architecture

Our introductory approach for sarcasm target identification is depicted in Figure 1. The input is a sarcastic sentence while the output is the sarcasm target. The approach consists of two kinds of extractors: (a) a **rule-based extractor** that implements nine rules to identify different kinds of sarcasm targets, and (b) a **statistical extractor** that uses statistical classification techniques. The two extractors individually generate lists of candidate sarcasm targets. The third component is the **integrator** that makes an overall prediction of the sarcasm target by choosing among the sarcasm targets returned by the individual extractors. The overall output is a subset of words in the sentence. In case no word is found to be a sarcasm target, a fall-back label 'Outside' is returned. In the forthcoming subsections, we describe the three modules in detail.

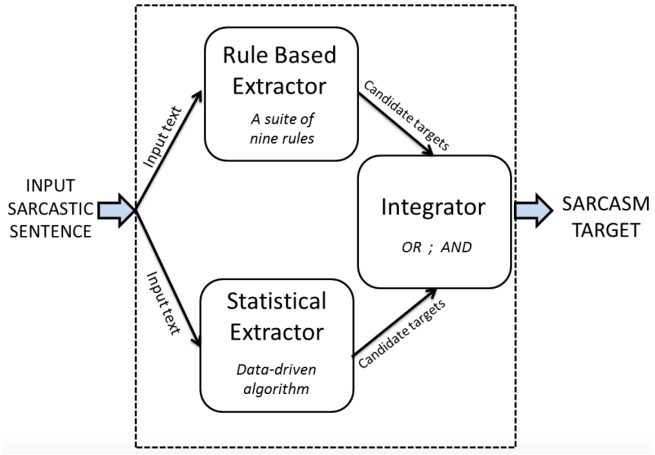


Figure 1: Architecture of our Sarcasm Target Identification Approach

5.1.1. Rule-based Extractor

Our rule-based extractor consists of nine rules that take as input the sarcastic sentence, and return a set of candidate sarcasm targets. The rules are summarized in Table 3. In detail, these rules are as follows:

1. **R1** (**Pronouns and Pronominal Adjectives**): R1 returns pronouns such as '*you, she, they*' and pronominal adjectives (followed by their object) (as in the case of '*your shoes*'). Thus, for the sentence '*I am so in love with my job*', the phrases '*I*' (pronoun) and '*my job*' (based on the pronominal adjective 'my') are returned as candidate sarcasm targets. This is based on observations by Shamay et al. (2005).
2. **R2** (**Named Entities**): Named entities in a sentence may be sarcasm targets. This rule returns all *named entities* in the sentence. In case of '*Olly Riley is so original with his tweets*', R2 predicts the phrase '*Olly Riley*' as a candidate sarcasm target.
3. **R3** (**Sentiment-bearing verb as the pivot**): This rule is based on the idea by (Riloff et al., 2013) that sarcasm may be expressed as a contrast between a positive sentiment verb and a negative situation. In case of '*I love being ignored*', the sentiment-bearing verb '*love*' is positive. The object of '*love*' is '*being ignored*'. Therefore, R3 returns '*being ignored*' as the candidate sarcasm target. If the sentiment-bearing verb is negative, the rule returns '*Outside*' as a candidate sarcasm target.
4. **R4** (**Non-sentiment-bearing verb as the pivot**): This rule applies in case of sentences where the verb does not bear sentiment. The rule identifies which out of subject or object has a lower sentiment score, and returns the corresponding portion as the candidate sarcasm target. For example, rule R4 returns '*to have a test on my birthday*' as the candidate sarcasm target in case of '*Excited that the teacher has decided to have a test on my birthday!*' where 'decided' is the non-sentiment-bearing verb. This is also based on Riloff et al. (2013).

Rule	Definition	Example
R1	Return pronouns and pronominal adjectives	Love when **you** don't have two minutes to send me a quick text .. ; I am so in love with **my job**.
R2	Return named entities as target	Don't you just love it when **Microsoft** tells you that you're spelling your own name wrong.
R3	Return direct object of a positive sentiment verb	I love **being ignored**.
R4	Return phrase on lower sentiment side of primary verb	So happy to just find out it has been *decided* **to reschedule all my lectures and tutorials for me to night classes at the exact same times**!
R5	Return Gerund and Infinitive verb phrases	**Being covered in hives** is so much fun!
R6	Return nouns preceded by a positive sentiment adjective	Yep, this is indeed an *amazing* **donut** ..
R7	Return subject of interrogative sentences	A murderer is stalking me. Could **life** be more fun?
R8	Return subjects of comparisons (similes)	**He** is as good at coding as **Tiger Woods** is at controversies.
R9	Return demonstrative adjective-noun pairs	Oh, I love **this jacket**!

Table 3: Summary of rules in the rule-based extractor; Boldfaced phrases indicate sarcasm targets

5. **R5** (**Gerund phrases and Infinitives**): R5 returns the gerund phrase '*being covered in rashes*' in case of '*Being covered in rashes is fun.*' as the candidate sarcasm target. Similarly, in case of '*Can't wait to wake up early to babysit!*', the infinitive '*to wake up early to babysit*' is returned.
6. **R6** (**Noun phrases containing positive adjective**): R6 extracts noun phrases of the form 'JJ NN' where JJ is a positive adjective, and returns the noun indicated by NN. Specifically, 1-3 words preceding the nouns in the sentence are checked for positive sentiment. In case of '*Look at the most realistic walls in a video game*', the noun '*walls*' is returned as the sarcasm target.
7. **R7** (**Interrogative sentences**): R7 returns the subject of an interrogative sentence as the sarcasm target. Thus, for '*A murderer is stalking me. Could life be more fun?*', the rule returns '*life*' as the target.
8. **R8** (**Sarcasm in Similes**): This rule captures the *subjects/noun phrases* involved in similes and 'as if' comparisons. The rule returns the subject on both sides, as in '*He is as good at coding as Tiger Woods is at avoiding controversy.*' Both '*He*' and '*Tiger Woods*' are returned as targets. This is derived from work on sarcastic similes by Veale et al. (2010).
9. **R9** (**Demonstrative adjectives**): This rule captures nouns associated with demonstrative adjectives - *this/that/these/those*. For example, for the sentence '*Oh, I love this jacket!*', R9 returns '*this jacket*' as the sarcasm target.

Combining the outputs of individual rules to generate candidate sarcasm targets of the rule-based extractor: To generate the set of candidate sarcasm targets returned by the rule-based extractor, a weighted majority approach is used as follows. Every rule above is applied to the input sarcastic sentence. Then, every word is assigned a score that sums the accuracy of rules which predicted that this word is a part of the sarcasm target. This accuracy is the overall accuracy of the rule as determined by solely the rule-based classifier. Thus, the integrator weights each word on the basis of how good a rule predicting it as a target was. Words corresponding to the maximum value of this score are returned as candidate sarcasm targets.

5.1.2. Statistical Extractor

The statistical extractor uses a classifier that takes as input a word (along with its features) and returns if the word is a sarcasm target. To do this, we decompose the task into n classification tasks, where n is the total number of words in the sentence. This means that every word in input text is considered as an instance, such that the label can be 1 or 0 depending on whether or not the given word is a part of sarcasm target. For example, '*Tooth-ache is fun*' with sarcasm target as '*tooth-ache*' is broken down into three instances: '*tooth-ache*' with label 1, '*is*' with label 0 and '*fun*' with label 0. In case the target lies outside the sentence, all words have the label 0.

We then represent the instance (*i.e.*, the word) as a set of following features: (A) **Lexical**: *Unigrams*, (B) **Part of Speech (POS)**-based features: *Current POS*, *Previous POS*, *Next POS*, (C) **Polarity**-based features: *Word Polarity* : Sentiment score of the word, *Phrase Polarity* : Sentiment score for the trigram formed by considering the previous word, current word and the next word together (in that order). These polarities lie in the range [-1,+1]. These features are based on our analysis that the target phrase or word tends to be more neutral than the rest of the sentence, and (D) **Pragmatic features**: *Capitalization* : Number of capital letters in the word. Capitalization features are chosen based on features from (Davidov et al., 2010).

The classifiers are trained with words as instances while the sarcasm target is to be computed at the sentence level. Hence, the candidate sarcasm target returned by the statistical extractor consists of words for which the classifier returned 1. For example, the sentence 'He is nice' is broken up into three instances: 'He', 'is' and 'nice'. If the classifier returns 1, 0, 0 for the three instances respectively, the statistical extractor returns 'He' as the candidate sarcasm target. Similarly, if the classifier returns 0, 0, 0 for the three instances, the extractor returns the fall-back label 'Outside'.

5.1.3. Integrator

The integrator determines the sarcasm target based on the outputs of the two extractors. We consider two configurations of the integrator:

1. Hybrid **OR**: In this configuration, the integrator predicts the set of words that occur in the output of either of the two extractors as the sarcasm target. If the lists are empty, the output is returned as 'Outside'.
2. Hybrid **AND** : In this configuration, the integrator predicts the set of words that occur in the output of both the two extractors as the sarcasm target. If the intersection of the lists is empty, the output is returned as 'Outside'.

The idea of using two configurations OR and AND is based on a rule-based sarcasm detector by (Khattri et al., 2015). While AND is intuitive, the second configuration OR is necessary because our extractors individually may not capture all forms of sarcasm target. This is intuitive because *our rules may not cover all forms of sarcasm targets*.

5.2. Experiment Setup

We use **SVM Perf** (Joachims, 2006) to train the classifiers, optimized for F-score with epsilon e=0.5 and RBF kernel[3]. We set C=1000 for tweets and C=1500 for snippets. We report our results on **four-fold cross validation** for both datasets. Note that we convert individual sentences into words. Therefore, the dataset in the case of book snippets has 6377 instances, while the one of tweets has 6610 instances. The four folds for cross-validation are created over these instances. With a word as instance, the task is binary classification: 1 indicating that the word is a sarcasm target and 0 indicating that it is not. For rules in the rule-based extractor, we use tools in NLTK (Bird, 2006), wherever necessary.
We consider two baselines with which our hybrid approach is compared:

1. **Baseline 1: All Objective Words**: As the first baseline, we design a naïve approach for our task: include all words of the sentence which are not stop words, and have neutral sentiment polarity, as the predicted sarcasm target.
2. **Baseline 2**: Sequence labeling has been reported for opinion target identification (Jin et al., 2009). Therefore, we use **SVM-HMM** (Altun et al., 2003) with default parameters as the second baseline.

We report performance using two metrics: Exact Match Accuracy and Dice Score. These metrics have been used in past work in information extraction (Michelson and Knoblock, 2007). As per their conventional use, these metrics are computed at the sentence level. The metrics are described as:

- **Exact Match (EM) Accuracy**: An exact match occurs if the list of predicted target(s) is exactly the same as the list of actual target(s). The accuracy is computed as number of instances with exact match divided by total instances.
- **Dice Score (DC)**: Dice score(Sørensen, 1948) is used to compare similarity between two samples. This is considered to be a better metric than Exact match accuracy because it accounts for missing words and extra words in the target. Let the two lists (predicted and actual) be X and Y. Dice score is given by $(2X \cap Y)/(X + Y)$.

[3]RBF Kernel performed better than linear kernel.

5.3. Results

Rule	Overall		Conditional	
	EM	**DS**	**EM**	**DS**
R1	7.14	**32.8**	7.65	35.23
R2	**8.48**	16.7	19.19	37.81
R3	4.91	6.27	16.92	21.62
R4	2.67	11.89	4.38	19.45
R5	1.34	6.39	2.32	11.11
R6	4.01	6.77	8.91	15.02
R7	3.12	10.76	9.46	32.6
R8	4.91	6.78	**35.02**	45.17
R9	4.46	6.94	34.48	**53.67**

Table 4: Results for individual rules for book snippets

Rule	Overall		Conditional	
	EM	**DS**	**EM**	**DS**
R1	6.32	19.19	8.69	26.39
R2	11.26	16.18	30.32	43.56
R3	**12.45**	20.28	34.24	**55.77**
R4	6.91	13.51	18.42	36.0
R5	9.28	**23.87**	15.36	39.47
R6	10.08	16.91	19.31	32.42
R7	9.88	15.21	32.25	49.65
R8	11.26	11.26	**50**	50
R9	11.46	13.28	43.59	50.51

Table 5: Results for individual rules for tweets

This section presents our results in two steps: performance of individual rules that are a part of the rule-based extractor, and performance of the overall approach.

5.3.1. Performance of rules in the rule-based extractor

Tables 4 and 5 present the performance of the rules in our rule-based extractor, for snippets and tweets respectively. The two metrics (exact match accuracy and dice score) are reported for two cases: Overall and Conditional. 'Overall' spans all text units in the dataset whereas 'Conditional' is limited to text units which match a given rule (*i.e.*, where the given linguistic phenomenon of, say, gerunds, etc. is observed). Considering the 'Conditional' case is crucial because a rule may be applicable for a specific form of sarcasm target, but may work accurately in those cases. Such a rule will have a low 'overall exact match/dice score' but a high 'conditional exact match/dice score.' Values in bold indicate the best performing rule for a given performance metric. As seen in the tables, the values for 'conditional' are higher than those for 'Overall'. For example, consider rule R7 in Table 4. Exact match of 3.12 (for overall accuracy) as against 9.46 (for conditional accuracy). This situation is typical of rule-based systems where rules may not cover all cases but be accurate for situations that they do

cover. For tweets, R3 has a very high dice score (conditional) (55.77). This rule is based on the intuition in (**?**) that contrast of positive sentiment with negative situation is a strong indicator of sarcasm target.

5.3.2. Overall performance

Tables 6 and 7 compare the six approaches for snippets and tweets respectively. All our approaches outperform the baseline in the case of exact match and dice score. in the case of tweets, Table 7 shows that the rule-based extractor achieves a dice score of 29.13 while that for statistical extractor is 31.8. Combining the two together (owing to our hybrid architecture) improves the dice score to 39.63. This improvement also holds for book snippets. This **justifies the 'hybrid' nature of our approach**. Hybrid OR performs the best in terms of Dice Score. However, for exact match accuracy, Hybrid AND achieves the best performance (16.51 for snippets and 13.45 for tweets). This is likely because Hybrid AND is restrictive with respect to the predictions it makes for individual words. The statistical extractor performs better than rule-based extractor for all three metrics. For example, in the case of tweets, the dice score for statistical extractor is 31.8 while that for rule-based extractor is 29.13. Also, nearly all results (across approaches and metrics) are higher in the case of tweets as compared to snippets. Since tweets are shorter than snippets (as shown in Table 2), **it is likely that tweets are *more direct* in their ridicule as compared to snippets.** We discuss an experiment to validate this observation in Section 5.4.. Thus, tackling the task of sarcasm target identification and the new dataset we present can help gain insights into the nature of sarcasm.

5.4. Study on nature of sarcasm in tweets versus book snippets

As an example of the kind of investigations into the nature of sarcasm that this dataset can facilitate, we conduct an experiment to test the hypothesis that 'tweets are more direct in their ridicule than snippets'. Book snippets and tweets containing the word 'man' are selected. Thirteen tweets contain the word 'man'. Many book snippets contain the word 'man', of which 13 are randomly chosen among these. 13 tweets and 13 snippets are randomly paired up. Two human annotators (not the ones involved in the sarcasm target annotations) are asked to choose which of the two was more 'direct' in its ridicule. The two human annotators are 25-30 years old, one male engineer (A1) and one female linguist (A2). The two annotators have no prior experience in sarcasm annotation but have studied English as a primary language from school onwards. The two annotators are not told about the claim to be verified, and are not told that one of the pair is a tweet and one of them a book snippet. For every pair, the annotator answers the question "*Which of the two is more direct in its ridicule?*". A2 selects tweets to be more direct in 11 cases out of 13 while A1 does so in 10 out of 13. In other words, in 11 out of 13 cases, A2 states that for a given pair, the tweet is more direct in its ridicule than the book snippet.

6. Error analysis

We now discuss sources of errors made by our system.

Approach	EM	DC
Baseline 1: All Objective Words	0.0	16.14
Baseline 2: Seq. Labeling	12.05	31.44
Only Rule-Based	9.82	26.02
Only Learning-Based	12.05	31.2
Hybrid OR	7.01	**32.68**
Hybrid AND	**16.51**	21.28

Table 6: Performance of sarcasm target identification for snippets

Approach	EM	DC
Baseline 1: All Objective Words	1.38	27.16
Baseline 2: Seq. Labeling	12.26	33.41
Only Rule-Based	9.48	29.13
Only Learning-Based	10.48	31.8
Hybrid OR	9.09	**39.63**
Hybrid AND	**13.45**	20.82

Table 7: Performance of sarcasm target identification for tweets

- **Confusion between *reason* and target**: Sometimes an action appears to be the target of sarcasm, but sometimes the action (or something else), on deeper reflection, may appear to be only the 'reason' for the speaker to be mocking the actual target via sarcasm. Consider the following examples :
  - For the sentence 'I love *being ignored*', we can only infer that the act of "being ignored" is the target, but in 'I love *being ignored by my girlfriend*', it is not exactly clear if "being ignored" is a part of the target, or here, it just becomes the reason for using sarcasm against the only target : "my girlfriend". It also makes it a bit hard for our system to learn common phrases and utilize patterns for detection.
  - 'I love when I see *people using the elevator at the gym*.' The speaker could be mocking the act of people using the elevator at the gym (by saying that he/she loves to see this act), or could me mocking the people, *because* they use the elevator at a gym!
- **Lack of Context**: Context may often be necessary to determine a sarcasm target. Consider the sentence : '*Oh, you are such a lovely couple! You two could even give those Potters living down the street a real run for their money!*' In this case, you/You two are being ridiculed. For '*those Potters*', two scenarios are possible : a) The Potters could be a well known '*lovely couple*', and their reference is used for augmenting the (sarcastic) praise or b) The Potters could be a couple with a famous *negative reputation*, and their reference was intended to make the listening couple aware of the speaker's sarcastic intentions. In case of the latter scenario, '*those Potters*' becomes a target as well.

Another source of error is cases where **the target lies outside the text**. We now describe such examples and compare the impact of these errors with the overall performance.
In our dataset of book snippets, there are 11 texts (5%) with sarcasm target outside the text. In case of tweets, such cases are much higher: 53 tweets (10%). Table 8 compares the results of our hybrid (OR) approach for the specific case of target being 'outside' the text (indicated by 'Outside cases' in the table), with the results on the complete dataset (indicated by 'Overall' in the table). Dice Score (DS) for book snippets is 6.81 for 'outside' cases as compared to 32.68 for the complete dataset. In general, the performance for the 'outside' cases is lower than the overall performance. This proves the difficulty that the 'Outside' cases presents. The EM and DS values for 'Outside' cases are the same by definition. This is because when the target is 'Outside', a partial match and an exact match are the same. Our approach correctly predicts the label 'Outside' for sentences like '*Yeah, just ignore me. That is TOTALLY the right way to handle this!*' However, our approach gives the incorrect output for some examples. For example, for '*Oh, and I suppose the apples ate the cheese*', the predicted target is not 'Outside' (the expected label) but '*I*'. Similarly, for '*Please keep ignoring me for all of senior year. It's not like we're friends with the exact same people*', the incorrectly predicted target is '*me*' instead of the expected label 'Outside'.

	Book Snippets		**Tweets**	
	EM	**DS**	**EM**	**DS**
Overall	7.01	32.68	9.09	39.63
'Outside' cases	6.81	6.81	4.71	4.71

Table 8: Comparison of performance of our approach in case of examples with target outside the text (indicated by 'Outside' cases), with complete dataset (indicated by 'Overall'); EM: Exact Match, DS: Dice Score

7. Conclusion & Future work

In this paper, we introduced a novel problem: sarcasm target identification. This problem aims to identify the target of ridicule in a sarcastic text. This target may be a subset of words in the text or a fall-back label 'Outside'. The task poses challenges such as multiple sarcasm targets or sarcasm targets that may not even be present as words in the sentence. We created and released the first dataset which labels the target of ridicule in sarcastic tweets and book snippets. To motivate automatic identification of sarcasm target, we present an introductory approach which is a hybrid of two kinds of extractors: a rule-based and a statistical extractor. An integrator then combines the outputs of the two extractors in two configurations: OR and AND. We evaluate our approach on annotated tweets as well as book snippets. In general, our hybrid OR system performs the best with a Dice score of 39.63. This is higher than two baselines: a naïve baseline designed for the task, and a baseline based on sentiment target identification. Our hybrid approach is also higher than the two extractors individually. This shows that the two extractors collectively form a good sarcasm target identification approach. We finally present an analysis of errors due to target being outside the text. The dataset and the results obtained from our introductory approach will aid sentiment analysis systems to attribute the negative sentiment expressed via sarcasm to the appropriate entity or aspect.
Our work forms a foundation for future approaches to identify sarcasm targets. As future work, additional rules in the rule-based extractor and novel sets of features in the statistical extractor may be used. Using discourse relations to extract sarcasm targets will be useful. In addition, syntactic dependencies have been found to be useful in case of opinion target extraction (Qiu et al., 2011). Applying these techniques for sarcasm target identification can be useful. A special focus on the 'outside' cases (*i.e.*, cases where the target of ridicule in a sarcastic text is beyond the words present in the sentence) is likely to be helpful for sarcasm target identification.

8. Bibliographical References

Altun, Y., Tsochantaridis, I., and Hofmann, T. (2003). Hidden markov support vector machines. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 3–10.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Campbell, J. D. and Katz, A. N. (2012). Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459–480.

Davidov, D., Tsur, O., and Rappoport, A. (2010). Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 107–116. Association for Computational Linguistics.

González-Ibánez, R., Muresan, S., and Wacholder, N. (2011). Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*, pages 581–586. Association for Computational Linguistics.

Jin, W., Ho, H. H., and Srihari, R. K. (2009). Opinionminer: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1195–1204. ACM.

Joachims, T. (2006). Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226. ACM.

Joshi, A., Sharma, V., and Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection. In *ACL (2)*, pages 757–762.

Joshi, A., Bhattacharyya, P., and Carman, M. J. (2016a). Automatic sarcasm detection: A survey. *arXiv preprint arXiv:1602.03426*.

Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P., and Carman, M. (2016b). Are word embedding-based fea-

tures useful for sarcasm detection? *arXiv preprint arXiv:1610.00883*.

Khattri, A., Joshi, A., Bhattacharyya, P., and Carman, M. (2015). Your sentiment precedes you: Using an author's historical tweets to predict sarcasm. In *Proceedings of the 6th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 25–30.

Lu, B., Ott, M., Cardie, C., and Tsou, B. K. (2011). Multi-aspect sentiment analysis with topic models. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pages 81–88. IEEE.

McAuley, J. J. and Leskovec, J. (2013). From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, pages 897–908. ACM.

Michelson, M. and Knoblock, C. A. (2007). Unsupervised information extraction from unstructured, ungrammatical data sources on the world wide web. *International Journal on Document Analysis and Recognition*, 10(3):211–226.

Mukherjee, A. and Liu, B. (2012). Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 339–348. Association for Computational Linguistics.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 115–124. Association for Computational Linguistics.

Pang, B., Lee, L., et al. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Qiu, G., Liu, B., Bu, J., and Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational linguistics*, 37(1):9–27.

Rajadesingan, A., Zafarani, R., and Liu, H. (2015). Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 97–106. ACM.

Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., and Huang, R. (2013). Sarcasm as contrast between a positive sentiment and negative situation. In *EMNLP*, volume 13, pages 704–714.

Shamay-Tsoory, S. G., Tomer, R., and Aharon-Peretz, J. (2005). The neuroanatomical basis of understanding sarcasm and its relationship to social cognition. *Neuropsychology*, 19(3):288.

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr.*, 5:1–34.

Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., and Li, P. (2011). User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1397–1405. ACM.

Titov, I. and McDonald, R. (2008). A joint model of text and aspect ratings for sentiment summarization. *proceedings of ACL-08: HLT*, pages 308–316.

Tsur, O., Davidov, D., and Rappoport, A. (2010). Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *ICWSM*, pages 162–169.

Veale, T. and Hao, Y. (2010). Detecting ironic intent in creative comparisons. In *ECAI*, volume 215, pages 765–770.

Annotating Opinions and Opinion Targets in Student Course Feedback

Janaka Chathuranga, Shanika Ediriweera,
Ravindu Hasantha, Pranidhith Munasinghe and Surangika Ranathunga
Department of Computer Science and Engineering
University of Moratuwa, Katubedda 10400, Sri Lanka
{janaka.13, shanika.13, ravindu.13, pranidhith.13, surangika}@cse.mrt.ac.lk

Abstract

In this paper, we present a student course feedback corpus, a novel resource for opinion target extraction and sentiment analysis. The corpus is developed with the main aim of summarizing general feedback given by students on undergraduate-level courses. In this corpus, opinion targets, opinion expressions, and polarities of the opinion expressions towards the opinion targets are annotated. Opinion targets are basically the important key points in feedback that the students have shown their sentiment towards, such as "Lecture Slides", and "Teaching Style". The uniqueness of the corpus, annotation methodology, difficulties faced during annotating, and possible usages of the corpus are discussed in this paper.

Keywords: Corpus Annotation, General Student Feedback, Opinion Targets, Summarization, Sentiment

1. Introduction and Motivation

Student feedback is widely used in present in order to enhance the quality of teaching and learning process. Feedback is collected from students as online forms, as well as handwritten documents. Since it takes a considerable effort to read and understand all the feedback given by the students, the best way is to read all the feedback and create a summary that covers all the aspects of the given feedback.

Our intention was to design a system to automatically summarize student feedback for different aspects of a course, such as teaching style of the lecturer, learning environment, and presentation slides. The summarization process contains 3 phases.

1. Identifying and extracting all the opinion targets in the given feedback.
2. Clustering the opinion targets into unique categories.
3. Determining the sentimental polarity of the targets and getting a statistic of polarity for each target cluster.

In order to extract opinions and targets, the most promising technique is supervised learning (Luo & Litman, 2015; Luo, Liu, & Litman, 2016; Luo, Liu, W., Liu, F., & Litman, 2016). In order to apply the supervised machine learning techniques, we need a student course feedback corpus annotated with a suitable annotation scheme.

Our data corpus consists of student responses collected from an undergraduate Computer Science and Engineering course. General responses were collected from 20 lectures and workshops. They contain 973 student responses in total with 2,395 sentences.

In the annotation scheme, we annotate the opinion target and the opinion expression that shows the polarity of the target. For example, in the sentence "The lecture is really good", we annotate "lecture" as the opinion target and "really good" as the opinion expression. Polarity of the target "lecture" becomes positive by the annotated opinion expression.

We used the BIO (Beginning-Inside-Outside) (Sang & Veenstra, 1999) sequence labelling scheme to tag the annotated word phrases. BIO tagging includes "B" tags for the first word in a phrase, "I" tags for the other annotated words inside the phrase, and "O" tags for the words that are not annotated. We used BIO tagging because it is the most promising sequence labelling scheme that is used for many supervised machine learning models such as CRF (Conditional Random Fields) (Luo & Litman, 2015; Luo, Liu, & Litman, 2016; Luo, Liu, W., Liu, F., et al., 2016), in order to extract opinion targets and opinions.

This data corpus is unique because there are no such annotated corpora available for general student feedback. General Feedback means that feedback is taken for every aspect of the lecture by asking a question such as "Give feedback on today's lecture". Therefore, this data corpus contains both positive and negative opinions towards a target.

The rest of the paper is organized as follows. Section 2 describes about the previous research done on the similar contexts. The next section describes about the statistics of dataset and the data collection process. Section 4 describes the pre-processing steps that we carried out. Then Section 5 elaborates on the annotation scheme that we developed in order to feed the data to the machine learning models. Finally, Section 6 concludes the paper with an outlook into the future.

2. Related Work

In order to perform both aspect extraction and sentiment analysis using supervised learning techniques, the dataset has to be properly annotated. For annotation, either sequence labeling or sentence labeling based on aspects have been commonly used.

2.1 Sequence Labeling

Sequence labelling (Erdogan, 2010) is a type of pattern recognition task that involves the algorithmic assignment of a categorical label to each member of a sequence of observed values. In text analysis, tokens are taken as members.

Among the sequence labeling approaches, one of the most promising techniques to extract aspects/ opinion targets is Conditional Random Fields (CRF) (Lafferty, McCallum, & Pereira, 2001). In order to train a CRF model, word phrases are required to be annotated using BIO tags. Previous research (Turian, Ratinov, Bengio, & Turian, 2010) has used this labelling scheme for text chunking and Named Entity Recognition (NER) as well.

Other than BIO, POS tags can be identified as another labelling scheme. It has been used as a feature for highest scoring systems in SemEval-2015 Task 12: Aspect Based Sentiment Analysis (Zhang, Z., & Lan, M., 2015).

2.2 Sentence-level Labeling

In this approach, the dataset is evaluated sentence by sentence, categorizing them based on the aspect and the opinion into predefined labels or topics.

Supervised aspect extraction tasks have also used sentence-level labelling (Pontiki et al., 2016). Unlike sequence labelling that annotates the dataset inline, data used in sentence-level labelling (Pontiki et al., 2016) contain sentences each annotated with one of the pre-defined aspect categories. In instances where more than one aspect is found in a single sentence, the sentence is tagged with all the relevant aspect labels[1].

2. 3 Student Course Feedback

Student course feedback related previous work (Luo & Litman, 2015; Luo, Liu, & Litman, 2016; Luo, Liu, F., Liu, Z., et al.2016) has been done using reflective prompts. Since the polarity of the opinions is implied in the prompt itself, only opinion targets are annotated. Here, a sequence labeling scheme has been used. Moreover, reflective prompts have been designed carefully in such a way that only topics discussed in the class emerge as opinion targets. Human annotators have been used to divide feedback into relevant topics (Luo et al., 2016). In later work (Luo, Liu, & Litman, 2016), the annotation scheme has been improved by introducing a highlighting scheme that assigns a specific colour to similar topics. Then extractive methods such as Integer Linear Programming (Luo et al., 2016), phrase-based approach, clustering, and ranking approaches (Luo, Liu, & Litman, 2016) have been used to summarize student feedback. A dataset that has more than 900 responses for each reflective prompt has been used in this research.

In another work related to student feedback, Welch et al., (2016) have used a dataset consisting of 1,042 responses acquired from a Facebook student group. Only course names and instructor names were annotated as opinion targets with the respective polarity of the opinions towards them as positive or negative.

3. Data Collection

Our student course feedback corpus consists of student responses collected from an undergraduate Computer Science and Engineering Course in a South Asian university. Feedback given for lectures and workshops was collected through an online Learning Management System. The responses are anonymous and the language used is English. All the lectures were done by the same lecturer whereas each workshop was done by different presenters. Statistics of the student course feedback corpus are given in Table 1.

No. of Lectures	8
No. of Workshops	12
Total no. of student responses	973
Total no. of sentences	2,395
Avg. responses per lecture/workshop	48.65
Max. responses in a lecture/workshop	81
Min. responses in a lecture/workshop	7

Table 1: Statistics of the corpus

The prompts we used to collect responses were general prompts such as "What is your opinion about today's lecture?". Therefore, students had the freedom to write regarding any aspect of the lecture, unlike what they would write in response to a reflective prompt. In addition, there was no sentence limitation for providing feedback. The outstanding quality of this corpus is that it has student feedback that consists of both positive and negative opinions and comments about all the aspects of the course such as lecturer's qualities, course material, course content, and learning environment.

The sentiment of feedback about certain aspects such as "lecturer" tends to be biased towards positive. This nature could be a characteristic of south Asian cultural background.

4. Pre-Processing

In a country where English is considered as a 3rd language, it is not so surprising for students to make many errors when writing in English. Furthermore, unlike in other situations (e.g. writing a project report), since while giving feedback students deliberately do not worry about using correct English, there were many spelling mistakes in the corpus. Out of the 973 responses, 44.3% of the responses had spellings mistakes. In certain cases, we noticed that the meaning of the complete sentence gets distorted because of the spelling mistakes. For example, some students have used "Work shop is fine" instead of "Workshop is fine", which gives a very different meaning compared to what is intended by the student.

[1]SemEval-2016 Task 5 Available at: http://alt.qcri.org/semeval2016/task5/index.php?id=data-and-tools

Therefore, correcting the spelling mistakes beforehand the annotation phase became very critical for target and sentiment extraction. Thus, we carried out context-sensitive spell correction ("Microsoft Cognitive Services— Bing Spell Check API | Microsoft Azure," n.d.) on the corpus . A few examples for the corrections made are "uncomplete" to "incomplete", "some times" to "sometimes", "work shop" to "workshop", "people who came their without any Java knowledge" to "people who came there without any Java knowledge" and "jawa" to "java".

5. Annotation Scheme

The dataset was annotated from scratch by 4 undergraduate Computer Science students. The resulting annotations were then inspected again, and corrected (if needed). Borderline cases were resolved collaboratively by annotators.

This dataset consists of many opinionated responses. Each of those responses focuses their opinion towards a target entity or an aspect of an entity, which is called an opinion target. The phrase that carries the opinion is called an opinion expression.

We used a novel way of annotating student feedback. This was required mainly because of the nature of the data. In previous work (Luo, Liu, & Litman, 2016), data related to student feedback only had opinion targets in them whereas the positive / negative expressions were in the prompt itself. The following scheme was created to annotate the student course feedback corpus that contains responses with both opinion targets and positive/ negative expressions.

5.1 Basic Annotation Rule

We annotated important points in the response as opinion targets. Then we annotated the phrase that contains the student's opinion towards the opinion target as a positive opinion expression or a negative opinion expression, considering the polarity of the expression. For example, consider the sentence "Lectures are really good". Here, "Lectures" is the target and "really good" is the opinion expression, which is positive**.**

5.2 Annotating Pronouns

Most of the time, students refer to important entities using pronouns.

E.g.: - 1) It is good
2) It was very good to have an in-class assignment and it motivated us.

In the first example, the word "It" refers to the lecture itself. In the second example, first "It" is not an opinion target because it refers to "having an in-class assignment", which is explicitly mentioned in the sentence. Therefore, it is not needed to annotate the first "It". The second 'it" refers to "in-class assignment". Our annotation scheme is designed to annotate these pronouns. Later they will be resolved for exact entities using co-reference resolution.

5.3 Multi-word opinion targets

Consider the sentence "I think time and weight for documentation of the project is too much". Here, opinion target is "time and weight for documentation of the project", which has a negative opinion.

5.4 Single target, single expression

Consider the sentence "Lectures were really good." Here, the target is "Lectures" and the positive expression towards it is "really good".

5.5 Single target, multiple expressions

Consider the sentence "Overall lecture session was great, well organized and very helpful". Here, the target "overall lecture session" has three positive expressions towards it.

5.6 Opinion target with both positive and negative opinions

Consider the sentence "Lecture was good but a little bit fast".

Here, the student expresses his opinions about the opinion target "lecture" where the student has two opinions towards the "lecture". The opinion expression "good" expresses a positive opinion and the opinion expression "a little bit fast" expresses a negative opinion.

5.7 Single expression, multiple opinion targets

Consider the sentence "Keeping interactions with students, asking questions, giving in class activities and discussing them within the class were greatly helpful for me to develop my oop skills".

A positive opinion is expressed here for all the following aspects/ targets of the lecture: "keeping interactions", "asking questions", "giving in class activities".

5.8 Ambiguity about which opinion target to take

To resolve the matters in ambiguity while annotating the dataset, we had to come up with a hierarchy based on the entity-aspect relationship to select the best suitable target to annotate. Figure 1 explains this hierarchy.

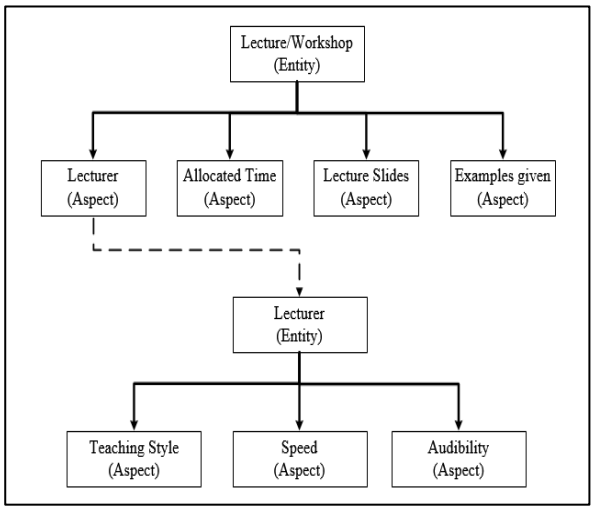


Figure 1: Example opinion target hierarchy

Aspect of one entity-aspect relationship can become the entity of another entity-aspect relationship, forming a multi-level hierarchy as shown in the Figure 1, where "Lecturer" is identified as both an aspect and an entity.

If we can identify both the entity and the aspect separately, we considered annotating both of them as targets. For example, consider the sentence "The lecture is good but the

time is not enough". Here we annotated both "lecture" (entity) and "time" (aspect) as opinion targets.

But in the cases where we cannot separate the entity and the aspect, we only annotated the higher valued target in the hierarchy, which is the target.

For example, consider the sentence "Both lecturers did a great job on delivering the subject matter." Here, out of two opinion targets that can be identified: "Both lecturers" (entity), and "delivering the subject matter" (aspect), it was difficult to decide on which target the opinion was focused on. To resolve this matter, we used the above hierarchy to prioritize the targets and annotated with the higher valued target. Since "delivering the subject matter" is done by lecturers, it becomes an aspect of "Both lecturers". Therefore "Both lecturers" is in a higher level of the hierarchy and has a higher priority. So accordingly, we annotated "Both lecturers" as the target.

We annotated the dataset manually using the above-mentioned method. This annotation scheme first identifies sentences or phrases with opinions and then marks the opinion target. Finally, the annotated phrases are converted into BIO tags. Since we annotated both the targets and the expressions, we used the following notation.

- B-T: Beginning of the Target
- I-T: Inside of the Target
- B-PO: Beginning of the Positive Opinion
- I-PO: Inside of the Positive Opinion
- B-NO: Beginning of the Negative Opinion
- I-NO: Inside of the Negative Opinion

For example, the sentence "Lectures were really good" was annotated as shown in Table 2.

Lectures	were	really	good.
B-T	O	B-PO	I-PO

Table 2: BIO tagging example

5.9 Statistics of the annotated corpus

Statistics of the student course feedback corpus annotations are given in Table 3. Out of 2,395 sentences in the corpus, 1,780 sentences contain 2,125 opinion targets.

Total no. of Opinion targets	2,125
Avg. opinion targets per lecture/ workshop	106
Max. opinion targets in a lecture/ workshop	199
Min. opinion targets in a lecture/ workshop	9

Table 3: Statistics of Annotations

Statistics of unique targets in the annotated data set are given in Table 4. Only 29 out of 106 targets were unique from an average course feedback set.

Avg. unique targets per lecture/workshop	29
Max. unique targets per lecture/workshop	54
Min. unique targets per lecture/workshop	4

Table 4: Statistics of unique targets

Distribution of the opinion targets in the student course feedback corpus annotation is depicted in Figure 2.

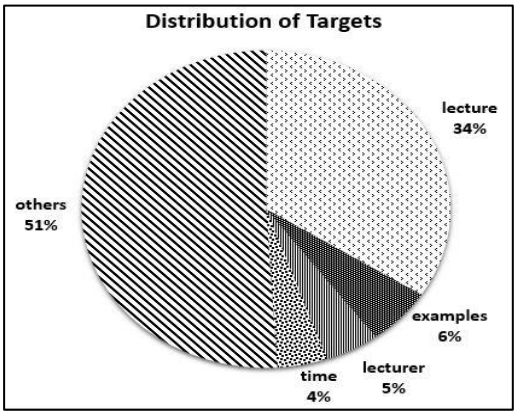


Figure 2: Distribution of opinion targets

Most frequently mentioned opinion targets by the students are "lecture", "examples", "lecturer" and "time". "Others" category which cannot be resolved to any of the hierarchies thus not shown in Figure 1, contains all medium and low frequent opinion targets, where more than 50% of the opinion targets are mentioned around 5-20 times. Least frequently mentioned opinion targets are "homework", "book", "board", "UML", "fundamentals of TDD", "log4j", "Junit", "threads" and "databases". Most of these are specific to a single workshop/lecture. These low frequent opinion targets are difficult to identify using supervised learning approaches. If we considered a fixed set of aspect categories rather than annotating each different target, less frequent targets will not be extracted. It was the reason to follow a novel annotation scheme as described in section 5.

The reliability of the annotations was verified using inter-annotator reliability from which the percent agreement was 89.2%, and the Kappa (Cohen, 1960) was 0.616. The percent agreement was calculated by token wise considering whether it is within an opinion target or an opinion expression.

The high ambiguity of the dataset resulted in this lower Kappa statistic. For example, consider the sentence "The first 7 lectures have been really good". In this example, one annotator has annotated "first 7 lectures", while the other one has annotated only "lectures" as the target. In general, most of these conflicts are due to the ambiguity of aspects and entities. Since we are annotating targets, they could be entities as well as nouns. In some cases, for annotators it is a difficult task to decide whether to annotate noun or the entity. In some cases, annotating both could be appropriate. This is because the ambiguity and noise of general feedback. Therefore, inter-annotator agreement and Kappa

value is quite low compared to reflective prompt-based annotations.

6. Conclusion

In this work, we have focused on annotating a corpus of general student feedback, which contains more noise and complex data compared to feedback collected through reflective prompts. However, unlike traditional reflective prompts, general feedback contains more useful information. Therefore, it is important to analyse general feedback. Annotating general feedback is a challenging task because of the ambiguity and noise. Here we proposed a simple annotation scheme with clarity to annotate general feedback for sentiment analysis. We have addressed the ambiguity and noise with various solutions in this dataset of 20 feedback sets with 973 responses.

The main aim of the corpus is to be used for opinion target extraction, polarity detection, and summarization of general student feedback. Since this dataset contains both positive and negative sentiments for a target, this dataset can be used to extract both positive and negative sentiments towards different aspects of a course unlike in other related corpora. We believe that this corpus can be used to improve and explore features to be used for target extraction using sequence labelling.

7. Bibliographical References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*(Jan), 993–1022.

Brody, S., & Elhadad, N. (2010). An unsupervised aspect-sentiment model for online reviews. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 804–812.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*(1), 37–46.

Erdogan, H. (2010). Sequence labeling: Generative and discriminative approaches hidden markov models, conditional random field and structured svm. *Proceedings of the Tutorial at International Conf. Machine Learning and Applications*, 726–733.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 168–177.

Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of Intl. Conf. on Machine Learningm*, 282-289.

Luo, W., & Litman, D. (2015). Summarizing Student Responses to Reflection Prompts. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1955–1960.

Luo, W., Liu, F., & Litman, D. (2016). An Improved Phrase-based Approach to Annotating and Summarizing Student Course Responses. *Proceedings of the 26th International Conference on Computational Linguistics (COLING-16)*, 53–63.

Luo, W., Liu, F., Liu, Z., & Litman, D. (2016). Automatic Summarization of Student Course Feedback. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 80–85.

Microsoft Cognitive Services—Bing Spell Check API | Microsoft Azure. (n.d.). Retrieved September 9, 2017, from https://azure.microsoft.com/en-us/services/cognitive-services/spell-check/

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub,M., Zhao, et al. (2016). SemEval-2016 Task 5: Aspect Based Sentiment Analysis. *Proceedings of the 10th International Workshop on Semantic Evaluation*, 19–30.

Saif, H., Fernandez, M., He, Y., & Alani, H. (2013). Evaluation datasets for Twitter sentiment analysis a survey and a new dataset, the STS-Gold. *Proceedings of the CEUR Workshop*, *1096*, 9–21.

Sang, E. F., & Veenstra, J. (1999). Representing text chunks. *Proceedings of the Ninth Conference on European Chapter of the Association for Computational Linguistics*, 173–179.

Turian, J., Ratinov, L., Bengio, Y., & Turian, J. (2010). Word Representations: A Simple and General Method for Semi-supervised Learning. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–394.

Welch, C., Mihalcea, R., Street, H., & Arbor, A. (2016). Targeted Sentiment to Understand Student Comments. *Proceedings of the 26th International Conference on Computational Linguistics*, (1), 2471–2481.

Zhang, Z., & Lan, M. (2015). Ecnu: Extracting effective features from multiple sequential sentences for target-dependent sentiment analysis in reviews. *Proceedings of the 9th International Workshop on Semantic Evaluation* (SemEval 2015), 736-741.

Generating a Gold Standard for a Swedish Sentiment Lexicon

Jacobo Rouces, Nina Tahmasebi, Lars Borin, Stian Rødven Eide
Sprakbanken, University of Gothenburg
{jacobo.rouces, nina.tahmasebi, lars.borin, stian.rodven.eide}@gu.se

Abstract

There is an increasing demand for multilingual sentiment analysis, and most work on sentiment lexicons is still carried out based on English lexicons like WordNet. In addition, many of the non-English sentiment lexicons that do exist have been compiled by (machine) translation from English resources, thereby arguably obscuring possible language-specific characteristics of sentiment-loaded vocabulary. In this paper we describe the creation of a gold standard for the sentiment annotation of Swedish terms as a first step towards the creation of a full- fledged sentiment lexicon for Swedish – i.e., a lexicon containing information about *prior* sentiment (also called polarity) values of lexical items (words or disambiguated word senses), along a scale negative–positive. We create a gold standard for sentiment annotation of Swedish terms, using the freely available SALDO lexicon and the Gigaword corpus. For this purpose, we employ a multi-stage approach combining corpus-based frequency sampling and two stages of human annotation: direct score annotation followed by Best-Worst Scaling. In addition to obtaining a gold standard, we analyze the data from our process and we draw conclusions about the optimal sentiment model.

Keywords: sentiment analysis, Swedish, gold standard, lexical resource

1. Introduction

There is an increasing demand for multilingual sentiment analysis, and most work on sentiment lexicons is still carried out based on English lexicons like WordNet (Fellbaum, 1998). In addition, many of the non-English sentiment lexicons that do exist have been compiled by (machine) translation from English resources, e.g., by Mohammad and Turney (2010)[1] and Chen and Skiena (2014), thereby arguably obscuring possible language-specific characteristics of sentiment-loaded vocabulary.

In this paper we describe the creation of a gold standard for the sentiment annotation of Swedish terms as a first step towards the creation of a full- fledged sentiment lexicon for Swedish – i.e., a lexicon containing information about *prior* sentiment (also called polarity) values of lexical items (words or disambiguated word senses), along a scale negative–positive. For this purpose, we use human annotations of items sampled from a general-purpose computational lexical resource. More specifically, we employ a multi-stage approach combining corpus-based frequency sampling, direct score annotation and Best- Worst Scaling (BWS) (Kiritchenko and Mohammad, 2016).

2. State of the art

We base our gold standard on *SALDO* (Språkbanken, 2015a), which is an existing Swedish lexical-semantic computational resource (Borin et al., 2013). It is organized as a lexical-semantic network of word senses, whose topology reflects semantic distance among the word senses. Each word sense in SALDO is additionally connected to one or more form units (lemmas plus part of speech and full inflectional and compounding information). These are formally organized as an independent lexical resource – *SALDO's Morphology* (Språkbanken, 2015b) – which consequently can be used in NLP applications independently of SALDO, e.g., for lemmatization and morphological analysis of Swedish text. For the work described here, we use SALDO v. 2.3, which contains 131,020 word senses. SALDO is freely available (under a CC-BY license).

Different ways of modeling sentiment for a word sense or unit of text are possible.

The simplest (but not necessarily the less appropriate) model is the bipolar model, which assigns to each lexical unit a scalar, which is often normalized in the interval $[-1, +1]$, with -1 representing the most negative possible sentiment, and $+1$ the most positive.

SentiWordNet (Baccianella et al., 2010) and its gold standard Micro-WNOp (Cerini et al., 2007) use a model with two degrees of freedom. Each semantic unit in WordNet (Fellbaum, 1998) is assigned a three-dimensional vector (pos, neg, neu) with positive, negative and neutral components, normalized so that $\text{pos}+\text{neg}+\text{neu} = 1$ (this effectively gives 2 degrees of freedom). This model can be trivially converted to the previous one using $\text{sen} = \text{pos} - \text{neg}$.

3. Annotation

We aim to have a gold standard that assigns a sentiment to each SALDO entry. The bipolar sentiment model should be supported, but we also want to investigate the feasibility and convenience of using the SentiWordNet model.

First, an initial sampling from SALDO was done following the distribution given by the estimated frequency of each word sense in the Gigaword corpus (Eide et al., 2016), which is a one-billion-word mixed-genre corpus of written Swedish (Språkbanken, 2016).[2] Due to the Zipfian distribution of many kinds of linguistic items (Baayen, 2001), the gold standard would include, like the underlying lexicon SALDO does, mostly words that occur very rarely in written text, including rather obscure and outdated terms, as the lexicon has been designed to cover a time period from the mid-20th century until today. Thus, by using a sampling based on corpus of the last two decades, the gold standard

[1] http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

[2] The corpus is freely available under a CC-BY license.

becomes more representative of modern written language. By filtering out obscure and dated terms, we also reduce the proportion of terms that the annotators may not understand. Having annotators directly assign continuous sentiment scores (be it (sen) or (pos,neg,neu)) to lexicon entries has several issues. It is difficult for annotators to remain consistent throughout their own annotation and across themselves; this is a rationale for instead relying on *Best-Worst Scaling* BWS annotation (Kiritchenko and Mohammad, 2016). With BWS, annotators are presented tuples (usually 4-tuples) of elements to annotate, and they select the highest and lowest according to the score at hand (in this case, the most positive and the most negative). If certain statistical properties are ensured about the appearance of elements in the tuples, then the number of times an element is chosen as most positive minus the number of times it is chosen as most negative can be used as a sentiment score.

However, we experienced that if the BWS annotation is performed by directly sampling from the lexicon (or from a general corpus, for that matter), most 4-tuples would not contain any items with a clear non-neutral polarity, let alone one most positive and one most negative item. Increasing the size of the tuples could solve this, but would imply a higher cognitive load for the annotator. Our solution to this problem is pre-filtering the initial set of terms by means of a preceding direct, but coarse-grained annotation that allows us to feed into the BWS annotation a subset of word senses with a more even distribution of sentiment values.

3.1. Direct Annotation

The initial sampling from SALDO was performed following the distribution given by the estimated frequency of each word senses in the Gigaword corpus (Eide et al., 2016), which is a one-billion-word corpus of Swedish text comprising newspaper and scientific articles, government reports, fiction and social media.[3] We used the subset of the corpus containing text written from 1990 to the present. Because the tokens in the corpus are not sense-disambiguated, we followed a simple heuristic. The different word senses for a given lemma are not annotated for their corpus frequency in SALDO, but the first sense is by design the most common one.

Because the most common sense for a lemma in SALDO tends to be the referred sense around 70% of the time (Nieto Piña and Johansson, 2016) (this figure is also a good approximation for other sense-disambiguation tasks in general (Gale et al., 1992; Kilgarriff, 2004)), we assume a distribution where the first sense is given a probability of $\hat{p} = 0.7$, and each of the n remaining ones are given $\hat{p} = 0.3/n$. This provides a reasonable approximation for the zipfian distribution, whose biggest differential is between the first and the second most common elements.

Using this distribution, we independently sampled 1998 word senses from SALDO, creating the set of words that would be annotated directly, W_{DA}. Each sample was performed independently, without replacement, using as probability distribution the normalized count, $c/\sum c$. The sampling was filtered in order to avoid having too many difficult-to-judge non-content items (SALDO contains all parts of speech) in the annotation set. We also left out all multi-word expressions and single-letter lemmas (typically corresponding to the names of letters of the alphabet, musical notes, or units of measurement). Thus only single-word adjectives, interjections, nouns, and verbs, having a lemma two letters or longer were sampled.

We also sampled 200 additional word senses that were used for a joint annotation exercise across all annotators of W_{DA}, with the purpose of standarizing the annotation criteria to a reasonable degree.

After the joint annotation, each of the three annotators independently assigned a label to each word sense in W_{DA}. The possible labels are "positive","negative" or "neutral". All three annotators are language technology/ natural language processing researchers with formal backgrounds in linguistics and computer science, and native-level knowledge of Swedish.

The annotated value of each word $w \in W_{\text{DA}}$ was calculated as for Equation 1, where A_{DA} is the set of DA annotators.

$$\text{sen}_{\text{DA}}(w) = \frac{\sum_{a \in A_{\text{DA}}} l_{\text{DA}}(a, w)}{|A_{\text{DA}}|}$$

$$l_{\text{DA}}(a, w) = \begin{cases} 1 & \text{if } a \text{ annotated } w \text{ as positive} \\ 0 & \text{if } a \text{ annotated } w \text{ as neutral} \\ -1 & \text{if } a \text{ annotated } w \text{ as negative} \end{cases} \quad (1)$$

3.2. Best-Worst Scaling Annotation

For the BWS annotation, we selected the set W_{BWS} defined as those elements from W_{DA} that had been labeled as non-neutral (positive or negative) by at least two annotators (Equation 2), which ensured that most 4-tuples had clear candidates for most positive and most negative.

$$W_{\text{BWS}} = \{w : w \in W_{\text{DA}} \wedge |\text{sen}_{\text{DA}}(w)| \geq 2/3\} \quad (2)$$

Since $|W_{\text{BWS}}| = 278$, we generated 572 4-tuples, which is greater than $|W_{\text{BWS}}| \cdot 2$ and thus largely sufficient for BWS (Kiritchenko and Mohammad, 2016).

We developed a web application that allows annotators to assign sentiments to SALDO word senses, using Best-Worst Scaling. The user can select the most positive and most negative SALDO entry for each tuple, including an 'I don't know' option. It includes an interactive menu of pending groups, and the ability to save and load partial annotations to and from local files, allowing the annotators to spread their work over several sessions.

Word senses in SALDO do not have definitions or glosses, so in order for the user to be able to distinguish different senses of a lemma, we include a list of other lemmas that correspond to word senses which are associated to the item at hand. These are obtained using the semantic-network structure of SALDO.

Figure 1 includes a screenshot of the application. It is publicly available.[4]

[3] The corpus is freely available (under a CC-BY license) at `https://spraakbanken.gu.se/eng/resource/gigaword`.

[4] `http://demo.spraakdata.gu.se/jacobo/bws-annotation/main.html`

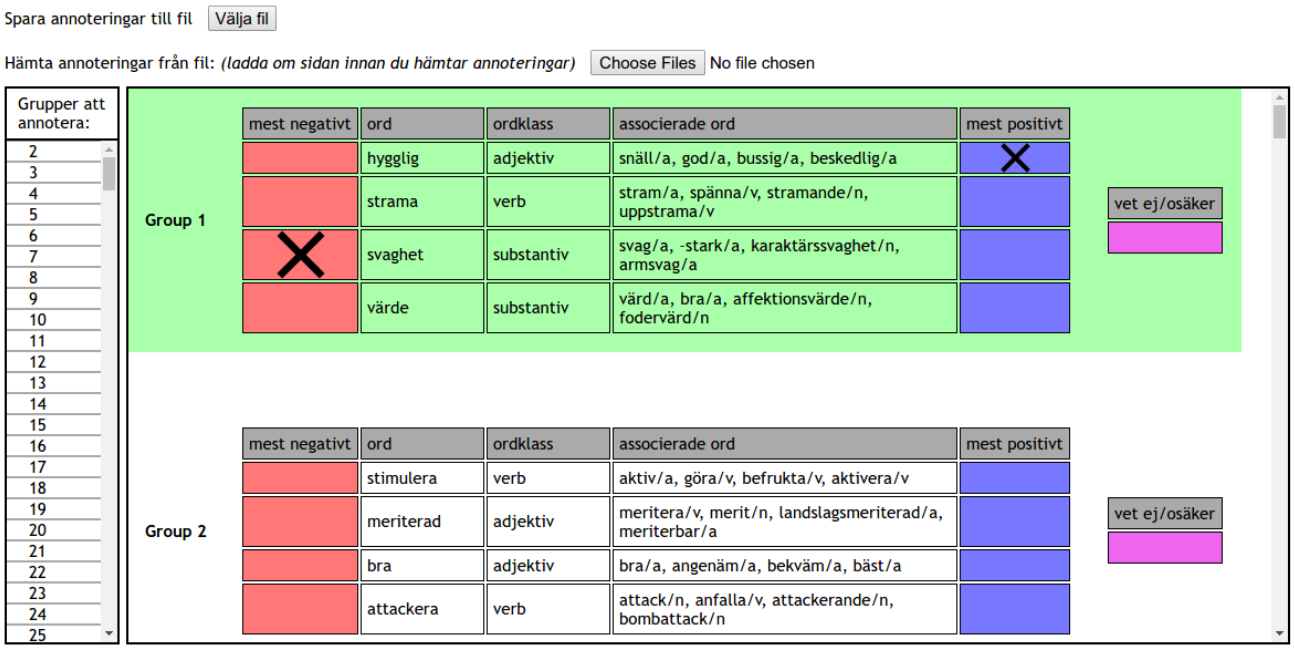


Figure 1: Screenshot for the Best-Worst Scaling annotation interface. The labels for each group are 'most negative', 'word', 'part of speech', 'associated words', 'most positive', 'don't know/uncertain' from left to right.

We employed 4 annotators, who were different from the previous ones but also had formal background in (computational) linguistics and/or computer science, as well as native-level knowledge of Swedish.

The annotated values of each word $w \in W_{\text{BWS}}$ were calculated as for Equation 3, where A_{BWS} is the set of BWS annotators and $T(w)$ is the set of 4-tuples that contain the SALDO entry w. sen_{BWS} corresponds to the typical output used when applying BWS (either for sentiment analysis or for any other type of scaling). However, in Section 4. we will also analyze pos_{BWS}, neg_{BWS}, and neu_{BWS}, in order to determine whether they could be used to obtain a representation under the SentiWordNet model.

$$
\begin{aligned}
\text{pos}_{\text{BWS}}(w) &= \frac{\sum_{a \in A_{\text{BWS}}} \sum_{t \in T(w)} l^{\text{pos}}_{\text{DA}}(a,t,w)}{|A_{\text{BWS}}| \cdot |T(w)|} \\
l^{\text{pos}}_{\text{BWS}}(a,t,w) &= \begin{cases} 1 & \text{if } a \text{ annotated } w \text{ as most positive in } t \\ 0 & \text{otherwise} \end{cases} \\
\text{neg}_{\text{BWS}}(w) &= \frac{\sum_{a \in A_{\text{BWS}}} \sum_{t \in T(w)} l^{\text{neg}}_{\text{DA}}(a,t,w)}{|A_{\text{BWS}}| \cdot |T(w)|} \\
l^{\text{neg}}_{\text{BWS}}(a,t,w) &= \begin{cases} 1 & \text{if } a \text{ annotated } w \text{ as most negative in } t \\ 0 & \text{otherwise} \end{cases} \\
\text{neu}_{\text{BWS}}(w) &= 1 - \text{pos}_{\text{BWS}}(w) - \text{neg}_{\text{BWS}}(w) \\
\text{sen}_{\text{BWS}}(w) &= \text{pos}_{\text{BWS}}(w) - \text{neg}_{\text{BWS}}(w)
\end{aligned}
\tag{3}
$$

4. Results

4.1. Interannotator agreement

Table 2 shows the interannotator agreement for the two annotation stages: direct and BWS.

We chose Krippendorff's alpha instead of Cohen's or Fleiss's kappa, because in addition to the nominal metric, it allows for an interval metric as well. The interval metric takes into account that some annotation labels are more similar than others, i.e. the set of labels can be bijected to some metric space.

Under the interval metric, the inter-annotator agreement for BWS outperforms the one obtained from direct annotation, specially for the sen_{BWS}, which is the most comparable variable to direct annotation.

	nominal	**interval**
$\text{sen}_{\text{DA}}(w)$	0.480	0.529
$\text{pos}_{\text{BWS}}(w)$	0.551	0.889
$\text{neg}_{\text{BWS}}(w)$	0.621	0.893
$\text{neu}_{\text{BWS}}(w)$	0.446	0.744
$\text{sen}_{\text{BWS}}(w)$	0.462	0.927

Table 2: Interannotator agreements (Krippendorff's alpha, nominal and interval) for scores obtained from best-worst scaling (BWS) and direct annotation (DA). Since we used three annotators for sen_{DA}, in order to make the Krippendorff's alpha values comparable, we take the first 3 of the 4 annotators we used for BWS.

4.2. Annotation results

Figure 3 shows the histogram of the values obtained from the direct annotation stage. Figures 4a and 4b show the histogram of the values obtained from the BWS annotation, and Table 3 shows basic statistics. The data is publicly available under an open-source CC-BY 4.0 license.[5]

[5] http://demo.spraakdata.gu.se/jacobo/bws-annotation/data

w	gloss	$\mathbf{pos_{BWS}}(w)$	$\mathbf{neg_{BWS}}(w)$	$\mathbf{neu_{BWS}}(w)$	$\mathbf{sen_{BWS}}(w)$
medalj..1	'medal'	0.4722	0.0000	0.5278	0.4722
lugna..1	'calm (v)'	0.3333	0.0000	0.6667	0.3333
svår..1	'difficult'	0.0500	0.3250	0.6250	-0.2750
gnista..3	'spark, spunk'	0.5938	0.0000	0.4062	0.5938
slippa..1	'be spared'	0.2500	0.1944	0.5556	0.0556
depression..2	'depression	0.0000	0.4688	0.5312	-0.4688
möda..1	'difficulty'	0.0312	0.2500	0.7188	-0.2188
stimulera..1	'stimulate'	0.1250	0.0000	0.8750	0.1250
världsmästare..1	'world champion'	0.5312	0.0000	0.4688	0.5312
absurd..1	'absurd'	0.0625	0.4375	0.5000	-0.3750
rätt..1	'correct (a)'	0.3125	0.0000	0.6875	0.3125
tryck..4	'pressure'	0.1389	0.0000	0.8611	0.1389
protest..1	'protest (n)'	0.1875	0.3125	0.5000	-0.1250
kraftfull..1	'powerful'	0.3750	0.0000	0.6250	0.3750
överdrift..1	'exaggeration'	0.1786	0.1429	0.6786	0.0357
mista..1	'lose'	0.0833	0.5833	0.3333	-0.5000
trög..2	'sluggish'	0.0500	0.2250	0.7250	-0.1750
misstanke..1	'suspicion'	0.0750	0.3250	0.6000	-0.2500
hyllning..1	'tribute'	0.8750	0.0000	0.1250	0.8750
förbud..1	'prohibition'	0.0000	0.4286	0.5714	-0.4286

Table 1: Examples of sentiment scores obtained from BWS.

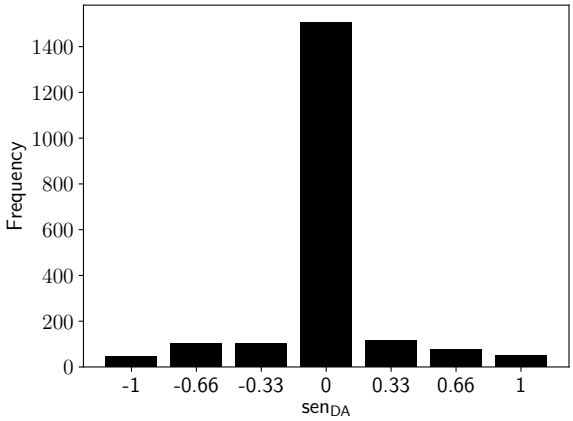
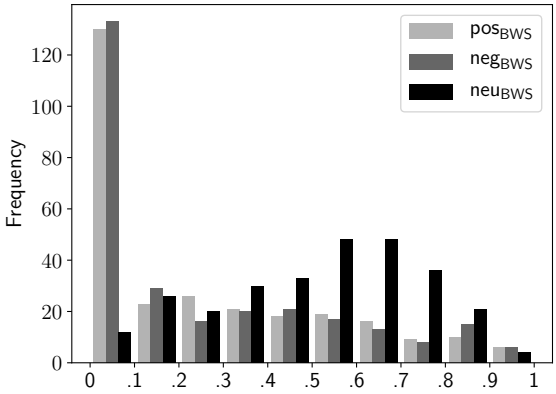
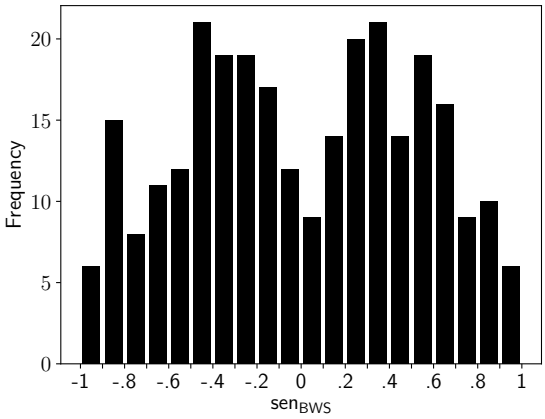


Figure 3: Histogram of the values obtained from the direct annotation stage.



(a) Histogram of the (pos,neg,neu) values

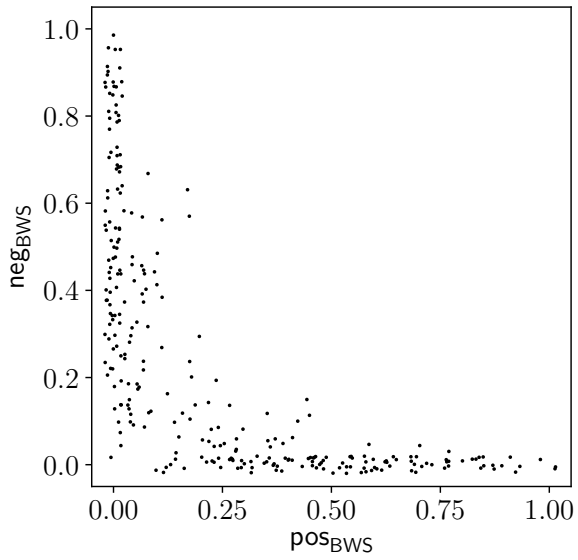


(b) Histogram of the sen value

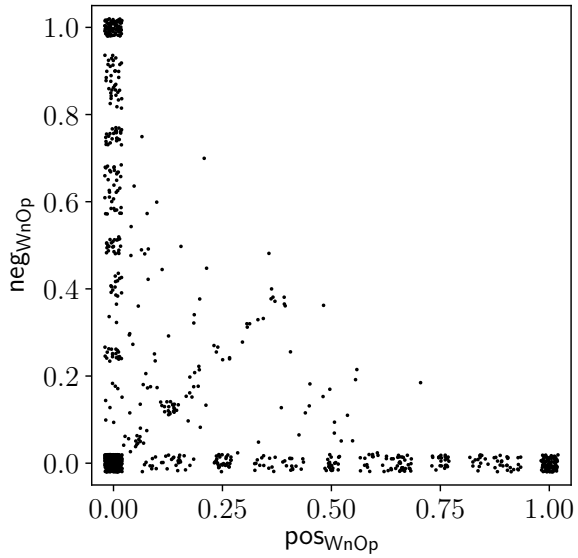
Figure 4: Histogram of the values obtained from the BWS annotation

	pos_{BWS}	neg_{BWS}	neu_{BWS}	sen_{BWS}
mean	0.248	0.246	0.505	0.002
std.	0.278	0.288	0.228	0.518

Table 3: Mean and standard deviation for values obtained from the BWS annotation.



(a) Scatterplot of the BWS annotation



(b) Scatterplot of Micro-WNOp

Figure 2: Scatterplots of (pos,neg) values

4.3. Choice of Sentiment Model

The output of the BWS annotation could be used both for the SentiWordNet and the bipolar model ($\text{sen}(w) = \text{sen}_{\text{BWS}}(w), \text{pos}(w) = \text{pos}_{\text{BWS}}(w), \ldots$). In this section we analyze our data and the data from the gold standard used for building SentiWordnet, in order to determine whether the SentiWordNet model offers some advantage in return for its added complexity.

From the results of the BWS annotation, 86 of 278 SALDO ids have $\text{pos}_{\text{BWS}}(w) > 0$ and $\text{neg}_{\text{BWS}}(w) > 0$, but in many cases one of these components is small and a strong bias is common. The average over w of the value $\min(\text{pos}_{\text{BWS}}(w), \text{neg}_{\text{BWS}}(w))$, which reflects the overlap between the positive and negative components, is 0.022. In contrast, for Micro-WNOp, the gold standard used for SentiWordNet, which uses the same model but was obtained from direct annotation of the two variables 'pos' and 'neg', it is 0.015. Our higher value is probably due to the fact that we made W_{BWS} with a high proportion of non-neutral word senses, and therefore, a non-negligible proportion of the BWS 4-tuples contained elements that either were all negative or all positive, making the choice for most positive or most negative a sort of "lesser evil" or "lesser good", respectively. As an example, *absurd* from Figure 1, appeared in the annotation interface in a tuple containing [*dålig* 'bad' , *utplåna* 'obliterate' , *irriterad* 'irritated' , *absurd*].

Figure 2a shows a scatterplot of the BWS annotation results adapted to the SentiWordNet model, using uniform ($[-0.02, 0.02]$) and independently distributed dithering. Figure 2b shows the equivalent plot for Micro-WNOp. The consistently low overlap between negative and positive components seems to indicate that the multi-dimensional SentiWordNet model is not necessary, or at least does not offer sufficient advantages to outweigh the simplicity and efficiency of the bipolar model.

5. Extensions and Future Work

The obtained gold standard is now being used to train and compare different lexicon-based algorithms for creating a complete sentiment lexicon for Swedish. In particular, we have experimented with growing sentiment lexicons based on a set of initial items and the lexical-semantic network structure of resources such as SALDO, plus contextual information from large corpora. We describe this work in more detail in Rouces et al. (2018). The resulting lexical resource, *SenSALDO* (Språkbanken, 2018), is freely available under a CC-BY license.

In parallel, we are also considering translating sentiment lexicons (from English). This is still future work, which will provide us with an opportunity to compare the results of the two approaches of building a Swedish sentiment lexicon from scratch based on monolingual resources, or of basing it on translation of an existing sentiment lexicon for another language.[6]

The more linguistic aspects of our work on the gold standard are treated in a companion publication to the present paper (Rouces et al., forthcoming), where we also discuss potential applications of (Swedish) sentiment lecicons in text mining for digital humanities research,

Acknowledgements

This work has been supported by a framework grant (*Towards a knowledge-based culturomics*[7] – contract 2012-5738) as well as funding to Swedish CLARIN (*Swe-Clarin*[8] – contract 2013-2003), both awarded by the Swedish Research Council, and by infrastructure funding granted to Språkbanken by the University of Gothenburg.

6. Bibliographical References

Baayen, R. H. (2001). *Word frequency distributions*. Kluwer Academic Publishers, Dordrecht.

[6] In the literature, these two approaches are often referred to as the *merge* (monolingual) and *expand* (translation) approach, respectively (Vossen, 1998).

[7] `https://spraakbanken.gu.se/eng/culturomics`

[8] `https://sweclarin.se/eng`

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of LREC*, pages 2200–2204.

Borin, L., Forsberg, M., and Lönngren, L. (2013). SALDO: A touch of yin to WordNet's yang. *Language Resources and Evaluation*, 47(4):1191–1211.

Cerini, S., Compagnoni, V., Demontis, A., Formentelli, M., and Gandini, C. (2007). Micro-WNOp: A gold standard for the evaluation of automatically compiled lexical resources for opinion mining. In Andrea Sansò (ed), *Language Resources and Linguistic Theory*, pages 200–210. Franco Angeli, Milano.

Chen, Y. and Skiena, S. (2014). Building sentiment lexicons for all major languages. In *Proceedings of ACL 2014*, pages 383–389, Baltimore. ACL.

Eide, S. R., Tahmasebi, N., and Borin, L. (2016). The Swedish culturomics gigaword corpus: A one billion word Swedish reference dataset for NLP. In *Proceedings of the* From Digitization to Knowledge *workshop at DH 2016, Kraków*, pages 8–12, Linköping. LiUEP.

Fellbaum, C. (ed). (1998). *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, Mass.

Gale, W., Church, K. W., and Yarowsky, D. (1992). Estimating upper and lower bounds on the performance of word-sense disambiguation programs. In *Proceedings of ACL 1992*, ACL '92, pages 249–256, Newark. ACL.

Kilgarriff, A. (2004). How dominant is the commonest sense of a word? In Petr Sojka, et al. (eds), *Text, Speech and Dialogue*, pages 103–111, Berlin. Springer.

Kiritchenko, S. and Mohammad, S. M. (2016). Capturing reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proceedings of NAACL 2016*, pages 811–817, San Diego. ACL.

Mohammad, S. and Turney, P. (2010). Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles. ACL.

Nieto Piña, L. and Johansson, R. (2016). Embedding senses for efficient graph-based word sense disambiguation. In *Proceedings of TextGraphs-10*, pages 1–5, San Diego. ACL.

Rouces, J., Tahmasebi, N., Borin, L., and Eide, S. R. (2018). SenSALDO: Creating a sentiment lexicon for Swedish. In *Proceedings of LREC 2018*, Miyazaki. ELRA.

Rouces, J., Borin, L., Tahmasebi, N., and Eide, S. R. (forthcoming). Defining a gold standard for a Swedish sentiment lexicon: Towards higher-yield text mining in the digital humanities. In *Proceedings of DHN 2018, Helsinki*, Aachen. CEUR WS.

Vossen, P. (1998). Introduction to EuroWordNet. *Computers and the Humanities*, 32(2–3):73–89.

7. Language Resource References

Språkbanken. (2015a). *SALDO*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: SALDO, v. 2.3. A lexical-semantic resource for modern Swedish (∼130k entries), URL `https://spraakbanken.gu.se/eng/resource/saldo`.

Språkbanken. (2015b). *SALDO's Morphology*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: SALDO's Morphology, v. 2.3. A morphological lexicon (full-form and inflectional paradigms) for modern Swedish (∼128k entries), URL `https://spraakbanken.gu.se/eng/resource/saldom`.

Språkbanken. (2016). *The Swedish Culturomics Gigaword Corpus*. Språkbanken, University of Gothenburg, Sweden, Språkbanken corpus resources: Swedish Culturomics Gigaword Corpus, v. 1.0. A mixed-genre corpus of modern written Swedish comprising one billion words, URL `https://spraakbanken.gu.se/eng/resource/gigaword`.

Språkbanken. (2018). *SenSALDO*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: SenSALDO, v. 0.1. A word-sense prior-polarity sentiment lexicon for modern Swedish (∼7.6k entries), URL `https://spraakbanken.gu.se/eng/resource/sensaldo`, `http://hdl.handle.net/10794/sensaldo`.

WordKit: a Python Package for Orthographic and Phonological Featurization

Stéphan Tulkens, Dominiek Sandra, Walter Daelemans
CLiPS, University of Antwerp
Prinsstraat 13, 2000, Antwerpen
firstname.lastname@uantwerpen.be

Abstract

The modeling of psycholinguistic phenomena, such as word reading, with machine learning techniques requires the featurization of word stimuli into appropriate orthographic and phonological representations. Critically, the choice of features impacts the performance of machine learning algorithms, and can have important ramifications for the conclusions drawn from a model. As such, featurizing words with a variety of feature sets, without having to resort to using different tools is beneficial in terms of development cost. In this work, we present `wordkit`, a python package which allows users to switch between feature sets and featurizers with a uniform API, allowing for rapid prototyping. To the best of our knowledge, this is the first package which integrates a variety of orthographic and phonological featurizers in a single package. The package is fully compatible with `scikit-learn`, and hence can be integrated into a variety of machine learning pipelines. Furthermore, the package is modular and extensible, allowing for the future integration of a large variety of feature sets and featurizers. The package and documentation can be found at `github.com/stephantul/wordkit`

Keywords: Psycholinguistics, Phonology, Featurization

1. Introduction

Psycholinguistic models of word reading have been extensively investigated using computational models. Examples of such modeling efforts include the Interactive Activation (IA) model (McClelland and Rumelhart, 1981), the Triangle family of models (Seidenberg and McClelland, 1989; Harm and Seidenberg, 2004) the MROM model (Jacobs et al., 1998), the Dual Route Cascaded (DRC) model (Coltheart et al., 2001), and, more recently, the Devlex (Farkas and Li, 2002; Li and Farkaš, 2002) and the BLINCS models (Shook and Marian, 2013). All the implementations of these models, however different their core assumptions may be, have in common that they assume that words can be represented as numbers.

In spite of this reliance on feature-based representations, the debate surrounding these models has mostly focused on the way these models process their featurized word representations, not on the features themselves. The way words are featurized is usually treated as a factor which only deserves peripheral mention. This has ramifications for the way research progresses: first, it hampers empirical comparisons between models; while there is some consensus on *which* words are valid to use in simulations of psycholinguistic experiments, there is no consensus on how to featurize them. This can lead to situations in which models are deemed to be comparable because they use the same words as input, while they are actually not comparable because they featurize them in different ways (Plaut et al., 1996). Second, because features are not regarded as part of their respective models, there is a distinct lack of freely available featurizers, which hampers comparability and reproducibility.

In this work, we attempt to remedy these issues by presenting `wordkit`, a package which implements a variety of orthographic and phonological featurizers and corresponding feature sets. `wordkit` aims to be a one-stop, modular and extensible featurization package, which can be used for creating feature sets for psychological research, as well as input stimuli for computational models. The current version of `wordkit` contains three orthographic featurizers, five phonological featurizers, several orthographic and phonological feature sets, three corpus readers, and a sampler. These components are all realized as separate modules, which allows users to reorder or omit modules depending on their needs and resource availability. We will proceed as follows: In section 2. we will discuss some theoretical arguments for the use of different featurization techniques and how related work has used different featurization techniques, while in section 3. we will give a high-level overview of `wordkit`. In section 4., we will perform several experiments which evaluate the system by computing correlations between different featurization techniques.

2. Theoretical Discussion

The use of featurizers such as those included in `wordkit` touches upon at least two theoretical issues: the structure of the orthographic code, and the influence of phonology in word reading.

2.1. The orthographic code

In general, reading speeds are modulated to a large degree by the way words orthographically resemble one another; words in dense neighborhoods are, on average, read faster than words with the same length but fewer neighbors (Carreiras et al., 1997; Perea and Pollatsek, 1998; Pollatsek et al., 1999), and non-words with more neighbors are rejected more slowly than non-words with fewer neighbors (Arduino and Burani, 2004). This notion of neighborhood can be defined in several ways, but it is often defined using Levenshtein-based measures, such as Coltheart's N (Coltheart et al., 1977) or OLD20 (Yarkoni et al., 2008).

Regardless of how one chooses to define neighborhoods, the base assumption seems to be that words similar to the word being activated are co-activated because they have similar orthographic codes (Grainger, 2008). In this view, orthographic featurization techniques can be viewed as implementations of orthographic codes. While there is little agreement

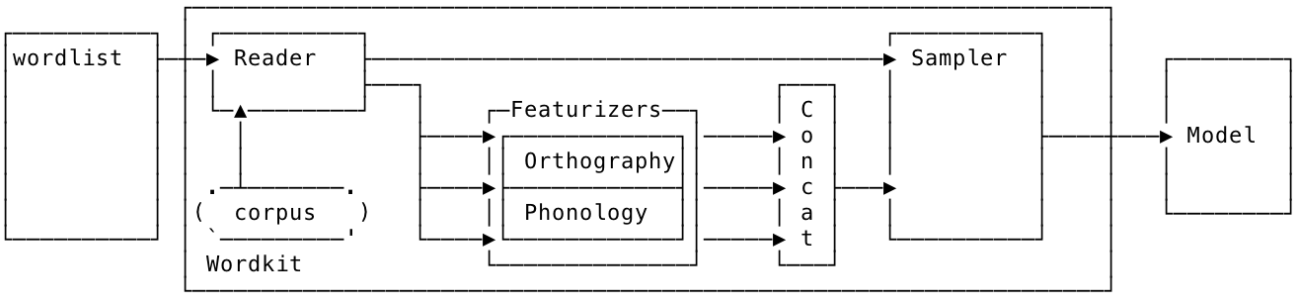


Figure 1: A sample wordkit pipeline

on which featurization technique is valid (Grainger, 2008), most models, including the Interactive Activation family of models, the DRC model and the Triangle family of models assume that the orthographic code consists of a sequence of slots, which can be filled with letters. This is striking, as a purely sequential coding of orthography is unlikely to be correct, as pointed out by Davis (Davis, 2012); readers tend to judge words with transposed or added letters as highly similar, e.g. 'garden'-'graden' and 'men'-'amen', even though the vector representations of these words are quite different when featurized using a sequential encoding.

An alternative to sequential coding is a so-called open bigram encoding (Grainger and Van Heuven, 2004; Schoonbaert and Grainger, 2004; Grainger and Whitney, 2004), which is formed by taking the ordered 2-combination of each letter in a word. That is, the open bigram encodings of the words 'salt' and 'slat' are $\{sa, sl, st, al, at, lt\}$ and $\{sl, sa, st, la, lt, at\}$, respectively. It is worth noting that the open bigram encoding has also been deemed unsatisfactory; as shown by Davis and Bowers, the open bigram coding has no way of representing whether the constituent letters of the bigrams are contiguous (Davis and Bowers, 2006). Taking the 'salt'-'slat' example above, we can see that both of the bigram sets contain the bigram at, but it is only contiguous in the case of "slat". Furthermore, Davis and Bowers have shown that participants judge substitution neighbors ('stop'-'shop') to be more similar than transposition neighbors, which is exactly the opposite of what open bigram encoding predicts (Davis and Bowers, 2006).

Another alternative to sequential coding is the spatial coding offered by the SOLAR (Davis, 2001), and the unconstrained open bigrams used by the SERIOL (Whitney, 2001) models. These featurization techniques, which offer a better fit to empirical data, are not implemented in `wordkit` because, rather than being featurizers, SERIOL and SOLAR are actually full-blown models of letter recognition. That is, SERIOL and SOLAR do not produce the type of vectors which can be used as input by other models, and are more like the models that accept the input vectors which are generated by our transformers.

In the same vein, it is worth noting that orthographic codes have also been obtained by training feed-forward supervised neural networks (Dandurand et al., 2010; Hannagan et al., 2011). While interesting from a theoretical point of view, the hidden states of such a model do not constitute the kind of representations suitable for input in most models, and, like SERIOL and SOLAR, producing these representations requires a representation of the orthographic code.

Even though strictly sequential coding and open bigram coding are unlikely to explain all observed phenomena in orthographic similarity judgments, we still include them in `wordkit`. The goal of this package is, after all, the facilitation of comparisons between different featurization techniques for replication and research purposes. A similar argument holds for the included features, which are described below. It is quite unlikely that letters are best represented by a single scalar, as in the DISLEX model of Miikkulainen (Miikkulainen, 1997), but we still include them because it facilitates reproduction and comparison between models.

2.2. The influence of phonology

While the debate surrounding the orthographic code has mostly centered on *which* information readers extract from visual stimuli, the debate in phonology has centered on the absence or presence of phonological influence during lexical access. One the one hand, this is not surprising; it is undebatable that orthographic information plays a role in reading, but phonological information is one step removed from printed words, and therefore the question of whether it influences reading is less straightforward. Furthermore, it has been extremely difficult to determine whether phonology plays a role in lexical access, as opposed to post-access; even if reliable priming effects from words to phonological neighbors are obtained, opponents of a phonological theory of word reading can still claim that these effects occur post-access, and do not influence the actual access of lexical items.

As a consequence, little attention has been given to the exact featurization of phonology in word reading. Most models that provide phonological information assume that graphemes are first converted to phonemes, after which the resulting phonemic representations co-activate similar-sounding representations (Seidenberg and McClelland, 1989; Miikkulainen, 1997; Harm and Seidenberg, 2004).

There exist two main strands in phonological featurization; one based on Onset Nucleus Coda (ONC) featurization (Keuleers and Daelemans, 2007), and the other on Consonant Vowel (CV) featurization (Li and MacWhinney, 2002). These methods differ only in the way they order phonemes, not in the way they represent individual phonemes. ONC featurizers group phonemes in subdivisions of syllables, where each syllable is sub-divided in an Onset, a Nucleus, and a

	Phonology	Syllables	Frequency
Celex	✓	✓	✓
CMU	✓		
Deri	✓		

Table 1: The corpus readers and their descriptors

Orthography	Phonology	Frequency
wind	waɪnd	298
wind	wɪnd	2170

Table 2: The structured output of 'wind' from the Celex corpus reader

Coda, while CV featurization techniques group phonemes according to a grid with alternating Consonant and Vowel slots. Note that, while these featurization techniques solve the problem of aligning phonological strings, they do not solve the problem of aligning syllabic units. That is, while phonemes within syllables are reasonably aligned when using both the CV and ONC featurizer, there is no guarantee that syllables themselves are aligned. In that sense, the problem of alignment is only partially solved by aligning phonemes to a grid Both CV and ONC transformers have been implemented in `wordkit`, and may provide a testbed for theories of phonologically driven theories of word reading.

3. Overview of `Wordkit`

`wordkit` is a modular system, consisting of discrete components, which can be combined to create an end-to-end featurization pipeline. Additionally, all these components can be used separately, or integrated into other systems. To facilitate the incorporation of modules into other machine learning experiments, all `wordkit` modules inherit `scikit-learn` (Pedregosa et al., 2011) base classes, which allows modules to be directly integrated into `scikit-learn` pipelines. Currently, there are three types of modules:

1. **Readers:** modules which retrieve words from a corpus, and output structured information about that word in that corpus.

2. **Transformers:** modules which take in structured information and turn them into feature vectors.

3. **Samplers:** modules which take a set transformed words as input and sample from this set according to their frequency.

3.1. Readers

Readers are modules which take as input strings representing the orthographic form of words, e.g. 'wood' or 'dog', and extract structured information corresponding to this string from a corpus. The information extracted from a corpus depends on the information present in that corpus: every corpus reader defines a set of so-called descriptors, which denote which data fields are contained within that corpus. We currently offer readers for three corpora: Celex (Baayen et al., 1993), a large multilingual collection of corpora (Deri and Knight, 2016), and CMUdict. The current corpus readers and their fields are listed in Table 1. If the phonological information in the corpus does not correspond to the IPA (International Phonetic Alphabet), as in the case of Celex and CMUdict corpora, we convert the representation to IPA, which serves to make these corpora comparable. As an example of use, consider the word 'wind', which is a phonologically ambiguous homograph. Table 2 shows the structured records we get as output when we present 'wind' as input to the Celex corpus reader. Corpus readers can also be combined in parallel, which allows one to, for example, perform multilingual experiments by including both the Dutch and English Celex corpora, or to combine frequency estimates from one corpus with phonological forms from another. Finally, corpus readers accept an arbitrary filtering function which filters the output given by the reader. This, for example, allows users to specify constraints on the number of syllables, frequency, word length, among others. These features make `wordkit` applicable to the creation of stimulus sets for research in experimental psychology, in which a diverse set of constraints need to be satisfied.

3.2. Transformers

In `wordkit`, a transformer is a module which turns structured information into feature vectors which can be fed into a machine learning system or otherwise used in experiments. Throughout this paper, we denote single modules as transformers, and combinations of different transformers as a featurizer. Like the corpus readers, the transformer modules can be combined in parallel, allowing for the featurization of a word into multiple feature vectors, which are then concatenated into a single vector. We will describe all transformers in order.

Linear The Lineartransformer is directly based on the featurization scheme in the Interactive Activation (IA) model (McClelland and Rumelhart, 1981), which is a slot-based featurization scheme. This implies that the orthographic code is defined as consisting of a number of slots, which are filled with feature values. In our case, the Lineartransformer represents words as concatenations of feature vectors, the values of which are determined by the feature set in use. Like other slot-based transformers, the lineartransformer makes the naive assumption that letter transpositions do not impact word recognition, and that words are strictly left-justified. This means, for example, that "stop" and "top" do not share any letters according to this encoding scheme. Nevertheless, a substantial amount of reading models, like the aforementioned IA models, assume a slot-based encoding, and its inclusion in `wordkit` is therefore warranted. The Lineartransformer can also be applied to phonological feature strings, with the caveat that this requires vowels and consonants to have the same number of features, and that vowels and consonants do not overlap in features. This is a difficult assumption to maintain, and therefore the Lineartransformer is usually not applicable to phonological feature strings.

Wickel We also include the family of Wickeltransformers, which represent words as unordered collections of *n*grams (Rumelhart and McClelland, 1985), also commonly called

	Dim
One hot	# of unique characters
Fourteen segment	14
Sixteen segment	16
Miikkulainen	1

Table 3: The available orthographic feature sets

wickelphones when used for phonology. While wickelphones have traditionally been extracted using trigrams, we extend the wickeltransformer to accept arbitrary values of *n*, although increasing *n* results in extremely large feature spaces. Wickeltransformers can be used for both phonological and orthographic strings, and do not require features to function, which makes them widely applicable. Wickeltransformers can represent order information, while still being able to flexibly account for some substitution effects. As pointed out by Rumelhart and McClelland, the Wickelphone representation can not represent one-letter substitutions correctly; the words "silt" and "slit" have 0 overlapping features (McClelland et al., 1987).
Note that using Wickelfeatures has been shown to have detrimental effects on performance in the SM89 Triangle model (Plaut et al., 1996).

Open $n$grams The open bigram coding is a featurization procedure which takes into account transposition effects by using ordered bigrams, which is equivalent to taking the 2-combination of all letters in a word (Grainger and Van Heuven, 2004; Schoonbaert and Grainger, 2004). Like in the case of the Wickeltransformer above, we extend the open bigram coding to an open $n$gram coding, in which the user can specify n herself. The Open$n$gramtransformer is also highly similar to the Wickeltransformer in other respects: it doesn't require any features, results in an unordered bag of features, and works for both arbitrary phonological and orthographic strings.

Consonant Vowel The Consonant Vowel (CV) transformer is a generalization of `patpho` (Li and MacWhinney, 2002), which represents phonology by assigning phonemes to a consonant-vowel grid, and then replacing them with binary or real-valued feature vectors. Transforming vectors using a CV grid generally applies to phonology, unless one assumes that orthographic characters are also segmented into consonants and vowels, an assumption which is rarely made[1]. The CVtransformer generalizes `patpho` because it supports variable-length grids and the use of different feature sets, while giving the same results as `patpho` when given the same grid and features. Because CV assigns words to a phoneme grid, words which do not have fully aligned orthographic forms might still be reasonably aligned in their phonological form. For example, the words 'pot' and 'spot', which share no characters in any position, are still phonologically aligned using a CVtransformer. Unlike the original version of `patpho`, our transformer can automatically optimize its grid size for a given dataset.

Onset Nucleus Coda Like the CVtransformer, the Onset Nucleus Coda (ONC) transformer creates vector representations by aligning phonemes on a grid, and replacing these by feature vectors. The main difference here is that the ONCtransformer relies on syllable information to segment phonemes into sub-syllabic Onset, Nucleus and Coda clusters. As such, this information needs to be present in the corpus if one wants to attempt featurization using an ONCtransformer; we do not provide a built-in syllabifier. Adhering to syllable structure predictably leads to a more rigid, but also a larger representation, in the sense that vectors created with the ONC transformer tend to be larger in dimensionality. Like the CV transformer, the feature grid of the ONC transformer can be automatically adapted based on the data, which means that the number of syllables, as well as the number of Onset, Nucleus and Coda clusters per syllable will be automatically determined.

[1] but see (Shook and Marian, 2013)

	Vowel Dim	Consonant Dim
patpho binary	5	7
patpho real	3	3
Plunkett	6	6
One hot phoneme	# of vowels	# of consonants
One hot feature	39	76
Miikkulainen	2	3
Binary feature	5	8

Table 4: The available phonological feature sets. Note that the One hot features and Binary features can have variable sizes, depending on the features the user decides to extract from IPA phonemes.

3.3. Feature sets

Table 3 summarizes the orthographic feature sets, i.e. the character vectors. The 14 segment display was used in the Interactive Activation model(McClelland and Rumelhart, 1981), and represents words as sets of lines, as in a rudimentary microwave display. The 16 segment display is a simple expansion of the 14 segment feature set that can also represent a wide variety of punctuation marks. The Miikkulainen feature set represents each letter by a single scalar: the proportion of black and white pixels for an uppercase version of each letter (Miikkulainen, 1997; Shook and Marian, 2013).
Table 4 details the phonological feature sets. All phonological feature sets distinguish between consonants and vowels, because the features for both types of phonemes are different. As such, allowing overlap between consonants and vowels would lead to incorrect results. The patpho features are from (Li and MacWhinney, 2002), the Miikkulainen features are from (Miikkulainen, 1997), the Plunkett features are from (Plunkett and Marchman, 1993), while the binary feature set is a modified version of the patpho feature set.
In addition, we offer the option of one-hot encoded feature sets for both phonology and orthography. The usage of these feature vectors implies that characters and phonemes are categorical, as their vectors are completely orthogonal.

3.4. Samplers

Samplers are modules which take as input structured information from a corpus which includes a frequency descriptor, as well as the featurized versions of these words. The sam-

Orth Phon	Wickel Wickel	Linear (fourteen) CV (binary)	Linear (sixteen) CV (binary)	Linear (one hot) NGram	NGram NGram	Linear (one hot) CV (binary)
1	logarithm	logarithm	logarithm	logarithm	logarithm	alpinism
2	rhythm	vulgarism	alpinism	allegory	allegory	aphorism
3	alga	aphorism	algerian	algeria	alligator	logarithm
4	allegory	dogmatism	aphorism	aphorism	vulgarism	vulgarism
5	gorilla	despotism	alligator	alga	glamorize	mannerism
6	amalgam	magnetism	algeria	algerian	altruism	vandalism
7	alum	anglicism	ultimatum	gallery	grim	embolism
8	al	allopathy	generator	calorie	aphorism	alga
9	algeria	mongolism	decorator	alacrity	allegoric	anarchism
10	allegoric	mesmerism	allopathy	vulgarism	lariat	galvanism

Table 5: The 10 nearest neighbors to the word "Algorithm" for a variety of featurizers. The names in braces, if any, are the feature names used by the featurizer. The feature sets and featurizers were chosen to be as diverse as possible.

pler then samples from the featurized dataset according to the frequency of these words. A sampler allows researchers to create ecologically valid datasets, insofar the frequencies in the sampled corpora are indicative of the actual frequencies with which words occur to actual language users. `wordkit` includes samplers which operate on raw frequencies, one-smoothed raw frequencies, or log-transformed frequencies.

4. Experiments

In this section we will perform two correlation experiments on a large set of featurizers. The goal of these experiments is twofold: First, we aim to show that iterating over a large set of features is easy using the provided tools in `wordkit`. Second, we aim to show that the features themselves can already account for a significant amount of variation in word reading times. The first experiment looks for a correlation between feature sets on a large (N = 17682) set of words from Celex. The second experiment uses a slightly smaller set of words (N = 10487) and correlates the average distance to differences in lexical decision times from the British Lexicon Project (BLP) (Keuleers et al., 2012). In both experiments we limit ourselves to a description of the observed differences, and leave a full-blown statistical analysis for future work.

4.1. Experiment 1

To show how the choice of features impacts the outcome of experiments, thereby demonstrating the utility of `wordkit`, we perform a correlation analysis of the distances between featurized words over a large set of featurizers. If the correlation between two feature sets is high, the choice between these feature sets makes little difference in the performance of the model. Hence, if two models use highly correlated feature sets and still show different results, the architecture, and not the features, is likely responsible for the observed differences in performance between said models. If the models use feature sets that show low correlation, any observed difference in performance using these feature sets can not be solely ascribed to architectural differences between the models.

We proceeded by first creating a set of orthographic featurizers by selecting all pairwise combinations between orthographic transformers and feature sets, which resulted in a set of six possible orthographic featurizers. Following this, we extracted all pairwise combinations between phonological featurizers and feature sets, excluding all feature sets which did not distinguish between long and short vowels (a selection of four feature sets), which led to a set of 10 possible phonological featurizers. We then defined our set of featurizers as all pairwise combinations between the set of orthographic and phonological featurizers, leading to a set of 60 total featurizers (six possible orthographic featurizers, and 10 possible phonological featurizers). Note that we did not vary any of the parameters in the featurizers, for example the values of n in the Wickeltransformer and Open$n$gramtransformer, which would have drastically increased the number of possible combinations.

We then proceeded by extracting all words whose phonological forms consisted of fewer than 12 phonemes and which were shorter than 10 characters from the English part of Celex. We also removed words if any featurizer was not able to featurize them appropriately, e.g. due to missing phonemes, special characters. This resulted in a set of 17682 words.

For each of the featurizers we calculated the cosine similarity from each word in our selected set of words to each other word, leading to a 17682×17862 square matrix for each featurizer. We then calculated the pairwise correlation between every distance matrix, leading to a set of 1770 matrix comparisons ((60 * 59) / 2). Figure 2 shows the pairwise correlations between each of the matrices. Note that the X-axis lists the names of the featurizers and feature combinations (CV, ONC, W for Wickel, and O for the Open$n$gramFeaturizer). For lack of space in the figure, we do not list the phonological feature sets, of which there were 4. These were, in order, one hot encoded phonemes, one hot encoded features, the features from Miikkulainen (Miikkulainen, 1997), and the modified set of patpho features (Li and MacWhinney, 2002).

The correlations between the distance matrices show that there exist differences between the featurization techniques: First off, the sub-orthographic features, i.e. the fourteen- and sixteen segment encodings (McClelland and Rumelhart, 1981) correlate heavily with models with the same sub-orthographic features, and erase any effect of the phonological featurizer. Also of interest is the fact that models encoded with fourteen- and sixteem segment encodings re-

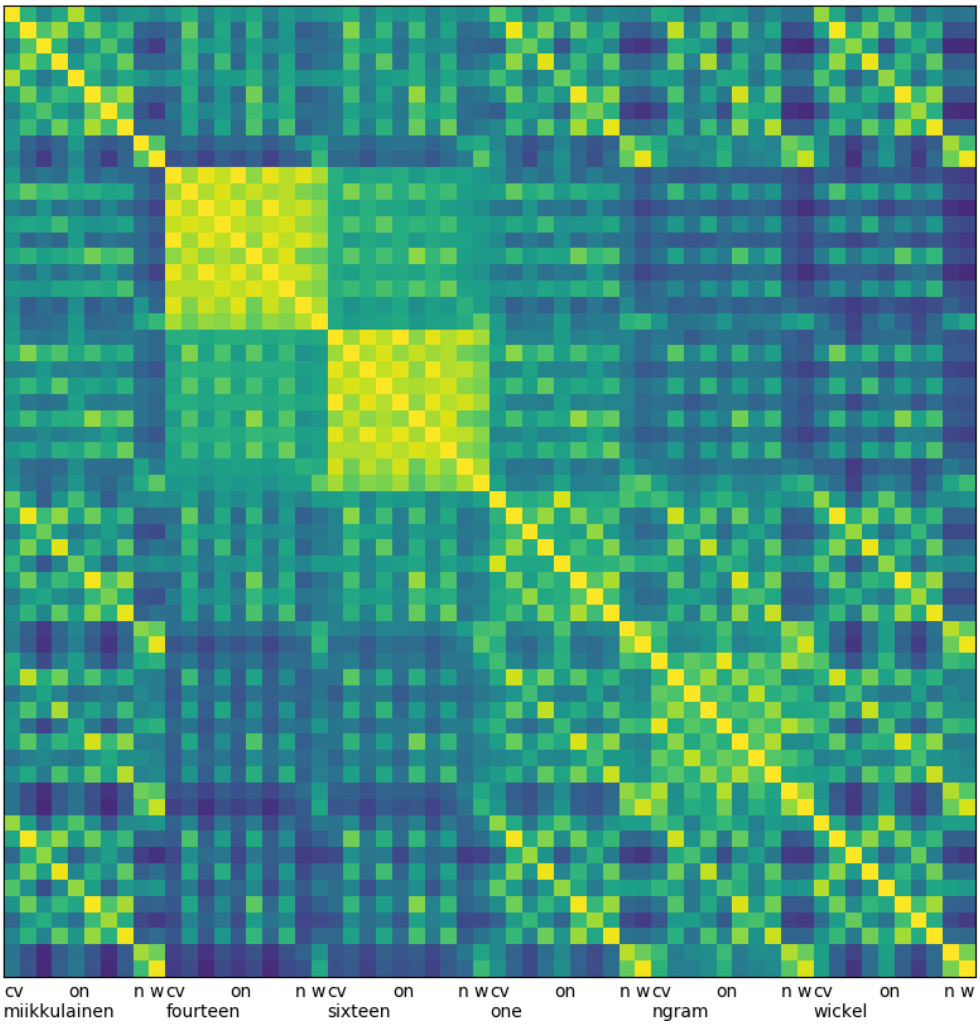


Figure 2: The correlations between featurization pipelines on a set of 17682 words, where lighter colors imply higher correlation. The phonological feature sets are not shown.

semble each other more than they do other models. This again points to a kind of primacy of sub-orthography; models with sub-orthographic features are inherently different from other models, regardless of their feature set.

A second effect is concerns phonological encodings involving ngrams, i.e. the Wickeltransformer or the Open$n$gramtransformer. Recall that these transformers do not involve any pre-defined feature sets, but instead encode stimuli using a bag of ngrams. As we can see from the figure, the correlations between featurizers which encode phonology using a Wickeltransformer or Open$n$gramtransformer are higher than the correlations between the other transformers. This again leads to a dominating effect; the effect of any orthographic encoding on the distance matrices is effectively removed, and subsumed by the effect of the phonological featurizer. In one sense this is not surprising, as these two transformers, unlike the other transformers, do not use a slot-based encoding. Looking at the effect the Open$n$gramtransformer and Wickeltransformer have on orthography, we see a less pronounced difference, although we see a larger set of correlations for the Open$n$gramtransformer than the Wickeltransformer.

Concluding this experiment, we see that both the choice of features as well as the choice of featurization techniques, greatly impacts the similarities between words, and hence the information offered to the model. More importantly, we observe that choosing a certain feature set can cause orthography or phonological similarity to dominate over other similarities. This has important implications for models of word reading; if the modeler chooses a certain feature set without validating said feature set, the modeler risks making conclusions about the relative importance of, for example, phonology, without knowing whether the features or the model architecture contributed to the observed difference.

Table 5 shows the 10 nearest neighbors to the word ”algorithm” for a variety of featurizers. Here we see that using different features and transformer combinations leads to a variety of different nearest neighbors.

Using a Consonant Vowel phonological grid in conjunction with the slot-based encoding of the linear transformer leads to a bias towards words with similar lengths. In contrast, the Wickel and NGram transformers have a lower bias towards length, and select a variety of as nearest neighbors.

Comparing the first, second, and final columns gives an idea of the influence of features, as the featurizers in each of these columns use the same features, and only differ in

the selection of orthographic features. Here we see that the selection of features leads to a difference in the extracted words. This corroborates the evidence from Figure 2, in which we saw that both features and transformers influence the correlations between feature sets.

4.2. Experiment 2

In experiment 2 we used a set of 50 featurizers. We used the same featurizers as in the experiment above, except that chose to leave out the miikkulainen features on account of these not being binary features. In this experiment, we correlated the overlap distance from each word to each of its 20 closest neighbors with the Reaction Times (RT) of the BLP items (Keuleers et al., 2012). Previous work has shown that the mean Levenhstein distance to the 20 closest neighbors (OLD20) is a good predictor of reading time, as measured on multiple corpora (Yarkoni et al., 2008), and outperforms a neighborhood measure in which the orthographic neighborhood of a word is defined as the number of words within a Levenshtein distance of 1 (Coltheart et al., 1977). In this experiment, we provide a rough analogue to the OLD20 neighborhood metric, but defined on arbitrary feature spaces.

We used the same set of words as in the previous experiment, but we removed any words that do not occur in the set of BLP items, i.e. words for which we do not have reaction time measurements. This lead to a set of 10487 words. As in the previous experiment, we calculate the pairwise hamming distance between each word and each other word, leading to a $N \times N$ matrix for each featurizer.

We replaced the cosine distance in the previous experiment because the hamming distance is a straightforward analogue of perceptual distance for binary vectors. The cosine, being normalized by vector length, assigns higher weights to individual components for vectors with fewer components. This is useful when measuring the correlation between distance measures, as the magnitude of the differences between the different distance matrices no longer influences the perceived correlation.

As in OLD20 (Yarkoni et al., 2008), we define the neighborhood of a word as the mean distance to its 20 closest neighbors. Like OLD20, this neighborhood metric can then be compared to the RT measurements from the BLP. We straightforwardly chose to use the 20 closest neighbors to conform to the value reported by Yarkoni et al., although we stress that there is no intrinsic reason for choosing this value (Yarkoni et al., 2008). Additionally, Yarkoni et al. report that the choosing 20 neighbors has little effect on the explained variance of the OLD20 measure. Although we did not test this assumption, we assume that the same holds for our featurizers, and leave the investigation of this assumption for future work.

For each of the featurizers, we calculated the Pearson correlation between each word and the RT from the BLP. To get more robust statistical estimates, we bootstrapped 100 samples of 9000 words from our base set of 10487 words. Figure 3 shows the correlation and confidence intervals for each of the transformers with respect to the RTs from the BLP. For each word in each sample we also calculated the OLD20, thus also obtaining robust statistical estimates for

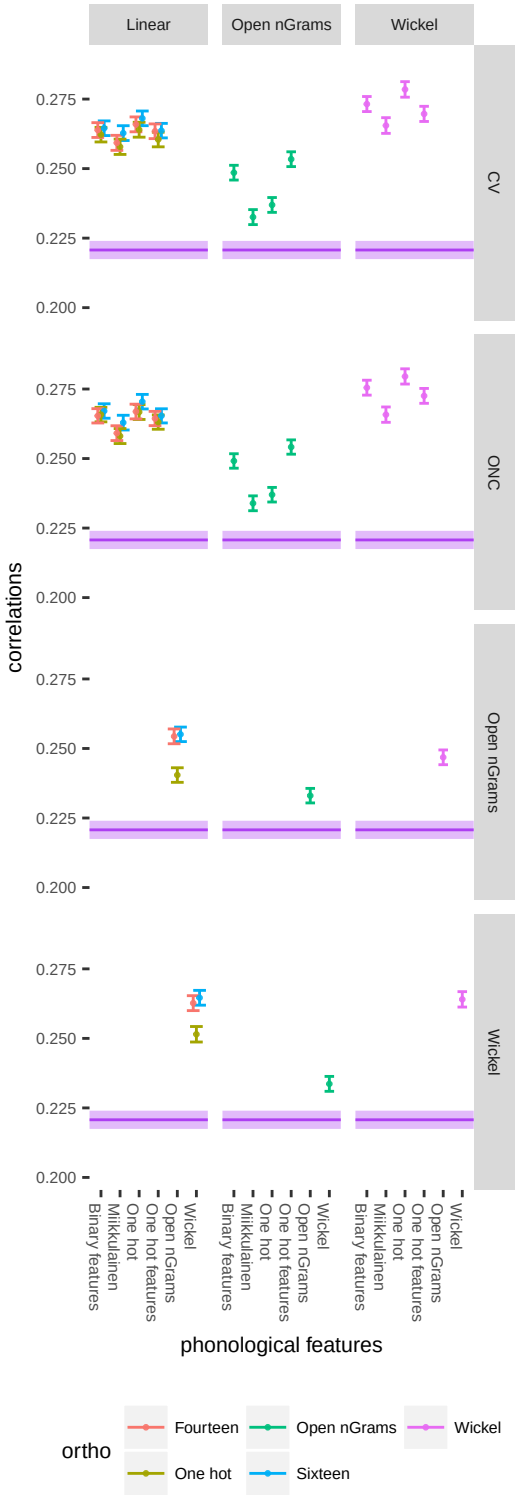


Figure 3: The mean correlations with 99.9% confidence intervals between the different transformers and the RT values from the BLP, bootstrapped from 100 samples. Each panel lists a unique Orthography $\times$ Phonological transformer combination. The columns *within* each figure specify the phonological feature sets, while the colors specify the orthographic feature sets. The purple bar at the bottom indicates the mean and confidence interval of OLD20, also bootstrapped from 100 samples.

comparing OLD20 to the other featurizers.[2]

The figure shows that all featurizers outperform OLD20. Additionally, we see that switching between the ONC and CV transformers has almost no effect on performance. It seems that simply aligning phonemes on a grid is an important factor; but that the type of grid apparently changes little in performance. Furthermore, we see that using Open nGrams as either the phonological or orthographic transformer has a severely detrimental effect on performance. Out of all transformers, OLD20 excluded, all featurizers with the lowest performance included open nGram transformers. Finally, we see that using a Wickeltransformer in the orthographic position increases the fit, and leads to the best fit overall; using it in the phonological position does not seem to increase performance. This further confirms empirical research by Davis and Bowers (Davis and Bowers, 2006): as said above, subjects tend to judge transposition neighbors to be less similar than substitution neighbors. Open bigrams, being unordered, have no way of representing whether two letters are contiguous. Similarly, the ngram encoding of the Wickeltransformer represents both order information as well as some information about some substitutions, which might explain their increased performance in this task.

One thing we did not account for in the current experiment is the presence or absence of phonological information; OLD20 is calculated only on the basis of orthographic information, while the other featurizers use a combination of orthographic and phonological information.

While this may have biased the experiment towards the featurizers, calculating OLD20 on phonological strings directly makes less sense, since phonological strings are clearly ordered in syllable clusters. Put in another way: a slot-based encoding, while unsatisfactory for orthography, is highly desirable in the case of phonology. Note that there also exists a PLD20, which is the OLD20 analogue for phonology. This measure does not work directly on phonological strings, and instead requires a metaphonetic representation.

5. Conclusion

In this work we presented `wordkit`, a versatile tool for the featurization and retrieval of word stimulus sets from a variety of corpora. We have demonstrated the utility of the toolkit by doing a large-scale analysis of different featurization pipelines, which showed that the choice of featurization pipeline leads to different nearest neighbors. As such, changing features in a model most likely has a big effect on the performance of the model. Hence, we argue that features should be seen as a part of any psycholinguistic model and the accompanying theoretical framework, and not theory-agnostic. Additionally, we performed an experiment in which we compare the performance of 60 featurizers to the well known OLD20 measure in correlating with RT judgments in a lexical decision task. This showed that there is considerable difference in performance between different pipelines, again showing that the choice of featurization is an important consideration. As for future work; we aim to expand the number of transformers and corpora in `wordkit`. We also aim to present a more thorough evaluation of the effect of orthography and phonology, and a more thorough comparison to both OLD20 and PLD20.

[2] we wrote an open-source package to estimate OLD20: `www.github.com/stephantul/old20`

6. Acknowledgments

The first author is supported by a PhD scholarship from the FWO Research Foundation - Flanders. We would like to thank Giovanni Cassani and Robert Grimm for help with the visualization and helpful discussion. Robert Grimm wrote a first version of the Python port of `patpho`, which we generalized to the CVtransformer included in the package.

7. Bibliographical References

Arduino, L. S. and Burani, C. (2004). Neighborhood effects on nonword visual processing in a language with shallow orthography. *Journal of Psycholinguistic Research*, 33(1):75–95.

Baayen, R. H., Piepenbrock, R., and van H, R. (1993). The CELEX lexical data base on CD-ROM.

Carreiras, M., Perea, M., and Grainger, J. (1997). Effects of the orthographic neighborhood in visual word recognition: Cross-task comparisons. *Journal of experimental psychology: learning, memory, and cognition*, 23(4):857.

Coltheart, M., Davelaar, E., Jonasson, T., and Besner, D. (1977). Access to the internal lexicon.

Coltheart, M., Rastle, K., Perry, C., Langdon, R., and Ziegler, J. (2001). Drc: a dual route cascaded model of visual word recognition and reading aloud. *Psychological review*, 108(1):204.

Dandurand, F., Grainger, J., and Dufau, S. (2010). Learning location-invariant orthographic representations for printed words. *Connection Science*, 22(1):25–42.

Davis, C. J. and Bowers, J. S. (2006). Contrasting five different theories of letter position coding: Evidence from orthographic similarity effects. *Journal of Experimental Psychology: Human Perception and Performance*, 32(3):535.

Davis, C. J. (2001). *The self-organising lexical acquisition and recognition (SOLAR) model of visual word recognition.* Ph.D. thesis.

Davis, C. J. (2012). The orthographic similarity of printed words. *Visual Word Recognition: Models and Methods, Orthography and Phonology*, 1:120–206.

Deri, A. and Knight, K. (2016). Grapheme-to-phoneme models for (almost) any language. In *ACL (1)*.

Farkas, I. and Li, P. (2002). Devlex: A self-organizing neural network model of the development of lexicon. In *Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on*, volume 5, pages 2546–2551. IEEE.

Grainger, J. and Van Heuven, W. J. (2004). Modeling letter position coding in printed word perception.

Grainger, J. and Whitney, C. (2004). Does the huamn mnid raed wrods as a wlohe? *Trends in cognitive sciences*, 8(2):58–59.

Grainger, J. (2008). Cracking the orthographic code: An introduction. *Language and Cognitive Processes*, 23(1):1–35.

Hannagan, T., Dandurand, F., and Grainger, J. (2011). Broken symmetries in a location-invariant word recognition network. *Neural Computation*, 23(1):251–283.

Harm, M. W. and Seidenberg, M. S. (2004). Computing the meanings of words in reading: cooperative division of labor between visual and phonological processes. *Psychological review*, 111(3):662.

Jacobs, A. M., Rey, A., Ziegler, J. C., and Grainger, J. (1998). Mrom-p: An interactive activation, multiple readout model of orthographic and phonological processes in visual word recognition.

Keuleers, E. and Daelemans, W. (2007). Memory-based learning models of inflectional morphology: A methodological case-study. *Lingue e linguaggio*, 6(2):151–174.

Keuleers, E., Lacey, P., Rastle, K., and Brysbaert, M. (2012). The british lexicon project: Lexical decision data for 28,730 monosyllabic and disyllabic english words. *Behavior research methods*, 44(1):287–304.

Li, P. and Farkaš, I. (2002). Modeling the development of lexicon with devlex: A self-organizing neural network model of lexical acquisition. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 24.

Li, P. and MacWhinney, B. (2002). Patpho: A phonological pattern generator for neural networks. *Behavior Research Methods, Instruments, & Computers*, 34(3):408–415.

McClelland, J. L. and Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychological review*, 88(5):375.

McClelland, J. L., Rumelhart, D. E., Group, P. R., et al. (1987). *Parallel distributed processing*, volume 2. MIT press Cambridge, MA:.

Miikkulainen, R. (1997). Dyslexic and category-specific aphasic impairments in a self-organizing feature map model of the lexicon. *Brain and language*, 59(2):334–366.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Perea, M. and Pollatsek, A. (1998). The effects of neighborhood frequency in reading and lexical decision. *Journal of Experimental Psychology: Human Perception and Performance*, 24(3):767.

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., and Patterson, K. (1996). Understanding normal and impaired word reading: computational principles in quasi-regular domains. *Psychological review*, 103(1):56.

Plunkett, K. and Marchman, V. (1993). From rote learning to system building: Acquiring verb morphology in children and connectionist nets. *Cognition*, 48(1):21–69.

Pollatsek, A., Perea, M., and Binder, K. S. (1999). The effects of" neighborhood size" in reading and lexical decision. *Journal of Experimental Psychology: Human Perception and Performance*, 25(4):1142.

Rumelhart, D. E. and McClelland, J. L. (1985). On learning the past tenses of english verbs. Technical report, la jolla institute for cognitive science.

Schoonbaert, S. and Grainger, J. (2004). Letter position coding in printed word perception: Effects of repeated and transposed letters. *Language and Cognitive Processes*, 19(3):333–367.

Seidenberg, M. S. and McClelland, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological review*, 96(4):523.

Shook, A. and Marian, V. (2013). The bilingual language interaction network for comprehension of speech. *Bilingualism: Language and Cognition*, 16(2):304–324.

Whitney, C. (2001). How the brain encodes the order of letters in a printed word: The seriol model and selective literature review. *Psychonomic Bulletin & Review*, 8(2):221–243.

Yarkoni, T., Balota, D., and Yap, M. (2008). Moving beyond coltheart?s n: A new measure of orthographic similarity. *Psychonomic Bulletin & Review*, 15(5):971–979.

Pronunciation Variants and ASR of Colloquial Speech: A Case Study on Czech

David Lukeš, Marie Kopřivová, Zuzana Komrsková, Petra Poukarová
Institute of the Czech National Corpus, Charles University, Faculty of Arts, Czech Republic
{david.lukes, marie.koprivova, zuzana.komrskova, petra.poukarova}@ff.cuni.cz

Abstract

A standard ASR system is built using three types of mutually related language resources: apart from speech recordings and orthographic transcripts, a pronunciation component maps tokens in the transcripts to their phonetic representations. Its implementation is either lexicon-based (whether by way of simple lookup or of a stochastic grapheme-to-phoneme converter trained on the source lexicon) or rule-based, or a hybrid thereof. Whichever approach ends up being taken (as determined primarily by the writing system of the language in question), little attention is usually paid to pronunciation variants stemming from connected speech processes, hypoarticulation, and other phenomena typical for colloquial speech, mostly because the resource is seldom directly empirically derived. This paper presents a case study on the automatic recognition of colloquial Czech, using a pronunciation dictionary extracted from the ORTOFON corpus of informal spontaneous Czech, which is manually phonetically transcribed. The performance of the dictionary is compared to a standard rule-based pronunciation component, as evaluated against a subset of the ORTOFON corpus (multiple speakers recorded on a single compact device) and the Vystadial telephone speech corpus, for which prior benchmarks are available.

Keywords: speech recognition, pronunciation variants, spontaneous speech

1. Introduction

One of the components of an automatic speech recognition (ASR) system is a pronunciation dictionary, which provides a mapping between a conventional symbolic transcript of speech, which can exhibit varying degrees of arbitrariness, and an acoustically / phonetically motivated one. On the one hand, during the training phase (phone-level alignment, creation of acoustic models, henceforth AMs), the phonetic transcription provides information about where to expect recurring patterns in the acoustic signal, which will be abstracted away and generalized in the acoustic models for the individual (tri)phones. On the other hand, during decoding, it mediates between (to borrow a neurolinguistic analogy) the top-down (predictive) processes based on the linguistic knowledge (the language models, henceforth LMs), and the bottom-up (recognitive) processes based on spotting acoustic patterns in the incoming sound data.

The complexity of the mapping between the standardized orthographic representation of a word and an encoding of its pronunciation as a string of phones depends on the characteristics of the writing system of the language in question. The possibilities range from more or less complete arbitrariness to varying degrees of systematicity in the relationship between graphemes and phonemes (see Sampson (2015) for a book-length account). The variety present in this respect in the world's languages is best attested by sketching a few illustrative points along this continuum:

- Chinese logographic writing bears little to no relation to the phonetic form of words; phonetically speaking, two Mandarin Chinese words such as *mother* and *hemp* may have the same segmental content ([ma]) and differ "only" in their suprasegmental features (1st vs. 2nd tone), yet their traditional written forms will betray none of their acoustic similarity (媽 vs. 麻);

- in general, alphabetic writing systems exhibit higher degrees of consistency in the relationship between graphemes and phonemes, because they are historically built on modeling regularities in speech sounds, but variation is still possible: for instance, contemporary English orthography, though it has evolved over the centuries, still bears the marks of earlier pronunciations and dialectal variants which have been weeded out by sound change in speech but became fossilized in writing, as well as numerous inconsistencies introduced by lexical loans borrowed from other languages (and thus other writing systems);

- conversely, the orthography of a language like Czech is fairly predictable based on pronunciation and vice versa, even increasingly so over the centuries thanks to several spelling reforms (see Kučera (1998, Fig. 5, p. 195), and the whole article for a treatment of the evolution of the efficiency and complexity of Czech spelling), although loanwords with preserved original spelling, mostly coming from English, are a recent disruptive factor.

In the case of Chinese, where the grapheme-to-phoneme correspondence is arbitrary, the only way of creating a pronunciation dictionary is compiling it manually. For English, where the correspondence is obscured by layers of historical development, it is the only *practical* solution (popular resources are the CMU Pronouncing Dictionary[1] for American English and the BEEP Dictionary[2] for British English), because creating a rule-based system would be too time-consuming, error-prone and above all exception-ridden, which means it would have to rely on extensive lists of lexical items to be treated specially anyway, so why not directly store the transcriptions for all items. A sequence-to-sequence mapping tool like *SequiturG2P* (Bisani and

[1] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[2] http://svr-www.eng.cam.ac.uk/comp.speech/Section1/Lexical/beep.html

Ney, 2008) can then optionally be used as a second step to automatically analyze recurring patterns in the dictionary items (pairs of orthographic and phonetic transcripts) and generate a stochastic grapheme-to-phoneme converter. This can be used as a less reliable[3] fallback procedure, allowing the system to handle out-of-vocabulary (OOV) items in a fairly cost-effective manner, as a "free" side-product of putting together the dictionary itself.

For Czech however, creating such a set of rules is a comparatively easy task (see (Psutka et al., 2006)), so this has become a *de facto* standard approach for the language in NLP applications in general and ASR in particular. Three potential problems with it come to mind:

- irregularities: mild (latinate words) and heavy (English loans, irregularly spelled / pronounced named entities); these may or may not be of interest to the application at hand, and if they are, exceptions for them may be hardcoded *ad hoc* into the rules;
- variants: the higher the frequency of a word, the more syllables it has "canonically" and the less formal the situation, the higher the likelihood that this word will have formally reduced pronunciation variants (see Klimešová et al. (2017, p. 153) for examples from Czech and Ernestus and Warner (2011) for a discussion of phonetic erosion in spontaneous speech in general); of course, dialectal variation can also be subsumed under this heading;
- connected speech processes: a potential problem for any system which considers transcription of lexical items atomically, without paying attention to context.

The inclination to ignore variation may be particularly strong with languages like Czech where a reasonably accurate rule-based pronunciation component can be built with comparatively little effort, but in general, whether the grapheme to phoneme component of the ASR system be lexicon- or rule-based, the latter two sources of variability tend to be downplayed. Since at the same time, they play a significant role in colloquial, spontaneous speech, it makes sense to ask whether this might be a handicap for ASR in some settings.

To explore this topic, the research presented in this paper leverages the ability of current ASR systems to handle a 1-to-many mapping in their pronunciation dictionary components. Whereas traditionally, these have often been "armchair language resources" (in analogy to the notion of "armchair linguist"), our approach was to sift through manual phonetic transcriptions of spontaneous speech and compile a lexicon of commonly attested pronunciation variants. Of course, the 1-to-many mapping addresses only some of the variability issues that ASR systems have to contend with. For a comprehensive overview, see Strik and Cucchiarini (1999); for example, differences in the temporal and spectral properties of instances of the "same" phone belong to the domain of the AM.

[3]Because it can only follow statistical trends and has no way of knowing about lexical exceptions.

2. Data and method

The procedures and overall setup are heavily based on the recipes for the Czech portion of the Vystadial telephone speech corpus (Korvas et al., 2014), more specifically their versions destined for the Kaldi ASR toolkit (Povey et al., 2011). For the most part, changes to the existing code were only cosmetic (adapting transcript and recording file preparation scripts for non-Vystadial data, fixing common typos in transcripts), except for the routines which generated the pronunciation lexicons, which were at the heart of our present undertaking (see 2.1.).

As far as data is concerned, Czech Vystadial data was also used in some of the experiments to offer a few basic comparisons, but the main focus was on recordings from the ORTOFON corpus of informal spontaneous spoken Czech (Kopřivová et al., 2014b). This is a corpus of intimate discourse recorded on a compact device (a Sony ICD-UX5xx series recorder) in natural settings (at home, at work, in cafés etc.), between groups of two or more well-acquainted people. It features two manually created transcription layers (a basic one, which is fairly close to accepted Czech orthography, and a phonetic one), as well as some additional paralinguistic annotation and sociolinguistic metadata (Kopřivová et al., 2014a). Data collection is ongoing and a sociolinguistically balanced sample of about 1M running words in length has been published in June 2017 via the online query interface at `https://korpus.cz`. It is also available for download in two different formats from the LINDAT/CLARIN repository (Kopřivová et al., 2017a; Kopřivová et al., 2017b).

The nature of the material entails a variety of challenges which can be faced with some success by human annotators, but are at present mostly insurmountable for automatic processing. These are generally related to the constraint that recordings are made on a single, preferably inconspicuously placed device, which is in turn dictated by a desire to capture natural linguistic behavior:

- finding a good compromise for placing the recording device and setting microphone sensitivity is hard, some speakers involved in the communication situation tend to be too close, others too far away;
- on a related note, depending on the setting, the speech can occasionally be drowned in noise (domestic appliances, vehicles passing by, café chatter), because microphone sensitivity needs to be set to pick up mid-range distance signals;
- unstructured interactions inevitably result in portions with overlapping speech by multiple speakers.

We keep track of the quality of the recording as an impressionistic rating made by its original transcriber. The sample of raw ORTOFON data used in the experiments reported in this article was selected from a population of utterances which contained no overlaps and were taken exclusively from recordings with the highest quality rating. To facilitate iterative development, the sample was deliberately kept relatively small (see Tab. 1) in order to speed

up training and decoding. As the main goal was not absolute performance but mutual comparisons between various approaches to generating the pronunciation dictionary, we feel this is a justified choice.

Table 1: Data sets employed in the experiments and their sizes: length of audio (hours:minutes), number of recordings, number of tokens.

data set	length	# recordings	# tokens
ORTOFON			
train	3:25	3978	38,593
dev	0:28	497	5081
test	0:26	497	4869
Vystadial			
train	15:25	22,567	126,333
dev	1:23	2000	11,478
test	1:22	2000	11,204

Following the original Vystadial experiments, all sound files were converted to mono WAV PCM sampled at a 16 kHz rate and 16 bit depth. Unlike the original Vystadial data however, ORTOFON data comes with information about speaker gender and identity across recordings, so these were specified in the relevant files where Kaldi asks for this information, instead of assuming the same gender everywhere and no recurring speakers.

2.1. Generating pronunciation variants

First of all, the vanilla rule-based pronunciation algorithm provided as part of the Vystadial scripts was used as a baseline (hereafter `vanilla`). It implements the best practice rules of Czech pronunciation as traditionally employed within the NLP community (Psutka et al., 2006) and yields exactly one pronunciation per lexicon item.

Pronunciation variants were then extracted from a working sample of the ORTOFON corpus about 1M running words in size; recall that these are hand-transcribed pronunciations spotted in naturally occurring colloquial Czech. Apart from recognizably legitimate variants, this database also turned out to contain some highly idiosyncratic ones as well as orth-to-phon alignment errors, identifiable as low frequency items or even *hapax legomena*. It quickly became clear that this variability and noise had to be trimmed down in order to become manageable by Kaldi. Two types of approaches were used: an automatic frequency based thresholding heuristic (2.1.1.) and manual filtering (2.1.2.). Consider that the two most phonetically variable lexical items, *protože* (*because*) and *prostě* (*simply*, and also a lexical filler similar to *like*), had 248 and 133 different pronunciations respectively. Given the fairly limited size of the training set, Kaldi needs help in distinguishing which of these to consider as even remotely viable candidates. In general, the goal is to span the continuous space of acoustic variability in as few discrete variants as possible, so that the ASR system only has to deal with useful and meaningful complexity and uncertainty.

Variability also leads to heightened homophony which particularly affects words that are short by nature or prone to drastic formal reduction as a result of their frequency of use. An especially insidious case of homophony is empty pronunciations. Even though these are linguistically well-motivated, because highly frequent function words might be completely elided in informal speech, as the speaker can expect listeners to be able to infer them based on context and their knowledge of the language, they were systematically removed since they might result in the spontaneous addition of words to the transcript during decoding with no corresponding acoustic evidence, if not properly constrained by a sufficiently strong LM.

The rule-based transcription procedure used as fallback for items not occurring in our variants database was based on `vanilla`, with a few small emendations. For instance, assimilation of voicing was reimplemented [4], and *j*-epenthesis between a close front vowel and another vowel was added.

2.1.1. Automatic threshold

The goal of the automatic thresholding procedure was to drastically reduce the maximum number of variants allowed for an item while retaining the spread in variability, i.e. the mapping should strive to preserve a distinction between highly, mildly and marginally variable items. Several options were explored and the following heuristics were retained in the end (L is a list of items from the pronunciation database sorted in decreasing order by their number of variants, M indicates the highest attested number of variants, i.e. the number of variants of the first element of L):

1. lexicon items were split into variability groups by dividing the interval $[0; M]$ into up to N non-overlapping intervals of size at least M/N, always extending the lower boundary to the next attested number of variants if the M/N step fell in between; groups were established based on membership of the items' variant counts in these intervals; items in the 1st group were limited to at most their N most frequent variants, the 2nd group to $N-1$, etc.; these were still subject to the additional filtering heuristics defined below;

2. *hapax* variants were discarded for items with at least one variant which had been seen multiple times;

3. variants containing rare phones, i.e. phones seen less than 10 times in the lexicon generated for a given experiment,[5], were discarded;

4. variants which were short (less than 2 phones) and homophonous (shared by multiple items) were discarded.

[4] The original algorithm uses a cascade of regular expressions. Since assimilation of voicing should spread across multiple neighboring phones, it cannot be implemented in one pass of regular expression substitutions, because these need to be linearly ordered, and new assimilation-triggering contexts may emerge in the process of applying them that would necessitate restarting from the top.

[5] This might seem like a low threshold, but in practice, there was quite a considerable gap between single-figure count phones and the rest.

Two versions of this approach were tested based on the value of N specified in heuristic 1 above, `thresh9` and `thresh4` ($N = 9$ and $N = 4$, respectively).

2.1.2. Manual filtering

A human expert in the phonetics of spontaneous Czech speech went through the pronunciation variants of the 100 most common word forms in the ORTOFON and Vystadial data sets and manually removed those that were deemed too ambiguous or poorly representative. Beyond this frequency peak, transcriptions were produced by the rule-based fallback method; in other words, in this setup, multiple pronunciation variants were allowed only for the most frequent words.

As in 2.1.1., two versions of this approach were tested. `manual1` was more lenient, allowing multiple similar variants both within and across items wherever they made sense and keeping the variants "as is". `manual2` was more aggressive, taking into account which differences are perceptually and acoustically salient and weeding out variants which were judged too similar to other ones. The variants themselves were also altered on occasion, mapping less frequently occurring phones (schwa, labiodental nasal) onto more well-attested counterparts.

We are well aware that the replicability of such a manual procedure is questionable. It would perhaps be better to characterize it by its purpose, which was to act as a sieve which is both *more aggressive* and *more intelligent* than the one defined purely based on frequency rules in 2.1.1.. The human expert acts as a post-editor of the decisions made by the original transcribers of the recordings. As such, s/he should have sufficient prior acquaintance with the material and training in phonetics in order to be able to:

- remove implausible variants
- substitute rare phones with related higher frequency phones[6]
- spot variants which are acoustically similar and retain only a single representative for the entire group

The variant lexicon is the permitted to include only such manually verified items. Together with the overall purpose defined above serving as guiding principle, these are the essential parameters of the manual filtering procedure to bear in mind when replicating it.

3. Results and discussion

AM training and LM creation followed the Vystadial recipe. We therefore performed experiments with zerogram and bigram language models[7] and the following acoustic models:

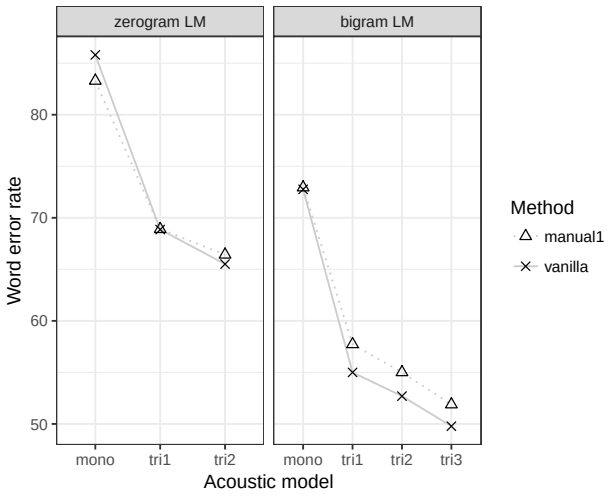


Figure 1: Word error rate results for different acoustic models, language models and pronunciation dictionary generation methods applied to the Vystadial data.

- `mono`: monophone model trained with MFCCs, Δ and $\Delta\Delta$ features;
- `tri1`: basic generative triphone model trained using Viterbi training; corresponds to `tri` $\Delta + \Delta\Delta$ in (Korvas et al., 2014);
- `tri2`: triphone model with Linear Discriminative Analysis (LDA, Haeb-Umbach and Ney (1992)) and Maximum Likelihood Linear Transformation (MLLT, Gopinath (1998)) feature transformations, trained using alignments from `tri1`; corresponds to `tri LDA+MLLT` in (Korvas et al., 2014);
- `tri3`: triphone model trained using discriminative Boosted Maximum Mutual Information (BMMI, Povey et al. (2008)) training on top of `tri2`; corresponds to `tri LDA+MLLT+BMMI` in (Korvas et al., 2014).[8]

Results are presented visually in Fig. 1 (for Vystadial data) and Fig. 2 (for ORTOFON data).

The Vystadial experiments were more computationally intensive because of the size of the data, so only `vanilla` and `manual1` methods were run. The `vanilla` results are comparable to those reported in (Korvas et al., 2014), suggesting that our basic experimental setup was sound. The word error rate (WER) for `manual1` is similar under the zerogram LM, suggesting that the acoustic models themselves are neither hampered nor (sadly) improved by having access to a wide array of pronunciation variants of highly frequent words. Under the bigram LM, except for the `mono` AMs, `manual1` is clearly outperformed by `vanilla`, suggesting that the amount of homophony

[6] For purely practical reasons: when training the ASR system, it is unlikely that a meaningful generalization would be inferred from just a handful of exemplars. Cf. the focus on "useful and meaningful complexity and uncertainty" in 2.1..

[7] Inferred from test and train data, respectively; the purpose of the zerogram model is "to evaluate solely the quality of the acoustic models without being affected by a language model or presence of out-of-vocabulary (OOV) words in the test set" (Korvas et al., 2014, p. 4426).

[8] Note that this model "needs a *language model* (LM) in order to compute the objective function. Here we use the [aforementioned] bigram LM" (Korvas et al., 2014, p. 4426), so the decoding performance with the zerogram LM has no clear interpretation and will be omitted from the results.

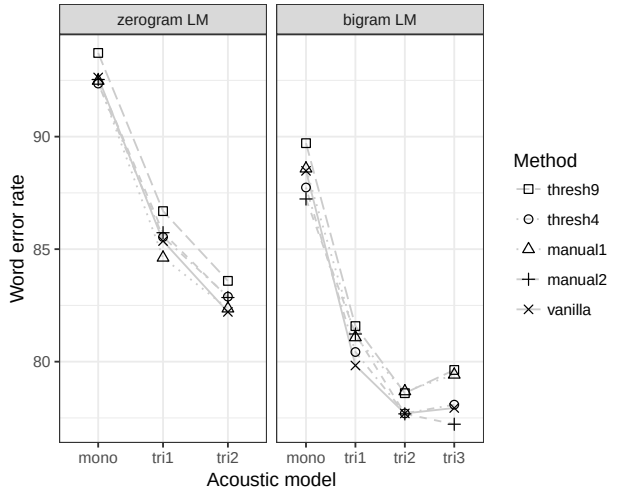


Figure 2: Word error rate results for different acoustic models, language models and pronunciation dictionary generation methods applied to the ORTOFON data.

introduced by the variants prevents the system from efficiently taking advantage of the predictive power of the LM when decoding.

With ORTOFON data, the more lenient automatic thresholding method `thresh9` evidently retains too much variability to be able to compete with the other ones even under the zerogram LM. As with Vystadial, `manual1` performs roughly on par with the others under the zerogram LM but lags behind along with `thresh9` under the bigram LM. The more stringent `thresh4` and `manual2` remain competitive with `vanilla`, with `manual2` providing the only hint that pronunciation variants might be performing useful work, as contrasted with `vanilla`, under the bigram LM and `tri3` AM: it is the only method that improves, if slightly, when BMMI is added. Of course, further work on larger data sets and more careful and systematic manual tweaks of the variants database will be needed to determine if this is indeed a viable path to follow.

In general, there seems to be a clear divide between the type of pronunciation variants human experts use to encode hints for other linguists / phoneticians to use in doing linguistic research, and the type of variation information an ASR system implementing current state-of-the-art methods can leverage. Or, to be more precise, the former needs to be thinned down quite drastically in order to become the latter, and even then, benefits are hard to glean. In light of this, it seems pointless to invest energy and resources into trying to extend the pronunciation rules for Czech to yield more than one pronunciation per item and thus account for empirically unattested but theoretically possible variants differing in voicing assimilation, vowel length or elision. However, judiciously adding attested high frequency variants should not be ruled out.

Frequency-based heuristics, whether implemented fully automatically or with manual cleanup, are fairly efficient at this sieving, but especially the manual ones, which only look at the frequency peak, also get rid of many potentially useful pronunciations of less common irregular words. Ideally, we should have additional criteria for inclusion that circumvent this drawback, e.g. based on the minimum edit distance between the hand-transcribed variant and the rule-generated variant.

Another topic for future research is that we have not yet investigated the option of providing frequency-based weights for the pronunciations, as Kaldi allows the possibility of a probabilistic pronunciation lexicon, instead of using binary thresholding mechanisms, or perhaps a combination of both approaches. Even more remotely, we may have discarded empty pronunciations as mentioned in 2.1., but they remain a valid notion from the linguistic point of view. They might just yet prove helpful in combination with stronger, more constraining LMs which do a better job at modeling language knowledge.

4. Conclusion

There is a hackneyed adage in the NLP community according to which firing the linguist / phonetician on the team invariably leads to an improvement in one's performance metric. The experiments reported in this article seem to corroborate this piece of folk wisdom, to the extent that linguists seem to be interested in details of variation which, while empirically motivated and linguistically relevant, result in confusion and an explosion of the search space when fed into an ASR system which has to bootstrap itself on limited amounts of data.

However, as we've seen with `manual2`, linguists might still have a useful contribution to bring to the table every now and then, if ever so modest, as long as they stick to another cliché saying: Less is more when it comes to pronunciation variants, at least in terms of how they fit together with the remaining components of current state-of-the-art ASR systems.

5. Acknowledgements

This research was made possible by the Czech National Corpus project (LM2015044) funded by the Ministry of Education, Youth and Sports of the Czech Rep. within the framework of Large Research, Development and Innovation Infrastructures.

6. Bibliographical References

Ernestus, M. and Warner, N. (2011). An introduction to reduced pronunciation variants. *Journal of Phonetics*, 39:253–260.

Gopinath, R. A. (1998). Maximum likelihood modeling with gaussian distributions for classification. In *Proceedings of ICASSP*, pages 661–664.

Haeb-Umbach, R. and Ney, H. (1992). Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *IEEE International Conference onAcoustics, Speech, and Signal Processing*, volume 1, pages 13–16. IEEE.

Klimešová, P., Komrsková, Z., Kopřivová, M., and Lukeš, D. (2017). Avenues for corpus-based research on informal spoken Czech. In Piotr Pęzik et al., editors, *Language, Corpora and Cognition*, volume 51 of *Łódź Studies in Language*, pages 145–162. Peter Lang Edition.

Kopřivová, M., Goláňová, H., Klimešová, P., Komrsková, Z., and Lukeš, D. (2014a). Multi-tier transcription of informal spoken Czech: The ORTOFON corpus approach. In *Complex Visibles Out There. Proceedings of the Olomouc Linguistics Colloquium 2014: Language Use and Linguistic Structure*, Olomouc Modern Language Series, Vol. 4, pages 529–544. Univerzita Palackého.

Kopřivová, M., Klimešová, P., Goláňová, H., and Lukeš, D. (2014b). Mapping diatopic and diachronic variation in spoken Czech: The ORTOFON and DIALEKT corpora. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 376–382. European Language Resources Association (ELRA).

Kučera, K. (1998). Vývoj účinnosti a složitosti českého pravopisu od konce 13. do konce 20. století [Development of orthographic efficiency and complexity in Czech texts from the end of the 13th century to the end of the 20th century]. *Slovo a slovesnost*, 59(3):178–199.

Povey, D., Kanevsky, D., Kingsbury, B., Ramabhadran, B., Saon, G., and Visweswariah, K. (2008). Boosted MMI for model and feature-space discriminative training. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4057–4060.

Psutka, J., Müller, L., Matoušek, J., and Radová, V. (2006). *Mluvíme s počítačem česky [Speaking Czech to the Computer]*. Academia.

Sampson, G. (2015). *Writing Systems*. Equinox Publishing Limited, second edition.

Strik, H. and Cucchiarini, C. (1999). Modeling pronunciation variation for ASR: A survey of the literature. *Speech Communication*, 29:225–246.

7. Language Resource References

Bisani, M. and Ney, H. (2008). Joint-sequence models for grapheme-to-phoneme conversion. *Speech Communication*, 50(5):434–451.

Kopřivová, M., Komrsková, Z., Lukeš, D., Poukarová, P., and Škarpová, M. (2017a). ORTOFON v1: balanced corpus of informal spoken czech with multi-tier transcription (transcriptions). `http://hdl.handle.net/11234/1-2580`.

Kopřivová, M., Komrsková, Z., Lukeš, D., Poukarová, P., and Škarpová, M. (2017b). ORTOFON v1: balanced corpus of informal spoken czech with multi-tier transcription (transcriptions & audio). `http://hdl.handle.net/11234/1-2579`.

Korvas, M., Plátek, O., Dušek, O., Žilka, L., and Jurčíček, F. (2014). Free English and Czech telephone speech corpus shared under the CC-BY-SA 3.0 license. In *Proceedings of the Eigth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 4423–4428.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlíček, P., Qian, Y., Schwarz, P., Silovský, J., Stemmer, G., and Veselý, K. (2011). The Kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.

Epitran: Precision G2P for Many Languages

David R. Mortensen, Siddharth Dalmia, Patrick Littell
Language Technologies Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
dmortens@cs.cmu.edu, sdalmia@andrew.cmu.edu, plittell@cs.cmu.edu

Abstract

Epitran is a massively multilingual, multiple back-end system for G2P (grapheme-to-phoneme) transduction which is distributed with support for 61 languages. It takes word tokens in the orthography of a language and outputs a phonemic representation in either IPA or X-SAMPA. The main system is written in Python and is publicly available as open source software. Its efficacy has been demonstrated in multiple research projects relating to language transfer, polyglot models, and speech. In a particular ASR task, Epitran was shown to improve the word error rate over Babel baselines for acoustic modeling.

Keywords: G2P, grapheme, phoneme, International Phonetic Alphabet, IPA, X-SAMPA, multilingual, speech, transfer, polyglot

1. Introduction

Epitran is a massively multilingual G2P system. To maximize its usefulness, it is written in Python and distributed as open source software under an MIT license. Out of the box, it supports 61 languages. Additional languages can easily be added using either a simple, rule-based framework or by adding other back-ends. It has a number of advantages over other G2P and romanization packages like Unitran and URoman including sensible handling of different Latin scripts, precision transduction for each language-script pair (important when multiple languages use the same script differently), and proper use of international and *de facto* standards for phonetic representation (IPA, X-SAMPA).

2. Motivation and Related Work

A number of Speech and NLP tasks require a G2P step—the conversion of orthographic representations to phonemic or phonetic representations (Daelemans and van den Bosch, 1996; Black et al., 1998; Sproat, 2006; Bisani and Ney, 2008; Laurent et al., 2009; Rao et al., 2015). Such tasks include TTS, ASR, the training of polyglot phonetic models for (non-speech) NLP tasks, and the implementation of approximate phonetic matching. G2P research has focused on some of the most difficult cases, including English. Less attention has been paid to providing G2P coverage for the wide range of languages with a more direct relationship between orthography and phonology. Many such languages require more than simple a mapping table for G2P but can be handled adequately by a more sophisticated rule-based system and certainly do not require a machine learning approach. This is fortunate, because many of these are also low resource languages in which sufficient training data, even for training a WFST, are not available but adequate linguistic descriptions do exist.

There are a few tools that already occupy this niche. Unitran is a tool that converts orthographic text to WorldBet and X-SAMPA (Qian et al., 2010). It is limited, though, in that it does not support Roman scripts and it does not have a mechanism for specifying different behavior for different languages that use the same script. For example, Amharic and Tigrinya both use the Ethiopic script and are treated exactly the same by Unitran even though their use of the Ethiopic script differs somewhat. URoman[1] is not a true G2P system; rather, it is a romanizer which generates an English-compatible romanization of the orthographic input rather than a phonemically sound output using an explicitly formalized standard (like WorldBet, X-SAMPA, or IPA). This makes it very useful for translating data into a form that English-speaking researchers can easily read and makes it somewhat useful for phonetically-driven entity linking tasks but limits its usefulness in speech and other tasks where the phonetic identity of segments is important. While Epitran is more limited in its coverage than Unitran and particularly URoman, it provides precision G2P that is easy to augment and debug on a language-by-language basis. It outputs both IPA—which can be consumed by related tools like PanPhon (Mortensen et al., 2016)—and X-SAMPA, which is now widely used in crosslingual speech applications.

A well-constructed Epitran mode will return a plausible IPA output for every well-formed input token. This provides a clear advantage over lexically-based resources for grapheme-to-phoneme tasks. This means there are effectively no vocabulary limits—there is a pronunciation for every word presented. The advantages are even more clear when compared to manually annotated data, as is often used in speech tasks. It is not uncommon for multiple annotators to have a low degree of agreement, particularly for phonemes that are phonetically close (differing in few articulatory features). This can cause inconsistency in training labels leading to a bad ASR engine. Epitran addresses this problem by providing a perfectly consistent mapping between graphemes and phonemes.

Epitran also places all converted data in a common phonetic space. This is useful when working with data coming from different sources (including data written in different scripts), or different languages and makes certain types of polyglot models possible.

It is reasonable to ask whether G2P is necessary any longer. After all, it has been shown that a respectable ASR system can be constructed for Vietnamese working directly from the orthography (Luong and Vu, 2016). Two points must be made here: Epitran is focused primarily at languages in

[1] `https://www.isi.edu/~ulf/uroman.html`

low data scenarios (though high resource languages are included for transfer purposes). With very little data, the G2P step becomes important. Relationships between orthography and pronunciation, which might be "obvious" to a system in a high data setting, are less obvious when data resources are constrained. Furthermore, the use of G2P is not limited to speech tasks alone. There are other NLP tasks, like approximate phonetic matching, where grapheme-to-phoneme conversion are important. Perhaps most significantly, certain models of cross-lingual transfer are facilitated by placing data from related languages in a common phonetic space (Bharadwaj et al., 2016). This topic is taken up at greater length in §6.2.

3. Interface

Epitran provides a simple Python (2 and 3) interface to all backends (See §4.):

```
>>> import epitran
>>> epi = epitran.Epitran('eng-Latn')
>>> epi.transliterate('Berkeley')
'bɹ̩kli'
>>> epi.xsampa_list('Berkeley')
['b','r\\=','k','l','i']
```

The primary API consists of methods on `Epitran` objects. The most important of these are `transliterate` and `xsampa_list`. The `transliterate` method takes a word token and returns its IPA representation. The `xsampa_list` method takes a word token a returns a list of X-SAMPA segments. Other methods include `trans_list` (like `transliterate` but returns a list of IPA segments) and `trans_delimiter` (like `transliterate` but returns IPA with a delimiter between segements).

4. Architecture

Epitran is structured as a single Python interface to various backends. The current backends are:

- **Flite** (for English)
- **Epihan** (for Simplified and Traditional Chinese)
- **Simple Epitran** (for all other languages)

The Flite backend requires the installation of the `lex_lookup` binary from a recent version of the Flite Speech Synthesis System (Black and Lenzo, 2001). The Epihan backend requires the installation of the open source CC-CEDict dictionary[2]. Most of the languages supported require the installation of no software or data other than Epitran itself. These use the Simple Epitran backend.

4.1. Simple Epitran

Every language with Simple Epitran support has a map file that defines a mapping between orthographic strings and phonemic strings. This is easy to produce and maintain. In many cases, it can be extracted automatically or semi-automatically from tables in existing resources. When it is not adequate, a preprocessor and a postprocessor can be added to manipulate the representations before and after mapping.

[2] `https://cc-cedict.org`

```
::consonant:: = b|bw|d|dw|d͡ʒ|d͡ʒw|f|fʲ|fw|h|hw|j|k|kʷ
|l|lw|m|mʲ|mw|n|nw|p|pw|pʼ|pʼw|q|qʰ|qʰw|qʷ|r|rʲ|rw|s|sw
|t|tw|tʼ|tʼw|t͡sʼ|t͡sʼw|t͡ʃ|t͡ʃw|t͡ʃʼ|t͡ʃʼw|v|vw|w|x|xʷ
|z|zw|ħ|ħw|ŋ|ŋw|g|gʷ|ɲ|ɲw|ʃ|ʃw|ʒ|ʒw|ʔ|ʔw|ʕ

0 -> ɨ / #(::consonant::)_(::consonant::)
0 -> ɨ / #(::consonant::)_#
0 -> ɨ / (::consonant::)_(::consonant::)#
0 -> ɨ / (::consonant::)_(::consonant::)(::consonant::)
```

Figure 1: The postprocessor from a Tigrinya (`tir-Ethi-red`) mode.

4.1.1. Map Files

Simple Epitran map files are simply two-column CSV files with a field for orthographic representation and a field for phonemic representation. They are interpreted greedily: The longest orthographic string matching a prefix of the input string is removed from the input string and the corresponding phonemic string is appended to the output string. This process proceeds iteratively until the input string is consumed. By default, input characters that are not found in the mapping table are added to the output unchanged. The order of the pairs in the map file is not significant.

4.1.2. Pre- and Post-Processors

While map files are easy to produce, often with the kinds of data available in references like Wikipedia[3] and Omniglot [4] as well as published grammars, they are not adequate for languages with complex mappings between orthography and phonology. For these circumstances, Epitran provides preprocessors and postprocessors. Each of these is essentially a cascade of context-sensitive rewrite rules. Figure 1 illustrates the structure of Simple Epitran preprocessors and postprocessors through an example of a Tigryina postprocessor. The first line (broken into four lines here) defines a constant, `::consonant::`, that can be used in the rules. Its value is a regular expression that matches any of the consonant phonemes in the language. The rules insert the default vowel /ɨ/ between two word-initial consonants, at the end of a word consisting only of a consonants, between word-final consonants, and between the second and third consonant of a three-consonant sequence. The function of this postprocessor in the Simple Epitran pipeline is illustrated in Figure 2. The input orthographic string passes through the preprocessor unchanged. It is then remapped into a phonemic space as defined in the mapping table. However, this representation does not include certain vowels whose distribution is predictable. One such vowel is inserted into 'bee', yielding /nɨhbi/ as the output.

4.2. Design Decisions

It should be evident that the preprocessors, postprocessrs, and maps all define regular relations and can thus be modeled using Finite State Transducers. Indeed, at an early stage of development, we considered using XFST or Foma as a basis for Epitran. There were three reasons we ulti-

[3] `http://www.wikipedia.org`
[4] `https://www.omniglot.com`

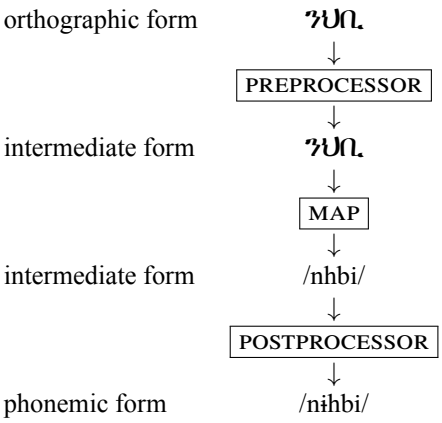


Figure 2: The Simple Epitran pipeline illustrated by the Tigrinya word ንህቢ ‘bee’.

mately chose not to do this (and to adopt—for Simple Epitran, at least—a pure Python implementation using string operations and regular expressions):

1. We wished to minimize the set of dependencies for Simple Epitran and wanted to limit them to Python libraries that could be installed from PyPI with `pip`.

2. We wanted a specific, lightweight format for the map files that could easily be extracted from sources with minimal processing. The greedy matching described above could be achieved through composing finite state transducers such that “rules” with a longer left-hand side would be ordered first, but this would require programmatic reordering of the list of mappings.

3. Some languages require metathesis (the “swapping” of substrings) in order to map orthographic order onto the temporal order needed for phonemic representation. It is possible to implement this using FSTs, but it is awkward to express using XFST/Foma’s rule notation. In Python, we were able to implement syntactic sugar that makes general metathesis rules easy to write.

5. Current Coverage

Epitran currently has coverage over the languages in Table 1. Languages annotated as “Provisional” need additional testing. “Naive phonemic” modes naively assume that the orthography of the language is transparently phonemic. A few languages written with Perso-Arabic scripts (Arabic, Persian, Urdu) are rendered without short vowels (which are not present in the orthographies). Reduced inventory modes are alternate modes for a language-script pair that reanalyze the phonemic system to minimize the number of phoneme types in the output.

Variation is an issue that inevitably arises in any attempt to model pronunciation. For Spanish, for example, their are numerous varieties that realize the same orthographic consonants differently. European varieties differ among themselves and also differ from Latin American varieties. For example ⟨z⟩ is realized as [θ] in Northern and Central Spain, but as [s] elsewhere in the Spanish-speaking world. Similarly, ⟨ll⟩ may be realized phonetically as [ʎ], [j̟] or [d͡ʒ], depending on the region. The mode `spa-Latn` represents a compromise Latin American Spanish, for historical reasons. In principle, there is no reason that a Castilian Spanish mode could not be added with modifying suffix appended to the language-script code. In practice, one variety—or a compromise variety—has been chosen for each ISO 639-3 code. In some cases, the choice of a compromise variety has been influenced by certain use cases. For example, the

Code	Language (Script)	Notes
aar-Latn	Afar	
amh-Ethi	Amharic	Naive mode
amh-Ethi-pp	Amharic	Precise phonemic mode
amh-Ethi-red	Amharic	Reduced inventory mode
ara-Arab	Arabic	No short vowels
aze-Cyrl	Azerbaijani (Cyrillic)	
aze-Latn	Azerbaijani (Latin)	
ben-Beng	Bengali	
cat-Latn	Catalan	
ceb-Latn	Cebuano	
cmn-Hans	Chinese (Simplified)	
cmn-Hant	Chinese (Traditional)	
ckb-Arab	Sorani	Provisional
deu-Latn	German	
deu-Latn-np	German	Naive phonemic
eng-Latn	English	
fas-Arab	Farsi	No short vowels
fra-Latn	French	Provisional
fra-Latn-np	French	Naive phonemic
hat-Latn-bab	Haitian	Following Babel
hau-Latn	Hausa	
hin-Deva	Hindi	
hun-Latn	Hungarian	
ilo-Latn	Ilocano	
ind-Latn	Indonesian	
ita-Latn	Italian	
jav-Latn	Javanese	
kaz-Cyrl-bab	Kazakh (Cyrillic)	Following Babel
kaz-Cyrl	Kazakh (Cyrillic)	
kaz-Latn	Kazakh (Latin)	
khm-Khmr	Khmer	Provisional
kin-Latn	Kinyarwanda	
kir-Arab	Kyrgyz (Perso-Arabic)	
kir-Cyrl	Kyrgyz (Cyrillic)	
kir-Latn	Kyrgyz (Latin)	
kmr-Latn	Kurmanji	
lao-Laoo	Lao	Provisional
mar-Deva	Marathi	
mon-Cyrl-bab	Mongolian	Following Babel
mlt-Latn	Maltese	
msa-Latn	Malay	
mya-Mymr	Burmese	Provisional
nld-Latn	Dutch	
nya-Latn	Chichewa	
orm-Latn	Oromo	
pan-Guru	Punjabi	
pol-Latn	Polish	
por-Latn	Portuguese	
rus-Cyrl	Russian	Provisional
sna-Latn	Shona	
som-Latn	Somali	
spa-Latn	Spanish	
swa-Latn	Swahili	
swe-Latn	Swedish	
tam-Taml	Tamil	
tel-Telu	Telugu	
tgk-Cyrl	Tajik	
tgl-Latn	Tagalog	
tha-Thai	Thai	
tir-Ethi	Tigrinya	Naive
tir-Ethi-pp	Tigrinya	Precise phonemic mode
tir-Ethi-red	Tigrinya	Reduced inventory mode
tuk-Cyrl	Turkmen (Cyrillic)	
tuk-Latn	Turkmen (Latin)	
tur-Latn-bab	Turkish	Following Babel
tur-Latn	Turkish	
uig-Arab	Uyghur	
ukr-Cyrl	Ukrainian	Provisional
urd-Arab	Urdu	No short vowels
uzb-Cyrl	Uzbek (Cyrillic)	
uzb-Latn	Uzbek (Latin)	
vie-Latn	Vietnamese	
xho-Latn	Xhosa	
yor-Latn	Yoruba	
zha-Latn	Zhuang	
zul-Latn	Zulu	

Table 1: Language modes in Epitran

⟨r⟩ phoneme in German and French has been mapped to [r] (found in conservative non-standard dialects) rather than [ʁ] or [ʀ]. This decision was made in order to minimize the phonetic distance between French and German words and related vocabulary items in other languages. Enriching the documentation to clarify the nature of these decisions is an important next step in improving Epitran's usability.

6. Downstream Evaluation and Applications

Evaluation of a multilingual resource like Epitran is difficult because each of the languages supported must effectively be evaluated separately. Furthermore, since Epitran concentrates on low resource languages, it is often difficult to find a sensible baseline and ground truth for many of the languages supported. Unitran could be used as a baseline for languages with non-Latin scripts but not for those with Latin scripts (the majority of Epitran languages). With these caveats in mind, we find good evidence from task-based evaluation that at least some of Epitran's language modes are very valuable.

6.1. ASR

To measure the performance of Epitran G2P in an ASR task, we trained an acoustic model of the EESEN ASR system (Miao et al., 2015) with a 6 layer bidirectional LSTM having 140 cells followed by a linear projection to 70 layer network after each LSTM[5]. The model takes in filter bank and pitch features extracted from telephonic audio recorded at 8kHz and produces a sequence of phonemes as the output. For the baseline experiment we used the G2P dictionary (lexicon file) provided as part of the dataset for each of the languages. For Epitran, we trained the same model by generating a new pronunciation file for all words in the baseline lexicon. For both the models we kept 5% of the data as a held out cross validation data to choose the best epoch.

We ground these phonemes to words using Weighted Finite State Transducers (WFSTs). We generate separate WFSTs to encode information about the tokens, grammar, and lexicon, and fuse them into a single compressed search graph which is used to generate the words. The details of the process can be seen in (Miao et al., 2015). This way we can compare the results on "gold transcription" text. We experimented on the eleven languages that were both in data released as part of the IARPA BABEL Research Program (IARPA-BAA-11-02) and for which there was a fully-developed Epitran mode. The word error rate results are shown in Table 2. In all cases, Epitran is competitive with the baseline. In a majority of cases, the word error rate for the system trained using Epitran has a lower word error rate that the baseline system.

One persistent advantage of Epitran (over lexical resources) is its ability to produce consistent and accurate pronunciations for all words that a system encounters, not just thus listed in the lexicon. This aids in several ASR-related tasks. For example, it makes it possible to build phoneme-based language models with no out-of-vocabulary (OOV) words. This has applications beyond speech technologies (in cross-lingual language applications). It also enables us to recognize words outside of the lexicon on which a model is trained. It can produce precise pronunciations of an utterance which can provide useful feedback on how the model is performing on OOV words during evaluation.

[5]The code to train can be found in `https://github.com/srvk/eesen/tree/tf_clean`

Language	Baseline WER	Epitran WER	Vocabulary Size
Amharic	58.6	**57.2**	36971
Cebuano	60.3	**57.1**	15534
Javanese	71.3	**65.7**	15541
Kazakh	60.7	**57.8**	22371
Kurmanji	70.9	**68.4**	14425
Swahili	**60.7**	61.2	18796
Tagalog	**54.6**	55.7	22627
Tamil	**74.8**	76.8	58484
Telugu	81.5	**77.9**	37654
Turkish	**55.7**	56.9	41157
Zulu	67.7	**65.2**	60627

Table 2: Word Error Rate (% WER) for the baseline model and a model trained with Epitran

6.2. Additional Applications

Epitran has been further applied to a number of tasks undertaken by the ARIEL team as part of the DARPA LORELEI program. In the speech domain, it has also been used in the cross-lingual and cross-domain transfer of ASR models built in Amharic to Tigrinya and Oromo. Since it performs G2P using similar rules for all the languages, the phonetic spaces to which the languages are mapped end up being very close. This is extremely important for transfer learning, especially when the target language data is minimal. It was also used to combine Amharic datasets from different sources by generating a new G2P dictionary. This technique is extremely useful for low resource languages where collecting data from multiple sources is essential. It has led to significant improvements in the robustness and performance of ARIEL's ASR.

In the language domain, it has been used to facilitate cross-lingual transfer of named entity annotations from Uzbek to Turkish and from Uzbek and Turkish to Uyghur by allowing the projection of all these languages into a common, phonetic space (Bharadwaj et al., 2016). It has also been used to transduce all languages in a large multilingual parallel corpus into IPA for training a polyglot machine translation model. Furthermore, it has been used in cross-lingual entity linking between languages like Amharic, Tigrinya, and Oromo, on the source side, and English, on the target side. Unfortunately, none of these cases were well adapted to ablation experiments in which the contribution of Epitran could be straightforwardly evaluated.

An additional application space for Epitran—and one that is even harder to evaluate—lies in helping non-native speaker linguists perform annotation tasks. While foreign scripts may be opaque even to linguistically-trained annotators, IPA representations are widely recognizable and read-

able. Epitran has been pivotal in allowing members of the ARIEL team to perform named entity annotation on languages like Uyghur, Amharic, and Tigrinya without special competence in these languages.

7. Conclusion

Epitran provides a lightweight and precise means of mapping orthographic data into the phonetic space. The out-of-the-box availability of many languages, as well as the ease with which high-quality modes may be added for new languages, make it a useful resource for researchers and developers working in the speech and cross-lingual NLP spaces.

8. Acknowledgements

This project was sponsored by the Defense Advanced Research Projects Agency (DARPA) Information Innovation Office (I2O), program: Low Resource Languages for Emergent Incidents (LORELEI), issued by DARPA/I2O under Contract No. HR0011-15-C-0114.

This work used releases IARPA-babel105b-v0.4, IARPA-babel106-v0.2g, IARPA-babel202b-v1.0d, IARPA-babel204b-v1.1b, IARPA-babel205b-v1.0a and IARPA-babel307b-v1.0b provided by IARPA BABEL Research Program

9. Bibliographical References

Bharadwaj, A., Mortensen, D., Dyer, C., and Carbonell, J. G. (2016). Phonologically aware neural model for named entity recognition in low resource transfer settings. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1462–1472, August.

Bisani, M. and Ney, H. (2008). Joint-sequence models for grapheme-to-phoneme conversion. Speech Communication, 50(5):434–451, May.

Black, A. W. and Lenzo, K. (2001). Flite: a small fast run-time synthesis engine. In ISCA, 4th Speech Synthesis Workshop, pages 157–162, Scotland.

Black, A. W., Lenzo, K., and Pagel, V. (1998). Issues in building general letter to sound rules. In 3rd ESCA Workshop on Speech Synthesis, pages 77–80.

Daelemans, W. and van den Bosch, A., (1996). Language-Independent Data-Oriented Grapheme-to-Phoneme Conversion, pages 77–90. Springer Verlag, New York.

Laurent, A., Deléglise, P., and Meignier, S. (2009). Grapheme to phoneme conversion using an SMT system. In 10th Annual Conference of the International Speech Communication Association 2009 (INTERSPEECH 2009), pages 716–719, Brighton, United Kingdom, September.

Luong, H.-T. and Vu, H.-Q. (2016). A non-expert Kaldi recipe for Vietnamese speech recognition system. In Proceedings of WLSI/OIAF4HLT, Osaka, Japan, December.

Miao, Y., Gowayyed, M., and Metze, F. (2015). EESEN: End-to-End Speech Recognition using Deep RNN Models and WFST-based Decoding. ArXiv e-prints, July.

Mortensen, D. R., Littell, P., Bharadwaj, A., Goyal, K., Dyer, C., and Levin, L. (2016). Panphon: A resource for mapping IPA segments to articulatory feature vectors. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3475–3484, Osaka, Japan, December.

Qian, T., Hollingshead, K., Yoon, S., Kim, K., and Sproat, R. (2010). A Python toolkit for universal transliteration. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta, may. European Language Resources Association (ELRA).

Rao, K., Peng, F., Sak, H., and Beaufays, F. (2015). Grapheme-to-phoneme conversion using Long Short-Term Memory recurrent neural networks. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4225–4229, April.

Sproat, R. (2006). A Computational Theory of Writing Systems. Cambridge University Press, Cambridge.

A Multilingual Approach to Question Classification

Aikaterini-Lida Kalouli, Katharina Kaiser, Annette Hautli-Janisz, Georg A. Kaiser, Miriam Butt
Department of Linguistics, University of Konstanz
firstname.lastname@uni-konstanz.de

Abstract

In this paper we present the Konstanz Resource of Questions (KRoQ), the first dependency-parsed, parallel multilingual corpus of information-seeking and non-information-seeking questions. In creating the corpus, we employ a linguistically motivated rule-based system that uses linguistic cues from one language to help classify and annotate questions across other languages. Our current corpus includes German, French, Spanish and Koine Greek. Based on the linguistically motivated heuristics we identify, a two-step scoring mechanism assigns intra- and inter-language scores to each question. Based on these scores, each question is classified as being either information seeking or non-information seeking. An evaluation shows that this mechanism correctly classifies questions in 79% of the cases. We release our corpus as a basis for further work in the area of question classification. It can be utilized as training and testing data for machine-learning algorithms, as corpus-data for theoretical linguistic questions or as a resource for further rule-based approaches to question identification.

Keywords: (non-)information-seeking questions, parallel multilingual corpora, question classification, Bible

1. Introduction

A central phenomenon in natural language as well as human-computer interaction is that of questions. Although this is a central phenomenon, it has been understudied in computational linguistics. Most of the existing work has concentrated on dealing with 'factoid' questions such as *When was Alan Turing at Bletchley Park?* This research has mostly been driven by the goal of building Question-Answering (QA) systems and finding intelligent, quick and reliable ways of matching a query to terms found in a given text collection, e.g., see Wang and Chua (2010). Comparatively less research has focused on understanding the structure of questions per se or on distinguishing different types of questions, i.e. information-seeking vs. rhetorical or discourse-structuring questions, among several other types. While a few approaches explicitly focus on non-information-seeking questions (Harper et al., 2009; Paul et al., 2011; Li et al., 2011), this work is either based on big data or on information gained from crowdsourcing. It does not tend to take recent theoretical linguistic work on questions into account, as has also been observed within the CReST project (Kübler et al., 2012), in which the existing PennTreebank (Marcus et al., 1993) annotation scheme was amended for yes-no and back-channeling questions.

This paper makes use of theoretical linguistic insights for automatically classifying questions as information-seeking or non-information-seeking. We devise a rule-based system with a heuristic scoring methodology that uses linguistic indicators to classify questions into information-seeking (ISQ) or non-information seeking questions (NISQs) across four different languages: German, French, Spanish and Koine Greek.[1] One result of our work has been the creation of a new resource: KRoQ (Konstanz Resource of Questions), a first dependency-parsed, parallel multilingual corpus of ISQs vs. NISQs.

The work described in this paper makes the following contributions to the field: For one, we provide a novel parsed and annotated corpus for question classification that can be used as a first reliable and improvable source for further theoretical and computational linguistic research. For another, we assign scores to determine how likely a question is information-seeking or not – these scores are also incorporated into the resource to make the classification more transparent and to be used for further research. Finally, our multilingual, rule-based technique can be applied as-is on further, multilingual data for question classification and can also be improved and adapted depending on the task.

Section 2. provides an overview of related work. In Section 3. we describe the corpus data and the linguistic indicators we made use of. Section 4. provides a description and evaluation of our multilingual system. Section 5. briefly describes the corpus we make available and in the final section we discuss our findings and our goals for future work.

2. Background

Everyday conversation frequently contains questions – in a randomly sampled 2-million tweets corpus compiled by Efron and Winget (2010), 13% of phrases are questions. But questions are far from being a homogeneous group. One type of question is posed to elicit information and get an answer — these are canonical, information-seeking questions (ISQs). Questions where the speaker does not expect an answer but instead triggers a certain type of speech act (Dayal, 2016) are treated as non-canonical, non-information-seeking questions (NISQs). The latter type is in itself a heterogenous class, including various subtypes.

2.1. Theoretical Linguistic Viewpoint

Perhaps the most well-known and well-recognized type of NISQ is a rhetorical question. This has the syntactic structure of a canonical question, but the pragmatic value of a declarative (Sadock, 1971; Han, 2002) and is often used to make a sarcastic comment or statement (*Have you ever even touched a computer?*). Another type, echo questions, are

[1] Dialect of Greek, also known as Alexandrian dialect, common Attic, Hellenistic or Biblical Greek; spoken and written during the Hellenistic and Roman Antiquity and the early Byzantine era.

used when the listener does not hear or understand properly what is being said, or when the listener wants to express incredulity or surprise (*She said what?*). Other types of NISQs are deliberative questions (*When shall we three meet again?*) (Wheatley, 1955) or self-addressed questions (*Where have I left my keys?*) (Ginzburg et al., 2013). A further well-known type are the ability/inclination questions, which are used as directives, requests or orders (*Can you pass the salt?*) (Dayal, 2016). Suggestive questions are used to imply that a certain answer should be given in response (*Don't you think that eating chips is unhealthy?*). In contrast, tag questions (*He is not coming, is he?*) are used when the speaker asserts something while also seeking confirmation for the assertion (Cattell, 1973). Looking at natural language corpora we also detect less-researched cases of NISQs, such as quoted questions (*She always asks "When will we meet?"*) or discourse-structuring questions (*What have we learned from this? We have learned that we need a better education system*). For the current work we do not employ a finer-grained distinction between these types but treat all of them as NISQ. Initial experimentation with a more detailed annotation scheme has shown that finer-grained subclassifications are difficult to be achieved consistently given the current state of our understanding of question types. We thus leave a finer-grained classification for further research.

2.2. Computational Approaches

In computational linguistics, one body of work uses social media data to classify ISQs and NISQs, training models on a limited set of manually annotated data (Harper et al., 2009; Li et al., 2011; Zhao and Mei, 2013; Ranganath et al., 2016). Paul et al. (2011) use crowdsourcing techniques to collect human classifications for a large amount of Twitter questions. While social media data has its own set of problems (e.g., length of the turn, ungrammaticality of sentences, spelling mistakes), the data is enriched with information like usernames, hashtags and urls, which helps in identifying the type of the question. Bhattasali et al. (2015) develop a machine-learning mechanism to identify rhetorical questions in the Switchboard Dialogue Act Corpus; Zymla (2014) uses a rule-based approach to heuristically identify rhetorical questions in German Twitter data.

The challenges for this type of work are manifold. First, distinguishing ISQs from NISQs based on syntactic properties is difficult because they are mostly structurally indistinguishable. Instead, context and intonation play a much bigger role (Bhatt, 1998; Zymla, 2014). Secondly, only some languages have special lexical markers that might indicate the type of question, e.g. German tends to use discourse particles in NISQs (Maibauer, 1986). Thirdly, expressions such as *give a damn*, *lift a finger* or *even*, which have been identified as generally conveying NISQs (Bhatt, 1998), are not frequent enough in real texts for computational purposes.

From a data perspective, it is not trivial to find suitable resources for looking into questions. Whereas real data in large quantities, e.g., Twitter, contains many questions of both types (Wang and Chua, 2010), the context in which they are found is either limited or lacking altogether and thus it is hard even for humans to decide on the two categories. On the other hand, corpora with well-edited text such as newspapers, books and speeches are generally less suitable since questions, in particular NISQs, tend to appear more often in spontaneous, unedited communication.

In order to overcome some of the challenges above, we developed our own rule-based system. This system leverages linguistic cues in one language for the scoring and classification of questions in other languages. This multilingual approach helps us in the classification because even if there are no indicators for the type of the question in one language, it is probable that there will be some in the other languages. The multilingual approach allows the pooling of information from multiple sources: the language of the question itself and the other three. The motivation to use such a multilingual approach goes back to Gale et al. (Gale et al., 1992), who used parallel corpora for word-sense-disambiguation (WSD). The logic behind such a technique is simple: get the things you cannot get from the current language from another language. In the WSD field this means that a polysemous word in one language can be looked up in parallel corpora and its translation in the other languages conveys the correct sense of the word in the current language. We developed our approach along similar lines: we identify linguistic cues in each of the languages and then use those cues to help question classification of the translations in the other languages.

3. The System

3.1. Data

Collection The data underlying our system is a parallel Bible corpus[2] in four languages, namely German, French, Spanish and Koine Greek. The choice of the four languages is deliberate. All four languages allow for the use of specific linguistic markers to indicate NISQs (Maibauer, 1986; Escandell, 1999; Bonifazi et al., 2016). Moreover, Koine Greek is the original language of the New Testament and the first language into which the Old Testament was translated and thus the language of the primary biblical text from which the Bible was translated into Latin (Tov, 2011). This means that it is the most suitable to be used as the prototypical version, which is crucial for our implementation.

We also deliberately decided on the Bible as a corpus. It is available in many languages, is inherently aligned, is for

[2]The Bible was crawled from online resources: German (translation of 1980, 73 books, `https://www.die-bibel.de/bibeln/online-bibeln/einheitsuebersetzung/bibeltext/`, Einheitsübersetzung der Heiligen Schrift © 1980 Katholische Bibelanstalt GmbH, Stuttgart), French (translation of 1997, 71 books, Textes bibliques tirés de la Bible en français courant © Société biblique française – Bibli'O, 1997 Avec autorisation. La responsabilité de la Société biblique française – Bibli'O est engagée uniquement sur les textes bibliques cités dans cet ouvrage. `http://lire.la-bible.net`), Spanish (translation of 1995, 66 books, `https://www.unitedbiblesocieties.org`, © 2018 United Bible Societies), Koine Greek Bible (Septuagint translation of the Old Testament of the 3rd century BCE and the Original New Testament, 77 books, `http://www.bibles.gr/`.

the most part written in prose and contains a large amount of narration and dialogues (Kaiser, 2015; de Vries, 2007).

Preprocessing The question extraction is rule-based in that we look for sentences ending with question marks of each of the four languages ('?' for French, German and Spanish and ';' for Greek). This provides us with 3,081 questions for German, 2,960 for French, 3,164 questions in Spanish and 3,300 in Koine Greek.[3] In addition, all texts except for the Koine Greek version are parsed with the Mate Parser (Bohnet and Nivre, 2012; Bohnet and Kuhn, 2012) and converted to the CoNLL-U format. We should note that the parsing is not used further in our approach; our approach solely uses the surface forms of the questions. Nevertheless, we wished to provide the corpus in a state-of-the-art, parsed format so that it is directly usable for further research.

3.2. Linguistic Indicators

Our system builds on language-specific markers that serve as possible indicators of ISQs vs. NISQs, based on ongoing theoretical linguistic work in the area. The core methodology of our system involves combining these insights with a multilingual corpus approach where markers in one language serve to classify questions across languages. It needs to be stressed that these cues do not represent absolute markers of ISQs vs. NISQs, but need to be taken as possible indicators of an ISQ vs. an NISQ. The higher the number of these indicators that can be found, particularly across languages, the higher the likelihood that the classification will be correct. As detailed below, we supplement information about the linguistic cues (originating in theoretical linguistic work) with heuristics we have developed ourselves based on observations of language structure in this and other corpora.

German Discourse particles are frequently found in German questions and in particular in NISQs (Maibauer, 1986; Zymla, 2014). These particles make subtle pragmatic contributions to an utterance and often convey a speaker's stance towards a proposition, situating that proposition in the web of information that comprises the discourse. We use the following as indicators for NISQs: *denn* 'lit. then', *schon* 'lit. already', *denn schon* 'lit. then already', *jemals* 'ever' and *niemals* 'never', following Maibauer (1986). The lexical item *ob* 'if' at the beginning of the question and its co-existence with *wohl* 'probably' as well as the tag element *oder?* 'right?' at the end of the question further serve as NISQ indicators.

French For French, we use *vraiment* 'really' and the tag phrase *n'est-ce pas?* 'isn't it?' at the end of the question as relevant NISQ indicators, in accordance with our own observations. We additionally use the presence of a negated predicate structure as indicative of NISQs (Maibauer, 1986; Sadock, 1971). Moreover, we make use of the fact that the French translation we used encodes direct speech with quotation marks. This allows us to track the dialogue turns with the assumption that a speaker who continues an utterance after posing a question is rather not seeking information, but is using the question to structure the dialog or to promote some other underlying meaning.

Spanish For Spanish we use *acaso* 'really' at the beginning of a question and *verdad?* 'true?' at the end of the question (Escandell, 1999). Both are considered good indicators for NISQs. The same goes for the tag element *no?* 'no?' at the end of a question. We also use the existence of negated predicate structures as NISQ indicators (Maibauer, 1986; Sadock, 1971).

Koine Greek For Koine Greek we also make use of particles, in particular some of those presented by Bonifazi et al. (2016): ἆρά γε 'maybe?', ποτε 'ever' and ἆρα 'maybe?'. We also take negated predicates as markers for NISQs (Maibauer, 1986; Sadock, 1971). In addition, we use our own observation for Koine Greek: If a question in Koine Greek is not translated as a question in the other languages, but as a declarative, we assume that this question is most probably a NISQ. In other words, if the translators chose to not translate it as a question, then the question is not asking for information but is rather accomplishing other communicative goals. This is also why it was important to choose Koine Greek as one of our languages — this assumption can only be made if we know the source language of the translation.

3.3. Scoring

The core methodology of the system is to use a scoring mechanism that indicates how strongly a question belongs to the group of ISQs or NISQs as no absolute markers are available. The higher the score of the question, the more likely it is to be an NISQ rather than an ISQ. The scoring is done in two steps, first an *individual scoring* where each question is analyzed individually for each language, and then a *cross-linguistic scoring* where we take into account the question and its translations.

Individual scoring We assign a score of either 1 or 2 to each of the linguistic indicators discussed in §3.2.. The score is based on the theoretical literature about how strongly each indicator correlates with being a NISQ. Table 1 provides an overview of the different indicators and their scores.

Heuristics	**Score**
German particles, *ob...wohl* and *oder?*	2
Spanish negated predicate	1
Spanish *no?*, *acaso* and *verdad?*	2
French *vraiment?* and negated predicate	1
French *n' est-ce pas?*	2
French dialogue turns	1
Greek particles	2
Greek negated predicate	1
Greek question not found in language X	2

Table 1: Scores of the heuristics used.

Then, we look for these indicators within each question and, if present, we add up their scores so that each question

[3] In Koine Greek 70 questions have no translation in any of the other languages (the questions of the books *1 Esdras*, *3 Maccabees*, *4 Maccabees* and *Sosana*) and thus only 3,230 were aligned and annotated.

is assigned an overall score (assuming that the initial score of each question is 0). This means that the more of the described patterns present in the question, the higher the question is scored. The following examples are meant to make the individual scoring clearer. If we have the question *Herr, mein Herr, was willst du mir schon geben?* 'Lord, my Lord, what do you *schon* ('really') want to give to me?', it is assigned a score of 2 because it contains the particle *schon* which has a score of 2. If the question would also include a further marker, e.g. *jemals* 'ever', then its score would have been increased by another 2 and the overall score of the question would be 4. Another example is the question *N'as-tu qu'une seule bénédiction?* 'Do you have only one blessing?'. The question gets an initial score of 1 because it starts with a negated predicate — the first part of the split expression *ne ... que'* ('only') — and another 1 because of the dialogue turns; if we look at the text following, we will see that the same person goes on speaking which we can tell because French conveniently marks direct speech with quotation marks as explained in section 3.2.. Thus, this question will get a total score of 2. This scoring has the benefit of showing tendencies – giving us something like a "quantified tendency" that a question belongs to one type or the other. The higher score means higher probability that the question will be NISQ rather than ISQ.

Cross-linguistic scoring After all questions of the four languages have been given individual scores, we assign the final score of each question (across languages) based on the individual scores of its translated instances. For that, we align the questions of the four languages based on the verse number of the Bible, e.g. the Spanish verse *Génesis 3:9* is mapped to the French *Génèse 3:9* and the question contained in each of them is mapped to each other. In the case that one verse contains more than one question, we map the complete verses because the corpus is not sentence-aligned. If one language does not have a question in a verse where the other languages feature a question, we only map the verses of those languages containing a question.

After the questions are aligned across languages, the individual scores of all aligned, translated instances of a question are added up to one final score. With this, every question — across the four languages — receives one overall, final score. The following example should make the cross-linguistic scoring clearer. The four aligned verses with the ID *Genesis 27:38* contain the question 'Did you have only one blessing, father?' in the four languages as shown in Table 2.

Language	Verse text	Score
German	Hattest du denn nur einen einzigen Segen, Vater?	2
French	N' as-tu qu' une seule bénédiction?	2
Spanish	¿No tienes más que una sola bendición, padre mío?	1
Greek	ἡ εὐλογία μία σοί ἐστι, πάτερ·	0
	Cross-linguistic Score	**5**

Table 2: Example of the cross-linguistic scoring.

The individual scoring described in the previous subsection assigns a score to each translated instance of the question, based on the existence of the predefined markers. After the four translated instances have been aligned across the languages, the individual scores of all four language instances are added up to one total, cross-linguistic score. With this the question *Genesis 27:38* gets the final overall score of 5.

3.4. Classification

The final step of classifying and annotating the questions as either ISQs or NISQs is solely based on the scoring; there is no training process involved. Different classifications are possible depending on what score is taken to be the threshold for the classification. In general, the lower the score, the more likely the question is an ISQ. The higher the score, the stronger the tendency that the question is an NISQ. For the current version of the KRoQ corpus we provide, we take 0 to be the threshold and thus classify all questions with scores equal to 0 as ISQ and questions with scores greater than 0 as NISQ because this proved the best setting, as shown in the next section.

4. Evaluation

Data In order to evaluate the appropriateness of our heuristics, we manually created a gold standard of the first 200 aligned questions of the Bible, annotating them as ISQ vs. NISQ. Questions that were only realized as questions in German or Spanish or French, but which were declaratives in all other three languages were considered translation anomalies and were left out from the evaluation. However, if the question was only found in Koine Greek as a question and as a declarative in the other languages, it was still considered for evaluation (see §3.2.). Questions of one language that did not exist in the other languages at all (neither as questions nor as declaratives) were also not included in the evaluation set. We also excluded direct questions containing other questions in direct speech, as in those cases we could not decide if we should annotate the main or the embedded question (e.g. *Warum lacht Sara und sagt : Soll ich wirklich noch Kinder bekommen, obwohl ich so alt bin?* 'Why does Sara laugh and say: am I really going to have children, although I am already so old?').

The 200 question instances of each language (800 in total) were each manually annotated by two expert annotators. We then took all eight manual annotations for each question across languages and went with the majority vote, yielding an inter-annotator agreement of 0.75%. We assume that the same question has the same status (ISQ or NISQ) across languages: the same parallel question has the same context and co-referents in all languages and thus the same status. A question was therefore classified as ISQ or NISQ across languages based on what most annotators had marked it as.

Results To evaluate the quality of our system, we compare the automatic scoring (and thus classification) with the manually-created gold standard. In order to test how the system performs given the score, we employ three different evaluation settings where we set different thresholds and compare the results. In setting 1 – which is also the one applied on the provided corpus – we classify all questions with total scores equal to 0 as ISQ and all others with total

scores greater than 0 as NISQ; this means that all questions where at least one NISQ indicator is found are taken to be NISQ. In setting 2, all questions with a score higher than 1 are treated as NISQs, in setting 3 the threshold is set at 2. The two latter settings classify questions as NISQ only if more than one of the indicators are present in the questions. By comparing the different settings we could investigate how many indicators are necessary in order to correctly annotate the questions.

The results are shown in Table 3. In setting 1 the automatic system achieves an F-score of 0.83. In settings 2 and 3, precision increases, however at the cost of a significantly lower recall. This means that setting 1, in which we consider all questions where at least one linguistic indicator is found, is the best performing setting. This result tallies with what the theoretical literature has found: First, there is no linguistic cue that consistently marks NISQs. Secondly, one linguistic indicator per question might be sufficient for classifying the question correctly but raising the threshold to two or more indicators improves the precision. This means that the more indicators available and the more languages we can test in parallel, the better and more reliable the classification results become. However, the higher thresholds fail to capture many cases, leading to a low recall. Thirdly, the linguistic indicators that we chose, e.g., particles, are well-motivated and allow for a robust classification.

	Setting 1	Setting 2	Setting 3
Precision	0.85	0.89	0.97
Recall	0.82	0.5	0.31
F-score	**0.83**	0.64	0.46
Accuracy	0.79	0.62	0.53

Table 3: Evaluation results

As was reported in section 2.2., the previous work in the field uses different training and testing data, e.g. Twitter data, the Switchboard Dialogue Act Corpus, etc., which means that no direct comparison of the systems is possible. Nevertheless, the absolute results of our rule-based approach, in particular those of setting 1, are comparable to some of these machine-learning approaches, showing a higher F-score than Ranganath et al. (2016) and Bhattasali et al. (2015) (F-scores of 64,04% and 53,71%, respectively) and a comparable accuracy to Li et al. (2011) and Zhao and Mei (2013) who report an accuracy of 77,5% and 79%, respectively. It is only in comparison to the system of Harper et al. (2009) with 89,7% accuracy, that our system shows a lower score. Note that it is difficult to conduct a uniform comparison of systems as they rely on different approaches and different data. Our rule-based mechanism is designed for parallel, multilingual corpora. As mentioned in the introduction and as discussed in the conclusion, our approach can be seen as a first step for annotating large, parallel datasets so that these can be further used as training data for machine-learning approaches.

5. The KRoQ Corpus

The first version of the KRoQ corpus is made available under `https://github.com/kkalouli/BIBLE-processing`[4] and contains the French, the Spanish and the Greek Bible texts, along with our annotations. The German Bible text cannot be made available because of copyright restrictions. For French and Spanish, the corpus is provided in the CoNLL-U format to facilitate further processing of the corpus. For consistency, the Greek text is also provided in this format, but is not parsed. Every sentence of each of the Bible translations contains its original Bible index (e.g. Génèse 1:1 for Genesis, Chapter 1, Verse 1), the original sentence, its parsed structure and a comment field named "annotation". This field captures the annotations of the questions and is left blank for all non-questions. For the questions, the annotation field contains the final score that was automatically assigned across the four languages and our annotation based on that score and on setting 1, as presented in Section 4.

For better reproduction of the results, we provide the gold standard files we used in addition to the main corpus. The gold standard contains the first 200 aligned questions of the Bible for each of the four languages. For German, French and Spanish, the "gold" questions are provided in the CoNLL-U format. For Greek they are provided as raw text. Each item of the gold standard is numbered according to its occurrence in the Bible and contains the original question, its Bible index (e.g. Génèse 1:1 for Genesis, Chapter 1, Verse 1) and the parsed structure. The annotations are captured in the comment fields of each item: the final score that was automatically assigned across the four languages and the annotation based on that score and on setting 1 of Section 4 (in the comment field *annotation*) and the manual, gold standard annotation of ISQ vs. NISQ (in the comment field *gold_annotation*). In order to see the alignment across languages, we add a spreadsheet where the 200 questions and their instances in the other languages are aligned.

6. Conclusions and Future Work

The performance achieved by our current system (in the first setting) shows that the generated annotated corpus can be used as a reliable resource for further research. Our annotated data can be utilized as training and testing data for machine-learning algorithms, as corpus-data for theoretical linguistic questions and as a resource for further symbolic approaches. Additionally, the quality of the classification gives us confidence that the implemented mechanism can be used to classify further data, which can then be used to augment the existing resource.

In our future work we would like to pursue three courses of action. First, we would like to improve the system by adding more parallel languages because additional languages are bound to give us additional markers which can help us distinguish more reliably between the types of questions (as indicated by the second and third evaluation setting). Second, we would like to investigate a different multilingual corpus to see if our system also delivers satisfactory results for other kinds of corpora. As a third goal, we would like to use the annotated data as training data for a machine-learning approach.

[4] Due to its large size the corpus cannot be made available through the LRE Map.

7. Acknowledgements

We thank the German Research Foundation (DFG) for the financial support within the projects P8 "Questions Visualized" and P2 "Word Order Variation in Wh-questions: Evidence from Romance" of the Research Group FOR 2111 "Questions at the Interfaces" at the University of Konstanz.

8. Bibliographic References

Bhatt, R. (1998). Argument-adjunct asymmetries in rhetorical questions. In *North East Linguistic Society (NELS 29)*, Dalware.

Bhattasali, S., Cytryn, J., Feldman, E., and Park, J. (2015). Automatic Identification of Rhetorical questions. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, page 743–749.

Bohnet, B. and Kuhn, J. (2012). The Best of Both Worlds – A Graph-based Completion Model for Transition-based Parsers. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 77–87.

Bohnet, B. and Nivre, J. (2012). A Transition-Based System for Joint Part-of-Speech Tagging and Labeled Non-Projective Dependency Parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1455–1465, Jeju Island, Korea.

Bonifazi, A., Drummen, A., and de Kreij, M. (2016). *Particles in Ancient Greek Discourse: Five Volumes Exploring Particle Use across Genres*. Hellenic Studies Series 74. Washington, DC: Center for Hellenic Studies, Washington, DC.

Cattell, R. (1973). Negative Transportation and Tag Questions. 49(3):612–639.

Dayal, V. (2016). *Questions*. Oxford University Press, Oxford.

de Vries, L. (2007). Some remarks on the use of Bible translations as parallel texts in linguistic research. 60:148–157.

Efron, M. and Winget, M. (2010). Questions are content: a taxonomy of questions in a microblogging environment. In *Proceedings of ASIST '10*.

Escandell, V. V. (1999). Los enunciados interrogativos. Aspectos semánticos y pragmáticos. In I. Bosque et al., editors, *Gramática descriptiva de la lengua española. Vol. 3: Entre la oración y el discurso. Morfología*, pages 3929–3991. Espasa Calpe, S.A, Madrid.

Gale, A., Church, K., and Jarowsky, D. (1992). A Method for Disambiguating Word Senses in a Large Corpus. *Computer and Humanities*, 26(5/6):415–439.

Ginzburg, J., Fernandez, R., and Schlangen, D. (2013). Self-addressed questions in disfluencies. In *Proceedings of Disfluency in Spontaneous Speech (DiSS) 2013*.

Han, C. (2002). Interpreting interrogatives as rhetorical questions. *Lingua*, 112:201–229.

Harper, F., Moy, D., and Konstan, J. A. (2009). Facts or Friends? Distinguishing Informational and Conversational Questions in Social Q&A Sites. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI 2009)*, pages 759–768.

Kaiser, G. A. (2015). Zur Verwendung von Bibelübersetzungen in der (romanistischen) Sprachwissenschaft. In *Biblicum Jassyense*, pages 5–18.

Kübler, S., Baucom, E., and Scheutz, M. (2012). Parallel syntactic annotation in CReST. *Linguistic Issues in Language Technology (LiLT)*, 7(4).

Li, B., Si, X., Lyu, M. R., King, I., and Chang, E. Y. (2011). Question Identification on Twitter. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management (CIKM'11)*.

Maibauer, J. (1986). *Rhetorische Fragen*. Max Niemeyer Verlag, Tübingen, 1 edition.

Marcus, M., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Paul, S. A., Hong, L., and Chi, E. H. (2011). What is a question? Crowdsourcing tweet categorization. In *CHI 2011, Workshop on Crowdsourcing and Human Computation*.

Ranganath, S., Hu, X., Tang, J., SuhangWang, and Liu, H. (2016). Identifying Rhetorical Questions in Social Media. In *Proceedings of the 10th International AAAI Conference on Web and Social Media (ICWSM 2016)*.

Sadock, J. (1971). Queclaratives. In D. Adams, et al., editors, *Papers from the 7th Regional Meeting of the Chicago Linguistic Society*, pages 223–232, April.

Tov, E. (2011). Early Bible translations into Greek and renderings derived from them. In H. Kittel, et al., editors, *Ein internationales Handbuch zur Übersetzungsforschung. An International Encyclopedia of Translation Studies. Encyclopédie internatioonale de la recherche sur la traducion.*, volume 3, pages 2351–2354. Berlin: de Gruyter.

Wang, K. and Chua, T.-S. (2010). Exploiting salient patterns for question detection and question retrieval in community based question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING10)*, page 1155–1163.

Wheatley, J. M. O. (1955). Deliberative questions. *Analysis*, 15(3):49–60.

Zhao, Z. and Mei, Q. (2013). Questions about Questions: An Empirical Analysis of Information Needs on Twitter. In *Proceedings of the International World Wide Web Conference Committee (IW3C2)*, pages 1545–1555.

Zymla, M.-M. (2014). Extraction and Analysis of non-canonical Questions from a Twitter-Corpus. Master thesis, University of Konstanz.

Dataset for the First Evaluation on Chinese Machine Reading Comprehension

Yiming Cui[†], Ting Liu[‡], Zhipeng Chen[†], Wentao Ma[†], Shijin Wang[†] and Guoping Hu[†]
[†]Joint Laboratory of HIT and iFLYTEK, iFLYTEK Research, Beijing, China
[‡]Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology, Harbin, China
[†]{ymcui,zpchen,wtma,sjwang3,gphu}@iflytek.com
[‡]tliu@ir.hit.edu.cn

Abstract

Machine Reading Comprehension (MRC) has become enormously popular recently and has attracted a lot of attention. However, existing reading comprehension datasets are mostly in English. To add diversity in reading comprehension datasets, in this paper we propose a new Chinese reading comprehension dataset for accelerating related research in the community. The proposed dataset contains two different types: cloze-style reading comprehension and user query reading comprehension, associated with large-scale training data as well as human-annotated validation and hidden test set. Along with this dataset, we also hosted the first Evaluation on Chinese Machine Reading Comprehension (CMRC-2017) and successfully attracted tens of participants, which suggest the potential impact of this dataset.

Keywords: Chinese Reading Comprehension, Question Answering, Evaluation

1. Introduction

Machine Reading Comprehension (MRC) has become enormously popular in recent research, which aims to teach the machine to comprehend human languages and answer the questions based on the reading materials. Among various reading comprehension tasks, the cloze-style reaing comprehension is relatively easy to follow due to its simplicity in definition, which requires the model to fill an exact word into the query to form a coherent sentence according to the document material. Several cloze-style reading comprehension datasets are publicly available, such as CNN/Daily Mail (Hermann et al., 2015), Children's Book Test (Hill et al., 2015), People Daily and Children's Fairy Tale (Cui et al., 2016).

In this paper, we provide a new Chinese reading comprehension dataset[1], which has the following features

- We provide a large-scale automatically generated Chinese cloze-style reading comprehension dataset, which is gathered from children's reading material.
- Despite the automatic generation of training data, our evaluation datasets (validation and test) are annotated manually, which is different from previous works.
- To add more diversity and for further investigation on transfer learning, we also provide another evaluation datasets which is also annotated by human, but the query is more natural than the cloze type.

We also host the 1st Evaluation on Chinese Machine Reading Comprehension (CMRC2017), which has attracted over 30 participants and finally there were 17 participants submitted their evaluation systems for testing their reading comprehension models on our newly developed dataset, suggesting its potential impact. We hope the release of the dataset to the public will accelerate the progress of Chinese research community on machine reading comprehension field.

We also provide four official baselines for the evaluations, including two traditional baselines and two neural baselines. In this paper, we adopt two widely used neural reading comprehension model: AS Reader (Kadlec et al., 2016) and AoA Reader (Cui et al., 2017).

The rest of the paper will be organized as follows. In Section 2, we will introduce the related works on the reading comprehension dataset, and then the proposed dataset as well as our competitions will be illustrated in Section 3. The baseline and participant system results will be given in Section 4 and we will made a brief conclusion at the end of this paper.

2. Related Works

In this section, we will introduce several public cloze-style reading comprehension dataset.

2.1. CNN/Daily Mail

Some news articles often come along with a short summary or brief introduction. Inspired by this, Hermann et al. (2015) release the first cloze-style reading comprehension dataset, called CNN/Daily Mail[2]. Firstly, they obtained large-scale CNN and Daily Mail news data from online websites, including main body and its summary. Then they regard the main body of the news as the *Document*. The *Query* is generated by replacing a name entity word from the summary by a placeholder, and the replaced named entity word becomes the *Answer*. Along with the techniques illustrated above, after the initial data generation, they also propose to anonymize all named entity tokens in the data to avoid the model exploit world knowledge of specific entities, increasing the difficulties in this dataset. However,

[1]CMRC 2017 Public Datasets: https://github.com/ymcui/cmrc2017.

[2]The pre-processed CNN and Daily Mail datasets are available at http://cs.nyu.edu/~kcho/DMQA/

		Cloze Track		User Query Track	
	Train	Validation	Test	Validation	Test
# Query	354,295	2,000	3,000	2,000	3,000
Max # tokens in docs	486	481	484	481	486
Max # tokens in query	184	72	106	21	29
Avg # tokens in docs	324	321	307	310	290
Avg # tokens in query	27	19	23	8	8
Vocabulary			94,352		

Table 1: Statistics of the dataset for the 1st Evaluation on Chinese Machine Reading Comprehension (CMRC-2017).

as we have known that world knowledge is very important when we do reading comprehension in reality, which makes this dataset much artificial than real situation. Chen et al. (2016) also showed that the proposed anonymization in CNN/Daily Mail dataset is less useful, and the current models (Kadlec et al., 2016; Chen et al., 2016) are nearly reaching ceiling performance with the automatically generated dataset which contains much errors, such as coreference errors, ambiguous questions etc.

2.2. Children's Book Test

Another popular cloze-style reading comprehension dataset is the Children's Book Test (CBT)[3] proposed by Hill et al. (2015) which was built from the children's book stories. Though the CBT dataset also use an automatic way for data generation, there are several differences to the CNN/Daily Mail dataset. They regard the first 20 consecutive sentences in a story as the *Document* and the following 21st sentence as the *Query* where one token is replaced by a placeholder to indicate the blank to fill in. Unlike the CNN/Daily Mail dataset, in CBT, the replaced word are chosen from various types: Name Entity (NE), Common Nouns (CN), Verbs (V) and Prepositions (P). The experimental results showed that, the verb and preposition answers are not sensitive to the changes of document, so the following works are mainly focusing on solving the NE and CN genres.

2.3. People Daily & Children's Fairy Tale

The previously mentioned datasets are all in English. To add diversities to the reading comprehension datasets, Cui et al. (2016) proposed the first Chinese cloze-style reading comprehension dataset: People Daily & Children's Fairy Tale, including People Daily news datasets and Children's Fairy Tale datasets. They also generate the data in an automatic manner, which is similar to the previous datasets. They choose short articles (several hundreds of words) as *Document* and remove a word from it, whose type is mostly named entities and common nouns. Then the sentence that contains the removed word will be regarded as *Query*. To add difficulties to the dataset, along with the automatically generated evaluation sets (validation/test), they also release a human-annotated evaluation set. The experimental results show that the human-annotated evaluation set is significantly harder than the automatically generated questions. The reason would be that the automatically generated data is accordance with the training data which is also automatically generated and they share many similar characteristics, which is not the case when it comes to human-annotated data.

3. The Proposed Dataset

In this section, we will briefly introduce the evaluation tracks and then the generation method of our dataset will be illustrated in detail.

3.1. The 1st Evaluation on Chinese Machine Reading Comprehension (CMRC-2017)

The proposed dataset is typically used for the 1st Evaluation on Chinese Machine Reading Comprehension (CMRC-2017)[4], which aims to provide a communication platform to the Chinese communities in the related fields. In this evaluation, we provide two tracks. We provide a shared training data for both tracks and separated evaluation data.

- Cloze Track: In this track, the participants are required to use the large-scale training data to train their cloze system and evaluate on the cloze evaluation track, where training and test set are exactly the same type.
- User Query Track: This track is designed for using transfer learning or domain adaptation to minimize the gap between cloze training data and user query evaluation data, i.e. training and testing is fairly different.

Following Rajpurkar et al. (2016), we preserve the test set only visible to ourselves and require the participants submit their system in order to provide a fair comparison among participants and avoid tuning performance on the test set. The examples of Cloze and User Query Track are given in Figure 1.

3.2. Definition of Cloze Task

The cloze-style reading comprehension can be described as a triple $\langle \mathcal{D}, \mathcal{Q}, \mathcal{A} \rangle$, where $\mathcal{D}$ represents **D**ocument, $\mathcal{Q}$ represents **Q**uery and the $\mathcal{A}$ represents **A**nswer. There is a restriction that the answer should be a single word and should appear in the document, which was also adopted in (Hill et al., 2015; Cui et al., 2016). In our dataset, we mainly focus on answering common nouns and named entities which require further comprehension of the document.

[3]Available at `http://www.thespermwhale.com/jaseweston/babi/CBTest.tgz`

[4]CMRC 2017 Official Website: `http://www.hfl-tek.com/cmrc2017/index.html`.

	Cloze Track	User Query Track
Document	1 ||| 为了 让 森林 变 得 更加 茂盛 ， 大伙 都 在 努力 地 工作 着 。 1 ||| To let the forest become more lush, every is working hard. 2 ||| **XXXXX** 每天 天 不 亮 就 起来 ， 在 这 棵 树 上 啄啄 ， 在 那 棵 树上 敲敲 ， 他 的 尖利 的 长嘴 ， 使 害虫 没有 藏身 之 地 。 2 ||| **XXXXX** wake up very early everyday, digging the tree trunk with its sharp beak and there is no hiding place for the insects. ..... 5 ||| 惟有 大象 没 活 干 ， 他 整天 游荡 ， 大家 问 他 为什么 不 干活 ， 他 说 ： “ 我 没有 **啄木鸟** 的 长嘴 ， 也 没有 猴子 的 巧手 和 松鼠 的 尖牙利爪 ， 我 能 干 什么 呢 ？ 5 ||| He said: “I have no beak as **woodpecker** and no hands like monkey or squirrel. So what can I do?” ... 13 ||| 不久 ， 在 原先 堆 着 枯树干 的 地方 ， 长 出 一 支 支 小 绿苗 。 13 ||| Soon, there are a few green seedlings where the dead tree trunk piled. 14 ||| 大伙 夸奖 大象 有 一 只 多么 能干 的 鼻子 。 14 ||| Everyone praised the elephant that he has a competent nose.	1 ||| 一 只 **驴子** ， 捎 着 木料 ， 向前 走 去 。 1 ||| The **donkey** is going forward carrying wood. 2 ||| 一不小心 ， 摔 在 池 里 ， 辗转 不 能 出 水 ， 他 便 唉声叹气 地 悲哀 起来 。 2 ||| Accidentally, it fell into the pool and cannot get out from it with sad sighs. 3 ||| 许多 蛙 ， 是 生惯 在 池里 的 ， 他们 听见 了 驴子 的 呼救声 ， 都 来 围观 。 3 ||| Many frogs are used to live in the pool. They heard the cry for help and head for that. 4 ||| 他们 对 驴子 说 ： " 你 不过 在 池 里 只 一刻儿 功夫 ， 便 这样 地 大 嚷着 救命 ， 请 告诉 我们 这 是 什么 缘故 ？ 4 ||| They talk to the donkey: “Why you are shouting for staying in the water only for a moment?” 5 ||| 万一 你 像 我们 一样 无穷期 地 居住 在 这里 ， 你 又 得 怎样 呢 ？ 5 ||| What if you live here like us? 6 ||| " 这 便是 群蛙 给予 驴子 的 讥刺 的 慰藉 。 6 ||| The frogs are giving gibing comfort to the donkey.
Query	**XXXXX** 每天 天 不 亮 就 起来 ， 在 这 棵 树 上 啄啄 ， 在 那 棵 树上 敲敲 ， 他 的 尖利 的 长嘴 ， 使 害虫 没有 藏身 之 地 。 **XXXXX** wake up very early everyday, digging the tree trunk with its sharp beak and there is no hiding place for the insects.	谁 在 大 嚷 着 救命 ？ Who was shouting for help?
Answer	啄木鸟 Woodpecker	驴子 Donkey

Figure 1: Examples of the proposed datasets (the English translation is in grey). The sentence ID is depicted at the beginning of each row. In the Cloze Track, “XXXXX” represents the missing word.

3.3. Automatic Generation

Following Cui et al. (Cui et al., 2016), we also use similar way to generate our training data automatically. Firstly we roughly collected 20,000 passages from children’s reading materials which were crawled in-house. Briefly, we choose an answer word in the document and treat the sentence containing answer word as the query, where the answer is replaced by a placeholder “XXXXX”. The detailed procedures can be illustrated as follows.

- **Pre-processing**: For each sentence in the document, we do word segmentation, POS tagging and dependency parsing using LTP toolkit (Che et al., 2010).
- **Dependency Extraction**: Extract following dependencies: COO, SBV, VOB, HED, FOB, IOB, POB[5], and only preserve the parts that have dependencies.
- **Further Filtering**: Only preserve SBV, VOB and restrict the related words not to be pronouns and verbs.
- **Frequency Restriction**: After calculating word frequencies, only word frequency that greater than 2 is valid for generating question.
- **Question Restriction**: Only five questions can be extracted within one passage.

3.4. Human Annotation

Apart from the automatically generated large-scale training data, we also provide human-annotated validation and test data to improve the estimation quality. The annotation procedure costs one month with 5 annotators and each question is cross-validated by another annotator. The detailed procedure for each type of dataset can be illustrated as follows.

[5]Full descriptions of abbreviations can be found at `http://www.ltp-cloud.com/intro/en/#dp_how`.

3.4.1. Cloze-style Reading Comprehension

For the validation and test set in cloze data, we first randomly choose 5,000 paragraphs each for automatically generating questions using the techniques mentioned above. Then we invite our resource team to manually select 2,000 questions based on the following rules.

- Whether the question is appropriate and correct
- Whether the question is hard for LMs to answer
- Only select one question for each paragraph

3.4.2. User Query Reading Comprehension

Unlike the cloze dataset, we have no automatic question generation procedure in this type. In the user query dataset, we asked our annotator to directly raise questions according to the passage, which is much difficult and time-consuming than just selecting automatically generated questions. We also assign 5,000 paragraphs for question annotations in both validation and test data. Following rules are applied in asking questions.

- The paragraph should be read carefully and judged whether appropriate for asking questions
- No more than 5 questions for each passage
- The answer should be better in the type of nouns, named entities to be fully evaluated
- Too long or too short paragraphs should be skipped

4. Experiments

In this section, we will give several baseline systems for evaluating our datasets as well as presenting several top-ranked systems in the competition.

Rank	System	Single Model		Ensemble	
		Validation	Test	Validation	Test
-	Baseline - Random Guess	1.65	1.67	-	-
-	Baseline - Top Frequency	14.85	14.07	-	-
-	Baseline - AS Reader (default settings)	76.05	77.67	-	-
-	Baseline - AoA Reader (default settings)	**77.20**	**78.63**	-	-
1	6ESTATES	75.85	74.73	**81.85**	**81.90**
2	Shanghai Jiao Tong Univeristy BCMI-NLP	76.15	**77.73**	78.35	80.67
3	XinkTech	77.15	77.53	79.20	80.27
4	East China Normal University (ECNU)	**77.95**	77.40	79.45	79.70
5	Ludong University	74.75	75.07	77.05	77.07
6	Wuhan University (WHU)	78.20	76.53	-	-
7	Harbin Institute of Technology at Shenzhen (HITSZ)	76.05	75.93	-	-
8	HuoYan Technology	73.55	75.77	-	-
9	Wuhan University of Science and Technology (WUST)	73.80	74.53	-	-
10	Beijing Information Science and Technology University	70.05	70.20	-	-
11	Shanxi Univerisity (SXU-2)	62.60	64.70	66.65	68.47
12	Shenyang Aerospace University (SAU)	63.15	65.80	-	-
13	Shanxi University (SXU-1)	64.85	64.67	-	-
14	Zhengzhou Univerisity (ZZU)	52.80	54.53	-	-

Table 2: Results on Cloze Track. The best baseline and participant systems are depicted in bold face.

4.1. Baseline Systems

We set several baseline systems for testing basic performance of our datasets and provide meaningful comparisons to the participant systems. In this paper, we provide four baseline systems, including two simple ones and two neural network models. The details of the baseline systems are illustrated as follows.

- **Random Guess**: In this baseline, we randomly choose one word in the document as the answer.
- **Top Frequency**: We choose the most frequent word in the document as the answer.
- **AS Reader**: We implemented Attention Sum Reader (AS Reader) (Kadlec et al., 2016) for modeling document and query and predicting the answer with the Pointer Network (Vinyals et al., 2015), which is a popular framework for cloze-style reading comprehension. Apart from setting embedding and hidden layer size as 256, we did not change other hyper-parameters and experimental setups as used in Kadlec et al. (2016), nor we tuned the system for further improvements.
- **AoA Reader**: We also implemented Attention-over-Attention Reader (AoA Reader) (Cui et al., 2017) which is the state-of-the-art model for cloze-style reading comprehension. We follow hyper-parameter settings in AS Reader baseline without further tuning.

In the User Query Track, as there is a gap between training and validation, we follow (Liu et al., 2017) and regard this task as domain adaptation or transfer learning problem. The neural baselines are built by the following steps.

System	Validation	Test
Baseline - Random Guess	1.50	1.47
Baseline - Top Frequency	10.65	8.73
Baseline - AS Reader	-	49.03
Baseline - AoA Reader	-	**51.53**
ECNU (Ensemble)	90.45	**69.53**
ECNU (single model)	85.55	65.77
Shanxi University (Team-3)	47.80	49.07
Zhengzhou University	31.10	32.53

Table 3: Results on User Query Track. Due to the using of validation data, we did not report its performance.

- We first use the shared training data to build a general systems, and choose the best performing model (in terms of cloze validation set) as baseline.
- Use User Query validation data for further tuning the systems with 10-fold cross-validations.
- Increase dropout rate (Srivastava et al., 2014) to 0.5 for preventing over-fitting issue.

All baseline systems are chosen according to the performance of the validation set.

4.2. Participant Systems

The participant system results[6] are given in Table 2 and 3.

4.2.1. Cloze Track

As we can see that two neural baselines are competitive among participant systems and AoA Reader successfully

[6]Full CMRC 2017 Leaderboard: `http://www.hfl-tek.com/cmrc2017/leaderboard.html`.

outperform AS Reader and all participant systems in single model condition, which proves that it is a strong baseline system even without further fine-tuning procedure. Also, the best performing single model among participant systems failed to win in the ensemble condition, which suggest that choosing right ensemble method is essential in most of the competitions and should be carefully studied for further performance improvements.

4.2.2. User Query Track

Not surprisingly, we only received three participant systems in User Query Track, as it is much difficult than Cloze Track. As shown in Table 3, the test set performance is significantly lower than that of Cloze Track, due to the mismatch between training and test data. The baseline results give competitive performance among three participants, while failed to outperform the best single model by ECNU, which suggest that there is much room for tuning and using more complex methods for domain adaptation.

5. Conclusion

In this paper, we propose a new Chinese reading comprehension dataset for the 1st Evaluation on Chinese Machine Reading Comprehension (CMRC-2017), consisting large-scale automatically generated training set and human-annotated validation and test set. Many participants have verified their algorithms on this dataset and tested on the hidden test set for final evaluation. The experimental results show that the neural baselines are tough to beat and there is still much room for using complicated transfer learning method to better solve the User Query Task. We hope the release of the full dataset (including hidden test set) could help the participants have a better knowledge of their systems and encourage more researchers to do experiments on.

6. Acknowledgements

We would like to thank the anonymous reviewers for their thorough reviewing and providing thoughtful comments to improve our paper. We thank the Sixteenth China National Conference on Computational Linguistics (CCL 2017) and Nanjing Normal University for providing space for evaluation workshop. Also we want to thank our resource team for annotating and verifying evaluation data. This work was supported by the National 863 Leading Technology Research Project via grant 2015AA015409.

7. Bibliographical References

Che, W., Li, Z., and Liu, T. (2010). Ltp: A chinese language technology platform. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, pages 13–16. Association for Computational Linguistics.

Chen, D., Bolton, J., and Manning, D. C. (2016). A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2358–2367. Association for Computational Linguistics.

Cui, Y., Liu, T., Chen, Z., Wang, S., and Hu, G. (2016). Consensus attention-based neural networks for chinese reading comprehension. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1777–1786. The COLING 2016 Organizing Committee.

Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., and Hu, G. (2017). Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 593–602. Association for Computational Linguistics.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1684–1692.

Hill, F., Bordes, A., Chopra, S., and Weston, J. (2015). The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

Kadlec, R., Schmid, M., Bajgar, O., and Kleindienst, J. (2016). Text understanding with the attention sum reader network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 908–918. Association for Computational Linguistics.

Liu, T., Cui, Y., Yin, Q., Zhang, W.-N., Wang, S., and Hu, G. (2017). Generating and exploiting large-scale pseudo training data for zero pronoun resolution. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 102–111. Association for Computational Linguistics.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Vinyals, O., Fortunato, M., and Jaitly, N. (2015). Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700.

A Multi-Domain Framework for Textual Similarity. A Case Study on Question-to-Question and Question-Answering Similarity Tasks

Amir Hazem Basma El Amal Boussaha Nicolas Hernandez
Laboratoire des Sciences du Numérique de Nantes (LS2N)
Université de Nantes, 44322 Nantes Cedex 3, France
{Amir.Hazem,Basma.Boussaha,Nicolas.Hernandez}@univ-nantes.fr

Abstract

Community Question Answering (CQA) websites have become a very popular and useful source of information, which helps users to find out answers to their corresponding questions. On one hand, if a user's question does not exist in the forum, a new post is created so that other users can contribute and provide answers or comments. On the other hand, if similar or related questions already exist in the forum, the system should be able to detect them and redirect the user towards the corresponding threads. This procedure of detecting similar questions is also known as question-to-question similarity task in the NLP research community. Once the correct posts have been detected, it is important to provide the correct answer since some posts can contain tens or hundreds of answers/comments which make the user's research more difficult. This procedure is also known as the question-answering similarity task. In this paper, we address both tasks and aim at providing the first framework on the evaluation of similar questions and question-answering detection on a multi-domain corpora. For that purpose, we use the community question answering forum Stack-Exchange to extract posts and pairs of questions and answers from multiple domains. We evaluate two baseline approaches over 19 domains and provide preliminary results on multiple annotated question-answering datasets to deal with question-answering similarity task.

Keywords: Community question answering, textual similarity, word embeddings

1. Introduction

The increasing popularity of question answering websites has led to the emergence of a new area of research called community question answering, which has to deal with two distinct but complementary tasks. The first task, called question-to-question similarity, has to provide related questions to a given original question. The identification of similar question pairs aims at preventing duplicate posts in the forums and to redirect users towards posts that might contain an appropriate answer. The second task, called question-answering similarity, aims at providing a correct answer to a given original question. If several users contribute to a given post, it is important to automatically extract the correct answers among tens or hundreds of answers since a manual exploration becomes hard to achieve. These tasks offer a key challenge while they have to deal with textual similarity not only in terms of lexical similarity but also in terms of reformulation, paraphrasing, duplicates and near duplicates, textual entailment, semantics, etc.

Over the past years, there have been several studies on community question answering (Qiu and Huang, 2015; Filice et al., 2016; Barrón-Cedeño et al., 2016; Franco-Salvador et al., 2016; Nakov et al., 2016; Nakov et al., 2017; Patra, 2017), etc. Most of them addressed this task through specific datasets such as the programming Q&A website Stackoverflow [1], Quora dataset for duplicate extraction [2], Yahoo!Answers dataset (Qiu and Huang, 2015), Qatar living corpus via SemEval shared task [3], etc. Also, in most of the evaluations, the candidates of a given question are often limited in number (around 10 per question). For instance, in the Qatar Living corpus of SemEval which has to deal with expatriates questions, the question-to-question similarity task consists of reranking 10 related questions given an original question, while the question answering similarity task consists of reranking 10 answers given an original question. If these two tasks on the Qatar Living corpus are interesting to address, the evaluation procedure is not realistic while we have to deal with only 10 candidates for each question. In a real scenario, if a new question is posted in a forum, the system should compare this question to all the questions that have been already posted. The aim of this paper is to provide multi-domain datasets with a more realistic evaluation workflow since the candidates are not limited in number but concerns the entire set of the forum questions. Hence, based on StackExchange datasets, we provide 19 corpora of several domains ranging from politics, economics, history, philosophy to music, sport, travel, cooking, etc. We evaluate two baselines on the question-to-question and question-answering similarity tasks. The first baseline is a sentence similarity approach based on word embeddings ($SentEmb$) and the second approach ($MappSent$) is a textual similarity approach based on a mapping matrix. Recently, $MappSent$ approach (Hazem et al., 2017) obtained better results than the winner system of 2016 and 2017 SemEval sharedtask editions on the question-to-question similarity task. By providing a large coverage of datasets and a more realistic evaluation procedure, we hope that this work serves as a cornerstone for future evaluations on question-to-question and question-answering similarity tasks. On the short term, we also aim at enriching the framework with the entire StackExchange datasets which consists of about 180 corpora.

The remainder of this paper is organized as follows. Section 2. describes the different linguistic resources used in our experiments. Section 3. describes the state of art ap-

[1] https://stackoverflow.com/
[2] https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs
[3] http://alt.qcri.org/semeval2017/task3/

proaches. The experimental setup and the obtained results are respectively presented in Sections 4. and 5. Section 6. discusses the approaches behaviour and their obtained results and finally, Section 7. concludes this work.

2. StackExchange Datasets

In this section, we present the processed datasets that have been extracted from StackExchange community question answering resource as well as some statistics about original and related questions.

2.1. StackExchange CQA

StackExchange is a multi-domain community question answering framework which contains topics in varied fields. Its purpose is to enable users to post questions and to answer or comment[4] them. A voting system is also available and users can vote for good answers and comments. The more votes an answer has, more likely it is to be an appropriate answer to a given question. To build the 19 StackExchange datasets, we based our selection on the users voting scores as well as the tag *Answer* provided by the metadata to select answers with regards to given questions. For each question, we select the answer with the highest voting score. We made this strong hypothesis to select correct pairs of questions and answers. If this hypothesis can naturally be discussed and criticized, we did an extensive manual verification and we found a strong correlation between users score votes and correct answers. However the voting score can be adjusted to a certain threshold that guarantees reliable answers[5]. The next section gives several statistics about the extracted datasets.

2.2. Datasets Statistics

Table 1 enumerates the 19 extracted datasets that can be found in our git-hub repository[6]. The first column represents the size of the datasets in terms of number of tokens. The second column shows the number of posts for each corpus and the third column represents the number of posts of a filtered version of each corpus. In most of the cases, the subject field of the StackExchange resource corresponds to a question, and the body field to an expanded version of the question (i.e. a question closely related to the one mentioned in the subject field, with a context which provides more details). We define the questions in the subject field as the *original questions* and the content of the body field as the *related questions*. Since in some cases the post's subjects contain keywords and not questions, the filtering process aims at selecting the posts for which the subject is a question. We ensure this requirement by only selecting posts where the subject field contains a question mark. The fourth and final column of the table shows the size of the test sets in terms on number of pairs of original and related questions and their corresponding answers.

[4]We didn't consider the comments in our datasets. We let this kind of posts for the future.

[5]This score depends on the topic. We fixed a minimum voting score of 5 and discarded all posts with lower voting score.

[6]`https://github.com/hazemAmir/StackExchange`

Corpus	#token	#all posts	#filtered posts	#test
Earth Science	221K	2.2k	1.6k	169
Expatriates	185k	2.6k	1.3k	137
Health	276k	2.9k	2.2k	223
Sports	264K	3.2k	2.3k	240
Politics	415K	3.2k	2.8k	282
Pets	373k	3.4k	2.5k	253
Economics	333k	4.1k	2.1k	210
Law	609k	5.1k	3.6k	365
History	741k	6.1k	5.2k	522
Philosophy	1.1M	7.3k	5.7k	575
Music	701k	9.1k	5.4k	544
Workplace	1.7M	12.9k	8.2k	830
Biology	1.1M	14.1k	10k	1001
Cooking	1.2M	16k	11.3k	1132
Chemistry	1.3M	18.5k	10.4k	1042
Travel	1.6M	20.6k	12.9k	1297
Physics	7.02M	87.2k	44.4k	4443
AskUbuntu	11.1M	248k	79.1k	7912
Math	28.8M	702k	168k	16820

Table 1: Size of the multi-domain datasets in terms of number of tokens (column 1), number of posts (column 2), number of filtered posts (column 3) and number of test questions (column 4).

The main information conveyed by Table 1 is the diversity of the datasets in terms of topics and size. The smallest datasets contain about 200k tokens and about 2k posts before filtering such as: *Earth Science* and *Expatriates* corpora, while the largest datasets such as: *Physics*, *Math*, *AskUbuntu* for instance, range from 1M to 50M tokens and thousands of posts. The multiple characteristics of these datasets (size, topics, etc.) may offer a better way to evaluate approaches and systems performance.

Table 2 gives some statistics about the size of the original/related questions and the answers in addition to their ratio. The first and second columns show the mean length[7] of the original and related questions while the third column shows the mean length of the answers. Finally, the two last columns show the mean ratio between original and related questions[8](column 4) and the ratio between original questions and the answers (column 5). We observe that the mean average length of the original questions is short ranging from 11 to 16 tokens, while the mean average length of the related questions is much larger ranging from 123 to 246 tokens. With no surprise, the mean average length of the answers is often much larger than the questions and ranges from 167 to 367. We also observe that the mean ratio is very small which shows that the related

[7]The number of tokens of the original question.

[8]The closer to 1 is the ratio, most similar are the original and related questions in terms of number of tokens.

Corpus	#OriQ	#RelQ	#Ans	# ratioQ	# ratioA
Earth Science	12	150	323	0.13	0.05
Expatriates	15	152	228	0.15	0.07
Health	12	142	259	0.13	0.03
Sports	12	124	268	0.16	0.07
Politics	13	169	367	0.13	0.05
Pets	12	171	293	0.11	0.06
Economics	13	183	239	0.13	0.05
Law	14	193	246	0.12	0.07
History	13	163	346	0.13	0.06
Philosophy	12	237	352	0.09	0.05
Music	12	147	250	0.12	0.08
Workplace	14	246	246	0.08	0.08
Biology	11	136	282	0.13	0.04
Cooking	11	123	191	0.14	0.1
Chemistry	12	149	241	0.13	0.06
Travel	13	139	250	0.14	0.08
Physics	12	183	254	0.12	0.07
AskUbuntu	11	163	167	0.13	0.1
Math	16	197	197	0.13	0.12

Table 2: Statistics of the multi-domain datasets in terms of the mean number of tokens of the original question (Column 1), mean number of tokens of the related question (column 2), mean number of tokens of the answers (column 3), the mean ratio between the original and related questions (column 4) and the mean ratio between the original questions and the answers (column 5).

questions and answers contain much more information than the original questions. Hereafter an example of an original question and its related question extracted from the *pets* training corpus:

Original question (Subject):
what 's an appropriate diet for a small river turtle ?

Related question (Body):
i get my little brother a small river turtle, and i would like to know what it eat. it's somewhat relatively fast, longish slim tail, seem to be energetic, but i 'm not sure what it eat. Any idea?

Answer:
In general a turtle's diet is comprised of 50% protein, 30% vegetable, 10% green, and 10% fruit. The protein can even be divided up so that it's 25% commercial food, and 25 % fresh food. You should feed the turtle once daily since it's so young right now, but as it get old you can move to a schedule of every other day. Some people even go so low every third day; others keep feed them daily but in small amount. It's important to keep track of how much you're feeding your turtle because turtle is notorious beggar, and you can risk get your turtle fat if you give in to it demand too often. The rule of thumb for feed time is as much as the turtle can eat in a 15-minute time-span, or an amount of food that match the size of the turtle's head and neck.

From the above example we see that the related question contains more details about the question. We can also see that the answer is very productive with a lot of details. According to the statistics of Table 2, the major part of the question pairs and question-answering pairs follow the same schema that is: short original questions in the subject of the post and long related questions in the body and very long answers. In general, a manual observation of a sample of StackExchange datasets confirms these statistics. Also, we could see that the body of the posts contains much more details and context comparing to the question provided in the subject of the post. This particularity may have some inconvenience while modeling pairs of questions. In addition, the long answers with different information can act negatively to identify the question-answering pairs, since questions and answers are unbalanced in terms of content.
The first contribution of this paper is to provide a set of 19 multi-domain datasets. We provide four versions of the datasets: raw data, tokenized data, lemmatized data and pos-tagged data. The tokenization and pos-tagging are conducted using nltk[9]. The second contribution of this paper is a systematic evaluation of two textual similarity-based approaches (SentEmb and MappSent) on the 19 datasets for question-to-question and question-answering similarity tasks.

3. Baseline Approaches

In this Section we describe the two implemented baselines that is: (i) the sentence embedding approach (SentEmb) and (ii) the mapping approach (MappSent).

3.1. SentEmb

The sentence embedding approach consists of representing each question (original or related) and each answer by an embedding vector. The embedding vector is the sum of the vector embedding of each word of the question or answer (Mikolov et al., 2013b; Wieting et al., 2016; Arora et al., 2017; Hazem et al., 2017). Then, to extract similar pairs of questions or pairs of questions/answers, the cosine similarity is computed. The related questions (in the question-to-question similarity task) and the answers (in the question-answering similarity task) are ranked according to their scores regarding the original questions.
It is to note that each sentence (question or answer) is preprocessed[10]. We also remove stop-words and only keep nouns, verbs and adjectives. We also conducted experiments without the POS-TAG and stop-words filtering process but the results were lower.

3.2. MappSent

MappSent approach (Hazem et al., 2017) is an extension of SentEmb and aims at providing a better representation

[9] https://github.com/nltk/nltk
[10] Tokenization, part-of-speech tagging and lemmatization.

of pairs of similar sentences, paragraphs and more generally, pieces of texts of any length. A prior condition is to have a training dataset of pairs of similar sentences. The main idea is: given a set of similar sentences, the goal is to build a more discriminant and representative sentence embedding space. Word embeddings of the entire corpus are first computed, then, each sentence is represented by an element-wise addition of its word embedding vectors. Finally, a mapping matrix is built using the Singular Values Decomposition (SVD) to project sentences in a new subspace. Similar sentences are moved closer thanks to a mapping matrix (Artetxe et al., 2016) learned from a training dataset containing pairs of similar sentences. Basically, a set of similar sentence pairs is used as seed information to build the mapping matrix. The optimal mapping is computed by minimizing the Euclidean distance between the seed sentence pairs.
MappSent approach consists of the following steps:

1. We train a Skip-gram [11] model using Gensim toolkit[12] on a lemmatized training dataset.

2. Each training and test sentence is pre-processed. We remove stop-words and only keep nouns, verbs and adjectives while computing sentence embedding vectors and the mapping matrix. This step is not applied when learning word embeddings (cf. Step 1).

3. For each given pre-processed sentence, we build its embedding vector which is the element-wise addition of its words embedding vectors (Mikolov et al., 2013a; Wieting et al., 2016; Arora et al., 2017). Unlike Arora et al. (2017) we did not use any weighting procedure while computing vectors embedding sum[13].

4. We build a mapping matrix where test sentences can be projected. We adapted Artetxe et al. (2016) approach in a monolingual scenario as follows:

   - To build the mapping matrix we need a mapping dictionary which contains similar sentence pairs.
   - The mapping matrix is built by learning a linear transformation which minimizes the sum of squared Euclidean distances for the dictionary entries and using an orthogonality constraint to preserve the length normalization.
   - While in the bilingual scenario, source words are projected in the target space by using the bilingual mapping matrix, in our case, original and related questions/answers are both projected in a similar subspace using the monolingual sentence mapping matrix. This consists of our adaptation of the bilingual mapping.

5. Test sentences are projected in the new subspace thanks to the mapping matrix.

6. The cosine similarity is then used to measure the similarity between the projected test sentences.

4. Experimental Setup

To evaluate the quality of the different approaches, we use in all the experiments the mean average precision *MAP* (Manning et al., 2008).

$$MAP = \frac{1}{|W|} \sum_{i=1}^{|W|} \frac{1}{Rank_i} \quad (1)$$

where $|W|$ corresponds to the size of the question-to-question and question-answering evaluation lists, and $Rank_i$ corresponds to the ranking of a correct question/answer candidate i.
For word embeddings, we used as settings a window size of 10 words, negative sampling of 5, sampling of 1e-3 and training over 15 iterations. We applied both Skip-gram and CBOW models[14] to create vectors of dimensions of 100 and 300. We used hierarchical softmax for training the Skip-gram model. We only report the results of the Skip-gram model as it has shown the best results on our development datasets.

5. Results

We present in this section the preliminary results on the question-to-question and question-answering similarity tasks over the 19 datasets of the two baselines $SentEmb$ and $MappSent$.
Table 3 shows the results of $SentEmb$ and $MappSent$ on the question-to-question similarity task for the development and test sets. We observe that the results vary according to the domain and the size of the datasets. Better results are obtained when data size is small, for instance: Earth Science, Expatriates, etc. while results drop for large datasets such as AskUbuntu or Math. Overall, MappSent almost always outperforms SentEmb in both the development and test sets.
Table 4 shows the results of $SentEmb$ and $MappSent$ on the question-answering similarity task for the development and test sets. We observe that the results are much lower than the question-to-question similarity task. This may be an indicator about the difficulty of identifying question-answering pairs. Also, $MappSent$ outperforms $SentEmb$ with a huge gap. Regarding the results, $sentEmb$ seems not appropriate to question/answering pairs identification.

6. Discussion

The first purpose of this paper was to provide a more realistic multi-topic datasets to evaluate systems performance on textual similarity tasks. More specifically, we targeted question-to-question and question answering similarity tasks which represent a key challenge in community

[11] CBOW model had also been experienced but it turned out to give lower results while compared to the Skip-gram model.

[12] To ensure the comparability of our experiments, we fixed the python hash function that is used to generate random initialization. By doing so, we are sure to obtain the same embeddings for a given configuration.

[13] We explored this direction without success.

[14] To train word embedding models we used the gensim toolkit (Rehurek and Sojka, 2010).

Corpus	*SentEmb*		*MappSent*	
	Dev	Test	Dev	Test
Earth Science	**67.7**	68.2	67.4	**73.1**
Expatriates	65.7	68.2	**68.8**	**71.1**
Health	45.8	66.3	**46.5**	**66.8**
Sports	62.7	57.5	**64.3**	**59.7**
Politics	70.4	70.3	**73.3**	**72.0**
Pets	60.9	61.6	**63.7**	**63.2**
Economics	66.2	61.0	**67.2**	**61.4**
Law	60.7	**71.1**	**62.1**	70.8
History	51.7	60.3	**52.9**	**62.7**
Philosophy	35.9	40.8	**40.3**	**44.6**
Music	46.9	44.1	**49.2**	**45.9**
Workplace	40.4	39.6	**43.1**	**41.5**
Biology	50.0	37.4	**52.8**	**38.9**
Cooking	54.0	50.7	**57.1**	**53.2**
Chemistry	37.0	41.2	**38.9**	**43.6**
Travel	53.9	53.8	**56.6**	**57.2**
Physics	37.1	32.4	**40.1**	**34.5**
AskUbuntu	13.6	18.5	**14.7**	**19.8**
Math	6.23	5.83	**6.71**	**6.13**

Table 3: Results (MAP%) of SentEmb and MappSent on the question-to-question similarity task using 19 Q/Q datasets.

Corpus	*SentEmb*		*MappSent*	
	Dev	Test	Dev	Test
Earth Science	16.9	9.01	**41.4**	**41.2**
Expatriates	7.02	5.35	**26.0**	**25.9**
Health	4.63	4.25	**24.4**	**22.4**
Sports	10.0	11.4	**42.2**	**33.5**
Politics	8.09	6.60	**32.4**	**36.1**
Pets	9.64	9.66	**27.3**	**33.2**
Economics	9.29	4.44	**32.5**	**27.7**
Law	6.24	5.89	**26.7**	**25.2**
History	7.45	8.47	**33.0**	**33.4**
Philosophy	6.03	5.44	**26.1**	**22.1**
Music	12.1	11.4	**25.7**	**27.7**
Workplace	9.71	**16.1**	**14.3**	13.4
Biology	2.32	1.85	**28.1**	**27.8**
Cooking	9.66	3.31	**25.9**	**27.1**
Chemistry	2.01	2.97	**17.0**	**18.1**
Travel	3.75	5.37	**23.4**	**24.8**
Physics	1.07	1.17	**13.7**	**14.1**
AskUbuntu	0.45	0.29	**4.88**	**5.58**
Math	0.21	0.12	**4.07**	**6.43**

Table 4: Results (MAP%) of SentEmb and MappSent on the Question-Answering similarity task using 19 Q/A datasets.

question answering. We chose StackExchange as it offers varied topics and also metadata annotations that allow a better selection of posts according to users voting system. We provide the first version of 19 raw and pre-processed datasets of various topics. These datasets will be gradually extended and enriched in the near future to provide the 180 datasets contained in StackExchange.

The second purpose of this paper was to evaluate two baselines to have an overview of their performance over the multi-topic framework. We could see that the performance depends on the size of the datasets and on the topics. Also, according to the results, the question-answering similarity task seems to be more difficult than the question-to-question similarity task. We could see that a simple cosine similarity between embedding vectors of questions and answers (SentEmb approach) is not appropriate for the question answering task. This might be obvious while answers does not contain only lexical similarities with there corresponding questions. However, using a mapping matrix to learn embedding regularities has shown interesting results (MappSent approach). Even if we can't state that MappSent captures rhetorical and dependency relations between question answering pairs, it seems that it captures types of relations that allow a better performance.

7. Conclusion

This work provides the first multi-topic community question answering environment for the evaluation of question-answering similarity. We make available 19 question-to-question and question-answering similarity datasets. All the corpora were extracted from the community question answering forum StackExchange. We also evaluated two baseline methods on these corpora which we hope will serve as a basis for future evaluations on these tasks. For future work, we will gradually enrich this resource with the remaining datasets of StackExchange and the final goal is to process the entire community question answering framework for an extensive multi-topic textual similarity evaluation .

8. Acknowledgments

The current work was supported by the ANR 2016 PASTEL (ANR-16-CE33-0007) project[15].

9. Bibliographical References

Arora, S., Yingyu, L., and Tengyu, M. (2017). A simple but tough to beat baseline for sentence embeddings. In *Proceedings of the 17th International Conference on Learning Representations (ICLR'17)*, pages 1–11.

[15] http://www.agence-nationale-recherche.fr/?Projet=ANR-16-CE33-0007.

Artetxe, M., Labaka, G., and Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP'16)*, pages 2289–2294, Austin, TX, USA.

Barrón-Cedeño, A., Da San Martino, G., Joty, S., Moschitti, A., Al-Obaidli, F., Romeo, S., Tymoshenko, K., and Uva, A. (2016). Convkn at semeval-2016 task 3: Answer and question selection for question answering on arabic and english fora. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 896–903, San Diego, California, June. Association for Computational Linguistics.

Filice, S., Croce, D., Moschitti, A., and Basili, R. (2016). Kelp at semeval-2016 task 3: Learning semantic relations between questions and answers. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1116–1123, San Diego, California, June. Association for Computational Linguistics.

Franco-Salvador, M., Kar, S., Solorio, T., and Rosso, P. (2016). UH-PRHLT at semeval-2016 task 3: Combining lexical and semantic-based features for community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 814–821.

Hazem, A., el amel Boussaha, B., and Hernandez, N. (2017). Mappsent: a textual mapping approach for question-to-question similarity. Recent Advances in Natural Language Processing, RANLP 2017, 2-8 September, 2017, Varna, Bulgaria.

Manning, D. C., Raghavan, P., and Schütze, H. (2008). Introduction to information retrieval. Cambridge University Press.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013a). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Mikolov, T., Yih, S. W.-t., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2013)*. Association for Computational Linguistics, May.

Nakov, P., Màrquez, L., Moschitti, A., Magdy, W., Mubarak, H., Freihat, A. A., Glass, J., and Randeree, B. (2016). SemEval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June. Association for Computational Linguistics.

Nakov, P., Hoogeveen, D., Màrquez, L., Moschitti, A., Mubarak, H., Baldwin, T., and Verspoor, K. (2017). SemEval-2017 task 3: Community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, SemEval '17, Vancouver, Canada, August. Association for Computational Linguistics.

Patra, B. (2017). A survey of community question answering. *CoRR*, abs/1705.04009.

Qiu, X. and Huang, X. (2015). Convolutional neural tensor network architecture for community-based question answering. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 1305–1311. AAAI Press.

Rehurek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2016). Towards universal paraphrastic sentence embeddings. *Internationa Conference on Learning Representations, CoRR*, abs/1511.08198.

WorldTree: A Corpus of Explanation Graphs for Elementary Science Questions supporting Multi-Hop Inference

Peter A. Jansen*, Elizabeth Wainwright†, Steven Marmorstein‡, Clayton T. Morrison*
*School of Information, †Department of Linguistics, ‡Department of Computer Science
University of Arizona, Tucson, USA
pajansen@email.arizona.edu

Abstract

Developing methods of automated inference that are able to provide users with compelling human-readable justifications for why the answer to a question is correct is critical for domains such as science and medicine, where user trust and detecting costly errors are limiting factors to adoption. One of the central barriers to training question answering models on explainable inference tasks is the lack of gold explanations to serve as training data. In this paper we present a corpus of explanations for standardized science exams, a recent challenge task for question answering. We manually construct a corpus of detailed explanations for nearly all publicly available standardized elementary science question (approximately 1,680 3^{rd} through 5^{th} grade questions) and represent these as "explanation graphs" – sets of lexically overlapping sentences that describe how to arrive at the correct answer to a question through a combination of domain and world knowledge. We also provide an explanation-centered tablestore, a collection of semi-structured tables that contain the knowledge to construct these elementary science explanations. Together, these two knowledge resources map out a substantial portion of the knowledge required for answering and explaining elementary science exams, and provide both structured and free-text training data for the explainable inference task.

Keywords: question answering, explanations, explainable inference

1. Introduction

Question answering (QA) is a high-level natural language processing task that requires automatically providing answers to natural language questions. The approaches used to construct QA solvers vary depending on the questions and domain, from inference methods that attempt to construct answers from semantic, syntactic, or logical decompositions, to retrieval methods that work to identify passages of text likely to contain the answer in large corpora using statistical methods. Because of the difficulty of this task, overall QA task performance tends to be low, with generally between 20% and 80% of natural (non-artificially generated) questions answered correctly, depending on the questions, the domain, and the knowledge and inference requirements.

Standardized science exams have recently been proposed as a challenge task for question answering (Clark, 2015), as these questions have very challenging knowledge and inference requirements (Clark et al., 2013; Jansen et al., 2016), but are expressed in simple-enough language that the linguistic challenges are likely surmountable in the near-term. They also provide a standardized comparison of modern inference techniques against human performance, with individual QA solvers generally answering between 40% to 50% of multiple choice science questions correctly (Khot et al., 2015; Clark et al., 2016; Khashabi et al., 2016; Khot et al., 2017; Jansen et al., 2017, inter alia), and top-performing ensemble models nearly reaching a passing grade of 60% on middle school (8^{th} grade) science exams during a recent worldwide competition of 780 teams sponsored by the Allen Institute for AI (Schoenick et al., 2017).

One of the central shortcomings of question answering models is that while solvers are steadily increasing the proportion of questions they answer correctly, most solvers generally lack the capacity to provide human-readable explanations or justifications for why those answers are correct. This "explainable inference" task is seen as a limitation of current machine learning models in general (e.g. Ribeiro et al., (2016)), but is critical for domains such as science or medicine where user trust and detecting potentially costly errors are important. More than this, evidence from the cognitive and pedagogy literature suggests that explanations (when tutoring others) and self-explanations (when engaged in self-directed learning) are an important aspect of learning, helping humans better generalize the knowledge they have learned (Roscoe and Chi, 2007; Legare, 2014; Rittle-Johnson and Loehr, 2016). This suggests that explainable methods of inference may not only be desirable for users, but may be a requirement for automated systems to have human-like generalization and inference capabilities.

Building QA solvers that generate explanations for their answers is a challenging task, requiring a number of inference capacities. Central among these is the idea of *information aggregation*, or the idea that explanations for a given question are rarely found in a contiguous passage of text, and as such inference methods must generally assemble many separate pieces of knowledge from different sources in order to arrive at a correct answer. Previous estimates (Jansen et al., 2016) suggest elementary science questions require an average of 4 pieces of knowledge to answer and explain those answers (here our analysis suggests this is closer to 6), but inference methods tend to have difficulty aggregating more than 2 pieces of knowledge from free-text together due to the semantic or contextual "drift" associated with this aggregation (Fried et al., 2015). Because of the difficulty in assembling training data for the information aggregation task, some have

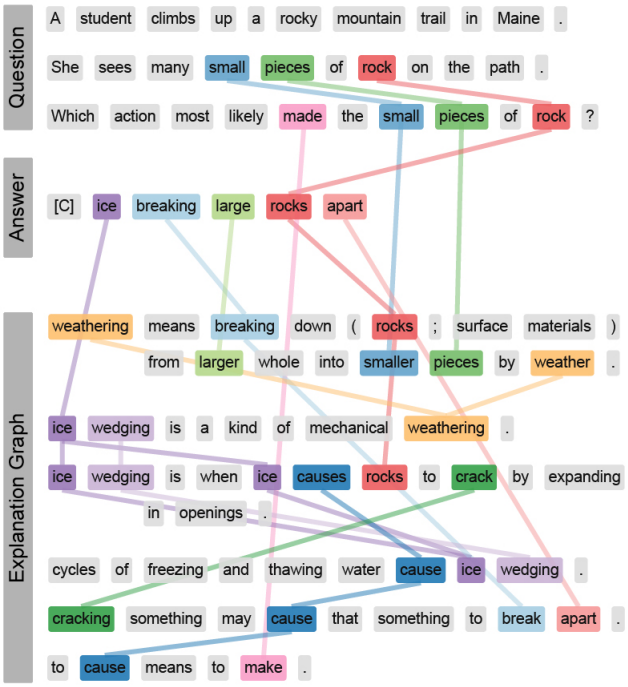


Figure 1: An example multiple choice science question, the correct answer, and a sample explanation graph for why that answer is correct. Here, the explanation graph consists of six sentences, each interconnected through lexical overlap with the question, answer, and other explanation sentences.

approached explanation generation as a distant supervision problem, with explanation quality modelled as a latent variable (Jansen et al., 2017; Sharp et al., 2017). While these techniques have had some success in constructing short explanations, semantic drift likely limits the viability of this technique for explanations requiring more than two pieces of information to be aggregated.

To address this, here we construct a large corpus of explanation graphs (see Figure 1) to serve as training data for explainable inference tasks. The contributions of this work are are:

- We construct a set of explanations for 1,680 standardized elementary science exam questions, represented as both free-text, and as lexically-overlapping "explanation graphs" that provide training data for inference models by detailing explicit connections between knowledge in different sentences of an explanation.

- We provide an explanation-centered "tablestore", a set of 62 semi-structured tables containing 4,950 rows that provide a substantial portion of the knowledge required to answer non-spatial, non-mathematical elementary science questions.

- We provide an analysis of the knowledge growth and explanation overlap properties of this corpus, suggesting both requirements for inference algorithms to make use of explanation corpora, as well as methods of estimating the difficulty in constructing explanation corpora in other domains.

2. Related Work

In terms of question answering, the ability to provide compelling human-readable explanations for answers to questions has been proposed as a complementary metric to assess QA performance alongside the proportion of questions answered correctly. Jansen et al. (2017) developed a QA system for elementary science that answers questions by building and ranking explanation graphs built from aggregating multiple sentences read from free text corpora, including study guides and dictionaries. Because of the difficulty in constructing gold explanations to serve as training data, the explanations built with this system were constructed by modeling explanation quality as a latent variable machine learning problem. First, sentences were decomposed into sentence graphs based on clausal and prepositional boundaries, then assembled into multi-sentence "explanation graphs". Questions were answered by ranking these candidate explanation graphs, using answer correctness as well as features that capture the connectivity of key-terms in the graphs as a proxy for explanation quality. Jansen at al. (2017) showed that it is possible to learn to generate high quality explanations for 60% of elementary science questions using this method, an increase of 15% over a baseline that retrieved single continuous passages of text as answer justifications. Critically, in their error analysis Jansen et al. found that for questions answered incorrectly by their system, nearly half had successfully generated high-quality explanation graphs and ranked these highly, though they were not ultimately selected. They suggest that the process of building and ranking explanations would be aided by developing more expensive second-pass reranking processes that are able to better recognize the components and structure of high-quality explanations within a short list of candidates.

Knowledge bases of tables, or *"table stores"*, have recently been proposed as a semi-structured knowledge formalism for question answering that balances the cost of manually crafting highly-structured knowledge bases with the difficulties in acquiring this knowledge from free text (Yin et al., 2015; Sun et al., 2016; Jauhar et al., 2016). The methods for question answering over tables generally take the form of constructing chains of multiple table rows that lead from terms in the question to terms in the answer, while the tables themselves are generally either collected from the web, automatically generated by extracting relations from free text, or manually constructed.

At the collection end of the spectrum, Pasupat and Liang (2015) extract 2,108 HTML tables from Wikipedia, and propose a method of answering these questions by reasoning over the tables using formal logic. They also introduce the WikiTableQuestions dataset, a set of 22,033 question-answer pairs (such as *"Greece held its last Summer Olympics during which year?"*) that can be answered using these tables. Demonstrating the ability for collection at scale, Sun et al. (2016) extract a total of 104 million tables from Wikipedia and the web, and develop a model that constructs relational chains between table rows using a deep-learning framework.[1] Using their system and ta-

[1] Sun et al. (2016) note that the 99 million tables extracted from

ble store, Sun et al. demonstrate state-of-the-art performance on several benchmark datasets, including WebQuestions (Berant et al., 2013), a set of popular questions asked from the web designed to be answerable using the large structured knowledge graph Freebase (e.g. *"What movies does Morgan Freeman star in?"*).

In terms of automatic generation, though relations are often represented as $< subject, relation, argument >$ triples, Yin et al. (2015) create a large table containing 120M *n-tuple* relations using OpenIE (Etzioni et al., 2011), arguing that the extra expressivity afforded by these more detailed relations allows their system to answer more complex questions. Yin et al. use this to successfully reason over the WebQuestions dataset, as well as their own set of questions with more complex prepositional and adverbial constraints.

Elementary science exams contain a variety of complex and challenging inference problems (Clark et al., 2013; Jansen et al., 2016), with nearly 70% of questions requiring some form of causal, process, or model-based reasoning to solve and produce an explanation for. In spite of these exams being taken by millions of students each year, elementary students tend not to be fast or voluminous readers by adult standards, making this a surprisingly low-resource domain for grade-appropriate study guides and other materials. The questions also tend to require world knowledge expressed in grade-appropriate language (like that *bears have fur* and that *fur keeps animals warm*) to solve. Because of these requirements and limitations, table stores for elementary science QA tend to be manually or semi-automatically constructed, and comparatively small.

Khashabi et al. (2016) provide the largest elementary science table store to date, containing approximately 5,000 manually-authored rows across 65 tables based on science curriculum topics obtained from study guides and a small corpus of questions. Khashabi et al. also augment their tablestore with 4 tables containing 2,600 automatically generated table rows using OpenIE triples. Reasoning is accomplished using an integer-linear programming algorithm to chain table rows, with Khashabi et al. reporting that an average of 2 table rows are used to answer each question. Evaluation on a small set of 129 science questions achieved passing performance (61%), with an ablation study showing that the bulk of their model's performance was from the manually authored tables.

To help improve the quality of automatically generated tables, Dalvi et al. (2016) introduce an interactive tool for semi-automatic table generation that allows annotators to query patterns over large corpora. They demonstrate that this tool can improve the speed of knowledge generation by up to a factor of 4 over manual methods, while increasing the precision and utility of the tables up to seven fold compared to completely automatic methods.

All of the above systems share the commonality that they work to connect (or aggregate) multiple pieces of knowledge that, through a variety of inference methods, move towards the goal of answering questions. Fried et al. (2015) report that information aggregation for QA is currently very challenging, with few methods able to combine more than two pieces of knowledge before succumbing to *semantic drift*, or the phenomenon of two pieces of knowledge being erroneously connected due to shared lexical overlap, incomplete word-sense disambiguation, or other noisy signals (e.g. erroneously aggregating a sentence about *Apple computers* to an inference when working to determine whether *apples* are a kind of fruit). In a generating a corpus of natural-language explanations for 432 elementary science questions, Jansen et al. (2016) found that the average question requires aggregating 4 separate pieces of knowledge to explainably answer, with some questions requiring much longer explanations.

Though few QA solvers explicitly report the aggregation limits of their algorithms, Fried et al. (2015), Khabashi et al. (2016) and Jansen et al. (2017) appear to show limits or substantial decreases in performance after aggregating two pieces of knowledge. To the best of our knowledge, of systems that use information aggregation, only Jansen et al. (2017) explicitly rate the explanatory performance of the justifications from their model, with good explanations generated for only 60% of correctly answered questions. Taken together, all of this suggests that performance on information aggregation and explainable question answering is still far from human performance, and could substantially benefit from a large corpus of training data for these tasks.

3. Design Goals

We began with the following design goals:

Computable explanations: Explanations should be represented at different levels of structure (explanation, then sentences, then relations within sentences). The knowledge links between explanation sentences should be explicit through lexical overlap, which can be used to form an "explanation graph" that describes how each sentence is linked in an explanation.

Depth: Sufficient knowledge should be present in explanations such that that the answer could be arrived at with little extra domain or world knowledge – i.e. where possible, explanations should be targeted at the level of knowledge of a 5-year old child, or lower (see below for a more detailed discussion of explanatory depth).

Reuse: Where possible, knowledge should be re-used across explanations to facilitate automated analysis of knowledge use, and identifying common explanation patterns across questions.

3.1. Explanation Depth

The level of knowledge required to convincingly explain why an answer to a question is correct depends upon one's familiarity with the domain of the question. For a domain expert (such as an elementary science teacher), a convincing explanation to why *thick bark* is the correct answer to *"Which characteristic could best help a tree survive the heat of a forest fire?"* might need only take the form of explaining that *one of bark's primary functions is to provide protection for the tree*. In contrast, for a domain novice, such as an elementary science student, this explanation

the web introduce more noise into the inference process than the high-quality tables from Wikipedia

Question	Which of the following characteristics would best help a tree survive the heat of a forest fire?	
Answers	[A] large leaves	[B] shallow roots
	*[*C] thick bark*	[D] thin trunks

Levels of explanatory knowledge:

Domain Expert (e.g. teacher)
Bark is a protective covering around the trunk and branches of a tree.

Domain Novice (e.g. 4th grade student)
As an object's thickness increases, it's resistance to damage will also increase.

Young child (e.g. 5-year old)
Protecting something means preventing harm.
Fire causes harm to trees, forests, and other living things.
Thickness is a measure of how thick an object is.
A tree is a kind of living thing.

First Principles
Protecting a living thing has a positive impact on it's survival and health.

Table 1: Levels of explanatory knowledge depth in order of increasing specificity, and example explanatory sentences for each level. For a domain expert who is already fluent in the reasoning of a domain, brief explanations may be sufficient to completely understand why a given answer is correct. As the level of explanatory knowledge moves towards increasing specificity, less domain and world knowledge is assumed, and this knowledge must be explicitly included in the explanations. Explanatory levels are additive, i.e. an explanation targeted at the *young child* level would also include the knowledge at the *domain novice* and *domain expert* levels. In this work, we target authoring explanations at a level between *young child* and *first principles*.

might need to be elaborated to include more knowledge to make this inference, such as that *thicker things tend to provide more protection*. Here we identify four coarse levels of increasing explanatory knowledge depth, shown in Table 1.

For training explainable inference systems, a high level of explanatory depth is likely required. As such, in this work we target authoring explanations between the levels of *young child* and *first principles*. Pragmatically, in spite of their ultimate utility for training inference systems, building explanations too close to *first principles* becomes laborious and challenging for annotators given the level of abstraction and the large amount of implicit world knowledge that must be enumerated, and we leave developing protocols and methods for building such detailed explanations for future work.

4. Explanation Authoring

We describe our representations, tools, and annotation process below.

4.1. Questions

We author explanation graphs for a corpus of 2,201 elementary science questions (3^{rd} through 5^{th} grade) from the AI2 Science Questions V2 corpus, consisting of both standardized exam questions from 12 US states, as well as the separate AI2 Science Questions Mercury dataset, a set of questions licensed from a student assessment entity. Each question is a 4-way multiple choice question, and only those questions that do not involve diagram interpretation (a separate spatial task) are included. Approximately 20% of explanations required specialized domain knowledge (for example, spatial or mathematical knowledge) that did not easily lend itself to explanation using our formalism, resulting in a corpus of 1,680 questions and explanations.

4.2. Tables and Table Rows

Explanations for a given question consist of a set of sentences, each of which is on a single topic and centered around a particular kind of relation, such as *water is a kind of liquid (a taxonomic relation)*, or *melting means changing from a solid to a liquid through the addition of heat energy (a change relation)*.

Each explanation sentence is represented as a single row from a semi-structured table defined around a particular relation. Our tablestore includes 62 such tables, each centered around a particular relation such as *taxonomy, meronymy, causality, changes, actions, requirements,* or *affordances*, and a number of tables specified around specific properties, such as *average lifespans of living things, the magnetic properties of materials,* or the *nominal durations of certain processes (like the Earth orbiting the Sun)*. The initial selection of table relations was drawn from a list of 21 common relations required for science explanations identified by Jansen et al. (2016) on a smaller corpus, and expanded as new knowledge types were identified. Subsets of example tables are included in Figure 2. Each explanation in this corpus contains an average of 6.3 rows.

Fine-grained column structure: In tabular representations, columns represent specific roles or arguments to a specific relation (such as *X is when Y changes from A to B using mechanism C*). In our tablestore we attempt to minimize the amount of information per cell, instead favouring tables with many columns that explicitly identify common roles, conditions, or other relations. This finer-grained structure eases the annotator's cognitive load when authoring new rows, while also better compartmentalizing the relational knowledge in each row for inference algorithms. The tables in our tablestore contain between *2 and 16* content columns, as compared to 2 to 5 columns for the Ariso tablestore (Khashabi et al., 2016).

Natural language sentences: QA models use a variety of different representations for inference, from semantic roles and syntactic dependencies to discourse and embeddings. Following Khashabi et al. (2016), we make use of a specific form of table representation that includes "filler" columns that allow each row to be directly read off as a stand-alone natural language sentence, and serve as input to any model. Examples of these filler columns can be seen in Figure 2.

4.3. Explanation Graphs and Sentence Roles

Explanations for a given question here take the form of a list of sentences, where each sentence is a reference to a specific table row in the table store. To increase their utility for knowledge and inference analyses, we require that

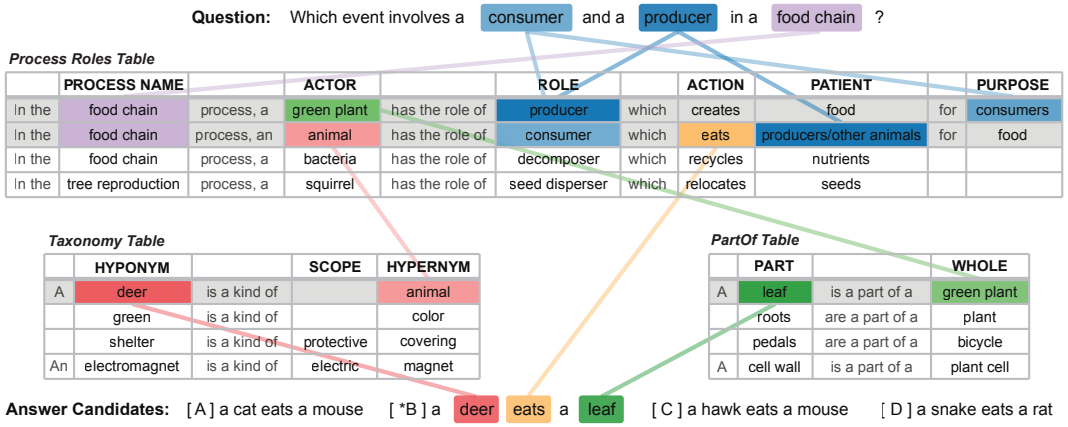


Figure 2: Examples of tables and table rows from the tablestore, grounded in an example question and explanation. Table columns define the primary roles or arguments for a given relation (e.g. *process name, actor, role, etc*). Unlabeled "filler" columns allow each row to be used as a stand-alone natural language sentence. Note that for clarity only 4 example rows per table are shown.[3]

Question	Which occurs as the kinetic energy of water molecules increases?
Answer	[*D] liquid water becomes water vapor
Central role	
	As a molecule's kinetic energy increases, temperature will increase.
	Boiling means changing from a liquid into a gas by adding heat energy.
Grounding role	
	Water is a kind of liquid.
	Water is in the gas state, called water vapor, for temperatures greater than 100 degrees celsius.
Background role	
	Matter is made of molecules.
Lexical glue role	
	To add means to increase.
	Temperature is a measure of heat energy.

Table 2: Examples of the four coarse classes of explanation sentence roles, *central, grounding, background, and lexical glue*.

each sentence in an explanation be explicitly lexically connected (i.e. share words) with either the question, answer, or other sentences in the explanation. We call this lexically-connected set of sentences an *explanation graph*.

In our preliminary analysis, we observed that the sentences in our explanations can take on very different roles, and we hypothesize that differentiating these roles is likely important for inference algorithms. We identified four coarse roles, listed in Table 2, and described below:

- **Central:** The central concept(s) that a question is testing, such as *changes of state* or the *coupled relationship between kinetic energy and temperature*.
- **Grounding:** Sentences linking generic or abstract terms in a *central* sentence with specific instances of those terms in the question or answer. For example, for questions about changes of state, grounding sentences might identify specific instances of *liquids* (such as water) or *gasses* (such as water vapor).
- **Background:** Extra information elaborating on the topic, but that (strictly speaking) isn't required to arrive at the correct inference.
- **Lexical glue:** Sentences that lexically link two concepts, such as *"to add means to increase"*, or *"heating means adding heat"*. This is an artificial category in our corpus, brought about by the need for explanation graphs to be explicitly lexically linked.

For each sentence in each authored explanation, we provide annotation indicating which of these four roles the sentence serves in that explanation.

4.4. Annotation Tool

To facilitate explanation authoring, we developed and iterated the web-based collaborative authoring tool shown in Figure 3. The tool displays a given question to the explanation author, and allows the author to progressively build an explanation graph for that question by querying the tablestore for relevant rows based on keyword searches, as well as past explanations that are likely to contain similar content or structure (increasing consistency across explanations, while reducing annotation time). A graphical visualization of the explanation graph helps the author quickly assess gaps in the explanation content to address by highlighting lexical overlap between sentences with coloured edges and labels. The tablestore takes the form of a shared Google Sheet[4] that the annotators populate, with each table represented as a separate tab on the sheet.

4.5. Procedure and Explanation Review

For a given question, annotators identified the central concept the question was testing, as well as the inference required to correctly answer the question, then began progressively constructing the explanation graph. Sentences in the graph were added by querying the tablestore based on keywords, which retrieved both single sentences/table rows, as well as entire explanations that had been previously annotated. If any knowledge required to build an explanation did

[3] Note that this figure also appears in an earlier workshop submission on identifying explanatory patterns (Jansen, 2017)

[4] http://sheets.google.com

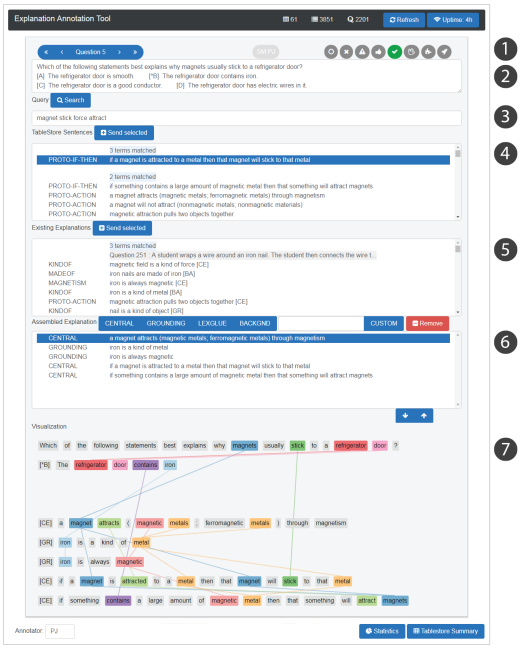


Figure 3: The explanation authoring web tool. Interface components include: (1) A list of user-settable flags to assist in the annotation and quality review process; (2) Question and answer candidates; (3) Query terms for search; (4) Query results (tablestore); (5) Query results (complete explanations); (6) Current explanation being assembled; (7) Explanation graph visualization of lexical overlap within the explanation.

not exist in the tablestore, this was added to an appropriate table, then added to the explanation.

New tables were regularly added, most commonly for property knowledge surrounding a particular topic (e.g. whether a particular material is recyclable). Because explanations are stored as lists of unique identifiers to table rows, tables and table rows could regularly be refactored, elaborated, or entirely reorganized without requiring existing explanations to be rewritten. We found this was critical for consistency and ensuring good organization throughout corpus construction.

One of the central difficulties with evaluating explanation authoring is determining metrics for interannotator agreement, as many correct explanations are possible for a given question, and there are many different wordings that an annotator might choose to express a given piece of knowledge in the tablestore. Similarly, the borders between different levels of explanatory depth are fuzzy, suggesting that one annotator may express their explanation with more or less specificity than another.

To address these difficulties we included two methods to increase consistency. First, as a passive intervention during the explanation generation process, annotators are presented with existing explanations that can be drawn from to compose a new explanation, where these existing explanations share many of the same query terms being used to construct the new explanation. Second, as an active intervention, each explanation goes through four review passes to ensure consistency. The first two passes are completed by the original annotator, before checking a flag on the annotation tool signifying that the question is ready for external review. A second annotator then checks the question for completeness and consistency with existing explanations, and composes a list of suggested edits and revisions. The fourth and final pass is completed by the original annotator, who implements these suggested revisions. This review process is expensive, taking approximately one third of the total time required to annotate each question.

Each annotator required approximately 60 hours of initial training for this explanation authoring task. We found that most explanations could be constructed within 5-10 minutes, with the review process taking approximately 5 more minutes per question.

5. Explanation Corpus Properties

Here we characterize three properties of the explanation corpus as they relate to developing methods of explainable inference: *knowledge frequency, explanation overlap,* and *tablestore growth.*

5.1. Knowledge Use and Row Frequency

The tables most frequently used to author explanations are shown in Table 3, broken down into three broad categories identified by Jansen et al. (2016): *retrieval types, inference-supporting types, and complex inference types*. Because the design of this corpus is data driven – i.e., knowledge is generally added to a table because it is required in one or more explanations[5] – we can calculate how frequently the rows in a given table are reused to obtain an approximate measure of the generality of that knowledge. On average, a given table row is used in 2.9 different explanations, with 1,535 rows used more than once, and 531 rows used 5 or more times. The most frequently reused row (*"an animal is a kind of organism"*) is used in 89 different explanations. Generic "change of state" knowledge (e.g. solids, liquids, and gasses) is also frequently reused, with each row in the StatesOfMatter table used in an average of 15.7 explanations. Usage statistics for other common tables are also provided in Table 3.

5.2. Explanation Overlap

One might hypothesize that questions that require similar inferences to correctly answer may also contain some of the same knowledge in their explanations, with the amount of knowledge overlap dependent upon the similarity of the questions. We plan to explore using this overlap as a method of inference that can generate new explanations by editing, merging, or expanding known explanations from similar, known questions (see Jansen (2017) for an initial study). For this to be possible, an explanation corpus must reach a sufficient size that a large majority of questions have substantial overlap in their explanations.

Figure 5 shows the proportion of questions in the corpus that have *1 or more*, *2 or more*, *3 or more*, *etc.*, overlapping rows in their explanations with *at least one other*

[5]For compatibility, we do include several property tables from the Aristo tablestore, though a large proportion of rows from these tables are not actively used. Our tablestore includes 4,950 rows, 3,686 of which are actively used in at least one explanation.

Knowledge Type	Prevalence (% of expl.)	Rows in Table	Avg. Row Freq.
Retrieval Types			
Taxonomic	78%	1,119	1.2
Synonymy	61%	639	1.6
PartOf	14%	148	1.6
Properties (Generic)	11%	173	1.1
MadeOf	7%	72	1.7
Contains	6%	75	1.4
Examples	5%	58	1.4
Measurements (P)	4%	23	3.0
Locations (P)	3%	47	1.1
InheritedTraits (P)	3%	22	2.3
StatesOfMatter (P)	3%	3	15.7
Conductivity (P)	3%	9	4.9
Resources (P)	3%	16	2.7
Inference Supporting Types			
Actions	25%	259	1.6
UsedFor	19%	191	1.7
Requires	15%	121	2.1
SourceOf	14%	81	2.8
Affect	12%	77	2.6
Opposites	8%	35	3.8
FormedBy	4%	40	1.9
Affordances	4%	48	1.3
Complex Inference Types			
If/Then	21%	229	1.6
Cause	17%	183	1.6
Changes (discrete)	14%	62	3.8
Transfer	9%	46	3.3
Changes (vector)	9%	62	2.4
CoupledRelationships	7%	126	0.9
ProcessRoles	3%	12	3.8

Table 3: The proportion of explanations that contain knowledge from a given table, sorted by most frequent knowledge, and broken down by the knowledge type of a given table. Tables not used in at least 3% of explanations are not shown. (P) indicates a given table describes properties, e.g. whether a given material is conductive. Average Row Frequency refers to the average number of explanations a given row from that table is used in.

question in the corpus.[6] Similarly, to ground this, Figure 4 shows a visualization of questions whose explanations have *2 or more* overlapping rows. For a given level of overlapping explanation sentences, Figure 5 shows that the proportion of questions with that level of overlap increases logarithmically with the number of questions.

This has two consequences. First, it allows us to estimate the size of corpus required to train hypothetical inference methods for the science exam domain capable of producing explanations. If a given inference method can work successfully with only minimal overlap (for example, 1 shared table row), then a training corpus of 500 explanations in this domain should be sufficient to answer 80% of questions. If an inference method requires 2 shared rows, the corpus requirements would increase to approximately 2,500 questions to answer 80% of questions. However, if an inference method requires 3 or more rows, this likely would not be possible without a corpus of at least 20,000 questions and explanations – a substantial undertaking. Second, because this relationship is strongly logarithmic, if it transfers to domains outside elementary science, it should be possible to estimate the corpus size requirements for those domains after authoring explanations for only a few hundred questions.

[6]Though not included for space, the number of questions with *N or more* rows in common in their explanations increases linearly with the number of questions. For this corpus, for a given question, on average there are 17 questions that have 1 or more overlapping rows in their explanation, 9 questions with 2 or more shared rows in their explanation, and 5 questions with 3 or more shared rows in their explanation.

5.3. Explanation Tablestore Growth

Finally, we examine the growth of the tablestore as it relates to the number of questions in the corpus. Figure 6 shows a monte-carlo simulation of the number of unique tablestore rows required to author explanations for specific corpus sizes. This relationship is strongly correlated *(R=0.99)* with an exponential proportional decrease.[7] For this elementary science corpus, this asymptotes at approximately 6,000 unique table rows, and 10,000 questions, providing an estimate of the upper-bound of knowledge required in this domain, and the number of unique questions that can be generated within the scope of the elementary science curriculum.

The caveat to this estimate is that it estimates the knowledge required for elementary science exams as they currently exist, with the natural level of variation introduced by the test designers. Questions are naturally grounded in examples, such as *"Which part of an oak tree is responsible for undertaking photosynthesis?" (Answer: the leaves)*. While the corpus often contains a number of variations of a given question that test the same curriculum topic and have similar explanations, many more variations on these questions are possible that ground the question in different examples, like *orchids*, *peach trees*, or other plants. As such, while we believe that these estimates likely cover the core knowledge of the domain, many times that knowledge would be required to make the explanation tablestore robust to small variations in the presentation of those existing exam questions, or to novel unseen questions.

6. Conclusion

We provide a corpus of explanation graphs for elementary science questions suitable for work in developing explainable methods of inference, and show that the knowledge frequency, explanation overlap, and tablestore growth properties of the corpus follow predictable relationships. This work is open source, with the corpus and generation tools available at `http://www.cognitiveai.org/explanationbank`.

7. Acknowledgements

We thank the Allen Institute of Artificial Intelligence for funding this work, Peter Clark at AI2 for thoughtful discussions, and Paul Hein for assistance constructing the annotation tool.

[7]Here, this exponential proportional decrease takes the form of $R = 434 - (-2.93/0.00054) \cdot (1 - e^{-0.00054 \cdot Q})$, where R is the size of the tablestore in rows, to explainably answer Q questions.



Figure 4: Questions in this explanation corpus connected by explanation overlap. Here, nodes represent questions and their explanations, and edges between nodes represent two questions having *at least 2 or more (i.e. 2+)* shared rows (i.e. sentences) in their explanations, with at least one of these shared rows being labelled as having a CENTRAL role to the explanation. Topic clusters (labels) naturally emerge for questions requiring similar methods of inference, based on the shared content of their explanations.

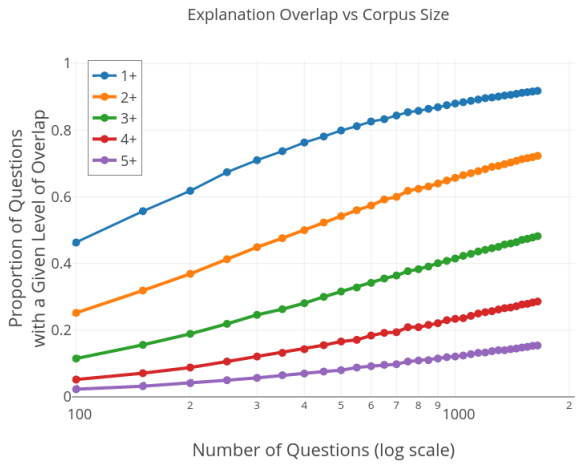


Figure 5: Monte-carlo simulation showing the proportion of questions whose explanations overlap by *1 or more, 2 or more, 3 or more, ...,* explanation sentences. The proportion increases logarithmically with the number of questions in the corpus. Each point represents the average of 100 simulations.

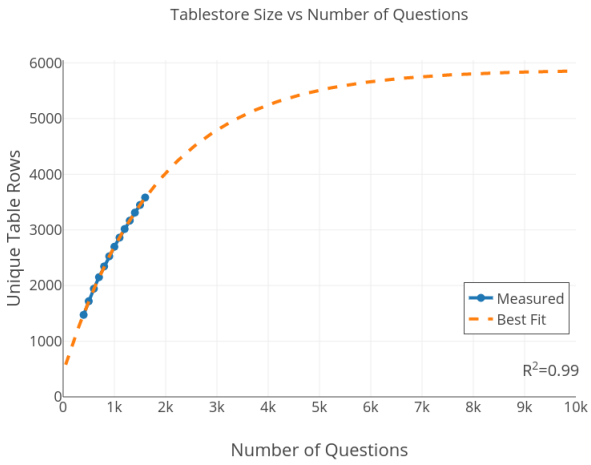


Figure 6: Monte-carlo simulation showing the number of unique table rows required to explainably answer a given number of questions. The line of best fit (dashed) suggests that this is a proportional decay relationship ($R^2 = 0.99$), asymptoting at approximately 6,000 table rows and 10,000 questions. Each point represents the average of 10,000 simulations.

8. Bibliographical References

Berant, J., Chou, A., Frostig, R., and Liang, P. (2013). Semantic parsing on freebase from question-answer pairs. In *EMNLP*.

Clark, P., Harrison, P., and Balasubramanian, N. (2013). A study of the knowledge base requirements for passing an elementary science test. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*, AKBC'13, pages 37–42.

Clark, P., Etzioni, O., Khot, T., Sabharwal, A., Tafjord, O., Turney, P. D., and Khashabi, D. (2016). Combining retrieval, statistics, and inference to answer elementary science questions. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 2580–2586.

Clark, P. (2015). Elementary school science and math tests as a driver for AI: take the aristo challenge! In Blai Bonet et al., editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 4019–4021. AAAI Press.

Dalvi, B., Bhakthavatsalam, S., Clark, C., Clark, P., Etzioni, O., Fader, A., and Groeneveld, D. (2016). IKE - an interactive tool for knowledge extraction. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, AKBC@NAACL-HLT 2016, San Diego, CA, USA, June 17, 2016*, pages 12–17.

Etzioni, O., Fader, A., Christensen, J., Soderland, S., and Mausam, M. (2011). Open information extraction: The second generation. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (ICJAI)*, pages 3–10.

Fried, D., Jansen, P., Hahn-Powell, G., Surdeanu, M., and Clark, P. (2015). Higher-order lexical semantic models for non-factoid answer reranking. *Transactions of the Association for Computational Linguistics*, 3:197–210.

Jansen, P., Balasubramanian, N., Surdeanu, M., and Clark, P. (2016). What's in an explanation? characterizing knowledge and inference requirements for elementary science exams. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2956–2965, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Jansen, P., Sharp, R., Surdeanu, M., and Clark, P. (2017). Framing qa as building and ranking intersentence answer justifications. *Computational Linguistics*.

Jansen, P. (2017). A study of automatically acquiring explanatory inference patterns from corpora of explanations: Lessons from elementary science exams. In *Proceedings of the 2017 Workshop on Automated Knowledge Base Construction*, AKBC'17.

Jauhar, S. K., Turney, P. D., and Hovy, E. H. (2016). Tables as semi-structured knowledge for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Khashabi, D., Khot, T., Sabharwal, A., Clark, P., Etzioni, O., and Roth, D. (2016). Question answering via integer programming over semi-structured knowledge. In *Proceedings of the International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 1145–1152.

Khot, T., Balasubramanian, N., Gribkoff, E., Sabharwal, A., Clark, P., and Etzioni, O. (2015). Exploring markov logic networks for question answering. In *EMNLP*.

Khot, T., Sabharwal, A., and Clark, P. (2017). Answering complex questions using open information extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 311–316.

Legare, C. H. (2014). The contributions of explanation and exploration to children's scientific reasoning. *Child Development Perspectives*, 8(2):101–106.

Pasupat, P. and Liang, P. (2015). Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 1135–1144, New York, NY, USA. ACM.

Rittle-Johnson, B. and Loehr, A. M. (2016). Eliciting explanations: Constraints on when self-explanation aids learning. *Psychonomic bulletin & review*, pages 1–10.

Roscoe, R. D. and Chi, M. T. (2007). Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors' explanations and questions. *Review of Educational Research*, 77(4):534–574.

Schoenick, C., Clark, P., Tafjord, O., Turney, P., and Etzioni, O. (2017). Moving beyond the turing test with the allen ai science challenge. *Communications of the ACM*, 60(9):60–64.

Sharp, R., Surdeanu, M., Jansen, P., Valenzuela-EscÃ¡rcega, M. A., Clark, P., and Hammond, M. (2017). Tell me why: Using question answering as distant supervision for answer justification. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*.

Sun, H., Ma, H., He, X., Yih, W.-t., Su, Y., and Yan, X. (2016). Table cell search for question answering. In *Proceedings of the 25th International Conference on World Wide Web (WWW)*, pages 771–782.

Yin, P., Duan, N., Kao, B., Bao, J., and Zhou, M. (2015). Answering questions with complex semantic constraints on open knowledge bases. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1301–1310. ACM.

Analysis of Implicit Conditions in Database Search Dialogues

Shun-ya Fukunaga Hitoshi Nishikawa Takenobu Tokunaga
Tokyo Institute of Technology
fukunaga.s.ab@m.titech.ac.jp, {hitoshi, take}@c.titech.ac.jp

Hikaru Yokono Tetsuro Takahashi
Fujitsu Laboratories Ltd.
{yokono.hikaru, takahashi.tet}@jp.fujitsu.com

Abstract

This paper reports an annotation to a corpus of database search dialogues on real estate, and the analysis on implicit information in the utterances for constructing database queries. Two annotators annotated 50 dialogues with a set of database field tags, resulting in a high inter-annotator agreement (Cohen's κ=0.79), and the analysis revealed that 10% of the utterances included non-database-field information. We further investigated these utterances to find that more than 93% of them included useful information for figuring out search conditions, which we call the *implicit condition*. The contribution of this paper is to present the existence and importance of the implicit conditions in the database search dialogues and both qualitative and quantitative analysis of them. Our corpus can provide a fundamental language resource for constructing a dialogue system which can utilise the implicit conditions. The paper concluded with possible approaches to achieve our long-term goal, extracting the implicit conditions in the database search dialogues and utilising them to construct queries.

Keywords: database search dialogue, implicit condition, corpus annotation

1. Introduction

The information that a dialogue system should extract from user utterances highly depends its backend application. When being used as a natural language interface of a database system, the dialogue system should be able to extract pieces of information corresponding to the record fields of the database for constructing a query. There have been several attempts to extract this type of information from user utterances in database search dialogues. For instance, several studies (Raymond and Riccardi, 2007; Mesnil et al., 2015; Liu and Lane, 2016) tried to extract values for the database field defined in the ATIS (The Air Travel Information System) corpus (Hemphill et al., 1990; Dahl et al., 1994) from the user utterances. The ATIS corpus includes a set of dialogues between users and an air travel system that were collected through the Wizard-of-Oz method. The tags corresponding to the backend database fields (e.g. departure city, arrival date, etc.) were annotated to the expressions in the user utterances. However, the utterances in real dialogues include information that does not always directly correspond to a database field but provides useful information for constructing database queries. It will be more efficient and natural if the dialogue system can utilise this type of information for retrieving the database. For instance, in real estate search dialogues, which is our target domain, the number of family members provides useful information for deciding the size of a house, but it is rarely a field of the real estate database. The number of family members is an attribute of the customer rather than that of houses. We call this type of information the *implicit condition*.

Our long-term goal is to establish a method that is capable of extracting implicit conditions from the user utterances in the database search dialogue and utilising them to construct queries. As the first step in achieving this goal, this paper reports a corpus construction in which we annotated user utterances with a set of fields of a real estate database. We further analysed the utterances that were not assigned a database field and found that the majority of those utterances included the implicit conditions.

2. Related Work

There have been several attempts to annotate semantic information in database search dialogues. He and Young (2005) annotated the domain-specific lexical classes (e.g. "city name", "airport name") and concepts (e.g. ARRIVE, FROMLOC) in the ATIS corpus. The annotated tags were derived from the ATIS database schema. They considered only information corresponding to the database fields. The implicit conditions are out of their scope.

Asri et al. (2017) annotated a frame structure to each utterance in dialogues for searching package tours. A frame consists of a dialogue act and a set of slot-value pairs representing the content of the dialogue act. The most slots correspond to the database field, but some do not. These non-database-field slots mainly describe relations to other frames. In this respect, they are meta-level information than the same level information as the filed-slot information; therefore they are different from the implicit condition.

Dinarelli et al. (2009) annotated a semantic frame of FrameNet (Baker, 2012) in task-oriented dialogues. Since the FrameNet frames are designed for general purpose, the annotated frame are not necessarily useful for the database search task. In contrast, we annotated the information that is relevant for constructing database queries regardless whether it corresponds to the database field or not.

The above three studies annotated only predefined tags in the dialogues, i.e. they did not explore the information that does not fit into the predefined tags even though it is useful for the task. In contrary, Mitsuda et al. (2017) explored what kind of information can be extracted and also should be extracted from a chat-oriented dialogue system. We follow the same line, but we focus on exploring the useful information for constructing queries in the database search dialogues even though it does not directly correspond to a database field.

The most similar annotation scheme to ours is the one of the TourSG corpus (Kim et al., 2015) used in the Dialog State Tracking Challenge (DSTC) 4 and 5 (Kim et al., 2016b; Kim et al., 2016a). In this corpus, dialogue states are annotated in the form of slot-value pairs to each segment consisting of multiple utterances in touristic domain dialogues. A special slot called "INFO" is prepared for the information in the segment that does not directly correspond to any predefined regular slots. Their scheme is similar to ours in this respect, i.e. considering the information not corresponding to predefined slots. However, they did not analyse the details of such information nor its relation to the predefined slots.

3. Target Corpus: A Real Estate Corpus

We annotate a Japanese dialogue corpus developed by Takahashi and Yokono (2017). The corpus includes 986 dialogues between pairs of crowd workers who play a real estate agent and their customer each. The dialogues were collected through a keyboard chat system. The goal of the dialogues is finding a house or an apartment room that fulfils the customer needs. The agent does not search in a real database but completes the dialogue when having acquired necessary information for search.

In a dialogue, a customer was assigned one of ten predefined profiles and was instructed to interact with the agent regarding their assigned profile. The customer profile was not open to the agent. An example profile looks like "You are moving to a new place from your current studio apartment to live with your long-standing boyfriend. On this occasion, you want to improve your cooking skills and thus you prefer a place equipped with an easy-to-use kitchen with a multi-burner range."

4. Annotation

4.1. Database Field Tags

Referring to the search conditions in the real estate search site SUUMO[1], we defined 38 database (DB) field tags as shown in Table 1. We adopted the search conditions of SUUMO as it is one of the established search sites for real estate in Japan. In addition to these 38 tags, we defined the Other tag that does not fit into any of the database fields, expecting that utterances with the Other tag contain the implicit conditions.

4.2. Annotation Guidelines

The annotator annotated the corpus following the guidelines below.

- The minimal annotation unit is an utterance, i.e. the annotator assigns the DB field tag(s) to a whole utterance instead of a part of the utterance. We expect that this scheme reduces annotator's load.
- The annotator assigns the DB field tag(s) to an utterance by referring to only the target utterance and its preceding utterances; the annotator cannot refer to the succeeding context.

[1] http://suumo.jp

Location	*Facilities*
- available_railway_lines	- room_facilities
- walking_distance_to_a_station	- air_conditioning
- nearest_station_facility	- storage
- zone	- bathroom
- surrounding_facilities	- kitchen
- land_characteristics	- TV_and_Internet
- distance_to_a_specific_place	*Property*
Building	- property_type
- building_age	- rent
- floor_plan	- price
- floor_area	- conditions_for_rent
- room_placement_in_the_building	- available_date
- room_size	- target_demographic
- building_structure	- status
- sunlight	- appearance
- building_facilities	- ownership
- security_system	- available_discount
- number_of_storeys	- subsidy
- number_of_households	- certificate
- renovation	- warranty

Table 1: Database (DB) field tags

- When the annotator cannot find any appropriate DB field tag for the utterance, they annotate the Other tag and describe its content in the free format.
- When an utterance mentions to multiple database fields, the multiple tags concatenated with "+" are assigned. When an utterance is ambiguous in DB field tags, the candidate tags concatenated with "/" are assigned.

Table 2 shows an example of an annotated dialogue, which is a translation of the original Japanese dialogue. The speaker (SP) "A" denotes the real estate agent and "C" denotes the customer. The content description for the Other tag is shown in the parentheses. In utterance 2, the customer states that they are looking for a room close to their workplace, and they live alone. Therefore the distance_to_a_specific_place tag and the Other tag are assigned to the utterance. The annotator has added "number of persons (to live in the room)" to the Other tag as its content. Since the utterance includes both tags, they are concatenated with "+". Utterance 10 mentions "one-bedroom apartment" (floor_plan), "rent" (rent) and "location". Since the expression "location" has two interpretations: surrounding_facilities and zone, they are conjoined by "/" and put together with other two tags with "+".

4.3. Agreement Analysis

Two annotators A and B annotated 50 dialogues in the corpus introduced in the previous section with the DB field tags defined in Table 1. Five dialogues were randomly selected from the dialogues of each customer profile to constitute the 50 dialogues of the annotation target. The total number of utterances is 1,305. As we focus on the content-level analysis of the dialogues, we excluded the utterances at the dialogue management level, e.g. greeting utterances as utterance 1 in Table 2, prompting utterances like "Do you have any other requests?" and concluding utterances at the end of the dialogue such as "Thanks for your coming

	SP	utterance	assigned tag (content of the Other tag if any)
1	A	What kind of property are you looking for?	Other(general)
2	C	I am looking for a room for a single life that is close to my workplace.	distance_to_a_specific_place+Other(number of persons)
3	A	Where is your workplace?	Other(workplace)
4	C	Ikebukuro.	Other(workplace)
5	A	How do you commute?	Other(transportation)
6	C	I prefer a walking distance place.	Other(transportation)
7	A	How much do you pay for a place within walking distance of Ikebukuro?	distance_to_a_specific_place+rent
8	C	Around 100,000 yen.	rent
9	A	Do you have any preference in the floor plan?	floor_plan
10	C	I prefer a one-bedroom apartment. I put importance on the location and rent.	floor_plan+(surrounding_facilities/zone)+rent
11	A	Do you need a parking lot?	building_facilities
12	C	Yes, I need a parking lot. I would like to have a bicycle parking space as well.	building_facilities
13	A	Do you have any particular request on facilities?	building_facilities/room_facilities
14	C	Since I come home late, I need a tight security system and prefer a place populous and bright even at night.	security_system+Other(living environment)

Table 2: Annotation example

annotator A		annotator B	
Other	176	Other	152
building_facilities	128	building_facilities	121
property_type	118	rent	107
rent	105	property_type	100
floor_type	94	floor_type	98

Table 3: Number of frequently assigned tags (top 5)

annotator A		annotator B	
number of persons	62	number of persons	56
living environment	38	transportation	14
resident type	8	purpose of the property	13
accessibility	8	family structure	13
transportation	7	ambient noise	12

Table 4: Content descriptions of the Other tag

	annotator A		annotator B	
	type	token	type	token
implicit conditions	21	165	11	145
customer's profile	17	114	9	121
vague requests	4	51	2	24
no relevant DB fields	4	11	2	7
potential DB fields	1	4	1	4
no relation to DB	3	7	1	3
total	25	176	13	152

Table 5: Classification of the content of the Other tags

to our agency." We used remaining 1,194 utterances for the following analysis. The average number of the target utterances in a dialogue is 23.9, and the average length of the utterances is 19.8 characters in Japanese.

Out of 1,194 utterances, 964 utterances (80.7%) were assigned completely the same tag(s) by both annotators; that measures 0.79 in Cohen's κ (Cohen, 1960). We have a quite high inter-annotator agreement, concluding that we have no serious problem in our annotation scheme. In remaining 230 utterances that were assigned different tags across the annotators, there were 77 cases (33.5%) where an annotator assigned the tags in Table 1 but the other assigned the Other tag to the utterance. As we will discuss in the next section, such inconsistency comes from the fact that there are many utterances with the Other tag containing information tightly related to the search condition, i.e. it is difficult to decide they should be assigned the DB field tags or the Other tag. In Table 2, for instance, annotator A assigned the Other tag with the description "living environment" to utterance 14 for the expression "a place populous and bright even at night". In contrary, annotator B assigned the surrounding_facilities tag to the same utterance for the same expression. In reality, this expression does not directly correspond to the available database fields, but annotator B thought it corresponded to the DB field tag. This is because we can infer necessary values for the surrounding_facilities field through further clarification exchanges if necessary.

To investigate this hypothesis quantitively, we extracted utterances such that they are assigned a single tag by both annotators and these tags are inconsistent; one is the Other tag, and the other is a DB field tag. Out of such 32 utterances, 24 DB field tags (75.0%) can be easily derived from the content description of the Other tag through inference based on one of the authors subjective judgement. The annotator who assigned a DB field tag to these utterances presumably used some inference to derive the DB field tag regardless consciously or unconsciously.

5. Discussion

Table 3 shows the five most frequently assigned tags by each annotator. We counted multiple assigned tags to an utterance individually. The annotators assigned 1,464 (annotator A) and 1,467 (annotator B) tags respectively. Table 3 shows that the most frequently assigned tag is the Other tag that accounts for 12.0% (annotator A) and 10.4% (annotator B), indicating that the amount of information that does not directly correspond to the database fields is not negligible.

Table 4 shows the five most frequently mentioned content of the Other tag. Although the annotators were instructed to be consistent in surface expressions of the content description, as they were described in the free format, the expressions across annotators might be different for semantically the same content. The total numbers of the content description in type by the annotators are 25 (annotator A) and 13 (annotator B).

The utterances with the Other tag can be classified into two groups regarding if they include useful information for constructing a database search query.

The first group, which we introduced as the *implicit condition* in section 1, includes the information that does not directly match the database field but provides a clue for the field information through some inference. Observing the content descriptions of the Other tags, we can further categorise implicit conditions into two subgroups: customer's profiles and vague requests on the property. For instance, a customer's utterance "We are planning to live with two of us." describes the number of persons to live in the property, which is categorised to the customer's profile. This information can provide useful information for deciding room_size and floor_plan because an agent can easily infer the number and the size of rooms required for living with two persons. Also, the customer's profile of "workplace" and "transportation" described by utterances from 3 to 6 in Table 2 is useful information for deciding zone and available_railway_lines. In the collected corpus, we observe that the agent often asks the customer's profile. This would be because the agent preferred a question that draws out the information for multiple search fields at a time instead of a sequence of questions asking individual search field values one by one.

The second subgroup of the implicit conditions concerns customer's vague requests, which are related to multiple database fields. For example, a customer's utterance "I prefer a safe neighbourhood.", which corresponds to "living environment" in Table 4, is categorised to this subgroup, since the safeness of the neighbourhood is decided by both zone and surrounding_facility. Also, a customer's utterance "I prefer a convenient place for transportation." is related to all database fields about transportation such as available_railway_lines, walking_distance_to_a_station and nearest_station_facility. Being triggered by these kinds of utterances, we might be able to infer the field values through further clarification exchanges.

Table 5 summarises the counts of the implicit conditions and its subgroups. It also shows the counts of utterances that provide no information for the database fields. The table indicates that the majority of the utterances with the Other tag includes the implicit condition, i.e. the useful information to construct database search queries but not having corresponding database fields. This suggests insufficiency of the database fields we adopted, but the further investigation reveals that this is not the case. We point out two reasons of difficulty in defining the implicit conditions as the database fields.

The first reason concerns the mismatch in the information type between user's needs and the database. The real estate database stores information of real properties, while the user's needs realised as queries are tightly related to their objectives of a search and their backgrounds, which include diverse user's attributes. For instance, a family structure affects many database fields such as zone, surrounding_facilities and room_size, but the family structure is defined by many attributes such as the number of family members, their gender, their age and individual working style. It is impractical to add these attributes constituting the family structure in the real estate database. There is a gap in type between user's attributes and the real estate attributes.

Related to the first one, the second reason is that mapping from user's needs to the conditions on the database fields involves the user's evaluation of the values of the database fields. For instance, the implicit condition "living environment", which is the second most frequent by annotator A in Table 4, can be fulfilled differently in terms of the field values of properties depending on the user. In other words, a good property for a user is not always relevant for other users. There is a gap in evaluation of the field values by individual users. These reasons can apply to the database systems in general not being restricted the real estate database. The implicit condition triggers inference for bridging these kinds of gaps. Our corpus can provide a fundamental language resource for constructing a dialogue system which can utilise the implicit conditions.

We also have a minor group of utterances that does not provide any information regarding the current database fields. However, a few of them can be a potential field of the database ("potential DB fields" in Table 5). We found one such case where the customer asked if a preview of the property was available. The database we used does not include this information, but it would be useful for customers and is an attribute of real properties. Thus it can be a field of the database.

6. Conclusion

This paper reported an annotation to a corpus of database search dialogues on the real estate, and the analysis on implicit information in the utterances for constructing database queries. Two annotators annotated 50 dialogues with a set of database field tags, resulting in a high inter-annotator agreement (Cohen's κ=0.79). The analysis revealed that 10 % of the utterances included the information that did not directly match the database fields. We further investigated these utterances to find that more than 93% of them included useful information for constructing search queries by providing the database filed with a certain value, which we call the *implicit condition*. To realise a versatile dialogue system for database search that can naturally interact with users, we need to establish a method to extract the database field information from the implicit condition utterances. Our corpus can provide a fundamental language resource for constructing such a dialogue system. We plan to make it available for scientific communities.

Toward implementation of dialogue systems which can utilise implicit conditions, we need to challenge at least the following three tasks.

(1) identification of the utterances including the implicit condition

(2) interpretation of the implicit condition in the utterances as the corresponding database fields and their values

(3) identification of the span in the utterance representing the implicit condition

We will tackle these tasks with the following approaches. For the task (1), we can construct a binary classifier for deciding if an utterance includes the implicit condition or not. The annotated corpus described in this paper is usable for the training data of the classifier.

For the task (2), we plan to annotate the corresponding database fields to the implicit conditions in the utterances manually. As the number of database fields is limited, we believe this manual annotation is feasible according to our preliminary annotation trial. Being trained on this additional annotation results, a classifier can be constructed which classifies the implicit condition to the database fields. As an implicit condition can correspond to multiple database fields, e.g. "number of persons" can be classified to both room_size and floor_plan, this classifier should be a multi-label classifier.

To confirm that the interpretation of the system is correct, it might need clarification to the user. For constructing a clarification utterance by the system, the surface expression corresponding to the implicit condition in the user utterance is useful. For instance, having obtained the number of persons in the family, the system interprets it as floor_plan and might confirm the interpretation by saying "Then, a four-room apartment is necessary for three persons, isn't it?". The task (3) is, therefore, important for natural interaction. For this task, we plan to use the machine learning techniques which can provide the rationales for the output (Lei et al., 2016). We aim at identifying the span in the utterance (rationale) for the interpretation of the implicit condition (output).

7. Bibliographical References

Asri, L. E., Schulz, H., Sharma, S., Zumer, J., Harris, J., Fine, E., Mehrotra, R., and Suleman, K. (2017). Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2017)*, pages 207–219.

Baker, C. F. (2012). FrameNet, current collaborations and future goals. *Language Resources and Evaluation*, 46(2):269–286.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Dahl, D. A., Bates, M., Brown, M., Fisher, W., Hunicke-Smith, K., Pallett, D., Pao, C., Rudnicky, A., and Shriberg, E. (1994). Expanding the scope of the ATIS task: The ATIS-3 corpus. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, pages 43–48.

Dinarelli, M., Quarteroni, S., Tonelli, S., Moschitti, A., and Riccardi, G. (2009). Annotating spoken dialogs: From speech segments to dialog acts and frame semantics. In *Proceedings of the 2Nd Workshop on Semantic Representation of Spoken Language*, SRSL '09, pages 34–41.

He, Y. and Young, S. (2005). Semantic processing using the hidden vector state model. *Computer speech & language*, 19(1):85–106.

Hemphill, C. T., Godfrey, J. J., and Doddington, G. R. (1990). The ATIS spoken language systems pilot corpus. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '90, pages 96–101.

Kim, S., D'Haro, L. F., Banchs, R. E., Williams, J. D., and Henderson, M., (2015). *Dialog State Tracking Challenge 4 Handbook v.3.0*. http://www.colips.org/workshop/dstc4/Handbook_DSTC4.pdf.

Kim, S., D'Haro, L. F., Banchs, R. E., Williams, J. D., and Henderson, M. (2016a). The fourth dialog state tracking challenge. In *Proceedings of the 7th International Workshop on Spoken Dialogue Systems (IWSDS 2016)*.

Kim, S., D'Haro, L. F., Banchs, R. E., Williams, J. D., Henderson, M., and Yoshino, K. (2016b). The fifth dialogu state tracking challenge. In *Proceedings of the 2016 IEEE Workshop on Spoken Language Technology (SLT)*.

Lei, T., Barzilay, R., and Jaakkola, T. (2016). Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 107–117.

Liu, B. and Lane, I. (2016). Joint online spoken language understanding and language modeling with recurrent neural networks. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2016)*, pages 22–30.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., and Zweig, G. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 23(3):530–539.

Mitsuda, K., Higashinaka, R., and Matsuo, Y. (2017). What information should a dialogue system understand?: Collection and analysis of perceived information in chat-oriented dialogue. In *Proceedings of the 8th International Workshop on Spoken Dialogue Systems (IWSDS 2017)*. 9 pages.

Raymond, C. and Riccardi, G. (2007). Generative and discriminative algorithms for spoken language understanding. In *Eighth Annual Conference of the International Speech Communication Association*, pages 1605–1608.

Takahashi, T. and Yokono, H. (2017). Two persons dialogue corpus made by multiple crowd-workers. In *Proceedings of the 8th International Workshop on Spoken Dialogue Systems (IWSDS 2017)*. 6 pages.

HAI Alice - An Information-Providing Closed-Domain Dialog Corpus

Jelte van Waterschoot[1]✉, Guillaume Dubuisson Duplessis[2], Lorenzo Gatti[3], Merijn Bruijnes[1] and Dirk Heylen[1]
[1]Human Media Interaction, University of Twente, 7522 NB, Enschede, The Netherlands
[2]CNRS, ISIR, Sorbonne Université, 75005, Paris, France
[3]FBK, University of Trento, 38123, Trento, Italy
l.gatti@fbk.eu, gdubuisson@telecom-paristech.fr
{j.b.vanwaterschoot, m.bruijnes, d.k.j.heylen}@utwente.nl

Abstract

The contribution of this paper is twofold: 1) we provide a public corpus for Human-Agent Interaction (where the agent is controlled by a Wizard of Oz) and 2) we show a study on verbal alignment in Human-Agent Interaction, to exemplify the corpus' use. In our recordings for the Human-Agent Interaction Alice-corpus (HAI Alice-corpus), participants talked to a wizarded agent, who provided them with information about the book *Alice in Wonderland* and its author. The wizard had immediate and almost full control over the agent's verbal and nonverbal behavior, as the wizard provided the agent's speech through his own voice and his facial expressions were directly copied onto the agent. The agent's hand gestures were controlled through a button interface. Data was collected to create a corpus with unexpected situations, such as misunderstandings, (accidental) false information, and interruptions. The HAI Alice-corpus consists of transcribed audio-video recordings of 15 conversations (more than 900 utterances) between users and the wizarded agent. As a use-case example, we measured the verbal alignment between the user and the agent. The paper contains information about the setup of the data collection, the unexpected situations and a description of our verbal alignment study.

Keywords: Corpus, Human-Agent Interaction, Wizard of Oz, Closed Domain, Information-Providing, Unexpected Situations, Verbal Alignment

1 Introduction

This paper presents the Human-Agent Interaction Alice-corpus (HAI Alice-corpus), a corpus of conversations between a user and an embodied conversational agent (ECA) operated by a wizard via the Wizard of Oz (WOz) method (Dahlbäck et al., 1993). The corpus has been collected as part of the ARIA-VALUSPA[1] project, in which a multimodal virtual agent for information retrieval that can deal with unexpected situations is being developed. The goal of the project is to provide interested parties, such as fellow researchers and industry, with a toolkit for building their own virtual agent (Valstar et al., 2016; Bruijnes et al., 2013). We collected the corpus to investigate how users react to unexpected situations in a conversation with an autonomous state-of-the-art virtual agent. However, in a classic WOz approach where the wizard uses a button interface, it is nearly impossible to improvise in unexpected situations. This is why we gave our wizard the freedom to choose his own words and facial expressions, to create and respond to unexpected situations. To the best of our knowledge most state-of-the-art agents cannot yet cope with interruptions effectively, nor perform human-level verbal alignment (Dubuisson Duplessis et al., 2017b). However, we believe we are approaching such capabilities in agents, thanks to recent efforts in agent development. Therefore, we wanted to simulate an agent as closely as possible with this type of WOz setup. We analyzed the corpus for verbal alignment in Human-Agent (H-A) interaction. Verbal alignment is a process during a dialog where participants reuse lexical and syntactic structures (Pickering and Garrod, 2004). One example of reusing structures is by speaking with similar words in the conversation as other dialogue participants (Reitter et al., 2006).



Figure 1: The conversational agent operated by the wizard

First, we will describe some related corpora in Section 2. In Section 3 we will provide more details on the setup of our Wizard of Oz system. More details on the data collection will be given in Section 4. Section 5 will be a description of what will be released with the corpus, and an example of a dialog from the corpus. In Section 6 we briefly describe our use-case for the corpus on the topic of verbal alignment. In Section 7 we will discuss the limitations of the corpus, suggestions for its use, and some future studies.

2 Related Work

More and more dialog corpora are available involving Human-Human interactions as well as Human-Machine interactions; see (Serban et al., 2015) for a recent and extensive study. However, a closer look at the Human-Machine

[1] www.aria-agent.eu

corpora reveals corpora are mostly available for spoken dialog systems, but very few are available in the Human-Agent community. The biggest difference between these corpora is that agents are often embodied and use multi-modal information, compared to only speech in spoken dialog systems. A well-known example of a corpus in H-A is the SEMAINE corpus (McKeown et al., 2010), dedicated to emotions in H-A Interactions. Recent efforts in collecting interaction data between a chatbot and a user include the UCAR corpus (Dubuisson Duplessis et al., 2016)[2], as well as the (RE-)WOCHAT international effort, which aims at collecting and annotating dialog data for chatbots and conversational agents (D'Haro et al., 2016)[3].
We believe that more H-A corpora can be very useful to analyze H-A interaction, in order to improve agents' communicative capabilities (e.g. a corpora study about adaptation and verbal alignment in H-A Interaction (Dubuisson Duplessis et al., 2017b)). This paper and the associated HAI Alice-corpus are a humble step towards sharing such valuable corpora for the H-A Interaction community.

3 Wizard of Oz System

The goal of the WOz system was to allow a user to have an interaction with an accomplice, who appears to the user as a virtual agent, and to record this interaction. Additionally, the user should believe (s)he is interacting with an advanced autonomous agent system[4].

3.1 Wizard and Agent

The wizard and participant were located in different rooms. The wizard could see and hear the participant through a Skype video-conference connection. The agent was rendered in real-time on the wizard's PC, and sent over the Skype connection to the participant. The wizard was trained to speak like a robot, with little pitch variation and a rhythm that is characteristic of generated speech. Additionally, using VoiceMeeter[5] and Reaper[6], his voice was passed through audio filters to make it sound more robotic. The facial animations of the agent were controlled using the wizard's facial expressions. The Unity game engine, using the FacePlus plugin by Mixamo[7], mapped the wizard's expressions on to the face of the agent. The body postures and gestures of the agent were controlled by the wizard through a button interface. With these tools, our wizard was able to improvise in unexpected situations, which would not have been possible in a button interface for classic wizards.

3.2 Recording

The system allowed for multi-modal data collection on the interaction between the user and wizard. The audio-video data in this corpus consists of the audio of both the wizard and the participant and the video of the agent.

4 Data Collection

In this section we will explain the scenario in the corpus, the variations and manipulations, information about the participants and the experiment protocol.

4.1 Domain

The conversational topic of our corpus is Alice's Adventures in Wonderland (also known as Alice in Wonderland), the famous novel of Lewis Carroll. For the data collection experiment, a virtual agent controlled by a wizard acted as an expert on Alice's Adventures in Wonderland (AiW). The agent was standing in a library and had the appearance of a middle-aged male intellectual, as can be seen in Figure 1. The wizard controlling the agent was well informed on the topic of AiW. The agent's utterances were prepared beforehand, based on possible questions that the agent might be asked during the interaction. To prepare for possible questions, a survey was held before the experiment and a pilot study was conducted via Reddit. Additionally, agent utterances to suggest topics were generated. The wizard studied these utterances, so that he could respond as fast as possible. He was allowed to paraphrase the utterances, as long as the meaning would remain the same. The participant, uninformed of the WOz setup, was instructed to ask questions about AiW that the agent had knowledge about: the events of the book, its characters, the author Lewis Carroll, and some other information around the book, such as its adaptations in films and others media.

4.2 Experimental Variations

Two variations of the AiW scenario exist. Condition **A** was intended to prime the user to try out the agent's capabilities rather than asking relevant questions on the book. Condition **B** was intended to be a credible version of an advanced and knowledgeable virtual agent that would increase the users' curiosity on the book.

- Condition **A**: the agent was not knowledgeable on the book, often repeated the same sentences ("*I unfortunately don't know about that*"), and intentional mistakes (such as using the wrong gender when saluting, or giving an incorrect answer to a question);
- Condition **B**, the agent answered questions to the best of its ability (and hid his ignorance when he could not answer), tried not to repeat himself, and took more time before answering.

4.3 Unexpected Situations

One aim was to record how participants respond to unexpected situations during an interaction with a virtual agent. We intentionally introduced sources of distraction and mistakes by the agent. These mistakes reflect technical limitations that exist in current state-of-the-art systems and that therefore often occur during a conversation with an autonomous agent. In particular, we created the following unexpected situations:

[2]Available at `https://ucar.limsi.fr`

[3]`http://workshop.colips.org/wochat/data/`

[4]For this reason we refer to the embodiment as an agent, and not as an avatar.

[5]`http://www.vb-audio.com/Voicemeeter/`

[6]`https://www.reaper.fm/`

[7]`https://www.mixamo.com`. Unfortunately, the FacePlus plugin has been removed from the Mixamo package since August 2017: `http://blogs.adobe.com/adobecare/2017/05/23/download-assets-from-mixamo/`

#	Sp.	Utterance
1	U	Hello
2	A	Hello madam, my name is James
3	U	I'm not a madam, but okay
4	A	I know everything about "Alice in Wonderland." Is there anything you would like to know?
5	U	Who wrote it?
6	A	Sorry, I did not get that
7	U	Who wrote "Alice in Wonderland"?
8	A	"Alice in Wonderland" was written by Charles [*indistinguishable audio*], or Lewis Carroll, which was his pen name [. . .]
9	U	So have you read the whole book, the "Alice in Wonderland"?
10	A	Yes, I am an expert of "Alice in Wonderland"
11	U	Yes, but have you read it?
12	A	Of course.
13	U	For how long was Alice falling?
14	E	[*interruption*] I'm sorry. Do you like some water, coffee or tea?
15	A	Don't you think it's rude to give attention to someone else during our conversation?
16	U	Given the fact that you're not a human, no
17	A	Okay. Is there something you would like to know about the animated picture?

Table 1: An excerpt from the corpus. (Sp=Speaker, U=User, A=Agent, E=Experiment leader)

Noise. Video and audio signals often contain recognition errors. This leads in autonomous agents to social signal processing and speech recognition with low accuracy and thus to an agent that does not understand the user. In the corpus this was mimicked by the wizard by saying "*Sorry, I did not get that*" when he did not understand the question (e.g., #6 in Table 1).

Missing Answers. Even with perfect recognition scores, autonomous agents do not know the answer to every question the user can pose and often have to say, for example, "*Sorry, I don't know that*". In the corpus the wizard said this when a question or answer was not in the script.

Wrong Responses. Autonomous agents sometimes respond in an inappropriate manner. For example, in an information-retrieval or query-answer matching approach the utterance of a user is mapped to existing query-answer pairs. The QA-pair with the most similar query to the user's utterance is selected and the agent performs a response with the answer of this pair. While easy to implement and robust in simple question-answering applications, this approach sometimes selects a 'wrong' answer (see for example (Schooten and op den Akker, 2011)). The wizard also used query-answer pairs to select the answer for the agent. In addition, sometimes the wizard intentionally gave a wrong answer to elicit a response from the user to this type of unexpected situation. An example of an inappropriate answer is #10 in Table 1.

Gender Detection. Autonomous agents' mistakes can lead to socially awkward situations. For example, referring to the gender of the user after misclassifying it. In the corpus the gender of the user was intentionally classified incorrectly (e.g., #2 in Table 1) to be able to record the reaction of users to this faux pas.

Interruptions. Interruptions in human-human conversations occur frequently and often without negatively impacting the flow of conversation. In H-A conversations interruptions are uncommon and often lead to a breakdown in communication when they do occur (op den Akker and Bruijnes, 2012). An interruption is an unexpected situation for an autonomous agent, but an agent successfully dealing with an interruption might be an unexpected situation for a user. Therefore, an (external) interruption was added to the corpus. An accomplice asked the participant whether they wanted a drink about four minutes into the conversation. This forced the participant to shift his attention and shortly lose engagement in the conversation. Additionally, the wizard commented on the interruption. This can be seen in Table 1, #14-15.

4.4 Recording Protocol

Participants were asked to read the two first pages of *Alice's Adventures in Wonderland*, before having a 7 minute conversation with the 'virtual expert'. Participants signed an informed consent form and filled in a short questionnaire, asking for their demographics, and knowledge on virtual agents and the AiW book. Participants were seated in front of a computer screen showing the agent and a Kinect camera. The experiment leader started the recordings and then left the room. The wizard, and thus the agent, started smiling and greeted the user, starting the conversation. After four minutes the experiment leader introduced an unexpected situation. He entered the room to interrupt the conversation by asking if the participant would like something to drink and left after the participant had responded. Three minutes later the experiment leader entered the room again to end the experiment. Finally, participants were debriefed, given the opportunity to ask questions about the experiment, and were introduced to the wizard.

5 Description of the Corpus

The corpus is available under the CC 4.0 license. We transcribed the corpus and distributed the transcription work amongst ourselves evenly (i.e. each author transcribing four dialogs) and one of the authors reviewed the full set of transcriptions afterwards for consistency.

5.1 Participants

A total of 16 volunteers were recruited to participate in the recording of this corpus. The participants of this study were divided across two conditions, **A** and **B**. In condition **A** there were 7 male and 1 female participants, of which 2 were native English speakers, with an average age of 30.38. In condition **B**, there were 4 male and 4 female participants, no native English speakers, with an average age of

Tag	Attributes	Explanation
<dialog>	id, condition, duration, gender	the type of condition (A or B), the duration of the conversation and the gender of the participant
<part>	type	parts of the experiment (before, during or after the interruption)
<utterance>	id, speaker	the text of the dialogs and who is speaking (user or agent)
<overlap>	id	used to annotate overlapping utterances (e.g., during interruptions)
<noise>	n/a	tag for responses of the agent because of noise
<missing>	n/a	tag for responses of the agent not knowing a response
<wrong>	n/a	tag for intentionally wrong responses by the agent

Table 2: Tags used in the XML data files

dialogs	15
Utterances	989 (A: 482, H: 487, E: 20)
Unique utterances	836 (A: 373, H: 450, E: 19)
Tokens	9900 (unique: 810)

Table 3: Figures about the collected corpus. (A=Agent, H=Human, E=Experimenter)

Type	average (std/min/max)
Utterances	65.93 (10.34 / 48 / 80)
Tokens	660 (93.71 / 502 / 914)
Tokens/human	316.8 (120.64 / 131 / 633)
Tokens/system	329.67 (51.96 / 225 / 441)
Tokens/human utterance	9.76 (7.86 / 1 / 60)
Tokens/system utterance	10.26 (6.07 / 1 / 50)

Table 4: Figures about the collected dialogs.

34.13. One participant in condition **A** did not give permission to include their interaction in the corpus and has therefore been excluded.

5.2 Contents

The video files of the H-A Interactions are available. Anonymized transcriptions of the dialogs are included and contain the utterances of the user, the agent, and the experimenter. Annotations are provided on the unexpected situations that were introduced in the interaction, see Table 2. The corpus is quantitatively described in Tables 3 and 4. We also include anonymized information about the participants, such as their demographic information and their familiarity with virtual agents.

6 Example Use-Case: Verbal Alignment

In this section, we will present a corpus study to show the importance of having an H-A Interaction corpus to make analysis of such interactions possible; and improve agent communicative capabilities.

6.1 Verbal Alignment in H-A Interactions

One striking observation of Human-Human (H-H) Interaction is that the communicative behaviors of dialog participants (DPs) tend to converge (Gallois et al., 2005) and automatically align at several levels (such as the lexical, syntactic and semantic ones) (Pickering and Garrod, 2004). In particular, DPs tend to reuse lexical and syntactic structures (Reitter et al., 2006). One consequence of successful alignment at several levels between DPs is a certain repetitiveness in dialog, leading to the development of a lexicon of fixed expressions. As a matter of fact, dialog participants tend to automatically establish and use fixed expressions that become dialog routines. Alignment is a subconscious phenomenon that naturally occurs in H-H dialogs (Pickering and Garrod, 2004). It has been shown to facilitate successful task-oriented conversations (Nenkova et al., 2008; Friedberg et al., 2012).

Linguistic alignment occurs in H-A Interactions. Indeed, users adopt lexical items and syntactic structures used by a system (Brennan, 1996; Stoyanchev and Stent, 2009; Parent and Eskenazi, 2010; Branigan et al., 2010). However, this alignment is only one-way: the system is usually not able to align on the user.

While verbal alignment has been investigated for H-H Interactions, it has been studied less for H-A Interaction. Recent work aims at studying H-A Interaction corpora to characterize the verbal alignment process (Dubuisson Duplessis et al., 2017b). Their study contrasts H-H and H-A Interaction corpora on a negotiation task over a definite set of objects (based on the H-H and H-A negotiation corpora (Gratch et al., 2016), unfortunately not publicly available). Among other things, they show that verbal alignment is symmetrical in H-H Interactions at the level of lexical repetitions, while it is asymmetrical in H-A Interactions. More specifically, they have shown that the human participant verbally aligns more, by adopting more agent-initiated lexical patterns, and by dedicating more tokens to the repetition of previously employed lexical patterns.

In this work, we aim at employing the same measures on the Alice corpus, to study the verbal alignment process in H-A interaction on an information-providing task driven by a sophisticated WOz able to adapt to the user.

6.2 Approach

Recent work proposes automatic verbal alignment measures based on the repetition between speakers at the lexical level (Dubuisson Duplessis et al., 2017b). In particular, they focus on which words and lexical patterns are shared between dialog participants. DPs are said to verbally align when they share and use common lexical pat-

terns ranging from single words (e.g., “Alice”, “cat”) to more elaborated patterns (e.g., “white rabbit”, “Alice in Wonderland”, “would like to know about”). This approach describes global and speaker-specific measures of verbal alignment based on repetition at the lexical level in dialog transcripts. It operates by mining lexical patterns (following the work of (Dubuisson Duplessis et al., 2017a)), automatically building a lexicon of shared expressions and deriving verbal alignment measures. The expression lexicon keeps track of shared expressions in a dialog and valuable features about these expressions (e.g., who first produced this expression, its frequency). Then, straightforward measures are derived by leveraging both the dialog transcript and the dialog lexicon. In this paper, we focus on the following verbal alignment measures proposed in (Dubuisson Duplessis et al., 2017b):

Expression Variety (EV). The total number of tokens in the dialog. This ratio indicates the variety of the expression lexicon relatively to the length of the dialog. The higher it is, the more there are different expressions established between DPs.

Expression Repetition (ER). Ratio of produced tokens belonging to a repetition of shared expressions. The higher the ER is, the more DPs dedicate tokens to the repetition of expressions.

Initiated Expression (IE_S). Ratio of shared expressions initiated by locutor S.

Expression Repetition (ER_S). Ratio of produced tokens belonging to the repetition of a shared expressions for locutor S. The software used to compute the measures is free and open-source[8].

6.3 Preliminary Results

We first look at the speaker-independent measures. The expression variety (EV) ranges from approx. 0.07 to 0.12 (mean=0.095, std=0.014, median=0.098). This is less than what is reported for the H-A negotiation corpus in (Dubuisson Duplessis et al., 2017b). This indicates that DPs constitute less varied expression lexicons in the information-providing dialogs than in the negotiation ones. The expression repetition (ER) ranges from approx. 0.23 to 0.40 (mean=0.330, std=0.048, median=0.346). Once again, this is less than what is reported for the H-A negotiation corpus. This indicates that DPs in the Alice corpus dedicate fewer tokens to the repetition of shared expressions.

Next, we take a closer look at each speaker in the dialog in terms of initiated expressions (IE_S) and expression repetition (ER_S). A clear asymmetry between the agent and the human appears at the initiation of shared expressions (see Figure 2). Here, the agent initiates most of the shared expressions with an IE_A ranging from 0.50 to 0.72 (mean=0.61, std=0.06, median=0.61). This difference is statistically significant (Wilcoxon signed rank test, $V = 120$, p-value=6.104×10^{-5}). However, this asymmetry does not appear at other measurable levels. Indeed, the agent and the users dedicate the same number of tokens for repeating expressions (ER_S), produce a comparable number of tokens in dialog and relatively share the same amount of vocabulary with each other[9].

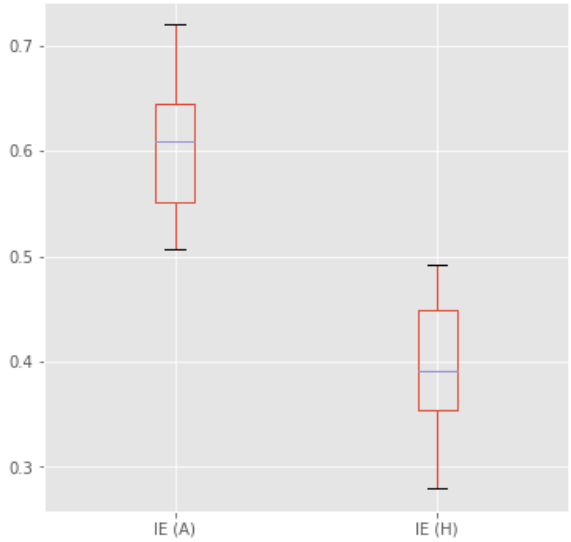


Figure 2: Initiated Expressions (IE_S). “A” is the agent, “H” is the human participant. The difference is significant ($p < 0.001$).

We have not found any significant difference in the verbal alignment measures between conditions **A** and **B**. This is possibly due to the small sizes of these groups (7 dialogs for condition **A**, 8 for condition **B**).

6.4 Discussion of the Use-Case

In this section, we have provided a preliminary quantitative study of verbal alignment on the Alice corpus with a comparison to previous work. This study is one of the few that focuses on verbal alignment in H-A interaction. Its originality lies in the nature of the corpus (information-providing dialogs), and in the sophisticated WOz system that allows the wizard to adapt to the user. This constitutes a step towards a better understanding of verbal alignment processes in Human-Agent Interaction. Globally, our preliminary study indicates that the Alice corpus presents a quantitatively weaker verbal alignment process at the level of lexical pattern repetition than the H-A negotiation corpus studied in (Dubuisson Duplessis et al., 2017b). This is shown by the emergence between DPs of less varied expression lexicons, and by the fact that DPs dedicate less tokens to repeating lexical patterns.

At the speaker level, it turns out that the Alice corpus displays an asymmetry between the agent and the human; which is a feature that seems to discriminate H-H interactions from H-A ones in terms of verbal alignment (Dubuisson Duplessis et al., 2017b). Here, the agent initiates more shared expressions than the human participant. One explanation is that the agent leads the interaction and often triggers the questions (e.g., “Is there anything you would like to know about falling jars?”). However, the agent and the user repeat lexical patterns to the same degree. This shows

[8] Software available at: https://github.com/GuillaumeDD/dialign

[9] Relative shared vocabulary for S_1 is computed as follow: $\text{SV}_{S_1} = \frac{\#(\text{Tokens}_{S_1} \cap \text{Tokens}_{S_2})}{\#(\text{Tokens}_{S_1})}$

a tendency towards a symmetrical verbal alignment, which is closer to what has been previously observed in H-H Negotiation Interactions. In other words, this indicates a more "human-like" verbal alignment in the Alice corpus compared to the H-A Negotiation corpus. One explanation can be found in the WOz system. In the Alice corpus, the WOz operator is not strictly constrained in its linguistic choices, and can thus verbally align with the user. Conversely, the WOz system in the negotiation corpus is restricted to preformatted utterances and templates, which prevent the WOz from aligning with the user.

All in all, this preliminary study indicates that variations in verbal alignment at the level of shared lexical patterns not only can be quantified between H-H and H-A Interactions but also between H-A corpora. Further studies are thus required to better understand the verbal alignment processes occurring in H-A Interactions, and to improve the adaptive communicative capabilities of agents interacting with humans.

7 Discussion and Conclusion

We have described a Wizard of Oz setup and corpus (HAI Alice-corpus) in an information-providing and closed-domain setting, containing multiple types and variants of unexpected situations. There were two conditions, one in which the user would focus more on the agent behavior and one where the user would focus more on the content of the dialog. We expect this corpus to be a useful resource for the Human-Agent Interaction community, because few similar corpora exist. In the full paper we will describe a study on verbal alignment in human-machine interaction to showcase the usefulness of the corpus.

We are aware of the limitations of this corpus. The size of the HAI Alice-corpus is quite small, it contains 15 dialogs. However, the corpus might be combined with other corpora to create a larger dataset. Furthermore, we have only transcribed the speech of the conversations and we did not annotate the data with time markers corresponding to the video. Lastly, non-verbal behaviors have not been annotated even though these could contain interesting information for other researchers.

Currently, we are integrating a verbal alignment tool into the ARIA-VALUSPA platform. This will allow an autonomous agent to utilize verbal alignment strategies in conversations with a user. We will use the HAI Alice-corpus, and in particular the verbal alignment scores from the corpus, to evaluate the performance of the autonomous agent on verbal alignment.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 645378. Special thanks to Siewart van Wingerden and Adrien Luxey for their efforts in recording the corpus.

References

Branigan, H. P., Pickering, M. J., Pearson, J., and McLean, J. F. (2010). Linguistic alignment between people and computers. *Journal of Pragmatics*, 42(9):2355–2368.

Brennan, S. E. (1996). Lexical entrainment in spontaneous dialog. In *Proceedings of the International Symposium on Spoken Dialogue (ISSD)*, volume 96, pages 41–44, Philadelphia (CA), USA.

Bruijnes, M., Kolkmeier, J., op den Akker, H. J. A., Linssen, J. M., Theune, M., and Heylen, D. K. J. (2013). Keeping up stories: design considerations for a police interview training game. In *Proceedings of the Social Believability in Games Workshop (SBG2013)*, Enschede, The Netherlands.

Dahlbäck, N., Jönsson, A., and Ahrenberg, L. (1993). Wizard of Oz studies - why and how. *Knowledge-based systems*, 6(4):258–266.

D'Haro, L. F., Shawar, B. A., and Yu, Z. (2016). RE-WOCHAT 2016–Shared task description report. In *Proceedings of the Workshop on Collecting and Generating Resources for Chatbots and Conversational Agents - Development and Evaluation (RE-WOCHAT)*, page 39, Portorož, Slovenia.

Dubuisson Duplessis, G., Letard, V., Ligozat, A.-L., and Rosset, S. (2016). Purely corpus-based automatic conversation authoring. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2728–2735, Portorož, Slovenia.

Dubuisson Duplessis, G., Charras, F., Letard, V., Ligozat, A.-L., and Rosset, S. (2017a). Utterance Retrieval based on Recurrent Surface Text Patterns. In *Proceedings of the 39th European Conference on Information Retrieval (ECIR 2017)*, pages 199–211, Aberdeen, UK.

Dubuisson Duplessis, G., Clavel, C., and Landragin, F. (2017b). Automatic Measures to Characterise Verbal Alignment in Human-Agent Interaction. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL 2017)*, pages 71–81, Saarbrücken, Germany.

Friedberg, H., Litman, D., and Paletz, S. B. (2012). Lexical entrainment and success in student engineering groups. In *Proceedings of the 2012 IEEE Spoken Language Technology Workshop (SLT 2012)*, pages 404–409, Miami (FL), USA.

Gallois, C., Ogay, T., and Giles, Howard, H. (2005). Communication accommodation theory: A look back and a look ahead. In W. Gudykunst, editor, *Theorizing about intercultural communication*, pages 121–148. Sage, Thousand Oaks, CA.

Gratch, J., DeVault, D., and Lucas, G. (2016). The Benefits of Virtual Humans for Teaching Negotiation. In *Proceedings of the 16th International Conference on Intelligent Virtual Agents (IVA 2016)*, pages 283–294, Los Angeles (CA), USA.

McKeown, G., Valstar, M. F., Cowie, R., and Pantic, M. (2010). The SEMAINE corpus of emotionally coloured character interactions. In *Proceedings of the 2010 IEEE International Conference on Multimedia and Expo (ICME 2010)*, pages 1079–1084, Singapore.

Nenkova, A., Gravano, A., and Hirschberg, J. (2008). High frequency word entrainment in spoken dialogue. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language*

Technologies (ACL-08: HLT): Short papers, pages 169–172, Columbus (OH), USA.

op den Akker, H. J. A. and Bruijnes, M. (2012). *Computational models of social and emotional turn-taking for embodied conversational agents: a review*. Number TR-CTIT-12-13 in CTIT Technical Report Series. University of Twente, Centre for Telematics and Information Technology (CTIT).

Parent, G. and Eskenazi, M. (2010). Lexical entrainment of real users in the Let's Go spoken dialog system. In *Proceedings of INTERSPEECH 2010*, pages 3018–3021, Makuhari, Japan.

Pickering, M. J. and Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(02):169–190.

Reitter, D., Keller, F., and Moore, J. D. (2006). Computational modelling of structural priming in dialogue. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (HLT-NAACL 2006): Short Papers*, pages 121–124, New York City (NY), USA.

Schooten, B. and op den Akker, R. (2011). Vidiam: Corpus-based development of a dialogue manager for multimodal question answering. *Interactive Multimodal Question-Answering*, pages 25–56.

Serban, I. V., Lowe, R., Charlin, L., and Pineau, J. (2015). A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742*.

Stoyanchev, S. and Stent, A. (2009). Lexical and syntactic priming and their impact in deployed spoken dialog systems. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (NAACL-HLT): Short Papers*, pages 189–192, Boulder (CO), USA.

Valstar, M., Baur, T., Cafaro, A., Ghitulescu, A., Potard, B., Wagner, J., André, E., Durieu, L., Aylett, M., Dermouche, S., et al. (2016). Ask Alice: an artificial retrieval of information agent. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction)ICMI 2016)*, pages 419–420, Tokyo, Japan.

Augmenting Image Question Answering Dataset by Exploiting Image Captions

Masashi Yokota, Hideki Nakayama
Graduate School of Information Science and Technology,
The University of Tokyo
Tokyo, JP
{yokota, nakayama}@nlab.ci.i.u-tokyo.ac.jp

Abstract

Image question answering (IQA) is one of the tasks that need rich resources, i.e. supervised data, to achieve optimal performance. However, because IQA is a challenging task that handles complex input and output information, the cost of naive manual annotation can be prohibitively expensive. On the other hand, it is thought to be relatively easy to obtain relevant pairs of an image and text in an unsupervised manner (e.g., crawling Web data). Based on this expectation, we propose a framework to augment training data for IQA by generating additional examples from unannotated pairs of an image and captions. The important constraint that a generated IQA example must satisfy is that its answer must be inferable from the corresponding image and question. To satisfy this, we first select a possible answer for a given image by randomly extracting an answer from corresponding captions. Then we generate the question from the triplets of the image, captions and fixed answer. In experiments, we test our method on the Visual Genome dataset varying the ratio of seed supervised data and demonstrate its effectiveness.

Keywords: Semi-Supervised Learning, Self-labeling, Image Question Generation, Image Question Answering

1. Introduction

The objective of image question answering (IQA) is to predict a correct answer given an image and a question. To achieve a satisfactory performance, we generally require a large amount of human-annotated data which is expensive to prepare. One of the possible ways to mitigate the burden of collecting annotations is generating a pseudo dataset, which is one of the effective approaches in semi-supervised learning. Although this approach has been well-known to benefit a variety of tasks such as image classification (Lee, 2013), neural machine translation (He et al., 2016), and reading comprehension (Yang et al., 2017), its effectiveness has not yet been demonstrated for IQA.

In this study, we address the resource problem in IQA by proposing a framework of pseudo data generation exploiting captions as auxiliary information. In IQA, a generated pseudo dataset that comprises triplets consisting of an image I, question Q, and answer A should satisfy an important constraint: the components of a triplet (I, Q, A) must be relevant to each other. In other words, the answer A must be inferable from an image-question pair (I, Q). We address the problem of component relevancy by focusing on text captions C as the additional information. Our expectation is that pairs of mutually relevant image and captions are relatively easily obtainable from the Web (e.g. Wikipedia, BBC News, etc.). These captions are expected to contain information in images such as object colors, human actions, relationships between objects, etc. We first fix the answer A by sampling a token from captions as a possible answer for an image, and then generate a question conditioned by the triplet (I, C, A). For example, as shown in Fig. 1, we randomly select the token "bat" from the caption "A man holding a baseball bat." and then use the token as an answer. Finally, we generate the question "What is the man holding?" using the given image, caption and selected answer.

Our overall semi-supervised IQA framework consists of three phases:

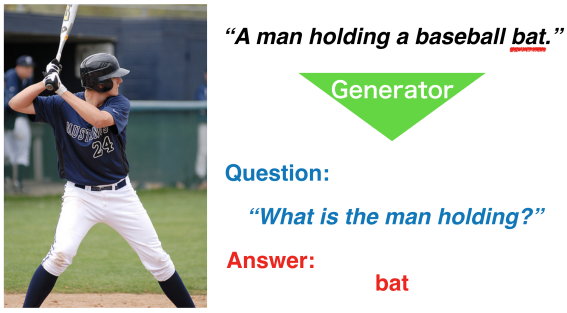


Figure 1: The example of a generated IQA triplet. The answer, underlined in red, is extracted from the given caption "A man holding a baseball bat." The generator produces the question using the image, caption and extracted answer.

(1) training an image question generation (IQG) model with a small annotated (i.e. original) dataset $\{(I, Q, A, C)\}$,

(2) generating a pseudo IQA dataset which consists of triplets $\{(I, Q, A)\}$ from the additional unannotated data $\{(I, C)\}$,

(3) training an IQA model with both the original and pseudo IQA datasets.

Figure 1 shows an example of a pseudo IQA triplet (I, Q, A) generated by our framework. The overview of the phase (2) and (3) is shown in Fig. 2.

Our experiments show that pseudo data generated by our method improves IQA performance as compared to the baseline trained with the original dataset and outperforms other naive data-augmentation methods.

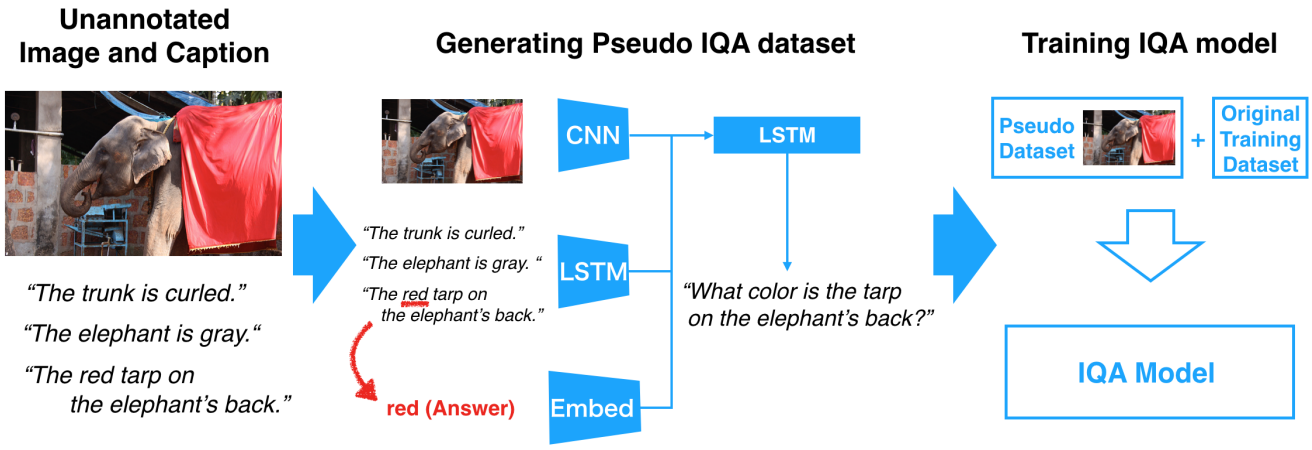


Figure 2: Overview of phase (2) and (3) of our framework. In phase (2), we extract an answer A from captions C associated with an image I, and subsequently generate a question from triplet (I, C, A). In phase (3), we train an IQA model with an original training dataset and generated pseudo dataset.

2. Related Work

Semi-supervised learning has achieved outstanding performance typically in single modal tasks, e.g. image classification and machine translation. The semi-supervised learning approaches exploiting unlabeled data can be categorized into four types: graph embedding methods (Yang et al., 2016), manifold learning methods (Rifai et al., 2011), generative models (Kingma et al., 2015; Rasmus et al., 2015) and self-labeling methods (Lee, 2013; He et al., 2016).

Roughly speaking, the first three approaches aim to obtain better representations or decision boundaries using unlabeled data. However, it is difficult to directly apply these methods to multi-input tasks because multi-modal data often has complex structures, making it unclear how to improve representations or decision boundaries efficiently.

The self-labeling approaches utilize unlabeled data by labeling it with the classes predicted by the model. Yang et al. (2017) has improved the performance of reading comprehension task with this approach. Their framework has two models: the question answering (QA) model and question generation (QG) model. The QA model aims to predict a correct answer from a given question and document. The goal of the QG model is to generate QA pairs from unlabeled documents. In the training phase, the QA model is updated using supervised learning with both the original dataset and generated pseudo dataset. The QG model is updated using reinforcement learning whose rewards are the accuracy of the QA model. Inspired by their work, our framework consists of the QA model and QG model. There are two differences between their work and our framework: using a multi-modal dataset and applying a supervised manner to both the QA model and QG model. Because of using only a supervised manner, our method is simpler than their work.

3. Question Generation

We train an IQG model with an original supervised training dataset consisting of quadruplets $\{(I, Q, A, C)\}$. Using the trained model, a pseudo IQA dataset is then generated from the additional unannotated data $\{(I, C)\}$ where we extract answers $\{A\}$ from captions $\{C\}$, and then generate questions from triplets $\{(I, C, A)\}$ by using the IQG model.

3.1. Image Question Generation Model

Our IQG model generates a question from a triplet (I, C, A). The model consists of sub networks of an image encoder, an answer encoder, a caption encoder and a question decoder. The image encoder and answer encoder are a convolutional neural network (CNN) and a word embedding layer, respectively. The caption encoder and the question decoder are both recurrent neural networks. Similarly to (Xu et al., 2015), the decoder predicts a question with an attention mechanism. Figure 3 shows the overview of the IQG model.

Image Encoder

We use a CNN to extract image representation vectors $\boldsymbol{X}$ that consist of L vectors from an image,

$$\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_L\}. \quad (1)$$

Each representation vector $\boldsymbol{x}_i$ corresponds to the local regional vector of the image.

Caption Encoder

Each sentence in a caption is a sequence of word tokens. We encode each caption using a Long Short-Term Memory Network (LSTM) (Hochreiter and Schmidhuber, 1997). Let the number of captions relevant to the image be K. The caption representation vectors $\boldsymbol{D}$ are computed as follows:

$$\boldsymbol{D} = \{\boldsymbol{d}_1, ..., \boldsymbol{d}_K\}, \boldsymbol{d}_i = \mathrm{LSTM}(\boldsymbol{S}_i), \quad (2)$$

where $\boldsymbol{S}_i$ is the i-th sentence of the caption and $\boldsymbol{d}_i$ is the hidden state of LSTM given $\boldsymbol{S}_i$.

Answer Encoder

We tokenize and embed the input answer to obtain the answer feature vector $\boldsymbol{a}$,

$$\boldsymbol{a} = \boldsymbol{E}\boldsymbol{w}, \quad (3)$$

where $\boldsymbol{E}$ is a lookup table and $\boldsymbol{w}$ is a one-hot vector of the input answer.

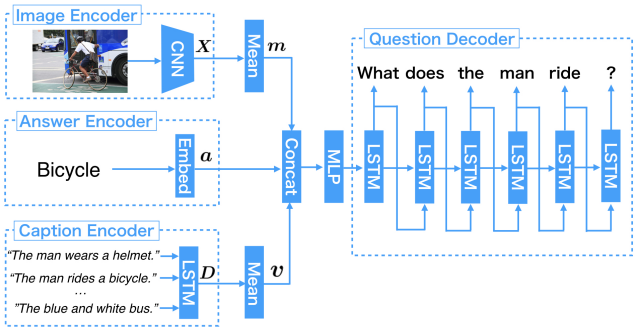


Figure 3: Overview of the IQG model which consists of three encoders and a decoder. Its input is a triplet (I, Q, A). $\boldsymbol{X}$ and $\boldsymbol{D}$ are the image feature and caption feature encoded respectively by a CNN and LSTM. The answer feature vector $\boldsymbol{a}$ is encoded by an embedding layer. $\boldsymbol{m}$ is the average of $\boldsymbol{X}$, and $\boldsymbol{v}$ is the average of $\boldsymbol{D}$. Using $\boldsymbol{m}$, $\boldsymbol{v}$ and $\boldsymbol{a}$ through the multilayer perceptron (MLP), we initialize the memory cell and hidden state of a question decoder which is an LSTM.

Question Decoder

We use an LSTM as a question decoder. $\boldsymbol{c}_t$ and $\boldsymbol{h}_t$ are the memory cell and hidden state of the question decoder at the timestep t. The initial memory cell $\boldsymbol{c}_0$ and hidden state $\boldsymbol{h}_0$ of the question decoder are given by

$$\boldsymbol{c}_0 = \mathrm{F}([\boldsymbol{m}; \boldsymbol{v}; \boldsymbol{a}]), \tag{4}$$

$$\boldsymbol{h}_0 = \mathrm{G}([\boldsymbol{m}; \boldsymbol{v}; \boldsymbol{a}]), \tag{5}$$

where $[\,\cdot\,;\,\cdot\,]$ represents vector concatenation, F and G indicate two-layer perceptrons, $\boldsymbol{m}$ is the average of $\boldsymbol{X}$, and $\boldsymbol{v}$ is the average of $\boldsymbol{D}$.

The decoder uses a soft attention mechanism (Xu et al., 2015) to utilize the features of the image and captions. At timestep t, the image context vector $\boldsymbol{y}_t$ and caption context vector $\boldsymbol{z}_t$ are given by

$$\boldsymbol{y}_t = \sum_{l=1}^{k} \alpha_{t,l} \boldsymbol{x}_l, \tag{6}$$

$$\alpha_{t,l} = \mathrm{softmax}(\mathrm{Q}([\boldsymbol{h}_{t-1}; \boldsymbol{a}; \boldsymbol{x}_l])), \tag{7}$$

$$\boldsymbol{z}_t = \sum_{l=1}^{k} \beta_{t,l} \boldsymbol{d}_l, \tag{8}$$

$$\beta_{t,l} = \mathrm{softmax}(\mathrm{R}([\boldsymbol{h}_{t-1}; \boldsymbol{a}; \boldsymbol{d}_l])), \tag{9}$$

where $\alpha_{t,l}$ and $\beta_{t,l}$ are image and caption attention weights, respectively, and Q and R are two-layer perceptrons.

The hidden state of the decoder is updated using the expression by the following rule:

$$\boldsymbol{h}_t = \mathrm{LSTM}([\boldsymbol{q}_{t-1}; \boldsymbol{h}_{t-1}; \boldsymbol{y}_t; \boldsymbol{z}_t; \boldsymbol{a}]). \tag{10}$$

The question $\boldsymbol{q}$ is a sequence of word tokens

$$\boldsymbol{q} = (\boldsymbol{q}_1, ..., \boldsymbol{q}_n), \tag{11}$$

where n is the question length. The decoder computes the probability $p(q_t|\boldsymbol{I}, \boldsymbol{D}, \boldsymbol{a})$ of a word token in the following way:

$$p(q_t|\boldsymbol{I}, \boldsymbol{D}, \boldsymbol{a}) = \mathrm{softmax}(\mathrm{H}(\boldsymbol{q}_{t-1}; \boldsymbol{h}_t; \boldsymbol{y}_t; \boldsymbol{z}_t; \boldsymbol{a})), \tag{12}$$

where H is a two-layer perceptron.

For training the IQG model, we minimize the softmax cross-entropy loss of each output by Stochastic Gradient Descent.

3.2. IQA Dataset Generation Procedure

We explain how to generate a pseudo IQA dataset from the additional unannotated data $\{(I, C)\}$ using the trained IQG model. The key question here is how to get plausible answers to use as inputs for question generation. Our dataset approach consists of two steps. We first sample the answers $\{A\}$ from N-grams of captions $\{C\}$ that contain any of the answer classes appearing in the original (seed) training data. The reason for using N-grams is that the target answer types in this study are words or phrases, e.g. "apple", "playing tennis" etc. Subsequently, we generate questions $\{Q\}$ from triplets $\{(I, C, A)\}$ using the IQG model. Finally, we use the generated triplets $\{(I, Q, A)\}$ as a pseudo IQA dataset.

4. Experiment

We conduct experiments using various sizes of the supervised training dataset to see how our data-augmentation framework improves IQA performance with respect to the amount of available annotated data.

4.1. Settings

The Visual Genome (VG) (Krishna et al., 2016) dataset is used in our experiments. As the VG dataset does not have a specified test split, we split the whole dataset into partitions of 20%/10%/10%/60% for training/validation/test/generation set, respectively. The training/validation/test datasets consist of quadruplets $\{(I, Q, A, C)\}$. The generation dataset contains pairs $\{(I, C)\}$. We use the training dataset to train both IQG and IQA models. The validation dataset is used to determine the hyperparameters of both models. The generation dataset is used as a pool of unannotated data which is used for generating pseudo IQA data. The test dataset is only used for the evaluation of final IQA model trained on the training dataset plus generated pseudo training dataset.

We use a question vocabulary consisting of words appearing over 100 times in the questions in the training dataset. A caption vocabulary is prepared in the same way as the question vocabulary. We replace out-of-vocabulary words with a special token ⟨UNK⟩. Answers appearing more than 100 times in the training dataset are used as answer classes. To evaluate an IQA model, we calculate the average classification accuracy on the test dataset. We use a subset of the test dataset consisting of examples whose answers are included in the answer classes defined above.

4.2. Baselines

We describe two simple baselines that produce a pseudo dataset.

A paraphrase-based approach generates question paraphrases of the original dataset. Given a triplet (I, Q, A)

of the original dataset, the method generates a paraphrase of the given question. We use the lexical paraphrases from the PPDB (Ganitkevitch et al., 2013) dataset (size S). If a question contains a word that appears in the PPDB dataset, the method generates a question by paraphrasing the word.
An object-detection-based approach generates pseudo IQA pairs by using YOLO V2 (Redmon and Farhadi, 2017), an object detection model. An object detection model can locate objects in a given image. We can generate an IQA pair by using the predicted information: the class of the object, and the number of the objects. We generate questions by using two templates: ("What is in the picture?", "*class*") and ("How many *class* are there?", "*number*"). For example, if the model detects one "cat" in the image, we generate two QA pairs: (question, answer) = ("What is in the picture?", "cat"), ("How many cats are there?", "one").

4.3. Implementation Details

We explain the implementation details of both the IQG and IQA models.

4.3.1. IQG Model

We use features from the pre-trained Resnet-152 (He et al., 2015) as the image representation vectors for the IQG model. All input images are rescaled to 448×448. The image features are extracted from the last convolutional layer (and their shape is $14 \times 14 \times 2048$). The sizes of the hidden state of the caption encoder and question decoder are 512. The dimensionality of the answer representation vector is 1024.
We optimize our IQG model with SMORMS3 (Funk, 2015) optimizer. The learning rate is set to 1.0×10^{-4}, and the batch size is 50. K, the number of captions relevant to an image is 15. Dropout (rate 0.5) (Srivastava et al., 2014) is applied to each layer for regularization. We also use weight decay of 1.0×10^{-5}.

4.3.2. IQA Model

We use DeeperLSTMQ, which is the VQA (Agrawal et al., 2015) baseline model, as the IQA model. We use the same hyperparameters and training manner described in the original paper.

4.4. Results and Analysis

Table 1 shows the results of our methods using subsets of training data (*Seed*) of varying sizes: 75K, 100K, 150K, 200K, 300K. $|L|$ and $|U|$ denote the sizes of the seed and pseudo dataset, respectively. *Para*, *OD* and *QG* denote the pseudo datasets generated by the paraphrase-based method, object-detection-based method and our proposed method, respectively.
The results show that each data-augmentation method improves the performance of IQA models as compared to the simple supervised learning (*Seed*). Although the improvement is not always significant, it becomes more prominent when the size of *Seed* data is small. Particularly, our method decreases the test error by 1.24% in the case of the annotated data of size 75K. This result suggests the effectiveness of our approach in the context of low-resource learning.

$\|L\|$	$\|U\|$	Training Data	Acc. [%]
75K	-	*Seed*	36.68
	300K	*Seed* + *Para*	37.28
		Seed + *OD*	37.76
		Seed + *QG*	**37.84**
100K	-	*Seed*	38.61
	300K	*Seed* + *Para*	39.03
		Seed + *OD*	38.95
		Seed + *QG*	**39.09**
150K	-	*Seed*	40.03
	300K	*Seed* + *Para*	40.26
		Seed + *OD*	40.13
		Seed + *QG*	**40.36**
200K	-	*Seed*	40.97
	300K	*Seed* + *Para*	41.04
		Seed + *OD*	41.09
		Seed + *QG*	**41.28**
300K	-	*Seed*	41.95
	300K	*Seed* + *Para*	41.96
		Seed + *OD*	41.98
		Seed + *QG*	**42.00**

Table 1: The IQA model performance on the test dataset of Visual Genome with various sizes of the annotated data. $|L|$ is the annotated dataset size, and $|U|$ is the size of generated data. *Seed* denotes the original training dataset. *Para* and *OD* stand for the pseudo datasets produced by the baselines. *QG* is the pseudo dataset produced by our proposed method.

Moreover, we can observe the effectiveness of captions from the results. From a theoretical viewpoint, it is not very surprising that our method outperforms the baselines since our method uses additional information (captions) to generate questions. However, considering that we can now easily obtain image-caption pairs from the web, we believe that our method is a practically promising approach to improve an IQA model with less human effort.

5. Conclusion

We have proposed a method to generate a pseudo IQA dataset which we believe is a promising approach to tackle resource problem in IQA. The key notion in our approach is that, to satisfy the component relevancies of triplets $\{(I, Q, A)\}$, we extract plausible answers A from external captions, and then produce questions from triplets $\{(I, C, A)\}$. Our model outperforms simple supervised-only baseline as well as other data-augmentation methods, which indicates the effectiveness of image-caption pairs for question generation.

Acknowledgements

We thanks the anonymous reviewers for their constructive comments on this work. This work was supported by JSPS KAKENHI Grant Number 16H05872.

References

Agrawal, A., Lu, J., Antol, S., Mitchell, M., Zitnick, C. L., Batra, D., and Parikh, D. (2015). Vqa: Visual question answering. In *Proc. of ICCV*.

Funk, S. (2015). Rmsprop loses to smorms3 - beware the epsilon! Technical report.

Ganitkevitch, J., Durme, B. V., and Callison-Burch, C. (2013). PPDB: The paraphrase database. In *Proc. of NAACL*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. Technical report.

He, D., Xia, Y., Qin, T., Wang, L., Yu, N., Liu, T.-Y., and Ma, W.-Y. (2016). Dual learning for machine translation. In *Proc. of NIPS*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9:1735–1780.

Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. (2015). Semi-supervised learning with deep generative models. In *Proc. of NIPS*.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., and Fei-Fei, L. (2016). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *arXiv preprint arXiv:1602.07332*.

Lee, D.-H. (2013). Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. In *Proc. of ICML*.

Rasmus, A., Berglund, M., Honkala, M., Valpola, H., and Raiko, T. (2015). Semi- supervised learning with ladder networks. In *Proc. of NIPS*.

Redmon, J. and Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *Proc. of CVPR*.

Rifai, S., Dauphin, Y. N., Vincent, P., Bengio, Y., and Muller, X. (2011). The manifold tangent classifier. In *Proc. of NIPS*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15:1929–1958.

Xu, K., Ba, J. L., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proc. of ICMP*.

Yang, Z., Cohen, W., and dinov, R. S. (2016). Revisiting semi-supervised learning with graph embeddings. In *Proc. of ICML*.

Yang, Z., Hu, J., Salakhutdinov, R., and Cohen, W. W. (2017). Semi-supervised qa with generative domain-adaptive nets. *arXiv preprint arXiv:1702.02206*.

Semi-supervised Training Data Generation for Multilingual Question Answering

Kyungjae Lee, Kyoungho Yoon, Sunghyun Park, Seung-won Hwang
Yonsei University, Department of Computer Science
Yonsei-ro 50, Seoul, South Korea
{lkj0509, e05087, pshy410, seungwonh}@yonsei.ac.kr

Abstract

Recently, various datasets for question answering (QA) research have been released, such as SQuAD, Marco, WikiQA, MCTest, and SearchQA. However, such existing training resources for these task mostly support only English. In contrast, we study semi-automated creation of the Korean Question Answering Dataset (K-QuAD), by using automatically translated SQuAD and a QA system bootstrapped on a small QA pair set. As a naïve approach for other language, using only machine-translated SQuAD shows limited performance due to translation errors. We study why such approach fails and motivate needs to build seed resources to enable leveraging such resources. Specifically, we annotate seed QA pairs of small size (4K) for Korean language, and design how such seed can be combined with translated English resources. These approach, by combining two resources, leads to 71.50 F1 on Korean QA (comparable to 77.3 F1 on SQuAD).

Keywords: QA dataset, Machine Comprehension, Multilingual Resource

1. Introduction

Understanding text and answering questions about the text is an important yet challenging task for machines. To train machines for doing that, large-scale resources are necessary. For English, SQuAD (Rajpurkar et al., 2016) provides more than 100K pairs of questions and supporting passages: For example, for the question, such as "what causes precipitation to fall?", they locate the supporting "span", or the relevant answer found in the passage, such as "precipitation ... falls under gravity". As the advent of deep learning techniques turns algorithms data hungry, building resources of a significant size is a critical task, question answering (QA) systems, as demonstrated not only by SQuAD, but also by ImageNet for object recognition (Deng et al.,).

However, existing datasets for these tasks are mostly built on English. For other languages, building resources of large scale is labor intensive. Meanwhile, some approaches transfer English resources to resource-poor language for NLP tasks. As improvement of neural-based Machine Translation (MT) (Bahdanau et al., 2014) and availability for 100+ languages, some research takes advantage of MT systems for multilingual tasks such as sentiment analysis (Balahur and Turchi, 2014), query-document relevance (Ture and Boschee, 2014), and named entity recognition (Dandapat and Way, 2016). Given this opportunity, our research questions are as follow:

- **RQ1:** Can we build QA resource for other language, solely by automatically translating English resources?
- **RQ2:** When annotating QA resource manually for other language, can we save efforts by leveraging existing resources for English?
- **RQ3:** How can we combine the two to complement each other?

Regarding **RQ1**, we study **M**achine-**T**ranslated resource of a SQuAD, denoted as **MT**. Assuming perfect translation, **MT** should be sufficient to train a QA system of comparable performance to English QA trained on SQuAD. However, there are two challenges: (1) The position of answer span changes or is lost in translation. (2) Translation quality varies over QA pairs. Without overcoming these challenges, QA performance on **MT** is merely 52.49 in F1 (while 77.3 for SQuAD).

Regarding **RQ2**, the low performance on **MT** naturally motivates building language-specific (Korean) resource for QA. We construct and release such QA resource annotated by multiple human annotators, which we explain in Section 3. More interestingly, we show that, by selectively leveraging **MT**, manual effort for annotating Korean resources decreases drastically, from 100K+ pairs of SQuAD to mere 4K.

Finally, for **RQ3**, we train a QA system for Korean QA, combining both small-scale annotated data and large-scale translated resources we discussed above. As motivated from RQ1, there still remains the problem that the quality of the translation varies. To overcome this problem, we predict **translation certainty** for QA pairs, to use only high-quality QA pairs for training. With such selective training, we achieve 71.50 in F1 score.

2. Related Works

The availability of training resources has been driving QA research, ranging from early and small datasets such as WikiQA (Yang et al., 2015), to more recent and big datasets such as SQuAD (Rajpurkar et al., 2016), MS Marco (Nguyen et al., 2016), and SearchQA (Dunn et al., 2017) dataset. Our work complements these efforts by studying how to annotate small resources while selectively leveraging large resources developed for another language.

Most of existing competitive models for QA systems build on neural attention mechanism, to guide systems to focus on a targeted area in the passage. A baseline we adopt in this category is BiDAF model (Seo et al., 2016) which employs variant co-attention mechanism to match the question

Table 1: Statistics of dataset

	Human dataset	Translated dataset	Total
# of passages	1.4K	19.8K	21.2K
# of Q-A pairs	4K	77K	81K
Avg length of question	6.10	5.86	-
Avg length of answer	3.69	1.81	-

and passage mutually. Alternatively, R-Net adopts a gated attention-based recurrent network, with the gate modeling the importance of passage parts to the particular question, as such importance can differ for reading comprehension and question answering purposes respectively.

3. Model

3.1. Preliminary

As a basic QA model, we adopt Bi-Directional Attention Flow model (BiDAF) (Seo et al., 2016), with the highest accuracy for SQuAD among open-sourced implementations.[1] Two metrics are used to evaluate models: Exact Match (EM) and a softer metric, F1 score, which measures the weighted average of the precision and recall rate at word level. This model (by itself without ensemble) achieves an **EM score of 68.0 and an F1 score of 77.3**.

The inputs of this model are passage and question representation as both character- and word-level embeddings. They pass through bi-directional LSTM with attention to obtain a query-aware context representation. The output layer of the model is the probability distribution of the start and the end indices of the answer in the passage. The probabilities are calculated using the equation below:

$$P(y_1 = s) = softmax(w_1 * [G; M]) \quad (1)$$

$$P(y_2 = e) = softmax(w_2 * [G; LSTM(M)]) \quad (2)$$

where s and e indicate the start and the end position of answer span, G is the output of attention flow module according to passage, M is the output that is passed through two layers of LSTM according to the G, w_1 and w_2 are the weight of each output. In test, the model selects the answer span to maximize the product of two probabilities:

$$P(y_1 = s) * P(y_2 = e) \quad (3)$$

We will use this score to check translation quality of QA pairs in Section 3.3.

3.2. Dataset Construction

Our dataset consists of **MT** and **Seed**, representing machine-translated (large-scale) and manually annotated (small-scale) datasets respectively. Some statistics of the datasets are presented in Table 1.

3.2.1. MT Construction

For addressing **RQ1**, we study the challenge of translating answer spans in English into those in Korean translation. From observing Google Translate[2] of SQuAD QA pairs into Korean, we identify the following four cases:

[1] https://github.com/allenai/bi-att-flow

[2] https://translate.google.com

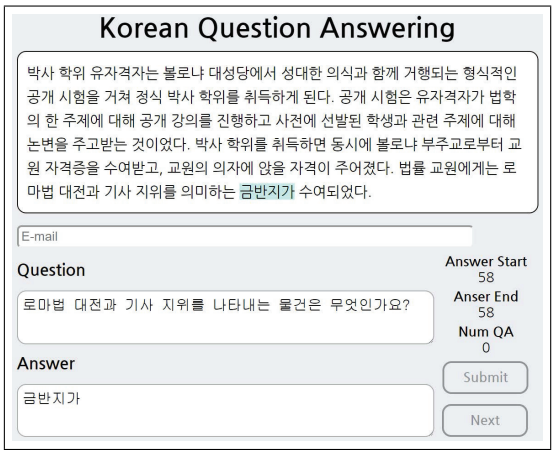


Figure 1: The web interface used to collect the Korean QA dataset. The labeling policy is based on SQuAD (Rajpurkar et al., 2016)

- Exact matching (35.5%): Terms in ESP (English answer spans) are translated into Korean terms which exactly match the Korean terms translating the matching English passage.
- Paraphrase matching (36.6%): Terms in KSP (Korean translated answer spans) are the paraphrase of terms used in Korean passage.
- Multiple spans (8.1%): Multiple ESPs are marked for a single sentence.
- Spans unpreserved: Google Translation cannot preserve answer spans in translation due to the language gap or translation inaccuracy.

To keep the first three cases (80.2% in total of SQuAD pairs) as MT resources, we adopt the following strategies. First case of exact matching can be trivially supported by string matching. For the second case, we mark answer spans in quotation, to serve the dual purposes of (1) giving a hint to translators to preserve the span boundary and (2) finding a matching paraphrase. For the third case, we preprocess QA pairs, to detach k ESPs on the same sentence into k pairs of one ESP and the sentence. Lastly, answer spans may be lost in translation, for which case, we do not use as training resources.

3.2.2. Seed Construction

The performance of QA, trained only on **MT**, is not effective with F1 score of 52.49, while that of English is 77.30. The main obstacle is poorly translated QA pairs, as shown in Figure 2(b), from which, neither machine nor human can find the right answer.

We thus build a small-scale **Seed** resources to serve dual purposes of (a) training a weak QA system and (b) predicting the translation quality of machine translated data. The advantage of **Seed** is near perfect precision, with the disadvantage of being labor intensive.

Our key contribution is to show that **Seed** does not have to be big and annotate such data ourselves. Unlike SQuAD

Passage : Napoleon maintained strict, efficient work habits, prioritizing what needed to be done. {…} Critics said he won many battles simply because of **luck**.

Question : According to critics, what was the reason Napoleon won many battles?

Passage : 나폴레옹은 엄격하고 효율적인 작업 습관을 유지하면서 해야 할 일에 우선 순위를 매겼습니다. {…} 비평가들은 단순히 **행운때문에** 많은 전투에서 승리했다고 말했다.

Question : 비평가들에 따르면, 나폴레옹이 많은 전투에서 승리 한 이유는 무엇입니까?

Certainty: 0.994

(a) example of excellent translation

Passage : During **World War II**, the island (Norfolk Island) became a key airbase and refuelling depot between Australia and New Zealand, and the Solomon Islands.

Question : During what major event did Norfolk Island become an important airbase and refuelling station?

Passage : **2 차 세계 대전기간** 동안 이 섬은 호주와 뉴질랜드, 뉴질랜드와 솔로몬 군도 사이의 주요 기지와 연료 보급 기지가 되었습니다.

Question : 노퍽 섬은 중요한 행사가 진행되는 동안 중요한 공군 기지 및 연료 보급소가 되었습니까?
Human re-translation : Did Norfolk Island become a major air base and fuel depot during important events?

Certainty: 0.125

(b) example of bad translation

Figure 2: Two examples of certainty of Korean QA pairs. Texts colored in red indicate answer span.

dataset containing 100K+ QA pairs, we show that the annotation of 4K is sufficient and release our annotation for future research[3].

To obtain seed QA pairs with comparable quality to SQuAD, we deploy a UI as shown in Figure 1, inspired by SQuAD conventions. We sample the 100 articles of Korean Wikipedia, to extract paragraphs of significant length without images. The result was 1464 paragraphs for the 100 articles covering a wide range of topics.

The UI in Figure 1 was deployed to human annotators, to read paragraph, enter questions, then highlight the spans including answers. Verbatim copying of Wikipedia text was discouraged, by disabling copy and pasting.

3.3. Translated Data Refinement

With **Seed**, we now study how to complement each other, rather than simply merge the two datasets. If mistranslated QA pairs are added to training set, the performance of the model may be worse, therefore we selectively denoise **MT** data as the quality of translation.

For such selection, we quantify **translation certainty**, as shown in Figure 2, high for correct translation (Figure 2(a)) and low for incorrect translation (Figure 2(b)). We use the weak QA system, or BiDAF built on **Seed** for such prediction. A naive solution is using Equation (3): Based on the starting and end positions s and e, we use the equation to compare with the probabilistic certainty of the span predicted by the model, or BiDAF on the translated pair. This score will be high if our weak QA finds the right answer, which might be near impossible without reasonable translation quality. In other words, selecting QA pairs with high score would ensure high translation quality of the data.

However, this computation is rather strict, disallowing a minor disagreement in span boundaries, of including one more (or less) word before and after the span. We thus extend the equation to tolerate a minor error of one word, in both the start and end, or s and e, as below:

$$\Big(P(y_1 = s-1) + P(y_1 = s) + P(y_1 = s+1)\Big) * \Big(P(y_2 = e-1) + P(y_2 = e) + P(y_2 = e+1)\Big) \quad (4)$$

Figure 2 shows the example scores this model computed: For example, in Figure 2(a) with high certainty, the passage and question were well translated into Korean, and a human can easily deduce correct answer through only this information. While, in Figure 2(b), the intention of the question was lost in translation, and the answer cannot be inferred from these question. This QA pair will not be used in training, guided by the low certainty score computed from the model.

4. Experiment

4.1. Experiment Setting

For test, we partitioned the annotated QA pairs randomly into a training set (2K) and a test set (2K), where the two sets do not share the same passages and articles. The implementation details used for this task are based on that of BiDAF model (Seo et al., 2016). We set a mini-batch size of 60 for 10 epochs on GPU Titan X, and a dropout rate of 0.5. Other hyper-parameters are the same with BiDAF model. For Korean-specific implementation of tokenizer and embedding, we adopt the state-of-the-arts KoNLPY[4] and skip-gram model (Mikolov et al., 2013) trained on Korean Wikipedia corpus and QA dataset.

4.2. Models

We compare our model with the following baselines:

- **Seed**: BiDAF using only manually annotated seed resources of a small scale.
- **MT**: BiDAF using only machine translated resources of a large scale.
- **Seed+Rand**: Hybridization of Seed and MT, by running BiDAF on Seed and $x\%$ of **Randomly** selected QA pairs from MT. For example, S+MT(25%) is the results of running BiDAF on Seed and 25% randomly selected QA pairs.

[3] https://e05087.github.io/

[4] http://konlpy-ko.readthedocs.io/

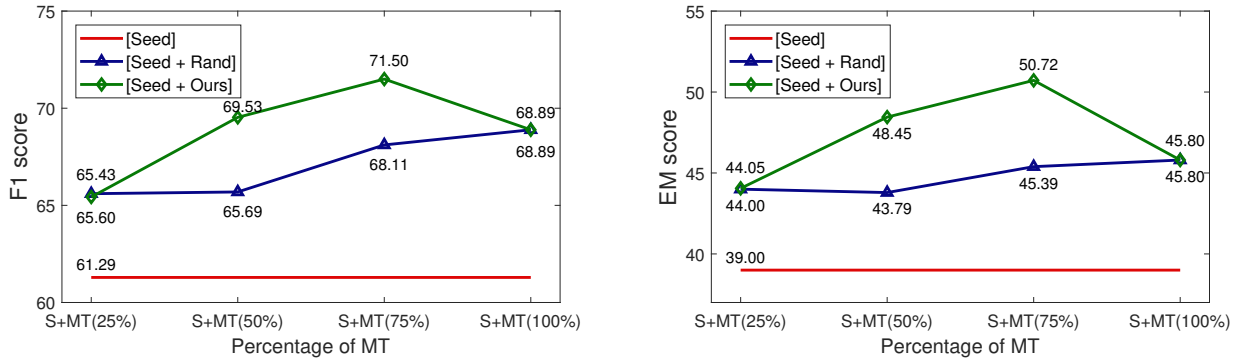


Figure 3: From left to right, the result of (a) F1 score and (b) EM score

Table 2: The results when using only Seed/MT, and ours which is the highest performance in our results.

Metric	F1 score	Exact Match
Seed	61.29	39.00
MT	52.49	10.13
Ours	71.50	50.72

Our proposed model prioritizes high-quality QA pairs from MT, and thus enables a **prioritized** selection of **MT** QA pairs. In this model, we use the techniques discussed in Section 3.3 to decide $x\%$ of the highest translation quality.

4.3. Result and Discussion

The results are reported in Table 2. In terms of both F1 and Exact Match (EM) scores, using only **Seed** or **MT** resources show poor performances, such as 61.29, 52.49 in F1 and 39.00, 10.13 in EM, respectively. In such case, the result on **Seed** is higher than that on **MT**, even though **Seed** is a small size compared to **MT**. This supports the need to build language-specific resource.

Meanwhile, as shown in Figure 3, using both **Seed** and **MT** set improves accuracy. As we add a random sampling of **MT** 25% set per each step, its performance also increases, and peaks at 68.89 in F1 and 45.80 in EM when $x = 100\%$. In the random sampling, the performances of F1 and EM tend to increase as the size of training data. Compared to the random sampling, our model, by refining **MT**, peaks at 71.50 in F1 and 50.72 in EM when $x = 75\%$. The peak value is higher than the result value at $x = 100\%$. This means that excluding refined 25% supports the improvement of the QA model. In all $x\%$ except at 25%, prioritized selection outperforms random selection. Although we use the same dataset and model, the prioritization of our approach enables a significant improvement, of F1 2.61 points increase. Therefore, our model is successful in effectively using small seed and large translation data to improve the performance of the QA model.

5. Conclusion

In this work, we study the feasibility of using translated resources for training QA systems. Inspired by our observations of challenges in using such translated resources for the task, we then build and release a 4K seed QA training resources for Korean language. We then study how we can combine such seed resources with the selective translated resources, for which we propose a model quantifying the translation certainty for the selective use of high quality resources. Lastly, we study the performance of QA systems on this combination of translated and seed resources. This release of seed resource and the proposed method of combining seed with large-scale resources available for another language is useful for follow-up research for providing QA services on many languages.

6. Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2017-2016-0-00464-002) supervised by the IITP(Institute for Information communications Technology Promotion)

7. References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Balahur, A. and Turchi, M. (2014). Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech & Language*, 28(1):56–75.

Dandapat, S. and Way, A. (2016). Improved named entity recognition using machine translation-based cross-lingual information. *Computación y Sistemas*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (). Imagenet: A large-scale hierarchical image database. In *CVPR 2009. IEEE Conference on*. IEEE.

Dunn, M., Sagun, L., Higgins, M., Guney, U., Cirik, V., and Cho, K. (2017). Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*.

Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., and Deng, L. (2016). Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Seo, M., Kembhavi, A., Farhadi, A., and Hajishirzi, H. (2016). Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Ture, F. and Boschee, E. (2014). Learning to translate: a query-specific combination approach for cross-lingual information retrieval. In *EMNLP*, pages 589–599.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, pages 2013–2018.

PhotoshopQuiA: A Corpus of Non-Factoid Questions and Answers for Why-Question Answering

Andrei Dulceanu[†*], Thang Le Dinh[‡], Walter Chang[*], Trung Bui[*], Doo Soon Kim[*], Manh Chien Vu[‡], Seokhwan Kim[*]

[†]Universitatea Politehnica București, România

[*] Adobe Research, California, United States

[‡]Université du Québec à Trois-Rivières, Québec, Canada

andrei.dulceanu@cs.pub.ro, {wachang, bui, dkim, seokim}@adobe.com, {thang.ledinh, manh.chien.vu}@uqtr.ca

Abstract

Recent years have witnessed a high interest in non-factoid question answering using Community Question Answering (CQA) web sites. Despite ongoing research using state-of-the-art methods, there is a scarcity of available datasets for this task. Why-questions, which play an important role in open-domain and domain-specific applications, are difficult to answer automatically since the answers need to be constructed based on different information extracted from multiple knowledge sources. We introduce the PhotoshopQuiA dataset, a new publicly available set of 2,854 why-question and answer(s) (WhyQ, A) pairs related to Adobe Photoshop usage collected from five CQA web sites. We chose Adobe Photoshop because it is a popular and well-known product, with a lively, knowledgeable and sizable community. To the best of our knowledge, this is the first English dataset for Why-QA that focuses on a product, as opposed to previous open-domain datasets. The corpus is stored in JSON format and contains detailed data about questions and questioners as well as answers and answerers. The dataset can be used to build Why-QA systems, to evaluate current approaches for answering why-questions, and to develop new models for future QA systems research.

Keywords: question answering, community question answering, non-factoid question, Why-QA

1. Introduction

The success of IBM's Watson in the *Jeopardy!* TV game-show in 2011 and the significant investments of large tech companies in building personal assistants (e.g., Microsoft Cortana, Apple Siri, Amazon Alexa or Google Assistant) have strengthened the interest in the Question Answering field. These systems have in common the fact that they mostly tackle **factoid questions**. These are questions that "can be answered with simple facts expressed in short text answers"; usually, their answers include "short strings expressing a personal name, temporal expression, or location" (Jurafsky and Martin, 2017). An example of a factoid question and its answer is:

Q: *Who is Canada's prime minister?*
A: *Justin Trudeau.*

By contrast, **non-factoid questions** ask for "opinions, suggestions, interpretations and the like"(Tomasoni and Huang, 2010). Answering and evaluating the quality of the provided answers for non-factoid questions have proved to be non-trivial due to the difficulty of the task complexity as well as the lack of training data. To address the latter issue, numerous researchers have tried to take advantage of user-generated content on Community Question Answering (CQA) web sites such as Yahoo! Answers [1], Stack Overflow [2] or Quora [3]. These web forums allow users to post their own questions, answer others' questions, comment on others' replies, and upvote or downvote answers. Usually, if a user is the original questioner, he/she is allowed to select the most relevant answer to his/her question. Although CQA web sites have lots of experts, it still takes their time to give pertinent, authoritative answers to user questions and not all the content shares the same characteristics. Some key differences (Blooma and Kurian, 2011) in answer quality and availability between traditional QA systems and CQA web sites include: the type of questions (factoid vs. non-factoid), the quality of the answers (high vs. varying from answerer to answerer) and the response time (immediate vs. several hours or days).

Among all categories of non-factoid questions, namely list, confirmation, causal and hypothetical (Mishra and Jain, 2016), we are especially interested in why-questions that are related to causal relations. Why-questions are difficult to answer automatically since the answers often need to be constructed based on different information extracted from multiple knowledge sources. For this reason, why-questions need a different approach than factoid questions because their answers usually cannot be stated in a single phrase (Verberne et al., 2010).

A why Question Answering (Why-QA) system trying to answer questions using CQA data needs to be able to distinguish between relevant and irrelevant answers (answer selection task). Most of the time these systems also produce a sorted output of relevant answers (answer re-ranking task). Both tasks require curated and informative datasets on which to evaluate proposed methods.

In this paper, we introduce the PhotoshopQuiA dataset, a corpus consisting of 2,854 (WhyQ, A) pairs covering various questions and answers about Adobe Photoshop [4]. We

[1] https://answers.yahoo.com
[2] https://stackoverflow.com
[3] https://www.quora.com

[4] Adobe Photoshop is the *de facto* industry stan-

chose Adobe Photoshop because it is a popular and well-known product, with a lively, knowledgeable and sizable QA community. PhotoshopQuiA is the first large Why-QA only English dataset that focuses on a product, as opposed to previous open-domain datasets. Our dataset focuses on why-questions that occur while a user is trying to complete a task (e.g., changing color mode for an image, or applying a filter). It contains contextual information about the answer, which in turn makes it easier to build a QA system that is able to find the most relevant answers. We named the corpus PhotoshopQuiA, because *quia* (first and last letters capitalized as in Question Answering) means *because* in Latin and therefore hints at the expected why-answer type.

One of the challenges that often arises with CQA data is the high variance in quality for both questions and answers. To address this problem, we include in our dataset mostly official answers (65.5% from total pairs) given by Adobe Photoshop experts. We choose to provide both text and HTML representations of questions and answers because the raw HTML snippets often include relevant information like documentation links, screenshots or short videos which would be otherwise lost. We analyze the (WhyQ, A) pairs for presence of certain linguistic cues such as causality markers (e.g., *the reason for*, *because* or *due to*).

2. Related Work & Datasets

In recent years, numerous datasets have been released in the domain of question-answering (QA) systems to promote new methods that integrate natural language processing, information retrieval, artificial intelligence and knowledge discovery. The majority of these datasets were open-domain (Bollacker et al., 2008; Ignatova et al., 2009; Cai and Yates, 2013; Yang et al., 2015; Chen et al., 2017). There are still a few QA datasets for specific fields such as BioASQ and WikiMovies. The BioASQ dataset contains questions in English, along with reference answers constructed by a team of biomedical experts (Tsatsaronis et al., 2015). The WikiMovies dataset contains 96K question-answer pairs in the domain of movies (Miller et al., 2016). Table 1 introduces selected recent open-domain QA datasets.

Several existing datasets focus on the data taken from CQA web sites. The data structure of our dataset (question-answer(s) pair with the best answer labeled), resembles the one in (Hoogeveen et al., 2015). However, it does not include comments and tags, making it more suitable for Why-QA than previous structures which include only questions (Iyer et al., 2017). Our work is mostly related to the SemEval-2016 Task 3 dataset (Nakov et al., 2016) which contains more than 2,000 questions associated with ten most relevant comments (answers). It also shares some characteristics with Yahoo's Webscope[5] L4 used by (Surdeanu et al., 2008) and (Jansen et al., 2014), L6 and with nfL6[6], a non-factoid subset of L6 focusing only on how-questions (Cohen and Croft, 2016). Although Yahoo Webscope L6 certainly contains many why-QA pairs which should be fairly trivial to extract from the entire dataset, we believe this limits its usefulness for building Why-QA systems. While all these datasets focus on CQA forums data, there are some key differences between our work and existing datasets (Table 2).

Dataset	Description
WebQuestions and Free917	for training semantic parsers, which map natural language utterances to denotations (answers) via intermediate logical forms (Berant et al., 2013)
CuratedTREC	2,180 questions extracted from the datasets from TREC (Baudiš and Šedivỳ, 2015)
WikiQA	3,000 questions sampled from Bing query logs associated with a Wikipedia page presumed to be the topic of the question (Yang et al., 2015)
30M Factoid QA Corpus	30M natural language questions in English and their corresponding facts in the knowledge base Freebase (Serban et al., 2016)
SQuAD	100,000 question-answer pairs on more than 500 Wikipedia articles (Rajpurkar et al., 2016)
Amazon	1.4 million answered questions from Amazon (Wan and McAuley, 2016)
Baidu	42K questions and 579K evidences, which are a piece of text containing information for answering the question (Li et al., 2016)
Allen AI Science Challenge	2,500 questions. Each question has 4 answer candidates (Chen et al., 2017)
Quora	Over 400K sentence pairs of which, almost 150K are semantically similar questions; no answers are provided (Iyer et al., 2017)

Table 1: Recent datasets for QA systems

Difference	This work	Previous work
Scope	closed domain (focus on product usage)	open domain
Answer authority	picked by a domain expert (65.5%)	n/a
Question types	why-questions only	various (focus on how-questions)

Table 2: Major differences between our dataset and previous datasets focusing on CQA forums data

The difference in scope allows researchers to verify

dard raster graphics editor developed and published by Adobe Systems for macOS and Windows. https://en.wikipedia.org/wiki/Adobe_Photoshop

[5] https://webscope.sandbox.yahoo.com/catalog.php?datatype=l

[6] https://ciir.cs.umass.edu/downloads/nfL6

whether all previous research achievements made on open domain datasets such as Yahoo Webscope could be translated into a closed domain such as ours, or whether domain adaptation is needed. The authors are aware of the corpora from (Prasad et al., 2007) and (Dunietz et al., 2017), but these were not considered for this work, because their datasets do not address CQA and/or Why-QA.

Some of the previous studies in Why-QA systems tried to extract why-questions from QA datasets related to general questions; however, the size and quality of why-questions were limited. Previous datasets used in Why-QA task contain few (WhyQ, A) pairs (under 1,000), are handcrafted, are not available online anymore (Verberne et al., 2007; Mrozinski et al., 2008; Higashinaka and Isozaki, 2008) or target Japanese (Higashinaka and Isozaki, 2008; Oh et al., 2012). There is a need for a public specific why-question dataset for English to advance the research and development in Why-QA.

3. PhotoshopQuiA Dataset

In this section we describe the process of creating the PhotoshopQuiA dataset and succinctly compare our data collection approach to previous related approaches.

3.1. Data Sources

We identified the following five web sites as appropriate sources for collecting why-questions about Adobe Photoshop: Adobe Forums [7], Stack Overflow, Graphic Design [8], Super User [9] and Feedback Photoshop [10]. Although there were additional CQA web sites containing Photoshop-related questions and answers, we selected only the sources above because they all have moderated, recent, high-quality and authoritative content. Regarding the last two points, it is worth mentioning that the dataset contains a high ratio of answers coming from Adobe experts working in the Photoshop team (65.5% of total answers).

When using one of the above-mentioned forums, the original questioner has the possibility to select the most relevant answer to his/her question. This is often referred as the accepted answer. If such an answer does not exist, does not fully meet established criteria or even does not solve the problem at hand, registered users may upvote or downvote it. As stated previously, PhotoshopQuiA includes all answers available for each why-question, labeling the correct answer. We strove to always label as correct accepted answers only, but when such answers were not available, we selected the most voted answer instead. If the most voted answer had at least one downvote, we did not include the (WhyQ, A) pair altogether.

Our approach resembles previous datasets described in related work section, where a dataset item contained either the full conversation thread, (Hoogeveen et al., 2015)(although we did not include tags or comments), or multiple relevant answers for a question (Nakov et al., 2016). One of the key advantages of our approach is that our answers are more reliable and correct - two thirds of the (WhyQ,A) pairs have an accepted answer authored by a Photoshop expert.

3.2. Web Crawling

We used Scrapy [11], an application framework written in Python for our web scraping task. Web scraping includes two main stages: web crawling (i.e., fetching or downloading a web page) and data extraction (i.e., extracting structured content from a fetched page). In order to successfully scrape a CQA web site, Scrapy needs a `Spider` definition containing the following:

- an initial list of URLs which the `Spider` will begin to crawl from. This is provided in `start_urls` attribute or via `start_requests()` method.
- an implementation of the default `parse()` callback method, which is a generator function under the hood. This method is accountable for all the heavy lifting needed for processing the response corresponding to each request. When all content is available on the requested web page, it yields a Python dictionary filled with data of interest. If additional requests need to be performed (i.e. following a new URL found in the page), it yields a new request which in turn needs
- an implementation of a new, custom, user defined callback method for parsing the new response. This method will yield the final scraped items.

All Scrapy requests are processed and scheduled asynchronously, enabling fast concurrent crawls. In order to politely scrape mentioned CQA web sites, we overrode Scrapy default settings and limited the number of concurrent requests per IP to one, with a five seconds download delay between each request.

After taking a look at the building blocks of a Scrapy `Spider`, few words should be mentioned about how the actual item scraping works in the `parse()` callback. This is done in two phases: selection and extraction. Selection employs CSS selectors for selecting HTML elements in the response. Once the desired elements are selected, XPath expressions can be used for a more fine-grained control of the extracted content.

3.3. Data Collection

The same steps described below were followed for each CQA web site considered. For brevity, we only describe the full workflow used for scraping Stack Overflow. We first manually performed a search containing "why Photoshop" keywords, quotes excluded. In the next step, we used resulting URL and number of result pages to handcraft the `start_urls` list. For example, performing a search on Stack Overflow using our search query and clicking the first result page at the bottom resulted in the following link: `https://stackoverflow.com/search?page=1&tab=Relevance&q=why%20Photoshop`. We iterated over the number of result pages returned by the search

[7] `https://forums.adobe.com/welcome`
[8] `https://graphicdesign.stackexchange.com`
[9] `https://superuser.com`
[10] `https://feedback.photoshop.com`
[11] `https://scrapy.org`

and included the appropriate URL (page number changed) in `start_urls`.

In the `parse()` method we iterated over each search result on the page and created a new `WhyQuestionAnswerPair` item. This item along with the actual URL of the search result were passed in turn to `parse_question_answer_pair()` callback. This method extracted all the information needed from the new URL and returned the populated item. Finally, each result item was appended to the consolidated JSON file.

3.4. Annotation

We retain the following data and meta-data for each (WhyQ, A) pair: question identifier on the source web site, question URL, question title, question date, questioner, questioner level, question state (open or resolved), full question text, full question HTML and for each answer available: answer date, answerer, answerer level, answer votes, answer text, full answer HTML and a boolean property indicating if this is the best answer or not. Beautiful Soup[12] was used for extracting question and answer(s) text from HTML; links provided inside raw snippets were enclosed in parentheses and kept in the final text excerpt.

Since almost all previous properties are self-explanatory we will give more insight into questioner/answerer level properties. Depending on the source web site these properties can either be a text describing user seniority/level on the web site (e.g., "Level 1", "Adobe Community Professional", etc. for Adobe Forums), a number describing the number of posts written so far (Feedback Photoshop) or the reputation score of the user (Stack Overflow and the like). We treat question identifier, questioner level, answerer level and answer votes as optional properties. All other properties not listed as optional are required. We enforce these constraints and validate our dataset by using a JSON schema [13]. A sample (WhyQ,A) pair from our corpus is available in the appendix.

3.5. Post-Processing

After the web scraping phase, we ended up with a consistently larger number (4,365 pairs obtained) of (WhyQ,A) answer pairs than those included in our final dataset (2,854 pairs). We went further and refined these pairs in a two-stage manual cleanup process in which we removed duplicates, questions which mentioned Adobe Photoshop incidentally and questions containing different question types (e.g., "how to", "definition") than our intended why-question type.

3.6. Availability & Reuse

The dataset, its JSON schema and a copy of this paper are available on *GitHub*[14] under cc by-sa 3.0 with attribution required license[15]. We also plan to add to that repository the Python scripts used for web scraping and manipulating the data.

4. Data Statistics & Analysis

The statistics of the PhotoshopQuiA dataset are described in Table 3. When crunching the data we ignored 13 (WhyQ,A) pairs with long question content. The questioners posted OS- and Photoshop-related configurations which artificially increased the numbers.

Metric	Value
Avg. no. of words in question title	10.69
Avg. no. of words in question text	102.29
Avg. no. of words in best answer	95.60

Table 3: Statistics of the PhotoshopQuiA dataset

Causality, causation or **causal relation** is "the relation between a cause and its effect or between regularly correlated events or phenomena"[16]. Words connecting the cause and effect parts of a causal relation are called **causal markers**. As outlined by (Girju, 2003), causative constructions can be explicit (introduced by causal markers like *cause*, *effect*, *consequence*, etc.) or implicit (i.e., without any explicit marker). Blanco et al. (2008) refine causal relations categories by introducing ambiguity (if the causal marker does not always signal a causation, e.g., *since*) and completeness (when both cause and effect parts are present).

For our analysis, we are interested in explicit causal markers, ignoring ambiguity and completeness of the causal relations. To come up with an extended list of causal markers, we started with the adverb *therefore* and looked for its synonyms in the Oxford Thesaurus of English. We found that causality markers are present in 1,583 answers out of 2,854 total answers (55.46%). Figure 1 shows the distribution of the 22 markers found. A future study needs to be done to remove the ambiguous markers.

To better visualize the diversity of the questions in the dataset, we include in Figure 2 a breakdown of questions based on the first wh-word contained in question title.
When crunching the numbers behind these statistics, we noticed that 239 questions (8.37% of the dataset) have multiple wh-words in their title. A distinct category emerged from questions asking for OS specific details or solutions, totaling 286 questions, or 10.26% of the dataset. Moreover, 763 questions (accounting for 26.73% of the total questions) contained the adverb *not*. The key takeaway here is that the two most important features when it comes to question titles are the presence of the word *why* and of the negation. At first, it might seem odd that 9.76% of the questions with at least one wh-word contain the word *how* and are still considered why-questions. An example of such a question is:

Q Title: *Photoshop: how to change from RGB to CMYK without any color loss?*
Q Content: *[...] When converting this from RGB to*

[12] `https://www.crummy.com/software/BeautifulSoup`
[13] `http://json-schema.org`
[14] `https://github.com/dulceanu/photoshop-quia`
[15] `https://creativecommons.org/licenses/by-sa/3.0/`
[16] From Merriam-Webster's Online Dictionary

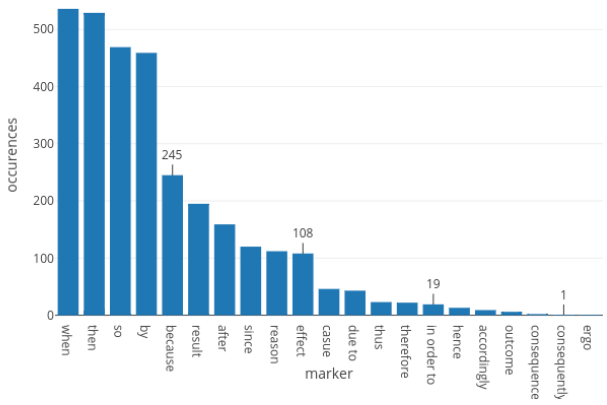


Figure 1: Distribution of explicit causality markers found in PhotoshopQuiA (1,583 answers)

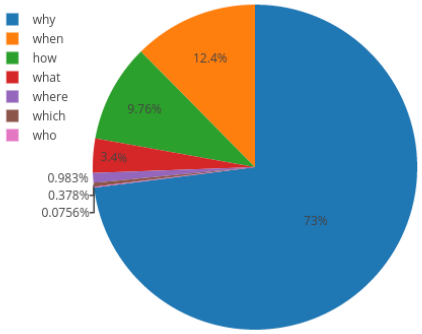


Figure 2: Distribution of questions based on first occurrence of a wh-word in question title (1,322 questions)

CMYK, my original colors are changing. [...]
A: *By changing color mode you essentially change colors. [...] Because of that, [...] the colors you get [...] will be made as close to original as possible - but not identical.[...]*

As it can be seen, the questioner was not interested in a general recipe about changing the color mode of his/her document, but rather wanted to do it under special circumstances (i.e. without color loss). In other words, this question is equivalent to a why-question like *Why do I have a color loss when changing from RGB to CMYK?*. Often, people use *how*, usually followed by *come*, to informally ask for causes of events, as outlined by this example from Merriam-Webster's Online Dictionary: *How come you can't go?*.

A detailed classification of the questions containing the word *why* in their title is included in Figure 3. The numbers are presented again in percentages and the categories are not mutually exclusive. In orange we present the percent of why-questions in that sub-category which have also the adverb *not* in their title. Noteworthy here is the strong connection between the modal *can* and the negation (e.g., *Why can't I amend my mask with the brush tool ?*) and between the auxiliary *will* and the negation (e.g., *Why won't Photoshop let me rename my layers?*).

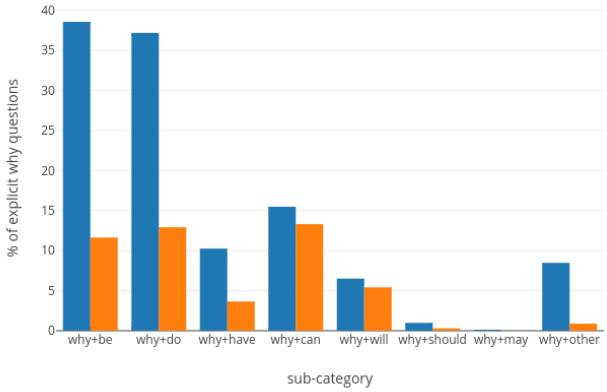


Figure 3: Explicit why questions sub-categories (1,006 questions)

5. Expected Use

Below we suggest some use cases of tasks that we are currently working on:
- Question Classification: what classification models for why-questions can be used to facilitate the tasks of Why-QA systems and how to classify the why-questions based on the selected classification model.
- Semantic parsing: how to convert a why-question into a structured format (or a canonical form) to speed up the matching process.
- Question to question matching: what approaches can be used to match the asked question and the questions in the knowledge base (information retrieval, bag-of-word, knowledge based approach, or neural network).

6. Conclusion and Future Work

In this paper, we present PhotoshopQuiA, a new dataset for Why-QA. The dataset is constructed in a natural and practicable manner so that it can be used to observe different characteristics and behaviors of the why-questions.

We believe that the PhotoshopQuiA dataset enables new research in Why-QA, which has received less focus, and that our work stimulates further research to advance the QA technology needed for smart services such as recommendation systems, chatbots, and intelligent assistants.

7. Acknowledgements

The authors express their sincere thank to the University Gift Funding of Adobe Systems Incorporated for the partial financial support for this research.

Appendix: JSON format of the PhotoshopQuiA dataset


```
"id": 20518335,
"url": "https://stackoverflow.com/questions/20518335/why-do-photoshop-files-start-with-8bps",
"title": "Why do Photoshop files start with 8BPS?",
"questionDate": "2013-12-11T11:48:22Z",
"questioner": "Party Ark <https://stackoverflow.com/users/548664/party-ark>",
"questionerLevel": "532",
"questionState": "resolved",
"questionText": "From pretty much the beginning of time Photoshop files have begun with 8BPS. (I have verified this back to version 2.5) It must have had some meaning at some point.\nThe 8B I thought might refer to bits/channel, but it makes no difference saving it 16 or 32. PS is probably PhotoShop, but might not be. Something to do with the way Mac saved files?",
"questionHtml": "<div class=\"post-text\" itemprop=\"text\">\r\n\r\n<p>From pretty much the beginning of time Photoshop files have begun with 8BPS. (I have verified this back to version 2.5) It must have had some meaning at some point.</p>\n\n<p>The 8B I thought might refer to bits/channel, but it makes no difference saving it 16 or 32. PS is probably PhotoShop, but might not be. Something to do with the way Mac saved files? </p>\n    </div>",
"answers": [
      {
      "answerDate": "2015-02-20T22:31:45Z",
      "answerer": "Community <https://stackoverflow.com/users/-1/community>",
      "answererLevel": "1",
      "answerVotes": 4,
      "answerText": "8B is shorthand for Adobe. I guess \"eight bee\" sounds a little bit like Adobe; more so if you're Italian - \"Otto Bee\".\nAnd PS is \"Photoshop\". So, 8BPS is \"Adobe Photoshop\".\n8B crops up in quite a few places in Adobe file extensions or internal types. Wikipedia has a list (http://en.wikipedia.org/wiki/8B).",
      "answerHtml": "<div class=\"post-text\" itemprop=\"text\">\r\n<p><code>8B</code> is shorthand for Adobe. I guess \"eight bee\" sounds a <em>little</em> bit like Adobe; more so if you're Italian - \"Otto Bee\".</p>\n\n<p>And <code>PS</code> is \"Photoshop\". So, <code>8BPS</code> is \"Adobe Photoshop\".</p>\n\n<p>8B crops up in quite a few places in Adobe file extensions or internal types. <a href=\"http://en.wikipedia.org/wiki/8B\" rel=\"nofollow\">Wikipedia has a list</a>.</p>\n    </div>",
      "bestAnswer": true
      },
      {
      "answerDate": "2013-12-11T11:53:19Z",
      "answerer": "Syjin <https://stackoverflow.com/users/733368/syjin>",
      "answererLevel": "1,922",
      "answerVotes": 0,
      "answerText": "That is just the Signature to identify the file as a Photoshop file. From the Specification:\nSignature: always equal to '8BPS' . Do not try to read the file if the signature does not match this value.\nSee the Photoshop File Format Specification (http://www.adobe.com/devnet-apps/photoshop/fileformatashtml/#50577409_pgfId-1036097) for more detailied information.",
      "answerHtml": ... truncated due to space constraints...,
      "bestAnswer": false
      }
    ]
}
```


Listing 1: An example of an item containing all properties from PhotoshopQuiA dataset

8. Bibliographical References

Baudiš, P. and Šedivỳ, J. (2015). Modeling of the question answering task in the yodaqa system. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 222–228. Springer.

Berant, J., Chou, A., Frostig, R., and Liang, P. (2013). Semantic parsing on freebase from question-answer pairs. In *EMNLP*, volume 2, page 6.

Blanco, E., Castell, N., and Moldovan, D. I. (2008). Causal relation extraction. In *Lrec*.

Blooma, M. J. and Kurian, J. C. (2011). Research issues in community based question answering. In *PACIS*, page 29.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Cai, Q. and Yates, A. (2013). Large-scale semantic parsing via schema matching and lexicon extension. In *ACL (1)*, pages 423–433.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Cohen, D. and Croft, W. B. (2016). End to end long short term memory networks for non-factoid question answering. In *Proceedings of the 2016 ACM on International Conference on the Theory of Information Retrieval*, pages 143–146. ACM.

Dunietz, J., Levin, L., and Carbonell, J. (2017). The because corpus 2.0: Annotating causality and overlapping relations. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 95–104.

Girju, R. (2003). Automatic detection of causal relations for question answering. In *Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering-Volume 12*, pages 76–83. Association for Computational Linguistics.

Higashinaka, R. and Isozaki, H. (2008). Corpus-based question answering for why-questions. In *IJCNLP*, pages 418–425.

Hoogeveen, D., Verspoor, K. M., and Baldwin, T. (2015). Cqadupstack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian Document Computing Symposium*, page 3. ACM.

Ignatova, K., Toprak, C., Bernhard, D., and Gurevych, I. (2009). Annotating question types in social q&a sites. In *Tagungsband des GSCL Symposiums 'Sprachtechnologie und eHumanities*, pages 44–49.

Iyer, S., Dandekar, N., and Csernai, K. (2017). First quora dataset release: Question pairs.

Jansen, P., Surdeanu, M., and Clark, P. (2014). Discourse complements lexical semantics for non-factoid answer reranking. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 977–986.

Jurafsky, D. and Martin, J. H. (2017). *Speech and language processing*. Unpublished Draft, 3rd edition.

Li, P., Li, W., He, Z., Wang, X., Cao, Y., Zhou, J., and Xu, W. (2016). Dataset and neural recurrent sequence labeling model for open-domain factoid question answering. *arXiv preprint arXiv:1607.06275*.

Miller, A., Fisch, A., Dodge, J., Karimi, A.-H., Bordes, A., and Weston, J. (2016). Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

Mishra, A. and Jain, S. K. (2016). A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences*, 28(3):345–361.

Mrozinski, J., Whittaker, E., and Furui, S. (2008). Collecting a why-question corpus for development and evaluation of an automatic qa-system. *Proceedings of ACL-08: HLT*, pages 443–451.

Nakov, P., Màrquez, L., Moschitti, A., Magdy, W., Mubarak, H., Freihat, a. A., Glass, J., and Randeree, B. (2016). Semeval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 525–545, San Diego, California, June. Association for Computational Linguistics.

Oh, J.-H., Torisawa, K., Hashimoto, C., Kawada, T., De Saeger, S., Kazama, J., and Wang, Y. (2012). Why question answering using sentiment analysis and word classes. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 368–378. Association for Computational Linguistics.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). The penn discourse treebank 2.0 annotation manual.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Serban, I. V., García-Durán, A., Gulcehre, C., Ahn, S., Chandar, S., Courville, A., and Bengio, Y. (2016). Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. *arXiv preprint arXiv:1603.06807*.

Surdeanu, M., Ciaramita, M., and Zaragoza, H. (2008). Learning to rank answers on large online qa collections. *Proceedings of ACL-08: HLT*, pages 719–727.

Tomasoni, M. and Huang, M. (2010). Metadata-aware measures for answer summarization in community question answering. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 760–769. Association for Computational Linguistics.

Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M. R., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., et al. (2015). An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

Verberne, S., Boves, L., Oostdijk, N., and Coppen, P.-A. (2007). Evaluating discourse-based answer extraction for why-question answering. In *Proceedings of the*

30th annual international ACM SIGIR conference on Research and development in information retrieval, pages 735–736. ACM.

Verberne, S., Boves, L., Oostdijk, N., and Coppen, P.-A. (2010). What is not in the bag of words for why-qa? *Computational Linguistics*, 36(2):229–245.

Wan, M. and McAuley, J. (2016). Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 489–498. IEEE.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, pages 2013–2018.

BioRead: A New Dataset for Biomedical Reading Comprehension

Dimitris Pappas[1,2], Ion Androutsopoulos[1], Haris Papageorgiou[2]
[1]Department of Informatics, Athens University of Economics and Business,
Patission 76, GR-104 34 Athens, Greece
[2]Institute for Language and Speech Processing, ATHENA Research Center,
Epidavrou & Artemidos 6, GR-151 25 Maroussi, Greece
pappasd@aueb.gr, ion@aueb.gr, xaris@ilsp.gr

Abstract

We present BioRead, a new publicly available cloze-style biomedical machine reading comprehension (MRC) dataset with approximately 16.4 million passage-question instances. BioRead was constructed in the same way as the widely used Children's Book Test and its extension BookTest, but using biomedical journal articles and employing MetaMap to identify UMLS concepts. BioRead is one of the largest MRC datasets, and currently the largest one in the biomedical domain. We also provide a subset of BioRead, BioReadLite, for research groups with fewer computational resources. We re-implemented and tested on BioReadLite two well-known MRC methods, AS Reader and AOA Reader, along with four baselines, as a first step towards a BioRead (and BioReadLite) leaderboard. AOA Reader is currently the best method on BioReadLite, with 51.19% test accuracy. Both AOA Reader and AS Reader outperform the baselines by a wide margin on the test subset of BioReadLite. Our re-implementations of the two MRC methods are also publicly available.

Keywords: BioRead, biomedical, dataset, corpus, evaluation, reading comprehension, question answering, deep learning.

1. Introduction

Machine Reading Comprehension (MRC) systems (Hermann et al., 2015) are given a passage (e.g., from a news article or book) and they are required to answer a question by considering the information of the passage. Manually constructing MRC datasets is very labour-intensive and leads to datasets that may not be large enough to train data-hungry deep learning methods (Goodfellow et al., 2016; Goldberg, 2017). For example, the BioASQ dataset (Tsatsaronis et al., 2015), which was constructed by biomedical experts, currently contains only two thousand questions approximately. The largest manually curated MRC dataset we are aware of, SQuAD (Rajpurkar et al., 2016), which was crowdsourced, comprises approximately 100k passage-question instances.

Larger datasets can be constructed automatically by considering only 'cloze-style' questions, which require filling in a missing word or phrase in a given sentence about the passage (e.g., "___ has been implicated in the pathogenesis of PD."). For example, CBTest (Hill et al., 2015) contains passages from children's books; each cloze-style question is a sentence that follows the corresponding passage in its book, with a randomly selected common noun, named entity, verb, or preposition of the sentence removed and turned into a slot to be filled in. CBTest contains approximately 687k passage-question instances. It was more recently expanded to BookTest (Bajgar et al., 2016), which comprises approximately 14 million passage-question instances, by applying the same methodology to a much larger collection of books. The CNN and Daily Mail datasets (Hermann et al., 2015) were produced in a similar manner. They comprise news articles and cloze-style questions constructed by removing words from sentences summarising the articles; they contain approx. 380k and 880k instances, respectively.

Apart from constituting a testbed for natural language understanding algorithms, MRC is also useful as a component of larger systems. We are interested in a setting where an Information Retrieval engine retrieves document passages that may be relevant to a question, and then MRC is used to identify exact answers (e.g., named entities) in the passages (Sultan et al., 2016; Chen et al., 2017). We focus on the biomedical domain, where this setting is included in the BioASQ challenges (Tsatsaronis et al., 2015). There is currently, however, no sufficiently large publicly available biomedical MRC dataset to train deep learning models. We, therefore, constructed and provide a new biomedical MRC dataset, called BioRead, with approx. 16.4 million cloze-style questions, each paired to a passage and candidate answers. BioRead was constructed in the same manner as CBTest and BookTest, using randomly selected biomedical articles from PubMed Central.[1] To the best of our knowledge, it is currently one of the largest MRC datasets, and the largest one in the biomedical domain. We also provide a subset of BioRead, called BioReadLite, with 900k instances, for groups with fewer computational resources.

We re-implemented (in PyTorch[2]), trained, and tested on BioReadLite two well-known MRC methods, AS Reader (Kadlec et al., 2016) and AOA Reader (Cui et al., 2017). We report their performance, along with the performance of four simpler baselines, as a first step towards a BioRead (and BioReadLite) leaderboard. We open-source the re-implementations to make it easier to replicate our experiments and build upon previous MRC methods.[3]

Automatically generated cloze-style MRC datasets are of lower quality compared to manually constructed ones. For example, Chen et al. (2016) reported that the CNN and Daily Mail datasets contain both questions that are too easy

[1]Consult `https://www.ncbi.nlm.nih.gov/pmc/`.

[2]See `http://pytorch.org/`.

[3]BioRead and the re-implementations will be made available at `http://nlp.cs.aueb.gr/software.html`. The original implementation of AS Reader (in Theano) is available at `https://github.com/rkadlec/asreader/`. The original implementation of AOA Reader does not appear to be online.

	BioRead				BioReadLite			
	Training	**Development**	**Test**	**Total**	**Training**	**Development**	**Test**	**Total**
Instances	∼15,1M	∼600,7k	∼652,9k	∼16.4M	800k	50k	50k	900k
Avg candidates	25.9	27.3	26.3	26.0	18.89	20.8	19.4	19.0
Max candidates	40	40	40	40	30	30	30	30
Min candidates	2	2	2	2	2	2	2	2
Avg context len.	456.9	464.5	455.9	457.1	317.2	320.8	298.9	316.4
Max context len.	999	999	999	999	400	400	400	400
Min context len.	26	56	48	26	30	30	30	30
Avg question len.	33.4	35.5	34.8	33.5	16.8	16.8	16.8	16.8
Max question len.	300	300	300	300	25	25	25	25
Min question len.	5	5	5	5	5	5	5	5

Table 1: Statistics of BioRead and BioReadLite. Lengths in tokens.

to answer using simple hand-crafted features (e.g., simple paraphrases of passage sentences) and questions that even people cannot answer. Nevertheless, BioRead (and BioReadLite) is the only sufficiently large biomedical MRC dataset to train deep learning methods on. In future work, we plan to investigate if systems trained (or pre-trained) on BioRead could also cope (possibly after further training) with real biomedical questions, like those of BioASQ.

2. The BioRead Dataset

To construct BioRead, we randomly selected approx. 90.6k from the approx. 3.4M articles (from approx. 7k biomedical journals) of the Open Access Subset of PubMed Central (PMC).[4] We then applied MetaMap (Aronson and Lang, 2010) to each one of the selected articles.[5] MetaMap recognises words or phrases referring to concepts of the Unified Medical Language System (UMLS).[6] As an example, the words and phrases shown in red or green in the 'context' of the left column of Table 2 were recognised as UMLS concepts. MetaMap also provides the 'preferred name' of each concept. For example, 'carcinoma of the lung', 'lung cancer', and 'malignant tumor of the lung' all refer to the same concept; the preferred name is the first one.

To reduce the size of the vocabulary and avoid confusing MRC methods by synonyms, we replaced each concept that MetaMap recognized by its preferred name. Borrowing the notation of the CNN and Daily Mail datasets (Hermann et al., 2015), each preferred name (possibly multi-token) was then mapped to a pseudo-token of the form @entityID (Fig. 2, right), where ID is an integer identifier that is (a) unique within the particular passage-question instance (i.e., the same ID will generally denote a different concept in another instance), or (b) unique in the entire dataset (the same ID will always denote the same concept). Hermann et al. (2015) ensure that expressions referring to the same entity get the same ID within the same passage-question instance only, which corresponds to option (a). This does not allow systems to learn information about an entity from multiple passages of the training set. By contrast, in option (b), where each entity (concept) has the same ID in the entire dataset, an MRC method may, at least in principle, learn properties of an entity from the passages of multiple training instances. We adopt Hermann et al.'s option (a) in most of our experiments, but we also train the best method with option (b), which improves performance.

As in CBTest and BookTest (Hill et al., 2015; Bajgar et al., 2016), having replaced the recognized entities (concepts) by @entityID pseudo-tokens, we applied a sliding window of 21 sentences to the texts of the approx. 90.6k articles.[7] For each position of the window, we examined each @entityID of the 21st sentence. If an @entityID of the 21st sentence was also present (with the same ID) anywhere in the first 20 sentences, that @entityID was replaced by a @placeholder pseudo-token in the 21st sentence, indicating a slot to be filled in by one of the @entityID tokens of the first 20 sentences; the 21st sentence became a cloze-style question (Fig. 2, right), the first 20 sentences became the 'context' (passage) of the question, the @entityID tokens of the context became the candidate answers, and the particular @entityID that was turned into @placeholder became the correct answer. If multiple @entityID tokens of the 21st sentence were present in the first 20 sentences, multiple context-question-candidates-answer tuples were obtained; hence, strictly speaking each *instance* of BioRead is a context-question-candidates-answer tuple, not just a context and question pair. If no @entityID tokens of the 21st sentence were present in the first 20 sentences, no instance was produced from that position of the window.

To lower the computational resources required to process the dataset, we also set a maximum context length of 999 tokens, a maximum question length of 300 tokens, and a maximum of 40 candidate answers per instance. We discarded instances exceeding these thresholds, obtaining approx. 16.4M instances from the approx. 90.6k articles. BioRead contains these instances, split into training, development, and test sets (Table 1). We also provide a subset of BioRead, called BioReadLite, for researchers with fewer computational resources. BioReadLite was created by setting the maximum context length to 400 tokens, the maximum question length to 25 tokens, and the maximum number of candidate answers to 30. The vocabulary of each dataset includes all the words (and pseudo-tokens) that occurred at least 5 times in the corresponding training subset,

[4] Consult `https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/`. The articles were available in plain text and HTML format; we used the former.

[5] See `https://metamap.nlm.nih.gov/`.

[6] See `https://www.nlm.nih.gov/research/umls`.

[7] We used NLTK's sentence splitter (`http://www.nltk.org/`) and a basic white-space tokeniser.

Context: *salsolinol* (100mg/kg i.p.) or *l-dopa* (100mg/kg i.p.) was acutely administered (100mg/kg i.p.). in the combined treatment group, *l-dopa* (100mg/kg i.p.) was administered once 15min after *salsolinol* administration. the *rats* were decapitated 2h after *injection*. the *concentration* of *dopamine* and its metabolites were measured using *hplc* . the results are expressed as the means sem (n=710 animals per group). the data were analyzed via two-way anova followed by duncans test. statistical significance: [...] *l-dopa* (f[1,27]=26.9, p<0.01) on the level of 3-mt (table1). however, neither *treatment* with *salsolinol* (f[1,27]=0.09, n.s.) nor the interaction between *salsolinol* and *l-dopa* (f[1,27]=0.03, n.s.) was significant (table1).	**Context:** *@entity10* (100mg/kg i.p.) or *@entity5* (100mg/kg i.p.) was acutely administered (100mg/kg i.p.). in the combined treatment group, *@entity5* (100mg/kg i.p.) was administered once 15min after *@entity10* administration. the *@entity6* were decapitated 2h after *@entity0* the *@entity3* of *@entity7* and its metabolites were measured using *@entity9* the results are expressed as the means sem (n=710 animals per group). the data were analyzed via two-way anova followed by duncans test. statistical significance: [...] *@entity5* (f[1,27]=26.9, p<0.01) on the level of 3-mt (table1). however, neither *@entity2* with *@entity10* (f[1,27]=0.09, n.s.) nor the interaction between *@entity10* and *@entity5* (f[1,27]=0.03, n.s.) was significant (table1).
Question: the duncans post hoc test showed that *l-dopa* induced an increase in the *concentration* of 3-mt (by approximately 300%, p<0.01) but that *salsolinol* did not influence this effect of *l-dopa* (table1).	**Question:** the duncans post hoc test showed that *@placeholder* induced an increase in the *@entity3* of 3-mt (by approximately 300%, p<0.01) but that *@entity10* did not influence this effect of *@entity5* (table1).
Candidates: injection, control group, treatment, concentration, substantia nigra, l-dopa, rats, dopamine, dopac, hplc, salsolinol, analysis	**Candidates:** @entity0, @entity1, @entity2, @entity3, @entity4, @entity5, @entity6, @entity7, @entity8, @entity9, @entity10, @entity11
Answer: l-dopa	**Answer:** @entity5

Table 2: An example instance of BioRead, before (left) and after (right) replacing recognized UMLS concepts by pseudo-tokens. Red words and phrases are wrong candidate answers. The correct answer is shown in green and underlined.

after replacing all digits with 'D' (e.g., 'type-3' becomes 'type-D'). The resulting vocabulary sizes of BioRead and BioReadLite are approx. 3.9M and 597k, respectively. Out-of-vocabulary words have been replaced by 'UNK'.

By replacing (possibly multi-token) concept names with @entityID tokens, we allow MRC methods that can only select a single token from the passage (the two methods we re-implemented belong in this category) to cope with cases where the correct answer is actually multi-token. Furthermore, by replacing concept names with @entityID, we do not let MRC systems look up the concepts in external resources (e.g., biomedical ontologies), forcing them to base their responses on the passages of the dataset. The use of MetaMap, however, also adds noise, since MetaMap is not entirely accurate. For example, in Fig. 2 (left), it failed to recognise 'metabolites' as a biomedical concept.[8] Similar noise was introduced in CBTest and BookTest by the named entity recognisers that were used during their construction.

3. Re-implemented Methods and Baselines

We re-implemented and experimented with AS Reader (Kadlec et al., 2016), because it is one of the simplest and most well-known deep learning MRC methods. It has also been shown (Bajgar et al., 2016) that increasing the size of the training set of AS Reader (using BookTest instead of CBTest) leads to much larger performance gains than training more complex MRC methods, like AOA Reader (Cui et al., 2017) and EpiReader (Trischler et al., 2016), on the original training set (CBTest). We also reimplemented and experimented with AOA Reader (Cui et al., 2017), an extension of AS Reader that uses a more complex attention mechanism, because it is one of the best performing methods on CBTest (Bajgar et al., 2016). We make both re-implementations publicly available, as already noted.

AS Reader (Kadlec et al., 2016) uses a bidirectional recurrent neural network (biRNN) (Schuster and Paliwal, 1997; Seo et al., 2016) with GRU units (Cho et al., 2014) to process the passage (context) and another one to process the question. The states of the first biRNN (the concatenated states of the two directions, for each token position) are used as context-sensitive embeddings of the passage tokens, whereas the last states of the second biRNN (the concatenated last states of the two directions) represent the question. The dot product between the question representation and the context-sensitive embedding of each passage token is then computed, and a softmax is applied to the dot products to turn them into attention scores from 0 to 1. The candidate answers can only be single tokens of the passage. If a candidate answer occurs multiple times in the passage, its attention scores are summed. Finally, the candidate answer with the largest (summed) attention score is selected.

AOA Reader (Cui et al., 2017) uses a biRNN to create context-sensitive embeddings for each passage token, as in AS Reader. Another biRNN processes the question, but instead of keeping only the (concatenated) last states of the two directions as the question representation, all the states

[8] We configured MetaMap for high precision, by setting its minimum score of recognised concepts to 10.

of the question biRNN (the concatenated states from both directions, for each question token position) are kept as context-sensitive embeddings of the question tokens. The dot product between each context-sensitive embedding of the passage and each context-sensitive embedding of the question is then computed, leading to a matrix $M = C \times Q$ of dot products, where C and Q are the lengths of the passage (context) and question, respectively, in tokens. Intuitively, each element $m_{i,j}$ of M shows how relevant token i of the passage is to token j of the question. The i-th row of M contains Q scores, showing how relevant each token of the question is, from the viewpoint of the i-th token of the passage. The rows of M are averaged (after applying a softmax to each row first) to obtain a single row-vector q with Q scores that shows how relevant each token of the question is with respect to *all* the tokens of the passage.
Similarly, the j-th column of M contains C scores showing how relevant each token of the passage is, from the viewpoint of the j-th token of the question. The matrix-vector multiplication $M'q^T$, where M' is the original M with a softmax applied to each column, produces C scores that show how important each passage token is from the viewpoint of the entire question, as captured by q. A softmax is applied to the C scores, to turn them into attention scores from 0 to 1. As in AS Reader, the candidate answers can only be single tokens of the passage. If a candidate answer occurs multiple times in the passage, its attention scores are summed. Finally, the candidate answer with the largest (summed) attention score is selected.

Baselines: The first baseline, called BASE1, returns the candidate answer (@entityID) that occurs most frequently in the context (passage), on the grounds that this candidate answer is more likely to have also occurred in the question (a sentence that follows the passage) and, hence, more likely to have been converted to @placeholder. The second and third baselines, BASE2 and BASE3, return the candidate answer that occurs first or last in the context, respectively. The last candidate answer is arguably more likely to be repeated in the question and, hence, more likely to have been converted to @placeholder, whereas the first candidate is the least likely to be repeated in that sense. We also suspected that the biRNN encoder of the passage of AS Reader and AOA Reader would tend to 'remember' more the last (in the forward RNN) and the first (in the backwards RNN) tokens (and candidate answers) of the passage, in an extreme case behaving like BASE2 and BASE3.
In the fourth baseline, BASE4, we first extract all the token n-grams ($n = 2$) of the question that contain the @placeholder.[9] For each candidate answer (@entityID), we then replace the @placeholder in all the extracted n-grams by the particular candidate answer, and count the total number of occurrences of the resulting n-grams in the context. The candidate answer with the largest total number of n-gram occurrences is returned as the answer.

Human performance: To get a rough estimate of how easily humans can answer the questions of BioRead, we randomly selected 30 instances from BioRead's test subset and gave them to three human annotators (the first two authors and a colleague), who had no biomedical background. The annotators were shown the context and question of each instance (as in Fig. 2, right) in a user interface that displayed @entityIDs as hyperlinks, and they were asked to select (click on) the correct candidate answer (@entityID). When the annotators felt they were clueless (or very uncertain) about the correct answer, they could indicate this by clicking on a button, but they were instructed to select an answer when they felt it was probably the correct one, even if they were not entirely sure. The mean accuracy of the three annotators was 68.01% (77.27%, 65.22%, 61.54% per annotator), counting only instances they answered (78.89% on average, 73.33%, 76.67%, 86.67% per annotator). The mean pairwise inter-annotator agreement, measured as Cohen's Kappa (Cohen, 1960), was 68.57, considering only questions answered by both annotators in each pair. If not answering a question is treated as an additional candidate answer, the mean pairwise Kappa becomes 50.32.

4. Experimental Results

Table 3 summarises our experimental results on BioRead-Lite; we did not have the computational resources to experiment with the full BioRead dataset, but we hope that others may be able to do so.[10] With the exception of the last row of Table 3, in all other cases we used option (a) of Section 2, i.e., the identifier of each @entityID was unique only within the particular instance. For AS Reader and AOA Reader, we used the same hyper-parameter values as in the work of Kadlec et al. (2016) and Cui et al. (2017), respectively. Hence, a direct possible improvement would be to fine-tune the hyper-parameters for BioRead (or BioRead-Lite), which requires, however, substantial computational resources. We stopped training the two methods when their development loss had converged, i.e., after 5 epochs for AS Reader, 15 epochs for AOA Reader when using option (a), and 20 epochs for AOA Reader when using option (b); recall that in option (b) the identifier of each @entityID is unique in the entire dataset. A single training epoch (including computing the development loss) takes 17, 21, and 22 hours, respectively (Table 3). Performance is measured in terms of accuracy, i.e., number of correctly answered development or test instances, divided by the total number of development or test instances.
Table 3 shows that AOA Reader is clearly more accurate than AS Reader, at the expense of training speed, reaching 50.44% and 49.94% development and test accuracies with option (a), compared to 37.90% and 42.01% for AS Reader, respectively. These results confirm that the more elaborate attention mechanism of AOA Reader is important, as also reported in previous work (Cui et al., 2017; Bajgar et al., 2016; Munkhdalai and Yu, 2016). Despite its simplicity, BASE1 (most frequent candidate answer in the passage) is a reasonably strong baseline, reaching 26.86% development and 28.87% test accuracy, but AS Reader and AOA Reader outperform it by a wide margin. BASE2 and BASE3 are much weaker, suggesting that AS Reader and

[9] We experimented with $2 \leq n \leq 6$, and selected $n = 2$, which led to the best results on the development set of BioReadLite.

[10] We used a PC running Ubuntu, with 64 GB RAM, a 16 core CPU, and a GeForce GTX TITAN X GPU with 12GB memory.

Method	Dev. Accuracy	Test Accuracy	Training Epochs
BASE1 (a)	26.86	28.87	n/a
BASE2 (a)	8.14	9.38	n/a
BASE3 (a)	16.48	17.28	n/a
BASE4 (a)	40.10	37.20	n/a
AS Reader (a)	37.90	42.01	5 × 17 h
AOA Reader (a)	50.44	49.94	15 × 21 h
AOA Reader (b)	**52.41**	**51.19**	**20 × 22 h**

Table 3: BioReadLite results (%), and number of epochs (and time) required for the development loss to converge, when each entity ID is unique (a) in the particular instance only, or (b) in the entire dataset.

AOA Reader do not just remember the first or last candidate answers of the passage. The best baseline is BASE4 (n-grams). It scored 40.10% development and 37.20% test accuracy, surpassing AS Reader on the development subset, and challenging AS Reader on the test subset. Nevertheless, AOA Reader outperformed BASE4 by a wide margin (Table 3). The performance of AOA Reader improved further (from 50.44% to 52.41% development accuracy, from 49.94% to 51.19% test accuracy), at the expense of additional training time, when option (b) was used, i.e., when each entity ID was unique in the entire dataset, suggesting that AOA Reader was able to learn properties of at least some entities (concepts) from multiple training passages.

We also trained the best method, AOA Reader with option (b), on smaller subsets of BioReadLite to study the effect of the size of the training set. We always used 20 epochs in this experiment, the number of epochs it took for the development loss to converge when using the entire training set of BioReadLite (Table 3, last row). Table 4 shows that increasing the size of the training set leads to improved development accuracy. We see a similar trend in test accuracy (from 49.22% to 51.19%) when going from 50% to 100% of the training set, but surprisingly the best test accuracy (51.51%) was obtained when using only 25% of the training set. The latter may be the result of a random fluctuation (e.g., the optimizer may have managed to find a better local minimum of the loss function in that case). It would be better to repeat each experiment multiple times, with different random parameter initializations, and report mean results (and standard deviations), but we did not have the required resources. Overall, however, it seems worth experimenting with the entire BioRead dataset, instead of BioReadLite, to see if its larger training subset would lead to significant improvements in accuracy. We also note that the average accuracy of the human annotators was 68.01% (Section 3). This score was computed only on a sample of 30 test questions, and it does not consider questions the annotators left unanswered, but it is an indication that there is headroom for improvements in the performance of MRC methods.

Training Subset	Dev. Accuracy	Test Accuracy	Training Epochs
25%	47.06	**51.52**	20 × 6 h
50%	50.25	49.22	20 × 11 h
100%	**52.41**	51.19	20 × 22 h

Table 4: BioReadLite results (%) of AOA Reader, with option (b), using the entire or only subsets of the training set.

5. Conclusions and Future Work

We constructed and make publicly available a new cloze-style biomedical MRC dataset, BioRead, with approx. 16.4 million instances, currently one of the largest MRC datasets and the only one of its kind in the biomedical domain. We also provide a subset of BioRead, BioReadLite, with 900k instances, for groups with fewer resources. Both datasets were constructed in the same way as CBTest and BookTest, but using biomedical journal articles and employing MetaMap to identify biomedical entities (concepts) and replace them by their preferred UMLS names. We also re-implemented and tested on BioReadLite two well-known MRC methods, AS Reader (Kadlec et al., 2016) and AOA Reader (Cui et al., 2017), along with four baselines, as a first step towards a BioRead (and BioReadLite) leaderboard. Our re-implementations are also publicly available.

BioRead and BioReadLite are available in two forms, where each identified entity is replaced by an identifier that is unique (a) only in the particular passage-question instance, or (b) in the entire training corpus. AOA Reader is currently the best method on BioReadLite, and its performance improves (reaching 52.41% development and 51.19% test accuracy) when option (b) is used, suggesting that it manages to learn properties of at least some entities from multiple training passages. The best baseline, which uses n-grams, surpasses the second best method, AOA Reader, on the development set of BioReadLite and performs reasonably well on the test set. Nevertheless, AOA Reader outperforms it by a wide margin.

Future work could use BioRead (and BioReadLite) to test other existing MRC methods in the biomedical domain or develop new MRC methods. It would also be interesting to examine if methods trained (or pre-trained) on BioRead could also cope with real-world biomedical questions, like those of BioASQ, possibly after further training.

6. Acknowledgments

We acknowledge support of this work by the project "Computational Science and Technologies: Data, Content and Interaction" (MIS 5002437) which is implemented under the Action "Reinforcement of the Research and Innovation Infrastructure", funded by the Operational Programme "Competitiveness, Entrepreneurship and Innovation" (NSRF 2014-2020) and co-financed by Greece and the European Union (European Regional Development Fund). We are also grateful to Ryan McDonald, for many useful discussions and his participation in the human performance experiment. Special thanks to Yannis Almirantis and Anastasios Nentidis, who participated in a similar preliminary experiment with biomedical experts.

7. Bibliographical References

Aronson, A. R. and Lang, F.-M. (2010). An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17:229–236.

Bajgar, O., Kadlec, R., and Kleindienst, J. (2016). Embracing Data Abundance: BookTest Dataset for Reading Comprehension. *CoRR*, abs/1610.00956.

Chen, D., Bolton, J., and Manning, C. D. (2016). A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 2358–2367, Berlin, Germany.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading Wikipedia to Answer Open-Domain Questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1870–1879, Vancouver, Canada.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., and Hu, G. (2017). Attention-over-Attention Neural Networks for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 593–602, Vancouver, Canada.

Goldberg, Y. (2017). *Neural Network Methods in Natural Language Processing*. Morgan and Claypool Publishers.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Hermann, K. M., Kociský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching Machines to Read and Comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems*, pages 1693–1701, Montreal, Quebec, Canada.

Hill, F., Bordes, A., Chopra, S., and Weston, J. (2015). The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. *CoRR*, abs/1511.02301.

Kadlec, R., Schmid, M., Bajgar, O., and Kleindienst, J. (2016). Text Understanding with the Attention Sum Reader Network. *CoRR*, abs/1603.01547.

Munkhdalai, T. and Yu, H. (2016). Reasoning with Memory Augmented Neural Networks for Language Comprehension. *CoRR*, abs/1610.06454.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas.

Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Seo, M. J., Kembhavi, A., Farhadi, A., and Hajishirzi, H. (2016). Bidirectional Attention Flow for Machine Comprehension. *CoRR*, abs/1611.01603.

Sultan, M. A., Castelli, V., and Florian, R. (2016). A Joint Model for Answer Sentence Ranking and Answer Extraction. *Transactions of the Association of Computational Linguistics*, 4:113–125.

Trischler, A., Ye, Z., Yuan, X., Bachman, P., Sordoni, A., and Suleman, K. (2016). Natural language comprehension with the epireader. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, Austin, Texas.

Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., Almirantis, Y., Pavlopoulos, J., Baskiotis, N., Gallinari, P., Artieres, T., Ngonga, A., Heino, N., Gaussier, E., Barrio-Alvers, L., Schroeder, M., Androutsopoulos, I., and Paliouras, G. (2015). An overview of the BioASQ Large-Scale Biomedical Semantic Indexing and Question Answering Competition. *BMC Bioinformatics*, 16(138).

MMQA: A Multi-domain Multi-lingual Question-Answering Framework for English and Hindi

Deepak Gupta, Surabhi Kumari, Asif Ekbal and Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology Patna
Patna, India-801106
{deepak.pcs16, surabhi.mtmc16, asif, pb}@iitp.ac.in

Abstract

In this paper, we assess the challenges for multi-domain, multi-lingual question answering, create necessary resources for benchmarking and develop a baseline model. We curate 500 articles in six different domains from the web. These articles form a comparable corpora of 250 English documents and 250 Hindi documents. From these comparable corpora, we have created 5,495 question-answer pairs with the questions and answers, both being in English and Hindi. The question can be both factoid or short descriptive types. The answers are categorized in 6 coarse and 63 finer types. To the best of our knowledge, this is the very first attempt towards creating multi-domain, multi-lingual question answering evaluation involving English and Hindi. We develop a deep learning based model for classifying an input question into the coarse and finer categories depending upon the expected answer. Answers are extracted through similarity computation and subsequent ranking. For factoid question, we obtain an MRR value of 49.10% and for short descriptive question, we obtain a BLEU score of 41.37%. Evaluation of question classification model shows the accuracies of 90.12% and 80.30% for coarse and finer classes, respectively.

Keywords: Multi-lingual Question answering, Answer extraction, Neural network, Question classification

1. Introduction

Question answering (QA) is an important area with a wide range of applicability in various Natural Language Processing (NLP) tasks, such as information retrieval, information extraction etc. The aim of a QA system is to automatically extract/generate the answer(s) for a given question from the data repository (e.g., web, document etc.). In a QA system questions are formulated in natural languages and answers are also dealing with natural languages. Unlike search engine instead of extracting information, here a QA system usually focus on extracting relevant and precise answer(s). In other words, we can say QA system is the extended modification in the search engine. For achieving QA system, usually three subprocesses are followed i) question classification ii) document(s)/passage(s) extraction and iii) appropriate answer(s) extraction. Most of the existing works focus on retrieving answers in the same language in which the questions are posed. However, with the rapid growth of multilingual contents on the web, it is necessary to build an automated system that retrieves information from the documents written in multiple languages.

Posing questions in multiple languages and retrieving answers accordingly is known as Multilingual Question Answering (MQA), which has emerged as an interesting research area in QA. This enables the situation where the question could be in a different language from the language of documents where the answer(s) lies. This allows users to interact in their native languages, facilitating multilingual information access, which is immensely useful in a country like India. MQA system can contribute to conserving the endangered languages which are losing their existence and prestige as mentioned in (Knott et al., 2001). Hindi is a widely spoken language in India, and in terms of native speakers, it ranks fourth all over in the world. In India, a sum of 53.60% of total population speak Hindi as compared to English (12.18%). English, on the other hand, is used for all kinds of official communications. There is often need to exchange information from Hindi to the other popular language(s) such as English.

In recent year several multilingual and cross lingual QA systems have been built. These systems are seeking to overcome the issue of accessing and retrieving information in multiple languages. The majority, however, are based on translating relevant sections of the question – usually with the aid of machine translation system - which is used to access to a collection containing relevant information. The basic goal of MQA framework is to set up a common system to evaluate both bilingual and cross-lingual question answering that process queries in either Hindi or English language and retrieve answer in either language from documents in Hindi or English. The main motivations and/or contributions of the current work are as follows:

1. Most of the existing works are in resource-rich languages such as the English. Indian languages are resource-scarce, and developing a multi-lingual QA system involving English and Hindi has the benefit of utilizing resources and tools available for the resource-rich language like English.

2. Creating a benchmark setup for multi-lingual QA involving Indian languages will be beneficial for multi-lingual information access. To the best of our knowledge, this is the very first attempt in this direction.

3. Question classification is an important step in Question-Answering (QA). We propose a method based on deep Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) for question classification.

Article (English)		Article (Hindi)		
Shimla is the capital of Himachal Pradesh and was also the summer capital in pre-Independence India. Shimla derives its name from Shyamala Devi, an incarnation of the goddess Kali, whose temple existed in the dense forest covering the Pradesh Jakhu Hill in the early 19th century. Shimla is the capital of Himachal in pre-Independence India. Covering an area of 25 sq km at a height of 7,238 ft, Shimla is surrounded by pine, deodar and oak forests.		शिमला, एक ख़ूबसूरत हिल स्टेशन है जो हिमाचल प्रदेश की राजधानी है। समुद्र की सतह से 2202 मीटर की ऊँचाई पर स्थित इस जगह को 'समर रिझूज' और 'हिल स्टेशनों की रानी' के रूप में भी जाना जाता है। वर्तमान का शिमला जिला 1972 में निर्मित किया गया था। इस जगह का यह नाम 'माँ काली' के दूसरे नाम 'श्यामला' से व्युत्पन्न है। जाखू, प्रॉस्पैक्ट,ऑब्सर्वेटरी, एलीसियम और समर इस जगह की महत्वपूर्ण पहाड़ियाँ हैं।		
Question (English)	**Answer (English)**	**Question (Hindi)**	**Answer (Hindi)**	**Answer Availability**
What is the capital of Himachal Pradesh?	Shimla	हिमाचल प्रदेश की राजधानी क्या है?	शिमला (*Shimla*)	EN and HI
How much area is covered by Shimla?	25 sq km	शिमला का क्षेत्रफल कितना है?	-	EN
From which goddess name the Shimla word derived?	Shyamala Devi	शिमला शब्द किस देवी के नाम से लिया गया है?	श्यामला (*Shyamala*)	EN and HI
What is the height of Shimla from sea level?	7,238 ft	समुद्र के स्तर से शिमला की ऊंचाई क्या है?	2202 मीटर (*2202 meter*)	EN and HI
In which year Shimla established?	-	शिमला की स्थापना किस साल हुई?	1972	HI

Table 1: An example of comparable articles from English and Hindi and set of question-answer pairs created from the given articles.

Reasoning Types	Question & Answer Sentence	Descriptions
Interlingual semantic word	**Q:** What was introduced in the Sixth Five-Year Plan? **Answer Sentence:** छठी पंचवर्षीय योजना (1980-1985) ने भी आर्थिक उदारीकरण की शुरुआत को चिन्हित किया । (*Chhati panchvarshiya yojna (1980-1985) ne bhi arthik udarikaran ki shuruaat ko chinhit kiya.*)	An interlingual semantic word knowledge is required to provide the answer. Only translation may not help to extract the answer.
Multiple sentence reasoning	**Q:** What is the part of the Adam's Bridge? **Answer Sentence: Pamban Island** is situated in the Gulf of Mannar between India and Srilanka. It is a part of the Adam's Bridge.	There is an anaphora, or fusion of multiple sentences is required to answer.
Word matching	**Q:** What is the collection of Vedic hymns or mantras called? **Answer Sentence:** The collection of Vedic hymns or mantras is called the **Samhita**.	Word matching between question and answer sentence can provide the answer.
Interlingual syntax variation	**Q:** प्लेग किसके लिए जिम्मेदार है ? (*Plaque kiske liye jimmedar hai?*) **Answer Sentence:** Plaque deposited on the teeth and under the gumline irritates the gum tissue, and causes **gingivitis**.	Syntactic structure of question and answer sentence vary across the language.
Single Sentence	**Q:** What is Sustainable development? **Answer Sentence:** Scale defines the relationship between distance on a map and on the earth's surface. Sustainable development: **Development that does not exploit resources more rapidly than the renewal of those resources**, ...	Answer of short descriptive question can be a single sentence from a paragraph.
Multiple Sentence	**Q:** Why India is considered to be an eastern country? **Answer Sentence:** ... **India lies east of the Prime Meridian. Therefore India is considered to be an eastern country because of its situation in the Eastern Hemisphere**....	Answer of short descriptive question can be multiple sentence from a single or multiple paragraph.

Table 2: The set of possible reasoning types with the corresponding question-answer pair example and descriptions. Reasoning types show the difficulty of the question in terms of finding their answer. The answer in answer sentence has been shown in bold font.

Problem Definition

Given a natural language question Q (factoid or short descriptive) in either language, English or Hindi. The QA system should return the answer A for the given question Q from the comparable English and Hindi documents. The returned answer should be in the same language[1] as the question Q.

2. Related Work

In literature, we found a very few existing works related to question-answering (QA) in Hindi or English (Sekine and Grishman, 2003; Kumar et al., 2005; Sahu et al., 2012; Stalin et al., 2012). However, none of these focuses on multilingual QA. (Kumar et al., 2005) implemented the Hindi search engine. The task of the search engine is to retrieve relevant passages from the collection of the passages. In the proposed architecture various modules were introduce. Automatic Entity Generator module identified domain related entities from which user can ask questions. Question classification module has several categories of question. An answer extraction module extracts the answer. By using ranking, answer selection module selects the answer among the candidate answers.

(Sahu et al., 2012) discussed an approach for question answering system for the Hindi language. This work deals with four types of questions when, where, how many and what time. For given question, the answer was retrieved from Hindi text. Each sentence in the text was analyzed to understand its meaning. In this work, they represent the questions using query logic language(QLL) which is a subset of Prolog. For identification of the noun, verb and question word Hindi shallow parser was used.

(Stalin et al., 2012) implemented the web based Hindi question answer. In this work the question and answer deal with only Hindi language, if the answer was not presented in Hindi document then it was retrieved from Google.

(Sekine and Grishman, 2003) proposed a question answering system for Hindi and English. The questions were created in Hindi language and the answers retrieved from Hindi newspaper in the Hindi language. These answers were then converted into the English language. In this work, an English Hindi bilingual dictionary was used to find top 20 Hindi articles which were used to find candidate answers.

(Reddy and Bandyopadhyay, 2006) proposed question answering system in the Telugu language. The system was dialogue based and railway specific domain. The architecture was based on the keyword approach. The query analyzer generates the tokens and keywords. From tokens, SQL statements were generated. Using SQL query the answer was retrieved from the database.

(Reddy and Bandyopadhyay, 2006) develop the question answer system in English and Punjabi language. In this work a pattern and matching algorithm was introduced to retrieve the most relevant appropriate answer from multiple sets of answers for a given question.

3. Resource Creation

We create QA dataset MMQA (Multi-domain Multilingual Question Answering) in Hindi and English languages covering multiple domains. We focus on creating *factoid* and

[1]Whenever required, a in-house language identification module (Gupta et al., 2014) and translation are used.

Domains	English/Hindi			
	# Articles	# Paragraphs	# Sentences	# Words
Tourism	112/112	1,569/1,186	5,077/3,799	90,222/71,863
History	68/68	563/597	2,518/2,224	40,368/46,418
Diseases	31/31	441/298	1,932/1,171	33,787/28,128
Geography	16/16	81/171	304/520	8,915/9,443
Economics	13/13	146/144	667/477	10,633/10,875
Environment	10/10	64/54	290/272	5,319/6,109
Total (EN/HI)	250/250	2,864/2,450	10,788/8,463	189,244/172,836
Total (EN+HI)	500	5,314	19,251	362,080

Table 3: Statistics of comparable English and Hindi articles from various domains.

short descriptive questions. The MMQA dataset is created in three different stages: *Comparable Article Curation*, *Question Answer Formulation* and *Validation*.

3.1. Comparable Article Curation

Since our objective was to build a multilingual QA dataset, therefore we curated comparable articles from the various web sources covering different domains. Rather than translating an article from one language to the other, we curated comparable articles, mainly for two reasons: **(i)** to bridge the information gap between two different language articles, and **(ii)** to assess the challenges of dealing with two syntactically divergent texts to retrieve answers for the given questions. From each of these articles, we extracted individual paragraphs, and removed images, links, and tables. We curated a total of 500 articles from 6 different domains. We curated these texts from the different web sources using a web crawler[2]. We provide the statistics of the comparable articles in Table 3.

3.2. Question and Answer Formulation

We engage annotators[3] to formulate the question-answer pairs in their own word. The annotators are provided with a interface displaying domain name, article name and the comparable article in parallel. They were asked to formulate questions in English and Hindi by looking into both the comparable articles. When they formulate a question by looking a paragraph in one (English or Hindi) article, they also have to verify whether the question of interest is available in the comparable article or not. In particular, they have to provide the *question, answer, answer source (sentence or paragraph where the answer exists), type of question (factoid, descriptive)* in both the languages, if exist. Additionally, annotators were encouraged to set questions in their own words. Statistics[4] of question and answers in both the languages are shown in Table 4.

We ask two other annotators to verify the questions and answers generated in both the languages. Annotators were given a free hand to correct the answers to some extent, or by eliminating the question-answer pairs, if found not fitting.

[2]Tourism (EN):www.india.com/travel
Tourism (HI): https://hindi.nativeplanet.com
Diseases (EN, HI): https://simple.wikipedia.org/wiki/List_of_diseases,
rest of the domains are curated from http://www.jagranjosh.com/

[3]The annotators are equally proficient in both the languages

[4]Total of 7120 questions (English+Hindi) for which the answer exists in either of two language documents.

3.3. Validation

Validation stage is performed to ensure that we obtain a high quality datasets at the end. We ask two other annotators to verify the questions and answers generated in both the languages. Annotators were given a free hand to correct the answers to some extent, or by eliminating the question-answer pairs, if found not fitting. The validation stage is applicable for the question-answer pair of both the languages.

3.4. Analysis

We analyze the questions and answers of the proposed MMQA dataset. It is required to understand its property and usefulness as a multilingual dataset. Our analysis focuses on studying the difficulty level of questions and diversity of answers. We provide some examples in Table 2 to give some ideas about the difficulty levels associated. For better understanding and thorough analysis of various answer types, similar to Rajpurkar et al. (2016) and Trischler et al. (2016), we categorize the answers of factoid questions into 8 entities and phrases. Statistics of the answer types for English and Hindi QA pairs are provided in Table 7.

An example of QA pairs formulated from a comparable articles is given in Table 1. Some examples of short descriptive QA pair from our dataset are given in Table 5. The direct comparison of our dataset with the Cross-Language Evaluation Forum (CLEF) datasets (Pamela et al., 2010) is not possible because we have created question answers pair in both language (MQA) in contrast the CLEF dataset have the question and answer pair in the different languages. However, we have shown the comparison in various terms as shown in Table 6.

4. Evaluation: Proposed Approach

We develop a translation based approach for multilingual QA. As English is a resource-rich language, we translate Hindi question and articles into English. Our proposed model comprises of *Knowledge Source Preparation*, *Question Processing* and *Answer Extraction*, We describe the details of each component in the following.

4.1. Knowledge Source Preparation

In this step, an information source (articles) from which answers are to be derived was set-up. We translate Hindi questions and articles into English by Google Translate[5]. The complete English articles are indexed at passage level using inverted indexing mechanism. We use the Lucene[6] implementation of inverted indexing.

4.2. Question Processing:

The question processing (QP) step is responsible for analyzing and understanding the questions posed to the QA system. We perform question classification with the question classes proposed by Li and Roth (2002). Question class provides us the semantic constraint on the sought-after answer. We propose a deep learning based question classification

[5]https://translate.google.com

[6]https://lucene.apache.org/

Domains	QA pair in only English (Fact/Desc)	QA pair in only Hindi (Fact/Desc)	QA pair in Both Languages (Fact/Desc)	Total QA pair (Fact/Desc)	Total QA pair
Tourism	456/14	403/5	422/10	1,281/29	1,310
History	110/75	126/78	1,118/588	1,354/741	2,095
Diseases	81/54	33/26	48/40	162/120	282
Geography	55/7	29/10	174/202	258/219	477
Economics	25/4	14/5	682/218	721/227	948
Environment	9/3	2/1	226/142	237/146	383
Total	736/157	607/125	2,670/1,200	4,013/1,482	5,495

Table 4: Statistics of QA pairs for factoid and short descriptive questions in English and Hindi.

Question (English): Why did Alexander marched back in 325 BC?
Question (Hindi): अलेक्जेंडर 325 ईसा पूर्व में क्यों चला गया?
Answer (English): After Alexander's last major victory in India as his forces refused to go any further. They were too tired to carry on with the Alexander's expedition and wanted to return home. Moreover, the might of Magadhan Empire (the Nanda Rulers) also dissuaded them. Alexander marched back in 325 BC after making necessary administrative arrangement for the conquered territories. He died at the age of 33 years when he was in Babylon.
Answer (Hindi): हालांकि, यह जीत भारत में उसकी आखिरी बड़ी जीत साबित हुई क्योंकि उसकी सेना ने इसके बाद आगे जाने से इनकार कर दिया था। वे सिकंदर के अभियान के साथ जाने से काफी थक गए थे और वापस घर लौटना चाहते थे। इसके अलावा, मगधियन साम्राज्य (नंदा शासक) की ताकत से भी वो भयभीत थे। विजय प्राप्त प्रदेशों के लिए आवश्यक प्रशासनिक व्यवस्था करने के बाद सिकंदर 325 ईसा पूर्व वापस चले गया।

Question (English): What does Buddhist texts such as Jatakas reveal?
Question (Hindi): बौद्ध ग्रंथों जैसे जतकस क्या बताते हैं?
Answer (English): ABuddhist texts such as Jatakas reveal socio-economic conditions of Mauryan period while Buddhist chronicles Mahavamsa and Dipavamsa throws light on the role of Ashoka in spreading Buddhism to Sri Lanka.
Answer (Hindi): Not Available

Table 5: Examples of short descriptive QA pairs from the dataset.

model to classify the question at coarser and finer level. The proposed question classification model is described as follows:

4.2.1. Question Classification

This is a vital component of any scoring based answer extraction technique. Its performance is the major concern as the errors in this component can propagate through the next stage and can affect the subsequent stages. In general question classification categorizes a question at coarser and finer level based on the answer type. For example, when considering the question Q: *When did Mandi become a part of India?*, we wish to classify this question as coarse class: *Numeric* and finer class: *date*, implying that only candidate answers that are dates need to be considered. With the recent developments in deep learning, neural network models have shown promise for QA. Deep neural network being perform exceptionally well in other NLP problem. Inspired by the success of deep neural network we adapt neural network architecture to develop our question classification model. Our question classification model is based on CNN and RNN. The model comprises of *Question embedding layer, Convolution layer, Recurrent layer, Softmax classification layer*. Our question classification model is inspired from (Kim, 2014) and (Xiao and Cho, 2016). The input to the model is an English question. Now we describe each component of the model:

- **Question embedding layer:** It is responsible for obtaining the sequence of dense, real-valued vectors, $E = [v_1, v_2 \dots v_T]$ of a given question having T tokens. We keep the maximum size of token $T = 15$ in this layer. The distributed representation $v_i \in R^k$ is the k-dimensional word vector. The distributed representation v is looked up into the word embedding matrix W. In our experiment we have used the pre-trained word embedding [7] matrix by (Mikolov et al., 2013).
- **Convolution layer:** This layer performs convolution operation. Similar to (Xiao and Cho, 2016) and (Kim, 2014) we obtain convolution feature c_t at given time t. Then we generate the feature vectors $C = [c_1, c_2 \dots c_T]$. The convolution operations are performed with the filter size of 3, 4 and 5.
- **Recurrent layer:** This layer performs recurrent operations over the convolution output c at given time t. Similar to (Xiao and Cho, 2016) we obtained the forward and backward hidden states at every step time t using the gated recurrent unit (GRU) (Cho et al., 2014). Xiao et al. (2016) have used LSTM unit, however we have employed GRU (Cho et al., 2014) due to its less complex architecture compare to long short term memory (LSTM).

$$\mathbf{z}_i = \sigma(\mathbf{W}_z c_i + \mathbf{V}_z \mathbf{h}_{i-1} + \mathbf{b}_z)$$
$$\mathbf{r}_i = \sigma(\mathbf{W}_r c_i + \mathbf{V}_r \mathbf{h}_{i-1} + \mathbf{b}_r)$$
$$\mathbf{c}_i = tanh(\mathbf{W} c_i + \mathbf{V}(\mathbf{r}_i \odot \mathbf{h}_{i-1}) + \mathbf{b})$$
$$\mathbf{h_i} = z_i \odot \mathbf{h}_{i-1} + (1 - \mathbf{z}_i) \odot \mathbf{c}_i$$

[7] https://code.google.com/archive/p/word2vec/

	2003	2004	2005	2006	2007	2008	2009	**Our data**
Target lang.	3	7	8	9	10	11	9	2
Collection	News 1994		+ News 1995		+ Wikipedia Nov. 2006		JRC-Acquis	Web
No. of questions	200						500	**7120**
Type of questions	200 Factoid		+ Temp. restrict +Defn	-Type of question +List	+ Linked question +Closed lists		-Linked +Reason +Purpose +Procedure	Factoid Descriptive
Supporting info.	Document			Snippet			Paragraph	Document

Table 6: Comparison of our dataset with the various released Cross-Language Evaluation Forum (CLEF) dataset

where $\mathbf{z}_i$, $\mathbf{r}_i$ and $\mathbf{c}_i$ are update gate, reset get and new memory content, respectively. c_i is the convolution output at time t.The final output of recurrent layer h is obtained as the concatenation of the last hidden state of forward and backward hidden states.

- **Softmax classification layer:** Finally, the fixed-dimensional vector h is fed into the softmax classification layer to compute the predictive probabilities for all the question classes (coarse or fine).

4.2.2. Query Formulation

In order to form the query, we remove all the stop word, punctuation symbol from the question. We tag the question with Stanford PoS tagger (Toutanova et al., 2003). Then we concatenate all the noun, verb and adjective in the same order in which it appears in the question.

4.3. Passage Retrieval

The candidate passage that contains the answer(s) to the given question(s) are extracted in this stage. We exploit the Lucene's text retrieval functionality to retrieve passage. It retrieves and ranks the passages using a combination of a Boolean model and the BM25 vector space model (Zaragoza et al., 2004). The query obtained from the *question processing* stage, serve as an input to the scorer module. The most relevant 30 passages were retrieved for subsequent processing.

4.4. Candidate Answer Extraction

This depends on the output of question classification. For factoid question, the coarse class and finer class guide this stage to extract the appropriate entities from the candidate passage(s). We tag the candidate passage with Stanford named entity tagger (Finkel et al., 2005). We utilize the coarse class and finer class of a question to extract the suitable candidate answers. For a descriptive question, candidate answers are extracted by segmenting the relevant passage.

4.5. Answer Scoring and Ranking

Each candidate answer is assigned a score using the candidate answer extraction phase. We segment the candidate passage into several candidate answer sentences. Thereafter, we calculate the score for each of the candidate answer sentences.

1. **Term coverage (TS):** It calculates the number of query terms appearing in the candidate answer sentence. This is normalized w.r.t the number of terms present in the given query.

2. **Proximity score (PS):** It calculates the length of the shortest span that covers the query contained in the candidate answer sentence. This is again normalized in the same way.

3. **N-Gram coverage score (NS):** We compute the n-gram coverage till $n = 4$. Finally, the n-gram score between a query (q) and a candidate answer sentence (S) is calculated based on the following formula.

$$NGCoverage(q, S, n) = \frac{\sum_{ng_n \in S} Count_{common}(ng_n)}{\sum_{ng_n \in q} Count_{query}(ng_n)} \quad (1)$$

$$NGScore(q, S) = \sum_{i=1}^{n} \frac{NGCoverage(q, S, i)}{\sum_{i=1}^{n} i} \quad (2)$$

4. **Semantic Similarity Score (SS) :** Query and candidate answer are represented using the semantic vectors. Cosine similarity is then computed between the query and candidate answers.

$$\texttt{VEC}(X) = \frac{\sum_{t_i \in X} \texttt{VEC}(t_i) \times \text{tf-idf}_{t_i}}{\textit{number of look-ups}} \quad (3)$$

where X is query q or candidate answer sentence S, $\texttt{VEC}(t_i)$ is the word vector of word t_i. *number of look-ups* represents the number of words in the question for which pre-trained word embeddings[8] are available.

5. **Pattern matching score (MS):** This score is used in the descriptive question only. We design a set of patterns similar to the (Joho, 1999) to match a query against the candidate answers. We setup a score for each pattern according to their importance.
For factoid and descriptive questions the weighted aggregate score for each candidate answer (A) is calculated as:

$$\begin{aligned} S_f(Q, A) &= W_1^f * TC + W_2^f * PS + W_3^f * NS + W_4^f * SS \\ S_d(Q, A) &= W_1^d * TC + W_2^d * PS + W_3^d * NS \\ &\quad = + W_4^d * SS + W_5^d * MS \end{aligned} \quad (4)$$

[8] https://code.google.com/archive/p/word2vec/

Answer type	Proportion (English/Hindi)	Examples (English/Hindi)
Person	12.22 / 14.28	Krishna /तुलसीदास (*Tulasidas*)
Location	17.26 / 14.89	Madurai /भुवनेश्वर (*bhubaneswar*)
Organization	7.69 / 8.96	International Monetary Fund/ टाटा मूलभूत अनुसंधान संस्थान (*Tata Mulbhut Anusandhan Sansthan*)
Noun Phrase	23.79 / 24.57	Hotel Apsara / स्टील प्लांट (*Steel plant*)
Verb Phrase	2.58 / 1.57	planned economic development/ तनाव से छुटकारा (*Tanav se chhutkara*)
Adjective Pharse	1.98 / 1.02	Smiling Buddha/ 14 प्रमुख भारतीय बैंक (*14 pramukh Bhartiya bank*)
Date / Numbers	32.43 / 33.17	580 / 7 किलो (*7 kilo*)
Other	2.05 / 1.54	at least two / सापूतरा का मतलब है 'नागों का वास'। (*Saputra ka matlab hai 'Nagon ka vas*)

Table 7: Set of various answer type categories (only for factoid questions) from the dataset with their proportion (in %) for English and Hindi answer.

Here, W_k^f and W_k^d are the learning weights for factoid and descriptive question, respectively. Optimal values [9] [10] are determined through the validation data. For factoid question, the candidate having the maximum score is returned as an answer to the given question. Answers to the descriptive questions may sometimes cover multiple sentences. At first, we consider the sentence having the maximum score, and then include the other sentences which have scores closer to the highest one.

5. Experiments, Result and Analysis

The experiments performed on the benchmark English-Hindi dataset can be categorized in two-fold: English question classification and answer extraction.

5.1. Question Classification

We perform the experiment on coarse and fine class set of the questions using the model discussed in Section 4.2.. For training, we use three datasets

1. A dataset[11] of $5,452$ questions collected from Hovy et al. (2001), TREC 8 and TREC 9 questions dataset,

2. A dataset of 500 questions from TREC 10[12].

3. We also manually label $1,022$ questions at coarse and finer labels with the taxonomy guidelines provided by Li and Roth (2002). These questions were randomly taken from the set of curated questions.

We perform 5-fold cross-validation to evaluate the question classification model. We obtain the accuracies of 90.12% and 80.30% for question classification under coarse (i.e. 6 classes) and fine classes (i.e. 63 classes), respectively. This model is used to classify the incoming questions while we perform answer extraction.

[9] optimal weights for factoid (0.31, 0.18, 0.39, 0.12)
[10] optimal weight for descriptive (0.21, 0.09, 0.23, 0.19,0.28)
[11] http://cogcomp.org/Data/QA/QC/
[12] http://cogcomp.org/Data/QA/QC/TREC_10.label

Network Training and Hyper-parameters

We have applied the rectified linear units (ReLu) (Nair and Hinton, 2010) as the activation function in our experiment. We use the development data to fine-tune the hyper-parameters. In order to train the network, the stochastic gradient descent (SGD) over mini-batch is used and Backpropagation algorithm (Hecht-Nielsen, 1992) is used to compute the gradients in each learning iteration. In order to prevent the model from over-fitting, we employed a dropout regularization (set to 50%) proposed by (Srivastava et al., 2014) on the penultimate layer of the network. We have used cross-entropy loss as the loss function.

5.2. Answer Extraction

We perform experiments for the factoid and descriptive questions using the model proposed in Section 4.. We use 10% of the total dataset of factoid and descriptive QA pairs, shown in Table 4, as the validation dataset to fine-tune the weight parameters. Mean reciprocal rank (MRR) and exact match (EM) (Trischler et al., 2016) are used to evaluate the model performance on factoid question. For descriptive questions, we use the well-known machine translation evaluation metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004).While evaluating we translate Hindi answer to English and create a gold answer set by combining the actual English answer and the translated English answer for each question. Performance of the system is reported in Table 8.

For factoid questions, we obtain the maximum MRR value of 65.72 for the domain *Environment*. We obtain the lowest EM and MRR values for the domain *Diseases*. One possible reason could be that most of the factoid answers are the phrases and the PoS tagger could not extract these correctly. The system achieves the maximum BLEU of 48.51 and ROUGE-L of 45.72 scores for the domains *Diseases* and *History*, respectively. Our model could not perform well for the descriptive questions of the domain *Tourism*. However, it is to be noted that *Tourism* contains only a few (29) short descriptive questions. Our close analysis reveals that the system suffers due to the errors encountered in the linguistic components such as PoS tagger and named entity (NE) tagger. The NE tagger could not detect some of the

Domains	Factoid		Descriptive	
	EM	MRR	BLEU	ROUGE-L
Environment	39.13	65.72	45.81	42.56
History	29.53	57.19	42.84	45.72
Geography	35.55	52.27	43.02	44.61
Diseases	23.29	34.78	48.51	39.19
Economics	26.28	46.89	45.12	44.77
Tourism	27.68	37.79	22.96	24.29
Total	30.24	49.10	41.37	40.19

Table 8: Performance (in %) of the proposed model for factoid and descriptive questions

entities present in the translated Hindi passage, may be due to the errors encountered during translation.

6. Conclusion

In this paper, we propose a new multilingual QA dataset: MMQA. The dataset has wide coverage of various entities as the answer. It can be used to build a monolingual (EN: English, HI: Hindi), cross-lingual (EN $\rightarrow$ HI, HI $\rightarrow$ EN) and multilingual (EN $\leftrightarrow$ HI) QA system. We have collected $5,495$ QA pairs from 500 articles covering various domains. Our analysis yields divers answer types and a significant proportion of questions that require some reasoning ability to solve. We expect that MMQA will facilitate research in multilingual QA, involving Indian languages. We have also built a deep CNN-RNN based model for question classification. Our scoring based answer extraction module will serve as a useful baseline for further research. In future, we would like to extend the dataset by adding more QA pairs from various languages and different types of questions such as list and complex questions. We would also like to propose an end-to-end model for multilingual QA in the near future.

Acknowledgements

Asif Ekbal greatfully acknowledges the Young Faculty Research Fellowship (YFRF) Award, supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

7. Bibliographical References

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October. Association for Computational Linguistics.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

Gupta, D. K., Kumar, S., and Ekbal, A. (2014). Machine learning approach for language identification & transliteration. In *Proceedings of the Forum for Information Retrieval Evaluation*, pages 60–64. ACM.

Hecht-Nielsen, R. (1992). Neural networks for perception (vol. 2). chapter Theory of the Backpropagation Neural Network, pages 65–93. Harcourt Brace & Co., Orlando, FL, USA.

Hovy, E., Gerber, L., Hermjakob, U., Lin, C.-Y., and Ravichandran, D. (2001). Toward semantics-based answer pinpointing. In *Proceedings of the first international conference on Human language technology research*, pages 1–7. Association for Computational Linguistics.

Joho, H. (1999). *Automatic detection of descriptive phrases for Question Answering System: A simple pattern matching approach*. Ph.D. thesis, University of Sheffield, Department of Information Studies.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Knott, A., Bayard, I., de Jager, S., Smith, L., Moorfield, J., and O'Keefe, R. (2001). A question-answering system for english and ma_ri. In *Proceedings of the Fifth Biannual Conference on Artificial Neural Networks and Expert Systems (ANNES), University of Otago*, pages 223–228.

Kumar, P., Kashyap, S., Mittal, A., and Gupta, S. (2005). A hindi question answering system for e-learning documents. In *Intelligent Sensing and Information Processing, 2005. ICISIP 2005. Third International Conference on*, pages 80–85. IEEE.

Li, X. and Roth, D. (2002). Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics, COLING 2002*, pages 1–7. Association for Computational Linguistics.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, et al., editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pages 807–814, USA. Omnipress.

Pamela, F., Danilo, G., Bernardo, M., Anselmo, P., Rodrigo, Á., and Sutcliffe, R. (2010). Evaluating multilingual question answering systems at clef. In *Seventh conference on International Language Resources and Evaluation (LREC'10)*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002).

Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Reddy, R. R. N. and Bandyopadhyay, S. (2006). Dialogue based question answering system in telugu. In *Proceedings of the Workshop on Multilingual Question Answering*, pages 53–60. Association for Computational Linguistics.

Sahu, S., Vasnik, N., and Roy, D. (2012). Prashnottar: A hindi question answering system. *International Journal of Computer Science & Information Technology*, 4(2):149.

Sekine, S. and Grishman, R. (2003). Hindi-english cross-lingual question-answering system. *ACM Transactions on Asian Language Information Processing (TALIP)*, 2(3):181–192.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

Stalin, S., Pandey, R., and Barskar, R. (2012). Web based application for hindi question answering system. *International Journal of Electronics and Computer Science Engineering*, 2(1):72–78.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics.

Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., and Suleman, K. (2016). Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Xiao, Y. and Cho, K. (2016). Efficient character-level document classification by combining convolution and recurrent layers. *CoRR*, abs/1602.00367.

Zaragoza, H., Craswell, N., Taylor, M. J., Saria, S., and Robertson, S. E. (2004). Microsoft cambridge at trec 13: Web and hard tracks. In *TREC*, volume 4, pages 1–1.

The First 100 Days: A Corpus Of Political Agendas on Twitter

Nathan Green, Septina Larasati
Marymount University, Charles University
Arlington Virginia, Prague Czech Republic
ngreen@marymount.com, septina.larasati@gmail.com

Abstract

The first 100 days corpus is a curated corpus of the first 100 days of the United States of America's President and the Senate. During the first 100 days, the political parties in the USA try to push their agendas for the upcoming year under the new President. As communication has changed this is primarily being done on Twitter so that the President and Senators can communicate directly with their constituents. We analyzed the current President along with 100 Senators ranging the political spectrum to see the differences in their language usage. The creation of this corpus is intended to help Natural Language Processing (NLP) and Political Science research studying the changing political climate during a shift in power through language. To help accomplish this, the corpus is harvested and normalized in multiple formats. As well, we include gold standard part-of-speech tags for selected individuals including the President. Through analysis of the text, a clear distinction between political parties can be found. This analysis shows the important item of their political agendas during the first 100 days of a new party in power.

Keywords: corpus construction, politics, social media

1. Introduction

Political communication in the United States has changed dramatically over recent years. Gone are the fireside chats on the radio, open press conferences on television are becoming less frequent. The current form of communication of politicians to their constituents and to other politicians is Twitter (Conway et al., 2015; Perlmutter, 2008). A new administration's public agenda would have once been in the purview of the media but now it is in more control of the campaign using new media such as Twitter (L. Towner and Dulio, 2012). While we will examine the 2017 USA government, this same phenomenon has been shown in other countries as well during recent elections (Dang-Xuan et al., 2013)

2. Background

Studying politics and political power shifts has been a difficult problem due to limited resources (Menini and Tonelli, 2016) and lack of examples during the age of social media. Shifts in political ideology tend to happen over longer periods of time. In the last United States election, the executive branch (President) switched from a Democratic President to a Republican administration. This is considered a switch from the political left to the political right. Two years prior in 2015 the United States' Senate switch in the same direction. This was the first time since 2009 that all branches of government were under the same party's power when all branches were run by the Democratic party. The previous time this had happened for Republicans was in 2003, the first election after the 9/11 terror attacks and before that we would have to go all the way back to 1953. Given these long gaps, this is the first time we can examine their ability to push new agendas via social media.

There has been previous research in the area of social media and politics in the USA dealing with topic classification (Glava et al., 2017) and Summarization (Egan et al., 2016). These have lead to successful attempts to build models to determine someone's political ideology. Prior to social media analysis, research was conducted on modeling political agendas from Senate press releases (Grimmer,).

Twitter has been used as a source of corpus (Johnson and Goldwasser, 2016) and model building for some time (Cieliebak et al., 2017; Bermingham and Smeaton, 2011) but to the best of our knowledge, the 100 day corpus is the first to offer resources for tracking agendas in social media driven by a political power change.

3. Methodology

3.1. Corpus Construction

In order to produce the corpus, Twitter data was scrapped during the first 100 days of the new political term. This ranges from January 20, 2017 to April 29, 2017. The first 100 days in the USA is traditionally used to evaluate what is important to a new administration to evaluate if they can accomplish their main goals. For this reason, politicians are very active on social media trying to push what they consider to be most important to the public and their constituents.

The corpus is a collection of tweets of the new President along with 100 Senators. A tweet is a singular post made to the social media site. For each tweet we collect its:

- ID
- Tweet Text
- Date and Time Posted
- Number of times it was favorited
- Number of times it was retweeted (shared)
- From what software it was posted

All 101 Twitter accounts that are scrapped are labeled with their respective political party. In our release the corpus contains multiple variations (text only, lower cased, tokenized) and different formats. The corpus is available in raw data, comma separated, text only, and available with

part-of-speech (POS) annotation. The scripts used to generate, convert, and process this corpus will be made available with the data.

Along with the corpus, we include a list of each Senator and their approximate GPS coordinates for their home region that they represent. This information can be used to study which parts of the country are most active online and which Senators converse with which regions of the USA according their their @ references.

3.2. Corpus Annotation

The Corpus currently contains two annotations. Each folder contains 1 individual Twitter user (Senator or President) and has been annotated with their appropriate political party (Republican, Democrat, Independent). After that we hand annotated the POS tags for the President.

POS annotation was done using Brat (Stenetorp et al., 2012). For the time being all Senators are automatically annotated with their POS using python and the Natural Language Tool Kit(NLTK) (Loper and Bird, 2002). These will be hand annotated in future releases of the Corpus. Scripts have been included to convert back and forth from NLTK to Brat's stand off annotation format so researchers can convert the data into the format of their choice.

3.3. Metrics

To evaluate differences in Twitter users' language we use the log-likelihood ratio statistic (Dunning, 1993; Rayson and Garside, 2000). We only take results that are significant with a $p < .01$. We run this both on unigram and bigram phrases comparing the political parties as a whole.

To evaluate communications we form a network graph where each node is a Twitter user and each edge is a reference. The thicker an edge the heavier the weight and the more the communication was used via the @ symbol. The degree of a node indicates how many different Twitter users with which the politician communicates.

4. Results

Republicans have control of the Senate with 52 members. We include the President in the total for Republicans with 53 members. Democrats have the minority position with 46 members. There are two Independent Senators who both caucus with the Democrats so we have included them in the Democrat's contingent with 48 members.

In Table 1 we break down the basic statistics behind the corpus dividing most per political party with Republicans symbolized with an R and Democrats with a D. The corpus has a whole has over half a million tokens with Democrats providing the majority of the text despite being in the political minority.

In a 100 day period the Democrats provided 27.37% more tweets than the Republicans. Despite the Republican President being famous for the amount he tweets, the opposition party seems to be the most vocal. The breakdown in Table 2 and Table 3 seems to give us more hints at why this might be the case. Democrats did not only tweet more but they added more vocabulary to the corpus as well with a 12.29% larger vocabulary. This could be attributed to the complexity of their message or it could be attributed to being "off" message where the Republicans seem to use similar language across members.

Stats	**counts**
Tokens	574,095
Tokens (R)	233,547
Tokens (D)	340,548
Members (R)	53
Members (D)	48
Number of Tweets (R)	25,803
Number of Tweets (D)	33,986
Vocab Size (R)	38,498
Vocab Size (D)	43,540
Avg Number of Tweets (R)	486
Avg Number of Tweets (D)	708
Avg Length of Tweet (R)	9.05
Avg Length of Tweet (D)	10.02

Table 1: Corpus Stats for the First 100 days corpus

4.1. Political Word Usage

Republican	**Type**	**Democrat**	**Type**
gorsuch	NE	health	N
enjoyed	V	trump	NE
#senate	N	gop	NE
#jobs	N	must	AV
meeting	N	care	N
great	JJ	fight	V
obamacare	NE	stand	V
hearing	N	climate	N
foxnews	NE	muslimban	NE

Table 2: 10 Most Overused words per political party according to the log-likelihood ratio statistic. Type is our categorization of the term. NE (Named Entity), N (Noun), AV (Auxiliary Verb), V (Verb), JJ (Adjective)

Table 2 shows the most overused words by each political party during the first 100 days. This is calculated using the log-likelihood ratio statistic. Basic stop words are removed from the list so the end result indicates a truer objective of each party. The Republican party was coming off of an election win and seems to be focused on governing objectives. "gorsuch" refers to the Republicans first governing task of nominating and confirming a new Supreme Court Justice, Judge Neil Gorsuch. Democrats rarely used his name, while it was the Republicans most overused word, and in the end it was their first legislative achievement in the new Senate. Republicans continued with the governing theme with "#senate", "#jobs","meeting","hearing",and "obamacare" all possibly indicating they were trying to move past the election and get to work. There are some sentiment words included such as "enjoyed" and "great" indicating some post victory celebration. "foxnews" is the more conservative of the

news stations (Devaney, 2013) in the USA so presumable the Senators were referencing stories in their favor.

Democrats seem to be overusing words that indicate their platform from the previous President that they want to keep. One achievement of the previous administration is the Affordable Care Act (ACA), commonly refereed to as Obamacare. The top overused word from Democrats, "health", seems to indicate they are ready to protect their legacy on this issue or they are fearing changes to it. The two other issues that come up in their list are "climate" and "muslim-ban" both which, during the campaign, they were opposed to the new President's promises on these issues. While Republicans seem to move on from the election, Democrats seem to be referencing the opposing part via overusing "trump" and "gop", the nickname of the Republican Party". Democrats overuse some language associated with opposition such as "fight" and "stand" as well. We observed that the more overused words on this list tended to be Verbs, possibly indicating their willingness for action.

We observed that Republicans tended to overuse words that were Nouns, possibly indicating that they were pushing their overall objectives and agenda items. Meanwhile Democrats almost exclusively overused Verbs. This is likely an attempt to show their supporters that they are ready for action in response to their election loss.

Republicans	**Democrats**
look forward	we must
i enjoyed	health care
pleased to	millions of
the #senate	fight for
hearing on	this is
neil gorsuch	we need
judge gorsuch	climate change
meeting with	cuts to
congratulations to	will fight

Table 3: 10 Most Overused bi-grams per political party according to the log-likelihood ratio statistic

To go a little deeper into what each party is stressing we next looked at the bigrams overused by each party, which can be seen in Table 3. Once again we can see the winning party is very forward looking and overall with a positive sentiment. The bigrams still show they are pushing their main agenda item of confirming a new Supreme Court Justice, Neil Gorsuch. The Democrats, on the other hand, use language that indicates they are ready to take a stand for certain issues, "climate change" and "health care" in particular. As it turns out, these were two of the larger fights in the first 6 months of the new President's term.

In both Table 2 and Table 3, we evaluated overused terms. This indicates that both sides mentioned the term at least once. To see agenda items that one side said that the other one never mentioned we examine Table 4

We feel Table 4 shows the true polarizing views of each party. Republicans again are focusing on their first achievement on the Supreme Court and celebrating their win with "#inauguration2017". They also bring up issues that are purely on the conservative spectrum in the USA such as

Spoken by Republican	**Spoken by Democrat**
#confirmgorsuch	#trumpcare
#neilgorsuch	#aca
well-qualified	#nobannowall
#inauguration2017	#womensmarch
#iran	#broadbandprivacy
#marchforlife	#climatechange
#repealandreplace	#stopgorsuch

Table 4: Agenda words spoken by one party but not the other

- "#iran" : opposing the previous administration's deal with Iran.
- "#marchforlife" : traditionally a march against abortion
- "#repealandreplace" : a rallying cry for those who want to get rid of the Affordable Care Act (Obamacare)

The more polarizing agenda items on the Democrats side includes mostly alternative views to the previous views such as "#stopgorsuch" and "#climatechange". Along with these some new issues that did not previously come up appeared in the form of hashtags such as:

- "#trumpcare" and "#aca" : Democrats attempt to defend the Affordable Care Act while labeling new proposals as Trumpcare, a similar tactic done with the previous administration to label health care Obamacare
- "#nobannowall" : One January 27th the President signed an executive order that was consider by the Democrats to be a ban on Muslims. Additionally during the campaign, the President often declared he would build a wall between the USA and Mexico. This hashtag is in opposition to the new administration.
- "#womensmarch" : Also on January 27th a Woman's March took place where approximately 440,000 women took to the streets in DC and an estimate 5 million women marched around the world. In the USA this was seen as an opposition to the President who had made many remarks towards women in the campaign that were seen as negative.
- "#broadbandprivacy" : On April 3rd the President signed into law Senate Joint Resolution 34 which effectively allows telecommunication companies to sell private data of there customers. This hashtag appears to try to bring attention to that bill.

One noticeable difference across Table 2,3, and 4 is the consistency of the Republican party. Across all three segments (unigrams, bigrams, and agenda words), the Republican party consistently repeats the same themes and language. The democrats seem to change their

focus over the three areas and have a wider amount of topics that they cover. This corresponds to what we saw in Table 1 with Republicans using less vocabulary, less tweets, and less words per tweet. Their message appears to be short and on topic across most of the Senators. Alternatively, the Democrats seems to cover more topics , tweet more, and use more vocabulary to get their agenda across.

4.2. Mentions and Dialogs

The Twitter handles in this corpus represents real public figures with real public agendas that they want to push. We picked Democratic and Republican political party's tweets to analyze because of their opposing nature and it is the way they try to push their agenda and construct dialogs. This corpus allows us to analyze how the two sides connect and communicate using a social media platform.

Here we use Twitter mentions as a sign of a reference to initiate a conversation or dialog with another Twitter user. We collect all the mentions in the tweets and see how the communications are done in the senate.

Total	**To (D)**	**To (R)**	**To Other**
By (D)	548 / 11.42	767 / **15.98**	6101 / 127.10
By (R)	197 / **3.72**	468 / 8.83	6128 / 115.62
Unique	**To (D)**	**To (R)**	**To Other**
By (D)	328 / 6.83	201 / 4.19	3662 / 76.29
By (R)	132 / 2.49	220 / 4.15	3652 / 68.91

Table 5: Mention Count / Mention Average in 100 days period

There are total of 14,209 mentions in the corpus. Table 5 shows that on average each Democrat uses 15.98 mentions while Republican only uses 3.72 mentions to their opposing side in the 100 days period.

We assume a dialog happens when there is a mutual reference where senator A mentions Senator B and also the other direction (This may include a self-reference in a tweet).

Table 6 shows there are total of 239 dialogs that happened across party lines. The data also shows Democrats have more bi-directional conversation among Democrats compared to Republicans among Republicans.

	Democrat	**Republican**
Democrat	128	44
Republican	44	67

Table 6: Number of Dialog in the Tweets by Party

We draw these connections as graphs using an open source graph visualization tools, Gephi (Bastian et al., 2009). We represent the Twitter user as a node and a mention of a user B from a user A as an directed edge from node A to B. The edges are also weighted as how many mentions occur.

Since we also have the data of the states the Senators represent, we visualize the node based on their location. This way we can see geographically how the conversation happened in the 100 days. In the visualization, we can see that the mentions directed to other senators regardless of where they are and not bound geographically, although there are still some Senators that only have dialogs among the Senators from the same state, e.g. senators from Alaska and Mississippi.

The graph can also focus on a particular node and show how they are being mentioned and how they mention other senators as seen in Figure 1

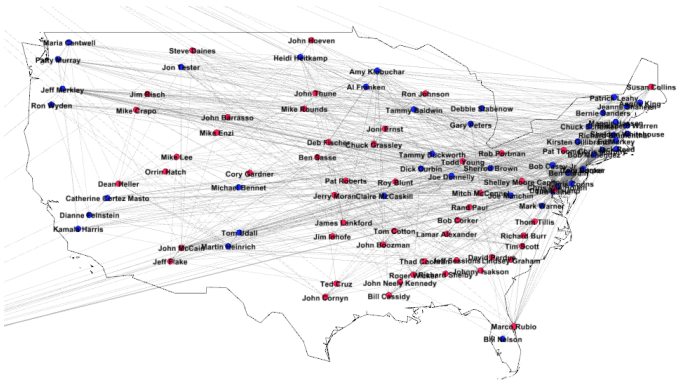


Figure 1: The caption of the figure.

We provide the .gephi Graph file and .csv node and edge files as part of the distributes corpus.

5. Conclusion

The 100 day corpus is unique in its slice of time and in its overall purpose. It is the first corpus dedicated to showing the change in language between opposing political factions during a change in power. Through corpus analysis we are able to pinpoint key agenda items, as well as differencing linguistic styles towards their persuasion.

The corpus includes over half a million tokens annotated for political party, location, and part-of-speech. The release of the corpus additionally contains the python scripts necessary to recreate the corpus, evaluate the corpus, and convert the corpus into a variety of working formats. The corpus will be available at the Marymount Data Fusion Center. With its availability we hope the community can gain further insight into the use of social media in the political realm.

6. Future Work

In the future we plan on continuing work on the corpus by making all Senators have gold standard POS annotations. We believe annotating dependency structure we will add additional insight into how linguistics can be used to push a political agenda. Annotations of sentiment will also aid in researchers ability to draw conclusions about this particular slice in political time.

7. Bibliographical References

Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: An open source software for exploring and manipulating networks.

Bermingham, A. and Smeaton, A. F. (2011). On using twitter to monitor political sentiment and predict election results.

Cieliebak, M., Deriu, J., Egger, D., and Uzdilli, F. (2017). A twitter corpus and benchmark resources for german sentiment analysis.

Conway, B., Kenski, K., and Wang, D. (2015). The rise of twitter in the political campaign: Searching for intermedia agenda-setting effects in the presidential primary. 20, 05.

Dang-Xuan, L., Stieglitz, S., Wladarsch, J., and Neuberger, C. (2013). An investigation of influentials and the role of sentiment in political communication on twitter during election periods. 16:795–825, 06.

Devaney, H. (2013). Perceptions of media bias: viewing the news through ideological cues, April.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.*, 19(1):61–74, March.

Egan, C., Siddharthan, A., and Wyner, A. Z. (2016). Summarising the points made in online political debates. In *ArgMining@ACL*.

Glava, G., Nanni, F., and Ponzetto, S. P. (2017). Cross-lingual classification of topics in political texts.

Grimmer, J. (). A bayesian hierarchical topic model for political texts: Measuring expressed agendas in senate press releases. In *In Proceedings of the First Workshop on Social Media Analytics, SOMA ?10*.

Johnson, K. and Goldwasser, D. (2016). "all i know about politics is what i read in twitter": Weakly supervised models for extracting politicians' stances from twitter. In *COLING*.

L. Towner, T. and Dulio, D. (2012). New media and political marketing in the united states: 2012 and beyond. 11:95–119, 01.

Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA. Association for Computational Linguistics.

Menini, S. and Tonelli, S. (2016). Agreement and disagreement: Comparison of points of view in the political domain. In *COLING*.

Perlmutter, D. D. (2008). Political blogging and campaign 2008: A roundtable. *The International Journal of Press/Politics*, 13(2):160–170.

Rayson, P. and Garside, R. (2000). Comparing corpora using frequency profiling. In *Proceedings of the Workshop on Comparing Corpora - Volume 9*, WCC '00, pages 1–6, Stroudsburg, PA, USA. Association for Computational Linguistics.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: A web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

Medical Sentiment Analysis using Social Media: Towards building a Patient Assisted System

Shweta Yadav, Asif Ekbal, Sriparna Saha, Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology Patna
Patna, Bihar, India
{shweta.pcs14, asif, sriparna,pb}@iitp.ac.in

Abstract

With the enormous growth of Internet, more users have engaged in health communities such as medical forums to gather health-related information, to share experiences about drugs, treatments, diagnosis or to interact with other users with similar condition in communities. Monitoring social media platforms has recently fascinated medical natural language processing researchers to detect various medical abnormalities such as adverse drug reaction. In this paper, we present a benchmark setup for analyzing the sentiment with respect to users' medical condition considering the information, available in social media in particular. To this end, we have crawled the medical forum website 'patient.info' with opinions about medical condition self narrated by the users. We constrained ourselves to some of the popular domains such as *depression*, *anxiety*, *asthma*, and *allergy*. The focus is given on the identification of multiple forms of medical sentiments which can be inferred from users' medical condition, treatment, and medication. Thereafter, a deep Convolutional Neural Network (CNN) based medical sentiment analysis system is developed for the purpose of evaluation. The resources are made available to the community through LRE map for further research.

Keywords: Medical sentiment, Medical blog, Machine learning, Deep learning, Convolutional Neural Network

1. Introduction

Attention towards sentiment analysis has been flourishing over the last two decades because of the immense popularity of social media. The phenomenal rise in blogging trend is observed in health communities such as medical forums which are swamped by millions of users (many of whom are patients) seeking for health-related information, sharing medical problems or experiences and opting for informational support or opinions from the other users (patients, health-professional or doctors). These self-narrated texts provide a platform to peek into a blogger's state-of-the-mind for several reasons:
(i) the subjective nature of the contents generated by blogger's; **(ii)** the temporal aspect of the blog which can be formed into thread; **(iii)** the abundance of blog data which allows cumulation of opinions, sentiments and thoughts in a very wide spectrum.
Medical sentiment analysis has its major applications in assessing the clinical records and in providing an automated decision support system for health professional. According to the study conducted by the Pew Internet & American Life Project[1], almost 80 percent of Internet users in US have explored health-related topic online. More often, people look for the information about specific medical problem (63%) over the internet. Nearly 47% of the users search for the medical treatment or procedure in the internet. With such a tremendous amount of freely available medical texts in the web, it is necessary to harness the crucial and important information. Analyzing these texts by capturing the sentiments is helpful because opinions are central to almost all human activities and are key influencer of our behaviors. Although, several techniques exist to capture sentiments in general domains, the sentiments expressed in medical narratives have not been well analyzed and exploited in the required measure as yet. The research in medical sentiment analysis mainly focuses on biomedical literature and Electronic Medical Record (EMR) documents. Recently, preliminary study was conducted by (Denecke and Deng, 2015) to capture the medical sentiment from clinical narratives and medical social media sources. Several shared-task challenges (Losada et al., 2017; Hollingshead et al., 2017) have also been conducted to study the social media texts to capture the user's opinion in medical setting. For this purpose, they generated domain-specific corpus containing clinical documents (nurse letters, radiology reports and discharge summaries) collected from the MIMIC II database. Furthermore, they extracted drug reviews from medical blogs such as WebMD, DrugRating. This study provides the quantitative assessment of sentiment in terms of six corpora with 1000 documents.
Literature survey shows that medical sentiment analysis, nowadays, is a topic of growing research interest. In this work, we have explored how sentiment analysis from medical forums can be effective for building a patient assisted healthcare system. We have provided a benchmark setup for mining patient opinions extracted from the medical web forums. Towards this end, we have studied different aspects of medical sentiment in health related texts that may relate to the following:

- **Health status:** Alteration in the health status which may vary over a particular time period.
- **Degree of medical condition that impacts patient life** (e.g., *severe headache impacts the patient's life more than the mild headache.*)
- **Consequence of a treatment** (e.g., there may be positive or negative impacts in a patient's treatment, such

[1] http://www.pewinternet.org/

Classification 1: Medical Condition			Classification 2: Medication	
Medical Blog	Label		Medical Blog	Label
This morning I had an **attack** of it that was **very sudden and very intense**. I felt an **incredible surge of unsteadiness**.	Exist		Hi been on Sertaline now for abut 4 weeks. Maybe nearer 5. I started on 25mg and now been on 50 for around 2 or 3 weeks. My **mood** has **definitely improved** and I am **alot calmer**	Effective
Previously I have taken flixonase and beconase which has given **no long term relief** 10 days ago I went back to the doctor and was given Betnesol. This has **immediately relieved me all symptoms**.	Recover		Had anxiety for few months on citalopram propanolol and 2mg diazapam. Took first diazapam today as I have health anxiety and was **scared** to take them. **Nothing seems to help been in bed for two days can't sleep** waking at 3 4 5 am	Ineffective
I recently started lexapro 3 days, I'm absolutely lost I feel **weak and shaky** everyday and **can't eat right** I don't sleep normal. I'll **die young** and the cause will be cardiac arrest	Deteriorate		I've been feeling a bit off still.. Day 6 that I haven't taken my citalopram. Anxiety is down, but now I'm starting to **feel more and more off.. Random high chest pains** Plus feeling a **bit foggy and spacey**..	Serious adverse effect (SAE)

Table 1: Exemplar description of annotation scheme. The words in bold represent possible sentiments.

as *'flixonase and beconase which has given no long term relief'*.)

- **Opinions towards a treatment** (e.g., a patient can have an adverse reaction after consumption of drug)
- **Certainty of a diagnosis:** (e.g., Health professional can be certain of some diagnosis.)

Medical sentiment can be studied at the various aspects like 'medical condition', 'treatment', 'procedure', etc. which can directly impact the users' health conditions. By analyzing patient status periodically, any progress or deterioration can be identified. Any user expresses his/her medical condition implicitly or explicitly. Implicit sentiment in medical context concerns mentioning of the symptoms, for instance consider the blog: *'I recently started **lexapro** 3 days, I'm on **extreme weight loss***. Here, '**weight loss** as in such does not reflect anything negative but in the above sentence, it represents the adverse drug reaction where sentiment is implicitly defined to be negative. These require additional information for making correct interpretation. In case of explicit sentiment, it is relatively much easier to analyze the health conditions. For example, consider "*I recently started **lexapro** 3 days, I'm **absolutely lost** I feel **weak and shaky** everyday". Here, **absolutely lost, weak and shaky*** provide symptoms which are explicitly defined.

For this work, we have collected a corpus consisting of $7,490$ user blog posts from popular medical forum 'patient.info' which is split on the basis of two major medical sentiment aspects, namely 'medical condition' and 'treatment'. The corpus is manually annotated with a predefined set of categories. Finally a Convolutional Neural Network (CNN) based model is developed for medical sentiment classification.

To the best of our knowledge, there is no existing benchmark setup available for medical sentiment analysis. We believe that creating such a resource might be beneficial for building a patient assisted healthcare monitoring system. Below the contributions of our work are summarized:

1. Introduction of a novel annotation scheme for analyzing medical sentiment;
2. Development of an annotated medical sentiment corpus;
3. Building a deep CNN based medical sentiment classification system;
4. More deeper analysis of the sentiments with respect to medical domain.

The paper is structured as follows: Related works is presented in Section 2. The annotation scheme is introduced in Section 3. Section 4 discusses the method for corpus collection and annotation. In Section 5, we describe our method for capturing the medical sentiments. The results of the annotation study are presented in Section 6. At the end of the paper, we provide pointers for future work.

2. Related Works

Recent years have witnessed rapid proliferation in research on identifying and assembling subjective expressions or other non-factual expressions of textual contents characterizing peoples' opinions, feelings or emotions using medical blog texts. In general, we can categorize the existing works utilizing medical blog texts into three groups as follows:

- **Polarity Classification:** Some of the popular works include the studies carried out by Xia et al. (2009) which aimed to classify the patient opinions in eight categories and observed its polarity (positive, negative). Sokolova et al. (2011) also focused on classifying the tweets on the basis of sentiment (positive, negative and neutral). They used bag-of-words (BoWs) as features to learn several classifiers such as naive Bayes, decision trees and support vector machines. Study conducted by Biyani et al. (2013) used online cancer community user data to determine the polarity. They have adapted supervised machine learning techniques using hand-crafted features, which cover both domain-dependent as well as domain-independent features. They identified sentiments on two discourse

functions such as expressive and persuasive. A supervised machine learning model (multi-nominal naive Bayes) is developed using frequency-based features.

- **Adverse Drug Relation:** For medical domains, social media texts (corresponding to medical forums) have been utilized in the works such as DS (Leaman et al., 2010; Nikfarjam and Gonzalez, 2011; Liu and Chen, 2013), MedHelp (Yang et al., 2012) and PatientsLikeMe (Wicks et al., 2011). Non-medical social media forums like Twitter (Nikfarjam et al., 2015) have been exploited to capture adverse drug effect. With the availability of the extensive Adverse Drug Reaction (ADR) lexicons such as Side Effect Resource (SIDER)[2] (Kuhn et al., 2010), Coding Symbols for a Thesaurus of Adverse Reaction Terms (COSTART), Consumer Health Vocabulary (CHV) (Zeng-Treitler et al., 2008) and Medical Dictionary for Regulatory Activities (MedDRA) (Mozzicato, 2009), some prominent studies (Leaman et al., 2010; Yates and Goharian, 2013) focus on exploiting these pre-existing lexicons to identify ADR mentions in user posts. Some of the other popular studies include the works of (Na et al., 2012; Sharif et al., 2014) utilizing machine learning based NLP techniques to identify the ADR.

- **Emotion Classification:** Sokolova and Bobicev (2013) studied different forms of emotions from medical web documents. They analyzed the categories such as encouragement (hope, happiness), confusion (worry, doubt, concern), gratitude, facts, and facts+encouragement. They applied naive Bayes classifier with the features derived from lexicon WordNetAffect. Study conducted by Melziet et al. (2014) on emotion classification learned SVM using the feature set consisting of BoWs, n-grams and specific attributes.

3. Benchmarking and Annotation Scheme

In this section, we define the benchmark setup by studying the sentiment expressed in medical blog posts. Here, we focus on fine-grained medical sentiment aspects of the users' health status and treatment. Our intention is that the annotation scheme should be able to capture multiple perspectives of user health status. Below we provide two important medical aspects with the possible categories of sentiment values:

Classification 1: Medical Problem: Exist, Recover, Deteriorate.

Classification 2: Medication: Effective, Ineffective, Serious adverse effects.

We categorize medical problems into the following three possible sentiment classes:
Exist: Here user shares the symptoms (negative sentiment) of any medical problem.
Recover: The user shares the recovering status (positive sentiment) from the previous medical problems.
Deteriorate: The user describes its medical condition to be worsen (negative sentiment) over the span of medical treatment.

The other classification strategy concentrates on the effect of the medication. We describe below the possible sentiment values :
Effective: User shares the positive sentiment in the form of usefulness of the treatment.
Ineffective: The no effect of the treatment is reported in the user narration.
Serious adverse effect: User shares the negative opinion towards the treatment mainly in the form of adverse drug effect. The blog post falling under this category has to have the explicit mention of the drug name in the text.
From the examples as presented in Table-1, we analyze that sentiment in clinical narratives cannot always be manifested in single terms or phrases, rather it heavily depends on the context. The concept of medical sentiment is very complex and has multiple facets making it very interesting, but also challenging for automatic analysis.

4. CMS: Corpora for Medical Sentiment

Attributed to the fact of growing interest in users' self stated medical reviews, we crawl the medical forums where multiple users discuss on various medical conditions. We consider the following points while selecting the source of information from which to extract the corpus:

- It should be extremely popular and reliable site in search of medical issues with reasonable number of users.

- There should exist fair number of opinions which must either have discussions on medical conditions or medications.

In order to obtain potential and effective sources which satisfy the above requirements, we did exhaustive search exploiting multiple medical forums. The task was quite tedious as most of the forums either do not have sufficient number of users or the text was heavily noisy. After surveying several websites, we chose the 'patient.info' [3] medical forum. This forum contains on an average 1500 opinions per medical discussion group. We selected popular discussion groups such as *Anxiety*, *Depression*, *Asthma* and *Allergy* having $5,000$ blog posts on an average. In total we collected $10,000$ blog posts of which $5,188$ posts concern about the medical conditions and $2,302$ contain medication related blog posts which were collected during the period of 25^{th} September 2016 to 15^{th} November 2016. We removed $2,510$ blog-posts which did not have any mention of medication or medical condition. To ensure the confidentiality of user, all the user related information were removed. The statistics of corpus are presented in Table-2 and Table-3. A team of three expert annotators independently annotated the user posts with three classes on both the classification strategies. The Cohen's kappa approach (Cohen, 1960) was

[2]http://sideeffects.embl.de/

[3]https://patient.info/

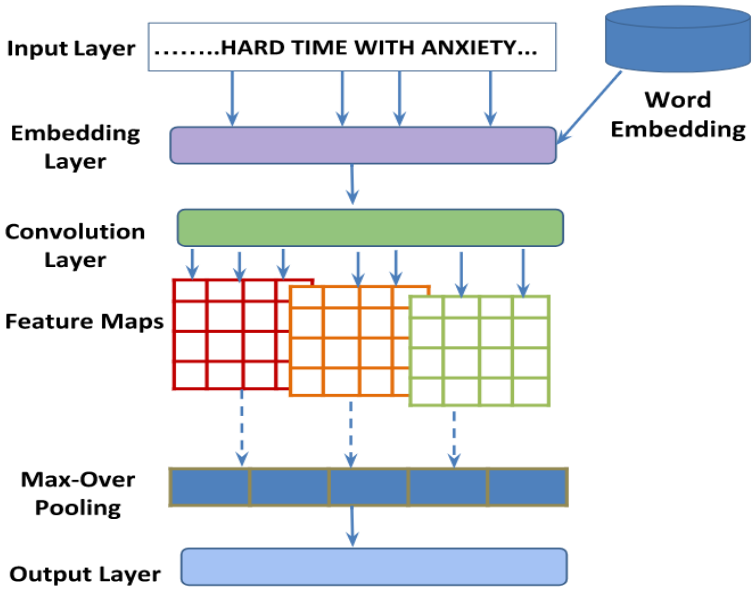


Figure 1: Proposed architecture for predicting the medical sentiment from blog-post.

Classification 1: Medical Condition				
Exist	Recover	Deteriorate	Avg # of sentences	Avg # of words
2396	703	2089	10	192

Table 2: Dataset statistics for classification-1

Classification 2: Medication				
Effective	Ineffective	Serious Adverse Effect	Avg # of sentences	Avg # of words
462	613	1,226	9	176

Table 3: Dataset statistics for classification-2

used to measure the inter-annotator agreement. We observe high agreement ratio of 0.79 for exact matching of the class with respect to each blog post.

5. Approach for Capturing Medical Sentiment

In this section we have presented the approach developed for extracting sentiments of users' posts in medical blogs.

5.1. Network for Identifying Severity Level

In this section we propose a method based on CNN that exploits sentiments from health forums (or, medical blogs) in augmentation layer. As presented in Figure-1, the proposed model has four different components which are similar to the conventional CNN components as proposed by (Kim, 2014). The first layer represents the input layer which takes a complete blog post in the form of vector representation (word embedding) and outputs a probability corresponding to the classification types. We use max-pooling over the whole blog post to obtain global features through all the filters. This pooled feature is fed into the fully connected neural network. In the output layer, we use the softmax classifier to automatically classify the post into three output classes. We describe below the layers of our proposed model in details:

1. **Input layer:** Each blog post is provided as the input to the model.

2. **Word embedding layer:** This layer encodes every word into a real-valued vector. Given a blog text M consisting of n words $w_1, w_2, w_3, ...w_n$, each word w_i is transformed into real-valued vector x_i. Each word in M is looked up in the corresponding word embedding matrix $W \in R^{k \times |V|}$, where $|V|$ represents fixed length vocabulary and k is the word embedding size. The blog-post representation matrix $x_{1:n_W}$ can be constituted as:

$$x_{1:n_W} = x_1 \otimes x_2 \ldots \otimes x_{n_W} \tag{1}$$

where $\otimes$ represents the concatenation operator. We perform zero padding in case the number of the words in blog text is less than n to fix the length.

3. **Convolution layer:** Word embedding is fed as the input to the convolutional layer where filter $\mathbf{F} \in \mathbb{R}^{m \times k}$ is convoluted to the context window $x_{i:i+m-1}$ of h words for each blog-post as follows.

$$c_i = f(\mathbf{F}.x_{i:i+m-1} + b) \tag{2}$$

where f is non-linear function[4] and b is a bias term. The feature map f is generated by applying given filter $\mathbf{F}$ to every potential window of word in the blog-post.

$$\begin{aligned} f = [&g(\mathbf{F} \cdot x_{1:1+h-1} + b), g(\mathbf{F} \cdot x_{2:1+h-1} \\ &+ b) \ldots g(\mathbf{F} \cdot x_{n-h+1:n} + b) \\ = [&f_1, f_2, f_3 \ldots . f_{n-h+1}] \end{aligned} \tag{3}$$

[4] In our experiments we have used the rectified linear unit as a non linear function.

Classification Strategy	Classification Models		Precision	Recall	F1-Score
Medical Condition	Baselines	SVM	0.42	0.49	0.43
		Random Forest	0.45	0.48	0.46
		MLP	0.41	0.43	0.40
	CNN		0.68	0.60	0.63
Medication	Baselines	SVM	0.74	0.76	0.75
		Random Forest	0.72	0.73	0.73
		MLP	0.74	0.75	0.74
	CNN		0.86	0.77	0.82

Table 4: Performance comparison of CNN architecture with other baseline classifiers

Effective	**Ineffective**	**Adverse Drug Effect**
feeling wonderful More energy	feeling down Moods fluctuating	feel odd sensations skin
feel like normal person now	feel like death! feel	feeling like drunk night
feel most comfortable pacing	feel totally hopeless almost	feel horrible dizzy sickly
feel great better than done	feel really down and hard	feel super nauseous sleeping

Table 5: The set of informative 4-grams over different classes of Medication category

In order to increase the coverage of n-gram model, multiple filters with different window sizes can be applied.

4. **Pooling layer:** The function of the pooling layer is to gradually minimize the spatial size of the representation by identifying the most abstracted feature generated by the convolutional layer. It involves non-linear down sampling to extract most relevant set of features. In our work, we apply max-pooling operation over feature map and set the maximum value as a feature for this particular filter. The max-pooling operation is performed over feature map as follows:

$$\hat{d} = max(f_1, f_2, f_3.....f_{n-h+1}) \quad (4)$$

5. **Output layer:** The blog-level feature vector is passed to the softmax layer to label 'y' from a discrete set of classes for the corresponding blogs 'M'.

5.2. Hyperparameter Settings in CNN

The values of hyper-parameters are determined from preliminary experiments by evaluating the model's performance using 5-fold cross validation by varying the convolution feature sizes (100, 200 & 300). Word embedding is generated through pre-trained Google news word embedding model. Specifically, all the deep learning models use the 300-dimension word embedding, feature map size of 300 on multiple filters with window sizes of 3, 4, 5. We use Adam (Kingma and Ba, 2014) as our optimization method with a learning rate of 0.001. Training was performed using stochastic gradient descent over mini-batches considering the Adadelta (Zeiler, 2012) update rule. As a regularizer, we use dropout (Hinton et al., 2012) with a probability of 0.5. After training, we choose the best performing model to be evaluated on the test sets. The model introduced in this paper is implemented on Theano [5].

[5]http://deeplearning.net/software/theano/

6. Experimental Results and Analysis

To evaluate the effectiveness of our algorithm, we have developed three strong baselines models:

Baseline 1: The first baseline model is constructed by training SVM (Cortes and Vapnik, 1995).

Baseline 2: In this model, we use Random Forest (Breiman, 2001) based classification model.

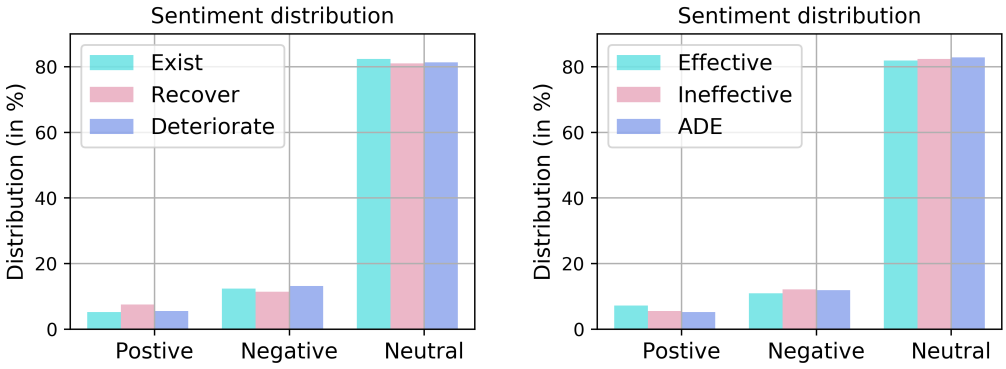
Baseline 3: Multi-layer perceptron (MLP) (Collobert and Bengio, 2004) is utilized to learn the model. In order to learn the baseline classifiers, we used the following feature set which is specific to the forum data.

- **N-grams:** This feature plays a very important role in capturing the contextual information of the blog. We generated uni-grams, bi-grams, tri-grams with respect to the target words within the window size of $[-2, 2]$.
- **Medical abbreviated feature:** Generally, users tend to use abbreviated wordforms to describe medical condition or treatment for e.g., ECG/EKG for Electrocardiogram. We created medical abbreviation dictionary by crawling medical acronym and abbreviation related website[6]. We generated binary feature which sets the feature value to 1, if the target word is present in the dictionary else the value is set to 0.
- **Sentiment feature:** We designed three real valued features which compute the positive, negative and neural sentiment scores of the blog by finding number of positive, negative and neutral words in a document. The sentiment score was calculated by using most recent and popular lexicon, SentiWordnet [7]. These three sentiment scores were calculated by the following equation:

$$Score^{(K)}(blog) = \sum_{i=1}^{n} SC^{(K)}(w_i)/n \quad (5)$$

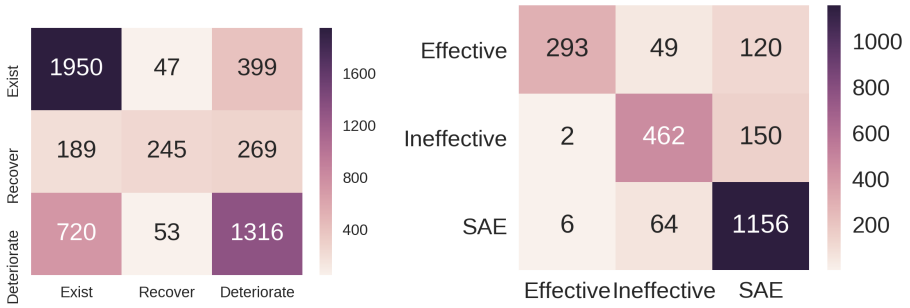
[6]http://www.health.am/acronyms/

[7]http://sentiwordnet.isti.cnr.it/



(a) Medical Condition (b) Medication

Figure 2: Sentiment word distribution through SentiWordNet



(a) Medical Condition (b) Medication

Figure 3: Confusion Matrix for both the classification strategies

Here $SC^{(K)}$ denotes the sentiment score of the word for k^{th} sentiment where $K \in \{+, -, neutral\}$ and n is the number of words (w) in a medical blog.

We have reported the results obtained by our CNN based sentiment classification model along with other baseline models in Table-4. The CNN system that uses only the pre-trained embedding achieves 63% and 81% F1-Score values on *medical condition* and *medication* classification strategies, respectively. The confusion matrix for both types of classification problems is presented in Figure-3. Our CNN based model obtains significant performance improvements over all the three baselines for both the classification strategies. Feature ablation experiments are also conducted to analyze the importance of different features selected. Analysis shows that medical abbreviation and sentiment scores are not effective features in medical setting.

6.1. Major Analysis

Our analysis on the user-generated medical blog reveals that the usual health status information is presented in an elusive way by the user. The word usage in the medical blog is more implicit and requires deeper analysis of metaphor and sarcasm. We have illustrated these scenarios in Figure-2, where our system was unable to capture the implicit negative or positive sentiment present in the users' posts and thus the posts were classified into neutral.

The general SentiWordNet(SWN) lexicon is observed to be not assisting the system in capturing the sentiment in medical setting. For example, consider a text from medical blog-post

"all the sudden my heart like drops and feels like its going to stop.".

Here, the phrase 'heart like drops' and 'going to stop' are user's narrated symptoms presenting the examples for implicit negative sentiment. However, the SWN lexicon provides the label neutral to these words as in general these do not carry any positive or negative sentiment.

After deep analysis of data, we observed that majority of the medical sentiment occurs in the vicinity of the term 'feel' and its variations. We have generated the 4-grams taking 'feel' as the target word and as shown in Table-5, we have observed that these 4-gram words can provide an effective clue in capturing the sentiment. Further, more semantics and context-dependent features are required to capture the peculiarity of the medical blog text.

We have also observed that deeper understanding of sentiment in MS analysis further requires consideration of the context which may not be available on the blog. For example, problems with a 'head' can be captured with multiple symptoms: 'headache, nausea, fever'. Thereby it is highly required to utilize background knowledge in order to cluster these symptoms to the similar medical condition.

7. Conclusions and Future Works

We have presented a large corpus of annotated data collected from the 'Patient.info' forum containing users' original posts written on the 'Anxiety', 'Depression', 'Asthma'

and 'Allergy' forums. This paper provides fine-grained annotation scheme to capture the sentiment in medical setting which concentrates on detailed medical aspects such as 'medication' and 'medical condition' instead of conventional polarity (positive or negative) to judge user's health status. We have also presented a deep convolutional neural network based classification framework to predict the possible medical sentiment category for both 'medication' and 'medical condition' classification schemas. We are able to obtain significant performance improvements over the baseline in all the cases. In future, we aim to develop the medical sentiment specific lexicon and would like to propose a method to capture implicit, metaphoric & sarcastic phrases.

8. Acknowledgements

Asif Ekbal greatfully acknowledges the Young Faculty Research Fellowship (YFRF) Award, supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

9. Bibliographical References

Biyani, P., Caragea, C., Mitra, P., Zhou, C., Yen, J., Greer, G. E., and Portier, K. (2013). Co-training over domain-independent and domain-dependent features for sentiment analysis of an online cancer support community. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 413–417. ACM.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32, Oct.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Collobert, R. and Bengio, S. (2004). Links between perceptrons, mlps and svms. In *Proceedings of the twenty-first international conference on Machine learning*, page 23. ACM.

Cortes, C. and Vapnik, V. (1995). Support vector machine. *Machine learning*, 20(3):273–297.

Denecke, K. and Deng, Y. (2015). Sentiment analysis in medical settings: New opportunities and challenges. *Artificial intelligence in medicine*, 64:17–27.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Kristy Hollingshead, et al., editors. (2017). *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology — From Linguistic Signal to Clinical Reality*. Association for Computational Linguistics, Vancouver, BC, August.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Kuhn, M., Campillos, M., Letunic, I., Jensen, L. J., and Bork, P. (2010). A side effect resource to capture phenotypic effects of drugs. *Molecular systems biology*, 6(1):343.

Leaman, R., Wojtulewicz, L., Sullivan, R., Skariah, A., Yang, J., and Gonzalez, G. (2010). Towards internet-age pharmacovigilance: extracting adverse drug reactions from user posts to health-related social networks. In *Proceedings of the 2010 workshop on biomedical natural language processing*, pages 117–125. Association for Computational Linguistics.

Liu, X. and Chen, H. (2013). Azdrugminer: an information extraction system for mining patient-reported adverse drug events in online patient forums. In *International Conference on Smart Health*, pages 134–150. Springer.

Losada, D. E., Crestani, F., and Parapar, J. (2017). Clef 2017 erisk overview: Early risk prediction on the internet: Experimental foundations.

Melzi, S., Abdaoui, A., Azé, J., Bringay, S., Poncelet, P., and Galtier, F. (2014). Patient's rationale: Patient knowledge retrieval from health forums. In *eTELEMED: eHealth, Telemedicine, and Social Medicine*, volume 2014, page 140.

Mozzicato, P. (2009). Meddra. *Pharmaceutical Medicine*, 23(2):65–75.

Na, J.-C., Kyaing, W. Y. M., Khoo, C. S., Foo, S., Chang, Y.-K., and Theng, Y.-L. (2012). Sentiment classification of drug reviews using a rule-based linguistic approach. In *International Conference on Asian Digital Libraries*, pages 189–198. Springer.

Nikfarjam, A. and Gonzalez, G. H. (2011). Pattern mining for extraction of mentions of adverse drug reactions from user comments. In *AMIA Annu Symp Proc*, volume 2011, pages 1019–1026.

Nikfarjam, A., Sarker, A., O'Connor, K., Ginn, R., and Gonzalez, G. (2015). Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, page ocu041.

Sharif, H., Abbasi, A., Zafar, F., and Zimbra, D. (2014). Detecting adverse drug reactions using a sentiment classification framework. In *Proceedings of the sixth ASE international conference on social computing (SocialCom)*.

Sokolova, M. and Bobicev, V. (2011). Sentiments and opinions in health-related web messages. In *RANLP*, pages 132–139.

Sokolova, M. and Bobicev, V. (2013). What sentiments can be found in medical forums? In *RANLP*, volume 2013, pages 633–639.

Wicks, P., Vaughan, T. E., Massagli, M. P., and Heywood, J. (2011). Accelerated clinical discovery using self-reported patient data collected online and a patient-matching algorithm. *Nature biotechnology*, 29(5):411–414.

Xia, L., Gentile, A. L., Munro, J., and Iria, J. (2009). Improving patient opinion mining through multi-step classification. In *International Conference on Text, Speech and Dialogue*, pages 70–76. Springer.

Yang, C. C., Yang, H., Jiang, L., and Zhang, M. (2012). Social media mining for drug safety signal detection. In *Proceedings of the 2012 international workshop on Smart health and wellbeing*, pages 33–40. ACM.

Yates, A. and Goharian, N. (2013). Adrtrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites. In *European Conference on Information Retrieval*, pages 816–819. Springer.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zeng-Treitler, Q., Goryachev, S., Tse, T., Keselman, A., and Boxwala, A. (2008). Estimating consumer familiarity with health terminology: a context-based approach. *Journal of the American Medical Informatics Association*, 15(3):349–356.

An Italian Twitter Corpus of Hate Speech against Immigrants

Manuela Sanguinetti⋆, Fabio Poletto⋆, Cristina Bosco⋆, Viviana Patti⋆, Marco Stranisci⋄
⋆Dipartimento di Informatica, University of Turin, Italy
⋄ACMOS, Turin, Italy
fabio.poletto@edu.unito.it, {msanguin, bosco, patti}@di.unito.it,marco.stranisci@acmos.net

Abstract

The paper describes a recently-created Twitter corpus of about 6,000 tweets, annotated for hate speech against immigrants, and developed to be a reference dataset for an automatic system of hate speech monitoring. The annotation scheme was therefore specifically designed to account for the multiplicity of factors that can contribute to the definition of a hate speech notion, and to offer a broader tagset capable of better representing all those factors, which may increase, or rather mitigate, the impact of the message. This resulted in a scheme that includes, besides hate speech, the following categories: aggressiveness, offensiveness, irony, stereotype, and (on an experimental basis) intensity. The paper hereby presented namely focuses on how this annotation scheme was designed and applied to the corpus. In particular, also comparing the annotation produced by CrowdFlower contributors and by expert annotators, we make some remarks about the value of the novel resource as gold standard, which stems from a preliminary qualitative analysis of the annotated data and on future corpus development.

Keywords: hate speech, social media, immigrants, Italian

1. Introduction

The global spread of the so-called "web 2.0" and of social network sites allows users to find, create and share knowledge more easily than ever before, with scarcely any skill and cost required. This enormously increased the amount of user-generated content, within a process that some call "democratization" of the web (Silva et al., 2016). Yet, this freedom also allows for the publication of content which is abusive and harmful both towards the principles of democracy and the rights of some groups of people - namely *hate speech* (henceforth, HS). HS can be defined as any expression "*that is abusive, insulting, intimidating, harassing, and/or incites to violence, hatred, or discrimination. It is directed against people on the basis of their race, ethnic origin, religion, gender, age, physical condition, disability, sexual orientation, political conviction, and so forth*" (Erjavec and Kovačič, 2012).

Although definitions and approaches to HS are varied and depend on the juridical tradition of the country, many agree that what is identified as such can not fall under the protection granted by the right to freedom of expression, and must be prohibited. Online platforms like Twitter or Youtube discourage hateful content, but its removal mainly relies on users reports and lacks a systematic control. In this regard, a promising direction of research is the training of automated classifiers based on manually annotated corpora.

Our Contribution The work hereby presented deals with the creation of a Twitter corpus aimed at obtaining a large and richly-annotated dataset for the development of an automatic system of HS identification[1]. Moreover, given the multiplicity of factors that can contribute to the definition of the HS notion, the annotation scheme was specifically designed to account for this complexity and to offer a broader tagset capable of better representing the possible nuances of the message.

The annotation process will be described in the paper also comparing the contribution given by CrowdFlower users and that of expert annotators; this is in order to make a point on the complexity involved in the development of this kind of resource, where a balance among truth and subjectivity must be achieved.

The complete resource is going to be made freely available and accessible for non-commercial use by the end of 2018[2], along with the annotation guidelines.

The paper is organized as follows. Section 2. briefly overviews some cornerstone researches in the field of HS detection, while in Section 3. we present the criteria we used to collect our corpus and design the annotation scheme, and we describe the parallel annotations carried out on two different sub-sets. The annotation scheme used for both tasks is described in details in Section 4., and we discuss the annotation results in Section 5.. Finally, in Section 6. we present some conclusions on the present work and ideas for future developments.

2. Related Work

Hate speech is a complex notion, especially in a computational perspective. Attempts to define annotation labels that can account for such complexity are found in Ross et al. (2017), where data are labeled with regard to HS (*yes - no*) and to offensiveness (on a scale of 1 through 6), but also in Del Vigna et al. (2017), where the labels used are *no hate - weak hate - strong hate*, and in Kwong and Wang (2013), where tweets are classified for their offensiveness rated on a scale of 1 through 5. Despite the fact that offensiveness is often used interchangeably to refer to HS – which is not necessarily the case (Waseem, 2016) – all these works suggest that a simple binary label does not meet the required level of complexity for analyzing HS (although this may

[1]The work forms part of the wider Hate Speech Monitoring program coordinated by the Computer Science Department of the University of Turin. See `http://hatespeech.di.unito.it/`

[2]`https://github.com/msang/hate-speech-corpus`

come at the expense of reliability). We thus opted for such an approach, though separating the key notions of HS and offensiveness. Moreover, we also included multiple annotation categories, thus building a multi-faceted scheme, as described in section 4..

Contrary to many works where data are collected through a set of typically hateful words, Waseem and Hovy (2016) combine it with a set of neutral words, which are found to frequently occur with hateful content, without directly conveying hate. This allows the identification of a broader range of HS expressions that are not necessarily conveyed by offensive words. Although a much simpler scheme is used, compared to the one hereby described, a set of neutral keywords is a choice that proved consistent also with our findings.

While most of the available works are based on English language, there are a few that collect and analyze an Italian corpus. Del Vigna et al. (2017) identify six categories of HS (*Religion*, *Disability*, *Social status*, *Politics*, *Race*, *Sex and gender issues*, plus *Others*), and test a three-fold label for the annotation process. Musto et al. (2016) give an important contribution to the understanding of HS in relation to other social phenomena: by collecting geo-tagged data from Twitter, the authors create a Hate Map that locates the breeding grounds of five different types of hateful content (*Homophobic*, *Racist*, *Sexist*, *Anti-semitic* and *Against disability*). However, contrary to the approach we have followed in this work, the keywords used for filtering the data consisted of swear words frequently used against the five HS targets.

Eventually, stereotypes are as well among the crucial elements of prejudice and hatred against minority groups (Brown, 2011). Their relevance in analyzing HS, also highlighted in Warner and Hirschberg (2012), led us to introduce in our annotation scheme a novel orthogonal layer specifically devoted to mark the presence of stereotypes in the corpus. However, while in Warner and Hirschberg's contribution the use of stereotypes implicitly presupposes the presence of hateful content (although the words used to convey it may not be hateful themselves), in our study, stereotype alone is not sufficient to define hate speech.

To conclude, although inspired by the related work mentioned in this section, we followed the idea of developing a novel, finer-grained scheme where several facets of the phenomena involved can be represented. As further described in the remainder of the paper, this proved a very challenging direction.

3. Corpus Creation and Description

The corpus development forms part of the Hate Speech Monitoring program[3], coordinated by the Computer Science Department of the University of Turin (Italy) with the aim at detecting, analyzing and countering HS with an inter-disciplinary approach (Bosco et al., 2017).

Considering that among the minority groups targeted by HS, one is especially vulnerable and garners constant attention - often negative - from the public opinion, i.e. immigrants, we decided to work mainly on HS against immigrants. Nevertheless, considering that an operational definition of HS may be better extracted from data where a larger set of targets are considered and compared, we collected data where also other HS targets occur, namely Roma and Muslims.

For the data filtering, we opted for a common keyword-based approach, selecting a small set of neutral keywords associated with each target. We obtained a dataset of 236,193 tweets, from which we randomly selected a subset to be annotated. The detailed description of the entire pipeline of the data collection and annotation can be found in Poletto et al. (2017).

Given the higher degree of complexity that applying such scheme entailed, we first annotated 1,827 tweets, then we performed another data filtering starting from neutral words that more frequently occur in texts annotated as HS in this first dataset: *invadere* (invade), *invasione* (invasion), *basta* (enough), *fuori* (out), *comunist** (communist*), *african** (African), *barcon** (migrants boat*). After a further removal of duplicates and off-topic tweets, this resulted in a new portion of 4,182 tweets to be annotated. The final version of the corpus thus consists of 6,009 tweets, annotated according to the scheme and guidelines described in the next section, and by two different groups of annotators. The first section of the corpus, i.e.the tweets of the preliminary dataset and 1,327 tweets of the newly retrieved data, were annotated by a team of expert annotators. The annotation task was carried out by four independent annotators working in pairs, with one half of the corpus assigned to each pair. A fifth independent annotator was finally involved in order to solve the cases where at least one category was labeled differently by the previous two annotators. Furthermore, with the twofold aim of enlarging our annotated corpus and of comparing the accuracy of our team against that of a different group of judges, we had a new set of 2,855 tweet annotated on CrowdFlower. Here we carefully describe the settings we used for collecting this annotation.

CrowdFlower[4] is a crowdsourcing platform that allows researchers to have their data evaluated or annotated by contributors, who can be selected or discarded according to their accuracy. For our task, we uploaded a novel dataset to be annotated and provided a subset of 600 tweets from our gold standard corpus, used as test questions to monitor the contributors' reliability throughout their job. The annotation scheme we asked contributors to apply is exactly the same we used for our annotation, and is described in details in Section 4..

To compute contributors' accuracy, CrowdFlower simply checks if their answers to a given test question match the gold standard exactly. If not, the whole question is marked as failed. If a user fails too many test questions, his reliability gets below a threshold: he is then discarded and his judgments are marked as tainted. In our case though, the task was extremely complex and presented a huge array of possible combinations: for each tweet, contributors had to answer 5 or 6 (intensity being dependent on the presence of hate speech) multiple choice questions, with up to 4 an-

[3] http://hatespeech.di.unito.it/

[4] https://www.crowdflower.com/

swers each. Due to this reason, in order to avoid discarding too many contributors, we chose to assess users' reliability considering only their judgments on hate speech, and we required to keep a minimum reliability of 65% throughout the job.
Since the corpus language is Italian and we believe that only native speakers can fully grasp even the subtlest linguistic cues, at first we made our experiment available only to those users who claimed to be Italian speakers residing in Italy, assuming them to be native speakers[5]. Yet, due to the poor number of participants, we then opened the task also to Italian-speaking users residing abroad - who, with due exceptions, are likely to be second language speakers. Anyway, this measure only slightly increased the number of contributors.
Furthermore, contributors on CrowdFlower can give feedbacks on a test question when they miss it: this is sometimes helpful, as some test questions can be unclear or unfair and thus undermine accuracy of otherwise reliable judges. We removed a few after observing that contributors would repeatedly fail and contest them - this was not in order to artificially increase their accuracy score, but to make sure it was only tested against fair questions.
The annotators results on both sub-sets are reported and discussed in Section 5.. Next section briefly introduces our annotation scheme along with the main guiding principles for the annotation task.

4. Annotation Scheme: Tagset Design and Issues

HS identification is a challenging task that can be subject to individual biases (Waseem, 2016; Ross et al., 2017). In Weber (2009) these challenges are discussed by illustrating the European Court of Human Rights *modus operandi*, and in particular stressing the fact that there is no single distinctive factor in drawing the line between lawful and illicit, but a set of variables that the Court must consider case by case. Bearing this in mind, we attempted to annotate each tweet not only based on the presence or absence of HS, but also on other parameters that may even increase, or rather mitigate, the impact of the message.
As a result, we came up with a set of annotation categories and guidelines that attempt to encompass all those variables in a single coherent framework. Such categories include, besides HS, aggressiveness, offensiveness, irony and stereotype.
After the first annotation phase, we measured the Inter-Annotator Agreement (also described in Poletto et al. (2017)) and the results showed a high disagreement in all annotation categories (with a coefficient ranging from k=0.37 for offensiveness to k=0.54 for hate speech). In light of these results, we discussed the possible sources of disagreement, and revised the guidelines accordingly. Nevertheless, considering the complexity of this annotation task also for humans, we also discussed the inherent complexity of the task and the possibility of finding a single ground truth, given the topic addressed. What emerged from such discussion is described in Section 5.

As regards HS category alone, we decided to consider two aspects for its identification:

- the **target**, which must be a group identified as one of the three categories included in the search, or even an individual considered for its membership in that category (and not for its individual characteristics);
- the **action**, or more precisely the illocutionary force of the utterance (Searle, 1969): this means that we must deal with a message that spreads, incites, promotes or justifies hatred or violence towards the given target, or a message that aims at dehumanizing, delegitimizing, hurting or intimidating the target.

The joint presence of both elements in a tweet was considered essential to determine whether the tweet contained HS, as in the example below:

la prossima resistenza la dovremo fare subito contro gli invasori islamici!
(our next resistance movement should be right against Muslim invaders!)

In case even just one of these conditions was not detected, HS was assumed not to occur. Furthermore, a few more aspects are not considered HS in our study: offensiveness (either weak or strong) alone, blasphemy, historical denialism, overt incitement to terrorism, offense towards public servants and police officers, and defamation.
Below we provide a brief description of the remaining categories:

aggressiveness: it focuses on the user intention to be aggressive, harmful, or even to incite, in various forms, to violent acts against a given target; if present, it can be distinguished between *weak* and *strong*. For example, a message that implies or legitimates discriminating attitudes or policies is considered weakly aggressive:

Gli Italiani prima di tutto!
(Italians first!)

while the reference – whether explicit or just implied – to violent actions is considered strongly aggressive:

tutto tempo danaro e sacrificio umano sprecato
senza eliminazione fisica dei talebani e dei radicali musulmani è tutto inutile
(it's all a waste of time, money and human lives
without the extermination of Taliban and radical Muslims it's all useless)

offensiveness: conversely to aggressiveness, it rather focuses on the potentially hurtful effect of the tweet content on a given target; offensiveness also, if present, can be distinguished between *weak* and *strong*, based on the extent of the offense. If, for example, the given target is associated with typical human flaws, this is considered weakly offensive:

[5]This is, of course, an approximation. Unfortunately, to our knowledge, CrowdFlower does not allow stricter rules on users' language.

Italiani sfrattati e immigrati viziati
(Italians [are] evicted and immigrants [are] spoiled)

while if the target is addressed to by means of outrageous or degrading expressions, the tweet is annotated as strongly offensive:

Barletta, sgomberato mega-campo rom... #raccoltadifferenziata
(Barletta, big Roma camp evacuated ... #recycling)

irony: similar to Bosco et al. (2013), this has been used as a general term to cover other nuances such as sarcasm, humor, and satire. In the corpus, irony has a binary value (*no* or *yes*). The introduction of this category in the scheme was led by preliminary observations of the data, which highlighted how it was a fairly common linguistic expedient used to mitigate or indirectly convey a hateful content, as in the example below:

Toh, che caso: clandestino, islamico radicale e terrorista
(Uh, what a coincidence: clandestine, radical Muslim and terrorist)

stereotype: it determines whether the tweet contains any implicit or explicit reference to (mostly untrue) beliefs about a given target. Even in this case, the inclusion of this category in the scheme is motivated by some considerations on the fact that hatred against minority groups is often characterized by the presence of prejudices (as also mentioned in Section 2.). In the scheme, stereotype as well has a binary value (*yes* or *no*); here an example:

gli immigrati non muoiono di fatica . sono spesati di tutto.
(immigrants don't work themselves to death. they have everything paid for.)

The features and tags conceived for the last category, that of intensity, are discussed more in detail in the next section.

4.1. Going Deeper in the Annotation Task: the Incitement Degree

What emerged from a more detailed observation of the annotated data, especially regarding tweets that were considered as HS, is that these data consistently differed from one another, spanning over a broad range of intensity and harm. We thus proceeded to a further step of the research, developing an annotation framework which could account for different types of HS on the basis of what we namely defined as its "intensity": i.e. the degree to which incitement (to hate, and even violent acts) is present in the tweet.

In a pragmatical perspective, we noticed that some mitigation devices seemed to play a role in determining the intensity of hateful discourse. In our corpus, we observed that such forms of mitigation seem to interact in determining different degrees of HS. The framework describes five degrees of intensity modulated by mitigation strategies, with a 1-4 value scale for HS tweets, and 0 for the other ones:

- **degree 0**: there is no incitement at all. The message at issue, despite being annotated as aggressive, offensive or other, does not contain HS:

  Come sempre #Italia rifugio sicuro per terroristi!"
  (As usual #Italy [is] a safe haven for terrorists!)

- **degree 1**: there is no explicit incitement, but the acts ascribe a negative feature or quality to a targeted group. These cases are more similar to insults or judgements based on stereotypes; sometimes they suggest that the negative feature may pose a threat to the reader:

  Anche il PD se ne accorge: "I migranti sanno solo ostentare l'ozio. La gente è stufa."
  (Even the Democratic Party realized it: Migrants can only show off their laziness. People are fed up.)

- **degree 2**: there is no explicit incitement, but the acts aim at dehumanizing or delegitimizing the targeted group, or claim that the granting of its basic rights and needs is instead an unjust privilege, or that it damages the reader, and should therefore no longer be granted. These acts are not calls to violence, but they raise aversion or hate towards the targeted group:

  La polizia i controllori fermano solo italiani rom e immigrati non li avvicina nemmeno rischiano la vita.
  (Policemen [and] conductors only inspect Italians they don't even get close to Roma or immigrants they risk their lives.)

- **degree 3**: there is explicit incitement to violent or discriminatory actions, but the speaker refrains from assuming responsibilities for those actions and only justifies them or express his/her wish that they may happen:

  Quella schifosa rom prende anche in giro, speriamo che cn i loro fuochi tossici si brucino e crepino tutti alla svelta, TOLLERANZA 0.
  (That filthy Roma woman is even mocking, [I hope] they are all burned down by their toxic fires and croak quickly, NO TOLERANCE.)

- **degree 4**: there is explicit incitement to violent or discriminatory actions; the speaker overtly suggests or calls for these actions, and declares him/herself ready to carry them out, or take part in their realization:

  Hanno rotto il cazzo con tutti questi atti terroristici. Io sono pronto alla guerra.
  (They're pissing me off with all these terrorist attacks. I'm ready for war.)

To sum up, the complete annotation scheme is composed of the following categories and tags:

- **hate speech**: *no - yes*
- **aggressiveness**: *no - weak - strong*

- **offensiveness**:*no - weak - strong*
- **irony**: *no - yes*
- **stereotype**: *no - yes*
- **intensity**: *0 - 1 - 2 - 3 - 4*

4.2. Annotation Examples

Table 1 shows few examples of how such categories and their tags are applied in our corpus. As stated above, the only annotation constraint posed by our scheme is related to the annotation of intensity, which depends on that of HS: if the latter is not present, its intensity degree will be equal to 0, otherwise the degree will range from 1 through 4. Except for this case, all the other labels are mutually independent, in that the presence of a given category does not imply nor exclude any of the others. It is therefore possible, among other things, that a tweet contains HS, but not other phenomena represented by the other categories (see tweet number 1 in Table 1), that other phenomena are encountered along with HS (tweet number 2), or even that all the possible phenomena but HS are encountered (tweet number 3).

In the example tweet number 1, the message expresses a feeling of strong aversion towards migrants and their presence on the Italian soil, and implies a subtle encouragement to act in order to ban those who are in Italy or prevent others from coming. Hence the choice to annotate it as HS, with intensity equal to 2 (because of the implicit incitement to take action).

The tweet number 2 reports a news headline about a young Somali arrested in Italy for crimes committed in a refugee camp. This would be considered a tweet with a neutral content, if not for the comment that precedes the headline (*Risorse da accogliere...*, "resources to be welcomed...") and that completely reverses the annotators judgment. In fact, the comment reflects not only, once again, a strong aversion towards immigrants, but also an implicit incitement to see immigrants as a whole as criminals and a potential threat to the country and the safety of its citizens. This attitude is considered not only loaded with hate and stereotypes, but also as weakly aggressive and offensive. In addition, irony has also been detected in this tweet, in particular in the sarcastic use of the term *risorse* ("resources"), referred to immigrants, and in the use of the expression *da accogliere* ("to be welcomed"), which clearly intends exactly the opposite.

Finally, tweet number 3 is not considered an example of HS because it neither contains incitement to hate or violent actions, nor is targeted to any of the minority groups selected in our study. In fact, the tweet is presumably addressed to politicians, who reportedly tend to give a higher priority to migrants' needs compared to those of their compatriots. Such assumption is considered as strongly influenced by stereotypes, as well as weakly aggressive and offensive towards migrants (implicitly considered as people not worthy of help). The whole message is finally expressed in sarcastic tones.

The examples just described mainly serve the purpose of making clear to the reader some of the annotation choices adopted in the corpus creation; however, they also highlight a critical point of our study, which is related to the definition of precise and unambiguous linguistic criteria for the selection of proper labels.

Although there are recurring expressions that can be easily associated with HS, especially in reference to immigrants (e.g. *stop invasione!*, "stop invasion!"), the multiple ways in which it can be conveyed, as well as our choice to use only neutral keywords - rather than more explicit terms - to filter the corpus, somehow prevent the selection of precise lexical patterns in the identification of HS, as well as of the other categories. As a result, also recalling what stated at the beginning of Section 4., the selection of the tags to be associated with each tweet is determined case by case, based on its very content, on the context it refers to (whenever such information can be extrapolated from the text), and on the general principles indicated in the guidelines. Needless to say that this kind of approach has several drawbacks, being a strong disagreement one of those. On the other hand, in this work we rather look at the latter point as a *signal* (in Aroyo and Welty's (2015) words) of the inherent complexity of the task, given in particular by the potential ambiguity of the data at hand, as well as of the possibile solving strategies that can be put forward.

Section 5.1. is namely devoted to a wider discussion of such disagreement, its distribution in the two sub-corpora (i.e the one produced using CrowdFlower and the one annotated by field experts) and its possible causes.

5. Results and Discussion

In this section we extend the preliminary qualitative analysis of the data presented in a previous study on the tag distribution (Poletto et al., 2017). Figure 1 sums up such distribution over the final version of our corpus. However, bearing in mind that the main goal of our work is studying HS and the possible factors contributing to its automatic identification, we hereby provide an analysis of the annotated data centered on HS and, in particular, on its intensity, rather than on every single categories conceived in our scheme.

The categories that co-occur more frequently with HS are, expectedly enough, stereotype (72% of cases), aggressiveness (66%) and offensiveness (51%)[6]. Therefore, in the analysis of intensity degrees and their distribution, we thus focused on these aspects, so as to better understand whether an interdependence among all these categories actually exists and, ultimately, to come up with a "data-driven" definition of HS based on these findings. For this reason, we did not include in the frequency count the tweets annotated with a 0 degree, as they do not contain HS.

What emerged from the distribution of the intensity degrees (in Figure 1) preliminarily confirms what we discussed in the previous section, i.e. that HS and incitement are often mitigated and conveyed in subtler ways. In fact, most of the hateful tweets contain an implicit incitement (intensity degrees equal to 1 and 2), while a far smaller number of users explicitly incite to engage in violent or discriminatory actions: we thus observe a general trend by Twitter users to

[6]Irony is present in only 11% of hateful tweets.

tweet	hs	aggr.	off.	iro.	ster.	intens.
(1) *basta migranti in Italia, basta!* (no more migrants in Italy, I've had enough!)	yes	no	no	no	no	2
(2) *Risorse da accogliere... Omicidi e stupri nel campo profughi in Libia: arrestato 22enne somalo a Milano.* (Resources to be welcomed ... Murders and rapes in the refugee camp in Libya: 22-year-old Somali arrested in Milan.)	yes	weak	weak	yes	yes	1
(3) *hai la mia solidarietà ma se lasciamo fare a questi ti porteranno via anche l'auto per metterci qualche migrante* (you have my sympathy but if we leave it up to them they might as well get your car to put migrants in it)	no	weak	weak	yes	yes	0

Table 1: Annotation examples of three different tweets having immigrants as a target, one containing hate speech only (tweet number 1), along with its intensity, one containing HS as well as other categories (tweet number 2), and one where all categories are present except for HS and its intensity (tweet number 3).

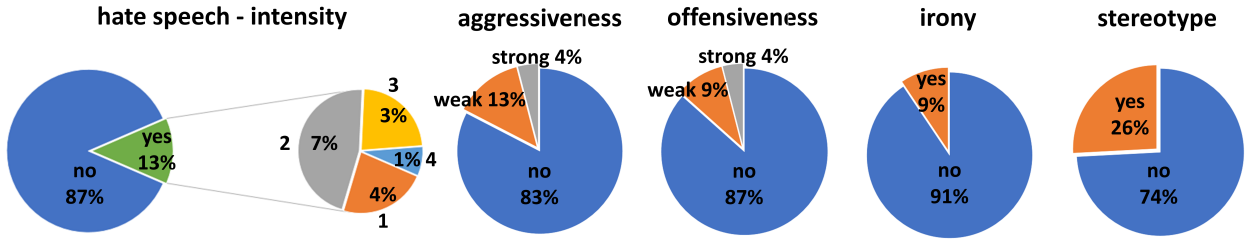


Figure 1: Distribution of all tags in the final version of the corpus.

limit the exposure and the risks arising from reprehensible, or even dangerous, claims.

We then investigated the possible interconnections between the intensity degree and the presence of stereotype, aggressiveness and offensiveness attributed in the previous phase. Results in Figure 2, 3 and 4 show the distribution of these tags across the 1 through 4 intensity degrees.

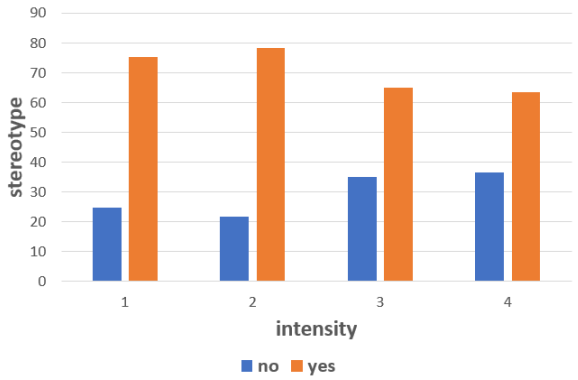


Figure 2: Distribution of stereotype tags (expressed in percentage) across the 1 through 4 intensity degrees.

The presence of stereotype is more frequent in all intensity degrees, though mostly in the lower ones (1 and 2). Such findings are quite consistent with our interpretation and definition of implicit incitement as typically based on, and promoting, prejudices, discrimination and hatred against a given target group (see Section 4.1.). On the other hand, stereotype largely co-occurs also with higher degrees,

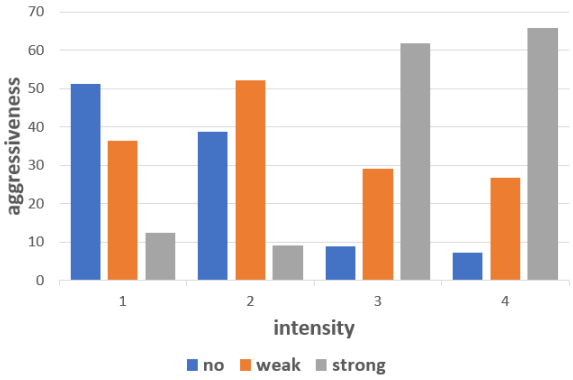


Figure 3: Distribution of aggressiveness tags (expressed in percentage) across the 1 through 4 intensity degrees.

which suggests us that it might be considered a fundamental factor in the definition of HS.

As regards aggressiveness, Figure 3 shows that almost all cases where aggressiveness is absent are concentrated in tweets annotated with a lower intensity degree (1 and 2); to a partially similar extent, most of the tweets considered as weakly aggressive constitute an example of implicit incitement (therefore with an intensity degree equal to 1 or 2). Conversely, the tweets expressing explicit incitement, in its different degrees (namely 3 and 4), are for most part strongly aggressive. Aggressiveness as well can thus be taken into account while providing a definition of HS and incitement.

This does not seem to be the case with offensiveness, whose

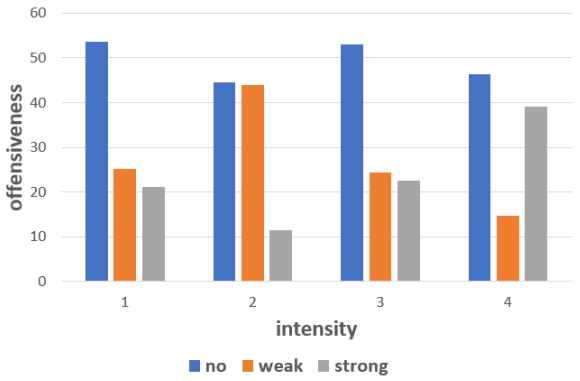


Figure 4: Distribution of offensiveness tags (expressed in percentage) across the 1 through 4 intensity degrees.

distribution shows a less coherent pattern, with all its possible tags spanned over all intensity degrees. The strongly offensive tweets, as expected, are mostly concentrated in the highest degree cluster, and weakly offensive tweets are more frequent in the 2-degree instances; on the other hand, the majority of tweets in all the intensity degrees were not considered as offensive at all. This partially confirms the idea that offensive language does not necessarily involve forms of hate or violence (Waseem, 2016) and supports our choice to select only neutral keywords while filtering the data for the corpus (see Section 3.).
All such patterns will become useful when we will exploit the corpus for automatic HS detection, in a machine learning perspective.

5.1. Agreement Discussion

Given the complexity of the scheme described in the previous sections and the subjectivity involved in the topic, we expected the annotation to pose a number of challenges and problems. That is why, after a first stage where only expert annotators were involved, we also carried out the annotation experiment using CrowdFlower.
As the following analysis will show, a large number of cases appeared to be particularly tricky, also resulting in a very poor annotation agreement (see Table 2).
Besides, we observed peculiar patterns in the behavior of CrowdFlower annotators. First of all, conversely to our expectations, the number of contributors remained low (five annotators carried out more than 90% of the job). Secondly, as said before, the guidelines we used for our annotation are the same we provided CrowdFlower users with; nonetheless, some of their replies and feedbacks seem to suggest that they have not read or taken into due account our definitions and examples. Thus, although their accuracy score remained above the threshold, we should keep in mind that their judgments can not be compared to those by experts annotators, and that their fluctuation is probably due not only to shortcomings in the annotation scheme but also to a certain negligence among the judges.
Considered that the annotation process was carried out in different stages and with different methods, as described in Section 3., agreement as well was computed separately and with different coefficients, based on the number of judments available for each tweet. In the first sub-corpus,

	hs	**aggr.**	**off.**	**iro.**	**ster.**	**intens.**
experts	0.45	0.48	0.45	0.32	0.41	0.21
CF	0.38	0.25	0.30	0.12	0.20	0.31

Table 2: Agreement for each annotation category in both sub-sets, i.e. the one annotated by our expert team (first row in the table) and the one by CrowdFlower contributors (second row).

two expert annotators worked on the same set of tweets, while for the CrowdFlower experiment each tweet was expected to have at least three judgments; therefore the inter-annotator agreement was assessed by using the Cohen's κ (Carletta, 1996) for the former and the Krippendorff's α (Krippendorff, 2007) for the latter.
The results for the two groups, however low in both cases, show the greater reliability of the corpus annotated by the experts, hence its (relatively) better quality overall. On the other hand, the strongest disagreement between the expert annotators is found for intensity, which, conversely, is the second category after HS where CrowdFlower users seem to reach a higher number of consistent annotations.
This highlights that intensity is the most controversial point of our scheme. While we believe it is crucial to acknowledge that not all hate speech is the same and that there are indeed different shades of intensity, our results show that much work is still to be done before these shades can be effectively defined and detected. The low agreement suggests the presence of shortcomings in the guidelines, which are still ambiguous and not always helpful in settling doubtful cases. Furthermore, distinctions between the four levels are often based on pragmatic rather than semantic features: this results in annotators giving more weight to the attitude of the author than to the actual content of its tweet.
Thus, according to our guidelines, tweet (4) below is to be considered more intense - and therefore more dangerous - than tweet (5), only because the former's author uses a first-person construction which entails individual responsibilities, while the latter's uses a more detached and impersonal form.

(4) *Milva e la "sua" Goro: "Se vivessi ancora lì, i migranti li avrei ospitati io" ///dacci l'indirizzo...te li porto io...almeno una dozzina*
(Milva and "her" Goro: "If I still lived there, I'd have hosted those migrants myself" ///give us your address...I'll bring you some...a dozen at least)[7]

(5) *Sarebbe da VIETARE il culto dell'islam, bisognerebbe DISTRUGGERE le moschee, DEPORTARE tutti gli islamici e dichiarare l'islam FUORI LEGGE!*
(Islamic faith should be BANNED, mosques should be DESTROYED, all Muslims should be DEPORTED and Islam should

[7]In October 2016, some residents in the little town of Goro and Gorino, Italy, erected barricades to prevent 12 asylum-seekers from entering the town and being hosted in a tourism facility, as determined by legal authorities. Milva is a popular Italian singer, born in Goro, who spoke out in favor of the asylum-seekers.

be OUTLAWED!)

Cases such as this suggest that the present scheme is not always suitable for understanding intensity and dangerousness of HS. Therefore, future work will necessarily have to focus on a thorough rethink of how intensity is conceived and annotated. The scheme will have to be simpler, featuring maybe only two levels - for example "weak" and "strong", as proposed in (Del Vigna et al., 2017) for hate speech and in this paper for aggressiveness and offensiveness; and it will have to be clearer with regard to distinctive features.

6. Conclusion

In this paper we describe an Italian Twitter corpus of HS against immigrants and propose a novel multi-layered annotation scheme to account for different aspects of this multifaceted and complex phenomenon. Besides the presence of HS, we annotated its intensity, as well as the presence of aggressiveness, offensiveness, irony and stereotypes. A preliminary analysis of annotation results is proposed, that opens new perspectives for the exploitation of our data set for the development of HS detection systems.
The choice of such a rich and fine-grained scheme is not flawless, nor without drawbacks, all highlighted and discussed in this paper. On the other hand, namely due to its greater complexity, the corpus lends itself to more detailed and systematic analyses of the possible linguistic patterns associated not only with HS itself, but also to all the other categories included in our scheme.

Acknowledgments

C. Bosco, F. Poletto, M. Sanguinetti and V. Patti were partially funded by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media*, S1618_L2_BOSC_01) and by Fondazione CRT (*Hate Speech and Social Media*, 2016.0688), which instead totally funded the work of F. Poletto.

7. References

Aroyo, L. and Welty, C. (2015). Truth is a lie: Crowd truth and the seven myths of human annotation. *Artificial Intelligence Magazine*, pages 15–24.

Bosco, C., Patti, V., and Bolioli, A. (2013). Developing corpora for sentiment analysis: The case of irony and SENTI-TUT. *IEEE Intelligent Systems*, 28(2):55–63.

Bosco, C., Viviana, P., Bogetti, M., Conoscenti, M., Ruffo, G., Schifanella, R., and Stranisci, M. (2017). Tools and resources for detecting hate and prejudice against immigrants in social media. In *Proceedings of First Symposium on Social Interactions in Complex Intelligent Systems (SICIS), AISB Convention 2017, AI and Society*, Bath, UK.

Brown, R. (2011). *Prejudice: Its Social Psychology*. Wiley.

Carletta, J. (1996). Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254, June.

Del Vigna, F., Cimino, A., Dell'Orletta, F., Petrocchi, M., and Tesconi, M. (2017). Hate me, hate me not: Hate speech detection on Facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), Venice, Italy, January 17-20, 2017.*, pages 86–95.

Erjavec, K. and Kovačič, M. P. (2012). "You don't understand, this is a new war!" Analysis of hate speech in news web sites' comments. *Mass Communication and Society*, 15(6):899–920.

Krippendorff, K. (2007). Computing Krippendorff's alpha reliability. Departmental papers (ASC) 43.

Kwok, I. and Wang, Y. (2013). Locate the hate: Detecting tweets against blacks. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, pages 1621–1622.

Musto, C., Semeraro, G., de Gemmis, M., and Lops, P. (2016). Modeling community behavior through semantic analysis of social data: The italian hate map experience. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization, UMAP 2016, Halifax, NS, Canada, July 13 - 17, 2016*, pages 307–308.

Poletto, F., Stranisci, M., Sanguinetti, M., Patti, V., and Bosco, C. (2017). Hate speech annotation: Analysis of an italian Twitter corpus. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, volume 2006 of *CEUR Workshop Proceedings*, Rome, Italy. CEUR-WS.org.

Ross, B., Rist, M., Carbonell, G., Cabrera, B., Kurowsky, N., and Wojatzki, M. (2017). Measuring the reliability of hate speech annotations: The case of the European refugee crisis. *CoRR*, abs/1701.08118.

Searle, J. R. (1969). *Speech acts: an essay in the philosophy of language*, volume 626. Cambridge University Press.

Silva, L., Mondal, M., Correa, D., Benevenuto, F., and Weber, I. (2016). Analyzing the targets of hate in online social media. In *Proceedings of the 10th International Conference on Web and Social Media, ICWSM 2016*, pages 687–690. AAAI Press.

Warner, W. and Hirschberg, J. (2012). Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, LSM '12, pages 19–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June. Association for Computational Linguistics.

Waseem, Z. (2016). Are you a racist or am i seeing things? annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas, November. Association for Computational Linguistics.

Weber, A. (2009). *Manual on hate speech*. Council of Europe.

A Large Multilingual and Multi-domain Dataset for Recommender Systems

Giorgia Di Tommaso, Stefano Faralli*, Paola Velardi
Department of Computer Science, *Unitelma
Sapienza, University of Rome, Italy
{ditommaso,velardi}@di.uniroma1.it, stefano.faralli@unitelmasapienza.it

Abstract

This paper presents a multi-domain interests dataset to train and test Recommender Systems, and the methodology to create the dataset from Twitter messages in English and Italian. The English dataset includes an average of 90 preferences per user on music, books, movies, celebrities, sport, politics and much more, for about half million users. Preferences are either extracted from messages of users who use Spotify, Goodreads and other similar content sharing platforms, or induced from their "topical" friends, i.e., followees representing an interest rather than a social relation between peers. In addition, preferred items are matched with Wikipedia articles describing them. This unique feature of our dataset provides a mean to derive a semantic categorization of the preferred items, exploiting available semantic resources linked to Wikipedia such as the Wikipedia Category Graph, DBpedia, BabelNet and others.

Keywords: social mining, recommender systems, Twitter, users' interest dataset

1. Introduction

Recommender systems are widely integrated in online services to provide suggestions and personalize the on-line store for each customer. Recommenders identify preferred items for individual users based on their past behaviors or on other similar users. Popular examples are Amazon (Linden et al., 2003) and Youtube (Davidson et al., 2010). Other sites that incorporate recommendation engines include Facebook, Netflix, Goodreads, Pandora and many others.

Despite the vast amount of proposed algorithms, the evaluation of recommender systems is very difficult (Fouss and Saerens, 2008). In particular, if the system is not operational and no real users are available, the quality of recommendations must be evaluated on existing datasets, whose number is limited and what is more, they are focused on specific domains (i.e, music, movies, etc.). Since different algorithms may be better or worse depending on the specific purpose of the recommender, the availability of multi-domain datasets could be greatly beneficial. Unfortunately, real-life cross-domain datasets are quite scarce, mostly gathered by "big players" such as Amazon and eBay, and they not available to the research community[1].

In this paper we present a methodology for extracting from Twitter a large dataset of user preferences in multiple domains and in two languages, Italian and English. To reliably extract preferences from users' messages, we exploit popular services such as Spotify, Goodreads and others. Furthermore, we infer many other preferences from users' friendship lists, identifying those followees representing an interest rather than a peer friendship relation. In this way we learn, for any user, several interests concerning books, movies, music, actors, politics, sport, etc. The other unique feature of our dataset, in addition to multiple languages and domains, is that preferred items are matched with corresponding Wikipedia pages, thus providing the possibility to generalize users' interests exploiting available semantic resources linked to Wikipedia, such as the Wikipedia Category Graph, Babelnet, DBpedia, and others.

The paper is organized as follows: Section 2. summarizes previous research on creating datasets for recommender systems, Section 3. describes our methodology to collect data, and Section 4. presents and evaluates our results. Finally, in Section 5. we draw conclusions providing some directions for future research.

2. Related work

Most recommender systems (hereafter RS) are based on one of three basic approaches (Felfernig et al., 2014): collaborative filtering (Schafer et al., 2007) generates recommendations collecting preferences of many users, content-based filtering (Pazzani and Billsus, 2007) suggests items similar to those already chosen by the users, and knowledge-based recommendation (Trewin, 2000) identifies a semantic correlation between user's preferences and existing items. Hybrid approaches are also widely adopted (Burke, 2002). All approaches share the need of sufficiently large datasets to learn preferences and to evaluate the system, a problem that is one of the main obstacles to a wider diffusion of RS (Gunawardana and Shani, 2009) since only a small number of researchers can access real users data, due to privacy issues.

To overcome the lack of datasets, challenges as RecSys have been lunched[2], and dedicated web sites have been created (e.g., SNAP[3] or Kaggle[4]), where researchers can upload their datasets and make them available to the community. However it is still difficult to find appropriate data for novel types of recommenders, as the majority is focused on a single topic, like music (Dror et al., 2012), (Shepitsen et al., 2008),) food ((Kamishima and Akaho, 2010), (Sawant and Pai, 2013)), travel ((Wang et al., 2010), (Mavalankar et al., 2017)) and more (Çano and Morisio, 2015). Fur-

[1] https://recsys.acm.org/wp-content/uploads/2014/10/recsys2014-tutorial-cross_domain.pdf

[2] https://recsys.acm.org/

[3] http://snap.stanford.edu/data

[4] https://www.kaggle.com/datasets/?sortBy=hottest&group=all

thermore, while a small number of large datasets are available, such as Movielens (Harper and Konstan, 2016), Million song dataset (McFee et al., 2012) and Netflix Prize Dataset (Bennett et al., 2007), many others are quite small and based on very focused experiments.

Concerning the source of data for extracting preferences, social networks are often used, since their content is available with more or less severe restrictions. For example, (Chaabane et al., 2012) use Facebook, perhaps the most appropriate platform for this type of study, as it provides incentive mechanisms for sharing interests and content. However, a disadvantage is the difficulty in extensively accessing profiles due to user privacy issues. In (Chaabane et al., 2012), 104,000 public and 2000 private profiles (obtained by volunteers) have been extracted. Another study (Yan et al., 2014) presents an overview of users' interests derived from multiple platforms to which the same user is registered (e.g., Twitter, Youtube, etc.). To find the same account on multiple social networks, Google+ is used, where users are encouraged to share and link the addresses of their accounts. Overall 143,259 accounts were collected, of which, 11,850 provided multiple accounts. Many other studies use Twitter (eg. (Gesualdo et al., 2013), (Adamopoulos and Tuzhilin, 2014)) as a platform for extracting users' information, although existing restrictions limit the amount of freely accessible traffic to 1%.

Data extraction from Twitter messages is expensive since it requires natural language processing techniques to analyze the text. To overcome this difficulty, a number of studies exploited platforms (e.g., Youtube, Spotify) that integrate among their services the ability to post the user's personal content on the most popular social network sites, such as movies that users are watching. Sharing this information is done in a simple and predefined way. Depending on the social network chosen, the content, for example a Youtube video, will be shared with a pre-formatted message formed by the video name, a link, a self-generated text and, if provided, a numerical rating (eg. "How It's Made: Bread" `https://youtu.be/3UjUWfwWAC4` via YouTube). The message can also be enriched and personalized by the user. In (Pichl et al., 2015) this types of messages are extracted from Twitter, to detect music interests. The dataset is based on 100,000,000 tweets with the #nowplaying main tag. Tweets are extracted via Twitter APIs over 3-years and next, MusicBrainz and Spotify are used to add more details. Other studies extract data about music (Schinas et al., 2013) or sport (Nichols et al., 2012) events. However, all the datasets generated in this way concern only one domain of interest.

To the best of our knowledge, the only really multi-domain dataset[5] is presented in (Dooms et al., 2014), where pre-structured tweets about three domains - movies, books and video-clips - are extracted respectively from IMDb (Internet Movies Database), Youtube and Goodreads. With respect to this work, we collect a much wider number of interests, since in addition to pre-formatted messages based on a number of available services, we reliably extract many additional types of interests exploiting users' followees lists. Furthermore, as shown in Section 4., we collected many interest *types* for each user, while the dataset released in (Dooms et al., 2014) includes only 7 users with at least 3 types of interests.

[5]Another is the ConcertTweets `https://github.com/padamop/ConcertTweets`, however it is focused on music events.

3. Workflow and data sources

This section summarizes the data sources and workflow to create our multi-domain dataset. We extract preferences from a user's messages and from his/her friendship list, identifying those followees which represent an interest rather than a peer friendship relationship. The process is in three steps:

1. *Extracting interests from users' textual communications*. The first step is to extract preferences from Twitter messages. Using textual features extracted from users' communications, profiles or lists seems a natural way for modeling their interests. However, this information source has several drawbacks when applied to large data streams, such as the set of Twitter users. First, it is computationally very demanding to process millions of daily tweets in real time; secondly, the extraction process is error prone, given the highly ungrammatical nature of micro-blogs. To reliably extract preferences from users' messages, in line with other works surveyed in Section 2., we use a number of available services, described hereafter, that allow to share activities and preferences in different domains - movies, books etc. - using pre-formatted expressions (e.g, for Spotify: #NowPlaying) followed by the url of a web site, from which we can extract information without errors. The drawback is that a relatively small number of users access these services and in addition, preferences are extracted only in few domains.

2. *Extracting interests from users' friendship lists*. In (Barbieri et al., 2014) the authors argue that users' interests can also be implicitly represented by the authoritative (*topical*) friends they are linked to. This information is available in users' profiles and does not require additional textual processing. Furthermore, interests inferred from topical fiends are less volatile since, as shown in (Myers and Leskovec, 2014), "common" users tend to be rather stable in their relationships. Topical friends are therefore both relatively stable and readily accessible indicators of a user's interest. Another advantage is that average Twitter users have hundreds of followees, many of which, rather than genuine friends, are indicators of a variety of interests in different domains, such as entertainment, sport, art and culture, politics, etc.

3. *Mapping interests onto Wikipedia pages*. The final step is to associate each interest, either extracted from messages or inferred from friendship relations, with a corresponding Wikipedia page, e.g., @nytimes $\Rightarrow$ WIKI:EN:The_New_York_Times (in this example, @nytimes is a Twitter account extracted from a user's friendship list). Although not all interests can be mapped on Wikipedia, our experiments show that this

is possible in a large number of cases, since Wikipedia articles are created almost in real-time in correspondence with virtually any popular entity, either book, or song, actor, event, etc.

We applied this workflow to two Twitter streams in two languages, English and Italian, as detailed in what follows.

3.1. Extracting preferences from messages

Everyday a huge number of people uses on-line platforms (eg. Yelp, Foursquare, Spotify, etc.) that allow to share activities and preferences on different domains on a social network in a standard way. Among the most popular services accessed by Twitter users, we selected those providing pre-formatted messages, as detailed hereafter.

- **Spotify:** Spotify is a music service offering on-demand streaming of music, both desktop and mobile. Users can also create playlists, share and edit them in collaboration with other users. In addition to accessing the Spotify web site, users can retrieve additional information such as the record label, song releases, date of release etc.. Since 2014, Spotify is widely used in America, Europe and Australia. Spotify is among the services allowing to generate self-generated content shares in Twitter. An example of these tweets is: "#NowPlaying The Sound Of Silence by Disturbed `https://t.co/d8Sib5EDVf`". The standard form of these tweets is:

  `#NowPlaying <title> by <artist > <URL>`

  By filtering the tweets stream and using Twitter APIs for hashtag detection, we generated a stream of all the users who listened music using Spotify.

- **Goodreads and aNobii:** Similarly to Spotify, a number of platforms allows to share opinions and reviews on books. In these platforms, users can share both titles and ratings. Similarly to Spotify, generated tweets have a predefined structure and point to an URL. In the book domain, we use Goodreads (10 million users and 300 million books in the database) and for Italian, the more popular aNobii service.

- **IMDb and TVShowTime:** In the domain of movies, currently there are no dominant services. Popular platforms in this area are Flixter, themoviedb.org and iCheckMovies. However, many of these platforms use the IMDb database, owned by Amazon, which handles information about movies, actors, directors, TV shows, and video games. We also use the TvShowTime service for Italian tweets.

First, we collect in a Twitter stream all messages including a hashtag related to one of the above mentioned services (#NowPlaying, #IMDb ..). Next, we extract from tweets the music, movie and book preferences for a set of users U who accessed these services. Unlike (Dooms et al., 2014), we avoid parsing tweets using specific regular expressions, since users are free to insert additional text in the pre-formatted message. Rather, we exploit an element that all these pre-formatted tweets have: the URL, as in (Pichl et al., 2014). Every URL points to the website containing all the information, such as, e.g., title, author and publisher for books. Since the URL in the tweets is a short URL, we first extend the original URL so that all URLs belonging to a given platform can be identified (for example, all Goodreads URLs contain the "goodreads.com" string). Next, we access the web site and scrape its content. The reason for extracting the information from the URL (which is computationally more demanding) rather than from the tweet itself is twofold:

1. Tweets can be ambiguous or malformed, and furthermore, users can insert additional text in the pre-formatted message, e.g, "*#NowPlaying Marty*. This guy is amazing.♫`http://t.co/jwxvLiNenW`". Scraping the html page at the URL address ensures that we extract data *without errors*, even for complex items such as book and movie titles;

2. The URL includes additional information (e.g., not only the title of a song, but also the singer and the record label), which provide us a context to reliably match the extracted entity (song, book, movie) with a Wikipedia article, as detailed in Section 3.3..

3.2. Extracting preferences from users' "topical" friends

We denote as *topical friends* those Twitter accounts in a user's followees list representing popular entities (celebrities, products, locations, events . . .). For example, if a user follows @David_Lynch, this means that he/she likes his movies, rather than being a genuine friend of the director.
There are several clues to identify topical friends in a friendship list: first, topical relationships are mostly *not reciprocated*, second, popular users have a high in-degree. However, these two clues alone do not allow to distinguish e.g., bloggers or very social users from truly popular entities.
To lean a model of topical friends we first collected a network of Verified Twitter Accounts. Verified accounts [6] are authentic accounts of public interest. We started from a set of seed verified contemporary accounts in 2016, and we then crawled the network following only verified friends, until no more verified accounts could be found. This left us with a network of 107,018 accounts of verified contemporary users (V), representing a "model" of authoritative users' profiles. Next, from the set U of users in our datasets (separately for the English and Italian streams), we collected the set F of Twitter accounts such that, for any $f \in F$ there is at least one $u \in U$ such that u follows f.
In order to identify candidate topical friends $F_t \subset F$, we learned a model of popularity, using the set V and a random balanced set of $\neg V$ users. For each account, we extracted three structural features (in degree, out degree and their ratio) and one binary textual feature (presence in the user's account profile of role words such as *singer, artist, musicians, writer*..). Then, we used 80% of these accounts to train a SVM classifier with Laplacian kernel and the remaining 20% for testing with cross-validation, obtaining a

[6] `https://developer.twitter.com/en/docs/api-reference-index`

total accuracy of 0.88 (true positive rate 0.95 and true negative rate 0.82). Finally, the classifier was used to select a subset $F_t \subseteq F$ of *authoritative* users representing "candidate" topical friends. The last filtering step to identify "true" topical friends in F_t, i.e., genuine users' *interests*, consists in determining which members of the set F_t have a matching Wikipedia page. The intuition is that, if one such match exists, the entity to which the Twitter account belongs is indeed "topical".

3.3. Mapping to Wikipages

Mapping interests extracted from users' messages to Wikipedia pages is a very reliable process, given the additional contextual information extracted from the URL (see Section 3.1.). For a complete example, see Section 3.4..
On the contrary, matching interests extracted from a user's friendship list with corresponding Wikipedia pages is far more complex, because of synonymy, polysemy and ambiguity, as argued in (Faralli et al., 2017). Furthermore, the information included in a user's Twitter profile is very sketchy and in some case misleading, therefore it may not provide sufficient context to detect a similarity with the correspondent Wikipedia article. For example, Bill Gate's description field[7] in his Twitter profile is: "*Sharing things I'm learning through my foundation work and other interests...*" which has little in common with his Wikipedia page: "*William Henry Gates III (born October 28, 1955) is an American business magnate, investor, author, philanthropist, humanitarian and co-founder of the Microsoft Corporation along with Paul Allen.*"
To find the Wikipedia page, if any, associated to a topical friend we used the methodology that was first presented in (Faralli et al., 2015) and improved in (Faralli et al., 2017), summarized in what follows:

1. *Selection of candidate senses*: For any f in F_t, find a (possibly empty) list of *candidate wikipages*, using BabelNet synonym sets (in BabelNet, each "BabelSynset" points to a unique Wikipedia entry (Navigli and Ponzetto, 2012));

2. *BoW Disambiguation*: Compute the bag-of-words (BoW) similarity between the user description in f's Twitter account and each candidate wikipage. The BoW representation for each wikipage is obtained from its associated BabelNet relations (Delli Bovi et al., 2015);

3. *Structural Similarity*: If no wikipages can be found with a sufficient level of similarity (as for the previous example of Bill Gates), select from f's friendship list those friends already mapped to a wikipage (if any), and compute the similarity between those wikipages and candidate wikipages.

3.4. Anecdotic examples and evaluation

We provide hereafter examples of the process outlined in previous Sections. For the sake of space, we consider only examples of interests extracted from users' messages.

[7] as retrieved on January 2018

1. **detection of "interesting" tweets**
   We collect all tweets containing the selected hashtags and discard those which do not include an url.
   accepted #NowPlaying High by James Blunt https://t.co/7EiepE2Bvz
   discarded #NowPlaying CaSh Out - Cashin' Out

2. **extraction of the url**
   Next, we retrieve the original url from short url. If the url does not contain the platform domain (eg. spoti.fi), we discard it.
   accepted: `https://t.co/oShYDc6DeL` → `http://spoti.fi/2cTPn0U`
   discarded: `https://tunein.com/radio/Pratt-Radio-s50434/`

   Then, we extract information about an item (movie, book or music) from the platform site through APIs, (when available) or web-scraping. For each platform we obtain the following data:

   (a) **Music**: <Title, Author (eg. singer, band)>

   (b) **Books**: <Title, Author>

   (c) **Movie**: <Title, Year of production, Type (eg. movie, tv series)>

3. **mapping to Wikipedia**
   Wikipedia mapping is obtained by a cascade of weighted boolean query on a Lucene Index.The index is based on a tdf-idf with vector space model.

   (a) Searching the Wikipage of an item
   $< TITLE \in WikiTitle >^{w_1}$
   $\wedge < AUTHOR \in WikiGloss >^{w_2}$
   $\wedge\big(\big(< WORDS \in WikiTitle >^{w_3}$
   $\vee < AUTHOR \in WikiTitle >^{w_4}$
   $\vee < WORDS \in WikiText >^{w_5} \big)$
   $\vee\neg\big(< WORDS \in WikiTitle >^{w_3}$
   $\vee < AUTHOR \in WikiTitle >^{w_4}$
   $\vee < WORDS \in WikiText >^{w_5} \big)\big)$
   Where
   w_i is a weigth assigned to a query
   $< WORDS >$ for music = {"song"}
   $< WORDS >$ for books = {"books", "novel", "saga"
   $< WORDS >$ for movie = {"film", "series", "TV series", "episode"}
   When the page doesn't exist or is not available we search the page of the item's author.

   (b) Searching the wikipage of the item's author
   $< AUTHOR \in WikiTitle >^{w_1}$
   $\wedge\big(\big(< WORDS \in WikiTitle >^{w_2}$
   $\vee < TITLE \in WikiText >^{w_3}$
   $\vee < WORDS \in WikiText >^{w_4} \big)$
   $\vee\neg\big(< WORDS \in WikiTitle >^{w_2}$
   $\vee < TITLE \in WikiText >^{w_3}$
   $\vee < WORDS \in WikiText >^{w_4} \big)\big)$
   Where

USER ID:787930***		
Source	**Interest**	**Wikipage**
IMDb	Eyes Wide Open - 2009 - movie	WIKI:EN:Eyes_Wide_Open_(2009_film)
	Okja - 2017 - movie	WIKI:EN:Okja
Goodreads	The Beautifull Cassandra - Jane Austen	WIKI:EN:Jane_Austen
	The Beach - Alex Garland	WIKI:EN:The_Beach_(novel)
Spotify	I Don't Know What I Can Save You From - Kings of Convenience	WIKI:EN:Kings_of_Convenience!
	Nothing Matters When We're Dancing - **The Magnetic Fields**	WIKI:EN:The_Magnetic _Fields
Topical friends	**@IMDb**	WIKI:EN:IMDb
	@UNICEF_uk	WIKI:EN:UNICEF_UK
	@TheMagFields	WIKI:EN:The_Magnetic _Fields
	@BarackObama	WIKI:EN:Barack_Obama
	@Spotify	WIKI:EN:Spotify

Table 1: Excerpt of a Twitter user's interests

w_i is a weigth assigned to a query
$< WORDS >$ for music = {"singer", "band", "artist", "songwriter", "composer", "musician", "record producer"}
$< WORDS >$ for books = {"writer", "novelist", "cartoonist", "journalist", "orator", "poet", "Japanese manga author"}

Examples of positive results when searching an item's title:
Tweet: I rated Arrow: Disbanded (S5.E18) 9/10 #IMDb `https://t.co/Oo4qu6tHl7`
Original url: `http://www.imdb.com/title/tt5607516/`
WikiPage: WIKI:EN:Arrow_(TV_series)
Note that WikiTitle contains the term "TV_series" in WORDS.

Examples with positive results when searching an item's author:
Tweet: 4 of 5 stars to Silken Prey by John Sandford `https://t.co/AyF5Iuyc9s`
Original url: `https://www.goodreads.com/review/show/2105147499`
WikiPage: WIKI:EN:John_Sandford_(novelist)
Note that WikiTitle contains the term "novelist" in WORDS.

Examples with incorrect results:
The system returns no results or incorrect results in 3 cases:

1. the title page of the item or item's author page doesn't exist:

   Tweet: #NowPlaying Cherry Garcia by Dingus `https://t.co/t7g4EQ3ucp`
   Original url: `https://open.spotify.com/track/2vfZpWZGUtfM2VYVomh7MZ`
   WikiPage: WIKI:EN:Eric_Dingus
   The wikipage of the correct singer (which is not Eric Dignus) doesn't exist.

2. the extracted data is wrong because information extraction fails for various reasons (eg. missing or poorly structured information in the platform):

   Tweet: 4 of 5 stars to Love, Rosie by Cecelia Ahern `https://t.co/EtS1RoCarK`
   Original url: `https://www.goodreads.com/review/show/1965312746`
   WikiPage: WIKI:EN:Love,_Rosie_(film)
   In this case Wikipedia shows the title page with the initial name of the book "WIKI:EN:Where_Rainbows_End", but the original name of the book was modified in the reprint.

3. the searched Wikipedia page does not contain enough context to match the query.

Overall, the methodology to extract and map preferences from messages proved to be very reliable. We evaluated the precision (with adjudication) on a randomly selected balanced sample of 1200 songs, books, and movies in English, obtaining a precision of 96%. For the Italian dataset, we evaluated 750 songs, books, and movies, obtaining a precision of 98%.

As far as the topical interests mapping performance is concerned, in (Faralli et al., 2017) the authors mention that inducing interests from topical friends and subsequent mapping to Wikipedia has an accuracy of 84%. Since our aim in this work is to generate a highly accurate dataset, we considered only the subset F'_t in F_t with indegree (with respect to our population U) higher than 40. In fact, we noted that less popular topical friends may still include bloggers or Twitter users for which, despite some popularity, a Wikipage does not exists. In these cases, our methodology may suggest false positives. When applying the indegree filter, the precision -manually evaluated with adjudication on 1250 accounts randomly chosen in this restricted population F'_t- is as high as 90%.
Finally, we note that we are not concerned here with measuring the recall, since our aim is to release a dataset with high precision and high coverage, in terms of number of interests per user, over the considered populations. To this end, the indegree threshold 40 was selected upon repeated experiments to obtain the best trade-off between the distribution of interests in the population U and precision of the mapping, as shown in Section 4..

message-based interests ($\|U\|$=444,744 English speaking users)	**Music**	**Books**	**Movie**	**Total**
platform	*Spotify*	*Goodreads*	*IMDb*	*All*
#crawled tweets (tweets with selected hashtags)	19,941,046	693,975	97,772	20,732,793
#cleaned tweets (tweets for which an URL was extracted)	2,519,166	139,882	88,355	2,747,403
# of unique interests with a mapping to a Wikipage	253,311	20,710	8,282	282,303
average #interests per user	6	8	6	6
average #users per interest	7	3	7	6
precision of Wikipedia mapping (on 3 samples of 400 items each)	94%	96%	97%	**96%**

Table 2: 6-months (April-September 2017) statistics on **message-based** interests extracted from English-speaking users

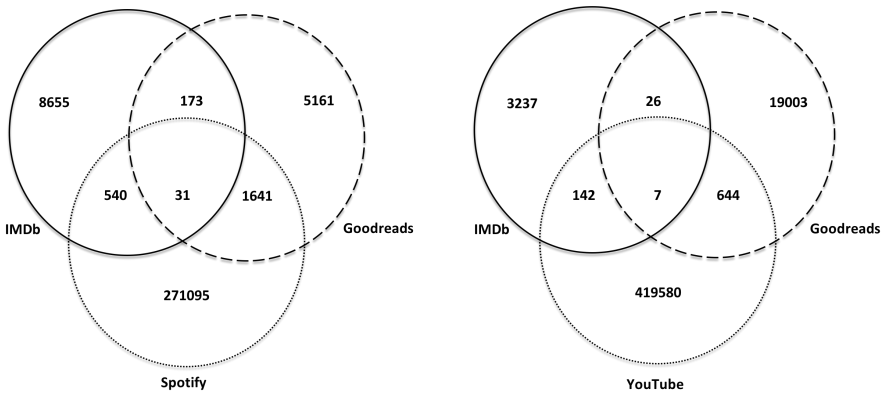


Figure 1: Venn Diagram of message-based interest *types* for our English dataset (left) and the dataset in Dooms et al.(2014)

message-based interests ($\|U\|$= 25,135 Italian speaking users)	**Music**	**Books**	**Movie**	**Total**
platform	*Spotify*	*ANobii*	*IMDb & TVShowTime*	*All*
#crawled tweets (tweets with selected hashtags)	273,256	12,198	2,229	287,683
#cleaned tweets (tweets fro which an URL was extracted)	70,330	12,193	2,119	84,642
# of unique interests with a mapping to a Wikipage	9,926	4,690	279	14,895
average #interests per user	3	9	7	6
average #users per interest	5	2	5	4
precision of Wikipedia mapping (on 3 samples of 250 items each)	96%	98%	100%	**98%**

Table 3: 6-months (April-September 2017) statistics on **message-based** interests extracted from Italian-speaking users

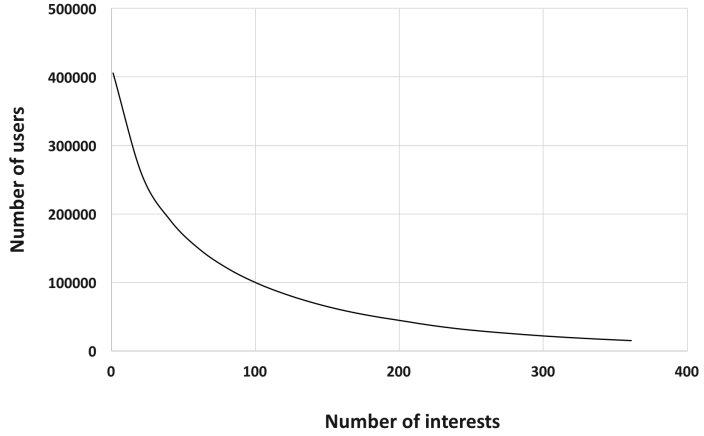


Figure 2: Distribution of interests induced from users's topical friends (English dataset)

Interests induced from topical friends ($\|U\|$=444,744 English speaking users)	
# of topical friends F'_t with indegree $\geq$ **40** in U	**409,743**
# of unique interests with a mapping to a Wikipage	58,789
average #interests per user	82
precision of Wikipedia mapping (tested on a sample of 1,250 items in F'_t)	90%

Table 4: 6-months (April-September 2017) statistics on interests induced from **topical friends** of English-speaking users

4. Description of the Dataset

The outlined process has been applied to two streams of Twitter data, in English and Italian, extracted during 6 months (April-September 2017) using Twitter APIs. We collected the maximum allowed Twitter traffic of English users mentioning service-related hashtags (e.g., #NowPlaying for Spotify), and the full stream of messages in Italian, since they do not exceed the maximum. As a final result, we obtained for a large number of users a variety of interests along with their corresponding Wikipedia pages. An excerpt of a Twitter user's interests is shown in Figure 1. In the example, we selected two interests from each of the four sources from which they have been induced:IMDb (movies), Goodreads (books), Spotify (music) and four interests from the user's topical friends. Although a detailed analytics of interest categories is deferred to further studies, the example shows the common trend that a user's interests, either extracted from his/her messages or from topical friends, are strongly related, and in same case identical. For example, the user in Table 1 frequently accesses the IMDb and Spotify services, and he/she is also a follower of the IMDb and Spotify Twitter account. Furthermore, his/her interest in the band The Magnetic Field emerges from both source types.

Overall, we followed 444,744 English-speaking and 25,135 italian-speaking users (the set U) who accessed at least one of the services mentioned in Section 3.1.. The general statistics of interests extracted from users' messages are shown in Tables 2 and 3.

In the English dataset we crawled more than 20M tweets from these users, of which, about 2.7M could be associated to the URL of a corresponding book, movie or music. On average, we collected 6 interests per user. What is more, several users have interests in at least two of the three domains. Figure 1 compares the Venn diagram of interest types in our dataset (left) with that reported in (Dooms et al., 2014) (right), to demonstrate the superior coverage of our dataset, even when considering only preferences extracted from users' messages.

The number and variety of extracted preferences is however mostly determined by the interests induced from users' topical friends, as shown in Table 4 (English dataset). Although, to ensure a high precision of the Wikipedia mapping step, we mapped only topical friends in $F'_t \subset F_t$ with a high in-degree from users in U (see Section 3.4.), the average number of interests induced for each user is as high as 82, and the distribution is shown in Figure 2. The Figure shows, e.g., that there are 100,000 users in U with ≥ 100 interests induced from their topical friends.

When merging the two sources of information, our dataset includes an average of 90 interests per user for about 450k users, in a large variety of domains. To the best of our knowledge, this is the largest multi-domain interest dataset reported in literature, and furthermore, we provide the unique feature of a reliable mapping to Wikipedia.

We release under creative commons license the dataset of English-speaking users along with their preferences. (`http://lmm.tweets.di.uniroma1.it/lmm/`). The dataset is in five files. Details are provided in the *readme* file. Further note, as we explained in Section 3., that the process of extracting interests from messages is almost free of errors (96% precision), while inducing interests from topical friends and subsequent mapping to Wikipedia has an estimated 10% error rate. However, as mentioned in Section 1., semantic techniques can be applied to reliably identify the main *categories* of interest for each user, an enhancement that we leave to future work.

5. Concluding Remarks

In this paper we presented a new dataset that captures, from Twitter messages and friendship lists, users' interests in multiple domains. We described the methodology to create the interests dataset and released a dataset extracted from an English Twitter stream collected during April-September 2017. The interests dataset can be extended to more languages and domains through the same methodology. Thus, the dataset and its extensions can be used in a number of applications in the domain of Recommender Systems, but not only. Although the dataset possibly includes extraction errors (which is a common problem in large, automatically extracted resources), the unique feature of mapping interests to Wikipedia articles and the large number of interests associated to each user, offer the possibility to identify for each user the "dominant" interest *categories*, on which Recommender Systems could rely when suggesting new items.

6. Acknowledgements

We greatly thank Dr.Giovanni Stilo, who developed the framework for extracting real-time data from Twitter streams, under the restrictions specified by Twitter. This work has been in part supported by the IBM Faculty Award #2305895190 (2017).

7. Bibliography

Adamopoulos, P. and Tuzhilin, A. (2014). Estimating the value of multi-dimensional data sets in context-based recommender systems. In *RecSys Posters*.

Barbieri, N., Bonchi, F., and Manco, G. (2014). Who to follow and why: link prediction with explanations. In *20th ACM SIGKDD*, pages 1266–1275. ACM.

Bennett, J., Lanning, S., et al. (2007). The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35. New York, NY, USA.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370.

Çano, E. and Morisio, M. (2015). Characterization of public datasets for recommender systems. In *Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), 2015 IEEE 1st International Forum on*, pages 249–257. IEEE.

Chaabane, A., Acs, G., Kaafar, M. A., et al. (2012). You are what you like! information leakage through users? interests. In *19th (NDSS)*.

Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., et al. (2010). The youtube video recommendation system. In *4th ACM conference on Recommender systems*, pages 293–296. ACM.

Delli Bovi, L., Telesca, L., and Navigli, R. (2015). Large-scale information extraction from textual definitions through deep syntactic and semantic analysis. *TACL*, 3:529–543.

Dooms, S., De Pessemier, T., and Martens, L. (2014). Mining cross-domain rating datasets from structured data on twitter. In *23rd International Conference on World Wide Web*, pages 621–624. ACM.

Dror, G., Koenigstein, N., Koren, Y., and Weimer, M. (2012). The yahoo! music dataset and kdd-cup?11. In *KDD Cup 2011*, pages 3–18.

Faralli, S., Stilo, G., and Velardi, P. (2015). Large scale homophily analysis in twitter using a twixonomy. In *24th, IJCAI, Buenos Aires, July 25-31, 2015*, pages 2334–2340.

Faralli, S., Stilo, G., and Velardi, P. (2017). Automatic acquisition of a taxonomy of microblogs users' interests. *Web Semantics: Science, Services and Agents on the World Wide Web*, 45:23–40.

Felfernig, A., Jeran, M., Ninaus, G., Reinfrank, F., Reiterer, S., and Stettinger, M. (2014). Basic approaches in recommendation systems. In *Recommendation Systems in Software Engineering*, pages 15–37. Springer.

Fouss, F. and Saerens, M. (2008). Evaluating performance of recommender systems: An experimental comparison. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 735–738. IEEE Computer Society.

Gesualdo, F., Stilo, G., Gonfiantini, M. V., Pandolfi, E., Velardi, P., Tozzi, A. E., et al. (2013). Influenza-like illness surveillance on twitter through automated learning of naïve language. *PLoS One*, 8(12):e82489.

Gunawardana, A. and Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research*, 10(Dec):2935–2962.

Harper, F. M. and Konstan, J. A. (2016). The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(4):19.

Kamishima, T. and Akaho, S. (2010). Nantonac collaborative filtering: A model-based approach. In *4th ACM conference on Recommender systems*, pages 273–276. ACM.

Linden, G., Smith, B., and York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80.

Mavalankar, A. A., Gupta, A., Gandotra, C., and Misra, R. (2017). Hotel recommendation system. *Internal Report*.

McFee, B., Bertin-Mahieux, T., Ellis, D. P., and Lanckriet, G. R. (2012). The million song dataset challenge. In *21st International Conference on World Wide Web*, pages 909–916. ACM.

Myers, S. A. and Leskovec, J. (2014). The bursty dynamics of the twitter information network. In *23rd international conference on World wide web*, pages 913–924. ACM.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Nichols, J., Mahmud, J., and Drews, C. (2012). Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 189–198. ACM.

Pazzani, M. J. and Billsus, D. (2007). Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer.

Pichl, M., Zangerle, E., and Specht, G. (2014). Combining spotify and twitter data for generating a recent and public dataset for music recommendation. In *Grundlagen von Datenbanken*, pages 35–40.

Pichl, M., Zangerle, E., and Specht, G. (2015). # nowplaying on# spotify: Leveraging spotify information on twitter for artist recommendations. In *International Conference on Web Engineering*, pages 163–174. Springer.

Sawant, S. and Pai, G. (2013). Yelp food recommendation system.

Schafer, J. B., Frankowski, D., Herlocker, J., and Sen, S. (2007). Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer.

Schinas, E., Papadopoulos, S., Diplaris, S., Kompatsiaris, Y., Mass, Y., Herzig, J., and Boudakidis, L. (2013). Eventsense: Capturing the pulse of large-scale events by mining social media streams. In *17th Panhellenic Conference on Informatics*, pages 17–24. ACM.

Shepitsen, A., Gemmell, J., Mobasher, B., and Burke, R. (2008). Personalized recommendation in social tagging systems using hierarchical clustering. In *2008 ACM RecSys*, pages 259–266. ACM.

Trewin, S. (2000). Knowledge-based recommender systems. *Encyclopedia of library and information science*, 69(Supplement 32):180.

Wang, H., Lu, Y., and Zhai, C. (2010). Latent aspect rating analysis on review text data: a rating regression approach. In *16th ACM SIGKDD i*, pages 783–792. ACM.

Yan, M., Sang, J., and Xu, C. (2014). Mining cross-network association for youtube video promotion. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 557–566. ACM.

RtGender: A Corpus for Studying Differential Responses to Gender

Rob Voigt[a], David Jurgens[b], Vinodkumar Prabhakaran[a], Dan Jurafsky[a], Yulia Tsvetkov[c]
[a]Stanford University, [b]University of Michigan, [c]Carnegie Mellon University
robvoigt@stanford.edu, jurgens@umich.edu, vinodkpg@stanford.edu, jurafsky@stanford.edu, ytsvetko@cs.cmu.edu

Abstract

Like many social variables, gender pervasively influences how people communicate with one another. However, prior computational work has largely focused on linguistic gender difference and communications *about* gender, rather than communications *directed to* people of that gender, in part due to lack of data. Here, we fill a critical need by introducing a multi-genre corpus of more than 25M comments from five socially and topically diverse sources tagged for the gender of the addressee. Using these data, we describe pilot studies on how differential responses to gender can be measured and analyzed and present 30k annotations for the sentiment and relevance of these responses, showing that across our datasets responses to women are more likely to be emotive and about the speaker as an individual (rather than about the content being responded to). Our dataset enables studying socially important questions like gender bias, and has potential uses for downstream applications such as dialogue systems, gender detection or obfuscation, and debiasing language generation.

Keywords: gender-annotated corpora, gender difference, gender bias, discourse, computational social science

1. Introduction

Language is a means for the construction of identity and social categories like gender; social issues such as gender bias, in turn, often take form in language. Linguistic datasets have been used both to debunk gender-biased myths — for example, contrary to stereotype women are not actually more talkative than men (Mehl et al., 2007) — and to identify social issues. For instance, women[1] journalists reach a smaller audience in terms of social media impressions (Matias and Wallach, 2012), and traditional gender stereotypes and unbalanced gender representation occur even in contemporary stories and movies (Fast et al., 2016; Sap et al., 2017).

Large datasets are particularly of use in this context due to the complex nature of differential responses to gender. However, previous computational work on language and gender has focused mainly on language *about* or *portraying* persons of a particular gender (Wagner et al., 2015; Flekova et al., 2016; Agarwal et al., 2015).

We thus present a large multi-genre dataset of online communication to enable research in a category of gender difference understudied in computational work: **responses to gender** in language. These include posts and talks labeled for the gender of the source,[2] along with comments given in response to the source texts. We collect such data from a variety of contexts, including:

- **Facebook (Politicians)**: Responses to Facebook posts from members of the U.S. House and Senate
- **Facebook (Public Figures)**: Responses to Facebook posts from other public figures, e.g., television hosts, journalists, and athletes
- **TED**: Responses to presentations from TED speakers
- **Fitocracy**: Responses to posts about fitness progress
- **Reddit**: Responses to Reddit comments across a variety of subreddits

These diverse datasets offer multiple perspectives on responses to gender. The first two sources (from Facebook and TED) represent the "broadcast" case, in which source texts (online posts and speech) from a small number of individuals (experts, authorities, and other public figures) receive a large number of responses which the source is unlikely to read and a discussion between the source and the responder is unlikely to continue. The second two (Fitocracy and Reddit) represent the "personal" case in which the responses are individualized, the source and responder may know one another and have an ongoing interaction afterwards.

2. Responses to Gender

Here we aim to encourage research on *responses to gender*. Contrasting with language *about* or *portraying* a given gender which address abstract representations of social categories, responses to gender are directed towards an individual person. We know that social characteristics of the addressee influence linguistic behavior; existing computational work has shown, for instance, that the gender of the interlocutor influences lexical choices of a speaker in spoken and written interactions (Boulis and Ostendorf, 2005; Jurgens et al., 2017; Prabhakaran and Rambow, 2017).

Looking at responses to gender also allows us to consider the important social issue of gender bias. Since important forms of bias (e.g., dehumanization or treating a person as their social category) often happen at the level of individual responses, responses to gender are an understudied but critical lens for studying gender bias.

The issue is related to that of abusive language (Xu et al., 2012; Clarke and Grieve, 2017), though often gender bias takes a less overt form than straightforward abuse. Social issues like gender bias are often not just about hostility but also behaviors such as stereotype-reinforcing benevolence (Eagly and Mladinic, 1989; Glick and Fiske, 1996;

[1]Throughout this paper we use the terms "woman" and "man" as labels for gender in preference to "female" and "male" since the latter terms are more commonly used as markers of sex.

[2]We use "source" to refer to the producer of the text being responded to (online posts and talk videos), and "responder" for the producer of the comment or response, regardless of its format.

	Dataset	Source Individuals		Source Text Count	Response Count	Response Word Count
BROADCAST	Facebook (Politicians)	M: 306	W: 96	399,037	13,866,507	376,114,950
	Facebook (Public Figures)	M: 41	W: 64	117,811	10,667,500	123,753,913
	TED Talks	M: 1,071	W: 349	1,671	190,425	15,549,984
PERSONAL	Fitocracy	M: 52,432	W: 47,498	318,535	318,535	6,606,087
	Reddit	M: 19,010	W: 11,116	1,453,512	1,453,512	44,537,612

Table 1: Basic statistics about the subcorpora within *RtGender*.

Jha and Mamidi, 2017). Nevertheless, biased responses to social categories like gender can lead to marginalization (Sue, 2010) and negatively impact a person's self-esteem and ability through mechanisms such as stereotype threat (Spencer et al., 1999). Perhaps most related to our work, Fu et al. (2016) analyze questions directed at men and women tennis players, finding that questions directed at men tend to be more about the game while questions directed at women are more likely to stray to topics about their appearance and off-court relationships. Tsou et al. (2014) similarly find comments on TED talks are more likely to be about the presenter than the content if the presenter is a woman.

In looking at responses to gender in our datasets (§3.), we note that some instances of gender bias may be overt, such as direct references to stereotypes ("Cool story babe, now make me a sandwich" - Facebook comment to television anchor Megyn Kelly) or inappropriate comments about physical appearances ("wow she is very sexy..."- TED comment to researcher Rachel Botsman); however, the larger problem is a subtle one in part because much of social bias is also not overt but rather implicit (Greenwald et al., 1985; Nosek et al., 2011). More commonly, many biases are exhibited through small but systematic differences in language which normally go unnoticed but, when viewed in aggregate, reveal large scale patterns in behavior towards a particular gender.

In the next section, we present **RtGender** – a corpus of responses to gender, compiled according to the following desired characteristics. First, it would be sufficiently large to allow for uncovering the subtle type of differentiation and bias mentioned above. Second, it would cover multiple genres and linguistic contexts to facilitate generalizable results. Third, it would allow for content and topic in the source texts to be controlled as much as possible so that researchers could know people are responding to the same types of sources, especially given existing research demonstrating gender-correlated clustering behavior by topic (Argamon et al., 2003; Bamman et al., 2014). Fourth, it would contain source texts from both authority figures and everyday persons, to facilitate the analysis of such subtle phenomena as implicit bias towards women authority figures (Rudman and Kilianski, 2000), while allowing for comparison to non-authority figures. Finally, it would ideally have gender labels for both the sources and the responders, to allow for gender-interaction analysis of interesting psychological phenomena like the propagation of self-favorable gender stereotypes (Rudman et al., 2001).

3. RtGender Datasets

We present five distinct datasets regarding *responses to gender* which fulfill many of the aforementioned desiderata. These data represent a variety of interactional contexts and relationships between the source and the responders.

The Facebook and TED "broadcast" datasets presented here contain many instances of responses to people in positions of authority or renown (politicians, topic experts, television personalities), and so can be analyzed with prior knowledge about the power differential between the source and the responders. The Fitocracy and Reddit "personal" datasets will allow research to contrast responses to gender in the public domain with more one-on-one interactions. In these datasets having interactional dyads of post-response also opens possibilities for studying normativity, for instance by asking whether comments on non-normative posts are more likely to exhibit elements of bias.

3.1. Facebook

Our largest dataset is comprised of top-level comments on Facebook posts from public figures, scraped from their public pages. We only include top-level comments (that is, comments directly responding to the post) to reduce the influence of comment-internal discussion so each comment is a response directly to the original poster. Each post is associated with the page of its relevant public figure, and includes metadata such as whether the post was text-only or included an image, video, or link.

The posts and responses in question are all public; however, to protect the anonymity of Facebook users in our dataset we remove all identifying user information as well as Facebook-internal information such as User IDs and Post IDs, replacing these with randomized ID numbers. Therefore users whose comments appear multiple times in our dataset may be compared, but without revealing their identity. We also only report commenter first names, since this is less identifying but still allows for running gender-identification algorithms. As a baseline for convenience we provide masculine/feminine ratios for these first names from Bergsma and Lin (2006).

We collect posts and their associated top-level comments for the categories of speakers described below. In each case we find the page for the speaker with a novel method for finding gender-labeled speakers from Wikipedia. Specifically, our method takes as input a Wikipedia category page such as `https://en.wikipedia.org/wiki/Category:American_female_tennis_players`, and for each name listed runs a search for public pages using Facebook's Graph API. If an exact match for the name appears in the top three results, and the category of the page matches a relevant category (for instance, "Public Figure" or "Athlete" in the case of female tennis players), and their gender is listed, and the page is "verified" with Facebook, we accept it as a member of that category and scrape the

Category	Example
Remarks on Appearance	Hot presenter.
Patronizing Tone	Stick to actually talking about the tenets of the topic and defer the blah blah blah to the politicians alone..
Doubting Expert Knowledge	I always thought the first approach to scientific study was to examine all evidence that dissproves an hypothesis.
Self-promotion is Perceived Negatively	After watching this, I know much more about Rachel Pike and what she does than the actual subject matter.

Table 2: Examples of categories of comments displaying potential forms of gender bias from the TED dataset. These categories were primarily observed in comments to women presenters.

relevant posts and comments.

Politicians. This subset contains all posts and associated top-level comments for all 412 current members of the United States Senate and House who have public Facebook pages meeting the requirements outlined above. This dataset inherently includes a strong control for content, since members of Congress tend to be talking about the same sorts of topics; each Congressperson is also labeled with their party affiliation to further facilitate controlling for cross-party stylistic and topical differences.

Public Figures. Beyond Congress, we consider other US public figures from the realms of journalism, fiction writing, television, film, and athletics. This subset contains posts and associated top-level comments for 105 such public figures, currently drawn from the following sets of Wikipedia categories:

- American television news anchors, American television journalists, American television talk show hosts, Political analysts
- American film actresses, American male film actors, American television actresses, American male television actors
- American male tennis players, American female tennis players, Olympic track and field athletes of the United States
- 21st-century American novelists

3.2. TED

TED Talks are influential videos from experts on a variety of topics ranging from education, business, science, tech and creativity.[3] The TED website also allows viewers to post comments in response to each video, which provides us an opportunity to study how these responses are impacted by the gender of the expert presenters. We include a dataset scraped from the TED website of 190,425 labeled for presenter gender. Gender labels were initially drawn from Mirkin et al. (2015) and the remaining labels were done manually. Talks that consisted solely of a dance or music performance, or talks presented by more than one speaker were excluded.

This domain has been previously explored in NLP: we know that TED presenters are more commonly male, and videos of talks by male speakers are more viewed and liked on YouTube (Sugimoto et al., 2013). Furthermore, considering responses to gender, Tsou et al. (2014) note that commenters are more emotional when the presenter is a woman. However, existing sources of this data such as `https://www.idiap.ch/dataset/ted` are not labeled for presenter gender.

3.3. Fitocracy

Fitocracy[4] is a social media fitness website in which users log and discuss their fitness-related activities. We include a dataset of 318,536 "status updates" and their corresponding top-level comments from Fitocracy which users have posted about their progress. We include only the first comment after a post because comments are not nested, so discussions can diverge as following comments may quickly become responses to previous comments; however, the first post is necessarily in direct response to the original post. Building upon the observations of Fu et al. (2016) on gender differences in questions posed to tennis players, we view Fitocracy as an ideal dataset for examining how gender stereotypes around fitness and sports play out in everyday interactions. In this dataset we have confident self-reported labels for the gender of most posters and commenters; over 91% of users of Fitocracy self-report both gender and age on their profile pages, and we include this information in the dataset.

3.4. Reddit

Posts on Reddit are a common source of data for computational linguistic analysis; in this corpus, we include a dataset of 1,453,512 Reddit post-response pairs for which we know the gender of the source poster. The data was gathered by finding gender-indicating flairs used on different subreddits (e.g,. "male" on /r/AskMen). For each of these users, we then find all of their posts in other subreddits and collect the first response to each post - as in other contexts, we take only the first response to guarantee it is directed towards the source poster. We also tag the responder for gender when we have that data available, which occurs for about 9.2% of our examples.

This dataset covers a wide variety of subreddits, so while the sources of our gender tags are from a relatively limited domain, ultimately researchers can control for content substantially by sampling the dataset in particular subreddits of interest.

4. Analysis and Challenges

In this section we discuss a preliminary qualitative and quantitative analysis regarding differential responses to gender in our new datasets, designed to illustrate the kind of studies they enable.

[3] `https://www.ted.com/talks`

[4] `https://www.fitocracy.com/`

Responder Gender		*Source Gender* WOMAN	MAN
	WOMAN	girl, gorgeous, can, if, yay, exercise, it, find, girlie, to, love, we, feel, ", do, each, mama, site, or, yoga, walk, help, started, go, healthy	HAPPY_EMOJI, thank, thanks, ..., haha, ?, no, well, problem, :p, mister, !, pleasure, follow, props, prop, lol, welcome, :d, bomb, handsome, very, for, my, course
	MAN	HAPPY_EMOJI, !, you, welcome, thank, your, follow, great, pp, pleasure, hope, love, back, following, very, luck, you're, girl, well, :d, are, awesome, thanks, for, young, beautiful, smile, hi, fun	man, bro, mate, dude, ., brother, buddy, [NUMBER], brah, bench, ,, bud, shit, 0x0, yeah, i, squat, press, lifts, sets, fuck, gains, 0kg, chest, last, strength, week, guy, this, ohp

Table 3: Top 30 words in comments in the Fitocracy dataset by log-odds based on the gender of the commenter and original poster.

Dataset	Source Gender Prediction Accuracy
Facebook (Political)	63.9%
Facebook (Public Figures)	80.3%
TED Talks	80.5%
Fitocracy	57.7%
Reddit	53.5%

Table 4: Cross-validation accuracy across contexts at predicting the gender of the source from the text of their post/talk.

Table 2 presents some qualitative examples of TED comments directed towards women presenters that exhibit possible gender bias. Some are overt, such as remarks on appearance, but others are more subtle. For authors of each gender, Table 3 gives the top 30 words in Fitocracy responses most associated with the responder's gender, computed using the weighted log-odds method of Monroe et al. (2008). The word preferences of responders show a substantial gender-correlated signal in this data. Men commenting on posts by men use many close terms of informal address ("bro," "dude," "brother," "buddy") and likewise for women commenting on posts by women ("girl," "girlie," "mama"). Cross-gender comments, however, are more emotive, with prominent use of emoticons, emoji, and exclamation points, as well as more playful and interactive language (talk of "following" each other and use of second person pronouns) and discussion of the addressee's appearance, e.g., "beautiful" (M→W) and "handsome" (W→M).

Each dataset in turn presents a unique challenge for researchers. The Facebook data is large and noisy: the comments are relatively unmoderated and may also be responding to photos and URLs in the source posts, rather than just the textual content of the post itself. The TED talks exemplify the challenge of separating gender difference from topical choice, since selection bias on the part of the TED organizers means there are more talks from men and talks from women are more likely to be about gendered topics. For Fitocracy, as Table 3 shows, the language used is often very positive overall, so a computational definition of bias must be able to also capture benevolent differential treatment. The Reddit data covers a very broad spectrum of topical content, and in the majority of subreddits gendered flair is not visible so the signal for differential responses to gender is much more subtle.

One important axis of variation across the linguistic environments of these contexts is to consider how differently men and women tend to speak in that context; a simple way to quantify this is to ask how well a predictive model can distinguish source posts written by men versus women. Table 3.3. shows ten-fold cross-validation accuracies of a simple unigram logistic regression model at predicting the gender of the source from the text in the source post. In the case of TED, accuracy is given at predicting gender from a sample of lines in the source transcript. Notice the wide diversity across contexts. While gender differences in the "personal" Reddit and Fitocracy contexts are relatively minimal, gender difference of the source is amplified in the "broadcast" contexts where posts by men and women are highly separable even with a simple unigram model.

5. Relevance and Sentiment Annotations

Our pilot analyses revealed that the RtGender datasets have the potential to offer interesting insights on differential responses to gender across diverse domains. To expand the possible range of questions that may be asked of this data, we conducted a crowdsourced annotation effort on a sample of the responses across our datasets.

Inspired by the annotation task for TED talk comments proposed by Tsou et al. (2014), we labeled over 15,000 post-response pairs with annotations for the relevance of the response to the source and the sentiment of the response. For this task we asked crowd workers on Amazon Mechanical Turk to read a post-response pair and mark whether it was relevant to the CONTENT ONLY, POSTER ONLY, CONTENT AND POSTER, or if it was IRRELEVANT. In the case of comments on TED talks since there was no "original post" we provided a list of the talk's keywords to help participants determine its relevance. We then asked about the sentiment of response (POSITIVE, NEGATIVE, MIXED, or NEUTRAL), regardless of its relevance. Our annotation interface is shown in Figure 1.

Crowd workers were paid $0.20 for completing one run of 5 post-response pairs. To control for annotation quality, on each run for a random one of the pairs we replaced the response with a snippet of text with a known expected response. These were drawn from the following sources:

- Random sentences from articles in the New York Times in 2007 (expected response: IRRELEVANT)



Figure 1: Screenshot of our relevance and sentiment annotation interface.

Dataset	Annotated Examples
Facebook (Politicians)	3,872
Facebook (Public Figures)	2,884
TED Talks	2,648
Fitocracy	2,900
Reddit	2,728

Table 5: Quantity of available post-response pairs labeled with relevance and sentiment annotations for each dataset.

- Random phrases with known polarity from the Stanford Sentiment Treebank (Socher et al., 2013) (expected response: IRRELEVANT or POSITIVE/NEGATIVE, respectively)
- Poster/speaker-directed utterances automatically generated with a heuristic method to have known polarity (expected response: relevant to POSTER ONLY or CONTENT+POSTER; known POSITIVE/NEGATIVE polarity as appropriate). Examples:
  - you are just fantastic, believe in yourself! (POSITIVE)
  - Stop trying. You are so garbage! (NEGATIVE)

Any runs for which these control questions were answered incorrectly were discarded, constituting 11.9% of total runs.

We performed basic analyses on these annotations to better understand the types of future research they might enable. Firstly, we ran a set of mixed-effects models predicting aspects of response relevance and sentiment, with gender as a fixed effect and dataset context as a random effect. Our overall results replicate Tsou et al. (2014), finding that in general responses to women were more likely to be about the source poster or speaker as an individual ($b=0.20$, $p<0.01$) and were more emotive (having non-neutral sentiment) ($b=0.17$, $p<0.01$) than responses to men. Interestingly, sentiment in responses to women was higher across the board; whether this represents "benevolent sexism" or genuine positive sentiment is an interesting and complex topic for future research.

However, the contexts represented by each dataset acted very differently. When we restrict the above analyses to only the "personal" Fitocracy and Reddit contexts we find no gender-based differences in response relevance ($b=0.01$, $p=0.87$), and the magnitude of the emotiveness difference is greatly reduced ($b=0.11$, $p=0.046$). This finding suggests a potential powerful effect of social distance in exacerbating gender bias, in line with classic social psychological findings on how group diffusion of responsibility can lead to increased dehumanization (Bandura et al., 1975).

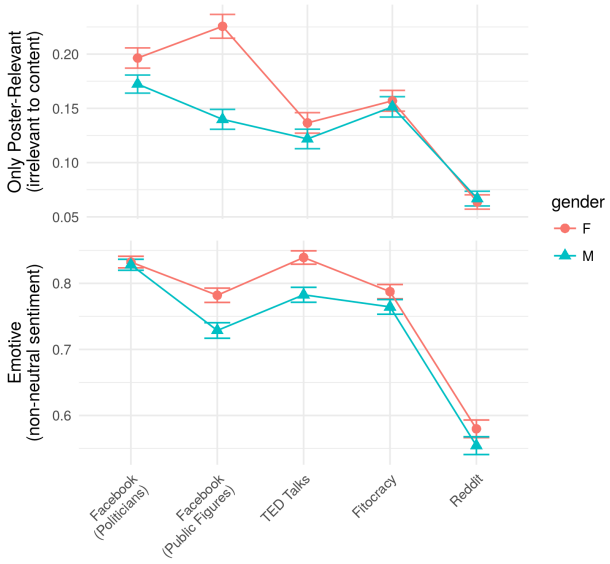


Figure 2: Cross-context characteristics of the responses per the relevance and sentiment annotations in RtGender.

6. Conclusion

Gender is a performative social phenomenon in which individual behavior is often shaped – subtly, over a lifetime – by the responses to that behavior (Lakoff, 1973; Butler, 1990). To encourage computational study in this area, in this paper we presented five large datasets in a corpus called RtGender that capture differential responses to gender online in a variety of genres, contexts, and social roles of the interacting participants, and publicly available for research purposes.[5] We labeled a sample of the responses in the corpus with annotations for relevance and sentiment, and gave some initial analyses of the data and resulting annotations. We found qualitative and quantitative evidence for gender bias in the responses, suggesting a need for future work in this area that we hope this corpus will facilitate. By studying responses to gender we can learn a great deal about the social construction of gender and other social categories in general.

Acknowledgments

This work was supported by the Stanford Data Science Initiative and the National Science Foundation through award IIS-1526745. The first author gratefully acknowledges the support of the Stanford Interdisciplinary Graduate Fellowship.

7. Bibliographical References

Agarwal, A., Zheng, J., Kamath, S. V., Balasubramanian, S., and Dey, S. A. (2015). Key Female Characters in Film Have More to Talk About Besides Men : Automating the Bechdel Test. *uman Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 830–840.

Argamon, S., Koppel, M., Fine, J., and Shimoni, A. R. (2003). Gender, genre, and writing style in formal writ-

[5] https://cs.cmu.edu/~ytsvetko/rtgender/

ten texts. *Text - Interdisciplinary Journal for the Study of Discourse*, 23(3):321–346.

Bamman, D., Eisenstein, J., and Schnoebelen, T. (2014). Gender and variation in social media. *Journal of Sociolinguistics*, 18(2):135–160, apr.

Bandura, A., Underwood, B., and Fromson, M. E. (1975). Disinhibition of aggression through diffusion of responsibility and dehumanization of victims. *Journal of Research in Personality*, 9(4):253–269, dec.

Bergsma, S. and Lin, D. (2006). Bootstrapping path-based pronoun resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL - ACL '06*, pages 33–40, Morristown, NJ, USA. Association for Computational Linguistics.

Boulis, C. and Ostendorf, M. (2005). A quantitative analysis of lexical differences between genders in telephone conversations. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics - ACL '05*, (June):435–442.

Butler, J. (1990). *Gender Trouble - Feminism and the Subversion of Identity*, volume 53. Routledge.

Clarke, I. and Grieve, J. (2017). Dimensions of Abusive Language on Twitter. In *Proceedings of the First Workshop on Abusive Language Online*, pages 1–10.

Eagly, A. H. and Mladinic, A. (1989). Gender stereotypes and attitudes toward women and men, dec.

Fast, E., Vachovsky, T., and Bernstein, M. (2016). Shirtless and dangerous: Quantifying linguistic signals of gender bias in an online fiction writing community. In *Proceedings of the 10th International AAAI Conference on Web and Social Media (ICWSM '16)*, number Icwsm, pages 112–120.

Flekova, L., Carpenter, J., Giorgi, S., Ungar, L., and Preo\ctiuc-Pietro, D. (2016). Analyzing Biases in Human Perception of User Age and Gender from Text. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 843–854.

Fu, L., Danescu-Niculescu-Mizil, C., and Lee, L. (2016). Tie-breaker: Using language models to quantify gender bias in sports journalism. *Proceedings of the IJCAI Workshop on NLP Meets Journalism*.

Glick, P. and Fiske, S. (1996). The ambivalent sexism inventory: Differentiating hostile and benevolent sexism. *Journal of personality and social psychology*.

Greenwald, A. G., Krieger, L. H., Greenwaldt, A. G., Kriegertt, L. H., Eberhardt, J., Kang, J., Newkirk, T., and Rachlinski, J. (1985). Implicit Bias: Scientific Foundations. *California Law Review*, 47(445):35–421.

Jha, A. and Mamidi, R. (2017). When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 7–16.

Jurgens, D., Tsvetkov, Y., and Jurafsky, D. (2017). Writer profiling without the writer's text. In *Proceedings of the 9th International Conference on Social Informatics (SocInfo)*.

Lakoff, R. T. (1973). *Language and Woman's Place*.

Matias, J. N. and Wallach, H. (2012). Working Paper : Modeling Gender Discrimination by Audiences of Online News.

Mehl, M. R., Vazire, S., Ramírez-Esparza, N., Slatcher, R. B., and Pennebaker, J. W. (2007). Are women really more talkative than men? *Science (New York, N.Y.)*, 317(5834):82, jul.

Mirkin, S., Nowson, S., Brun, C., and Perez, J. (2015). Motivating personality-aware machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1102–1108, Lisbon, Portugal, September. Association for Computational Linguistics.

Monroe, B. L., Colaresi, M. P., and Quinn, K. M. (2008). Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Nosek, B. A., Hawkins, C. B., and Frazier, R. S. (2011). Implicit social cognition: from measures to mechanisms. *Trends in cognitive sciences*, 15(4):152–9, apr.

Prabhakaran, V. and Rambow, O. (2017). Dialog structure through the lens of gender, gender environment, and power. *D&D*, 8(2):21–55.

Rudman, L. and Kilianski, S. (2000). Implicit and explicit attitudes toward female authority. *Personality and social psychology*.

Rudman, L. A., Greenwald, A. G., and McGhee, D. E. (2001). Implicit Self-Concept and Evaluative Implicit Gender Stereotypes: Self and Ingroup Share Desirable Traits. *Personality and Social Psychology Bulletin*, 27(9):1164–1178.

Sap, M., Prasetio, M. C., Holtzman, A., Rashkin, H., and Choi, Y. (2017). Connotation Frames of Power and Agency in Modern Films. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2319–2324.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Spencer, S. J., Steele, C. M., and Quinn, D. M. (1999). Stereotype Threat and Women's Math Performance. *Journal of Experimental Social Psychology*, 35(1):4–28, jan.

Sue, D. W. (2010). *Microaggressions in everyday life: Race, gender, and sexual orientation*. John Wiley & Sons.

Sugimoto, C. R., Thelwall, M., Larivière, V., Tsou, A., Mongeon, P., and Macaluso, B. (2013). Scientists Popularizing Science: Characteristics and Impact of TED Talk Presenters. *PLoS ONE*, 8(4):e62403, apr.

Tsou, A., Thelwall, M., Mongeon, P., and Sugimoto, C. R. (2014). A community of curious souls: An analysis of commenting behavior on TED Talks videos. *PLoS ONE*, 9(4):e93609, apr.

Wagner, C., Garcia, D., Jadidi, M., and Strohmaier,

M. (2015). It's a Man's Wikipedia? Assessing Gender Inequality in an Online Encyclopedia. *Arxiv*, 1501.06307:1–10.

Xu, J., Jun, K., Zhu, X., and Bellmore, A. (2012). Learning from Bullying Traces in Social Media. *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 656–666.

A Neural Network Model for Part-Of-Speech Tagging of Social Media Texts

Sara Meftah*, Nasredine Semmar*, Fatiha Sadat+
*CEA, LIST, Vision and Content Engineering Laboratory
F-91191, Gif-sur-Yvette, France
{sara.meftah, nasredine.semmar}@cea.fr
+Université du Québec à Montréal, UQÀM
201 Président Kennedy Avenue H2X 3Y7, Montréal, Canada
sadat.fatiha@uqam.ca

Abstract

In this paper, we propose a neural network model for Part-Of-Speech (POS) tagging of User-Generated Content (UGC) such as Twitter, Facebook and Web forums. The proposed model is end-to-end and uses both character and word level representations. Character level representations are learned during the training of the model through a Convolutional Neural Network (CNN). For word level representations, we combine several pre-trainned embeddings (Word2Vec, FastText and GloVe). To deal with the issue of the poor availability of annotated social media data, we have implemented a Transfer Learning (TL) approach. We demonstrate the validity and genericity of our model on a POS tagging task by conducting our experiments on five social media languages (English, German, French, Italian and Spanish).

Keywords: Part-of-Speech Tagging, Social Media Texts, Low-Resources Languages, Neural Networks, Transfer Learning

1. Introduction

Recent approaches based on end-to-end Deep Neural Networks (DNNs) have shown promising results for Natural Language Processing (NLP). Most of proposed neural models for sequence labeling (including POS taggers) use Recurrent Neural Networks (RNNs) and its variants (Long Short-Term Memory networks - LSTMs and Gated Recurrent Units - GRUs), and Convolutional Neural Networks (CNNs) for character-level representations. Indeed, previous studies (Jozefowicz et al., 2016) have shown that CNNs represent an effective approach to extract morphological information (root, prefix, suffix, etc.) from words and encode it into neural representations, especially for morphological rich texts (Chiu and Nichols, 2015; Ma and Hovy, 2016).

The actual performance of POS taggers trained from treebanks in the newswire domain, such as the Wall Street Journal (WSJ) corpus of the Penn TreeBank (PTB) (Marcus et al., 1993) and evaluated on in-domain data is close to human level, thanks to deep learning techniques trained on huge annotated datasets (97.64% accuracy by (Choi, 2016)). Contrariwise, approaching human-level accuracy on more complex domains such as User Generated Content (UGC) on social media is still a hard problem. Especially conversational texts (Twitter, Web blogs, SMS texts, etc.). This is due to the conversational nature of the text, the lack of conventional orthography, the noise, linguistic errors, spelling inconsistencies, informal abbreviations and the idiosyncratic style. Also, Twitter poses an additional issue by imposing 280 characters limit for each tweet.

The application of models trained on well-structured corpora such as WSJ fails to work effectively on noisy text. As illustrated in (Gimpel et al., 2011), the accuracy of the Stanford POS tagger (Toutanova et al., 2003) trained on WSJ falls from 97% on standard English to 85% accuracy on tweets. The main reason for this drop in accuracy is that tweets contain lot of Out-Of-Vocabulary (OOV) words compared to standard text. In addition, NLP's DNNs models often require to be trained on huge volumes of annotated data to produce powerful models and prevent over-fitting. Hence, the construction of a DNN model for UGC data needs huge amounts of annotated data with POS labels to provide high performances. However, available annotated in-domian datasets are very small.

In this paper, we present a POS tagger for multiple social media datasets, using a Transfer Learning (TL) based end-to-end neural model. In a TL scenario, the knowledge learned by handling one problem is used to help solving different but related problems.

The goal of this work is to examine the effectiveness of TL for POS tagging across domains and tasks. Experiments show significant improvements over several languages (English, French, German, Italian and Spanish).

2. Related Work

Our work is related to two lines of research: (1) Transfer Learning (2) POS tagging of social media texts. Below we discuss the state-of-the-art of each one.

2.1. Transfer Learning

As discussed in the introduction, high performing NLP's neural models often require huge volumes of annotated data to produce powerful models and prevent over-fitting. Consequently, in the case of social media content, it is difficult to achieve the performances of state-of-the-art models based on hand-crafted features by applying neural models trained on small amounts of annotated data. For this reason TL was proposed to exploit huge annotated out-of-domain data-sets. TL aims at performing a task on a target dataset using features learned from a source dataset (Pan and Yang, 2010).

Furthermore, the successes of neural models for many tasks over the last few years have intensified the interest for studying TL for neural networks.

In particular, TL was largely exploited in computer vision using pre-trained CNNs to generate representations for

novel tasks; some of the parameters learned on the source dataset are used to initialize the corresponding parameters of the CNNs for the target dataset.

In the past few years, few studies have been conducted on TL for neural based models in the field of NLP. It consists in performing a task on a low-resource target problem using features learned from a high-resource source problem. For instance, TL has been successfully applied in neural speech processing and machine translation (Zoph et al., 2016).

Two studies have been recently performed on TL for neural networks based models in sequence labeling: Yang et al. (2017) examined the effects of TL for deep hierarchical recurrent networks across domains, applications, and languages, and showed that significant improvement can be obtained. Lee et al. (2017) used cross-domain TL for Named Entity Recognition (NER) (specifically patient note de-identification), and showed that TL may be especially beneficial for a target dataset with small number of examples.

2.2. Part-Of-Speech Tagging of Social Media Texts

POS tagging is a sequence labeling problem, by assigning to each word its disambiguate part-of-speech (Verb, Noun, Adjective, etc.) in the sentential context in which the word is used. This information is useful for higher-level NLP applications such as semantic relations extraction, sentiment analysis, automatic summarization and machine translation.

Most performing traditional POS tagging models for social media content are linear statistical models, including Hidden Markov Models (HMM), Maximum Entropy Markov Models (MEMMs), Conditional Random Fields (CRF) and linear classifiers like SVM-based taggers.

There are two principal state-of-the-art works for English tweets POS tagging, both based on hand-crafted features, Ritter et al. (2011) published a set of 787 hand-annotated English tweets and proposed in (Derczynski et al., 2013) a model based on hidden Markov Models and a set of normalization rules, external dictionaries and lexical features. Gimpel et al. (2011) and Owoputi et al. (2013) constructed 1827 and 547 hand-annotated tweets, respectively, using the same tag-set. They proposed a model based on First-order maximum entropy Markov model (MEMM), engineered features like brown clustering and lexical features.

Nooralahzadeh et al. (2014) proposed a POS tagging system for French Social Media content using Conditional random fields (CRFs) with a set of several hand-crafted features.

These models rely heavily on hand-crafted features and task specific resources (morphological, orthographic and lexical features and external resources such as gazetteers or dictionaries). However, such task-specific knowledge is costly to develop and making sequence labelling models difficult to adapt to new tasks or new domains.

Recently a neural network model for English tweets POS tagging was proposed by Gui et al. (2017) (TPANN), they used Adversarial Neural Networks to leverage huge amounts of unlabeled tweets and labeled out-of-domain data (WSJ). TPANN achieves high performances compared to the former works. The model proposed in (Gui et al., 2017) requires that labeled in-domain-data and labeled out-of-domain data share the same tag-set (a mapping is necessary in case of tag-sets mismatch).

3. Contributions

This work is built on the basis of the recently published paper (Meftah et al., 2017), where cross-domain TL was successfully used for English tweets POS tagging by exploiting available huge amounts of POS labeled corpora of a similar domain (standard English). The knowledge learned on the parent neural network trained on enough standard English labeled data was transferred to initialize the child network, further fine-tuned on small annotated English Twitter corpus. Nevertheless, the present paper includes the following new contributions:

- We investigate a second scenario, cross-task TL, where the parent network is trained on in-domain data annotated with Named Entities (NE).
- We show that TL method is efficient on multiple social media languages (English, French, Spanish, German and Italian).
- We analyze how cross-task TL may address the issue of the low-availability of annotated data and improve performances.

4. Neural Model Architecture

The neural model that we use for TL experiments is the same used in (Meftah et al., 2017), based on bidirectional hierarchical Gated Recurrent Units (GRUs). Figure 2 shows an overview of the model's architecture[1].

4.1. Features Representation

In order to preserve both semantic and syntactic information of words, each word from the input sequence is represented by a combination of two vectors of features, character-level and word-level embedding. Therefore, each word in the input sentence is represented by a combination of two vectors:

1. Pre-trained words embedding: We initialize word-level embedding with a concatenation of different pre-retrained words embedding (details in section 6.3.) to accurately capture words' semantics.
2. Character level embedding: To learn orthographic features at the character level, we use a CNN architecture similar to that of Ma and Hovy (2016). As illustrated in figure 1. Each word is represented with a $v \times l$ dimensional matrix, next it's embedded into a $d \times l$ dimensional matrix, where v is character's vocabulary size, l is the maximal length of words and d is character embedding's dimension. Then, we take the character embeddings and apply (30×3)-stacked convolutional layers, followed by a max-pooling operation. Finally, the result is passed to a fully-connected layer using a Rectifier Linear Unit (ReLU) activation function.

[1]The model's architecture is the same among all datasets and tasks.

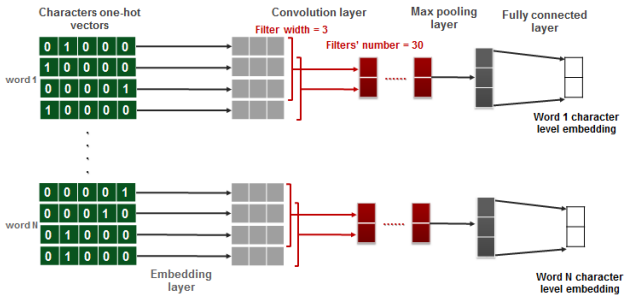


Figure 1: Convolutional Neural Network architecture for character-level embedding.

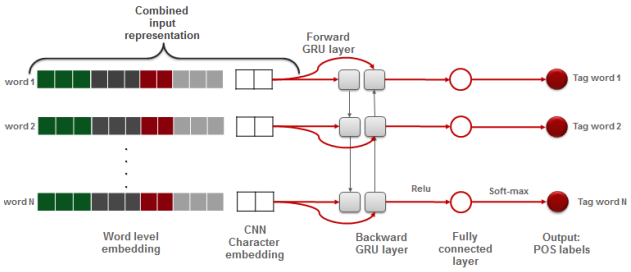


Figure 2: Overall system design. First, the system embeds each word of the current sentence into two representations: character level representation using a CNN network and a word level representation by combining different pre-trained models. Then, the two representations are combined and fed into a bidirectional GRU layer, the resulting vector is fed to a fully connected layer and finally a softmax layer to perform POS tagging.

4.2. Sequence Labelling with Gated Recurrent Units (GRUs) Layer

Word vectors (the combination between character level embedding and word level embedding CNN) are fed into a 100 dimension Gated Recurrent Units (GRUs) layer, a variant of RNNs.

Let $(x_1, x_2, ..., x_t, ..., x_n)$ the input sequence of the GRUs layer, which is in our case a sequence of n D-dimensional word vectors, where n is sentence's length and D is word vectors' dimension.

Let h_t be the GRU hidden state at time-step t. Formally, a GRU unit at a time-step t takes x_t and the previous hidden state h_{t-1} as input, and outputs the current hidden state h_t. Each gated recurrent unit can be expressed as follows:

$$r_t = \sigma(W_{rx}x_t + W_{rh}h_{t-1}) \quad (1)$$

$$z_t = \sigma(W_{zx}x_t + W_{zh}h_{t-1}) \quad (2)$$

$$\hat{h}_t = \tanh(W_{hx}x_t + W_{hh}(r_t \otimes h_{t-1})) \quad (3)$$

$$h_t = z_t \otimes h_{t-1} + (1 - z_t) \otimes \hat{h}_t \quad (4)$$

Where W's are model parameters of each unit, $\hat{h}_t$ is a candidate hidden state that is used to compute h_t, σ is an element-wise sigmoid logistic function defined as $\sigma(x) = 1/(1 + e^{-x})$, and $\otimes$ denotes element-wise multiplication of two vectors. The update gate z_t controls how much the unit updates its hidden state, and the reset gate r_t determines how much information from the previous hidden state needs to be reset.

4.3. Fully-connected Layer and Softmax Layer

The output of the forward GRUs and the backward GRUs at each time-step are combined and fed through a 80 dimension linear (fully connected) layer with a ReLU activation, followed by a final dense layer with a softmax activation to generate a probability distribution over the output classes at each time-step.

5. Transfer Learning Approach

TL is applied to address the problem of the need in annotated data for POS tagging of social media texts. It consists in learning a parent neural network on a source problem with enough data, then transferring a part of its weights to represent data of a target problem with few training examples.

We experiment two scenarios of TL. The first scenario is cross-domain transfer; knowledge is transferred from a source domain to a target domain. In our case, the source domain is the standard form (well-established) of a language and the target domain is the social media text of the same language. The source and the target problems are trained for the same task (POS tagging), even if source and target datasets do not share the same tag-set.

As illustrated in the figure 3, we have a parent neural network N_p with a set of parameters θ_p splitted into two sets: $\theta_p = (\theta_p^1, \theta_p^2)$. And a child network N_c with a set of parameters θ_c splitted into two sets: $\theta_c = (\theta_c^1, \theta_c^2)$.

(1) We learn the parent network on annotated data from the source problem on a source dataset D_s. (2) We transfer weights of the first set of parameters of the parent network N_p to the child network N_c: $\theta_c^1 = \theta_p^1$. (3) Then, the child network is fine-tuned to the target problem by training it on the target dataset D_c.

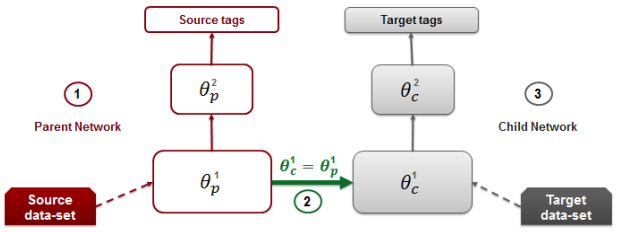


Figure 3: Cross-domain Transfer Learning scheme.

The second scenario is cross-task transfer; the source and the target problems share the same domain and the same language (social media text of the same language). However, tasks are different (The source problem's task is NER and the target's is POS tagging) to exploit the underlying similarities of the two tasks.

As illustrated in the figure 4, the parent neural network and the child network share the same first set of parameters (The feature extractor) : $\theta_p^1 = \theta_c^1 = \theta^1$.

θ^1 are jointly optimized by the two tasks, while task specific parameters θ_c^2 and θ_p^2 are trained for each task separately.

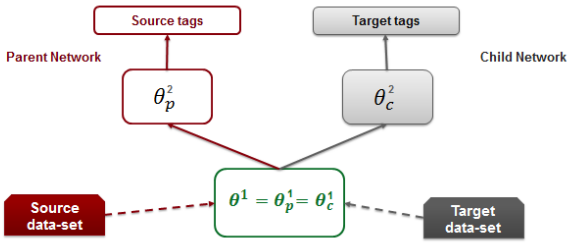


Figure 4: Cross-task Transfer Learning scheme.

6. Experimental Setup

6.1. Datasets

We use two types of source datasets for parent neural network training: (1) large out-of-domain POS-labeled data from resource-rich domains for the first TL scenario, and (2) NE-labeled in-domain data for the second scenario. For child neural network model fine-tuning, we use small POS labeled in-domain datasets.
In this section, we report the source and target datasets for each language, on which we perform our evaluations. The statistics of the datasets[2] are described in table 1.

6.1.1. English

As a source dataset for English experiments, we use a standard English corpus, the **Wall Street Journal (WSJ)** part of the PTB, annotated with 36 POS tags.
We evaluate our approach on three target datasets. The **NPS IRC Chat Corpus** (Forsythand and Martell, 2007) of 10,567 posts gathered from various online chat services. And two Twitter datasets:

- **The T-PoS corpus** of 787 hand-annotated English tweets, introduced by (Ritter et al., 2011), which uses the same tag-set as PTB's (Marcus et al., 1993), plus four Twitter special tags: URL for urls, HT for hashtags, USR for username mentions and RT for retweet signifier (40 tags in total). For our experiments on T-PoS, we use the same data splits used in (Derczynski et al., 2013); 70:15:15 into training, development and test sets named T-train, T-dev and T-eval.
- **The ARK corpus** was published on two parts, the first, Oct27 of 1827 hand-annotated English tweets, published in (Gimpel et al., 2011) and the second, Daily547 of 547 tweets published by Owoputi et al. (2013), using a novel and coarse grained tag-set (25 tags). For example, its V tag corresponds to any verb, conflating PTB's VB, VBD, VBG, VBN, VBP, VBZ, and MD tags. We split the Oct27 dataset into training-set and development-set (70:30) (data splits portions are not mentioned in original papers) and Daily547 as a test set.

6.1.2. French

As a source dataset for French experiments, we use a standard French corpus, **French-Tree-Bank (FTB)** (Abeillé et al., 2003), a POS-annotated French newspaper corpus.
We evaluate our approach on two publicly available POS-labeled User Generated (UG) French content datasets:

- **The French web 2.0 (Fr2.0) (Seddah et al., 2012)** is a set of 1700 sentences extracted from various types of French Web. (1) Micro-blogging: Facebook and Twitter. (2) web forums: French health forum DOCTISSIMO[3] and video games website JEUXVIDEOS[4].

  The tag-set includes 28 POS tags from FTB, plus combined tags for contracted tokens. For instance, the non-standard French contraction *tes* (widely used by French web's users), which stands for *tu es*, would have been tagged CLS and V (subject clitic and finite verb) in FTB. The non-standard contracted token *tes* is then tagged CLS+V. And specific tags to social media, including HT and RT. Twitter at-mentions as well as urls and e-mail addresses have been tagged NPP which is the main difference with other annotations of UG content.
- **ExtremeUGC dataset (UGC) (Alonso et al., 2016)**: contains user-generated content from three different sources. Two of them are logs of multi-player video-game chat sessions: MINECRAFT and LEAGUE OF LEGENDS, the last one is cooking-related user questions from MARMITON, a popular cooking French website. Datasets are annotated with the same scheme as the Fr2.0.

6.1.3. Spanish, Italian and German:

The xLiMe Twitter Corpus (Rei et al., 2016): is a Multilingual Social Media Linguistic Corpus, contains manually annotated Spanish, German and Italian tweets [5]. The corpus is annotated with POS tags and NE. The POS tag-set consists of the Universal Dependencies tag-set, plus Twitter specific tags based on (Gimpel et al., 2011). For NE, they used the same tag-set used in CoNLL-2003 Shared Task (Person, Location, Organization, and Miscellaneous). Since there is no standard training/dev/test data split for xLime corpora, we randomly split it 80:10:10 into training, development and test sets.

6.2. Baselines

We compare the performance of our system to performances of prior works described in section 2.2.:

6.2.1. English

- Derczynski et al. (2013) performed experiments on T-PoS corpus. For training, they used T-train (2.3K tokens), 50K tokens from the WSJ part of the PTB and 32K tokens from the NPS IRC corpus, achieving an accuracy of 88.69% on T-eval. Furthermore, they achieved 90.54% token accuracy using supplementary 1.5M training tokens annotated by vote-constrained bootstrapping.

[2] All corpora are in the CoNLL format, they are already tokenized.

[3] forum.doctissimo.fr

[4] www.jeuxvideo.com

[5] http://nl.ijs.si/janes/wp-content/uploads/2016/09/A-Multilingual-Social-Media-Linguistic-Corpus.html

Language	Domain	Corpus	Task	# Sentences	# Tokens
English	Source	WSJ	POS	67,786	1,2M
	Target	NPS	POS	10,567	45,000
	Target	T-POS	POS	787	15,000
	Target	Ark dataset (Oct27 + Daily547)	POS	1,827 + 547	26,594 + 7,707
French	Source	FTB	POS	21,634	624,187
	Target	French Web 2.0	POS	1,700	20,557
	Target	ExtremeUGC	POS	974	8,099
Spanish	Source	xLime Spanish NER	NER	7,668	140,852
	Target	xLime Spanish POS	POS	7,668	140,852
German	Source	xLime German NER	NER	3,400	60,873
	Target	xLime German POS	POS	3,400	60,873
Italian	Source	xLime Italian NER	NER	8,601	162,269
	Target	xLime Italian POS	POS	8,601	162,269

Table 1: Statistics of the different source and target datasets used in this paper.

- Owoputi et al. (2013) performed experiments on T-Pos, Ark and NPS corpora, achieving 90.40%, 93.20% and 93.4% accuracy respectively.

- Gui et al. (2017) performed experiments on T-PoS, ArK and NPS datasets, achieving 90.92%, 92.80% and 94.1% accuracy respectively. For training, they leverage 1,17M token from unlabeled tweets and more than 1,17M from labeled WSJ. In order to use WSJ labeled data in experiments on ARK dataset, they performed a mapping between PTB and ARK tag-sets.

6.2.2. French

- Nooralahzadeh et al. (2014) proposed a French POS tagging system using a discriminative sequence labeling model (CRF). They achieved 91.9% accuracy on Fr2.0 corpus. The same system setup was evaluated on T-POS and NPS English corpora achieving 90.1% and 92.7% accuracy respectively.

- Alonso et al. (2016) experimented POS tagging on ExtremeUGC dataset using Melt tagger (Denis and Sagot, 2009) with a set of normalization rules, achieving 84.72% accuracy.

6.2.3. Spanish, German and Italian

Rei et al. (2016) reported inter-Annotator Agreement per language on xLime dataset, 88% for German, 87% for Italian and 85% for Spanish.

6.3. Word Embedding

Words embeddings initialization is computed by a look-up table of each of pretrained model. All words are lowercased before passing through the look-up table for conversion to their corresponding vectors.

Multiple sets of published pre-trained vectors are publicly available for English. Experiments in (Meftah et al., 2017) showed that an initialization with a combination of several pre-trained embedding vectors (from different pre-trained models) improves significantly the performances. Therefore, for English experiments, we initialize word embedding with a concatenation of four pre-trained models:

1. Word2vec (Mikolov et al., 2013), trained on part of Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for 3 million words.

2. FastText (Bojanowski et al., 2016), which is very similar to Word2vec (Using SkipGram) but it also uses sub-word information in the prediction model. FastText Facebook embedding is trained on Wikipedia for 294 languages and contains 300-dimensional words vectors.

3. GloVe (Pennington et al., 2014) is a model based on global word-word co-occurrence statistics. We use two Glove's models. The first, which we name "Glove", trained on 42 billions words from a web crawling, contains 300-dimensional vectors for 1.9M words. And the second, which we name "Glove-Twitter", trained on 2 billion tweets, contains 200-dimensional vectors for 1.2M words.

For experiments on French, Spanish, German and Italian, we use FastText 300-dimensional pre-trained embedding vectors trained on Wikipedia.

6.4. Transfer Learning Setup

The first scenario (cross-domain TL) is evaluated on English and French, following three main phases: (1) training the parent network on the source problem on rich out-of-domain data (WSJ for English and FTB for French) (2) transferring weights of the first set of parameters to the target problem. These weights are used to initialize the child model's first set of parameters, rather than starting from a random position[6]. And finally (3) fine-tinning the child network on low-resource in-domain data.

Since we have multiple target datasets for each language, a jointly training is performed in the step of child model's fine-tuning[7].

[6]The weights of the second set of parameters of the child model are randomly initialized.

[7]Using a smaller learning rate for weights that will be fine-tuned (first set of weights), in comparison to the randomly initialized weights (second set of weights) lead to slightly improvements.

Language	English			French		Spanish	German	Italian
Dataset	T-Pos	ARK	NPS	Fr2.0	UGC	xlime		
Acc. without transfer Learning (%)	89.13	91.33	92.9	91.14	87.89	90.87	90.1	89.41
Acc. with transfer learning (%)	**90.90**	**92.01**	**93.2**	**91.99**	**88.07**	**91.03**	**90.33**	**89.66**

Table 2: Our system accuracy (acc.) with and without transfer learning. Cross-domain transfer is performed for English and French and cross-task transfer for Spanish, German and Italian.

Method	**Acc. T-Pos (%)**	**Acc. ARK (%)**	**Acc. NPS (%)**
Derczynski et al. (2013)	88.69	–	–
Owoputi et al. (2013)	90.40	**93.20**	93.4
Nooralahzadeh et al. (2014)	90.1	–	92.7
Gui et al. (2017)	**90.92**	92.8	**94.1**
Our results	*90.90*	*92.01*	*93.2*

Table 3: Our system's performance on English social media datasets compared to state-of-the-art works.

The second scenario (cross-task TL) is evaluated on Spanish, German and Italian, we use TL approach by a jointly training of source and target tasks (NER and POS).

The training procedure for cross-task TL is as follows: At each epoch, we perform training on a batch from both datasets (source and target), and then, we perform the parameters optimization according to the loss function of the given task (The shared parameters are optimized to improve the performances of both tasks. However, each set of task's parameters is optimized only to improve the corresponding task). Training on NER is stopped before the POS tagging in order to preserve more specific features of the POS tagging task.

Mou et al. (2016) showed that the features represented by the lowest layers of neural networks are more general than topmost layers features in NLP applications. And the knowledge to transfer from the parent network to the child network depends on the relatedness of the source and the target tasks and data-sets. For this purpose, we followed the same experiments realized in (Meftah et al., 2017) to study the transferability of each layer of the neural network for each dataset, and to choose the set of layers to transfer from the parent problem to the child problem.

6.5. Training Settings

All experiments described in this section are implemented using the PyTorch library. The hyper-parameters have been chosen using cross-validation on the reported splits (In section 6.1.) for all the results reported in the following section. We use the Adam optimizer in all experiments. We set the character embedding dimension at 30, the dimension of hidden states of the GRUs layer at 100 and fully connected layer (FCL) dimension at 80. We use dropout training before the input to LSTM and FCL layers with a probability in order to avoid overfitting

7. Results and Discussion

7.1. Transfer Learning Performances

In this section, we compare in table 2 performances of the neural network model described in section 4. trained only on target dataset (without TL) against the neural network trained with TL. We can see that the TL method improves results on all languages.

Table 2 further shows that the improvements made by cross-domain TL (English and French results) are more important than improvements made by cross-task TL (Spanish, German and Italian results). This phenomenon can be explained by the fact that the underlying similarities between the source task and the target task are less transferable, hence the improvement is less substantial.

Additionally, an interesting note on French experiments, where the improvement brought by TL is more important on the French social media 2.0 (Fr2.0) dataset (+0.85%) compared to ExtremeUGC dataset (0.28%), that can be explained by the fact that the last dataset is more noisy (high divergence from the source dataset FTB) than Fr2.0.

We can also observe that the improvement brought by TL is more important on the T-Pos dataset (+1.77%) compared to Ark dataset (0.68%), that can be explained by the fact that T-POS dataset have similar tokenization and annotation scheme than the source dataset (PTB) in contrast to Ark dataset.

7.1.1. Cross-task TL performances

In order to understand how cross-task TL improves POS tagging performances on Spanish, German and Italian social media content. In particularly, which POS tags benefit more from transferring knowledge from NER task. Table 6 shows an important improvement on the accuracy of the POS tag "Noun" compared to the overall accuracy.

We provide an example in the table 7, where cross-task TL helps to assign the correct tag to the Spanish word *Internacional* (i.e International in English), tagged as an adjective by the model without TL. Although, the word *Internacional* is an adjective in most cases. However, in this case *Amnistía Internacional* is an organization, and the information brought by NER task helps to solve the ambiguity.

7.2. Comparison with State-of-the-art Results

In tables 3, 4 and 5, we show our system's performances compared to state-of-the-art results. We can see that our results are competitive compared to the state-of-the-art systems.

Method	Acc. Fr2.0 (%)	Acc. UGC (%)
Nooralahzadeh et al. (2014)	91.9	–
Alonso et al. (2016)	–	84.72
Our results	**91.99**	**88.07**

Table 4: Our system's performance on French social media datasets compared to state-of-the-art works.

Method	Acc. Spanish (%)	Acc. German (%)	Acc. Italian (%)
inter-Annotator Agreement	85	88	87
Our results	**91.03**	**90.33**	**89.66**

Table 5: Our system's performance on xLime datasets compared to inter-Annotator Agreement.

Language		Sp	Ger	IT
W/o TL	**Overall acc. (%)**	90.87	90.1	89.41
	Acc. on nouns (%)	*92.54*	*91.98*	*94.12*
W TL	**Overall acc. (%)**	91.03	90.33	89.66
	Acc. on nouns (%)	*97.65*	*98.02*	*98.2*

Table 6: Improvement of the accuracy of the tag "Noun" compared to the improvement of the overall accuracy after using cross-domain transfer learning, on Spanish (Sp), German (Ger) and Italian (IT).

W/o TL	... de/**ADP** Amnistía/**NOUN** Internacional/**ADJ** :/**.** #EEUU/**#** ...
W TL	... de/**ADP** Amnistía/**NOUN** Internacional/**NOUN** :/**.** #EEUU/**#** ...

Table 7: Our model POS tagging example of a Spanish tweet, without TL (W/o TL) in the first line and with TL (W TL) in the second line.

Tables 4 and 5 show that our model outperforms state-of-the-art systems on French, Spanish, German and Italian.
On table 3, we can see that our method outperforms state-of-the-art approaches (Derczynski et al., 2013) and (Owoputi et al., 2013) on T-POS experiments. However, it performs worse than (Owoputi et al., 2013) on ARK-dataset. Our model is end-to-end and the most of errors in our system were caused by hashtags and proper nouns. These issues were resolved in (Owoputi et al., 2013) by adding external knowledge (a list of named entities) and rules to detect hashtags, etc.
We can observe that (Gui et al., 2017) achieves better performances than our model in both datasets, an effective model (Adversarial Neural Network) was used in their work with huge amounts of unlabeled in-domain-data (More than 1.17 millions token) and 1 million token from WSJ. In addition, they used regular expressions to perfectly tag Twitter-specific tags: retweets, @usernames, hashtags, and urls, contrariwise our model which is end-to-end and does not use hand-crafted rules.

7.3. Improving Results Using Pre-processing

As discussed in the above section, most of errors in our system were caused by Twitter specific tokens (e.g. our model accuracy on T-POS dataset on hashtags is equal to 62%). In this section, we normalize Twitter specific tokens, we substitute every word starting with a # character by a special token. Similarly, all words starting with the prefix http are replaced by a url token.
Table 8 illustrates our model performances on English datasets after pre-processing rules for hashtags, urls, usernames and at-mentions. We can see a significant improvement on accuracy on all of English datasets, outperforming state-of-the-art work (Gui et al., 2017) on T-Pos dataset.

Method	Without prep	With prep
Acc. on T-Pos (%)	90.90	**91.03**
Acc. on ARK (%)	92.01	**92.6**
Acc. on NPS (%)	93.2	**93.41**

Table 8: Performances on English social media datasets before (Without prep) and after (With prep) the integration of pre-processing rules for hashtags, Urls, usernames and at-mentions.

8. Conclusion

This paper presented a neural network model using Transfer Learning (TL) for Part-of-speech (POS) tagging of social media texts. Two scenarios of TL were experimented. The first is cross-domain TL, where we leverage available huge amounts of POS-labeled standard English and French to improve English and French social media texts POS tagging. The second scenario is cross-task TL, where we use named entities labeled data to improve POS tagging of Spanish, German and Italian social media texts. Our experiments show that both scenarios of TL improve the performance of the POS tagging task. For future work, we plan to model the similarities and differences between the source and target languages in order to incorporate this external linguistic knowledge in the neural network model. In addition to that, we aim to conduct a study on morphologically rich and complex languages such as Arabic that is well known for its diverse dialects (22 dialects distributed over 5 regional categories) that we can find on social media.

9. Acknowledgments

This research work is supported by the ASGARD project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 700381.

10. Bibliographical References

Abeillé, A., Clément, L., and Toussenel, F. (2003). Building a treebank for french. In *Treebanks*, pages 165–187. Springer.

Alonso, H. M., Seddah, D., and Sagot, B. (2016). From noisy questions to minecraft texts: Annotation challenges in extreme syntax scenarios. *WNUT 2016*, page 13.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Chiu, J. P. and Nichols, E. (2015). Named entity recognition with bidirectional lstm-cnns. *arXiv preprint arXiv:1511.08308*.

Choi, J. D. (2016). Dynamic feature induction: The last gist to the state-of-the-art. In *HLT-NAACL*, pages 271–281.

Denis, P. and Sagot, B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art pos tagging with less human effort. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, volume 1.

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013). Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *RANLP*, pages 198–206.

Forsythand, E. N. and Martell, C. H. (2007). Lexical and discourse analysis of online chat dialog. In *Semantic Computing, 2007. ICSC 2007. International Conference on*, pages 19–26. IEEE.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics.

Gui, T., Zhang, Q., Huang, H., Peng, M., and Huang, X. (2017). Part-of-speech tagging for twitter with adversarial neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2401–2410.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Lee, J. Y., Dernoncourt, F., and Szolovits, P. (2017). Transfer learning for named-entity recognition with neural networks. *arXiv preprint arXiv:1705.06273*.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Meftah, S., Semmar, N., Zennaki, O., and Sadat, F. (2017). Using transfer learning in part-of-speech tagging of english tweets.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mou, L., Meng, Z., Yan, R., Li, G., Xu, Y., Zhang, L., and Jin, Z. (2016). How transferable are neural networks in nlp applications? *arXiv preprint arXiv:1603.06111*.

Nooralahzadeh, F., Brun, C., and Roux, C. (2014). Part of speech tagging for french social media data. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1764–1772.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Rei, L., Mladenic, D., and Krek, S. (2016). A multilingual social media linguistic corpus. In *Proceedings of the 4th Conference on CMC and Social Media Corpora for the Humanities, Ljubljana, Slovenia*.

Ritter, A., Clark, S., Etzioni, O., et al. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534. Association for Computational Linguistics.

Seddah, D., Sagot, B., Candito, M., Mouilleron, V., and Combet, V. (2012). The french social media bank: a treebank of noisy user generated content. In *COLING 2012-24th International Conference on Computational Linguistics*.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics.

Yang, Z., Salakhutdinov, R., and Cohen, W. W. (2017). Transfer learning for sequence tagging with hierarchical recurrent networks. *arXiv preprint arXiv:1703.06345*.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*.

Utilizing Large Twitter Corpora to Create Sentiment Lexica

Valerij Fredriksen, Brage Jahren, Björn Gambäck
Department of Computer Science
Norwegian University of Science and Technology
Trondheim, Norway
valerij92@gmail.com, brage.jahren@gmail.com, gamback@ntnu.no

Abstract

The paper describes an automatic Twitter sentiment lexicon creator and a lexicon-based sentiment analysis system. The lexicon creator is based on a Pointwise Mutual Information approach, utilizing 6.25 million automatically labeled tweets and 103 million unlabeled, with the created lexicon consisting of about 3 000 entries. In a comparison experiment, this lexicon beat a manually annotated lexicon. A sentiment analysis system utilizing the created lexicon, and handling both negation and intensification, produces results almost on par with sophisticated machine learning-based systems, while significantly outperforming those in terms of run-time.

Keywords: Pointwise Mutual Information, Sentiment lexica, Lexicon-based sentiment analysis

1 Introduction

A popular social medium providing opinionated texts is the micro-blogging service Twitter. On Twitter, users can post textual entries of up to 140 characters, commonly called *tweets*. Each day, approximately 500 million new tweets are posted; a fraction of those are made available through Twitter's public API. Large datasets can therefore easily be acquired, making Sentiment Analysis (SA) of tweets particularly popular.

The present study shows that two features in particular stand out when building Twitter sentiment classification systems: the effect of using sentiment lexica in the classification process and the run-time performance. The use of sentiment lexica is shown to be the single most valuable system component in terms of overall system performance. Without using sentiment lexica, the performance dropped as much as 4%. In general, most present-day Twitter specific lexica only contain word unigrams and bigrams. Here, in contrast, the effect of including longer phrases in sentiment lexica is explored, as well as utilizing the long phrases in the classification process, while striving to create a lexicon-based SA system with real time performance. Section 2 gives an overview of the state-of-art in automatic creation of sentiment lexica and lexicon-based SA, while Section 3 describes the datasets used. In Section 4, the overall architecture of both the lexicon-based SA system and the lexicon creator system are detailed. Section 5 includes the tests conducted on the created lexicon and lexicon-based SA system, which are further discussed in Section 6. Finally, Section 7 concludes and outlines future work.

2 Related Work

Lexicon-based sentiment analysis is the task of performing SA solely based on sentiment lexica. The overall performance of a lexicon-based SA system is hence closely related to the quality of the sentiment lexica, but also to *how* the sentiment lexicon information is used. In recent years, most SA systems use a Machine Learning (ML) approach, with sentiment lexica as a feature among others. The number of new lexicon-based SA systems that have appeared is therefore few. Although the task of classification differs between a lexicon-based and an ML-based SA system, the sentiment lexica features in ML systems are grounded in the same ideas as lexicon-based SA systems. The common approach consists of two main steps: sentence analysis and sentiment calculation.

In the first step, each sentence in a tweet is analysed in search of specific features. A state-of-the-art feature set was identified by Mohammad et al. (2013). It includes word and character n-grams, word clusters, part-of-speech tags, the number of emoticons and consecutive punctuation marks, and negation. In the second step, these features are used to adjust raw sentiment values for the n-grams in a sentence that are looked up in Prior Polarity Sentiment Lexica. To calculate the final sentiment value, the sentiments of all n-grams in the sentence are summed up, and adjusted in accordance to any negation and intensification detected during the analysis step.

Prior Polarity Sentiment Lexica can be created in several ways; most commonly by manual annotation. However, manual annotation of large amounts of data is costly. This also applies to the task of annotating sentiment lexica, where each n-gram should be assigned a sentimental strength. Before beginning the annotation process, the n-grams to include in the lexicon are collected, e.g., from a large corpus of texts capturing the language features the lexicon is aimed at. Mohammad et al. (2013) selected all unigrams and bigrams as candidate entries to include in the lexicon, while Velikovich et al. (2010) selected all n-grams up to length 10 before filtering out n-grams based on their frequency and mutual information.

When using a corpus of labeled documents to annotate candidate n-grams, the association measure Pointwise Mutual Information (PMI) can be used. PMI quantifies the information shared between events (Fano, 1961). The mutual information $I(x, y)$ between two events x and y with probabilities $P(x)$ and $P(y)$, and joint probability $P(x, y)$ is defined as:

$$I(x, y) = log_2 \frac{P(x, y)}{P(x)P(y)}$$

If there is an association between x and y, the joint probability will be higher than the product of the individual prob-

Name	Pos	Neut	Neg	Total
2013-train	3283	4175	1290	8748
2013-test	1258	1367	462	3087
2014-test	794	564	151	1509
2015-test	1038	987	365	2390
2016-test	7059	10342	3231	20632

Table 1: Overview of the SemEval datasets as used here

abilities ($P(x, y) > P(x)P(y)$), while it will be equal to the product of the individual probabilities if the events are independent (i.e., $P(x, y) = P(x)P(y) \Rightarrow I(x, y) = 0$).
Turney and Littman (2002) proposed a method for creating sentiment lexica where the sentimental orientation of a word could be calculated from the PMI value of a word w in a positive context minus the value of the same word in a negative context. They used a seed set of positive and negative words, and decided the context of a word by its proximity to a seed word. Mohammad et al. (2013) similarly used a seed set, but one containing positive and negative hashtags and emoticons to create the *Sentiment140* and *HashtagSentiment* lexica.
Another method using unlabeled documents is the graph approach, where similarity between all candidate n-grams, and between those and a seed set of positive and negative words are calculated before each n-gram is initialized as a node in a graph. Two variants have been proposed: In *graph propagation* (Velikovich et al., 2010), the nodes representing the seed set are initialized with a sentiment score according to their orientation, before propagating their sentiment value to the nodes laying on a path of length k away. In *label propagation* (Zhu and Ghahramani, 2002), the sentiment value is calculated as the weighted average of an n-gram's neighbours. Whereas each node in graph propagation only holds the max path from a seed word, the nodes in label propagation can possibly hold multiple paths to a specific seed word. In both variants, the edges between the nodes are weighted according to their similarity.

3 Datasets

To be able to create a sentiment lexicon based on raw tweets, about 400 million tweets were downloaded using the Twitter Streaming API[1] in the period 18/12/2015–19/03/2016 (92 days), with about 4 million tweets downloaded each day. The raw tweets were filtered, removing all tweets that contained "RT @" (retweets), a URL or the ° symbol (automated weather or GPS location services). In addition, tweets ending with a number were removed, since spammers often keep tweeting the same message with an incrementing number at the end, to combat Twitter's spam detection. The resulting filtered dataset contained about 103 million tweets and was used to generate the n-grams.
For comparison, five manually annotated Twitter sentiment dataset from International Workshop on Semantic Evaluation (SemEval) were utilized: the training and test sets from SemEval 2013, and test sets from SemEval 2014, 2015 and 2016 (Nakov et al., 2016). Some of the tweets have been deleted since the datasets were originally created and are no longer available. Table 1 shows the class distributions of the datasets as used in this work. The 2013-train set was used for training, and the four test sets for testing only.
To be able to use the PMI approach when creating a sentiment lexicon, a large dataset of labeled tweets was also needed. For the approach to produce good and reliable results, the amount of labeled tweets need to be much higher than the tweets made available through SemEval. Hence our lexicon based sentiment analysis system (Section 4.3) was used together with the manually annotated sentiment lexicon *AFINN* (Nielsen, 2011) to automatically label the tweets. For the labeling to be as accurate as possible, only tweets with absolute sentiment score above a certain value were extracted. The value chosen was found using grid search and is a trade-off between precision and recall. Based on the 103 million unlabeled tweets, the labeling process yielded a labeled dataset containing 6.25 million classified tweets, of which 58.7% were labeled as positive and the rest as negative. The overall dataset creation process is shown in Figure 1.

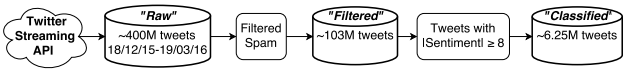


Figure 1: Datasets created using Twitter Streaming API

4 Architecture

A PMI-based approach was used to build a system for automatic creation of a sentiment lexicon, and a lexicon based classifier was developed, utilizing the features of the lexicon created. Both systems use the same tweet preprocessing and vocabulary tokenization components.
The preprocessing includes handling of word elongation (e.g., 'goddd') using Levenshtein distance. The process consists of two parts: a four step dictionary creation part inspired by Brody and Diakopoulos (2011) and a word correction part using the created dictionary. The word is first reduced to its condensed form, before being looked up in the dictionary. If found, the dictionary returns the most likely spellings of the initial word. The Levenshtein distance is then calculated between the initial word and each of the returned words, before correcting the initial word to the closest match. This way, with a dictionary containing both 'good' and 'god', 'goddd' is 2 deletions away from 'god' or 2 deletions and 1 addition from 'good', so will be reduced to 'god'.
Tokenization consists of splitting sentences into the longest non-overlapping n-grams also found in a given vocabulary. n-grams not found in the provided vocabulary are tokenized as unigrams. The reason for keeping the words not present in the vocabulary is that these words may be intensifiers or negators, which are excluded from the created lexicon.

4.1 PMI Lexicon

The PMI lexicon builder consists of vocabulary identification, counting vocabulary occurrences in a polarized dataset, and sentiment calculation as illustrated in Figure 2.

[1] https://dev.twitter.com/streaming/reference/post/statuses/filter

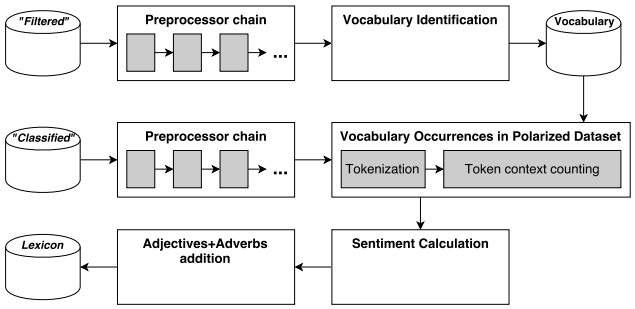


Figure 2: PMI lexicon creation architecture

The vocabulary identification step extracts and selects candidate PMI n-grams for a context vocabulary based on the large unlabeled, filtered dataset. The process of selection uses PMI n-grams (and is thus quite different from the graph propagation's strictly n-gram frequency based selection described below). For an n-gram with $n > 1$ to be selected, the PMI of the included words needs to be higher than a predefined threshold. This way only n-grams containing words that together mean something or form a common phrase are selected. In addition, n-grams ending on a stopword or containing intensifiers are filtered out. Unigrams are not selected as candidate entries, but are introduced later to the system.

In order to calculate sentiment values of n-grams in the context vocabulary, the automatically labeled (classified) dataset is used. The tokenizer splits the entries into the longest possible non-overlapping n-grams. Each individual token holds counters for occurrences in positive and negative contexts. For each n-gram in the context vocabulary, a sentiment value is calculated as $log_2 \frac{freq(w,pos) \cdot freq(neg)}{freq(w,neg) \cdot freq(pos)}$, using the number of times the n-gram occurs in positive tweets and in negative tweets. The lexicon is created by adding all n-grams with absolute sentiment value above a defined threshold and an occurrence frequency in the labeled dataset above a set frequency.

With the created lexicon, all unigrams are run through an adjective and adverb addition algorithm, adding all missing adjective and adverb forms of the unigram to increase the coverage of the lexicon. The missing adverbs and adjective forms are derived based on a set of rules for forming comparatives and superlatives,[2] and for forming adverbs from adjectives.[3] These are added to the lexicon only if they were previously encountered in the tokenization process and assigned the same sentiment value as their related n-gram. The resulting lexicon forms the final PMI lexicon.

4.2 Graph Propagation Lexicon

For comparison, a lexicon was also created using graph propagation. The implementation of the graph propagation approach consists of four steps: vocabulary identification, context-vector creation, graph creation, and sentiment propagation. During the vocabulary identification step a set of candidate n-grams are identified and selected, forming a context vocabulary. Each tweet is processed into all possible n-grams with length up to $n = 5$, where candidate n-grams are selected based on their occurrence frequency in the large unlabeled tweet dataset. All n-grams below a frequency threshold, in addition to all n-grams ending with a stopword or containing an intensifier are filtered out.

To compare candidate n-grams, their context vectors are created using the COALS method (Rohde et al., 2006), i.e., by summing up the word frequencies of x words before and x words after the n-gram, over all mentions of the n-gram in the dataset, weighted by the distance from the n-gram using a ramped window of size 6 (words next to the n-gram on either side increase the frequency of that word in the context vector by 6 for that side, etc.). A matrix containing all n-grams is created, with rows containing the context vectors of all selected n-grams and columns containing the occurrence frequencies of words occurring in the vectors. The values are normalized using Pearson Correlation, and all negative values are then set to 0, while all positive values are squared.

Given the candidate entries and their context vectors, the graph is created, with nodes representing the candidate entries, and edges with weights representing node similarity. To create the edges, cosine similarity is calculated between all n-gram pairs, represented by their context vectors. As in Velikovich et al. (2010), an edge is created between two nodes if their similarity is greater then a set threshold. The seed nodes then propagate their sentiment values through the finished graph. In our implementation, each seed node can affect nodes that are connected to it via two or less other nodes. When all seed nodes have propagated their sentiment value, the final sentiment value of each node is calculated by subtracting the sum of all negative max paths from the sum of all positive max paths. The negative and positive max paths to a node are the maximum sentiment values each connected seed node affects the node with. A node with more paths to positive than negative seed nodes will most likely get a positive sentiment value. Finally, the lexicon is created by extracting all the n-grams and their sentiment values.

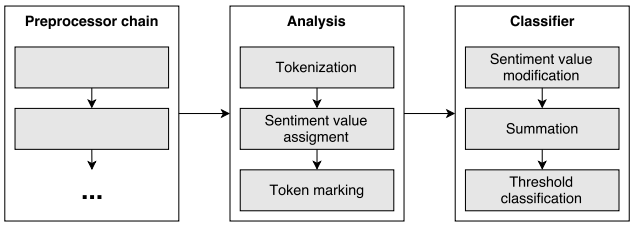


Figure 3: Lexicon classifier system architecture

4.3 Lexicon Based Sentiment Analysis

The lexicon based Sentiment Analysis system accepts single tweets or a set of tweets, and outputs a predicted classification per tweet. The predicted classification is determined by running each tweet through three main stages: preprocessing, analysis and classification, as shown in Figure 3.

The analysis consists of detecting negation cues, intensifiers words, and punctuation marks, and assigning sentiment values. This is done by first applying preprocessing

[2] http://www.eflnet.com/tutorials/adjcompsup.php

[3] http://www.edufind.com/english-grammar/forming-adverbs-adjectives/

System	2013	2014	2015	2016
Graph	0.4942	0.4844	0.4571	0.4744
PMI	0.6130	0.6170	0.5711	0.5685

Table 2: Graph approach vs. PMI approach

Lexicon	2013	2014	2015	2016
Sentiment140	0.5260	0.5415	0.4788	0.5186
AFINN	0.6259	0.6009	0.5801	0.5882
PMI	0.6748	0.6578	0.6201	0.6032

Table 3: F_1 scores of different lexica

and tokenization, splitting the tweet into tokens that are looked up in the provided sentiment lexicon and assigned the lexicon value, if found, or a sentiment value of zero, if not found in the lexicon. If a token matches a negation cue (*'not'*, *'wouldn't'*, etc.), a simple method proposed by Das et al. (2001) is used, whereby n consecutive words appearing after the negation cue are marked as negated. If a token matches an intensifier, the next token is marked as intensified. Finally, a sentence final '!' or '?' marks all tokens in the sentence as intensified. The sentiment value of each token found in the lexicon is then calculated as $(L \cdot I) - N$, where L is the token's lexicon value, I the intensification value ($I = 1$ if not intensified), and N the negation value.
The value of I for a token is dependent on what intensifier word or punctuation mark it is affected by. Some intensifier words such as 'kind of' or 'hardly' will work as a dampeners with I-values between 0 and 1. Words like 'incredibly' or 'extremely' will on the other hand work as boosters with I-values above 1. A token can be intensified by both an intensifier and a punctuation mark. In such case, I is the product of the intensification constant of the intensifier and the intensification constant of the punctuation mark. The I-values of '!' and '?', and the N value along with their influence ranges are determined through grid search.
The final sentiment score of a tweet is calculated as the sum of its tokens' sentiment values, and it is classified as either positive, negative or neutral by comparing its sentiment value against two thresholds that set the lower and upper bounds for tweets to be classified as neutral. The thresholds are also determined through grid search.

5 Experiments

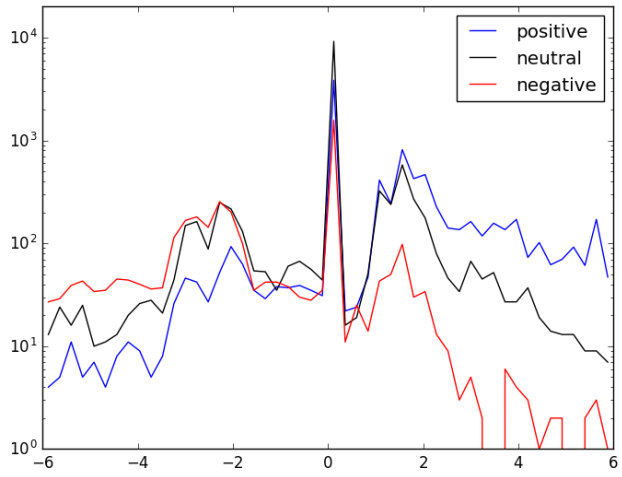
A series of experiments were conducted, both to determine how well the lexicon based classifier, utilizing the PMI lexicon, fares against more complex classifiers, and to investigate the effect of using its output as a feature in an ML classifier. For the system to perform as well as possible, the optimal parameters for both the lexicon creator and the classifier need to be identified. Rather than a complete grid search where all combinations of parameter values are explored, the search consisted of a series of iterative steps, with each step testing all predefined values of a single parameter, while keeping all the other variables constant.
The following parameters were optimised for the inclusion of an n-gram in the vocabulary: maximum length and minimum occurrence frequency; and for inclusion in the lexicon: minimum occurrence in polarized dataset, and minimum PMI and absolute sentiment values. For the lexicon classifier, the optimised parameters were: the negation constant N (Section 4.3) and the number of tokens after a negation cue to include in the scope, the intensification constants for tokens inside sentences ending with '!' resp. '?', and the thresholds between *neutral* and *positive* resp. *negative* sentiment classification.

5.1 Graph-based Approach vs. PMI-based

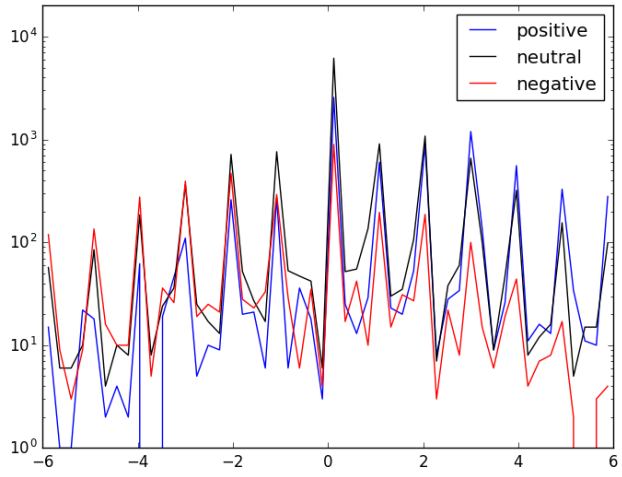
Applying the graph propagation to lexicon creation encountered several problems. First with creating the seed set: a small seed produced smaller lexica with weaker sentiment values since fewer words were propagating the values, while a large seed set had problems setting good starting sentiment values. Second with creating the context vector: similarity between n-grams is calculated as similarity between their context vectors, but the relationship between mutli-grams and their neighbouring words is less clear. A third problem is that calculating the cosine similarity for each word with every other word makes the computational complexity exponential.
Most importantly, the PMI approach clearly outperformed graph propagation when tested on the 2013–2016 SemEval data, as shown in Table 2.

5.2 Lexicon Comparison

Table 3 compares the PMI lexicon against other previously created sentiment lexica. The *Sentiment140* lexicon (Mohammad et al., 2013) was also created using a PMI approach, making it an obvious choice for comparison, in addition to the *AFINN* lexicon (Nielsen, 2011) used in the creation of our *Labeled* dataset. The comparison was done by running our lexicon based classifier on the four SemEval datasets, using the PMI lexicon, *Sentiment140*, and *AFINN* as lexicon, respectively. As can be seen in the table, the PMI lexicon outperforms the other two sentiment lexica. *AFINN* is the closest, with the smallest difference of approximately 0.015 on the 2016 dataset.
Although the results seem to point to the PMI lexicon being the best, it should be noted that this lexicon was specifically tailored to work with the classifier. *AFINN* should not be affected by this, since it only contains words without special characters and has a single sentiment value per lexicon entry similar to the PMI lexicon. For *Sentiment140* on the other hand, the classifier is not able to utilize all features the lexicon provides: negation is handled differently and the preprocessor removes all non-alphanumerical characters except characters forming emoticons, so there are *Sentiment140* entries the classifier never is able to use. Hence no clear conclusion can be drawn for *Sentiment140*. We do, however, stipulate that the method of creating a labeled dataset for the lexicon creator is superior to the hashtag and emoticon approach used to create *Sentiment140*, and that the use of larger n-grams benefits the PMI lexicon.
Since the comparison between the PMI lexicon and *AFINN* is the most reasonable, Figure 4 depicts the distribution of predicted positive, negative and neutral tweets. In Figure 4a



(a) PMI lexicon



(b) AFINN lexicon

Figure 4: Sentiment value histograms

we can see a clear distinction between the classes, while in Figure 4b, the classes are much closer and overlapping. The zig-zagged pattern in Figure 4b is due to *AFINN* lexicon only using integer sentiment values. The separability of the different classes displayed in the graphs shows why the PMI lexicon will generally classify more tweets correctly.

5.3 System Performance

To test the overall system performance, the lexicon based classifier, utilizing the PMI lexicon, was compared against a baseline system (Jahren et al., 2016) and VADER, Valence Aware Dictionary and sEntiment Reasoner (Hutto and Gilbert, 2014), a lexicon based sentiment analysis tool specifically tuned towards social media. VADER utilizes a Support Vector Machine (SVM) classifier and goes beyond bag-of-words by taking into consideration word order and degree modifiers.

The baseline system was created as a state-of-the-art Twitter Sentiment Analysis (TSA) system based on the most common approaches within the field. It was implemented in the Scikit-Learn (Pedregosa et al., 2011) machine learning framework, also using an SVM classifier, but trained on word and character n-grams, word clusters, part-of-speech tags, emoticons, punctuation, and sentiment information from VADER, from the automatically annotated lexica *Sentiment140* and *HashtagSentiment* (Mohammad et al., 2013), and from the manually annotated lexica *MPQA* (Wilson et al., 2005), *BingLiu* (Hu and Liu, 2004), *AFINN*, and *NRC Emoticon* (Mohammad et al., 2013). Note that this is a strong baseline: the system achieved the second highest tweet-level accuracy in SemEval'16 Task 4.

	System	Prec	Rec	F_1	Time
2013	VADER	0.6540	0.6573	0.6508	1.29
	Baseline	0.7209	0.7120	0.7073	74.01
	Lexicon	0.6866	0.6803	0.6748	0.04
2014	VADER	0.6536	0.6421	0.6441	0.63
	Baseline	0.7091	0.6832	0.6847	41.38
	Lexicon	0.6818	0.6554	0.6578	0.02
2015	VADER	0.6310	0.6201	0.6197	0.99
	Baseline	0.6862	0.6548	0.6527	64.80
	Lexicon	0.6483	0.6213	0.6201	0.03
2016	VADER	0.5939	0.5919	0.5928	8.63
	Baseline	0.6461	0.6434	0.6431	538.88
	Lexicon	0.6085	0.6028	0.6032	0.19

Table 4: Sentiment classifier performance

System	2013	2014	2015	2016
Baseline	0.7073	0.6847	0.6527	0.6431
+ Lexicon	0.7216	0.6911	0.6586	0.6356

Table 5: Baseline system with PMI lexicon

All systems were trained on the SemEval 2013 training data, and their performance compared across four SemEval test datasets is shown in Table 4. The *VADER Sentiment* system scores below the lexicon based classifier across all performance measures on all four datasets, while the state-of-the-art baseline system consistently scores above it. However, the lexicon based system significantly outperforms the other systems when it comes to run-time. Even on the smallest dataset (2014), it executes approximately 31.5 times faster than VADER and approximately 2 000 times faster than the baseline system. With an execution time of 0.19 seconds on the 2016 dataset, it classifies tweets with a speed of 108 600 tweets per second, against the speed of the baseline system with 38 tweets per second.

Experiments were also run using the lexicon based classifier as a feature in the baseline system. Table 5 shows a clear increase in performance in terms of F_1 score from 0.7073 to 0.7216 on the 2013 dataset when adding the lexicon based classifier as a feature. There is also an increase on the 2014 and 2015 datasets, although not as apparent. On the 2016 dataset on the other hand, the performance actually drops, which might be due to that dataset containing more noise and annotation errors than the other datasets.

Size	2013	2014	2015	2016
500	0.3387	0.2736	0.3029	0.3936
1 500	0.6533	0.6343	0.6049	0.6016
3 000	0.6748	0.6578	0.6201	0.6032
10 000	0.6674	0.6507	0.6201	0.6038
25 000	0.6328	0.6203	0.6020	0.6011
50 000	0.6052	0.6092	0.5650	0.5976
100 000	0.6040	0.6090	0.5648	0.5851
200 000	0.5965	0.6020	0.5626	0.5938

Table 6: Comparison of different sized PMI lexica

Size	NZS	TwM	WiL	FWiL
500	0.0435	0.0066	0.3567	0.0081
1 500	0.3902	0.0086	0.3793	0.0655
3 000	0.4726	0.0412	0.3632	0.0826
10 000	0.4922	0.0702	0.3647	0.0846
25 000	0.8421	0.1788	0.4012	0.1712
50 000	0.9968	0.3322	0.5229	0.4371
100 000	0.9992	0.4703	0.6122	0.7019
200 000	0.9996	0.6012	0.6692	0.9889

Table 7: Lexicon size vs. coverage

5.4 Lexicon Size Comparison

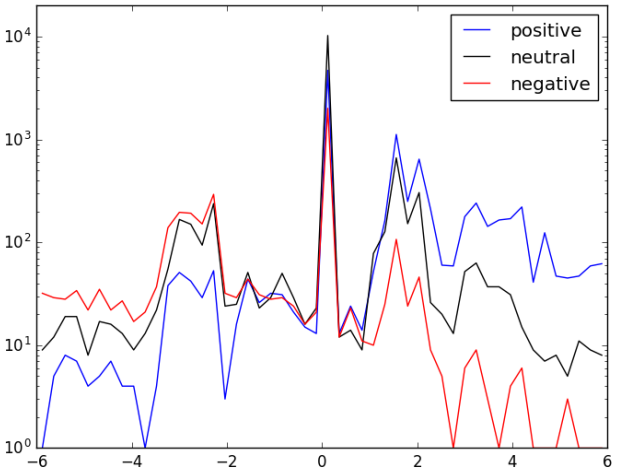
To explore how the size of the created sentiment lexicon affects the performance, eight sentiment lexica of sizes ranging from 500 to 200 000 entries were created and tested on the four SemEval datasets. As we can see from Table 6, as long as the lexicon size is above a certain threshold, the overall performance remains acceptable. However, there is a continuous drop in performance when the lexicon size passes 10 000 entries. The best performing lexicon from the above test has 3 000 entries, which is quite interesting looking at the *Sentiment140* lexicon with approximately 300 000 entries in comparison. More words and phrases do not necessary lead to better results.

In addition to the F_1-score, the difference in coverage between the different sized lexica was explored. Four different coverage measures were used:

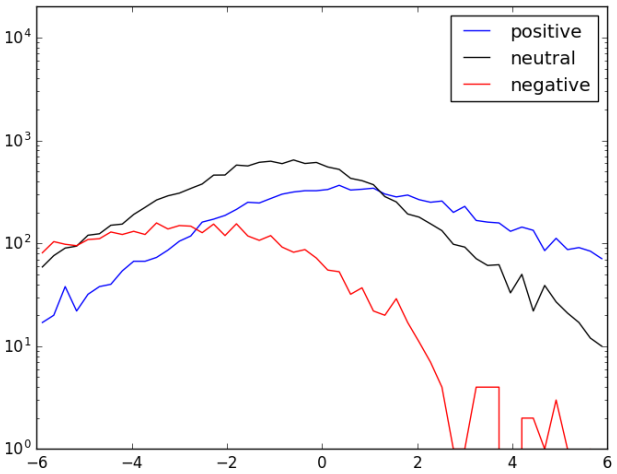
- **NZS** (Tweets with Non-Zero Sentiment): Ratio of tweets where final predicted sentiment value was $\neq 0$.
- **TwM** (Tweets with Multi-grams): Ratio of tweets that included at least one multi-gram from the lexicon.
- **WiL** (Words in Lexicon): Ratio of words in tweets found in the lexicon.
- **FWiL** (Frequent Words in Lexicon): Ratio of top 1 000 most common, non-stop-word words found in the lexicon.

Table 7 shows that the best performing lexicon with a size of 3 000 only contains words found in 47% of the tweets. That means that almost half of the tweets end up with a sentiment score of zero and are classified as neutral. In addition, only approximately 4% of the tweets contain multi-grams (n-grams with $n > 2$) found in that lexicon. Although the scores for the different coverage measures increase with the lexicon size, meaning larger lexicon lead to higher coverage, it does not look like higher coverage means better classification performance.

Figure 5 shows the distribution of positive, negative and neutral tweets given their sentiment value predicted by the classifier on a PMI lexicon of size 1 500 and of size 200 000. In comparison to Figure 4a, showing the same distribution for the best PMI lexicon, the difference in performance is even further visualized. Compared to the distribution of our best PMI lexicon, the distribution of the 200 000 lexicon, shown in Figure 5b is almost a uniform distribution, the large spike around 0 is gone, but the differences between classes are smaller. The other extreme is the distribution of the small lexicon, shown in Figure 5a. It classifies some of the positive and negative tweets well, but most tweets are left with a score of 0.



(a) Size 1 500 PMI lexicon



(b) Size 200 000 PMI lexicon

Figure 5: Lexicon size sentiment value histograms

6 Discussion

After identifying the best performing PMI lexicon, a few statistics were gathered in order to see whether or not the resulting lexicon contained the features we expected. It also provides a general insight into the lexicon created. From Table 8, we can clearly identify all of the top 10 positive lexicon entries as actual positive phrases and the top 10 negative as negative phrases. In addition we do not see any uni-

Positive		Negative	
n-gram	Value	n-gram	Value
you have a great day	5.00	a bad bitch	-5.00
you have an amazing day	4.48	bitch ass nigga	-4.95
hope its a good one	4.33	dumb ass	-4.82
happy birthday i hope	4.21	fuck that bitch	-4.75
you had a great day	4.10	fuck a bitch	-4.75
you have a good day	4.08	bad right now	-4.73
you have a wonderful day	4.02	weak ass	-4.72
hope you have a great	4.01	fuck this shit	-4.68
you have a good one	3.99	fuck fuck fuck	-4.66
i love love love	3.96	fool me twice	-4.63

Table 8: Top 10 positive and negative entries

Positive			Negative		
Emoji	Alias	Value	Emoji	Alias	Value
🎂	Birthday Cake	2.76	😡	Pouting Face	-2.64
🎁	Wrapped Present	2.60	🅿	Parking Sign	-2.41
🎈	Balloon	2.51	😠	Angry Face	-2.41
🎊	Confetti Ball	2.47	😤	Triumph Face	-2.40
🎉	Party Popper	2.46	🚮	Litterbox	-2.34

Table 9: Top 5 positive and negative emojis

grams, meaning that the scoring of longer n-grams higher than shorter n-grams seems to be working. The detected emojis were treated the same way as the phrases, as seen in Table 9. All emojis listed as positive are commonly used to further express positive sentiment in positive sentences, while all of the emojis listed as negative are commonly used in negative sentences.

To detect the impact each feature of the lexicon creator and classifier imposes on the performance, an ablation study was conducted. Table 10 shows that the impact of each feature is very subtle, with the single most important feature being the adjective and adverb addition which increases the F_1-score on the 2013 dataset by 0.01. This means that the classifier performance almost entirely depends on the quality of the PMI lexicon, with the additional features only contributing a small amount.

7 Conclusion and Future Work

The paper has shown that the Pointwise Mutual Information (PMI) lexicon creation approach works well: Lexicon comparison experiments showed a fully automatically created PMI lexicon beating the manually annotated *AFINN* lexicon on all datasets and across all performance measures. In addition to verifying the quality of the lexicon, this proves that creating sentiment lexica automatically is highly viable. While previous work building PMI lexica have been based on rather small sets of tweets, the present work is based on over 100 million tweets, while still being highly efficient.

However, for a classifier to utilize a sentiment lexicon's full potential, the classifier must be specifically tailored to work with that specific lexicon. This is a consequence of the different sentiment lexica creation methods, where the

Features	2013	2014	2015	2016
All	0.6748	0.6578	0.6201	0.6032
- Adj/Adv	0.6653	0.6525	0.6127	0.6032
- Negation	0.6732	0.6561	0.6180	0.6030
- Intensification	0.6733	0.6548	0.6193	0.6036
- PMI n-grams	0.6714	0.6503	0.6161	0.6010

Table 10: Feature ablation results (F_1-scores)

creators of the different available lexica apply different features specifically meant to work well in another system or classifier. This is more important than lexicon size: Larger lexica did lead to a better coverage as shown in Table 7, but classification performance was not improved accordingly. Although the results point to that larger lexica with high coverage would not perform better than relatively small lexica with medium coverage, no definite conclusions can be drawn. The fact that larger lexica created with the lexicon creator did not lead to better performance might also be caused by the quality of the labeled dataset used.

Regarding the run-time performance of lexicon based Sentiment Analysis (SA) systems compared to more sophisticated SA systems, the results clearly suggest that lexicon based systems are the most viable SA systems to use in real-time classification applications. The lexicon based classifier achieves a classification speed of 108 600 tweets per second, meaning that our system could have classified all of the 500 million tweets posted on Twitter each day in real-time 19 times over. With this result it would be possible to add more advanced features to the classifier, trading off run-time performance for better classification performance and still classify fast enough to be a real-time classifier.

The source of most misclassifications are the tweets with sentiment value of 0. In the current implementation, all these tweets are classified as neutral by default. It should be possible to extract several other values for each n-gram to the lexicon that can actually help classify a tweet as neutral instead of classifying all tweets with sentiment value of 0 as neutral. For example, each word could have a sentiment score and an objectivity score. The sentiment score would be calculated as here, while the objectivity score would be calculated also using the PMI approach, but on a dataset labeled as subjective/objective.

Where the PMI lexicon creation approach is only concerned with finding the sentiment values of n-grams, the graph propagation approach is most concerned with the relationship between the different n-grams and strives to find n-grams used in similar contexts. Since one of the problems of the graph propagation approach is to find a good seed set with appropriate weights, it could be interesting to explore a lexicon creation combining the PMI and graph propagation approaches. The PMI-approach could be used to identify a seed set with sentiment values, while the graph propagation could be used to find n-grams similar to the ones already present in the seed set. With more appropriate seed set values, the most similar n-grams found during graph propagation would tentatively be assigned more appropriate sentiment values.

8 Bibliographical References

Brody, S. and Diakopoulos, N. (2011). Cooooooooooooooollllllllllllll!!!!!!!!!!!!!!: using word lengthening to detect sentiment in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 562–570. ACL.

Das, S. R., Chen, M. Y., Agarwal, T. V., Brooks, C., shee Chan, Y., Gibson, D., Leinweber, D., Martinez-jerez, A., Raghubir, P., Rajagopalan, S., Ranade, A., Rubinstein, M., and Tufano, P. (2001). Yahoo! for Amazon: Sentiment extraction from small talk on the web. In *8th Asia Pacific Finance Association Annual Conference*.

Fano, R. M. (1961). *Transmission of Information: A Statistical Theory of Communications*. MIT Press Classics. M.I.T. Press.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177.

Hutto, C. and Gilbert, E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. In *8th International Conference on Weblogs and Social Media (ICWSM-14)*.

Jahren, B. E., Fredriksen, V., Gambäck, B., and Bungum, L. (2016). NTNUSentEval at SemEval 2016 Task 4: Combining general classifiers for fast Twitter sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, pages 103–108, San Diego, California. ACL.

Mohammad, S., Kiritchenko, S., and Zhu, X. (2013). NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In *Proceedings of the 7th International Workshop on Semantic Evaluation Exercises (SemEval-2013)*, Atlanta, Georgia, USA.

Nakov, P., Ritter, A., Rosenthal, S., Stoyanov, V., and Sebastiani, F. (2016). SemEval-2016 task 4: Sentiment analysis in Twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, San Diego, California, June. ACL.

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages*, volume 718, pages 93–98, Heraklion, Crete, Greece.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rohde, D. L. T., Gonnerman, L. M., and Plaut, D. C. (2006). An improved model of semantic similarity based on lexical co-occurence. *Communications of the ACM*, 8:627–633.

Turney, P. and Littman, M. (2002). Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus. Technical report NRC technical report ERB-1094, Institute for Information Technology, National Research Council Canada.

Velikovich, L., Blair-Goldensohn, S., Hannan, K., and McDonald, R. (2010). The Viability of Web-derived Polarity Lexicons. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 777–785, Stroudsburg, PA, USA. ACL.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354.

Zhu, X. and Ghahramani, Z. (2002). Learning from Labeled and Unlabeled Data with Label Propagation. Technical report, Citeseer.

The Nautilus Speaker Characterization Corpus: Speech Recordings and Labels of Speaker Characteristics and Voice Descriptions

Laura Fernández Gallardo, Benjamin Weiss
Quality and Usability Lab, Technische Universität Berlin, Germany
Ernst-Reuter-Platz 7, 10587 Berlin, Germany
{laura.fernandezgallardo, benjamin.weiss}@tu-berlin.de

Abstract

The Nautilus Speaker Characterization corpus is presented. It comprises conversational microphone speech recordings from 300 German speakers (126 males and 174 females) made in 2016/2017 in the acoustically-isolated room *Nautilus* of the Quality and Usability Lab of the Technische Universität Berlin, Germany. Four scripted and four semi-spontaneous dialogs were elicited from the speakers, simulating telephone call inquiries. Additionally, other spontaneous neutral and emotional speech utterances and questions were produced. Interactions between speakers and their interlocutor (who also conducted the recording session) are provided in separate mono files, accompanied by timestamps and tags that define the speaker's turns. One of the recorded semi-spontanous dialogs has been labeled by external assessors on 34 interpersonal speaker characteristics for each speaker, employing continous sliders. Additionally, 20 selected speakers have been labeled on 34 naïve voice descriptions. The corpus labels permit to investigate the speech features that contribute to human perceptions and automatic recognition of speaker social characteristics and interpersonal traits.

Keywords: speech resource, corpus design, speech recordings, speech labeling

1. Motivation

This paper presents the Nautilus Speaker Characterization (NSC) Corpus[1], a new language resource that has been recently collected and labeled for the study of speakers' interpersonal characteristics. More specifically, we investigate the correspondence between acoustic parameters and speaker social characteristics and interpersonal traits, such as confidence, competence, and vocal attractiveness.

The NSC corpus has been designed to study how the subjective perception and automatic recognition of the speakers' social traits are affected by different degradations introduced by voice channel transmissions. Most of existing publicly available databases present speech segments that have already been transmitted/distorted through telephone channels and hence cannot be employed for our analyses. Their sampling frequency is not sufficient for the evaluation of super-wideband (SWB, 50–14,000 Hz) conditions, or the speech was recorded in noisy or uncontrolled environments. A detailed review is provided in (Fernández Gallardo, 2016). In contrast, the newly acquired NSC corpus presents:

- Clean speech recordings made in the acoustically-isolated *Nautilus* room (which gives name to this database) with the high-quality AKG C 414B-XLS microphone. The format of the recordings is audio/wav, 48 kHz, 16 bit, 1-channel.
- Speech from 300 (126 m, 174 f) native German speakers without marked regional dialect, aged 18 to 35 years.
- Human-human conversational speech and interactions. Recordings of four scripted dialogs, four semi-spontaneous dialogs, and spontaneous neutral and emotional speech statements and questions.
- Speakers' demographic information and self-assessed personality.
- For the 300 speakers, externally-assessed continuous numeric labels of 34 interpersonal speaker characteristics on one of the semi-spontaneous dialogs. For 20 selected speakers, also continuous numeric labels of 34 naïve voice descriptions for the same speech material.

Recording sessions of about 45 minutes have been conducted for each speaker individually by a recording assistant, who acted as interlocutor. When the 300 sessions were completed, we proceeded to collect labels given by external raters listening to semi-spontaneous dialog turns. This speech was evaluated in terms of interpersonal speaker characteristics (SC), such as likable, attractive, competent, childish, etc. An average of 15 external evaluators rated each speaker. They employed a 34-item semantic differential questionnaire, which we will refer to as SC-Questionnaire (Fernández Gallardo and Weiss, 2017a).

According to the two major dimensions determined by factor analysis on the SC ratings (*warmth* and *attractiveness*), a set of 20 "extreme" speakers has been selected to study their voice peculiarities. These have been evaluated by 26 listeners who completed the voice descriptions (VD)-Questionnaire, also a 34-item semantic differential rating scale, for each of the 20 voices.

Our long-term aim is to examine acoustic correlates of subjective speaker attributions and the influence of transmission channels for understanding human communication and behavior. Furthermore, being able to automatically predict

[1]The ISLRN of this corpus is 157-037-166-491-1. The data has been made available at the CLARIN repository: `hdl.handle.net/11022/1009-0000-0007-C05F-6` under the CLARIN ACA+BY+NC+NORED license (freely available for scientific research).

speakers' traits from speech features may assist the development of human-machine conversational systems, which could adapt to the detected user's attributes (Burkhardt et al., 2007; Berg, 2014).

The remainder of this paper is as follows. Section 2 describes the recorded speech and their tags, while Section 3 specifies the speech recording setup. Section 4 is devoted to present the collection of database labels and the factor analyses conducted for determining perceptual factors of speaker characteristics and of voice descriptions. This paper concludes with Section 5, which discusses possibilities of using the described NSC resource.

2. Speech Material

During the recordings, the speaker sat in the acoustically isolated room *Nautilus*, and the interlocutor in the office room *Belafonte* of the Quality and Usability Lab of the Technische Universität Berlin, Germany. Two females undertook the task of the interlocutor for 279 and for 21 sessions, respectively, and they also participated in other sessions as speakers.

Scripted and semi-spontaneous dialogs were held between the speaker and the interlocutor. In addition, spontaneous speech was recorded, which corresponds to neutral and emotional speech and questions that are not part of any proposed dialog and were casually uttered during the recording session when speaker and interlocutor interacted (open microphone setting).

For the scripted dialogs, the speakers were asked to read given dialog turns (scripts) as naturally as possible and maintaining the wording. Differently, for the semi-spontaneous dialogs, the speaker and the interlocutor followed a given conversational scenario, which had been extracted and adapted from the Short Conversation Tests of (ITU-T Recommendation P.805, 2007). Table 1 shows the topics of the recorded dialogs, which simulated telephone calls involving some inquiries. The speaker assumed the client's role, while the interlocutor played the role of a contact person or agent.

Tag	Description
1 (a,b,c,d)	Dialog 1: health insurance inquiry
2 (a,b,c)	Dialog 2: mobile phone rate plan inquiry
3 (a,b,c,d)	Dialog 3: car rental inquiry
4 (a,b,c,d)	Dialog 4: real state agency inquiry
5	Dialog 5: car rental booking
6	Dialog 6: pizza order
7	Dialog 7: book from the library
8	Dialog 8: doctor's appointment
d	(repeated) semi-spontaneous dialog turn
s	neutral spontaneous speech
e	spontaneous emotional excerpt
q	spontaneous question
f	spontaneous short feedback

Table 1: Topics of the recorded scripted dialogs (first block), semi-spontaneous dialogs (second block), recorded spontaneous events (third block), and their tags.

During the recording session, speakers spontaneously manifested some natural emotions (e.g. amusement, excitement, frustration of needing many turn repetitions), uttered questions, mentioned something related to the recording tasks, or provided short feedbacks (e.g. "aha", "ok", etc.) to indicate their understanding of the interlocutor's instructions. All events were recorded as well as the interlocutors' speech. After voice activity detection for the delimitation of speech segments, these were tagged with the dialog turn they correspond to, or as belonging to any of the mentioned events, as indicated in Table 1. We refer as *Interaction* to the three files provided together:

- one audio/wav file corresponding to the speaker's speech,
- one audio/wav file corresponding to the interlocutor's speech, and
- a csv file containing the tags and timestamps of the speaker segments.

These *Interactions* are provided for the semi-spontaneous and for the spontaneous speech.

Additional details on the NSC data, as well as the instructions given to the speakers and their consent form, dialog scripts and scenarios, etc. can be found in the NSC documentation.

3. Recoding Setup

As previously mentioned, the recruited speakers performed the recordings in the acoustically-isolated room *Nautilus*. The room's dimensions are 2.75 m x 2.53 m x 2.10 m, and RT60 = 0.08 s at 2 kHz. The approximate distance from the microphone to the speaker's mouth was 35 cm. The interlocutor sat in the adjacent room *Belafonte*, subject to background noises. She listened to the speaker, gave the pertinent instructions, and acted as dialog partner by using the headset Sennheiser HMD 46.

The hardware connections between the speaker's and the interlocutor's room are depicted in Figure 1. The speech signals were recorded using the software Cubase 4 with 48 kHz sampling frequency and 32-bit quantization.

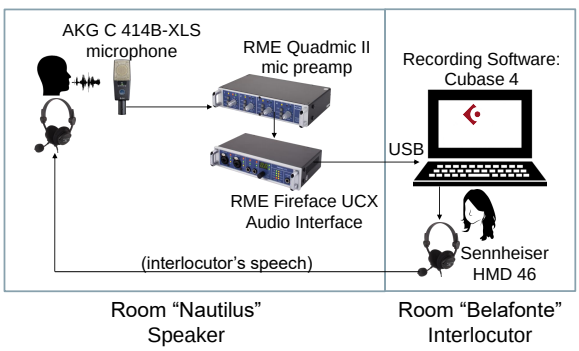


Figure 1: Diagram of device connections for the speech recording sessions.

4. Metadata and Speech Labels

Apart from the speech turn tags, also speakers' metadata and SC and VD labels are provided, as described in this section. The metadata was self-reported by each speaker at the end of the recording session.

Differently, the labels were collected by performing controlled listening tests with naïve normal-hearing external assessors (none of them participated as speaker). The semi-spontaneous speaker turns of Dialog 6 (pizza order) was chosen as speech stimulus (with mean duration 23.0 s, standard deviation of 3.3 s, range: 15.5 s–33.2 s). By controlling the speakers' dialog task, possible biases caused by disfluencies and different contents can be avoided for the estimation of speaker and voice attributes. At the same time, manifestations of speaker traits can still be perceived, as opposed to scripted speech.

4.1. Self-Reported Metadata

The socio-demographic data collected for each speaker comprise: age; gender; place of birth; chronological places of residence and duration of stay; place of birth of the mother; place of birth of the father; highest education level; educational background; main occupation; past occupations (if any); years of work experience (if any).

In addition, 273 (117 males and 156 females) out of the 300 speakers also completed a questionnaire for personality self-assessment (Rammstedt and John, 2007; Rammstedt and Danner, 2017) and self-assessed their vocal attractiveness using a continuous slider.

4.2. Interpersonal Speaker Characteristics

A group of 114 listeners (70 males and 44 females, 24.5 years old on average) participated in a series of listening tests to label perceived interpersonal speaker characteristics (SC), using the SC-Questionnaire with continuous sliders. 93 out of the 114 listeners spoke German as mother tongue, whereas the rest spoke other 10 different languages and claimed to have very good knowledge of German.

The SC-Questionnaire items (first two columns of Table 2) are based on previous research on interpersonal traits (Wiggins et al., 1988; Jacobs and Scholl, 2005); the three dimensional evaluations *valence, activity, potence* (Osgood et al., 1957); frequent social and physiological attributions (Weiss et al., 2018); and aspects of longer-term interpersonal attraction (Aronson et al., 2009). A first version of this questionnaire was validated (Weiss and Möller, 2011), and later applied (Fernández Gallardo and Weiss, 2017b) employing only a small set of 15 male voices. The version used for this work and its results are examined in (Fernández Gallardo and Weiss, 2017a).

The listeners first completed the SC-Questionnaire for a set of male voices and then for a set of female voices. Each assessor listened to and rated 16.4 male speakers and 23.2 female speakers on average, which results on an average of 15 x 34-dimensional continuous interpersonal ratings given by different listeners to each of the 300 speakers. They wore Shure SRH240 headphones (diotic listening, frequency range 20–20,000 Hz) and performed the test in a quiet office using a laptop and a mouse. They could listen to each speaker dialog as many times as they wished. Pauses were taken every 10 minutes approximately to avoid tiredness.

Using all continuous ratings, an exploratory factor analysis was conducted for male and for female speakers separately with `oblimin` rotation and minimum residual factoring method. The number of factors was determined by Horn's parallel analysis. Questionnaire items were retained when the main loading was greater than 0.5 and the difference between main loading and cross-loading exceeded 0.2. A second factor analysis was conducted on the remaining items, which explained 58% and 56% of data variance for male and for female voices, respectively. Cronbach's alphas were been examined and some items were removed to reach the maximum internal consistency possible for each factor.

We have identified five factors that are similar for both genders. These factors can be seen as perceptual dimensions that represent subjective attributions measured from observers' first impressions of speakers based on speech only. They are named (for male speech):

1. *warmth* (males: $\alpha = .88$, females: $\alpha = .89$)
2. *attractiveness* (males: $\alpha = .84$, females: $\alpha = .86$)
3. *confidence* (males: $\alpha = .78$, females: $\alpha = .80$)
4. *compliance* (males: $\alpha = .78$, females: $\alpha = .78$)
5. *maturity* (males: $\alpha = .76$, females: $\alpha = .71$)

The first, third, and fourth dimensions are described by the interpersonal circumplex (Wiggins et al., 1988; Jacobs and Scholl, 2005), while the second and fifth dimensions have their foundation on interpersonal attraction (Aronson et al., 2009) and age in speech (Weiss et al., 2018). For female speech, the same dimensions are found although in a slightly different ordering: *compliance* and *confidence* are, respectively, the 3rd and the 4th dimension for female speakers (Fernández Gallardo and Weiss, 2017a).

The questionnaire items associated with each factor and the corresponding loadings are presented in Tables 3 and 4 for male and for female speech, respectively. As factor scores, means of retained items are calculated, weighted by the item loadings.

4.3. Naïve Voice Descriptions

Considering the speakers' factor scores on the first two dimensions (warmth–attractiveness, with factor correlation of .59 and .63 for male and for female speech, respectively), a group of 20 "extreme" speakers have been selected: five male and five female speakers scoring lowest, and five female and five male speakers scoring highest in the warmth–attractiveness dimensional space. The subjective voice attributes of these speakers' speech has been thoroughly examined as indicated in this subsection, in order to better understand which acoustic cues are related to the different attributed speaker traits.

The 34-item semantic differential VD-Questionnaire (3rd and 4th columns of Table 2) is based on previous work (Scherer, 1974; Voiers, 1964; Fagel et al., 1983; Boves, 1984), was revised after (Weiss and Möller, 2011; Weiss et al., 2018), and then validated in (Weiss, 2016). This questionnaire has been completed by 26 assessors (13 males and 13 females, 26.6 years old on average, Ger-

SC: Antonyms (German)	SC: English translation
sympathisch / unsympathisch	likable / non-likable
unsicher / sicher	insecure / secure
unattraktiv / attraktiv	unattractive / attractive
verständnisvoll / verständnislos	sympathetic / unsympathetic
entschieden / unentschieden	decided / indecisive
aufdringlich / unaufdringlich	obtrusive / unobtrusive
nah / distanziert	close / distant
interessiert / gelangweilt	interested / bored
emotionslos / emotional	unemotional / emotional
genervt / nicht genervt	irritated / not irritated
passiv / aktiv	passive / active
unangenehm / angenehm	unpleasant / pleasant
charaktervoll / charakterlos	characterful / characterless
reserviert / gesellig	reserved / sociable
nervös / entspannt	nervous / relaxed
distanziert / mitfühlend	distant / affectionate
unterwürfig / dominant	conformable / dominant
affektiert / unaffektiert	affected / unaffected
gefühlskalt / herzlich	cold / hearty
jung / alt	young / old
sachlich / unsachlich	factual / not factual
aufgeregt / ruhig	excited / calm
kompetent / inkompetent	competent / incompetent
schön / hässlich	beautiful / ugly
unfreundlich / freundlich	unfriendly / friendly
weiblich / männlich	feminine / masculine
provokativ / gehorsam	offensive / submissive
engagiert / gleichgültig	committed / indifferent
langweilig / interessant	boring / interesting
folgsam / zynisch	compliant / cynical
unaufgesetzt / aufgesetzt	genuine / artificial
dumm / intelligent	stupid / intelligent
erwachsen / kindlich	adult / childish
frech / bescheiden	bold / modest

VD: Antonyms (German)	VD: English translation
klangvoll / klanglos	sonorous / flat
tief / hoch	low / high
nasal / nicht nasal	nasal / not nasal
stumpf / scharf	blunt / sharp
gleichmäßig / ungleichmäßig	even / uneven
akzentfrei / mit Akzent	accented / without accent
dunkel / hell	dark / bright
leise / laut	quiet / loud
knarrend / nicht knarrend	creaky / not creaky
variabel / monoton	variable / monotonous
angenehm / unangenehm	pleasant / unpleasant
deutlich / undeutlich	articulate / inarticulate
rau / glatt	coarse / not coarse
klar / heiser	clear / hoarse
unauffällig / auffällig	not remarkable / remarkable
schnell / langsam	quick / slow
kalt / warm	cold / warm
unnatürlich / natürlich	unnatural / natural
stabil / zittrig	stable / shaky
unpräzise / präzise	imprecise / precise
brüchig / fest	brittle / firm
unmelodisch / melodisch	not melodious / melodious
angespannt / entspannt	tense / relaxed
holprig / gleitend	bumpy / smooth
lang / kurz	long / short
locker / gepresst	lax / pressed
kraftvoll / kraftlos	powerful / powerless
flüssig / stockend	fluent / halting
weich / hart	soft / hard
professionell / unprofessionell	professional / unprofessional
betont / unbetont	emphasized/ not emphasized
sanft / schrill	gentle / shrill
getrennt / verbunden	disjointed / jointed
nicht behaucht / behaucht	not breathy / breathy

Table 2: 34 semantic-differential items of the SC (left) and VD (right) questionnaires. The 300 speakers have been labeled on the SC-Questionnaire items and the 20 selected "extreme" speakers on the VD-Questionnaire items. Continuous scales from 0 to 100 have been employed to evaluate every item.

Right adjective	Male SC factor loadings				
(translated)	warm.	attr.	conf.	comp.	matu.
hearty	.85				
affectionate	.84				
distant	-.76				
friendly	.59				
unsympathetic	-.58				
non-likable	-.52				
not irritated	.51				
attractive		.85			
ugly		-.79			
pleasant		.58			
interesting		.48			
secure			1.00		
indecisive			-.60		
submissive				.87	
cynical				-.71	
old					.82
childish					-.73

Table 3: Male SC factor loadings

Right adjective	Female SC factor loadings				
(translated)	warm.	attr.	comp.	conf.	matu.
hearty	.84				
affectionate	.84				
distant	-.78				
friendly	.56				
unsympathetic	-.49				
not irritated	.49				
non-likable	-.45				
attractive		.83			
ugly		-.81			
pleasant		.59			
submissive			.80		
cynical			-.72		
secure				.82	
indecisive				-.81	
childish					-.81
old					.68

Table 4: Female SC factor loadings

Right adjective (translated)	Male VD factor loadings *neg prof.	tens.	melo.	brig.
firm	-.78			
precise	-.72			
unprofessional	.65			
smooth	-.65			
shaky	.65			
halting	.64			
inarticulate	.61			
hard		.70		
pressed		.66		
relaxed		-.59		
warm		-.55		
jointed		-.50		
shrill		.49		
monotonous			-.78	
melodious			.65	
not emphasized			-.59	
bright				.87
high				.76
sharp				.58

Table 5: Male VD factor loadings

Right adjective (translated)	Female VD factor loadings *neg flue.	brig.	*neg prof.	shri.
shaky	.75			
firm	-.71			
smooth	-.67			
halting	.61			
high		.85		
bright		.79		
not coarse		.47		
hoarse		-.31		
unpleasant			.90	
inarticulate			.66	
unprofessional			.58	
shrill				.72
hard				.66
warm				-.52

Table 6: Female VD factor loadings

man as mother tongue), for each of the 20 selected "extreme" speakers using the same speech material (the pizza dialog).

Our procedure for factor analysis, conducted analogously as the one previously described, revealed four different dimensions for the description of the male and the female voices. The second factor analysis explained 54% and 53% of data variance for male and for female voices, respectively. For male speech, the dimensions found are:

1. *proficiency precision and fluency* (*negative, $\alpha = .87$)
2. *tension* ($\alpha = .79$)
3. *melody* ($\alpha = .83$)
4. *brightness* ($\alpha = .81$)

and, for female speech:

1. *fluency* (*negative, $\alpha = .81$)
2. *brightness* ($\alpha = .76$)
3. *proficiency precision* (*negative, $\alpha = .82$)
4. *shrillness* ($\alpha = .71$)

The gender difference in the dimensions found might be due to the small number of speakers tested. It can be speculated that *shrillness* is only manifested for female speech due to their generally higher pitch level compared to males. Tables 5 and 6 present the factor loadings calculated for each of the retained questionnaire items for male and for female speech, respectively. It has to be noted that items would load on positive factors tagged with **negative* with the opposite sign as the one indicated in the tables, e.g. the *precise* and *unprofessional* adjectives would load on positive *proficiency precision and fluency* for male speech with .72 and -.65, respectively.

We then examined the effects of speakers' warmth–attractiveness on the obtained VD factor scores. Conducted Wilcoxon rank-sum tests suggested that, for male speakers, *melody* and *brightness* factor scores differ significantly for perceived low warm–attractive speakers compared to perceived high warm–attractive speakers ($p < .01$ and $p < .05$, respectively). For female speakers, this statistical significant difference has been found for *fluency* ($p < .01$), *brightness* ($p < .05$), and *proficiency precision* ($p < .01$). These findings indicate the plausibility to classify perceived speaker traits based on speech features related to their voice descriptions. However, the voice descriptions of more speakers need to be analyzed in order to better determine the statistical effects between the SC and VD dimensions.

5. Conclusions and Outlook

In this paper we have presented the NSC corpus, which comprises clean conversational speech recordings from 300 German speakers and continuous numeric externally-assessed labels of speaker characteristics and voice descriptions.

The NSC corpus has been made freely available to the scientific community at the CLARIN repository, as mentioned before. The participating speakers gave their consent that "*the collected data will be exclusively used for scientific research and teaching activities. Accredited scientific institutions may access the data but not distribute them to third parties.*" (translated from German). It is foreseen that the data will also be available at the ELRA and LDC repositories.

The entire corpus material (50 GB of data) comprises 9192 minutes of speech (wav), the files employed as stimuli for SC- and VD-labeling (wav, 115 minutes), csv files with speakers' turn tags, listeners' ratings using the SC and VD questionnaires, SC and VD items–dimensions information, factor scores derived from the conducted factor analyses, speakers' metadata, and database documentation.

The collected data contributes to the investigation towards the detection of acoustic and linguistic cues that manifest subjective speaker social attributes. Following the Brunswik Lens Model revised by Scherer (1978), the features that can be extracted from the speech signal (e.g. pitch and formant frequencies, speech tempo, etc.) can be seen as "Distal Cues", whereas the collected VC-subjective labels

represent the "Proximal Percepts" that directly account for the final listeners' impressions of speakers (i.e. the SC dimensions identified in our analysis in Subsection 4.2.). The NSC data facilitates the research necessary to clarify the relationship between "Distal Cues" and "Proximal Percepts", which should lead to machines reaching the human performance in the attribution of speaker social characteristics.

The automatic detection of speaker interpersonal characteristics and traits is relevant to improve adaptive human-machine speech dialog systems. Speech and prosody production and conversational behavior in human-human interactions can be studied by analyzing speaker's and interlocutor's turns of semi-spontaneous and spontaneous speech. Based on recognized attributes from the users, systems should be able to adapt their dialog strategy and language generation mechanisms pursuing higher user acceptance (Berg, 2014).

The NSC data material may also be of interest to phoneticians and speech scientists requiring high-quality clean recordings in German.

6. Acknowledgments

This work has been supported by the German Research Foundation (DFG) [grant FE 1603/1-1].

7. Bibliographical References

Aronson, E., Wilson, T. D., and Akert, R. M. (2009). *Social Psychology*. Prentice Hall, 7 edition.

Berg, M. M. (2014). *Modelling of Natural Dialogues in the Context of Speech-based Information and Control Systems*, volume 108 of *Dissertations in Database and Information Systems*. Akademische Verlagsgesellschaft in cooperation with IOS Press.

Boves, L. (1984). *The Phonetic Basis of Perceptual Ratings of Running Speech*. Dordrecht, Holland ; Cinnaminson, U.S.A. : Foris Publications.

Burkhardt, F., Huber, R., and Batliner, A. (2007). Application of Speaker Classification in Human Machine Dialog Systems. In C. Müller, editor, *Speaker Classification I - Fundamentals, Features, and Methods*. Springer, Berlin, Heidelberg.

Fagel, W. P. F., Van Herpt, L. W. A., and Boves, L. (1983). Analysis of the Perceptual Qualities of Dutch Speakers Voice and Pronunciation. *Speech Communication*, 1(4):315–326.

Fernández Gallardo, L. and Weiss, B. (2017a). Perceived Interpersonal Speaker Attributes and their Acoustic Features. In *Phonetik und Phonologie im deutschprachigen Raum (PundP13)*.

Fernández Gallardo, L. and Weiss, B. (2017b). Towards Speaker Characterization: Identifying and Predicting Dimensions of Person Attribution. In *Annual Conference of the International Speech Communication Association (Interspeech)*, pages 904–908.

Fernández Gallardo, L. (2016). Recording a High-Quality German Speech Database for the Study of Speaker Personality and Likability. In *Phonetik und Phonologie im deutschprachigen Raum (PundP12)*, pages 43–46.

ITU-T Recommendation P.805, (2007). *Subjective Evaluation of Conversational Quality*. International Telecommunication Union, CH-Geneva.

Jacobs, I. and Scholl, W. (2005). Interpersonale Adjektivliste (IAL). *Diagnostica – Zeitschrift für Psychologische Diagnostik und Differentielle Psychologie*, 51(3):145–155.

Osgood, C., Suci, G., and Tannenbaum, P. (1957). *The Measurement of Meaning*. Illini Books, IB47. University of Illinois Press.

Rammstedt, B. and Danner, D. (2017). Die Facettenstruktur des Big Five Inventory (BFI). *Diagnostica*, 67:70–84.

Rammstedt, B. and John, O. P. (2007). Measuring Personality in One Minute or Less: A 10-Item Short Version of the Big Five Inventory in English and German. *Journal of Research in Personality*, 41(1):203–212.

Scherer, K. R. (1974). Voice Quality Analysis of American and German Speakers. *Journal of Psycholinguistie Research*, 3:281–298.

Scherer, K. R. (1978). Personality Inference from Voice Quality: the Loud Voice of Extroversion. *European Journal of Social Psychology*, 8:467–487.

Voiers, W. D. (1964). Perceptual Bases of Speaker Identity. *The Journal of the Acoustical Society of America*, 36:1065–1073.

Weiss, B. and Möller, S. (2011). Wahrnehmungsdimensionen von Stimme und Sprechweise. In *Elektronische Sprachsignalverarbeitung (ESSV)*, pages 261–268.

Weiss, B., Estival, D., and Stiefelhagen, U. (2018). Studying Vocal Perceptual Dimension of Non-experts by Assigning Overall Speaker (Dis-)Similarities. *Acta Acustica united with Acustica*, 104:174–184.

Weiss, B. (2016). Voice Descriptions by Non-Experts: Validation of a Questionnaire. In *Phonetik und Phonologie im deutschprachigen Raum (PundP12)*, pages 228–231.

Wiggins, J. S., Trapnell, P., and Phillips, N. (1988). Psychometric and Geometric Characteristics of the Revised Interpersonal Adjective Scales (IAS-R). *Multivariate Behavioral Research*, 23(4):517–530.

Evaluation of Automatic Formant Trackers

F. Schiel, Th. Zitzelsberger
Bavarian Archive for Speech Signals, Ludwig-Maximilians-Universität
Schellingstr. 3, 80799 München, Germany
{schiel,tzom}@bas.uni-muenchen.de

Abstract

Four open source formant trackers, three LPC-based and one based on Deep Learning, were evaluated on the same American English data set VTR-TIMIT. Test data were time-synchronized to avoid differences due to different unvoiced/voiced detection strategies. Default output values of trackers (e.g. producing 500Hz for the first formant, 1500Hz for the second etc.) were filtered from the evaluation data to avoid biased results. Evaluations were performed on the total recording and on three American English vowels [i:], [u] and [ʌ] separately. The obtained quality measures showed that all three LPC-based trackers had comparable RSME error results that are about 2 times the inter-labeller error of human labellers. Tracker results were biased considerably (in average too high or low), when the parameter settings of the tracker were not adjusted to the speaker's sex. Deep Learning appeared to outperform LPC-based trackers in general, but not in vowels. Deep Learning has the disadvantage that it requires annotated training material from the same speech domain as the target speech, and a trained Deep Learning tracker is therefore not applicable to other languages.

Keywords: formant tracker, evaluation, VTR-TIMIT

1. Introduction

This paper presents the methodology and results of a technical evaluation of four open source algorithms for automatic formant tracking (in the following referred to as 'formant trackers'). Currently, a small number of open source formant trackers is widely used by speech scientists and speech engineers. A number of earlier studies evaluated selections of these formant trackers on selected vowels, on words spoken in isolation and/or on speech of a small number of speakers: Derdemezis et al. 2016 evaluated 4 LPC-based formant trackers on 4 different vowels in isolated words uttered by 8 speaker groups; Deng et al, 2006 evaluated their own formant tracker compared to WaveSurfer based on VTR-TIMIT; Harrison, 2004 compared three LPC-based algorithms on data of two speakers. But to our knowledge there exists no objective evaluation of the quality of more than two algorithms based on the same data set comprising a reasonable number of speakers of both genders and fluent speech. Partly this is due to the fact that a manually controlled reference corpus of fluent speech for formant tracks is difficult to obtain. Fortunately, with the publication of the VTR-TIMIT (Vocal Tract Resonances TIMIT) corpus by Deng et al (2006) we are now able to perform such an evaluation, at least for US American English.

Aside from the direct comparison of different formant trackers as presented in this study there exist numerous studies that compare the out-come of a single formant tracker to one or more expert annotations, or evaluate the influence of methodological parameters such as the time point of measurement (Kendall & Vaughn 2015), the number of formants or LPC order (Vallabha & Tuller, 2002; Harrison 2004), the signal quality (Rathcke et al. 2016), or even the outcome of different expert groups using the same formant tracker but different methodologies (Duckworth et al. 2011). Derdemezis et al (2016) investigated the influence of several measurement parameters; they also give a very detailed discussion of existing studies regarding parameter manipulation (see also Burris et al, 2014). Most of these earlier studies are in agreement that the quality of formant trackers' results can be improved by adjusting the parameters of the algorithm to the given task, i.e. dependent on the algorithm itself, the age, gender and health of speakers, the point of measurement, and the quality of the recording. On the other hand, many authors in the literature agree that, even when given methodological rules that may improve format tracker output, in practical terms most researcher tend to use formant trackers with their respective default settings and do not adjust tracking parameters as advised.

In this study we therefore do not give any methodological recommendations for the four tested formant trackers but rather compared the out-come when using default tracker parameters, and the impact caused by the voiced/unvoiced detection, the speaker genders and three vowel classes of American English.

2. Formants

A formant is a resonance in the speech signal caused by the geometry of the physiological tubular system of the speaker's vocal tract. Formants are considered to be the primary phonetic feature for distinguishing vowel classes as well as place of articulation in consonant-vowel transitions. Furthermore, since formants are determined by the ideosyncratic physiological form of a speaker's vocal tract, they play a crucial role in forensic speaker recognition, automatic speaker identification and verification, sex and age recognition (e.g. Rose, 2003, pp. 221).

A formant is typically defined by three parameters: the center frequency (often called formant frequency), the bandwidth and the amplitude of the resonance. Technically, formants in a digitized speech signal are often decribed as complementary poles in the z-transform of the vocal tract filter, where the radial position of the pole defines the center frequency and the distance to the unit circle (and distance to neigboring poles/zeros) defines bandwidth and amplitude.

The lower formants 1-5 are widely used as phonetic features in linguistic-phonetic and forensic analysis but also as basic features in speech technology applications (such as speech morphing, speech and speaker recognition). It is therefore not surprising that the development of algorithms to detect and track lower

formants automatically in a speech recording has a long tradition going back to Rabiner & Schafer (1970).

3. Formant Trackers

Since formants are basically caused by an acoustical filter operation where the glottal source signal is filtered by a infinite-impulse-response filter, i.e. a filter having 5 or more complementary poles in its z-transform, they cannot be determined analytically from the recorded speech signal without prior knowledge of the source signal (which is usually not available). Most algorithms to estimate formant parameters from the recorded speech signal therefore either apply homo-morphic analysis of the spectral envelope (e.g. cepstral analysis followed by a peak-picking strategy), or LPC analysis (Markel & Grey, 1982) to estimate the z-transform followed by a pole-picking strategy, or trained pattern recognition techniques (e.g. support vector machine, random forest or deep learning). The latter requires a training set of labelled formant tracks and is in most cases language dependent, while the former two approaches are inherently language independent and do not require any training material.

The task of formant tracking is further complicated by the fact that some algorithms assume a voiced signal for analysis, i.e. the spectral envelope encloses a harmonic spectrum consisting of a fundamental frequency line and the respective harmonic spectral lines at multiples of the fundamental frequency. Such algorithms typically produce more or less random results when applied to unvoiced parts of the speech signal. Therefore the formant estimation algorithm is often combined with a voiced-unvoiced detector to suppress formant analysis in unvoiced parts of the speech signal. Since the voiced/unvoiced detection in itself is error prone (e.g. in creaky voice), the output of the combined algorithm (= the formant tracker) is influenced by the performance of both algorithms.

In this study four formant trackers have been investigated:

- **PRAAT**, the built-in formant tracker of the praat tool by Boersma & Weenink (2017), 'Burg' method (cf. Childers 1978, pp. 252)
- **SNACK**, the formant tracker (version 2.2) of the Snack Sound Toolkit of KTH Stockholm by Kåre Sjölander (2017)
- **ASSP**, the formant tracker *forest* (version 2.8) contained in the Advanced Speech Signal Processor library by M. Scheffer (Scheffer, 2017), also contained in the R language package *wrassp*, and part of the Emu database management system (EMU-SMDS, Winkelmann 2017)
- **DEEP,** DeepFormant, a formant tracker (Keshet, 2017) based on deep learning techniques and trained on the training set of VTR-TIMIT (Dissen & Keshet, 2016). Contrary to the three other formant trackers this algorithm produces formant frequency estimates at all time points, i.e. there is no voiced/unvoiced detection.

PRAAT, SNACK and ASSP are based on LPC analysis; no homo-morphic formant tracker was evaluated in this study (cf. Kammoun et al, 2006 for a discussion of LPC vs. homo-morphic formant analysis).

4. Test Data VTR-TIMIT

Vocal Tract Resonance TIMIT (VTR-TIMIT) is an open source subcorpus annotation of TIMIT[1] (Garofolo et al, 1992) with 516 manually annotated recordings spoken by 186 (113m and 73f) speakers of American English (Deng et al, 2004). The subcorpus contains 282 phonetically compact (SX in TIMIT terminology) and 234 phonetically rich sentences (SI), but no dialectal speech (SA). The speech was first analysed by the formant tracker algorithm described in Deng et al, 2004, and subsequently hand-corrected. The manual correction was performed by a group of labelers based on visual inspection of the first three formants in the spectrogram (higher formants, bandwidths and amplitudes were not corrected). Inter-labeller agreement tests on a small sub-sample (16 sentences per 5 different labeller-pairings) yielded average frequency deviations of about 78Hz for the first formant (F1), 100Hz for F2 and 111Hz for F3 (Deng et al, 2006).

For technical reasons the VTR-TIMIT formant reference tracks are continuous over the total recording, i.e. there is no indication of where the speech is voiced or where formants are or are not visible in the spectrogram. Formants in unvoiced or silent parts of the signal were either interpolated linearly from the two adjacent voiced parts or horizontally extended at the initial or final voiced portion of the recording. This interpolation facilitates an evaluation of formant tracker output independently of the voiced-unvoiced detection of the tracker algorithm (because for every time frame of the recording there exists a reference value); on the other hand the resulting quality measure might be compromised: if a tracker is 'conservative' in the sense that it produces output only for the parts of the input signal where it is quite confident (clearly voiced parts), then this tracker will outperform other trackers who produce results in parts of the signal where the tracking is compromised for instance by creaky voice or noise associated with consonantal constrictions etc.

Since the formant tracker DEEP was trained on parts of VTR-TIMIT, the following evaluations of DEEP were only performed on the test part of VTR-TIMIT (a subset with 8f and 16m speakers). We did not restrict the tests of the LPC-based trackers on this subset (this would have compromised the statistical power of the analysis considerably), since results are not comparable between DEEP and the remaining algorithms anyway: DEEP has been trained on the training part of VTR-TIMIT and is therefore language- and corpus-dependent, while the three other formant tracker algorithms are language- and corpus-independent.

5. Evaluation Methodology

Quality Measures

Two quality measures were calculated to quantify the distance of formant tracker output to the annotation reference:

- **RSME**: root mean squared error calculated over the complete recording to quantify overall performance (zero being perfect match between formant tracker output and reference).

[1] TIMIT itself and thus the signals of VTR-TIMIT are not open source; refer to the Linguistic Data Consortium.

- **AVG**: average difference between reference and formant tracker output calculated over the complete recording to indicate systematic errors: a positive value indicates that the tracker tends to calculate formant estimates in average too low; a negative value indicates formant estimates are too high; zero indicates a perfect symmetry of errors around the annotation reference. Please note that the AVG *value* does not give any information about the quality.

Histograms of the AVG errors were plotted to identify multi-modal distributions, for instance caused by systematic formant confusion errors.

Formant Tracker Parameters

The four formant trackers were evaluated with their default parameter settings, except for the frame step size which was set to 10msec and the window length which was set to 25msec for all trackers to yield comparable number of evaluation frames. Other parameters were not changed under the assumption that the developpers have choosen these default parameter sets to optimize for best performance. As we will see in the results, using the default settings causes differences in measurement quality depending on the speaker's sex, since two of the trackers, SNACK and ASSP, use default parameters optimized for male speakers, while PRAAT uses a parameter set optimized for female speakers. However, using speaker-individual parameter settings in the evaluation pose a problem, since not all trackers offer a standard parameter set for female and male speakers. Instead we propose to inspect the results of trackers sorted according to their default sex parameter setting, i.e. to look at the results of female speakers for PRAAT and the results of male speakers for SNACK and ASSP.

The formant tracker DEEP has no parameters to influence speaker sex, but is trained to a dominantly male training data set (67f vs. 95m speakers) and is therefore expected to perform slightly better on male speech. Table 1 lists the available and chosen parameter settings (defaults are marked with an asterix *)

Parameter	ASSP	SNACK	PRAAT	DEEP
formants	4*	4*	5*	4*
LPC	18*	12*	10*	n/a
preemph	0.96*	0.7*	50Hz/oct*	unknown
window	blackman*	cos4*	gauss*	unknown
w. length	25ms	25ms	25ms*	unknown
stepsize	10ms	10ms	10ms	10ms*

Table 1: formant tracker parameter sets

Tests conditions

As mentioned earlier formant trackers have different strategies to distinguish between parts of the signal where formants can be detected versus parts of the signal that are unvoiced (or otherwise compromised in a way that no formants can be detected). To prevent quality measures from being skewed by this behavior we decided to run three different tests:

1. **NORM**: all trackers use the same voiced-unvoiced detection. We used the pitch detector in praat (which is an independent tool from the praat formant tracker) to determine in all VTR-TIMIT recordings when tracker output is to be evaluated. If a tracker did not deliver `real' results within these defined areas, these `fake' or zero results were excluded. For instance, SNACK outputs default formant values in parts of the signal. Fortunately, these can easily be detected and filtered from the evaluation data (less than 6% loss of evaluation data).
2. **DEFAULT**: trackers decided individually for which portions of the recording formant values were produced. Again, detectable `fake' values were filtered from the evaluation. This is basically the `normal' way to use a formant tracker, since it is very unlikely that a user will not accept the built-in voiced-unvoiced decision of a formant tracker. Since the formant tracker DEEP does not perform a voiced/unvoiced detection and PRAAT uses the same voiced/unvoiced detection as in the NORM test condition, only SNACK and ASSP results were evaluated in this test condition.
3. **VOWELS**: tracker outputs were restricted to the same voiced-unvoiced segmentation as in test NORM, but additionally restricted to the segments of the three vowels [i:], [u] and [ʌ] which are roughly the corner positions in the American English vowel space. This test was motivated by the fact that the first three formants are the predominant features for vowel quality. The three selected American vowels can be seen as representative sounds for high front, high back and low vowels.

6. Results

In the following RMSE and AVG results for all test conditions (see section 5) are presented.

6.1 Test Condition NORM

Table 2 lists the RMSE measures in Hz in test condition NORM for formants F1...F3 and for female (f) and male (m) speakers. As expected, absolute errors increased with formant order; the relative errors were in about the same range for all three formants. Errors of the DEEP tracker were exceptionally low, but the comparison with the remaining three trackers is not fair, since the test set was much smaller for DEEP (see section 4) and DEEP was trained to VTR-TIMIT and therefore has an advantage. The formant tracker PRAAT showed systematic errors with male speakers, while SNACK and ASSP performed better on male than on female speakers. However, when comparing the average RMSE error value for *female* speakers in PRAAT (194Hz) with *male* speakers in SNACK/ASSP (234/177Hz) we can see that the performance was in the same range (underlined values in Table 2). It is unlikely that these differences between

formant trackers are significant, considering that averaged human inter-labeller errors on the reference were in the range of 96Hz (see section 4). But it is clear that the LPC-based trackers in average (210Hz error) performed significantly worse than human labellers (about 2 times worse).

	F1		F2		F3	
	f	**m**	**f**	**m**	**f**	**m**
SNACK	126	**100**	291	**227**	313	**375**
ASSP	113	**96**	479	**211**	512	**225**
PRAAT	**116**	234	**217**	338	**249**	404
DEEP	120	97	195	167	252	169

Table 2: RMSE errors in the test condition NORM (underlined values: formant tracker parameters match speaker sex).

Looking at averaged AVG measures in Table 3, the main result of the evaluation was that ASSP tended to underestimate formant frequencies F2/F3 for female speakers (200Hz too low), while PRAAT did the opposite, i.e. it overestimated all three formant frequencies for male speakers (120Hz too high). The averaged AVG errors for SNACK and DEEP were quite balanced especially for lower formants F1/F2.

	F1		F2		F3	
	f	**m**	**f**	**m**	**f**	**m**
SNACK	23	**-9**	1	**-8**	-45	**-102**
ASSP	-12	**-14**	187	**11**	216	**24**
PRAAT	**-38**	-116	**11**	-114	**-13**	-188
DEEP	86	56	5	-68	116	-26

Table 3: AVG errors in the test condition NORM rounded to two decimal places (underlined values: formant tracker parameters match speaker sex).

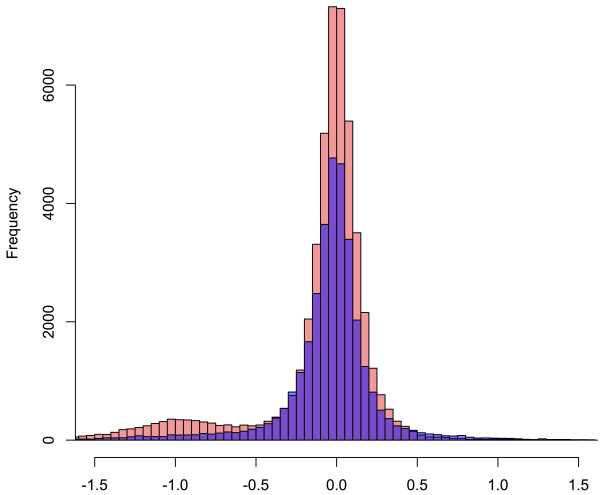


Figure 1: AVG error histogram of F3 (in kHz) estimated by SNACK; positive AVG error values denote estimates lower than reference, and vice versa; female speakers are light blue, male speakers are pink, overlap dark blue.

Selected AVG Histograms

Fig. 1 shows the AVG error histogram of F3 estimated by SNACK. Male values (pink) displayed a second peak at about -1000Hz AVG error (= estimated 1000Hz too high), indicating that SNACK in some cases confused F4 with F3 (tracks F3 in the location of F4); this error was not visible for female speakers (blue), probably because F4 of female speakers is much higher than of male speakers and therefore not easily confused with F3.

Fig. 2 shows the histogram of AVG errors for F2 estimated by ASSP. Here female speakers (light blue) showed more positive AVG errors, indicating that the ASSP estimates for F2 for female speakers were often too low. In contrast to Figure 1 there was no visible second peak which means that these lower estimates were probably not caused by classical formant confusion.

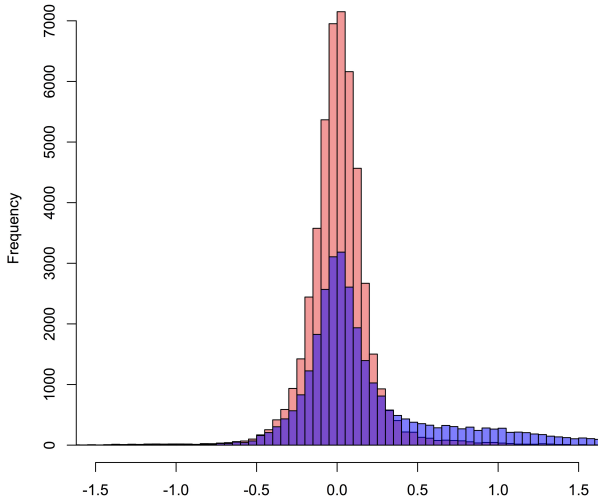


Figure 2: AVG error histogram of F2 (in kHz) estimated by ASSP (see Figure 1)

Fig. 3 displays the histogram of AVG errors for the first formant by the tracker PRAAT. One can clearly see that measurements for male Speakers (pink) were consistently overestimated (AVG error negatively skewed), and that formant tracker parameters were optimized for female speakers (light blue) which yielded AVG errors around zero.

6.2 Test Condition DEFAULT

The RMSE and AVG measures obtained in test condition DEFAULT for the formant trackers SNACK and ASSP were almost congruent with the results in the test condition NORM. Bonferroni corrected t-tests on *alpha = 0.01* applied on speaker aggregated errors RMSE and AVG (to avoid repeated measures) showed no significant changes in both quality measures and for the three first formants between test condition NORM and DEFAULT, except for the RMSE measure of the first formant F1 in formant tracker ASSP (but contrary to our expectation the error *increased* for the DEFAULT condition in this case). In

total, when evaluating SNACK and ASSP the normalisation of pitch detection seemed not to be a significant factor (for brevity we do not show the error measure tables here). The hypothesis that the different voiced/unvoiced segmentations of the formant trackers would have a significant impact on evaluation results has therefore been falsified.

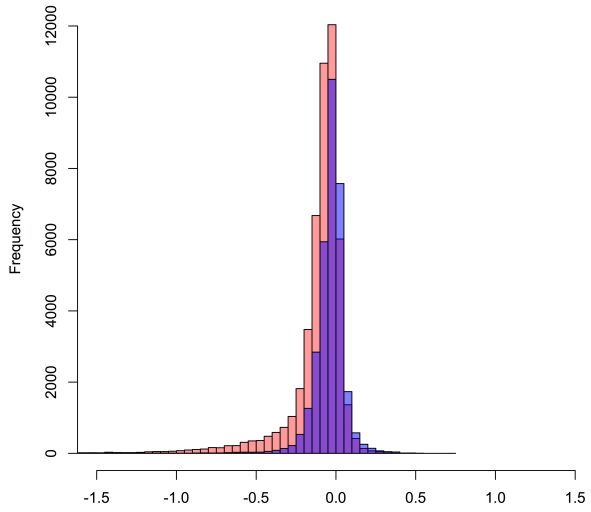


Figure 3: AVG error histogram of F1 (in kHz) estimated by PRAAT (see Figure 1 for details)

6.3 Test Conditions VOWELS

Table 4 shows RMSE errors for three vowels [i:] (5694 frames), [u] (1373 frames) and [ʌ] (2671 frames).

		F1		F2		F3	
		f	**m**	**f**	**m**	**f**	**m**
SNACK	**[i:]**	91	**82**	372	**234**	239	**373**
	[u]	73	**54**	224	**163**	300	**421**
	[ʌ]	143	**105**	190	**155**	276	**364**
ASSP	**[i:]**	76	**66**	744	**245**	342	**200**
	[u]	65	**48**	280	**154**	317	**151**
	[ʌ]	117	**94**	347	**139**	634	**215**
PRAAT	**[i:]**	**62**	254	**216**	192	**190**	276
	[u]	**57**	469	**127**	300	**225**	425
	[ʌ]	**91**	145	**99**	207	**205**	281
DEEP	**[i:]**	97	80	268	185	226	167
	[u]	92	66	169	124	348	124
	[ʌ]	132	109	144	130	264	194

Table 4: RMSE errors in vowel segments [i:], [u] and [ʌ] (underlined values: formant tracker parameters match speaker sex).

RMSE results were significantly better for vowel segments than across the total recording which is not surprising. Again, formant tracker parameters that do not match speaker's sex caused larger errors (e.g. 744Hz RMSE error on F2 for female speakers in ASSP). The advantage of DEEP against the LPC-based trackers noted in the NORM test condition was not as prominent here. The low centralized vowel [ʌ] seemed to be more difficult to track than the high vowels [i:] and [u]; one possible explanation is that reduced/centralized vowels in American English are often labelled with [ʌ] and are therefore more often hypo-articulated than [i:] and [u]. Hypo-articulation correlates with lower formant amplitudes and higher bandwidths which in turn makes the tracking of the formant frequency more difficult. Another explanation is that F1 and F2 tend to be close together for [ʌ] and might therefore be harder to separate by the formant tracker.

7. Conclusion

Four open source formant trackers, three LPC-based and one based on Deep Learning, were evaluated on an American English data set. The three traditional LPC-based formant trackers performed similarly well, when their respective parameter sets matched the sex of the tracked speaker; otherwise results were sometimes heavily skewed in one direction which could easily lead to the misinterpretation of tracker results. The average performance in terms of RMSE (210Hz) was about two times higher than reported comparable inter-labeller agreement error for human labellers on the same data set (96Hz, Deng et al 2006). The SNACK formant tracker turned out to be robust against wrong speaker sex settings, but sometimes produced default values as output (F1=500Hz, F2=1500Hz, ...) without warning; if not filtered these could be misinterpreted as formant measurements. The Deep Learning formant tracker appeared to out-perform traditional LPC-based methods in general but not when tested on vowels only. However, since the Deep Learning formant tracker was trained on the training set of the same speech corpus used for testing, this finding will probably not hold for other data sets (and especially not for other languages).

Some take home messages when dealing with formant trackers:

- LPC-based formant trackers show about 2 times less precision than human labellers.
- whenever possible, adjust your tracker to the sex of the target speaker.
- check for and remove default output values (repetitions of exactly the same value).
- if a histogram of tracker results shows more than one peak, this could be an indication of formant confusion (e.g. F4 is sometimes recognized as F3); a possible solution is to increase the number of formants or reduce the spectral range (depending on the tracker algorithm)
- expect less reliable results in centralized vowels (and in hypo-articulated speech in general) and in lower vowels.

8. Bibliographical References

Boersma, P. & Weenink, D. (2017). Praat: doing phonetics by computer [Computer program]. Version 6.0.23, retrieved 2017-04-17 from http://www.praat.org/

Burris, C. & Vorperian, H.K. & Fourakis, M. & Kent, R.D. & Bolt, D.M. (2014). Quantitative and descriptive comparison of four acoustic analysis systems: vowel measurements. Journal of Speech, Language and Hearing Research, 2014 Feb;57(1):26-45. doi: 10.1044/1092-4388(2013/12-0103).

Childers, D.G. (1978). Modern spectrum analysis. IEEE Press Selected Reprint Series, Hoboken, NJ: John Wiley & Sons Inc.

Derdemezis, E. & Vorperian, H.K. & Kent, R.D. & Fourakis, M. & Reinicke, E.L. & Bolt, D. M. (2016). Optimizing Vowel Formant Measurements in Four Acoustic Analysis Systems for Diverse Speaker Groups. American Journal on Speech & Language Pathology, 2016 Aug; 25(3): 335–354. doi: 10.1044/2015_AJSLP-15-0020.

Dissen (2016). Formant Estimation and Tracking using Deep Learning. In: Proceedings of the INTERSPEECH, pp. 958 – 962.

Duckworth, M. & McDougall, K. & de Jong, G. & Shockey, L. (2011). Improving the consistency of formant measurement. International Journal of Speech, Language & Law 18, 35-51.

Garofolo, J.S. & Lamel, L. & Fisher, M.W. & Fiscus, J. & S. Pallett, D.S. & Dahlgren, N.L. & Zue, V. (1992). TIMIT Acoustic-phonetic Continuous Speech Corpus. Philadelphia, PA: Linguistic Data Consortium.

Harrison, P. (2004). Variability of formant measurements. MA Dissertation. York, UK:University of York.

Kammoun, M.A. & Gargouri, D. & Frikha, M. & Hamida, A.B. (2006). Cepstrum vs. LPC: A Comparative Study for Speech Formant Frequencies Estimation. In: GESTS Int'l Trans. Communication and Signal Processing,Laboratoire d'Electronique et de la Technologie de l'Information (LETI), Vol. 9, pp. 87-102.

Kåre Sjölander (2017). Snack-Sound-Toolkit, retrieved 2017-04-17 from http://www.speech.kth.se/snack.

Kendall, T. & Vaughn, C. (2015). Measurement variability in vowel formant estimation: A simulation experiment. In: Proc. of the International Conference on Phonetic Sciences 2015, Glasgow.

Keshet, J. (2017). DeepFormant, retrieved 2017-04-30 from https://github.com/MLSpeech.

Markel, J.E. & Gray, A.H. (1982). Linear Prediction of Speech. New York, NY: Springer .

Rabiner, L.R. & Schafer (1970). System for automatic formant analysis of voiced speech. JASA Vol 47, pp. 634-648.

Rathcke, T. & Stuart-Smith, J. & Torsney, B. & Harrington, J. (2016). The beauty in a beast: Minimising the effects of diverse recording quality on vowel formant measurements in sociophonetic real-time studies. Speech Communication 86 (2016), pp. 24-41.

Rose, Ph. (2003). Forensic Speaker Identification. International Forensic Science and Investigation, CRC Press.

Scheffer, M. (2017). Advanced Speech Signal Processor (libassp), retrieved 2017-04-17 from http://www.sourceforge.net/projects/libassp.

Vallabha, G., Tuller, B. (2002). Systematic errors in the formant analysis of steady-state vowels. Speech Communication 38, 141-160.

Winkelmann, R. & Harrington, J. & Jänsch, K. (2017). EMU-SMDS: Advanced speech database management and analysis in R. Computer, Speech & Language, 45 (2017), pp. 392-410.

9. Language Resource References

Deng, L. & Cui, X. & Pruvenok, R. & Huang, J. & Momen, S. & Chen, Y.N. & Alwan, A. (2006). A Database of Vocal Tract Resonance Trajectories for Research in Speech Processing. In: Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing.

Design and Development of Speech Corpora for Air Traffic Control Training

Luboš Šmídl, Jan Švec, Daniel Tihelka, Jindřich Matoušek, Jan Romportl, Pavel Ircing
Department of Cybernetics, University of West Bohemia
Univerzitní 8, 306 14 Plzeň, Czech Republic
{smidl, honzas, dtihelka, jmatouse, rompi, ircing}@kky.zcu.cz

Abstract

The paper describes the process of creation of domain-specific speech corpora containing air traffic control (ATC) communication prompts. Since the ATC domain is highly specific both from the acoustic point-of-view (significant level of noise in the signal, non-native English accents of the speakers, non-standard pronunciation of some frequent words) and the lexical and syntactic perspective (prescribed structure of utterances, rather limited vocabulary), it is useful to collect and annotate data from this specific domain. Actually, the ultimate goal of the research effort of our team was to develop a voice dialogue system simulating the responses of the pilot that could be used for training aspiring air traffic controllers. In order to do so, we needed – among other modules – a domain-specific automatic speech recognition (ASR) and text-to-speech synthesis (TTS) engines. This paper concentrates on the details of the ASR and TTS corpora creation process but also overviews their usage in preparing practical applications and provides links to the distribution channel of the data.

Keywords: speech corpus, speech recognition, text-to-speech, air traffic control communication

1. Introduction

The air traffic control (ATC) constitutes a crucial segment of the whole air traffic industry – the air traffic controllers communicate with the pilots almost continuously in order to ensure the fluent and safe flow of the aerial traffic. The job of a controller is very demanding and requires – besides the specific personal prerequisites – an intensive training.

This training is mainly focused on teaching and reinforcing the communication skills of the aspiring controller. The current state-of-the-art training procedure (at least in the Czech Republic) involves so-called *pseudopilots*. These are usually retired pilots that prepare training scenarios and consequently act as pilots of virtual plane (usually more than one at a time), communicate with the controller in training (*trainee*) and process the spoken prompts received from trainees to the form that can be entered into the software that simulates the plane movement on the radar screen.

Two major drawbacks were identified in such a training setting. The first is rather obvious – the length of the controller training (approx. 2 years on average) and the relatively high salaries of the pseudopilots make the whole process very expensive. This was actually the first incentive that sparked the idea of developing an automatic training simulator based on the intelligent spoken dialogue system. Only after delving into the specifics of the ATC communication, we have realized that the scenario involving the pseudopilots in actually quite unrealistic in several aspects:

First, In the real ATC scenarios, the controller will need to understand the English utterances pronounced mostly by non-native speakers (remember that we are talking about the air space situated in the Central Europe), sometimes with quite an exotic accent. On the other hand, during the training sessions involving the human pseudopilots, it is usually the case that Czech trainees are listening to the English utterances pronounced by a retired Czech pilot and thus the mutual understanding is naturally much easier. Moreover, the training environment lacks the noise that is massively present in the real-world VHF radio communication; this might lead to a drastic decrease of the unprepared controller's ability to understand the communication once he is put into service.

On the other hand, the human pseudopilot usually handles several virtual airplanes. This might result into confusion of the trainee as he hears the same voice from different simulated aircrafts.

So, when designing the spoken dialogue system (let us call it the *artificial pseudopilot – aPP* and see its simplified block diagram on Figure 1) that should replace the human pseudopilot, we tried to rectify the shortcomings mentioned above.

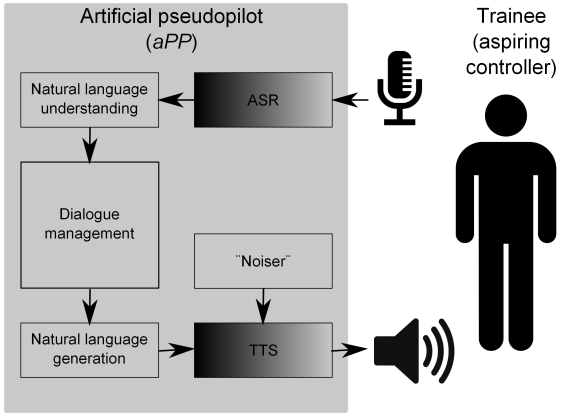


Figure 1: Block diagram of the artificial pseudopilot

All the specifics of the ATC communication that we have listed are actually connected with the "interface" blocks of the *aPP* – that is, the automatic speech recognition (ASR) and text-to-speech (TTS) modules. The state-of-the-art methods for development of those modules naturally require large speech corpora for either training the acoustic

and language models (ASR) or building the inventory of basic speech units (TTS). Since the ATC data are only rarely being collected[1], we have decided to design and collect our own corpora, taking into account the peculiarities of the ATC communication and tailored to our specific needs.

2. ASR Corpora

We took advantage of the fact that one of our industrial partners develops complex IT solutions for several ATC authorities and airports and, as such, has access to the ATC communication recordings. We were therefore able to secure almost 140 hours of recorded communication in the following structure:

- GRP (ground control) – communication before takeoff and after landing – 19.2 hours of data
- TWR (tower control) – communication during takeoff, landing and landing standby – 22.5 hours
- APP (approach control) – communication during landing approach – 25.5 hours
- ACC (area control) – communication during overflights and cruises – 71.3 hours

Those data were first segmented and the segments were classified as either speech or non-speech using an in-house voice activity detector (Prcín et al., 2002). The segments classified as speech were consequently imported into the Webtransc annotation tool (Valenta and Šmídl, 2015) that serves for online annotation of multimedia data. It enables annotators to play the segments, to transcribe their content (as well as to add various markers dedicated for non-speech events) and to add several types of metadata (such as the speaker's communication role – pilot or controller, in this case). The screenshot of Webtransc is shown on Figure 2.
The majority of the employed annotators already had some experience with speech corpora annotation. We have nevertheless prepared a detailed transcription manual, paying special attention to instructions that concern handling of non-standard pronunciations, special ATC terminology, spelling alphabet and other issues peculiar to ATC.
Here are the most interesting instructions from the manual:

- the utterances from pilots are marked as *Air* whereas the controllers' utterances are tagged with the label *Ground*. Naturally, this distinction is very important as both channels have significantly different acoustic qualities.
- the words pronounced in a non-standard way are manually equipped with the actual phonetic transcription (written using the Arpabet transcription code[2])
- the numerals pronounced correctly according to the ATC protocols[3] are written simply as numbers separated by space – otherwise they are also equipped with a actual phonetic transcription. The same notation is used for letter that are spelled out using the spelling alphabet. For example, the transcription `3 0 3 V direct to H D O` means that this particular fragment was pronounced "*tree zero tree victor* direct to *hotel delta oscar*"
- three broad classes of *non-speech events* were identified and marked in the transcriptions:
  - *hesitation sounds* made by the speaker – e.g. "ehm", "uh-huh"', "hmm", etc.
  - non-speech sounds produced by the speaker – e.g. coughing, lip-smacking, exceptionally loud breath, laughing, etc.
  - environmental noises that stand out from the (already quite high) natural noise in the background

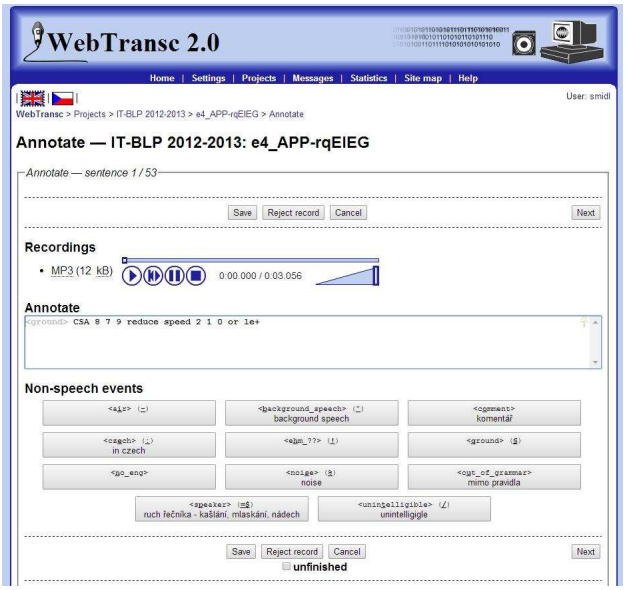


Figure 2: Screenshot of the WebTransc tool

After the transcription phase, the data samples were checked and, based upon this inspection, several rewriting rules (in the form $A \rightarrow B$) were designed to correct the most common typos (such as "aftrenoon" to "afternoon") and unify possibly inconsistent transcription of certain names (e.g. "Germanwings" to "German_Wings").
The unique words from such cleaned and normalized transcripts were then used as a basis for creation of the pronunciation lexicons. The actual pronunciations (phonetic baseforms) are either:

- extracted from the existing pronunciation lexicons available for English, namely
  - BEEP (BrE)
  - CALLHOME (AmE)
  - SWITCHBOARD (AmE)
  - CMU Dict (AmE)
- taken from transcripts where they could be written by the annotators

[1] The only other ATC corpus known to authors is the Air Traffic Control Complete (Godfrey, 1994) which is quite dated and also of rather poor technical quality.

[2] `http://en.wikipedia.org/wiki/Arpabet`

[3] See for example `http://aviationknowledge.wikidot.com/aviation:nato-phonetic-alphabet`

- manually created by an expert in cases when neither of the above sources provided any outcome

The resulting lexicon also employs the above mentioned Arpabet transcription code that is (most notably) used in the CMU Dict. There is a one-to-one mapping between Arpabet and the International Phonetic Alphabet (IPA) symbols. The portion of the speech recordings (approx. 20 hours) with corresponding transcriptions obtained in the way described in the previous paragraphs was released via the LINDAT/CLARIN repository (Šmídl, 2011). The pronunciation lexicon together with unigram, bigram and trigram word counts that can be directly used for language modeling was released later on via the same distribution channel (Šmídl, 2013).

3. TTS Corpora

For the reasons that were already mentioned in the introduction, in order to keep the artificial pseudopilots realistic, we should prepare a relatively broad portfolio of English TTS voices with various accents. Standard commercial high-quality unit-selection TTS voices are based on speech corpora counting more than 10 thousand phonetically rich sentences recorded by a professional speaker in a sound studio, often with an electroglottograph. Recording of such amounts of data is very time consuming (and thus costly) both for the speaker and for the subsequent expert annotation. Moreover, if the speaker is supposed to record in English which is not his/her native language, practical difficulties and expenses increase even more.

Therefore, we have decided to utilize a domain-specific approach to TTS voice creation, relying on the fact that the natural language generating module of the artificial pseudopilot generates utterances only from the restricted domain of typical ATC radio communication (albeit the number of potential different utterances is unlimited). The key to high quality TTS in such a scenario is to record a speech corpus that is both small in size and at the same time rich in variability – that is, it covers well the domain as the whole and/or allows high quality generation of the missing pieces. First step is to select chunks of texts (not necessarily the whole sentences) to be recorded. Using the algorithm described in (Jůzová and Tihelka, 2014), we have selected 1 000 chunks[4] from more than 34 000 transcripts of the speech uttered by pilots that appeared in the ASR corpus described in Section 2.. This limited set still covers 46.6% of types (unique words) of the original text data.

For the actual corpus recording, we have used the tool described in (Matoušek et al., 2008). Since the texts selected for recording are, due to the selection algorithm, actually just the fragments of sentences (chunks) that might not be always meaningful, it can be difficult for speaker to record them with a required prosody. In order to help the speaker to adhere to the correct prosody style during recording, we grouped the chunks derived from a particular phrase together, and presented them in relation with their source phrase. That way the speaker has the chance to read the phrase as a whole, and then to read the individual chunks in the same style as he/she read the chunk within the whole phrase.

The first *aPP* voice was recorded by a non-professional Taiwanese male speaker with a very typical and strong Mandarin accent in English. In order to test the "worst-case scenario", we deliberately did not use a professional sound studio but recorded the utterances in a standard office room instead, using a regular PC with a high-end external sound card and a microphone; no electroglottograph signal was recorded. The recording of the previously mentioned 1 000 text chunks took about 10 hours that were split into two days of recording. The annotations were subsequently manually checked and corrected where necessary. The pitch marks (i.e, the glottal closure instants) were identified directly from the speech signal using an algorithm described in (Legát et al., 2011).

The resulting corpus was used for the development of the actual TTS system (see Section 4. for details) and also released via the LINDAT/CLARIN repository (Jindřich Matoušek, 2014b). The released version includes – besides the obvious speech files and corresponding transcriptions – the information about the pitch marks, the pronunciation lexicon and the corresponding phonetic alphabet (the Arpabet transcription code, the same that was used in the ASR lexicon mentioned in Section 2.).

Later on, the the procedure described in this section was used to prepare and release the corpora containing the domain-specific English voices with Serbian (Jindřich Matoušek, 2014a), Czech (Jindřich Matoušek, 2015a) and German (Jindřich Matoušek, 2015b) accents.

4. Usage of the Corpora for Application Development

The ASR corpus described in Section 2. was used to train several versions of the automatic speech recognition system. All of them utilize the Hidden Markov Model (HMM) architecture for acoustic modeling and word n-grams as language models. The initial acoustic model was trained using only 17.5 hours of annotated speech data where the utterances of the pilots (denoted *Air*) and controllers (*Ground*) were mixed together. This baseline model was used essentially for "sanity checking" and setting up the basic acoustic model parameters (such as the number of HMM states and Gaussian mixtures). The next set of acoustic models consists of models trained separately for *Air* and *Ground* data, using the entire corpus of transribed speech (except for the 1.4 hours that were put aside as a test set). The ASR results achieved on the test set are summarized in Table 1.

Data source	Training data size [hours]	WER [%]
Air	54.9	25.27
Ground	78.8	7.59

Table 1: ASR results for individual data sources

The table shows that the recognition performance for controllers' utterances is far better than the the one for the pilots' data. This was of course to be expected as the data

[4]The number of selected chunks was chosen rather arbitrarily – the motivation was not to exceed two days of corpus recording.

are recorded at the control tower and thus the ground control speech is naturally acoustically much cleaner than the speech transmitted from the planes via radio. It was also good news for the prospective development of the *aPP* dialogue system as it is supposed to recognize controllers' speech, not pilots' (cf. Figure 1).

The acoustic models for the speech recognition module that is used in the actual dialogue system were trained using the ASR speech corpus described in this paper (140 hours in total) augmented with additional 460 hours of LibriSpeech data (Panayotov et al., 2015). The final set of models employs three-state HMMs with 2 000 states in total and 16 Gaussian mixtures per state. The demo of this ASR system can be found at `https://itblp.zcu.cz/asr-sc/`.

The TTS systems developed for the *aPP* dialogue system are based on the concatenative speech synthesis paradigm, employing unit-selection algorithms with diphones as basic units. In order to enhance the realistic feeling of the artificial pseudopilot's speech, a module capable of adding several types of noises typical for the ATC communication was also designed. This module can simulate:

- various background noises that last for longer periods of time and overlap with the speech
- short noises like the sound of the transceiver switching on/off, various on-board messages and beeps, etc.
- sudden or graduate changes in the signal volume and even the total signal outage where it replaces the signal with a noise

and many more.

The resulting synthetic speech produced by the TTS system build from the corpus recorded by the Taiwanese speaker was evaluated in several listening tests. Preliminary small-scale listening tests showed acceptable overall synthetic voice quality and very identifiable and realistic Mandarin accent. Especially when the synthetic speech is mixed with simulated typical radiocommunication and cockpit ambient noise, the result is very convincing. However, a relatively high number of problematic or even unintelligible synthetic speech segments (words) were observed. Therefore, more formal listening tests were carried out, with two independent listeners evaluating 500 testing sentences from the held-out data of the source sentence database. The listeners were instructed to note and localize all disturbing synthetic artifacts in the generated speech causing disfluencies, unintelligible segments or just subjectively uncomfortable phenomena. This procedure resulted into approx. 50 manual interventions into the speech corpus annotation and especially segmentation, showing that most of the problems were caused by local failures of the automatic phonetic segmentation algorithm.

After these interventions, the quality and intelligibility of the synthetic speech improved significantly, which means that our method of minimalistic domain-specific speech corpus recording of non-native English speakers seems to be practically useful for acquisition of a rich inventory of non-native synthetic pseudopilot voices in the ATC simulator. The demo of the TTS system (with actually more foreign accents of English than it is presented in this paper) can be found at `http://itblp.zcu.cz/tts/index.html`.

5. Conclusion

The paper presented the motivation and the process of creation of the speech corpora that can be used for developing ASR and TTS applications in the domain of air traffic control. The evaluation experiments of the ASR and TTS systems build upon the corpora described in this paper have shown that the presented language resource are indeed useful for practical application. Both system, evaluated within this paper as stand-alone systems, were also already incorporated into the full *aPP* dialogue system. Its description is outside the scope of this paper and can be found in (Šmídl et al., 2016) and (Stanislav et al., 2016). However, the interested reader can test it at `https://itblp.zcu.cz/app-demo`.

All the corpora described in the paper are available in the LINDAT/CLARIN repository (see Section 8. for individual links)

6. Acknowledgments

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic projects LINDAT/CLARIN (No. CZ.02.1.01/0.0/0.0/16_013/0001781) and PUNTIS (No. LO1506).

7. Bibliographical References

Jůzová, M. and Tihelka, D. (2014). Minimum text corpus selection for limited domain speech synthesis. In Petr Sojka, et al., editors, *Text, Speech and Dialogue*, volume 8655 of *Lecture Notes in Computer Science*, pages 398–407. Springer International Publishing.

Legát, M., Matoušek, J., and Tihelka, D. (2011). On the detection of pitch marks using a robust multi-phase algorithm. *Speech Communication*, 53(4):552–566.

Matoušek, J., Tihelka, D., and Romportl, J. (2008). Building of a speech corpus optimised for unit selection tts synthesis. In *LREC 2008, proceedings of 6th International Conference on Language Resources and Evaluation*, pages 1296–1299. ELRA.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an ASR corpus based on public domain audio books. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)*. IEEE.

Prcín, M., Müller, L., and Šmídl, L. (2002). Statistical based speech/non-speech detector with heuristic feature set. In *6th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2002) / 8th International Conference on Information Systems Analysis and Synthesis (ISAS 2002)*, pages 264–269, Orlando, FL.

Stanislav, P., Šmídl, L., and Švec, J. (2016). An automatic training tool for air traffic control training. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 782–783.

Valenta, T. and Šmídl, L. (2015). Webtransc — a WWW interface for speech corpora production and processing. In Andrey Ronzhin, et al., editors, *Speech and Computer: 17th International Conference, SPECOM 2015, Athens, Greece, September 20-24, 2015, Proceedings*, pages 487–494. Springer International Publishing, Cham.

Šmídl, L., Chýlek, A., and Švec, J. (2016). A multimodal dialogue system for air traffic control trainees based on discrete-event simulation. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 379–380.

8. Language Resource References

John J. Godfrey. (1994). *Air Traffic Control Complete*. Linguistic Data Consortium – LDC, 1.0, ISLRN 367-677-522-995-8.

Jindřich Matoušek, Daniel Tihelka. (2014a). *English TTS speech corpus of air traffic (pilot) messages - Serbian accent*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11234/1-1462.

Jindřich Matoušek, Daniel Tihelka. (2014b). *English TTS speech corpus of air traffic (pilot) messages - Taiwanese accent*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11234/1-1461.

Jindřich Matoušek, Daniel Tihelka. (2015a). *English TTS speech corpus of air traffic (pilot) messages - Czech accent*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11234/1-1587.

Jindřich Matoušek, Daniel Tihelka. (2015b). *English TTS speech corpus of air traffic (pilot) messages - German accent*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11234/1-1588.

Luboš Šmídl. (2011). *Air Traffic Control Communication*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11858/00-097C-0000-0001-CCA1-0.

Luboš Šmídl. (2013). *ATCC: Pronunciation lexicon and n-gram counts for ASR module*. University of West Bohemia, distributed via LINDAT/CLARIN, IT-BLP resources, 1.0, ISLRN http://hdl.handle.net/11858/00-097C-0000-000D-EC92-F.

A First South African Corpus of Multilingual Code-switched Soap Opera Speech

Ewald van der Westhuizen, Thomas Niesler
Department of Electrical and Electronic Engineering
Stellenbosch University, South Africa
ewaldvdw@sun.ac.za, trn@sun.ac.za

Abstract

We introduce a speech corpus containing multilingual code-switching compiled from South African soap operas. The corpus contains English, isiZulu, isiXhosa, Setswana and Sesotho speech, paired into four language-balanced subcorpora containing English-isiZulu, English-isiXhosa, English-Setswana and English-Sesotho. In total, the corpus contains 14.3 hours of annotated and segmented speech. The soap opera speech is typically fast, spontaneous and may express emotion, with a speech rate that is between 1.22 and 1.83 times higher than prompted speech in the same languages. Among the 10343 code-switched utterances in the corpus, 19207 intrasentential language switches are observed. Insertional code-switching with English words is observed to be most frequent. Intraword code-switching, where English words are supplemented with Bantu affixes in an effort to conform to Bantu phonology, is also observed. Most bigrams containing code-switching occur only once, making up between 64% and 92% of such bigrams in each subcorpus.

Keywords: code-switching, spontaneous speech, South African languages, isiZulu, isiXhosa, Setswana, Sesotho

1. Introduction

South Africa has 11 official languages and since the majority of South Africans are multilingual, code-switching occurs commonly and spontaneously. Code-switching is the phenomenon of using more than one language within the same conversation or utterance (Van Dulm, 2007). Language switches may even occur mid-word, as examples in our data will demonstrate.

In an effort to be representative and accommodate a large viewer base, conversations in South African soap operas are multilingual and exhibit code-switching. Soap operas are therefore an interesting source of multilingual code-switched speech data. The corpus we describe in this paper displays various types of code-switching. Intersentential code-switching occurs when the language alternates between utterances of a conversation. For example, a speaker may utter a sentence in English while the following sentence is spoken in isiZulu. Intrasentential code-switching occurs when the language alternates within a single utterance. For example, an utterance may start in English and mid-sentence switch to isiZulu. Intrasentential code-switching can further be divided into alternational and insertional switching. Alternational code-switching occurs when an utterance starts with a phrase in one language and switches to a phrase in another language and where the phrases conform to the grammar of the respective languages. Insertional code-switching occurs when a language element from a secondary language is embedded into the structure of a matrix language. Code-switching can also occur at morpheme-level within word boundaries (intraword). Examples of intraword switching from our corpus include the joining of Bantu language class affixes to English stems.

Currently our corpus contains 14.3 hours of language-balanced speech compiled from soap opera broadcasts. Four Bantu languages are paired with English to yield subcorpora containing English-isiZulu, English-isiXhosa, English-Setswana and English-Sesotho code-switched speech. IsiZulu and isiXhosa are part of the Nguni language family, while Setswana and Sesotho are part of the Sotho-Tswana language family. Hence these five languages represent an interesting basis for studies into acoustic and language modelling for code-switched speech. The corpus is planned to be made available for research use.

2. Background

A number of code-switched corpora have already been described in the literature. Below we summarise a few notable examples. The list is however not intended to be exhaustive.

1. The SEAME corpus from Nanyang Technological University, Singapore, and Universiti Sains Malaysia, contains 63 hours of spontaneous Mandarin-English code-switched conversational and interview speech uttered by Malaysian and Singaporean speakers (Vu et al., 2012; Dong et al., 2010; Adel et al., 2015).
2. The HKUST Mandarin-English Corpus from the Hong Kong University of Science and Technology (Li et al., 2012; Li and Fung, 2013) consists of code-switched spontaneous speech from meetings and interviews and comprises 5 hours of transcribed and 15 hours of untranscribed speech.
3. The CECOS Chinese-English Corpus was compiled at the National Cheng Kung University in Taiwan (Shen et al., 2011). It contains 12 hours of speech collected from 77 speakers uttering prompted code-switch sentences.
4. The CUMIX Cantonese-English speech corpus (Chan et al., 2005) was compiled at The Chinese University of Hong Kong. It contains 17 hours of code-switched speech read by 80 speakers.
5. A small English-Spanish Corpus was compiled at the University of Texas. The corpus contains 40 minutes of transcribed spontaneous conversations with a vocabulary of 1516 words (Franco and Solorio, 2007).



Figure 1: A screenshot of ELAN media annotation tool. Only four annotation tiers are shown due to limited space.

6. A Frisian-Dutch corpus of radio broadcast speech was compiled at Radboud University, Nijmegen, containing 18.5 hours of speech with code-switching (Yılmaz et al., 2016).
7. A corpus of Sepedi-English code-switched speech was compiled by the South African CSIR (Modipa et al., 2013). It contains 10 hours of prompted speech, sourced from radio broadcasts and read by 20 Sepedi speakers.

The list above demonstrates that data collection for code-switched speech has focussed mostly on English with Asian languages. Recently this has been extended by a European language pair (Frisian-Dutch). Only one corpus containing an African language could be found. Our corpus extends this field by providing spontaneous data in several African languages, some of which are related.

3. Data Collection

The corpus is compiled from digital video recordings of 626 South African soap opera episodes. Mono audio sampled at 32kHz and coded as 16-bit PCM was extracted from the original source videos for each episode. The ELAN media annotation tool (Wittenburg et al., 2006) was used to segment the audio into segments corresponding to sentences. Figure 1 shows how ELAN displays the audio waveform and annotation tiers of a transcribed code-switched utterance. Sentences containing code-switching were further subdivided into segments corresponding to a single language. Such language segments were transcribed and tagged with a language label. Hence the following annotation tiers were used:

- monolingual sentence text,
- monolingual sentence language,
- code-switch segment text,
- code-switch segment language, and
- speaker name or ID.

Comparisons between the soap opera scripts and the uttered speech showed that the actors display a strong tendency to ad-lib. Therefore the speech can be regarded as spontaneous. Actors who code-switch usually do so between English and their Bantu mother tongue. Nevertheless, many utterances do contain more than two languages. These have been excluded from the current corpus, which focusses on code-switching in language pairs.

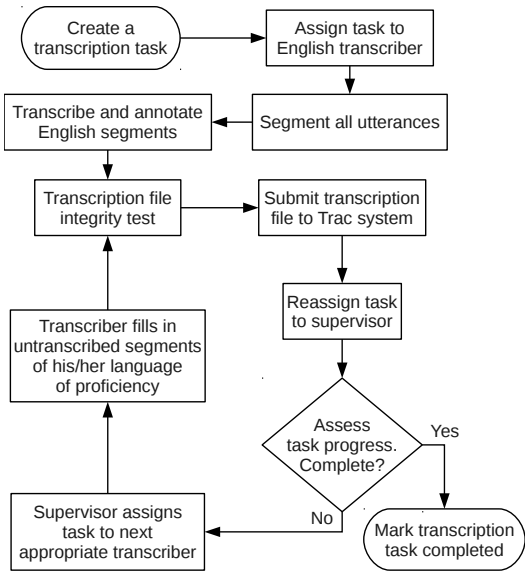


Figure 2: Flowchart depicting the transcription process for a single soap opera episode.

All speech was transcribed by fluent bilingual speakers. The Trac software project management, wiki and bug tracking system (Edgewall Software, 2012) was used to manage the tasks assigned to the team of transcribers. The flowchart in Figure 2 shows the steps involved in transcribing the speech of a single episode. A transcription task is created and first assigned to the principal transcriber. The principal transcriber performs the initial utterance segmentation and English transcription. Before a transcription file is submitted to the repository, a transcriber has to perform a transcription error and spell check. The transcription error check reports on common errors that occurred during transcription. For example, if transcription text was provided, but the segment language or speaker name was omitted, an error is flagged. After submission of the transcription file, the transcription task is reassigned to the supervisor to assess overall progress for the episode. If incomplete utterance segments still exist, the task is assigned to another transcriber with proficiency in the relevant languages. This continues until the supervisor confirms the episode to be fully transcribed and marks the task as complete. During this process, an episode is passed to at least one transcriber of each of the five languages, each annotating speech in their language of proficiency.

No overlapping speech was transcribed. Contracted forms of words were transcribed as closely to the speaker's actual pronunciation as possible. Contracted pronunciations are widespread in the Bantu languages where postlexical deletion regularly occurs in fast spoken spontaneous speech (Van der Westhuizen and Niesler, 2016). Using the example shown in Figure 1, the word *kukhona* is transcribed as *k'khon'*, with the apostrophes indicating the deletion of the *u* and *a* in the pronunciation.

All transcriptions are made in lower case, except for proper nouns and the English word 'I'. Since the language label for a proper noun can be ambiguous, the following rule was applied. Proper nouns are assigned the language label of

English-isiZulu												
set	engtok	zultok	tottok	engtyp	zultyp	tottyp	utt.cnt	emdur	zmdur	ecdur	zsdur	dur
train	28033	24350	52383	3608	6765	10373	8381	1.55h	1.55h	45.86m	56.99m	4.81h
dev	838	734	1572	415	443	858	225	0.0	0.0	4.01m	3.96m	8m
test	2459	3199	5658	871	1420	2291	768	0.0	0.0	12.76m	17.85m	30.4m
Total	31330	28283	59613	3842	7425	11269	9374	1.55h	1.55h	1.04h	1.31h	5.45h
English-isiXhosa												
set	engtok	xhotok	tottok	engtyp	xhotyp	tottyp	utt.cnt	emdur	xmdur	ecdur	xcdur	dur
train	20332	12223	32555	2630	5108	7738	6996	65.22m	53.55m	18.04m	23.73m	160.54m
dev	1155	1149	2304	484	764	1248	554	2.86m	6.48m	2.21m	2.13m	13.68m
test	1151	1501	2652	500	899	1399	391	0.0	0.0	5.56m	8.78m	14.34m
Total	22638	14873	37511	2829	5997	8826	7941	68.08m	60.03m	25.81m	34.64m	3.143h
English-Setswana												
set	engtok	tsntok	tottok	engtyp	tsntyp	tottyp	utt.cnt	emdur	tmdur	ecdur	tcdur	dur
train	16155	19570	35725	2360	1450	3810	5290	40.4m	30.96m	34.37m	34.01m	139.74m
dev	1168	2539	3707	513	539	1052	517	0.76m	4.26m	4.54m	4.27m	13.83m
test	1960	2979	4939	728	526	1254	496	0.0	0.0	8.87m	8.96m	17.83m
Total	19283	25088	44371	2606	1627	4233	6303	41.16m	35.22m	47.78m	47.24m	2.86h
English-Sesotho												
set	engtok	sottok	tottok	engtyp	sottyp	tottyp	utt.cnt	emdur	smdur	ecdur	scdur	dur
train	15372	19825	35197	2253	2086	4339	5659	49.34m	35.32m	23.02m	34.04m	141.72m
dev	840	2227	3067	436	614	1050	475	1.09m	5.05m	3.03m	3.59m	12.77m
test	1789	2265	4054	658	535	1193	429	0.0	0.0	7.80m	7.74m	15.54m
Total	18001	24317	42318	2518	2437	4955	6563	50.43m	40.37m	33.85m	45.37m	2.83h

Table 1: Subcorpora statistics. (train; Training set; dev: Development set; test: Evaluation test set; eng: English; zul: isiZulu; xho: isiXhosa; tsn: Setswana; sot: Sesotho; tot: Total; tok: tokens; typ: types; utt.cnt: Utterance count; emdur: Duration of English monolingual sentences; zmdur: Duration of isiZulu monolingual sentences; xmdur: Duration of isiXhosa monolingual sentences; tmdur: Duration of Setswana monolingual sentences; smdur: Duration of Sesotho monolingual sentences; ecdur: Duration of English code-switched segments; zcdur: Duration of isiZulu code-switched segments; xcdur: Duration of isiXhosa code-switched segments; tcdur: Duration of Setswana code-switched segments; scdur: Duration of Sesotho code-switched segments; dur: Total speech duration.)

English-isiZulu utt.cnt			English-isiXhosa utt.cnt			English-Setswana utt.cnt			English-Sesotho utt.cnt		
em	zm	ezcs	em	xm	excs	em	tm	etcs	em	sm	escs
1226	4364	3784	3372	2877	1692	1790	1827	2686	2493	1889	2181

Table 2: Monolingual and code-switched utterance counts for each full subcorpus. (utt.cnt: utterance count; em: English monolingual; zm: isiZulu monolingual; ezcs: English-isiZulu code-switched; xm: isiXhosa monolingual; excs: English-isiXhosa code-switched; tm: Setswana monolingual; etcs: English-Setswana code-switched; sm: Sesotho monolingual; escs: English-Sesotho code-switched)

the preceding word, e.g.:
`w1_en w2_en` **`name_en`** `w1_zu w2_zu`,
where `_en` and `_zu` are the English and isiZulu language labels, respectively.

Many short English function words, such as *but*, *why* and *if* occur very frequently in Bantu speech. It is tedious and time-consuming to mark all instances of such words as English. It could even be argued that such common English words have been adopted as loanwords by the Bantu languages. As a time-saving strategy during annotation, these words were either:

- labelled with the language they are embedded in, e.g. **`but_zu`** `w1_zu w2_zu w1_en w2_en`, or
- labelled as English when adjacent to an English word, `w1_zu w2_zu` **`but_en`** `w1_en w2_en`.

Since the set of English function words in question is closed, these words can easily and unambiguously be identified in the transcriptions.

4. Data Analysis

Tables 1 and 2 show statistics for the current version of our corpus. For each subcorpus, the tables show the speech durations, word token, type and utterance counts as well as the divisions into training, development and test sets. Since the transcription process is on-going, these statistics are subject to change in future and will be updated accordingly.
Table 3 shows the number of times the language switches in the corpus. Switching to English occurs slightly more often than switching from English to another language. We can consequently reason that, for our data, there is a slightly higher probability of an imminent switch when the current language of use is not English.

Subcorpus	Switch.dir	Count
engzul	eng→zul	3099
	zul→eng	3717
engxho	eng→xho	1195
	xho→eng	1479
engtsn	eng→tsn	2600
	tsn→eng	2728
engsot	eng→sot	2109
	sot→eng	2280
Total		19207

Table 3: Number of intrasentential language switches for full subcorpora. (Switch.dir: direction of language switch)

seg.len	EZ		EX		ET		ES	
	eng	zul	eng	xho	eng	tsn	eng	sot
1	2776	1859	1030	774	2019	766	1707	680
2	782	1502	294	655	644	804	500	674
3	388	988	163	419	379	631	285	553
4	280	534	118	226	265	555	196	447
5	214	292	101	113	177	378	131	300
6	168	186	96	69	132	308	83	255
7	138	90	69	37	102	213	55	151
8	84	48	46	19	70	126	49	110
9	51	27	36	9	53	101	25	88
> 10	144	49	79	13	118	173	51	229

Table 4: Segment lengths (seg.len) in number of words for monolingual segments of code-switched sentences against frequency count per language across each of the full subcorpora. (EZ: English-isiZulu; EX: English-isiXhosa; ET: English-Setswana; ES: English-Sesotho)

Next, we consider the length of monolingual segments in code-switched sentences. Segment lengths are measured in terms of number of word tokens. Table 4 tabulates the frequency of segments of a particular length. For example, there are 2776 instances of single-word English segments among the 3784 code-switched sentences in the English-isiZulu subcorpus. Single-word English segments are the most frequent within each subcorpus and typically represent insertional code-switching. English segments consisting of two or more words are much less frequent. The insertion of an English word into a Bantu matrix language is therefore the most common form of code-switching in our corpus. For the Bantu languages, there is a more gradual decline in frequency as the segment length increases.

Table 5 shows the training set occurrence counts of bigrams containing code-switching for the four subcorpora. The majority of code-switched bigrams occur only once. This is least pronounced for eng→tsn, where 1210 of 1878 (64%) such bigrams occur only once, while it is most pronounced for eng→xho, where 719 of the 784 (92%) occur only once. The most frequent eng→zul code-switched bigram occurs 31 times. These figures emphasise the sparsity of the data with respect to code-switch events. This will pose a challenge in terms of their effective modelling. In particular, since the majority of code-switch bigrams occur only once in the training data, it will in general not be possible to model code-switch events occurring in new and unseen data directly from training data examples (Van der Westhuizen and Niesler, 2017).

From English to isiZulu			
Trigger	Count	Target	Count
and	128	-a	179
no	94	ukuthi	158
so	92	-e	70
but	73	mina	52
sure	70	yakho	44
if	70	i-	43
for	43	-i	39
then	42	yami	38
why	41	wena	37
okay	39	nje	37
From isiZulu to English			
Trigger	Count	Target	Count
i-	465	and	98
u-	205	right	88
ama-	145	for	66
ukuthi	119	I	61
e-	116	sure	60
ngi-	83	you	57
le	64	but	39
ne-	55	so	33
uku-	39	understand	31
kwi-	39	or	30

Table 6: Token counts for English and isiZulu trigger and target types.

A code-switched bigram consists of two tokens. We define the first token as the trigger and the second as the target. Tables 6 to 9 show the most frequent word types serving either as a trigger or target in the subcorpora. Triggers in the Bantu languages include prefixes which join to English stems. Conversely, Bantu targets include suffixes similarly joining to English stems. Such affixes serve to supplement the pronunciation of an English word to conform to Bantu phonology and was also observed by (Modipa et al., 2013). An atypical trigger is the Sesotho suffix '-a' (Table 9) which precedes a switch to English. Examples of such cases are shown in the following two sentence fragments:

`u_so interview_en` **`-a_so`** `a_en while_en ago_en`

and

`o_so na_so push_en` **`-a_so`** `for_en this_en.`

In both cases the Sesotho suffix `-a` follows an English stem and precedes a switch to an English word, sandwiching the suffix between two English units.

Finally, Table 10 shows the speech rates calculated from aligned phone-level transcriptions for our corpus as well as for prompted speech in the same languages (Barnard et al., 2014). We see that the spontaneous soap opera speech is notably faster than the prompted speech in all cases.

5. Conclusion

We have compiled a 14.3 hour corpus of spontaneous multilingual code-switched speech from South African soap opera broadcasts. The corpus contains four language-balanced subcorpora: English-isiZulu, English-isiXhosa, English-Setswana and English-Sesotho. Many interesting

Bigram occurrence count	1	2	3	4	5	6	7	8	9	10	11	13	14	15	26	31	Total # bgtok
eng→zul #bgs	1838	87	26	8	7	3	3	0	1	0	0	0	0	0	0	1	2236
zul→eng #bgs	2064	163	38	14	9	6	1	0	2	1	0	2	0	1	1	0	2743
eng→xho #bgs	719	18	7	2	0	0	0	0	0	0	0	0	0	0	0	0	784
xho→eng #bgs	916	35	4	2	1	0	0	0	0	0	0	0	0	0	0	0	1011
eng→tsn #bgs	1210	137	49	16	10	5	5	2	2	2	0	0	1	0	0	0	1878
tsn→eng #bgs	1527	132	22	5	5	3	3	0	0	1	0	0	0	0	0	0	1951
eng→sot #bgs	1083	98	23	17	12	1	4	0	1	2	0	1	0	0	0	0	1552
sot→eng #bgs	1333	108	27	7	1	5	1	1	0	0	1	0	0	0	0	0	1719

Table 5: Occurrence counts of code-switch bigrams for the subcorpora training sets. (#bgs: number of bigrams; bgtok: bigram tokens.)

From English to isiXhosa			
Trigger	Count	Target	Count
and	56	ukuba	34
so	32	yakho	27
mean	25	yam	22
look	24	wena	21
but	23	yakhe	16
right	22	kodwa	15
for	16	-a	15
no	15	i-	13
well	14	-e	12
then	14	ngoku	11
From isiXhosa to English			
Trigger	Count	Target	Count
i-	157	and	46
u-	44	right	41
ne-	30	I	31
nge-	22	you	30
la-	22	is	24
kwi-	21	so	21
le	19	family	21
e-	19	for	19
yi-	18	but	17
ukuba	16	please	14

Table 7: Token counts for English and isiXhosa trigger and target types.

From English to Setswana			
Trigger	Count	Target	Count
so	70	-a	234
if	62	ke	193
why	60	o	182
well	51	ya	178
and	51	-e	169
then	49	go	157
like	47	e	106
know	39	a	87
mean	38	le	61
for	38	nna	55
From Setswana to English			
Trigger	Count	Target	Count
di	209	you	61
le	178	and	52
ke	174	I	47
re	123	for	44
o	115	like	38
ka	79	right	32
e	75	a	29
mo	67	sure	28
ko	65	the	27
go	65	but	25

Table 8: Token counts for English and Setswana trigger and target types.

examples of code-switching are observed, with English word insertions the most frequent. However, the distribution of code-switch examples is sparse, which could make modelling of unseen code-switch events challenging. Intraword code-switching, where Bantu affixes are joined to English stems in an effort to change English words to conform with Bantu phonology, is also often observed. The corpus is useful in the study of code-switched automatic speech recognition, discourse and dialogue analysis. The corpus is planned to be made available for research use.

6. Acknowledgements

We would like to thank e.tv and Yula Quinn at Rhythm City for assistance with data compilation. We pay tribute to Dumi Masilela (29) who tragically passed away during the course of this work. He portrayed the character Sifiso Ngema.

7. Bibliographical References

Adel, H., Vu, N. T., Kirchhoff, K., Telaar, D., and Schultz, T. (2015). Syntactic and semantic features for code-switching factored language models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):431–440.

Barnard, E., Davel, M. H., Van Heerden, C., De Wet, F., and Badenhorst, J. (2014). The NCHLT speech corpus of the South African languages. In *Proceedings of the 4th International Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2014)*, St Petersburg, Russia, 14-16 May 2014.

Chan, J. Y. C., Ching, P. C., and Lee, T. (2005). Development of a Cantonese-English Code-Mixing Speech Corpus. In *Proceedings of Interspeech 2005*, Lisbon, Portugal, 4-8 September 2005.

Dong, M., Chan, P., Cen, L., Li, H., Teo, J., and Kua, P. J. (2010). SEAME: A Mandarin-English Code-Switching

From English to Sesotho			
Trigger	Count	Target	Count
if	50	-a	226
so	40	-e	127
maybe	36	ke	117
know	36	u	98
why	34	ya	88
well	29	ha	70
then	29	e	69
for	28	hore	67
mean	26	o	63
because	24	ea	59
From Sesotho to English			
Trigger	Count	Target	Count
di	175	and	47
ke	128	right	34
le	123	something	33
ka	78	you	30
u	75	I	27
mo	71	the	25
-a	62	like	25
ho	60	maybe	22
o	57	for	22
e	56	so	19

Table 9: Token counts for English and Sesotho trigger and target types.

Corpus	Language	Rate (ph/s)	
		Soap opera spontaneous	NCHLT prompted
EZ	English	13.09	10.62
	isiZulu	16.50	9.04
EX	English	15.38	–
	isXhosa	19.98	8.77
ET	English	13.37	–
	Setswana	16.29	10.39
ES	English	13.24	–
	Sesotho	16.03	9.86

Table 10: Speech rate in phones per second (ph/s) of spontaneous soap opera speech and prompted speech for English and isiZulu. (EZ: English-isiZulu; EX: English-isiXhosa; ET: English-Setswana; ES: English-Sesotho)

Speech Corpus in South-East Asia. In *Proceedings of Interspeech 2010*, Makuhari, Japan, 26-30 September 2010.

Edgewall Software. (2012). Trac: Integrated SCM & Project Management. `https://trac.edgewall.org/`, Version 0.12.3. Release date: 2012-02-06.

Franco, J. C. and Solorio, T. (2007). Baby-steps towards building a Spanglish language model. In Gelbukh A., editor, *Computational Linguistics and Intelligent Text Processing. CICLing 2007. Lecture Notes in Computer Science, vol 4394*, Berlin, Heidelberg. Springer.

Li, Y. and Fung, P. (2013). Improved mixed language speech recognition using asymmetric acoustic model and language model with code-switch inversion constraints. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, 26-31 May 2013.

Li, Y., Yu, Y., and Fung, P. (2012). A Mandarin-English Code-Switching Corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul, Turkey, 23-25 May 2012.

Modipa, T. I., Davel, M. H., and de Wet, F. (2013). Implications of Sepedi/English code switching for ASR systems. In *Proceedings of the Twenty-Fourth Annual Symposium of the Pattern Recognition Association of South Africa (PRASA 2013)*, Johannesburg, South Africa, 3 December 2013.

Shen, H.-P., Wu, C.-H., Yang, Y.-T., and Hsu, C.-S. (2011). CECOS: A Chinese-English code-switching speech database. In *Proceedings of the 2011 International Conference on Speech Database and Assessments (Oriental COCOSDA 2011)*, Hsinchu City, Taiwan, 26-28 October 2011.

Van der Westhuizen, E. and Niesler, T. R. (2016). The effect of postlexical deletion on automatic speech recognition in fast spontaneously spoken Zulu. In *Proceedings of Interspeech 2016*, San Francisco, CA, 8-12 September 2016.

Van der Westhuizen, E. and Niesler, T. R. (2017). Synthesising isiZulu-English code-switch bigrams using word embeddings. In *Proceedings of Interspeech 2017*, Stockholm, Sweden, 20-24 August 2017.

Van Dulm, O. (2007). *The grammar of English-Afrikaans code switching: A feature checking account.* LOT, Utrecht, The Netherlands.

Vu, N. T., Lyu, D.-C., Weiner, J., Telaar, D., Schlippe, T., Blaicher, F., Chng, E.-S., Schultz, T., and Li, H. (2012). A first speech recognition system for Mandarin-English code-switch conversational speech. In *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2012)*, Kyoto, Japan, 25-30 March 2012.

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). ELAN: a Professional Framework for Multimodality Research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, 24-26 May 2006.

Yılmaz, E., Andringa, M., Kingma, S., Dijkstra, J., Van der Kuip, F., Van de Velde, H., Kampstra, F., Algra, J., Van den Heuvel, H., and Van Leeuwen, D. (2016). A longitudinal bilingual Frisian-Dutch radio broadcast database designed for code-switching research. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz, Slovenia, 23-28 May 2016.

A Web Service for Pre-segmenting Very Long Transcribed Speech Recordings

Nina Poerner*, **Florian Schiel**†
*Center for Information and Language Processing, University of Munich, Germany
†Bavarian Archive for Speech Signals, University of Munich, Germany
poerner@cis.uni-muenchen.de, schiel@bas.uni-muenchen.de

Abstract

The run time of classical text-to-speech alignment algorithms tends to grow quadratically with the length of the input. This makes it difficult to apply them to very long speech recordings. In this paper, we describe and evaluate two algorithms that pre-segment long recordings into manageable "chunks". The first algorithm is fast but cannot guarantee short chunks on noisy recordings or erroneous transcriptions. The second algorithm reliably delivers short chunks but is less effective in terms of run time and chunk boundary accuracy. We show that both algorithms reduce the run time of the MAUS speech segmentation system to under real-time, even on recordings that could not previously be processed. Evaluation on real-world recordings in three different languages shows that the majority of chunk boundaries obtained with the proposed methods deviate less than 100 ms from a ground truth segmentation. On a separate German studio quality recording, MAUS word segmentation accuracy was slightly improved by both algorithms. The chunking service is freely accessible via a web API in the CLARIN infrastructure, and currently supports 33 languages and dialects.

Keywords: text-to-speech alignment, speech segmentation, speech processing, MAUS

1. Introduction

Text-to-speech alignment plays an important role in Phonetic research, speech technology, and in the production of video subtitles. The Bavarian Archive for Speech Signals offers a free web service, Munich AUtomatic Segmentation (MAUS) (Schiel, 1999), to the scientific community. MAUS segments transcribed speech recordings into words and phones. Its main advantage is that, for many of its 33 languages and dialects, it cannot only perform forced alignment, but also model variable pronunciation. MAUS has recently been evaluated as the best aligner, out of ten candidates, on a recording by J.F. Kennedy (Oesch and Sidler, 2017).

One major disadvantage of MAUS is its run time on long recordings. This is because MAUS performs optimal Viterbi alignment, which, while more accurate than beam search, has quadratic run time with signal duration and transcription length (see Figure 1). Long run times mainly affect users who wish to use MAUS on certain types of real-world recordings, such as public speeches, interviews or audio books, as these tend to be longer than audio material produced for scientific research.

One way to combat MAUS's run time problem is to pre-segment long recording-transcription pairs into short "chunks", i.e., slices of matching audio and text material of a few seconds or minutes. Recently, we introduced the "chunker", an algorithm that does this job automatically (Poerner and Schiel, 2016). The chunker has since been made available as a web service by the Bavarian Archive for Speech Signals (Kisler et al., 2017).

In the following, we give an overview of the algorithm, including a number of new developments (Section 2.). Sections 3. and 4. describe an evaluation on three real-world educational and political recordings and a studio quality recording.

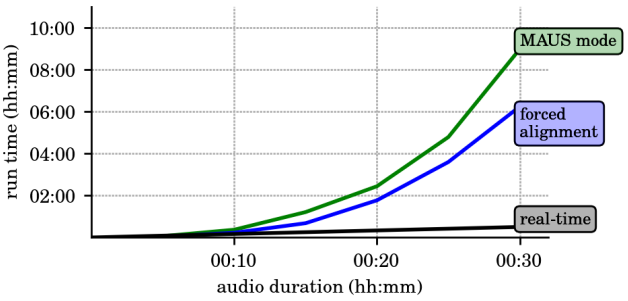


Figure 1: Run time of MAUS text-to-speech alignment with pronunciation modeling ("MAUS mode") and with simple forced alignment. Inputs are subsequences of the material used in Section 3.1.4.

2. Algorithms

The chunker offers two different technical solutions to the chunking problem: the so-called "standard" algorithm (default) and the "forced" algorithm.

2.1. Standard chunking algorithm

The standard chunking algorithm is a combination of the T (token-based) and P (phoneme-based) chunking algorithms presented in Poerner and Schiel (2016). The T algorithm builds on work by Moreno et al. (1998). It can be divided into the following steps:

Language model – A smoothed HTK (Young et al., 2006) bi-gram language model is trained on the transcription.

Recognition – The HTK-HVite speech recognition engine recognizes the signal, using the bi-gram model as well as language-specific acoustic models from MAUS.

Alignment – The recognized text is symbolically aligned with the transcription using the Hirschberg algorithm

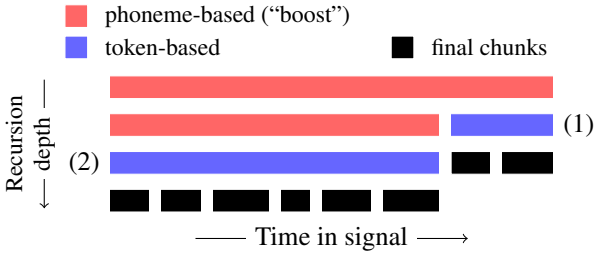


Figure 2: Schematic representation of standard algorithm with boost phase. (1) Switch from P to T chunker because chunk is short enough. (2) Switch from P to T chunker because P failed to find more boundaries.

(Hirschberg, 1975). Alternatively, users can opt for a faster, approximate algorithm with linear run time, similar to one proposed by Anguera et al. (2011).

Boundary selection – After symbolic alignment, the chunker searches for “anchor” regions where the transcription and the recognition result are perfectly aligned for a minimum number of words (e.g., 3). The signal and transcription are then split at a word boundary inside this region. The chunker prioritizes inter-word pauses.

Recursion – Unless all chunks are short enough, or no new boundaries were found, the entire process is repeated recursively on the discovered chunks.

Speech recognition run time grows with the duration of the recording and with the number of unique word types in the transcription. Hence, the run time of the T chunker – while faster than MAUS – was found to be unacceptable on recordings that exceed one hour.

The P chunker algorithm, which is based on work by Bordel et al. (2012), alleviates this problem. Its recognition engine and symbolic alignment run on the phoneme level instead of the word level. Since phoneme inventories are limited, recognition can thus be done in linear time. On the downside, the P chunker finds fewer boundaries and is less successful at locating inter-word pauses (Poerner and Schiel, 2016).

The “standard” algorithm combines the speed of the P chunker with the accuracy of the T chunker. More specifically, the P chunker is used in a so-called “boost phase” to break the signal into relatively long chunks (duration > 1 minute, red bars in Figure 2). After the boost phase, the T chunker does the fine-grained chunking (blue bars in Figure 2). In cases where the P chunker fails to find a sufficient number of boundaries, the T chunker is called as a back-up.

2.2. Forced chunking algorithm

There are cases where the standard algorithm fails to find a sufficient number of anchor regions, resulting in chunks that are too long for further processing. While we have found that this often indicates bad signal and/or transcription quality (which the user may want to deal with anyway), we decided to develop a second algorithm that guarantees a chunking result even in problematic cases.

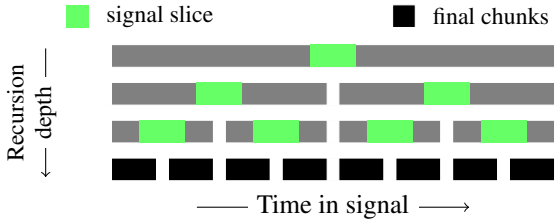


Figure 3: Schematic representation of the forced algorithm.

The forced chunking algorithm builds on a method of Moreno and Alberti (2009). They describe a factor automaton language model that is able to find any substring of a given text in the signal. We use this strategy in a divide-and-conquer way (see Figure 3):

Language model – A factor automaton language model is built on the transcription.

Recognition – A slice (e.g., 2 minutes) is cut from the middle of the signal. It is recognized by HTK-HVite using the factor automaton model. Since recognition with a factor automaton produces an alignment as a side-effect, no symbolic alignment is necessary.

Boundary selection – A word boundary from the recognized slice is used as a chunk boundary. As with the standard algorithm, inter-word pauses are prioritized.

Recursion – The process is recursively repeated on the resulting halves of the signal.

The size of the factor automaton grows linearly with the length of the transcription, and the necessary recursion depth grows logarithmically with audio duration. Hence, the forced algorithm tends to be slower than the (boosted) standard algorithm.

3. Evaluation

3.1. Data

We tested the chunker system on three real-world recordings of German, Italian and American English, whose durations range between 58 and 89 minutes, as well as on a 55 minute “pseudo recording” made up of concatenated studio quality audio clips (see Table 1 for details). MAUS on its own failed to produce a segmentation for any of these recordings within 48 hours (at which point the processes had to be interrupted).

3.1.1. Lecture recording (German)

The German real-world recording is a university lecture on Aristotelian philosophy from the euroWiss project (Heller et al., 2013)[1]. The lecture is held by a male native German speaker, and there are short contributions by students. Audio quality is high for the main speaker, but other speakers are difficult to understand as they speak far away from the microphone. There are frequent instances of background noises and reverberations. The signal was downloaded as a

[1] hdl.handle.net/11022/0000-0001-7DBA-2

Recording	Signal duration	Transcription # words	Standard algorithm		Forced algorithm	
			Chunker run time	MAUS run time	Chunker run time	MAUS run time
Lecture	88:27	11969	41:44	23:60	165:59	24:10
Seminar	73:47	9046	23:55	15:18	100:00	18:30
Address	58:29	7562	23:10	12:11	60:01	13:33
Concat	55:08	8231	25:38	16:21	76:00	16:29

Table 1: Signal duration, number of words, chunker run time and subsequent MAUS run time in minutes

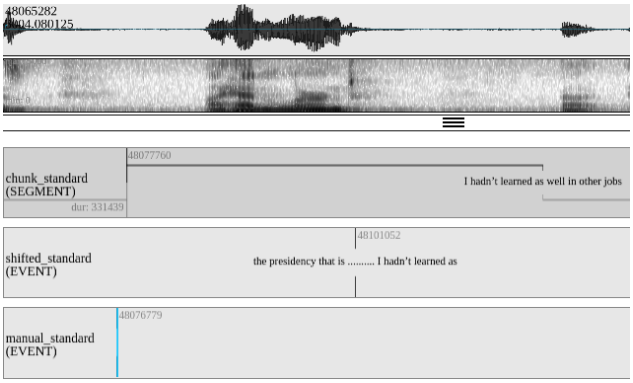


Figure 4: Manually annotated chunk boundary. Annotation levels from top to bottom: chunker prediction (invisible during annotation), randomly shifted prediction (visible during annotation), manually annotated boundary.

stereo MP3 file and converted into a mono 16 kHz WAVE file. The orthographic transcription was extracted from the accompanying EXMARaLDA annotation. The transcription is narrow and includes hesitations and false starts. We deliberately did not use the chunk segmentation provided.

3.1.2. Seminar recording (Italian)

The Italian real-world recording also stems from the euroWiss project. It contains a seminar on the author Walter Benjamin, held by a female native Italian speaker, again with short contributions by students. Remarks made on the audio and transcription quality of the German recording apply to the Italian recording too. Note that while the audio contains a transcribed and an untranscribed part, we only used the transcribed part (the first 73:47 minutes).

3.1.3. Address (American English)

The American English real-world recording is an address given by former US president Barack Obama (Obama, 2015)[2]. Most of the material is spoken by the main speaker, but there are short contributions by other speakers, including many non-native speakers. Audio quality is high for all speakers, but there are background noises and reverberations. The recording was downloaded as a mono MP3 file and converted into a mono 16 kHz WAVE file. The transcription was crawled from the corresponding web site. It is relatively broad, meaning that it does not contain hesitations and false starts. Furthermore, a number of sentences in the audio differ or are missing from the transcription.

[2]www.americanrhetoric.com/speeches/barackobama/barackobamaYSEALIWashingtonDC.htm

3.1.4. Concatenated recording (German)

The concatenated recording is made up of 401 clips from the German Verbmobil corpus (Burger et al., 2000)[3]. The clips are studio quality, 16 kHz mono WAVE. They contain speaker turns from timetabling role-plays (6 male, 16 female speakers), and they come with a manual broad phonetic segmentation. To mimic a long recording, the clips were concatenated into a single 55 minute file.

3.2. Processing

All recording-transcription pairs were converted into EMU databases (Winkelmann et al., 2017)[4]. We used the G2P[5] grapheme-to-phoneme converter (Reichel, 2012) to produce a phonemic (SAMPA) representation of every word in the transcription. In the next step, we ran the chunker in two different settings:

standard – Boost phase with anchor length 4 phonemes, followed by word-based phase with anchor length 3 words. Minimum chunk duration 5 seconds, alignment by Hirschberg algorithm.

forced – Forced algorithm with slice duration 2 minutes. Minimum chunk duration 5 seconds.

Finally, we used MAUS to produce a phonetic segmentation based on the outcomes of the two chunker variants. To compare the accuracy of MAUS with and without pre-chunking, we also ran it directly on the concatenated recording. Since MAUS on its own could not process the full 55 minutes in a reasonable amount of time, the recording was manually pre-segmented into ten minute chunks, effectively giving MAUS a head start over the chunker.

3.3. Ground truth

3.3.1. Word boundaries

In the German concatenated recording, the existing manual phonetic annotation was used to locate the ground truth start and end times of words. In the three real-world recordings, no such annotation exists; hence, word segmentation accuracy could not be evaluated on them.

3.3.2. Chunk boundaries

A predicted chunk boundary has the form (w_1, t, w_2), where w_1 and w_2 are the word tokens to the left and right of the boundary, and t is the predicted time. Its true time is

[3]hdl.handle.net/11022/1009-0000-0000-EB31-0, sessions m112d-m117d, m119d, m222d, m224d, m230d, m231d

[4]ips-lmu.github.io/EMU.html

[5]All tools were called using EMU's interface to the BAS web services (Poerner and Winkelmann, 2017). Hence, run times reported in Table 1 contain a slight overhead for signal uploads.

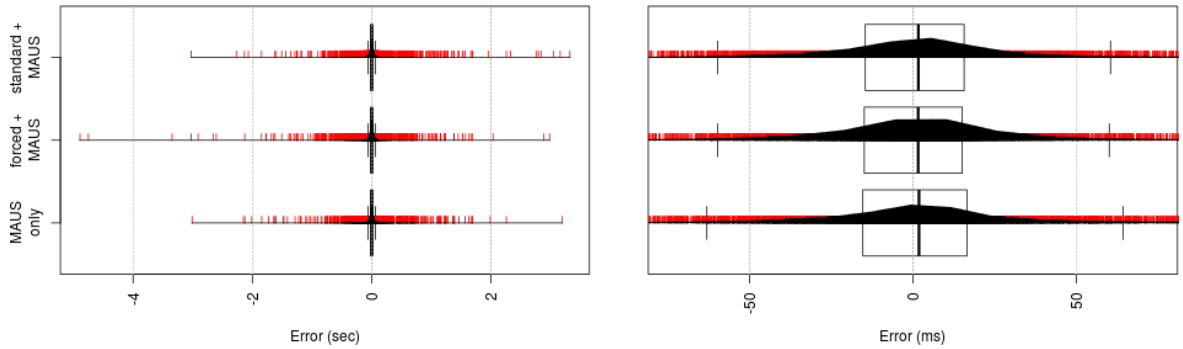


Figure 5: MAUS word segmentation errors relative to manual annotation, with and without pre-chunking, on concatenated German recording. Box plots with error distributions. Red dashes: Single data points. Full range (left) and zoomed view (right).

the mid-point between the true end time of w_1 and the true start time of w_2. If there is no pause between w_1 and w_2, end time, start time and mid-point coincide.
In the concatenated recording, true chunk boundaries were derived from the existing manual annotation. In the real-world recordings, true boundaries were annotated by the first author using the EMU-webApp (Winkelmann and Raess, 2014). To avoid confirmation bias, predicted boundaries were randomly shifted to the left or right by up to two seconds before annotation, while the actual predictions were hidden and later restored (see Figure 4).

3.4. Results

Chunk boundary errors are defined as the difference between predicted chunk boundary times and true chunk boundary times (see Section 3.3.2.). Figure 6 shows distributions of chunk boundary errors on all recordings. Word segmentation errors are defined as the difference between word start times according to MAUS and word start times according to the manual phonetic segmentation. Figure 5 shows distributions of word segmentation errors for the concatenated recording. Table 2 lists the percentages of chunk boundaries that were correctly placed inside inter-word pauses of duration 100 ms or more. Figure 8 shows distributions of chunk durations on all recordings.

4. Discussion

4.1. Run time

On all recordings, both chunking algorithms enabled MAUS to run in less than real-time (see Table 1). The standard chunking algorithm ran in under real-time too, while the forced algorithm took up to four times as long. Still, both methods reduced overall run time compared to MAUS-only segmentation, which took more than 48 hours.

4.2. Effects on word segmentation accuracy

Figure 5 suggests that MAUS word segmentation accuracy was not harmed by pre-chunking. Quite the contrary, segmentation accuracy slightly improved relative to the baseline. It is worth keeping in mind that word segmentation accuracy could only be evaluated on the studio quality recording, which might not be representative of real-world recordings.

	Lecture	Seminar	Address	Concat
Standard	58.4%	52.6%	51.5%	69.5%
Forced	75.6%	75.1%	65.1%	79.0%

Table 2: Percentage of chunk boundaries that were placed inside a pause of 100 ms or more between their previous and next word.

4.3. Finding pauses

As mentioned in Section 2.1., the chunker aims to place boundaries inside inter-word pauses to reduce the risk of cutting into the previous or following word. Table 2 shows that the forced algorithm was more successful at finding inter-word pauses than the standard algorithm, which is probably due to its greater choice of potential boundary locations.

4.4. Chunk durations

The standard algorithm produced chunks with a duration of up to 1:47 minutes (see Figure 8). While this is sufficiently short for fast MAUS processing, it suggests that there were regions where the standard method failed to find any boundaries. Manual inspection suggests that these regions often coincided with low-quality stretches in the recording (e.g., prolonged background noise, turns by speakers far away from the microphone, regions with audio-transcription mismatches).
The forced algorithm, on the other hand, returned shorter chunks, because it is forced to cut inside low-confidence regions.

4.5. Chunk boundary errors

The majority of absolute chunk boundary errors made by the standard algorithm were below 100 ms (see Figure 6). We revisited all absolute errors beyond 500 ms and found that the standard algorithm has a tendency to get confused whenever a speaker utters the same (or approximately the same) phrase several times in a row (e.g., *Ja,* ***wir wollen die*** *Nahrung,* ***wir wollen die*** *äh* ***wir wollen die*** *Ernährung*, German lecture). A possible explanation is that the recognition engine correctly recognizes one repetition but not the others, resulting in a situation where the recognized phrase gets aligned to the wrong instance in the transcription.

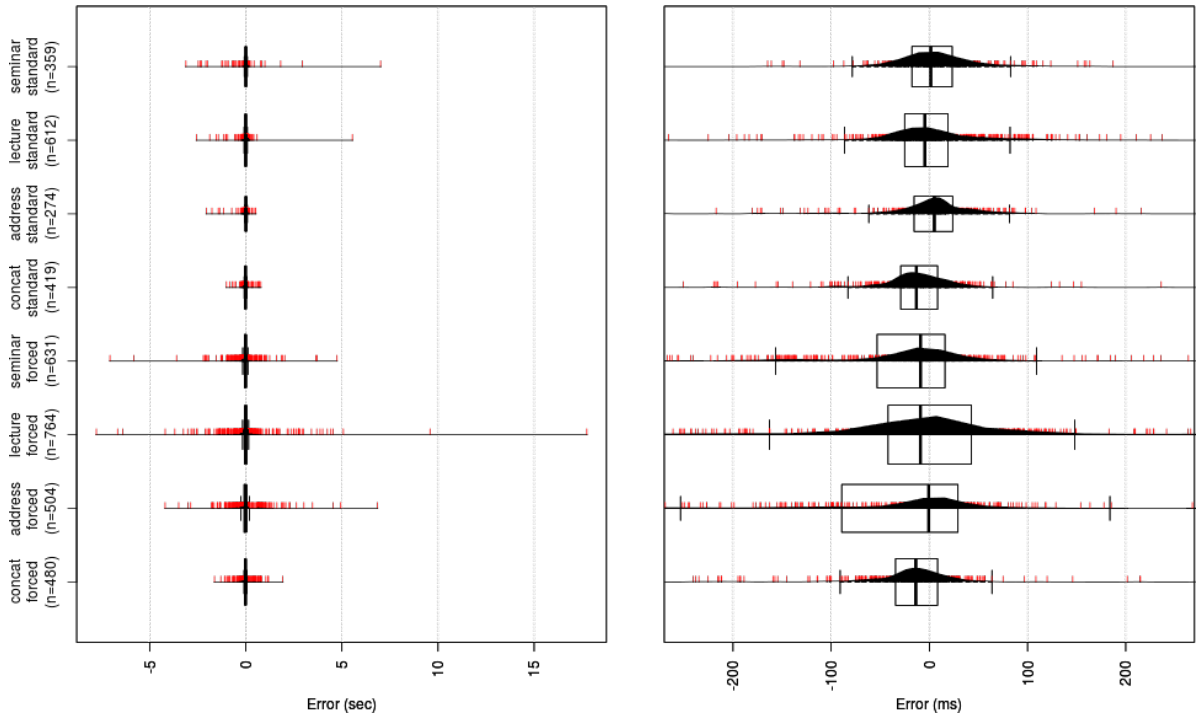


Figure 6: Chunk boundary errors relative to manual annotation. Box plots with error distributions. Red dashes: Single data points. Full range (left) and zoomed view (right).

While the forced algorithm guarantees short chunks, it turned out to be more prone to errors than the standard algorithm, with a greater range of error outliers and broader error distributions. Manual analysis of these outliers revealed that most of them come down to one of two patterns:

- The chunker misinterprets the above-mentioned low-quality regions, e.g., by "imagining" a subsequence of the transcription inside a long noisy pause. This kind of mistake can lead to major mis-alignments, especially if it happens early in the recursion.
- In a scenario where a pause is preceded or followed by a short or clipped word (e.g., *I* or *of*), the forced chunker sometimes locates that word on the wrong side of the pause (see Figure 7). The difference between predicted and true chunk boundary is half the duration of the pause, which may be several seconds. However, the error is less severe than its magnitude suggests, since all words except one end up in the correct chunk.

5. Summary

We have presented two algorithms that automatically break long transcribed speech recordings into "chunks", i.e., shorter text-audio pairs. They are intended as a pre-processing tool to the Viterbi-based phonetic segmentation system MAUS, which cannot cope with very long recordings on its own. The combination of chunker and MAUS allows researchers to align and segment very long speech recordings (e.g., interviews, speeches or audio books) quickly and without the need for time-consuming manual pre-segmentation.

The standard chunking algorithm applies a combination of fast phoneme-based and accurate word-based recognition. The forced chunking algorithm is slower and less accurate, but it guarantees a result even in the face of poor signal or transcription quality. Hence, it can be used as a fall-back in extreme cases where the standard algorithm fails to deliver chunks that are short enough.

We evaluated both algorithms on long recordings in three different languages. The comparison with a manual annotation showed that most chunk boundaries deviate from the ground truth by less than 100 ms. MAUS word segmentation accuracy does not seem to be negatively affected by pre-chunking; instead, there is a slightly positive impact.

Both algorithms are offered as free-to-use web services within the CLARIN infrastructure.[6]

[6]http://hdl.handle.net/11022/1009-0000-0001-232C-7

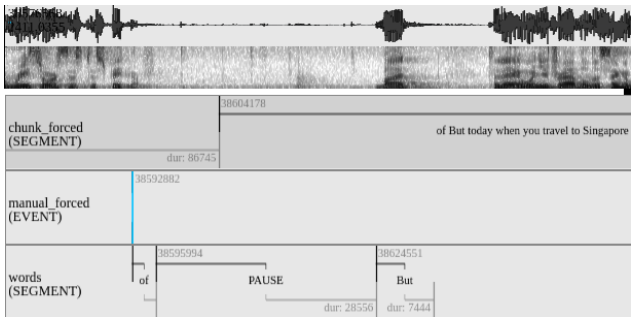


Figure 7: Forced algorithm locates short word on wrong side of a pause. Annotation levels from top to bottom: chunker prediction, manual boundary, manual locations of the words *of* and *But*, and their inter-word pause.

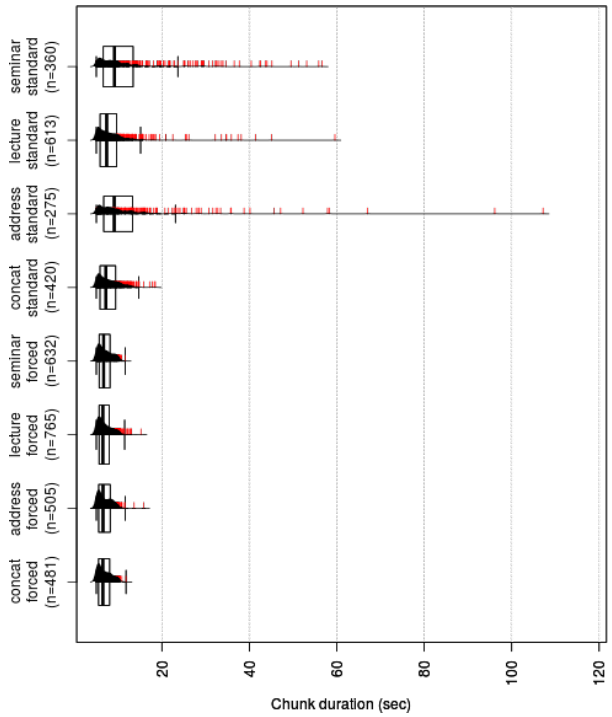


Figure 8: Chunk durations. Box plots with distributions. Red dashes: Single data points.

6. Bibliographical References

Anguera, X., Perez, N., Urruela, A., and Oliver, N. (2011). Automatic synchronization of electronic and audio books via TTS alignment and silence filtering. In *Proc. ICME*, Barcelona, Spain, July.

Bordel, G., Peñagarikano, M., Rodríguez-Fuentes, L. J., and Varona, A. (2012). A simple and efficient method to align very long speech signals to acoustically imperfect transcriptions. In *Proc. Interspeech*, pages 1840–1843, Portland, USA, September.

Hirschberg, D. (1975). A linear space algorithm for computing maximal common subsequences. *Communications of the ACM*, 18(6):341–343.

Kisler, T., Reichel, U., and Schiel, F. (2017). Multilingual processing of speech via web services. *Computer Speech & Language*, 45:326–347.

Moreno, P. J. and Alberti, C. (2009). A factor automaton approach for the forced alignment of long speech recordings. In *Proc. ICASSP*, pages 4869–4872, Taipei, Taiwan, April.

Moreno, P. J., Joerg, C. F., Van Thong, J.-M., and Glickman, O. (1998). A recursive algorithm for the forced alignment of very long audio segments. In *Proc. ICSLP*, pages 2711–2714, Sydney, Australia, December.

Oesch, J. and Sidler, A. (2017). Videx: Indexing videos for the "official bulletin" of the swiss federal parliament. *Haute école d'ingénierie et de gestion du canton de Vaud*, `www.jonasoesch.ch/portfolio/videx`.

Poerner, N. and Schiel, F. (2016). An automatic chunk segmentation tool for long transcribed speech recordings. In *Proc. Phonetik und Phonologie*, pages 144–146, Munich, Germany, October.

Poerner, N. and Winkelmann, R. (2017). Interfacing the BAS speech science web services and the EMU speech database management system. In *Proc. Phonetik und Phonologie*, Berlin, Germany, September.

Reichel, U. (2012). PermA and Balloon: Tools for string alignment and text processing. In *Proc. Interspeech*, pages 1874–1877, Portland, USA, September.

Schiel, F. (1999). Automatic phonetic transcription of non-prompted speech. In *Proc. ICPhS*, pages 607–610, San Francisco, USA, August.

Winkelmann, R. and Raess, G. (2014). Introducing a web application for labeling, visualizing speech and correcting derived speech signals. In *Proc. LREC*, pages 4129–4133, Reykjavik, Iceland, May.

Winkelmann, R., Harrington, J., and Jänsch, K. (2017). EMU-SDMS: Advanced speech database management and analysis in R. *Computer Speech & Language*, 45:392–410.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., and Povey, D. (2006). *The HTK Book*. Cambridge University.

7. Language Resource References

Burger, Susanne and Weilhammer, Karl and Schiel, Florian and Tillmann, Hans G. (2000). *Verbmobil data collection and annotation*. Springer. `hdl.handle.net/11022/1009-0000-0000-EB31-0`.

Heller, Dorothee and Hornung, Antonie and Redder, Angelika and Thielmann, Winfried. (2013). *euroWiss: linguistic profiling of European academic education*. Hamburger Zentrum für Sprachkorpora. `hdl.handle.net/11022/0000-0001-7DBA-2`.

Obama, Barack. (2015). *Southeast Asian Youth Initiative Fellows Address*. American Rhetoric. `www.americanrhetoric.com`.

A Real-life, French-accented Corpus of Air Traffic Control Communications

Estelle Delpech[1], Marion Laignelet[2], Christophe Pimm[2], Céline Raynal[2], Michal Trzos[3], Alexandre Arnold[1], Dominique Pronto[1]

[1]AIRBUS, Toulouse, France - [2]Safety Data-CFH, Toulouse, France - [3]HONEYWELL, Brno, Czech Republic

{estelle.e.delpech;alexandre.arnold;dominique.pronto}@airbus.com, {laignelet;pimm;raynal}@safety-data.com, michal.trzos@honeywell.com

Abstract

This paper describes the creation of the AIRBUS-ATC corpus, which is a real-life, French-accented speech corpus of Air Traffic Control (ATC) communications (message exchanged between pilots and controllers) intended to build a robust ATC speech recognition engine. The corpus is currently composed of 59 hours of transcribed English audio, along with linguistic and meta-data annotations. It is intended to reach 100 hours by the end of the project. We describe ATC speech specificities, how the audio is collected, transcribed and what techniques were used to ensure transcription quality while limiting transcription costs. A detailed description of the corpus content (speaker gender, accent, role, type of control, speech turn duration) is given. Finally, preliminary results obtained with state-of-the-art speech recognition techniques support the idea that accent-specific corpora will play a pivotal role in building robust ATC speech recognition applications.

Keywords: speech corpus, spoken language, controlled language, air traffic control phraseology, speech recognition, accented English.

1. Introduction

The AIRBUS-ATC corpus was developed in order to build a speech recognition system able to process Air Traffic Control (ATC) communications - messages exchanged between pilots and controllers using an English-based controlled language known as the ICAO phraseology (ICAO, 2007) - and Automatic Terminal Information Service messages (ATIS, airport information broadcast about weather conditions, available runways, etc.). The goal, in the long term, is to help pilots by giving them reliable ATC information in a persistent and visual way.

Pilot: Montana, F-CD, request cancel my IFR flight, proceeding VFR estimating Borton at 1701

Controller: F-CD, IFR flight cancelled at 35, contact Montanan Information 125.750

Figure 1: Example of ICAO phraseology.

ATC communications being very different from everyday conversations (see table 1), voluminous datasets like the SWITCHBOARD (Godfrey et al. 1992) and FISHER (Cieri et al. 2004) corpora cannot be used to build such a system although they were key in achieving human parity for conversational speech (Hannun et al. 2014; Xiong et al. 2016; Saon et al. 2017).

	SWB/FISHER	**ATC speech**
intelligibility	good (phone)	bad (radio transmission)
accents	US English	diverse and non-native
lexicon & syntax	oral syntax everyday topics	limited to ICAO phraseology or related
speech rate	standard	high (Cauldwell, 2007)
other	-	code switching

Table 1: SWB & FISHER corpora vs. ATC speech

The AIRBUS-ATC corpus intends to address this issue by providing a real-life corpus of transcribed English ATC messages spoken by non-native speakers.

2. Existing ATC speech corpora

Six ATC speech corpora were found in the literature. Three of them are unavailable: the nnMTAC corpus (Pigeon et al. 2007) – 24 hours of real-life, non-native military ATC, the VOCALISE dataset (Graglia et al. 2005) and the corpus of Lopez et al. (2013) – respectively 150h and 22h of real-life French-accented civil ATC.

Available resources are the ATCOSIM (Hofbauer et al. 2008), HIWIRE (Segura et al. 2007) and NIST (Godfrey; 1994) corpora. ATCOSIM and HIWIRE are rather small corpora (resp. 10,7 hours and 8,100 utterances of 1 to 12 words) containing various non-native accents (resp. 3 and 4 distinct accents). Their main limitation is that they were collected in simulated situations. ATCOSIM contains only controller messages uttered during training sessions and there is no radio transmission noise. HIWIRE is geared towards vocal commands and thus includes some datalink commands (pilot-controller texting tool limited to a subset of ATC messages). The text of the commands was generated with a deterministic grammar and then read. Cockpit noise was added afterwards and there is no radio transmission noise. The NIST Air Traffic Control Corpus (Godfrey, J., 1994) is the best fit for our goal: 70 hrs of real-life ATC from 3 different US airports. Still, it shall be extended with non-native data, which we expect to do with the AIRBUS-ATC corpus.

3. Corpus description

The corpus is composed of 2,160 paired audio + transcription files amounting to nearly 59 hrs of transcribed English. Audio files are mono-channel *.wav* files, with 16 kHz sampling rate and 32 bits resolution. Transcription files are in XML-based *.trs* format which is

the format of the transcription software *Transcriber*[1]. In addition to text and time stamps, the *.trs* format allows the encoding of information about speakers, recording context, speech turns, phonetic events and semantic entities.

3.1 Corpus characteristics

The corpus contains three types of Air Traffic Control: approach (APPR) and tower (TWR) controls, both coming from the same airport. It also contains ATIS recordings from 35 French airports. The audio contains both French and English speech, but only the English parts are transcribed.

Results in table 2 highlight that a huge quantity of raw audio is required to obtain English utterances alone, especially when the audio comes from French airports, where French and English are official ATC languages. As for ATIS, the same message is repeated until a change occurs in airport conditions which makes quite small the percentage of relevant data.3.2 Speech turn characteristics

APPR-TWR and ATIS speech turns have distinctive durations. ATIS turns are 6 times longer than APPR-TWR ones.

	Total duration	**English**	**% of English**
APPR-TWR	101:50:08	48:23:13	47.5%
ATIS w/ repetition	285:08:41	1:46:32	0.6%
ATIS w/o repetition	17:18:34	8:47:30	50.8%
TOTAL	404:17:23	**58:57:15**	**14.6%**

Table 2: Corpus content

	average turn duration (sec.)	**average of turn with foreign inclusion (%)**
APPR-TWR	4.4 sec	16.14%
ATIS	29 sec.	0.06%
TOTAL	**5.2 sec**	**15.65%**

Table 3: Speech turns characteristics

Around 16% of English APPR-TWR turns include at least one foreign word. Foreign words are forbidden by ICAO phraseology, these short occurrences correspond to courtesy words, ex. *bonjour* ("hello"), *merci* ("thank you").

3.3 Speakers and language characteristics

18 different native accents are indexed in the APPR-TWR part of the corpus. As expected, French accent is the most represented; other most frequent accents are English, German and Spanish. ATIS is once again distinctive because only spoken by local French natives.

Concerning the distribution of turns between controllers and pilots, speaking times between pilots and controllers are well-balanced as shown in Table 4. This is compliant with ATC phraseology rules: to one sentence emitted by the controller, an assessment from pilots shall follow with a repetition of the initial content that assesses the given instruction.

	% Pilots	**% Controllers**	**% not recog.**
APPR-TWR	55%	44.7%	0.3%

Table 4: Representativeness of roles

The ratio between men and women is disproportionate but it is representative of the real-life working situation.

	% Male	**% Female**	**% unknown**
APPR-TWR	75.3%	24.4%	0.3%
ATIS	68%	31%	1%
TOTAL	75%	24.6%	0.4%

Table 5: Gender representativeness

4. Corpus acquisition and processing

4.1 Audio collection

One key requirement is to collect ATC communications with audio quality as close as possible to real-life conditions. The chosen technical solution was to use a software-defined radio receiver connected to an aeronautical antenna and set to capture local airport APPR-TWR and ATIS broadcasts (~85% of corpus). This setup can collect up to 283 GB of audio data over a 30 days period. The remaining 15% were collected by automatically calling airport dedicated ATIS phone numbers. This means of collection is less tedious but does not provide audio with VHF quality.

4.2 Preprocessing

The raw audio files contain long sequences of silences and are too big to be processed by transcription tools. Preprocessing automatically deletes silences (ie. very low intensity signal over 300 ms duration) and cuts each raw audio file into smaller-sized files. This results in keeping around 25% of the initial input duration. In addition to each new audio file, a corresponding transcription file is created with automatically generated candidate speech turns (based on silence splits).

4.3 Transcription

The transcription was conducted by two types of transcribers: 1) students in the aeronautical field: they are not familiar with the transcription activity itself but they are highly specialized in aeronautics; 2) language specialists from translation/transcription companies, who are not especially familiar with the aeronautical phraseology but master the language. They used the free tool *Transcriber*.

All English utterances are transcribed according to American-English spelling rules. Mispronunciations are not annotated: the intended word is transcribed instead.

Besides the transcription of the pilots/controller's

[1] http://trans.sourceforge.net/

exchanges in English, we asked the transcribers to annotate additional information, listed in Table 6.

Class of info.	Type of info.	Values
Speaker	Function	pilot / controller
	Unique identifier	integer
	Native language	ISO 639-1 code
	Geo. lang. variant	ISO 3166-1 code
	Gender	male/female/unk
Spoken particularities	Gap fillers	huh
	False starts	- (e.g. *del- delta*)
	Not intellig.words	_ (underscore)
	Breaks/pauses	/
	Noise	#
	Foreign lang.	@

Table 6: Linguistic annotations

Transcriptions also contain *call sign* annotations, with distinctive annotations for full forms and short forms. Call signs are used by pilots and controllers to identify to whom they are speaking. Full forms are used when it is the first aircraft identification or when there is a risk of confusion with another aircraft identifier, ex.: *"hello* ***Ryanair*** ***nine*** ***seven papa foxtrot*** *holding november four*". Short forms are used when no confusion is possible: ***C: "Lufthansa four three uniform*** *expect to vacate via mike two"; **P: "four three uniform"***.

Other specificies of ATC are: large use of numbers and figures; use of the ICAO alphabet (*alpha* for *A*, *bravo* for *B*...); use of procedural words (*okay, wilco, roger*) and acronyms (*QNH, CAVOK*...) that are either spelled out or read; heavy use of geographical references such as waypoints or cities. Transcription homogeneity was ensured by constraining transcription guidelines and by using reference lexicons.

4.4 Quality assurance process

Formal aspects of the transcriptions, like their encoding, the syntax of annotations, the presence of unauthorized characters, etc. are checked automatically. Transcriptions shall be 100% compliant. Then, a manual check is performed by a senior linguist with expertise of ATC phraseology on randomly selected samples (50% of transcriptions). Some criteria are allowing a margin of tolerance (e.g. 1 spelling error each 10 min. of audio); other are not (e.g. 0 error on speaker features).

4.5 Post processing

Transcribed files are split into two subsets: 88% (~52 hrs for now) to be used as a train/validation dataset to develop speech recognition engines; and 12% (~7 hrs for now) to be used as an undisclosed gold-standard to benchmark speech engines at the very end of the project.

5. Application to French-accented ATC speech recognition

The goal of this section is to compare what performance may be expected with state-of-the art speech recognition techniques on French-accented ATC with: 1) AIRBUS-ATC data only; 2) NIST corpus only (real-life ATC, mainly US-English accented); 3) AIRBUS-ATC data combined with NIST corpus.

The experiments were conducted on the train/validation part of the AIRBUS-ATC corpus. 80% of the train/validation corpus (~42 hrs) was used to train a state-of-the-art engine (alone or in combination with NIST). The remaining 20% (~10 hrs) were used to evaluate the models (see WER results in Table 7).

The speech recognition engine uses state-of-art techniques. The acoustic model is a Time Delay Deep Neural Network (TDNN) (Peddinti, 2015) containing 6 layers with 4 hidden layers and a total of 6.1M parameters. The language model (LM) is a 4-gram LM built from the transcriptions. We used CMU-Sphinx[2] dictionary and added pronunciation of all the words missing from the dictionary but present in the transcriptions.

Results in table 7 tend to support the claim that accent-specific corpora are key to obtaining good performances on accented speech.

We hypothesize that poor performance of the NIST corpus is largely due to the language model: NIST corpus is older than AIRBUS-ATC corpus (NIST has been recorded in 2004); US ATC terminology differs from the EU terminology that has to follow ICAO (International Civil Aviation Organization) rules; and AIRBUS-ATC also has a lot of lexical items in common with the test corpus (geographical references like waypoints, cities, specific airlines call signs, etc.). Regarding the 0.3% WER decrease of the AIRBUS-ATC+NIST combination, we hypothesize that this is due to the acoustic model performing better trained on both data sets than each of the sets separately.

Training data	Volume	Word Error Rate
NIST	70 hrs	94.7%
AIRBUS-ATC	42 hrs	12.7%
AIRBUS-ATC + NIST	112 hrs	12.4%

Table 7: Performance of the ASR systems

6. Lessons learned

As with all spoken language corpora, constituting an ATC corpus is a complex and tricky task. We faced technical challenges like the one during the recording phase (§4.1) but also methodological issues. The transcription task is

[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

undoubtedly time-consuming and a costly investment.

6.1 Transcription guidelines and training

We initially proposed complex high precision transcription guidelines, with a lot of expected annotation details. Indeed, we asked transcribers to make distinctions between instantaneous and lasting events (silence or noise), to be able to distinguish what belongs to ICAO phraseology or not, to highlight bad pronunciation, human noises like breathing, etc. All in all, there were more than 45 rules and this was too complex both to transcribe and to check. Consequently, we decided to optimize this part of our process by a strong simplification of the transcription guidelines while respecting the objectives and the expected quality of the transcriptions.

Moreover, we underestimated the importance of specifically training transcribers to this task. As we first worked in collaboration with aeronautical specialists, we thought that simply giving the transcription guidelines and a short briefing would be sufficient to get quality transcriptions. But, the first results were not only transcribed with a lot of spelling errors but also delivered later than expected: we misevaluated the time needed for transcription. In a second phase, we then contacted language specialists. We strongly insisted on the training, with face-to-face workshops on what is expected in terms of transcription and annotations and with continuous support. This greatly improved the quality of transcriptions.

6.2 Issues with spoken language specificities

For most transcribers, a major difficulty lies in discriminating repeated, broken words and lasting syllables. In fact, a repeated word with a specific lasting accentuation on the last syllable was often transcribed as a broken word. In the examples below we underline the lasting syllable for a better understandability; the hyphen ("–") is the convention used to indicate a broken word:

1. *continue- continue*
2. *wind is one two zero degrees two- two zero knots*

In both examples, the first word of the repeated sequence is a complete word, and not part of a word although the hyphen should correspond to a break before the end of a word. Expected transcriptions are:

1. *continue continue*
2. *wind is one two zero degrees two two zero knots*

The identification of a speaker's characteristics like native language (and geographical variant) are also really difficult. It is indeed tricky for a French native to differentiate between a speaker coming from China or from Japan. Moreover, a speaker generally talks many times, in more than one non-adjacent speech turns: the transcriber needs a high level of attention and audio memory to be able to recognize and set a unique identifier to the same speaker.

Finally, code-switching between French and English words can be misleading. In the following examples, the controller starts his sentence in English and ends it with French .

> *Quality zero five four sierra huh @* → where @ equals to *"alerte huh relief vérifiez votre altitude*" (terrain obstacle check your altitude)

In the example below, a pilot utters a sentence in English except for the numbers which are enunciated in French:

> *cleared for takeoff huh three two right Quality @ hotel* → where @ equals to "*cent vingt quatre*" (hundred twenty four)

6.3 Transcription work time

	TWR-APPR	ATIS
Under trained **ATC Specialists** French speakers	20 to 40 min	20 min
Under trained Not ATC Specialists **English speakers**	60 min	N/A.
Trained Not ATC Specialists **French/English speakers**	6 to 20 min	N/A.

Table 8: Transcription duration for 1 min of audio

The figures from Table 8 show that being an ATC specialist or an English native is not particularly an advantage for the transcription of ATC communications in English. The best results are given by trained language specialists with a mixed team of native English and French people.

7. Conclusion and perspectives

This paper synthesizes the work that led to the collection of unique real-life, French-accent, speech corpus of Air Traffic Control communications aimed at building an ATC-specific speech recognition engine. Preliminary results obtained with the corpus using a state-of-the-art engine are encouraging. We also shared techniques used to collect the data, ensure quality transcription and lessons learnt from this experience. Our major perspective lies in the improvement of the linguistic resources: increasing size and accent variety of the audio data as well as developing ATC-specific pronunciation lexicons. A detailed evaluation will be performed to investigate the influence of other accents, control type (APPR-TWR vs. ATIS), speaker type (controller vs. pilot), etc. on error rate.

8. Acknowledgements

This work was funded by AIRBUS, HONEYWELL and the Clean Sky 2 research program.

10. Bibliographical References

Cauldwell, R. (2007). SpeechInAction: Fluency for Air Traffic Control. PTLC. London. August 2017. Online: http://www.phon.ucl.ac.uk/ptlc/proceedings/ptlcpaper_19e.pdf

Graglia, L., Favennec, B., and Arnoux, A. (2005). Vocalise: Assessing the impact of data link technology on the R/T channel. The 24th Digital Avionics Systems Conference, Vol. 1.

Hann, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., and A. Y. Ng. (2014). Deep Speech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567.

ICAO (2007). Manual of Radiotelephony. Doc 9432-AN/925, 4th edition.

Junqua, J-C., Fincke, S., and Field, K. (1999). The Lombard effect: A reflex to better communicate with others in noise. IEEE International Conference on Acoustics, Speech, and Signal Processing. Vol. 4.

Peddinti, V., Povey, D., Khudanpur, S. (2015). A Time Delay Neural Network Architecture for Efficient Modeling of Long Temporal Contexts. In proceedings of INTERSPEECH 2015. Dresden, Germany, pp. 3214-3218.

Lopez, S., Condamines, A., Josselin-Leray, A., O'Donoghue, M., and Salmon, R. (2013). Linguistic analysis of English phraseology and plain language in air-ground communication. Journal of Air Transport Studies 4.1: 44-60.

Saon, G., Kurata, G., Sercu, T., Audhkhasi, K., Thomas, S., Dimitriadis, D., Cui, X., Ramabhadran, B., and Picheny, M. (2017). English conversational telephone speech recognition by humans and machines. arXiv preprint arXiv:1703.02136, pp. 132-136.

Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., Yu, D. and Zweig, G. (2016). Achieving human parity in conversational speech recognition. arXiv preprint arXiv:1610.05256.

11. Language Resource References

Cieri, C., Miller, D., and Walker, K. (2004). The Fisher Corpus: a Resource for the Next Generations of Speech-to-Text. LREC. Vol. 4, pp. 69-71.

Godfrey, J. (1994). *Air Traffic Control Complete* LDC94S14A. Web Download. Philadelphia: Linguistic Data Consortium.

Godfrey, J., Holliman, E., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. IEEE International Conference on Acoustics, Speech, and Signal Processing. Vol. 1, pp. 517-520.

Hofbauer, K., Petrik, S., and Hering, H. (2008). The ATCOSIM Corpus of Non-Prompted Clean Air Traffic Control Speech. LREC.

Pigeon, S., Shen, W., and van Leeuwen, D. (2007). Design and characterization of the non-native military air traffic communications database (nnMATC). INTERSPEECH.

Segura, J. C., Ehrette, T., Potamianos, A., Fohr, D., Illina, I., Breton, P.-A., Clot, V., Gemello, R., Matassoni M., and Maragos, P. (2007). The HIWIRE database, a noisy and non-native English speech corpus for cockpit communication. Online. http://www.hiwire.org.

Creating Lithuanian and Latvian Speech Corpora from Inaccurately Annotated Web Data

Askars Salimbajevs
Tilde, Vienibas gatve 75A, Riga, Latvia
University of Latvia, Raina bulv 19, Riga, Latvia
askars.salimbajevs@tilde.lv

Abstract

This paper describes the method that was used to produce additional acoustic model training data for the less-resourced languages of Lithuanian and Latvian. The method uses existing baseline speech recognition systems for Latvian and Lithuanian to align audio data from the Web with imprecise non-normalised transcripts.

From 690 hours of Web data (300h for Latvian, 390h for Lithuanian), we have created additional 378 hours of training data (186h for Latvian and 192 for Lithuanian). Combining this additional data with baseline training data allowed to significantly improve word error rate for Lithuanian from 40% to 23%. Word error rate for the Latvian system was improved from 19% to 17%.

Keywords: speech recognition, web data, alignment, Lithuanian, Latvian, speech corpora, low-resourced languages

1. Introduction

Training of an acoustic model for an automatic speech recognition (ASR) system requires large amounts of transcribed audio, especially for the state-of-the art deep neural network models. For Low Resource Languages (LRLs), where there may only be a few hours of transcribed audio, this is a very serious problem. For Latvian and Lithuanian languages the situation is better, for example, there are a 100h Lithuanian speech corpus created for the LIEPA project[1], an 84h corpus from BMMG(Alumäe and Tilk, 2016) and a 100h Latvian speech corpus(Pinnis et al., 2014). However, even this is not much, as neural network acoustic models can make use of many thousands of hours of data and achieve significant improvement in recognition accuracy.

In recent years, improvements in data storage and networking technology have made it feasible to provide Internet users with access to large amounts of multimedia content. This content can be automatically collected and processed for the purpose of training statistical models. However, in many cases, this content is not structured or organised in an accurate and machine-readable form.

For example, both the Lithuanian Parliament (Seimas) and Latvian Parliament (Saeima) websites contain a large archive of video recordings of parliamentary sessions and edited transcripts. One may want to collect this data and use it for the development of a general purpose speech recognition system. This data can also be used for adapting existing ASR for transcription of Seimas sessions. Edited transcripts can be used as the training data for the language model, and video recordings can be used for adapting the acoustic model.

Unfortunately, edited transcripts do not have any timing information. Also, these transcripts are non-normalised. This means that numbers, dates, percent signs, etc. are written with digits and symbols, not as words. Converting these tokens back to words is complicated and error-prone for inflected languages like Latvian and Lithuanian.

While speaking humans can make grammar mistakes, repeat words, make corrections and restart whole sentences. Edited transcript is a written document, thus it contains only final, grammatically correct and re-formulated sentences, which makes it not 100% accurate.

In this paper, we describe the method we used to obtain additional training data by aligning audio and edited transcripts from Lithuanian and Latvian parliament websites. This helped us to significantly improve the Lithuanian general purpose ASR and also resulted in noticeable improvement for the Latvian ASR.

The alignment between long audios and their corresponding transcripts has been previously studied in the context of various applications. (Panayotov et al., 2015) use existing ASR and audio-alignment techniques for creation of a large training corpus from public domain audio-books, and (Anguera et al., 2014) use ASR for different languages and a clever phoneme-based alignment approach for training speech recognition with very limited language resources. (Prahallad and Black, 2011) describe the creation of aligned corpora for building text-to-speech systems, and (Hazen, 2006) focuses on the automatic alignment and correction of inaccurate text transcripts through an iterative process.

Also, Lithuanian is one of the development languages within the IARPA BABEL research program (Harper, 2013), and therefore, Lithuanian is one of the test languages in many papers that have studied low-resource training methods for speech recognition, e.g. (Mendels et al., 2015; Davel et al., 2015; Gales et al., 2015) and many others. While BABEL focuses more on conversational telephone speech (Lileikyte et al., 2018), broadcast speech recognition for Lithuanian was addressed within the Quaero research program. In (Lileikyte et al., 2016), a semi-supervised method was used to train an acoustic model with only three hours of transcribed data and 360

[1]LIEPA (Services Controlled by Lithuanian Voice), Vilnius University, LEU, Siauliai University, Institute of Lithuanian language. https://www.xn–ratija-ckb.lt/liepa

hours of untranscribed data. The untranscribed data was iteratively automatically transcribed and added to the training data. The system achieved a remarkable word error rate (WER) of 18.3% on Quaero broadcast speech evaluation data. Unfortunately, we can not compare this result with our method, as Quaero systems and evaluation data are not publicly available.

The method that we are using in this paper is similar to (Panayotov et al., 2015) and (Hazen, 2006). The main differences are the use of the SpkDiarization toolkit(Rouvier et al., 2013) for segmenting large audio recordings into smaller manageable segments and for providing speaker diarisation, an optional intermediate step where we use the retrained model for better alignment, a different aligned segment extraction process and the fact that it is being done for the less-researched and less-resourced Latvian and Lithuanian languages.

2. Alignment of Parliament session transcripts

2.1. Data collection and processing

A web-crawler script is used for collecting video recordings of parliamentary sessions and the corresponding human-edited reference transcripts from both Lithuanian Seimas and Latvian Saeima websites.

Then, audio is extracted from each video file and processed by the LIUM SpkDiarization toolkit(Rouvier et al., 2013), which segments audio into smaller parts and groups them into clusters (that should correspond to different speakers). Later we will add these segments directly to the training data, so each segment should be reasonably short and contain speech only from one speaker. Also, clustering by speaker is important for correct Speaker Adaptive Training(Anastasakos et al., 1997; Miao et al., 2014).

Reference transcripts are normalised, all punctuation and non-alphabetic characters are removed, all words are lowercased. We also rewrite numbers and dates into words, a module from a text-to-speech engine is used for this purpose for Latvian, but for Lithuanian we just use the nominal forms as we do not have the necessary tools.

Audio segments are then processed by ASR. Clustering information is used during recognition for speaker adaptation. After all these processing steps, from each video file the following files are obtained:

- A corresponding inaccurate reference transcript in normalised form.
- A set of short audio files that roughly correspond to separate utterances. This set is sorted in chronological order and clustered into different speakers.
- A raw ASR transcript for each short audio file.
- Word alignment information (time, when each word is pronounced and length of pronunciation) from ASR for each audio file.

2.2. First Alignment Step

Next, we take each reference transcript file and perform a global alignment between the inaccurate reference text and per-utterance ASR transcripts. Similarly to (Panayotov et al., 2015), we use the Smith-Waterman alignment algorithm (Smith and Waterman, 1981). An example of such alignment is shown in Figure 1. The grey boxes represent boundaries between different utterances obtained by the LIUM SpkDiarization toolkit.

After the alignment, in each utterance, we select continuous sequences of matched words that are longer than some threshold (e.g. 3 in Figure 1). Using word alignment from ASR, these word sequences are extracted from utterance audio files and are added to the new training data set together with their transcripts from ASR. Speaker diarisation is also preserved, so that utterances from the same speaker are grouped together.

A length threshold is needed to filter out possible alignment errors, for example, short word sequences like ”un

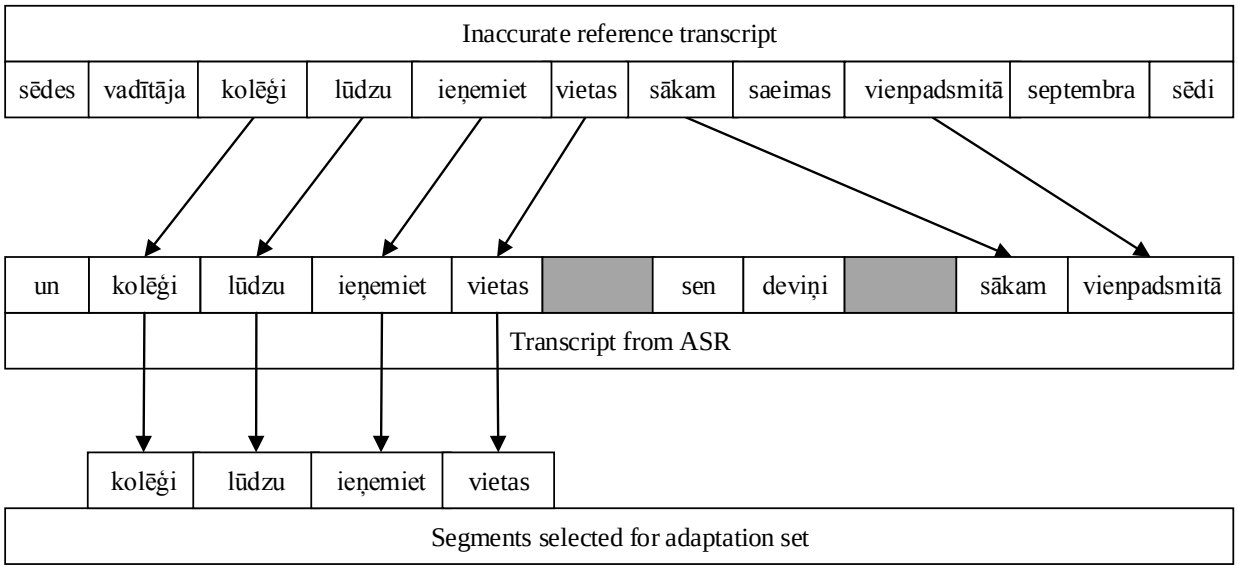


Figure 1: Alignment between the reference transcript and ASR output.

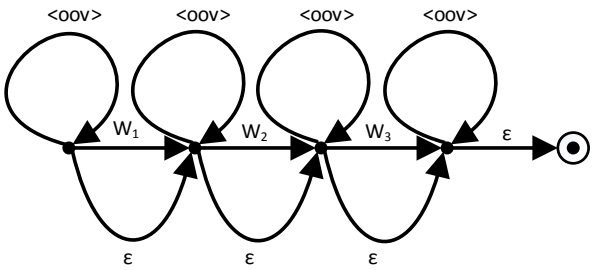


Figure 2: Example FSA for the pseudo-forced alignment of the word sequence w_1, w_2, w_3 with insertions, deletions and substitutions allowed.

tas ir" ("and that is"), "un ir" ("and is"), etc. are rather frequent and can be either misrecognised by ASR or misaligned. Also, we assume that ASR word alignment for longer sequences is more accurate, so extracting longer sequences is less likely to cut off word beginnings and endings. In the first alignment step we use a threshold of 5 or more consecutive words in this work.

2.3. Second Alignment Step

Word sequences extracted in the first alignment step can already be used for training acoustic models. However, the improvement from adding these sequences to the training data will be limited because existing ASR already recognised them correctly; they already match the acoustic model quite well. The parts that were not recognised accurately (and not aligned) can be much more useful, as they are examples of when the existing acoustic model is not good enough.

Successfully extracted segments could be used as "anchor points" so that the audio between anchor points will be mapped to the text between anchor points. This mapping then can be used to help ASR to recognise this part correctly and produce a better alignment.

However, before that, there is an optional intermediate second alignment step that is needed to improve the alignment and get more anchor points. We append the data extracted in the first step to the training data, retrain the acoustic model (we will typically retrain only the DNN part of the acoustic model). The retrained model used for repeated decoding and alignment.

The extraction threshold is changed from 5 consecutive words to 22 consecutive phones. The number 22 represents the average length of 3 average words in both Latvian and Lithuanian and is calculated on vocabularies of both systems. This less strict threshold creates more anchor points.

2.4. Pseudo-Force Alignment

After obtaining mappings between misrecognised audio segments and reference text, the natural choice would be to perform a classic force alignment. However, in our case, it's not suitable, as the reference text is not 100% accurate, so instead we perform a pseudo-force alignment step like in (Hazen, 2006).

We use successfully extracted segments as "anchor points" so that the audio between anchor points will be mapped to the text between anchor points. In our case of edited transcripts, reference text can contain long regions of insertions and audio can contain long non-speech regions, so we have limits for text and audio length. Mappings that are too long are filtered out.

We assume that errors in the transcript are possible, and we allow insertions of new words, deletions and substitutions for existing words. This process is realised through the composition of a pseudo-forced alignment finite state acceptor (FSA) with a lexical transducer from the baseline ASR. An example alignment FSA that allows insertions, substitutions and deletions is shown in Figure 2.

Insertions are modelled through the use of an out-of-vocabulary (OOV) word filler model. This is are single phone model with 5 HMM states. During baseline acoustic model training this model was used to consume OOV and foreign words, fragmented words and spoken noise.

After decoding the pseudo-force aligned audio segments, we select segments with a length of 22 or more phonemes and append them to the training data.

3. Results

3.1. Baseline Speech Recognition Systems

Baseline Latvian and Lithuanian ASR systems are implemented in the open-source Kaldi toolkit (Povey et al., 2011). The recipes for both systems are very similar:

- Grapheme-based pronunciation model.
- HMM-DNN p-norm (Zhang et al., 2014) acoustic model with iVectors for speaker adaptation (Miao et al., 2014).
- Vocabulary of about 800,000 word forms.
- Language models are trained on texts collected from web news portals. After filtering, each training corpus contains about 40M sentences.
- 2-gram language model for decoding and 3-gram for rescoring.

The Lithuanian ASR is trained on speech recordings collected by the LIEPA project and 40h of Seimas recordings from year 2016. About a half of the LIEPA corpus is silence (because it contains isolated voice commands), so the resulting size of training data set is 92h. While the Latvian ASR is trained on a specifically designed 100h Latvian Speech Recognition Corpus (LSRC)(Pinnis et al., 2014).

3.2. Improving Latvian Speech Recognition

For this research, we downloaded and processed about 300 hours of video recordings of Saeima sessions in the period of 2011-2014. After the first alignment step we obtained 120 hours of data.

Next, we evaluated the impact of adding these 120h to the acoustic model training data. For this evaluation, we used a 1 hour test set that is recorded using smartphones and contains recordings of 10 non-professional speakers reading news on different topics from different news web pages and some excerpts from fiction literature. The corpus was manually annotated.

	Size, h	WER, %
Google Cloud Speech	n/d	33-44
Baseline (LSRC)	100	19.6
+ Saeima (1st step)	220	17.0
+ Saeima (2nd step)	249	17.0
+ Pseudo-force aligned	286	**16.9**

Table 1: Evaluation of Latvian speech recognition on the general domain test set.

An improvement in recognition quality was observed (see Table 1) after adding these 120 hours of Saeima recordings to the acoustic model training data. Word error rate was reduced from 19.6% to 17.0% (13% relative).

We have also processed the same test set with the Google Cloud Speech service for comparison. Unfortunately, Cloud Speech service filters out parts with low confidence, so direct calculation of WER, shows very poor result (44%) with a lot of deletions. If deleted segments are not counted WER is 33%. We believe that real WER for Google's system should be somewhere in the middle of this interval.

In the second alignment step, the ASR improved by the new training data from the first step was used to repeat the decoding and alignment procedure. This allowed to improve the alignment, first to 136 hours and then, by changing the threshold to 22 characters, to 149 hours. However, no improvement in WER was detected in either case.

Because a large part of the data has been successfully decoded in the first two steps, only 57 hours of data were selected for pseudo-force alignment. 37 hours were successfully aligned and added to the training set, the WER of ASR trained on all of the data combined is 16.9%.

3.3. Improving Lithuanian Speech Recognition

For the evaluation of Lithuanian speech recognition, we used a 1-hour "general domain" test set that consists of manually transcribed randomly picked audio segments from various radio and TV shows (mainly "Atviras pokalbis" and "Labas rytas, Lietuva"), and Seimas sessions from 2017. The corpus does not have speaker information, so speaker adaptation is done only a the utterance level.

We have also processed the same test set with the Google Cloud Speech service, however, as in previous case, the service omits large segments for which it has low confidence, so the WER is high (40%) because of deletions (if these segments are not counted, WER is about 27%). We have also evaluated ASR from (Alumäe and Tilk, 2016) on this set (compounding was not performed).

For first experiment, we have downloaded about 270 hours of video recordings of Seimas sessions from years 2015 and processed with our baseline system. After alignment and extraction, we obtained an additional 52 hours of data. The retrained ASR system achieved a WER of 25.5%. Repeating retraining and changing the threshold to 22 characters allowed obtaining 13 more hours (65 hours of aligned data in total), but no improvement in WER was detected.

Pseudo-force alignment produced 71 hours of additional aligned data (from 138 hours). Appending this data to the training data resulted in an improvement of WER from 25.5% to 24.4% (4% relative).

As alignment resulted in only about 130h of data (comparing to 180h for Latvian), we decided to download more Seimas audio (120h of recordings in the period from November 2016 till April 2017) and process it in the exactly same way. This allowed to obtain 56h of additional aligned data. Appending this data to the training data resulted in an improvement of WER from 24.4% to 23.3% (5% relative). The results are summarised in Table 2.

4. Discussion

In the first and second alignment steps, we extract segments of data that have a 100% match with the reference transcript. This means that our existing ASR already recognises such segments correctly. However, after adding these segments to the training, we see an improvement in WER. We believe that this is caused by (1) better senone coverage in the larger training set and (2) a large "language model bias" that allowed some segments to be recognised correctly even when the acoustic score was too low.

When doing pseudo-force alignment, this "language model bias" is even larger, so adding pseudo-force aligned data should improve word error rate even more. However, experiments showed a small improvement for both languages. This means that not all of the pseudo-force aligned data are complementary to the data extracted in the first two steps.

The data aligned in the first step can be used to adapt the acoustic model and repeat the decoding. Our experiments show that this improves the second alignment (i.e. more data are successfully aligned), however adding new data from the second alignment to the training, does not result in WER improvements for both languages.

Both ASR systems significantly outperform Google ASR for given languages on our test set, but this is probably because Google systems are strongly adapted to specific domain and specific usage scenario. Also, it's problematic to correctly evaluate Google systems as a lot of transcript post-processing is performed (filtering of low-confidence segments, inverse text normalization etc).

Also, by adding automatically aligned Seimas data to the training we managed to outperform the Lithuanian speech recognition by (Alumäe and Tilk, 2016). It is unknown how our system will perform on the broadcast speech recognition, however it seems that result will be comparable to Quaero(Lileikyte et al., 2016) and (Alumäe and Tilk, 2016).

	Size, h	WER, %
Google Cloud Speech	n/d	27-40
(Alumäe and Tilk, 2016)	84	25.2
Baseline	92	27.4
+ Seimas 2015 (1nd step)	144	25.5
+ Seimas 2015 (2nd step)	157	25.5
+ Pseudo-force aligned	228	24.4
+ Seima 2017 + PFA	284	**23.3**

Table 2: Evaluation of Lithuanian speech recognition on the general domain test set.

5. Conclusions

In this work, we have automatically aligned audio recordings with imprecise transcriptions and created additional training data for the less-resourced languages of Latvian and Lithuanian (about 190h for each). The data was automatically collected from the websites of the Latvian and Lithuanian parliaments.

Training ASR with new training data allowed improving WER for both Latvian and Lithuanian languages. This suggests that the collected training data are complementary to the baseline training corpora. The improvement is similar for both languages and is about 13% relative in this work.

We believe that experiment results show that the method is language independent and can be used to collect more training data for other less-resourced languages.

In future work, we plan to use this method to process more recordings for both languages and obtain a larger speech corpora.

6. Acknowledgements

The research has been supported by the European Regional Development Fund within the project "Neural Network Modelling for Inflected Natural Languages" No. 1.1.1.1/16/A/215.

7. Bibliographical References

Alumäe, T. and Tilk, O. (2016). Automatic speech recognition system for lithuanian broadcast audio. In *Human Language Technologies - the Baltic Perspective : Proceedings of the Seventh International Conference, Baltic HLT 2016*, volume 289, pages 39–45. IOS Press.

Anastasakos, T., McDonough, J., and Makhoul, J. (1997). Speaker adaptive training: a maximum likelihood approach to speaker normalization. *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2:0–3.

Anguera, X., Luque, J., and Gracia, C. (2014). Audio-to-text alignment for speech recognition with very limited resources. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (INTERSPEECH'2014)*, pages 1405–1409.

Davel, M., Barnard, E., Van Heerden, C., Hartmann, W., Karakos, D., Schwartz, R., and Tsakalidis, S. (2015). Exploring minimal pronunciation modeling for low resource languages. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2015-January, pages 538–542.

Gales, M. J., Knill, K. M., and Ragni, A. (2015). Unicode-based graphemic systems for limited resource languages. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2015-August, pages 5186–5190.

Harper, M. (2013). The BABEL program and low resource speech technology. In *ASRU*.

Hazen, T. J. (2006). Automatic alignment and error correction of human generated transcripts for long speech recordings. In *Proceedings of the 9th International Conference on Spoken Language Processing (INTERSPEECH'2006)*, volume 2006, pages 1606–1609.

Lileikyte, R., Gorin, A., Lamel, L., Gauvain, J. L., and Fraga-Silva, T. (2016). Lithuanian Broadcast Speech Transcription Using Semi-supervised Acoustic Model Training. In *Procedia Computer Science*, volume 81, pages 107–113.

Lileikyte, R., Lamel, L., Gauvain, J. L., and Gorin, A. (2018). Conversational telephone speech recognition for Lithuanian. *Computer Speech and Language*, 49:71–82.

Mendels, G., Cooper, E., Soto, V., Hirschberg, J., Gales, M., Knill, K., Ragni, A., and Wang, H. (2015). Improving speech recognition and keyword search for low resource languages using web data. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2015-January, pages 829–833.

Miao, Y., Zhang, H., and Metze, F. (2014). Towards speaker adaptive training of deep neural network acoustic models.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE.

Pinnis, M., Auzina, I., and Goba, K. (2014). Designing the Latvian Speech Recognition Corpus. In *Proceedings of the 9th edition of the Language Resources and Evaluation Conference (LREC'14)*, pages 1547–1553.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The Kaldi Speech Recognition Toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society.

Prahallad, K. and Black, A. W. (2011). Segmentation of monologues in audio books for building synthetic voices. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1444–1449.

Rouvier, M., Dupuy, G., Gay, P., Khoury, E., Merlin, T., and Meignier, S. (2013). An open-source state-of-the-art toolbox for broadcast news diarization. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH'2013)*.

Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197.

Zhang, X., Trmal, J., Povey, D., and Khudanpur, S. (2014). Improving deep neural network acoustic models using generalized maxout networks. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 215–219.

Discovering Canonical Indian English Accents: A Crowdsourcing-based Approach

Sunayana Sitaram*, Varun Manjunath†, Varun Bharadwaj†, Monojit Choudhury*, Kalika Bali*, Michael Tjalve+

*Microsoft Research, Bangalore, India
susitara, monojitc, kalikab@microsoft.com

†PES University, Bangalore, India
varunmanjunath2012, varunbharadwaj1995@gmail.com

+Microsoft Corporation, Redmond, USA
University of Washington, Seattle, USA
mitjalve@microsoft.com

Abstract

Automatic Speech Recognition (ASR) systems typically degrade in performance when recognizing an accent different from the accents in the training data. One way to overcome this problem without training new models for every accent is adaptation. India has over a hundred major languages, which leads to many variants in Indian English accents. Making an ASR system work well for Indian English would involve collecting data for all representative accents in Indian English and then adapting Acoustic Models for each of those accents. However, given the number of languages that exist in India and the lack of a prior work in literature about how many Indian English accents exist, it is difficult to come up with a set of canonical accents that could sufficiently capture the variations observed in Indian English. In addition, there is a lack of labeled corpora of accents in Indian English. We approach the problem of determining a set of canonical Indian English accents by taking a crowdsourcing based approach. We conduct a mobile app based user study in which we play audio samples collected from all over India and ask users to identify the geographical origin of the speaker. We measure the consensus among users to come up with a set of candidate accents in Indian English and identify which accents are best recognized and which ones are confusable. We extend our preliminary user study to a web app-based study that can potentially generate more labeled data for Indian English accents. We describe results and challenges encountered in a pilot study conducted using the web-app and future work to scale up the study.

Keywords: non-native accents, crowdsourcing, Indian languages, speech processing

1. Introduction

Automatic Speech Recognition (ASR) systems have reached performance on par with humans on some tasks and languages. However, the performance of ASR systems is significantly worse in the presence of accents that are different from the accents in the data used for training the ASR system. Since commercial ASR systems require hundreds or thousands of hours to train, it is not always feasible to train a separate system for every accent the ASR system will have to recognize. One solution to this is to adapt Acoustic Models that are trained on one accent to a target accent by using a small amount of speech data in the target accent (Huang et al., 2014).

India is an extremely multilingual country. According to the 2001 Census of India (Banthia, 2001) there are 122 major and 1599 other languages in India; 23 languages have been granted the status of official languages. The existing literature on Indian English (IE) accents either focuses on its difference with British or American English, or on very specific phonetic features that mark IE (Sahgal and Agnihotri, 1988) (Kachru, 2005) but fails to define what a canonical IE accent is, or what a set of canonical IE accents could be. Recently, it has been shown that using native language (L1) data could help adapt Acoustic Models to an accent influenced by that L1 (Aditya Siddhant, 2017). However, in the case of IE, we face the issue of not knowing which native language(s) should be chosen for adaptation. Labeled data exists for some major accents in English, however, there does not exist any labeled data covering all the possible accents in IE.

In this work, we follow a crowdsourcing based approach to finding out what the set of canonical accents in IE could be. We build mobile and web-based interfaces using which users listen to audio samples of IE and annotate on a map of India where they think the speaker is originally from. Users have the option of choosing one of five geographical regions in India, or one of the 29 states. Then, we analyze responses from all users to find geographical regions where labels have high agreement and regions that are confusable. In this case, we use geographical region as a proxy for L1, which in turn is assumed to influence the IE accent of a speaker. This assumption is reasonable because many states in India have their own language, and there is a correspondence between state and a major language except in the case of some states in North India.

Because of the lack of annotated accent data, or information about the L1 of the speakers in our audio samples, we have no ground truth L1 labels. So, we use geographical location of an audio file initially as a proxy of what the accent of the speaker could be. This clearly does not hold true for people who have migrated from one part of the country to another. In addition, exposure to different languages, travel, the level of education and other socio-linguistic factors play an important role in determining one's accent. We describe how we propose to handle some of these challenges in the pilot and the large-scale web-based study.

In addition to finding accents that are well identified, we are also interested in knowing which geographical regions may produce accents that are not easily distinguishable as a particular accent, as such accents can be thought of as neutral or mild IE accents. A neutral IE accent may also be useful for a personal digital assistant that has to have a common accent for users from all over the country.

The rest of this paper is organized as follows. First, we describe datasets available in other English accents and prior work on Indian English accents. Next, we describe our mobile app-based pilot study and data analysis, followed by the design of our larger scale web-based study. We describe preliminary findings from a pilot study conducted using the web app, and some of the challenges faced. We conclude with ongoing and future work.

2. Relation to Prior Work

Labeled data for accent modeling and adaptation exists for some English accents. The CLSU Foreign Accent Corpus (Lander, 2007) consists of 1-2 hours of spontaneous speech by native speakers of 22 languages including Hindi and Tamil. In addition, the CSLU corpus also has judgments on the heaviness or mildness of each accent on a four point scale. The ABI (D'Arcy et al., 2004) corpus consists of 95 hours of recordings from 300 speakers, representing 15 accents of the British Isles. There does not exist a comprehensive labeled corpus that covers all the major L1s of speakers of Indian English.

Most studies on IE accents have focused on vowel analysis. (Phull and Kumar, 2016) describe a study on vowel analysis for four IE accents - North, South, East and West Indian. They found that there was a significant difference in the first four formants in these accents. (Maxwell and Fletcher, 2009) study the acoustic and durational properties of vowels of speakers whose L1 is Hindi or Punjabi, and find contrasts between the two. (Kalashnik and Fletcher, 2007) suggest that North Indian English shows distinct vowel patterns making it a separate sub-variety of IE.

(Sirsa and Redford, 2013) carried out a study to compare the sound structures of IE produced by native Hindi and Telugu speakers. They found that the L1 influenced the production of some segments in IE, but L1 temporal patterns were not found in IE. They also asked experienced and naive listeners to distinguish the speech based on L1. Experienced listeners could do so better than naive listeners. (Maxwell, 2014) studied the intonational phonology of Kannada and Bengali Indian English and found intonational differences within IE and between IE and other Englishes.

Although there have been studies on specific varieties of Indian English accents, to the best of our knowledge, there does not exist prior work or data to identify a set of canonical Indian English accents.

3. User Study

3.1. Data

We used an in-house data set of spoken queries to a speech recognition system as data for the pilot study. We divided India into 5 geographical regions and selected 25 representative cities and towns in total spread out over these regions. We tried selecting cities without a very high immigrant population to circumvent the problem of not knowing the true L1 of a speaker. We tried to avoid very large cities, because speakers from large cities may have milder, more urban accents that are harder to identify. However, in some cases our choice of city was dependent on the availability of data. In preliminary user studies, we found that users could not identify accents when the utterances were shorter than a few seconds. So, we manually listened to and collected data that was sufficiently long and that did not reveal any location based information, thus avoiding sentences like 'What is the weather like in Mysore?'.

Initially, we created an Android app which would play an audio file and show the user either a map of the country, or a drop-down menu of states and regions of India. In preliminary studies, we found that users preferred the map-based interface and decided to use that as the interface for the study. In the map-based interface, users were shown a map of the country with state boundaries. If they clicked on the map, one of 5 geographical regions would be highlighted. They had the option of zooming into a region once to see all the states in the region, if they wished to make a finer-grained decision at the state-level.

We selected 10 sentences from each of the 25 cities and towns, leading to 250 audio examples in all. Each user listened to 15 examples, with 3 examples from each of the 5 geographical regions. Even though this did not guarantee that they were listening to accents from all the regions, overall this gave users a reasonable distribution to listen to. We wanted each audio file to be labeled by at least three users, so we conducted the user study with 60 users, with each user listening to 15 utterances and having the option of skipping an audio file if they did not want to provide a judgment for it.

We also collected optional demographic information including the users' L1, other languages they knew, their educational qualifications and a list of places they had lived in for more than a year. Participants in this pilot consisted of students and researchers from our research lab or visitors to the lab. Most participants were between 20-35 years of age and were undergraduates, graduate students or post graduates. The participants spoke the following L1s: Hindi, Tamil, Telugu, Konkani, Malayalam, Punjabi, Bengali, Kannada, Gujarati, and Marathi. Among these L1s, Hindi, Tamil, and Telugu were the most common.

At the end of the study, participants were given the option of recording a short paragraph taken from the Accent Project at GMU (Weinberger, 2014). This paragraph was designed to capture most sounds in English and consisted of familiar words, but some difficult sound sequences. All participants volunteered to record this paragraph. The recording was conducted using the mobile app, so there was some background noise present in the recordings, leading to realistic training data for a speech recognizer. We obtained around 60 audio recordings of the paragraph by participants with corresponding demographic data.

3.2. Analysis

Next, we present an analysis of the data we collected in the pilot study. We compared participant responses to the ge-

ographical location of the audio sample from our in-house database of queries. Although we used geographical location as a proxy for accent, which has some limitations as described earlier, we saw some general trends in the data.

Participants had the option of choosing two levels of granularity while making their selection: region or state. 32% of the annotations were state level, while the rest were region-level. Figure 1 and Figure 2 show the confusion matrix of the region and inferred region-based judgments for all the audio samples. Figure 1 shows the actual regions picked by the users, while Figure 2 also shows the region that we inferred based on the state that the users chose. The color of the boxes and the numbers inside them indicate the absolute number of judgments, with the Y axis containing the true label (actual geographical origin) of the audio sample and the X axis indicating the judgments. From both the figures, we see that South India is the region that has maximum agreement between the geographical origin and judgment, while North India is second. We also see that there is a large difference in the counts between the correctly predicted region of South India in the two figures, which indicates that users were confident enough to also pick a South Indian state while making the judgment - roughly one-third of the time, users picked a South Indian state, and picked the entire region of South India two-thirds of the time. We see similar trends for North India as well. Both figures also show that Central India was confused with North India.

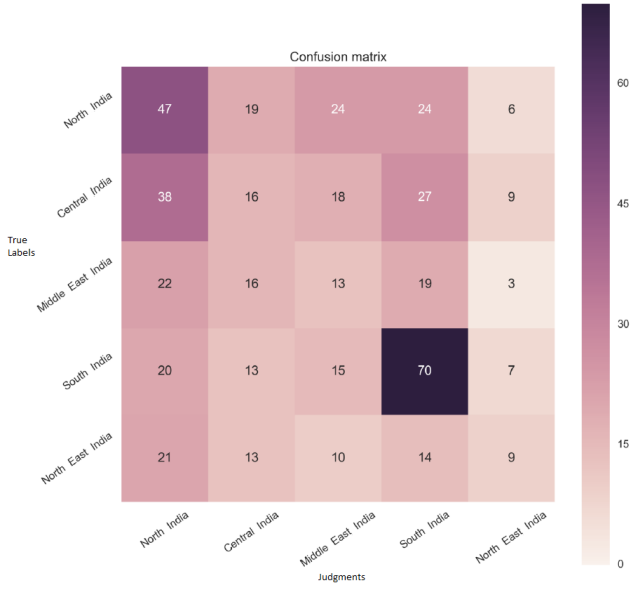


Figure 1: Region confusion matrix

It is important to note that our definition of regions was based on a particular grouping of Indian states into zones. A different grouping of regions based on the similarity of Indian languages could lead to more interpretable results. Similarly, we could choose to replace state boundaries with regions where a major Indian language is spoken.

We created a similar confusion matrix for the state-level judgments, shown in Figure 3. Some observations are as follows.

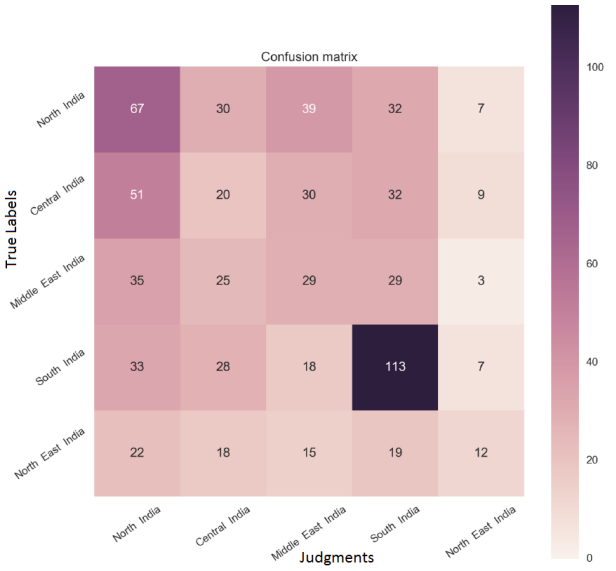


Figure 2: Inferred region confusion matrix

1. Punjab was most likely to be identified as Uttar Pradesh, but Uttar Pradesh was most likely to be identified as itself.

3. Tamil Nadu was well identified in general but sometimes confused with Maharashtra and UP

4. Karnataka was identified most often as itself, but was sometimes confused with Tamil Nadu, Kerala, Andhra Pradesh

5. Andhra Pradesh was most often identified as Maharashtra

6. None of the other states had strong diagonal values in the confusion matrix, which meant that they were not easily distinguishable

7. Some states, such as Telangana, Uttarakhand, Jharkhand, Himachal Pradesh and Sikkim were not present in the data and were not picked as candidate states by users. The states of Telangana, Uttarakhand and Jharkhand are newly-formed states in India.

Each file received at least three annotations and the geographical origin was used as the reference. We calculated normalized scores for each state that had been annotated by the users as follows. For each label, if it was an exact match with the geographical origin state of the file, we gave it a score of 1. If it was a neighboring state, we gave it a score of 0.5. If the state was not a neighboring state but in the same geographical region, we gave it a score of 0.25. We aggregated the scores for all the files for each state, which is shown in Figure 5.

Next, we wanted to calculate the agreement among participants in choosing labels. If there was high agreement for a particular file but a mismatch with the geographical origin, this could indicate that the speaker could have been an immigrant. If there was low agreement, it could indicate that the accent was difficult to guess. For state-level judgments, we found that there was very poor agreement due to the lack of more than one state-level label for most files. The states Karnataka, Kerala, Maharashtra and Tamil Nadu had only one instance of agreement between users, while Uttar Pradesh, Punjab and Bihar had two. So, we extended this

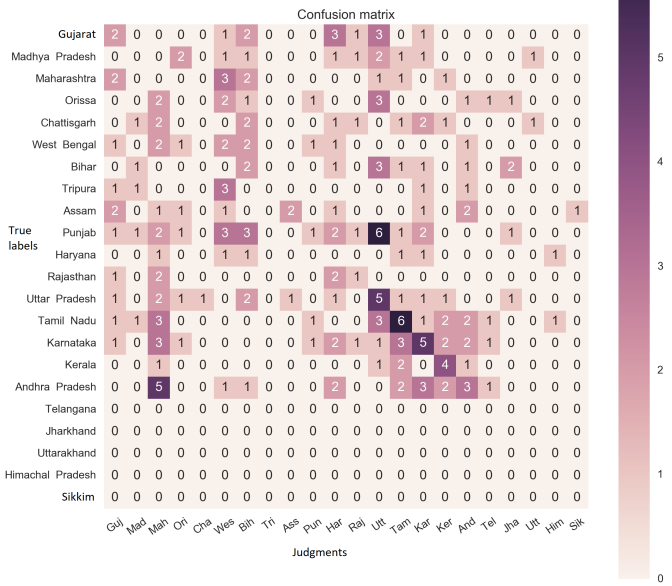


Figure 3: State-level confusion matrix

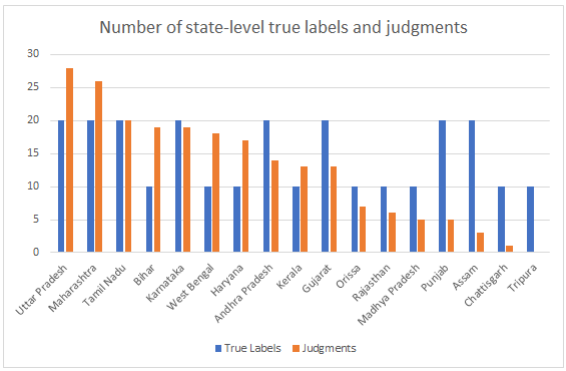


Figure 4: Number of state-level true labels and judgments

score to include region-level agreement as well. If there was any agreement, either at the state or region level for a file, it received a score of one. We added a point to the agreement score for a file for each pair of annotators who agreed, and aggregated this score over each geographical origin state, as shown in Figure 5.

From the figure we can see that some states had high agreement among participants, but a low match with the geographical origin of the audio sample. This could be due to the presence of immigrants in those states.

Although we found some interesting trends, to achieve our original goal of discovering canonical accents in Indian English, we needed to have higher confidence of what the L1 of each speaker was. Going forward, we wanted to scale up the study by using user-reported L1 as the true label instead of using geographical location as a proxy.

In addition, some of the findings of the pilot study may have been influenced by the fact that all participants were either living in or visiting Bangalore, Karnataka, which is why they may have been able to distinguish Southern Indian accents better. Participants also had similar educational and socio-economic backgrounds. So, we wanted to create a web-based application going forward that could potentially be accessed by anyone in the country.

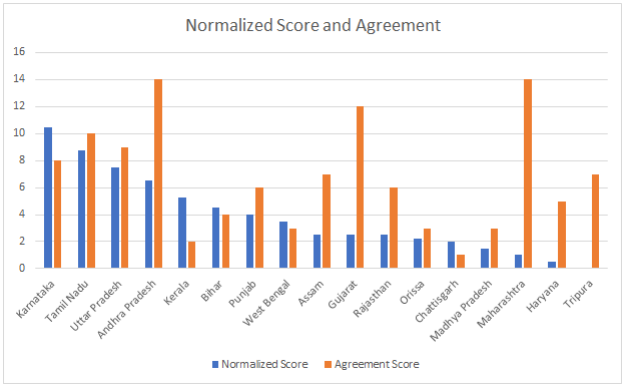


Figure 5: Normalized Scores and Agreement between users

3.3. Feedback from users

As part of the user study, we collected feedback from users about the task and interface. Users felt that the task was difficult and reported that selecting a state was hard compared to selecting a region. However, from our analysis we see that a third of the judgments were state-level.

Users wanted to see their score at the end of the study to know how well they did at identifying where the speakers were from. They also wanted to know which accents (or regions) they did well on, and which ones they got wrong. This was not possible in the pilot study because we did not have ground truth labels for the files. We incorporated this feedback while designing our web-based study.

4. Web-based Study

We designed a web-based study in the form of a game to scale up our pilot study, in which we used the audio recorded by participants of the pilot as the audio examples to play to users. Since we had demographic information about the users who recorded audio including their L1, we could use it as a ground truth label for the accent.

Figure 6 shows the interface of the web-based study. Users were shown a map of the country and a button to toggle a region view or a state view. They could play the audio sample and annotate a region or state on the map that they felt corresponded with the place of origin of the speaker.

Since we had the ground truth labels in this study, we could calculate a score that we could show users at the end of the game. The score was calculated by matching the state or region that the user's L1 and place of birth corresponded to the state or region that was selected by the user. An exact match at the state-level was given a score of 10, while a region-level match got a score of 5. Each user listened to 10 files in the web-based study. We divided scores into 5 equal buckets, each of 20 points and assigned a humorous label to each bucket to show to the users at the end of the game.

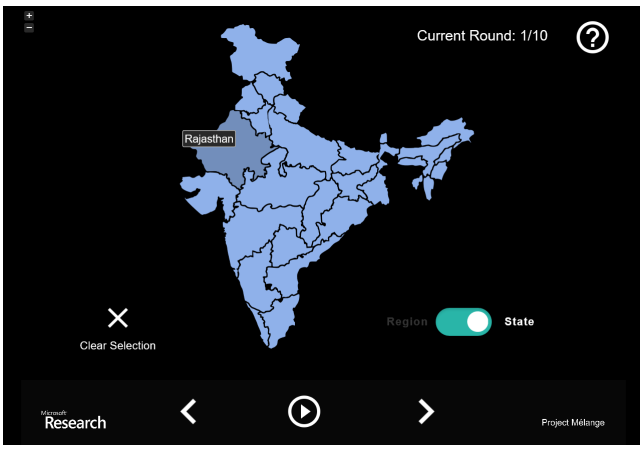


Figure 6: Screenshot of the web-based study interface



Figure 7: Screenshot of the final judgments with correct answers

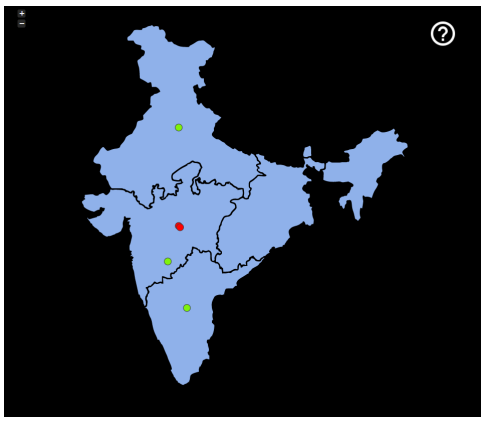


Figure 8: Screenshot of a map with markers indicating correct and wrong judgments

4.1. Feedback from users

10 users from the research lab tested the web-based application. All users said that they enjoyed playing the game, however, their scores fell into the lowermost or second lowest bucket, which meant that they got a score of 40 or less out of 100. This could be attributed to the following reasons. Users said that some of the files had background noise, which was due to the fact that these had been recorded during the pilot on mobile phones. More importantly, users pointed out that the accents were difficult to guess because the speakers were very urban with mild accents. This was due to the fact that we had collected the data used for the web app-based study from participants visiting our research lab. The data used for the earlier mobile app-based study was from users of a speech recognition system which was a more diverse, but presumably quite urban population.

We also received feedback that users wanted to know which accents they got correct and which ones they made mistakes on. We modified the interface to show this to users, by giving them the choice of listening to every file that they got incorrect and by showing them a map with dots in different colors indicating which ones they got correct and wrong. This would allow participants to see if they performed differently on accents from different regions. Figures 7 and 8 show screenshots of what the participants see once they finish the game. Figure 7 shows participants the state or region they selected along with the true state or region, and also gives them the option of listening to the corresponding audio files again. Figure 8 shows a map with green dots indicating correct judgments by state or region, and red dots indicating incorrect judgments.

Going forward, we would like to use the web-based study to scale up both the annotation and collection of accented data. Our first challenge is to make the task easier for users to do by providing them with a diverse set of accents from urban, semi-urban and if possible rural speakers. The purpose of the web based-study is to collect such data automatically by participants uploading their own audio files. To bootstrap this process, we plan to collect some data which includes some strong accents, so that participants are able to identify some accents in the game more easily.

In addition, we also plan to make the game more engaging and fun by showing a leaderboard so that participants within a group (such as a university) can compare scores with each other more easily, if they wish to. Currently, all participants who wish to upload an audio file in their own voice read the same paragraph mentioned in Section 3.1. We plan to allow users to record any sentence they wish to in future iterations of the game, so that we can collect a variety of accented data that can potentially be useful for training ASR models.

5. Discussion

In this paper, we have described a first attempt to identify canonical accents in Indian English. Due to the lack of labeled accented data or a large corpus of speech that covers all regions of India, we used a crowdsourcing-based approach by asking users to identify the geographical origin of a speaker at a state or region level, which we used to represent their L1. In the mobile app-based pilot study, we found that users were able to distinguish North and South Indian accents better than other accents and were able to give state-level judgments a third of the time. The geographical location of speakers residing in states such as Uttar Pradesh, Tamil Nadu and Karnataka were identified with higher accuracies than other states, however, the agreement between participants was low for state-level judgments in general.

To overcome the limitations due to the lack of ground truth about the L1 of the speaker, we used speech data collected during the pilot with self-reported L1 to bootstrap a web-based accent recognition game. We conducted preliminary

user studies with the web-based app, and incorporated user feedback to show file-by-file results, and a map with markers to indicate how well the users performed on accents from different regions of the country. We plan to release the web app to a much larger set of people from all over the country to obtain more judgments and more accented speech data. To overcome the difficulty of providing judgments in the current game due to the presence of mild urban accents, we plan to collect some heavily accented data and use it to bootstrap the game till we can collect data from a diverse set of participants.

In our analysis of the data collected from this study, we plan to focus on the correlation between the reported L1 of the speaker and the judgment provided by participants. In addition, we plan to study the effects of L1 and language exposure of participants on their judgments.

6. References

Aditya Siddhant, Preethi Jyothi, S. G. (2017). Leveraging native language speech for accent identification using deep siamese networks. In *Proceedings of ASRU 2017*.

Banthia, J. K. (2001). *Census of India, 2001*, volume 1. Controller of Publications.

D'Arcy, S. M., Russell, M. J., Browning, S. R., and Tomlinson, M. J. (2004). The accents of the british isles (abi) corpus. *Proceedings Modélisations pour l'Identification des Langues*, pages 115–119.

Huang, Y., Yu, D., Liu, C., and Gong, Y. (2014). Multi-accent deep neural network acoustic model with accent-specific top layer using the kld-regularized model adaptation. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Kachru, B. B. (2005). *Asian Englishes: beyond the canon*, volume 1. Hong Kong University Press.

Kalashnik, O. and Fletcher, J. (2007). An acoustic study of vowel contrasts in north indian english. In *Proceedings of the 16th international congress of phonetic sciences*, pages 953–956.

Lander, T. (2007). Cslu: Foreign accented english release 1.2. *Linguistic Data Consortium, Philadelphia*.

Maxwell, O. and Fletcher, J. (2009). Acoustic and durational properties of indian english vowels. *World Englishes*, 28(1):52–69.

Maxwell, O. (2014). *The Intonational Phonology of Indian English: An Autosegmental-Metrical Analysis Based on Bengali and Kannada English*. Ph.D. thesis, University of Melbourne, School of Languages and Linguistics.

Phull, D. K. and Kumar, G. B. (2016). Vowel analysis for indian english. *Procedia Computer Science*, 93:533–538.

Sahgal, A. and Agnihotri, R. K. (1988). Indian english phonology: A sociolinguistic perspective. *English World-Wide*, 9(1):51–64.

Sirsa, H. and Redford, M. A. (2013). The effects of native language on indian english sounds and timing patterns. *Journal of phonetics*, 41(6):393–406.

Weinberger, S. H. (2014). Speech accent archive. george mason university. *http://accent.gmu.edu*.

Extending Search System based on Interactive Visualization for Speech Corpora

Tomoko Ohsuga, Yuichi Ishimoto, Tomoko Kajiyama
Shunsuke Kozawa, Kiyotaka Uchimoto, Shuichi Itahashi
National Institute of Informatics, National Institute for Japanese Language and Linguistics, The Open University
Gunosy Inc., National Institute of Information and Communications Technology, Tsukuba University
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan
10-2 Midori-cho, Tachikawa City, Tokyo 190-8561, Japan
Walton Hall, Milton Keynes MK7 6AA, United Kingdom
Roppongi Hills Mori Tower, 6-10-1 Roppongi, Minato-ku, Tokyo 106-6125, Japan
4-2-1 Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8577, Japan
osuga@nii.ac.jp, yishi@ninjal.ac.jp, tomoko.kajiyama@open.ac.uk
shunsuke.kozawa@gmail.com, uchimoto@nict.go.jp, itabashi.shuichi.da@alumni.tsukuba.ac.jp

Abstract

This paper describes a search system that we have developed specifically for speech corpus retrieval. It is difficult for speech corpus users to compare and select suitable corpora from the large number of various language resources in the world. It would be more convenient for users if each data center used a common specification system for describing its corpora. With the "Concentric Ring View (CRV) System" we proposed, users can search for speech corpora interactively and visually by utilizing the attributes peculiar to speech corpora. We have already proposed a set of specification attributes and items as the first step towards standardization, and we have added these attributes and items to the large-scale metadata database "SHACHI", then we connected SHACHI to the CRV system and implemented it as a combined speech corpus search system.

Keywords: speech corpus, retrieval, visualization

1. Introduction

Speech corpora are indispensable to speech research; several data centers of language resources have been set up worldwide to meet this demand by serving as a repository for language resources that include various speech corpora. They include the European Language Resources Association (ELRA), the Linguistic Data Consortium (LDC) in the U.S.A., the Chinese LDC / Chinese Corpus Consortium (CCC), the Speech Information Technology & Industry Promotion Center (SiTEC) in Korea, and the Speech Resources Consortium at the National Institute of Informatics (NII-SRC) / Language Resources Association (Gengo Shigen Kyokai; GSK) / Advanced Language Information Forum (ALAGIN) in Japan. The amount of data distributed from such data centers is very large. The diversity of the data gives users more freedom of choice, but it has become difficult to select suitable corpora for the intended purpose from the wide variety of corpora that have been made available. Although it is possible to search for these data from the website of each data center, the metadata of the corpus descriptions are not unified and also we cannot specify speech-specific conditions such as the recording environment in the search, so it is not easy for users to find the necessary corpus. Therefore, it would be more convenient for corpus users if the catalogue specifications of the corpora were standardized among all the various data centers and if the speech corpora could be retrieved by utilizing speech-specific conditions.

We have already proposed an interactive visualization and search system for speech corpora called the "Concentric Ring View (CRV)" system, which simultaneously creates a visual display while performing data retrieval (Itahashi, 2011). Using only a mouse, users can choose appropriate search keys for each of the attributes, and they can easily filter information by adjusting the keys. Retrieved results are displayed inside the rings, and users can filter and browse them in real time. On the other hand, a large-scale database system called "SHACHI" [1] was developed to collect metadata, such as tag sets, formats, and recorded contents, from language resources worldwide (Tohyama, 2008). To combine these two systems, we have revised corpus specifications of SHACHI and added attributes peculiar to speech corpora (Itahashi, 2014). This paper reports the revised version of the CRV system obtained by incorporating about 1200 items of speech corpus information from the SHACHI system.

In the following, we give an outline of the CRV system in Section 2, followed by an outline of the SHACHI metadata database and the attributes of the speech corpus specifications in Section 3. Section 4 presents the revisions of the CRV system, Section 5 describes an experiment carried out to verify the effectiveness of the revised system and its results, and finally, Section 6 concludes the paper.

2. Concentric Ring View (CRV) System

One of the authors previously developed a novel search and display system called the CRV system. This system is an interactive environment for integrating searching and browsing of various items, and its effectiveness has already been shown in applications such as a search system for iPhone applications, a pictorial book of flora, and so forth (Kajiyama, 2014). The CRV system is composed of several concentric rings, each of which corresponds to a selected attribute. The authors have adopted this system as the search system for speech corpora.

[1] http://shachi.org/

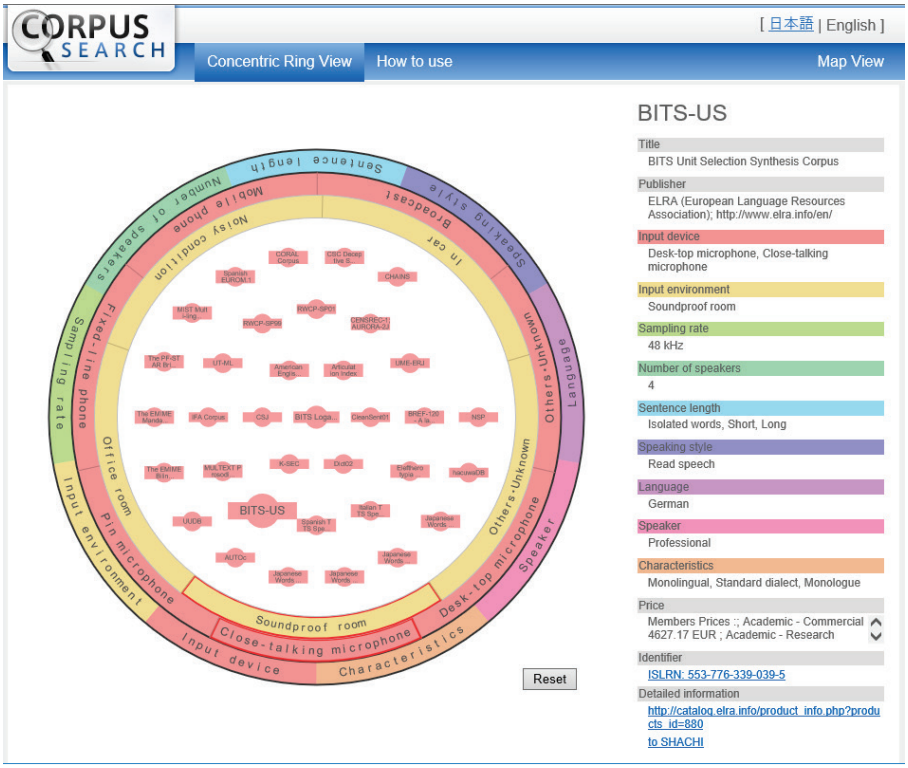


Figure 1: Screenshot of CRV displaying search results for corpora specifying “Close-talking microphone” for “Input device” attribute and “Soundproof room” for “Input environment” attribute, and selecting the “BITS-US” corpus from the corpora in the rings to display its detailed information.

Figure 1 illustrates an example of a screenshot during a search for speech corpora. Initially, only the outermost ring is displayed, which expresses the attributes of the corpora. It is divided into several sectors, each corresponding to an attribute. By clicking a certain sector, another ring, an item ring, appears inside. This item ring contains the category items that correspond to the attribute category specified on the attribute ring. The item ring shown inside has the same color as that of the corresponding sector of the outermost attribute ring. These corpora specified by the attribute and item on the rings are displayed inside the rings. A user can rotate the item ring and adjust the item by dragging a suitable sector of the item ring and browse the search results shown inside the rings. The current item is always shown at the bottom of each ring in a highlighted sector, so the user can easily check the current position or condition. The displayed information can be narrowed down by specifying more attributes, which causes other rings to appear inside. This results in AND retrieval of the specified items, and it is also possible to perform AND retrieval by displaying multiple rings of the same attribute. The search results do not depend on the order of rings displayed. By clicking a displayed ring again, the ring disappears and its retrieval condition can be canceled. This technique allows users to easily and precisely specify each item. The details of a specified corpus are displayed on the right of the screen by clicking the desired corpus shown inside the rings. This is an attribute-based search and users can search for corpora by any attribute in any order.

Table 1 shows the list of attributes that can be used as the retrieval key and the items of each attribute. Initially, only the attribute ring on the left of the table is displayed; by clicking the necessary attribute sector, the corresponding item on the right of the table will be displayed inside the ring. The attribute of “Language” indicates the geographical area of the languages, such as Asia, Europe, and Africa, which were originally used in SHACHI. The specific language name is shown in the detailed information area on the right of the screen, as shown in Fig. 1. Further, specific values for the “Sampling rate” or “Number of speakers” are displayed in the “Detailed information” area.

Of course, many data centers have their own corpus retrieval system that can search the corpora they provide, but they cannot search the corpora by specifying the recording environment or speaking style as shown in Table 1. For example, the items that can be used in the LDC catalogue include “Publication name”, “Author”, “Catalog number”, “Language(s)”, “Member year(s)”, “DCMI type(s)”, “Data source(s)”, “Research project(s)”, “Recommended application(s)” and free keywords. META-SHARE[2], which is an open network of repositories for sharing and exchanging language data and tools, can search by selecting “Media type”, “Language”, and other criteria. CLARIN - European Research Infrastructure for Language Resources and Technology[3] provides a search system using facets including “Resource type”, “Modality”, and “Format”. In contrast, the CRV retrieval system proposed in this paper can retrieve speech corpora utilizing

[2] http://www.meta-share.org/

[3] https://www.clarin.eu/

Table 1: Set of attributes and items for retrieval of speech corpora

Attribute	Item
Input device	Desk-top microphone, Close-talking microphone, Lapel microphone, Fixed-line phone, Mobile phone, Broadcast, Others/Unknown
Input environment	Soundproof room, Office room, Noisy condition, In-car, Others/Unknown
Sampling rate	SR < 10 kHz, SR < 20 kHz, 20 kHz ≦ SR, Unknown
Number of speakers	No < 10, No < 100, No < 1000, 1000 ≦ No, Unknown
Sentence length	Isolated words, Short, Long, Others/Unknown
Speaking style	Read speech, Acted speech, Spontaneous speech, Others/Unknown
Language	Japan, Asia, Europe, Africa, America, Oceania, Others/Unknown
Speaker	Non-native, Professional, Child, Senior, Others/Unknown
Characteristics	Multilingual, Dialect, Dialogue, Emotional, Non-speech, Others/Unknown

speech-specific retrieval conditions; another advantage is that users can manipulate the system visually and that they can add or modify retrieval conditions interactively. In particular, it is useful when users do not know the corpus name and when they want to search corpora suitable for their own objectives or experimental conditions.

3. SHACHI, Large-scale Metadata Database of Language Resources

SHACHI is a large-scale metadata database of language resources developed jointly by the National Institute of Information and Communications Technology (NICT) and Nagoya University of Japan (Tohyama, 2008). It collects detailed metadata information on language resources worldwide, including the resources provided by ELRA, LDC, CCC, SiTEC, NII-SRC, and so forth.

The metadata set adopted by SHACHI conforms to the OLAC[4] metadata set, which is based on 15 fundamental elements of the Dublin Core[5] and constitutes an extended version of OLAC with originally added metadata elements that were considered to be indispensable for describing the characteristics of language resources. It would be more convenient if we could use a set of attributes that describes a corpus to search through the various speech corpora. However, SHACHI mainly focuses on text corpora and does not have sufficient suitable attributes to describe speech corpora. Therefore, we have revised the set of attributes and items. In detail, we have added attributes such as "Input device", "Input environment", "Sampling rate", "Number of speakers", and "Speaking style" to SHACHI (Itahashi, 2014). The attributes of "Sentence length", "Language", and each item of "Speaker" and "Characteristics" were originally used in SHACHI. At present, there are 55 metadata elements in the SHACHI system.

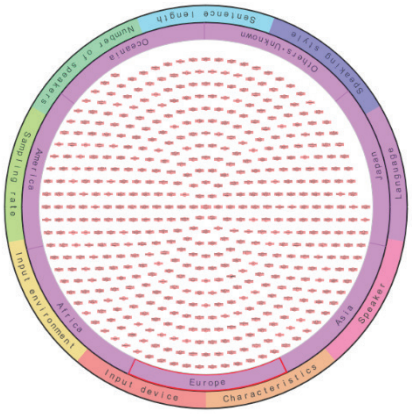


Figure 2: Screenshot of CRV displaying search results for corpora specifying "Europe" for the "Language" attribute.

SHACHI contains 3300 compiled language resources as of Sep. 2017, including 1214 speech corpora. ELRA also has a similar system, i.e., a universal catalogue[6]. It also has its own search function based on keywords, for instance, one can find 749 products among 1645 corpora by specifying the "Speech" keyword as of Sep. 2017. The most important feature of SHACHI is that the automatically collected metadata are manually corrected and thus error-free (Tohyama, 2008). We think that SHACHI is the most suitable database for integration with the search system as it has been extended by adding attributes characterizing speech corpora such as the speech recording environment and speaking style.

4. Connecting SHACHI with CRV Search System and Revision of the Search System

Because it would be much more convenient if SHACHI could be combined with a visual search system such as CRV, we tentatively developed software for extracting the corpus metadata from SHACHI and transferring them to the CRV system.

We confirmed that it is possible to retrieve target corpora using the proposed system in a preliminary estimation experiment using 50 corpora (Itahashi, 2011). We have now incorporated all 1214 speech corpora enrolled in SHACHI in the CRV system. After increasing the number of corpora to be retrieved, however, it turned out that the existing system had a problem in the user interface. The size of the retrieved corpus icons to be displayed inside the ring was fixed in the previous system, and so when there

[4] http://www.language-archives.org/

[5] http://dublincore.org/

[6] http://universal.elra.info/

were many corpora to be displayed, they were divided into multiple pages and the next page could be displayed by rotating the ring. However, users may have found it difficult to understand that the operations of changing the retrieval keys and showing the next page are performed by the same ring operation in the case that each attribute has discrete values. Also, it was not easy to grasp the number of retrieved corpora. Thus, we have devised a new system that can display all the retrieved corpora in a single page by reducing the size of each corpus icon. When there are too many retrieved results, each corpus icon is illustrated with a very small size, suggesting that the user should add more retrieval conditions to reduce the number of retrieved corpora. Figure 2 shows an example of retrieved results when "Europe" is specified as the "Language" attribute. It can be seen that there are too many retrieved results and that it is necessary to add more attributes to reduce the number of retrieved results.

This modification makes it possible for users to intuitively grasp the approximate number of retrieved corpora corresponding to the retrieval items by rotating the ring.

Moreover, we have also added identifiers such as the International Standard Language Resource Number (ISLRN) and the Digital Object Identifier (DOI) as links to each landing page. These items are displayed in the detailed information area, as shown in Fig. 1. Incidentally, detailed information such as the formal title, the publisher, the price, and the URL originally existed in SHACHI, but the ISLRN and DOI links have been newly added to the SHACHI system as one of the present modifications.

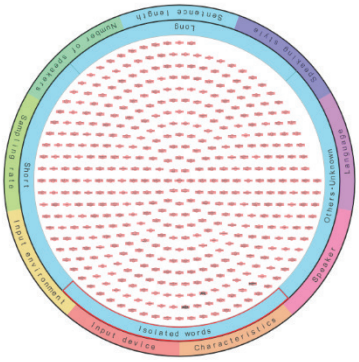
5. Verifying the Extended Search System

We performed a retrieval experiment to verify the effectiveness of the revised search system. Here we show the process of retrieving corpora with read-aloud isolated words under a noisy condition. First, the "Sentence length" attribute in the screenshot of Fig. 3(a) was clicked, and then the inside ring was rotated until the "Isolated words" item reached the bottom as shown in Fig. 3(b). Next, the "Noisy condition" item was selected as the "Input environment" attribute as shown in Fig. 3(c). Because quite a large number of corpora still remained, we added the item "20 kHz ≦ SR" for the "Sampling rate" attribute, which reduced the number of retrieved corpora to four, as shown in Fig. 3(d).

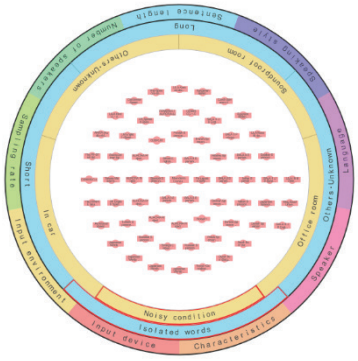
We conducted a preliminary questionnaire survey of 15 people (nine males and six females) who have experience of using speech corpora. After completing a simple retrieval task, we asked them if the proposed system was useful, for which obtained an average value of 3.8 on a five-step scale from one to five. This figure is better than the survey result of 3.2 for our former MDS-based retrieval system (Yamakawa, 2008). The result shows that this system is useful and effective for retrieving speech corpora. We also obtained an average score of 3.5 for "Relevance of attributes", with some comments that it is not easy to predict what items exist in each attribute. We plan to add a "Show attribute" button on the screen so that users can see the table of attributes any time they want. As future work, enrichment of the metadata is necessary to reduce the number of corpora classified into the "Unknown" category for each attribute. Also it is desirable to smoothly navigate among the attributes that should be chosen next for effective retrieval.



(a): Initial screenshot showing only attribute ring.



(b): Specifying "Isolated words" as "Sentence length".



(c): Specifying "Noisy condition" as "Input environment".



(d): Specifying "20 kHz ≦ SR" as "Sampling rate".

Figure 3: Screenshots of CRV display during search process.

6. Conclusions

We have proposed corpus specification attributes and items to give corpus users easy access to the speech corpora catalogues, and we connected SHACHI to our search system for speech corpora. As the search system, we adopted the Concentric Ring View (CRV) system, which simultaneously searches for and displays various objects. Users can search speech corpora interactively and visually by utilizing the attributes peculiar to speech corpora as a web application system. The present system can be accessed at the URL indicated at the end of this paragraph. We plan to continue adding more corpora and to carry out a more elaborate assessment of the proposed system by user questionnaire.

http://corpus-search.nii.ac.jp/ring/

7. Bibliographical References

Itahashi, S., Kajiyama, T., Yamakawa, K., Ishimoto, Y. and Matsui, T. (2011). Interactive visualization search system for speech corpora. *Proc. Oriental COCOSDA 2011*, pp. 157–161.

Itahashi, S., Ohsuga, T., Ishimoto, Y., Kojima, H., Uchimoto, K. and Kozawa, S. (2014). Revised catalogue specifications of speech corpora with user-friendly visualization and search system. *Proc. Oriental COCOSDA 2014*, pp. 60–64.

Kajiyama, T. and Satoh, S. (2014). An interaction model between human and system for intuitive graphical search interface. *International Journal of Knowledge and Information Systems*, 39(1) : 41–60.

Tohyama, H., Kozawa, S., Uchimoto, K., Matsubara, S. and Isahara, H. (2008). Construction of a metadata database for efficient development and use of language resources. *Proc. LREC 2008*, pp. 1687–1692.

Yamakawa, K., Matsui, T. and Itahashi, S. (2008). MDS-based visualization method for multiple speech corpora. *Proc. INTERSPEECH2008*, pp.1666–1669.

German Radio Interviews: The GRAIN Release of the SFB732 Silver Standard Collection

Katrin Schweitzer, Kerstin Eckart, Markus Gärtner, Agnieszka Falenska, Arndt Riester, Ina Rösiger, Antje Schweitzer, Sabrina Stehwien, Jonas Kuhn

University of Stuttgart, Institute for Natural Language Processing (IMS)
Pfaffenwaldring 5B, 70569 Stuttgart, Germany
{firstname.lastname}@ims.uni-stuttgart.de

Abstract

We present GRAIN (German RAdio INterviews) as part of the SFB732 Silver Standard Collection. GRAIN contains German radio interviews and is annotated on multiple linguistic layers. The data has been processed with state-of-the-art tools for text and speech and therefore represents a resource for text-based linguistic research as well as speech science. While there is a gold standard part with manual annotations, the (much larger) silver standard part (which is growing as the radio station releases more interviews) relies completely on automatic annotations. We explicitly release different versions of annotations for the same layers (e.g. morpho-syntax) with the aim to combine and compare multiple layers in order to derive confidence estimations for the annotations. Therefore, parts of the data where the output of several tools match can be considered clear-cut cases, while mismatches hint at areas of interest which are potentially challenging or where rare phenomena can be found.

Keywords: corpus, silver standard, text-speech-interface

1. Introduction

Over the years, many resources for natural language such as tools or corpora have been developed and are used in research and applications. Processing is usually focused on a specific type of primary data, which we call *canonical* data for the respective tool type or branch of research. However, with a large set of efficient tools available, the time has come to make the step beyond canonical data, at the same time connecting research branches such as those based on text and speech. The *SFB732 Silver Standard Collection* is intended to serve as a resource in this respect. It is a non-static collection, i.e. more and different primary data are added over time, as are different annotation layers.

This paper focuses on the first released part of the data set, containing richly annotated German radio interviews, made available for research and education in the GRAIN release of the collection. Different research groups contributed in its curation, among them were groups usually working primarily with written language data as well as groups who usually focus on speech, i.e. the tools applied are state-of-the art tools from both disciplines. Thereby, the resource can provide a starting point for joint exploration of speech and text. Moreover, bringing together tools from different research branches can result for some tools in encountering data that is non-canonical. For instance, parsers trained on newspaper text are to deal with incomplete sentences when being presented with spoken language data.

In the remainder of the paper, we first present the idea of a *Silver Standard*, relying on automatic annotations, then we describe the GRAIN release: the primary data is described in Section 3, details about the existing annotations can be found in Section 4. Section 5 outlines how the workflow of creating the resource is documented, Section 6 details its availability. Annotations currently under development are referred to in Section 7 and Section 8 concludes.

2. Silver Standard

The size and non-static nature of the data set make it inherently unfeasible to provide manual annotations for the entire resource. Instead only a small subset was chosen to be covered by gold standard annotations (see Section 3) and the remaining parts received automatically created annotations. For several automatic annotations we employed a "silver standard" approach (Rebholz-Schuhmann et al., 2010).

The idea behind this is that it is possible to provide a level of annotation quality, that is better than unchecked output of automated processing, even though it might not reach gold standard. To this end, automatic output has been enriched with additional confidence estimations (Gärtner and Eckart, 2017) that serve as quality indicators to increase the usability of the data. In this way, users can a) gauge the quality of the data they are working with, b) select subsets of the data where the annotations come with high confidence estimation or c) find areas of interest, for instance, when the confidence of the system is low.

Most automatic annotation systems do not provide quality indicators in their output even if they are internally aware of the relative reliability of their annotations, as is the case for all systems using probabilistic methods, e.g. stochastic parsers. Therefore, we relied on external methods for estimating confidence values, some of which have been presented in Eckart and Gärtner (2016). Note that all confidence estimations are provided as additional (meta-)-annotation layers and therefore can be used directly with regular linguistic features for visualization or search in tools such as ICARUS (Gärtner et al., 2013).

3. Primary Data

The primary data consists of German radio interviews, with a duration of just under 10 minutes each. For each in-

cluded interview an audio file (*.mp3*) and an edited transcript (mostly *.pdf*, sometimes *.doc*) were available from the radio station. The transcript has been intensely edited by the radio station to produce a version of the interview for reading and thereby omits features of orality, e.g. by excluding slips of the tongue and repetitions and by rephrasing utterances to adhere to written syntax (see Eckart and Gärtner (2016) for an example). Based on the *.pdf* and *.doc* files, which we consider raw data, primary data for the collection was extracted in the form of UTF-8 encoded plain text files.[1]

The audio files are heterogeneous in their characteristics: there are stereo as well as mono files, with either 44.1kHz sampling rate or 48kHz, and with varying audio bitrates (64-135kbps). The release contains the original mp3 files. For processing the files with our tools, we converted them to 16kHz mono wav-files. These more consistent files are made available, as well. Note that the basis though is mp3, i.e. the wav-files do not provide better quality. Each interview involves two speakers, a host and a guest. The guest appears in a professional role, and the questions of the host usually refer to a current political or social discussion at the time of recording.

Gold standard part and training interviews. A set of 20 interviews has been selected to serve as primary data for a gold standard part of the collection. Three additional interviews have been marked as training interviews for annotators being introduced to guidelines for manual annotation. Since the annotation efforts are conducted by several projects, globally defining these interviews within the collection minimizes the set of interviews for which specific restrictions apply, e.g. in evaluation settings.[2] The interviews for the gold standard part were balanced as well as possible with respect to sex and variety of the host[3] and sex and role of the guest[4].

Size of the dataset. Since the collection is non-static more interviews are added over time. The current status is 144 interviews, with about 221,000 word tokens and a duration of about 23 hours.

4. Available Annotation Layers

The GRAIN release provides annotation layers resulting from state-of-the-art text and speech processing and is therefore suited for examinations from either spoken or written language research, as well as studies at the interface of the two. The text-based annotation layers are linked with LAF anchors, the annotations referring to the audio signals are anchored via timestamps. The mapping between the two is done on the basis of word tokens.

[1] Apache PDFBox 1.8.7.

[2] It is of course possible to include further training interviews if needed or to annotate a training interview with gold quality.

[3] Two (of at that time three encountered) female hosts with five guests each, five (of at that time ten encountered) male hosts with two guests each.

[4] For the balancing we distinguished between guest interviewed w.r.t. to a political or a non-political role, i.e. the data is balanced for their role in the interview, not their usual background or former professions.

In what follows we discuss the annotations that are part of this GRAIN release.

4.1. Automatic Annotations

Automatic annotations are created for the entire dataset, i.e. all interviews for which the radio station provided both the audio file and the edited transcript. This part of the corpus, which contains only automatic annotations is what we refer to as the silver-standard part of the release, since it offers the possibility do combine various annotation levels to gauge the quality of the annotations by e.g confidence estimations.

Preprocessing. As a first step, anchors according to LAF (ISO 24612:2012) were introduced for the primary data text files. The base units of these files are UTF-8 characters, each identified by two numerical anchors, describing the one-character span in the data. The anchors allow for several layers of (stand-off) annotations to be linked to the primary data document. Further annotations for the textual data include speaker turn spans and document structure. Based on the described information available, several input formats for subsequent processing steps could be created.

Tokenizing and sentence segmentation. The data was tokenized with the TreeTagger (Schmid, 1994). Sentence segmentation was done on top, based on punctuation tokens.

Acoustic segmentation and alignment. The data was force-aligned for phone, syllable and word boundaries (Rapp, 1995).

Parametrized intonation events. PaIntE parameters were calculated for each syllable in the data (Möhler, 2001; Möhler and Conkie, 1998; Möhler, 1998). PaIntE stands for "Parametrized Intonation Events" and presents a way to describe the shape of local maxima in the pitch contour (that is, highly probable candidates for pitch accents or boundary tones) by means of six parameters. Parametrization is carried out using a function over time which approximates the fundamental frequency contour. The function comprises six free parameters that are fitted in such a way that the actual fundamental frequency curve is matched best. All six parameters are linguistically interpretable: parameter `d` corresponds to the height of the peak in Hertz, parameter `b` encodes its temporal anchoring within a three syllable window where the syllables are normalized for time, such that the current syllable ranges between 0 and 1. Parameters `c1` and `c2` stand for size of the increase before and the decline after the peak, again in Hertz, and parameters `a1` and `a2` encode the gradient of the rise and fall, respectively. Figure 1 (adapted from Möhler (2001)) displays the parameters in the 3-syllable window.

PaIntE-based prediction of intonation event types. Intonation events, in terms of GToBI(S) labels (Mayer, 1995) for pitch accents and boundary tones, were annotated automatically with the procedure described in Schweitzer (2010). This method takes into account PaIntE parametrizations and normalized phone durations, phonological features and higher linguistic information.

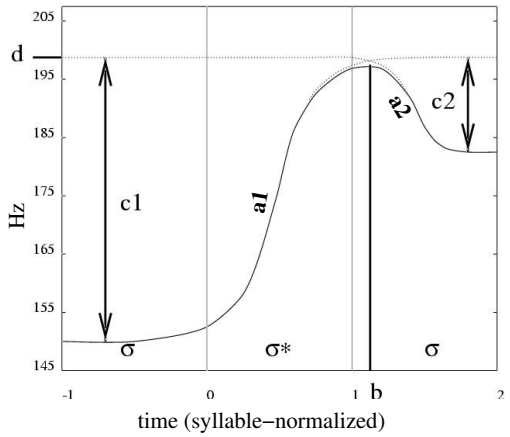


Figure 1: The PaIntE function. Approximation takes place over three syllables. $\sigma*$ marks the syllable for which approximation is currently being carried out. In the depicted case, the parameter setting corresponds to a peak in the following syllable (encoded in parameter b) with a pronounced rise before the peak (c1) and a small fall after (c2). Absolute peak height is encoded in the value of parameter d, and the gradients of the rise and fall can be derived from parameters a1 and a2.

CNN-based prediction of pitch accent placement. The annotations in this layer consist of binary placement information for pitch accents at the word level. Even though no type of pitch accent was assigned, this layer makes it possible to apply the silver standard idea to prosodic annotations: placement of the automatically predicted pitch accent labels derived from PaIntE features (see above) can be compared and combined with this layer for confidence estimations. Speech signals and time aligned word labels were input to a convolutional neural network-based binary classifier (CNN) that predicts for each word whether it carries a pitch accent or not (Stehwien and Vu, 2017). The input to the CNN was a frame-based representation of the speech signal using low-level acoustic descriptors extracted using OpenSMILE (Eyben et al., 2013). The model was trained on the German radio news corpus DIRNDL (Eckart et al., 2012).

Agreement on pitch accent placement. Since GRAIN contains no manually annotated intonation events, the prediction accuracy could only be estimated on a small amount of data labeled by a human expert. In the following, we report the agreement with respect to word-level pitch accent placement between the two annotation layers (PaIntE-based and CNN-based) and compared to human annotation. Table 1 shows the performance of the two tools measured against the reference annotations. In terms of accuracy, both tools provide annotations of similar quality. The CNN-based tool yields a higher recall, while the PaIntE-based annotations have a higher precision. Table 2 compares the precision on either label classes *accent* and *none* obtained using both methods and when counting only the labels that the two tools agree on. These results show that in cases where both annotations agree wrt presence or absence of pitch accents, the precision increases considerably for both classes (up 10% for the accent class) compared to when used alone. While this evaluation can only provide an estimation of performance (especially when using only one human labeler), the reported numbers do show that both tools, using entirely different methods of prosodic modeling, complement each other and that using both can be used to estimate the confidence of automatic annotation. It also demonstrates our idea of a silver standard: combining two layers of automatic annotation on the same annotation level achieves a better quality than just one level alone.

	PaIntE-based	CNN-based
accents	494	629
accuracy	80.9%	80.0%
precision	67.2	62.2
recall	65.2	76.8
F1-score	66.2	68.7

Table 1: Agreement with human labeling of word-level pitch accent placement: The PaIntE-based and CNN-based automatic annotations are compared on one example file containing 1778 words and 509 pitch accents.

class	PaIntE-based	CNN-based	both
accents	67.2	62.2	76.0
none	86.2	89.7	94.2

Table 2: Precision for both label classes *accent* and *none* obtained using either pitch accent labeling method and when taking only the labels into account on which both tools agree.

Additional phonetic features. The data was preprocessed using the Festival (Black, 1997) version of the University of Stuttgart (IMS Festival, 2010) to retrieve some of the features needed in the automatic annotation process for prosodic events. We release some of the syllable-based features, for convenience. We provide the duration of each syllable, its position in the word, and the number of phonemes in onset and rhyme, as well as the Van Santen/Hirschberg Classification (van Santen and Hirschberg, 1994)
of onset and rhyme.

Morpho-syntax. On the morpho-syntactic level we employed a series of very different pipeline implementations (BitPar (Schmid, 2006; Schmid, 2004), IMS-SZEGED-CIS (Björkelund et al., 2013), Mate (Bohnet and Nivre, 2012; Bohnet, 2010), IMSTrans (Björkelund and Nivre, 2015; Björkelund et al., 2016) and Stanford CoreNLP components such as the Stanford Parser (Chen and Manning, 2014)) to generate automatic parses and underlying morpho-syntactic annotations for the entire data set (see Table 3). Since we do not have (morpho-)syntactic gold standard annotations for this release, we post-processed the automatic system output to improve its usability, but without actually changing or correcting it. That is, following the idea of a silver-standard, we introduced additional confidence estimations as meta annotations for individual predictions based on the agreement between different systems (see Figure 2a for an example of trees predicted by three different parsing systems for the same sentence and Figure 2b for corresponding global confidence estimations).

While this extra step does not directly increase the annotation quality per se, it provides valuable information about the relative reliability of individual annotations. Researchers can then use those indicators to find data points which might be of interest or should be ignored for certain research questions.

System	Constituency	Dependency
BitPar	+	
IMS-SZEGED-CIS	+	+
Mate		+
IMSTrans		+
Stanford Parser	+	+

Table 3: List of automatic (pipeline) systems for parsing used to generate concurrent annotations for the corpus.

4.2. Manual Annotations

Manual annotations were conducted on the interviews of the gold standard part of the collection. That is, the manual annotations are additional annotations (not corrections) besides the automatic annotations. We refer to the part of the corpus for which manual annotations are available as the gold-standard part of this release. This part constitutes a subset of the silver-standard part, but has been labeled independently from any automatic annotations.

Unnormalization. To provide a textual version of the interviews suited for several processing pipelines, the edited versions were modified (cf. Eckart and Gärtner (2016) for a motivation and additional details of this additional layer): Based on the audio signal, some features of orality were re-introduced. However, fillers and partially uttered words were not included. This resulted in transcripts that are slightly closer to an orthographic transcription of the utterances as compared to the edited versions from the radio station. We call this process *unnormalization*.[5] Guidelines have been defined and each interview was modified independently by two annotators and adjudication was then done by a third person.
In Eckart and Gärtner (2016) we quantified the difference between the edited version provided by the radio station and the unnormalized versions in terms of the quality of automatic parsing. For the current release of GRAIN, we additionally computed a raw measure of difference between the edited versions and the result of our unnormalization. Using Levenshtein Distance on entire interviews and treating each token as a symbol we calculated edit distances that ranged between 21 and 148 with an average of about 54.
The manual annotations described in the following sections have been conducted on the unnormalized version of the interviews. For details on which automatically generated annotations also use this layer we refer to the documentation that is part of the release.

Part-of-speech tagging. The interviews were annotated with part-of-speech labels based on the STTS guidelines (Schiller et al., 1999) including the modifications from the TIGER corpus (Albert et al., 2003). Some additional guidelines were set up for the interview corpus (Seeker, 2016) but due to the specificities of the interviews no further categories were needed (cf. Westpfahl et al. (2017) for a broader set for spoken data). Three annotators were involved in the process and each interview was annotated by two of them independently, applying the Synpathy tool[6]. The annotators achieved pair-wise agreement with a Cohen's κ of 0.97, ranging between 0.96 and 0.98. In an adjudication step all three annotators then decided on the annotation, and remaining hard cases were discussed in the project context and documented separately. After all interviews had been manually annotated and discussed, an implementation of the DECCA-Tools (Dickinson and Meurers, 2003) in ICARUS (Thiele et al., 2014) was applied to the interviews, automatically finding potential cases of inconsistent annotation.

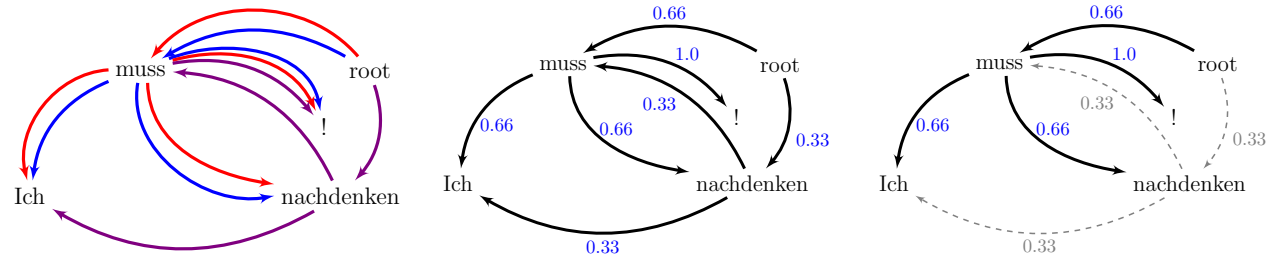
Referential information status. From the gold part 20 interviews and the three training interviews were annotated with referential information status (Baumann and Riester, 2012), following the guidelines in Riester and Baumann (2017). This means that all referring expressions in the interviews (and a number of verb phrases and sentences functioning as antecedents for abstract anaphors) were categorized as to whether they are given/coreferential, bridging anaphors, deictic, discourse-new, idiomatic etc. The interviews furthermore contain coreference chains and bridging links. Each of the interviews was annotated independently by two annotators, applying the Slate tool (Kaplan et al., 2012). Adjudication was either done by a third person, or in a discussion round of the project group.
The inter-annotator-agreement has been computed for markables with the same span, where we have achieved substantial agreement, with a Cohen's κ of 0.75. Five different annotators were involved in the annotation (all students of computational linguistics) and the pair-wise agreement for different annotator pairs (Cohen's κ) ranges between 0.64 and 0.82. For more details on the inter-annotator agreement, please refer to Pagel (2018) and Draudt (2018).

Questions under discussion (QUD). From the gold part, ten interviews and the three training interviews were analyzed according to the *QUD-tree* method (Reyle and Riester, 2016; Riester et al., to appear), which involves a new (sub-sentential) text segmentation into information-structurally relevant discourse units, the reconstruction of implicit *questions under discussion* (QUDs) for each unit, based on a number of pragmatic principles, and the construction of question-based discourse trees (QUD trees) with TreeAnno (De Kuthy et al., 2018). The annotations were created by two annotators. Adjudication was subsequently done within the project group. More detail and evaluation of the annotation of QUDs can be found in De Kuthy et al. (2018).

Information structure. As described in Riester et al. (to appear), the QUD-tree method is a joint approach for the

[5] This term is inspired by a step called normalization often applied to map non-canonical representations to some sort of standard or processable forms.

[6] http://www.mpi.nl/tools/synpathy.html



(a) Predictions from three different parsers. (b) Confidence estimations. (c) Merged tree.

Figure 2: An example sentence "Ich muss nachdenken!" (eng. "I have to think!") and different layers of automatically predicted dependency annotations.

analysis of both discourse structure and information structure. The implicit QUDs define which parts of the discourse units receive either of the labels focus, contrastive topic, background and non-at-issue. We used the annotation tool Slate (Kaplan et al., 2012) for this annotation task. For more details on the annotation of discourse structure and information structure we also refer to De Kuthy et al. (2018).

5. Documentation

Besides the primary data and our various annotation layers we also created detailed documentation for the entire workflow of resource creation. The specific version of each annotation tool, the versions and nature of data used to configure or train it and also the settings used for the actual analysis are all crucial information needed to properly evaluate the output and its suitability (in this case of the final corpus resource) in the context of a certain research question. We therefore used a simple metadata scheme similar to the one proposed by Gärtner et al. (2018) for recording *process metadata*.

Metadata for manual annotations includes amongst other information annotator ids[7], details of manual curation as well as applied annotation guidelines. In the case of automatic steps the recorded metadata is very similar and additionally contains version information for involved resources and/or tools, where available. Figures 3 and 4 show (condensed) instances of this metadata for manual and automatic processing steps, respectively. The entire process metadata is available as part of the corpus release together with the overall documentation.

6. Availability

Due to the high number of different tools involved in the creation of GRAIN and in order to accommodate researchers from different communities, various representation formats are part of this release. They contain, for instance, popular tabular formats such as those used in the CoNLL Shared Tasks of 2009 (Hajič et al., 2009) and 2012 (Pradhan et al., 2012) and extended versions (Björkelund et al., 2014), XML-based formats such as TIGER XML (König et al., 2003), or Praat TextGrids (Boersma, 2001). For an exhaustive list of formats used and annotation layers contained in them we refer to the official documentation

[7] In anonymized form.

```
[result: ["swr2-interview-der-woche-20150620.txt-mod"],
 input: ["swr2-interview-der-woche-20150620.txt","swr2-
     interview-der-woche-20150620.6444m.mp3"],
 workflowSteps:[
   [description: "transcription modification",
    mode: "manual",
    operators:[
      [name: "OP01",
       components:[
         [name: "transcription-change-guidelines",
          version: "1.1.0",
          type: "guidelines"
         ]]
      ]
      [name: "OP02",
       components:[
         [name: "transcription-change-guidelines",
          version: "1.1.0",
          type: "guidelines"
         ]]
      ]]
   ]
   [description: "conflict resolution",
    mode: "manual",
    operators:[
      [name: "OP04",
       components:[
         [name: "transcription-change-guidelines",
          version: "1.1.0",
          type: "guidelines"
         ]]
      ]]
   ]]
]
```

Figure 3: Example process metadata for two independent manual annotation steps by different annotators (`operators`) for unnormalization and a subsequent adjudication step by a third person.

that is part of the resource. Individual annotation files in the corpus also follow a simple naming scheme that contains the part of the primary data the annotations are associated with and reflects the processing step that created the data. The release, as well as a detailed documentation is published in the framework of CLARIN[8] and available via a persistent identifier[9] in order to ensure sustainability.

7. Annotation Layers under Development

The SFB732 Silver Standard Collection is non-static, i.e. primary data and annotation layers will continue to be added. The annotations currently under development or planned for future releases are listed below.

[8] `https://www.clarin.eu/`

[9] `http://hdl.handle.net/11022/1007-0000-0007-C632-1`

```
{"result": ["swr2-interview-der-woche-20140517.3.2.1.0.
    tag-tt"],
 "input": ["swr2-interview-der-woche-20140517.1.0.0.0.
     tok-tt-mod"],
 "workflowSteps":[
   {"description":"TreeTagger: lexicon lookup, part-of-
       speech and lemma annotations, error correction",
    "mode": "automatic",
    "operators":[
      {"name": "tree-tagger-3.2.1-german-notok-nosgmlrec
          ",
       "version": "3.2.1 (TreeTagger)",
       "parameters": "",
       "components":[
         {"name": "german-lexicon-utf8.txt",
          "version": "3.2.1 (TreeTagger)",
          "type": "lexicon"
         },
         {"name": "german-utf8.par",
          "version": "3.2",
          "type": "parameter file"
         }]
      }]
   }]
}
```


Figure 4: Example process metadata for an automatic processing step consisting of part-of-speech tagging and lemmatization with a subsequent automatic error correction.

7.1. Automatic Annotations

In addition to the annotations listed in Section 4.1 we plan to include the following annotations in subsequent releases of the silver standard part of the corpus.

Merged dependency parses. As part of the parsing outputs described in Section 4.1 several parallel dependency trees are available for every sentence. While this provides a rich foundation for comparison, it can also be challenging for users to work with multiple concurrent trees. We will therefore provide additional merged versions of dependency trees. That is, we will include a majority decision of the dependency parsers under tree constraints.
This is done by employing blending, also known as reparsing (Sagae and Lavie, 2006). We combine all the silver standard trees for a sentence into one graph and assign scores to arcs depending on the confidence estimations (see Figure 2b for an example of a combined graph). We then use the Chu-Liu-Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967) to find the maximum spanning tree in the combined graph (see Figure 2c for the resulting maximum spanning tree). For every resulting arc we select the most frequent label across all the labels previously assigned to it. This additional layer of automatic annotations has two purposes. It improves the usability of the syntactic layer for users who may prefer to work with only a single dependency tree instead of multiple predicted ones. Secondly, it increases the reliability of the syntactic annotations, since Björkelund et al. (2017) showed that blending can achieve higher performance than single parsers.

CNN-based prediction of boundary tone placement. In addition to the pitch accent placement labels predicted using a CNN-based classifier (described in Section 4.1), a similar model (extended to include duration and pause information) will be trained to label each word as bearing a phrase boundary tone or not. This model will require more annotated data from additional English sources e.g. from BURNC (Ostendorf et al., 1995).

7.2. Manual Annotations

The manual annotation layers under development will be added to the gold standard subset of the corpus, which was also used for the annotations described in Section 4.2.

Unedited orthographic transcripts. Currently, orthographic transcriptions of another granularity are being created, additionally to the edited version provided by the radio station (see Section 3) and the "unnormalized" version which was used as a basis for the text-based manual annotations (see Section 4.2). This version is as close to the audio files as possible, i.e. it contains fillers and other non-lexical information, partially uttered words, mispronunciations, non-standard pronunciations etc., and gives information about overlap between the speakers. Guidelines have been defined which are based on the guidelines used for the GECO corpus (Schweitzer and Lewandowski, 2013) and some aspects of the definition of the "verbal tier" in the HIAT guidelines (Rehbein et al., 2004).

8. Conclusion

We presented the GRAIN release of the SFB732 Silver Standard Collection. The data comprises audio files of German radio interviews and their transcripts provided by the broadcasting station. We provide (manual) gold standard annotations for a subset of 20 radio interviews. These annotations include a transcript which is closer to the audio files than the edited transcript, POS tags, referential information status annotations, and questions under discussion. Additionally, a much larger data set (currently 160 interviews) has been annotated with silver standard annotations, i.e. automatic annotations which can be combined and compared, both across as well as between layers, in order to make it possible to infer a confidence estimation for the annotations. These annotations include information about speakers and their roles, time-aligned word, phone and syllable labels, parametrized intonation events, GToBI(S) intonation labels, CNN-based annotation of pitch accents, additional syllable features and morpho-syntactic annotations. For the syntax annotations, confidence estimations are already provided in this release. The silver standard part of the data is growing: as the radio station releases more interviews, they are being collected and automatically processed. , new annotation layers are currently being created. These will comprise automatic annotations in the form of merged dependency parses, CNN-based boundary tone placement as well as manually created information structure labels and a version of the transcript with all features of orality.

9. Acknowledgements

We would like to thank the numerous dedicated annotators for their contributions. This work was funded by the German research foundation (DFG) via SFB 732, projects INF, A4, A6, A8 and D8.

10. Bibliographical References

Albert, S., Anderssen, J., Bader, R., Becker, S., Bracht, T., Brants, S., Brants, T., Demberg, V., Dipper, S., Eisenberg, P., Hansen, S., Hirschmann, H., Janitzek, J., Kirstein, C., Langner, R., Michelbacher, L., Plaehn, O., Preis, C., Pußel, M., Rower, M., Schrader, B., Schwartz, A., Smith, G., and Uszkoreit, H., (2003). *TIGER Annotationsschema*. Universität des Saarlandes, Universität Stuttgart, Universität Potsdam.

Baumann, S. and Riester, A. (2012). Referential and lexical givenness: Semantic, prosodic and cognitive aspects. In Gorka Elordieta et al., editors, *Prosody and Meaning*, pages 119–161. De Gruyter Mouton, Berlin.

Björkelund, A. and Nivre, J. (2015). Non-Deterministic Oracles for Unrestricted Non-Projective Transition-Based Dependency Parsing. In *Proceedings of the 14th International Conference on Parsing Technologies*, pages 76–86, Bilbao, Spain, July. Association for Computational Linguistics.

Björkelund, A., Cetinoglu, O., Farkas, R., Mueller, T., and Seeker, W. (2013). (re)ranking meets morphosyntax: State-of-the-art results from the SPMRL 2013 shared task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145, Seattle, Washington, USA, October. Association for Computational Linguistics.

Björkelund, A., Eckart, K., Riester, A., Schauffler, N., and Schweitzer, K. (2014). The extended DIRNDL corpus as a resource for coreference and bridging resolution. In Nicoletta Calzolari et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 3222–3228, Reykjavik.

Björkelund, A., Faleńska, A., Seeker, W., and Kuhn, J. (2016). How to train dependency parsers with inexact search for joint sentence boundary detection and parsing of entire documents. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1924–1934, Berlin, Germany, August. Association for Computational Linguistics.

Björkelund, A., Falenska, A., Yu, X., and Kuhn, J. (2017). Ims at the conll 2017 ud shared task: Crfs and perceptrons meet neural networks. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 40–51. Association for Computational Linguistics.

Black, A. W. (1997). The Festival speech synthesis system. www.cstr.ed.ac.uk/projects/.

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345.

Bohnet, B. and Nivre, J. (2012). A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465, Jeju Island, Korea, July. Association for Computational Linguistics.

Bohnet, B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In Alessandro Moschitti, et al., editors, *EMNLP*, pages 740–750. ACL.

Chu, Y. and Liu, T. (1965). On the shortest aborescence of a directed graph. *Science Sinica*, 14:1396–1400.

De Kuthy, K., Reiter, N., and Riester, A. (2018). QUD-based annotation of discourse structure and information structure: Tool and evaluation. In *Proceedings of the 11th Language Resources and Evaluation Conference (LREC)*, Miyazaki, JP.

Dickinson, M. and Meurers, W. D. (2003). Detecting errors in part-of-speech annotation. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, pages 107–114, Budapest, Hungary. Association for Computational Linguistics.

Draudt, A.-C. (2018). "Inter-Annotator Agreement von Informationsstatus-Annotationen im GRAIN-Korpus" (BSc thesis).

Eckart, K. and Gärtner, M. (2016). Creating Silver Standard Annotations for a Corpus of Non-Standard Data. In Stefanie Dipper, et al., editors, *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, volume 16 of *BLA: Bochumer Linguistische Arbeitsberichte*, pages 90–96, Bochum, Germany.

Eckart, K., Riester, A., and Schweitzer, K. (2012). A discourse information radio news database for linguistic analysis. In Christian Chiarcos, et al., editors, *Linked Data in Linguistics. Representing and Connecting Language Data and Language Metadata*, pages 65–75. Springer, Heidelberg.

Edmonds, J. (1967). Optimum branchings. *Journal of Research of the National Bureau of Standards*, 71(B):233–240.

Eyben, F., Weninger, F., Groß, F., and Schuller, B. (2013). Recent developments in openSMILE, the Munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 835–838.

Gärtner, M. and Eckart, K. (2017). In support of self-assessment – exploiting available information from tools. Poster accepted at the DGfS CL Poster Session 2017, Saarbrücken.

Gärtner, M., Thiele, G., Seeker, W., Björkelund, A., and Kuhn, J. (2013). ICARUS – an extensible graphical search tool for dependency treebanks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Sofia, Bulgaria, August. Association for Computational Linguistics.

Gärtner, M., Hahn, U., and Hermann, S. (2018). Supporting sustainable process documentation. In Georg Rehm et al., editors, *Language Technologies for the Challenges of the Digital Age*, pages 284–291, Cham. Springer International Publishing.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D.,

Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., and Zhang, Y. (2009). The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, CoNLL '09, pages 1–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

IMS Festival. (2010). IMS German Festival home page.

Kaplan, D., Iida, R., Nishina, K., and Tokunaga, T. (2012). Slate - A Tool for Creating and Maintaining Annotated Corpora. *Journal for Language Technology and Computational Linguistics*, 26(2):89 – 101.

König, E., Lezius, W., and Voormann, H., (2003). *TIGERSearch 2.1 User's Manual. Chapter V - The TIGER-XML treebank encoding format*. IMS, Universität Stuttgart.

Mayer, J. (1995). Transcribing German intonation – the Stuttgart system. Technical report, Universität Stuttgart.

Möhler, G. and Conkie, A. (1998). Parametric modeling of intonation using vector quantization. In *Proceedings of the Third International Workshop on Speech Synthesis (Jenolan Caves, Australia)*, pages 311–316.

Möhler, G. (1998). Describing intonation with a parametric model. In *Proceedings of the International Conference on Spoken Language Processing*, volume 7, pages 2851–2854.

Möhler, G. (2001). Improvements of the PaIntE model for F_0 parametrization. Technical report, Institute of Natural Language Processing, University of Stuttgart. Draft version.

Ostendorf, M., Price, P. J., and Shattuck-Hufnagel, S. (1995). The Boston University Radio News Corpus. Technical Report ECS-95-001, Electrical, Computer and Systems Engineering Department, Boston University, Boston, MA.

Pagel, J. (2018). Rule-based and Learning-based Approaches for Automatic Bridging Detection and Resolution in German (MSc thesis).

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *EMNLP-CoNLL: Shared Task*, pages 1–40, Jeju Island, Korea, July.

Rapp, S. (1995). Automatic phonemic transcription and linguistic annotation from known text with Hidden Markov models—An aligner for German. In *Proc. of ELSNET Goes East and IMACS Workshop "Integration of Language and Speech in Academia and Industry" (Russia)*.

Rebholz-Schuhmann, D., Jimeno-Yepes, A. J., van Mulligen, E. M., Kang, N., Kors, J., Milward, D., Corbett, P., Buyko, E., Tomanek, K., Beisswanger, E., and Hahn, U. (2010). The calbc silver standard corpus for biomedical named entities - a study in harmonizing the contributions from four independent named entity taggers. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Rehbein, J., Schmidt, T., Meyer, B., Watzke, F., and Herkenrath, A., (2004). *Handbuch für das computergestützte Transkribieren nach HIAT*.

Reyle, U. and Riester, A. (2016). Joint information structure and discourse structure analysis in an Underspecified DRT framework. In Julie Hunter, et al., editors, *Proceedings of the 20th Workshop on the Semantics and Pragmatics of Dialogue (JerSem)*, pages 15–24, New Brunswick, NJ, USA.

Riester, A. and Baumann, S. (2017). *The RefLex Scheme – Annotation Guidelines*, volume 14 of *SinSpeC. Working Papers of the SFB 732*. University of Stuttgart.

Riester, A., Brunetti, L., and De Kuthy, K. (to appear). Annotation guidelines for Questions under Discussion and information structure. In Evangelia Adamou, et al., editors, *Information Structure in Lesser-Described Languages: Studies in Syntax and Prosody*. Benjamins, Amsterdam.

Sagae, K. and Lavie, A. (2006). Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 129–132, New York City, USA, June. Association for Computational Linguistics.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Schmid, H. (2004). Efficient Parsing of Highly Ambiguous Context-Free Grammars with Bit Vectors. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, Geneva, Switzerland.

Schmid, H. (2006). Trace prediction and recovery with unlexicalized pcfgs and slash features. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, Sydney, Australia, July. Association for Computational Linguistics.

Schweitzer, A. and Lewandowski, N. (2013). Convergence of articulation rate in spontaneous speech. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association (Interspeech 2013, Lyon)*, pages 525–529.

Schweitzer, A. (2010). *Production and Perception of Prosodic Events – Evidence from Corpus-based Experiments*. Doctoral dissertation, Universität Stuttgart.

Seeker, W. (2016). Guidelines for the Annotation of Syntactic Structure in the IMS Interview Corpus.

Stehwien, S. and Vu, N. T. (2017). Prosodic event detection using convolutional neural networks with context information. In *Proceedings of Interspeech*, pages 2326–2330.

Thiele, G., Seeker, W., Gärtner, M., Björkelund, A., and

Kuhn, J. (2014). A graphical interface for automatic error mining in corpora. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 57–60, Gothenburg, Sweden, April. Association for Computational Linguistics.

van Santen, J. and Hirschberg, J. (1994). Segmental effects on timing and height of pitch contours. In *Proceedings of the 3rd International Conference on Spoken Language Processing (ICSLP 94)*, pages 719–722, Yokohama, Japan, 09.

Westpfahl, S., Schmidt, T., Jonietz, J., and Borlinghaus, A. (2017). Stts 2.0. guidelines für die annotation von pos -tags für transkripte gesprochener sprache in anlehnung an das stuttgart tübingen tagset (stts).

11. Language Resource References

Calhoun, S., Carletta, J., Brenier, J., Mayo, N., Jurafsky, D., Steedman, M., and Beaver, D. (2010). The nxt-format switchboard corpus: A rich resource for investigating the syntax, semantics, pragmatics and prosody of dialogue.

Eckart, Kerstin and Riester, Arndt and Schweitzer, Katrin. (n.d.). *DIRNDL – Discource Information Radio News Database for Linguistic analysis*.

Ostendorf, Mari and Price, Patti J. and Shattuck-Hufnagel, Stefanie. (1995). *The Boston University Radio News Corpus*. Electrical, Computer and Systems Engineering Department, Boston University.

Schweitzer, Antje and Lewandowski, Natalie. (2013). *German Conversations: The IMS GECO database*. IMS, University of Stuttgart.

Preparing Data from Psychotherapy for Natural Language Processing

Margot Mieskes, Andreas Stiegelmayr
Hochschule Darmstadt
Darmstadt, Germany
{firstname.lastname}@h-da.de

Abstract

Mental health and well-being are growing issues in western civilizations. But at the same time, psychotherapy and further education in psychotherapy is a highly demanding occupation, resulting in a severe gap in patient-centered care. The question which arises from recent developments in natural language processing (NLP) and speech recognition is, how these technologies could be employed to support the therapists in their work and allow for a better treatment of patients. Most research in NLP focuses on analysing the language of patients with various psychological conditions, but only few examples exist that analyse the therapists behavior and the interaction between therapist and patient. We present ongoing work in collecting, preparing and analysing data from psychotherapy sessions together with expert annotations on various qualitative dimensions of these sessions, such as feedback and cooperation. Our aim is to use this data in a classification task, which gives insight into what qualifies for good feedback or cooperation in therapy sessions and employ this information to support psychotherapists in improving the quality of the care they offer.

Keywords: psychotherapy, speech resources, German, data preparation, guidelines, data preprocessing

1. Introduction

Mental health care is a very demanding occupation, where quality assurance and improvement is very time- and resource intensive. Improving mental health care through psychotherapy involves two aspects: First, the point of view of the therapist, who is required to constantly attend further education measures, to ensure the quality of their work remains high and by improving it through supervision. Second, the patients' point of view, who has a right for high quality care, especially for conditions, where the treatment greatly benefits from psychotherapy. But due to a lack of therapists, these patients are still treated with drugs, which are less effective. As there is a lack in further education options for therapists, the number of qualified therapists is hardly increasing, resulting in a vicious circle. This is especially apparent in the treatment of patients with schizophrenic conditions. To improve the situation, psychological therapists are collecting data from therapy sessions, which are then judged by experts from their field. Their guidelines define 14 quality dimentions, such as *Feedback* or *Cooperation*, rated on a seven-point Likert scale. This work is currently carried out manually and is very time-consuming and resource extensive. Previous work on applying Natural Language Processing (NLP) shows that NLP can be used to extract certain aspects in human-human communication. In our work, we build on this previous work to support the analysis of the therapy session. Before this, the data has to be processed, which requires methods and tools, we partially develop ourselves. Our work describes efforts in creating a pipeline for preprocessing therapy recordings to use NLP and machine learning to support psychotherapists in analysing their work. This preprocessing itself poses several challenges: As patients are quite sensitive to their surroundings, the therapy sessions should not be disturbed by technical equipment. The solution found by psychologists is using mobile phones as recording devices, which results in a low recording quality. Second, the material is very sensitive and has to be processed accordingly. And finally, most NLP tasks work on very fine-grained parts of language and speech, wheras the judgement of the therapy sessions by experts is on a very high, session-wide level, rather than a single part. In order to be able to apply NLP to this type of data, we present our guidelines for the transcription, which take into account the high level judgements, as well as pointers from the literature on indicators for (for example) good feedback or cooperation. Additionally, we describe our pipeline to extract a range of features for the classification. Previous work in this area has been primarily carried out on English and there is very little information on how applicable the results are to other languages such as German, which we study here. Even though we cannot publish the original recordings due to data protection laws, we will release our guidelines and implementations for further usage.[1]

2. Related Work

As our project is heavily motivated by research results from psychology research, we present not only previous work on NLP in this context, but also results from research on treating psychological conditions, with a focus on the treatment of schizophrenic patients.

2.1. Research in Psychology

Patients with schizophrenia, who suffer from persecution complex and hearing voices are normally treated in therapeutic institutions, using antipsychotic drugs. Only about 1% of the patients are treated with psychotherapy on top (Görgen and Engler, 2005). But there are numerous studies which indicate that psychotherapy is beneficial for this group of patients (see for example (Lincoln, 2014)). At the same time other studies show that therapists lack possibilities for advanced training in this specific area (Lincoln et al., 2014; Mehl and Lincoln, 2014), which leads to a vicious circle, as there are too few therapists for this patient group.

[1] https://github.com/mieskes/Paranoia/

This circle is hard to break and this reduces the chances of patients receiving the best possible care.

2.2. Research in NLP

In recent years, NLP has been applied to the context of psychotherapy, specifically focusing on spoken language. There are efforts in finding acoustic correlates for specific emotions in the spoken language (see for example (Schuller and Batliner, 2013)). Additionally, work on finding correlates for stress in general (Paul et al., 2015) or cognitive stress (Hecht et al., 2015) has been carried out. In most cases both acoustic features, as well as linguistic features are used. These elements correlate to quality dimensions as annotated by psychotherapists. But so far, except for general work on emotion detection, work has been primarily carried out on English data. Work on German is rare.

Flekova et al. (2015) work in the educational domain and look at the topic of the quality in classroom interaction. This bears some similarity to determining the quality in therapy sessions and also the quality dimensions relevant for educational researchers and psychologists share common notions, such as *Feedback* and *Cooperation*. Their findings indicate that how participants phrase their utterances and what kind of words they use are indicative of the quality of the interaction.

Chakravarthula et al. (2015) look at the behavior of the therapist, rather than the patients, especially their empathy level in the context of addiction counseling. The authors model the behavior and find that they are able to reliably classify the empathic state based on the output of an automatic speech recognition system.

2.3. NLP and Schizophrenia

Recently, there has been some work on analyzing the language in the context of schizophrenia. Howes et al. (2013) look at the distribution of topics in therapy sessions of schizophrenic patients and relate them to the therapy outcome. They use Latent Dirichlet Allocation (LDA) to determine the topics with respect to hand-coded topics.

Mitchell et al. (2015) analyse data from social media based on information given by the users themselves. Therefore, the authors mention that the data and the results have to be treated cautiously. Nevertheless, they find that various features in the language differed from the language of people in a control group. The authors find differences in the character n-grams, the distribution of topics, measured using Latent Dirichlet Allocation (LDA) and also the usage of LIWC categories differ between the schizophrenic users and the control group.

Kayi et al. (2017) also use social media data in addition to essays written both by patients and control persons on two topics: *their average Sunday* and *what makes them angriest*. The authors look at a range of syntactic, semantic and pragmatic features in the writings of all persons. The authors report that among the syntactic features, Part-of-Speech tags are very predictive, but they also reveal some problems, as the tag for foreign words (FW) primarily marked misspelled words. Among the semantic features, clusters based on word vectors, topics based on LDA and semantic role labelling (SRL) achieve good results. Among the pragmatic features, sentiment intensitiy features perform better than sentiment features.

3. Background on the data collection

We are collaborating with a psychotherapeutic study, where currently data is being collected from real therapeutic sessions. The goal of that project is to prove the effectiveness of a specific therapy method. Therefore, patients are aware that they are being recorded and that the material is used for research. The patients, as well as the therapists explicitely gave their consent to being recorded and the data being used for research purposes. The sessions are then analyzed and classified by at least two trained psychotherapists using a rating manual which describes 14 quality criteria on a seven-point Likert scale (Lecomte et al., 2017). In our initial study, we focus on few of these criteria, which are also reflected in the language used, such as *Feedback*, *Positive Focus* and *Collaboration*, which we briefly describe in the following. As pointed out by Mitchell et al. (2015) with respect to their data, it is possible that the data of this study is also not representative. Both patients and therapists participate voluntarily. This might skew the distribution towards very motivated patients and therapists. Additionally, the recording might have a slight influence on their behaviour, which the data collectors tried to keep to a minimum.

We received the recordings of the sessons via encrypted, password-protected hard drives to reduce the risk of the data being compromised. The recording quality is very poor, as the recording device was a mobile phone, placed on the table, which is occasionally moved, papers are put on top, etc. which disturbs the recordings. Additionally, the data was stored in an .mp3-format, which reduces the quality even further.[2] We did not receive any meta data about the persons in the recordings, therefore, we did not know any personal details about them. Nevertheless, as patients give details about their situation, the data is too sensitive to be made publicly available.

3.1. Quality Dimensions

In our work we use a subset of the 14 quality dimensions. These seven dimensions are described in the following. More details on these and the remaining dimensions can be found in the original manual (Lecomte et al., 2017).

Positive Focus is based on the observation that patients often face prejudices, which focus on their deficiencies or handicaps. The therapist is required to focus on the positive aspects of the person, their strengths and goals, rather than their problems. Facts can be used in order to inform the patient about their symptoms. The seven-point Likert scale used for the rating, states that 0 means that the therapist only focuses on the problems and symptoms and completely neglects goals or strengths of the patient, whereas 6 means that the therapist is aware of the strengths and goals of the patient and supports him/her in finding new strengths and further develop existing ones.

[2] Improving the recording quality by using dedicated recording devices and head-mounted microphones would disturb the therapy session too much. (Personal communication with one of the lead researcher in the data collection study.)

Feedback goes back and forth between patient and therapist. The latter has to ensure, that the patient is involved in the therapy. But the patient should not feel evaluated or judged in a school-like way, but rather that feedback is supposed to be an exchange of points of view in order that both participants understand each other. A rating of 0 states that there is no feedback, neither does the participant ask for it, nor does the therapist ask for the patients approval. If feedback is requested, it is ignored by the therapist. A rating of 6 means that feedback is given regularly and effectively. If the patient reacts unexpectetly the therapist in turn reacts in a positive way and does not discard it as a misunderstanding. Feedback feels natural.

Cooperation is required to successfully treat a patient. It often occurs that patients withdraw and behave passively. The therapist is required to encourage the patient to cooperate and become more active. A rating of 0 indicates that the therapist does not involve the patient in any decision making process and does not motivate him/her to cooperate. A rating of 6 indicates that both parties participate in the session, if there are any problems the therapist reacts sensibly and both come to a decision.

Access to Emotions which means that people can regulate their emotions, which is very difficult for patients with schizophrenia. It is important that the therapist is able to name emotions that show themselves in the here and now. A rating of 0 indicates that the therapist does not even try to name emotions. A rating of 6 indicates that the therapist looks at emotions from different perspectives not only in the here and now, but also in the past.

Identification of Cognition helps the patient to develop a link between their thoughts and their emotions and behaviour in order to change it. The therapist should understand which thoughts are central. A rating of 0 indicates that the therapist does not try to discuss any thoughts or point them out. A rating of 6 indicates that the therapist deals with current lines of thought and supports the patient in identifying behvioural elements in specific situations.

Agenda checks whether there is one or not and whether a previously set agenda actually matches the patients needs. As patients often face difficulties remembering, the agenda should also track the past. A rating of 0 means that there is no agenda or it was not mentioned. A rating of 6 means that an agenda was mentioned and contains all relevant aspects. The patient and the therapists agreed on the set agenda.

Identification of Behaviour is important as the therapist should support the patient in seeing connections between thoughts and behaviour. The therapist should focus on behavioural aspects that hinder success. A rating of 0 indicates that the therapist does not even try to discuss the patients' behaviour. A rating of 6 means that the therapist frequently considers current behaviour and tries to support the patient in connecting emotions and thoughts with his/her current behaviour.

3.2. Manual Annotation

Table 1 shows the distribution of the quality criteria analysed. In total 35 sessions were analysed, although 4 are missing for *Identification Cognition* and *Access to Emotions*. The distribution is skewed towards the higher ratings. No session was rated with 0 for any of these categories and only few have been marked as 1 or 2. The annotators gave a rating of 4 and more for most of the sessions on most of the quality criteria. An exception to this is *Identification Cognition* where 14 sessions were marked with 3.

Quality	1	2	3	4	5	6
Agenda	1	1	0	10	15	8
Positive Focus	0	1	5	9	11	6
Feedback	0	1	4	7	14	9
Cooperation	0	0	2	8	12	13
Access to Emotions	0	3	7	8	8	5
Identification Cognition	0	2	14	7	6	3
Identification Behaviour	0	2	8	9	11	1

Table 1: Frequency of ratings.

We analyse the annotation using DKPro Agreement (Meyer et al., 2014). The agreement on the categories we consider varies greatly. While *Feedback* achieves a fairly high Fleiss' κ score ($\kappa = 0.497$), the categories *Identification of Cognition* and *Identification of Behaviour* achieve a κ of close to 0. One of the reasons is that not all sessions were annotated by at least 2 annotators. Looking in detail at the annotation, we observe that most differences occur in the range of one point (i.e. 3 vs. 4), which are probably hard to distinguish in a task that is fairly hard to begin with.

4. Processing

The first step in processing the data is the transcription. We experimented with off-the-shelf automatic speech recording tools, but the recording quality proved to be too low. Therefore, we have to first transcribe the data manually, for which a tool and guidelines are required.

4.1. Evaluation of Transcription Tools

We put together a list of features the tools had to have. This includes information on the operating system it ran, the quality of the documentation, the graphical user interface, but also import and export file formats. After a preselection process, we examine three tools according to our requirements (ELAN[3] (Wittenburg et al., 2006), OpenSmile[4] and Audacity[5]). We discover a lot of similarities between the tools. Major differences are in the area of supported file formats and the views offered by the tools. Some tools offer unique characteristics, such as the possibility to segment the data. As the objective analysis based on the features did not allow for a clear distinction, we did a preliminary study with a sample of the data and found, that ELAN offered the best usability and therefore, decided to use it.

4.2. Transcription Guidelines

Based on previous work in quality estimation and information in the quality annotation guidelines used in analysing

[3] https://tla.mpi.nl/tools/tla-tools/elan/

[4] http://audeering.com/technology/opensmile/

[5] http://www.audacity.de/

the therapy sessions, we develop transcription guidelines to ensure that the transcription captures phenomena which would be helpful for the classification of quality dimensions in psychotherapy. Additionally, we address specific phenomena, which we observe in our data, such as the usage of dialectal elements, which have to be treated accordingly.[6]

Dialect We translate dialectal elements to standard German, as most NLP tools can hardly deal with dialectal speech. During transcription, we still keep the verbatim expressions, but add a translation to standard German to normalize for varying dialects.

Non-verbal elements We transcribe non-verbal elements in the recordings. This includes elements from social noise, such as variants of *hm*, which can express acknowledgement, question, surprise etc. We also consider elements such as crying and laughter. To analyse the duration of the noise we are marking them via time stamps.

Pauses Pauses are marked also with respect to their length to differentiate between short and long pauses.

Disfluencies Hesitations and disfluencies indicate various emotional states (nervousness, insecurity, etc.). Therefore, we transcribe hesitations, repairs etc. with great detail, following earlier work by Heeman and Allen (1999)

Citations If the patient or therapists cites him-/herself or somebody else, this is transcribed and marked accordingly, to not confuse the words used by a third person with the person saying them.

Punctuation As spoken language not always uses sentences in a traditional sense, we refrain from using a full stop to indicate the end of a sentence. Rather, commas indicate clauses. We use full stops only for obvious sentence ends. Question marks indicate a question, also for rethorical questions. Exclamations marks indicate stressed words. A colon indicates elongated words.

Not understandable elements Due to the recording quality, we have to take into account that elements of the recordings might not be understandable. The annotators mark those elements accordingly. If they can understand something, they can put the most likely transcription there, but nevertheless, keep the non-understandable marking. These elements can then be re-checked and subsequently used cautiously.

4.3. Manual Preprocessing

As of today, we have transcribed over two hours of recordings, which are almost equally split between the therapists and the patients (see Table 2). We see that while therapists have more segments than patients, the patient segments are considerably longer, indicating that they contribute a lot to the conversation. Pauses are on average fairly short, considering the amount of over 400 pauses distributed over patients and therapists. Patients use slightly more pauses than therapists, which are shorter than therapists pauses, suggesting that therapists are careful about their wording. Dialectal elements are more frequently used by patients rather than therapists, indicating that they do not restrict themselves in the way they express themselves.

[6]The guidelines can also be found in our Github repository.

Observation	Amount
# Segments (Patient)	720
# Segments (Therapist	948
avrg. length of segments (Patient)	00:00:05.355
avrg. length of segments (Therapist)	00:00:03.866
# of disfluencies (Patient)	85
# of disfluencies (Therapist)	56
# of pauses (Patient)	270
# of pauses (Therapist)	146
avrg. length of pauses (Patient)	00:00:01.837
avrg. length of pauses (Therapist)	00:00:02.123
# dialectal elements (Patient)	911
# dialectal elements (Therapist)	535

Table 2: Statistical Information on the current data set.

We observe that the patients have considerably more segments that contain not understandable elements (approx. 30%), wheras the therapist only has about 18% not understandable elements. This supports our approach of focusing on the therapist for the analysis of the therapy quality.

5. Classification

In the following, we describe our pipeline, the features we extract and some preliminary classification results.[7] Our primary tools are DKPro Core[8] for the linguistic preprocessing and Weka[9] for the machine learning.

5.1. Feature Extraction

In order to do meaningful classifications using machine learning methods, we need features extracted from the data. These features are based on the literature presented in Section 2. above and on the guidelines by the psychotherapists (see Section 3.). Therefore, we focus on elements such as pauses, social noise, disfluencies, but also on the vocabulary used. Earlier work indicates, that for example words from specific word groups indicating insight, understanding, etc. point to a good cooperation and a well established feedback culture (Flekova et al., 2015). Among the features we extract, are surface features, which represent the content of each speakers' part in the conversation. This includes number of sentences, but also pauses and their lengths. As Kayi et al. (2017) shows that Part-of-Speech (POS) tags are very predictive to determine the patients, we use syntactic features such as POS, but also the number of questions and the usage of tense (future, past, present). We also look in detail at hesitations and stuttering.

5.2. Preliminary Results

In an initial experiment we distinguish therapist from patient. This is motivated by the final classification, which looks at the interaction between the two, but with a focus on the therapist to find correlates to therapy quality until the recording problems are solved. A first experiment using the full feature set resulted in an accuracy of 73.6.%
Table 4 presents results on our initial experiments using ten-fold cross-validation. Surface features, such as segments

[7]The implementation is also available from our github repository.

[8]https://dkpro.github.io/dkpro-core/

[9]http://www.cs.waikato.ac.nz/ml/weka/

Feature Set	Precision	Recall	F-Measure
Full	0.742	0.737	0.735
surface	0.732	0.725	0.723
disfluencies	0.492	0.493	0.474
syntactic	0.624	0.624	0.624
segment length only	0.695	0.678	0.671

Table 3: Results of a preliminary classification experiment distinguishing between patients and therapists.

Feature Set	Precision	Recall	F-Measure
Full	0.774	0.775	0.774
Top 10	0.783	0.782	0.782
Stutter	0.598	0.598	0.596
Speech Break	0.590	0.590	0.590
Noise	0.588	0.588	0.584
Social noise	0.618	0.614	0.615
LIWC Data	0.726	0.711	0.694
Dialect	0.566	0.567	0.566
Sentiment	0.581	0.583	0.582

Table 4: Results of Random Forest Algorithm with 10-fold cross validation.

and token lengths allow for a good distinction between patient and therapist – even slightly better than for the full feature set. Interestingly, the features based on disfluencies give the worst results, which are well below chance. Syntactic features give results better than chance, but not comparable to surface features (F-measure of 0.735). The segment length alone (measured in seconds) gives very good results, comparable to those of the complete feature set. As the full transcription is very time-consuming to carry out, this result gives us a good starting point to distinguish between patient and therapist and developing methods to judge the quality of the interaction between the two participants based on features from the audio only.

5.3. Quality Criteria Classification

As all sessions have to be rated on all quality criteria it makes sense – due to the limited data set size – to train models for the quality criteria independently. We use the same feature set we use for the distinction between patient and therapist also for the classification. Of the 35 sessions we have available to date, eleven have been segmented and partially transcribed. Table 1 indicates that the distribution of the classes is quite skewed and tends towards the higher marks. This supports our assumption that the data set might not be representative and contains very good therapists and motivated patients. Additionally, it indicates that the majority baseline is already quite hard to beat. As the classes 1-3 are hardly used, we collapsed them into one class, which leaves us with four classes to distinguish.

Table 5 shows the results for each quality criteria we looked into using ten-fold cross validation and a Random Forest learning algorithm. For most cases the classification is comparable or above the baseline. Only in the case of *Cognition* our results are below the baseline. These low results can be explained by the skewed data distribution, which often tends towards a rating of 5 or 6.

As the full transcription of the data is extremely time-consuming, we also experimented with the same machine-learning setup using only features available from segmenting the data[10]. Table 6 shows the results for these experiments. For most quality criteria the results drop, which is expected. Surprisingly, results for *Agenda* and *Behaviour* increased. While the results for both quality criteria did not exceed the baseline in the original setup, using the reduced feature set we achieve results better than the baseline.

Quality	Prec	Rec	F	Maj
Feedback	0.44	0.5	0.46	0.4
Cognition	0.31	0.42	0.34	0.5
Behaviour	0.25	0.17	0.2	0.35
Positive Focus	0.34	0.45	0.46	0.34
Agenda	0.39	0.5	0.43	0.43
Access Emotion	0.30	0.33	0.30	0.26
Cooperation	0.44	0.5	0.46	0.37

Table 5: Quality Criteria Classification Random Forest

Quality	Prec	Rec	F	Maj
Feedback	0.42	0.44	0.41	0.4
Cognition	0.25	0.21	0.23	0.5
Behaviour	0.5	0.46	0.48	0.35
Positive Focus	0.29	0.28	0.27	0.34
Agenda	0.46	0.52	0.47	0.43
Access Emotion	0.25	0.20	0.22	0.26
Cooperation	0.41	0.44	0.41	0.37

Table 6: Quality Criteria Classification Random Forest – segment features only

5.4. Most Important Features

In the following, we look at the individual quality criteria and also take a closer look at the best performing features. As we have some of the data fully transcribed, while some of the data was only segmented, we also looked at the best performing features in both feature sets.

Feedback The most important features to classify *Feedback* are social noise elements, but also the frequency of incomplete sentences. This indicates that both participants reflect to each other, that they are continiously listining and processing what the other is saying.

When we use only the segmentation based features, we observe that the duration of the segments is of imporantance, but also the relative frequency of segments by the therapist. As the therapist is using a lot of social noise, this increases the number of segments, although they are of course very short. Also very short breaks are of importance, indicating that there is little silence during the session.

Cognition The identification of *Cognition* is charactized by the importance of features such as the ratio of questions. It is interesting to note that also a high amount of stuttering and incomplete sentences is highly predictive features for this quality criterium. As is to be expected, words indicating sentiment or emotion are also very important.

[10]Please note that this is necessary in this setup, as we only received mono recordings of the sessions.

Based on the segmentation alone, we find that the breaks are very important in judging the quality of this criterium.

Behaviour The patients are required to also reflect on their behaviour. In our machine learning approach, we see that highly ranked features are those that are based on sentiment bearing words (both positive and negative), but not necessarily words associated with a specific emotion.
Using only segmentation features the amount of segments by the patient is very important. Additionally short breaks are mor important than longer breaks, contrary to classifying the quality criteria *Cognition*.

Positive Focus When determining the score for *Positive Focus* we see that the features ranked highest are those based on the LIWC dictionary. Interestingly, we see that a lot of negative emotions or sentiments are important and metaphoric expressions play a role.
When we use only segmentation based features, we see that short breaks are an important feature, but also the frequency of segments by the therapist.

Agenda Classifiying the quality criteria *Agenda* is also based heavily on words from the LIWC dictionary. Features based on words from the domain of space and time, but also sports are ranked highly. Also words from the domain of cause are important, which indicates that the participants reason about the why of their approach.
Using only segmentation-based features short breaks are important features for the classification.

Access Emotion For the classification of the category *Access to Emotion* we see that the duration of segments is of importance. Contrary to the description of this category, emotion or sentiment bearing words are less important in this machine learning based approach. Rather, surface features such as the number of characters per token and how many questions were asked are important.
The segmentation-based features also show a high interaction, as only short breaks play a role and the number of segments by the therapist are as important as the number of segments by the patient.

Cooperation The most important features for classifying the *Cooperation* category are also based on the LIWC dictionary. Among those are words from the category we, you and communication. This indicates a high amount of interaction and expressions that the two participants consider themselves as a team.
From the segmentation-based features short breaks and the duration of the segments are the most important features.

5.5. Discussion

Looking in detail at the results, we see that the full transcription of the data is necessary to achieve good results and that features based on the LIWC dictionary are especially important. These are only accessible through a thorough transcription. It is also interesting to note that for two quality criteria (*Agenda* and *Behaviour*) the smaller feature set based only on the segmentation features achieve considerably better results than using the full feature set. This is especially surprising for *Agenda*, where the top ranked features are actually from a matching domain in the LIWC dictionary. For *Behaviour* the features are less conclusive, with general sentiment features are ranked at the top.
It is also interesting to note that for both quality criteria that deal with the self-reflection of the patient among the high ranking features we see features that indicate difficulties such as hesitations, incomplete sentences and incomplete phrases. For most of the quality criteria the performance drops when only the limited feature set is used.

6. Conclusion and Future Work

We presented ongoing work in collecting and processing data from therapy sessions with patients with schizophrenic disorder in order to allow for semi-automatic processing based on natural language processing. Our aim is to classify the therapy quality, which is currently carried out manually with a high effort in time and expert man-power. Due to the recording quality a lot of effort as of now went into the transcription of the recordings. Based on the quality dimensions used by psychologists in the supervision of therapeutic sessions we identified specific dimensions which manifest themselves in natural language – either in *what* a person says or in *how* it is being said, and defined the extracted features accordingly. Our initial results indicate, that we can distinguish the therapist and the patient based on a range of features, most notably through information about the therapist segments only.
In the next step, we focused on the therapists and related their behaviour to the quality dimensions evaluated by experts, in order to build a classifier for these specific dimensions and evaluate them. Our results indicated that a high quality transcription is necessary to allow for a machine-learning based classification of the quality criteria used here. With the exception of two quality dimensions, which performed better with a reduced feature set, all quality dimensions were more reliably classified using the full feature set. Especially features based on the LIWC dictionary proved very valuable, which is in line with previous work.
In the future, we plan to extend the current feature set to also include acoustic features, which give an additional dimension of how people express themselves, beyond *what* they say, but rather *how* they say it.

7. Acknowledgements

This work was supported by the program *Forschung für die Praxis* by the Hessian Ministry for Sciences and Art, Germany and by the research center *Digital Communication and Media Innovation* (DKMI), University of Applied Sciences, Darmstadt. We specially thank our project partner at the Frankfurt University of Applied Sciences, under the supervision of Prof. Mehl who performed the manual quality annotation of the data and our two student helpers who were segmenting and transcribing the data.

8. Bibliographical References

Chakravarthula, S. N., Xiao, B., Imel, Z. E., Atkins, D. C., and Georgiou, P. G. (2015). Assessing empathy using static and dynamic behavior models based on therapist's language in addiction counseling. In *INTERSPEECH 2015, 16th Annual Conference of the Inter-*

national Speech Communication Association, Dresden, Germany, September 6-10, 2015, pages 668–672.

Flekova, L., Sousa, T., Mieskes, M., and Gurevych, I. (2015). Document-level school lesson quality classification based on german transcripts. *Journal for Language Technology and Computational Linguistics*, 30(1):99–124.

Görgen, W. and Engler, U. (2005). *Ambulante psychotherapeutische Versorgung von psychosekranken Menschen sowie älteren Menschen in Berlin*. Psychotherapeutenverlag – Verlagsgruppe Hüthig Jehle Rehm GbmH.

Hecht, R. M., Bar-Hillel, A., Tiomkin, S., Levi, H., Tsimhoni, O., and Tishby, N. (2015). Cognitive workload and vocabulary sparseness: theory and practice. In *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, pages 3394–3398.

Heeman, P. A. and Allen, J. F. (1999). Speech repairs, intonational phrases, and discourse markers: Modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, 25(4).

Howes, C., Purver, M., and McCabe, R. (2013). Using Conversation Topics for Predicting Therapy Outcomes in Schizophrenia. *Biomedical Informatics Insights*, 6(1):39–50.

Kayi, E. S., Diab, M., Pauselli, L., Compton, M., and Coppersmith, G. (2017). Predictive linguistic features of schizophrenia. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 241–250, Vancouver, Canada, August 3–4, 2017. Association for Computational Linguistics.

Lecomte, T., Kingdon, D., and Munro-Clark, D. (2017). Cognitive therapy scale for psychosis – revised version. in preparation.

Lincoln, T. M., Rief, W., Westermann, S., Ziegler, M., Kesting, M.-L., Heibach, E., and Mehl, S. (2014). Who stays, who benefits? Predicting dropout and change in cognitive behaviour therapy for psychosis. *Psychiatry Research*, 216(2):198 – 205.

Lincoln, T. (2014). *Kognitive Verhaltenstherapie bei Schizophrenie*. Hogrefe, 2 edition.

Mehl, S. and Lincoln, T. (2014). *Therapie-Tools Psychosen*. Beltz Verlag, Weinheim Basel.

Meyer, C. M., Mieskes, M., Stab, C., and Gurevych, I. (2014). DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. In Lamia Tounsi et al., editors, *Proceedings of the 25th International Conference on Computational Linguistics: System Demonstrations (COLING)*, pages 105–109, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics. https://www.ukp.tu-darmstadt.de/software/dkpro-statistics/.

Mitchell, M., Hollingshead, K., and Coppersmith, G. (2015). Quantifying the language of schizophrenia in social media. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 11–20, Denver, Colorado. Association for Computational Linguistics.

Paul, W., Alm, C. O., Bailey, R. J., Geigel, J., and Wang, L. (2015). Stressed out: what speech tells us about stress. In *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, pages 3710–3714.

Schuller, B. and Batliner, A. (2013). *Computational Paralinguistics: Emotion, Affect and Personality in Speech and Language Processing*. John Wiley & Sons, Inc.

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). ELAN: a Professional Framework for Multimodality Research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA).

MirasVoice: A bilingual (English-Farsi) speech corpus

Amir Vaheb[1], Ali Janalizadeh Choobbasti[1], Behnam Sabeti[1], S.H.E. Mortazavi Najafabadi[1], Saeid Safavi[2]

[1]Miras Technologies International
NO. 3, 2nd Alley, North Sheikh Bahai St., Tehran, Iran
{Amir, Ali, Behnam, Hani}@miras-tech.com,
[2] School of Engineering and Technology, University of Hertfordshire
College Lane Campus, Hatfield AL10 8PE, Hertfordshire, UK
s.safavi@herts.ac.uk

Abstract

Speech and speaker recognition is one of the most important research and development areas and has received quite a lot of attention in recent years. The desire to produce a natural form of communication between humans and machines can be considered the motivating factor behind such developments. Speech has the potential to influence numerous fields of research and development.
In this paper, MirasVoice which is a bilingual (English-Farsi) speech corpus is presented. Over 50 native Iranian speakers who were able to speak in both the Farsi and English languages have volunteered to help create this bilingual corpus. The volunteers read text documents and then had to answer questions spontaneously in both English and Farsi. The text-independent GMM-UBM speaker verification engine was designed in this study for validating and exploring the performance of this corpus. This multilingual speech corpus could be used in a variety of language dependent and independent applications. For example, it can be used to investigate the effects of different languages (Farsi and English) on the performance of speaker verification systems. The authors of this paper have also investigated speaker verification systems performances when using different train/test architectures.
Keywords: Speech Processing, Automatic Speaker Verification, Speech Corpus

1. Introduction

Automatic speaker verification (ASV) systems architecture offers a flexible and low-cost solution for biometric authentication. Although the research and development of ASV systems in recent years have improved their performance to the point of mass-market deployment; due to advances in noise and channel compensation techniques, these systems can be concerned vulnerable to spoofing (Wu et al., 2015). In recent years a number of counter spoofing techniques have been proposed. The reason behind this attention is the development of robust ASV systems for biometric authentication which has many applications in the security sector. This problem has been widely studied for English speakers, but not for Iranians or users that know multiple languages. There are several audio corpora for dominant languages like English. For instance, the Santa Barbara corpus of spoken American English (Bois et al., 2000 2005) which consists of 249,000 words spoken with the transcriptions. Another example is the Callhome American English Speech corpus developed by the Linguistic Data Consortium (LDC) (Canavan et al., 1997) which is made up of 120 unscripted 30-minute long on the phone conversations that were made in North America. There are also some purely Farsi speech corpora, but none of them are bilingual. For example, (Bijankhan et al., 1994) speech corpus containing recordings of 300 native Farsi speakers from 10 different dialect regions in Iran.
In this study, we present MirasVoice which is a bilingual audio corpus in Farsi and English. MirasVoice contains high quality recorded content from 50 speakers (27 male, 23 female), 40 of which are currently labeled. There is both read and spontaneous audio in both Farsi and English. The speakers read 3 text documents in both English and Farsi and then had to answer 17 questions in both languages. The participants were all native Iranian speakers who were educated in English. MirasVoice contains more than 33 hours of audio and can be used for a variety of audio and signal processing applications like audio speaker recognition, gender recognition and in general pattern recognition problems.
In this study, the effects of different languages (Farsi and English) on speaker verification systems was investigated. The attained results show that the best performance is obtained when the language used for both training and test phases are the same.
In the remainder of this paper, Section 2 provides more details about MirasVoice. Validation of the corpus and the experimental results are presented in Section 3. Section 4 concludes the paper.

2. Corpus Description

The MirasVoice Speech Corpus (MVSC) is one of the largest Farsi-English voice datasets currently available for general purpose studies and expert-system development. Some of the applications this dataset can be used for is for speaker recognition systems, speech recognition studies, gender recognition, cognitive science, and pattern recognition. This dataset is expected to grow larger. It currently consists of 50 individuals speaking 2 languages on 4 different texts. The convent of the read text is explained in the next section.

2.1. Speech Materials

The MVSC consists of both read speech and spontaneous speech materials. The text material read by the volunteers

Context of Text Material	Material Amount
Word	250
Sentence	63
Number	80
Question	17

Table 1: Corpus reading material information.

is available alongside the dataset. The text includes a number of words, sentences, and numbers in English. This text has been translates into Farsi by educated native speakers. This translated text has also been read by all participants. As shown in table 1, There are 250 words, 63 sentences and 80 numbers in the text. The numbers start off as easy and progress on to more complicated numbers. We also had 17 questions which we asked each participants which they answered spontaneously. We also gathered information on whether the participants smoked or not, their blood pressure, age, height, accent, birth country, mothers birthplace (province), fathers birthplace (province), time of recording and which province they grew up in. The voice files are stored in 1 minute long .wav files. there is also one 7 minute long .wav audio file in which the participant is answering the questions.

2.2. Labeling

Labeling of the dataset has been done by the authors themselves. The audio files labeled in a special manner. The generic form of the labels is LLXXXTNN. the LL stands for the Language in which the speaker is speaking and is either EN (English) or FA (Farsi/Persian). The second part XXX stands for the index the person has in the overall information .csv file. The third part represents the text the participant is reading in this file which can have 4 different character values W, S, N, Q. The characters respectively stand for words, sentences, numbers, and questions. The last part of the name, NN stands for the file number for that particular text (e.g. 03 means the third audio file). For example, the name EN002S03 means the third English audio file recorded from the second participant that was reading the sentences text. Participants name have not been shared for privacy reasons.

2.3. Recording Procedure

The recordings for the MVSC have mostly been carried out in the conference room at Miras Technologies International central office using a large microphone and stored directly onto a PC in Wave file format using a high quality microphone with a sample rate of 48kHz, a bit rate of 16 bits, a frequency response of 20Hz to 20kHz and a max SPL of 120db. Before each recording, the participants would first read out a text while the recording settings were being adjusted. Since the authors of the dataset didn't want the file lengths becoming too long, they would pause the recording when an audio file reached the threshold of 1 minute and would ask the participants to continue on a separate recording.

For the questions, the authors gave the participants a sheet containing the questions and would ask them to read the questions and answer them respectively in one recording. The length of the questions audio file is 7 minutes long. An initial plan was to collect approximately 40 minutes of recording per each speaker.

2.4. Filing System

In the dataset repository, the audio file containing male and female participants have been separated into different directories. Each directory contains a number of sub-directories indexed by the participants' index located in the .csv file at the root directory of the repository. The starting index for male participants is 001 and ends at 020, and for females, it starts from 021 and goes up to 040.

2.5. Tools

The microphone used for the recordings was a model Yeti microphone from Blue. The microphones pattern setting was set to Cardioid mode for all speakers. Cardioid mode records audio sources that are directly in front of the microphone, delivering rich, full-bodied audio. The authors also measured the speakers' blood pressure using the Beurer wrist blood pressure monitor model BC 40. We recorded the speakers systolic and diastolic blood pressure.

3. Corpus Application

MVSC is a bilingual speech corpus and could be used in speaker verification problems. This study investigates the effect of different languages on the performance of speaker verification systems. A schematic of the process of speaker verification is shown in figure 2.

3.1. Speaker Verification System

Speaker Verification is a branch of bio-metric authentication and means acceptance or refusal of speaker's claim as one of the known users of the system. During the above process, one's speech information will be compared with a speech model analogous with the claimed identification; the results would be acceptance or refusal of the speaker's claim. If the match is above a predefined threshold, the identity is accepted, otherwise, it is rejected. Both speaker identification and speaker verification tasks could be divided further into text-dependent and text-independent categories, based on a number of constraints on the contents of the test and train utterances. While in text-dependent, speakers speak the same text on both training and test phases, in the text-independent phase voice samples should be different (Mporas et al., 2016) (Singh et al., 2012).
This study uses Text-Independent Speaker Verification System (TISV) to show one of the applications of MVSC.

3.1.1. Signal Analysis

Feature extraction was performed as follows. Periods of silence were discarded using an energy-based Speech Activity Detector (SAD). The speech was then segmented into 20-ms frames (10-ms overlap) and a Hamming window was applied. The short-time magnitude spectrum, obtained by applying an FFT, is passed to a bank of 30 Mel-spaced triangular band-pass filters, spanning the frequency region from 0Hz to 44000Hz.

3.1.2. GMM-UBM System

Automatic speaker verification engine used in this study is based on the Gaussian Mixture Model - Universal Background Model (GMM-UBM) method (Reynolds et al., 2000). In this approach, the feature vectors of the parameterization section could be presented as a weighted sum of multiple Gaussian distributions, which is called the GMM. Each single Gaussian distribution has its own mean, weight, and covariance. We used all the recorded conversations wave files for building the background model. At the end class dependent speaker models are built by map adapting the means of UBM with respect to the class dependent enrollment data.

3.1.3. Verification Experiments

Verification experiments were conducted using a similar version of the methodology developed for the NIST speaker recognition evaluations. Each test utterance was scored against the "true" (correct) speaker model and 10 other "impostor" models. Results are presented in terms of percentage Equal Error Rate (EER), calculated using the standard NIST software (Safavi et al., 2012). In speaker verification systems, EER is one of the measure to evaluate the system performance (Wang and Cheng, 2004).

Type of speech	Recorded Audio
Total amount of labeled data (minute)	1588
Amount of data used for training models (minute)	1115
Amount of data used for testing models (minute)	473

Table 2: Database information.

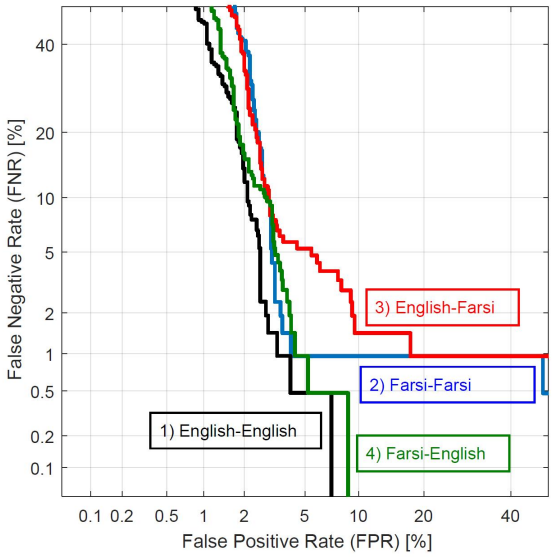


Figure 1: Comparison of 4 Different Setups

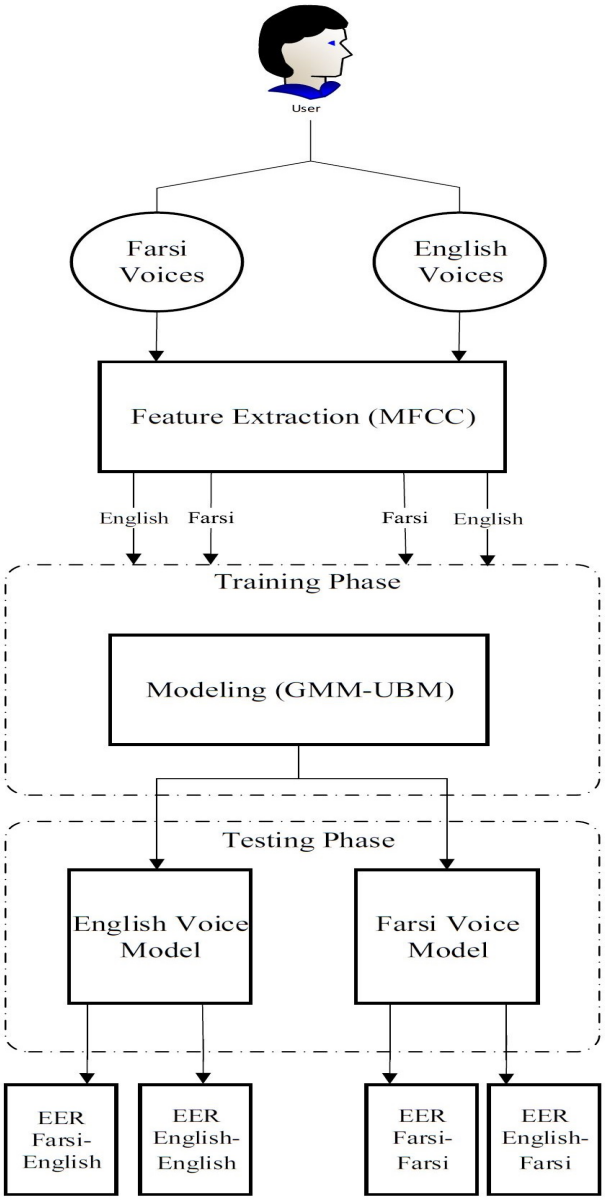


Figure 2: Block diagram of the proposed experiment system which uses text-independent speaker verification system

3.2. Experimental Results

In this study, four experiments were conducted. In the training phase, voice samples from volunteers speaking Farsi and English have been used. Speaker dependent models were then modeled for both English and Farsi. Same as training, in the testing phase we use voice samples both in English and Farsi languages. 70% of the data was used for training and the remaining 30% was used for testing. As shown in figure 1, the Evaluation was then done using 4 different setups:

1. Performance of the system when trained using English data and tested with English data as well, using the English Background Model.

2. Performance of the system when trained using Farsi data and tested with Farsi data as well, using the Farsi Background Model.

3. System performance when trained using English data and tested with Farsi data, by the means of the English Background Model.

4. Performance of the system when trained using Farsi data and tested with English data, by the means of the Farsi Background Model.
   At the end of this section we are going to compare the attained results from each setups.

Index	Training-Testing Phases	EER(%)
1	English-English	2.5837
2	Farsi-Farsi	3.2381
3	English-Farsi	5.7585
4	Farsi-English	3.5885

Table 3: The Equal Error Rate (EER) for different experiments

As shown in table 3, English-English (training-testing phases) experiment has the best performance, which demonstrates that the system is more accurate in English which might be because the English language has better overall structure and a more tangible pattern for the model. The results show that having different languages in the training and testing phases increase EER. This means that the system performance is highly language-dependent.
It is worth mentioning that it's best for the non-English language community to use their native language over English in the training phase in order to use the TISV model.

4. Conclusion

In this study, MVSC which is a bilingual speech corpus in English and Farsi is presented. All the volunteers that are native Farsi speakers educated in English have been given 3 texts to read and a sheet of questions to answer in both languages. The speaker would read the texts and answer the questions in a quiet conference room with a high-quality microphone. Features like age, height, gender, etc. have also been added in the corpus repository. Currently, over 33 hours of high-quality audio from 50 speakers are available in the corpus repository but the goal is to record 50 more speakers in order to have approximately 66 hours of audio from 100 speakers in the near future.
In order to Validate MVSC, this study presents and compares the result of experiments in TISV for two different languages. The effect of using different languages on the performance of speaker verification systems is also investigated. Based on the results, it is shown that the TISV model can perform best if the data used in the training phase is the speaker's native language than English. Results revealed that the best performance is obtained when English is used for both training and testing. However, in the cases which the users are only capable of speaking their mother-tongue, it is best to use that language for both training and testing.

5. Acknowledgements

The authors of this paper would like to thank everyone at Miras Technologies International and all the volunteers especially the members of team Meissa and team Draco who put a lot of effort into building the presented dataset.

6. References

Bijankhan, M., Sheikhzadegan, J., and Roohani, M. R. (1994). Farsdat- the speech database of farsi spoken language. *Proceedings Australian Conference On Speech Science And Technology*, 2:826–830.

Bois, J. W. D., Wallace, L. C., Meyer, C., Thompson, S. A., Englebretson, R., and Martey, N. (2000 - 2005). Santa barbara corpus of spoken american english, parts 1-4. *Philadelphia: Linguistic Data Consortium.*

Canavan, A., Graff, D., and Zipperlen, G. (1997). Callhome american english speech ldc97s42. *Philadelphia: Linguistic Data Consortium*.

Mporas, I., Safavi, S., Gan, H. C., and Sotudeh, R. (2016). Evaluation of classification algorithms for text dependent and text independent speaker identification. *IEICE*.

Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. (2000). Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10:19–41.

Safavi, S., Najafian, M., Hanani, A., Russell, M., Jancovic, P., and Michael, J. C. (2012). Speaker recognition for children's speech. *INTERSPEECH*, 2(1):1836–1839.

Singh, N., Khan, R. A., and Shree, R. (2012). Applications of speaker recognition. *Elsevier Ltd.*, 54:0975–8887.

Wang, H. C. and Cheng, J. M. (2004). A method of estimating the equal error rate for automatic speaker verification. *International Symposium on Chinese Spoken Language Processing (ISCSLP)*.

Wu, Z., Evans, N., Kinnunen, T., Yamagishi, J., Alegre, F., and Lia, H. (2015). Spoofing and countermeasures for speaker verification: a survey. *Elsevier*, 66:130–153.

Dialog Intent Structure: A Hierarchical Schema of Linked Dialog Acts

Silvia Pareti, Tatiana Lando
Google Research Europe
{spareti,tlando}@google.com

Abstract

In this paper, we present a new hierarchical and extensible schema for dialog representation. The schema captures the pragmatic intents of the conversation independently from any semantic representation. This schema was developed to support computational applications, be applicable to different types of dialogs and domains and enable large-scale non-expert annotation. The schema models dialog as a structure of linked units of intent, *dialog acts*, that are annotated on minimal spans of text, *functional segments*. Furthermore, we categorise dialog acts based on whether they express a primary or secondary intent and whether the intent is explicit or implicit. We successfully tested the schema on an heterogeneous corpus of human-human dialogs comprising both spoken and chat interactions.

Keywords: Dialog Acts, Speech Acts, Intents, Dialog

1. Introduction

With the increasing popularity of conversational systems and chatbots, we are faced with the challenge of creating suitable dialog representations and annotated data to support the development of dialog modeling systems. Relying on form-filling strategies, with predefined dialog states sequences, falls short of providing a flexible and adaptable system that supports natural conversation, where the interaction is not constrained by a fixed plot with one precise goal. On the other hand, treating utterances or messages independently from each other fails to recognise the interconnected dialog structure and phenomena that are crucial to correctly understand and generate dialog interactions. Part of a larger dynamic and interactive language exchange, dialog messages have both semantic content and communicative functions. The latter provide the conversational context needed to correctly interpret the semantic content and determine what the current dialog state is.

The challenge of representing the communicative intent of dialog segments with dialog acts (DAs) has been addressed multiple times over the past decades. While earlier efforts, e.g. Map Task (Carletta et al., 1997), have focused on representing specific tasks and domains, the attention has shifted to more general purpose schemas such as the Dialogue Act Markup using Several Layers (DAMSL) (Allen, 1997) and the ISO 24617-2(Bunt et al., 2010; Bunt et al., 2012), a standard for dialog act annotation. The latter offers a powerful representation comprising 9 core dimensions and around 60 communicative functions. Communicative functions are linked to functional segments (FSs), defined by (Bunt et al., 2012) as "the unit of dialogue act annotation".

These schemas were not specifically developed to represent human-computer conversation. By developing a schema for the specific application, we can ensure that we identify useful categories that can help us understand the human dialog input as well as generate a suitable machine reaction while also reducing the tagset complexity. The latter is crucial to conduct large-scale non-expert annotation. In particular, such representation should enable non-constrained interactions were the dialog structure is not predefined around a specific scenario with a single task to solve, but transitions between dialog states are learned from data.

For this purpose, we developed a new hierarchical and extensible schema for dialog structure representation. This defines a set of dialog acts that express conversational intents and are tagged as being explicit or implicit and constituting the primary or a secondary intent of a FS. We defined such representation by developing the schema on a corpus of user-assistant dialogs that were collected between human participants. The schema was successfully applied to different dialogs (in terms of written or spoken modality as well as what the dialog is trying to achieve and around which topic), demonstrating that it provides a sufficiently abstract and robust representation of dialog interaction. This schema can be used to annotate resources and support the development of dialog understanding and generation systems that can successfully model the complexity of natural language.

2. Dialog Act Schema

The schema we introduce represents dialog, intended as a conversation between two or more participants, as a graph of interconnected intents. We developed the schema to support large-scale non-expert annotation and piloted it on different types of dialogs to ensure it is robust and meets an acceptable rate of inter-annotator agreement. The goals of the schema are to:

- Represent dialog structure as a sequence of dialog acts expressing a communicative function. This kind of representation supports understanding the intents expressed by each turn in the conversation and generating appropriate follow-up turns.
- Support any kind of dialog relevant to the development of chatbots and dialog agents, in spite of the medium of communication, the number, type and role of the participants or the domain or topic discussed.
- Be independent from semantic or domain specific representations.
- Support extensible granularity through a set of coarse and fine grained tags hierarchically organised. Tags

can be further extended to support finer-grained distinctions, such as capturing the exact type of request to repeat (e.g. louder, rephrase, same).

This representation was developed to support modeling an artificial dialog agent by providing tags that are informative of the conversational expectations at each given point in the dialog and help to determine what the agent response should be. For example, while the ISO 24617-2 (Bunt et al., 2012) would tag 'What?' as AutoNegative thus capturing that the speaker is signaling non-understanding, we instead focus on what reaction the speaker expects to trigger in the listener by tagging this as a request to repeat. This way we can capture that 'What?' and 'Can you repeat?' have the same conversational function and should trigger the same reaction.

We did limit the number of categories by avoiding infrequent and less-useful distinctions. For example, while in the ISO standard, response acts are categorised by the communicative function of the act they are responding to, we considered this as unnecessary since in our representation response FSs are linked to the FS that triggered the response. For example, instead of having 'DeclineRequest, DeclineSuggest, DeclineOffer' we just have one category for 'reject' that can be linked to an instruct, suggest or offer act.

2.1. Turns and Dialog acts

We use the **turn** as a basic unit of dialog, intended as an uninterrupted sequences of messages or speech events from a single participant. Depending on the transcription convention, backchannels can be transcribed as a separate interrupting turn. Turns may comprise multiple sentences or even paragraphs. We avoid using the utterance as a minimal unit of dialog since there is little consensus on how this should be defined and because it only applies to spoken dialogs. Dialog Acts are annotated on **functional segments (FSs)** (Bunt et al., 2010; Bunt et al., 2012): minimal spans of text that express one or more dialog intents. Since in our data FSs were very rarely conveyed more than 2 intents, we decided to constrain the annotation to assign no more than 2 DAs per FS. FS division do not correspond to sentence or clause division, one functional segment can cover several sentences. In Ex.1 the answer consists of several sentences which all correspond to the same intent - provide information in response to a question.

(1) A: Today I want to learn about social psychology.

B: In psychology, social psychology is the scientific study of how people's thoughts, feelings, and behaviors are influenced by the actual, imagined, or implied presence of others. In this definition, scientific refers to the empirical method of investigation...

All text in a dialog should be part of one and only one FS, i.e. we do not allow for text to be left unannotated nor for FSs to overlap or be nested. Nested or overlapping tags can provide additional challenges when processing the data automatically, both for building machine-readable representations and for developing tools to support the annotation. In the ISO standard (Bunt et al., 2012), FSs can theoretically be discontinuous or run across turn boundaries. To simplify the automatic processing and encoding of the annotation, we restricted FSs to be continuous spans of text within a turn. To account for cases where a conversational intent was expressed over a non-continuous span of text, we introduced a specific technical label 'goes_with' to connect the detached parts of the DA. The same label can be used to connect two parts of an utterance interrupted by another participant, which can frequently happen in spoken data (Ex.2).

(2) A: Assistant: For soup, they have Horiatiki, Marouli, Ascolibri...

B: User: Assistant!

C: Assistant:...for the Mesquite Grill they have House Specialty - Arni Paithakia

D: Assistant: Yes?

Spans of text that do not express any identifiable function, such as abandoned or unintelligible spans, are annotated with the 'no-intent' tag.

2.2. Structure of the schema

We devised a hierarchy of categories grouped based on the component that is needed to process and react to a given state in the conversation. Tags are grouped into 3 categories focusing on the assistant reaction: interactional, social, other. These are motivated by the different processing that is needed: interactional acts require that we understand and process the information conveyed; social acts are often formulaic and do not contribute towards the goal of the conversation; other acts include DAs that do not express any intent or any recognised intent or that we need to merge with another FS before processing it.

The tagset inventory (Table 1) is organised in a hierarchy defined on 3 levels of granularity, comprising: 6 higher level tags, 14 middle level tags and 34 lower level tags. The naming of the tags follows a hierarchical approach and inherits the full path through the schema, such as: request.query.open. The act name on the right of the dot is a sub-act of the name on the left. The fine-grained level of DAs can be further splitted into sub-acts to introduce finer distinctions.

The 6 major classes group DAs according to the intent of the speaker and the goal they are trying to achieve:

- **Request**: The set of acts that are intended to elicit some reaction from the listener. Information request usually take the form of a question (directly or indirectly formulated). Action requests are formulated as instructions to accomplish a task or require some action on either or both the speaker and listener's sides.

- **Respond**: Respond acts are complementary to request acts, which they usually follow. Respond acts are answers to information requests and also reactions to an action request.

- **Assert**: Assert acts cover the transmission of information that is not requested in previous discourse, unlike respond acts, nor expect the listener to provide any sort of reaction. These are usually expressions of opinion or statements setting the ground for further interaction.
- **Social**: Social acts have purely social intent and are usually expressed in natural conversation to conform to social expectations. These include greetings, politeness expressions and expressions of sympathy and agreement. These expressions are often formulaic and they can be omitted from the conversation without affecting its structure or comprehension.
- **Other**: A set of acts used to handle cases that do not fit in the 4 previous categories. This includes FSs where the intention of the speaker is unclear or not collaborative, like self talk or abandonment.
- **Add_on**: A set of technical labels used to handle corrections or the continuation of an interrupted segment. These labels should be used to pre-process the data by joining these FSs with those they attach to.

2.3. Dialog Act primary and implicit tagging

When more than one intent is expressed by one single FS, we categorize them as **primary** and **secondary**. Ex.3(B2) conveys two intents: agreeing to the proposal to get some tea and expressing gratitude. These are anchored to the same span of text and cannot be separated as we do in Ex.3(B1) where the pipe denotes the boundary between two distinct FSs. The primary intent must correspond to the most salient intent at that point in the conversation, i.e. the intent that we need to identify to understand the dialog state and provide a conversationally appropriate reaction, in Ex.3 - answering to a proposal. Every functional segment is annotated with one primary dialog act and might also have one secondary act.

(3) A: Would you like some tea?

B(1): No,| thanks!

B(2): Thank you so much.

Both primary and secondary intents can be explicitly or implicitly expressed. In Ex.3(B2) the primary intent (agreeing to a proposal) was implied by explicitly using a polite expression. We introduce a specific label, **implicit**, to mark similar cases where the DA is not expressed by the span of text corresponding to the functional segment, but rather conveyed through entailment or implicature. If a FS has only one intent it is always considered primary and explicit. While other existing schemas allow for a FS to have multiple DAs associated, they do not explicitly mark these as being the most salient intent of the FS nor whether the intent was implicitly expressed. We take the primary intent to connect the DAs into a graph structure and we identify implicit intents since they can require different reaction strategies. Ex.4 and Ex.5 both express a request, however, in the first example this is implicitly expressed through a conversational implicature. While the final goal might be in both cases to find a restaurant, the assistant answers are not interchangeable. In Ex.4, since the information was implicit requested, the assistant can offer to provide it while also making sure that that is indeed the intent. In Ex.5 instead, an offer would not be an appropriate answer and the assistant can directly provide the information.

(4) A: I'm hungry.

B: Would you like help finding nearby restaurants?

(5) A: Find me nearby restaurants

B: Sure. There is an Italian ...

2.4. Dialog Act linking

In addition to assigning DA labels to FSs, we also connect them in a graph. The structure of the dialog is nonlinear, that is, a DA is not necessarily connected to the immediately preceding one, but might relate to any preceding DA. This can be due to the medium of communication causing delays such as a chat message being written while the conversation continues, but also to the way we structure and connect thoughts. One example is when one participant asks a number of questions in a row and another answers them one by one, as in Ex.6:

(6) A(1): who are the actors in Pete's Dragon?

A(2): how are the reviews for this movie?

A(3): Is it directed by anyone famous?

B(1): Of the main characters, Bryce Dallas Howard plays Grace, Robert Redford plays Meacham, Oakes Fegley plays Pete...

B(2): Rotten Tomatos gives it a Tomatometer of 87%, and an Audience Score of 82%

Keeping track on how DAs connect is crucial to computationally model dialog by representing conversational expectations as transition probabilities between states. We therefore represent dialog structure as a graph where each FS (except the very first one in the conversation) attaches to another one already existing in the dialog. We restrict linking to the primary intent expressed by the FS and do not link any secondary intent. In case an intent is not a follow-up of another intent (e.g. a topic starter), it is simply attached to the immediately preceding FS. In this case, we consider the FSs to be connected by a relation of proximity in the dialog. In applying the schema, we follow these principles:

- Each FS has one single primary dialog act associated and can have one additional secondary DA.
- Any DA can be marked as implicit.
- Each primary DA in the dialog is linked with only one preceding DA and with one or more following DAs.

An example of the full annotation is shown in Table 2. The dialog is a shortened version of an actual dialog form the annotated corpus we describe in the next section.

request	instruct	task, cancel, backtrack
	query	open, select, yn
	propose	suggest, offer
	check	align, confirmation, repeat
assert	provide	elaboration, statement, opinion
respond	yes	agree, accept
	no	disagree, reject
	reply	open, select
	notify	acknowledge, buy_time, success, failure
social	greetings	opening, closing
	politeness	apology, thanks, acknowledge_thanks
	interpersonal	feedback
other	other	other_intent, no_intent
add_on	add_on	goes_with, correct

Table 1: Dialog acts tags inventory.

Speaker	Text	Index and link	DA annotation
User	I want Sushi	1.1	Instruct.Task (primary, implicit) Provide.Statement (secondary, explicit)
Assistant	Hanabi sushi near Mountain view serves sushi	2.1	Propose.Suggest (primary, implicit) Provide.Statement (secondary, explicit)
User	When does the place close?	3.1	Query.Open (primary, explicit)
User	I want to have a long dinner.	3.2	Provide.Statement (primary, explicit)
Assistant	It closes at 11 pm.	4.1 linked to 3.1	Reply.Open (primary, explicit)

Table 2: Fully annotated excerpt of a dialog from the corpus

3. Corpus

The scheme was applied to a corpus of scenario-based English dialogs collected between two human participants. The corpus comprises both spoken and chat data. The data was collected in different locations, with different participants and different settings. The scenarios provided the participants with a role and a goal. Participants were asked to act either as an assistant (human or virtual) or a person asking for assistance (user), or as people talking to each other. Goals included locating information (either to make a selection or exploring a topic), performing a task or chitchat and casual conversation. After familiarizing themselves with the instructions participants improvised the dialog.

Chat dialogs were collected among volunteers through Google Hangouts. Spoken dialogs were recorded between hired participants either sitting in the same room separated by a divider to prevent them from seeing each other (to avoid non-verbal interactions) or talking over the phone. Spoken dialogs were then transcribed into text, including some non-verbal phenomena, while annotators had no access to the audio.

The full corpus was annotated with the scheme presented in Sec.2. and comprises 65 English dialogs: 28 spoken and 37 chats. Spoken dialogs are on average longer than chats resulting in the following turn distribution: 617 turns in spoken dialogs and 513 in chats. The corpus provided an heterogeneous testbed (see Table 3) to evaluate the schema applicability on multiple domains and different types of dialog).

Dimensions	
Modality	chat (28) , spoken (37)
Goal	information exploration (18)/information seeking (6) task-based (27), chitchat (14)
Topic	restaurants (13), movies (13), travel (8) sports (6), music (4), other (1-2 per topic)
Participants	person-person (14), person-assistant (51)

Table 3: Corpus composition

3.1. Pilot Corpora

The schema was piloted three times, each time on a new set of data. All the data used in earlier pilots was reannotated according to the final version of the tagset and contributed to the corpus described above. In the next section we will describe in more details how the pilots were organised, show the results and discuss the lessons learnt. The final (third) pilot was conducted on 20 English dialogs (10 spoken and 10 chat) and on an additional dataset of 2 spoken and 8 chat Italian scenario-based dialogs. The different datasets used in the pilots are summarized in Table 4.

4. Inter-annotator Agreement

In order to ensure that the schema is applicable to different types of dialogs with a satisfactory rate of inter-annotator agreement, we conducted three rounds of pilot annotation.

Datasets:	Dialogs	Turns	FSs
Full Corpus	65	1130	2116
- spoken	28	617	1253
- chats	37	513	863
2nd English pilot	24	492	953
3rd English pilot	20	458	558
- spoken	10	234	314
- chats	10	224	244
Italian pilot	10	305	364

Table 4: Corpora summary table

All pilots were conducted with professional linguists, however, minimal training was provided. Each pilot enabled us to identify critical points and further tune the schema and guidelines.

We run the first pilot in an early stage of the tagset development. This allowed us to test the initial design and intuitions on real data and to define a hierarchical and more intuitive structure. We also used the data to group never or rarely-occurring tags and to discover tag distinctions that were not satisfactorily covered by the schema. We then run a second pilot to put the tagset and guidelines to the test. This highlighted some remaining issues. Finally, we run a third pilot with the latest tagset and revised instructions. To provide more substantial training, we asked the annotators to revise the annotation on the data from the previous 2 pilots according to the latest guidelines and to collect and discuss any point of disagreement. In addition to annotating English dialogs, we started to put the applicability of the schema to other languages to test by running the third pilot also on the Italian dataset. While we did not encounter any substantial differences and we achieved comparable agreement results, we foresee that language and cultural characteristics have an impact on the dialog strategies used which could affect the dialog structure, tag distribution and the frequency of intents being conveyed implicitly. We plan to conduct pilot annotations in several languages from different families in the future.

In this section we describe the results from the third pilot and compare them to the second pilot.

4.1. Methodology

The annotation was performed in three steps:

- `functional segments splitting`: each turn in the dialog was split into one or more functional segments. This was performed by 2 annotators and the annotation was then revised and reconciled.
- `dialog act annotation`: 3 annotators assigned one or more dialog act labels to each functional segment. In addition, they added labels for *primary intent* (one per functional segment) and *implicit*.
- `dialog act linking`: 2 annotators identified non-linear links in the dialog and manually added the index of the preceding FS. Annotations were then revised and reconciled.

For the Dialog Act annotation, we introduced fallback tags *other* on the lower level of the hierarchy (e.g. request.instruct.other). The guidelines created for the pilot instructed annotators to use the fallback tags when they could not find a suitable fine-grained tag. This approach ensured that any tag missing from the original tagset would be identified. The initial tagset for the pilot did not contain the tag notify.buy_time. The pilot data from the second pilot showed significant use of the tag notify.other (3.46%) to annotate stalling intents (e.g. *Give me a second*). As a result this was added to the final tagset and the use of a fallback tag decreased in the third pilot to 0.98%.

4.2. Pilot Results

4.2.1. Splitting into Functional Segments

As the first step, annotators were presented with the dialogs consisting of turns and their task was to identify the boundaries of the functional segments. This task was performed two-way. We then reconciled and revised the annotations to create a gold reference annotation. Agreement was calculated as precision and recall with respect to the gold annotation. The set of the boundaries needed was considered the gold annotation and we calculated precision and recall for identifying these boundaries. This metric does not account for all possible boundaries in the corpus: theoretically each turn could be split at any space between words. Thus the metric is quite pessimistic but gives more insight into annotators performance. The average F-score for the annotators was 92.2% for English data and 87.6% for Italian, however it appears that some of the annotators did not fully understand the task and performed significantly worse than others as can be seen in Table 5. More substantial training and examples in the guidelines can increase annotators' performance in the future. The annotation was reconciled before proceeding to the next step: Dialog Act annotation.

English	Annotator 1	Annotator 2	Average
Recall	91.3%	93.2%	92.3%
Precision	98.8%	86.3%	92.5%
Italian	**Annotator 1**	**Annotator 2**	**Average**
Recall	100%	100%	100%
Precision	63%	95.7%	79.3%

Table 5: Annotators' performance on FS splitting task

4.2.2. Dialog Act annotation

At this step annotators were presented with dialogs presegmented into functional segments. Their task was to assign one or two Dialog Acts to the FSs and in case they assigned two to identify which Dialog act is the primary intent of the FS and whether any of the intents is implicit. If they assigned only one DA it was considered Primary and Explicit by default. The task was performed three-way. The Kappa inter-annotator agreement (IAA) on Primary Dialog Acts is summarized in Table 6. For English we achieved .71 Kappa and for Italian .64, which are both reasonably good results given the complexity of the task.

The IAA agreement was very low on secondary DAs since annotators could (and often did) choose not to assign any

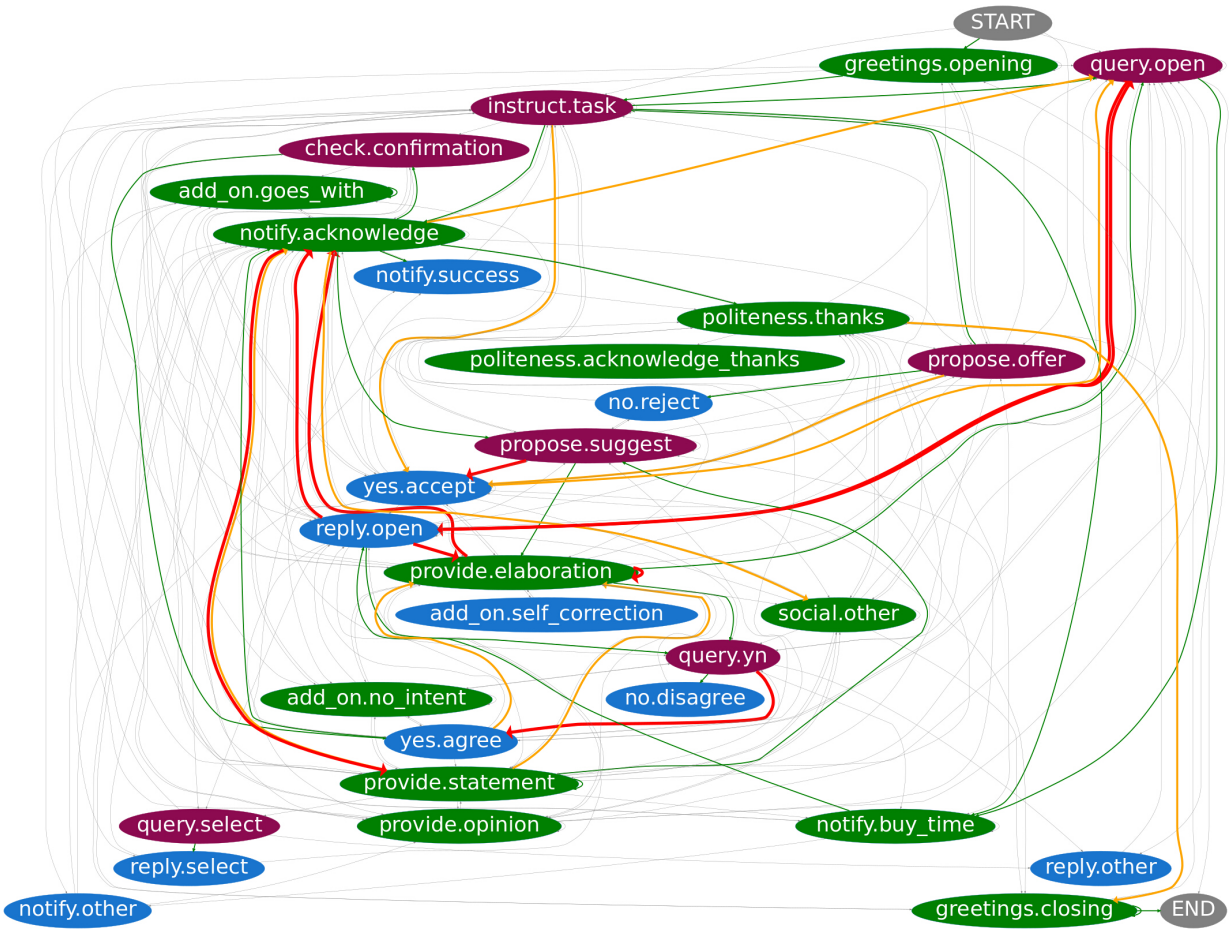


Figure 1: Dialog flow. Social DAs are in blue, request DAs in purple, respond DAs in blue and assert DAs in grey. Links are colour-coded based on how frequently that transition occurred in the corpus: > 5: grey; > 20: green; > 30: orange; > 50: red.

	Fine grained tags	Coarse grained
3rd English pilot	.71	.78
2nd English pilot	.58	.63
Italian pilot	.64	.72

Table 6: Kappa inter-annotator agreement for selecting the primary DA tag of a FS. Fine-grained tags comprised 34 DA labels, while coarse-grained tags 14.

secondary dialog act. Around 23% of the FSs were tagged by at least one of the annotators as having a secondary intent. In the reconciled pilot corpus only 11% of the FSs have two DAs associated.

The IAA was higher on chat dialogs. This could be explained by the lack of context for spoken dialogs and possible transcription errors and simplifications. The significant agreement increase from the second to the third pilot can be explained by several factors:

- The tag hierarchy was reorganised: we introduced a new tag to the inventory and removed rarely used ones. Moreover, there was no consensus on where to draw the boundaries between a request for information (i.e. the request to perform the task of answering a question as in 'tell me the weather forecast for tomorrow') and an instruct act or a question. We therefore collapsed these tags into only two distinctions based on whether we expect the interlocutor to provide some information or perform a task (that does not only result in providing information);
- Several tags were renamed making them more transparent for the annotators;
- We introduced a more substantial training stage and updated the guidelines based on annotators' feedback.

4.2.3. Dialog Act Linking

For this step, annotators were presented with the data pre-segmented into FSs and annotated with DAs and their labels. Their task was to identify cases when the FS expressed an intent that was not a response or reaction to the intent expressed in the immediately preceding FS and identify the correct antecedent. The average F-score for the annotators was 57.3% for English and 83.2% for Italian, detailed metrics are summarized in the Table 7.

The extremely low results from the English Annotator 2 can be explained by a misundertanding of the guidelines. The annotator systematically missed the links in those cases where there was a Notify or Social DA between Request

English	Annotator 1	Annotator 2	Average
Recall	95.2%	19%	57.1%
Precision	78.4%	57.1%	67.8%
Italian	Annotator 1	Annotator 2	Average
Recall	88%	80%	84%
Precision	78.6%	87%	82.8%

Table 7: Annotators' performance on FS linking task

and Response, i.e. in Ex.(7) FS (C) should be linked to FS (A) as the reply is following the answer and not the stalling expression.

(7) A: User: How about a different song from the same album, please?

B: Assistant: Hold on just a second

C: Assistant: here's the song All Over Me by Blake Shelton from the Same album.

4.2.4. Further steps

Although the IAA was very satisfactory for the primary DA annotation, we have identified some measures to further improve the consistency of the annotation:

- Annotators had no access to the audio of spoken dialogs. This can provide additional context (i.e. intonation) and reduce the ambiguity for DA-tagging and FS splitting.
- Annotators had to choose between 34 tags. While expert annotators could successfully perform this task, for large-scale non-expert annotation we plan to split the task into separate sub-tasks. We will ask annotators to make decisions on coarse-grained tags first and therefore simplify the task by reducing the number of tags to select from.
- Limited training was provided for FS splitting and linking which led to some annotators misunderstanding the task. This can be easily corrected with more training.
- Secondary intents are more subtle and harder to identify, for example any indirect request of the form 'Can you ...?' is implicitly deriving from a yes/no question, however this is such a common way to formulate requests that we no longer recognise this as a question. To achieve acceptable agreement also on this subtask, we have revised the guidelines to present more examples and have introduced rules to help annotators spot FS that have more than one intent.

5. Corpus Analysis

All the data from previous pilots was reannoted to match the final version of the tagset. The resulting corpus consists of 65 English dialogs and provides some insights into the intent structure of user-assistant free conversation. We generated the graph in Fig.1 from the corpus data to identify how dialog is structured and how DAs are connected.

The distribution of the DAs in the corpus shows that human interaction, even in a user-assistant scenario, is not limited to information exchanges and task accomplishment. The frequency of all Request and Respond DAs in the corpus sums up to 65.4%. Around 29.9% of DAs comprise Assert, Social or Other acts. The rest 4.7% of FSs were annotated as Add_on. These results highlight that by constraining human-machine dialog interaction to be a sequence of requests and responses we are failing to capture all the conversational phenomena that make a dialog natural.

Although turns are a readily available unit in dialog data, they cannot replace FSs as the dialog unit of intention. In the human-human dialogs we collected, participants produce around 1.87 FSs per turn with some difference between spoken and written dialogs (2.03 vs 1.68 respectively).

7.6% of the FSs in the corpus have an implicit intent associated, with 2.4% having a primary implicit intent.

The tag distribution in the corpus is rather balanced, with the 10 most frequently occurring DAs, shown in Table 8, covering around 60% of the data. For the secondary DAs, the most frequent ones are: Provide.statement (28% frequency over secondary tags) which is often associated to an implicit primary act expressing a request; Query.yn (12% frequency) which is usually a question in yes/no form that expects however more information to be supplied; Check.confirmation and Yes.accept (both 9% frequency).

Dialog Act	Occurrences	Frequency
notify.acknowledge	211	9.97%
provide.elaboration	192	9.07%
reply.open	191	9.03%
query.open	177	8.36%
provide.statement	120	5.67%
instruct.task	113	5.34%
yes.accept	106	5.01%
propose.suggest	83	3.92%
politeness.thanks	80	3.78%
query.yn	79	3.73%

Table 8: Top 10 most frequently occurring primary DAs.

6. Conclusion

The dialog schema presented in this paper was created to support computational conversation modelling and represents dialog as a graph of intents providing a structure that can be used by NLG and NLU systems. The schema and annotation tasks were refined through several pilot iterations, leading to the final representation presented in this paper. The latest pilot was conducted on a corpus of 20 English and 10 Italian dialogs of different types. For the identification of the primary intent of a functional segment, we achieved Kappa agreement among three annotators of .71 for English and .64 for Italian. The tasks of splitting turns into functional segments and linking dialog acts achieved reasonable agreement. We plan to adapt the task and pilot the scheme with non-expert annotators and on several different languages.

7. Acknowledgements

We would like to thank Prof. Marilyn Walker from UC Santa Cruz and our colleagues Dilek Hakkani-Tur, Anna Khasin and Soha Sultan for the constructive discussions that laid the foundations of this schema.

8. Bibliographical References

Allen, J. (1997). Draft of DAMSL : Dialog Act Markup in Several Layers.

Bunt, H., Alexandersson, J., Carletta, J., Choe, J.-W., Chengyu Fang, A., Hasida, K., Lee, K., Petukhova, V., Popescu-Belis, A., Romary, L., Soria, C., and Traum, D. (2010). Towards an ISO Standard for Dialogue Act Annotation. In *Seventh conference on International Language Resources and Evaluation (LREC'10)*, La Valette, Malta, May.

Bunt, H., Alex, J., woong Choe, J., Fang, A. C., Hasida, K., Petukhova, V., Popescu-belis, A., and Traum, D. (2012). ISO 24617-2: A Semantically-based Standard for Dialogue Annotation. In *In Proceedings Eighth International Conference on Language Resources and Evaluation (LREC 2012*.

Carletta, J., Isard, S., Doherty-Sneddon, G., Isard, A., Kowtko, J. C., and Anderson, A. H. (1997). The Reliability of a Dialogue Structure Coding Scheme. *Computational linguistics*, 23(1):13–31.

JDCFC: A Japanese Dialogue Corpus with Feature Changes

Tetsuaki Nakamura[†], **Daisuke Kawahara**[†‡]
†Kyoto University, ‡JST PRESTO
Yoshida-Honmachi, Sakyo-ku, Kyoto, Japan
tnakamura@nlp.ist.i.kyoto-u.ac.jp, dk@i.kyoto-u.ac.jp

Abstract

In recent years, the importance of dialogue understanding systems has been increasing. However, it is difficult for computers to deeply understand our daily conversations because we frequently use emotional expressions in conversations. This is partially because there are no large-scale corpora focusing on the detailed relationships between emotions and utterances. In this paper, we propose a dialogue corpus constructed based on our knowledge base, called the Japanese Feature Change Knowledge Base (JFCKB). In JFCKB and the proposed corpus, the feature changes (mainly emotions) of arguments in event sentences (or utterances) and those of the event sentence recognizers (or utterance recognizers) are associated with the event sentences (or utterances). The feature change information of arguments in utterances and those of the utterance recognizers, replies to the utterances, and the reasonableness of the replies were gathered through crowdsourcing tasks. We conducted an experiment to investigate whether a machine learning method can recognize the reasonableness of a given conversation. Experimental result suggested the usefulness of our proposed corpus.

Keywords: emotion, commonsense knowledge, dialogue corpus, crowdsourcing, neural network, LSTM

1. Introduction

In recent years, the importance of dialogue understanding systems has been increasing because interactive interfaces handling a natural language such as smart speakers have become popular. However, it is difficult for computer programs to understand our daily conversations because we frequently use emotional expressions in conversations. This is partially because there are no large-scale corpora focusing on the detailed relationships between emotions and utterances.

Many dialogue corpora have been developed because they are essential language resources needed to train and evaluate machine learning methods. For instance, the Dialog State Tracking Challenge (DSTC) dataset is used to estimate a user's goal in a spoken dialog system (Kim et al., 2016). While the DSTC corpus is made from manually transcribed Skype dialogues, there are corpora that consist of conversations extracted from SNS websites (Ritter et al., 2010; Ritter et al., 2011; Sordoni et al., 2015; Shang et al., 2015). There is also a corpus based on a collection of logs extracted from Ubuntu-related chat rooms that is mainly composed of technical support conversations (Lowe et al., 2015; Lowe et al., 2016). The Dialogue Breakdown Detection Challenge database is a corpus used to detect incorrect replies generated by dialogue systems (Higashinaka et al., 2016). Although these corpora are very useful resources for understanding actual human-human dialogue or human-machine dialogue, it is difficult to understand a speaker's/replier's motivations because such corpora do not record a speaker's/replier's inner state (in particular, his/her emotions). Even if such corpora include some keywords as clues for inferring a speaker's/replier's inner state, it is necessary to develop a method to extract inner state information from the corpora, which are composed of raw text.

Dialogue corpora that include various feature changes of arguments in utterances and the reactions to speakers can be used to understand a speaker's motivations. In the dialogue corpus used in Hasegawa et al. (2013), each utterance is annotated with the addressee's emotions. Although this corpus is useful for understanding the relationships between utterances and emotions in a conversation, the understandable relationships are limited to the addressee's direct emotional expressions because the corpus is annotated based on an explicit keyword list. In the keyword list, explicit keywords such as "afraid" and "happy" are manually associated with emotions "fear" and "joy" respectively. There are other relationships between utterances and emotions in conversations, such as relationships that concern the speaker's emotion, addressee's emotion, and emotions of any arguments in the utterances. We think that these relationships are also important for understanding speakers' motivations in conversations (especially in emotional conversations) in addition to the relationships used in Hasegawa et al. (2013). It is necessary to construct corpora designed to treat both of explicit and implicit emotional expressions because explicit emotional expressions are not always used in daily conversations. For example, when someone says "my wife hit my child," he probably wants to convey some kinds of information about his "surprise," "anger," and "disgust."

In this paper, we propose a dialogue corpus constructed based on our knowledge base, called the Japanese Feature Change Knowledge Base (*JFCKB*) (Nakamura and Kawahara, 2018). In the proposed corpus, feature changes (mainly emotions) of arguments in utterances and those of the utterance recognizers (i.e., utterers and addressees) are associated with the utterances. Because of the lack of large-scale corpora focusing on detailed relationships between emotions and utterances, the dialogue corpora constructed based on JFCKB will be useful for developing robots and software that can handle natural language. To validate the usefulness of our dialogue corpus, we conducted an experiment to investigate whether a machine learning method can recognize the reasonableness of

Sentence	Case (word)	Probability	Trigger utterance	Reply
Tsuma ga kodomo wo hippataku. (My wife hits my child.)	*ga* (nominative) (wife)	*joy* (+, −, UNC) = (0, 1, 0)	*Tsuma ga kodomo wo hippataita yo.* (My wife hit my child.)	*Hidoi ne.* (How terrible.) (○, ×, UNK) = (1, 0, 0)
		:		
	wo (accusative) (child)	*anger* (+, −, UNC) = (1, 0, 0)		*Nande*? (Why?) (○, ×, UNK) = (1, 0, 0)
		:		
	ni (dative) (NULL)	N/A		*Bouryoku ha ikenai yo.* (Violence is bad.) (○, ×, UNK) = (1, 0, 0)
	reader (NULL)	*disgust* (+, −, UNC) = (0.99, 0, 0.01)		
		:		:

Table 1: Example of the proposed dialogue corpus (JDCFC). The actual corpus is in Japanese. Each sentence has various feature changes for *readers* and three cases (*ga*, *wo*, and *ni*), which are Japanese language syntactic cases that roughly correspond to the nominative, accusative, and dative, respectively. *Readers* are not arguments in the sentence. The left three columns (sentence, case, and probabilities) in JDCFC are the same information of JFCKB. The trigger utterance corresponds to the event sentence. Replies are given probabilities for their reasonableness. In the "Probability" column, symbols +, −, and UNC denote *increased*, *decreased*, and *unchanged*, respectively. In the "Reply" column, symbols ○, ×, and UNK denote *reasonable*, *unreasonable*, and *unknown*, respectively.

a given conversation (i.e., a dialogue). This corpus is for Japanese.

2. Proposed Dialogue Corpus Based on a Feature Change Knowledge Base

Since the publication of our previous work (Nakamura and Kawahara, 2016), we have been constructing a knowledge base of argument feature changes in event sentences with controlled granularity. We call this knowledge base JFCKB (Nakamura and Kawahara, 2018). In JFCKB, arguments in event sentences are associated with various feature changes caused by the events. The feature changes of sentence readers (i.e., sentence recognizers) are also associated with the sentences in the current version of JFCKB. For example, in the case of "my wife hits my child," "my child" is associated with some feature changes, such as *increase in pain*, *increase in anger*, *increase in disgust*, *decrease in joy*, and *decrease in trust*. The sentence is also associated with feature changes such as *increase in a reader's anger* and *increase in a reader's disgust*. We gathered such information through crowdsourcing.

In this paper, we propose a dialogue corpus constructed using JFCKB to address the lack of large-scale corpora that focus on detailed relationships between emotions and utterances. We first briefly explain JFCKB, then we explain the proposed dialogue corpus.

2.1. JFCKB

JFCKB is composed of three types of information for event sentences, as shown the left three columns (sentence, case, and probabilities) in Table 1. As shown in the table, for each sentence, arguments in the sentence are associated with various features. Each feature in each argument has a triple (*increased*, *decreased*, and *unchanged*) whose values are probabilities.

We controlled the granularity of knowledge (i.e., features), and designed the 47 features shown in Table 2. These features were designed to correspond with *basic level categories* in cognitive linguistics (Rosch et al., 1976; Taylor, 1995) as much as possible. This design was based on a traditional emotion study (Plutchik, 1980), Japanese thesauri (Ikehara, 1997; NINJAL, 2004), sentiment analysis studies (Tokuhisa et al., 2008; Tokuhisa et al., 2009), and features used in the VerbCorner project (Hartshorne et al., 2014). Although our final version of JFCKB will have all the features listed in Table 2, emotional and sensory features are mainly investigated in the current study.

Event sentences for JFCKB were created as follows. Step 1: the 200 most frequent verbs, 1,000 most frequent verbs, and all verbs were respectively extracted from the *Kyoto University Web Document Leads Corpus* (*KWDLC*) (Hangyo et al., 2012)[1], *Kyoto University Case Frames* (*KUCF*) (Kawahara and Kurohashi, 2006)[2], and the Japanese version of the Winograd Schema Challenge dataset (*JWSC*) (Levesque, 2011; Shibata et al., 2015). KWDLC is a Japanese text corpus that comprises 5,000 documents (15,000 sentences) with annotations of morphology, named entities, dependencies, predicate-argument structures including zero anaphora and coreferences. KUCF is a database of case frames automatically con-

[1] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KWDLC
[2] http://www.gsk.or.jp/en/catalog/gsk2008-b/

Category	Sub category	Feature
physical	form	**length**, **size**, width, thickness (around) thickness (depth)
	color	redness, orangeness yellowness, greenness blueness, purpleness brownness, whiteness blackness, brightness
	touch	**temperature**, rigidness roughness, stickiness
	smell	goodness, badness
	sound	silence
	taste	sweetness, sourness bitterness, astringency hotness (not temperature)
	density	denseness
	amount	quantity
mental	emotion	**joy**, **trust**, **surprise** **disgust**, **fear**, **sadness** **anger**, **anticipation**
	evaluation	polarity
sensory	sensory	**pain**, **sleepiness** **tiredness**
relation	relation	interaction, possession physical contact physical force existence social relationship
	position	closeness

Table 2: Features assumed in this study. These features were decided by considering various studies such as traditional psychological studies, studies on cognitive development of infants, Japanese thesauri, sentiment analysis studies, and VerbCorner project. Features with bold fonts have been investigated so far.

structed from a corpus of 10 billion Japanese sentences taken from the Web. Case frames describe what kinds of nouns are related to each verb. Many Japanese verbs have some meanings. Examples are shown in Table 3. In KUCF, each case frame is composed of case frame ID, verb, cases, nouns filled in the cases, frequencies of the nouns in the Web corpus. KUCF has about 110,000 predicates with 5.4 case frames on average for each predicate. WSC (JWSC) dataset is basically composed of two sentences including one anaphor, two antecedent candidates, and a correct antecedent. Step 2: For each case frame of the extracted verbs in KUCF, representative event sentences were created. These sentences were composed of representative words for three cases in Japanese grammar (*ga*, *wo*, and *ni*: these cases roughly correspond to nominative, accusative, and dative, respectively). Each representative word was one of the most frequent words for each case in KUCF. Each case frame has one or some representative sentences because each case in the case frame has one or a few representative words. Step 3: we conducted a crowdsourcing task to discard nonsense sentences from the created sentences. In total, 1,559 crowdsourcing workers participated in this task and were asked to answer whether the presented sentences were comprehensible or not.

After event sentence creation, we conducted a crowdsourcing task to gather feature changes of arguments in the event sentences. In this task, in addition to feature changes of the arguments in sentences, we attempted to gather those of sentence readers. Crowdsourcing workers were asked to answer the feature changes of the arguments presented in the sentences (e.g., *anger* of "my child" in "my wife hits my child") or those of the workers themselves (e.g., *anger* of each worker himself/herself when he/she reads the presented sentence "my wife hits my child"). In total, 33,683 workers participated in this task. As a result, feature changes of 9,073 event sentences (types) were acquired (including 975 verbs (types) and 19,052 arguments (tokens) (4,882 types), 5,647 case frames (types)).

For every crowdsourcing task described above, we calculated probabilities that each answer would be selected by crowdsourcing workers based on the aggregation method proposed by Whitehill et al. (2009). Unlike majority voting, this method calculates the probabilities based on worker agreements.

2.2. JDCFC: JFCKB Based Dialogue Corpus

The proposed dialogue corpus, *JDCFC*, is based on JFCKB. JDCFC is composed of records for event sentences, as shown in Table 1. Each record is composed of feature change information for three Japanese syntactic cases (*ga*, *wo*, and *ni*: roughly corresponding to nominative, accusative, and dative, respectively) and sentence readers (sentence recognizers), the trigger utterance corresponding to the event sentence, and reply candidates to the trigger utterance with their probabilities of reasonableness. In common with JFCKB, the features in Table 2 are used in JDCFC.

Trigger utterances in JDCFC were created based on event sentences in JFCKB. 2,428 sentences were extracted from JFCKB as trigger utterances because these sentences have one or more feature changes caused by the events in total. For each utterance, we regarded a feature whose probability of *increased* or *decreased* is 0.75 or more as a feature changed by the event. The difference between the trigger utterances and event sentences in JFCKB is the expression of the predicates. In the trigger utterances, predicates are in past tense and attached with the postposition *yo*[3], while those in JFCKB are in present tense. This arrangement of predicates in trigger sentences is based on our speculation that (in Japanese, at least) such sentences are more natural as utterances than those in present tense in conversations. After trigger utterance creation, JDCFC was constructed using two crowdsourcing tasks.

The first task was to acquire replies to trigger utterances. In this task, crowdsourcing workers were given a trigger sentence and asked to answer appropriate replies, as shown in Figure 1. A total of 8,370 workers participated in the

[3] *Yo* is a postposition in the Japanese grammar, which represents familiarity. For example, many Japanese people have more friendly feeling towards "*Ii tenki da yo* (It is nice weather)" than "*Ii tenki da*."

Verb: case frame ID	Case	Word
yaku: yaku1 (bake)	*ga* (nominative)	*watashi* (I): 114, *haha* (mom): 75, *musume* (daughter): 74, ...
	wo (accusative)	*pan* (bread): 54076, *ke-ki* (cake): 31693, *niku* (meat): 14059, ...
	de (tools/ingredients)	*koubo* (yeast): 888, *be-kari-* (bakery): 768, *o-bun* (oven): 515, ⋯
yaku: yaku2 (have difficulty)	*ga* (nominative)	*mina* (all persons): 23, *sensei* (teacher): 11, *hito* (person): 8, ...
	wo (accusative)	*te* (hand): 26449
	ni (dative)	*kodomo* (child): 168, *musuko* (son):108, ...
yaku: yaku3 (burn)	*ga* (nominative)	*daitouryou* (president): 1, śhidousya (mentor): 1, ...
	ni (dative)	CD:13812, DVD:12200, ...

Table 3: Case frame examples. Each row denotes one case frame. In the "Word" column, each number denotes the frequency of the noun in the Web corpus.

Please answer your reply to the speaker.	
speaker's utterance	**My wife hit my child.**
your reply	

Figure 1: Question used for a crowdsourcing task to gather dialogue replies. Although this example is written in English, it was written in Japanese in the actual task.

Is the pair below a reasonable conversation? (speaker B's utterance is the reply to speaker A) (select *yes*, *no*, or *neither yes nor no*)	
speaker A	**My wife hit my child.**
speaker B	**Why?**

Figure 2: Question used for a crowdsourcing task to determine the reasonableness of replies to trigger utterances. Although this example is written in English, the actual task was written in Japanese.

task. After the task, for each trigger utterance, overlapping replies and extremely low-quality replies (such as copied and pasted trigger utterances and empty replies) were discarded. As a result, 23,196 replies (2,428 types of trigger utterances) were acquired. Hence, the average number of replies for each trigger utterance is approximately 9.6.

The second task was to determine the reasonableness of replies acquired in the first task. Crowdsourcing workers different from those in the first task were given a dialogue (a trigger utterance and one of its acquired replies) and asked to judge whether the given pair is reasonable (Figure 2). In total, 5,605 workers participated in this task, and ten workers were assigned to each dialogue. As a result, judgements on the reasonableness of the 23,196 dialogues were acquired. The probabilities of the reasonableness were calculated using the aggregation method proposed by Whitehill et al. (2009), just as used in JFCKB construction. The example shown in Table 1 presents typical acquired data. We regarded dialogues whose probability of the reasonableness was more than or equal to 0.8 as reasonable dialogues. The number of such dialogues was 22,357. We used the 22,357 dialogues in the evaluation experiment described in the next section.

2.3. Natural Dialogues versus Semi-artificial Dialogues

Our dialogue corpus is constructed semi-artificially because trigger utterances are collected artificially and replies are made through crowdsourcing tasks. It is desirable to construct a corpus through gathering naturally occurring dialogues from somewhere and annotating them with feature change information. However, to do this ideal construction procedure, there are problems described below at least: (1) it is difficult to get large scale dialogues, (2) there is no guarantee that utterances gathered from natural dialogues always have simple structures. Especially, in Japanese conversations, sentences often have no predicates or no essential arguments such as subjects and objects because ellipses are frequently used.

We are planning to start with applying feature change information to understanding of simple texts. Therefore, we made artificial trigger utterances based on event sentences in JFCKB that are based on highly frequent words in various Web documents.

3. Evaluation

In a dialogue, it is often necessary to use emotional knowledge to make a response to an utterance. For example, it is difficult to make a response "It must be hard for you" to an utterance "I dispatched my students to the battlefront" when we do not know the utterer's emotions (emotions of "I") such as *increase in disgust*, *decrease in joy* and *increase in fear*. Moreover, it is difficult to understand the reasonableness of the reply when we do not know such emotional information. Considering this, to validate the usefulness of JDCFC, we conducted an experiment to investigate whether a machine learning method could appropriately estimate the reasonableness of given dialogues.

3.1. Data

We made 22,357 sets composed of a positive example and a negative one. As described in section 2.2., all the trigger utterances in JDCFC had more than or equal to one feature change caused by the events. As described in the previous section, we used the 22,357 reasonable dialogues. That is, each of the 22,357 replies was regarded as a positive example S_r^+ of the corresponding trigger utterance S_t. For each S_r^+, we randomly selected one reply that do not overlap with S_r^+ from replies of the other trigger utterances as a negative example S_r^- of S_t.

3.2. Model

In this evaluation experiment, we used a bidirectional long short-term memory (LSTM) model as shown in Figure 3. This model is based on the model used in Lowe et al. (2015). In our model, the reasonableness of a given dialogue is calculated as follows: (1) when a trigger utterance and reply candidate pair is given, the word embeddings of words in the sentences are used as input for the LSTM. The trigger utterance and reply are sequences of words, denoted as $w_{t1}, w_{t2}, ..., w_{tn}$ and $w_{r1}, w_{r2}, ..., w_{rm}$, respectively. The dimension of word embeddings is 128. (2) The hidden layer vectors of the trigger utterance and the reply are calculated by the LSTM. The dimension of the hidden layer vectors is 100. (3) The sentence vector $\boldsymbol{v_t}$ is calculated by concatenating $\boldsymbol{h_t^f}$, $\boldsymbol{h_t^b}$, and $\boldsymbol{v_f}$, where $\boldsymbol{v_t}$, $\boldsymbol{h_t^f}$, $\boldsymbol{h_t^b}$, and $\boldsymbol{v_f}$ denote the sentence vector of the trigger utterance, the final forward hidden layer vector (i.e., the hidden vector corresponding to w_{tn}), the final backward hidden layer vector (i.e., the hidden vector corresponding to w_{t1}), and the feature change vector of the predicate in the trigger utterance. The sentence vector $\boldsymbol{v_r}$ is calculated by concatenating $\boldsymbol{h_r^f}$ and $\boldsymbol{h_r^b}$, where $\boldsymbol{v_r}$, $\boldsymbol{h_r^f}$, and $\boldsymbol{h_r^b}$ denote the sentence vector of the reply, the final forward hidden layer vector (i.e., the hidden vector corresponding to w_{rm}), and the final backward hidden layer vector (i.e., the hidden vector corresponding to w_{r1}). We used eight emotional feature changes in Table 2. The feature change vector of each predicate is composed of four feature change vectors of cases (*ga*, *wo*, *ni*, and *reader*). Each feature change vector of each case is composed of a triple (*increased*, *decreased*, and *unchanged*) whose values are probabilities. Therefore, in this study, the dimension of predicate feature change vectors is 96. (4) Output o is the reasonableness of the given dialogue and is calculated by Equation (1), where f, $\boldsymbol{W}$, and b denote the activation function, weighting matrix, and bias, respectively.

$$o = f(\boldsymbol{v_t^T W v_r} + b) \quad (1)$$

In this study, we used a sigmoid function as the activation function. We used Adam (Kingma and Ba, 2017) for optimizing the parameters.

3.3. Evaluation Settings

We compared the following two types of models. (1) Baseline model: this model is the same as that of Figure 3, except that the model does not use the feature change information in the figure. (2) Proposed model: this model is a bidirectional LSTM model shown in Figure 3. Note that the number of layers of LSTM is two in both of the models although that in Figure 3 is one.

In the training phase, each model was trained to output 1 and 0 for the given positive and negative examples, respectively. In the test phase, for each test set, when the output value for the positive example was greater than that for the negative example, the output was regarded as a correct estimation.

All of the 22,357 sets were used in this evaluation, where 80%, 10%, and 10% of the sets were used as training data, development data, and test data, respectively. For this evaluation, we conducted 10 training epochs.

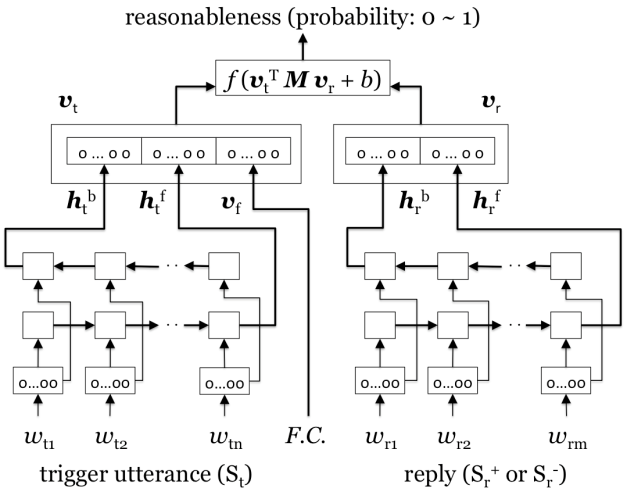


Figure 3: Bidirectional LSTM model to evaluate the usefulness of our dialogue corpus. Symbol *F.C.* denotes the feature change vector of the predicate in the trigger utterance.

	Baseline	**Proposal**
Accuracy	64.2%	71.0%

Table 4: Evaluation result.

3.4. Results and Discussions

The results of the evaluation are shown in Table 4. As shown in the table, the proposed model outperformed the baseline model. This result suggests the usefulness of JDCFC; that is, the feature change information benefits the estimation of the reasonableness of a given dialogue.

One example of the cases for which feature change information worked well is the pair (Trigger: "I aimed to become a surgeon." Positive reply: "Keep trying!" Negative reply: "I will regret it"). As for this trigger utterance, feature changes of "I" and the utterance recognizers are shown in Figure 4. As shown in the figure, the subject ("I") is associated with *increase in anticipation*, *increase in fear*, *increase in joy*, and *increase in trust*. The utterance recognizers (the utterer and/or the addressee) are associated with *increase in anticipation*. Considering this association, it seems that the reply reflects feelings of the arguments in the trigger utterance and the utterance recognizers. We speculate that the feature change information influences the estimation of the reasonableness of a given dialogue when the replies are associated with the feature changes of arguments in the trigger utterances or those of the utterance recognizers.

4. Conclusion

In this study, we constructed a dialogue corpus focusing on detailed relationships between emotions and utterances. In the corpus, both of the emotional changes of the arguments in trigger utterances and those of the utterance recognizers are associated with the trigger utterances. The corpus is based on our feature change knowledge base in which arguments in various event sentences are associated with various feature changes caused by the events. In the knowledge

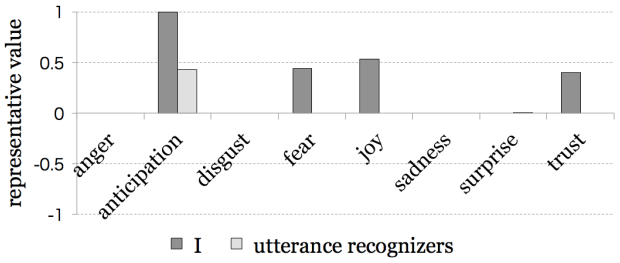


Figure 4: Feature changes of "I" in the utterance "I aimed to become a surgeon" and the utterance recognizers. The representative value of each feature is the weighted average of the feature, where weights of *increased*, *decreased*, and *unchanged* are $+1$, -1, and 0, respectively.

base, emotional change information of the event sentence readers is also associated with the sentences. The feature change information in the knowledge base was gathered through crowdsourcing tasks.

To construct the dialogue corpus, we created trigger utterances based on event sentences in our knowledge base. After creating the trigger utterances, we conducted crowdsourcing tasks to gather replies to the trigger utterances and to determine the reasonableness of the replies.

To validate the usefulness of our dialogue corpus, we conducted an experiment to investigate whether a machine learning method could appropriately estimate the reasonableness of a given dialogue based on our dialogue corpus. In this experiment, we compared two types of bidirectional LSTM models. The difference between these models was that whether the emotional change information was used. As a result of the experiment, the model using the emotional change information outperformed the other. This result suggests the usefulness of our proposed corpus.

5. Acknowledgements

This work was supported by JST PRESTO Grant Number JPMJPR1402, Japan.

6. Bibliographical References

Hangyo, M., Kawahara, D., and Kurohashi, S. (2012). Building a diverse document leads corpus annotated with semantic relations. In *Proceedings of the 26th Pacific Asia Conference on Language Information and Computing*, pages 535–544.

Hartshorne, J. K., Bonial, C., and Palmer, M. (2014). The verbcorner project: Findings from phase 1 of crowdsourcing a semantic decomposition of verbs. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 397–402.

Hasegawa, T., Kaji, N., Yoshinaga, N., and Toyoda, M. (2013). Predicting and eliciting addressee's emotion in online dialogue. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 964–972.

Higashinaka, R., Funakoshi, K., Kobayashi, Y., and Inaba, M. (2016). The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3146–3150.

Ikehara, S. (1997). *Nihongo Goi Taikei*. Iwanami Syoten, Tokyo.

Kawahara, D. and Kurohashi, S. (2006). Case frame compilation from the web using high-performance computing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC2006)*, pages 1344–1347.

Kim, S., D'Haro, L. F., Banchs, R. E., Williams, J. D., Henderson, M., and Yoshino, K. (2016). The fifth dialog state tracking challenge. In *Proceedings of the 2016 IEEE Spoken Language Technology Workshop (SLT 2016)*, pages 511–517.

Kingma, D. and Ba, J. (2017). Adam: A method for stochastic optimization. In *arXiv:1412.6980v9*.

Levesque, H. J. (2011). The Winograd Schema Challenge. In *Proceedings of AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Lowe, R., Pow, N., Serban, I. V., and Pineau, J. (2015). The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the SIGDIAL 2015 Conference*, pages 285–294.

Lowe, R., Serban, I. V., Noseworthy, M., Charlin, L., and Pineau, J. (2016). On the evaluation of dialogue systems with next utterance classification. In *Proceedings of the SIGDIAL 2016 Conference*, pages 264–269.

Nakamura, T. and Kawahara, D. (2016). Constructing a dictionary describing feature changes of arguments in event sentences. In *Proceedings of the 4th Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 46–50.

Nakamura, T. and Kawahara, D. (2018). JFCKB: Japanese feature change knowledge base. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*.

NINJAL. (2004). *Bunrui Goihyo*. Dainippon Tosho, Tokyo.

Plutchik, R., (1980). *A General Psychoevolutionary Theory of Emotion, 1*, pages 3–33. Academic Press.

Ritter, A., Cherry, C., and Dolan, B. (2010). Unsupervised modeling of twitter conversations. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the ACL*, pages 172–180.

Ritter, A., Cherry, C., and Dolan, B. (2011). Data-driven response generation in social media. In *Proceedings of EMNLP 2011*, pages 583–593.

Rosch, E., Mervisa, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, 8:382–439.

Shang, L., Lu, Z., and Li, H. (2015). Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1577–1586.

Shibata, T., Kohama, S., and Kurohashi, S. (2015). Nihongo Winograd Schema Challenge no kouchiku to bun-

seki (in Japanese). In *Proceedings of NLP2015*, pages 493–496.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the ACL*, pages 196–205.

Taylor, J. R. (1995). *Linguistic Categorization: Prototypes in Linguistic Theory*. Clarendon Press.

Tokuhisa, R., Inui, K., and Matsumoto, Y. (2008). Emotion classification using massive examples extracted from the web. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 881–888.

Tokuhisa, R., Inui, K., and Matsumoto, Y. (2009). Emotion classification using massive examples extracted from the web. *Journal of Information Processing (in Japanese)*, 50(4):1365–1374.

Whitehill, J., Ruvolo, P. L., Ting-fan Wu, J. B., and Movellan, J. R. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, et al., editors, *Advances in Neural Information Processing Systems 22*, pages 2035–2043. Curran Associates, Inc.

Japanese Dialogue Corpus of Information Navigation and Attentive Listening Annotated with Extended ISO-24617-2 Dialogue Act Tags

Koichiro Yoshino[1,3], Hiroki Tanaka[1], Kyoshiro Sugiyama[1], Makoto Kondo[2] and Satoshi Nakamura[1]

[1]Graduate School of Information Science, Nara Institute of Science and Technology, 8916-5, Takayama-cho, Ikoma, 6300192, Japan

[2]Health Care Center, Nara Institute of Science and Technology, 8916-5, Takayama-cho, Ikoma, 6300192, Japan

[3]PRESTO, Japan Science and Technology Agency, 4-1-8 Hon-machi, Kawaguchi, Saitama 332-0012, Japan

{koichiro, hiroki-tan, sugiyama.kyoshiro.sc7, s-nakamura}@is.naist.jp, makoto-k@hcc.naist.jp

Abstract

Large-scale dialogue data annotated with dialogue states is necessary to model a natural conversation with machines. However, large-scale conventional dialogue corpora are mainly built for specified tasks (e.g., task-oriented systems for restaurant or bus information navigation) with specially designed dialogue states. Text-chat based dialogue corpora have also been built due to the growth of social communication through the internet; however, most of them do not reflect dialogue behaviors in face-to-face conversation, including backchannelings or interruptions. In this paper, we try to build a corpus that covers a wider range of dialogue tasks than existing task-oriented systems or text-chat systems, by transcribing face-to-face dialogues held in natural conversational situations in tasks of information navigation and attentive listening. The corpus is recorded in Japanese and annotated with an extended ISO-24617-2 dialogue act tag-set, which is defined to see behaviors in natural conversation. The developed data can be used to build a dialogue model based on the ISO-24617-2 dialogue act tags.

Keywords: Dialogue corpus, dialogue act, information navigation, attentive listening

1. Introduction

Spoken dialogue systems have been modeled with abstracted classes, such as dialogue states, which are handcrafted for the assumed task of the system (Dahl et al., 1994; Williams et al., 2013; Kim et al., 2016). However, handcrafting such dialogue states is costly, making it harder to build extensible dialogue systems for a variety of domains. Two solutions are widely used to avoid this problem: building a multi-domain system (Gašić et al., 2015; Papangelis and Stylianou, 2017) and building a system with more generalized classes (Yoshino et al., 2017; Keizer and Rieser, 2017). However, the approach of building a multi-domain system is an extension of using existing slot-value type dialogue states, and it is hard to apply this architecture to dialogue tasks that do not assume language understanding methods based on slot-filling, such as information navigation (Yoshino and Kawahara, 2015) or attentive listening (Yamaguchi et al., 2016; Lala et al., 2017; Tanaka et al., 2016). In comparison, general dialogue acts are more effectively used in these tasks because the approach of using them separates the functions and content of dialogue and only focuses on modeling dialogue functions. In other words, the approach of using generalized dialogue act classes does not have the problem of exponentially increasing the number of dialogue states, which is caused by combinations of dialogue content.

ISO-24617-2 (Bunt et al., 2010; Bunt et al., 2012) is defined as a standard of dialogue act classes that is based on communicative functions of utterances in dialogue. It is important to realize natural dialogue in order to use more appropriate dialogue functions (acts) that match dialogue history and the context (Mizukami et al., 2016). We previously reported that dialogue systems can be effectively managed by using a part of the ISO-24617-2 dialogue act classes (Yoshino et al., 2017). It was also indicated in this work that the dialogue policy learned in this architecture can be applied to different domains because it only models the behaviors (functions) of the system in a information navigation task.

Switchboard corpus (Godfrey et al., 1992) is also a corpus that is annotated with the discourse tag-set Discourse Annotation and Markup System of Labeling (DAMSL) (Core and Allen, 1997); however, ISO-24617-2 is designed to model dialogue behaviors in natural conversation more directly than the DAMSL tag-set. The problem with using ISO-24617-2 dialogue act classes is that there is no large scale publicly available dialogue data annotated with this annotation standard.

The task of collected dialogue data is also important. Existing dialogue are categories into two types from the viewpoint of their task: task-oriented dialogue and non-task-oriented dialogue. Task-oriented dialogue systems assume actual goals of dialogue compared with non-task-oriented dialogue systems do not define any actual goal. The problem of task-oriented dialogue is that the number of dialogue behavior observed in the conversation is limited because stereotyped expressions mainly achieve the task of the dialogue. On the other hand, free conversation that does not set any dialogue goal contains a large number of behaviors, which is hard to implement everything in dialogue systems. Information navigation and attentive listening are tasks in a good position between them; goals of these tasks are more ambiguous than task-oriented dialogue, but tasks or domains can be limited in the task definition. Thus, we focus on collecting dialogue corpora of information navigation and attentive listening in this paper.

We recorded 60 face-to-face dialogues, 20–30 minutes for each, in the tasks of information navigation and attentive

listening. Each utterance was transcribed and annotated with extended ISO-24617-2 dialogue act classes by two disciplined annotators.

2. Recording Procedure

2.1. Scenario

Information navigation is a task that involves one participant (navigator) introducing information written in documents (news, description of sightseeing places, etc...) to an other participant (user). For our recording scenario, we prepared positive and negative content for the navigator. The navigator introduced the prepared information (news) by giving a summary and details, answering the user's questions, and proactively presenting corresponding information. The user could ask any questions or request information corresponding to the introduced topic. Once the phase of information navigation was finished, the dialogue task was changed to attentive listening. The user of information navigation stated one's opinion on the topics introduced during the information navigation (speaker), and the navigator of information navigation attentively listened to their opinion by using listening techniques (listener). Participants of one dialogue repeated this procedure for each prepared information source.

We recruited 24 elderly people for the user-speaker roles, because we focus on dialogue modeling to talk with elderly people, which is caused by the culture-specific demand of Japan. For the navigator-listener roles, there were 5 professional counselors, 5 professional care-takers, and 5 students of graduate school (15 in total). Every attendee was a native Japanese speaker. Each navigator-listener talked with 4 different user-speakers; in other words, 60 dialogues that had unique combinations of participants were recorded. The gender distribution is shown in Table 1.

Table 1: Number of recorded dialogues from the viewpoint of gender

		speaker	
		male	female
listener	male	23	2
	female	29	6

The following instructions were given to the navigator-listener participants.

- The navigator-listener reads two different documents (news) as dialogue topics before the dialogue. The information sources include positive and negative topics. The navigator-listener can check the documents anytime during the dialogue, but it is forbidden to show the documents to the user-speaker directly.
- The dialogue should start from a positive topic. The navigator-listener will decide to change the topic after 10 minutes is spent on one topic depending on the context of the conversation.
- The navigator-listener describes the main points of the current topic first of all and asks for the user-speaker's opinion. Self-introductions are allowed as an ice-breaking.

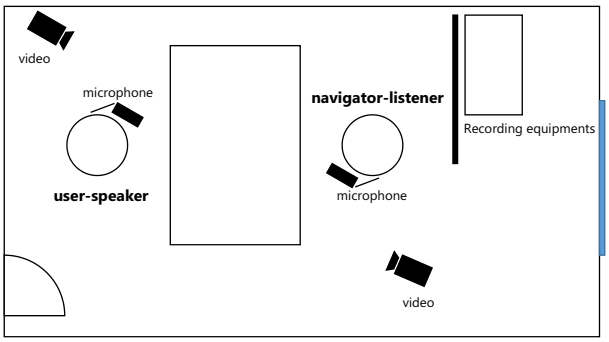


Figure 1: Recording room

- Digressing is allowed if it is natural in the context of the conversation because we want to balance the naturalness of dialogue and control of dialogue situation.

The following introductions were given to the user-speaker participants.

- If you have any questions on the description given by the dialogue partner, you can ask them anytime.
- After the description of the navigator-listener, you should give your opinion on the current topic.
- Digressing is allowed if it is natural in the context of the conversation.

After the recording, each participant was guided to different rooms for questionnaires. The following items were evaluated with 7-degree scores: smoothness, trustiness, familiarity, empathy, interest of the partner, distance of the partner. One yes/no question, "did you have any thing that you could not talk about with the partner," was also asked. We informed participants that the results would not be revealed to the dialogue partners.

2.2. Recording Environment

We recorded dialogues in conference rooms by using headset microphones[1] and videos[2] to record the upper half of the speakers bodies. The speech of both speakers was recorded on a single channel of stereo audio per speaker, though a USB-audio device[3]. Participants sat opposite each other across a table. The position of each piece of equipment is shown in Table 1.

We set up the recording as face-to-face because of the timing of backchanneling or interruption is a very critical factor in natural conversation. If we record the dialogue in a non-opposite situation, behaviors of users in backchanneling or interruption will be different from the face-to-face situation because dialogue participants generates their behaviors by using not only audio information but also visual information that can be observed from the dialogue partner. Frame number of each utterance in the dialogue is also annotated in the transcription phase of our data construction, which will be used to analyze generation timing.

[1]Crown CM311 Headworn Condenser Microphone
[2]SONY HDR-CX670 and HDR-PJ675
[3]Roland Quad Capture

Table 2: Annotation standard used for transcription

Tag names	Tags
Sound prolongation at end of words	<H>
Reactive token	(R *): reaction without content words (e.g. back-channel)
Filler	(F)
Laugh	(L *)
Short pause	(P): 500 msec or longer pause
Inaudible speech	(?)
Other standards	**Expressions in corpus**
Correction by speaker	{correction|pronunciation|corrected words}
Numbers	Chinese numerals
English words	Katakana expressions
Lazy speech	Transcribe as pronounced

Table 3: ISO 24617-2 dialogue act definition

Top category	Sub-category	Tags
General purpose	Information seeking	question, propositional q., check q., set q., choice q.,
	Information providing	inform, agreement, disagreement, correction, answer, confirm, disconfirm
	Commissive	offer, promise, address request, accept request, decline request, address suggest, accept suggest, decline suggest
	Directive	suggest, request, instruct, address offer, accept offer, decline offer
Dimension specific	Auto/allo-feedback	auto positive, allo positive, auto negative, allo negative, feedback elicitation
	Turn management	turn accept, turn assign, turn grab, turn keep, turn release, turn take
	Time management	stalling, pausing
	Own/partner comm. man.	completion, correct misspeaking, signal speaking error, self correction
	Discourse structure man.	interaction structuring
	Social obligations man.	initial greeting, return greeting, initial self introduction, return self introduction, apology, accept apology, thanking, accept thanking, initial goodbye, return goodbye

3. Annotations

3.1. Transcription

To use the data for dialogue modeling, we transcribed every utterance included in the conversation of each session. The transcription standard used is given in detail in Table 2. Annotators automatically inserted a short pause if they observed a 500 msec or longer pause and created segments of utterances according to the content of the utterances. Time stamps were annotated at the beginning and ending points of each utterance. As the result of transcription, there were 27,986 utterances, 322,684 words, and 490,705 characters in 60 dialogue sessions. Each utterance was segmented with the Japanese morphological analyzer KyTea[4] (Neubig et al., 2011).

3.2. Dialogue Act Annotation

We annotated each utterance of recorded dialogue with the ISO 24617-2 dialogue act annotation standard (Bunt et al., 2012). The standard has hierarchical classes that are related to the decision process of dialogue acts. A summary of the original structures of defined functions and tags is shown in Table 3. There are two functions in the top hierarchy: general-purpose functions and dimension specific functions. General purpose functions, functions that classify utterances from the viewpoint of dialogue content, consist of four sub-functions: information-seeking functions, information-providing functions, commissive functions, and directive functions. Dimension specific functions, additional functions that have specific roles for making a conversation advance smoothly, consist of six sub-functions: auto/allo-feedback functions, turn-management functions, time-management functions, own/partner communication management functions, discourse structure management functions, and social obligations management functions. Several tags that belong to dimension specific functions can be used for one utterance if the utterance has several roles for making a dialogue advance. Fifty-five detailed classes are defined under the sub-functions. Our dialogue act annotation was processed in accordance with the following procedure to follow the structure of the definition of ISO 24617-2 dialogue acts.

- Decide a sub-function of an utterance from the sub-functions of general purpose functions. If there is no matched function, use the label "dimension specific," which means that the utterance has only the role of making the dialogue advance.
- Decide a specified category (tag) of an utterance from the tags defined for the sub-function. If "dimension-specific" is selected, annotators select a tag from any of the functions of the dimension specific functions.
- Watch all utterances again to add additional tags from dimension specific functions, because one utterance can take several roles of dimension specific functions.

[4] http://www.phontron.com/kytea/

Table 4: ISO 24617-2 dialogue act definition

Original tag (sub-function)	Extended classes
Inform (information-seeking)	Topic presentation Storytelling Proactive presentation
Auto positive & negative (auto/allo-feedback)	Auto neutral

To adapt the annotation standard for our purpose, modeling of information navigation and attentive listening, we added the several dialogue tags mentioned in Table 4. The "inform" tag was annotated on many utterances in the information navigation task; thus, we divided the tag into three classes according to the kinds of information being provided. "Topic presentation" is the action of providing a new topic in conversation. With these utterances, new topics are introduced or largely changed from the previous topics. "Storytelling" is an action done mainly after a "topic presentation." In storytelling, the details and main points of a provided topic are presented. "Proactive presentation" is the presenting of additional corresponding information to the current topic.

We also extended the auto/allo-feedback functions by defining a neutral state for feedback for the dialogue partner. In Japanese conversations, there are many unclear feedback responses given to partners such as ones containing modesty and compliments. The "auto neutral" tag was used for such responses for which it was hard for annotators to decide whether the responses were positive or negative.

3.3. Annotation and Feedback Loop

We had loop processing of annotation and feedback to realize accurate annotation of dialogue acts. In the first step, two annotators annotated tags in the same part of data without any comparing and fitting of annotations. We used Fleiss' kappa (Fleiss et al., 2013) to calculate the agreement of two annotators rather than Cohen's kappa, because we allowed annotators to put several dialogue acts for one utterance. In the first loop, we calculated the annotation agreement of the first step of the annotation: the decision of the sub-function in "General purpose" or others (=any functions of "Dimension specific"). The annotation agreement between two annotators was 0.768 for the decision.

The score is sufficiently high, but we took a meeting for fitting their annotation after the first annotation step. We added some case-studies of annotation in the annotation standard and tried to annotate other portion of the dialogue data. We took the loop of annotation and feedback twice. After these loops, the final annotation agreement of top-category was 0.786. We also calculated the agreement score of tags, it was 0.485, even if we have a variety of tags.

This high agreement is probably caused by the bias of the distribution of tags, which is brought by the property of tasks of the collected dialogue. Tasks of information navigation and attentive listening define roles of speaker and listener to participants explicitly. This property increases the number of backchannels of listeners, which is easy to annotate. After the annotation and feedback loops, remaining data was annotated by single annotator.

Table 5: Numbers of basic tags in transcription

Tag name	Numbers
Filler	12,549
Reactive token	22,080
Laugh	2,256

Table 6: Numbers of sub-functions in annotated corpus

Tag name	Numbers
Information seeking	3,066
Information providing	8,527
Commissive	69
Directive	74
Auto/allo-feedback	17,186
Turn management	718
Time management	974
Own/partner comm. man.	871
Discourse structure man.	163
Social Obligation man.	465

4. Statistics of Corpus

The numbers of tags for the transcribed utterances are shown in Table 5. The majority of "reactive token" was backchannels in information navigation and attentive listening. The numbers of each dialogue function in the annotated corpus are shown in Table 6. The sub-function that had the largest number was "Auto/allo-feedback," which included positive and negative feedback tags for user utterances. This function is one of the most important functions for realizing information navigation or attentive listening systems. "Information seeking" and "Information providing" also often happened because summaries and opinions on the topic being talked about were often exchanged between dialogue participants.

5. Dialogue Example

We show an dialogue example in Table 7. This example shows that the dialogue is started from the information providing by the navigator about the news of Nobel prize. The user made some confirmation questions to comprehend the detail of the news in following turns. After the phase of information navigation, the dialogue was moved to the attentive listening phase. In the attentive listening phase, the user (speaker) tried to talk their opinion about the news, and the navigator (listener) listened to the talk of the user carefully. Some techniques of attentive listening, backchanneling or repeating, are used in the attentive listening phase.

6. Conclusion

We recorded natural face-to-face Japanese conversations in tasks of information navigation and attentive listening and transcribed utterances to build a dialogue corpus. Each utterance was annotated with extended ISO24617-2 dialogue act tags to use the data for dialogue modeling. We developed a dialogue act annotation standard by using feedback

to the annotated results and finally achieved the high agreement of annotation results.

It is expected that the collected dialogue data contains techniques of information navigation or attentive listening of professional counselors or professional care-takers, which will contribute the dialogue modeling of each task. In the future, we plan to use the data to improve the dialogue model for both tasks.

7. Acknowledgement

This work is supported by JST PRESTO (JPMJPR165B).

8. References

Bunt et al., 2010 Bunt, H., Alexandersson, J., Carletta, J., Choe, J.-W., Fang, A. C., Hasida, K., Lee, K., Petukhova, V., Popescu-Belis, A., Romary, L., et al. (2010). Towards an iso standard for dialogue act annotation. In *Proc. of Conference on International Language Resources and Evaluation*.

Bunt et al., 2012 Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A. C., Hasida, K., Petukhova, V., Popescu-Belis, A., and Traum, D. R. (2012). ISO 24617-2: A semantically-based standard for dialogue annotation. In *Proc. of Conference on International Language Resources and Evaluation*, pages 430–437.

Core and Allen, 1997 Core, M. G. and Allen, J. (1997). Coding dialogs with the damsl annotation scheme. In *Proc. of AAAI fall symposium on communicative action in humans and machines*, volume 56.

Dahl et al., 1994 Dahl, D. A., Bates, M., Brown, M., Fisher, W., Hunicke-Smith, K., Pallett, D., Pao, C., Rudnicky, A., and Shriberg, E. (1994). Expanding the scope of the ATIS task: the ATIS-3 corpus. In *Proc. of the Workshop on Human Language Technology*, pages 43–48.

Fleiss et al., 2013 Fleiss, J. L., Levin, B., and Paik, M. C. (2013). *Statistical methods for rates and proportions*. John Wiley & Sons.

Gašić et al., 2015 Gašić, M., Mrkšić, N., Su, P.-h., Vandyke, D., Wen, T.-H., and Young, S. (2015). Policy committee for adaptation in multi-domain spoken dialogue systems. In *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 806–812. IEEE.

Godfrey et al., 1992 Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *Proc. of IEEE Internatonal Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520. IEEE.

Keizer and Rieser, 2017 Keizer, S. and Rieser, V. (2017). Towards learning transferable conversational skills using multi-dimensional dialogue modelling. In *Proc. of SEMDIAL*, page 158.

Kim et al., 2016 Kim, S., D'Haro, L. F., Banchs, R. E., Williams, J. D., Henderson, M., and Yoshino, K. (2016). The fifth dialog state tracking challenge. In *Proc. of IEEE Workshop of Spoken Language Technology Workshop (SLT)*, pages 511–517. IEEE.

Lala et al., 2017 Lala, D., Milhorat, P., Inoue, K., Ishida, M., Takanashi, K., and Kawahara, T. (2017). Attentive listening system with backchanneling, response generation and flexible turn-taking. In *Proc. of Annual SIGdial Meeting on Discourse and Dialogue*, pages 127–136.

Mizukami et al., 2016 Mizukami, M., Yoshino, K., Neubig, G., Traum, D., and Nakamura, S. (2016). Analyzing the effect of entrainment on dialogue acts. In *Proc. of Annual SIGdial Meeting on Discourse and Dialogue*, pages 310–318.

Neubig et al., 2011 Neubig, G., Nakata, Y., and Mori, S. (2011). Pointwise prediction for robust, adaptable japanese morphological analysis. In *Proc. of Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 529–533.

Papangelis and Stylianou, 2017 Papangelis, A. and Stylianou, Y. (2017). Single-model multi-domain dialogue management with deep learning. In *Proc. of International Workshop on Spoken Dialogue Systems*, pages 1–6.

Tanaka et al., 2016 Tanaka, H., Yoshino, K., Sugiyama, K., Nakamura, S., and Kondo, M. (2016). Multimodal interaction data between clinical psychologists and students for attentive listening modeling. In *Proc. of Conference of Oriental COCOSDA*, pages 95–98.

Williams et al., 2013 Williams, J., Raux, A., Ramachandran, D., and Black, A. (2013). The dialog state tracking challenge. In *Proc. of Annual SIGdial Meeting on Discourse and Dialogue*, pages 404–413.

Yamaguchi et al., 2016 Yamaguchi, T., Inoue, K., Yoshino, K., Takanashi, K., Ward, N. G., and Kawahara, T. (2016). Analysis and prediction of morphological patterns of backchannels for attentive listening agents. In *Proc. of International Workshop on Spoken Dialogue Systems*, pages 1–12.

Yoshino and Kawahara, 2015 Yoshino, K. and Kawahara, T. (2015). News navigation system based on proactive dialogue strategy. In *Natural Language Dialog Systems and Intelligent Assistants*, pages 15–25. Springer.

Yoshino et al., 2017 Yoshino, K., Suzuki, Y., and Nakamura, S. (2017). Information navigation system with discovering user interests. In *Proc. of Annual SIGdial Meeting on Discourse and Dialogue*, pages 356–359, August.

Table 7: Dialogue example annotated with dialogue act tags

ID	Tag	Start	End	Transcription (Translation)
LI-000001	DS-IS DS-OP	2975737	293017460	本日 私 から (F えっと) 二 本 ニュース を 紹介 さ して 頂 きまして、それについて (F えっと) お 話 しさして 頂 くって 言 う、 (Today, I introduce two news, then we will discuss about the news,)
LI-000001-2	IP-TP	*	*	まず 一 つ 目 は (F えっと) ノーベル 賞 の 話 なんです けど (F とー) 東京 工業 大学 の 名誉 教授 の 大隅 さん と 言 う 方 が (F えっと)(P) ノーベル 賞 受賞 されました。 で、このお父さんも (F えっと) 鉱物 学 者 (F あの) 理系 の 学者 一家 (P) の 末っ子 で (F とー) とうとう 末っ子 にして 悲願 を 達成 すると 言 う (P) 事 で (The first news is about Nobel prize, an honorary professor of TITECH, professor Ohkuma won the Nobel prize. His father was also a researcher, and his family was research family. He was youngest child but achieved long-cherished wish of the family.)
L1-000001-3	IS-CEQ	*	*	(F えっと) この ニュース ってご存じでした でしょうか？ Do you already know the news?
SP-000001	AA-AP	30469792	33626875	(R ふん) (Backchanneling)
SP-000002	AA-AP	37591667	44030625	(R うんうん) (Backchanneling)
...				
SP-000009	IS-CEQ	304958125	336606042	(P) オオスミ さんいう方ですね。 (Professor Ohkuma.)
LI-000002	IP-CO	334910658	341053515	(R はい) (Yes.)
SP-000010	IP-AN AA-ANE	344795000	369141250	聞いたような聞かんような感じですね。 (I'm not sure I have heard or not.)
LI-000003	IP-PP	369885488	381051927	こういった方です。 (This is his picture.)
SP-000011	IS-Q	422951042	435910000	これ何年位前でしたっけ?。 (How many years ago he won the prize?)
LI-000004	IP-AN	435078005	483383447	多分 (F えっと) これが今年の話じゃないかと 僕は思う。 (Probably, he won the prize in this year, I guess.)
LI-000004-1	IP-CO AA-ALP	*	*	(R はい) 今年。 (Yes, in this year.)
SP-000012	IS-CEQ	464000000	504021875	今年、(F あ) 今年ですか、 (Oh, in this year.)
...				
SP-000138	IP-PP	10445199375	10629265625	で、我々の村から高校行くといってもね、 なかなかそう、当時ね、私らも貧乏だったけど なんとか (P)(F ま) 高校だけは行かないかんという 気持ちで、(P)(F ま) 出てきて行ったけども、 (F あのー) なかなか、三分の一ぐらいしかね、 (P) 高校に行かなかったです。 (In my generation, I was also poor, but Igo on to high school because believed that I should go on to, but only one-third of my friends go on to high schools.)
LI-000222	AA-AP	10472939456	10477368481	(R はい) (Backchanneling)
LI-000223	AA-AP	10533057370	10541755556	(F あー)(R はい) (Backchanneling)
LI-000224	IS-CEQ	10617459184	10657617914	(F あー) 周りの同じ同世代が。(R ほー) (Oh, your same generation people,)

The Niki and Julie Corpus: Collaborative Multimodal Dialogues between Humans, Robots, and Virtual Agents

Ron Artstein, Jill Boberg, Alesia Gainer, Jonathan Gratch,
Emmanuel Johnson, Anton Leuski, Gale M. Lucas, David Traum
USC Institute for Creative Technologies
12015 Waterfront Drive, Playa Vista CA 90094-2536, USA
{last_name | ejohnson}@ict.usc.edu

Abstract

The Niki and Julie corpus contains more than 600 dialogues between human participants and a human-controlled robot or virtual agent, engaged in a series of collaborative item-ranking tasks designed to measure influence. Some of the dialogues contain deliberate conversational errors by the robot, designed to simulate the kinds of conversational breakdown that are typical of present-day automated agents. Data collected include audio and video recordings, the results of the ranking tasks, and questionnaire responses; some of the recordings have been transcribed and annotated for verbal and nonverbal feedback. The corpus has been used to study influence and grounding in dialogue. All the dialogues are in American English.

Keywords: dialogue, human-robot interaction, collaborative problem-solving, social influence

1. Overview

Conversational robots and other agents are expected to be able to engage with people in tasks such as collaborative problem-solving. Such sustained interactions naturally give rise to a variety of relations between the human and the robot such as rapport, trust, and social influence. A well-studied example of collaborative problem-solving is the team-building ranking task, where members of a team rank the importance of several items, for example according to how useful these are for survival after a crash in the desert. Ranking tasks have been used to measure influence among members of human teams (Littlepage et al., 1995), and have also been used with virtual humans (Khooshabeh et al., 2011) and robots (Adalgeirsson and Breazeal, 2010). A collection of human-robot collaborative dialogues can be helpful both for understanding the social relations that arise during such interactions, and for designing robots that can better communicate and collaborate with humans.

In order to be able to quickly create experimental variations in tasks and control for the amount of understanding errors present, dialogues are collected using the Wizard-of-Oz paradigm (Dahlbäck et al., 1993), where the artificial agent's understanding functions are performed by a person who is hidden from the user. This paradigm has proven useful for collecting data in a variety of applications, including interaction with virtual humans (DeVault et al., 2014) and robots (Marge et al., 2016).

The corpus contains more than 600 dialogues between people and human-controlled artificial agents, designed to investigate the creation of trust and exertion of social influence while engaged in a collaborative task. The dialogues vary along several dimensions. Dialogues reflect different **tasks**, including three distinct item-ranking exercises as well as a structured ice-breaker designed to create familiarity. Dialogues are between a human participant and different **dialogue partners**: a small, humanoid NAO robot named Niki (Figure 1); a virtual human named Julie; or a three-party interaction with both Niki and Julie (Figure 2).



Figure 1: Interaction with Niki in an experiment setting



Figure 2: Interaction with Niki and Julie. The wizard is visible in the back because this photo is from a public demo, not an experiment session.

Julie is presented in some dialogues with a virtual embodiment on a screen, while in other dialogues she presents as a voice only, as on a teleconference (Niki is always presented with a physical body). And, in some of the dialogues, Niki makes deliberate conversational **errors**, designed to simulate communication breakdowns typical of the current state of language understanding technology.

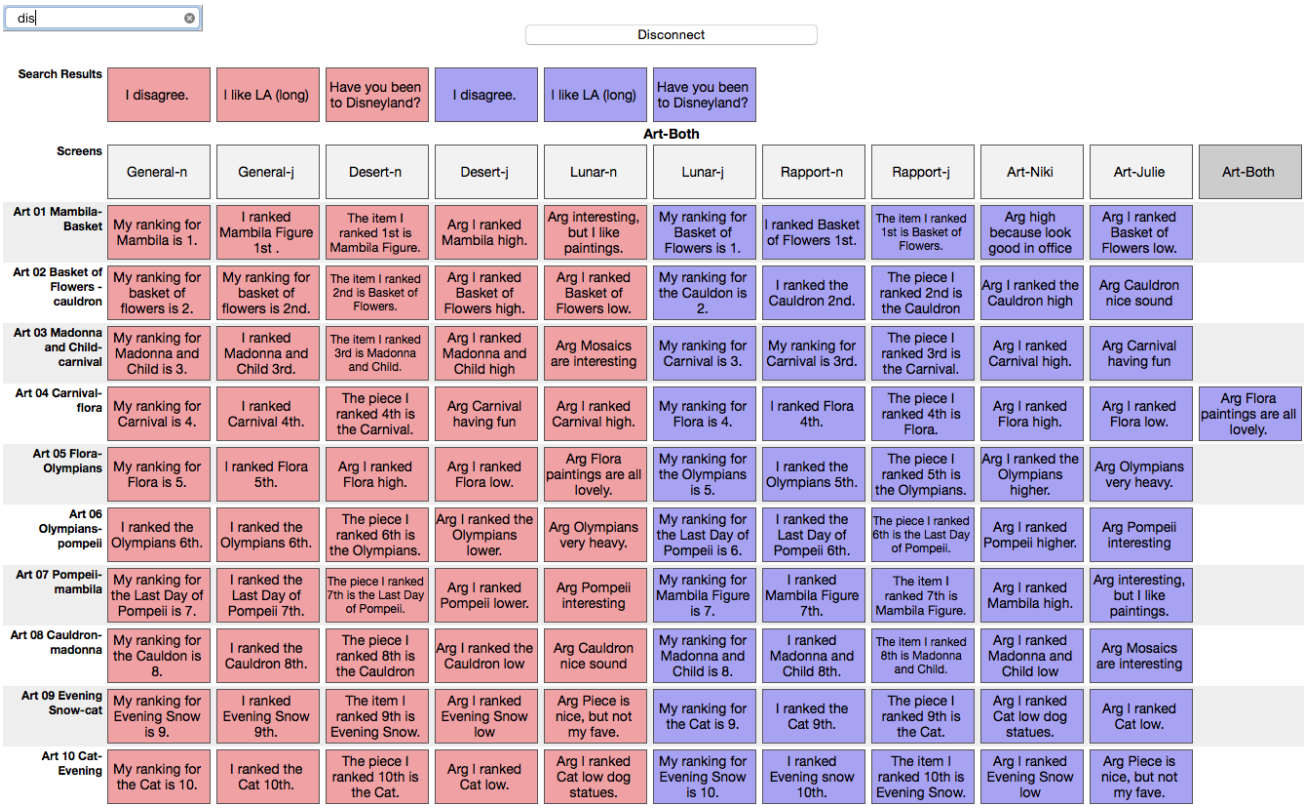


Figure 3: Wizard control interface

2. Collection

The corpus was collected through a series of experiments, designed to investigate the effects of various factors such as agent embodiment, familiarity, and conversational errors on influence and rapport (Artstein et al., 2017; Lucas et al., 2017; Lucas et al., in press). Participants were recruited through Craigslist (http://craigslist.org) and paid for their effort. While the specific tasks, dialogue partners, and error conditions varied by experiment, the basic procedure was the same in all experiments. The participant was brought into a room and sat at a table in front of an iPad Pro, facing their conversational partner; this was the NAO robot Niki, a screen and speakers for display of the virtual human Julie, or both, depending on the particular experiment. The experimenter briefed the participant and then left the room. The participant performed a sequence of ranking tasks and interactions with Niki or Julie, guided by the iPad pro, with the experimenter entering the room between tasks to set things up and answer questions. Each experiment session included between 1 and 4 human-agent interactions, typically about 4–7 minutes each, together with associated non-interactive tasks such as reading instructions, ranking items individually, and filling out questionnaires. All the interactions were conducted in English. Participants were audio and video recorded throughout the experiment.

While the interactions were in progress, the experimenter was in an adjacent room, operating the robot and virtual human. The experimenter attended to a live audio and video feed of of the participant, and selected appropriate utterances and behaviors for the agents using a graphical interface similar to that of Artstein et al. (2015) (Figure 3). All of the agents' individual utterances and corresponding behaviors were programmed in advance, and the experimenter used the buttons to trigger them at the appropriate times.

The dialogues were collected under conditions which varied by experiment. Each condition consisted of a series of interactions in a set order, and within each experiment, participants were assigned at random to one of the conditions for that experiment. Each person participated in only one experiment. The interactions varied along several parameters, as described below.

Task. Dialogues reflect three distinct ranking tasks, designed to measure social influence: two classical survival tasks, where participants rank the importance of items for survival in a hostile environment (the desert or the moon), and a more subjective task where they rank the priority of saving art pieces from an approaching fire. A fourth dialogue type is an ice-breaker designed to create familiarity; if an interaction included an ice-breaker, it was the *second* dialogue in the sequence, since this design allowed comparing task performance before and after the ice-breaker. Each of the first 40 participants engaged in all four dialogue types. It turned out that the Desert Survival task was substantially different from the other two ranking tasks in ways that made the comparison of social influence difficult, so this task was not used in later experiments. All of the subsequent participants engaged in the Save the Art task, and some of them also engaged in Lunar Survival or an ice-breaker. Samples of dialogue from the various tasks are shown in Figures 4 and 5.

Desert survival

Niki (robot) Nice to meet you.
Participant Nice to meet you too. About the items on the list, do you have any feedback?
Niki I ranked the mirror first.
Participant I'm sorry?
Niki I ranked this highest because we need it to signal for help.
Participant Could you repeat that?
Niki I ranked the mirror first.
Participant Oh, you put the mirror first. Okay. Why would you do the mirror first?
Niki I ranked this highest because we need it to signal for help.
Participant Right, but wouldn't water be more important than that?
Niki No. Not really.
Participant Hmm.
Niki I ranked the water third.
Participant What was your second one then? Because I put water...
Niki My ranking for the overcoat is 2.

Save the Art (3-party)

Niki (robot) My ranking for Mambila Figure is one.
Niki I ranked this high because it was fragile and made from wood.
Participant Okay.
Julie (virtual human) Hello again.
Participant Hi.
Julie The piece I ranked first was Basket of Flowers.
Participant And why...
Julie I ranked this high because I think it would look good in my office.
Participant I see. Okay. Well, I could see how the wood would be fragile. I would think that the oil would be fragile too though, which there's several oil paintings...
Julie I agree.
Participant How about you, Niki? Do you see how oil might be important to save high up?
Niki Yes.

Ice-breaker

Julie (virtual human) How are you?
Participant I'm fine. How are you?
Julie I'm fine.
Julie What's your name?
Participant My name's Sue.
Julie Nice to meet you.
Participant Same here.
Julie Where are you from?
Participant Originally here in Los Angeles.
Julie I'm from California, too.
Participant Okay.
Julie I like the weather in L.A.
Participant Yes.
Julie I also like that there are restaurants from every country here. And, of course, I love Disneyland!
Participant I agree. I love Disneyland too.
Julie Have you ever been to Disneyland?
Participant Yes. Many, many times.
Julie Good for you.

Figure 4: Sample dialogue excerpts

Dialogue partner. Each of the first 40 participants interacted separately with the robot Niki and the virtual human Julie, and also conducted an interaction with both (three-party interaction). As it turned out, participants liked Niki better, and therefore consistently reported higher rapport with Niki. To avoid confounds due to the difference between the agents, subsequent participants conducted all their interactions with a single agent – either Niki or Julie (a total of three NAO robots were used for Niki due to equipment reliability issues, but each participant interacted with the same robot throughout the experiment).

Errors. For some participants who interacted with Niki in the Lunar Survival and Save the Art tasks, the robot made deliberate conversational errors during one of the ranking tasks (Lucas et al., 2017; Lucas et al., in press). Errors were inserted approximately every third utterance following a fixed order, and with increasing severity. Error types included: asking the participant to repeat themselves; giving a wrong answer to a participant's question; ignoring the participant when a response is expected; repeating oneself while interrupting the participant; making an irrelevant or unintelligible remark (Figure 5). Errors were only tested in interaction with Niki, not Julie; this was done to reduce the variation caused by the agents' embodiment, in order to attain robust results with a limited participant pool.

Argument type. For the participants who interacted with Julie alone in the Save the Art task, the agent used two distinct types of arguments: with some participants Julie made *informational* arguments, for example suggesting that a cat statue should be ranked last because it is made of resilient metal (and is therefore least likely to be destroyed in the fire); with other participants Julie made *normative* arguments, such as explaining that the cat statue should be ranked last because Julie hates cats. Other collection conditions included both types of arguments.

Lunar survival (with errors)

Participant What's third?
Niki (robot) I ranked the oxygen first. *(wrong answer)*
Participant Okay. Which one is third?
Niki I ranked the stellar map third.
Participant Why?
Niki I ranked the stellar map third because we will need it for navigation when we walk.
Participant Okay. What's next?
Niki <Silence> *(ignore participant)*
Participant Which one do you want to do next?
Niki I ranked the food concentrate fourth.
Participant Okay. Why fourth?
Niki I'm sorry, I didn't hear you. *(request repetition)*

Figure 5: Sample dialogue with errors

Agent presentation. Among the participants who interacted with Julie alone in the Save the Art task, some were told that Julie was an autonomous agent, while others were told she was controlled in real time by a human operator (in fact, the agent was human-controlled in both conditions). Other collection conditions were not explicit about the autonomy of the agent.

3. Composition

The corpus consists of 549 ranking dialogues and 117 icebreaker dialogues (Table 1, Table 2). The length of individual dialogues ranges from 1:30 minutes to over 15 minutes, with median lengths for the different tasks ranging from 4 to 7 minutes. The vast majority of dialogues contain separate audio and video tracks (a few tracks are missing due

Task	Robot		Vhuman	Both
	No Err	Err	No Err	No Err
Desert Survival	21		19	
Lunar Survival	120	53	21	
Save the Art	102	52	121	40
Ice-breaker	96		21	

Table 1: Number of dialogues in the corpus

Experiment 1 (Artstein et al., 2017)	$N = 40$
Niki Desert; Julie Ice-breaker; Julie Lunar; Both Art	11
Niki Lunar; Julie Ice-breaker; Julie Desert; Both Art	10
Julie Desert; Niki Ice-breaker; Niki Lunar; Both Art	9
Julie Lunar; Niki Ice-breaker; Niki Desert; Both Art	10
Experiment 2 (Lucas et al., 2017)	$N = 101$
Niki: Lunar; Ice-breaker; Art	24
Niki: Lunar; (no ice-breaker); Art	25
Niki: Lunar; Ice-breaker; Art (errors)	26
Niki: Lunar; (no ice-breaker); Art (errors)	26
Experiment 3 (Lucas et al., in press)	$N = 53$
Niki: Lunar (errors); Ice-breaker; Art	27
Niki: Lunar (errors); (no ice-breaker); Art	26
Experiment 4 (Khooshabeh and Lucas, in press)	$N = 121$
Julie Art: Informational, Autonomous	29
Julie Art: Informational, Human-controlled	29
Julie Art: Normative, Autonomous	32
Julie Art: Normative, Human-controlled	31

Table 2: Number of participants in the various experimental conditions used for data collection

to equipment failure). A total of 160 dialogues have been transcribed to date (40 from each of Desert Survival, Lunar Survival, Save the Art, and Ice-breaker, all from the first experiment). When available, agents' time-stamped utterances were retrieved from the logs of the wizard interface, and participant utterances were transcribed manually between them. In some cases logs were not available, so both participant and agent utterances were transcribed manually.

The transcribed dialogues have been annotated for indicators of feedback by the human participant (Hee et al., 2017). These include the gestures of head shake, head nod, eyebrow raise, and laugh (Allwood et al., 1992); verbal feedback such as *mhm*, *uh-huh*, etc.; and the functions of understanding and attitudinal reactions of agreement and disagreement (Allwood et al., 2007). The annotations use a simple scheme that marks the temporal extents of the verbal and non-verbal actions, and separately marks the temporal extents of the understanding and agreement. Annotations were performed using the ELAN tool from the Max Planck Institute for Psycholinguistics (Brugman and Russel, 2004) (https://tla.mpi.nl/tools/tla-tools/elan/).

In addition to the dialogues, the corpus contains the participants' ranking of items before and after each ranking task interaction, as recorded by the experiment software (the software was not able to track progressive changes during the interaction), as well as self-reported rapport after each interaction and responses to some general questions (Artstein et al., 2017).

4. Usage

The corpus has been used to support a variety of research efforts. The participants' item rankings and self-reported rapport have been used to study how various factors affect social influence and rapport. Results show that building familiarity through dialogue increases social influence, and while people feel higher rapport with the robot than with the virtual human, they are influenced by both agents to a similar extent (Artstein et al., 2017). Conversational errors result in a loss of trust and consequent reduction in influence by the robot (Lucas et al., 2017), though the effect of errors depends on the timing on errors and interacts with the presence of social dialogue (Lucas et al., in press). Additional factors that affect social influence include the type of arguments given by the agent and participants' beliefs about the agent: informational arguments resulted in greater social influence than normative arguments, and informational arguments were more influential when participants believed the agent was autonomous rather than controlled by a person (described in Khooshabeh and Lucas, in press).

The annotated dialogues were used for studying multimodal grounding between humans and artificial agents. Results show that people display more feedback behavior when interacting with the robot than with the virtual human (perhaps paralleling the higher perceived rapport), and that substantially more feedback is displayed with either agent in the ice-breaker dialogue than in the ranking tasks (Hee et al., 2017).

We have begun using the corpus to bootstrap language understanding components for the development of autonomous versions of Niki and Julie; the eventual goal is to build autonomous agents that can engage in collaborative item-ranking tasks with humans. Evaluation of this effort remains for future work.

The procedure for collecting the data, as well as some software components relating to the wizard interface and control of the robot, have been shared with partner institutions for use in similar experiments.

5. Discussion and future work

The Niki corpus is a valuable resource for studying collaborative dialogue between humans and co-present humanoid robots and virtual agents. The corpus consists of speech and video data, and is partly transcribed and annotated. The corpus has been used in several completed and ongoing research projects. We are continuing the annotation efforts, and we hope to be able to make the corpus available to the research community.

6. Acknowledgments

The experiments used in collecting the corpus were funded in part by Honda Research Institute Japan Co., Ltd. We are

grateful to Mikio Nakano for his help in designing and running these studies. Thanks to Madeline Carlson, Cristian Cepeda, Anya Hee, Su Lei, Michael Sumida, Dylan Sutro, Selah Wright, and Zahin Ibne Anis for their annotation efforts. This work was supported in part by the U.S. Army; statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

7. Bibliographical References

Adalgeirsson, S. O. and Breazeal, C. (2010). Mebot: A robotic platform for socially embodied presence. In *Proc. HRI*, pages 15–22. IEEE Press.

Allwood, J., Nivre, J., and Ahlsén, E. (1992). On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9(1):1–26, January.

Allwood, J., Cerrato, L., Jokinen, K., Navarretta, C., and Paggio, P. (2007). The MUMIN coding scheme for the annotation of feedback, turn management and sequencing phenomena. *Language Resources and Evaluation*, 41(3–4):273–287, December.

Artstein, R., Leuski, A., Maio, H., Mor-Barak, T., Gordon, C., and Traum, D. (2015). How many utterances are needed to support time-offset interaction? In *Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference*, pages 144–149, Hollywood, Florida, May. AAAI Press.

Artstein, R., Traum, D., Boberg, J., Gainer, A., Gratch, J., Johnson, E., Leuski, A., and Nakano, M. (2017). Listen to my body: Does making friends help influence people? In *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference*, pages 430–435, Marco Island, Florida, May. AAAI Press.

Dahlbäck, N., Jönsson, A., and Ahrenberg, L. (1993). Wizard of Oz studies – why and how. *Knowledge-Based Systems*, 6(4):258–266, December.

DeVault, D., Artstein, R., Benn, G., Dey, T., Fast, E., Gainer, A., Georgila, K., Gratch, J., Hartholt, A., Lhommet, M., Lucas, G., Marsella, S., Morbini, F., Nazarian, A., Scherer, S., Stratou, G., Suri, A., Traum, D., Wood, R., Xu, Y., Rizzo, A., and Morency, L.-P. (2014). SimSensei kiosk: A virtual human interviewer for healthcare decision support. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'14)*, Paris.

Hee, E., Artstein, R., Lei, S., Cepeda, C., and Traum, D. (2017). Assessing differences in multimodal grounding with embodied and disembodied agents. In *5th European and 8th Nordic Symposium on Multimodal Communication*, Bielefeld, Germany, October.

Khooshabeh, P. and Lucas, G. (in press). Virtual human role players for studying social factors in organizational decision making. *Frontiers in Psychology*.

Khooshabeh, P., McCall, C., Gandhe, S., Gratch, J., and Blascovich, J. (2011). Does it matter if a computer jokes? In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 77–86. ACM.

Littlepage, G. E., Schmidt, G. W., Whisler, E. W., and Frost, A. G. (1995). An input-process-output analysis of influence and performance in problem-solving groups. *Journal of Personality and Social Psychology: Interpersonal Relations and Group Processes*, 69(5):877–889, November.

Lucas, G. M., Boberg, J., Traum, D., Artstein, R., Gratch, J., Gainer, A., Johnson, E., Leuski, A., and Nakano, M. (2017). The role of social dialogue and errors in robots. In *Proceedings of the 5th International Conference on Human-Agent Interaction*, pages 431–433, Bielefeld, Germany, October. ACM.

Lucas, G. M., Boberg, J., Traum, D., Artstein, R., Gratch, J., Gainer, A., Johnson, E., Leuski, A., and Nakano, M. (in press). Getting to know each other: The role of social dialogue in recovery from errors in social robots. In *HRI'18: Proceedings of the 2018 ACM/IEEE International Conference on Human Robot Interaction*, Chicago, March.

Marge, M., Bonial, C., Pollard, K. A., Artstein, R., Byrne, B., Hill, S. G., Voss, C., and Traum, D. (2016). Assessing agreement in human-robot dialogue strategies: A tale of two wizards. In David Traum, et al., editors, *Intelligent Virtual Agents: 16th International Conference, IVA 2016, Los Angeles, CA, USA, September 20–23, 2016 Proceedings*, volume 10011 of *Lecture Notes in Artificial Intelligence*, pages 484–488, Heidelberg, October. Springer.

8. Language Resource References

Brugman, H. and Russel, A. (2004). Annotating multimedia / multi-modal resources with ELAN. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 2065–2068, Lisbon, Portugal, May.

Constructing a Chinese Medical Conversation Corpus Annotated with Conversational Structures and Actions

Nan Wang† **Yan Song**‡ **Fei Xia**♮
†Sociology Department, University of California, Los Angeles, CA 90095, USA
‡Tencent AI Lab, Shenzhen, Guangdong Province, China
♮Linguistics Department, University of Washington, Seattle, WA 98104, USA
nwang3@ucla.edu, clksong@tencent.com, fxia@uw.edu

Abstract

Overuse of antibiotics and the attributed bacterial resistance is one of the most serious global public health crises today. Previous research reported that patients' advocacy for antibiotic treatment was consequential on antibiotic over-prescribing. To investigate how the advocacy and other factors contribute to antibiotic over-prescribing, qualitative and quantitative analysis of doctor-patient conversation can yield valuable findings. In this paper, we introduce AMed (**A**nnotated Corpus of **Med**ical Conversations), a manually transcribed corpus of medical dialogue in Chinese pediatric consultations, with annotation of conversational structures and actions. Based on the annotation, a significant association between patient request for antibiotic and antibiotic over-prescribing is discovered. As this corpus is the first with annotation of conversational structures and actions on medical consultation conversations in Chinese, it can be a valuable resource for discourse and dialogue research in general, and for the understanding of human collaboration and negotiation behavior in clinical consultations in particular. Furthermore, findings from analyses of the corpus can shed light on ways to improve physician-patient communication in order to reduce antibiotic over-prescribing.

Keywords: Conversation Analysis, physician-patient communication, antibiotics over-prescribing

1. Introduction

Overuse of antibiotics and the attributed bacterial resistance is one of the most serious global public health crises today. Multiple and rising levels of resistance are found worldwide; some of the common illnesses become fatal again as the old drugs become ineffective and the pipeline for developing new antibiotics become dry (Laxminarayan et al., 2013). The antibiotic overuse crisis is extremely severe in China. The country was the second largest consumer of antibiotics in the world during 2000-2010 (Boeckel et al., 2014) , and its bacterial resistance has reached alarming levels (Xiao et al., 2012). As a result, antibiotic overuse is an extremely severe and urgent problem in China. On average, 138g antibiotics were consumed per person a year in 2012 —ten times that consumed in the United States (Yuan, 2014). One main contributor to antibiotic overuse is the high prescription rate of antibiotics, which was over 50% in outpatient care according to a previous study (Wang et al., 2014). A study in 784 health institutions across the country found that antibiotics were prescribed twice as many as recommended by the WHO standard (Li et al., 2012). Overall, antibiotic prescription rates were over 50% in outpatient care; 75% of the patients were prescribed one antibiotic, and 25% were prescribed two or more antibiotics (Wang et al., 2014). With increasing global travel and exchange, antibiotic overuse in China is not only a domestic issue but also has a global impact.

Over-prescribing of antibiotics can be resulted from interaction-generated problems (e.g., misunderstanding in communication) that has little to do with sound medical judgment (Macfarlane et al., 1997). In a series of studies on doctor-parent interaction in the US pediatric setting, Stivers (Stivers, 2007) found that, besides overt advocacy, parent communication actions such as diagnosis resistance and treatment resistance are frequently perceived by physicians as expecting antibiotics, even when parents do not report actual expectations. This is consequential, as physicians are significantly more likely to prescribe inappropriately, when they perceive parental expectations for antibiotics (Mangione-Smith et al., 1999).

Compared to the US and the UK, research on medical conversation in Chinese clinical setting and its impact on antibiotic over-prescribing is limited, partly due to difficulty in obtaining conversational data in clinical settings, let alone a corpus where patient (or caregiver) communication actions and physician's prescribing decisions can be annotated and analyzed. To address this problem, we have built a corpus of naturally occurring conversations with rich annotations of conversational structures, physician-caregiver communication actions, as well as antibiotic prescribing outcomes. The results showed that caregivers' conversational actions such as advocacy for antibiotics indeed has a strong association with physicians' antibiotic prescriptions.

2. Methodology

As a theoretical framework and analytical method, we use Conversation Analysis (hence after, CA) to investigate the proposed research problem. CA is a method developed in sociology for research on human social interaction as a form of small-scale social institution (Sidnell et al., 2013). In this section, we provide a brief overview to CA and highlight how its main concepts are adopted in annotating conversational structures and actions in our corpus.

2.1. Conversational Structures

CA views conversation as a structural product of human social interaction with order at all points (Sacks et al., 1974). The main idea is that 'a course of action' can be thought of as built up out of basic adjacency pairs, which can be expanded in order to accomplish projected actions and activities (Schegloff, 1968). Conversational structures are analyzed at three levels (Schegloff, 2007):

- **Adjacency Pairs**: how two adjacent turns are organized as pairs, in which the 'first pair parts' (FPPs) project 'second pair parts' (SPPs) of a related type (e.g., question-answer, request-grant). Adjacency pair is the most pervasive and basic mode of organization in conversation (Schegloff, 2007).

- **Sequences**: how a sequence of turns is organized in a trajectory, through which courses of action are enacted coherently and orderly. For example, *'Are you free tonight?'* can be understood as preliminary to a base adjacency pair of invitation-acceptance. Sequences are the vehicle for getting activities accomplished.

- **Overall Organization**: how multiple, ordered sequences are organized to accomplish particular project (e.g., a medical project normally involves five ordered activities - problem presentation, information-gathering (history-taking and/or physical examination), diagnosis, treatment, and closing).

In our corpus, we annotate the first two levels with Adjacency Pair Part (APP) and Sequence Link (SL) as explained in Section 3.3., and leave the annotation of overall organization to future work.

2.2. Conversational Actions

CA also provides a systematic analytical method for analysis of participants' conversational actions. Actions in CA can be understood as the 'main job' that a turn is performing. To understand the primary action of a turn, analysts look at both its sequential position and grammatical composition (Schegloff, 2007). For instance, a turn (e.g., *'Isn't it raining?'*) can be understood as doing the action of answering to a question (e.g., *'Are we going to the game?'*) primarily because of its sequential position (i.e., as a second pair part in an adjacency pair).

Based on CA, we identify and annotate several types of physician-caregiver conversational actions such as caregivers' requesting actions for antibiotic treatment and physician's treatment recommendations, as explained in Section 3.4..

3. Corpus Construction

In this study, we created an **A**nnotated Corpus of **Med**ical Conversation (AMed) consisting of medical conversations with CA-style annotation.

3.1. Data Collection

To construct the corpus, we video-recorded 318 pediatric consultations from five hospitals in China between Sept. 2013 and Dec. 2013. Due to its pediatric setting, the conversations were mostly between physicians and patients' caregivers, where the caregivers could be parents, grandparents, or other relatives. We call each conversation (i.e., a video-recording of a complete medical consultation) a visit. Table 1 shows statistics of the raw data.

Ethical Consideration: All research procedures were reviewed and approved by the UCLA IRB (Ref# 13-000748).

Item	Number
# of Visits	318
# of Hospitals	5
# of Physicians	9
# of Patient (accompanied by caregivers)	318
Average length of a visit	4.9 minutes
Total length of the recordings	26 hours

Table 1: Statistics of the raw data

All identifiable information (e.g., person, institution, location names) has been removed from the corpus.

In the rest of the section, we provide details of our annotation scheme. Table 2 shows an example of annotation. Most annotation is at the turn level (all the columns in Table 2 except *Speech text*), but there is annotation inside the *Speech text* column (see Section 3.2.) and at the visit level (e.g., meta data for each visit, not shown here).

The meta data for each visit includes visit type (whether a visit is an acute visit or a follow-up visit) and unique identifiers (IDs) of the hospital, physicians, patients, and caregivers. The detail of hospitals (e.g., hospital type), physicians (e.g., gender and age range), caregivers (e.g., gender, relation to the patient, age range), and patients (e.g., gender) are stored in separate files.

3.2. Speech to Text

Following Jeffersonian transcribing system (Jefferson, 2004), we transcribe video-recordings of physician-caregiver conversation to capture both what is said and how it is said. The transcription is done in two passes: the first pass transcribes the Chinese verbatim of a turn, and the second pass transcribes speech production features (e.g., intonations and prolongations) and non-verbal activities of a turn (e.g., nodding, conducting a physical examination). Table 3 shows the list of transcribing symbols of the Jeffersonian convention. The punctuation marks in the text are transcribing symbols, and they are used to represent the speech production features.

The transcribing of the speech production features relies on annotators' judgment by comparing to the surrounding talk, e.g., a timed silence is annotated if there is a noticeable period of silence longer than a natural beat of silence. More specifically, a silence is transcribed as an individual turn if there is a recognizable completion point of the turn, both grammatically and pragmatically; while a short period of silence is transcribed as a pause within a turn, if there is a possible completion point at the Turn-Construction-Unit (TCU) boundary though the turn is recognizable as incomplete either grammatically or pragmatically. **Non-verbal activities** are transcribed when 1) there is a large chunk of silence AND the speakers are involved in the activity which may prevent them from talking; or 2) there is a nodding or head-shaking, which has pragmatic meaning but may be understood as lack of response if not transcribed.

The de-identification procedure is operationalized through wrapping the identifiable information (e.g. speaker name, institution name, region/area name, etc.) with the symbol '@'. This procedure enables deletion of the identifiable in-

TID	PR	APP	SL	Speech text	Action	Outcome
83	D	1	0	轻度 的 贫血. 哦, 你 先 给 他 吃 点 消炎药 吧,[好 吧, Mild anemia, ok? You give him some oral antibiotics to take, ok?	B2	
84	M	2	83	[家里 有 头孢. We have Cephalo at home.	C1	D1
85	D	1	84	行. (.) 退烧 药 呢? Ok. What about fever medicine?		
86	M	2	85	>退- 退烧 药 要 开 一 瓶.< Fever medicine, we ' ll need one bottle.		
87	D	1	86	哦. (.) 溃疡 的 呢? 喷 喉咙 的 药 有 吧,= Ok. What about ulcer medicine? You have the spray medicine for his throat?		
88	M	1	87	=那个: 利巴韦林 是 吧, That Ribavirin, right?		
89	D	2	88	嗯. Yeah.		
90	M	2	87	有 呢. Yes, we do.		

Table 2: An example of AMed annotation. TID: Turn ID; PR: Participant Role (M: Mother, D: Doctor); APP: Adjacency Pair Part; SL: Sequence Link; Action: Conversational Action; Outcome: Prescribing Outcome.

Symbol	Description
,	Continuing or slightly rising intonation.
.	Falling or terminal intonation.
?	Rising intonation.
=	Utterances are latched or ran together, with no gap of silence.
-	Preceding sound is cut off or interrupted.
[	Onset of overlapping talk, in pairs of adjacent turns.
:	Preceding sound is extended or stretched.
><	Talk with increased pace relative to surrounding talk.
<>	Talk with decreased pace relative to surrounding talk.
(*1.0*)	Silence measured in seconds, e.g., 1 second.
(.)	A micropause less than 0.2 second.
((*Cough*))	Nonverbal activities, e.g, Cough.
@N@	Representation of participants' identifiable information for confidentiality.

Table 3: The list of Jeffersonian transcribing symbols. The last row is a new symbol we added to replace identifiable information for the sake of privacy.

formation with automatic methods.

During the transcription, a conversation is broken into turns, and each turn is automatically assigned a **Turn ID**, which is a sequential number indicating the position of the turn in the conversation. Each turn is also given a **Participant Role** (RP), which is a label in a pre-defined label set, indicating the role of the speaker in this conversation (e.g., label *D* for doctor, and *M* for mother of the patient). The PR label is particularly informative when a conversation involves more than two participants.

Two annotators transcribed for each case and the transcripts were then verified by a third person (the first author).

3.3. Annotation of Conversational Structure

To annotate adjacency pairs and the sequences in a conversation (see Section 2.1.), we add two labels at each turn: Adjacency Pair Part (APP) and Sequence Link (SL).

APP has two possible values: '1' being FPPs, which initiate some exchange, and '2' being SPPs, which responds to some prior FPPs. For instance, in Table 2, the doctor's treatment recommendation (TID:83) is a FPP, which initiates a recommendation and projects an acceptance from the mother as a SPP (TID:84), and their APPs are 1 and 2, respectively.

Ideally, an adjacency pair consists of two adjacent turns; however, that is not always true in actual conversation. Here are some exceptions:[1]

- An adjacency pair can be incomplete, as the SPP of an adjacency pair is oriented as needed by speakers but sometimes not provided. A lack of or delayed production of a SPP may generate implications for understanding such as socially dispreferred actions (Pomerantz, 1984).

- An adjacency pair can be separated by various forms of sequence expansions, in order to deal with a wide range of contingencies in accomplishing the projected actions (Schegloff, 2007). For instance, in Table 2, the adjacency pair formed by Turn 87 and 90 are separated by another pair, Turn 88 and 89. Here, the second pair is dealing with understanding problems of the FPP of the first pair. We call the second pair an **insert expansion** of the first pair.

- A turn can be linked to multiple turns, when there are multiple speakers. For instance, when a doctor asks

[1]One complication that we will not get into detail here is that, within each turn, there could be multiple Turn-Construction-Units (TCUs) (Sacks et al., 1974); each of the TCUs can take different APP and SL values. Since our primary focus in this study concerns with conversational actions that are often performed at the turn level, the current APP/SL/Action annotation is at the turn level, but our annotation scheme can be easily extended to the TCU level if needed.

a question and the question is answered by two caregivers, this leads to an adjacency pair with one FPP and two SPP turns.

Given these exceptions, APP labels alone will not be sufficient to indicate adjacency pairs. To solve this problem, we give each turn another attribute called **Sequence Link (SL)**. If a turn is the SPP of an adjacency pair, its SL will be the Turn ID of the FPP of the pair. Recall that in Conversation Analysis (CA), a sequence is a cluster of turns that are organized to accomplish particular courses of action or activities. If a turn is the FPP of an adjacency pair, its SL will be the Turn ID of the closest left neighbor in the same sequence.

The SL value of the first turn in a sequence is zero, indicating the beginning of a sequence. For instance, in Table 2 the physician has just delivered a diagnosis (not shown in the excerpt), the SL of Turn 83 is annotated as 0, showing this turn starts a new sequence about treatment recommendations.

Together, APP and SL indicate the locations of adjacency pairs and the sequences in the conversation. A sequence consists of a base adjacency pair and all of its expansions, and the first turn in a sequence has value 0 for SL. Three most common sequence expansion forms (Schegloff, 2007) are **insert expansion** (mentioned above), **pre-expansion** and **post-expansion**.

- Pre-expansion. For example, a pre-invitation sequence consists of a Q-A pair preceding the projected base invitation pair.

  Pre-expansion A: *Are you free tonight?* B: *Yeah.*

  Base pair A: *Let's go to a movie!* B: *Ok.*

- Insert Expansion. For example, a request-grant/reject base sequence can be intervened by a repair adjacency pair..

  Base FPP A: *Could you pass me the salt?*

  Insert FPP B: *Huh?*

  Insert SPP A: *Could you pass me that table salt?*

  Base SPP A: *Sure!*

- Post-expansion. For example, the third turn in the conversation below is produced after the production of the base adjacency pair SPP, registering the speaker's understanding of the sequence closing.

  Base pair *A: Shall we call it a day?* B: *Ok.*

  Post-expansion A: *Alright.*

3.4. Conversational Actions Related to Antibiotic Treatment Negotiation

To investigate how physician-caregiver conversation affects antibiotic over-prescribing, we identify three types of conversational actions at the turn level:

Caregivers' request for antibiotics in the form of (A1) explicit requests for antibiotic treatment, (A2) statements of desires for antibiotic treatment, (A3) inquiries about antibiotic treatment, and (A4) evaluations of past treatment. The four requests differ in their overtness in turn design and degree of caregiver agency in instigating the request.

Physicians' treatment recommendation in the form of (B1) pronouncements, (B2) proposals, (B3) preference-inquiries, The three forms of treatment recommendations vary in physicians' deontic authority in proposing the treatment. Besides, we also add a flag to indicate whether the physician recommends antibiotics to the patient in this turn.

Response to treatment request/recommendation are either non-acceptance (C0) or acceptance (C1).

In addition to those action types, we also record whether a patient is prescribed with antibiotics in a visit, which is either yes (D1) or no (D0). We call it **prescribing outcomes**. We annotate the prescribing outcome at the turn level, although it stands for the physician's final prescribing outcome at the visit level.

3.5. Additional Annotation

The transcribed text is automatically segmented into words using an in-house CRF word segmenter trained on the Chinese Penn Treebank (Xia et al., 2000). A small subset of the corpus (about 25 visits) is also translated into English for users who might be interested in cross-lingual dialogue research. Note that the transcribing symbols annotated in the Chinese text layer are not carried over to the English translation as the word order in two languages can be quite different. In addition, as previously mentioned, we store meta data that specifies information such as hospital id, physician id, gender and age ranges of the physician and the caregivers.

For all the types of annotation discussed in this section, speech-to-text was done by two annotators for each visit and the transcripts were then verified by a third person (the first author). The inter-annotator agreement is 91%.[2] Other types of annotation were done by one person.

4. Results

We have completed the annotation of the corpus and conducted some preliminary studies on the association between participants' actions and the prescribing outcomes.

4.1. Corpus Statistics

Table 4 summarizes the statistics of the corpus. The corpus contains nearly 40K turns with 470K Chinese characters, which is considerably large in terms of manually annotated natural human conversations. On average, each visit has three participants (the physician might talk to more than one caregiver), and the turns form 63 adjacency pairs which are grouped into 29 sequences, suggesting that topic shift occurs frequently in an efficiency-driven medical context.

4.2. Physician-caregiver Conversation Actions and Prescribing Outcomes

To allow comparison with research conducted in similar settings and designs (Stivers, 2007), we restrict our statisti-

[2]The character error rate is 8.9% when treating one transcript as the reference.

Total # of Chinese characters	468,162
Total # of Chinese words	270,042
Total # of visits	318
Total # of turns	39,216
Total # of non-verbal turns	5,815
Avg. # of Chinese words per visit	849.19
Avg. # of turns per visit	123.32
Avg. # of non-verbal turns per visit	18.29
Avg. # of adjacency pairs per visit	63.28
Avg. # of sequences per visit	28.66

Table 4: Statistics of the annotated corpus.

Request Type	**# of Visits Observed**	**% of Visits Observed**
(A1) Explicit requests	10	5.35
(A2) Desire statements	14	7.49
(A3) Inquiries	50	26.74
(A4) Evaluations	26	13.90
Total	100	53.48

Table 5: Caregiver's requesting actions in the 187 acute visits. There could be more than one instance of requesting action in one visit.

cal analyses and report distribution of the conversation actions and prescribing outcomes to **acute visits**. Out of 318 visits, 187 are considered *acute visits*. Table 5 shows the distribution of caregivers' requesting actions in the corpus. The first columns is the numbers of visits that contain a caregiver's requesting action, and the second column is the percentage of visits that contain a requesting action.

Table 6 shows the distribution of prescribing outcomes of the acute visits with respect to whether the caregivers make antibiotics requests during the visit. Running χ^2 test on the data shows that caregiver's requesting action is highly correlated with prescribing outcome ($\chi^2 = 13.14, p < 0.001$). In addition, the association between caregiver's requesting actions and prescribing outcome is even more significant, if the caregiver's treatment advocacy is for the specific treatment modality of IV antibiotics ($\chi^2 = 21.08, p < 0.001$).

5. Discussion

This study makes contributions in three areas, as explained below.

5.1. A new scheme for annotating conversation

Our work explores a new annotation scheme for coding conversational structures and actions. For conversation structures, while the existing theories have recognized that utterances in conversation has higher-level forms of hierarchical structures (Grosz and Sidner, 1986; Carletta et al., 1997), most of the existing schemes have only implemented annotations of conversational structures at turn level or between a pair of turns (e.g., by distinguishing Forward Communicative Function and Backward Communicative Function (Core and Allen, 1997; Jurafsky et al., 1997)). Informed by the Conversation Analysis (CA) theory and approach, our annotation scheme allows us to capture hierarchical structures of conversation at several levels, including turn, adjacency pair, sequence, and overall organization.

For conversational actions, most of the existing annotation schemes of dialog acts (DAs) were based on speech act theory (Core and Allen, 1997; Jurafsky et al., 1997; Stolcke et al., 2000; Hoxha et al., 2016) ; however, it was found that comprehension of indirect speech acts were very difficulty with the available schemes, primarily due to the fact that classifications of actions were based on the surface format of an utterance. Based on CA theory of actions in conversation, which considers the sequential position of a turn as critical for action recognition and ascription, our annotation scheme allows classifications of actions based on a turn's structural position in conversation. It thus entails great flexibilities for annotating indirect conversational actions.

Prescriptions	**w/ Request**	**w/o Request**	**Total**
Antibiotics	72	28	100
Non-antibiotics	39	48	87
Total	100	87	187

Table 6: Distribution of prescribing outcomes by occurrence of caregiver's requesting action.

5.2. A new annotated corpus of conversations

As far as we know, AMed is the first corpus consisting of naturally occurring clinical conversations in Chinese which are manually transcribed and annotated with conversational structure and actions. Upon releasing, the corpus can be used not only for research of general purposes such as conversational understanding, modeling human social behavior of cooperation and coordination, but also for more specific purposes such as identifying risk factors for antibiotic over-prescribing.

5.3. Understanding the impact of caregivers' behavior and prescribing outcome

Antibiotic over-prescribing and bacterial resistance has been considered as one of the biggest global public health crises today. The problem is particularly severe in China. Most of the existing research argued that antibiotic over-prescribing in China was driven by financial incentives tied to physicians' prescriptions; thus a substantial amount of social resources flew to address the problem from the supply side. However, little is known whether caregivers' demands for antibiotic prescriptions play a role.

Our findings show that caregivers' requesting actions are significantly associated with physicians' antibiotic prescriptions. It presents empirical evidence to show that over-prescribing can also be resulted from caregiver advocacy for antibiotics in physician-patient/caregiver communication. In addition, our findings reveal that compared to caregivers in American pediatric context, the Chinese caregivers not only demand for antibiotic prescriptions, but also use more overt forms of advocating actions more frequently. Studies in similar setting in the U.S. showed that although the American caregivers also use similar actions to advocate for antibiotics in medical consultations, these

actions were only observed 9% of the time. Moreover, among the four types of advocating actions, the most overt form of requesting actions (i.e., explicit requests) were observed in 5.35% of the Chinese consultations; in comparison, this action was observed in less than 1% of the time in the American context (Stivers, 2007; Stivers, 2002). These findings thus have important implications for reducing antibiotic over-prescribing in China. Intervention measures such as providing physicians with trainings of communication skills to resist caregiver pressure are likely to produce desirable outcomes.

6. Conclusion and Future Work

In this paper, we present a manually transcribed and annotated corpus of medical conversation in Chinese clinical setting, aiming to contribute a high-quality language resource for research on treatment decision-making in clinical setting and antibiotic over-prescribing in pediatrics. The results proved our hypotheses that communication in medical consultations is significantly associated with medical decision outcomes such as antibiotic over-prescription. It thus provides basis on which effective intervention measures (e.g., public health education campaign, physician training of communication skills) can be devised in order to address the issue.

In addition, our proposed work provide an unique and valuable resource for research relevant to conversational understanding, treatment decision-making in doctor-patient communication, as well as antibiotic over-prescribing.

For future work, we will expand our annotation scheme to mark overall organization of the conversations, and test the usefulness of the scheme on conversations from other domains. For the AMed corpus, we plan to extend that by adding more data and also examine additional factors in clinical interaction that may lead to antibiotic over-prescribing.

7. Bibliographical References

Boeckel, V. T., Gandra, S., Ashok, A., Caudron, Q., Grenfell, B., Levin, S., and Laxminarayan, R. (2014). Global antibiotic consumption 2000 to 2010: an analysis of national pharmaceutical sales data. *Lancet Infectious Diseases*, 14(8):742–750.

Carletta, J., Isard, A., Isard, S., Kowtko, C. J., Doherty-Sneddon, G., and H. Anne, A. (1997). The Reliability of a Dialogue Structure Coding Scheme. *Computational Linguistics*, 23(1):13–31.

Core, G. M. and Allen, F. J. (1997). Coding Dialogs with the DAMSL Annotation Scheme. In *Proceedings of AAAI Fall Symposium on Communicative Action in Humans and Machines*.

Grosz, B. and Sidner, C. (1986). Attention, Intentions, and the Structure of discourse. *Computational Linguistics*, 12(3):175–204.

Hoxha, J., Chandar, P., He, Z., Cimino, J., Hanauer, D., and Weng, C. (2016). DREAM: Classification Scheme for Dialog Acts in Clinical Research Query Mediation. *Journal of Biomedical Informatics*, 59:89–101.

Jefferson, G. (2004). Glossary of transcript symbols with an introduction. In Gene H. Lerner, editor, *Conversation Analysis: Studies from the First Generation*, chapter 2, pages 13–31. John Benjamins, Amsterdam / Philadelphia.

Jurafsky, D., Shriberg, E., and Biasca, D. (1997). Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual, Draft 13. Technical report, University of Colorado, Boulder.

Laxminarayan, R., Duse, A., Wattal, C., Zaidi, A., Wertheim, H., Sumpradit, N., Vlieghe, E., Hara, G., Gould, I., Goossens, H., Greko, C., So, A., Bigdeli, M., Tomson, G., Woodhouse, W., Ombaka, E., Peralta, A., Qamar, F., Mir, F., Kariuki, S., Bhutta, Z., Coates, A., Bergstrom, R., Wright, G., Brown, E., and Cars, O. (2013). Global antibiotic consumption 2000 to 2010: an analysis of national pharmaceutical sales data. *Lancet Infectious Diseases*, 13(12):1057–1098.

Li, Y., Xu, J., Wang, F., Wang, B., Liu, L., Hou, W., Fan, H., Tong, Y., Zhang, J., and Lu, Z. (2012). Overprescribing in china, driven by financial incentives, results in very high use of antibiotics, injections, and corticosteroids. *Health Affairs (Project Hope)*, 31(5):1075–1082.

Macfarlane, J., Holmes, W., Macfarlane, R., and Britten, N. (1997). Influence of patients' expectations on antibiotic management of acute lower respiratory tract illness in general practice: questionnaire study. *BMJ*, 315(7117):1211–1214.

Mangione-Smith, R., McGlynn, E. A., Elliott, M. N., Krogstad, P., and Brook, R. H. (1999). The relationship between perceived parental expectations and pediatrician antimicrobial prescribing behavior. *Pediatrics*, 103(4):711–718.

Pomerantz, A. (1984). Agreeing and disagreeing with assessments: some features of preferred/dispreferred turn shapes. In J. Maxwell Atkinson, et al., editors, *Structures of Social Action: Studies in Conversation Analysis*, chapter 4, pages 57–101. Cambridge University Press, Cambridge, U.K.

Sacks, H., Schegloff, E. A., and Jefferson, G. (1974). A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4, Part 1):696–735.

Schegloff, E. A. (1968). Sequencing in conversational openings. *American Anthropologist*, 70(6):1075–1095.

Schegloff, E. (2007). *Sequence organization in interaction: Volume 1: A primer in conversation analysis*. Cambridge University Press.

Sidnell, J., Stivers, T., and Eds. (2013). *The Handbook of Conversation Analysis*. Wiley-Blackwell.

Stivers, T. (2002). Participating in decisions about treatment: Overt parent pressure for antibiotic medication in pediatric encounters. *Social Science & Medicine*, 54(7):1111–1130.

Stivers, T. (2007). *Prescribing under Pressure: Parent-physician Conversations and Antibiotics*. Oxford University Press, London.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V., and Meteer, M. (2000). Dialogue Act Modeling for

Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics*, 26(3):339–373.

Wang, J., Wang, P., Wang, X., Zheng, Y., and Xiao, Y. (2014). Use and prescription of antibiotics in primary health care settings in china. *JAMA Internal Medicine*, 174(12):1914–1920.

Xia, F., Palmer, M., Xue, N., Okurowski, M. E., Kovarik, J., Chiou, F., Huang, S., Kroch, T., and Marcus, M. (2000). Developing Guidelines and Ensuring Consistency for Chinese Text Annotation. In *Proceedings of the Second Language Resources and Evaluation Conference (LREC)*.

Xiao, Y., Shen, P., Wei, Z., Chen, Y., and Kong, H. (2012). Mohnarin report of 2011, monitoring of bacterial resistance in China. *Chinese Journal of Nosocomiology*, 22:4946–4952.

Yuan, S. (2014). China should reduce the overuse of intravenous infusion. *British Medical Journal*, 348:1262.

Predicting Nods by using Dialogue Acts in Dialogue

Ryo Ishii, Ryuichiro Higashinaka, Junji Tomita
NTT Media Intelligence Laboratories, NTT Corporation
1-1 Hikari-no-oka, Yokosuka-shi, Kanagawa, Japan
{ishii.ryo, higashinaka.ryuichiro, tomita.junji@lab.ntt.co.jp

Abstract

In addition to verbal behavior, nonverbal behavior is an important aspect for an embodied dialogue system to be able to conduct a smooth conversation with the user. Researchers have focused on automatically generating nonverbal behavior from speech and language information of dialogue systems. We propose a model to generate head nods accompanying utterance from natural language. To the best of our knowledge, previous studies generated nods from the final morphemes at the end of an utterance. In this study, we focused on dialog act information indicating the intention of an utterance and determined whether this information is effective for generating nods. First, we compiled a Japanese corpus of 24 dialogues including utterance and nod information. Next, using the corpus, we created a model that estimates whether a nod occurs during an utterance by using a morpheme at the end of a speech and dialog act. The results show that our estimation model incorporating dialog acts outperformed a model using morpheme information. The results suggest that dialog acts have the potential to be a strong predictor with which to generate nods automatically.

Keywords: nod, dialogue act, japanese dialogue

1. Introduction

Nonverbal behavior in human communication has important functions of transmitting emotions and intentions in addition to verbal behavior (BirdWhistell, 1970). This means that an embodied dialogue system should be able to express nonverbal behavior according to the utterance to communicate smoothly with the user (McBreen and Jack, 2001; Watanabe et al., 2003; Ishi et al., 2010). Against such a background, researchers have focused on constructing automatic generation models of nonverbal behavior from speech and linguistic information. Among nonverbal behaviors, nodding of the head is very important for emphasizing speech, giving and receiving speech authority, giving feedback, expressing conversational engagement, and intention of starting to speak (Senko Maynard, 1987; Senko Maynard, 1989; Ishii et al., 2015b; Ishii et al., 2017b; Ooko et al., 2011). It has been shown that nodding improves the naturalness of avatars and dialog systems and promotes conversation.

Nodding accompanying an utterance has the effect of strengthening the persuasive power of speech and making it easier for the conversational partner to understand the contents of the utterance (Lohse et al., 2014). Researchers have tried to generate nods during speaking from speech and natural language. In particular, they used several acoustic features, such as sound pressure and prosody, for generating nods (Yehia et al., 2002; KG et al., 2004; Graf et al., 2002; Busso et al., 2007; Beskow et al., 2006; Iwano et al., 1996; Ishi et al., 2010). However, it has been difficult to accurately generate nods at an appropriate time according to an utterance from only speech.

A few studies have tackled the problem of generating nods from natural language. These studies focused on the final morphemes in the phrase of an utterance and analyzed the co-occurrences with nods. They found that morphemes related to the interjections, feedback, and questionnaire and conjunctions appearing in turn-keeping (Ishi et al., 2006; Ishi et al., 2007) tend to co-occur with nods. On the basis of this information, a simple automatic nod-generation model was proposed (Ishi et al., 2010; Sakai et al., 2015). It was found that the behavior of humanoid robots and avatars that generated nods with the model gave a better impression of naturalness. It is thought that if a model that can generate nodding more accurately is constructed, it will lead to smoother communication between the dialog system and user. Therefore, a more accurate nod-generation model should be constructed by clarifying the relevance of more detailed language information and nodding. It is also known that the relevance of a speech feature to nodding and vice versa depends on the language; for instance this is weaker in Japanese (Yehia et al., 2002; Ishi et al., 2007). A detailed examination of a nod-generation model using language information is thus considered important.

In this research, we constructed a highly accurate head-nod-generation model using natural Japanese language by focusing on the dialogue acts of utterances, which has not been investigated. Dialogue acts are information indicating the intention of the speaker throughout the utterance, and it is considered that the occurrences of nods change according to the intention. Therefore, in contrast to the features of the morphemes in the final phrase that were taken into account in previous studies (Ishi et al., 2006; Ishi et al., 2007), the dialogue acts we handle include the information intended by the speaker throughout the utterance. We constructed our nod-generation model using dialogue act information of utterances and determined whether dialogue acts are useful for generating nods.

We collected a corpus consisting of 24 Japanese dialogues including utterances and head-nod information. Next, we used the corpus to create our model that estimates whether a nod occurs during an utterance by using the morpheme at the end of the speech and dialogue act. In an experiment, we found that our estimation model using dialogue act information outperformed that using morpheme information alone. We also found that a model using both dialog act information and morpheme information did not perform any



Figure 1: Photograph of two participants having dialogue

better than the model using only dialogue acts. We thus concluded that dialog acts have the potential to be a strong predictor that can be used to generate nods automatically.

2. Corpus

To collect a Japanese conversation corpus including verbal and nonverbal behaviors for generating nods in dialogue, we recorded 24 face-to-face two-person conversations (12 groups of two different people). The participants were Japanese males and females in their 20s to 50s who had never met before. They sat facing each other (Fig. 1). To gather more data on nodding accompanying utterances, we adopted the explanation of an animation participants have not seen as the conversational content. Before the dialogue, they watched a famous popular cartoon animation called "'Tom & Jerry" in which the characters do not speak. In each dialogue, one participant explained the content of the animation to the conversational partner within ten minutes. At any time during this period, the partner could freely ask questions about the content.

We recorded the participants' voices with a pin microphone attached to the chest and videoed the entire discussion. We also took bust (chest, shoulders, and head) shots of each participant (recorded at 30 Hz). In each dialogue, the data on the utterances and nodding behaviors of the person explaining the animation were collected in the first half of the ten-minute period (120 minutes in total) as follows.

- Utterances: We built an utterance unit using the inter-pausal unit (IPU) (Koiso et al., 1998). The IPU was used in the same manner as in previous studies dealing with morpheme information (Ishi et al., 2006; Ishi et al., 2007). The utterance interval was manually extracted from the speech wave. A portion of an utterance followed by 200 ms of silence was used as the unit of one utterance. We collected 2965 IPUs in total.
- Head nod: A head nod is a gesture in which the head is tilted in alternating up and down arcs along the sagittal plane. A skilled annotator annotated the nods by using bust/head and overhead views in each frame of the videos. We regarded nodding continuously in time as one nod event.
- Gaze: The participants wore a glass-type eye tracker (Tobii Glass2). The gaze target of the participants and the pupil diameter were measured at 30 Hz.

Label	Dialogue Act	Label	Dialogue Act
DA0	Greeting	DA15	Question (habit)
DA1	Provision	DA16	Question (desire)
DA2	Self-disclosure (fact)	DA17	Question (plan)
DA3	Self-disclosure (experience)	DA18	Question (evaluation)
DA4	Self-disclosure (habit)	DA19	Question (other)
DA5	Self-disclosure (positive preference)	DA20	Question (Yourself)
DA6	Self-disclosure (negative preference)	DA21	Sympathy
DA7	Self-disclosure (neutral preference)	DA22	Non-sympathy
DA8	Self-disclosure (desire)	DA23	Confirmation
DA9	Self-disclosure (plan)	DA24	Proposal
DA10	Self-disclosure (other)	DA25	Repeat
DA11	Acknowledgment	DA26	Paraphrase
DA12	Question (information)	DA27	Approval
DA13	Question (fact)	DA28	Thanks
DA14	Question (experience)	DA29	Apology
		DA30	Filler
		DA31	Admiration
		DA32	Other

Table 1: Dialogue act labels

- Hand gesture and body posture: The participants' body movements, such as hand gestures, upper body, and leg movements, were measured with a motion capture device (Xsens MVN) at 240 Hz.

All verbal and nonverbal behavior data were integrated at 30 Hz for display using the ELAN viewer (Wittenburg et al., 2006). This viewer enabled us to annotate the multimodal data frame-by-frame and observe the data intuitively. In this research, we only handled utterance and head-nod data in the corpus we constructed. Nods occurred in 1601 out of the 2965 IPUs.

3. Head-Nod-Generation Model

The goal of our research was to demonstrate that the dialogue act of an utterance is useful for generating nods. We evaluated our proposed model for estimating nods from dialogue acts and the previously constructed estimation model using the final morphemes at the end of utterance (Ishi et al., 2006; Ishi et al., 2007). As a primitive feature value, we created a feature value related to the length of an utterance and decided to treat the estimation model using this feature value as a baseline We constructed another estimation model using all information including dialogue act, final morphemes at the end of utterance, and length of utterance to evaluate the effectiveness of fusion (All model). The feature values for each IPU were as follows.

- Length of utterance (LU): Number of characters in an IPU. The feature value is a one-dimensional vector.
- Final morphemes (FM): This is binary feature as to whether morpheme injunctions related to feed-

Used Feature values	Precision	Recall	F-score
Chance level	0.500	0.500	0.500
LU	0.410	0.640	0.500
FM	0.412	0.646	0.521
DA	0.662	0.670	**0.666**
LU+FM	0.423	0.651	0.513
LU+**DA**	0.666	0.672	**0.669**
FM+**DA**	0.662	0.670	**0.666**
LU+FM+**DA**	0.666	0.672	**0.669**

Table 2: Partial excerpt of attribute weight rank in 19 features in All model.

back (e.g., "'en', "ee", "'aa", "hi", etc.) and particles related to questioning and turn-keeping (e.g., "de", "kara", "kedo", "'kana", "janai", etc.) co-occurring with the nod (as indicated in the previous studies (Ishi et al., 2006; Ishi et al., 2007) are included in the last morpheme of the IPU. Morphological analysis was conducted for this purpose. We used J-tag (Fuchi and Takagi, 1998), a general morphological analysis tool for Japanese. Based on the output result, it was judged whether the last morpheme is related to nods. The feature value is a one-dimensional vector.

- Dialogue act (DA): The dialogue act was extracted using an estimation technique for Japanese (Meguro et al., 2010; Higashinaka et al., 2014). The technique can estimate the dialogue act using the word N-grams, semantic categories (obtained from a Japanese thesaurus Goi-Taikei), and character N-grams. The dialog acts and number of IPUs are listed in Table 1. There was very little data for DA4, 10, 12, 13, 14-19, 22, 24, and 26-29, so they were excluded from feature values. The label of each dialogue act was expressed as a binary value as to whether the dialogue act appeared. Therefore, the feature value was a 17-dimensional vector.

We constructed the estimation models by using SMOreg (Keerthi et al., 2001), which implements a support vector machine (SVM) in Weka (Bouckaert et al., 2010) and evaluated the accuracy of the models and the effectiveness of each feature. The settings of the SVM — the polynomial kernel, C (cost parameter), and γ (hyper parameter of the kernel)—were determined using a grid search technique. The class was a binary value as to whether a nod occurred. We used 24-fold cross validation using a leave-one-person-out technique with the data for the 24 participants. We evaluated how well a participant's nods could be estimated with an estimator generated only from data of other people. As shown in Table 1, DA (dialogue act) had the highest performance among the models using only LU, FM, or DA, with an F-score of 0.666 ($t(23) = -9.46$, $p < .01$ in LU vs. DA; $t(23) = -9.46$, $p < .01$ in FM vs. DA). In addition, no differences in performance were found between any of the models in which other features were added to DA and that using only DA. These results suggest that DA is more useful for nod estimation than morpheme information at the end of speech. The results also suggest that adding morpheme information to DA does not lead to any performance improvement.

Rank	Feature	Attribute weight
1	DA3: Self-disclosure (experience)	-1.1044
2	DA20: Question (self)	1.0430
3	DA0: Greeting	1.0223
4	DA32: Confirmation	1.0217
5	DA30: Filler	-0.9358
6	DA25: Repeat	0.9255
7	LC (Length of utterance)	0.9123
8	DA23: Confirmation	0.9628
9	DA11: Acknowledgment	-0.9589
10	DA31: Admiration	-0.9583
11	DA21: Sympathy	0.9228
12	DA5: Self-disclosure (positive preference)	0.8653
13	DA6: Self-disclosure (negative preference)	0.8646
14	DA1: Provision	0.8458
15	DA9: Self-disclosure (plan)	0.8266
16	DA2: Self-disclosure (fact)	0.7872
17	FM (Final morphemes)	0.0330
18	DA7: Self-disclosure (neutral preference)	0.0000
19	DA3: Self-disclosure (experience)	0.0000
20	DA32: Other	0.0000
21	DA0: Greeting	0.0000
22	DA24: Proposal	0.0000
23	DA8: Self-disclosure (desire)	0.0000
24	DA13: Question (fact)	0.0000

Table 3: Excerpt of attribute weight ranking of 24 features in All model.

4. Discussion

The experimental results suggest that dialogue act information is most useful for estimating nods. Many of the spoken words in the IPUs used in this study are considerably collapsed compared with the written language, and it is considered that the accuracy of estimation using dialogue acts is not high. Nonetheless, the estimation model using dialogue acts performed well, so dialogue acts seem to have potential for nod estimation. The attribute weight of each feature value in the All model was calculated. The results are shown in Table 3 and suggest that DA20 (Question (self)), DA0 (Greeting), DA32 (Confirmation), DA25 (Repeat), DA23 (Confirmation), DA21 (Sympathy), DA5 (Self-disclosure (positive preference)), DA6 (Self-disclosure (negative preference)), DA1 (Provision), DA9 (Self-disclosure (plan)), and DA2 (Self-disclosure (fact)) contribute to accurate estimation of when a nod will occur. It can also be seen that DA3 (Self-disclosure (experience)), DA30 (Filler), DA11 (Acknowledgment), and DA31 (Admiration) contribute to estimating when a nod will not occur. The evaluation also showed that FM, which was used in previous research, is not very effective. The previous study focused on free dialogue, whereas in this research, one of the participants was instructed to explain something to the other party; hence, the results are dependent on the content of the dialogue. It would be interesting to see whether the effectiveness of feature values changes

depending on the dialogue scene.
In this research, we used an IPU as a unit of utterance and tried to determine whether nodding occurs in IPUs. We did not consider the detailed timing of occurrences or the number of nods in an utterance. We plan to focus on linguistic information other than dialogue acts and clarify co-occurrence relations with nods. Furthermore, we would like to work on constructing a model that can generate the occurrence timing and occurrence frequency within an utterance. The results of this study suggest that it is also possible that dialogue acts may be effective for nonverbal behaviors other than nods.

5. Conclusion

We constructed a highly accurate head-nod-generation model using natural Japanese language. In this research, we focused on using dialogue acts, which indicate the intention of utterances, for estimating nodding behavior accompanying utterance and demonstrated that they are effective information for generating nods. In an experiment, we found that our estimation model using dialogue acts outperformed those using morpheme information alone. We also found that a model using both dialogue act information and morpheme information showed no difference in performance from that using only dialogue acts. These results suggest that dialog act information has the potential to be a strong predictor that can be used to generate nods automatically.
In the future, we will focus on linguistic information other than dialogue acts and demonstrate the co-occurrence relation with nodding accompanying an utterance. Furthermore, we plan to construct a model for generating the occurrence timing of nods within an utterance and a model for generating other nonverbal behaviors such as gaze, which is important for turn management (Ishii et al., 2013b; Ishii et al., 2014; Ishii et al., 2015a; Ishii et al., 2015b; Ishii et al., 2016a; Ishii et al., 2016b; Ishii et al., 2017a) and expression of conversational engagement (Ishii et al., 2006; Ishii and Nakano, 2008; Ishii and Nakano, 2010; Nakano and Ishii, 2010; Ishii et al., 2011; Ishii et al., 2013a) and body posture.

6. Bibliographical References

Beskow, J., Granstrom, B., and House, D. (2006). Visual correlates to prominence in several expressive modes. In *INTERSPEECH*.

BirdWhistell, R. L. (1970). *Kinesics and context*. University of Pennsylvania Press.

Bouckaert, R. R., Frank, E., Hall, M. A., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2010). WEKA–experiences with a java open-source project. *J. Machine Learning Research*, 11:2533–2541.

Busso, C., Deng, Z., Grimm, M., Neumann, U., and Narayanan, S. (2007). Rigid head motion in expressive speech animation: Analysis and synthesis. In *IEEE Transactions on Audio, Speech, and Language Processing*, pages 1075–1086.

Fuchi, T. and Takagi, S. (1998). Japanese morphological analyzer using word cooccurrence -jtag. In *International conference on Computational linguistics*, pages 409–413.

Graf, H. P., Cosatto, E., Strom, V., and Huang, F. J. (2002). Visual prosody: Facial movements accompanying speech. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 381–386.

Higashinaka, R., Imamura, K., Meguro, T., Miyazaki, C., Kobayashi, N., Sugiyama, H., Hirano, T., Makino, T., and Matsuo, Y. (2014). Towards an open-domain conversational system fully based on natural language processing. In *International conference on Computational linguistics*, pages 928–939.

Ishi, C. T., Ishiguro, H., and Hagita, N. (2006). Analysis of prosodic and linguistic cues of phrase finals for turn-taking and dialog acts. In *International Conference of Speech and Language*, pages 2006–2009.

Ishi, C. T., Haas, J., Wilbers, F. P., Ishiguro, H., and Hagita, N. (2007). Analysis of head motions and speech, and head motion control in an android. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 548–553.

Ishi, C. T., Ishiguro, H., and Hagita, N. (2010). Head motion during dialogue speech and nod timing control in humanoid robots. In *ACM/IEEE International Conference on Human-Robot Interaction*, pages 293–300.

Ishii, R. and Nakano, Y. (2008). Estimating user ' s conversational engagement based on gaze behaviors, 09.

Ishii, R. and Nakano, Y. I. (2010). An empirical study of eye-gaze behaviors: Towards the estimation of conversational engagement in human-agent communication. In *Proceedings of the 2010 Workshop on Eye Gaze in Intelligent Human Machine Interaction*, EGIHMI '10, pages 33–40, New York, NY, USA. ACM.

Ishii, R., Miyajima, T., Fujita, K., and Nakano, Y. I. (2006). Avatar's gaze control to facilitate conversational turn-taking in virtual-space multi-user voice chat system. In *Intelligent Virtual Agents, 6th International Conference, IVA 2006, Marina Del Rey, CA, USA, August 21-23, 2006, Proceedings*, page 458.

Ishii, R., Shinohara, Y., Nakano, Y., and Nishida, T. (2011). Combining multiple types of eye-gaze information to predict user's conversational engagement. 02.

Ishii, R., Nakano, Y. I., and Nishida, T. (2013a). Gaze awareness in conversational agents: Estimating a user's conversational engagement from eye gaze. *ACM Trans. Interact. Intell. Syst.*, 3(2):11:1–11:25, August.

Ishii, R., Otsuka, K., Kumano, S., Matsuda, M., and Yamato, J. (2013b). Predicting next speaker and timing from gaze transition patterns in multi-party meetings. In *Proceedings of the International Conference on Multimodal Interaction*, pages 79–86.

Ishii, R., Otsuka, K., Kumano, S., and Yamato, J. (2014). Analysis and modeling of next speaking start timing based on gaze behavior in multi-party meetings. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 694–698.

Ishii, R., Kumano, S., and Otsuka, K. (2015a). Multimodal fusion using respiration and gaze behavior for predicting next speaker in multi-party meetings. In *ICMI*, pages 99–106.

Ishii, R., Kumano, S., and Otsuka, K. (2015b). Predicting

next speaker using head movement in multi-party meetings. In *ICASSP*, pages 2319–2323.

Ishii, R., Otsuka, K., Kumano, S., and Yamamoto, J. (2016a). Predicting of who will be the next speaker and when using gaze behavior in multiparty meetings. *The ACM Transactions on Interactive Intelligent Systems*, 6(1):4.

Ishii, R., Otsuka, K., Kumano, S., and Yamamoto, J. (2016b). Using respiration to predict who will speak next and when in multiparty meetings. *The ACM Transactions on Interactive Intelligent Systems*, 6(2):20.

Ishii, R., Kumano, S., and Otsuka, K. (2017a). Analyzing gaze behavior during turn-taking for estimating empathy skill level. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, ICMI 2017, pages 365–373, New York, NY, USA. ACM.

Ishii, R., Kumano, S., and Otsuka, K. (2017b). Prediction of next-utterance timing using head movement in multi-party meetings. In *Proceedings of the 5th International Conference on Human Agent Interaction*, HAI '17, pages 181–187, New York, NY, USA. ACM.

Iwano, Y., Kageyama, S., Morikawa, E., Nakazato, S., and Shirai, K. (1996). Analysis of head movements and its role in spoken dialogue. In *International Conference on spoken language*, pages 2167–2170.

Keerthi, S. S., Shevade, S. K., Bhattacharyya, C., and Murthy, K. R. K. (2001). Improvements to platt's smo algorithm for svm classifier design. *Neural Computation*, 13(3):637–649.

KG, M., JA, J., DE, C., T, K., and E, V.-B. (2004). Visual prosody and speech intelligibility: head movement improves auditory speech perception. 15(2):133–7.

Koiso, H., Horiuchi, Y., Tutiya, S., Ichikawa, A., and Den, Y. (1998). An analysis of turn-taking and backchannels based on prosodic and syntactic features in japanese map task dialogs. In *Language and Speech*, volume 41, pages 295–321.

Lohse, M., Rothuis, R., Gallego-Pérez, J., Karreman, D. E., and Evers, V. (2014). Robot gestures make difficult tasks easier: The impact of gestures on perceived workload and task performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1459–1466, New York, NY, USA. ACM.

McBreen, H. and Jack, M. (2001). Evaluating humanoid synthetic agents in e-retail applications. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 31:5.

Meguro, T., Higashinaka, R., Minami, Y., and Dohsaka, K. (2010). Controlling listening-oriented dialogue using partially observable markov decision processes. In *International conference on computational linguistics*, pages 761–769.

Nakano, Y. I. and Ishii, R. (2010). Estimating user's engagement from eye-gaze behaviors in human-agent conversations. In *Proceedings of the 15th International Conference on Intelligent User Interfaces*, IUI '10, pages 139–148, New York, NY, USA. ACM.

Ooko, R., Ishii, R., and Nakano, Y. I. (2011). Estimating a user's conversational engagement based on head pose information. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents*, IVA'11, pages 262–268, Berlin, Heidelberg. Springer-Verlag.

Sakai, K., Ishi, C. T., Minato, T., and Ishiguro, H. (2015). Online speech-driven head motion generating system and evaluation on a tele-operated robot. In *IEEE International Symposium on Robot and Human Interactive Communication*, pages 529–534.

Senko Maynard. (1987). Interactional functions of a nonverbal sign: Head movement in japanese dyadic casual conversation. *Journal of Pragmatics*, 11:589–606.

Senko Maynard. (1989). Japanese conversation: Self-contextualization through structure and interactional management. *Norwood, New Jersey: Ablex Publishing Corporation*.

Watanabe, T., Danbara, R., and Okubo, M. (2003). Effects of a speech-driven embodied interactive actor interactor on talker's speech characteristics. In *IEEE International Workshop on Robot-Human Interactive Communication*, pages 211–216.

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). Elan a professional framework for multimodality research. In *International Conference on Language Resources and Evaluation*.

Yehia, H. C., Kuratate, T., and Vatikiotis-Bateson, E. (2002). Linking facial animation, head motion and speech acoustics. 30(3):555–568.

Modeling Collaborative Multimodal Behavior in Group Dialogues: The MULTISIMO Corpus

Maria Koutsombogera, Carl Vogel
Trinity College Dublin
College Green, Dublin 2, Ireland
{koutsomm, vogel}@cs.tcd.ie

Abstract

We present a multimodal corpus that has been recently developed within the MULTISIMO project and targets the investigation and modeling of collaborative aspects of multimodal behavior in groups that perform simple tasks. The corpus consists of a set of human-human interactions recorded in multiple modalities. In each interactive session two participants collaborate with each other to solve a quiz while assisted by a facilitator. The corpus has been transcribed and annotated with information related to verbal and non-verbal signals. A set of additional annotation and processing tasks are currently in progress. The corpus includes survey materials, i.e. personality tests and experience assessment questionnaires filled in by all participants. This dataset addresses multiparty collaborative interactions and aims at providing tools for measuring collaboration and task success based on the integration of the related multimodal information and the personality traits of the participants, but also at modeling the multimodal strategies that members of a group employ to discuss and collaborate with each other. The corpus is designed for public release.

Keywords: multimodal corpus, multiparty interaction, collaboration, task success, social behavior, personality

1. Introduction

The development of multimodal corpora is an essential step in the investigation of the human behavior, as the knowledge reflected in them can lead to models of social behavior to be integrated in intelligent interfaces, making the human-computer interaction as effective, natural and intuitive as possible. Multimodal corpora provide information about the way different expressive modalities shape the structure of the interaction (i.e. turn management) and convey the speakers' cognitive and affective state in any given moment (including feedback responses and emotions), thus demonstrating the speakers' interactional and social behavior (Vinciarelli et al., 2015; Esposito et al., 2015). While two-party interaction is already a rich and informative setup, multiparty interaction is even more challenging because of the dynamics developed among group members (Gatica-Perez et al., 2017).

In recent years, various multimodal and multiparty corpora have been created to study and analyse human behavior and collaboration aspects of it, in two-party (HCRC Map Task Corpus (Thompson et al., 1993)) and in group interaction settings, such as the AMI corpus (Carletta et al., 2005), the Mission Survival Corpus-2 (Mana et al., 2007), the Canal 9 political debates satabase (Vinciarelli et al., 2009) and the Idiap Wolf Corpus (Hung and Chittaranjan, 2010), to name a few. The importance of analyzing such corpora lies in decoding communicative patterns involving verbal and non-verbal modalities. By definition, group dialogue is a canvas where different communicative intentions, personalities, lexical choices that may affect the outcome and the effectiveness of the interaction are manifested by the participants. In terms of behavior modelling, efforts focus on automatically analyzing various facets of group interactions and collecting this knowledge to improve the quality of the interaction either in human-human or in human-machine settings. Related work that exploits group dialogue and multiparty corpora studies dominance and leadership (Jayagopi et al., 2009; Nakano and Fukuhara, 2012) personality traits (Mohammadi and Vinciarelli, 2012), deception and competition (Hung and Chittaranjan, 2010).

In this work we discuss the design and implementation of the MULTISIMO corpus, a multimodal corpus consisting of collaborative group interactions where two players need to provide answers to a quiz and are guided by a facilitator. Participants work together while the facilitator monitors their progress and provides feedback and hints when needed. In this setup, collaboration refers to the process where the two players coordinate their actions to achieve their shared goal, i.e. find the appropriate answers and rank them. Collaboration will be measured based on communicative features that may show equal participation, for example the number of turns that each player produces, the number of times persons address their co-players, the mutual turn exchanges, the mutual gazes, etc.; or collaborative turn organization, e.g. in the cases where the facilitator allocates turns to each of the players, or when players address their co-participant to discuss different options instead of providing themselves a direct answer.

Though the development of multimodal and multiparty corpora is not a new domain, this corpus serves to fill in the gap in the investigation of the factors that influence collaboration and task success in a three-party setting and in providing tools for measuring group success. Specifically, the corpus will be exploited to investigate the factors that contribute to collaborative behaviors, e.g. the impact of personality on the participants' behaviors; the impact of dominant participants on the quality of collaboration; participants' attitudes; description and justification of the turn managing practices of the facilitator; multimodal strategies that the group participants employ to collaborate with each other; gender, nationality, language nativeness and familiarity correlations with collaborative behavior. Also, the automatic measurements and manual annotations on the corpus form

materials that can be viewed as the individual speakers profiling and can thus be investigated and revisited at various levels, i.e. linguistic, acoustic, visual, psychological and sociolinguistic.

Our goal is to investigate the way people discuss and collaborate with each other in order to complete a task. We seek to improve the understanding of the structure of collaborative interactions as well as to interpret the behavior of the participants. The corpus will serve as the knowledge base for identifying measurable behavioral and psychological variables of group members with the goal of creating behavioral models. These models may be exploited in human-computer interfaces, and specifically in the design of embodied conversational agents, i.e. agents that need to be able to extract information about their interlocutors to increase the intuitiveness and naturalness of the interaction. The expectations of a user interacting with such interfaces would be that the interaction is ideally as natural as possible. Specifically, users expect that the interface can understand who's talking, recognize signals of human behavior, understand them, coordinate the discussion and address humans in a natural form and at the correct time, as a human would do, facilitating overall collaboration on a common problem (Bohus and Horvitz, 2010; Al Moubayed et al., 2012).

The aforementioned challenges can be pursued if appropriate corpora are available to develop and train models. In this context, the design and implementation of a multimodal, as naturalistic as possible, data collection involving humans engaged in interaction tasks become meaningful in that they help us clarify and answer our research questions. In the next sections we present the experimental design and the data acquisition setup, the data collection, and ongoing work on the corpus analysis.

2. Scenario and Experimental Setup

The scenario was designed in a way that would elicit the desired behavior from the participants, that is, encourage their collaboration towards a goal. We thus designed sessions, in which 3 members of a group, 2 players and 1 facilitator, collaborate with each other to solve a quiz. The sessions were carried out in English and the task of the players was to discuss with each other, provide the 3 most popular answers to each of 3 questions (based on survey questions posed to a sample of 100 people), and rank their answers from the most to the least popular. Participants expressed and exchanged their personal opinions when discussing the answers, and they announced the facilitator the ranking once they reached a mutual decision.

The questions were taken from a database related to the Family Feud game.[1] No specific knowledge nor terminology was required to answer the specific questions; instead, the questions were selected so that they would be easy to address for both native and non-native English speakers, trigger the discussion among the players, and also elicit multimodal behavior, such as performing gestures when describing an object or an idea.[2] After the end of each session the participants filled in a brief questionnaire to assess their impression of the experiment.

Participants were assisted by the facilitator who coordinated this discussion, i.e. provided the instructions of the game and confirmed participants' answers, but also helped participants throughout the session and encouraged them to collaborate. The facilitator role is of key importance in the setup design, considering that it is a role that would be modeled for an embodied conversational agent that would coordinate group interaction and would help participants achieve their goals. In this respect, the facilitator role was designed in a way that will enable the extraction of behavioral cues for the development of an agent responsible for managing the interaction and choosing actions that maximize the collaboration effort and the performance of the group participants.

Participants' recruitment was announced on the online and wall notice boards at Trinity College Dublin (TCD) and via mailing lists. In total, 49 participants were recruited, the majority of them being students or researchers at TCD. 46 were assigned the role of players and were paired in 23 groups. The remaining 3 participants shared the role of the facilitator throughout the 23 sessions. Facilitators were selected in advance and were briefly trained before the actual recordings, i.e. they were given the quiz questions and answers and they were instructed to monitor the flow of the discussion and, if necessary, intervene to help players or to balance their participation. For this role we looked for participants who were teaching or tutoring professionals, who had a pedagogical training and were familiar with instructing and guiding groups of people in completing a task, properties that are desirable for a facilitator to have. To ensure consistency, the facilitators were of the same gender, nationality, professional background and level of English language competence (i.e. female, Greek, English teachers).

2.1. Technical Setup

The recording of the sessions took place in the premises of the School of Computer Science and Statistics (SCSS) in a quiet, though not sound-proof, room. The participants were seated around a table. The equipment used includes three HD cameras, one 360 camera, three head-mounted microphones, one omnidirectional microphone and one Kinect 2 sensor. Two of the HD cameras (1920x1080 px, 29.97 fps) were placed opposite each of the two players capturing their front view (Figure 1). The third HD camera (1920x1080 px, 25 fps) was placed opposite the facilitator and captures the whole scene (Figure 2); its zoomed angle is used to isolate the facilitator's front view. The 360 camera (3840x2160 px, 29.97 fps) was placed in the middle of the table to capture the whole scene from a low angle (Figure 3). The head-mounted microphones were recording the individual audio signals (SR 44.1 kHz), while the omnidirectional microphone was used as a backup audio

[1] `http://familyfeudfriends.arjdesigns.com//`, last accessed 15.02.2018

[2] The 3 questions were: (a) *name a public place where it's likely to catch a cold or a flu bug*; (b) *name 3 instruments you can find in a symphony orchestra*; and (c) *name something that people cut*.

source (SR 44.1 kHz). Finally, the Kinect 2 was placed in a way that it would perform the skeletal tracking of all participants (Figure 4). The experiment, together with two surveys that participants filled in, i.e. a personality test (cf. section 2.3) and an experience assessment questionnaire, lasted about 40 minutes per participant and each participant was rewarded with a 10 euro token.



Figure 1: Front views of participants.



Figure 2: The setup of a corpus session.



Figure 3: 360 camera angle of a group session.

2.2. Ethics Management

Since the data collection involves humans, the experimental process leading to it was supervised by the SCSS ethics committee and followed its ethical standards. Moreover, our aim was that the corpus, i.e. the recorded data together with the survey materials (personality tests and experience assessment questionnaires) will be made open for reuse and repurposing. However, since the core information is located in the audio and video signals, complete data anonymization was not an option. To assure participants' right to

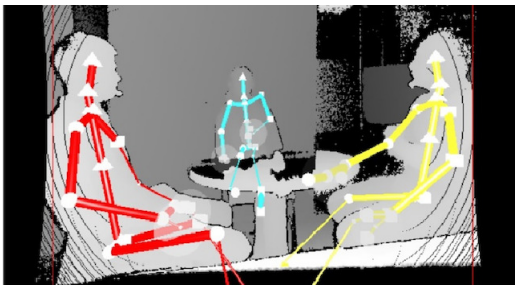


Figure 4: 3D view of Kinect 2 performing skeletal tracking.

control their personal information, participants were fully informed of the overall process and signed the respective consent forms (Koutsombogera and Vogel, 2017). Participants needed to select one of the three options in the consent forms, i.e. to allow the use of their data (a) for the purposes of the specific project, (b) for teaching and academic research purposes, and (c) for sharing in an open access repository.

The majority of the participants gave their consent to their data being publicly available in the future, therefore allowing the sharing of a large part of the corpus after the end of the project. The corpus to be released will be accompanied by an appropriate licensing scheme and will be linked to an open research repository that will provide long-term access.[3]

2.3. Personality Traits

One of our aims has been to investigate the effect of personality traits on the task success but also on participants' collaborative behavior, including their engagement, attention and the tendency to create or the ability to manage conversational conflicts. Personality variables are an important tool for the interpretation of social behavior. At the same time it has been widely acknowledged by personality researchers that there is a necessity to have an accepted classification scheme to categorise empirical findings and that the 5-factor model is a robust and meaningful framework enabling the formulation and testing of hypotheses related to individual differences in personality (Goldberg, 1992).

Therefore, out of the wide spectrum of personality measures, we opted for the big five personality traits[4] and especially those that have a communicative value with regard to the interaction behaviour of the speaker, such as conscientiousness and extraversion. Before the recordings participants completed the Big Five Inventory (BFI), a self-report inventory designed to measure the Big Five dimensions (John et al., 1991; John et al., 2008). The test consists of 44 items (statements) and the participants were asked to rate each statement to indicate the extent to which they agree or disagree with it. As a result, a list of scores per

[3]More information and access details will be published at `https://www.scss.tcd.ie/clg/MULTISIMO/`

[4]The five personality traits assessed are: Extraversion, Agreeableness, Conscientiousness, Neuroticism and Openness to Experience.

personality trait and per participant was created. The percentile rank of each participant across the five personality traits was then calculated using local norms, i.e. the norms were constructed upon the groups population. Details about the calculation of scores and percentiles are included in the corpus documentation.

3. Corpus Description

The collected dataset consists of 23 sessions of average duration of 10 minutes (min=6, max=16), resulting in a total duration of approximately 4 hours. The pairing of players was randomly scheduled, and was based on their availability to attend the recordings. While in other corpora the familiarity among group participants is a controlled variable (Thompson et al., 1993), in our case there was no attempt to pair players based on whether they know each other or not. This was decided to ensure that the least number of constraints is imposed on the experimental design, allowing for more flexibility in the formulation of research questions. In most of the sessions the participants don't know each other, although there are a few cases (i.e. in four groups) where the players are either friends or colleagues. The average age of the participants is 30 years old (min=19, max=44). Furthermore, gender is balanced, i.e. with 25 female and 24 male participants. Nevertheless, the gender distribution varies, depending on the pairing of the players. For example, there are groups where both of the players are female, or groups with male players, and groups with both genders. The participants come from different countries and span eighteen nationalities, one third of them being native English speakers. Tables 1 and 2 present the details about the gender of the participants, the number of groups whose players are familiar with each other or not, the number of native and non-native English speakers, as well as their different nationalities.

The corpus includes all files from the different cameras and microphones employed, all fully synchronized. It is also complemented by personality test results, that give evidence of various personality types, as well as by the experience assessment survey ratings. Sample data from the corpus are available for viewing at `https://www.scss.tcd.ie/clg/MULTISIMO/`.

Gender	
Male	24
Female	25
Language	
EN Native	16
EN Non-native	33
Familiarity (groups)	
Familiar	4
Non-familiar	19

Table 1: Distribution of corpus participants per gender, language and familiarity among group players

3.1. Corpus Annotations

The data analysis is based on manual annotations, automatically extracted features and the scores of the surveys used, i.e. personality self-assessments and experience assessments. Manual and automatic annotations form the ground truth on which some components will be trained and tested upon (e.g. automatic personality perception from speech and video).

Participants' Nationalities

Greek	13		Croatian	1
Irish	13		Egyptian	1
French	4		Italian	1
Brazilian	2		German	1
British	2		Kazach	1
Indian	2		Mexican	1
Pakistani	2		Romanian	1
American	1		Slovenian	1
Chinese	1		Thai	1

Table 2: Distribution of participants' nationalities

3.1.1. Manual Annotations

Manual encoding includes annotations on the audio and video streams, as well as perceived assessments, as is the case with experiments involving the other-assessment of personality traits. All corpus sessions are fully transcribed using Transcriber,[5] including segmentation of speaker turns. Video annotation tasks in progress include the labelling of gaze focus, head movements and the communicative functions that these entail, i.e. the semantics of speakers' actions in terms of feedback and turn management. For the last two elements we use the relevant values of the MUMIN coding scheme (Allwood et al., 2007). Video annotation is performed on ELAN[6] editor. Figure 5 presents a sample of an annotated file in ELAN, with information about speech transcription, speaking turns and gaze direction. Finally, manual annotations also include a set of assessments from annotators who are ranking the perceived dominance and perceived personality traits of the sessions participants.

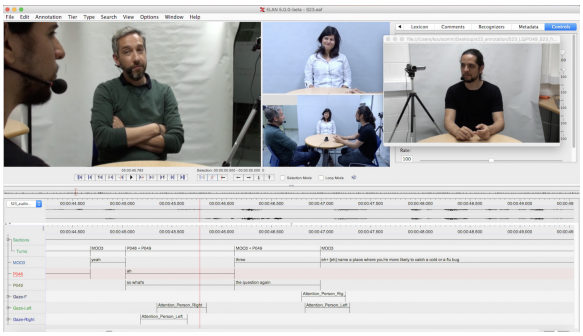


Figure 5: Sample of an annotated file in ELAN (multicamera view).

[5] `http://trans.sourceforge.net/` last accessed 15.02.2018

[6] `https://tla.mpi.nl/tools/tla-tools/elan/` last accessed 15.02.2018

3.1.2. Automatic Measurements

In parallel to the manual transcriptions, that provide information about speaking activity, voice activity and pauses have been automatically extracted. Audio files have been processed with Praat[7] to extract pitch and loudness features. The front view videos of the participants were processed with the Emotient module of iMotions (iMotions A/S, 2016), resulting in a set of measurements for each of seven emotions, [8] the facial action units involved, as well as the (visual) emotion valence. For tasks in progress related to gaze detection from video and personality perception and emotion recognition from audio, the OpenFace (Amos et al., 2016) and openSMILE (Eyben et al., 2013) toolkits are used to extract visual and acoustic features respectively.

4. Research Topics in Corpus Analysis

The corpus is considered a source that will provide tools for measuring collaboration and task success in group interactions. In this respect, we are currently investigating the following research questions: (a) the multimodal strategies that the group participants employ to collaborate with each other; (b) the factors that affect success in a collaborative discussion; and (c) behavioral cues as a source of speaker profiling.

4.1. Multimodal Turn-taking

The verbal and non-verbal strategies of the speakers (including facial expressions, gestures, body pose and head shifts, speaking times etc.) provide evidence about the structure of turn-taking, but also inform about the degree of participation of the speakers in the discussion, revealing possibly unbalanced participation or signs of dominance from certain speakers. In this particular setting, studying the role of the facilitators becomes significant, in the sense that it's crucial to understand the mechanisms they use to coordinate the interaction and to encourage equal participation of both players. Thus, the focus is on the turn-taking mechanism based on both verbal and non-verbal cues, examining who takes a turn and when, how turn allocation is performed, the number and type of overlaps, as well as on the role of pauses in turn management.

4.2. Collaboration and Task Success

We will follow a principled approach to measuring collaboration and task success. Their estimation will be based on measurable features, such as (a) the number of correct answers; (b) the amount of time groups need to complete a task; and most importantly (c) the level of participation of both players in a session, as measured by the number of turns and the duration of their speech activity, but also the amount of mutual gazes and the number of times the players address each other. In this respect, successful completion of the task does not always entail successful collaboration between players. For example, in cases where one of the players is a dominant speaker and manages to answer all questions correctly and quickly, the task may be successful (in the sense that all answers are provided), however collaboration has not been successful, as the two players have actually not shared their ideas nor participated equally in the discussion.

Interestingly, due to the nature of the game, correctness of answers is mainly a matter of popularity and not suitability. For example, players may express options that are reasonable possible answers to the specific question; however, if they're not the exact, the 3 most popular answers according to the game database, then it is considered that the correct answer is still missing. This criterion (correctness) is thus a subjective one; however, it has been noticed that it encourages the expression of emotions (e.g. surprise, curiosity, fun or frustration) and stimulates the creativity of the players in that they keep guessing and they approach their co-player to collaborate with him/her.

Furthermore, the success of the interaction and the task at hand will be correlated with the participants' personality self-assessment scores at an individual level, but also at group level, that is, combinations of personalities within the groups. The personality test scores will serve as an objective tool to interpret aspects of participants' behavior that relates to their attitudes towards the task but also towards their co-player. In parallel, personality other-assessments will be performed by observers, who will listen to and watch slices of audio and video respectively, and decide upon the participants' big 5 traits. These assessments are an additional source of information, and they will be linked to the extraction of acoustic and visual features that are related to personality, to train relevant models of personality perception.

Conversational dominance is a phenomenon closely linked to collaboration, in the sense that a participant may wish to control or dominate the other player's behavior, resulting in asymmetry in participation. Dominance in our corpus is being computed based on audio (number of turns per speaker, turn length, number of words, number of successful interruptions) and visual (mutual gaze, attention to either speaker) cues. Perception experiments are also being carried out to assess the dominance level of the players, where observers watch the videos and rank participants' dominance levels on a scale from 1 to 5.

4.3. Speaker Profiling

The corpus data, the annotations performed either manually or automatically, as well as the assessment scores, give access to a valuable source of information related to the corpus speakers. In this respect, this set of multi-faceted information constitutes the profiling of the speakers that participate in the corpus. Table 3 lists a set of cues related to speakers, that can be accessible from the corpus, its annotation and analysis.

In terms of group composition and demographic data of the speakers, we believe that gender, language nativeness or non-nativeness and familiarity level are important features to be correlated with collaboration and task success, as well as with the interpretation of the participants' social behavior. Speech cues provide important measurements about the overall activity of speakers within the sessions, but also the type of activity, e.g. whether they ask a lot of questions

[7] http://www.fon.hum.uva.nl/praat/ last accessed 15.02.2018

[8] Joy, anger, surprise, fear, sadness, disgust and contempt.

or they show a tendency to dominate the interaction by interrupting their co-participants. Acoustic and visual cues provide quantitative low-level features (e.g. action units) that characterize the speakers and may correlate with annotations or assessments to determine high-level qualitative features (e.g. type of emotion). At a lexical level, the transcripts are are a rich source of information that may reveal vocabulary variation (type/token ratio) of speakers, lexical or syntactic patterns they may use, as well as a set of markers that are related to politeness, engagement and co-operation. Therefore, the dataset is profiled according to the aforementioned cues and can be revisited accordingly at various levels and signal- or natural language processing tasks.

Speech	Number of turns
	Turn length
	Number of overlaps
	Number of interruptions
	Number of questions asked
Acoustic	Pitch and intensity
	Features related to emotions
	Features related to personality
	Features related to dominance
Visual	Emotions
	Emotion polarity
	Action units
	Gaze patterns
	Head poses
Lexical	Transcripts
	Number of words per speaker
	Type/token ratio
	Discourse markers
Perceptual	Personality other-assessment scores
	Dominance other-assessment scores
Survey	Personality self-assessment scores
	Experience self-assessment scores
Demographic	Gender
	Native language
	Familiarity
	Age

Table 3: Available cues for each speaker in the corpus

5. Conclusion

In this paper we introduced a new multimodal corpus of group collaborative interactions and we presented work in progress regarding the exploitation of the corpus in the analysis and modeling of human-human group interactions. Considering the fact that the quality of the models depends on the dataset quality, this work was driven by the need to design and implement multimodal, as naturalistic as possible, data collections involving humans engaged in interaction tasks. A suitable amount of data was collected and will enable us to draw conclusions on aspects of collaborative interactions as well as ways of measuring collaboration and task success, based on the exploitation of visual, audio and text features.

The corpus itself is a major output of this research in the sense that a large part of it (i.e. the sessions for which the participants have given their consent) will be publicly available for research purposes, so that it can be reused by other researchers to explore further questions. Also, the documentation regarding the materials used is rich enough so that the recordings experiment can be duplicated, including details about the experimental protocol and the guidelines and training that the facilitators had before the recordings. The corpus release will include audio, video (in high and low resolution formats) and Kinect files, the annotation scheme, manual annotations and automatic measurements, together with the perception experiments and surveys scores and the related documentation.

6. Acknowledgements

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 701621 (MULTISIMO). We would also like to thank the TCD Speech Communication Lab for contributing to the experimental setup.

7. Bibliographical References

Al Moubayed, S., Beskow, J., Skantze, G., and Granström, B. (2012). Furhat: A back-projected human-like robot head for multiparty human-machine interaction. In Anna Esposito, et al., editors, *Cognitive Behavioural Systems. Lecture Notes in Computer Science*, pages 114–130. Springer, Berlin, Heidelberg.

Allwood, J., Cerrato, L., Jokinen, K., Navarretta, C., and Paggio, P. (2007). The MUMIN coding scheme for the annotation of feedback, turn management and sequencing phenomena. *Language Resources and Evaluation*, 41(3/4):273–287.

Amos, B., Ludwiczuk, B., and Satyanarayanan, M. (2016). OpenFace: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science.

Bohus, D. and Horvitz, E. (2010). Facilitating multiparty dialog with gaze, gesture, and speech. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, ICMI-MLMI '10, pages 5:1–5:8, New York, NY, USA. ACM.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska, A., McCowan, I., Post, W., Reidsma, D., and Wellner, P. (2005). The AMI meeting corpus: A pre-announcement. In Steve Renals et al., editors, *MLMI'05: Proceedings of the Workshop on Machine Learning for Multimodal Interaction*, number 3869 in LNCS, pages 28–39. Springer-Verlag.

Esposito, A., Esposito, A. M., and Vogel, C. (2015). Needs and challenges in human computer interaction for processing social emotional information. *Pattern Recognition Letters*, 66:41–51.

Eyben, F., Weninger, F., Gross, F., and Schuller, B. (2013). Recent developments in openSMILE, the Munich open-source multimedia feature extractor. In *Proceedings of*

the 21st ACM International Conference on Multimedia, MM '13, pages 835–838, New York, NY, USA. ACM.

Gatica-Perez, D., Aran, O., and Jayagopi, D. (2017). Analysis of small groups. In Judee K. Burgoon, et al., editors, *Social Signal Processing*, pages 349–367. Cambridge University Press.

Goldberg, L. R. (1992). The development of markers for the big-five factor structure. *Psychological Assessment*, 4(1):26–42.

Hung, H. and Chittaranjan, G. (2010). The Idiap Wolf Corpus: Exploring group behaviour in a competitive role-playing game. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 879–882, New York, NY, USA. ACM.

iMotions A/S. (2016). iMotions Biometric Research Platform 6.0. Copenhagen, Denmark.

Jayagopi, D., Hung, H., Yeo, C., and Gatica-Perez, D. (2009). Modeling dominance in group conversations from non-verbal activity cues. *IEEE Transactions on Audio, Speech and Language Processing*, 17(3):501–513.

John, O. P., Donahue, E. M., and Kentle, R. L. (1991). The big five inventory versions 4a and 54. Technical report, University of California, Berkeley, Institute of Personality and Social Research.

John, O. P., Naumann, L. P., and Soto, C. J. (2008). Paradigm shift to the integrative big five trait taxonomy. *Handbook of personality: Theory and research*, 3:114–158.

Koutsombogera, M. and Vogel, C. (2017). Ethical responsibilities of researchers and participants in the development of multimodal interaction corpora. In *8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pages 277–282.

Mana, N., Lepri, B., Chippendale, P., Cappelletti, A., Pianesi, F., Svaizer, P., and Zancanaro, M. (2007). Multimodal corpus of multi-party meetings for automatic social behavior analysis and personality traits detection. In *Proceedings of the 2007 Workshop on Tagging, Mining and Retrieval of Human Related Activity Information*, TMR '07, pages 9–14, New York, NY, USA. ACM.

Mohammadi, G. and Vinciarelli, A. (2012). Automatic personality perception: Prediction of trait attribution based on prosodic features. *IEEE Transactions on Affective Computing*, 3(3):273–284.

Nakano, Y. and Fukuhara, Y. (2012). Estimating conversational dominance in multiparty interaction. In *Proceedings of the 14th ACM International Conference on Multimodal Interaction*, ICMI '12, pages 77–84, New York, NY, USA. ACM.

Thompson, H. S., Anderson, A., Bard, E. G., Doherty-Sneddon, G., Newlands, A., and Sotillo, C. (1993). The HCRC map task corpus: Natural dialogue for speech recognition. In *Proceedings of the Workshop on Human Language Technology*, HLT '93, pages 25–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vinciarelli, A., Dielmann, A., Favre, S., and Salamin, H. (2009). Canal9: A database of political debates for analysis of social interactions. In *3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–4.

Vinciarelli, A., Esposito, A., André, E., Bonin, F., Chetouani, M., Cohn, J. F., Cristani, M., Fuhrmann, F., Gilmartin, E., Hammal, Z., et al. (2015). Open challenges in modelling, analysis and synthesis of human behaviour in human–human and human–machine interactions. *Cognitive Computation*, 7(4):397–413.

A Semi-autonomous System for Creating a Human-Machine Interaction Corpus in Virtual Reality: *Application to the ACORFORMed System for Training Doctors to Break Bad News*

M. Ochs[1], P. Blache[2], G. Montcheuil[2,3,4], J.-M. Pergandi[3], J. Saubesty[2], D. Francon[5], and D. Mestre[3]
Aix Marseille Université CNRS ENSAM, Université de Toulon, [1]LSIS UMR 7296,
[2]LPL UMR 7309, [3]ISM UMR7287 ; [4]Boréal Innovation, [5]Institut Paoli-Calmettes (IPC), Marseille, France

Abstract

In this paper, we introduce a two-step corpora-based methodology, starting from a corpus of human-human interactions to construct a semi-autonomous system in order to collect a new corpus of human-machine interaction, a step before the development of a fully autonomous system constructed based on the analysis of the collected corpora. The presented methodology is illustrated in the context of a virtual reality training platform for doctors breaking bad news.

Keywords: Virtual patient, Virtual reality, Corpus, Health domain, Training

1. Introduction

This paper presents a *corpus-based methodology* elaborated to design a *virtual reality environment* aiming at training doctors to *break bad news to a virtual patient*. Many works have shown that doctors should be trained not only to perform medical or surgical acts but also to develop skills in communication with patients (Baile et al., 2000; Monden et al., 2016; Rosenbaum et al., 2004). Among all possible bad new, doctors can be faced to the complex situation of announcing a damage associated to a care, and that can engage their responsibility: unforeseeable medical situation, dysfunction, medical error, etc. The way *doctors deliver bad news related to damage associated with care* has a significant impact on the therapeutic process: disease evolution, adherence with treatment recommendations, litigation possibilities (Andrade et al., 2010). However, both experienced clinicians and medical students consider this task as difficult, daunting, and stressful.

Training health care professional to break bad news is now recommended by several national agencies (e.g. the French National Authority for Health, HAS)[1]. Such trainings are organized as workshops during which doctors disclose bad news to actors playing the role of patients. This solution is complex to implement: it requires several persons, it is costly, and time consuming (each 30 mn. session requires one hour of preparation). Our project[2] aims at developing *a virtual reality training system* with an embodied conversational agent playing the role of a virtual patient.

The approaches classically used to develop an interactive system including an embodied conversational agent consist in analyzing automatically or manually *one* annotated corpus of human-human interactions (Cassell, 2000). Then, a fully autonomous prototype is developed and evaluated through perceptive studies. In the approach presented here, the methodology is based on several steps, making it possible to acquire specific data and information about the required modules and functionalities. More precisely, the procedure starts classically with the collection of natural data (recordings of human-human interactions), but we introduce a new step based to acquire human-machine interactional data. The idea consists in developing a conversational agent implementing both automatic and manual modules, making it possible to simulate a fully automatized human-machine interaction, without needing to develop the entire system (in particular the comprehension module).

This methodology presents several advantages. First, both human-human and human-machine interaction corpora correspond to the same context. Their study leads to specify the verbal or non-verbal behavioral characteristics that the trainee may use when faced with a virtual human compared to interpersonal interactions. As a consequence, the prototype completely fits with the observed users behavior during human-machine interaction. Moreover, using a semi-autonomous system enables one to abstract from critical modules, difficult to develop and crucial for a successful interaction, for example the speech recognition system, that may strongly deteriorate the interaction in case of failure. Such modules are integrated in a third step without any impact on the evaluation of the first prototype. In the remaining of this paper, we illustrate an application of this methodology with the development of a prototype for training doctors to break bad news to patients.

The paper is organized as follows. In the first section, we present the global methodology based on different corpora. In section 3, we present the multimodal corpus of human-human interaction used to develop the first semi-autonomous prototype. Section 4 is dedicated to the presentation of the semi-autonomous platform and the different tools developed to collect an automatically annotated corpus of human-machine interaction. In section 5, we describe the human-machine corpus and how it is used to develop the final prototype.

[1]The French National Authority for Health is an independent public scientific authority with an overall mission of contributing to the regulation of the healthcare system by improving health quality and efficiency.

[2]ACORFORMed, (Ochs et al., 2017), http://www.lpl-aix.fr/ãcorformed/.

2. The Corpus-based Methodology

The methodology used for developing the virtual training platform consists of several steps, each based on the creation and the analysis of a corpus (human-human or human-machine interaction) in the context of breaking bad news situations. These steps can be specified as follows (see Figure 1):

1. Analysis of a human-human interaction corpus: specification of the discourse organization, definition of the different behavioral modules, development of a prototype (a semi-autonomous agent).
2. The prototype is used to collect a new corpus of human-machine interaction.
3. The new human-machine corpus is analyzed to specify the missing modules and improve the different functionalities in order to develop the autonomous version of the system.
4. Evaluation of the system, classically with perceptive experiments.

The initial corpus of doctor-patient interaction (recorded during training sessions) has been annotated manually (see section 3). The results of the analysis have been used to develop a first version of the virtual reality training platform. This platform, as described in Section 4, is semi-autonomous: some modules of the architecture are simulated by an experimenter. This semi-autonomous platform has been used to collect a new corpus of human-machine interaction, considering different devices of interaction (PC, virtual reality headset, and virtual reality cave). The collected corpora constitute the basis of the missing modules and improve the initial prototype in the perspective of a fully autonomous virtual training platform.

3. Multimodal Human-Human Corpus Analysis to Model Virtual Agent's Behavior

The modeling of the virtual patient is based on an audio-visual corpus of interactions between doctors and actors playing the role of patients (called "Standardized patients") during real training sessions in French medical institutions (it is not possible, for ethical reasons, to record real breaking bad news situations). The use of "Standardized Patients" in medical training is a common practice. The actors are carefully trained (in our project, actors are also nurses) and follow pre-determined scenarios defined by experts to play the most frequently observed patients reactions. The recommendations of the experts, doctors specialized in breaking bad news situations, are global and related to the attitude of the patient ; the verbal and non-verbal behavior of the actor remains spontaneous. Note that the videos of the corpus have been selected by the experts as representative of real breaking bad news situations.

On average, a simulated consultation lasts 9 minutes. The collected corpus is composed of 13 videos of patient-doctor interaction (each video involves a different doctor and/or a different actor-patient pair), with different scenarios[3]. The size of the corpus remains small. However, our objective is not to learn a model (in a machine learning point of view) but to extract automatically and manually information to model the virtual patient's behavior (Porhet et al., 2017) that, then, will be validated through perceptive studies.

The initial corpus has been semi-manually annotated, leading to a total duration of 119 minutes. Different tools have been used in order to annotate the corpus. First, the corpus has been automatically segmented using SPPAS (Bigi, 2012) and manually transcribed using Praat (Boersma and Weenik, 1996). The doctors' and patients non-verbal behaviors have been manually annotated using ELAN (Sloetjes and Wittenburg, 2008). Different gestures of both doctors and patients have been annotated: head movements, posture changes, gaze direction, eyebrow expressions, hand gestures, and smiles. Three experts annotated one third of the corpus each. In order to validate the annotation, 5% of the corpus has been annotated by one more annotator. The inter-annotator agreement, using Cohen's Kappa, was satisfying (k=0.63). The corpus and the annotations are described in more detail in (Porhet et al., 2017).

The annotated corpus has been analyzed for two different purposes:

- to build the *dialog model of the virtual patient*: the dialog model of the virtual patient is based on the notion of "*common ground*" (Garrod and Pickering, 2004; Stalnaker, 2002), *i.e.* a situation model represented through different variables that is updated depending on the information exchange between the interlocutors. The variables describing the situation model (e.g. the cause of the damage), specific to breaking bad news situations, have been defined based on the manual analysis of the transcribed corpus and in light of the pedagogical objective in terms of dialog. We used for the implementation the dialog system OpenDial (Lison and Kennington, 2016). In this approach, the model selects automatically the verbal patient's reaction depending on the recognized verbal utterance of the doctor (matched to flexible patterns defined through regular expressions) as well as the history of the dialog (the information already delivered by the doctor being encoded in the *common ground*). The dialog model is described in more detail in (Ochs et al., 2017) ;

- to design *non-verbal behaviors of the virtual patient*: the corpus has been used to enrich the non-verbal behavior library of the virtual patient with gestures specific to breaking bad news situations. The recurrent gestures identified in the corpus are those used to indicate pain. In total, we have enriched the virtual character's non-verbal behavior library with 16 new gestures specific to this context.

[3]The corpus is on Ortolang

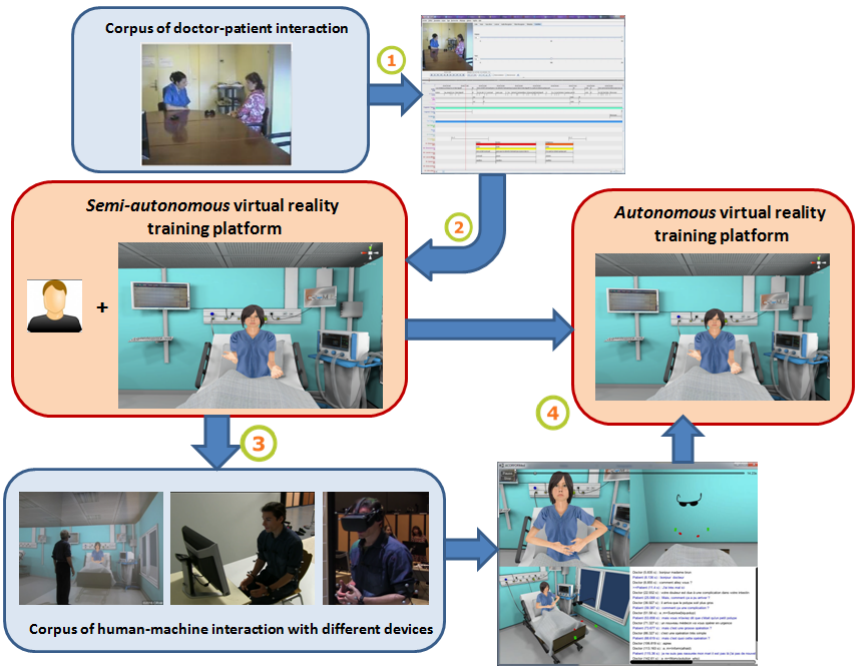


Figure 1: Methodology

The dialog model as well as the non-verbal behavior library of the virtual patient have been used to implement a first prototype of a virtual reality training platform, as described in the next section.

4. Semi-Autonomous Virtual Reality Training Platform

In order to evaluate the general architecture of the training platform and to collect a corpus of human-machine interaction specific to our project, we have developed a semi-autonomous platform, which architecture is described in Figure 2.

The platform is *semi-autonomous* because some modules of the system are automatic (for example the dialogue generation) where some others are manual. In particular, the speech recognition and the comprehension modules are simulated by a human: the doctor verbal production is interpreted in real time by the operator which selects the adequate input signal to be transmitted to the dialogue system. Indeed, these modules may be particularly critical in case of failure and then damage strongly the interaction. They represent moreover the most difficult part of the system to be developed. Replacing the module by the operator comes to a perfect speech recognition and comprehension. This makes it possible to completely control the corresponding parameters and concentrate on the evaluation of the others modules, such as the dialog supervision and the non-verbal behavior of the virtual patient. Moreover, it renders possible the evaluation of the overall interaction (e.g. presence, satisfaction, believability).

A specific interface has been designed for this purpose to enable the experimenter to select the sentences as close as possible to that has been said by the doctors (Figure 3).

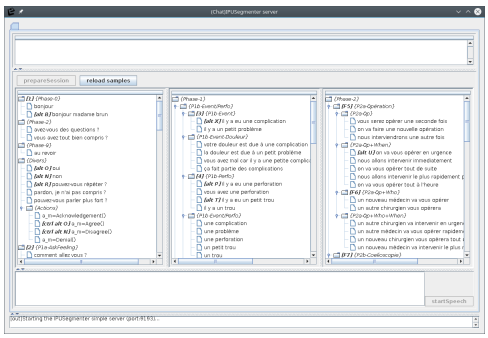


Figure 3: Screen-shot of the interface of the experimenter to select the corresponding doctor's recognized sentences

The interface contains 136 prototypical sentences (or patterns) organized into different dialog phases: greetings, asking the patient's feelings, description of the surgical problem, description of the remediation. These sentences have been defined based on the analysis of the transcribed corpus of doctor-patient interaction (see section 3). Each prototypical sentence encodes a family of possible utterances, as identified in the corpus. The sentences are encoded into an XML file. Keyboard shortcuts are associated to each sentence/pattern, and can be configured in order to be easily selected by the experimenter. Several pre-tests have been built to test the interface and train the experimenter. Note that at the difference with a "Wizard of Oz", the experimenter does not select the virtual patient's reaction but only send to the dialog model the recognized doctor's sentence. In fact in general, in a Wizard of Oz set-up, the experimenter plays the role of both the recognition module and the dialog module (understanding and selection of the response). In the proposed set-up, only the recogni-

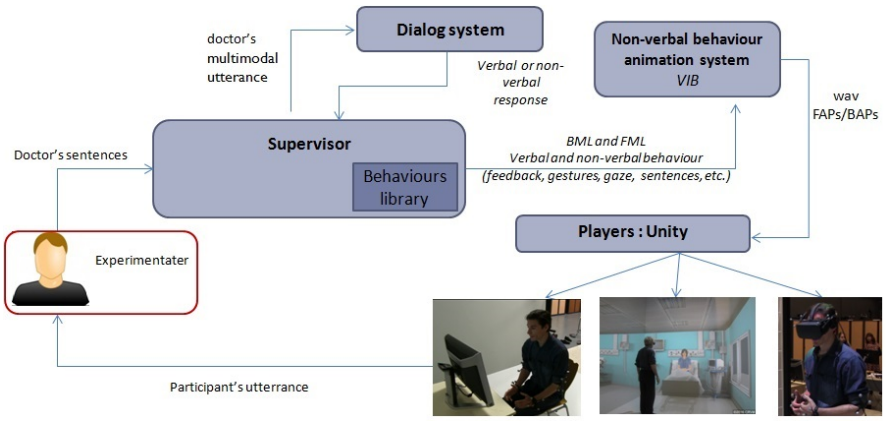


Figure 2: Overall architecture of the virtual reality training platform

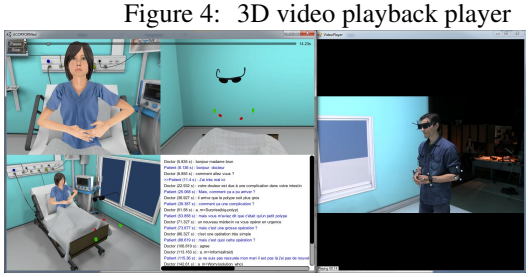
tion module is replaced by the experimenter.

The dialogue system then generates a sequence of instructions, to be sent to a non-verbal behavior animation system called VIB (Pelachaud, 2009). This system computes the animation parameters (*Facial Animation Parameters* - FAP - and *Behavioral Animation Parameters* - BAP) to animate the face and the body of the virtual patient. The result is encoded in XML and describes the communicative intention to perform (encoded in FML, *Function Markup Language*) as well as the non-verbal signals to express (encoded in BML, *Behavior Markup Language*). Moreover, the VIB system contains a text-to-speech synthesis (Aylett and Pidcock, 2007) for generating the speech, in synchronization with the non-verbal behavior (including lips animation).

In order to experiment as broadly as possible the validity of the approach, we have implemented the virtual patient on different platforms: PC, virtual reality headset, and virtual reality cave. The virtual reality cave is constituted of a 3m deep, 3m wide, and 4m high cubic space with three vertical screens and a horizontal screen (floor). A cluster of graphics machine makes it possible to deliver stereoscopic, wide-field, real-time rendering of 3D environments, including spatial sound. This offers an optimal sensorial immersion of the user. The environment has been designed to simulate a real recovery room where the breaking bad news are generally performed. The virtual agent based on the VIB platform has been integrated in by means of the *Unity* player.

In order to collect the interaction and create the corpus of human-machine interaction in the context of breaking bad news, we have implemented a specific methodology. First, the doctor is filmed using a camera. His gestures and head movements are digitally recorded from the tracking data: his head (stereo glasses), elbows and wrists are equipped with tracked targets. A high-end microphone synchronously records the participant's verbal expression. As for the virtual agent, its gesture and verbal expressions are recorded from the Unity Player. The visualization of the interaction, is done through a 3D video playback player we have developed (Figure 4). This player replays synchronously the animation and verbal expression of the virtual agent as well as the movements and video of the participant.

Figure 4: 3D video playback player



This environment facilitates the collection of corpora of doctor-virtual patient interaction in order to analyze the verbal and non-verbal behavior in different immersive environments. The collected corpus is the basis of the development of a fully autonomous virtual reality training platform.

5. From Human-Machine Corpus to a Fully Autonomous Training Platform

Using the semi-autonomous system described in the previous section, we have collected 108 interactions of participants with the virtual patient. In total, 36 persons have participated to the experimentation. Ten of them are real doctors that already have an experience in breaking bad news to real patients. The others are student from the University. Each participant has interacted with the systems 3 times with three different devices: PC, virtual reality headset, and virtual reality room. The task of the participants was to announce a digestive perforation after a gastroenterologic endoscopy in immediate post operative period[4]. Before the interaction, written instructions were presented to the participants: the role they have to play is a doctor that had just operated the virtual patient to remove a polyp in the bowel. A digestive perforation occurred during the surgery.

[4]The scenario has been carefully chosen with the medical partners of the project for several reasons (e.g. the panel of resulting damages, the difficulty of the announcement, its standard characteristics of announce).

These written instructions explains precisely the causes of the problem, the effects (pain), and the proposed remediation (a new surgery, urgently). Participants are asked to read the instructions several times as well as before each interaction. The understanding is verified by means of a questionnaire. Each participant has the instruction to announce this medical situation to the virtual patient three times with three different devices: PC, virtual reality headset, and virtual reality room. The order of the conditions were counterbalanced.

The collected corpus is composed of 108 videos (36 per device). The total duration of the corpus is 5h34 (among which two hours with real doctors). In average, an interaction lasts 3mn16 (an example of interaction is presented on the ACORFORMed site). Note that thanks to the tools described in the previous section, some of the non-verbal participant behavior can be automatically annotated.

In order to validate this first prototype, we asked the participants to fill different questionnaires on their subjective experience to measure their feeling of presence (with the *Igroup Presence Questionnaire*, IPQ (Schubert, 2003)), feeling of co-presence (Bailenson et al., 2005), and perception of the believability of the virtual patient (questions extracted from (Gerhard et al., 2001)). These subjective evaluations enabled us to *tag* the video of the corpus with the results of these tests and then to correlate objective measures (e.g. verbal and non-verbal behavior of the participants) to subjective measures (e.g. feeling of presence and perception of the virtual patient's believability).

We are currently analyzing the corpus before entering into the development of the fully autonomous training platform (in particular the comprehension and generation module). We already have used the corpus to train and test the speech recognition system, in order to ensure that the speech recognition system can accurately recognize the participants. We also verify that the recognized words and sentences activate correctly the expected rules in the dialog model. These comprehension rules are adapted in consequence. Moreover, the corpus is also used in order to compare the non-verbal behaviors through the different devices. We formulate the hypothesis that participants use less gestures in the virtual reality headset condition since they do not see their body. Depending on the results, the automatic gestures recognition could be adapted.

6. Conclusion

The development of embodied conversational agent with specific conversational skills is a challenge for human-machine communication, not only concerning the dialogue capacities of the agent, but also the adequacy of the environment to the task. The creation of rich and large corpora is a pre-requisite towards this goal. We have presented in this paper a semi-automatic platform answering these needs: offering the possibilities of an ecological human-machine interaction within a virtual reality environment following the final architecture of the system by implementing or simulating the functionalities of all the modules.

7. Acknowledgements

This work has been funded by the French National Research Agency project ACORFORMED (ANR-14-CE24-0034-02) and supported by grants ANR-16-CONV-0002 (ILCB), ANR-11-LABX-0036 (BLRI) and ANR-11-IDEX-0001-02 (A*MIDEX).

8. Bibliographical References

Andrade, A., Bagri, A., Zaw, K., Roos, B., and Ruiz, J. (2010). Avatar-mediated training in the delivery of bad news in a virtual world. *Journal of palliative medicine*, 13(12):1415–1419.

Aylett, M. P. and Pidcock, C. J. (2007). The cerevoice characterful speech synthesiser sdk. In *IVA*, pages 413–414.

Baile, W., Buckman, R., Lenzi, R., Glober, G., Beale, E., and Kudelka, A. (2000). Spikes-a six-step protocol for delivering bad news: application to the patient with cancer. *Oncologist*, 5(4):302–311.

Bailenson, J. N., Swinth, K., Hoyt, C., Persky, S., Dimov, A., and Blascovich, J. (2005). The independent and interactive effects of embodied-agent appearance and behavior on self-report, cognitive, and behavioral markers of copresence in immersive virtual environments. *Presence: Teleoperators and Virtual Environments*, 14(4):379–393.

Bigi, B. (2012). Sppas: a tool for the phonetic segmentations of speech. In *The eighth international conference on Language Resources and Evaluation*, pages 1748–1755.

Boersma, P. and Weenik, D. (1996). Praat: a system for doing phonetics by computer. report of the institute of phonetic sciences of the university of amsterdam. *Amsterdam: University of Amsterdam.*

Cassell, J. (2000). *Embodied conversational agents*. MIT press.

Garrod, S. and Pickering, M. J. (2004). Why is conversation so easy? *Trends in cognitive sciences*, 8(1):8–11.

Gerhard, M., Moore, D. J., and Hobbs, D. J. (2001). Continuous presence in collaborative virtual environments: Towards a hybrid avatar-agent model for user representation. In *International Workshop on Intelligent Virtual Agents*, pages 137–155. Springer.

Lison, P. and Kennington, C. (2016). OpenDial: A Toolkit for Developing Spoken Dialogue Systems with Probabilistic Rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Demonstrations)*, pages 67–72, Berlin, Germany. Association for Computational Linguistics.

Monden, K., Gentry, L., and Cox, T. (2016). Delivering bad news to patients. *Proceedings (Baylor University. Medical Center)*, 29(1).

Ochs, M., Montcheuil, G., Pergandi, J.-M., Saubesty, J., Donval, B., Pelachaud, C., Mestre, D., and Blache, P. (2017). An architecture of virtual patient simulation platform to train doctor to break bad news. In *International Conference on Computer Animation and Social Agents (CASA)*.

Pelachaud, C. (2009). Studies on gesture expressivity for a virtual agent. *Speech Communication*, 51(7):630–639.

Porhet, C., Ochs, M., Saubesty, J., Montcheuil, G., and Bertrand, R. (2017). Mining a multimodal corpus of doctor's training for virtual patient's feedbacks. In *Proceedings of 19th ACM International Conference on Multimodal Interaction (ICMI), Glasgow, UK*.

Rosenbaum, M., Ferguson, K., and Lobas, J. (2004). Teaching medical students and residents skills for delivering bad news: A review of strategies. *Acad Med*, 79.

Schubert, T. W. (2003). The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness. *Zeitschrift für Medienpsychologie*, 15(2):69–71.

Sloetjes, H. and Wittenburg, P. (2008). Annotation by category: Elan and iso dcr. In *LREC*.

Stalnaker, R. (2002). Common ground. *Linguistics and Philosophy*, 25(5):701–721.

Construction of English-French Multimodal Affective Conversational Corpus from TV Dramas

Sashi Novitasari[1,2], Quoc Truong Do[1], Sakriani Sakti[1], Dessi Lestari[2], and Satoshi Nakamura[1]
[1]Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama-cho, Ikoma-shi, Nara, Japan
{do.truong.dj3,ssakti,s-nakamura}@is.naist.jp
[2]Department of Informatics, Bandung Institute of Technology, Indonesia
Jl. Ganesha No.10, Bandung, Jawa Barat, Indonesia
{sashinovitasari11,dessipuji}@gmail.com

Abstract

Recently, there has been an increase of interest in constructing corpora containing social-affective interactions. But the availability of multimodal, multilingual, and emotionally rich corpora remains limited. The tasks of recording and transcribing actual human-to-human affective conversations are also tedious and time-consuming. This paper describes construction of a multimodal affective conversational corpus based on TV dramas. The data contain parallel English-French languages in lexical, acoustic, and facial features. In addition, we annotated the part of the English data with speaker and emotion information. Our corpus can be utilized to develop and assess such tasks as speaker and emotion recognition, affective speech recognition and synthesis, linguistic, and paralinguistic speech-to-speech translation as well as a multimodal dialog system.

Keywords: Corpus construction, multimodal parallel data, affective conversation, television dramas

1. Introduction

Researchers have been working on spoken language processing for decades. Such technologies as speech recognition, speech synthesis, speech translation, and spoken dialog systems have been developed and progressed from a simple machine that responds to a small set of sounds to a more sophisticated artificial agent that can handle conversational speech. Unfortunately, most of these current technologies remain limited to recognizing what was said without addressing how it was said. For example, in conventional speech-to-speech translation, the verbal content of speech is translated, but its non-verbal content or paralinguistic information is ignored.

On the other hand, based on text, speech, and video, research on emotion recognition is gaining considerable traction in the fields of human-machine communication and multimedia retrieval (Schuller et al., 2009). Numerous official emotion recognition challenges (Schuller et al., 2009; Schuller et al., 2010; Schuller et al., 2011) have been held that improved the features and the classifiers that capture the traits of spoken emotions. Furthermore, different approaches, which involve the characteristics of sounds and prosody based on speaking styles and the expressions of emotional speech classification in anime films, have also been proposed (Hara and Itou, 2010). However, these studies only focused on non-verbal information for recognizing/classifying types of emotions without addressing verbal content.

Developing an artificial agent that mimics human interaction requires a speech-oriented interface that can handle both the verbal and non-verbal content often found in conversations. Unfortunately, much less work has examined the technologies that consider both matters, because performing such emotion-affected spoken language processing tasks is not trivial. Previous studies (Williams and Stevens, 1972; Picard, 1997; Murray and Arnott, 1993) reported that emotion largely changes acoustic realization, including pitch range, speech rate, voice quality, etc. Several approaches in emotional speech recognition (Mukaihara et al., 2017) and expressive speech synthesis (Tachibana et al., 2005) have attempted to enrich the models that include prosody and emotion information. However, these studies were mostly based on speech data that were read by professional actors. Affective communication is even more complex in cross-lingual situations because of expression differences in languages and cultures. Several studies (Anumanchipalli et al., 2012; Do et al., 2016; Kano et al., 2013) have recently attempted to translate paralinguistic information across different languages, but they remain based on speech read by bilingual speakers. As a result, natural conversation that includes the expression of emotions, which play an important role during human communication, has generally not been achieved yet by these systems.

Since the nature of data determines system's quality, the utilized data must have a gap that is as small as possible with real life emotion occurrences. The availability of multimodal, multilingual, and emotionally rich corpora is still limited. The tasks of recording and transcribing actual human-to-human affective conversations are also tedious and time-consuming. For such efforts, collecting affective conversational data acts as a starting point. This paper describes the construction of a multimodal affective conversational corpus based on recorded TV dramas that have already been broadcast. The data contain parallel English-French language with lexical, acoustic, and facial features. In addition, we annotated the part of the English data with speaker and emotion information.

2. Related Works

Several works are aiming to construct multimodal non-acted affective corpora. Douglas-Cowie et al. constructed

the HUMAINE Database, a multimodal corpus that consists of natural and induced data showing emotion in a range of contexts (Douglas-Cowie et al., 2007). Another is the SEMAINE Database, an emotion-rich conversational database, which was carefully constructed by recording interactions between Sensitive Artificial Listener (SAL) and users; each recording was transcribed and annotated with the actor's emotions (McKeown et al., 2012). Most databases were constructed with specific recording settings to obtain high-quality data. However, constructing such data is time-consuming and costly. Furthermore, these databases are mostly based only on monolingual transcription.

In multilingual corpora, the ATR basic travel expression corpus (BTEC) has served as the primary source for developing broad-coverage speech translation systems (Kikui et al., 2006). Its sentences were collected by bilingual travel experts from Japanese/English sentence pairs in travel domain phrasebooks. The ATR-BTEC has been translated into 18 languages, including French, German, Italian, Chinese, Korean, and Indonesian. Each language is comprised of 160,000 sentences. This corpus contains only text-based data. The Formosa Speech Database (Formosa) (Lyu et al., 2004), a multilingual corpus for Taiwanese-Hakka-Mandarin, was created by recording 49 hours of speech. Its corpus construction project took over one year to collect recordings from thousands of speakers. The constructed corpus consists of speech and text data. Recently, the Multi30K Database (Elliott et al., 2016), which is from Multilingual English-German Image Descriptions, was created for a WMT Shared Task of Multimodal Machine Translation. It is based on the Flickr30K Entities dataset (Plummer et al., 2015) that was selected and manually translated into German and French by human translators. However, this corpus also contains only image and text data.

On the other hand, several works have explored corpus construction from such existing data as video from movies or television. A conversation dialog corpus from movies and television (Nio et al., 2014) has been constructed to provide transcriptions of natural conversations of humans since the recording and transcription of actual human-to-human conversations are tedious and time-consuming to construct (Nio et al., 2014). Yasuhara et al. also constructed a large-scale multimodal dialog corpus from movies (Yasuhara et al., 2016). Their corpus, which consists of movie files and annotations that indicate the timing of dialog segments, contains 149,689 dialogue segments from 1,722 movies. Even though both corpora were constructed to provide patterns of human communication, they also only focused on monolingual data.

Compared to previous works, we construct a multimodal and multilingual affective conversational corpus from TV-series data. In addition, we annotated some of the English data with speaker and emotion information. Our corpus can be utilized for the development and the assessment of various tasks, such as speaker and emotion recognition, affective speech recognition and synthesis, linguistic and paralinguistic speech-to-speech translation as well as a multimodal dialog system.

3. Corpus Construction

3.1. TV-Series Data Resources

The corpus was constructed from American TV dramas which have already been broadcast and its DVD has been released in market. We used 40 episodes as resources. Each episode approximately consists of 600 utterances spoken by 40 to 60 speakers. The series were originally broadcast in English and have been dubbed into French. To construct parallel data, we utilized speech data which consist of their audio-visual data and text from both the original and French-dubbed versions of the TV series. The English-French data consist of parallel text subtitles, speech audio, and video images. A detailed overview of the resource can be seen in Table 1.

Total number of utterances	25,663
Average duration of each utterance	2 sec
Minimum duration of each utterance	0.8 sec
Maximum duration of each utterance	6 sec

Table 1: Raw resource data of TV series

3.2. Data Filtering

First, from the raw English text-resources, we selected the least noisy speech segments that were only spoken by a single speaker and removed the non-English speech utterances. Then we confirmed the accuracy of the subtitles and manually corrected them if they included typos. For easier processing in subsequent phases, we reformatted the timing information from an hour into a milliseconds format. The timing information from the newly reformatted transcription data were used to cut the speech audio into utterance-based segments.

Next, to construct parallel English-French data, we selected utterances in both languages that have identical timing and discarded the rest. Due to this process, the amount of resulting utterances was greatly reduced since the number of different-timed parallel utterances was quite high.

3.3. Feature Extraction

The feature extraction phase constructed a feature dataset of three modalities: acoustic, lexical, and facial cues.

- Lexical cues:
  We extracted the lexical features based on Google's Word2Vec (Mikolov et al., 2013). The Word2Vec model generates word-level features. After that, we calculated the average value of each attribute from each word of the utterances. Note that some utterances do not have corresponding lexical features with the names of places or people because they were not included in Word2Vec model's dictionary.

- Acoustic cues:
  We used openSMILE toolkit (Eyben et al., 2010) with feature configuration from the INTERSPEECH 2010 Paralinguistic Challenge. It consists of 38 low-level descriptors and 21 functionals that resulted in 1582 acoustic features.

- Facial cues:
  We used openFace toolkit (Baltrusaitis et al., 2016) to extract the facial features from the video. To generate utterance-based facial features, first we extracted the feature of each episode using openFace and then segmented the features by calculating the average value of each attribute of the features based on the utterance's timing information that was provided in the transcription data. Since actors in recorded productions/shows often move in various ways and positions from the screen, it was not possible to generate facial cues for such scenes if speaker faces were not captured on screen.

To avoid utterances with absence-features, we also synchronized the utterance-features to make sure that all the resulting utterances have all types of features. The synchronization proceeded by extracting each modality of the features, and then we automatically generated a list of utterances with every type of feature. The details of the process are shown in Fig. 1.

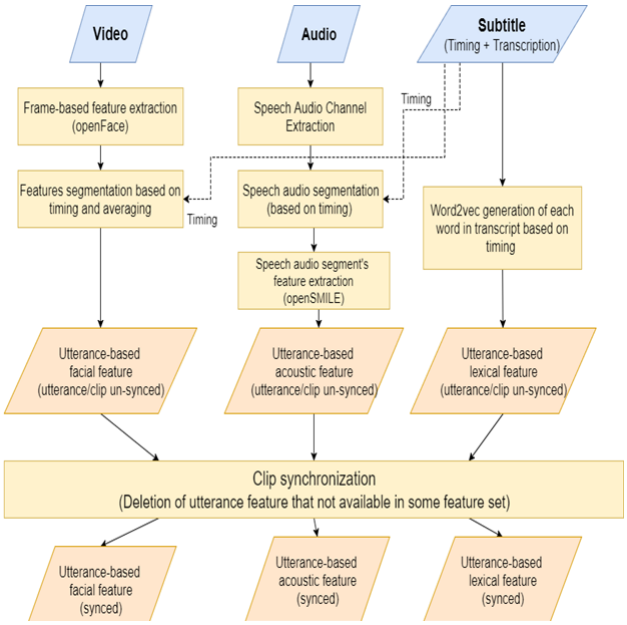


Figure 1: Multimodal feature extraction

4. Speaker and Emotion Annotation

In addition to the above multimodal dataset, we also annotated part of the English corpus with speaker and emotion information.

4.1. Speaker Annotation

Each utterance in the dataset is labeled with speaker information. The annotations were conducted manually by listening to and verifying the speaker of the utterances. The speaker label consists of 57 names of major characters who appeared in the TV series. We constructed a list of speakers by selecting those who made more than ten utterances in randomly selected episodes and appeared in more than one episode.

4.2. Emotion Annotation

We defined the emotion scope based on the circumplex model of affect (Cowie et al., 2011). Here, each utterance is labeled with emotion information based on its valence and arousal states. Valence measures the polarity of emotion; for example, 'happy' indicates a positive valence and 'sad' indicates a negative valence. Arousal measures the activity of an emotion; for example, 'tense' indicates high arousal and 'calm' indicates low arousal. To simplify the labeling, both valence and arousal were discretized into three labels: positive, neutral, and negative. The annotation for emotion was done manually by one person with a general trace program (GTrace) emotion annotation toolkit (Cowie et al., 2011) that consists of an emotion bar and a video screen. By using this toolkit, annotation was done by moving the bar's pointer to a location that corresponds to a particular emotion. Since GTrace resulted in real-valued annotation, we defined the ranges of the values or the thresholds for each class for each emotion measure. Then the annotation results by GTrace were classified based on the thresholds of the classes. Each utterance was labeled depending on the annotator's evaluation regarding the utterance's emotion. For example, an utterance that was made in an upset tone might be labeled as a 'negative' valence state and a 'positive' arousal state. Figs. 2 and 3 respectively show examples of valence and arousal annotation using Gtrace toolkit.

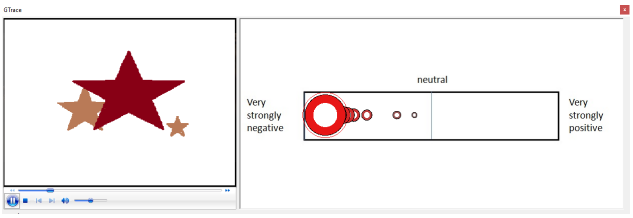


Figure 2: Valence annotation with Gtrace toolkit

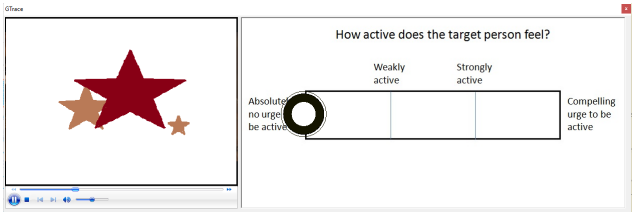


Figure 3: Arousal annotation with Gtrace toolkit

5. Corpus Analysis

The resulting corpus consists of 25,420 utterances that were spoken in English, 6,157 of which were annotated with speaker and emotion states. From the annotated utterances, 2,761 utterances have representations in acoustic, lexical, and facial cues. Among all the English-spoken utterances, only 6,114 have exact timing parallel utterances that were spoken in French.

Since affective communication in different languages and cultures might have differences in expressions, we analyzed

the constructed corpus to find them. Table 2 shows examples of English and French parallel utterances. Notice that parallel utterances don't have identical linguistic meaning. For example, the French utterance, *'c'est bien'*, should be *'it's good'* in English. In the TV series, the phrase is used by the speaker when certain work, which was done by other characters, is finished. Both utterances are used in the TV series in their respective language releases. The difference occurs because utterances in a TV drama form a conversation that is affected by the story's settings, characters, and language style. Even though the linguistic meaning is not identical, the parallel utterances have the same purpose or intention to be conveyed, and a cultural difference might affect the choice of words in the translation.

Timing (ms)	English	French
132348-133579	that's it	c'est bien
210927-212258	aye, but we have the wind	oui, mais on a le vent en poupe
215532-216624	you're coming home	tu rentres la maison

Table 2: Example of English-French parallel utterances

Next we analyzed the distribution of speakers and emotion types and focused on English data. Fig. 4 and Table 3 describe the distribution of the speaker class. In the constructed corpus, most speakers only made 10-25 utterances, while a particular speaker, probably that show's star, made over 200 utterances. As for emotion, Figs. 5 and 6 respectively illustrate the valence and arousal class distributions. Most of these utterances contain neutral emotions.

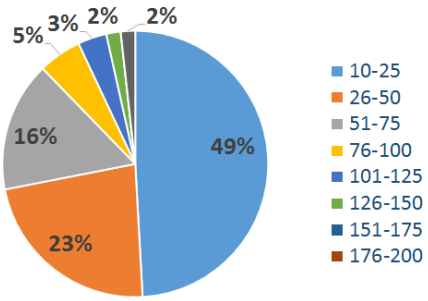


Figure 4: Speaker class distribution

Samples	Speakers
10-25	28
26-50	13
51-75	9
76-100	3
101-125	2
126-150	1
176-200	1

Table 3: Number of samples and speakers in data

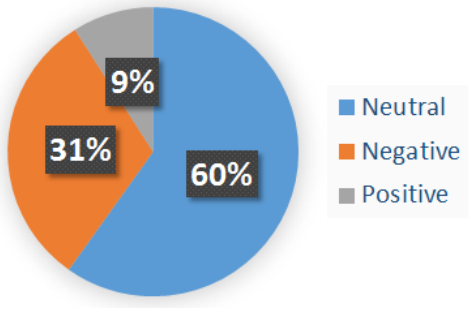


Figure 5: Valence class distribution

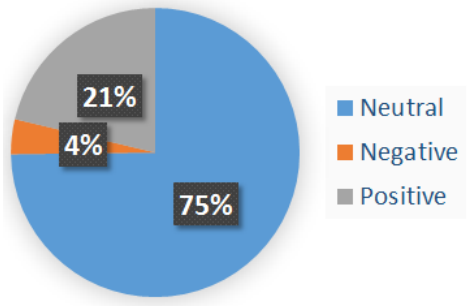


Figure 6: Arousal class distribution

6. Conclusion

In this work, we constructed a multimodal and multilingual conversational corpus from TV dramas. The data, which contain parallel English-French language in lexical, acoustic, and facial features, were annotated with speaker and emotion information. From our constructed corpus, even though we found that parallel speech may not have linguistically identical meaning, it still denotes the same thing or the same purpose. We can learn this by watching the video, although we may not realize it if we rely on speech itself for our understanding. In other words, such external speech factors as situation, cultural background, and speaker may affect word choices and speech meanings. We conclude that our corpus can be utilized to develop a paralanguage processing system that considers such factors. Future work will deepen our analysis of English-French speech data from the resources of TV-series data to increase the size of multilingual corpora since the current parallel utterances are only based on the exact timing of utterances in both languages.

7. Acknowledgement

Part of this work was supported by JSPS KAKENHI Grant Numbers JP17H06101 and JP17K00237.

8. Bibliographical References

Anumanchipalli, G., Oliveira, L., and Black, A. (2012). Intent transfer in speech-to-speech machine translation. In *Proc. of SLT*, pages 153–158.

Baltrusaitis, T., Robinson, P., and Morency, L. P. (2016). Openface: An open source facial behavior analysis

toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10, March.

Cowie, R., Cox, C., Martin, J., Batliner, A., Heylen, D., and Karpouzis, K., (2011). *Issues in Data Labelling*, chapter Emotion-Oriented Systems: The Humaine Handbook, pages 215–244. Springer-Verlag Berlin Heidelberg.

Do, Q., Toda, T., Neubig, G., Sakti, S., and Nakamura, S. (2016). Preserving word-level emphasis in speech-to-speech translation. *IEEE Transactions on Audio, Speech and Language Processing*, 25:544–556.

Eyben, F., Wollmer, M., and Schuller, B. (2010). Opensmile: The munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 1459–1462, New York, NY, USA. ACM.

Hara, Y. and Itou, K. (2010). Classification of emotional speech in anime films by using automatic temporal segmentation. In *Proc. of the second International Conference on Creative Content Technologies (CONTENT)*, pages 61–68, Lisbon, Portugal.

Kano, T., Takamichi, S., Sakti, S., Neubig, G., Toda, T., and Nakamura, S. (2013). Generalizing continuous-space translation of paralinguistic information. In *Proc. of INTERSPEECH*, pages 2614–2618.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, NIPS'13, pages 3111–3119, USA. Curran Associates Inc.

Mukaihara, K., Sakti, S., and Nakamura, S., (2017). *Recognizing Emotionally Coloured Dialogue Speech using Speaker-Adapted DNN-CNN Bottleneck Features*, chapter Speech and Computer Lecture Notes in Computer Science, pages 632–641. Lecture Notes in Computer Science. Springer.

Murray, I. and Arnott, L. (1993). Toward the simulation of emotion in synthetic speech: a review of the listerature on human vocal emotion. *Journal Acoustical Society of America*, 93(2):1097–1108.

Nio, L., Sakti, S., Neubig, G., Toda, T., and Nakamura, S. (2014). Conversation dialog corpora from television and movie scripts. In *2014 17th Oriental Chapter of the International Committee for the Co-ordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA)*, pages 1–4, Sept.

Picard, R. (1997). *Affective Computing*. MIT Press.

Schuller, B., Steidl, S., and Batliner, A. (2009). The INTERSPEECH 2009 emotion challenge. In *Proc. INTERSPEECH*, pages 312–315, Brighton, United Kingdom.

Schuller, B., Steidl, S., Burkhardt, F., Devillers, L., Muller, C., and Narayanan, S. (2010). The INTERSPEECH 2010 paralinguistic challenge. In *Proc. INTERSPEECH*, pages 2794–2797, Makuhari, Japan.

Schuller, B., Valstar, M., Eyben, F., McKeown, G., Cowie, R., and Pantic, M. (2011). AVEC 2011 - the first international audio/visual emotion challenge. In *Proc. International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 415–424, Memphis, Tennessee.

Tachibana, M., Yamagishi, J., Masuko, T., and Kobayashi, T. (2005). Speech synthesis with various emotional expressions and speaking styles by style interpolation and morphing. *IEICE*, 88:2484–2491.

Williams, C. and Stevens, K. (1972). Emotion and speech: Some acoustical correlates. *J. Acoust. Soc. Amer*, 52:1238–1250.

9. Language Resource References

Douglas-Cowie, E., Cowie, R., Sneddon, I., Cox, C., Lowry, O., Mcrorie, M., Martin, J., Devillers, L., Abrilian, S., Batliner, A., Amir, N., and Karpouzis, K. (2007). The humaine database: Addressing the collection and annotation of naturalistic and induced emotional data. In *Proceedings of the 2Nd International Conference on Affective Computing and Intelligent Interaction*, ACII '07, pages 488–500, Berlin, Heidelberg. Springer-Verlag.

Elliott, D., Frank, S., Sima'an, K., and Specia, L. (2016). Multi30k: Multilingual english-german image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74.

Kikui, G., Yamamoto, S., Takezawa, T., and Sumita, E. (2006). Comparative study on corpora for speech translation. 14(5):1674–1682.

Lyu, R., Liang, M., and Chiang, Y. (2004). Toward constructing a multilingual speech corpus for taiwanese (min-nan), hakka, and mandarin chinese. volume 9.

McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schroder, M. (2012). The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE Trans. Affect. Comput.*, 3(1):5–17, January.

Nio, L., Sakti, S., Neubig, G., Toda, T., and Nakamura, S. (2014). Conversation dialog corpora from television and movie scripts. In *2014 17th Oriental Chapter of the International Committee for the Co-ordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA)*, pages 1–4, Sept.

Plummer, B., Wang, L., Cervantes, C., Caicedo, J., Hockenmaier, J., and Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. volume abs/1505.04870.

Yasuhara, R., Inoue, M., Suga, I., and Kosaka, T. (2016). Large-scale multimodal movie dialogue corpus. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI 2016, pages 414–415, New York, NY, USA. ACM.

QUEST: A Natural Language Interface to Relational Databases

Vadim Sheinin, Elahe Khorashani, Hangu Yeo, Kun Xu, Ngoc Phuoc An Vo, Octavian Popescu
IBM Research, Yorktown Heights, US
firstname.lastname@us.ibm.com

Abstract

Natural language interfaces to databases systems allow the user to use natural language to interrogate a database. Current systems mainly focus on simple queries but neglect nested queries, which are predominant in real cases. We present a NLIDB system, QUEST, which is able to cope with nested logic queries, without imposing any restrictions on the input query. QUEST outperforms a strong baseline system by 11% accuracy.

Keywords: natural language interface to relational database, machine learning.

1. Introduction

Natural language interfaces to database systems (NLIDB) allow the user to use natural language to interrogate a database. This is an old desideratum, but there is no easy way to attain it. To begin with, even for simple queries that can be solved into a single SQL select query, namely **S**imple **L**ogic **Q**ueries (SLQs), may be challenging to get the SQL query because of different joins needed to be executed. Moreover, the queries that need a succession of select in SQL, namely **N**ested **L**ogic **Q**ueries (NLQs), are very difficult to be handled automatically. NLQs usually display a wide range of linguistic phenomena like ellipsis and anaphora, garden path parsing, etc. Due to this, many systems tend to impose strict limitations on the input query language, in a way that makes the extraction of logical relationships controllable. But this is not actually a very realistic scenario.

There is a large literature on this research field describing systems that are purely rule based, semi-supervised, and unsupervised. We point to excellent overviews of such systems in (Copestake and Jones, 1990; Androutsopoulos et al., 1995; Li and Jagadish, 2014).

The recent developments in deep learning algorithms seem to bring back into the foreground the NLIDB. Working on related topics, such as learning inferences over constrained outputs, global inference in natural languages, open domain relation extraction (Schwartz et al., 2014) , or more recently, resolving algebraic problems expressed in natural language (Koncel-Kedziorski et al., 2015), provide a strong ground for approaching the NLIDB by taking advantage of the power of semi-supervised and unsupervised techniques. However, most systems did not deal with nested logic queries.

In this paper we present a semi-supervised methodology to deal with NLQs in the most general setting: (i) we do not presuppose any restriction on the input query, and (ii) we do not presuppose any restriction on the structure of the database. From a practical point of view, we do not rely on large training data, but rather we build on different technologies that carry out linguistic processing of a query up to a point where a specialized engine is able to produce a series of SQL queries that return the answer in a recursive bottom-up process of computation. We developed a system, QUEST, on top of IBM Watson UIMA pipeline (McCord et al., 2012) and Cognos[1], which aims to provide effective SQL execution performance. In a nutshell, firstly a database connection is created in Cognos. In a Watson UIMA pipeline, a NLQ is decomposed into a set of sentences that individually can be resolved by Cognos. The modules involved in processing the query and handling Cognos are transparent to the user, who interacts only with the web interface. The user is unaware of the complexity involved either in the language analysis or in the nested SQL.

As shown in Figure 1, our system is based on a rule-based kernel whose role is to ensure a correct translation from simple queries to SQL. Generally, we have two main steps: (1) rule based semantic parsing (*Quest Engine* Box); and (2) further SQL generation and execution on databases (*Cognos Engine* Box). In the first step, we generated the lexicalized rules that would be used in the semantic parsing (offline part). Specifically, we relied on the schema independent rule templates to automatically extract the lexicalized rules from the schema annotation file. In the online part, we built a semantic parser using several Watson NLP components in a pipeline fashion, namely English Slot Grammar parser (ESG), Predicate Argument Structure (PAS), and Subtree Pattern Matching Framework (SPMF). Our semantic parser will produce a list of SQL sub-queries that are fed to the second step. In the step 2, we employed the Cognos server to combine these sub-queries into a final SQL query which is then executed on the DB2 server. Experimental results suggest that this approach helps to improve 11% accuracy over a very strong baseline.

2. Nested Question Decomposition

One of the main challenges in the NLIDB task is to handle nested questions which require additional aggregation operations over the intermediate results. For example, "*which company manufactured more products than Samsung*" is a nested question. To answer this question, we need to solve the following two sub-questions orderly:

1. how many products did Samsung manufacture ?
2. which company manufactured products more than <the answer of question #1> ?

[1] `http://www.ibm.com/analytics/us/en/technology/cognos-software/`

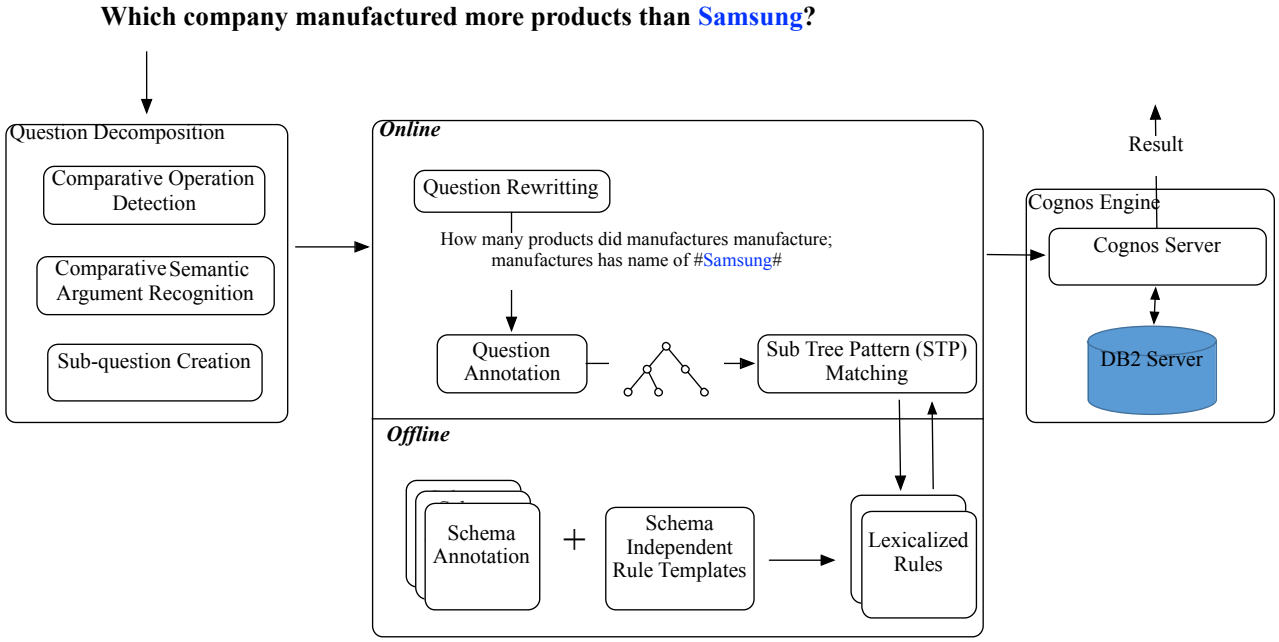


Figure 1: The Quest Architecture.

Intuitively, one way to solve the nested questions is to first decompose the question into several sub-questions, which may be then resolved in a specific order.

By analyzing real query logs in our system, we observed that in most cases, nested questions include comparative expressions, such as "*more than*" and "*higher than*". Being motivated by this observation, we proposed a decomposition method based on analyzing the comparative expressions and their arguments. Specifically, we first developed a syntactic based method to detect these comparative expressions. Then we used a syntactic scope delimiter to determine the arguments of the expressions, such as "*products*" and "*Samsung*" in Figure 2. Note that, sometimes the arguments may not be compatible due to the ellipsis in natural language, also indicating that this question needs the decomposition. For example, "*products*" which is the left argument of the comparative phrase is countable, while "*Samsung*" is not. To resolve this ellipsis, we identified the semantic arguments of the comparative phrase, which in our case, is "*the products manufactured by Samsung*". To represent the decomposition, we created the sub-questions for these arguments and replaced these arguments with symbols in the original question.

3. Quest Engine

After the nested question decomposition, each decomposed question is fed to the Quest engine, which consists of two main components, rule generation (offline part) and rule-based semantic parsing (online part). The main idea of the Quest engine is to automatically create a set of schema-dependent rules, such as sub tree patterns, and finding matches among these rules for different parts of the parsed input question. From those matched rules, we can create a set of SQL sub-queries which will be later combined and executed in the Cognos server.

3.1. Rule-based Semantic Parsing

With recent progress in the structured databases, semantic parsing against these databases has attracted more and more research interests. Almost all of the existing semantic parsers need to be trained either with the direct supervision such as logical forms (Kwiatkowski et al., 2011) or with the weak supervision such as question-answer pairs (Berant et al., 2013). However, in real application scenarios, collecting sufficient question-answer pairs to train the model remains a challenge. To address this, in this paper we proposed a rule-based semantic parser which consists of the following three components.

Question Rewriting The input question is first processed through the entity annotator, which detects entities in the question and rewrites the question by replacing these entities with their semantic types as described in schema annotation. We explicitly specified the values of these semantic types with additional description sentences. The key motivation behind this is that entities that appear in a database column can be meaningless or unrecognizable to the parser. Also, replacing these entities with relative less and finite semantic types can further ease the pattern matching problem.

Question Annotation After detecting entities and rewriting the question, the question is processed through a sequence of annotators, such as numerical annotator, date annotator, comparator annotator and predicate argument structure (PAS) annotator. Specifically, comparator annotators detect particular types of values and operators for those values that appear in the question. PAS annotator detects the predicate argument structure of the question. For instance, for the question in Figure 1, comparative annotator may create an annotation with operator "*more*". Date annotator may create an annotation if there is any temporal expression in the question.

Nested SQL

which company manufactured more products than SAMSUNG

```
"query1" AS
  (
  SELECT
    COUNT_BIG("PRODUCTS"."PRODUCT_ID") AS "PRODUCTS_PRODUCT_ID",
    "MANUFACTURERS"."NAME" AS "MANUFACTURERS_NAME"
  FROM
    "NLIDBSCH"."MANUFACTURERS" "MANUFACTURERS"
      INNER JOIN "NLIDBSCH"."PRODUCTS" "PRODUCTS"
      ON "MANUFACTURERS"."MANUFACTURER_ID" = "PRODUCTS"."MANUFACTURER_ID"
  WHERE
    LOWER("MANUFACTURERS"."NAME") LIKE LOWER('SAMSUNG')
  GROUP BY
    "MANUFACTURERS"."NAME"
  ),
"query2" AS
  (
  SELECT
    COUNT_BIG("PRODUCTS"."PRODUCT_ID") AS "PRODUCTS_PRODUCT_ID",
    "MANUFACTURERS"."NAME" AS "MANUFACTURERS_NAME"
  FROM
    "NLIDBSCH"."MANUFACTURERS" "MANUFACTURERS"
      INNER JOIN "NLIDBSCH"."PRODUCTS" "PRODUCTS"
      ON "MANUFACTURERS"."MANUFACTURER_ID" = "PRODUCTS"."MANUFACTURER_ID"
  GROUP BY
    "MANUFACTURERS"."NAME"
  ),
"auxQuery1" AS
  (
  SELECT
    SUM("query1"."PRODUCTS_PRODUCT_ID") AS "PRODUCTS_PRODUCT_ID"
  FROM
    "query1"
  ),
"auxQuery2" AS
  (
  SELECT
    SUM("query2"."PRODUCTS_PRODUCT_ID") AS "PRODUCTS_PRODUCT_ID",
    "query2"."MANUFACTURERS_NAME" AS "MANUFACTURERS_NAME"
  FROM
    "query2"
  GROUP BY
    "query2"."MANUFACTURERS_NAME"
  )
SELECT DISTINCT
  "auxQuery2"."MANUFACTURERS_NAME" AS "MANUFACTURERS_NAME"
FROM
  "auxQuery1"
    INNER JOIN "auxQuery2"
    ON "auxQuery2"."PRODUCTS_PRODUCT_ID" > "auxQuery1"."PRODUCTS_PRODUCT_ID"
```

Figure 2: SQL formulas generated by QUEST.

Sub Tree Pattern Matching The last step of our semantic parser is to search previous annotation results to find all the patterns in the lexicalized rules that match the input question. These patterns determine the types of the SQL sub-queries that should be produced and the values of the filter items if there are any.

3.2. Rules Generation

We automatically generate the lexicalized rules that are used in the sub tree pattern matching from the schema annotation and rule templates. The rule templates include a set of rule patterns for generic sentence structures. There is no schema related lexical information in these patterns, but only grammatical information that is applicable to any schema. The schema annotation is manually created for each specific database schema. It describes the different elements (tables and columns) of a database and their relationships with each other. Since the schema annotation is primarily created manually it limits the coverage of expressions against the relations in the database. Although the schema annotation is primarily created manually, it is enriched using automatically acquired paraphrases.

3.3. SQL Generation and Execution

For each question, the generated SQL sub-queries from the Quest engine are sent to the Cognos server, where the final SQL queries are produced and submitted to database.

4. Experiments

In this section we introduce the experimental setup, the main results and detailed analysis of our system. We looked separately at SLQs vs. NLQs, as indeed a large gap in performance is noticed between these two cases. Quest has been evaluated against a complex corpus, which includes both SLQs and NLQs. The baseline we used is a system that achieves state of the art results on language to logical form conversion.

4.1. Baseline and Evaluation Corpora

We implemented a deep neural network combining the insights from (Wang et al., 2015) and (Dong and Lapata, 2016). The network is a sequence to tree long short term memory (seq2tree LSTM). The input query is decoded into its logical form by a hierarchical tree decoder which identifies the arguments of a predicate.

First, we used the GeoQuery dataset (Zelle and Mooney, 1996) which contains 880 geography questions annotated with logical forms. This corpus is used only to evaluate the effectiveness of the baseline system as there are no SQL queries associated with these questions, but only their logical forms. Moreover, all these queries are simple logic queries that makes it possible for the baseline to attain an accuracy above 90%. GeoQuery dataset is standardly split into 600 queries for training, and 280 queries for testing.

We manually created a dataset from real queries on golf tournaments, named GolfQuery, made up of 2,350 queries. This corpus is made out of two parts: GF_NLQ contains 1,200 queries that are nested SQL queries, and GF_SLQ is made of the remaining 1,150 queries that are all simple logic queries. From GF_NLQ we randomly chose 249 queries for testing, the remaining 951 queries were used for training. From GF_SLQ a random set of 350 queries were chosen for testing, the rest 800 were used for training.

4.2. Results and Discussion

In the first experiment we examined the baseline system using the GeoQuery dataset. The accuracy computed as ratio between the number of correct resolved queries vs. the total number of queries, was 92% which differs insignificantly from the current state of the art performances on this corpus, and proves that this is indeed a strong baseline.

In Table 1 we presented the accuracy of the baseline system Seq2Tree and of our system Quest on GF_NLQ and GF_SLQ. Overall, Quest attained near 73% accuracy, leading to 11% increase over the baseline. While for SLQs, both system attain very good performances, above 92%, for NLQs, the Quest improves the baseline by 17%.

Looking at the results of GF_SLQ vs. GF_NLQ we can see a large difference between the accuracy of SLQs vs. NLQs. We wanted to understand how this gap is related to the number of training examples, and if there is a scalability problem for the baseline system.

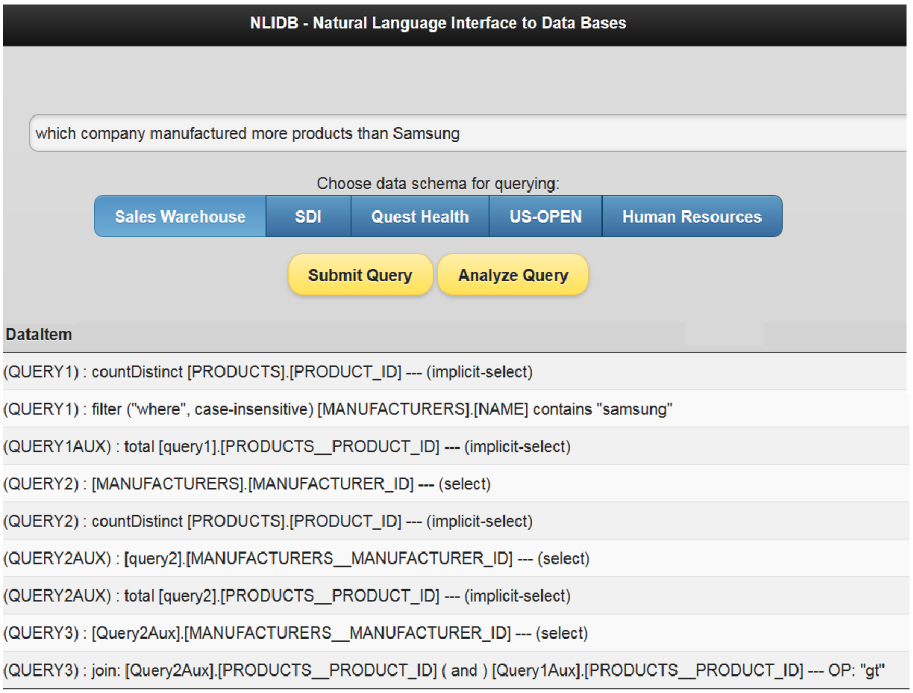


Figure 3: The natural language interface of Quest to the relational databases.



Figure 4: Two running examples of Quest system.

Scalability To evaluate the scalability of the baseline approach, we investigated the relation between the number of training examples vs. SLQ and NLQ accuracy. As shown in Figure 5, the accuracy of the baseline system grows quickly as the number of training examples increases for SLQs, both for GeoQuery and GF_SLQ datasets. However, for GF_NLQ, the learning curve looks different both for the direct test on the chosen 249 testing queries (GF_TST) of

	GF_SLQ	GF_NLQ	overall
Quest	98%	38%	73%
Seq2Tree	92%	21%	62%

Table 1: Quest and seq2Tree on Golfquery

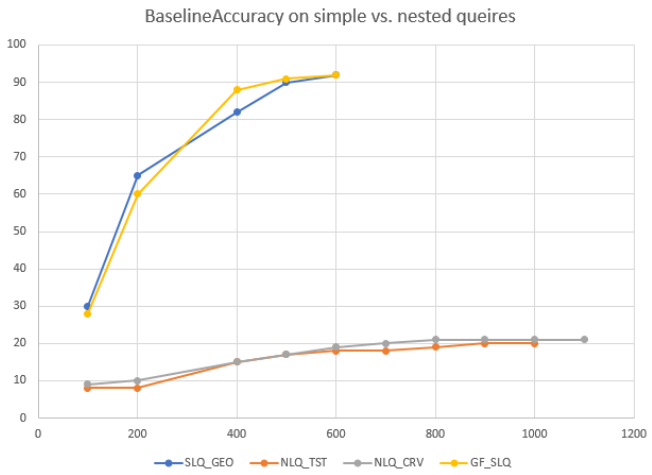


Figure 5: Seq2Tree learning curve

GF_NLQ, and for 10-fold cross validation (GF_CRV) for the whole GF_NLQ dataset. This result suggests that the number of training examples needed for a higher accuracy may be very high. This is a serious bottleneck, as in all real scenarios we are aware of, training data is unavailable.

Nested question decomposition We selected a set of 64 sentences from the GF_NLQ test set that are particularly complex. Then we run Quest with 10-fold cross validation on GF_NLQ dataset, and we imposed that a third of the test queries of each fold to be chosen randomly from the 64 complex sentences. The performance on each fold varied considerably from 30% to 63%. This is mainly due to ambiguous attachment which is an issue that we plan to address in the future.

4.3. Error Analysis

We analyzed 155 questions that are not answered correctly by our system. These errors are mainly due to three reasons: (1) QUEST now does not support certain logical operations, such as "NOT" or "OR", (2) limited sub tree pattern rules can not cover all queries, and (3) schema annotation does not define some operations. For example, although the QUEST was able to answer the question "*Who was the runner up at the 1950 US Open?*", it could not answer the question "*Who was the second place in the 2015 US Open?*" because the semantic of "*second*" was not defined in schema annotation file.

4.4. Demo

In Figure 3, we showed the system output for the query whose automatic SQL was presented in Figure 2. In Figure 4, we displayed an example of complex queries, in which the right order of operators "*per date*" and "*highest*" must be inferred. As being showed, the system solved them correctly.

5. Conclusion

In this paper we presented the main characteristics of an NLIDB system which outperforms a strong baseline (seq2Tree LSTM). It showed that the system obtained much better accuracy on nested logic queries which are the major source of difficulty in this task. The most efficient further directions for improvement are of two types: (i) negation and logical operators handling in natural language, and (ii) a better linguistic processing of the sentences in order to correctly identify the arguments of SQL queries. According to the experiments that were carried out, this could add a significant increase in accuracy.

6. Bibliographical References

Androutsopoulos, I., Ritchie, G. D., and Thanisch, P. (1995). Natural language interfaces to databases - an introduction. *Natural Language Engineering*, 1(1):29–81.

Berant, J., Chou, A., Frostig, R., and Liang, P. (2013). Semantic parsing on freebase from question-answer pairs. In *EMNLP*.

Copestake, A. A. and Jones, K. S. (1990). Natural language interfaces to databases. *Knowledge Eng. Review*, 5(4):225–249.

Dong, L. and Lapata, M. (2016). Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Koncel-Kedziorski, R., Hajishirzi, H., Sabharwal, A., Etzioni, O., and Ang, S. D. (2015). Parsing algebraic word problems into equations. *TACL*, 3:585–597.

Kwiatkowski, T., Zettlemoyer, L. S., Goldwater, S., and Steedman, M. (2011). Lexical generalization in CCG grammar induction for semantic parsing. In *ACL*, pages 1512–1523.

Li, F. and Jagadish, H. V. (2014). Constructing an interactive natural language interface for relational databases. *PVLDB*, 8(1):73–84.

McCord, M. C., Murdock, J. W., and Boguraev, B. (2012). Deep parsing in watson. *IBM Journal of Research and Development*, 56(3):3.

Schwartz, E. M., Bradlow, E. T., and Fader, P. S. (2014). Model selection using database characteristics: Developing a classification tree for longitudinal incidence data. *Marketing Science*, 33(2):188–205.

Wang, Y., Berant, J., and Liang, P. (2015). Building a semantic parser overnight. In *ACL*.

Zelle, J. M. and Mooney, R. J. (1996). Learning to parse database queries using inductive logic programming. In *AAAI*, pages 1050–1055.

TF-LM: TensorFlow-based Language Modeling Toolkit

Lyan Verwimp, Hugo Van hamme, Patrick Wambacq
ESAT – PSI, KU Leuven
Kasteelpark Arenberg 10, 3001 Heverlee, Belgium
{lyan.verwimp, hugo.vanhamme, patrick.wambacq}@esat.kuleuven.be

Abstract

Recently, an abundance of deep learning toolkits has been made freely available. These toolkits typically offer the building blocks and sometimes simple example scripts, but designing and training a model still takes a considerable amount of time and knowledge. We present language modeling scripts based on TensorFlow that allow one to train and test competitive models directly, by using a pre-defined configuration or changing it to their needs. There are several options for input features (words, characters, words combined with characters, character n-grams) and for batching (sentence- or discourse-level). The models can be used to test the perplexity, predict the next word(s), re-score hypotheses or generate debugging files for interpolation with n-gram models. Additionally, we make available LSTM language models trained on a variety of Dutch texts and English benchmarks, that can be used immediately, thereby avoiding the time and computationally expensive training process. The toolkit is open source and can be found at `https://github.com/lverwimp/tf-lm`.

Keywords: language modeling, LSTM, deep learning, toolkit

1. Introduction

Language models (LMs) play a crucial role in many speech and language processing tasks, among others speech recognition, machine translation and optical character recognition. The current state of the art are recurrent neural network (RNN) based LMs (Mikolov et al., 2010), and more specifically long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) LMs (Sundermeyer et al., 2012). Building, training, optimizing and testing these networks from scratch would require a huge amount of expertise and time. There exist many deep learning frameworks that offer the building blocks: TensorFlow (Abadi et al., 2015), Keras (Chollet, 2015), Torch (Collobert et al., 2011), Theano (Theano Development Team, 2016), Caffe (Jia et al., 2014) and others. Researchers proposing a new type of model also frequently publish their code, but typically do not offer a more general framework.

Among the deep learning frameworks, TensorFlow is arguably the most widely used, as the Github statistics in table 1. demonstrate. However, as far as we know (see section 2. for a more detailed description of the existing tools), there does not exist a toolkit that allows one to quickly design, train and test their own 'baseline' LMs with TensorFlow. We release a toolkit that contains modular code that should be easy to adapt, standard recipes to train competitive baseline models that can be adapted in a simple configuration file and LMs that are pre-trained on a variety of English benchmarks and corpora of spoken Dutch.

Framework	Stars	Forks
TensorFlow	86,940	33,924
Caffe	20,003	12,275
Keras	19,261	6,978
Torch	7,232	2,140
Theano	6,862	2,277

Table 1: Github statistics for several deep learning frameworks, recorded on Sept 6, 2017.

In the remainder of this paper, we discuss other toolkits that provide language modeling scripts in section 2., describe the documentation and functionality that our toolkit provides (sections 3. and 4.) and the pre-trained LMs (section 5.). We end with experimental results (section 6.) and a conclusion (section 7.).

2. Related work

Although there already exist open-source neural language modeling toolkits, they only offer feedforward NNs (Schwenk, 2013), vanilla RNNs (Mikolov et al., 2014) or are not easy to adapt and hence not very attractive to researchers (Sundermeyer et al., 2014). TheanoLM (Enarvi and Kurimo, 2016) should be more flexible and offers many state-of-the-art models such as LSTMs and GRUs, but is built on Theano. TensorFlow has a larger community than Theano and is updated frequently, also including state-of-the-art models. Moreover, it has been announced that the development of Theano will not be continued.

The TensorFlow documentation offers a tutorial on recurrent neural network language modeling (TensorFlow, 2017), with code to train a word-level LM. It trains on batches that go across sentence boundaries, while not resetting the state of the LSTM for every new sentence. This implies that the model in theory can remember words from the previous sentence(s). However, for some architectures (e.g. bidirectional models) or applications, working on the sentence level is a more natural choice. As explained in section 4., we offer many more options in our toolkit, among others the choice between training on sentence-level or 'discourse'-level batches.

Several Github projects provide code to train LMs with TensorFlow, but they are often quite specialized. For example, *word-rnn-tensorflow* (Kim, 2017) and *char-rnn-tensorflow* (Ozair, 2017) allow one to train respectively word-level and character-level RNN LMs that one can use to sample text (this corresponds to what we call 'predicting the next word(s)', see section 4.4.).

To the best of our knowledge, we are not aware of a similar TensorFlow-based language modeling toolkit that offers the same variety of options as ours.

3. Documentation

Documentation about the toolkit can be found in the Github repository: we provide a high-level overview of the options the toolkit offers, an overview of what every script does, a list of possible combinations of options together with examples of commands and a link to the pre-trained models. We also give a detailed description of what the configuration should look like: mandatory versus optional parameters, expected type for every parameter and a description of what they do. The web page containing the pre-trained LSTM LMs also gives some information about the data and models.

4. Functionality

TF-LM offers different options with respect to device placement, data reading, unit of input/output, batching, training schemes and testing.

Firstly, the code will automatically run on GPU if one is available, but if one does not want to use the available GPU, it is possible to run on CPU by simply using the command line option *--device cpu*.

Secondly, with respect to data handling, all data is read at once and kept in memory, but this is not possible when dealing with large datasets. For those datasets one can choose to read the data sentence per sentence. This option can so far only be combined with sentence-level batches (see section 4.2. for more information about batching).

Thirdly, the main script will automatically train, validate and test the model specified in the configuration file, unless certain options are switched off (with command line options *--train False*, *--valid False* or *--test False*). Thus, if one for example only wants to check the perplexity of a certain data set for an already trained model, this can be done by using these switches.

All other options that the toolkit offers are specified in a configuration file (command line option *--config*), which contains a list of parameters and values. Certain parameters should always be specified, such as the path where the model should be saved and the path where the data can be found, others are optional. More detailed information about every option, and about the possible combinations of options can be found in the Github READMEs.

We tried to design the toolkit in such a manner that implementing new models or new manners of feeding data, training or testing the models should be easy. One can typically start from a base class and adapt only the parts that are necessary: e.g. the LM that takes as input character n-grams rather than words inherits from the baseline LM class, and only adapts the initialization of variables and the manner in which the input embeddings are generated.

4.1. Words, characters or both?

As regards unit of input/output, there are four options: word, character, word and characters and character n-grams. The word-level LM is the default option, and trains a model that predicts the next word given the previous words. The input embedding $\mathbf{e}_t$ for the word-level model is calculated as follows:

$$\mathbf{e}_t = \mathbf{W}_w \times \mathbf{w}_t \tag{1}$$

with $\mathbf{W}_w$ the word embedding matrix and $\mathbf{w}_t$ the one-hot vector of the word at time step t. The character-level LM does exactly the same but for characters: $\mathbf{c}_t$ is the one-hot vector of the current character.

$$\mathbf{e}_t = \mathbf{W}_c \times \mathbf{c}_t \tag{2}$$

In the output layer, this model predicts the next character.

The combination of word and characters feeds the current word to the input along with a predetermined number of characters occurring in the word, as in (Verwimp et al., 2017b). The word and character embeddings are concatenated and the result of this operation is fed to the LSTM:

$$\begin{aligned}\mathbf{e}_t^\top = [(\mathbf{W}_w \times \mathbf{w}_t)^\top (\mathbf{W}_c^1 \times \mathbf{c}_t^1)^\top \\ (\mathbf{W}_c^2 \times \mathbf{c}_t^2)^\top \ \ldots \ (\mathbf{W}_c^n \times \mathbf{c}_t^n)^\top]\end{aligned} \tag{3}$$

where $\mathbf{c}_t^1$ is the one-hot encoding of the first character, $\mathbf{W}_c^1$ its embedding matrix and n the total number of characters added to the model. This option can be specified with the parameter *word_char_concat* (set to *True*), the number of characters to be added with *num_char*, the size of the character embeddings with *char_size* and the order in which the characters should be added with *order*: *begin_first* implies that we start at the beginning of the word (e.g. 5 characters from 'pineapple': *p, i, n, e, a*), *end_first* that we start at the ending of the word (e.g. 5 characters from 'pineapple': *e, l, p, p, a*), and *both* that we add both the first *num_char* characters starting from the beginning and from the ending (e.g. 3 characters from 'pineapple': *p, i, n; e, l, p*). In the output layer the model predicts the next word.

The option of character n-grams (parameter *char_ngram*) feeds a vector containing the counts of all character n-grams that occur in the current word to the network:

$$\mathbf{e}_t = \mathbf{W}_g \times \mathbf{g}_t \tag{4}$$

where $\mathbf{g}_t$ is *not* a one-hot vector but a vector of the length of the character n-gram vocabulary, containing for every character n-gram its frequency in the current word. For example, for the word 'home', the character 2-grams are: *<bow>h, ho, om, me and e<eow>* (we append a beginning- and end-of-word token *<bow>, <eow>*). It is also possible to use skipgrams (parameter *skipgram* with an integer value specifying the number of skips), for example with skips of 1 character, 'home' has the following skipgrams: *<bow>o, hm, oe, m<eow>*.

In this model, the representation needs more memory, since every word is represented by a vector instead of an index, which is the implicit representation of a one-hot vector. To restrict the size of the input vocabulary, one can choose to set a frequency cutoff for the character n-grams (*ngram_cutoff*): all n-grams not in the resulting vocabulary are mapped to an *unknown-ngram*-symbol. Another option to reduce the vocabulary is to only use lowercase characters

and add a special symbol <*cap*> to the vector and assign it the number of capitals in the word (parameter *capital*).
Finally, it is also possible to reserve a part of the total input embedding for a standard word embedding, in a similar manner as for character-word LMs (Verwimp et al., 2017b):

$$\mathbf{e}_t^\top = [(\mathbf{W}_w \times \mathbf{w}_t)^\top (\mathbf{W}_g \times \mathbf{g}_t)^\top] \quad (5)$$

In this case, the input embedding consists of a concatenation of the word embedding and the character n-gram input. In the output layer, the character n-gram model always predicts a distribution over words.

4.2. Going beyond sentence boundaries?

By default, the models are trained on batches that optimally make use of the available space. For example (extract taken from Penn TreeBank):

> *"owned by <unk> & <unk> co. was under contract with <unk> to make the cigarette filters <eos> the finding probably"*

The batches may contain (multiple) (parts of a) sentence, separated from each other with the end-of-sentence token <*eos*>. Since the hidden state of the LSTM is transferred across the batches, it is not a problem that sentences are spread over several batches and the LSTM can in theory remember information from previous sentences. If the parameter *per_sentence* is present in the configuration file, the model will train on batches that contain only one sentence, padded until they all have the same length:

> *"<bos> the plant which is owned by <unk> & <unk> co. was under contract with <unk> to make the cigarette filters <eos> @ @ @ ..."*
>
> *"<bos> the finding probably will support those who argue that the u.s. should regulate the class of asbestos including <unk> more <unk> than the common kind of asbestos <unk> found in most schools and other buildings dr. <unk> said <eos> @ @ @ ..."*

In the above examples, '@' is the padding symbol and the number of padding symbols that has to be added to obtain the length of the longest sentence in the whole dataset is often large. We also introduce a beginning-of-sentence symbol <*bos*> to be able to predict the first word in the sentence. In this model, a lot of memory is wasted on padding, although in principle no extra computation is done using TensorFlow's dynamic RNN. To reduce the memory usage, it is possible to stream the data sentence per sentence. The state of the LSTM is always reset after each sentence, such that it does not remember the previous sentence(s). The perplexity calculation is adapted to exclude the padding. We will refer to this condition as 'sentence LSTM', whereas the one that should remember previous sentences is referred to as 'discourse LSTM'.
It is important to note that most research papers on language modeling do not explicitly mention whether their models go beyond sentence boundaries or not. Some exceptions are (Pelemans et al., 2016) and (Chelba et al., 2017), who report worse perplexity results for models trained on sentence level than on discourse level, supporting the intuition that knowledge of the previous sentence(s) can have a positive influence on the language modeling capacity. However, for certain architectures or applications, such as bidirectional models, sentence-level models are required.

4.3. Training

Certain parameters for training always have to be specified in the configuration file. Firstly, the vocabulary size should be specified (*vocab_size*) and if it is smaller than the full vocabulary of the data, words that do not appear in the vocabulary should be mapped to an *unknown* token (we also make available a script to do this mapping). All models are trained with an open vocabulary: the *unknown* token is part of the input and output vocabulary and hence an 'unknown' word can be predicted. If one does not want to use a vocabulary based on the frequency of the words, one can load their own vocabulary with the option *read_vocab_from_file*.
Secondly, the size of the TensorFlow graph can be specified in number of layers, number of neurons per layer, batch size and number of steps to unroll the network for backpropagation through time. The models are randomly initialized with a uniform distribution between *-init_scale* and *+init_scale*. Several optimizers are included (stochastic gradient descent, adam and adagrad) but a new optimizer can be easily added. With respect to regularization, by default dropout is used on the input embeddings and on the outputs of the LSTM cell, and the norm of the gradients is clipped to avoid exploding gradients.
If the configuration parameter *bidirectional* is added, the LM is trained as a bidirectional LSTM, whereby the forward state of the current time step and the backward state of next time step + 2 are combined:

$$y_{t+1} = f(W_o^f \; h_t^f + W_o^b \; h_{t+2}^b + b) \quad (6)$$

In the equation above, f is the softmax function, y_{t+1} is the probability distribution predicted for the word at time step $t+1$, W_o^f and W_o^b the output weight matrices of respectively the forward and backward LSTM, h_t^f the hidden state of the forward LSTM for the current time step, h_{t+2}^b the hidden state of the backward LSTM for the next time step + 2 and b the output bias vector. We use h_{t+2}^b because for the task of language modeling, the input from the next time step is equal to the target from the current time step.
There are two training schemes available: training a fixed number of epochs or training with early stopping. For early stopping, the validation perplexity of the current epoch is compared with those of the x (specified with *early_stop*) previous epochs, and training is stopped when it has been higher x times. If this is not the case, training is continued until the maximum number of epochs (*max_max_epoch*), similar to the other training scheme. Both schemes can be combined with an exponentially decaying learning rate: during the first epochs (specified with *max_epoch*), the same initial learning rate is used. Then an exponential decay is applied:

$$\eta_i = \alpha \; \eta_{i-1} \quad (7)$$

where η_i is the learning rate at epoch i and α the learning rate decay.

4.4. Testing

TF-LM offers several options to test the performance of a trained LM. Firstly, the test or validation perplexity of the standard data sets can be calculated by running the same configuration with the *--train* and respectively *--valid* or *--test* arguments switched off. Other test sets can be specified with the *other_test* parameter.

Secondly, a trained model can be used for re-scoring sentences: given a list of sentences/hypotheses, the model will assign log probabilities to every sentence, that can be used for re-ranking hypotheses.

A third option is generating the most likely (sequence of) word(s) given a certain history according to a trained model. This option can be specified with *predict_next* and takes as input a file containing word sequences that serve as the 'history'. It then iteratively samples the most likely word given the history, then given the history including the previously sampled word, and so on. It is possible to specify the maximum number of words that should be generated with *max_num_predictions* (the default is 100). The generation will stop if the <*eos*>-symbol is predicted or else if the maximum number of predictions has been generated. Optionally, one can choose to not generate the most likely word at every step, but to sample from a multinomial distribution as specified by the softmax probabilities.

The final option for testing can be used to easily calculate interpolation weights for LSTM LMs and n-gram LMs. SRILM (Stolcke, 2002) offers a script to calculate the optimal interpolation weights between several models (*compute-best-mix*) based on the outputs of running different LMs on the same test set (with the *-debug 2* option, which prints the probabilities per word). The debug file generated by our code has a similar structure as SRILM's debug file, containing (log) probabilities per word.

Re-scoring, generating the next word(s) and generating a debug file are all implemented on sentence-level batches, even if the LM was trained on discourse-level batches.

5. Pre-trained LSTM Language Models

We make available LSTM LMs trained on several English and Dutch datasets, that can be found at `http://homes.esat.kuleuven.be/~lverwimp/lstm_lm/`.

The English datasets are two publicly available benchmarks: Penn TreeBank (PTB) (Marcus et al., 1993) and WikiText (Wiki) (Merity et al., 2016). PTB contains 900k word tokens for training, 70k word tokens as validation set and 80k words as test set and has a vocabulary of 10k words. For WikiText we used the small dataset for training, WikiText-2, which contains 2M words for training, 210k words for validation and 240k words for testing. We used the same 33k vocabulary as Merity et al. (2016).

The Dutch datasets are two corpora of spoken Dutch, the Corpus of Spoken Dutch (CGN) (Oostdijk, 2000) and a dataset of subtitles (Sub). The CGN dataset contains normalized versions of all components, both Flemish and Dutch. The dataset is split in a training set of 8M words, a validation set of 200k words and a test set of 240k words. The Sub dataset contains 45M words for training, 100k words for validation and 120k words for testing. It contains subtitles for a variety of TV shows from the Flemish national broadcaster, including fiction, documentaries, talk shows, quizzes, lifestyle programs and news. For more information about this dataset, including a reference to pre-trained n-gram LMs, see (Verwimp et al., 2017a); for more details about the normalization, we refer the reader to (Verwimp et al., 2016). The vocabulary for both Dutch datasets is limited to the 100k most frequent words (the full vocabulary size is 145k for CGN and 330k for the subtitles dataset).

6. Experimental results

6.1. Perplexity

In table 2, we report perplexity results for the pre-trained models.

Data	5-gram	sentence LSTM	discourse LSTM
PTB	147.9	102.4	**84.1**
Wiki	231.0	150.6	**98.2**
CGN	395.2	257.6	**192.6**
Sub	114.5	74.4	**65.1**

Table 2: Test perplexities for the pre-trained sentence and discourse LSTM LMs compared to 5-gram LMs.

We compare a 5-gram model with interpolated modified Kneser-Ney smoothing (Chen and Goodman, 1999) trained with SRILM (Stolcke, 2002), with two baseline LSTM models, one trained on sentence-level batches and one trained on discourse-level batches. More details about the complexity of the models can be found on the Github repository and the download page. The perplexities for the English benchmark datasets are comparable to results for baseline word-level models reported in the literature: e.g. Jozefowicz et al. (2015) report a perplexity of 81.4 for a standard LSTM model on PTB and Grave et al. (2017) report a baseline perplexity of 99.3 for WikiText. As expected, discourse-level batching performs better than sentence-level batching: the improvement is largest for WikiText that has many long-term dependencies.

Input/output unit	Perplexity
word	84.1
character 2-gram	101.1
word-char	**83.6**

Table 3: Test perplexities for models trained with different input/output units on Penn TreeBank. The 'word-char' model is one to which 9 characters starting from the end of the word have been added.

In table 3, we report results for LMs trained with different input units. We see that models with concatenated word and character embeddings (last row) perform slightly better than word-level models, and that character 2-grams as input decreases the performance. Note that we did not do extensive experiments with character n-gram input, so it is very well possible that with a more thorough investigation, better results can be obtained. For example, this model might be interesting for other, more morphologically complex languages. Using a convolutional layer before the LSTM layer might also improve them. A character-level model with the

same hidden size, which has characters as input and output (and hence a different output vocabulary than the models in table 3), has a perplexity of 2.8.

6.2. Interpolation weights

The toolkit can also be used to calculate interpolation weights between different neural LMs and/or n-gram LMs. As an example, we generate a debugging file for the validation set of PTB with our pre-trained LSTM LM. If we generate a debugging file for a 5-gram LM with interpolated modified Kneser-Ney smoothing with the SRILM toolkit, we can automatically define the optimal interpolation weights on the validation set, which are 0.24 for the n-gram model and 0.76 for the LSTM model. These weights can subsequently be used in applications such as N-best rescoring. The perplexity of the interpolated model (see table 4) has improved 8% with respect to the LSTM LM alone.

Model	Valid Perplexity	Test Perplexity
5-gram	155.1	147.9
LSTM	107.5	102.4
interpolation	**98.6**	**94.7**

Table 4: Validation and test perplexities for a 5-gram LM, an LSTM LM and their interpolation on PTB.

6.3. Predicting next word(s)

In table 5, we show some examples of generating text with trained LMs. We show the difference between generating text with a model trained on PTB or on WikiText, and between generating the most likely word at every time step ('-p') or sampling based on a multinomial distribution ('-s'). Since the class of unknown words is treated as a word itself, unknown words can in principle be predicted too, but in the case of sampling from the multinomial distribution, we sample another word is the unknown word has been chosen. We observe that sampling from the multinomial distribution, hence not always choosing the most probable word, results in sequences that are more 'free' in the sense that they are often longer and contain less frequent words.

6.4. Rescoring

The rescoring option can be used to assign log probabilities to a list of sentences with a trained LM. For example, in table 6 we show some results after rescoring the 100-best lists from the DARPA WSJ'92 and WSJ'93 data sets, used by among others (Xu et al., 2009) and (Filimonov and Harper, 2009). The LM used for rescoring is the pre-trained PTB LM. Note that these log probabilities should be scaled by the number of words in the sentence (with a *word insertion penalty*) to properly compare them, since hypotheses with more words get a lower probability as more log probabilities are added. Compare for example the log probability for *he made a sales goal he says* (-31.997), and for the exact same sentence but with an extra word at the end, *he made a sales goal he says it* (-35.9978).

seed	**consumers may...**
PTB-p	be able to <unk> the <unk> of the <unk>
PTB-s	no longer be active
Wiki-p	be used to be a <unk>
Wiki-s	have made 0 - 3 victory in Mahwah
seed	**consumers may want...**
PTB-p	to be <unk>
PTB-s	to keep the gop golden share
Wiki-p	to be the first to be <unk>
Wiki-s	to have a strong impact on the storytelling
seed	**in recent...**
PTB-p	years
PTB-s	months after four years of investments
	last month for the year alone
Wiki-p	years
Wiki-s	years , broadcasts by Conservative Party theatre
	in the 18th century
seed	**The city 's growth has reflected the push and pull of many social...**
PTB-p	security benefits
PTB-s	states in the areas of southeast asia
Wiki-p	contexts
Wiki-s	forms

Table 5: Results of predicting the most probable sequence ('-p') or sampling based on a multinomial distribution ('-s') with the pre-trained PTB and Wiki models given several seeds extracted from the validation data from PTB and Wiki.

Hypothesis	Log prob
he made a sales goal he says	-31.9997
he made a sales coal he says	-32.0020
he made a sales goal he sets	-32.0021
he made a sales call he says	-32.0053
he made its sales goal he says	-32.0184
he made its sales coal he says	-32.0221
he made its sales call he says	32.0255
he made as sales goal he says	-32.0304
he made as sales coal he says	-32.0331
he made as sales call he says	-32.0360
he made it sails call he says	-32.0619
he may to a sales goal he says	-35.9857
he made a sales goal he says it	-35.9978
he made a sale of coal he says	-35.9994
he made a sales to call he says	-36.0034
he made a sales call he says it	-36.0043
he made a sale to call he says	-36.0083

Table 6: Sorted log probabilities for part of the WSJ N-best list development set, assigned by a model trained on PTB.

7. Conclusion

We release an open-source toolkit for language modeling based on TensorFlow: `https://github.com/lverwimp/tf-lm`. It contains several options for input/output unit, batching, training and testing and is easy to adapt. We show that it obtains state-of-the-art performance on English benchmarks, and release LMs trained on those benchmarks and on corpora of spoken Dutch.

8. Acknowledgements

This research is funded by the Flemish government agency IWT project 130041, SCATE.

9. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Andrew Harp, G. I., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng., X. (2015). Tensorflow: Large-scale machine learning on heterogeneous systems. *Software available from tensorflow.org*.

Chelba, C., Norouzi, M., and Bengio, S. (2017). N-gram language modeling using recurrent neural network estimation. Technical report, Google.

Chen, S. F. and Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 17:359–394.

Chollet, F. (2015). Keras. `https://github.com/fchollet/keras`.

Collobert, R., Kavukcuoglu, K., and Farabet, C. (2011). Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*.

Enarvi, S. and Kurimo, M. (2016). TheanoLM – An Extensible Toolkit for Neural Network Language Modeling. In *Proceedings Interspeech*, pages 3052–3056.

Filimonov, D. and Harper, M. (2009). A joint language model with fine-grain syntactic tags. In *Proceedings Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Grave, E., Joulin, A., and Usunier, N. (2017). Improving neural language models with a continuous cache. In *Proceedings International Conference on Learning Representations (ICLR)*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

Jozefowicz, R., Zaremba, W., and Sutskever, I. (2015). An Empirical Exploration of Recurrent Network Architectures. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pages 2342–2350.

Kim, S. (2017). word-rnn-tensorflow. `https://github.com/hunkim/word-rnn-tensorflow`.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a Large Annotated Corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. (2016). Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Mikolov, T., Karafiát, M., Burget, L., Černocký, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Proceedings Interspeech*, pages 1045–1048.

Mikolov, T., Kombrink, S., Deoras, A., Burget, L., and Černocký, J. (2014). RNNLM - recurrent neural network language modeling toolkit. In *Proceedings Interspeech*, pages 2093–2097.

Oostdijk, N. (2000). The Spoken Dutch Corpus. Overview and first Evaluation. In *Proceedings Language Resources and Evaluation Conference (LREC)*, pages 887–894.

Ozair, S. (2017). char-rnn-tensorflow. `https://github.com/sherjilozair/char-rnn-tensorflow`.

Pelemans, J., Shazeer, N., and Chelba, C. (2016). Sparse non-negative matrix language modeling. *Transactions of the Association for Computational Linguistics*, 4:329–342.

Schwenk, H. (2013). CSLM – a modular open-source continuous space language modeling toolkit. In *Proceedings Interspeech*, pages 1198–1202.

Stolcke, A. (2002). SRILM an extensible language modeling toolkit. In *Proceedings International Conference Spoken Language Processing*, pages 901–904.

Sundermeyer, M., Schlüter, R., and Ney, H. (2012). LSTM Neural Networks for Language Modeling. In *Proceedings Interspeech*, pages 1724–1734.

Sundermeyer, M., Schlüter, R., and Ney, H. (2014). rwthlm – The RWTH Aachen University Neural Network Language Modeling Toolkit. In *Proceedings Interspeech*, pages 2093–2097.

TensorFlow. (2017). Recurrent neural networks. `https://www.tensorflow.org/tutorials/recurrent`.

Theano Development Team. (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688*.

Verwimp, L., Desplanques, B., Demuynck, K., Pelemans, J., Lycke, M., and Wambacq, P. (2016). STON: Efficient subtitling in Dutch using state-of-the-art tools. In *Proceedings Interspeech*, pages 780–781.

Verwimp, L., Pelemans, J., Lycke, M., Van hamme, H., and Wambacq, P. (2017a). Language Models of Spoken Dutch. *arXiv preprint arXiv:1709.03759*.

Verwimp, L., Pelemans, J., Van hamme, H., and Wambacq, P. (2017b). Character-Word LSTM Language Models. In *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 417–427.

Xu, P., Karakos, D., and Khudanpur, S. (2009). Self-supervised discriminative training of statistical language models. In *Proceedings IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*.

Grapheme-level Awareness in Word Embeddings for Morphologically Rich Languages

Suzi Park, Hyopil Shin
Department of Linguistics, Seoul National University
Gwanakro 1, Gwanak-gu, Seoul, South Korea

{mam3b,hpshin}@snu.ac.kr

Abstract

Learning word vectors from character level is an effective method to improve word embeddings for morphologically rich languages. However, most of these techniques have been applied to languages that are inflectional and written in Roman alphabets. In this paper, we investigate languages that are agglutinative and represented by non-alphabetic scripts, choosing Korean as a case study. We present a grapheme-level coding procedure for neural word embedding that utilizes word-internal features that are composed of syllable characters (Character CNN). Observing that our grapheme-level model is more capable of representing functional and semantic similarities, grouping allomorphs, and disambiguating homographs than syllable-level and word-level models, we recognize the importance of knowledge on the morphological typology and diversity of writing systems.

Keywords: Character-level CNN, Word Embedding, Grapheme-level embedding

1. Introduction

Semantic word representations are essential to natural language processing. Most embedding models have learned the meaning of words from their contexts. These techniques work for English, in which word forms rarely change according to their contexts. Though, in morphologically rich languages, inflectional forms of the same base are considered and learned as distinct words. Even if the base is frequent in a large corpus, each word forms rarely occurs, which leads to data sparsity. This weakness has been overcome to a certain extent by composing representation of subword units such as morphemes or characters. Character-based models can be generalized in a sense that they do not require a pre-trained morphological analyzer, and they enable to calculate vector representations even for out-of-vocabulary words.

However, vector representations of characters present the same problem as those of words, due to a trade-off between vocabulary size and token frequency. Alphabet systems, like Roman alphabets, contain a comparatively small number of symbols that occur frequently enough in a corpus. On the contrary, syllabaries have a more extensive inventory of syllable characters each of whose frequency is lower, making an analogy with morphologically rich languages.

Our objective in this paper is to address the issue of data sparsity, which we argue is caused by not only the morphological system of a language and but also its writing system. We begin by remarking on the morphological typology and diversity of writing systems. We then report two phenomena of languages that are agglutinative, represented by non-alphabetic scripts, or both, Korean as a case study. We find that they cannot be captured by the default method to code Korean text on the syllable-character level. From this, we propose a character-internal or grapheme-level coding procedure for neural word embedding that utilizes word-internal features that are composed of characters. We end by observing that our grapheme-level model is capable of representing functional and semantic similarities, grouping allomorphs, and disambiguating homographs. We conclude that it can improve existing character-level approaches to Korean NLP tasks.

2. Challenges in character-based approaches to word embeddings for morphologically rich languages

2.1. Scalability and multidimensionality of richness

2.1.1. Richness in morphology

With the distribution hypothesis(Harris, 1954; Sahlgren, 2008) as their basis, most approaches to word vector representation use co-occurrence statistics. In order for frequencies of word co-occurrence to be counted efficiently and accurately, each word should have a form that is invariant with respect to its location in a sentence. This is not likely to be the case in morphologically rich languages, however. The resulting data sparsity of morphologically rich languages has been long discussed and examined in relation to a variety of NLP tasks. As far as is known, languages that have been categorized as morphologically rich have been non-English European languages such as German and Czech, with languages such as Turkish and Arabic occasionally being included as well. Character-based approaches have been applied to improve the performance of word embeddings(Ling et al., 2015; Kim et al., 2016), and are acknowledged as non-language-specific and not requiring linguistic knowledge, however these approaches were used comparatively less frequently with other morphologically rich languages, such as Japanese and Korean. In order to shed some light on the matter, recall that it is widely held among linguists that languages are classified into more than two morphological types. Comrie (1989), for example, distinguishes between fusional and agglutinative languages:

- *Isolating or analytic languages* have no or little inflectional morphology. Since a lemma does not have various word forms and word order plays a crucial role to determine grammatical features, N-gram models work well.

- *Fusional or inflected languages* have grammatical affixes, each of which encodes multiple grammatical features simultaneously. Typically, a noun declension includes gender, number, and case, and a verb conjugation includes tense, aspect, and voice. A Lemma has multiple word forms, but their number is restricted by the number of declension/inflection morpheme types. In Latin, a verb can have a maximum of 144 inflected forms(Schinke et al., 1996). This number does not seem so small, but would be reduced in its descendants, the Romance languages including French, Spanish, Portuguese, and Italian.

- *Agglutinating languages* also have a variety of grammatical morphemes, but each of them has no more than one feature, and more than two morphemes can be concatenated to a stem.[1] Therefore, the set of possible word forms of a stem is larger than in inflected languages. For instance, Karlsson (1986) reported that a Finnish noun can be inflected into 2,200 forms, and Argüelles and Kim (2004) introduced 623 (non-exhausted) forms of each Korean verb conjugation.

Though both inflectional and agglutinating languages are morphologically richer than an isolating language, the complexity of the latter should not be overlooked.
The examples in Table 1 show phrases consisting of more than four words in English that are translated into just one word in Korean, which consists of five morphemes: a noun stem 고양이 (*ko.yang.i*)[2] 'cat' has a masculinity prefix 수 (*swu*), a plural suffix 들 (*tul*) '-s', a dative marker 에게 (*ey.key*) 'to', and finally an emphasis marker 조차 (*co.cha*) 'even'. Also note that agglutination can occur beyond the noun declension or verb conjugation that can be found in fusional languages such as Portuguese. In English, the case and gender of the noun *cat* is realized lexically(*to* and *male*), and its number, morphologically(*-s*). Even if these three features can be encoded in a pure inflectional language such as Latin, the adverb *even*, which is not a part of declension, cannot be morphologically attached to the noun head. However, agglutination can encode features that would not have been realized morphologically but have become lexical items like adverbs, conjunctions, and light verbs.

2.1.2. Richness in writing system

By comparing an agglutinating language with a fusional one, we saw that the morphological richness of a language is scalable rather than binary. Now we turn our attention to another kind of richness that can be observed in natural language processing.
Since we are dealing with writing systems that are in the form of text, it is reasonable to take account of writing systems. As a language can be morphologically rich, a writing system, or script, can be graphonomically rich. These two properties do not depend on one another. The Roman alphabet, for example, is used in a variety of languages that span across a wide range of morphological richness: isolated languages, such as English and Vietnamese, inflectional languages, such as French and Polish, and agglutinative lanugages, such as Basque and Hungarian.
The amount of information that a character can encode varies according to the writing system to which it belongs, as shown in Table 2. For example, a non-alphabetic character may not be the smallest unit of a writing system. From this, we can dempose a character into a greater number of sub-components just as words are processed as sequences of characters in various applications of NLP.

Analytic	EN	5	[Even] [to] [the] [male] [cats]
Fusional	PT	3	[até] [aos] [gatos]
agglutinating	KO	1	[수코양이들에게조차] (*swu.kho.yang.i.tul.ey.key.co.cha*)
Analytic	EN	4	[if] ... [had] [passed] [away]
Fusional	PT	3	[se] [tivesse] [falecido]
agglutinating	KO	1	[돌아가셨었다면] (*tol.a.ka.syess.ess.ta.myen*)

** EN: English, PT: Portuguese, KO: Korean

Table 1: Realization of grammatical features in three morphological types of languages

Writing system	What is assigned to each symbol
Logographic script	Morpheme
Logosyllabic script	Syllabic & semantic values
Syllabary	Syllable
Abjad	Consonant
Alphabet	Consonant or vowel

Table 2: Categorization of scripts(Daniels 1992)

2.2. Case study on a language both morphologically and graphonomically rich

We explore the possibility that the richness of languages used in language agnostic character-based approaches is restricted. Moving beyond the more commonly examined languages, Korean is a good case study as it is both a morphologically rich agglutinative language and graphonomically rich. The Korean script, Hangul, is alphabetic as each grapheme represents a phoneme. However, a grapheme occurs only as a component of a syllable. For example, the character 몸 (*mom*) is a syllable whose structure consists of an initial consonant ㅁ (*m*), a medial vowel ㅗ (*o*), and a final consonant ㅁ (*m*), each of which cannot be written separately in text.

2.2.1. Mismatch between morpheme boundary and character boundary

Korean poses additional challenges for word embeddings. First, some morpheme boundaries do not agree with character boundaries. One class of grammatical morphemes of high frequency includes morphemes such as ㅆ (*ss*) '-ed', ㅁ (*m*) '-ing', ㄴ (*n*) '(that) has/have ...ed', and ㄹ (*l*) '(that) will ...' that are all final consonants that are realized with

[1] Non-agglutinating languages also have multiple affixation (e.g. *un-accept-abil-ity* and *de-caffein-ate*), but it is restricted in derivation rather than inflection, so they are less productive than agglutinating languages.

[2] We use the Yale Romanization for Korean. It is a transliteration rather than a phonemic or phonological transcription, and is better suited to representing forms written in the Korean script.

different stems consisting of different syllabic characters. All words in Table 3 end with a gerund marker ㅁ (*m*), but they have no other characters in common.

잠	(*ca**m***)	'sleeping'	세움	(*sey.wu**m***)	'stopping'
뜀	(*ttwi**m***)	'jumping'	부름	(*pwu.lu**m***)	'calling'
달림	(*tal.li**m***)	'running'	마침	(*ma.chi**m***)	'finishing'

Table 3: Gerund formation by ㅁ (*m*) '-ing' in Korean

This phenomenon can be found also in Japanese. Table 4 shows that there are no characters in common between よむ (*yo.mu*) 'read' and いく (*i.ku*) 'go', even though they belong to the same grammatical category.

Pres. ind.		Volitional		Imperative		
いく	(*i.k**u***)	いこう	(*i.k**o.u***)	いけ	(*i.k**e***)	'go'
とぶ	(*to.b**u***)	とぼう	(*to.b**o.u***)	とべ	(*to.b**e***)	'fly'
よむ	(*yo.m**u***)	よもう	(*yo.m**o.u***)	よめ	(*yo.m**e***)	'read'

Table 4: Examples of Japanese verb conjugations

2.2.2. Variety of allomorphs

The second challenge posed by the Korean is a result of its large variety of allomorphs. At the character level, two allomorphs of the same morpheme do not overlap each other. For words of Chinese origin [3] many characters have multiple readings, which are differentiated only by an initial, as exemplified in Table 5.

率	(***l**ywul*)	확**률** 確**率**	(*hwak.**lywul***)	'probability'
	(*ywul*)	비**율** 比**率**	(*pi.**ywul***)	'ratio'
樂	(***l**ak*)	쾌**락** 快**樂**	(*khoey.**lak***)	'pleasure'
	(*nak*)	**낙**관 **樂**觀	(***nak**.kwan*)	'optimism'

Table 5: Multiple readings of Chinese characters in Korean

Most frequent noun case markers and verb tense markers are also allophonic. For instance, two accusative case markers 을 (*ul*) and 를 (*lul*) are distincted by an initial ㅇ (∅) or ㄹ (*l*), and past tense markers 았 (*ass*) and 었 (*ess*), by a medial ㅏ (*a*) or ㅓ (*e*).

3. Word-internal and Character-internal Models

Building upon the case study of the previous section, we propose a character-internal coding procedure for word embeddings that utilizes word-internal features that are composed of characters. We use the character-based CNN-LSTM language model proposed by Kim et al. (2016)[4], and apply several methods to decompose characters into more finely grained units. In Korean text, the most superficial characters represent syllables; they consist of graphemes. We have observed that the contrasting unit is not a grapheme alone, but a grapheme along with a fixed position. For example, verbs of the same inflectional type share a final consonant, and allomorphs sharing a Chinese origin are distinguished by their initital consonants. This leaves us with two options when considering how to approach the decomposition of syllables.

Graphemes with Initial-Medial-Final distinction The syllable 몸 (*mom*) has three grapheme types ᄆ (Unicode: U+1106)(*m*), ᅩ (U+1169)(*o*), and ㅁ (U+11B7)(*m*).

Graphemes with Consonant-Vowel distinction 몸 (*mom*) consists of two ㅁ (U+3141)(*m*)'s and one ㅗ (U+3157)(*o*).

Syllables as characters 몸 (*mom*), whose unicode designation is U+BAB8, is regarded as a single character with no internal structure.

4. Language Modelling

4.1. Experiments

4.1.1. Datasets

To learn our language model, we utilize the Korean Sejong Corpus(The National Institute of the Korean Language, 2010). Though the corpus contains morphologically and syntactically parsed texts, but we use just raw texts to make our models work without morphological knowledge. During preprocessing, we categorize words with a frequency of less than 5 instance as `<unk>`, and replace characters from the Chinese script and Roman alphabet with H and R, respectively. Digits are replaced with N. Our training set consists of 1,562,497 word tokens, 102,532 word types, 1,766 syllable types, and 164 grapheme types (with IMF distinction).

4.1.2. Setup

In order to determine a baseline, we compare three character-level CNN models (GraphIMF, GraphCV, and Syl) and two word-level Word2Vec models (Word-CBOW and Word-Skip) as baseline. For CNN models we borrow the Char-CNN algorithm(Kim et al., 2016).

We use a default parameter setting of each architecture except for Syl-CNN. Kim et al. (2016) set the character embedding dimension as $d = 15$ and the filter numbers as $h = 1, 2, \ldots, 7$, but we modify them into $d = 150$ and $h = 1, 2, \ldots, 4$ for our Korean syllable vocabulary of size 1,766.

4.2. Inspection on Learned Vector Spaces

4.2.1. Visualizing Vector Spaces

To survey our results in Figure 1, we visualize vectors representing the most frequent 1,000 words for each model using t-Distributed Stochastic Neighbor Embedding(Van Der Maaten and Hinton, 2008). If each vector space reflects similarity between words properly, similar word vectors will form a cluster and different word vectors will be distant from each other.

In the first two word-level models, most vectors are scattered. We see small clusters on the periphery, but they are restricted to inflections of verb stems such as 있 (*iss*) 'exist'

[3] In spite of etymology, these words are almost always written in the Korean script, not Chinese characters.

[4] `https://github.com/yoonkim/lstm-char-cnn`

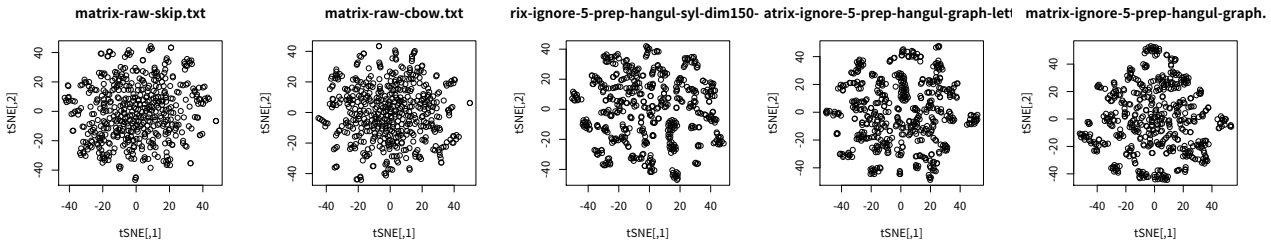


Figure 1: t-SNE visualization (Word-Skip, Word-CBOW, Syl-CNN, GraphCV-CNN, and GraphIMF-CNN)

and 않(*anh*) 'not', whose frequency is very high and whose meaning is functional rather than lexical.

The other three character-level models reveal clusters more clearly. Some of the clusters represent noun and pronoun cases, which each share a case marker: nominative 이(*i*) or 가(*ka*), accusative 을(*ul*) or 를(*lul*), locative 에(*ey*) and so on. Verbs are also clustered with respect to markers like a sentence final 다(*ta*) and relative clause 는(*nun*). We also note that our GraphIMF-CNN model gathers some negative polarity items including 아무(*a.mwu*) 'any', 결코(*kyel.kho*) 'ever', and 전혀(*cen.hye*) 'at all'.

4.2.2. Clustering Vectors

By visualizing vector spaces, we may be able to find salient groupings of words. To expand our observation to the rest, we identify what words each cluster consists of and how homogeneous it is using the K-means clustering algorithm(Hartigan and Wong, 1979).

GraphIMF-CNN	GraphCV-CNN	Syl-CNN
1. N(cit.)	1. N(cit.)	1. N(cit.)
2. N(nom.)	2. N(nom.)	2. N(cit.), Demonstratives, A and V with connectivs
3. N(acc.)	3. N(acc.)	3. N(nom.)
4. N(loc., top., ...)	4. N(gen.)	4. N(acc.)
5. N(top.), V(rel.)	5. N(top.), V(rel.)	5. N(gen.) V(rel.) A(att.)
6. A(att.), V(rel.)	6. A(rel.), V(rel.)	6. N(top.) V(rel.)
7. Adverbs	7. Nominal Adverbials??	7. Nominal and verbal adverbials??
8. Nominal Adverbials??	8. Adverbs and V with connectives	8. Adverbs?
9. V with connectives	9. V with connectives	9. V(rel.) V with connectives
10. Sentence finals	10. Sentence finals	10. Sentence finals

The symbol ?? means that a marked cluster is noisy and dubious.

Table 6: 10-means clustering over the most frequent 1,000 words

The results with $K = 10$ for three character-level models are shown in Table 6. Even though we started with texts without part-of-speech tagging, we can see that the clusters we have generated largely overlap with syntactic categories. Using word-level models, on the other hand, we cannot find a common property among words in the same cluster. The least heterogeneous cluster might be tagged as Medical because 62 out of 151 words, such as 근육(*kun.ywuk*) 'muscle(citation)', 장애가(*cang.ay.ka*) 'disability(nominative)', 질환을(*cil.hwan.ul*) 'disease(accusative)', are relative with the topic. The other 89 words include more generic terms and conjunctions like 변화(*pyen.hwa*) 'change(citation)' and 또는(*tto.nun*) 'or'. Setting $K = 30$, citation forms of nouns are subcategorized into six clusters in the GraphIMF-CNN model, three clusters in the GraphCV-CNN model, and five clusters in the Syl-CNN model. We can manually cluster in the GraphIMF-CNN model as in Table 7, which generates results that are less noisy than those generated by the other models.

Citation forms of nouns
Concrete and basic morning, water, tree, road, room, rice, book, fruit, ...
Pronouns and personal he, I, grandmother, mom, uncle, mother, dad, child, author, ...
Collective Korea, USA, Japan, China, North Korea, Our country, France, party, Western, humankind, ...
Abstract (1) kind, history, literature , culture, market, cinema, environment, development, ...
Abstract (2) public, movement, school, society, liberation, economy, ...
Temporal year, minute, afternoon, this year, next year, every year, one year, spring, ...

We translated all examples from Korean to English.

Table 7: Part of 30-means clustering from the GraphIMF-CNN model

4.3. Discussion

We examine the nearest neighbors of words using cosine similarity. We select words representing the characteristics of morphologically rich languages such as those we mentioned in Section 2.

4.3.1. The same morpheme in different syllables

As we saw in Table 3, a non-syllabic morpheme can appear in syllables of various forms. Two such examples are the final consonant -ㄴ (*-n*) '(that) -' and the medial vowel -ㅕ (*-e*) '-ing.' They combine with a stem 기쁘(*ki.ppu*) 'be glad' and make the words 기쁜(*ki.ppun*) '(that) is/are glad' and 기뻐(*ki.ppe*) 'being glad' respectively. Table 8 shows what each model predicts as the most similar words to these two targets.

For 기쁜(*ki.ppun*) '(that) is/are glad', at least 4 out of 5 words have the right morpheme ㄴ (*n*) in all of the models, on both the character- and word- level. One error in the Syl-CNN model is 홈런(*hom.ren*) 'home run', which is morphologically totally different from the target. Moreover, the sixth nearest neighbor is 타이슨(*tha.i.sun*) 'Tyson.' The syllable-level model's prediction wrongly includes loanwords that end with ㄴ (*n*).

	Graphemes (IMF)		Graphemes (CV)		Syllable characters		Word2Vec (CBOW)		Word2Vec (Skip-gram)	
기쁜	슬픈	'sad'	슬픈	'sad'	어설픈	'poor'	하자는	'suggesting to do'	바쁜	'busy'
'(that) is/are glad'	즐거운	'pleasant'	가쁜	'breathless'	홈런	'home run'	억울한	'suffering'	반가운	'welcomed'
	무서운	'dreadful'	힘든	'hard'	슬픈	'sad'	힘든	'hard'	슬픈	'sad'
	가쁜	'breathless'	궂은	'sad'	즐거운	'pleasant'	정작	'but ... actually'	못된	'wicked'
	나쁜	'bad'	옳은	'right'	쉬운	'easy'	그땐	'then'	간다는	'insisting on going'
기뻐	써서	'writing'	추워서	'cold'	꽈르릉	(onomatopoeia)	보아서	'looking'	평상시와	'usual'
'being glad'	꺼내서	'taking out'	주워서	'picking up'	아슬아슬	(memetic word)	결단을	'decision(acc.)'	부러워하는	'envicious'
	갈아서	'changing'	꺼내서	'taking out'	쟁그랑	(onomatopoeia)	아내도	'also a wife'	잊어버리기	'to forget'
	떠서	'lifting'	아까워서	'sparing'	까악까악	(onomatopoeia)	당부를	'request(acc.)'	하라니까	'I told you, do it!'
	장가가서	'marrying'	누워서	'lying'	소리니	'Is it a sound?'	않았소	'did not(indicative)'	자기한테	'to oneself'

** Bold face denotes success in finding similarity, and an underline denotes a notable error.

Table 8: Nearest neighbors of the targets 기쁜 (*ki.ppun*) '(that) is/are glad' and 기뻐 (*ki.ppe*) 'being glad' respectively

The results of the latter 기뻐 (*ki.ppe*) 'being glad' get worse in the Syl-CNN, Word-CBOW, and Word-Skip. Only GraphIMF-CNN and GraphCV-CNN find all true neighbors. We also note that the first four candidates from the Syl-CNN model are all onomatopoeic. This is unexpected as the syllables that make up 기뻐 (*ki.ppe*) 'being glad' are rarely used in onomatopoeic words.

The results so far suggest that grapheme-level granularity captures morphological (and then functional) similarity more effectively than syllable-level and word-level granularity.

4.3.2. Sino-Korean polyphony and homophony

We also indicated that multiple Korean readings of Sino-Korean characters become a challenge for character-aware word embeddings. For example, in the structure of a word 확률 (*hwak-lywul*) 'probability', 률 (率) *lywul* is analyzed as a head but it can appear as 율 (*ywul*) in other words. Table 9 shows that each model predicts as the nearest neighbors of the nominative and accusative forms of the stem.

In the GraphIMF-CNN model, the first four for nominative case and the first three for accusative case have the target syllable 률 (*lywul*) or its allomorph 율 (*ywul*). The GraphCV-CNN and Syl-CNN models find one or two 율 (*ywul*) or 률 (*lywul*) and their results are less satisfactory both in quality and quantity. They include "false friends" such as 규율이 (*kywu.ywul.i*) 'discipline(nominative)' and 운율은 (*wun.ywul.un*) 'rhyme(topic)', where 율 (*ywul*) is not 率 but its homomorph 律. Moreover, 효율 (*hyo.ywul*) 'efficiency' is the only 率-type word that the Syl-CNN model found, but its meaning is less similar with other words of form X率 'rate or probability of X.' This observation indicates that graphemes with IMF distinction capture semantic similarity represented by formal similarity better than other units and they also and they also clarify ambiguities caused by homonymy and polysemy.

5. Related Work

5.1. Character-level Approaches to Word Embeddings

Our work is based on studies of constructing vector representations of words through their constituent characters. Ling et al. (2015) first introduced this model by composing characters into representations of words using bidirectional long-short term memory networks, and then looked at language modeling and part-of-speech tagging performance on English, Portuguese, Catalan, German and Turkish. Kim et al. (2016) used convolutional neural networks and highway layers to capture morphological similarities and semantic similarities between words, and claimed that their language model is language non-specific. Their model outperformed word and morpheme-level baseline performance on Arabic, Czech, French, German, Spanish, and Russian, which are morphologically richer than English.

5.2. Subword information on Chinese

The importance of characters has been a focus for approaches dealing with Chinese text, a language in which characters are written without spaces (Sproat et al., 1996). Because each Chinese character has an internal structure consisting of radicals, units that are more fine-grained than characters have been exploited in research utilizing word embeddings(Yin et al., 2016; Xu et al., 2016) and character embeddings(Sun et al., 2014; Li et al., 2015; Peng and Cambria, 2017).

5.3. Korean Word Embeddings

To the best of our knowledge, in most studies on Korean word embeddings, characters were coded at the syllable level.Cinarel and Zhang (2016) incorporated a word with syllable N-grams using a subword model proposed by Bojanowski et al. (2017). Choi (2017) used convolutional neural networks to extract a word vector from syllable vectors and a skip-gram model to learn the distribution of each word. Choi evaluated the model by measuring training losses and cosine similarities on WS353(Finkelstein et al., 2001), and implementing out-of-vocabulary tests on a several nonce words and morphology tests on two types of noun forms.

Yu and Ko (2017) added a syllable embedding to a word representation as an input of bidirectional LSTM CRF models for Korean named entity recognition. As in our grapheme model the nearest neighbors of the word 마르크스 *ma.lu.khu.su* 'Marx' consist mostly of loanwords from German. We can therefore expect that a grapheme-level approach would capture a typical phonotactic pattern of proper names better than a syllable-level approach does in NER tasks.

5.4. Adoptation of graphemes for Korean

Choi et al. (2016) adopted a grapheme-level approach for constructing a Korean morphological analyzer. They did not specify from which unicode block their morphemes were encoded, but we deduce that they used CV distinction.

	Graphemes (IMF)		Graphemes (CV)		Syllable characters		Word2Vec (CBOW)		Word2Vec (Skip-gram)	
확률이	환**율**이	'exchange rate'	환**율**이	'exchange rate'	효**율**이	'efficiency'	평상시와	'usual'	저렴하게	'cheaply'
'probability'	투표**율**이	'voter turnout'	바벨탑이	'Tower of Babel'	규율이	'discipline'	부러워하는	'envious'	음미될	'to be appreciate'
(nominative)	성장**률**이	'growth rate'	용수량이	'water capacity'	확산이	'diffusion'	잊어버리기	'to forget'	말해볼	'to try telling'
	손해**율**이	'loss rate'	예측이	'expectation'	수법이	'method'	하라니까	'I told you, to it!'	정도라면	'if (it is) that much'
	진덕왕이	'Queen Jindeok'	성장**률**이	'growth rate'	변증법이	'dialectic'	자기한테	'to oneself'	높다고	'(say that it is) high'
확률을	수익**률**을	'profit rate'	환**율**을	'excange rate'	품격을	'dignity'	일치할	'to coincide'	참여는	'participation(topic)'
'probability'	합격**률**을	'acceptance rate'	운율을	'rhyme'	징역을	'imprisonment'	빠져들	'to sink into'	아래에서는	'under(topic)'
(accusative)	성장**률**을	'growth rate'	경영을	'management'	변별력을	'discrimination'	참조할	'to refer to'	가시적인	'visible'
	염병을	'epidemic'	관우를	'Guan Yu'	효**율**을	'efficiency'	규정될	'to be defiend'	허용될	'to be allowed'
	원형을	'prototype'	집행을	'execution'	경계심을	'wariness'	제시할	'to suggest'	완벽에	'to perfection'

** Bold face denotes success in finding similarity, and an underline denotes a notable error.

Table 9: Nearest neighbors of the targets 확률이 (*hwak.lywul.i*) 'probability(nominative)' and 확률을 (*hwak.lywul.ul*) 'probability(accusative)' respectively

They reported that the non-CRF version of their model had misanalyzed 차우세스쿠 (*cha.wu.se.su.kwu*) 'Ceauşescu', 창 (*chang*) and ㅜ세스크 (*wu.se.su.ku*). From Table 10, we note that this error could have been avoided by IMF distinction, in which 창 (*chang*) cannot be analyzed as a substring of 차우세스쿠 (*cha.wu.se.su.kwu*), and ㅜ세스크 (*wu.se.su.ku*) is not a valid string.

Consonant-Vowel distinction	
차우세스쿠	[ㅊㅏㅇ]ㅜㅅㅔㅅㅡㅋㅜ
(*cha.wu.se.su.kwu*)	*U+314A 314F 3147* 315C 3145 3154 ...
창	[ㅊㅏㅇ]
(*chang*)	*U+314A 314F 3147*
Initial-Medial-Final distinction	
차우세스쿠	[ᄎ ᅡ] ᄋ ᅮ ᄉ ᅦ ᄉ ᅳ ᄏ ᅮ
(*cha.wu.se.su.kwu*)	*U+110E 1161* **110B** 116E 1109 1166 ...
창	[ᄎ ᅡ] ᄋ
(*chang*)	*U+110E 1161* **11BC**

Table 10: Different substring relations in different grapheme decomposition methods

6. Conclusion

We have presented a grapheme-level approach to character-aware word embeddings for languages represented by syllabaries. We merely decompose syllable characters into graphemes in the framework of current neural word representations, without the cost of morphological analysis. Our grapheme-level model disambiguates homographs and finds both functional and semantic similarities among allomorphs of the same morpheme, which syllable-level and word-level models fail to capture. These results show that our model represents various word forms well even in the complex morphological system of agglutinative languages. We believe that our results apply to other languages with different morphological systems and writing systems.

Acknowledgement

This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2017S1A5A2A01023336).

7. Bibliographical References

Argüelles, A. and Kim, J.-R. (2004). *A Handbook of Korean Verbal Conjugation*. Dunwoody Press.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Choi, J., Youn, J., and Lee, S.-g. (2016). A grapheme-level approach for constructing a Korean morphological analyzer without linguistic knowledge. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 3872–3879. IEEE, dec.

Choi, S.-H. (2017). The modeling and training methods for syllable-based korean word embeddings. Master's thesis, Seoul National University.

Cinarel, C. and Zhang, B.-T. (2016). Better Word Embeddings for Korean. In *Proceedings of the Korea Information Science Society Winter Conference*, pages 627–629.

Comrie, B. (1989). *Language universals and linguistic typology: Syntax and morphology*. University of Chicago press.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

Karlsson, F. (1986). Frequency considerations in morphology. *STUF-Language Typology and Universals*, 39(1-4):19–28.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2016). Character-Aware Neural Language Models. *AAAI*, pages 2741–2749.

Li, Y., Li, W., Sun, F., and Li, S. (2015). Component-Enhanced Chinese Character Embeddings. *Emnlp*, (September):829–834.

Ling, W., Luís, T., Marujo, L., Astudillo, R. F., Amir, S., Dyer, C., Black, A. W., and Trancoso, I. (2015). Finding Function in Form: Compositional Charac-

ter Models for Open Vocabulary Word Representation. (September):1520–1530.

Peng, H. and Cambria, E. (2017). Radical-Based Hierarchical Embeddings for Chinese Sentiment Analysis at Sentence Level. In *The Thirtieth International Florida Artificial Intelligence Research Society conference*.

Sahlgren, M. (2008). The distributional hypothesis. *Italian Journal of Disability Studies*, 20:33–53.

Schinke, R., Greengrass, M., Robertson, A. M., and Willett, P. (1996). A Stemming Algorithm for Latin Text Databases. *Journal of Documentation*, 52(2):172–187.

Sproat, R., Gale, W., Shih, C., and Chang, N. (1996). A stochastic finite-state word-segmentation algorithm for chinese. *Computational linguistics*, 22(3):377–404.

Sun, Y., Lin, L., Tang, D., Yang, N., Ji, Z., and Wang, X. (2014). Radical-Enhanced Chinese Character Embedding. In *International Conference on Neural Information Processing*, pages 279–286.

The National Institute of the Korean Language. (2010). 21st Century Sejong Project.

Van Der Maaten, L. J. P. and Hinton, G. E. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Xu, J., Liu, J., Zhang, L., Li, Z., and Chen, H. (2016). Improve Chinese Word Embeddings by Exploiting Internal Structure. *Naacl*, pages 1041–1050.

Yin, R., Wang, Q., Li, R., Li, P., and Wang, B. (2016). Multi-Granularity Chinese Word Embedding. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-16)*, pages 981–986.

Yu, H. and Ko, Y. (2017). Expansion of Word Representation for Named Entity Recognition Based on Bidirectional LSTM CRFs. *Journal of Korean Institute of Information Sciencentist and Engineers*, 44:306–313.

Building a Constraint Grammar Parser for Plains Cree Verbs and Arguments

Katherine Schmirler[1], Antti Arppe[1], Trond Trosterud[2], Lene Antonsen[2]
[1]University of Alberta, [2]UIT Arctic University of Norway
{schmirle, arppe}@ualberta.ca, {trond.trosterud, lene.antonsen}@uit.no

Abstract

This paper discusses the development and application of a Constraint Grammar parser for the Plains Cree language. The focus of this parser is the identification of relationships between verbs and arguments. The rich morphology and non-configurational syntax of Plains Cree make it an excellent candidate for the application of a Constraint Grammar parser, which is comprised of sets of constraints with two aims: 1) the disambiguation of ambiguous word forms, and 2) the mapping of syntactic relationships between word forms on the basis of morphological features and sentential context. Syntactic modelling of verb and argument relationships in Plains Cree is demonstrated to be a straightforward process, though various semantic and pragmatic features should improve the current parser considerably. When applied to even a relatively small corpus of Plains Cree, the Constraint Grammar parser allows for the identification of common word order patterns and for relationships between word order and information structure to become apparent.

Keywords: Plains Cree, Constraint Grammar, automatic parser

1. Introduction

This paper presents the first developmental stage of a Constraint Grammar (CG) parser for Plains Cree, a North American Indigenous language spoken by several thousands of people primarily in the Canadian provinces of Alberta and Saskatchewan. It is a member of the Algonquian language family and the westernmost language of the Cree-Montagnais-Naskapi continuum spoken across much of Canada. A number of texts have been published in Plains Cree, totalling several hundreds of thousands of words; while these comprise a small corpus compared to many available for languages such as English, they do allow for the development and testing of computational tools such as morphological and syntactic models. Once such tools have been developed, they can be used not only for corpus investigations, but can be implemented in various language technological applications for speakers, teachers, and students of Plains Cree. These include morphologically "intelligent" online dictionaries, morphosyntactically tagged corpora, spell checkers, grammar checkers, and intelligent computer-assisted language learning (ICALL) applications.

1.1 Research Questions

The present research aims to answer three questions. First, to what extent can basic syntactic relationships in Plains Cree be adequately modelled using only morphological information? Second, of those syntactic relationships that cannot be identified, how can they be accounted for in future development? Third, when applied to a corpus of Plains Cree, what word order patterns can be identified?

2. Plains Cree Morphosyntax

Like all Algonquian languages, Plains Cree is highly polysynthetic with a complex morphological system. The morphosyntactic features and inflectional system of Plains Cree are modelled by means of a morphological analyser (Snoek et al, 2014; Harrigan et al., 2017), the output of which includes morphosyntactic feature tags and all possible analyses of ambiguous forms. These analyses become the input for the Constraint Grammar parser. The features discussed in this section are tagged in the output of the morphological analyser and are referenced in the constraints specified in the current CG parser.

2.1 Nominal and Verbal Features

2.1.1 Nominal Features

Plains Cree nouns are classified in terms of animacy, with two genders, animate and inanimate. All words for people, animals, and trees are animate, while most other forms are inanimate. However, there are many words such as *asikan* 'sock' or *kôna* 'snow'[1] that are semantically inanimate but behave as animate nouns in Plains Cree. In this way, it is truly a grammatical distinction rather than a purely semantic one (e.g. Ahenakew, 1987). Sex-based gender distinctions are not made in Cree (e.g. *wiya* 'he, she, it [animate]'). Animate nouns are also inflected for obviation, a pragmatic category that distinguishes between more topical (proximate) and less topical (obviative) animate third persons. Plains Cree obviative third persons are marked with the suffix *-a* and do not distinguish number (Wolfart, 1973). Obviation occurs, and overt marking is required, whenever more than one animate third person participant is present in a discourse. This includes one animate third person acting on another, but also when one animate third person possesses another.

Obviation can be considered part of a larger person hierarchy, which is particularly relevant in the discussion of transitive verbs below. In this hierarchy, animates are ranked over inanimates, proximate over obviative, and speech act participants (local) over non-speech act participants (non-local). Additionally, second person is ranked over first person. The person hierarchy is visualized in (1).

(1) Algonquian person hierarchy (adapted from Wolvengrey, 2011, p. 57)

2 > 1 >> 3 > 3′ >> 0

Both nouns and demonstrative pronouns may stand alone as the arguments of verbs, though nouns and demonstrative pronouns may also co-occur. However, due to extensive

[1] Circumflexes or macrons are used to mark long vowels in written Plains Cree; all Cree forms throughout this manuscript are written using the Standard Roman Orthography.

verbal morphology (discussed below), it is more common that no overt arguments are present.

2.1.2 Verbal Features

Plains Cree verbs are classified by their transitivity and by the animacy of their participants. In the context of parsing, these classes allow the CG to determine how many participants (transitivity) of which noun classes (animacy) can potentially be associated with a given verb. Following Algonquianist tradition, verbal arguments are referred to as *actors* and *goals*, rather than as *subjects* and *objects*.[2] There are four classes, inanimate intransitive (VII), animate intransitive (VAI), transitive inanimate (VTI), and transitive animate (VTA). The intransitive classes are each marked for one participant, VIIs may take an inanimate actor and VAIs an animate actor. VTIs are also only marked for one animate actor, but an inanimate goal may be syntactically present. VTAs present the most complex person morphology, as two animate participants are marked on the verb. Furthermore, Plains Cree makes use of a direct-inverse system, where, rather than coding for actor/subject or goal/object (e.g. as done in case-marking languages), the direction of an action is marked on the verb. The person and number marking does not mark the role of a participant; instead, the direction theme sign indicates which is the actor and which is the goal (Wolvengrey, 2011, p. 173-6). Direction is determined by means of the person hierarchy (above): direct morphology occurs when a more topical participant acts on a less topical one and inverse morphology occurs in the opposite situation. Examples are given in (2).

(2) Plains Cree (Wolvengrey, 2011, p. 175)

a. *câniy kî-wîcihêw mêrîwa*
   câniy kî-wîcih-**ê**-w mêrî-wa
   John.PROX PST-help.VTA-**DIR**-3SG Mary-OBV
   'Johnny helped Mary.'

b. *câniy kî-wîcihik mêrîwa*
   câniy kî-wîcih-**ik(w)**-(w) mêrî-wa
   John.PROX PST-help.VTA-**INV**-3SG Mary-OBV
   'Mary helped Johnny.'

Sentences like these also demonstrate the key way in which proximate and obviative marking are unlike grammatical cases. In Plains Cree, regardless of the semantic or syntactic role of an argument, the nominal marking remains unchanged while the roles are indicated by direction morphology and the relative topicality of the arguments.

2.2 Syntactic relationships

As Plains Cree is a non-configurational language, word order is not used to determine syntactic relationships such as actor and goal, as demonstrated in the above examples. When determining relationships between nouns and verbs, one can rely almost entirely on the morphology of the verb and any lexicalised arguments. However, there are still some linear relationships that can be used in a CG parser. For example, though both nouns and demonstratives may occur on their own, when they occur adjacent to each other with agreeing features, they can be described as noun phrases (NPs), specifying something akin to "this/that N".

Though linear order is not used to determine syntactic relationships, pragmatic positions have been identified, e.g. topical and focused nouns generally occur before a verb (e.g. Dahlstrom, 1995). Some patterns can be seen in a corpus investigation, suggesting avenues for future research.

3. The Plains Cree Corpus

The current corpus for Plains Cree consists of narratives, dialogues, speeches, and lectures recorded by F. Ahenakew in the 1980s and 1990s. These were then transcribed, translated, and edited by F. Ahenakew and H.C. Wolfart and are available in several published volumes (Ahenakew, 2000; Bear et al., 1992; Kâ-Nîpitêhtêw, 1998; Masuskapoe, 2010; Minde, 1997; Vandall & Douquette, 1987; Whitecalf, 1993). Together, these total 108,413 tokens (18,649 types), of these, 73,189 tokens (15,994 types) are identifiable Plains Cree words.

This corpus has been evaluated using a morphological analyser for Plains Cree, the first versions of which are described in Snoek et al. (2014) and Harrigan et al. (2017). After the initial analysis was performed, the corpus was hand-verified, correcting erroneous analyses and adding analyses that the analyser was unable to produce. This has served to both identify areas for improvement in the morphological model and, more importantly for the present purpose, to include as many correct analyses as possible for use as input to the syntactic parser. For further information on the annotation process, see Harrigan et al. (2017). The hand-verified corpus is referred to as the morphological Gold Standard. Using the morphological Gold Standard, the CG parser described below was tested using a smaller portion of this corpus, hand-coded for basic syntactic relationships. This is referred to as the syntactic Gold Standard. The online version of the corpus uses the Korp interface, based on the open-source tools in the IMS Open Corpus Workbench (Evert & Hardie, 2011; Borin et al., 2012) and can be found at http://altlab.ualberta.ca/korp. The corpus resources, including the texts, the model coverage compared to the hand-verified Gold Standards, and the online corpus interface and its search capabilities are discussed in detail in Arppe et al. (2017).

Additional Plains Cree texts, both historical (e.g. Bloomfield, 1930, 1934; Demers et al., 2010) and modern (e.g. Wolvengrey 2007, a number of children's books, etc.) are available for future inclusion in the corpus. Further development of the morphological analyser to model archaic morphological features is also underway to more thoroughly analyse historical texts.

4. Constraint Grammar

A Constraint Grammar (or Constraint Grammar Parser) is a list of descriptive, context-based constraints designed to parse natural language. The constraints disambiguate forms using morphological and lexical information to output a single surface morphosyntactic reading for each utterance. Unlike a generative approach to grammar, no input is considered incorrect or ungrammatical; every input is instead analysed as best as the parser allows, regardless of

[2] *Actor* and *goal* do not correspond to semantic roles, however, as they can be agents, experiencers, etc.

its grammaticality (Karlsson, 1990). The fundamental underpinnings of Constraint Grammar are the rules of a language that a linguist can easily identify as categorical, and do not need to be learned as tendencies; the constraints can then be tested against a corpus of natural language to determine how accurate the rules identified by linguists are (F. Karlsson, pers. comm. to A. Arppe).

The input for the CG is morphological analysis, such as that returned by a morphological analyser. The morphological analyses offered for each word form are the *readings* in its *cohort*. The CG then has two main goals: 1) the disambiguation of forms with a cohort of more than one morphological analysis and 2) the assignment of syntactic functions (e.g. dependencies) using the sentential context of each word form. CG can delineate the range in which to look for dependencies by referencing clause boundary punctuation in the text, such as periods, question marks, exclamation points, commas, colons, quotation marks, etc.

Constraints are used to narrow down (disambiguate) the relationships between words in a sentence to return an analysis (Karlsson, 1995a). The constraints in CG are of several different types: 1) constraints that disambiguate based on the context, 2) constraints that map the clause boundaries using punctuation and capitalization, and 3) constraints that map the syntactic functions of word forms. Alongside context-based constraints, heuristic-based parsing can also be used to improve the analysis. Heuristic constraints may be used to disambiguate where context cannot, such as choosing a reading because it was contextually selected previously, or to simplify issues by ignoring constraints and enforcing analyses. They can also be purely probabilistic and choose the analysis that is the most likely based on prior quantitative analysis (Karlsson, 1995b). Though not yet widely implemented for Plains Cree, heuristic constraints would prove useful for a number of frequent ambiguous forms for which context is not sufficient for disambiguation (see below).

The CG formalism for the current Plains Cree parser uses the VISLCG-3 compiler (e.g. Bick & Didrikson, 2015; documentation can be found at http://visl.sdu.dk/cg3.html). This newer compiler includes various capabilities not available in earlier versions, such as the use of regular expressions in constraints, the ability to specify relationships between constraints, easier control over the scope of parameters, the chunking of heads and modifiers, and the identification of dependencies between objects and their complements or between anaphoric or discourse relationships. Of these, only regular expressions are currently used in the Plains Cree CG parser, though other capabilities will be used as development of the parser progresses. Additionally, the current formalism also allows for the implementation of lexicosemantic information that can serve to refine the constraints further and better represent syntactic relationships; adding lexicosemantic information is planned for the immediate future.

Though an admittedly rather archaic technology, the CG formalism offers several advantages for modelling the syntax of languages such as Plains Cree. One, the rich agreement morphology of Plains Cree lends itself readily to the identification of arguments within the non-configurational syntax, a well-known capability of the formalism. Two, the implementation of categorical rules identified by linguists is straightforward within the formalism, requiring very little training to compose basic constraints and begin testing. Three, these categorical rules are often sufficient to model much of the syntax, which is particularly advantageous for languages with only several tens of thousands of words available in a corpus, where stochastic modelling would be impractical; straightforward categorical rule identification also speeds the model development. Four, for understudied or endangered languages such as Plains Cree, speed of development is crucial for tools such as syntactic parsers for inclusion in applications for use by speakers such as grammar checkers and ICALL applications. It is for these reasons that Constraint Grammar has been selected for modelling Plains Cree syntax.

5. Building a Parser for Plains Cree

The current iteration of the Plains Cree parser implements 67 disambiguation constraints and 105 function constraints. The basic patterns found in the constraints, as well as their coverage of a hand-coded text of ~3,200[3] words (Vandall & Douquette, 1987) are laid out below.

5.1 Disambiguation

Twenty-seven of the 67 of the constraints used for disambiguation in the Plains Cree parser are required not because of ambiguity inherent to the language, but because of ambiguity introduced by a descriptive version of the morphological analyser, which ignores vowel length distinctions in favour of recognising as many forms as possible. For example, the verbal suffix *-yân* marks first person singular in certain verb forms, and constitutes a minimal pair with *-yan*, which marks second person singular. The descriptive analyser offers both analyses regardless of how the word is spelled, so we have written constraints that choose the one that matches the spelling, which for the current texts we deem to be accurate.[4] While the ambiguity introduced by the analyser presents a number of challenges, here we instead discuss how the constraints handle the inherent ambiguities.

Diminutive nouns present one such ambiguity. The morphological analyser may give up two analyses, one where the diminutive noun is its own lexical entry, if one is available, and one where it is derived from another lexical entry. For these situations, we have opted to choose the reading where the diminutive is its own lexical entry, as it often has slightly different semantics than simply "little X". In the following example, *nêhiyâsis* does not simply mean "small Plains Cree person" (from *nêhiyaw* 'Plains Cree person'), but 'young Plains Cree person'. We use the

[3] These ~3,200 words comprise approximately 4.5% of the current Plains Cree corpus; additional texts in a hand-coded reference corpus would be beneficial to future development of the CG parser.

[4] This solution, while workable for the present corpus, will not be sufficient in the eventual implementation of a generally applicable CG parser, as many speakers do not mark vowel length.

constraint given in (3) to disambiguate the cohort given in (4); a semicolon <;> marks the removed reading.[5]

(3) REMOVE:DerNo Der/Dim (0C N) ;

(4) "<nêhiyâsisak>" 'young Plains Cree person'
; "nêhiyaw" N AN Der/Dim N AN Pl
"nêhiyâsis" N AN Pl

Similarly, there are a number of forms that can be either proximate or obviative. These may be homophonous animate and inanimate nouns, but more frequently these are demonstrative pronouns that may be either inanimate plural or animative obviative. These are disambiguated based on their context, such as an adjacent noun with agreeing features. An exemplary constraint is given in (5).[6]

(5) REMOVE:DemANObvnotIN IN + Pl (1 N + AN + Obv) (0 Dem + AN + Obv) ;

This type of constraint applies in a case such as the following example in (6), where an ambiguous pronoun is identified as obviative when adjacent to an obviative noun.

(6) "<ôhi>" 'this/those'
; "ôma" Pron Dem Prox IN Pl
"awa" Pron Dem Prox AN Obv
"<otêma>" 'his dog/dogs (obviative)'
"atim" N AN Sg Pl Px3Sg Obv

5.1.1 Coverage

The effectiveness of the disambiguation constraints can be examined with a text that has been manually coded for disambiguation and function assignment. In this text of 3,226 Plains Cree words, of which 524 have more than one possible reading before disambiguation, 544 readings were manually marked to be removed. For this same text, the disambiguation constraints remove a total of 374 readings. Of these, 335 are removed in both the manually-coded text and by the constraints. Therefore, the recall rate for removal of an ambiguous reading is 62% while the precision rate is 90%. Many of the problematic cases are those which cannot be determined by sentential context alone, and so lower rates are to be expected. For readings that are not removed (i.e., treated as correct or preferred), 3,241 are marked as correct in the manually-coded text and 3,202 remain after the disambiguation constraints have been applied; therefore, the recall rate for correctly preferred readings is 99%. In Table 1 below, the number of Plains Cree word forms and the numbers of readings both before and after disambiguation are given. When a correct analysis remains, this is indicated with a plus sign <+>; when a correct analysis is removed, this is indicated with a minus sign <->.

n	Before	After	Accuracy	%
2,704	1	1	+	83.8
276	2	1	+	8.6
199	2	2	+	6.2
18	4	1	+	0.6
13	2	1	-	0.4
6	3	2	+	0.2
5	4	1	-	0.2
3	3	1	+	0.09
2	4	2	+	0.06
2	3	3	+	0.06

Table 1: Pre- and post-disambiguation results

Of these word forms, those that have only one reading both before and after disambiguation are assumed to be correct as they are drawn from the hand-verified morphological Gold Standard corpus.[7] There are 529 word forms with two or more readings before disambiguation; 297 of these have one correct reading after disambiguation. This gives a recall rate of 56% for the selection of the one correct reading. After disambiguation, 315 forms are reduced to one reading; the 297 with the correct analysis remaining give a precision rate of 94%. There are 209 word forms with one or more reading remaining after disambiguation, 6% of the 3,226 total word forms presented here. However, in all cases where more than one reading remains, the correct reading has not been removed.

5.2 Function Assignment

5.2.1 Nouns and Demonstratives

Relationships between nouns and demonstratives are determined by their linear and morphological relationships. To identify a demonstrative as dependent on a noun, it must be immediately adjacent to the noun and agree for animacy, number, and obviation. An exemplary constraint for modelling such a relationship is given in (7).[8]

(7) MAP:DemNANSgR @<N TARGET Dem + AN + Sg IF (NOT -1 Obv)(-1 N + AN + Sg BARRIER CLB) ;

Clause boundaries (CLB) are used to limit the scope of constraints. Clause boundaries include a set of punctuation, including periods and other sentence-level punctuation, as well as semicolons, commas, etc. These do not result in clauses in the traditional sense, i.e. containing a verb, but simply serve to divide the text into more manageable sections within which the constraints attempt to identify relationships. The assumption made here is that punctuation identifies natural pauses, and therefore to some degree the intonational contours, that would occur in speech and ideally correspond to syntactic units of some kind. For the Plains Cree corpus, which has been transcribed from recorded speech, we have assumed that

[5] This constraint is interpreted as follows: *remove* the reading containing the tag *Der/Dim*, in the context that the word itself (*0*) must be (*C*) a noun (*N*). The name of the constraint for reference purposes is given after the direction REMOVE, here saying we do not want the derived reading (*DerNo*).

[6] This constraint also *removes* an unwanted reading; an inanimate plural (*IN+Pl*) demonstrative reading is removed when there is an animate obviative noun (*N+AN+Obv*) immediately to the right (position indicated by *1*). The second context indicates that the demonstrative itself (*0*) must also have an *AN+Obv* reading.

[7] Though single analyses here are understood as correct, a single analysis will never be removed by a CG parser. In texts where analyses have not been verified, a word form with only one analysis cannot be guaranteed to be correct.

[8] Function constraints *map* syntactic function tags. Here, the tag is *@<N*, which marks a demonstrative that is dependent on a noun (*N*) to the left (<). The position immediately to the left is indicated with *-1* in the context conditions. Function constraints also make use of *barriers*, which direct the constraint not to look past certain elements, here a clause boundary (*CLB*).

punctuation adequately approximates pauses in the recorded speech. While these boundaries may occasionally separate verbs and arguments, they are still used for the purposes of syntactic modelling.

Multiple constraints of this format are included in the CG parser for Plains Cree; they apply for animate and inanimate nouns, both plural and singular, and animate obviative, as well as looking both to the left and right of demonstrative. A total of 10 constraints are used for these relationships. However, further modification of these constraints is required in future development, as they do not yet account for intervening modifiers such as numerals.

5.2.2 Arguments of Verbs

The arguments of verbs, when lexicalised, can be either nouns, demonstrative pronouns, and personal pronouns. Personal pronouns rarely occur overtly with verbs, but constraints are also included for these relationships. These constraints are written to assign @ACTOR and @GOAL tags to the arguments of verbs, where they are present. As above, constraints are specified for animacy, number, and obviation combinations, as well as whether the verb upon which a nominal is dependent is to its left or right. Examples for assigning @ACTOR and @GOAL functions are given in (8) and (9) respectively.[9]

(8) MAP:AITIACTSgR @<ACTOR TARGET N + AN + Sg IF (NOT 0 Loc)(NOT 0 Obv)(*-1 AI + 3Sg OR TI + 3Sg BARRIER V OR CLB) ;

(9) MAP:TAGOAL3R @<GOAL TARGET N + AN + Sg IF (NOT 0 Loc)(NOT 0 Obv)(*-1 TA + 3SgO BARRIER V OR CLB) ;

A total of 72 constraints are required to map the actor and goal functions to nouns and demonstrative and personal pronouns in the current CG parser. Just as for the disambiguation constraints, some function constraints are also required due to limitations introduced by the morphological analyser. Ten constraints are required to mark the pronoun *êkoni* 'those (ones)' as dependent on an adjacent noun or demonstrative pronoun, rather than marking both as an actor or goal. The word class and agreement features of *êkoni* are not fully specified in the morphological analyser, so constraints targeting the form itself using regular expressions are implemented.

Though not yet tested and refined, broad constraints also assign @{<}OBL{>} (oblique) to nouns that are not morphologically associated with a nearby verb. Many current instances of @OBL are due to overapplication of these constraints and so they are not included in the present results, though further development of these constraints is underway. These nouns generally include roles that are not specified by the features of the verb (e.g. VAIs that may take goals, indirect objects of benefactive VTAs), as well as instruments that indicate the means by which an action is performed.[10] Such constraints will be implemented in the future with reference to lexicosemantic features. Some oblique nouns will, however, always be incorrectly marked to some extent. Chief among these are the use of inanimate nouns as animate nouns for pragmatic reasons—if a narrative requires an inanimate entity to act with some degree of agency, it will occur with animate verbs and demonstratives. However, as the morphological analyser will still identify these as inanimate nouns, such pragmatically animate nouns will likely never be parsed automatically.

5.2.3 Coverage

The morphological feature tags on nouns and verbs are generally sufficient for assigning syntactic roles; the CG has both a recall and precision rate of 92% for @ACTOR and @GOAL assignment when compared to the manually-coded text. Where mismatches of argument assignment occur (n = 15), the nouns have features that allow them to agree with verbs (as actor or goal) on either side of them and the parser has selected an option different from the manual coding. In situations where the CG has not identified a manually identified argument (n = 9), further refinement of constraints, particularly where pronouns and obviative nouns are concerned, is required. Incorrectly identified obliques, discussed above, also fall into this category. Where the CG has assigned an incorrect argument tag (n = 15), these are all instances of *ôma* 'this, it is this' being misidentified as a pronominal VII actor or VTI goal rather than a focus particle. As *ôma* is generally a problematic case for disambiguation, these situations cannot be solved only with morphosyntactic features and syntactic context and will instead require the addition of lexicosemantic information.

6. Results and Discussion

6.1 Overall Phrase Order Patterns

When applied to a corpus of ~73,000 Plains Cree words, the CG parser can be used to investigate word order patterns on a larger scale than previously possible for most indigenous languages. First and foremost, we can see to what extent overt arguments occur in the language: 47% of all clauses containing verbs contain no overt arguments. This is the most common pattern for all four verb classes, as seen in Table 2. This table contains the 22 most common phrase order patterns, excluding oblique elements and particles, out of the 19,734 phrases containing verbs in the corpus. Other general patterns can also be identified, for example, goals also occur more often than actors; phrase order patterns where VTAs occur with actors appear in less than 1% of the total verbal clauses.

[9] Unlike the above constraints, these allow for the verb with agreeing features to occur anywhere to the left of the target nominal; this is indicated by the asterisk <*> in the context conditions. Otherwise, these constraints are not unlike those for nouns and demonstratives: they assign functions when agreement conditions are met, and do not look beyond clause boundaries or, in these cases, other verbs (*V*).

[10] Fortunately for the syntactic modelling of Plains Cree, many oblique functions that are served by nouns in languages such as English (e.g. temporal or spatial functions) are instead achieved by verbal constructions (e.g. *kâ-nîso-kîsikâk* '(when) it is Tuesday' ~ 'on Tuesday') and particles. Spatial functions are performed by particles or by nouns with locative marking, which can never be the arguments of verbs; these are ruled out by the context (*NOT 0 Loc*), as in the constraints given in (8) and (9).

n	Verb class and arguments	%
4865	@PRED-AI	24.7
2128	@PRED-TA	10.8
1432	@PRED-TI	7.3
801	@PRED-II	4.1
539	@PRED-TA @<GOAL	2.7
521	@ACTOR> @PRED-AI	2.6
503	@PRED-AI @<ACTOR	2.5
450	@GOAL> @PRED-TI	2.3
386	@GOAL> @PRED-TA	2.0
324	@PRED-TI @<GOAL	1.6
242	@PRED-AI @PRED-AI	1.2
211	@ACTOR> @PRED-II	1.1
173	@ACTOR> @PRED-TA	0.9
162	@PRED-AI @N> @<ACTOR	0.8
159	@PRED-II @<ACTOR	0.8
148	@PRED-TA @<ACTOR	0.7
132	@PRED-TA @N> @<GOAL	0.7
116	@PRED-TA @PRED-AI	0.6
114	@PRED-TI @PRED-AI	0.6
111	@ACTOR> @PRED-TI	0.6
103	@PRED-TI @<ACTOR	0.5
76	@PRED-TI @N> @<GOAL	0.4

Table 2: Overall phrase order patterns

Following verbs with no overt arguments, the next most common patterns are transitive verbs (VTA and VTI) with goals, both following and preceding the verbs. VAIs with arguments are also seen with comparable frequencies, but as VAIs are the most common verb subtype in the corpus, their frequency with or without overt arguments is unsurprising. As frequency decreases, the patterns begin to demonstrate actors with transitive verbs;[11] it is not until phrase patterns make up less than 1% of those in the corpus that both actors and goals are lexicalised.

6.2 VTA Phrase Order Patterns

VTAs present opportunities for deeper investigations, as they allow for two arguments to be lexicalised and contain the greatest amount of morphological information. Clauses containing VTAs represent 4,551 of the overall 19,734 clauses. The 24 most common word order patterns for these 4,551 clauses are given in Table 3. The more frequent patterns from Table 2 are repeated here, though as larger proportions of VTAs. For example, where a given word order pattern with both a VTA and an actor occurs in less than 1% of overall clauses, similar clauses occur with a rate of less than 4% of all VTA clauses. Though a larger percentage than found in the overall corpus, it is still small, confirming that VTAs with actors are indeed rare. Still, as patterns decrease in frequency, similar patterns to those above are generally present: goals are lexicalised more often than actors, actors are more likely to precede the verb, and arguments without demonstrative pronouns associated with them are more common. Note that in some cases, ACTOR or GOAL tags occur multiple times in the same clause; this is due to nouns and demonstrative pronouns occurring with intervening material, so they are not correctly associated with each other. Further refinement of constraints is required to solve these issues.

n	VTA phrase orders	%
2128	@PRED-TA	46.8
539	@PRED-TA @<GOAL	11.8
386	@GOAL> @PRED-TA	8.5
173	@ACTOR> @PRED-TA	3.8
148	@PRED-TA @<ACTOR	3.3
132	@PRED-TA @N> @<GOAL	2.9
54	@PRED-TA @N> @<ACTOR	1.2
35	@ACTOR> @PRED-TA @<GOAL	0.8
28	@GOAL> @PRED-TA @<GOAL	0.6
23	@N> @GOAL> @PRED-TA	0.5
22	@PRED-TA @<GOAL @<GOAL	0.5
21	@ACTOR> @PRED-TA @<ACTOR	0.5
20	@GOAL> @GOAL> @PRED-TA	0.4
17	@ACTOR> @GOAL> @PRED-TA	0.4
17	@N> @ACTOR> @PRED-TA	0.4
13	@GOAL> @<N @PRED-TA	0.3
12	@PRED-TA @<ACTOR @<GOAL	0.3
12	@GOAL> @PRED-TA @<ACTOR	0.3
11	@GOAL> @PRED-TA @N> @<GOAL	0.2
11	@GOAL> @ACTOR> @PRED-TA	0.2
9	@ACTOR> @ACTOR> @PRED-TA	0.2
9	@PRED-TA @<ACTOR @<ACTOR	0.2
8	@ACTOR> @<N @PRED-TA	0.2
6	@ACTOR> @PRED-TA @N> @<GOAL	0.1

Table 3: Overall VTA phrase order patterns

Similar to all verb types combined, 47% of VTA clauses occur without overt arguments. Therefore, verbs with third person proximate and obviative participants are the better candidates to investigate how direct and inverse morphology combined with phrase order information can offer a surface-syntactic insight into how word order reflects information structure, namely topicality and focus. Phrase order patterns for direct and inverse third-person VTAs are presented in Tables 4 and 5 respectively. Note that for direct verbs, the actor is proximate and the goal is obviative, while the reverse is true for inverse verbs. Table 4 gives the 15 most common of 985 third person direct phrases, while Table 5 represents the nine most common of 217 third person inverse phrases in the corpus.

n	Phrase order	%
303	@PRED-TA	30.8
189	@PRED-TA @<GOAL	19.2
133	@GOAL> @PRED-TA	13.5
47	@ACTOR> @PRED-TA	4.8
46	@PRED-TA @N> @<GOAL	4.7
37	@PRED-TA @<ACTOR	3.8
29	@ACTOR> @PRED-TA @<GOAL	2.9
15	@PRED-TA @N> @<ACTOR	1.5
10	@ACTOR> @GOAL> @PRED-TA	1.0
8	@GOAL> @PRED-TA @<GOAL	0.8
8	@GOAL> @PRED-TA @<ACTOR	0.8
8	@GOAL> @ACTOR> @PRED-TA	0.8
8	@ACTOR> @PRED-TA @<ACTOR	0.8
7	@PRED-TA @<ACTOR @<GOAL	0.7
6	@PRED-TA @<GOAL @<GOAL	0.6
5	@ACTOR> @ACTOR> @PRED-TA	0.5

Table 4: Direct third person phrase order patterns

[11] VIIs are ignored in the present paper, as it is known that a number of errors are present in the assignment of actors; lexicosemantic information will be required in the parser to solve these issues.

n	Phrase order	%
87	@PRED-TA	40.1
21	@PRED-TA @<ACTOR	9.7
15	@PRED-TA @<GOAL	6.9
15	@ACTOR> @PRED-TA	6.9
7	@GOAL> @PRED-TA	3.2
4	@PRED-TA @N> @<ACTOR	1.8
2	@GOAL> @PRED-TA @N> @<ACTOR	0.9
2	@GOAL> @PRED-TA @<ACTOR	0.9
2	@ACTOR> @PRED-TA @<ACTOR	0.9

Table 5: Inverse third person phrase order patterns

In these patterns, we see that regardless of whether the actor or goal is obviative, the less topical argument is always more likely to be lexicalised, as qualitative descriptions would suggest (e.g. Dahlstrom, 1995; Wolvengrey, 2011). Additionally, the more topical participant is more likely to occur earlier in the phrase, particularly when both arguments are lexicalised.

This cursory investigation of VTAs is only one way in which the phrase order patterns can be examined. In the future, phrase order patterns may be investigated in a number of other ways. One such avenue is the internal structure of arguments: whether they are nouns, pronouns, or nouns plus demonstratives can be further indications of topicality. Additionally, phrase orders in each text can be compared; this is already to some degree possible with the VTA phrase order patterns in Vandall and Douquette (1987), as these were previously summarised in Wolvengrey (2011). Between just Vandall and Douquette (1987) and the entire corpus, the percentages of phrase orders demonstrate similar patterns: a single verb is most common and V GOAL ACTOR is the least frequent. In between, the phrase order frequencies descend in nearly identical orders. However, where VTA clauses without any lexicalised arguments make up nearly half (47%) of the VTA clauses in the overall corpus, they make up only 31% of the VTA clauses in Vandall and Douquette (1987).

These discrepancies may be due to inherent differences in the styles of texts in the collection; Vandall and Douquette (1987) is a collection of shorter narratives, while other texts such as Bear et al. (1992), Kâ-Nîpitêhtêw (1998), and Masuskapoe (2010) include longer narratives and dialogues, where there may be more opportunity for non-lexicalised arguments as more verbs refer to the same topical participant. Additionally, Vandall and Douquette (1987) speak mostly about history and relate others' stories, so there are many third person verbs, which can potentially occur with multiple overt arguments. On the other hand, texts like Bear et al. (1992) involve more stories about the speaker's own lives, and dialogues where speakers directly address each other, so there are likely to be a greater proportion of first and second person verbs, which rarely occur with overt arguments, such as personal pronouns.

7. Conclusions

The Constraint Grammar formalism has proven to be an excellent tool for the computational modelling of verbs and arguments in the non-configurational syntax of Plains Cree, making use of its rich agreement morphology. Furthermore, Plains Cree syntactic roles can be adequately modelled using only morphological feature information. Moreover, the type of semantic information concerning certain words and sets of words that would substantially improve the model coverage are readily apparent when the CG parsing rules are tested.

When applied to a Plains Cree corpus, the parser can be used to investigate the phrase order patterns in the language, which display generally expected patterns even without access to higher-level discourse information. A large-scale analysis of clauses reveals that nearly 50% of verb clauses in this corpus occur without overt lexicalised arguments; even for VTAs with two third person arguments, over one third occur without overt arguments. Among those that do occur, the interplay between semantic role and obviation also presents itself: the less topical participant is more likely to be lexicalised, but the more topical is more likely to occur earlier in a clause.

These investigations also suggest that different genres may occur with different lexicalisation patterns. Further development of the CG parser will allow not only for more accurate descriptions of Plains Cree syntax, but also for implementation within tools and resources that can be used by students and speakers of the language.

8. Acknowledgements

This work has been undertaken in the Alberta Language Technology Lab (URL: http://altlab.artsrn.ualberta.ca), University of Alberta, and it has been funded by the Social Sciences and Humanities Research Council of Canada (SSHRC) through a Connections Outreach Grant (611-2016-0207), a Partnership Development Grant (890-2013-0047), and a Doctoral Fellowship (752-2017-2105), as well as by a KIAS Research Cluster Grant (University of Alberta).

9. References

Ahenakew, F. (1987). *Cree language structure: A Cree approach*. Winnipeg: Pemmican Publications.

Ahenakew, A. (2000). *âh-âyîtaw isi ê-kî-kiskêyihtahkik maskihkiy / They Knew Both Sides of Medicine: Cree Tales of Curing and Cursing Told by Alice Ahenakew*. H.C. Wolfart (Ed.). Winnipeg: University of Manitoba Press.

Arrpe, A., Schmirler, K., Harrigan, A. G., & Wolvengrey, A. (2017). A morpho-syntactically tagged corpus for Plains Cree. Paper presented at the 49th Algonquian Conference, Montreal, QC, 27-29 October, 2017.

Bear, G., Fraser, M., Calliou, I., Wells, M., Lafond, A., & Longneck, R. (1992). *kôhkominawak otâcimowiniwâwa / Our Grandmothers' Lives as Told in Their Own Words*. F. Ahenakew & H.C. Wolfart (Eds.). Regina: Canadian Plains Research Center.

Bick, E., & Didriksen, T. (2015, May). CG-3—Beyond Classical Constraint Grammar. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania* (No. 109, pp. 31-39). Linköping University Electronic Press.

Borin, L, Forsberg, M., & Roxendal, J. (2012). Korp-the corpus infrastructure of Spräkbanken. In *LREC*, pp. 474-78.

Bloomfield, L. (1930). *Sacred stories of the Sweet Grass Cree*. National Museum of Canada Bulletin, 60

(Anthropological Series 11). Reprinted 1993, Saskatoon: Fifth House.

Bloomfield, L. (1934). *Plains Cree texts*. American Ethnological Society Publications 16. New York. Reprinted 1974, New York: AMS Press.

Dahlstrom, A. (1995). *Topic, focus and other word order problems in Algonquian*. Winnipeg: Voices of Rupert's Land.

Demers, P., McIlwraith, N.L., Thunder, D., & Wolvengrey, A. (Eds.). (2010). *The Beginning of Print Culture in Athabasca Country. A Facsimile Edition & Translation of a Prayer Book in Cree Syllabics by Father Émile Grouard, OMI, Prepared and Printed at Lac La Biche in 1883 with an Introduction by Patricia Demers*. Edmonton: University of Alberta Press.

Evert, S., & Hardie, A. (2011). Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium. In *Proceedings of the Corpus Linguistics 2011 conference*, University of Birmingham, UK.

Harrigan, A. G., Schmirler, K., Arppe, A., Antonsen, L., Trosterud, T., & Wolvengrey, A. (2017). Learning from the Computational Modelling of Plains Cree Verbs: Analysis and Generation Using Finite State Transduction. *Morphology*, *27*(4), 565-598.

Karlsson, F. (1990). Constraint grammar as a framework for parsing running text. In *Proceedings of the 13th Conference on Computational Linguistics, Vol. 3* (pp. 168-173). Association for Computational Linguistics.

Karlsson, F. (1995a). Designing a parser for unrestricted text. In F. Karlsson, A. Voutilainen, J. Heikkilae, & A. Anttila (Eds.), *Constraint Grammar: a language-independent system for parsing unrestricted text*. Walter de Gruyter, pp. 1–40.

Karlsson, F. (1995b). The formalism and environment of Constraint Grammar parsing. In F. Karlsson, A. Voutilainen, J. Heikkilae, & A. Anttila (Eds.), *Constraint Grammar: a language-independent system for parsing unrestricted text*. Walter de Gruyter, pp. 41–88.

Kâ-Nîpitêhtêw, J. (1998). *ana kâ-pimwêwêhahk okakêskihkêmowina / The Counselling Speeches of Jim Kâ-Nîpitêhtêw*. F. Ahenakew & H.C. Wolfart (Eds.). Winnipeg: University of Manitoba Press.

Masuskapoe, C. (2010). *piko kîkway ê-nakacihtât: kêkêk otâcimowina ê-nêhiyawastêki*. H.C. Wolfart and F. Ahenakew (Eds.). Winnipeg: Algonquian and Iroquoian Linguistics.

Minde, E. (1997). *kwayask ê-kî-pê-kiskinowâpatihicik / Their Example Showed Me the Way: A Cree Woman's Life Shaped by Two Cultures*. F. Ahenakew and H.C. Wolfart (Eds.). Edmonton: University of Alberta Press.

Snoek, C., Thunder, D., Lõo, K., Arppe, A., Lachler, J., Moshagen, S., & Trosterud, T. (2014). Modeling the Noun Morphology of Plains Cree. *ComputEL: Workshop on the use of computational methods in the study of endangered languages*, 52nd Annual Meeting of the ACL.

Vandall, P. & Douquette, J. (1987). *wâskahikaniwiyiniw-âcimowina / Stories of the House People, Told by Peter Vandall and Joe Douquette*. F. Ahenakew (Ed.). Winnipeg: University of Manitoba Press.

Whitecalf, S. (1993). *kinêhiyawiwiniwaw nêhiyawêwin / The Cree Language is Our Identity: The La Ronge Lectures of Sarah Whitecalf*. H.C. Wolfart and F. Ahenakew (Eds.). Winnipeg: University of Manitoba Press.

Wolfart, H. C. (1973). *Plains Cree: A grammatical study* (Vol. 63.5). Philadelphia: American Philosophical Society.

Wolvengrey, A. (Ed.). (2007). *wawiyatācimowinisa / Funny Little Stories*. Regina: Canadian Plains Research Center.

Wolvengrey, A. E. (2011). *Semantic and pragmatic functions in Plains Cree syntax* (Unpublished doctoral dissertation). LOT. Retrieved from http://dare.uva.nl/record/1/342704.

BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages

Benjamin Heinzerling and Michael Strube
AIPHES, Heidelberg Institute for Theoretical Studies
Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg, Germany
{benjamin.heinzerling | michael.strube}@h-its.org

Abstract

We present *BPEmb*, a collection of pre-trained subword unit embeddings in 275 languages, based on Byte-Pair Encoding (BPE). In an evaluation using fine-grained entity typing as testbed, BPEmb performs competitively, and for some languages better than alternative subword approaches, while requiring vastly fewer resources and no tokenization. BPEmb is available at `https://github.com/bheinzerling/bpemb`.

Keywords: subword embeddings, byte-pair encoding, multilingual

1. Introduction

Learning good representations of rare words or words not seen during training at all is a difficult challenge in natural language processing. As a makeshift solution, systems have typically replaced such words with a generic *UNK* token. Recently, based on the assumption that a word's meaning can be reconstructed from its parts, several subword-based methods have been proposed to deal with the unknown word problem: character-based recurrent neural networks (RNN) (Luong and Manning, 2016), character-based convolutional neural networks (CNN) (Chiu and Nichols, 2016), word embeddings enriched with subword information (FastText) (Bojanowski et al., 2017), and byte-pair encoding (BPE) (Sennrich et al., 2016), among others. While pre-trained FastText embeddings are publicly available, embeddings for BPE units are commonly trained on a per-task basis (e.g. a specific language pair for machine-translation) and not published for general use.

In this work we present *BPEmb*, a collection of pre-trained subword embeddings in 275 languages, and make the following contributions:

- We publish BPEmb, a collection of pre-trained byte-pair embeddings in 275 languages;
- We show the utility of BPEmb in a fine-grained entity typing task; and
- We show that BPEmb performs as well as, and for some languages better than, alternative approaches while being more compact and requiring no tokenization.

2. BPEmb: Byte-pair Embeddings

Byte Pair Encoding is a variable-length encoding that views text as a sequence of symbols and iteratively merges the most frequent symbol pair into a new symbol. E.g., encoding an English text might consist of first merging the most frequent symbol pair *t h* into a new symbol *th*, then merging the pair *th e* into *the* in the next iteration, and so on. The number of merge operations o determines if the resulting encoding mostly creates short character sequences (e.g. $o = 1000$) or if it includes symbols for many frequently occurring words, e.g. $o = 30,000$ (cf. Table 1). Since the BPE algorithm works with any sequence of symbols, it requires no preprocessing and can be applied to untokenized text.

We apply BPE[1] to all Wikipedias[2] of sufficient size with various o and pre-train embeddings for the resulting BPE symbol using GloVe (Pennington et al., 2014), resulting in byte-pair embeddings for 275 languages. To allow studying the effect the number of BPE merge operations and of the embedding dimensionality, we provide embeddings for 1000, 3000, 5000, 10000, 25000, 50000, 100000 and 200000 merge operations, with dimensions 25, 50, 100, 200, and 300.

3. Evaluation: Comparison to FastText and Character Embeddings

To evaluate the quality of BPEemb we compare to FastText, a state-of-the-art approach that combines embeddings of tokens and subword units, as well as to character embeddings.

FastText enriches word embeddings with subword information by additionally learning embeddings for character n-grams. A word is then represented as the sum of its associated character n-gram embeddings. In practice, representations of unknown word are obtained by adding the embeddings of their constituting character 3- to 6-grams. We use the pre-trained embeddings provided by the authors.[3]

Character embeddings. In this setting, mentions are represented as sequence of the character unigrams[4] they consist of. During training, character embeddings are learned for the k most frequent characters.

Fine-grained entity typing. Following Schütze (2017) and Yaghoobzadeh and Schütze (2017), we use fine-grained en-

[1] We use the SentencePiece BPE implementation: `https://github.com/google/sentencepiece`.

[2] We extract text from Wikipedia articles with WikiExtract (`http://attardi.github.io/wikiextractor`), lowercase all characters where applicable and map all digits to zero.

[3] `https://github.com/facebookresearch/fastText`

[4] We also studied character bigrams and trigrams. Results were similar to unigrams and are omitted for space.

Merge ops	Byte-pair encoded text
5000	豊 田 駅 (と よ だ え き) は 、 東京都 日 野 市 豊 田 四 丁目 にある
10000	豊 田 駅 (と よ だ えき) は 、 東京都 日 野市 豊 田 四 丁目にある
25000	豊 田駅 (とよ だ えき) は 、 東京都 日 野市 豊田 四 丁目にある
50000	豊 田駅 (とよ だ えき) は 、 東京都 日 野市 豊田 四丁目にある
Tokenized	豊田 駅 （ と よ だ え き ） は 、 東京 都 日野 市 豊田 四 丁目 に ある
10000	豐 田 站 是 東 日本 旅 客 鐵 道 (JR 東 日本) 中央 本 線 的 鐵路 車站
25000	豐田 站是 東日本旅客鐵道 (JR 東日本) 中央 本 線的鐵路車站
50000	豐田 站是 東日本旅客鐵道 (JR 東日本) 中央 本線的鐵路車站
Tokenized	豐田站 是 東日本 旅客 鐵道 （ JR 東日本 ） 中央本線 的 鐵路車站
1000	to y od a _station is _a _r ail way _station _on _the _ch ū ō _main _l ine
3000	to y od a _station _is _a _railway _station _on _the _ch ū ō _main _line
10000	toy oda _station _is _a _railway _station _on _the _ch ū ō _main _line
50000	toy oda _station _is _a _railway _station _on _the _chū ō _main _line
100000	toy oda _station _is _a _railway _station _on _the _chūō _main _line
Tokenized	toyoda station is a railway station on the chūō main line

Table 1: Effect of the number of BPE merge operations on the beginning of the Japanese (top), Chinese (middle), and English (bottom) Wikipedia article TOYODA_STATION. Since BPE is based on frequency, the resulting segmentation is often, but not always meaningful. E.g. in the Japanese text, 豊 (toyo) and 田 (ta) are correctly merged into 豊田 (Toyoda, a Japanese city) in the second occurrence, but the first 田 is first merged with 駅 (eki, *train station*) into the meaningless 田駅 (ta-eki).

tity typing as test bed for comparing subword approaches. This is an interesting task for subword evaluation, since many rare, long-tail entities do not have good representations in common token-based pre-trained embeddings such as word2vec or GloVe. Subword-based models are a promising approach to this task, since morphology often reveals the semantic category of unknown words: The suffix *-shire* in *Melfordshire* indicates a location or city, and the suffix *-osis* in *Myxomatosis* a sickness. Subword methods aim to allow this kind of inference by learning representations of subword units (henceforth: SUs) such as character ngrams, morphemes, or byte pairs.

Method. Given an entity mention m such as *Melfordshire*, our task is to assign one or more of the 89 fine-grained entity types proposed by Gillick et al. (2014), in this case `/location` and `/location/city`. To do so, we first obtain a subword representation

$$s = SU(m) \in R^{l \times d}$$

by applying one of the above SU transformations resulting in a SU sequence of length l and then looking up the corresponding SU embeddings with dimensionality d. Next, s is encoded into a one-dimensional vector representation

$$v = A(s) \in R^{d}$$

by an encoder A. In this work the encoder architecture is either averaging across the SU sequence, an LSTM, or a CNN. Finally, the prediction y is:

$$y = \frac{1}{1 + exp(-v)}$$

(Shimaoka et al., 2017).

Data. We obtain entity mentions from Wikidata (Vrandečić and Krötzsch, 2014) and their entity types by mapping to Freebase (Bollacker et al., 2008), resulting in 3.4 million English[5] instances like (*Melfordshire*: `/location`,`/location/city`). Train and test set are random subsamples of size 80,000 and 20,000 or a proportionally smaller split for smaller Wikipedias. In addition to English, we report results for a) the five languages having the largest Wikipedias as measured by textual content; b) Chinese and Japanese, i.e. two high-resource languages without tokenization markers; and c) eight medium- to low-resource Asian languages.

Experimental Setup. We evaluate entity typing performance with the average of strict, loose micro, and loose macro precision (Ling and Weld, 2012). For each combination of SU and encoding architecture, we perform a Tree-structured Parzen Estimator hyper-parameter search (Bergstra et al., 2011) with at least 1000 hyper-parameter search trials (English, at least 50 trials for other languages) and report score distributions (Reimers and Gurevych, 2017). See Table 3 for hyper-parameter ranges.

4. Results and Discussion

4.1. Subwords vs. Characters vs. Tokens

Figure 1 shows our main result for English: score distributions of 1000+ trials for each SU and architecture. Token-based results using two sets of pre-trained embeddings (Mikolov et al., 2013; Pennington et al., 2014) are included for comparison.

Subword units. BPEmb outperforms all other subword units across all architectures (BPE-RNN mean score 0.624 ± 0.029, max. 0.65). FastText performs slightly worse (FastText-RNN mean 0.617 ± 0.007, max. 0.63)[6],

[5] Numbers for other languages omitted for space.

[6] Difference to BPEmb significant, $p < 0.001$, Approximate Randomization Test.

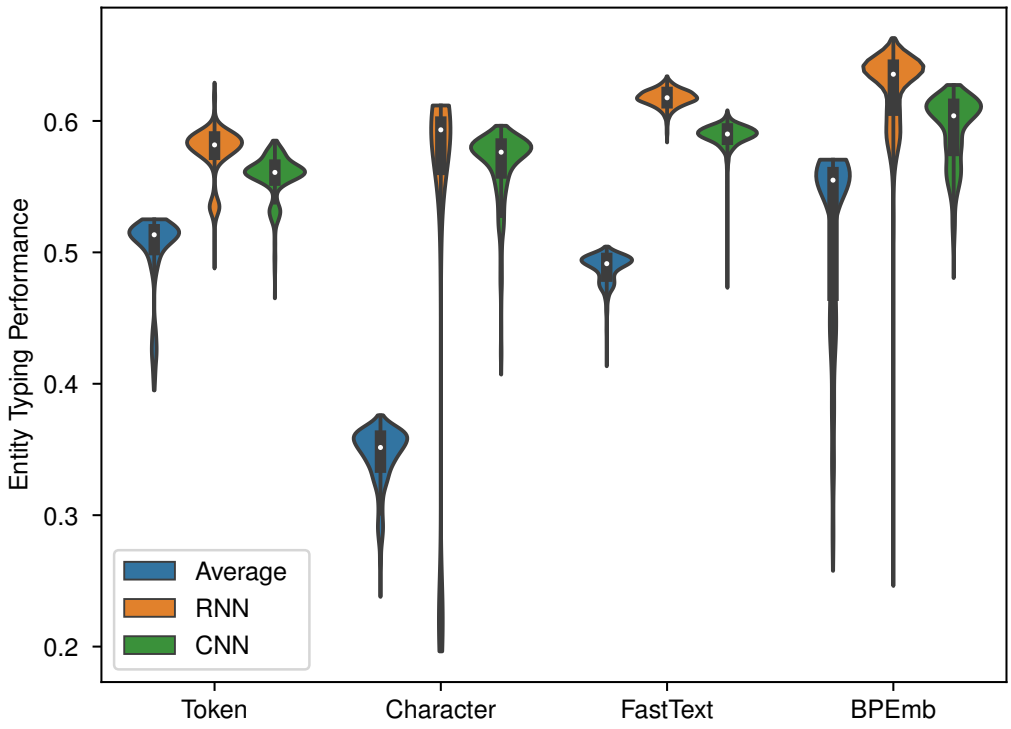


Figure 1: English entity typing performance of subword embeddings across different architectures. This violin plot shows smoothed distributions of the scores obtained during hyper-parameter search. White points represent medians, boxes quartiles. Distributions are cut to reflect highest and lowest scores.

even though the FastText vocabulary is much larger than the set of BPE symbols.

BPEmb performs well with low embedding dimensionality Figure 2, right) and can match FastText with a fraction of its memory footprint (6 GB for FastText's 3 million embeddings with dimension 300 vs 11 MB for 100k BPE embeddings (Figure 2, left) with dimension 25.). As both FastText and BPEmb were trained on the same corpus (namely, Wikipedia), these results suggest that, for English, the compact BPE representation strikes a better balance between learning embeddings for more frequent words and relying on compositionality of subwords for less frequent ones.

FastText performance shows the lowest variance, i.e., it robustly yields good results across many different hyper-parameter settings. In contrast, BPEmb and character-based models show higher variance, i.e., they require more careful hyper-parameter tuning to achieve good results.

Architectures. Averaging a mention's associated embeddings is the worst architecture choice. This is expected for character-based models, but somewhat surprising for token-based models, given the fact that averaging is a common method for representing mentions in tasks such as entity typing (Shimaoka et al., 2017) or coreference resolution (Clark and Manning, 2016). RNNs perform slightly better than CNNs, at the cost of much longer training time.

Language	FastText	BPEmb	Δ
English	62.9	**65.4**	2.5
German	65.5	**66.2**	0.7
Russian	**71.2**	70.7	-0.5
French	**64.5**	63.9	-0.6
Spanish	**66.6**	66.5	-0.1
Chinese	71.0	**72.0**	1.0
Japanese	**62.3**	61.4	-0.9
Tibetan	37.9	**41.4**	3.5
Burmese	**65.0**	64.6	-0.4
Vietnamese	81.0	81.0	0.0
Khmer	**61.5**	52.6	-8.9
Thai	63.5	**63.8**	0.3
Lao	44.9	**47.0**	2.1
Malay	75.9	**76.3**	0.4
Tagalog	**63.4**	62.6	-1.2

Table 2: Entity typing scores for five high-resource languages (top), two high-resource languages without explicit tokenization, and eight medium- to low-resource Asian languages (bottom).

4.2. Multilingual Analysis

Table 2 compares FastText and BPEmb across various languages. For high-resource languages (top) both approaches

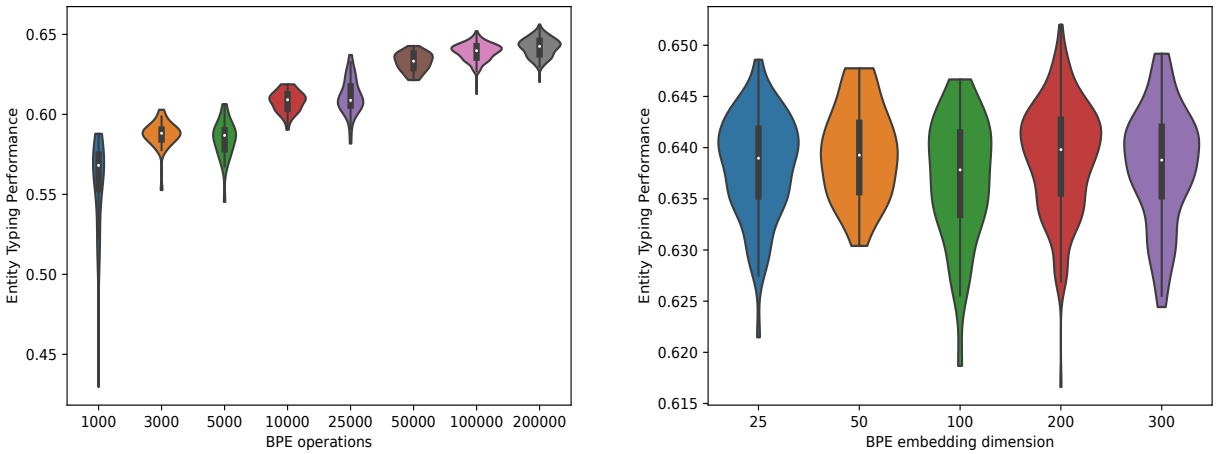


Figure 2: Impact of the number of BPE merge operations (left) and embedding dimension (right) on English entity typing.

perform equally, with the exception of BPEmb giving a significant improvement for English. For high resources languages without explicit tokenization (middle), byte-pair encoding appears to yield a subword segmentation which gives performance comparable to the results obtained when using FastText with pre-tokenized text[7].

Results are more varied for mid- to low-resource Asian languages (bottom), with small BPEmb gains for Tibetan and Lao. The large performance degradation for Khmer appears to be due to inconsistencies in the handling of unicode control characters between different software libraries used in our experiments and have a disproportionate effect due to the small size of the Khmer Wikipedia.

5. Limitations

Due to limited computational resources, our evaluation was performed only for a few of the 275 languages provided by BPEemb. While our experimental setup allows a fair comparison between FastText and BPEmb through extensive hyper-parameter search, it is somewhat artificial, since it disregards context. For example, *Myxomatosis* in the phrase *Radiohead played Myxomatosis* has the entity type `/other/music`, which can be inferred from the contextual music group and the predicate *plays*, but this ignored in our specific setting. How our results transfer to other tasks requires further study.

6. Replicability

All data used in this work is freely and publicly available. BPEmb and code to replicate our experiments is available at `https://github.com/bheinzerling/bpemb`.

7. Conclusions

We presented BPEmb, a collection of subword embeddings trained on Wikipedias in 275 languages. Our evaluation showed that BPEmb performs as well as, and for some languages, better than other subword-based approaches. BPEmb requires no tokenization and is orders of magnitudes smaller than alternative embeddings, enabling potential use under resource constraints, e.g. on mobile devices.

[7]Tokenization for Chinese was performed with Stanford CoreNLP (Manning et al., 2014) and for Japanese with Kuromoji (`https://github.com/atilika/kuromoji`).

Acknowledgments

This work has been supported by the German Research Foundation as part of the Research Training Group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES) under grant No. GRK 1994/1, and partially funded by the Klaus Tschira Foundation, Heidelberg, Germany.

Unit	Hyper-parameter	Space
Token	embedding type	GloVe, word2vec
Character	vocabulary size	50, 100, 200, 500, 1000
	embedding dimension	10, 25, 50, 75, 100
FastText	-	-
BPE	merge operations	1k, 3k, 5k, 10k, 25k 50k, 10k, 200k
	embedding dimension	25, 50, 100, 200, 300

Architecture	Hyper-parameter	Space
RNN	hidden units	100, 300, 500, 700, 1000, 1500, 2000
	layers	1, 2, 3
	RNN dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5
	output dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5
CNN	filter sizes	(2), (2, 3), (2, 3, 4), (2, 3, 4, 5), (2, 3, 4, 5, 6), (3), (3, 4), (3, 4, 5), (3, 4, 5, 6), (4), (4, 5), (4, 5, 6), (5), (5, 6), (6)
	number of filters	25, 50, 100, 200, 300, 400, 500, 600, 700
	output dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5
Average	output dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5

Table 3: Subword unit (top) and architecture (bottom) hyper-parameter space searched.

8. Bibliographical References

Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, pages 2546–2554.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data,* Vancouver, B.C., Canada, 10–12 June 2008, pages 1247–1250.

Chiu, J. and Nichols, E. (2016). Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Clark, K. and Manning, C. D. (2016). Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* Berlin, Germany, 7–12 August 2016.

Gillick, D., Lazic, N., Ganchev, K., Kirchner, J., and Huynh, D. (2014). Context-Dependent Fine-Grained Entity Type Tagging. *ArXiv e-prints*, December.

Ling, X. and Weld, D. S. (2012). Fine-grained entity recognition. In *AAAI*.

Luong, M.-T. and Manning, C. D. (2016). Achieving open vocabulary neural machine translation with hybrid word-character models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1054–1063, Berlin, Germany, August. Association for Computational Linguistics.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of the ICLR 2013 Workshop Track*.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Reimers, N. and Gurevych, I. (2017). Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. *CoRR*, abs/1707.09861.

Schütze, H. (2017). Nonsymbolic text representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 785–796, Valencia, Spain, April. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Shimaoka, S., Stenetorp, P., Inui, K., and Riedel, S. (2017). Neural architectures for fine-grained entity type classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1271–1280, Valencia, Spain, April. Association for Computational Linguistics.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Yaghoobzadeh, Y. and Schütze, H. (2017). Multi-level representations for fine-grained typing of knowledge base entities. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 578–589, Valencia, Spain, April. Association for Computational Linguistics.

Reference production in human-computer interaction: Issues for Corpus-based Referring Expression Generation

Danillo da Silva Rocha, Ivandré Paraboni
University of São Paulo, School of Arts, Sciences and Humanities
São Paulo, Brazil
rochadan60@gmail.com,ivandre@usp.br

Abstract

In the Natural Language Generation field, Referring Expression Generation (REG) studies often make use of experiments involving human subjects for the collection of corpora of definite descriptions. Experiments of this kind usually make use of web-based settings in which a single subject acts as a speaker with no particular addressee in mind (as a kind of monologue situation), or in which participant pairs are engaged in an actual dialogue. Both so-called monologue and dialogue settings are of course instances of real language use, but it is not entirely clear whether these situations are truly comparable or, to be more precise, whether REG studies may draw conclusions regarding attribute selection, referential overspecification and others regardless of the mode of communication. To shed light on this issue, in this work we developed a parallel, semantically annotated corpus of monologue and dialogue referring expressions, and carried out an experiment to compare instances produced in both modes of communication. Preliminary results suggest that human reference production may be indeed affected by the presence of a second (specific) human participant as the receiver of the communication in a number of ways, an observation that may be relevant for studies in REG and related fields.

Keywords: Natural Language Generation, Referring Expression Generation, Corpus

1. Introduction

In Natural Language Generation (NLG) studies, the collection of referring expressions - usually in the form of definite descriptions - produced by human subjects is a common task in Referring Expression generation (REG) and related fields (Krahmer and van Deemter, 2012). Descriptions of this kind are usually elicited from visual stimuli representing a context in which there is one particular target and additional distractor objects. Figure 1 illustrates a stimulus image from the Stars2 corpus (Paraboni et al., 2017a).

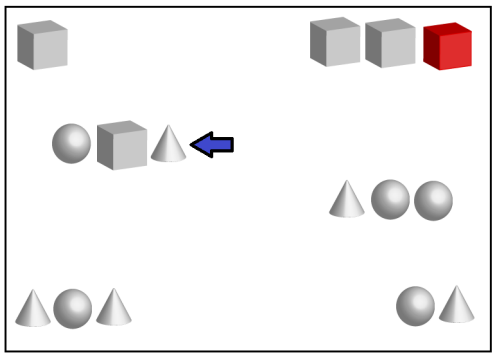


Figure 1: A stimulus image from the Stars2 corpus.

Given a context of this kind, the task of the human subject - who acts as a speaker or writer - is to produce a uniquely identifying description of the intended target. This could be accomplished, for instance, by producing a definite description as in 'The cone next to a grey box'.

Experiments involving human subjects for the collection of referring expression corpora are often implemented as a web-based data collection task, that is, without a particular addressee in mind. When there is no risk of confusion, we will hereby call these *monologue* situations[1].

So-called monologue situations are the method of choice for collecting data in TUNA (Gatt et al., 2007), GRE3D3 (Dale and Viethen, 2009), GRE3D7 (Viethen and Dale, 2011), Wally (Clarke et al., 2013) and other similar resources. By contrast, a number of data collection tasks make use of participant pairs in some form of dialogue. These include GIVE-2 (Gargett et al., 2010), ReferIt (Kazemzadeh et al., 2014), Stars2 (Paraboni et al., 2017a), *b5-ref* (Ramos et al., 2018) and others.

Both dialogue and monologue are of course instances of real language use but, at least from these studies, it is not entirely clear whether the two situations are truly comparable or, to be more precise, whether REG studies that rely on these methods may draw conclusions regarding attribute selection (Dale and Reiter, 1995), referential overspecification (Pechmann, 1989; Paraboni et al., 2017b) and others regardless of the mode of communication.

To shed light on this issue, in this work we developed a parallel, semantically annotated corpus of monologue and dialogue descriptions, and we present an experiment to compare definite descriptions produced in both modes of communication. The objective of the experiment is to investigate whether certain aspects of human reference production - which are particularly relevant for REG studies - may or may not be affected by the presence of a second human participant as the receiver of the communication.

The rest of this paper is organised as follows. Section 2 describes related work in the REG field. Section 3 presents our main experiment, whose results are discussed in Section 4 . Finally, Section 6 presents a number of conclusions and discusses future work.

[1]For a comprehensive discussion on this issue, we refer to (Ginzburg and Poesio, 2016).

2. Related work

This section briefly outlines existing work on REG methods and REG corpora.

2.1. Computational Referring Expression Generation

The attribute selection task for REG is generally modelled as an algorithm that receives as an input a context C comprising at least one target object r (or intended referent) and additional distractor objects. Objects are represented as sets of semantic properties, usually in the form of attribute-value pairs as in *colour*-red. The primary goal of a REG algorithm of this kind is to produce a uniquely identifying description L so as to distinguish r from every other distractor object within C.

Existing REG algorithms include early approaches such as the Greedy (Dale, 2002) and the Incremental (Dale and Reiter, 1995) algorithms. The Graph-based approach (Krahmer et al., 2003) allows the use of relational properties in a novel formulation of the task, and more recently the use of machine learning methods have been considered (Viethen and Dale, 2010; Ferreira and Paraboni, 2017). For a review of the main challenges in the field, see (Krahmer and van Deemter, 2012) and (van Deemter, 2016).

To illustrate the work of a typical REG algorithm, and to highlight a number of issues that may influence its outcome, let us consider a simplified visual context as in Figure 2.

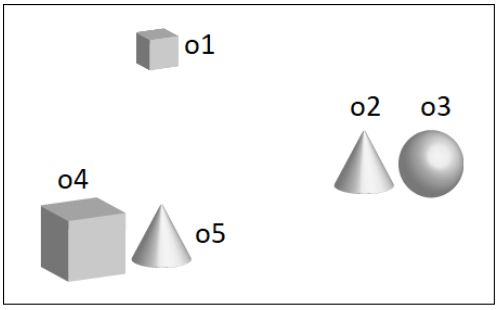


Figure 2: A visual context.

A scene of this kind may be represented as a knowledge base as follows.

o1 <*type*,box>,<*size*,small>

o2 <*type*,cone>,<*size*,large>,<*near*,o3>

o3 <*type*,ball>,<*size*,large>,<*near*,o2>

o4 <*type*,box>,<*size*,large>,<*near*,o5>

o5 <*type*,cone>,<*size*,large>,<*near*,o4>

Let us consider the goal of describing the target object r = $o5$ in this scene by making use of a domain-dependent list of preferred attributes P = <*type*, *size*, *near*>.

A standard attribute selection algorithm may start by making an empty set L (representing the output description) and then considering the first attribute in P, which in the present example is *type*. Since selecting *type* would rule out at least one distractor object (or, in this case, all objects that are not cones $o1$, $o3$ and $o4$), this attribute is included in the output description L, and the relevant objects are removed from the context C.

Next, the second attribute in P is considered, that is, *size*. Since all remaining objects in C share the same *type* value (large), the selection of *size* is disregarded.

Finally, the *near* attribute is examined. Since the target is the only object near $o4$, *near* is selected for inclusion in L and the context is emptied. The algorithm may now terminate or may be called recursively to describe $o4$ as well, resulting in a description that could be subsequently realised as in, e.g., 'the cone near a/the large box'.

Attribute selection is usually driven by discriminatory power (Olson, 1970) (e.g., in the above example, only properties that rule out at least one distractor object are selected), and it is heavily influenced by the order P in which attributes are considered for inclusion in L. As a result, output descriptions may vary considerable both in length and in the kinds of information that they convey (Paraboni, 2003). Given that the ultimate goal of computational REG is (arguably) the generation of descriptions that resemble those produced by human speakers, corpora of human-produced referring expressions are often collected to gain insights on these issues, and in some cases to provide training data for machine learning REG models.

2.2. Corpora for REG

Data collection for REG (e.g., as training data for corpus-based approaches as in (Ferreira and Paraboni, 2014a; Ferreira and Paraboni, 2014b)) may in some cases lead to the development of a so-called REG corpora. In this section we briefly review some of the resources that are more directly relevant to standard REG, in the sense proposed in (Dale and Reiter, 1995) and others.

The TUNA corpus (Gatt et al., 2007) was implemented as a web-based data collection task involving single participants acting as speakers. The corpus comprises two domains: Furniture, containing descriptions of pieces of furniture (e.g., desks, chairs etc.), and People, containing descriptions of human photographs. Both Furniture and People scenes contain from three to seven objects each. The corpus was developed for the study of the content selection task of atomic descriptions, and as a dataset for a series of Shared Tasks (Gatt et al., 2009). TUNA contains 2280 descriptions produced by 60 speakers, being 1200 Furniture descriptions and 1080 People descriptions in so-called monologue situations.

The GRE3D3 and GRE3D7 corpora (Dale and Viethen, 2009; Viethen and Dale, 2011) were also implemented as web-based collection tasks involving single participants in the role of speakers. In both cases, the domain consisted of visual scenes containing either three (in GRE3D3) or seven (GRE3D7) geometric objects (boxes and balls) each, with limited variation in colour and size. The goal of the data collection was to investigate the use of *relational descriptions* (Krahmer et al., 2003; dos Santos Silva and Paraboni, 2015) as in 'the ball next to the yellow cube' in a context in which an atomic description (e.g., 'the ball') would suffice for the purpose of identification. GRE3D3 contains 630 descriptions produced by 63 participants, and GRE3D7 contains 4480 descriptions produced by 287 participants, in both cases presented in monologue situations.

The Stars2 corpus (Paraboni et al., 2017a) was imple-

mented as a series of collaborative tasks involving speaker-hearer participant pairs. The domain consisted of visual scenes containing 15 objects each. The corpus was developed for the study of referential overspecification of atomic and relational descriptions alike. Stars2 contains 884 descriptions produced by 56 speakers in dialogue situations.
More recently, a number of data collection tasks for REG have relied on crowd sourcing methods. These include, for instance, the issue of reference in open domains such as the Referit corpus (Kazemzadeh et al., 2014), conveying descriptions of visual elements in real photographs. The tasks in this case involves describing vague target objects (e.g., the central region of a picture, which may be variously described as 'the old man', 'the middle of the picture', 'a person's nose' etc.), which may be considered distinguished from standard REG.

3. Current work

Unsupervised data collection for REG (e.g., as elicited in a web-based monologue situation) may arguably produce lower quality definite descriptions in general. For instance, without a particular hearer in mind, participants of a REG experiment may be less inclined to craft their referring expressions so as to be unambiguously understood. Moreover, even assuming that all speakers are sufficiently careful during the experiment, situations of communication that are deemed less critical or somewhat less important may still affect attribute choice and referential overspecification (Arts et al., 2011), issues that are at the heart of many REG studies. As a result, monologue and dialogue situations of communication may not be entirely comparable or, at the very least, may elicit different referring expressions.
To clarify this, we designed an experiment to compare definite descriptions produced in dialogue and monologue situations of communication. The goal of the experiment is to investigate whether certain aspects of human reference production - which are particularly relevant for studies in Referring Expression Generation - may or may not be affected by the presence of a second human participant acting as a hearer or reader.
The experiment consists of reproducing a number of trials presented in the Stars2 data collection task (which was originally carried out as a limited form of dialogue between speaker-hearer participant pairs) and by replacing the hearer participant for a simple web interface with no feedback. In other words, we intend to reproduce a number of Stars2 dialogues in monologue situations not unlike those implemented in a number of prominent data collection tasks for REG, including TUNA (Gatt et al., 2007), GRE3D3 (Dale and Viethen, 2009) and others.

3.1. Hypotheses

The experiment investigates four research questions concerning possible differences between definite descriptions produced in monologue and dialogue situations. These questions address both quantitative and qualitative aspects of reference production, and are based on the semantic annotation associated to the descriptions under evaluation as defined by the annotation scheme of the Stars2 corpus (Paraboni et al., 2017a).

Generally speaking, our four research questions assume that, when a particular hearer is present (that is, in a true dialogue situation), speakers will design their referring expressions more carefully than if they were alone (that is, in a so-called monologue situation). This general principle is consistent with studies on referring expression generation in critical situations of communication (Arts et al., 2011) and others. The four hypotheses to be investigated are detailed as follows.

> *h1: Descriptions produced in dialogue situations are, on average, longer than those produced in monologue situations.*

Hypothesis $h1$ will be tested by comparing the average number of annotated properties produced in dialogue situations with descriptions produced under identical circumstances (i.e., in the same context) in monologue situations. Dialogue descriptions are expected to convey more information than monologue descriptions.

> *h2: Descriptions produced in dialogue situations contain, on average, more spatial relations than those produced in monologue situations.*

Hypothesis $h2$ will be tested by comparing the average number of spatial relations observed in dialogue and monologue situations. Dialogue situations are expected to present more relational descriptions.

> *h3: Atomic descriptions produced in dialogue situations include, on average, more information about the target object than those produced in monologue situations.*

Hypothesis $h3$ will be tested by measuring the level of target overspecification in atomic descriptions (e.g., 'the box' versus 'the *red* box' in a context in which there is only one box object) produced in dialogue and monologue situations alike. To this end, the level of overspecification is defined as the number of target properties beyond what would be strictly required for disambiguation, that is, the number of properties added to an otherwise minimally distinguishing description. Atomic descriptions in dialogue situations are expected to be more overspecified than those in monologue situations.

> *h4: Relational descriptions produced in dialogue situations include, on average, more information about the landmark object than those produced in monologue situations.*

Hypothesis $h4$ is similar to $h3$, but now focusing on the landmark portion of relational descriptions (e.g., 'the cube next to a *red* box' in a context in which there is only one box object). This hypothesis will be tested by measuring the level of landmark overspecification in dialogue and monologue situations alike. Descriptions of landmark objects in dialogue situations are expected to be more overspecified than those in monologue situations.

3.2. Subjects

24 native speakers of Brazilian Portuguese, on average 31 years-old and predominantly male (22), drawn from the same population as in (Paraboni et al., 2017a). All participants had normal or corrected vision.

3.3. Materials

24 trials[2] originally presented in the Stars2 data collection task. Each trial consists of a sequence of 16 stimulus images in the same order in which they were presented in the original experiment. The selected trials were those in which there was no misunderstanding or repetition, that is, those in which the hearer always managed to identify the target described by the speaker without any further clarification.

3.4. Procedure

The experiment followed the same procedure as in (Paraboni et al., 2017a), except that there was no hearer participant available, that is, each speaker worked on his/her own by interacting with a WEB interface without any feedback. Participants were required to provide basic information regarding age, gender and an informed consent. This was followed by instructions on how to complete the task. The instructions were the same as in the original experiment - essentially, participants were requested to uniquely identify the object pointed by an arrow - and they did not include any actual examples of referring expression.

After reading the instructions, participants were directed to the experiment proper. This consisted of presenting a series of stimulus images taken from (Paraboni et al., 2017a), one by one, and by requesting the participant to provide a description for the target of each scene. An initial set of four images was presented for practice only, and their responses were not recorded. The following 16 images represented the actual stimuli for the data collection task. During the practice session participants were allowed to ask questions regarding the task to the research assistant. After practice, participants were left unattended and they could not make further questions so as to establish the intended monologue communication setting.

4. Results

A set of 384 descriptions was collected and subsequently annotated by two judges following the same 19-attribute annotation scheme in (Paraboni et al., 2017a), plus an additional 'others' attribute intended to represent any other kind of information outside the scope of the original study. Put together, monologue and dialogue descriptions comprise a parallel corpus of semantically-annotated referring expressions to be made available for further studies on both modes of reference production.

Before discussing our research hypotheses, a preliminary test was carried out so as to assess the closeness between the two (dialogue and monologue) data sets by measuring Dice coefficients (Dice, 1945). As a result, an average Dice score of 0.72 was obtained, which may suggest that the difference between dialogue and monologue datasets was considerably high. The following analysis will discuss this difference in more detail.

Table 1 summarizes our main results for descriptions obtained both in dialogue and monologue situations, in which significant differences between dialogue and monologue descriptions are highlighted. Results are represented as the average description *length* (measured as the number of annotated properties according to the original annotation scheme in (Paraboni et al., 2017a)), the average number of spatial relations (*rel-count*), the average number of overspecified properties in the target portion of the descriptions (*over-tg*), and the average number of overspecified properties in the landmark portion of the descriptions (*over-lm*) as required for evaluating hypotheses $h1..h4$.

The number of description (n) considered in each case is either 384 (i.e., the entire dataset) or 192. The latter corresponds to the tests based on *over-tg* and *over-lm*, which focus on the half of the situations in which the use of relational properties was either optional or compulsory for disambiguation.

5. Discussion

Overall results in Table 1 in principle suggest that descriptions produced in dialogue situations convey, on average, more information than monologue descriptions. To verify this, a between-subjects ANOVA test was carried out for each variable (*length*, *rel-count*, *over-tg* and *over-lm*) as follows.

Regarding hypothesis $h1$, we notice that dialogue descriptions are, on average, longer than those produced in monologue situations. The difference is significant ($F(1,101)=8.52$, $MSE=26.285$ $p<0.05$). This offers support to hypothesis $h1$.

Regarding $h2$, we notice that dialogue descriptions are more likely to include a spatial relation than monologue descriptions. The difference is significant ($F(1,101)=5.15$, $MSE=3.245$ $p<0.05$). This supports $h2$.

Regarding $h3$, results suggest that dialogue target descriptions may be less overspecified than monologue descriptions. However, the difference was not significant. This outcome does not offer support to $h3$.

Finally, regarding $h4$, we notice that dialogue landmark descriptions are more overspecified than monologue descriptions. The difference is also significant ($F(1,101)=4.635$, $MSE=1.622$ $p<0.05$). This supports $h4$.

6. Final remarks

This paper presented an experiment to compare descriptions produced in dialogue and monologue situations, and provided a parallel corpus of semantically-annotated referring expressions in the two modes of communication. Despite the small scale of our study, preliminary results suggest a number of quantitative and qualitative differences between the two kinds of data collection, and which may arguably impact studies on content selection, referential overspecification and other issues that play a central role in REG and related fields.

As future work, we intend to compare monologue and dialogue descriptions against those produced in simulated dia-

[2]We reproduced Stars2 trials 52, 89, 105, 115, 136, 307, 439, 455, 503, 538, 585, 597, 621, 704, 788, 823, 832, 858, 895, 898, 969, 972, 978 and 998 from (Paraboni et al., 2017a).

Table 1: Dialogue vs. Monologue results

	h1: length		h2: rel-count		h3: over-tg		h4: over-lm	
	dial.	mono.	dial.	mono.	dial.	mono.	dial.	mono.
mean	**3.35**	2.98	**0.95**	0.82	0.59	0.51	**0.84**	0.71
var	3.57	2.60	0.68	0.58	0.47	0.32	0.38	0.32
n	384	384	384	384	192	192	192	192

logue, that is, by making use of purpose-made tool to play the role of the hearer participant in a traditional dialogue setting. In doing so, our long-term goal is to obtain referring expressions that resemble those that would be obtained from a real dialogue task at a lower cost, that is, by making use of fewer experiment participants.

The monologue-dialogue parallel corpus - hereby named Stars2MD - is available for research purposes upon request.

7. Acknowledgements

This work has been supported by grant # 2016/14223-0, São Paulo Research Foundation (FAPESP).

8. Bibliographical References

Arts, A., Maes, A., Noordman, L. G. M., and Jansen, C. (2011). Overspecification in written instruction. *Journal of Pragmatics*, 43.

Clarke, A. D. F., Elsner, M., and Rohde, H. (2013). Where's Wally: The influence of visual salience on referring expression generation. *Frontiers in Psychology*, 4(329).

Dale, R. and Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263.

Dale, R. and Viethen, J. (2009). Referring expression generation through attribute-based heuristics. In *Proceedings of ENLG-2009*, pages 58–65.

Dale, R. (2002). Cooking up referring expressions. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 68–75.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.

dos Santos Silva, D. and Paraboni, I. (2015). Generating spatial referring expressions in interactive 3D worlds. *Spatial Cognition & Computation*, 15(03):186–225.

Ferreira, T. C. and Paraboni, I. (2014a). Classification-based referring expression generation. *Lecture Notes in Computer Science*, 8403:481–491.

Ferreira, T. C. and Paraboni, I. (2014b). Referring expression generation: taking speakers' preferences into account. *Lecture Notes in Artificial Intelligence*, 8655:539–546.

Ferreira, T. C. and Paraboni, I. (2017). Generating natural language descriptions using speaker-dependent information. *Natural Language Engineering*, pages 1–22.

Gargett, A., Garoufi, K., Koller, A., and Striegnitz, K. (2010). The GIVE-2 corpus of giving instructions in virtual environments. In *Proceedings of LREC-2010*.

Gatt, A., van der Sluis, I., and van Deemter, K. (2007). Evaluating algorithms for the generation of referring expressions using a balanced corpus. In *Proc. of ENLG-07*.

Gatt, A., Belz, A., and Kow, E. (2009). The TUNA challenge 2009: Overview and evaluation results. In *Proceedings of the 12nd European Workshop on Natural Language Generation*, pages 174–182.

Ginzburg, J. and Poesio, M. (2016). Grammar is a system that characterizes talk in interaction. *Frontiers in Psychology*, 7(1938).

Kazemzadeh, S., Ordonez, V., Matten, M., and Berg, T. (2014). ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798. Association for Computational Linguistics.

Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Krahmer, E., van Erk, S., and Verleg, A. (2003). Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72.

Olson, D. R. (1970). Language and thought: aspects of a cognitive theory of semantics. *Psychological Review*, 77(4):257–273.

Paraboni, I., Galindo, M., and Iacovelli, D. (2017a). Stars2: a corpus of object descriptions in a visual domain. *Language Resources and Evaluation*, 51(2):439–462.

Paraboni, I., Lan, A. G. J., de Sant'Ana, M. M., and Coutinho, F. L. (2017b). Effects of cognitive effort on the resolution of overspecified descriptions. *Computational Linguistics*, 43(2).

Paraboni, I. (2003). *Generating references in hierarchical domains: the case of Document Deixis*. Ph.D. thesis, University of Brighton.

Pechmann, T. (1989). Incremental speech production and referential overspecification. *Linguistics*, 27(1):98–110.

Ramos, R. M. S., Neto, G. B. S., Silva, B. B. C., Monteiro, D. S., Paraboni, I., and Dias, R. F. S. (2018). Building a corpus for personality-dependent natural language understanding and generation. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

van Deemter, K. (2016). *Computational Models of Referring: A Study in Cognitive Science*. MIT Press.

Viethen, J. and Dale, R. (2010). Speaker-dependent variation in content selection for referring expression generation. In *Australasian Language Technology Association ws.*, pages 81–89.

Viethen, J. and Dale, R. (2011). GRE3D7: A corpus of distinguishing descriptions for objects in visual scenes. In *Proceedings of UCNLG+Eval-2011*, pages 12–22.

Definite Description Lexical Choice: taking Speaker's Personality into account

Alex Gwo Jen Lan, Ivandré Paraboni
University of São Paulo, School of Arts, Sciences and Humanities
São Paulo, Brazil
alex.lan@usp.br, ivandre@usp.br

Abstract

In Natural Language Generation (NLG), Referring Expression Generation (REG) lexical choice is the subtask that provides words to express a given input meaning representation. Since lexical choices made in real language use tend to vary greatly across speakers, computational models of lexicalisation have long addressed the issue of human variation in the REG field as well. However, studies of this kind will often rely on large collections of pre-recorded linguistic examples produced by every single speaker of interest, and on every domain under consideration, to obtain meaning-to-text mappings from which the lexicalisation model is built. As a result, speaker-dependent lexicalisation may be impractical when suitable annotated corpora are not available. As an alternative to corpus-based approaches of this kind, this paper argues that differences across human speakers may be at least partially influenced by personality, and presents a personality-dependent lexical choice model for REG that is, to the best of our knowledge, the first of its kind. Preliminary results show that our personality-dependent approach outperforms a standard lexicalisation model (i.e., based on meaning-to-text mappings alone), and that the use of personality information may be a viable alternative to strategies that rely on corpus knowledge.

Keywords: Referring Expressions, Lexical choice, Personality, Big Five

1. Introduction

In Natural Language Generation (NLG), *lexical choice* is understood as the task of selecting words to express an input meaning representation. This paper focuses on the particular subtask of definite descriptions lexical choice, that is, the generation step that follows Referring Expression Generation (REG) content selection (Krahmer and van Deemter, 2012) in a traditional NLG architecture (Reiter and Dale, 2000).

The input to the lexicalisation task is a set of meanings (or properties) represented as attribute-value pairs to be expressed in surface form, and the output is a word string. For simplicity, in what follows we shall focus on the choice of words that realise input properties, and we will leave aside issues of linearisation, agreement and others.

Let us consider the goal of producing a lexicalisation for a possible description of the person illustrated in Figure 1, taken from Face Place[1] images (Righi et al., 2012).



Figure 1: An example image from *Face Place*.

In a context of this kind, an underlying REG algorithm may produce a description as in

{<*gender-male*>,<*hair.style-curly*>}.

The task of the lexicalisation model in this case is to assign words to these properties, which in the present example may result in a word string, as in

'the man with curly hair'.

Existing approaches to definite description lexicalisation will often generate a single, fixed surface realisation from the given input. Human descriptions, on the other hand, show much greater linguistic variation, that is, different speakers will often choose different words to express the same meaning. In the previous example, these may include, for instance, 'the guy with wavy hair', 'the boy with frizzy hair' and many others.

The issue of human variation in definite description lexicalisation has been addressed in a few REG studies. In particular, the work in (Hervás et al., 2013) has extensively analysed the lexicalisation of referring expressions from the TUNA corpus (Gatt et al., 2007), and it has provided a number of insights on how human speakers may be grouped together according to their lexical preferences. Corpus-based studies of this kind, however, will usually rely on a large collection of pre-recorded linguistic examples produced by every single speaker of interest, and on every domain of interest. As a result, corpus-based lexicalisation may not always be a viable solution for practical NLG applications.

As an alternative to corpus-based lexicalisation, we notice that differences across speakers may be at least partially influenced by *personality* traits. Personality models such as the well-known Big Five model (Goldberg, 1990) are largely motivated by linguistic choices made by individuals (e.g., an extrovert may use more words than an introvert etc.) and, in particular, by their use of adjectives (which are ubiquitous in definite descriptions). In addition to that, we notice that personality traits are easily obtainable from a number of sources (e.g., inferable from text on social networks as in (Mairesse et al., 2007)), and this may be usually

[1] Stimulus images courtesy of Michael J. Tarr, Center for the Neural Basis of Cognition and Department of Psychology, Carnegie Mellon Univ. Funding provided by NSF award 0339122.

accomplished at a lower cost than collecting a large REG corpus.

Based on these observations, this paper describes a study on personality-dependent definite description lexical choice that is, to the best of our knowledge, the first of its kind. We propose a machine learning approach to lexical choice that takes as an input, in addition to the intended meaning representations, the personality traits of a target speaker. Results show that our personality-dependent approach outperforms standard lexical choice (i.e., based on meaning representations alone), and suggest that the use of personality information may be a viable alternative to strategies that rely on corpus knowledge.

The rest of this paper is organised as follows. Section 2 describes existing work on personality-based NLG and related fields. Section 3 presents our first experiment, concerning the issue of personality recognition from referring expressions. Section 4 proposes the personality-dependent lexicalisation model, and Section 5 describes its evaluation work. Finally, Section 6 presents a number of conclusions and discusses future work.

2. Background

This section briefly discusses the issue of human variation in the lexical choice task for definite description generation, the Big Five personality model, and the corpus to be taken as our test data.

2.1. Lexical choice and human variation

In Referring Expression Generation, once the generation of a description has been decided (Paraboni and van Deemter, 1999; Paraboni and van Deemter, 2002) and its semantic contents have been determined (Dale and Reiter, 1995), the next and final step consists of performing lexical choice to produce actual text. Given an input meaning representation (i.e., the output of a content selection algorithm) consisting of a set of semantic properties represented as attribute-value pairs, as in *gender-male*, the goal of a lexicalisation model is to generate the surface form of a definite description in a target language (e.g., 'the guy').

Although human variation is a popular research topic in REG content selection (Bohnet, 2008; Fabbrizio et al., 2008; Viethen et al., 2013; Ferreira and Paraboni, 2014), there are few studies focused on the issue of lexical choice of definite descriptions. A remarkable exception is the work in (Hervás et al., 2013; Hervás et al., 2015), which presents a corpus-based approach to lexical choice that attempts to mimic descriptions produced by human speakers in the TUNA domain (Gatt et al., 2007). The study compares a standard baseline model and a proposal that takes individual preferences into account, and results show that the proposal leads to a 40% decrease in similarity error against the reference corpus.

2.2. The Big Five personality model

Studies in Psychology and related fields have devoted great attention to the Big Five model (Goldberg, 1990), which contemplates five fundamental dimensions of human personality: Openness to experience, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. Big Five personality traits are largely motivated by the linguistic choices made by an individual, and may be estimated by a wide range of methods proposed in the Psychology field, the most common being the use of personality inventories. Among these, the need for a quick assessment tool led to the 44-item *BFI* inventory (John et al., 1991), which consists of brief statements containing personality-related adjectives that capture the most essential aspects of each factor in the Big Five model, such as 'Is depressed, blue'. BFI items are answered in a scale ranging from 1 (totally disagree) to 5 (totally agree), and these responses are combined using positive and negative weights to form as a single scalar value representing each of the five dimensions of human personality and additional facets (Soto and John, 2009).

2.3. Personality-based NLG

Given the close relation between personality and natural language, it is not surprising that the use of the Big Five model has been ubiquitous both in natural language understanding (Golbeck et al., 2011; Farnadi et al., 2013; Plank and Hovy, 2015; Najib et al., 2015) and NLG research (Mairesse and Walker, 2010; Marshall et al., 2015; Lukin et al., 2015). In particular, the work in (Mairesse and Walker, 2011) has addressed a wide range of generation decisions that may be driven by a target personality profile. The work focuses on practical, end-to-end language generation by presenting a configurable NLG system to generate restaurant textual recommendations. The system - called PERSONAGE - is trained on personality-annotated data, and the generated text is shown to be recognisable by human judges as reflecting certain well-defined personality traits.

Lexicalisation in PERSONAGE is performed for each content word in the text (and not only for the realisation of definite descriptions) through three parameters: lexicon frequency, lexicon word length and verb strength (e.g., 'suggest' versus 'recommend'). These parameters make use of knowledge obtained from several online lexical resources (e.g., WordNet and VERBOCEAN), and from corpus frequency counts.

2.4. The b5-ref corpus

As a means to investigate the relation between personality and the lexical choice in referring expressions, we make use of the *b5-ref* corpus of definite descriptions annotated with personality information about the individuals who produced them (Paraboni et al., 2017). The corpus is part of a larger dataset of text and accompanying personality information, the *b5* corpus (Ramos et al., 2018).

The *b5-ref* corpus contains descriptions of human photographs elicited from a set of 12 visual contexts built from Face Place (Righi et al., 2012) and further annotated with their semantic properties. This procedure is similar to standard data collection tasks intended to build referring expression corpora (Gatt et al., 2007; Dale and Viethen, 2009).

The choice for the Face Place domain was motivated by the observation that these images are annotated with affective information (e.g., sad, angry etc.), which may arguably help

to make more explicit the possible (personality) differences across speakers. An example of stimulus context from the *b5-ref* corpus is illustrated in Figure 2.



Figure 2: Example of stimulus image built from *Face Place* for the b5-ref corpus.

Based on situations of reference of this kind, subjects were instructed to complete a sentence in the form 'The person outlined in red is the ... ', which requires a uniquely identifying description of the target object. In this example, uniqueness could be achieved, for instance, by making use of expressions such as 'the guy with curly hair', ' the only man in the scene', etc.

The *b5-ref* corpus contains 1810 descriptions produced by 152 native speakers of Brazilian Portuguese who responded a 44-item BFI personality inventory (John et al., 2008) for this language (de Andrade, 2008). The descriptions were subsequently annotated with the 27 most frequent semantic properties observed in the corpus. Each property is represented as an attribute-value pair as in *hair.style-curly*. The collected descriptions are represented, on average, by four annotated properties each. Further details are provided in (Paraboni et al., 2017). Table 1 summarises the attribute frequencies in this domain.

3. Pilot study: Personality recognition from input meaning representations

Before addressing our personality-dependent lexical choice model in the next section, we first carried out an analysis to investigate the relation between Big Five personality traits and the annotated meaning representations that we intend to use as the input to our lexical choice model. To this end, an experiment on personality recognition from referential attribute sets was developed. The goal of this experiment was to illustrate to which extent *b5-ref* referring expressions - and, in particular, the underlying annotation scheme - would reflect personality differences across speakers.

The present Big Five recognition task is in principle analogous to personality recognition from text sources such as social networks (Golbeck et al., 2011), blogs (Iacobelli et al., 2011) essays (Mairesse et al., 2007) and others, except that our input consists of sets of semantic properties representing referring expressions. Although this may not have an obvious, real-world application *per se*, learning personality traits from referential attribute sets may provide indirect evidence that our personality-dependent approach (to be discussed in the next section) is feasible. We notice also that, in a related study, a subset of the actual *b5-ref* word strings (as opposed to their semantic annotation considered in the present case) was applied to a number of the personality recognition tasks. Details are provided in (dos Santos et al., 2017).

Attribute	Possible values	Instances	%
gender	{male,female}	1707	23.7%
race	{asian,black,cauc.}	794	11.0%
smile	{yes,no}	784	10.9%
isYoung	{yes}	705	9.8%
hair.colour	{dark,blonde}	633	8.8%
hair.length	{short,long }	434	6.0%
emotion	{pos.,neg.,neutral}	266	3.7%
eye.colour	{light,dark}	191	2.7%
ponytail	{yes,no}	174	2.4%
eyebrows	{other}	156	2.2%
skin	{fair,dark}	150	2.1%
hair.style	{straight,curly}	134	1.9%
nose	{other}	115	1.6%
face	{other}	109	1.5%
facial.hair	{yes}	109	1.5%
lips	{other}	97	1.3%
spots	{yes}	96	1.3%
eyes	{other}	67	0.9%
hair	{other}	66	0.9%
mouth	{shut, open}	64	0.9%
narrow.eyed	{yes,no}	55	0.8%
shape	{other}	55	0.8%
glasses	{yes,no}	52	0.7%
earring	{yes,no}	50	0.7%
fringe	{yes}	49	0.7%
unkempt	{yes}	41	0.6%
ears	{other}	37	0.5%

Table 1: Annotation scheme for the *b5-ref* corpus and attribute frequencies.

3.1. Computational models

Personality recognition is presently modelled as a binary classification task to determine whether an individual shows positive or negative tendency towards each trait. To this end, we assign positive/negative class labels based on the average score for each trait, that is: positive instances of the class representing a personality trait t consist of the individuals with an equal or above-average score for the trait t, and negative instances correspond to those individuals with below-average scores for t. As a result, 1656 instances were produced for each of the five personality traits. The distribution of positive and negative instances for each class is illustrated in Table 2.

Trait	positive	negative
Extraversion	828	828
Agreeableness	767	889
Conscientiousness	875	781
Neuroticism	829	827
Openness	802	854

Table 2: Learning instances distribution

As learning features, we consider a set of binary values representing the seven most frequent attributes in the corpus (cf. previous section.) Each value indicates whether

that particular attribute appeared in a referring expression or not. Moreover, since the choice of referential attributes may vary considerably across stimuli (e.g., in a scene in which nobody is smiling, the use of the *smile* attribute is far less common than in scenes in which one or more characters are smiling), the learning features also include a context identifier value.

3.2. Results

We performed five independent rule-based decision table classification tasks (Kohavi, 1995) with 10-fold cross-validation over the entire dataset. Precision, Recall and F1-measure results are summarised in Table 3.

	positive class			negative class		
Trait	P	R	F1	P	R	F1
Extraversion	0.55	0.51	0.53	0.55	0.60	0.57
Agreeableness	0.58	0.61	0.59	0.59	0.56	0.57
Conscientiousness	0.57	0.58	0.58	0.57	0.57	0.57
Neuroticism	0.53	0.49	0.51	0.52	0.56	0.54
Openness	0.54	0.52	0.53	0.54	0.55	0.54

Table 3: Personality recognition from attribute sets.

3.3. Discussion

Overall best results were observed for Agreeableness and Conscientiousness, and for the negative instances of Extraversion. On the other hand, results for Neuroticism were considerably lower, suggesting that this particular trait may have less influence over referential attribute choice in the present domain.

4. Lexical choice model

In what follows we will focus on the task of providing lexical choices for definite descriptions alone. This can be viewed as the final generation step that takes the output of a standard REG algorithm (e.g., (Dale and Reiter, 1995)) as its input, and then generates text in a target language - in the present case, Brazilian Portuguese.

Our lexical choice model takes as an input the context id (representing a visual scene in the *b5-ref* corpus) and a concept - hereby represented as a semantic property p - to produce the most likely wording w of p.

Using the entire corpus as test data, all properties with five or more references were mapped onto their lexical forms through manual annotation. For instance, *gender-male* was found to be lexicalised as 'man', 'boy', 'guy' and so on. Given our goal of learning alternative lexicalisations for a given input property, the trivial cases represented by properties with a single possible lexicalisation in the corpus were disregarded.

As a result of the annotation task, a set of 4,345 property-word mappings was obtained. The number of alternative wordings per property ranged from 2 to 9, with an average 4.6 wordings each.

Using the property-word mappings, lexical choice was modelled as a multi-class learning task. The goal of the model is to predict the wording of a given property based on its referential context and on the personality traits of the target speaker.

As learning features, we considered the input property p to be lexicalised, the context id in which p occurs, and five features representing the Big Five personality scores of the speaker as scalar values. The inclusion of the context identifier id is intended to reflect the practical observation that a concept may not have exactly the same meaning (and therefore not necessarily the same wording) in different contexts. For instance, *gender-male* may be lexicalised as 'boy' in a scene showing a child, and as 'young man' in a scene showing a slightly more mature individual.

5. Evaluation

In this section we discuss the evaluation of our personality-dependent lexical choice model. The model is compared against a baseline alternative in which the five personality-related features are omitted. This, in practice, amounts to a baseline method that chooses the most frequent wording for each input property. The goal of this evaluation is to show that the use of personality information leads to more accurate lexical choices than the baseline method.

Both models - with and without personality information - were built from the entire set of 4,345 lexicalisations discussed in the previous section using decision-tree induction with 10-fold cross-validation.

5.1. Results

Table 4 presents Precision, Recall and F1-measure results for the baseline and personality-dependent models for properties with 20 or more instances in the corpus. The 'choice' column shows the number of possible alternative lexicalisations available for each property in the data, and it is indicative of the complexity of each individual task. For brevity, properties for which both models achieved zero F1 scores (mainly due to data sparsity) are not represented.

5.2. Discussion

From the results in the previous section we notice that taking personality information into account generally increases (and never decreases) lexicalisation performance. This offers support to our main research hypothesis.

Personality-dependent lexical choice does seem to make more accurate decisions for most input properties, including even those with a relatively small number of instances. However, a post hoc analysis suggested that the use of personality information is particularly helpful in the lexicalisation of *affective information* (e.g., properties conveying attributes such as *smile*, *emotion* etc.), and only to a lesser extent in the case of more physical features. Although this outcome may seem in principle intuitive, more work is still require to determine why some concepts seem to be more dependent on personality than others.

6. Final remarks

This work has investigated the role of personality traits in the lexical choice in definite description generation. Based on a corpus of definite descriptions annotated with Big Five

			Baseline			Proposal		
Property	instances	choice	P	R	F1	P	R	F1
gender-male	708	7	0.13	0.25	0.17	0.43	0.31	**0.30**
gender-female	494	6	0.61	1.00	0.76	0.61	1.00	0.76
race-asian	363	6	0.14	0.33	0.20	0.49	0.47	**0.45**
smile-yes	347	4	0.32	0.50	0.39	0.51	0.50	**0.41**
smile-no	266	7	0.36	0.50	0.42	0.65	0.79	**0.70**
hair.col-blonde	238	9	0.57	1.00	0.72	0.57	1.00	0.72
race-black	227	2	0.92	0.94	0.93	0.90	0.96	0.93
hair.len-short	223	5	0.81	0.99	0.89	0.81	0.99	0.89
hair.col-dark	153	4	0.26	0.50	0.34	0.55	0.59	**0.55**
race-caucasian	139	3	0.33	0.50	0.40	0.68	0.73	**0.70**
emotion-neg.	75	4	0.15	0.28	0.19	0.30	0.31	**0.30**
hair.len-long	74	4	0.24	0.48	0.32	0.42	0.48	**0.40**
emotion-pos.	45	2	0.32	0.50	0.39	0.88	0.87	**0.88**
emotion-neut.	20	2	0.23	0.50	0.32	0.33	0.43	**0.37**

Table 4: Personality-dependent lexical choice. Best F1 scores for each class are highlighted.

personality information, we have shown that taking personality information into account increases lexical choice accuracy, an insight that may help the design of more realistic (i.e., human-like) models of Natural Language Generation. As future work, we intend to extend our current model to address the task of surface realisation in general, allowing the generation of full text sentences according to a set of target personality traits.

7. Acknowledgements

This work has been supported by grant # 2016/14223-0, São Paulo Research Foundation (FAPESP).

8. Bibliographical References

Bohnet, B. (2008). The fingerprint of human referring expressions and their surface realization with graph transducers. In *5th International Natural Language Generation Conference*, pages 207–210, Salt Fork, Ohio, USA. Association for Computational Linguistics.

Dale, R. and Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19.

Dale, R. and Viethen, J. (2009). Referring expression generation through attribute-based heuristics. In *12th European Workshop on Natural Language Generation*, ENLG '09, pages 58–65, Athens, Greece. Association for Computational Linguistics.

de Andrade, J. M. (2008). *Evidências de validade do inventário dos cinco grandes fatores de personalidade para o Brasil*. Ph.D. thesis, Universidade de Brasília.

dos Santos, V. G., Paraboni, I., and Silva, B. B. C. (2017). Big five personality recognition from multiple text genres. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 29–37, Prague, Czech Republic. Springer-Verlag.

Fabbrizio, G. D., Stent, A., and Bangalore, S. (2008). Trainable speaker-based referring expression generation. In *12th Conference on Computational Natural Language Learning*, pages 151–158, Manchester, UK. Association for Computational Linguistics.

Farnadi, G., Zoghbi, S., Moens, M.-F., and de Cock, M. (2013). Recognising personality traits using Facebook status updates. In *Proceedings of WCPR13 in conjunction with ICWSM-13*, Boston, USA. The AAAI Press.

Ferreira, T. C. and Paraboni, I. (2014). Classification-based referring expression generation. In *Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science 8403*, pages 481–491, Kathmandu, Nepal. Springer.

Gatt, A., van der Sluis, I., and van Deemter, K. (2007). Evaluating algorithms for the generation of referring expressions using a balanced corpus. In *Proceedings of ENLG-07*, Schloss Dagstuhl, Germany. Association for Computational Linguistics.

Golbeck, J., Robles, C., and Turner, K. (2011). Predicting personality with social media. In *CHI Í1 Extended Abstracts on Human Factors in Computing Systems*, CHI EA Í1, pages 253–262, Vancouver, BC, Canada. ACM.

Goldberg, L. R. (1990). An alternative description of personality: The Big-Five factor structure. *Journal of Personality and Social Psychology*, 59:1216–1229.

Hervás, R., Francisco, V., and Gervás, P. (2013). Assessing the influence of personal preferences on the choice of vocabulary for natural language generation. *Information Processing & Management*, 49(4):817–832.

Hervás, R., Arroyo, J., Francisco, V., Peinado, F., and Gervás, P. (2015). Influence of personal choices on lexical variability in referring expressions. *Natural Language Engineering*, inpress:1–34, 12.

Iacobelli, F., Gill, A. J., Nowson, S., and Oberlander, J. (2011). Large scale personality classification of bloggers. In Sidney K. D'Mello, et al., editors, *ACII (2)*, volume 6975 of *Lecture Notes in Computer Science*, pages 568–577, Memphis, TN, USA. Springer.

John, O. P., Donahue, E., and Kentle, R. (1991). The Big Five inventory - versions 4a and 54. Technical report, Inst. Personality Social Research, University of California, Berkeley, CA, USA.

John, O. P., Naumann, L. P., and Soto, C. J., (2008).

Paradigm Shift to the Integrative Big-Five Trait Taxonomy: History, Measurement, and Conceptual Issues, pages 114–158. Guilford Press, New York, NY.

Kohavi, R. (1995). The power of decision tables. In *8th European Conference on Machine Learning*, pages 174–189.

Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Lukin, S. M., Reed, L., and Walker, M. A. (2015). Generating sentence planning variations for story telling. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 188–197, Prague, Czech Republic. Association for Computational Linguistics.

Mairesse, F. and Walker, M. A. (2010). Towards personality-based user adaptation: psychologically informed stylistic language generation. *User Model. User-Adapt. Interaction*, 20(3):227–278.

Mairesse, F. and Walker, M. A. (2011). Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computational Linguistics*, 37(3):455–488.

Mairesse, F., Walker, M., Mehl, M., and Moore, R. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research (JAIR)*, 30:457–500.

Marshall, T. C., Lefringhausen, K., and Ferenczi, N. (2015). The Big Five, self-esteem, and narcissism as predictors of the topics people write about in Facebook status updates. *Personality and Individual Differences*, 85:35–40.

Najib, F., Cheema, W. A., and AdeelNawab, R. M. (2015). Author's traits prediction on Twitter data using content based approach. In Linda Cappellato, et al., editors, *CLEF 2015 Working Notes*, volume 1391 of *CEUR Workshop Proceedings*, Toulouse, France. CEUR-WS.org.

Paraboni, I. and van Deemter, K. (1999). Issues for the generation of document deixis. In *Procs. of workshop on Deixis, Demonstration and Deictic Belief in Multimedia Contexts, in association with the 11th European Summers School in Logic, Language and Information (esslli99)*, pages 44–48.

Paraboni, I. and van Deemter, K. (2002). Towards the generation of document-deictic references. In *Information sharing: reference and presupposition in language generation and interpretation*, pages 329–352. CSLI Publications.

Paraboni, I., Monteiro, D. S., and Lan, A. G. J. (2017). Personality-dependent referring expression generation. In *Text, Speech and Dialogue (TSD-2017) Lecture Notes in Artificial Intelligence vol. 10415*, pages 20–28, Prague, Czech Republic. Springer-Verlag.

Plank, B. and Hovy, D. (2015). Personality traits on Twitter - or - how to get 1,500 personality tests in a week. In *Proc. of WASSA-2015*, pages 92–98.

Ramos, R. M. S., Neto, G. B. S., Silva, B. B. C., Monteiro, D. S., Paraboni, I., and Dias, R. F. S. (2018). Building a corpus for personality-dependent natural language understanding and generation. In *11th International Conference on Language Resources and Evaluation (LREC-2018) (to appear)*, Miyasaki, Japan. ELRA.

Reiter, E. and Dale, R. (2000). *Building natural language generation systems*. Cambridge University Press, New York, NY, USA.

Righi, G., Peissig, J. J., and Tarr, M. J. (2012). Recognizing disguised faces. *Visual Cognition*, 20(2):143–169.

Soto, C. J. and John, O. P. (2009). Ten facet scales for the Big Five Inventory: Convergence with NEO PI-R facets, self-peer agreement, and discriminant validity. *Journal of Research in Personality*, 43(1):84–90.

Viethen, J., Mitchell, M., and Krahmer, E. (2013). Graphs and spatial relations in the generation of referring expressions. In *14th European Workshop on Natural Language Generation*, pages 72–81, Sofia, Bulgaria. Association for Computational Linguistics.

Referring Expression Generation in time-constrained communication

André Costa Mariotti, Ivandré Paraboni
University of São Paulo, School of Arts, Sciences and Humanities
São Paulo, Brazil
drehmariotti@gmail.com, ivandre@usp.br

Abstract

In game-like applications and many others, an underlying Natural Language Generation system may have to express urgency or other dynamic aspects of a fast-evolving situation as text, which may be considerably different from text produced under so-called 'normal' circumstances (e.g., without time constrains). As a means to shed light on possible differences of this kind, this paper addresses the computational generation of natural language text in time-constrained communication by presenting two experiments that use the attribute selection task of definite descriptions (or Referring Expression Generation - REG) as a working example. In the first experiment, we describe a psycholinguistic study in which human participants are engaged in a time-constrained reference production task. This results in a corpus of time-constrained descriptions to be compared with 'normal' descriptions available from an existing (i.e., with no time constraint) REG corpus. In the second experiment, we discuss how a REG algorithm may be customised so as to produce time-constrained descriptions that resemble those produced by human speakers in similar situations. The proposed algorithm is then evaluated against the time-constrained descriptions produced by the human subjects in the first experiment, and it is shown to outperform standard approaches to REG in these conditions.

Keywords: Natural Language Generation, Referring Expression Generation, time constraints

1. Introduction

In Natural Language Generation (NLG), the subtask of Referring Expression Generation (REG) (Krahmer and van Deemter, 2012) consists of providing linguistic forms to describe (or refer to) discourse objects. REG involves two distinct issues: content determination (or deciding what to say) and surface realisation (or deciding how to say it in a given language). In this work we focus on the former, addressing the issue of attribute selection of definite descriptions hereby called REG for simplicity.

Of particular interest to the present discussion is the task of generating natural language text from a visual context provided by, e.g., a video game application. One example of application of this kind is illustrated in Figure 1[1].

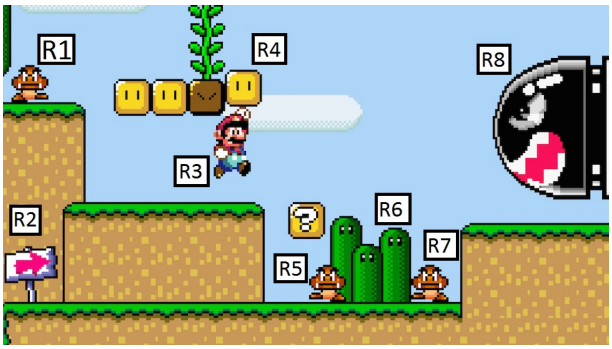


Figure 1: Example of a referential context

In this case, we may assume that an underlying NLG system would receive as an input the current state of the game, including all relevant objects in the scene and their properties, and would generate context-sensitive messages such as 'Try to jump as high as possible to avoid X!' in which X stands for a description of a particular target object r. For instance, assuming r to correspond to the object labelled as $R5$, the description X may be realised as 'the little monster under the question mark', 'the brown monster at the bottom', 'the nasty little thing in front of you', among many others.

The choice of contents to be expressed as a definite description is of course determined by its communicative goals (Paraboni and van Deemter, 1999). In certain game-like applications, however, we notice that these goals may be influenced by the need to express urgency or other dynamic aspects of a fast-evolving situation and, as a result, descriptions to be produced under 'normal' circumstances may be different from those that should be produced under time constrains or other atypical situations (Arts et al., 2011). Possible differences of this kind, and how time-constrained descriptions may be generated using existing REG algorithms, are the focus of the present work.

This paper addresses the computational generation of natural language in time-constrained communication by presenting two experiments that use the attribute selection task of definite descriptions as a working example. In the first experiment, we present a psycholinguistic study in which human participants are engaged in a time-constrained reference production task. From this study, a corpus of time-constrained descriptions - hereby called Stars2T - is elicited, and then compared with 'regular' descriptions available from an existing REG corpus.

Based on these initial findings, in the second experiment we discuss how a REG algorithm may be customised to produce time-constrained descriptions that resemble those produced by human speakers in similar situations by adding a certain amount of overspecified information. The proposed method is evaluated on time-constrained data, and it is shown to outperform standard approaches to REG in these (admittedly unusual) conditions.

The paper is structured as follows. Section 2 briefly de-

[1] Image captured from Nintendo's Super Mario World.

scribes previous work on REG and related topics. Section 3 describes the experiment evolving human subjects in a reference production task. Section 4 presents and evaluates our REG algorithm. Finally, Section 5 draws a number of conclusions and suggests future work.

2. Background

2.1. REG attribute selection

In the Natural Language Generation field, the attribute selection task for REG has been the focus of a wide range of computational methods (Dale and Reiter, 1995; Krahmer et al., 2003; Ferreira and Paraboni, 2014a), reference corpora (Gatt et al., 2007) and shared tasks (Gatt et al., 2009).

REG is typically implemented as an algorithm that receives as an input a context C containing a target object r and additional distractor objects. Objects are represented as sets of semantic properties, usually in the form of attribute-value pairs as in *type*-monster. For instance, the following is a possible representation of the context conveying the eight objects labelled R1..R8 in the previous Figure 1.

R1 <*type*,monster>,<*colour*,brown>

R2 <*type*,sign>,<*colour*,white>

R3 <*type*,man>,<*colour*,red>, <*size*,small>,<*below*,R4>

R4 <*type*,block>,<*colour*,yellow>,<*above*,R3>

R5 <*type*,monster>,<*colour*,brown>,<*left*,R6>

R6 <*type*,cactus>,<*colour*,green>,<*left*,R7>,<*right*,R5>

R7 <*type*,monster>,<*colour*,brown>,<*right*,R6>

R8 <*type*,bomb>,<*colour*,black>,<*size*,large>

Key aspects of the REG attribution selection task are easily illustrated by the Dale & Reiter Incremental algorithm (Dale and Reiter, 1995). The algorithm takes as an input an intended target object r, a context C containing a number of distractor objects from which r has to be distinguished, and a domain-dependent list of preferred attributes P.

The goal of the algorithm is to produce a set L of properties of r such that L distinguishes r from every other object in C. The original Incremental algorithm focuses on the selection of atomic properties only, but handling relational properties (e.g., 'left') is sufficiently straightforward as well (Krahmer et al., 2003; dos Santos Silva and Paraboni, 2015). In what follows we discuss an example that includes atomic and relational properties alike.

The Incremental algorithm iterates over the preferred order P and considers one attribute at a time. If an attribute a helps disambiguate the intended referent, a is included in the output description L. The algorithm terminates when L denotes r and no other object in C, or when all possible attributes in P have been considered. In the former case, L may be realised as a definite description as in 'the large bomb', and in the latter as an indefinite description as in 'a brown monster'.

To illustrate this, let us consider, for instance, the goal of describing the target r = $R5$ in the above context based on a preference order P = <*type*, *colour*, *left*>. The algorithm starts by making an empty set L and then considers the first attribute a in P, that is, *type*. Since this attribute rules out several distractor objects (namely, all objects that are not monsters, that is, $R2$, $R3$, $R4$, $R6$ and $R8$), a is included in L and the objects that have been ruled out by the operation are removed from C.

At this point, the context contains only two distractor objects left: $R1$ and $R7$, both of which being of the same type as r itself. Next in the preference order P, the *colour* attribute is considered. Since all objects in C share the same *colour* value (brown), this attribute does not rule out any distractor objects in C, and it is therefore disregarded.

Finally, the *left* attribute is considered. Since neither $R1$ or $R7$ are on the left side of anything sufficiently close for the purpose of reference (and therefore do not have a *left* property defined in the knowledge base), (*left-R6*) is also added to L, and both distractors are removed from C.

As a result, the context C is now empty, and the algorithm terminates by returning the expression L = {(*type*, *monster*), (*left*, *R6*)}, in which case $R6$ itself could be described recursively as {(*type*, *cactus*)}. This output description could be subsequently realised as, e.g., 'the monster on the left side of the cactus'. Similarly, $R1$ could have been (ambiguously) described as, e.g., 'a monster', and $R7$ as, e.g., 'the monster on the right side of the cactus'.

In addition to the fundamental goal of producing unambiguous descriptions, different REG algorithms may also consider additional requirements. In particular, a great deal of attention has been devoted to the question of *humanlikeness* or plausibility of the generated output (Gatt et al., 2009). As in many other NLG tasks, it is generally assumed that the output description should be as similar as possible to what human speakers would have produced under the same circumstances. This may involve the generation of brief descriptions (Dale, 2002), modelling domain preferences (Gatt et al., 2013), models of human variation (Ferreira and Paraboni, 2014b) and many others. For a detailed account of computational REG, we refer to (van Deemter, 2016).

2.2. Reference production in critical situations

The literature on time-constrained reference production is, from both linguistic and computational perspectives, scarce. The issue is however related to reference production in critical situations of communication, which is the focus of the study in (Arts et al., 2011).

The critical conditions considered in (Arts et al., 2011) are defined as situations in which the identification of a target object is understood by the participants of the dialogue as highly important for the success of an underlying task. To investigate the production of referring expressions in these situations, the study made use of an experiment involving human subjects engaged in a fictitious task of assisting a high-risk surgery that was occurring remotely, in which case the participants were requested to produce descriptions with minimal risk of misinterpretation.

In the experiment in (Arts et al., 2011) there were two groups of participants. The first group was in charge of producing descriptions under so-called 'normal' conditions, and the second group was in charge of producing descriptions under critical condition. Results of the comparison between the two (critical and non-critical) groups suggested that, among other findings, criticality affects the production of referring expressions. More specifically, the study

shows that criticality leads to *referential overspecification* (Pechmann, 1989), that is, the production of expressions containing more information than strictly necessary for disambiguation. An example of overspecified description for the target $R1$ in Figure 1 would be 'the *brown* little monster near the top-left corner', in which the reference to colour is not strictly necessary for identification, that is, it is said to be overspecified.

2.3. The Stars2 corpus

An investigation on the possible relation between referential overspecification and criticality (presently modelled as a time constraint) requires, at the very least, a linguistic domain in which referential overspecification is ubiquitous[2].
In this section we briefly describe the Stars2 corpus of referring expressions (Paraboni et al., 2017a) that will be the basis of the first experiment described in the next section.
Stars2 is a corpus of definite descriptions produced by human subjects in a controlled experiment involving situations of reference in which the use of relational properties between target and landmark objects (e.g., 'the cone next to a box') is likely to occur.
An example of referential context in this domain is illustrated in Figure 2.

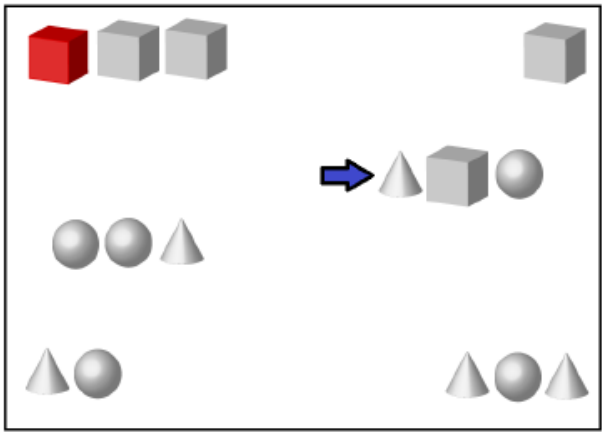


Figure 2: A context in Stars2

Based on stimulus images of this kind, participants of the data collection task were requested to produce a description that would allow the target object (in the example, the cone pointed by the arrow) to be uniquely identified. The goal in (Paraboni et al., 2017a) was to elicit a corpus of referring expressions from scenes that encouraged the use of spatial relations, and in which a range of alternatives for referential overspecification was available, including highly salient properties that were not strictly required for disambiguation.
The corpus contains 884 descriptions produced by 56 participants, and it was intended as a resource suitable to challenge preferences that are well-established in REG, such as the preference for colour (Pechmann, 1989), for absoluteness (van Gompel et al., 2014) and for shorter (e.g., atomic) descriptions over longer (e.g., relational) ones (Kelleher and Costello, 2009; dos Santos Silva and Paraboni, 2015). Given the relative domain complexity and proneness to referential overspecification, we assume Stars2 to be a likely useful resource for our present study on time-constrained reference production.

[2]For other possible triggers to referential overspecification, see for instance (Paraboni et al., 2017b).

3. Time-constrained human reference production

In this section we describe a simple psycholinguistic experiment in which human participants were engaged in a time-constrained reference production task, and we compare the elicited (time-constrained) descriptions with 'normal' descriptions available from the Stars2 corpus (cf. previous section).
The experiment was implemented as a game-like application in which participants were instructed to imagine that they were in charge of a highly critical, time-constrained task. To this end, we made use of a modified version of the Stars2 experiment setting in which participants had a limited time to complete the task, and were constantly reminded of the need to be quick.
The general hypothesis to be investigated is that referring expressions produced in time-constrained communication tend to present certain differences in length and attribute choice when compared to those produced under normal conditions.
Regarding description length, we assume that the effects of time-constrained communication may be comparable to those observed in critical situations discussed in (Arts et al., 2011). In other words, we expect that referring expressions produced under time-constrained conditions to be on average longer (and possibly overspecified) if compared to those produced under normal conditions.
Regarding attribute choice, we notice that studies as in (Pechmann, 1989) argue that *colour* is frequently selected even when not necessary for disambiguation, and for that reason we assume that the use of colour may be more frequent in time-constrained communication as well. Moreover, studies as in (van Gompel et al., 2014) suggest that relative *size* attributes (e.g., 'large', 'greater than', etc.) require considerable cognitive effort from speakers and hearers alike, and for that reason we assume that the use of *size* may be less frequent in time-constrained communication.
To verify these claims, we compared the semantic contents of a set of situations of reference with and without time constraint. This analysis takes into account six variables associated with a given description:

- The amount of information *length* represented by the number of annotated properties.
- The number of spatial relations *rel.count*.
- The degree of overspecification of the target object *over.tg* represented by the number of properties that could be removed from the description with no risk of ambiguity.
- The degree of overspecification of the first landmark object *over.lm1*.

- The overall frequency of the target colour attribute *colour.use*.
- The overall frequency of the target size attribute *size.use*.

3.1. Procedure

Participants of an in-person experiment similar to (Paraboni et al., 2017a) were briefed on the task, and were requested to produce descriptions for each target object to be identified in a sequence of images provided as quickly as possible in order to avoid a (fictitious) bomb explosion (and hence lose the game). Throughout the experiment, time constraint was reinforced with messages as in 'Hurry up, you need to describe all scenes or your time will run out!'.

3.2. Participants

23 volunteers with similar background as the Stars2 participants, with normal or corrected vision, being 12 (52%) female, and on average 32-years old.

3.3. Materials

13 image sequences from the Stars2 data collection task[3], in the same order observed in the construction of the corpus. Each sequence consisted of 20 images similar to Figure 2, being 4 fillers and 16 research stimuli. As in (Paraboni et al., 2017a), half of the stimuli represented contexts that favour the use of spatial relations as in the present example, and the other half did not, hence favouring the use of atomic description as in 'the red ball'.

3.4. Results

A series of 23 individual trials was performed upon appointment with each participant. As a result, a set of 368 descriptions of interest was obtained. Descriptions were subsequently annotated with their semantic properties by two annotators according to the annotation scheme adopted in (Paraboni et al., 2017a).

For the six variables of interest (*length*, *rel.count*, *over.tg*, *over.lm1*, *colour.use*, and *size.use*), Table 1 reports mean, variance and number of items n under consideration in both normal (i.e., as occurring in the Stars2 corpus) and time-constrained (the present experiment) conditions. The value of n corresponds to the total number of collected descriptions (368) except for the tests that distinguish relational and atomic conditions (*over.tg* and *over.lm1*), in which case n corresponds to half the data (184 descriptions each).

3.5. Discussion

We performed individual between-subjects ANOVA to compare the variables of interest in each condition with and without time restriction. Our findings revealed two significant differences discussed below. Other differences were not significant, an outcome that be explained by the small size of our data set.

First, we notice that time-constrained descriptions are, on average, more overspecified (*over.tg*) than those produced in normal situations. The difference is significant ($F(1, 101) = 15.947, p < 0.05$). This result is arguably consistent with the study in (Arts et al., 2011) regarding the production of referring expressions under critical conditions.

Second, we notice that time-constrained descriptions convey, on average, more references to colour (*colour.use*) than those produced in normal situations. The difference is significant ($F(1, 101) = 6.039, p < 0.05$). This result may be explained by the observation that, in the Stars2 domain, the use of colour information tends to be redundant, and possibly correlated with the overspecification of the target object itself. Thus, our findings for (*over.tg*) and (*colour.use*) may actually reflect the same phenomenon.

The presently annotated collection of time-constrained descriptions - hereby called Stars2T for analogy with the original, 'normal' dataset from (Paraboni et al., 2017a) - will be reused in the evaluation of a novel REG method described in the next section.

4. Time-constrained computational REG

As in the case of reference production in critical situations (Arts et al., 2011), the experiment described in the previous section suggests that time-constrained referring expressions may be more overspecified than those produced in so-called 'normal' circumstances. Based on this observation, this section describes a REG experiment involving a modified version of the Dale & Reiter algorithm (Dale and Reiter, 1995) in which the amount of referential overspecification is explicitly manipulated so as to generate descriptions that closely resemble those produced by human speakers in time-constrained situations.

4.1. Algorithms

The experiment makes use of a modified version of the Incremental approach (IA) (Dale and Reiter, 1995), hereby called IAL, that takes as an input an additional overspecification *Level* parameter to enable the generation of descriptions with a customisable amount of information. The value of *Level* is intended to represent the average size of descriptions in the relevant domain, and may in practice be computed from a small set of training examples as discussed in, e.g., (Koolen et al., 2012).

As in the original Incremental approach, IAL selects properties to compose an output description L by iterating over a domain-dependent list of preferred attributes P. At each step, properties that have some discriminatory power are selected until a uniquely identifying description is obtained, or until all properties in P have been attempted.

The difference between the Incremental approach and IAL is the stop condition. Once a suitable description has been obtained, the Incremental approach terminates. IAL, by contrast, uses the value of *Level* as a stop condition. If the number of selected properties in the output description L is equal to or greater than the expected *Level*, IAL also terminates. If not, then additional discriminatory properties will be added to L by following the list of preferred attributes P until the expected *Level* is reached.

To further illustrate the role of referential overspecification in time-constrained REG, we will also consider an implementation of the *Greedy* algorithm in (Dale, 2002) as an

[3]Randomly selected from Stars2 trials 52, 89, 105, 115, 136, 307, 439, 455, 503, 538, 585, 597, 621, 704, 788, 823, 832, 858, 895, 898, 969, 972, 978 and 998.

	length		rel.count		over.tg		over.lm1		colour.use		size.use	
	normal	tc	normal	tc	normal	tc	normal	tc	normal	tc	normal	tc
mean	3.14	3.34	0.90	0.94	0.52	0.79	0.74	0.73	0.18	0.26	0.19	0.20
var.	2.84	2.47	0.66	0.56	0.39	0.39	0.31	0.33	0.14	0.23	0.15	0.20
n	368	368	368	368	184	184	184	184	368	368	368	368

Table 1: Normal and time-constrained (tc) human reference production

additional baseline system. Briefly, this strategy always selects the property with greatest possible discriminatory power, leading to the generation of brief descriptions with little or no overspecification.

4.2. Hypothesis

We hypothesise that adding a certain amount of overspecified information to a description in order to reach the average description length for a given domain will lead to output descriptions that more closely resemble time-constrained human descriptions than those generated by the Incremental approach. This hypothesis will be tested by comparing IA and IAL descriptions with a reference set of time-constrained descriptions produced by human speakers (cf. previous section) while measuring Dice scores (Dice, 1945). We expect IAL to obtain, on average, higher Dice scores than IA.

4.3. Data

We use the Stars2T corpus of time-constrained descriptions described in the previous section. The corpus was systematically divided in training (276 descriptions) and test (92 descriptions) sets so that instances of descriptions produced by every participant, and referring to every possible context, were approximately balanced within each set.

4.4. Procedure

For both IA and IAL, the P input parameter was computed from training data based on attribute frequencies (in which the most frequent attribute is to be attempted first). In the case of IAL, the additional *Level* parameter was also computed from the same training data as the average description length for each context.

Both algorithms - and also the *Greedy* baseline - took as an input the same test data. Evaluation was carried out by comparing every generated description with its human counterpart available from the test corpus while computing Dice coefficients. In addition to that, overall accuracy and MASI scores (Passonneau, 2006) were computed for illustration purposes.

4.5. Results

Table 2 summarised the results for the three algorithms under consideration applied to the test data.

Algorithm	Acc.	Dice	MASI
Greedy	0.18	0.62	0.36
Incremental	0.22	0.66	0.40
IAL	0.33	0.72	0.48

Table 2: Time-constrained computational REG

4.6. Discussion

Results suggest that the proposed IAL method generally outperforms both IA and *Greedy*. A Wilcoxon test shows that the difference between IAL and the second best approach - the Incremental algorithm - is significant both in terms of Dice (W =- 1203, Z =-4.32, $p < 0.0001$) and MASI (W = - 1158, Z = -4.16, p <0.0001) coefficients. In time-constrained reference production, adding overspecified information up to a certain *Level* (as in IAL) outperforms standard REG (as in IA). This offers support to our research hypothesis.

5. Final remarks

This article described an experiment involving human subjects engaged in time-constrained reference production task, and proposed a novel algorithm for the generation of descriptions under these circumstances. The experiment suggested that, in time-constrained communication, human speakers produce expressions that are on average longer or more overspecified than those produced in 'normal' conditions. These differences were subsequently exploited by an algorithm that allows description length to be manipulated explicitly.

Our findings are in principle consistent with studies that address the related issue of reference production in critical situations of communication, and pave the way for the design of referential overspecification strategies that, unlike machine learning methods (Ferreira and Paraboni, 2014a), do not require large sets of linguistic examples on every domain under consideration as training data.

Since referential overspecification may in principle occur in any situation of communication (and not only when there is a time constraint), as future work we intend to refine and validate the current approach by using descriptions available from existing REG corpora, that is, by considering 'normal' situations of reference as well. In doing so, we expect to obtain a more general solution for the generation of human-like overspecified referring expressions.

The Stars2T corpus of semantically-annotated time-constrained referring expressions is freely available for research purposes upon request.

6. Acknowledgements

This work has been supported by grant # 2016/14223-0, São Paulo Research Foundation (FAPESP). The authors are also thankful to Alex G. J. Lan for his support regarding the data collection task.

7. Bibliographical References

Arts, A., Maes, A., Noordman, L., and Jansen, C. (2011). Overspecification in written instructions. *Linguistics*, 49(3):555–574.

Dale, R. and Reiter, E. (1995). Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263.

Dale, R. (2002). Cooking up referring expressions. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 68–75.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.

dos Santos Silva, D. and Paraboni, I. (2015). Generating spatial referring expressions in interactive 3D worlds. *Spatial Cognition & Computation*, 15(03):186–225.

Ferreira, T. C. and Paraboni, I. (2014a). Classification-based referring expression generation. *Lecture Notes in Computer Science*, 8403:481–491.

Ferreira, T. C. and Paraboni, I. (2014b). Referring expression generation: taking speakers' preferences into account. *Lecture Notes in Artificial Intelligence*, 8655:539–546.

Gatt, A., van der Sluis, I., and van Deemter, K. (2007). Evaluating algorithms for the generation of referring expressions using a balanced corpus. In *Proc. of ENLG-07*.

Gatt, A., Belz, A., and Kow, E. (2009). The TUNA challenge 2009: Overview and evaluation results. In *Proceedings of the 12nd European Workshop on Natural Language Generation*, pages 174–182.

Gatt, A., Krahmer, E., van Gompel, R., and van Deemter, K. (2013). Production of referring expressions: Preference trumps discrimination. In *35th Meeting of the Cognitive Science Society*, pages 483–488.

Kelleher, J. D. and Costello, F. J. (2009). Applying computational models of spatial prepositions to visually situated dialog. *Computational Linguistics*, 35(2):271–306.

Koolen, R., Krahmer, E., and Theune, M. (2012). Learning preferences for referring expression generation: Effects of domain, language and algorithm. In *Proceedings of INLG-2012*, pages 3–11.

Krahmer, E. and van Deemter, K. (2012). Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Krahmer, E., van Erk, S., and Verleg, A. (2003). Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72.

Paraboni, I. and van Deemter, K. (1999). Issues for the generation of document deixis. In *Procs. of workshop on Deixis, Demonstration and Deictic Belief in Multimedia Contexts, in association with the 11th European Summers School in Logic, Language and Information (esslli99)*, pages 44–48.

Paraboni, I., Galindo, M., and Iacovelli, D. (2017a). Stars2: a corpus of object descriptions in a visual domain. *Language Resources and Evaluation*, 51(2):439–462.

Paraboni, I., Lan, A. G. J., de Sant'Ana, M. M., and Coutinho, F. L. (2017b). Effects of cognitive effort on the resolution of overspecified descriptions. *Computational Linguistics*, 43(2):451–459.

Passonneau, R. (2006). Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In *International Conference on Language Resources and Evaluation (LREC)*.

Pechmann, T. (1989). Incremental speech production and referential overspecification. *Linguistics*, 27(1):98–110.

van Deemter, K. (2016). *Computational Models of Referring: A Study in Cognitive Science*. MIT Press.

van Gompel, R., Gatt, A., Krahmer, E., and Deemter, K. V. (2014). Testing computational models of reference generation as models of human language production: The case of size contrast. In *RefNet Workshop on Psychological and Computational Models of Reference Comprehension and Production*, Edinburgh, Scotland.

Incorporating Semantic Attention in Video Description Generation

Natsuda Laokulrat[1], **Naoaki Okazaki**[1,2], **Hideki Nakayama**[1,3]
[1]Artificial Intelligence Research Center (AIRC), AIST, Japan
[2]Tokyo Institute of Technology, Japan
[3]The University of Tokyo, Japan
natsudalkr@gmail.com, okazaki@c.titech.ac.jp, nakayama@ci.i.u-tokyo.ac.jp

Abstract

Automatically generating video description is one of the approaches to enable computers to deeply understand videos, which can have a great impact and can be useful to many other applications. However, generated descriptions by computers often fail to correctly mention objects and actions appearing in the videos. This work aims to alleviate this problem by including external fine-grained visual information, which can be detected from all video frames, in the description generation model. In this paper, we propose an LSTM-based sequence-to-sequence model with semantic attention mechanism for video description generation. The model is flexible so that we can change the source of the external information without affecting the encoding and decoding parts of the model. The results show that using semantic attention to selectively focus on external fine-grained visual information can guide the system to correctly mention objects and actions in videos and have a better quality of video descriptions.

Keywords: video description generation, deep learning, RNN, attention

1. Introduction

In the past few years, the image captioning task has been gaining popularity among researchers and high-quality image captions can be generated by deep learning techniques (Vinyals et al., 2014; Karpathy and Fei-Fei, 2015; Fang et al., 2015; Xu et al., 2015; You et al., 2016). The first work in this field was proposed by Vinyals et al. (2014). They proposed an end-to-end system consisting of a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN). The output of the last fully connected layer of the image classification CNN is used as an image feature and then is injected into the RNN-based language model to produce a meaningful sentence. Later, Xu et al. (2015) has proposed an attention-based framework for image captioning which can selectively focus on a portion of an image while producing each word. However, researchers still cannot achieve a satisfying quality of video descriptions generated by computers yet. As with image captioning, automatic video description generation combines two fields of artificial intelligence, computer vision and natural language processing, and has also been tackled by the combination of RNN and CNN.

Venugopalan et al. (2015b) proposed the first end-to-end system to translate a video to natural language by extending the CNN-RNN encoder-decoder framework for image captioning proposed by Vinyals et al. (2014) to generate descriptions for videos. They performed a mean pooling over CNN feature vectors of frames to generate a single vector representation for a video, and then use the vector as input to the RNN decoder to generate a sentence, ignoring the temporal ordering of videos. Subsequently, they have proposed an RNN-based sequence-to-sequence model for generating descriptions of videos (Venugopalan et al., 2015a). They used 2 layers of RNN for both encoding the videos and decoding to sentences, so their model is able to learn both a temporal structure of a sequence of video frames and a sequence model for generating sentences. Later, Laokulrat et al. (2016) applied the temporal attention mechanism to the sequence-to-sequence model to focus on a set of frames while generating each word of the describing sentence. With their attention mechanism, they were able to improve the scores without using any additional features. Yao et al. (2015) used CNN for encoding video frames and used RNN for building a language model at decoding time. They have also incorporated an attentional mechanism to video caption generation, taking into account both local and global temporal structures of videos by incorporating a spatial temporal 3D CNN.

Venugopalan et al. (2017) have attempted to describe objects unseen in paired image-text training data, by taking advantage of other external sources, e.g. labeled images from object recognition datasets, and semantic knowledge extracted from unannotated text.

One problem of video description generation is that generated descriptions by computers often fail to mention correct objects and actions appearing in videos. With the previously proposed sequence-to-sequence model, the video showed in Figure 1 is described as *'a man is riding a car'*. It is obvious that the model cannot detect the main subject *'woman'* and the object *'boat'* appearing in the video. In this work, we aim to solve this problem by integrating external information seamlessly to the conventional sequence-to-sequence model. We want the model to be flexible enough so that we can change the source of the external information without affecting the encoding and decoding parts of the model. Inspired by the image captioning model with semantic attention proposed by You et al. (2016), in this paper, we present a sequence-to-sequence encoder-decoder model with semantic attention mechanism, which is a novel approach to integrate fine-grained visual information appearing in video frames to help the model generate descriptions. By performing a set of experiments, the results show that the semantic atten-

This work has been done while the first author was working at AIRC, AIST.

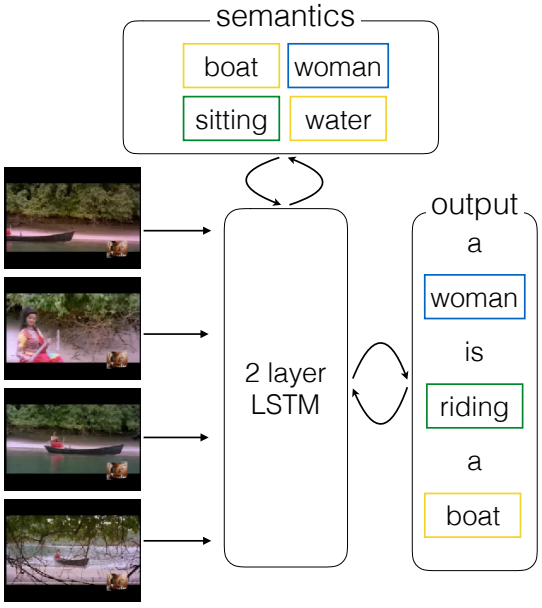


Figure 1: Using semantic attention in video description generation. The system attends to particular semantic words while generating each word of the output sentence. It focuses on the semantic *'woman'* while producing the word *'woman'* (see blue boxes) and focuses on the semantic words *'boat'* and *'water'* while producing the word *'boat'* in the output sentence (see yellow boxes).

tion mechanism can guide the system to correctly mention objects and actions, and a have better quality of video descriptions.

In Figure 1, the semantic-attention model attends to (a) the semantic word *'woman'* when producing with output word *'woman'* (see blue boxes), (b) the semantic word *'sitting'* when producing with output word *'riding'* (see green boxes), (c) the semantic words *'boat'* and *'water'* when producing the output word *'boat'* (see yellow boxes). By integrating the external information, we can fix the error words *'man'* and *'car'* in the output sentence and recover the correct objects *'woman'* and *'boat'*.

The rest of this paper is organized as follows: Section 2 describes in detail the non-attention model and and Section 3 introduces the semantic attention model, along with the mathematical formulas. Section 4 shows the experiments and the results in both qualitative and quantitive aspects. Section 5 discusses the experimental results and Section 6 concludes this paper.

2. LSTM encoder-decoder model

This section explains a sequence-to-sequence model (without attention mechanism) for generating video description. The model is based on the work previously proposed by Venugopalan et al. (2015a).

Given a video as a sequence of frames $V = \{v_1, v_2, ..., v_n\}$ where the video V has n frames and v_t is the t^{th} frame of the video. We extract a frame feature $e_t \in \mathbb{R}^{d_e}$ of each frame v_t by using a pre-trained image classification model IM, where d_e is the dimensionality of the original frame feature. The input feature e_t of the input frame v_t can be described as

$$e_t = \begin{cases} IM(v_t) & ,t \leq n \\ \vec{0} & ,t > n \end{cases} \quad (1)$$

Then, we embed it into a lower-dimensional vector $x_t = W_{ex} e_t \in \mathbb{R}^{d_x}$, where d_x is the dimensionality of the embedded frame feature and $W_{ex} \in \mathbb{R}^{d_x \times d_e}$ is a weight matrix. We omit bias terms for simplicity.

As depicted in the light grey area in Figure 2, the video frames are taken as input to the LSTMs one by one at encoding time, and are set to $\vec{0}$ at decoding time. Then, we can formulate the first (upper) LSTM layer as

$$h_t^{(1)} = LSTM^{(1)}(x_t, h_{t-1}^{(1)}) \quad (2)$$

where $h_t^{(1)} \in \mathbb{R}^{d_h}$ is the hidden state of the first LSTM layer, defined as $LSTM^{(1)}$, at time step t, and d_h is the dimensionality of the hidden states. The hidden state is initialized with a zero vector.

Let $w_t \in \mathbb{R}^r$, where r is the vocabulary size including $\langle UNK \rangle$, $\langle BOS \rangle$ and $\langle EOS \rangle$, be the word generated at time t. The input to the second (lower) LSTM layer is the concatenation of the hidden state of the first LSTM layer and the embedding of the word generated on the previous time step $q(w_{t-1}) \in \mathbb{R}^{d_q}$, where d_q is the dimensionality of the word embedding. So, the second LSTM layer can be described as

$$h_t^{(2)} = LSTM^{(2)}([q(w_{t-1}); h_t^{(1)}], h_{t-1}^{(2)}) \quad (3)$$

where $h_t^{(2)} \in \mathbb{R}^{d_h}$ is the hidden state of the second LSTM layer, defined as $LSTM^{(2)}$, at time step t. Note that the dimensionality of $h_t^{(1)}$ and $h_t^{(2)}$ is the same. At encoding time, w_{t-1} is set to $\vec{0}$ since there is no actual word being generated.

Let $Y = \{y_1, y_2, ..., y_m\} \in \mathbb{R}^r$ be the target sentence with m words. Lastly, the distribution over all the words at time step t can be computed by taking softmax over all possible words. This can be formulated as

$$p(w_t | w_1, ..., w_{t-1}, V) = softmax(W_s h_t^{(2)}) \quad (4)$$

where $W_s \in \mathbb{R}^{r \times d_h}$ is a weight matrix. The decoding process iterates until $\langle EOS \rangle$ symbol is produced. The model can be trained end-to-end by minimizing the softmax cross-entropy loss between y_t and w_t. The loss is computed only in decoding time.

At training time, we use y_{t-1} as an input to Equation 3 instead of w_{t-1}. As with (Venugopalan et al., 2015a), we have $x_t = \vec{0}$ at encoding time and $w_{t-1} = \vec{0}$ at decoding time in order to share the weights of the encoding and decoding LSTMs. This can help speed up the model training without losing much accuracy.

3. Semantic attention model

In this work, *'semantic'* refers to any visual concepts appearing in the video frames, including object, actions, color, shape, relationship, and so on. The objective of incorporating semantic attention mechanism is to enable the language model to focus on related concepts when producing each word of a sentence.

Figure 2 depicts our two-layer LSTM model with semantic attention for generating description sentence from a

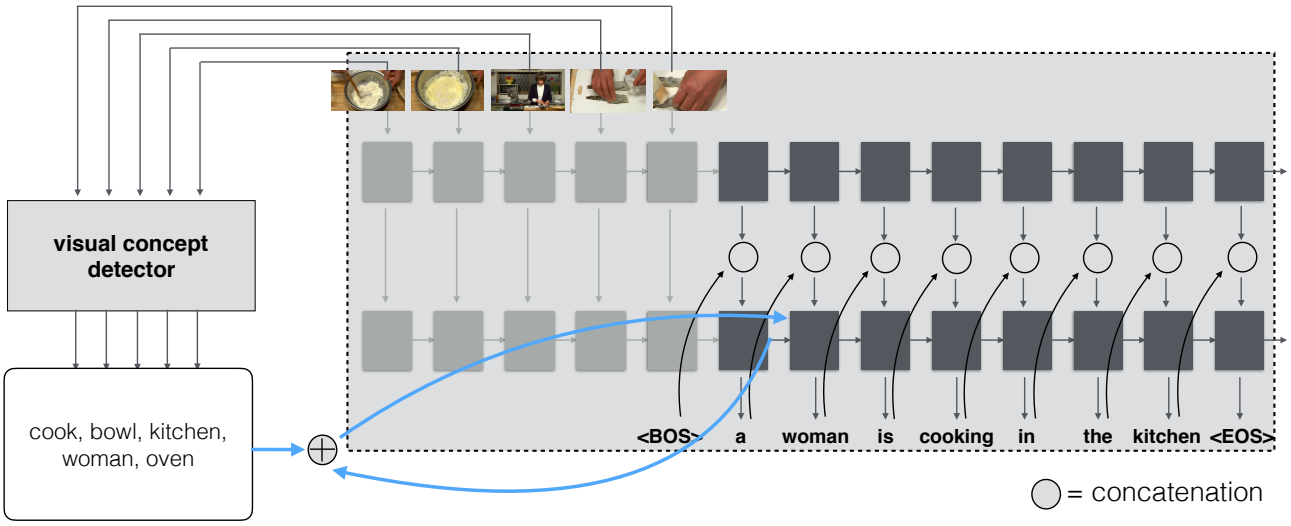


Figure 2: System architecture of the sequence-to-sequence model with semantic attention. In the figure, we omit the image embedding layer, the word embedding layer, and the softmax layer, due to the space constraint. The light grey area shows the non-attention model.

video. Given a set of visual concepts of the video $S = \{s_1, s_2, ..., s_k\} \in \mathbb{R}^{d_q}$ where s_i can be represented by a word embedding in the same space as word input $q(w_{t-1})$. In the semantic attention model, the second-layer LSTM at decoding time can be formulated as

$$h_t^{(2)} = LSTM^{(2)}([q(w_{t-1}); c_t; h_t^{(1)}], h_{t-1}^{(2)}) \quad (5)$$

where the context vector $c_t \in \mathbb{R}^{d_q}$, at the time step t in the decoding stage, is the weighted sum of visual concepts. The context vector c_t can be calculated by

$$c_t = \sum_{i=1}^{k} a_t(i) s_i \quad (6)$$

where $a_t(i) \in \mathbb{R}^1$ is the alignment weight of semantic word (visual concept) s_i at time step t. The weight $a_t(i)$ is computed at every decoding time step t by

$$a_t(i) = \frac{e^{score(h_{t-1}^{(2)}, s_i)}}{\sum_{j=1}^{k} e^{score(h_{t-1}^{(2)}, s_j)}} \quad (7)$$

where $score(h_{t-1}^{(2)}, s_i)$ is the score function used to calculate alignment weights between every visual concept s_i and the hidden state $h_{t-1}^{(2)}$. The score function can be formulated as

$$score(h_{t-1}^{(2)}, s_i) = v_a^\top \cdot tanh(W_a[h_{t-1}^{(2)}; s_i]) \quad (8)$$

The parameters $W_a \in \mathbb{R}^{(d_h+d_q)\times(d_h+d_q)}$ and $v_a \in \mathbb{R}^{(d_h+d_q)}$ of the score function are jointly learned during training.

3.1. Visual concept detection

We use the pre-trained model provided by Fang et al. (2015) to detect visual concepts from every frame of the downsampled videos. The visual concepts include actions, objects, attributes of objects, and also locations. The detected visual concepts of all frames of a video are combined into one collection. For one video, we select 20 concepts from the collection and treat them equally, ignoring their scores provided by the concept detector, as shown in the boxes in Figure 5.

Sentence	Number of occurrences
a man is playing a guitar	217
a man cooking his kichen	196
a man is playing guitar	115
a woman is riding a horse	86
a man is playing the guitar	74
a baby is laughing	67
a person is cooking	59
a cat is playing	57
a woman is peeling a potato	53
a man is singing	52

Table 1: 10 most frequently occurring sentences in the training set and the number of occurrences of each sentence.

4. Experiment

This section explains the dataset we used, the pre-processing steps we performed, the visual concept detection process, the experiment setting as well as the experimental results.

4.1. Dataset and pre-processing

We use *Microsoft Research Video Description Corpus (MSVD)* (Chen and Dolan, 2011) which is a set of 1,970 Youtube clips. For fair comparison with previous work, we split the dataset into train/validation/test sets following (Venugopalan et al., 2015b) and (Yao et al., 2015). The size of the train, validation, and test sets is 1,200, 100, and 670, respectively.

Figure 3 shows the histogram of the number of captions per a video clip in the training set. The average number is ≈40 captions/clip. The minimum number of sentence per a video clip is 18, and the maximum is 66.

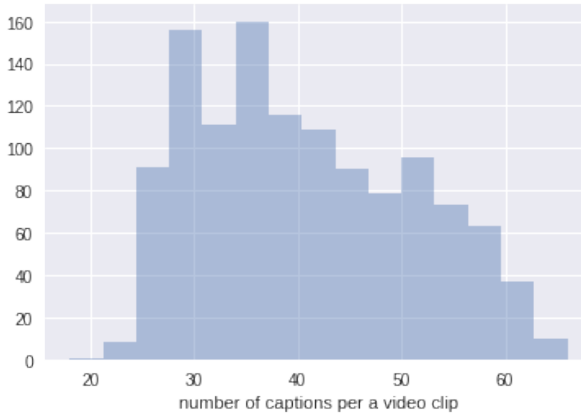


Figure 3: Histogram of number of captions per a video. mean = 40.65, min = 18, max = 66.

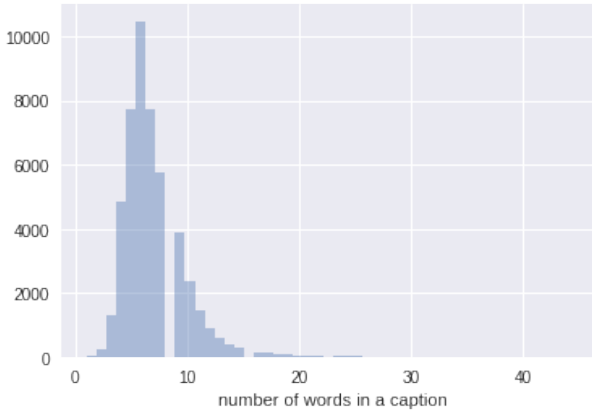


Figure 4: Histogram of number of words in a caption. mean = 7.03, min = 1, max = 45.

Most of the captions contain a single activity and can be described using only one sentence as shown in Table 1. Table 1 shows 10 most frequently occurring captions in the training set. The sentence *'a man is playing a guitar'* appears 217 times which is the maximum number in the training data.

Figure 4 shows the histogram of the number of words per a caption. The average length of the captions in the training set is ≈7 words. The minimum and maximum length are 1 and 45 words respectively.

We downsample the video clips by selecting every 8^{th} frame and resize them to 224x224. Then, we extract features for each frame using a pre-trained image classification model provided in *Caffe Model Zoo* (Jia et al., 2014). In this work, we use the 4096-dimensional fc7 layer of the VGG16 model (Simonyan and Zisserman, 2014) as frame features and embed them into 512-dimensional embeddings.

For text input, the pre-processing includes tokenizing, converting to lower case, and removing punctuations. We represent words with GloVe pre-trained word embeddings, proposed by Pennington et al. (2014). The words outside pre-trained GloVe embeddings are converted to $\langle UNK \rangle$. We then map the 300-dimensional GloVe word vectors into 1000-dimensional vectors. The visual concepts are treated in the same way as text input.

4.2. Experiment setting

In order to enable batch training, we constrain the number of encoding and decoding time steps to be 60 and 20, respectively. We use the Adam optimizer (Kingma and Ba, 2014) with the learning rate of 0.0001 and the mini-batch size of 200. The LSTM hidden layer size is set to 1,000. To avoid overfitting, we apply the dropout strategy (Srivastava et al., 2014) with the ratio of 0.3 at the frame input layer. All the parameters are jointly learned at training time. We apply the beam search strategy at decoding time (beam size = 5).

We implemented our system using *Chainer* (Tokui et al., 2015), and used the caption evaluation package provided by the Microsoft COCO Image Captioning Challenge (Chen et al., 2015). We performed a quantitative analysis of results based on four evaluation metrics, including

BLEU (Papineni et al., 2002), a precision-based evaluation metric used in machine translation.

METEOR (Denkowski and Lavie, 2014), an automatic metric for machine translation evaluation based on explicit word-to-word matching.

CIDEr (Vedantam et al., 2014), an automatic consensus metric of image description quality.

ROUGE-L (Lin, 2004), a recall-oriented evaluation metric popularly used in summarization.

These metrics are common for evaluating image captioning and video description generation systems.

4.3. Experimental results

Table 1 shows the experimental results on MSVD dataset (Chen and Dolan, 2011). We compared our model to the sequence-to-sequence models reported by Venugopalan et al. (2015a) and Laokulrat et al. (2016), when using the same image features (VGG16). We can see some promising results in Figure 5, even though the semantic attention mechanism cannot clearly improve the scores of the test set. The relevant visual concepts were focused and the alignment weights changed properly when each word of the sentences were being generated. By focusing on visual concepts, the model can generate more precise mentions of the objects appearing in the scenes.

In the top-left example, the semantic attention model can recover the mis-mentioned word *(man)* to the correct word *(girl)* with high attention scores on the visual concepts *'woman'* and *'hair'*. It is also interesting that the model focuses on the concept *'brushing'* when correctly producing the phrase *'is doing make up'*. In the bottom-left example, with the help from the visual concept detector, the attention model can fix the error words *'man'* and *'car'* to the correct words *'woman'* and *'boat'*. In the top-right example, the semantic attention model can correctly mention the *'boy'*, while the non-attention model cannot. Lastly, in the bottom-right example, the mis-mentioned object *(bicycle)* in non-attention model can be correctly identified *(bike)* by the model with semantic attention. The visual concept *'bike'* was given a high attention weight when producing the word.

As we can see in Figure 5, many irrelevant visual concepts were detected. This is because the visual concept detector

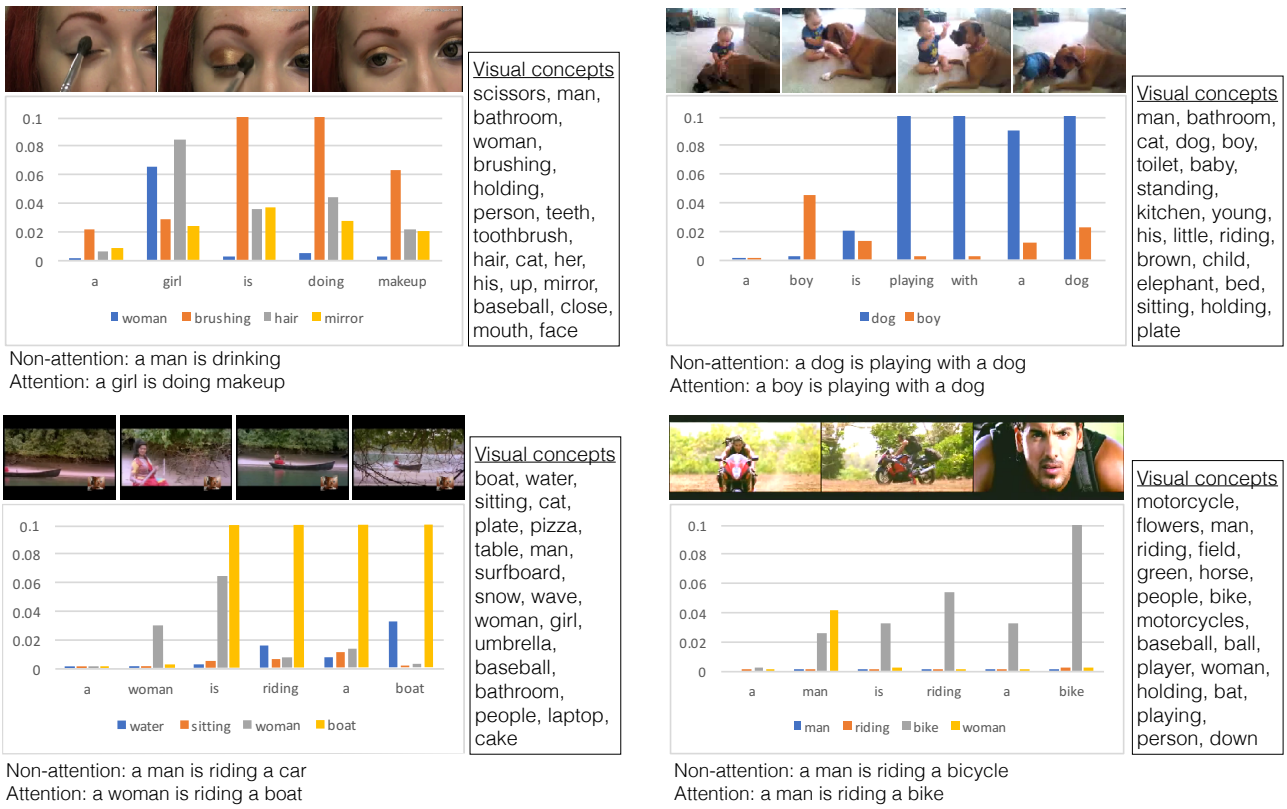


Figure 5: Example of generated descriptions and alignment weights of visual concepts when each word of the sentences was generated. The values are clipped at 0.1 for easier reading.

Model	BLEU	METEOR	CIDEr	ROUGE-L
System A				
Mean pooling	-	0.277	-	-
Seq-to-seq	-	0.292	-	-
Seq-to-seq + flow	-	**0.298**	-	-
System B				
Temp. attention	0.407	**0.310**	**0.615**	**0.676**
Ours				
Non-att.	0.430	**0.318**	**0.670**	0.616
Semantic-att.	**0.431**	0.317	0.668	**0.621**

Table 2: Scores of video description generation results on the MSVD dataset. System A is the results of the sequence-to-sequence model reported by Venugopalan et al (2015a). System B is the results of the sequence-to-sequence model with temporal attention by Laokulrat et al (2016). Note that, for System A, only METEOR scores were reported in the original paper.

was trained on another image dataset, so it could not perform well in video frames. We can apply this model to any kind of external information other than visual concepts.

5. Discussion

We have performed a deep analysis to find why the semantic attention model gets low scores even though it gives us very promising results. Below are the interesting points that we have found from the analysis.

Too specific description. The attention model tends to produce more specific descriptions and therefore is likely to be given lower scores. As shown in Figure 6 (a) and (b), in both examples, the sentences produced by the NON-ATT model[1] get perfect scores (BLEU=1.0, METEOR=1.0, ROUGE=1.0), while the sentences produced by the SEMANTIC-ATT model get much lower scores since they are not perfectly correct. The first video (Figure 6 (a)) shows *'a man pouring oil into a pan'*, and the second video (Figure 6 (b)) shows *'a man sprinkling spices into a pan'*.

Training data. The dataset contains bad examples. The sentence *'a man cooking his kichen'*[1] appears many times in the training data and the ground-truth sentences of the test data. So, if the model produces exactly this sentence, it will get a perfect score. See Figure 6 (a) and (b) for reference.

Furthermore, the misspelling of the training captions and the words outside pre-trained GloVe embeddings will both be converted to $\langle UNK \rangle$ tokens, which can worsen the learning of the models. We can fix this issue by correcting the misspelled words and re-training the word embeddings to cover our vocabularies.

Evaluation metrics. The evaluation metrics are not perfect. In Figure 6 (c), the sentence by the NON-ATT model gets a higher BLEU score even though it is wrong. Also, the

[1]Note that the grammatical and spelling errors are originally from the training data. For fair comparison with other previous work, we did not modify the data.



Figure 6: Videos used in the discussion. Read Section 5. for more detail[1].

word 'boy' does not appear in the ground-truth sentences. All of the ground-truth sentences use the word *'baby'*, so the score of the SEMANTIC-ATT model is even lowered.

For these reasons, the quantitative improvement is small and not obvious, but we believe that the results from the attention model are promising and potentially useful, especially when the visual concept detector can work well. We can replace the concept detector with other object/action prediction models, or combine collections of words detected by two or more detectors. Our semantic attention model can flexibly incorporate that external information into the conventional sequence-to-sequence model.

6. Conclusion

We have proposed a sequence-to-sequence model with semantic attention for video description generation, which can flexibly incorporate that external information into the conventional sequence-to-sequence model. The results show that the model is able to learn to focus on external fine-grained information of videos and a have better quality of video descriptions. The results from the attention model are promising and potentially useful, especially when the visual concept detector can work well. We can replace the concept detector with other object/action prediction models, or combine collections of words detected by two or more detectors.

Acknowledgements

This paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). We also would like to thank the anonymous reviewers for their insightful comments and suggestions, which were helpful in improving the quality of the paper.

7. Bibliographical References

Chen, D. L. and Dolan, W. B. (2011). Collecting highly parallel data for paraphrase evaluation. In *ACL 2011*.

Chen, X., Fang, H., Lin, T., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. (2015). Microsoft COCO captions: Data collection and evaluation server. *arXiv:1504.00325*.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

Fang, H., Gupta, S., Iandola, F. N., Srivastava, R. K., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J. C., Zitnick, C. L., and Zweig, G. (2015). From captions to visual concepts and back. In *CVPR 2015*.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv:1408.5093*.

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *CVPR 2015*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv:1412.6980v8*.

Laokulrat, N., Phan, S., Nishida, N., Shu, R., Ehara, Y., Okazaki, N., Miyao, Y., and Nakayama, H. (2016). Generating video description using sequence-to-sequence model with temporal attention. In *COLING 2016*.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *ACL 2002*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP 2014*.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Tokui, S., Oono, K., Hido, S., and Clayton, J. (2015). Chainer: a next-generation open source framework for deep learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*.

Vedantam, R., Zitnick, C. L., and Parikh, D. (2014). Cider: Consensus-based image description evaluation. *arXiv:1411.5726v2*.

Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., and Saenko, K. (2015a). Sequence to sequence – video to text. In *ICCV 2015*.

Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., and Saenko, K. (2015b). Translating videos

to natural language using deep recurrent neural networks. In *NAACL-HLT 2015*.

Venugopalan, S., Hendricks, L. A., Rohrbach, M., Mooney, R. J., Darrell, T., and Saenko, K. (2017). Captioning images with diverse objects. In *CVPR 2017*.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2014). Show and tell: A neural image caption generator. *arXiv:1411.4555*.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A. C., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. *arXiv:1502.03044v3*.

Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., and Courville, A. (2015). Describing videos by exploiting temporal structure. In *ICCV 2015*.

You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J. (2016). Image captioning with semantic attention. In *CVPR 2016*.

GenDR: A Generic Deep Realizer with Complex Lexicalization

François Lareau, Florie Lambrey, Ieva Dubinskaite, Daniel Galarreta-Piquette, Maryam Nejat
OLST, Département de linguistique et de traduction, Université de Montréal
C.P. 6128, succ. Centre-Ville, Montréal QC, H3C 3J7, Canada
{francois.lareau, florie.lambrey, ieva.dubinskaite, daniel.galarreta-piquette, maryam.sadat.nejat}@umontreal.ca

Abstract

We present a generic deep realizer called GenDR, which takes as input an abstract semantic representation of predicate-argument relations, and produces corresponding syntactic dependency structures in English, French, Lithuanian and Persian, with the possibility to fairly easily add more languages. It is generic in that it is designed to operate across a wide range of languages and applications, given the appropriate lexical resources. The focus is on the lexicalization of multiword expressions, with built-in rules to handle thousands of different cross-linguistic patterns of collocations (intensifiers, support verbs, causatives, etc.), and on rich paraphrasing, with the ability to produce many syntactically and lexically varied outputs from the same input. The system runs on a graph transducer, MATE (Bohnet et al., 2000; Bohnet and Wanner, 2010), and its grammar design is directly borrowed from MARQUIS (Lareau and Wanner, 2007; Wanner et al., 2009; Wanner et al., 2010), which we have trimmed down to its core and built upon. The grammar and demo dictionaries are distributed under a CC-BY-SA licence (http://bit.ly/2x8xGVO). This paper explains the design of the grammar, how multiword expressions (especially collocations) are dealt with, and how the syntactic structure is derived from the relative communicative salience of the meanings involved.

Keywords: multilingual natural language generation, deep realization, lexicalization, multiword expressions, collocations

1. Introduction

Natural language generation (NLG) is one of the rare tasks in natural language processing (NLP) that is not yet completely dominated by statistical or neuronal methods. We believe the main reason for this is that for real-life applications, the output of an NLG system must often be flawless, so that extremely high precision is required, and rule-based methods still outperform other approaches in that respect. A number of domain-independent text realizers have been developed to provide an easier way of producing text automatically. While some have impressive coverage, many expect as input a syntactic structure, thus offering very little flexibility in terms of lexical choice and structure. Others take a more abstract input, offering more syntactic flexibility, but their lexicalization model is rather rigid. We propose a multilingual generic deep realizer, GenDR, that is a platform for the modeling of the semantics-syntax interface in languages. In this paper, we will show how GenDR deals with two crucial tasks: **arborization** (building a syntactic structure that reflects the semantic structure) and **lexicalization** (picking the right words to express the desired meanings).

2. Previous work

Most realizers that we know of expect an input where both lexical choice and syntactic structure have already been computed, leaving the user with two particularly complex tasks. This is the case of FUF/SURGE (Elhadad, 1993; Elhadad and Robin, 1996), RealPro (Lavoie and Rambow, 1997; CoGenTex, 1998), SimpleNLG (Gatt and Reiter, 2009), its bilingual version, SimpleNLG-EnFr (Vaudry and Lapalme, 2013) and its Spanish version, SimpleNLG-ES (Ramos-Soto et al., 2017), JSReal (Daoust and Lapalme, 2015) and its bilingual version, JSRealB (Molins and Lapalme, 2015), as well as ATML3 (Weißgraeber and Madsack, 2017). KPML (Bateman, 1996) and OpenCCG (White, 2008) both start from a more abstract representation of the text's meaning, but they tend to focus on the grammar more than the lexicon, resulting in well-formed sentences that somehow lack lexical flexibility. More recently, statistical approaches have been applied to text generation from logical forms (Basile, 2015) or semantic structures (Mille, 2014), but again, lexical choice is rather rigid. All of these realizers have a hard time producing collocations.

MARQUIS (Lareau and Wanner, 2007; Wanner and Lareau, 2009; Wanner et al., 2010) was a multilingual data-to-text system for the air quality domain that addressed the issues that we are concerned with, and its linguistic realization component has been reused in a couple of projects from different domains, namely patents (Wanner et al., 2009; Wanner et al., 2011) and football (Bouayad-Agha et al., 2011; Bouayad-Agha et al., 2012). Its lexicalization model was designed to produce natural-sounding collocations and multiword expressions, and to be as generic as possible. However, the range of collocations it was able to produce was limited to the most common patterns. The system we present here extends this coverage by a very large margin.

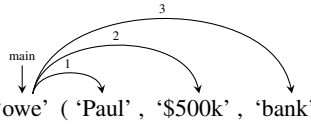
3. GenDR's architecture

GenDR runs on MATE, a graph transducer (Bohnet et al., 2000; Bohnet and Wanner, 2010). It consists of a graph grammar and some dictionaries, which we discuss below.

3.1. Input and output structures

The input to GenDR is a semantic structure *à la* Meaning-Text Theory (MTT) (Mel'čuk, 2012), which is a graph representation of first-order logical form where predicates are linked to their arguments by relations labelled with numbers indicating the arguments's position in the predicate. One of the meanings in the structure has to be flagged as the most salient meaning in the sentence. It is the dominant node of the sentence's rheme/focus (Mel'čuk, 2001) and it will be mapped to the syntactic root.

(1)



The semantic structure in (1) is a simplified visualization of the actual input structure, which is as follows:

```
structure Sem debt {
  S {
    owe {
      tense=PRES
      1-> Paul {class=proper_noun}
      2-> "$500K" {class=amount}
      3-> bank {number=SG definiteness=DEF}}
    main-> owe}}
```

The output of the system is a set of surface syntactic dependency structures (Mel'čuk, 1988). For example, with the input in (1), GenDR produces the six structures in (2).

(2) a. Paul owes $500k to the bank

b. Paul owes a debt of $500k to the bank

c. Paul has a debt of $500k to the bank

d. Paul's debt to the bank amounts to $500k

e. Paul's debt to the bank stands at $500k

f. Paul's debt to the bank totals $500k

For better readability, the sentences shown here are fully inflected and linearized, but the actual outputs are unordered trees with lemmas and grammatical features. They are to be fed to a surface realizer that computes the inflected forms and word order. GenDR focuses on the deeper tasks of lexicalization and arborization, and as one can see from this example, it offers a high level of lexico-syntactic flexibility. Between the semantic and surface syntactic levels of representation, there is a third, intermediate level: the deep syntactic structure (Mel'čuk, 1988). At this level, there are only meaningful lexemes and support verbs linked by two major types of relations: complementation (labelled with Roman numerals) and modification (labelled ATTR). Again, these structures are ordered here only for readability.

(3) a. PAUL OWE $500K BANK

b. PAUL OWE DEBT $500K BANK

c. PAUL HAVE DEBT $500K BANK

d. PAUL DEBT BANK AMOUNT $500K

e. PAUL DEBT BANK STAND $500K

f. PAUL DEBT BANK TOTAL $500K

3.2. Multilingual grammar

At its core, GenDR is a multilingual grammar. We have directly borrowed the core rules from MARQUIS (Lareau and Wanner, 2007; Wanner et al., 2009; Wanner et al., 2010), which dealt with Catalan, English, French, Polish, Portuguese and Spanish. We kept only the most basic rules, that described very general phenomena like simple lexicalization, complementation, modification, etc. Almost all of these rules are shared across languages, while a few language-specific rules model grammatical phenomena like auxiliaries, determiners and so on.

The mapping between semantic graphs and surface syntactic structures takes place in two steps, each handled by a different module of the grammar: the semantic module maps the input semantic structures onto deep syntactic structures (Mel'čuk, 2013), while the syntactic module maps deep syntactic structures to surface syntactic ones (Mel'čuk, 1988) – a layered architecture directly borrowed from MTT (Mel'čuk, 1973; Kahane, 2003; Milićević, 2006).

The semantic module contains 21 core rules, most of which were adapted from MARQUIS, and 132 bound lexicalization rules (see §5.4.) implementing lexical functions; they are documented in (Lambrey and Lareau, 2015; Lambrey, 2016). The syntactic module is much lighter, with only 20 rules, 12 of which are language-independent. Each rule models a linguistic phenomenon, and the grammar relies heavily on rich dictionaries.

4. Arborization

As shown above, one node in the input structure has to be marked as the main node, i.e., the most salient. This is because a semantic graph has no inherent hierarchy. Consider for instance the structure in (4).

(4) 'small' 'cat' 'lie' 'sofa' 'red'

Any of the predicates 'small', 'lie' and 'red' is a root. In a tree, however, there can only be one root, so we have to pick one. Picking 'small' yields *The cat lying on the red sofa is small*, while picking 'lie' produces *The small cat is lying on the red sofa*, and 'red', *The sofa on which the small cat is lying is red*. This choice can be left to GenDR, but typically one wants to control the communicative organization of the sentence by specifying the main node. The deep syntactic tree is then built in a top-down fashion, starting from this main node and using the semantic graph as a blueprint. This algorithm is inspired by (Wanner and Bateman, 1990; Wanner, 1992; Polguère, 2000) and is similar to the one used in MARQUIS. We provide a step-by-step example in §6.

1. We build the root of the syntactic tree and make it correspond to the main node of the semantic structure. In this step, we only create the node without lexicalizing it, but we do add some constraints to it. The main constraint is that it must be a verb in the indicative mood (though we could add alternative rules to handle, say nominal headlines, or languages with adjectival roots).
2. After a node has been created and constrained, we look for a lexicalization rule that can satisfy these constraints while expressing the desired meaning, cf. §5.

3. After lexicalization has taken place, we look at the edges attached to the corresponding semantic node. Edges leaving it point to its arguments, which must be realized as syntactic complements. Edges entering it lead to predicates that apply to it, which must be realized as modifiers. If we have a modifier, we create a dependent attached with the relation ATTR. If it is a complement, then we look up the government pattern (GP) of the governor in a dictionary, as explained in §5.1. The GP models the mapping between semantic, deep, and surface syntactic actants of a word. It also specifies the part of speech of the complements, as well as prepositions. It can also constrain certain grammatical features (e.g., impose a certain mood on a verbal complement) (Mel'čuk, 1995; Mel'čuk, 2014). This step creates new nodes that have not been lexicalized but that have some constraints. Now, for each of these nodes, we go back to step 2.

5. Lexicalization

Lexicalization in GenDR involves three levels of representation: semantics, deep syntax and surface syntax. The first step is to pick a deep lexical unit to express a given semanteme; this is deep lexicalization (or δ-lexicalization). It introduces meaningful words and support verbs. Then, surface lexemes are chosen to express the deep lexical units; this is superficial lexicalization (or σ-lexicalization). It introduces function words.

GenDR performs six types of lexicalization: simple lexicalization for lexemes, template lexicalization for idioms, bound lexicalization for collocations, class-based lexicalization for proper nouns, numbers and such, fallback lexicalization for unknown words, and grammatical lexicalization for function words. Though idioms and collocations are often conflated in a vague "multiword expressions" category, they really are distinct linguistic phenomena and, accordingly, are treated differently.

To make this text hopefully easier to read, we will refer to the nodes involved in the lexicalization process as α for semantic nodes, β for deep syntactic ones, and γ for the ones in surface syntax. These nodes bear a labelling feature: `sem` for α nodes, `dlex` for β nodes, and `slex` for γ nodes.

Lexicalization is supported by three types of dictionaries: a semantic and a lexical dictionary for each language, and one language-independent dictionary of lexical functions (LFs). All are feature structures represented in a straighforward JSON-like format. We will first briefly introduce these lexical resources before explaining how lexicalization takes place.

5.1. Lexical resources

The **semantic dictionary (semanticon)** of a language maps semantemes onto simple lexemes or idioms in that language. A semanteme may be mapped onto several lexical units, which yields lexical paraphrases. This is not limited to synonymy, but also applies across parts of speech (POS). For example, the meaning 'cause(x, y)' may be realized as *x causes y*, *x is the cause of y*, *y (happens) because of/due to x*, *y is due to x*, etc., so the mapping is 'cause' → {BECAUSE, DUE_{ADJ}, DUE_{ADV}, CAUSE_{N}, CAUSE_{V}, ...}.

The **lexical dictionary (lexicon)** of a language should give detailed information about every lexical unit in that language. Obviously, in practice we have to make do with a subset of that. Both simple lexemes and idioms have entries in this dictionary. An entry provides information about the POS of the word, its diathesis (mapping of its semantic to syntactic actants), its subcategorization (including any constraints it imposes on its actants: POS, preposition, mood, definiteness, etc.), and the collocations it controls. Collocations (e.g., *make a decision*) are described in the entry of the base (in this example, *decision*), which is linked to its collocates via lexical functions (e.g., $\texttt{Oper}_1$(DECISION)=MAKE, see boxed text). Currently, GenDR has basic lexical dictionaries for the ~1500 most common words in English and French (most of which have not been disambiguated yet), and demo dictionaries for Lithuanian (~180 entries from the crime news domain) (Dubinskaite, 2017) and Persian (~60 entries).

The **lexical function dictionary (LF dictionary)** describes the semantics and syntax of ~37,000 simple and (mostly) complex LFs. For example, the LF $\texttt{Oper}_1$ (used for support verbs like *make a decision*) has the description {`dpos=V, gp={1=I, L=II}`} (where *dpos* stands for *deep part of speech* and *gp* for *government pattern*). This means that the collocate returned by $\texttt{Oper}_1$ is a verb that has no (or negligible) meaning in itself (it lacks a `sem` attribute) and that takes the first argument of the base as its first syntactic actant (`1=I`) and the base itself as its second actant (`L=II`). This is equivalent to the algebraic notation $(\frac{\#}{\text{V[1,\#]}})$ proposed by (Kahane and Polguère, 2001). The LF `Magn` has the entry {`sem=Magn, dpos=(Adj|Adv), gp={L=ATTR}`}, equivalent to $(\frac{\#\text{Magn}}{\text{(Adj|Adv)[\#\^{}]}})$. For more details, see (Lambrey and Lareau, 2015; Lambrey, 2016).

Lexical functions (LFs)

Collocations tend to be instances of recurrent patterns across languages. For example, *strong preference*, *gravely ill*, *intense flavour* and *win hands down* are all instances of the same pattern where a base is intensified by a syntactic modifier (the collocate). What defines a collocation is not really its apparent lack of compositionality, but the special relationship that exists between the base and the collocate it selects. This relationship is modeled as a function. Intensification collocations, for instance, are described with the `Magn` function: `Magn`(PREFERENCE)=STRONG, `Magn`(FLAVOUR)=INTENSE, etc. Over the years, ~60 basic LFs have been identified, and they combine to form a large number of complex LFs. For detailed discussions of LFs and their use in NLP, see (Žolkovskij and Mel'čuk, 1967; Mel'čuk, 1995; Mel'čuk, 1996; Mel'čuk, 1998; Wanner, 1996; Apresjan, 2000; Kahane and Polguère, 2001; Apresjan et al., 2002; Mel'čuk, 2007; Polguère, 2007; Jousse, 2010; Mel'čuk, 2014; Lambrey, 2016; Fonseca et al., 2016b).

5.2. Simple lexicalization: lexemes

This is the most basic and most common form of lexicalization. During δ-lexicalization, we look at the entry for α.sem in the semanticon and retrieve the set of lexical units that can express this meaning. For each item in this set, we look at its POS in the lexicon and keep only the ones that match the constraint present on β, if any. We create as many output structures as there are elements left, and set β.dlex to a different lexical unit in each. This is an important source of paraphrasing. During σ-lexicalization, we just recopy β.dlex as γ.slex if it is a simple lexeme; if it is an idiom, we branch to template lexicalization.

5.3. Template lexicalization: idioms

This type of lexicalization is only relevant to σ-lexicalization and may take place regardless of how β was lexicalized during δ-lexicalization. It is triggered when the entry for β.dlex in the lexicon has an attribute idiom. Its value is a feature structure that specifies the type of idiom, the lexemes that make it up, and the syntactic relations between them. Each type of idiom is echoed by a rule in the syntactic module that builds a template subtree with placeholders that will be filled with the lexemes and relation labels found in the lexicon entry's idiom attribute.

5.4. Bound lexicalization: collocations

The most complex type of δ-lexicalization is bound lexicalization, which produces collocations. A collocate is a lexical unit that has a certain meaning only when used in the context of a specific base (some collocates being more restrictive than others). Collocations are best described as LFs in the base's entry in the lexicon, because for a lexicographer, it is much easier to think of all the collocates controlled by a common base than the other way around.

Each LF used in the lexicon must have an entry in the LF dictionary that describes the semantics and syntax of the collocation pattern it corresponds to. The information in the LF dictionary is actually a set of parameters that are used by rules of the semantic module. Similarly to how the syntactic structure of idioms is described in the syntactic module, the LF rules of the semantic module are templates with placeholders for deep lexical units and names of syntactic relations. However, these templates are a lot more intricate and diverse. We have 132 rules for bound δ-lexicalization, each corresponding to a family of LFs, so that in total, our rules, combined with the LF dictionary, implement ~26,000 LFs (some of the ~37,000 LFs described in the LF dictionary have not been implemented in the rules yet). Each LF itself is a generic pattern that may be used to describe thousands of instances of actual collocations in a given language. We do not have the space in this abstract to explain how these rules are implemented; see (Lambrey and Lareau, 2015; Lambrey, 2016). The good news is that the user does not need to understand these rules to use them. All one needs to do is list LFs in the entries of the lexicon, like so:

```
fear_n { ...
  lf={name=Magn value=great}
  lf={name=Oper1 value=have}
  lf={name=CausFunc1 value=instill}
  lf={name=Propt value="out of"} ... }
```

When GenDR encounters a meaning in the input structure that corresponds to the meaning of a LF as per the LF dictionary (say, intensification or causation), it treats it as a potential collocation. It knows where to find the base in the semantic structure because that information is in the LF dictionary. Then it looks for the corresponding LFs in the base's entry in the lexicon. If a value is specified, it uses it to produce idiomatic forms like *dirt cheap*, *sharp increase*, *bleed profusely*, etc. If no value is specified, it falls back to a generic value, selected according to the base's POS (and possibly its semantic properties), to produce generic expressions like *very cheap*, *big increase*, *bleed a lot*.

5.5. Class-based lexicalization: numbers, etc.

There are meanings for which we don't want to have an entry in the semanticon and lexicon because their number is large and their behaviour is predictable: numbers, proper nouns, dates, etc. For such meanings, we use an attribute class on α. Its value points to a matching entry for the class in the lexicon, where information on POS and grammatical features may be provided. When α.class has a value, then α.sem is copied to β.dlex and any feature specified in the lexicon for the class is copied to β. During σ-lexicalization, we copy β to γ as is.

5.6. Fallback lexicalization: unknown words

When α.sem has no entry in the semanticon, and it is not the meaning of a known LF, and α.class is not defined, then we have to guess. If there is an entry in the lexicon for this label, we suppose it was just omitted in the semanticon and proceed with simple δ-lexicalization (taking into account POS constraints that may exist on β)—in other words, meanings with trivial one-to-one δ-lexicalization need not be listed in the semanticon. If the word is not in the lexicon either, then we take a shot in the dark. We recopy α.sem to β.dlex, but we have no way of verifying that it matches the POS constraints on β. If there are such constraints however, we treat the word as if it belonged to that POS. This triggers the default GP for that POS (given in the lexicon). If there are no constraints on β, we bet it is probably a noun and treat it as such. Guessed words are flagged with a special feature in the output structures, so that they can be filtered out or sent to an external program for further processing.

5.7. Grammatical lexicalization: function words

This type of lexicalization only happens in σ-lexicalization. It introduces two types of function words: the ones that express a grammatical meaning appearing as features on β, and the ones that are imposed by the GP of a word. Auxiliaries and determiners are of the first type. They do not appear as nodes in deep syntax, but as grammatical features on the verb or noun they apply to, and they have to be introduced as an extra node in surface syntax. These words belong to closed classes with a small number of items, so we have a specific rule for each of them in a language. Governed prepositions and complementers are of the second type. They also do not appear as nodes in deep syntax. They are introduced in surface syntax as an extra node γ_f between a governing word γ_g and its dependent γ_d. The label γ_f.slex is retrieved from the GP of γ_g.slex in the lexicon.

6. Step-by-step example

In this section, we will show how the semantic graph in (1) gets realized as the deep syntactic trees in (3). The formalism used for GenDR rules is explained in full detail in the accompanying manual, as well as in MATE's manual. Introducing it here again would take too much space and bring little to the discussion, so we will only provide below a textual explanation of what the most important rules do.

6.1. Lexical information

The semanticon provides two lexicalizations for 'owe':

```
owe {lex=owe lex=debt}
```

It does not contain information on 'Paul' (a proper noun) or '$500k' (an amount), both of which are candidates for class-based lexicalization need not be in the semanticon. These nodes are marked in the semantic graph with a `class` attribute. For illustration purposes, we also ommitted 'bank' from the semanticon to trigger fallback lexicalization.

The lexicon contains the following entries:

```
owe {
  dpos=V spos=verb
  gp = { // X owes Y to Z
    1=I 2=II 3=III  // trivial diathesis
    I={dpos=N rel=subj}
    II={dpos=Num rel=dobj}
    III={dpos=N rel=iobj prep=to} } }
debt {
  dpos=N spos=noun
  gp = { // X's debt of Y to Z
    1=II 2=I 3=III  // special diathesis
    I={dpos=Num rel=ncomp prep=of}
    II={dpos=N rel=det case=GEN}
    III={dpos=N rel=ncomp prep=to} }
  lf={name=Oper1 value=have}
  lf={name=Oper13 value=owe}
  lf={name=Func2 value=amount_v_1}
  lf={name=Func2 value=stand_v_2}
  lf={name=Func2 value=total_v} }
```

There are also entries for every lexical unit linked to these two: OF, TO, HAVE, AMOUNT_V**1**, STAND_V**2**, TOTAL_V. Their content, however, is not relevant to our discussion.

The LF dictionary contains entries for each of the LFs. The ones relevant to us are the ones used in DEBT above:

```
Oper1 {  // e.g., X has a debt
  dpos = V
  gp = { 1=I L=II } }
Oper13 { // e.g., X owes a debt to Z
  dpos = V
  gp = { 1=I L=II 3=III } }
Func2 {  // e.g., the debt amounts to Y
  dpos = V
  gp = { L=I 2=II } }
```

The meaning of these instructions is explained in §5.1. Note that these entries are not specific to DEBT: the description provided here for, say, $\texttt{Oper}_1$ is valid for any instance of that LF in any language (*take a walk*, *make a choice*, etc.).

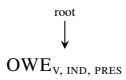
6.2. Semantics ⇒ deep syntax

Now, let us see how (1) gets realized in deep syntax.

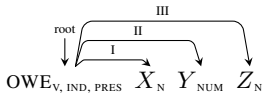
1. A new deep syntactic tree is created with only an empty root in it, marked as corresponding to the main node of (1), 'owe'. It is not lexicalized but it is constrained to be a verb in the indicative mood (at least in English). At this point, the output structure is:



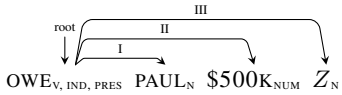
2. We try to lexicalize 'owe' in this syntactic position. The semanticon provides two competing simple lexicalizations, OWE and DEBT, only one of which is a verb and thus compatible with the constraints on the syntactic node. This yields only one simple δ-lexicalization. Now, the output structure gets updated with this label, and the tense feature is carried over:



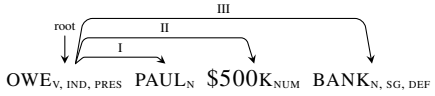
3. We now look for edges leaving 'owe' in the input structure. There are three semantic arguments, that we will realize syntactically according to the diathesis of OWE as encoded in the lexicon (`gp = {1=I 2=II 3=III}`, meaning that the first semantic argument becomes the first syntactic actant, and so on). The GP also constrains actants I and III to be nouns (`I={dpos=N}`, `III={dpos=N}`), and actant II to be a number/amount (`II={dpos=Num}`). Three more nodes are created in the output structure:



4. These nodes need to be lexicalized but none of them has an entry in the semanticon (remember we removed 'bank'), so simple lexicalization is not possible. Since 'Paul' and '$500k' bear a `class` feature, class-based δ-lexicalization applies, as explained in §4. The label of the semantic node is just copied and we check that the constrains on the node match the POS specified in the lexical entries for the classes (not shown here, but they are trivial: proper nouns are subtypes of nouns, and amounts are subtypes of numbers).

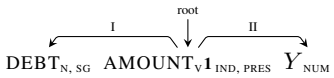


5. Finally, since we have nothing on 'bank' in the semanticon, we use fallback lexicalization. If BANK exists in the lexicon, we can match the POS. If not, then we assume it must be a noun and just copy the input label and carry its grammatical features:

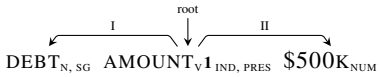


2′. At this point, we have a full deep syntactic structure. But there are alternative lexicalizations that we have not considered. Let us backtrack to step 2. We need a verbal lexicalization and the semanticon only provides

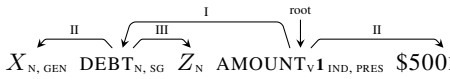
OWE, the other one being DEBT, a noun. However, the lexicon entry for DEBT contains several support verbs, with values for LFs $\texttt{Oper}_1$, $\texttt{Oper}_{13}$ and $\texttt{Func}_2$. This allows the bound lexicalization rules to apply, where a single semantic node, in this case 'owe', is realized in deep syntax by two nodes linked with a specific construction (say, DEBT + AMOUNT_V**1**). Because there are five support verbs and that all of them are compatible with the current state of the output structure, we create five copies of the structure and use a different support verb in each, so that at this point we have six output structures (these five and the simple one from step 2 above. Below, we show only the one for $\texttt{Func}_2$(DEBT)=AMOUNT_V**1**. (The `sg` feature on DEBT is a default value, and the `Num` constraint on Y comes from the GP of AMOUNT_V**1** in the lexicon.)



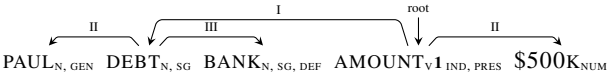
3′. We apply class-based lexicalization, as above, yielding:



4′. There are two semantic arguments of 'owe' that have not been lexicalized, so we open positions for them in the syntactic tree, as in step 3. This time, however, the lexeme controlling the diathesis is DEBT, not OWE. Its diathesis is `gp = {1=II 2=I 3=III}`, meaning that the first semantic argument becomes the second syntactic actant, and vice-versa. The GP also constrains the POS of the actants and imposes the genitive case on its second actant. So, the semantic arguments 1 and 3 of 'owe' become, respectively, the syntactic actants II and III of DEBT. It has no actant I because what could have filled this slot has already been realized as an actant of the support verb. Now, we have:



5′. Finally, class-based and fallback lexicalizations are applied, as in steps 4 and 5 above:



The treatment of $\texttt{Oper}_1$ and $\texttt{Oper}_{13}$ differs only in how some of the semantic arguments are attached to the support verb, as per the patterns described in the LF dictionary.

As one can see, arborization and lexicalization are intertwined: choosing a lexeme imposes a GP, which in turn imposes restrictions on the part-of-speech and potentially other features on its actants, hence restricting what lexemes can be chosen to fill the slot. Support verbs provide a way to fill a verbal slot with a nominal lexeme, thus giving more flexibility in lexico-syntactic choices.

Our top-down approach has a slight bias towards syntax. Every time a node is created, it is immediately constrained (either by the GP of its governor or by the rule that builds the root), limiting the range of lexemes that can be used in a given position. This is desirable of course, to avoid putting, say, a verb where a noun is needed. However, the downside is that there is no guarantee that such a lexicalization exists for a given node that satisfies the constraints, so we might end up with a dead-end structure. It is easy however to filter these out, and since the system usually produces several outputs, there is usually at least one valid output structure.

7. Conclusion and future work

GenDR is a multilingual deep realizer that provides a platform for modeling the semantics-syntax interface in languages. Its salience-driven arborization algorithm ensures high syntactic flexibility. The main strength of this system is its approach to lexicalization, especially its handling of a very wide range of collocation patterns. We have formally described the semantics and syntax of ~37,000 LFs in a dictionary, ~26,000 of which have also been implemented in grammar rules that are ready to be used. This makes it relatively easy to develop resources for new languages that produce texts with rich idiomatic expressions, as the process boils down to lexicography and requires little knowledge of formal grammars.

In lieu of an evaluation, we put this system to the test in a seminar where graduate linguistics students without prior experience in NLG had to generate texts in a domain and language of their choosing. In less than two months, they successfully adapted the system to generate syntactic trees for wedding ring descriptions, speech pathologist reports, weather forecasts and flight details in French, soccer game descriptions in English, and lexicographic illustrative sentences for the lexical field of emotions in Mandarin, all with very satisfying results. To be clear, they only developed the lexical and grammatical resources for the mapping between manually written semantic representations and automatically generated surface syntactic structures, not the whole generation pipeline. But this "experiment" showed that the system is flexible enough to handle very different kinds of texts and can be easily and quickly adapted to a new domain or language. GenDR also proved to be a potent pedagogical tool for teaching formal linguistics and lexicography.

Our coverage of LFs is by far the most comprehensive that we know of, the closest being Lexfom (Fonseca et al., 2016a; Fonseca et al., 2016b; Fonseca, 2018) with ~600, Ayeye (Lareau et al., 2011; Dras et al., 2012; Lareau et al., 2012) with ~220, and MARQUIS with ~30. Our lexical coverage, however, is limited to the ~1500 most common words for English and French, and demo dictionaries for a pair of other languages. Obviously, we will work on increasing these figures. In terms of grammatical coverage, the core phenomena relevant to the semantics-syntax interface (lexicalization, complementation, modification) are implemented, but relative clauses and coordination were hastily implemented and require more serious work. Also, GenDR only properly handles active voice at the moment.

In order to address the problem of grammatical voice, more fine-grained control over the communicative/information structure is needed. In particular, one must be able to control what the theme/topic of the sentence is. A shift of thematic structure has a big impact on the syntactic structure (the theme is often expressed as the grammatical subject in many languages), which impacts lexical choice. We are starting to investigate this aspect.

We have nearly finished weaving VerbNet's (Kipper Schuler, 2006) syntactic frames into our English grammar and we plan to do the same with its French equivalent, Verbəlet (Pradet et al., 2014). This will drastically reduce the amount of work necessary to add new verbs to these languages, and since verbs control most of the syntax of a sentence, it will greatly increase the coverage of our system.

We also plan to map our current output structures to Universal Dependency structures (Nivre et al., 2016) as well as SimpleNLG (English, French and Spanish versions) (Gatt and Reiter, 2009; Vaudry and Lapalme, 2013; Ramos-Soto et al., 2017) for surface realization.

Finally, we want to merge GenDR with FORGe (Mille et al., 2017; Mille and Wanner, 2017), which has fared pretty well on a recent WebNLG challenge (Gardent et al., 2017). Both systems share a common ancestor, so they should be reasonably compatible.

8. Acknowledgements

This project was financed by the Fonds de recherche du Québec—Société et culture (FRQSC) (#2016-NP-191042).

9. Bibliographical References

Apresjan, J. D., Boguslavsky, I. M., Iomdin, L. L., and Tsinman, L. L. (2002). Lexical functions in actual NLP applications. In *Computational Linguistics for the New Millennium: Divergence or Synergy? Festschrift in Honour of Peter Hellwig on the occasion of his 60th Birthday*, pages 55–72. Peter Lang, Frankfurt.

Apresjan, J. (2000). *Systematic lexicography*. Oxford University Press, Oxford.

Basile, V. (2015). *From Logic to Language: Natural Language Generation from Logical Forms*. Ph.D. thesis, University of Groningen.

Bateman, J. A. (1996). *KPML Development Environment*. GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt.

Bohnet, B. and Wanner, L. (2010). Open source graph transducer interpreter and grammar development environment. In *Proceedings of LREC'10*, pages 211–218, Malta.

Bohnet, B., Langjahr, A., and Wanner, L. (2000). A development environment for an MTT-based sentence generator. In *Proceedings of INLG'00*, volume 14, pages 260–263, Mitzpe Ramon.

Bouayad-Agha, N., Casamayor, G., Wanner, L., Díez, F., and López Hernández, S. (2011). FootbOWL: Using a generic ontology of football competition for planning match summaries. In *Proceedings of ESWC'11*, Heraklion.

Bouayad-Agha, N., Casamayor, G., Mille, S., and Wanner, L. (2012). Perspective-oriented generation of football match summaries: Old tasks, new challenges. *ACM Transactions on Speech and Language Processing*, 9(2):art. 3, 1–31.

Miriam Butt et al., editors. (2012). *Proceedings of LFG'12*, Denpasar, Indonesia. CSLI.

CoGenTex, (1998). *RealPro: General English Grammar*. User manual.

Daoust, N. and Lapalme, G. (2015). JSREAL: A text realizer for web programming. In Núria Gala, et al., editors, *Language Production, Cognition, and the Lexicon*, pages 361–376. Springer, Zürich.

Dras, M., Lareau, F., Börschinger, B., Dale, R., Motazedi, Y., Rambow, O., Turpin, M., and Ulinski, M. (2012). Complex predicates in Arrernte. In Butt and King (Butt and King, 2012), pages 177–197.

Dubinskaite, I. (2017). Développement de ressources lituaniennes pour un générateur automatique de texte multilingue. Master's thesis, Université Grenoble Alpes.

Elhadad, M. and Robin, J. (1996). An overview of SURGE: A reusable comprehensive syntactic realization component. In *Proceedings of INLG'96*, number 96-03, pages 1–4, Brighton.

Elhadad, M. (1993). FUF: the universal unifier. User manual version 5.2. Technical report, Computer Science, Ben Gurion University of the Negev, Beer Sheva, Israel.

Fonseca, A., Sadat, F., and Lareau, F. (2016a). Lexfom: a lexical functions ontology model. In *Proceedings of CogALex'16 at COLING*, Osaka.

Fonseca, A., Sadat, F., and Lareau, F. (2016b). A lexical ontology to represent lexical functions. In *Proceedings of LangOnto'16 at LREC*, Portorož.

Fonseca, A. (2018). *Représentation des collocations dans un réseau lexical à l'aide des fonctions lexicales et des formalismes du web sémantique*. Ph.D. thesis, Université du Québec à Montréal.

Gardent, C., Shimorina, A., Narayan, S., and Perez-Beltrachini, L. (2017). The WebNLG challenge: Generating text from RDF data. In *Proceedings of INLG'17*, pages 124–133, Santiago de Compostela.

Gatt, A. and Reiter, E. (2009). SimpleNLG: A realisation engine for practical applications. In *Proceedings of ENLG'09*, pages 90–93, Athens.

Jousse, A.-L. (2010). *Modèle de structuration des relations lexicales basé sur le formalisme des fonctions lexicales*. Ph.D. thesis, Université de Montréal/Université Paris 7.

Kahane, S. and Polguère, A. (2001). Formal foundation of lexical functions. In *Proceedings of ACL 2001*, Toulouse.

Kahane, S. (2003). The Meaning-Text Theory. In Vilmos Ágel, et al., editors, *Dependenz und Valenz: Ein internationales Handbuch der zeitgenössischen Forshung/ Dependency and Valency: An International Handbook of Contemporary Research*, volume 1, pages 546–570. Walter de Gruyter, Berlin/New York.

Kipper Schuler, K. (2006). *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania, Philadelphia.

Lambrey, F. and Lareau, F. (2015). Le traitement des collocations en génération de texte multilingue. In *Actes de TALN'15*, pages 579–585, Caen.

Lambrey, F. (2016). Implémentation des collocations pour la réalisation de texte multilingue. Master's thesis, Université de Montréal.

Lareau, F. and Wanner, L. (2007). Towards a generic multilingual dependency grammar for text generation. In Tracy Holloway King et al., editors, *Proceedings of the GEAF07 Workshop*, pages 203–223, Stanford. CSLI.

Lareau, F., Dras, M., Börschinger, B., and Dale, R. (2011). Collocations in multilingual natural language generation: Lexical functions meet lexical functional grammar. In Diego Mollá et al., editors, *Proceedings of ALTA'11*, pages 95–104, Canberra.

Lareau, F., Dras, M., Börschinger, B., and Turpin, M. (2012). Implementing lexical functions in XLE. In Butt and King (Butt and King, 2012), pages 362–382.

Lavoie, B. and Rambow, O. (1997). A fast and portable realizer for text generation systems. In *Proceedings of ANLP 1997*, pages 265–268, Washington.

Mel'čuk, I. A. (1973). Towards a linguistic "Meaning-Text" model. In Ferenc Kiefer, editor, *Trends in Soviet Theoretical Linguistics*, pages 33–57. Reidel, Dordrecht.

Mel'čuk, I. A. (1988). *Dependency syntax: theory and practice*. SUNY series in linguistics. State University of New York Press, Albany.

Mel'čuk, I. (1995). The future of the lexicon in linguistic description and the explanatory combinatorial dictionary. In I.-H. Lee, editor, *Linguistics in the morning calm*, volume 3. Hanshin, Seoul.

Mel'čuk, I. A. (1996). Lexical functions: A tool for the description of lexical relations in a lexicon. In Wanner (Wanner, 1996), pages 37–102.

Mel'čuk, I. (1998). Collocations and lexical functions. In A. P. Cowie, editor, *Phraseology. Theory, analysis, and applications*, pages 23–53. Clarendon, Oxford.

Mel'čuk, I. A. (2001). *Communicative organization in natural language: the semantic-communicative structure of sentences*. Studies in language companion series. John Benjamins, Amsterdam/Philadelphia.

Mel'čuk, I. A., (2007). *Phraseology: An international handbook of contemporary research*, volume 1, chapter 11: Lexical functions. Walter de Gruyter, Berlin/New York.

Mel'čuk, I. A. (2012). *Semantics: From Meaning to Text*, volume 1. John Benjamins, Amsterdam/Philadelphia.

Mel'čuk, I. A. (2013). *Semantics: From Meaning to Text*, volume 2. John Benjamins, Amsterdam/Philadelphia.

Mel'čuk, I. A. (2014). *Semantics: From Meaning to Text*, volume 3. John Benjamins, Amsterdam/Philadelphia.

Milićević, J. (2006). A short guide to the Meaning-Text theory. *Journal of Koralex*, 8:187–233.

Mille, S. and Wanner, L. (2017). A demo of FORGe: the Pompeu Fabra open rule-based generator. In *Proceedings of INLG'17*, pages 245–246, Santiago de Compostela.

Mille, S., Carlini, R., Burga, A., and Wanner, L. (2017). FORGe at SemEval-2017 task 9: Deep sentence generation based on a sequence of graph transducers. In *Proceedings of SemEval'17*, pages 920–923, Vancouver.

Mille, S. (2014). *Deep stochastic sentence generation: Resources and strategies*. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona.

Molins, P. and Lapalme, G. (2015). JSrealB: A bilingual text realizer for web programming. In *Proceedings of ENGL'15*, pages 109–111, Brighton.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of LREC'16*, pages 1659–1666, Portorož.

Polguère, A. (2000). A "natural" lexicalization model for language generation. In *Proceedings of SNLP 2000*, pages 37–50, Chiangmai.

Polguère, A. (2007). Lexical function standardness. In Leo Wanner, editor, *Selected lexical and grammatical issues in the Meaning-Text theory: in honour of Igor Mel'čuk*, volume 84 of *Language Companion Series*. John Benjamins, Amsterdam/Philadelphia.

Pradet, Q., Danlos, L., and de Chalendar, G. (2014). Adapting VerbNet to French using existing resources. In *Proceedings of LREC'14*, Reykjavik.

Ramos-Soto, A., Janeiro-Gallardo, J., and Bugarín, A. (2017). Adapting SimpleNLG to Spanish. In *Proceedings of INLG'17*, pages 144–148, Santiago de Compostela.

Vaudry, P.-L. and Lapalme, G. (2013). Adapting SimpleNLG for bilingual English-French realisation. In *Proceedings of ENLG'13*, pages 183–187, Sofia.

Wanner, L. and Bateman, J. A. (1990). A Collocational Based Approach to Salience-Sensitive Lexical Selection. In *Proceedings of INLG'90*, Dawson, PA.

Wanner, L. and Lareau, F. (2009). Applying the Meaning-Text Theory model to text synthesis with low- and middle-density languages in mind. In Sergei Nirenburg, editor, *Language Engineering for Lesser-Studied Languages*, volume 21 of *NATO Science for Peace and Security Series - D: Information and Communication Security*. IOS Press, Amsterdam.

Wanner, L., Brügmann, S., Diallo, B., Giereth, M., Kompatsiaris, Y., Pianta, E., Rao, G., Schoester, P., and Zervaki, V. (2009). PATExpert: Semantic processing of patent documentation.

Wanner, L., Bohnet, B., Bouayad-Agha, N., Lareau, F., and Nicklaß, D. (2010). MARQUIS: Generation of user-tailored multilingual air quality bulletins. *Applied Artificial Intelligence*, 24(10):914–952.

Wanner, L., Bott, S., Bouayad-Agha, N., Casamayor, G., Ferraro, G., Graën, J., Joan Casademont, A., Lareau, F., Mille, S., and Vidal, V. (2011). Paraphrasing and multilingual summarization of patent claims. In Leo Wanner, et al., editors, *PATExpert: A Next Generation Patent Processing Service.* Technical documentation.

Wanner, L. (1992). Lexical Choice and the Organization of Lexical Resources in Text Generation. In *Proceedings of the European Conference on Artificial Intelligence*, Vienna, Austria.

Leo Wanner, editor. (1996). *Lexical functions in lexicography and natural language processing*. John Benjamins, Amsterdam/Philadelphia.

Weißgraeber, R. and Madsack, A. (2017). A working, non-trivial, topically indifferent NLG system for 17 languages. In *Proceedings of INLG'17*, pages 156–157, Santiago de Compostela.

White, M., (2008). *OpenCCG Realizer Manual*.

Žolkovskij, A. and Mel'čuk, I. (1967). O semantičeskom sinteze. *Problemy kibernetiki*, 19:177–238.

A Detailed Evaluation of Neural Sequence-to-Sequence Models for In-domain and Cross-domain Text Simplification

Sanja Štajner[1,*], **Sergiu Nisioi**[2,*]
[1] DWS Research Group, University of Mannheim, Germany
[2] Human Language Technologies Research Center, University of Bucharest, Romania
sanja@informatik.uni-mannheim.de, sergiu.nisioi@fmi.unibuc.ro

Abstract

We present a detailed evaluation and analysis of neural sequence-to-sequence models for text simplification on two distinct datasets: Wikipedia and Newsela. We employ both human and automatic evaluation to investigate the capacity of neural models to generalize across corpora, and we highlight challenges that these models face when tested on a different genre. Furthermore, we establish a strong baseline on the Newsela dataset and show that a simple neural architecture can be efficiently used for in-domain and cross-domain text simplification.

Keywords: neural text simplification, sequence-to-sequence models, evaluation

1. Introduction

The aim of text simplification (TS) is to transform given texts into their syntactically and/or lexically simpler variants which are more understandable for the target population (e.g. children, non-native speakers, people with low literacy levels, or people with various kinds of cognitive or reading impairments). It is usually applied on the sentence level and encompasses three major simplification operations: sentence splitting, deletions, and lexical paraphrases (Xu et al., 2016).

In recent years, the problem of automated sentence simplification has often been addressed as the monolingual machine translation (MT) task of translating from original to simple sentences. The MT models used were, however, adapted to the specificities of the TS task, e.g. by adding phrasal deletions to the standard phrase-based MT model (Coster and Kauchak, 2011) to account for a common sentence shortening and phrasal deletions in TS, or by reranking the output of the phrase-based MT model (Wubben et al., 2012), since in the standard phrase-based MT model applied on TS, the first hypothesis tends to leave the input unchanged (Specia, 2010; Coster and Kauchak, 2011). Until recently, the state of the art for automated text simplification was the syntax-based machine translation system (SBMT) with specific optimizations for TS (Xu et al., 2016), such as the use of a large paraphrase database (PPDB) to boost the coverage of the phrasal simplifications, and the use of SARI, a tuning metric that particularly rewards simplicity (Xu et al., 2016).

Following more recent advancements in machine translation using neural networks, we proposed a neural text simplification system, which significantly outperformed the state of the art on various evaluation metrics for the Wikipedia dataset (Nisioi et al., 2017). Our model was constructed as a vanilla encoder-decoder architecture with global attention and input feeding. More recently, a neural network model fine-tuned using reinforcement learning (Zhang and Lapata, 2017) has been proposed for text simplification, the authors reporting several improvements over the previous systems. Given that they were proposed around the same time, the two neural TS models have not yet been directly compared.

One commonly raised issue with most supervised TS systems is the usage of English Wikipedia – Simple English Wikipedia (EW–SEW) sentence-aligned corpora for training the systems, especially since the quality of SEW for modeling TS has often been disputed (Amancio and Specia, 2014; Štajner et al., 2015; Xu et al., 2016). Another parallel corpus of original news articles and their manually simplified versions on four different complexity levels, following strict guidelines and quality control – the Newsela corpus (Newsela, 2016)[1] – has been recently released. Xu *et al.* (2016) show that it has better potential than the EW–SEW dataset for the TS task. However, up until recently, the Newsela corpus was only provided with alignments at the document level.

We use the freely available software[2] for sentence-alignment across different Newsela levels (Štajner et al., 2017; Štajner et al., 2018), and then train neural text simplifications models on the sentence-aligned Newsela and EW–SEW (Hwang et al., 2015) datasets. We compare our systems with the SBMT system (Xu et al., 2016) and the recently proposed state-of-the-art reinforcement learning NTS model (Zhang and Lapata, 2017) to show that a simple neural architecture can be efficiently used for in-domain and cross-domain TS.

Last but not least, we provide a detailed human and automatic evaluation of neural sequence-to-sequence models trained and tested in-domain and cross-domain on each of the two corpora, to discuss the ability of these models to generalize across registers.

2. Methodology

In this section, we describe our neural text simplification (NTS) models, the datasets used for training and testing, and the evaluation procedures.

* Both authors have contributed equally to this work

[1] Freely available for research upon request at `Newsela.com`

[2] `https://github.com/neosyon/SimpTextAlign`

2.1. Models

Following the success of neural sequence-to-sequence models in TS (Nisioi et al., 2017), our simplification systems are based on neural networks (Graves, 2012) with global attention in combination with input feeding (Luong et al., 2015). We use the *OpenNMT* framework (Klein et al., 2017) to train and build an architecture with two LSTM layers (Hochreiter and Schmidhuber, 1997), 500 hidden units, embedding size of 300, and 0.3 dropout probability (Srivastava et al., 2014). We train the model for 14 epochs, regardless of the dataset used, with stochastic gradient descent optimizer and a learning rate decay of 0.7 starting from epoch 7. To be able to have comparable results in-domain and across multiple corpora, we do not use pre-trained embeddings. Several changes made in the meantime for the *OpenNMT* framework and other third party libraries trigger different results than the ones reported by Nisioi et al. (2017). To be able to have a comparable overview across systems, we set the beam size to 12 and re-generate the output of our systems. The models trained on Wikipedia together with the outputs on the same corpus are publicly released.[3]

It may be the case that this type of sequence-to-sequence architecture has by now become a standardized vanilla model for machine translation (Bojar et al., 2017). Nevertheless, for text simplification, we notice a few particularities that helped improve the learning. First of all, for our datasets, we do not use sub-word models (Sennrich et al., 2015; Luong and Manning, 2016), since English does not present rich morphological agglutinations. Secondly, we observe that a size of 300 for internal word embeddings is enough for both Newsela and Wikipedia datasets, the system producing lexical changes without the use of any external information. And lastly, we note the importance of keeping a reduced size of the vocabulary - no more than 50,000 words. On the one hand, this limits the amount of low frequency words that the model learns in order to produce lexical simplifications, but on the other hand, it ensures the grammaticality and meaning preservation of the output, by simply re-using words from the source sentence.

For simplifying sentences, we use beam search to sample four hypotheses corresponding to the given input. As shown in our previous work (Nisioi et al., 2017), the first hypothesis generated by a sequence-to-sequence model is not always the most relevant for text simplification. The reason behind this is that the model, by default, tends to preserve the meaning and the words from the input, whereas, the hypotheses with lower likelihood scores tend to present a greater degree of content reduction and lexical changes. This is further corroborated by the human and automatic evaluation presented in Section 3. More explicitly, we model the hypothesis number as a hyper-parameter that we select after the model finished training. For each model trained on Wikipedia and on Newsela, we generate predictions on each individual development set, obtaining 4 hypotheses corresponding to the training-test data pairs. When predicting on the test set, we choose the hypothesis based on the maximum average SARI score of that hypothesis number on the development set, as shown in Table 1.

[3] https://github.com/senisioi/NeuralTextSimplification

	Train		
	Newsela	**Wikipedia**	
Dev set	SARI score		**Hyp.**
Newsela	29.71	29.55	1
	38.89	33.40	2
	39.34	33.48	3
	39.25	**33.79**	4
Wikipedia	30.99	31.19	1
	35.10	**36.39**	2
	35.81	35.71	3
	35.54	36.23	4

Table 1: SARI scores on Newsela and Wikipedia predictions on the development set by models trained on the two datasets.

2.2. Datasets

2.2.1. Newsela Datasets

Newsela offers original news articles and their manual simplifications at four different levels of simplification. We automatically sentence-aligned the English part of the Newsela corpus across different levels using the freely available, recently released CATS software particularly made for this purpose (Štajner et al., 2017; Štajner et al., 2018). A thorough human evaluation performed on over 3,000 sentence pairs showed that the accuracy of automatic alignment between the sentences from two neighboring Newsela levels of simplification is between 83% (for levels 3 and 4) and 98% (for levels 0 and 1), while the alignment method aligns sentences from the hardest and the easiest levels (0 and 4), with only 58% accuracy (Štajner et al., 2017). Therefore, we aligned and used only the neighboring Newsela levels (0–1, 1–2, 2–3, and 3–4) as training data. The alignment procedure has been shown to successfully align sentences with full semantic overlap, which can be used to model lexical and syntactic paraphrases, as well as those with only partial semantic overlap, which can be used to model deletion and addition operations (Štajner et al., 2018). The CATS software also provides '1-n' alignments, which can be used to model sentence splitting, where appropriate (Štajner et al., 2018).

The Newsela corpus is organized in unique files based on the topics being addressed, covering approximately 2,000 topics. We split our training, development, and test data disjointly based on the topic files, ensuring that the sentences from the same story (regardless of their complexity levels) never appear in both the training and test data. The exact size of each subset is provided in Table 2. For testing purposes, we use the common sentences from the alignments between 0–1, 0–2, 0–3, and 0–4 to create multiple simplification hypotheses for more accurate calculation of automatic evaluation scores. Both the development references and the test references used to report the scores in Tables 1 and 4 are created from these types of alignments.

	#topic files	#sentences	#tokens O	#tokens S
train	1,835	298,814	5,791,417	5,823,546
dev	56	9,372	180,682	181,742
test	19	655	17,132	17,506

Table 2: The size in terms of topic files, sentences, and tokens for the original (O) and simplified (S) versions of the dataset that we used for training, testing, and developing our Newsela models.

	#sentences	#tokens O	#tokens S
train	284,677	7,400,499	5,634,834
dev	16,000	349,944	308,856
test	359	8,110	7,957

Table 3: Statistics regarding the number of sentences and tokens for the original (O) and simplified (S) versions of Wikipedia dataset used to train our models.

2.2.2. Wikipedia Datasets

Our Wikipedia dataset consists of the latest sentence-aligned version (Hwang et al., 2015) based on manual and automatic alignments between standard English Wikipedia and Simple English Wikipedia (EW–SEW). We discard the uncategorized matches, and use only *good matches* and *partial matches* which were above the 0.45 threshold (Hwang et al., 2015), totaling to 280K aligned sentences (around 150K full matches and 130K partial matches). Unlike the earlier EW–SEW version[4] (Kauchak, 2013) which only contains *full matches* (167K pairs), the newer dataset that we use also contains *partial matches*, and is thus not only larger, but also allows for learning sentence shortening transformations. From this dataset, we remove those sentence pairs whose original sentences are present in the Wikipedia test set compiled by Xu et al. (2016). We also opt for this test set, as it contains, for each of the 359 original sentences, eight manually simplified versions that can be used as multiple references for more accurate calculation of automatic evaluation scores. Statistics regarding the size of the Wikipedia datasets are rendered in Table 3. Unlike the Newsela datasets, the Wikipedia datasets do not contain examples of sentence splitting, as the original EW–SEW dataset (Hwang et al., 2015) only contains one-to-one sentence alignments.

2.3. Evaluation Procedures

2.3.1. Automatic Evaluation

BLEU (Papineni et al., 2002) is a standardized metric for machine translation evaluation that reports a similarity score between the output of a system and the 'gold standard' references. In this paper, we report BLEU with NIST smoothing as implemented in NLTK (Bird et al., 2009). One downside of this score for text simplification, however, stems from the sole comparison of the output against references without considering the initial sentence. Based on this idea, a metric that compares system output against references and against input - SARI (Xu et al., 2016), has been proposed to reward additions, copying, and deletions from the input that are present in the output and references. Unlike BLEU, the SARI score has been shown to better predict the simplicity of the output (Xu et al., 2016).

To account for all three aspects (grammaticality, meaning preservation, and simplicity) of the output sentences, we report both BLEU and SARI scores. Nevertheless, both those automatic metrics are used only as additional evaluation metrics, while the discussion is based solely on a detailed human evaluation.

2.3.2. Human Evaluation

For human evaluation, we follow the procedure proposed in our recent paper (Nisioi et al., 2017) and report:

- The percentage of sentences which undergone at least one change;
- The total number of changes;
- The percentage of correct changes;
- Grammaticality of the simplified sentence;
- Meaning preservation of the simplified sentence;
- Relative simplicity of the simplified sentence in comparison to the original sentence.

The total number of changes counts the number of changes regardless of their type (lexical changes, syntactic changes such as phrase reordering or sentence splitting, deletions, and additions). In the case of phrasal substitutions, the changes of a whole phrase (e.g. "*become defunct*" → "*was dissolved*") are counted as one change. In the case of content reduction (deletion), we instructed the annotators to count the deletion of each array of consecutive words as one change. The count was performed by two native English speakers. The sentences for which the two annotators did not agree were given to a third annotator to obtain the majority vote.

Those changes that preserve the original meaning and grammaticality of the sentence (assessed by two native English speakers) and, at the same time, make the sentence easier to understand (assessed by two non-native fluent English speakers) are marked as *Correct*. Given that our systems were trained to model not only full lexical and syntactic paraphrasing, but also content reduction (due to the partial matches in our training datasets), we instructed the annotators to consider the meaning unchanged if the main information of the sentence was retained and unchanged. The sentences for which the two annotators did not agree were given to a third annotator to obtain the majority vote.

Grammaticality (*G*) and meaning preservation (*M*) of the simplified sentences were assessed by three native English speakers using a 1–5 Likert scale (1 → very bad; 5 → very good). The final scores were computed as the arithmetic mean of the scores by the three annotators. Only those sentences which have undergone at least one modification are taken into account for calculating the G and M scores. This way, we make sure that the systems which leave many input sentences unchanged do not get rewarded for that and result in higher G and M scores, as the sentences which are left unchanged always get the highest G and M score (they are

[4] http://www.cs.pomona.edu/~dkauchak/simplification/

grammatically correct and have exactly the same meaning as the original sentence).

The simplicity (*S*) score was assigned by three non-native fluent English speakers. The annotators were shown original (reference) sentences and target (output) sentences, one pair at the time, and asked whether the target sentence is: +2 → much simpler; +1 → somewhat simpler; 0 → equally difficult; -1 → somewhat more difficult; -2 → much more difficult, than the reference sentence. The final simplicity score was calculated as the arithmetic mean of the scores assigned by all three annotators.

We did not explicitly instruct the annotators regarding the influence of grammaticality on the simplicity score. The post-evaluation analysis revealed that ungrammatical and meaningless sentences were penalized by receiving a negative simplicity score by all three annotators. Therefore, if one was to chose just one evaluation measure to compare the performances of different TS systems, the simplicity score assigned in this manner would probably be the right choice. However, as the annotators were not instructed to take into account meaning preservation while assigning the simplicity score, the meaning preservation scores of the systems would have to be additionally checked.

Here is also important to note that while the correctness of changes takes into account the influence of each individual change on the grammaticality, meaning preservation and simplicity of a sentence, the G, M, and S scores take into account the mutual influence of all changes within a sentence.

3. Results and Discussion

We first explore the BLEU and SARI scores on the first four hypothesis in both in-domain and cross-domain scenarios (Section 3.1). Next, we evaluate (automatically and manually) our NTS models for in-domain and cross-domain text simplification (Section 3.2), comparing the models which always choose the default first hypothesis (as the baselines of our models) with those that use the SARI score to choose the best hypothesis. Finally, we compare the performances of our best models with the performances of the state-of-the-art SBMT model (Xu et al., 2016) and the state-of-the-art reinforcement learning NTS model (Zhang and Lapata, 2017) on the Wiki test set (Xu et al., 2016) in Section 3.3.

3.1. Automatic Evaluation of Hypotheses

We automatically evaluate models trained on each of our two datasets, in a pairwise fashion, against each test set. Table 4 contains the results of the automatic evaluation for each beam search hypothesis from 1 to 4 (last column).

On the one hand, if we focus on the BLEU evaluation score, we notice that the first hypothesis, the one most likely given the beam search score, always obtains the highest BLEU score, regardless of the training-test pairs. On the other hand, if we focus on the TS-specific metric SARI (Xu et al., 2016), the best scores are never obtained by the first hypotheses, but rather by the ones with lower probability and less content from the input. This is expected given that SARI especially rewards the output which is the least similar to the input. For example, the following sentence:

	Train				
	Newsela		Wikipedia		
Test	**BLEU**	**SARI**	**BLEU**	**SARI**	**Hyp.**
Newsela	**77.06**	28.21	**64.16**	30.81	1
	71.66	37.06	59.14	33.69	2
	70.51	**38.84**	57.81	33.67	3
	70.31	37.76	58.43	**34.0**	4
Wikipedia	**89.49**	30.33	**84.69**	30.54	1
	84.75	35.00	77.57	**35.78**	2
	83.8	**36.48**	77.21	35.67	3
	83.57	36.15	75.77	35.76	4

Table 4: SARI and BLEU scores on Newsela and Wikipedia predictions by models trained on the two datasets.

- *In its pure form, dextromethorphan occurs as a white powder.*

as an input to the model trained on the Wikipedia dataset, generates the following four hypotheses:

1. *In its pure form, dextromethorphan occurs as a white powder.*
2. *Dextromethorphan occurs as a white powder.*
3. *Dextromethorphan is a white powder.*
4. *It is a white powder.*

For this particular short sentence, the first hypothesis is identical to the input, whereas the lower likelihood hypotheses 2, 3 and 4 present more traits of simplification.

3.2. Sequence-to-Sequence Models

The evaluation scores for our in-domain and cross-domain NTS systems, both with default ranking of the hypotheses and with reranking of the hypotheses according to the SARI score on the dev set are reported in Table 5.

3.2.1. Reranking

We can notice that reranking of the output according to the SARI metric (instead of using the default first hypothesis h1) always leads to a significant increase in percentage of sentences that were changed (up to more than three times more sentences changed in the case of in-domain TS on the Newsela dataset), and higher grammaticality (G) and meaning preservation (M) scores. In most of the cases, it also leads to a substantial increase in simplicity score (S) and in percentage of correct changes. The most striking difference between the system that chooses the default hypothesis h1 and the one that chooses the hypothesis with the best SARI scores on the dev set is achieved in the case of in-domain TS on the Newsela dataset. One potential reason for this might be that our Newsela training data contains only the consecutive alignments (e.g. 0–1, 1–2, 2–3, and 3–4) making the model learn small changes that appear from one level to another.

The only case in which the system with the default hypotheses h1 outperforms the system with the reranked output is the cross-domain TS where the systems were trained on the Wikipedia dataset and tested on the Newsela dataset. In

Domain	Training-Test	Rerank.	Hypothesis	changed sent.	Changes		Scores		S	SARI	BLEU
					Total	Correct	G	M			
In	News - News	default	h1	27.1%	23	21.7%	4.52	2.31	+0.02	28.21	**77.09**
In	News - News	SARI	h3	**90.0%**	**76**	**54.1%**	**4.97**	**3.87**	**+0.50**	**38.84**	70.51
In	Wiki - Wiki	default	h1	41.4%	37	48.6%	4.59	3.41	+0.30	30.54	**84.69**
In	Wiki - Wiki	SARI	h2	**87.1%**	**78**	**59.0%**	**4.77**	**4.05**	**+0.49**	**35.78**	77.57
Cross	Wiki - News	default	h1	47.1%	46	**28.3%**	3.87	2.55	**+0.23**	30.81	**64.16**
Cross	Wiki - News	SARI	h4	**85.5%**	**77**	25.0%	**4.48**	**3.52**	+0.21	**34.00**	58.43
Cross	News - Wiki	default	h1	40.0%	37	18.9%	3.86	2.90	+0.04	30.33	**89.49**
Cross	News - Wiki	SARI	h3	**97.1%**	**102**	**23.4%**	**4.34**	**3.19**	**+0.28**	36.48	83.80

Table 5: Human evaluation results on the first 70 test sentences (the highest scores obtained on each test set by each evaluation criterion are shown in bold) and automatic evaluation measures (BLEU and SARI) on the full test sets.

Approach	changed sent.	Changes		Scores		S	SARI	BLEU
		Total	Correct	G	M			
Our NTS - training on Wiki - h2 (best SARI)	87.1%	78	**59.0%**	**4.77**	**4.05**	**+0.49**	35.78	77.57
Our NTS - training on News - h3 (best SARI)	**97.1%**	102	23.4%	4.34	3.19	+0.28	36.48	**83.80**
SBMT (SARI+PPDB) (Xu et al., 2016)	82.9%	**143**	34.3%	4.28	3.57	+0.03	**38.59**	73.62
Dress-LS (Zhang and Lapata, 2017)	67.1%	63	42.9%	4.27	3.80	+0.14	32.74	81.16

Table 6: Human evaluation results on the first 70 test sentences (the highest scores obtained on each test set by each evaluation criterion are shown in bold) and automatic evaluation measures (BLEU and SARI) on the full test sets of our two best systems (in-domain and cross-domain) and the state-of-the-art systems on the Wiki test set.

this case, the default hypotheses led to a slightly higher percentage of correct changes and a slightly higher simplicity gain (S). Nevertheless, the achieved grammaticality (G) and meaning preservation (M) scores were substantially lower for the system with the default hypothesis h1.

The examples *1a–1d* and *2a–2c* in Table 8 illustrate the cases in which the default hypotheses h1 fails to make any changes to the input, while the best SARI ranked hypotheses improve the simplicity of the sentence by introducing correct changes.

3.2.2. In-domain vs. Cross-domain Simplification

When we compare the performances of our NTS systems on in-domain and cross-domain tasks, we notice that, in both cases, they result in similar percentages of changed sentences. However, the cross-domain TS results in substantially lower percentage of correct changes and substantially lower simplicity gain (S). The cross-domain NTS also achieves lower grammaticality (G) and meaning preservation (M) scores. Two examples of ungrammatical output obtained by the cross-domain NTS, and one example of wrong lexical substitution applied in cross-domain NTS are shown in Table 8 (examples *3d–3e*, and *4d* for the ungrammatical output, and example *2c* for the wrong lexical substitution). In all three cases, the in-domain NTS resulted in grammatical output.

3.2.3. Wikipedia vs. Newsela Datasets

The systems with the default hypotheses h1 achieve higher evaluation scores for the Wikipedia than for the Newsela dataset. We can only explain this phenomenon by the amount of variety from the two training sets: the Wikipedia corpus contains a high variety of topics that can appear in both training and test datasets, while the Newsela training set contains a fixed amount of stories that are repeated for different levels of simplification. For example, the training data may contain exactly the same sentence both as original and simplified example. What for levels 0–1 can be an example of simplification, for levels 1–2 it is a complex sentence that needs to be further simplified, and so on. In this case, the default hypothesis will likely be biased. Therefore, we believe it is essential to sample different hypotheses from the model in order to get multiple sources of truth. If the hypotheses with the best SARI scores on the dev set are used instead, then our NTS systems perform equally well on both domains.

3.3. Comparison with the State of the Art

Given that the SBMT system (Xu et al., 2016) and all its components are not freely available, and the Newsela splits used for the Dress-LS systems (Zhang and Lapata, 2017) were not available at the time of our experiments, we were able to directly compare our systems with those two state-of-the-art systems only on the Wiki test set (Xu et al., 2016), for which the outputs of both those systems are freely available.

The results of the manual evaluation (Table 6) show that on the Wiki test set, our in-domain NTS model (trained on the Wikipedia dataset) outperforms both state-of-the-art TS systems (SBMT and Dress-LS). It results in higher percentage of sentences which undergone at least one change, higher percentage of correct changes, and higher grammaticality, meaning preservation and simplicity scores. More importantly, the difference in the obtained simplicity gain (S) is substantial (+0.49 as opposed to +0.03 and +0.14, respectively), as well as the difference in grammaticality and meaning preservation scores.

Train-Test	Split	Short	NE
Wiki–Wiki	0	24	2
Wiki–News	0	18	2
News–News	3	10	20
News–Wiki	5	15	3

Table 7: The number of sentence splittings, sentence shortenings (removal of at least five consecutive words, or an apposition, or a part of a noun phrase), and wrong NE replacements in the first 70 test sentences.

Interestingly, even our cross-domain NTS system (with the reranking of hypotheses according to the SARI scores on the dev set) performs better than the state-of-the-art TS systems. It outperforms both systems (SBMT and Dress-LS) by the number of sentences which undergone at least one change (97.1% instead of 82.9% and 67.1%, respectively) and by the simplicity gain (+0.28 as opposed to +0.03 and +0.14, respectively), while achieving similar grammaticality of the output (4.34 as opposed to 4.28 and 4.27, respectively). Although it has a lower percentage of correct individual changes, the output of our NTS model trained on Newsela achieves much higher simplicity gain (S) than the SBMT and Dress-LS systems, indicating thus that those fewer correct changes still have significant impact on the simplicity of the output. This is probably due to the sentence shortenings and sentence splittings learned by our NTS model (see Table 7).

For illustration, Table 8 contains two examples in which our NTS systems perform better than the SBMT and Dress-LS systems (examples *1a–1d* and *6a–6e*). The examples *3b* and *4b* show the wrong lexical substitutions performed by the SBMT system which led to low grammaticality and meaning preservation scores for that system. In example *5*, the Dress-LS system performed one correct lexical substitution which our NTS systems did not, but at the same time, due to the sentence shortening, an important piece of information was lost.

4. Further Analysis

To better understand some phenomena noticed during the manual evaluation, which are specific for our NTS approach and datasets, we count the number of sentence splittings, sentence shortenings and wrong named entity (NE) replacements on all our train-test combinations (Table 7).

The NTS models trained on the Newsela dataset were able to learn sentence splitting operations, unlike the models trained on the Wikipedia dataset. The second is not surprising, given that the Wikipedia dataset (Hwang et al., 2015) does not allow for one-to-many sentence alignments and therefore does not contain sentence splitting examples. However, when trained on a dataset that contains examples of sentence splittings (the Newsela dataset), our NTS models were able to learn this simplification operation and successfully apply it on a test set from either the same domain (examples *2a–2c*, Table 8) or from another domain (examples *5a–5b*, Table 8).

All our NTS models were able to successfully apply sentence shortening in all train-test combinations. The models trained on the Wikipedia dataset performed more sentence shortenings, which is probably due to the fact that the Wikipedia dataset contains an abundant amount of partial matches (which are good training material for sentence shortening), which is not the case in the Newsela training dataset which consists only of the sentence pairs from the neighboring levels.

The number of sentence splittings and sentence shortenings obviously reflects the type of training data and the way it was collected. Nevertheless, it is important that our NTS models seem to be able to learn whichever type of sentence transformation is present in the training dataset and apply it even on another domain and text genre.

The high number of NE errors found in the NTS models trained and tested on the Newsela dataset probably reflects the facts that: (1) news contain an abundant amount of named entities; and (2) we did not allow for the same topics/news stories in the training and test Newsela datasets, thus creating the ideal opportunity for the unseen NEs in the test set. As the Wikipedia dataset does not have the information about the exact article from which the sentence pair was taken, we could not use the same constraints on the Wikipedia datasets. That is probably the reason for the much lower number of NE errors by the NTS systems trained and tested on the Wikipedia dataset.

However, the large number of NE errors made by the NTS models trained and tested on the Newsela dataset does not seem to have greatly influenced the overall performance of the NTS systems (see Table 6). We believe this is due to the fact that wrong entity substitutions in a sentence do not damage its grammaticality as much as wrong substitutions of other words (see examples *1c*, *3b*, *4b*, and *6e* in Table 8). Since one wrong entity replacement does not make a sentence more difficult to understand (rather it changes its meaning) and sentence splitting significantly simplifies a sentence, the NTS model trained on the Newsela dataset was better ranked than the SBMT and Dress-LS systems for its relative simplicity to the original (the S score). The wrong substitutions clearly affect the meaning, and this is reflected in the results presented in Table 6, where our NTS system trained on the Newsela dataset obtained a lower M score than the SBMT and Dress-LS systems.

5. Conclusions

In recent years, text simplification was often modeled as the monolingual machine translation task of translating original sentences into their simpler variants. Following the latest trends in machine translation, several text simplification models with neural architecture were proposed last year and they became the state of the art in English TS.

In this study, we focused on a sequence-to-sequence model to investigate its performance for in-domain and cross-domain text simplification, providing detailed automatic and human evaluations. As expected, the in-domain NTS models achieve better SARI scores and the human evaluation scores compared to those of the cross-domain NTS models.

According to our results, neural networks prove once again to be powerful tools to model text simplification, presenting significant improvements over the earlier proposed syntax-

Ex.	System	Output
1a	Original-W, NTS-h1-any	She remained in the United States until 1927 when she and her husband returned to France.
1b	NTS-SARI-any	She **stayed** in the United States until 1927 when she and her husband returned to France.
1c	SBMT	She *is still* in the United States until 1927 when she and her husband returned to France.
1d	Dress-LS	She stayed in the United States until 1927 when she *was married* to France.
2a	Original-N, NTS-h1-any	Both newcomers and advanced learners trained together, but those with more experience were given more challenging training.
2b	NTS-SARI-in	Both newcomers and advanced learners trained together**. However,** those with more experience were given more challenging training.
2c	NTS-SARI-cross	Both newcomers and advanced *atheists* trained together, but those with more experience were given more challenging training.
3a	Original-W, NTS-h1-cross	Disney received a full-size Oscar statuette and seven miniature ones, presented to him by 10-year-old child actress Shirley Temple.
3b	SBMT	Disney won a full-size Oscar statue and seven *the mini*, *made* to him by *10 year child* actress Shirley Temple.
3c	NTS-h1-in	*It was* presented to him by 10-year-old child actress Shirley Temple.
3d	NTS-SARI-in, Dress-LS	Disney received a full-size Oscar statuette and seven miniature ones.
3e	NTS-SARI-cross	Disney received a full-size Oscar statuette and seven miniature ones**. They** *presented* to him by 10-year-old child actress Shirley Temple.
4a	Original-W, Dress-LS NTS-h1-cross, NTS-SARI-in	At the Voyager 2 images Ophelia appears as an elongated object, the major axis pointing towards Uranus.
4b	SBMT	At the Voyager 2 images Ophelia seems as an elongated object, the main axis *show up on* Uranus.
4c	NTS-h1-in	At the Voyager 2 images Ophelia appears as **a stretched** object, the major axis pointing towards Uranus.
4d	NTS-SARI-cross	At the Voyager 2 images Ophelia, the major axis pointing towards Uranus.
5a	Original-W, NTS-h1-out, SBMT	Graham attended Wheaton College from 1939 to 1943, when he graduated with a BA in anthropology.
5b	NTS-SARI-cross	Graham attended Wheaton College from 1939 to 1943**. He** graduated with a BA in anthropology.
5c	NTS-SARI-in, NTS-h1-in	Graham graduated from Wheaton College from 1939 to 1943.
5d	Dress-LS	Graham **went** to Wheaton College from 1939 to 1943.
6a	Original-W, NTS-h1-any	As a result, although many mosques will not enforce violations, both men and women when attending a mosque must adhere to these guidelines.
6b	Dress-LS	As a result, although many mosques will not enforce violations, both men and women.
6c	NTS-SARI-cross	As a result, many mosques will not enforce violations, both men and women when attending a mosque must **follow** these guidelines.
6d	NTS-SARI-in	As a result, although many mosques will not enforce violations, both men and women when attending a mosque must **stick** to these guidelines.
6e	SBMT	As a result, while many mosques will not *meet the breach*, both men and women when **go to** a mosque must **meet** these guidelines.

Table 8: Simplification examples on the sentences from Wikipedia and Newsela (Newsela, 2016). Correct changes are presented in bold, and the incorrect changes in italics.

based machine translation (SBMT) model. Furthermore, we show that a simple approach to sample multiple hypotheses from a vanilla encoder-decoder can outperform a more complex neural text simplification model tuned with reinforcement learning (Dress-LS), on all human evaluation metrics.

We acknowledge that more work is needed to make sequence-to-sequence models flexible enough for handling out-of-vocabulary words, especially in a cross-domain text simplification. However, neural TS systems were still able to produce grammatical output and correctly model sentence splittings and sentence shortenings even across different text genres.

Our work revealed the challenges that these models face when training and predicting cross-domain, as well as their capacity to correctly perform significant content reduction and improve over the existing text simplification systems.

Acknowledgements

We would like to thank Newsela for making their corpora available, and to Luke Orland for his comments and suggestions. This work has been partially supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI – UEFISCDI, project number PN-III-P2-2.1-53BG/2016, within PNCDI III.

6. Bibliographical References

Amancio, M. A. and Specia, L. (2014). An Analysis of Crowdsourced Text Simplifications . In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 123–130.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media, Inc., 1st edition.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., et al. (2017). Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214.

Coster, W. and Kauchak, D. (2011). Learning to Simplify Sentences Using Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1–9.

Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hwang, W., Hajishirzi, H., Ostendorf, M., and Wu, W. (2015). Aligning Sentences from Standard Wikipedia to Simple Wikipedia. In *Proceedings of NAACL&HLT*, pages 211–217.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1537–1546. ACL.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Luong, M.-T. and Manning, C. D. (2016). Achieving open vocabulary neural machine translation with hybrid word-character models. *arXiv preprint arXiv:1604.00788*.

Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421. The Association for Computational Linguistics.

Newsela. (2016). Newsela article corpus. `https://newsela.com/data`. Version: 2016-01-29.

Nisioi, S., Štajner, S., Ponzetto, S. P., and Dinu, L. P. (2017). Exploring neural text simplification models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 85–91.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Specia, L. (2010). Translating from complex to simplified sentences. In *Proceedings of the 9th international conference on Computational Processing of the Portuguese Language (PROPOR)*, volume 6001 of *Lecture Notes in Computer Science*, pages 30–39. Springer Berlin Heidelberg.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Štajner, S., Bechara, H., and Saggion, H. (2015). A Deeper Exploration of the Standard PB-SMT Approach to Text Simplification and its Evaluation. In *Proceedings of ACL&IJCNLP (Volume 2: Short Papers)*, pages 823–828.

Štajner, S., Franco-Salvador, M., Ponzetto, S. P., Rosso, P., and Stuckenschmidt, H. (2017). Sentence alignment methods for improving text simplification systems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 97–102.

Štajner, S., Franc-Salvador, M., Rosso, P., and Ponzetto, S. P. (2018). CATS: A Tool for Customised Alignment of Text Simplification Corpora. In *Proceedings of the 11th Language Resources and Evaluation Conference (LREC)*.

Wubben, S., van den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL): Long Papers - Volume 1*, pages 1015–1024. Association for Computational Linguistics.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Zhang, X. and Lapata, M. (2017). Sentence Simplification with Deep Reinforcement Learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594.

Don't Annotate, but Validate: a Data-to-Text Method for Capturing Event Data

Piek Vossen, Filip Ilievski, Marten Postma, Roxane Segers
Vrije Universiteit Amsterdam
De Boelelaan 1105, 1081 HV Amsterdam
The Netherlands
{piek.vossen, f.ilievski, m.c.postma, r.h.segers}@vu.nl

Abstract

In this paper, we present a new method to obtain large volumes of high-quality text corpora with event data for studying identity and reference relations. We report on the current methods to create event reference data by annotating texts and deriving the event data a posteriori. Our method starts from event registries in which event data is defined a priori. From this data, we extract so-called Microworlds of referential data with the Reference Texts that report on these events. This makes it possible to easily establish referential relations with high precision and at a large scale. In a pilot, we successfully obtained data from these resources with extreme ambiguity and variation, while maintaining the identity and reference relations and without having to annotate large quantities of texts word-by-word. The data from this pilot was annotated using an annotation tool created specifically in order to validate our method and to enrich the reference texts with event coreference annotations. This annotation process resulted in the *Gun Violence Corpus*, whose development process and outcome are described in this paper.

Keywords: event coreference, text corpora, structured data

1. Introduction

Events and entities are central to referential semantics. Semantic parsing of news articles not only concerns detecting the meaning of words and their relations, but especially establishing the referential relations to the outside world. For entities, it is straightforward what this referential world is. However, compared to entities, events are less tangible (Guarino, 1999; Hovy et al., 2013) for various reasons: 1. we use a small vocabulary to name events, which results in large referential ambiguity 2. events are more open to interpretation and framing, which leads to more variation in making reference 3. events are less persistent in time than entities 4. each event has many idiosyncratic properties, e.g. unique participants playing different roles in a unique spatio-temporal context, making generalization harder. Due to these properties, textual data on events is more fragmented than textual data on entities. In news, events are mentioned during a very short period of time and they rapidly lose their news value (except for a few events like *9/11*), whereas popular entities tend to be mentioned across different texts over longer periods of time. We thus do not find a typical Zipfian distribution for events, with a few events that dominate the news (the head) and a long tail of low-frequent events, but a more even low-frequent distribution.

Given this fragmented distribution, it is not surprising that NLP tasks on event detection, event relation detection, and event coreference are difficult, which is reflected by relatively low inter-annotator-agreements and small amounts of data with event annotations.

Most corpora with event annotations do not consider how they relate to the same or similar events in the world, e.g. PropBank (Kingsbury and Palmer, 2002), NomBank (Meyers et al., 2004), and FrameNet (Baker et al., 2003). In these corpora, syntactic structures are taken as the starting point and all predicates mark mentions of events. Similarity of events follows from the assigned event type, the meaning of the word or frame, and from having similar argument structures. These corpora, however, lack a notion of event reference and are not very well-suited for studying the different ways we describe the same or similar events. For the latter purpose, specific event coreference corpora have been created: ECB+ (Cybulska and Vossen, 2014), RED (O'Gorman et al., 2016), among others. Event coreference annotations have been created using what we call a **text-to-data (T2D)** approach. In the T2D approach, annotators start from the text and first decide what phrases are labeled as event mentions after which different event mentions are related to each other through an event coreference relation. The coreference relations establish event identity across event mentions a posteriori by chaining event mentions that share coreference relations. Due to the complexity and labor-intensity of this T2D approach, only a limited amount of referential event data has been created so far (see Table 1).

The research on event coreference faces a data bottleneck because it is both too difficult and too costly to gather sufficient data following the traditional T2D method. We therefore present a novel semi-automatic method, called **structured-data-to-text (D2T)** to address this data problem. Instead of deriving event identity a posteriori after the text annotation, this approach starts from event registries (structured data on events) in which the events are defined a priori and the texts that report on these events are mapped to the event data. By preserving the relation between the world and texts reporting on it, D2T allows us to create large volumes of referential event data in a more efficient way, with high agreement, and capturing more variation in the language making reference.

The paper is structured as follows. In Section 2., we analyze the current state for the event coreference data collected so far and summarize the major issues and bottlenecks. In Section 3. we explain our proposal to follow a D2T approach and give an overview of the potential data archives that can be used. Section 4. reports on a pilot study to cre-

ate an event coreference corpus, called the *Gun Violence Corpus (GVC)*, following this method. An annotation tool was specifically designed to pair structured data with reference texts and annotate the event mentions per incident. We analyzed the annotation results and efforts in terms of volume, speed, agreement and variation of referring expressions. Finally, we conclude and discuss our future plans in Section 5.

2. From text to data

In this Section, we first give an overview and the overall statistics of the most studied datasets for event coreference, which are all created using a T2D method. Next, we discuss their annotation process.

2.1. State of T2D datasets

In Table 1, we present an overview of the text corpora that dominated the event coreference research in the last decade. The Table shows the number of documents in each dataset, the number of mentions of events, and the number of so-called coreference clusters (groups of mentions that refer to the same event). The final column indicates if the coreference clusters span across documents (cross-document coreference) or only within a single document (within-document coreference). We observe that the number of documents and mentions is small for both within- and cross-document relations: less than four thousand documents and less than forty thousand mentions in total (10 mentions per document on average). The ratios between mentions and clusters vary considerably, which is due to the different ways in which the datasets have been compiled: either subsets of the sentences and/or event types were annotated or all mentions in a full article.

Cross-document data is more sparse than within-document data, as can be seen in Table 1. ECB+ (Cybulska and Vossen, 2014) therefore extended the Event Coreference Bank (ECB) (Bejan and Harabagiu, 2008; Lee et al., 2012) from 482 articles to 982 articles by including more events of the same type. This slightly increased the referential ambiguity and variation, but, nevertheless, only a few sentences per article were annotated (1.8 sentences per article on average in ECB+). The Rich Entities, Relations and Events corpus (Song et al., 2015) and the Richer Event Description corpus (O'Gorman et al., 2016) are two recent initiatives to manually create similar annotations for *all* sentences in articles and also partially across documents, but the number of documents covered is small.

We analysed the referential annotations in a number of these datasets, revealing that they, despite efforts such as the creation of ECB+, hardly reflect referential ambiguity and show very little variation (Cybulska and Vossen, 2014; Ilievski et al., 2016). For example, ECB with 482 documents contains 8 news articles on one specific murder, but since there are no other murders in the dataset, searching for the word "murder" results in almost all mentions of that specific incident with high accuracy: one-form-one-referent and one-referent-one-form. Cybulska and Vossen (2014) demonstrated that the so-called lemma baseline to establish coreference relations[1] scores already very high in this dataset and is difficult to beat by state-of-the-art systems. From the perspective of a real-world situation and the many different ways in which events can be described and framed in language, these datasets are far too sparse and do not reflect true ambiguity and variation. Partly due to this lack of data and variation, automatic event coreference detection has made little progress over the years, especially across documents (Chen and Ji, 2009; Bejan and Harabagiu, 2010; Lee et al., 2012; Liu et al., 2014; Peng et al., 2016; Lu and Ng, 2016; Vossen and Cybulska, 2016).

All data listed in Table 1 are created according to the T2D approach: a selection of text is made and interpreted by annotators who add an annotation layer. Creating data following a T2D approach is expensive and labor-intense, as all mentions of events need to be cross-checked against all other mentions across documents for coreference relations. With the size of the data, the effort increases exponentially.

2.2. State of text-to-data guidelines

Besides meta-level choices on what needs to be annotated, guidelines and annotations tend to differ in criteria for deciding on the text span to be annotated as a mention. In some cases, complete phrases (e.g. "inflicted a fatal gunshot wound") or even whole sentences are annotated, whereas in other cases only semantic main verbs ("pull the trigger") are annotated; implicit events ("murderer", "killer", "victim") are included or excluded; coreference, subset and subevent relations are lumped together ("attack", "pointing a weapon", "shootings", "5 shots", "4 hits one fatal", "3 injured, 1 killed, 1 died in the hospital"); quantification of events is or is not ignored (the phrase "the 2 earthquakes" refers to two different earthquake events) or generic events ("measles is a deadly disease") are excluded; only realis events are annotated or also irrealis events; aspectual verbs ("begin", "stop", "continue", "happen", "take place") are sometimes seen as events and sometimes not; adjectival or adverbial modifiers ("fatal accident") are not marked, etc. Such choices are based on a priori criteria regardless of the types of events annotated and they tend to vary depending on the specific task for which the data were annotated e.g. semantic role detection (Kingsbury and Palmer, 2002), detecting temporal and causal event relations (Boguraev et al., 2007; Pustejovsky and Verhagen, 2009; Bethard et al., 2015; Caselli and Morante, 2016), or event coreference relations (Hovy et al., 2013).

Besides the differences in guidelines, annotators following guidelines may also have different interpretations, which may lead to relatively low inter-annotator-agreement and conservative annotation strategies. Due to the complexity of the task, annotators may for example stay on the safe side and create identity relations only when the same word is used, hence eliminating variation. Such difficulties in defining events, event relations, and event coreference have led to the creation of the KBP2015 dataset (Mitamura et al., 2015) in which a weaker definition of an event has been applied, so-called Event Nuggets, to ease the annotation and the task for establishing coreference relations. In the KBP2015 dataset, "attack", "shooting", and "murder" do not represent separate event instances, but are considered

[1] all occurrences of the same word, e.g. "murder", mention a single unique event and hence are coreferential

Table 1: Event coreference corpora for English created by a text-to-data method

Name	Reference	nr. docs	nr mentions	mention/ docs.	nr clusters	mention/ cluster	cross doc.
ACE2005	(Peng et al., 2016)	599	5268	8.79	4046	1.30	NO
KBP2015	(Mitamura et al., 2015)	360	13113	36.43	2204	5.95	NO
OntoNotes	(Pradhan et al., 2007)	1187	3148	2.65	2983	1.06	NO
IC	(Hovy et al., 2013)	65	2665	41.00	1300	2.05	NO
EECB	(Lee et al., 2012)	482	2533	5.26	774	3.27	YES
ECB+	(Cybulska and Vossen, 2014)	982	6833	6.96	1958	3.49	YES
MEANTIME	(Minard et al., 2016)	120	2096	17.47	1717	1.22	YES
EER	(Hong et al., 2016)	79	636	8.05	75	8.48	YES
RED	(O'Gorman et al., 2016)	95	8731	91.91	2390	3.65	YES
Total		3874	36292	9.37	15057	2.41	
GVC	this publication	510	7298	14.31	1411	5.17	YES

as mentions of the same underspecified event represented at a more coarse-grained level of granularity, so-called event-hoppers.

In all the T2D approaches described above, event reference is established a posteriori after annotating texts word-by-word and sentence-by-sentence to mark events and event relations. Figure 1 gives a schematic overview for the T2D approach that indirectly constructs a referential representation from annotated mentions of events and participants. First, event and participant mentions need to be annotated in a single document and after that all these annotations need to be compared across all news articles to establish cross-document coreference. The more documents are included in the data set, the more comparisons need to be made. In the next section, we propose a new method that starts from registered events that are given a priori when annotating event references in texts so that we only need to compare mentions across relevant documents.

3. From data to text

For the reasons discussed in the previous Section, T2D methods do not provide the means nor the datasets to study *identity*, *reference* and *perspectives* of events on a large scale, since they are too small and lack sufficient ambiguity and variation. We therefore propose a novel *structured-data-to-text* (D2T) methodology, based on the notions **microworlds** and **reference texts**. *Microworlds* are structured representations of referents related to specific world events (e.g. human calamities or economic events). *Reference texts* are documents reporting on this data, e.g. news articles, blogs, and Wikipedia pages. In the D2T method, we start from some event registry that has been created by people a priori by hand and is publicly available as structured data. From these registries, we derive microworld representations of the unique event instances, their participants, location and date as a referential graph, as shown in Figure 2. Assuming that reference texts mainly refer to the corresponding microworld and not to other events and participants, we can establish the referential relation relatively easily and partially automatically.

By combining microworlds for similar but different events with their paired reference texts, we increase the referential ambiguity for systems that need to reconstruct the microworld from the texts, hence approximating the complexity of reference relations in reality across large volumes of text. By collecting news from different sources on the same or similar events, we approximate true variation in making reference from different perspectives. Furthermore, the fact that the data on events from which we start has been created from the perspective of general human interest (e.g. gun violence incident reports) avoids the never-ending discussion on what establishes an event in text. More practically, the D2T method is much less labor-intensive than T2D, because a rich and consistent set of event properties and links to its supporting documents are often provided within a microworld by the original data author. Finally, since the underlying data is often created manually, its quality is very high.

3.1. Desiderata

Our method operates best on resources with: 1. *links between structured data and reporting texts* 2. *disambiguated/unique and consistently defined* events and event properties following Linked Data principles 3. *open, available* data 4. *high volume*, since more data typically exhibits higher referential ambiguity and variation. If all four desiderata are fulfilled, the conversion of the data to microworlds and reference texts is a matter of writing data manipulation scripts. In practice, resource properties are often not ideal, thus requiring some additional work - however, the amount of annotation or retrieval needed is far lower/incomparable to the exhaustive annotation processes in T2D.

3.2. Resource availability

Table 2 provides description of several public resources that satisfy most of the desiderata. The resources register event incidents with rich properties such as participants, location, and incident time, and they provide pointers to one or more reference texts. The number of events and documents is usually high, for instance there are $\sim$9K incidents in RA, and $\sim$231K incidents in GVA.

In a pilot, we successfully obtained data from these resources with extreme ambiguity and variation, while maintaining the identity and reference relations and without having to annotate large quantities of texts word-by-word. Following the D2T method, we obtained over ten thousand news articles and over five thousand incidents from GVA and FR. The data from this pilot was used as basis for a referential quantification task entitled "counting events and

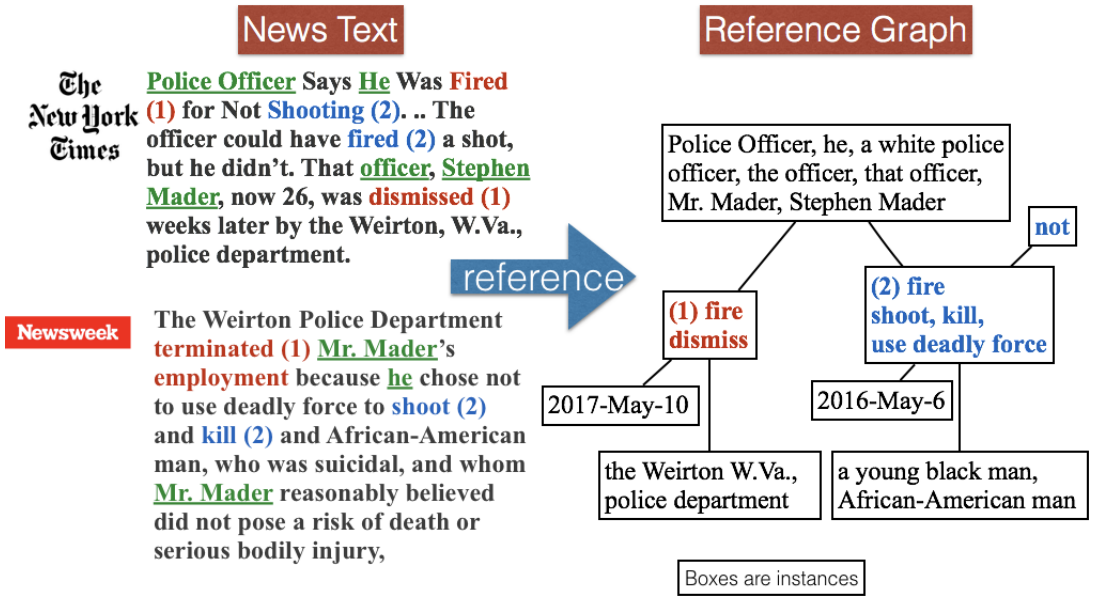


Figure 1: Overview of the T2D method: deriving a referential graph from mentions across different news text.

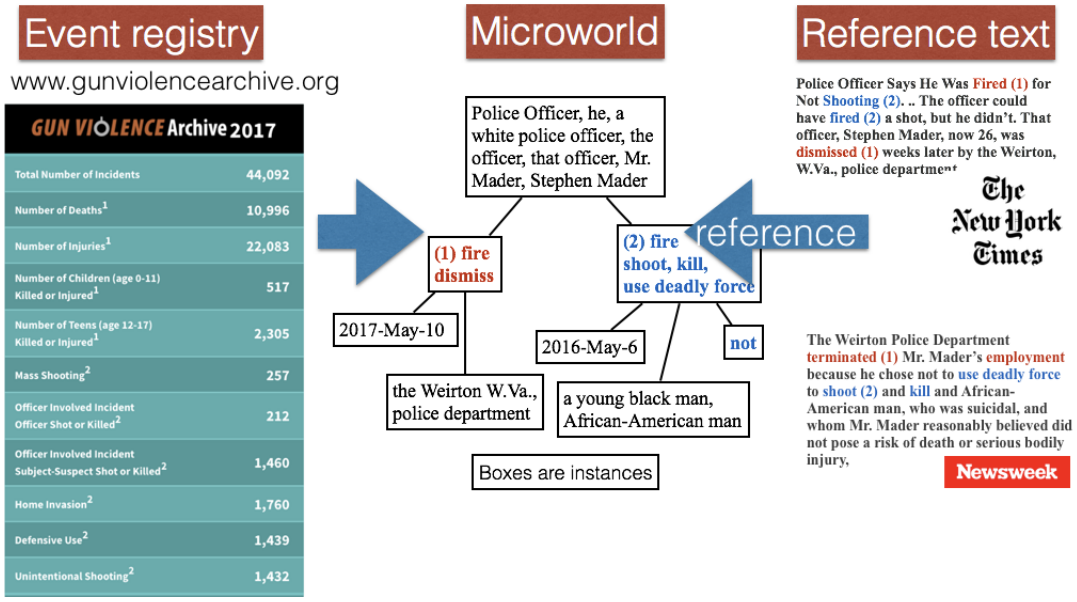


Figure 2: Overview of the D2T method: representing structured event data first as microworlds and secondly pairing it with reference texts.

participants within highly ambiguous data covering a very long tail", hosted at SemEval-2018.[9]

In addition to the resources presented in Table 2, data with similar properties can be obtained from Wikipedia and structured databases such as Wikidata[10], Yago2[11], and DBpedia[12] with little effort. This can either be done through direct extraction, or through smart querying of the data (Elbassuoni et al., 2010; Knuth et al., 2015; Hewlett et al., 2016). For example, a simple query on Wikidata for event instances belonging to certain event classes (i.e. explosion, crime, natural disaster, accident, sport, election), already yields over 70k events with structured data (type of event, location and time) that can form the basis for creating microworlds. Many of these events can be traced back to Wikipedia pages, that describe these events in textual form. Such Wikipedia pages often include further links to news articles as references to substantiate the information given. By using Wikipedia as the glue between the structured microworld data and the reference texts, one can obtain a reliable mapping of texts with framings and representations of the referential events.

[9] https://competitions.codalab.org/competitions/17285

[10] https://query.wikidata.org

[11] http://www.yago-knowledge.org

[12] http://events.dbpedia.org/ns/core#Event

Table 2: Potential event data for extracting microworlds and reference texts. Numbers marked with '*' are estimates.

Name	Topic	Structured data	Nr docs	Nr incidents	From year	To year	Locations	Reference texts
ASN incident database[2]	aircraft safety occurrences	fatalities, locations, time, other domain data	32K	21K	1919	2017	world	news, reports, social media
ASN Wikibase[3]	aircraft safety occurrences	fatalities, locations, time, other domain data	310K	207K	1905	2017	world	news, reports, social media
Fire Incident Reports (FR)[4]	fire disasters	publishing time and location	1K	1K	2004	present	USA	reports
Global nonviolent action DB[5]	social justice/protests	incident location and time	*6K	1K	1955	present	world	various
Gun Violence Archive (GVA)[6]	gun violence	fatalities, locations, time, participant roles, weapon information	*462K	231K	2012	present	USA	news
Legible news[7]	science, sports, business, economics, law, crime, disasters, accidents, ...	/	*20K	*15K	2014	present	world	news
Railways Archive (RA)[8]	railway accidents	casualties, locations, time, vehicle operators	5K	9K	1803	present	UK, Ireland	news
TOTAL			***836K**	***485K**				

4. The Gun Violence Corpus

D2T resources provide a very valuable link between microworlds and reference texts. We hypothesize that this link is also very useful for mention annotation of reference texts, because the microworld information provides a summary of the incidents reported. Annotation of mentions can then be seen merely as a task of marking evidence for the incident and its characteristics in the supporting text documents. Following this new view on the annotation process, we created the Gun Violence Corpus on top of our pilot data. This Section describes the details of its development.

4.1. Annotation Task and Guidelines

The Gun Violence Corpus (GVC) consists of 241 unique incidents for which we have structured data on a) location, b) time c) the name, gender and age of the victims and d) the status of the victims after the incident: killed or injured. For these data, we gathered 510 news articles following the D2T approach. The structured data and articles report on a variety of gun violence incidents, such as drive-by shootings, murder-suicides, hunting accidents, involuntary gun discharges, etcetera.

The documents have been manually annotated for all mentions that make reference to the gun violence incident at hand. More specifically, the annotation process involved three basic steps:

- Annotating the *event type* of every mention that refers to a gun violence incident in the structured data;
- Annotating the *victim(s)* involved in the mention referring to a shooting in the structured data;
- Annotating every mention related to gun violence but NOT referring to the incident in the structured data (*other incidents or generic mentions*).

Based on these annotations, we can infer coreference relations: in case that two or more mentions have the same annotations (event type and victims) AND they both relate to the same incident ID in the structured data, we can infer that these mentions are coreferential.

To further capture the referential complexity and diversity of event descriptions in text, we designed an event schema that captures subevent relations in addition to the above incident references, see Figure 3.

The main event ("the gun incident") is basically a container that can be split into several more fine-grained events that stand in some implication relation to each other. In this case the bag of events consists of five events: *Firing a gun*, *Hitting someone* or *Missing someone*. From *Hitting someone* follows *Injuring* and in some cases *Death*. Apart from these events, many articles also contain references to gun violence in a more general way or not related to the structured data. These have been labeled *Generic* and *Other*.

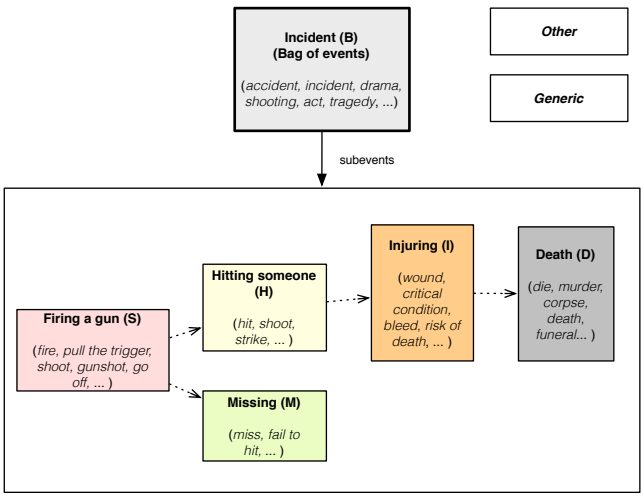


Figure 3: The event scheme used for the annotation of gun violence events.

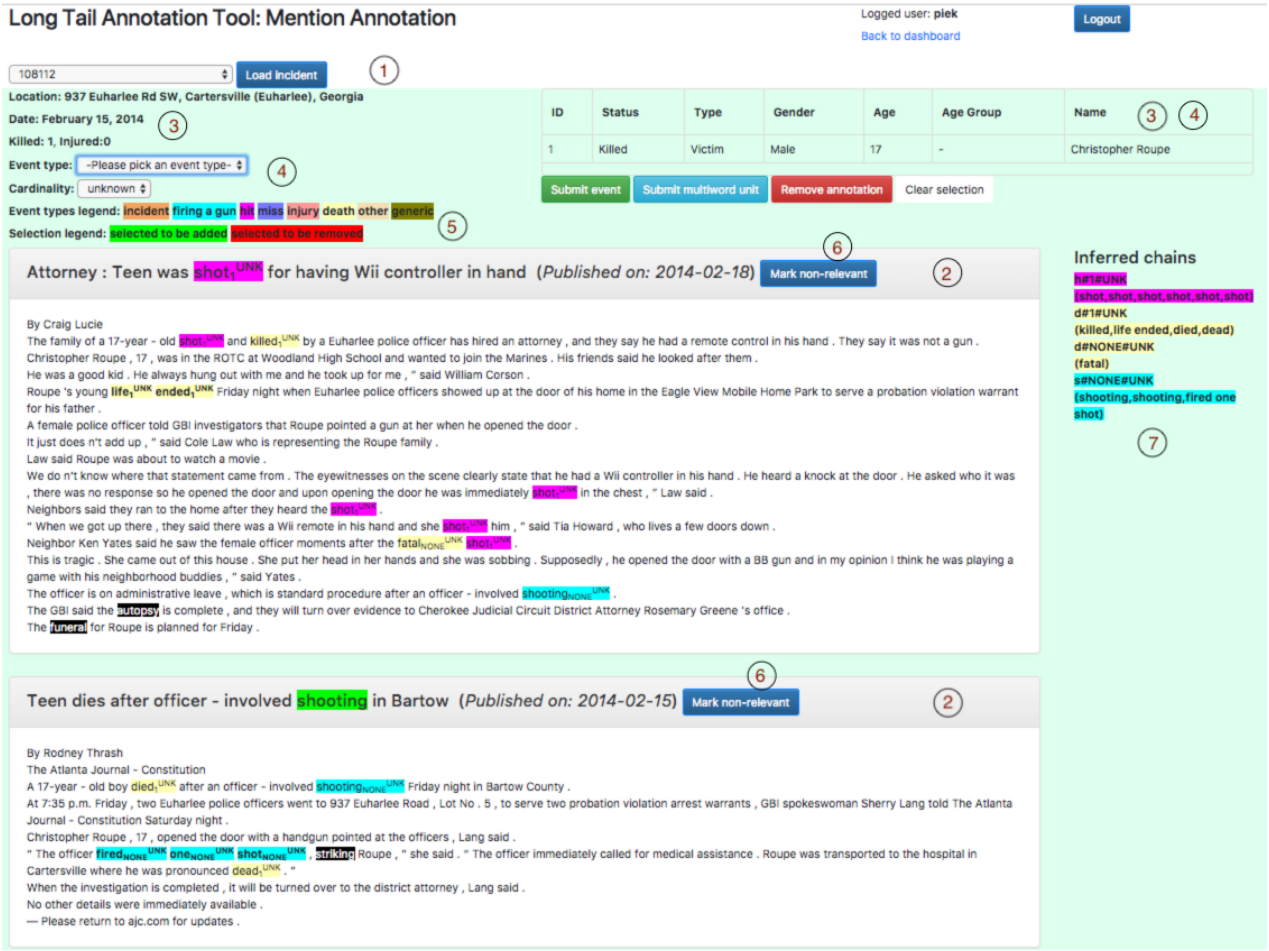


Figure 4: Annotation environment for annotating mentions in Reference Texts related to structured data.

4.1.1. The GVC Annotation Guidelines

We annotated all mentions denoting but also implying one of the predefined event classes. For example, a *funeral*, an *autopsy* or the process of *grieving* imply that someone died. A *shooter* and *killer* imply respectively the event types *Firing a gun* and again *Death* in the context of this domain. Besides annotating verbal and nominal expressions, we also annotated mentions of: other parts of speech (including adjectives and adverbs), idioms, multiword units, and collocations. In principle, we annotated the minimal span of a mention, usually the head, unless this would result in a meaningless annotation, e.g. we would annotate *critical condition* as a multiword unit instead of just the head *condition*.

Additional specification of the annotation decisions, such as: how we handled negation, the irrealis, ellipsis, phrasal verbs, and various cases of implicit event mentions, can be found in the full guidelines of GVC.[13]

The annotation of events has similarities to the Entities Relations Events (ERE) (Song et al., 2015) and the Rich Event Description (RED) (O'Gorman et al., 2016), but also differences. The Bag of Event level annotation corresponds to the event hopper annotation in ERE, but we differentiate specific subevent and subset relations as well, as is done in RED. Our annotation is more restricted to the specific events of this database only. However, it can be extended to other domains by defining a different event schema.

4.2. Annotation environment

To the best of our knowledge, there is no tool that starts from structured event data to annotate event mentions and event coreference relations. We therefore built our own environment for annotating events in reference texts that are related to structured data on an incident.

The goal of the tool is to allow annotators to find evidence in the reference texts for the event properties in the structured data. To support this goal, the tool reads the structured event data and presents the event properties, e.g. time, location, and participants, in a table. Annotators mark the event mentions, select the participants involved and select the type of event. The annotators only need to annotate the mentions of the predefined schema and not all other types of events.

By applying this strategy to all mentions within and across Reference Texts of an incident, we establish coreference and identity across the mentions. Notably, it is not needed to annotate coreference explicitly. Instead, the coreference chains are inferred by the annotation environment, based on the combination of two factors of the individual mention

[13] The full guidelines are available at `https://goo.gl/Yj1Hra`

annotations: event type and participants.
In addition, we have built in lexical support for the annotators, based on the set of already annotated event mentions. Reference text mentions which have been frequently annotated in other texts but not in the current one, are visually suggested to be also annotated. The annotators can then decide whether to accept this suggestion.
Figure 4 provides a screenshot of the mention annotation environment when the incident *108112* is loaded by the user *piek*. The incident selection menu is marked with (1) in the Figure. The selected incident is supported by two reference texts, rendered in the middle of the screen (marked with (2)). Annotators can select one or multiple mentions from this area for annotation. The top panel contains the structured data about the current incident (marked with (3)), followed by menus and a table for annotations of properties for the selected mention (4). Mentions in colors have already been annotated by this user, and the event type is signaled by the color. The color scheme is explained in detail in the legend (5). Moreover, inversely colored mentions (e.g. "funeral" and "autopsy" in Figure 4) are the ones proposed by the tool to be annotated additionally. Annotators can also discard individual documents with the 'Mark non-relevant' button (6). Finally, the area on the right displays the coreferential chains that the tool has inferred so far about the current incident (marked with (7) in the Figure).
The source code of the annotation software is publicly available on Github.[14]

4.3. Annotation process

Two linguistic students were hired to perform the annotations. After completing the training phase, which resulted in some simplifications of the guidelines, the students started with the annotation of the Gun Violence Corpus. In six weeks, the students annotated the 510 documents that are part of the corpus. In addition, 25 documents were selected in order to compute the inter-annotator-agreement (IAA). The first annotator annotated 432 event mentions in this set, whereas the second one annotated 457 event mentions. The annotators provided the same annotation in 350 cases, resulting in a Cohen's kappa coefficient (Cohen, 1960) of 0.72. According to Landis and Koch (1977), a score between 0.61 and 0.80 is considered *substantial*, from which we conclude that there was high agreement between the annotators. For comparison, ECB+ reported a Cohen's kappa coefficient of 0.68 for a similar size and agreement analysis to ours. ECB+ annotators only had to consider 2 incidents per topic with about 10 articles per incident and 1.8 sentences on average per article, whereas in our case, 510 documents need to be annotated for a few hundred incidents. In terms of speed, one annotator averaged 5 minutes per document, whereas the other took 4 minutes to annotate one document on average.
Unlike in T2D, our method scales only linearly instead of exponentially. Namely, to include documents that report on a new incident, one does not need to compare their mentions to all other incidents, since the structured data already guarantees they are not coreferential. In Table 1, we report statistics on the size of our corpus. Although our corpus annotated with mentions is currently smaller than existing datasets, the speed and the linear scalability of our method provide a promise that its size can increase up to the limit posed by the original structured data sources.

4.4. Corpus description

The GVC[15] contains 7,298 mentions, referring to 241 incidents. In total, 510 documents contain at least one mention. Table 3 presents the annotation frequency for each event type.

event type	annotation frequency
Death	2,206
Firing a gun	1,622
Hitting	1,122
Bag of events	755
Injuring	726
Other	596
Generic	270
Missing	2

Table 3: Mention frequency of each event type.

Most mentions in our Gun Violence Corpus refer to the event types *Death* and *Firing a gun*, respectively. In addition, about 4% of all mentions (i.e. 270 mentions), refer to generic uses of shooting and killings. Finally, it is not uncommon that the text refers to other incidents than the main incident of the article, which happens in about 8% of all mentions (i.e. 596). This means that systems can not fully rely on a one-incident-per-document heuristic to detect coreference chains.
Table 4 presents the most used expressions for each event type.

event type	most common expressions
Death	dead (305) died (285) killed (283)
Firing a gun	shooting (680) gunshot (247) went off (72)
Hitting	shot (801) shooting (83) struck (46)
Bag of events	shooting (247) incident (164) it (88)
Injuring	wound (175) injured (75) injuries (68)
Other	shot (105) shooting (70) killed (47)
Generic	accident (57) shooting (13) tragedy (11)
Missing	surgery (1) missed (1)

Table 4: Most common expressions used for event types

As presented in this Table, the most common expressions that are used to refer to event types are covered well in resources such as WordNet. For example, the most common expressions for the event type *Death* can be detected by correctly identifying the WordNet synsets *kill.v.01* (cause to die; put to death, usually intentionally or knowingly) and *killing.n.02* (the act of terminating a life). However, this

[14] https://github.com/cltl/LongTailAnnotation

[15] The corpus can be downloaded at: https://github.com/cltl/GunViolenceCorpus

is not the case for all expressions in the GVC. For example, expressions like *mourn* and *autopsy* that refer to the event type *Death* show that manual and automatic annotators can not fully rely on resources to detect all event types correctly, but that additional reasoning is needed. We analyze the referential potential of this corpus further in Vossen et al. (2018).

5. Conclusions

We discussed the problems in collecting large scale and high-quality text corpora for event extraction, and specifically with respect to identity and reference. We concluded that most data has been created through a text-to-data method, which faces the obstacles of data size and scalability. To circumvent these, we propose a scalable data-to-text method to create far more data with high quality, ambiguity, and variation. Following this method, we created the Gun Violence Corpus, whose development is reported in this paper. We present the specification and the guidelines of the annotation, as well as our annotation environment which was purposefully developed to support mention annotation for data-to-text use cases. Finally, we show that we achieve high agreement and annotation speed, and report statistics of the resulting corpus. For future works, we aim to compare our annotation process to traditional annotation using text-to-data tools such as CAT (Lenzi et al., 2012) to annotate the same documents used in this study.

6. Acknowledgements

The work presented in this paper was funded by the Netherlands Organization for Scientific Research (NWO) via the Spinoza grant, awarded to Piek Vossen in the project "Understanding Language by Machines".

7. Bibliographical References

Bejan, C. A. and Harabagiu, S. (2010). Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1412–1422. Association for Computational Linguistics.

Chen, Z. and Ji, H. (2009). Graph-based event coreference resolution. In *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, pages 54–57. Association for Computational Linguistics.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Cybulska, A. and Vossen, P. (2014). Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 4545–4552.

Elbassuoni, S., Ramanath, M., Schenkel, R., and Weikum, G. (2010). Searching RDF graphs with SPARQL and keywords.

Guarino, N. (1999). The role of identity conditions in ontology design. In *International Conference on Spatial Information Theory*, pages 221–234. Springer.

Hewlett, D., Lacoste, A., Jones, L., Polosukhin, I., Fandrianto, A., Han, J., Kelcey, M., and Berthelot, D. (2016). Wikireading: A novel large-scale language understanding task over wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1545. Association for Computational Linguistics.

Hovy, E., Mitamura, T., Verdejo, F., Araki, J., and Philpot, A. (2013). Events are not simple: Identity, non-identity, and quasi-identity. In *Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 21–28.

Ilievski, F., Postma, M., and Vossen, P. (2016). Semantic overfitting: what 'world' do we consider when evaluating disambiguation of text? In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1180–1191.

Knuth, M., Lehmann, J., Kontokostas, D., Steiner, T., and Sack, H. (2015). The DBpedia Events Dataset. In *International Semantic Web Conference (Posters & Demos)*.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics.

Lenzi, V. B., Moretti, G., and Sprugnoli, R. (2012). CAT: the CELCT Annotation Tool. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC) 2012*. European Language Resources Association (ELRA).

Liu, Z., Araki, J., Hovy, E. H., and Mitamura, T. (2014). Supervised within-document event coreference using information propagation. In *LREC*, pages 4539–4544.

Lu, J. and Ng, V. (2016). Event Coreference Resolution with Multi-Pass Sieves. In *LREC 2016*, pages 3996–4003.

Peng, H., Song, Y., and Roth, D. (2016). Event detection and co-reference with minimal supervision. In *EMNLP*, pages 392–402.

Vossen, P. and Cybulska, A. (2016). Identity and granularity of events in text. *Cicling 2016. Konya, Turkey*.

Vossen, P., Postma, M., and Ilievski, F. (2018). ReferenceNet: a semantic-pragmatic network for capturing reference relations. In *Global Wordnet Conference 2018, Singapore*.

8. Language Resource References

Baker, C. F., Fillmore, C. J., and Cronin, B. (2003). The structure of the framenet database. *International Journal of Lexicography*, 16(3):281–296.

Bejan, C. A. and Harabagiu, S. M. (2008). A Linguistic Resource for Discovering Event Structures and Resolving Event Coreference. In *LREC*.

Bethard, S., Derczynski, L., Savova, G., Pustejovsky, J.,

and Verhagen, M. (2015). SemEval-2015 task 6: Clinical TempEval. In *SemEval@ NAACL-HLT*, pages 806–814.

Boguraev, B., Pustejovsky, J., Ando, R., and Verhagen, M. (2007). TimeBank evolution as a community resource for timeml parsing. *Language Resources and Evaluation*, 41(1):91–115.

Caselli, T. and Morante, R. (2016). Vuacltl at semeval 2016 task 12: A crf pipeline to clinical tempeval. *Proceedings of SemEval*, pages 1241–1247.

Cybulska, A. and Vossen, P. (2014). Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 4545–4552.

Hong, Y., Zhang, T., O'Gorman, T., Horowit-Hendler, S., Ji, H., and Palmer, M. (2016). Building a cross-document event-event relation corpus. *LAW X*.

Hovy, E., Mitamura, T., Verdejo, F., Araki, J., and Philpot, A. (2013). Events are not simple: Identity, non-identity, and quasi-identity. In *Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 21–28.

Kingsbury, P. and Palmer, M. (2002). From TreeBank to PropBank. In *LREC*, pages 1989–1993. Citeseer.

Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The NomBank project: An interim report. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*.

Minard, A.-L., Speranza, M., Urizar, R., Altuna, B., van Erp, M., Schoen, A., and van Son, C. (2016). MEANTIME, the NewsReader Multilingual Event and Time Corpus. In *Proceedings of LREC 2016*.

Mitamura, T., Yamakawa, Y., Holm, S., Song, Z., Bies, A., Kulick, S., and Strassel, S. (2015). Event nugget annotation: Processes and issues. In *Proceedings of the 3rd Workshop on EVENTS at the NAACL-HLT*, pages 66–76.

O'Gorman, T., Wright-Bettner, K., and Palmer, M. (2016). Richer Event Description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56.

Peng, H., Song, Y., and Roth, D. (2016). Event detection and co-reference with minimal supervision. In *EMNLP*, pages 392–402.

Pradhan, S. S., Loper, E., Dligach, D., and Palmer, M. (2007). SemEval-2007 task 17: English lexical sample, SRL and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 87–92. Association for Computational Linguistics.

Pustejovsky, J. and Verhagen, M. (2009). SemEval-2010 task 13: evaluating events, time expressions, and temporal relations (TempEval-2). In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 112–116. Association for Computational Linguistics.

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98. Association for Computational Linguistics.

RDF2PT: Generating Brazilian Portuguese Texts from RDF Data

Diego Moussallem[1,2], Thiago Castro Ferreira [2], Marcos Zampieri[3], Maria Claudia Cavalcanti[4], Geraldo Xexéo[5], Mariana Neves[6], Axel-Cyrille Ngonga Ngomo[1,7]
[1]University of Leipzig, Germany, [2]Tilburg University, The Netherlands,
[3]University of Wolverhampton, United Kingdom, [4]Military Institute of Engineering - Brazil,
[5]Federal University of Rio de Janeiro - Brazil, [6]German Federal Institute for Risk Assessment - Germany
[7]Paderborn University, Germany
[1]lastname@informatik.uni-leipzig.de

Abstract

The generation of natural language from Resource Description Framework (RDF) data has recently gained significant attention due to the continuous growth of Linked Data. A number of these approaches generate natural language in languages other than English, however, no work has been proposed to generate Brazilian Portuguese texts out of RDF. We address this research gap by presenting RDF2PT, an approach that verbalizes RDF data to Brazilian Portuguese language. We evaluated RDF2PT in an open questionnaire with 44 native speakers divided into experts and non-experts. Our results suggest that RDF2PT is able to generate text which is similar to that generated by humans and can hence be easily understood.

Keywords: natural language generation, verbalization, semantic web

1. Introduction

Natural Language Generation (NLG) is the process of generating coherent natural language text from non-linguistic data (Reiter and Dale, 2000). Despite community agreement on the actual text and speech output of these systems, there is far less consensus on what the input should be (Gatt and Krahmer, 2017). A large number of inputs have been taken for NLG systems, including images (Xu et al., 2015), numeric data (Gkatzia et al., 2014), semantic representations (Theune et al., 2001) and Semantic Web (SW) data (Ngonga Ngomo et al., 2013; Bouayad-Agha et al., 2014).

Presently, the generation of natural language from SW, more precisely from RDF data, has gained substantial attention (Bouayad-Agha et al., 2014; Staykova, 2014). Some challenges have been proposed to investigate the quality of automatically generated texts from RDF (Colin et al., 2016). Moreover, RDF has demonstrated a promising ability to support the creation of NLG benchmarks (Gardent et al., 2017). However, English is the only language which has been widely targeted. Even though there are studies which explore the generation of content in languages other than English, to the best of our knowledge, no work has been proposed to generate texts in Brazilian Portuguese from RDF data.

In this paper, we propose RDF2PT, a rule-based approach to verbalize RDF data to Brazilian Portuguese. While the exciting avenue of using deep learning techniques in NLG approaches (Gatt and Krahmer, 2017) is open to this task and deep learning has already shown promising results for RDF data (Sleimi and Gardent, 2016), the morphological richness of Portuguese led us to develop a rule-based approach. This was to ensure that we could identify the challenges imposed by this language from the SW perspective before applying Machine Learning (ML) algorithms.

RDF2PT is able to generate either a single sentence or a summary of a given resource. In order to validate our approach, we evaluated RDF2PT using experts in Natural Language Processing (NLP) and SW as well as non-experts who are lay users or non-users of SW technologies. Both groups are native speakers of Brazilian Portuguese. The results suggest that RDF2PT generates texts which can be easily understood by humans and also help to identify some of the challenges related to the automatic generation of Brazilian Portuguese (especially from RDF). The version of RDF2PT used in this paper, all experimental results and the texts generated for the experiments are publicly available.[1]

2. Related Work

According to Staykova (2014) and Bouayad-Agha et al. (2014), there has been a plenty of works which investigated the generation of Natural Language (NL) texts from Semantic Web Technologies (SWT) as an input data. However, the subject of research has only recently gained significant momentum. This attention comes from the great number of published works such as (Cimiano et al., 2013; Duma and Klein, 2013; Ell and Harth, 2014; Biran and McKeown, 2015) which used RDF as an input data and achieved promising results. Also, the works published in the WebNLG (Colin et al., 2016) challenge, which used deep learning techniques such as (Sleimi and Gardent, 2016; Mrabet et al., 2016), also contributed to this interest. RDF has also been showing promising benefits to the generation of benchmarks for evaluating NLG systems (Gardent et al., 2017; Perez-Beltrachini et al., 2016; Mohammed et al., 2016; Schwitter et al., 2004; Hewlett et al., 2005; Sun and Mellish, 2006).

Despite the plethora of works written on handling SWT data, only a few have exploited the generation of languages other than English, for instance, Keet and Khumalo (2017) to Zulu language. Additionally, a considerable number of NLG approaches can be found to European or Brazilian

[1] https://github.com/dice-group/RDF2PT

Portuguese languages (Pereira and Paraboni, 2008; Cuevas and Paraboni, 2008; de Novais et al., 2009; de Novais et al., 2010; de Novais et al., 2012; de Novais and Paraboni, 2013; De Oliveira and Sripada, 2014; Pereira et al., 2015), however, none of them have exploited the generation of NL from RDF. Therefore, to the best of our knowledge, RDF2PT is the first proposed approach to this end.

3. The RDF2PT approach

In this section, we first give a brief description of how RDF data is able to represent useful linguistic information and we detail our approach RDF2PT in a sequence.

3.1. Preliminary RDF concepts

Although previous works such as Sun and Mellish (2006) have already introduced how a single RDF statement contains linguistic information, we briefly explain the concept for a better understanding of RDF2PT.

RDF (RDF Working Group, 25 February 2014) statements are based on graph data models for representing knowledge. Thus, an RDF graph is a set of facts. Facts are expressed as so-called triples in the form `(subject predicate object)`. The *subjects* and *predicates* are Internationalized Resource Identifiers (IRI)s and *objects* are either IRIs or literals. Literals, in general, have a datatype that defines its kind of values. For example, a literal can be a date, a number, a measure, a word or a group of words. On the other hand, a predicate denotes a binary relation between the *subject* and *object* as an argument. Additionally, RDF vocabulary comprises some built-in properties. The most common one is `rdf:type`, which states that a resource denoted by the subject is an instance of the class specified by the object of the triple. For example, the Listing 1 shows a fragment of Albert Einstein DBpedia's resource[2] which represents the following information: *"Albert Einstein was a scientist who worked in physics area. He was born in Ulm and died in Princeton."*.

```
:Albert_Einstein rdf:type dbo:Scientist
:Albert_Einstein dbo:field :Physics
:Albert_Einstein dbo:birthPlace :Ulm
:Albert_Einstein dbo:deathPlace :Princeton
```

Listing 1: An excerpt of RDF triples.

3.2. Approach

RDF2PT approach is akin to the approach SPARQL2NL (Ngonga Ngomo et al., 2013) from which the project SemWeb2NL[3] originated. SemWeb2NL comprises rule-based and template-based approaches which aim to verbalize texts and concepts not only from RDF triples but also from ontologies and SPARQL queries into English. In addition, SemWeb2NL is able to produce automatically educational Question Answering (QA) systems for self-assessment (Bühmann et al., 2015). Despite the RDF2PT approach being capable of generating single sentences from distinct RDF triples, for the sake of space, our description focuses on how RDF2PT can output a simplified summary of a given resource.

A generic NLG pipeline is composed by three tasks which are *Document Planing*, *Micro Planning* and *Realization*. RDF2PT operates mostly at the level of the first two and to the *Realization* task, RDF2PT uses an adaption of SimpleNLG to Brazilian Portuguese (De Oliveira and Sripada, 2014).

In the following sections, we describe the RDF2PT steps according to an NLG system pipeline (Gatt and Krahmer, 2017). We then use the Portuguese version of DBpedia as a Knowledge Base (KB) (Auer et al., 2007; Lehmann et al., 2015) and as source for our examples.

3.3. Document Planning

This initial phase is divided into two sub-tasks. First, *Content determination*, which is responsible for deciding what information a certain NLG system should include in the generated text. Second, *Discourse planning* (also known as Text structuring), which determines the order of the information in paragraphs and its rhetorical relation.

Content determination RDF2PT assumes the description of a resource to be the set of RDF statements of which this resource is the subject. Hence, given a resource, RDF2PT first performs a SPARQL query to get its most specific class through the predicate `rdf:type`. Afterward, RDF2PT gets all resources which belong to this specific class and ranks their predicates by using Page Rank (Page et al., 1999) over the KB. By these means, our approach can determine the most popular facts of this specific class.[4]. Once the predicates are ranked, RDF2PT considers only the top seven most popular predicates of the class to describe the input resource.[5] For example, given `dbr:Albert_Einstein` as a resource, RDF2PT determines its most specific class to be `dbo:Scientist`. Then, it ranks all the predicates used by this class per popularity according to its resources (see Listing 2).

```
rdf:type                    dbo:field
dbo:deathPlace              dbo:almaMater
dbo:knownFor                dbo:award
dbo:doctoralStudent
```

Listing 2: Most popular predicates of a scientist.

Discourse planning In this step, RDF2PT clusters and orders the triples. The subjects are ordered with respect to the number of their occurrences, thus assigning them to those input triples that mention them. RDF2PT processes the input in descending order with respect to the frequency of the variables they contain, starting with the projection variables and only after that, turning to other variables. This method has already been used by other approaches and is the most effective method to follow regarding rule-based approaches to RDF (Bouayad-Agha et al., 2014). As an example, consider the following triples in Listing 3.

```
:Albert_Einstein  dbo:deathPlace :Princeton.
:Princeton        dbo:Country    :USA.
```

[2] `http://dbpedia.org/resource/Albert_Einstein`

[3] `https://github.com/AKSW/SemWeb2NL`

[4] The predicates can vary according to the classes and KB

[5] This choice was based on Gardent et al. (2017) which states that seven triples is a reasonable number for describing a resource.

```
:Albert_Einstein  rdf:type       :Scientist.
:Albert_Einstein  dbo:knownFor   :General_relativity.
:Albert_Einstein  dbo:knownFor   :Brownian_motion.
:Albert_Einstein  dbo:birthPlace :Ulm.
:Ulm              rdf:type       :City.
:Ulm              dbo:Country    :Germany.
```

Listing 3: Example of triples before planning

Listing 3 presents three subjects, `:Albert_Einstein`, `:Ulm` and `:Princeton`. As `:Albert_Einstein` is assigned to more triples than the others, it takes the first place in the discourse, followed by `:Ulm`,`:Princeton` respectively (see Listing 4). However, RDF2PT still considers the popularity of predicates from the previous steps and organizes triples based on it, for instance, `rdf:type` comes before others due to its frequency in the KB.

```
:Albert_Einstein  rdf:type       :Scientist
:Albert_Einstein  dbo:birthPlace :Ulm
:Albert_Einstein  dbo:deathPlace :Princeton
:Albert_Einstein  dbo:knownFor   :General_relativity.
:Albert_Einstein  dbo:knownFor   :Brownian_motion.
:Ulm              rdf:type       :City
:Ulm              dbo:Country    :Germany
:Princeton        dbo:Country    :USA
```

Listing 4: Example of triples after planning

3.4. Micro Planning

This step is concerned with the planning of a sentence. It comprises three sub-tasks. Firstly, *Sentence aggregation* decides whether information will be presented individually or separately. Second, *Lexicalization* chooses the right words and phrases in natural language for expressing the semantics about the data. Third, *Coreference generation* (also known as *Referring expression*) is the task responsible for generating syntagms (references) to discourse entities, for example, whether the text should refer to an entity using a definite description, a pronoun or a proper noun (Ferreira et al., 2016). In the following, we describe the challenges behind the tasks entailed.

Sentence aggregation This task is based on Ngonga Ngomo et al. (2013). It is divided into two phases, *subject grouping* and *object grouping*. *Subject grouping* collapses the predicates and objects of two triples if their subjects are the same. *Object grouping* collapses the subjects of two triples if the predicates and objects of the triples are the same.

The common elements are usually subject noun phrases and verb phrases (verbs together with object noun phrases). In order to maximize the grouping effects, we additionally collapse common prefixes and suffixes of triples, irrespective of whether they are full subject noun phrases or complete verb phrases.

In Listing 5, the predicate `dbo:knownFor` shares the same subject `:Albert_Einstein` and also has two objects, `:General_relativity` and `:Brownian_motion`. Additionally, the predicate `dbo:birthPlace` shares the same object `:Ulm` and has two subjects, `:Albert_Einstein` and `:Gabriel_Steiner`. They therefore can be collapsed using a conjunction `AND`, thus organizing and omitting repetitive triples. Moreover, we remove repetitions that arise when triples' verbalizations lead to the same natural language representation.

```
:Albert_Einstein dbo:knownFor :General_relativity.
:Albert_Einstein dbo:knownFor :Brownian_motion.
:Albert_Einstein dbo:birthPlace :Ulm.
:Gabriel_Steiner dbo:birthPlace :Ulm.
```

Listing 5: Grouping subjects and objects

Lexicalization This step comprises the main contribution of RDF2PT for verbalizing the triples in Brazilian Portuguese. In contrast to English, Brazilian Portuguese is a morphologically rich language which contains the grammatical gender of words. Grammatical gender plays a key role because it affects the generation of determiners and pronouns. It also influences the inflection of nouns and verbs. For instance, the passive expression of the verb `nascer` (en: "be born") is `nascida` if the subject is feminine or `nascido` if masculine. Thus, the gender of words is essential for comprehending the semantics of a given Portuguese text. Also, Brazilian Portuguese has different possibilities in the expression of subject possessives. Hence, RDF2PT has to deal with the following phenomena while lexicalizing:

- **Grammatical gender** - In Portuguese, the gender varies between masculine and feminine. This variation leads to supplementary challenges when lexicalizing words automatically. For example, a gender may be represented by articles "um" and "o" (masculine) or "uma" and "a" (feminine). However, the gender also affects the inflection of words. For instance, for the word "cantor" (en: "singer"), if the subject is feminine, the word becomes "cantora". However, there are words which do not inflect, e.g., the word "gerente" (en: "manager"). If the subject is a woman, we only refer to it by using the article "a", i.e., "a gerente". Therefore, there are some challenges to tackle for recognizing the gender and assigning it correctly. A tricky example to solve automatically is "O Rio de Janeiro é uma cidade" (en: Rio de Janeiro is a city). In this case, the subject is masculine but its complement is feminine. Devising handcrafted rules to handle these phenomena can become a hard task. To address this challenge, we use a Part-Of-Speech tagger (TreeTagger in our case) as it retrieves the gender along with the parts of speech.[6] All the obtained genders are attached along with the lexicalizations for supporting the realization step.
- **Classes and resources** - The lexicalizaton of classes and resources is carried by using a SPARQL query to get their Portuguese labels through the `rdfs:label` predicate[7]. In case such a label does not exist, we use either the fragment of their URI (the string after the # character) if it exists, or the string after the last occurrence of "/". Finally, this natural language representation is lexicalized as a noun phrase. Afterwards, RDF2PT recognizes the gender. In case the

[6] see the POS tags `http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/Portuguese-Tagset.html`

[7] Note that it could be any property which returns a natural language representation of the given URI, see (Ell et al., 2011).

resource is recognized as a person, RDF2PT applies a string similarity measure (0.8 threshold) between the lexicalized word with a list of names provided by SemWeb2NL. This list is divided by masculine and feminine which in turn results in the gender. If the resource is not a person, we use Tree-tagger.

- **Properties** - The lexicalization of properties relies on one of the results of Ngonga Ngomo et al. (2013), i.e., that most property labels are either nouns or verbs. To determine which lexicalization to use automatically, we rely on the insight that the first and last words of a property label in Portuguese are commonly the key to determining the type of property. We then use the Tree-Tagger to get the part of speech of predicates. Properties whose label begins with a verb are lexicalized as verbs. For example, the predicate `dbo:knownFor`, which Portuguese label is "conhecido por", has the first word identified as an inflection of the verb "conhecer" (en:know). Therefore, RDF2PT lexicalizes and sets it as a verb. We devised a set of rules to capture this behavior, which we omit due to space restrictions.[8] Moreover, RDF2PT uses some pre-defined templates for improving the quality of lexicalization. For example, the predicate `dbo:birthPlace`, RDF2PT uses the verb "nascer" (eng: be born) along with the predicate "em" (en: "in"), so this predicate can be lexicalized as "nasceu em" (en: was born in).

  For predicates which are recognized as nouns, RDF2PT relies on labels. For instance, `dbo:birthDate` is labeled as "data de nascimento" and recognized as a noun phrase because of its first word "data". RDF2PT also uses the first word of predicates to set the gender. For example, `dbo:deathPlace` is transliterated as "local de falecimento". "local" is masculine. Hence, the determiner to be used in front of this predicate needs to be "o". In contrast to `dbo:birthDate` ("data de nascimento"), the word "data" is feminine, thus the determiner must be "a".

- **Literals** - In an RDF graph, literals usually consist of a *lexical form* `LF` and a *datatype IRI* `DT`. If the datatype is `rdf:langString`, a non-empty *language tag* is specified and the literal is denoted as a *language-tagged string*.[9] Accordingly, the lexicalization of strings with language tags is carried by using simply the lexical form, while omitting the language tag. For example, `''Albert Einstein"@pt` is lexicalized as "Albert Einstein" or `"Alemanha"@pt` ("Germany"@en) is lexicalized as "Alemanha". For other types of literals, we differentiate between built-in[10] and user-defined datatypes. For built-in literals, we use the lexical form, e.g., `"123"^^xsd:int` ⇒ "123". User-defined types are processed by using the literal value together with the (pluralized) natural language representation of the datatype IRI. Thus, we lexicalize `"123"^^dt:squareKilometre` as "123 quilômetros quadrados" (en: "123 square kilometres").

Coreference generation In this step, RDF2PT relies on the number of subjects contained by the RDF statements and only uses other expressions to refer to a given subject in case there is more than one mention of it. RDF2PT replaces the subject by possessive or personal pronouns with the corresponding gender depending on the predicates. For instance, given a triple `dbr: Albert_Einstein dbo:birthPlace dbr:Ulm`, the predicate is a noun phrase then the subject is replaced by a possessive form which is "seu" (en:"his"). However, Brazilian Portuguese has two different ways to express possession and this variation exists due to the necessity of handling complex syntaxes in some sentences and also because the gender of pronouns agrees with objects instead of subjects. For example, "A professora proibiu que o aluno utilizasse `seu` dicionário." (eng: "The teacher forbade the student to use `his/her` dictionary"). The possessive pronoun `seu` in this sentence does not indicate explicitly to whom the dictionary belongs, if it belongs to the `professora` (eng:teacher) or `aluno` (eng:student). Thus, we have explicitly to define the possessive pronoun in order to decrease the ambiguity in texts and it is obviously important when generating text from data. If this sentence was translated into English, we would have indicated to whom the dictionary belonged, `her` or `his`. To this end, we handle the ambiguity of possessive pronouns by interspersing the alternative forms, e.g., `dele` (eng:his) or `dela` (eng: her)" which agrees with the subject. However, it is used just in case more than one subject exists in the same description.
In case the predicate is recognized as a verb (e.g, `dbr: Albert_Einstein dbo:knownFor dbr:General_relativity`), the subject is replaced by its respective personal pronoun `ele` (eng: "he"). While setting the pronouns, RDF2PT recognizes the gender's subject. The `dbo:knownFor` is a verb phrase, thus the subject is replaced by the personal pronoun ":ele"(see Table 1).

Triples before co-reference
1 - (Albert Einstein, ser, cientista)
2 - (Albert Einstein, local de nascimento, Ulm)
3 - (Albert Einstein, ser conhecido por, teoria da relatividade.)
Triples after co-reference
1 - (Albert Einstein, ser, cientista)
2 - (**seu**, local de nascimento, Ulm.)
3 - (**ele**, ser conhecido por, teoria da relatividade).

Table 1: Example of triples in the coreference generation task.

3.5. Linguistic realisation

This last step is responsible for mapping the obtained descriptions of sentences from the aforementioned tasks and

[8] All rules can be found in our code.

[9] In RDF 1.0 literals have been divided into "plain' literals with no type and optional language tags, and typed literals.

[10] List of data types:`http://tinyurl.com/y95mxyxa`

verbalizing them syntactically, morphologically and orthographically into a correct natural language text. To this end, we perform this step by relying on a Brazilian adaptation of SimpleNLG (De Oliveira and Sripada, 2014).[11]

The realization of a triple `(s p o)` depends mostly on the lexicalization of its predicate `p`. If `p` can be realized as a noun phrase, then a possessive clause can be used to express the semantics of `(s p o)`. For example, if (`p`) is a relational noun like `death place` e.g. in the triple `(:Albert_Einstein :deathPlace :Princeton)`, then the verbalization is `o local de falecimento de Albert Einstein foi Princeton.` (eng: The death place of Albert Einstein was Princeton) which formally can be expressed as in equation 1. In case there is a previous triple which shares the same subject, it would be `seu local de falecimento foi Princeton` (eng: his death place was Princeton).

$$\rho(s,p,o) \Rightarrow \texttt{poss}(\rho(p),\rho(s)) \wedge \texttt{subj}(\texttt{BE},\rho(p)) \wedge \texttt{dobj}(\texttt{BE},\rho(o)) \quad (1)$$

In case `p`'s lexicalization is a verb, then the triple is verbalized setting the predicate as a verb. For example, in `(:Albert_Einstein :influenced :Nathan_Rosen)` ρ(`p`) is the verb `influenciar` (en: `influence`), thus, the verbalization is `Albert Einstein influênciou Nathan Rosen` (eng: Albert Einstein influenced Nathan Rosen) and may be formalized as in equation 2.

$$\rho(s,p,o) \Rightarrow \texttt{subj}(\texttt{BE},\rho(p)) \wedge \texttt{dobj}(\texttt{BE},\rho(o)) \quad (2)$$

RDF2PT is able to merge sentences that were derived from the same cluster for generating a readable summary, thus resulting in coordinate sentences. For example, for the triples `(:Albert_Einstein :birthPlace :Ulm)` and `(:Albert_Einstein :deathPlace :Princeton)`, if p_1 and p_2 can be verbalized as nouns, then we apply the following rule:

$$\rho(s,p_1,o_1) \wedge \rho(s,p_2,o_2) \Rightarrow \texttt{conj}(\texttt{poss}(\rho(p_1),\rho(s)) \wedge \texttt{subj}(\texttt{BE}_1,\rho(p_1)) \wedge \texttt{dobj}(\texttt{BE}_1,\rho(o_1)) \wedge \texttt{poss}(\rho(p_2),\rho(\texttt{pronoun}(s))) \wedge \texttt{subj}(\texttt{BE}_2,\rho(p_2)) \wedge \texttt{dobj}(\texttt{BE}_2,\rho(o_2)) \quad (3)$$

Note that `pronoun(s)` returns the correct pronoun for a resource based on its type and gender (see subsection 3.4.). Therewith, we can generate `O local de nascimento de Albert Einstein Ulm e seu local de falecimento é Princeton.` (eng: Albert Einstein's birthplace is Ulm and his death place is Princeton). In addition, in case the KB provides the ending date of a given resource through some predicate, for example dbo:deathDate, RDF2PT is able to lexicalize all the verbs in the past tense.

[11] See the complete list of dependency parsing tags in Ngonga Ngomo et al. (2013).

4. Evaluation

We based our evaluation methodology on Gardent et al. (2017) and Ferreira et al. (2016). Our main goal was to evaluate how well RDF2PT represents the information obtained from the data. We hence divided our evaluation set into expert and non-expert users. Both sets were made up of native speakers of Brazilian Portuguese. We selected six DBpedia categories like Gardent et al. (2017) for selecting the topic of texts. The categories were Astronaut, Scientist, Building, WrittenWork, City, and University. We detail below how we carried out both evaluation sets.

Experts - We aimed to evaluate the adequacy and fluency of the generated texts from the perspective of experts. All experts hold at least a master degree in the fields NLP or SW. In the questionnaire, we used the same two questions as Gardent et al. (2017): (1) Adequacy: Does the text contain only and all the information from the data? (2) Fluency: Does the text sound fluent and natural? We asked the 10 experts to evaluate 12 texts distributed across the aforementioned DBpedia categories, with two pieces of text from each category. All texts were generated automatically by the RDF2PT approach. The answers were on a scale from 1 to 5.[12]

Non-experts - We evaluated the clarity and fluency of the generated texts. To this end, we created three types of texts. First, the texts were generated using a baseline of RDF2PT approach, which removes the functional words and also does not apply coreference rules. This version served as baseline as there is no other work pertaining to generating Brazilian Portuguese from RDF. Second, we used the texts generated using the RDF2PT approach outline at section 3. The third type of texts were created manually by three different human annotators. Table 2 depicts an example of text in the three versions.

In total, we created three versions of 18 texts (one text per resource) selected randomly from the aforementioned DBpedia categories (total: 54 texts). These texts were distributed over three lists, such that each list contained one variant of each text, and there was an equal number of texts from the three types (Baseline, RDF2PT, Human). The experiment was run on CrowdFlower and is publicly available.[13]

The experiment was performed by 30 participants (10 per list). They were asked to rate each text considering the clarity and fluency based on two questions from Ferreira et al. (2016) on a scale from 1 (Very Bad) to 5 (Very Good). The questions were: (1) Fluency: Does the text present a consistent, logical flow? (2) Clarity: Is the text easy to understand?

4.1. Results

Experts Figure 1 displays the average fluency and clarity of the texts. The results suggest that RDF2PT is able to capture and represent the information from data adequately. Also, the generated texts are fluent enough to be understood by humans.

[12] Questionnaire: `http://tinyurl.com/y9vegl4g`

[13] `https://ilk.uvt.nl/~tcastrof/semPT/evaluation/`

Version	Text
Baseline	Albert Einstein é cientista, Albert Einstein campo é física, Albert Einstein lugar falecimento Princeton. Albert Einstein ex-instituição é Universidade Zurique, Albert Einstein é conhecido Equivalłncia massa-energia, Albert Einstein prêmio é Medalha Max Planck, Albert Einstein estudante doutorado é Ernst Gabor Straus.
Modelo	Albert Einstein foi um cientista, o campo dele foi a física e ele faleceu no Princeton. Além disso, sua ex-instituição foi a Universidade de Zurique, ele é conhecido pela Equivalência massa-energia, o prêmio dele foi a Medalha Max Planck e o estudante de doutorado dele foi o Ernst Gabor Straus.
Humano	Albert Einstein era um cientista, que trabalhava na área de Física. Era conhecido pela fórmula de equivalência entre massa e energia. Formou-se na Universidade de Zurique. Einstein ganhou a medalha Max Planck por seu trabalho. Em Princeton, onde morreu, teve sob sua orientação Ernst Gabor Straus.

Table 2: Example of text in the Baseline, RDF2PT approach and Human version.

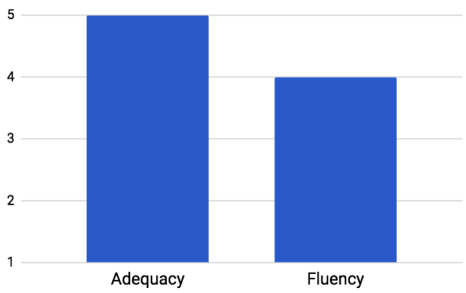


Figure 1: RDF2PT results in experts survey

Non-experts Figure 2 depicts the average fluency and clarity of the texts where their topics are described by *Baseline*, RDF2PT and *Human* approaches respectively. Inspection of this figure clearly shows that *Baseline* texts are rated lower than both the RDF2PT and *Human* texts, in fact, RDF2PT is superior to *Baseline* and close to *Human*.

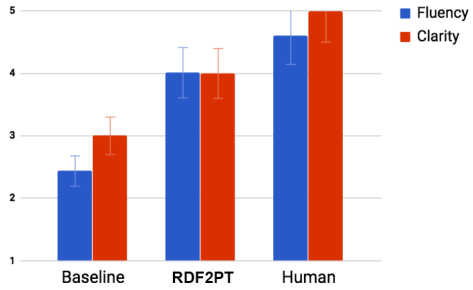


Figure 2: Results in non-experts experiment

We performed a statistical analysis in order to measure the significance of the difference between the types (Baseline, RDF2PT, Human). First, we carried out a Friedman test (Friedman, 1937) which resulted in a significant difference in the fluency (x^2 = 193.61, ρ <0.0001) and clarity (x^2 = 180.9, ρ <0.0001) for the three kinds of texts. Afterward, we conducted a post-hoc analysis with Wilcoxon signed-rank test corrected for multiple comparisons using the Bonferroni method, resulting in a significance level set at ρ <0.017. Texts of the Baseline are hence significantly less statistically understandable (Z=525 and ρ <0.017.) and fluent (Z=275.5 and ρ <0.017.) than those generated by the RDF2PT approach. However, RDF2PT also generates texts less comprehensible (Z=1617.5 and ρ <0.017.) and fluent (Z=1640.0 and ρ <0.017.) than those generated by humans. Clearly, humans were superior to Baseline in terms of comprehensibility (Z=234.5 and ρ <0.017.) and fluency (Z=264.0.0 and ρ <0.017.), as we expected. Therefore, there is a significant difference among all models, being baseline < model < human.

4.2. Discussion

During the development of RDF2PT, some challenges to our rule-based algorithm became clear. The first challenge was to identify when the object of a predicate is an adjective. Consider the following triple, `(:Albert_Einstein dbo:nationality :Áustria)`, its object `Áustria` is a demonym and should be lexicalized as an adjective. However, it is lexicalized as a noun because the part-of-speech recognized by RDF2PT considers only the label `Austria`, which is a noun, and does not consider the predicate nationality, which is an important part, thus decreasing the quality of the generated texts. Second, RDF2PT's algorithm is totally dependent on ontology terms, thus when a given ontology contains wrong labels, RDF2PT is not able to recognize by itself the error and lexicalizes the terms wrongly. Third, the gender continues to be a hard task and RDF2PT sometimes presents poor results. For example, "`Os Lusíadas é uns obra literária`", the determiner `uns` should be feminine and singular, because `obra` is singular and has a feminine gender. However, it is accorded to the subject `Os Lusíadas`. This example is similar to the example presented in section 3.4. We hence envision the use of ML algorithms for improving the gender recognition and generation. The last challenge observed was the generation of coordinated sentences by RDF2PT which helped the users in our experimental setup recognize if a given text was generated by RDF2PT or humans. This behavior is because humans are likely to write subordinate sentences. For example, while RDF2PT is able to generate `Albert Einstein foi um cientista e ele nasceu em Ulm.` (eng: Albert Einstein was a scientist and he was born in Ulm), a human would write this same sentence in the following way, `Albert Einstein foi um cientista que/cujo nasceu em Ulm` (eng: Albert Einstein was a scientist who was born in Ulm). This difference was crucial in the perspective of our evaluators. Therefore, the generation of subordinate sentences in Portuguese must be investigated in the near

future.

5. Further Application Scenarios

We envision two promising applications using RDF2PT. The first aims to support the automatic creation of benchmarking datasets to Named Entity Recognition (NER) and Entity Linking (EL) tasks. In Brazilian Portuguese, there is a lack of gold standards datasets for these tasks, which makes the investigation of these problems difficult for the scientific community. Our aim is to create Brazilian Portuguese silver standard datasets which are able to be uploaded into GERBIL(Usbeck et al., 2015) for an easy evaluation. To this end, we aim to implement RDF2PT in BENGAL (Ngomo et al., 2017), which is an approach for automatically generating NER benchmarks based on RDF triples and Knowledge Graphs. This application has already resulted in promising datasets which we have used to investigate the capability of multilingual entity linking systems[14] for recognizing and disambiguating entities in Brazilian Portuguese texts. The second appealing application of RDF2PT is the generation of automatic QA systems based on RDF for self-assessment. Therefore, the aim is to develop a Portuguese version of ASSES (Bühmann et al., 2015), which is a self-assessment platform for students based on DBpedia.

6. Conclusion

We presented the RDF2PT approach which verbalizes RDF data to Brazilian Portuguese texts. The results demonstrated that RDF2PT generates texts with a good quality of fluency and clarity compared to human texts. In addition, we identified important challenges for generating Brazilian Portuguese texts from RDF using a rule-based approach. We intend to exploit the application of ML models along with other Brazilian Portuguese resources[15] to produce more fluent results and also to investigate the usage of ML classification algorithms to improve the choice of grammatical gender of words.

Moreover, we aim to create a multilingual version of RDF2PT which will consist of French, German, Italian and Spanish. To this end, we will exploit the similarity among their syntaxes in the micro-planning task and we will reuse their respective SimpleNLG versions (Mazzei et al., 2016; Bollmann, 2011; Vaudry and Lapalme, 2013; Ramos-Soto et al., 2017) for the realization task.

Acknowledgments

This work has been supported by the H2020 project HOBBIT (GA no. 688227) and supported by the Brazilian National Council for Scientific and Technological Development (CNPq) (no. 206971/2014-1 and no. 203065/2014-0.). This research has also been supported by the German Federal Ministry of Transport and Digital Infrastructure (BMVI) in the projects LIMBO (no. 19F2029I), OPAL (no. 19F2028A) and GEISER (no. 01MD16014E) as well as by the BMBF project SOLIDE (no. 13N14456).

[14] `http://gerbil.aksw.org/gerbil/experiment?id=201801050040` and `http://gerbil.aksw.org/gerbil/experiment?id=201801110012`

[15] Resources: http://www.nilc.icmc.usp.br/nilc/index.php/tools-and-resources

Bibliographical References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The semantic web*, pages 722–735.

Biran, O. and McKeown, K. (2015). Discourse planning with an n-gram model of relations. In *EMNLP*, pages 1973–1977.

Bollmann, M. (2011). Adapting simplenlg to german. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 133–138. Association for Computational Linguistics.

Bouayad-Agha, N., Casamayor, G., and Wanner, L. (2014). Natural language generation in the context of the semantic web. *Semantic Web*, 5(6):493–513.

Bühmann, L., Usbeck, R., and Ngomo, A.-C. N. (2015). Assessautomatic self-assessment using linked data. In *International Semantic Web Conference*, pages 76–89. Springer, Cham.

Cimiano, P., Lüker, J., Nagel, D., and Unger, C. (2013). Exploiting ontology lexica for generating natural language texts from rdf data. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 10–19, Sofia, Bulgaria, August. Association for Computational Linguistics.

Colin, E., Gardent, C., Mrabet, Y., Narayan, S., and Perez-Beltrachini, L. (2016). The webnlg challenge: Generating text from dbpedia data. In *Proceedings of the 9th International Natural Language Generation conference*, pages 163–167.

Cuevas, R. R. M. and Paraboni, I. (2008). A machine learning approach to portuguese pronoun resolution. In *Ibero-American Conference on Artificial Intelligence*, pages 262–271. Springer, Berlin, Heidelberg.

de Novais, E. M. and Paraboni, I. (2013). Portuguese text generation using factored language models. *Journal of the Brazilian Computer Society*, 19(2):135–146.

de Novais, E. M., de Oliveira, R. L., Pereira, D. B., Tadeu, T. D., and Paraboni, I. (2009). A testbed for portuguese natural language generation. In *Information and Human Language Technology (STIL), 2009 Seventh Brazilian Symposium in*, pages 154–157. IEEE.

de Novais, E. M., Tadeu, T. D., and Paraboni, I. (2010). Text generation for brazilian portuguese: the surface realization task. In *Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas*, pages 125–131. Association for Computational Linguistics.

de Novais, E. M., Paraboni, I., and da Silva Junior, D. F. (2012). Portuguese text generation from large corpora. In *LREC*.

De Oliveira, R. and Sripada, S. (2014). Adapting simplenlg for brazilian portuguese realisation. In *INLG*, pages 93–94.

Duma, D. and Klein, E. (2013). Generating natural language from linked data: Unsupervised template extraction. In *IWCS*, pages 83–94.

Ell, B. and Harth, A. (2014). A language-independent method for the extraction of rdf verbalization templates. In *INLG*, pages 26–34.

Ell, B., Vrandecic, D., and Simperl, E. P. B. (2011). Labels in the web of data. In *Proceedings of ISWC*, volume 7031, pages 162–176. Springer.

Ferreira, T. C., Krahmer, E., and Wubben, S. (2016). Towards more variation in text generation: Developing and evaluating variation models for choice of referential form. In *ACL (1)*.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, 32(200):675–701.

Gardent, C., Shimorina, A., Narayan, S., and Perez-Beltrachini, L. (2017). Creating training corpora for nlg micro-planning. In *Proceedings of ACL*.

Gatt, A. and Krahmer, E. (2017). Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *arXiv preprint arXiv:1703.09902*.

Gkatzia, D., Hastie, H. F., and Lemon, O. (2014). Comparing multi-label classification with reinforcement learning for summarisation of time-series data. In *ACL (1)*, pages 1231–1240.

Hewlett, D., Kalyanpur, A., Kolovski, V., and Halaschek-Wiener, C. (2005). Effective nl paraphrasing of ontologies on the semantic web. In *Workshop on end-user semantic web interaction, 4th int. semantic web conference, galway, ireland*.

Keet, C. M. and Khumalo, L. (2017). Toward a knowledge-to-text controlled natural language of isizulu. *Language Resources and Evaluation*, 51(1):131–157.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Mazzei, A., Battaglino, C., and Bosco, C. (2016). Simplenlg-it: adapting simplenlg to italian. In *INLG*, pages 184–192.

Mohammed, R., Perez-Beltrachini, L., and Gardent, C. (2016). Category-driven content selection. In *Proceedings of the 9th International Natural Language Generation conference*, pages 94–98.

Mrabet, Y., Vougiouklis, P., Kilicoglu, H., Gardent, C., Demner-Fushman, D., Hare, J., and Simperl, E. (2016). Aligning texts and knowledge bases with semantic sentence simplification. *WebNLG 2016*.

Ngomo, A. N., Röder, M., Moussallem, D., Usbeck, R., and Speck, R. (2017). Automatic generation of benchmarks for entity recognition and linking. *CoRR*, abs/1710.08691.

Ngonga Ngomo, A.-C., Bühmann, L., Unger, C., Lehmann, J., and Gerber, D. (2013). Sorry, i don't speak sparql: translating sparql queries into natural language. In *Proceedings of the 22nd international conference on World Wide Web*, pages 977–988. ACM.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Pereira, D. and Paraboni, I. (2008). Statistical surface realisation of portuguese referring expressions. *Advances in Natural Language Processing*, pages 383–392.

Pereira, J. C., Teixeira, A., and Pinto, J. S. (2015). Towards a hybrid nlg system for data2text in portuguese. In *Information Systems and Technologies (CISTI), 2015 10th Iberian Conference on*, pages 1–6. IEEE.

Perez-Beltrachini, L., Sayed, R., and Gardent, C. (2016). Building rdf content for data-to-text generation. In *COLING*, pages 1493–1502.

Ramos-Soto, A., Janeiro-Gallardo, J., and Bugarín, A. (2017). Adapting SimpleNLG to spanish. In *10th International Conference on Natural Language Generation*.

RDF Working Group, W. (25 February 2014). *RDF 1.1 Concepts and Abstract Syntax*. W3C Recommendation. Available at `https://www.w3.org/TR/rdf11-concepts/`.

Reiter, E. and Dale, R. (2000). *Building natural language generation systems*. Cambridge university press.

Schwitter, R., Tilbrook, M., et al. (2004). Controlled natural language meets the semantic web. In *Proceedings of the Australasian Language Technology Workshop*, volume 2, pages 55–62.

Sleimi, A. and Gardent, C. (2016). Generating paraphrases from dbpedia using deep learning. *WebNLG 2016*, page 54.

Staykova, K. (2014). Natural language generation and semantic technologies. *Cybernetics and Information Technologies*, 14(2):3–23.

Sun, X. and Mellish, C. (2006). Domain independent sentence generation from rdf representations for the semantic web. In *Combined Workshop on Language-Enabled Educational Technology and Development and Evaluation of Robust Spoken Dialogue Systems, European Conference on AI, Riva del Garda, Italy*.

Theune, M., Klabbers, E., de Pijper, J.-R., Krahmer, E., and Odijk, J. (2001). From data to speech: a general approach. *Natural Language Engineering*, 7(1):47–86.

Usbeck, R., Röder, M., Ngonga Ngomo, A.-C., Baron, C., Both, A., Brümmer, M., Ceccarelli, D., Cornolti, M., Cherix, D., Eickmann, B., Ferragina, P., Lemke, C., Moro, A., Navigli, R., Piccinno, F., Rizzo, G., Sack, H., Speck, R., Troncy, R., Waitelonis, J., and Wesemann, L. (2015). GERBIL – general entity annotation benchmark framework. In *24th WWW conference*.

Vaudry, P.-L. and Lapalme, G. (2013). Adapting simplenlg for bilingual english-french realisation. In *ENLG*, pages 183–187.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057.

Towards a music-language mapping

Michele Berlingerio, Francesca Bonin
IBM Research Ireland
{mberling, fbonin}[at] ie.ibm.com

Abstract

We explore a novel research idea, that we call Musical Language Processing (MLP), which investigates the possibility of a musical input to speech interaction systems. We present the first attempts at finding a mapping between musical pieces and dialogues, based on the frequency of musical patterns. Our findings on one possible alignment between classical piano compositions and dialogues from popular TV series are encouraging, and open the way to further investigations along this line of research.

Keywords: Language generation, lexical analysis, data-driven methods

1. Introduction

Natural Language Processing applications are becoming more and more pervasive in our every day lives. Dialogue systems like Amazon Echo, Google Home, Microsoft Cortana, or Apple Siri, are assisting us in many tasks like searches, purchases, simple calculations, etc. As of today, all these technologies are language-dependent, must be re-trained for each different language and they imply the knowledge of the input language for any form of communication. In this work, we investigate the possibility of musical input to dialogue systems. While we appreciate that musical input certainly requires the knowledge of the basis of musical theory for allowing communication, it presents several advantages: *i)* it is universal among different languages; *ii)* it would facilitate the communication with dialogue system for persons with linguistics disabilities, that have been proven to have particular musical skills (Heaton et al., 1998; Mottron et al., 2006; Happé, 1999); *iii)* it would be simple to learn for a large class of users, i.e. musicians. We call this line of research Musical Language Processing (MLP).

The fist step is to find a mapping between musical pieces and dialogues, in order to investigate the possibility of an alignment. To this aim, in this preliminary work, we study one possible mapping between classical piano compositions and dialogues from popular TV series, by investigating the frequency distributions of words in dialogues and chords in music. From this analysis we aim at creating a lexical mapping between chords and the English vocabulary, which would allow, at a later step, to investigate possible syntactical mappings (n-chords and n-grams), and semantic mappings, where the musical language will be able to express simple meaningful sentences.

Note that, instead of trying to map a chord to a word, and then forcing users to learn the mapping and play the right sequence of chords to form a sentence, our final goal is to "reverse-engineer" already existing music, and find a mapping that any person with "some" knowledge of music would be able to reproduce, without having to learn a new language, or a new kind of music. This is a what-if analysis of what would happen if, for example, we would map existing Bach's compositions, chord by chord: would they form a meaningful sentence?

While there are works involving music and Natural Language Processing, to the best of our knowledge this is the first attempt at mapping a musical language to a spoken one.

The paper is structured as follows: in Section 2. we present previous works; in Section 3. we describe the methodology; Section 4. describes the datasets used; preliminary experiments are presented in Section 5. and conclusions are drawn in Section 6.

2. Related work

Previous works have been focussing on the correlation between music and language. (Longfellow, 1835) defines music as the universal language of humanity, showing how music has always been considered a means of communication, sharing many characteristics with languages.

Approaches to apply NLP techniques to music have been conducted by (Bod and others, 2001). The authors apply syntactic parsing to musical compositions noticing how ambiguities is a common problem, hence an interesting similarity between music and language.

From a prosodic point of view, (Patel and Daniele, 2003) showed the relations between rhythm in language and music.

Some works have been conducted on the relation between music and emotions, in sentiment analysis, (Mihalcea and Strapparava, 2012; Strapparava et al., 2012), exploiting the music and the lyrics of songs.

(Davis and Mohammad, 2014) created a system to generate music from text, using a mechanism to determine sequences of notes that capture the emotional activity in text.

In light of the similarities emerged between music and language in previous works, in this paper we investigate the possibility to map musical chords and words based on their frequency. Differently from (Davis and Mohammad, 2014), we do not create music from text, but we explore the first steps needed to produce the best possible mapping between English lexicon and chords of six classical composers.

3. Methodology

The purpose of this paper is to show a what-if analysis of what would happen if we were to try to translate music into spoken language, in such a way that it should be possible to control an ad-hoc smart personal assistant.

A simple interpretation of this would be to train a musician to play "words" instead of notes, or chords. Our approach is the opposite, and takes advantage of the large availability of music on the Internet: we start from existing music, and reverse-engineer a possible music-to-language mapping, by means of a data-driven approach.

We looked at a large set of musical pieces, trying to map notes, or chords, to words, and find a *good* mapping, so that any musician would have to invest little effort in adapting to the mapping, starting from the music (s)he already knows. However, defining a good mapping is not an easy task. We should try to align music and language not only at the lexicon level, but also the syntactical and the semantical one. In this paper, we make the first attempt in finding such a mapping directly from available music, starting from the lexical level. In future work, we will refine our methodology and focus also on other linguistics levels.

To focus on the lexicon, one of the first steps is exploring possible manners to compose a word out of music. There are several options: we could map a note to a phoneme, a note to a word, a sequence of notes to a word, etc. The first solution presents a problem: in music, several notes can be played at the same time (which is the definition of a *chord*), while, in linguistics, phonemes are uttered in sequence. The second solution, mapping notes to words, besides suffering from the problem above, would also dramatically limit our vocabulary: a piano keyboard has 88 keys, while there are hundreds of thousands words in modern English.

To solve this issue, one could consider also the chords, to broaden the space. If we allow for any combination of the 88 keys of a piano keyboard to correspond to a word, this would give us $2^{88} = 2.09 * 10^{26}$ possibilities, which is more than enough for any given language in the world. However, not all these combinations are used by composers, as well as not all the words of a vocabulary are used in a given conversation. Actually, in both music and language, we can compute the frequency of appearance of a particular combination of notes or words. From there, we can also compute n-grams (i.e. sequences of chords), chords co-occurence, and evolve into a syntactical and semantic mapping.

Moreover, we could add punctuation and map musical *phrases* to linguistic sentences: John White defines a musical phrase as "the smallest musical unit that conveys a more or less complete musical thought. Phrases vary in length and are terminated at a point of full or partial repose, which is called a cadence." (White, 1976).

In Section 5. we show the results of applying this kind of mapping, hereafter *chord-word* mapping, to six music datasets presented in Section 4.. Our first step is to compare high level statistics of the musical datasets with 6 textual dataset (described in 4.): number of different chord-words used by a composer (or, in a musical piece) vs number of different words used in a given TV series, frequency distribution of the resulting chord-words (unigrams) as well as bi and trigrams, and chord-words per minute vs words per minute in a TV series. The aim is to assess if even basic high-level properties of the spoken language, such as the Zipf's law of the word frequencies, are properties of our mapping as well.

At a high level, given a music file, our procedure is as follows:

1. we scan the music score and keep track of the notes being played at the same time: every time these change, we record a "chord" (even if this is composed by less than three notes, for simplicity)
2. we assign an id to each resulting chord
3. we repeat the procedure for each piece from the same composer, forming the composer's vocabulary
4. we sort the vocabulary by descending frequency by composer, and map it to different words according to the descending frequency in different samples of spoken language.

Data	# pieces	Avg # of notes per piece
Bach	153	2747.82
Beethoven	17	8311.53
Chopin	49	4388.08
Grieg	17	2812.88
Schubert	30	9482.50
Schumann	25	2955.64

Table 1: Basic statistics for the music datasets

Data	# episodes	Avg # of words per ep.
BB	58	3033.71
GOT	51	3786.96
HIMYM	188	2635.03
HOC	41	5037.59
MF	170	3248.56
S	77	5902.74

Table 2: Basic statistics for the TV series datasets

4. Datasets

We used two kinds of data: MIDI[1] music files of classical piano compositions, and TV series subtitles for dialogues. In particular, we used compositions from: Bach[2], Beethoven, Chopin, Grieg, Schubert, and Schumann[3]. As TV series, we used the subtitles from all available episodes of six popular shows: Breaking Bad (BB), Game of Thrones (GOT), How I Met Your Mother (HIMYM), House of Cards (HOC), Modern Family (MF), and Suits (S)[4]. The aim of this broad choice was to try to differentiate the styles, both of the musical pieces and of the dialogues, by varying composers (from different eras), and topics and setting for the TV series. We acknowledge that, in future work, an even broader and more elaborate choice should be made to reduce potential bias. Moreover, standard datasets should be added for spoken dialogues, to be able to assess the results against well known properties of those datasets.

Tables 3 and 4 show some basic statistics of our datasets: for each musical composer, we report the number of pieces

[1] https://www.midi.org/specifications

[2] http://www.bachcentral.com/midiindexcomplete.html

[3] http://www.piano-midi.de

[4] http://www.tvsubtitles.net

Data	# distinct chord-words	Avg distinct c.-w./piece
Bach	16727	109.33
Beethoven	15218	895.18
Chopin	18388	375.27
Grieg	3708	218.12
Schubert	19945	664.83
Schumann	8160	326.40

Table 3: Distinct chord-words - music datasets

Data	# distinct words	Avg distinct w./ep.
BB	10325	178.02
GOT	8381	164.33
HIMYM	21461	114.15
HOC	10802	263.46
MF	20776	122.21
S	15257	198.14

Table 4: Distinct words - TV series datasets

in our collection, and the average number of notes per piece; for each TV series, we report the number of episodes in our collection and the average number of words per episode.

5. Experiments

We implemented our chord-word mapping in Python, using the mido library for handling MIDI files[5].

Tables 3 and 4 show the number of distinct chord-words and words in different composers or TV series, respectively, as well as the average number of them by piece or episode. Note that we did not apply any kind of normalization, while we are aware that episodes from different series have different lenghts, and this is true also within the pieces of a given composer. However, we think of a single piece or episode as a story *per se*, therefore we are interested in studying the language used to tell *one* story. Different normalizations are possible here (for example, number of words or chord-words per minute), and they will be studied in future work.

Figures 1 and 2 show the distributions of frequency of the n-grams (n from 1 to 3) in three different TV series and three different composers, respectively.

We see that both the distributions of the n-grams in the analysed compositions and TV series follow a Zipf's Law (Powers, 1998). Moreover, we notice that the change of α exponent between unigrams, bigrams and trigrams in the composers is similar to the one in the TV series. This interesting parallelism poses the bases for a possible lexical mapping based on chords frequency.

Note that we had datasets of different lengths for different composers, therefore the data for Grieg and Schumann appear much sparser. However, being Schumann a composer from practically the same musical era of all the other ones, his distribution does not differ from the others. Grieg, instead, was from a different era, the romanticism, and his language is indeed different. This may suggest that composers of even more recent eras, like the jazz one, may present much different languages. Jazz, for example, is well known to follow non-repetitive, improvised, schemas, by continuously introducing and relieving musical *tension*. This is obtained by playing notes and chords that are far from the current harmony, or by delaying or anticipating notes. It would be interesting to compare jazz to particular types of dialogues, such as quarrels, or jokes.

[5] https://github.com/olemb/mido

Rank	Unigram	Bigram	Trigram
1	i	you know	i don't know
2	you	i don't	you want to
3	the	in the	oh my god
4	to	this is	what are you
5	a	and i	what do you

Table 5: Most frequent n-grams in HIMYM

Rank	Unigram	Bigram	Trigram
1	rest	A#4 rest	G5-A#5-C#6 F#5-A5-A#5-D6 F#5-A5-D6
2	A#4	G4 rest	D#4-B4 D4-D#4-G#4-A#4-B4 D4-G#4-A#4
3	C6	C5 rest	D#4-G#4-B4 D#4-B4 D4-D#4-G#4-A#4-B4
4	F6	F5 rest	G5-A5-A#5-C#6 G5-A#5-C#6 F#5-A5-A#5-D6
5	D#6	D#5 rest	D4-D#4-G#4-A#4-B4 D#4-G#4-B4 D#4-B4

Table 6: Most frequent n-grams in Beethoven

Tables 5 and 6 show the five most frequent unigrams, bigrams and trigrams, for HIMYM and Beethoven, respectively (due to lack of space, we are not reporting this information for the others). As our aim was to map the entire language, we did not perform any filtering of stop words, nor applied lemmatization. It is not a surprise, then, that the top words in the TV series are usually considered stop words. It is interesting to note that their musical counterpart could be considered *musical stop words* as well: the most frequent chords are a pause, and then single notes in the central area of the human voice. Note that, due to the way we processed MIDI files and subtitles, we do have pauses in the music datasets, but we did not process the silences in the TV series. In the latter, in fact, they are also affected by non-dialogue scenes, while pauses in music are usually brief. Nevertheless, this result in music confirms composer Stravinski's view of the musical language, when he said that music is composed mainly of silence (Cantoni, 2014).

6. Conclusions and future work

We have reported the preliminary investigation of a music-language mapping, to pose the basis for a new language to be used in speech interaction systems. We have shown interesting parallelisms between classical music and the lexicon of dialogues in TV series. We intend to proceed with the analysis between music and the syntactic level of the spoken language as well as investigating different musical and linguistic genres. Finally we will be looking at the semantic level, to understand how to ensure that a so constructed musical piece would *make sense* in the common language.

7. Bibliographical References

Bod, R. et al. (2001). Probabilistic grammars for music. In *Belgian-Dutch Conference on Artificial Intelligence (BNAIC)*. Citeseer.

Cantoni, A. (2014). *The Language of Stravinsky*. Musikwissenschaftliche Publikationen. Olms, Georg.

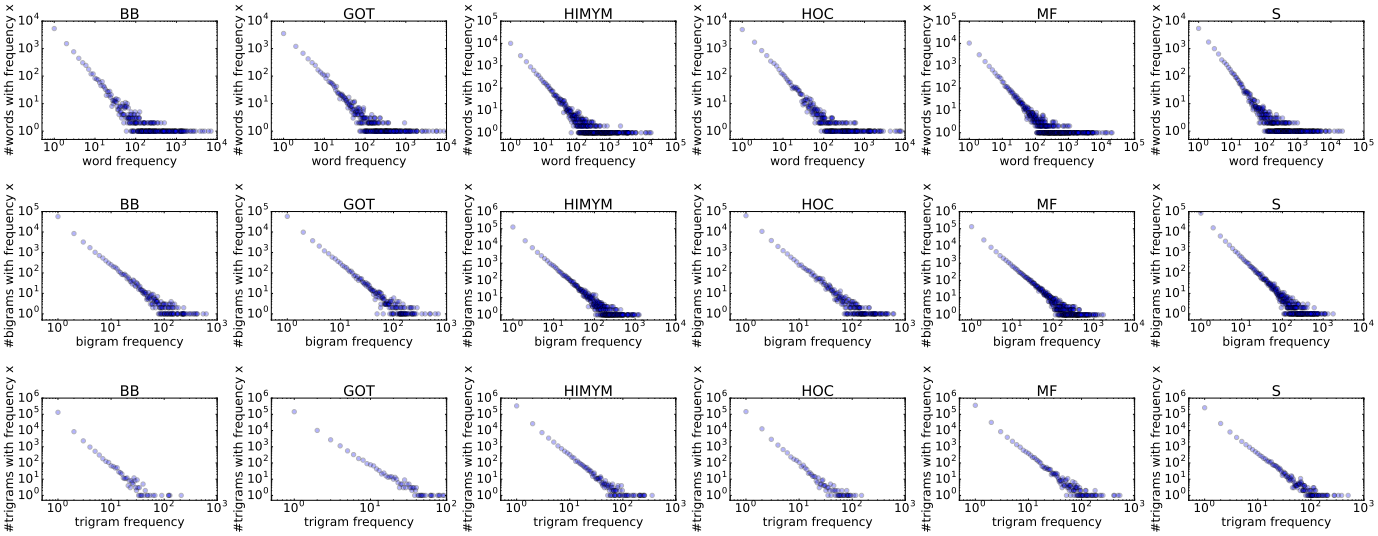


Figure 1: N-grams frequencies for the TV series

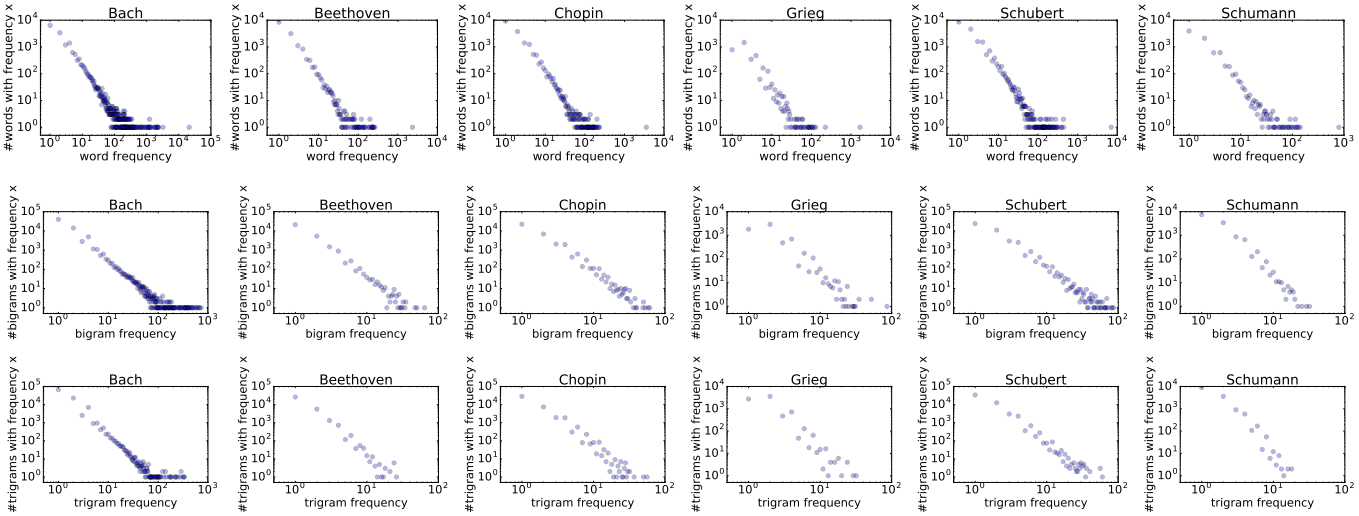


Figure 2: N-grams frequencies for the composers

Davis, H. and Mohammad, S., (2014). *Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL)*, chapter Generating Music from Literature, pages 1–10. ACL.

Happé, F. (1999). Autism: cognitive deficit or cognitive style? *Trends in cognitive sciences*, 3(6):216–222.

Heaton, P., Hermelin, B., and Pring, L. (1998). Autism and pitch processing: A precursor for savant musical ability? *Music Perception: An Interdisciplinary Journal*, 15(3):291–305.

Longfellow, H. (1835). *Outre-mer: A Pilgrimage Beyond the Sea*. Number v. 1-2 in Outre-mer: A Pilgrimage Beyond the Sea. Harper.

Mihalcea, R. and Strapparava, C. (2012). Lyrics, music, and emotions. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 590–599. ACL.

Mottron, L., Dawson, M., Soulieres, I., Hubert, B., and Burack, J. (2006). Enhanced perceptual functioning in autism: An update, and eight principles of autistic perception. *Journal of autism and developmental disorders*, 36(1):27–43.

Patel, A. D. and Daniele, J. R. (2003). An empirical comparison of rhythm in language and music. *Cognition*, 87(1):B35 – B45.

Powers, D. M. W. (1998). Applications and explanations of zipf's law. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, NeMLaP3/CoNLL '98, pages 151–160, Stroudsburg, PA, USA. ACL.

Strapparava, C., Mihalcea, R., and Battocchi, A. (2012). A parallel corpus of music and lyrics annotated with emotions. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

White, J. D. (1976). *The analysis of music*. Englewood Cliffs, N.J. : Prentice-Hall. Includes bibliographical references and index.

Up-cycling Data for Natural Language Generation

Amy Isard, Jon Oberlander, Claire Grover
Language Technology Group,
Institute for Language, Cognition and Communication,
School of Informatics, University of Edinburgh
10 Crichton St, Edinburgh, UK
{amy.isard, claire.grover}@ed.ac.uk

Abstract

Museums and other cultural heritage institutions have large databases of information about the objects in their collections, and existing Natural Language Generation (NLG) systems can generate fluent and adaptive texts for visitors, given appropriate input data, but there is typically a large amount of expert human effort required to bridge the gap between the available and the required data. We describe automatic processes which aim to significantly reduce the need for expert input during the conversion and up-cycling process. We detail domain-independent techniques for processing and enhancing data into a format which allows an existing NLG system to create adaptive texts. First we normalize the dates and names which occur in the data, and we link to the Semantic Web to add extra object descriptions. Then we use Semantic Web queries combined with a wide coverage grammar of English to extract relations which can be used to express the content of database fields in language accessible to a general user. As our test domain we use a database from the Edinburgh Musical Instrument Museum.

Keywords: Natural Language Generation, Cultural Heritage, Semantic Web

1. Introduction

There are large collections of cultural heritage data which are currently not able to be widely shared and exploited because they are stored in databases whose structure and format are inaccessible to the general public. We use a number of Natural Language Processing techniques to bridge the gap between existing databases and Natural Language Generation systems to create varied, adaptive texts which can be tailored to particular visitors on a journey through an exhibition.

A number of research projects have focussed on using NLG systems to create multilingual adaptive texts from structured cultural heritage data. ILEX (O'Donnell et al., 2001) and M-PIRO (Isard et al., 2003; Oberlander et al., 2008) worked from hand-authored resources. They used language-independent databases in a specific format containing up to a fifty objects and a few hundred triples describing attributes of the objects, created in collaboration with curators. The linguistic resources for each language available (English for ILEX, and English, Italian and Greek for M-PIRO) were also hand-created by computational linguists. The generated texts could be tailored to a museum visitor's progress through an exhibition, allowing for comparisons between exhibits, and preventing the repetition of background information. More recent systems have generated texts from Semantic Web ontologies; NaturalOWL (Galanis and Androutsopoulos, 2007; Androutsopoulos et al., 2013) generated texts in English and Greek from OWL ontologies, and Dannélls et al. (2013) generated texts from Semantic Web data in 15 languages, but in both cases expert input was still required to create the necessary linguistic resources.

Sun and Mellish (2007), Mellish and Pan (2008), Mellish (2010) and Androutsopoulos et al. (2013) have experimented with performing NLG using OWL/RDF ontologies which do not have domain-dependent linguistic resources, using the relations provided by the ontologies as a starting point for the English presentation of the facts represented, and Gardent et al. (2017) have used DBpedia (Lehmann et al., 2015) crowdsourcing methods to extract large numbers of linguistic resources which can be used by NLG systems.

However, many museum databases contain information which is structured, but less regular than that found on the Semantic Web. Data may have been annotated over a number of years by multiple authors before being collected together, and the relation names used cannot always be relied upon to contain the information necessary to derive resources suited to NLG. From an NLG point of view, the museum data is often *inconsistent*, for example where the same date or company appears in multiple versions, *insufficient*, for example where it is not clear how to express a given relation, and *incomplete*, in that there is further information which could be added from other sources to enrich the texts presented to a visitor. We aim to bridge this gap by using automatic methods which can be applied to any museum database in any domain to provide all of the resources needed to generate texts using the Methodius NLG system (Isard, 2016), which generates texts from structured data (described in Section 2.2.).

We use the Edinburgh Musical Instrument Museum (MIMEd) as an example domain throughout the paper, but the techniques are designed to be used with any Cultural Heritage dataset. In Figure 1 we show parts of the current MIMEdh web pages for two cornets and a bassoon - the information displayed comes directly from the database and is not very engaging for a museum visitor. In contrast, Figure 2 shows a mock-up of a potential visitor experience of a virtual web musem visit using texts generated by Methodius using the techniques we will describe below. In this example, the visitor has first selected a cornet, followed by another cornet and then a bassoon. The current web page gives the information about dates and makers, but the Methodius texts put the facts in context and also contain some background information about the instruments

and companies involved. This paper describes the methods used to automatically acquire the extra data necessary for the generation of the texts with the minimum of expert input.

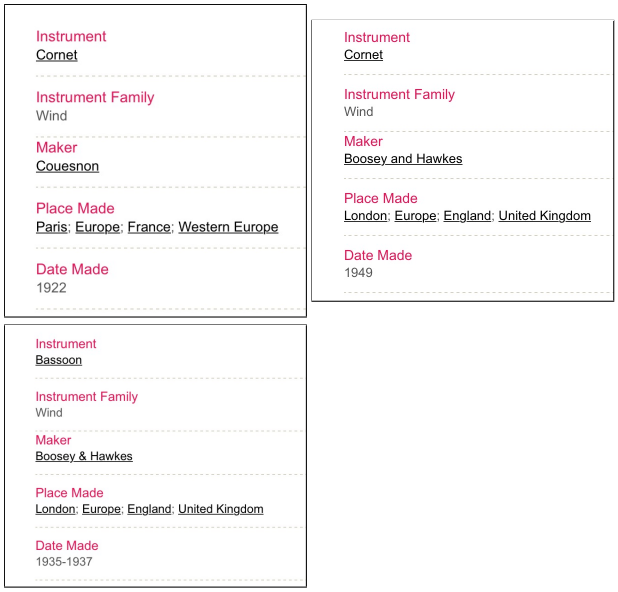


Figure 1: Screenshots of text from current Musical Instrument Museum web page, including canned text snippets

In the rest of the paper, we first describe the Edinburgh Musical Instrument Museum data which we have used as our test domain (see Section 2.1.) and the Methodius NLG system (see Section 2.2.). Figure 3 shows a summary of the automatic processing which we carry out in order to create the resources used as input by the Methodius system. We then describe the three stages of automatic domain processing which we have carried out. Firstly we perform data and name normalization, described in Section 3.1. We then extract relations which are used to link objects to their descriptions (see Section 3.2.). Finally we add descriptions of objects where information is available from DBpedia (see Section 3.3.). We conclude with some ideas for evaluations and future work (see Section 4.).

2. Background

2.1. Musical Instruments Museums Edinburgh

We are using as our test domain the Musical Instruments Museums Edinburgh Collection (MIMEd)[1], which was originally part of the European Musical Instrument Museums Online project (MIMO)[2]. The MIMEd collection contains photos and metadata information for about 5000 instruments. The MIMEd data is stored in an XML format based on the Dublin Core Metadata Initiative[3]. We have filtered the data in order to select a subset of the fields which can be used for NLG, and to select only the 4232 exhibits which have a specified instrument type.
An example of part of the data for an instrument is shown in Figure 4. In order for this data to be used with the Methodius system, we need to normalize the representation of dates

[1]http://collections.ed.ac.uk/mimed
[2]http://www.mimo-international.com/MIMO/
[3]http://dublincore.org

This instrument is a cornet, made by Couesnon in Paris in 1922. The cornet is a brass instrument very similar to the trumpet, distinguished by its conical bore, compact shape, and mellower tone quality. The most common cornet is a transposing instrument in Bb. It is not related to the renaissance and early baroque cornett.

This instrument is another cornet, made by Boosey & Hawkes in London in 1949. Boosey & Hawkes is a British music publisher purported to be the largest specialist music publisher in the world. Until 2003, it was also a major manufacturer of brass, string and woodwind musical instruments. Formed in 1930 through the merger of two well-established British music businesses, the company owns the copyrights or agencies to much major 20th century music, including works by Bartók, Leonard Bernstein, Britten, Copland, Kodály, Prokofiev, Richard Strass and Stravinsky.

This instrument is a bassoon, manufactured in 1946. Like the last cornet you saw, this bassoon was made in London by Boosey and Hawkes. The bassoon is a woodwind instrument in the double reed family that typically plays music written in the bass and tenor clefs, and occasionally the treble. Appearing in its modern form in the 19th century, the bassoon figures prominently in orchestral, concert band, and chamber music literature. The bassoon is a non-transposing instrument known for its distinctive tone color, wide range, variety of character and agility. Listeners often compare its warm, dark, reedy timbre to that of a male baritone voice. Someone who plays the bassoon is called a bassoonist.

Figure 2: Web museum mock-up displaying Methodius generated texts

(Section 3.1.), and acquire linguistic information so as to be able to generate sentences like “this bassoon was made by Buffet Crampon” (Section 3.2.). In addition, we can add information which is not present in the original database (Section 3.3.).

2.2. Methodius NLG system

The Methodius NLG system (Isard, 2007; Marge et al., 2008; Isard, 2016) is a descendant of the Exprimo system, which was developed during the M-PIRO project (Isard et al., 2003). The M-PIRO web interface allowed users to navigate through a small collection of ancient Greek artefacts by clicking on thumbnail images of the objects. Methodius was designed to be a more robust and modular NLG system, which can deal with collections of at least a million objects, and can be used for any domain in which an ontology of objects and attributes exist.
The system uses a typical NLG architecture based on the pipeline model described in Reiter and Dale (2000), which appears on the right in Figure 3. Once a user has chosen an object in which they are interested, the first phase of the generation is Content Selection where an algorithm is used

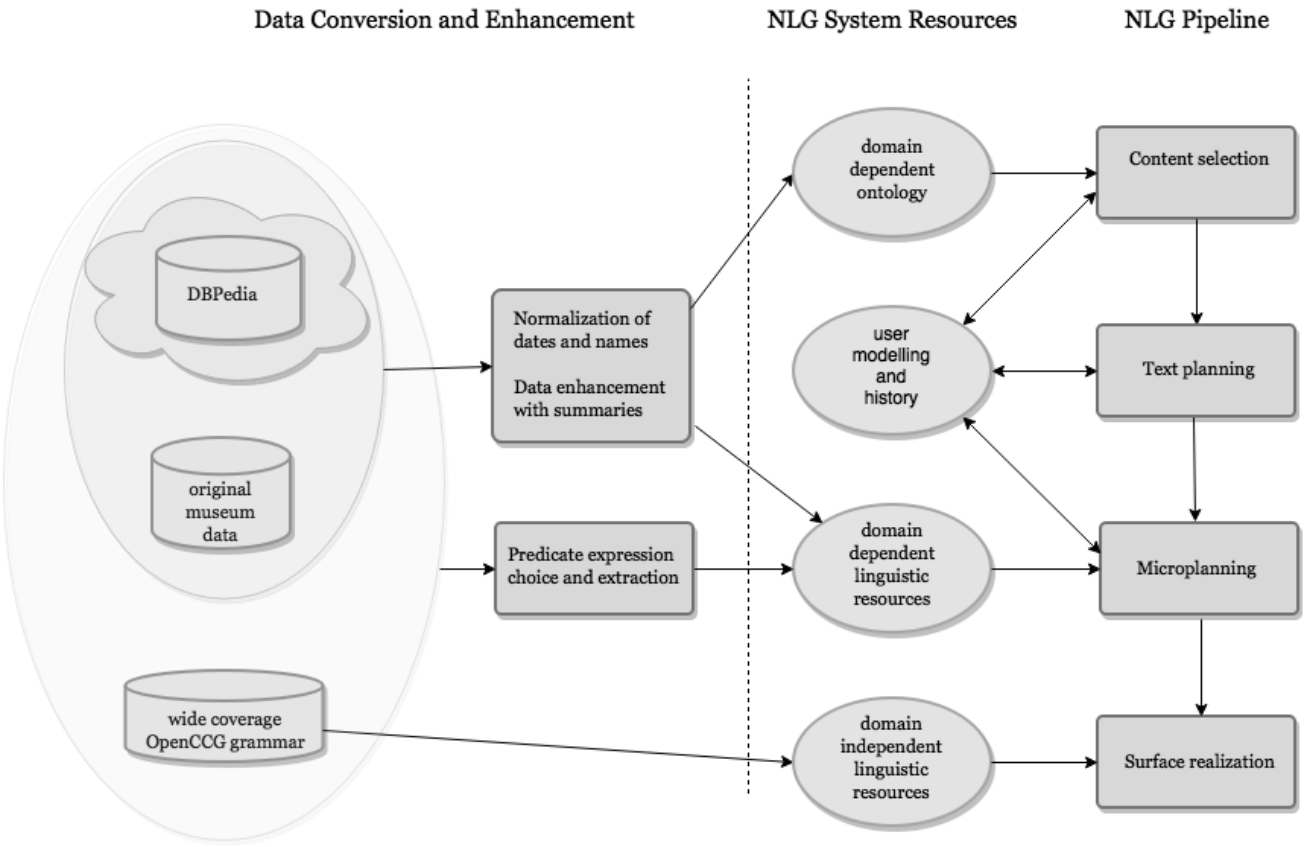


Figure 3: Normalization and enhancement techniques used to create the resources used by the NLG system

```
<dublin_core>
 <dcvalue qualifier="" element="identifier">
  164</dcvalue>
 <dcvalue element="type" qualifier="">
  Bassoon</dcvalue>
 <dcvalue element="contributor"
          qualifier="author">
  Buffet Crampon</dcvalue>
 <dcvalue element="coverage"
          qualifier="temporal">
  1921</dcvalue>
 <dcvalue element="coverage"
          qualifier="spatial">
  Paris
 </dcvalue>
</dublin_core>
```

Figure 4: MIMEd Object Specification

to select a subset of the available facts about the object, based on user modelling information, including a history of previously viewed objects. The next stage is Text Planning, where Rhetorical Structure (Mann and Thompson, 1998; Isard, 2016) is used to group and reorder the selected facts, adding comparisons with previous objects where available (Isard, 2007) and using aggregation rules to combine multiple facts as fluently and coherently as possible. This is followed by Microplanning, during which a logical form representing the sentences is built, and then sent to the Surface Realization component, which outputs the finished text using an OpenCCG grammar (see Section 2.2.2.).

The generated texts can be tailored to a museum visitor's progress through an exhibition, allowing for comparisons between exhibits (Isard, 2007), and preventing the repetition of background information. The resources used by the system consist of a domain ontology containing a hierarchical structure of the types of entities included in the domain, a set of domain-dependent linguistic resources, a set of domain-independent linguistic resources for each language for which texts are to be produced, and a user model and history, which stores a representation of each user's progress through a collection of objects in a virtual or real museum.

2.2.1. Methodius Domain Files

The information for a particular Methodius domain ontology is stored as a set of XML files which represent information about entities and their attributes, and the relationships between entities. The files consist of:

a hierarchy of entity types as in the example in Figure 5, which states that in this domain, the parent of type "bassoon" is "wind" and the parent of type "wind" is "instrument". It also states the noun to be used for

```
<type name="bassoon">
 <parents>
  <parent name="wind"/>
 </parents>
</type>
<type name="wind">
 <parents>
  <parent name="instrument"/>
 </parents>
</type>
```

Figure 5: Extract of Methodius Type Hierarchy

```
<defobject type="bassoon"
           is="object164">
 <role slot="maker" filler="Buffet Crampon"/>
 <role slot="creation-time" filler="1921"/>
 <role slot="original-location
       filler="Paris"/>
 </defobject>
```

Figure 6: Methodius Object Specification

each type, which links to the OpenCCG grammar (see Section 2.2.2.).

a set of entity instances with associated type, each of which has a number of fields containing attribute-value pairs, such as the ones shown in Figure 6 - this example contains the information extracted from the MIMEd data in Figure 4, which states that this exhibit is of type bassoon, was made by Buffet Crampon, was created in 1921 and was originally from Paris.

a set of linguistic text plans , such as the one shown in Figure 7, which states that the "maker" field applies to any object of type exhibit, and is expressed using the verb "make-verb" in the passive voice and the preposition "by". This information is used to build the appropriate logical form for the OpenCCG grammar, as described in Section 2.2.2.

2.2.2. OpenCCG Grammar

During the Surface Realization stage of the NLG process, Methodius uses the OpenCCG library, which provides parsing and realisation tools based on the CCG grammar formalism (White, 2006; White et al., 2007). It depends on a lexicon, a set of grammar rules, and a logical form which describes the structure of the content to be generated. In Methodius, the logical form is created during the Microplanning stage, described above, and then passed to the OpenCCG generation component, which produces the final text. Previous Methodius domains have relied on hand-written lexicons, but as part of this research we will use the wide coverage grammar of English provided with OpenCCG (White, 2014) after extracting the necessary domain relations, as described in Section 3.2.

```
<expression id="maker">
 <arg-one type="exhibit"
          refexp="default"/>
 <arg-two type="entity"
          refexp="default"/>
 <verb tense="past" voice="passive"
       pred="make-verb"/>
 <preposition id="by"/>
</expression>
```

Figure 7: Methodius Linguistic Expression

3. Data Wrangling and Up-Cycling

From an NLG point of view, the museum data is often *inconsistent*, for example where the same date or company appears in multiple versions, *insufficient*, for example where it is not clear how to express a given relation, and *incomplete*, in that there is further information which could be added from the Semantic Web to enrich the texts presented to a visitor.

We apply automatic processing in a number of stages in order to:

- normalize dates and names
- extract a Methodius type hierarchy
- extract modifier terms
- create a list of entities with fields and types
- extract common nouns (types) and proper nouns (entity names) for the OpenCCG lexicon
- automatically add descriptions of instrument types and entities such as companies or people

3.1. Date and Name Normalization

Many of the objects in the MIMEd data have fields which contain a date, but these have been annotated over many years by different authors, and are expressed in a wide variety of formats. There are date recognition software packages such as HeidelTime (Strötgen and Gertz, 2010) and SUTIME (Chang and Manning, 2012), but they were not able to parse many of the dates we found, as they are not geared towards historical dates with mis-spellings and expressions for uncertainty and vagueness. Blog posts from the Bentley Historical Library at the University of Michigan (Pillen, 2015a; Pillen, 2015b) describe techniques similar to ours.

In addition, during the processing we have created a hierarchy of time expressions so that where possible we can use the Methodius algorithms to generate further meaningful comparisons between objects. For example, if a visitor looks at a clarinet made in 1874 and then another made in 1888 we could say "like the last clarinet you saw, this one was made in the late 19th century".

The date normalization was implemented as a Python date processing module. The number of individual dates is greatly reduced by this process; for example the following seven strings from the Mimed data set, are all assigned to the date "second half of the 19th century", with the last three being assigned "possibly"' or "probably" modifiers to be used during generation.

```
Second half of the 19thcentury
Second half of 19thcentury
second half of 19th century
Second half of the 19th century
Probably second half of the 19thcentury
Probably second half of the 19th century
Possibly second half of the 19th century
```

As well as individual years, we process decades, centuries and date ranges, and in addition to the modifiers above, we search for the various ways in which approximate dates are expressed, such as "Circa 1840-1860", "c1860-1879", "1960s or a little later", "1855 or shortly before".

The MIMEd corpus contains a total of 4005 dates, from which we have extracted 941 unique date expressions, with 91 remaining unprocessed at present. Some of the remainder could be processed with the addition of more rules, but others contain so much free text that automatic processing is not possible, for example "1778-1830. The maker's mark is from the earlier part of this period but the hallmark dates from between 1809 and c1819."

The proper names in the database can also be rationalized, by using text processing techniques as well as by taking advantage of the DBpedia `redirect` facility. We perform DBpedia queries on all of the names we find in the corpus in order to add description texts as described in section 3.3., and this also allows us to gather together names which are considered to be synonymous by DBpedia. For example, we have many references to the manufacturer Boosey & Hawkes. As with the dates, we begin by extracting "probably" and "possibly" modifiers. We then have the following names, ordered by the number of occurrences in our data:

Boosey & Co 104

Boosey & Hawkes 77

Boosey and Hawkes 27

Boosey and Co. 21

Boosey & Co. Ltd 2

The first three all redirect to "Boosey & Hawkes" on DBpedia, allowing us to group them together. The third and fourth do not initially find a match on DBpedia, but if we apply some further text processing to remove the string "Ltd" and then any trailing full stops, we can capture all five versions. We would like to capture as many variations and typos as possible without writing individual rules for every possible eventuality, so we concentrate on the most frequent instances.

3.2. Extracting Relations

Previous work has looked into extracting domain-dependent linguistic resources from an ontology where the resources are not already available (Androutsopoulos et al., 2013; Han et al., 2015). These techniques relied on the ontology terms being linguistically related to the appropriate English terms, but In contrast, in the MIMEd data, the Dublin Core roles do not always directly relate to the English meaning.

For example, in the example from the MIMEd data in Figure 4, the role "author" is used for instrument manufacturers, and this role cannot be used directly to express the relationship in English. We therefore use a two-stage process to attempt to automatically find possible ways of expressing the relationship between instruments and manufacturers. First, we take a list of all of the instrument and manufacturer pairs in the domain, and also use the instrument type hierarchy to include all of the direct parents of each instrument. We then use SPARQL[4] queries to DBpedia (Lehmann et al., 2015) to retrieve all texts which describe one of the entities, and select all the sentences which contain the other half of the pair in question. Because we have many objects which have the same role, we have many opportunities to find suitable expressions, and if the same verb occurs repeatedly with different instruments and manufacturers, it can be considered to be an excellent candidate. For example, using the pair "bow" and "James Tubbs", we first retrieve the text below from DBPedia.

> James Tubbs (b 1835-d 1921)- one of the most celebrated English bow makers, and is considered "The English Tourte". Together with his son Alfred (d. 1912) they produced more than 5000 bows. It is universally accepted that James Tubbs ranks among the five or six most important bow makers in history. The Tubbs family made bows and instruments as early as the 1800s, and five generations have practiced the craft. In 1885 he won a Gold medal for his bows at the Inventions Exhibition held that year in London, after which he was made bowmaker by Special Appointment to HRH the Duke of Edinburgh.

We then parse each sentence in the text using the OpenCCG wide coverage grammar of English (White, 2014), and look for sentences in which there is a main verb which has the instrument as object and another noun phrase as the subject. We allow all subject noun phrases as there are many instances where the subject of the sentence is a different phrasing of the original name, a combination of names, or a pronoun. Although this means that some of the sentences will not in fact refer to the desired subjects, we make the assumption that there will be enough results in total to ensure that the incorrect verbs which may be captured will be infrequent and therefore not chosen as the top candidates.

As an example from the MiMEd data, the OpenCCG parse of a partial sentence "they produced more than 5000 bows" is shown in Figure 8. Here we have the main verb "produce.01" with subject "they" and object "bows" (with several modifiers, which we ignore). From the text above we find two potential main verbs - make.01 and produce.01, and we can then create Methodius expressions using just the subject, object and main verbs which will be used to generate sentences for this role, which in this case would eventually result in the generated sentences "James Tubbs produced bows" and "James Tubbs made bows". Because many instruments share the same role, we will also be able to generate for any other instrument which has an "author" field.

[4] https://www.w3.org/TR/rdf-sparql-query/

```
<node id="w10" pred="produce.01"
      tense="past">
 <rel name="Arg0">
  <node id="w9" pred="they" />
 </rel>
 <rel name="Arg1">
  <node id="w14" pred="bows" det="nil"
        num="sg">
   <rel name="Mod">
    <node id="w13" pred="5000">
     <rel name="Mod">
      <node id="w12" pred="than">
       <rel name="Arg1">
        <node id="w11" pred="more"/>
       </rel>
      </node>
     </rel>
    </node>
   </rel>
  </node>
 </rel>
</node>
```

Figure 8: OpenCCG parse

At the moment we are only investigating verbal expressions, but we plan to include noun phrases in future, so that we can also capture phrases such as "bow maker" or "instrument manufacturer".

3.3. Adding Description Snippets

The MIMEd metadata does not contain any descriptions of types of instruments, or of any entities mentioned in descriptions of individual instances, such as companies or people. We have already retrieved descriptions of some entities from DBpedia during the extraction of relations, and we also collect descriptions of instrument types where available, and add them to the Methodius data to provide richer descriptions. This not only allows us to add the text snippets to the MIMEd data, but also provides a method for entity disambiguation as described in Section 3.1.. We used SPARQL queries to download the rdfs:comment fields for all the instrument types in the ontology, where there is a DBpedia page available for the instrument, creating a generic instance for the instrument with a link to the text snippet. The snippet itself will be stored in the OpenCCG lexicon and retrieved through a reference in the logical form. For example, the comment field retrieved for bassoon is:

> The bassoon is a woodwind instrument in the double reed family that typically plays music written in the bass and tenor clefs, and occasionally the treble. Appearing in its modern form in the 19th century, the bassoon figures prominently in orchestral, concert band, and chamber music literature. The bassoon is a non-transposing instrument known for its distinctive tone color, wide range, variety of character and agility.

```
PREFIX rdfs:
  <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?name ?comment
WHERE {
  SERVICE <http://dbpedia.org/sparql>
  ?instrument rdfs:label ?name .
  FILTER(?name = "Bassoon"@en) .
  ?instrument rdfs:comment ?comment .
  FILTER(langMatches(lang(?comment), "EN"))
  }
}
```

Figure 9: SPARLQ for instrument names

4. Conclusions and Future Work

We have described a number of automatic processing techniques which aim to bridge the gap between existing cultural heritage databases and NLG systems, to allows the generation of fluent and adaptive texts for musem visitors. We described our methods for normalizing dates and names, and adding object description text snippets, and for using the Semantic Web to extract relations which can be used to express the content of database fields. We provided examples from our test domain from the Edinburgh Musical Instrument Museum.

We will carry out a number of evaluations on parts of the extraction process and the output of the Methodius system using the automatically acquired resources. First we will test the acceptability of the extracted relation expressions using a crowdsourcing platform. When the system is complete, we will generate texts as part of a virtual museum experience, and evaluate the acceptability of the texts on a number of levels including fluency and coherence, and also relating to a number of other new features currently in the process of being added to Methodius but not part of the work described here.

5. Acknowledgements

This work is funded by a UK EPSRC PhD studentship. We would like to thank Vasilis Karaiskos for helpful comments on drafts of the paper. None of this would have been possible without Jon Oberlander and many years of entertaining discussions, encouragement, enthusiasm and support.

6. Bibliographical References

Androutsopoulos, I., Lampouras, G., and Galanis, D. (2013). Generating natural language descriptions from OWL ontologies: the NaturalOWL system. *Journal of Artificial Intelligence Research*, 48:671–715.

Chang, A. X. and Manning, C. D. (2012). SUTIME: A Library for Recognizing and Normalizing Time Expressions. In *8th International Conference on Language Resources and Evaluation (LREC 2012)*, May.

Dannélls, D., Ranta, A., Enache, R., Damova, M., and Mateva, M. (2013). Multilingual access to cultural heritage content on the Semantic Web. *Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 13)*, August.

Galanis, D. and Androutsopoulos, I. (2007). Generating Multilingual Descriptions from Linguistically Annotated OWL Ontologies: The NaturalOWL System. In *Proceedings of the Eleventh European Workshop on Natural Language Generation (ENLG 07)*, pages 143–146, Stroudsburg, PA, USA.

Gardent, C., Shimorina, A., Narayan, S., and Perez-Beltrachini, L. (2017). Creating Training Corpora for NLG Micro-Planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada.

Han, X., Ioris, A. A., and Lin, C. (2015). Web as Corpus Supporting Natural Language Generation for Online River Information Communication. In *Proceedings of the 24th International Conference on World Wide Web Companion*, pages 363–364. International World Wide Web Conferences Steering Committee.

Isard, A., Oberlander, J., Matheson, C., and Androutsopoulos, I. (2003). Speaking the users' languages. *Intelligent Systems, IEEE*, 18(1):40–45.

Isard, A. (2007). Choosing the best comparison under the circumstances. In *Proceedings of the International Workshop on Personalization Enhanced Access to Cultural Heritage (PATCH07)*, Corfu, Greece, June.

Isard, A. (2016). The Methodius Corpus of Rhetorical Discourse Structures and Generated Texts. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, May.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., and others. (2015). DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2):167–195.

Mann, C., W. and Thompson, A., S. (1998). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 3:243–281.

Marge, M., Isard, A., and Moore, J. (2008). Creation of a new domain and evaluation of comparison generation in a natural language generation system. In *Proceedings of the Fifth International Natural Language Generation Conference (INLG08)*, pages 169–172, Ohio, USA, June.

Mellish, C. and Pan, J. Z. (2008). Natural language directed inference from ontologies. *Artificial Intelligence*, 172(10):1285–1315, June.

Mellish, C. (2010). Using semantic web technology to support NLG case study: OWL finds RAGS. In *Proceedings of the 6th International Natural Language Generation Conference (INLG10)*, pages 85–93, Dublin, Ireland, July.

Oberlander, J., Karakatsiotis, G., Isard, A., Androutsopoulos, I., and others. (2008). Building an adaptive museum gallery in Second Life. *Proceedings of Museums and the Web, Montreal, Quebec, Canada*.

O'Donnell, M., Mellish, C., Oberlander, J., and Knott, A. (2001). ILEX: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(03):225–250.

Pillen, D. (2015a). ArchivesSpace Dating Advice. In *Bentley Historical Library Blog*.

Pillen, D. (2015b). Normalizing Dates with OpenRefine. In *Bentley Historical Library Blog*.

Reiter, E. and Dale, R. (2000). *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Strötgen, J. and Gertz, M. (2010). HeidelTime: High Quality Rule-based Extraction and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 321–324, Los Angeles, California. Association for Computational Linguistics.

Sun, X. and Mellish, C. (2007). An Experiment on "Free Generation" from Single RDF Triples. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, ENLG '07, Stroudsburg, PA, USA.

White, M., Rajkumar, R., and Martin, S. (2007). Towards broad coverage surface realization with CCG. In *Proc. of the Workshop on Using Corpora for NLG: Language Generation and Machine Translation (UCNLG+ MT)*.

White, M. (2006). CCG chart realization from disjunctive inputs. In *Proceedings of the Fourth International Natural Language Generation Conference*, pages 12–19. Association for Computational Linguistics.

White, M. (2014). Towards Surface Realization with CCGs Induced from Dependencies. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, Philadelphia, U.S.A. Association for Computational Linguistics.

Neural Models of Selectional Preferences for Implicit Semantic Role Labeling

Minh Le and Antske Fokkens
Computational Lexicology and Terminology Lab
Vrije Universiteit Amsterdam, the Netherlands
{m.n.le, antske.fokkens}@vu.nl

Abstract

Implicit Semantic Role Labeling is a challenging task: it requires high-level understanding of the text while annotated data is very limited. Due to the lack of training data, most researches either resort to simplistic machine learning methods or focus on automatically acquiring training data. In this paper, we explore the possibilities of using more complex and expressive machine learning models trained on a large amount of explicit roles. In addition, we compare the impact of one-way and multi-way selectional preference with the hypothesis that the added information in multi-way models are beneficial. Although our models surpass a baseline that uses prototypical vectors for SemEval-2010, we otherwise face mostly negative results. Selectional preference models perform lower than the baseline on ON5V, a dataset of five ambiguous and frequent verbs. They are also outperformed by the Naïve Bayes model of Feizabadi and Pado (2015) on both datasets. We conclude that, even though multi-way selectional preference improves results for predicting explicit semantic roles compared to one-way selectional preference, it harms performance for implicit roles. We release our source code, including the reimplementation of two previously unavailable systems to enable further experimentation.

Keywords: neural network, implicit semantic role labeling, selectional preferences

1. Introduction

Defined as the recovery of semantic roles beyond immediate syntactic structure, implicit Semantic Roles Labeling (iSRL) can contribute valuable information for obtaining complete semantic interpretations of text. Yet, it has been elusive since its first shared task eight years ago (Ruppenhofer et al., 2010).

The main difficulty faced by researchers is the small size of training data. Compared to traditional SRL datasets, SemEval-2010 is hundreds-fold smaller, containing only slightly more than a hundred of training examples (Table 1). Early work applying traditional semantic role labeling (SRL) techniques to iSRL was met with deflating results. Therefore, researchers limited themselves to simplistic machine learning models such as Naïve Bayes (Feizabadi and Pado, 2015, among others) or abandoned machine learning altogether (Laparra and Rigau, 2013). Several studies were devoted to the automatic expansion of training data (see Section 2 for an overview).

This paper presents an attempt to recover implicit semantic roles using neural networks. We take advantage of the fact that OntoNotes contains a vast amount of manually annotated explicit semantic roles from which we can learn the selectional preference of frames (e.g. *look.01* prefers animate fillers for role *A0* (looker)). A neural network is used to capture complex interactions between a predicate, a target role and its co-occurring roles. In addition, we compare the impact of one-way selectional preference, taking only the selectional preference of the predicate for the target role into account, to multi-way selectional preference, which uses information from all semantic roles related to the predicate.

The contribution of this paper is twofold: First, we experimented with a class of simple neural models for iSRL and two types of selection preference. While the results are mostly negative, they highlight the importance of discourse information (see Section 4.4 and 4.5) and suggest future directions that should (not) be taken.

	SemEval		OntoNotes
	Train	Test	
Words	8K	9K	1,700K
Frames	344	371	7,007
Predicates	811	1,008	324,996
Predicates with DNI	102	118	0

Table 1: Statistics of an iSRL dataset (SemEval-2010, PropBank version) and a traditional SRL dataset (OntoNotes).

The second contribution lies in addressing the challenges we met in carrying out this research and interpreting our results. The nature of these challenges lies in the fact that (1) all resources for implicit Semantic Role Labeling are small, (2) previous approaches differ in the dataset and the metrics they use for evaluation, and (3) to our knowledge, none of the existing systems is available as open source code. This has led to a situation that is typical for challenging tasks using small datasets: it is almost impossible to determine what the state-of-the-art approach is and how new work relates to this. Even results from papers that are evaluated on the same dataset are difficult to compare, because differences in results can be due to the difference in features, machine learning algorithm, method of extending data, heuristics or (as pointed out in Fokkens et al. (2013)) choices in preprocessing and data preparation. As part of this research, we built an experimental platform for iSRL. This platform provides open source implementations for the experiments reported in this paper, for the system described in Schenk and Chiarcos (2016) which inspired our own approach and for Feizabadi and Pado (2015)'s approach which provided state-of-the-art performance on SemEval-2010.

The rest of the paper is organized as follows: Section 2 summarizes the foundation of iSRL and related work. In Section 3, we outline our models of selectional preference. Section 4 quantifies the effectiveness of selectional preference with regard to iSRL. Section 5 concludes the work and outlines future directions of research.

2. Background and Related Work

In this section, we explain what implicit Semantic Role Labeling entails. This is followed by an overview of previous work on this task. Next, we address related work that uses selectional preferences. Consider the following sentence from SemEval-2010 training set:

(1) Apparently [the tenants]$_{A0}$ had [brought]$_{bring.01}$ [little or nothing]$_{A1}$ with them, and all the furniture down to the smallest details had been taken over with [the house]$_{A2}$.

The roles *A0* and *A1* of the predicate *bring.01* can be filled with phrases in the immediate syntactic structure while the filler of *A2* falls into a separate clause. Typically, a SRL system would annotate the fillers for *A0* and *A1* and ignore *A2*. It is therefore called a Null Instantiation (NI).

Null-instantiations can be indefinite (INI) and definite (DNI). To reuse examples from Ruppenhofer et al. (2010), in the blog headline *More babbling about what it means to know*, the subject of knowing is not expected to be instantiated within the discourse. In contrast, in the sentence *Don't tell me you didn't know!*, the hearer expects a concrete filler for the role of what (s)he should know and it can be expected to be present in previous context. The first example is a case of INI while the second is a DNI.

2.1. Previous work on iSRL

Traditional SRL techniques led to very low results for iSRL due to data sparseness (Chen et al., 2010; Tonelli and Delmonte, 2010). Researchers therefore explored simpler alternatives such as BayesNet (Silberer and Frank, 2012; Roth and Frank, 2013; Roth and Frank, 2015), Naïve Bayes (Feizabadi and Pado, 2015), and memory-based learning (Schenk et al., 2015). Others proposed non-parametric approaches such as observed frequency (Laparra and Rigau, 2012), prototypical vectors (Schenk and Chiarcos, 2016) and other heuristics (Laparra and Rigau, 2013; Gorinski et al., 2013).

In addition to methods of machine learning and heuristics, previous work investigated the possibilities of increasing training data. Feizabadi and Pado (2015) combine multiple corpora and apply domain adaptation methods to deal with the difference in genre. They demonstrated that combining two iSRL corpora led to improved performance. Silberer and Frank (2012) and Roth and Frank (2015) used heuristics to generate iSRL training examples from manually and automatically annotated SRL corpora. This work differs from these approaches, because their research focused on creating iSRL training examples of reasonable quality rather than using a SRL resource directly.

2.2. Selectional preferences

Selectional preference has a long research tradition (Katz and Fodor, 1963) and has been applied in various tasks such as syntactic parsing (Zhou et al., 2011), textual inference (Ritter et al., 2010), and semantic role labeling (Zapirain et al., 2013). The idea is simple: a role is filled with some words more frequently than others. For example, *the man* is much more likely a filler for the role *A0* (leader) of the predicate *lead.01* than e.g. *the bottle* (an inanimate object) although one can construct a grammatically and semantically correct example for each filler.

Next to the role's semantics, co-occurring roles also have an influence. For example, if *lead.01*'s role *A4* (goal) is filled by *the guest house*, *the nation* is an implausible filler for *A1* (thing led), while it is perfectly plausible had we not known what fills *A4*. This is known as *multi-way selectional preference* (van de Cruys, 2014).

One-way selectional preferences have been applied to implicit semantic role labeling before. Silberer and Frank (2012)'s system include a feature calculated using weighted similarity to head words that are observed to fill a role. The selectional preference model itself is described in (Erk, 2007) and (Resnik, 1996). A simpler model that uses unweighted similarity is used by Schenk and Chiarcos (2016). Our results show that adding multi-way selectional preference improves results on explicit semantic roles, but not for iSRL. Recently work by Do et al. (2017) is closest to our work but they apply their methods on nominal data and did not compare one-way and multi-way selectional preference.

2.3. Neural networks

Recent years have witnessed a surge of research interest in neural networks for natural language processing (Goldberg, 2016). Plenty of models have been proposed for various tasks (Godbole et al., 2015; Zhou and Xu, 2015; Andor et al., 2016, among many others). Apart from Do et al. (2017) who uses a different architecture for a different version of the task, we are not aware of work that applies neural networks to iSRL.

3. Models

In this study, we focus exclusively on *DNI resolution*, the last and hardest step in iSRL. For each test case, we assume that the predicate p is already identified and disambiguated, the target role r^* is given, and overt roles are coupled with their fillers $\{(r_j, g_j)|j = 1..m\}$. The goal is to rank the correct filler highest among the candidates $\{c_i|i = 1..n\}$. To test a simple multi-way selectional preference model, we use the following formula to assign a score for each candidate:

$$s(c_i) = \bigoplus_{j=1}^{m} f(p, r^*, r_j, g_j, c_i) \quad (1)$$

where m is the number of explicit roles known to the system, $\bigoplus$ is a aggregation function (e.g. *sum* or *max*). In the case of *one-way* selectional preference, the formula degenerates into:

$$s'(c_i) = f'(p, r^*, c_i) \quad (2)$$

f and f' are neural networks that have the same architecture, except the number of inputs. The precise form of the neural networks and the aggregation function is determined via a hyperparameter search (see Section 4.3).

Role fillers $\{g_j\}$ and candidates $\{c_i\}$ can be transformed into features by extracting the head word, but other features can also be used. Together with predicate and role names, they are embedded into a vector space. The embedding matrix is trainable and can be initialized with pretrained word vectors for better performance.

Compared to a model that computes prototypical vectors as the average of observed vectors in the fashion of Schenk and Chiarcos (2016), our neural models have (at least) two advantages:

- Distributed representation is used to represent predicates and roles, not only fillers, allowing the model to work in cases of unseen predicates or predicate-role combinations.
- The representation enables the sharing of statistical strength between predicates, i.e. rare predicates can get more accurate predictions by means of resemblance to frequent predicates.

An additional motivation is that multi-way preference can offer a solution to the context-dependent nature of semantic role labeling. Our current results, however, do not provide sufficient evidence to support such a claim.

4. Experiments

We evaluate our models for DNI resolution by comparing our model to a baseline and to Feizabadi and Pado (2015). In addition, we perform an ablation analysis to find out which components of the model are useful.

4.1. Data

We train our selectional preference model on OntoNotes (Weischedel et al., 2013), a balanced 1.7M words corpus with over 320K manually annotated predicates and their explicit arguments.

SemEval-2010 (Ruppenhofer et al., 2010) is a standard dataset to evaluate iSRL systems. It contains chapters of Sherlock Holmes, one for training and two for testing, annotated with both implicit and explicit semantic roles. The organizers provide two versions of the same dataset: one annotated with FrameNet roles and the other PropBank. Because OntoNotes was compiled using PropBank, we also use the PropBank version of SemEval-2010.

Note that OntoNotes differs from SemEval-2010 in task (explicit versus implicit SRL), genres (news, weblogs and conversations versus novel) and time period (20th century versus 19th century). Training on OntoNotes SRL and testing on SemEval-2010 iSRL can be seen as a form of domain adaptation and requires powerful generalization.

We also test our models on ON5V (Moor et al., 2013) which poses a different challenge. Implicit semantic roles were manually annotated on top of explicit semantic roles and other linguistic information on a selection of OntoNotes documents. The authors chose five highly frequent verbs to annotate in order to create "high-volume of annotations for specific verb predicates".[1] As a result, the words and phrases that fill each role are very diverse, as illustrated by the examples in Table 2. To achieve high performance on this dataset, a model needs to be selective yet general enough to encompass different types of fillers.

Predicate	Role	Filler head	Filler full
pay.01	A2	refugees	the refugees
pay.01	A2	U.S.	U.S.
pay.01	A2	families	the victims' families
pay.01	A2	trust	this trust
pay.01	A2	Warner	AOL Time Warner
pay.01	A2	they	they
pay.01	A2	lenders	lenders
pay.01	A2	lawyers	lawyers
pay.01	A2	one	one

Table 2: Examples from ON5V showing the diversity of fillers in terms of semantic types, part-of-speech, and topics.

4.2. Baseline

Our baseline is inspired by Schenk and Chiarcos (2016). For every <predicate, role> pair found in our training set, it computes a prototypical vector and, at test time, returns the candidate that is closest to the prototypical vector. Following their best model, we use the pretrained embeddings from Collobert et al. (2011).

Due to some differences between research questions and experimental setup, the results cannot be compared to Schenk and Chiarcos's algorithm directly. Firstly, for a fair comparison with selectional preference-based models, we use only the head word of each candidate (whereas they average all words in the phrase). Secondly, we train and evaluate on PropBank-style datasets while they use FrameNet-style data.

4.3. Experimental Setup

We use the baseline described in the previous section and a Naïve Bayes model trained on SemEval-2010 data (Feizabadi and Pado, 2015) to compare to our model's performance. To quantify the effect of different aspects of the model, we investigate the following variants:

ONEWAY captures one-way selectional preference and represents fillers by their syntactic head.

MULTIWAY captures multi-way selectional preference and represents fillers by their syntactic head.

SYNSEM uses richer features for fillers rather than selectional preferences. We use five syntactic and semantic features from Feizabadi and Pado (2015), namely, *Expected roles*, *Semantic Type*, *Word Frequency*, *POS*, and *Constituent type*.[2]

[1] They are: *pay*, *give*, *bring*, *leave*, *put*.

[2] See Table 2 in their paper. We did not use their discourse features because they require iSRL annotations which is not available in OntoNotes.

	Train (%)	Validation (%)
ONEWAY	59.52	46.61±0.23
MULTIWAY	58.52	47.64±0.19

Table 3: Accuracy of selectional preference models on OntoNotes (for validation set we report mean and standard deviation over 5 runs).

SYNSEM+ONEWAY combines richer features with one-way selectional preference.

SYNSEM+MULTIWAY takes into account co-occurring roles to capture multi-way selectional preference.

We construct one training example for each argument found in OntoNotes and split the data into 90% for training and 10% for development. Models are trained to choose the right filler for each target role with as input: the predicate, the role and, if applicable, other explicit arguments. We evaluate on the NI-only test set from SemEval-2010 using the standard evaluation script (Ruppenhofer et al., 2010).

We initiated the embedding matrix with 27K vectors from the pretrained embeddings of Collobert et al. (2011). We use AdaGrad (Duchi et al., 2011) for optimization; the initial learning rate was customized for each model to avoid gradient explosion. All models were trained until no improvement was observed on the development set (but not more than 1,000 epochs, for practical reasons). To account for random initialization in neural networks, we run each model 15 times and average the results. An arbitrary but fixed random seed was used for each run to ensure reproducibility.

All hyperparameters were tuned on OntoNotes development set. We tested sum and max for aggregation function; sigmoid, tanh, and cube for activation function (Chen and Manning, 2014); different strength of dropout (Hinton et al., 2012), regularization, and learning rate. Because of limited computational resource, we performed a random hyperparameter search to find the best setting. As discussed in Section 3, fillers can be represented in different ways. We observed that using both the head word and the closest coreferent non-pronoun head word is better than using the head word only on our development set. Notice that gold coreference chains are assumed to be available at test time and were used in previous work (Silberer and Frank, 2012) as well as the system we compare to (Feizabadi and Pado, 2015).

The source code of all experiments, including random seeds and replication instructions, is publicly available at: `https://bitbucket.org/cltl/isrl-sp`.

4.4. Results on SemEval-2010

Table 3 shows the performance of selectional preference models with regard to resolving the explicit roles of OntoNotes. Selectional preference alone (without the help of syntactic structures) can find the correct filler in more than 47% of the cases. We observe a small but statistically significant ($p < 0.05$) improvement on the validation set by adding multiway selectional preference.

The results in Table 4 show that our models significantly increase the F_1 score above the baseline on SemEval-2010 dataset. Both neural models show significant improvement in precision and an even bigger improvement in recall. This can be attributed to their ability to generalize to unseen predicate-role combinations and abstract away from observed ones in their hidden layer. Contrary to our expectation, MULTIWAY is inferior to ONEWAY ($p < 0.05$). ONEWAY and MULTIWAY do not outperform F&P which is simpler in terms of machine learning architecture, but is trained on in-domain, iSRL data, and uses more features. To bridge the gap between the models, we also integrate Feizabadi and Pado's syntactic and semantic features into our neural models but they do not lead to improved performance.

Table 3 and Table 4 reveal an increase in random fluctuation when moving from OntoNotes to SemEval-2010, probably because of a difference of some orders of magnitude in size. Moreover, SYNSEM+ONEWAY gets an F_1-score of 16.54% for one of its runs (lower than the mean of all other neural models) and 29.43% for another (higher than all means). These observations stress the importance of running experiments multiple times when random factors (such as parameter initialization and the order of training examples) are involved. Based on a single run, a model might be heavily over- or underrated.

	P	R	F_1
Baseline	26.85	21.80	24.07
F&P	35.04	30.83	32.80
ONEWAY	28.80	28.67	28.74 ± 1.63
MULTIWAY	27.02	26.82	26.92 ± 1.42
SYNSEM	16.63	16.54	16.58 ± 1.51
ONEWAY+SYNSEM	24.05	23.91	23.98 ± 5.05
MULTIWAY+SYNSEM	17.29	17.29	17.29 ± 0.00

Table 4: Results on SemEval-2010. Results of neural models are averaged over 15 runs. F_1 scores are reported with mean and standard deviation when possible.

	P	R	F_1
Baseline	13.00	13.00	13.00
F&P	16.72	15.19	15.90
ONEWAY	10.64	10.64	10.64 ± 1.44
MULTIWAY	9.14	9.14	9.14 ± 1.46
SYNSEM	5.92	5.92	5.92 ± 2.10
ONEWAY+SYNSEM	10.37	10.37	10.37 ± 5.50
MULTIWAY+SYNSEM	1.24	1.24	1.24 ± 0.00

Table 5: Results on ON5V

4.5. Results on ON5V

In Table 5, we report the results of models on ON5V (Moor et al., 2013). Again, the Naïve Bayes model using both local and discourse information proposed by Feizabadi and Pado (2015) clearly provides the best performance, whereas neural models do not show improvement over the baseline ($p < 0.05$).

The disappointing performance points to its inherent limitation: it expects one prototypical filler per ⟨predicate, role⟩ pair. As shown in Table 2, this assumption breaks in ON5V, resulting in a lower mean and higher variance.

We expected that MULTIWAY would alleviate this problem by varying the predicted vector based on surrounding roles. While it achieves that for explicit SRL on OntoNotes (Table 3), the result does not carry over to ON5V.

Local syntactic and semantic information do not improve results for SemEval. This applies even more strongly to ON5V. SYNSEM leads to very low results when standing alone and does not improve performance when combined with ONEWAY or MULTIWAY (the difference between ONEWAY and its combination with SYNSEM is not statistically significant). In comparison with F&P, this result emphasizes the importance of discourse information in the task.

5. Conclusions

In this paper, we investigated the use of more expressive machine learning models for implicit Semantic Role Labeling. We proposed novel neural models that use selectional preference and applied them to iSRL. Our empirical results show that neural models are only better than a lookup table of prototypical vectors in a natural setting such as SemEval-2010 while underperforming for highly frequent and ambiguous words in ON5V. Furthermore, the added expressive power does not help neural models to overcome a simpler model trained on in-domain data and equipped with discourse features (though it should be noted that we tested only a small family of simple architectures, cf. Do et al. (2017)). Multi-way preference is found to be helpful in the case of (explicit) semantic role labeling but not for iSRL. Although the results are mostly negative, our research provides hard-earned insights into this challenging task which we believe will be useful for researchers.

We release all of our models and the implementation of Schenk and Chiarcos (2016) and Feizabadi and Pado (2015)'s models as open-source software. We also report the fluctuation of results which stresses the importance of measuring a model multiple times when stochastic factors are involved.

Overall, this paper provides a solid basis for further research. Our observations on fluctuation and significance suggest more evaluation data may be needed to identify the true impact of specific models and features.

6. Acknowledgements

The work presented in this paper was funded by the Netherlands Organization for Scientific Research (NWO) via the Spinoza grant, awarded to Piek Vossen and via VENI grant 275-89-029 awarded to Antske Fokkens. Experiments were carried out on the Dutch national e-infrastructure with the support of SURF Cooperative.

7. Bibliographical References

Andor, D., Alberti, C., Weiss, D., Severyn, A., Presta, A., Ganchev, K., Petrov, S., and Collins, M. (2016). Globally Normalized Transition-Based Neural Networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 2442–2452, Berlin, Germany. Association for Computational Linguistics.

Chen, D. and Manning, C. (2014). A Fast and Accurate Dependency Parser using Neural Networks. In *EMNLP 2014*, pages 740–750. ACL.

Chen, D., Schneider, N., Das, D., and Smith, N. A. (2010). SEMAFOR: Frame argument resolution with log-linear models. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 264–267, Uppsala, Sweden. ACL.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Do, Q. N. T., Bethard, S., and Moens, M.-F. (2017). Improving Implicit Semantic Role Labeling by Predicting Semantic Frame Arguments. *arXiv:1704.02709*.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Erk, K. (2007). A Simple, Similarity-based Model for Selectional Preferences. In *ACL 2007*, pages 216–223, Prague, Czech Republic.

Feizabadi, P. S. and Pado, S. (2015). Combining Seemingly Incompatible Corpora for Implicit Semantic Role Labeling. In **SEM 2015*, pages 40–50.

Fokkens, A., Erp, M., Postma, M., Pedersen, T., Vossen, P., and Freire, N. (2013). Offspring from Reproduction Problems: What Replication Failure Teaches Us. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1691–1701. Association for Computational Linguistics.

Godbole, V., Liu, W., and Togneri, R. (2015). An Investigation of Neural Embeddings for Coreference Resolution. *Computational Linguistics and Intelligent Text Processing*, 9041:241–251.

Goldberg, Y. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, 57:345–420.

Gorinski, P., Ruppenhofer, J., and Sporleder, C. (2013). Towards Weakly Supervised Resolution of Null Instantiations. *Proceedings of IWCS*, pages 119–130.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, pages 1–18.

Katz, J. J. and Fodor, J. A. (1963). The structure of a semantic theory. *Language*, 39(2):170–210.

Laparra, E. and Rigau, G. (2012). Exploiting Explicit Annotations and Semantic Types for Implicit Argument Resolution. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, pages 75–78, 9.

Laparra, E. and Rigau, G. (2013). ImpAr: A Deterministic Algorithm for Implicit Semantic Role Labelling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1180–1189.

Moor, T., Roth, M., and Frank, A. (2013). Predicate-specific Annotations for Implicit Role Binding: Corpus

Annotation, Data Analysis and Evaluation Experiments. In *Proceedings of the 10th International Conference on Computational Semantics*, pages 369–375.

Resnik, P. (1996). Selectional constraints: an information-tehoretic model and its computational realization. *Cognition*, 61(508):127–159.

Ritter, A., Etzioni, M., and Etzioni, O. (2010). A Latent Dirichlet Allocation method for Selectional Preferences. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 423–434.

Roth, M. and Frank, A. (2013). Automatically Identifying Implicit Arguments to Improve Argument Linking and Coherence Modeling. *Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM)*, 1:306–316.

Roth, M. and Frank, A. (2015). Inducing Implicit Arguments from Comparable Texts: A Framework and its Applications. *Computational Linguistics*, 41(4):625–664.

Ruppenhofer, J., Sporleder, C., Morante, R., Baker, C., and Palmer, M. (2010). SemEval-2010 Task 10: Linking Events and Their Participants in Discourse. In *Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010*, pages 45–50, Uppsala, Sweden.

Schenk, N. and Chiarcos, C. (2016). Unsupervised Learning of Prototypical Fillers for Implicit Semantic Role Labeling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1473–1479, San Diego, California, 6. Association for Computational Linguistics.

Schenk, N., Chiarcos, C., and Sukhareva, M. (2015). Towards the Unsupervised Acquisition of Implicit Semantic Roles. In *RANLP 2015*, pages 570–578.

Silberer, C. and Frank, A. (2012). Casting Implicit Role Linking As an Anaphora Resolution Task. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 1–10, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tonelli, S. and Delmonte, R. (2010). VENSES++: Adapting a deep semantic processing system to the identification of null instantiations. In *Proceedings of the 5th International Workshop on Semantic Evaluations, ACL 2010*, number July, pages 296–299, Uppsala, Sweden. Association for Computational Linguistics.

van de Cruys, T. (2014). A Neural Network Approach to Selectional Preference Acquisition. *EMNLP 2014*, pages 26–35.

Zapirain, B., Agirre, E., Màrquez, L., and Surdeanu, M. (2013). Selectional Preferences for Semantic Role Classification. *Computational Linguistics*, 39(3).

Zhou, J. and Xu, W. (2015). End-to-end learning of semantic role labeling using recurrent neural networks. *ACL 2015*, pages 1127–1137.

Zhou, G., Zhao, J., Liu, K., and Cai, L. (2011). Exploiting Web-Derived Selectional Preference to Improve Statistical Dependency Parsing. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1556–1565.

8. Language Resource References

Moor, Tatjana and Roth, Michael and Frank, Anette. (2013). *The OntoNotes 5 Verbs Non-local Role Linking Data Set (ON5V)*. Heidelberg Universität.

Ruppenhofer, Joseph and Sporleder, Caroline and Morante, Roser and Baker, Colin and Palmer, Martha. (2010). *SemEval-2010 Task 10: Linking Events and Their Participants in Discourse*. Association of Computational Linguistics.

Weischedel, Ralph and Palmer, Martha and Marcus, Mitchell and Hovy, Eduard and Pradhan, Sameer and Ramshaw, Lance and Xue, Nianwen and Taylor, Ann and Kaufman, Jeff and Franchini, Michelle and El-Bachouti, Mohammed and Belvin, Robert and Houston, Ann. (2013). *OntoNotes Release 5.0 LDC2013T19*. Linguistic Data Consortium, ISLRN 151-738-649-048-2.

A database of German definitory contexts from selected web sources

Adrien Barbaresi, Lothar Lemnitzer, Alexander Geyken
Berlin-Brandenburg Academy of Sciences
Jägerstr. 22/23 – 10117 Berlin – Germany
{barbaresi,lemnitzer,geyken}@bbaw.de

Abstract

We introduce work on the detection of definitory contexts designed to speed up two lexicographical tasks: searching for the exact meaning(s) of terms and providing usable input for paraphrasing. Our database is built from a specialized web corpus using a robust pattern-based extraction method. The corresponding interface displays information for a large range of lexical units. The contributions of this article are threefold: we describe both acquisition and extraction, provide a qualitative assessment of the method, and present an interface to access the data.

Keywords: Definition extraction, definitory contexts, computational lexicography, resource acquisition

1. Introduction

German is well-known for making extensive use of compounds (Schlücker, 2012), as composition is a productive process of German word formation. In the context of lexical resources and dictionaries, the number of potential words to define is very large if not infinite. For that matter, corpus data is crucial in order to perform lexicographic work. Furthermore, automatic definition extraction can be used to complement existing entries or help lexicographers creating new ones. This article introduces both a corpus and a database built to support lexicographers as well as interested users in sorting out the meaning(s) of lexemes. The acquisition draws on selected web sources containing a large number of lexical descriptions, which are expected to contain potential definitory contexts, from which handwritten definitions will be derived. We describe an experimental setting and front-end for pattern-based definition extraction from such resources.

We first describe the context and motivations of our work. In section 2 we summarize related work and place our contribution into this context. In section 3 we present our acquisition and extraction method. Section 4 describes the database, an interface to it as well as the results of a qualitative evaluation. In section 5 we conclude with the insights gained from our investigation.

1.1. Context and motivation

The Digital Dictionary of the German Language (*Digitales Wörterbuch der deutschen Sprache*, DWDS[1]) is a long term project of the Berlin-Brandenburg Academy of Sciences (BBAW). The goal of the DWDS project (Klein and Geyken, 2010) is to create a large-scale aggregated word information system based on legacy dictionaries, large corpora, word statistics and automated methods to provide additional types of linguistic information as well as to speed up the process of updating and amending the existing lexical resources.

The DWDS web platform provides access to this information and is among the most frequently used academic dictionary platforms. The dictionary component of the DWDS is based on high-coverage and detailed dictionaries of contemporary German, including the *Wörterbuch der deutschen Gegenwartssprache* and *Duden Großwörterbuch der deutschen Sprache*. Thus, the DWDS dictionary component with more than 170,000 entries provides a good coverage of the core vocabulary of German.

Having covered this core vocabulary, the current work focuses on the description of lexical units whose meaning reaches deeply into highly specialized domains. In order to revise the entries of existing legacy dictionaries and to work on the full lexicographic description of headwords that are not yet recorded, a team of 6 lexicographers has been employed for more than 4 years, with another 6 years to come. We would like to present to our users detailed information concerning the form and grammar as well as the meaning of as many headwords as possible. For instance, it has been shown in a corpus-based quantitative study that in a German newspaper corpus of one billion running words, a number of 4-5 million distinct base forms, i.e. lexical units in our sense, could be detected (Klein, 2013). In a more recent investigation using our morphological analyzer and lemmatizer and a corpus of ca. 5 billion running words, we have collected a total of 16.3 million base forms, most of which appear rather infrequently in the corpora.

However, providing hand-crafted definitions for such a large number of headwords is far beyond the scope of our project. That is why any corpus-based support is highly valuable, with two main areas of interest: on one hand the identification of senses and usage of specialized lexical units and on the other hand the elaboration of appropriate definitions (Geyken et al., 2017a).

In this context, it is expected that a specialized corpus and database with definitory contexts from selected web sources is highly relevant. Since we are reaching out into the periphery of the German vocabulary, i.e. targeting lexical units of specialized use in many domains, the research that is necessary for a correct description of the word senses is costly in time and effort. Typically, the lexicographers are not experts in such domains. That is why any support based on a corpus of reliable internet resources can be helpful. Secondly, it would also be an asset for the users of our dictionary if we can provide contexts for a large share of these words that help the users to sort out for themselves the meaning, hyperonyms, translation equivalents of such

[1] https://www.dwds.de/

lexical units. Geyken et al. (2017b) present a study that is based on user queries of the DWDS platform and show that 17% of the queries did not match a headword in our dictionary and would therefore not return any lexical information. A database of definitory contexts on top of our high-quality lexical descriptions would indeed be of great help and improve on coverage with our lexical information system.

In short, the corpus and database of contexts will support lexicographers in crafting definitions and help users of our information systems to glean useful semantic information even for headwords which fall beyond the scope of lexicographic work planned for the DWDS.

2. Related work

Automatic definition extraction has already been applied in a range of different contexts and tasks in computational linguistics (Navigli and Velardi, 2010), and the extraction of definitory contexts from free texts is currently a trending topic in lexicography and neighboring disciplines. Method and corpora are two key issues. The method of choice has been the drafting of lexico-syntactic patterns or features (Hearst, 1992) that more or less precisely describe the surface forms of prototypical definitory contexts. Empirical approaches such as the use of conditional random fields in order to label beginning and content of a definition as well as non-related material (Anke, 2013) detect common patterns such as the use of the verb "to be". While most of the work has focused on English (Borg, 2009; Zhang et al., 2014), there is also work to be found for Slavic languages (Przepiórkowski et al., 2007) and Dutch (Westerhout, 2010). Work on German includes the search for technical terms with a series of structural patterns with verbal predicates (Storrer and Wellinghoff, 2006) as well as methods for proper definition extraction based on hand-crafted rules or patterns, evaluated in numerical terms (Cramer, 2011) or with regard to their statistical relevance (Schumann, 2014). We chose to first follow the work of Cramer (2011) and employ some of her patterns, namely the ones she classifies as most efficient. While a qualitative evaluation of the patterns as well as a classification in terms of efficiency is included, her study does not present any accountable results in empirical, quantitative terms. In this sense, our article can be seen as a replication study on our data. We use the categories introduced by Cramer (2011) and compare them with a baseline consisting of more loosely defined patterns.

The sources used in related work mostly include small domain-specific datasets, e.g. instructional texts compiled for teaching purposes (Borg, 2009); large encyclopedic resources such as Wikipedia (Kovář et al., 2016); or general-purpose web corpora (Navigli et al., 2010); unfiltered noisy data such as the CommonCrawl (Seitner et al., 2016); or small sets of webpages (Schumann, 2014). Our approach is centered on a larger but still specialized web corpus of glossaries and similar lexical resources. While Kovár et al. (2016) for example restrict themselves to a version of the Wikipedia that is part of the *Sketch Engine*, we aim at a richer and more diverse collection of web sources. Having focused on specialized resources, the extraction in our case is slightly easier than on general-purpose texts, work reported e.g. by Zhang et al. (2014), and slightly more complex than in work targeting Wikipedia. The main challenge therefore is to find relevant information in loosely and diversely structured data, namely the headword (*definiendum*) and the defining context (*definiens*).

3. Acquisition, extraction and exploitation

3.1. Definitions and definitory contexts

In a strict sense that is applied e.g. in language philology and terminology, a definition supplies information that is sufficient to explicate either the content (or intension) of a concept or the set of individuals that form the extension of the concept. The classical logical form of such a definition is the formulation of *genus proximum* (the hyponym) and *differentiae specificae* (exactly those features of the concept that makes it distinguishable from all other concepts which are represented by co-hyponyms). For example, in Princeton WordNet the lexical unit *broom* is defined as "a cleaning implement for sweeping".[2] We do not employ this strict sense of definition in our work.

In the wider sense that is commonly employed in lexicography, a definition or, more precisely, a meaning paraphrase, is provided as part of the explication of the meaning of a particular sense of a lexical unit. Such a paraphrase is typically more elaborate than a *definition stricto sensu*, it is derived from examples of the usage of a word (and not necessarily based on an abstract concept) and in many dictionaries it is accompanied by a set of crafted or manually selected usage examples that illustrate the rules and conventions of the usage of the lexical unit. It is therefore not restricted to conceptual information but should also provide typical and relevant world knowledge that is related with the lexical unit. It might be worth mentioning in an extended meaning paraphrase that brooms are typically made of stiff fibers and typically have a long handle.[3] We will, in the following, use the terms *definition* and *definitory context* in that wider sense.[4]

3.2. Examples

Above we argued that for our lexicographical work we are particular interested in definitory contexts for lexical units of highly specialized domains. We will illustrate this point with some examples from our database. The examples consist of the headword, an English equivalent of it (in brackets), the definition proper, a pointer to the source of the definition and some further comments.

Auseinandersetzungsbilanz (dissolution balance)

Eine Auseinandersetzungsbilanz – auch Abschichtungsbilanz genannt – ist die Bilanz einer Personengesellschaft, die als Grundlage für die Auszahlung eines oder mehrerer Gesellschafter dienen soll. Das Ergebnis dieser Bilanz ist das Auseinanderset-

[2] cf. http://wordnetweb.princeton.edu/perl/webwn?s=broom

[3] cf. https://en.wiktionary.org/wiki/broom

[4] For an in-depth discussion of the difference between scientific definitions and lexical meaning paraphrases (Wiegand, 1989).

zungsguthaben. (Source: Anlegerlexikon[5]).
This is a highly specialized and non-transparent term form the domain of company / corporation law. A synonymous term is given as well as the factually related term *Auseinandersetzungsguthaben* (credit balance).

Barett, Birett (Biretta)
Barett – auch Birett – ist die Kopfbedeckung von Akademikern und Geistlichen, ausgezeichnet durch die vier- oder fünfeckige Form. (Source: Das Heiligenlexikon[6]).
This is a simplex from the domain of church and religion. The term Barett itself is ambiguous. In addition, the reader learns something about the form of this kind of headdress.

Direktionsrecht (employer's executive prerogatives)
Unter Direktionsrecht – auch Weisungsrecht genannt – wird das Recht des Arbeitgebers verstanden, die Leistungspflichten des Arbeitnehmers einseitig näher auszugestalten. (Source: Lexikon Recht[7]).
This is a non-transparent compound from the domain of labour relations. Both parts of the compound are highly ambiguous. In addition, a synonymous term, *Weisungsrecht* is given.

Pelletheizung (pellet stove)
Pelletheizung ist eine Holzheizung, in deren Heizkessel zu Stäbchen geformte Holzabfälle – sogenannte Holzpellets – verbrannt werden. (Source: Baulexikon[8]).
This is a very recent term from the domain of heating engineering. This definition follows closely the pattern of *genus proximum* and *differentiae specificae*.

3.3. Acquisition

We chose to build a specialized web corpus, that is a collection of web documents targeting web pages which are defined in advance (Barbaresi, 2015), after identifying and manually selecting a series of relevant websites. We make the hypothesis that there are webpages in which it is probable to find definitions for highly specialized domains, because some feature a higher density of specialized vocabulary than others and some are explicitly characterized as explanatory. Thus, in order to find potential sources, we sift through large lists of URLs collected during web corpus construction for DWDS corpora and look for expressions such as "lexicon" or "glossary". Heuristic guesses on the URL determine the probable home page of the given website. The retrieved URLs are then manually screened with respect to their potential; out of more than 4200 candidates, a list of 285 websites forms the basis of the present experiment. They include highly specialized lexical domains such as apiculture, astronomy, chemistry, electronics, finance, fishing, metallurgy, politics, religion, and wine-making.
The corpus has been acquired by using focused crawling techniques (Olston and Najork, 2010). This strategy involves finding all pages located at levels deeper than the starting page, which means here that in the best case all definitory material is downloaded and stored. There is also a certain amount of noise: there might be pages with a poor yield in terms of definitions, e.g. *impressum* or unrelated content. This is a difficulty common to most web corpora, which require filtering operations. Because of the diversity of the websites to crawl, it is not possible to define a precise retrieval strategy. Furthermore, not all pre-selected domains are suitable for crawling, as they are deeply ramified and feature a large number of sub-pages. As a result, the corpus of downloaded pages consists of 268 different websites with a total of 501,308 web documents and about 29 Gb of data.
Additionally, we take already existing, generic web corpora into account in order to manually assess if this specially acquired corpus effectively provides more definitory contexts than a broad search in larger and more diverse corpora.

3.4. Extraction

The extraction process also has to be generic, because of the large number of lexica it would be too cumbersome to craft a targeted extraction algorithm for each webpage. However, the pages do not offer structural patterns which could allow for a reliable boilerplate and metadata extraction (Barbaresi, 2016). These are often loosely structured and mostly provide information in the form of tables, lists or simple paragraphs. On website level, the information can be divided in two different ways: either a web page features a series of entries, for example for each letter of the alphabet, or they provide a single page for each lexical entry.
We used the vocabulary coverage (known words and todo list) of the DWDS project in order to provide lexemes to look for. Then a series of syntactic cues had to be defined. We chose a pattern-based method in order to extract the contexts in a robust manner. Most patterns are derived from the work of Cramer (2011), which is also the occasion to apply them systematically and to review them in terms of adequacy and efficiency. Our patterns first match a lexeme of our list and then look for cues left and right of the *definiendum*. We adopt the categories defined by Cramer ("strict", "less strict", "opportunistic"[9]) and evaluate them with respect to their helpfulness for definition writing. Additionally, heuristic criteria are used to determine which definitions may be directly transferable or not. For example, a "strict" pattern is a constrained syntactic structure which is expected to be a strong case for a definitory context. The patterns by Cramer (2011) are described in human language, we translate them into regular expressions, which seem powerful enough to capture the desired context. The example below integrates a number of potential variants for the pattern "under a X1, one understands X2", where X1 is a *definiendum* and X2 a catchall

[5] www.anleger-beteiligungen.de/htm/de/html/Info_Center-Glossarf74a.html
[6] www.heiligenlexikon.de/Glossar/Priester-Ordensgewaender.html
[7] www.musterkanzlei.info/1041317/portal/lexikon/-recht/d/direktionsrecht
[8] www.das-baulexikon.de/lexikon/Pelletheizung.htm
[9] In the original: *sehr genau*, *genau* and *mäßig genau*.

expression expected to contain a definition: */[Uu]nter (?:eine[mr])? $definiendum versteh(?:t|en) (?:man|wir) $catchall/*

A "less strict" pattern is a structure which features less constraints and whose output in less certain, for example "X1 is used for X2": */$definiendum verwendet man für $catchall/*

This pattern also raises issues concerning the definitory context, since the extracted sentences may not be strictly of lexicographic nature but rather entail practical advice. We believe that such patterns are still valuable, since they help determining what the lexeme is about.

An "opportunistic" pattern is a loosely constrained structure which may be a cue for a definitory context but which is also expected to be subject to noise. The most basic structure in our patterns is accordingly "a X1 is a X2", which can be translated into a pattern such as */[Ee]ine? $definiendum ist eine? $catchall/*

4. Database

We acquired web data corresponding to specialized lexical resources and extracted definitory contexts which were loaded into a database. In a first run, we could identify 191,951 contexts. 14,460 text snippets are supposed to be relevant contexts according to the extraction patterns of Cramer. In addition, we established a baseline by applying some looser, opportunistic patterns, that yielded another 177,491 hits. In the following subsection, we will describe an interface to this database and a lexicographic evaluation of the data.

4.1. Interface

We made the data available through a simple interface where users can search for a particular term and get a weighted list of resulting definitory contexts along with metadata such as origin and pattern type.

Figure 1 shows a prototype version of the interface with a wildcard search displaying results for *Trennscheibe*, where useful contexts can be returned for at least two of the senses of the word, and the adjective *trennscharf*, with several contexts to choose from. Figure 2 demonstrates the display of a single definition with an evaluation menu. In that case, the definition for *Freihandelsabkommen* (free-trade agreement), is a typical example of specialized vocabulary for which automatic definition extraction can be of great help.

4.2. Results

In order to perform a qualitative evaluation, we selected a random sample of 1000 definitory contexts containing both targeted and baseline contexts. A trained lexicographer assessed the quality of the data as follows:

0 Not a definition at all (634; including missed targets, truncated definitory contexts and doublets);

1 Provides helpful information but is not suitable for display (276);

2 Appropriate for unfiltered presentation to a user of the dictionary website, with some minor flaws or errors (86);

3 Can be directly integrated (14).

All in all, more than a third of the data proved to be helpful (classes 1 to 3), which considering the specialization degree of the lexemes is already worthwhile. In order to refine the extraction process, we performed an error analysis and listed the most common characteristics that disqualify a text snippet as a good definitory context. We found three main reasons: the *definiendum* could often not be detected correctly; there are many doublets in the data; or the text of the *definiens* is not complete due to markup issues or the definition is split into several sentences.

We assume that our extraction process has to be refined. Contrarily to the expectations raised by the patterns, regular expressions alone do not always perform well in practice on structured data, most notably on tables. The overall quality of the data can be improved with a more in-depth analysis of the structural properties of the resources found online, which involves answering questions relative to the structure of a *definiendum*. Removing doublets would also be beneficial, this issue is directly linked to the extraction which may have to be made less tolerant.

Incomplete contexts could be addressed by a surface analysis of the syntax, although the content is fragmentary on a few webpages or do not lead to full sentences due to HTML extraction; 100 contexts could be properly extracted and proved to be appropriate for presentation on the website (classes 2 and 3).

The opportunistic patterns performed just as well as the "strict" patterns so that we can afford to rely on loose constraints for the extraction. This finding can be explained by the quality of the input data: the URLs of the homepages have been screened manually before download, and the retrieved data seem to confirm that we nearly exclusively deal with explanatory contexts. Thus, we can confirm that the selection of our corpus is optimized for the task at hand since it contains a large number of definitions by its construction principle. The questions we raised are rather of lexicographic nature, concerning the detection we do not need to discriminate patterns based on their efficiency *a priori* and can afford to perform an opportunistic extraction.

Finally, we started looking for definitions in other web corpora to provide a comparison. We used free texts as input which have no relation to lexicography whatsoever. The extraction results seem to be far worse than in the lexical resources. This confirmed our intuition that it makes sense to acquire a dedicated corpus of lexical resources of all kinds and build a database of definitory contexts based on such a corpus.

Gesucht wird nach *Trenns%*
29 Treffer.

Begriff	Definition
Trennscheibe	alle 14 Tage für insgesamt eine halbe Stunde besucht werden , Kontakte sind nur mit **Trennscheibe** erlaubt . Auch die Anwälte können mit ihren Mandanten nur mit **Trennscheibe** sprechen , die Verteidigerpost wird kontrolliert . Das einzige was sich haben die Betroffenen haben zu schulden kommen lassen ist , dass sie
Trennscheibe	Die Piraten haben ihre eigene Zeitrechnung . Es gibt ein Davor und Danach , die **Trennscheibe** ist der 18 . September 2011 , als die Partei mit knapp neun Prozent in das Berliner Abgeordnetenhaus einzog . Seit dem Triumph im Herbst ist wenig geblieben wie es war Parteivertreter werden plötzlich für
Trennscheibe	Lokalsenders WCNC im Gefängnis ein Interview . In der orangefarbenen Häftlingskleidung saß er hinter einer **Trennscheibe** und sagte "Es ist das erste Mal , dass ich in Konflikt mit dem Gesetz gekommen bin." Er habe niemanden erschrecken wollen , sagte Verone der " Gaston Gazette". Falls er der Bankangestellten einen Schrecken
Trennscheibe	auf dem Grundstück eines Verdächtigen im Mordfall Julia gefunden hat Als Sven vor die verspiegelte **Trennscheibe** tritt , ist es stockdunkel im Zimmer . Die sechs jungen Männer im benachbarten Gegenüberstellungsraum können ihn nicht sehen . Sie werden eines Verbrechens verdächtigt und halten jeder ein Schild mit einer Nummer in den
Trennscheibe	. SPIEGEL TV Keine Seltenheit Psychische Ausnahmezustände bei den Tätern Als Sven vor die verspiegelte **Trennscheibe** tritt , ist es stockdunkel im Zimmer . Die sechs jungen Männer im benachbarten Gegenüberstellungsraum können ihn nicht sehen . Sie werden eines Verbrechens verdächtigt und halten jeder ein Schild mit einer Nummer in den
Trennscheibe	kann der Maybach 62 das Modell mit 6 , 17 Meter Länge mit einer versenkbaren **Trennscheibe** bestückt werden . Auf Knopfdruck verwandelt sich das elektrotransparente Glas darüber hinaus in eine blickdichte Wand . Über Preise wird , man ahnt es schon , in diesen Autosphären nur höchst ungern gesprochen . Das
Trennscheibe	ertragen nicht länger , wie der 60-Jährige gefesselt in einen Raum geführt und an eine **Trennscheibe** gesetzt wird . Wegen der Handschellen kann er den Telefonhörer nicht selbst halten , zuletzt half ihm ein Überwachungsbeamter. "Herr M . brach in dieser für ihn sehr emotionalen weil menschenunwürdigen Situation in Tränen aus
trennscharf	Geschlechter geeignet sind Die Bezeichnungen Tanga , String , Stringtanga und Thong werden häufig nicht **trennscharf** verwendet , da sie einander kaum ausschließen . Unterscheidungsmerkmale Name Seitenteile Steiß Schritt Tanga immer Schnur , ggf . gebunden meist Dreieck oft mehr als Schnur String oft Schnur , aber beliebig beliebig immer Schnur
trennscharf	Fazit . Wie schon die Unterscheidung zwischen ? Spekulation ? und ? Anlage ? wenig **trennscharf** ist und für jeden Investor von individuellen Risikopräferenzen und -empfindungen abhängt , so lassen sich auch die Discount-Optionsscheine nicht eindeutig ? etwa anhand von Kennzahlen o . ä. ? kategorisieren . Die dargestellte Strategie ist
trennscharf	stark pragmatischen Charakter auf und beruht auf intensiven Praxiserfahrungen . Daher sind die Begriffsabgrenzungen nicht **trennscharf** , und die Systematisierung ist interpretationsfähig . Auch wenn als einziges Ziel die Kostensenkung bei gleichbleibender Leistung des Produkts genannt wird , so zeigen die Ausführungen und die Resultate der Praxis , dass die Wertanalyse
trennscharf	. neue Informationen zu einer anderen Einschätzung führen . Die beiden Änderungssachverhalte sind mitunter nicht **trennscharf** . So kann gem . APB 20 . 11 eine geänderte Bilanzierungsmethode die Änderung der Zukunftsschätzung nach sich ziehen . Die Bewertungsmethodenstetigkeit kann durch beide genannten Änderungen betroffen sein . Grundsätzlich sind Bewertungsmethodenänderungen unter Berücksichtigung

Figure 1: Search interface, results of the query *trenns**

Metadaten

Terminus	Qualität	Quelle	ID
Freihandelsabkommen	Cramer	www.eufis.eu/eu-glossar.html?&type=0&uid=136&cHash=8598503ea97b2ac4457f551f54016b54&print=1.html	1400083

Definition

Definition	Bewertung
Ein Freihandelsabkommen ist ein Handelsabkommen , das die Zölle zwischen den Verhandlungspartnern vollständig beseitigt und mengenmäßige Beschränkungen von Handelsprodukten untersagt.	○ Gut ○ Schlecht

Oder

Bearbeiten

☐ Nie wieder Kandidaten aus *www.eufis.eu*

Senden

Figure 2: Definition window with evaluation options

5. Conclusions

We introduced work on the detection of definitory contexts which is meant to support lexicographers in writing definitions for a dictionary of contemporary German. We described a specialized web corpus built for this task as well as robust pattern-based extraction processes on bare HTML documents. The resulting database entails about 191,000 candidates taken from about half a million webpages. The corresponding interface displays information for a large range of lexical units which can then be assessed by lexicographers.

Altogether, the definitory contexts in the database can be really helpful in alleviating two key tasks of the lexicographical workflow, firstly because it speeds up the search for the exact meaning(s) of terms – in the present case highly specialized terms – and secondly because it provides usable input for the task of formulating an appropriate meaning paraphrase.

Our method yielded some useful results but can benefit from some improvements, mostly in structural analysis and selection. Given the type of web texts we use, a qualitative review of the extraction patterns does not seem to be very relevant, less constraints lead to more potential definitions, so that it can be said in our case that "looser is better". Beyond an opportunistic setting, future challenges reside in finding the right balance between generic approaches and in-depth analysis.

6. Bibliographical References

Anke, L. E. (2013). Towards Definition Extraction Using Conditional Random Fields. In *Proceedings of the Student Research Workshop associated with RANLP*, pages 63–70.

Barbaresi, A. (2015). *Ad hoc and general-purpose corpus construction from web sources*. Ph.D. thesis, École Normale Supérieure de Lyon.

Barbaresi, A. (2016). Efficient construction of metadata-enhanced web corpora. In Paul Cook, et al., editors, *Proceedings of the 10th Web as Corpus Workshop*, pages 7–16. Association for Computational Linguistics.

Borg, C. (2009). *Automatic definition extraction using evolutionary algorithms*. Ph.D. thesis, University of Malta.

Cramer, I. (2011). *Definitionen in Wörterbuch und Text*. Ph.D. thesis, TU Dortmund.

Geyken, A., Barbaresi, A., Didakowski, J., Jurish, B., Wiegand, F., and Lemnitzer, L. (2017a). Die Korpusplattform des "Digitalen Wörterbuchs der deutschen Sprache" (DWDS). *Zeitschrift für germanistische Linguistik*, 45(2):327–344.

Geyken, A., Wiegand, F., and Würzner, K.-M. (2017b). On-the-fly Generation of Dictionary Articles for the DWDS Website. In *Proceedings of the eLex 2017 Conference*, pages 560–570.

Hearst, M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of COLING*, volume 2, pages 539–545. Association for Computational Linguistics.

Klein, W. and Geyken, A. (2010). Das digitale wörterbuch der deutschen sprache (dwds). In *Lexicographica*, pages 79–96. De Gruyter.

Klein, W. (2013). Von Reichtum und Armut des deutschen Wortschatzes. In *Reichtum und Armut der deutschen Sprache. Erster Bericht zur Lage der deutschen Sprache*, pages 15–56. De Gruyter.

Kovář, V., Močiariková, M., and Rychlý, P. (2016). Finding Definitions in Large Corpora with Sketch Engine. In *Proceedings of LREC*, pages 391–394. ELRA.

Navigli, R. and Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327. Association for Computational Linguistics.

Navigli, R., Velardi, P., and Ruiz-Martínez, J. M. (2010). An Annotated Dataset for Extracting Definitions and Hypernyms from the Web. In *Proceedings of LREC*, pages 3716–3722. ELRA.

Olston, C. and Najork, M. (2010). Web Crawling. *Foundations and Trends in Information Retrieval*, 4(3):175–246.

Przepiórkowski, A., Degórski, Ł., Wójtowicz, B., Spousta, M., Kuboň, V., Simov, K., Osenova, P., and Lemnitzer, L. (2007). Towards the automatic extraction of definitions in Slavic. In *Proceedings of the 6th Workshop on Balto-Slavonic Natural Language Processing: Information Extraction and Enabling Technologies*, pages 43–50. Association for Computational Linguistics.

Schlücker, B. (2012). Die deutsche Kompositionsfreudigkeit. Übersicht und Einführung. In Livio Gaeta et al., editors, *Deutsche als kompositionsfreudige Sprache. Strukturelle Eigenschaften und systembezogene Aspekte*, pages 1–25. de Gruyter.

Schumann, A.-K. (2014). *Linguistische Analyse und korpusbasierte Extraktion deutscher und russischer wissenshaltiger Kontexte*. Ph.D. thesis, Universität Wien.

Seitner, J., Bizer, C., Eckert, K., Faralli, S., Meusel, R., Paulheim, H., and Ponzetto, S. P. (2016). A Large Database of Hypernymy Relations Extracted from the Web. In *Proceedings of LREC*, pages 360–367. ELRA.

Storrer, A. and Wellinghoff, S. (2006). Automated detection and annotation of term definitions in German text corpora. In *Proceedings of LREC*, pages 2373–2376. ELRA.

Westerhout, E. (2010). *Definition extraction for glossary creation: A study on extracting definitions for semi-automatic glossary creation in Dutch*. Ph.D. thesis, University of Utrecht.

Wiegand, H. E. (1989). Die lexikographische Definition im allgemeinen einsprachigen Wörterbuch (The lexicographic definition in the general monolingual dictionary. In F.J. Hausmann, et al., editors, *Wörterbücher. Dictionaries. Dictionnaires*, volume 5.1 of *Handbücher zur Sprach- und Kommunikationswissenschaft*, pages 530–588. De Gruyter.

Zhang, J., Wang, Y., and Yang, D. (2014). Automatic learning common definitional patterns from multi-domain Wikipedia pages. In *Data Mining Workshop (ICDMW), 2014 IEEE International Conference on*, pages 251–258. IEEE.

Annotating Abstract Meaning Representations for Spanish

Noelia Migueles-Abraira, Rodrigo Agerri, Arantza Diaz de Ilarraza
IXA NLP Group, University of the Basque Country UPV/EHU
noeliamigueles@gmail.com,{rodrigo.agerri,a.diazdeilarraza}@ehu.eus

Abstract

Until recently, semantic annotations for different semantic phenomena were independent and unconnected. The Abstract Meaning Representation (AMR) project arised out of the need to create a broad-coverage semantic bank containing a unified set of semantic information represented in simple, single-rooted, easy-to-read structures. Because the semantic representation language proposed in AMR is biased towards English, annotating AMR structures for other languages, such as Spanish, is not a trivial task. In this paper we propose a linguistic method that we believe would help lay the groundwork for building a large semantic bank for Spanish and would guide those who would like to implement it for other languages. Thus, we analyze a broad spectrum of Spanish linguistic phenomena to come up with suggestions to adapt the current guidelines so that it is possible to annotate AMRs for Spanish. As a result of this work, we make available the first public online repository containing manually annotated Spanish AMRs.

Keywords: AMR, Sembank, Semantics, Spanish Language Resources

1. Introduction

The construction of large manually annotated corpora has been crucial in order to advance Natural Language Processing (NLP) technology. Among others, the construction of the Penn Treebank (Marcus et al., 1993) made possible the existence of many statistical parsers trained on syntactic treebanks which perform at an accuracy of about 90% (Charniak and Johnson, 2005). In contrast, computational semantics is lagging behind in this respect, as most of the semantic annotated resources for English are rather scattered or simply nonexistent for the large majority of languages.

This lack of a unified sembank of natural language sentences paired with their sentential, logical meanings is what led to the appearance of Abstract Meaning Representation (AMR), a semantic representation language introduced by (Banarescu et al., 2013). This approach promotes the representation of the logical meaning of sentences as single rooted, directed graphs (or AMRs) with labelled nodes (concepts) and edges between them (relations). These incorporate semantic roles, among other linguistic phenomena. In a propositional-style logic, AMR is able to capture who is doing what to whom in a sentence. Furthermore, AMR tries to abstract away from both morphological and syntactic idiosyncrasies that account for several cross-lingual differences. However, there is still a problem with this approach, as it is strongly biased (by design) to annotate English sentences. To overcome this fundamental limitation, we decided to explore the possibility of annotating AMRs for Spanish texts.

Thus, the central goal of this work is to investigate how to create Spanish AMRs in order to build a sizable Spanish sembank. We address this objective by answering the following questions:

1. Is it possible to follow the current guidelines to annotate Spanish AMRs? And if not, how can the guidelines be refined in order to annotate Spanish AMRs? Also, what resources do we need to carry out such task?
2. How similar are English and Spanish AMRs?
3. What can be learned from the gathered information for future annotation efforts?

The result of this study has made it possible to manually annotate, for the first time, a Spanish corpus with AMR structures. Furthermore, we believe that our study will help laying the groundwork for building a large semantic bank for Spanish as well as a reference for other languages.

LOGIC format:

∃ w, b, g:
instance(w, want-01) ∧ instance(g, go-01) ∧
instance(b, boy) ∧ arg0(w, b) ∧
arg1(w, g) ∧ arg0(g, b)

AMR format (based on PENMAN):

```
(w / want-01
  :arg0 (b / boy)
  :arg1 (g / go-01
          :arg0 b))
```

GRAPH format:

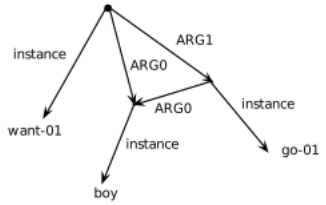


Figure 1: Equivalent formats for representating the meaning of "The boy wants to go" according to Banarescu et al. (2013).

Prior to this study, others had worked on the adaptation of AMR to different languages different to English (Uresova et al., 2014; Li et al., 2016; Xue et al., 2014) . However, for the time being only a small bank of Chinese sentences paired with their semantic representations is available to the public[1].

[1]The Chinese Abstract Meaning Representation (CAMR)

2. Background

AMR as a semantic representation language appears based on the assumption that we lack a simple readable semantic bank of natural language sentences "paired with their whole-sentence, logical meanings" (Banarescu et al., 2013). Thus, they started annotating the logical meaning of sentences, for which a particular sentence is encoded as a Directed Acyclic Graph (DAG). The ultimate goal is to encourage advances in different NLP tasks, including Statistical Natural Language Understanding (SNLU) or Statistical Machine Translation (SMT), amongst others. The idea is to do this by enabling rapid human annotation of broad coverage corpora. Some phenomena that AMR addresses include discourse connectives, semantic roles, intrasentential coreference, named entities (and wikification), questions, negation, and modality. AMR features a three-way annotation format: a logic-based format (Davidson, 1980), an AMR format, and a graph format. The three are equivalent. An example can be seen in Figure 1.

AMR strives to capture many aspects of meaning in a single simple data structure. To do that, it tries to abstract away from morpho-syntactic idiosyncrasies. AMR uses PropBank (PB) framesets (Kingsbury and Palmer, 2002). Thus, each frame presents annotators with a list of senses which contains its own number of core arguments. AMR defines around a hundred semantic relations. These relations consist of core ":ARGX" roles (frame arguments), non-core roles (general semantic relations), roles for quantities, for date-entities, operators such as ":opX" and ":prep-X", multi-sentence roles, and a conjunction role. Simple roles often correspond to a reified concept. AMR does not dictate a mandatory way of applying rules. Instead, it promotes personal interpretation of researchers about how strings are related to meanings. For specific details about AMR annotation guidelines, please read Banarescu et al. (2013)[2].

3. Methodology

Our methodology aims to determine whether if it is possible to annotate Spanish meaning representations following the current Abstract Meaning Representation (AMR) guidelines. The key is to detect aspects of meaning in Spanish sentences that cannot be represented when applying the current AMR guidelines. After the identification of such relevant linguistic aspects, we propose a new set of guidelines so that the annotation of these within the AMR framework will be possible. During the preliminary stage of this phase, we manually created a sample of Spanish AMRs according to the original AMR annotation scheme. Next, we identified any missing aspects of meaning that AMR failed to represent. Finally, we designed an extension of the annotation scheme to potentially meet those needs. This refinement in the AMR annotation standard stays true to the AMR syntax.

To carry out this analysis, we decided to use The Little Prince Corpus[3]. The corpus contains 1652 sentences represented in AMR structures for both English and Chinese. As the AMR project points out, using this freely available corpus would allow researchers to compare different representations of the same text. The selection of these sentences was not a random process. First, we translated about 250 sentences from The Little Prince. Then, we made a comparison between the resulting pairs of English-Spanish sentences. Out of those, we narrowed down the number to 50, resulting in a selection of sentences that includes a wide range of linguistic phenomena, including nominal ellipses, clitic pronouns, gender, verbal periphrases and locutions, double negatives, nominalization and verbalization, affixes, and some key words that have a special treatment in AMR. To choose these, we also paid attention to the level of structural cross-lingual variation that the sentence pairs exhibit, just so we could study both language pairs whose structures align well and those whose structures do not align well. Declarative, interrogative, exclamatory, and imperative sentences are covered. For our purpose, we consider that this number of sentences, in spite of being small, is good enough to detect a reasonable amount of Spanish-specific constructions that the current English-only version of AMR is not able to represent.

The following section details the main Spanish linguistic phenomena identified in this study and the solutions we propose in order to provide an AMR representation addressing them.

4. Annotation Examples

We have identified seven main Spanish linguistic phenomena that cannot be represented in AMR under the current guidelines: NP ellipses, third person possessive pronouns, third person clitic pronouns, the usage of *se*, gender, certain verbal constructions such as verbal periphrases and locutions, and double negatives. We therefore decide to extend the AMR annotation guidelines to be able to annotate such cases (Migueles Abraira, 2017). In this paper we discuss the first 4 out of these 7 phenomena. Before we start, though, it is important to explain why we chose to adapt the current guidelines by converting some roles, reifications, modals, and special words into their Spanish counterparts. This need arises from the fact that sometimes certain roles and relations which receive a special treatment in AMR lead to confusion if we try to use them as such in Spanish. For instance, many roles like ":prep-X" present a problem. Not only because the annotator may not know to which such role is referring to but also because the equivalences between Spanish and English prepositions might not be straightforward, one-to-one correspondences. For example, the English equivalents for the Spanish preposition *en* could be one of the following: "in", "inside", "into", "within", or "by". In such case, an annotator would have to figure out which preposition should be used, and how, because not all prepositions are annotated with the role ":prep-X.". To solve this and other issues, we propose the conversion of roles, reifications, modals, and special words from English to Spanish. Hence, from now onwards, we

Bank is available at:http://www.cs.brandeis.edu/~clp/camr/camr.html

[2]Also, the more detailed annotation guidelines are available here: https://github.com/amrisi/amr-guidelines/blob/master/amr.md

[3]https://amr.isi.edu/download.html

will apply such conversions (see (Migueles Abraira, 2017) for more details).

4.1. NP Ellipses

Because of the nature of Spanish grammar, nominal ellipsis in Spanish is pervasive. In most cases, we will find either a conjugated verb and/or a clitic pronoun that indicates person. So, technically, there is no need to add a Noun Phrase (NP) unless we need to clarify the subject.

(1) Carla tiene prisa. Tiene cita con el dentista.
Carla is in a hurry. (She) has an appointment with the dentist.

If we look at the second sentence on its own, we know that we are talking about a single female entity in the English version. In Spanish, however, we do not. While annotating the English version, one would add "she" rather than "he" or "it" as the entity that has an appointment with the dentist. Unfortunately, in the Spanish version, the only information that we have about such entity is that it refers to a second-person singular entity. There is a reference to an entity, even if we do not know which entity due to a lack of content (remember that AMR treats the semantic meaning of single sentences). The same would happen with the plural pronoun "they" - as it would be addressed in the sentence. However, the Spanish version *Tienen cita con el dentista* could imply that the entity is either the plural form *ellos* (masculine or generic masculine) or the plural form *ellas* (feminine). The ellipsis in this case has left the gender of the agent role underspecified. If we represent such sentence following AMR guidelines, we do not explicitly state that there is such underspecification.

```
(t / tener
    :ARG1 (c/cita
    :prep-con (d / dentista)))
```

Thus, whenever there is a nominal ellipsis, we propose to use a concept *ente* ("being") that is mapped to a non-core role *:sinnombre* (":nameless") and followed by a concept of the same name. This decision is based on the idea that not including an entity that performs an action in the annotation, when the sentence evidently states that there is one, would lead to an inaccurate semantic meaning representation. If there is a referenced or, at least partially referenced, entity, the logical thing to do would be to add such entity to the resulting AMR. Below there is an example of the proposed annotation of a third person nominal ellipsis following this new rule.

```
(t / tener
    :ARG0 (e / ente
        :sinnombre (s / sinnombre))
    :ARG1 (c / cita
        :prep-con (d / dentista)))
```

As we can see in this example, the concept *ente* would be related to the root node *tener*. In the English AMR, this would actually correspond to the concept "she."

```
(h / have-03
    :ARG0 (s / she)
    :ARG1 (a / appointment-02
        :ARG0 s
        :ARG1 (d/ dentist)))
```

4.2. Third Person Possessive Pronouns

When it comes to annotate ellipses in possessive NPs, another issue arises. This has to do with the third person singular and plural possessive pronouns *su* and *sus*. The problem, once again, is not knowing much about the possessor. For instance, *su* could be translated as "his", "her", "its", "they," or even "your" (formal language). The difference here, unlike in the previous phenomenon, is that the possessive entity is actually present in the sentence. AMR currently marks possession using a *:posee* operator following by the concept (his, her, its, etc.). Given that for Spanish this is underspecified, the argument of the operator *:posee* should be *su* but we need to explicitly state that its meaning is not fully specified. Again, bear in mind that to know the exact entity to which any of these pronouns is referring to, one needs to have access to contextual information. Because AMR only looks at individual sentences, we are not sure who the possessor is. Then, rather than implying that the possessor is "he", "she", "they," or "you" without really knowing if that is the case, whenever there is ellipsis in possessive NPs, we also annotate an entity with the concept *ente*. This, however, is tagged with the non-core role *:sinespecificar* (":unspecified"). The latter is followed by the possessive pronoun in singular form, as it is shown in the following AMR structure:

(2) Su casa es grande.
(His/her/its/your) house is big.

```
(g / grande
    :campo (c / casa
        :posee (e / ente
            :sinespecificar (s / su))))
```

4.3. Third Person Clitic Pronouns

A similar complication occurs when attempting to annotate third person clitic pronouns. Consider the following example in which *lo*, as a clitic pronoun, fails to provide much information of the entity that it refers to because it is semantically underspecified. As far as we know, it could be equally associated to an animate or inanimate being. For instance, we could break down the word *mándaselo* into three components: *manda* + *se* + *lo*, where *manda* is the second person singular of the verb *mandar* ("to send"), *se* is an enclitic pronoun that means "to (him/her/it)" and *lo* is another enclitic pronoun that refers to an underspecified entity in third person singular.

(3) Mándaselo ahora.
Sent (it/him) to (him/her/it) now.

Once again, following the idea that ignoring a present entity would lead to an inaccurate semantic meaning representation, we generally annotate enclitic and proclitic pronouns

Type of *se*	Description	Examples
Se as an indirect object	Substitutes le or les when they appear together with direct object pronouns	*Se lo regala* ("(S)he/they gave it to (someone)")
Reflexive *se*	The subject and the object have the same referent	*Se ducha* ("(S)he showers")
Reciprocal *se*	Expresses mutual action or relationship	*Se miran* ("They look at each other")
Se as a component of a pronominal verb	Acts as a morpheme and has no syntactic function	*Se quejan* ("They complain")
Intransitive *se* with a voluntary subject	Makes the verb intransitive; the subject performs the action	*Se traslada* ("(S)he relocates")
Intransitive *se* with an involuntary subject	Makes the verb intransitive; the subject experiences or receives the action	*Se alegraron* ("They rejoiced")
Emphatic *se*	Intensifies the meaning of a verb	*Se bebió una botella de vino* ("(S)he drank a bottle of wine")
Accidental *se*	Corresponds to non-intentional actions	*Se les derrama el café* ("They spill the coffee over")
Aspectual *se*	Adds aspectual meaning	*Se duerme* ("(S)he falls asleep")
Passive *se*	Places the influence on the action rather than on the doer/actor	*Se diseñó la catedral* ("The cathedral was designed")
Impersonal *se*	Makes no specific reference (refers to people in general)	*En este restaurante se come bien* ("You eat well in this restaurant")

Figure 2: *Se* usage with examples according to Lozano (2005)

in the same way that we annotate third-person possessive NPs to cover this information. Consider the next annotation.

```
(m / mandar
    :modo imperativo
    :ARG0 (t / tu)
    :ARG1 (e / ente
        :sinespecificar (l / lo))
    :ARG2 (e2 / ente
        :sinespecificar (s / se))
    :tiempo (a / ahora))
```

Constructions of this kind tend to be quite straightforward in English (see example below). But in Spanish, the semantic meaning may not be that clear without taking previous sentences into consideration. To not come up with either an incorrect or too ambiguous annotation, we consider the use of this relation to help the system. For instance, while translating AMRs to plain text.

```
(s / send-01
    :ARG0 (y / you)
    :ARG1 (i / it)
    :ARG2 (h / he)
    :time (n / now))
```

4.4. *Se* Usage

With at least eleven uses (Figure 2), *se* is quite possibly the most versatile pronoun in the Spanish language ((Lozano, 2017);(Millán, 1990); (Haywood et al., 2008)). In most cases, the problem that we encounter is not knowing who or what is the direct or the indirect object. As seen in the previous example *mándaselo ahora*, we simply do not know to whom (or what) it needs to be sent. Only context could tell us that but AMR currently does not focus on the analysis of interconnected sentences. At this point, we are left with no other choice but to include something to cover semantic underspecification. To do that, we follow the aforementioned clitic annotation procedure. Semantic underspecification is also the norm for the rest of the uses of *se*. However, the solutions provided to represent each use in AMR vary. Thus, when *se* is reflexive, we use reentrancy, as we can see in the AMRs below:

(4) Se gusta.
(She/he/it) likes (her/him/it)self.

```
(g / gustar
    :ARG0 (e  / ente
        :sinespecificar (s / se))
    :ARG1 e)
```

The criterion to annotate reciprocal *se* is slightly different. We also use reentrancy, but we add a concept named *se-recíproco* (*reciprocal-se*).

(5) Se gustan.
They like each other.

```
(g / gustar
    :ARG0 (e / ente
        :sinespecificar (s / se-reciproco))
    :ARG1 e)
```

Note the difference with the English equivalent:

```
(l / like-01
    :ARG1 (e / each
        :mod (o / other)))
```

If we consider impersonal and passive usages of *se*, a different issue applies. Once again, we do not know who or what performs a given action. However, in these cases there is no subject that is explicitly stated in the sentence.

(6) Se vende casa rural.
Country house on sale.

```
(v / vender
    :ARG1 (c / casa
    :mod (r / rural)))
```

5. Concluding Remarks

This research project is the first step towards the construction of a sizable sembank of Spanish sentences paired with their sentential, logical meanings. Our main goal was to study how to create Spanish Abstract Meaning Representations (AMRs).

As we have seen, the current guidelines, as they are, fail to represent Spanish semantic representations fully. This is no surprise, since it is clearly stated in the guidelines that AMR is not an interlingua. On a positive note, we have

demonstrated that it is, in fact, possible to adjust the guidelines accordingly to cover their lack of certain meaning aspects in Spanish that cannot be ignored. To annotate Spanish AMRs, we need the guidelines to be adapted for this task, and we need an editor that is connected to a refined and updated version of AncoraNet (Taulé et al., 2008).

Although a substantial amount of work remains to be done, the information that we have obtained serves as the foundation for future work. Because of this study, we now know what is needed to take the next step in this ongoing effort to build a Spanish semantic bank. In short, based on our work, we know how to cope with linguistic phenomena that did not have a way to be represented before. And, what is more, based on the limitations faced during the annotation process, we know the resources that are needed to achieve this goal.

For future work, we believe that a refinement and update of the AnCora corpus is crucial. The mappings of senses need to be more accurate but they also need to be updated so that can be connected to an up-to-date version of the PropBank's inventory. At the same time, we think nominal and adjectival relations should be included as well. Also, the development of a tool for Spanish AMR annotation would greatly help annotators. Furthermore, including wikification would also be interesting in order to avoid certain differences in the annotation of referents. Finally, we think it would be interesting to annotate the full The Little Prince with Spanish AMRs in order to compare the resulting AMRs to the English and Chinese versions.

The final 50 Spanish sentences with their AMR annotations can be found online[4], together with their English equivalents.

6. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Charniak, E. and Johnson, M. (2005). Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 173–180.

Davidson, D. (1980). *Essays on Actions and Events*. Oxford University Press, Oxford.

Haywood, L., Thompson, M., and Hervey, S. (2008). *Thinking Spanish Translation: A Course in Translation Method, Spanish to English*. Thinking Translation. Routledge.

Kingsbury, P. and Palmer, M. (2002). From treebank to PropBank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, Las Palmas, Spain, May.

Li, B., Wen, Y., Bu, L., Qu, W., and Xue, N. (2016). Annotating the Little Prince with Chinese AMRs. In *Proceedings of LAW X – The 10th Linguistic Annotation Workshop*, pages 7–15, Berlin, Germany. Association for Computational Linguistics.

Lozano, L. (2017). Los diferentes tipos de "se" en ele. *Red ELE, 3, retrieved from https://sede.educacion.gob.es/publiventa/d/20267/19/0*.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotation corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330.

Migueles Abraira, N. (2017). A study towards spanish abstract meaning representation. *Master Thesis, University of the Basque Country UPV/EHU*.

Millán, A. (1990). Categorías, funciones y valores del español: Proyección didáctica. *Revista de Filología y su Didáctica, 13*, pages 161–201.

Taulé, M., Martí, M. A., and Recasens, M. (2008). AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of LREC 2008*, pages 96–101.

Uresova, Z., Hajic, J., and Bojar, O. (2014). Comparing Czech and English AMRs. In *Proceedings of the Workshop on Lexical and Grammatical Resources for Language Processing*, pages 55–64, Dublin, Ireland. Association for Computational Linguistics.

Xue, N., Bojar, O., Hajic, J., Palmer, M., Uresova, Z., and Zhang, X. (2014). Not an Interlingua, But Close: Comparison of English AMRs to Chinese and Czech. In *In Proceedings of the 9th International Conference on Language Resources and Evaluation*, pages 1765–1772, Reykjavík, Iceland. European Language Resources Association.

[4] http://ixa.eus/node/8916

Browsing the Terminological Structure of a Specialized Domain: A Method Based on Lexical Functions and their Classification

Marie-Claude L'Homme, Benoît Robichaud, Nathalie Prévil
Observatoire de linguistique Sens-Texte (OLST)
Université de Montréal
C.P. 6128, succ. Centre-ville
Montréal (Québec) H3C 3J7 CANADA
{mc.lhomme, benoit.robichaud, nathalie.previl}@umontreal.ca

Abstract

This paper describes a method for browsing relations between terms and unveiling the terminological structure of a specialized domain. The method consists in expanding a graph that takes as input the relations encoded in a multilingual terminological resource called the *DiCoEnviro* that contains terms in the field of the environment. In the DiCoEnviro, terminological relations are encoded using lexical functions (Melčuk et al. 1995) and further classified in families defined on the basis of the properties of relations. We seek to provide users with an explicit and intuitive representation of a wide variety of relations. We also make the most of the richness of the encoding, while implementing some graphical choices to make their interpretation as clear as possible for end users. The method is implemented in a tool called *NeoVisual* that provides access to more than 11,000 relations in English and 15,000 relations in French. Portuguese is also included and coverage in all languages will increase as new entries are added to the DiCoEnviro.

Keywords: terminological relation; paradigmatic relation; syntagmatic relation; graphical representation; lexical function

1. Introduction

In terminology, it is assumed that concepts in specialized domains and terms used to express them are part of a structure. Concepts or terms are defined according to the place they have in this structure and the relations they hold with others. This principle is taken for granted (at least partially) when describing terminological data and some resources represent this structure explicitly (thesauri, terminological knowledge bases, ontologies).[1]

In this paper we describe a method for browsing relations between terms and gradually unveiling the terminological structure of a specialized domain. The method is based on the contents of a terminological resource called *DiCoEnviro* that contains terms linked to the domain of the environment. This specific work builds on three sources (that are further presented in Section 2):

- Terminological relations manually encoded in the DiCoEnviro with lexical functions (Mel'čuk et al. 1995);
- A method devised by Robichaud (2012) developed on the basis of relations encoded in another term base very similar in structure to the DiCoEnviro, i.e. the DiCoInfo (on computing and the Internet);
- A classification of terminological relations and natural language explanations superimposed on LFs to facilitate their consultation in a user-friendly version of our resources (L'Homme et al. 2012).

The method consists in expanding a graph that takes the relations from the DiCoEnviro as input and allow users to explore different parts of the terminological structure in which they are interested. Our method is targeted at end users of terminological resources. So we seek to provide them with an explicit and intuitive representation of a wide variety of relations. We want to make the most of the richness of the encoding of terminological relations in the DiCoEnviro, while implementing some graphical choices to make their interpretation as clear as possible for users. The graphical display should complement textual (dictionary-like) information and not replace it altogether. It is assumed that from the point of view of users a graph is a more suitable tool to visualize and navigate through the terminological structure of a domain especially since some of its features can be exploited by designers to highlight different properties of relations and the nodes that these relations link. However, we also believe that other forms of terminological information (definitions, annotated examples) are more easily obtained in textual format.

The paper is organized as follows. Section 2 provides some details on how terminological relations are encoded in the DiCoEnviro and previous choices that were made to facilitate their interpretation. Section 3 presents other resources that resort to graphs to display relations between lexical units, terms or concepts. Section 4 describes our method and the tool in which is it implemented. Finally, Section 5 draws some conclusions and mentions a few areas that we wish to explore in the future.

2. Relations and their encoding in the DiCoEnviro

The DiCoEnviro is an online terminological resource under construction that contains terms in different languages, i.e. English, French, Spanish and Portuguese. The resource differs from most terminological repositories in the sense that it encodes and describes terms viewed as lexical units rather than labels for concepts. In addition to relations commonly taken into consideration, such as hypernymy,

[1] However, it is worth pointing out that the most widely used terminological resources, i.e. term banks, do not represent terminological structure explicitly.

meronymy and exact synonymy, the DiCoEnviro describes a large variety of other relations, including paradigmatic and syntagmatic ones. Table 1 gives a short list of relations that appear in the resource along with examples for each.

Relations are manually encoded by terminologists using lexical functions, LFs (Mel'čuk et al. 1995; Polguère 2014). LFs can represent both paradigmatic and syntagmatic relations and can take into account up to three different linguistic properties: the syntactic structure of a collocation, the general and abstract meaning of a relation and, finally, the relation between related term and the argument structure of the keyword. Technically, an LF applies to a key word and yields one or a short list of values. Examples of LFs are given in Table 1.

Relation	**Example(s)**	**LF**
Same meaning		
Exact synonymy, variants, symbols	*carbon dioxide* → *carbonic acid gas* *carbon dioxide* → *CO_2*	Syn Symb
Related meaning		
Near synonymy	*agriculture* → *farming*	QSyn
Generic	*carbon dioxide* → *gas*	Gener
Opposites		
Antonymy	*sustainable* → *unsustainable*	Anti1, Anti2, Rev1, Rev2
Contrastiveness	*fauna* → *flora*	Contr
Conversiveness	*propel* → *run*	$Conv_{ij}$
Word families		
Same meaning, different POS	*abundant* → *abundance* *warm* → *warming*	A_0, S_0, V_0, Adv_0
Adjective with added meaning	*erode* → *eroding* *erode* → *erodible*	A_1, A_2, $Able_1$, $Able_2$, etc.
Linguistic realizations of arguments		
Role label (e.g. Agent, Patient)	*warm* → (Patient) *climate, atmosphere, temperature, ocean*	Encoded in lists with role labels
Types of		
Intensification	*toxicity* → *high* ~	Magn
According to a location	*habitat* → *terrestrial* ~	Hypo – Lieu
Combinations		
Typical use	*habitat* → *inhabit in a* ~	$Real_i$, $Fact_i$, $Labreal_{12}$
Existence	*species* → ~ *survives*	$Func_i$
Creation	*territory* → *establish a* ~	$Caus_iFunc_0$
Others		
Meronymy	*Earth* → *continent*	[Part], [Tot]. Mult, Sing
Quantity	*greenhouse gas* → *concentration of* ~	Quant

Table 1: Examples of terminological relations in the DiCoEnviro

From the point of view of encoding, LFs have several advantages. First, as was mentioned above, they take into account different properties of relations (syntactic, semantic and argument structure) and thus allow us to classify related terms accordingly. Furthermore, they are language-independent. Hence relations in different languages that have the same meaning are encoded with the same LF.

However, from the point of view of their presentation in resources, they raise some challenges. In the online version of the DiCoEnviro, terminological relations appear in an ordered list that can be quite extensive in some entries. Going through this list can soon become cumbersome. In previous work, we investigated different ways to alleviate the exploration of relations.

A first strategy – that was implemented early on in the resource – consists of grouping sets of relations in *families* based on their properties (paradigmatic vs. syntagmatic, related meaning vs. opposites, etc.). This is reflected in the way that relations are presented in Table 1: "Related meanings" and "Opposites" correspond to these families.[2]

Then, since LFs are not well known and can be rather difficult to decipher, a second strategy consists of adding a natural language explanation and superimpose it on each LF (L'Homme et al. 2012, based on Mel'čuk and Polguère 2007). For instance, the LF $Real_i$ is often explained as follows: *The … uses a* ~ (here the '~' stands for the key word and '…' stands for the argument that involved in this collocation). While LFs are language-independent, natural language explanations must be adapted to each language taken into consideration in the resource.

The strategies that we mentioned in the previous paragraphs are presented textually to the users of our resources. An alternative method was developed by Robichaud (2012) to show part of the relations graphically and implemented in a tool called *DiCoInfo Visuel*. The tool displays relations included in the families "Same meaning" and "Related meaning" (distributed slightly differently in the *Visuel*), "Opposites", "Word families" in a graph. It also shows the typical arguments of predicative terms along with a label that corresponds to their semantic roles. Figure 1 shows how relations for the two meanings of the verb *erode* are displayed (yellow arrows are used for arguments; purple ones for word families).

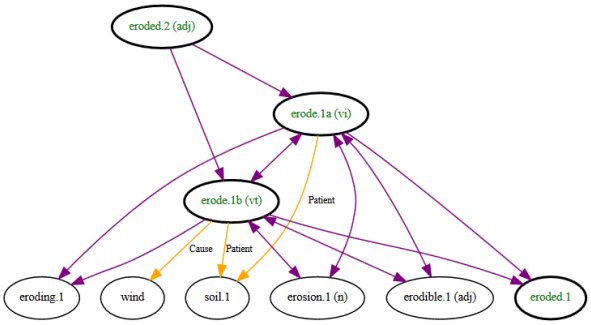


Figure 1: *Erode* and related terms in *DiCoEnviro Visuel* (Robichaud 2012)

[2] Later on we refined this first subdivision. In the French version of the DiCoInfo (L'Homme et al. 2012; L'Homme and Jia 2015), we defined sets of lexical relations and organized them into a hierarchy of classes. In this work, we referred to this system when selecting proper LFs and explanations for lexical relations. Of course, some adaptations were necessary for relations in the field of the environment.

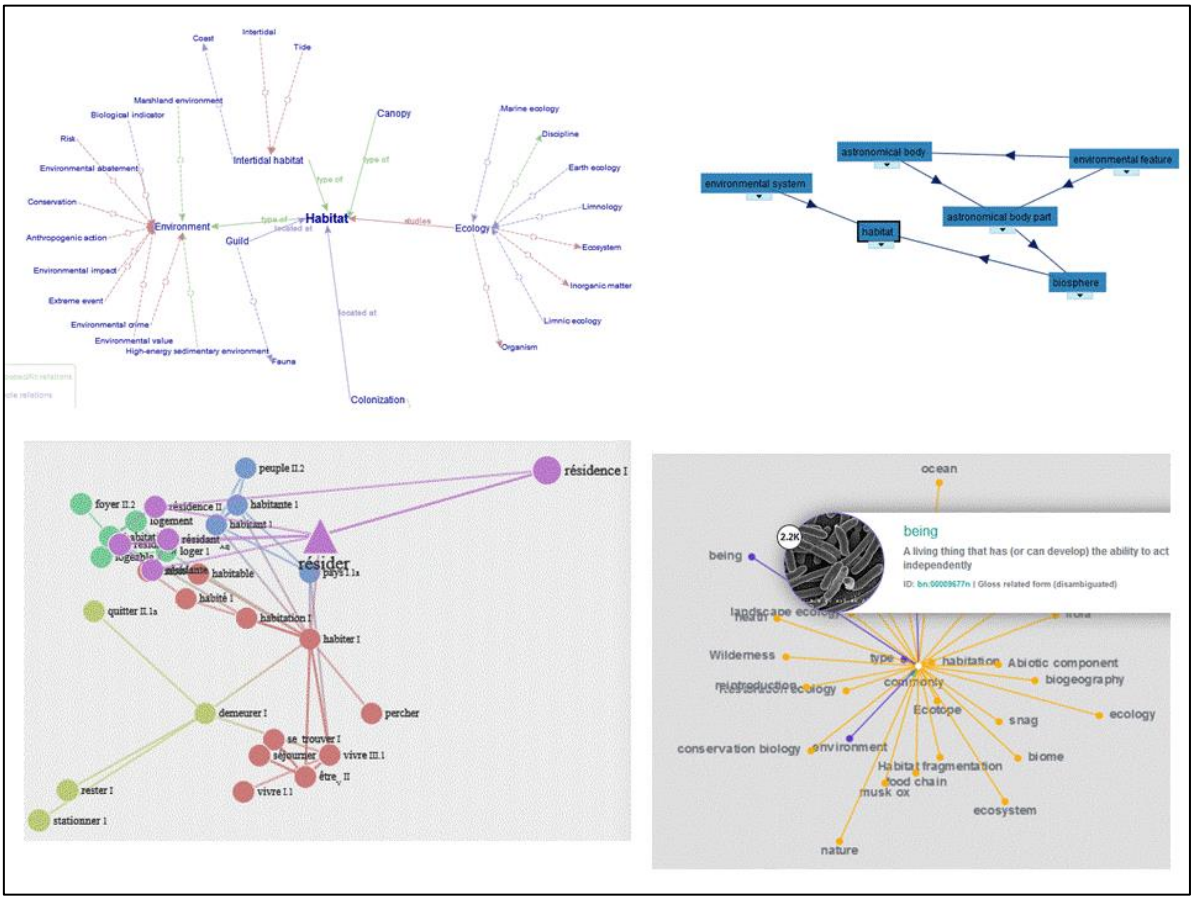


Figure 2: Figure 2: Habitat in EcoLexicon (2017); habitat in ENVO (2017); résider un the RLF (2017); habitat in BabelNet (2017).

The method that we now propose builds on the strategies mentioned in this section. However, in contrast with Robichaud (2012) that focuses on a chosen subset of relations, we take them all into account assuming that they are equally relevant to the terminological structure. Our method is designed to make the most of the rich encoding of relations in our resource as well as their classification in broader families while ensuring that the information presented can be readily interpreted. Hence we seek to establish a balance between rich encoding and simplicity of presentation and interpretation.

Furthermore, we want to allow users to discover the terminological structure gradually starting from a specific term and the relations it holds with others. Then users can use this first substructure to unveil other subparts by browsing through new sets of terms and relations.

Finally, since the DiCoEnviro is multilingual, we want to allow users to move easily from one language to another when browsing relations and explore relations as they appear in different languages.

3. Graphical representations of lexical and terminological relations

Representing terminological relations in the form of graphs has become a standard method for displaying the various interconnections between lexical units, terms or concepts and for labelling relations they hold explicitly. In some cases, graphs are superimposed on textual representations.

In what follows, we make a selection of resources based on their relevance for our own work, but many more resources resort to graphs. Corresponding illustrations appear in Figure 2.

BabelNet (Navigli and Ponzetto 2012), a large multi-lingual lexical resource, proposes a graphical view of disambiguated lexical items in which lexical, terminological and encyclopedic information can be displayed. Relations are subdivided into broad categories that can be easily identified since different colors differentiate them, such as semantically related form (e.g., *habitat → nature*), gloss related form (*ecosystem → environment*), derivationally related form (*inhabit → livable*), see also (*inhabit → live out*).

Other resources, such as the ontology Envo (2017) and the terminological knowledge base (EcoLexicon, Faber et al. 2016) for the environment, focus on conceptual relations. Envo proposes a directed graph that is superimposed on a textual taxonomy. EcoLexicon implements a directed graph with a variety of conceptual relations (e.g., "canopy" type of "habitat"; "ecology" studies "habitat"; "colonization" located at "habitat"; etc.). In addition to the graph, more information on concepts (definitions, illustrations, etc.) is provided.

Another lexical resource, The French Lexical Network, FLN (Polguère 2014) is, to our knowledge, the resource that proposes the most fine-grained labeling of relations between disambiguated lexical units. It represents lexical systems, a non-ontological model of the lexicon (Polguère

2014). The labeling is based on lexical functions (Mel'čuk et al. 1995): the graphical representation is superimposed on a textual encoding of relations and both are made available. The representation itself exploits the mathematical properties of graphs to visualize small world networks (Gaume 2008).[3]

The types and granularity of relations taken into account vary from one resource to another. As was mentioned above, we focus on lexical relations (paradigmatic and syntagmatic) between terms associated with a specialized field of knowledge. Thus our resource can complement others that represent mainly relations between lexical units (BabelNet or the FLN) or relations between specialized concepts (Envo and EcoLexicon). As far as specialized resources are concerned, our resource is probably the one presenting the largest variety of relations.

Regarding granularity, our method is closest to the FLN in the sense that each relation is labeled with a lexical function that captures three different properties (syntactic, semantic and argumental). We wish to make some of these properties visually explicit in the graph while not overloading it with textual information. However, we also group relations into larger families in order to facilitate their discovery and consultation by users. In other resources, it seems that the price to pay for obtaining clarity of presentation is to present very broad semantic categories or ignore some important distinctions between different kinds of units (terms, encyclopedic knowledge and lexical units). Conversely, resources that favor granularity of description result in a presentation that is less user-friendly. Our method combines both strategies and allows users to enter the network of relations linked to a specific term starting from general categories and breaking down these categories into smaller and more specific pieces instead of the other way around.

The resources mentioned in this section use various strategies to make important distinctions visible in graphs. Different colors or shapes are used for different sets of nodes or edges that stand for different relations. Specific relations (or sets or relations) can also be displayed and others hidden or placed in the background on demand. Our method also uses strategies that are available with the objective of making the rich encoding of our relations explicit while remaining clear for users. We also want to allow users to navigate through a network of relations in an intuitive way.

4. Organization of terminological relations in the resource

We developed our method based on all the terminological relations encoded in the DiCoEnviro (Table 2) for English, French and Portuguese.[4] It should be kept in mind that the resource is still under construction. Hence, the number of entries and relations are most likely to increase in each language. Furthermore, the coverage differs quite drastically from one language to another due to the fact that work is some languages started later than others.

Language	Entries	Relations
English	982	11,942
French	1,309	16,723
Portuguese	37	563

Table 2: Data in the DiCoEnviro taken into account in the graph (as of February 2018)

For each relation, the following information is encoded in an XML editor: the related term, a lexical function, an explanation to be displayed in the online version. As was said above, terminological relations are also placed by terminologists in more general families according to their formal or semantic properties or their relationship with the head word (see Table 1).

All relations are represented using lexical functions, most of them as defined in Mel'čuk et al. (1995). However, some adaptations were made to describe domain-specific relations as well as to account for some methodological choices. Three of these adaptations are explained in this section.

First, although a term can only share one semantic relation with another (disambiguated unit), some related terms can appear in different families. This occurs with related terms such as *unsustainable* with respect to *sustainable*: *unsustainable* is both an antonym and a term that shares a morphological relation with *sustainable*. Since morphological relations are extremely productive in specialized domains, we want our graph to account for both the semantic relation and the formal one. Hence, when encoding these term pairs, terminologists account for them with the relevant LF that describes their semantic relation with another term (*unsustainable* is an antonym of *sustainable* and vice versa), but they also indicate that they are related formally. An XML attribute is added to the description of the relation.

Secondly, most families account either for paradigmatic relations (Related meanings, Arguments, Opposites, etc.) or syntagmatic relations (Combinations). Still, some families — such as "Types of" and "Others" — contain both single-word terms and collocations since the same relation can produce two different kinds of linguistic structures. For instance, the idea of size can be realized in the form of a new term or in the form of a collocation: *habitat → microhabitat*[5]*; territory → large ~*. These relations are represented with similar LFs and in the same family. However, when encoding collocations,

[3] The FLN was made available to us as a beta version. An enhanced version will be made publicly available in the near future.

[4] Spanish will be included in the near future as soon as the entries are thoroughly revised by a native speaker. The Spanish version includes 172 entries and 2,313 relations. For the time being, the equivalent, when available, is displayed but navigation in relations is not permitted in Spanish.

[5] Moreover, *microhabitat* shares a morphological relation with *habitat*. For this particular related term, we need to account for its meaning ("small") with respect to *habitat*, the paradigmatic and the morphological relationships. This is taken into account in our graph features (Figure 3).

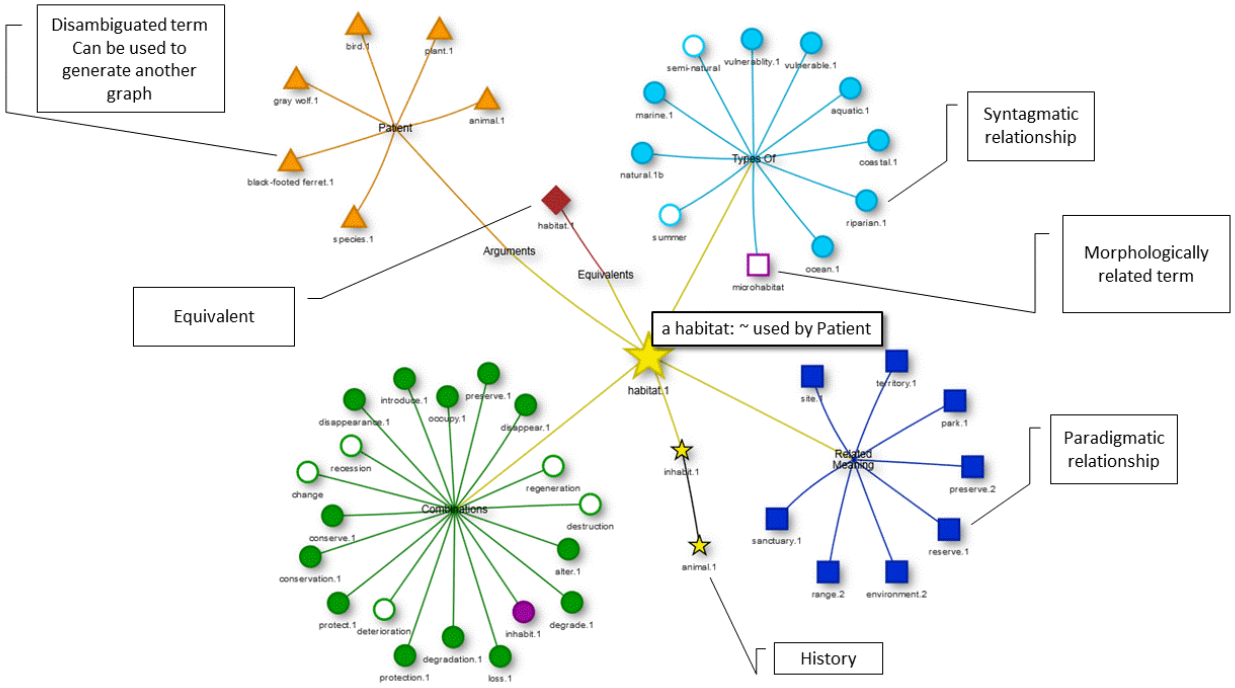


Figure 3: *Habitat* in the NeoVisual

terminologists also specify the way the collocate combines with the key word (e.g. habitat → *large ~, occupy a ~*).

Thirdly, some relations are domain-specific and could not be encoded with standard LFs, especially in the "Types of" family. Previous work was carried out to propose an encoding adapted to terms in the field of computing (L'Homme and Jia 2015). We used some of these proposals but needed to refine the encoding for environmental specific relations. We also subdivide exact synonyms into more specific categories: synonym (*carbon dioxide → carbonic acid gas*), symbol (*carbon dioxide → CO_2),* abbreviation (*greenhouse gas → GHG);* variant (*microorganism → micro-organism)*; feminine (Fr. *expert → experte*).

4.1 Tool used to present relations

For some years now (and especially since the emergence of social networks), there has been a large number of computer programs available to generate (or draw) visual graphs representing networks. Within the present project, we selected the package *Network* from vis.js (de Jong et al. 2015-2017), which is a library of programmable functions that can describe and generate natively dynamic graphs directly (as JavaScript code) in a Web navigation browser.

It is worth mentioning that with this particular package we do not have to manage the orientation of the layout, or the placement of nodes. Apart from the aesthetic aspect of the drawing itself, the feature we preferred – since it was in line with our objectives – was the ability to easily interact with the graphs created by means of popups, clicks and drag features. In particular, this package supports dynamic changes in the shapes and colors of nodes and edges under certain conditions as users explore the network.

4.2 Organization of information in the graph

In our method, users start discovering the terminological structure with a disambiguated term (a choice also made in BabelNet[6]). Although polysemy is reduced when focusing on lexical items from the point of a specific subject field, there is still a sizable amount of polysemy in the DiCoEnviro. Furthermore, the listing of relations and their fine-grained encoding can only be carried out if a single meaning is considered.

In order to meet the objectives we defined (clarity of presentation and ease of interpretation), we made the following preliminary choices (Figure 3):

- Users first obtain a graph that displays the searched term and all other terms to which it is connected.
  - A search field appears on the left hand side of the screen. It is equipped with an auto-completion feature that shows which terms appear in the resource. The list contains terms in three of the languages taken into account in the DiCoEnviro (English, French and Portuguese).
  - The central node of the graph is the disambiguated term (a reminder of its meaning is given by means of its argument structure displayed on demand).
  - Users obtain a global view of the relations held by the term they searched and other terms.
- Only those terms that are directly connected to the searched term are presented.

[6] The FLN also accounts for disambiguated lexical units; Morever, it presents the relations between the different meanings of a polysemous item.

- Related terms in a language are clustered into up to eight different families (displayed only if relevant for the search term).
- Labels are provided for families and a different color is assigned to each.
- Morphologically related terms can appear in a family labeled "Word family" colored in purple. However, they can also be encoded elsewhere. Hence, the corresponding node is colored in purple throughout the graph even if the related term is encoded in another family (Figures 3 and 4).

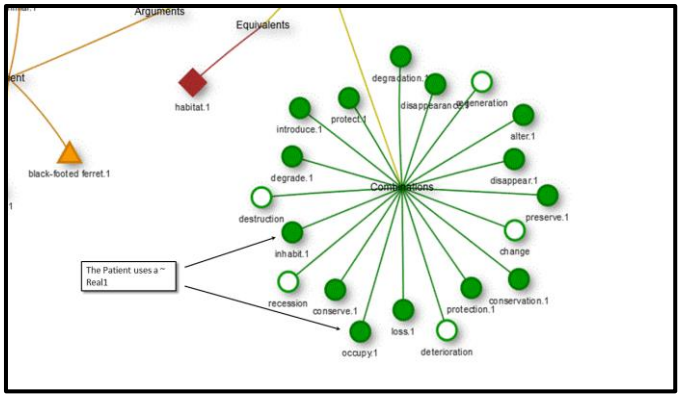


Figure 5: Labelling of relations with lexical functions and explanations

- Equivalents when available are clustered in a ninth family colored in brown.

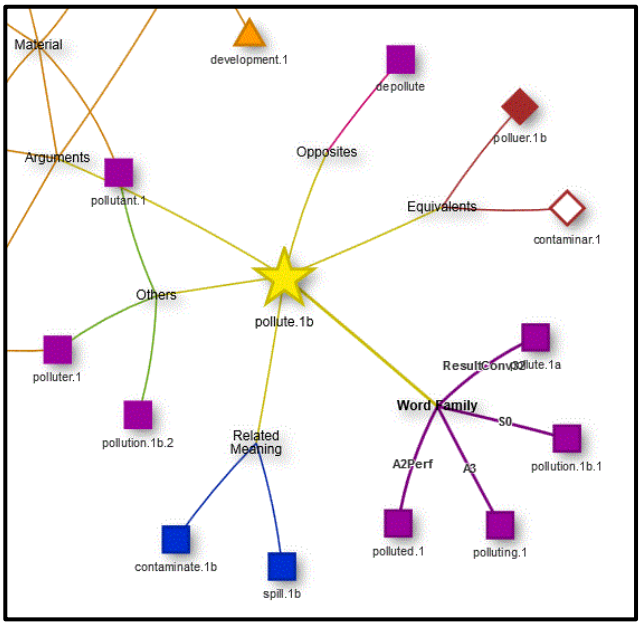


Figure 4: Morphologically related terms highlighted throughout the graph

- Within families, related terms are further distinguished with nodes of varying shapes:
  - Nodes that correspond to paradigmatically related terms are shaped in squares.
  - Nodes that correspond to syntagmatically related terms are shaped in circles.
  - Nodes that correspond to instantiations of arguments are shaped in triangles.
  - Nodes that correspond to equivalents in other languages are shaped in diamonds.

- Users can explore more deeply the relations within a family:
  - Selecting a family label results in having the relations in this family highlighted.
  - Users can also zoom into a specific relation by selecting an edge.
  - The label of the relation (an LF) and an explanation are provided in the form of a pop-up (Figure 5).

The navigation through the terminological structure is permitted with the following features:

- Nearly all nodes[7] in a graph are disambiguated terms and can be used to generate another graph that has the properties mentioned above.
  - A special feature is provided for synonyms and other lexical forms (abbreviations, variants) that convey the same meaning as the term represented by the central node. Selecting a synonym will highlight all edges leading to families indicating that relations are also valid for synonyms (Figure 6).

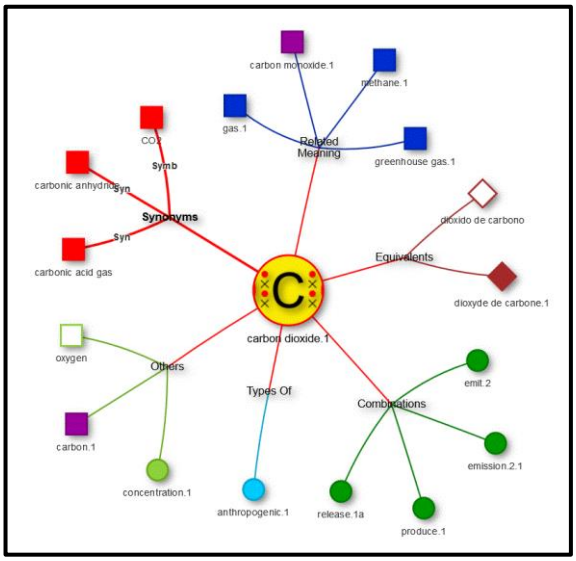


Figure 6: Highlighting links in red for exact synonyms

- Users can keep track of the navigation history (as in BabelNet) and return back to a graph already visited with the same clickable mechanism. If users navigate between languages, the history allows

[7] Since the DiCoEnviro is under construction, some nodes have not been completely disambiguated yet, but they will be once these terms are described in a separate entry of the resource. In these cases, nodes are hollow to indicate that the associated term is not fully disambiguated. This feature applies to related terms within a language but also to equivalents in other languages.

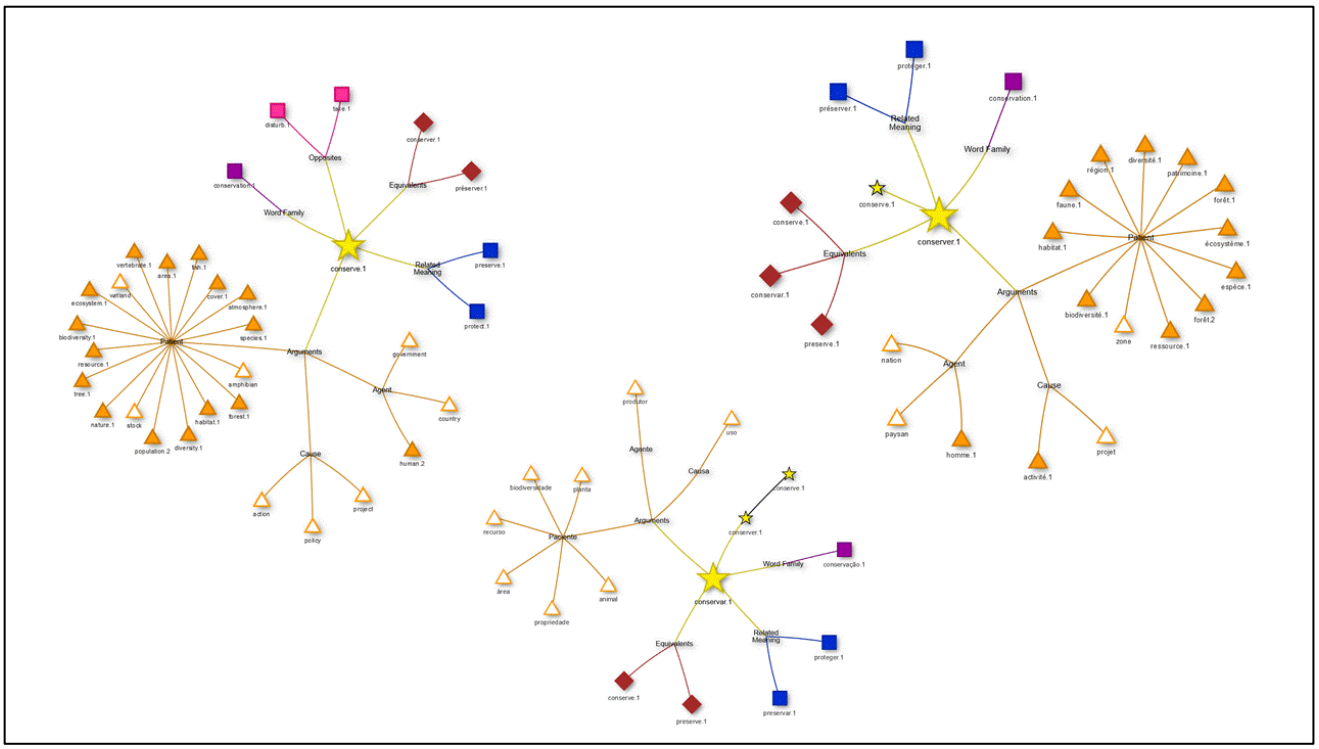


Figure 7: *Conserve* in English, *conserver* in French and *conservar* in Portuguese and their terminological relations

them to go back to a previous one. The history is displayed as long as users do not select a new term in the search window.

- Users can navigate in a different language by selecting an equivalent. This will result in generating graphs in different languages (Figure 7).

- Access to the textual version (the original DiCoEnviro) is permitted when clicking on the central node of the graph. Similarly, users can refer to the graph from each entry in English, French and Portuguese.

5. Conclusion and future work

The method described above is implemented in a tool called NeoVisual that is available at: http://olst.ling.umontreal.ca/dicoenviro/ neovisual/. This graphical tool takes manually encoded relations from a terminological resource and presents them in a way that allows users to first obtain a general picture of the set of relations shared by a specific term with others. Then, they can access more information about a set of relations – a family – or a specific relation by selecting an item in the graph. The tool also allows them to browse the terminological structure of a specialized domain, i.e. the environment. Since the NeoVisual takes the relations as they are encoded in the original resource, terminologists do not need to change any aspect of their methodology when they add information to the entries. The number or relations taken into account by the NeoVisual increases as more data is added to the DiCoEnviro. Furthermore, terms in new languages could be added to the resource and taken into account graphically. Spanish should be added to the tool in the near future.

Our objectives were to make the most of a rich encoding based on lexical functions while focusing on the clarity their presentation. The graphical functionalities allow users:

1. To organize terminological relations according to their general meaning by first displaying them in *families*;
2. To further explain their meaning by adding labels on specific edges;
3. To further classify terminological relations according to formal properties (paradigmatic, morphological, syntagmatic). All these properties can be visualized all at once without having to regenerate graphs for a specific term.

The navigation functionalities are also designed to facilitate navigation through relations between terms in a specific language and to move from one language to another.

We believe that a graphical display of terminological relations nicely complements a textual (dictionary-like) resource. It has the potential of making distinctions between relations readily visible (paradigmatic vs. syntagmatic; different families, etc.). These distinctions would be much more difficult to account for in textual format. However, it should not replace the textual resource altogether since other forms of information are better represented in textual format (definitions, contexts, etc.).

Our resource could be used for teaching purposes in terminology or specialized translation. Our next step is to collect user feedback and assess to what extent graphical displays can be used as a source of information for understanding the meaning of terms and visualize their position in a terminological structure. Users could be asked to use resources that present relations in the form of text and graph; or compare graphs with different features.

6. Acknowledgements

This work is supported by the Social Sciences and Humanities Research Council (SSHRC) of Canada. We would like to thank reviewers and colleagues for providing us with useful feedback on different versions of the NeoVisual.

7. Bibliographical References

Faber, P., León-Araúz, P. and Reimerink, A. (2016). EcoLexicon: New features and Challenges. In *GLOBALEX 2016: Lexicographic Resources for Human Language Technology in conjunction with the 10th edition of the Language Resources and Evaluation Conference*, ed. by Kernerman, I., I. Kosem Trojina, S. Krek, and L. Trap-Jensen, 73-80. Portorož.

Gaume B. (2008). Mapping the Forms of Meaning in Small Worlds. *Journal of Intelligent Systems* 23: 848–862.

de Jong, J., A. de Mulder and Stellingwerff, L. (2015-2017). *Vis.js: A dynamic, browser based visualization library*. Netherlands: Almende B.V.

L'Homme, M.C. and Jia, Z. (2015). Classement des combinaisons lexicales spécialisées à base nominale dans un dictionnaire d'informatique. *Cahiers de lexicologie* 106, pp. 229-251.

L'Homme, M.C., Robichaud, B. and Leroyer, P. (2012). Encoding collocations in *DiCoInfo: from formal to user-friendly representations*, In Granger, S. and Paquot, M. (eds.). *Electronic Lexicography*, Oxford: Oxford University Press, pp. 211-236.

Mel'čuk, I., Clas, A., and Polguère, A. (1995). *Introduction à la lexicologie explicative et combinatoire*. Duculot : Louvain-la-Neuve.

Mel'čuk, I. and Polguère, A. (2007). *Lexique actif du français. L'apprentissage du vocabulaire fondé sur 20 000 dérivations sémantiques et collocations du français*. Bruxelles: De Boeck.

Navigli, R. and Ponzetto, S. (2012). BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. In *Artificial Intelligence* 193, Elsevier, pp. 217-250.

Polguère, A. (2014). From Writing Dictionaries to Weaving Lexical Networks. In *International Journal of Lexicography* 27(4).

Robichaud, B. (2012). Logic Based Methods for Terminological Assessment. In N. Calzolari et al. (eds.). *Proceedings of the Eight International Conference on Language Resources and Evaluation* (LREC'12), Istanbul, Turkey

8. Language Resource References

BabelNet (http://babelnet.org/)
DiCoEnviro (http://olst.ling.umontreal.ca/dicoenviro)
DiCoInfo (http://olst.ling.umontreal.ca/dicoinfo)
DiCoInfo Visuel (http://olst.ling.umontreal.ca/dicoinfo/visuel.php)
EcoLexicon. Terminological knowledge base (http://ecolexicon.ugr.es/en/index.htm)
Environmental Ontology, ENVO (http://bioportal.bioontology.org/ontologies/ENVO)
Viz.js. A dynamic, browser-based visualization library (http://visjs.org).

Rollenwechsel-English: a large-scale semantic role corpus

Asad Sayeed[1], **Pavel Shkadzko**[2], **Vera Demberg**[2]
[1]CLASP, Department of Philosophy, Linguistics, and Theory of Science, University of Gothenburg
[2]Saarland University
asad.sayeed@gu.se, p.shkadzko@gmail.com, vera@coli.uni-saarland.de

Abstract

We present the Rollenwechsel-English (RW-eng) corpus, a large corpus of automatically-labelled semantic frames extracted from the ukWaC corpus and BNC using Propbank roles. RW-eng contains both full-phrase constituents for labelled roles as well as heads identified by a series of heuristics. This corpus is of a scale and size suitable for new deep learning approaches to language modelling and distributional semantics, particularly as it pertains to generalized event knowledge. We describe the structure of this corpus, tools for its use, and successful use cases.

Keywords: semantic roles, web corpus, labelled data

1. Motivation

Semantic role labelling is a comparatively mature task in natural language processing. Typically, it takes the form of supervised classification or language modeling task, although there is more recent work in unsupervised induction of semantic roles (Titov and Klementiev, 2012). These approaches tend to have application goals in traditional areas of text-based natural language processing.

A major area of psycholinguistic research is the influence of semantic structure on adult sentence processing. Does the association between "cake" and "cutting" have an effect on processing difficulty of future sentence constituents in a sentence that starts with "The child cut the cake…", and to what extent is this association influenced by the semantic role-based selectional preferences of the verb (McRae et al., 1998; McRae et al., 2005; Ferretti et al., 2001; Bicknell et al., 2010)? A further question pertains to the internal representation of linguistic knowledge: to what extent do distributions of roles observed in text really reflect the internal state of human knowledge (generalized event knowledge)? Human being easily conceive of a knife as a proper instrument for cake-cutting—but knowledge of affordances allows them also to see other sharp objects (e.g. swords, floss) as potential cake-cutters, even if the co-occurrence frequency in the corpus is low. This possible mismatch between corpus frequency and the underlying cognitive model (Amsel et al., 2015) affects such application areas as dialogue systems or indeed any application in which intuitions similar to human ones are required.

Nevertheless, the expansion of corpus availability, increase in computing power, and powerful extension of traditional machine learning techniques such as deep learning provides new opportunities to understand these questions. What has been missing until recently, however, is a variety of data sources capable of supporting the type of hypothesis-testing about the ability of the new techniques to acquire the latent information about semantic relationships within the sentence.

We contribute towards addressing this in this paper by introducing the Rollenwechsel-English (RW-eng) corpus. RW-eng[1] is labelled automatically with semantic roles which are then reprocessed heuristically to yield a rich representation of verb-noun relationships and subcategorization frames. RW-eng is based on the full ukWaC corpus (Ferraresi et al., 2008) and the British National Corpus (BNC Consortium, 2007). The semantic role labelling is done by SENNA (Collobert and Weston, 2007; Collobert et al., 2011), a labeller that does not, as most other SRL tools do, rely directly on the syntactic parse of the sentence, allowing it to capture relationships that syntax-based SRL does not, and therefore implicitly permitting some investigation of semantic roles that are not totally confounded with an underlying syntactic theory.

In the remainder of this paper, we outline the process by which the sentences are labelled, the output format of the corpus and thus the data and relationships the corpus encodes, and publicly-available tools for corpus generation and access. We also discuss some scientific use cases for the corpus and results already obtained from it.

2. Corpus generation

2.1. Preprocessing

The bulk of our Python-based processing pipeline (`ukwac2tensor`) for the RW-eng corpus is available as a `git` repository on the web[2]. The initial input for the pipeline are the ukWaC and BNC corpora parsed by MaltParser (Nivre et al., 2007) and tagged with part-of-speech labels, in a column format supplied to us by the creators of ukWaC. For parallelization purposes, we divide the corpus up into 3500 segments approximately equal by number of documents. This division stays with the pipeline to the final output.

Since the ukWaC data comes from web data that is sometimes not codepage-consistent, we replaced or removed special characters so that it was compliant with UTF-8. Our goal was to produce XML output that would validate properly with common Python-based XML parsers.

[1] `http://rollen.mmci.uni-saarland.de/RW-eng/`

[2] `https://github.com/tastyminerals/ukwac2tensor/`

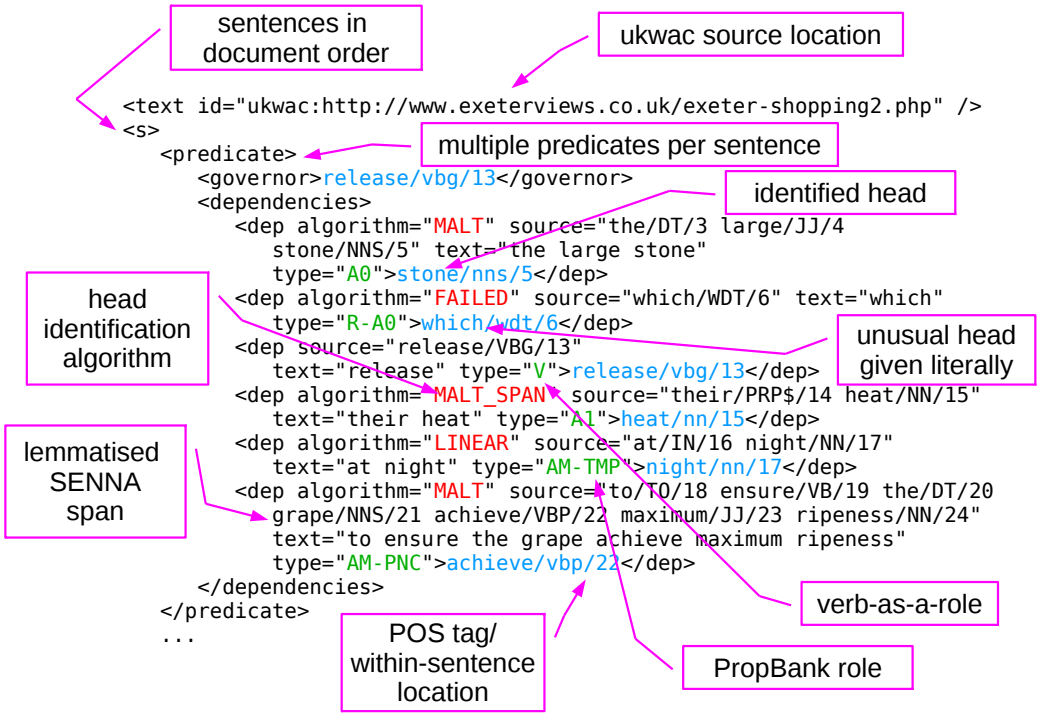


Figure 1: Excerpt of a single annotated predicate from the RW-eng corpus.

2.2. Labelling process

The labelling processing depends on the SENNA semantic role labeller. Although there are more recent and powerful labellers, we chose SENNA because it does not depend on an existing syntactic parser or heavy linguistic analysis while still maintaining reasonable accuracy (75.49% on the CoNLL 2005 task). SENNA performs labeling comparatively rapidly (36s on a Macbook i7 Pro for the CoNLL 2007 task), allowing for more flexible experimentation in corpus development over corpora the size of RW-eng. SENNA's characteristics also matched our research goals, which was to have a large-scale semantic role data source that was not completely confounded with inferences over syntactic data, as it would have been if a sophisticated parsing apparatus had been involved.

The overall procedure for labelling the corpus is as follows. For every sentence:

1. We ran the sentence through the SENNA labeller using the default model supplied with SENNA. The output consists of identified verbal predicates, and labelled spans of text connected with PropBank-style roles (Kingsbury and Palmer, 2002) to each predicate. SENNA can label overlapping stretches of text with different roles for different predicates. In our output, there is always guaranteed to be at least one verbal predicate associated with a SENNA-identified role-labelled span.

2. Each predicate verb and its associated role-labelled spans were identified in the SENNA output and grouped together.

3. Each role-labelled span was run through a heuristic process to detect the head of the span, based on a combination of part of speech tags and the MaltParser dependency trees we received with the sentences.

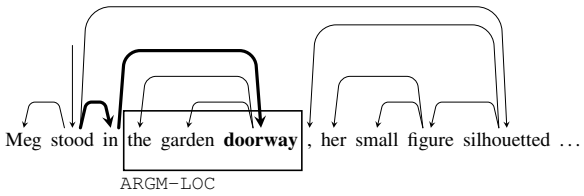


Figure 2: We illustrate the MALT heuristic with the above sentence, in which SENNA has assigned ARGM-LOC (AM-LOC in RW-eng's format) to the"the garden doorway". Transitively passing through "in", we find that "doorway" is the head, as it is the first item we encounter.

Head-detection is an issue where the nominal span is not a singleton word. This is most often the case. We take as heads only items that have nominal or verbal (open-class) POS tags. We used three heuristics for identifying the head:

- MALT – dependency links are followed from the associated verbal predicate iteratively through the dependency tree until the first word with an open-class POS tag is encountered within the bounds of the span (figure 2).

- MALT-SPAN – if MALT fails to find a verbal or nominal connection inside the span, instead look for the open-class POS-tagged word that is directly connected via the MALT parse tree to the leftmost word within the span, relative to the connections of all the other

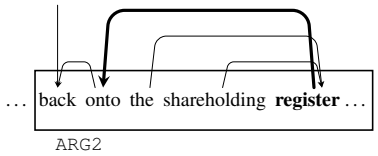


Figure 3: We illustrate the MALT-SPAN heuristic with the above ARG2-labelled phrase. The noun "register" has a connection that reaches the furthest left of all the other constituents of the span that are eligible to be identified as a head (i.e., not "onto").

neighbours. The intuition behind this is that spans for which the MALT parse does not provide a direct connection to an eligible word are often governed by a preposition or other typical left-peripheral function word (figure 3).

- LINEAR – if MALT-SPAN fails to find a qualified head, the words in the labelled span are considered iteratively from left to right using POS-based heuristics similar to those of (Magerman, 1994). More specifically, it skips over words with tags likely to be adjuncts, such as adverbs and adjectives, and when it encounters a noun or a verb, it looks ahead to make sure that it is not in an adjunct position. It returns the first noun or verb that does not have evidence, by POS tag heuristics, of being an adjunct.

The heads discovered by these heuristics are labelled as such in the output corpus, and if none of the heuristics encounter an appropriate item, it is labelled as FAILED and accepted literally as a single, full constituent.

2.3. Output format

Figure 1 contains an example sentence annotation extracted from the corpus. We describe the main features of the corpus here.

Every word in the original corpus that is mentioned in the RW-eng annotation takes the form *word/pos/N*, where *N* is the position of the word relative to the first word in the sentence, starting at 1.

Each document boundary is heralded by the text tag with the source identifier from the original corpus. Every sentence in the original corpus obtains a corresponding s tag, which contains all predicates identified by SENNA inside the sentence[3].

Each predicate tag contains one governor tag and one dependencies tag. The governor tag mentions the verb that governs the entire predicate. The dependencies tag contains a series of dep tags, which are the roles assigned by the verb and the associated text spans. A dep tag contains a source attribute, which is the tagged full span; a text attribute, which is the lemmatized text without the tagging; a type attribute, which is the Propbank-based semantic role label assigned by SENNA; and usually an algorithm attribute, which mentions one of the head-finding algorithms or FAILED if none of them worked. The verb is also repeated here as a dep tag will a type of V and no algorithm. The contents of the dep tag is the tagged word discovered by the head-finding algorithm, or the full phrase in the case of FAILED spans.

[3]Because the original corpora have access restrictions, we do not provide the original sentences, which can be obtained by matching with ukWaC and BNC. RW-eng contains word position information that only permits partial reconstruction of the original sentences; any text that SENNA does not associate with a verb requiring roles or a role-labelled text span is missing. The provided text is also lemmatized.

3. Tools

Python-based tools for manipulating the corpus exist include the aforementioned ukwac2tensor and the ukwac-heads-api[4]. The ukwac2tensor tools not only contain the scripts that convert the MALT dependency-parsed corpora to our role-labelled XML-format, but they also contain a tool to convert the XML-formatted corpus into a Pandas dataframe representing an order-3 tensor stored in HDF5 format. This tensor represents links between verbs and role-fillers found in the corpus via either counts or pointwise mutual information statistics. This representation can be used, for example, to efficiently extract role-specific feature vectors.

The ukwac-heads-api is intended for efficient access to the RW-eng data. The API contains functionality to traverse the corpus and create filtered vocabularies as well as to produce efficient Python generators for querying and sampling the corpus for applications such as deep learning. The API allows for filtering according to, for example, head-finding algorithm, and it permits iteration through randomly selected role-sets.

4. The corpus in use

RW-eng contains not only identified verbs and role-sets but also much of the underlying evidence used to detect them, such as POS tags and positional information. The head-detection as well as the full text spans of role-fillers permit the analysis of the effect of modifiers and adjuncts on thematic fit.

4.1. Corpus characteristics

RW-eng contains approximately 78 million sentences over 2.3 million documents. All together, the sentences, contain approximately 210 million identified predicates with dependent roles and 704 million identified role-fillers.

In the next section, we refer to our published results for which we used the heads discovered by our head-finding algorithm; this extrinsic evaluation of our heuristics proved to be highly successful. However, an author examined 200 of the heads in order to obtain an intrinsic estimate of how well the heuristic did, assuming the SENNA labelling as given in order to assess the heuristic alone. In 39% of the identified roles, the head-finding algorithm was not neces-

[4]https://github.com/tastyminerals/ukwac-heads-api/

Model	*Coverage* (%)	ρ
TypeDM+SDDM (Malt-only)	99	59
SDDM (Malt-only)	99	56
TypeDM	100	51
Padó	97	51
ParCos	98	48
DepDM	100	35

Table 1: SDDM and TypeDM+SDDM (Sayeed et al., 2015) are unsupervised thematic fit models based on a development version of RW-eng. These outperform other unsupervised models on a thematic fit correlation task with a common human judgement correlation task (Padó, 2007), including the very similar syntax-based TypeDM model (Baroni and Lenci, 2010).

sary because SENNA identified a single word[5]. In 48.5% of the roles, the head-finding algorithm correctly found a head in a filler with multiple words. In 4.5% of the roles, SENNA was attempting to analyze a non-sentence (such as a properly listing) and found a "gibberish" predicate and role-filler for which no head could be evaluated. 1% of the roles were multi-word expressions to which the heuristics applied a `FAILED` label. Finally, 7% of the roles were interpretable multi-word role fillers for which the heuristics identified a wrong head.

In the same sample, the `MALT` heuristic obtained a head 47% of the time. `MALT-SPAN` identified the head 11% of the time, and `LINEAR` 32%. The heuristics `FAILED` 10% of the time.

4.2. Distributional semantics and deep learning for thematic fit

We are providing RW-eng to the public domain after having developed it for our thematic fit modeling project, which has produced a number of successful results based on it. Thematic fit is the extent to which a a role-filler satisfies a given thematic role for a given predicate; it differs from selectional preferences in that it measures the extent to which a native speaker would accept the role-filler in that role, as opposed to which fillers a native speaker most expects in that role. For example, a given native speaker may expect that a secretary might be highly likely to take notes (a selectional preference), but may have a high degree of thematic fit for a doctor taking notes, even if it is not the first category of professions of which the native speaker thinks in that context. Computational models of thematic fit are typically evaluated by correlation with averaged human ratings. RW-eng has successfully been used to show that a count-based, unsupervised model of thematic fit based on PropBank roles (Sayeed et al., 2015) either outperforms a similar syntax-based model either overall or over different parts of the evaluation data; combining them produces the best-performing unsupervised models (table 1). RW-eng has also been involved in the evaluation of the role of verb senses in thematic fit modelling (Greenberg et al., 2015) and in visualization of thematic fit spaces (Sayeed et al., 2016), which is available on the web[6]. More recently, RW-eng has been used to train neural network models of thematic fit that allow for the prediction of role-fillers given varying combinations of other role-fillers, producing the best-performing models over less frequent roles like instrument and location(Tilk et al., 2016).

[5]Of these 97 items in total, 35% had ambiguous heads with possible conjoined candidates, multi-word noun compounds, ambiguous pronominal references and so on; our heuristics only return one head and thus favour precision to recall.

5. Future work

There are a number of directions in which development of this corpus could proceed, including the update and expansion of this corpus to contain more genres such as Wikipedia text. One possible future direction would be to replace SENNA with a more recent semantic role-labeller and see if the added accuracy produces better performance on our modelling tasks. However, our preliminary experiments with this have suggested that more accurate labelling produces an overfitting effect in our unsupervised models (which depend on cosine similarity in high-dimensional space).

Another area in which we are investigating the potential for improvements is in the labelling of nominal predicates (e.g., a house can fill the patient role of a sale). Our predicates are strictly based on verbs, because that is how many SRL tools including SENNA are trained. However, not only are nominal predicates frequent in language, they may be distributed differently from verbs; a resource at RW-eng's scale may be required to investigate these differences from the perspective of distributional semantics.

6. Concluding remarks

The principal effort in developing Rollenwechsel-English was that of processing and aligning heterogeneous data sources at large scale. SENNA and MaltParser produce analyses that sometimes do not directly match, and there are items in the ukWaC corpus that create practical challenges for each of these, such as special characters, accidental non-English content, specialized genres that are not sentences, and so on. However, we have derived a resource that has broad potential applications not only in psycholinguistic modeling, which was our original application, but also in other areas of semantically-aware language modeling, and we now provide it in a form that other researchers can use.

Acknowledgements

This research was funded in part by the German Research Foundation (DFG) as part of SFB 1102 "Information Density and Linguistic Encoding" as well as EXC 284 Cluster of Excellence "Multimodal Computing and Interaction" during the first author's previous employment. The work reported in this paper was also supported by a grant from the Swedish Research Council for the establishment of the Centre for Linguistic Theory and Studies in Probability (CLASP) at the University of Gothenburg.

[6]`http://roleo.coli.uni-saarland.de/`

7. Bibliographical References

Amsel, B. D., DeLong, K. A., and Kutas, M. (2015). Close, but no garlic: Perceptuomotor and event knowledge activation during language comprehension. *Journal of Memory and Language*, 82(Supplement C):118 – 132.

Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Bicknell, K., Elman, J. L., Hare, M., McRae, K., and Kutas, M. (2010). Effects of event knowledge in processing verbal arguments. *Journal of Memory and Language*, 63(4):489–505.

BNC Consortium. (2007). The British National Corpus, version 3 (BNC XML Edition). Distributed by Oxford University Computing Services on behalf of the BNC Consortium.

Collobert, R. and Weston, J. (2007). Fast semantic extraction using a novel neural network architecture. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 560–567, Prague, Czech Republic, June. Association for Computational Linguistics.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Ferraresi, A., Zanchetta, E., Baroni, M., and Bernardini, S. (2008). Introducing and evaluating ukWaC, a very large web-derived corpus of english. *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can we beat Google*, pages 47–54.

Ferretti, T. R., McRae, K., and Hatherell, A. (2001). Integrating verbs, situation schemas, and thematic role concepts. *Journal of Memory and Language*, 44(4):516–547.

Greenberg, C., Sayeed, A., and Demberg, V. (2015). Improving unsupervised vector-space thematic fit evaluation via role-filler prototype clustering. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL HLT)*.

Kingsbury, P. and Palmer, M. (2002). From TreeBank to PropBank. In *LREC*. European Language Resources Association.

Magerman, D. M. (1994). *Natural Lagnuage Parsing as Statistical Pattern Recognition*. Ph.D. thesis, Stanford University.

McRae, K., Spivey-Knowlton, M. J., and Tanenhaus, M. K. (1998). Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language*, 38(3):283–312.

McRae, K., Hare, M., Elman, J. L., and Ferretti, T. (2005). A basis for generating expectancies for verbs from nouns. *Memory and Cognition*, 33(7):1174–1184.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Padó, U. (2007). *The integration of syntax and semantic plausibility in a wide-coverage model of human sentence processing*. Ph.D. thesis, Saarland University.

Sayeed, A., Demberg, V., and Shkadzko, P. (2015). An exploration of semantic features in an unsupervised thematic fit evaluation framework. In *IJCoL vol. 1, n. 1 december 2015: Emerging Topics at the First Italian Conference on Computational Linguistics*, pages 25–40. Accademia University Press.

Sayeed, A., Hong, X., and Demberg, V. (2016). Roleo: Visualising thematic fit spaces on the web. In *Proceedings of ACL-2016 System Demonstrations*, pages 139–144, Berlin, Germany. Association for Computational Linguistics.

Tilk, O., Demberg, V., Sayeed, A. B., Klakow, D., and Thater, S. (2016). Event participant modelling with neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 171–182.

Titov, I. and Klementiev, A. (2012). A Bayesian approach to unsupervised semantic role induction. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 12–22. Association for Computational Linguistics.

Towards a Standardized Dataset for Noun Compound Interpretation

Girishkumar Ponkiya*, **Kevin Patel***, **Pushpak Bhattacharyya***, **Girish K Palshikar**†
*Indian Institute of Technology Bombay, Mumbai, India {girishp,kevin.patel,pb}@cse.iitb.ac.in
†TRDDC, Pune, India gk.palshikar@tcs.com

Abstract

Noun compounds are interesting constructs in Natural Language Processing (NLP). Interpretation of noun compounds is the task of uncovering a relationship between component nouns of a noun compound. There has not been much progress in this field due to lack of a standardized set of relation inventory and associated annotated dataset which can be used to evaluate suggested solutions. Available datasets in the literature suffer from two problems. Firstly, the approaches to creating some of the relation inventories and datasets are statistically motivated, rather than being linguistically motivated. Secondly, there is little overlap among the semantic relation inventories used by them. We attempt to bridge this gap through our paper. We present a dataset that is (a) linguistically grounded by using Levi (1978)'s theory, and (b) uses frame elements of FrameNet as its semantic relation inventory. The dataset consists of 2,600 examples created by an automated extraction from FrameNet annotated corpus, followed by a manual investigation. These attributes make our dataset useful for noun compound interpretation in a general-purpose setting.

Keywords: noun compounds, compounds, interpretation, semantic relations.

1. Introduction

Noun compounds are continuous sequences of nouns that act as a single semantic construct. They raise interesting challenges in Natural Language Processing. Without proper interpretation and paraphrasing of noun compounds, NLP methods may fail miserably at different tasks. The meaning of a noun compound is composed of the meanings of the individual constituents and the way they are semantically related.

Noun compound interpretation is the task of detecting this underlying semantic relation (*e.g., student protest*: *student* ← AGENT ← *protest*). It is an important submodule for various NLP tasks such as machine translation (Baldwin and Tanaka, 2004; Balyan and Chatterjee, 2014), question answering (Ahn et al., 2005), *etc.*

Noun compound interpretation can manifest itself in two settings: out-of-context interpretation and context-dependent interpretation. In out-of-context interpretation, given the noun compound, the task is either to annotate it with a relation from a semantic relation inventory (*e.g., student protest*: AGENT), or to produce a paraphrase (*e.g., student protest*: "*protest carried out by student*").

Any automated approaches for noun compound interpretation need a semantic relation inventory of noun-noun relations and an annotated dataset on which models can be trained. However, there is little agreement among researchers regarding the set of relations that can hold between the constituents of a noun compound. None of the proposed semantic relation inventories has been accepted by the community as complete and appropriate for general-purpose text. Some are coarse-grained, while others are fine grained. There is little overlap among them. Also, some of these inventories and their accompanying dataset are created from another application's perspective, and not for the sake of creating a noun compound dataset. Thus they cannot be used for learning noun compound interpretation in a general-purpose setting.

A dataset that can be used in general-purpose setting needs to be linguistically grounded. One such work is that of Levi's, who claims that noun compounds are created either through predicate deletion or through predicate nominalization. For example, *student protest* and *student demonstration* are examples of predicate nominalization with heads as verbal form and nominalized form, respectively. *Orange juice* is an example of predicate deletion as connecting predicate (like, *made_of*) has been simply dropped while creating the compound. We ground our dataset on this theory.

FrameNet is a lexical resource based on the theory of frame semantics. Among other things, it captures predicate-argument interactions. Such information can be used for compounding. For instance, *border camp* with RESIDENCE:LOCATION and *rescue attempt* with ATTEMPT:GOAL are examples of predicate deletion and predicate nominalization, respectively, with corresponding frame and frame-element as labels. Intuitively, the frame elements are descriptive enough of the relation between the predicate and argument.

One can use FrameNet information – definition and examples of frame elements – to develop a system for automatic interpretation. In addition, one can use FrameNet dataset (annotated at sentence level) to add-on the frame element prediction for noun compounds. One can also use hierarchy of frame elements (defined along with frame relations) to generalize the semantic relations. Thus, through this paper, we release a dataset[1] that is linguistically grounded (by Levi's theory) and uses frame elements as semantic relation inventory.

The rest of the paper is organized as follows: Section 2. covers some background needed for further discussion. Section 3. discusses other semantic relation inventories, highlighting their shortcomings. Section 4. discusses the creation and statistics of our dataset. Section 5. presents several observations during this activity, followed by the conclusion and future work.

[1]Available at `http://www.cfilt.iitb.ac.in/standard_nc_sr`

2. Background

A noun compound can be of any length. A typical way for interpretation of longer (having more than two components) is parsing it to get a binary tree based on head-modifier pairs and interpret each internal node of the tree with two children of the node as components. For example, parse trees (in bracketed form) for "*plastic water bottle*" and "*water bottle cap*" are as follows:

[plastic [water bottle]] *[[water bottle] cap]*

After parsing, the problem reduces to the interpretation of two components of each internal node. In literature, most work focuses on the interpretation of noun-noun compounds, *i.e.*, noun compounds composed of two nouns. In the rest of this paper, by noun compound, we mean noun-noun compounds.

For representation of the semantic relation between the components of noun compounds, there are two major ways:

Paraphrasing: paraphrase a noun compound to show how the components are related (*e.g.*, *orange juice*: "*juice made of orange*", "*a drink consisting of the juice from oranges*", *etc.*) (Butnariu et al., 2009; Hendrickx et al., 2013). There can be multiple paraphrase of a noun compound.

Labeling: Assign a relation from predefined set of abstract relations (*e.g.*, *orange juice*: SUBSTANCE/MATERIAL/INGREDIENT). (Levi, 1978; Warren, 1978; Tratz and Hovy, 2010)

Labeling is the most widely used representation in literature for noun compound. There are some attempts to paraphrase noun compounds. In between the two representation, researchers have also used scoring of template-based paraphrases for assigning abstract labels (Nakov, 2008; Nakov and Hearst, 2013).

2.1. FrameNet

FrameNet (Baker et al., 1998)[2] is a lexical database that shows usage of words in actual text based on annotated examples. It is based on a theory of meaning called Frame Semantics (Fillmore, 1976). The theory claims that meanings of most words can be inferred from a semantic frame: a conceptual structure that denotes the type of event, relation, or entity and the involved participants. For example, the concept of walking involves a person walking (SELF_MOVER), the PATH on which walking occurs, the DIRECTION in which the walking occurs, and so on. In FrameNet, this information is represented by a frame called SELF_MOTION. SELF_MOVER, PATH, DIRECTION, *etc.* are called frame elements (FEs). Such frames are invoked in running text via words known as lexical units (LUs). Continuing the above example, some of the lexical units for the frame SELF_MOTION are *advance, crawl, dash, drive, march, run, walk, etc*. Most LUs are verbs. But, it can be a noun or an adjective, too.

[2] https://framenet.icsi.berkeley.edu

An example sentence in FrameNet annotated data contains a target word along with linked LU, arguments of the target, and an FE for each of the targets. The following is an example of SELF_MOTION frame with *march.v* LU:

> [$_{Time}$ On Jan. 15] [$_{Self_mover}$ up to 20,000 students and pacifists] MARCHED^{Target} [$_{Path}$ through Madrid] .

In this work, we generate noun compounds from the FrameNet annotated sentences, and assign FEs as semantic relations. For example, from the above sentence, we generate *student march*: SELF_MOVER, *pacifist march*: SELF_MOVER and *Madrid march*: PATH.

3. Related Work

For interpretation (as well as other) tasks, we need a representation of semantic relations (SRs) which is based on linguistic intuition. Many inventories of abstract relations have been proposed over the years. But, we found that each inventory had some shortcoming.

Levi (1978)'s study on noun compound generation is the most influential one. The study categorizes noun compounds based on the compounding process as (1) predicate deletion, where a predicate between the components is simply dropped to create a compound, and (2) predicate nominalization, where the head is nominalized form of a verb and modifier is an argument of the verb. They proposed a set of abstract predicates for the former category, but no labels for the latter category.

In contract to Levi (1978)'s study, Warren (1978) proposed a four-level hierarchy of semantic relations based on analysis of the Brown corpus. Nastase and Szpakowicz (2003) extended Warren (1978)'s approach. Their proposed set of relations is also based on Barker and Szpakowicz (1998)'s semantic relations.

Barker and Szpakowicz (1998)'s proposed set of relations based on Levi (1978)'s theory and Warren (1978)'s inventory. They claim that SRs in their inventory are the most widely used and can improve with time. Kim and Baldwin (2005) prepare a dataset for this inventory, but the dataset is highly imbalanced. For instance, out of 20 relations, TOPIC relation has 42% examples and PURPOSE relation has 23% examples in contrast to less than 10 examples of 3 SRs.

Vanderwende (1994) used 13 relations based on the syntactical category and types of questions. Girju et al. (2005) provided another inventory of semantic relation based on Moldovan et al. (2004)'s semantic relation in noun phrases. But, most examples in the dataset uses prepositions as SRs. Also, fourteen of total thirty-five SRs has not any example in their dataset, and seven more SRs has less than 1% examples. For an inventory of SRs, if an SR has no example, then it raises a question on the base of the inventory.

Ó Séaghdha and Copestake (2009) proposed an inventory of SRs based on RDP (recoverable deleted predicates) of Levi (1978). Along with five SRs for compositional NCs – the meaning of the compound is composed of the meaning of the components – they proposed five more SRs for other categories like lexicalized compounds, wrongly tagged compounds. The five compositional SRs has been

further categories in total eleven categories. They have also prepared a dataset with 1443 examples for the five coarse-grained and eleven fine-grained relations.
In addition to the above-mentioned dataset for the general domain, Rosario et al. (2002) proposed an inventory of SRs for medical domain.
Tratz and Hovy (2010) claims that they have a created a new inventory of semantic relations by comparing and consolidating the existing inventories. But, in contract, their inventory creation process is an iterative process to improve inter-annotator agreement. Ponkiya et al. (2016) reports many problems with this inventory.
Unfortunately, these inventories are not used in actual scenarios which need an interpretation of noun compounds. For instance, Balyan and Chatterjee (2014) shows how the interpretation of NCs can help automatic machine translation (MT). But, they didn't use any existing repository.
We now propose our dataset.

4. Proposed Dataset

In this version of the dataset, we generate only noun-noun compounds (noun compounds with only 2 components).

4.1. Dataset Fields

Our dataset contains the following fields:

1. w_1: The first word of the noun compound.
2. w_2: The second word of the noun compound.
3. Frame: ID and name of the frame from which the example was created.
4. FE: ID and name of the frame element from which the example was created.
5. KB05: Label in Kim and Baldwin (2005)'s dataset (hereafter, KB05) (NA if not found).
6. OS09: Label in Ó Séaghdha and Copestake (2009)'s dataset (hereafter, OS09) (NA if not found).
7. TH10: Label in Tratz and Hovy (2010)'s dataset (hereafter, TH10) (NA if not found).
8. Type: Type of noun compound according to Levi's theory.

4.2. Dataset Creation

The dataset was created in two phases: an automated phase where candidate noun compounds are extracted from FrameNet annotated corpus, followed by a manual phase where we annotate each candidate according to Levi's theory.

Automated Phase
In this phase, we take the example sentences for each frame F from FrameNet. Each example sentence is processed as follows:

1. Find the target word T
2. Let S_T be the set of all possible verbal forms and nominalized forms of T
3. For each chunk C annotated with frame element E
   (a) Let H be the head word of the dependency parse of the chunk C
   (b) If $\langle H, W \rangle$ occur in either KB05, OS09, or TH10 (where $W \in S_T$)
   - Output $\langle H, W \rangle$ as candidate NC, along with F, E, and labels in KB05, OS09, or TH10

Consider the example sentence from the frame PROTEST:

> The civil war that began in February with [$_{Degree}$ mass] PROTESTS^{Target} [$_{Issue}$ against Kadafi 's rule] had paralyzed the industry.

Here, the target word is *protest*. The chunk *mass* is annotated with frame element DEGREE, and the word itself is the head word. ⟨*mass protest*⟩ is present only in TH10, thus the process outputs {*mass*, *protest*, PROTEST, DEGREE, NA, NA, COMMUNICATOR_OF_COMMUNICATION}.
Similarly, the chunk *"against Kadafi's rule"* is annotated with frame element ISSUE, and has *rule* as the head word. However ⟨*rule protest*⟩ is not present in either KB05, OS09, or TH10. Thus, we do not consider it as a candidate noun compound.
Note that a candidate noun compound can be generated from more than one frames, thereby having multiple {frame$_i$, frame_element$_i$} labels. In that case we repeat the candidate noun compound, once for each label.

Manual Phase
In this phase, we check the correctness of labels (specially, frame and FE) manually. For example, consider the following annotated sentence:

> If the scientists are right, then a major clue about how [$_{Entity}$ cancer] DEVELOPS^{Target} [$_{Place}$ in children] has been found.

The previous automated phase generates *children development* as a candidate noun compound with the label {COMING_TO_BE, PLACE}. But, in general, usage of *children development* means "*development of children*", and not "*development in children*". We simply drop such noun compounds.
Then, each of these candidate NCs were manually annotated as follows:

- PD: The candidate is an NC created through predicate deletion. Examples: *orange juice* (juice made of orange), *cricket bat* (bat made for cricket), *etc.*
- PN: The candidate is an NC created through predicate nominalization. Examples: *student protest*, *gender segregation*, *etc.*

4.3. Statistics

The final dataset contains 2,600 noun compounds, formed through the combination of 818 modifiers and 806 heads. The set of unique modifiers and heads contains 1401 words, with 223 words appearing both as modifiers and as heads. The total number of unique frames is 409. The total number of unique frame + frame element combinations is 893.

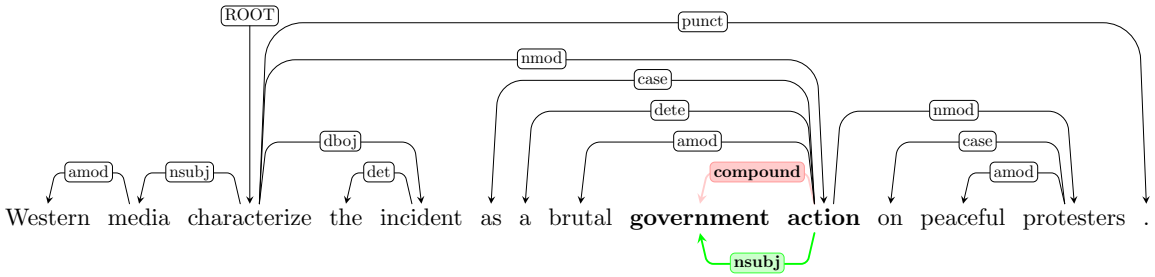


Figure 1: An example of dependency parsing output from CoreNLP. For *government action*, the CoreNLP assigns `compound` (red-colored label). Using our approach, we can revise it as `nsubj` (in green color).

5. Discussion

Here, we discuss some of the advantages of creating the dataset in the proposed manner.

- In FrameNet, hierarchy of FE (defined in addition to FE relations) can help generalization of the relations. For instance, our dataset has *student protest*: PROTESTER as an example. Using FE relations (*e.g.*, PROTESTER of PROTEST –ISA→ AGENT of INTENTIONALLY_ACT), we can infer *student protest*: AGENT.

- As each noun compound has been annotated with frame and FE, details of the corresponding frame and FE helps in paraphrasing the NC.

- On an average, the number of frames that can be invoked by a head of an NC is not too high (3.27 for our dataset). This limits the number of corresponding FEs, thereby reducing ambiguity. So, even though we have thousands of FEs in FrameNet, the actual search space for an NC is relatively small.

- Example of FEs from FrameNet annotated data can help in disambiguation of FEs for a given NC. For example, while labeling *fee-hike protest* with FEs of PROTEST frame, there may be confusion between ISSUE and PURPOSE. In such case, examples of those FEs from FrameNet annotated data can help in disambiguation.

- We intended to create a dataset that can be used as a gold standard for further research in noun compound interpretation. Therefore, we tried to reduce false positives as much as possible, *i.e.,* ensure that a noun compound included in the dataset is labeled with correct frame and frame element. In the process, we decided to remove certain examples where the assigned label seemed to be a corner case. Consider the following example sentence:

  $[_{Entity}$ The theater] PRESENTS^{Target} $[_{Phenomenon}$ sky shows and IMAX films].

  Our automated phase generates *theatre presentation* as a candidate noun compound, with the label {CAUSE_TO_PERCEIVE, ENTITY}, implying that the presentation is by the theatre. However, this seems like a corner case, as a presentation is more likely to be at the theatre. Thus, we do not include such examples in our dataset.

- This exercise also lead to fixing errors present in other datasets. For instance, Ó Séaghdha and Copestake (2009) states that their dataset contains noun compounds created only by predicate deletion. However, we observed that out of the 145 noun compounds that matched with their dataset, 25 were of the type predicate nominalization. For instance, they label *questionnaire reply* as predicate deletion, but it is an example of predicate nominalization.

6. Potential Applications

A direct application of this dataset that can benefit the community is its potential to enrich dependecy parsing. We believe that a dependency parser can be modified to include noun compound detection, and then use the appropriate frame element to improve erroneously labeled arcs.
The dependency parser from CoreNLP tool (Manning et al., 2014)[3] tags the dependecy between components of a compounds as `compound`. There are 50 different pairs (of parts of speech) which are connected with `compound` relation. Among instances of those 50 types, NOUN+NOUN appear around 53% times.[4]
For NOUN+NOUN compounds, we can extract such compounds, reparse the dependency (if required; in case of more than two consecutive nouns) and tag it with more meaningful dependency labels than the `compound`.
For instance, as shown in Fig. 1, CoreNLP tags *government action* as *government* $\xleftarrow{compound}$ *action*. The knowledge that *government* is an AGENT in INTENTIONALLY_ACT frame (invoked by *action*), can help the parser to correctly parse it as *government* $\xleftarrow{nsubj}$ *action*.[5]

7. Conclusion and Future Work

In this paper, we proposed a dataset for noun compound interpretation. The dataset is linguistically grounded using Levi's theory. It uses frames and frame elements of FrameNet as the semantic relation inventory. We took this steps with the goal of creating a standardized dataset, the

[3] `https://stanfordnlp.github.io/CoreNLP`
[4] `http://universaldependencies.org/treebanks/en/en-dep-compound.html`
[5] As per guideline of `nsubj` relation. `http://universaldependencies.org/u/dep/nsubj.html`

lack of which is severely affecting research in noun compound interpretation. Our dataset contains 2,600 examples. Each noun compound is annotated according to the type of noun compound (predicate deletion vs. predicate nominalization), the frame and frame element through which the noun compound was created in the first place, and its label in three other datasets. We also discussed how this dataset could be useful to improve dependency parsers.

In the future, we will extend this dataset to include more noun compounds. Currently, we severely restricted our dataset size by considering only those noun compounds that occur in other datasets as valid noun compounds. For example, out of the 259 candidates generated automatically, a manual investigation suggested 58 valid noun compounds. However, our restrictions led to the inclusion of only six noun compounds. Thus, there is a scope for including many more noun compounds. We will also investigate whether the set of frame elements applicable to noun compounds is a proper subset of the entire set of frame elements.

8. Acknowledgments

We thank CFILT members at IIT Bombay for their valuable comments and suggestions. This work is funded by Tata Consultancy Services Limited (TCS) under NGIE (Next Generation Information Extraction) with project code 13TCSIRC001.

9. Bibliographical References

Ahn, K., Bos, J., Kor, D., Nissim, M., Webber, B. L., and Curran, J. R. (2005). Question answering with QED at TREC 2005. In *Text REtrieval Conference*.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *COLING-ACL '98: Proceedings of the Conference*, pages 86–90, Montreal, Canada.

Baldwin, T. and Tanaka, T. (2004). Translation by machine of complex nominals: Getting it right. In *Proceedings of the Workshop on Multiword Expressions: Integrating Processing*, pages 24–31. Association for Computational Linguistics.

Balyan, R. and Chatterjee, N. (2014). Translating noun compounds using semantic relations. *Computer Speech & Language*.

Barker, K. and Szpakowicz, S. (1998). Semi-automatic recognition of noun modifier relationships. In *Proceedings of the 17th international conference on Computational Linguistics-Volume 1*, pages 96–102. Association for Computational Linguistics.

Butnariu, C., Kim, S. N., Nakov, P., Ó Séaghdha, D., Szpakowicz, S., and Veale, T. (2009). Semeval-2010 task 9: The interpretation of noun compounds using paraphrasing verbs and prepositions. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 100–105. Association for Computational Linguistics.

Fillmore, C. J. (1976). Frame semantics and the nature of language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, 280(1):20–32.

Girju, R., Moldovan, D., Tatu, M., and Antohe, D. (2005). On the semantics of noun compounds. *Computer speech & language*, 19(4):479–496.

Hendrickx, I., Nakov, P., Szpakowicz, S., Kozareva, Z., Séaghdha, D. O., and Veale, T. (2013). Semeval-2013 task 4: Free paraphrases of noun compounds. *Atlanta, Georgia, USA*, page 138.

Kim, S. N. and Baldwin, T. (2005). Automatic interpretation of noun compounds using wordnet similarity. In *Natural Language Processing–IJCNLP 2005*, pages 945–956. Springer.

Levi, J. N. (1978). *The syntax and semantics of complex nominals*. Academic Press New York.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Moldovan, D., Badulescu, A., Tatu, M., Antohe, D., and Girju, R. (2004). Models for the semantic classification of noun phrases. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, pages 60–67. Association for Computational Linguistics.

Nakov, P. I. and Hearst, M. A. (2013). Semantic interpretation of noun compounds using verbal and other paraphrases. *ACM Transactions on Speech and Language Processing (TSLP)*, 10(3):13.

Nakov, P. (2008). Noun compound interpretation using paraphrasing verbs: Feasibility study. In *Artificial Intelligence: Methodology, Systems, and Applications*, pages 103–117. Springer.

Nastase, V. and Szpakowicz, S. (2003). Exploring noun-modifier semantic relations. In *Fifth International Workshop on Computational Semantics (IWCS-5)*, pages 285–301.

Ó Séaghdha, D. and Copestake, A. (2009). Using lexical and relational similarity to classify semantic relations. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 621–629. Association for Computational Linguistics.

Ponkiya, G., Bhattacharyya, P., and Palshikar, G. K. (2016). On why coarse class classification is a bottleneck for noun compound interpretation. In *13th International Conference on Natural Language Processing*, page 293.

Rosario, B., Hearst, M. A., and Fillmore, C. (2002). The descent of hierarchy, and selection in relational semantics. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 247–254. Association for Computational Linguistics.

Tratz, S. and Hovy, E. (2010). A taxonomy, dataset, and classifier for automatic noun compound interpretation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 678–687. Association for Computational Linguistics.

Vanderwende, L. (1994). Algorithm for automatic interpretation of noun sequences. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*,

pages 782–788. Association for Computational Linguistics.

Warren, B. (1978). Semantic patterns of noun-noun compounds. *Acta Universitatis Gothoburgensis. Gothenburg Studies in English Goteborg*, 41:1–266.

Structured Interpretation of Temporal Relations

Yuchen Zhang and Nianwen Xue
Brandeis University
415 South Street, Waltham, MA
yuchenz@brandeis.edu, xuen@brandeis.edu

Abstract

Temporal relations between events and time expressions in a document are often modeled in an unstructured manner where relations between individual pairs of time expressions and events are considered in isolation. This often results in inconsistent and incomplete annotation and computational modeling. We propose a novel annotation approach where events and time expressions in a document form a dependency tree in which each dependency relation corresponds to an instance of temporal anaphora where the antecedent is the parent and the anaphor is the child. We annotate a corpus of 235 documents using this approach in the two genres of news and narratives, with 48 documents doubly annotated. We report a stable and high inter-annotator agreement on the doubly annotated subset, validating our approach, and perform a quantitative comparison between the two genres of the entire corpus. We make this corpus publicly available.

Keywords: Temporal Relation, Dependency Structure, Data Annotation

1. Introduction

Understanding temporal relations between events and temporal expressions in a natural language text is a fundamental part of understanding the meaning of text. Automatic detection of temporal relations also enhances downstream natural language applications such as story timeline construction, question answering, text summarization, information extraction, and others. Due to its potential, temporal relation detection has received a significant amount of interest in the NLP community in recent years.

Most of the research attention has been devoted to defining the "semantic" aspect of this problem – the identification of a set of semantic relations between pairs of events, between an event and a time expression, or between pairs of time expressions. Representative work in this vein includes TimeML (Pustejovsky et al., 2003a), a rich temporal relation markup language that is based on and extends Allen's Interval Algebra (Allen, 1984). TimeML has been further enriched and extended for annotation in other domains (O'Gorman et al., 2016; Styler IV et al., 2014; Mostafazadeh et al., 2016). Corpora annotated with these schemes (Pustejovsky et al., 2003b; O'Gorman et al., 2016) are shown to have stable inter-annotator agreements, validating the temporal relations proposed in the TimeML. Through a series of TempEval shared tasks (Verhagen et al., 2007a; Verhagen et al., 2010a; UzZaman et al., 2012; Bethard et al., 2015; Bethard et al., 2016; Bethard et al., 2017), there has also been significant amount of research on building automatic systems aimed at predicting temporal relations.

Less attention, however, has been given to the "structural" aspect of temporal relation modeling – answering the question of which other events or time expressions a given time expression or event depends on for the interpretation of its temporal location. Having an answer to this question is important to both linguistic annotation and computational modeling. From the point of view of linguistic annotation, without an answer to this question, an annotator is faced with the choice of: (i) labeling the relation between this event/time expression with all other events and time expressions, or (ii) choosing another event/time expression with which the event/time expression in question has the most salient temporal relation. (i) is impractical for any textual document that is longer than a small number of sentences. Without a solid linguistic foundation, adopting (ii) could lead to inconsistent and incomplete annotation as annotators may not agree on which temporal relations are the most salient.

From a computational perspective, without knowing which time expressions and events are related to each other, an automatic system has to make a similar choice to predict the temporal relations between either all pairs of events and time expressions, or only a subset of the temporal relations. If it chooses to do the former, there will be $\binom{n}{2}$ pairs for n events and time expressions. Not only is this computationally expensive, there could be conflicting predictions due to the transitivity of temporal relations (e.g. "A before B" and "B before C" imply "A before C", which a pair-wise approach may make conflicting predictions) and additional steps are necessary to resolve such conflicts (Chambers and Jurafsky, 2008; Yoshikawa et al., 2009; Do et al., 2012).

We propose a novel annotation approach to address this dilemma. Specifically we propose to build a dependency tree structure for the entire document where the nodes of the tree are events and time expressions, as well as a few pre-defined "meta" nodes that are not anchored to a span of text in the document The building blocks of this dependency structure are pairs of events and time expressions in which the *child* event/time expression depends on its *parent* event/time expression for its temporal interpretation. The dependency relation is based on the well-established notion of temporal anaphora where an event or time expression can only be interpreted with respect to its reference time (Reichenbach, 1947; Partee, 1973; Partes, 1984; Hinrichs, 1986; Webber, 1988; Bohnemeyer, 2009). In each dependency relation in our dependency structure, the parent is the *antecedent* and the child is the *anaphor* that depends on its antecedent for its temporal interpretation. Consider the following examples:

1. He arrived on Thursday. He got here at 8:00am.

2. He arrived at school, walked to his classroom, and then the class began.

In (1), the antecedent is "Thursday" while "8:00am" is the anaphor. We won't know when exactly he arrived unless we know the 8:00am is on Thursday. In this sense, "8:00am" depends on "Thursday" for its temporal interpretation. We define the *antecedent of an event as a time expression or event with reference to which the temporal location of the anaphor event can be most precisely determined.* In (2), the antecedent for the event "the class began" is "walked to his classroom" in the sense that the most specific temporal location for the event "the class began" is after he walked to the classroom. Although "the class began" is also after "he arrived at school", the temporal location we can determine based on that is not as precise.

In order for the events and time expressions to form a dependency tree, one key assumption we make is *there is exactly one antecedent event/time expression for each anaphor*. This ensures that there is exactly one head for each dependent, a key formal condition for a dependency tree.

Once this dependency structure is acquired, manually or automatically, additional temporal relations may be inferred based on the transitive property of temporal relations, but we argue that this dependency structure is an intuitive starting point that makes annotation as well as the computational modeling more constrained and tractable.

We annotate a corpus of 235 documents with temporal dependency structures, with 48 documents double-annotated to evaluate inter-annotator agreement. The annotated data are chosen from two different genres, new data from the Xinhua newswire portion of the Chinese TreeBank (Xue et al., 2005) and Wikipedia news data used for CoNLL Shared Task on Shallow Discourse Parsing in 2016 (Xue et al., 2016), and narrative story data from Grimm fairy tales. The two genres are chosen because the temporal structure of texts from those two genres unfolds in very different ways: news reports are primarily in report discourse mode in the sense of (Smith, 2003) while Grimm fairy tales are primarily in narrative mode and time advances in those two genres in very different ways, as we will discuss in more detail in Section 4.2.. We report a stable and high inter-annotator agreement for both genres, which validates the intuitiveness of our approach. This corpus is publicly available.[1]

The main contributions of this paper are:

- We propose a novel and comprehensive temporal dependency structure to capture temporal relations in text.
- We analyze different types of time expressions in depth and propose a novel definition, as far as we know, for the reference time of a time expression (§3.2.1.).
- We produce an annotate corpus with this temporal structure that covers two very different genres, news and narratives and achieved high inter-annotator agreements for each genre. An analysis of the annotated data show that temporal structures are very genre-dependent, a conclusion that has implications for how the temporal structure of a text can be parsed.

In the next few sections, we will briefly discuss related work (§2.), describe our annotation scheme (§3.), and present our annotation experiments (§4.). We summarize our work in §5.

2. Related Work

Using a dependency structure to represent temporal relations in a document has been proposed before (Kolomiyets et al., 2012), but our work is more comprehensive and linguistically grounded in the following ways. First, their dependency structure is based on events, to the exclusion of time expressions. Time expressions are a strong source of temporal location information for events and excluding them will result in incomplete temporal structures. We cover both events and time expressions to form a complete temporal structure for a text. Second, they exclude stative events such as modalized events, while we provide a more complete temporal structure that include stative events. Third, although they link events in a text to form a dependency structure, they do not explicitly spell out the linguistic basis for the temporal dependencies and annotators are only instructed to identify the most plausible parent for each event. In contrast, we explicitly specify how antecedents of events or time expressions are determined based on a long line of theoretical and computational linguistic research (Reichenbach, 1947; Partee, 1973; Partes, 1984; Hinrichs, 1986; Webber, 1988; Bohnemeyer, 2009; Wuyun, 2016) and these specifications are given to annotators as guidelines when they annotated the data. And lastly, their annotation work is only performed on children's stories (narrative data), while our annotated corpus covers both news and narrative genres. Annotating two different genres is crucial for us to show that the temporal structure for the two genres are very different, an observation that has implication for automatic parsing strategies.

3. Temporal Structure Annotation Scheme

In our annotation scheme, a temporal dependency tree structure is defined as a 4-tuple (T, E, N, L), where T is a set of time expressions, E is a set of events, and N is a set of pre-defined "meta" nodes not anchored to a span of text in the document. T, E, N form the nodes in the dependency structure, and L is the set of edges in the tree. Detailed descriptions for each set are in the following subsections, followed by some examples.

3.1. Nodes in the temporal dependency tree

The nodes in a temporal dependency tree includes time expressions, events, and a set of pre-defined nodes. We elaborate on each type of nodes below:

3.1.1. Time Expressions

TimeML (Pustejovsky et al., 2003a) treats all temporal expressions as markable units and classifies them into three categories: fully specified temporal expressions ("June 11,

[1] https://github.com/yuchenz/structured_temporal_relations_corpus

1989", "Summer, 2002"); underspecified temporal expressions ("Monday", "next month", "last year", "two days ago"); and durations ("three months", "two years"). The purpose of our dependency structure annotation is to find all time expressions that can serve as a reference time for other events or time expressions. We observe that while the first two TimeML categories of time expressions can serve as reference times, the last category, "durations", typically don't serve as reference times, unless they are modified by expressions like "ago" or "later". For example, the "10 minutes" in (3) can serve as a reference time because it can be located in a timeline as a duration from 8:00 to 8:10, while the "10 minutes" in (4) can't serve as a reference time.

3. He arrived at 8:00am. 10 minutes later, the class began.

4. It usually takes him 10 minutes to bike to school.

Therefore, in our annotation scheme, we make the distinction between time expressions that can be used as reference times and the ones that cannot. The former includes fully specified temporal expressions, underspecified temporal expressions, as well as time durations modified by "later" or "ago". The latter include unmodified durations. In our annotation, only the former are considered to be valid nodes in our time expression set T.

3.1.2. Events

We adopt a broad definition of events following Pustejovsky et al. (2003a), where "an event is any situation (including a process or a state) that happens, occurs, or holds to be true or false during some time point (punctual) or time interval (durative)." Based on this definition, unless stated explicitly, events for us include both eventive and stative situations. Adopting the minimal span approach along the lines of (O'Gorman et al., 2016), only the headword of an event is labeled in actual annotation. Since different events tend to have different temporal behaviors in how they relate to other events or time expressions(Wuyun, 2016), we also assign a coarse event classification label to each event before linking them to other other events or time expressions to form a dependency structure. Adapting the inventory of situation entity types from Smith (2003) and from Zhang and Xue (2014), we define the following eight categories for events.

- An **Event** is a process that happens or occurs. It is the only eventive type in this classification set that advances the time in a text. An example event is "I *went* to school yesterday".
- A **State** is a situation that holds during some time interval. It is stative and describes some property or state of an object, a situation, or the world. For example, "she *was* very shy" describes a state.

The remaining event types are all statives that describe an eventive process.

- A **Habitual** event describes the state of a regularly repeating event, as in "I *go* to the gym three times a week".
- An **Ongoing** event describes an event in progress, as in "she was *walking* by right then".
- A **Completed** event describes the completed state of an event, as in "She's *finished* her talk already".
- A **Modalized** event describes the capability, possibility, or necessity of an event, as in "I have to *go*".
- A **Generic Habitual** event is a Habitual event for generic subjects, as in "The earth *goes* around the sun".
- A **Generic State** is a state that hold for a generic subject, as in "Naked mole rats don't *have* hairs".

All valid events from a document, represented by their headwords, form the event set E.

3.1.3. Pre-defined Meta Nodes

In order to provide valid reference times for all events and time expressions, and to form a complete tree structure, we designate the following pre-defined nodes for the set N.
ROOT is the root node of the temporal dependency tree and every document has one ROOT node. It is the parent of (i) all other pre-defined nodes, and (ii) absolute concrete time expressions (Example 11, see §3.2.1. for more on time expression classification). The meta node **DCT** is the Document Creation Time, a.k.a. Speech Time. Following Pustejovsky et al. (2003a), we define meta nodes **PRESENT_REF, PAST_REF, FUTURE_REF** as the general reference times respectively for generic present, past, and future times. Lastly, **ATEMPORAL** is designated as the parent node for atemporal events, such as timeless generic statements (Example 12).
These generic reference times are necessary for time expressions and events that don't have a more specific reference time in the text as their parents. For example, it is common to start a narrative story with a few descriptive statements in past tense without a specific time (Example 5), or a general time expression referring to the past (Example 6). Both cases take "Past_Ref" as their parent.

5. It was a snowy night. [Past_Ref]
6. Once upon the time, ... [Past_Ref]

It is worth noting that "DCT" and "Present_Ref" are not interchangeable. "DCT" is usually a very specific timestamp such as "2018-02-15:00:00:00", while "Present_Ref" is a general temporal location reference. We use "DCT" as the parent for relative concrete time expressions (example 10), and for vague time expressions, their antecedent is "Present_Ref" (Example 7). See §3.2.1. for more details on time expression classification.

7. China annual economic output results have grown increasingly smooth in recent years. [Present_Ref]
8. Economists who try to estimate actual growth tend to come up with lower numbers. [Present_Ref]
9. China will remain a trade partner as important to Japan as the United States in the future. [Future_Ref]
10. The economy expanded 6.9 percent last year. [DCT]
11. A trend of gradual growth began in 2011. [ROOT]
12. The earth goes around the sun. [Atemporal]

3.2. Edges in the temporal dependency tree

As we discussed above, each dependency relation consists of an antecedent and an anaphor, with the antecedent being the parent and the anaphor being the child. Based on the well-established notion of temporal anaphora (Reichenbach, 1947; Partee, 1973; Partes, 1984; Hinrichs, 1986; Webber, 1988; Bohnemeyer, 2009), we assume each event or time expression in the dependency tree has only one antecedent (i.e. one reference time), which is necessary to form the dependency tree. In this section, we will first discuss what can serve as a reference time for time expressions in our annotation scheme, then we will discuss what can be a reference time for events. All links between events/time expressions and their reference times form our link set L.

3.2.1. Reference Times for Time Expressions

In previous work such as the TimeBank (Pustejovsky et al., 2003a) the temporal relations between time expressions are annotated with temporal ordering relations such as "before", "after", or "overlap" just like events in a pairwise without considering the dependencies between them. For example, consider the three time expressions "2003", "March", and "next year" in (13), using a pair-wise annotation approach, three temporal relations will be extracted:

13. The economy expanded 6.6 percent in $\underline{2003}_{t1}$, reaching its peak 7.1 percent in $\underline{\text{March}}_{t2}$. The growth rate doubled in the $\underline{\text{next year}}_{t3}$.

(2003, includes, March)
(2003, before, next year)
(March, before, next year)

We argue the sole purpose for annotating temporal relations between time expressions is to properly "interpret" time expressions that "depend" on another time expression for their interpretation. In the context of time expressions, "interpretation" means normalizing time expressions in a format that allows the ordering between the time expressions to be automatically computed. Time expression normalization is necessary in many applications. For example, in a question answering system, our model needs to be able to answer "2004" when it is asked "Which year did China's export rate double?", instead of answering "next year" which is uninterpretable taken out of the original context. In order for the time expressions to be properly interpreted, it is important to annotate the dependency between "March" and its reference time "2004" because the former depends on the latter for its interpretation. Similarly, it is also important to establish the dependency between "next year" and its reference time "2004" as we won't know which year is "next year" until we know it is with reference to "2004". With the these dependencies identified and the time expressions normalized, the temporal relations between all pairs of time expressions in a text can be automatically computed, and explicit annotation of the temporal relation between all pairs of time expressions will not be necessary. For example, with "March" normalized to "2003-03" and "next year" normalized to "2004", the relation between 2003-03 and 2004 can be automatically computed. We argue that this notion of reference time for time expressions is intuitive and easy to define. Annotating temporal dependency between time expressions is also more efficient than annotating the temporal ordering between all pairs of time expressions.
Based on these considerations, we propose a novel definition of the reference time for time expressions:

Definition 1 *Time expression A is the reference time for time expression B, if B depends on A for its temporal location interpretation.*

In other words, a time expression can depend solely on its reference time to be interpreted and normalized. We use a generic **Depend-on** label for these relations. Take (1) as an example, annotators only need to determine that the temporal interpretation of '8am" depends on "Thursday". With "Thursday" normalized to, for example, "2003-04-05", we can then compute a normalized time "2003-04-05:08:00:00" for "8am", and easily compute the temporal ordering between them: ("2003-04-05" includes "2003-04-05:08:00:00").
We now consider the question of what types of nodes can serve as the reference time or antecedent for a time expression. First, since a time expression relies on its reference time for its temporal interpretation, naturally an event cannot serve as its reference time. Second, since some time expressions (e.g., "2003") can be interpreted (and normalized) on its own without any additional information, while others can not, further categorization of time expressions is needed to precisely specify which time expressions need a reference time for their interpretation and which do not, and what time expressions can serve as reference times and which do not.
First, we make the distinction between Concrete and Vague time expressions. A **Concrete Time Expression** is a time expression that can be located onto a timeline as an exact time point or interval, e.g. "June 11, 1989", "today". Their starting and ending temporal boundaries on the timeline can be determined. A **Vague Time Expression** (e.g., "nowadays", "recent years", "once upon the time") expresses the concept of (or a period in) general past, general present, or general future, without specific temporal location boundaries. The reference time for Vague time expressions are the pre-defined nodes PRESENT_REF, PAST_REF, and FUTURE_REF.
Concrete time expressions are further classified into **Absolute Time Expressions** and **Relative Time Expressions**, corresponding to fully-specified ("June 11, 1989", "Summer, 2002") and underspecified temporal expressions ("Monday", "Next month", "Last year", "Two days ago") in Pustejovsky et al. (2003a) respectively. Relative concrete time expressions take either DCT or another concrete time expression as their reference time. Absolute concrete time expressions can be normalized independently and don't need a reference time. Therefore, we stipulate that their parent in the dependent tree is the pre-defined node ROOT. For example, "1995", "20th century" are absolute concrete time expressions, while "today", "last year", "the future three years", "January 20th", "next Wednesday" are relative concrete time expressions, and "recent years", "in the

Taxonomy				Examples	Possible Reference Times
Time Expressions	Locatable Time Expressions	Concrete	Absolute	May 2015	ROOT
			Relative	today, two days later	DCT, another Concrete
		Vague		nowadays	Present/Past/Future_Ref
	Unlocatable Time Expressions			every month	-

Table 1: Taxonomy of time expressions in our annotation scheme, with examples and possible reference times.

past a few years", "nowadays", "once upon the time" are vague time expressions.

An example of a concrete relative time expression having a concrete absolute temporal expression as its reference time is given in (13) . Consider the time expression "March". In order to be able to interpret it and normalize it into a valid temporal location on a timeline, we need to establish "2003" is its reference time. Then it is possible to normalize it into a formal representation as "2003-03".

Lastly, in order to form a complete tree structure, all pre-defined nodes (except for ROOT) take ROOT as their parent. A complete taxonomy of time expressions in our annotation scheme with examples and their possible reference times is illustrated in Table 1.

3.2.2. Reference Times for Events

The reference time for an event is a time expression or pre-defined node or another event with respect to which the most specific temporal location of the event in question can be determined. Unlike time expressions, for which the possible reference times can only be other time expressions or pre-defined nodes, the possible reference times for events are not as restrictive and can be any of the three categories. The dependency relation that we use to characterize the relationship between the reference time / antecedent and an event is a temporal relation between them.

Definition 2 *Time expression/pre-defined node/event A is the reference time for event B, if A is the most specific temporal location which B depends on for its own temporal location interpretation.*

There has been significant amount of work attempting to characterize the temporal relationship between events, and between time expressions and events. One of the first attempts to model temporal relations is Allen's Interval Algebra theory (Allen, 1984). They introduced a set of distinct and exhaustive temporal relations that can hold between two time intervals, which are further adapted and extended in Pustejovsky et al. (2003a), THYME (Styler IV et al., 2014), etc. A detailed comparison of these sets can be found in Mostafazadeh et al. (2016). Mindful of the need to produce consistent annotation, and in line with the practice of some prior work such as the TempEval evaluations (Verhagen et al., 2007b; Verhagen et al., 2009; Verhagen et al., 2010b) we adopt a simplified set of 4 temporal relations to characterize the relationship between an event and its reference time. The set of temporal relations we use with their mappings to their corresponding TimeML temporal relations are shown shown in Table 2.

Although an event can in principle take a time expression, another event, or a pre-defined node as its antecedent, different types of events have different tendencies as to the types of antecedents they take. An eventive event usually takes either a time expression or another eventive event as its reference time. They advance the time in the narrative of a text, so it usually has a (time expression, Includes, event) relation with its antecedent, or a (event, Before, event) relation. For example, in (1) the time expression "Thursday" has "Includes" relation with the event "arrived", and the time expression "8:00am" has an "Includes" relation with the event "got here". And in (2) the event "arrived" has a "Before" relation with the event "walked".

Our Scheme	TimeML
Before	Before, IBefore
After	-
Overlap	Ends, Begins, Identity, Simultaneous
Includes	During

Table 2: Our temporal relation set for events with mappings to TimeML's set.

A stative event can take a time expression, another event, or a pre-defined node (except for ROOT) as its reference time. It generally describes a state that holds during the time indicated by its antecedent time expression, event, or generic time. It usually has an "Overlap" relation with their reference times. For example, in (4) the event "takes" is a stative Habitual event, which describes a state of the present situation for "him", so its reference time is the pre-defined node "Present_Ref", and has an "Overlaps" relation with "Present_Ref".

An eventive event rarely takes a stative event as its reference time. As discussed above, we pick the most specific temporal location as the reference time for an event. Since more specific temporal locations are usually available (such as another eventive event), a stative event rarely serves as the reference time for an eventive event.

Readers are referred to our more detailed guidelines[2] on time expression and event recognition, classification, and reference time annotation, which details basic principles for specific cases and discusses extra rules for special scenarios.

3.3. Full Temporal Structure Examples

We present a full example temporal dependency structure for a short news report paragraph (14), as illustrated in Figure 1, and another one for a narrative passage (15), as illustrated in Figure 2. Subscript e denotes eventive events, t denotes time expressions, and s denotes stative events. Unlabeled edges are "depend-on" relations.

[2]https://github.com/yuchenz/structured_temporal_relations_corpus

The two examples provide a sharp contrast between the typical temporal dependency structures for newswire documents and narrative stories, with the former generally having a flat and shallow structure and the latter having a narrow and deep structure.

14. Jorn Utzon, the Danish architect who designed the Sydney Opera House, has $\underline{\text{died}}_{e1}$ in Copenhagen. $\underline{\text{Born}}_{e2}$ in $\underline{1918}_{t1}$, Mr Utzon was $\underline{\text{inspired}}_{e3}$ by Scandinavian functionalism in architecture, but $\underline{\text{made a number of inspirational trips}}_{e4}$, including to Mexico and Morocco. In $\underline{1957}_{t2}$, Mr Utzon's now-iconic shell-like design for the Opera House unexpectedly $\underline{\text{won}}_{e5}$ a state government competition for the site on Bennelong Point on Sydney Harbour. However, he $\underline{\text{left}}_{e6}$ the project in $\underline{1966}_{t3}$. His plans for the interior of the building $\underline{\text{were not completed}}_{s1}$. The Sydney Opera House $\underline{\text{is}}_{s2}$ one of the world's most classic modern buildings and a landmark Australian structure. It was $\underline{\text{declared}}_{e7}$ a UNESCO World Heritage site $\underline{\text{last year}}_{t4}$. [3]

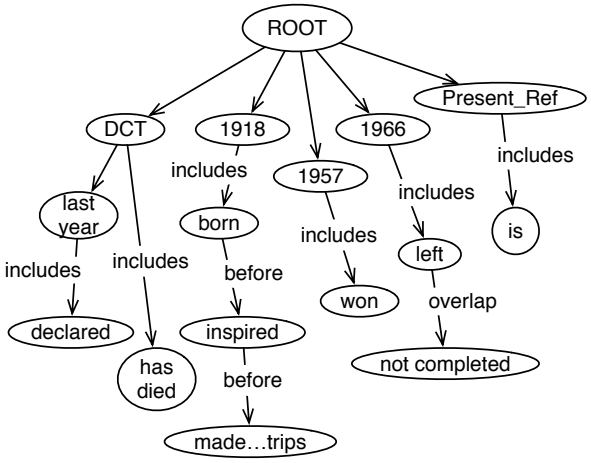


Figure 1: An example full temporal dependency structure for news paragraph (14).

15. There $\underline{\text{was}}_{s1}$ $\underline{\text{once}}_{t1}$ a man who had seven sons, and still he $\underline{\text{had}}_{s2}$ no daughter, however much he $\underline{\text{wished}}_{s3}$ for one. At length his wife again $\underline{\text{gave}}_{e1}$ him hope of a child, and when it $\underline{\text{came}}_{e2}$ into the world it $\underline{\text{was}}_{s4}$ a girl. The joy $\underline{\text{was}}_{s5}$ great, but the child $\underline{\text{was}}_{s6}$ sickly and small, and had to be privately $\underline{\text{baptized}}_{s7}$ on account of its weakness. The father $\underline{\text{sent}}_{e4}$ one of the boys in haste to the spring to fetch water for the baptism. The other six $\underline{\text{went}}_{e5}$ with him, and as each of them wanted to be first to fill it, the jug $\underline{\text{fell}}_{e6}$ into the well. There they $\underline{\text{stood}}_{s8}$ and did not $\underline{\text{know}}_{s9}$ what to do, and none of them dared to $\underline{\text{go}}_{s10}$ home. As they still did not return, the father $\underline{\text{grew}}_{e7}$ impatient, and $\underline{\text{said}}_{e8}$, they have certainly $\underline{\text{forgotten}}_{s11}$ it while playing some game, the wicked boys. He $\underline{\text{became}}_{e9}$ afraid that the girl would have to die without being baptized.[4]

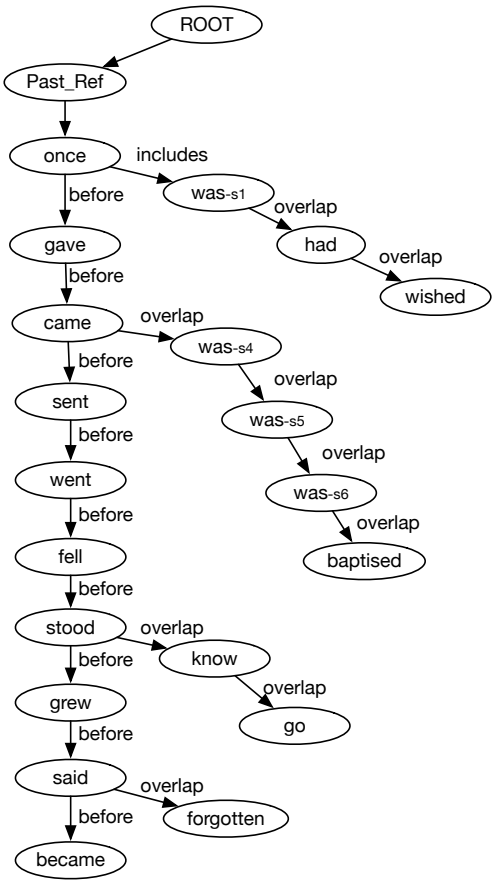


Figure 2: An example full temporal dependency structure for narrative paragraph (15).

3.4. Annotation Process

We use a two-pass annotation process for this project. In the first pass, annotators do temporal expression recognition and classification, and then reference time resolution for all time expressions. The purpose of this pass is to mark out all possible reference times realized by time expressions and recognize their internal temporal relations, in order to provide a backbone structure for the final dependency tree. In the second pass, event recognition and classification, and then reference time resolutions for all events are annotated, completing the final temporal dependency structure of the entire document.

4. Annotation Analysis

4.1. Corpus

A corpus of 115 news articles, sampled from Chinese TempEval2 data (Verhagen et al., 2010a) and Wikinews data, [5] and 120 story articles, sampled from Chinese Grimm fairy tales, [6] are compiled and annotated. 20% of the documents are double annotated by native Chinese speakers. Table 4 presents the detailed statistics. High and stable inter-annotator agreements are reported in Table 5.

On event annotation, our work is comparable to the annotation work in Kolomiyets et al. (2012). They report inter-annotator agreements of 0.86, 0.82, and 0.70 on event

[3] From a news report on *The Telegraph*

[4] From Grimm's fairy tale *The Seven Ravens*

[5] zh.wikinews.org

[6] https://www.grimmstories.com/zh/grimm_tonghua/index

		Pre-defined Node	Time Expression	Eventive Event	Stative Event
News	Time Expression	1078 (92%)	89 (8%)	0	0
	Eventive Event	103 (9%)	290 (26%)	716 (65%)	0
	Stative Event	149 (8%)	192 (11%)	432 (24%)	1029 (57%)
Narratives	Time Expression	95 (83%)	20 (17%)	0	0
	Eventive Event	20 (0%)	25 (1%)	4875 (99%)	0
	Stative Event	25 (1%)	74 (2%)	1655 (49%)	1612 (48%)

Table 3: Distribution of parent types for each child type. Rows represent child types, and columns represent parent types.

		Docs	Sent	Timex	Events
News	Single	91	2,271	901	3,759
	Double	24	570	266	1,048
	Total	115	2,841	1,167	4,807
Narratives	Single	96	3,034	91	9,024
	Double	24	628	40	1,952
	Total	120	3,662	131	10,976

Table 4: Corpus annotation statistics. (*Timex* stands for time expressions.)

recognition, unlabeled relations, and labeled relations respectively on a narrative data. We argue that the comparable or better agreements on narratives as shown in Table 5 show that incorporating the notion of linguistic temporal anaphora helps annotators make more consistent decisions. High (above 90%) agreements on time expression recognition and parsing indicate that our new definition of the reference time for time expressions is clear and easy for annotators to operate on. While event annotations receive lower agreements than time expressions on both genres, they are in general easier on news than on narratives, especially for event reference time resolution and edge labeling.

		News	Narratives
Timex	Recognition	.97	1.
	Classification	.95	.94
	Parsing	.93	.94
Event	Recognition	.94	.93
	Classification	.77	.75
	Relations (unlabeled)	.86	.83
	Relations (labeled)	.79	.72

Table 5: Inter-Annotator Agreement F scores on 20% of the annotations.

4.2. Analysis Across Different Genres

During our annotation, we discovered that narrative texts are very different from news with respect to their temporal structures. First, news texts are usually organized with abundant temporal locations, while narrative texts tend to start with a few temporal locations setting the scene and proceed with only events. As shown in Table 4, around 20% (1166) nodes in the news data are time expressions and 80% (4805) are event nodes, while in the narrative data the ratio of time expressions to events are 0.01%/99.99% (132/10314).

Second, descriptive statements are more common in news data than in narratives, while long chains of time advancing eventives are more common in narratives. We can see from Table 7 that in news data only 30% events are eventive, leaving the rest 70% stative descriptions, while in narrative data over half of the events (51%) are eventive. From Table 8 we can also see that the major temporal relation in news is "overlap" (54%), representing dominative stative statements in reporting discourse mode, while narrative texts are dominated by the "before" relation (53%), with eventive statements advancing the story line.

Timex type	News	Narratives
Absolute Concrete	313 (27%)	16 (14%)
Relative Concrete	598 (51%)	20 (17%)
Vague	256 (22%)	79 (67%)

Table 6: Distribution of time expression types.

Event type	News	Narratives
Event	1457 (30%)	5594 (51%)
State	1802 (37%)	3366 (31%)
Habitual	102 (2%)	459 (4%)
Modalized	321 (7%)	458 (4%)
Completed	1041 (22%)	900 (8%)
Ongoing Event	80 (2%)	175 (2%)
Generic State	1 (0%)	17 (0%)
Generic Habitual	2 (0%)	5 (0%)

Table 7: Distribution of event types.

Edge label	News	Narratives
Includes	1096 (18%)	157 (1%)
Before(After)	507 (8%)	5885 (53%)
Overlap	3246 (54%)	4914 (44%)
Depend-on	1125 (19%)	151 (1%)

Table 8: Distribution of temporal relations.

Another difference is that statives serve different major roles in news and narrative texts. News tend to have deep branches of overlapping statives with a time expression, DCT, or a general present/past/future reference time as their parent (descriptive statements as discussed above). Narrative texts have much less such long stative branches, however, they tend to have numerous short branches of statives

with an eventive event as their parent. These statives serve as the event's accompanying situations. For example, in (15) "was$_{s4}$", "was$_{s5}$", "was$_{s6}$", and "baptised$_{s7}$" are accompanying statives to "came$_{e2}$", describing the baby and the family and the situation they were in at that time. For each type of node, we compiled the distribution of its possible types of parent, shown in Table 3. It's worth noting that more than twice as much statives in news have a stative parent (57%) than the ones having an eventive parent (24%), contributing to deep stative branches, while in narratives a much higher percentage of statives directly depend on an eventive (49%), contributing to a large number of short stative branches.

These different temporal properties of news and narratives further result in shallow dependency structures for news texts with larger number of branches on the root node, yet deep structures for narrative texts with fewer but long branches. These differences are illustrated intuitively on Figure 1 and Figure 2.

5. Conclusion

In this work, we proposed a novel approach to model temporal relations in a document – building a temporal dependency tree structure for the document. We argue that this structure is linguistically intuitive, and is amenable to computational modeling. High and stable inter-annotator agreements in our annotation experiments provide further evidence supporting our structured approach to temporal interpretation. In addition, a significant number of documents covering two genres have been annotated. This corpus is publicly available for research on temporal relation analysis, story timeline construction, as well as numerous other applications.

6. Bibliographical References

Allen, J. F. (1984). Towards a general theory of action and time. *Artificial intelligence*, 23(2):123–154.

Bethard, S., Derczynski, L., Savova, G., Pustejovsky, J., and Verhagen, M. (2015). Semeval-2015 task 6: Clinical tempeval. In *SemEval@ NAACL-HLT*, pages 806–814.

Bethard, S., Savova, G., Chen, W.-T., Derczynski, L., Pustejovsky, J., and Verhagen, M. (2016). Semeval-2016 task 12: Clinical tempeval. *Proceedings of SemEval*, pages 1052–1062.

Bethard, S., Savova, G., Palmer, M., and Pustejovsky, J. (2017). Semeval-2017 task 12: Clinical tempeval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572, Vancouver, Canada, August. Association for Computational Linguistics.

Bohnemeyer, J. (2009). Temporal anaphora in a tenseless language. *The expression of time in language*, pages 83–128.

Chambers, N. and Jurafsky, D. (2008). Jointly combining implicit constraints improves temporal ordering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 698–706. Association for Computational Linguistics.

Do, Q. X., Lu, W., and Roth, D. (2012). Joint inference for event timeline construction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 677–687. Association for Computational Linguistics.

Hinrichs, E. (1986). Temporal anaphora in discourses of english. *Linguistics and philosophy*, 9(1):63–82.

Kolomiyets, O., Bethard, S., and Moens, M.-F. (2012). Extracting narrative timelines as temporal dependency structures. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 88–97. Association for Computational Linguistics.

Mostafazadeh, N., Grealish, A., Chambers, N., Allen, J., and Vanderwende, L. (2016). Caters: Causal and temporal relation scheme for semantic annotation of event structures. In *Proceedings of the The 4th Workshop on EVENTS: Definition, Detection, Coreference, and Representation, San Diego, California, June. Association for Computational Linguistics*.

O'Gorman, T., Wright-Bettner, K., and Palmer, M. (2016). Richer event description: Integrating event coreference with temporal, causal and bridging annotation. *Computing News Storylines*, page 47.

Partee, B. H. (1973). Some structural analogies between tenses and pronouns in english. *The Journal of Philosophy*, 70(18):601–609.

Partes, B. H. (1984). Nominal and temporal anaphora. *Linguistics and philosophy*, 7(3):243–286.

Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003a). Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003b). The timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

Reichenbach, H. (1947). *Elements of Symbolic Logic*. The MacMillan Company, New York.

Smith, C. S. (2003). *Modes of discourse: The local structure of texts*, volume 103. Cambridge University Press.

Styler IV, W. F., Bethard, S., Finan, S., Palmer, M., Pradhan, S., de Groen, P. C., Erickson, B., Miller, T., Lin, C., Savova, G., et al. (2014). Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

UzZaman, N., Llorens, H., Allen, J., Derczynski, L., Verhagen, M., and Pustejovsky, J. (2012). Tempeval-3: Evaluating events, time expressions, and temporal relations. *arXiv preprint arXiv:1206.5333*.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007a). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M.,

Katz, G., and Pustejovsky, J. (2007b). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic, June. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Moszkowicz, J., and Pustejovsky, J. (2009). The TempEval Challeage: Identifying Temporal Relations in Text. *Lang Resources & Evaluation*, 43:161–179.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010a). Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62. Association for Computational Linguistics.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010b). Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, Uppsala, Sweden, July. Association for Computational Linguistics.

Webber, B. L. (1988). Tense as discourse anaphor. *Computational Linguistics*, 14(2):61–73.

Wuyun, S. (2016). The influence of tense interpretation on discourse coherence - a comparison between mandarin narrative and report discourse. *Lingua*, 179:38 – 56.

Xue, N., Xia, F., Chiou, F.-D., and Palmer, M. (2005). The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural language engineering*, 11(2):207–238.

Xue, N., Ng, H. T., Pradhan, S., Rutherford, A., Webber, B., Wang, C., and Wang, H. (2016). Conll 2016 shared task on multilingual shallow discourse parsing. *Proceedings of the CoNLL-16 shared task*, pages 1–19.

Yoshikawa, K., Riedel, S., Asahara, M., and Matsumoto, Y. (2009). Jointly identifying temporal relations with markov logic. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 405–413. Association for Computational Linguistics.

Zhang, Y. and Xue, N. (2014). Automatic inference of the tense of chinese events using implicit linguistic information. In *EMNLP*, pages 1902–1911. Citeseer.

NL2Bash: A Corpus and Semantic Parser for Natural Language Interface to the Linux Operating System

Xi Victoria Lin*, Chenglong Wang, Luke Zettlemoyer, Michael D. Ernst
Salesforce Research, University of Washington, University of Washington, University of Washington
xilin@salesforce.com, {clwang,lsz,mernst}@cs.washington.edu

Abstract

We present new data and semantic parsing methods for the problem of mapping English sentences to Bash commands (NL2Bash). Our long-term goal is to enable any user to perform operations such as file manipulation, search, and application-specific scripting by simply stating their goals in English. We take a first step in this domain, by providing a new dataset of challenging but commonly used Bash commands and expert-written English descriptions, along with baseline methods to establish performance levels on this task.

Keywords: Natural Language Programming, Natural Language Interface, Semantic Parsing

1. Introduction

The dream of using English or any other natural language to program computers has existed for almost as long as the task of programming itself (Sammet, 1966). Although significantly less precise than a formal language (Dijkstra, 1978), natural language as a programming medium would be universally accessible and would support the automation of highly repetitive tasks such as file manipulation, search, and application-specific scripting (Wilensky et al., 1984; Wilensky et al., 1988; Dahl et al., 1994; Quirk et al., 2015; Desai et al., 2016).

This work presents new data and semantic parsing methods on a novel and ambitious domain — natural language control of the operating system. Our long-term goal is to enable any user to perform tasks on their computers by simply stating their goals in natural language (NL). We take a first step in this direction, by providing a large new dataset (NL2Bash) of challenging but commonly used commands and expert-written descriptions, along with baseline methods to establish performance levels on this task.

The NL2Bash problem can be seen as a type of semantic parsing, where the goal is to map sentences to formal representations of their underlying meaning (Mooney, 2014). The dataset we introduce provides a new type of target meaning representations (Bash[1] commands), and is significantly larger (from two to ten times) than most existing semantic parsing benchmarks (Dahl et al., 1994; Popescu et al., 2003). Other recent work in semantic parsing has also focused on programming languages, including regular expressions (Locascio et al., 2016), IFTTT scripts (Quirk et al., 2015), and SQL queries (Kwiatkowski et al., 2013; Iyer et al., 2017; Zhong et al., 2017). However, the shell command data we consider raises unique challenges, due to its irregular syntax (no syntax tree representation for the command options), wide domain coverage ($>$ 100 Bash utilities), and a large percentage of unseen words (e.g. commands can manipulate arbitrary files).

We constructed the NL2Bash corpus with frequently used Bash commands scraped from websites such as question-answering forums, tutorials, tech blogs, and course materials. We gathered a set of high-quality descriptions of the commands from Bash programmers. Table 1 shows several examples. After careful quality control, we were able to gather over 9,000 English-command pairs, covering over 100 unique Bash utilities.

We also present a set of experiments to demonstrate that NL2Bash is a challenging task which is worthy of future study. We build on recent work in neural semantic parsing (Dong and Lapata, 2016; Ling et al., 2016), by evaluating the standard Seq2seq model (Sutskever et al., 2014) and the CopyNet model (Gu et al., 2016). We also include a recently proposed stage-wise neural semantic parsing model, Tellina, which uses manually defined heuristics for better detecting and translating the command arguments (Lin et al., 2017). We found that when applied at the right sequence granularity (sub-tokens), CopyNet significantly outperforms the stage-wise model, with significantly less pre-processing and post-processing. Our best performing system obtains top-1 command structure accuracy of 49%, and top-1 full command accuracy of 36%. These performance levels, although far from perfect, are high enough to be practically useful in a well-designed interface (Lin et al., 2017), and also suggest ample room for future modeling innovations.

2. Domain: Linux Shell Commands

A shell command consists of three basic components, as shown in Table 1: utility (e.g. `find`, `grep`), option flags (e.g. `-name`, `-i`), and arguments (e.g. `"*.java"`, `"TODO"`). A utility can have idiomatic syntax for flags (see the `-exec ... {} \;` option of the `find` command).

There are over 250 Bash utilities, and new ones are regularly added by third party developers. We focus on 135 of the most useful utilities identified by the Linux user group (http://www.oliverelliott.org/article/computing/ref_unix/), that is, our domain of target commands contain only those 135 utilities.[2] We only considered the target commands that can

* Work done at the University of Washington.

[1] The Bourne-again shell (Bash) is the most popular Unix shell and command language: https://www.gnu.org/software/bash/. Our data collection approach and baseline models can be trivially generalized to other command languages.

[2] We were able to gather fewer examples for the less common ones. Providing the descriptions for them also requires a higher level of Bash expertise of the corpus annotators.

Natural Language	Bash Command(s)

find .java files in the current directory tree that contain the pattern 'TODO' and print their names	`grep -l "TODO" *.java`
	`find . -name "*.java" -exec grep -il "TODO" {} \;`
	`find . -name "*.java" | xargs -I {} grep -l "TODO" {}`

display the 5 largest files in the current directory and its sub-directories	`find . -type f | sort -nk 5,5 | tail -5`
	`du -a . | sort -rh | head -n5`
	`find . -type f -printf '%s %p\n' | sort -rn | head -n5`

search for all jpg images on the system and archive them to tar ball "images.tar"	`tar -cvf images.tar $(find / -type f -name *.jpg)`
	`tar -rvf images.tar $(find / -type f -name *.jpg)`
	`find / -type f -name "*.jpg" -exec tar -cvf images.tar {} \;`

Table 1: Example natural language descriptions and the corresponding shell commands from NL2Bash.

In-scope	1. Single command 2. Logical connectives: `&&`, `||`, parentheses `()` 3. Nested commands: - pipeline `|` - command substitution `$()` - process substitution `<()`
Out-of-scope	1. I/O redirection <, << 2. Variable assignment = 3. Compound statements: - `if`, `for`, `while`, `util` statements - functions 4. Non-bash program strings nested with language interpreters such as `awk`, `sed`, `python`, `java`

Table 2: In-scope and out-of scope syntax for the Bash commands in our dataset.

be specified in a single line (one-liners).[3] Among them, we omitted commands that contain syntax structures such as I/O redirection, variable assignment, and compound statements because those commands need to be interpreted in context. Table 2 summarizes the in-scope and out-of-scope syntactic structures of the shell commands we considered.

3. Corpus Construction

The corpus consists of text–command pairs, where each pair consists of a Bash command scraped from the web and an expert-generated natural language description. Our dataset is publicly available for use by other researchers: https://github.com/TellinaTool/nl2bash/tree/master/data.
We collected 12,609 text–command pairs in total (§3.1.). After filtering, 9,305 pairs remained (§3.2.). We split this data into train, development (dev), and test sets, subject to the constraint that neither a natural language description nor a Bash command appears in more than one split (§3.4.).

3.1. Data Collection

We hired 10 Upwork[4] freelancers who are familiar with shell scripting. They collected text–command pairs from web pages such as question-answering forums, tutorials, tech blogs, and course materials. We provided them a web inferface to assist with searching, page browsing, and data entry.
The freelancers copied the Bash command from the webpage, and either copied the text from the webpage or wrote the text based on their background knowledge and the webpage context. We restricted the natural language description to be a single sentence and the Bash command to be a one-liner. We found that oftentimes one sentence is enough to accurately describe the function of the command.[5]
The freelancers provided one natural-language description for each command on a webpage. A freelancer might annotate the same command multiple times in different webpages, and multiple freelancers might annotate the same command (on the same or different webpages). Collecting multiple different descriptions increases language diversity in the dataset. On average, each freelancer collected 50 pairs/hour.

3.2. Data Cleaning

We used an automated process to filter and clean the dataset, as described below. Our released corpus includes the filtered data, the full data, and the cleaning scripts.

Filtering The cleaning scripts removed the following commands.

- Non-grammatical commands that violate the syntax specification in the Linux man pages (https://linux.die.net/man/).
- Commands that contain out-of-scope syntactic structures shown in Table 2.
- Commands that are mostly used in multi-statement shell scripts (e.g. `alias` and `set`).
- Commands that contain non-bash language interpreters (e.g. `python`, `c++`, `brew`, `emacs`). These commands contain strings in other programming languages.

Cleaning We corrected spelling errors in the natural language descriptions using a probabilistic spell checker (http://norvig.com/spell-correct.html). We also manually corrected a subset of the spelling errors that bypassed the spell checker in both the natural language and the shell commands. For Bash commands, we removed `sudo` and the shell input

[3] We decided to investigate this simpler case prior to synthesizing longer shell scripts because one-liner Bash commands are practically useful and have simpler structure. Our baseline results and analysis (§6.) show that even this task is challenging.

[4] http://www.upwork.com/

[5] As discussed in §6.3., in 4 out of 100 examples, a one-sentence description is difficult to interpret. Future work should investigate interactive natural language programming approaches in these scenarios.

# sent.	# word	# words per sent.		# sent. per word	
		avg.	median	avg.	median
8,559	7,790	11.7	11	14.0	1

Table 3: Natural Language Statistics: # unique sentences, # unique words, # words per sentence and # sentences that a word appears in.

# cmd	# temp	# token	# tokens / cmd		# cmds / token	
			avg.	median	avg.	median
7,587	4,602	6,234	7.7	7	11.5	1

# utility	# flag	# reserv. token	# cmds / util.		# cmds / flag	
			avg.	median	avg.	median
102	206	15	155.0	38	101.7	7.5

Table 4: Bash Command Statistics. The top table contains # unique commands, # unique command templates, # unique tokens, # tokens per command and # commands that a token appears in. The bottom table contains # unique utilities, # unique flags, # unique reserved tokens, # commands a utility appears in and # commands a flag appears in.

.

prompt characters such as "$" and "#" from the beginning of each command. We replaced the absolute pathnames for utilities by their base names (e.g., we changed `/bin/find` to `find`).

3.3. Corpus Statistics

After filtering and cleaning, our dataset contains 9,305 pairs. The Bash commands cover 102 unique utilities using 206 flags — a rich functional domain.

Monolingual Statistics Tables 3 and 4 show the statistics of natural language (NL) and Bash commands in our corpus. The average length of the NL sentences and Bash commands are relatively short, being 11.7 words and 7.7 tokens respectively. The median word frequency and command token frequency are both 1, which is caused by the large number of open-vocabulary constants (file names, date/time expressions, etc.) that appeared only once in the corpus.[6]

We define a command template as a command with its arguments replaced by their semantic types. For example, the template of `grep -l "TODO" *.java` is `grep -l [regex] [file]`.

Mapping Statistics Table 5 shows the statistics of natural language to Bash command mappings in our dataset. While most of the NL sentences and Bash commands form one-to-one mappings, the problem is naturally a many-to-many mapping problem — there exist many semantically equivalent commands, and one Bash command may be phrased in different NL descriptions. This many-to-many mapping is common in machine translation datasets (Papineni et al., 2002), but rare for traditional semantic parsing ones (Dahl et al., 1994; Zettlemoyer and Collins, 2005).

# cmd per nl			# nl per cmd		
avg.	median	max	avg.	median	max
1.09	1	9	1.23	1	22

Table 5: Natural Language to Bash Mapping Statistics

As discussed in §4. and §6.2., the many-to-many mapping affects both evaluation and modeling choices.

Utility Distribution Figure 1 shows the top 50 most common Bash utilities in our dataset and their frequencies in log-scale. The distribution is long-tailed: the top most frequent utility `find` appeared 6,268 times and the second most frequent utility `xargs` appeared 1,047 times. The 52 least common bash utilities, in total, appeared only 984 times.[7]

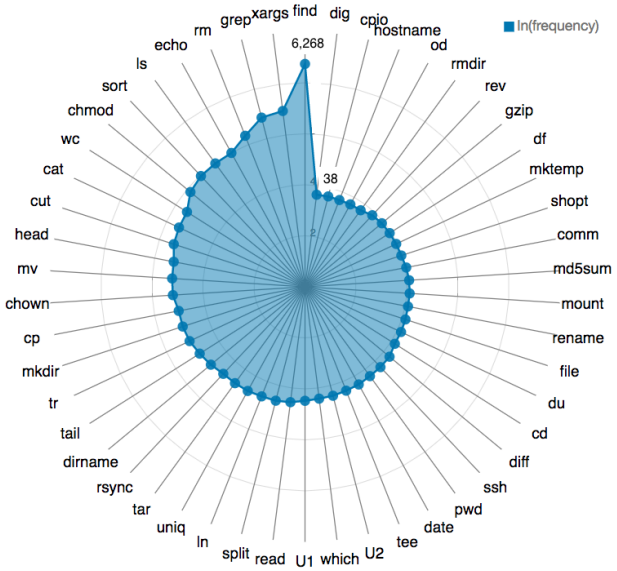


Figure 1: Top 50 most frequent bash utilities in the dataset with their frequencies in log scale. `U1` and `U2` at the bottom of the circle denote the utilities `basename` and `readlink`.

The appendix (§10.) gives more corpus statistics.

3.4. Data Split

We split the filtered data into train, dev, and test sets (Table 6). We first clustered the pairs by NL descriptions — a cluster contains all pairs with the identical normalized NL description. We normalized an NL description by lowercasing, stemming, and stop-word filtering, as described in §6.1.

We randomly split the clusters into train, dev, and test at a ratio of 10:1:1. After splitting, we moved all development and test pairs whose command appeared in the train set into the train set. This prevents a model from obtaining high accuracy by trivially memorizing a natural language

[6] As shown in figure 1, the most frequent bash utilities appeared over 6,000 times in the corpus. Similarly, natural language words such as "files", "in" appeared in 5,871 and 5,430 sentences, respectively. These extremely high frequency tokens are the reason for the significant difference between the averages and medians in Tables 3 and 4.

[7] The utility `find` is disproportionately common in our corpus. This is because we collected the data in two separated stages. As a proof of concept, we initially collected 5,413 commands that contain the utility `find` (and may also contain other utilities). After that, we allow the freelancers to collect all commands that contain any of the 135 utilities described in §2..

description or a command it has seen in the train set, which allows us to evaluate the model's ability to generalize.

	Train	Dev	Test
# pairs	8,090	609	606
# unique nls	7,340	549	547

Table 6: Data Split Statistics

4. Evaluation Methodology

In our dataset, one natural language description may have multiple correct Bash command translations. This presents challenges for evaluation since not all correct commands are present in our dataset.

Manual Evaluation We hired three Upwork freelancers who are familiar with shell scripting. To evaluate a particular system, the freelancers independently evaluated the correctness of its top-3 translations for all test examples. For each command translation, we use the majority vote of the three freelancers as the final evaluation.

We grouped the test pairs that have the same normalized NL descriptions as a single test instance (Table 6). We report two types of accuracy: top-k full command accuracy (Acc_{F}^k) and top-k command template accuracy (Acc_{T}^k). We define Acc_{F}^k to be the percentage of test instances for which a correct full command is ranked k or above in the model output. We define Acc_{T}^k to be the percentage of test instances for which a correct command template is ranked k or above in the model output (i.e., ignoring incorrect arguments).

Table 7 shows the inter-annotator agreement between the three pairs of our freelancers on both the template judgement (α_{T}) and full-command judgement (α_{F}).

Pair 1		Pair 2		Pair 3	
α_{F}	α_{T}	α_{F}	α_{T}	α_{F}	α_{T}
0.89	0.81	0.83	0.82	0.90	0.89

Table 7: Inter-annotator agreement.

Previous approaches Previous NL-to-code translation work also noticed similar problems.

(Kushman and Barzilay, 2013; Locascio et al., 2016) formally verify the equivalence of different regular expressions by converting them to minimal deterministic finite automaton (DFAs).

Others (Kwiatkowski et al., 2013; Long et al., 2016; Guu et al., 2017; Iyer et al., 2017; Zhong et al., 2017) evaluate the generated code through execution. As Bash is a Turing-complete language, verifying the equivalence of two Bash commands is undecidable. Alternatively, one can check command equivalence using test examples: two commands can be executed in a virtual environment and their execution outcome can be compared. We leave this evaluation approach to future work.

Some other works (Oda et al., 2015) have adopted fuzzy evaluation metrics, such as BLEU, which is widely used to measure the translation quality between natural languages (Doddington, 2002). Appendix C shows that the n-gram overlap captured by BLEU is not effective in measuring the semantic similarity for formal languages.

5. System Design Challenges

This section lists challenges for semantic parsing in the Bash domain.

Rich Domain The application domain of Bash ranges from file system management, text processing, network control to advanced operating system functionality such as process management. Semantic parsing in Bash is equivalent to semantic parsing for each of the applications. In comparison, many previous works focus on only one domain (§7.).

Out-of-Vocabulary Constants Bash commands contain many open-vocabulary constants such as file/path names, file properties, time expressions, etc. These form the unseen tokens for the trained model. Nevertheless, a semantic parser on this domain should be able to generate those constants in its output. This problem exists in nearly all NL-to-code translation problems but is particularly severe for Bash (§3.3.). What makes the problem worse is that oftentimes, the constants corresponding to the command arguments need to be properly reformatted following idiomatic syntax rules.

Language Flexibility Many bash commands have a large set of option flags, and multiple commands can be combined to solve more complex tasks. This often results in multiple correct solutions for one task (§3.3.), and poses challenges for both training and evaluation.

Idiomatic Syntax The Bash interpreter uses a shallow syntactic grammar to parse pipelines, code blocks, and other high-level syntax structures. It parses command options using pattern matching and each command can have idiomatic syntax rules (e.g. to specify an `ssh` remote, the format needs to be `[USER@]HOST:SRC`). Syntax-tree-based parsing approaches (Yin and Neubig, 2017; Guu et al., 2017) are hence difficult to apply.

6. Baseline System Performance

To establish performance levels for future work, we evaluated two neural machine translation models that have demonstrated strong performance in both NL-to-NL translation and NL-to-code translation tasks, namely, Seq2Seq (Sutskever et al., 2014; Dong and Lapata, 2016) and CopyNet (Gu et al., 2016). We also evaluated a stage-wise natural language programing model, Tellina (Lin et al., 2017), which includes manually-designed heuristics for argument translation.

Seq2Seq The Seq2Seq (sequence-to-sequence) model defines the conditional probability of an output sequence given the input sequence using an RNN (recurrent neural network) encoder-decoder (Jain and Medsker, 1999; Sutskever et al., 2014). When applied to the NL-to-code translation problem, the input natural language and output commands are treated as sequences of tokens. At test time, the command sequences with the highest conditional probabilities were output as candidate translations.

CopyNet CopyNet (Gu et al., 2016) is an extension of Seq2Seq which is able to select sub-sequences of the input sequence and emit them at proper places while generating the output sequence. The copy action is mixed with the regular token generation of the Seq2Seq decoder and the whole model is still trained end-to-end.

Tellina The stage-wise natural language programing model, Tellina (Lin et al., 2017), first abstracts the constants in an NL to their corresponding semantic types (e.g. `File` and `Size`) and performs template-level NL-to-code translation. It then fills the argument slots in the code template with the extracted constants using a learned alignment model and reformatting heuristics.

6.1. Implementation Details

We used the Seq2Seq formulation as specified in (Sutskever et al., 2014). We used the gated recurrent unit (GRU) (Chung et al., 2014) RNN cells and a bidirectional RNN (Schuster and Paliwal, 1997) encoder. We used the copying mechanism proposed by (Gu et al., 2016). The rest of the model architecture is the same as the Seq2Seq model.
We evaluated both Seq2Seq and CopyNet at three levels of token granularities: token, character and sub-token.

Pre-processing We used a simple regular-expression based natural language tokenizer and the Snowball stemmer to tokenize and stem the natural language. We converted all closed-vocabulary words in the natural language to lowercase and removed words in a stop-word list. We removed all NL tokens that appeared less than four times from the vocabulary for the token- and sub-token-based models. We used a Bash parser augmented from Bashlex (https://github.com/idank/bashlex) to parse and tokenize the bash commands.
To compute the sub-tokens[8], we split every constant in both the natural language and Bash commands into consecutive sequences of alphabetical letters and digits; all other characters are treated as an individual sub-token. (All Bash utilities and flags are treated as atomic tokens as they are not constants.) A sequence of sub-tokens as the result of a token split is padded with the special symbols `SUB_START` and `SUB_END` at the beginning and the end. For example, the file path "/home/dir03/*.txt" is converted to the sub-token sequence: `SUB_START`, "/", "home", "/", "dir", "03", "/", "*", ".", "txt", `SUB_END`.

Hyperparameters The dimension of our decoder RNN is 400. The dimension of the two RNNs in the bi-directional encoder is 200. We optimized the learning objective with mini-batched Adam (Kingma and Ba, 2014), using the default momentum hyperparameters. Our initial learning rate is 0.0001 and the mini-batch size is 128. We used variational RNN dropout (Gal and Ghahramani, 2016) with 0.4 dropout rate. For decoding we set the beam size to 100. The hyperparameters were set based on the model's performance on a development dataset (§3.4.).
Our baseline system implementation is released on Github: https://github.com/TellinaTool/nl2bash.

[8]As discussed in §6.2., the simple sub-token based approach is surprisingly effective for this problem. It avoids modeling very long sequences, as the character-based models do, by preserving trivial compositionality in consecutive alphabetical letters and digits. On the other hand, the separation between letters, digits, and special tokens explicitly represented most of the idiomatic syntax of Bash we observed in the data: the sub-token based models effectively learn basic string manipulations (addition, deletion and replacement of substrings) and the semantics of Bash reserved tokens such as `$`, `"`, `*`, etc.

Model		Acc_F^1	Acc_F^3	Acc_T^1	Acc_T^3
Seq2Seq	Char	0.24	0.27	0.35	0.38
	Token	0.10	0.12	**0.53**	0.59
	Sub-token	0.19	0.27	0.41	0.53
CopyNet	Char	0.25	0.31	0.34	0.41
	Token	0.21	0.34	0.47	**0.61**
	Sub-token	**0.31**	**0.40**	0.44	0.53
Tellina		0.29	0.32	0.51	0.58

Table 8: Translation accuracies of the baseline systems on 100 instances sampled from the dev set.

6.2. Results

Table 8 shows the performance of the baseline systems on 100 examples sampled from our dev set. Since manually evaluating all 7 baselines on the complete dev set is expensive, we report the manual evaluation results on a sampled subset in Table 8 and the automatic evaluation results on the full dev set in Appendix C.
Table 11 shows a few dev set examples and the baseline system translations. We now summarize the comparison between the different systems.

Token Granularity In general, token-level modeling yields higher command structure accuracy compared to using characters and sub-tokens. Modeling at the other two granularities gives higher full command accuracy. This is expected since the character and sub-token models need to learn token-level compositions. They also operate over longer sequences which presents challenges for the neural networks. It is somewhat surprising that Seq2Seq at the character level achieves competitive full command accuracy. However, the structure accuracy of these models is significantly lower than the other two counterparts.[9]

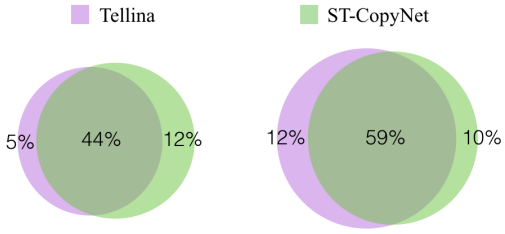
Copying Adding copying slightly improves the character-level models. This is expected as out-of-vocabulary characters are rare. Using token-level copying improves full command accuracy significantly from vanilla Seq2Seq. However, the command template accuracy drops slightly, possibly due to the mismatch between the source constants and the command arguments, as a result of argument reformatting. We observe a similarly significant full command accuracy improvement by adding copying at the sub-token level. The resulting ST-CopyNet model has the highest full command accuracy and competitive command template accuracy.

End-To-End vs. Pipline The Tellina model which does template-level translation and argument filling/reformatting in a stage-wise manner yields the second-best full command accuracy and second-best structure accuracy. Nevertheless, the higher full command accuracy of ST-CopyNet (especially on the Acc_T^3 metrics) shows that learned string-level transformations out-perform manually written heuristics

[9] (Lin et al., 2017) reported that incorrect commands can help human subjects, even when their arguments contain errors. This is because in many cases the human subjects were able to change or replace the wrong arguments based on their prior knowledge. Given this finding, we expect pure character-based models to be less useful in practice compared to the other two groups if we cannot find ways to improve their command structure accuracy.

Model	$\text{Acc}_{\text{F}}^{1}$	$\text{Acc}_{\text{F}}^{3}$	$\text{Acc}_{\text{T}}^{1}$	$\text{Acc}_{\text{T}}^{3}$
ST-CopyNet	**0.36**	**0.45**	0.49	0.61
Tellina	0.27	0.32	0.53	0.62

Table 9: Translation accuracies of ST-CopyNet and Tellina on the full test set.



(a) Template Error (b) Full Command Error

Figure 2: Error overlap of ST-CopyNet and Tellina. The number denotes the percentage out of the 100 dev samples.

when enough data is provided. This shows the promise of applying end-to-end learning on such problems in future work.

Table 9 shows the test set accuracies of the top-two performing approaches, ST-CopyNet and Tellina, evaluated on the entire test set. The accuracies of both models are higher than those on the dev set[10], but the relative performance gap holds: ST-CopyNet performs significantly better than Tellina on the full command accuracy, with only a mild decrease in structure accuracy.

Section 6.3. furthur discusses the comparison between these two systems through error analysis.

6.3. Error Analysis

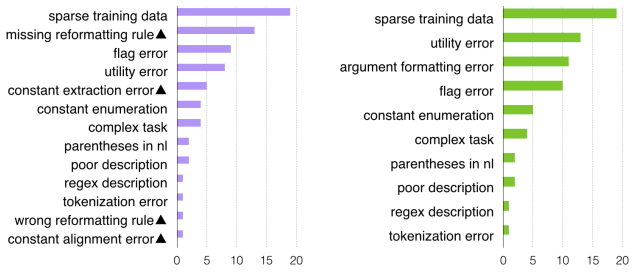
We manually examined the top-1 system outputs of ST-CopyNet and Tellina on the 100 dev set examples and compared their error cases.

Figure 2 shows the error case overlap of the two systems. For a significant proportion of the examples both systems made mistakes in their translation (44% by command structure error and 59% by full command error). This is because the base model of the two systems are similar — they are both RNN based models that perform sequential translation. Many such errors were caused by the NL describing a function that rarely appeared in the train set, or the GRUs failing to capture certain portions of the NL descriptions. For cases where only one of the models makes mistakes, Tellina makes fewer template errors and ST-CopyNet makes fewer full command errors.

We categorized the error causes of each system (Figure 3), and discuss the major error classes below.

Sparsity in Training Data For both models, the top-one error cause is when the NL description maps to utilities or flags that rarely appeared in the train set (Table 10). As mentioned in section 2., the bash domain consists of a large number of utilities and flags and it is expensive to gather enough training data for all of them.

[10]One possible reason is that two different sets of programmers evaluated the results on dev and test.



(a) Tellina Error Classes (b) ST-CopyNet Error Classes

Figure 3: Number of error instances in each error classes of ST-CopyNet and Tellina. The classes marked with ▲ are unique to the pipeline system.

Sparsity in training data
find all the text files in the file system and search only in the disk partition of the root.

Constant enumeration
Answer "n" to any prompts in the interactive recursive removal of "dir1", "dir2", and "dir3".

Complex task
Recursively finds all files in a current folder excluding already compressed files and compresses them with level 9.

Intelligible/Non-grammatical description
Find all regular files in the current directory tree and print a command to move them to the current directory.

Table 10: Samples of natural language descriptions for the major error causes.

Common Errors of RNN Translation Models The second major error class is commonly-known errors for RNN-based translation models (utility error, flag error and argument formatting error in Figure 3). When the RNN misinterprets or overlooks certain chunks of NL descriptions, the decoder can generate a wrong utility/flag or omit a utility/flag from the output sequence. Since the ST-CopyNet model also relies on the RNNs to generate sub-token contents, it suffers more from such problems — the sub-token based models in general have more command structure errors and they frequently generated arguments that are a few edit distance away from the correct ones. Interestingly, we noticed that few command template errors are syntax errors. The output commands often remain executable despite the semantic errors in different Bash components.

Constant Enumeration In some cases, the NL descriptions contain sequences of constant values as an enumeration of system objects or string patterns (Table 10). We observed that both models struggled to extract all the constants correctly from this type of descriptions and usually failed to set the extracted constants into the correct command slots. Moreover, long sequences of OOVs also down-qualify the RNN encodings and both models made more command structure errors in such cases.

Natural Language	Model Output	Model
Delete all empty directories in minimum 2 levels down the root directory	`find root -mindepth 2 -type d -empty -delete`	Human
	`find root -mindepth 2 -type d -empty -delete`	C-Seq2Seq
	`find root -mindepth 2 -type d -empty -delete`	C-CopyNet
	`find root -mindepth 2 -type d -empty -delete`	T-Seq2Seq
	`find root -mindepth 2 -type d -empty -delete`	T-CopyNet
	`find root -mindepth 2 -type d -empty -delete`	ST-Seq2Seq
	`find root -mindepth 2 -type d -empty -delete`	ST-CopyNet
	`find . -mindepth 2 -type d -empty -delete`	Tellina
Find all Subscription.java files/directories under current directory and enter into the parent directory of the first one found	`cd $(find . -name Subscription.java -printf '%h\n')`	Human
	`find . -name Subscription.java ... incomplete`	C-Seq2Seq
	`find . -name Subscription.java | xargs grep Subscription.java`	C-CopyNet
	`cd $(find . -name Subscription.java | xargs -I {} dirname {})`	T-Seq2Seq
	`cd $(find . -name Subscription.java | xargs -I {} dirname {})`	T-CopyNet
	`cd $(find . -name Subscription.java | xargs -I {} dirname {})`	ST-Seq2Seq
	`cd $(find . -name Subscription.java | xargs -I {} dirname {})`	ST-CopyNet
	`cd $(find . -name "Subscription.java" | xargs -I {} dirname {})`	Tellina
Add executable permission to "pretty-print"	`chmod +x pretty-print`	Human
	`chmod +x prety-printf`	C-Seq2Seq
	`chmod pretty-print pretty-print`	C-CopyNet
	`chmod +x UNK`	T-Seq2Seq
	`chmod +x UNK`	T-CopyNet
	`chmod +x UNK-UNK-UNK`	ST-Seq2Seq
	`chmod +x pretty-print`	ST-CopyNet
	`chmod [permission] "pretty-print"`	Tellina
Search for the string 'git' in all the files under current directory tree without traversing into '.git' folder and excluding files that have 'git' in their names	`find . -path ./.git -prune -o -not -name '*git*' -print |xargs grep git`	Human
	`find . -name git -prune -or -print ... incomplete`	C-Seq2Seq
	`find . -name .git -prune -or -name "*.git" -print ... incomplete`	C-CopyNet
	`find . -not -name "*.*" -not -path UNK -not -path UNK ... incomplete`	T-Seq2Seq
	`find . -not \( -path UNK -prune \) -or \( -type f -and ! -name UNK \) -print0 | xargs -0 -I {} grep ... incomplete`	T-CopyNet
	`find . -name '*.git' -not -path '.git' | xargs -I {} grep 'git' {}`	ST-Seq2Seq
	`find . -not -name ".git" -not -path "*.git*" -not -name "*git*" | xargs -I {} grep git {}`	ST-CopyNet
	`find "git" -not -path ".git" -not -name "*" | grep "git"`	Tellina
Rename "file.txt" in directories "v_1", "v_2", and "v_3" each to "v_1.txt", "v_2.txt", and "v_3.txt" respectively and print the conversion	`ls -d v_1,2,3 | xargs -i mv -v {}/file.txt {}/{}.txt`	Human
	`mv file.txt v_1.txt`	C-Seq2Seq
	`mv file.txt v_3.txt`	C-CopyNet
	`mv UNK UNK`	T-Seq2Seq
	`mv UNK UNK`	T-CopyNet
	`diff current_1 {}`	ST-Seq2Seq
	`ssh -i v_1.txt v_3.txt`	ST-CopyNet
	`no output`	Tellina

Table 11: Example predictions of the baseline approaches. The prediction errors are underlined.

Complex Task We found several cases where the NL description specifies a complex task and would be better broken into separate sentences (Table 10). When the task gets complicated, the NL description gets verbose. As noted in previous work (Bahdanau et al., 2014), the performance of RNNs decreases for longer sequences. Giving high-quality NL description for complex tasks are also more difficult for the users in practice — multi-turn interaction is probably necessary for these cases.

Other Classes For the rest of the error cases, we observed that the model failed to translate the specifications in `()`, long descriptions of regular expressions and intelligible/non-grammatical NL descriptions (Table 10). There are also errors propogated from the pre-processing tools such as the NL tokenizer. In addition, the stage-wise system Tellina made a significant number of mistakes specific to its non-end-to-end modeling approach, e.g. the limited coverage of its set of manually defined heuristic rules.

Based on the error analysis, we recommend future work to build shallow command structures in the decoder instead of synthesizing the entire output in sequential manner, e.g. using separate RNNs for template translation and argument filling. The training data sparsity can possibly be alleviated by semi-supervised learning using unlabeled Bash commands or external resources such as the Linux man pages.

7. Comparison to Existing Datasets

This section compares NL2Bash to other commonly-used semantic parsing and NL-to-code datasets.[11] We compare the

[11] We focus on generating utility commands/scripts from natural language and omitted the datasets in the domain of programming challenges (Polosukhin and Skidanov, 2018) and code base model-

Dataset	PL	# pairs	# words	# tokens	Avg. # w. in nl	Avg. # t. in code	NL collection	Code collection	Semantic alignment	Introduced by
IFTTT	DSL	86,960	–	–	7.0	21.8	scraped	scraped	Noisy	(Quirk et al., 2015)
C#2NL*	C#	66,015	24,857	91,156	12	38				(Iyer et al., 2016)
SQL2NL*	SQL	32,337	10,086	1,287	9	46				
RegexLib	Regex	3,619	13,491	179$^{\bullet}$	36.4	58.8$^{\bullet}$			Good$^{\circledast}$	(Zhong et al., 2018)
HeartStone	Python	665	–	–	7	352$^{\bullet}$	game card description	game card source code		(Ling et al., 2016)
MTG	Java	13,297	–	–	21	1,080$^{\bullet}$				
StaQC	Python	147,546	17,635	137,123	9	86	extracted using ML	extracted using ML	Noisy	(Yao et al., 2018)
	SQL	119,519	9,920	21,413	9	60				
NL2RX	Regex	10,000	560	45$^{\bullet\dagger}$	10.6	26$^{\bullet}$	synthesized & paraphrased	synthesized	Very Good	(Locascio et al., 2016)
WikiSQL	SQL	80,654	–	–	–	–				(Zhong et al., 2017)
NLMAPS	DSL	2,380	1,014	–	10.9	16.0	synthesized given code	expert written	Very Good	(Haas and Riezler, 2016)
Jobs640*	DSL	640	391	58†	9.8	22.9	user written	expert written given NL		(Tang and Mooney, 2001)
GEO880	DSL	880	284	60†	7.6	19.1				(Zelle and Mooney, 1996)
Freebase917	DSL	917	–	–	–	–				(Cai and Yates, 2013)
ATIS*	DSL	5,410	936	176†	11.1	28.1				(Dahl et al., 1994)
WebQSP	DSL	4,737	–	–	–	–	search log			(Yih et al., 2016)
NL2RX-KB13	Regex	824	715	85$^{\bullet\dagger}$	7.1	19.0$^{\bullet}$	turker written			(Kushman and Barzilay, 2013)
Django*	Python	18,805	–	–	14.3	–	expert written given code	scraped		(Oda et al., 2015)
NL2Bash	Bash	9,305	7,790	6,234	11.7	7.7				Ours

Table 12: Comparison of datasets for translation of natural language to (short) code snippets. *: Both C#2NL and SQL2NL were originally collected to train systems that explain code in natural language. $^{\bullet}$: The code length is counted by characters instead of by tokens. †: When calculating # tokens for these datasets, the open-vocabulary constants were replaced with positional placeholders. $^{\circledast}$: For these datasets, the NL-code pairs in their original data sources were not compiled for the purpose of semantic parsing. *: Both Jobs640 and ATIS consist of mixed manually-generated and automatically-generated NL-code pairs. * The Django dataset consists of pseudo-code/code pairs.

datasets with respect to: (1) the programming language used, (2) size, (3) shallow quantifiers of difficulty (i.e. # unique NL words, # unique program tokens, average length of text and average length of code) and (4) collection methodology. Table 12 summarizes the comparison. We directly quoted the published dataset statistics we have found, and computed the statistics of other released datasets to our best effort.

Programming Languages Most of the datasets were constructed for domain-specific languages (DSLs). Some of the recently proposed datasets use Java, Python, C#, and Bash, which are Turing-complete programming languages. This shows the beginning of an effort to apply natural language based code synthesis to more general PLs.

Collection Methodology Table 12 sorts the datasets by increasing amount of manual effort spent on the data collection. NL2Bash is by far the largest dataset constructed using practical code snippets and expert-written natural language. In addition, it is significantly more diverse (7,790 unique words and 6,234 unique command tokens) compared to other manually constructed datasets.

The approaches of automatically scraping/extracting parallel natural language and code have been adopted more recently. A major resource of such parallel data are question-answering forums (StackOverflow: https://stackoverflow.com/) and cheatsheet websites (IFTTT: https://ifttt.com/ and RegexLib: http://www.regexlib.com/). Users post code snippets together with natural language questions or descriptions in these venues. The problem with these data is that they are loosely aligned and cannot be directly used for training. Extracting good alignments from them is very challenging (Quirk et al., 2015; Iyer et al., 2016; Yao et al., 2018). That being said, these datasets significantly surpasses the manually gathered ones in terms of size and diversity, hence demonstrating significant potential for future work.

Alternatively, Locascio et al. (2016) and Zhong et al. (2017a) proposed synthesizing parallel natural language and code using a synchronous grammar. They also hired Amazon Mechanical Turkers to paraphrase the synthesized natural language sentences in order to increase their naturalness and diversity. While the synthesized domain may be less diverse compared to naturally existed ones, they served as an excellent resource for data augmentation or zero-shot learning. The downside is that developing synchronous grammars for domains other than simple DSLs is challenging, and other data collection methods are still necessary for them.

The different data collection methods are complimentary and we expect to see more future work mixing different strategies.

ing (Nie et al., 2018).

8. Conclusions

We studied the problem of mapping English sentences to Bash commands (NL2Bash), by introducing a large new dataset and baseline methods. NL2Bash is by far the largest NL-to-code dataset constructed using practical code snippets and expert-written natural language. Experiments demonstrated competitive performance of existing models as well as significant room for future work on this challenging semantic parsing problem.

9. Acknowledgements

The research was supported in part by DARPA under the DEFT program (FA8750-13-2-0019), the ARO (W911NF-16-1-0121), the NSF (IIS1252835, IIS-1562364), gifts from Google and Tencent, and an Allen Distinguished Investigator Award. We thank Zexuan Zhong for providing us the statistics of the RegexLib dataset. We thank Kenton Lee, Luheng He, Omer Levy, and the anonymous reviewers for their constructive feedbacks on the paper draft. We thank the UW NLP/PLSE groups for helpful conversations on the work.

10. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Cai, Q. and Yates, A. (2013). Large-scale semantic parsing via schema matching and lexicon extension. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 423–433. The Association for Computer Linguistics.

Chung, J., Gülçehre, Ç., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling.

Dahl, D. A., Bates, M., Brown, M., Fisher, W., Hunicke-Smith, K., Pallett, D., Pao, C., Rudnicky, A., and Shriberg, E. (1994). Expanding the scope of the atis task: The atis-3 corpus. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, pages 43–48, Stroudsburg, PA, USA. Association for Computational Linguistics.

Desai, A., Gulwani, S., Hingorani, V., Jain, N., Karkare, A., Marron, M., R, S., and Roy, S. (2016). Program synthesis using natural language. In *Proceedings of the 38th International Conference on Software Engineering*, number 1 in ICSE '16, pages 345–356, New York, NY, USA. ACM.

Dijkstra, E. W. (1978). On the foolishness of "natural language programming". In Friedrich L. Bauer et al., editors, *Program Construction, International Summer School, July 26 - August 6, 1978, Marktoberdorf, germany*, volume 69 of *Lecture Notes in Computer Science*, pages 51–53. Springer.

Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, pages 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Dong, L. and Lapata, M. (2016). Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43, Berlin, Germany, August. Association for Computational Linguistics.

Gal, Y. and Ghahramani, Z. (2016). A theoretically grounded application of dropout in recurrent neural networks. In Daniel D. Lee, et al., editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1019–1027.

Gu, J., Lu, Z., Li, H., and Li, V. O. K. (2016). Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Guu, K., Pasupat, P., Liu, E. Z., and Liang, P. (2017). From language to programs: Bridging reinforcement learning and maximum marginal likelihood. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1051–1062.

Haas, C. and Riezler, S. (2016). A corpus and semantic parser for multilingual natural language querying of openstreetmap. In Kevin Knight, et al., editors, *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 740–750. The Association for Computational Linguistics.

Iyer, S., Konstas, I., Cheung, A., and Zettlemoyer, L. (2016). Summarizing source code using a neural attention model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Volume 1: Long Papers*, pages 2073–2083, Berlin, Germany.

Iyer, S., Konstas, I., Cheung, A., Krishnamurthy, J., and Zettlemoyer, L. (2017). Learning a neural semantic parser from user feedback. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 963–973.

Jain, L. C. and Medsker, L. R. (1999). *Recurrent Neural Networks: Design and Applications*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization.

Kushman, N. and Barzilay, R. (2013). Using semantic unification to generate regular expressions from natural language. In Lucy Vanderwende, et al., editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013*, pages 826–836, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA. The Association for Computational Linguistics.

Kwiatkowski, T., Choi, E., Artzi, Y., and Zettlemoyer, L. (2013). Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1556, Seattle, Washington, USA, October. Association for Computational Linguistics.

Lin, X. V., Wang, C., Pang, D., Vu, K., Zettlemoyer, L., and Ernst, M. D. (2017). Program synthesis from natural language using recurrent neural networks. Technical Report UW-CSE-17-03-01, University of Washington Department of Computer Science and Engineering, Seattle, WA, USA, March.

Ling, W., Blunsom, P., Grefenstette, E., Hermann, K. M.,

Kociský, T., Wang, F., and Senior, A. (2016). Latent predictor networks for code generation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Locascio, N., Narasimhan, K., DeLeon, E., Kushman, N., and Barzilay, R. (2016). Neural generation of regular expressions from natural language with minimal domain knowledge. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP, November 1-4, 2016*, pages 1918–1923, Austin, Texas, USA.

Long, R., Pasupat, P., and Liang, P. (2016). Simpler context-dependent logical forms via model projections. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Mooney, R. J. (2014). Semantic parsing: Past, present, and future.

Nie, P., Li, J. J., Khurshid, S., Mooney, R., and Gligoric, M. (2018). Natural language processing and program analysis for supporting todo comments as software evolves. In *Proceedings of the AAAI Workshop of Statistical Modeling of Natural Software Corpora*.

Oda, Y., Fudaba, H., Neubig, G., Hata, H., Sakti, S., Toda, T., and Nakamura, S. (2015). Learning to generate pseudo-code from source code using statistical machine translation (T). In Myra B. Cohen, et al., editors, *30th IEEE/ACM International Conference on Automated Software Engineering, ASE 2015, Lincoln, NE, USA, November 9-13, 2015*, pages 574–584. IEEE Computer Society.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Polosukhin, I. and Skidanov, A. (2018). Neural Program Search: Solving Programming Tasks from Description and Examples. *ArXiv e-prints*, February.

Popescu, A.-M., Etzioni, O., and Kautz, H. (2003). Towards a theory of natural language interfaces to databases. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, pages 149–157, New York, NY, USA. ACM.

Quirk, C., Mooney, R. J., and Galley, M. (2015). Language to code: Learning semantic parsers for if-this-then-that recipes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Volume 1: Long Papers*, pages 878–888, Beijing, China. The Association for Computer Linguistics.

Sammet, J. E. (1966). The use of english as a programming language. *Communications of the ACM*, 9(3):228–230.

Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks. *Trans. Sig. Proc.*, 45(11):2673–2681, November.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Tang, L. R. and Mooney, R. J. (2001). Using multiple clause constructors in inductive logic programming for semantic parsing. In Luc De Raedt et al., editors, *Machine Learning: EMCL 2001, 12th European Conference on Machine Learning, Freiburg, Germany, September 5-7, 2001, Proceedings*, volume 2167 of *Lecture Notes in Computer Science*, pages 466–477. Springer.

Wilensky, R., Arens, Y., and Chin, D. (1984). Talking to unix in english: An overview of uc. *Commun. ACM*, 27(6):574–593, June.

Wilensky, R., Chin, D. N., Luria, M., Martin, J., Mayfield, J., and Wu, D. (1988). The berkeley unix consultant project. *Comput. Linguist.*, 14(4):35–84, December.

Yao, Z., Weld, D., Chen, W.-P., and Sun, H. (2018). Staqc: A systematically mined question-code dataset from stack overflow. In *Proceedings of the 27th International Conference on World Wide Web, WWW 2018, Lyon, France, April 23 - 27, 2018*.

Yih, W., Richardson, M., Meek, C., Chang, M., and Suh, J. (2016). The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*.

Yin, P. and Neubig, G. (2017). A syntactic neural model for general-purpose code generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 440–450.

Zelle, J. M. and Mooney, R. J. (1996). Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, pages 1050–1055. AAAI Press.

Zettlemoyer, L. S. and Collins, M. (2005). Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI'05, pages 658–666, Arlington, Virginia, United States. AUAI Press.

Zhong, V., Xiong, C., and Socher, R. (2017). Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*.

Zhong, Z., Guo, J., Yang, W., Xie, T., Lou, J.-G., Liu, T., and Zhang, D. (2018). Generating regular expressions from natural language specifications: Are we there yet? In *Proceedings of the AAAI Workshop of Statistical Modeling of Natural Software Corpora*.

Appendices

A Additional Data Statistics

A1. Distribution of Less Frequent Utilities

Figure 4 illustrates the frequencies of the 52 least frequent bash utilities in our dataset. Among them, the most frequent utility `dig` appeared only 38 times in the dataset. 7 utilities appeared 5 times or less. We discuss in the next session that many of such low frequent utilities cannot be properly learned at this stage, since the limited number of training examples we have cannot cover all of their usages, or even a reasonably representative subset.

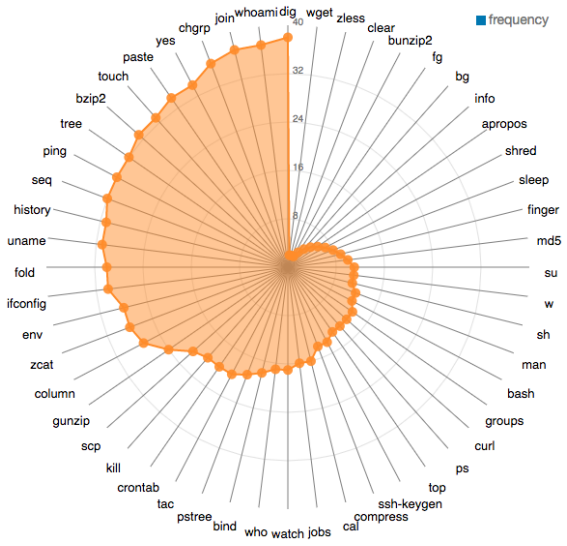


Figure 4: Frequency radar chart of the 52 least frequent bash utilities in the datasets.

A2. Flag Coverage

Table 13 shows the total number of flags (both long and short) a utility has and the number of flags of the utility that appeared in the training set. We show the statistics for the 10 most and least frequent utilities in the corpus. We estimate the total number of flags a utility has by the number of flags we manually extracted from its GNU man page. The estimation is a lower bound as we might miss certain flags due to man page version mismatch and human errors.

Noticed that for most of the utilities, only less than half of their flags appear in the train set. One reason contributed to the small coverage is that most command flags has a full-word replacement for readability (e.g. the readable replacement for `-t` of `cp` is `--target-directory`), yet most Bash commands written in practice uses the short flags. We could solve this type of coverage problem by normalizing the commands to contain only the short flags. (Later we can use deterministic rules to show the readable version to the user.) Nevertheless, for many utilities a subset of their flags are still missing from the corpus. Conducting zero-shot learning for those missing flags is an interesting future work.

Utility	# flags	# flags in train set
find	103	68
xargs	32	15
grep	82	42
rm	17	7
echo	5	2
sort	50	19
chmod	14	4
wc	13	6
cat	19	4
sleep	2	0
shred	17	4
apropos	30	0
info	34	2
bg	0	0
fg	0	0
wget	171	2
zless	0	0
bunzip2	14	0
clear	0	0

Table 13: Training set flag coverage. The upper-half of the table shows the 10 most frequent utilities in the corpus. The lower-half of the table shows the 10 least frequent utilities in the corpus.

B Data Quality

We asked two freelancers to evaluate 100 text-command pairs sampled from our train set. The freelancers did not author the sampled set of pairs themselves. We asked the freelancers to judge the correctness of each pair. We also asked the freelancers to judge if the natural language description is clear enough for them to understand the descriptor's goal. We then manually examined the judgments made by the two freelancers and summarize the findings below.

The freelancers identified errors in 15 of the sampled training pairs, which results in approximately 85% annotation accuracy of the training data. 3 of the errors are caused by the fact that some utilities (e.g. `rm`, `cp`, `gunzip`) handle directories differently from regular files, but the natural language description failed to clearly specify if the target objects include directories or not. 4 cases were typos made by our annotators when copying the constant values in a command to their descriptions. Being able to automatically detect constant mismatch may reduce the number of such errors. (Automatic mismatch detection can be directly added to the annotation interface.) The rest of the 8 cases were caused by the annotators mis-interpreted/omitted the function of certain flags/reserved tokens or failed to spot syntactic errors in the command (listed in Table 14). For many of these cases, the Bash commands are only of medium length — this shows that accurately describing all the information in a Bash command is still an error-prone task for Bash programmers. Moreover, some annotation mistakes are more thought-provoking as the operations in those examples might be difficult/unnatural for the users to describe at test time. In these cases we should solicit the necessary information from the users through alternative ways, e.g. asking multi-choice questions for specific options or asking the user for examples.

Only 1 description was marked as "unclear" by one of the freelancers. The other freelancer still judged it as "clear".

Find all executables under /path directory

```
find /path -perm /ugo+x
```

"Executables generaly means executable files, thus needs `-type f`. Also, `/ugo+x` should be `-ugo+x`. The current command lists all the directories too as directories generally have execute permission at least for the owner (`/ugo+x` allows that, while `-ugo+x` would require execute permission for all)."

*Search the current directory tree for all regular non-hidden files except *.o*

```
find ./ -type f -name "*" -not -name "*.o"
```

"Criteria not met: non-hidden, requires something like -not -name `'.*'`."

Display all the text files from the current folder and skip searching in skipdir1 and skipdir2 folders

```
find .  \( -name skipdir1 -prune , -name skipdir2 -prune -o -name "*.txt" \) -print
```

"Result includes `skipdir2` (this directory name only), the `-o` can be replaced with comma , to solve this."

Find all the files that have been modified in the last 2 days ... missing `-daystart` description

```
find .  -type f -daystart -mtime -2
```

"`daystart` is not specified in description."

Find all the files that have been modified since the last time we checked

```
find /etc -newer /var/log/backup.timestamp -print
```

"'Since the last time we checked', the backup file needs to be updated after the command completes to make this possible."

Search for all the .o files in the current directory which have permisssions 664 and print them.

```
find .  -name *.o -perm 664 -print
```

"Non-syntactical command. Should be `.o` or `"*.o"`."

Search for text files in the directory "/home/user1" and copy them to the directory /home/backup

```
find /home/user1 -name '*.txt' | xargs cp -av --target-directory=/home/backup/ --parents
```

"`--parents` not specified in description, it creates all the parent dirs of the files inside target dir, e.g, a file named `a.txt` would be copied to `/home/backup/home/user1/a.txt`."

Search for the regulars file starting with HSTD ... missing case insensitive description which have been modified yesterday from day start and copy them to /path/tonew/dir

```
find .  -type f -iname 'HSTD*' -daystart -mtime 1 -exec cp {}/path/to new/dir/ \;
```

"Case insensitive not specified but `-iname` used, extra spaces in `/path/to new/dir/`."

Table 14: Training examples whose NL description has errors (underlined). The error explanation is written by the freelancer.

Similar trend were observed during the manual evaluation — the freelancers have little problem understanding each other's descriptions.
It is worth noting that while we found 15 wrong pairs out of 100, for 13 of them the annotator only misinterpreted one of the command tokens. Hence the overall performance of the annotators is high, especially given the large domain size.

C Automatic Evaluation Results

We report two types of fuzzy evaluation metrics automatically computed over full dev set in table 15. We define TM as the maximum percentage of close-vocabulary token (utilities, flags and reserved tokens) overlap between a predicted command and the reference commands. (TM is a command structure accuracy measurement.) TM^k is the maximum TM score achieved by the top-k candidates generated by a system. We use BLEU as an approximate measurement for full command accuracy. $BLEU^k$ is the maximum BLEU score achieved by the top-k candidates generated by a system. We set the BLEU score weights to be (0.25, 0.25, 0.25, 0.25).

Model		$BLEU^1$	$BLEU^3$	TM^1	TM^3
Seq2Seq	Char	49.1	56.7	0.57	0.64
	Token	36.1	43.9	0.65	**0.75**
	Sub-token	46	52	0.65	0.71
CopyNet	Char	49.1	56.8	0.54	0.61
	Token	44.9	54.2	0.65	0.74
	Sub-token	**55.3**	**61.8**	0.64	0.71
Tellina		46	52	0.61	0.70

Table 15: Automatically measured performance of the baseline systems on the full dev set.

First, we observed from table 15 that while the automatic evaluation metrics agrees with the manual ones (Table 8) on the system with the highest full command accuracy and the system with the highest command structure accuracy, they do not agree with the manual evaluation in all cases (e.g. character-based models have the second-best BLEU score). Second, the TM score is not discriminative enough – several systems scored similarly on this metrics.

World Knowledge for Abstract Meaning Representation Parsing

Charles Welch[1], **Jonathan K. Kummerfeld**[1], **Song Feng**[2], **Rada Mihalcea**[1]
University of Michigan[1], IBM Research[2]
{cfwelch, jkummerf, mihalcea}@umich.edu, sfeng@us.ibm.com

Abstract

In this paper we explore the role played by world knowledge in semantic parsing. We look at the types of errors that currently exist in a state-of-the-art Abstract Meaning Representation (AMR) parser, and explore the problem of how to integrate world knowledge to reduce these errors. We look at three types of knowledge from (1) WordNet hypernyms and super senses, (2) Wikipedia entity links, and (3) retraining a named entity recognizer to identify concepts in AMR. The retrained entity recognizer is not perfect and cannot recognize all concepts in AMR and we examine the limitations of the named entity features using a set of oracles. The oracles show how performance increases if it can recognize different subsets of AMR concepts. These results show improvement on multiple fine-grained metrics, including a 6% increase in named entity F-score, and provide insight into the potential of world knowledge for future work in Abstract Meaning Representation parsing.

Keywords: semantic parsing, Abstract Meaning Representation, world knowledge, named entity recognition

1. Introduction

Abstract Meaning Representation (AMR), introduced by Banarescu et al. (2012), aims to capture the semantic meaning of sentences using directed acyclic graphs where nodes are labeled with *concepts* and edges are labeled with *relations*. An example is shown in Figure 1. A number of recent studies use AMR graphs for downstream tasks (Pan et al., 2015; Liu et al., 2015; Sachan and Xing, 2016; Burns et al., 2016) and growing amounts of annotated data enable the development of statistical parsing algorithms and standardized evaluations.

Several parsers have been created that generate an AMR graph given a sentence (Flanigan et al., 2014; Wang et al., 2015b; Damonte et al., 2016), but even the most recent results suggest that there is still significant room to improve the performance for this challenging task.

In this paper, we explore the role of world knowledge for the task of semantic parsing with AMR. The paper makes three main contributions. First, we examine the effect of different types of world knowledge for semantic parsing with AMR for the first time.[1] Second, we examine the upper bound on world knowledge using gold annotations, and provide new insights into the potential of world knowledge in computational approaches to AMR parsing. Finally, we show that we can improve the parsing score over a state-of-the-art parser, with improvement on multiple fine-grained evaluation metrics, including a 6% increase in named entity F-score.

2. Background

There are several semantic parsers built for AMR annotations (Flanigan et al., 2014; Zhou et al., 2016; Wang et al., 2015a; Damonte et al., 2016; Barzdins and Gosko, 2016; Misra and Artzi, 2016) using data released through the LDC (LDC2014T12, LDC2015E86, LDC2016E25, LDC2017T10). The 2014 dataset contains 13,000 sentences, which increased to almost 20,000 in the 2015 set. The 2016 and 2017 datasets contain around 39,000 sentences. The data consist of sentences from English broadcast conversations, weblogs, discussion forums, and newswire data and are annotated with AMR graphs. The datasets build off of each other and include corrections and extensions of the AMR specification (e.g., 2015 introduces wikification and new PropBank frames).

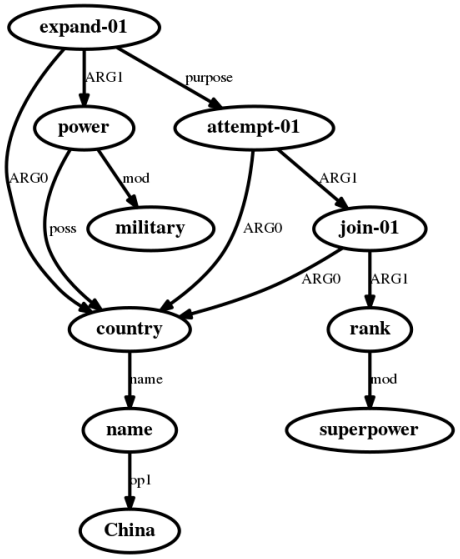


Figure 1: Example AMR graph for the sentence *"China is expanding its military power to attempt to join the ranks of the superpowers"*. Concepts are represented as nodes and relations as edges between those nodes.

The evaluation of these AMR parsers is typically based on the SMATCH F1 tool (Cai and Knight, 2013) which measures the overlap of concept-relation-concept triples in a generated AMR graph as compared to the gold graph. In addition, Damonte et al. (2016) introduced finer-grained evaluations of the subtasks of AMR parsing, which

[1]The code modifications and features are available at `https://github.com/cfwelch/amr_world_knowledge`.

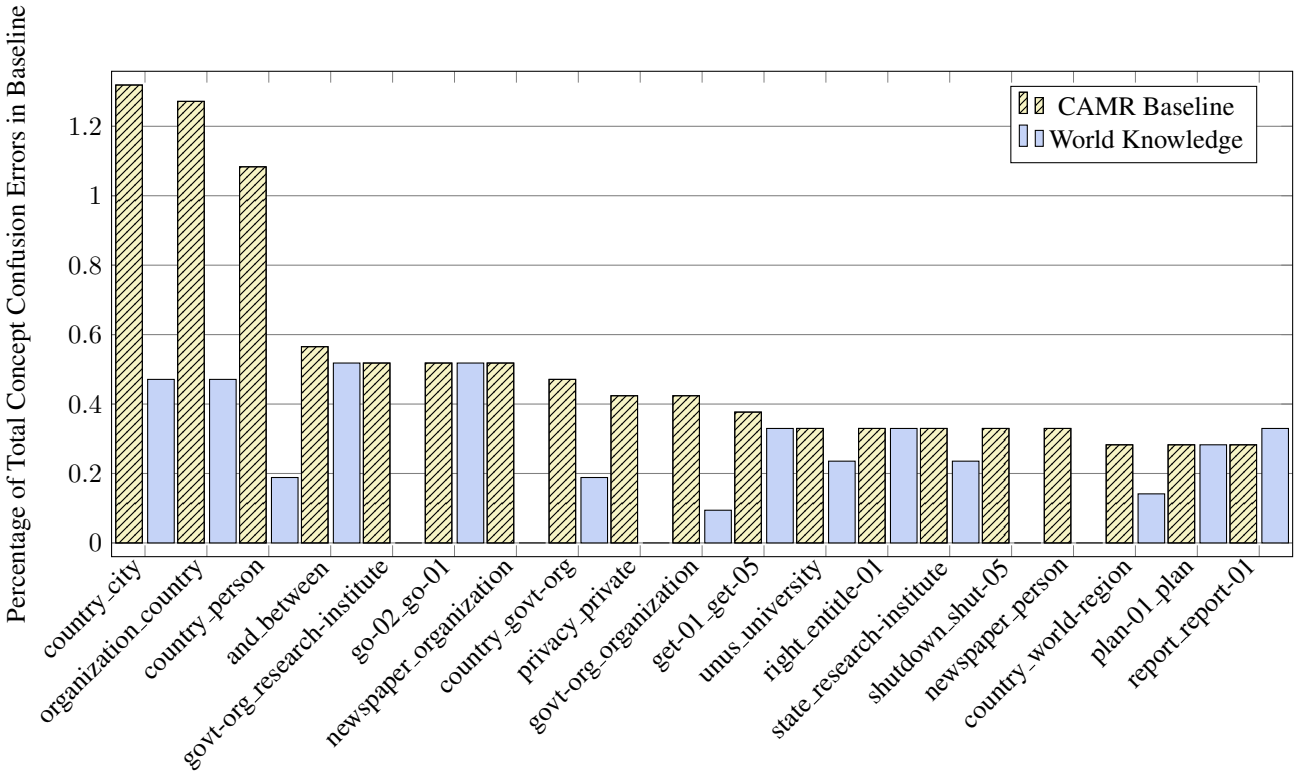


Figure 2: The top 10% of concepts incorrectly identified by the CAMR baseline compared to the percentage of concept confusion errors of each of these types after the addition of world knowledge. The first concept listed is the gold label and the second is the concept it was incorrectly labeled with. We use govt as an abbreviation for 'government'. The numbers after concept names represent their word senses. For instance, get-01 means 'to come into posession', whereas get-05 means 'to get to a state'.

measure parser effectiveness in terms of capturing named entities, concepts, negations, word sense disambiguations and semantic roles.

At the time these experiments were performed many of the high performing parsers were based on JAMR or CAMR (Flanigan et al., 2014; Wang et al., 2015a). We chose to base our parser on CAMR, which was the highest performing entry on SemEval 2016 Task 8 and the parser on which most of the entries for the 2016 shared task were based (May, 2016; Wang et al., 2016). Comparing to previous extensions of CAMR, our work is the first attempt to integrate various forms of world knowledge as features for AMR parsing.

3. AMR Parsing with World Knowledge

We hypothesize that introducing world knowledge could potentially increase the overall performance of AMR parsers. In order to identify useful features and effective approaches to improve an existing AMR parser, we first examine the errors produced by AMR parsers. By looking at errors made by CAMR, we found that a significant number of concepts are either mislabeled or missing. The lighter, striped bars in Figure 2 show the most frequent concept identification errors, representing 10% of the overall concept identification errors that CAMR makes. For instance, as seen in this chart, the concept of *country* is incorrectly labeled as *city*, *organization* is incorrectly labeled as *country*, and *country* is incorrectly labeled as *person* as the top 3 most common errors. These 3 errors make up 3.7% of the total concept identification errors. Based on this error analysis, we chose to integrate world knowledge to reduce concept identification errors.

3.1. World Knowledge for AMR

We examine three types of world knowledge: semantic classes (WordNet classes), named entities, and encyclopedic knowledge (entity links to Wikipedia).

WordNet Classes and Supersenses. WordNet organizes words into semantic hierarchies, and therefore it can be used to abstract words to more general concepts (also referred to as *supersenses* (Miller, 1995)). We use WordNet in two ways.

First, we use a set of 45 WordNet supersenses, 26 of which are for nouns, as assigned by the lexicographers who developed WordNet. Every noun or verb in WordNet is subsumed by one of these supersenses.

Second, we abstract even further by taking advantage of the hypernym hierarchy for nouns. Given a word, we identify its synsets, then for each synset we generate a trace from that node to the root of the hierarchy following hypernym links. Next, we take the labels of nodes in the

traces three steps from the root, which keeps the number of classes small while still having meaningful abstractions. The resulting set of fifteen classes is {`group, thing, measure, change, object, substance, causal agent, relation, matter, horror, communication, psychological feature, set, process, attribute`}.

Named Entities. We use the Stanford named entity tagger (Finkel et al., 2005), and retrain it on the training set of LDC2014T12. To retrain the tagger we need spans in the AMR training sentences and their assigned labels.

To generate annotated data, we use the alignments automatically generated using the JAMR aligner for AMR graphs. The aligner creates token spans corresponding to the generation of concepts in the gold AMR graph. We then use the frequent concept labels as a set of classes for retraining the NE tagger. We consider two methods for choosing the classes for our named entity system. The first method is to list all the concept labels by their frequency and choose the top twenty, removing all the non-noun labels, and removing classes which have low F1 scores when retrained, meaning that the NE model cannot reliably recognize this entity type (we end up with nine classes). The second is to look at the most frequent occurrences of nodes that have name relations, and use this set of types as classes (excluding the *name* type itself).

Entity Linking to Wikipedia. We apply the TAGME entity linker (Ferragina and Scaiella, 2010), which takes a sentence as input and produces an annotated output showing which spans of tokens in the sentence correspond to Wikipedia entries. It also provides a confidence score for the linking. In our implementation, we only consider the entity links that have a confidence over 50%.

3.2. Integration with the CAMR Parser

We integrate world knowledge into the CAMR parser (Wang et al., 2015b), which is one of the top performing AMR parsers. CAMR first generates a dependency parse of a sentence and then iterates over nodes in the dependency tree and decides at each point which of a set of transitions to take. To integrate world knowledge into this parser, we change the feature set generated by each node in the current context window. CAMR scores each possible transition at each step of the parse. The context window can contain the current buffer node, a node representing a potential edge with the current buffer node, and a potential parent node. For each of these nodes, we add features that either have a categorical or boolean value, reflecting the world knowledge available for that node.

4. Experiments

We perform two sets of experiments. In the first set, we automatically infer the values of the three types of world knowledge and use these as features in conjunction with the CAMR parser. In the second set of experiments we use gold standard NER annotations, which we then use to train CAMR to determine an upperbound on the performance of this parser when using world knowledge. We use max-margin learning with AdaGrad instead of the perceptron method in the original code. Previous work has shown this learning method to be effective for a variety of language processing tasks and we observe the same effect (Kummerfeld et al., 2015). In all these experiments, the original CAMR parser naturally constitutes our baseline.

Dataset and Evaluations Metrics. We evaluate our work in two ways: one is overall SMATCH performance (Cai and Knight, 2013), which most of the previous work adopts; the other is the finer-grained evaluation introduced by Damonte et al. (2016), which evaluates the quality of each subtask of AMR parsing.
We focus on a few particularly relevant metrics:

- UNLABELED is the SMATCH score computed on the predicted graphs ignoring all edge labels. It could help tell us if world knowledge helps improve performance on graph structure prediction.
- NO WSD is the score computed by ignoring the word senses after concepts (e.g. 'get-01' and 'get-05' would both become 'get').
- NAMED ENT(ITIES) is the score only checking if the named entity concepts are correct.
- NEGATIONS considers 'polarity' edges in the graph and computes the accuracy of negated concepts.
- CONCEPTS is the score looking only at concept labels and not edge labels.
- REENTRANCY is the score only of reentrant edges in the AMR graph.
- SRL is the score for semantic role labeling which only considers the edge labels.

The dataset we primarily experiment with is LDC2014T12, which was originally used by the CAMR system we extend and does not include Wikification. We experimented with LDC2015E86 and found lower performance comparing to LDC2014T12, which is consistent with recent findings (Zhou et al., 2016; Damonte et al., 2016). Our SMATCH is comparable with Damonte et al. (2016) and outperforms Wang et al. (2015b), however is 3% lower than the graph-based approach used by Flanigan et al. (2016).[2]

4.1. Automatic World Knowledge Augmentations

We augment the CAMR features with the three types of world knowledge described in Section 3. As seen in the left side of Table 1, the addition of world knowledge features leads to an increase in the overall SMATCH F-score as well as several of the finer-grained evaluations. The largest improvement is observed in the named entity subtask, by an absolute 6%. Interestingly, using all of our features in

[2]Flanigan et al. (2016) does not use the types of world knowledge integrated in this paper. We leave integration of world knowledge into the JAMR parser to future work.

		Retrained NER					Oracle NER			
METRIC	CAMR	EL	NE	WN	NE+WN	ALL	9-CLASS	NE	NON-NE	ALL
SMATCH	64.7	**66**	**66**	**66**	**66**	65	67	65	68	**69**
UNLABELED	70.8	**72**	**72**	**72**	**72**	71	73	71	**74**	**74**
NO WSD	65.7	**67**	**67**	**67**	**67**	66	68	66	69	**70**
NAMED ENT.	74.6	77	77	77	**81**	77	**79**	77	78	**79**
NEGATIONS	14.1	**16**	**16**	13	15	14	15	15	**16**	15
CONCEPTS	80.2	80	80	80	**81**	80	81	80	**85**	**85**
REENTRANCY	36.2	38	36	**40**	37	36	**37**	36	**37**	**37**
SRL	**59.1**	59	59	59	58	58	60	59	**61**	**61**

Table 1: Comparisons of CAMR modifications and improvement over the default CAMR model on LDC2014T12 are shown on the left side of the table as SMATCH F1 scores. We examine feature subsets for entity links (EL), named entities (NE), and WordNet features (WN). The right side shows experiments using gold NER concept labels. The highest numbers for each row of each side are in bold.

combination did not perform as well as the combination of named entity and WordNet features. The noise associated with the entity link feature may be the reason why this feature set does not contribute to the overall best classifier. Additionally, we saw 6% reduced false-positives (concepts identified in the parse that should not exist) when using world knowledge than in the CAMR baseline.

Overall, our approach confirms the hypothesis that world knowledge can help an AMR parser. Specifically focusing on the types of errors identified in Section 3, the darker shaded bars in Figure 2 show the errors obtained in the presence of world knowledge. We observe a clear decrease in error, e.g., the percentage of mistakes for "government organization and research institution" drops to zero.

4.2. Gold Standard World Knowledge

To gain additional insight into the role played by NER in the AMR parser, we examine what would happen if we had an NER model that was perfectly accurate for a given set of concept labels. We obtain these NER gold standard labels from the annotations available in the dataset that we use, after aligning the annotation graphs with the raw text. We look at four different scenarios: (1) one scenario where we have labels for the nine classes we used in Section 3.; (2) a second one where we use all named entity concept labels as features; (3) a third scenario using all non-named-entity concept labels as features; and (4) a fourth one where all concept labels of aligned tokens are used as features.

The first column in the right side of Table 1 shows the SMATCH score achieved by using the gold labels for our first method of generating NE tags, which limited our tagger to nine classes. In the second and third experiments we partition the set of labels into NEs and non-NEs. The NEs are the set of concept labels that have 'name' edges in the AMR training data in LDC2014T12. The gold IOB labels for these 252 classes are used to train the parser in experiment 2, and in experiment 3 we use all concepts that are not included in this set which includes about 10k types of labels. The NE score is lower than the 9-class score because NE does not include the 'name' label itself, which is a separate node in the AMR parse and gets aligned to a large number of tokens.

In the last experiment we assume that an NER system can be trained on all concept label types simultaneously and we train and test the parser using these labels as features. As expected, the performance increases significantly in this case. Interestingly, as the performance for most evaluation types increases, the named entity performance is highest when using the real output of the NER system. The gold NER outputs have no effect on the negation or reentrancy scores. Having correct labels intuitively has less to do with these two aspects of AMR graphs.

5. Discussion and Conclusions

Our experiments confirmed the hypothesis that some forms of world knowledge can improve existing AMR parsers. In particular, we found that the combination of named entities and WordNet features outperforms other methods on almost all metrics, except for slightly lower SRL and negation scores. The learning method itself, when properly tuned, gave us an improvement over the CAMR baseline. When looking at the concept confusions in Figure 2 we found that world knowledge helped reduce these errors. We also found false-positives reduced by 6% over the CAMR baseline. The entity link feature did not provide much improvement and it did not help the parser when combined with our other features.

Our analyses of gold standard NER features also revealed some of the limitations of using world knowledge with existing AMR parsers. The upperbound that we identified for this type of knowledge, while clearly above the performance of previous parsers, is still far below the expected performance when gold annotations are being used.

We also attempted to include other forms of world knowledge, encoded in the form of word embeddings (Mikolov et al., 2013) or node embeddings (Grover and Leskovec, 2016), which did not work as expected, and did not lead to improvements. This suggests that future research avenues in AMR parsing should instead focus on improvements in parsing algorithms or training data.

Acknowledgments

This material is based in part upon work supported by IBM under contract 4915012629. Any opinions, findings, conclusions or recommendations expressed above are those of the authors and do not necessarily reflect the views of IBM.

6. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2012). Abstract meaning representation (amr) 1.0 specification. In *https://www.isi.edu/ ulf/amr/help/amr-guidelines.pdf*, pages 1533–1544.

Barzdins, G. and Gosko, D. (2016). Riga at semeval-2016 task 8: Impact of smatch extensions and character-level neural translation on AMR parsing accuracy. In *Proceedings of SemEval*, pages 1143–1147.

Burns, G. A., Hermjakob, U., and Ambite, J. L. (2016). Abstract meaning representations as linked data. In *International Semantic Web Conference*, pages 12–20. Springer.

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *Proceedings of The 52nd Annual Meeting of the Association for Computational Linguistics*, pages 748–752.

Damonte, M., Cohen, S. B., and Satta, G. (2016). An incremental parser for abstract meaning representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 536–546.

Ferragina, P. and Scaiella, U. (2010). Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1625–1628. ACM.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370.

Flanigan, J., Thomson, S., Carbonell, J., Dyer, C., and Smith, N. A. (2014). A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1426–1436.

Flanigan, J., Dyer, C., Smith, N. A., and Carbonell, J. (2016). CMU at semeval-2016 task 8: Graph-based AMR parsing with infinite ramp loss. In *Proceedings of SemEval*, pages 1202–1206.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM.

Kummerfeld, J. K., Berg-Kirkpatrick, T., and Klein, D. (2015). An empirical analysis of optimization for max-margin nlp. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 273–279.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 1077–1086.

May, J. (2016). Semeval-2016 task 8: Meaning representation parsing. In *Proceedings of SemEval*, pages 1063–1073.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Misra, D. K. and Artzi, Y. (2016). Neural shift-reduce CCG semantic parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1775–1786.

Pan, X., Cassidy, T., Hermjakob, U., Ji, H., and Knight, K. (2015). Unsupervised entity linking with abstract meaning representation. In *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 1130–1139.

Sachan, M. and Xing, E. P. (2016). Machine comprehension using rich semantic representations. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 486–492.

Wang, C., Xue, N., and Pradhan, S. (2015a). Boosting transition-based AMR parsing with refined actions and auxiliary analyzers. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics*, pages 857–862.

Wang, C., Xue, N., and Pradhan, S. (2015b). A transition-based algorithm for AMR parsing. In *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 366–375.

Wang, C., Pradhan, S., Xue, N., Pan, X., and Ji, H. (2016). CAMR at semeval-2016 task 8: An extended transition-based AMR parser. In *Proceedings of SemEval*, pages 1173–1178.

Zhou, J., Xu, F., Uszkoreit, H., Qu, W., Li, R., and Gu, Y. (2016). AMR parsing with an incremental joint model. In *Proceedings of SemEval*, pages 680–689.

Improved Transcription and Indexing of Oral History Interviews for Digital Humanities Research

Michael Gref[1], **Joachim Köhler**[1], **Almut Leh**[2]
[1]Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, 53757 Sankt Augustin, Germany
[2]Institute for History and Biography, University of Hagen, 58097 Hagen, Germany
{michael.gref, joachim.koehler}@iais.fraunhofer.de, almut.leh@fernuni-hagen.de

Abstract

This paper describes different approaches to improve the transcription and indexing quality of the Fraunhofer IAIS Audio Mining system on Oral History interviews for the Digital Humanities Research. As an essential component of the Audio Mining system, automatic speech recognition faces a lot of difficult challenges when processing Oral History interviews. We aim to overcome these challenges using state-of-the-art automatic speech recognition technology. Different acoustic modeling techniques, like multi-condition training and sophisticated neural networks, are applied to train robust acoustic models. To evaluate the performance of these models on Oral History interviews a German Oral History test-set is presented. This test-set represents the large audio-visual archives "Deutsches Gedächtnis" of the Institute for History and Biography. The combination of the different applied techniques results in a word error rate reduced by 28.3% relative on this test-set compared to the current baseline system while only one eighth of the previous amount of training data is used. In context of these experiments new opportunities are set out for Oral History research offered by Audio Mining. Also the workflow is described used by Audio Mining to process long audio-files to automatically create time-aligned transcriptions.

Keywords: acoustic modeling, robust speech recognition, multi-condition training, speech retrieval, oral history

1. Introduction

The use of automatic speech recognition technology (ASR) to transcribe and index Oral History interviews has started with the MALACH project (Psutka et al., 2002) where the interviews of the *Survivors of the Shoah Visual History Foundation* (VHF) were processed with a state-of-the-art speech recognition in 2002. The main challenge of this activity was the variety and quality of the recordings and the variety of the languages. (Oard, 2012) investigated how speech recognition technology can be used for Oral History research.

In comparison to other speech collections Oral History recordings have a lot challenges. Often the audio recording quality is low. Further the interviewed persons are elderly persons speaking very spontaneous. This leads to poor recognition performance when applying off-the-shelf speech recognition technology.

On the other hand we observe huge progress in speech recognition using different neural network learning frameworks. Recently many researchers, e.g. from Microsoft and IBM, have reported excellent results on conversational recognition tasks, like switchboard. Also with the open-source Kaldi ASR toolkit huge advances are reported using different sophisticated neural network architecture and training methods.

The Fraunhofer IAIS Audio Mining system is designed to automatically create segmented and time-aligned transcriptions from very long, unstructured audiovisual media files. In this work we want to present the opportunities and advantages offered by the Audio Mining system for the Digital Humanities Research by automatically creating transcriptions of Oral History interviews. Moreover we present the challenges for automatic speech recognition systems - such as used in the Audio Mining system - posed by Oral History interviews and approaches to address these challenges.

In this work we apply the Kaldi ASR toolkit to train sophisticated acoustic models for German ASR systems to improve the speech recognition performance on Oral History data. For evaluation we present and apply a novel ASR test-set representing the challenging Oral History data collection of the Institute for History and Biography of the University of Hagen.

The paper is organized as follows. In section 2. the Fraunhofer IAIS Audio Mining system is described. Also the workflow of the audio analysis is presented that automatically segments and transcribes audiovisual media data. The research on Oral History at the Institute for History and Biography is described in section 3. where the main focus is put on the Oral History database. Furthermore we describe the advances and new opportunities offered by the Audio Mining system for Digital Humanities Research using Oral History. In section 4. we describe the challenges that have to be met in order to achieve reasonable results for the automatic transcription of Oral History interviews. We also present the approaches taken to face these challenges. Training and evaluation of sophisticated acoustic models using these approaches for robust speech recognition are presented in section 5.

2. The Fraunhofer IAIS Audio Mining System

2.1. Overview

The Fraunhofer IAIS Audio Mining system is designed to analyze the audio-signal of audiovisual media files automatically. The aim is to create a time-aligned transcription of the spoken words, as it is normally made by professional human transcribers. To achieve an optimal result this does not only include automatic speech recognition but an entire workflow including segmentation of the audio signal, speech detection, speaker analysis and keyword extrac-



Figure 1: Graphical Web User Interface of the Fraunhofer IAIS Audio Mining system

tion using several state-of-the-art pattern recognition algorithms.

Currently, the Fraunhofer IAIS Audio Mining system enables journalists, archivists and hosts of audiovisual broadcast data to face the challenges caused by the continuously increasing amounts of large audiovisual data. This is achieved by making the files both text-searchable and structured. Thus the amount of time a user needs to work with AV-data is noticeably reduced.

For example, the system enables end-users to quickly navigate within interviews using a Graphical User Interface (GUI) that exploits the analysis-results provided by the Audio Mining system. One example for such a GUI using the Fraunhofer IAIS Audio Mining system is shown in figure 1. An embedded media-player allows users to directly play segments of specific speakers. Different speakers are represented by a unique color for each speaker in the time-bar below the video. Non-speech is represented by grey. Furthermore, a search-engine enables the user to find all media files in which a keyword or phrase was spoken by searching the transcripts of spoken words provided by the automatic speech recognition. The GUI highlights all occurrences of the searched words in the time-bar of the currently played media-file.

2.2. Workflow of the Audio Analysis Subsystem

In the following we describe the aforementioned workflow of the *Audio Analysis Subsystem* as component of the Audio Mining system in more detail. This description is based on recent work at the Fraunhofer IAIS Institute (Schmidt et al., 2016). The schematic structure of the audio analysis workflow for one media-file is illustrated in figure 2.

2.2.1. Audio Segmentation

The audio signal is first cut into segments by an audio segmentation algorithm. For this the *Bayesian Information Criterion* (BIC) is applied on full covariance Gaussian models of the mel-frequency cepstral coefficents (MFCC) (Tritschler and Gopinath, 1999).

2.2.2. Concept Detection

After segmentation, each segment is classified using a speech/non-speech detection. Segments that are assumed to contain speech are additionally classified using a gender detection and then passed to the following processing steps. The two detection algorithms are *Gaussian Mixture Model-Universal Background Model* (GMM-UBM) approaches trained for the respective classification task.

2.2.3. Speaker Clustering

The aim of speaker clustering is to classify all speech-segments in which the same speaker is talking. Recently we adapted iVectors (Dehak et al., 2011) for this task and achieved increased performance compared to our previous BIC-based algorithm.

2.2.4. Automatic Speech Recognition

The automatic speech recognition used within the current version of Audio Mining system is trained using the widely adopted Kaldi ASR toolkit (Povey et al., 2011). The currently used acoustic model is a hybrid *Hidden Markov Model-Deep Neural Network* (HMM-DNN) approach with a fully-connected DNN trained on the 1005h large-scale German broadcast corpus *GerTV1000h* (Stadtschnitzer et al., 2014) of the Fraunhofer IAIS Institute.
The language model is trained on a text-corpus of German broadcast data. For the experiments described in this work a



Figure 2: Audio analysis workflow

lexicon with about $500,000$ words is used. However, we recently trained language models for specific tasks with more than one million words in the lexicon.

The ASR system was evaluated on the DiSCo corpus (Baum et al., 2010). DiSCo is a German evaluation corpus for challenging problems in the broadcast domain and is split in four evaluation sets: planned and spontaneous speech each in clean and mixed noise conditions. The system achieves 15.3% word error rate (WER) on clean planned speech and 19.4% WER on clean spontaneous speech. However, as the following sections show, Oral History interviews are far more challenging for the ASR-system. To achieve satisfactory results advanced modeling approaches have to be applied.

2.2.5. Keyword Extraction

In the last step of analysis keywords are extracted from the ASR-generated transcript using a *tf-idf* (term frequency-inverse document frequency) approach. This allows users to search and filter media files that contain a specific topic.

3. Description of Hagen Database

3.1. Oral History Interviews as Sources for the Humanities

Research based on interviews with witnesses to historical events and the interest in biographical processes and subjective personal information have a long tradition in the social sciences and humanities. Since the early 1980s biographical research emerged in almost all areas of the humanities: sociology and pedagogy, ethnography and ethnology, historical and literary studies, as well as in psychoanalysis and psychology. In the historical sciences, research conducting and analyzing interviews with contemporary witnesses has become known as *Oral History*. Particularly in Germany this research was focused above all on the period of National Socialism and the Second World War. But in the meantime, it has also come to include many other topics and historical periods. The past forty years have seen a multitude of witnesses to a wide range of historical events interviewed by researchers. Today it is hard to imagine the presentation of historical information in exhibitions, documentations and films without using witness accounts to the relevant events.

This method in which most of the interviews in question were conducted is characterized by the fact that rather than structuring the interview around questions, the interviewer encourages the interviewee to freely narrate his or her life story. In terms of biographical research, the outcome is qualified as a narrative life-story interview lasting very often at least three or four hours.

Such an interview is representing a highly individual testimony in which the interviewee has presented large parts of his life story and his world view in a way that is often unguarded and sometimes contradictory. Due to the open character of the narration and the life-story dimension such an interview is worth for more than one interpretation and a valuable source for later re-use the more as many witnesses died meanwhile leaving only their recorded account. For the same reason analyzing as well as archiving Oral History interviews is useful and challenging.

Today archives, museums, historical sites and documentation centers preserve and provide Oral History interviews for historical research, social sciences and other humanities.

3.2. The Oral History Archives "Deutsches Gedächtnis"

The archives "Deutsches Gedächtnis" ("German Memory") provide about 2,500 Oral History interviews conducted from 1975 to this day in more than 100 projects using various recording technologies and interview settings. The average length of the interviews is 3.5 hours. The interview is not structured by questions but is open for the course of memories coming into the interviewee's mind when telling his entire life story from birth and childhood into the present. The interviewees were born between 1895 and 1980.

The original analog recordings of the 2,000 audio and 500 video-interviews are digitized. Although retrieval as well as analysis is based on the transcription to date only half of the interviews are transcribed and saved as text files. Only ten percent of the transcripts are time aligned so that the transcript works as subtitle to the audio or video recording. All interviews are equipped with archival, technical and biographical meta-data.

3.3. The ASR-Test-Set

The ASR-test-set is a subset of the "Deutsches Gedächtnis" archives representing the wide range of interviews with respect to recording technology, interview methodology, dialects and pronunciation. The recording quality and the pronunciation had to be understandable for humans as a precondition to be used as data for the test-set. The selection should include early interviews as well as recent conducted ones and represent the interview method of various academic disciplines. With respect to gender and age the selection should display the entire collection. Within these criteria the test-set was randomly selected. The test-set contains 102 audio files from 35 different speakers with an overall length of about 3.5 hours, $27,053$ spoken words and $4,592$ unique words. The recordings used for the test-set took place between 1980 and 2012.

3.4. Advantages Offered by Audio Mining for Interview-Based Research

Audio Mining offers advantages for archiving and retrieval as well as for analysis and interpretation of Oral History in-

terviews. For both, archiving and analyzing, time-aligned transcription and indexing with keywords is essential. Currently transcribing, labeling and annotating speech recordings is performed completely manually. Due to the huge effort in terms of time and human resources required to do this the efficiency to exploit Oral History interviews for digital humanities research is severely limited.

By transcribing audio-visual recordings semi-automatically and providing additional speech related analysis features - indexing and structuring the content - the Audio Mining tools allows to process huge amounts of Oral History data and enhance retrieval and research based on these interviews.

Regarding archiving and retrieval the Audio Mining tools allow full-text search with direct access to the audio/video recording so that search results can be checked immediately.

With respect to the research process new approaches in quantity and quality are made possible. Covering more data comparative studies and quantitative analysis become reasonable. Furthermore the speech technologies allow to analyze verbal and non-verbal aspects of communication more deeply thus opening new dimensions for qualitative research.

From the early days of Oral History to this day oral historians insist on the oral nature of their sources. Following their demand the audio tape or file is the primary source and should be the main subject of research. In consequence the transcript is only a necessary additive for the analysis. In fact and in practice, the transcript is very often the only source for interpretation and analysis, replacing the audio completely. The subtitled audio/video recording overcomes this discrepancy thus fulfilling the essential demand to analyze not only the transcript but the complete oral source. Thus Audio Mining allows to take full advantage of the untapped potential of Oral History leading back to its original roots.

The special challenge of Oral History interviews is determined by spontaneous colloquial speech and the poor technical quality of the audio material recorded decades ago with insufficient equipment.

4. Challenges and Advanced Modeling Approaches

4.1. General State-of-the-Art Acoustic Modeling Approaches

At the Fraunhofer IAIS we started applying deep-learning for acoustic modeling with the well-known (classical) hybrid HMM-DNN approach some years ago using a fully-connected deep neural network. One of these models is still used in the current production system of Audio Mining. In one of our recent previous works (Schmidt et al., 2016) we trained *recurrent neural networks* (RNNs) with *connectionist temporal classification* (CTC) (Graves et al., 2006) as the objective function and the *Eesen-ASR-Toolkit* (Miao et al., 2015) for acoustic modeling achieving increased speech recognition performance. A comparison of word error rates achieved by hybrid HMM-DNN-Models and CTC-RNN-Models we trained in previous works are given in table 1.

However, many recent works (e.g. (Saon et al., 2017) as one of many) show a significant increase in performance on acoustic modeling using the (classical) hybrid HMM-DNN approach with sophisticated neural network architectures. Thus we decided to intensify our research on acoustic modeling using hybrid HMM-DNN models for the German language. For this we train acoustic models with different state-of-the-art network architectures using the Kaldi ASR toolkit. We aim to increase the general performance of the automatic speech recognition system as well on broadcast data as on Oral History interviews and future applications.

4.2. Audio Signal Quality

The audio signal quality is one of the main challenges we are facing when adapting the ASR system to Oral History interviews. The broadcast audio signals of the GerTV1000h corpus used for training the acoustic model were recorded and post-processed using highly professional equipment. The recordings usually have no background noise, barely perceptible reverberation and the levels are well adjusted. However, Oral History interviews are usually recorded in the living rooms of the contemporary witnesses using commonly available recording equipment. This equipment changed during the years of recordings and result in a wide range of different sound qualities: from nearly clean recordings to recordings with noises, reverberation and even clipping.

There are two main approaches how to face these challenges: *Speech Enhancement* and *Multi-Condition Training*. Speech enhancement aims to modify the signal itself and increase the audio quality by reducing noises and compensating corruptions. *Multi-Condition Training* on the other hand aims to make the acoustic model robust against corruptions by showing the model a wide range of features of corrupted audio signals during training. This approach aims to let the model generalize to unseen noise-types and distortions by relying only on robust acoustic features. In this work we utilize the latter approach and try to train such a robust acoustic model.

Therefore we virtual corrupt the GerTV1000h corpus with *data augmentation* as described in subsection 5.4. Defining discrete-time-signals as sequences of sample values the augmentation can be described as

$$(x_n)_{n\in\mathbb{N}} := (s_n)_{n\in\mathbb{N}} * (h_n)_{n\in\mathbb{N}} \tag{1}$$

if only reverberation and no background noise affects the speech signal or as

$$(\tilde{x}_n)_{n\in\mathbb{N}} := (s_n)_{n\in\mathbb{N}} * (h_n)_{n\in\mathbb{N}} + (w_n)_{n\in\mathbb{N}} * (\tilde{h}_n)_{n\in\mathbb{N}} \tag{2}$$

if also noise inside the room affects the speech signal.

Here $*$ is the convolution operation for sequences, $(s_n)_{n\in\mathbb{N}}$ the sequence of the clean speech signal, $(h_n)_{n\in\mathbb{N}}$, $(\tilde{h}_n)_{n\in\mathbb{N}}$ are room-impulse-responses modeling the reverberation of one room at different positions and $(w_n)_{n\in\mathbb{N}}$ the sequence of the noise-signal.

Such approaches are quite common but still state-of-the-art methods to create multi-condition training data when only clean data is available. For example Google recently used a similar approach to train Google Home on far-field, multichannel speech recognition (Li et al., 2017).

4.3. Language

Other big challenges posed by Oral History interviews are colloquial language used in spontaneous speech, hesitations, age- and health-related changes in the way of speaking and domain specific words used in the interviews that usually do not occur in everyday speech (e.g. *Kriegerwitwensöhne*, German for *sons of a war widow*). These challenges must be addressed by adapting the language model and will be part of our future work. In this work we focus on acoustic modeling only and use our default language model described in subsection 2.2.4..

5. Experiments and Evaluation

5.1. Experimental Setup

In the following we use the Oral-History ASR-Test-Set described in subsection 3.3. along with the aforementioned DiSCo test-sets to measure the gain proposed by the approaches described in the previous section.
In the first set of experiments we train hybrid HMM-DNN acoustic models using different types of sophisticated neural networks. These models are build and trained using the *nnet3*-implementations provided within the Kaldi-framework. We run experiments on the following architectures of neural networks:

- **LSTM** (*Long short-term memory* neural networks):
  Long short-term memory is an architecture for recurrent neural networks proposed by (Hochreiter and Schmidhuber, 1997). The Kaldi-nnet3-implementation uses LSTM-Layers with forget-gates (Gers et al., 2000), peephole connections (Gers and Schmidhuber, 2000) and projection layers (Sak et al., 2014). The LSTM-Network configuration we used consists of three LSTM-Layers.

- **BLSTM** (Bidirectional LSTMs):
  Bidirectional LSTMs were first proposed by (Graves and Schmidhuber, 2005). However, the underlying concept of bidirectional recurrent neural networks was proposed by (Schuster and Paliwal, 1997). The neural networks has three BLSTMs-Layers in the configuration we used.

- Chain-Models (Povey et al., 2016):
  The Chain-Models we trained used both LSTM and TDNN-Layers (Waibel et al., 1989), (Peddinti et al., 2015) in one network. We show the results of two Chain-Model configurations:

  - **Chain A**: The network consists of 10 layers (7 TDNN-Layer and 3 LSTM-Layer). See (Cheng et al., 2017)) for a detailed description of the setup.
  - **Chain B**: The network architecture is equal to **Chain A** except for the usage of the (default) LSTM-Layer-implementation *LSTMp* instead of *FastLSTMp*, the application of per-frame-dropout, as described in (Cheng et al., 2017), and minor training parameters changes.

The LSTM-Layers in all networks have a cell-dimension of 1024 and a projection-dimension of 256 for unidirectional topologies. In bidirectional networks a projection-dimension of 128 is used.
The models are trained on 40-dimensional MFCC features with 5 consecutive frames at each time-step append with a 100-dimensional iVector (Dehak et al., 2011). This gives a 300-dimensional input at each time-step.
As in our previous works we train the acoustic models on the GerTV1000h corpus. However, due to the computational time needed for training sophisticated neural networks we decided to use a 128 h subset of the GerTV1000h corpus for the experiments in this work. For each experiment the development set (Dev Set) is used to adjust the Language-Model-weight to a fixed value across all test-sets within the respective experiment.

5.2. Previous Results

In our previous work we focused on speech recorded in the broadcast-domain. A detailed description of this work is given in (Schmidt et al., 2016). As described before, recordings in the broadcast-domain can be considered very clean. Therefore, for evaluation we mostly used the clean subset of the DiSCo corpus in our previous work. The word error rates of different acoustic models trained in our previous work are summarized in table 1. All these models were trained on the entire GerTV1000h corpus. For evaluation the same language model for evaluation was used - except for the last row in the table, where we applied a lower pruning factor (*Prun. Fact.*) on the language model.

Model	**Prun. Fact.**	**Dev Set**	**DiSCo clean**	
			plan.	spont.
HMM-DNN*	1e-7	21.3	15.3	19.4
HMM-pDNN	1e-7	18.8	13.3	16.5
CTC-RNN	1e-7	18.1	12.8	15.4
CTC-RNN	1e-8	**17.2**	**11.9**	**14.5**

Table 1: Overview of the word error rates of our previous acoustic modeling approaches trained on the entire GerTV1000h Corpus

The HMM-DNN-model marked with * is the acoustic model used in the current production system we aim to improve. This baseline-system gives a 55.0% word error rate on the Oral History test-set.

5.3. Training State-of-the-Art Acoustic Models On a 128 h Training-Data Subset

First we analyze the influence of using the previously described sophisticated neural network architectures for the hybrid HMM-DNN acoustic modeling approaches. We trained all models on a 128 h subset of the GerTV1000h corpus to save computational time. All experiments were performed using the same 500,000 words language model that was already used in our previous works. We used 1e-8 as the pruning factor since we achieved best results with this configuration in our previous work. The word error rates for all test-sets are summarized in table 2.

Model	Dev Set	DiSCo clean		DiSCo mixed		Oral Hist.
		plan.	spont.	plan.	spont.	
LSTM	16.6	12.1	13.9	16.7	26.0	**45.3**
BLSTM	17.2	12.8	14.6	17.6	27.6	48.2
Chain A	17.0	11.6	13.8	17.4	28.9	53.4
Chain B	**15.5**	**10.6**	**12.4**	**15.1**	**25.0**	50.2

Table 2: Overview of the word error rates using sophisticated acoustic modeling approaches trained on a 128 h subset of the GerTV1000h corpus

The best results on the DiSCo test-sets are achieved using the Chain B model. Both Chain models outperform our previously best model (Table 1, CTC-RNN with 1e-8 PF) on the DiSCo clean planned and spontaneous test-sets, even though the Chain models were only trained on a 128 h subset of the GerTV1000h corpus.
However, the Chain models perform rather bad on the Oral History test-set compared to the simpler HMM-LSTM-Models. One possible reason could be some kind of overfitting of the Chain models to the clean training data. The unidirectional LSTM-Model achieves the best word error rate on the Oral History test-set (45.3%) reducing the word error rate by 9.7% absolute and 17.6% relative compared to the baseline system.
Surprisingly, the bidirectional LSTM model performs worse than the unidirectional LSTM on all test-sets. This could be a lack of generalization during training due to the reduced size of training data using the 128 h subset. However since both the LSTM and Chain B model perform better than the BLSTM, we neglect the BLSTM in the following experiments.

5.4. Training Robust Acoustic Models On 128h Multi-Condition Data

With the following set of experiments we analyze the influence of multi-condition training on robustness of the acoustic models. These experiments are all carried out training the LSTM acoustic model setup that achieved best results on the Oral-History test-set in the previous experiments. We assume that the results of these multi-condition training generalize to other network architectures and training on the entire GerTV1000h Corpus as well. This will be analyzed in future our work.
We apply the data augmentation techniques described in section 4.2. to artificially distort the utterances of the GerTV1000h corpus creating three noisy variants of the corpus:

- **Reverb**: All signals are convolved according to equation (1) with randomly selected room impulse responses of small or medium-sized rooms. No noise is applied here.
- **R+AWGN**: Similar to **Reverb** but added white Gaussian noise to the signals.
- **R+RealNoise**: Similar to **R+AWGN** but instead of AWGN non-stationary noises are added that have been randomly selected from real-life recordings, e.g. street noises, noises in a bus, police sirens, hairdryers.

The noises in **AWGN** and **RealNoise** were also convolved with a room-impulse-response before superposing them with the reverberant speech signal. According to equation (2) we used a different room-impulse-response of the same room, than the one applied to the speech-signal, if one was available. Otherwise we used the same room impulse response. A random signal-to-noise ratio between 10 and 20 was applied for the superposition.
We used 266 room-impulse-responses of small and medium-sized rooms collected from different sources. The real-life noises contain 14.5 h of recordings. To avoid overfitting to the noises we superposed up to three different randomly selected noises for one audio file.
In our multi-condition experiments we want to analyze the influence of training acoustic models using different mixtures of the **Clean**, **Reverb**, **R+AWGN** and **R+RealNoise** data-sets. Thus the data-sets we actually use for training are created by randomly selecting each file from one of the four different data-sets using one of the distributions set out in table 3. This way we created four different multi-condition training-sets (v1 to v4) beside the clean one. For the 128 h subset we then selected the same utterances that were used in the previous (clean) experiments.

Conf.	Clean	Reverb	R+AWGN	R+RealNoise
Clean	100 %	0 %	0 %	0 %
v1	50 %	50 %	0 %	0 %
v2	40 %	40 %	20 %	0 %
v3	35 %	35 %	15 %	15 %
v4	40 %	40 %	0 %	20 %

Table 3: Overview of the different multi-condition-training configurations used for the GerTV1000h Corpus

Conf.	Dev Set	DiSCo clean		DiSCo mixed		Oral Hist.
		plan.	spont.	plan.	spont.	
Clean	**16.6**	**12.1**	**13.9**	**16.7**	**26.0**	45.3
v1	17.1	12.4	14.4	16.9	27.0	40.3
v2	17.3	12.7	14.5	16.8	26.7	39.7
v3	17.4	12.8	15.1	17.3	26.8	40.1
v4	17.3	12.4	14.6	16.7	26.5	**39.4**

Table 4: Word error rates of an LSTM-Model trained on 128 h training data using different multi-condition configurations

The word error rates of all test-sets and all models trained on the different multi-condition training-data-sets are summarized in table 4. It is very promising that all LSTMs generalized well being trained on the different presented configurations. On the one hand the difference of word error rates on the Oral History test-set between v1, v2, v3 and v4 is below 1% absolute. On the other hand the results on all four DiSCo test-sets remain well compared to the clean-

trained model. For v4 the word error rate on DiSCo increases by 0.7% absolute in the worst case while achieving the best result on the Oral History test-set (39.4%). With this configuration we were able to reduce the word error rate on the Oral History test-set furthermore by 5.9% absolute compared to the clean trained LSTM. Overall we improved the word error rate by 15.6% absolute and 28.3% relative compared to the baseline-system.

6. Conclusion and Outlook

In this work we presented the opportunities and advantages offered by automatic speech recognition systems, such as the Fraunhofer IAIS Audio Mining system, for Digital Humanities Research by automatically creating segmented and time-aligned transcriptions of Oral History interviews. We also present the challenges caused by Oral History interviews for ASR-systems. State-of-the-art acoustic modeling approaches such as sophisticated neural network architectures and multi-condition training were applied to cope with these challenges. Evaluated the systems on our proposed Oral History ASR test-set, we were able to improve the word error rate by 15.6% absolute and 28.3% relative compared to the baseline-system while only one eighth (128 h) of the previous amount of the training data was used.

We plan to utilize the Fraunhofer IAIS Audio Mining system to automatically process all not-transcribed interviews of the archives "Deutsches Gedächtnis" of the Institute for History and Biography. Future user studies need to show if the automatic transcription can furthermore be used as a basis for creating a perfect (time-aligned) transcript by human error-correction in the productive use.

However, to achieve satisfactory results the speech recognition performance has to be improved further. Therefore our future work aims to further decrease the word error on Oral History interviews. We want to further improve robust acoustic modeling, e.g. by training models on the entire amount of the GerTV1000h training data and run experiments to further increase the amount of training-data by a factor of 3 or 4 using the proposed data-augmentation methods. Moreover we plan to train domain-specific language models for Oral History interviews.

We also plan to time-align the manually transcribed but not time-aligned interviews of the "Deutsches Gedächtnis" archives using *forced alignment*-approaches for very long audio-files. This would provide new opportunities for Oral History research and could also be utilized for acoustic model training.

Acknowledgement

This research has been funded by the Federal Ministry of Education and Research of Germany (BMBF) in the project *KA*^{3} *- Kölner Zentrum für Analyse und Archivierung von AV-Daten* (Cologne center for the analysis and archiving of audiovisual data) (project number: 01UG1511B).

7. Bibliographical References

Cheng, G., Peddinti, V., Povey, D., Manohar, V., Khudanpur, S., and Yan, Y. (2017). An exploration of dropout with lstms. In *Proc. Interspeech 2017*, pages 1586–1590.

Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, May.

Gers, F. A. and Schmidhuber, J. (2000). Recurrent nets that time and count. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, volume 3, pages 189–194 vol.3.

Gers, F. A., Schmidhuber, J., and Cummins, F. (2000). Learning to forget: Continual prediction with lstm. *Neural Computation*, 12(10):2451–2471, Oct.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052 vol. 4, July.

Graves, A., Fernández, S., and Gomez, F. (2006). Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *In Proceedings of the International Conference on Machine Learning, ICML 2006*, pages 369–376.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780, November.

Li, B., Sainath, T. N., Narayanan, A., Caroselli, J., Bacchiani, M., Misra, A., Shafran, I., Sak, H., Pundak, G., Chin, K., Sim, K. C., Weiss, R. J., Wilson, K. W., Variani, E., Kim, C., Siohan, O., Weintraub, M., McDermott, E., Rose, R., and Shannon, M. (2017). Acoustic modeling for google home. In *Proc. Interspeech 2017*, pages 399–403.

Miao, Y., Gowayyed, M., and Metze, F. (2015). Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 167–174, Dec.

Oard, D. (2012). Can automatic speech recognition replace manual transcription? In Doug Boyd, et al., editors, *Oral History in the Digital Age*, Washington, D.C. Institute of Museum and Library Services.

Peddinti, V., Povey, D., and Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts. In *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, pages 3214–3218.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, December. IEEE Catalog No.: CFP11SRW-USB.

Povey, D., Peddinti, V., Galvez, D., Ghahremani, P., Manohar, V., Na, X., Wang, Y., and Khudanpur, S. (2016). Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Interspeech 2016*, pages

2751–2755.

Psutka, J., Ircing, P., Psutka, J. V., Radová, V., Byrne, W. J., Hajič, J., Gustman, S., and Ramabhadran, B. (2002). Automatic transcription of czech language oral history in the malach project: Resources and initial experiments. In Petr Sojka, et al., editors, *Text, Speech and Dialogue: 5th International Conference, TSD 2002 Brno, Czech Republic, September 9–12, 2002 Proceedings*, pages 253–260, Berlin, Heidelberg. Springer Berlin Heidelberg.

Sak, H., Senior, A. W., and Beaufays, F. (2014). Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *CoRR*, abs/1402.1128.

Saon, G., Kurata, G., Sercu, T., Audhkhasi, K., Thomas, S., Dimitriadis, D., Cui, X., Ramabhadran, B., Picheny, M., Lim, L.-L., Roomi, B., and Hall, P. (2017). English conversational telephone speech recognition by humans and machines. In *Proc. Interspeech 2017*, pages 132–136.

Schmidt, C. A., Stadtschnitzer, M., and Köhler, J. (2016). The fraunhofer iais audio mining system: Current state and future directions. *Speech Communication; 12. ITG Symposium, Paderborn, Germany, 2016*, 12:115–119.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, Nov.

Tritschler, A. and Gopinath, R. (1999). Improved speaker segmentation and segments clustering using the bayesian information criterion. In *EUROSPEECH'99*, Budapest, Hungary, september. ISCA.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K. J. (1989). Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):328–339, Mar.

8. Language Resource References

Baum, Doris and Schneider, Daniel and Bardeli, Rolf and Schwenninger, Jochen and Samlowski, Barbara and , Winkler Thomas and Köhler, Joachim. (2010). *DiSCo - A German Evaluation Corpus for Challenging Problems in the Broadcast Domain*. Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10).

Stadtschnitzer, Michael and Schwenninger, Jochen and Stein, Daniel and Köhler, Joachim. (2014). *Exploiting the Large-Scale German Broadcast Corpus to Boost the Fraunhofer IAIS Speech Recognition System*. Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14).

Sound Signal Processing with Seq2Tree Network

Weicheng Ma[†], Kai Cao[‡], Zhaoheng Ni[◇], Peter Chin[†], Xiang Li[‡]
[†]Boston University
[‡]Cambia Health Solutions
[◇]City University of New York
[†]{wcma, spchin}@bu.edu, [◇]zni@gradcenter.cuny.edu
[‡]{kai.cao, xiang.li}@cambiahealth.com

Abstract

Long Short-Term Memory (LSTM) and its variants have been the standard solution to sequential data processing tasks because of their ability to preserve previous information weighted on distance. This feature provides the LSTM family with additional information in predictions, compared to regular Recurrent Neural Networks (RNNs) and Bag-of-Words (BOW) models. In other words, LSTM networks assume the data to be chain-structured. The longer the distance between two data points, the less related the data points are. However, this is usually not the case for real multimedia signals including text, sound and music. In real data, this chain-structured dependency exists only across meaningful groups of data units but not over single units directly. For example, in a prediction task over sound signals, a meaningful word could give a strong hint to its following word as a whole but not the first phoneme of that word. This undermines the ability of LSTM networks in modeling multimedia data, which is pattern-rich. In this paper we take advantage of Seq2Tree network, a dynamically extensible tree-structured neural network architecture which helps solve the problem LSTM networks face in sound signal processing tasks—the unbalanced connections among data units inside and outside semantic groups. Experiments show that Seq2Tree network outperforms the state-of-the-art Bidirectional LSTM (BLSTM) model on a signal and noise separation task (CHiME Speech Separation and Recognition Challenge).

Keywords: Deep Learning, Tree Model, Dynamic Neural Network

1. Introduction

Recent RNN-based approaches are achieving high performance in speech processing tasks, including but not limited to the signal and noise separation task (Erdogan et al., 2015; Zhu and Vogel-Heuser, 2014; Wu et al., 2015; Barker et al., 2015). The underlying hypothesis is that the energy in each frequency bin over a period of time is continuous and predictable. However, in real life scenes noises can break in at any time and intertwine with the sound signal with no predictable pattern, which undermines these models' ability to predict the distribution of noise over frequency bins.

To address the problem of finding correct boundaries of noises, some variants of the original LSTM network are used. The current state-of-the-art system on this task applies BLSTM, which tries to bound noises by foreseeing future information (Erdogan et al., 2015; Weninger et al., 2015; Grais et al., 2014). Nevertheless, information from the future also contains more distant sound signals, which does not solve the signal superposition problem. Furthermore, we believe the future for speech processing should be dominated by real-time speech processing techniques, which BLSTM models are not able to handle.

What's more important, phonemes in sound signals make no sense if not combined into "words", which are not found in noise signals. So, the sound waveforms should not be understood as a chain of phonemes, but rather on a "word" level. This leads to a natural choice of tree structured modeling of the waveforms.

In this paper we introduce two variants of Seq2Tree (Ma et al., 2018b; Ma et al., 2018a), a novel architecture which extends LSTM networks to be able to parse sequential input into a tree structure and show its superiority in decomposing sound and noise signals. Seq2tree network architecture differs from the standard LSTM since each node inherits the hidden state not from the previous state in time sequence but from its parent in the tree structure, based on its position predicted by the network itself.

Our evaluation demonstrates the advancement of Seq2Tree network compared to the BLSTM baseline on the signal and noise separation task (Barker et al., 2015). Experiments show that our system shows comparable performance to the BLSTM implementation, while outperforming it in more complex scenarios. Further optimization and adjustment to this task will follow.

2. LSTM Network

RNNs have the advantage of processing input sequences regardless of their lengths. The network reads an element in a sequence at a time and passes it to an activation function recursively together with the current state of this network. The sequence and elements can be of arbitrary types—for example, phonemes in a piece of sound when it comes to the task of speech processing.

Generally the input elements are represented as vectors, and the state at a certain time t is a distributed representation with a preset dimension d. Based on the recurrent nature of RNNs, the state at time t stores the information from all the states before time t. The commonly accepted activation functions in RNNs are often an affine transformation of the previous state h_{t-1} and the current input x_t combined with a non-linear function σ:

$$h_t = \sigma(Wx_t + Uh_{t-1} + b). \quad (1)$$

Though RNNs are designed to store previous information, they are easily trapped by the explosive growth or rapid

vanishment of the gradient over long distances (Hochreiter, 1998; Hochreiter et al., 2001). This makes it difficult for RNNs to represent long-term information.

The LSTM network is introduced to deal with this problem (Hochreiter, 1998; Hochreiter et al., 2001; Zaremba and Sutskever, 2014; Zaremba et al., 2014). Different from directly passing the previous state and the current input to the transition function on which the gradient is calculated, LSTM uses a memory cell to preserve the longer-term information. Using the settings in (Zaremba and Sutskever, 2014; Zaremba et al., 2014), the LSTM transition functions are as follows:

$$
\begin{aligned}
i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{t-1} + b^{(i)}, \\
f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_{t-1} + b^{(f)}, \\
o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{t-1} + b^{(o)}, \\
u_t &= tanh(W^{(u)}x_t + U^{(u)}h_{t-1} + b^{(u)}, \\
c_t &= i_t \odot u_t + f_t \odot c_{t-1}, \\
h_t &= u_t \odot tanh(c_t)
\end{aligned}
\tag{2}
$$

where i_t, f_t, o_t, c_t are the input gate, forget gate, output gate and the memory cell, respectively, and $\odot$ refers to element-wise multiplication. In the equations, the input gate decides how much information from the new input will be added to the memory cell. Similarly, the forget gate f controls how much information to forget from the previous states, and the output gate limits the amount of information to expose. By balancing the incoming and outgoing information amount, LSTM is able to prevent the gradient vanishment and explosion problems.

Ordinary LSTM is based on chain-structured sequences. There exist two common variants of LSTM networks by structure, namely BLSTM and Multilayer LSTM, which combines multiple LSTM networks together to provide additional information in the prediction at each time step. Tree LSTM (Tai et al., 2015) could be regarded as one variation of Multilayer LSTM with the dependency relation reversed.

3. Seq2Tree Network

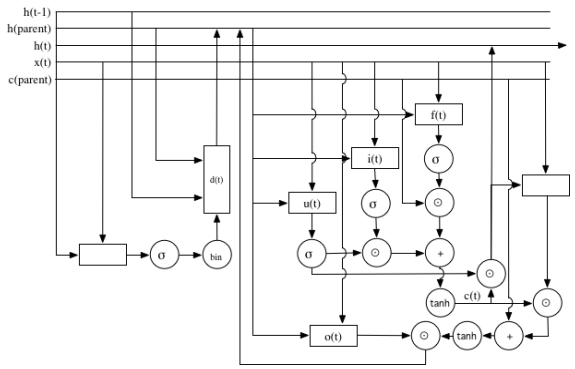


Figure 1: Seq2Tree Network Architecture

The LSTM architectures described in the previous section all have limits in constructing a tree structure from sequential input. Though Multilayer LSTM and Tree LSTM networks are able to maintain multilevel dependencies, Multilayer LSTM exposes children cells to all the other units, and Tree LSTM requires tree-structured input. These characteristics limit their use in speech processing tasks where no reliable parser exists, especially in the case of online speech processing. Hence we in this paper apply Seq2Tree network, a dynamic tree-structured neural network architecture we developed, on sound signal processing tasks. Because of the structural characteristics of sound signal data, we in this paper introduce two variants to the general idea of Seq2Tree network—Single Level Seq2Tree and Multilayer Seq2Tree architectures. Both variants are able to find dependencies from adjacent signals, while the Multilayer Seq2Tree architecture catches deeper, weaker-bounded correlations.

Similar to original LSTM networks, at each time step our Seq2Tree architecture passes information from a preceding state with a hidden unit h_t, accepts new information from the input x_t gated by an input gate i_t, controls the output by an output gate o_t, drops unimportant data in an amount decided by the forget gate f_t, and keeps long-term information from the beginning of the input sequence in a memory cell c_t. The difference is that instead of taking the previous state as the preceding state, Seq2Tree networks use one additional direction gate d_t to choose the direction to go at time step t. The path selection gate is implemented differently in Single Level Seq2Tree and Multilayer Seq2Tree architectures.

3.1. Single Level Seq2Tree

The Single Level Seq2Tree architecture allows at most a depth of 1 for all the nodes in the generated tree structure. It is based on a simplified hypothesis that children states under a parent state do not affect outer units. On speech processing tasks, for example, this means no two signals overlap each other if they are not within the exact same phase.

Since the height of the tree is limited to 2, at each step there exist only 2 possible directions to go: up and down. The strategy is that if the predicted direction to go is "up", the parent node's hidden state becomes the input hidden state h_{t-1} and the parent hidden state is assigned the hidden state of the current input after processing it. If the direction is "down" while the previous state is already a child node, the new node becomes the sibling of the former one and inherits the hidden state from the previous state. Otherwise, the new unit takes the hidden state from its previous neighbor and becomes the child of its preceding state.

After processing each state, its parent node's information is updated. The forget gate of the child state f_t controls the amount of change to give its parent state. This mechanism is inspired by the Tree LSTM networks. The transition functions of the Single Level Seq2Tree network are as follows:

$$
\begin{aligned}
d_t &= bin(\sigma(W^{(d)}x_t + U^{(d)}h_{t-1} + b^{(d)})), \\
h_{parent} &= d_t \begin{pmatrix} h_{parent} \\ h_{t-1} \end{pmatrix}, \\
i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{parent} + b^{(i)}),
\end{aligned}
$$

$$
\begin{aligned}
f_t &= \sigma(W^{(f)}x_t + U^{(f)}(d_t \begin{pmatrix} \mathbf{0} \\ h_{parent} \end{pmatrix}) + b^{(f)}),\\
o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{parent} + b^{(o)}),\\
u_t &= tanh(W^{(u)}x_t + U^{(u)}h_{parent} + b^{(u)}),\\
c_t &= i_t \odot u_t + f_t \odot c_{parent},\\
h_t &= u_t \odot tanh(c_t),\\
\Delta f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_t + b^{(f)}),\\
c_{parent} &= c_{parent} + \Delta f_t \odot c_t,\\
h_{parent} &= o_{parent} \odot tanh(c_{parent}).
\end{aligned} \tag{3}
$$

where bin denotes a binary threshold activation function, σ represents the sigmoid function, and $\odot$ is element-wise multiplication.

3.2. Multilayer Seq2Tree

The Single Level Seq2Tree architecture can perfectly model the superposition of signals without an overlap of three or more signals with different phases. However, in speech processing tasks the boundaries of noise signals are not necessarily distinct from each other.

To model the more complicated scenarios, a deeper tree structure is needed so that when noises overlap with each other, the layer $l+1$ represents phonemes which come before the $l-th$ sound waveform ends. This architecture is an extension to the Single Level Seq2Tree architecture at the point that at each time step, there exist three directions instead of two. Multiple jumps towards the root in one time step is also allowed. Moreover, at each jump an update gate is used to control the amount of change on a parent layer, and the remainder is passed to even higher states in the tree if there are further jumps. The transition functions differ from those of Single Level Seq2Tree on parent state selection and parent state update mechanisms:

$$
\begin{aligned}
d_{kt} &= bin(\sigma(W^{(d)}x_t + U^{(d)}h_k + b^{(d)})),\\
h_{parent} &= \prod_{k=0}^{d_{kt}\neq\binom{0}{0}} d_{kt} \begin{pmatrix} h_{t-k-1} \\ h_{t-k} \end{pmatrix},\\
i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{parent} + b^{(i)}),\\
f_t &= \sigma(W^{(f)}x_t + U^{(f)}(d_{0t} \begin{pmatrix} \mathbf{0} \\ h_{parent} \end{pmatrix}) + b^{(f)}),\\
o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{parent} + b^{(o)}),\\
u_t &= tanh(W^{(u)}x_t + U^{(u)}h_{parent} + b^{(u)}),\\
c_t &= i_t \odot u_t + f_t \odot c_{t-1},\\
h_t &= u_t \odot tanh(c_t),\\
\Delta f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_t + b^{(f)}),\\
\Delta c_t &= \Delta f_t \odot c_t,\\
c_{kt} &= c_{kt} + \Delta c_t - \sum_{i=0}^{k-1} u_i \Delta c_t,\\
h_{kt} &= o_{kt} \odot tanh(c_{kt}).
\end{aligned} \tag{4}
$$

where kt indexes the parent nodes in the path until the root from the node at time step t, and d_{0t} represents the gate d at the current time step under the selected parent node.

4. Task and Model

4.1. Signal and Noise Separation Task

We test our Seq2Tree architecture on the signal and noise separation task, the goal of which is to predict a mask which weakens the energy of noise when applied to the input sound. The task is defined in the Second CHiME Speech Separation and Recognition Challenge (Vincent et al., 2013).

4.2. Tree2Seq Signal-Masking Model

For this task, at each time step t we want to predict a mask over all frequency bins. We achieve this by training a softmax regression matrix which takes the current hidden state:

$$mask = softmax(U^{(R)}h)$$

where $U^{(R)}$ is the regression matrix.

We train our signal-masking model in two stages using two different loss calculations, as is suggested in (Weninger et al., 2014; Erdogan et al., 2015). The two losses we applied are:

$$
\begin{aligned}
J_1(t) &= -\frac{1}{c}\sum_{i=1}^{c}(mask_i - label_i)^2\\
J_2(t) &= -\frac{1}{c}\sum_{i=1}^{c}(\|x_t\|(mask_i - label_i))^2
\end{aligned}
$$

where c is the number of frequency bins, $mask_i$ is the predicted mask at time t for bin i and $label_i$ is the labeled mask on bin i at time t.

5. Experiments

We evaluate our Seq2Tree architecture on the signal and noise separation task. The data is a fraction of 1500 audio files from the CHiME dataset (Vincent et al., 2016), in which 10% is used for test and the rest for training. Each input file is Fourier Transformed and fed to the models. Every model predicts a mask given the input matrix. The quality of the mask is evaluated in terms of Overall Perceptual Score (OPS) by applying the mask onto the source waveform, given the noise-removed audio gold standard (Emiya et al., 2011). In our experiment, the shape of the training data is 50×513, representing the energy at 50 time steps in 513 frequency bins. The test data has variable length over time steps, taking advantage of LSTM models' ability to deal with variable length inputs.

We compare the results generated by our Single Level Seq2Tree with those output by the BLSTM baseline. The hidden layer size for our Seq2Tree network is set to 1024, and we list the results with different numbers of iterations. The BLSTM baseline applies a 256 hidden layer size, and is trained for 30 epochs. Due to long training time cost, our Multilayer Seq2Tree model for this task is only trained for 20 runs with the same parameter settings as our Single Level Seq2Tree model. Best and worst scores of our Single Level model are also included.

As is shown in the results table, our Single Level Seq2Tree model has comparable performance to the BLSTM implementation. The accuracy increases with more training iterations, indicating a preference of more training data and

Implementation	OPS(dB)
BLSTM	25.01
Single Level Seq2Tree	25.13
Single Level Seq2Tree (Worst Case)	23.87
Single Level Seq2Tree (Best Case)	27.96
Multilayer Seq2Tree	24.41

Table 1: Evaluation Results.

more training epochs. Also when looked into the specific WAV files, in more complex cases where noises overlap with each other, our Multilayer Seq2Tree model largely outperformed the other models, which agrees with our estimation. Further experiments are needed to demonstrate the effectiveness of the Multilayer Seq2Tree architecture on the noise separation task.

6. Conclusion & Future Work

In this paper, we introduced a generative tree-structured LSTM network architecture. The Seq2Tree architecture can be applied to arbitrary sequential input with potential local dependencies among nodes. We demonstrated its effectiveness by evaluating two Seq2Tree-based models on a signal and noise separation task. However, due to time constraints we are only able to thoroughly study the performance of the Single Level Seq2Tree architecture. Our results are comparable to the current state-of-the-art model in this task, though leaving some minor errors indicating a preference to the multilayer tree structure and the need for more careful parameter tuning. We will keep refining our model to fit the noise separation task, and we will try to expand the use of our Seq2Tree architecture to other tasks. Since deep learning model have widely been used in AI tasks, we propose that the seq2tree model can be used in different NLP tasks, such as NLP tasks and multimedia tasks. Syntactic structures have been implemented with deep neural networks and applied to build tree-strutured LSTMs, however tree-structured LSTMs have not been applied to syntactic parsers. In the future, we are going to build a Seq2Tree based dependency parser. Dependency parsers have been utilized in quite a few NLP tasks such as Relation Extraction and Event Extraction systems. For example, (Cao et al., 2015) and (Cao et al., 2016) includes syntactic relations with dependency regularizations in event detection systems. Deep neural networks have also applied in semantic relations such as Abstract Meaning Representation parsers. The Seq2Tree structure can also be applied in AMR parsing because the AMR semantic structure is also a tree. AMR parser is widely explored with different NLP tasks such as event detection (Li et al., 2015) and natural language generation (Flanigan et al., 2016).

7. Bibliographical References

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (2015). The third 'chime' speech separation and recognition challenge: Dataset, task and baselines. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 504–511. IEEE.

Cao, K., Li, X., and Grishman, R. (2015). Improving event detection with dependency regularization. In *Proceedings of RANLP*.

Cao, K., Li, X., and Grishman, R. (2016). Leveraging dependency regularization for event extraction. In *Proceedings of FLAIRS*.

Emiya, V., Vincent, E., Harlander, N., and Hohmann, V. (2011). Subjective and objective quality assessment of audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2046–2057.

Erdogan, H., Hershey, J. R., Watanabe, S., and Le Roux, J. (2015). Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 708–712. IEEE.

Flanigan, J., Dyer, C., and Smith, Noah A.and Carbonell, J. (2016). Generation from abstract meaning representation using tree transducers. In *HLT-NAACL*.

Grais, E. M., Sen, M. U., and Erdogan, H. (2014). Deep neural networks for single channel source separation. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 3734–3738. IEEE.

Hochreiter, S., Bengio, Y., Frasconi, P., and Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116.

Li, X., Nguyen, T. H., Cao, K., and Grishman, R. (2015). Improving event detection with abstract meaning representation. In *Proceedings of ACL-IJCNLP Workshop on Computing News Storylines (CNewS 2015)*.

Ma, W., Cao, K., Ni, Z., Li, X., and Chin, P. (2018a). Tree structured multimedia signal modeling. In *The Florida Artificial Intelligence Research Society*.

Ma, W., Cao, K., Ni, Z., Ni, X., and Chin, P. (2018b). Sound signal processing based on seq2tree networks. In *Interspeech workshop on Vocal Interactivity in-and-between Humans, Animals and Robots*.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Vincent, E., Barker, J., Watanabe, S., Le Roux, J., Nesta, F., and Matassoni, M. (2013). The second ?chime?speech separation and recognition challenge: An overview of challenge systems and outcomes. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 162–167. IEEE.

Vincent, E., Watanabe, S., Nugraha, A. A., Barker, J., and Marxer, R. (2016). An analysis of environment, microphone and data simulation mismatches in robust speech recognition. *Computer Speech & Language*.

Weninger, F., Hershey, J. R., Le Roux, J., and Schuller, B. (2014). Discriminatively trained recurrent neural networks for single-channel speech separation. In *Sig-*

nal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on, pages 577–581. IEEE.

Weninger, F., Erdogan, H., Watanabe, S., Vincent, E., Le Roux, J., Hershey, J. R., and Schuller, B. (2015). Speech enhancement with lstm recurrent neural networks and its application to noise-robust asr. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99. Springer.

Wu, M., Liu, Z., Xu, L., and Chen, D. (2015). Accurate and cost-effective technique for jitter and noise separation based on single-frequency measurement. *Electronics Letters*, 52(2):106–107.

Zaremba, W. and Sutskever, I. (2014). Learning to execute. *arXiv preprint arXiv:1410.4615*.

Zaremba, W., Sutskever, I., and Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zhu, K. and Vogel-Heuser, B. (2014). Sparse representation and its applications in micro-milling condition monitoring: noise separation and tool condition monitoring. *The International Journal of Advanced Manufacturing Technology*, 70(1-4):185–199.

Open ASR for Icelandic: Resources and a Baseline System

Anna Björk Nikulasdóttir, Inga Rún Helgadóttir, Matthías Pétursson, Jón Guðnason
Reykjavik University
Reykjavik Iceland
{annabn, ingarun, matthias, jg}@ru.is

Abstract

Developing language resources is an important task when creating a speech recognition system for a less-resourced language. In this paper we describe available language resources and their preparation for use in a large vocabulary speech recognition (LVSR) system for Icelandic. The content of a speech corpus is analysed and training and test sets compiled, a pronunciation dictionary is extended, and text normalization for language modeling performed. An ASR system based on neural networks is implemented using these resources and tested using different acoustic training sets. Experimental results show a clear increase in word-error-rate (WER) when using smaller training sets, indicating that extension of the speech corpus for training would improve the system. When testing on data with known vocabulary only, the WER is 7.99%, but on an open vocabulary test set the WER is 15.72%. Furthermore, impact of the content of the acoustic training corpus is examined. The current results indicate that an ASR system could profit from carefully selected phonotactical data, however, further experiments are needed to verify this impression. The language resources are available on http://malfong.is and the source code of the project can be found on https://github.com/cadia-lvl/ice-asr/tree/master/ice-kaldi.

Keywords: language resources, automatic speech recognition, Icelandic

1. Introduction

With advances in automatic speech recognition (ASR) technology and access to open source ASR tools like Kaldi[1] (Povey et al., 2011), the threshold for the development of ASR systems has been lowered substantially. The core ASR algorithms, including state-of-the-art neural network architectures, are already implemented, leaving the development of language resources as the main task.

The development of sufficient language resources, however, poses a challenge for less-resourced languages. Two specialized resources are needed for automatic speech recognition: a speech corpus with matching units of speech and text, and a pronunciation dictionary. Additionally, a large text corpus is needed for the development of a language model.

In this paper we describe the development of language resources for an Icelandic large vocabulary speech recognition system (LVSR), and the setup of a baseline system in Kaldi.

The first speech recognizer for Icelandic was *Hjal*, developed in 2003 as an isolated word recognizer (Rögnvaldsson, 2003). The *Hjal* project was successful as a pilot project, but has not been maintained. Currently, three ASR systems are being developed for Icelandic using Kaldi: an ASR system for parliamentary speeches (Helgadóttir et al., 2017), a system for automating transcriptions of radiology dictations[2], and the LVSR project described in this paper. Google already supports Icelandic in its speech recognition applications[3].

We inspect available language resources for Icelandic and prepare them for use in a speech recognition system. After defining the characteristics of the resources and testing an initial setup of an LSVR system, we make suggestions for further development of the resources. In particular we address three questions: a) is the available speech corpus sufficiently large for the development of a state-of-the-art LVSR system? b) is the pronunciation dictionary large enough for an LVSR system? and c) what is the impact of the content of the speech corpus on the results of the system?

2. Data

To develop and train an ASR system three kinds of language resources are needed: a speech corpus to train an acoustic model, a text corpus for language modeling, and a pronunciation dictionary. For the current project available resources for Icelandic were examined.

2.1. Málrómur Speech Corpus

During the years 2010-2011 the Reykjavik University collected Icelandic speech samples in cooperation with Google (Guðnason et al., 2012; Steingrímsson et al., 2017). Over 550 speakers of both genders and various age recorded a collection of sentences and isolated words using smartphones. The version of Málrómur used in this project has been cleaned of recordings marked as 'faulty' by the speakers themselves (normally empty recordings), and contains about 122,000 utterances or approx. 150 hours of speech. The recorded utterances represent 33,135 unique prompts read by the speakers. This list of prompts is composed as follows: the largest part are short sentences from news articles, then carefully selected sentences regarding diphone and triphone distribution (developed during the *Hjal* project (Rögnvaldsson, 2003)), proper names, Icelandic streets and locations, international cities and countries, numbers, miscellaneous entries, and a list of URLs. We call the sentence list from the *Hjal* project *phonotactical sentence list* since its purpose is to thoroughly represent the phonotactics of Icelandic. As shown in Fig. 1, the news sentences are 66% of all entries in the list of prompts, location names and proper names are about 20% and 10% respectively, and

[1] http://kaldi-asr.org/

[2] http://laeknaromur.is/en.html

[3] see e.g. https://cloud.google.com/speech/

other categories have much fewer entries. In the Málrómur speech corpus the distribution of categories changes. Most notably the few entries from the phonotactical list and the list of URLs are recorded multiple times, leading to their increasing partition of the speech corpus.

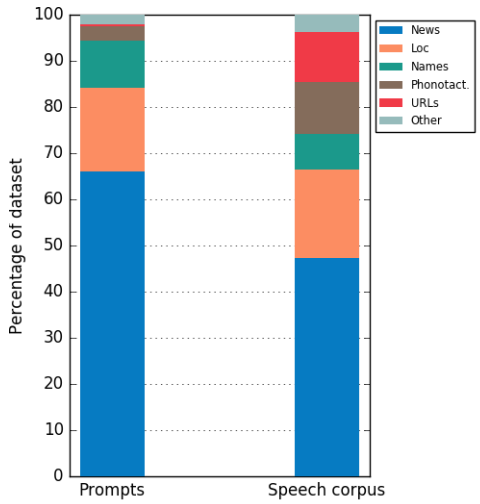


Figure 1: Distribution of prompt categories in the prompts list (~33k entries) and the Málrómur speech corpus (~122k entries). The three smaller categories, numbers, international cities and countries, and miscellaneous entries are subsumed under *Other*

The amount of repetitions is not evenly distributed among the prompts as shown in Table 1: 23% of the utterances have only one representation in the corpus and 74% are recorded 2 to 10 times, the remaining 3% are recorded up to over 100 times. When looking at the absolute numbers of recordings, 7,450 (6%) are unique, 83,752 (69%) recordings belong to the group of 2 to 10 times repeated utterances and almost 10,000 or 8% represent the 86 utterances repeated more than 100 times in the corpus.

Repetition	No. of prompts	No. of recordings
1x	7,450	7,450
2x	8,751	17,502
3-4x	11,090	37,428
5-10x	4,671	28,822
11-20x	937	12,866
21-50x	82	2,321
51-100x	68	6,001
>100x	86	9,495
TOTAL	33,135	121,822

Table 1: The repetition of prompts in the Málrómur corpus. Most prompts are recorded 1-4 times.

The prompts from news articles and the phonotactical sentences are multi word sentences but most of the prompts from the other categories are single word utterances. In the prompts list about 30% of the prompts are single words, the remaining 70% of the prompts contain 2 to 13 words. In the recorded speech corpus 40% of the utterances are single words and 60% multi word utterances. On average the multi word utterances contain 5.3 words.

2.2. LCC Text Corpus

The *Leipzig Corpora Collection (LCC)* (Goldhahn et al., 2012) contains an Icelandic text corpus, mainly collected from news sites. For this project we have access to 10 million sentences or 180 million tokens from this corpus. The sentences are scrambled, i.e. each sentence is isolated from its original context, and duplicate sentences have been removed.

2.3. Pronunciation Dictionary

During the *Hjal* project (see Section 1.) a pronunciation dictionary with almost 50,000 word forms was developed. The vocabulary selection was done using a frequency list especially compiled for the task, from news papers and other sources. It was transcribed using the SAMPA phonetic alphabet, but the dictionary has since then been extended to about 60,000 unique word forms with 65,000 entries. Furthermore, IPA transcriptions have been added.

3. Resource Processing

3.1. Speech Corpus Training and Test Sets

The largest part of the speech corpus is needed to train the speech recognition system. But a fraction of the data will have to be extracted to create test and development sets used for decoding experiments. A speech corpus created in the same way as Málrómur is expected to contain errors, i.e. not all recordings will exactly match the corresponding text (Hughes et al., 2010; Steingrímsson et al., 2017). By sufficient amount of data this is not necessarily a problem for the acoustic training set. For a test set, however, it is very important that the utterances match the corresponding texts to a high degree. An attempt was made to select high quality recordings for test and development sets. The aim was to collect utterances from reliable speakers for each of the two sets. This was done by listening to 3,000 random recordings from the corpus. Four evaluators (two researchers, two students) used the Eyra application (Pétursson et al., 2016; Guðnason et al., 2017) to rate the recordings and rate them from 1 (=bad) to 4 (=very good). The kappa (Fleiss) value for the evaluation results of the four evaluators was 0.598, which is at the lower limit for substantial agreement (Landis and Koch, 1977). Unifying grades 1 and 2 to one grade (bad) and grades 3 and 4 to one grade (good), a kappa value of 0.64 was computed. The results of the evaluation were used to identify speakers whose recordings generally match the prompts. From those, 40 speakers of each gender were randomly selected, having at least 100 sentence utterances and 50 one word utterances in the corpus. Two constraints were defined for the test and development sets: they should not contain utterances from the phonotactical sentence list and no URLs. The primary role of the phonotactical sentences is to collect acoustic material on as many valid di- and triphones as possible. The sentences, however, often contain very rare

words and combinations and are thus not likely to be predicted by a general language model. Another reason not to include phonotactical sentences in the test sets is that readers are more likely to make errors. As for the URLs, during this first experiment they were not normalized, making it impossible for the ASR system to decode them correctly. The final test and development sets consisted of 6,000 utterances from 20 female and 20 male speakers each. All utterances from the remaining speakers in the Málrómur corpus built the general acoustic training set, 88,285 utterances or approx. 109 hours of speech. Additionally, two experimental training sets were compiled from the general training set: one set were all sentences from the phonotactical sentence list were deleted and another set of the same size where the phonotactical sentences were kept but the corresponding number of news utterances were deleted. These sets contain 78,529 utterances each. The training set with the phonotactical sentences has approx. 95 hours of speech whereas the other set has about 93 hours. The phonotactical sentences are often longer than the news paper sentences, resulting in this difference in duration.

3.2. Text Normalization

The Icelandic LCC corpus was used for language modeling for the ASR system. When using a raw, unrestricted corpus for language modeling, one has to be aware of non-standard words, symbols and punctuation. In line with (Sproat et al., 2001) a taxonomy of non-standard words in Icelandic was developed to prepare text normalization. The normalization of Icelandic texts is a work in progress. Normalization tasks for non-standard words have been defined and for Icelandic, decompounding should also be considered (Adda-Decker and Adda, 2000; Ordelman et al., 2003). In this first setup of the ASR system only basic normalization was performed: all non-relevant symbols and punctuation was removed, common acronyms and abbreviations were replaced, and some corpus specific processing made to remove web page navigation tokens. The text was tokenized, thus separating punctuation from words, and lowercased. Next step would be to adapt the normalization model developed for ASR of Icelandic parliament speeches to replace digits with their spelled out forms (Helgadóttir et al., 2017). This is a non-trivial problem in Icelandic, since cardinal and ordinal numbers from one to four (and all larger numbers ending with these numbers) behave much like adjectives: they are inflected by case and gender and some even by number.

3.3. Extending the Pronunciation Dictionary

For a highly inflective language like Icelandic, a lexicon of 60,000 words is not sufficient for an LVSR system. The Database of Modern Icelandic Inflection contains almost 280,000 inflection paradigms with 2.8 million distinct word forms[4] (Bjarnadóttir, 2012). The text representation of the Málrómur corpus contains 435,000 tokens. The out-of-vocabulary rate (OOV) of this corpus when using the pronunciation dictionary is 19.5%, and of the 32,765 types 17,676 or 54% are contained in the dictionary. Some of the missing types are numbers, acronyms, URLs, foreign words or words with spelling errors. Nevertheless, an extension of the lexicon was necessary in order to reduce the OOV rate and increase lexical coverage. To select words to add to the lexicon two sources were used: a) the LCC text corpus and b) the text representation of the acoustic training set from Málrómur. The 200,000 most frequent words from LCC and all words occurring at least three times in the Málrómur training set were extracted and heuristics used to compile an extended lexicon. These include filtering out words with characters not contained in the Icelandic alphabet (*c, w, ä,* etc.), validation based on upper and lower case, analysis of common n-grams, etc. All words in the original lexicon not occurring in this new word list were deleted. Furthermore, as a general rule, for words that can be written either as one or two words, the one word version was deleted, e.g. *hinsvegar* vs. *hins vegar*. The resulting version of the lexicon contains 134,866 words, 46,209 from the original pronunciation dictionary and 88,657 new words.
For the transcriptions a fixed set of 58 IPA symbols was used. The transcriptions in the original pronunciation dictionary are somewhat inconsistent regarding this set of symbols, and thus some automatic correction procedures were run. Furthermore, some dialect variations were removed, these would need to be added again when training an ASR system where recordings from dialect speakers are included[5]. Overall, transcriptions of about 7,200 words were changed in some way, corrected and/or variations deleted. These revised transcriptions were used to train a grapheme-to-phoneme model using the Sequitur G2P converter[6] (Bisani and Ney, 2008). The extended lexicon was then automatically transcribed using this model. The pronunciation dictionary with the core content from the (corrected) manually transcribed dictionary plus the automatically transcribed word list generated from LCC and the Málrómur training corpus now contains 136,082 entries. This reduces the OOV rate for the whole Málrómur corpus from 19.5% to 8.9% and 76.2% of the types are now included in the pronunciation dictionary compared to 54% before.

4. Training a Baseline ASR System

DNNs are a current standard in acoustic modeling for ASR and represent a very active research field within the ASR community. We used the Kaldi ASR toolkit for the development of our baseline DNN system (Povey et al., 2011) and aligned the acoustic model training to a recipe developed for the Switchboard corpus[7]. The first step in the acoustic model training (after extracting the necessary Mel-Cepstrum-Coefficients (MFCCs)), is to train a hidden Markov model with Gaussian mixture models (HMM-GMM) and to apply Linear Discriminant Analysis (LDA) and Maximum Likelihood Linear Transform (MLLT) to the acoustic features. Then a sequence model based on time delay deep neural network (TD-DNN) (Waibel et al., 1989) layers and long-short term memory (LSTM) network (Sak

[4] http://bin.arnastofnun.is/DMII/

[5] Icelandic does, however, not have highly diverse dialects

[6] https://www-i6.informatik.rwth-aachen.de/web/Software/g2p.html

[7] https://github.com/kaldi-asr/kaldi/blob/master/egs/swbd/s5c/local/chain/tuning/run_tdnn_lstm_1e.sh

et al., 2014) is trained (Povey et al., 2016). The network takes 40 dimensional LDA feature vectors and a 100 dimensional ivector as input and it is composed of ten hidden layers, of which seven are TDNN layers and three are LSTM layers. Two language models were trained using the MITLM toolkit (Hsu and Glass, 2008). Both models are a trigram model with modified Kneser-Ney smoothing (Chen and Goodman, 1996). One model uses 1 million tokens from the LCC corpus and is used in the first pass decoding in combination with the acoustic model. The second one is used for LM rescoring and uses 10 million tokens from LCC. The extended Icelandic pronunciation dictionary used in the system was described in Section 3.

5. Experimental setup

We trained three ASR models using the training setup as described in the previous Section. The only difference between the models is due to the speech corpus training set: a) ALL-DATA: all training data, 88,285 utterances or approx. 109 hours of speech b) NO-PHONOTACT.: no utterances from the set of phonotactical sentences, and c) PHONOTACT.: a training set of the same size as NO-PHONOTACT. including all phonotactical utterances, but with corresponding reduction in utterances from the news category. In the original test set the OOV-rate is 5.7%, and 13.2% of the types are not contained in the lexicon of the ASR system. An additional test set only with known words was created, and each ASR model used to decode each of the both test sets.

6. Evaluation

Table 2 shows the results of the decoding experiments.

Training data	Full test set	Sub test set
ALL-DATA (109h)	15.72% WER	7.99% WER
NO-PHONOT.. (93h)	16.27% WER	8.60% WER
PHONOTACT. (95h)	16.17% WER	8.41% WER

Table 2: Test results of the Open ASR for Icelandic. Best word-error-rate (WER) is reached when using the largest available acoustic training set (ALL-DATA), approx. 109 hours of speech. The NO-PHONOT. training data set is a slightly smaller set were all phonotactical sentences have been deleted, and the PHONOTACT. set is similar in size to NO-PHONOT., but contains all these specially selected sentences.

The results of the system trained on the whole training set show 15.72% word-error rate (WER) on the full test set. Removing 14-16 hours from the training set increases the WER to 16.17% and 16.27%. Getting more training data would therefore most likely improve the results of the current system.

There is a large gap between the results for the full test set and the test set without unknown words. Recall that the OOV rate of the full test set is only 5.7%, compared to an absolute difference in WER of 7.67% - 7.76% between the full test set and the no OOV test set results. This leads to the assumption that an error caused by an unknown word causes more errors in the succeeding words, as has been shown to be the case for other languages (see e.g. (Salimbajevs and Strigins, 2015)).

One question to be answered by the experiment, was if the carefully selected sentences in the training set have an impact on decoding results. For the full test set and especially for the no OOV test set, the results of the PHONOTACT. set (containing the specially selected sentences) are clearly better than for the set without these sentences. The PHONOTACT. set, however, is slightly larger in duration, approx. 95 hours of speech compared to approx. 93 hours of speech in the NO-PHONOTACT. set. Further experiments are needed to explore this impact. Tests with still smaller training sets, where the phonotactical sentences comprise a larger part of the PHONOTACT. set and where both sets are still closer in duration would be needed. It would be a valuable insight to be able to quantify this impact better. If the impact is substantial, then it is worth the effort to create such a carefully crafted list if working with a low-resourced language. If, on the other hand, the impact is minimal, the work invested in the creation of the list can be skipped, plus that these sentences, at least in the case of the Icelandic sentences at hand, are difficult to read because of very rare and often long and complicated words and unusual word combinations. This diminishes the quality of the speech corpus, measured in match between spoken utterances and their corresponding prompts.

7. Conclusion and Future Work

We described the preparation of language resources for use in an LVSR system for Icelandic. The speech recognition system was implemented in Kaldi, using deep neural networks. The results of this baseline setup show the need for more acoustic training data to improve overall WER, and a larger pronunciation dictionary to be able to deal with open vocabulary decoding.

Experiments with different content of the acoustic training data indicate that careful selection of phonotactical data could be advantageous when developing a speech corpus of limited size. However, due to the extra effort needed to collect such data and the possible loss in recording quality due to reading mistakes, a clear positive impact of special phonotactical data should be evident. More experiments are needed to make a definite statement in this direction.

In addition to the extension of the speech corpus and dictionary, the ongoing work on text normalization and language modeling will be continued, as well as experiments with acoustic model architectures.

8. Acknowledgements

The project Open ASR for Icelandic was supported by the Icelandic Language Technology Fund (ILTF).

9. Bibliographical References

Adda-Decker, M. and Adda, G. (2000). Morphological decomposition for ASR in German. In *Proceedings of the Workshop on Phonetics and Phonology in Automatic Speech Recognition*, volume 5, pages 129–143.

Bisani, M. and Ney, H. (2008). Joint-sequence models for grapheme-to-phoneme conversion. *Speech Communication*, 50(5):434–451.

Bjarnadóttir, K. (2012). The Database of Modern Icelandic Inflection. In *Proceedings of the Workshop on Language Technology for Normalization of Less-Resourced Languages - SaLTMiL 8 - AfLaT2012*, pages 13–18.

Chen, S. F. and Goodman, J. (1996). An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics.

Goldhahn, D., Eckart, T., and Quasthoff, U. (2012). Building large monolingual dictionaries at the Leipzig Corpora Collection: From 100 to 200 languages. In *LREC*, pages 759–765.

Guðnason, J., Kjartansson, O., Jóhannsson, J., Carstensdóttir, E., Vilhjálmsson, H. H., Loftsson, H., Helgadóttir, S., Jóhannsdóttir, K. M., and Rögnvaldsson, E. (2012). ALMANNARÓMUR: An open Icelandic speech corpus. In *SLTU*, pages 80–83.

Guðnason, J., Pétursson, M., Kjaran, R., Klüpfel, S., and Nikulásdóttir, A. (2017). Building ASR corpora using Eyra. In *Proceedings of Interspeech*, pages 2173–2177.

Helgadóttir, I., Kjaran, R., Nikulásdóttir, A., and Guðnason, J. (2017). Building an ASR corpus using Althingi's parliamentary speeches. In *Proceedings of Interspeech*, pages 2163–2167.

Hsu, B.-J. and Glass, J. (2008). Iterative language model estimation: Efficient data structure & algorithms. In *Proceedings of Interspeech*, pages 841–844.

Hughes, T., Nakajima, K., Ha, L., Vasu, A., Moreno, P., and LeBeu, M. (2010). Building transcribed speech corpora quickly and cheaply for many languages. In *Interspeech*, pages 1914–1917.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Ordelman, R., van Hessen, A., and de Jong, F. (2003). Compound decomposition in Dutch large vocabulary speech recognition. In *Interspeech*, pages 225–228.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, B., Hannemann, M., Molticek, P., Quian, Y., Schwarz, P., Silovský, J., Stemmer, G., and Veselý, K. (2011). The Kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society.

Povey, D., Peddinti, V., Galvez, D., Gharhmani, P., Manohar, V., Na, X., Wang, Y., and Khudanpur, S. (2016). Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Proceedings of Interspeech*, pages 2751–2755.

Pétursson, M., Klüpfel, S., and Guðnason, J. (2016). Eyra - speech data acquisition system for many languages. In *Proceedings of STLU*.

Rögnvaldsson, E., (2003). *The Icelandic speech recognition project Hjal*, pages 239–242.

Sak, H., Senior, A., and Beaufays, F. (2014). Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*.

Salimbajevs, A. and Strigins, J. (2015). Error analysis and improving speech recognition for Latvian language. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pages 563–569.

Sproat, R., Black, A., and et al., S. C. (2001). Normalization of non-standard words. *Computer Speech & Language*, 15:287–333.

Steingrímsson, S., Guðnason, J., Helgadóttir, S., and Rögnvaldsson, E. (2017). Málrómur: A manually verified corpus of recorded Icelandic speech. In *Proceedings of the 21st Nordic Conference of Computational Linguistics*, pages 237–240.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K. (1989). Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):328–339.

Towards Neural Speaker Modeling in Multi-Party Conversation: The Task, Dataset, and Models

Zhao Meng,[1,2] **Lili Mou,**[1,3] **Zhi Jin**[1,*]
[1]Key Laboratory of High Confidence Software Technologies, MoE; Software Institute, Peking University
[2]Department of Computer Science, ETH Zurich
[3]David R. Cheriton School of Computer Science, University of Waterloo
zhmeng@student.ethz.ch, doublepower.mou@gmail.com, zhijin@sei.pku.edu.cn

Abstract

Neural network-based dialog systems are attracting increasing attention in both academia and industry. Recently, researchers have begun to realize the importance of speaker modeling in neural dialog systems, but there lacks established tasks and datasets. In this paper, we propose *speaker classification* as a surrogate task for general speaker modeling, and collect massive data to facilitate research in this direction. We further investigate temporal-based and content-based models of speakers, and propose several hybrids of them. Experiments show that speaker classification is feasible, and that hybrid models outperform each single component.

Keywords: Speaker Classification, Speaker Modeling, Multi-Party Conversation

1. Introduction

Human-computer conversation has long attracted attention in both academia and industry. Researchers have developed a variety of approaches, ranging from rule-based systems for task-oriented dialog (Ferguson et al., 1996; Graesser et al., 2005) to data-driven models for open-domain conversation (Ritter et al., 2011).

A simple setting in the research of dialog systems is context-free, i.e., only a single utterance is considered during reply generation (Shang et al., 2015). Other studies leverage context information by concatenating several utterances (Sordoni et al., 2015) or building hierarchical models (Yao et al., 2015; Serban et al., 2016). The above approaches do not distinguish different speakers, and thus speaker information would be lost during conversation modeling.

Speaker modeling is in fact important to dialog systems, and has been studied in traditional dialog research. However, existing methods are usually based on hand-crafted statistics and *ad hoc* to a certain application (Lin and Walker, 2011). Another research direction is speaker modeling in a multi-modal setting, e.g., acoustic and visual (Uthus and Aha, 2013), which is beyond the focus of this paper.

Recently, neural networks have become a prevailing technique in both task-oriented and open-domain dialog systems. After single-turn and multi-turn dialog studies, a few researchers have realized the role of speakers in neural conversational models. Li et al. (2016) show that, with speaker identity information, a sequence-to-sequence neural dialog system tends to generate more coherent replies. In their approach, a speaker is modeled by a learned vector (also known as an *embedding*). Such method, unfortunately, requires massive conversational data for a particular speaker to train his/her embedding, and thus does not generalize to rare or unseen speakers.

Ouchi and Tsuboi (2016) formalize a new task of addressee selection on online forums: by leveraging either the temporal or utterance information, they predict whom a post is talking to. While tempting for benchmarking speaker modeling, the task requires explicit speaker ID mentions, which occurs occasionally, and thus is restricted.

In this paper, we propose a *speaker classification* task that predicts the speaker of an utterance. It serves as a surrogate task for general speaker modeling, similar to *next utterance classification* (Lowe et al., 2015, NUC) being a surrogate task for dialog generation. The speaker classification task could also be useful in applications like *speech diarization*,[1] where text understanding can improve speaker segmentation, identification, etc. in speech processing (Li et al., 2009; Meng et al., 2017).

We further propose a neural model that combines temporal and content information with interpolating or gating mechanisms. The observation is that, what a speaker has said (called `content`) provides non-trivial background information of the speaker. Meanwhile, the relative order of a speaker (e.g., the i-th latest speaker) is a strong bias: nearer speakers are more likely to speak again; we call it `temporal` information. We investigate different strategies for combination, ranging from linear interpolation to complicated gating mechanisms inspired by Differentiable Neural Computers (Graves and others, 2016, DNC).

To evaluate our models, we constructed a massive corpus using transcripts of TV talk shows from the Cable News Network website. Experiments show that combining content and temporal information significantly outperforms either of them, and that simple interpolation is surprisingly more efficient and effective than gating mechanisms.

Datasets and code are available on our project website.[2]

2. Task Formulation and Data Collection

We formulate speaker classification as follows.

Assume that we have segmented a multi-party conversation into several parts by speakers; each segment com-

[*]Corresponding author.

[1]Speech diarization aims at answering "who spoke when" (Anguera et al., 2012).

[2]`https://sites.google.com/site/neuralspeaker/`

Data partition	# of samples
Train	174,487
Validation	21,071
Test	20,501

Table 1: Dataset statistics.

prises one or a few consecutive sentences $u_1, u_2, \cdots, u_N$, uttered by a particular speaker. A candidate set of speakers $\mathcal{S} = \{s_1, s_2, \cdots, s_k\}$ is also given. In our experiments, we assume $u_1, u_2, \cdots, u_N$'s speaker s_i is in $\mathcal{S}$. The task of speaker classification is to identify the speaker s_i of $u_1, \cdots, u_N$.

Following the spirit of distributed semantics (e.g., word embeddings), we represent the current utterance(s) as a real-valued vector $\boldsymbol{u}$ with recurrent neural networks. Speakers are also represented as vectors $\boldsymbol{s}_i, \cdots, \boldsymbol{s}_k$. The speaker classification is accomplished by a softmax-like function

$$\widetilde{p}_i = \exp\left\{\boldsymbol{s}_i^\top \boldsymbol{u}\right\} \tag{1}$$

$$p(s_i) = \frac{\widetilde{p}_i}{\sum_j \widetilde{p}_j} \tag{2}$$

Because the number of candidate speakers may vary, the "weights" of softmax are not a fixed-size matrix, but the distributed representations of candidate speakers, $\boldsymbol{s}_1, \cdots, \boldsymbol{s}_k$. In Section 3., we investigate several approaches of modeling $\boldsymbol{s}_i$ based on what a speaker says or the relative order of a speaker in the dialog; we also propose to combine them by interpolating or gating mechanisms.

To facilitate the speaker classification task, we crawled transcripts of more than 8,000 episodes of TV talk shows.[3] We assumed that the current speaker is within the nearest k speakers. ($k = 5$, but at the beginning, k may be less than 5.) Since too few utterances do not provide much information, we required each speaker having at least 3 previous utterances, but kept at most 5. Samples failing to meet the above requirements were filtered out during data preprocessing.

We split train/val/test sets according to TV show episodes instead of sentences; therefore no utterance overlaps between training and testing. Table 1 shows the statistics of our dataset partition.

3. Methodology

We use a hierarchical recurrent neural network (Serban et al., 2016) to model the current utterances $u_1, \cdots, u_N$ (Figure 1a). In other words, a recurrent neural network (RNN) captures the meaning of a sentence; another LSTM-RNN aggregates the sentence information into a fixed-size vector. For simplicity, we use RNN's last state as the current utterances' representation ($\boldsymbol{u}$ in Equation 2).

In the rest of this section, we investigate content-based and temporal-based prediction in Subsections 3.1. and 3.2.; the spirit is similar to "dynamic" and "static" models, respectively, in Ouchi and Tsuboi (2016). We combine content-based and temporal-based prediction using gating mechanisms in Subsection 3.3..

3.1. Prediction with Content Information

In this method, we model a speaker by what he or she has said, i.e., content.

Figure 1b illustrates the content-based model: a hierarchical RNN (which is the same as Figure 1a) yields a vector $\boldsymbol{s}_i$ for each speaker, based on his or her nearest several utterances. The speaker vector $\boldsymbol{s}_i$ is multiplied by current utterances' vector $\boldsymbol{u}$ for softmax-like prediction (Equation 2). We pick the candidate speaker that has the highest probability.

It is natural to model a speaker by his/her utterances, which provide illuminating information of the speaker's background, stance, etc. As will be shown in Section 4., content-based prediction achieves significantly better performance than random guess. This also verifies that *speaker classification* is feasible, being a meaningful surrogate task for speaker modeling.

3.2. Prediction with Temporal Information

In temporal-based approach, we sort all speakers in a descending order according to the last time he or she speaks, and assign a vector (embedding) for each index in the list, following the "static model" in Ouchi and Tsuboi (2016). Each speaker vector is randomly initialized and optimized as parameters during training. The predicted probability of a speaker is also computed by Equation 2.

The temporal vector is also known as a *position embedding* in other NLP literature (Nguyen and Grishman, 2015). Our experiments show that temporal information provides strong bias: nearer speakers tend to speak more; hence, it is also useful for speaker modeling.

3.3. Combining Content and Temporal Information

As both content and temporal provide important evidence for speaker classification, we propose to combine them by interpolating or gating mechanisms (illustrated in Figure 1d). In particular, we have

$$\boldsymbol{p}^{(\text{hybrid})} = (1 - g) \cdot \boldsymbol{p}^{(\text{temporal})} + g \cdot \boldsymbol{p}^{(\text{content})} \tag{3}$$

Here, g is known as a *gate*, balancing these two aspects. We investigate three strategies to compute the gate.

1. **Interpolating after training.** The simplest approach, perhaps, is to train two predictors separately, and interpolate after training by validating the hyperparameter g.

2. **Interpolating while training.** We could also train the hybrid model as a whole with cross-entropy loss directly applied to Equation 3.

3. **Self-adaptive gating.** Inspired by hybrid content- and location-based addressing in Differentiable Neural Computers (Graves and others, 2016, DNCs), we design a learnable gate in hopes of dynamically balancing temporal and content information. Different

[3] https://transcripts.cnn.com

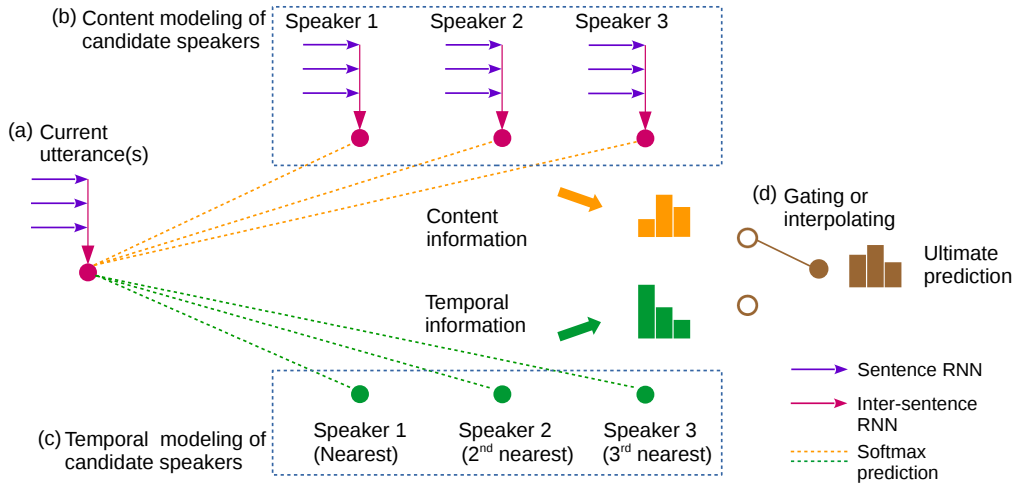


Figure 1: Hybrid content- and temporal-based speaker classification with a gating mechanism.

Model	Macro F_1	Weighted F_1	Micro F_1	Acc.	MRR.
Random guess	19.93	34.19	27.53	27.53	N/A
Majority guess	21.26	62.96	74.01	74.01	N/A
Hybrid random/majority guess	25.26	61.99	69.29	69.29	N/A
Temporal information	26.07	63.60	73.99	73.99	84.85
Content information	42.61	65.04	61.82	58.58	74.86
+ static attention	42.50	65.28	61.79	58.99	74.89
+ sentence-by-sentence attention	42.56	65.96	62.86	59.81	75.58
Hybrid Interpolating after training	**44.25**	**71.35**	**76.10**	**75.84**	**85.73**
Hybrid Interpolating while training	41.30	70.10	75.57	75.31	85.20
Hybrid Self-adaptive gating	39.45	69.55	74.11	74.09	84.85

Table 2: Model performance (in percentage).

from DNCs, however, the gate here is not based on input (i.e., $\boldsymbol{u}$ in our scenario), but the result of content prediction $\boldsymbol{p}^{(\text{content})}$. Formally

$$g = \text{sigmoid}\left(w \cdot \text{std}[\,p^{(\text{content})}\,] + b\right) \qquad (4)$$

where we compute the standard deviation (std) of p. w and b are parameters to scale $\text{std}[\,p^{(\text{content})}\,]$ to a sensitive region of the sigmoid function.

4. Experimental Results

Setup. All our neural layers including word embeddings were set to 100-dimensional. We tried larger dimensions, resulting in slight but insignificant improvement. We did not use pretrained word embeddings but instead randomly initialized them because our dataset is large. We used the Adam optimizer (Kingma and Ba, 2015) mostly with default hyperparameters. We set the batch size to 10 due to GPU memory constraints. Dropout rate and early stop were also applied by validation. Notice that validation was accomplished by each metric itself because different metrics emphasize different aspects of model performance.

Performance. Table 2 compares the performance of different models. Majority-class guess results in high accuracy, showing that the dataset is screwed. Hence, we choose macro F_1 as our major metric, which addresses minority classes more than other metrics. We nevertheless present other metrics including accuracy, mean reciprocal ranking (MRR), and micro/weighted F_1 as additional evidence.

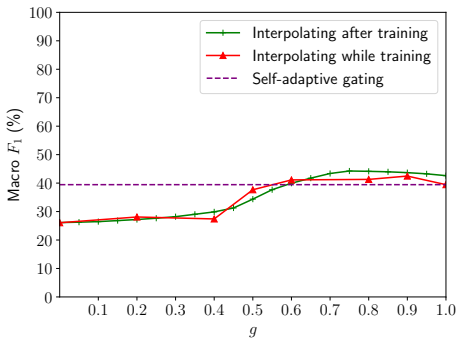


Figure 2: Performance vs. the hyperparameter g (solid lines). $g = 0$: temporal only; $g = 1$: content only. The self-adaptive gating mechanism is also plotted for comparison (which is not associated with a particular value of g).

As shown, the content-based model achieves higher performance in macro F_1 than majority guess, showing the effectiveness of content information. Following Rocktäschel et al. (2016), we adopt a static or sentence-by-sentence attention mechanism. The LSTM-RNN attends to speaker s_i to obtain speaker vector $\boldsymbol{s}_i$ while it is encoding current utterances. However, such attention mechanisms bring little improvements (if any). Hence, we do not use attention in our hybrid models for simplicity.

All hybrid models achieve higher performance compared with either content- or temporal-based prediction in terms

of most measures, which implies content and temporal information sources capture different aspects of speakers.

Among different strategies of hybrid models, the simple approach "interpolating after training" surprisingly outperforms the other two. A plausible explanation is that training a hybrid model as a whole leads to optimization difficulty in our scenario; that simply interpolating well-trained models is efficient yet effective. However, the hyperparameter g is sensitive and only yields high performance in the range $(0.6, 0.9)$. Thus, the learnable gating mechanism could also be useful in some scenarios, as it is self-adaptive.

5. Conclusion and Future Work

In this paper, we addressed the problem of neural speaker modeling in multi-party conversation. We proposed *speaker classification* as a surrogate task and collected massive TV talk shows as our corpus. We investigated content-based and temporal-based models, as well as their hybrids. Experimental results show that speaker classification is feasible, being a meaningful task for speaker modeling; that interpolation between content- and temporal-based prediction yields the highest performance.

In the future, we would like to design more dedicated gating mechanisms to improve the performance; we would also like to explore other aspects of speaker modeling, e.g., incorporating dialog context before current utterances. The collected dataset is also potentially useful in other applications.

6. Bibliographical References

Anguera, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., and Vinyals, O. (2012). Speaker diarization: A review of recent research. *IEEE Transactions on Audio Speech and Language Processing*, 20(2):356–370.

Ferguson, G., Allen, J. F., and Miller, B. (1996). Trains-95: Towards a mixed-initiative planning assistant. In *Proceedings of Artificial Intelligence Planning Systems Conference*, pages 70–77.

Graesser, A. C., Chipman, P., Haynes, B. C., and Olney, A. (2005). AutoTutor: an intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education*, 48(4):612–618.

Graves, A. et al. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476.

Kingma, D. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the International Conference for Learning Representations*.

Li, R., Schultz, T., and Jin, Q. (2009). Improving speaker segmentation via speaker identification and text segmentation. In *INTERSPEECH*, pages 904–907.

Li, J., Galley, M., Brockett, C., Spithourakis, G., Gao, J., and Dolan, B. (2016). A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 994–1003.

Lin, G. I. and Walker, M. A. (2011). All the world's a stage: Learning character models from film. In *Proceedings of the Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, pages 46–52.

Lowe, R., Pow, N., Serban, I., and Pineau, J. (2015). The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 285–294.

Meng, Z., Mou, L., and Jin, Z. (2017). Hierarchical RNN with static sentence-level attention for text-based speaker change detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2203–2206. ACM.

Nguyen, T. H. and Grishman, R. (2015). Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 39–48.

Ouchi, H. and Tsuboi, Y. (2016). Addressee and response selection for multi-party conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2133–2143.

Ritter, A., Cherry, C., and Dolan, W. B. (2011). Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 583–593.

Rocktäschel, T., Grefenstette, E., Hermann, K. M., Kočiskỳ, T., and Blunsom, P. (2016). Reasoning about entailment with neural attention. In *Proceedings of the International Conference on Learning Representations*.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. (2016). Building end- to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 3776–3783.

Shang, L., Lu, Z., and Li, H. (2015). Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1577–1586, July.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205.

Uthus, D. C. and Aha, D. W. (2013). Multiparticipant chat analysis: A survey. *Artificial Intelligence*, 199:106–121.

Yao, K., Zweig, G., and Peng, B. (2015). Attention with intention for a neural network conversation model. *arXiv preprint arXiv:1510.08565*.

Discriminating between Similar Languages on Imbalanced Conversational Texts

Junqing He[†‡], Xian Huang[§], Xuemin Zhao[†‡], Yan Zhang[†‡], Yonghong Yan[†‡♣]
[†]Institute of Acoustics, Chinese Academy of Science, Bejing, China
[‡] University of Chinese Acadamy of Sciences, Bejing, China
[§]PLA Information Engineering University Luoyang Division, Henan, China
[♣]Xinjiang Technical Institute of Physics and Chemistry, Chinese Academy of Sciences, Urumchi, China
hejunqing@hccl.ioa.ac.cn, a77huang852yi@126.com, {zhaoxuemin,zhangyan,yanyonghong}@hccl.ioa.ac.cn

Abstract

Discriminating between similar languages (DSL) on conversational texts is a challenging task. This paper aims at discriminating between limited-resource languages on short conversational texts, like Uyghur and Kazakh. Considering that Uyghur and Kazakh data are severely imbalanced, we leverage an effective compensation strategy to build a balanced Uyghur and Kazakh corpus. Then we construct a maximum entropy classifier based on morphological features to discriminate between the two languages and investigate the contribution of each feature. Empirical results suggest that our system achieves an accuracy of 95.7% on our Uyghur and Kazakh dataset, which is higher than that of the CNN classifier. We also apply our system to the out-of-domain subtasks of VarDial'2016 DSL shared tasks to test the system's performance on short conversational texts of other similar languages. Though with much less preprocessing, our system outperforms the champions on both test sets B1 and B2.

Keywords: discriminating between similar languages, imbalanced data, morphological features, maximum entropy

1. Introduction

Automatic language identification (LID) aims to identify the language a document is written in, which is an important branch in Natural Language Processing (NLP) (Zampieri et al., 2015a). The past two decades had witnessed fast development in LID and state-of-the-art systems have achieved high accuracy (Simões et al., 2014) and wide coverage (Brown, 2014) on standard texts. However, identifying languages from very little data, from multi-languages input or discriminating between extremely similar languages are bottlenecks of this field (Ljubešic and Kranjcic, 2014; Zampieri et al., 2015b). What's more, identifying similar languages with limited resource is unsolved.

Uyghur and Kazakh, widely used in Middle East and North West of China, are similar languages. They both belong to the Turkic group of Altaic family and are agglutinative languages. According to Wang et al. (2013), the similarity between Uyghur and Kazakh at sentence and word level are over 80% and 90% respectively. They have many characteristics in common: (1) They are both written in Arabic alphabets in the right-to-left order. (2) Theoretically there are 32 letters in Uyghur and 33 in Kazakh. The two languages share 26 letters and encoding areas with another 2 letters look exactly the same. (3) There is a large overlap of vocabulary and syntax between the two languages. It is very difficult to identify them by looking up the words in dictionaries. (4) In both languages, a great amount of prefixes and suffixes are attached to a word, which makes word stemming and recognition difficult.

Here we define "short conversational texts" as short texts people used to communicate with each other through mobile devices, communicational software and social-media platforms. They can be (1) short messages people send to each other through cell phones; (2) chatting records of communicational software such as Wechat and MSN; (3) posts and comments on social-media platforms such as Twitter, Facebook and Microblog. These texts are obstacles for NLP tasks for the following reasons: (1) Each text is pretty short. Lengths of most sentences range from 3 to 9 words. (2) There exist enormous spelling and grammatical mistakes in the texts, which make it time and energy consuming in word stemming and error correction. (3) Abbreviations and colloquial expressions are widely used. (4) It takes much time and energy to collect short conversational texts, resulting in the imbalance and inadequacy of the corpus. (5) Without unified input methods, people use various characters other than standard ones. In fact, more than 100 letters of different encoded bytes are found in our corpus. This strengthened the difficulty of discriminating between Uyghur and Kazakh short conversational texts.

2. Related Work

Since more and more researchers are concerned with discriminating between similar languages (DSL), a series of shared tasks were organized by the workshop series for Similar Languages, Varieties and Dialects (VarDial), which was collocated with either COLING, RANLP or EACL. According to Malmasi et al. (2016), high-order character n-grams were the most successful feature, and the best classification models included SVM, logistic regression, and language models, while deep learning approaches did not perform very well.

To deal with short and sparse texts, solutions (Phan et al., 2008; Rehurek and Kolkus, 2009; Tromp and Pechenizkiy, 2011; Dai et al., 2013) were proposed to enrich short text representation by bringing in additional semantics. The additional semantics could be from data collection itself or be derived from a much larger external knowledge base. Dealing with tweets, Zubiaga et al. (2014) summarized the TweetLID shared task and workshop held at SEPLN 2014 and pointed out several shortcomings in current researches.

When it comes to discriminating between Uyghur and Kazakh, Hasimu et al. (2015) employed unique characters to identify the Uyghur , Kazakh and Kyrgyz languages. They carried out experiments on the written texts longer than 70 words and achieved a 96.67% accuracy. But in the web corpus of less than 10 words, the precision of Kazakh fell dramatically to only 65.31%.

In this paper, we made two contributions: (1) We constructed a corpus of conversational texts in Uyghur and Kazakh for similar languages identification and proposed a method for corpus augmentation. (2) We designed a system that can effectively discriminate between similar languages on conversational texts.

3. Data Construction

3.1. Data Collection

With the popularization of social network and chatting applications on mobile phones, people are more likely to communicate with each other via short instant messages. Thus natural languge processing on short conversational messages is of great significance.

We collected 48680 texts from the chatting messages sent by mobile phones and used as our training set after anonymization. Likewise, 973 colloquial messages that were sent in a day were collected as our test set. Then all the texts were tagged by linguistic experts. In our training set, we found that 48432 samples were written in Uyghur while 148 samples were Kazakh. As for test set, 687 and 286 texts were annotated as Uyghur and Kazakh separately. The scales of Uyghur and Kazakh texts in the training set were severely imbalanced, which exceeded the proportion of 327:1.

3.2. Data Augmentation

Since a highly imbalanced training corpus may hinder the effectiveness of discrimination between the two similar languages, we decided to balance the corpus by supplementing Kazakh texts.

We did not collect more Kazakh short messages in the same way because of inefficiency since the linguistic experts have to skim more than 300 Uyghur samples to get a Kazakh sample. To obtain data that are similar to short messages which are conversational, informal and short, we decided to crawl data from Kazakh forums instead of the Kazakh news web pages and Twitter. The reasons are as follows: (1) News are formal written texts, which have little overlap of words and characters with short communicational messages. (2) Although tweets are short and informal, Twitter is rarely used in China. (3) Posts on Kazakh forums are informal and conversational, which resemble the nature of short messages. What's more, the contents are almost entirely written in Kazakh.

We crawled 70909 web pages from a Kazakh forum[1]. However, some texts in these web pages were longer than the chatting messages. To make the crawled texts more similar to the short messages, we picked out 339,609 samples of no more than 14 words. Then we randomly chose 48000 texts from the filtered samples to match the number of Uyghur texts in the original training set. In this way, we constructed a balanced corpus of Uyghur and Kazakh short conversational texts used for discriminating between the two similar languages. The final corpus contains a training set with 48432 Uyghur and 48148 Kazakh samples; and and a test set with 687 Uyghur and 286 Kazakh samples. More details of the corpus are listed in Table 1 and Table 2.

[1]From `http://bbs.senkazakh.com`

	Training Set		Test Set	
Language	Uyghur	Kazakh	Uyghur	Kazakh
Vocab. Size	37237	43876	2528	1161
Instances	48432	48148	687	286
Avg. Length	5.98	4.39	5.94	5.77

Table 1: Statistics of the training and test set of the Uyghur and Kazakh data. Avg. Length represents the average number of tokens in each instance.

Num. of Tokens	Test Set	Training Set		
		Uyghur	Kazakh	Sup. Kazakh
1	25	135	21	4032
2	93	1433	9	7907
3	135	4847	16	8558
4	116	7668	13	7755
5	88	8170	17	5688
6	74	7352	10	4359
7	81	6227	12	3637
8	158	5409	17	2751
9	118	3898	18	1856
10	63	2299	10	1046
11	19	837	4	341
12	3	198	1	51
13	0	42	0	9
14	0	7	0	1
Sum	973	48432	148	48000

Table 2: Distributions of instances on different number of tokens in the Uyghur and Kazakh data for DSL. Sup.Kazakh is referred to the supplemental Kazakh data .

4. Our System

4.1. Feature Extraction

Since all texts in the corpus are extremely short, we assume the lexical n-gram features cannot play an important role in DSL in short conversational texts. Based on linguistic, in particular morphological analysis of the two languages, we mainly used the following features to discriminate between the two languages:

- **Unique characters**. Once a unique character is found in a text, we can determine that the text is written in the language the unique character belongs to.
- **Character n-grams**. The sequence and combination of characters is different among various languages, even though the languages share a lot of characters. The Uyghur Latin word "men" corresponds to "man" in Kazakh Latin for the same meaning. (For the convenience of typing and visualization, here we use Latin

charcters to embody the Uyghur and Kazakh instead of the Arabic letters.)

- **Prefixes and suffixes**. As agglutinative languages, both Uyghur and Kazakh have numerous affixes. On many occasions, affixes of the two languages are different. For example, to express the same meaning, suffix "lar" is used in Uyghur, while "dar" is used in Kazakh. Likewise, "o" can be the first letter in Kazakh but cannot be found at this position in Uyghur. One thing we should note is that misspelling problems make this feature hard to extract, thus we use the the first and last n characters of the words as a substitute of prefixes and suffixes. Here n ranges from 1 to 3.

- **Word unigrams**. The frequency of a word represents how likely it belongs to a language. If a text contains a high-frequency word of a language, it is more likely to belong to the corresponding language.

- **Bin on text length**. We can divide the texts into different bins according to the lengths of the texts. Models trained in certain bin length will be more accurate.

4.2. Classifiers

Nowadays there are many state-of-the-art classifiers that achieve steady and desirable performance, no matter whether they are based on machine learning or neural networks.

The maximum entropy (MaxEnt) classifier is one of the best models among the machine learning algorithms. The MaxEnt classifier computes the conditional likelihood and relativity of the features mutually for each category in the training step. Based on the statistics, for each sample, the classifier adjusts the weights of corresponding features to maximize the max entropy of the sentence under the constraints of all the conditional likelihood above. When predicting, scores of samples of each category is computed and the class of the highest score is chosen as its label. Therefore, feature dependence is taken into account in MaxEnt. In this paper, we applied a MaxEnt classifier in our system using the Stanford classifier toolkit[2].

With the convolutional neural networks (CNN) successfully applied to image recognition (Krizhevsky et al., 2012) and text classification (Kim, 2014), CNN became one of the most popular deep learning classifier algorithms. We also built a CNN classifier based on character embeddings considering the features are mainly of the character level and then compared the performances of the two classifiers.

4.3. Evaluation

Since we take the DSL task as an issue of classification, we use the evaluation metrics of classification systems. Precision (P), recall (R) and accuracy (Acc) are used to evaluate the performance of our system.

5. Experiments and Discussion

In this section, we conducted four experiments to examine the effect of the supplemented Kazakh samples, the contribution of each morphological feature we use, the performance of the CNN and MaxEnt classifiers, and the performance of our system in dealing with the out-of-domain test sets B1, B2 of the VarDial'2016 DSL shared tasks. The B1 and B2 data sets are considered to be out-of-domain because the training data are collected from news while the test sets comprise of tweets.

5.1. Experiment on the Supplemented Data

In this experiment, we use the MaxEnt classifier based on all the features except bin. Table 3 shows the results of our system trained on the original imbalanced and the final supplemented training sets.

Training Sets	Uyghur		Kazakh		
	P	R	P	R	Acc
Original	89.0	99.3	97.6	70.6	90.0
Final	**98.5**	95.1	89.0	**96.5**	95.5

Table 3: The influence of data augmentation. All the results are in percentage.

Results of the experiment reveal that the model trained on the imbalanced training data can recall almost all the Uyghur texts but can only recall 70.6% of true Kazakh samples. It is indicated that the system regards most of the samples as Uyghur with only 89.0 % of precision since the high probability of its appearance in the original training set. After the data augmentation, the recall of the Kazakh samples improves by 25.9% and is close to that of the Uyghur samples. It is suggested that the augmentation strategy is useful and it is important to keep the scales of the training data for each language even.

5.2. Experiment on the Features' Contribution

To investigate the contribution of each feature, we evaluate the performance of our system using all the features, and without each one of them each time separately, *e.g.* using all the features without unique characters or character n-grams. In this way, we can see how importance each feature is by observing the decrease of performance, compared to that of using all the features. The MaxEnt classifier is employed and trained on the final Uyghur and Kazakh training set. Results are shown in Table 4.

Features	Uyghur		Kazakh		
	P	R	P	R	Acc
All	98.5	95.2	89.3	96.5	95.6
-unique chars	98.3	94.5	87.9	96.2	95.0
-char n-gram	97.8	90.5	80.7	95.1	91.9
-pre/suf-fixs	98.5	94.6	88.2	96.5	95.2
-word unigrams	**98.5**	**95.3**	**89.6**	**96.5**	**95.7**
-bin	98.5	95.0	89.0	96.5	95.5

Table 4: Performance of our system without each kind of features. All the results are in percentage. The bold results are the best performance in the same metric.

As we can see, with each feature removed, the performance of our system decreases to different extent except

[2]Available at `https://nlp.stanford.edu/software/`

for the word unigrams. It proves that all the features except word unigrams are useful in this task. We also observe the sharpest decline in accuracy when the character n-grams are removed, which implies it contributes the most among all the features. On the contrast, the accuracy increases slightly without the word unigrams, which reveals that word level features are helpless and even undermine the performance of language identification on the data set. Therefore, we stop using this feature in the following experiments.

5.3. Experiment on the Classifiers

In this experiment we respectively use the CNN classifier and MaxEnt classifier trained on the final Uyghur and Kazakh corpus to compare their performance. For the MaxEnt classifier, all the features except word unigrams are used. For the CNN classifier, the samples are represented at the character level with each character mapped into an embedding of 50 dimensions. Convolutional kernel widths are set to [1,2,3,4] to resemble the character n-grams of size 1 to 4 used in MaxEnt classifier. Numbers of kernels are set to be [50,200,300,500] separately since there is no improvement when using more kernels. A dropout layer with a 0.5 dropout rate is applied. The character embeddings are randomly initialized between (-0.05,0.05) under the uniform distribution. The performances of the two classifiers using the best parameters are listed in Table 5.

Classifiers	Uyghur		Kazakh		
	P	R	P	R	Acc
CNN	30.7	10.1	71.4	93.6	69.1
MaxEnt	98.5	95.3	89.6	96.5	95.7

Table 5: Performances of the MaxEnt and CNN classifier. All results are in percentage.

As Table 5 indicates, the MaxEnt classifier turned out to be much more competitive and effective than the CNN classifier. This finding echoes Zampieri et al. (2017) and Malmasi et al. (2016)'s findings that CNN fails to perform well in DSL tasks. Having an insight into the CNN's performance on Uyghur and Kazakh samples, we can see much better results on Kazakh compared to that on Uyghur samples. According to Table 1, we assume that the CNN classifier needs more training data to learn the character n-gram patterns in Uyghur than in Kazakh samples since they have longer sentences. This lack of Uyghur training data may lead to the failure in identifying Uyghur samples.

5.4. Experiment on the VarDial'2016 DSL shared task

Since the MaxEnt classifier using the morphological features achieved a high accuracy in discriminating between Ughur and Kazakh, we intended to test the performance of our system in discriminating other similar languages on short conversational texts.

5.4.1. Data Description

We chose the two social media data (B1 and B2) of Subtask 1 in VarDial'2016 DSL shared task[3] as the test materials. The training set consists of 18000 instances of journalistic data per language for training and 2000 instances for development. Each of the test sets includes 100 Twitter users' tweets per language. A varying number of tweets from a user are concatenated as a test sample (98.88 and 50.47 tweets per user for B1 and B2 in average separately). The two test sets cover two groups of closely-related languages : South-Slavic (Bosnian, Croatian, Serbian) and Portuguese (Brazilian and European). For each sample in the test set which contains five language/variants in a messed order, we have to find out which language it belongs to.

The reasons why we chose the two test sets are as following: (1) The test sets consists of Tweets, which are short, informal and conversational. (2) The test sets are out-of-domain data, which can test the classifier's robustness in handling out-of-domain data. (3) We can test whether the features are effective in discriminating between other similar languages than the Uyghur and Kazakh.

In the preprocessing process, no measures was taken to deal with the training set. As to the test sets we just removed the links, at-mentions and hash tags in them. Details of the training set and processed test sets B1 and B2 of subtask 1 are shown in Table 6.

Data Sets	Language Varieties	Instances	Vocab. Size	Avg. Length
Train	Bosnian	18000	77851	36.51
	Croatian	18000	82670	42.70
	Serbian	18000	74726	39.64
	BP	18000	44415	48.44
	EP	18000	39056	44.43
Test B1	Bosnian	100	30418	1270.61
	Croatian	100	24754	966.33
	Serbian	100	31278	1219.20
	BP	100	16457	960.94
	EP	100	14878	843.29
Test B2	Bosnian	100	25225	997.31
	Croatian	100	17811	613.59
	Serbian	100	23103	791.23
	BP	100	3922	121.22
	EP	100	2803	78.40

Table 6: Statistical analysis of Subtask 1 in VarDial'2016 DSL shared tasks. BP is short for Brazilian Portuguese and EP is short for European Portuguese.

5.4.2. Evaluation

In the DSL shared task, average accuracy (Acc) and macro-averaged F1-score (F1) were used as the official scores. Therefore we use the same metrics in this experiment. Since the DSL datesets of the subtask are balanced with the same number of examples for each language variety, we

[3]The dataset is version 3.0 of DSLCC, which is available at `http://ttg.uni-saarland.de/resources/DSLCC`

mainly use the average accuracy for comparison in the following subsection.

5.4.3. Results and Discussion

We applied the MaxEnt classifier and with the character n-gram feature (n ranges from 1 to 7) to compare with other participant systems. Results of our system as well as the top participant systems in B1 and B2 in the VarDial'2016 DSL shared tasks are listed in Table 7 and 8 respectively.

Team	Acc	F1	Approach
Our system	**0.930**	**0.930**	MaxEnt with char n-grams (n=1-7)
GW_LT3	0.920	0.919	Logistic Reg. with char/word n-grams
nrc	0.914	0.913	Two-stage SVM with char 6-grams
UniBucNLP	0.898	0.897	Logistic Reg. With word 1,2-grams
UPV_UA	0.888	0.886	String kernels and discriminant analysis
tubasfs	0.862	0.860	SVM with char n-grams (n=1-7)

Table 7: Results of top systems and our system on B1 in subtask 1 of Vardial'2016 DSL task. The bold results are the best performance in the same metric.

Team	Acc	F1	Approach
Our system	**0.890**	**0.890**	MaxEnt with char n-grams (n=1-7)
GW_LT3	0.878	0.877	Logistic Reg. with char/word n-grams
nrc	0.878	0.913	Two-stage SVM with char 6-grams
UPV_UA	0.858	0.857	String kernels and discriminant analysis
UniBucNLP	0.838	0.897	Logistic Reg. With word 1,2-grams
tubasfs	0.822	0.818	SVM with char n-grams (n=1-7)

Table 8: Results of top systems and our system on B2 in subtask 1 of Vardial'2016 DSL task. The bold results are the best performance in the same metric.

As is shown in Table 7 and 8, GW_LT3 ranked first in the subtask of discriminating between similar languages on the tweets dataset. It used character n-gram (n=2-6) and word n-gram (n=1-3) with term-frequency weighting, and took many preprocessing measures. Our system outperforms it by 1.0% in B1 and 1.2% in B2 in accuracy. It is implied that, besides Uyghur and Kazakh, our system is also highly efficient in DSL tasks in other similar languages on short conversational texts. Compared with tubasfs, which also used character n-grams as a feature (n=1-7), the accuracies of our system in B1 and B2 are both 6.8% higher. This indicates that MaxEnt is better than SVM in this task. In addition, while our system achieved the accuracy of 95.7% on the Uyghur and Kazakh dataset, we just set n to be 1 to 4 in the character n-gram feature. When dealing with B1 and B2 test sets, we set n to be 1 to 7, and the accuracies we got were 93.0% and 89.0% respectively, which are lower than that we got in dealing with Uyghur and Kazakh. The reason for the unsatisfying result is that the training set of DSL 2016 subtask1 are journalistic news, which are different from short conversational texts to some extent. That can also show that when discriminating between similar languages on short conversational texts, contents in related forums is a better resource than news as the training data.

6. Conclusion

In this paper, we have constructed a corpus of short conversational Uyghur and Kazakh texts used for DSL. To solve the severe imbalance problem of the two languages with limited resource, we proposed a data augmentation method. That was to crawl a Kazakh forum and choose the materials which were short, informal as supplemental data. It is suggested that our augmentation strategy is effective and texts from forums are more suitable than news texts for the DSL task.

Then we designed a MaxEnt classifier with morphological features to discriminate between Uyghur and Kazakh conversational texts. Our empirical study shows that the character level features we exploited are helpful while employing the word unigrams led to worse performance. Experimental results also indicate that our system can not only discriminate between Uyghur and Kazakh on short conversational texts at a high accuracy of 95.7%, but also outperforms the state-of-the-art systems in DSL on Tweets with out-of-domain training data in the VarDial'2016 DSL task. It is also implied that CNN is not a competitive model for this task and the MaxEnt performs better than the SVM classifier using the same features.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (Nos. 11461141004, 61271426, U1536117, 11504406, 11590770-4, 11590771), the Strategic Priority Research Program of the Chinese Academy of Sciences (Nos. XDA06030100, XDA06030500, XDA06040603), National 863 Program (No. 2015AA016306), National 973 Program (No. 2013CB329302) and the Key Science and Technology Project of the Xinjiang Uygur Autonomous Region (No. 201230118-3).

7. Bibliographical References

Brown, R. D. (2014). Non-linear mapping for improved identification of 1300+ languages. In *EMNLP*, pages 627–632.

Dai, Z., Sun, A., and Liu, X.-Y. (2013). Crest: Cluster-based representation enrichment for short text classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 256–267. Springer.

Hasimu, M., Silamu, W., Mushajiang, W., and Youliwasi, N. (2015). Unique character based statistical language identification for uyghur,kazak and kyrgyz. *Journal of Chinese Information Processing*, 29(2):111–117.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *International Conference on Neural Information Processing Systems*, pages 1097–1105.

Ljubešic, N. and Kranjcic, D. (2014). Discriminating between very similar languages among twitter users. In *Proceedings of the Ninth Language Technologies Conference*, pages 90–94.

Malmasi, S., Zampieri, M., Ljubešić, N., Nakov, P., Ali, A., and Tiedemann, J. (2016). Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task. In *Proceedings of the third VarDial Workshop*, Osaka, Japan.

Phan, X.-H., Nguyen, L.-M., and Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, pages 91–100. ACM.

Rehurek, R. and Kolkus, M. (2009). Language identification on the web: Extending the dictionary method. In *CICLing*, pages 357–368. Springer.

Simões, A., Almeida, J. J., and Byers, S. D. (2014). Language identification: a neural network approach. In *OASIcs-OpenAccess Series in Informatics*, volume 38. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Tromp, E. and Pechenizkiy, M. (2011). Graph-based n-gram language identification on short texts. In *Proc. 20th Machine Learning conference of Belgium and The Netherlands*, pages 27–34.

Wang, L., Yidemucao, D., and Silamu, W. S. (2013). An investigation research on the similarity of uyghur kazakh kyrgyz and mongolian languages. *Journal of Chinese Information Processing*, 27(6):180–186.

Zampieri, M., Gebre, B. G., Costa, H., and van Genabith, J. (2015a). Comparing approaches to the identification of similar languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 66–72. Association for Computational Linguistics.

Zampieri, M., Tan, L., Ljubešic, N., Tiedemann, J., and Nakov, P. (2015b). Overview of the dsl shared task 2015. In *Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*.

Zampieri, M., Malmasi, S., Ljubešić, N., Nakov, P., Ali, A., Tiedemann, J., Scherrer, Y., and Aepli, N. (2017). Findings of the vardial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 1–15, Valencia, Spain, April. Association for Computational Linguistics.

Zubiaga, A., Vicente, I. S., Gamallo, P., Campos, J. R. P., Loinaz, I. A., Aranberri, N., Ezeiza, A., and Fresno-Fernández, V. (2014). Overview of tweetlid: Tweet language identification at SEPLN 2014. In *Proceedings of the Tweet Language Identification Workshop co-located with 30th Conference of the Spanish Society for Natural Language Processing, TweetLID@SEPLN 2014, Girona, Spain, September 16th, 2014.*, pages 1–11.

Data-Driven Pronunciation Modeling of Swiss German Dialectal Speech for Automatic Speech Recognition

Michael Stadtschnitzer, Christoph Schmidt
Fraunhofer IAIS, Schloss Birlinghoven, 53757 Sankt Augustin
{michael.stadtschnitzer, christoph.andreas.schmidt}@iais.fraunhofer.de

Abstract

Automatic speech recognition is a requested technique in many fields like automatic subtitling, dialogue systems and information retrieval systems. The training of an automatic speech recognition system is usually straight forward given a large annotated speech corpus for acoustic modeling, a phonetic lexicon, and a text corpus for the training of a language model. However, in some use cases these resources are not available. In this work, we discuss the training of a Swiss German speech recognition system. The only resources that are available is a small size audio corpus, containing the utterances of highly dialectical Swiss German speakers, annotated with a standard German transcription. The desired output of the speech recognizer is again standard German, since there is no other official or standardized way to write Swiss German. We explore strategies to cope with the mismatch between the dialectal pronunciation and the standard German annotation. A Swiss German speech recognizer is trained by adapting a standard German model, based on a Swiss German grapheme-to-phoneme conversion model, which was learned in a data-driven manner. Also, Swiss German speech recognition systems are created, with the pronunciation based on graphemes, standard German pronunciation and with a data-driven Swiss German pronunciation model. The results of the experiments are promising for this challenging task.

Keywords: Robust Speech Recognition, Swiss German, Dialectal Speech, Data-Driven Pronunciation Modeling

1. Introduction

Switzerland has four national languages: German/Swiss German (63%), French (22.7%), Italian (8.1%), Romansh (0.5%); the numbers in brackets are the percentages of the population speaking them[1]. As can be derived from Figure 1, French is spoken in the west, Italian is spoken primarily in Ticino, Val Bregaglia and Val Pschiavo and Romansh speakers are distributed over Graubünden. Swiss German, which is primarily spoken in the center and east of Switzerland, is highly dialectical. Typically, speakers speak a dialect representative of the region. To be understood by visitors, Swiss German speakers switch to standard German (Garner et al., 2014). Swiss German and its dialectal variants do not have a standard written form, instead the standard written form is standard German. Although regional Swiss German dialects are manifold and their differences can be very subtle, there is a common Swiss German speaking style which is used in Swiss broadcasts (e.g. weather reports of *Schweizer Radio und Fernsehen*[2] (SRF)), and that is understood well by the vast majority of the Swiss German speaking people. Broadcast companies are typically interested in the automatic transcription of speech data including (live) subtitling and automatic speech recognition (ASR). In the case of Swiss German, the desired output of the ASR system is standard German, as there is no standardized written form of Swiss German dialect. Due to the mismatch between the dialectal pronunciation of Swiss German and the written form of standard German, these system often fail. In this paper we are exploring methods to close this gap, amongst others a data-driven method.

In the remainder of the paper, we first describe the data investigated in this work and the available annotation (Section 2.). In Section 3., we then describe the involved ASR recognition methods including the data-driven method to improve the pronunciation model and evaluate the results. A conclusion is given in Section 4..



Figure 1: Geographical distribution of the languages of Switzerland (2000); Swiss Federal Statistical Office, `www.bfs.admin.ch`

2. Resources

This section describes the speech corpora that are used in this work.

2.1. SRF Meteo weather report dataset

In this paper we describe the Swiss German SRF Meteo dataset, which *Schweizer Radio und Fernsehen* generously provided us for research purposes. This dataset consists of Swiss German weather reports of SRF Meteo. The speakers speak Swiss German dialect, and the textual annotation is standard German. The dataset consists of 290 Meteo weather report broadcasts with a total of 10,201 speech segments and a total of 6.5 hours of annotated speech and a total of 83,449 annotated words. The contained speech is

[1] `http://www.swissinfo.ch`
[2] `http://www.srf.ch`

to a large extent about weather forecasts and contain a large number of place names.

2.2. GerTV1000h German Broadcast corpus

In LREC 2014 we presented the German broadcast corpus GerTV1000h (Stadtschnitzer et al., 2014). The corpus consists of approximately 1,005 hours of German speech data from the broadcast domain and covers a broad selection of news, interviews, talk shows and documentaries, both from television and radio across several stations. The data subsets from this corpus that were used in this work are summarized in Table 1. Note that we discarded a small amount of utterances in the training set due to mispronunciations, unintelligible words and word fragments.

2.3. Difficult Speech Corpus (DiSCo)

In LREC 2010, Fraunhofer IAIS presented the Difficult Speech Corpus (DiSCo) (Baum et al., 2010). The DiSCo corpus is a collection of datasets from German broadcast domain with challenging acoustical situations. It is intended for the evaluation of speaker and speech recognition systems. The datasets are separated into planned and spontaneous speech. Challenging acoustical situations that are covered by the data subsets are clean, music, applause, and mixed condition. In this work we use the DiSCo corpora for evaluation purposes. The DiSCo data subsets used in this work are listed in Table 2.

3. Experimental setup

3.1. Motivation

The training of a speech recognition system given annotated speech data, a pronunciation model and text is straight forward. However in this setup, we want to train a speech recognition system which is able to translate highly dialectal Swiss German speech data in standard German text. This is desirable, because there is no standardized way of writing Swiss German other than standard German.

3.2. Data preparation

For the experiments the Meteo dataset was split into a training, a development and a testing set, as can be seen in Table 3. We choose to have 260 weather reports in the training set and each 15 weather reports in the development and the testing set. The distribution of the weather reports into the datasets was performed randomly. When considering only the text of the training set for the training of a language model, the development set and the test set have an out-of-vocabulary (OOV) rate of $\text{OOV}_{\text{dev}} = 7.6\%$ and $\text{OOV}_{\text{test}} = 9.1\%$. This seems quite high, however the running OOV rate is acceptable considering the small amount of training data, namely $\text{OOV}_{\text{r,dev}} = 1.4\%$ and $\text{OOV}_{\text{r,test}} = 1.7\%$.

3.2.1. Standard German Speech Recognition

To perform the following data-driven experiments, where a standard German phoneme recognizer is involved, we need to train a standard German speech recognition system. In (Schmidt et al., 2016) we proposed the standard German broadcast ASR system based on recurrent neural networks (RNN) as implemented in (Miao et al., 2015). For the lexicon, we use the pronunciation model trained with Sequitur G2P (Bisani and Ney, 2008) and the German pronunciation database Phonolex (BAS - Bavarian Archive for Speech Signals, 2013). We use the 1-best pronunciation of the model for each word of the lexicon. We use the broadcast text corpora consisting of 75 millions of words, which we already used in other works (Stadtschnitzer et al., 2014) for the training of the language model. We use a 5-gram model, trained with modified shift beta algorithm with back-off weights using IRSTLM (Federico et al., 2008) and a dictionary size of approximately 500,000 words and a language model pruning factor of 10^{-8}. For training of the acoustical model, the GerTV1000h corpus (Section 2.2.) was used.

By the use of time delay neural network (TDNN) architecture with speed-perturbated training data as proposed in (Peddinti et al., 2015) and implemented in the Kaldi Toolkit (Povey et al., 2011), we were able to improve the speech recognition results on the DiSCo evaluation sets as indicated in Table 4. By both the employment of projected long-short memory networks (LSTMP) in the TDNN architecture as proposed in (Cheng et al., 2017) and the use of gated convolutional neural networks (GCNN) (Dauphin et al., 2017) (as implemented in TheanoLM Toolkit (Enarvi and Kurimo, 2016)) for n-best hypotheses rescoring ($n = 200$) we could further improve our standard German broadcast ASR system.

3.3. Swiss German Data-Driven Pronunciation

For the experiments regarding the data-driven Swiss German pronunciation model, we employ the TDNN ASR model (as described in Section 3.2.1., because this was the best configuration available at the time point of the experiments. The results of the standard German TDNN ASR system, which performed well for the standard German evaluation data (Table 4), are naturally worse on the Meteo data ($\text{WER}_{\text{dev}} = 81.0\%$, $\text{WER}_{\text{test}} = 79.5\%$), since there is a large mismatch in speech, phonetics and language between standard German and Swiss German. By replacing the language model trained from broadcast text by a language model trained on the text of the Meteo training dataset, we can reduce this mismatch for the Meteo evaluation data to $\text{WER}_{\text{dev}} = 64.98\%$ and $\text{WER}_{\text{test}} = 64.73\%$. In the following we try to further reduce the mismatch, especially for the mismatch caused in the pronunciation, in a data-driven manner.

3.3.1. Standard German Speech Phoneme Decoder

We first create a phoneme decoder and then use the phoneme decodings to create a Swiss German G2P model. For the training of the standard German Speech phoneme decoder, we use the TDNN acoustical models discussed in Section 3.2.1.. For the training of the standard German phoneme language model, which is required for the phoneme decoder, we replace the words from the text of the Meteo training dataset by its pronunciations derived from the standard German G2P model. Then we train a 5-gram phoneme language model and use it for decoding of the speech signals.

Dataset	#Segments	#Words	Avg. Words	#Unique	Size(h)	Avg. Length (s)
Train	773,631	9,406,119	12.2	243,313	991.9	4.6
Dev	2,348	33,748	14.4	6,377	3.5	5.3

Table 1: Statistics of the GerTV1000h data subsets used in this work

Dataset	#Segments	#Words	Avg. Words	#Unique	Size(h)	Avg. Length (s)
Planned, clean	1,364	9,184	6.7	2,939	0.9	2.4
Spontaneous, clean	2,861	20,740	7.4	4,019	1.9	2.4

Table 2: Statistics of the DiSCo data subsets used in this work

Dataset	#Shows	#Segments	#Words	Avg. Words	#Unique	Size(h)	Avg. Length (s)
Meteo train	260	9,181	75,215	8.2	2,981	5.9	2.3
Meteo dev	15	493	3,995	8.1	742	0.3	2.2
Meteo test	15	527	4,242	8.1	778	0.3	2.2

Table 3: Statistics of the Meteo data subsets used in this work

Model	GerTV dev	DiSCo planned	DiSCo spont.
RNN	17.2	11.9	14.5
TDNN	15.6	11.1	13.2
TDNN-LSTMP	13.7	8.9	10.4
TDNN-LSTMP-GCNN	**12.7**	**8.1**	**9.3**

Table 4: WER [%] results of the standard German speech recognition systems

3.3.2. Data-Driven Pronunciation Modeling

By decoding the Swiss German Meteo training set using the phoneme language model, we get some suggestions of how the speech in the audio data was pronounced. However, the data is organized in utterances, rather than words. Nonetheless, we train a Swiss German G2P model by using the phrases (whitespaces are replaced by an underscore) followed by the pronunciations from the phoneme decodings. The trained Swiss German pronunciation model is able to provide some good suggestions in the n-best list for the pronunciation of several words, as can be derived from Table 5. In this table we also show a non-standardized Swiss German dialectal text annotation of an online Swiss German dictionary[3] for comparison. The pronunciations from the Swiss German G2P were found in a data-driven manner, without any knowledge of the online Swiss German dictionary. As can be learned from Table 5 the pronunciations learned from the Swiss German G2P are often quite near to the textual correspondents from the online Swiss German dictionary.

We then created several lexicons, which were composed by the 1-best standard German pronunciation and a n-best list of the data-driven Swiss German G2P. The intention was to keep the 1-best standard German pronunciation as a backup, when no meaningful Swiss German pronunciation can be found by the method. We then used the enriched lexicons with the standard German TDNN models and a language model trained on the text from the Meteo training dataset. The results are depicted in Figure 2. We optimized the parameter n on the Meteo development set. Derived from the results of the experiments, the optimal variant is to add a 2-best list of the data-driven Swiss German G2P to the 1-best standard German pronunciations. Using this adapted configuration, which includes both reasonable Swiss German and standard German pronunciations, the WER could be reduced for the Meteo development and test set to $\text{WER}_{\text{dev}} = 60.3\%$ and $\text{WER}_{\text{test}} = 56.4\%$.

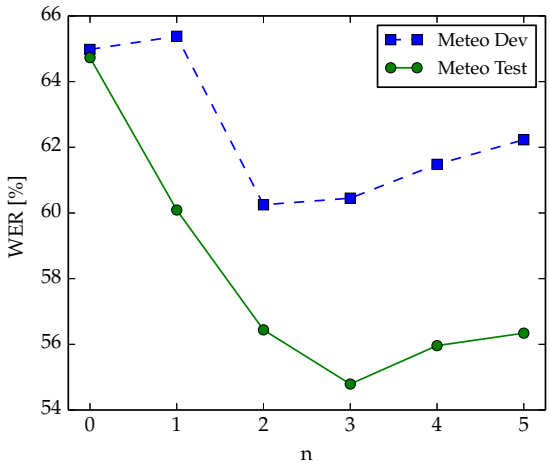


Figure 2: WER for different n for configurations with 1-best standard German pronunciation and n-best Swiss German pronunciations from the speech data driven G2P model

3.4. Supplementary Experiments

We also wanted to evaluate how far we can get, when we train the Swiss German models in a straight forward manner by either using grapheme pronunciations, standard German phoneme pronunciations or the combined pronunciation as described in Section 3.3.2.. When using a grapheme pronunciation, each word is modeled by a sequence of its graphemes (i.e. Montag ⇒ m o n t a g). When using standard German phoneme pronunciations, we use the standard German pronunciation model, which is trained on the standard German Phonolex pronunciation lexicon (BAS - Bavarian Archive for Speech Signals, 2013) using Sequitur G2P (Bisani and Ney, 2008). For the training of the acoustical model we use the training dataset of the SRF Meteo

[3] https://www.pauker.at/pauker/DE_DE/SC/wb

Standard German	German G2P	Data-Driven Swiss German G2P	Swiss German Online Dictionary
Montag	m o: n t a: k	m a: n t i: k	Mäntig
Dienstag	d i: n s t a: k	ts i: S t i: k	Ziischtig
Mittwoch	m I t v O x	m I t b u: x	Mittwuch
Donnerstag	d O n 6 s t a: k	d a n S t i: k	Danschtig
Freitag	f r aI t a: k	f r i: t I k	Fritig
Samstag	z a m s t a: k	Q a m S t i: k	Samschtig
Sonntag	z O n t a: k	z o d I k	Sunntig

Table 5: Phoneme translations of standard German words using the standard German and the speech data-driven Swiss German G2P

dataset. For training the language model we use IRSTLM toolkit (Federico et al., 2008) and we use a 5-gram model with modified shift beta algorithm with back-off weights. For training of the Swiss German ASR system, we either use Eesen (Miao et al., 2015) toolkit, when using long short term memory (LSTM) recurrent neural networks (RNN) with connectionist temporal classification (CTC) training, or the Kaldi toolkit (Povey et al., 2011), when using Hidden Markov Models with Gaussian Mixture Models (HMM-GMM), or hybrid HMM with feed forward Deep Neural Networks (HMM-DNN) or the state-of-the-art time delay neural networks with projected long short-term memory (TDNN-LSTMP) layers. The results are shown in Table 6. The HMM-GMM, DNN and TDNN-LSTMP models from the Kaldi toolkit are trained with bootstrapping and provide more stable results in this setup (i.e. a setup with a small amount of training data) compared to the RNN models, which use Connectionist Temporal Classification (CTC) instead, and which are trained directly on the audio data. It is also remarkable that there is no big difference when comparing standard German grapheme pronunciations to standard German phoneme pronunciations. Both setups perform almost equally well. The use of the combined standard German and Swiss German pronunciation performed slightly worse compared to standard German and grapheme pronunciations for the HMM-GMM case. We believe this is the case because during training the algorithm needs a consistent single pronunciation, so the algorithm can model the pronunciation including the possible mismatches consistently. The TDNN-LSTM models trained with the standard German G2P pronunciations performed best on the Meteo test set ($\text{WER}_{\text{test}} = 23.8\%$) given the experiments performed.

Model	Pron.	Meteo dev	Meteo test
HMM-GMM	GG2P	39.7	28.9
HMM-GMM	Graph.	40.3	29.6
HMM-GMM	SGG2P	41.3	30.8
RNN	GG2P	44.5	32.7
RNN	Graph.	45.0	32.3
HMM-DNN	GG2P	37.1	27.1
HMM-DNN	Graph.	37.7	27.0
TDNN-LSTMP	GG2P	34.9	**23.8**
TDNN-LSTMP	Graph.	**34.8**	24.3

Table 6: WER [%] results of directly trained Swiss German speech recognitions systems using different types of pronunciation lexicons; standard German G2P (GG2P), combined data-driven Swiss German and standard German G2P (SGG2P) or grapheme sequences (Graph.)

4. Conclusion

In this paper we explored the creation of a Swiss German speech recognition system by employing a small Swiss German dataset. Since there is no standardized way to write Swiss German other than standard German, the annotations of the Swiss German audio corpus are standard German, in contrast to the audio material which is highly dialectical Swiss German. The desired output of the Swiss German speech recognition system is again standard German. Unfortunately we lack a Swiss German pronunciation lexicon that maps standard German words into Swiss German pronunciations. We approach this problem by successfully adapting a high-performance standard German speech recognition system to the Swiss German pronunciations by the employment of a Swiss German G2P model which was learned in a data-driven manner by phoneme decodings derived from the standard German speech recognition system with the use of a phoneme language model. It turned out that by adding a 2-best list of Swiss German pronunciations derived from the data-driven Swiss German G2P model to the 1-best standard pronunciations, the adapted model provided the best results, when adapting the standard German model. However, the training of an ASR system directly on the Swiss German data by replacing the missing Swiss German pronunciation by either a standard German phoneme or grapheme sequences, provided even better results. The use of the combined lexicon did not prove to be beneficial when training a system directly on the Swiss German audio data in contrast to the adaptation of the standard German model. For both standard German and Swiss German models, the use of TDNN-LSTMP provided the best results with word error rates as low as 8.1% and 23.8% respectively on the corresponding test sets. This are encouraging results given the small amount of available training data in the Swiss German case.

5. Acknowledgements

The authors would like to thank the *Schweizer Radio und Fernsehen* (SRF) for supporting our research and for their generosity of providing us data for research purposes.

6. Bibliographical References

Bisani, M. and Ney, H. (2008). Joint-Sequence Models for Grapheme-to-Phoneme Conversion. *Speech Communication*, 50:434–451, July.

Cheng, G., Peddinti, V., Povey, D., Manohar, V., Khudanpur, S., and Yan1, Y. (2017). An exploration of dropout with LSTMs. In *Proceedings of INTERSPEECH*, Stockholm, Sweden, Aug.

Dauphin, Y. N., Fan, A., Auli, M., and Grangier, D. (2017). Language modeling with gated convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia.

Enarvi, S. and Kurimo, M. (2016). TheanoLM - An Extensible Toolkit for Neural Network Language Modeling. In *Proc. of INTERSPEECH*, San Francisco, USA.

Federico, M., Bertoldi, N., and Cettolo, M. (2008). IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. In *Proceedings of Interspeech*, Brisbane,Australia.

Garner, P. N., Imseng, D., and Meyer, T. (2014). Automatic Speech Recognition and Translation of a Swiss German Dialect: Walliserdeutsch. In *Proceedings of Interspeech*, Singapore, China, September.

Miao, Y., Gowayyed, M., and Metze, F. (2015). EESEN: End-to-End Speech Recognition using Deep RNN Models and WFST-based Decoding. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, pages 167–174, Scottsdale, Arizona, USA, December.

Peddinti, V., Povey, D., and Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts. In *Proceedings of INTERSPEECH*, Dresden, Germany, September.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The Kaldi Speech Recognition Toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, December. IEEE Catalog No.: CFP11SRW-USB.

Schmidt, C., Stadtschnitzer, M., and Köhler, J. (2016). The Fraunhofer IAIS Audio Mining System: Current State and Future Directions. In *Proceedings of ITG Fachtagung*, Paderborn, Germany.

7. Language Resource References

BAS - Bavarian Archive for Speech Signals. (2013). *Pronunciation lexicon PHONOLEX*. `https://www.phonetik.uni-muenchen.de/forschung/Bas/BasPHONOLEXeng.html`.

Doris Baum and Daniel Schneider and Rolf Bardeli and Jochen Schwenninger and Barbara Samlowski and Thomas Winkler and Joachim Köhler. (2010). *DiSCo - A German Evaluation Corpus for Challenging Problems in the Broadcast Domain*. In Proceedings of the 7th Conference on International Language Resources and Evaluation (LREC), Valetta, Malta.

Michael Stadtschnitzer and Jochen Schwenninger and Daniel Stein and Joachim Koehler. (2014). *Exploiting the large-scale German Broadcast Corpus to boost the Fraunhofer IAIS Speech Recognition System*. Proceedings of the 9th Language Resources and Evaluation Conference (LREC), 2014, Reykjavik, Iceland.

Simulating ASR errors for training SLU systems

Edwin Simonnet, Sahar Ghannay, Nathalie Camelin, Yannick Estève
LIUM, Le Mans University, France
firstname.lastname@univ-lemans.fr

Abstract

This paper presents an approach to simulate automatic speech recognition (ASR) errors from manual transcriptions and describes how it can be used to improve the performance of spoken language understanding (SLU) systems. In particular, we point out that this noising process is very usefull to obtain a more robust SLU system to ASR errors in case of insufficient training data or more if ASR transcriptions are not available during the training of the SLU model. The proposed method is based on the use of both acoustic and linguistic word embeddings in order to define a similarity measure between words dedicated to predict ASR confusions. Actually, we assume that words acoustically and linguistically close are the ones confused by an ASR system. By using this similarity measure in order to randomly substitute correct words by potentially confusing words in manual annotations used to train CRF- or neural- based SLU systems, we augment the training corpus with these new noisy data. Experiments were carried on the French MEDIA corpus focusing on hotel reservation. They show that this approach significantly improves SLU system performance with a relative reduction of 21.2% of concept/value error rate (CVER), particularly when the SLU system is based on a neural approach (reduction of 22.4% of CVER). A comparison to a naive noising approach shows that the proposed noising approach is particularly relevant.

Keywords: spoken language understanding, data augmentation, noising, automatic speech recognition, errors

1. Introduction

Spoken language understanding (SLU) consists in extracting semantic information from speech, and can refer to different tasks. In (De Mori, 2007), the author defines it as "(...) the interpretation of signs conveyed by a speech signal". Similar to previous works from other authors (Hahn et al., 2011; Mesnil et al., 2015), the SLU task targeted in this paper consists in automatically extracting semantic concepts and concept/values pairs from the automatic transcriptions in order to feed a dialogue manager. This task can also be perceived as a slot filling task.

Usually, SLU needs first an automatic transcription of user utterances thanks to an automatic speech recognition (ASR) system. These recognized words are then analyzed in order to extract their meanings. Even best SLU systems see their performance drop when making the transition from processing manual transcriptions to automatic ones, since ASR errors make the SLU task harder. In order to reduce this unavoidable performance decline, it would be relevant to prepare SLU systems to ASR errors during their training. For instance, it is known that Spoken Dialog System (SDS) applied to automatic transcriptions perform better when they are trained on automatic transcriptions rather than manual ones. Since large automatic transcription corpora needed for SDS training are rare, some approaches have been presented in order to simulate ASR errors to train these SDS (Pietquin and Beaufort, 2005; Schatzmann et al., 2007). Such ASR error simulation has also been applied to train discriminative language models in order to improve ASR performance in terms of word error rate (Jyothi and Fosler-Lussier, 2010).

Nowadays, SLU systems are often built through a data-driven approach (Raymond et al., 2006; De Mori et al., 2008; Hahn et al., 2011; Sarikaya et al., 2014; Mesnil et al., 2015; Hakkani-Tür et al., 2016). For slot/filling tasks, manual annotations are usually produced to tag manual transcriptions with semantic labels in order to build a training corpus. In the study presented in this paper we make the assumption – and check it – that building SLU systems from automatic transcriptions yields to SLU systems more robust to ASR errors. Nevertheless, getting automatic transcriptions implies the availability of audio recordings related to the manual semantic annotations, and the availability of an ASR system. More, to get an effective ASR system, some training or adaptation data are needed to tune it while these data are usually the same as the ones used to train the SLU module: this implies to manipulate these data very carefully, in order to avoid biases coming, for instance, from overfitting.

Our objective is to propose an approach to simulate ASR errors from manual transcriptions, in order to create a SLU training corpus closer to the data that the SLU system will have to process on test. In that way, robust SLU systems can be trained even if no ASR data on the specific task is available. This simulation consists in introducing errors in a manual corpus by substituting correct words by similar ones. We assume that words confusable by an ASR system are words that are acoustically close. Such assumption was also retained in (Fosler-Lussier et al., 2002; Stuttle et al., 2004), where ASR simulation was based on the similarity of the phonetization of words to evaluate their confusability. We also consider that these confusable words are also linguistically close. To compute a confusability measure between words, we present in this paper a new approach based on the use of both acoustic and linguistic word embeddings. In our experiments, we measure the impact of this ASR simulation used to modify the training corpus of two different data-driven SLU architectures: one based on conditional random fields (Lafferty et al., 2001) (CRF) and the other one based on a neural network encoder-decoder with attention mechanism (Cho et al., 2014) (NN-EDA). These experiments are carried on the French MEDIA cor-

pus, on which CRF still perform better than neural approaches (Vukotic et al., 2015; Simonnet et al., 2017).

2. ASR confusability measure and simulation of ASR errors

The proposed confusability measure is based on the use of linguistic and acoustic similarities. These similarities are computed from cosine similarities between linguistic and acoustic word embeddings.

The linguistic word embeddings correspond to a combination through a principal component analysis (PCA) of different kinds of word embeddings: *word2vecf* on dependency trees (Levy and Goldberg, 2014), *skip-gram* provided by *word2vec* (Mikolov et al., 2013), and *GloVe* (Pennington et al., 2014), as described in (Ghannay et al., 2016).

The acoustic embeddings correspond to the projection of an arbitrary or fixed dimensional speech segment in a fixed-dimensional space, in a manner that preserves acoustic similarity between words. The approach used to build the acoustic word embeddings was proposed by (Bengio and Heigold, 2014).

2.1. Linear interpolation of linguistic and acoustic similarities

In this study, we propose to use linguistic and acoustic word embeddings to predict ASR confusions. With the purpose to take benefit from both linguistic and acoustic similarities, we propose to use a linear interpolation to combine them. This results to a similarity called $LA_{SimInter}$, defined as:

$$LA_{SimInter}(\lambda, x, y) = (1-\lambda)\times L_{Sim}(x, y)+\lambda\times A_{Sim}(x, y)$$

where x and y are two words, λ is the interpolation coefficient, while L_{Sim} and A_{Sim} are respectively the linguistic and acoustic similarities computed with the cosine similarity applied to respectively the linguistic and acoustic word embeddings of x and y.

Since our goal is to predict ASR confusions, we aim to optimize the λ value for this purpose. To estimate λ, a list of known substitution errors made by an ASR system is used. Let define h an erroneous word hypothesis and $\overline{r}$ the reference word that is substituted with h.

For each word pairs $(h, \overline{r})$ in the list, we compute the probability of using h when the reference word $\overline{r}$ is wrong, *i.e.* the probability of substituting the reference word with the hypothesis one, which is defined as:

$$P(h|\overline{r}) = \frac{\#(h, \overline{r})}{\#\overline{r}}$$

where $\#(h, \overline{r})$ refers to the number of occurrences of the word pair and $\#\overline{r}$ is the number of errors (deletion + substitution) on the reference word.

Based on the similarity score $LA_{SimInter}(h, \overline{r})$ and the probability $P(h|\overline{r})$, we choose the interpolation coefficient $\hat{\lambda}$ that minimizes the mean squared error (MSE) such as:

$$\hat{\lambda} = \underset{\lambda}{\operatorname{argmin}}\, MSE(\forall(h, \overline{r}) : P(h|\overline{r}), LA_{SimInter}(\lambda, h, \overline{r}))$$

By using $LA_{SimInter}$ with $\hat{\lambda}$, it is now possible to propose for a given word its linguistically and acoustically nearest neighbors. We consider the value of $LA_{SimInter}(\hat{\lambda}, x, y)$ as a confusability measure between words x and y, and we call it $confus(x, y)$.

2.2. Simulating errors

To simulate ASR errors, we apply the confusability measure $confus(x, y)$ in order to substitute some correct words from manual transcriptions by one of its confusable words. By fixing a targeted word error rate e (only substitutions), we randomly modify e percent of occurrences of words. These substitutions are made after defining two thresholds: the value c that refers to the lowest value of $confus(\overline{r}, h)$ that permits to substitute the word $\overline{r}$ by the confusable word h, and the value n that limits the number of the possible substitutions of $\overline{r}$ to the n closest h_i words (*i.e.* the words h_i such as the $confus(\overline{r}, h_i)$ value is one of the n highest values for a given $\overline{r}$). The word h is randomly chosen from the list of h_i words that respect the constraints of the n and c thresholds

3. Experimental Setup

This section describes the experimental setup of our work, which is inspired from our previous study (Simonnet et al., 2017).

3.1. The MEDIA corpus

The corpus used here is the MEDIA corpus, collected in the French Media/Evalda project (Bonneau-Maynard et al., 2005) and dealing with negotiation of tourist services. It contains three sets of telephone human/computer dialogues, namely: a training set (TRAIN) with approximately 17.7k sentences, a development set (DEV) with 1.3k sentences and an evaluation set (TEST) containing 3.5k sentences. The corpus was manually annotated with semantic concepts characterized by a label and its value. Evaluations are performed with the DEV and TEST sets and report concept error rates (CER) for concept labels only and concept-value error rates (CVER) for concept-value pairs. It is worth mentioning that the number of concepts annotated in a turn has a large variability and may include more than 30 annotated concepts.

For these experiments, a variant of the ASR system developed by LIUM that won the last evaluation campaign on French language has been used (Rousseau et al., 2014). This system is based on the Kaldi speech recognition toolkit (Povey et al., 2011). A detailed description of the ASR system is given in our previous study (Simonnet et al., 2017).

The word error rates for the training, development, and test corpora are respectively 23.7%, 23.4% and 23.6%.

3.2. SLU features and architectures

Two basis SLU architectures are considered to carry experiments on the MEDIA corpus. The first one is an encoder/decoder recurrent neural architecture with a mechanism of attention (NN-EDA) similar to the one used for machine translation proposed in (Cho et al., 2014). The second one is based on CRF. Both architectures build their training model on the same features encoded with continuous values in the first one and discrete values in the second one.

3.2.1. Set of Features

Word features are added in input with the words. They help the SLU system to achieve better understanding, in-

spired from (Hahn et al., 2011). The features used here (for a given word) are the following: its pre-defined semantic categories which are the MEDIA specific categories and more general categories; sets of syntactic and morphological features; and two ASR confidence measures. The confidence measures are the ASR *posterior* probability (*pap*) and the Multi-Stream Multi-Layer Perceptron (MS-MLP) confidence measure as described in our previous work (Simonnet et al., 2017). Both of these features are an estimation of the reliability of the recognized word.
The detailed description of these features is described in (Simonnet et al., 2017).
The two SLU architectures take all these features except for the confidence measures where only one is taken for a purpose of experimental consistency as it will be described in subsection 3.3.. These architectures also need to be calibrated on their respective hyper-parameters in order to give the best results. The way the best configuration is chosen is described in 4..

3.2.2. Neural Network EDA system

The proposed NN-EDA system, which is inspired from a machine translation architecture, was implemented by starting from the *nmtpy* toolkit (Caglayan et al., 2017). The concept tagging process is considered as a translation problem from words (source language) to semantic concept tags (target language).
The bidirectional RNN encoder is based on Gated Recurrent Units (GRU) and computes annotations for each word from the input sequence. These annotations are the concatenation of the matching forward and backward hidden layer states obtained respectively by the forward and the backward RNN comprising the bidirectional RNN. Thus they contain the summaries of the dialogue turn contexts respectively preceding and following a considered word.
The sequence of annotations is then used by the decoder to compute a context vector (recomputed after each emission of an output label). This computation takes into account a weighted sum of all the annotations computed by the encoder. This weighting depends on the current output target, and is the core of the attention mechanism: a good estimation of these weights allows the decoder to choose parts of the input sequence to pay attention to, in order to make a decision about the current label output.
A more detailed description of the NN-EDA system is given in (Simonnet et al., 2017).

3.2.3. CRF system

Past experiments described in (Hahn et al., 2011) have shown that the best semantic annotation performance on manual and automatic transcriptions of the MEDIA corpus were obtained with CRF systems. More recently in (Vukotic et al., 2015), this architecture has been compared to popular bi-directional RNN (bi-RNN). The result was that CRF systems outperform a bi-RNN architecture on the MEDIA corpus, while better results were observed by bi-RNN on the ATIS (Hemphill et al., 1990) corpus. This is probably explained by the fact that MEDIA contains semantic contents whose mentions are more difficult to disambiguate, and CRFs make it possible to exploit complex contexts more efficiently.
For the sake of comparison with the best SLU system proposed in (Hahn et al., 2011), the Wapiti toolkit was used (Lavergne et al., 2010) in our study. Nevertheless, the set of input features used by the system proposed in this paper is different from the one used in (Hahn et al., 2011). Among the novelties used in our system, we consider syntactic and ASR confidence features and our configuration template is different. After many experiments performed on DEV, our final feature template includes the previous and following instances for words and POS in a unigram or a bigram to associate a semantic label with the current word. Also associated with the current word are semantic categories of the two previous and two following instances. The other features are only considered at the current position.
Furthermore, the tool discretize4CRF[1] is used to apply a discretization function to the ASR confidence measures in order to obtain several discrete values that can be accepted as input features by the CRF.

3.3. ASR simulation

We applied the method presented in sub-section 2.2. in order to simulate ASR errors. Starting from the manual annotations of the MEDIA corpus (without error), we build different datasets. In these simulations we fixed the e value to 20, that represents the rate of words we corrupt randomly in the manual transcriptions.
Two different simulations were tested, by choosing different threshold values n and c;

- **N.7 corpus**: $n = 7$ and $c = 0.4$;
- **N.10 corpus**: $n = 10$ and $c = 0.5$.

Another artificial dataset was created, called **noise.naive corpus**: this corpus does not take into account of the confusability measure. In this dataset, the same $e = 20$ percent of words from manual transcriptions are randomly substituted, by simply choosing randomly a word from the entire MEDIA vocabulary. When a correct word is substituted with a confusable one, we use their confusability measure as an ASR confidence measure. For a purpose of experimental consistency, when working on ASR, we only give one ASR confidence measure among the two available ones in order to always have the same number of confidence measure.

4. Experimental results

Experiments were carried with the MEDIA corpus. Both SLU architectures are optimized to get the best CVER. The training is done on the TRAIN set. For the NN-EDA, validations during training are performed on the DEV set in order to choose the best parameters.
Results on TEST in terms of CER and CVER are reported in tables 1 and 2, where **M** refers to manual corpus, **A** to a corpus composed by automatic transcriptions, and **N** to a noised corpus. TEST corpus is made by ASR transcriptions only, while the nature of TRAIN or DEV corpora varies in our experiments.

[1] https://gforge.inria.fr/projects/discretize4crf/

4.1. Tuning on ASR transcriptions

Since evaluation on TEST is made on ASR transcriptions, we first consider that a DEV corpus composed of automatic transcriptions is available to tune our SLU systems. Such corpus is less hard to collect than a training corpus, since it contains only about 1300 sentences (in comparison to the 17700 in the MEDIA training corpus). Above all, as evocated in the introduction, processing data that are outside the training corpus is easier since no problem of bias and overfitting can be introduced.

Experimental results in this configuration are visible in table 1. We can first notice that our assumption on the importance of getting automatic or ASR simulated transcriptions to get training data as close as possible to the test data is checked: with an **A** TRAIN set, results for both NN-EDA and CRF are very significantly better than with the use of a **M** TRAIN set. It also appears that the CRF architecture significantly outperforms NN EDA for both **M** and **A** training corpora. It is also clear that training a SLU system on manual transcriptions only is largely insufficient to handle ASR transcriptions. The system needs to be prepared to ASR errors.

	NN-EDA		*CRF*	
TRAIN set	*CER*	*CVER*	*CER*	*CVER*
M	31.6	36.2	27.5	31.6
A	22.5	28.3	19.9	25.1
N.7	23.8	29	22.6	27.7
Double N.7	23.2	28.8	26.3	31.3
M+N.7	22.7	28.1	22.6	27.7
M+N.10	23.3	28.5	23.2	28.3
M+N.naive	23.7	28.8	25	30.3
M+A	20.7	25.8	20.2	25.3
M+N.7+A	20.2	26	29.1	33.0

Table 1: Comparison on CER and CVER obtained on ASR TEST with an ASR DEV.

Training however on a noised corpus (line N.7) gets interesting results. It clearly gets an improvement from the poor results gotten on manual transcriptions only. It gets close to the results of using pure ASR transcription and so confirm that our approach to simulate ASR errors seems acceptable for this task. Training on a double noised corpus (line Double N.7, in which two successive ASR error simulations on the same train were applied) can improve a little the results on the NN-EDA while it decreases strongly the CRF results.

Better results can be achieved by combining manual and noised corpus. By using the N.7 dataset in combination with the manual corpus, the results are just as good as the pure ASR for the NN-EDA. The CRF gets the same results as for N.7 only.

We can also see the comparison between the different types of noise. The N.7 gets better results than N.10 showing that by substituting correct words with globally less similar words decreases the results. Furthermore, even if applying naive noise gets better results than using manual transcriptions without errors, we obtain the worst scores among noising approaches with it. This globally shows the importance of an intelligently generated noise, and implicitly validates our ASR error simulation approach.

Finally the best achieved results that outperform even pure ASR alone are obtained by training the SLU system on a combination of ASR and manual corpus. Both SLU systems find their best results in this configuration and the gap between CRF and NN-EDA has been strongly reduced from the experiments on ASR only or manual only. Training on a triple combination of manual, ASR and noised corpus does not increase more these results.

In general, CRF significantly outperforms NN-EDA when these systems are trained from a manual or an ASR corpus. But NN-EDA takes better benefit from ASR simulation, or from manual and ASR combination than CRF. At the end, best results for both NN-EDA and CRF are now very close, showing some potentialities of neural networks, not shared by CRF, to learn relevant information from noisy data.

4.2. Tuning on manual transcriptions

In this section we explore the scenario in which ASR data are *not* available to tune the SLU system. In that case, this explicitly means that the DEV corpus can not be issued from ASR. This can become problematic when the SLU system needs to compute some validations during training, which is the case for NN-EDA. CRF otherwise do not use the DEV while training. Thus the results visible in table 2 are only for NN-EDA (the CRF scores stay unchanged).

		NN EDA	
TRAIN set	*DEV set*	*CER*	*CVER*
M	M	33.9	38.2
N.7	N.7	23.5	28.6
M+N.7	N.7	23.1	28.5

Table 2: Comparison on CER and CVER obtained on ASR TEST with no ASR TRAIN or DEV.

In general, except for the noised train corpus alone, it gives better results to validate on an ASR DEV corpus, closer in nature to the TEST data.

Nevertheless, even if those results are a little bit worse than the ones reached by validating on an ASR DEV corpus, we can notice that it is possible to very significantly improve the SLU systems by applying our ASR error simulation approach in order to enrich or to noise the SLU training and development data which are composed by only manual transcriptions.

5. Conclusion

Two SLU architectures based on NN-EDA and CRF were compared in this study. An ASR error simulation based on a confusability measure built from acoustic and linguistic word embeddings has been proposed and used in order to noise a manual annotated corpus. Experiments show that this noising process is relevant to enrich and to prepare an SLU training corpus. If no ASR system is available to prepare these data, our proposition offers a very significant improvement of the SLU performances, from 36.2% of CVER with only manual annotations in the training corpus, in

comparison to 28.5% of CVER by applying our approach: this represents a relative reduction of 21.2% of the concept/value errors. Another interesting result of this study is the contraction of the differences, in terms of CER or CVER, between CRF and NN-EVA on the MEDIA corpus. No advance on this corpus has been made from 2011 (Hahn et al., 2011) and CRF are still dominant. Our results show that it is now possible to get similar results with neural network architecture. We expect to propose new contributions to make neural networks more effective than CRF, that have reached a plateau several years ago on this task. In a close future, we will also consider other ASR error simulation approaches to compare their impact to ours to prepare and enrich SLU training corpus. We will also experiment the use of our ASR simulation on other tasks, like ASR error detection for instance.

6. Bibliographical References

Bengio, S. and Heigold, G. (2014). Word embeddings for speech recognition. In *INTERSPEECH*, pages 1053–1057.

Bonneau-Maynard, H., Rosset, S., Ayache, C., Kuhn, A., and Mostefa, D. (2005). Semantic annotation of the french media dialog corpus. In *Ninth European Conference on Speech Communication and Technology*.

Caglayan, O., García-Martínez, M., Bardet, A., Aransa, W., Bougares, F., and Barrault, L. (2017). Nmtpy: A flexible toolkit for advanced neural machine translation systems. *arXiv preprint arXiv:1706.00457*.

Cho, K., van Merrienboer, B., Gulcehre, C., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*.

De Mori, R., Bechet, F., Hakkani-Tur, D., McTear, M., Riccardi, G., and Tur, G. (2008). Spoken language understanding. *IEEE Signal Processing Magazine*, 25(3).

De Mori, R. (2007). Spoken language understanding: A survey. In *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, pages 365–376. IEEE.

Fosler-Lussier, E., Amdal, I., and Kuo, H.-K. J. (2002). On the road to improved lexical confusability metrics. In *ISCA Tutorial and Research Workshop (ITRW) on Pronunciation Modeling and Lexicon Adaptation for Spoken Language Technology*.

Ghannay, S., Favre, B., Esteve, Y., and Camelin, N. (2016). Word embedding evaluation and combination. In *of the Language Resources and Evaluation Conference (LREC 2016), Portoroz (Slovenia)*, pages 23–28.

Hahn, S., Dinarelli, M., Raymond, C., Lefevre, F., Lehnen, P., De Mori, R., Moschitti, A., Ney, H., and Riccardi, G. (2011). Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1569–1583.

Hakkani-Tür, D., Tur, G., Celikyilmaz, A., Chen, Y.-N., Gao, J., Deng, L., and Wang, Y.-Y. (2016). Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *Proceedings of The 17th Annual Meeting of the International Speech Communication Association*.

Hemphill, C. T., Godfrey, J. J., Doddington, G. R., et al. (1990). The atis spoken language systems pilot corpus. In *Proceedings of the DARPA speech and natural language workshop*, pages 96–101.

Jyothi, P. and Fosler-Lussier, E. (2010). Discriminative language modeling using simulated asr errors. In *Eleventh Annual Conference of the International Speech Communication Association*.

Lafferty, J., McCallum, A., Pereira, F., et al. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289.

Lavergne, T., Cappé, O., and Yvon, F. (2010). Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics, July.

Levy, O. and Goldberg, Y. (2014). Dependency based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 302–308.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 23(3):530–539.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, volume 12.

Pietquin, O. and Beaufort, R. (2005). Comparing asr modeling methods for spoken dialogue simulation and optimal strategy learning. In *Ninth European Conference on Speech Communication and Technology*.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, number EPFL-CONF-192584. IEEE Signal Processing Society.

Raymond, C., Bechet, F., De Mori, R., and Damnati, G. (2006). On the use of finite state transducers for semantic interpretation. *Speech Communication*, 48(3):288–304.

Rousseau, A., Boulianne, G., Deléglise, P., Estève, Y., Gupta, V., and Meignier, S. (2014). LIUM and CRIM ASR system combination for the REPERE evaluation campaign. In *International Conference on Text, Speech, and Dialogue*, pages 441–448. Springer.

Sarikaya, R., Hinton, G. E., and Deoras, A. (2014). Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(4):778–784.

Schatzmann, J., Thomson, B., and Young, S. (2007). Error

simulation for training statistical dialogue systems. In *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, pages 526–531. IEEE.

Simonnet, E., Ghannay, S., Camelin, N., Esteve, Y., and Renato, D. M. (2017). ASR error management for improving spoken language understanding. In *INTERSPEECH*.

Stuttle, M., Williams, J., and Young, S. (2004). A framework for dialog systems data collection using a simulated asr channel. In *ICSLP 2004*.

Vukotic, V., Raymond, C., and Gravier, G. (2015). Is it time to switch to word embedding and recurrent neural networks for spoken language understanding? In *InterSpeech*.

Evaluation of Feature-Space Speaker Adaptation for End-to-End Acoustic Models

Natalia Tomashenko[1,2] **and Yannick Estève**[1]
[1] LIUM, University of Le Mans, France
[2] ITMO University, Saint-Petersburg, Russia
{natalia.tomashenko, yannick.esteve}@univ-lemans.fr

Abstract

This paper investigates speaker adaptation techniques for bidirectional long short term memory (BLSTM) recurrent neural network based acoustic models (AMs) trained with the connectionist temporal classification (CTC) objective function. BLSTM-CTC AMs play an important role in end-to-end automatic speech recognition systems. However, there is a lack of research in speaker adaptation algorithms for these models. We explore three different feature-space adaptation approaches for CTC AMs: feature-space maximum linear regression, i-vector based adaptation, and maximum a posteriori adaptation using GMM-derived features. Experimental results on the TED-LIUM corpus demonstrate that speaker adaptation, applied in combination with data augmentation techniques, provides, in an unsupervised adaptation mode, for different test sets, up to 11–20% of relative word error rate reduction over the baseline model built on the raw filter-bank features. In addition, the adaptation behavior is compared for BLSTM-CTC AMs and time-delay neural network AMs trained with the cross-entropy criterion.

Keywords: Speaker adaptation, end-to-end speech recognition, GMM-derived features, deep neural network, acoustic model

1. Introduction

Recently, various neural end-to-end approaches to automatic speech recognition (ASR) have been proposed in the literature (Hannun et al., 2014; Bahdanau et al., 2016; Collobert et al., 2016; Fritz and Burshtein, 2017; Chan et al., 2016; Audhkhasi et al., 2017). End-to-end acoustic models (AMs) (Chorowski et al., 2014; Graves and Jaitly, 2014; Miao et al., 2015; Zhang et al., 2017) attempt to map an acoustic signal to a phoneme or grapheme sequence directly by means of neural network models. They have been developed as an alternative to the traditional hybrid approach based on hidden Markov models coupled to deep neural networks (HMM-DNNs) (Hinton et al., 2012).

Speaker adaptation is an essential component of state-of-the-art hybrid HMM-DNN AMs, and a variety of adaptation methods have been developed for DNNs. They include linear transformations, that can be applied at different levels of the DNN-HMM architecture (Gemello et al., 2006; Seide et al., 2011); regularization techniques, such as L2-prior regularization (Liao, 2013) or Kullback-Leibler divergence regularization (Yu et al., 2013); model-space adaptation (Siniscalchi et al., 2013; Swietojanski and Renals, 2014; Huang et al., 2014); multi-task learning (MTL) (Price et al., 2014; Li et al., 2015; Huang et al., 2015); factorized adaptation (Li et al., 2014); adaptation with speaker codes (Xue et al., 2014); the use of auxiliary features, such as i-vectors (Saon et al., 2013; Senior and Lopez-Moreno, 2014) or GMM-derived (GMMD) features (Tomashenko and Khokhlov, 2014), and many others.

However, the major part of the published works, devoted to end-to-end technology, does not use any speaker adaptation techniques. This lack may be justified by the strong focus of these papers on the neural core of the technology they introduce.

A few papers have offered some preliminary and promising information about the benefits provided by some speaker adaptation techniques to end-to-end AMs. In (Miao et al., 2016), vocal tract length normalization (VTLN) (Lee and Rose, 1996) has been applied to filterbank features, for a neural end-to-end AM training through connectionist temporal classification (CTC), providing 3% of relative word error rate reduction (WERR). Speaker i-vectors, appended to the acoustic features, are used in (Audhkhasi et al., 2017) for training phone and word CTC models. Also features, adapted using feature-space maximum likelihood linear regression (fMLLR), are used to train attention-based RNNs (Chorowski et al., 2014). However in these works (Audhkhasi et al., 2017; Chorowski et al., 2014), no comparison results with the unadapted models are given. Work (Yi et al., 2016) proposes a CTC regularized model adaptation method for the accent adaptation task. Speaker adaptation with speaker codes of RNN-BLSTM AMs is studied in (Huang et al., 2016) for the phone recognition task, where AMs were trained with cross-entropy (CE) criterion, and the adaptation provides about 10% of relative reduction in phone error rate.

The aim of this paper is to explore the efficiency of speaker adaptation for end-to-end ASR systems on the example of CTC-BLSTM AMs (or shortly, CTC AMs). For this purpose we implemented three different speaker adaptation algorithms to this type of AMs and performed an experimental analysis of these methods. Furthermore, a comparative study of the adaptation techniques was conducted for CTC AMs and time-delay neural network (TDNN) AMs trained with traditional frame-wise cross-entropy (CE) criterion.

The rest of the paper is organized as follows. A quick overview of the end-to-end AMs, studied in this paper, is introduced in Section 2. A speaker adaptation technique, based on the use of GMMD features and recently proposed by the authors for DNN-HMM AMs, is presented in Section 3. for CTC AMs. Section 4. describes the experimental results for different adaptation algorithms. Finally, the conclusions are given in Section 5.

2. Review of End-to-End Speech Recognition

One of the first major steps in the direction of end-to-end systems was introduced in (Graves et al., 2013) where, for the phoneme recognition task, a deep BLSTM recurrent neural network (RNN) model was trained to map directly acoustic sequences to phonetics ones. This was done by using the CTC objective function (Graves et al., 2006). The BLSTM-CTC models are used in this study. Alternative approaches to end-to-end ASR include attention mechanism (Chorowski et al., 2014; Bahdanau et al., 2016; Chan et al., 2016), convolutional neural networks (CNNs) trained with CTC loss (Collobert et al., 2016; Zhang et al., 2014; Wang et al., 2017), RNN transducers (Graves et al., 2013) and others.

2.1. Deep Bidirectional LSTMs

RNNs provide a powerful extension of feed-forward DNN models by adding connections between different types of units, including backward connections to previous layers. The use of recurrence over the temporal dimension allows RNNs to model the dynamic temporal behavior of the process.

In order to capture information from the whole input sequence, the bidirectional RNN (BRNN) architecture was proposed (Schuster and Paliwal, 1997). In BRNNs, data are processed in two directions with two hidden layers, which are then input further to the same output layer. As shown in the upper part of Figure 1, for a sequence of input vectors $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$, a recurrent forward hidden layer of a BRNN $\overrightarrow{\mathbf{H}} = \{\overrightarrow{\mathbf{h}}_1, \ldots, \overrightarrow{\mathbf{h}}_T\}$ computes a sequence of hidden outputs for $t = 1, \ldots, T$, and an additional recurrent layer $\overleftarrow{\mathbf{H}} = \{\overleftarrow{\mathbf{h}}_1, \ldots, \overleftarrow{\mathbf{h}}_T\}$ computes the backward sequence of hidden outputs for $t = T, \ldots, 1$:

$$\begin{cases} \overrightarrow{\mathbf{h}}_t = f(\mathbf{W}_{x\overrightarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{\overrightarrow{h}}), \\ \overleftarrow{\mathbf{h}}_t = f(\mathbf{W}_{x\overleftarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{\overleftarrow{h}}), \\ \mathbf{y}_t = g(\mathbf{W}_{\overrightarrow{h}y}\overrightarrow{\mathbf{h}}_t + \mathbf{W}_{\overleftarrow{h}y}\overleftarrow{\mathbf{h}}_t + \mathbf{b}_y), \end{cases} \quad (1)$$

where $\mathbf{W}_{x\overrightarrow{h}}$, $\mathbf{W}_{x\overleftarrow{h}}$ are the weight matrices connecting inputs to hidden units; $\mathbf{W}_{\overrightarrow{h}\overrightarrow{h}}$, $\mathbf{W}_{\overleftarrow{h}\overleftarrow{h}}$ are the weight matrices connecting hidden units from time $t-1$ to time t; $\mathbf{W}_{\overrightarrow{h}y}$, $\mathbf{W}_{\overleftarrow{h}y}$ are the weight matrices connecting the output layer to the hidden layer; $\mathbf{b}_h$, $\mathbf{b}_y$ are bias vectors for the hidden states and the outputs correspondingly; $f(\cdot)$, $g(\cdot)$ are the hidden and output layer activation functions correspondingly.

State-of-the-art ASR systems have deep architectures with several hidden layers, where the forward and backward hidden outputs at time t are concatenated and this concatenation $\overrightarrow{\mathbf{h}}_t \oplus \overleftarrow{\mathbf{h}}_t$ is input into the next recurrent layers.

For training RNN models, a back-propagation-through-time (BPTT) learning algorithm is typically used (Werbos, 1990). However, in practice, training RNNs to learn long-term temporal dependencies can be difficult due to the vanishing and exploding gradient problems (Bengio et al., 1994). To avoid the long-term dependency problem, LSTM neural networks were introduced in (Hochreiter and Schmidhuber, 1997). In the end-to-end ASR framework, LSTMs units are used as the structural elements of BRNNs (Miao et al., 2015; Miao et al., 2016; Graves et al., 2013; Sak et al., 2015).

2.2. Connectionist Temporal Classification

In the CTC approach, the alignment between the inputs and target labels is unknown. CTC can be implemented with a softmax output layer using an additional unit for the blank label $\emptyset$. The symbol $\emptyset$ corresponds to no output and is used to estimate the probability of outputting no label at a given time. The network is trained to optimize the total log-probability of all valid label sequences for training data. A set of valid label sequences for an input sequence is defined as the set of all possible label sequences of the input with the target labels in the correct order with repetitions and with label $\emptyset$ allowed between any labels. Targets for CTC training can be computed using finite state transducers (FSTs) (Sak et al., 2015), and the forward-backward algorithm can be used to calculate the CTC loss function.

State transition probability distribution and state priors are not required for CTC approach, in contrast to the hybrid DNN-HMM system. Several types of output units for CTC training have been explored in the literature, such as phones (or graphemes) (Miao et al., 2015), words (Audhkhasi et al., 2017) or grams (Liu et al., 2017). Due to the large number of word outputs in acoustic-to-word CTC models, they require significantly more training data in comparison with traditional ASR systems (Audhkhasi et al., 2017). A maximum a posteriori (MAP) training criterion instead of CTC was used in (Fritz and Burshtein, 2017) to train an end-to-end ASR system.

3. Speaker Adaptation

In this paper we focus on the feature space adaptation techniques for end-to-end acoustic models. Three different types of AM adaptation were explored in this paper: (1) fMLLR (Gales, 1998), (2) adaptation using i-vectors (Senior and Lopez-Moreno, 2014), and (3) MAP (Gauvain and Lee, 1994) adaptation using GMMD features (Tomashenko and Khokhlov, 2014; Tomashenko et al., 2016b). In this section we describe the adaptation approach, which is based on using speaker-adapted GMMD features for training BLSTM-CTC models.

3.1. GMM-Derived Features for BLSTM-CTC Models

The use of log-likelihoods from a GMM model for training a multilayer perceptron (MLP) recognizer was investigated in (Pinto and Hermansky, 2008). Construction of GMMD features for adapting hybrid DNN-HMM AMs was proposed in (Tomashenko and Khokhlov, 2014; Tomashenko and Khokhlov, 2015; Tomashenko et al., 2016a), where it was demonstrated, using MAP and fMLLR adaptation as examples, that this type of features provide a solution for efficient transferring GMM-HMM adaptation algorithms into the DNN framework.

We can train DNN models directly on GMMD features, as it was done in (Tomashenko and Khokhlov, 2014; Tomashenko and Khokhlov, 2015; Tomashenko et al.,

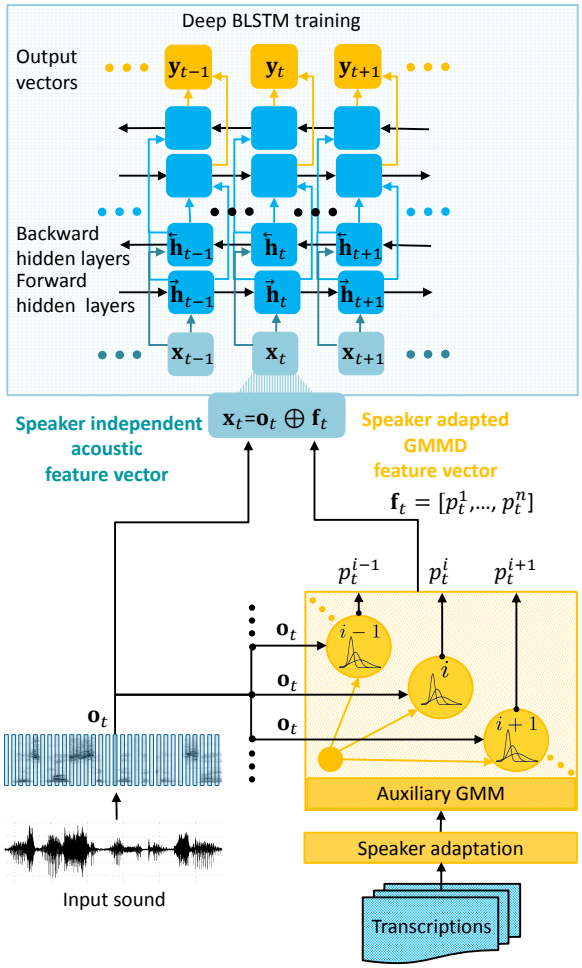


Figure 1: Speaker adaptive training for the BLSTM AM using GMMD features.

2016c), or use them in combination with other conventional features. In this paper we present incorporation of the adapted GMMD features into the recipe for training sequence-to-sequence AMs.

The scheme for speaker adaptive training (SAT) of AMs models with GMM-based adaptation framework is shown in Figure 1. An auxiliary monophone GMM-HMM model is used to transform acoustic feature vectors into log-likelihoods vectors. At this step, speaker adaptation of the auxiliary speaker-independent (SI) GMM-HMM model is performed for each speaker in the training corpus using correct transcriptions and a new speaker-adapted (SA) GMM-HMM model is created in order to obtain SA GMMD features.

For a given acoustic feature vector, a new GMM-derived feature vector is obtained by calculating log-likelihoods across all the states of the auxiliary GMM model on the given vector. Suppose $\mathbf{o}_t$ is the acoustic feature vector at time t, then the new GMM-derived feature vector $\mathbf{f}_t$ is calculated as follows:

$$\mathbf{f}_t = [p_t^1, \ldots, p_t^n], \tag{2}$$

where n is the number of states in the auxiliary GMM-HMM model,

$$p_t^i = \log\left(P(\mathbf{o}_t \mid s_t = i)\right) \tag{3}$$

is the log-likelihood estimated using the GMM-HMM. Here s_t denotes the state index at time t.

The adapted GMMD feature vector $\mathbf{f}_t$ is concatenated with the original vector $\mathbf{o}_t$ to obtain vector $\mathbf{x}_t$. These features are used as the input for training a SAT BLSTM-CTC AM. The proposed approach can be considered a feature space transformation technique with respect to BLSTM-CTC AM trained on GMMD features.

3.2. MAP Adaptation

In this work we use the MAP adaptation algorithm (Gauvain and Lee, 1994) in order to adapt the SI GMM model. Speaker adaptation of a DNN-HMM model built on GMMD features is performed through the MAP adaptation of the auxiliary GMM model, which is used for calculating GMMD features. Let m denote an index of a Gaussian in the SI acoustic model (AM), and $\boldsymbol{\mu}_m$ the mean of this Gaussian. Then the MAP estimation of the mean vector is

$$\widehat{\boldsymbol{\mu}}_m = \frac{\tau \boldsymbol{\mu}_m + \sum_t \gamma_m(t)\mathbf{o}_t}{\tau + \sum_t \gamma_m(t)}, \tag{4}$$

where τ is the parameter that controls the balance between the maximum likelihood estimate of the mean and its prior value; $\gamma_m(t)$ is the posterior probability of Gaussian component m at time t.

4. Experiments

4.1. Data Sets

The experiments were conducted on the TED-LIUM corpus (Rousseau et al., 2014). We used the last (second) release of this corpus. This publicly available data set contains 1495 TED talks that amount to 207 hours (141 hours of male, 66 hours of female) speech data from 1242 speakers, 16kHz. For experiments with SAT and adaptation we removed from the original corpus data for those speakers, who had less than 5 minutes of data, and from the rest of the corpus we made four data sets: training set, development set and two test sets. Characteristics of the obtained data sets are given in Table 1. For evaluation a 4-gram lan-

Characteristic		Data set			
		Train	Dev.	Test_1	Test_2
Duration, hours	Total	171.66	3.49	3.49	4.90
	Male	120.50	1.76	1.76	3.51
	Female	51.15	1.73	1.73	1.39
Duration per speaker, minutes	Mean	10.0	15.0	15.0	21.0
	Min.	5.0	14.4	14.4	18.3
	Max.	18.3	15.4	15.4	24.9
Number of speakers	Total	1029	14	14	14
	Male	710	7	7	10
	Female	319	7	7	4
Number of words	Total	-	36672	35555	51452

Table 1: Data sets statistics.

guage model (LM) with 152K word vocabulary was used. The LM is similar to the "small" one, which is currently used in the Kaldi *tedlium s5_r2* recipe. The only difference is that we modified a little a training set, removing from it those data, that present in our test and development sets.

4.2. Baseline Systems

We used the open-source Kaldi toolkit (Povey et al., 2011) and the Eesen system (Miao et al., 2015) for the experiments presented in this paper. Three baseline SI AMs were trained using the Eesen system in a similar manner, and differ only in the front-end processing. The following three type of features were used:

1. *fbanks* $\oplus\Delta\oplus\Delta\Delta$ *(dimension* $= 120$*)*: 40-dimensional filterbank features appended with their first and second-order temporal derivatives;
2. *high-resolution MFCC features (dimension* $= 40$*)*: features extracted without dimensionality reduction, keeping all 40 cepstra;
3. *bottleneck (BN) features (dimension* $= 40$*)*.

The first type of features is the same, as proposed in the original Eesen recipe for the TED-LIUM corpus. For the AMs with the two other types of features, also the two types of data augmentation strategies were applied for the speech training data: speed perturbation (with factors 0.9, 1.0, 1.1) and volume perturbation, as in (Peddinti et al., 2015).
The first baseline AM was trained as described in (Miao et al., 2015) with the CTC criterion and the deep BLSTM architecture. The BLSTM network contains five bidirectional LSTM layers with 320 memory cells in each forward and backward sub-layer. The input features were normalized with per-speaker mean subtraction and variance normalization. The output layer is a 41-dimensional softmax layer with the units, corresponding to 39 context-independent phones, 1 noise model and 1 blank symbol.
The third SI AM was trained on BN features (Grézl et al., 2007). A DNN model for extraction 40-dimensional BN features was trained with the following topology: one 440-dimensional input layer; four hidden layers (HLs), where the third HL was a BN layer with 40 neurons and other three HLs were 1500-dimensional; the output layer was 4052-dimensional. The input features for training this BN extractor were 440-dimensional (40×11): 40-dimensional high-resolution MFCCs spliced across 11 neighboring frames (± 5).

4.3. Adapted Models

Three types of AM adaptation were empirically explored in this section: fMLLR, adaptation using i-vectors, and MAP adaptation using GMMD features. For all the adapted AMs the same data augmentation strategies were applied during the training, as for the SI ones. All the SAT models were trained with the same neural network topology (except for the input layer) and training criterion, as described in Section 4.2. for SI AMs. The six SAT AMs were trained on the following features:

4. *MFCC* $\oplus$ *i-vectors (dimension* $= 140$*)*;
5. *BN* $\oplus$ *i-vectors (dimension* $= 140$*)*;
6. *BN with fMLLR (dimension* $= 40$*)*;
7. *MFCC* $\oplus$ *GMMD (dimension* $= 167$*)*;
8. *BN* $\oplus$ *GMMD (dimension* $= 167$*)*;
9. *BN with fMLLR* $\oplus$ *GMMD (dimension* $= 167$*)*.

For the AMs trained on features #4 and #5, the 100-dimensional on-line i-vectors were calculated as in (Peddinti et al., 2015), and the statistic for i-vectors was updated every two utterances during the training.
For AMs #7–#9 we used BN features to train the auxiliary GMM model for GMMD feature extraction. The speaker-adapted GMMD features were obtained in the same way as described in Section 3. Parameter τ in MAP adaptation (see Formula (4)) was set equal to 5 for both acoustic model training and decoding.

4.4. Adaptation Results for CTC AMs

Unless explicitly stated otherwise, the adaptation experiments were conducted in an unsupervised mode on the test data using transcripts from the first decoding pass obtained by the best baseline SI model.

#	Features	WER,%		
		Dev.	Test_1	Test_2
1	fbanks $\oplus\Delta\oplus\Delta\Delta$	14.57	11.71	15.29
2	MFCC	13.21	11.16	14.15
3	BN	13.63	11.84	15.06
4	MFCC $\oplus$ i-vectors	12.92	10.45	14.09
5	BN $\oplus$ i-vectors	13.47	11.37	14.31
6	BN with fMLLR	12.45	10.96	13.79
7	MFCC $\oplus$ GMMD	11.95	10.20	14.04
8	BN $\oplus$ GMMD	11.66	10.14	13.88
9	BN with fMLLR $\oplus$ GMMD	**11.63**	**9.91**	**13.58**
10	BN $\oplus$ GMMD*	11.67	10.11	13.70
11	BN with fMLLR $\oplus$ GMMD*	**11.41**	9.93	**13.47**

Table 2: Summary of adaptation results for CTC AMs. GMMD* correspond to the GMMD features, obtained using the first decoding pass by the SAT AM (by default, in all other experiments, the SI model is used instead)

#	Features	WER,%		
		Dev.	Test_1	Test_2
2	MFCC	13.69	11.34	14.38
3	BN	12.32	10.48	14.00
4	MFCC $\oplus$ i-vectors	11.63	9.62	13.28
5	BN $\oplus$ i-vectors	11.62	9.75	13.30
6	BN with fMLLR	**10.70**	**9.28**	**12.84**
7	MFCC $\oplus$ GMMD	11.30	9.75	13.74
8	BN $\oplus$ GMMD	11.07	9.75	13.55
9	BN with fMLLR $\oplus$ GMMD	10.92	9.54	13.27
10	BN $\oplus$ GMMD*	10.29	9.20	13.04
11	BN with fMLLR $\oplus$ GMMD*	**10.15**	**9.06**	**12.84**

Table 3: Summary of adaptation results for TDNN AMs.

The performance results in terms of word error rate (WER) for SI and SAT AMs models are presented in Table 2. The first three lines of the table (#1–#3) correspond to the baseline SI AMs, which were trained as described in Section 4.2., where the very first line represents the Eesen baseline (Miao et al., 2015). The next six lines (#4–#9) show the results for the adapted models. The numeration in Table 2 coincides with the numeration in Sections 4.2. and 4.3..

The two last lines of the table (#10 and #11) are obtained with the same AMs as the lines #8 and #9 correspondingly, but for the extraction of GMMD-adapted features in #10 and #11 (marked with the "*" in Table 2, and further in Figure 2 and Tables 3, 4), we used the transcriptions from the adapted model #6). Notice, that for all other tests (#7–#9) we used transcriptions from the SI model #2.

The best result among all the systems #1–#9 is obtained by system #9, which corresponds to the use of MAP-adapted GMMD features appended with fMLLR-adapted BN features. It can be only slightly improved (#11) for two sets by using the adapted model in the first decoding pass (for GMMD*). Among all the adaptation methods, applied separately (#4–#8), the MAP adaptation of GMMD features shows the best performance with both BN and MFCC features.

4.5. Comparison of Adaptation Behavior for BLSTM-CTC and TDNN AMs.

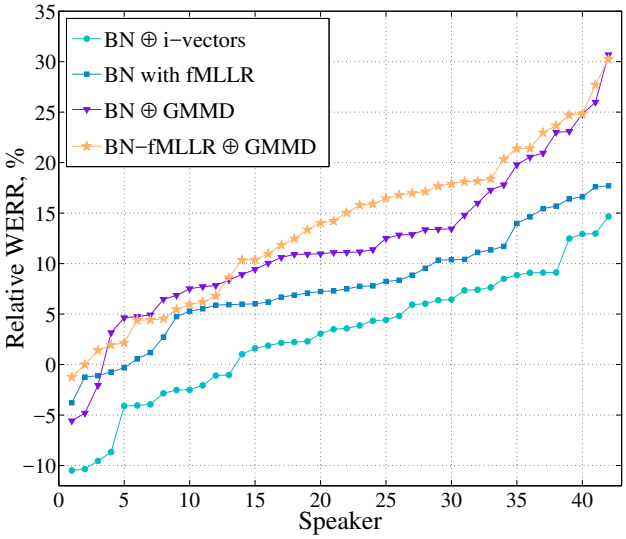
In this series of experiments we aim to compare the adaptation behavior of SAT CTC models with the different type of neural network AMs. For this purpose we chose a TDNN model topology, because such models are shown to achieve the best result in many state-of-the ASR systems (Peddinti et al., 2015). These AMs were trained the with the CE criterion.

We built the same set of SI and SAT AMs, as before for CTC-AMs (see Sections 4.2. and 4.3.), except for #1. All SI ans SAT TDNN models were trained in a similar way and have the same model topology. They differ only in the type of the input features.

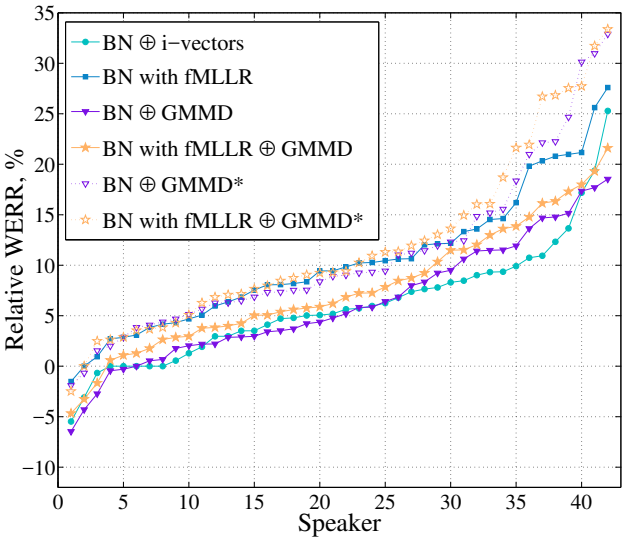
The topology of the TDNN models was similar to the one described in (Peddinti et al., 2015), except for the number of hidden layers and slightly different subsequences of splicing and sub-sampling indexes. The temporal context was $[t-16, t+12]$ and the splicing indexes used here were $[-2, 2]$, $\{-1, 2\}$, $\{-3, 3\}$, $\{-7, 2\}$, $\{0\}$, $\{0\}$. This model had 850-dimensional hidden layers with rectified linear units (ReLU) (Dahl et al., 2013) activation functions and about 4000-dimensional output layer.

The results for TDNN AMs are reported in Table 3. Also Figure 2 presents the comparison of different adaptation algorithms in terms of relative WERR for the speakers from test and development datasets for BLSTM-CTC (Figure 2a) and TDNN (Figure 2b) AMs. The relative WERR is calculated with respect to the SI AMs trained on BN features. For TDNN AMs we also added in Figure 2b the results obtained with the use of SAT AMs for the first decoding pass, because they provide a consistent additional improvement in performance in comparison with the use of SI AMs.

Table 4 shows relative WERRs for BLSTM-CTC and



(a) CTC



(b) TDNN

Figure 2: Relative WER reduction (WERR) for the speakers from test and development datasets for different adaptation algorithms with respect to the SI AMs, trained on BN features (#3). Results are ordered in ascending WERR order for each AM.

TDNN AMs in comparison with the best corresponding SI AMs (#2 for CTC and #3 for TDNN). We can see, that the optimal choice of features depends on the AM architecture. For SI AMs, BNs have appeared to perform better than MFCCs for TDNN AMs, but for CTC AMs the situation is reversed. Also for SAT CTC and SAT TDNN AMs the ranking of the systems by the WER is different.

5. Conclusions

This paper has explored how the end-to-end ASR technology can benefit from speaker adaptation and demonstrated that speaker adaptation has remained an essential mecha-

#	Features	**CTC**: rel. WERR,%			**TDNN**: rel. WERR,%		
		Dev.	Test_1	Test_2	Dev.	Test_1	Test_2
4	MFCC $\oplus$ i-vectors	2.2	6.4	0.4	5.6	8.2	5.1
5	BN $\oplus$ i-vectors	-2.0	-1.9	-1.1	5.7	7.0	5.0
6	BN with fMLLR	5.8	1.8	2.5	13.2	11.5	8.3
7	MFCC $\oplus$ GMMD	9.5	8.6	0.8	8.3	7.0	1.9
8	BN $\oplus$ GMMD	11.7	9.1	1.9	10.2	7.0	3.2
9	BN with fMLLR $\oplus$ GMMD	12.0	**11.2**	4.0	11.4	9.0	5.2
10	BN $\oplus$ GMMD*	11.7	9.4	3.2	16.5	12.2	6.9
11	BN with fMLLR $\oplus$ GMMD*	**13.6**	11.0	**4.8**	**17.6**	**13.6**	**8.3**

Table 4: Relative WER reduction (WERR) for adapted BLSTM-CTC and TDNN AMs in comparison with the best SI AMs for each AM type (#2 for CTC and #3 for TDNN). Relative WERR values are calculated based on the results from Tables 2 and 3.

nism for improving the performance of an ASR system in the new end-to-end speech recognition paradigm. Experimental results on the TED-LIUM corpus showed that in an unsupervised adaptation mode, the adaptation and data augmentation techniques can provide approximately a 10–20% relative WERR on different adaptation sets, compared to the SI BLSTM-CTC system built on filter-bank features. The best results, for BLSTM-CTC AMs, in average, were obtained using GMM-derived features and MAP adaptation, which can be further slightly improved by combination with fMLLR adaptation technique.

We found out, that the type of the neural network AM architecture can differently influence the adaptation performance. The comparison with the TDNN-CE AMs showed that for these models, in contradiction to BLSTM-CTC AMs, MAP adaptation using GMMD features outperforms fMLLR only when it uses SAT model in the first decoding pass to obtain transcriptions for adaptation.

Also the obtained results allow us to compare TDNN-CE and BLSTM-CTC AMs in the realistic conditions, when the speaker adaptation is applied, which is important because usually end-to-end and hybrid AMs are compared on incomplete unadapted systems. The best SI TDNN-CE AM outperforms the best SI BLSTM-CTC AM on 1–7% of relative WER reduction for different test sets. For the best SAT AMs this gap in WER for TDNN-CE and BLSTM-CTC AMs increases and reaches 5–13% of relative WER reduction.

6. Bibliographical References

Audhkhasi, K., Ramabhadran, B., Saon, G., Picheny, M., and Nahamoo, D. (2017). Direct acoustics-to-word models for english conversational speech recognition. *arXiv preprint arXiv:1703.07754*.

Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. (2016). End-to-end attention-based large vocabulary speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 4945–4949. IEEE.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166.

Chan, W., Jaitly, N., Le, Q., and Vinyals, O. (2016). Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 4960–4964. IEEE.

Chorowski, J., Bahdanau, D., Cho, K., and Bengio, Y. (2014). End-to-end continuous speech recognition using attention-based recurrent nn: First results. *arXiv preprint arXiv:1412.1602*.

Collobert, R., Puhrsch, C., and Synnaeve, G. (2016). Wav2letter: an end-to-end convnet-based speech recognition system. *arXiv preprint arXiv:1609.03193*.

Dahl, G. E., Sainath, T. N., and Hinton, G. E. (2013). Improving deep neural networks for LVCSR using rectified linear units and dropout. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8609–8613, May.

Fritz, L. and Burshtein, D. (2017). End-to-end map training of a hybrid hmm-dnn model. *arXiv preprint arXiv:1703.10356*.

Gales, M. (1998). Maximum likelihood linear transformations for HMM-based speech recognition. *Computer speech and language*, 12(2):75–98.

Gauvain, J.-L. and Lee, C.-H. (1994). Maximum a posteriori estimation for multivariate gaussian mixture observations of Markov chains. *IEEE Trans. Speech and Audio Proc.*, 2:291–298.

Gemello, R., Mana, F., Scanzio, S., Laface, P., and De Mori, R. (2006). Adaptation of hybrid ANN/HMM models using linear hidden transformations and conservative training. In *Proc. ICASSP*, pages 1189–1192.

Graves, A. and Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, volume 14, pages 1764–1772.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM.

Graves, A., Mohamed, A.-R., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE.

Grézl, F., Karafiát, M., Kontár, S., and Cernocky, J. (2007).

Probabilistic and bottle-neck features for LVCSR of meetings. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–757. IEEE.

Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., et al. (2014). Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-R., Jaitly, et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huang, Z., Li, J., Siniscalchi, S. M., Chen, I.-F., Weng, C., and Lee, C.-H. (2014). Feature space maximum a posteriori linear regression for adaptation of deep neural networks. In *Proc. INTERSPEECH*, pages 2992–2996.

Huang, Z., Li, J., Siniscalchi, S. M., Chen, I.-F., Wu, J., and Lee, C.-H. (2015). Rapid adaptation for deep neural networks through multi-task learning. In *Proc. INTERSPEECH*, pages 2329–9290.

Huang, Z., Tang, J., Xue, S., and Dai, L. (2016). Speaker adaptation of RNN-BLSTM for speech recognition based on speaker code. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5305–5309. IEEE.

Lee, L. and Rose, R. C. (1996). Speaker normalization using efficient frequency warping procedures. In *Proc. ICASSP*, volume 1, pages 353–356. IEEE.

Li, J., Huang, J.-T., and Gong, Y. (2014). Factorized adaptation for deep neural network. In *Proc. ICASSP*, pages 5537–5541. IEEE.

Li, S., Lu, X., Akita, Y., and Kawahara, T. (2015). Ensemble speaker modeling using speaker adaptive training deep neural network for speaker adaptation. In *Proc. INTERSPEECH*, pages 2892–2896.

Liao, H. (2013). Speaker adaptation of context dependent deep neural networks. In *Proc. ICASSP*, pages 7947–7951. IEEE.

Liu, H., Zhu, Z., Li, X., and Satheesh, S. (2017). Gram-CTC: Automatic unit selection and target decomposition for sequence labelling. *arXiv preprint arXiv:1703.00096*.

Miao, Y., Gowayyed, M., and Metze, F. (2015). Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding. In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 167–174. IEEE.

Miao, Y., Gowayyed, M., Na, X., Ko, T., Metze, F., and Waibel, A. (2016). An empirical exploration of CTC acoustic models. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 2623–2627. IEEE.

Peddinti, V., Povey, D., and Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts. In *INTERSPEECH*, pages 3214–3218.

Pinto, J. P. and Hermansky, H. (2008). Combining evidence from a generative and a discriminative model in phoneme recognition. Technical report, IDIAP.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al. (2011). The Kaldi speech recognition toolkit. In *Proc. ASRU*.

Price, R., Iso, K., and Shinoda, K. (2014). Speaker adaptation of deep neural networks using a hierarchy of output layers. In *Proc. SLT*, pages 153–158. IEEE.

Rousseau, A., Deléglise, P., and Estève, Y. (2014). Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks. In *Proc. LREC*, pages 3935–3939.

Sak, H., Senior, A., Rao, K., Irsoy, O., Graves, A., Beaufays, F., and Schalkwyk, J. (2015). Learning acoustic frame labeling for speech recognition with recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4280–4284. IEEE.

Saon, G., Soltau, H., Nahamoo, D., and Picheny, M. (2013). Speaker adaptation of neural network acoustic models using i-vectors. In *ASRU*, pages 55–59.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Seide, F., Li, G., Chen, X., and Yu, D. (2011). Feature engineering in context-dependent deep neural networks for conversational speech transcription. In *Proc. ASRU*, pages 24–29. IEEE.

Senior, A. and Lopez-Moreno, I. (2014). Improving DNN speaker independence with i-vector inputs. In *Proc. ICASSP*, pages 225–229.

Siniscalchi, S. M., Li, J., and Lee, C.-H. (2013). Hermitian polynomial for speaker adaptation of connectionist speech recognition systems. *Audio, Speech, and Language Processing, IEEE Trans. on*, 21(10):2152–2161.

Swietojanski, P. and Renals, S. (2014). Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models. In *Proc. SLT*, pages 171–176. IEEE.

Tomashenko, N. and Khokhlov, Y. (2014). Speaker adaptation of context dependent deep neural networks based on MAP-adaptation and GMM-derived feature processing. In *Proc. INTERSPEECH*, pages 2997–3001.

Tomashenko, N. and Khokhlov, Y. (2015). GMM-derived features for effective unsupervised adaptation of deep neural network acoustic models. In *Proc. INTERSPEECH*, pages 2882–2886.

Tomashenko, N., Khokhlov, Y., and Esteve, Y. (2016a). On the use of Gaussian mixture model framework to improve speaker adaptation of deep neural network acoustic models. In *Proc. INTERSPEECH*, pages 3788–3792.

Tomashenko, N., Khokhlov, Y., and Esteve, Y. (2016b). A new perspective on combining GMM and DNN frameworks for speaker adaptation. In *Statistical Language and Speech Processing: 4th International Confer-*

ence, SLSP 2016, Pilsen, Czech Republic, October 11-12, 2016, Proceedings, volume 9918, pages 120–132. Springer.

Tomashenko, N., Khokhlov, Y., Larcher, A., and Estève, Y. (2016c). Exploring GMM-derived features for unsupervised adaptation of deep neural network acoustic models. In *International Conference on Speech and Computer*, pages 304–311. Springer.

Wang, Y., Deng, X., Pu, S., and Huang, Z. (2017). Residual convolutional CTC networks for automatic speech recognition. *arXiv preprint arXiv:1702.07793*.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

Xue, S., Abdel-Hamid, O., Jiang, H., Dai, L., and Liu, Q. (2014). Fast adaptation of deep neural network based on discriminant codes for speech recognition. *Audio, Speech, and Language Processing, IEEE/ACM Trans. on*, 22(12):1713–1725.

Yi, J., Ni, H., Wen, Z., Liu, B., and Tao, J. (2016). CTC regularized model adaptation for improving LSTM RNN based multi-accent mandarin speech recognition. In *Chinese Spoken Language Processing (ISCSLP), 2016 10th International Symposium on*, pages 1–5. IEEE.

Yu, D., Yao, K., Su, H., Li, G., and Seide, F. (2013). KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In *Proc. ICASSP*, pages 7893–7897.

Zhang, X., Trmal, J., Povey, D., and Khudanpur, S. (2014). Improving deep neural network acoustic models using generalized maxout networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 215–219. IEEE.

Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Bengio, C. L. Y., and Courville, A. (2017). Towards end-to-end speech recognition with deep convolutional neural networks. *arXiv preprint arXiv:1701.02720*.

Creating New Language and Voice Components for the Updated MaryTTS Text-to-Speech Synthesis Platform

Ingmar Steiner, Sébastien Le Maguer
Saarland University & DFKI GmbH
{steiner, slemaguer}@coli.uni-saarland.de

Abstract

We present a new workflow to create components for the MaryTTS text-to-speech synthesis platform, which is popular with researchers and developers, extending it to support new languages and custom synthetic voices. This workflow replaces the previous toolkit with an efficient, flexible process that leverages modern build automation and cloud-hosted infrastructure. Moreover, it is compatible with the updated MaryTTS architecture, enabling new features and state-of-the-art paradigms such as synthesis based on deep neural networks (DNNs). Like MaryTTS itself, the new tools are free, open source software (FOSS), and promote the use of open data.

Keywords: text-to-speech synthesis (TTS), front-end, multilingual

1. Introduction

Over the last 15 years, MaryTTS (Schröder and Trouvain, 2001) has become one of the reference systems for open source text-to-speech synthesis (TTS). Today, it is actively used by researchers working in speech science, human-computer interaction (HCI), and related fields, as well as by professional and enthusiast software developers in free, open source software (FOSS) or enterprise settings. Its popularity is due in part to the number of languages and voices which are freely available as open resources, as well as the possibility of extending it to support new languages and building custom synthetic voices, or even integrating MaryTTS as a component into more complex applications, such as TTS web services, accessibility software, or spoken dialog systems (SDSs). Because of its implementation in the Java programming language, MaryTTS can be used on any device or computer with a Java Runtime Environment (JRE), and its modular design allows developers and users alike to inspect and customize the entire processing pipeline from input text to speech output.

However, the number of people who have participated in, and contributed to, MaryTTS development over the years has led to a complex and overburdened system. Consequently, a reboot of the system became unavoidable; until now, we focused on restructuring the system core and explained the philosophy behind the new architecture (Le Maguer and Steiner, 2017a; Le Maguer and Steiner, 2017b).

Independently, the process of creating new synthetic voices and support for new languages in MaryTTS has also fundamentally evolved since it was presented by Pammi et al. (2010). Therefore, the current paper presents the new language and voice building workflow for MaryTTS.

The remainder of the paper is structured as follows. Section 2 provides a brief background on build automation in MaryTTS. In Section 3, we present the new workflow to add support for a new language. Then, in Section 4, we focus on the new voice building pipeline. Finally, in Section 5, we present the reorganized source code and project hosting, particularly from a user perspective.

2. Background

Development on MaryTTS has adopted several significant paradigms which had become best practice in Java-based software engineering in the years since the project's inception. These include,

dependency management, where required software libraries are downloaded from cloud-based repositories,[1]

software testing, and

convention over configuration, where common standards are integrated into the software build lifecycle without the need for redundant specification.

In the latest version of MaryTTS, all of these aspects are managed through the *Gradle* build automation tool.[2]

The increase in flexibility and efficiency provided by Gradle is not limited to the development "under the hood". Rather, we leverage Gradle as a user-facing tool which replaces the custom applications previously required to add new languages to MaryTTS, or build new synthetic voices. This shift removes numerous limitations on performance and functionality, and solves common, recurring problems with installing third-party tools and writing boilerplate code for new MaryTTS components. At the same time, the text and speech data itself — required to build new components — can be managed as dependencies, and the components can be built, tested, and distributed more efficiently.

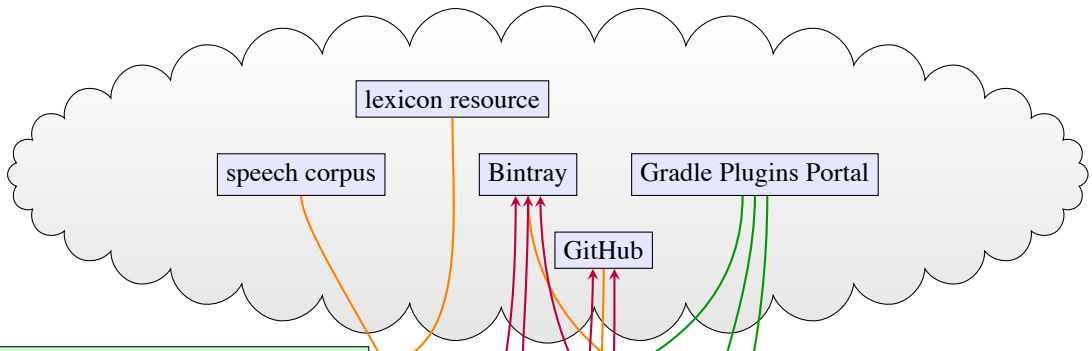
An overview of the entire workflow to create new language and synthetic voice components is shown in Figure 1. However, this workflow can be broken up into several independent steps, which are described in the following sections.

3. New Language Support

The purpose of a language component in MaryTTS is to allow the system to extract linguistic features from orthographic text using natural language processing (NLP). This includes, at the very least, the sequence of phonemes, i.e., the pronunciation, but typically also other features related to

[1] Examples of such dependencies in MaryTTS include third-party libraries for text tokenization (JTok), number expansion (ICU4J), and part-of-speech (POS) tagging (OpenNLP).

[2] https://gradle.org/



obtains lexicon resource
applies `lexicon-compiler-plugin`
trains lexicon FST and G2P rules
publishes `marytts-lexicon-xy`

(a) The lexicon project, which resolves or contains lexicon resources and is published to Bintray.

depends on `marytts-lexicon-xy`
provides NLP modules for language "xy"
publishes `marytts-lang-xy`

(b) The language project depends on the lexicon project, may contain other modules, and is published to Bintray.

obtains audio and text resources
obtains phonetic annotation *or*
applies `kaldi-mfa-plugin`
publishes
`speaker-somename-xy-data`

(c) The data project resolves audio and text from a speech corpus and either converts provided phonetic annotation or runs forced alignment using the language project. It can be published to GitHub as a release asset.

depends on `marytts-lang-xy` and
`speaker-somename-xy-data`
applies `voicebuilding-plugin`
1. extracts acoustic features from audio
2. extracts linguistic features from text
3. aligns features based on phonetic labels
4. builds models
5. packages data (for unit selection)

publishes `voice-somename-xy`

(d) The voice project resolves the processed data from the cloud (e.g., downloading from GitHub), runs all steps required to build a voice, and is published to Bintray. Large unit selection voice packages can be published to GitHub as release assets.

Figure 1: Overview of the complete workflow for a new language "xy" and synthetic voice components. Dashed blue arrows visualize the dependency of the voice project (Figure 1d) on the language and data projects (Figures 1b and 1c, respectively), the dependency of the data project on the language project, and the language project on the lexicon project (Figure 1a). All of these depend on the core MaryTTS runtime libraries (not shown), which are resolved from Bintray. Orange and purple arrows show the actual dependency resolution from, and publishing to, cloud-hosted services, respectively. Green arrows show plugins resolved from the Gradle Plugins Portal. Note that all or part of the cloud-hosted infrastructure (shown inside the cloud) could also be replaced by internal, non-public repositories.

phonology and used for the prediction of acoustic parameters, such as segment duration and fundamental frequency (F_0). Pronunciation prediction in MaryTTS is handled by a language-specific "Phonemiser" module, which looks up each text token in a lexicon and returns the sequence of phonemes. For any out-of-vocabulary (OOV) tokens, the module falls back to rules for grapheme-to-phoneme (G2P) prediction.

To add support for a new language to MaryTTS, the first step is to define the set of phonemes to be used, along with their standard phonological features, based on the International Phonetic Alphabet (IPA). The next step is to obtain (or create) a lexicon resource, ideally a text file, spreadsheet, etc., containing a list of words with their orthographic and corresponding phonetic transcription.

Finally, the lexicon is automatically compiled into a finite state transducer (FST)-based representation, relying in part on the WEKA toolkit (Hall et al., 2009). In the past, this was done using the custom *TranscriptionTool* GUI application (Pammi et al., 2010), which however suffers from various usability and performance issues. To improve this situation, we have developed a Gradle plugin[3] to convert the lexicon into the format required by MaryTTS. Furthermore, we are currently developing a more state-of-the-art G2P approach based on TensorFlow (Abadi et al., 2016), comparable to that of, e.g., van Esch et al. (2016).

It is possible to create further NLP modules for the new language component, handling text normalization to expand acronyms, numbers, and so on, into pronounceable repre-

[3]https://github.com/marytts/gradle-marytts-lexicon-compiler-plugin

sentations, POS tagging, etc. Alternatively, MaryTTS can just fall back to generic modules for such tasks. All of these modules are then combined to build the new language component, which will be used to process input text, and represents a dependency of the synthetic voice building, and ultimately, full TTS in the new language.

4. Voice Building

Building a new synthesis voice for MaryTTS consists of three distinct stages, (a) data preparation, (b) feature extraction, and (c) model building, which are described in the following subsections. All three stages are handled efficiently using Gradle plugins,[4] which wrap third-party tools and can run tasks in parallel where appropriate, speeding up the voice building process significantly compared to the old toolkit.

4.1. Data Preparation

When preparing the recording of speech data intended to create a new synthesis voice, it is common practice to create a prompt list which covers the phonetic (and possibly prosodic) inventory of the corresponding language, as well as the content of the voice's domain. These prompts are then read out by the voice talent over one or more recording sessions, preferably in a studio environment.

The previous voice creation toolkit (Pammi et al., 2010) promoted the use of a custom Java-based recording tool named *Redstart*, which is able to display a sequence of prompts on a computer screen and record the user reading them through the computer's microphone. While MaryTTS Redstart remains fully functional, it may not be usable in every recording scenario. For instance, in a professional recording studio, the voice talent is typically recorded using a digital audio workstation (DAW), and any visual presentation of prompts may only be possible using a separate computer. In other cases, the goal may be to record a more fluent performance (such as an audiobook), and a user experience that forces the voice talent to pause for each prompt would be too disruptive.

Regardless of which text prompts are selected, or how they are recorded, the outcome of this process is a set of text and audio files with corresponding contents. However, before these files can be used to build a synthetic voice for MaryTTS, they have to be phonetically annotated. This step requires determining the pronunciation of each text prompt, i.e., the sequence of phonetic units, and mapping them to the recorded audio's time domain; the process is related to automatic speech recognition (ASR), except that the expected content is known, and the sequence of phonetic units can be forced to align with the audio; this is known as *forced alignment*. In the past, the MaryTTS voice building tools relied on integrating third-party tools for this task, including *Sphinx-4* (Walker et al., 2004), *HTK* (Young et al., 2006), or the *FestVox* tool *EHMM* (Prahallad et al., 2006); however, MaryTTS users often report problems installing or running them, and errors are difficult to solve. More recently, *Kaldi* (Povey et al., 2011) has emerged as a leading ASR toolkit, and it has been integrated into the *Montreal Forced Aligner* tool (McAuliffe et al., 2017). This tool in turn has been integrated into the MaryTTS data preparation workflow in the form of a Gradle plugin.[5] The pronunciation can be predicted using MaryTTS and collected into a custom dictionary for Kaldi, then acoustic models are trained from the recorded data, and the phonetic unit boundaries are aligned and stored in the form of Praat TextGrids; this process is fully automated and can take a few minutes or hours, depending on the amount of recorded data.

Previously, the forced alignment process was described as part of the voice building process in MaryTTS (Pammi et al., 2010), but it can be more appropriately regarded as a prerequisite. While it is still possible to use both the forced alignment and voice building plugins in the same Gradle project, a more efficient workflow is to build a data *artifact*, which is then available as a dependency for the proper voice building process. Therefore, this stage can be skipped if a corpus of speech data is already available with appropriate orthographic and phonetic annotations.

4.2. Feature Extraction

At the core of the voice building process, the recorded speech data is converted to a *feature representation*. It is this feature representation which allows the use of machine learning techniques to train models to predict prosody and/or vocoder parameters from text during the actual TTS process in the runtime system.

The feature extraction stage of the voice building process yields a combination of frame-wise feature vectors from acoustic analysis of the audio, and time-aligned symbolic features based on linguistic analysis of the corresponding text; the alignment is based on the phonetic annotation obtained in the data preparation (cf. Section 4.1).

Acoustic features include F_0, tracked using Praat (Boersma, 2001), and mel-frequency cepstral coefficients (MFCCs), extracted using the Edinburgh Speech Tools (EST) (King et al., 2003). The linguistic features are obtained depending on the MaryTTS language component for the corresponding language. When creating a synthetic voice for a new language, this is where the new language component built previously (cf. Section 3) is used. The linguistic features extracted and assigned to the feature vectors include several related to phonology (e.g., distinctive features, position in the syllable, stress, accent), syntax (e.g., POS, distance to phrase and sentence boundaries), and — optionally — speaking style (Steiner et al., 2010; Charfuelan and Steiner, 2013), information density (Le Maguer et al., 2016), or other high-level context features.

4.3. Model Building

Depending on the underlying synthesis paradigm, it is possible to build a *unit selection* voice or a *statistical parametric synthesis* voice.

4.3.1. Unit Selection

Unit selection synthesis concatenates halfphone-sized snippets of natural speech selected from a voice database, given

[4] https://github.com/marytts/gradle-marytts-voicebuilding-plugin

[5] https://github.com/marytts/gradle-marytts-kaldi-mfa-plugin

target features computed for an input utterance. The output can sound very natural, but often suffers from audible glitches when synthesizing out-of-domain utterances, and prosody control is limited. Moreover, the voice database can be very large, as it contains the actual audio data.

Building a unit selection voice for MaryTTS involves storing the feature representation and related metadata for each unit, training statistical models for sparse prosody prediction, and packaging these along with the actual audio data. We have created a Gradle plugin which wraps some of the old toolkit's components to assemble unit selection voices which are backward-compatible with the current stable release of MaryTTS (v5.2). In addition, we are developing new build tools to support audio compression and enable prosody modeling and target feature prediction using hidden Markov models (HMMs) or deep neural networks (DNNs), paving the way for state-of-the-art "hybrid" TTS.

4.3.2. Statistical Parametric Synthesis

MaryTTS has supported statistical parametric synthesis for numerous years, using a Java port of the HMM based speech synthesis system (HTS) engine API[6] with a mel-generalized log spectrum approximation (MLSA) vocoder. Although such synthesis can sound rather buzzy and unnatural, these HMM-based voices offer higher flexibility and more consistent quality than unit-selection synthesis, as well as a much smaller memory footprint. However, some drawbacks are (a) that building HMM-based voices for MaryTTS has a high technical overhead, and (b) that the Java port has become quite outdated, while HTS development has seen significant progress. The former has been mitigated by providing a consistent, pre-configured *Docker* container, while to address the latter, we are developing completely new functionality. This includes the possibility to train models for third-party frameworks such as *Merlin* (Wu et al., 2016) and to allow other vocoders to be used, including *STRAIGHT* (Kawahara et al., 1999) or *WORLD* (Morise et al., 2016).

The parametric voice building process comprises three stages: the input and output feature packing, the model training, and the voice configuration generation. The voice configuration generation is similar to the unit selection voice building part (cf. Section 4.3.1). The output feature packing goal is just to adapt the acoustic features (e.g., mel-generalized cepstrum (MGC), F_0, band aperiodicity (BAP), etc.) to be compatible with the process used to train the models. Currently this means computing the delta and delta-delta coefficients and generating the binary observation vector for each utterance. The input feature packing consists of calling MaryTTS with a serializer dedicated to the training process.

The model training is a specific plugin implementing the process to train the models needed for the synthesis stage. We have developed a Gradle plugin dedicated to train HTS models (HMM-GMM or HMM-DNN).[7] This plugin can be adapted to the kind of parametric synthesis model or system we want to use.

[6] http://hts-engine.sourceforge.net/

[7] https://github.com/marytts/gradle-hts-voicebuilding-plugin

4.4. New Configuration Mechanism

Previously a configuration was attached to an artifact to configure the different modules. Moving forward, we consider three levels of configuration: the default configuration, the voice configuration, and the user configuration. The first of these is given in the module itself. The voice configuration corresponds to the parametrization of each module used during the voice building process and has priority over the default configuration. Finally, a user configuration can be specified at runtime, to override the other configurations.

5. Global Project Management

Refactoring the core system and of the voice building process has allowed us to separate the source code management (SCM) for each language and each voice project. Therefore, each language and voice can have its own SCM repository hosted on GitHub,[8] while the released artifacts are published to Bintray[9] and indexed in JCenter.[10] Any large data objects (specifically unit selection audio data) can be hosted on GitHub as release assets.

This makes the custom *Component Installer* GUI from previous MaryTTS versions obsolete, and allows us to replace it with a lightweight wrapper around the dependency management. A user can install and run MaryTTS voices and language components simply by executing Gradle tasks with the corresponding names; this is demonstrated by a new web installer for MaryTTS.[11]

Meanwhile, developers and researchers looking to integrate MaryTTS into their projects, only need to declare a dependency on the desired voice artifacts, and this will automatically resolve all transitive dependencies on the corresponding languages and other libraries.

6. Conclusion

In conclusion, we have presented a new language and voice building workflow designed for the updated MaryTTS system. We have detailed our reliance on the Gradle build automation tool, which provides a much more efficient and powerful framework via its extensible plugin system than the previous toolkit. We have also seen that the language components maintain the same concepts as in previous versions, but the methodologies used are updated. Finally, we have described the redesigned and extended voice building process, as well as our leverage of cloud-based infrastructure for hosting and distribution.

The next stage is to integrate the new MaryTTS core, state-of-the-art synthesis paradigms, and the new build system more deeply to provide the fully modular, modern TTS platform we are aiming for. Moreover, we are working to release the first preview of MaryTTS v6.0 in the coming months.

7. Acknowledgements

This work was funded by the German Research Foundation (DFG) under grants EXC 284 and SFB 1102.

[8] e.g., https://github.com/marytts/voice-dfki-spike

[9] https://bintray.com/marytts/marytts

[10] https://bintray.com/bintray/jcenter

[11] https://github.com/marytts/marytts-installer

8. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 265–283, Savannah, GA, USA.

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345.

Charfuelan, M. and Steiner, I. (2013). Expressive speech synthesis in MARY TTS using audiobook data and EmotionML. In *Interspeech*, pages 1564–1568, Lyon, France.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations*, 11(1):10–18.

Kawahara, H., Masuda-Katsuse, I., and de Cheveigné, A. (1999). Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds. *Speech Communication*, 27(3-4):187–207.

King, S., Black, A. W., Taylor, P., Caley, R., and Clark, R., (2003). *Edinburgh Speech Tools Library*. Centre for Speech Technology, University of Edinburgh.

Le Maguer, S. and Steiner, I. (2017a). The 'uprooted' MaryTTS entry for the Blizzard Challenge 2017. In *Blizzard Challenge*, Stockholm, Sweden.

Le Maguer, S. and Steiner, I. (2017b). Uprooting MaryTTS: Agile processing and voicebuilding. In *28th Conference on Electronic Speech Signal Processing (ESSV)*, pages 152–159, Saarbrücken, Germany.

Le Maguer, S., Möbius, B., and Steiner, I. (2016). Toward the use of information density based descriptive features in HMM based speech synthesis. In *8th International Conference on Speech Prosody*, pages 1029–1033, Boston, MA, USA.

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., and Sonderegger, M. (2017). Montreal Forced Aligner: Trainable text-speech alignment using Kaldi. In *Interspeech*, pages 498–502, Stockholm, Sweden.

Morise, M., Yokomori, F., and Ozawa, K. (2016). WORLD: A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems*, E99-D(7):1877–1884.

Pammi, S., Charfuelan, M., and Schröder, M. (2010). Multilingual voice creation toolkit for the MARY TTS platform. In *7th International Conference on Language Resources and Evaluation (LREC)*, pages 3750–3756, Valletta, Malta.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlíček, P., Qian, Y., Schwarz, P., Silovský, J., Stemmer, G., and Veselý, K. (2011). The Kaldi speech recognition toolkit. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Waikoloa, HI, USA.

Prahallad, K., Black, A. W., and Mosur, R. (2006). Subphonetic modeling for capturing pronunciation variations for conversational speech synthesis. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume I, pages 853–856, Toulouse, France.

Schröder, M. and Trouvain, J. (2001). The German text-to-speech synthesis system MARY: A tool for research, development and teaching. In *4th Speech Synthesis Workshop (SSW)*, Perthshire, Scotland.

Steiner, I., Schröder, M., Charfuelan, M., and Klepp, A. (2010). Symbolic vs. acoustics-based style control for expressive unit selection. In *7th Speech Synthesis Workshop (SSW)*, pages 114–119, Kyoto, Japan.

van Esch, D., Chua, M., and Rao, K. (2016). Predicting pronunciations with syllabification and stress with recurrent neural networks. In *Interspeech*, pages 2841–2845, San Francisco, CA, USA.

Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., Wolf, P., and Woelfel, J. (2004). Sphinx-4: A flexible open source framework for speech recognition. Technical Report TR-2004-139, Sun Microsystems.

Wu, Z., Watts, O., and King, S. (2016). Merlin: An open source neural network speech synthesis system. In *9th Speech Synthesis Workshop (SSW)*, pages 218–223, Sunnyvale, CA, USA.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X. A., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P., (2006). *The HTK Book (for HTK Version 3.4)*. Cambridge University Engineering Department.

Speech Rate Calculations with Short Utterances: A Study from a Speech-to-Speech, Machine Translation Mediated Map Task

Hayakawa Akira, [1] **Carl Vogel,** [1] **Saturnino Luz,** [2] **Nick Campbell** [1]
[1] School of Computer Science and Statistics, Trinity College Dublin, Ireland
[2] Usher Institute of Population Health Sciences & Informatics, University of Edinburgh, UK
campbeak@tcd.ie, vogel@cs.tcd.ie, S.Luz@ed.ac.uk, nick@tcd.ie

Abstract

The motivation for this paper is to present a way to verify if an utterance within a corpus is pronounced at a fast or slow pace. An alternative method to the well-known Word-Per-Minute (wpm) method for cases where this approach is not applicable. For long segmentations, such as the full introduction section of a speech or presentation, the measurement of wpm is a viable option. For short comparisons of the same single word or multiple syllables, Syllables-Per-Second (sps) is also a viable option. However, when there are multiple short utterances that are frequent in task oriented dialogues or natural free flowing conversation, such as those of the direct Human-to-Human dialogues of the HCRC Map Task corpus or the computer mediated inter-lingual dialogues of the ILMT-s2s corpus, it becomes difficult to obtain a meaningful value for the utterance speech rate. In this paper we explain the method used to provide a alternative speech rate value to the utterance of the ILMT-s2s corpus and the HCRC Map Task corpus.

Keywords: speech rate, utterance duration comparison, task oriented dialogues

1. Introduction

Computer mediated communication is becoming more frequent. The next step in this new communication style, is Inter-Lingual Computer Mediated Communication. Recently, Microsoft released the Skype Translator (Lewis, 2015) that translates up to 10 languages in Speech-to-Speech Machine Translation. The Japanese Ministry of Internal Affairs and Communication has announced that the Tokyo Olympics in 2020 are to employ information systems that use multilingual machine mediated communication in Speech-to-Speech (S2S) form for 17 languages, using the VoiceTra system (Matsuda et al., 2013). Computer mediated multi-lingual communication will create a conversation style that few users have adequate exposure (Hara and Iqbal, 2015) and may have difficulty adapting to. For example, using the speech rate calculation method explained in this paper, gender differences in the adaptation speech rate have been identified (Hayakawa et al., 2017b) but with little to no improvement to the automatic speech recognition results. With users of computer mediated interaction, the "Speakers are unlikely to have a good model of what computers are likely to know, presumably in part because natural language systems tend to be tailored to particular and quite specific uses." (Branigan et al., 2010, p. 2366). Being able to provide feedback to the user about the changes to their speech rate, should help prevent the hyper-articulation reported by Hayakawa et al. (2017b).

The method in this paper is not new in theory. Sztahó et al. (2015) use a similar method of comparing the syllable duration it takes a subject to utter a specific phrase with previous recordings of the utterance by the same subject. The method explained in this paper has also been used to calculate the speech rate in previous publications by the author (Hayakawa et al., 2015; Hayakawa et al., 2016a; Hayakawa et al., 2017a), but continues to attract interest and has yet to be explained in detail.

2. The Dataset

Two corpora have been used in this analysis, the ILMT-s2s corpus (Hayakawa et al., 2016b) and the HCRC Edinburgh Map Task corpus (Anderson et al., 1991).

The ILMT-s2s corpus: The corpus consists of 15 dialogues of English speakers communicating with Portuguese speakers to perform the HCRC Edinburgh Map Task (Anderson et al., 1991), a task where the subject is to guide the interlocutor along a predefined route on the map of one of the subjects. The subjects are situated in different rooms and communicate in their mother tongue to their interlocutor using a Speech-to-Speech Machine Translation (S2S-MT) system that Hayakawa et al. (2016b) call the *ILMT-s2s System*. The corpus consists of ≈9.5 hours of audio, video and biological signal recordings of interlingual system mediated communication of 15 subject pairs (15 English and 15 Portuguese speakers).

The HCRC Edinburgh Map Task corpus: The corpus consists of 128 English dialogues of direct human-to-human task based interactions using the map task technique to elicit the comunication. The recordings were split in two settings, with half the subjects being able to see their interlocutor's face (i.e., with eye-contact), while the other half had screens placed between them (i.e., without eye-contact). To standardise the data, only dialogues that used the same maps (maps 1 & 7) as those used in the ILMT-s2s corpus were kept for this study, resulting in a total of only 16 dialogues out of the 128 (half male, half female).

3. Calculating the Speech Rate

To calculate if any given utterance in the ILMT-s2s corpus was uttered quickly or slowly, the duration of the subject utterance was compared with that of the same utterance repeated by the TTS system used in the ILMT-s2s System during the collection of the corpus.

This method was chosen for two reasons. The first, was because we wanted to verify if the subjects were aligning their speech rate to the ILMT-s2s System output. This would

result in the subject utterance aligning with the 180 wpm speech rate setting of the ILMT-s2s System's TTS output. The second reason was because the utterances in the ILMT-s2s corpus were expected to be low in word count and short in duration from the pre-corpus collection test runs, and hence, make it difficult to get a meaningfull speech rate value. This expectation was materialised in the ILMT-s2s corpus collection with the median word count per utterances of 4 words and a median duration per utterances of 1.50 seconds as indicated in Table 1.

Word Count	*Mdn*	*M*	*SD*	Count
ILMT-s2s All Subjects	4	5.168	6.01	3,628
ILMT-s2s English Subjects	4	4.919	6.76	1,980
ILMT-s2s Portuguese Subjects	4	5.466	4.97	1,648
Duration (sec.)	*Mdn*	*M*	*SD*	Count
ILMT-s2s All Subjects	1.493	2.244	2.75	3,628
ILMT-s2s English Subjects	1.285	1.939	2.96	1,980
ILMT-s2s Portuguese Subjects	1.874	2.610	2.49	1,648

Table 1: Summary of word count and duration in corpus

A summary of the rate that each utterance in the ILMT-s2s corpus was uttered is reported in Table 2 using the measure of wpm. The utterances are indicated to have been

Speech Rate (wpm)	Min.	*Mdn*	*M*	Max.	*SD*
ILMT-s2s All Subjects	15.24	141.00	153.40	618.60	65.28
ILMT-s2s En. Subjects	16.61	155.30	167.70	618.60	68.75
ILMT-s2s Pt. Subjects	15.24	127.90	136.20	459.60	56.26

Table 2: Summary of wpm per utterance

spoken as slowly as 15.40 wpm and as fast as 618.60 wpm with an arithmetic mean speed of 153.4 wpm and a *SD* of 65.28. Of this range, 60% of the utterances were reported to have been uttered within a 100 wpm scope between 100.03 wpm (20^{th} percentile) and 200.00 wpm (80^{th} percentile). Though these wpm values indicate that the subject uttered their speech relatively slowly, it is not clear yet if the sample utterance was intentionally spoken slowly or fast, or if the wpm calculation method is providing numbers that only represent slow or fast speech rates.

If the orthodox way of calculating the Words Per Minute (wpm) as $(W/T \times 60)$, where W is the word count per utterance and T is the duration of the utterance in seconds was used, the imbalanced nature of the spoken words would create a variance that is presumably difficult to interpret. One example of this is the different wpm value of the three single word utterances, "Perfect", "Yes" and "Okay" that are indicated in Table 3. The maximum wpm value for the

Speech Rate (wpm)	Min.	1st	*Mdn*	*M*	3rd	Max.
Perfect	98.2	104.0	110.5	113.4	123.7	130.4
Yes	73.1	117.9	138.9	135.8	153.1	183.5
Okay	55.5	112.2	139.9	154.0	173.9	618.6
Pebbled shore	98.2	100.1	101.7	106.9	111.2	120.7
Go down	98.4	121.6	129.6	124.5	133.1	136.1
Then where?	206.9	213.1	219.4	216.7	221.6	223.9

Table 3: Summary of single and two word subject utterance duration and wpm sample

word "Perfect" is 130.40 wpm, but did the subject utter this word at the same intentional rate as the median wpm value for "Yes" at 138.90 wpm or "Okay" at 139.90 wpm? Unlike measurements for traveling distance speeds of kilometres per hour (kph) or miles per hour (mph) where the 2 items of distance and time are consistent, the length of distance that is measured as kilometres or miles, and the duration of time that is measured as hours, the measurement of wpm only has 1 consistent item, the durations that is measured in minutes. The measurement of a word is not of a consistent length — one might say that this variable length in words is a characteristic of the "stress-timed" English and Portuguese languages, but the variability in duration has also been found in a theoretically isochronous "syllable-timed" language such as Japanese, too (Arai and Greenberg, 1997). Since the length of the single word is different, it is not possible to know if the 130.40 wpm value for the single word utterance of "Perfect" is fast or slow when compared with the given example of "Yes" or "Okay", as it is possible with kph/mph where the numerator as well as the denominator are of a consistent measurement. This is not just a problem of single word utterances, but also in multiple word utterances as indicated in Table 3 with the different wpm values of the three multi word utterances. The median wpm values for "Pebbled shore" (an item on the map), "Go down" and "Then where?" are 101.70 wpm, 129.60 wpm and 219.40 wpm respectively. However there is no way of knowing if the difference in the wpm values of these three multi word utterances and even the single word utterances in Table 3 are the result of a speech rate difference or the difference in the duration to pronounce the given word. The bar chart

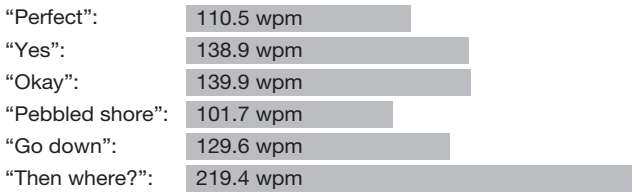


Figure 1: *Mdn* wpm for the six samples — Subject

in Figure 1 provides no more information than five words were spoken slowly and one was not.

The concept of wpm as a reference of speech rate is already an average of word combinations within the utterance of a minute. The random mixture of words with long and short duration are mixed into an utterance that is then calculated into a quantitive measure. Though the frequency of word duration has been previously investigated in written language (Zipf, 1945) and also in spoken language (Greenberg, 1999; Batliner et al., 2001; Bell et al., 2002; Bell et al., 2009), given the varying short sentences, in the unscripted speech of the ILMT-s2s corpus, the irregular duration of a word used in the short phrases would not provide an accurate wpm value to provide a reliable quantitive measure. If the window were a longer period of time, such as the first quarter of the dialogue, there might be enough words to balance out the variance, but not in this situation where the window is a single utterance. The other method of using *syllables per second* as a measure may be more reliable than wpm, however this would require highly trained

annotators to phonologically segment the data. With this comes a new issue of whether the utterance is annotated as indicated in the dictionary or following the actual phonetic sound that was uttered.[1] A fully unsupervised method of calculating the speech rate estimation based on syllable nuclei detection was presented by de Jong and Wempe (2009) as a Praat (Boersma and van Heuven, 2001) script, though we have yet to test this method on the data of the ILMT-s2s corpus to verify whether similar reliability can be obtained. However, analysis of the *syllables per second* may not provide any better results since an analysis of the Switchboard corpus data (Godfrey et al., 1992), which is a corpus of a collection of natural and spontaneous telephone conversations, by Greenberg (1999, p. 167) indicated the following:

> Although only 22% of the Switchboard lexicon is composed of monosyllabic forms, approximately 80% of the corpus tokens are just one syllable in length [...]. The portion of the lexicon consisting of three or more syllables (38%) is rarely exhibited in spontaneous language, accounting for less than 5% of the spoken instances [...].

This would indicate that even using the measure of syllables per second, the length of the dialogues in the ILMT-s2s corpus may be too short and the problem identified with the wpm measure may also continue to persist.

Since the main objectives of the studies with the ILMT-s2s corpus were to investigate the alignment or adaptation of the subject to the ILMT-s2s System, it was perceived that a comparison with the ILMT-s2s System's TTS speech rate would be adequate as it would provide a one-to-one reference point with the utterances of the ILMT-s2s corpus. The rationale behind this idea was to recreate every utterance that was uttered by the subjects of the ILMT-s2s corpus using the same TTS output system used in the ILMT-s2s System and compare like-for-like utterance durations between the human subject and the ILMT-s2s System's TTS voice. This method would theoretically remove the variance in the resulting speech rate value that is created by the differing duration in pronouncing words of differing lengths, since the reference will also be pronouncing the same word — therefore, creating a more stable speech rate. To compare the speech rate, the transcription text of all speakers was output to a plain text file that was then read out by the TTS system at a speech rate setting of 180 wpm and saved as an audio file. Using Praat, the audio file was segmented manually by the first author using the transcription from the plain text file to provide the boundaries of TTS output of each utterance. Once completed, the duration of each TTS output reference was calculated from the start and end times of the segmented audio files. A summary of the duration and wpm of each TTS output is as reported in Table 4.

4. Analysis

As with the subject utterances wpm values in Table 1, the TTS output data show a big variance between the slow output measures and the fast output measures, but with a smaller *SD* when compared to the subject utterance measures. The TTS output are as slow as 45.09 wpm and as fast as 536.10 wpm with an arithmetic mean speed of 185.50 wpm and a *SD* of 56.79. Of this range, 60% of the TTS output were reported to have been within a ≈100 wpm scope between 137.57 wpm (20th percentile) and 231.69 wpm (80th percentile) which is similar to the subject utterance range reported earlier. This summary

Duration (sec.)	Min.	*Mdn*	*M*	Max.	*SD*
ILMT-s2s All Subjects	0.112	1.250	1.694	56.500	1.90
ILMT-s2s En. Subjects	0.166	1.124	1.508	56.500	2.03
ILMT-s2s Pt. Subjects	0.112	1.491	1.917	14.240	1.69
Speech Rate (wpm)	Min.	*Mdn*	*M*	Max.	*SD*
ILMT-s2s All Subjects	45.09	179.40	185.50	536.10	56.79
ILMT-s2s En. Subjects	45.82	190.90	192.90	390.20	58.50
ILMT-s2s Pt. Subjects	45.09	170.10	176.60	536.10	53.33

Table 4: Summary of duration and wpm of the subject utterances output by the TTS system

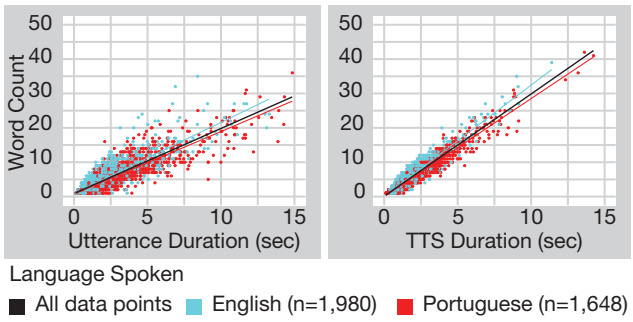


Figure 2: Scatter plot of ILMT-s2s corpus utterance durations by word count

in itself does not show much, but a simple linear regression model of the two wpm values highlights and confirms the differences between the two speech rates (Figure 2). The regression equation of the word count and the duration for the subject utterance is "$Word\ Count = 0.61648 + 2.02693\ Utterance\ duration$", with an $Adjusted\ R^2$ value of 0.858 ($p < 2.2e-16$) and for the TTS output it is "$Word\ Count = -0.06870 + 3.08959\ TTS\ output\ duration$", with an $Adjusted\ R^2$ value of 0.948 ($p < 2.2e-16$). There are two interesting details from these results that illustrate the data: (i) the slope difference of 1.062 which predicts that the speech rate of the TTS output will be ≈ 30% faster than the subject utterance, and (ii) the 0.948 $Adj\ R^2$ value that indicates there is a variance in the speech rate of the TTS output since it is not the maximum value of 1.0, but from the 0.09 difference, this variance is smaller than that of the subject utterances. These two points are retrievable from the summary of the data in Table 2, and Table 4, however the linear regression equation and Figure 2 represents this more clearly. Although this identifies the speech rate difference and variation difference of the two measurements of subject utterance speed and TTS output speed, it does not help clarify the speech rate differences of the six single and multiple word examples provided in Table 3. The main difference

[1] Greenberg (1999, p. 163) reported eighty different pronunciations of the word "and" in a study of the Switchboard Transcription Corpus (Greenberg, 1997)

that can be observed when comparing the one and two word example phrases of the subject utterances and TTS output, is again the reduction in the variability of the speech rate, the *SD* values (Table 4). This reduction indicates the regularity in which the phrase is repeatedly output by the TTS system. Which is an effect that is completely expected since this is the exact reason that the TTS output was used; to provide a standardised, regular speech rate. However, the question of which phrase is spoken fast or slow is not identifiable from this data. The bar chart in Figure 3 does not provide any better understanding of the speech rate of short utterances.

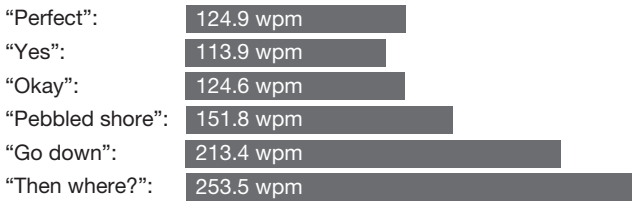


Figure 3: *Mdn* wpm for the six samples — TTS output

Taking advantage of the regular speech rate of the TTS output, the TTS output duration can be used as a reference that the subject utterance is compared against. By dividing the subject utterance duration by the TTS output duration, a value that identifies the difference from the regular speech rate obtained by the TTS output can be retrieved with an equation of (S/T), where S is the duration of the speaker's utterance and T is the duration of the TTS output. The resulting values of this measurement method are reported in Table 5. The range between zero

Speech Rate	Min.	1st Qu.	*Mdn*	*M*	3rd Qu.	Max.
ILMT-s2s All	0.167	0.982	1.233	1.379	1.588	16.400
ILMT-s2s En	0.167	0.925	1.188	1.287	1.525	9.759
ILMT-s2s Pt	0.414	1.044	1.291	1.490	1.670	16.400

Table 5: *Summary of subject utterance duration divided by TTS output duration — "S/T"*

(0) and one (1) are values from utterances that were spoken faster than the TTS output and the range of one (1) and above are utterances that were spoken slower than the TTS output. The *Mdn* value being slower than the TTS output can also be confirmed from the slope of the regression equation of "$Subject\ utterance\ duration = 0.239752 + 0.648037\ TTS\ output\ duration$" with an $Adjusted\ R^2$ value of 0.883 ($p < 2.2e-16$) as indicated in the graph on the right in Figure 4. With the slope being smaller than one (1), the regression equation is indicating that the subject utterance duration will not be shorter than the TTS output duration, and therefore the majority of subject utterance will be slower than the TTS output. This is evident from the lack of values within the pink coloured region of Figure 4, which covers the area of a slope greater than one (1). Unfortunately understanding the figures of Table 5 requires a moment of mental arithmetic and it is not immediately identifiable from just a glance at the results if the utterance was spoken quickly or not. To make the measurement values immediately evident, one (1) was subtracted from the

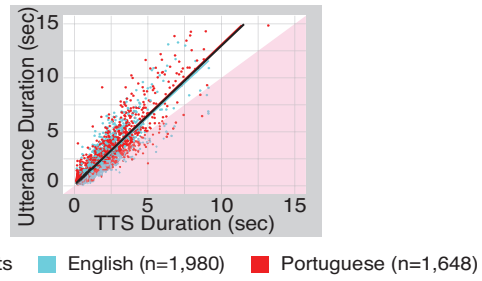


Figure 4: Scatter plot subject utterance by TTS output

resulting value to differentiate utterances that were faster or slower than the TTS output by using zero (0) as a reference division line, and the value was also multiplied by minus one (-1) to display utterances spoken slower than the TTS output as a negative value and faster utterances as a positive value. The resulting values of this conversion are summarised in Table 6 and displayed as percentage values. With this modification, we think that the values in reference to speech rate of the utterance are easier to understand as illustrated in Figure 6.

Speech Rate	Min.	1st	*Mdn*	*M*	3rd	Max.
ILMT-s2s All	−1539.0	−58.8	−23.4	−37.9	1.9	83.3
ILMT-s2s En	−878.7	−52.4	−18.8	−28.7	7.6	83.3
ILMT-s2s Pt	−1539.0	−67.0	−29.1	−49.0	−4.4	58.6

Table 6: *Summary of subject utterance duration divided by TTS output duration — "$1 - (S/T)$"*

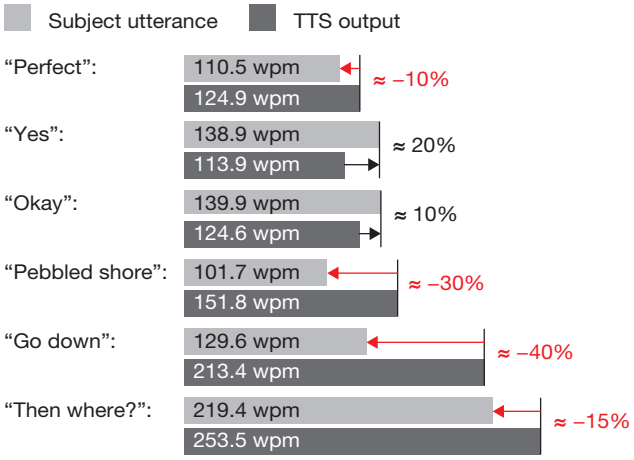


Figure 5: *Example speech rate comparison — Subject utterance & TTS output*

With this comparison, where the duration required to pronounce the word is taken into account, it is possible to say with a higher degree of certainty that the "4.22%" maximum value of the single word utterance of "Perfect" represents a speech rate that is a similar rate to the "3.45%" first quarter value of "Yes" and the "9.00%" mean value of "Okay" as listed in Table 7. Using this method of comparing the subject utterance duration and the TTS output duration, the meaning that was lacking in the wpm values in Table 3 can now be presented with a more comprehensive value — the durational difference between the subject

Difference from TTS (%)					
Utterance	Min.	1st Qu.	*Mdn*	*M*	Max.
Perfect	−27.65	−18.98	−13.61	−11.06	4.22
Yes	−55.82	3.45	18.01	12.91	38.21
Okay	−130.70	−8.14	13.54	9.00	83.30
Pebbled shore	−54.61	−52.06	−49.50	−43.29	−25.77
Go down	−117.00	−74.98	−64.39	−73.36	−56.97
Then where?	−22.55	−19.07	−15.58	−17.31	−13.81

Table 7: *Comparison with TTS duration sample — Percentage difference with TTS output duration*

utterance and a regular speech rate of the TTS output.
This method of measuring the speech rate of the subjects was repeated for the data of the HCRC Edinburgh Map Task corpus and as illustrated in Figure 6 and listed in Table 8, it is possible to see a clearer indication of the subject utterance speech rate. For example, when the measurement of wpm is used, it is not possible to say if the median (*Mdn*) utterance speed for "Okay" at 156.9 wpm was really spoken at half the intentional speed of "Got you" at 372.7 wpm. However, by comparing the utterance duration of the subject and the duration of the TTS output, both utterances can be indicated with a more reasonable median (*Mdn*) speech rate value and say that "Got you" at 42.6% was spoken with a slightly faster speech rate than "Okay" at 33.29%, but not twice as fast as the wpm value would lead one to believe.

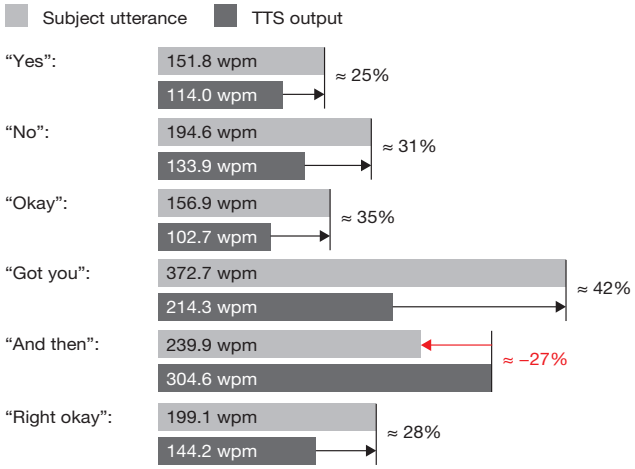


Figure 6: *Example HCRC corpus speech rate comparison — Subject utterance & TTS output*

5. Discussion and Conclusion

By comparing the speech rate of the subject utterance with a reference TTS output of the same utterance, I have indicated that the resulting value is easier to interpret and more robust than the wpm measurement method, by comparing the variable subject utterance duration with a standard measurement of the TTS output duration. Using this method, it becomes possible to visualise the speech rate of short utterances with a higher degree of certainty. For example, Figure 7 illustrates how the values from this method can illustrate how the speech rate changes from utterance to utterance within a dialogue of two different corpora, indicating how speech rates differ when talking directly to ones interlocutor (HCRC corpus) and when one's communication is

Utt. Speech Rate (wpm)						
	Min.	1st Qu.	*Mdn*	*M*	3rd Qu.	Max.
Yes	92.7	134.2	151.8	157.3	180.7	235.6
No	80.3	164.8	194.6	238.8	287.2	653.6
Ok	77.8	134.2	156.9	169.5	189.3	421.1
Got you	308.7	351.2	372.7	381.1	395.1	485.8
And then	158.3	218.1	239.9	271.2	255.7	475.1
Right ok	87.8	165.4	199.1	214.1	243.9	481.3
Difference from TTS (%)						
	Min.	1st Qu.	*Mdn*	*M*	3rd Qu.	Max.
Yes	−23.00	15.58	23.56	23.01	37.02	51.30
No	−66.90	18.43	31.64	32.30	52.78	79.60
Ok	−31.78	22.46	33.29	33.27	45.52	75.93
Got you	38.59	38.93	42.60	43.96	45.78	55.42
And then	−98.43	−47.24	−26.98	−23.83	−12.98	39.67
Right ok	−64.22	12.93	26.54	23.14	40.30	67.41

Table 8: *Comparison with TTS duration sample*

mediated by a Speech-to-Speech Machine Translation system.

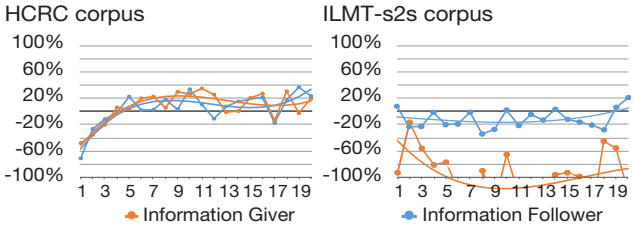


Figure 7: Example of speech rate comparison of speakers

5.1. Pause for thought

However, Yuan et al. (2006, p. 542), in their study of the Switchboard corpus (Godfrey et al., 1992) and the Chinese CallHome and CallFriend corpus (Yuan and Jurafsky, 2005) have indicated that utterances speech rates between one and seven words change drastically:

> We can see that in both English and Chinese, there is an abrupt rise of speaking rate for the segments containing from one to seven words. For the segments having eight to about 30 words, however, the speaking rate stays level. And then, especially in English, the speaking rate rises again, but with a more gradual slope.

A similar phenomenon was observed with the wpm measurements of the subject utterances (Information Giver, and Information Follower), and the TTS output of the ILMT-s2s corpus as illustrated in Figure 8. From the boxplots in Figure 8 it is possible to observe that the mean wpm speech rate increases up to utterances with a word count of 5 and then starts to fluctuate after that. Since the ILMT-s2s corpus data has indicated that there is a speech rate difference between Female (♀) and Male (♂) subjects (Hayakawa et al., 2017b), the grouping of gender was verified and illustrated as Figure 9. From Figure 9 it is possible to observe that the TTS output and Male (♂) subjects' speech rate (wpm) increase rapidly from one word to four or five words, while the Female (♀) subjects' speech rate (wpm) does not increase much and remains basically flat, which is an obser-

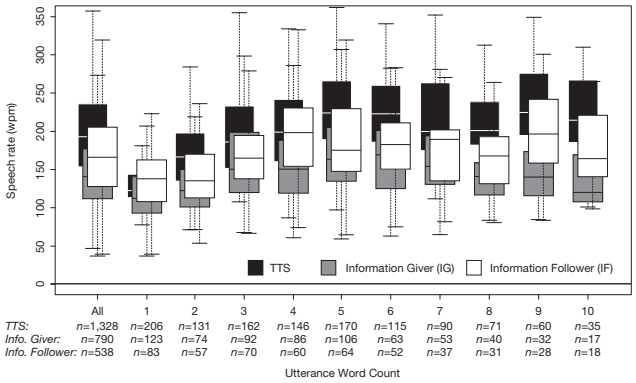


Figure 8: *Boxplot of the 3 talk types' multi word utterance speech rates in various groupings.*

vation that was not reported by Yuan et al. (2006). The

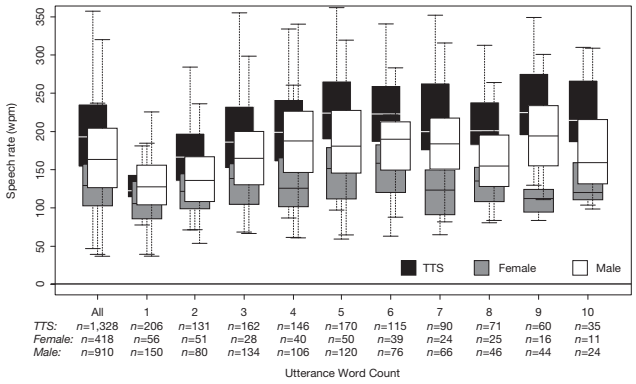


Figure 9: *Boxplot of the 3 talk types' multi word utterance speech rates in various groupings.*

gender difference reported by Yuan et al. (2006, pp. 543–544) was that the Female (♀) subjects had a slower speech rate, but with a minor difference to the Male (♂) subjects.

> Males tend to speak faster than females [. . .] difference between them is, however, very small, only about 4 to 5 words or characters per minute (2%), though it is statistically significant. It might be due to things that we would not normally think of as speech-rate parameters, such as differences in word-frequency distributions.

To verify if this pattern also exists in the HCRC Edinburgh Map Task corpus, the wpm speech rate divided by gender and role were compared as illustrated in Figure 10, with similar curves in the graphs being observed. This indicated that the data of the HCRC Edinburgh Map Task corpus also shows a curvature similar to that reported by Yuan et al. (2006) and also a curvature more similar to the TTS output than the data of the ILMT-s2s corpus. However the increase that Yuan et al. (2006) reported to start with utterances about 30 words long, can be seen to start from ≈12 words for both the subjects of HCRC Edinburgh Map Task corpus and the TTS output. It is important not to forget that the analysis by Yuan et al. (2006) was performed on the SwitchBoard corpus (Godfrey et al., 1992) which is

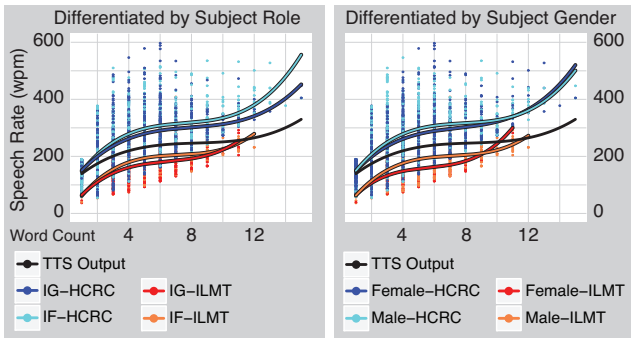


Figure 10: *Scatter plot of ILMT-s2s corpus utterance durations by word count*

a collection of telephone dialogues. Morikawa and Maesako (1998, p. 149) in their reference to the publication in Japanese of Yoshida and Kakuta of a study in audio telephone communication, indicate that the subjects of their study consider communication over the telephone as a different alternative to Face-to-Face communication with its own communication style.

> A study by Yoshida et al. showed that college students in and around Kyoto–Osaka (effective response 549) cited the following reasons for the phone's popularity: speed, available anytime, available anywhere, no visual information about the other end, and easier way to say what we want than in face-to-face conversation. They also cited the following disadvantages: no visual information about the other end and difficulty in conveying subtle emotion. How interesting that no visual information about the other end becomes both an advantage *and* disadvantage !
>
> The above study suggests that the telephone provides its own conversation environment rather than a substitute for face-to-face conversation – in short, people have found a new way to communicate using voice alone.

Due to the different communication methods and the fundamental fact that the data being analysed is different, identical results are not expected, but the pattern is present in both the HCRC Edinburgh Map Task corpus and the ILMT-s2s corpus. The results from Yuan et al. (2006) is of concern because 60% of the data in the ILMT-s2s corpus are located between a word count of 2 words (20th percentile) and 8 words (80th percentile). This range between 2 and 8 words is where Yuan et al. (2006, p. 542) mention that there is an "abrupt rise of speaking rate". The median word count for the Information Giver and the Female (♀) subjects are five and the median word count for the Information Follower and Male (♂) is four. Following this theory, the speech rate of the Information Follower should have a slower wpm speech rate than the Information Giver, and Male (♂) subjects too should have a slower wpm speech rate than the Female (♀) subjects. From Table 9 the Information Giver and Female (♀) subjects and the combination of four word Information Giver and five word Information Follower utterances follow this trend of a faster speech rate

ILMT-s2s corpus (English wpm speech rate)	Min.	1st Qu.	Mdn.	Mean	3rd Qu.	Max.	SD	Count
Subject Role								
Information Giver — 4 word utterances	60.7	119.1	150.5	163.9	188.2	390.9	68.63	86
Information Follower — 4 word utterances	74.0	154.5	198.3	194.2	228.6	362.0	61.39	60
Information Giver — 5 word utterances	59.2	134.9	163.4	171.3	204.9	307.1	55.74	106
Information Follower — 5 word utterances	64.6	148.7	175.2	186.2	229.4	319.5	61.74	64
Subject Gender								
Female (♀) — 4 word utterances	61.3	101.8	125.7	141.0	160.4	332.4	58.89	40
Male (♂) — 4 word utterances	60.7	146.6	187.6	189.7	226.3	390.9	65.55	106
Female (♀) — 5 word utterances	59.2	112.7	151.5	153.0	177.9	287.9	52.19	50
Male (♂) — 5 word utterances	64.6	145.7	180.8	186.8	227.3	319.5	58.09	120

Table 9: *Summary and standard deviation of 4 and 5 word utterances (wpm)*

for utterances with more words, but it is not the case for the Information Follower, Male (♂) subjects or the difference between four word Information Follower and five word Information Giver utterances. In both cases, the effect of the role of the subject (Information Giver being slower than the Information Follower) or the gender of the subject Female (♀) being slower than the Male (♂)) is stronger than the influence of the utterance word count. For this reason, the wording of "an abrupt rise of speaking rate for the segments containing from one to seven words" by Yuan et al. (2006, p. 542) would not be appropriate for the data of the ILMT-s2s corpus.

A further point of interest is the fact that Yuan et al. (2006, pp. 544) also identified the ambiguity of the wpm measurement method caused by the different "word-frequency distributions" when describing the gender speech rate difference: "It might be due to things that we would not normally think of as speech-rate parameters, such as differences in word-frequency distributions". When Figure 10

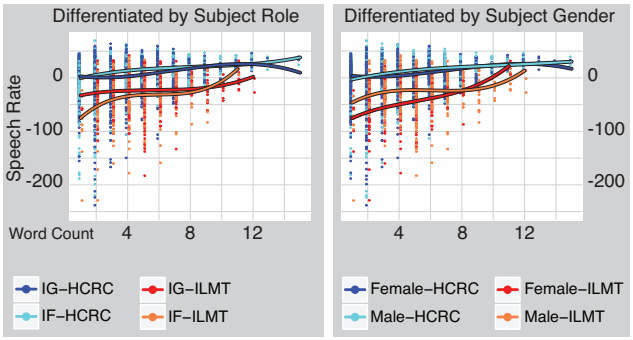


Figure 11: *Scatter plot of ILMT-s2s corpus utterance durations by word count*

which used the wpm as the speech rate indicator is represented using the speech rate calculation method of comparing the TTS output duration with the subject utterance duration, it is possible to see that the speech rate of the subjects do not show "an abrupt rise" as indicated in Figure 11 with the lines representing the HCRC Edinburgh Map Task corpus closer to a linear model than a non-linear model from 1 word all the way to 12 word utterances. This is both an indication of (i) how close the TTS output is to direct human communication, and (ii) the lack of "an abrupt rise" in the speech rate, with the word-frequency distributions no longer affecting the measurement values of the method explained in this paper.

6. Acknowledgements

This research is supported by Science Foundation Ireland through the CNGL Programme (Grant 12/CE/I2267) and the Research Centres Programme (Grant 13/RC/2106) in the ADAPT Centre (www.adaptcentre.ie) at Trinity College Dublin. The ADAPT Centre for Digital Content Technology is co-funded under the European Regional Development Fund.

7. Bibliographical References

Anderson, A. H., Bader, M., Bard, E. G., Boyle, E., Doherty, G., Garrod, S., Isard, S., Kowtko, J., McAllister, J., Miller, J., Sotillo, C., Thompson, H. S., and Weinert, R. (1991). The HCRC Map Task Corpus. *Language and Speech*, 34(4):351–366.

Arai, T. and Greenberg, S. (1997). The Temporal Properties of Spoken Japanese are Similar to those of English. In *Proceeding of EUROSPEECH '97*, pages 1011–1014, Rhodes, Greece. ISCA.

Batliner, A., Nöth, E., Buckow, J., Huber, R., Warnke, V., and Niemann, H. (2001). Whence and whither prosody in automatic speech understanding: a case study. In *Proceedings of the Workshop on Prosody and Speech Recognition*, pages 3–12. ISCA.

Bell, A., Gregory, M. L., Brenier, J. M., Jurafsky, D., Ikeno, A., and Girand, C. (2002). Which Predictability Measures Affect Content Word Durations? In *Proceedings of the ISCA Workshop on Pronunciation Modeling and Lexicon Adaptation for Spoken Language Technology*, pages 1–5. ISCA.

Bell, A., Brenier, J. M., Gregory, M., Girand, C., and Jurafsky, D. (2009). Predictability effects on durations of content and function words in conversational English. *Journal of Memory and Language*, 60(1):92–111.

Boersma, P. and van Heuven, V. (2001). Speak and unSpeak with PRAAT. *Glot International*, 5(9-10):341–347.

Branigan, H. P., Pickering, M. J., Pearson, J., and McLean, J. F. (2010). Linguistic alignment between people and computers. *Journal of Pragmatics*, 42(9):2355–2368.

de Jong, N. H. and Wempe, T. (2009). Praat script to detect syllable nuclei and measure speech rate automatically. *Behavior Research Methods*, 41(2):385–390, May.

Godfrey, J., Holliman, E., and McDaniel, J. (1992). Switchboard: telephone speech corpus for research and development. *Proceedings of ICASSP-92*, pages 517–520.

Greenberg, S. (1997). The Switchboard Transcription Project. Technical report, Johns Hopkins University, Baltimore, Maryland, USA.

Greenberg, S. (1999). Speaking in shorthand – a syllable-centric perspective for understanding pronunciation variation. *Speech Communication*, 29(2–4):159–176.

Hara, K. and Iqbal, S. T. (2015). Effect of Machine Translation in Interlingual Conversation: Lessons from a Formative Study. In *Proceedings of CHI'15*, pages 3473–3482, New York, NY, USA. ACM.

Hayakawa, A., Cerrato, L., Campbell, N., and Luz, S. (2015). A Study of Prosodic Alignment in Interlingual Map-Task Dialogues. In The Scottish Consortium for ICPhS, editor, *Proceedings of the 18th International Congress of Phonetic Sciences*, Glasgow, UK. The University of Glasgow. Paper number 0760.1-9.

Hayakawa, A., Haider, F., Luz, S., Cerrato, L., and Campbell, N. (2016a). Talking to a system and oneself: A study from a Speech-to-Speech, Machine Translation mediated Map Task. In *Proceedings of Speech Prosody 2016 (SP8)*, pages 776–780, Boston, Massachusetts, USA. ISCA.

Hayakawa, A., Luz, S., Cerrato, L., and Campbell, N. (2016b). The ILMT-s2s Corpus — A Multimodal Interlingual Map Task Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of LREC 2016*, pages 605–612, Paris, France. ELRA.

Hayakawa, A., Vogel, C., Luz, S., and Campbell, N. (2017a). Perception Changes With and Without the Video Channel: A Study from a Speech-to-Speech, Machine Translation Mediated Map Task. In *Proceedings of CogInfoCom 2017*, pages 401–406, Debrecen, Hungary. IEEE.

Hayakawa, A., Vogel, C., Luz, S., and Campbell, N. (2017b). Speech Rate Comparison when Talking to a System and Talking to a Human: A study from a Speech-to-Speech, Machine Translation mediated Map Task. In *Proceedings of INTERSPEECH'17*, pages 3286–3290, Stockholm, Sweden. ISCA.

Lewis, W. D. (2015). Skype translator: Breaking down language and hearing barriers. *Translating and the Computer (TC37)*.

Matsuda, S., Hu, X., Shiga, Y., Kashioka, H., Hori, C., Yasuda, K., Okuma, H., Uchiyama, M., Sumita, E., Kawai, H., and Nakamura, S. (2013). Multilingual Speech-to-Speech Translation System: VoiceTra. In *Proceedings of the 2013 IEEE 14th International Conference on Mobile Data Management*, volume 2, pages 229–233, Milan, Italy. IEEE.

Morikawa, O. and Maesako, T. (1998). HyperMirror: Toward Pleasant-to-use Video Mediated Communication System. In *Proceedings of CSCW '98*, pages 149–158, New York, NY, USA. ACM.

Sztahó, D., Kiss, G., and Vicsi, K. (2015). Estimating the Severity of Parkinson's Disease from Speech Using Linear Regression and Database Partitioning. In *Proceedings of INTERSPEECH'15*, pages 498–502, Dresden, Germany. ISCA.

Yuan, J. and Jurafsky, D. (2005). Detection of questions in Chinese conversational speech. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.*, pages 47–52, Nov.

Yuan, J., Liberman, M., and Cieri, C. (2006). Towards an Integrated Understanding of Speaking Rate in Conversation. In *Proceedings of INTERSPEECH'06*, pages 2197–2200, Pittsburgh, PA, USA. ISCA.

Zipf, G. K. (1945). The Meaning-Frequency Relationship of Words. *The Journal of General Psychology*, 33(2):251–256.

8. Language Resource References

Hayakawa, A. and Luz, S. and Cerrato, L. and Campbell, N. (2015). *The ILMT-s2s Corpus*. CNGL Programme, distributed via ELRA, Trinity College Dublin, 1.0, ISLRN 100-610-774-625-0.

Beyond Generic Summarization: A Multi-faceted Hierarchical Summarization Corpus of Large Heterogeneous Data

Christopher Tauchmann*, Thomas Arnold*, Andreas Hanselowski*, Christian M. Meyer* and Margot Mieskes‡
Research Training Group AIPHES
*Technische Universität Darmstadt; ‡Hochschule Darmstadt
https://www.aiphes.tu-darmstadt.de

Abstract

Automatic summarization has so far focused on datasets of ten to twenty rather short documents, typically news articles. But automatic systems could in theory analyze hundreds of documents from a wide range of sources and provide an overview to the interested reader. Such a summary would ideally present the most general issues of a given topic and allow for more in-depth information on specific aspects within said topic. In this paper, we present a new approach for creating hierarchical summarization corpora from large, heterogeneous document collections. We first extract relevant content using crowdsourcing and then ask trained annotators to order the relevant information hierarchically. This yields tree structures covering the specific facets discussed in a document collection. Our resulting corpus is freely available and can be used to develop and evaluate hierarchical summarization systems.

Keywords: hierarchical summarization, large corpora, heterogeneous sources, crowdsourcing, aspect-oriented summarization

1. Introduction

Automatically created summaries are most useful if they allow readers to save time when reading long and/or many documents from a large number of sources. However, many state-of-the-art approaches in automatic multi-document summarization (MDS) are still evaluated on small clusters of ten to twenty short articles. The most prominent document collections from the DUC and TAC conferences have, for example, only about 6,700 (DUC '04) and 17,400 (DUC '06) tokens per topic cluster.[1] This evaluation setup does not cover the full potential of automatic summarization, which *could* easily aggregate collections of over hundred documents with more than 100,000 tokens.

In some respects, the current setup is not even very realistic, as the vast majority of the available datasets cover only newswire text about a single event or entity (Nenkova, 2005). Given the large amount of redundancy in this text type, a human reader could read only one or two of the source documents and quickly skim over the remaining ones to get a good overview of the article's main event or entity – albeit update summaries would be helpful in this situation. Even more recent work in social media and real-time summarization is based on high-redundancy text (Chua and Asur, 2013; Lin et al., 2016). In large heterogeneous document collections, there are important facts and arguments that appear only in few of the available documents and are therefore missed by generic summary strategies and absent from both automatic and reference summaries.

With increasing volume, velocity, and variety of the source documents, it gets, however, extremely difficult to construct suitable evaluation corpora. Assuming a reading speed of 228 ± 30 words per minute for English (Trauzettel-Klosinski and Dietz, 2012), it already takes more than seven hours (excluding breaks) to read a document collection with 100,000 words. It is hardly possible for an individual annotator to stay equally concentrated for that many hours. This yields a bias in the resulting summary, as the annotators will gradually shift their notion of what is important – especially in heterogeneous low-redundancy texts where frequency of occurrence is not a good indicator for importance. Although query-focused or aspect-oriented summaries yield a frequency-agnostic notion of importance, the resulting summarization corpora cover only a small fraction of the collection's content, which makes the annotation less cost-efficient. Corpora covering only a few narrow queries also lack the general overview of the large variety of facets typically discussed in broad and large collections.

In this work, we propose a novel approach to create summarization corpora for large document collections by structuring the important information hierarchically. We particularly focus on controversial topics from the educational domain, such as *alternative ADHD treatments*. This topic also serves as a running example throughout the paper, as it may be viewed from many different facets (or points of view), including ADHD prevalence, risk groups, diagnosis, nutrition treatment, herbal treatment, hypnosis, and music therapy. We would expect this kind of information in a generic summary about the topic. However, each facet should also branch off and discuss the most important symptoms for affirming or excluding a diagnosis in one branch, as well as different procedures, their advantages and disadvantages, and evidence for their effectiveness in other treatment-specific branches. A hierarchical structure of this and similarly complex topics therefore covers general information about the topic as well as detailed information on each facet discussed in the document collection. Methods for automatically creating such hierarchical summaries are highly relevant to complex information seeking processes that assist users in gaining an overview *and* diving into specific facets of a controversial topic. However, we require new hierarchical summarization corpora in or-

[1] http://duc.nist.gov, http://tac.nist.gov

der to research and evaluate automatic systems. Our approach is suitable to create such corpora for large, heterogeneous datasets of over 100,000 tokens spanning multiple genres (e.g., scientific articles, blogs, forum posts).

Our key idea is to first collect the most relevant information independent of the actual use for the summary and then identify redundancy, granularity, and facet by organizing the collected information bottom-up into a hierarchy. Each tree of this hierarchy covers a different facet discussed in the document collection, including general definitions, specific facts, and opinions. More general information resides near the root of the tree, while more specific facts and opinions branch off to deeper tree levels grouped by topical or argumentative strand. Within the same hierarchy, we also mark redundant information by combining two information nuggets in a single tree node.

Figure 1 shows an overview of our corpus construction approach. For the first step, *content selection*, we use crowdsourcing, which allows us to process large document collections. For the second step, we rely on expert annotators and provide them with clear guidelines and a novel open-source annotation tool enabling the *hierarchical organization* of the content.

The scientific community can benefit from the proposed solution in multiple ways: Our corpus of hierarchical summaries can be used as a benchmark for automatic hierarchical summarization and information structuring methods, such as the works by Christensen et al. (2014) and Erbs et al. (2013), where there is yet almost no data available. While the hierarchical structure qualifies as a useful summary in itself, our data additionally allows us to generate textual summaries based on different parts of the hierarchy. A particular advantage of this approach is that we can summarize *all* facets discussed in a document collection by summarizing each tree of the hierarchy individually. This will save much time when creating large multi-faceted summarization corpora compared to summarizing documents for a few predefined facets, as it has been done, for example, for TAC 2010. By considering a tree's depth, we additionally gain control over the length and the level of detail of the resulting summaries.

Furthermore, we provide detailed information on our crowdsourcing setup and we publish the novel annotation tool for hierarchical summarization as open-source software in order to foster the creation of new summarization corpora.[2]

2. Related Work

Christensen et al. (2014) propose automatic hierarchical summarization, but they evaluate their system using an existing news dataset without hierarchical structure and they focus mostly on the temporal clustering of news events. Given this limited evaluation setup, we see a clear demand for new evaluation corpora that explicitly contain a hierarchical organization of the source documents' information and cover text types different from news. This will also bridge the gap between research into summarization and text structuring, such as (Erbs et al., 2013; Pembe and Güngör, 2010).

Zhang et al. (2017) discuss recursive summarization for online forums. They iteratively replace parts of the discussion with summaries, yielding a hierarchy of summaries. Our work differs in that we suggest a holistic rather than an incremental approach, which allows us to group information from discussion strands that cover related topics.

Nakano et al. (2010) focus on information credibility. They create survey reports by asking expert annotators to highlight important information in crawled web documents and describe its relation to a given topic. Based on the annotated data, they formulate summaries and investigate the impact of the annotators' information credibility descriptions on the final summary. Though they also work with large document collections, their data is not publicly available.

Falke and Gurevych (2017) recently proposed concept-map-based summarization to structure information in large document collections. Their notion of a concept map yields a generic summary that conflates all facets into a single structure of about 25 related concepts. Our work differs from that, as we organize a document collection according to the multiple facets discussed in a strict hierarchy. This enables us to induce multiple aspect-oriented summaries at varying levels of detail. Additionally, we do not rely on open information extraction, which would ignore much context and abstract from complex discourse structures, such as argumentation. Instead, we work with verbatim segments of the source texts.

Li et al. (2017) raise the issue that multi-document summarization falls short of including varying facets in the source documents. They focus on news reports and related reader comments and opinions, for which they observe that information items will not be included in a summary unless they are salient – even if the information might be interesting to readers. Li et al. (2017) also discuss comments expressing sentiments that contradict the source documents. Our proposed corpus aligns well with their work, since a hierarchy contains both salient information typically found in generic reports and opinionated and controversial statements from user comments.

Query-focused summarization (Allan et al., 2008; Baumel et al., 2016) and real-time summarization (Lin et al., 2016) are similar tasks to our work, since they aim at summarizing a specific facet discussed in a document collection or address the summarization of large amounts of data. Our hierarchical corpus construction approach yields interesting evaluation data for these tasks, since query-focused summarization systems can be trained towards multiple facets discussed in a document collection at the same time, whereas real-time summarization systems have to decide about the importance even if they do not have access to all source documents yet. Hierarchical summarization systems that generate a hierarchy similar to our manually constructed ones could yield a promising solution to this task.

So far, a lot of research in automatic summarization has been done on news documents, which has a range of shortcomings, as dicussed by Zopf et al. (2016) and Benikova et al. (2016). They argue that the spectrum of possible applications is severely limited when focusing on homogeneous

[2] GitHub repository with available data and software: https://github.com/AIPHES/HierarchicalSummarization

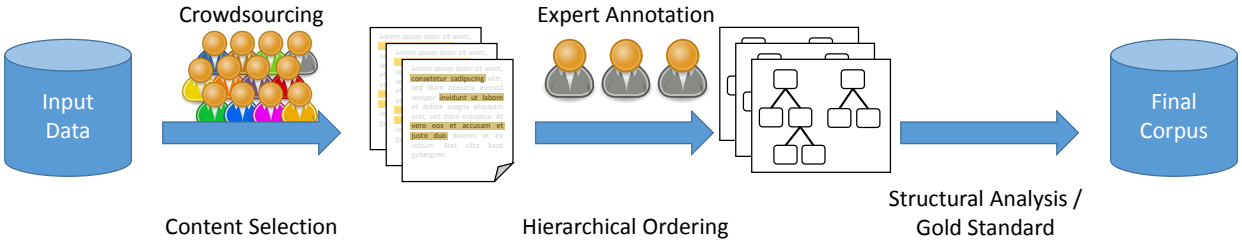


Figure 1: Overview of our corpus construction approach for hierarchically summarizing large document collections

datasets of a single text type. Both approaches propose heterogeneous summarization corpora of generic, text-based summaries, which are different from our hierarchical summaries. Nevertheless, our document collections have similar properties of incorporating heterogeneous text types.

3. Content Selection

Figure 1 shows the main steps of our corpus construction approach. In this section, we describe the content selection step, including the heterogeneous sources we use as input data, our methodology to frame the selection of important information nuggets as a crowdsourcing task, and the analysis of the resulting data.

3.1. Heterogeneous Sources

The basis for our experiment is the ClueWeb12-based focused retrieval dataset by Habernal et al. (2016). This dataset consists of 49 broad educational topic clusters with about 40–100 English documents per topic cluster. The documents are highly heterogeneous, including scientific articles, blogs, forums, personal ads, etc. Accordingly, we find both objective facts and opinionated or controversial content in this dataset. We remove duplicate sentences and documents and use only sentences that are marked relevant for a given topic in the focused retrieval dataset. This reduces the corpus from 4,820 documents with 628,026 sentences to 3,984 documents with 171,976 sentences. For our corpus, we have selected ten of those broad topic clusters. Table 1 shows the number of documents, sentences, and tokens in each topic cluster. While all topic clusters are much larger than the commonly used DUC '06 data, we sample three large (> 125,000 tokens), four medium-sized (> 50,000), and three smaller topic clusters (< 50,000). This allows us to analyze the scalability of our corpus construction approach.

3.2. Crowdsourcing and HIT Design

For the selection of important content, we use crowdsourcing. This allows us to process large document collections by breaking down the complex task into many small microtasks (Cheng et al., 2015) – so-called human intelligence tasks (HIT). Since Lloret et al. (2013) report unsatisfactory results when crowdsourcing extractive summarization, we propose a different crowdsourcing setup and further break down the summarization task into manageable microtasks by asking the crowd workers to *collect* what they consider relevant for a summary rather than to assess or rank the importance of each information at the same time.

Topic clusters	Doc.	Sent.	Tokens
Concerns about religious classes	87	7,654	210,211
School punishment policy	89	6,409	149,268
Parents of kids doing drugs	78	6,183	125,584
Children's obesity	90	3,916	90,963
Sleep problems in preschools	86	3,119	65,216
Student loans	95	2,346	54,434
Discipline in elementary school	83	2,586	53,592
Alternative ADHD treatments	57	1,475	28,281
Kids with depressions	39	1,209	21,644
Cellphone use in schools	61	902	21,384
Total	786	38,304	820,577

Table 1: Overview of our document collections and topics

We therefore generate HITs showing seven consecutive sentences from our input data at a time. In each HIT, we ask the crowd workers to mark all facts, opinions, hypotheses/statements and claims (called *information nuggets* henceforth) that they would include in a summary on the overall topic of the document collection. Our notion of information nugget is similar to previous definitions of nugget (Voorhees, 2004; Benikova et al., 2016) and semantic content unit (Nenkova et al., 2007). Workers should select only information nuggets of at least three words and a maximum length of one sentence. Each nugget should include a verb and be understandable without further context. The workers may identify multiple information nuggets within a HIT. In case they cannot find any relevant nugget, we ask them to describe the document's content to avoid spammers. Below the task description, we show two examples to illustrate the HIT. Along with the full paper, we provide a HIT template and all collected data.

Figure 2 shows a HIT for our running example. The task description is located at the top of the page. Using the *examples* button, the workers can show or hide a number of annotated examples to understand the task. Recurring workers doing multiple HITs typically do not need the examples anymore, but immediately start the annotation. They create an information nugget by clicking on its first and last word in the text. The spanned words will then be highlighted in yellow and the information nugget will be listed as a *relevant text segment*. If workers cannot find any information nuggets in a text, we ask them to summarize the text in two to three keywords. This enforces involvement and prevents workers from submitting HITs without carefully reading them.

We determine the optimal task length, payment, and num-

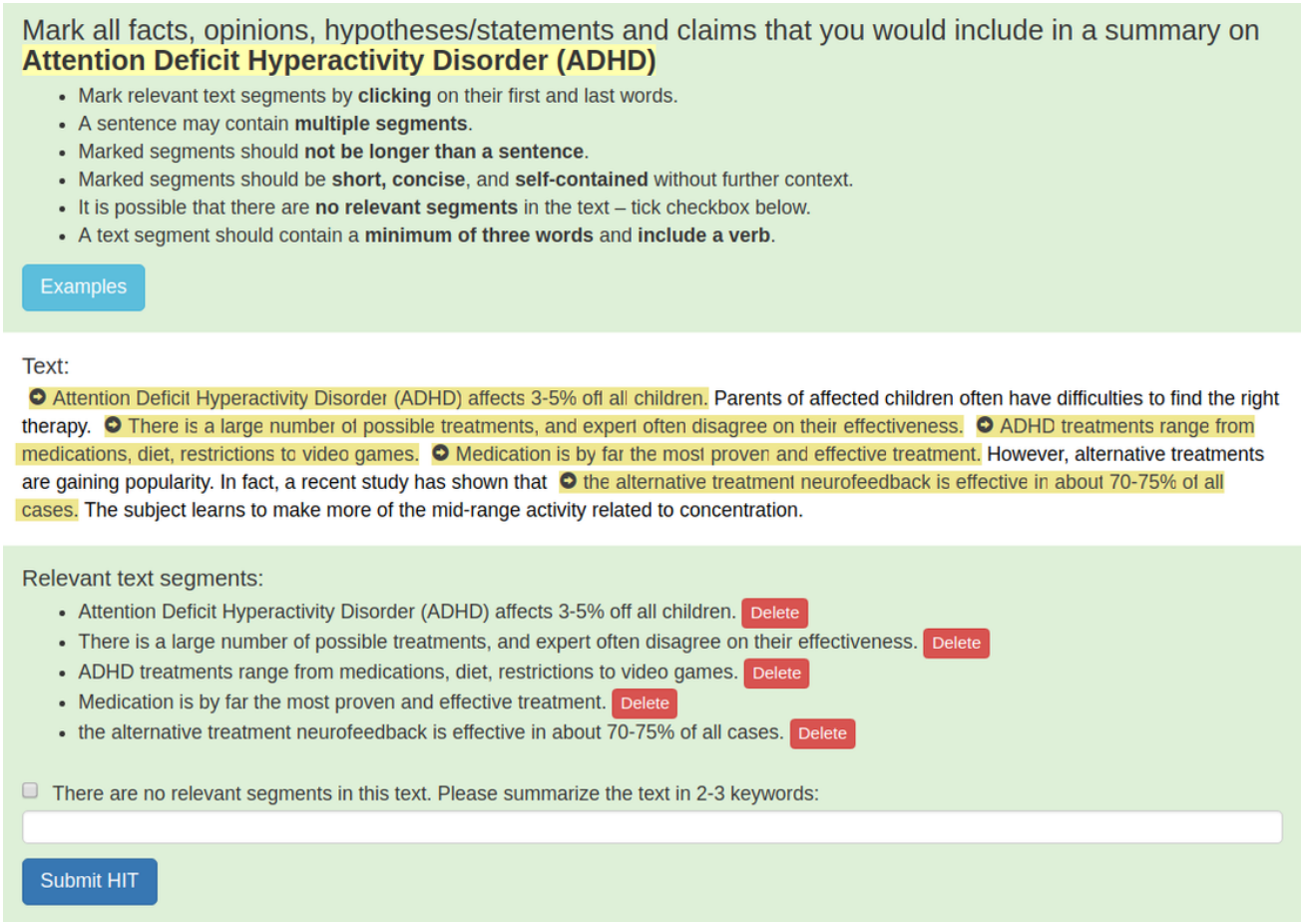


Figure 2: Screenshot of a HIT for the alternative ADHD treatments topic cluster

ber of annotators in a preliminary study. As a good trade-off between the number of HITs and the amount of work, we suggest to show short paragraphs of seven sentences in a single HIT. For each completed HIT, we pay US$ 0.07, which we find reasonable for a task of 60–90 seconds. The payment is high enough to attract reliable workers, while discouraging spammers. As quality is hard to control in a crowdsourcing setup (Bigham et al., 2015), we assign each HIT to seven workers. We select only workers with an acceptance rate of at least 98 %, we manually check annotations, reject work that does not meet our standards, and block workers where necessary.

3.3. Inter-Annotator Agreement

The crowd workers marked 68,220 information nuggets in total. Table 2 shows their inter-annotator agreement, computed using three commonly used metrics: percentage agreement A_O, Fleiss' κ (Fleiss, 1971), and Krippendorff's α_U (Krippendorff, 1995) as implemented in DKPro Agreement (Meyer et al., 2014). While A_O and κ measure agreement at the token level, α_U considers agreement between spans of selected tokens (i.e., the entire information nuggets). Both κ and α_U are chance-corrected agreement metrics (Artstein and Poesio, 2008).

The first row of Table 2 shows the scores for annotator agreement between all seven workers. The agreement is similar to previous work in summarization (Zechner, 2002; Benikova et al., 2016). In the second to fourth row, we report the agreement for the small, medium-sized, and large topic clusters individually without noticing a clear drop in annotation quality. This confirms that our crowdsourcing setup scales to large document collections.

	A_O	κ	α_U
All crowd workers	0.664	0.149	0.201
only large topic clusters	0.691	0.152	0.222
only medium topic clusters	0.634	0.127	0.189
only small topic clusters	0.666	0.170	0.186
MACE vs. Experts	0.688	0.314	0.311

Table 2: Inter-annotator agreement

To validate our results, we compare the best annotations of the seven workers according to MACE (Hovy et al., 2013) to an expert annotator, who selected information nuggets from 322 sentences. The results in the fifth row show that we reach relatively high agreement, with κ of 0.311 and α_U of 0.314. This indicates that the crowd workers selected reliable information nuggets.

3.4. Gold Standard

Most of the 68,220 information nuggets have been annotated by just a single crowd worker. To avoid singular nugget selections for the nonce, we consider only nuggets for our corpus that have been selected by at least three annotators. We remove nuggets shorter than three tokens

and merge overlapping ones. This remaining dataset has 4,983 information nuggets (7.3% of the original information nuggets), which is a manageable size for expert annotation. Within our corpus repository, we provide the source documents, the original information nuggets from Amazon Mechanical Turk, and the post-processed nuggets that serve as input for the annotation tool. The annotations are licensed under CC-BY 4.0.

4. Hierarchical Ordering

After collecting the information nuggets through a crowdsourcing approach, we structure them into hierarchies. We propose a new annotation process and a tool supporting this process. We analyze the resulting hierarchies by means of a novel evaluation metric we call *hierarchy overlap*. We finally discuss the resulting gold standard corpus of multi-faceted hierarchical summaries.

4.1. Expert Annotation and Annotation Tool

A *hierarchy* $H(V, E)$ is a forest – i.e., a directed and acyclic graph with a set of nodes V and a set of hierarchical relations $E \subseteq V \times V$. Each node $v \in V$ contains one or more information nuggets. Thus, V is a partition of the set of all information nuggets N with $\bigcup_{i \in V} v_i = N$. Each edge $(v_1, v_2) \in E$ connects more general nuggets in v_1 with more specific nuggets in v_2 discussing the same facet. There is no shared root node, so the hierarchy typically consists of multiple *facet trees*. Each facet tree contains all nuggets from one facet of the overarching topic (e.g., prevalence of ADHD), which branches off from general (e.g., overall average prevalence) to more specific information (e.g., prevalence among certain age groups or regions. To create such a hierarchy, an annotator needs to find the globally best position within the current facet trees or start a new one. The results by Lloret et al. (2013) suggest that this task cannot be broken down to a crowdsourcing setup without suffering quality problems. Therefore, we hire three expert annotators from the field of computational linguistics. This is reasonable, since the amount of data that remains after the content selection step is manageable.

To allow for an efficient annotation, we have developed a novel open-source hierarchy annotation tool with a graphical user interface. Figure 3 shows a screenshot. Input for this tool is a list of information nuggets with unique IDs and additional context from the source text, in our case the preceding and succeeding sentence.

Our tool presents a list of information nuggets that still have to be included in the hierarchy, and a working space displaying the current state of the hierarchy. Information nuggets can be added as new nodes, or into existing nodes to indicate redundant information. Alternatively, the user may structure nodes both vertically by descending salience and granularity and horizontally in new facet trees if they discuss a new facet of the overall topic. The output of the tool is the hierarchical structure in a simple XML file format.

4.2. Qualitative Analysis

For the three largest topic clusters, the annotators created hierarchies that contain 10 to 30 facet trees with an average depth of five levels. They require about six hours on average per topic cluster. One beneficial characteristic of the hierarchical structures is that different facets of controversial topics are naturally structured. Thereby, the parent node represents a specific facet and the leaf nodes different viewpoints. In the topic cluster on alternative ADHD treatments, for example, the annotators have decided to distinguish different kinds of treatments and collected claims and evidence which confirm or refute their effectiveness. Table 3 shows the number of nodes, facet trees, and average facet tree depth of all annotated hierarchies per topic.

Our qualitative analysis shows that annotators are able to structure the facets of a topic in different parts of a hierarchy. Motivated by these results, we quantify the annotators' agreement on creating the hierarchies.

4.3. Structural Analysis

To compare two hierarchies H_1 and H_2 for the same topic cluster and nugget set N, we use a modification of the *taxonomy overlap* (Maedche and Staab, 2002)

$$TO(n, H_1, H_2) = \frac{|SC(n, H_1) \cap SC(n, H_2)|}{|SC(n, H_1) \cup SC(n, H_2)|}$$

where $SC(n, H)$ is the set of all nuggets contained in sub- or supernodes (the semantic cotopy) of the node containing information nugget $n \in N$ in hierarchy H.

The averaged similarity between two hierarchies is the sum of the taxonomy overlap of all nuggets, normalized by the number of nuggets:

$$TO(H_1, H_2) = \frac{1}{|N|} \sum\nolimits_{n \in N} TO(n, H_1, H_2)$$

This metric was originally developed to measure the similarity between taxonomies and ontologies. It has been used and adapted for a variety of tasks (Euzenat and Shvaiko, 2007). However, in this metric, the order of the nodes is not important, as the metric should also compare ontologies with symmetric relations (e.g., similar-to). In our work, the relations are strictly hierarchical. Using the TO metric, a hierarchy H_1 with edges $(v_1, v_2), (v_2, v_3) \in E_1$ ("v_1 over v_2 over v_3") compared to a hierarchy H_2 with edges $(v_3, v_2), (v_2, v_1) \in E_2$ ("v_3 over v_2 over v_1") would yield a score of $TO(H_1, H_2) = 1$ (a perfect match), which contradicts our notion of a hierarchy branching from general to specific information.

Therefore, we propose our new modification called the *hierarchy overlap*

$$\begin{aligned} HO(H_1, H_2) = a \cdot TO(H_1, H_2) \\ + b \cdot SupO(H_1, H_2) + c \cdot SubO(H_1, H_2) \end{aligned}$$

which is the weighted sum of TO, the superset overlap $SupO$, and the subset overlap $SubO$ score. We compute $SupO$ and $SubO$ from taxonomy overlap TO variants that replace the full semantic cotopy SC with the nugget set of sub- *or* supernodes, respectively. Choosing the right values for the parameters a, b and c sets a trade-off between overall facet tree content and correct ordering. For our scenario, we create a small test case, explore different values

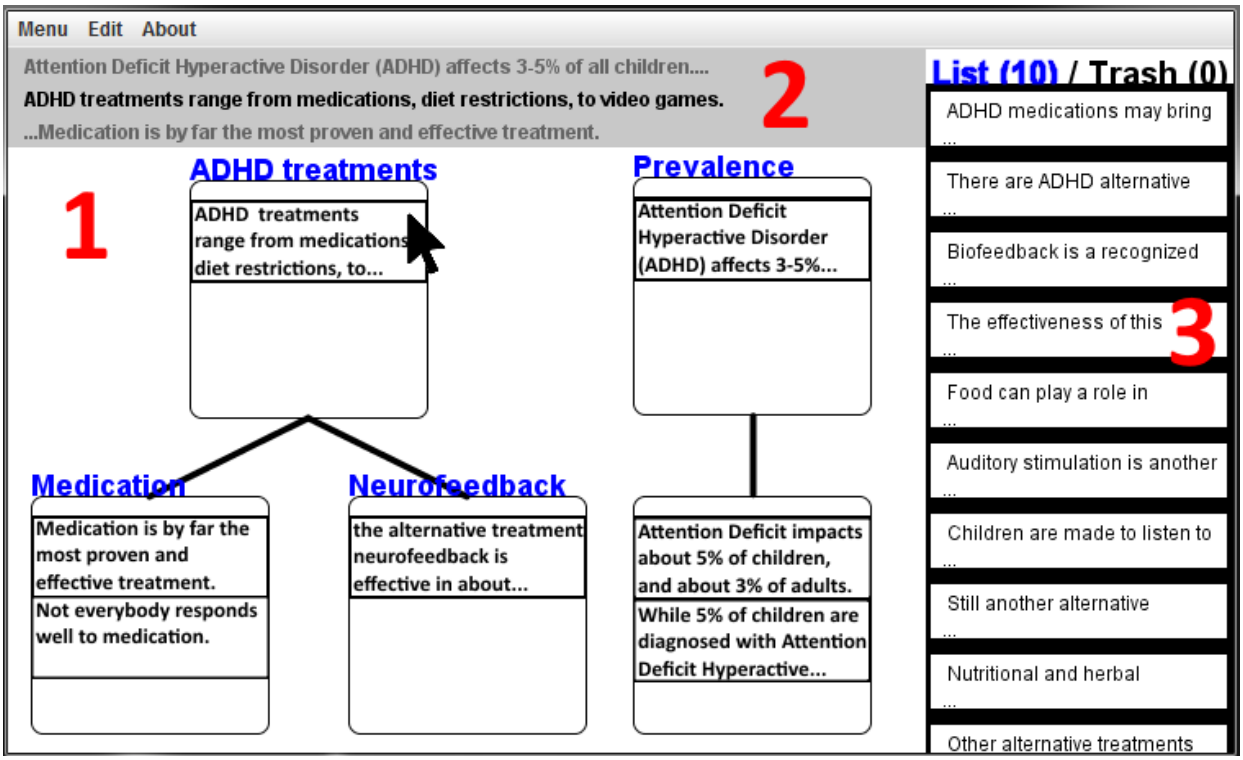


Figure 3: Screenshot of the annotation tool user interface. Area 1 is the main working space, with two annotated facet trees. Area 2 shows the full text of the hovered nugget, with preceding and succeeding sentences from the original document as context. Area 3 is a list of remaining nuggets that still have to be included in the hierarchy.

Topic	Nuggets	Nodes			Facet trees			Depth		
		A_1	A_2	A_3	A_1	A_2	A_3	A_1	A_2	A_3
Concerns about religious classes	717	705	706	711	33	81	20	5.42	2.23	3.80
School punishment policy	796	704	787	747	22	29	13	5.45	2.55	6.62
Parents of kids doing drugs	1,221	1,033	1,214	1,132	31	139	10	5.35	2.06	7.50
Children's obesity	445	415	441	434	10	60	11	8.80	2.25	4.45
Sleep problems in preschools	408	401	400	390	17	56	5	7.35	2.25	8.60
Student loans	586	521	586	507	26	44	15	5.92	2.34	4.20
Discipline in elementary school	341	334	338	336	23	48	14	5.13	2.50	3.42
Alternative ADHD treatments	235	185	221	204	14	13	5	3.00	3.77	4.80
Kids with depressions	146	144	143	144	4	33	6	8.50	2.03	6.00
Cellphone use in schools	88	86	88	88	3	25	8	8.00	1.76	4.38

Table 3: Input nuggets, number of nodes, facet trees and average facet tree depth of final hierarchies (3 annotators per topic)

for the parameters and evaluate them manually. Since the partitioning of information nuggets into facet trees is our biggest priority, we use $a = 0.8$ and $b = c = 0.1$. In this case, $SupO$ and $SubO$ do not have major impact, but act as tie breakers to ensure correct information nugget order. The final HO score is still between 0 and 1.

As a simple baseline, we compute HO on randomly generated hierarchies for every topic cluster, which is between 0.09 and 0.15, depending on the topic size. In comparison, the pairwise HO of the three manually annotated hierarchies is between 0.16 and 0.28. The higher hierarchical overlap indicates that the expert annotators did agree on substantial parts of the hierarchies.

Hierarchy Overlap Example

Figure 4 shows two example hierarchies. The semantic cotopy of nugget X in hierarchy H_1 consists of all nuggets contained in sub- or supernodes of X, {A, B, C, D, E}. The semantic cotopy of nugget X in H_2 is exactly the same set. Therefore, the taxonomy overlap of nugget X in hierarchies H_1 and H_2 equals

$$\frac{|SC(X,H_1) \cap SC(X,H_2)|}{|SC(X,H_1) \cup SC(X,H_2)|} = \frac{|\{A,B,C,D,E\}|}{|\{A,B,C,D,E\}|} = 1$$

The intersection of the respective **super**sets consists of only one nugget $\{A\}$, the union has four nuggets $\{A, B, D, E\}$. The superset overlap $SupO(H_1, H_2)$ equals

$$\frac{|SupS(X,H_1) \cap SupS(X,H_2)|}{|SupS(X,H_1) \cup SupS(X,H_2)|} = \frac{|\{A\}|}{|\{A,B,D,E\}|} = \frac{1}{4}$$

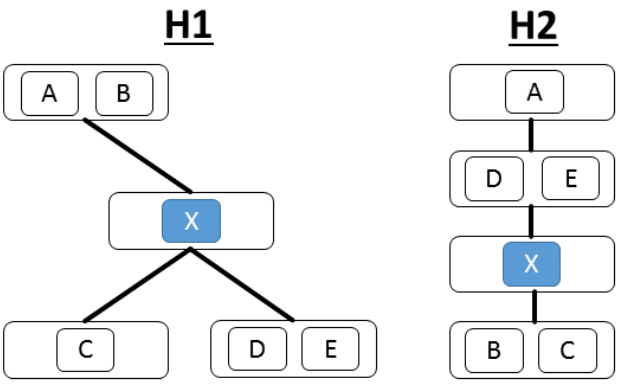


Figure 4: Hierarchy Overlap Example figure (see section 4.3. for explanations)

with the set of all nuggets $SupS(n, H)$ contained in supernodes of the node containing nugget n.
Similarly, the intersection of the **sub**sets consists of only one nugget $\{C\}$, the union has four nuggets $\{B, C, D, E\}$. The subset overlap $SubO(X, H_1, H_2)$ is $\frac{1}{4} = 0.25$. With $a = 0.8$ and $b = c = 0.1$, as proposed, the hierarchy overlap of nugget X equals

$$HO(X, H_1, H_2) = 0.8 * 1 + 0.1 * 0.25 + 0.1 * 0.25 = 0.85$$

4.4. Gold Standard

The proposed comparison metric HO enables us to create a gold standard hierarchy H_G from the three manually annotated hierarchies H_1, H_2, and H_3 for a given topic cluster. In this automatic process, we consecutively add each information nugget $n \in N$ to an empty hierarchy with a greedy strategy in order to maximize $\frac{1}{3}\sum_{i=1}^{3} HO(H_G, H_i)$. Then, we improve the resulting hierarchy with a local optimization method: We successively remove each information nugget from H_G and insert it again at the best possible position, again maximizing $\frac{1}{3}\sum_{i=1}^{3} HO(H_G, H_i)$. We repeat this process until there are no further changes. Since this local optimization can technically run into any (possibly bad) local optima, we analyze the effects of different random seeds. For one topic cluster, we perform the gold standard construction with ten differently shuffled nugget insertion orders. The normalized hierarchical overlap to the three manually annotated hierarchies varies from 0.464 to 0.496, with a mean of 0.481 and a standard deviation of 0.010. This shows that the initial position within the result space does influence the optimization result, but the effects are small. Therefore, we run each optimization with ten different random seeds and use the result with the highest $\frac{1}{3}\sum_{i=1}^{3} HO(H_G, H_i)$ as the gold standard.
In our corpus repository, we provide the Java source code of the hierarchy annotation tool, a runnable jar-file, all manually annotated hierarchies by the three annotators, and the gold standard hierarchies per topic in XML format. The software is licensed under the GNU General Public License v3.0.

5. Conclusion and Future Work

We introduced a novel approach to construct hierarchical summarization corpora, which enables us to summarize information from large document collections in a structured way. The resulting hierarchical summaries can be viewed from two perspectives: The root nodes and main branches of each tree in the hierarchy can be considered a generic summary, while each individual tree focuses on a specific facet discussed in the document collection yielding multiple aspect-oriented summaries. Our corpus can be used in a variety of problem settings within the field of automatic summarization, including table-of-contents generation, information exploration, structuring argumentative information, but also generic and query-based summarization. The logical next step is to use our corpus to train and evaluate automatic hierarchical summarization systems. We are not aware of any other dataset which can be used to evaluate all steps of such a system. Based on our annotation tool and HIT design, our approach can be easily reused by other researchers working on similar corpora for other domains or languages.

Acknowledgements

This work has been supported by the German Research Foundation as part of the research training group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES) under grant No. GRK 1994/1.

6. Bibliographical References

Allan, J., Carterette, B., Aslam, J. A., Pavlu, V., Dachev, B., and Kanoulas, E. (2008). Million Query Track 2007 Overview. In *Proceedings of the Sixteenth Text REtrieval Conference (TREC)*, pages 85–104, Gaithersburg, MD, USA.

Artstein, R. and Poesio, M. (2008). Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

Baumel, T., Cohen, R., and Elhadad, M. (2016). Topic Concentration in Query Focused Summarization Datasets. In *Proceedings of the 30th National Conference on Artificial Intelligence (AAAI)*, pages 2573–2579, Phoenix, AZ, USA.

Benikova, D., Mieskes, M., Meyer, C. M., and Gurevych, I. (2016). Bridging the gap between extractive and abstractive summaries: Creation and evaluation of coherent extracts from heterogeneous sources. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1039–1050, Osaka, Japan.

Bigham, J. P., Bernstein, M. S., and Adar, E. (2015). Human-computer interaction and collective intelligence. In *Handbook of Collective Intelligence*, pages 57–84. Cambridge: MIT Press.

Cheng, J., Teevan, J., Iqbal, S. T., and Bernstein, M. S. (2015). Break It Down: A Comparison of Macro- and Microtasks. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI)*, pages 4061–4064, Seoul, Republic of Korea.

Christensen, J., Soderland, S., Bansal, G., and Mausam. (2014). Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computlational Linguistics (ACL)*, pages 902–912, Baltimore, MD, USA.

Chua, F. C. T. and Asur, S. (2013). Automatic Summarization of Events from Social Media. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pages 81–90, Cambridge, MA, USA.

Erbs, N., Gurevych, I., and Zesch, T. (2013). Hierarchy Identification for Automatically Generating Table-of-Contents. In *Proceedings of 9th Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 252–260, Hissar, Bulgaria.

Euzenat, J. and Shvaiko, P. (2007). *Ontology Matching*. Berlin/New York: Springer.

Falke, T. and Gurevych, I. (2017). Bringing Structure into Summaries: Crowdsourcing a Benchmark Corpus of Concept Maps. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2969–2979, Copenhagen, Denmark.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Habernal, I., Sukhareva, M., Raiber, F., Shtok, A., Kurland, O., Ronen, H., Bar-Ilan, J., and Gurevych, I. (2016). New Collection Announcement: Focused Retrieval Over the Web. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 701–704, Pisa, Italy.

Hovy, D., Berg-Kirkpatrick, T., Vaswani, A., and Hovy, E. (2013). Learning Whom to Trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL/HLT)*, pages 1120–1130, Atlanta, GA, USA.

Krippendorff, K. (1995). On the reliability of unitizing contiguous data. *Sociological Methodology*, 25:47–76.

Li, P., Bing, L., and Lam, W. (2017). Reader-Aware Multi-Document Summarization: An Enhanced Model and The First Dataset. In *Proceedings of the EMNLP Workshop on New Frontiers in Summarization*, pages 91–99, Copenhagen, Denmark.

Lin, J., Roegiest, A., Tan, L., McCreadie, R., Voorhees, E., and Diaz, F. (2016). Overview of the TREC 2016 real-time summarization track. In *Proceedings of the 25th Text REtrieval Conference (TREC)*, Gaithersburg, MD, USA.

Lloret, E., Plaza, L., and Aker, A. (2013). Analyzing the capabilities of crowdsourcing services for text summarization. *Language Resources and Evaluation*, 47(2):337–369.

Maedche, A. and Staab, S. (2002). Measuring Similarity between Ontologies. In Asunción Gómez-Pérez et al., editors, *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web: 13th International Conference, EKAW 2002 Sigüenza, Spain, October 1–4, 2002 Proceedings*, pages 251–263. Berlin/Heidelberg: Springer.

Meyer, C. M., Mieskes, M., Stab, C., and Gurevych, I. (2014). DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING): System Demonstrations*, pages 105–109, Dublin, Ireland.

Nakano, M., Shibuki, H., Miyazaki, R., Ishioroshi, M., Kaneko, K., and Mori, T. (2010). Construction of Text Summarization Corpus for the Credibility of Information on the Web. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*, pages 3125–3131, Valletta, Malta.

Nenkova, A., Passonneau, R., and McKeown, K. (2007). The Pyramid Method: Incorporating Human Content Selection Variation in Summarization Evaluation. *ACM Trans. Speech Lang. Process.*, 4(2), May.

Nenkova, A. (2005). Automatic text summarization of newswire: Lessons learned from the document understanding conference. In *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI)*, pages 1436–1441, Pittsburgh, PA, USA.

Pembe, F. and Güngör, T. (2010). A Tree Learning Approach to Web Document Sectional Hierarchy Extraction. In *Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART)*, pages 447–450, Valencia, Spain.

Trauzettel-Klosinski, S. and Dietz, K. (2012). Standardized Assessment of Reading Performance: The New International Reading Speed Texts IReST. *Investigative Opthalmology & Visual Science*, 53(9):5452.

Voorhees, E. M. (2004). Overview of the TREC 2003 Question Answering Track. In *Proceedings of The Twelfth Text REtrieval Conference (TREC)*, pages 54–68, Gaithersburg, MD, USA.

Zechner, K. (2002). Automatic Summarization of Open-Domain Multiparty Dialogues in Diverse Genres. *Computational Linguistics*, 28(4):447–485.

Zhang, A. X., Verou, L., and Karger, D. (2017). Wikum: Bridging Discussion Forums and Wikis Using Recursive Summarization. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW)*, pages 2082–2096, Portland, OR, USA.

Zopf, M., Peyrard, M., and Eckle-Kohler, J. (2016). The Next Step for Multi-Document Summarization: A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1535–1545, Osaka, Japan.

A New Annotated Portuguese/Spanish Corpus for the Multi-Sentence Compression Task

Elvys Linhares Pontes[1], **Juan-Manuel Torres-Moreno**[1,2], **Stéphane Huet**[1], **Andréa Carneiro Linhares**[3]
[1]CERI/LIA, Université d'Avignon et des Pays de Vaucluse, Avignon, France
[2]École Polytechnique de Montréal, Montréal, Canada
[3]Universidade Federal do Ceará, Sobral-CE, Brasil
{elvys.linhares-pontes, juan-manuel.torres, stephane.huet}@univ-avignon.fr
andrea.linhares@ufc.br

Abstract

Multi-sentence compression aims to generate a short and informative compression from several source sentences that deal with the same topic. In this work, we present a new corpus for the Multi-Sentence Compression (MSC) task in Portuguese and Spanish. We also provide on this corpus a comparison of two state-of-the-art MSC systems.

Keywords: Annotated Corpus, Multi-Sentence Compression, Multilingual Corpus.

1. Introduction

Among the various applications of Natural Language Processing, Automatic Text Summarization (ATS) aims at summarizing one or more texts automatically. Summarization systems identify relevant data and create a summary from key information. The (Multi-)Sentence Compression task can be seen as a subproblem of ATS with the objective to generate a shorter, informative and correct sentence from source sentence(s).

In many cases, state-of-the-art NLP systems are evaluated with experiments restrained to the English language, in part because there are a lot of available English resources for most NLP tasks. As regards Multi-Sentence Compression (MSC), the available resources are unfortunately limited; to our knowledge, only one dataset is freely available and it is confined to the French language (Boudin and Morin, 2013).

In this work, we present a new annotated corpus in the Portuguese and Spanish languages for the MSC task. Using this corpus, we evaluate two state-of-the-art systems and show that the use of several languages leads to more mitigated results on the superiority of one system than the use of the French corpus alone.

The remainder of this paper is organized as follows. In Section 2, we characterize MSC with respect to related tasks from the perspective of the available corpora. Section 3 describes the creation and the features of our corpus. In Section 4 we analyze the results achieved by state-of-the-art methods using our dataset. Finally, conclusions are set out in Section 5.

2. Related Work

Sentence Compression (SC) aims at producing a reduced grammatically correct sentence from a source sentence. SC can be used in the context of the abstractive summarization of documents, the generation of article titles or the simplification of complex sentences, using diverse methods (optimization, syntactic structure, deletion of words and/or generation of sentences). The corpora for SC can be divided in two categories: deletion-based and summarization-based SC.

In the case of SC by deletion of words, sentences are compressed by removing irrelevant words (Filippova et al., 2015; Ive and Yvon, 2016). Knight and Marcu (2002) developed a SC corpus by aligning abstracts and sentences extracted from the Ziff-Davis corpus, which is a collection of newspaper articles announcing computer products. Clarke and Lapata (2008) provided two manually created two-reference corpora for deletion-based compression. Filippova and Altun (2013), and Filippova et al. (2015) extracted and released deletion-based compressions by aligning news headlines to the first sentences. Finally, Ive and Yvon (2016) developed an English-French parallel corpus for the compression and simplification tasks.

SC by generations of sentences analyzes a whole sentence and generates a new shorter sentence with the core information of the source sentence (Rush et al., 2015; Ganitkevitch et al., 2011; Cohn and Lapata, 2008; Toutanova et al., 2016). Ganitkevitch et al. (2011) created a corpus of compression paraphrases composed of parallel English-English sentences obtained from multiple reference translations. Rush et al. (2015) produced compression pairs made up of the headline of each article and its first sentence; they released their code to extract data from the annotated Gigaword (Graff et al., 2011). Cohn and Lapata (2008) and Toutanova et al. (2016) describe two manually created abstractive compression corpora that are publicly available. The dataset presented in Cohn and Lapata (2008) comprises a single-reference sentence pairs for abstractive summary, while the corpus developed by Toutanova et al. (2016) has multiple references for short paragraph compressions.

Multi-Sentence Compression (MSC), also known as Multi-Sentence Fusion, is a variation of SC. MSC aims at analyzing a cluster of similar sentences to generate a new sentence, which is shorter than the average length of source sentences and has the key information of the cluster (Barzilay and McKeown, 2005; Filippova, 2010). MSC enables summarization and question-answering systems to gener-

Characteristics	French		Portuguese		Spanish	
	Source	Reference	Source	Reference	Source	Reference
#tokens	20,224	2,362	17,998	1,425	30,588	3,694
#vocabulary (tokens)	2,867	636	2,438	533	4,390	881
#sentences	618	120	544	80	800	160
avg. sentence length (tokens)	33.0	19.7	33.1	17.8	38.2	23.1
type-token ratio	38.8%	50.1%	33.7%	67.9%	35.2%	43.4%
sentence similarity [0,1]	0.46	0.67	0.51	0.59	0.47	0.64

Table 1: Statistics of the corpora.

ate outputs combining fully formed sentences from one or several documents. Various corpora have been developed for MSC and are composed of clusters of similar sentences from different source news in English, French, Spanish or Vietnamese languages (Barzilay and McKeown, 2005; Filippova, 2010; Boudin and Morin, 2013; Thadani and McKeown, 2013; Luong et al., 2015). Filippova's corpus as well as Boudin and Morin's contain clusters of similar sentences, each cluster composed of at least 7 or 8 sentences, whereas the datasets introduced in (McKeown et al., 2010) and (Luong et al., 2015) have only a pair of source sentences per cluster. McKeown et al. (2010) collected 300 English sentence pairs taken from newswire clusters using Amazon's Mechanical Turk. Likewise, the dataset built by Luong et al. (2015) contains 250 Vietnamese sentences divided into 115 groups of similar sentences with 2 sentences per group. Thadani and McKeown (2013) presented an English corpus with 1,858 clusters having between 2 and 4 sentences; this dataset was built using automatic methods from annotations made for the DUC[1] and TAC[2] evaluations. The corpora presented in (McKeown et al., 2010), (Boudin and Morin, 2013) and (Luong et al., 2015) are publicly available, but among these three datasets only the second one is more suited to multi-document summarization or question-answering tasks because the documents to analyze are usually composed of many similar sentences.

3. Dataset Description

We introduce a novel annotated corpus collected from Portuguese and Spanish Google News.[3] This corpus is composed of clusters of similar sentences along with reference compressions for each cluster. The data are described in the following subsections. Table 1 summarizes the characteristics of the corpus and Table 2 shows a small example of our Portuguese dataset.

3.1. Source Sentences

In keeping with the methodology introduced by Filippova (2010), we collected links from Google News in Spanish and Portuguese between July and September 2016. These links redirect international news sites in Spanish (*La Jornada, Milenio, El Economista, BBC Mundo, El Colombiano, El País, CNN en español*, etc.) and in Portuguese (*G1, Uol Notícias, Estadão, O Globo, etc.*). Each cluster is composed of related sentences describing a specific event and was chosen among the first sentence from different articles about Science, Sports, Economy, Health, Business, Technology, Accidents/Catastrophes, General Information and other subjects. During the collection period, sentences were gathered among news threads that had at least 8 different sources. The source sentences of each cluster were manually selected so that they best describe the news, while sentences dealing with less relevant information were discarded. Each source sentence is composed of at least 8 tokens and a verb. In order to ensure the variability of source sentences inside a cluster, we removed all duplicated sentences, by assuming that sentences were too similar when the cosine similarity[4] computed from one-hot vectors was higher that 0.8. We used the TreeTagger system[5] to tag the source sentences with Parts-of-Speech.

3.2. Reference Compressions

Like in (Filippova, 2010; Boudin and Morin, 2013), reference compressions are edited by human annotators, all native Portuguese or Spanish speakers, who analyzed the most relevant facts of a cluster and generated a condensed sentence of this cluster. We suggested that the annotators should use the same vocabulary and n-grams as the source sentences and only select the most relevant information about the topic. We also recommended that they should generate compressions that are shorter than the length average of the source sentences. The following sections provide details about the Portuguese and Spanish parts of the corpus and, as a matter of comparison, briefly recalls the main characteristics of the French corpus built by Boudin and Morin.

3.2.1. Portuguese Dataset

The Portuguese corpus is composed of 40 clusters. Each cluster has at least 10 similar sentences by topic and 2 reference compressions made by 2 human annotators. This corpus contains 17,998 tokens and has a vocabulary of 2,438 tokens. Source sentences have an average of 33.1 tokens per sentence with a standard deviation of 9.9 tokens. The Type-Token Ratio (TTR) indicates the reuse of tokens in the cluster and is defined by the number of unique tokens divided by the number of tokens in each cluster; the lower

[1] http://duc.nist.gov

[2] http://www.nist.gov/tac

[3] The Spanish and Portuguese MSC datasets are freely available, under GPL license on the DOI website: http://dev.termwatch.es/~fresa/CORPUS/MSF2/.

[4] The cosine similarity between two vectors u and v associated with two sentences is defined by $\frac{u \cdot v}{||u|| \, ||v||}$ in the [0,1] range.

[5] Website: http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

Source sentences :
A Tesla fez uma oferta de compra à empresa de serviços de energia solar SolarCity por mais de 2300 milhões de euros.
A Tesla Motors , fabricante de carros elétricos , anunciou aquisição da SolarCity por US$ 2,6 bilhões .
A fabricante de carros elétricos e baterias Tesla Motors disse nesta segunda-feira (1) que chegou a um acordo com a SolarCity para comprar a instaladora de painéis solares por US$ 2,6 bilhões , em um grande passo do bilionário Elon Musk para oferecer aos consumidores um negócio totalmente especializado em energia limpa , informou a Reuters .
Reference compressions :
A Tesla Motors anunciou acordo para comprar a SolarCity por US$ 2,6 bilhões .
A fabricante Tesla Motors vai adquirir a instaladora de painéis solares da SolarCity .

Table 2: Small example of our Portuguese dataset.

the TTR, the greater the reuse of tokens in the cluster. The sentence similarity represents the average cosine similarity of the sentences in a cluster. Using these metrics, references have an average length of 17.8 tokens and a standard deviation of 1.5 tokens, while the Portuguese source corpus has a TTR of 33.7%. The Portuguese annotators generated the compressions with a TTR of 67.9% and a sentence similarity of 0.59. Finally, the average compression ratio between the reference and source sentences is 54%.

3.2.2. Spanish Dataset

The Spanish part is also composed of 40 clusters. It has 30,588 tokens and a vocabulary of 4,390 tokens. Each cluster has 20 similar sentences on the same topic and 4 reference compressions made by 4 human annotators. Source sentences have an average of 38.2 tokens per sentence with a standard deviation of 10.7 tokens and an average TTR of 35.2%. Reference compressions contain the same vocabulary as source sentences while keeping an average size of 23.1 tokens, a standard deviation of 2.4 tokens and a TTR of 43.4%. The sentence similarity between the compressions is 0.64. The average compression rate is 61%.

3.2.3. French Dataset

We used in the following experiments the French corpus developed by Boudin and Morin (2013). This corpus also has 40 clusters composed of 618 sentences (33 tokens on average). The clusters are composed of 15 sentences on average and the TTR of the corpus is 38.8%. Reference compressions have a compression rate of 60%.

4. Experimental Evaluation

We used our corpus to provide a more thorough evaluation of state-of-the-art approaches for MSC than the study on the French corpus alone. We tested on our dataset a simple baseline, as well as (Filippova, 2010) and (Boudin and Morin, 2013) methods. Filippova modeled clusters of similar sentences as Word Graphs based on the cohesion of tokens and their Part-of-Speech (PoS). Inspired by the good results of the Filippova's method, Boudin and Morin used the TextRank method as a re-rank method to analyze the sentences generated by Filippova's method in order to produce well punctuated and hopefully more informative compressions. The baseline system creates a Word Graph (WG) like Filippova's method, but this time all arcs have the same weight. Then, the system generates a compression represented by the shortest path in the WG that has at least 8 tokens. Algorithms were implemented using the Python programming language and the takahe[6] library.

4.1. Automatic and Manual Metrics

The most important features of MSC are informativeness and grammaticality. Informativeness is the percentage of the main information retained in the compression, while grammaticality analyzes whether a sentence is correct or not.

References are assumed to contain the most important information. Thus we calculated informativeness scores based on the common information between the output of the MSC system and the references using ROUGE (Lin, 2004). In particular, we used the f-measure metrics ROUGE-1, ROUGE-2 and ROUGE-SU4. Like in Boudin and Morin (Boudin and Morin, 2013), ROUGE metrics are calculated using stop words removal and stemming.[7]

We also led a manual evaluation with 4 native speakers for each language. The native speakers of each language judged the compression in two aspects: informativeness and grammaticality. In the same way as (Filippova, 2010; Boudin and Morin, 2013), the native speakers evaluated the grammaticality in a 3-point scale: 0 point for an ungrammatical compression, 1 point for compression with minor mistakes; and 2 points for a correct compression. The informativeness evaluation process is similar for grammaticality: 0 point if the compression is not related to the main topic, 1 point if the compression misses some relevant information and 2 points if the compression conveys the gist of the main event.

4.2. Results with Automatic Metrics

Table 3 shows f-score ROUGE scores for the French, Portuguese and Spanish datasets.[8] Boudin and Morin's system generated better compressions with higher ROUGE scores than Filippova's and the baseline for all datasets.

[6]Website: `http://www.florianboudin.org/publications.html`

[7]http://snowball.tartarus.org/

[8]Although we used the same system and data as (Boudin and Morin, 2013) for the French corpus, we were not able to reproduce exactly their results. The ROUGE scores given in their article are close to ours for their system: 0.6568 (ROUGE-1), 0.4414 (ROUGE-2) and 0.4344 (ROUGE-SU4), but using Filippova's system we measured higher scores than them: 0.5744 (ROUGE-1), 0.3921 (ROUGE-2) and 0.3700 (ROUGE-SU4).

Method	French			Portuguese			Spanish		
	RG-1	RG-2	RG-SU4	RG-1	RG-2	RG-SU4	RG-1	RG-2	RG-SU4
Baseline	0.3681	0.1904	0.1758	0.3199	0.1273	0.1309	0.2700	0.0990	0.0984
Filippova (2010)	0.6384	0.4423	0.4297	0.5388	0.2971	**0.2938**	0.5004	0.2983	**0.2847**
Boudin and Morin (2013)	**0.6674**	**0.4672**	**0.4602**	**0.5532**	**0.3029**	0.2868	**0.5140**	**0.2960**	0.2801

Table 3: ROUGE f-scores measured on the French, Portuguese and Spanish datasets. The best ROUGE results are in bold.

Table 4 provides statistics on the length and the compression ratio of the sentences generated by the systems. The baseline system output the shortest compressions, which translated into the worst ROUGE scores. For the three tested datasets, Filippova's method generated shorter compressions with a smaller standard deviation than Boudin and Morin's system. Let us note that for this last system the lengths of the outputs are less regular across the three languages.

The Portuguese and Spanish languages derive from Latin and are closely related languages. However, they differ in many details of their grammar and lexicon. Moreover, the datasets produced for the three languages are unlike according to several features. First, our corpus contains a smaller (Portuguese corpus) and a larger (Spanish corpus) dataset in terms of sentences than the original French corpus. Besides, the compression rates of the three datasets (see Section 3.) indicates that the Portuguese source sentences have more irrelevant tokens. The sentence similarity (Table 1, last line) describes the variability of sentences in the source sentences and in the references, and reflects here that the sentences are slightly more diverse for the Portuguese corpus. It can be noticed that the references are more similar too each other than source sentences since they only retain the main information. Finally, the French corpus has a TTR of 38.8% whereas the Portuguese and Spanish datasets have TTRs of 33.7% and 35.2%, respectively.

The baseline system generated the shortest compression because all arcs of the WG have the same weights. However, this system analyzes neither the grammaticality nor the most used n-grams in the clusters. Consequently, the baseline system generated compressions with the worst ROUGE scores.

4.3. Human Evaluation

ROUGE only analyzes the overlapping between the candidate compression and the references. Since this analysis is not reliable enough, we led a further manual evaluation to study the informativeness and grammaticality of compressions, as described in Section 4.1.. Given the poor results of the baseline with ROUGE, we only analyzed the Filippova's and Boudin and Morin's methods (Table 5).

We measured inter-rater agreement on the judgments we collected, obtaining values of Fleiss' kappa of 0.418, of 0.305 and 0.364 for French, Portuguese and Spanish respectively. These results show that human evaluation is rather subjective. Questioning evaluators on how they proceed to rate sentences reveals that they often made their choice by comparing outputs for a given cluster. As the differences of the grammaticality and the informativeness scores for the methods are not statistically significant, we move our investigation on the average and standard deviation of the results. Both methods generated compressions of good quality (scores higher than 1) for all datasets, especially for the French and the Portuguese parts where scores above 1.5 for grammaticality and above 1.2 for informativeness were obtained. Filippova's method generated more correct compressions (except for the Portuguese corpus where both methods obtained almost the same results), which shows that the re-ranking step tends to moderately deteriorate grammaticality. By contrast, Boudin and Morin's method consistently improves informativeness, which validates the interest of integrating the analysis of key phrases inside candidate compressions. This reranking method combines the cohesion score of Filippova and the relevance of key phrases[9] to generate more informative compression. This method selects the path of Word Graph that has relevant key phrases even if this path has a lower cohesion quality.

All in all, Boudin and Morin's method generated more informative but also longer compressions than Filippova's, CR showing a relative increase of 18% between both systems (Table 4).

5. Conclusion and Future Work

Multi-Sentence Compression aims to generate a short informative text summary from several sentences with related and redundant information. This task can be used in the domain of multi-document summarization or question answering to provide more informative and concise texts.

In this paper, we presented a new annotated corpus in two languages that extends the French data made available in (Boudin and Morin, 2013). We also compared two state-of-the art systems on this new dataset. We hope this corpus will help the NLP community to develop and validate multi-language methods for multi-sentence compression.

In order to extend the multi-language resources to more diverse languages, we plan to create a similar MSC dataset for Arabic. We also want to use our corpus to test other competitive MSC systems, such as the one based on integer linear programming we introduced in (Linhares Pontes et al., 2016).

6. Acknowledgments

This work was partially financed by the European Project CHISTERA-AMIS ANR-15-CHR2-0001.

[9]Key phrases are multi-word phrases composed of the syntactic pattern $(ADJ)^*(NPP|NC)^+(ADJ)^*$, which ADJ are adjectives, NPP are proper nouns and NC are common nouns. French, Portuguese and Spanish have similar syntactic patterns.

Method	French		Portuguese		Spanish	
	Length	CR	Length	CR	Length	CR
Baseline	9.6 ± 1.5	29%	9.5 ± 1.2	29%	9.1 ± 0.3	24%
Filippova (2010)	16.9 ± 5.1	51%	17.3 ± 5.3	52%	16.5 ± 6.4	43%
Boudin and Morin (2013)	19.7 ± 6.9	59%	22.9 ± 6.3	69%	23.4 ± 8.4	61%

Table 4: Length (average and standard deviation of tokens) and compression ratio (CR) of system outputs.

Method	French		Portuguese		Spanish	
	Gram.	Info.	Gram.	Info.	Gram.	Info.
Filippova (2010)	$\mathbf{1.65 \pm 0.58}$	1.25 ± 0.76	1.61 ± 0.64	1.51 ± 0.66	$\mathbf{1.51 \pm 0.69}$	1.02 ± 0.72
Boudin and Morin (2013)	1.56 ± 0.62	$\mathbf{1.48 \pm 0.68}$	$\mathbf{1.66 \pm 0.62}$	$\mathbf{1.70 \pm 0.59}$	1.30 ± 0.76	$\mathbf{1.16 \pm 0.82}$

Table 5: Manual evaluation of compressions (ratings are expressed on a scale of 0 to 2). All results are statistically equivalent.

7. Bibliographical References

Barzilay, R. and McKeown, K. R. (2005). Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328, September.

Boudin, F. and Morin, E. (2013). Keyphrase extraction for N-best reranking in multi-sentence compression. In *NAACL*, pages 298–305.

Clarke, J. and Lapata, M. (2008). Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research (JAIR*, 31:399–429.

Cohn, T. and Lapata, M. (2008). Sentence compression beyond word deletion. In *COLING*, pages 137–144.

Filippova, K. and Altun, Y. (2013). Overcoming the lack of parallel data in sentence compression. In *EMNLP*, pages 1481–1491.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence compression by deletion with LSTMs. In *EMNLP*, pages 360–368.

Filippova, K. (2010). Multi-sentence compression: Finding shortest paths in word graphs. In *COLING*, pages 322–330.

Ganitkevitch, J., Callison-Burch, C., Napoles, C., and Durme, B. V. (2011). Learning sentential paraphrases from bilingual parallel corpora for text-to-text generation. In *EMNLP*, pages 1168–1179.

Ive, J. and Yvon, F. (2016). Parallel sentence compression. In *COLING, Technical Papers*, page 1503–1513.

Knight, K. and Marcu, D. (2002). Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, 139(1):91–107.

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In *Workshop Text Summarization Branches Out (ACL'04)*, pages 74–81.

Linhares Pontes, E., Gouveia da Silva, T., Linhares, A. C., Torres-Moreno, J.-M., and Huet, S. (2016). Métodos de otimização combinatória aplicados ao problema de compressão multifrases. In *Anais do XLVIII Simpósio Brasileiro de Pesquisa Operacional (SBPO)*, pages 2278–2289.

Luong, A. V., Tran, N. T., Ung, V. G., and Nghiem, M. Q. (2015). Word graph-based multi-sentence compression: Re-ranking candidates using frequent words. In *Seventh International Conference on Knowledge and Systems Engineering (KSE)*, pages 55–60.

McKeown, K., Rosenthal, S., Thadani, K., and Moore, C. (2010). Time-efficient creation of an accurate sentence fusion corpus. In *HLT-NAACL*, pages 317–320.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *EMNLP*, pages 379–389.

Thadani, K. and McKeown, K. (2013). Supervised sentence fusion with single-stage inference. In *Sixth International Joint Conference on Natural Language Processing, IJCNLP*, pages 1410–1418.

Toutanova, K., Brockett, C., Tran, K. M., and Amershi, S. (2016). A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In *EMNLP*, pages 340–350.

8. Language Resource References

Boudin, Florian and Morin, Emmanuel. (2013). *Keyphrase Extraction for N-best Reranking in Multi-Sentence Compression*. NAACL (2013). Available on https://github.com/boudinfl/lina-msc.

Graff, David and Cieri, Christopher and Kong, Junbo and Chen, Ke and Maeda, Kazuaki. (2011). *English Gigaword*. Linguistic Data Consortium, 5th, ISLRN 911-942-430-413-0.

Live Blog Corpus for Summarization

Avinesh P.V.S., Maxime Peyrard, Christian M. Meyer
Research Training Group AIPHES and UKP Lab
Computer Science Department, Technische Universität Darmstadt
www.aiphes.tu-darmstadt.de, www.ukp.tu-darmstadt.de

Abstract

Live blogs are an increasingly popular news format to cover breaking news and live events in online journalism. Online news websites around the world are using this medium to give their readers a minute by minute update on an event. Good summaries enhance the value of the live blogs for a reader but are often not available. In this paper, we study a way of collecting corpora for automatic live blog summarization. In an empirical evaluation using well-known state-of-the-art summarization systems, we show that live blogs corpus poses new challenges in the field of summarization. We make our tools publicly available to reconstruct the corpus to encourage the research community and replicate our results. `https://github.com/UKPLab/lrec2018-live-blog-corpus`

Keywords: Live blogs, Summarization Corpus, Corpus Construction, Focused Crawling, Online Journalism

1. Introduction

A live blog is a dynamic news article providing a rolling textual coverage of an ongoing event. It is a single article continuously updated by one or many journalists with timestamped micro-updates typically displayed in chronological order. Live blogs can contain a wide variety of media, including text, video, audio, images, social media snippets and links. At the end of the broadcasting, a journalist usually summarizes the main information about the event. For more extended events, journalists may also write intermediate summaries. Figure 1 and 2 show an example live blog provided by the *BBC* on "Last day of Supreme Court Brexit Case" and *The Guardian* on "US elections 2016 campaign". The timestamped information snippets are on the right, the human-written bullet-point summary is at the top left.

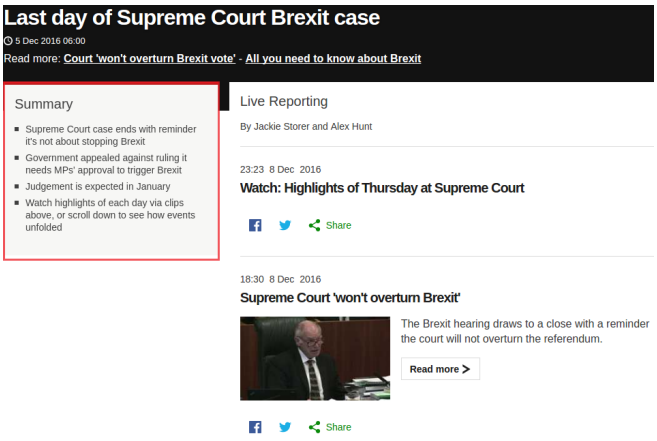


Figure 1: BBC.com live blog on "Last day of Supreme Court Brexit Case"

In the last decade, live-blogging has become very popular. It is commonly used by major news organizations, such as the *BBC*, *The Guardian* or *The New York Times*. Several different kinds of events are regularly covered by live blogs, including sport games, elections, ceremonies, protests, conflicts and natural disasters. Thurman and Schapals (2017, p.1) report a journalist's view that "live blogs have transformed the way we think about news, our sourcing, and everything". Thanks to this new journalistic trend, many live blogs – and their human-written summaries – are available online and new ones are generated every day.

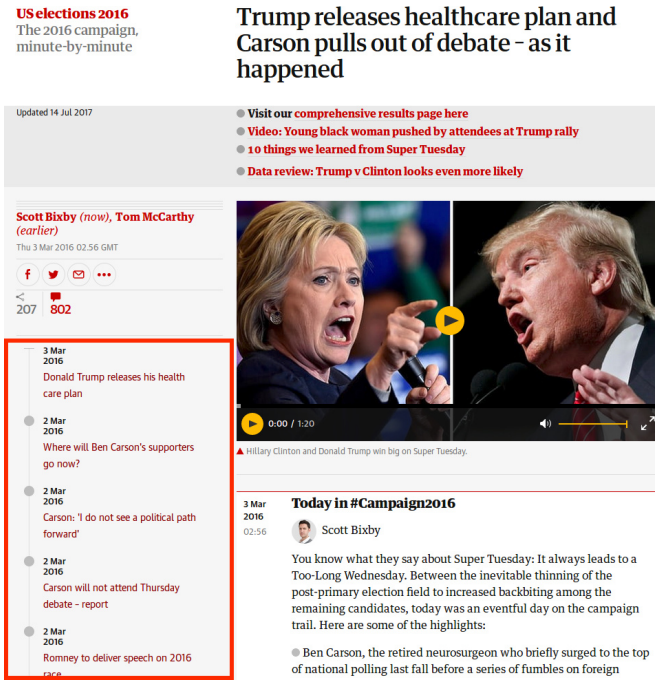


Figure 2: TheGuardian.com live blog on "US elections 2016 campaign"

In this work, we propose to leverage this data and investigate the task of automatic live blog summarization by crawling a new dataset. Live blog summarization has more direct applications in Journalism than the traditional but rather artificial tasks of a single document and multi-document summarization. Systems capable of summarizing live streams of heterogeneous content can be directly beneficial to users and even assist journalists during their daily work.

However, this new task also comes with new challenges. Live blogs are a list of short snippets of heterogeneous information and they do not form one coherent piece of text. The non-cohesive snippets make the task different from sin-

gle document summarization. Furthermore, most single documents are easily summarized by the baseline extracting the first few sentences. Such an approach is not effective for live blogs due to their heterogeneity and chronological order. The snippets are typically small, focused, numerous and rarely redundant which contrasts with the well-studied task of multi-document summarization. The topic is continually shifting and many sub-topics may arise and become central at some point. This even differs from the single topic shift found in classical update summarization tasks. For example, the live blog in Figure 1 on "Last day of Supreme Court Brexit case" consists of topic shift across "Supreme court judgment", "Government appeal", "Opinions of MP's on Brexit" and others. Moreover, when summarizing a live blog, one has to account for the whole past and all sub-topics previously discussed, which differs from real-time summarization setups like TREC.

We focus on two online news websites for acquiring live blogs, the *BBC*[1] and *The Guardian*[2], because they contain a lot of easily accessible live blogs that we automatically crawl and process.

In summary, our contributions are:

- We introduce a new task: live blog summarization.
- We suggest a pipeline to collect and extract live blogs with human-written summaries from two major online newspapers and release it for the community[3].
- We benchmark the dataset with commonly used summarization methods to stimulate further research into this challenging task.

The rest of the paper is structured as follows: Section 2. details existing summarization corpora and related works. Section 3. discusses our approach to collect live blogs from *BBC* and *The Guardian*, followed by a discussion on the statistics and properties of our live blog corpus in section 4.. The performance of well-established summarization baselines on this new dataset is discussed in section 5., followed by the conclusion and future work.

2. Related Work

In this section, we describe previous works related to summarization corpora. They were focused on single and multi-document summarization, update summarization, and real-time summarization. We are not aware of any previous work on live blog summarization.

Single and multi-document summarization. The most widely used summarization datasets have been published in the Document Understanding Conference[4] (DUC) series. In total, there are 139 document clusters with 376 human-written reference summaries across DUC '01, '02, and '04. Although the research community has often used these corpora, creating the manual summaries is time-consuming and labor-intensive.

Large datasets typically exist for single document summarization tasks, for example, the ACL Anthology Reference Corpus (Bird et al., 2008) and the CNN/Daily Mail dataset (Hermann et al., 2015). The latter contains large pairs of 312k online news articles and multi-sentence summaries used for neural summarization approaches (Nallapati et al., 2016; See et al., 2017). However, their dataset contains only one source document, whereas live blogs have a larger number of information snippets, typically more than 100.

Another recent work uses social media's reactions on Twitter to create large-scale multi-document summaries for news (Lloret and Palomar, 2013; Cao et al., 2016). Cao et al. (2016) use hashtags to cluster the documents into the same topic and use tweets with hyperlinks to generate optimal reference summaries. Their corpus consists of 204 document clusters with 1,114 documents and 4,658 reference tweets. Although this approach uses social media information to create a summarization corpus, they produce synthetic summaries, which are not written by a human. Moreover, they only use the corpus for training supervised learning approaches and not for evaluating summarization systems.

Other multi-document summarization datasets focus on heterogeneous sources (Zopf et al., 2016; Benikova et al., 2016; Nakano et al., 2010), multiple languages (Giannakopoulos et al., 2015), and reader-aware multi-document summaries (Li et al., 2017), which jointly aggregate news documents and reader comments.

Update summarization. After the DUC series, the Text Analysis Conference[5] (TAC) series ('08, '09) introduced the update summarization task (Dang and Owczarzak, 2008). In this task, two summaries are provided for two sets of documents and the summary of the second set of documents is an update of the first set. Although the importance of text to be included in the summary solely depends on the novelty of the information, the task usually observes only a single topic shift. In live blogs, however, there are multiple sub-topics and the importance of the sub-topics changes over time.

Real-time summarization. Real-time summarization began at the Text REtrieval Conference[6] (TREC) 2016 and represents an amalgam of the microblog track and the temporal summarization track (Lin et al., 2016). In real-time summarization, the goal is to automatically monitor the stream of documents to keep a user up to date on topics of interest and create email digests that summarize the events of that day for their interest profile. The drawback of this task is that they have a predefined time frame for evaluation due to the real-time constraint, which makes the development of systems and replicating results arduous. Note that live blog summarization is very similar to real-time summarization, as the real-time constraint also holds true for live blogs if the summarization system is applied to the stream of snippets. Moreover, the Guardian live blogs do consist of updated and real-time summaries, but this requires different real-time crawling strategies which are out of the scope of this work.

[1] http://www.bbc.com
[2] https://www.theguardian.com
[3] https://github.com/UKPLab/lrec2018-live-blog-corpus
[4] http://duc.nist.gov/
[5] http://www.nist.gov/tac/
[6] http://trec.nist.gov/

3. Corpus Construction

In this section, we describe the three steps to construct our live blogs summarization corpus: (1) live blog crawling yielding a list of URLs, (2) content parsing and processing, where the documents and corresponding summaries with the metadata are extracted from the URLs and stored in a JSON format, and (3) live blog pruning as a final step for creating a high-quality gold standard live blog summarization corpus.

Live blog Crawling. On the Guardian, a frequently updated index webpage[7] references all archived live blogs. We took a snapshot of this page that provided us with 16,246 unique live blogs.

In contrast, the BBC website has no such live blog archive. Thus, we use an iterative approach similar to BootCaT (Baroni and Bernardini, 2004) as described in Algorithm 1 to bootstrap a corpus utilizing a set of seed terms extracted from ten BBC live blog links from the web. The iterative procedure starts with a small set of seed terms (K_0) and gathers new live blog links using automated Bing queries[8] by exploiting patterns (P) in live blog URLs (i.e. "site:http://www.bbc.com/news/live/[key term]" as in line 5). We collect all the valid links returned by the Bing queries (line 6) and look for new key terms in the recently retrieved live blogs (line 10). In our implementation, key terms are terms with high TF*IDF scores. The new key terms are then used in the Bing queries of the subsequent iterations. The process is repeated until no new live blogs are discovered anymore (line 7). With this process, we ran 4,000 search queries returning each around 1,000 results on average and we collected 9,931 unique URLs.

Although our method collected a majority of the live blogs in the 4,000 search queries, a more sophisticated key terms selection could minimize the search queries and maximize the unique URLs. Additionally, this methodology can be applied to other news websites featuring live blogs like *The New York Times*, *Washington Post* or *Der Spiegel*.

An important point to note is that we find the collected BBC live blog URLs predominantly cover more recent years. This usage could be due to the Bing Search API preferring recent articles for the first 100 results. To collect a broad range of news articles the queries need to be precise.

Content Parsing and Processing. Once the URLs are retrieved, we fetch the HTML content, remove the boilerplate and store the cleaned data in a JSON file.

During this step, unreachable URLs were filtered out. We discard live blogs for which we could not retrieve the summary or correctly parse the information snippets. Indeed, live blogs can have changing patterns over time rendering the automatic extraction difficult.

Parsing of BBC live blogs can be automated easily because both bullet-point summaries and information snippets follow a consistent pattern. For the Guardian, we identify several recurring patterns which cover most of the live blogs.

[7] `http://www.theguardian.com/tone/minutebyminute`

[8] `https://azure.microsoft.com/en-us/services/cognitive-services/bing-web-search-api`

Algorithm 1 Iterative Live Blog Retrieval

```
1: procedure LIVEBLOGRETRIEVAL()
2:     input: Seed terms K_0, Live blog Pattern P
3:     L_0 ← ∅
4:     for t = 1...T do
5:         Q_t ← makeQueries(K_{t-1}, P)
6:         L_t ← getLinks(Q_t)
7:         if ∪_{i=0}^{t-1} L_i = ∪_{i=0}^{t} L_i then
8:             return ∪_{i=0}^{t} L_i
9:         else
10:            K_t ← extractKeyTerms(L_t) − ∪_{i=0}^{t-1} K_i
11:        end if
12:    end for
13: end procedure
```

Dataset	Crawling	Processing	Pruning
BBC	9,931	7,307	974
Guardian	16,246	6,405	1,681

Table 1: Number of topics for BBC and the Guardian

The Guardian live blogs were in use since 2001 but were in experimental phase till 2008. Due to the lack of a specific structure or a summary during this experimental phase, we remove 10k of the crawled live blogs. However, after 2008, live blogs have had a prominent place in the editorial with a consistent structure.

We parse metadata like URL, author, date, genre, summaries and documents for each live blog using site-specific regular expressions on the HTML source files.

After this step, 7,307 live blogs remain for BBC and 6,450 for Guardian.

Live blog Pruning. To further clean the data, we decided to remove live blogs exhibiting several topics as they can be quite noisy. For example, BBC provides some live blogs covering all events happening in a given region within a given time frame (e.g., *Essex: Latest updates*). We also prune live blogs about sport games and live chats, because the summaries are based on simple templates.

We further prune live blogs based on their summaries. We first remove a sentence of a summary if it has less than three words. Then, we discarded live blogs whose summaries have less than three sentences. This is to ensure the quality of the corpus, as overly short summaries would yield a different summarization goal similar to headline generation and they are typically an indicator for a non-standard live blog layout.

After the whole pruning step, 974 live blogs remained for BBC and 1,681 for the Guardian.

Overall, 10% of the initial set of live blogs, both for BBC and Guardian remained after selective pruning. This is to ensure high-quality summaries for the live blogs. Although the pruning rejects 90% of the live blogs, the size of the live blog corpus is 20–30 times larger than the classical corpora released during DUC, TREC and TAC tasks.

Code Repository. To replicate our results and advance research in live blog summarization we publish our tools

for reconstructing the live blog corpus open-source under the Apache License 2.0. The repository consists of (a) raw and processed URLs, (b) tools for crawling live blogs, (c) tools for parsing the content of the URLs and transforming content into JSON, and (d) code for calculating baselines and corpus statistics.

4. Corpus Statistics

We compute several statistics about the corpora and report them in Table 2. The number of documents (or snippets) per topic is around 95 for BBC and 56 for the Guardian. In comparison, standard multi-document summarization datasets like DUC '04[9] and TAC '08A[10] have only 10 documents per topic.

Furthermore, we observe that snippets are quite short as there is an average of 62 words per snippet for BBC and 108 for the Guardian. Summaries are also shorter than summaries in standard datasets. Indeed, in DUC2004 and TAC2008A summaries are expected to contain 100 words. Our corpora are larger because, together, they contain 2,655 topics and 186,999 documents. With many data points, machine learning approaches become readily applicable.

Statistic	BBC	Guardian
# topics	974	1,681
# documents	92,537	94,462
# documents / topic	95.01	56.19
# words / document	61.75	107.53
# words / summary	59.48	42.23

Table 2: Corpus statistics for BBC and the Guardian

Domain Distribution. Live blogs cover a wide range of subjects from multiple domains. In Table 3, we report the distribution of different domains in our combined datasets (BBC and the Guardian). While we observe that politics, business and news are the most prominent domains, there is also a number of well-represented domains like local and international events or culture.

Heterogeneity. The resulting corpus is expected of exhibiting various levels of heterogeneity. Indeed, there contain various topics with mixed writing styles (short-to-the-point snippets vs. longer descriptive snippets). Furthermore, live blogs are subject to topic shifts which could be observed by the change in words used.

To measure this textual heterogeneity, we use information theoretic metrics on word probability distributions like it was done before in analyzing the heterogeneity of summarization corpora (Zopf et al., 2016). Based on Jensen-Shannon (JS) divergence, they defined a measure of textual heterogeneity TH for a topic T composed of documents $d_1, \cdots, d_n$ as

$$TH_{JS}(T) = \frac{1}{n} \sum_{d_i \in T} JS(P_{d_i}, P_{T \setminus d_i}) \quad (1)$$

[9] http://duc.nist.gov/duc2004

[10] https://tac.nist.gov/2008

Domain	# topics	proportion (%)
Politics	834	31.41
Business	421	15.86
General News	369	13.90
UK local events	368	13.86
International events	337	12.69
Culture	186	7.01
Science	60	2.26
Society	27	1.02
Others	53	2.00

Table 3: Corpus distribution across multiple domains for BBC and the Guardian

	BBC	Guardian	DUC '04	TAC '08A
TH_{JS}	0.5917	0.5689	0.3019	0.3188

Table 4: Average textual heterogeneity of our corpora compared to standard datasets

Here, P_{d_i} is the frequency distribution of words in document d_i and $P_{T \setminus d_i}$ is the frequency distribution of words in all other documents of the topic except d_i. The final quantity TH_{JS} is the average divergence of documents with all the others and provides, therefore, a measure of diversity among documents of a given topic.

We report the results in Table 4. To put the numbers in perspective, we also report the textual heterogeneity of the two standard summarization datasets DUC '04 and TAC '08A. These corpora were created during shared tasks and focused on multi-document news summarization. The heterogeneity in BBC and Guardian are similar and both much higher than DUC '04 and TAC '08A, meaning that our corpora contain more lexical variation inside topics.

5. Results and Analysis

In this section, we describe the automatic summarization methods and the upper bounds we compute for our live blog summarization dataset.

5.1. Baselines

As benchmark results, we employ methods that have been successfully used for both single and multi-document summarization. Some variants of them have also been applied to update summarization tasks.

TF⋆IDF (Luhn, 1958) scores sentences with the TF*IDF of their terms. The best sentences are then greedily extracted.

LexRank (Erkan and Radev, 2004) is a well-known graph-based approach. A similarity graph $G(V, E)$ is constructed where V is the set of sentences and an edge e_{ij} is drawn between sentences v_i and v_j if and only if the cosine similarity between them is above a given threshold. Sentences are then scored according to their PageRank in G.

LSA (Steinberger and Jezek, 2004) is an approach involving a dimensionality reduction of the term-document matrix via singular value decomposition (SVD). The sentences extracted should cover the most important latent topics.

Systems	BBC (L)			Guardian (L)			BBC ($2 * L$)			Guardian ($2 * L$)		
	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4	R1	R2	SU4
TF*IDF	.227	.067	.064	.153	.021	.027	.367	.115	.147	.248	.037	.065
LexRank	.276	.080	.079	.188	.029	.038	.421	.138	.176	.297	.051	.089
LSA	.212	.046	.052	.135	.013	.021	.341	.084	.123	.220	.024	.051
KL	.267	.086	.080	.178	.026	.035	.397	.132	.165	.272	.041	.076
ICSI	.302	.104	.091	.210	.046	.046	.461	.176	.201	.322	.071	.101
UB-1	**.514**	.273	.218	**.422**	.177	.145	**.754**	.388	.435	**.640**	.256	.304
UB-2	.494	**.312**	.210	.389	**.230**	.137	.709	**.453**	.419	.584	**.334**	.277

Table 5: ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-SU4 (SU4) scores of multiple systems compared to the extractive upper bounds for ROUGE-1 (UB-1) and ROUGE-2 (UB-2) extractive for summary lengths of L and $2 * L$

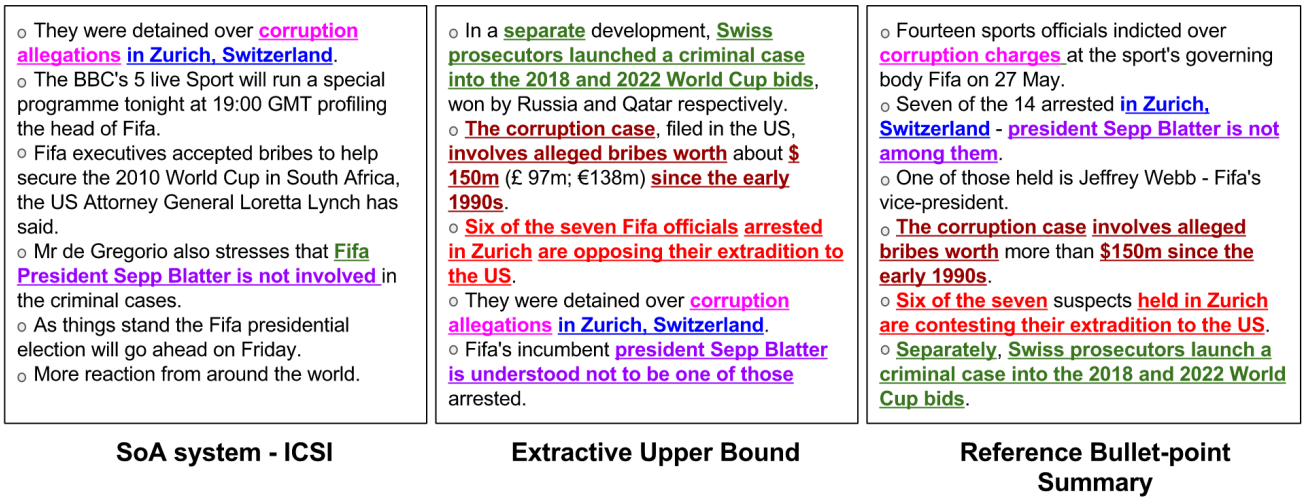


Figure 3: BBC.com live blog on "FIFA corruption inquiry"

KL-Greedy (Haghighi and Vanderwende, 2009) minimizes the Kullback-Leibler (KL) divergence between the word distributions in summary and the documents.

ICSI (Gillick and Favre, 2009) is a global linear optimization that extracts a summary by solving a maximum coverage problem considering the most frequent bigrams in the source documents. ICSI has been among the state-of-the-art MDS systems when evaluated with ROUGE (Hong et al., 2014).

5.2. Upper bound

For comparison, we compute two upper bounds. The upper bound for extractive summarization is retrieved by solving the maximum coverage of n-grams from the reference summary (Takamura and Okumura, 2010; Peyrard and Eckle-Kohler, 2016; P.V.S. and Meyer, 2017). This is cast as an Integer Linear Programming (ILP) and depends on two parameters: N, the size of n-grams considered and L, the maximum length of the summaries. In our work, we set $N = 1$ and $N = 2$ and compute the upper bound for ROUGE-1 (UB-1) and ROUGE-2 (UB-2) respectively.

5.3. Experimental Setup

We report scores for the ROUGE metrics identified by Owczarzak et al. (2012) as strongly correlating with human evaluation methods: ROUGE-1 (R1) and ROUGE-2 (R2) recall with stemming and stop words not removed. For completeness, we also report the best skip-grams matching metric: ROUGE-SU4 (SU4).

5.4. Analysis

Table 5 shows the results of benchmark summarization methods widely used in the summarization community on our live blog corpus. We explore two different summary lengths: L, length of the human-written bullet-point summary, and $2 * L$, twice the length of the human-written summary to give leeway for compensating the excessive compression ratio of the human live blog summaries. The results show the state-of-the-art ICSI system is .2 ROUGE-1 and .3 ROUGE-2 lower than the upper bounds for both BBC and the Guardian with length constraint L and $2 * L$ respectively. ICSI is only able to reach one-third of the upper bound, which emphasizes that live blog summarization is a challenging task and we need new techniques tackling live blog summarization.

Figure 3 shows the output of the ICSI system as compared to the extractive upper bound on BBC live blog on "FIFA corruption inquiry".[11] It can be seen that the ICSI system extracts sentences with most frequent concepts (e.g., FIFA, president, world cup), but misses to identify topic shifts in these information snippets. Although the information snip-

[11] http://www.bbc.com/news/live/world-europe-32897157

pets collected by the ICSI system are related to FIFA corruption, it misses capturing relative importance of the information snippets.

Additionally, factors which determine the difficulty of the summarization task are the length of the source documents and the summary (Nenkova and Louis, 2008). The input document sizes of the BBC and the Guardian are on an average 5,890 and 6,048 words, whereas the summary sizes are around 59 and 42 words respectively. Thus, the high compression ratio makes live blog summarization even more challenging.

6. Conclusion and Future Work

We introduce a new task: live blog summarization which has direct applications for journalists and news readers. Our goal is constructing a reference corpus for this new task. In this paper, we suggest a pipeline to collect live blogs with human written bullet-point summaries from two major online newspapers, which can be extended to live blogs from other news agencies like *The New York Times*, *Washington Post* or *Der Spiegel*.

We further analyze the live blog corpus and provide benchmark results for this dataset by applying commonly used summarization methods. Our results show that off-the-shelf summarization systems cannot be used, as they are far from reaching the upper bound. This calls for new solutions that take the task characteristics into account. As future work, we plan to research novel approaches to live blog summarization and investigate algorithms to identify important information from multiple topic shifts and a large number of information snippets.

Code for constructing and reproducing the live blog corpus and the automatic summarization experiments are published under the permissive Apache License 2.0 and can be obtained from `https://github.com/UKPLab/lrec2018-live-blog-corpus`.

Acknowledgments

This work has been supported by the German Research Foundation as part of the Research Training Group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES) under grant No. GRK 1994/1. We also acknowledge the useful comments and suggestions of the anonymous reviewers.

7. Bibliographical References

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping Corpora and Terms from the Web. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 1313–1316, Lisbon, Portugal.

Benikova, D., Mieskes, M., Meyer, C. M., and Gurevych, I. (2016). Bridging the gap between extractive and abstractive summaries: Creation and evaluation of coherent extracts from heterogeneous sources. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1039–1050, Osaka, Japan.

Bird, S., Dale, R., J. Dorr, B., Gibson, B., Joseph, M., Kan, M.-Y., Lee, D., Powley, B., Radev, D., and Fan Tan, Y. (2008). The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics. In *Proceedings of Language Resources and Evaluation Conference (LREC)*, Marrakech, Morocco.

Cao, Z., Chen, C., Li, W., Li, S., Wei, F., and Zhou, M. (2016). Tgsum: Build tweet guided multi-document summarization dataset. In *Proceedings of the Thirtieth Conference on Artificial Intelligence (AAAI)*, pages 2906–2912, Phoenix, AZ, USA.

Dang, H. and Owczarzak, K. (2008). Overview of the TAC 2008 update summarization task. In *Proceedings of the First Text Analysis Conference (TAC)*, pages 1–16, Gaithersburg, MD, USA.

Erkan, G. and Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality As Salience in Text Summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Giannakopoulos, G., Kubina, J., Conroy, J., Steinberger, J., Favre, B., Kabadjov, M., Kruschwitz, U., and Poesio, M. (2015). MultiLing 2015: Multilingual Summarization of Single and Multi-Documents, On-line Fora, and Call-center Conversations. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 270–274, Prague, Czech Republic.

Gillick, D. and Favre, B. (2009). A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, pages 10–18, Boulder, CO, USA.

Haghighi, A. and Vanderwende, L. (2009). Exploring Content Models for Multi-document Summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370, Boulder, CO, USA.

Hermann, K. M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching Machines to Read and Comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*, pages 1693–1701, Montreal, Canada.

Hong, K., Conroy, J., Favre, b., Kulesza, A., Lin, H., and Nenkova, A. (2014). A repository of state of the art and competitive baseline summaries for generic news summarization. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 1608–1616, Reykjavik, Iceland.

Li, P., Bing, L., and Lam, W. (2017). Reader-Aware Multi-Document Summarization: An Enhanced Model and The First Dataset. In *Proceedings of the EMNLP Workshop on New Frontiers in Summarization*, pages 91–99, Copenhagen, Denmark.

Lin, J., Roegiest, A., Tan, L., McCreadie, R., Voorhees, E., and Diaz, F. (2016). Overview of the trec 2016 real-time summarization track. In *Proceedings of The Twenty-Fifth Text REtrieval Conference (TREC)*, Gaithersburg, MD, USA.

Lloret, E. and Palomar, M. (2013). Towards automatic tweet generation: A comparative study from the text

summarization perspective in the journalism genre. *Expert Systems with Applications*, 40(16):6624–6630.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research Development*, 2:159–165.

Nakano, M., Shibuki, H., Miyazaki, R., Ishioroshi, M., Kaneko, K., and Mori, T. (2010). Construction of Text Summarization Corpus for the Credibility of Information on the Web. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)*, pages 3125–3131, Valletta, Malta.

Nallapati, R., Zhou, B., dos Santos, C., Gulcehre, C., and Xiang, B. (2016). Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany.

Nenkova, A. and Louis, A. (2008). Can you summarize this? identifying correlates of input difficulty for multi-document summarization. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 825–833, Columbus, OH, USA.

Owczarzak, K., Conroy, J. M., Dang, H. T., and Nenkova, A. (2012). An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9, Montreal, Canada.

Peyrard, M. and Eckle-Kohler, J. (2016). Optimizing an Approximation of ROUGE – a Problem-Reduction Approach to Extractive Multi-Document Summarization. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1825–1836, Berlin, Germany.

P.V.S., A. and Meyer, C. M. (2017). Joint Optimization of User-desired Content in Multi-document Summaries by Learning from User Feedback. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1353–1363, Vancouver, Canada.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1073–1083, Vancouver, Canada.

Steinberger, J. and Jezek, K. (2004). Using latent semantic analysis in text summarization and summary evaluation. In *Proceedings of the 7th International Conference on Information Systems Implementation and Modelling (ISIM)*, pages 93–100, Rožnov pod Radhoštěm, Czech Republic.

Takamura, H. and Okumura, M. (2010). Learning to generate summary as structured output. In *Proceedings of the 19th ACM international Conference on Information and Knowledge Management*, pages 1437–1440, Toronto, Canada.

Thurman, N. and Schapals, A. K. (2017). Live blogs, sources, and objectivity: The contradictions of real-time online reporting. In *The Routledge Companion to Digital Journalism Studies*, pages 283–292. London/New York: Routledge.

Zopf, M., Peyrard, M., and Eckle-Kohler, J. (2016). The Next Step for Multi-Document Summarization: A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1535–1545, Osaka, Japan.

TSix: A Human-involved-creation Dataset for Tweet Summarization

Minh-Tien Nguyen[1,2], **Dac Viet Lai**[1], **Huy-Tien Nguyen**[1], **Le-Minh Nguyen**[1]
[1]Japan Advanced Institute of Science and Technology,
1-1 Asahidai, Nomi, Ishikawa, 923-1292, Japan.
{tiennm, vietld, ntienhuy, nguyenml}@jaist.ac.jp
[2] Hung Yen University of Technology and Education, Hung Yen, Vietnam.

Abstract

We present a new dataset for tweet summarization. The dataset includes six events collected from Twitter from October 10 to November 9, 2016. Our dataset features two prominent properties. Firstly, human-annotated gold-standard references allow to correctly evaluate extractive summarization methods. Secondly, tweets are assigned into sub-topics divided by consecutive days, which facilitate incremental tweet stream summarization methods. To reveal the potential usefulness of our dataset, we compare several well-known summarization methods. Experimental results indicate that among extractive approaches, hybrid term frequency – document term frequency obtains competitive results in term of ROUGE-scores. The analysis also shows that polarity is an implicit factor of tweets in our dataset, suggesting that it can be exploited as a component besides tweet content quality in the summarization process.

Keywords: Tweet summarization, hybrid TF-IDF, dataset, corpus, annotation.

1. Introduction

The growth of micro-blogging services such as Twitter encourages users sharing their viewpoints regarding an event. For example, users following `US Election` can post their tweets (short messages with a maximum of 140 characters) on their timelines. After posting, their friends can immediately update new information about this event. Those who are out of their networks can also track the event by using the keyword-search function provided by Twitter. However, search results are usually overwhelming due to millions of returned tweets, which span for weeks. Even if the filter is enabled, digging a large number of tweets for interesting contents would be a nightmare due to their noise. These demand a topic-driven system extracting high-quality tweets for user interests.

The bottleneck of social short-text summarization is the shortage of standard datasets for evaluating summarization methods whereas well-known DUC datasets have been freely published for document summarization. For tweet summarization, authors usually create their data. For example, although (Shou et al., 2013) released a dataset including events for evaluating their method, the dataset is now inaccessible. (Imran et al., 2014) published a dataset for disaster response during the Joplin tornado collected from Twitter. Although this dataset contains more than 230,000 tweets, its lack of references challenges the evaluation.

This paper leverages tweet summarization by introducing a new dataset including six events collected from Twitter. The involvement of humans in creating gold-standard references facilitates the evaluation. To show the potential usability of our dataset, we employ hybrid term frequency – inverse document frequency (TF-IDF) for extracting important tweets. Experimental results show that the hybrid model achieves competitive ROUGE scores over baselines. We also further analyze the polarity aspect of tweets. The analysis shows that extracted tweets tend to be non-sentiment. The dataset can be publicly accessible.[1]

[1]https://goo.gl/kXBof9

2. Summarization

This section presents our proposal in creating the dataset in two steps: data creation and the summarization model including tweet scoring and selection.

2.1. Data Creation

Data collection To create the dataset, we first defined a list of topics that satisfy following conditions. Firstly, trending topics are preferred in order to collect a large amount of data from various sources. Second, they are potential for last 30 days. Thirdly, they must be impressive to news providers. Once the list of trends was filed, we assigned each topic with a list of keywords. After that, those keywords were utilized to crawl data by tracking tweet streams using the public REST APIs[2] of Twitter.

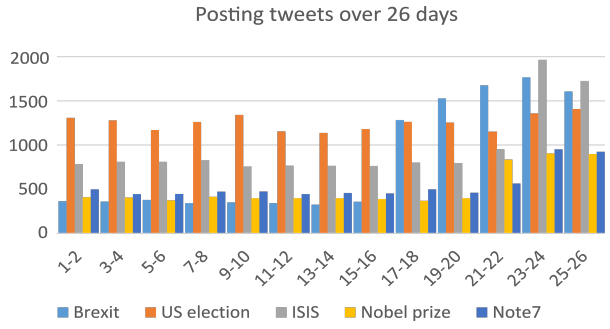


Figure 1: Tweets posted over 26 days after preprocessing.

Figure 1 plots the distribution of posting tweets over 26 days after pre-processing. Table 1 shows the statistics of the collected data and the used keywords. They can be used to estimate the quality of tweets by measuring how much important information is generated by social users in tweets.

Data segmentation Since the average number of tweets per day is not so large (around 600, Figure 1[3]), we set the period for a day. We collected tweets by their posting time to create a collection corresponding to each time

[2]https://dev.twitter.com/rest/public
[3]We plot five events due to space limitation.

Table 1: Six events, collected from Oct 10 to Nov 9, 2016.

Event	#tweets	#hashtags	Keywords
Brexit	10,978	9,705	brexit, #brexitshambles, #Brexiters, #BrexitCentral, England Europe exit.
US election	17,714	8,566	Donald Trump, Trump, Hilary Clinton, Clinton #debate, election.
ISIS	13,047	9488	ISIS, IS Syria, IS Mosul, IS Iraq, ISIS Aleppo, ISIS US, ISIS Rusia.
SS Note 7	7,362	7,465	Galaxy Note 7, #note7, #GalaxyNote7, thegalaxynote7, #SamsungGalaxyNote7.
Nobel prize	6,812	2,780	Nobel prize 2016, Nobel peace, Nobel chemistry, Nobel economy, Nobel physics.
SpaceX	4,982	2,417	Facebook SpaceX, SpaceX Explosion, Falcon 9 exploded, Falcon 9 explosion.

step. Tweets in each time step were assigned into clusters. The intuition of clustering tweets is that even the number of tweets per day is small, directly extracting a subset of these tweets may eliminate other important ones. By clustering, our goal is to keep representative tweets as many as possible. The time step can be arbitrary, e.g. per hour.
To foster the real-time aspect of a tweet summarization system, we adopted the Affinity Propagation (AP)[4] algorithm for clustering (Frey and Dueck, 2007) because traditional clustering methods such as k-means (MacQueen, 1967; Forgy, 1965) require a pre-defined number of clusters k. However, in real-time scenarios, identifying k is nontrivial (Busch et al., 2012; Nguyen et al., 2015). AP identifies a subset of data points as exemplars and forms clusters by assigning remaining data points into one of the exemplars. After clustering and eliminating days which contain a very small number of tweets (less than 10), we formed six sets corresponding to six events in 26 days and each day includes a set of clusters, which can be seen as subtopics.

Standard reference creation Once clusters have been formed, we have to create standard references for evaluation. We followed the two-stage method (Shou et al., 2013) in order to avoid tremendous human labor. In the extraction stage, since the number of tweets in each cluster is quite large, we applied three different extractive methods to create reference candidates. `Luhn` is a heuristic method for extraction (Luhn, 1958). `Lexrank` is a graph-based method, which builds a sentence similarity graph and selects important ones based on their eigenvector centrality (Erkan and Radev, 2004). `DSDR-non` bases on non-negative linear data reconstruction (He et al., 2012). The extracted tweets from the three methods form three candidate sets. In practice, suppose that n is the number of tweets in each cluster, we conditioned the number of extracted tweets is $n_{ext} = \frac{n}{2}$ if $n \leq 30$; otherwise $n_{ext} = \frac{n}{3}$.
In the selection step, we asked two annotators to select references from the candidate ones via a Web interface.[5] Each annotator reads whole candidate references in each cluster (after extracting) and estimates the importance of each candidate reference. A gold-standard reference is a tweet, which satisfies two conditions: (i) it is important in the viewpoint of each annotator regarding the event and (ii) it belongs to at least two over three candidate sets. Each cluster contains more than five and less than 25 tweets. Since the judgment of annotators is objective, therefore we kept the selected tweets from the two annotators as the references. As a result, the outputs of each extractive method have to compare to two references.[6] We show the upper bounds of ROUGE scores by using extracted tweets from the three methods in Table 2.

Table 2: Upper bound ROUGE scores.

Method	ROUGE-1	ROUGE-2	ROUGE-SU
Luhn	0.556	0.502	0.250
LexRank	0.581	0.516	0.273
DSDR-non	0.419	0.310	0.145

Data observation Unlike other types of data, tweets are created to share human thoughts and emotions under a length constraint. Consequently, people tend to put emotional words and hashtags in their tweets. The emotional words show their interests and the hashtags include important information regarding an event. We performed an observation of sentiment words and hashtag utilization as follow. We projected each tweet to a dictionary[7] of sentiment words to assess whether this tweet contains the sentiment aspect. Meanwhile, we counted the number of tweets consisting of hashtags. Figure 2 shows the observation.

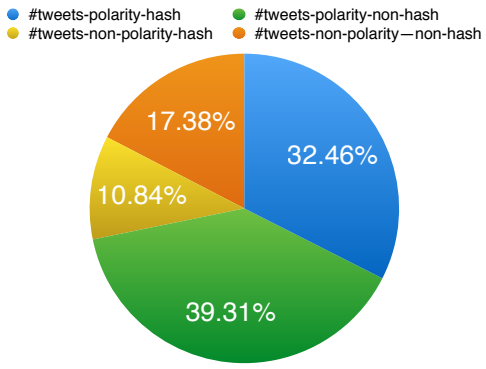


Figure 2: Hashtag and polarity observation on six datasets.

In Figure 2, the number of tweets containing polarity is considerable (around 72%). It shows that polarity analysis may potentially affect the extraction step. The number of tweets which owns the polarity aspect and includes hashtags is large (32.46%) whereas only 17.38% of tweets do not contain polarity and hashtags. Note that the number of tweets containing polarity may change if we use a classifier instead of using a dictionary.

[4]http://scikit-learn.org/stable/modules/generated/sklearn.cluster.AffinityPropagation.html

[5]https://s242-097.jaist.ac.jp/doc-sum-annotator/annotate

[6]This is similar to DUC, which includes four references from four annotators for each topic.

[7]http://mpqa.cs.pitt.edu/

2.2. Summarization with Hybrid TF-IDF

Tweet scoring Term frequency – inverse document frequency (TF-IDF) (Luhn, 1958) is a well-known method for information retrieval and text summarization.

$$TF_IDF = tf_{i,j} * \log_2 \frac{N}{df_j} \quad (1)$$

where: $tf_{i,j}$ is the frequency of term T_j in the document D_i, N is the total number of documents, and df_j is the number of documents containing the term T_j. After scoring, sentences containing terms with high weights are extracted as a summary.

Equation (1) composes TF and IDF with the logarithm to balance the effect of the IDF component. In the context of document summarization, TF-IDF has been shown advantages to select important sentences (Luhn, 1958). For tweet extraction, since tweets are informal documents; therefore, Eq. (1) exists two issues. Firstly, if we consider all tweets in a cluster as a document, Eq. (1) can compute the TF across all tweets. However, the IDF is limited due to only one document. On the other hand, we can assume that each tweet as a document to tackle the limitation of IDF. However, the TF is problematic because each tweet consists of a handful of words, hence, it receives a small TF value.

From the limitations of the traditional TF-IDF for tweet extraction, we, therefore, adopted a hybrid TF-IDF method (Inouye and Kalita, 2011). It differs the traditional one by regarding all tweets as a single document when computing TF and each tweet as a separate document when calculating IDF. Eqs. (2) - (6) present the hybrid TF-IDF model.

$$h_{TFIDF}(t) = \frac{\sum_{i=0}^{\#WordsInTweet} W(w_i)}{nf(t)} \quad (2)$$

$$W(w_i) = tf(w_i) * \log_2(idf(w_i)) \quad (3)$$

$$tf(w_i) = \frac{\#OccurencesOfWordInAllTweets}{\#WordsInAllTweets} \quad (4)$$

$$idf(w_i) = \frac{\#Tweets}{\#TweetsInWhichWordOccurs} \quad (5)$$

$$nf(t) = \#WordsInTweet \quad (6)$$

where: w_i is a term i^{th} in the tweet t, $W()$ returns the weight of a term, $tf()$ returns the TF score, $idf()$ is the IDF score, $nf()$ is a normalization factor for the tweet t because the traditional TF-IDF model usually biases to select longer tweets. We used stemming (Porter, 2011) in `NLTK` (Bird et al., 2009) for non-stopwords because stop words contribute insignificantly in the scoring step.

Tweet selection After scoring tweets in each subtopic by using Eq. (2), top *m* ranked tweets having the highest scores were selected as a summary for each cluster.

3. Experimental Setup

Settings In the preprocessing step, URLs were removed to reduce the noise. Standard Cosine similarity (threshold = 0.85) was also used to remove duplicate tweets (those have a very similar content). After being removed, clusters having more than 10 tweets were considered for summarization. The output of summarization methods was fix by $m = 15$ if $n > 15$; otherwise $m = \frac{n}{2}$; where n is the number of tweets in each cluster after preprocessing and removing duplicate ones.

Baselines We compared the hybrid method to basic models, which have been widely used for extractive summarization. **KL (Kullback-Leibler) Divergence** measures the difference of unigram probability distributions learned from seen documents (original documents) and unseen documents (summaries) based on KL-Divergence (Sripada and Jagarlamudi, 2009). **LSA** uses latent semantic analysis with the usage of SVD to rank tweets (Gong and Liu, 2001). **Sumbasic** bases on the impact of frequency on various aspects of summarization (Nenkova and Lucy, 2005). **TextRank** utilizes a graph-based ranking algorithm (Mihalcea and Tarau, 2004) for phrases and sentence extraction. **Retweet** represents the importance of a tweet based on retweet (Busch et al., 2012). **DSDR-linear** bases on data reconstruction with linear combination (He et al., 2012).

Evaluation method The evaluation was conducted on each cluster, by matching extracted tweets with the references. We employed `ROUGE-1.5.5` (Lin and Hovy, 2003) by using `pyrouge`[8] with ROUGE-1, 2, and SU F-score to balance precision and recall.

4. Results and Discussion

4.1. Experimental Results

ROUGE scores with gold-standard references We report the summarization performance of the hybrid model on our dataset with the average of ROUGE scores in 26 days compared to the baselines.

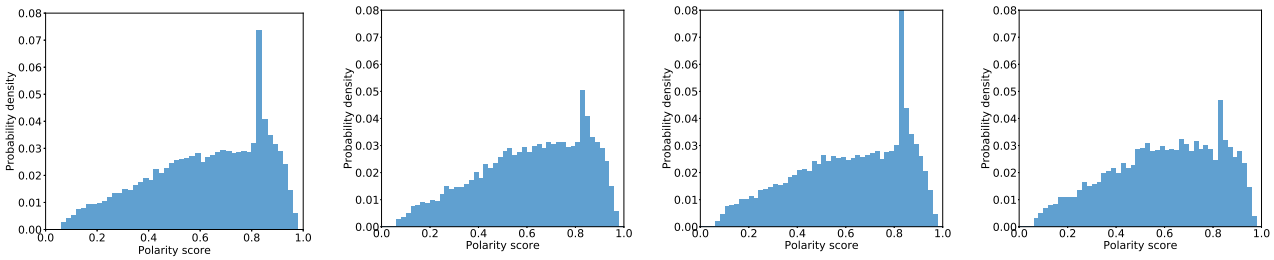
Table 3: The average ROUGE scores over six datasets. Text means the hybrid model significantly outperforms with $p \leq 0.05$. **Bold** is the best, *italic* is second bet.

Method	ROUGE-1	ROUGE-2	ROUGE-SU
KL	0.394	0.263	0.146
LSA	0.462	0.368	0.175
Sumbasic	0.444	0.298	0.174
TextRank	**0.495**	**0.418**	**0.213**
Retweet	0.384	0.264	0.129
DSDR-lin	0.460	0.351	0.183
h-TFIDF	*0.482*	*0.384*	*0.199*

The ROUGE scores indicate that the hybrid model obtains very competitive results, where it significantly outperforms almost methods (using the pair $t-$test[9]), except for TextRank. It confirms the efficiency of the model in summarizing short texts (Inouye and Kalita, 2011). However, a large margin between the ROUGE scores of the hybrid model and the upper bounds (Table 2) suggest that its performance can be improved. TextRank is the best model for all metrics

[8] parameters: -c 95 -2 -1 -U -r 1000 -n 4 -w 1.2 -a -m

[9] https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/scipy.stats.ttest_ind.html



(a) All data (b) H-TFIDF (c) KL (d) DSDR-non

Figure 3: Polarity distribution of original and extracted tweets from three models.

because it is similar to LexRank, which achieves the highest upper bound of ROUGE scores in Table 2. Small margins between the hybrid method and TextRank indicate that we can still increase the performance of the hybrid model. Methods based on the content quality analysis (except for KL and retweet) are competitive, confirming that content quality is a critical factor for tweet selection (Inouye and Kalita, 2011; Duan et al., 2012; Shou et al., 2013; Nguyen et al., 2015).

ROUGE scores with hashtags We also evaluated all the methods by using hashtags. The intuition is that tweets usually include hashtags, which show important information regarding user's interests. To do that, we extracted all hashtags of each cluster to form its artificial references.

Table 4: The average ROUGE scores over six datasets.

Method	ROUGE-1	ROUGE-2	ROUGE-SU
KL	0.122	0.033	0.015
LSA	0.111	*0.034*	*0.017*
Sumbasic	**0.137**	*0.034*	*0.018*
TextRank	0.100	0.030	0.011
Retweet	0.101	0.024	0.007
DSDR-lin	0.117	0.033	0.011
DSDR-non	*0.123*	**0.036**	0.010
Luhn	0.105	0.032	0.011
LexRank	0.118	0.033	0.013
h-TFIDF	0.113	0.031	0.010

ROUGE scores from Table 4 indicate that the hybrid model is still better than some methods, but results are slightly worse than those in Table 3. The margin among these models is small because hashtags are short and single words.

4.2. Polarity Observation

We argue that tweets usually include users' opinions defined as the polarity aspect (Turney, 2002; Pang et al., 2002; Liu, June 2015). Figure 2 also supports our argument. To reveal this aspect, we trained a polarity classifier to predict whether an input tweet contains polarity (sentiment/non-sentiment). We adapted Semeval datasets[10] because of the unfortunate lack of this kind of dataset for our task. From Semeval 2013 to 2016, we obtained 22,591 neutral tweets, 19,903 positive, and 7,840 negative tweets. We randomly selected 15,680 neutral ones as non-sentiment tweets, and 7,840 positive and 7,840 negative tweets to form sentiment tweets. We employed convolutional neural networks (CNN) (Kim, 2014; Kalchbrenner et al., 2014; Zhang and Wallace, 2015) for training as the setting of (Kim, 2014). The number of each region size is 300 and the dimension of a penultimate NN layer (with Dropout rate $p = 0.5$) is 100. We finally applied the trained model to our data. After predicting, a score of each tweet was converted to polarity intensity in [0, 1], where tweets with high scores (close to 1) are non-sentiment whereas those close to 0 are sentiment. Figure 3a shows that many tweets distribute in [0.4, 1]. The number of non-sentiment tweets (in [0.5, 1]) is larger than that of sentiment ones (in [0.5, 1]).

We also observed extracted tweets from three methods: hybrid TF-IDF, KL, and DSDR-non to investigate polarity. The distributions in Figures 3b, 3c and 3d are quite similar to Figure 3a, where tweets selected by the three models mainly range in [0.5, 1]. For example, the density of extracted tweets from DSDR-non, one of the competitive models, mainly distributes in [0.5, 1], showing that salient tweets tend to be non-sentiment. The same patterns appear in the result of hybrid TF-IDF and KL. The distribution in Figure 3 suggests a deeper analysis in combining polarity and content quality in the summarization process.

5. Conclusion

In this paper, we present a new dataset for tweet summarization. It includes six events collected from October 10 to November 9, 2019. The major property of our dataset is human involvement in creating gold-standard references, which provide reliability to evaluate extractive methods. Tweets also are assigned in sub-topics in consecutive days, which facilitate continuous tweet stream summarization. Experimental results conclude that the hybrid TF-IDF model obtains very competitive ROUGE scores.

We encourage to validate other advanced methods on our dataset. The preliminary analysis of polarity reveals the fact that tweets usually include users' opinions. It motivates a possible direction to exploit the polarity of tweets to improve the scoring step.

6. Acknowledgments

This work was supported by JSPS KAKENHI Grant number 15K16048, Japan; and Center for Research and Applications in Science and Technology, Hung Yen University of Technology and Education, under the grant number UTEHY.T026.P1718.04.

[10] http://alt.qcri.org/semeval2016/task4/

7. Bibliographical References

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

Busch, M., Gade, K., Larson, B., Lok, P., Luckenbill, S., and Lin, J. J. (2012). Earlybird: Real-time search at twitter. In *Proceedings of 28th International Conference on Data Engineering, pp. 1360-1369. IEEE*.

Duan, Y., Chen, Z., Wei, F., Zhou, M., and Shum, H.-Y. (2012). Twitter topic summarization by ranking tweets using social influence and content quality. In *Proceedings of the 24th International Conference on Computational Linguistics, pp. 763-780. Association for Computational Linguistics*.

Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research, 22, pp. 457-479*.

Forgy, E. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics, 21: 768?769*.

Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *Science 315(5814), pp. 972-976*.

Gong, Y. and Liu, X. (2001). Generic text summarization using relevant measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 19-25. ACM*.

He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., and He, X. (2012). Document summarization based on data reconstruction. In *AAAI, pp. 620-626*.

Imran, M., Castillo, C., Lucas, J., Meier, P., and Vieweg, S. (2014). Aidr: Artificial intelligence for disaster response. In *WWW (Companion Volume): 159-162*.

Inouye, D. I. and Kalita, J. K. (2011). Comparing twitter summarization algorithms for multiple post summaries. In *Proceedings of Third Inernational Conference on Social Computing and Privacy, Security, Risk and Trust, pp. 298-306. IEEE*.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp. 655-665. Association for Computational Linguistics*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of Conference on Empirical Methods in Natural Language Processing, pp. 1746-1751. Association for Computational Linguistics*.

Lin, C.-Y. and Hovy, E. H. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, pp. 71-78. Association for Computational Linguistics*.

Liu, B. (June 2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research Development, 2(2), pp. 159-165*.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol. 1, no. 14, pp. 281-297*.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 404-411. Association for Computational Linguistics*.

Nenkova, A. and Lucy, V. (2005). The impact of frequency on summarization. Technical report, Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005 101.

Nguyen, M.-T., Kitamoto, A., and Nguyen, T.-T. (2015). Tsum4act: A framework for retrieving and summarizing actionable tweets during a disaster for reaction. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 64-75. Springer International Publishing*.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, pp. 79-86. Association for Computational Linguistics*.

Porter, M. F. (2011). Snowball: A language for stemming algorithms.

Shou, L., Wang, Z., Chen, K., and Chen, G. (2013). Sumblr: continuous summarization of evolving tweet streams. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 533-542. ACM*.

Sripada, S. and Jagarlamudi, J. (2009). Summarization approaches based on document probability distributions. In *Proceedings of The 23rd Pacific Asia Conference on Language, Information and Computation, pp. 521-529*.

Turney, P. D. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 417-424. Association for Computational Linguistics*.

Zhang, Y. and Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

A Workbench for Rapid Generation of Cross-Lingual Summaries

Nisarg Jhaveri, Manish Gupta[†], Vasudeva Varma
International Institute of Information Technology, Gachibowli, Hyderabad, Telangana - 500 032, India.
nisarg.jhaveri@research.iiit.ac.in, {manish.gupta, vv}@iiit.ac.in

Abstract

The need for cross-lingual information access is more than ever with the easy accessibility to the Internet, especially in vastly multilingual societies like India. Cross-lingual summarization can help minimize human effort needed for achieving publishable articles in multiple languages, while making the most important information available in target language in the form of summaries. We describe a flexible, web-based tool for human editing of cross-lingual summaries to rapidly generate publishable summaries in a number of Indian Languages for news articles originally published in English, and simultaneously collect detailed logs about the process, at both article and sentence level. Similar to translation post-editing logs, such logs can be used to evaluate the automated cross-lingual summaries, in terms of effort needed to make them publishable. The generated summaries along with the logs can be used to train and improve the automatic system over time.

Keywords: information access, cross-lingual summarization, machine translation, human evaluation of summaries

1. Introduction

In Indian news media, most of the content gets published in English first and then in regional languages, especially for news categories such as national or international news, technology or lifestyle news. The delay can be just a few hours, days or sometimes the news does not appear in the regional languages at all. On the other hand, some content gets generated and consumed in regional languages alone. With the Internet becoming easily accessible and the rise of digital journalism, it is now crucial to make the large amount of news published in English or other popular languages on the Internet available to the readers of other languages having fewer native publications.

Advances in Machine Translation (MT) and other fields of Computational Linguistics in recent years make it possible to automate cross-lingual news access. However, the current state of Machine Translation is not able to generate publishable articles in most Indian Languages from English. Although, post-editing MT output has been shown to increase translator's productivity over translating from scratch (Aziz et al., 2012), it still requires a significant amount of human effort to produce end-user consumable articles.

Making the highlights or summaries of articles originally published in English accessible to non-English speaking users helps in making a large amount of *critical* information accessible as fast as possible with minimal human effort. We aim to make the process completely or partially automatic so that the gist of the articles can be published in regional languages with minimal delay.

Working in this direction, we've developed a pluggable system, implementing a pipeline for cross-lingual summarization of news articles. While summarization and translation are two major modules in the automatic cross-lingual summarization pipeline, a number of other modules can be included, for example, preprocessing, automatic post-editing, etc. After automatic processing, the articles are sent to humans for post-editing the summary and the automatically translated text to produce publishable cross-lingual summaries.

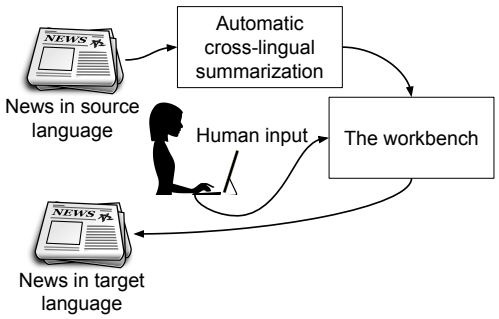


Figure 1: Overall flow of information

In this paper, we present a web-based tool for post-editing cross-lingual summaries, and briefly describe a pluggable pipeline for cross-lingual summarization. The source code of the tool, which we refer to as the workbench, is available on GitHub[1].

The workbench is pluggable by design, which enables it to work with a number of different MT systems, summarization systems and other tools available for different languages.

The workbench also records edit logs and other parameters while editing the automatic summary and translation. These logs can give meaningful insights for the task or can also be used as continuous feedback to the system.

The rest of the paper is organized as follows: Section 2 presents some related works. Section 3 describes the workbench in detail, including its main functionalities, interface and architecture. In Section 4 we show some examples of data collected and result of a pilot study. In Section 5 we discuss other possible use-cases for the workbench and possible future works. At the end, Section 6 contains conclusions.

[†]The author is also a Principal Applied Scientist at Microsoft.

[1]https://github.com/nisargjhaveri/news-access

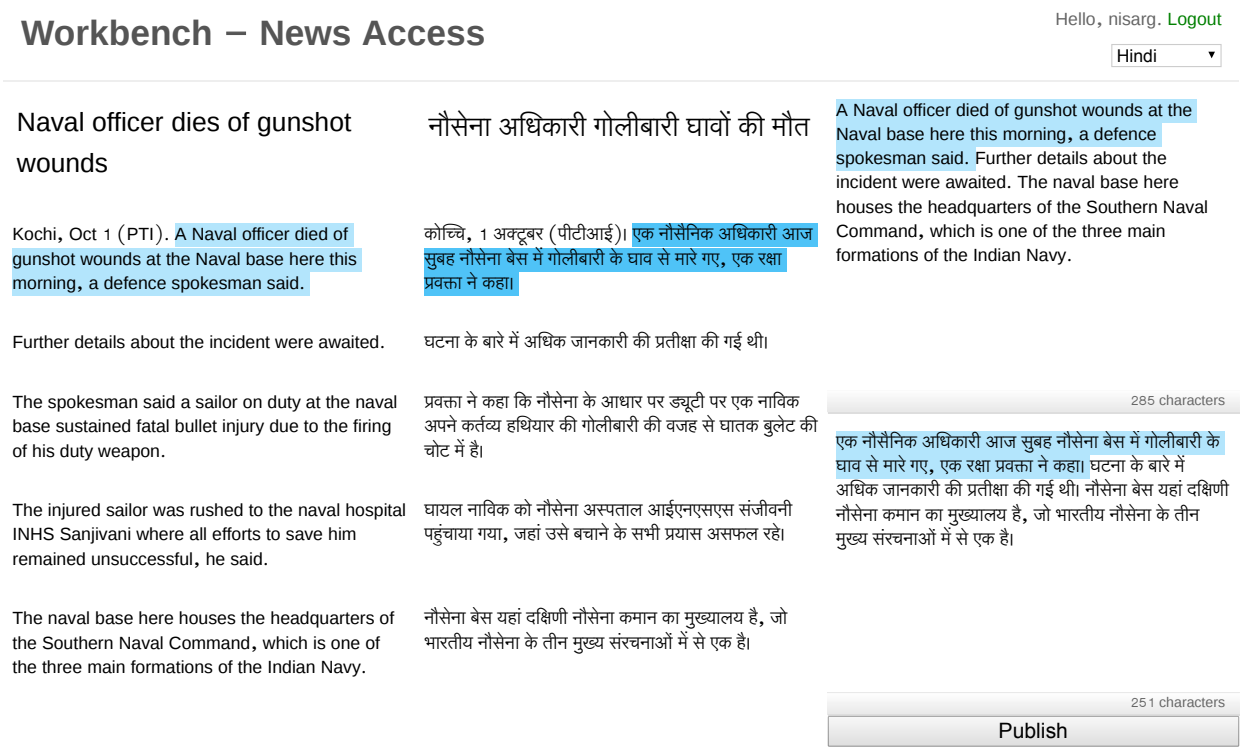


Figure 2: Screenshot of the workbench, with a sentence highlighted.

2. Related Work

To the best of our knowledge, there is no available end-to-end system that allows post-editing of automated cross-lingual summaries to generate publishable summaries and at the same time can collect useful data about the process.

Computer Aided Translation (CAT) tools or translation post-editing tools like *SDL Trados Studio*[2], *MateCat*[3], *OmegaT*[4], *PET* (Aziz et al., 2012) and *CATaLog online* (Pal et al., 2016) are available. They compare with our system in following ways. *a*) While they support translation post-editing, we support editing of cross-lingual summaries as well. *b*) A few of them allow the recording of various kinds of logs about the translation post-editing process, while we allow recording comprehensive logs about the human editing of summary and translations.

Some work exists on cross-lingual summarization. Most recently, Zhang et al. (2016) proposed abstractive cross-lingual summarization. Yao et al. (2015) proposed compressive cross-lingual summarization inspired by phrase-based translation models. Wan (2011) proposed summarization using information from both source and translated article, while Wan et al. (2010) proposed to summarize considering the translation quality prediction.

Most extractive cross-lingual summarization systems have a sequential pipeline architecture. Additionally, most of them output a proposed mono-lingual summary and its translation at the end (Litvak et al., 2010; Orasan and Chiorean, 2008; Pingali et al., 2007; Wan et al., 2010; Wan, 2011; Yao et al., 2015).

This motivates the design of the workbench where the annotator can edit the mono-lingual summaries and its translation easily to get publishable cross-lingual summaries, and which can also collect various logs.

3. The Workbench

The workbench is a flexible, language independent tool for editing automatically generated cross-lingual summaries. The main features of the workbench are:

- The workbench provides a unique user-friendly environment for annotators to edit summaries in the source language, the cross-lingual summaries, and optionally, translation of the original article, in a seamless way.
- The pluggable and generic architecture provides possibility of using the workbench for almost any language pair, and with any set of external tools to plug into the pipeline.
- The workbench collects a wide range of logs from the editing jobs, which can be used as feedback by any module in the pipeline to improve the automatic process over time, and can also provide useful insights for the task in question.

3.1. Interface

Figure 2 shows the default interface of the workbench with source and target languages being English and Hindi respectively.

The default interface for the workbench splits the working area in three vertical columns. The first column contains the source article. The second column contains the translation of the article in the target language. Paragraphs are aligned

[2] http://www.sdltrados.com/
[3] https://www.matecat.com/
[4] http://omegat.org/

in both these columns and the scrolling is synced. The third column contains the mono-lingual extractive summary in source language and the translation of the summary in the target language.

The annotator can edit the sentences in the mono-lingual summary and the translations of the sentences in summary, with the source and automatically translated article as the context. Optionally, the sentences in the translation of the complete article can also be edited.

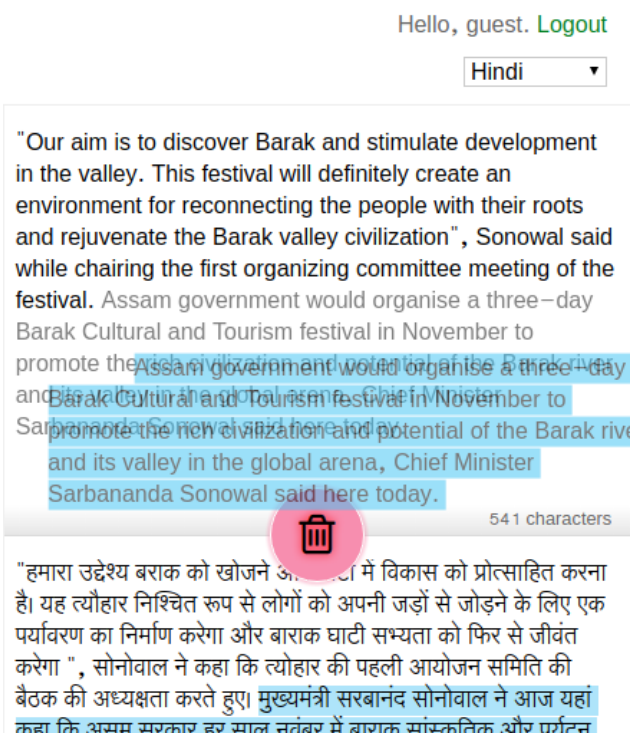


Figure 3: Screenshot of a summary sentence being dragged for deletion.

To edit the mono-lingual summary, a simple drag-drop interface is provided to make the usage intuitive. The annotator can add or remove sentences from the mono-lingual summary with an easy drag-drop interface. When the annotators starts dragging a sentence from the mono-lingual summary, a bin is shown near the end of the summary box, see Figure 3. The annotator can either drop the sentence at some position inside the box to reorder or they can drop it on the bin to remove the sentence from the summary. To add a sentence from original article into the summary, the annotator has to pick a sentence from the source article and drop into the summary box at the desired position. The position of the sentence is previewed live while the sentence is being dragged. Additionally, the content of the sentences in the summary can be changed in-line by clicking on it. Any changes in mono-lingual summary reflects in the cross-lingual summary immediately. The number of characters in the summary is also shown below the summary.

Contrary to the translation post-editing tools, since the structure of the article is important for summarization, we cannot show the original article broken into segments, or even sentences. The view shows paragraphs of the source and target article aligned with synced scrolling for easy navigation. To distinguish between multiple sentences in the paragraph, the sentence which is hovered or is active for editing is highlighted along with all the linked sentences, such as the source sentence, the corresponding sentence in mono-lingual summary if included, and the corresponding sentence in cross-lingual summary.

3.2. Architecture

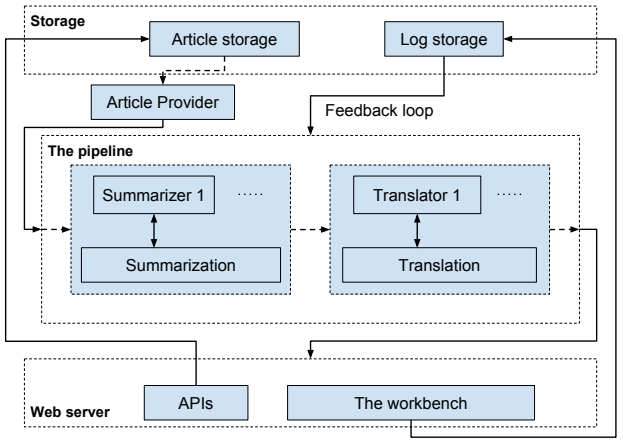


Figure 4: Overall architecture diagram

Figure 4 shows overall architecture of the system. The dotted arrows in the diagram indicate a possible flow, which is quite flexible.

The modular architecture of the system results in high flexibility in terms of use of external resources. The article provider module retrieves news article from either a corpus or other external sources.

The article is then passed through the automated pipeline, which depending on specifics, populates the article object with translation of the article, mono-lingual summary and the cross-lingual summary obtained based on the mono-lingual summary.

Each module in the pipeline can potentially access external resources, make API calls or access previous edit logs in order to generate the output. The modules themselves can be composite of other components.

In our system, summarization and translation are the two major modules along with pre-processing stage. Multiple summarizers and translators are integrated in summarization and translation module respectively. For example, in our case, the translation module may use Google Translate[5] or Microsoft Translator[6] depending on the language pair or a configuration option.

3.3. Log Collection

One of the primary goal of the workbench is to generate a large amount of cross-lingual news summaries in multiple languages, which can be used to build new systems in the area. Along with that, various kinds of user-activity logs are collected by the workbench, which can also be used to evaluate or improve new systems.

[5] https://translate.google.com/
[6] https://www.microsoft.com/translator

Log level	Log type	Short description
Sentence-level	Focus/Blur	When the sentence is activated for editing or de-activated
	Keystrokes	Key presses, along with IME compositions
	Text selection	Text selection by any means, e.g. with mouse or shift-arrow keys
	Text input	When the text is changed, including copy/cut/paste events
Summary-level	Add/Remove sentence	When a sentence is added or removed from the summary
	Sentence reordering	When the order of the sentences in the summary is changed
Article-level	Page events	Events about page load, article load, and completion of editing

Table 1: Summary of various kinds of logs collected by the workbench

Table 1 summarises the different kinds of logs collected by the workbench. All events are logged with precise time in milliseconds.
The following sentence-level editing events are logged for all editable sentences in source language or target language.

- Focus/Blur: Focus and blur events are logged for each editable item when the item is activated for editing and when leaving the focus from the item respectively. A single item can be focused multiple times. The time spent on the item can be estimated by adding difference between all focus-blur pairs for the item.
- Keystrokes: All keystroke information available are logged, including IME (Input Method Editor) composition information.
- Text selection: Text selection by any means (mouse, shift-arrow) is logged.
- Text input: For all items, all the changes in text are logged, along with all copy/cut/paste events. This is specifically important as keystroke logging doesn't provide accurate and wholesome information in case of complex scripts and use of IMEs.

For translation post-editing, Human-targeted Translation Edit Rate (HTER) (Snover et al., 2006), along with information about insertion, deletion, and substitution of single words as well as shifts of word sequences can be calculated and stored when the translation is finalized.
For editing of mono-lingual summary, along with the sentence level editing logs, following summary-level events are also logged by the workbench.

- Add/Remove sentence: An event is logged when a sentence is added to the summary or removed from the summary, along with its position in the summary before adding or after removing.
- Sentence reordering: When the ordering of the sentences in the summary is changed, an event is logged with the information about previous and new ordering of sentences in the summary.

Additionally, some article-level events described below are also logged.

- Page events: The events about page load, article load, and completion of editing of an article are logged.

A Naval officer died of gunshot wounds at the Naval base here this morning, a defence spokesman said. ~~Further details about the incident were awaited.~~ ~~The naval base here houses the headquarters of the Southern Naval Command, which is one of the three main formations of the Indian Navy.~~ *The spokesman said a sailor on duty at the naval base sustained fatal bullet injury due to the firing of his duty weapon.*

Figure 5: Edits made to the mono-lingual summary. ~~Removed sentences~~, *Added sentences*

ek nausainika adhikArI Aja subaha nausenA ~~besa meM~~ *aDDe para* gollbArI ke ghAva se ~~mAre gae~~ *mArA gayA*, eka rakSA pravaktA ne kahA. pravaktA ne kahA ki nausenA ~~ke AdhAra~~ *aDDe* para DyUTI ~~para~~ *ke daurAn* ek nAvika *ko* apane kartavya hathiyAra ~~kI gollbarI~~ *se goli calane* kI vajaha se *goli kI* ghAtaka ~~buleTa kI coTa meM hai~~ *coTa lagI thI*.

Figure 6: Edits made to the Hindi translation of the summary. ~~Removed~~, *Added*

The total time taken for editing can be calculated as the time difference between completion of editing and article load time, and reducing the difference by the amount of time the annotator was marked away.
In addition to these logs, annotator's browser and platform information is also collected. This information is important to give us a better idea of client's environment, and helps interpreting the logs with a better context.
With all these logs, we can virtually replay the complete editing process for any article.

4. Examples

Figure 2 shows an example article without editing. The title of the article is wrongly translated to "nausenA adhikArI golIbArI ghAvoM kI mauta", which is neither syntactically nor semantically correct. Using the workbench, an annotator can fix the translation to "nausenA adhikArI kI golIbArI ghAvoM se mauta".
Additionally, we can see that the summary shown in Figure 2 is not very informative, as it is not completely coherent with the title. Figure 5 shows the edits made to the mono-lingual summary using the workbench, which is also reflected in cross-lingual summary. Once the sentences in mono-lingual summary are fixed, with a few corrections in

Article	Total Time	Summary Editing			Final Summary	
		Sentences Added	Sentences Removed	Sentences Reordered	Total Sentences	Total Characters
1	320s	0	0	0	3	371
2	251s	1	1	0	2	364
3	389s	1	1	0	2	310

Table 2: Statistics for summary level editing of example articles

Article	Source Article		Summary					Title			
				Source		Target		Source		Target	
	Sent.	Words	Sent.	Words	Chars	Words	Chars	Words	Chars	Words	Chars
1	8	139	3	72	430	77	371	6	30	9	36
2	6	163	2	38	276	46	364	5	40	9	49
3	5	139	2	53	337	60	310	8	53	10	56

Table 3: Statistics of example articles edited

Article	Total Time	Estimated Translation Editing Time	Total Items	Number of Items Corrected	HTER (Summary and Title)				
					Ins	Del	Sub	Shft	Total
1	320s	223s	4	3	6	2	14	0	25.58 (22/86)
2	251s	49s	3	1	3	0	1	0	7.27 (4/55)
3	389s	277s	3	3	4	0	9	2	22.06 (15/68)

Table 4: Translation statistics for example articles

the translation of the summary, similar to what we have shown for the title, we can generate a publishable cross-lingual summary. Figure 6 shows an example of edits made to the translation of the summary in Hindi.

For demonstration, we used the workbench to generate cross-lingual summaries of three randomly selected articles of similar sizes. The articles were originally in English and we set the target language to Hindi. Table 2 shows the total time taken to generate human edited cross-lingual summaries and the summary-level edits made the the articles. Table 3 shows the number of sentences, words and characters in source articles, as well as mono-lingual and edited cross-lingual summaries. Table 4 shows the estimated time taken for editing erroneous translations and HTER along with number of insertions, deletions, substitutions and shifts performed.

4.1. Pilot Study

We conducted a pilot study to understand the usability and effectiveness of the workbench. One language expert was hired to generate translations of the articles and cross-lingual summaries for English-Gujarati language pair using the workbench.

For the pilot study, the workbench was configured to allow editing of the translation of the entire article along with the mono-lingual and the cross-lingual summary. The mono-lingual extractive summaries were provided by Veooz[7] and Google Translate was used for automatic translations.

The language expert was told to follow the following sequence to correct the article.

- First, correct the translation of all the sentences in the source article.
- Once the translation is corrected, fix the mono-lingual summary.
- The cross-lingual summary automatically picks up the sentences and their translation included in the mono-lingual summary from the article. Fix the translation errors in cross-lingual summary if required.

In this setting, we observed that the mono-lingual summary was never getting changed. On investigating, the feedback from the language expert was that the summaries are "good enough". Although, the summaries were not always good, we observed that in this setting, due to multiple tasks (translation of the article, mono-lingual and cross-lingual summary), the "good enough" summaries were not getting enough attention.

Another feedback from the language expert was that "it would be easier and faster to translate by hand instead of post-editing the machine translation". To verify this claim, we did a small experiment to compare time and effort taken in both the cases. The results and statistics of the experiment are shown in Table 5.

We can clearly see that the post-editing approach is faster. We see that post-editing machine translations takes about 33% lesser time compared to translating by hand. Though, post-editing machine translation takes lesser time, we notice that it might be more difficult to correct an erroneous translation compared to translation by hand and the difference noticed in time taken might be simply due to the fact that all sentences do not need to be corrected.

[7] https://www.veooz.com/

	Translation by hand	Post-editing machine translation
Avg. time per article	14.9 min	9.85 min
Avg. number of sentences per article	13.70	13.17
Avg. number of sentences edited per article	100%	57.8%
Total number of articles included in study	112	395

Table 5: Comparison between human translation and Post-Editing machine translation

5. Discussion and Future Work

The workbench can be used to collect human edited cross-lingual summaries along with mono-lingual summaries and translations of the original article for English to a number of Indian Languages. Such a dataset can be used for training statistical mono-lingual or cross-lingual summarizers as well as can help research efforts on machine translation. We also aim to collect comprehensive logs and use that as continuous feedback to some of the modules in our pipeline.

As the workbench and the architecture is not limited to a specific set of languages, the same can be used with a number of other language pairs too.

The flexibility of the workbench and the pipeline makes it possible to use the system for a number of other related tasks. The workbench can be used for extractive or abstractive mono-lingual summary generation or post-editing. It can also be used just as another translation post-editing tool, or can be used to prepare paraphrasing datasets.

Apart from the human edited data collected by the workbench, the logs collected about the process can also be important. Keystroke logs, along with translation time and HTER is a common measure of Translation Quality Estimation (Aziz et al., 2012). Automated Post-Editing (APE) systems also use similar information to automatically post-edit and try and remove systematic errors made by a particular MT system. The workbench can also be used to compare different settings of the pipeline such as different MT systems or different approaches to summarization, etc., by comparing time taken to edit or other relevant measures.

In future, following possible modules could be integrated with the workbench to improve the usability and the effectiveness of the workbench.

- Cross-lingual dictionaries to refer words and get possible translations. The dictionary can be triggered by double-clicking or selecting a word or phrase and can be shown as a pop-up to ease the work-flow.
- A Translation Quality Estimation module, that adapts and learn from previous edits, and can highlight sentences or part of sentences that need attention.
- An Automatic Post-Editing module, to automatically remove common errors made by the MT system, based on previous usage of the workbench.

6. Conclusion

We presented the workbench, a flexible, language independent, web-based tool for human editing of cross-lingual summaries along with a pluggable pipeline for cross-lingual summarization. We also explained some of the core features of the tool and possible usage scenarios for the tool, and discussed briefly about the possible uses of the data collected by the tool.

7. Acknowledgements

This work is done in collaboration and with partial support from Veooz Labs (now NewsPlus)[8].

8. Bibliographical References

Aziz, W., Castilho, S., and Specia, L. (2012). Pet: a tool for post-editing and assessing machine translation. In *LREC*, pages 3982–3987.

Litvak, M., Last, M., and Friedman, M. (2010). A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 927–936. Association for Computational Linguistics.

Orasan, C. and Chiorean, O. A. (2008). Evaluation of a cross-lingual romanian-english multi-document summariser. In *LREC*.

Pal, S., Zampieri, M., Naskar, S. K., Nayak, T., Vela, M., and van Genabith, J. (2016). Catalog online: Porting a post-editing tool to the web. In *LREC*.

Pingali, P., Jagarlamudi, J., and Varma, V. (2007). Experiments in cross language query focused multi-document summarization. In *Workshop on Cross Lingual Information Access Addressing the Information Need of Multilingual Societies in IJCAI2007*.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200.

Wan, X., Li, H., and Xiao, J. (2010). Cross-language document summarization based on machine translation quality prediction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 917–926. Association for Computational Linguistics.

Wan, X. (2011). Using bilingual information for cross-language document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1546–1555. Association for Computational Linguistics.

[8] https://www.veooz.com/

Yao, J.-g., Wan, X., and Xiao, J. (2015). Phrase-based compressive cross-language summarization. In *EMNLP*, pages 118–127.

Zhang, J., Zhou, Y., and Zong, C. (2016). Abstractive cross-language summarization via translation model enhanced predicate argument structure fusing. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(10):1842–1853.

Annotation and Analysis of Extractive Summaries for the Kyutech Corpus

Takashi Yamamura and Kazutaka Shimada
Department of Artificial Inteligence, Kyushu Institute of Technology
680-4 Kawazu Iizuka Fukuoka Japan
{t_yamamura, shimada}@pluto.ai.kyutech.ac.jp

Abstract

Summarization of multi-party conversation is one of the important tasks in natural language processing. For conversation summarization tasks, corpora have an important role to analyze characteristics of conversations and to construct a method for summary generation. We are developing a freely available Japanese conversation corpus for a decision-making task. We call it the Kyutech corpus. The current version of the Kyutech corpus contains topic tags of each utterance and reference summaries of each conversation. In this paper, we explain an annotation task of extractive summaries. In the annotation task, we annotate an importance tag for each utterance and link utterances with sentences in reference summaries that already exist in the Kyutech corpus. By using the annotated extractive summaries, we can evaluate extractive summarization methods on the Kyutech corpus. In the experiment, we compare some methods based on machine learning techniques with some features.

Keywords: multi-party conversation, the Kyutech corpus, summarization, annotation

1. Introduction

Conversation summarization is useful to understand the content of conversations for both participants and non-participants. Many researchers have studied meeting and conversation summarization (Banerjee et al., 2015, Mehdad et al., 2014, Oya et al., 2014).

For the summarization tasks, summarization systems need to recognize significant content from each utterance to cover all important information in the conversation. Therefore, corpora are very important to analyze characteristics of conversations and to construct a method for summary generation. There are some corpora in English, such as the AMI corpus (Carletta, 2007) and the ICSI corpus (Janin et al., 2003). These meeting corpora contain meeting record data with many annotations, such as dialogue acts and summaries and researchers have effectively used such annotations in the summarization task.

We have constructed the Kyutech corpus (Yamamura et al., 2016); it is a Japanese conversation corpus about a decision-making task with four participants. To the best of our knowledge, the Kyutech corpus is the first Japanese corpus annotated for summarization tasks and freely available to anyone[1]. The current Kyutech corpus consists of nine conversations[2] with four scenarios of which discussion settings differ from each other; topic tags of each utterance, and reference summaries of each conversation. On the other hand, the AMI corpus contains numerous annotations, such as extractive summaries and dialogue acts.

In this paper, we focus on an annotation task of extractive summaries. The purpose of extractive summarization is to select important utterances automatically. We annotate an importance tag to each utterance by linking sentences in reference summaries. The annotated utterances are extractive summaries for each conversation. By using the extractive summaries, we can apply extractive summarization methods that have already been proposed by several researchers to the Kyutech corpus because there has been a lot of works on extractive techniques in conversation summarization (Murray et al., 2005, Hirohata et al., 2006). Therefore, construction of extractive summaries is effective and leads to a deeper analysis of the Kyutech corpus.

The final goal of our study is to generate an abstractive summary from a multi-party conversation. This annotation is also useful to apply abstractive summarization methods to the Kyutech corpus because there have been some studies on abstractive models by using extractive techniques (Mehdad et al., 2013).

The contributions of this paper are as follows:

- We extend the Kyutech corpus by annotating extractive summaries.
- As a case study, we examine extractive summarization methods using supervised approaches.

2. Related Work

Extractive summarization has been studied in various domains, such as news articles (Nallapati et al., 2016) and meeting records (Tixier et al., 2017). In multi-party conversation, extractive summarization is a difficult challenging task because the meeting transcripts are composed informal and disfluency utterances with overlapping speakers (McKeown et al., 2005). Therefore, extractive approaches for conversation summarization often differ from techniques of other domains, such as document summarization. Feature-based approaches are commonly used for meeting summarization. Xie et al. (2008) have evaluated the effectiveness of different types of features, such as lexical, structural, discourse and topic features.

In this paper, we focus on a summarization task on a Japanese conversation corpus. The Corpus of Spontaneous Japanese (CSJ) (National Institute for Japanese Language and Linguistics, 2006) is a famous speech corpus with audio data, and most of the speech materials are spontaneous monologues. Hirohata et al. (2006) have been proposed sentence extractive speech summarization on the corpus.

[1] http://www.pluto.ai.kyutech.ac.jp/~shimada/resources.html

[2] The discussion time is 20 minutes per each conversation.

While the CSJ is a spontaneous monologues corpus, the Kyutech corpus is a multi-party conversation corpus for a decision-making task. Therefore, the technique of Hirohata et al. (2006) to summarize a monologue is not necessarily suitable for the Kyutech corpus.

In a Japanese multi-party conversation, Tokunaga and Shimada (2015) introduced machine learning approaches with verbal and nonverbal features at the sentence extraction step. In our past research, we reported that additional features about time information were relatively variable (Yamamura et al., 2015). In this paper, we compare three machine learning approaches with these features; Support Vector Machines (SVMs) (Vapnik, 1999), Conditional Random Fields (CRFs) (Lafferty et al., 2001), and Random Forests (Breiman, 2001).

3. The Kyutech corpus

In this section, we explain the current Kyutech corpus. It contains topic tags for each utterance and three reference summaries for each conversation (Yamamura et al., 2016).

The Kyutech corpus contains multi-party conversations with four participants randomly selected from sixteen male students and four female students. The participants pretended managers of a virtual shopping mall in a virtual city and then determined a new restaurant, as an alternative to a closed restaurant, from three candidates. Before the discussion, the participants read a 10-pages document including information about the three candidates, the closed restaurant and the existing restaurants in the mall, the city information, statistical information about the shopping mall, and so on. They read the document for 10 minutes, then discussed the candidates for 20 minutes and finally determined one restaurant as a new restaurant opening. The current Kyutech corpus consists of nine conversations based on four scenarios of which task settings differ from each other.

The transcription rules were based on the construction manual of the Corpus of Spontaneous Japanese (CSJ) by (National Institute for Japanese Language and Linguistics, 2006). All utterances in the corpus were separated by 0.2-second interval by the guideline and annotated some tags such as filler, question, and so on. Each utterance was not always sentence-level because it depended on the 0.2-second interval rule. Therefore, other tags were appended to the end of each utterance for sentence-level identification. The corpus consists of 4,509 utterances in nine conversations, with a total of 2,810 sentences.

The Kyutech corpus contains the annotations for conversation summarization. In general, topic segmentation (Galley et al., 2003) has an important role as the first step in the meeting summarization (Banerjee et al., 2015, Oya et al., 2014). By dividing the utterances into topic units by topic segmentation, it is possible to take into account of topics of the discussion in conversation summarization. Each utterance has topic tags[3] representing its topics to analyze topic sequences. Table 1 shows the tag names and the descriptions.

[3] At least one tag is annotated to one utterance, and up to two additional tags are also allowed.

Topic	Description
CandX	Topic about the candidate 1
CandY	Topic about the candidate 1
CandZ	Topic about the candidate 3
Cands	Topic about the candidates
Closed	Topic about the closed restaurant
Exist1	Topic about the existing restaurant 1
Exist2	Topic about the existing restaurant 2
Exist3	Topic about the existing restaurant 3
Exist4	Topic about the existing restaurant 4
Exist5	Topic about the existing restaurant 5
Exist6	Topic about the existing restaurant 6
Exists	Topic about the existing restaurants
ClEx	Topic about the existing restaurants and the closed restaurant
Mall	Topic about the shopping mall
OtherMall	Topic about other shopping malls
Location	Topic about the positional relation among restaurants
Area	Topic about areas and cities
People	Topic about the target customers
Price	Topic about the price
Menu	Topic about the menu
Atomos	Topic about the atmosphere
Time	Topic about the business hours
Seat	Topic about the number of seats
Sell	Topic about the sales
Access	Topic about the access to the shopping mall
Meeting	Topic about the proceedings and final decision
Chat	Chats that not related to the task
Vague	Others and unknown

Table 1: Topic tags in the Kyutech corpus.

At the summary generation steps, we complied with the guideline of abstractive hand summaries of the AMI corpus[4]. Based on the manual, two abstractive summaries for one conversation were generated, and the size of each summary was 250 characters to 500 characters[5]. When each annotator generates a summary, they receive the following message for the summary generation: "Write a summary that is understandable for somebody who was not present at the meeting. We also generated third reference summaries from the two summaries of annotators as the consensus summary.

4. Extractive summarization

In this section, we explain an extractive summarization task in the Kyutech corpus. First, we detect important utterances in each conversation. Here the important utterances denote utterances that relate to sentences in reference summaries. Next, we explain some extractive summarization models based on machine learning techniques.

4.1. Annotation

The current Kyutech corpus contains the three reference summaries of each conversation. We created extractive

[4] http://groups.inf.ed.ac.uk/ami/corpus/guidelines.shtml

[5] The number of words was approximately 150 content words on average. The number of unique words was 80 words on average.

(a) At the meeting, participants discussed which of candidate restaurants is suitable as an alternative to the closed restaurant *Japanese WAYA*.
(b) They held a meeting to select a new restaurant from *Ethnic Food Okinawa*, *Taiwan Noodle*, and *BonoPasta* as an alternative to the closed restaurant *Yakiniku Mr.K* that was unprofitable.
(c) They discussed a new restaurant from *The Ramen Kaibutsu*, *The Tsukemen Fujin,* and *BonoPasta* to replace the closed restaurant *Japanese WAYA*.
(d) They discussed the selection of a new restaurant which is most suitable in *Ethnic food Okinawa, Taiwan Noodles,* and *BonoPasta* to make up for the closed restaurant *Yakiniku Mr.K*.
(e) After that, they discussed the merits and demerits of the remaining candidates, *The Tsukemen Fujin* and *BonoPasta.*

Figure 1: An example of reference summary sentences not related to any utterance. Italic fonts denote the restaurant names in the Kyutech corpus.

Dialog ID	Utterances	Links	Ratio (%)
0313_C1	759	240	31.6
0320_C1	505	124	24.6
0326_C1	502	76	15.1
0326_C2	566	160	28.3
0327_C2	284	52	18.3
0323_C3	324	102	31.5
0327_C3	445	118	26.5
0320_C4	637	69	10.8
0326_C4	487	98	20.1

Table 2: The number of utterances and links with reference summary of each conversation in the Kyutech corpus

summaries using the consensus summaries for each conversation. First, the annotator compared all utterances with each sentence in the consensus summary for each conversation. Next, if an utterance implies the content of a sentence in a reference summary, the annotator links it with the sentence in the reference. More precisely, we annotate the sentence number in the reference summary to the utterance. In other words, we link important utterances with sentences in reference summaries that already exist in the Kyutech corpus. We regard them as extractive summaries for each conversation.

Table 2 shows the annotation results. "Utterances" in the table denotes the total number of utterances of the conversation and "Links" denotes the total number of utterances linked with reference summaries as extractive summaries of the conversation. "Ratio" in the table denotes the percentage of "Links" to "Utterances". The ratio on all conversations was 23 percent on average.

In this annotation, there were 19 percent of the sentences not related to any utterances although most sentences in reference summaries were linked with several utterances. Figure 1 shows examples of these non-linked sentences. The sentences (a) to (d) were the lead sentences of the reference summaries and described the task of the discussion, e.g., theme and setting. It is hard to link the description and utterances in the conversation. Moreover, as shown in the sentence (e), it is also difficult to uniquely determine an utterance related to the reference summary because there are many utterances related to this description in the conversation. The annotation for these sentences is one important future work.

4.2. Extractive summarization method

One of the main purposes of this study is to summarize a multi-party conversation. In this section, we explain some extractive summarization methods.

Some abstractive conversation summarization approaches utilize extractive summarization techniques in the process (Banerjee et al., 2015). In other words, extractive summarization has an important role in summary generation. Therefore, we evaluate the extractive summarization task as the first step for our abstractive summarization.

We have studied extractive summarization for another conversation corpus (Tokunaga and Shimada, 2015, Yamamura et al., 2015). In this paper, we introduce our previous approaches to the Kyutech corpus.

We examine extractive summarization by using three machine learning techniques; SVMs, CRFs, and Random Forests. As features of these machine learning techniques, we use 20 types of verbal and nonverbal information that utilized in (Tokunaga and Shimada, 2015, Yamamura et al., 2015). The following are simple lists of the featurse:

Criteria	Random Forests	SVMs	CRFs
Precision	0.422	0.301	0.411
Recall	0.311	0.308	0.294
F-measure	0.346	0.294	0.326

Table 3: Macro-averaged precision, recall, and F-measure scores.

Features in an utterance

Speaker information (Speaker ID), Utterance position in a conversation, Topic tags in the Kyutech corpus, The number of morphemes in an utterance, Length of an utterance, Presence of declinable words and Presence of interrogatives.

Features between utterances

Difference of lengths of current and previous utterances , Word frequency, Presence of same words, Presence of consecutive utterances of one person.

Nonverbal features

Utterance speed, Utterance timing, Overlap of speech.

5. Experiment

The Kyutech corpus contains nine conversations. We evaluated our methods on the Kyutech corpus with nine-fold cross validation for nine conversations. In other words, we evaluated one test conversation with the model that was generated from the other conversations and repeated this process for all conversations. We computed the precision, recall rates, and F-measure for each conversation and took an average of the overall scores (macro-averaging).

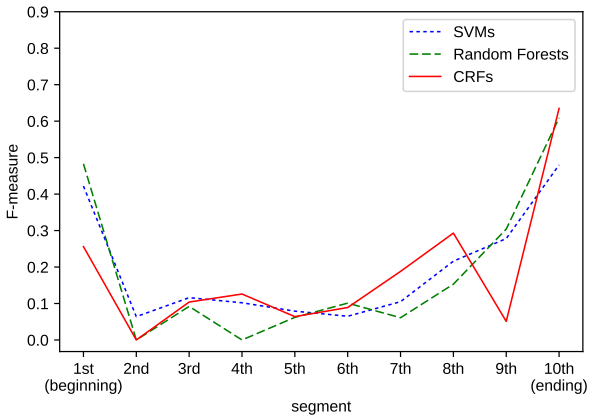


Figure 2: Macro-averaged F-measure scores' comparison for each segment of the meetings. The x-axis, shown as segment, denotes the number of each divided utterances group. The 1st segment means the beginning part of a conversation and the 10th segment means the ending part of a conversation.

Table 3 shows the experimental results. The F-measures of each method were not high on the whole. We also evaluated our methods with some combination patterns of the features to confirm the effectiveness of each feature. As the result of the analysis, the utterance position feature, namely utterance numbers normalized by the total number of utterances, was most effective in our features. Moreover, there was almost no difference between a method using only the utterance position feature and others. In other words, our models heavily depended on the utterance position feature. In order to understand why the feature is effective, we analyzed utterances extracted by our models. The result showed that the models with the utterance position feature mainly extracted utterances in the beginning and the ending parts of conversations as important utterances.

As you can see from Table 3, the performance was not enough. We analyzed the results more deeply. The current models tended to extract important utterances from the beginning and the ending parts in conversations. Therefore, we compared the F-measure scores in each segment in conversations. We divided each conversation into ten segments equally. Figure 2 shows the F-measure scores of each segment for each model. All models obtained high scores in the last segment of the conversation, as compared with the other segments. The last segment of each conversation tends to contain important utterances because it usually contains the final decision of the conversation. Therefore, our models could extract the most important points, namely the decision, in each conversation although the overall scores shown in Table 3 were not enough. The F-measure scores of the 1st segment were relatively higher than the other segments. The beginning part of the conversation usually contains a trigger of the discussion. Therefore, the higher accuracy in the segment shows potential effectiveness of the models.

6. Discussion

In this section, we discuss the following points: the experiment, another extractive summaries, and some summarization tasks.

As mentioned in Section 4.1., there were 19 percent of the reference summary sentences not related to any utterances. To generate these summary sentences, we need to capture not only the contents of the conversation but also discussion settings. Another problem in the experiment was that the size of the Kyutech corpus was not always sufficient for the statistical methods. In general, machine learning methods need a large dataset to generate a strong classifier. Therefore, scaling up the Kyutech corpus is the most important future work. Future work should also focus on applying unsupervised techniques, such as submodularity framework (Tixier et al., 2017) and relation extraction (Wang and Cardie, 2012).

In this work, we designed an extractive summary annotation task for linking utterances in a conversation with sentences in a reference summary. However, there are cases when someone often wants to know the course of discussion in a meeting. For this demand, our extractive summaries are not suitable. However, we have already studied the annotation process of extractive summaries that keep the meaning and context of the original conversation in our past research (Tokunaga and Shimada, 2015). It is also important to apply this process to the Kyutech corpus.

Although we handled a generic summarization task, some researchers have studied other aspects of summarization: focused meeting summarization that creates abstract summaries of specific aspects of meeting such as decisions, actions, and problems (Murray et al., 2010, Wang and Cardie, 2013) and query-based summarization that generates abstract summaries based on users phrasal queries (Mehdad et al., 2014). Introducing these summarization systems is also the future work.

7. Conclusion

In this paper, we explained the annotation task for extractive summarization on the Kyutech corpus. We annotated importance utterances that already exist in the Kyutech corpus, on the basis of reference summaries. As a result, we linked 23 percent of utterances in conversations with sentences in reference summaries as extractive summaries.

We examined some extractive summarization methods. The utterance position feature was most effective in our designed features. However, the accuracy was not enough. Improving the accuracy is one of the most important future work. In addition, scaling up the Kyutech corpus is important future work. In this work, we used features about surface information of utterances and topic tags. On the other hand, there are other important features in conversations, such as dialogue acts (Bunt et al., 2012). We are currently developing the Kyutech corpus with dialogue acts (Hino et al., 2016). We will apply the dialogue acts to extractive summarization models in future work. In addition, we have a plan to open the annotated tags; namely, extractive summaries and dialogue acts, shortly.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 17H01840.

References

Banerjee, S., Mitra, P., and Sugiyama, K. (2015). Generating abstractive summaries from meeting transcripts. In *Proceedings of ACM Symposium on Document Engineering (DocEng '15)*, pages 51–60.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Bunt, H., Alexanderssona, J., Choe, J.-W., Fang, A. C., Hosida, K., Petukhova, V., Porescu-Belis, A., and Trauim, D. (2012). Iso 24617-2 : A semantically-based standard for dialogue annotation.

Carletta, J. (2007). Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

Galley, M., McKeown, K., Fosler-Lussier, E., and Jing, H. (2003). Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL 2003)*, pages 562–569.

Hino, M., Yamamura, T., and Shimada, K. (2016). Design and analysis of dialogue acts for the kyutech corpus. In *IEICE Technical Report, Natural Language Understanding and Models of Communication (In Japanese)*, volume 116, pages 1–6.

Hirohata, M., Shinnaka, Y., Iwano, K., and Furui, S. (2006). Sentence-extractive automatic speech summarization and evaluation techniques. In *Speech Communication*, volume 46, pages 1151–1161.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., and Wooters, C. (2003). The ICSI meeting corpus. In *Proceedings of the IEEE ICASSP*, pages 364–367.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pages 282–289.

McKeown, K., Hirschberg, J., Galley, M., and Maskey, S. (2005). From text to speech summarization. In *Proceedings of the 30th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 997–1000.

Mehdad, Y., Carenini, G., Tompa, F. W., and Ng, R. T. (2013). Abstractive meeting summarization with entailment and fusion. In *Proceedings of the 14th European Natural Language Generation (ENLG - SIGGEN 2013)*, pages 136–146.

Mehdad, Y., Carenini, G., and Ng, R. T. (2014). Abstractive summarization of spoken and written conversations based on phrasal queries. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1220–1230.

Murray, G., Renals, S., and Carletta, J. (2005). Extractive summarization of meeting recordings. In *Proceedings of the 9th European Conference on Speech Communication and Technology*, pages 593–596.

Murray, G., Carenini, G., and Ng, R. T. (2010). Generating and validating abstracts of meeting conversations: a user study. In *Proceedings of INLG 2010*, pages 105–113.

Nallapati, R., Zhou, B., Santos, C., Gulcehre, C., and Xiang, B. (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Neural Language Learning (CoNLL)*, pages 280–290.

Oya, T., Mehdad, Y., Carenini, G., and Ng, R. (2014). A template-based abstractive meeting summarization: Leveraging summary and source text relationships. In *Proceedings of INLG 2014*, pages 45–53.

National Institute for Japanese Language and Linguistics. (2006). Construction manual of corpus of spontaneous japanese (csj) (in japanese).

Tixier, A., Meladianos, P., and Vazirgiannis, M. (2017). Combining graph degeneracy and submodularity for unsupervised extractive summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 48–58.

Tokunaga, Y. and Shimada, K. (2015). Extractive summarization based on a combined method using several features for multi-party conversation. In *ACIS International Journal of Computer and Information Science*, volume 16, pages 12–21.

Vapnik, V. N. (1999). *Statistical Learning Theory*. Wiley.

Wang, L. and Cardie, C. (2012). Focused meeting summarization via unsupervised relation extraction. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 304–313.

Wang, L. and Cardie, C. (2013). Domain independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1395–1405.

Xie, S., Liu, Y., and Lin, H. (2008). Evaluating the effectiveness of features and sampling in extractive meeting summarization. In *Spoken Language Technology Workshop*, pages 157–160.

Yamamura, T., Tokunaga, Y., and Shimada, K. (2015). Multi-party conversation summarization using time information and text segmentation. In *IEICE Technical Report, Natural Language Understanding and Models of Communication (In Japanese)*, volume 115, pages 43–48.

Yamamura, T., Shimada, K., and Kawahara, S. (2016). The Kyutech corpus and the topic segmentation using a combined method. In *Proceedings of the 12th Workshop on Asian Language Resources*, pages 95–104.

A Repository of Corpora for Summarization

Franck Dernoncourt[1], **Mohammad Ghassemi**[2], **Walter Chang**[1]
[1]Adobe Research, [2]MIT
franck.dernoncourt@adobe.com, ghassemi@mit.edu, wachang@adobe.com

Abstract

Summarization corpora are numerous but fragmented, making it challenging for researchers to efficiently pinpoint corpora most suited to a given summarization task. In this paper, we introduce a repository containing corpora available to train and evaluate automatic summarization systems. We also present an overview of the main corpora with respect to the different summarization tasks, and identify various corpus parameters that researchers may want to consider when choosing a corpus. Lastly, as the recent successes of artificial neural networks for summarization have renewed the interest in creating large-scale corpora for summarization, we survey which corpora are used in neural network research studies. We come to the conclusion that more large-scale corpora for summarization are needed. Furthermore, each corpus is organized differently, which makes it time-consuming for researchers to experiment a new summarization algorithm on many corpora, and as a result studies typically use one or very few corpora. Agreeing on a data standard for summarization corpora would be beneficial to the field.

Keywords: abstractive summarization, extractive summarization, artificial neural networks, corpora

1. Introduction

Automatic summarization has been studied for over half a century (Luhn, 1958). Over the decades, many summarization tasks, systems, metrics, and corpora have been created. Summarization approaches may be categorized in abstractive, extractive, and compressive approaches. When the output of a summarization system is a newly generated text, distinct from the original document, it is referred to as *abstractive*. Systems that compose summaries by combining and restructuring various segments of the original text, are referred to as *extractive*, or *sentential extractive* if sentences from the original text are selected to form the summary. Lastly, systems that compose summaries by pruning tokens from the original text are referred to as *compressive*.[1]

Summarization systems may span single, or multiple-documents, and can produce outputs of varying lengths and structures. When the original text spans over multiple documents, the task is called *multi-document* summarization. The length of the summaries differs across corpora: for example, *sentence-level* summarization aims at summarizing a text into a single sentence, typically abstractively, and *headline generation* aims at summarizing the text into a headline, which tends to be shorter than a sentence.

Summarization is a subjective task (Rath et al., 1961; Lin and Hovy, 2002), requiring human input to assess performance. *Generic* summarization aims at creating a summary that is as reader-independent as possible, i.e. satisfying as many readers as possible. There has been some work on non-generic summarization, such as *query-based* and *topic-based* summarization, which bias the summary toward a query or a topic expressed by the intended reader (Hand, 1997).

Automated evaluation methods have been developed. The most widely used automated evaluation metric for summarization is ROUGE and its variants (Lin and Hovy, 2003; Lin, 2004), followed by METEOR (Banerjee and Lavie, 2005). Other metrics include Basic Elements (Hovy et al., 2005), LSA-based evaluation measures (Steinberger and Ježek, 2012) and SIRA (Cohan and Goharian, 2016). Ideally, one can perform human-based evaluation strategies, such as the pyramid method (Nenkova and Passonneau, 2004).

Given the diversity of summarization approaches, and assessment protocols, it may be challenging for researchers to identify the subset of corpora that are best-suited for a given summarization research task. In this paper, we attempt to solve this problem by presenting an overview of existing corpora, and evaluating their utility for common summarization tasks.

2. Corpora

2.1. Overview

Table 1 presents an overview of the main summarization corpora. The most widely used corpora are the Document Understanding Conference (DUC) and the Text Analysis Conference (TAC) corpora. The DUC corpora were released as part of the summarization shared task hosted at the Document Understanding Conference[2], which took place yearly from 2001 to 2007. Over et al. (2007) provides a detailed overview of the DUC 2001 to 2006 datasets. In 2008, DUC was replaced by the Text Analysis Conference[3], which is organized annually and had a summarization shared task in 2008, 2009, 2010, 2011, and 2014.

Historically, the summarization field has focused on generic extractive summarization. Prior to 2010, work in abstractive summarization had been quite limited (Ganesan et al., 2010). However, over the past few years, artificial neural networks have shown promising results for abstractive summarization. The DUC and TAC corpora, each of them having fewer than 1000 summaries, are too small to train neural networks (Nallapati et al., 2016b; Cheng and

[1]Some researchers use the terms extractive and compressive interchangeably. *Sentential extractive* is unambiguous.

[2]http://duc.nist.gov

[3]https://tac.nist.gov

Dataset	A/E	Lang.	Domain	Multi-doc	Size	Output length	Generic
DUC 2001 (Over and Yen, 2001)	a	en	news	both	60x10	50,100,200,400	y
DUC 2002 (Over and Liggett, 2002)	a,e	en	news	both	60x10	10,50,100,200,400	y
DUC 2003 (Over and Yen, 2003)	a	en	news	both	60x10,30x25	10,100	both
DUC 2004 (Over and Yen, 2004)	a	en,ar	news	both	100x10	10,100	both
DUC 2005 (Dang, 2005)	a	en	news	y	50x32	250	query-focused
DUC 2006 (Dang, 2006)	a	en	news	y	50x25	250	query-focused
DUC 2007 (Dang, 2007)	a	en	news	y	25x10	100	update
TAC 2008 (Dang and Owczarzak, 2008)	a	en	news	y	48x20	100	update,query
TAC 2009 (Dang and Owczarzak, 2009)	a	en	news	y	44x20	100	guided
TAC 2010 (Owczarzak and Dang, 2010)	a	en	news	y	46x20	100	guided
TAC 2011 (Owczarzak and Dang, 2011)	a	en	news	y	44x20	100	guided
ICSI (Janin et al., 2003)	a,e	en	meetings	n	57	390	y
AMI (McCowan et al., 2005)	a,e	en	meetings	n	137	300	y
Opinosis (Ganesan et al., 2010)	a	en	reviews	y	51x100	25	y
Gigaword (David and Cieri, 2003)	a	en	news	n	4,111,240	headline	y
Gigaword 5 (Parker and others, 2011)	a	en	news	n	9,876,086	headline	y
LCSTS (Hu et al., 2015)	a	zh	blogs	n	2,400,591	a few sentences	y
CNN/Daily Mail (Hermann et al., 2015)	a	en	news	n	312,084	50 average	y
MSR Abstractive (Toutanova et al., 2016)	a	en	misc	n	6,000	a few sentences	y
arXiv (Cohan et al., 2018)	a	en	science	n	194,000	220	y
PubMed (Cohan et al., 2018)	a	en	science	n	278,000	216	y

Table 1: Overview of existing datasets for summarization. Abbreviations; a: abstractive; ar: arabic; e: extractive; en: English; multi-doc: multi-document summarization; n: no; y: yes; zh: Chinese. The size is expressed in terms of number of summarized texts. For multi-document summarization corpora, 60x10 means that the corpus contains 60 clusters of documents, each of them is comprised of 10 documents. The output length corresponds to the length of the gold summaries (unless mentioned otherwise, the unit is word). For DUC 2001, 2002, 2003, and 2004, gold abstracts of different lengths are provided (e.g., 50, 100, 200, and 400 words). All datasets are freely available except the Gigaword corpora. Gigaword corpora are also available in Arabic, Chinese, French, German, and Spanish. Aside from Gigaword, any corpus that comprises texts and their titles may be used for title generation.

Lapata, 2016; Nallapati et al., 2017). As a result, recent studies have employed larger datasets, mostly based on Cable News Network (CNN), Daily Mail and Gigaword documents. In Table 2, we present a list of the corpora used in several studies that investigate the use of neural networks for summarization.

2.2. Converting abstractive summaries into extractive

Most corpora for summarization have abstractive summaries as gold-standard targets. In order to circumvent this limitation, several methods have been developed to convert an abstractive summary into an extractive summary. They rely on selecting sentences from the document that maximize a given metric with respect to gold abstractive summaries.

Methods differ with respect to the score, and the sentence selection strategies: Nallapati et al. (2016c) use ROUGE as the score, Cheng and Lapata (2016) use a semantic correspondence metric (Woodsend and Lapata, 2010), Nallapati et al. (2016c) use ROUGE as the score, and Cheng and Lapata (2016) use some semantic correspondence metric (Woodsend and Lapata, 2010). Nallapati et al. (2016c) use a greedy sentence selection approach, Cao et al. (2016a) rely on integer linear optimization for scoring, and Svore et al. (2007) train a neural network.

The choice of abstract-to-extract conversion method is one more parameter making challenging to compare published studies against each other. Note that for the evaluation, one can simply evaluate the predicted extractive summary against a gold abstractive summary with a typical summarization quality metric such as ROUGE, as (Nallapati et al., 2016c) did.

Converting abstractive summaries into extractive is often imperfect though. For example, Jing (2002) analyzed 300 news articles and showed that 19% of human-generated summary sentences contain no matching article sentence, and that only 42% of the summary sentences match the content of a single article sentence (with potentially a few semantic and syntactic modifications between the article sentence and the summary sentence).

2.3. Special types of summarization

There exist many other special types of summarization in addition to the traditional summarization tasks that we have mentioned earlier. These include:

- *Update summarization*: it aims at summarizing what changed between an old text and a more recent text.

Paper	A/E	Corpora
(Cohan et al., 2018)	a	arXiv, PubMed
(Narayan et al., 2017)	e	CNN with image captions
(Paulus et al., 2017)	a	CNN/DM
(See et al., 2017)	a	CNN/DM
(Nallapati et al., 2017)	a,e	CNN/DM, DUC 2002 (t)
(Nallapati et al., 2016c)	e	DM, DUC 2002 (t)
(Cheng and Lapata, 2016)	e	CNN/DM, DUC 2002 (t)
(Ayana et al., 2016)	a	Gw, DUC 2003-4 (t)
(Cao et al., 2016b)	e	DUC 2005, 2006, 2007
(Gu et al., 2016)	a	LCSTS
(Chopra et al., 2016)	a	Gw, DUC 2004
(Nallapati et al., 2016b)	a	Gw, DUC 2003+2004 (t)
(Nallapati et al., 2016a)	a	Gw
(Gulcehre et al., 2016)	a	Gw
(Ranzato et al., 2015)	a	subset of Gw
(Rush et al., 2015)	a	Gw, DUC 2003+2004
(Cao et al., 2015)	e	DUC 2001, 2002, 2004
(Yin and Pei, 2015)	e	DUC 2002 and DUC 2004
(Kågebäck et al., 2014)	e	Opinosis

Table 2: Overview of datasets used in recent studies developing neural network architectures for summarization. Abbreviations; a: abstractive; DM: Daily Mail; e: extractive; Gw: Gigaword (any version); (t): the dataset was used for test only, not training. DUC corpora are typically used for testing only, as they tend to be too small to train neural networks on.

TAC 2008 and 2009 had an update summarization track (Dang and Owczarzak, 2008). The Text Retrieval Conference (TREC) also organized an update summarization shared task yearly from 2013 to 2017, which they sometimes referred to as *temporal summarization* (Aslam et al., 2013; Aslam et al., 2015a; Aslam et al., 2015b) or *real-time summarization* (Lin et al., 2016).

- *Source code summarization*: it aims at either summarizing in a human language what a code snippet performs, or automatically folding blocks of code that are deemed less informative (also referred as the *autofolding problem*). Iyer et al. (2016) compiled a corpus from StackOverflow to summarize source code into English. Fowkes et al. (2017) presented a system to perform autofolding and created a corpus based on the source code of the top six most popular Java projects on GitHub.

- *Overview synthesis*: the task is very similar to multi-document summarization, except that the output is much longer than a typical summary. Zhang and Wan (2017) constructed a corpus based on Wikinews, where each Wikinews is regarded as the gold overview, while the linked news articles are the input of the overview synthesis system.

- *Sentence fusion*: this task is also very similar to multi-document summarization, except that the input is two sentences, and the output is one sentence. It has been shown that generic sentence fusion may lead to a low agreement between humans (Daume III and Marcu, 2004). Sentence fusion may be used to convert an extractive summary into a more abstractive summary (Barzilay and McKeown, 2005).

- *Sentence compression*: the objective is to summarize one single sentence, either abstractively or extractively. Filippova and Altun (2013) constructed the first large corpus for this task, containing 250,000 pairs of sentences. They later created a larger corpus, containing around 2 million pairs, but only 10,000 were publicly released (Filippova et al., 2015).

- *Concept-map-based summarization*: the task is to create a concept map from a text. Falke and Gurevych (2017) created a corpus of 30 educational topics, each containing around 40 source documents and a summarizing concept map that is the consensus of several crowdworkers.

Summarization may also be performed for non-textual input, such as single images (Fan et al., 2008), albums of images (Yu et al., 2017), videos (Evangelopoulos et al., 2008), or voice recording (e.g., meetings or presentations) (Zhang et al., 2007). Different type of inputs may also be combined to perform the summary, which is a task referred to as *multi-modal summarization* (Li et al., 2017).

2.4. Meta-information

When choosing a suitable corpus to train or evaluate a summarization algorithm, many parameters must be taken into account, including:

- Domain of the texts: the majority of corpora concentrate on news aticles. This is a significant shortcoming as supervised models trained on news aticles may have poor performances when applied to another domain. It also limits the evaluation of summarization algorithms to a particular domain.

- Type of gold summaries: abstractive, or extractive. We review in Section 2.2. several methods to convert an abstractive summary into an extractive summary, as most corpora are abstractive.

- Number of gold summaries per texts: typically a corpus contains one gold summary per text in the case of single-document summarization, or one gold summary per group of texts (often referred as *topic* or *cluster*) in the case of multi-document summarization. If each text has more than one gold summary, the corpus may be referred to as *multi-reference* (Toutanova et al., 2016). Note that the task of extractive single-document summarization may be counterintuitively more difficult than extractive multi-document summarization (Nenkova, 2005).

- Language: most existing corpora are in English. The only large-scale, non-English corpus are LCSTS

(Large Scale Chinese Short Text Summarization), which is in Chinese, and the Gigaword corpora, which are available in English, Arabic, Chinese, French, German, and Spanish.

- Length of the text to summarize: typically from a few sentences to a few pages. If the text to summarize is a single sentence, the task is often referred as *sentence compression*, even if the summary is abstractive (Cohn and Lapata, 2013)[4].

- Length of the reference summaries: it typically varies from a headline (headline generation) to several sentences (multi-sentence summary).

- Summarization intent: generic, or non-generic such as query-based or topic-based. Query-based summarization may be viewed as a form of question answering task.

- Presence of side information: some corpora may provide some side information in addition to the text to summarize. For example, Narayan et al. (2017) created a corpus based on CNN news articles that incorporate image captions in addition to the texts of the articles. It is however uncommon.

- Price, license, and access: corpora vary in terms of price, license, and access method. Fortunately, the vast majority of summarization corpora is freely available, with the notable exceptions of the Gigaword corpora, and LCSTS (free for research, potentially non-free for commercial use).

- Number of other studies using it: as a more direct way to assess the popularity of a corpus, one can look at the number of papers that used it. One has to keep in mind that as a result of the evolution of summarization algorithms and research interests, the most used corpora may change over time, as Table 2 shows.

2.5. Repository

The LRE map (Calzolari et al., 2012) contains a list of summarization datasets. However, we found it to have two significant limitations: 1) a few technical issues 2) lack of many summarization-specific metainformation, since it has to support any type of corpus.

In light of the increasing number of summarization corpora, as well as the amount of summarization-specific metainformation, we have created a repository for summarization corpora[5]. The repository aims at providing researchers a synopsis of existing corpora, by displaying metainformation for each corpus. We encourage contributions from anyone, either to improve the metainformation of listed corpora, or adding a new corpus.

[4]If the sentence compression is not abstractive, one can refer to it as *deletion-based* sentence compression (Filippova et al., 2015)

[5]`https://github.com/Franck-Dernoncourt/summarization-corpora`

3. Conclusion

In this paper, we have presented an overview of the main corpora for summarization, and introduced a repository aiming to list corpora for summarization as well as their metainformation. There exist many corpora, but most of them are small and cannot be used to train neural networks. More large-scale corpora for summarization are needed. Furthermore, each corpus has its own data organization; creating a data standard for summarization corpora would make research more efficient.

4. Bibliographical References

Aslam, J., Diaz, F., Ekstrand-Abueg, M., McCreadie, R., Pavlu, V., and Sakai, T. (2013). Trec 2013 temporal summarization.

Aslam, J., Diaz, F., Ekstrand-Abueg, M., McCreadie, R., Pavlu, V., and Sakai, T. (2015a). TREC 2014 temporal summarization track overview. Technical report, National Institute of Standards and Technology.

Aslam, J., Diaz, F., Ekstrand-Abueg, M., McCreadie, R., Pavlu, V., and Sakai, T. (2015b). TREC 2015 temporal summarization track overview. Technical report, National Institute of Standards and Technology.

Ayana, S. S., Liu, Z., and Sun, M. (2016). Neural headline generation with minimum risk training. *arXiv preprint arXiv:1604.01904*.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72.

Barzilay, R. and McKeown, K. R. (2005). Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Calzolari, N., Del Gratta, R., Francopoulo, G., Mariani, J., Rubino, F., Russo, I., and Soria, C. (2012). The lre map. harmonising community descriptions of resources. In *LREC*, pages 1084–1089.

Cao, Z., Wei, F., Dong, L., Li, S., and Zhou, M. (2015). Ranking with recursive neural networks and its application to multi-document summarization. In *AAAI*, pages 2153–2159.

Cao, Z., Chen, C., Li, W., Li, S., Wei, F., and Zhou, M. (2016a). Tgsum: Build tweet guided multi-document summarization dataset. In *AAAI*, pages 2906–2912.

Cao, Z., Li, W., Li, S., Wei, F., and Li, Y. (2016b). Attsum: Joint learning of focusing and summarization with neural attention. *arXiv preprint arXiv:1604.00125*.

Cheng, J. and Lapata, M. (2016). Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252*.

Chopra, S., Auli, M., and Rush, A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Cohan, A. and Goharian, N. (2016). Revisiting summarization evaluation for scientific articles. *Language Resources and Evaluation Conference (LREC)*.

Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., and Goharian, N. (2018). A discourse-aware attention model for abstractive summarization of long documents. Submitted to NAACL.

Cohn, T. and Lapata, M. (2013). An abstractive approach to sentence compression. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(3):41.

Dang, H. T. and Owczarzak, K. (2008). Overview of the tac 2008 update summarization task. In *Proc. of the First Text Analysis Conference*.

Daume III, H. and Marcu, D. (2004). Generic sentence fusion is an ill-defined summarization task. In *Proceedings of the ACL Text Summarization Branches Out Workshop*, pages 96–103.

Evangelopoulos, G., Rapantzikos, K., Potamianos, A., Maragos, P., Zlatintsi, A., and Avrithis, Y. (2008). Movie summarization based on audiovisual saliency detection. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2528–2531. IEEE.

Falke, T. and Gurevych, I. (2017). Bringing structure into summaries: Crowdsourcing a benchmark corpus of concept maps. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2969–2979. Association for Computational Linguistics.

Fan, J., Gao, Y., Luo, H., Keim, D. A., and Li, Z. (2008). A novel approach to enable semantic and visual image summarization for exploratory image search. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 358–365. ACM.

Filippova, K. and Altun, Y. (2013). Overcoming the lack of parallel data in sentence compression. In *EMNLP*, pages 1481–1491.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence compression by deletion with lstms. In *EMNLP*, pages 360–368.

Fowkes, J., Chanthirasegaran, P., Ranca, R., Allamanis, M., Lapata, M., and Sutton, C. (2017). Autofolding for source code summarization. *IEEE Transactions on Software Engineering*.

Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd international conference on computational linguistics*, pages 340–348. Association for Computational Linguistics.

Gu, J., Lu, Z., Li, H., and Li, V. O. (2016). Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*.

Gulcehre, C., Ahn, S., Nallapati, R., Zhou, B., and Bengio, Y. (2016). Pointing the unknown words. *arXiv preprint arXiv:1603.08148*.

Hand, T. F. (1997). A proposal for task-based evaluation of text summarization systems. *Intelligent Scalable Text Summarization*.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Hovy, E., Lin, C.-Y., and Zhou, L. (2005). Evaluating duc 2005 using basic elements. In *Proceedings of DUC*, volume 2005.

Hu, B., Chen, Q., and Zhu, F. (2015). Lcsts: A large scale chinese short text summarization dataset. *arXiv preprint arXiv:1506.05865*.

Iyer, S., Konstas, I., Cheung, A., and Zettlemoyer, L. (2016). Summarizing source code using a neural attention model. In *ACL (1)*.

Jing, H. (2002). Using hidden markov modeling to decompose human-written summaries. *Computational linguistics*, 28(4):527–543.

Kågebäck, M., Mogren, O., Tahmasebi, N., and Dubhashi, D. (2014). Extractive summarization using continuous vector space models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@ EACL*, pages 31–39.

Li, H., Zhu, J., Ma, C., Zhang, J., and Zong, C. (2017). Multi-modal summarization for asynchronous collection of text, image, audio and video. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1113. Association for Computational Linguistics.

Lin, C.-Y. and Hovy, E. (2002). Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4*, pages 45–51. Association for Computational Linguistics.

Lin, C.-Y. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.

Lin, J., Roegiest, A., Tan, L., McCreadie, R., Voorhees, E., and Diaz, F. (2016). Overview of the TREC 2016 real-time summarization track. In *Proceedings of the 25th Text REtrieval Conference, TREC*, volume 16.

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the Association for Computational Linguistic workshop*, volume 8.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Nallapati, R., Xiang, B., and Zhou, B. (2016a). Sequence-to-sequence rnns for text summarization. *arXiv preprint arXiv:1602.06023*.

Nallapati, R., Zhou, B., dos Santos, C., glar Gulçehre, Ç., and Xiang, B. (2016b). Abstractive text summarization using sequence-to-sequence rnns and beyond. *CoNLL 2016*, page 280.

Nallapati, R., Zhou, B., and Ma, M. (2016c). Classify or select: Neural architectures for extractive document summarization. *arXiv preprint arXiv:1611.04244*.

Nallapati, R., Zhai, F., and Zhou, B. (2017). Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *AAAI*.

Narayan, S., Papasarantopoulos, N., Lapata, M., and Cohen, S. B. (2017). Neural extractive summarization with side information. *arXiv preprint arXiv:1704.04530*.

Nenkova, A. and Passonneau, R. (2004). Evaluating content selection in summarization: The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-NAACL 2004*.

Nenkova, A. (2005). Automatic text summarization of newswire: Lessons learned from the document understanding conference. In *AAAI*, volume 5, pages 1436–1441.

Over, P., Dang, H., and Harman, D. (2007). DUC in context. *Information Processing & Management*, 43(6):1506–1520.

Paulus, R., Xiong, C., and Socher, R. (2017). A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2015). Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Rath, G., Resnick, A., and Savage, T. (1961). The formation of abstracts by the selection of sentences. part i. sentence selection by men and machines. *Journal of the Association for Information Science and Technology*, 12(2):139–141.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Steinberger, J. and Ježek, K. (2012). Evaluation measures for text summarization. *Computing and Informatics*, 28(2):251–275.

Svore, K. M., Vanderwende, L., and Burges, C. J. (2007). Enhancing single-document summarization by combining ranknet and third-party sources. In *EMNLP-CoNLL*, pages 448–457.

Toutanova, K., Brockett, C., Tran, K. M., and Amershi, S. (2016). A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In *EMNLP*, November.

Woodsend, K. and Lapata, M. (2010). Automatic generation of story highlights. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 565–574. Association for Computational Linguistics.

Yin, W. and Pei, Y. (2015). Optimizing sentence modeling and selection for document summarization. In *IJCAI*, pages 1383–1389.

Yu, L., Bansal, M., and Berg, T. (2017). Hierarchically-attentive rnn for album summarization and storytelling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 977–982. Association for Computational Linguistics.

Zhang, J. and Wan, X. (2017). Towards automatic construction of news overview articles by news synthesis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2101–2106. Association for Computational Linguistics.

Zhang, J. J., Chan, H. Y., and Fung, P. (2007). Improving lecture speech summarization using rhetorical information. In *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, pages 195–200. IEEE.

5. Language Resource References

Dang, H. T. and Owczarzak, K. (2008). Overview of the tac 2008 opinion question answering and summarization tasks. In *Proc. of the First Text Analysis Conference*, volume 2.

Dang, H. and Owczarzak, K. (2009). Overview of the tac 2009 summarization track (draft). In *In Proceedings of the Second Text Analysis Conference (TAC2009)*.

Dang, H. T. (2005). Overview of DUC 2005.

Dang, H. T. (2006). Overview of DUC 2006.

Dang, H. T. (2007). Overview of DUC 2007.

David, G. and Cieri, C. (2003). English gigaword LDC2003t05. *Linguistic Data Consortium*.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., et al. (2003). The icsi meeting corpus. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–I. IEEE.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., et al. (2005). The AMI meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88.

Over, P. and Liggett, W. (2002). Introduction to DUC-2002: An intrinsic evaluation of generic news text summarization systems. *ACL 2002, Workshop on Text Summarization*.

Over, P. and Yen, J. (2001). Introduction to DUC-2001: an intrinsic evaluation of generic news text summarization systems.

Over, P. and Yen, J. (2003). Introduction to DUC-2003: an intrinsic evaluation of generic news text summarization systems.

Over, P. and Yen, J. (2004). Introduction to DUC-2004: an intrinsic evaluation of generic news text summarization systems.

Owczarzak, K. and Dang, H. T. (2010). Overview of the tac 2010 summarization track. In *Proceedings of the Third Text Analysis Conference, Gaithersburg, Maryland, USA. National Institute of Standards and Technology.*

Owczarzak, K. and Dang, H. T. (2011). Overview of the tac 2011 summarization track: Guided task and aesop

task. In *Proceedings of the Text Analysis Conference (TAC 2011), Gaithersburg, Maryland, USA, November*.

Parker, R. et al. (2011). English gigaword fifth edition LDC2011T07. *Linguistic Data Consortium*.

auto-*h*MDS: Automatic Construction of a Large Heterogeneous Multilingual Multi-Document Summarization Corpus

Markus Zopf
Research Training Group AIPHES
Department of Computer Science, Technische Universität Darmstadt
Hochschulstraße 10, 64289 Darmstadt, Germany
zopf@aiphes.tu-darmstadt.de

Abstract

Automatic text summarization is a challenging natural language processing (NLP) task which has been researched for several decades. The available datasets for multi-document summarization (MDS) are, however, rather small and usually focused on the newswire genre. Nowadays, machine learning methods are applied to more and more NLP problems such as machine translation, question answering, and single-document summarization. Modern machine learning methods such as neural networks require large training datasets which are available for the three tasks but not yet for MDS. This lack of training data limits the development of machine learning methods for MDS. In this work, we automatically generate a large heterogeneous multilingual multi-document summarization corpus. The key idea is to use Wikipedia articles as summaries and to automatically search for appropriate source documents. We created a corpus with 7,316 topics in English and German, which has variing summary lengths and variing number of source documents. More information about the corpus can be found at the corpus GitHub page at https://github.com/AIPHES/auto-hMDS.

Keywords: Summarization, Corpus, Wikipedia

1. Motivation

More and more data is contained in unstructured information sources such as newswire articles, social media posts and micro-blogging messages. No human is able to process all the data belonging to important topics such as news about elections, opinions about the newest smartphone, statements in political discussions, trending topics in research, or natural disasters. Automatic preparation of information from heterogeneous sources is therefore a key challenge to enable humans to make use of all the data available on the Internet.

Previous work on automatic summarization usually use small and homogeneous datasets to evaluate their models. The application of supervised machine learning methods is limited mainly by the size of the datasets. This is in particularly true for abstractive summarization methods, which are usually trained on the only available large single-document summarization (SDS) corpus (Hermann et al., 2015). Recently, Zopf et al. (2016b) proposed a new method to create large multi-document summarization (MDS) corpora. Instead of using humans to write summaries for a specific topic based on previously collected source documents, they propose to use already available documents which can be considered to be summaries and search for appropriate source documents. With this method, Zopf et al. (2016b) created a heterogeneous multi-document summarization corpus manually.

In this work, we investigate if and how the manually performed process described in Zopf et al. (2016b) can be automated to create a large heterogeneous multi-document summarization corpus. Furthermore, we add German topics in addition to English topics to the new corpus. Most summarization corpora only contain English source documents and summaries. We call the newly created corpus auto-*h*MDS.

The analysis of our newly created corpus show that our corpus is indeed much larger than prior corpora. We show that a simple machine learning method can improve their performance if they are provided with more training data. We also provide results of standard baseline summarization methods to generate a reference point for future research.

2. Related Work

A popular subfield in automatic summarization is multi-document summarization with a focus on newswire articles. Popular multi-document summarization corpora were created for the Document Understanding Conference (DUC) shared tasks (Over et al., 2007). The datasets from the 2001-2004 shared tasks are often used to evaluate summarization systems (Erkan and Radev, 2004; Lin and Bilmes, 2011; Cao et al., 2015; Ren et al., 2016). Furthermore, they were used to evaluate the ROUGE and Pyramid evaluation system (Lin, 2004; Passonneau et al., 2005; Nenkova et al., 2007). More MDS corpora were produced for the Text Analysis Conference (TAC) 2008 and 2009 shared tasks and used by popular summarization models (Gillick et al., 2009) as well. The DUC and TAC corpora are rather small. They usually contain about 50 topics and are therefore too small to be used for training by machine learning models. Due to the small size, the evaluation is also problematic. Summarization models are usually evaluated with noisy automatic evaluation systems (Lin, 2004). Therefore, the evaluation results might be inaccurate if the summarization models are only evaluated on a few topic.

Recently, a large single-document summarization corpus (Hermann et al., 2015) has been published. Due to its size, it was used to train both extractive (Zopf et al., 2016a; Nallapati et al., 2017) and abstractive summarization (See et al., 2017; Tan et al., 2017) models. We see that as soon as larger corpora are available, the development of other summarization models becomes possible. This trend can-

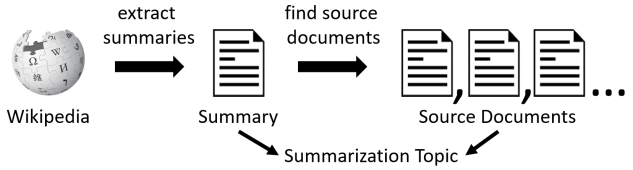


Figure 1: Illustration of the previously proposed corpus construction approach. Wikipedia articles are used as reference summaries (left). Summarization topics can be created in combination with automatically retrieved source documents (right).

not only be observed in summarization. Work on sentence compression (Chopra et al., 2016; Li et al., 2017) use the large Gigaword corpus (Napoles et al., 2012). The small DUC 2004 dataset is only used for testing, not for training. Large datasets are, in addition to new models, a key driver for new AI breakthroughs.[1]

We summarize that available MDS corpora are rather small. Large datasets for single-document summarization are also rare (See et al., 2017). Recent machine learning methods can make use of large SDS corpora for training. Even corpora which do not contain summaries such as the Gigaword corpus can be used as silver standard to train machine learning models. Large heterogeneous corpora are not available (Zopf et al., 2016b).

3. Corpus Construction

The system presented in this work automates the manually performed work in Zopf et al. (2016b). The general idea of the approach is illustrated in Figure 1.

Instead of coming up with topics, searching for source documents, and writing summaries for the source documents, Zopf et al. (2016b) proposed to select already available summaries from Wikipedia and to search for appropriate source documents. The advantage of the new process is that no new text has to be written since the seed for a topic are not the source documents but an already available summary. Since no summaries have to be written, which is usually a difficult and time-consuming task, it seems to be possible to automate the proposed process. In the following, we briefly explain the work performed by Zopf et al. (2016b) and describe how we automate this process.

3.1. Extracting Topics and Summaries

In a first step, Zopf et al. (2016b) select already available text documents on the Internet which can be considered to be a summary of a topic. They use the Wikipedia featured articles as source for summary texts, since they are (i) well-written, (ii) comprehensive, (iii) well-researched, (iv) neutral, and (v) stable according to the Wikipedia featured article criteria[2]. In particular, the first section of each featured article (also called the lead section) is supposed to be a good summary of the topic according to the Wikipedia guidelines. The lead of each featured article contains the most important information about a topic. (Zopf et al., 2016b) extracted 91 English featured article leads manually. We perform this task automatically and retrieved all currently available lead sections of all English and German Wikipedia articles. After creating a list with all Wikipedia featured articles, we use the MediaWiki Action API[3] with properties `extracts|info&exintro&explaintext` to retrieve the lead parts of 7,613 Wikipedia articles. Every article is the seed for one summarization topic. We use the full lead section of each article as summary and do not truncate longer lead sections. The lengths of different summaries can therefore vary a lot. This is an additional new property of the auto-*h*MDS corpus compared to traditional corpora such as the DUC and TAC corpora. These corpora usually have a fixed length for summaries (e.g. 100 words or 665 characters). In the summarization setting provided by auto-*h*MDS, summarization systems have to be able to generate short and long summaries. Details of the summary properties can be found in Section 4.

3.2. Finding Source Documents

In a second step, Zopf et al. (2016b) annotate information nuggets manually. Each extracted information nugget is considered to contain an important piece of information about the topic. The information nuggets are used together with the topic name as query terms in a web search engine to retrieve source documents for the topics. Since the extracted information nugget is used in the web search, the retrieved documents contain the information nugget and can therefore be considered to be source documents for the topics. The cited references in Wikipedia have not been used since we found that they often do not contain the suggested information or were not reachable anymore. By using search results containing the information nuggets, it can be ensured that the link is still alive and that the suggested information is contained in the source document.

The nuggets were labeled by humans in the work performed by Zopf et al. (2016b). Since we aim at creating the corpus fully automatically, a manual labeling step was not appropriate. Instead of extracting information nuggets, we use the sentences contained in the lead sections together with the topic name as search terms. We found that this strategy works very well since sentences in Wikipedia leads are usually rather short and often focused on one piece of information. In the manual extraction performed by Zopf et al. (2016b), usually only one nugget was annotated per sentence which further indicates that searching for sentences is similar to searching for information nuggets. Furthermore, a lot of web pages reuse sentences taken from Wikipedia. This simplifies the web search since it becomes easier to find web pages which fit to the sentences in the lead sections.

To split the lead sections of the retrieved Wikipedia articles, we used the Stanford Segmenter for the English documents and the OpenNLP Segmenter for the German documents. Both are available in the DKPro Core library (Eckart de Castilho and Gurevych, 2014) in version 1.8.0. We found

[1] `https://www.edge.org/response-detail/26587`

[2] `https://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria`

[3] `https://www.mediawiki.org/wiki/Special:MyLanguage/API:Action_API`

that the automatic sentence splitting worked very well, perhaps also due to the high quality of the Wikipedia featured articles.

For all sentences in all Wikipedia articles, we use the Google Custom Search Engine[4] (CSE) to search for source documents. We use the topic name together with the sentence as query term for the CSE. The *rights* field of the CSE is used to only find sources which can be freely used for non-commercial use-cases. In total, we performed about 93k searches. Since Google charges $5 per 1000 queries, the retrieval of the links for all the sentences costs about $500. This is fairly cheap compared to the costs for paying humans to come up with topics, search for source documents, and write summaries as it was performed for other multi-document summarization corpora.

3.3. Retrieving Source Documents

The result of the invocation of the Google CSE are link lists pointing to web pages which contain the provided query terms (topic name + sentence text). For each sentence, we retrieved up to 10 links. Since some of the sentences occur only rarely on the Internet, we did not obtain 10 links for each query. We obtained about 550k links, in average 5.90 links per query. For each sentence, we tried to downloaded the first web page in the query result. If the pages was not available, we continued with the next URL until we were able to download a page or reached the end of the query result list. To retrieve the best possible snapshot of the web page, we did not only download the HTML code of the web page, but rendered every web page using the Google Chrome browser. We used the Selenium[5] framework to interact with the browser programmatically. This creates better snapshots of web pages since dynamic content can be created or modified (e.g. with JavaScript) before the snapshot is taken. It turned out that using a browser to retrieve the page content improves the quality of the snapshots in particular for web pages which use a lot of JavaScript such as Youtube.

4. Analysis

In this section, we provide a detailed analysis of the result of the effort to automate the construction approach proposed by (Zopf et al., 2016b) and compare the result with previously constructed corpora.

4.1. Corpus Size

In total, we created 5,132 English and 2,481 German topics. Every topics contains one reference summary file, one file which contains one sentence per line (constructed with automatic sentences splitting), and for each sentence a list of URLs. 71,162 and 22,303 sentences are contained in the English and German summaries, respectively. We found 473,754 (in average 6.66 per sentence) and 75,594 (in average 3.39 per sentence) URLs for the German and English corpora, respectively. English and German lead sections have an average length of 13.87 and 8.99 sentences.

[4] https://developers.google.com/custom-search/json-api/v1/overview

[5] http://www.seleniumhq.org

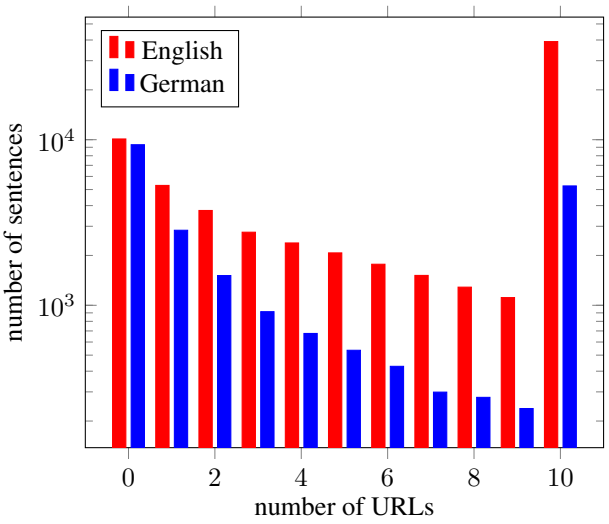


Figure 2: Distribution of retrieved URLs for sentences.

Figure 2 shows a distribution of number of URLs per sentences. We were not able to retrieve results for a significant number of sentences with the search engine in particular for German sentences (English: 10,102 (14.20%), German: 9,325 (41.81%) sentences with an empty search results). One reason for this is the search engine configuration since we only aimed at retrieving only URLs which can be freely used for non-commercial use-cases. We assume that results can be retrieved for almost all sentences if the search is not restricted. Important to note is that lacking source documents is no limitation of the quality of the corpus. Since source documents are missing, the best summary which can generated based on the source documents might not be as good as the reference summary. This, however, only reduces the upper bound reachable by summarization systems without reducing the corpus quality.

Based on the collected URLs, we tried to retrieve one source document for each sentences as described in Section 3.3.. We removed all topics for which we were not able to retrieve any source documents. This resulted in a final corpus size of 5,106 English and 2,210 German summarization topics. Figure 3 shows the distribution of number of source documents across the remaining topics.

	DUC04	TAC09	*h*MDS	auto-*h*MDS
topics	50	44	91	**7,316**
sources	500	440	1,265	**64,744**

Table 1: Size comparison of two standard multi-document summarization datasets, DUC 2004 and TAC 2009, the *h*MDS dataset and the in this paper presented auto-*h*MDS. We report the number of summarization topics and the number of source documents in the corpora.

The main goal of automating the corpus construction is to be able to generate a large corpus for training and evaluating machine learning models. We therefore compare the size of the generated corpus with the DUC 2004[6] and TAC

[6] http://duc.nist.gov/duc2004

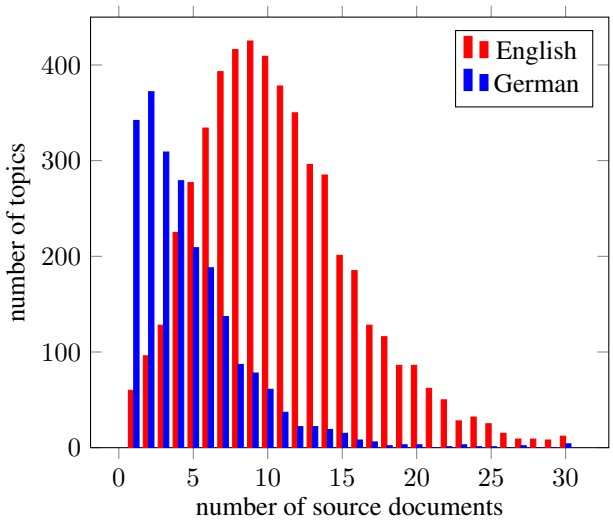


Figure 3: Distribution of source documents. ”30” means ”30 or more”.

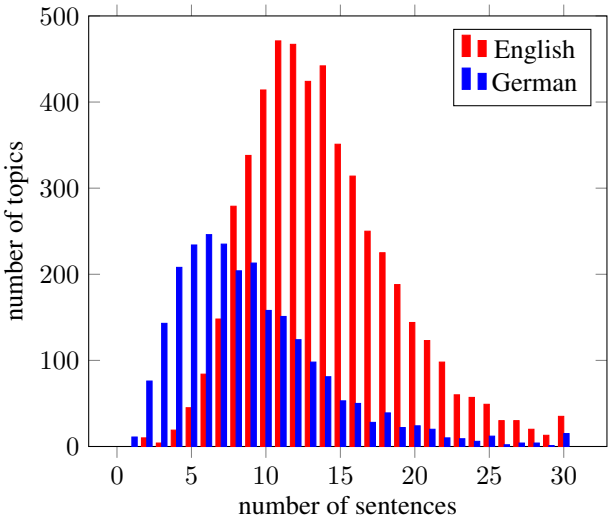


Figure 4: Distribution of sentence lengths in the reference summaries. ”30” means ”30 or more”.

2009[7] multi-document summarization datasets. Both are widely used standard MDS corpora. Other MDS corpora such as other DUC and other TAC corpora have similar sizes. Furthermore, we compare to the manually created *h*MDS corpus. Details are provided in Table 1.

The auto-*h*MDS corpus has many more topics than traditional summarization corpora. In total, we created 7,316 summarization topics. auto-*h*MDS is over 80 times larger in terms of topics (50 times larger in terms of source document) than *h*MDS and about 150 times larger (130 times larger) than standard MDS datasets.

		DUC04	TAC09	*h*MDS	auto-*h*MDS
word	src	672.15	633.89	2972.12	5862.51
	sum	118.12	110.15	245.52	312.42
sent	src	26.28	24.58	268.15	271.36
	sum	6.61	6.16	9.05	12.54

Table 2: Length comparisons according to average number words (word) and average number of sentences (sent) in the source documents (src) and the summaries (sum).

Table 2 provides details about the lengths of source documents and summaries according to the number of words and number of sentences in the corpora. Source documents and summaries in the auto-*h*MDS corpus are longer in comparison to the *h*MDS corpus.

As described in Section 3.1., the length of our summaries can vary in comparison to traditional datasets. Figure 4 illustrates the distribution of sentence lengths for both the English and the German part of the corpus. The English part of the corpus contains longer summaries than the German part. In general, the summaries in auto-*h*MDS are much longer than in traditional corpora where the summary lengths usually range from about 5 to 7 sentences.

4.2. Usability as Training Data

One motivation of building a large summarization corpus is to provide researchers with training data for building summarization systems using machine learning. To evaluate if our corpus can be used by machine learning models, we run experiments where we use the high quality but small *h*MDS corpus and a part of the large automatically created auto-*h*MDS corpus as training data for the summarization model proposed in Zopf et al. (2016a). We evaluate the learned model with the TAC 2009 multi-document summarization corpus. In the experiment, we created short summaries with a maximum length of 50 words. We use different ROUGE-based metrics (Lin, 2004) for evaluation. The results of the experiments are shown in Table 3. In the column *training data*, we report which corpus was used for training as well as how many topics from the corpus were used in brackets.

training data	ROUGE-1	ROUGE-2
*h*MDS (50)	0.2284	0.0307
*h*MDS (91)	**0.2361**	**0.0329**
auto-*h*MDS (50)	0.2362	0.0320
auto-*h*MDS (91)	0.2357	0.0324
auto-*h*MDS (100)	0.2404	0.0328
auto-*h*MDS (200)	0.2481	0.0326
auto-*h*MDS (300)	**0.2485**	**0.0336**

Table 3: Summarization results in the TAC 2009 corpus.

The results indicate that both *h*MDS and auto-*h*MDS can be used similarly well as training data even though the construction of auto-*h*MDS is much cheaper since we do not require any human annotations or interactions. Using only 50 training topics from the *h*MDS corpus yields lower results than using all 91 available topics suggesting that more training data helps the model to learn better. We observe the same effect for the auto-*h*MDS corpus: more training data leads to better results. The best results are achieved with the automatically constructed corpus meaning that the model is able to make better use of more medium-quality training data compared to less high-quality training data. Due to computational limitations, we were only able to use 300 topics of the newly created corpus as training data. We hope that new models which are better suited to learn form

[7] https://tac.nist.gov/2009/Summarization

large multi-document summarization training data will perform even better if more training topics are used.

4.3. Summarization Experiments

The created corpus can not only be used for training machine learning models but can also be used as dataset to test the performance of summarization models. Since the corpus is much larger than traditional corpora and covers a very wide variate of topics and genres, we expect a better performance estimation of summarization systems. Since ROUGE (Lin, 2004) also does not always estimate the performance accurately (Owczarzak et al., 2012), more topics help to improve accuracy in particularly if ROUGE is used as evaluation metric.

In Table 4, we report results for 4 simple extractive summarization baselines which can serve as reference points for future research experiments.

	100 words		200 words	
system	R-1	R-2	R-1	R-2
Random	0.1857	0.0185	0.2553	0.0325
Lead	0.1229	0.0261	0.1056	0.0228
ROUGE-1	0.4302	0.2161	0.4769	0.2117
ROUGE-2	0.4594	0.2927	0.4864	0.2924
Random	0.2290	0.0286	0.2841	0.0434
Lead	0.2524	0.0699	0.2676	0.0790
ROUGE-1	0.5601	0.3812	0.5168	0.3022

Table 4: Summarization performance of different summarization systems in the auto-*h*MDS corpus for different summary lengths (100 and 200 words) and different ROUGE versions (ROUGE-1 and ROUGE-2) for the German (top) and the English (bottom) part of the corpus.

The *Random* baseline chooses sentences randomly until the summary reached the desired length. *Lead* uses the first sentences of the source documents. *ROUGE-1* and *ROUGE-2* choose the best sentences in a greedy fashion according to the ROUGE-1 recall and ROUGE-2 recall score of individual sentences. To compute the scores, both summarization systems use the reference summary. Therefore, both *ROUGE-1* and *ROUGE-2* cannot be considered to be competitive summarization systems but are rather indicators for the best possible scores which can be achieved.

We observe that there is a large performance gap between the *Random* and the *Lead* baselines and the upper bounds achieved by *ROUGE-1* and *ROUGE-2*. This is a promising result since it indicates that the area between random guessing and a very good summarization system is large. The corpus will therefore be usable to distinguish between good and bad summarization systems.

Another interesting observation is that ROUGE scores for the English part are higher than in the German part of the corpus. Not only the baselines achieve higher scores but also the upper bound seems to be higher for English texts.

4.4. Heterogeneity

Last but not least, we analyze the heterogeneity of the created corpus. The topics belong to very diverse genres such as history, religion, sports, science, transport, music, culture, etc. and are therefore even more diverse than the topics in (Zopf et al., 2016b) in which all topics belong to the three genres (i) Art, Architecture, and Archeology (ii) History, and (iii) Law, Politics, and Government. Since we collected all featured articles from Wikipedia, we generated a corpus which covers a lot of interesting topics. The more people are interested in a topic, the more people collect information and work on the according Wikipedia article resulting in high quality articles for generally interesting topics.

5. Conclusions

In this paper, we presented a large, automatically generated multi-document summarization corpus containing topics in English and German. Usually, MDS corpora are rather small and their applicability as training data for machine learning models is limited. Our corpus is much larger than prior corpora and therefore closes this dataset gap. We showed that a machine learning model is indeed able to use the newly created corpus as training data. We hope that larger MDS training corpora will enable researchers to build and train better supervised machine learning models for automatic summarization similarly as we see this trend in single-document summarization with the availability of the large training corpus proposed by Hermann et al. (2015). We make both summaries and link lists containing the links to the source documents freely available. The retrieved source documents are available upon request. Further information can be found at the corpus GitHub page `https://github.com/AIPHES/auto-hMDS`.

6. Acknowledgments

This work has been supported by the German Research Foundation (DFG) as part of the Research Training Group Adaptive Preparation of Information from Heterogeneous Sources (AIPHES) under grant No. GRK 1994/1.

7. Bibliographical References

Cao, Z., Wei, F., Li, S., Li, W., Zhou, M., and Wang, H. (2015). Learning Summary Prior Representation for Extractive Summarization. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 829–833.

Chopra, S., Auli, M., and Rush, A. M. (2016). Abstractive Sentence Summarization with Attentive Recurrent Neural Networks. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, 2(1):1–11.

Erkan, G. and Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Gillick, D., Favre, B., Hakkani-Tür, D., Bohnet, B., Liu, Y., and Xie, S. (2009). The ICSI/UTD Summarization System at TAC 2009. In *Proceedings of the Second Text Analysis Conference*.

Hermann, K., Kocisky, T., and Grefenstette, E. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Li, P., Lam, W., Bing, L., and Wang, Z. (2017). Deep Recurrent Generative Decoder for Abstractive Text Summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2081–2090.

Lin, H. and Bilmes, J. (2011). A Class of Submodular Functions for Document Summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 510–520.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 25–26.

Nallapati, R., Zhai, F., and Zhou, B. (2017). SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3075–3081.

Napoles, C., Gormley, M., and Durme, B. V. (2012). Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction & Web-scale Knowledge Extracti*, pages 95–100.

Nenkova, A., Passonneau, R., and McKeown, K. (2007). The Pyramid Method. In *ACM Transactions on Speech and Language Processing*, volume 4, pages 1–23.

Over, P., Dang, H., and Harman, D. (2007). DUC in context. *Information Processing and Management*, 43(6):1506–1520.

Owczarzak, K., Conroy, J. M., Dang, H. T., and Nenkova, A. (2012). An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9.

Passonneau, R., Nenkova, A., McKeown, K., and Sigelman, S. (2005). Applying the pyramid method in DUC 2005. In *Proceedings of the Document Understanding Conference*.

Ren, P., Wei, F., and Chen, Z. (2016). A Redundancy-Aware Sentence Regression Framework for Extractive Summarization. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 33–43.

See, A., Liu, P. J., and Manning, C. D. (2017). Get To The Point : Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1073–1083.

Tan, J., Wan, X., and Xiao, J. (2017). Abstractive Document Summarization with a Graph-Based Attentional Neural Model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1171–1181.

Zopf, M., Loza Mencía, E., and Fürnkranz, J. (2016a). Beyond Centrality and Structural Features: Learning Information Importance for Text Summarization. In *Proceedings of the 20th Conference on Computational Natural Language Learning*, pages 84–94. Association for Computational Linguistics, aug.

Zopf, M., Peyrard, M., and Eckle-Kohler, J. (2016b). The Next Step for Multi-Document Summarization : A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 1535–1545.

PyrEval: An Automated Method for Summary Content Analysis

Yanjun Gao, Andrew Warner and Rebecca J. Passonneau
The Pennsylvania State University
Pennsylvania, USA
yug125@cse.psu.edu, acw5456@psu.edu, rjp49@cse.psu.edu

Abstract

The pyramid method is a content analysis approach in automatic summarization evaluation for manual construction of a content model from reference summaries, and manual scoring of unseen summaries with the pyramid model. PyrEval automates the manual pyramid method. PyrEval uses low-dimension distributional semantics to represent phrase meanings, and a new algorithm, EDUA (Emergent Discovery of Units of Attraction), to solve a set cover problem to construct the content model from vectorized phrases. Because the vectors are pretrained, and EDUA is an efficient greedy algorithm, PyrEval can apply pyramid content evaluation with no retraining, and in excellent time. Moreover, PyrEval has been tested on many datasets derived from humans and machine generated summaries, and shown good performance on both.

Keywords: Content Evaluation, Summarization, Summarization Evaluation Tool

1. Introduction

Automatic summarization methods that produce a paragraph to express the main ideas of one or more texts have shifted in recent years. The long-standing prevalence of extractive summarizers, which select complete sentences from source documents, has begun to give way to abstractive summarizers, which rewrite or generate sentences to eliminate less important content (Bing et al., 2015; Rush et al., 2015; Liu et al., 2015; Durrett et al., 2016; Nema et al., 2017). Automatic content evaluation of summarization systems has not seen the same progress. ROUGE has long been the dominant method for evaluating summary content, and is used in most of the papers cited above. Here we present PyrEval, which applies pyramid content analysis (Nenkova and Passonneau, 2004), a method intended for evaluation of abstractive summarization. The main innovation in PyrEval is a novel formulation of the pyramid construction problem, and a greedy algorithm whose approximate solution is both efficient and effective.

The pyramid method, introduced in (Nenkova and Passonneau, 2004), groups the distinct ideas (content units) mentioned in reference summaries to create a pyramid model of content, and differentiates the content units by importance as well as meaning. It then evaluates target summaries against the pyramid content model. Early work applied manual annotation of the pyramid method for largely extractive automated summarizers at a series of Document Understanding Conferences (DUC) and Text Analysis Conferences (TAC) administered by the National Institute of Standards (NIST) from 2006 through 2011 (Passonneau et al., 2006; Varma et al., 2009; Ji et al., 2010). Figure 1 shows a content unit from one of the pyramids created for the NIST pyramid evaluation at DUC 2006, along with a sentence in a summary from a peer (automated summarizer) that was matched to the content unit (CU). The illustrated content unit, labeled by the annotator, lists a phrase from each of three reference summaries (D, E, H) that express the labeled meaning. The importance weight of the CU is the number of contributing reference summaries.

Automated pyramid methods such as PyrScore (Passonneau et al., 2013) or PEAK (Yang et al., 2016) have been used in recent work on abstractive summarization, as in (Bing et al., 2015) (PyrScore) and (Peyrard and Eckle-Kohler, 2017) (PEAK), but have disadvantages that PyrEval addresses. PyrScore requires manual construction of a pyramid content model, and PEAK depends on external resources that are computationally inefficient for large-scale use. This paper introduces the PyrEval software and gives an overview of how PyrEval builds on the earlier PyrScore. [1] Details of each component will be presented in other publications. Tests of PyrEval performance on five years of manual pyramid evaluations conducted by NIST show that average performance on all topics for a given year correlates well with manual pyramid.

Example Content Unit

Summary	**Label**: Black boxes record vehicle data (**Wt** = 3)
D1	"black boxes" that record data in the last seconds before a crash
E3	record data on vehicle performance ("black box")
H4	black boxes . . . record data pertinent to an auto crash to lead to safety improvements

Matched phrase from a peer summary

Peer	a device similar to an airliner's black box, . . . that automatically records the vehicle's

Figure 1: A content unit (CU) of weight 3 from a NIST manual pyramid for topic D0608, and a phrase from an extractive summary (peer) that was matched to the CU.

2. Related Work

The most widely used content evaluation tool for machine-generated summaries is ROUGE (Lin, 2004), which takes model summaries as references and scores target summaries by matching substrings. It correlates well with human scores in ranking performance of systems on multiple summarization tasks. But it is not reliable for evaluating a

[1] Availablefordownloadfromhttps://github.com/serenayj/PyrEval

single summary (Louis and Nenkova, 2009). In addition, it does not provide informative feedback on the ideas contained in a summary, or the important ideas that have not been mentioned. In contrast, pyramid content scores are reliable for individual summaries, the scores can be interpreted as indicating whether a summary contains mainly important ideas, and whether it contains enough of the important ideas. Finally, the scores can be justified with respect to specific ideas that are included or missing.

Interest in the pyramid method has been revived by development of PEAK (Yang et al., 2016), a fully automated version that performs well on student summaries. (Peyrard and Eckle-Kohler, 2017) incorporate PEAK scores into an optimization objective for generating summaries.

PEAK is one of two recent automated systems that implement pyramid evaluation (Yang et al., 2016). It constructs a pyramid from reference summaries, and scores target summaries against its automated pyramid. It extracts subject-predicate-object triples from reference summary sentences using ClauseIE (Del Corro and Gemulla, 2013), then assembles a hypergraph where each triple is a hyperedge containing three nodes: $subject$, $predicate$ and $object$. The semantic similarity of all pairs of nodes in distinct hyperedges is measured using Align, Disambiguate and Walk (ADW) (Pilehvar et al., 2013), which relies on WordNet (Miller, 1995). Triples from distinct summaries with high similarity are combined into content units. To assess a new summary, its triples are extracted, and the Munkres-Kuhn algorithm (Kuhn, 1955) is applied to a bipartite graph from content unit (CU) nodes u to nodes v that represent target summary triples.

PyrScore (Passonneau et al., 2013) uses manual pyramids to score target summaries, with Weighted Textual Matrix Factorization (WTMF) for latent semantic representation (Guo and Diab, 2012), and ngram segmentation to find all possible combinations of covering segmentations of input sentences. The scoring procedure can be formulated as a graph covering problem, where each sentence is a subgraph, and each segmentation assigned to a sentence is a vertex of the graph. The goal is to align a candidate sentence segmentation with content units from a manual pyramid via semantic similarity between each segment and every contributor in a content unit, under the constraint that a content unit can be allocated to no more than one segment. PyrScore applies WMIN (Sakai et al., 2003), a greedy algorithm, to solve this weighted set covering problem.

Both PyrScore and PEAK correlate well with manual pyramid scores, yet both methods have drawbacks. PyrScore does not construct a pyramid, so is not fully automated. ADW does random walks over WordNet, and ClauseIE generates a large number of low-quality triples; these steps result in high computation time. The PEAK pyramids are less interpretable because they contain many variants of the same CUs. PyrEval applies the same latent semantic methods used in PyrScore, and includes all the functionality of PEAK. It is faster than PEAK, and includes additional functions. To construct a pyramid, PyrEval relies on a new algorithm we developed (detailed in a separate publication), to produce a pyramid with a single version of each CU. It incorporates an improved version of PyrScore. Here we report on PyrEval's performance and present it for public use.

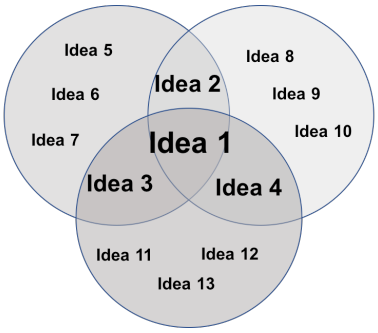


Figure 2: Venn diagram of summary content overlap

Segment from an automated summary:
Australia Sunday sent three Air Force C Hercules aircraft loaded with medical and food supplies on an urgent mission to help survivors of a devastating tsunami .

Matched content unit (Wt. = 4) from pyramid:
SUM1 Three Australian Cs were bringing food , medical supplies and personnel and a mobile hospital .
SUM2 Australia will transport relief supplies and provide a mobile hospital and medical personnel .
SUM3 Australia is sending medical supplies and food and is expecting to set up a mobile hospital .
SUM4 Australia also was sending a mobile hospital and doctors .

Figure 3: Example of a segment from D1004A automated summary and the matched content unit from D1004A PyrEval pyramid.

3. Pyramid Content Evaluation

The pyramid method was developed to capture the observation that high quality summaries from different individuals of the same source texts will express overlapping content, but will have even more content that is unique to each summary. This makes it challenging to define the gold standard content. Figure 2 illustrates the typical case by means of a Venn diagram to show that, given n reference summaries, among the set of all distinct ideas from the n summaries, relatively few ideas will occur in all n, and for each decreasing cardinality from n to 1, there will be a larger number of such ideas. Here one idea occurs in three summaries, three occur in two, and nine occur in only one. The font size of the schematically represented ideas in the figure reflects their importance, with the highest importance going to ideas that are expressed in all the reference summaries. Importance represents which ideas should be included in a new summary to adhere to the distribution of ideas found in the reference summaries.

The pyramid method defines content units (CUs) to correspond largely to clauses, and asks annotators to select all the phrases from distinct summaries that express the same content as a single CU (cf. Figure 1). The DUCView annotation tool developed for pyramid annotation elicits a label that the annotator writes. The CU consists of the label, and clauses (or sometimes phrases) selected from different reference summaries (e.g., D1, E3 and H4) that express the

Step	Description	Size	Runtime (sec)
1	Preprocessing with Stanford CoreNLP	26 summaries (312 sent; 6K words)	73.40
2	Sentence Decomposing and WTMF	same as above	208.68
3	Building Pyramid from Model Summaries	4 (human) summaries	4.35
4	Scoring Target Summaries by Pyramid	22 (machine) summaries	14.17

Table 1: Steps and times for PyrEval on one DUC topic(D0608).

same content. The weight is automatically assigned as the number of phrases. Each summary can contribute no more than once to the same content unit. The instructions require annotators to select, as far as possible, all the text from the reference summaries. The final pyramid consists of all the CUs identified by the annotator, with weights from 1 to N. To annotate a target summary against a pyramid model, the same annotation tool can be used to match phrases in the target summary to CUs in the pyramid. Figure 3 shows an example of scoring a segment from a target summary with the pyramid built by PyrEval. The sum of the weights of the matched CUs from the target summary is normalized in different ways to produce a score. Previous work has found this method to produce very stable and reliable scores (Nenkova et al., 2007), and to have very good inter-annotator agreement (Passonneau, 2010). Despite the labor intensiveness of manual annotation, pyramid has been applied in much work on summarization. We developed PyrEval to facilitate more widespread usage.

4. PyrEval

4.1. System Overview

Like PEAK, PyrEval implements pyramid construction and automated scoring. In contrast to PEAK, it can also score target summaries against a manually annotated pyramid that has been produced with the DUCView annotation tool. DUCView was developed for pyramid annotation, and has been used for all the NIST manual pyramid evaluations.

Two initial procedures for processing input texts are: Step 1-segmentation of sentences into clause-like units by

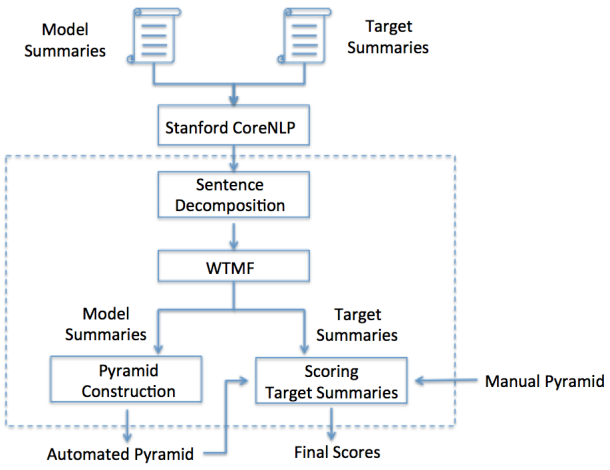


Figure 4: System Flow of PyrEval

The death toll, mostly children and old people, has reached 9 but is expected to rise and there are thousands of injured and homeless, with no food or water.

```
(ROOT (S
 (S
  (NP (NP (DT The) (NN death) (NN toll-3)) (, ,) (NP (NP (RB mostly)
  (NNS children)) (CC and) (NP (JJ old) (NNS people))) (, ,)) (VP (VP
  (VBZ has ) (VP (VBN reached-12)))
  (CC but) (VP (VBZ is) (VP (VBN expected -15) (S (VP (TO to) (VP (VB
  rise))))))))
  (CC and) (S
   (NP (EX there-19)) (VP (VBP are-20) (NP (NP (NNS thousands-21))
   (PP (IN of) (NP (JJ injured) (CC and) (JJ homeless)))) (, ,) (PP (IN
   with) (NP (DT no) (NN food) (CC or) (NN water)))))
 (. .)))
```

Figure 5: A sentence from D1004A and its constituency parse output from Stanford CoreNLP. The subtrees dominated by the nodes marked with bold font are the structures extracted by the decomposition parser.

PyrEval's sentence decomposition parser; Step 2-latent semantic representation via WTMF (see first two interior boxes in Figure 4). Then pairwise cosine similarities are computed between clause-like units from different model summaries, or clause-like units from target summaries and model CUs. Step 3 is Pyramid Construction, the second new component of PyrEval. Step 4 is PyrEval scoring; it is similar to PyrScore, but uses clause-like units rather than ngrams. Table 1 shows an example of running times for these four steps on one of the development sets from DUC06.

PyrEval was developed and tested on English. To apply it with other languages would require replacing the preprocessing steps that segment sentences into distinct clauses, and that produce low-dimensional vectors for their semantic representation. In principle, the current code package could easily be extended to substitute a sentence segmenter designed for a distinct language, and a pretrained matrix of words by sentences.

The installation for PyrEval requires Stanford CoreNLP (Manning et al., 2014), perl, python2.7 (or anaconda) with packages NLTK, statistics, beautifulsoup, networkx, sklearn and numpy. All the experiments reported here were conducted on an ubuntu machine, with 4 Intel i5-6600 CPUs.

4.1.1. Sentence Decomposition

PyrEval relies on our sentence decomposition parser to extract clause-like units (segments) from sentences. Each set of semantic units yielded by a sentence is a *segmentation*, and there will be at least one segmentation per sentence. The sentence decomposition parser takes results from the Stanford CoreNLP constituency parser and dependency parser as input. For each sentence, the decomposi-

root (ROOT-0 , reached-12), det (toll-3 , The-1) , compound (toll-3 , death-2), **nsubj (reached-12 , toll-3), nsubjpass (expected-15 , toll-3)** (extra), advmod (children-6 , mostly-5), appos (toll-3 , children-6), cc (children-6 , and-7), amod (people-9 , old-8), appos (toll-3 , people-9) (extra), conj:and (children-6 , people-9), aux (reached-12 , has-11), cc (reached-12 , but-13), auxpass (expected-15 , is-14), conj:but (reached-12 , expected-15), mark (rise-17 , to-16), xcomp (expected-15 , rise-17), cc (reached-12 , and-18), **expl (are-20 , there-19)**,conj:but (reached-12 , are-20), **nsubj (are-20 , thousands-21)**, case (injured-23 , of-22), nmod:of (thousands-21 , injured-23), cc (injured-23 , and-24), nmod:of (thousands-21 , homeless-25) (extra), conj:and (injured-23 , homeless-25), case (food-29 , with-27), neg (food-29 , no-28),nmod:with (are-20 , food-29),cc (food-29 , or-30), nmod:with (are-20 , water-31) (extra), conj:or (food-29 , water-31)

Figure 6: Enhanced dependency parser output from Stanford CoreNLP for the sentence in Figure 5.The bold fonts are the dependency relations extracted by our decomposition parser.

Sentence Decomposition Example

	Segmentation 1
Segment 1	The death toll, mostly children and old people has reached 9 but is expected to rise
Segment 2	and there are thousands of injured and homeless with no food or water .
	Segmentation 2
Segment 1	toll The death, mostly children and old people has reached 9 but
Segment 2	toll is expected to rise and there are thousands of of injured and homeless, with no food or water.

Figure 7: Sentence decomposition parser output for sentence shown in Figure 5

tion parser extracts verb phrases and clauses from the constituency parses, and relation arcs from dependency parses, then associates verb phrases with subjects. Unconsumed words are attached to segments based on their linear position relative to words in segments. Each segmentation contains one or more segments (see Figure 7), and covers all words in a sentence. Every segment is then represented as a WMTF vector.

4.1.2. Pyramid Construction

In this paper, we give a high level description of the algorithm for constructing the pyramid. Each candidate content unit is represented as an undirected, complete graph G where vertices are vectors of segments from different summaries, and edges are cosine similarities (See Figure 3). To construct CU graphs, pairwise similarities between segments from different summaries are computed, and edges are added if every pair of vertices v_i, v_j has a cosine similarity above a fixed threshold t. Semantic coherence of a CU graph is measured as the Average Similarity (AS) of all segment pairs. Given G with n vertices, we define k as the number of edges and calculate AS as:

$$AS = \frac{\sum_{u,v \in G, u \neq v} similarity(u, v)}{k} \quad (1)$$

CU weight produces a partition over all CUs; that is, all CUs of the same weight are one equivalence class. The sizes of the n equivalence classes for n reference summaries are observed to have a power law distribution inversely related to the weights (Nenkova et al., 2007); that is, the smallest class is the one with the highest weight. We developed a greedy algorithm to allocate segments to CUs that achieves the maximum average of the AS values for each equivalence class, and that adheres to a power law distribution.

4.1.3. Automated Scoring

PyrEval can score target summaries using a manually or automatically constructed pyramid. As with pyramid construction, sentences in target summaries are processed by the Stanford CoreNLP tools and sentence decomposition parser to produce candidate segmentations for each sentence, then WTMF is used to generate their vector representations. The scoring function is the one used in PyrScore. The quantitative output includes a raw score (the sum of matched CU weights), and several ways to normalize. The user can choose to output a qualitative analysis as well.

5. Experiments

We first tested PyrEval on single-document summaries from community college students that PEAK and PyrScore were tested on (Passonneau et al., 2017). PyrEval has four parameters: two that control the size of the equivalence classes corresponding to CUs of each weight (a, b), and two similarity thresholds (for CU construction, t_1; for scoring, t_2). In each set of reference summaries, cosine similarities are computed for all pairs of segments from different summaries. Values are distributed differently for different sets of reference summaries, so we define t_1 as a percentile over the range of observed cosine values for a given set of reference summaries, and conduct grid search over a range of percentile values, as detailed below. For t_2 we use 0.50 cosine similarity, as in previous work (Passonneau et al., 2017). The Pearson correlation with manual pyramid scores on 20 student summaries reached 89%, a 10% improvement over the previous results.

When we turned to DUC and TAC datasets (see above), consisting of extractive summarizers on newswire with model summaries written by NIST assessors, new challenges arose. The decomposition parser for all results presented here was extended to handle the more complex syntax found in newswire, relative to the student summaries.

We performed grid search for DUC06 dev consisting of four of the twenty topics, and selected the best four sets of parameters. For each next year, we tested these same four parameter sets on a randomly selected dev set of size four, and found the best performance for $a = 100$, $b = 2.5$, $t_1 = 83\%$; these values yielded the smallest standard deviations for average correlations with the manual pyramids. The value of the a parameter is apparently sensitive to the number of reference summaries. We tested PyrEval on the same student data reported in (Passonneau et al., 2017), which has five model summaries instead of the four that come with the NIST datasets. When we did grid search on a development set of the student data, we got better performance for $a \in [125, 250]$.

Table 2 presents results on five years of DUC/TAC. In each year, testing was performed on all data apart from the devel-

Dataset	Size	Avg	Med	Min	Max	StdDev
DUC06	16	41.47	46.62	-2.31	76.34	24.42
DUC07	19	44.66	39.63	-3.27	89.18	27.05
TAC08	96	42.26	43.29	-13.93	79.09	18.92
TAC09	88	55.14	57.36	15.64	84.54	16.67
TAC10	92	51.95	53.33	10.16	83.88	16.27

Table 2: Pearson correlations (Pearson's $r \times 100$) with manual pyramid scores from DUC and TAC.

opment set. Because the semantic vectors are constructed from pretrained data, there is no training set. As shown, the trend over time is for PyrEval's average correlation to be higher, and for the standard deviation to be lower. This could result simply from the much larger datasets as of 2008, or that PyrEval performs better on data from summarizers whose performance is better. Conroy & Dang (2008) report that the top ROUGE scores increase from about 0.27 in 2005 to about 0.34 in 2007. The maximum correlations for an individual topic in one year can be as high as 0.89. The minimum values tend to be many standard deviations below the mean, suggesting that these are outliers, and that the median is a better summary statistic. Apart from DUC07, the medians are higher than the means.

6. Discussion and Conclusion

The results in Table 2 show that PyrEval can achieve good average correlations of up to 0.55 (TAC10), and much higher (e.g., 0.89) on individual summarization tasks. As discussed in the next paragraph, we have identified a number of needed improvements we plan to address.

Inspection of the pyramid for one of the DUC06 dev topics (D0608) that had lower performance illustrates one of the two issues our future work will address, both of which are classic problems in NLP: noun-noun compounds (including Named Entities) and conjunction. As shown in CU 1 of Figure 8, there are noun-noun compounds and NEs in all four segments (e.g., "booster seat", "back seat", "crash test", "sport utility model"; "MercedesBenz", "BodySmart"). The vector representations treat the compounds as sequences of individual words, which does not capture the correct semantics. Identification of multi-word expressions prior to generating the phrase vectors might correct this. We have also observed that conjunctions, which typically introduce a great deal of syntactic ambiguity, are often parsed incorrectly. This can lead to incoherent output from the decomposition parser. Possible methods to address this include neural net dependency parsing, as in (Ficler and Goldberg, 2017), or clause-chart parsing (Kríž and Hladká, 2016).

Figure 8 also shows a CU produced by PyrEval that captures the same content as the manual content unit shown in Figure 1. The PyrEval CU has no label, but is otherwise similar to the manual CU: it has the same weight, and the contributing clauses come from the same sentences of the same summaries. The phrases selected by the human annotator for this CU are subphrases of the clauses that PyrEval groups together. Future work could rely on improved parsing to identify all distinct propositions, not just tensed clauses.

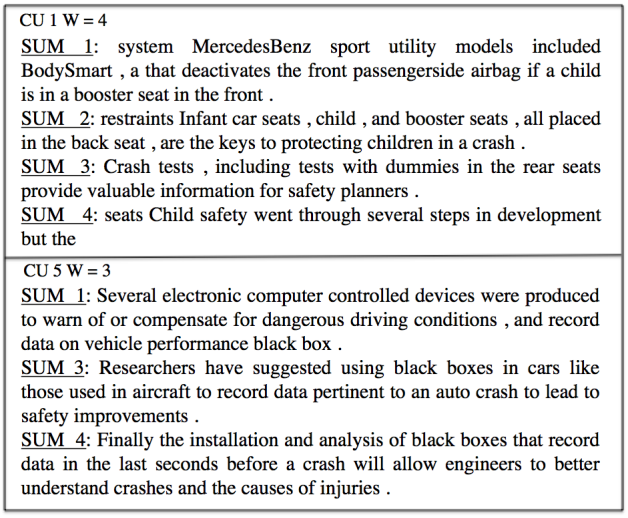


Figure 8: Examples of an incoherent CU (CU 1) and a coherent CU (CU 5) from DUC0608.

In sum, PyrEval is a fast, complete, multi-use tool. In tests on summaries from extractive, multi-document summarizers, it achieves good average correlations with manual pyramid evaluations on five years of DUC/TAC .

7. References

Bing, L., Li, P., Liao, Y., Lam, W., Guo, W., and Passonneau, R. (2015). Abstractive multi-document summarization via phrase selection and merging. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1587–1597, Beijing.

Conroy, J. M. and Dang, H. T. (2008). Mind the gap: Dangers of divorcing evaluations of summary content from linguistic quality. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1 (COLING '08)*, pages 145–152, Stroudsburg, PA, USA. Association for Computational Linguistics.

Del Corro, L. and Gemulla, R. (2013). ClausIE: clause-based open information extraction. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 355–366. ACM.

Durrett, G., Berg-Kirkpatrick, T., and Klein, D. (2016). Learning-based single-document summarization with compression and anaphoricity constraints. In *Proceedings of the 54th Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 1998–2008, Berlin.

Ficler, J. and Goldberg, Y. (2017). Improving a strong neural parser with conjunction-specific features. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL): Volume 2, Short Papers*, pages 343–348, Valencia, Spain, April. Association for Computational Linguistics.

Guo, W. and Diab, M. (2012). Modeling sentences in the latent space. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 864–872.

Ji, H., Grishman, R., Dang, H. T., Griffitt, K., and Ellis, J. (2010). Overview of the TAC 2010 knowledge base population track. In *Third Text Analysis Conference (TAC 2010)*, volume 3, pages 3–28.

Kríž, V. and Hladká, B. (2016). Improving dependency parsing using sentence clause charts. In *Proceedings of the ACL 2016 Student Research Workshop*, pages 86–92.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, 2(1-2):83–97.

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In Stan Szpakowicz Marie-Francine Moens, editor, *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8, pages 74–81. Barcelona, Spain.

Liu, F., Flanigan, J., Thomson, S., Sadeh, N., and Smith, N. A. (2015). Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086.

Louis, A. and Nenkova, A. (2009). Automatically evaluating content selection in summarization without human models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 306–314. Association for Computational Linguistics.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Nema, P., Khapra, M. M., Laha, A., and Ravindran, B. (2017). Diversity driven attention model for query-based abstractive summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1063–1072, Vancouver, Canada, July.

Nenkova, A. and Passonneau, R. (2004). Evaluating content selection in summarization: The pyramid method. In Susan Dumais, et al., editors, *Proceedings of Human Language Technologies and the North American Association of Computational Linguistics (HLT-NAACL 2004)*, pages 145–152, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Nenkova, A., Passonneau, R., and McKeown, K. (2007). The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(2):1–23. Article 4.

Passonneau, R. J., McKeown, K., Sigelman, S., and Goodkind, A. (2006). Applying the pyramid method in the 2006 document understanding conference. In *Proceedings of the Document Understanding Conference (DUC 06)*.

Passonneau, R. J., Chen, E., Guo, W., and Perin, D. (2013). Automated pyramid scoring of summaries using distributional semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 143–147.

Passonneau, R. J., Poddar, A., Gite, G., Krivokapic, A., Yang, Q., and Perin, D. (2017). Wise crowd content assessment and educational rubrics. *International Journal of Artificial Intelligence in Education*, pages 1–27.

Passonneau, R. J. (2010). Formal and functional assessment of the pyramid method for summary content evaluation. *Natural Language Engineering*, 16:107–131.

Peyrard, M. and Eckle-Kohler, J. (2017). Supervised learning of automatic pyramid for optimization-based multi-document summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1084–1094.

Pilehvar, M. T., Jurgens, D., and Navigli, R. (2013). Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1341–1351.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, September.

Sakai, S., Togasaki, M., and Yamazaki, K. (2003). A note on greedy algorithms for the maximum weighted independent set problem. *Discrete Applied Mathematics*, 126(2):313–322.

Varma, V., Pingali, P., Katragadda, R., Krishna, S., Ganesh, S., Sarvabhotla, K., Garapati, H., Gopisetty, H., Reddy, V. B., Reddy, K., et al. (2009). IIIT hyderabad at TAC 2009. In *Second Text Analysis Conference (TAC 2009)*.

Yang, Q., Passonneau, R. J., and de Melo, G. (2016). PEAK: Pyramid evaluation via automated knowledge extraction. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2673–2680.

Mapping Text to Scripts: An Entailment Study

Simon Ostermann, Hannah Seitz, Stefan Thater, Manfred Pinkal
Saarland University
Saarbrücken, Germany
{simono|hseitz|stth|pinkal}@coli.uni-saarland.de

Abstract

Commonsense knowledge as provided by scripts is crucially relevant for text understanding systems, providing a basis for commonsense inference. This paper considers a relevant subtask of script-based text understanding, the task of mapping event mentions in a text to script events. We focus on script representations where events are associated with paraphrase sets, i.e. sets of crowdsourced event descriptions. We provide a detailed annotation of event mention/description pairs with textual entailment types. We demonstrate that representing events in terms of paraphrase sets can massively improve the performance of text-to-script mapping systems. However, for a residual substantial fraction of cases, deeper inference is still required.

Keywords: script knowledge, annotation study, textual entailment

1. Introduction and Motivation

Scripts represent knowledge about everyday activities, or *scenarios*, like "going to the movies" or "having dinner at a restaurant" (Schank and Abelson, 1975). They consist of *events* like ORDER MENU or EAT that take place in the scenario, plus information about the typical temporal order in which these events happen, as well as knowledge about *participants* that play a role in the script, such as the waiter or plates and cutlery. Figure 1 shows a graphical representation of the script of the BAKING A CAKE scenario. The nodes in the graph represent events and the edges their temporal order. Relevant script participants, such as CAKE, KITCHEN or INGREDIENTS, are listed in the lower right corner. The dashed boxes represent possible linguistic realizations of the individual events in terms of *paraphrase sets*, described in more detail below.

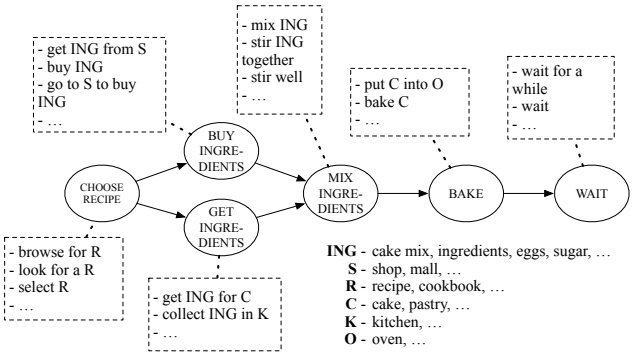


Figure 1: An example for a temporal script graph (BAKING A CAKE).

In communication, script knowledge is assumed to be part of the common ground, and people often do not mention events which can easily be inferred to have happened by the addressee. For instance, if someone tells about the last time they baked a cake, it is likely that they do not mention the fact that the cake was put into the oven, because it is obvious that this event took place. In contrast, a text understanding system that does not have access to script knowledge will probably not be able to draw this inference.

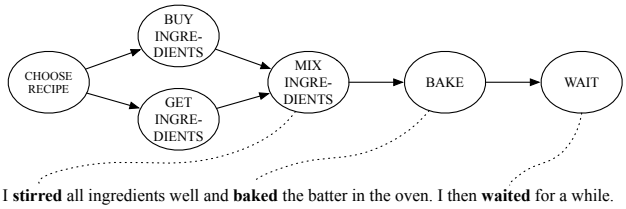


Figure 2: An example for text-to-script mapping in the BAKING A CAKE scenario

Script knowledge has been shown to be useful for a variety of tasks that are required for text understanding, such as event prediction (Chambers and Jurafsky, 2009; Pichotta and Mooney, 2014; Modi et al., 2016), event ordering (Modi and Titov, 2014), paraphrasing (Wanzare et al., 2017) or discourse referent prediction (Modi et al., 2017).

One important aspect in which models that make use of script knowledge differ is the representation of events. Chambers and Jurafsky (2009), for instance, use a "shallow" surface-oriented representation of events which is based on the verb of an event-denoting clause. A different approach has been proposed by Regneri et al. (2010) (henceforth, RKP), who adopt a richer representation of events in terms of *paraphrase sets*, as depicted in Figure 1. Script knowledge of this kind is acquired by first crowdsourcing alternative descriptions of an activity type in terms of sequences of short, telegram-style natural-language *event descriptions* (ED). Then, paraphrase sets are induced automatically as clusters of EDs, using multiple sequence alignment (Regneri et al., 2010) or semi-supervised clustering (Wanzare et al., 2016).

In order to tap the potential of script knowledge in text understanding, systems must be able to link event mentions in texts to the corresponding event types of a script, as indicated by the dotted lines in Figure 2. To our knowledge, Ostermann et al. (2017) is the only existing work on this *text-to-script mapping* task. Their approach is based on RKP-style script representations. Using this representation, the identification of the correct event type of an event mention is in many cases reduced to a simple identity check

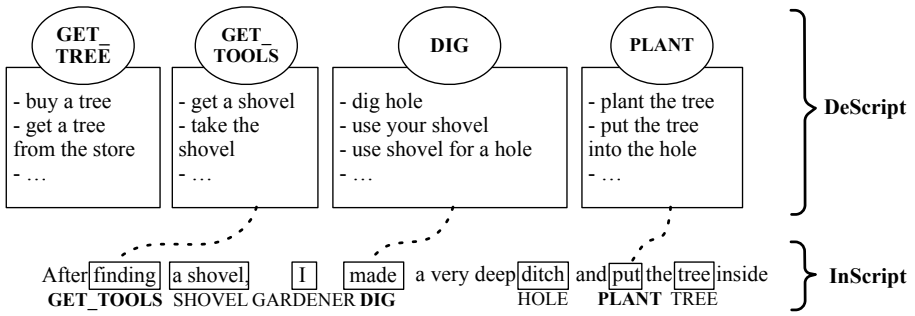


Figure 3: An example annotation in InScript (lower part), and the corresponding paraphrase sets from DeScript (upper part), for the PLANTING A TREE scenario.

between the event mention and one of the EDs in the corresponding paraphrase set. The model achieves promising results when compared to a non-trivial baseline (0.55 F_1 over a baseline of 0.40 F_1), but one would expect even better performance given that the crowdsourced paraphrase sets should substantially facilitate the mapping task.

In this paper, we investigate which types of knowledge and inference are required to assign correct event types to event mentions in text, and to which degree crowdsourced script representations help to facilitate text-to-script mapping. We do so by determining the type of semantic relations between pairs of an event mention and the paraphrase set representing the corresponding event type. In Figure 2, for example, we determine the type of semantic relation that needs to be modeled in order to align the phrase *stir all ingredients* with the MIX_INGREDIENTS paraphrase set.

The contributions of this work are as follows:

- We provide a manual annotation of word-level entailment types on verbs/events and nouns/participants (e.g. *Synonymy*, *Hypernymy*, *Inference* etc., Section 3.) based on an existing corpus of narrative texts and an existing collection of script data. In a second step, we compositionally derive clause-level entailment types (e.g. *Equality*, *Entailment*, etc., Section 4.) that illustrate the types of inference that need to be modeled for the alignment of the clause with an event paraphrase set.
- We provide a detailed analysis of our annotation and show that (1) paraphrase sets as a constitutive part of script representations massively reduce the difficulty of automatic text-to-script mapping and increase its accuracy. We also find that (2) a substantial sub-class of cases cannot be handled using just identity checks or shallow semantic modeling, but require deeper inference methods (Section 5.).
- We demonstrate the usefulness of our dataset as a testbed and diagnostic tool for the differentiated assessment of text-to-script mapping systems, by applying it to the system of Ostermann et al. (2017) (Section 6.).

2. Data

For our study, we use two existing resources that form the basis for evaluating text-to-script mapping: *InScript* (Modi et al., 2016), a collection of narrative texts centered around 10 script scenarios, and *DeScript* (Wanzare et al., 2016), a resource of structured script knowledge in the form of paraphrase sets, covering the same 10 scenarios.

InScript contains 910 stories in total. Verbs and nouns in InScript are manually annotated with participant and event type labels, respectively (see the lower part of Figure 3). The labels are based on manually created, scenario-specific templates, which list all central event and participant types, such as GET_TOOLS or DIG (events) and SHOVEL or GARDENER (participants) for the PLANTING A TREE scenario.

For target script representations, we use DeScript. It contains crowdsourced event sequence descriptions for 40 different everyday scenarios, including the 10 InScript scenarios, as well as manually created paraphrase sets for the latter, which are labeled with the same event types as used in InScript, as can be seen in the upper part of Figure 3: Each event that is labeled in InScript has a corresponding paraphrase set in DeScript. This provides a gold standard alignment between textual event mentions and paraphrase sets, as indicated by the dotted lines, which is the basis for our annotation. The paraphrase sets in this kind of representation contain not only lexical synonyms, but also scenario-specific paraphrases of the same event: *use your shovel* and *dig hole* are not synonyms in a narrow sense, but in the context of PLANTING A TREE, they both describe the DIG event.

While DeScript provides manually created gold event paraphrase sets, the corresponding information on participant level is missing. Since we need entailment relations on the noun phrase/participant level for our compositional derivation of clause-level entailment (see Section 4.), we extended DeScript with paraphrase sets for participant descriptions (as depicted in the lower right corner of Figure 1). For this purpose, we annotated all nouns in the 10 scenario subset of DeScript with labels from the InScript inventory of participant types. Data from the BUS and TREE scenarios were annotated by two annotators to assess the inter-annotator agreement, which was *almost perfect* (Landis and Koch (1977), $\kappa = 0.91$).

From this annotation, paraphrase sets for participants were derived, i.e. the sets of all nouns describing the same participant. In the BAKING A CAKE scenario for example, all different ingredients such as eggs, flour, milk etc. are members of the INGREDIENT paraphrase set.

For our study, we selected 3 out of the 10 scenarios that differ with respect to their complexity: TAKING THE BUS, BAKING A CAKE and PLANTING A TREE. We annotated

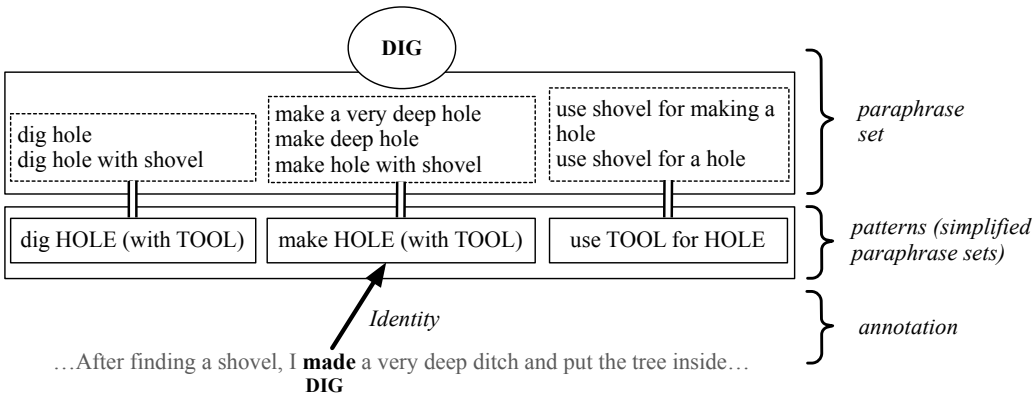


Figure 4: Combining the EDs in the DIG event into patterns (upper part) and the actual verb annotation (lower part).

all script-relevant verb instances and all relevant participant instances in every story.

3. Lexical Entailment: Annotation Study

To identify the type of semantic relation that needs to be modeled in order to align an event-denoting clause in the text with a paraphrase set representing the same event, the most straightforward way would be to conduct a clause-level entailment annotation between the clause and EDs in the paraphrase set, e.g. with a set of clausal entailment types similar to the ones used in (MacCartney and Manning, 2009).

We found, however, that the assessment of entailment types is time-consuming and unreliable, when based on a direct comparison of complex text clauses and paraphrase sets. We therefore simplify the task, breaking it down into two steps: First, annotators were asked to assign semantic relations between the event descriptions in the paraphrase sets and their instantiations in narrative texts *on the lexical level only*, labeling the event-denoting verbs and the participant-denoting noun phrases, as is described in this section. Second, we automatically derive an approximate clause-level entailment type from the lexical-level labels in a quasi-compositional way from the manually annotated lexical entailment relations, as addressed in Section 4.

In the following subsections, we describe the lexical entailment annotation we conducted on the DeScript and InScript data for events/verbs (Section 3.1.) and participants/nouns (Section 3.2.).

3.1. Events

To prepare the data for the event annotation, we made use of the gold alignment (cf. Figure 3): Each event-denoting verb in InScript was presented with the corresponding paraphrase set in DeScript.

Comparing *every* ED in the paraphrase set to the clause in the text is cumbersome due to the number of EDs (up to 76) in each paraphrase set. We therefore simplified the event paraphrase sets by building equivalence classes of EDs that use the same head verb, which we called *event patterns*. They are derived semi-automatically by summarizing EDs that contain the same main verb, and replacing noun phrases with their participant type label (upper part of Figure 4). Also, participants that do not appear in every ED are put in brackets to mark them as optional.

In the annotation process, annotators were only shown the patterns instead of the full paraphrase set. In the lower part of Figure 4, the verb labeled as DIG is compared to the event patterns extracted from the DIG paraphrase set. Verbs were presented in their sentential context and highlighted.

Instead of annotating each pattern, the guidelines required annotators to select only the most similar pattern for the event-denoting verb, and to do the annotation only for this pattern. While this selection results in a non-exhaustive annotation, no important information is lost: The procedure of selecting the most similar pattern retains the minimal inference steps required for the alignment.

After selecting a pattern, annotators were instructed to assess the relation between the verb in DeScript and the verb in InScript and use the context only for lexical disambiguation, i.e. not to assess clause-level entailment. In our annotation schema, we include the following labels:

- *Identity, Synonymy, Hyponymy, Hypernymy*. Defined as in *WordNet* (Fellbaum, 1998). *Hyponymy* describes the case in which the verb in the text is more specific than in the pattern, *Hypernymy* the opposite.

- *Incorporation*. One verb includes a participant, which is explicitly mentioned with the other verb.
  Example: I ***sprinkled flour*** *in the pan.* – ***flour*** CAKE_TIN

- *Diathesis*. This covers active/passive alternation (Ex. 1) and verbs that are conceptually equivalent but have different syntactic realizations. In *FrameNet* (Ruppenhofer et al., 2006), these verbs would typically be associated with the same frame (Ex. 2).
  Example 1: *The cake* ***was cut****.* – ***cut*** CAKE
  Example 2: *The cake* ***went*** *in the preheated oven* – ***put*** *(*CAKE*) in* OVEN

- *Phrasal Verb.* One of the verbs is a particle verb that has the same meaning as the other verb.
  Example: ***I went out*** *to the grocery store.* – ***go*** *to* STORE

- *Inference.* A complex inference is needed to associate the verbs with the same event.

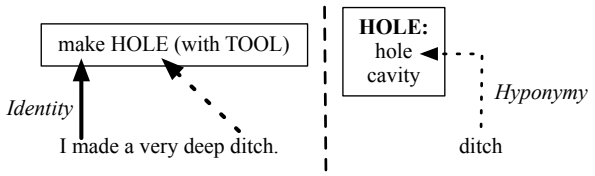


Figure 5: Aligning participants with a pattern (left), selecting a noun from the participant paraphrase set and labeling (right).

Example 1: *I **put** the decorations on the cake. - **use*** DECORATION
Example 2: *I **made sure** that I had all the ingredients that I needed. - **gather*** INGREDIENTS

- *NoMatch.* No match is possible. This is typically the case when annotation errors occur in InScript.

3.2. Participants

Associating verbs in a textual mention with the ED verb is straightforward, since there is only one head verb on each side. Linking participants in contrast requires additional annotation effort: They can be left out or appear more than once in one clause. We thus divided the participant annotation into two steps:

Participant Alignment. In order to distinguish missing from realized participants, annotators had to align all participants that are relevant to the event in question in the text with their counterparts in the pattern, i.e. with the participant type labels. In Figure 5, this step corresponds to the dotted arrow on the left side. Participants that were mentioned in the text and that play a role in the event, but that do not have a counterpart in the ED, were marked as *Additional*. Required participants in the pattern that were not aligned were automatically labeled as *Missing* afterwards.[1] As an example for this label, consider the text clause *I used a shovel*, with the matching pattern *use TOOL for HOLE*. The participant HOLE is omitted, but mentioned as mandatory in the pattern, i.e. it appears in every ED of the respective paraphrase set (*use shovel for making a **hole*** and *use shovel for **hole***). Therefore, it is marked as *Missing* in the pattern.

Lexical Entailment Annotation. To find the appropriate type of lexical entailment for the realized participants, annotators were then shown the participant paraphrase sets from DeScript for all aligned participants. Just as for event patterns, annotators had to choose the best matching, most similar noun from the set and assign a lexical entailment label.

As for verbs, the annotation schema includes the relation types, *Identity, Synonymy, Hyponymy, Hypernymy, Meronymy, Holonymy* and *Co-Hyponomy*, and the additional labels *Inference* and *NoMatch*, as defined in Section 3.1. We add the label *Instance*, which is used when the noun in InScript is a proper noun or entity mention of the type expressed by the DeScript noun (e.g. *number 77 – bus*).

The right side of Figure 5 illustrates this part of the annotation: The noun "ditch" (which was previously aligned with the pattern) is compared to all participant descriptions for HOLE, linked to the lexical description "hole" and labeled with *Hyponymy*. Figure 8 shows the fully labeled instance with participant and event annotations.

To simplify the annotation, we make the assumption that each noun has only one sense per scenario: In the PLANTING A TREE scenario, the polysemous word *stem* e.g. always describes a part of a tree. In order to reduce the annotation effort, we presented all different noun types per participant type only once, rather than every single mentioned token in its sentential context. This *on sense per scenario* assumption is similar to the *one sense per discourse* hypothesis, which is often used in word sense disambiguation models (Gale et al., 1992).

[1]The protagonist of the story, being very rarely mentioned in the EDs, was excluded from the alignment.

Entailment Type	**Labels**
Identity $=$	Identity
Equality $\equiv$	Synonymy, Phrasal Verb, Diathesis
Entailment $\sqsubset$	Hyponymy, Instance, *Additional*
Reverse ent. $\sqsupset$	Hypernymy, *Missing*
Partial Entailment ∞	Inference, Incorporation, Meronymy, Holonymy, Co-Hyponymy
Non-Entailment $\#$	NoMatch

Figure 6: Entailment types.

	$=$	$\equiv$	$\sqsubset$	$\sqsupset$	∞	$\#$
$=$	$=$	$\equiv$	$\sqsubset$	$\sqsupset$	∞	$\#$
$\equiv$	$\equiv$	$\equiv$	$\sqsubset$	$\sqsupset$	∞	$\#$
$\sqsubset$	$\sqsubset$	$\sqsubset$	$\sqsubset$	∞	∞	$\#$
$\sqsupset$	$\sqsupset$	$\sqsupset$	∞	$\sqsupset$	∞	$\#$
∞	∞	∞	∞	∞	∞	$\#$
$\#$	$\#$	$\#$	$\#$	$\#$	$\#$	$\#$

Figure 7: Combination table for lexical entailment classes.

4. Clause-Level Entailment: Composition

In the previous section, we described the lexical entailment annotation on verbs and nouns, i.e. on a sub-event level. In this section, we now explain a method for an automatic, quasi-compositional computation of clausal-level entailment types. We compose the types from the lexical-level entailment labels of the verb and all its annotated noun dependents.

Inspired by the textual inference method used in MacCartney and Manning (2007) and MacCartney and Manning (2009), we compute the type of clause-level entailment between InScript event mentions and DeScript patterns from the manually annotated word-level entailment labels. Following MacCartney, we group these labels according to their truth-conditional effects, and associate each group with one of six entailment types, shown in Figure 6. We adopt four entailment types from the schema of MacCartney and Manning (2009) and add two new types: We extend the schema with *Identity*, which is logically speaking a sub-case of *Equality*. Also, we use *Partial Entailment* to cover all cases of semantic relatedness which do not correspond to a direct entailment type; most prominent are the *Inference* cases.

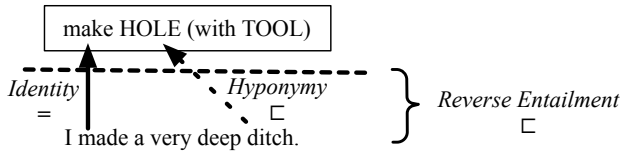


Figure 8: One fully labeled event instance, with compositionally derived clausal entailment label (*Reverse Entailment*).

Cases of *Additional* and *Missing* participants are deletions and insertions in MacCartney's terminology, and therefore have entailment and reverse entailment effects, respectively. To compute the clausal entailment type, we combine the word-level entailment labels for verbs and nouns, according to the composition table in Figure 7. The result of combining two lexical labels is in general the weaker entailment relation of the two[2]. The only exception is the pair $\{\sqsubset, \sqsupset\}$, which results in *Partial Entailment*. The application of the binary computation is commutative, so the clausal entailment type can be read off the set of lexical labels.

Figure 8 shows the running example with all lexical entailment annotations. The set of word-level entailment labels is $\{=, \sqsubset\}$, given the lexical entailment labels {*Identity, Hyponymy*}. The clausal entailment type is the weakest lexical type, i.e., $\sqsubset$ (*Reverse Entailment*).

5. Annotation Statistics

The annotation was performed on all 280 texts of the InScript corpus addressing one of the three selected scenarios. Annotation was done by two native speakers of German with a good command of English. A total of 3,427 verb mentions and 1248 noun types was annotated, respectively. We used SWAN[3] (Gühring et al., 2016) for the annotation.

5.1. Lexical Level

5.1.1. Inter-Annotator Agreement

For both verb and participant annotation, agreement is computed on two levels: First, we report how often the same pattern or head noun was selected. Second, in cases where the same pattern/noun was selected, we report the label agreement. For the verb annotation, annotators chose the same pattern in 82.4% of cases. For participants, the annotators chose the same realization from DeScript in 74.3% of cases.

On verbs, the annotators agreed on the label with $\kappa = 0.72$ (*substantial agreement*, (Landis and Koch, 1977)). On participants, they agreed with $\kappa = 0.742$ (*substantial agreement*).

Not every case of disagreement is critical: In many cases there is more than one plausible solution. In the PLANTING A TREE scenario, for example, the word *shovel* could be interpreted either as a *Synonym* or *Co-Hyponym* of *spade*. There are also no sharp boundaries between classes. In particular, annotators had difficulties with annotating *Inference* cases. Therefore, we decided not to adjudicate the annotation, but average over the distributions for evaluation.

[2]The list $=, \equiv, \{\sqsubset, \sqsupset\}, \infty, \#$ orders the labels from strongest to weakest.

[3]https://github.com/annefried/swan

Label	Events	Participants
Identity	58%	76%
Synonymy	5%	6%
Hyponymy	5%	5%
Phrasal Verb	5%	-
Inference	13%	1%
NoMatch	7%	7%
Other	7%	5%

Table 1: Distribution of lexical labels on events and participants.

5.1.2. Label Distribution

Table 1 gives label distributions for participants and events on the lexical level, averaged over both annotations. As mentioned before, we annotated each noun type only once per participant type rather than annotating every mention of a noun separately. To compute the numbers depicted in Table 1, we copy the type-level annotation to every single appearance of the noun, to give a better idea of the actual distribution.

For both verbs and participants, the most frequent relation chosen by both annotators is *Identity*. Among the lexical relations, *Hyponymy* is more frequent than *Hypernymy*, which is consistent with the expectation that concrete event/participant mentions use more specific verbs/nouns than the abstract descriptions in the script knowledge base (cf. Modi et al. (2016)). *Diathesis*, *Incorporation* and *Hypernymy* for verbs, and *Meronymy*, *Hypernymy*, *Holonymy*, *Instance* and *Co-Hyponymy* for participants appear only very rarely and are subsumed under *Other* in the table.

5.2. Clausal Level

Figure 9 shows the distribution of the resulting clausal entailment labels for both annotators. *Identity* makes up for the largest part of cases (38 %), illustrating the high lexical coverage of crowdsourced script representations.

Entailment cases are significantly more frequent than *Reverse Entailment*, which is in line with the leading assumption that an event mention should entail a description of its event type, and with the observation that event-denoting clauses usually use longer sentences and more specific vocabulary than event descriptions given by the script knowledge base.

Both *Entailment* and *Reverse Entailment* mainly contain cases in which the participant is not realized on one side, and are only rarely composed from *Hypernymy* or *Hyponymy* cases. A typical example from the TAKING A BUS scenario is the text clause *wait for several minutes*. The most similar ED in the scenario is just *wait*, so the time expression is an additional participant and thus results in a clause-level entailment.

There is a large number of *Partial Entailment* cases (20%), which are in many cases composed of one or several *Inference* labels. One typical example of such a case from the TAKING A BUS scenario is the text clause *the driver pulled over*. The paraphrase sets only contain phrases like *bus stops* or *arrive at destination*. In this case, a system would need to know that *pull over* in the bus context is a paraphrase for

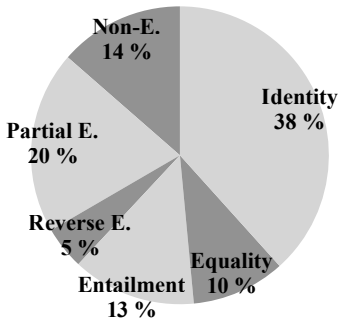


Figure 9: Distribution of composed labels.

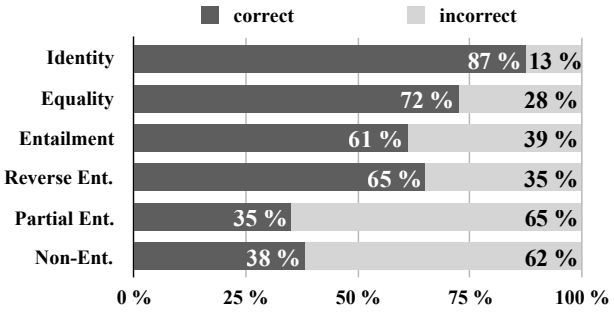


Figure 10: Performance of Ostermann et al. (2017) on clausal entailment classes.

stop, which requires contextual inference. These cases also appear in other scenarios, e.g. in the PLANTING A TREE scenario: *look at trees* is a contextual paraphrase of *choose a tree*. This can be seen as an indicator for the difficulty of the text-to-script mapping task, but it also indicates that the script resource is not exhaustive enough to contain all possible formulation variants for events.
Lastly, we found that a large number of *Non-Entailment* cases are derived from annotation errors in InScript.

6. Entailment Type and System Performance

The previous section investigated the difficulty of the task of text-to-script mapping itself. In this section, we now look at the performance of the only published model for text-to-script mapping and show that the cases we identified as complex are indeed most challenging for the model. We apply the entailment-type annotation to Ostermann et al. (2017)'s text-to-script mapping system, breaking down its event-labeling performance on our annotated InScript subcorpus to clause-level entailment classes. The results for all script-relevant clauses are shown in Figure 10.
Identity (verb and all head nouns are lemma-identical) is easiest to model, and so as expected provides the best results. That the accuracy is not 100% is due to the fact that sometimes the same verb lemma occurs in different paraphrase sets. This holds in particular for light verbs such as *get*, which can be used to instantiate many different events throughout a scenario. In the BAKING A CAKE SCENARIO, for example, *get* appears in the CHOOSE_RECIPE paraphrase set (*get your recipe*), as well as in GET_INGREDIENTS (*get a box of cake mix*), GET_UTENSILS (*get a pan*), etc.
In general, the important result is not the accuracy of the identity case in itself, but in connection with the high percentage of identity cases (38%). The high number of realization variants for event mentions contained in the paraphrase sets carry out a large part of the mapping task.
To illustrate the positive effect of a large paraphrase set on system performance, we compare the result with a situation in which there is only one ED per paraphrase set. To this end, we picked one ED randomly from the paraphrase sets (average size is 25, using a token-based count), as representative(s) of the event type, and then computed the number of verb identity cases automatically: They would drop from 58% to 26.9%. With a similar drop for participant identity cases, the resulting clause-based percentage of identity cases would be below 20% (instead of 38%), probably leading to a dramatic drop in labeling accuracy.
Partial entailment is the class that is most difficult to model. This result provides a complementary message about the script knowledge base. About 80% of the clause-level *Partial Entailment* cases contain a word-level *Inference* relation, which is a strong indicator of the need for methods able to handle more complex inference: Crowdsourced collections of linguistic realization variants help to avoid complex inference in many cases, but cannot completely replace it. This emphasizes the need for larger script data collections that cover even more description variants for events and participants.
The low performance on the *Equality* cases is mainly due to the fact that it subsumes the difficult *Diathesis* and *Phrasal Verb* relations. In contrast, the results for *Reverse Entailment* are better than one would expect: This is mostly due to the fact that most *Reverse Entailment* cases (approx. 85%) consist of combinations of *Equality* or *Identity* with an *Unrealized* participant. Even if one participant is unrealized, the correct label is mostly identified. For the text clause *I placed the tree*, *hole* is a required participant, occurring in every ED, but nevertheless the overlap is high enough to select the correct label. The more difficult *Hypernymy* cases make up only 15% of the *Reverse Entailment* cases. The situation is similar for the *Entailment* cases. Here, the simpler *Additional* participant cases make up 50%.
Finally, the model has low accuracy for the *Non-Entailment* cases. However, it is substantially above the random baseline. A possible reason is that the compositional computation of the clause-level entailment type amounts to a generalization to the worst case. Thus a number of pairs end up in the *Non-Entailment* class although there is only a minor local incompatibility.

7. Background and Related Work

Our work is based on script representations in which events are encoded as paraphrase sets. There also exist other research directions on modeling script knowledge. The most prominent alternative representation of script knowledge is that of *narrative chains*, proposed by Chambers and Jurafsky (2008) and subsequently extended by Chambers and Jurafsky (2009), Pichotta and Mooney (2014) and (Ahrendt and Demberg, 2016), to name but a few. Narrative chains

have been used for event prediction (Chambers and Jurafsky, 2009; Pichotta and Mooney, 2014; Rudinger et al., 2015; Modi et al., 2016) or the related *story cloze* task (Mostafazadeh et al., 2016; Pichotta and Mooney, 2016), in which complete sentences are predicted.

Narrative chains (and their aforementioned extensions) differ in two relevant aspects from the script representations used in our study. Instead of using paraphrase sets, events are represented as typed dependency relations between a verb and one of its dependents (the *protagonist*). Another difference is that narrative chains are intended to be learned automatically from large collections of unannotated text. By contrast, the script representations used in our study are learned from crowdsourced sequences of event descriptions, which are more focused and more detailed compared to narrative chains: They also contain events which are often not mentioned in text, since they are assumed to be background knowledge (Chambers, 2017). These two differences imply that the results of our annotation cannot easily be transferred to script representations along the lines of Chambers and Jurafsky (2008).

Text-to-script mapping is similar to the task of *recognizing textual entailment* (*RTE*, Dagan et al. (2006)), in which systems have to decide whether a *text* entails a *hypothesis*. The text entails the hypothesis if a human reader would infer from the text that the hypothesis is most likely true. In our case, event mentions and event descriptions correspond to texts and hypotheses, respectively. The lexical entailment annotation in our study is inspired by similar annotation efforts in the context of RTE, for instance Garoufi (2007), who used lexical entailment annotations to annotate the RTE-2 data set (Bar Haim et al., 2006). The label set we use is inspired by their set of lexical entailment categories, and they also conducted an alignment step similar to our *participant alignment*. Our computational derivation of clausal-level labels is built on MacCartney and Manning (2007), MacCartney and Manning (2009), and MacCartney (2009).

8. Conclusion

In this work, we annotated event mentions in narrative texts with semantic relations that need to be modeled when mapping the mentions to script events that are represented as paraphrase sets. We provide a lexical-level entailment annotation between event-denoting verbs and participant-denoting nouns of narrative texts on the one side, and event and participant descriptions of a script on the other side. We then derive clause-level entailment labels that highlight the coverage of crowdsourced paraphrase sets associated with event types, as compared to the textual variation in naturalistic texts. We find that script representations in the form of paraphrase sets can cover a large number of description variants of an event in a text. However, the alignment of a substantial amount of event mentions requires a deeper inference of multiple semantic relations.

Based on our annotation, we analyze the performance of an existing text-to-script mapping system on the different entailment classes. The results indicate (1) that paraphrase sets as constitutive part of script representations can massively increase the accuracy of text-to-script mapping systems and (2) that the tested model is mostly unable to account for the more complex cases.

The data set is available at `http://www.sfb1102.uni-saarland.de/?page_id=2582`.

Acknowledgements

We would like to thank the anonymous reviewers and Michael Roth for their helpful comments on the paper. Also, we thank our student assistants Sophie Henning, Sarah Mameche and Leonie Harter for their help with the annotations. This research was funded by the German Research Foundation (DFG) as part of SFB 1102 'Information Density and Linguistic Encoding'.

9. Bibliographical References

Ahrendt, S. and Demberg, V. (2016). Improving event prediction by representing script participants. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 546–551.

Bar Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., and Szpektor, I. (2006). The second PASCAL Recognising Textual Entailment Challenge.

Chambers, N. and Jurafsky, D. (2008). Unsupervised learning of narrative event chains. *Proceedings of ACL-08*.

Chambers, N. and Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*.

Chambers, N. (2017). Behind the scenes of an evolving event cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 41–45.

Dagan, I., Glickman, O., and Magnini, B. (2006). The PASCAL Recognising Textual Entailment Challenge. *Quiñonero-Candela, J.; Dagan, I.; Magnini, B.; d'Alché-Buc, F. (Eds.) Machine Learning Challenges. Lecture Notes in Computer Science*, 3944:177–190.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992). One sense per discourse. In *Proceedings of the Workshop on Speech and Natural Language*, pages 233–237. Association for Computational Linguistics.

Garoufi, K. (2007). Towards a better understanding of applied textual entailment. Master's thesis, Saarland University, Saarbrücken, Germany.

Gühring, T., Linz, N., Theis, R., and Friedrich, A. (2016). SWAN: an easy-to-use web-based annotation syste. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS). September 19-22, Bochum, Germany*.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):pp. 159–174.

MacCartney, B. and Manning, C. D. (2007). Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200. Association for Computational Linguistics.

MacCartney, B. and Manning, C. D. (2009). An extended model of natural logic. In *Proceedings of the eighth international conference on computational semantics*, pages 140–156. Association for Computational Linguistics.

MacCartney, B. (2009). *Natural Language Inference*. Ph.D. thesis, Stanford University.

Modi, A. and Titov, I. (2014). Inducing Neural Models of Script Knowledge. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*, Baltimore, MD, USA.

Modi, A., Anikina, T., Ostermann, S., and Pinkal, M. (2016). InScript: Narrative texts annotated with script information. *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 16)*.

Modi, A., Titov, I., Demberg, V., Sayeed, A., and Pinkal, M. (2017). Modelling Semantic Expectation: Using Script Knowledge for Referent Prediction. *Transactions of the Association for Computational Linguistics*, 5:31–44.

Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., Kohli, P., and Allen, J. (2016). A Corpus and Evaluation Framework for Deeper Understanding of Commonsense Stories. *Proceedings of NAACL-HLT 2016*, pages 839–849.

Ostermann, S., Roth, M., Thater, S., and Pinkal, M. (2017). Aligning Script Events with Narrative Texts. *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 128–134.

Pichotta, K. and Mooney, R. J. (2014). Statistical Script Learning with Multi-Argument Events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, volume 14, pages 220–229.

Pichotta, K. and Mooney, R. J. (2016). Using Sentence-Level LSTM Language Models for Script Inference. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 279–289.

Regneri, M., Koller, A., and Pinkal, M. (2010). Learning Script Knowledge with Web Experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 979–988, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rudinger, R., Rastogi, P., Ferraro, F., and Durme, B. V. (2015). Script Induction as Language Modeling. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1681–1686.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R., and Scheffczyk, J. (2006). FrameNet II: Extended theory and practice.

Schank, R. C. and Abelson, R. P. (1975). *Scripts, plans, and knowledge*. Yale University New Haven, CT.

Wanzare, L. D. A., Zarcone, A., Thater, S., and Pinkal, M. (2016). DeScript: A crowdsourced database of event sequence descriptions for the acquisition of high-quality script knowledge. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*.

Wanzare, L. D. A., Zarcone, A., Thater, S., and Pinkal, M. (2017). Inducing Script Structure from Crowdsourced Event Descriptions via Semi-Supervised Clustering. *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*.

Semantic Equivalence Detection: Are Interrogatives Harder than Declaratives?

João Rodrigues, Chakaveh Saedi, António Branco and João Silva
University of Lisbon
NLX-Natural Language and Speech Group, Department of Informatics
Faculdade de Ciências
Campo Grande, 1749-016 Lisboa, Portugal
{joao.rodrigues, chakaveh.saedi, antonio.branco, jsilva}@di.fc.ul.pt

Abstract

Duplicate Question Detection (DQD) is a Natural Language Processing task under active research, with applications to fields like Community Question Answering and Information Retrieval. While DQD falls under the umbrella of Semantic Text Similarity (STS), these are often not seen as similar tasks of semantic equivalence detection, with STS being implicitly understood as concerning only declarative sentences. Nevertheless, approaches to STS have been applied to DQD and paraphrase detection, that is to interrogatives and declaratives, alike. We present a study that seeks to assess, under conditions of comparability, the possible different performance of state-of-the-art approaches to STS over different types of textual segments, including most notably declaratives and interrogatives. This paper contributes to a better understanding of current mainstream methods for semantic equivalence detection, and to a better appreciation of the different results reported in the literature when these are obtained from different data sets with different types of textual segments. Importantly, it contributes also with results concerning how data sets containing textual segments of a certain type can be used to leverage the performance of resolvers for segments of other types.

Keywords: semantic text similarity, paraphrase detection, duplicate question detection

1. Introduction

Semantic Text Similarity (STS) is a Natural Language Processing (NLP) task whereby a system, given two input text segments, assigns to them a similarity score in a discrete or continuous scale that ranges from representing total similarity—for semantically equivalent segments—to representing total dissimilarity—for segments that are semantically independent.

The STS task has been part of the SemEval competitive shared tasks since 2012 (Agirre et al., 2012), together with other challenges for a wide variety of other tasks, such as plagiarism detection, sentiment analysis or relation extraction, to name but a few. More recently, SemEval embraced STS challenges that concern more focused tasks, like paraphrase detection, which consists of a binary decision on whether two input sentences are paraphrases of each other and, starting in 2016, a task on Duplicate Question Detection (DQD) (Nakov et al., 2016).

DQD appears as a special case of paraphrase detection, where the focus is on interrogative sentences: this task consists of a binary decision on whether two input interrogatives sentences are a duplicate of each other.

The motivation for the increasing interest in DQD, and the inclusion in SemEval of challenges dedicated to DQD, comes from the increasing popularity of on-line Community Question Answering (CQA) forums, such as Stack Exchange[1] or Quora[2]. These forums are quite open in allowing any user to post questions (and answer questions from other users) but from this arises a potential problem that may eventually affect the effectiveness of these on-line services, namely that many posted questions are duplicates of questions already answered. In such cases, the user posting the duplicate question should be directed to the already existing question. Duplicate questions are manually flagged by the users, but this effort quickly becomes unwieldy as the site grows in popularity, driving the need for automatic procedures for DQD.

Though the interest in DQD may be seen as relatively recent, there is an accumulated body of lessons learned about this task and the expected performance of systems tackling it, some of them being quite in line with what is known about data-driven approaches in general, while some others are more specific for this task. From existing work on DQD, such as (Bogdanova et al., 2015) (Rodrigues et al., 2017) and (Saedi et al., 2017), one learned that (i) training and evaluating over a specific domain with less data, rather than over a generic one with more data, will likely lead to better performance; (ii) training on as much data as possible, gathered from all different domains, and evaluating on a specific domain yields little more than random choice performance; (iii) when training on data sets of interrogative sentences, differences in the average length or in the level of grammaticality of sentences have little impact on performance; (iv) the differences in performance between the major types of approaches to DQD become smaller as the domain becomes more generic; and (v) the best variants of these major approaches all deliver competitive results when trained with general domain data sets with 30,000 sentence pairs, with accuracy scores falling within a range of just 2 to 3 percentage points.

The underlying semantic relation between sentences that STS is seeking to model is the one of synonymy. Interestingly, while it concerns the ultimate notion of semantic equivalence for both types of sentences, declaratives and interrogatives alike, the synonymy relation has quite different operational definitions for each one of them. Two declarative sentences are synonymous (or paraphrases of

[1] http://stackexchange.com/
[2] http://quora.com/

each other) just in case they can replace each other and the truth conditions of any text they happen to be part of are preserved under that substitution (modulo so-called opaque contexts). Two interrogatives, in turn, are synonymous (or duplicates of each other) just in case any successful answer to any one of them is also a successful answer to the other one.

- Declarative duplicate pair (from MSRPC)
  - Dogs, he said, are second only to humans in the thoroughness of medical understanding and research.
  - He said that dogs are second only to humans in terms of being the subject of medical research.
- Interrogative duplicate pair (from DupStack)
  - Where did the notion of "one return only" come from?
  - Should I return from a function early or use an if statement?

Figure 1: Examples of duplicate pairs

The examples in Figure 1 are instances that comply with these two operational definitions of semantic equivalence, for declaratives and for interrogatives. It does not go unnoticed that the superficial similarity—in terms of common words, word order, length, etc.—between the interrogative sentences is much more rarefied than between the declarative ones. And it is on the basis of superficial features that decisions on the eventual underlying relations of semantic equivalence are made. The contrast between these two illustrative pairs of examples thus strongly suggests that, when it comes to STS, we may be facing two tasks of synonymy detection of quite different levels of difficulty, depending on whether we are modeling synonymy between declaratives, or between interrogatives.

Against this background, what has not been studied yet, and remains an interesting research question, is whether the operational and qualitative difference between the synonymy relations for declarative and for interrogative sentences leads to an impact and a substantive difference between the performance of STS systems for declaratives, on the one hand, and for interrogatives, on the other hand. Or, in other words, in what concerns the automatic detection of semantic similarity, are interrogative sentences more difficult to handle than declarative ones given the current methods at hand to tackle them?

This is the driving research question we seek to address and that motivates the experiments reported in the present paper, as well as other subsidiary research questions.

The remainder of this paper is organized as follows. Section 2 describes the related work. We present the data sets and the experimental approaches to undertake STS in Section 3. In Sections 4, 5 and 6, the experiments carried out are reported and the results of their evaluation are presented, that address the research questions, respectively, whether interrogatives are harder than declaratives, how difficult are Tweets for semantic equivalence detection, and whether merging data sets for different type of textual segments improve the performance of the resolvers. Section 7 concludes with final remarks.

2. Related Work

In a recent study, Saedi et al. (2017) undertook a systematic comparison of the performance of different major approaches to DQD over progressively larger data sets, by considering approaches that have been identified in the literature as very competitive, namely rule-based, support vector machines classifiers, and deep convolutional neural networks.

A major finding of that systematic study of the learning curves of these major approaches is that there is no approach that beats all others in every point of the learning curve. Simpler, rule-based approaches, like the Jaccard index, are highly competitive for small data sets, but as more data becomes available, they lose out to more sophisticated approaches. In particular, and confirming a widespread assumption in Machine Learning, deep learning approaches come into their own, and its performance surpasses all other approaches, only when a sufficiently large amount of training data is available, containing above 30,000 pairs of sentences with a 50/50 split between duplicates and non-duplicates.

3. Data Sets and Approaches to Semantic Equivalence Detection

This Section describes the data sets and the different approaches to STS that were used in our experiments.

3.1. Data sets

To carry out the proposed experiment, at least two corpora are required, one with interrogative sentences and another with declarative sentences. The corpora should be as close as possible to each other in other aspects, particularly in terms of domain, size and class distribution, in order to obtain results that can be as comparable as possible. To support the testing of our central hypothesis that interrogatives are harder than declaratives in terms of semantic equivalence detection, we resorted to two data sets, one with interrogatives (Quora) and another with declaratives (MSRPC). We also included a third data set, with a mixture of declarative and interrogative sentences (DupStack), and a fourth one (PIT), with segments of highly compromised grammaticality.

These data sets, all in English, are introduced below.

MSRPC is the Microsoft Research Paraphrase Corpus (Dolan and Brockett, 2005). It consists of 5,801 pairs of edited and grammatically well-formed declarative sentences taken from news articles on various topics, with each pair being annotated with a binary label indicating if its sentences are paraphrases of each other or not. There are 3,900 pairs with paraphrases, and the sentences are on average 18.92 words long.

Quora is a corpus originating from the CQA forum Quora (Iyer et al., 2017). It contains over 404,289 pairs of

edited and grammatically well-formed questions, annotated as to whether they contain duplicates or non-duplicates, of which 149,263 are pairs with duplicates. These questions address any topic and on average are 11.06 words long.

DupStack is the acronym used in this paper for CQADupStack (Hoogeveen et al., 2015), a corpus composed of pairs of threads from StackExchange, another popular CQA site, annotated with information as to whether they are a duplicate. The 3,891,016,008 pairs of threads are sourced from 12 subforums of StackExchange resorting to the original classification split, covering a number of subtopics mostly in the ICT domain, of which 10,677 contain duplicate pairs. These segments have an average length of 8.48 words, with some of them presenting sub-optimal grammaticality. They were entered by the users as the title of a larger text where the information being searched for is indicated, and thus with many of them appearing clearly as declaratives.

PIT is the corpus Paraphrase and Semantic Similarity in Twitter for the SemEval-2015 Task 1 (Xu et al., 2015). It contains over 18,000 pairs of segments, taken from Twitter. These sentences tend thus to be relatively short and make extensive use of abbreviations and a highly compromised grammaticality. There are 5,641 pairs with paraphrases, and the segments have on average a length of 8.13 words.

For the sake of comparability of the experimental results to be obtained on the basis of these data sets, we take an equal number of sentence pairs from each corpus, randomly selected, but ensuring a balanced distribution with an equal number of duplicate and non-duplicate cases. The smallest corpus, MSRPC, with 3,900 duplicate (and 1,901 non-duplicate) pairs, constraints the maximum number of sentence pairs that can be picked. Accordingly, from each corpus, 3,900 duplicate and 3,900 non-duplicate pairs are randomly selected, for a total size of 7,800 pairs per corpus. Note that MSRPC only has 1,901 non-duplicate pairs. The other 1,999 non-duplicate pairs in the respective sub-corpus are generated by randomly pairing sentences taken from distinct pairs.
In all experiments, 80% of the pairs are used for training and 20% for testing.
Table 1 summarizes information on the type and size of the sub-corpora used.

	type	grammaticality	#tokens
MSRPC	declar.	ok	301,428
Quora	interrog.	ok	177,334
DupStack	mixed	sub-optimal	142,387
PIT	tweets	highly compromised	137,898

Table 1: The four sub-corpora, each with 7,800 pairs.

While the number of sentence pairs is the same for all corpora, MSRPC has a much higher number of tokens. This happens because the sentences in that corpus, which are taken from news articles, are usually longer than the questions from Quora and DupStack, or the tweets from the PIT corpus.

3.2. Approaches to semantic equivalence detection

We use the same set of approaches for DQD from (Rodrigues et al., 2017) and (Saedi et al., 2017), as these cover a range of different methods with state-of-the-art performance for the size of training data there are available for the present experiments. Given the hypothesis that detecting synonymy between interrogatives is harder than between declaratives, we resort to approaches to semantic equivalence detection with highly competitive performance for DQD, in order to explain away a possible justification for the difference in performance between the two types of sentences based on the putative weakness of the methods used vis a vis interrogatives.
In this paper we provide a short summary of each approach and direct the reader to the articles cited above for further information.

Jaccard The Jaccard index is a straightforward statistic based on the count of common of n-grams between the two segments being compared. It is used as a simple baseline that previous work has shown to nonetheless be very competitive (Wu et al., 2011), especially for small sized data sets below 30,000 pairs (Saedi et al., 2017). All n-grams, with n ranging from 1 to 4, are used.

SVM Support Vector Machine classifiers have been used with success in many NLP tasks and are able to cope with a great variety of features. The set of features used in this work is formed by (i) two vectors with the one-hot encodings of n-gram occurrences in each segment; (ii) the Jaccard index scores for 1, 2, 3 and 4-grams; (iii) the counts of negative words (e.g. *never*, *nothing*, etc.) in each segment; (iv) the number of nouns that are common to both segments; and (v) the cosine similarity between the vector representation of each segment.

DCNN Deep Neural Networks have, over the past few years, gained popularity and been applied to many NLP tasks, often surpassing by a large margin the other alternative approaches if sufficiently large training data is available. In this work, we use the architecture introduced in (Rodrigues et al., 2017), which combines a convolutional neural network and a deep network in a Siamese architecture.

4. Are Interrogatives harder than Declaratives?

In this Section, we present data and experiments whose results are suited to bring empirical evidence that can support our research hypothesis that interrogatives are harder than declaratives in terms of semantic equivalence detection.

4.1. Semantic equivalence

A first and straightforward experiment consists of training and evaluating each one of the three working approaches for semantic equivalence detection over each one of the two principal data sets, the ones with declaratives (MSRPC) and with interrogatives (Quora). The results are summarized in Table 2. Note that for SVM and DCNN, the scores shown are the average of 3 runs.

	Jaccard	SVM	DCNN
Declaratives (MSRPC)	**77.30**	**80.46**	**78.42**
Interrogatives (Quora)	69.29	69.10	72.78

Table 2: Accuracy scores (%) of the different approaches applied to the two different types of segments, with the **highest values** (in bold) for declaratives.

The three approaches perform better for declaratives than for interrogatives across the board by a substantial margin, that ranges from over 5 (DCNN) to over 11 percentage points (SVM). This provides clear empirical support for our research hypothesis.

4.2. Superficial overlap

As noted in Section 1, a major motivation to put forward our research question is the observation of the different operational definitions for the semantic equivalence of declaratives and of interrogatives. Under these definitions, for two sentences to be equivalent, less superficial commonalities between them seem to be expected to hold on average for interrogatives than for declaratives.

A way to obtain a quantitative test for this expectation is to measure the level of superficial overlap. For this purpose, we calculate the average BLEU score between the sentences in equivalent pairs and between the sentences in non-equivalent pairs, where higher scores reflect higher overlap. The results are reported in Table 3.

	equivalent	non-equivalent
Declaratives (MSRPC)	49.73	18.93
Interrogatives (Quora)	31.20	17.33

Table 3: Averaged BLEU scores of sentences in equivalent and non-equivalent pairs

With the highest score for declaratives (49.73), and a difference to interrogatives of over 18 BLEU points, for sentences that are semantic equivalents of each other, these scores provide an objective confirmation that equivalent interrogatives have less superficial overlap than equivalent declaratives. Hence, this grants objective support for the expectation that interrogatives should be harder than declaratives for semantic equivalence detection, as the mainstream approaches for this task are data-driven and rely on the superficial similarity of sentences for their operation.

4.3. Mix of types

An additional piece of evidence that may support our research hypothesis can be looked for in the performance of the equivalence detection systems when running over a data set like DupStack, composed by a mixture of declarative pairs and interrogative pairs, and even mixed-type pairs.

- Duplicate mixed-type pair (from DupStack)
  - Turn-by-turn direction using PgRouting
  - How to emulate Google Maps driving directions using pgRouting?
- Non-duplicate mixed-type pair (from DupStack)
  - SLD: OGC Filter set, but symbolizer expected
  - How to delete coordinate system from raster file with prj.adf?

Figure 2: Example of mixed-type pairs

If our hypothesis holds, the accuracy scores for this data set should lie between the higher and lower scores of declaratives and interrogatives, respectively. The results of this experiment are reported in Table 4.

	Jaccard	SVM	DCNN
Declaratives (MSRPC)	77.30	80.46	78.42
Mixed (DupStack)	**74.16**	**71.23**	81.51
Interrogatives (Quora)	69.29	69.10	**72.78**

Table 4: Accuracy scores (%) of the different approaches applied to the different types of segments, with the **in-between values** (in bold) for the data sets with a mix of declaratives and interrogatives on two of the three approaches.

In two (Jaccard and SVM) of the three approaches, the scores are in line with this prediction, with in-between values. Overall, this provides yet another piece of empirical evidence to the research hypothesis, with the value for DCNN (81.51) appearing as an outlier.

5. How difficult are Tweets after all?

In the context of the results reported in the previous Sections, a third interesting research question to address is to determine how difficult may be the task of semantic equivalence detection for Tweets.

Previous results (Rodrigues et al., 2017) indicate that to a certain extent, reducing the average size of interrogative segments and relaxing the grammaticality of interrogative segments have little impact in equivalence resolvers for interrogatives. But when compared with Tweets—much shorter and with much more compromised grammaticality—these appear as mild differences and the respective may be of little guidance when we turn to Tweets.

To answer this third research question, we trained and evaluated the three approaches for resolving semantic equivalence over PIT, the data set with Tweets. Their performance results are reported in Table 5.

The three approaches perform worse for Tweets than for the second worst type of segments, viz. interrogatives, by

- Duplicate pair (from PIT)
  - That 3pointer from Kevin Durant was lucky asf
  - the NBA gods showed favor to that 3
- Non-duplicate pair (from PIT)
  - Aye mac miller new music is aite
  - I swear my waiter is Mac Miller

Figure 3: Examples of Twitter pairs

	Jaccard	SVM	DCNN
Declaratives (MSRPC)	77.30	80.46	78.42
Mixed (DupStack)	74.16	71.23	81.51
Interrogatives (Quora)	69.29	69.10	72.78
Tweets (PIT)	**67.82**	**67.83**	**51.70**

Table 5: Accuracy scores (%) of the different approaches applied to the different types of segments, with the **lowest values** (in bold) for Tweets

a margin that ranges from almost 1.5 (SVM) to over 20 percentage points (DCNN). When the resolvers for Tweets are compared to the best resolvers, for grammatical, manually edited declaratives (MSRPC), this gap widens up for a range from about 10 (Jaccard) to almost 30 percentage points (DCNN). This indicates that Tweets are the hardest type of segments in terms of semantic equivalence detection.

6. Interrogatives and Declaratives Leveraging each other?

The results in Section 4 help to clarify that interrogatives are harder than declaratives in terms of the mainstream approaches to resolve semantic equivalence detection.
Interestingly, this relative advantage gets reverted when it comes to data sets. It is easier to find, collect and support interrogatives with larger data sets, than declaratives. The reason is that semantic equivalent interrogatives happen to be generated in real usage scenarios, as e.g. in the Quora service, making them easy to crowdsource, while declaratives are not.[3] Hence a next research question is to empirically determine whether, and to what extent, the more abundant data sets with interrogatives can help improve the detection of equivalent declaratives.

6.1. Cross training

A first experiment seeks to assess how performance is impacted by differences between the training data and the testing data. For this purpose, the overall best system, DCNN, is trained on each corpus and each resulting model is evaluated on each corpus. The results are summarized in Table 6. When trained on data sets with declaratives (MSRPC) and with mixed types (DupStack), the best performance is observed when the systems resolve the equivalence for similar types of segments, achieving 78.42 (first row) and 81.51 (second row), respectively.

	Evaluate on...		
Train on...	MSRPC	DupStack	Quora
Declar. (MSRPC)	**78.42**	56.28	61.67
Mixed (DupStack)	70.58	**81.51**	56.73
Interrog. (Quora)	**76.35**	54.23	72.78

Table 6: Accuracy scores (%) with rows showing training data sets and columns showing evaluation data sets, concerning DCNN, with the **highest value** (in bold) for each data set

Very interestingly, when the resolver is trained with the interrogatives (third row), it has the best performance when deciding about declaratives (76.35), and the second best about interrogatives themselves (72.78).
The opposite, however, does not hold. Resolving interrogatives (third column) with systems trained on another type of segments only delivers results that are clearly worse (61.67 and 56.73) than when they are trained on interrogatives themselves (72.78).
This result allows good hopes that the more abundant pairs of duplicate interrogatives can be of help to leverage the performance of resolvers of semantic equivalence between declaratives. Very large data sets with interrogatives collected from real usage scenarios may eventually support the development of resolvers with the best performance than the ones trained on the smaller, and hard to obtain and to expand by explicit manual annotation, data sets containing only declaratives paraphrases. This is the motivation for our next experiment.

6.2. Merged data sets

A second experiment consists thus in training resolvers for declaratives over data sets that contain larger data sets than the MSRPC corpus (with declaratives) alone. These data sets are obtained by resorting to a larger subset of Quora with interrogatives (from 7,800 to 100,000 pairs), and to the merging of this and the other data sets (with 7,800 pairs each) used in this paper.
The results of this experiment are reported in Table 7.

Train on...	Evaluate on... Decl(MSRPC)
all[1]	79.42
Decl(MSRPC) + Interr(Quora100k)	79.36
Decl(MSRPC)	78.42
Interr(Quora100k)	74.42

[1] Dec(MSRPC) + Int(Quora100k) + Mix(DupStack) + Tw(PIT)

Table 7: Accuracy scores (%) with rows showing training data sets and column showing the evaluation data set, concerning DCNN

This experiment permits to understand that by growing the size of the training data set with interrogatives by one order of magnitude—from 7,800 to 100,000 pairs, of the new

[3] In the MSRPC corpus, the sentences in the pairs had to be annotated by humans for the specific purpose of the construction of this corpus.

sub corpus Quora100K—is not enough to obtain better results for declaratives (74.42, last row) than when the training data set is smaller—7,800 pairs, of the MSRPC corpus — but made only of declaratives (78.42, penultimate row). Very interestingly, this experiment allows also to understand that when the larger training data set is obtained by including also the MSRPC data set with declaratives—79.36 by adding Quora100k to it, and 79.42 by adding this and all other data sets to it—, that is enough to overcome the performance of the system trained only with declaratives (78.42, penultimate row).

These results clearly indicate that adding pairs of interrogatives to the training data set of declaratives is an effective way to improve the performance of paraphrase resolvers. Importantly, this is also a procedure that dispenses with a further specific human effort for the construction of the training data set as pairs of duplicate interrogatives can be collected as a byproduct of on-line Community Question Answering forums.

7. Conclusions

In this paper we addressed three major research questions related to the task of semantic equivalence detection, and performed a number of experiments that permitted to gather empirical evidence aimed at finding answers for them.

The major driving question is whether interrogatives are harder than declaratives for semantic equivalence resolvers. The higher superficial overlap between declaratives in paraphrasing pairs, than between interrogatives in duplicate pairs, as measured with BLEU; and the substantially superior performance over declaratives, than over interrogatives, of different resolvers developed under major mainstream approaches: all these are major pieces of evidence supporting the observation that this task is harder with interrogatives.

A second important research question was whether the performance of resolvers for a given type of segments, interrogative or declarative, can be improved by obtaining larger training data sets that result from the merging of smaller data sets for different types. The contrasting levels of performance of a number of systems developed under these circumstances permitted to observe that this is actually the case with resolvers for declaratives (trained with the merging of data sets with declaratives and interrogatives), but not for interrogatives.

In the context of the previous questions and respective answers, a last research question was the inquiry on how difficult is the task of semantic equivalence detection for Tweets, in comparison to the similar task for grammatically well-formed declaratives and interrogatives. The inferior performance results across the board with different resolvers developed under major mainstream approaches as used in previous experiments permitted to gather empirical evidence indicating that Tweets are the most difficult type of segments for the task of semantic equivalence detection. These results contribute to a better understanding of the strengths and weaknesses of current mainstream methods for semantic equivalence detection, and allow to better appreciate and ponder on the different relevance of the results and scores reported in the literature when these are obtained from different data sets and with different types of textual segments.

8. Acknowledgments

The present research was partly supported by the Infrastructure for the Science and Technology of the Portuguese Language (CLARIN Língua Portuguesa), by the National Infrastructure for Distributed Computing (INCD) of Portugal, and by the ANI/3279/2016 grant.

9. References

Agirre, E., Gonzalez-Agirre, A., Cer, D., and Diab, M. (2012). Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics (*SEM 2012)*, pages 385–393.

Bogdanova, D., dos Santos, C. N., Barbosa, L., and Zadrozny, B. (2015). Detecting semantically equivalent questions in online user forums. In *Proceedings of the 19th Conference on Computational Natural Language Learning (CoNLL)*, pages 123–131.

Nakov, P., Marquez, L., Moschitti, A., Magdy, W., Mubarak, H., Freihat, A. A., Glass, J., and Randeree, B. (2016). Semeval-2016 task 3: Community question answering. In *Proceedings of the 11th International Conference on Semantic Evaluation (SemEval)*, pages 27–48.

Rodrigues, J., Saedi, C., Maraev, V., Silva, J., and Branco, A. (2017). Ways of asking and replying in duplicate question detection. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*.

Saedi, C., Rodrigues, J., Silva, J., Branco, A., and Maraev, V. (2017). Learning profiles in duplicate question detection. In *Proceedings of the IEEE 16th International Conference on Information Reuse and Integration (IEEE IRI 2017)*.

Wu, Y., Zhang, Q., and Huang, X. (2011). Efficient near-duplicate detection for Q&A forum. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 1001–1009.

10. Language Resource References

Dolan, B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the 3rd International Workshop on Paraphrasing (IWP 2005)*.

Hoogeveen, D., Verspoor, K. M., and Baldwin, T. (2015). CQADupStack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian Document Computing Symposium*, pages 3:1–3:8.

Iyer, S., Dandekar, N., and Csernai, K. (2017). First Quora dataset release: Question pairs.

Xu, W., Callison-Burch, C., and Dolan, B. (2015). Semeval-2015 task 1: Paraphrase and semantic similarity in Twitter (PIT). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval)*, pages 1–11.

CEFR-based Lexical Simplification Dataset

Satoru Uchida†, Shohei Takada‡, Yuki Arase‡
† Faculty of Languages and Cultures, Kyushu University
‡ Graduate School of Information Science and Technology, Osaka University
† uchida@flc.kyushu-u.ac.jp, ‡{takada.syouhei, arase}@ist.osaka-u.ac.jp

Abstract

This study creates a language dataset for lexical simplification based on Common European Framework of References for Languages (CEFR) levels (CEFR-LS). Lexical simplification has continued to be one of the important tasks for language learning and education. There are several language resources for lexical simplification that are available for generating rules and creating simplifiers using machine learning. However, these resources are not tailored to language education with word levels and lists of candidates tending to be subjective. Different from these, the present study constructs a CEFR-LS whose target and candidate words are assigned CEFR levels using CEFR-J wordlists and English Vocabulary Profile, and candidates are selected using an online thesaurus. Since CEFR is widely used around the world, using CEFR levels makes it possible to apply a simplification method based on our dataset to language education directly. CEFR-LS currently includes 406 targets and 4912 candidates. To evaluate the validity of CEFR-LS for machine learning, two basic models are employed for selecting candidates and the results are presented as a reference for future users of the dataset.

Keywords: Lexical simplification, ESL learners, paraphrasing, CEFR

1. Introduction

There is no doubt that vocabulary is the key to successful international communication. Laufer (1989) points out that learners of a foreign language need to know 95% of the words in the input text to be able to successfully understand the text message, which implies the importance of vocabulary in language learning. However, authentic English passages may contain difficult words that may hinder the readers' or listeners' comprehension.

One solution is to simplify words in the input text using statistical and computer-based methods (Horn et al., 2014; Biran et al., 2011; Glavas and Stajner, 2015). This task is called *lexical simplification* whose aim is to replace a difficult word (referred to as the *target* word in this paper) with a simpler word selected from candidates (referred to as *candidate* words in this paper), which has been featured at SemEval2012[1]. Although previous studies have shown that data-driven approaches to find a simpler word are useful to some extent (Horn et al., 2014; Glavas and Stajner, 2015), they do not place the focus on educational aspects and hence tend to discuss only technical issues. Also, the criteria for the word difficulty tend to be vague and subjective, which would imply that previous research methods may not be useful for suggesting simplifications for different proficiency levels of learners.

The present study attempts to construct a dataset that uses Common European Framework of Reference for Languages (CEFR) as a criterion for word levels using introductory parts of university textbooks whose contents are academically reliable, educationally important, and covering a variety of topics. In this sense, our dataset is education-oriented based on a solid and widely-used framework for language education. CEFR levels are assigned to both the target and candidate words for lexical simplification. This allows us to adjust the text level flexibly according to learners' proficiency, which is especially meaningful for educational purposes. Also, two basic models for lexical simplification will be applied to our dataset in order to show the performances of baseline methods.

Food is procured with its suckers and then crushed using its tough "beak" of chitin. Target: procured Candidates (frequency): obtained (17), gathered (9), gotten (8), grabbed (4), ...

Table 1: Horn et al. (2014) example 1

2. Related Work

Coster and Kauchak (2011b) created and published a corpus for *text* simplification pairing sentences from Wikipedia and Simple Wikipedia (hereafter referred to as the Wikipedia corpus). This dataset has 137K pairs of simplified and unsimplified sentences that can be used for creating simplification rules. Coster and Kauchak (2011a) was one such attempt, and they examined a variety of paraphrasing rules including lexical changes, reordering, insertions and deletions using the Wikipedia corpus. Another attempt was Horn et al. (2014), who evaluated paraphrases using language models based on several language resources.

For evaluation tasks of lexical simplification, Horn et al. (2014)[2] published a dataset consisting of 500 sentences from the Wikipedia corpus with the results of annotations by Amazon Mechanical Turk[3]. In this dataset, they asked 50 turkers for each sentence to replace a target with a simpler one, and ranked the candidate according to the frequency (see Table 1) . The results seem to be natural and intuitive, but the levels of the candidates are not necessarily easy for learners because of the lack of educational criteria.

[1] https://www.cs.york.ac.uk/semeval-2012/task1.html

[2] http://www.cs.pomona.edu/~dkauchak/simplification/

[3] https://www.mturk.com/mturk/welcome

The pulses were either short or long, representing the dots and dashes of Morse code. Target: pulse Candidates (ranking): sound (1st), beat (2nd), beep (3rd), pulse (4th), emanation (5th)

Table 2: SemEval 2012 dataset

But critics note that Francis Galton did not advocate coercion when he defined the principles of eugenics. Target: advocate Candidates: like, cause, apply, speak, allow, want, ...

Table 3: Horn et al. (2014) example 2

Another dataset to be mentioned here is task 1 of SemEval 2012 which asked researchers to rank the candidates in order of simplicity (see Table 2). The sentences were taken from task 10 of SemEval 2007[4], where candidates were given by five English native speakers and ranked by non-native speakers of English. Although this may possibly reflect learners' intuition about the difficulty of words, it is hard to say that the judgments are consistent because of the differences in their mastery of English.

It should be also pointed out that there are some words in the list of candidates that require outside knowledge for replacement. Table 3 from the dataset of Horn et al. (2014) serves as an example. If we take the verb "advocate" out of the context, it is difficult to list "like", "cause", and "apply" for substitution. In this sense, the candidate lists are not consistent, which may make the task using the dataset beyond the scope of lexical simplification.

3. CEFR-LS

To overcome the disadvantages of the existing datasets, this study created a CEFR-based language resource ver. 1.0 (hereafter referred to as CEFR-LS). The input sentences were taken from introductory chapters of university textbook available at OpenStax website[5] by Rice University[6]. One of the reasons to use university textbooks is that they are deemed to be excellent in quality and their contents are reliable and well-organized. Also, the OpenStax website provides introductory books on a number of academic fields, and hence the dataset can be extended to cover various topics. Finally, introductory textbooks are literally a gateway to academic fields, and thus it is meaningful to provide an assistance in lexical aspects of these textbooks.

3.1. Procedure

Some introductory textbooks on the OpenStax website were randomly selected (the dataset includes those of economics, psychology, sociology and so on) and the introduction parts were used for setting up the dataset.

Using CEFR-J wordlist (Tono, 2016) and English Vocabulary Profile (Cambridge University Press, 2015), a wordlist for CEFR levels was created. This list also contains part-of-speech information, which are useful to determine the level of a word in different part-of-speech (e.g. a word "address" is A1 when it is used as a noun, while B1 as a verb).

CEFR levels consist of A1 (elementary), A2, B1, B2, C1, and C2 (advanced). In our dataset, words that are ranked as B2, C1, and C2 are selected as target to be simplified.

For selecting candidates, an online thesaurus[7] was employed and words equal to or higher than B2 level were excluded. Table 4 shows an example in CEFR-LS. Synonyms for "scarcity" (C2) are "dearth" (NA), "lack" (A2), "paucity" (NA), "shortage" (B1) and so on, but only "lack" (A2) and "shortage" (B1) are listed as candidate in our dataset (NA represents that a candidate is not included in the CEFR wordlist).

Then, the paraphrasability was checked by a native speaker of English (male, late 30s), who has taught in Japan for more than 10 years. The dataset is still under development, but currently it includes 406 targets and 4912 candidates.

Economics is the study of how humans make decisions in the face of scarcity. Target: scarcity (C2) Candidates: *dearth* (NA), lack (A2), *paucity* (NA), shortage (B1), ...

Table 4: CEFR-LS example (words in italics are excluded)

3.2. Annotation Standard

To make the judgments objective, we set up a guideline for annotation, which can also be helpful for the future updates of our dataset accompanied with additional judgments by several annotators. The procedure of checking paraphrasability consists of three stages: grammatical reformation stage, definition stage, and context stage.

Grammatical reformation is defined as addition or removal of words to maintain the coherence of the sentence. Candidates that fail at this stage will be unable to complete the sentence without need for the sentence to be reformed, regardless of their semantic proximity to the target word. Note that the morphology of the target (third person singular, past tense etc.) is considered to be applied to the candidate automatically in the process of paraphrasing. The following case serves as an example:

(1) Chemical engineering, materials science, and nanotechnology combine chemical principles and empirical findings to produce useful substances, ranging from gasoline to fabrics to electronic.
Target: range (B2)
Candidate: cover (A2)

In this sentence, the verb "range" is the target to be

[4] http://nlp.cs.swarthmore.edu/semeval/tasks/task10/summary.shtml

[5] http://cnx.org/

[6] Those books were created under Creative Commons 4.0 License (https://creativecommons.org/) and can be reused and distributed for research purposes.

[7] http://www.thesaurus.com/

paraphrased. The candidate "cover" does not fit due to the need to remove the preposition "from" to make the sentence grammatically correct.
At the definition stage, the semantic aspects of both the target and candidate words are taken into consideration. Candidates will be judged as semantically proximate if they paraphrase a portion of the semantics of the target and they provide the reader with a word they are more likely to be familiar with (an example will be shown later).
If a candidate passes these stages, the word is examined in the context stage, where each candidate is checked if it successfully conveys the nuance of the target word in the specific context and does not affect the meaning of a sentence. Here the register of a word is not considered except in extreme cases – simplification often results in candidates being simpler and therefore more informal words than the target words.

(2) This civic engagement ensures that representative democracy will continue to flourish and that people will continue to influence government.
Target: influence (B2)
Candidates: determine (B1), impress (A2), change (A1)

In (2), where the target is the verb "influence" (B2), "determine" (B1) passes the grammatical reformation stage but fails at the definition stage because it does not cover the semantics of "influence" in the meaning of affecting the way someone thinks or behaves, though it might do so in another context where determination causes a change in someone's way of thinking. On the other hand, the word "impress" (A2) passes this stage because it can be used to mean to affect the way someone believes. However, it does not pass the context stage because "impressing government" is not synonymous to "influencing government" in this context. Finally, the candidate "change" (A1) passes all the stages because it is grammatically appropriate, semantically proximate and contextually synonymous to the target word.

3.3. Characteristics of CEFR-LS

One of the characteristics of our dataset is that it is based on CEFR. In the field of language education, CEFR has become one of the most widely-used criteria for evaluating language ability. This means that it is possible to share the standard of word levels among many language learners and teachers. Furthermore, it enables us to connect lexical aspects with other language aspects such as grammar[8] based on the framework of CEFR. Also, the word level can be adapted flexibly based on a learners' proficiency. If a researcher wants to establish a system that simplifies words into A levels, then he/she can omit B1 level candidates from the list.
Table 5 shows the distribution of CEFR levels for both target and candidate words in SemEval2012, Horn et al. (2014), and CFER-LS. It is clear that other two resources contain a number of A1, A2, and B1 level words as a target that might not be necessary to be replaced, and B2, C1, and C2 level words as candidate that might still be difficult for learners.

[8] http://www.englishprofile.org/english-grammar-profile

	SemEval2012		Horn2014		CEFR-LS	
	TAR	CAN	TAR	CAN	TAR	CAN
Total	1710	8596	500	5010	406	4912
A1	616	1528	16	1119	0	1127
A2	414	1429	73	923	0	1495
B1	335	1762	152	1077	0	2290
B2	224	1080	95	493	301	0
C1	20	156	14	65	35	0
C2	20	147	18	64	70	0
NA	81	2494	132	1269	0	0
TAR ($\geq$ B2)	15.4%		25.4%		100%	
CAN ($\leq$ B1)	54.9%		62.9%		100%	

Table 5: CEFR levels of target and candidate words, where "TAR" and "CAN" abbreviate "target" and "candidate," respectively.

	SemEval2012	Horn2014
Number of targets (Coverage in the thesaurus as a headword)	1710 (94.3%)	500 (94.8%)
Number of candidates (Coverage in the thesaurus)	8596 (40.5%)	5010 (36.5%)

Table 6: Coverage of candidates in the thesaurus

	CEFR-LS
Number of targets	406
Number of candidates	4912
Number of correct candidates	961
Average Number of candidates per target	12.1
Average Number of correct candidates per target	2.4
Number of sentences	271
Average Number of words per sentence	23.0

Table 7: Detailed statistics of CEFR-LS

Another feature is that candidate words in CEFR-LS are all taken from a thesaurus. This means that most simplifications in our dataset do not require the world knowledge. This makes the simplification task more feasible and consistent. Table 6 shows the percentages of candidate words in each study that appear in the thesaurus via target words as a headword. It is obvious that less than half of the words are not included in the dictionary suggesting that outside knowledge is required for replacement in many cases.

Finally, our dataset contains not only correct candidates but also incorrect candidates in the list (see Table 7 for details). This allows researchers to examine distinctive statistical scores between correct and incorrect ones.

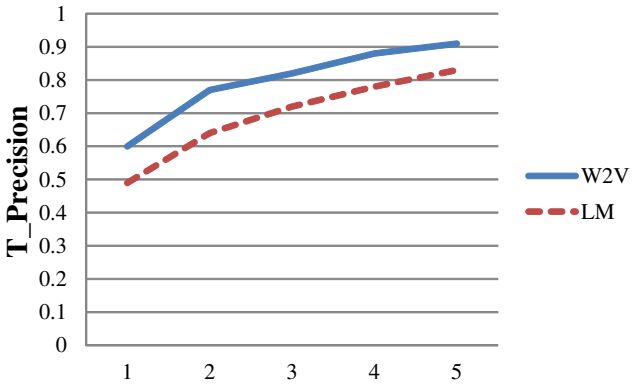


Figure 1: T_Precision curve for LM and W2V against top n

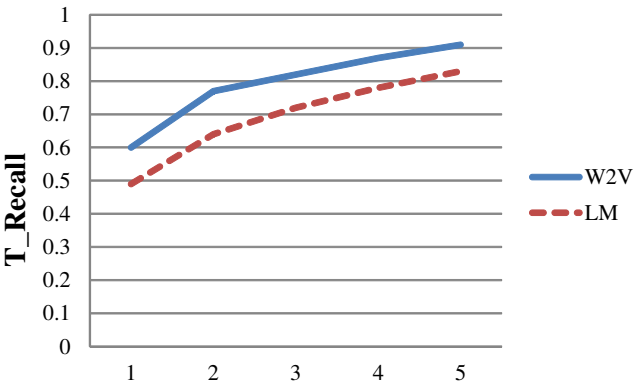


Figure 2: T_Recall curve for LM and W2V against top n

4. Performance of Baseline Methods

There have been several attempts in lexical simplification for selecting the correct candidate using statistical and machine learning methods such as Support Vector Machines (Horn et al., 2014), neural networks (Paetzold and Specia, 2017), and computation on word vectors (Glavas and Stajner, 2015). Also, we proposed (Takada et al., 2017) an approach using the collocation scores of target and candidate words. CEFR-LS is useful for evaluating these methodologies proposed in previous research.

As a preliminary attempt, this study uses two basic systems for selecting correct candidates to show the usefulness and validity of CEFR-LS[9]. One method is based on Horn et al. (2014) which uses a language model approach for candidate selection (hereafter LM). In this study, we constructed a language model using Google N-gram[10], and ranked candidates according to the language model probabilities of a sentence replaced a target with a candidate. The other method employed Word2Vec (Mikolov et al., 2013a; Mikolov et al., 2013b) and calculated the cosine similarity of candidates' vectors against targets' vectors (hereafter W2V), and candidates were arranged in order of similarity scores.

Precision and Recall were calculated with regard to the top n words in each model. Since finding one correct candidate for a target satisfies our purpose and the number of correct answers varies among targets, we adjusted these metrics to evaluate if at least a correct candidate is contained in the top n. We tentatively call them target-based precision and recall, represented as T_Precision and T_Recall in the paper, which are formally defined as follows.

$$\text{T_Precision} = \frac{C^{\text{tar}}_{\text{correct}}}{C^{\text{tar}}_{\text{out}}},$$

$$\text{T_Recall} = \frac{C^{\text{tar}}_{\text{correct}}}{C^{\text{tar}}},$$

where $C^{\text{tar}}_{\text{correct}}$ is the number of targets that are assigned at least a correct candidate, $C^{\text{tar}}_{\text{out}}$ is the number of targets that are assigned any candidates, C^{tar} is the number of all targets. Figure 1 shows the T_Precision curve and Figure 2 shows the T_Recall curve when varying the value of n. It turned out that LM and W2V have about 80% and 90% of T_Precision within the top five candidates respectively, which means a large portion of the targets that are assigned any candidates of CEFR-LS have at least one correct candidate in the top five words. However, it can be said that W2V outperforms LM in that it finds a correct answer more quickly (60% T_Precision for the top-ranked word).

On the other hand, LM has about 80% and W2V has about 90% T_Recall within the top five candidates. This result shows that 80% and 90% of the targets are assigned at least one correct candidate in the top five candidates, respectively.

There were 14 targets where both LM and W2V could not provide a correct paraphrase within the top 5 candidates. Among them are nine types of words including the following example:

(3) Division and specialization of labor only work when individuals can purchase what they do not produce in markets.
Target: division (B2)
Candidate: distribution (B1)

It is beyond the scope of the present paper to build a better algorithm for finding candidates, but we believe CEFR-LS would be a useful resource for this purpose.

5. Conclusion and Future Work

This study has shown that CEFR-LS is an education-oriented dataset for lexical simplification. It specifies CEFR levels for both target and candidate words, which makes the dataset consistent and especially relevant for language education. Also, candidates were selected using a thesaurus, making the simplification task feasible. The performance of two baseline methods on the CEFR-LS also provides a reference for future users of the dataset.

The current version of CEFR-LS is mainly intended as a pilot for constructing a language resource. Therefore, there remains room for future development. Clearly, one important task will be to increase the number of target words. It is planned to include $1,000$ targets in the second release of CEFR-LS. Also, judgments by several annotators should

[9]Stanford CoreNLP (Manning et al., 2014) was used for part-of-speech tagging and the results were matched with the wordlist for CEFR levels.

[10]LDC catalog number: LDC2006T13

be added for each candidate. Another possible extension is to include phrases in the dataset (Ganitkevitch et al., 2013; Pavlick and Callison-Burch, 2016). The current dataset is available at our website[11], which will be updated when the 2nd version is ready.

Acknowledgements

This research has been supported by the KDDI Foundation. We thank Professor Christopher G. Haswell for his contribution on annotation as well as valuable comments and discussions with us.

Bibliographical References

Biran, O., Brody, S., and Elhadad, N. (2011). Putting it Simply: A Context-Aware Approach to Lexical Simplification. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 496–501, June.

Coster, W. and Kauchak, D. (2011a). Learning to Simplify Sentences Using Wikipedia. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1–9, June.

Coster, W. and Kauchak, D. (2011b). Simple English Wikipedia: A New Text Simplification Task. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 665–669, June.

Ganitkevitch, J., Durme, B. V., and Callison-Burch, C. (2013). PPDB: The Paraphrase Database. In *Proc. of Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 758–764, June.

Glavas, G. and Stajner, S. (2015). Simplifying Lexical Simplification: Do We Need Simplified Corpora? In *Proc. of Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP)*, pages 63–68, July.

Horn, C., Manduca, C., and Kauchak, D. (2014). Learning a Lexical Simplifier Using Wikipedia. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 458–463, June.

Laufer, B. (1989). What Percentage of Text-Lexis is Essential for Comprehension? In *Chapter 25 in Special language: From humans thinking to thinking machines*, pages 316–323. Multilingual Maters.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 55–60, June.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. In *Proc. of International Conference on Learning Representations (ICLR)*, May.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. In *Proc. of Annual Conference on Neural Information Processing Systems (NIPS)*, pages 3111–3119, December.

Paetzold, G. H. and Specia, L. (2017). Lexical Simplification with Neural Ranking. In *Proc. of Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 34–40, April.

Pavlick, E. and Callison-Burch, C. (2016). Simple PPDB: A Paraphrase Database for Simplification. In *Proc. of Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 143–148, August.

Takada, S., Arase, Y., and Uchida, S. (2017). Lexical Simplification for English Education: A Collocation-based Approach. In *Proc. of Annual Meeting of the Association for Natural Language Processing (ANLP)*, pages 939–942, March (in Japanese).

Language Resource References

Cambridge University Press. (2015). *English Vocabulary Profile*. `http://www.englishprofile.org/wordlists`.

Yukio Tono. (2016). *CEFR-J Wordlist Version* 1.3. Tokyo University of Foreign Studies, `http://www.cefr-j.org/download.html`.

[11] `http://www-bigdata.ist.osaka-u.ac.jp/arase/pj/lex-simplification.zip`

CLARIN: Towards FAIR and Responsible Data Science Using Language Resources

Franciska de Jong[1], Bente Maegaard[2], Koenraad De Smedt[3], Darja Fišer[4], Dieter Van Uytvanck[5]
[1,5]CLARIN ERIC, [2]University of Copenhagen, [3]University of Bergen, [4]University of Ljubljana and Jožef Stefan Institute
[1,5]Utrecht, The Netherlands, [2]Copenhagen, Denmark, [3]Bergen, Norway, [4]Ljubljana, Slovenia
[1]f.m.g.dejong@uu.nl, [2]bmaegaard@hum.ku.dk, [3]desmedt@uib.no, [4]darja.fiser@ff.uni-lj.si, [5]dieter@clarin.eu

Abstract

CLARIN is a European Research Infrastructure providing access to language resources and technologies for researchers in the humanities and social sciences. It supports the study of language data in general and aims to increase the potential for comparative research of cultural and societal phenomena across the boundaries of languages. This paper outlines the CLARIN vision and strategy, and it explains how the design and implementation of CLARIN are compliant with the FAIR principles: findability, accessibility, interoperability and re-usability of data. The paper also explains the approach of CLARIN towards the enabling of responsible data science. Attention is paid to (i) the development of measures for increasing the transparency and explainability of the results from applying CLARIN technologies, in particular in the context of multidisciplinary research, and (ii) stimulating the uptake of its resources, tools and services by the various communities of use, all in accordance with the principles for Open Science.

Keywords: CLARIN, FAIR, research infrastructure, Open Science policies, multilinguality, digital humanities, responsible data science

1. Introduction: CLARIN Vision and Strategy

CLARIN[1] is a European Research Infrastructure providing access to language resources and tools. It focuses on the widely acknowledged role of language as social and cultural data and the increased potential for comparative research of cultural and societal phenomena across the boundaries of languages. Access to language data is crucial for scholars in the social sciences and the (digital) humanities (De Smedt et al., 2018). Language is a carrier of cultural content and information, both synchronically and diachronically. Language also plays a role as the reflection of scientific and societal knowledge, as an instrument for human communication, as one of the central components of the identity of individuals, groups, cultures or nations, as an instrument for human cognition and expression, as a formal system, and as historical records in need of documentation and preservation. These insights underline the vision and the mission of CLARIN.

CLARIN vision: All digital language resources and tools from all over Europe and beyond are accessible through a single sign-on on-line environment for the support of researchers in the humanities and social sciences.

CLARIN mission: Create and maintain an infrastructure to support the sharing, use and sustainability of language data and tools for research in the humanities and social sciences.

The vision is obviously very ambitious and the goal will not be fully reached in a foreseeable future, but it is the driving force for those who participate in the consortium and an encouragement for researchers to use and contribute to the infrastructure.

CLARIN has been implemented as a distributed research infrastructure with data centres (nodes) across Europe and beyond. The activities in CLARIN basically take place at two levels. One is the central level: the Board of Directors and the technical, communication, and administrative staff. The other is the national level: in each member country, the national activities have by far the largest volume. Previous CLARIN activities have been described in multiple publications that have been disseminated through many channels. At LREC 2014 an overview of language resources, tools, and services on offer through CLARIN was presented (Hinrichs and Krauwer, 2014). The current paper focuses on the overall vision and in particular the adherence to the principles of FAIR and Responsible Data Science.

1.1. A Bit of History

In the preparatory phase 2008–2011 CLARIN was funded by the European Commission. It was established as a European Research Infrastructure Consortium (ERIC)[2] in 2012; the basic funding comes from the member countries. When it was established in 2012 CLARIN had nine members, and in 2017 it has grown to 19 members and two observers. Additionally, CLARIN has a special agreement with Carnegie Mellon University in the USA. This growth, which is still on-going, shows that the basic idea of CLARIN was well chosen and timely, cf. the map in Figure 1.

An important factor for the success and sustainability of a research infrastructure like CLARIN is its scope and size: since CLARIN is collecting data for languages from all regions and periods that are of interest to the European research era, CLARIN is multilingual. A large part of the collected language resources come from the member countries, so the number of members matters for realizing wide coverage. Additionally, CLARIN is open to registering resources in all languages, and CLARIN is truly multilingual by providing metadata for at least five resources for each of more than 1,500 languages.

CLARIN has established the Virtual Language Observatory (VLO)[3], a registry of Language Resources (LRs) based on the CMDI metadata standard (cf. Section 2.). The VLO contains information about all LRs provided by the member countries, plus information from other registries that

[1] http://www.clarin.eu

[2] According to regulations, the consortium needs to have a majority of European countries, but it is not restricted to Europe.

[3] http://vlo.clarin.eu

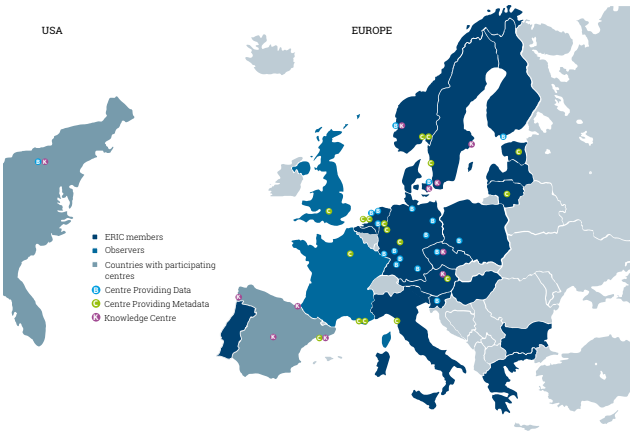


Figure 1: *Map of CLARIN members, observers, and participating centres at the start of 2018.*

want to be present in the VLO. Recently more than 700,000 records have been added from Europeana,[4] the EU digital platform for cultural heritage, which is of particular interest to humanities researchers.[5] With more than 1,600,000 entries in the VLO CLARIN supports access to language resources including cultural heritage in many languages.

1.2. Strategy

After the first 5–6 years, CLARIN is currently shaping its strategy for the next three years, 2018–2020. The strategy builds on the vision given above and further expressed in the Value Proposition[6] (Maegaard et al., 2017). The strategy covers four areas: uptake by researchers, technical infrastructure, knowledge sharing infrastructure and sustainability. [7]

The strategy concerning the latter two points will be addressed in the following subsections. Technical infrastructure will be treated below in Section 2., where also the strong connection to FAIR is described, while responsible data science is addressed in Section 3. and uptake is described in Section 4..

1.2.1. Knowledge Sharing Infrastructure

CLARIN's Knowledge Sharing Infrastructure (KSI) is the necessary 'glue' between the Technical Infrastructure and the researchers. Several instruments are used to share knowledge. Here we mention only a few of them. *Knowledge centres* provide expertise on data and methods to researchers. Knowledge centres may offer expertise and support for certain languages (e.g. the languages of Spain), or a technology or type of data (e.g. treebanks). CLARIN has nine knowledge centres at the time of writing and more are being established. *Workshops* are key to share knowledge and to develop new ideas. Workshops may be used to support new user communities, to make the needs of a particular community visible and to discuss known approaches across borders (country borders or discipline borders). Workshops are normally organised by members and financially supported by CLARIN ERIC.

1.2.2. Sustainability

Membership is an essential part of the sustainability strategy. Countries and international organizations can join the ERIC as members or observers. They contribute to the governance of CLARIN and the membership fees form the core of its business model.[8] The strategy is to maintain the current membership and to grow in Europe and beyond. An important aspect of the strategy is collaboration across borders: CLARIN aims at cross-institutional and cross-sectorial collaboration, e.g., with the GLAM sector (Galleries, Libraries, Archives, Museums) and with the industry. Collaboration with other research infrastructures, be it in the humanities and social sciences area or with eScience, is also pursued in order to foster multidisciplinarity and the inherent need for innovation of methodological frameworks. Through cross-border collaboration, as well as through the focus on training and education, on centres of expertise etc. CLARIN will increase its societal impact and will contribute to the development of methodologies for measuring impact of research infrastructures in the social sciences and humanities. Finally, an inherent goal of all these activities is to integrate and contribute to Europe's Open Science policies (De Smedt et al., 2018).

2. CLARIN and FAIR

With the wide emergence of digitally available language data (be it digitally native or retrodigitized), the possibilities beyond the mere archiving and viewing of such data sets have grown significantly. The idea of transitioning towards an ecosystem of registered, updatable and actionable data has been proposed in the context of the so-called Live Archives (Trilsbeek et al., 2008). These insights were guiding foundations during the establishment of the CLARIN infrastructure (Broeder et al., 2008).

More recently, similar principles have been put forward in the broader data science community, specifically in the context of the Research Data Alliance's Data Fabric working group.[9] The most visible embodiment of these principles are the so-called FAIR principles (Wilkinson et al., 2016) that are now also widely promoted as part of the Open Science paradigm. In the paragraphs below we describe the data architecture of CLARIN in a nutshell and demonstrate afterwards how it is largely FAIR-compliant.

2.1. CLARIN Data Architecture

As a distributed infrastructure, CLARIN exists of a network of technical centres (nodes). A CLARIN centre typically provides a data repository where language resources are stored and made available for researchers. Additionally, many centres also provide tools (web applications, web services or stand-alone applications) to process language data.

[4] https://pro.europeana.eu/

[5] https://www.clarin.eu/blog/bridging-europeana-and-clarin-infrastructures

[6] https://www.clarin.eu/value-proposition

[7] https://www.clarin.eu/content/vision-and-strategy

[8] See (OECD, 2017) on business models for research data infrastructures

[9] https://www.rd-alliance.org/group/data-fabric-ig.html

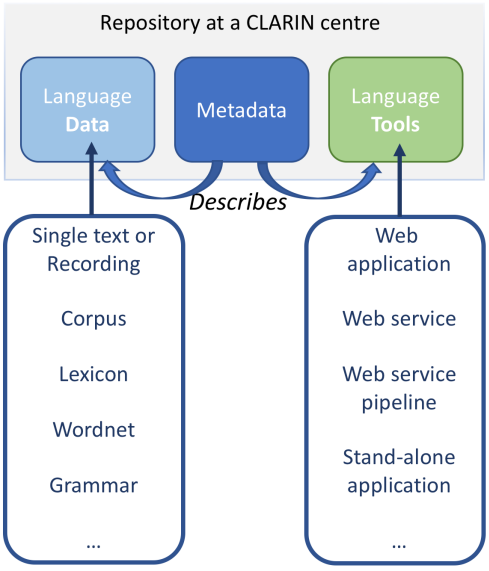


Figure 2: *Schematic overview of what CLARIN technical centres have on offer.*

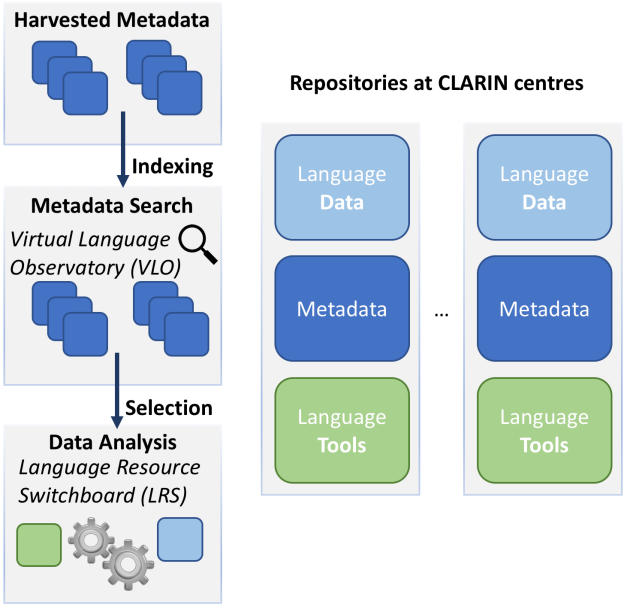


Figure 3: *The metadata for both data and tools from CLARIN repositories are harvested and indexed by the Virtual Language Observatory. Selected data can then be processed with relevant CLARIN data analysis tools.*

Figure 2 is a schematic description of what type of content a repository incorporates. Figure 3 illustrates how metadata from CLARIN repositories are harvested in order to make items searchable and how described datasets can be processed with CLARIN tools.

There are two important principles behind the establishment of CLARIN centres. The first one is the organic nature: CLARIN centres typically evolve from hotspots of know-how, e.g. academic research groups or language data archives. Such fields of specialization could be based on a specific language (e.g. Hungarian), a modality (e.g. speech data), data types (e.g. lexicons) or a processing paradigm (e.g. rule-based morphological analysis). Secondly, there are strict interoperability requirements, in particular for the technical centres. As free as the process of establishing a CLARIN centre can be, a reasonable degree of integration and compatibility is possible only if at the basis there is a clear technical and organisational framework. This is why each CLARIN centre is periodically assessed on various levels, to check if it is following the technical guidelines and if there is a sound organisational backing for the repository. One should note that these requirements only pertain to the protocol level (e.g., OAI-PMH for metadata exchange), leaving complete freedom to the individual centre to choose its preferred software stack.

Without going into details,[10] the main pillars for interoperability with CLARIN are based on HTTP-accessible data and metadata, persistent identifiers and single sign-on whenever resources are not openly available. In the subsequent sections we will relate these pillars to the FAIR principles.

[10]The full requirements are available at `http://hdl.handle.net/11372/DOC-78`.

2.2. Findable

Use and re-use of language resources is only possible when they can be easily found by researchers. FAIR therefore requires persistent identifiers, rich metadata which are indexed and searchable and links from the metadata to the data identifiers. As a metadata standard, CLARIN requires CMDI (Goosen et al., 2015). *Handles* are used to refer to all metadata records (and to data where this makes sense).

In addition, the Virtual Language Observatory (VLO) (Van Uytvanck et al., 2012) provides a powerful search portal for all the collected metadata records, probably the largest catalogue of its kind for language data. Figure 3 depicts how the VLO architecture supports the discovery of data and tools. One can search through all metadata descriptions to find a relevant data set. The VLO offers free text search as well as faceted search. An example is given in Figure 4.

Making a searchable catalogue for over a million resources is obviously a daunting challenge. Usability tests have been performed and have triggered a series of improvements to the user interface. Also, issues with harvested metadata are always to be expected; these are being addressed through continuous adaptation and curation of the metadata.

2.3. Accessible

Once language data are found, they should be easy to access. FAIR translates this requirement into the need for a standardized communication protocol, with the option for easy authentication and authorisation if needed. Metadata should remain accessible, even when the data are no longer available. CLARIN relies on the widely used HTTP protocol and SAML for federated single-sign-on for its implementation of these policies.

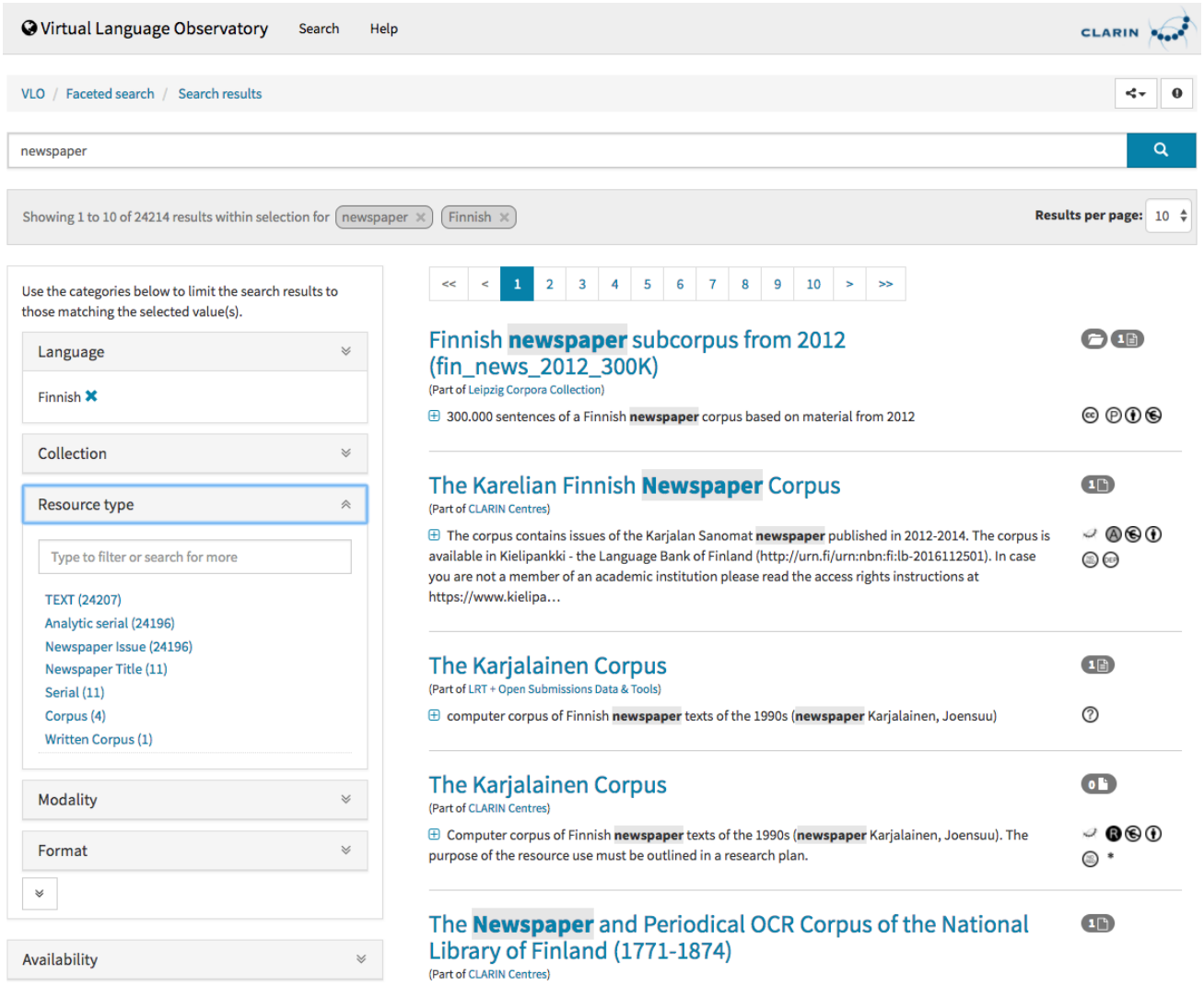


Figure 4: *Screenshot of a VLO search for the keyword "newspaper", restricting the "language" facet to "Finnish". Further restrictions can be imposed with other facets, such as "Resource type" for which a menu has been opened.*

2.4. Interoperable

To attain interoperability, FAIR demands a formal, shared and broadly applicable language for knowledge representation, using FAIR vocabularies and links between metadata and data. CLARIN relies on the CMDI framework as common metadata language, including links to standardized OpenSKOS[11] vocabularies and standardized ways of linking to datasets and landing pages. There are also recommendations for the use of standard data formats.

2.5. Re-usable

Researchers need information on how and under which conditions they can re-use existing resources. FAIR states that re-usable data require clear license and provenance information and adherence to community standards. CLARIN has clear recommendations on license disclosing and user-friendly ways of categorizing these (Arppe et al., 2011). Provenance is an important part of the metadata in CLARIN. While community standards are hard to define, the bottom-up structure of the centres definitely brings along close ties with such good practices.

3. CLARIN and Responsible Data Science

In order to serve the research community, the adherence to the principles of Open Science not only demands clarity about the data services offered, but also about what is to be expected from the analysis tools offered. In particular researchers that do know how to couple the data they aim to use but that have not developed tools themselves, are in need of guidance and explanation of the various aspects of quality. Given that CLARIN offers support for multiple data science paradigms (e.g., rule-based vs. data-driven), performance expectations need to be carefully managed.

3.1. FACT

Across almost all domains of research there is growing concern for how data, especially 'big data', are put to use. Any use of data which is biased, violates privacy or confidentiality, or lacks transparency, may distort conclusions or break trust relations. These concerns also pertain to language data and the analysis tools integrated in the CLARIN infrastructure.

[11] http://openskos.org

A recent expression of these concerns is by the Responsible Data Science consortium (RDS)[12] which aims to tackle ethical and legal challenges, promote data science techniques, infrastructures and approaches that are responsible in the sense that data and data use should be fair, accurate, confidential and transparent (FACT) (Van der Aalst et al., 2017). These aims complement the FAIR principles, especially in a context where the use of data-driven methods typically applied to larger datasets is on the rise.

CLARIN intends to contribute to responsible data science by the design as well as the governance of its infrastructure and to achieve an appropriate and transparent division of responsibilities between data providers, technical centres, and end users. License agreements, which are established between a data provider and a data centre (repository), regulate the terms under which some well-described data is made available. These terms include an end-user license agreement which, together with the terms of service at the data centre, may place some restrictions and responsibilities on the end user, particularly in the case of privacy concerns. The requirement of provenance data in CLARIN metadata makes data traceable and the use of PIDs makes data citable and their use replicable. Furthermore, CLARIN has started to provide guidance on which tool is recommended for which data through a service called the Language Resource Switchboard (Zinn, 2016).

3.2. From Big Data to Big Conclusions and Decisions?

Additional steps are foreseen which relate to the need to document and explain the performance levels that can be expected from the analysis tools, and thereby of the suitability of certain tools for specific scenarios of use. This is particularly relevant for the uptake of CLARIN functionality in the context of multidisciplinary collaboration where methodological frameworks rooted in multiple scholarly traditions have to be combined. This is not always straightforward, for example because quite diverse approaches exist towards the annotation of resources, ranging from part-of-speech tagging to the conceptual tagging of content.

The step from big data to big conclusions and decisions requires a certain level of transparency from the algorithms applied, since black box applications are not likely to be accepted as the basis for conclusions. Merely focussing on the effectiveness of tools on prediction tasks is only relevant within the narrow field of NLP. Sustainable scientific and societal impact of the tools offered can only be expected if the validity of analysis results can be assessed and explained to the communities that CLARIN aims to serve (Manovich, 2016; Nguyen et al., 2016). Clearly it may require the renewal of the quality evaluation frameworks proposed thus far if the CLARIN contribution to data science is to live up to the emerging standards for responsible data science.

4. Uptake

CLARIN's efforts towards uptake reflect and address the vision with two strands of core activities. First, a series of surveys have been designed to evaluate the comprehensiveness and usability of CLARIN services and prioritise future development efforts. In parallel, training models have been developed that stimulate the uptake of CLARIN resources, tools and services by researchers from a wide range of disciplines in the humanities and social sciences where in addition to the teaching of how the technologies and services work, an important goal is also to stimulate methodological and paradigm shifts towards quantitative approaches, interdisciplinary research design, open science policies and transnational collaboration.

4.1. Surveys

4.1.1. User Base

Language in textual, spoken and multimodal form is a rich source of social and cultural data. CLARIN's language resources are therefore highly relevant for a wide range of disciplines and have a big potential for their reuse and repurposing. Especially the combination of multiple resources and the analytic tools available for multiple languages make CLARIN an enabler of comparative studies across regions, periods and languages. In addition to supporting research excellence, CLARIN is therefore also an infrastructure that is likely to have societal impact along various dimensions. To be able to cater for the various scenarios of use successfully, we are monitoring user experience with the infrastructure and identifying obstacles that still need to be addressed through focus groups.

Apart from continued efforts to support existing users, systematic analyses of relevant active research networks are being conducted to increase user engagement, uptake and dissemination for CLARIN.

4.1.2. Resource Tracking

Use and re-use of language resources is only possible when they can be easily found and understood by researchers. The most prominent types of resources that are offered by most consortia, such as corpora of parliamentary records, newspaper corpora, social media corpora, parallel corpora and oral history data that have a big potential for interdisciplinary and comparative research are therefore evaluated in terms of findability through the VLO, comprehensiveness of the metadata records, and usability in concordancing or text analytics environments. While we have observed a good coverage of the surveyed resources in the CLARIN infrastructure overall, there is still room for improvement. The level of inclusion into the CLARIN infrastructure differs across the data types we have analysed (i.e., while many corpora have been added to national repositories, they still cannot be identified through VLO directly due to lacking, idiosyncratic or vernacular names, keywords or description fields. Similarly, uneven granularity of the deposited resources, which ranges from complete archives to single-file records, makes navigation and use of the resources more difficult). The second group of major issues is the incomplete documentation (metadata). While information on corpus size is available for 67 (83%) of the identified corpora, information on the time period covered by the data, linguistic annotation and license information is less readily available. Of all the four resource types we have

[12] http://www.responsibledatascience.org

analysed so far, the metadata on the parliament corpora are generally of the best quality (i.e., the only information that was lacking was related to annotation in the case of two out of the 11 corpora in the infrastructure). By contrast, the metadata on the parallel corpora is much poorer: information on the level of textual alignment is available for 43 corpora (52%). For more information on the overviews of the CLARIN key resource 'families' see (Fišer et al., 2018).

4.2. Training

It is of strategic importance for CLARIN to offer training opportunities for researchers at all stages and from various backgrounds. Live events and on-line training materials not only demonstrate the functionalities of the available tools but also showcase projects that have successfully used the CLARIN infrastructure, address issues related digitally-enhanced and data-driven research design, contribute towards the advancement of the methodological apparatus and, last but not least, actively promote cross-disciplinary and cross-border collaboration. Co-operation with related research infrastructures is part of the strategy for capacity training. The wider aim is to ensure that the knowledge and skills needed to use and maintain the infrastructure are in line with the dynamics in the overall landscape of infrastructures for research.

5. Concluding Remarks

CLARIN was first conceived more than 10 years ago and has been steadily expanding, both in terms of members and in terms of disciplines for which services can be offered. We have outlined the CLARIN strategy and have shown how from the outset CLARIN has been in line with the more recently formulated principles for FAIR data. Plans for being compliant with the Responsible Data Science framework are being developed, as well as efforts to reinforce the multidisciplinary potential and uptake of CLARIN.

Acknowledgements

The work reported in this paper has been supported by the member countries and observers in the CLARIN ERIC, and it has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 676529 for project CLARIN-PLUS, and under grant agreement No. 654119 for project PARTHENOS. The authors owe a special merci to Maria Eskevich for her insights and practical support during the writing of this paper.

Bibliographical References

Arppe, A., Bruun, S., Koskenniemi, K., Lindén, K., Oksanen, V., and Westerlund, H. (2011). A report including Model Licensing Templates and Authorization and Authentication Scheme. Technical Report D7S-2.1, CLARIN ERIC, Utrecht, The Netherlands.

Broeder, D., Nathan, D., Strömqvist, S., and Van Veenendaal, R. (2008). Building a federation of language resource repositories: the DAM-LR Project and its continuation within CLARIN. In *Sixth International Conference on Language Resources and Evaluation (LREC 2008)*. ELRA.

De Smedt, K., de Jong, F., Maegaard, B., Fišer, D., and Van Uytvanck, D. (2018). Towards an Open Science Infrastructure for the Digital Humanities: The Case of CLARIN. In *Proceedings of the Third International Conference on Digital Humanities in the Nordic Countries (DHN2018)*, Helsinki, Finland.

Fišer, D., Lenardič, J., and Erjavec, T. (2018). Meet CLARIN's Key Resource Families. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. ELRA.

Goosen, T., Windhouwer, M., Ohren, O., Herold, A., Eckart, T., Ďurčo, M., and Schonefeld, O. (2015). CMDI 1. 2: Improvements in the CLARIN Component Metadata Infrastructure. In *Selected papers from the CLARIN 2014 Conference, October 24–25, Soesterberg, The Netherlands*, pages 36–53. Linköping University Electronic Press.

Hinrichs, E. and Krauwer, S. (2014). The CLARIN Research Infrastructure: Resources and Tools for e-Humanities Scholars. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1525–1531.

Maegaard, B., Van Uytvanck, D., and Krauwer, S. (2017). CLARIN Value Proposition. Public Report CLARIN-CE-2017-1093-P001, CLARIN ERIC, Utrecht, The Netherlands.

Manovich, L. (2016). The science of culture? Social computing, digital humanities and cultural analytics. *Journal of Cultural Analytics*.

Nguyen, D., Doğruöz, A., Rosé, C., and de Jong, F. M. G. (2016). Computational sociolinguistics: A survey. *Computational Linguistics*, 42(3):537–593.

OECD. (2017). Business models for sustainable research data repositories. OECD Science, Technology and Innovation Policy Papers 47.

Trilsbeek, P., Broeder, D., Van Valkenhoef, T., and Wittenburg, P. (2008). A grid of regional language archives. In *Sixth International Conference on Language Resources and Evaluation (LREC 2008)*, pages 1474–1477. ELRA.

Van der Aalst, W. M. P., Bichler, M., and Heinzl, A. (2017). Responsible Data Science. *Business & Information Systems Engineering*, 59(5):311–313.

Van Uytvanck, D., Stehouwer, H., and Lampen, L. (2012). Semantic metadata mapping in practice: the virtual language observatory. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. ELRA.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., et al. (2016). The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, 3(160018).

Zinn, C. (2016). The CLARIN Language Resource Switchboard. In *Abstracts of the CLARIN Annual Conference 2016*, Aix-en-Provence, France.

From 'Solved Problems' to New Challenges: A Report on LDC Activities

Christopher Cieri, Mark Liberman, Stephanie Strassel, Denise DiPersio, Jonathan Wright, Andrea Mazzucchi, James Fiumara
Linguistic Data Consortium
3600 Market Street, Philadelphia, PA. 19104 USA
{ccieri, myl, strassel, dipersio, jdwright, amazzu, jfiumara} @ldc.upenn.edu

Abstract

This paper reports on the activities of the Linguistic Data Consortium, the next in a sequence of such data center reports included in each LREC meeting. This report begins by sketching the changing demands for Language Resources driven by the spread of Human Language Technologies throughout the market. One result of the successful deployment of HLT enabled applications is increased demand in ever more languages. This in turn places pressure on data centers to collaborate and form global networks in order to meet the demand for LRs of increasing complexity and linguistic diversity. The report next summarizes the over 100 Language Resources released since the last report, many of which have been contributed by research groups around the world. It also covers advances in Consortium infrastructure that assure the integrity of published data sets and support future collection and annotation. Finally, it discusses recent and current LR creation activities that lead to new LR publications followed by data related research activities particularly in clinical applications.

Keywords: language resources, human language technology, data center

1. Landscape

Each two years between LREC meetings sees significant changes in the landscape of language resources (LRs) and the research and technology development that rely upon them. Perhaps the biggest change is the spread of Human Language Technologies (HLTs) throughout the commercial market. Where HLTs were previously limited to the laboratory and highly constrained external uses, today every smart phone, GPS, web application, and telephone response system can exploit speech-to-text, dialogue management, text-to-speech and translation technologies and many of them do.

Another significant trend is the continuing diversification of languages addressed. While historically many LRs were built for a few dozen languages with the greatest number of speakers and sociopolitical prominence, the past several years have seen increased investment in a wider range of languages often labelled under-resourced. LRs for under-resourced languages tend to form clusters of the resources needed to create speech-to-text technologies or translation technologies thereby reflecting concepts such as the Basic Language Resource Kits (Krauwer, 2003), the REFLEX Language Kits or the earlier CALLHOME corpora.

In addition to the diversification by language, a wider range of research disciplines are embracing the creation use and sharing of LRs and Human Language Technologies. The European CLARIN and PARTHENOS programs now support a vast number of projects in the humanities and social sciences across twenty-one countries with common research infrastructure, especially HLTs, with additional support in the form of training, travel grants, workshops, help-desks and other outreach.

At the same time, clinical groups and the medical field are beginning to recognize the promise of LRs and HLTs not only in the mining and management of vast stores of text and speech data but also in the diagnosis and tracking of disorders and therapies.

These shifts in emphasis should not, and generally are not, seen as proof that our communities have fully solved traditional problems such as multiword expression extraction (Schone & Jurafsky, 2001), language identification from text (da Silva & Lopes, 2006), speech synthesis (Sproat, 2010, p. 206) or speech recognition (Schwartz et al., 2011, p. 399) even in well-studied English. However, in a world with limited resources, they do suggest that sponsors recognize the growing need to address a wider range of new languages and technologies and the growing power of the commercial sector to improve performance in market products.

2. Data Centers as Global Networks

LDC is a consortium of educational, research and technology development groups in the academic, government and commercial sectors joined by a common need for language resources. Not a store but rather a mutual support organization, the Consortium's members contribute typically in the form of annual fees, though other inputs are possible, and benefit from the many language resources published by the management office hosted at the University of Pennsylvania. The Consortium has always maintained an international focus and its membership has grown to form a global network of LR users and contributors necessary to support the goal of documenting the world's languages. A snapshot of LDC's global network of members and contributors appears in *Figure 1*.



Figure 1: LDC Global Network of select data sources including subcontractors and vendors (red squares), corpus authors (green circles), media providers (purple diamonds), LDC staff collections (gold diamonds), research collaborators (blue stars). Many markers represent multiple collaborators; many markers partially obscured by others.

The business model, established in 1992 by an advisory board of leaders of academic, government and industry research groups, distinguishes the costs of creating a

language resource from the costs of distributing copies of that resource to other users. Creation costs are generally covered by the sponsor or customer initially requiring the data set while distribution costs are amortized across recipients. While other models exist, and LDC accommodates sponsors who prefer to vary from this practice by, for example, subsidizing distribution, the original LDC model persists apparently because it is clear, shares the burden of LR cost and is sustainable beyond the duration of a single project or even the career of a specific researcher. As evidence of its continued effectiveness, LDC has distributed more than 140,000 copies of datasets to over 4,000 organizations in more than 80 countries.

3. Publications

The rate of LR publications have continued to increase over the life of the Consortium. During the first decade of its existence, LDC published an average of 23 data sets per year. That average increased to 33 during the second decade and 41 for the past four years. Cumulatively LDC has published 755 data sets and distributed a much larger number of parts and intermediate versions within sponsored programs and closed evaluations. Figure 2 shows the rate of LDC publications since inception including annual, average and cumulative numbers.

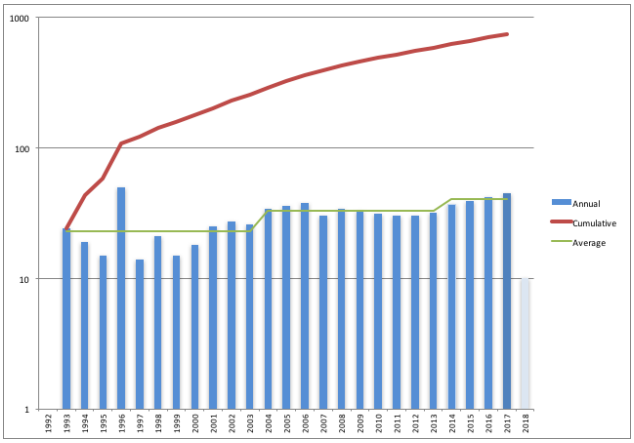


Figure 2: Publication Rate: number of data sets published per year in blue bars, with 10-year average in green and cumulative number of corpora in red. All y-axis values in log scale.

Since 2014, the date of our last report, LDC has released 167 new data sets including 55 corpora that resulted directly from ldc.upenn.edu broadcast conversation with transcripts; Arabic-English and Chinese-English parallel text from broadcast news and broadcast conversation transcripts as well as newswire and web text; word-level alignments of the parallel text and Arabic-English and Chinese-English parallel aligned Treebanks.

Another source of multiple corpora was the IARPA Babel program which produced numerous language packs of which 15 have been released so far via LDC: Assamese, Bengali, Cantonese, Georgian, Haitian, Kurmanji Kurdish, Lao, Pashto, Swahili, Tagalog, Tamil, Tok Pisin, Turkish, Vietnamese and Zulu.

A small sample of additional text publications include: phrase structure Treebanks in Arabic, Chinese, English and Spanish and an Arabic dependency Treebank contributed by the US Army Research Lab; a lexicon of Bambara; informal text (discussion fora, sms, chat) sometimes with translation and word alignment in Arabic and Chinese from the DARPA BOLT program; a range of formal and informal text including translated text annotated for Abstract Meaning Representation; Ancient Chinese text with word-segmentation and part-of-speech tags contributed by Nanjing Normal University; a collection of ~200 annotated samples of student written legal briefs from Georgia Institute of Technology; and a corpus provided by the University of Essex containing >19K words from 40 documents annotated for anaphora by players of the Phrase Detectives "game-with-a-purpose".

Some other examples of speech data include multi-language conversational telephone speech collections in Turkish and three regional clusters: Slavic (Polish, Russian, Ukrainian), South Asian (Bengali, Hindi, Western Panjabi, Tamil, Urdu) and Central Asian (Dari, Persian, Pushto); additional CIEMPISS data from the Universidad Nacional Autónoma de México containing ~18 hours of Mexican Spanish broadcast and transcripts to enable the building of acoustic models for ASR; the KSUEmotions from King Saud University containing five hours of emotional Modern Standard Arabic speech from 23 subjects from Yemen, Saudi Arabia and Syria; Florida Institute of Technology's Noisy TIMIT corpus which reproduces the original TIMIT adding 5-50dB of various noises (white, pink, blue, red, violet and babble); the SRI-FRTIV corpus of ~232 hours of English speech from 34 speakers produced at low, medium and high effort levels in interview, conversation, reading and oration styles to support text-independent speaker verification; and UCLA's collection of audio recording from 9 subjects with time aligned high-speed laryngeal video recordings.

LDC also published a number of data sets to support common task evaluations including the ASpIRE Challenge's data including telephone conversations and speech from far-field microphones in noisy, reverberant rooms; the CHiME Challenge data seeking to develop distant-microphone ASR in real-world environments and the 2010 NIST Speaker Recognition Evaluation Test Set; and the 2007 CoNLL Shared Task data in Arabic, Basque, Catalan, Czech, English, Greek, Hungarian, Italian & Turkish as well as the 2015-2016 Shared Task. The LDC Catalog[1] provides a complete inventory of all publications with descriptions, corpus documentation and samples.

4. LR Creation Projects

Although the Consortium has supported sponsored projects since its founding, LDC's role has expanded from archiving and distributing LRs created by other organizations to collecting and annotating data to create new resources, developing tools and best practices, and

[1] https://catalog.ldc.upenn.edu

increasingly to managing complex data creation teams consisting of multiple partners, subcontractors, vendors and employees with highly specific and complementary roles. Within these teams, LDC crowd-sources or otherwise outsources a growing range of well-defined annotation tasks focusing internal staff on new experimental and high risk collection and annotation paradigms, task definition, quality control and project management.

4.1 DEFT: Deep Exploration and Filtering of Text Program

The DARPA DEFT program which "*aims to address remaining capability gaps related to inference, causal relationships and anomaly detection*" is in its final months and will have ended by the time this paper is published. Over its life-cycle, the program distinguished itself from the earlier practice of extracting isolated information elements from individual sentences in a single language as it evolved toward whole-document, and then whole-corpus (cross-document and cross-lingual) understanding. Over time the program also increasingly focused on extraction of information about events and Sentiment/ Emotive/ Cognitive state (SEC), in addition to entities and relations. To support DEFT, LDC annotated news text and discussion fora for Entities, Relations and Events (ERE), Abstract Meaning Representation (AMR), Textual Entailment and Committed Belief. LRs created for the program are beginning to appear in the LDC Catalog including the DEFT Narrative Text Data which includes ~750,000 words of news text providing the source data for proxy reports based upon single or multiple documents. Proxy reports indicate the date, country and topic of the source text. The body of the report imitates the form of an analyst report.

4.2 Conflicting Accounts of Current Events (CACE)

CACE is new work sponsored by DARPA whose goal is to create an annotated multimedia corpus containing multiple accounts of the same current events in formal and informal data, covering multiple media types and genres including news, blogs, discussion forums, microblogs (Twitter), video, image, speech and other data sources as appropriate. The collected data are centered around a number of topics within a single scenario (e.g. Ukrainian-Russian Relations). Each topic is supported by a topic model that specifies the salient entities and events associated with that topic. A subset of the collected data for each topic is labeled for several features including salient entities as well as salient events and their arguments and attributes (i.e. slots). Additional annotation includes lightweight SEC (Sentiment/ Emotion/ Cognitive State) labeling and/or judgments regarding entailment and contradiction relationships among the various "information elements" that comprise a structured representation of a given event.

4.3 LORELEI

The DARPA Low Resource Languages for Emergent Incidents (LORELEI) program seeks to dramatically advance the state of computational linguistics and human language technology to enable rapid, low-cost development of capabilities for low-resource languages. Acknowledging that even with perfect translation, emergent events such as humanitarian assistance, disaster relief, peacekeeping or infectious disease response generate too much material for analysts to use effectively, LORELEI looks beyond machine translation to provide situational awareness by identifying information elements such as topics, names, events, sentiment and relationships in multilingual sources. LORELEI technology will be applicable to any incident in which a sudden need emerges for assimilation of information about a region of the world where low-resource languages are frequently used in formal and/or informal media. To date technology evaluations have involved Machine Translation, Named Entity Recognition, Entity Discovery and Linking, in which sites are required to link entities to an external knowledge base, and Situation Frame which extracts information from a streaming corpus in the incident language about an emerging situation, to build situational awareness: What's happening, when and where; who is involved; what are the needs and how urgent are they; what are the reactions/responses to the incident among the people involved.

To support the LORELEI Program, LDC is collecting and building a variety of linguistic resources including text in multiple formal and informal genres for several *representative* languages (Hausa, Turkish, Amharic, Arabic, Somali, Farsi, Russian, Spanish, Hungarian, Mandarin, Vietnamese, Yoruba, Tamil, Bengali, Hindi, Indonesian, Tagalog, Thai, Akan (Twi), Swahili, Wolof, Zulu and Uzbek) as well as several low resource *incident* languages whose identity is not disclosed until technology evaluations begin. Large volumes of monolingual and parallel (with English) text are harvested, and smaller volumes of parallel text are acquired via crowdsourcing or traditional translation. Portions of the non-English data that have been translated are labeled for simple named entity, full entity plus coreference, NP chunking, simple semantic annotation lightweight SEC, entity linking and situation frames. Each language includes a 10,000-lemma lexicon and a grammatical sketch, plus basic processing tools including tokenizers, encoding converters, segmenters and entity taggers. Incident languages involve smaller amounts of found data plus (pointers to) formal resources like dictionaries, grammars, gazetteers and primers. The Year 1 surprise language was Uyghur; Year 2 languages were Tigrinya and Oromo.

4.4 Support for NIST Technology Evaluations

In addition to its support for large multi-site technology development programs, LDC also regularly provides LRs to support multiple technology evaluation campaigns organized by the US National Institute of Standards and Technologies.

TAC KBP is the open evaluation series that DEFT performers participate in, along with many non-DEFT sites. As in prior years, LDC has provided data and assessment for all six TAC KBP tracks including the end-to-end Cold Start task, which builds a knowledge base from scratch along with evaluations with EDL, Slot Filling Events and Belief/Sentiment.

NIST has organized an SRE evaluation every year or two since 1996 with evaluations tackling increasingly difficult challenges in channel, room acoustics, language, vocal effort and interaction type. LDC managed data collection in 2015 for the 2016 evaluation under the new CallMyNet protocol. The corpus for the 2016 campaign contains telephone conversations from a total of 220 speakers each of whom completed 10 calls under a variety of noise conditions in one of 4 languages: Tagalog, Cantonese, Mandarin and Cebuano. Calls were manually audited for language, speaker identity and overall quality. LDC has begun collection for the next evaluation, details of which are confidential until the evaluation is complete.

NIST organized the first Language Recognition evaluation in 1996. Since 2003 NIST had held an LRE roughly every other year. LDC has often provided corpora to support LREs including the 2017 evaluation which takes place in September. The latest corpus includes data previously sequestered from the LDC Multi-Language Speech Collection described at a previous LREC (Jones et al., 2016) and containing conversation in two or more linguistic varieties of each of six language clusters: Arabic, Spanish, English, Chinese, Slavic and French. This continues the trend in the NIST LREs to shift from simple language recognition toward the more challenging task of distinguishing highly similar languages and mutually intelligible regional dialects. New data in the 2017 campaign will include speech excerpts extracted from video data.

5. Technical Infrastructure

Although demand for exotic hardware systems has decreased over time, LDC continues to maintain and use its specialized systems for collecting broadcast, telephone calls and messaging as well as its microphone library which addresses a broad range of recording conditions. The past few years have brought increasing demand to miniaturize speech collection systems and deploy them or guide others who deploy them using standard technology in locations around the world.

System infrastructure has evolved to provide ever greater redundancy, data integrity and disaster recovery. Specifically, LDC deploys ZFS as the filesystem of choice to implement continual checks against data corruption. We also distinguish dynamic from static data and internal data from copies stored in the DMZ for public access with appropriate write protection and backup frequency. All storage tiers include offsite storage for disaster recovery.

Our web based annotation infrastructure, WebAnn, first developed in 2011, has since been used to capture tens of millions of annotations. WebAnn is a single application that presents different tools to the user by reusing fixed components, granting much control to the manager of the annotation task. The application has continued to mature in its ability to allow managers to control their work, from a redesigned layout manager for tool widgets, to a more sophisticated assignment creation feature that tracks the managers' input and reports back on failures. In addition, a new version of the application is under development via the NIEUW project discussed below in section 5.1 that will address key improvements like portability and the ability to run offline.

Our infrastructure has also continued to expand the range of Human Language Technologies we integrate into the data creation pipeline, including those developed at LDC or by partners including Phonexia and Oxford Wave Research. Even well designed command line tools can be difficult to take advantage of in the context of a software group with its own code base and data pipelines. One recent addition has been the incorporation of third party tokenizers into an existing text pipeline which is not trivial as most tokenizers modify the input stream, and our internal processes typically maintain standoff annotation. An even more complex case was taking a recipe for creating a forced aligner for a new language and creating a turn crank procedure.

5.1 NIEUW

For the NIEUW (Novel Incentives and Workflows) project, LDC is building infrastructure and tools to increase the volume and variety of LRs through the use of novel incentives and crowdsourcing. Inspired by the successes of social media, citizen science and games with a purpose, this infrastructure will enable the creation of scalable data collection and annotation activities designed with appropriate incentives and available to the public via the web and mobile devices. We have identified three partially overlapping audiences as the most promising potential contributors: Language Professionals and Students, Citizen Scientists, and Game Players. In order to appeal to these contributor groups and their respective motivations, LDC is creating three portals with varying designs and incentive models. Data collection and annotation activities will be initially created by LDC, however, we will make toolkits available to allow collaborators to create their own activities which can be hosted on the portals.

The infrastructure for NIEUW will build upon LDC's WebAnn described above. By rewriting and enhancing WebAnn, the NIEUW project will create an open source, portable package that supports a wide range of collection and annotation activities with improved ease of use for creating activities, defining workflows, reporting progress, evaluating contributions, and extracting stable data.

6. Research

The Consortium has conducted basic research in addition to, and often integrated with, its data creation activities since inception. Here we distinguish '*research*' activities whose purpose is principally to investigate and report on some linguistic phenomenon but which often produce datasets as a by-product from '*data creation*' activities whose principal goal is to develop language resource which often require research and experimentation in the process. With that distinction in mind, research activities have increased significantly since the last report, particularly in the area of clinical data analysis.

6.1 Exploring the 'Space' of Autism Spectrum Disorders

A collaboration between LDC and the Center for Autism Research (CAR) of the Children's Hospital of Philadelphia

began with the goal of identifying linguistic correlates of Autism Spectrum Disorders. CAR had recorded video and audio from many hundreds of diagnostic sessions which included informal conversation and structured linguistic and social activities and for which gold-standard diagnoses were available based on the full range of diagnostic tools. Diagnostic instruments such as the Autism Diagnostic Observation Schedule (ADOS) make reference to multiple linguistic measures that one could imagine extracting from speech and transcripts in an automated fashion for more detailed analysis. Early results showed that relatively straightforward classifiers based on relatively simple linguistic features could predict the diagnoses with good accuracy (Parish-Morris et al., 2017). The corpus used for this research is now undergoing final review before release via LDC.

While encouraging, these results did not truly exercise the power of automated approaches since they relied upon a highly structured diagnostic session requiring several hours of time from both the patient and a highly-trained diagnostician with years of experience in eliciting behavior that distinguished ASD from neuro-typical patients. The real challenge of language technologies would be to achieve the same or better accuracy with shorter informal conversations that could be recorded at home or school via the internet or a telephone call. To support such work CAR is now recording both informal and structured sessions from a stratified sample of ASD and neuro-typical children which will also be released through LDC when complete.

6.2 Linguistic Correlates of Neuro-Degenerative Disorders

Following on promising results with autistic patients, LDC has begun a collaboration with the Fronto-Temporal Disorder laboratory of the Hospital of the University of Pennsylvania. The FTD lab provided digital recordings and transcripts of semi-structured interactions involving 32 patients previously diagnosed with bvFTD, behavioral variant frontotemporal dementia, matched to 17 healthy controls. LDC normalized, QCed and time aligned the transcripts and worked with the FTD lab to develop automatic methods to analyze prosody correlated with clinical evaluations. Specifically, we computed fundamental frequency and log-scale pitch range controlling for individual and sex differences and correlating to neuropsychiatric tests and measures of gray matter atrophy. bvFTD patients had significantly reduced f0 range compared to healthy controls (Nevler et al., 2017) reflecting impaired prosody and supporting the feasibility of automated analysis.

7. Bibliographical References

DARPA (2017) Deep Exploration and Filtering of Text, https://www.darpa.mil/program/deep-exploration-and-filtering-of-text, retrieved October 1 2017.

Jones, Karen Stephanie Strassel, Kevin Walker, David Graff, Jonathan Wright (2016) Multi-language Speech Collection for NIST LRE, Proceedings of LREC.

Krauwer, Steven (2003) The Basic Language Resource Kit (BLARK) as the First Milestone for the Language Resources Roadmap, International Workshop on Speech and Computer (SPECOM), October 27-29, Moscow, Russia.

Nevler, Naomi, Sharon, Ash, C. Jester, D. Irwin, Mark Liberman & Murray Grossman (2017) Automatic measurement of prosody in behavioral variant FTD, Neurology: July 19.

Olive, Joseph, Caitlin Christianson and John McCary, eds. (2011) Handbook of Natural Language Processing and Machine Translation. DARPA Global Autonomous Language Exploitation. New York: Springer.

Schone, Patrick and Daniel Jurafsky (2001) Is Knowledge-Free Induction of Multiword Unit Dictionary Headwords a Solved Problem? Proceedings of Empirical Methods in Natural Language Processing, Pittsburgh, PA.

Schwartz, Richard, Joseph Olive, John McCary and Caitlin Christianson (2011) Machine Translation from Speech in Olive, Joseph, Caitlin Christianson and John McCary, eds. (2011).

Da Silva, Ferreira, J. & Pereira Lopes, G. (2006). Identification of Document Language is Not yet a Completely Solved Problem, in Proceedings of the International Conference on Computational Intelligence for Modelling Control and Automation.

Parish-Morris, Julia, Mark Y. Liberman, Christopher Cieri, John D. Herrington, Benjamin E. Yerys, Leila Bateman, Joseph Donaher, Emily Ferguson, Juhi Pandey and Robert T. Schultz (2017) Linguistic camouflage in girls with autism spectrum disorder. Molecular Autism: Brain, Cognition and Behavior 2017 8:48/

Sproat, Richard (2010) Language, Technology, and Society, Oxford, UK: Oxford University Press.

New Directions in ELRA Activities

Valérie Mapelli, Victoria Arranz, Pawel Kamocki, Hélène Mazo, Vladimir Popescu
ELDA/ELRA
9 rue des Cordelières, 75013 Paris, France
Email: {mapelli, arranz, kamocki, mazo, popescu}@elda.org

Abstract

Beyond the generic activities (cataloguing, producing, distribution of Language Resources, dissemination of information, etc.) that make the overall ELRA mission an indispensable middle-man in the field of Language Resources (LRs), new directions of work are now being undertaken so as to answer the needs of this ever-moving community. This impacts the structure and the operating model of the association *per se* with the creation of a new technical committee dealing with Less-Resourced Languages and the modification of the ELRA membership policy. It also intrinsically impacts the axes of work at all steps of activities: offering new tools for sharing LRs and related information, adapting to new legal requirements, producing and offering field-specific data. This paper addresses these new directions and describes ELRA (and its operational body ELDA) regular activities updates. Future activities are also reported in the last part of the article. They consist in ongoing projects like the ELRC initiative, the start of another CEF-funded project, the European Language Resource Infrastructure (ELRI), the updating of the Review of existing Language Resources for languages of France, and the continuation of the ELRA Catalogue development.

Keywords: Less-Resourced languages, ELRA Membership, Cataloguing, Open Data, ISLRN, LR Production, Events

1. Introduction

Beyond the generic activities (cataloguing, producing, dissemination of information, etc.) that make the overall ELRA[1] mission an indispensable middle-man in the field of Language Resources (LRs), new directions of work are now being undertaken so as to answer the needs of this ever-moving community.

This impacts the structure and the operating model of the association *per se* with the creation of a new technical committee dealing with Less-Resourced Languages and the modification of the ELRA membership policy.

It also intrinsically affects the axes of work at all steps of activities: offering new tools for sharing LRs and related information, adapting to new legal requirements, producing and offering field-specific data.

This paper addresses these new directions and describes ELRA (along with its operational body ELDA) regular activities updates.

2. New operating directions at ELRA

2.1. LRL Committee and SIGUL

In October 2016, ELRA launched the ELRA-LRL TC (Less-Resourced Languages Technical Committee). It focuses on different actions to be undertaken to support the maintenance of linguistic diversity through technology and ICT for minority/less-resourced languages. Defining an agenda for LRTs development, allowing a better sharing of corresponding tools and LRs, sharing and spreading state-of-the art knowledge, exploring techniques for evaluation of LRTs, promoting related management plans, are part of the objectives.

In April 2017, with the support of the LRL Committee, ELRA partnered with the ISCA to create a joint Special Interest Group on Under-Resourced Languages (SIGUL)[2] aiming to bring together a number of professionals involved in the development of Language Resources and technologies for under-resourced languages.

Under the auspices of Interspeech 2017 in Stockholm, SIGUL organised its first event at the Special session on Digital Revolution for Under-resourced Languages (DigRev-URL)[3].

On 12th May 2018, in Miyazaki (Japan), in conjunction with LREC 2018, SIGUL will organise the 3rd CCURL Workshop, entitled "Sustaining knowledge diversity in the digital age"[4].

2.2. Enhancing ELRA Membership policy

Driven by the willingness to expand its membership base, the ELRA Board has appointed a Membership Task Force consisting of current and former Board members together with ELDA staff to rethink the current membership model. The Task Force work focused on revising the services proposed to the current institutional members, considering new membership categories and related services, and reassessing the membership fidelity programme.

The final proposal was submitted to the validation of ELRA General Assembly on 20 October 2017 and is being implemented in 2018.

Acknowledging that its membership base needs to be expanded so as to better represent the community, the ELRA Board has decided to bring drastic changes to its membership drive by opening the ELRA membership to individual members and by substituting discounts on membership to miles to reward the loyal institutional members.

The new membership at ELRA now encompasses two (2) types of members:

1. the Institutional Members (formerly called ELRA Members), including the ELRA Subscribers,
2. the Individual Members.

The Individual Members include:

a. the individual researchers and users of LRs who will join the association by paying individual membership fees

b. the employees of institutional ELRA members who are ELRA members by default, without having to pay for the individual membership fees since the institutional membership covers for that.

[1] http://www.elra.info

[2] http://www.elra.info/en/sig/sigul/

[3] http://ahclab.naist.jp/DigRevURL/index.html

[4] http://www.ilc.cnr.it/ccurl2018/

Individual members are represented by a Board member designated among the members and elected by them.
For the Institutional Members, joining ELRA remains the same. The main change lies in the discontinuation of the membership fidelity programme, replaced by a new mechanism rewarding the loyal members. The new membership schema now includes a discount on membership fees on a 3-year basis: the institutional members can cumulate up to 30% discount on the membership fees by renewing their membership for 3 consecutive years or more. Moreover, the fees have been harmonised to attract more subscribers: the EU organisations (above 50 employees) and non-EU organisations (subscribers) now pay the same membership fee/subscription.[5]
The services offered to the ELRA members are also currently being tailored by the Membership task force so as to extend ELRA's offer to the Individual Member statute. Current directions include in particular: LREC reduced fees, access to some Language Resources from the ELRA Catalogue, access to LRE Map and Universal Catalogue restricted services which are currently being defined, monthly members' news.

3. Cataloguing and licensing

3.1. New ELRA Catalogue

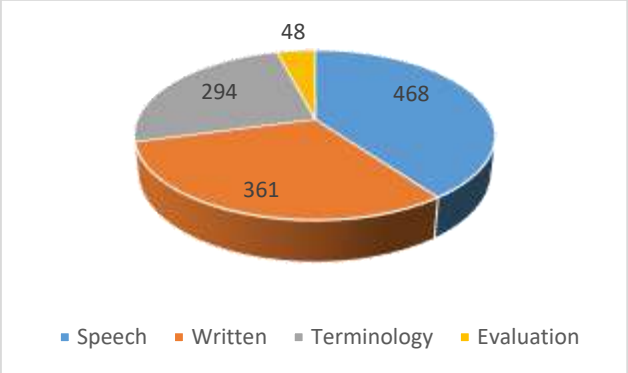
A new step forward has been taken in ELRA's ambition to offer an improved catalogue. In order to initiate the path to e-commerce services, the ELRA Catalogue of Language Resources[6] was completely redesigned, with a new interface and an improved navigation so as to allow visitors an easier access to the over 1070 Language Resources (LRs) and their corresponding descriptions. Among the new features, the Catalogue now offers extended metadata to describe the LRs, automatic submission to ISLRN, a refined search on the catalogue data for finding more specific information using criteria such as language, resource or media type, licence, etc. Currently, LRs can be selected and placed in a cart from where the user can send a request for quotation to initiate the order. When logging in, the user selects LRs and obtains distribution details (licensing information, prices) depending on his/her user status: ELRA member/non-member, Research vs Commercial organisation.
To allow this, e-commerce capabilities have been added[7] and aims to provide an end-to-end e-commerce management (order management, delivery, invoicing and payment facilities). Other functionalities such as an e-licensing module (automatic filling in and electronic signature) will also be developed and integrated at a later stage.

3.2. Clean version of Shared LRs available

Striving for constantly making LRs available to the widest community, ELRA had launched an "Open Challenge", known as "Share your LRs" at LREC 2014. All participants to the conference were proposed to share their LRs by giving access to the resource and corresponding description either through an external URL or by depositing them in a specific LREC repository before the main conference.
The Shared LRs set was manually checked and a cleaned version of the lists of LRs gathered at LREC 2014 and 2016 are available online[8]. These lists include LRs complying with the following criteria:
- LRs are accessible either through direct download or provided via an external URL;
- LRs belong to the following categories: corpus, grammar/language model, ontology, terminology, treebank, evaluation data/package.
These efforts also continue for the LREC 2018 edition.

3.3. Latest LRs in the ELRA Catalogue

Every year, new LRs are becoming available through the ELRA Catalogue. From 1 January 2016 to 31 December 2017, 62 LRs were released.
The distribution per type of resources in the catalogue as of the fourth quarter 2017 is given below:



Graph 1: LRs in the ELRA Catalogue (as of 31/12/2017)

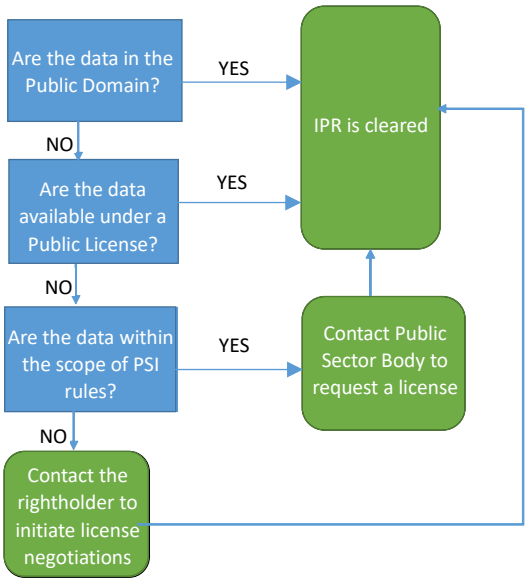
As part of the 62 LRs recently announced, a large set of resources produced within the PEA TRAD project, supported by the French Ministry of Defence and packaged by ELDA was released. They consist of Pashto monolingual and parallel corpora (aligned with French and English), Arabic-French and Arabic-English parallel corpora, Chinese-French and Chinese-English parallel corpora.
Beyond the Pashto language represented through PEA TRAD resources, less-resourced languages are also worth quoting in the following LRs: the Helsinki Corpus of Swahili, GlobalPhone Swahili, a Mongolian written corpus, a Persian Speech Corpus, the FAME! Speech Corpus (dealing with Frisian language).
A history of LRs released in the ELRA Catalogue since 2007 (listed in chronological order) can be viewed online[9].

3.4. Open Data Licensing

With ELDA's involvement in the European Language Resource Coordination (ELRC, see specific section further below), significant advances have been achieved towards a better knowledge of the license issues management workflow for language resources labelled as "open data"

[5] http://www.elra.info/en/join-elra/rewarding-our-faithful-inst-members/
[6] http://catalogue.elra.info
[7] http://www.meta-share.eu
[8] http://lrec2014.lrec-conf.org/en/shared-lrs/current-list-shared-lrs/ and http://lrec2016.lrec-conf.org/en/shared-lrs/
[9] http://www.elra.info/en/catalogues/language-resources-announcements/

and aimed to fuel MT@EC, the Machine Translation platform dedicated to public administrations.
Within the ELRC consortium, all the partners identified, collected and made available language resources indicated as "open" coming from public service administrations. To do so, ELDA drafted a workflow supporting anyone wishing to obtain open data (i.e. data that fall under the scope of the PSI Directive) while following the right steps to assess the rights attached to the data (see figure below). Such a workflow aims to be integrated in the IPR clearance



section of ELRA's Data Management Plan (Choukri et al., 2016).

Figure 1: Workflow for data that fall under the scope of PSI Directive

3.5. ISLRN latest news

As scientific work requires accurate citations of referenced work so as to allow the community to understand the whole context and also replicate the experiments conducted by other researchers, since 2016, LREC endorses the need to uniquely identify LRs through the use of the International Standard Language Resource Number (ISLRN, www.islrn.org), a Persistent Unique Identifier to be assigned to each Language Resource. The assignment of ISLRNs to LRs cited in LREC papers are offered at submission time.
As a follow-up of the LRE Map 2016 initiative, ELRA has processed the information on existing and newly-created Language Resources provided by the authors submitting to LREC 2016 Conference. In order to increase the visibility of these resources, ELRA has allocated an ISLRN to each of the 106 languages resources which distribute as follows: 73 corpora (ca. written, spoken, multimodal), 30 lexicons (including ontologies), 3 evaluation data.
Moreover, in January 2017, the Institute for Applied Linguistics (IULA) at the Universitat Pompeu Fabra (UPF), Spain, became a certified provider to the ISLRN system. This means that IULA can apply for ISLRNs on behalf of the developers of data managed and distributed via the IULA network. IULA has already submitted 107 LRs to the ISLRN. These include monolingual and multilingual written corpora and lexica for the languages spoken in Spain (Aragonese, Asturian, Basque, Castilian Spanish, Catalan, Galician), as well as other European languages (English, French, German, Greek, Italian, Occitan, Portuguese, Romanian), and Esperanto.
The meta-information for these language resources is also available on the ISLRN website[10] with a broad international audience.
As of 31 December 2017, the total number of Language Resources having been allocated an ISLRN persistent identifier amounts to 2583. These LRs include raw and annotated corpora, lexicons and dictionaries, speech resources (conversational, synthesis, etc.), evaluation sets and multimodal resources, and cover 218 distinct languages (including sign languages).
Like in 2016, the ISLRN identifier is part of LRE Map, the LREC submission feature, where the authors can describe the resources(s) cited in their papers or used/developed during their research.

4. Production and infrastructure projects

4.1. Sentiment Annotation in German and French Tweets

Within the framework of two public research projects, uComp, Embedded Human Computation for Knowledge Extraction and Evaluation and Request (funded by the European Union under a CHIST-ERA cooperation action), REcursive QUEry and Scalable Technologies (funded by the French Government under the "Projets Investissements d'Avenir" programme), ELDA has been commissioned by the LIMSI laboratory to work on a deep sentiment and opinion annotation of German and French tweets. The German tweets addressed climate change and hence were rather regular in their structure, and grammatical. On the other hand, the French tweets were expressing virulent stances on transportation issues. Hence, they were much less grammatical and regular than the German tweets. This posed significant challenges to annotators, who nonetheless managed to achieve relatively high average inter-annotator agreements for this kind of projects.
In both annotation tasks the same annotation scheme was used, with really minor modifications to account for the "virulence" aspects of French tweets, hence ELDA has been able to develop an in-house software toolkit based around a LIMSI-provided annotation tool. The toolkit allows users to sample annotations, automatically pre-annotate data, perform various data bookkeeping tasks (tweet origin reconstruction, source file retrieval in samples etc.), and compute different agreement statistics (Cohen's Kappa measures, F-measures, tag histograms, etc.). An important methodological improvement in the French tweet annotation campaign has been obtained by pipelining the annotation process into two stages:
(i) non-transport-related tweet identification (as these tweets are not further annotated);
(ii) annotation of the transport-related tweets. This process allowed for significant gains in terms of annotation time.
Overall, these two annotation campaigns took ELDA more than two years of almost continuous efforts, as we participated in several aspects of the annotation campaigns: annotator recruitment and management, annotation guide revision / update discussions with the LIMSI, data handling

[10] http://www.islrn.org/

tools implementation, annotation validation and quality control.

4.2. Infrastructure projects

4.2.1. MLi

MLi, Towards a Multilingual Data Services infrastructure, has been working to deliver the strategic vision and operational specifications needed for building a sustainable and comprehensive European MultiLingual data and services Infrastructure. The action has ended mid-2016 and ELDA has been strongly involved in investigating and assessing the Language Technologies, including Machine Translation, in the eCommerce sector in the European Union (Fernández-Barrera et al., 2016).
The group of experts behind this action (INMARK, ATOS, DCU, ELDA, ESTEAM, TILDE and USFD) produced a set of recommendations which was acknowledged by the European Commission.

4.2.2. CRACKER

Within the Cracker (Cracking the Language Barrier) H2020 project funded by the European Commission, ELDA participated in updating the technological base of the pan-European META-SHARE resource cataloguing platform. This represented a significant effort, as several years of maintenance and development lag had to be compensated for. Nonetheless, this effort has been successful, as, together with our partners from the ILSP Athena Research Centre in Greece, we have managed to make META-SHARE evolve, so that it can form the basis of other projects, such as the ELRC-SHARE repository, deployed within the ELRC project, described in Section 4.2.3.

4.2.3. European Language Resource Coordination

The 2-year project SMART 2014/1074 European Language Resource Coordination (ELRC)[11], funded by the European Commission within the (Connecting Europe Facility) programme, ended in April 2017 (Lösch et al., 2018).
Targeted data are those produced by the public sector administrations across all 28 EU Member States countries plus Norway and Iceland, which can be made available for re-use within the CEF Automated Translation platform (CEF.AT) and through the EU Open Data portal, with suitable copyright protection.
Together with the other 3 partners of the ELRC consortium (DFKI, TILDE and ILSP), ELDA contributed to the following achievements of this project:
- Collection, validation and delivery of 225 Language Resources that include 138 bi-/multi-lingual corpora, 50 terminologies and 37 mono-lingual corpora,
- Organisation of 29 workshops and 2 ELRC conferences to raise awareness about the importance of data sets and encourage participants to contribute to the collection of LRs needed,
- setting up and running of the Language Resource Board (LRB), as the governance body of ELRC which totals 60 members (Technology and Public Service National Anchor Points),
- setting up of ELRC infrastructure (website, legal and technical helpdesk, ELRC-Share repository).
The project final report is available through the EC website[12].
Two new projects inheriting from ELRC initiative were launched in December 2016 for a 3-year duration: SMART 2015/1091 ELRC+ L2 and ELRC+ L3. ELRC+ L2 focuses on the collection of LRs through awareness efforts via the organisation of a new batch of workshops and conferences. It also provides the technical infrastructure (ELRC-Share repository) to host collected LRs as well as legal and technical support through an online Helpdesk and dedicated services for IPR clearance. ELRC+ L3 aims to implement the acquisition of additional LRs and related refinement/processing services (e.g. anonymization), also necessitating the use of the ELRC-Share repository and IPR clearance services.
In parallel to the collection task of the first project, ELRC partners initiated a production activity by using the ILSP-Focus Crawler tool in order to crawl a number of public websites and obtain relevant aligned documents in different EU languages. To support the crawling and validation process, ELDA produced a crawling management toolkit, the ELDA_CMTK, available on Github[13]. This toolchain is currently being used in the novel resource production tasks within ELRC+ L3.

4.2.4. European Language Resource Infrastructure

ELRI (European Language Resource Infrastructure) is a 24-month action financed under the Connecting Europe Facility (CEF), Telecommunications Sector (CEF-TC-2016-3), and which started in October 2017. The action aims to deploy a network which allows for the accessible, secure and reliable sharing of LRs. ELRI targets LRs which are produced by translation centres and public institutions in Europe, with the partners' Member States (France, Ireland, Portugal and Spain) as starting point of the network and other countries joining in as future extensions. The Consortium is composed of the following institutions: VicomTech (coordinator, Spain), Linkare IT (Portugal), AMA (Portugal), ELDA (France), FCUL (Portugal), SESIAD (Spain) and DCU (Ireland).
The LRs contributed by the aforementioned stakeholders will be rendered available through different levels of accessibility, depending on the providers' restrictions to share their data. As a first step, the ELRI network will deposit the LRs in the national relay stations of the participating Member States. This first level will only grant access to national contributors and it is the option chosen by those adopters who do not wish to share their data further. It allows for a secure way to store and manage their data and get initiated into sharing. The second level of data sharing offered by ELRI will be the ELRC-Share repository. Those data holders willing to share their data at an EC (European Commission) level will have their data included in such repository, which shares its resources with the DGT (Directorate-General for Translation), so as to improve and customise their eTranslation platform. This platform is at the service of the different Digital Service Infrastructures (DSIs), thus benefiting European citizens

[11] http://lr-coordination.eu/
[12] https://ec.europa.eu/digital-single-market/en/news/european-language-resource-coordination-elrc-final-report-2015-2017
[13] https://github.com/ELDAELRA/elda_cmtk

from their usage. Finally, there is a third level of data sharing which implies opening data use to the community at large through repositories such as ELRA, META-SHARE or the Open Data Portal, all this subject to data providers' wishes.
ELRI will address Intellectual Property Rights (IPRs) for all data and metadata exchanged, making sure these are compliant with both the Public Sector Information (PSI) Directive and DSIs regulations.
Last but not least, ELRI will not only set-up the local relay stations and exchange network for data sharing but will also integrate data processing tools to ensure the creation of quality resources from the data contributed by the different stakeholders and will organise dissemination events to address potential data contributors. Moreover, the usage of such tools will be also IPR cleared by ELRI, who will support and advice stakeholders on the data processing procedures, and on sensitive data handling protocols, such as anonymisation.
All this work is done in close collaboration with other currently-ongoing data-collection and stakeholder awareness-raising actions under the CEF programme, such as ELRC+ L2 and ELRC+ L3. All these actions complement each other, avoiding the doubling of efforts.

5. Events and dissemination activities

5.1. LRE Map

During the last two years, several significant developments have been undertaken by ELRA for LRE Map software. First, a new resource submission form page has been added, which allows conference paper authors to submit resource metadata information along with minimal paper references (authors, paper identifier, conference name and year).
Moreover, the application runtime performance has been vastly improved, its underlying data model simplified and streamlined, and the software of the application has been upgraded to state-of-the-art technologies.
Results from previous conferences can be viewed from the LRE Map dedicated website[14].

5.2. LREC

The last edition of LREC (10th) took place at the Grand Hotel Bernardin in Portorož (Slovenia) on the week of 23 to 28 May, 2016.
The conference was held under the Honorary Patronage of His Excellency Mr. Borut Pahor, President of the Republic of Slovenia. The opening message from Mr Andrus Ansip, Vice-President of the European Commission, conveyed by Mr Zoran Stančič, the Head of the EU Representation in Slovenia, gave a positive and strong sign of the EC commitment on the multilingualism issues.
All in all, 1220 participants from 59 countries registered to the conference and the workshops. The HLT Village gathered 10 booths where projects and initiatives could disseminate their activities and results.
The Antonio Zampolli Prize was awarded to Professor Roger K. Moore, Chair of Spoken Language Processing Dept. Computer Science, University of Sheffield, UK.
The 11th edition of LREC will take place on May 7-12, 2018 at the Phoenix Seagaia Resort in Miyazaki (Japan). The conference website[15] provides general information on the conference. Information will also be circulated on the LREC 2018 Twitter account (@LREC2018) using #LREC2018. The contact email for any question on the conference is lrec@lrec-conf.org.

6. Future work

ELRC+ L2, ELRC+ L3 and ELRI projects will be one of ELDA's major focus for the following 2 years, the aim being to provide a sustainable infrastructure to support the acquisition of appropriate and qualitative LRs for key areas of CEF DSIs (Digital Service Infrastructures) so as to contribute to the customization and improvement of the CEF Automatic Translation system. Thus, these projects will work on Language Resource identification, collection, processing and IPR clearing for the shared data, as well as on the dissemination of the needs and activities through different workshops and conferences.
A follow-up of the Review of the existing Language Resources for languages of France is now being undertaken by ELRA under a new funding by the Délégation Générale à la Langue Française et aux Langues de France (Grouas et al., 2016). It will provide an up-to-date list of identified LRs and will initiate negotiations to make those LRs available.
Finally, the step by step implementation of e-commerce and e-licensing modules for the ELRA Catalogue will continue.

7. Bibliographical References

Choukri K., Mapelli V., Mazo H., Popescu V. (2016). ELRA Activities and Services. In Proceedings of LREC'16. Portorož, Slovenia, 2016.

Grouas T., Mapelli V., Samier Q. (2016). Review on the Existing Language Resources for Languages of France. In Proceedings of LREC'16. Portorož, Slovenia, 2016.

Fernández-Barrera M., Popescu V., Toral A., Gaspari F ., Choukri K. (2016). Enhancing Cross-border EU E-commerce through Machine Translation: Needed Language Resources, Challenges and Opportunities. In Proceedings of LREC'16. Portorož, Slovenia, 2016.

Lösch A., Mapelli V., Piperidis S., Vasiļjevs A., Smal L., Declerck T., Schnur E., Choukri K. and Van Genabith J. (2018). European Language Resource Coordination: Collecting Language Resources for Public Sector Multilingual Information Management. In Proceedings of LREC'18. Miyazaki, Japan, 2018.

8. Language Resource References

GlobalPhone Swahili, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 200-331-212-512-8, ELRA ID: ELRA-S0375.

Helsinki Corpus of Swahili, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 941-187-059-145-7, ELRA ID: ELRA-W0119.

[14] http://lremap.elra.info/

[15] http://lrec2018.lrec-conf.org/en

NUM 5M Mongolian written corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 492-817-146-504-9, ELRA ID: ELRA-W0120.

Persian Speech Corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 068-845-898-304-0, ELRA ID: ELRA-S0393.

The FAME! Speech Corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 340-994-352-616-4, ELRA ID: ELRA-S0391.

TRAD Pashto Broadcast News Speech Corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 918-508-885-913-7, ELRA ID: ELRA-S0381.

TRAD Pashto Monolingual text Corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 394-903-293-388-0, ELRA ID: ELRA-W0092.

TRAD Pashto-French Parallel corpus of transcribed Broadcast News Speech - Training data, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 802-643-297-429-4, ELRA ID: ELRA-W0093.

TRAD Pashto-French Parallel corpus of transcribed Broadcast News Speech - Test data, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 547-897-479-723-3, ELRA ID: ELRA-W0094.

TRAD Pashto-English Parallel corpus of transcribed Broadcast News Speech - Test data, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 006-102-605-738-4, ELRA ID: ELRA-W0095.

TRAD Pashto-French News Articles Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 649-628-149-051-7, ELRA ID: ELRA-W0096.

TRAD Pashto-English News Articles Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 612-936-517-010-2, ELRA ID: ELRA-W0097.

TRAD Arabic-French Newspaper Parallel corpus - Test set 1, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 922-732-502-473-8, ELRA ID: ELRA-W0098.

TRAD Arabic-English Newspaper Parallel corpus - Test set 1, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 764-187-795-074-0, ELRA ID: ELRA-W0099.

TRAD Arabic-French Newspaper Parallel corpus - Test set 2, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 722-323-886-920-3, ELRA ID: ELRA-W0100.

TRAD Arabic-French Parallel corpus of transcribed Broadcast News Speech, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 862-201-329-808-4, ELRA ID: ELRA-W0101.

TRAD Arabic-English Parallel corpus of transcribed Broadcast News Speech, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 812-050-111-234-9, ELRA ID: ELRA-W0102.

TRAD Arabic-French Web domain (blogs) Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 138-395-895-757-7, ELRA ID: ELRA-W0103.

TRAD Arabic-English Web domain (blogs) Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 762-161-069-435-5, ELRA ID: ELRA-W0104.

TRAD Arabic-French Mailing lists Parallel corpus - Test set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 895-850-015-188-4, ELRA ID: ELRA-W0105.

TRAD Arabic-English Mailing lists Parallel corpus - Test set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 858-529-510-480-2, ELRA ID: ELRA-W0106.

TRAD Arabic-French Mailing lists Parallel corpus - Development set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 333-026-450-858-0, ELRA ID: ELRA-W0107.

TRAD Arabic-English Mailing lists Parallel corpus - Development set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 213-044-240-074-6, ELRA ID: ELRA-W0108.

TRAD Chinese-French Web domain (blogs) Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 464-017-697-777-3, ELRA ID: ELRA-W0109.

TRAD Chinese-English Web domain (blogs) Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 982-341-079-331-4, ELRA ID: ELRA-W0110.

TRAD Chinese-French News Articles Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 153-566-144-442-2, ELRA ID: ELRA-W0111.

TRAD Chinese-English News Articles Parallel corpus, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 626-096-751-907-7, ELRA ID: ELRA-W0112.

TRAD Chinese-English Email Parallel corpus – Development Set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 447-281-370-489-0, ELRA ID: ELRA-W0113.

TRAD Chinese-French Email Parallel corpus – Development Set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 255-358-917-604-3, ELRA ID: ELRA-W0114.

TRAD Chinese-English Email Parallel corpus – Test Set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 985-956-234-357-3, ELRA ID: ELRA-W0115.

TRAD Chinese-French Email Parallel corpus – Test Set, in ELRA catalogue (http://catalogue.elra.info), ISLRN: 239-027-077-538-0, ELRA ID: ELRA-W0116.

A Framework for Multi-Language Service Design with the Language Grid

Donghui Lin[†], Yohei Murakami[*], and Toru Ishida[†]
[†]Department of Social Informatics, Kyoto University,
Yoshida-Honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
{lindh,ishida}@i.kyoto-u.ac.jp
[*]Unit of Design, Kyoto University,
91 Chudoji Awata-cho, Shimogyo, Kyoto 600-8815, Japan
yohei@i.kyoto-u.ac.jp

Abstract

To collect and share language resources like machine translators and dictionaries, we developed the Language Grid in 2006, a service-oriented language infrastructure on the Internet. Although we have put a lot of effort into improving the service grid technologies and collecting language services, international NPO/NGOs are struggling with the design and development of tools and systems for supporting multi-language communication in the real world by utilizing available language services. This paper proposes a framework for service design with the Language Grid by bridging the gap between language service infrastructures and multi-language systems. The proposed framework is implemented as a toolkit, Multilingual Studio, which is open to allow the users to design and develop multilingual communication services and tools in the real world.

Keywords: language service infrastructure, service design, multi-language systems

1. Introduction

The Language Grid, a service-oriented language service infrastructure on the Internet, enables the sharing of various language resources as language services around the world (Ishida, 2011). We started the research and development of the Language Grid in 2006, focusing on the technologies of service grid server software and mechanisms permitting its federated operation among multiple organizations (Murakami et al., 2012) (Murakami et al., 2014). The Language Grid has been widely used as a platform for the research of services computing (Lin et al., 2012), language resources (Lin et al., 2010), and human-computer interaction (Murakami et al., 2018).

During the past several years, the importance of language service infrastructures has been widely recognized by the research community of language resources. Typical examples of other language service initiatives include U-Compare (Kano et al., 2009), META-SHARE (Piperidis, 2012), PANACEA (Toral et al., 2011), DKPro (Eckart de Castilho and Gurevych, 2014), and the LAPPS Grid (Ide et al., 2014). Most of the work focus on the interoperability of language resources. Unfortunately, international NPO/NGOs still find it difficult to design and develop systems for supporting multi-language communication due to the complicated situations in the real world and the difficulties of customizing language resources to meet users' requirements.

Different types of users and developers have different requirements regarding language service infrastructures. To support language service developers, we have provided a series of standardized interfaces of atomic language services and composite language services with various Web service specifications in the Language Grid (Murakami et al., 2011). For end users of the Language Grid, we have developed several tools for trying out various language resources through a web browser and customizing multilingual communication components as well (Sakai et al., 2009) (Tanaka et al., 2010). To support application developers, this paper aims at providing a framework that deals with the gap between language service infrastructures and multi-language systems.

To achieve the above goal, we address the following two issues in this paper. First, since multi-language service design and development in the real world is always an iterative process (Lin and Ishida, 2013) (Lin and Ishida, 2014), language services need to be deployed as flexible components for easy invocation and composition. Second, it is necessary to enable application developers of multi-language systems to use their preferred programming languages without mastering Web service specifications and technologies.

The contributions of this paper are as follows. First, we propose a framework for multi-language service design to bridge the gap between the service-oriented language infrastructures and multi-language applications by introducing a layer of service invocation components that enables application developers to easily invoke language services in the Language Grid and flexibly manage customized local language services. Second, we implement the proposed framework as Multilingual Studio, which is a toolkit open to the users and application developers for designing and developing multi-language services in the real world.

The rest of this paper is organized as follows: In Sect. 2, we introduce development and operation of the Language Grid, and then explain the requirements for multi-language service design in Sect. 3. Section 4 describes our design concept and the proposed framework for multi-language service design with the Language Grid. Section 5 introduces the implementation of our proposed framework and the applications. Finally, the conclusion is presented in the last section.

2. The Language Grid

Developed as a service-oriented language infrastructure that enables users to share and create language resources on the Internet, the Language Grid (Ishida, 2011) is built on service grid server software, and consists of five parts: Service Manager, Service Supervisor, Grid Composer, Service Database, and Composite Service Container (Murakami et al., 2011). We have operated the Language Grid since 2007 for non-profit use and research use. To enhance the sharing of language resources, we established the federated operation of the Language Grid with three organizations in Thailand (Bangkok), Indonesia (Jakarta), and China (Urumqi) after 2010. As of February 2018, 225 language services are shared among the federated Language Grid[1]. Moreover, we started the Open Language Grid (Ishida et al., 2014) to allow users to access language services for profit use and personal use.

We have been aiming to support three types of users and developers in the Language Grid: language service developers who are familiar with Web service technologies, multi-language application developers who use the language services, and end users who use multi-language applications for various purposes.

To support language service developers, we have put effort into standardizing language services by constructing a Language Grid Ontology (Murakami et al., 2014). All the language services in the Language Grid are wrapped from language resources by standardized Web service interfaces defined by the Language Grid Ontology. Using the atomic Web services, we have also developed a series of composite services. For example, a composite machine translation service is composed of a morphological analysis service, a dictionary service, and a machine translation service. Some of the available service interface classes and examples of corresponding service types defined by the Language Grid Ontology are shown below.

- `translate`: BackTranslation, MultihopTranslation, Translation, TranslationWithTemporalDictionary
- `search`: BilingualDictionary, BilingualDictionaryWithLongestMatchSearch, ConceptDictionary, DialogCorpus, ParallelText, PictogramDictionary
- `parse`: DependencyParse
- `identify`: LanguageIdentification
- `analyze`: MorphologicalAnalysis
- `tag`: NamedEntityTagging
- `recognize`: SpeechRecognition
- `speak`: TextToSpeech
- `paraphrase`: Paraphrase
- `calculate`: SimilarityCalculation

To support end users, we developed two general tools: Language Grid Playground[2] and Language Grid Toolbox[3]. Language Grid Playground was developed as a Web application to allow users to access and try out language services in the Language Grid via a Web browser (Sakai et al., 2009). However, Language Grid Playground was designed as a showcase for displaying the Language Grid services rather than a tool for supporting intercultural collaboration (Tanaka et al., 2011). On the other hand, Language Grid Toolbox was developed for customizing multilingual communication tools by providing flexible modules of multilingual BBS, text translation, language resource management and language service creation (Tanaka et al., 2010). Language Grid Toolbox has been contributing to the support of intercultural collaboration and multi-language communication in various fields including education, medical care, and agriculture (Ishida, 2011) (Ishida, 2016).

In this paper, we focus on supporting application developers for multi-language service design with the Language Grid. To this end, we need to propose a framework that enables application developers to easily invoke and compose language services.

3. Multi-Language Service Design

So far the development and operation of the Language Grid have been focused on supporting language service developers and end users. However, application developers in international NPO/NGOs continue to struggle with available language services in creating multi-language systems. The reason lies in the fact that the complicated situations in the real fields make it difficult to customize language resources to meet users' requirements (Ishida et al., 2016). To design multi-language systems for the real world, the iterative service design approach is always applied; it consists of the following four phases (Lin and Ishida, 2014).

- **Observation:** This phase is to understand and abstract requirements for multi-language service design. Through observation in the real world, user requirements, multi-language service flows, available language services and evaluation criteria are clarified.
- **Modeling:** This phase is to model the multi-language service that can best satisfy the user requirements. Available language services are composed in this phase based on evaluation of quality of services.
- **Implementation:** There are two aspects of implementation of multi-language services: implementation of service composition, and embedding of composite services into application systems. Therefore, it is necessary to enable the application developers to easily design and test service composition.
- **Analysis:** This phase is to evaluate the multi-language service by analyzing log data and interview results based on defined evaluation criteria. Service usage information and problems with service composition are

[1] The list of available language services can be found at `http://langrid.org/service_manager/`

[2] `http://langrid.org/playground/`

[3] `http://langrid.org/tools/toolbox/`

explored for discovering more specific user requirements and improving the service process model in an iterative manner.

In many cases, human services are involved in the multi-language service design process to compensate the limitations of available language services (Lin et al., 2014). Moreover, the service design process might have many iterations when the situations in the real fields are complicated and change frequently (Lin and Ishida, 2014).

4. Design Concept

Our goal is to propose a framework for supporting application developers of multi-language service by using the Language Grid. Due to the gap between language service infrastructure and multi-language applications, we need a framework that enables language services to be deployed as flexible components for easy invocation and composition. Moreover, the proposed framework must allow developers to use their preferred programming languages without mastering knowledge of Web services and the different interfaces of the language resources.

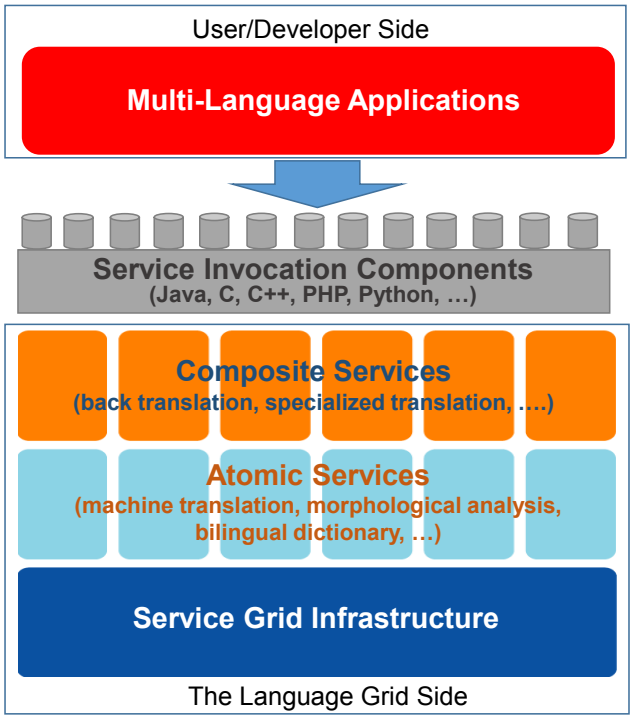


Figure 1: Framework for multi-language service design with the Language Grid by introducing a layer of service invocation components

4.1. The Proposed Framework

The Language Grid was originally designed with four layers (Ishida et al., 2011). The bottom layer is the service grid infrastructure that manages all requests to the Language Grid and invokes language services. The layers of atomic service and composite service enable language service developers to deploy services that are wrapped from language resources. The layer of multi-language applications include a series of collaboration tools.

To achieve our design goal, we propose a framework with a layer of service invocation components between the language services and multi-language applications as shown in Fig. 1. The purpose of introducing service invocation components is to transform Web service interfaces into libraries that support various programming languages for easy service invocation and management using APIs.

4.2. Service Invocation Components

Table 1 shows important functions provided in the service invocation components, including service authentication, atomic service invocation, composite service invocation, and local service management.

Table 1: Overview of functions in the service invocation components

Function	Description
Service Authentication	This function provides an API for specifying authentication information to access language services. Developers can either specify the default authentication information for the Language Grid or for each language service.
Atomic Service Invocation	This function provides an API for invoking each type of atomic language services that are defined by the Language Grid Ontology. Developers need to specify the service endpoint URL and values for invocation parameters when using an API for invoking an atomic service.
Composite Service Invocation	This function provides an API for invoking each type of composite language services that are defined by the Language Grid Ontology. Developers need to specify the service endpoint URL, values for invocation parameters, binding information with a service ID list of concrete atomic services that act as components of the composite service.
Local Service Management	This function provides a series of APIs for creating, updating, deleting a local language resource. It also provides the APIs for deploying the local resource as a local language service for invocation.

Table 2 shows an example of API for invoking a service with the service interface of **translate**. As described in Sect.2, the service interface class **translate** has a list of corresponding atomic/composite service types (**BackTranslation, MultihopTranslation, Translation, TranslationWithTemporalDictionary**) defined by the Language Grid Ontology. Therefore, the API example in Table 2 can be used to invoke either an atomic translation service or a com-

posite translation service, depending on the specified service endpoint URL.

Table 2: An example of API: service interface of **translate**

API	**Service Interface: translate** String translate(String $sourceLang, String $targetLang, String $source)
Description	This API invokes a translation service that belongs to the service interface of **translate**, following the translation setting identified by three parameters: source language, target language, and the string to be translated. This API is used to invoke an atomic translation service when specified with the service endpoint URL for atomic translation (e.g. **Translation**), or a composite translation service when specified with the service endpoint URL for composite translation (e.g. **TranslationWithTemporalDictionary**). Detailed examples of invoking atomic services and composite services using this API will be introduced in Sect.5.
Parameters	**$sourceLang**: source language **$targetLang**: target language **$source**: the string to be translated
Return Value	The translation result will be returned.

Provided the APIs for the service invocation components, application developers can easily invoke various language services registered in the Language Grid and manage customized local services for flexible multi-language service design and development.

5. Implementation of Multilingual Studio

We implemented the service invocation components in the Language Grid as a toolkit called Multilingual Studio[4]; it is open and allow users to design and develop multi-language systems in the real world. As of 2018, Multilingual Studio provides Java library and PHP library for language service invocation in the Language Grid. Libraries of other programming languages can be implemented using our proposed framework as well.
With Multilingual Studio, developers can easily invoke all 225 language services registered in the Language Grid by using their preferred programming languages. Moreover, developers can also manage, deploy and invoke local services like dictionaries and parallel texts.

5.1. Service Invocation using Multilingual Studio

Figure 2 shows an example of using PHP to invoke an atomic translation service using Multilingual Studio. The following information is required for invoking an atomic service: API client of the service type, Web service description URL of the atomic service (WSDL) as the service endpoint, authentication information, and specified values for service invocation parameters used in the API. To invoke an atomic service, developers only need to write several lines of source code by using Multilingual Studio.
Moreover, developers can reuse the source code for invoking the same type of atomic service by replacing the Web service description URL of the atomic service because the service interface type is standardized in the Language Grid. Figure 3 shows an example of reusing the source code to invoke a different translation service[5].

```
;; Create the atomic language service client
$client = ClientFactory::createTranslationClient
  ('http://langrid.org/service_manager/wsdl/
    kyoto1.langrid:KyotoUJServer');

;; Specify the service authentication information
$client->setUserId('someUserId');
$client->setPassword('somePassword');

;; Set invocation parameters and get the result
$result = $client->translate(
  Language::get('en'), Language::get('ja'),
  'Have a nice day!');
```

Figure 2: Example of invoking the atomic translation service `KyotoUJServer`

```
;; Create the atomic language service client
;; Only URL is different with that of previous example
$client = ClientFactory::createTranslationClient
  ('http://langrid.org/service_manager/wsdl/
    kyoto1.langrid:GoogleTranslate');

;; Specify the service authentication information
$client->setUserId('someUserId');
$client->setPassword('somePassword');

;; Set invocation parameters and get the result
$result = $client->translate(
  Language::get('en'), Language::get('ja'),
  'Have a nice day!');
```

Figure 3: Example of invoking the atomic translation service `GoogleTranslate`

Figure 4 shows an example using Multilingual Studio to invoke a composite translation service. We use the approach of hierarchical service composition in the Language Grid by introducing the *bind* function which can assign atomic services or composite services to the service invocation in a composite service, so that we can create service composition variant virtually at runtime (Nakaguchi et al., 2016). Therefore, the only additional information that is necessary for invoking a composite service is the service binding information, which specifies concrete atomic services forming the composite service as we have described in Sect.4. In the example of the composite translation service shown in Fig. 4, the developer specifies three atomic service bindings: **BilingualDictionaryPL** for a dictionary service, **MorphologicalAnalysisPL** for a morphological service, and **TranslationPL** for a machine translation service.

[4] `http://langrid.org/developer/`

[5] Both `KyotoUJServer` and `GoogleTranslate` belong to the same service type `Translation` in the Language Grid.

```
;; Create the composite language service client
$client = ClientFactory::createTranslationClient
  ('http://langrid.org/service_manager/wsdl/kyoto1.
    langrid:TranslationCombinedWithBilingualDictionary');

;; Specify the service authentication information
$client->setUserId('someUserId');
$client->setPassword('somePassword');

;; Specify the composite service binding information
$client->addBindings(new BindingNode
  ("BilingualDictionaryPL", "KyotoTourismDictionaryDb"));
$client->addBindings(new BindingNode
  ("MorphologicalAnalysisPL", "TreeTagger"));
$client->addBindings(new BindingNode(
  "TranslationPL", "KyotoUJServer"));

;; Set invocation parameters and get the result
$result = $client->translate(
  Language::get('en'), Language::get('ja'),
  'Mount Hiei lies between Kyoto and Shiga.');
```

Figure 4: Example of invoking a composite translation service that consists of a machine translation service, a morphological analysis service and a dictionary service

Similar with invoking an atomic service, using Multilingual Studio to invoke a composite service also requires just a few lines of source code. Moreover, multi-language application developers can reuse source code to invoke the same composite service by replacing bound atomic services. By this means, source code rewriting to invoke different composite services can be dramatically reduced, which is extremely important in the design process of multi-language systems. A detailed evaluation of the effects of hierarchical service composition was reported in our previous work (Nakaguchi et al., 2016).

5.2. Applications

Multilingual Studio has been used by application developers for various purposes including scientific analysis of multi-language activities, service development in the real multi-language field, and integration with other simulation tools for participatory service design.

Terui and Hishiyama conducted the research of cross-cultural analysis by developing a multilingual case method system for global classroom environments to benefit students of different cultures and native languages (Terui and Hishiyama, 2013). To analyze the effects of self-tagging during multilingual conversational chat, Nose and Hishiyama developed a multilingual gaming simulation environment (Nose and Hishiyama, 2013). Multilingual Studio was also used to develop multi-language tools for analyzing expert knowledge transmission (Suzuki and Hishiyama, 2016) and multi-language simultaneous display in translation systems (Sato and Hishiyama, 2017).

In the multi-language field activities, a typical example is multi-language service design for the YMC-Viet project during 2011 to 2014, which is an agricultural support project for Vietnamese farmers by Japanese experts through children of the farmers (Takasaki et al., 2015) (Lin et al., 2016). In the YMC-Viet project, Multilingual Studio was effectively used for designing and simulating the composite translation service by applying the iterative service design process we have described in Sect.2 (Yamaguchi et al., 2013) (Lin and Ishida, 2014).

Moreover, Multilingual Studio has been used together with a multi-agent gaming simulation tool called MAGCruise for participatory service design of multi-language systems (Lin and Ishida, 2013) (Nakajima et al., 2015).

6. Conclusion

The development and operation of the Language Grid have been focused on supporting language service developers and end users of multi-language systems. In this paper, we aimed at bridging the gap between language service infrastructures and multi-language systems to support application developers for service design. To achieve this goal, we proposed a framework for service design for the Language Grid by introducing a layer of service invocation components that enable developers to easily invoke language services in the design of multi-language systems. The proposed framework was implemented as Multilingual Studio, which transforms Web service interfaces of the Language Grid into libraries of different programming languages. Multilingual Studio has been used for scientific analysis of multi-language activities and multi-language service design in the real world.

7. Acknowledgements

We are grateful to Dr. Yuu Nakajima, Lecture of Toho University, Japan for helping design part of the specifications for Multilingual Studio. This research was supported by a Grant-in-Aid for Scientific Research (A) (17H00759, 2017-2020) and the Kyoto University Foundation.

8. Bibliographical References

Eckart de Castilho, R. and Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11.

Ide, N., Pustejovsky, J., Cieri, C., Nyberg, E., Wang, D., Suderman, K., Verhagen, M., and Wright, J. (2014). The Language Application Grid. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 22–30.

Ishida, T., Murakami, Y., and Lin, D. (2011). The Language Grid: Service-oriented approach to sharing language resources. In *The Language Grid: Service-oriented collective intelligence for language resource interoperability*, pages 3–17. Springer.

Ishida, T., Murakami, Y., Lin, D., Nakaguchi, T., and Otani, M. (2014). Open Language Grid - towards a global language service infrastructure. In *Proc. of the Third ASE International Conference on Social Informatics*.

Ishida, T., Lin, D., Otani, M., Matsubara, S., Murakami, Y., Hishiyama, R., Nakajima, Y., Takasaki, T., and Mori, Y. (2016). Field-oriented service design: A multiagent approach. In *Serviceology for Designing the Future*, pages 451–463. Springer.

Ishida, T. (2011). *The Language Grid: Service-oriented collective intelligence for language resource interoperability*. Springer Science & Business Media.

Ishida, T. (2016). Intercultural collaboration and support systems: A brief history. In *International Conference on Principles and Practice of Multi-Agent Systems*, pages 3–19. Springer.

Kano, Y., Baumgartner, W., McCrohon, L., Ananiadou, S., Cohen, K., Hunter, L., and Tsujii, J. (2009). U-Compare: Share and compare text mining tools with UIMA. *Bioinformatics*, 25(15):1997–1998.

Lin, D. and Ishida, T. (2013). Participatory service design based on user-centered QoS. In *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01*, pages 465–472. IEEE Computer Society.

Lin, D. and Ishida, T. (2014). User-centered service design for multi-language knowledge communication. In *Serviceology for Services*, pages 309–317. Springer.

Lin, D., Murakami, Y., Ishida, T., Murakami, Y., and Tanaka, M. (2010). Composing human and machine translation services: Language Grid for improving localization processes. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*.

Lin, D., Shi, C., and Ishida, T. (2012). Dynamic service selection based on context-aware QoS. In *Services Computing (SCC), 2012 IEEE Ninth International Conference on*, pages 641–648. IEEE.

Lin, D., Ishida, T., Murakami, Y., and Tanaka, M. (2014). QoS analysis for service composition by human and web services. *IEICE TRANSACTIONS on Information and Systems*, 97(4):762–769.

Lin, D., Ishida, T., and Otani, M. (2016). A value co-creation model for multi-language knowledge communication. In *Serviceology for Designing the Future*, pages 435–447. Springer.

Murakami, Y., Lin, D., Tanaka, M., Nakaguchi, T., and Ishida, T. (2011). Service grid architecture. *The language grid: Service-oriented collective intelligence for language resource interoperability*, pages 19–34.

Murakami, Y., Tanaka, M., Lin, D., and Ishida, T. (2012). Service grid federation architecture for heterogeneous domains. In *Services Computing (SCC), 2012 IEEE Ninth International Conference on*, pages 539–546. IEEE.

Murakami, Y., Lin, D., and Ishida, T. (2014). Service-oriented architecture for interoperability of multilanguage services. In *Towards the Multilingual Semantic Web*, pages 313–328. Springer.

Murakami, Y., Lin, D., and Ishida, T. (2018). *Services Computing for Language Resources*. Springer.

Nakaguchi, T., Murakami, Y., Lin, D., and Ishida, T. (2016). Higher-order functions for modeling hierarchical service bindings. In *Services Computing (SCC), 2016 IEEE International Conference on*, pages 798–803. IEEE.

Nakajima, Y., Hishiyama, R., and Nakaguchi, T. (2015). Multiagent gaming system for multilingual communication. In *Culture and Computing (Culture Computing), 2015 International Conference on*, pages 215–216. IEEE.

Nose, T. and Hishiyama, R. (2013). Analysis of self-tagging during conversational chat in multilingual gaming simulation. In *Future Generation Communication Technology (FGCT), 2013 Second International Conference on*, pages 81–86. IEEE.

Piperidis, S. (2012). The META-SHARE language resources sharing infrastructure: Principles, challenges, solutions. In *Proc. of the 8th International Conference on Language Resources and Evaluation Conference (LREC'12)*, pages 36–42.

Sakai, S., Gotou, M., Murakami, Y., Morimoto, S., Morita, D., Tanaka, M., and Ishida, T. (2009). Language Grid Playground: Light weight building blocks for intercultural collaboration. In *Proceedings of the 2009 international workshop on Intercultural collaboration*, pages 297–300. ACM.

Sato, M. and Hishiyama, R. (2017). An analysis of multi-language simultaneous display in the translation system. In *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual*, volume 2, pages 666–671. IEEE.

Suzuki, H. and Hishiyama, R. (2016). An analysis of expert knowledge transmission using machine translation services. In *Proceedings of the Seventh Symposium on Information and Communication Technology*, pages 352–359. ACM.

Takasaki, T., Murakami, Y., Mori, Y., and Ishida, T. (2015). Intercultural communication environment for youth and experts in agriculture support. In *Culture and Computing (Culture Computing), 2015 International Conference on*, pages 131–136. IEEE.

Tanaka, M., Murakami, Y., Lin, D., and Ishida, T. (2010). Language Grid Toolbox: Open source multi-language community site. In *Universal Communication Symposium (IUCS), 2010 4th International*, pages 105–111. IEEE.

Tanaka, M., Inaba, R., Nadamoto, A., and Shigenobu, T. (2011). Intercultural collaboration tools based on the Language Grid. In *The Language Grid: Service-oriented collective intelligence for language resource interoperability*, pages 35–49. Springer.

Terui, K. and Hishiyama, R. (2013). Multilingual case method system for cross-cultural analysis. In *Culture and Computing (Culture Computing), 2013 International Conference on*, pages 117–122. IEEE.

Toral, A., Pecina, P., Way, A., and Poch, M. (2011). Towards a user-friendly webservice architecture for statistical machine translation in the PANACEA project. In *Proc. of the 15th Conference of the European Association for Machine Translation (EAMT'11)*, pages 63–70.

Yamaguchi, T., Hishiyama, R., Kitagawa, D., Nakajima, Y., Inaba, R., and Lin, D. (2013). Evaluation of rewriting service in language translation web services workflow. In *Culture and Computing (Culture Computing), 2013 International Conference on*, pages 21–26. IEEE.

Language Technology for Multilingual Europe: An Analysis of a Large-Scale Survey regarding Challenges, Demands, Gaps and Needs

Georg Rehm, Stefanie Hegele
DFKI GmbH, Alt-Moabit 91c, 10559 Berlin, Germany
Corresponding author: georg.rehm@dfki.de

Abstract

We present the analysis of a large-scale survey titled "Language Technology for Multilingual Europe", conducted between May and June 2017. A total of 634 participants in 52 countries responded to the survey. Its main purpose was to collect input, feedback and ideas from the European Language Technology research and innovation community in order to assess the most prominent research areas, projects and applications, but, more importantly to identify the biggest challenges, obstacles and gaps Europe is currently facing with regard to its multilingual setup and technological solutions. Participants were encouraged to share concrete suggestions and recommendations on how present challenges can be turned into opportunities in the context of a potential long-term, large-scale, Europe-wide research, development and innovation funding programme, currently titled Human Language Project.

Keywords: Multilinguality, LR National/International Projects, Infrastructural/Policy Issues, LR Infrastructures and Architectures

1. Introduction

Europe is a multilingual society with 24 official Member State languages and many additional unofficial and regional languages as well as languages of minorities, immigrants and important trade partners. Nevertheless, day in and day out, language barriers keep severely hampering the free flow of information, thought, ideas, goods and products through the continent. Powerful multilingual as well as cross-lingual and monolingual language technologies, making use of the latest Artificial Intelligence algorithms in combination with ever-growing data sets, have the potential of helping to overcome language barriers.

The recent study "Language Equality in the Digital Age – Towards a Human Language Project", commissioned by the European Parliament's Science and Technology Options Assessment Committee (STOA), recommends, to the European Union, to initiate a new, large-scale European Language Technology research, development and innovation flagship programme, called, in the study, the *Human Language Project* (HLP) (STOA, 2017). It is foreseen to be a long-term European collaborative programme between research, innovation, industry, academia, administrations and citizens with the goal of achieving the next scientific breakthroughs for the automatic processing and generation of written or spoken natural language. In addition to basic research, the Human Language Project[1] is foreseen to include applied research as well as innovation and commercialisation activities. Important research themes are, among others, (1) Crosslingual Big Data Language Analytics, (2) High-Quality Machine Translation, (3) Meaning, Semantics and Knowledge as well as (4) Conversational Technologies.

A key goal of our survey was to get an overview of the current situation of Language Technology research activities throughout Europe and to determine where important gaps and obstacles exist. More concrete details on the uniqueness of the HLP, which needs to be specifically designed for Europe's demands, are discussed in Section 5.

2. Recent Developments

The principle that all 24 official languages share an equal status and are supported on the same level is perpetuated in the EU Charter (Article 22) as well as in the Treaty on the European Union (Art. 3(3) TEU). The META-NET White Paper Series, however, has revealed that there is a steadily increasing and rather severe threat of digital extinction for at least 21 European languages (Rehm and Uszkoreit, 2012; Rehm et al., 2014).

To address this threat and recognise Europe's opportunities, among others, in the fostering of a truly Digital Single Market, META-NET[2] (a Network of Excellence consisting of more than 60 research centers in 34 European countries) has been committed to support work on multilingual technologies and to provide strategic guidance since 2010 (Rehm and Uszkoreit, 2013; Rehm et al., 2016b; Rehm et al., 2016a). Selected META-NET activities were recently funded through the EU project CRACKER (2015-2017).[3] CRACKER's objectives encompass, among others, preparing and publishing research and innovation agendas (Rehm, 2015; Rehm, 2016; Rehm, 2017). It has also established the Cracking the Language Barrier[4] federation which acts as an umbrella initiative for European projects and organisations working on technologies for multilingual Europe.

Europe has a long-standing research, development and innovation tradition with several hundred universities and research centers performing excellent, highly visible and internationally recognised research on all European and many non-European languages. Especially in the field of Machine Translation most of the basic research has happened in Eu-

[1] While identical in name, the "Human Language Project" only bears a marginal relationship to previous initiatives bearing the same name. Among others, Abney and Bird (Abney and Bird, 2010) called their "universal corpus of the world's languages" the "Human Language Project". In 2012 TAUS launched their concept for an open language resources and tools platform, which TAUS called "Human Language Project", (see:https://www.taus.net/knowledgebase/index.php/Human_Language_Project). The initiative specified in the STOA report has a much broader scope and set of objectives than the two initiatives mentioned above.

[2] http://www.meta-net.eu

[3] http://www.cracker-project.eu

[4] http://www.cracking-the-language-barrier.eu

ropean research projects. Moses (Koehn et al., 2007), until 2016 the state of the art phrase-based statistical MT system, and recent European NMT results, especially those of the European research project QT21, are just two examples for excellence and world class research (Bojar et al., 2017). Nonetheless, challenges are omnipresent and must be addressed by the EU, the Member States as well as stakeholders from academia and industry.

3. Method

The survey contains a total of 29 questions (see Appendix A) of which 16 are open questions with free text answers. The remaining ones are a mixture of multiple choice and yes/no questions. The findings of this survey served as an important contribution to the final Strategic Research and Innovation Agenda[5] on Language Technologies for Multilingual Europe which was presented and discussed at the META-FORUM 2017 conference[6] on November 13/14 and published in its final version in December 2017.

The survey is divided into three main parts covering (1) background, research interests and projects of the participants, (2) visions for a large-scale European Language Technology research and development programme and (3) ideas on talent generation and retention in Europe. This division allowed to capture an overview of current and on-going research activities and developments in the field in the first part, reaching early-stage as well as more senior community members. The second part was intended to gather more expert knowledge with regard to visions and concrete plans for future work, in particular steps and prerequisites needed for initializing a large-scale Human Language Technology Project tailored especially to Europe's demands and current opportunities. The third and final part addresses the current challenge of the brain drain the European LT (and also AI) community is experiencing.

Participants were not obliged to answer all questions, but encouraged to fill in the ones they feel comfortable with. The survey was designed and set up using the service Typeform[7], a software for building online forms (see Figure 1).

4. Analysis

The survey was launched on 16 May and closed on 4 July 2017. As an incentive to maximise the number of answers, those who submitted the survey had the chance to win a tablet computer. After testing and making sure that the questions were phrased the right way, the survey was shared within a smaller circle (mainly members of META-NET, META, CRACKER as well as members of the Cracking the Language Barrier federation) with an appeal to share the survey within their own respective networks and also through social media. In a second round a wider audience of more than 4000 people was targeted, including participants of former META-FORUM and other conferences as well as respondents of the META-NET Open Letter campaign, conducted in 2015 (Rehm et al., 2016a).[8] We also announced the survey on the major mailing lists relevant for the field.

[5]http://cracker-project.eu/sria/
[6]http://www.meta-net.eu/events/meta-forum-2017/
[7]https://www.typeform.com
[8]http://multilingualeurope.eu

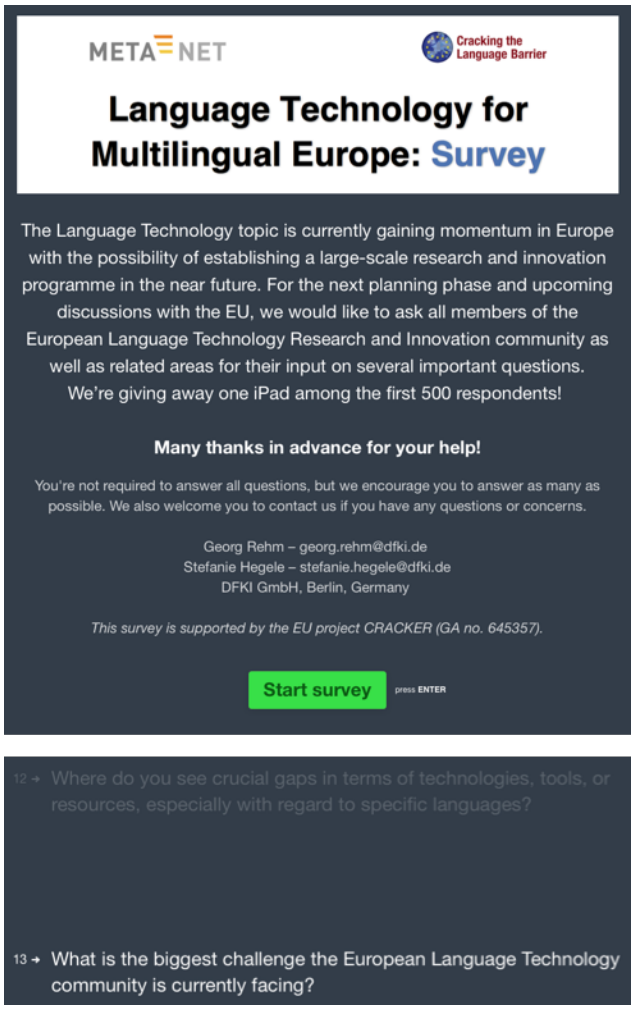


Figure 1: Welcome screen and example of survey questions

The survey created a total of 634 responses and, considering the number of questions, a surprisingly high completion rate of 27%. The average time needed for completing the survey was 35,48 minutes (see Figure 2). Both the completion rate and the average time indicate that the respondents are very passionate about the language topic and Europe's multilingual challenge. One major goal of this survey was to bring the European LT community together and gather responses from a wide and demographically distributed audience.

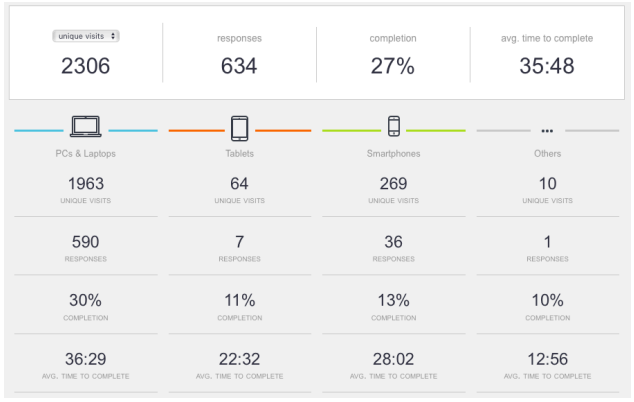


Figure 2: Survey completion rates on different devices

5. Analysis

The Human Language Project, as initially suggested in the STOA study (STOA, 2017), is to provide a sizable amount of funding in order for the field to reach a set of strategic research objectives. All European research, development and innovation projects that fall under the umbrella of the HLP are to be coordinated in a systematic way. The key scientific goal and also challenge of the HLP is an ambitious goal: Deep Natural Language Understanding by 2030 (including Generation). In the survey we asked the respondents multiple questions regarding key characteristics of a potential Human Language Project to get a better idea of where the needs, gaps and demands are. In the following we give a brief overview of the challenges, we discuss a possible setup of the HLP, provide details regarding important research areas mentioned in the responses as well as the economic impact (as suggested and discussed in the survey). Since the scope of this paper does not allow to analyse all 29 questions in detail, we focus on the ones we consider most insightful and provide relevant quantitative and qualitative statistics and findings (see Figure 5 for some of the key insights). We refer to specific questions, listed in Appendix A, using abbreviations in the form of Q1, Q2 etc. The analysis follows the survey's original tripartition.

5.1. Part 1: Background and Research Interests

The first part of the survey consisted of 14 questions aiming to collect background information of participants' organisations and their size (and also revenue if applicable) as well as the type of role and day-to-day responsibilities. Further, participants were asked to define the research fields, areas and sub-areas, methods and applications they work on. Particularly important for this survey was to assess in which economic sectors developed applications can be used. Finally, two open answer questions tackle the problems on current gaps (especially with regard to particular languages) and challenges within the European LT community.

Below we present the demographic details, the current challenges and gaps in terms of technology as well as their economic impact, especially with regard to the Digital Single Market (DSM).

5.1.1. Respondents Demographics

Access statistics of the survey web page and Google Analytics reveal that the survey was opened by potential respondents in 67 different countries with most views from 1) Germany, 2) Spain, 3) United Kingdom and 4) Italy. Completed surveys were collected from 52 countries (see Figure 4). Among the represented countries were 37 European countries including 27 EU Member States.

As for socioeconomic statistics, the distribution shows that a large majority of participants hold senior roles at their respective organisations (such as professor, senior researcher, group leader etc.). This information about the roles seen in context with the seniority level (53% have more than 20 years of work experience and another 27% more than 10 years) and the participation from 52 countries clearly portrays a wide and diverse range of the European Language Technology research and innovation community (and even beyond). This expertise and long experience are accordingly reflected in the high quality of answers collected (see Figure 3 for more statistics). The most commonly represented research fields include Language Technology (64%), Computational Linguistics (56%), General Linguistics (42%), Artificial Intelligence (39%) and Computer Science (31%).

The majority of participants is based at universities and research centers. The most frequently mentioned organisations were: Charles University in Prague, Vilnius University, University of Copenhagen and DFKI. However, the survey also reached a substantial group of participants from industry, 33 (corresponding to 5% of all respondents) from large enterprises with more than 10000 employees such as Microsoft, IBM, Intel and Nuance and 68 (11%) from SMEs. When it comes to day-to-day responsibilities 71% of all participants state an involvement in research, closely followed by 52% naming project management and 43% project execution as their most crucial tasks. This variety of engagement and responsibilities allows to get insightful input on concrete research topics (for basic and applied research as well as innovation topics), methods and best practices as asked in the second part of the survey. In addition, the vast expertise in management and project acquisition indicates competence for answering questions related to strategic planning as well as questions requiring a wider perspective on the field such as the impact Language Technology could have on the Digital Single Market.

5.1.2. Technological Gaps and Challenges

Regarding crucial gaps in terms of technologies for specific languages (Q13), almost 40% of all respondents highlight that there is insufficient research being done for minority languages and dialects, directly resulting in a shortage of available resources. This lack becomes most evident in Machine Translation applications for smaller European languages as well as other standard NLP tools and systems (according to approx. 19%). Further gaps mentioned are imposed by limited funding for low-resourced languages and copyright restrictions for certain data sets. Further, interoperability and standardisation need to be intensified.

When asked about the biggest challenge the European Language Technology field is facing at the moment (Q14) around 16% of all provided survey answers stress that the neglect of smaller languages is a severe threat, which is leading to a fragmented rather than a united and multilingual Europe. Assessing the languages most widely used in research (Q9 and Q10), around 90% state that they work with English (not exclusively though) since they are often given little incentive to solely focus on smaller or minority languages. For instance, when it comes to publishing research results there is a strong bias towards incorporating results for English. Still frequently used, though not even half as popular, are the big European languages: Spanish (49%), German (41%), French (37%) and Italian (23%).

Other challenges include the insufficient amount of data resources (approx. 13%), an unwillingness of collaboration within the community (approx. 8%) and, as already indicated above, a lack of funding (approx. 8%).

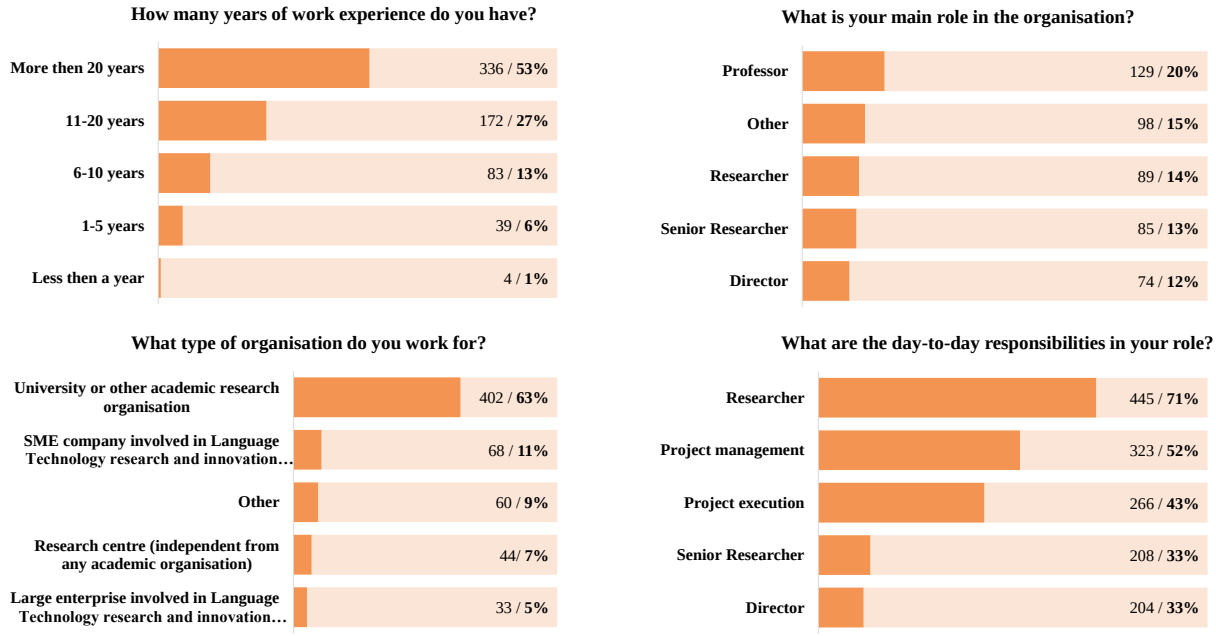


Figure 3: Overview of survey's socioeconomic statistics

5.1.3. Economic Impact and the DSM

We also asked the respondents questions (Q11, Q12) regarding the economic impact of language technologies, especially in the context of the Digital Single Market (DSM). Identified as the sectors to have the highest potential contribution to commercial growth are Education (71%), Information & Communication Technologies (64%) as well as Human Health & Social Work (45%). Specific services and applications that could benefit the Multilingual Digital Single Market comprise better Language Resources and Technologies (73%), Translation Services (46%), Multilingual Solutions for E-Learning (41%) and E-Health (38%). In the context of industries, sectors and verticals the necessity of an on-going knowledge transfer and effective collaboration between academia and industry is highlighted. The Health sector is unequivocally the most significant one, Education comes in second, closely followed by Tourism and Travel, Law and Justice, Translation, E-Commerce, Entertainment (incl. arts, creativity, culture and cultural heritage), Media, Business (incl. various services and business intelligence applications), Security, Public services and Administration, Government and Finance. Socio-economic opportunities are brought by guaranteeing better access to multilingual data and services for all people. This establishes a solid basis for the inclusiveness of minorities and people with special needs. Thus, in a wider context multilingualism helps remove barriers, fosters collaboration and creates more cultural awareness.

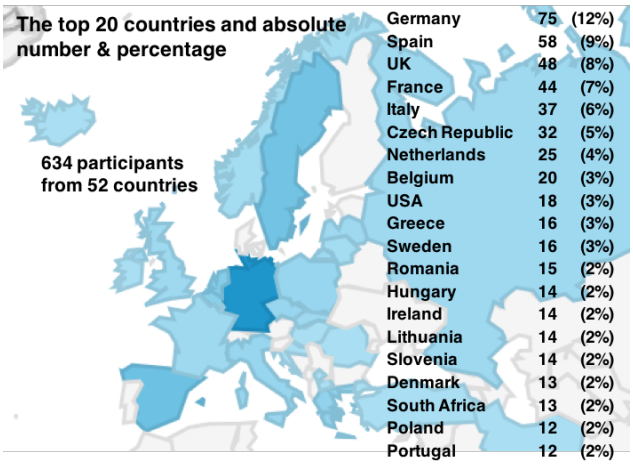


Figure 4: Number of collected responses sorted by country

5.2. Part 2: Visions for a Future Large-Scale Language Technology Programme

The second part entailed a total of 11 questions assessing the general support for a joint European Language Technology Project tackling the challenge of Deep Natural Language Understanding. In the following we analyse the questions on the organisational set up, strategic guidelines and governance of a potential Human Language Project, the most important research areas as well as applications and services that should be components of a HLP.

5.2.1. Support for a Human Language Project

The overall suggestion to initiate a large-scale Human Language Project (HLP) received substantial support from the group of respondents with 97% stating that they are in favour of establishing such a funding programme. Only a very small number of participants (3%) does not agree; their main arguments are unsuccessful previous attempts of similar programmes which did not achieve their targeted goals because of bureaucratic hurdles and a lack of focus. Further-

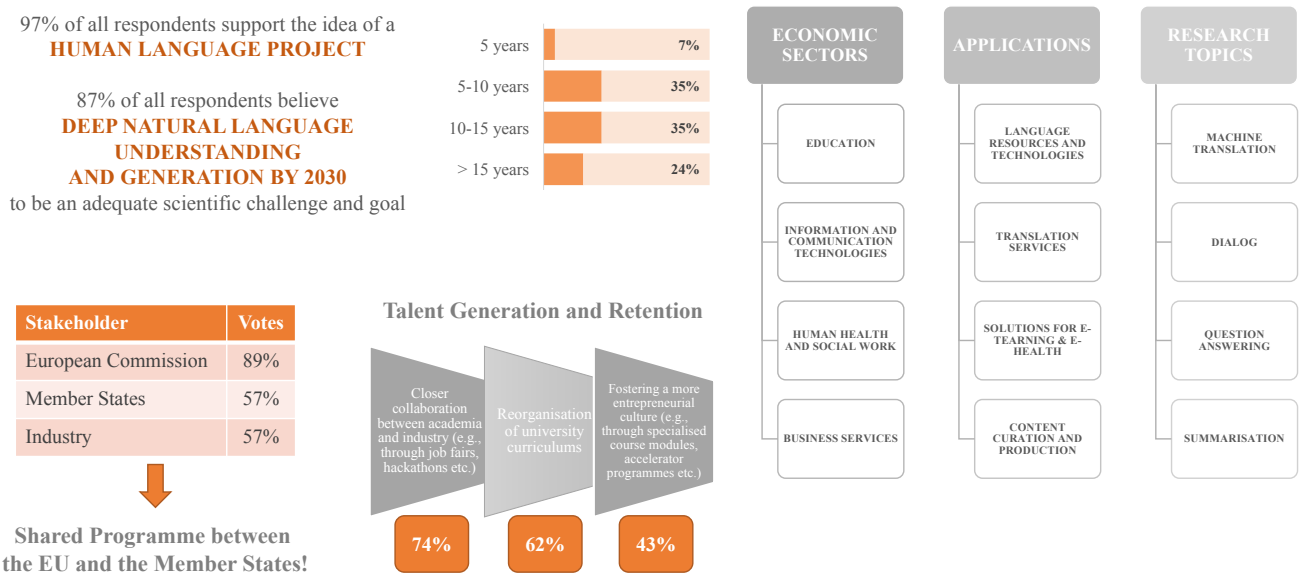


Figure 5: Overview of survey's key input on a Human Language Project

more, 87% consider the survey's suggested key strategic vision – to achieve Deep Natural Language Understanding and Generation by 2030 – as realistic and therefore an adequate scientific challenge. An appropriate timeframe would be likely to fall in the range of 10-15 years (7% believe that 5 years is a sufficient period, 35% opt for 5-10 years and another 35% for 10-15 years).

5.2.2. Organisational Frame and Governance

As far as funding is concerned a shared responsibility between the European Union, industry and member states was envisioned with the EU as the stakeholder that should be "naturally" responsible. The distribution of votes for stakeholder involvement looks as follows: European Commission (89%), Industry (57%) and Member states (57%).

When it comes to strategic guidance what can be derived from the survey responses is the strong suggestion to concentrate funding on smaller scale projects, starting bottom-up with smaller goals, and also to avoid heavy bureaucracy.

Regarding the governance of a potential HLP, one shared suggestion is the wish to put democratic organisation processes in place, e. g. with shifting presidents and elected committee and board members among institutions and countries. Also highlighted was the need to reposition the strategy of EU research with a focus on scientific breakthroughs in order to diversify from the US and large corporation paradigms. This involves fostering strong collaborations between stakeholders, better school and especially university education with more incentives for young researchers (see Section 5.3.), integration of user and customer experience as well as feedback processes, following market-driven approaches to ensure industrial growth.

5.2.3. Key Research Areas

In terms of research, the Human Language Project aims to tightly intertwine basic research, applied research, innovation and commercialisation (Q20).

As far as basic research is concerned a majority mentioned the further development of existing resources (incl. corpora, ontologies, dictionaries etc.) and improvement of data annotations (approx. 9%). In this context, effective legal frameworks for better accessibility are also necessary. Besides, basic research should be centered around deep learning and neural networks (approx. 7%) as well as Natural Language Understanding (approx. 7%). A majority also highlighted the need to further work on existing NLP tasks and tools such as Question Answering, Summarisation, Information Extraction and Sentiment Analysis (approx. 6%).

Applied research should strongly focus on MT according to around 13% of all respondents. Seen as crucial is thereby, again, the improvement of multilingual resources, data sets and terminology repositories, allowing for standardisation and interoperability (approx. 10%). In addition, there is a demand for improved open-source platforms with a wide range of available systems and applications and truly open and unencumbered data and code repositories (approx. 4%), which are further discussed in Section 5.2.4.

When it comes to innovation the inclusion of all languages and fostering of inter-cultural systems is regarded as a top priority (9%). This also presupposes better and stronger relations between academia and industry (7%). Also stressed is the need to bring together knowledge and methods developed for different fields and domains, e. g., e-health, e-government and e-justice (5%). In addition, there is an interest for more advanced visualisations and interfaces, new innovative tools incorporating NLU and seamless human-computer as well as human-robot interactions (5%).

5.2.4. Applications and Platforms

As for the most important topics, applications and platforms to be integrated (Q24), Machine Translation is uncontroversially the most important one according to approx. 14% of respondents. Considered as almost equally important are the availability of download services for multilingual resources including ontologies, lexicons, dictionaries etc. (approx. 10%). As for further applications a more in-depth development of already existing NLP tools is encouraged, especially speech applications (approx. 10%). Other listed applications include information extraction and retrieval, summarisation, search systems and intelligent assistants.

Among the topics and domains most relevant for the development of future applications and services are education, health, e-participation and e-government (10%).

Regarding the setup of a European Language Technology Data and Service platform and the collaboration between respective stakeholders (Q25), about 30% of all survey answers emphasise the importance of easy accessibility and open licensing for available tools and data. Commonly agreed upon exchange formats and standards also need to be set up. Almost 11% see an involvement of all stakeholders, i. e., data providers, LT providers and LT consumers, as necessary. Effective communication requires a unified, high-level, transparent and user-friendly approach with common goals (approx. 11%). Other recommendations submitted are to facilitate administrative processes on EU level, to adopt best practices from initiatives such as CLARIN[9] and META-NET, to enforce project evaluation processes and to establish business models and commercialisation plans to raise awareness for the ongoing work and the field of Language Technology in general.

5.3. Part 3: Talent Generation and Retention

The last part of the survey addresses another challenge Europe's LT community is currently facing, i. e., the constantly increasing brain drain (STOA, 2017). Q26 and Q27 assess the incentives needed for early stage researchers to stay in Europe as well as the skills that are mostly demanded in Language Technology and related fields. In order to best address the skill gap, 74% out of all respondents envision closer collaboration between academia and industry (e. g., through job fairs and hackathons). A large percentage of 62% also sees opportunities in the reorganisation of university curriculums, 43% emphasise the importance of fostering a more enterpreneurial culture through specialised course modules, accelerator programmes etc. Regarded as relevant skills are advanced linguistic knowledge and programming skills (approx. 21%). Linguistic expertise encompasses hereby all disciplines including semantics, syntax, phonetics, formal linguistics, corpus linguistics etc. The most popular programming language among the respondents is Python, closely followed by Java. Considered as the most essential soft skills are collaboration, team work and networking as well as innovative thinking, creativity and proactivity.

[9]https://www.clarin.eu

6. Conclusions

The survey has shown that there is a profound common interest and passion not only with regard to Multilingual Europe but also in making the ambitious idea of a large-scale, long-term Human Language Project a reality. A HLP should foster the creation of new approaches, algorithms, data sets and resources which can be employed across modalities, platforms and cultures. With regard to opportunities for research and technology development the three most prominent areas to focus on in the near future Natural Language Understanding, Machine Translation, Educational and Language Learning technologies as well as Deep Learning. Further, the answers emphasise that raising awareness for the Language Technology potential in Europe on a political level is more important now than ever before. The upcoming Brexit and trend of highly qualified researchers emigrating to the US leaves the European Language Technology community in a place where change is needed in order to compete with innovative systems and technologies built and research results produced in the US and elsewhere. Investing into the HLP would secure Europe's place in the pole position in this field for many years to come, solve the threat of Digital Language Extinction and create a truly multilingual Digital Single Market. In addition, it would open up a new direction in the education of young researchers, create attractive jobs for high potentials and foster innovation especially when it comes to new companies. Europe is in the position to shape and claim this topic as its own.

On top of that, the survey inspired plenty of positive comments, for example:

- *"This inspired my brains a lot. Thanks for good questions. I think this is the BEST questionnaire I have ever filled! Good luck with your work! Do not hesitate to contact me if you like to ask or discuss more. I would enjoy continuing in this kind of way, it makes me excited!"*
- *"Human Language Project is an excellent initiative."*
- *"Congratulations for the initiative and the option to include as many answers as possible."*
- *"Best wishes to the survey – this is one of the most important topics for Europe at the present time."*

Acknowledgements

CRACKER has received funding from the EU's Horizon 2020 research and innovation programme through the contract CRACKER (grant agreement no.: 645357).

Bibliographical References

Abney, S. and Bird, S. (2010). The human language project: Building a universal corpus of the world's languages. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 88–97, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi, M. (2017). Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark, September. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Georg Rehm et al., editors. (2012). *META-NET White Paper Series: Europe's Languages in the Digital Age*. Springer, Heidelberg, New York, Dordrecht, London. 31 volumes on 30 European languages. http://www.meta-net.eu/whitepapers.

Rehm, G. and Uszkoreit, H. (2013). *META-NET Strategic Research Agenda for Multilingual Europe 2020*. Springer Publishing Company, Incorporated.

Rehm, G., Uszkoreit, H., Dagan, I., Goetcherian, V., Dogan, M. U., Mermer, C., Váradi, T., Kirchmeier-Andersen, S., Stickel, G., Jones, M. P., Oeter, S., and Gramstad, S. (2014). An Update and Extension of the META-NET Study "Europe's Languages in the Digital Age". In *Proceedings of the Workshop on Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era (CCURL 2014)*, Reykjavik, Iceland, May.

Rehm, G., Hajic, J., van Genabith, J., and Vasiļjevs, A. (2016a). Fostering the Next Generation of European Language Technology: Recent Developments – Emerging Initiatives – Challenges and Opportunities. In Nicoletta Calzolari, et al., editors, *Proceedings of the 10th Language Resources and Evaluation Conference (LREC 2016)*, pages 1586–1592, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Rehm, G., Uszkoreit, H., Ananiadou, S., Bel, N., Bielevičienė, A., Borin, L., Branco, A., Budin, G., Calzolari, N., Daelemans, W., Garabík, R., Grobelnik, M., García-Mateo, C., van Genabith, J., Hajič, J., Hernáez, I., Judge, J., Koeva, S., Krek, S., Krstev, C., Lindén, K., Magnini, B., Mariani, J., McNaught, J., Melero, M., Monachini, M., Moreno, A., Odjik, J., Ogrodniczuk, M., Pęzik, P., Piperidis, S., Przepiórkowski, A., Rögnvaldsson, E., Rosner, M., Pedersen, B. S., Skadiņa, I., Smedt, K. D., Tadić, M., Thompson, P., Tufiş, D., Váradi, T., Vasiļjevs, A., Vider, K., and Zabarskaite, J. (2016b). The Strategic Impact of META-NET on the Regional, National and International Level. *Language Resources and Evaluation*. 10.1007/s10579-015-9333-4.

Rehm, G. (2015). Strategic Agenda for the Multilingual Digital Single Market – Technologies for Overcoming Language Barriers towards a truly integrated European Online Market, April. Version 0.5. April 22, 2015. Prepared by the EU-funded projects CRACKER and LT_Observatory.

Rehm, G. (2016). Language as a Data Type and Key Challenge for Big Data. Strategic Research and Innovation Agenda for the Multilingual Digital Single Market. Enabling the Multilingual Digital Single Market through technologies for translating, analysing, processing and curating natural language content, July. Version 0.9. July 04, 2016. Prepared by the Cracking the Language Barrier federation, supported by the EU-funded projects CRACKER and LT_Observatory.

Rehm, G. (2017). Language Technologies for Multilingual Europe: Towards a Human Language Project. Strategic Research and Innovation Agenda, November. Version 1.0. Unveiled at META-FORUM 2017 in Brussels, Belgium, on November 13/14, 2017. Prepared by the Cracking the Language Barrier federation, supported by the EU-funded project CRACKER.

STOA. (2017). Language equality in the digital age – Towards a Human Language Project. STOA study (PE 598.621), IP/G/STOA/FWC/2013-001/Lot4/C2, March 2017. Carried out by Iclaves SL (Spain) at the request of the Science and Technology Options Assessment (STOA) Panel, managed by the Scientific Foresight Unit (STOA), within the Directorate-General for Parliamentary Research Services (DG EPRS) of the European Parliament, March. http://www.europarl.europa.eu/stoa/.

A Survey Questions

Below we list all 29 survey questions, divided into three main blocks as well as two closing questions.

A1. Background, Research Interests, Projects

The first 14 questions focus on the demographic background, research interests and projects of the respondents.

- Q1: Personal details
- Q2: What is the name of the organization you work for?
- Q3: What type of organisation do you work for?
- Q4: What is your company's estimated annual revenue in Euro?
- Q5: What is the size of the organisation (total number of employees)?
- Q6: What is your main role in the organisation?
- Q7: What are the day-to-day responsibilities in your role?
- Q8: What are the key research fields, areas and subareas, methods and applications you work on?
- Q9: Which languages do you mainly work with in your research or offer in your products or services?
- Q10: Which languages would you like to include in your research, products or services in addition - but cannot due to a lack of technologies, tools or resources?

- Q11: In which of the following economic sectors do you see high potential for applications, opportunities for commercial growth or promising target markets for your research, products or services?
- Q12: Which of the following Language Technology applications and services for the Multilingual Digital Single Market could be improved through your research?
- Q13: Where do you see crucial gaps in terms of technologies, tools, or resources, especially with regard to specific languages?
- Q14: What is the biggest challenge the European Language Technology community is currently facing?

A2. Visions for a Future Large-Scale Language Technology Programme

Questions 15-25 focus on the vision of a Language Technology Programme (Human Language Project) in the context of Europe's multilingual challenges and gaps.

- Q15: Do you support the idea of setting up a large-scale Human Language Project?
- Q16: Are there any specific reasons why you do not support the setting up of a Human Language Project? Please specify if possible.
- Q17: Do you think Deep Natural Language Understanding by 2030 is the right vision and an adequate scientific challenge?
- Q18: Which strategic vision would you suggest instead?
- Q19: How long do you think the HLP needs to be so that it can reach the suggested scientific vision and have a significant impact?
- Q20: In the context of a HLP, what are, in your opinion, the (up to) five key challenges Europe needs to work in with regard to: a) Basic research, b) Applied research, c) Innovation, d) Industries/Sectors/Verticals
- Q21: Which are the top three research, technology development, or socio-economic opportunities that you personally envisage the HLP to bring about or to successfully address?
- Q22: Do you have any other additional suggestions or recommendations with regard to the HLP? For example, how it should be organised in terms of priority setting and governance or with regard to strategic guidance
- Q23: How should the Human Language Project be funded?
- Q24: What are, in your opinion, the five key topics,applications, services that must be included in such a platform?
- Q25: Do you have any additional recommendations regarding the setup of the European Language Technology Data andS ervice Platform? For example, regarding the collaboration between data providers, LT providers and LT consumers?

A3. Part 3: Talent Generation and Retention

Questions 26 and 27 focus on concepts for talent generation and retention in Europe.

- Q26: Which technical or soft skills do you personally consider most important for your specific area/projects?
- Q27: How can the skill gap best be addressed?

A4. Last but not least

Questions 28 and 29 focus on survey dissemination statistics and final comments.

- Q28: How did you find out about this survey?
- Q29: If you have any additional comments, concerns or suggestions please do not hesitate to share them.

Annotating High-Level Structures of Short Stories and Personal Anecdotes

Boyang Li[1], Beth Cardier[2], Tong Wang[3], and Florian Metze[4]
[1] Independent
[2] Sirius-Beta, Virginia Beach, VA
[3] University of Massachusetts Boston, Boston, MA
[4] Carnegie Mellon University, Pittsburgh, PA
boyangli@outlook.com, bethcardier@sirius-beta.com, tong.wang001@umb.edu, fmetze@cs.cmu.edu

Abstract

Stories are a vital form of communication in human culture; they are employed daily to persuade, to elicit sympathy, or to convey a message. Computational understanding of human narratives, especially high-level narrative structures, remain limited to date. Multiple literary theories for narrative structures exist, but operationalization of the theories has remained a challenge. We developed an annotation scheme by consolidating and extending existing narratological theories, including Labov and Waletsky's (1967) functional categorization scheme and Freytag's (1863) pyramid of dramatic tension, and present 360 annotated short stories collected from online sources. In the future, this research will support an approach that enables systems to intelligently sustain complex communications with humans.

Keywords: narrative structure, dramatic arc, story understanding

1. Introduction

Story is a fundamental form of human communication, sometimes argued to be more powerful than logical arguments (Bruner 1986; Fisher 1987). Stories can be used, for example, to persuade, to encourage, to elicit sympathy, and convey a moral, message, value or lesson. It follows that a computational understanding of stories will help computer systems communicate better with users. Recent years have witnessed growing interests in computational approaches for story understanding (Bamman *et al.* 2013; Ferraro and Van Durme 2016; Finlayson, M. A. 2016; Goyal *et al.* 2010; Ouyang and McKeown 2015; Huang *et al.* 2016; Pichotta and Mooney 2016; Tapaswi *et al.* 2016; Mostafazadeh *et al.* 2016; Chaturvedi *et al.* 2017; Wang *et al.* 2017; Dogan *et al.* 2018). Yet few attempts to understand high-level story structures, which are the focus of the present paper.

What constitutes story structure or a "story arc" may be debatable since there is more than one facet to a story. As an operating definition, we consider *story structure* to satisfy the following requirements: (1) it contains a small set of functions with typical orderings between them, though atypical orderings are sometimes possible. (2) The functions are independent of content and genre; they describe structures of stories with different content in any genre. (3) The functions carry significance on the dramatic arc, and (4) together they describe most of a story rather than a small part of it.

This definition rules out the functions proposed by Propp (1928) for Russian folklores, components of the hero's journey (Campbell 1949), and other similar theories because they are closely tied to one type or genre of stories and are not domain-independent. Event-level representations, such as plot units (Lehnert 1981), also do not fit the definition because not all events play important dramatic roles and the ordering between events can be rather arbitrary.

Instead, we investigate what was described by Aristotle (circa. 335 BC) as the beginning, the middle and the end of a story. Similar to Aristotle, Freytag (1863) proposed a dramatic structure containing five parts, whose modern version includes Exposition, Rising Action, Climax, Falling Action, and Dénouement. The parts are correlated with the rising and falling of dramatic tension, as illustrated in Figure 1. Labov and Waletzky's theory on narrative analysis (1967; Labov 2013) (henceforth L&W) provides another structure that starts with *Abstract* and *Orientation*, goes through *Complicating Actions*, *The Most Reportable Event*, *Evaluation*, to end with *Resolution* and *Coda*. In our opinion, structural functions proposed by the two theories both satisfy the four requirements laid out earlier.

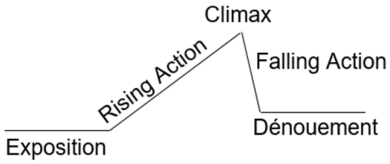


Figure 1. Freytag's story structural functions. The line indicates how dramatic tension heightens and lowers throughout the story.

Although these theories seem reasonable by themselves, some important open questions remain: (1) Are these theories compatible or mutually exclusive? If they are compatible, do they describe the same story stages using different terms? (2) Can we devise a unified version of these theories and operationalize it sufficiently, so that human annotators can reliably annotate stories using the theory?

In this paper, we attempt to answer these questions. First, we identify similar concepts that are described by both theories. Based on this understanding, we develop a new annotation scheme, which reconciles the two theories and provides additional functions that we find useful in annotating casual stories online. Finally, we trained annotators to label sentences in stories acquired from online sources and public datasets, yielding 360 unique annotated stories.

To our knowledge, this is the first effort aimed at creating an operational annotation schema that unifies different accounts of story macro-structures. Previous annotation schemata either focus on event-level representations that do not always have dramatic significance (Elson 2012;

Freytag	L&W	Prince	Todorov	Our Annotation
Exposition	Orientation	Starting State	Old Equilibrium	Orientation
Rising Action	Complicating Actions		Disruption	Complicating Actions
Climax	Most Reportable Event	State-changing Event	Efforts to repair the disruption	Most Reportable Event
Falling Action	Resolution	Ending State		(Minor) Resolution
Dénouement	Coda		New Equilibrium	Aftermath

Table 1. Correspondence between categories from different narrative theories and our annotation.

Lehnert 1981), or they use only part of the dramatic curve (Ouyang and McKeown 2015).

2. Related Work

There have been several attempts at annotating story semantics and computationally predicting semantic labels from text. Ouyang and McKeown (2015) (henceforth O&M) identified the most reportable event (MRE) in L&W as the "nucleus" of the story. Consequently, they annotated the MRE in roughly 500 stories collected from Reddit and built a classifier for identifying MREs from text, but omitted other categories from the theory. Rahimtoroghi *et al.* (2013) and Swanson *et al.* (2014) used a subset of categories from L&W, including orientation, action, and evaluation.

At the event level, Elson (2012) designed an annotation schema, Story Intention Graph (SIG), that captures timelines as well as beliefs, intentions and plans of story characters. We perceive similarities between this annotation and approaches for generating stories and character behaviors, such as Belief-Desire-Intention agents (Rao and Georgeff 1995) and intention-based story planning (Riedl and Young 2010). Lukin *et al.* (2016) annotated 108 personal stories using the SIG formalism. Finlayson (2016) produced extensive annotations for Propp's Russian folklores, ranging from co-reference and temporal ordering to semantic roles and word senses. Gervás *et al.* (2016) found Propp's functions limited to a single genre and created a new set of functions for annotating 42 musicals. In contrast to L&W's theory, we consider these annotations to be on the micro-structure of events rather than the macro-structure of the entire narrative.

Another influential event-based schema of story structure is plot units (Lehnert 1981). In this schema, an event is classified according to its sentiment as positive, negative, or a mental state with neutral sentiment. In addition, the schema further contains four types of causal links between the events: motivation, actualization, termination, and equivalence. These entities form basic plot units. For example, the pattern *success* contains an actualization link going from a mental state to a positive state. Lehnert further argued these basic units can be combined further to eventually capture story-level structure. Appling and Rield (2009) and Goyal *et al.* (2010) trained machine learning models to predict plot unit structures.

A bottom-up approach employs statistics from local regions of text to represent story structure, instead of a predefined set of function labels. Elsner (2015) collected frequency trajectories of character names combined with words expressing emotions and appearing in Latent Dirichlet Allocation (Blei *et al.* 2003) topics to represent story structure and measured similarity between trajectories. Reagan *et al.* (2016) extracted sentiment (i.e., the positive and negative polarities of emotions) from 1,327 story texts and identified 6 common patterns of how sentiment change throughout the stories.

To situate ourselves with regards to previous work, this work adopts the macro-structural view of L&W and Freytag, which differs from the event-based view or the bottom-up view of narrative structure. We propose a new functional schema that reconciles Freytag's theory with L&W, which we believe captures both dramatic tension and the social aspect of online narratives.

3. The Annotation Schema

In this section, we start by discussing several narrative theories, including Freytag (1863), L&W (1967; Labov 2013, 1997), Prince (1973), Todorov (1971), and O&M (2015). Table 1 shows the correspondence we identified between different narrative theories. After that, we propose a new set of functional labels that unify fundamental ideas from these theories.

3.1 Integrating Narrative Theories

Freytag's five-stage theory of story development in theater (1863) mainly follows the building and resolution of dramatic tension. Modern interpretations of the theory (e.g., Thursby 2006) generalize stories across numerous genres and media. The first stage, *Exposition*, introduces the narrative setting and has the lowest tension. Tension then increases during a process referred to as *Rising Action*, propelled by a crisis. Freytag's tension peaks at the *Climax*, where the forces of tension are concentrated. After the climax, we draw from Thursby's (2006) modern interpretation, where tension quickly falls towards a *Resolution* and then *Dénouement*.[1] In both professional and everyday instances of modern narratives, we have observed a swift drop in tension during this last stage, its fast resolution standing in contrast to the labor by which tension was built.

This pyramid structure is reminiscent of Todorov's analysis (1971) in which a story starts with an equilibrium, which is later disrupted. Efforts to restore the equilibrium are made and the new equilibrium is created in the end. In our interpretation, an equilibrium state has low tension. The

[1] In its second half, Freytag's original framework closely follows the nuances of tragic theatre, featuring several complicated turns of action that are difficult to generalize to other genres.

disruption leads to high tension and the restoration of equilibrium lowers the tension. Therefore, we align the old equilibrium with Exposition, the disruption with the Rising Action, and the new Equilibrium with the Dénouement.

In comparison, L&W's story structure focuses on the social relationship between the storyteller and the audience and on the surface shares little with Freytag and Todorov. The theory contains the following categories: Abstract, Orientation, Complicating Actions, The Most Reportable Event, Evaluation, Resolution and Coda. Labov argued that the entire story's purpose is to serve the MRE, which is "the event that is less common than any other in the narrative and has the greatest effect upon the needs and desires of the participants in the narrative (is evaluated most strongly)" (Labov 1997, p. 406).

Ouyang and McKeown (2015) went a step further by merging L&W with Prince's (1973) three basic states: the starting state, the ending state, and the transformational event in the middle. Hence, they provided a slightly modified definition for MRE as "the most unusual event that has the greatest emotional impact on the narrator and the audience". O&M further note the Orientation is the starting state and the Resolution is the ending state.

However, we have not yet found a correspondence for Rising and Falling Actions in L&W's framework. In Labov's scheme (1997, 2013) the Complicating Action is any event in a causal sequence, of which the MRE is one. Here we apply an additional requirement that a complicating action must cause the tension to rise, and must make the MRE causally possible. That is, it must cause something to become more complicated, as the name implies.

We also deviate from O&M by aligning Falling Action with Resolution and Dénouement with Coda. Labov defines Coda in terms of its ability to resolve all further questions the audience may have, "so that the question: 'What happened then?' is no longer appropriate" (Labov 1997, p. 402). We understand this as that a new equilibrium has been established and dramatic tension is minimum.

L&W's Abstract and Evaluation do not have corresponding functions in Freytag and others. We attribute this to difference in medium and context – L&W focused on oral stories, which are usually short and less formal than professional productions; the relationship between the storyteller and the listener is usually close. As such, L&W's Abstract draws attention from the listener and signals the following story. An example is when a friend calls and says, "I just had the most amazing experience at the park!" Evaluation usually provides a personal viewpoint from the storyteller, such as "That's why I avoid that restaurant." This type of message is rare in formal narrative productions, except perhaps for children's stories and fables.

The deliberate nature of the above discussion is to discover commonalities among narratological theories that appear different, at the possible risk of not being meticulously faithful to their authors' intentions. We believe this approach provides important insights, especially for operationalizing the theories into a practical annotation schema, which we present below.

3.2 The Proposed Annotation Schema

Based on the above theoretical analysis, we present an operationalized theory in terms of narrative functions that we use to label stories. A key practical consideration is to reduce ambiguity in the definitions, so the schema can be easily communicated and the number of ways that a story may be annotated is reduced. Here we describe the 10 functional labels.

A central idea throughout these 10 categories is the *story frame*. We consider events recounted as part of the story as within the story frame. From time to time, the narrator may step outside the frame to reflect on the tale's meaning or connect with the audience. Labov also observed two modes of engagement – one socially-oriented and another in which the speaker is "reliving events of his past" (Labov, 1972, p. 354).

Abstract: An abstract is a summarizing account of the key ideas in the tale, and is almost always found at the beginning of the text. Although it contains information about the story (including the gist of the MRE), it does not introduce the inciting action and thus sits outside the story frame. This label can also apply to a story title.

Orientation: This is the starting state of the story and thus, like the other stages that successively follow, it sits within the story frame. The orientation consists of a survey of the elements that set up the central action, which may include "time, place, persons and their activity or situation' (Labov, 1972, p. 364). It may also include general tendencies of a person or situation, such as "my brother is usually very healthy" or "my house is always cold".

Complicating Action: In general, a complicating action is a single event that increases the tension of the story. It also causes a situation to turn away from everyday routines and become remarkable. Finally, it has a causal component, in that it propels the critical action of the story towards the MRE. We use this label multiple times to indicate a series of complicating actions that build tension with each occurrence.

The Most Reportable Event: This is an event that introduces tension, in the same manner as a complicating action, but it also has some unique qualities that means there can only be one in a tale. A sentence or sentences qualify as an MRE if two criteria are fulfilled: (1) it is an explicit event at the highest tension point of the story. (2) If you only report one event as the summary of the story, it is this one.

Minor Resolution: This is an explicit event that allows tension to drop slightly during a series of complicating actions. It can occur in two ways: (1) by resolving a lesser mystery in a story, or part of it; (2) by resolving the tension of part of a problem in the story, without resolving the issues of the entire narrative.

Return of MRE: If the MRE theme comes back later after the resolution in a new way, either in time or in action, we say it is a 'Return of MRE'. This event is a new twist on the main theme. It must be at similar level of tension and importance as the MRE; it is also separated from the MRE by time or other narrative functions (if not, it is simply the same event as the MRE). On the tension curve, the Return of MRE allows the tension to rise again after the Resolution.

Story Text	Annotation
Yes. A hedgehog saved my life when I was at college.	Abstract
I used to work nights and cycle home at 4 am. There was a long decline, then a bend in the road, then an incline. I would usually cycle around the bend full speed, and rely on my hearing and the glare of headlights to see if anyone was coming.	Orientation
One night I was cycling home with headphones in as I'd had a bad shift, and going along full speed I saw a hedgehog in the road, and braked just before the bend. Usually I would have just dodged the hedgehog, but the song I was listening to just made me feel like stopping and watching it for a second.	Complicating Action
Just as I pulled up, a truck came around that bend at that moment full speed, taking the racing line with its headlights off. I would have not have [sic] even felt a thing.	MRE
The hedgehog was killed.	Resolution

Table 2. An annotated story example with a canonical story structure from Abstract to Resolution.

Resolution: This event on the main causal chain happens after the MRE and resolves the dramatic tension of the story. Hence, it is often a concluding action of the story, but can be followed by the Aftermath or the Evaluation.

Aftermath: This event occurs when a significant temporal gap has elapsed after the main event sequence has concluded. It indicates the long-term effect or broader implications of the recounted events – for example, how the story characters went on with their lives after the main events are over.

Evaluation: This is a comment from the narrator about the significance or meaning of the story itself and is focused on a moral, message, value or lesson. It could even be the absence of a lesson, such as "I didn't learn X". The storyteller stops recounting events to the audience and "tell[s] him what the point is" (Labov, 1972, p. 374). It aligns with Labov's notion of "external evaluation" (Labov, 1972, p. 371). This kind of comment occurs outside the story frame and is usually located after resolution or aftermath.

Direct comment to audience: A direct comment openly addresses the audience outside the story frame, for example: "You're not going to believe this." It can also include the reason for telling the story, an apology for the way the story is presented, or concern that telling the story will get the writer into trouble.

3.3 Annotated Examples

Tables 2 and 3 show two story examples annotated by the authors. The two stories were collected from Quora, as described by Wang *et al.* (2017) and contain small spelling and grammatical errors common to online, non-professional contributors.

Story Text	Annotation
Spanish captain was walking on his ship.	Orientation
A soldier rushes to him and says, "An enemy ship is approaching us".	Complicating Action 1
Captain replies calmly, "Go get my red shirt".	Complicating Action 2
The soldier gets the shirt for the captain.	
The enemy ship comes in; heavy rounds of fire are exchanged. Finally, the Spaniards win.	Minor Resolution
Soldier asks, "Congrats Sir, but why the red shirt?"	
Captain replies, "If I got injured, then my blood shouldn't be seen, as I didn't want my men to loose [sic] hope."	MRE
Moral: For success, hope is very important.	Evaluation
Just then, another soldier came in and said, "Sir, we just spotted another 20 enemy ships!" The captain calmly replied, "Now Go bring my yellow pants".. :p :p	Return of MRE

Table 3. An annotated story example showing Evaluation and Return of MRE.

The first story has a canonical structure from Abstract to Resolution. The Orientation provides the background of the story and the storyteller's general tendency to cycle too fast on a curvy road. Strictly speaking, the first half of the next sentence "one night I was cycling home …" can be seen as part of Orientation. However, for consistency we do not allow one sentence to be broken into multiple parts with different annotation labels. The story annotator must decide the focus of the sentence, which is the action of braking. This action is a Complicating Action as it increases tension and is on a causal chain leading to the MRE. The event of the hedgehog being killed is considered Resolution rather than Aftermath because it happens immediately after the MRE.

The second example is likely a retelling of a popular joke. The story's structure is less complete than the first one. In this example, the tension in the story rises again in the end with the Return of MRE and never gets fully resolved. We note this is a common structure in jokes, presumably to surprise the audience and let them figure out the outcome (Li 2016). This story has two Complicating Actions because they are two separate events and each raises tension. This contrasts with the first story's Complicating Action, which has a lengthy description focusing on the single action of braking. The additional complicating actions indicate the greater structural complexity of the second example, in terms of building and managing tension.

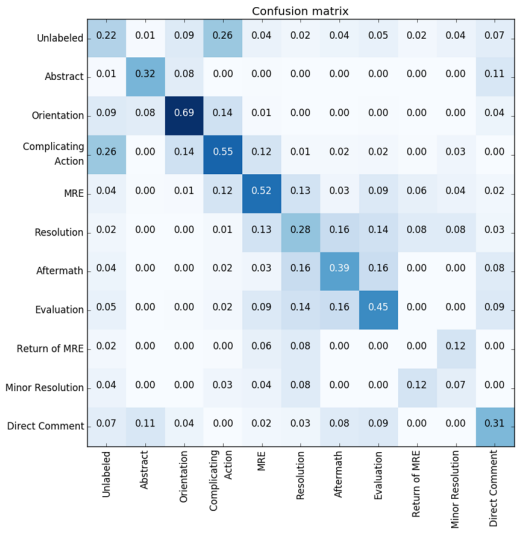


Figure 2. The confusion matrix for narrative functions.

4. The Annotation Procedure

The stories being annotated come from three sources: stories collected from Quora by Wang *et al.* (2017), stories collected from Reddit by O&W, and stories annotated by Lukin *et al.* (2016). For the Quora stories, two annotators determined if a text is a story and fits our purpose. The criteria include: the text must contain an MRE and is composed of only one story (not multiple stories); the text is shorter than 700 words, longer than 90 words and has less than 6 lines of dialogue; non-narrative elements, such as lengthy reflections that do not drive the story forward, must be less than 50% of the text. Stories that do not meet these criteria are rejected. Stories that contain offensive content are not annotated.

The annotation was mainly performed by two annotators who did not have backgrounds in linguistics or literature. They were asked to first read the entire story and pinpoint the most reportable event. With the MRE determined, they subsequently identify complicating actions on causal chains leading to the MRE, and then the Resolution, where the tension is resolved. The rest of the categories can be identified against this skeleton. They then broke up the text into sentences, which are the basic unit of annotation, and assign them to categories. One sentence cannot be broken into multiple parts with multiple labels. The annotators were encouraged to think in terms of events rather than raw text.

We adopt the following training and validation process. The annotators went through three rounds of tutorials over five weeks, and during each session they were given 25 stories to annotate. These annotations were then compared to the gold standard provided by the authors and corrections were explained. After training, the two annotators annotated the same 71 stories in order to compute interrater agreement. Subsequently, they annotated separate stories. One author of this paper also annotated a small number of stories.

In total, 480 stories containing 8,908 sentences were annotated. Excluding repeated stories, we obtain 360 unique stories, including 167 from the Quora dataset, 73 from Lukin *et al.* and 120 from O&W. A story contains 18.34 sentences on average.

5. Validation and Discussion

We computed interrater agreement using Cohen's Kappa between pairs of annotators separately, as not all annotators worked on the same set of stories. The agreement is computed at the sentence level. Among the two annotators and an author, the pair-wise kappas are 0.39, 0.41, and 0.42 respectively, indicating fair agreement among the annotators.

We further analyze the disagreements made by the annotators. Figure 2 shows a detailed confusion matrix among narrative functions and an additional "unlabeled" category. The numbers in the matrix are computed as follows. If annotator A and annotator B agree that a sentence is in category i, the count for the cell (i,i), denoted as c_{ii}, is incremented by 1. If one labels the sentence as category i and the other labels it as category j, the count for cells (j,i) and (i,j) are both increased by 0.5. Finally, the cells are normalized as $2c_{ij}/(\sum_k c_{ik} + \sum_k c_{kj})$.

From Figure 2, we observe that substantial agreement is achieved around the major categories of story structure that appear in all three schemes (ours, Labov's and that of Freytag/Thursby). These are the Orientation, Complicating Action, and MRE. We attribute this high agreement to the observation that the core progressions in a story's structure are usually less ambiguous than the rest.

The three categories close to the end of the story, Resolution, Evaluation, and Aftermath, tended to be mixed up by annotators. Early elements such as Abstract and Orientation also tended to get confused. Although these categories have clear definitions, the differences between them were finer, and in the wild terrain of real-world anecdotes these differences were harder to reliably identify.

This suggests our annotation schema can differentiate major components of the story structure, even though the annotation gets less accurate on the categories that are more specific to particular nuances of story structure. Infrequent categories such as Return of MRE and Minor Resolution are also difficult to annotate. After merging Resolution, Evaluation, and Aftermath into a single category, and treating Minor Resolution and Return of MRE as unlabeled, the three interrater agreement measures increase to 0.44, 0.49, and 0.47, respectively.

We reckon that being able to differentiate major categories across entire narratives is an achievement, especially given the complexity of the annotation scheme and the potential for ambiguity in real-world, non-professionally written texts.

We further note the high cognitive load created by this annotation task. The annotators need to keep the entire story in mind while evaluating each sentence's role in the entire story. In addition, they often need to mentally parse sentences to separate several events and recognize the most important event being described. In the future, the use of intelligent annotation tools could simplify the task and boost interrater agreement. For example, the annotation tool may decompose the annotation of one story into many smaller tasks and reduce the cognitive load. The annotation

tool may also utilize semantic role labeling to highlight different events and help the annotator recognize major events. Reducing the number of categories is also likely to further improve interrater agreement.

6. Conclusions

Understanding the macro structures of a narrative, such as where dramatic tension rises and falls, is an important link in enabling computer systems to understand the larger dynamics of narrative communication. Existing work tends to focus on categories that are specific to one genre and types of stories or a subset of the story structure.

In this paper, we provide a first attempt at integrating multiple narratological accounts to capture fundamental and holistic structures of a story. To do this, we propose a set of narrative functions that represent the overlap between schemata proposed by Labov and Waletzky (1967) and Freytag (1863), thereby capturing dramatic tension and the social aspect of story structure. We annotated 360 unique stories from three story sources in the literature and achieved fair interrater agreement on the annotations. Upon close inspection, we note confusion in the annotations are concentrated on a few fine-grained categories whereas the core stations of story progression were consistently identifiable. The annotation results suggest the annotation scheme allows the separation of major structural elements, despite the difficulty of the task. We believe this research will lead to further progress towards an artificial intelligence that can communicate with human users in the form of stories.

7. Acknowledgements

We gratefully acknowledge Jill F. Lehman for helping with organizing the annotation effort, Susi Burger and Kory Mathewson for valuable discussions.

8. Bibliographical References

Appling, D. S., & Riedl, M. O. (2009). Representations for Learning to Summarize Plots. In *Proceedings of the AAAI Spring Symposium on Intelligent Narrative Technologies II*.

Aristotle. (circa. 335 BC). *Poetics*.

Bamman, D., O'Connor, B., & Smith, N. A. (2013). Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*.

Blei, D., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*.

Bruner, J. (1986). *Actual minds, possible worlds*. Cambridge, MA: Harvard University Press.

Campbell, J. (1949). *The Hero with a Thousand Faces*. Bollingen Foundation.

Chaturvedi, S., Peng, H., & Roth, D. (2017). Story Comprehension for Predicting What Happens Next. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Dogan, P., Li, B., Sigal, L., & Gross, M. (2018) LSTM stack-based Neural Multi-sequence Alignment TeCHnique (NeuMATCH). In Proceedings of *the Conference on Computer Vision and Pattern Recognition.* In Press.

Elsner, M. (2015). Abstract representations of plot structure. *Linguistic Issues in Language Technology*, *12*.

Elson, D. K. (2012). DramaBank: Annotating Agency in Narrative Discourse. In *Proceedings of the International Conference on Language Resources and Evaluation*.

Ferraro, F., & Van Durme, B. (2016). A unified Bayesian model of scripts, frames and language. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*.

Finlayson, M. A. (2016). Inferring Propp's Functions from Semantically Annotated Text. *Journal of American Folklore*, *129*(511), 53–75.

Fisher, W. (1987). *Human communication as narration: Toward a philosophy of reason, value and action.* Columbia, SC: University of South Carolina Press.

Freytag, G. (1863). *Die Technik des Dramas*.

Gervás, P., Hervás, R., León, C., & Gale, C. V. (2016). Annotating Musical Theatre Plots on Narrative Structure and Emotional Content. In *Proceedings of the Seventh International Workshop on Computational Models of Narrative*.

Goyal, A., Riloff, E., & Daume III, H. (2010). Automatically Producing Plot Unit Representations for Narrative Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Huang, T.-H. K., Ferraro, F., Mostafazadeh, N., Misra, I., Devlin, J., Agrawal, A., Girshick, R., He, X., Kohli, P., Batra, D., Zitnick, L., Parikh, D., Vanderwende, L., Galley, M., & Mitchell, M. (2016). Visual Storytelling. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.*

Labov, W. (1997). Some further steps in narrative analysis. *Journal of narrative and life history*, (7), 395–415.

Labov, W. (2013). *The Language of Life and Death*. Cambridge University Press.

Labov, W., & Waletzky, J. (1967). Narrative analysis. In *Essays on the Verbal and Visual Arts*. Seattle, WA: University of Washington Press.

Labov, William. (1972). The Transformation of Experience in Narrative Syntax. In *Language in the Inner City: Studies in the Black English Vernacular* (pp. 354–397). Philadelphia: University of Philadelphia Press.

Lehnert, W. (1981). Plot Units and Narrative Summarization. *Cognitive Science*, *4*, 293–331.

Li, B. (2016). Humor: A Dynamic and Dual-Process Theory with Computational Considerations. *Advances in Cognitive Systems*.

Lukin, S. M., Bowden, K., Barackman, C., & Walker, M. A. (2016). PersonaBank: A Corpus of Personal Narratives and their Story Intention Graphs. In *Proceedings of the International Conference on Language Resources and Evaluation*.

Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., Kohli, P., & Allen, J. (2016). A Corpus and Evaluation Framework for Deeper Understanding of Commonsense Stories. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.*

Ouyang, J., & McKeown, K. (2015). Modeling reportable events as turning points in narrative. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Pichotta, K., & Mooney, R. J. (2016). Learning statistical scripts with LSTM recurrent neural networks. In *Proceedings of the 30th AAAI Conference on Artificial*

Intelligence.
Prince, G. (1973). *A Grammar of Stories: An Introduction*.
Propp, V. Y. (1928). *Morphology of the Folktale*.
Rahimtoroghi, E., Swanson, R., Walker, M. A., & Corcoran, T. (2013). Evaluation, orientation, and action in interactive storytelling. In *Proceedings of Intelligent Narrative Technologies 6*.
Rao, A. S., & Georgeff, M. P. (1995). BDI-agents: From theory to practice. In *Proceedings of the First International Conference on Multiagent Systems*.
Reagan, A. J., Mitchell, L., Kiley, D., Danforth, C. M., & Dodds, P. S. (2016). The emotional arcs of stories are dominated by six basic shapes. *EPJ Data Science*, 5(1).
Riedl, M. O., & Young, R. M. (2010). Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*.
Swanson, R., Rahimtoroghi, E., Corcoran, T., & Walker, M. (2014). Identifying Narrative Clause Types in Personal Stories. Presented at the Annual SIGdial Meeting on Discourse and Dialogue.
Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A. and Urtasun, R., & Fidler, S. (2016) MovieQA: Understanding Stories in Movies through Question-Answering. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*.
Thursby, Jacqueline. (2006). *Story: A Handbook*. Greenwood Publishing Group.
Todorov, T. (1971). The Two Principles of Narrative. *Diacritics*, *1*(1), 37–44.
Wang, T., Chen, P., & Li, B. (2017). Predicting the quality of short narratives from social media. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*.

Discovering the Language of Wine Reviews: A Text Mining Account

Els Lefever[1], Iris Hendrickx[2], Ilja Croijmans[2], Antal van den Bosch[2,4], Asifa Majid[2,3]
[1]Language and Translation Technology Team, Ghent University, Belgium
[2]Centre for Language Studies, Radboud University, Nijmegen, The Netherlands
[3]Max Planck Institute for Psycholinguistics, Nijmegen, The Netherlands
[4]Meertens Institute, Amsterdam, The Netherlands
els.lefever@ugent.be, {i.hendrickx,i.croijmans,a.vandenbosch,asifa.majid}@let.ru.nl

Abstract

It is widely held that smells and flavors are impossible to put into words. In this paper we test this claim by seeking predictive patterns in wine reviews, which ostensibly aim to provide guides to perceptual content. Wine reviews have previously been critiqued as random and meaningless. We collected an English corpus of wine reviews with their structured metadata, and applied machine learning techniques to automatically predict the wine's color, grape variety, and country of origin. To train the three supervised classifiers, three different information sources were incorporated: lexical bag-of-words features, domain-specific terminology features, and semantic word embedding features. In addition, using regression analysis we investigated basic review properties, i.e., review length, average word length, and their relationship to the scalar values of price and review score. Our results show that wine experts do share a common vocabulary to describe wines and they use this in a consistent way, which makes it possible to automatically predict wine characteristics based on the review text alone. This means that odors and flavors may be more expressible in language than typically acknowledged.

Keywords: wine reviews, wine vocabulary, classification, supervised learning, terminology extraction

1. Introduction

Few food categories are described as often as wine: a vast number of wine reviews appear in magazines, books, blogs, supermarkets, newspapers, and numerous other venues. These reviews contain descriptors, which chronicle the appearance, aroma (smell), flavor, and textural attributes of wines in loving detail. This is puzzling, as previously many scholars have claimed that smells and flavors are difficult, if not impossible to put into words (Sperber, 1975; Rouby et al., 2002; Levinson and Majid, 2014).

There is an ongoing debate as to whether wine is actually described in an informative manner in wine reviews. On the one hand, studies suggest wine experts use language in a consistent manner (Croijmans and Majid, 2016). Similarly, expert descriptions are more often correctly matched to a wine than descriptions written by novices (Solomon, 1990), with the suggestion that wine experts are more likely to use specific terminology which is more informative. On the other hand, studies suggest trained wine experts more often use vague and abstract terms (e.g., *complex, attractive*) when describing wines (Gawel, 1997). Similarly, metaphorical language is often encountered in wine descriptions (Suárez Toste, 2007; Caballero, 2007), suggesting wine experts employ vague and overly literary prose. Others have more directly critiqued wine reviews as being uninformative and mere "purple prose" (Quandt, 2007) (p.130).

Here, we study how wine experts express properties of wines in their reviews. Wine reviews convey both sensory descriptions of wines, as well as objective properties such as color, grape type, country of origin, and price; and reviews also convey an overall rating as to the quality of the wine. Previously, we showed that experts are indeed consistent in their descriptions and we were able to train a classifier on review texts to predict objective wine properties (color, country, grape, price) (Hendrickx et al., 2016). In the present investigation we expand this work in two directions.

First, we examined the usefulness of domain-specific terminology as feature representations for classification tasks. To investigate the terminological consistency in wine reviews, we set up a machine learning experiment to automatically predict the color, grape type, and country of origin of wines based on the information contained in review texts. To this end, we used a corpus of online wine reviews and their structured metadata and extracted three types of information from the review text: a set of lexical bag-of-words features, a set of domain-specific terminological features, and a set of semantic word embedding cluster features.

Second, we investigated a non-textual subjective property assigned to the wine, namely the rating that was given by the expert who wrote the review. We wished to establish whether there was a correlation between prices and ratings, and whether the rating also influenced the review text. Previous work suggests that wine experts (contrary to laypersons) prefer more expensive wines over cheaper ones (Goldstein et al., 2008). Furthermore, more expensive wines are described with longer reviews, measured in the total number of characters in the whole review (Ramirez, 2010). Another way in which price and rating may be reflected in the review may be in the average word length of the words used: More expensive wines may be described using "more expensive" words, i.e., longer words, for example. We performed a regression analysis to replicate the finding by Ramirez (2010) and to further explore the relationship between subjective ratings and price, and the length of wine reviews and average word length in the reviews.

The remainder of this paper is organized as follows: in Section 2 we give an overview of related research, while in Section 3 we provide details about our corpus of wine expert reviews. Section 4 describes the experimental setup and results of our classification experiments to predict wine color,

type, and country of origin. Section 5 presents the regression analysis where we focus on the aspects of price and rating. Section 6 summarizes our main findings.

2. Related research

A number of studies have covered related territory. For this abstract we briefly review some of the most pertinent studies.

Brochet and Dubourdieu (2001) carried out a lexical analysis of four corpora of wine tasting comments by performing χ^2 calculations for all word co-occurrences in the text. The resulting lexical fields were not organized along sensory dimensions only, but contained a mix of visual (*yellow*), olfactory (*apricot*), taste (*sweet*) and hedonistic (*good*) terms, among others, which appears to contradict reports of professional tasters who say they taste wine in an "analytic" manner. Moreover, the fact that word groups combine visual, olfactory and taste descriptors, support the idea that wine language is organized around wine "prototypes". Indeed, Solomon (1997) showed that features identified by experts significantly co-vary with grape types such that wines of the same grape are described more similarly by experts, suggesting these prototypes may be real.

Wine reviews also often feature ratings of wines (i.e., a numerical score). This score can be seen as a subjective expression of the quality of a wine (Oczkowski, 2016). Price, on the other hand, may be seen as a more objective reflection of a wine's quality, determined by several factors not influenced by the reviewer, but rather by other quality measures such as the growing season average temperature and rainfall (Oczkowski, 2016). Even though the relationship between price and quality is not always one-to-one (Goldstein et al., 2008), price may nevertheless give an accessible, rough approximation of quality, in addition to the more subjective expert rating. A wine review may reflect these aspects of quality of a wine in a number of ways: more expensive wines may be described using more words on average, for example (Ramirez, 2010).

The studies reviewed above exemplify a rich research tradition using statistical analysis of wine review corpora. However, there are few studies that have applied natural language processing (NLP) techniques to such data. To date, the research that has used NLP or data mining techniques of wine data have focused on machine learning applied to data containing information about the chemical components of wine (for example (Urtubia et al., 2007; Cortez et al., 2009)).

A previous study of ours did use NLP methods (Hendrickx et al., 2016), and asked whether wine experts use consistent terminology to describe wine, and if by consequence expert reviews contain enough information to automatically predict wine color, grape type, and country of origin by means of supervised classification techniques. In this paper, we take this work forward and examine *how* experts use consistent terminology to describe wine, and investigate the contribution of different types of information, viz. lexical, semantic and terminological feature groups. Our research differs from previous studies (such as (Brochet and Dubourdieu, 2001)) where corpus-based statistical analyses were used to examine wine language; whereas we train machine learning algorithms on a large corpus of expert wine reviews to automatically predict various characteristics. In addition, we investigate the relationship between subjective and objective wine quality indicators and basic review properties, i.e., the length of the review and average length of the words used in the review.

3. Corpus Description

We collected a corpus of wine reviews from `http://www.winemag.com/`, containing in total 76,410 unique reviews from 33 experts. These wine reviews are combined with additional information about the wines such as the name of the producer, production year, alcohol percentage, color, grape type(s), origin, and rating by the expert. These ratings vary between 80 and 100. The reviews are rather short, on average 39 (untokenized) words per review, and often these reviews combine a sensory description with some additional information, such as the producer or region. The following is an example of a review about an red wine produced in Italy in 2009 with a price of 45$ and which was rated 91 out of 100:

> Cantina del Pino makes some of the finest Barbaresco available today. This shows a succulent quality, with aromas of smoked bacon, wild berries and forest underbrush. Savory and sophisticated, this has loads of personality.

Take, for example, another review of a Spanish red wine with a low rating of 80 and a price of 17$:

> Best on the nose, but sharp and narrow as can be on the palate. Cranberry and sour cherry flavors dominate, while the finish is astringent. No amount of swirling and saving is going help it much.

Note that not all reviews had all metadata fields filled. In our experiments we only use those reviews for which we had non-zero values for the class to be predicted. So, for example the classification experiments excluded the 5,308 reviews of wines where the color was unknown.

4. Classification

We aim to study the usefulness of domain-specific terminology as feature representation for predicting wine properties on the sole basis of the wine review text. We experiment with different feature representations and we compare the terminology features against lexical bag-of-words features, and semantic word embedding features. To operationalize the task of automatically detecting objective wine characteristics, we build supervised machine learning systems and aim at predicting three wine characteristics: color, grape variety, and country of origin.

4.1. Experimental setup

The corpus was randomly split into a training (80%) and test (20%) partition. As evaluation measures, we report averaged micro F-scores on the held-out test set.

To predict the color of a wine, we limit ourselves to three categories: *white*, *red* and *rosé*. In order to have a better

understanding of the predictive power of this classifier, we removed color adjectives referring to the three color classes from the review text. To train a system predicting grape variety, we selected wines produced from a single grape with at least 200 reviews in the training set and removed all grape blends. We also merged grape names referring to the same grape (e.g., *Pinot Gris* and *Pinot Grigio*), which resulted in a total of 28 grape varieties to be predicted. Much variation can be seen in the number of training instances per grape, ranging from 5,706 reviews for *chardonnay* to 222 reviews for *carmenère*. The third classifier aims at predicting among 47 different countries of origin. Again, the class distribution is unbalanced, with some countries represented very well (e.g., US: 25,104 reviews, Italy: 9,912 reviews, France: 8,568 reviews) to countries only occurring once (Tunisia, South Korea, Montenegro, India) in the training set.

All wine reviews were linguistically preprocessed by means of the Stanford toolkit (Manning et al., 2014) involving tokenization, lemmatization and Part-of-Speech tagging. From the preprocessed review text, three different feature types were extracted to model the three classification tasks: lexical, semantic and terminology features.

4.1.1. Lexical features

We extracted a list of bag-of-words (BoW) unigram features from the review text containing lowercased lemmas. These BoW features were filtered on Part-of-Speech category to filter out function words and only keep content words (nouns, adjectives, verbs, and adverbs). The BoW features were incorporated as binary features, meaning that each BoW is a separate feature, which is assigned the value "1" if it occurs in the respective wine review, and "0" otherwise.

4.1.2. Semantic features

In order to reduce data sparsity, we also created word embeddings from the training reviews by means of Word2Vec (Mikolov et al., 2013). After traing, for any input word Word2Vec is able to produce a word vector containing distributional information, i.e., information about the surrounding lexical contexts in which the word occurs.

cluster size	Accuracy
100 clusters	92.398%
200 clusters	94.674%
300 clusters	**95.843**%
500 clusters	95.018%
1000 clusters	95.391%
2000 clusters	95.177%

Table 1: Cross-validation accuracy for a varying number of cluster sizes.

Word2Vec was run with standard settings, that is the BoW model with a context size of 8, and a word vector dimensionality of 200 features. To group word vectors for words that share common contexts in the wine reviews, and thus are located in close proximity in the vector space, we clustered the obtained word vectors using a K-means clustering algorithm. We then encoded the resulting clusters as binary features that were activated if the review text contained a word occurring in the respective clusters. To decide on the desired number of output clusters, we performed 10-fold cross-validation experiments on the training data with a varying number of cluster features (100, 200, 300, 500, 1000 and 2000 clusters). Table 1 shows the accuracy for the cross-validation experiments with varying cluster sizes. A manual inspection of the resulting clusters revealed that the clusters indeed contain semantically related terms. This is illustrated by cluster 82, which contains many terms referring to floral and other related aromas:

> abundant, acacia, aromatic, bee's, clover, dandelion, delicate, enticing, floral, flower, foremost, fragrant, freesia, fresh-cut, freshly, fuzz, garden, jasmine, light-weight, lilac, musk, oils, peony, petroleum, pretty, roses, rosewater, subtle, talcum, wax, wisp, wispy

4.1.3. Terminology features

As a third feature group, we extracted domain-specific terms from the wine review corpus. Terms are linguistically motivated units that refer to concepts within a given domain. The wine-specific terms were extracted by means of TExSIS (Macken et al., 2013), a hybrid terminology extraction tool. In a first step, linguistic preprocessing (Van de Kauter et al., 2013) is run on the wine corpus to perform tokenization, lemmatization, Part-of-Speech tagging, chunking, and named entity recognition. Subsequently, TExSIS makes use of this linguistic information to generate syntactically valid candidate terms. In a final step, statistical filters such as Termhood and C-value are applied to generate the list of single (e.g. *flavor, cherry, ripe, spice, finish*) and multi-word (e.g. *cherry fruit, berry flavors, firm tannins, black currant*) terms.

The underlying idea of the *termhood* filtering is that domain-specific terms (e.g. *tannin, nose*) have much higher relative frequencies in the domain-specific wine corpus than in a standard corpus of English, the Web 1T 5-gram v1 corpus[1] in this case. An inspection of the top-10 terms extracted by TExSIS with the highest termhood scores (i.e. *flavors, tannins, aromas, wine, acidity, fruit, palate, finish, off-dry, cherry*) reveals that these terms indeed belong to specialized wine vocabulary. Although some of these terms also occur in common language, they are much more frequent in the wine corpus than in a general background corpus.

The second statistical filter, the *C-value* filter (Frantzi and Ananiadou, 1999), checks the degree of cohesiveness inside multi-word terms. The C-value metric aims at handling the extraction of nested terms by examining the frequencies of a term used as part of a longer term. Examples of TExSIS multi-word terms with high C-value scores are: *black cherry, fruit flavor, Cabernet Sauvignon, pinot noir, tropical fruit, crisp acidity, dark chocolate* and *smoky oak*.

The resulting terms were again incorporated as binary features in the feature vector. For each review, we extracted

[1] https://catalog.ldc.upenn.edu/ldc2006t13

the list of unigrams, bigrams, trigrams and fourgrams, and set the binary features to '1' if the respective term occurs in one of the n-gram lists. In order to have a comparable amount of bag-of-words and terminological features, we included the same number of features for both feature groups. For the terminological features, we therefore considered the 15,357 most domain-specific terms, sorted by termhood.

4.1.4. Machine learning algorithm

As a classification algorithm, we used Support Vector Machines as implemented in the LIBSVM toolkit (Chang and Lin, 2011). We ran LIBSVM in three different settings: (1) RBF Kernel with standard settings, (2) RBF Kernel with optimized settings (by means of a Grid search, performed on a sample (5,000 randomized instances) of the training data for each classification task, and (3) Linear kernel with standard settings.

When building the feature vectors, we removed reviews where the respective category label was missing, resulting in a training and test set of varying size per classification task. Table 2 lists the number of instances per training and test for each task, as well as the number of categories to be predicted, which gives a good indication of the classification difficulty.

classification task	training	test	categories
colour	56,893	14,209	3
grape type	39,900	9,976	28
country	61,128	15,282	47

Table 2: Size of the training and test sets and number of categories to be predicted for each classification task.

4.2. Classification results and discussion

Table 3 lists the results for the LIBSVM Linear Kernel and optimized RBF Kernel for a varying feature vector. We show experimental results for the different feature groups in isolation and for a combination of all three information sources.

The results of the different feature representations in isolation show that for the grape variety and country classification experiments, the domain-specific terminology features selected by TExSIS outperform the BoW and Word2Vec features. For color however, we see that a simple BoW gives the best performance. In addition, it appears that the choice of SVM kernel does not have a huge effect, as each column shows similar tendencies. When we combine the three feature representations, we again see that this is beneficial for the performance on the grape variety and country classification experiments, but not for the color experiments where BoW features perform best.

We also investigated which domain-specific terms were most informative for the *country* and *variety* classification tasks. To this end, we calculated Information Gain (IG) weighting, which measures for an individual feature how much information it contributes to predicting the correct class label. This was done by computing a probability-weighted average of the informativeness of the different values of the feature, with the IG implementation provided by Timbl (Daelemans et al., 2009). We contrasted an analysis of the best performing terminology features with the most informative lexical descriptors from the bag-of-words feature set.

Setup	RBF opt	Lin Kernel
	Color	
BoW	96.75%	96.59%
Word2Vec	96.31%	96.18%
TExSIS	93.49%	91.66%
All features	96.09%	95.29%
	Grape Variety	
BoW	42.10%	48.28%
Word2Vec	57.39%	56.46%
TExSIS	72.53%	72.77%
All features	76.16%	76.61%
	Country	
BoW	66.50%	60.32%
Word2Vec	69.17%	68.27%
TExSIS	78.67%	79.06%
All features	82.27%	82.84%

Table 3: Averaged micro F-scores per category for the optimised RBF (*RBF opt*) and Linear (*Lin Kernel*) kernels.

Table 4 lists the top-25 most informative terminology (TExSIS) and bag-of-words (BoW) features for both classification tasks.

Country TExSIS	Variety TExSIS	Country BoW	Variety BoW
fruit	aromas	aroma	aroma
it	acidity	palate	tannin
aromas	cherry	acidity	acidity
finish	tannins	cherry	dry
palate	palate	tannin	palate
acidity	ripe	note	note
tannins	dry	ripe	ripe
has	has	spice	spice
cherry	dried-herb	drink	drink
ripe	oak	dry	oak
drink	drink	nose	sweet
notes	notes	black	not
spice	rich	berry	rich
nose	black	not	black
black	spice	sweet	blackberry
fresh	soft	plum	show
sweet	very	fresh	berry
rich	nose	show	now
berry	blackberry	rich	good
dry	shows	red	soft
red	fresh	oak	very
plum	berry	soft	nose
soft	red	good	year
oak	crisp	now	fresh
very	vanilla	blackberry	red

Table 4: Top-25 most informative terminology and bag-of-words features sorted by descending Information Gain scores for the *country* and *grape variety* tasks.

A first observation is that the Information Gain analysis re-

veals a big overlap between the most informative descriptors (1) for the two classification tasks, i.e. *country* and *variety* (overlap of 20 TExSIS and 23 BoW features) and (2) for the two different features groups, i.e., terminology and bag-of-words features (overlap of 20 for *country* and 22 for *variety*). Second, the most informative descriptors do not contain explicit mentions of the class labels (i.e., grape types, countries), but rather contains odour (e.g., *aroma*), flavour (e.g., *sweet, tannin*) and sight (e.g., *red*) descriptors. In addition, they contain source-based (e.g., *plum, oak*), as well as evaluative terms (e.g., *good*), verbs related to the wine tasting process (e.g., *finish, drink*), and more general vocabulary (e.g., *show*).

5. Correlation between price, score and text characteristics

Another aspect we were interested in was how strong the relationship was between basic level properties of the review (i.e., the length of the review in number of characters and the average length of words used), and the objective price of the wine and subjective rating given by the reviewer. To estimate this relationship, two regression models were run on the data: one predicting the price from word length and review length, and one predicting review score from word length and review length.

Average word length per review was established by dividing the amount of characters by the number of words. Reviews with an unknown score or price (standardized to dollars per 0.75L) were removed from the data set, resulting in 67,006 remaining reviews. Analyses were performed using R (R Core Team, 2013). The means and standard deviations of the different review features (price, score, average word length, and review length) can be found in Table 5.

feature	mean	SD
price	33.3	47.9
rating	87.6	3.2
review length	235.7	71.2
word length	6.0	0.4

Table 5: Means and standard deviations for the review features price in dollars, review score in points, and review length and average word length measured in characters.

	Estimate β	SE	t-value	p
Intercept	-12.98	2.74	-4.7	$< .001$
Av word length	2.61	0.45	5.8	$< .001$
Review length	.13	.003	50.9	$< .001$

Table 6: Predicting price from average word length and review length.

The regression model for review price and rating for all 67,006 reviews which had non-zero values are shown in Table 6 and Table 7 respectively. Review length and average word length were significantly related to the price of the wine, in such a way that for every additional character in a review, the wine was on average 13 cents more expensive, and the use of longer words by one character predicted an extra 2.6 dollars on top of the wine price, on average. Similarly, the review score was significantly correlated with review length and word length. For each additional character in the review, the score of the wine increased by .79 points. Average word length in a review had less influence, as the use of one-character longer words increased the score by only .025 points.

	Estimate β	SE	t-value	p
Intercept	7.76	.164	479.3	$< .001$
Av word length	.791	.027	29.5	$< .001$
Review length	.025	.002	149.6	$< .001$

Table 7: Predicting review rating score from average word length and review length.

6. Conclusion

This paper describes a set of classification and regression experiments aimed at predicting wine characteristics based on the review text. The results show that (1) wine experts indeed share a common vocabulary, making it possible to predict the color, grape variety and country of origin of the wine to a reasonable extent, and that (2) terminological features outperform bag of word features and semantic features when used in isolation. In addition, review length and average word length were shown to be significantly related to review price and rating.

In sum, this study shows that the language of wine reviews is richly informative (contra previous claims), and demonstrates the important role of NLP methods to address core questions about the limits and possibilities of language more generally.

7. Bibliographical References

Brochet, F. and Dubourdieu, D. (2001). Wine descriptive language supports cognitive specificity of chemical senses. *Brain and Language*, 77:187–196.

Caballero, R. (2007). Manner-of-motion verbs in wine description. *Journal of Pragmatics*, 39(12):2095–2114.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27:1–27:27. ISSN: 2157-6904.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553.

Croijmans, I. and Majid, A. (2016). Not all flavor expertise is equal: The language of wine and coffee experts. *PLoS ONE*, 11(6):e0155845.

Daelemans, W., Zavrel, J., Sloot, K., and Van Den Bosch, A. (2009). TiMBL: Tilburg Memory-Based Learner, version 6.2, Reference Guide. ILK Technical Report 09-01.

Frantzi, K. and Ananiadou, S. (1999). The C-value / NC-value domain independent method for multi-word term extraction. *Journal of Natural Language Processing*, 6:145–179.

Gawel, R. (1997). The use of language by trained and untrained experienced wine tasters. *Journal of Sensory Studies*, 12(4):267–284.

Goldstein, R., Almenberg, J., Dreber, A., Emerson, J. W., Herschkowitsch, A., and Katz, J. (2008). Do more expensive wines taste better? evidence from a large sample of blind tastings. *Journal of Wine Economics*, 3(1):1–9.

Hendrickx, I., Lefever, E., Croijmans, I., Majid, A., and van den Bosch, A. (2016). Very quaffable and great fun: Applying nlp to wine reviews. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 306–312, Berlin, Germany, August. Association for Computational Linguistics.

Levinson, S. C. and Majid, A. (2014). Differential ineffability and the senses. *Mind & Language*, 29(4):407–427.

Macken, L., Lefever, E., and Hoste, V. (2013). TExSIS: Bilingual Terminology Extraction from Parallel Corpora Using Chunk-based Alignment. *Terminology*, 19(1):1–30.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and Mc-Closky, D. (2014). The stanford corenlp natural lan- guage processing toolkit. In *Proceedings of the Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60. Association for Computational Linguistics.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems*, 26:3111–3119.

Oczkowski, E. (2016). Identifying the effects of objective and subjective quality on wine prices. *Journal of Wine Economics*, 11(2):249–260.

Quandt, R. E. (2007). On wine bullshit: some new software? *Journal of Wine Economics*, 2(02):129–135.

R Core Team, (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ramirez, C. (2010). Do Tasting Notes Add Value? Evidence from Napa Wines. *Journal of Wine Economics*, 5(1):143–163.

Rouby, C., Schaal, B., Dubois, D., Gervais, R., and Holley, A. (2002). *Olfaction, taste, and cognition*. Cambridge University Press.

Solomon, G. (1990). Psychology of Novice and Expert Wine Talk. *The American Journal of Psychology*, 103(4):495–517.

Solomon, G. (1997). Conceptual Change and Wine Expertise. *The Journal of the Learning Sciences*, 6(1):41–60.

Sperber, D. (1975). *Rethinking symbolism*. Number 11. CUP Archive.

Suárez Toste, E. (2007). Metaphor inside the wine cellar: On the ubiquity of personification schemas in winespeak. *Metaphorik. de*, 12(1):53–64.

Urtubia, A., Pérez-Correa, J. R., Soto, A., and Pszczolkowski, P. (2007). Using data mining techniques to predict industrial wine problem fermentations. *Food Control*, 18(12):1512–1517.

Van de Kauter, M., Coorman, G., Lefever, E., Desmet, B., Macken, L., and Hoste, V. (2013). LeTs Preprocess: The multilingual LT3 linguistic preprocessing toolkit. *Computational Linguistics in the Netherlands Journal*, 3:103–120.

Toward An Epic Epigraph Graph

Francis Bond, Graham John Matthews
Linguistics and Multilingual Studies, English
School of Humanities
Nanyang Technological University
www3.ntu.edu.sg/home/fcbond/
www.soh.ntu.edu.sg/Programmes/english/FacultyandStaff/Pages/Graham-Matthews.aspx
bond@ieee.org, gmatthews@ntu.edu.sg

Abstract

We present a database of epigraphs collected with the goal of revealing literary influence as a set of connections between authors over time. We have collected epigraphs from over 12,000 literary works and are in the process of identifying their provenance. The database is released under an open license.

Keywords: epigraph, influence, digital humanities

The best ideas are common property.

Seneca the Younger,
"On Old Age", Moral Letters to Lucilius

1. Introduction

An epigraph is a quotation at the beginning of some piece of writing. Genette (1987, Ch. 7, pp156–160) identifies four major uses: (i) commenting on and justifying the title of the work; (ii) commenting on the text of the work (the most canonical usage); (iii) claiming a relationship to the cited author (name dropping); (iv) signaling the genre of the work (romantic and gothic authors use more epigraphs, classical and realistic fewer). As a results authors use them both to set the theme and to link their work to the existing body of literature.

We have built a database of epigraphs, as a preliminary step to creating a network which maps epigraphs and their provenance (author, work, date, and country of origin) and consequently reveals literary influence as a set of connections between authors over time.

Epigraphs are indicative of the intended audience: for example, Evelyn Waugh uses epigraphs written in Ancient Greek, which suggest that the reader is expected to have some familiarity with the Classics, while Ali Smith utilises epigraphs from around the world, suggesting a more cosmopolitan readership. Most importantly however, they function as a clear marker of influence. From their choice of epigraph, we know that writers such as Jeanette Winterson and Margaret Atwood were influenced by Greek tragedy, Ian McEwan was influenced by Jane Austen, and that Roberto Bolaño's influences range from Malcolm Lowry to Petronius. Reading literature as a communal network of ideas, philosophies, and artistic practices rather than a set of discreet objects signals the ways in which societies, institutions and structures of power represent themselves in art and language and are shaped in turn by representations. Rather than focusing attention on the idiosyncratic group of writers that comprise the canon, this project examines the larger patterns that shape entire literary forms and periods. For instance, it is commonly understood that the writers of the English Renaissance were heavily influenced by the recently re-discovered Greek classics yet preliminary studies of the choice of epigraphs suggest that the Ancient Greeks were just as much, if not more, of an influence on British writers from the 1950s and 1960s, a notion that is born out by closer inspection of the novels' themes.

Epigraphs are often discussed within wider literary criticism as a way authors explicitly show their influences (Poplawski, 2017). However, to date there has been no large scale quantitative study of who cites whom. There is a tumblr (a microblog) of epigraphs: *epigraphic*[1], but the data cannot be downloaded and there is no explicit license. Our large collection of epigraphs makes it possible to explore questions such as the following:

- To examine the literary field as a whole through vastly larger sampling than the tiny number of novels that comprise the canon traditionally studied by literary scholars. Although epigraphs are a small part of the ecology of the novel, treated as a network they reveal fresh connections and perspectives on literature
- To study the hitherto untold history of the literary epigraph and its evolution in purpose and form. In particular, the project seeks to discover the origin of the first epigraph, to determine whether the use of epigraphs increases in frequency over time, and to contextualise its evolution in relation to a changing socio-historical context.
- To establish the groundwork for an encyclopaedia of literary epigraphs with entries written by leading experts in the field. This book is projected to be an indispensable resource for academics and students working at all levels.

The epigraph database has the potential to not only provide a new means of describing and analysing the development of the epigraph but to reshape our understanding of the dissemination and international reception of the novel more generally.

[1] https://epigraphic.tumblr.com/

Epigraphs also appear in more technical works, for example *The TEXbook* (Knuth, 1984) cites epigraphs both from the poet Byron and the Proceedings of the United Typothetæ of America. Although our database focuses on literary works, we also include a few technical works, and hope to add more in the future

2. The Epigraph Database

The database contains the following records:

- the epigraph
  - text of the epigraph
  - image of the page it appears on or link to online text
  - title of original text
  - author of original text
  - country of origin or more detailed region if available
  - year of origin
  - language of the epigraph
  - original language of the epigraph if different from the language it is cited int
  - medium (novel, play song, . . .)
  - ISBN of the original text (if it exists)
- the work
  - title
  - author(s)
  - country of first publication or more detailed region if available
  - year of first publication
  - language of work (all currently English)
  - original language of work
  - genre
  - ISBN (if it exists)
- Remarks (for other information such as whether the cited work is fictional)

For example for this paper, the data would be:

- Epigraph
  - The best ideas are common property.
  - [image of page 1]
  - "On Old Age", Moral Letters to Lucilius
  - Seneca the Younger
  - 65 AD*
  - Rome*
  - English
  - Latin*
  - Letters
  - ISBN=9780674990845 (Loeb Classical Library)
- Work
  - Toward An Epic Epigraph Graph
  - Francis Bond, Graham Matthews
  - Miyazaki, Japan
  - 2017
  - English
  - Academic
  - ISSN=???

* shows data that was deduced, rather than explicit in the original. For works such as *Moral Letters to Lucilius*, for which many editions exist, we select one – the goal is to link to a controlled vocabulary to allow further look up of metadata.

The database can be thought of as a very unconnected graph. However, by backing off to the meta-data (e.g. group authors from a location or time or genre as one node) the graph can be made more connected.

2.1. Selection Criteria

The acquisition of data followed a combination of opportunistic and guided data collection strategies. First the data collection started at Nanyang Technological University's Humanities and Social Sciences Library. Research assistants (RAs) photographed epigraphs and the novels' metadata using smartphones and then entered the data into the database. They used `https://isbnsearch.org` in order to accurately record the ISBN and `https://www.bookbrowse.com` to record the date of first publication. We took advantage of the locally hosted Singapore Literature in English Bibliography (Koh, 2008) to ensure that all Singaporean literary texts were checked for epigraphs.

The RAs then systematically worked through the novels in all of the public libraries in Singapore. At this stage, it became clear that Young Adult Fiction would dominate the database since these writers typically publish in large quantities and have a high proportion of epigraphs. The RAs added a tag to these texts so that they could be identified as such their effects on the database accurately recorded.

Since public libraries display a preference for contemporary literature, the RAs also developed a list of influential literature from the seventeenth century to the present (based on Wikipedia lists). They then sourced electronic copies of these texts using the Eighteenth Century Collections Online (ECCO),[2] the Literature Online Database (LION),[3] Google Books,[4] and Project Gutenburg[5] to record the epigraphs. The literature list records whether or not each novel contains an epigraph. This gave us a picture of the history of the epigraph and its spread. The RAs later returned to the dataset for quality control: removing repeat entries, double checking the accuracy of the metadata, and normalizing the data.

[2] `https://quod.lib.umich.edu/e/ecco/index.html`

[3] `http://www.proquest.com/products-services/literature_online.html`

[4] `https://books.google.com/`

[5] `https://www.gutenberg.org/`

2.2. The State of the Database

There are currently 10,921 records in the database, although not all fields are complete. Many epigraphs do not identify the source at all, or only the author's name, it is rare to give the work and date. In addition misquotation (or paraphrasing) is common.

The most common works cited in epigraphs are shown in Table 1, the most common authors in Table 2, the most common types of works in Table 3 and finally the most common quotations used in epigraphs are shown in Table 4. The most cited author (if we exclude the bible) is Shakespeare, by an extremely wide margin. Gothic and fantastic authors are common, reflecting the fact that these genres use epigraphs more. Finally, it is interesting to see that poetry is the most common source of epigraphs, with plays, songs, and proverbs all also popular.

# Cites	Work Cited
129	Bible
35	The Tempest
32	Hamlet
23	Paradise Lost
23	Macbeth
22	Romeo and Juliet
20	As You Like It
19	The Book of Counted Sorrows
18	King Lear
15	Alice's Adventures in Wonderland

Table 1: The most common works quoted in epigraphs

# Cites	Author Cited
323	Shakespeare, William
74	Dickinson, Emily
62	Poe, Edgar Allan
45	Wilde, Oscar
44	Whitman, Walt
43	Blake, William
37	Nietzsche, Friedrich
36	Carroll, Lewis
35	Thoreau, Henry David
35	Milton, John
34	Emerson, Ralph Waldo
33	Einstein, Albert
32	Twain, Mark
32	Frost, Robert

Table 2: The most common authors quoted in epigraphs

The most commonly cited quotations are all from famous literary works, and show a wide spread from Ancient Greece and China (Aristotle and Sun Zu) to modern novels (such as Faulkner). The most common works are dominated by the Bible and Shakespeare, and include one originally non-existent work *The Book of Counted Sorrows*.

This comes from a young adult series where the author cites poems from a fictional book at the start of each book.[6]

# Cites	Type of Work Cited
435	Poem
291	Novel
173	Play
149	Song
73	Proverb
66	Bible Verse
29	Letter
28	Fictional
24	Speech
24	Film
22	Essay
18	Definition of a word

Table 3: The most types of works quoted in epigraphs

Later, the author wrote the book (Koontz, 2001). Another book cited by the same author, *The Book of Counted Joys*, remains non-existent.

2.2.1. Access

The epigraph database is released under the Creative Commons Attribution 4.0 International (CC BY 4.0) license[7], which allows you to share and adapt in any medium or format for any purpose, so long as you give appropriate credit, provide a link to the license, and indicate if changes were made. A snapshot is currently available at `http://compling.hss.ntu.edu.sg/projects/epigraph/`, and future releases and analysis will be made available there.

3. Future Work

In future work, we intend to both increase the size and richness of the database, further analyze it and add support for visualizing patterns.

First we intend to add another 10,000 or so epigraphs, and fill in as much missing information as possible. This work will be completed within a year. The revised network will be published as an open-access online resource available to other scholars and researchers as well as members of the general public.

We wish to take advantage of linked open data to link locations to the geonames database[8] and the works, through ISBN, to further metadata, with the help of NTU's librarians. This helps both with normalization and checking of the data. Having an ISBN number allows us to link to the library catalogue's controlled vocabulary.

We also aim to display computer-generated visualisations of the map of literary influence in a manner accessible to a lay audience at venues such as the British Library and the Art-Science Museum in Singapore: we give sample visualizations in Figures 1 and 2. This will involve both clustering and visualization.

[6] In a letter dated August 10, 1992, Koontz stated: "Actually, there is no such book. I made it up. The way you made up footnote sources for fabricated facts in high-school English reports." `http://flavorwire.com/135568/5-real-books-inspired-by-fake-books`

[7] `https://creativecommons.org/licenses/by/4.0/`

[8] `http://geonames.org/`

# Cites	Epigraph Text	Author
6	The past is never dead. It's not even past.	Faulkner, William
5	If an injury has to be done to a man it should be so severe that his vengeance need not be feared.	Machiavelli, Niccolo
5	We shall not cease from exploration And the end of all our exploring Will be to arrive where we started And know the place for the first time.	Eliot, T.S
4	Truth is beautiful, without doubt; but so are lies.	Emerson, Ralph Waldo
4	Hell is empty and all the devils are here.	Shakespeare, William
3	What is a friend? A single soul dwelling in two bodies.	Aristotle
3	All war is deception.	Sun Tzu
3	If a man will begin with certainties, he shall end in doubts; but if he will be content to begin with doubts, he shall end in certainties.	Bacon, Francis
3	Time present and time past Are both perhaps present in time future, And time future contained in time past.	Eliot, T.S

Table 4: The most common epigraphs

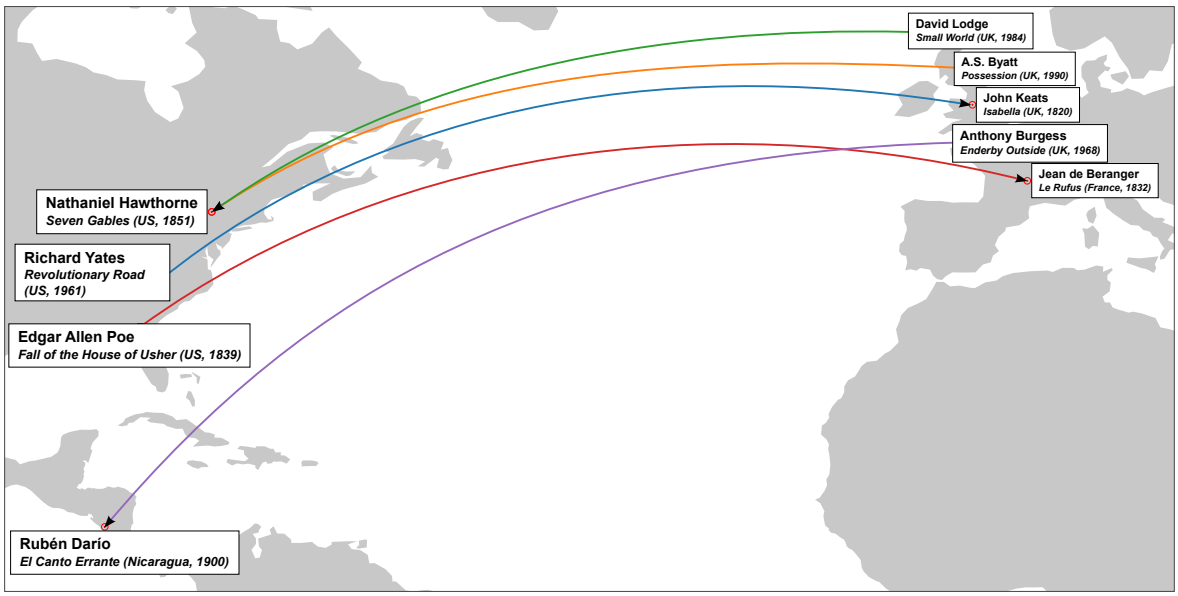


Figure 1: Epigraphs showing trans-atlantic influence

Arrows point from the work with the Epigraph to the cited work.

Further, we will use the network of literary epigraphs to test connections and densities. For instance, we can map the influence of key literary authors such as Charles Dickens or Jane Austen by seeing how much of the network they capture. We can similarly track the influence of women writers or writers of a particular nationality or movement. For example, this project would allow us to determine the extent of the influence of the key Russian realist writers, Turgenev, Tolstoy, and Dostoyevsky on European literature of the late-nineteenth century. When are the Ancient Greek and Roman writers in their ascendancy? Is eighteenth-century satire an influence on postmodern writers of the late-twentieth century? Are lines of literary influence intercontinental or do they cluster around Europe? Singapore has a burgeoning literary movement and this project can determine where in the world Singapore writers take their inspiration from: is it predominantly the English literary canon, the Chinese classics, Indian literature, Malay writers, self-contained, or derived from more eclectic sources? We can also determine the influence of Singaporean literature on world literature.

A figure like William Shakespeare is widely acknowledged as important by literary critics because his works are meaningful and there is a great deal to be said about them. However, if we remove Shakespeare from the network, we can then explore the extent to which the network is altered: is Shakespeare such a focal point that it breaks apart or do the connections occur with a high enough frequency that its shape is unaffected? Conducting a series of case studies such as these presents us with fresh insight into literary and cultural influence, the dissemination of ideas, and the circulation of people.

Finally, the data can also be used to track influence though miscitation. Scholars have found the academics often copy citation information (including mistakes) without checking it, and possibly without reading the paper (Simkin and Roychowdhury, 2003). Similarly in our database, in addition to six citations of *The past is never dead. It's not even past.*

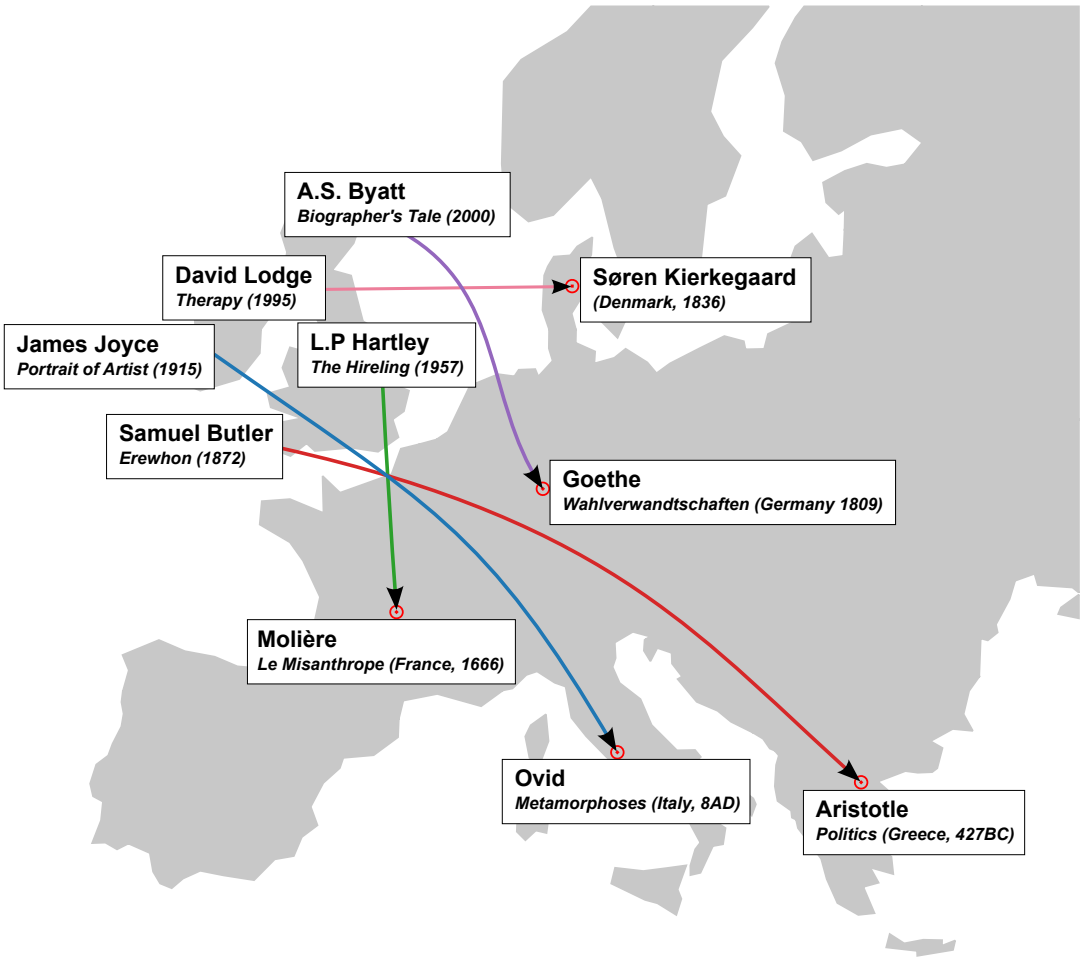


Figure 2: Epigraphs showing the diverse range of European works that have influenced twentieth-century British literature. Arrows point from the work with the Epigraph to the cited work.

from Faulkner (1951), we have several variants, shown in Table 5. We would like to investigate whether the change from *never* to *not*, which happens twice, reveals a connection between the two citing works. We add a link in the database between a variants of the same text.

4. Conclusions

We have created a database of 10,000 epigraphs, which we released under an open license. It allows us to study how works influence each other, as well as the use of the epigraph itself.

Acknowledgments

This research was supported by the Singapore MOE Tier 1 Grant: *Digital Mapping the Literary Epigraph: Quantitative Analysis of Literary Influence Using Network Theory and Thousands of Epigraphs* (RG158/16) and the NTU COHASS Cluster on Digital Humanities. We would like to thank the reviewers for their insightful comments.

5. Bibliographical References

Faulkner, W. (1951). *Requiem for a Nun*. Random House.

Genette, G. (1987). *Paratexts: Thresholds of Interpretation*. Cambridge University Press. (trans Jane E. Lewin).

Knuth, D. (1984). *The TEXbook*. Addison-Wesley.

Koh, T. A. (2008). *Singapore Literature in English: an Annotated Bibliography*. National Library Board Singapore and Centre for Liberal Arts and Social Sciences, Nanyang Technological University. (now online and updated: `https://eps.ntu.edu.sg/client/en_US/SingaporeLiterature/`).

Koontz, D. (2001). *The Book of Counted Sorrows*. Charnel House.

Paul Poplawski, editor. (2017). *English Literature in Context*. Cambridge University Press, second edition.

Simkin, M. and Roychowdhury, V. (2003). Read before you cite! *Complex Systems*, 14:269–274.

Variant	Cited in
The past is never dead. It's not even past.	cited by 6 authors
The past is never dead, it's not even past.	Langley Lee (2010)
The past is not dead. It is not even past.	Carey, Peter (2012)
The past is never dead. It isn't even past.	Wolff, Isabel (2014)
The past is not dead; it's not even past.	Ohanesian, Aline 2015)
The past isn't over. It isn't even the past.	Poulson, Christine (2016)

Table 5: Variants of *The past is never dead. . . .*
Differences underlined

Delta vs. N-Gram Tracing: Evaluating the Robustness of Authorship Attribution Methods

Thomas Proisl*, Stefan Evert*, Fotis Jannidis†, Christof Schöch‡, Leonard Konle†, Steffen Pielström†

*Friedrich-Alexander-Universität Erlangen-Nürnberg, †Julius-Maximilians-Universität Würzburg, ‡Universität Trier

*Bismarckstr. 6, 91054 Erlangen, †Am Hubland, 97074 Würzburg, ‡Universitätsring 15, 54296 Trier

*{thomas.proisl, stefan.evert}@fau.de,

†{fotis.jannidis, christof.schoech, leonard.konle}@uni-wuerzburg.de, pielstroem@biozentrum.uni-wuerzburg.de,

‡schoech@uni-trier.de

Abstract

Delta measures are a well-established and popular family of authorship attribution methods, especially for literary texts. N-gram tracing is a novel method for authorship attribution designed for very short texts, which has its roots in forensic linguistics. We evaluate the performance of both methods in a series of experiments on English, French and German literary texts, in order to investigate the relationship between authorship attribution accuracy and text length as well as the composition of the comparison corpus. Our results show that, at least in our setting, both methods require relatively long texts and are furthermore highly sensitive to the choice of authors and texts in the comparison corpus.

Keywords: authorship attribution, stylometry, evaluation

1. Introduction

Authorship attribution, i. e. the identification of the true author of a text of unknown or disputed authorship based on quantitatively measured linguistic evidence (Juola, 2006; Koppel et al., 2009; Stamatatos, 2009), has applications in many fields, including literary studies, history, forensic linguistics and corpus stylistics. It is based on the assumption that individual writers have idiosyncratic habits of language use (which they are usually not aware of) that lead to stylistic similarities between texts written by the same author. A wide range of stylometric features has been proposed to capture these idiosyncrasies, ranging from relative frequencies of function words to measures of vocabulary richness and syntactic complexity. Based on such feature vectors, a disputed text can then be attributed to the most similar of a set of candidate authors.

If we want to apply authorship attribution methods in real-world settings, e. g. in forensic linguistics, the reliability and robustness of the methods are of utmost importance. Various factors can have an impact on the methods, raising questions such as: To what extent does authorship attribution accuracy depend on the length of the disputed text and the size of the corpus against which it is compared? Is there a minimum text length below which results become too unreliable? What impact does the composition of the comparison corpus have? Are the methods robust with respect to the selection of authors and texts for the comparison corpus?

In this paper, we try to answer those questions, at least partially, for Delta and N-gram tracing, two particularly simple but very successful authorship attribution methods that only rely on word- and character-level features.

2. Background and Related Work

2.1. Delta Measures

Delta measures (Burrows, 2002; Argamon, 2008) are a popular family of authorship attribution methods. They represent texts as simple bags-of-words, focusing on the n most frequent words (nMFW) in the corpus. Word frequencies are standardized to z-scores across the corpus, with a mean of 0 and standard deviation of 1 for each word form type. Text similarity is then quantified by some metric on the resulting vectors of z-scores. Popular choices are Manhattan distance, resulting in the original Burrows's Delta (2002), and angular ("cosine") distance, leading to Cosine Delta proposed by Smith and Aldridge (2011). Typically, (hierarchical) clustering is applied to the distance matrix of all text pairs and similarities between the texts are visualized in the form of a dendrogram. For the purpose of authorship attribution, a disputed text is assigned to the author of the majority of texts in its cluster. Alternatively, a nearest-neighbour classifier can be used or the MFW statistics can serve as features for a supervised machine learning algorithm.

Jannidis et al. (2015) showed that Cosine Delta is usually superior to other variants of Delta. Cosine Delta is also robust with respect to the choice of nMFW, which is why we focus on this particular variant in our experiments. Other key results on Delta measures were obtained by Rybicki and Eder (2011), who investigated the relationship between the number of MFW and authorship attribution success depending on the language of the materials and found notable differences between languages and even within genres. Eder (2013a) showed how text length interacts with attribution quality and found that depending on language and genre a minimum text length of 2,500 to 5,000 words is required for successful authorship attribution. Eder (2013b) investigated the influence of noise, e.g. from OCR errors, on attribution success rates and found that Delta is robust to a certain amount of noise. It should be noted, that – with the exception of Jannidis et al. (2015), who controlled for number of authors – none of the studies mentioned above controlled for text length or number of different authors.

Furthermore, the observations regarding attribution quality made in previous studies are mainly differences between

individual corpora. Variability caused by the selection of the actual texts for the corpora might contribute to the observed differences. None of the previous studies has systematically investigated the influence of sampling effects in corpus composition.

2.2. N-Gram Tracing

N-gram tracing (Grieve et al., submitted) is a novel authorship attribution method from the field of forensic linguistics. It has been designed for the comparison of a short disputed text with a much larger comparison corpus of plausible candidate authors. N-gram tracing extracts all distinct word or character n-grams of a certain length from the disputed text, then determines the percentage of overlap with each author in the comparison corpus. It is important to note that – in marked contrast to Delta – the frequency of the n-grams plays no role at all. The only thing that matters is how many n-grams types also occur in the author-specific parts of the comparison corpus.[1] Grieve et al. (submitted) also suggest a majority voting scheme to combine n-grams of different lengths in order to improve the robustness of the attribution. In their experiments they found that both word 1-to-3-grams and character 4-to-10-grams worked particularly well. This is also what we do in our experiments: We choose the author that is suggested by the majority of word 1-to-3-grams or by the majority of character 4-to-10-grams.

3. Methodology

To answer the questions raised in the introduction, we perform four experiments: Two shortening experiments that examine the performance of Cosine Delta and N-gram tracing depending on text length and two sampling exeriments that evaluate the robustness of both methods with respect to the composition of the comparison corpus.

To allow for a better comparison between Cosine Delta and N-gram tracing, we do not perform a clustering of the Delta distance matrix, but simply attribute the disputed text to the author of its nearest neighbor (i. e. we use a nearest-neighbour classifier).

For Delta, we use the 3,000 most frequent words, which has previously been found to be a robust choice for all languages (Evert et al., 2017). Fig. 1 shows the interaction between text length and nMFW: even for shorter texts, 3,000 MFW achieve better results than the much shorter word lists used by Burrows (2002) and other early work on Delta.

3.1. Shortening Experiments

The shortening experiments are based on three corpora of German, English and French novels (Jannidis et al., 2015; Evert et al., 2017).[2] Each corpus consists of 75 novels from 25 authors, with three texts from each author. We evaluate authorship attribution accuracy via a stratified three-fold cross-validation scheme: in each iteration 25 novels (one per author) are treated as disputed texts. The previous studies on these corpora found that, on the full texts, Cosine Delta showed excellent and robust results, with Adjusted Rand Index (ARI) scores greater than 0.9 for all three corpora (in general, attribution success was higher for the German novels than for the English and French data), but left open the question of the minimal amount of text required for robust authorship attribution results.

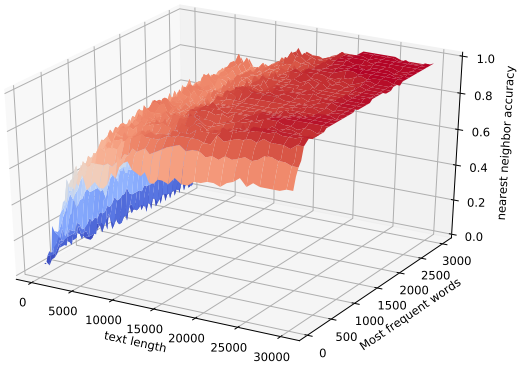


Figure 1: Interaction between text length and nMFW for Cosine Delta in experiment 1a (German corpus)

Experiment 1a: We shorten all texts in the corpus to the same number of tokens, skipping the first 10% of each text because we assume that beginnings and endings of literary texts differ in substantial ways from the rest.

Experiment 1b: We shorten only the disputed text and keep the size of the comparison corpus at a stable size of 30,000 tokens per text, again skipping the first 10% of each text.

3.2. Sampling Experiments

The sampling experiments are based on a collection of 973 German novels by 131 authors, with at least three novels from each author. All authors were native speakers, the collection contains no translations, and the novels were written between 1789 and 1914. We use this collection to draw a large number of samples similar in structure to the corpus of Jannidis et al. (2015), which was used for the shortening experiments.

Experiment 2a: We draw 5,000 random samples of 25 authors and randomly select three novels per author, i. e. each sample consists of 75 texts by 25 authors. Each text is shortened to 30,000 tokens, skipping the first 10%.

Experiment 2b: We select the 25 authors contributing the largest number of novels and draw 5,000 random samples of 75 texts (three per author). Each text is shortened to 30,000 tokens, skipping the first 10%.

The results are evaluated via a stratified three-fold cross-validation scheme as described in section 3.1.

4. Results

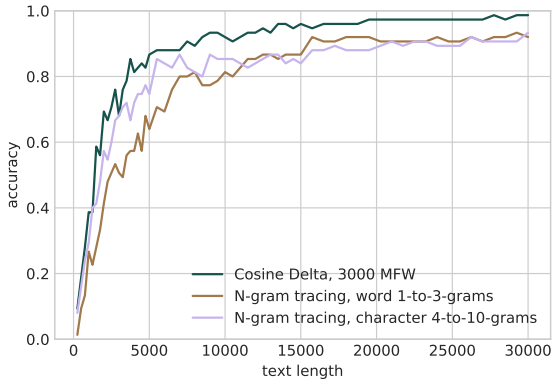
4.1. Shortening Experiments

4.1.1. Experiment 1a

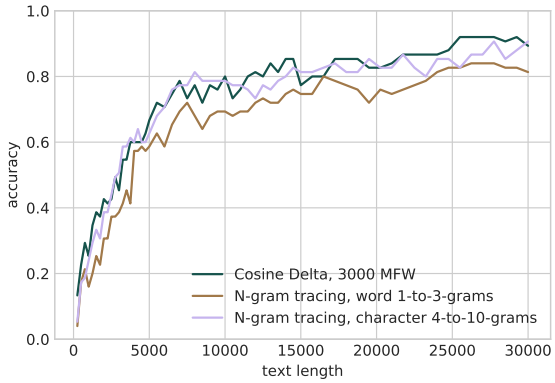
The results of experiment 1a, where we shorten all texts, are shown in Fig. 2. We display authorship attribution accuracy depending on text length for Cosine Delta and N-gram

[1] Wright (2017) describes a similar authorship attribution method based on word n-grams which uses the Jaccard coefficient instead. The Jaccard coefficient normalizes the number of n-grams that occur both in the disputed text and in the author-specific parts of the comparison corpus by the total number of distinct n-grams in those texts.

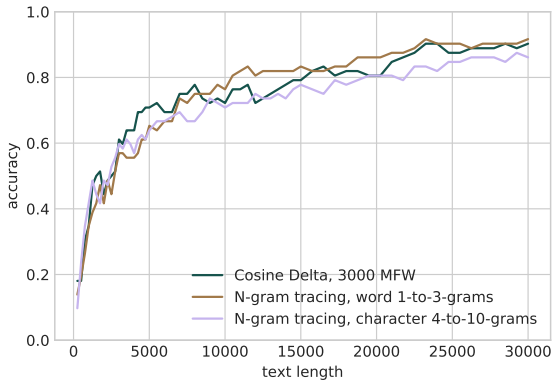
[2] https://github.com/cophi-wue/refcor



(a) German



(b) English



(c) French

Figure 2: Results of experiment 1a (shorten all texts)

tracing, using the majority votes of word 1-to-3-grams and character 4-to-10-grams as described in Section 2.2.

Unurprisingly, the accuracy of all three methods improves with larger text sizes. All methods perform rather poorly for very short texts, where they have to attribute, for example, a 250 word fragment to one of 25 possible authors with only 500 words of comparison text per author.

On the German corpus, shown in the top panel, Delta is consistently better than both N-gram tracing methods. Its performance is relatively stable for longer texts but drops for fewer than 5,000 words. The two N-gram tracing methods perform roughly identically on longer texts but for shorter text lengths character n-grams work better than word n-grams. The performance of both variants stabilizes for text lengths greater than 7,000 words.

On the English corpus, shown in the middle panel, Delta and character n-grams consistently outperform word n-grams. The performance of all methods drops for text lengths smaller than 7,000 words.

On the French corpus, shown in the bottom panel, Delta and word n-grams perform better than character n-grams. Performance drops at 5,000–7,000 words.

All in all, Delta usually performs better or at least approximately as well as N-gram tracing, and it is not entirely clear if word n-grams or character n-grams are better for the latter. Another observation that stands out is that in general the performance on the English and French corpora is notably worse than on the German corpus.

4.1.2. Experiment 1b

The results of experiment 1b, where we shorten only the disputed text, are shown in Fig. 3.

As was to be expected, the results for shorter text lengths are much better than in experiment 1a due to the much larger comparison corpus. In this scenario, N-gram tracing always outperforms Delta on very short texts by a large margin, achieving approximately 50% accuracy on 250-word fragments (while guessing would only achieve 4% accuracy).

On the German corpus, shown in the top panel, Delta is the best method for longer texts and N-gram tracing for shorter texts with less than 3,000 words. For both N-gram tracing methods, 1,000 tokens are sufficient for achieving more than 80% accuracy.

On the English corpus, shown in the middle panel, Delta and character n-grams are the best methods for longer texts, while both N-gram tracing methods are better than Delta for texts with less than 5,000 words. As in experiment 1a, performance is generally worse than on the German corpus and 3,000–4,000 words are needed for achieving 80% accuracy with N-gram tracing.

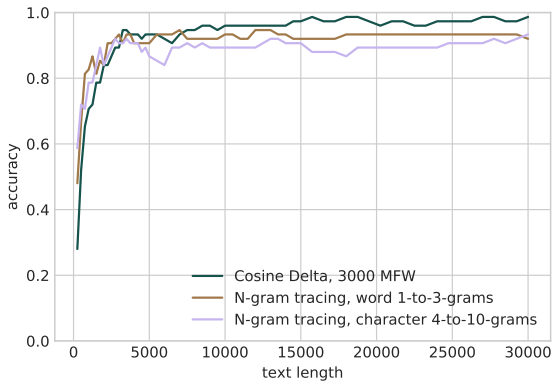
It is remarkable that on the French corpus, shown in the bottom panel, word n-grams are consistently the best method. Even though performance is not as good as on the German corpus, 1,000–2,000 words are sufficient for achieving more than 80% accuracy with word n-grams.

4.2. Sampling Experiments

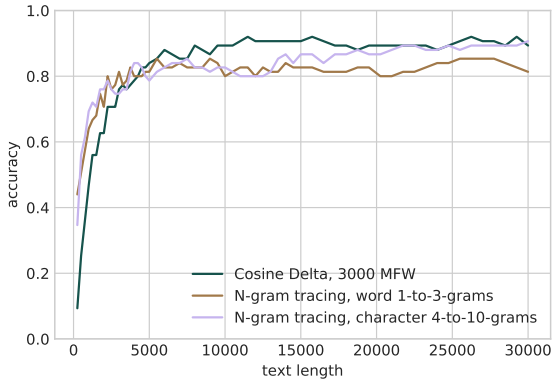
4.2.1. Experiment 2a

Fig. 4 shows the results of experiment 2a, where we draw 5,000 random samples of 25 German authors and shorten them to 30,000 words. For all methods, the central 50% of samples lie in a fairly narrow range of ± 5 percent points around the median (the colored boxes). However, for the remaining 50% there is considerable random variation: Classification accuracies lie between 80% and 100% for Cosine Delta and word n-grams and between 70% and 100% for character n-grams.

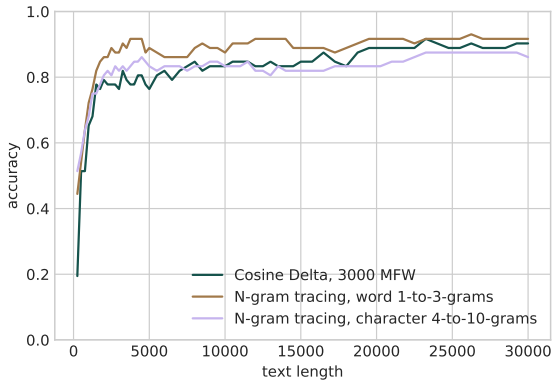
Cosine Delta seems to be a bit better than word n-grams and character n-grams seem to perform notably worse. But we cannot tell from Fig. 4 whether one method is usually better than the other on the same data because the differences between individual samples are much larger than the differences between methods. To this end, we computed pairwise accuracy differences between the three methods for



(a) German



(b) English



(c) French

Figure 3: Results of experiment 1b (shorten disputed text)

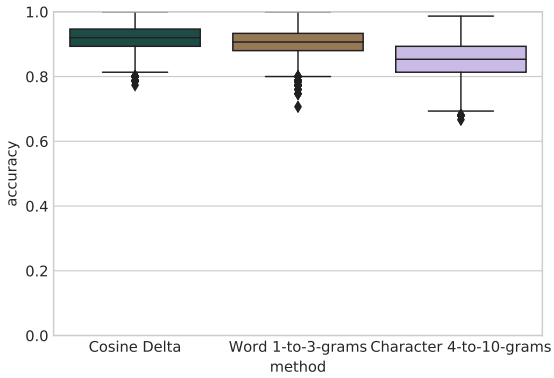


Figure 4: Boxplots with the results of experiment 2a (5,000 sets of 25 German authors)

each of the 5,000 samples. Their distribution is visualized in Fig. 5, showing that Cosine Delta in fact outperforms word n-grams for roughly 75% of the samples. Both Cosine Delta and word n-grams are almost always better than character n-grams.

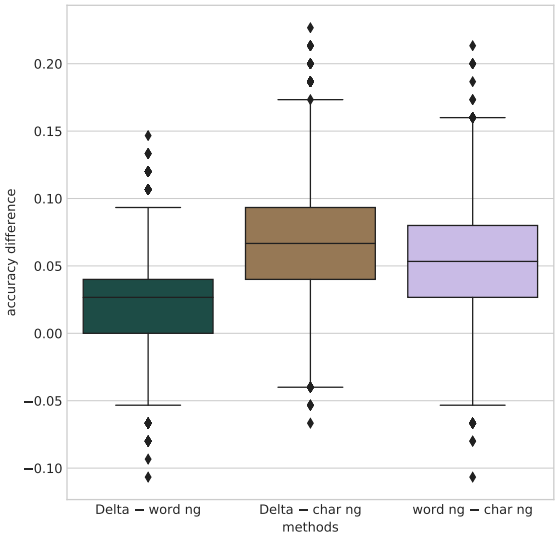


Figure 5: Boxplots with the accuracy differences between the methods in experiment 2a (larger values indicate that the first of the two methods is better)

4.2.2. Experiment 2b

The results of experiment 2b in Fig. 6 show that, as we would expect, sampling texts by the same set of authors results in somewhat less variability. However, the amount of variability is still surprising: Classification accuracies can easily fluctuate by 15 percent points. As before, Cosine Delta outperforms word n-grams and word n-grams outperform character n-grams.

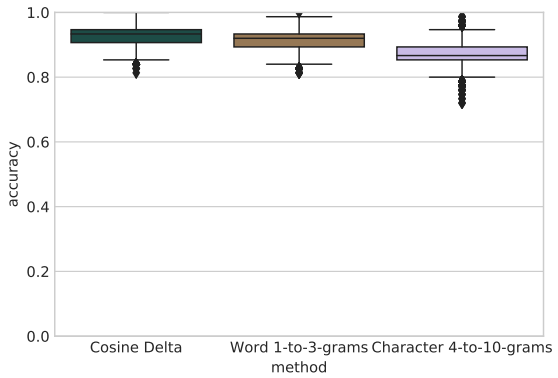


Figure 6: Boxplots with the results of experiment 2b (5,000 sets of texts from the same 25 German authors)

The pairwise differences between the methods are visualized in Fig. 7 and show the same pattern as for experiment 2a: Cosine Delta outperforms word n-grams for roughly 75% of the samples and both Cosine Delta and word n-grams are almost always better than character n-grams.

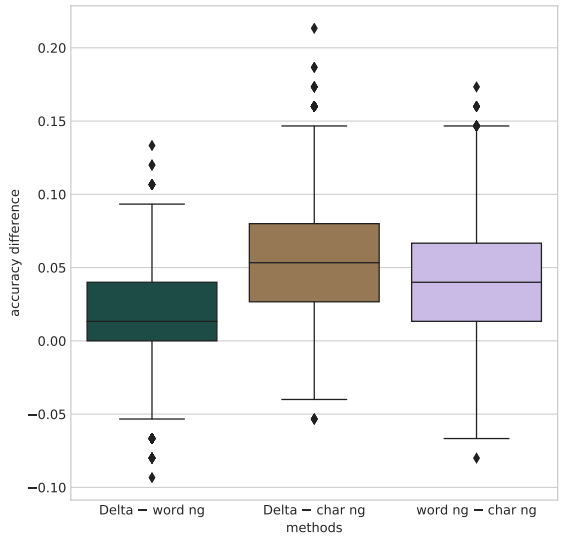


Figure 7: Boxplots with the accuracy differences between the methods in experiment 2b (larger values indicate that the first of the two methods is better)

4.2.3. Sampling vs. Shortening

The two sampling experiments showed that for both Cosine Delta and N-Gram tracing the accuracy of the authorship attribution depends to a certain extent on corpus composition. This raises the question of how meaningful the differences between the three corpora that we observed in the shortening experiments really are.

To address this question, we repeated experiment 1a on the 5,000 random samples of German authors drawn for experiment 2a. In Fig. 8, we show the results for Cosine Delta on the three corpora from experiment 1a. The grey area represents the range in which Cosine Delta lies in 95% of the 5,000 random samples.

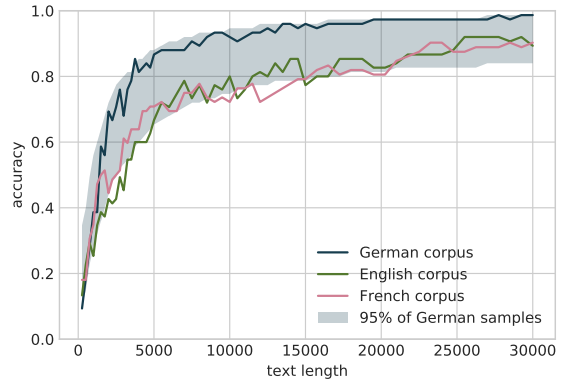


Figure 8: Accuracy of Cosine Delta in experiment 1a and on the 5,000 German samples from experiment 2a

As we can see, our German corpus happens to be a particularly easy sample. Fig. 8 also suggests that the differences observed in the shortening experiments may well be due to the selection of authors and texts.

5. Conclusion

For settings where the comparison corpus, i. e. the available amount of textual material written by the candidate authors, is large in comparison with the disputed text, we can confirm Grieve et al.'s (submitted) claim that N-gram tracing is more reliable than Delta for short texts up to 2,500–3,000. For longer texts, Delta is superior. In cases where both the disputed text and the comparison corpus are relatively small, neither Delta nor N-gram tracing yield reliable results. At least in our setting with a set of 25 possible authors, N-gram tracing requires text lengths of 1,000–3,000 words and a large enough comparison corpus to achieve an acceptable accuracy of 80%.

We also observed considerable performance differences of roughly ten percent points between English, French and German. It is tempting to blame those differences on typological differences between the languages and to speculate about the features that make the German language so well-suited for authorship attribution. However, as the sampling experiments show, the performance of the attribution methods varies considerably with corpus composition. Therefore, only future research comparing the spread of the measures based on many samples across languages will be able to answer the question whether the variance between languages is mainly a result of the corpus setup or whether there is also a factor in play related to language typology.

An interesting and somewhat worrying finding is that even with long texts the composition of the comparison corpus (i. e. the selection of authors and texts) has a large and unpredictable impact on the accuracy of the authorship attribution, which can easily fluctuate by as much as 20 percent points for all the methods tested. This aspect has so far been neglected and should both be kept in mind when interpreting previous results and be taken into account for future studies on authorship attribution.

The obvious next step would be to take the short analysis in Section 4.2.3. to the next level and to run shortening experiments on a large number of samples drawn from large collections of texts in many languages. Such experiments could in a reliable way shed light on the question whether the performance of authorship attribution methods varies between languages.

6. Bibliographical References

Argamon, S. (2008). Interpreting Burrows's Delta: Geometric and probabilistic foundations. *Literary and Linguistic Computing*, 23(2):131–147.

Burrows, J. (2002). 'Delta': a measure of stylistic difference and a guide to likely authorship. *Literary and Linguistic Computing*, 17(3):267–287.

Eder, M. (2013a). Does size matter? Authorship attribution, small samples, big problem. *Digital Scholarship in the Humanities*.

Eder, M. (2013b). Mind your corpus: systematic errors in authorship attribution. *Literary and Linguistic Computing*, 28(4):603–614.

Evert, S., Proisl, T., Jannidis, F., Reger, I., Pielström, S., Schöch, C., and Vitt, T. (2017). Understanding and explaining Delta measures for authorship attribution. *Digital Scholarship in the Humanities*, 32(suppl_2):ii4–ii16.

Grieve, J., Carmody, E., Clarke, I., Gideon, H., Heini, A., Nini, A., and Waibel, E. (submitted). Attributing the

Bixby Letter using n-gram tracing. *Digital Scholarship in the Humanities*. Submitted on May 26, 2017.

Jannidis, F., Pielström, S., Schöch, C., and Vitt, T. (2015). Improving Burrows' Delta – an empirical evaluation of text distance measures. In *Digital Humanities 2015: Conference Abstracts*.

Juola, P. (2006). Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–334.

Koppel, M., Schler, J., and Argamon, S. (2009). Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26.

Rybicki, J. and Eder, M. (2011). Deeper Delta across genres and languages: do we really need the most frequent words? *Literary and Linguistic Computing*, 26(3):315–321.

Smith, P. W. H. and Aldridge, W. (2011). Improving authorship attribution: Optimizing Burrows' Delta method. *Journal of Quantitative Linguistics*, 18(1):63–88.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556.

Wright, D. (2017). Using word n-grams to identify authors and idiolects. A corpus approach to a forensic linguistic problem. *International Journal of Corpus Linguistics*, 22(2):212–241.

An Attribution Relations Corpus for Political News

Edward Newell, Drew Margolin, Derek Ruths
edward.newell@mail.mcgill.ca, dm658@corenell.edu, derek.ruths@mcgill.ca
McGill University, Cornell University, McGill University

Abstract

An attribution occurs when an author quotes, paraphrases, or describes the statements and private states of a third party. Journalists use attribution to report statements and attitudes of public figures, organizations, and ordinary individuals. Properly recognizing attributions in context is an essential aspect of natural language understanding and implicated in many NLP tasks, but current resources are limited in size and completeness. We introduce the Political News Attribution Relations Corpus 2016 (PolNeAR)[2]—the largest, most complete attribution relations corpus to date. This dataset greatly increases the volume of high-quality attribution annotations, addresses shortcomings of existing resources, and expands the diversity of publishers sourced. PolNeAR is built on news articles covering the political candidates during the year leading up to US Presidential Election in November of 2016. The dataset will support the creation of sophisticated end-to-end solutions for attribution extraction and invite interdisciplinary collaboration between the NLP, communications, political science, and journalism communities. Along with the dataset we contribute revised guidelines aimed at improving clarity and consistency in the annotation task, and an annotation interface specially adapted to the task, for reproduction or extension of this work[2].

Keywords: attribution, quotation, sourcing, corpus, news, journalism, politics

1. Introduction

Attribution occurs when an author describes a propositional attitude (Russel, 1940) held by some third party: an agent's statements, intentions, beliefs, knowledge, perceptions, decrees, or sentiments about something (see **Table 1**). Quoting or paraphrasing another person is a familiar form of attribution. But self-attribution, and attribution to artifacts like reports, recordings, or databases are included in the definition.

Prior work defines attributions as consisting of three parts: (1) the source, to whom content is attributed; (2) the content that is attributed; and (3) a cue phrase used to signal attribution, such as "said" or "according to" (Pareti, 2012).

While the statements made by public figures are often inherently newsworthy, attribution to any source is a fundamental mechanism for journalists to lend credibility or authority to, or to nuance, an assertion. Accurate and informative attributions provide readers with transparency and accountability by making journalists' sourcing identifiable (Esser and Umbricht, 2014).

Lately, there has been increased skepticism directed at the mainstream media[1], with questions about the legitimacy of reporting often focusing on sourcing. Thus, attribution phenomena are of fundamental interest for the maintenance of journalistic standards, and should be carefully attended to by the critical reader.

Although attribution is a fundamental rhetorical mechanism in media, it has not received much attention from computational researchers, and partly as a result, datasets for its study are limited in number and depth. Recently, however, direct efforts have been made to study attribution, with the creation of PARC3 (Pareti, 2012), the largest attribution-relations dataset prior to PolNeAR. Despite its important contribution, PARC3 suffers from low annotator recall—many attributions have gone unnoticed by annotators. This impedes the creation of attribution models. In this study we take measures to improve recall, and describe an annotated corpus that doubles the rate of recall of PARC3 and improves inter-annotator agreement[2].

2. Existing resources

The phenomenon of attribution has not typically been studied directly, but instead as part of other phenomena such as opinion analysis, discourse analysis, or the analysis of dialogue in narrative. As a result, there are many existing resources that contain annotations relevant to attribution but which fail to fully capture all attributive phenomena or to adequately label all parts of attributions.

In some cases corpora dedicated to event detection and extraction overlap with attribution, where attribution is seen as a type of event. For example, TimeBank (Pustejovsky et al., 2003) annotates various kinds of events that correspond to attribution. However, because the focus is on events, attributions that do not meet the criteria for being events are not annotated.

Corpora dedicated to opinion analysis and extraction often annotate attribution relations, with Evans et al. (2007) providing an example in English, and Li et al. (2012) providing an example in German. One English resource in particular (Wiebe et al., 2005) provides a corpus of 692 news articles annotated with expressions of speech acts (direct and indirect quotes) and internal states, along with annotations that capture the main components of interest in to the study of attribution.

Various corpora dedicated to discourse analysis and discourse parsing exist, and include annotations of attribution phenomena, including the RST Discourse TreeBank (Carlson et al., 2002), GraphBank (Wolf and Gibson, 2005), with the largest being the Penn Discourse TreeBank 2 (PDTB2) (Prasad et al., 2007). While these corpora do annotate attribution relations, they are designed to annotate all rhetorical

[1]For example theguardian.com/media/2016/dec/18/what-is-fake-news-pizzagate

[2]Dataset: https://github.com/networkdynamics/PolNeAR
Annotation interface: https://github.com/networkdynamics/brat-attribution-annotation

Type	Example
statement	**Jimroy** said *"Sally can't stay."*
intention	**Sally** plans *to run for president*.
decree	**Judge Thomson** issued an injunction *blocking a bill as unconstitutional*.
knowledge	**I** don't know *why they don't pass the bill*.
belief	**He** still thinks *they're on his side.*
perception	**Jimroy** sees *a trend emerging*.
sentiment	**She** disapproved of *the deal*.

Table 1: Basic examples of various forms of attribution, presented in subject-verb-object form for easy comparison. Typesetting indicates the **sources**, cues, and *content*.

relations. Because these corpora are not specific to attribution, they do not indicate key elements of attributions, such as the source. They also tend not to annotate all attributions, missing the attribution of intentions, attributions with implicit sources, and attributions scoped as hypothetical.

Only a limited number of corpora have been designed to address attribution specifically. Elson and McKeown (2010) contribute a corpus of 3176 direct quotes linked to their sources in narrative text. This effort successfully links sources to content, but does not enable the investigation of more challenging cases, such as the attribution of indirect quotes and private states. O'Keefe et al. (2012) provided a corpus of 965 documents from Sydney Morning Herald which annotates direct quotes and their sources, and was later augmented to include indirect quotation. But again internal states are not included in this corpus.

Starting from PDTB2, Pareti (2012) created the PDTB Attribution Relations Corpus 3 (PARC3). This adds specific labels for not just source and content spans, but also *cues*, and annotates attribution relations originally missing from PDTB2. Before PolNeAR, PARC3 is the largest corpus of attribution relations, and the only one which annotates all types of propositional attitude and all three attribution components.

3. Addressing Limitations of Existing Resources

While PARC3 is a valuable resource, it has important limitations. Referring apparently to an earlier version of the dataset, the creators of PARC3 acknowledge that between 30% and 50% of attributions remain unlabelled (Pareti et al., 2013). In the final version of the dataset, there is only one attribution per 69 words, compared to one attribution every 32 words in PolNeAR (see **Table 4**). To compare annotation procedures, we re-annotated 56 randomly selected articles form PARC3, and found that the majority of attributions are un-annotated. Extrapolating the rate of "missed" attributions to the full dataset suggests more than 20 thousand attributions are missing in PARC3.

The creators of PARC3 attempt to mitigate the impact of missing annotations on attribution extraction models by detecting sentences containing words commonly used as cues, and eliminating those sentences that contain no attribution (Pareti et al., 2013). This complicates training. It also fails to address the fact that attributions with less common cues—which are less easily found by annotators—will remain as false-negatives.

We can expect a good extraction model to learn the same biases as, and miss the kinds of attributions missed by trained coders. Less obviously, models so-trained will be biased to over-annotate near common cue words, having never seen a sentence that contains such a word being used outside of the context of attribution. Also, patterns at the inter-sentential level contribute to successful source prediction (Elson and McKeown, 2010). Eliminating whole sentences will distort these patterns relative to raw text.

Improving annotator recall is thus of critical importance. We apply three tactics to increase it:

1. During pilot studies we collect and characterize "tough" attributions—ones which many annotators miss, and ones which annotators find ambiguous under the guidelines. We use these during training and as a reference during annotation, and we clarify the annotation guidelines with respect to those attributions.

2. We emphasize the importance of recall at each (weekly) quality control meeting with annotators. To do so (while keeping balanced precision), we visualize text alignments that show how all annotators handled particular attributions on test articles, allowing annotators to see their disagreements.

3. We allow annotators to indicate uncertainty about an attribution using a "discuss" flag. We encourage annotators to flag ambiguous cases as "discuss", so that other annotators can review it. The "discuss" flag allows us to easily inspect the annotations comprising the boundary of the annotation concept.

The success of these measures is reflected in the corpus statistics (estimated from the 32% of PolNeAR annotated at the time of writing[3]), see **Table 4**, which we discuss further in §6.

4. Corpus Curation

In addition to improving on annotation, we also take this opportunity to curate a corpus that is well-suited to studying attribution phenomena from sociological, journalistic, and political science perspectives. We focus on political news, in which attribution plays a particularly central role, and include articles from 7 publishers from across the political spectrum, and which represent traditional print news and newer online-only publishers:

[3]Finalized statistics: https://github.com/networkdynamics/PolNeAR

- New York Times
- Washington Post
- USA Today
- Breitbart
- Politico
- Huffington Post
- Western Journalism.

We focus on articles that cover the two presidential nominees, Hillary Clinton and Donald Trump, during the year of campaigning leading up to the 2016 US Presidential Election. This gives the corpus a coherent focus on a politically, socially, and journalistically important event.

To maximize the usefulness of the corpus in investigating a variety hypotheses, we selected articles for inclusion using stratified sampling. We binned articles according to the publisher, the candidate receiving the most mentions (Trump or Clinton), and divided the dataset into 12 month-long time periods. We then randomly sampled 6 articles from each bin corresponding to a specific publisher, candidate, and time period. Starting from 55,000 eligible articles, we drew 1008 articles using stratified sampling, plus an additional 20 articles for quality control purposes.

4.1. Collection

We began by obtaining all publicly available articles from each publisher having publication dates between 8 Nov 2015 to 8 Nov 2016 (election day), for a total of 404,000 articles covering a variety of topics and genres. The articles were obtained by download from LexisNexis (in the case of New York Times, Washington Post, and USA Today), or were downloaded from the publisher's website (all others).

In addition to capturing the headline and body text for each article, we captured the following metadata: the author, publication date, whether the article was produced from a newswire, and any tags or descriptors provided by the publisher (e.g. indicating that the article is editorial or news).

4.2. Processing

In extracting the headline and body text from the raw HTML obtained from LexisNexis or the respective publishers' websites, we performed several processing steps as follows.

The paragraph structure of articles was preserved using double line breaks to separate paragraphs. Any embedded tweets were normalized into a single XML tag (the only XML tag used in the corpus) which surrounds the tweet text and indicates author and timestamp as attributes (when available). Advertisements and links to related articles were removed. Blockquotes, which were often indicated in the original HTML document using CSS styles, were preserved by adding an opening quote to each block-quoted paragraph (except if already enquoted), and adding closing quotes only to the final paragraph of the blockquote, as is typical for plain-text multi-paragraph quotations. Each article includes its headline as the first paragraph, followed by the body text.

We processed the articles with Stanford's CoreNLP software (Manning et al., 2014) to provide tokenization, sentence splitting, POS tagging, constituency and dependency parsing, named entity recognition, and coreference resolution. These annotations are provided as part of the corpus, in parallel to the articles in plain text and the standoff attribution annotations.

4.3. Screening for Hard News

The collected articles spanned many genres, from hard news covering significant current events, to soft news such as celebrity gossip, as well as editorials, blog posts, and so on (Esser and Umbricht, 2014). We focused on hard news to avoid uncontrolled variations in attribution due to the stylistic differences of genres, and considering hard news to be most important from a journalistic standpoint. To select the hard news articles, we filtered articles using the metadata tags provided by the publisher which signal the topic or section from which the article was drawn. A considerable amount of effort was made to determine how articles were tagged by the publishers, and ensure that we selected hard news without arbitrarily excluding articles from the dataset. Some publishers, particularly Breitbart and Huffington Post, do not exhibit a sharp stylistic distinction between news and opinion, but we used all indicators made available by the publisher in the form of metadata and site structure.

4.4. Screening for Mentions of Political Candidates

Articles were scanned for mentions of the two presidential candidates using a combination of regular expressions and logical rules devised to provide high precision and recall in disambiguating candidate mentions. This screening procedure is described further in the supplementary material, but, for instance, it explicitly avoids high-profile false positives such as "Bill Clinton" or "Donald Trump Jr." The number of disambiguated mentions of both candidates was tallied for each article, and only those articles with at least one mention of one of the candidates were considered eligible for inclusion in the final corpus. Following this screening, we performed stratified sampling as described above.

5. Corpus Annotation

In constructing PolNeAR, we tried to adhere as closely as possible to the annotation scheme devised for PARC3, while addressing inconsistencies and ambiguities[4]. Here we describe the approach we took to refining the guidelines, and some key deviations. Other minor deviations from PARC3's guidelines are listed in the Appendix. Overall, the revisions were meant improve consistency in annotation without changing what should be considered an attribution.

5.1. Annotation Guidelines

We made significant efforts to clarify the annotation concept in pilot studies. We started by training annotators with the guidelines from PARC3, and then investigated cases of disagreement to uncover gaps and inconsistencies.

In revising the guidelines, we favored the following outcomes, by order of priority: reducing inconsistency, reducing ambiguity, and leaving rules and rulings on examples unchanged from the PARC3 guidelines.

Based on the pilot studies, we developed a set of "templates" consisting of a catalogue of examples and abstract attribution types serving to define the annotation concept

[4]Guidelines: https://github.com/networkdynamics/PolNeAR

of *attribution*. The examples were accumulated from cases generating disagreement during pilot testing. Based on the inspection of thousands of attributions, we created the "types"—abstractions that categorize the examples and illustrate commonality among them. The following types were included:

A source makes statements about,
decrees,
has knowledge of,
believes,
understands,
contemplates,
perceives,
desires,
likes or dislikes,
has an attitude about,
supports, or
has a feeling about *something*.

Many of the types relate to internal states as such examples tended to be a source of ambiguity. The types are not meant to be a mutually exclusive taxonomy. Instead, they serve as an abstract template against which annotators can match specific cases they face during annotation. Culled examples from piloting are organized under each type, so that annotators can easily compare cases they come across with several similar cases from the templates. The interested reader find the templates in the PolNeAR dataset[5].

5.2. Preference for Agentive Sources

All cases of attribution can be seen as belonging to one of two "supertypes":

1. **A communicative agent** expresses, issues an artifact expressing, or holds an internal state representing *something* (the agent may be implicit).
2. **An artifact** expresses or represents *something*.

The main difference between the two is whether content is attributed to an *agent* or an *artifact*. The following examples respectively show annotation according to the above two supertypes (using typesetting to indicate the **source**, cue, and *content*):

1. **Shell** boasted *exceptional earnings*.
2. **The press release** boasted *exceptional earnings*.

Many examples can be read as belonging to both supertypes:

1. **Shell's** press release boasted *exceptional earnings*.
2. **Shell's press release** boasted *exceptional earnings*.

In cases where both readings are possible, there is ambiguity under PARC3's guidelines. It seems more useful to take the attribution's source to be the agent that composed the content, rather than an intermediate message-bearing artifact. Therefore, we resolve the ambiguity by instructing annotators to favor annotation according to supertype (1), and use (2) only when a reading according (1) is not available.

[5]https://github.com/networkdynamics/PolNeAR

5.3. Qualified Scopes

An early question that arises in annotation is whether attributions scoped as conditional, hypothetical, uncertain, or negated, should be annotated. In keeping with PARC3, we consider a candidate attribution to be valid regardless of being under such a qualified scope[6]. For example, the following are valid attributions:

- **She** says *"Hi!"*
- If it were raining, **she** would not have hastily said *"Hi!"*

While this is done mainly for consistency with PARC3, it has the advantage of decoupling the attribution-extraction task from the task of detecting qualified scopes.

5.4. Univocality

One technically important deviation from the PARC3 annotation guidelines relates to vocality—the number of roles that a token can play in the annotation. PARC3 annotations are not univocal—within the same attribution, one token can be both part of both the source and cue.

Allowing multivocality means that the annotations can no longer be modelled directly as a sequence. This increases the space of unique annotations, and concomittantly the hypothesis space needed to fully model it.

Rather than capturing essential structure, this seems to be a quirk of the annotation guidelines. To avoid using degrees of freedom to model an annotation quirk, we stipulate using one label per token. For instance, our guidelines would provide the following annotation:

Sally's advice: *get out before it's too late.*

By means of a partial example, the PARC3 guidelines explicitly indicate that in such cases, "advice" should be labeled as both source *and* cue.

5.5. Exclusion of Nested Attributions

Attributions can themselves contain attributions. For example:

She said *he plans to appeal.*
→ **he** plans *to appeal.*

We do not include nested attributions in PolNeAR for two reasons. First, non-nested attributions are those which are reported by the journalist, whereas nested attributions are reported by the people and organizations described by the journalist. As professional and ostensibly unbiased reporters, journalists can be held to a high standard of accuracy, but this does not apply to the agents they report on. Nested attributions are not relevant to questions about journalistic standards and practice. Second, any attempt to model nested attributions directly severely violates univocality, and represents a qualitatively more difficult endeavor. After annotating non-nested attributions, the creators of PARC3 augmented it with nested attributions, but these have not been used to train or test models of attribution. From both the phenomenological and modelling

[6] This is one of a set of *invariance principles*, illustrated in the templates, and used to drive greater consistency in annotation.

perspectives, we consider nested attributions to be of secondary interest, and elect to focus annotation effort on the high-quality annotation of non-nested attributions.

6. Corpus Validation and Statistics

We now assess the the quality of annotations using multiple agreement-based metrics. The statistics we report in this section are based on a random 32% subset of the corpus[7] and a random selection of 56 PARC3 articles that were re-annotated by PolNeAR annotators.

6.1. Attribution-level agreement: *agr*

The first metric, called *agr*, is one adopted by PARC3, originally sourced from Wiebe et al. (2005). It measures the extent to which annotators agree on the existence of an attribution at a given location in the text, without concern for whether the boundaries of the composite spans are exactly aligned. Under this metric, two annotators agree on an attribution if the respective source, cue, and content from their annotations each have some overlap. Formally, *agr* averages the fraction of attributions by one annotator recalled by the other, and vice versa, for all annotator pairs.

This metric is relevant to questions about the distribution of attributions and journalistic practice, where the precise boundaries of spans are not important. Though annotators may disagree on the boundaries of a source, the spans will generally concur on the grammatical head of the source. For example, one annotator might label "Donald Trump" while another labels "Donald Trump, Republican nominee". For questions pertaining to the distribution of attributions throughout articles, and in terms of their focus and their sourcing, small differences in span boundaries are unlikely to systematically bias statistics.

PolNeAR achieves a high *agr* of 92.3% (see **Table 4**). This is an improvement over PARC3, whose *agr* was 83%[8] (although 87% *agr* was achieved in a pilot study by the PARC3 creators).

The form of *agr* is equivalent to the expected arithmetic mean of the precision and recall when one annotator is randomly selected as the ground truth, and the others are compared to it. From a modelling perspective, we could expect this to be a rough upper bound for the level of precision and recall of a model that learns to annotate like human annotators.

6.2. Attribution Pseudo-Recall

Given our concern over false negatives, we consider a metric that proxies for recall during annotation. We cannot, in principle, measure recall exactly, since we lack ground truth on what is and is not an attribution. Nevertheless, we can calculate a kind of pseudo-recall by comparing the performance of PolNeAR and PARC3 annotators on the articles from PARC3 that were annotated by both groups. In other words, we ask: how many of the PARC3 annotations are recalled by the PolNeAR annotators, and vice versa? We use the same notion of what counts as a matching attribution as used in *agr*.

Applying this to the 56 randomly selected PARC3 articles that were re-annotated by PolNeAR annotators, we find that the PARC3's pseudo-recall is only 31.3%, whereas PolNeAR's is 94.2%. In fairness, there are three factors that may explain the difference in pseudo-recall:

1. PARC3 annotators erroneously under-annotated,
2. PolNeAR annotators erroneously over-annotated, or
3. PolNeAR's annotation concept represents an expansion of PARC3's.

To tease apart these factors, it is necessary to inspect the specific attributions annotated by PolNeAR but missed by PARC3 annotators. We have randomly selected 6 such cases for display in **Table 2**.

We obtained these by first randomly choosing 4 of the 56 articles, and collecting all PolNeAR-annotated attributions not recalled by PARC3 annotators, of which there were 51. Inspecting these attributions manually, at least three represent over-zealous annotation on by PolNeAR annotators—we isolate these for display in **Table 3**. In another 5 cases, a matching PARC3 annotation *did* exist, but technically failed to match due to sufficient disagreement on one of the spans. Forgiving these 8 cases, we adjust the pseudo-recall accordingly, arriving at 41.8% quoted in **Table 4**. (We have not made such adjustments for PolNeAR's recall.)

It is from the remaining 43 attributions that we randomly sampled 6 shown in **Table 2**. To show that these are not merely over-zealous annotations, nor reflect an expanded annotation concept, we have collected annotations from elsewhere in PARC3 that bear resemblance to each missed attribution. Given our close adherence to PARC3's guidelines (with alterations only for resolving inconsistency, ambiguity, and multivocality), and given that similar attributions to the ones missed appear elsewhere in PARC3, we submit that the difference in pseudo-recall reflects true low recall in PARC3, and substantially improved recall in PolNeAR.

6.3. Token-level agreement: Krippendorff's α

Finally, we include an agreement metric that focuses on the extent to which the annotators agree on a detailed token-by-token basis, taking into account both discrepancies in whether an attribution exists, and where exactly the boundaries for the component spans lie. Treating each token as an independent labelling decision, we obtain a Krippendorff's α of 75.4%.

By most standards, this is an acceptable (but not high) level of agreement. To understand the sources of disagreement, we selected attributions where PolNeAR annotators had agreed on the existence of attributions, but which had poor overlap in the component spans.

In the majority of cases, the source span was to blame. The following example is characteristic:

1. *"How are you today?"* **Miller**, a retired worker from a nuclear plant, said pleasantly.
2. *"How are you today?"* **Miller, a retired worker from a nuclear plant,** said pleasantly.

[7] The portion of the corpus annotated at the time of writing. This reflects a random sample balanced across strata.

[8] This value reflects agreement on attributions not already present in the PDTB2 annotations, since PDTB2 annotated some attribution phenomena as discourse structure.

Missed by PARC3	Similar attribution annotated by PARC3 elsewhere
That may have pleased **the secretary**, but. . . .	**Maidenform Inc.** loves *to be intimate with its customers, but not with the rest of the public*.
. . . . but **he** has left no doubt *that he still likes the ideas the commission advanced nearly two years ago*.	There is doubt *that the change would accomplish much* but at least Congress, as in 1935, would . . . .
By January it should be fairly clear *what's hot—and what's not*.	It's understood *that MGM/UA recently contacted Rupert Murdoch's News Corp.*, which made two failed bids for the movie studio, *to see if the company was still interested*.
When traders see the Fed is in the exchange market it may make them tread a little carefully, for fear of *what the central bank may do*.	*The stock market's precipitous drop* frightened **foreign investors**, who quickly bid the dollar lower.
Searle, a unit of Monsanto Co., said *the "beta-blocker" high-blood-pressure drug Kerlone is the first product to reach the market through Lorex Pharmaceuticals*. . . .	*The Maidenform name "is part of American pop culture,"* says **Joan Sinopoli**. . . .
Owner Al Brownstein originally planned *to sell it for $60 a bottle*, but	**PaineWebber** considered *an even harder sell, recommending specific stocks*.

Table 2: Random sample of the 45 PolNeAR-annotated attributions not annotated in PARC3. Each such attribution is paired with an attribution that was annotated in PARC3 elsewhere, by way of showing that these attributions do fall under PARC3's annotation concept.

Take *Lake Vineyard Cabernet from Diamond Creek.*
This recommendation might have encouraged **a turf hungry bureaucrat** to try to expand his power ...
Earlier this month the **St. Louis Fed** held a conference to assess *the system's first 75 years*.

Table 3: The 3 PolNeAR annotations in four randomly-selected PARC3 articles, not annotated in PARC3, which seem to violate the PARC3 and PolNeAR annotation concepts.

	PARC3	PolNeAR
Articles	2294	1028
Publishers	1	7
Words (millions)	1.11	0.76
Attributions (thousands)	16.5	23.9
Words-to-attributions ratio	69	32
Token-wise Krippendorff's α (%)	—	75.4
Attibution-wise agreement (%)	83.0 - 87.0	92.3
Pseudo-recall (%)	41.8	94.2
Est. false negatives (thousands)†	22.9	1.06

Table 4: Overall statistics for PARC3 and PolNeAR, based on the 32% currently-annotated fraction of PolNeAR.
† Calculated from pseudo-recall.

As seen in this example, journalists often include background information on their sources. To varying degrees, this information may be needed to identify the source, or may be added to make a more interesting or compelling narrative. According to both PARC3's and PolNeAR's guidelines, such text should be included in the source span when it is needed to identify the source. But in pilot studies, as well as throughout the annotation to date, this notion has proven fraught and attempts to clarify it have failed so far.
Another important source of disagreement comes from ambiguity in the content span. The following example illustrates a common problem:

1. **Friends of Mr. Clinton.'s. . . .** say *that the ebullient energy he is known for*—whether addressing a crowd or spending an hour on a rope line with voters—*has matured into an elder statesman 's self-assurance*.
2. **Friends of Mr. Clinton.'s. . . .** say *that the ebullient energy he is known for—whether addressing a crowd or spending an hour on a rope line with voters—has matured into an elder statesman 's self-assurance*.

In the first annotation, the portion of text between dashes is attributed to "Friends of Mr. Clinton's", while in the second, it is considered an insertion by the author. In these cases, it is the author's intended reading that is unclear, as opposed to the annotation concept, so such cases seem insoluble without a proliferation of arbitrary rules.
For the most part, such ambiguities in source and content span boundaries will probably not have an impact on

substantive sociological, journalistic, or political scientifc questions. But, from the standpoint of model building, this represents noise in the target label sequence, which should be kept in mind in assessing model performance.

7. Conclusion

PolNeAR[9] is, to date, the largest corpus of attribution relations, in terms of number of attributions, and the most complete, in terms of annotator recall. These features address key limitations in existing resources needed to advance more sophisticated models of attribution such as those based on recurrent neural architectures.

PolNeAR is built on a corpus sampled from the coverage of an event of great sociological, political, and journalistic import—campaign coverage during the year leading up to the 2016 US Presidential Election. This period in recent history is of particular significance to attributive phenomena, due to widely held suspicions of bias and improper sourcing practices. Care has been taken to provide equal representation of candidates, publishers, and time periods in the corpus, to maximize statistical power and interpretability in substantive investigations. As automatic attribution extraction and analysis techniques continue to mature, we hope this will spur interdisciplinary work in which automated tools can be used in service of questions about mass media, communications, and political campaign coverage.

8. Appendix—Deviations from PARC3's Annotation Guidelines

Punctuation as cue. PARC3's guidelines instruct annotators to consider punctuation used to introduce attributions as part of the cue, but only when no other cue words are present. We instead consider such punctuation to *always* be part of the cue to provide greater consistency.

Univocality. Near the boundaries of the spans, there is sometimes ambiguity as to which of two spans a token belongs. We address gaps in the guildelines that permit the ambiguity, and then require that tokens can have at most one label. This means sequence-to-sequence models and transition-based parsers are straightforward to apply.

Crediting of work. We specifically address the issue of attributing works, such as plays, books, paintings, etc. The PARC3 guidelines are ambiguous on this issue. We do not consider the attribution of a work to its creator to be sufficient for attribution. But, the attribution of some *content*, to either the work or its author *is* annotated. This distinction is illustrated by examples in the templates (see §13).

Possessive edge clitic ('s). In attributions where the source has an edge clitic (*'s*), as in "**the author's** view is *that...*", we consider the *'s* to be part of the source, not the cue. This is for consistency with the treatment of attributions in which a possessive pronoun is the source (e.g. "**her** view is *that...*"), wherein the possessive pronoun is labelled as the source, and the possessed noun is the cue.

[9]Obtain the dataset: https://github.com/networkdynamics/PolNeAR

Empty content. PARC3 and PolNeAR both exclude "empty content", such that this would not be a valid attribution: "**John** said *these three words*". This seems reasonable because "these three words" refers not to the content, but to the medium of expression. However, PARC3's guidelines make an exception just in case it is anaphoric to content elsewhere in the document, as in "*"I am sorry."* **John** said *these three words*". For greater consistency, we never annotate tokens referent to the medium of communication as content. (This should not be confused with the annotation of non-personal pronouns which *are* coreferent with content, such as "**she** denies *it*". See the templates, §13, for examples of this distinction.)

9. Bibliographical References

Carlson, L., Okurowski, M. E., and Marcu, D. (2002). *RST discourse treebank*. Linguistic Data Consortium, University of Pennsylvania.

Elson, D. K. and McKeown, K. (2010). Automatic attribution of quoted speech in literary narrative. In *AAAI*.

Esser, F. and Umbricht, A. (2014). The evolution of objective and interpretative journalism in the Western press: Comparing six news systems since the 1960s. *Journalism & Mass Communication Quarterly*, 91(2):229–249.

Evans, D., Ku, L.-W., Seki, Y., Chen, H.-H., and Kando, N. (2007). Opinion analysis across languages: An overview of and observations from the ntcir6 opinion analysis pilot task. *Applications of Fuzzy Sets Theory*, pages 456–463.

Li, H., Cheng, X., Adson, K., Kirshboim, T., and Xu, F. (2012). Annotating opinions in german political news. In *LREC*, pages 1183–1188.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

O'Keefe, T., Pareti, S., Curran, J. R., Koprinska, I., and Honnibal, M. (2012). A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 790–799. Association for Computational Linguistics.

Pareti, S., O'Keefe, T., Konstas, I., Curran, J. R., and Koprinska, I. (2013). Automatically detecting and attributing indirect quotations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 989–999.

Pareti, S. (2012). A database of attribution relations. In *LREC*, pages 3213–3217.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). The penn discourse treebank 2.0 annotation manual.

Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003). The timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

Russel, B. (1940). *An Inquiry into Meaning and Truth*. W. W. Norton Company.

Wiebe, J., Wilson, T., and Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Wolf, F. and Gibson, E. (2005). Representing discourse coherence: A corpus-based study. *Computational Linguistics*, 31(2):249–287.

Face2Text: Collecting an Annotated Image Description Corpus for the Generation of Rich Face Descriptions

Albert Gatt*, Marc Tanti*, Adrian Muscat*, Patrizia Paggio*†, Reuben A Farrugia*, Claudia Borg*, Kenneth P Camilleri*, Michael Rosner*, Lonneke van der Plas*

*University of Malta

{albert.gatt, marc.tanti.06, adrian.muscat, patrizia.paggio, reuben.farrugia}@um.edu.mt

{claudia.borg, kenneth.camilleri, mike.rosner, lonneke.vanderplas}@um.edu.mt

†University of Copenhagen

paggio@hum.ku.dk

Abstract

The past few years have witnessed renewed interest in NLP tasks at the interface between vision and language. One intensively-studied problem is that of automatically generating text from images. In this paper, we extend this problem to the more specific domain of face description. Unlike scene descriptions, face descriptions are more fine-grained and rely on attributes extracted from the image, rather than objects and relations. Given that no data exists for this task, we present an ongoing crowdsourcing study to collect a corpus of descriptions of face images taken 'in the wild'. To gain a better understanding of the variation we find in face description and the possible issues that this may raise, we also conducted an annotation study on a subset of the corpus. Primarily, we found descriptions to refer to a mixture of attributes, not only physical, but also emotional and inferential, which is bound to create further challenges for current image-to-text methods. **Keywords:** face images, vision and language, image-to-text, Natural Language Generation, NLG, crowdsourcing

1. Introduction

This paper describes an annotation project that is being conducted by a cross-disciplinary group of researchers at the University of Malta, the RIVAL (Research In Vision And Language) group, to create a corpus of human face images annotated with rich textual descriptions. The initial goal of the project was to investigate in general how users describe images of human faces, and ultimately to create a resource that could be used to train a system to generate descriptions potentially as varied and rich as possible, thus moving the state-of-the-art of automatic description generation for images of faces from a feature-based process to one that takes advantage of complex textual descriptions. Here we report on a preliminary version of the corpus, focussing on how it was collected and evaluated.[1]

2. Background

Automatic image description research can rely on a wide range of image-description datasets. Such datasets consist of images depicting various objects and actions, and associated descriptions, typically collected through crowdsourcing. The descriptions verbalise the objects and events or relations shown in the images with different degrees of granularity. For example, the most widely-used image captioning datsets, such as Flickr8k (Hodosh et al., 2013), Flickr30K (Young et al., 2014), VLT2K (Elliott and Keller, 2013), and MS COCO (Lin et al., 2014), contain images of familiar scenes, and the descriptions are restricted to the 'concrete conceptual' level (Hodosh et al., 2013), mentioning what is visible, while minimising inferences that can be drawn from the visual information. Other datasets are somewhat more specialised. For example, the Caltech-UCSD Birds and Oxford Flowers-102 contain fine-grained visual descriptions of images of birds and flowers respectively (Reed et al., 2016). Some datasets also contain captions in different languages, as in the case of Multi30k (Elliott et al., 2016). A more extensive overview of image caption datasets can be found in Bernardi et al. (2016). Although images of faces may be included in these datasets, none of them specifically targets face descriptions.

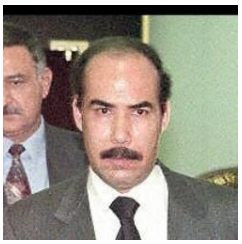
There are several datasets of faces that are widely used by the Image Processing community including the LFW (Huang et al., 2007, Learned-Miller et al., 2016), MegaFace (Kemelmacher-Shlizerman et al., 2016) and IJB-C datasets (Klare et al., 2015). These datasets however do not have labelled attributes. The LFWA (Huang et al., 2007, Learned-Miller et al., 2016) and CelebA (Liu et al., 2015) datasets on the other hand contain images that are labelled with features mainly referring not only to physical facial attributes, such as skin colour and hair style, but also attributes of the person, e.g. age and gender. A number of datasets also focus specifically on emotion recognition and rendering from images of faces (Yin et al., 2006, Tottenham et al., 2009). The expressions are typically either acquired after specific emotions were elicited in the photographed subjects or posed by actors, and were subsequently validated by asking annotators to tag each image with emotion labels. None of these datasets pairs images with text beyond simple feature labels.

The Face2Text dataset, a preliminary version of which is described in this paper, aims to combine characteristics from several existing datasets in a novel way by re-using a collection of images of human faces collected *in the wild*, in order to collect rich textual descriptions of varying semantic granularity and syntactic complexity, which refer to physical attributes, emotions, as well as inferred elements which are not necessarily directly observable in the image itself.

There are a number of reasons why face descriptions are

[1] The corpus will shortly be released to the public. The current version is available upon request.

(a) Male example



- I see a serious man. Such facial expressions indicate that the man is highly committed and dedicated to his work
- A middle eastern gentleman struggling with an administrative problem
- criminal
- Longish face, receding hairline although the rest is carefully combed with a low parting on the person's left. Groomed mustache. Could be middle-eastern or from the Arab world. Double chin and an unhappy face. Very serious looking.

(b) Female example



- blonde hair, round face, thin long nose
- While female , American stylish blonde hair and blue or green eyes wearing a suit , public speaks person
- Middle aged women, blond (natural ?) well groomed (maybe over groomed). Seems to be defending/justifying herself to a crowd/audience. Face of remorse/regret of something she has done.
- An attractive woman with a lovely blonde hair style, she looks pretty seductive with her red lips. She looks like a fashion queen for her age.

Figure 1: Examples of descriptions collected for two faces.

an interesting domain for vision-language research. Descriptions of faces are frequent in human communication, for example when one seeks to identify an individual or distinguish them from another person. They are also pervasive in descriptive or narrative text. The ability to adequately describe a person's facial features can give rise to more humanlike communication in artificial agents, with potential applications to conversational agents and interactive narrative generation, as well as forensic applications in which faces need to be identified from textual or spoken descriptions. Hence, the primary motivation for collecting the Face2Text corpus is that it promises to provide a resource that can break new ground in the problem of automatic description and retrieval of images beyond the current focus on naturalistic images of scenes and common objects.

In corpora which target a specific domain (for example, birds, plants, or human faces), the descriptions solicited tend to be fairly detailed and nuanced. This raises the possibility of more in-depth investigation of the use of language in a specific domain, including the identification and description of salient or distinguishing features (say, eye colour, the shape of a face, emotion or ethnicity) and the investigation of the conditions under which they tend to feature in human descriptions. Indeed, descriptions of faces produced by humans are often *feature-based*, focussing on distinguishing physical characteristics of a face and/or transient emotional states. Alternatively, they may involve inference or analogy. Examples of such descriptions can be seen in Figure 1.

Nevertheless, assuming the existence of an appropriate dataset, architectures for generating face descriptions are likely to share many of the challenges in the more familiar image description task. Hence, it is worth briefly outlining the ways in which the latter has been approached. Approaches to image description generation are based either on caption retrieval or direct generation (Bernardi et al., 2016).

In the generation-by-retrieval approach, human authored descriptions for similar images are stored in a database of image-description pairs. Given an input image that is to be described, the database is queried to find the most similar images to the input image and the descriptions of these images are returned. The descriptions are then either copied directly (which assumes that descriptions can be reused as-is with similar images) or synthesized from extracted phrases. (Ordonez et al., 2011) and (Kuznetsova et al., 2012) are examples of retrieval in visual space; other approaches rely on retrieval in multimodal space (Hodosh et al., 2013, Socher et al., 2014).

On the other hand, direct generation attempts to generate novel descriptions using natural language generation techniques. Traditionally, this was achieved by using computer vision (CV) detectors which are applied to generate a list of image content (e.g objects and their attributes, spatial relationships, and actions). These are fed into a classical NLG pipeline that produces a textual description, verbalising the salient aspects of the image. (Kulkarni et al., 2011) and (Mitchell et al., 2012) are early examples of such systems. The state of the art in image description makes use of deep learning approaches, usually relying on a neural language model to generate descriptions based on image analysis conducted via a pre-trained convolutional network (Vinyals et al., 2015, Mao et al., 2015, Xu et al., 2015, Rennie et al., 2016). While these systems are currently the state of the art, they suffer from a tendency to generate repetitive descriptions by generating a significant amount of descriptions that can be found as-is in the training set (Devlin et al., 2015, Tanti et al., 2018). This suggests that the datasets on

which they are trained are very repetitive and lack diversity. State of the art image captioning requires large datasets for training and testing. While such datasets do exist for scene descriptions, no data is currently available for the face description task, despite the existence of annotated image datasets. In the following section we describe how we addressed this lacuna, initiating an ongoing crowd-sourcing exercise to create a large dataset of face descriptions, paired with images which are annotated with physical features. Our long-term aim is to extend the scope of current image-to-text technology to finer-grained, attribute-focussed descriptions of specific types of entities, akin to those of birds and flowers (Reed et al., 2016), rather than objects and relations in scenes.

3. The Face2Text dataset

Descriptions for the Face2Text dataset are currently being collected via a custom-made crowd-sourcing website. A screen shot of the web page requesting a description of a face is shown in Figure 2.

Pilot study Prior to launching the crowd-sourcing interface, we conducted a small pilot study among approximately ten individuals, who were asked to write descriptions of a sample of the faces in the dataset. These were then used to provide participants with some examples of what was expected of them (see below).

Data 400 Images were selected[2] from the Faces in The Wild dataset[3] (Berg et al., 2005), with the aim of collecting as many descriptions as possible for each image. These are close-up images of the faces of public personalities, taken in naturalistic contexts (that is, without controlling for background, or other 'noise' in the image). The selected images were evenly divided between pictures of males and females.

Procedure The system is designed to allow a participant to describe as many images as they wish. At any stage in the process, a participant is shown an image selected randomly among those that have been given the fewest descriptions so far. In the long-term this equalises the number of descriptions per image.

Participants were given minimal instructions in order to encourage as much variation as possible This was done because the precise nature of face descriptions, and the distribution of various features, is to our knowledge under-researched. Hence, rather than explicit instructions regarding what they were meant to write, participants were shown four examples of faces, each accompanied by three different descriptions collected through our preliminary pilot study.

The main points in the instructions that were given are the following:

- No recognisable faces are to be identified by name.
- Descriptions should not be too short, but should not exceed a maximum of 500 characters.
- Only the face in the middle of the image should be described (some of the images include multiple faces which happened to be near the target face).

After reading the instructions, participants were requested to enter basic demographic information, specifically: gender, age bracket, country of origin and proficiency in English. The latter is a self-rated forced choice among the options *native speaker, non-native but fluent* and *not fluent*. Only one respondent has rated themselves as non-fluent.

Participants could interrupt the study at any point. However, the system saved session variables, meaning that there was a limited time period during which participants could revisit the online interface and resume the description exercise from where they had left off.

Participation and dataset Participation was voluntary and no financial or other incentives were offered to participants. To date, the crowdsourcing experiment has been advertised among staff and students in the University of Malta as well as on social media. A total of 1,400 descriptions have been collected from 185 participants. All 400 images have been described at least 3 times, with approximately 270 images having 4 descriptions.

3.1. Annotation and agreement

As shown in Figure 1, there is considerable variation in the descriptions that people write. While the majority include physical features, there are also emotional descriptors, as well as analogical descriptions (as when a person is described as resembling a criminal; see Figure 1a) and inferred characteristics (such as the inferred nationality of the man in Figure 1a). In addition, such data collection exercises raise a potential ethical concern, insofar as individuals may take advantage of the anonymity of the crowdsourcing exercise to produce texts which are racially or sexually offensive. We note, however, that while we are taking steps (described below) to identify and weed out such ethically problematic descriptions prior to dissemination and use of the data, we do not intend to exclude descriptions simply on the grounds that they describe nationality or ethnicity, as when a participant described the man in Figure 1a as 'middle eastern'. Indeed, such examples raise interesting questions about the salience of such features for different individuals, as a function, for instance, of where they come from (and hence, of what counts as 'frequent' or 'the norm' in their cultural milieu). By including country of origin, age and gender among the demographic details we request from participants, we hope to be able to address these questions in a more informed manner.

3.2. Annotation

To gain a better understanding of these issues, as well as potential challenges in annotating the data, we conducted an annotation exercise on a subset of the data. The study was conducted among all nine members of the RIVAL team. Eight of these were designated as annotators, while a ninth acted as a control annotator.

Each of the 8 annotators was assigned a random set of 194 descriptions. Of these, a random set of 20 descriptions were

[2] http://rival.research.um.edu.mt/facedesc/

[3] http://tamaraberg.com/faceDataset/

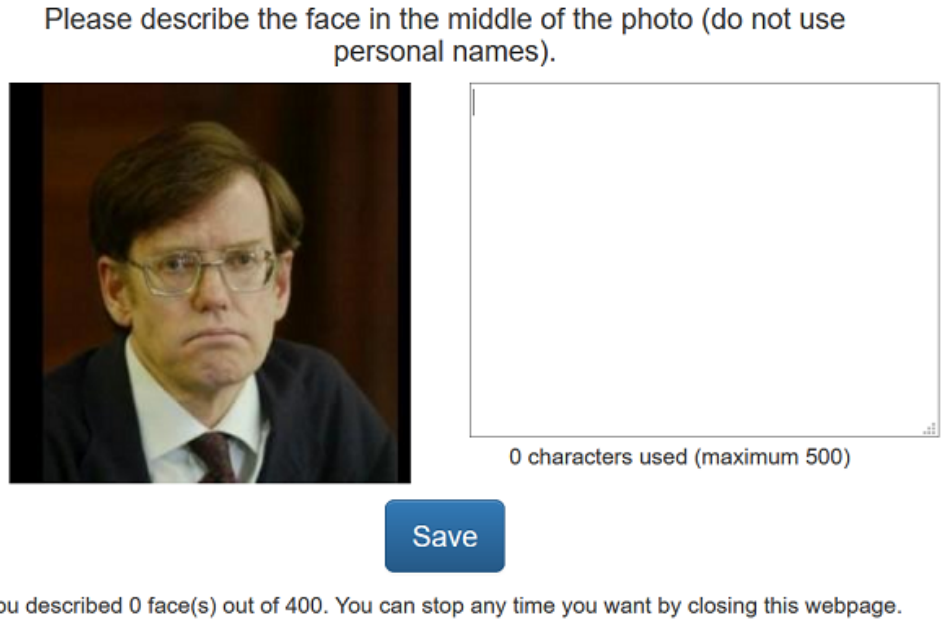


Figure 2: Screen shot of the crowd sourcing website asking a visitor for a description.

Question	**Positive responses**	**Overall agreement (Fleiss's κ)**	**Average κ agreement with control annotator**
1. Is the description garbage?	13%	0.9	0.65 (0.15)
2. Does the description contain elements which are inferred but external to the image?	46%	0.48	0.39 (0.16)
3. Does the description refer to the emotional state of the face?	44%	0.87	0.49 (0.33)
4. Does the description include physical attributes?	85%	0.71	0.76 (0.20)
5. Does the description contain hate speech	1%	1	NA

Table 1: Questions used in the annotation exercise, with the overall proportion of positive responses across all descriptions. Column 3 gives overall agreement using Fleiss's kappa among the 20 descriptions shared among all 9 participants, including the control annotator. Column 4 gives mean agreement between the 8 annotators and the control annotator, with standard deviations in parentheses. Percentages for questions 2-5 exclude descriptions for which question 1 was answered positively by at least one annotator.

shared among all eight annotators. This was used to compute overall agreement. The control annotator's data consisted of 180 descriptions consisting of (a) the shared subset of 20 descriptions; (b) a further 160 descriptions, 20 from each of the 8 annotators' set.

All participants viewed each description and its corresponding image, and responded to the set of yes/no questions shown in the left panel of Table 1. The first of these was intended to weed out descriptions which have no relationship to the image. Descriptions for which annotators answered this question positively are not included in subsequent analyses. The last question in Table 1 was intended to identify potentially racist or sexually discriminatory descriptions, modulo the provisions made above concerning the use of ethnic or other characteristics when these are not used in an offensive manner. As shown in Table 1, only one description was identified as potentially containing hate speech.

The overall proportions of positive responses suggest that the majority of descriptions focus on physical attributes, as expected, but a substantial proportion also incorporate inferred characteristics and/or emotional elements (such as whether a person looks happy or sad).

The table also gives the agreement between annotators on the shared set of 20 descriptions, estimated using Fleiss's kappa, which is appropriate for multiple annotators. As shown in the table, most questions had high levels of agreement with $\kappa \geq 0.7$. The exception is the second question, where agreement falls to 0.48. This indicates that what is 'inferred' is open to interpretation, with some annotators viewing features such as nationality, and even a person's age, as inferred, while others do not. During discussions between annotators, after the exercise was completed, it became clear that age was sometimes viewed as a physical

feature (insofar as it can be deduced from a person's physical characteristics), while others viewed it as an inference.

3.3. Agreement

As a sanity check, we also computed agreement between each of the 8 annotators and the control annotator. Recall that, in addition to the set of 20 descriptions shared among all participants, the latter annotated a further 160 descriptions, consisting of 20 from each of the 8 annotators' cases. The agreement results are shown in the final column of the table, with the exception of Question 5, since none of the descriptions shared with the control annotator were classified by any individual participant as containing hate speech. In this case, agreement figures are generally lower, but they are means over 8 distinct values. Perhaps the most notable drop is in question 4, which deals with descriptions containing emotion, where mean agreement with the control annotator drops to 0.49. Note, however, that this is also the case where variance is highest (SD=0.33). One important reason for the drop is that in this case, one annotator seems to have interpreted the question very differently from the control annotator, resulting in a negative agreement figure[4] and increasing the variance considerably (range: $\langle -0.19, 0.88 \rangle$). The mean without this outlier goes up to 0.60.

3.4. Discussion

Overall, the distribution in the sub-corpus included in the annotation study conforms to expectations, with a majority of descriptions incorporating physical attributes, and a sizeable proportion including emotional and possibly inferred attributes.

As for agreement, the annotators appear to have reliably identified descriptions as falling into the five categories of interest, with the possible exception of inference, which clearly needs a more precise definition. Agreement with the control annotator is generally lower across the board,

One of the consequences of this preliminary study is that we are better placed to predict what a face description dataset will contain, and what the challenges for automatic face description will be. In particular, descriptions are bound to refer to a mixture of attributes, not only physical, but also emotional. The latter are probably more challenging to identify with current CV tools, but may also raise interesting questions about how they should be expressed in relation with physical characteristics. Does a person who is smiling qualify as happy?

Clearly, what is expressed will also depend on the purpose which the descriptions are intended to achieve, though the present crowd-sourcing study did not specify a particular purpose, since the aim was to cast the net as wide as possible with a view to gaining a better understanding of the ways in which people describe faces.

4. Conclusions and Future Work

This paper described ongoing work on the construction of a large corpus of descriptions of human faces, together with an annotation and agreement study whose purpose was to identify the main distribution of types of descriptions, and the extent to which annotators can expect to agree on these types.

Our current work is focussing on extending the crowd-sourcing study to produce a dataset that is sufficiently large to support non-trivial machine learning work on the automated description of faces. This will extend the reach of current image-to-text systems, to a domain where the focus is necessarily less on scene descriptions involving objects and relations, and more on fine-grained descriptions using physical and other attributes. Based on the annotation reported in this paper, we intend to filter irrelevant ('garbage') descriptions and/or ethically problematic ones prior to dissemination.

Apart from extending the crowd-sourcing exercise to elicit more human-authored descriptions, we are actively exploring semi-automatic data augmentation techniques. One particularly promising avenue is to use the existing image-description pairs in the Face2Text corpus to harvest similar, publicly available images. This can be done via the web (e.g. using Google's image search service). Our goal is to mine the text surrounding such images to find portions of text that are similar to (portions of) the descriptions produced by the contributors in our corpus.

A second challenge will be to address possible differences in the purposes for which such descriptions can be produced. The possibilities here are very wide-ranging, from describing a face accurately enough for recognition (for instance, in a forensic context where a description is required to construct an identifiable image), to more gamified or humorous contexts, where descriptions might need to rely more on analogy or inference.

In the medium-term, one of our goals is to undertake a more fine-grained annotation exercise on our existing data, with a view to identifying portions of descriptions that pertain to particular features (physical, emotional, ethnic etc). Our agreement statistics already indicate that these can be identified with reasonably high reliability; by explicitly annotating the data, we hope to be able to develop techniques to automatically tag future descriptions with this high-level semantic information. Based on this, it will be possible to undertake a more fine-grained evaluation of the corpus, for example, to find the types of images for which certain attributes tend to recur in people's descriptions. A further aim is to correlate such trends with the demographic data we collect from participants, with a view to constructing a model to predict which aspects of a face will be salient enough to warrant explicit mention, given the describer's own characteristics and background.

[4] Negative agreement implies less agreement than expected by chance.

Acknowledgements

We thank our three anonymous reviewers for helpful comments on the paper. This work is partially funded by the Endeavour Scholarship Scheme (Malta). Scholarships are part-financed by the European Union - European Social Fund (ESF) - Operational Programme II - Cohesion Policy 2014-2020.

5. References

Berg, T. L., Berg, A. C., Edwards, J., and Forsyth, D. A. (2005). Who's in the picture. In *Advances in neural information processing systems*, pages 137–144.

Bernardi, R., Çakici, R., Elliott, D., Erdem, A., Erdem, E., Ikizler-Cinbis, N., Keller, F., Muscat, A., and Plank, B. (2016). Automatic description generation from images: A survey of models, datasets, and evaluation measures. *Journal of Artificial Intelligence Research (JAIR)*, 55:409 – 442.

Devlin, J., Cheng, H., Fang, H., Gupta, S., Deng, L., He, X., Zweig, G., and Mitchell, M. (2015). Language Models for Image Captioning: The Quirks and What Works. In *Annual Meeting of the Association for Computational Linguistics*.

Elliott, D. and Keller, F. (2013). Image description using visual dependency representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP '13*, pages 1292–1302.

Elliott, D., Frank, S., Sima'an, K., and Specia, L. (2016). Multi30k: Multilingual english-german image descriptions. *arXiv preprint arXiv:1605.00459*.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October.

Kemelmacher-Shlizerman, I., Seitz, S. M., Miller, D., and Brossard, E. (2016). The megaface benchmark: 1 million faces for recognition at scale. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4873–4882.

Klare, B. F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Grother, P., Mah, A., Burge, M., and Jain, A. K. (2015). Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1931–1939, June.

Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. (2011). Baby talk: Understanding and generating simple image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Kuznetsova, P., Ordonez, V., Berg, A. C., Berg, T. L., and Choi, Y. (2012). Collective Generation of Natural Image Descriptions. In *Annual Meeting of the Association for Computational Linguistics*.

Learned-Miller, E., Huang, G. B., RoyChowdhury, A., Li, H., and Hua, G., (2016). *Labeled Faces in the Wild: A Survey*, pages 189–248. Springer International Publishing, Cham.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December.

Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., and Yuille, A. (2015). Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). *ICLR*.

Mitchell, M., Dodge, J., Goyal, A., Yamaguchi, K., Stratos, K., Han, X., Mensch, A., Berg, A., Han, X., Berg, T., and Daume III, H. (2012). Midge: Generating Image Descriptions From Computer Vision Detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL'12)*, pages 747–756, Avignon, France. Association for Computational Linguistics.

Ordonez, V., Kulkarni, G., and Berg, T. L. (2011). Im2text: Describing images using 1 million captioned photographs. In *Advances in Neural Information Processing Systems*.

Reed, S., Akata, Z., Schiele, B., and Lee, H. (2016). Learning deep representations of fine-grained visual descriptions. In *IEEE Computer Vision and Pattern Recognition*.

Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., and Goel, V. (2016). Self-critical sequence training for image captioning. *CoRR*, abs/1612.00563.

Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., and Ng, A. (2014). Grounded Compositional Semantics for Finding and Describing Images with Sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218.

Tanti, M., Gatt, A., and Camilleri, K. P. (2018). Where to put the image in an image caption generator. *CoRR*, abs/1703.09137.

Tottenham, N., Tanaka, J. W., Leon, A. C., McCarry, T., Nurse, M., Hare, T. A., Marcus, D. J., Westerlund, A., Casey, B., and Nelson, C. N. (2009). The nimstim set of facial expressions: Judgments from untrained research participants. *Psychiatry research*.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Institute of Electrical and Electronics Engineers (IEEE), jun.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A. C., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proceedings of The 32nd International Conference on Machine Learning*, volume abs/1502.03044, pages 2048–2057.

Yin, L., Wei, X., Sun, Y., Wang, J., and Rosato, M. J. (2006). A 3d facial expression database for facial behavior research. In *Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on*, pages 211–216. IEEE.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Adapting Serious Game for Fallacious Argumentation to German: Pitfalls, Insights, and Best Practices

Ivan Habernal[†‡], **Patrick Pauli**[†], **Iryna Gurevych**[†‡]
[†]Ubiquitous Knowledge Processing Lab
[‡]Research Training Group AIPHES
Department of Computer Science, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany
www.ukp.tu-darmstadt.de www.aiphes.tu-darmstadt.de

Abstract

As argumentation about controversies is culture- and language-dependent, porting a serious game that deals with daily argumentation to another language requires substantial adaptation. This article presents a study of deploying *Argotario* (serious game for learning argumentation fallacies) in the German context. We examine all steps that are necessary to end up with a successful serious game platform, such as topic selection, initial data creation, or effective campaigns. Moreover, we analyze users' behavior and in-game created data in order to assess the dissemination strategies and qualitative aspects of the resulting corpus. We also report on classification experiments based on neural networks and feature-based models.

Keywords: argumentation, fallacies, serious games

1. Introduction

Computational argumentation and argument mining has been traditionally dealing with understanding argument's structure (Habernal and Gurevych, 2017; Stab and Gurevych, 2017; Eger et al., 2017; Daxenberger et al., 2017). Recently, attention has been paid to pragmatic aspects of arguments, such as convincingness (Habernal and Gurevych, 2016b; Habernal and Gurevych, 2016a) or overall quality (Wachsmuth et al., 2017a). A fairly unexplored area of computational argumentation is *fallacies*: arguments that seem to be valid but are not so (Hamblin, 1970). To tackle the nonexistence of corpora for dealing with fallacies computationally, Habernal et al. (2017) published *Argotario*—a serious game intended to educate players and at the same time collect annotated fallacious arguments.

The majority of research on computational argumentation is English-centric (with several exceptions, such as (Peldszus and Stede, 2015; Liebeck et al., 2016; Chow, 2016)). Not only the language itself but the discussed topics and controversies are culture-specific. For example, 'homeschooling' or 'death penalty' are almost non-existent in Germany, while being highly controversial subjects of discussion in the United States. As *Argotario* had been developed within the English context, simply translating the topics and existing arguments and fallacies into another language does not meet the expectations of a serious game user.

We thus asked the following research questions. First, what are the best means to tackle language adaptation in serious games that depend on world-knowledge, cultural context, and specific controversies (RQ1)? Second, we were interested in the dynamics and outcomes of deployment of the game, namely whether new lay users understand differences between various fallacies (RQ2), which qualitative aspects are to be expected in user-written fallacies (RQ3), and which advertising channels deliver the best return of investment (RQ4).

To answer these questions, we added German language support to *Argotario* and crowd-sourced initial data to face the 'cold-start' problem (RQ1), launched several campaigns (RQ4) and analyzed the obtained data and users' behavior (RQ2; RQ3). Moreover, we conducted several experiments with feature-based and deep-learning models for classifying fallacies. The main contributions of this article are (1) an extensive study of language adaptation of a fallacy-oriented serious game and (2) a dataset released to the community under CC-0 license which is, to the best of our knowledge the first corpus of German and English fallacious arguments.

2. Related work

Fallacies have been thoroughly studied in argumentation theory (Damer, 2013; Tindale, 2007; Schiappa and Nordin, 2013; Walton, 1995; van Eemeren et al., 2014). Despite the vast number of theoretical approaches, empirical research and analysis of fallacies in actual argumentative discourse has been rather limited in scope and size. Several recent endeavors in that direction include, e.g., a manual examination of fallacies found in articles supporting creationism by (Nieminen and Mustonen, 2014) or a manual analysis of fallacies in newswire editorials in major U.S. newspapers before invading Iraq in 2003 by (Sahlane, 2012). These examples demonstrate the enormous persuasive effect of fallacious argumentation; other examples of its rhetorical power can be found in (Macagno, 2013).

The computational perspective on fallacies in natural language arguments has been bound to the process of obtaining reliable data from the crowd and serious-game players (Pollak, 2016; Habernal et al., 2017). There are also several related works devoted to argumentation quality, such as confirmation bias (Stab and Gurevych, 2016), or qualitative assessment of arguments from the Web (Wachsmuth et al., 2017b).

3. Overview of Argotario

Argotario represents an instance of so-called serious games (Mayer et al., 2014) that deals with fallacies in everyday argumentation. *Argotario* is an open-source, platform-independent application with strong educational aspects.[1] It

[1] www.argotario.net

was primarily developed in English but has been extended to support multiple languages (Habernal et al., 2017). In short, players of *Argotario* learn to recognize several types of fallacies as well as to write them, both in a single-player and player vs. player scenarios (an example of an actual player vs. player round is shown later in Figure 2). All arguments[2] are thus composed and evaluated in-game with a minimal intervention, except of the initial data and topic selection.

3.1. Fallacy inventory

Labeling an argument as a fallacy of a certain type is usually clear in textbook examples only (Tindale, 2007; Govier, 2010). While several taxonomies exist in the literature, their empirical usefulness is usually not warranted (Boudry et al., 2015). We thus approached the selection of fallacy types using a bottom-up approach such that the inventory contains fallacy types that are distinguishable from each other and are common in everyday discourse. After several pilot studies, the final inventory consists of the following fallacy types; the examples are actual fallacies written by *Argotario* players.

Ad hominem The opponent attacks a person instead of arguing against the claims that the person has put forward. *Example:* "Yeah, and you are a guy who loves war, that's it. You like it when people die." (Topic: Should the fight versus the Islamic State include military operations?

Appeal to emotion This fallacy tries to arouse non-rational sentiments within the intended audience in order to persuade. *Example:* "Yes, all the polar-bears are dying, and we are next." (Topic: Is global warming really an issue?)

Red herring This argument distracts attention away from the thesis which is supposed to be discussed. *Example:* "I am a hunter. Animals need to die in order to keep balance in the forest." (Topic: Should we allow animal testing for medical purposes?)

Hasty generalization The argument uses a sample which is too small, or follows falsely from a sub-part to a composite or the other way round. *Example:* "Yes, Facebook is censoring racist comments against refugees. It works quite well. All media should be censored." (Topic: Is it effective to censor parts of the media?)

Irrelevant authority While the use of authorities in argumentative discourse is not fallacious inherently, appealing to authority can be fallacious if the authority is irrelevant to the discussed subject. *Example:* "Yes, my husband has the same opinion." (Topic: Is television an effective tool in building the minds of children?)

Non-fallacious argument None of the above. Note that we don't use the term "valid" or "good" argument due to the inherent subjective evaluative meaning of these adjectives.

[2]We will use *arguments* and *fallacies* to refer to the same concept here, namely a (potentially fallacious) argument.

3.2. Game design

In particular, the game is structured into *game rounds* (an atomic mini-game in which an interaction from the user is required and is usually rewarded with points), *levels* (predefined sequences of game rounds, for example with increasing difficulty), and *worlds* (a set of interconnected levels that are visualized as a landscape in the game).

Example One concrete example of a game round is the following: The user is shown an argument to a given topic and her quest is to guess whether the argument is fallacious and if so, which fallacy was committed (fallacy type classification, in other words). Let's assume the correct answer is known to the system.[3] The user is then awarded a point if answered correctly, or no reward is given otherwise. Other game rounds include, for instance, writing a fallacious argument given the topic and the intended fallacy type.

The pre-defined levels are either educative, thus gradually teaching the user all fallacy types using the above-mentioned fallacy classification rounds and writing rounds, or they require two players competing against each other.
The users have to achieve two goals. First, they must finish all predefined levels in the first worlds and therefore learn all fallacy types. The second goal is to achieve high ranking (overall score). The players' scores are shown on a global leaderboard with weekly and overall scores. Scoring high in the weekly leaderboard is another incentive to motivate the player. To get a better sense of the gameplay, we recommend watching the videos at `www.argotario.net`.

4. Porting Argotario to German

As *Argotario* comes with a set of predefined controversial topics, the content has to be interesting and relevant enough to grab players' attention from the beginning and at the same time diverse enough to remain entertaining for long-term players. Furthermore, the target group has to be taken into account. Therefore, translating the English topics into German results into mismatch due to different culture-related controversies.
We thus obeyed the following criteria when selecting topics for the German version, namely (a) presence in German mass media, (b) relevance for a political point of view as articles on politics are generally among the most commented ones in online newspapers and the comment sections are full of fallacious arguments, and (c) long-term orientation which would filter out short-term political scandals or quickly abating trends. We manually compiled a list of 30 topics that fit the criteria; see Appendix A for the full listing.
Another 'cold-start problem' is the need of an initial set of fallacious arguments. In *Argotario*, players have to first learn to recognize existing fallacies, before they are asked to write new ones. Therefore the initial set of arguments is important for the first impression of the game to a new user. We opted for paid crowdsourcing on the German platform Clickworker to collect fallacious arguments for all the 30 argumentation topics. Workers were paid €1.50 for writing three arguments that are pro or con to the given topics. As

[3]The 'correct answers' (gold labels) are estimated based on users' voting, see (Habernal et al., 2017) for details.

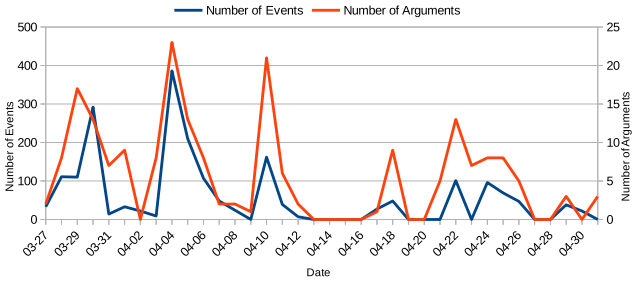


Figure 1: Game activity over time during the evaluation period

a result, we collected 90 high-quality[4] German arguments, which is sufficient for launching the game publicly.

We compared the price with English crowdsourced fallacies by Pollak (2016) who used Amazon Mechanical Turk (AMT). In his experiments, an average price was $0.15 per argument. However, our experiments with AMT were unsuccessful due the German-fluency requirements on the workers. It is thus significantly more difficult and expensive to crowdsource arguments for the game in languages other than English.

5. Data collection campaigns

During the evaluation period between March and April 2017, the game was advertised using three different channels: (1) personal recommendations by the authors, (2) postings advertising *Argotario* in internet forums about politics, language learning, and philosophy, and (3) a one-week paid Google AdWords campaign. These channels were used sequentially which allowed us to measure the success of each one individually. Overall, the majority of players (69%) played the game just once after registration and did not return later. But there is also a group of players who returned several days after the registration.

5.1. Players' activity

We logged several *events*, such as that a player entered a world, finished a world, started playing a level, or finished a game round. Figure 1 shows the number of logged events (in blue) and the number of written arguments (in red) over time during the five weeks of the evaluation period. The three phases distinguished previously are clearly visible in both graphs. The first peak (until April 2) is produced by players who followed personal recommendation (P1), the second and third peaks by players attracted by forum postings (P2, until April 13). The activity during the last two weeks is caused by the AdWords campaign (P3).

To further break down the activity of the three player groups (P1–P3), Table 1 summarizes the average and the distribution of the playing time span as well as the average number of arguments and judges. The player group P2 contains the most long-term users. 12% played the game for more than one day and 15% for at least two days with an interval of more than five days, which makes a percentage of 27% who were sufficiently attracted by the game to return to playing it. This observation is also affirmed by the average number of arguments written by one player which is shown in the penultimate column. As a result, it is clearly visible that the players attracted by forum postings have been the most active out of the three player groups.

5.2. Google AdWord campaign

Advertising using Google AdWords has already been applied in NLP research as a technique to attract users for annotating applications and games (Ipeirotis and Gabrilovich, 2014).

We launched the AdWords campaign for *Argotario* on April 20, 2017 with a €50 budget. Table 2 shows the performance of keywords in terms of clicks on the advertisement, as well as the number of impressions with this keyword (how many times the advertisement was displayed), and the Click-Through-Rate (CTR), which is the probability that a user clicks on the advertisement. Overall, the CTR is rather low as compared the total number of impressions.

All in all, the Google AdWords attracted only 9 players to register and play Argotario (from the 145 clicks), which is rather unsatisfying; the costs are about €5.5 per registered user. The effectiveness of advertisements could be improved by better analytics and targeting (such as in (Ipeirotis and Gabrilovich, 2014), who relied on the Google Analytics tool), or a more fine-grained keyword selection. The campaigns should rather advertise with a high investment for a shorter period than with low investment spread over a longer time span.

6. Data analysis

During the evaluation phase we collected 296 German arguments which we analyzed from different perspectives.

6.1. Language properties and quality

The arguments are on average 18.2 words long, and most of them consist of one or two sentences. This corresponds to findings of Best (2002) who showed that in German journalistic texts the sentence length ranges between 9.62 words (average length of the shortest sentence) and 22.91 words (average length of the longest sentence). Also, Pieper (1979) states that the median sentence length in German discussions is 11.83 words. This comparison shows that the collected arguments share the characteristics of German discussion texts.

Another aspect that serves as an indicator of how serious the game is taken by the players is the quality of orthography and grammar. We manually analyzed spelling mistakes (counted as orthographical errors) as well as grammatical errors (such as wrong case endings of nouns and adjectives, wrong usage of capitalization, missing punctuation or words, and wrong word order). Most arguments (85%) were completely error free, only a fraction contained orthographical errors (7%) or grammar errors (11%).

6.2. Discourse properties

One of the goals of *Argotario* is to collect a corpus of fallacies that resembles a typical Web discussion. This refers particularly to the arguments which are part of the Player

[4] In terms of whether they fit the requested fallacy type and whether they are 'fallacious' enough; a manual analysis was done by the authors.

			Playing Time Span (PTS)				Activity Level	
	#Players	Avg. PTS	<1h	<1d	<5d	>5d	Avg. #Args	Avg. #Judges
P1	23	1d 7h 46min	74%	17%	4%	4%	3.74	7.48
P2	26	**2d 11h 58min**	69%	4%	12%	**15%**	**4.65**	**13.81**
P3	9	0d 4h 14min	56%	33%	11%	0%	3.56	6.56
Total	58	1d 16h 8min	69%	14%	9%	9%	4.12	10.17

Table 1: Playing Time Span and average number of arguments and judges per player group

Keyword	Clicks	Impressions	CTR (%)
Total	145	31871	0.45
argumentieren (*argue/arguing*)	24	733	3.27
deutsch argumentieren (*arguing in German*)	24	342	7.02
politik (*politics*)	19	9897	0.19
lernspiel (*educational game*)	14	5659	0.25
bundestagswahl (*general election in Germany*)	11	1374	0.80
lernspiel online (*online educational game*)	10	2513	0.40
online lernspiel (*online educational game*)	7	2670	0.26
philosophie (*philosophy*)	5	4869	0.10
argumentieren lernen (*learn how to argue*)	3	53	5.66
duolingo	2	349	0.57

Table 2: Keywords statistics of the Google AdWord campaign for Argotario; top 10 'clicks' keywords are shown

A (Hasty Generalization): *"Da ohnehin alle Sportler dopen, hat ein Verbot keine Auswirkungen."* (As all athletes dope anyway, the prohibition has no effects.)
B (Ad Hominem): *"Sie sind doch ein Zyniker, wenn Sie das behaupten."* (You are a cynic when you assert this.)
A (Appeal to Emotion): *"Das hat mit Zynismus nichts zu tun - Solange wir so tun, als hätten wir ein funktionierendes System, solange leiden ehrliche Sportler massiv unter dieser Ungerechtigkeit. Das muss man sich immer wieder vor Augen führen, wer hier leidtragend ist!"* (This has nothing to do with cynicism - As long as we act as if we had a functional system, honest athletes suffer massively under this unfairness. You always have to visualize who is the bereaved here!)
B (Red Herring): *"Und deswegen soll man die Dopingkontrollen abschaffen? Wollen Sie dann auch alle Steuerprüfungen abschaffen, weil es momentan kein perfektes System gibt, mit dem man alle Steuersünder erwischt?"* (And that is why doping controls should be abolished? Do you then also want to abolish tax inspections because there is no perfect system at the moment with which all tax evaders are caught?)

Figure 2: Example dialog about doping in sports in a Player vs. Player round (A vs. B); parentheses show the desired fallacy type the player was instructed to compose.

vs. Player (PvP) round (42.2% of the corpus under investigation). One important feature is discourse coherence, in particular the presence of arguments that directly respond to the opponent's last argument. We define an argument as *dialogical* if the player refers to the opponent's argument and *monological* if the player just writes a stand-alone context-independent argument. Manual analysis revealed that 63% of arguments in the PvP round are dialogical. This is a satisfying result because players are not explicitly instructed to obey any discourse coherence and are only asked to compose a particular type of fallacious arguments. Figure 2 shows one PvP game about doping in sports with clear dialogical properties.

6.3. Fallacy type accuracy

The educational objective of *Argotario* is to teach players what a fallacy is and which fallacy types exist, in particular, players have to compose arguments of the given fallacy type. Analyzing the arguments written by the players contributes to answering the question to what extent this objective is reached. We investigated how accurate the written fallacies are by manually re-labeling the full set of 296 arguments (which we will call *expert fallacy type*). The originally requested fallacy type to be composed by the author is referred to as *intended fallacy type*. Moreover, players also 'judge' other players' fallacies. When at least three labels (votes) for the same argument are available, a gold standard label is estimated (which we call *voting fallacy type*); see (Habernal et al., 2017, p. 10) for details. These labels are available for a subset of 92 arguments that received three and more votes during our evaluation period.

Intended fallacy vs. expert fallacy The intended fallacy type corresponds with the expert fallacy type on 229 of the 296 arguments (average macro F1 score 77%). The results vary widely depending on the fallacy type. While the F1 score for *irrelevant authority* reaches 95%, *red herring* results are around 60%. We can conclude from this that, at least in the context of the game, composing arguments which distract the attention is more difficult in contrast to arguments that use irrelevant authority as backing; *red herring* arguments demand more creativity as it is not as trivially possible to refer to the previous contextual statements.

Voting vs. expert fallacy We further examined whether the 'collective intelligence' through in-game voting leads to accurate labels by comparing the result with the expert fallacy types. As many as 90 of 92 judged arguments (97.8% accuracy) match the correct expert fallacy label. We investigated the two wrongly predicted arguments and found that they were ironical response to the previous argument which features the fallacy of *irrelevant authority* and contains a

Fallacy type	Instances	(in %)
No fallacy	123	28
Appeal to emotion	101	23
Red herring	52	12
Ad hominem	62	14
Hasty generalization	38	9
Irrelevant authority	54	13

Table 3: Class distribution for classification experiments

Model	Accuracy (%)	Macro-F1 (%)
Random-guess baseline	19.6	16.7
Majority class baseline	28.6	7.4
Bi-LSTM	50.9	42.1
SVM	46.3	37.2

Table 4: Overview of the classification results

reference to German history. Three players have judged the arguments and were completely in disagreement. It is debatable if a (non-explicit) reference to Germany's war history of the 20th century is to be qualified as *appeal to emotion* (because it maybe intends to evoke guilt and shame) or as a valid argument. This is a good example for an argument where the correct fallacy type is not 100% clear and cannot be reliably decided by a single annotator. Overall, we can conclude that the wisdom of the crowd works here with very high reliability and even with only three votes it is possible to estimate the true gold label, correct the author of the original argument, and maintain a high quality. This is important for giving the right feedback to novice users when recognizing fallacies.

7. Classification experiments

As one of the long-term goals of *Argotario* is to provide training data for automatic fallacy recognition, we were interested to which extent this problem is solvable using the data gathered so far. The chosen NLP methods are not new but they do reflect the mainstream approaches to classification using deep learning (Goldberg, 2017), thus provide meaningful baselines for further endeavors.

This section thus sketches some classification experiments we performed on the German fallacy dataset. We use all arguments collected during the evaluation period, the arguments created as start-up data, and arguments collected during pilot testing. All arguments were re-labeled by the authors (similarly to *expert fallacy type*) to ensure their reliability. This resulted into 430 labeled arguments with six classes (*no fallacy*, *appeal to emotion*, *red herring*, *ad hominem*, *hasty generalization*, and *irrelevant authority*).

We experimented with a bi-directional LSTM model (Hochreiter and Schmidhuber, 1997) fed with German 64-dimensional word embeddings (Al-Rfou et al., 2013). As a second model, we opted for Support Vector Machines with a range of manually compiled features (question marks, discourse markers signaling nesting, punctuation, length, capitalization, pronouns signaling attacks, reported-speech words, and few others). All experiments were conducted using 10-fold cross validation. The dataset contains 430 gold-labeled arguments, distributed as shown in Table 3.

Overall results are shown in Table 4. A detailed examination of Bi-LSTM results revealed that the best performance was achieved for *no fallacy* and *ad hominem* (F1 score over 60%). Classes with the lowest F1 scores were *red herring* and *hasty generalization*, partly due to their limited presence in the dataset (12% and 8%, respectively). While SVM performed only slightly worse than Bi-LSTM, the *red herring* class was never predicted; this can be explained by the fact that the SVM features concentrate on formal and lexical cues, wile *red herring* arguments rarely contain signal words or other typical signatures. To better identify *red herrings*, it would be necessary to implement features on higher levels of the NLP pre-processing chain, including semantics and world knowledge.

We further compared our results to experiments with the same classification task on English dataset containing 1,160 arguments (Pollak, 2016). While Pollak (2016) achieved 40% macro F1 using Convolutional neural network (CNN), our CNN model performed worse (37%; not reported in Table 4). On the other hand, we achieved better F1 score using a simple LSTM (47%; not reported in Table 4) than Pollak (2016) (43%). However, these results must be taken with a grain of salt as both datasets are rather small to fully leverage the power of deep neural networks.

8. Conclusion and Outlook

We showed that porting a serious game dealing with fallacious argumentation to another language requires substantial effort in adapting topics and preparing high quality starting data (RQ1). We approached the topic selection empirically by relying on mass media and paid crowd-sourcing, which delivered a reasonable starting setup. We manually analyzed the in-game produced data and found that while users are only partially correct in writing fallacies of the given type, the 'wisdom of the crowd' through in-game voting caters for precise corrections and thus high-quality labels (RQ2; Section 6.3.). Furthermore, most users intuitively obey discourse coherence and clear writing (RQ3; Sections 6.1. and 6.2.). By exploring the dynamics of campaigns and user behavior, we found that a high conversion rate of most game rounds shows that players find *Argotario* attractive and do not quit the game early. Out of the three advertising methods, postings in forums have shown the most success (RQ4; Section 5.).

All data from *Argotario* are published under permissive Creative Commons Zero (CC0) license and can be obtained at `https://github.com/UKPLab/argotario`.

9. Acknowledgments

This work has been supported by the ArguAna Project GU 798/20-1 (DFG), and by the DFG-funded research training group "Adaptive Preparation of Information form Heterogeneous Sources" (AIPHES, GRK 1994/1).

10. Bibliographical References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed Word Representations for Multilingual NLP. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192,

Sofia, Bulgaria, August. Association for Computational Linguistics.

Best, K.-H. (2002). Satzlängen im deutschen: Verteilungen, mittelwerte, sprachwandel. *Göttinger Beiträge zur Sprachwissenschaft*, 7:7–31.

Boudry, M., Paglieri, F., and Pigliucci, M. (2015). The Fake, the Flimsy, and the Fallacious: Demarcating Arguments in Real Life. *Argumentation*, 29(4):431–456.

Chow, M. (2016). Argument Identification in Chinese Editorials. In *Proceedings of the NAACL Student Research Workshop*, pages 16–21, San Diego, CA, USA. Association for Computational Linguistics.

Damer, T. E. (2013). *Attacking Faulty Reasoning: A Practical Guide to Fallacy-Free Arguments*. Cengage Learning, Boston, MA, 7th edition.

Daxenberger, J., Eger, S., Habernal, I., Stab, C., and Gurevych, I. (2017). What is the Essence of a Claim? Cross-Domain Claim Identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2045–2056, Copenhagen, Denmark. Association for Computational Linguistics.

Eger, S., Daxenberger, J., and Gurevych, I. (2017). Neural End-to-End Learning for Computational Argumentation Mining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11–22, Vancouver, Canada. Association for Computational Linguistics.

Goldberg, Y. (2017). *Neural Network Methods for Natural Language Processing*, volume 37 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.

Govier, T. (2010). *A Practical Study of Argument*. Wadsworth, Cengage Learning, 7th edition.

Habernal, I. and Gurevych, I. (2016a). What makes a convincing argument? Empirical analysis and detecting attributes of convincingness in Web argumentation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1214–1223, Austin, Texas. Association for Computational Linguistics.

Habernal, I. and Gurevych, I. (2016b). Which argument is more convincing? Analyzing and predicting convincingness of Web arguments using bidirectional LSTM. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1589–1599, Berlin, Germany. Association for Computational Linguistics.

Habernal, I. and Gurevych, I. (2017). Argumentation Mining in User-Generated Web Discourse. *Computational Linguistics*, 43(1):125–179.

Habernal, I., Hannemann, R., Pollak, C., Klamm, C., Pauli, P., and Gurevych, I. (2017). Argotario: Computational Argumentation Meets Serious Games. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 7–12, Copenhagen, Denmark. Association for Computational Linguistics.

Hamblin, C. L. (1970). *Fallacies*. Methuen, London, UK.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Ipeirotis, P. G. and Gabrilovich, E. (2014). Quizz: Targeted Crowdsourcing with a Billion (Potential) Users Panagiotis. In *WWW '14: Proceedings of the 23rd international conference on World wide web*, pages 143–154, Seoul, South Korea. ACM.

Liebeck, M., Esau, K., and Conrad, S. (2016). What to Do with an Airport? Mining Arguments in the German Online Participation Project Tempelhofer Feld. In *Proceedings of the Third Workshop on Argument Mining*, pages 144–153, Berlin, Germany. Association for Computational Linguistics.

Macagno, F. (2013). Strategies of character attack. *Argumentation*, 27(4):369–401.

Mayer, I., Bekebrede, G., Harteveld, C., Warmelink, H., Zhou, Q., van Ruijven, T., Lo, J., Kortmann, R., and Wenzler, I. (2014). The research and evaluation of serious games: Toward a comprehensive methodology. *British Journal of Educational Technology*, 45(3):502–527.

Nieminen, P. and Mustonen, A.-M. (2014). Argumentation and fallacies in creationist writings against evolutionary theory. *Evolution: Education and Outreach*, 7(1):11.

Peldszus, A. and Stede, M. (2015). Joint prediction in MST-style discourse parsing for argumentation mining. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 938–948, Lisbon, Portugal. Association for Computational Linguistics.

Pieper, U. (1979). *Über die Aussagekraft statistischer Methoden für die linguistische Stilanalyse*. Gunter Narr, Tübingen.

Pollak, C. (2016). Serious games for learning fallacy recognition. Master Thesis, Techische Universität Darmstadt.

Sahlane, A. (2012). Argumentation and fallacy in the justification of the 2003 War on Iraq. *Argumentation*, 26(4):459–488.

Schiappa, E. and Nordin, J. P. (2013). *Argumentation: Keeping Faith with Reason*. Pearson UK, 1st edition.

Stab, C. and Gurevych, I. (2016). Recognizing the Absence of Opposing Arguments in Persuasive Essays. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 113–118, Berlin, Germany. Association for Computational Linguistics.

Stab, C. and Gurevych, I. (2017). Parsing Argumentation Structures in Persuasive Essays. *Computational Linguistics*, 43(3):619–659.

Tindale, C. W. (2007). *Fallacies and Argument Appraisal*. Cambridge University Press, New York, NY, USA, critical reasoning and argumentation edition.

van Eemeren, F. H., Garssen, B., Krabbe, E. C. W., Snoeck Henkemans, A. F., Verheij, B., and Wagemans, J. H. M. (2014). *Handbook of Argumentation Theory*. Springer, Berlin/Heidelberg.

Wachsmuth, H., Naderi, N., Habernal, I., Hou, Y., Hirst, G., Gurevych, I., and Stein, B. (2017a). Argumentation Quality Assessment: Theory vs. Practice. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 250–255. Association for Computational Linguistics.

Wachsmuth, H., Naderi, N., Hou, Y., Bilu, Y., Prabhakaran, V., Thijm, T. A., Hirst, G., and Stein, B. (2017b). Com-

putational Argumentation Quality Assessment in Natural Language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 17)*, page (to appear), April.

Walton, D. (1995). *A Pragmatic Theory of Fallacy*. The University of Alabama Press, Tuscaloosa, AL.

A List of German argumentation topics

- Es ist gut, dass Deutschland viele Flüchtlinge aufgenommen hat. *Germany accepting that many refugees was good.*
- Geschäfte sollten jeden Tag rund um die Uhr geöffnet sein. *Shops should be open 24-7.*
- Öffentliche Verkehrsmittel sollten kostenlos sein. *Public transportation should be free of charge.*
- Deutschland sollte ein Bedingungsloses Grundeinkommen einführen. *Germany should implement basic income.*
- In der Schule sollte es keine Noten geben. *Schools should abandon grades.*
- Der Konsum von Cannabis sollte legalisiert werden. *Marijuana consumption should be legalized.*
- An Universitäten sollte keine Forschung zu militärischen Zwecken durchgeführt werden. *Universities should not carry out any military research.*
- Man sollte schon ab 16 Jahren wählen dürfen. *Voting age should be reduced to 16.*
- Die Türkei sollte in die EU aufgenommen werden. *Turkey should not join the European Union.*
- Gentechnik ist etwas Gutes. *Genetic engineering is good.*
- Es sollte mehr Videoüberwachung im öffentlichen Raum geben. *There should be more surveillance cameras in public areas.*
- Die Erbschaftssteuer sollte erhöht werden. *The inheritance tax should be raised.*
- Es ist richtig, dass Hartz-IV-Empfängern ihre Leistungen gekürzt werden, wenn sie ein Job-Angebot ablehnen. *Reducing Hartz-IV*[5] *benefits of those who refuse a job offer is right.*
- Homosexuelle Paare sollten Kinder adoptieren dürfen. *Homosexual couples should be allowed to adopt children.*
- Man lebt besser, wenn man vegan lebt. *Vegan lifestyle is a better lifestyle.*
- Der Verfassungsschutz sollte abgeschafft werden. *The Verfassungschutz*[6] *should be abandoned.*
- Der Euro sollte abgeschafft werden. *The Euro currency should be abandoned.*
- Es sollte an staatlichen Schulen keinen Religionsunterricht geben. *Religion classes should not be taught at public schools.*
- Ausländer sollten an Kommunalwahlen teilnehmen dürfen. *Foreigner should be allowed to participate in local elections.*
- Die Wehrpflicht sollte wieder eingeführt werden. *Compulsory military service should be enforced again.*
- Es wird zu wenig gegen den Klimawandel unternommen. *Climate change measures are unsatisfying.*
- Die Agenda 2010 war gut für Deutschland. *The "Agenda 2010"*[7] *was good for Germany.*
- Es sollte eine Steuer auf Plastikverpackungen eingeführt werden. *Plastic packaging should be taxed.*
- Kriminelle Ausländer sollten sofort abgeschoben werden. *Criminal foreigners should be deported right away.*
- Doping im Sport sollte legalisiert werden. *Doping in sport should be legalized.*
- Deutsche Städte sollten sich für die Ausrichtung von Olympischen Spielen bewerben. *German cities should apply for Olympic games.*
- Es ist besser, Bücher zu lesen statt Filme zu schauen. *Reading books is better than watching movies.*
- Es ist gut, dass klassische Musik mit Steuergeld gefördert wird. *Supporting classical music with tax money is good.*
- Man sollte militärisch gegen den IS vorgehen. *Military operations should be taken against the IS.*
- Kinder sollten früh mit Computern und Smartphones umgehen lernen. *Children should learn how to operate computers and smartphones in the early age.*

[5] https://en.wikipedia.org/wiki/Hartz_concept

[6] https://de.wikipedia.org/wiki/Verfassungsschutz

[7] https://en.wikipedia.org/wiki/Agenda_2010

Crowdsourcing Regional Variation Data and Automatic Geolocalisation of Speakers of European French

Jean-Philippe Goldman[1], Yves Scherrer[2], Julie Glikman[3], Mathieu Avanzi[4], Christophe Benzitoun[5], Philippe Boula de Mareüil[6]

[1] LATL-CUI University of Geneva, [2] University of Helsinki,
[3] LILPA University of Strasbourg, [4] FNRS/Université catholique de Louvain,
[5] ATILF Université de Lorraine, [6] LIMSI-CNRS

jeanphilippegoldman@gmail.com

Abstract

We present the crowdsourcing platform *Donnez Votre Français à la Science* (DFS, or "*Give your French to Science*"), which aims to collect linguistic data and document language use, with a special focus on regional variation in European French. The activities not only gather data that is useful for scientific studies, but they also provide feedback to the general public; this is important in order to reward participants, to encourage them to follow future surveys, and to foster interaction with the scientific community. The two main activities described here are 1) a linguistic survey on lexical variation with immediate feedback and 2) a speaker geolocalisation system; i.e., a quiz that guesses the linguistic origin of the participant by comparing their answers with previously gathered linguistic data. For the geolocalisation activity, we set up a simulation framework to optimise predictions. Three classification algorithms are compared: the first one uses clustering and shibboleth detection, whereas the other two rely on feature elimination techniques with Support Vector Machines and Maximum Entropy models as underlying base classifiers. The best-performing system uses a selection of 17 questions and reaches a localisation accuracy of 66%, extending the prediction from the one-best area (one among 109 base areas) to its first-order and second-order neighbouring areas.

Keywords: language variation, regionalism, crowdsourcing, geolocalisation, linguistic geography, cartography

1. Introduction

Linguistic surveys and experiments can now easily reach thousands of people through the internet and smartphones. This 'crowdsourcing' methodology allows researchers to collect linguistic resources on a large scale (Cook et al. 2013), helping to describe linguistic phenomena (regional variation, non-normative forms, among others) that are underrepresented in traditional corpora. As a quick and easily reproducible way of collecting data, one can even conceive of using crowdsourcing over several years in order to compare successive snapshots of linguistic variables and thus describe variation in time.

In recent years, various crowdsourcing projects have been set up for collecting linguistic data through web applications, such as *Français de nos régions* (Avanzi et al. 2016) for European and Canadian French, *Atlas der deutschen Alltagssprache* (Möller & Elspaß 2015) for regional varieties of German, the *Harvard Dialect Survey* (Vaux & Bert 2013) for regional variation in the USA, *VerbaAlpina* (Krefeld and Lücke 2014) for dialects spoken in the Alps, and more recently, the twin websites *tonaccent/dindialäkt* focusing on the perception of Swiss French accents and Swiss German dialects (Goldman et al. 2018).

Other applications have been explicitly framed as geolocalisation games, aimed at predicting the provenance of the user on the basis of their naming of a set of objects. Whereas some projects have been distributed as smartphone apps, e.g. to document English and German dialectal variation (*Dialäkt Äpp*, Leemann et al. 2016; *English Dialects App*, Leemann et al. 2018a; *Grüezi, Moin, Servus*, Leemann et al. 2018b), other projects - mainly from outside the academic community - were designed as websites, e.g. the quizzes from the Belgian newspaper Le Soir (*«Quel français de Belgique parlez-vous?»*) or from the Télévision Suisse Romande (with *«le Parlomètre romand»*).

All these initiatives - based on such a popular theme as language variation - met with great success and collected large linguistic datasets, enabling novel scientific studies in this field.

In this paper, we present a crowdsourcing platform - *Donnez Votre Français à la Science* (DFS, or "*Give your French to Science*") - that includes several activities related to language variation. It is dedicated to French-speaking Europe (i.e., mainly Belgium, France and Switzerland[1]) and it is designed to be open to linguists to initiate their own projects. Moreover, it aims to deliver popularised feedback to the general public. After a general presentation of the platform, we present two activities: a linguistic quiz with immediate feedback to the participant, and a geolocalisation game - the first, to our knowledge, that covers French-speaking Europe. A large part of the paper is devoted to the development of the geolocalisation algorithm. The platform can be reached at this address: http://donnezvotrefrancais.fr

2. Crowdsourcing platform

The DFS platform is derived from PyBossa, a robust crowdsourcing framework that is designed for developing various interactive activities, based on the Flask micro-framework. The flexibility of the latter allowed for required adaptations such as having a limited number of questions/items per survey and a feedback mechanism, providing a score and/or a dynamically-computed map.

[1] Aosta Valley in Italy, despite having French as an official language, yielded too few participants.

Our ultimate goal is to help the linguists to set up their own survey, to collect data and to provide feedback to the participants. Our intention is that all collected data will eventually be made available on an open-source basis, making it accessible to third-party researchers.

Participants are invited to create an account for the platform (this can be facilitated via social networks) and provide basic sociolinguistic information (year of birth, gender, childhood location, and current location), following standard privacy cautions.[2] This information also allows us to entice contributors to return to the platform by informing them about new tasks. At the end of a task, the participant is invited to share their results on social media, thereby advertising the platform.

To this date, we have implemented a linguistic quiz (3.) and a geolocalisation task (4.) within the DFS platform. Both tasks give direct feedback to the participants. Such feedback is, in our view, very important, as it fosters the participant's understanding of the relevance of the task for research.

3. Linguistic quiz

In a similar vein to previous surveys on variation in European French (Avanzi et al. 2016), and on the basis of some of their data, we created a quiz in which the participants are asked to guess the meaning of regionalisms, in particular words or expressions. The quiz takes the form of a multiple choice questionnaire where a single answer is correct. An example concerning the regionalism *nareux* is given below:

(1) When one says about somebody that they are *nareux*, does this mean that that person:
   a. is picky with food?
   b. has a big nose?
   c. has a stuffy nose?
   d. has nausea?

The participant is immediately informed whether their selected answer is correct or wrong, and a short linguistic explanation is given, illustrated with a map. At the end of the quiz (i.e., after 12 questions), the participant is awarded a final score, which can be easily shared on social networks. This quiz has various aims. By its fun aspect (it is gamified with a final score, and the different answers proposed are often expressed in a humorous way), it is supposed to attract participants and make them aware of the platform in general. Through the explanations given during the quiz, it informs the participant about recent results of linguistic research. Finally, through the scores obtained by the participants, the linguist obtains a better picture of the vitality and passive knowledge of regionalisms in the population.

After a few weeks online, almost 3000 participants completed the quiz. 37% of these provided information about their linguistic origin (country and ZIP code). About the same proportion (39%) also gave sociolinguistic information such as birth year and gender. On average, participants correctly answered 6.7 out of 12 questions. Figure 1 depicts the distribution of scores for all participants.

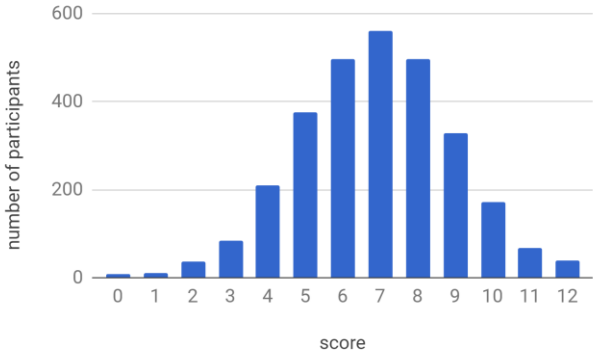


Figure 1. Distribution of scores (out of 12) for 2982 participants.

It is interesting to compare the quiz results, which represent passive knowledge about regionalisms, with results obtained in previous surveys (Avanzi et al. 2016), showing where the regionalisms are actively used (cf. the example questions in Section 4.1). Two patterns arise. Figure 2 demonstrates the first pattern, where the two maps are similar, meaning that a regionalism is not widely understood outside its area of active use.[3]

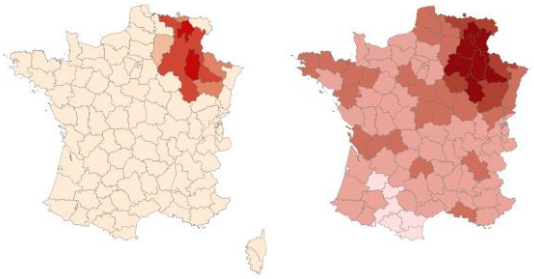


Figure 2. Active use (left, original survey with 8000 participants) vs. passive comprehension (right, crowdsourced quiz with 3000 participants) for the regionalism *nareux* 'picky'.

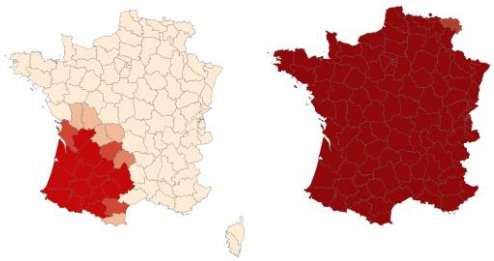


Figure 3. Active use (left, original survey with 12,000 participants) vs. passive comprehension (right, crowdsourced quiz with 3000 participants) for the regionalism *chocolatine* 'chocolate croissant'.

[2] The geolocalisation game can be played without creating an account or being signed in, in order not to raise suspicions that the sociolinguistic information will be used for the predictions.

[3] Corsica is not displayed on the quiz maps as we were not able to gather enough participants from this region.

Figure 3 illustrates the second pattern, where the two maps diverge, meaning that a regionalism is understood widely even though it is actively used only in a restricted area. In this particular case, the regionalism *chocolatine* had been the object of various lively discussions on social media over the last few years, resulting in nearly-universal comprehension in the whole French-speaking area of Europe.

Figure 4 plots the 12 quiz items according to their mean recognition rate (correct guesses) against the standard deviation of the guesses. While it shows a general tendency of lower variance with increasing mean scores (i.e., the better a regionalism is known by the participants, the higher the chances that it is known in the entire territory of inquiry), there are some noteworthy exceptions. The regionalism *péguer* 'to stick' has a fair recognition rate of about 0.6, but it has the highest standard deviation, meaning that its recognition rate varies widely across the area. On the contrary, the question about the number 80 yielded exceptionally low recognition scores, due to the particular way the question was asked.

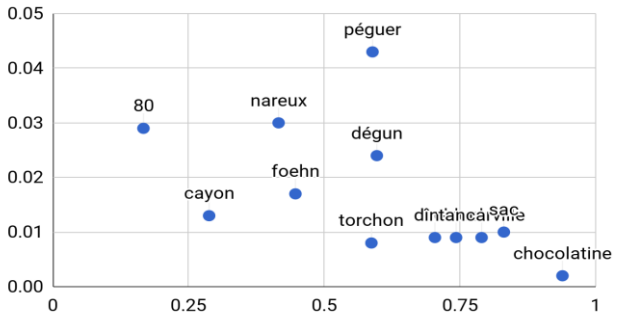


Figure 4. Mean recognition rate (horizontal axis) vs. standard deviation (vertical axis) for the 12 quiz items.

4. Geolocalisation

Using the same online surveys (Avanzi et al. 2016) as source material, we created another task focusing on geolocalisation. The main goal of this task was to provide a playful incentive to attract participants for further tasks on the DFS platform and to supplement the existing surveys with additional data points to continuously refine the accuracy of the geolocalisation task. At the same time, the semi-automatic selection of items for the task provided an interesting application of various methods of data analysis and machine learning.

There have been two major approaches to speaker geolocalisation (or dialect identification) in the literature: the **corpus-based approach** predicts the dialect of any text fragment extracted from a corpus; this approach has been followed by the VarDial shared tasks in recent years (e.g. Malmasi et al. 2016, Zampieri et al. 2017), but also by Scherrer & Rambow (2010) or Rahimi et al. (2017), for example. The **dialectological approach** tries to identify a small set of distinguishing dialectal features, which are then elicited interactively from the user in order to identify their dialect (Leemann et al. 2016, 2018a, 2018b). The task proposed here follows the dialectological approach.

The problem of geolocalisation consists in predicting the dialect/regiolect of a speaker (typically, a speaker that has not participated in the survey) by asking a set of questions (typically about a small subset of the surveyed variables). Given our motivations, the success of a geolocalisation method should not only be assessed in terms of the percentage of correct predictions, but also by its ability to entertain and surprise participants. Three parameters influence this success:

N - the number and type of questions to be asked. No more than 20 questions should be asked to keep the required attention span short.

M - the number of the areas to predict. The areas should reflect the relative scarcity of regional variation in current French, but too-large areas could make the problem look trivial and uninteresting.

A - the accuracy of the predictions. The method should obviously make as good predictions as possible, but we estimate that about 2/3 of correct predictions is required for a sustainable level of participant involvement.

In the following sections, we give some details about how we approached this optimisation problem. In other words, we wish to select the best set of questions from the previously collected survey data (with *N* being as low as possible) with the best set of prediction areas (i.e., the *M* areas being as small as possible), in order to achieve the highest accuracy *A*. In order to estimate the success of a crowdsourced geolocalisation task before its launch, we set up a simulation framework to find the optimal parameter settings.

4.1 Data

We rely on data from two surveys on regionalisms in European French (France, Belgium and Switzerland), which were carried out in 2015-2016 as a part of the project *Français de nos régions* (Avanzi et al. 2016); key information regarding the surveys is shown in Table 1.

Survey 1	Survey 2
May 2015- May 2016	September 2015 - May 2016
40 questions	90 questions
12'000 participants	8'000 participants

Table 1: Number of questions and participants in the two surveys.

Each participant was asked 40 or 90 multiple-choice questions on lexical regionalisms (small parts of the surveys also concerned morpho-syntactic and phonological variation). Some questions were illustrated by pictures. They could be direct questions of word usage (see question (2) below) or they might encompass a definition of a concept or an object (see question (3)). The number of possible answers varied from 2 to 11, and multiple answers were allowed.

(2) Do you use the word *s'entrucher*?
  a. yes
  b. no

(3) How do you call the piece of cloth that is used to wash the floor?
   a. serpillière
   b. torchon
   c. since
   d. wassingue
   e. loque
   f. pièce
   g. panosse
   h. toile
   i. chiffon
   j. lave-pont
   k. patte

4.2 Simulation framework

We applied two important pre-processing steps to the survey data. First, we settled on a set of 109 administrative areas as an upper bound for *M*: we considered 96 French departments, 7 Swiss cantons (of the French-speaking part of Switzerland, called *Romandie*), and 6 Belgian provinces (of the French-speaking part of Belgium, called *Wallonie-Bruxelles*). Although survey participants provided ZIP code information, we aggregated the subjects into 109 areas to avoid data scarcity issues in sparsely populated areas. Second, we matched participants from Survey 1 with participants from Survey 2 on the basis of their origin, leading to a total dataset of 6463 participants.

In order to evaluate different settings of the parameters *N*, *M* and *A*, we set up a simulation framework using solely the survey data in a leave-one-out fashion. The general idea is to train a model on the aggregated data of all except one participant, predict the origin of the left-out participant, and compare the prediction with the ground truth. However, contrary to a true leave-one-out setting, we chose not to remove the test participant from the training data for efficiency purposes (avoiding the need to train a new model for each participant). As the training data was aggregated and contains more than one participant for each area and question, there was never exactly the same data point in the training and test corpus, allowing us to take this methodological shortcut.

We considered two approaches to find the best parameter settings for geolocalisation, one based on clustering and shibboleth detection, and one based on feature elimination.

4.3 Clustering and shibboleth detection

This approach consisted of two steps: we first determined an optimal areal partition using hierarchical clustering, and then applied the shibboleth detection algorithm of Prokić et al. (2012) to find the most characteristic set of questions for each area.

Figure 5 shows an example of hierarchical clustering solutions using Ward's method and 10 target clusters, obtained using the complete dataset (6463 participants, 130 questions). It is worth pointing out that the aggregated data clusters nicely into geographically coherent and linguistically sensible regions, suggesting that the quality of the survey data is good.



Figure 5. Resulting areas after applying hierarchical clustering with Ward's method and ten target clusters.

The shibboleth detection algorithm was then used to list the five most characteristic linguistic variants ("shibboleths") per cluster. For example, it produced the variants *encoubler, septante, nonante, ça joue, souper* for the French Swiss area (light green area of the map), or *péguer, challer, soixante-dix, sèche-cheveux, quatre-vingt-dix* for the Provence area (cyan area of the map).

The success of this approach was limited in our case, owing essentially to two factors. First, it was very sensitive to the clustering parameters: a slight change in the number of target clusters, or a change in the clustering algorithm, led to considerable differences in simulation performance. As there are no universally applicable criteria for evaluating the quality of a hierarchical clustering, this essentially amounted to a trial-and-error process with little scientific value. Second, the core assumption of the shibboleth detection algorithm, namely that there are linguistic variants whose geographic distribution coincides with a cluster, did not seem to hold in our data. While clear regionalisms exist for Switzerland and Belgium (and are successfully identified, as shown above for the Swiss case), the inferred clusters within France are much less clearly correlated with single linguistic variants, but rather emerge through the combination of a large number of gradual linguistic differences. Due to these problems, we did not pursue this approach any further.

4.4 Feature elimination

In a second approach, we did not fix a geographical partition in advance, but kept the 109 areas as defined above while finding the optimal set of questions. For this, we applied feature elimination techniques, as detailed below. Once the questions were determined, we dynamically expanded the predictions to n-best areas or neighbours. The approach is summarised in the four following steps:

1. As the linguistic variables could have several variants with different distributions, we treated each variant separately and binarised the data from 130 n-ary variables to 639 binary variables. For example,

question (3) of section 4.1 represents an 11-ary variable (11 possible answers). This variable was converted into 11 binary variables of the form "Do you call the piece of cloth *serpillière*?", "Do you call the piece of cloth *loque*?", etc.

2. Some variants were hardly ever used or showed no geographic variation at all, so we discarded them with a single-pass feature elimination based on χ^2 score. In other words, we removed those variables that were the least statistically dependent on location. We found a lowest average distance between prediction and ground truth with 150 variants as shown in Figure 6, and settled on this value for the following steps. For the floor-cloth example, this step eliminated the variants *chiffon*, *patte* and *lave-pont*, which very few survey participants had selected.

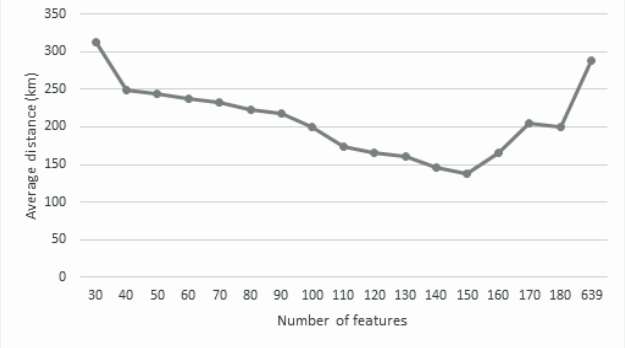


Figure 6. Average distance (in km) between predicted and actual areas as a function of the number of variables.

3. While we could have continued using the χ^2 method of step 2 to further reduce the number of variables to an acceptable value (recall that we aim for a value of N close to 20), we opted for recursive feature elimination techniques (RFE) instead, in order to take into account the dependencies between variables. Therefore, starting with the 150 binary variables of step 2, we trained a classifier and used RFE (Guyon et al. 2002) to repeatedly remove the variant that contributed least to the classification. We ran parallel experiments with two classification algorithms, SVM and MaxEnt. Both classifiers achieved much better simulation results than the χ^2 method, with SVM performing slightly better than MaxEnt (see Figure 7). We found that the χ^2 feature elimination - because it looks at each variable independently - ended up proposing a lot of variables that predict the same regional partition (e.g. Switzerland vs. France+Belgium, which is the most salient one), whereas the RFE methods yielded more complementary sets of variables. For the following steps, we settled on a smaller window of 10-40 binary variables (instead of 0-150 after step 2), as obtained by the SVM or MaxEnt methods.

4. We evaluated the simulation results in several ways. Figures 6 and 7 show average distances between the centroids of the predicted and true areas. A simpler measure is area accuracy (i.e. whether the true area has been correctly predicted or not), which is also reported below (Figure 8). We also extended area accuracy to immediate neighbours (i.e. whether the true area is equal to the predicted area or one of its neighbouring areas) and second-order neighbours.

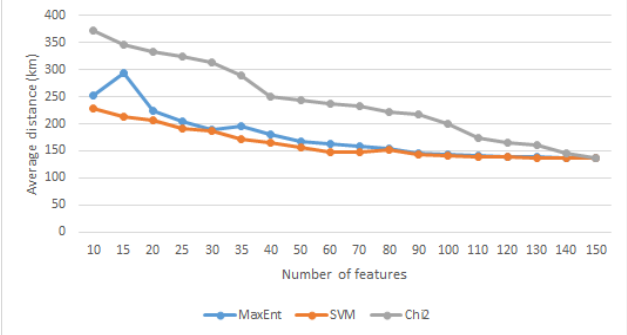


Figure 7. Average distance (in km) between predicted and actual areas as a function of the number of retained variables for SVM and MaxEnt classifier RFE and χ^2 feature elimination.

Figure 8 shows simulation results for both classifiers from which we can draw a few conclusions: first, the results stabilised at around 40 binary variables, i.e. about 30 n-ary questions. Second, extending the predictions to first-order neighbours improved accuracy by +30%, while extending them to second-order neighbours added a further +20%. With 20 variants (representing 17 n-ary questions), the accuracy score was 66.2% on second-order neighbours. This setting satisfies our target values for the variables *A* and *N*.

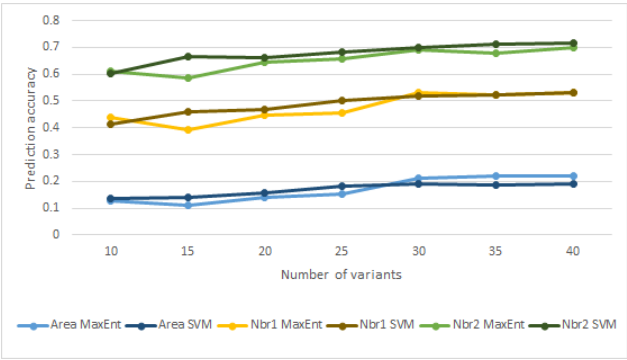


Figure 8. Prediction accuracy from 10 to 40 variants with SVM and MaxEnt classifiers and considering exact area accuracy, immediate neighbouring areas (Nbr1) and second-order neighbours (Nbr2).

4.5 Crowdsourcing implementation

The geolocalisation quiz was implemented in the DFS platform using the 15 most relevant n-ary questions as obtained by the MaxEnt and SVM RFE approaches.

With a sharing mechanism on social networks and media coverage, we were able to gather data from 8000 participants, who were led alternatively to the MaxEnt or SVM surveys. Later, we added a third survey based on a manual variant selection of 15 questions, which 500 participants completed.

Whatever the version of the quiz (MaxEnt, SVM or manual selection), a probability was computed for each of

the 109 areas after the 15 questions, and was displayed on a result map of the European French-speaking area, as in Figure 9.

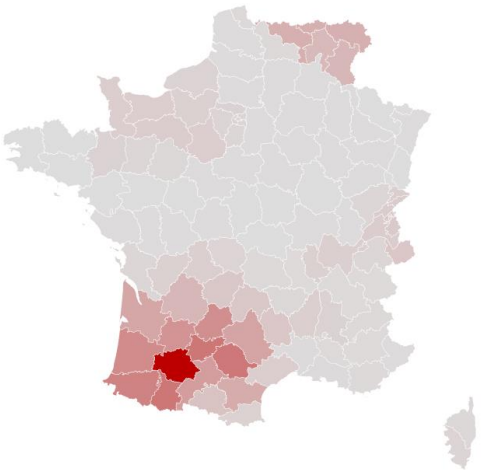


Figure 9. Example of result map of the geolocalisation presented to the participant.

Participants were also then asked for sociolinguistic information (country and ZIP code, age, gender). About 40% of participants provided these data.

4.6 Comparative results

Table 2 compares the classification results from the simulations (as in 4.4) with those for the real participants (as explained in section 4.5) who gave their true localisation information. With both automatic methods, we reached the desired accuracy threshold with comparable area sizes and number of variables (about 20). However, the variables selected by the SVM classifier intuitively corresponded better to the variation patterns observed in the original survey data.

	Part	Best	5-Best	Nbr-1	Nbr-2
RFE MaxEnt	simulated	14 %	49 %	47 %	64 %
	crowdsourced	11 %	43 %	40 %	62 %
RFE SVM	simulated	13 %	46 %	46 %	66 %
	crowdsourced	13 %	47 %	47 %	64 %
Manual selection	simulated	10 %	36 %	40 %	57 %
	crowdsourced	5 %	16 %	12 %	18 %
Random		<1%	4.5 %	4.5%	9%

Table 2. Geolocalisation results with crowdsourced and simulated data (percentages are f-scores for 109 areas).

4.7 Discussion

Our attempt to apply machine learning techniques for question (and area) selection led to a 66% correct-response rate (as in Table 2 with RFE SVM extended to second-order neighbours) which was confirmed with crowdsourced real data (64% of correct responses). The two main advantages of this automatic approach consist in optimising the selection of questions and estimating the success of a crowdsourced linguistic campaign before launch.

Although originally intended as an optimisation method for defining and optimising large areas in a region-guessing activity, we also ended up with a more fined-grained localisation (areas instead of clusters of areas) with a colour-scaled probability.

One major drawback of this approach is the dependency on earlier surveys for variable selection and simulation. Also, it proved difficult to convince the participants to fill out their sociolinguistic information after displaying the result of the quiz. As mentioned above, we did not want to prompt them to fill out the questionnaire beforehand in order not to raise suspicions.

5. Conclusion

This paper presents a new web platform for hosting activities such as linguistic surveys, including different types of foreseeable questions (text, picture or sound) and different types of answers (multiple-choice, free text, or sound recording). Although generic survey platforms already exist on the web, they are not well adapted for linguistic surveys.

We also present two activities that have been implemented on this web platform: a linguistic quiz about regionalisms and a geolocalisation task. For the latter, we compared several approaches for defining an optimised set of questions and areas. In a simulated setup, we found that a recursive feature elimination approach using a MaxEnt base classifier worked best, whereas the result of the crowdsourcing campaign showed a slight advantage for the SVM base classifier. In both cases, predicting a single area out of 109 proved difficult, but accuracy levels approached 66% when including both first- and second-order neighbours in the prediction. Also, automatic approaches to question selection turned out to work better than a linguistically informed manual selection, although the crowdsourced results for the latter should be taken with a pinch of salt due to the relatively low number of participants.

The presented framework could easily be localised and adapted to other languages. Also, provided that source surveys are available, it can easily be adapted to other French-speaking areas, such as parts of Africa or Canada, with minor adaptations to the maps.

Our next task is to extend the platform so that linguists can set up their own surveys. Moreover, additional information will be collected from the participants, such as educational level and job typology, in order to compare diatopic and diastratic variation patterns.

6. Acknowledgements

This research is funded by DGLFLF (Délégation générale à la langue française et aux langues de France) and supported by these academic partners: University of Geneva (Switzerland), University of Helsinki (Finland), UCLouvain (Belgium), University of Strasbourg (France), LIMSI-CNRS (France), and ATILF-Université de Lorraine (France).

7. Bibliographical References

Avanzi, M., C. Barbet, J. Glikman, J. Peuvergne (2016). *Présentation d'une enquête pour l'étude les régionalismes du français.* Actes du 5ème congrès mondial de linguistique française (CMLF). Tours, France, 1-15.

Cook, M., J. Barker & Lecumberri, M. L. G. (2013). *Crowdsourcing in Speech Perception*. In: Eskénazi, M., Levow, G.A., Meng, H., Parent, G. & Suendermann, D. (eds), *Crowdsourcing for Speech Processing: Applications to Data Collection, Transcription and Assessment*. Hoboken: John Wiley & Sons, 137-172.

Goldman J.Ph. et al. (2018). *Strategies and Challenges for Crowd-Sourcing Dialect Perception Data [for Swiss German and Swiss French]*. LREC Conference, Miyasaki, Japan.

Guyon I., J. Weston, S. Barnhill, V. Vapnik (2002). *Gene selection for cancer classification using support vector machines.* Machine Learning, 46(1-3): 389-422.

Krefeld, T., S. Lücke (2014). *VerbaAlpina - Der alpine Kulturraum im Spiegel seiner Mehrsprachigkeit.* Ladinia XXXVIII, 189-211.

Leemann, A., M.-J. Kolly, R. Purves, D. Britain, E. Glaser (2016). *Crowdsourcing language change with smartphone applications.* PLOS ONE.

Leemann, A., M-J. Kolly, D. Britain (2018a). *The English Dialects App: the creation of a crowdsourced dialect corpus.* Ampersand 5, 1-17.

Leemann, A., S. Elspaß, R. Möller, T. Grossenbacher (2018b). *Grüezi, Moin, Servus – Wie wir wo sprechen.* Hamburg: Rowohlt.

Malmasi, S., M. Zampieri, N. Ljubešić, P. Nakov, A. Ali, J. Tiedemann (2016). *Discriminating between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task.* Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), Coling. Osaka, Japan, 1-14.

Möller, R., S. Elspaß (2015): *Atlas zur deutschen Alltagssprache*. In: Kehrein, R., A. Lameli, S. Rabanus (eds.): *Regionale Variation des Deutschen – Projekte und Perspektiven.* Berlin, Boston: de Gruyter, 519-540.

Prokić, J., Ç. Çöltekin, J. Nerbonne (2012). Detecting Shibboleths. Proceedings of the EACL 2012 Joint Workshop of LINGVIS & UNCLH. Avignon, France, 72-80.

Rahimi, A., T. Baldwin, T. Cohn (2017). *Continuous Representation of Location for Geolocation and Lexical Dialectology using Mixture Density Networks.* Proceedings of EMNLP. Copenhagen, Denmark, 167-176.

Scherrer, Y., O. Rambow (2010). *Word-based Dialect Identification with Georeferenced Rules*. Proceedings of EMNLP. Cambridge, MA, 1151-1161.

Vaux, B., S. Golder (2003). *The Harvard Dialect Survey*. Cambridge, MA: Harvard University Linguistics Department.

Zampieri, M., S. Malmasi, N. Ljubešić, P. Nakov, A. Ali, J. Tiedemann, Y. Scherrer, N. Aepli (2017). Findings of the VarDial Evaluation Campaign 2017. Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), EACL. Valencia, Spain, 1-15.

Improving Machine Translation of Educational Content via Crowdsourcing

Maximiliana Behnke[1], Antonio Valerio Miceli Barone[1], Rico Sennrich[1], Vilelmini Sosoni[2], Thanasis Naskos[3], Eirini Takoulidou[3], Maria Stasimioti[2], Menno van Zaanen[4], Sheila Castilho[5] Federico Gaspari[5], Panayota Georgakopoulou[6] Valia Kordoni[7], Markus Egg[7], Katia Lida Kermanidis[3]

[1]School of Informatics, The University of Edinburgh
[2]Department of Foreign Languages, Translation and Interpreting, Ionian University
[3]Department of Informatics, Ionian University
[4]Department of Communication and Information Sciences,Tilburg University
[5]ADAPT Centre, Dublin City University
[6]Deluxe Media Europe
[7]Humboldt-Universität zu Berlin, Germany
maximiliana.behnke@ed.ac.uk; amiceli@inf.ed.ac.uk; rico.sennrich@ed.ac.uk; vilelmini@hotmail.com; anaskos@ionio.gr; rinoulit@gmail.com; stasimioti.maria@gmail.com; mvzaanen@uvt.nl; sheila.castilho@adaptcentre.ie; federico.gaspari@adaptcentre.ie; yota.georgakopoulou@bydeluxe.com; kordonie@anglistik.hu-berlin.de; markus.egg@anglistik.hu-berlin.de; kerman@ionio.gr

Abstract

The limited availability of in-domain training data is a major issue in the training of application-specific neural machine translation models. Professional outsourcing of bilingual data collections is costly and often not feasible. In this paper we analyze the influence of using crowdsourcing as a scalable way to obtain translations of target in-domain data having in mind that the translations can be of a lower quality. We apply crowdsourcing with carefully designed quality controls to create parallel corpora for the educational domain by collecting translations of texts from MOOCs from English to eleven languages, which we then use to fine-tune neural machine translation models previously trained on general-domain data. The results from our research indicate that crowdsourced data collected with proper quality controls consistently yields performance gains over general-domain baseline systems, and systems fine-tuned with pre-existing in-domain corpora.

Keywords: MOOCs, neural machine translation, crowdsourcing

1. Introduction

The European Union Horizon 2020 **TraMOOC** project (Translation for Massive Open Online Courses) aims at enhancing multilingual access to online education by providing machine translation solutions for the educational content available in MOOCs, i.e. video lecture subtitles, slides, assignments, quiz text, and course discussion forum text (Kordoni et al., 2016). Educational content is translated from English into eleven European and BRIC languages (Bulgarian, Chinese, Croatian, Czech, Dutch, German, Greek, Italian, Polish, Portuguese, and Russian).

This specific domain, i.e. the content of online courses, imposes a set of challenging properties, such as extensive use of domain-specific terms and entities, frequent occurrence of unknown words, subtitle segmentation, spoken language characteristics and social web text properties. Achieving high-quality machine translation in these conditions therefore requires significant amounts of in-domain data for training and testing. Creating such data by hiring professional translators would be expensive, especially considering that it would have to be done for eleven target languages. Therefore we turn our attention to crowdsourcing as a cost-saving alternative.

The impact of non-expert input, i.e. translations provided by non-professional translators, on the development and the evaluation of machine translation engines has been investigated in previous research (Callison-Burch, 2009; Zaidan and Callison-Burch, 2011; Ambati, 2012). Crowdsourcing has been used to this end, and the main research concern has been whether input by the general crowd, that has no expertise in linguistics or translation studies, can improve the quality of large-scale machine translation systems in a manner that is cost- and time-effective.

In this work we use crowdsourcing with carefully designed quality controls to collect translations of MOOC material from English to the eleven project target languages. We combine this data with pre-existing in-domain educational data and we use it to build translation systems in a transfer learning approach: we first train neural machine translation systems on general-domain data and then fine-tune them with the collected in-domain data. We report large improvements of translation quality obtained by using the crowdsourced data over both the general-domain baselines and the models tuned only with the pre-existing in-domain data.

2. Crowdsourcing Translations of Online Educational Content on a Large Scale

In order to facilitate worker recruitment and task implementation, we selected an established commercial crowdsourcing platform which had a crowd of workers who were already members and categorized by their experience on the platform. Demographic information (country, language) pertaining to the crowd, that enables crowd selection, was taken into account. We selected Crowdflower[1], due to logistic (e.g. payment options) as well as technical reasons (configurability, reliability, size of crowd channels, quality control mechanisms, technical support).

2.1 Data Description

Texts from courses of MOOC providers Coursera[2] and Iversity[3] constituted the in-domain English data sources to

[1] https://www.crowdflower.com/
[2] https://www.coursera.org/

be translated via crowdsourcing, as well as the QCRI Educational Domain Corpus (QED)[4] and the Videolectures.net online educational video library[5]. The course contents varied from technical (e.g. Finance) to non-scientific (e.g. Future of Storytelling). From these datasets, texts in two types of text genres were identified, namely formal (lecture subtitles, slides, assignments, quizzes) and informal forum discussions. More details pertaining to the data sources and their processing can be found in Sosoni et al. (2018).

The formal text presented a high occurrence of domain-specific terms and expressions, as well as spontaneous speech characteristics, i.e. repetitions, interjections, truncated clauses, segments marked as inaudible. This is not surprising, as a significant part of this type of text consists of subtitles. Informal text has the idiosyncrasies of social web text, i.e use of slang, misspellings, lexical variants, abbreviations, etc. as forum posts are treated like social network interactions. Furthermore, to a large extent, MOOC students are non-native speakers of English and their language skills vary (DeBoer et al., 2013).

Bearing in mind that the crowdsourced translations were to be used for training, tuning and testing the translation engines for the eleven language pairs involved in TraMOOC, the project aimed at collecting a significant number of translated segments. Approximately 95,000 segments were chosen to be translated per language pair; Table 1 shows the data source distribution in more detail.

	Training	Testing and tuning
Iversity	30000	2500
Videolectures.net	-	2500
Coursera	27000	-
QED	23000	-

Table 1: Data source distribution in number of (English) segments.

The segments constituting the testing and tuning datasets were to be translated by at least two workers each, for redundancy purposes.

2.2 Running and Monitoring the Experiment

Workers were provided with detailed language-independent, as well as language-specific instructions that explained how to cope with the various linguistic phenomena present in the source datasets. Instructions were very explanatory and included specific examples and ways to cope with typical issues. Test questions (which were based on choosing the optimal translation among three options) were designed to help evaluate a worker's performance and ban spammers during the data collection process. The copy functionality was disabled in order to discourage workers from copy-pasting output from online MT systems. Pilot trials were subsequently run on a sample of the dataset and a subset of the languages to help configure the settings of the crowdsourcing task, such as the number of segments to be displayed per page, the minimum and maximum time limits that a worker would be allowed to work on one page, the acceptable worker accuracy threshold, etc.

After further parameterisation of the crowdsourcing platform following the pilot trials, the translation crowdsourcing task was run for all language pairs and the entire dataset. In order to control spammers, as well as provide a training opportunity to trustworthy workers (who needed to familiarize themselves with the text domain and genre), the main task consisted of two phases: a quiz mode, where workers were asked to pass an evaluation microtask, and a work mode, where the actual translations were submitted.

Settings were adjusted and adapted to meet the needs of every language pair, as worker flow varied significantly among languages. The number of test questions (90-300), their difficulty level, the timeframes for task completion, the worker fee (0.04-0.08/segment), the countries that the task was open to, the workers' experience level (according to contributors' categorization in experience levels 1/inexperienced and 2/experienced by CrowdFlower), were all contingent on the flow of each language. The main translation collection task was run from March until June 2017.

Close and constant monitoring of the workers' annotation process was crucial for identifying malicious behaviour among workers, keeping track of the workflow and ensuring quality translations. The monitoring process was automated to a large extent in order to optimally handle multiple language pairs. Automated software tools were developed in order to keep track of the time spent by each worker per page, the page submission time, worker accuracy level, their geographic location, and the difference in length between the source and the translated sentences (a difference of more than 60% was assumed to indicate malicious input).

2.3 Results

After banning spammers and filtering malicious translations, the number of trusted annotations varied significantly among languages. This difference occurred due to the difference in the workflow rate among the languages from the respective Crowdflower-supported crowd channels, as well as the availability of workers for specific languages in the crowd. Figure 1 shows the number of trusted worker judgements collected for every language for the training and testing, as well as the tuning datasets.

We find that we were able to create parallel data totalling around 1 million segments, or 10 million words, for a crowdsourcing budget of approximately 45,000 EUR. We estimate that the creation of the corpus would have costed well over 10 times as much as using professional translators (assuming a cost of 0.05-0.07 EUR per word), and well over 40 times as much as subcontracting the work to a Language Service Provider that did not use a machine translation post-editing workflow (assuming average pricing of 0.21 EUR/word). This calculation does not consider overhead costs, which exist in different forms in all scenarios. We argue that some of the disadvantages of using crowdsourcing for translation, such as lower quality expectations, and the lack of guarantee that the full

[3] https://iversity.org/
[4] http://alt.qcri.org/resources/qedcorpus
[5] http://videolectures.net/

text will be translated within a specific time frame, are acceptable for our use case of creating domain-specific training data for a machine translation system, and that crowdsourcing is a cost-effective method for this purpose.

The corpus will be made available through the EU (according to the H2020 Open Research Data Pilot) for research purposes after the end of the project, and taking into account copyright restrictions imposed by each source.

3. The Impact on Machine Translation Performance

Our main success criteria of the crowdsourcing effort is whether the collected data can effectively improve the translation quality of a machine translation system in the target domain. In this section, we describe the translation systems built within the TraMOOC project and experimentally validate the effectiveness of the crowdsourcing effort by measuring its effect on the translation quality of machine translation systems.

3.1 Methodology

Our translation systems are built in two steps: for each language pair, we first train a baseline system on a large "mixed-domain" dataset, using data from various sources. Then, we fine-tune this system on an "in-domain" dataset representative of MOOC materials. This domain adaptation step is performed either using only pre-existing in-domain data, or using both the pre-existing data and the crowdsourced data that we collected.

We report performance of three systems for each language pair:

- a mixed-domain baseline system;
- a system adapted towards pre-existing in-domain data;
- a system adapted towards both existing and crowdsourced in-domain data.

We measure and report the translation quality of these systems with the BLEU metric. Evaluation results are computed on held-out test sets that were also created via crowdsourced translations. For Chinese, we compute character-level BLEU. For all other language pairs, the evaluation was de-tokenized and case-sensitive.

3.2 Training Data

For training, we collected bilingual corpora for use in our baseline systems. The following data sets are considered mixed-domain training data:

- Europarl (Koehn, 2005);
- JRC-Acquis 3.0 (Steinberger et al., 2006);
- DGT's Translation Memory (Steinberger et al., 2012) as distributed in OPUS (Tiedemann, 2012);
- OPUS European Central Bank (ECB);
- OPUS European Medicines Agency (EMEA);
- OPUS EU Bookshop;
- OPUS OpenSubtitles 7;
- WMT News Commentary;
- WMT CommonCrawl;
- Chinese WMT training data;
- Wikipedia names and titles (English-Russian);
- SETimes (Tyers and Alperen, 2010);
- Yandex English-Russian Parallel Corpus[6];
- The United Nations Parallel Corpus v1.0 (English-Chinese) (Ziemski et al., 2016);
- CzEng v1.6pre 8;
- Croatian-English parallel corpus hrenWaC 2.0 (Ljubešic et al., 2016).

We consider the following data sets to be pre-existing in-domain data sets for the purpose of MOOC translation:

- TED from WIT3 (Cettolo et al., 2012);
- QCRI Educational Domain Corpus (QED) (Abdelali et al., 2014);
- Parallel data provided by Coursera;
- Web-crawled data collected in the TraMOOC project.

The amount of pre-existing in-domain data differs greatly between languages, ranging from tens of thousands to millions segment pairs, while the crowdsourced data that we collected is in the order of the tens of thousands (Table 2). The amount of mixed-domain training data ranges between 20 and 60 million segment pairs per language, 100-1000 times the amount of crowdsourced in-domain training data.

	Pre-existing	Crowdsourced
en-bg	63000	54000
en-cs	2177000	46000
en-de	258000	48000
en-el	124000	66000
en-hr	89000	80000
en-it	336000	86000
en-nl	226000	34000
en-pl	246000	59000
en-pt	575000	74000
en-ru	2301000	69000
en-zh	647000	18000

Table 2: Amount of in-domain parallel training data (segment pairs) per language pair for domain adaptation.

3.3 Machine Translation Systems

Our baseline translation systems are GRU attentive sequence-to-sequence neural machine translation models (Bahdanau, 2015).

For training, we used the same configuration as the Edinburgh's submission to the WMT-17 news translation task (Sennrich et al., 2017), which provides a strong baseline.

[6] https://translate.yandex.ru/corpus

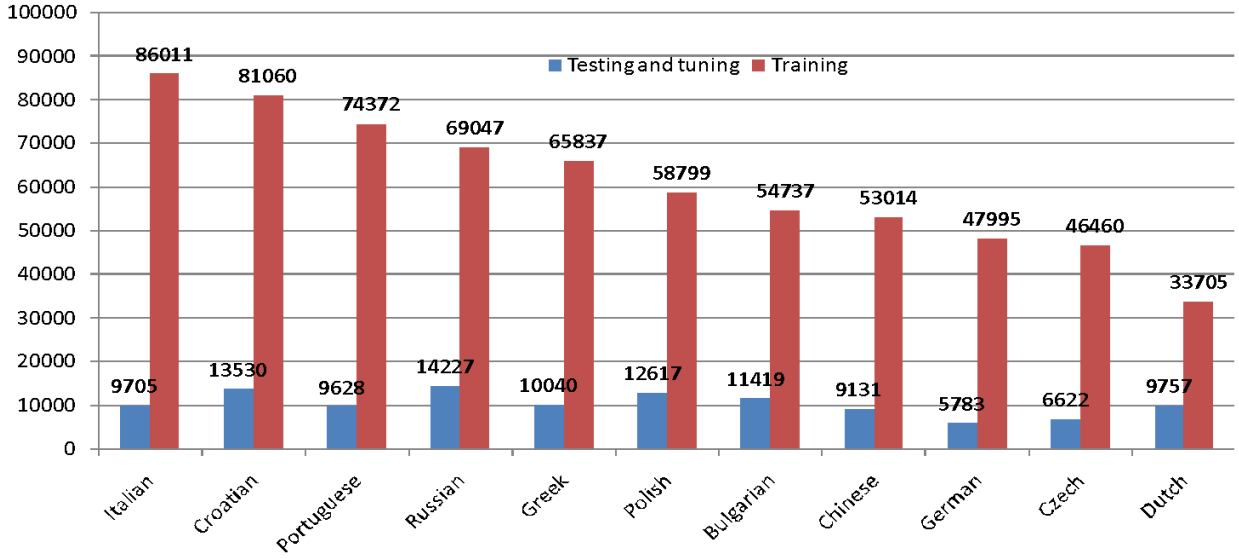


Figure 1: Number of trusted segments collected for each target language for training and testing/tuning.

We adapt the baseline systems to the in-domain MOOC data using continued training of the baseline system with MAP-L2 and dropout regularization (Miceli Barone et al. 2017).

3.4 Results

Evaluation results are shown in Table 3. We can see that domain adaptation via fine-tuning is effective for all language pairs.

	Baseline	+ Preexisting	+ Crowdsourced
en-bg	22.91	23.57	25.89
en-cs	29.86	31.06	32.06
en-de	29.29	32.14	33.69
en-el	35.54	38.01	40.76
en-hr	23.36	23.70	26.43
en-it	32.15	36.19	38.53
en-nl	35.59	38.04	40.07
en-pl	27.16	28.41	30.97
en-pt	39.44	47.68	48.71
en-ru	26.41	29.08	29.78
en-zh	27.93	28.51	29.77
avg	29.97	32.40	34.24

Table 3: Translation quality (BLEU) of baseline system, and systems adapted to domain with/without crowdsourced data.

Using only pre-existing in-domain data, we obtain improvements ranging from 0.34 BLEU (en-hr) to 8.24 BLEU (en-pt), and an average improvement of 2.43 BLEU.

As our main result we report the effect of the crowdsourced in-domain data on translation quality: we find consistent and strong improvements, ranging from 0.7 BLEU (en-ru) to 2.75 BLEU (en-el) with an average improvement of 1.84 BLEU over the systems with only pre-existing in-domain data, and an average 4.27 BLEU over the mixed-domain baselines.

There are various factors which affect the effectiveness of domain adaptation with crowdsourced in-domain data. Regarding the correlation between the amount of in-domain data for fine-tuning and translation quality, we note that Miceli Barone et al. (2017) found an approximately log-linear combination between the two, using random subsets of in-domain data of different size. In our experiments, this relationship is confounded by the fact that the baseline models are of varying quality, and that we have access to varying amounts of pre-existing data that we treat as "in-domain", with varying distance to our actual target domain.

Despite all confounding variables, if we consider that the amount of crowdsourced training data is 100-1,000 times smaller than the amount of out-of-domain training data used, and consistently smaller than the amount of pre-existing in-domain data, we conclude that the improvements that we observe from adding crowdsourced data cannot just be attributed to having more training data available. Based on the log-linear learning curves reported in related work (Koehn and Knowles, 2017; Miceli Barone et al. 2017), we would expect small or negligible improvements in translation quality if we added the same amount of out-of-domain training data, or pre-existing in-domain data, to the systems without crowdsourced data. This confirms the relevance of obtaining in-domain training data that is similar in terms of domain and genre

to the texts that are to be translated, and that the crowdsourced training data is of high value to the MT system.

4. Conclusion

We collected crowdsourced translations from English to eleven languages to create a parallel corpus for the educational domain.

We experimentally showed that using this data to train neural machine translation systems by means of domain adaptation provides large quality improvements, even on top of systems adapted using only existing in-domain translations. These results highlight the importance of in-domain training data for machine translation: even a small amount of crowdsourced translations, that may be noisy in nature, has a large positive impact on translation quality.

In conclusion, we show that crowdsourcing with proper quality controls is a viable and cost-effective way of creating valuable in-domain parallel resources for machine translation.

Acknowledgements

This work was done as part of the TraMOOC project (Translation for Massive Open Online Courses) funded by the European Commission under H2020-ICT-2014/H2020-ICT-2014-1 under grant agreement number 644333. This work was supported by grant EP/L01503X/1 for the University of Edinburgh School of Informatics Centre for Doctoral Training in Pervasive Parallelism from the UK Engineering and Physical Sciences Research Council (EPSRC).

References

Abdelali, A., Guzman, F., Sajjad, H., and Vogel, S. (2014). The AMARA Corpus: Building Parallel Language Resources for the Educational Domain. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland. European Language Resources Association (ELRA).

Ambati, V. (2012). Active learning and crowd-sourcing for machine translation. PhD Thesis. Carnegie Mellon University. ISBN: 978-1-267-58215-7.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. Proceedings of the International Conference on Learning Representations (ICLR).

Callison-Burch, C. (2009). Fast, cheap, and creative: evaluating translation quality using Amazon's Mechanical Turk. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1. Association for Computational Linguistics, pages 286-295.

Cettolo, M., Girardi, C., and Federico, M. (2012). WIT^3: Web Inventory of Transcribed and Translated Talks. In Proc. of the Annual Conf. of the European Assoc. for Machine Translation (EAMT), pages 261-268, Trento, Italy.

DeBoer, J, Stump, G., Breslow, L, and Seaton, D. (2013). Diversity in MOOC Students' Backgrounds and Behaviors in Relationship to Performance in 6.002x. Proceedings of the sixth learning international networks consortium conference.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In MT Summit X, Phuket, Thailand.

Koehn, P., Knowles, R. (2017). Six Challenges for Neural Machine Translation. Proceedings of the First Workshop on Neural Machine Translation, pages 28-39. Association for Computational Linguistics.

Kordoni, V., van den Bosch, A., Kermanidis K., Sosoni V., Cholakov, K., Hendrickx, I., Huck, M., and Way, A. (2016). Enhancing Access to Online Education: Quality Machine Translation of MOOC Content. Proceedings of the International Conference on Language Resources and Evaluation, pages 16-22, Portoroz, Slovenia.

Ljubešic, N., Esplà-Gomis, M., Ortiz Rojas, S., Klubíčka, F., and Toral, A. (2016). Croatian-English parallel corpus hrenWaC 2.0. Slovenian language resource repository CLARIN.SI.

Miceli Barone, A. V., Haddow, B., Germann, U., and Sennrich, R. (2017). Regularization techniques for fine-tuning in neural machine translation. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Barone, A. V. M., and Williams, P. (2017). The University of Edinburgh's Neural MT Systems for WMT17. Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers. Copenhagen, Denmark.

Sosoni, V., Kermanidis, K.L., Stasimioti, M., Naskos, T., Takoulidou, E., van Zaanen, M., Castilho, S., Georgakopoulou, P., Kordoni, V., and Egg, M. 2018. Translation Crowdsourcing: Creating a Multilingual Corpus of Online Educational Content. In Proc. of the Int. Conf. on Language Resources and Evaluation (LREC), Miyazaki, Japan. (to appear).

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlüter, P. (2012). DGT-TM: A freely Available Translation Memory in 22 Languages. In Proc. of the Int. Conf. on Language Resources and Evaluation (LREC) , pages 454-459, Istanbul, Turkey.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tus, D., and Varga, D. (2006). The JRC-Acquis: A Multilingual Aligned Parallel Corpus with 20+ Languages. In Proc. of the Int. Conf. on Language Resources and Evaluation (LREC), pages 2142-2147, Genoa, Italy.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'2012).

Tyers, F. M. and Alperen, M. S. (2010). South-East European Times: A parallel corpus of Balkan languages. In Proc. of the LREC Workshop on Exploitation of Multilingual Resources and Tools for Central and (South-) Eastern European Languages , pages 49-53, Malta.

Zaidan, O. F., and Callison-Burch, C. (2011). Crowdsourcing translation: Professional quality from non-professionals. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, pages 1220-1229. Association for Computational Linguistics.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The United Nations Parallel Corpus v1. 0. In Language Resources and Evaluation (LREC 16). Portorož, Slovenia, May 2016.

Grounding Gradable Adjectives through Crowdsourcing

Rebecca Sharp, Mithun Paul, Ajay Nagesh, Dane Bell, and Mihai Surdeanu
CLU Lab, Dept. of Computer Science, University of Arizona
{bsharp, mithunpaul, ajaynagesh, dane, msurdeanu}@email.arizona.edu

Abstract

In order to build technology that has the ability to answer questions relevant to national and global security, e.g., on food insecurity in certain parts of the world, one has to implement machine reading technology that extracts causal mechanisms from texts. Unfortunately, many of these texts describe these interactions using vague, high-level language. One particular example is the use of gradable adjectives, i.e., adjectives that can take a range of magnitudes such as *small* or *slight*. Here we propose a method for estimating specific concrete groundings for a set of such gradable adjectives. We use crowdsourcing to gather human language intuitions about the impact of each adjective, then fit a linear mixed effects model to this data. The resulting model is able to estimate the impact of novel instances of these adjectives found in text. We evaluate our model in terms of its ability to generalize to unseen data and find that it has a predictive R^2 of 0.632 in general, and 0.677 on a subset of high-frequency adjectives.

Keywords: grounded semantics, gradable adjectives, crowdsourcing

1. Introduction

In order to understand the interplay of various entities and events in complex systems such as climate change or crop yields, scientists make use of complex quantitative models (e.g., DSSAT (Jones et al., 2003) or AGMIP (Rosenzweig et al., 2013)) with qualitative hypotheses (e.g., Parry and Rosenzweig (1993), Zhang el al. (2011), Zhao et al. (2017), *inter alia*). Typically these models must be hand-generated by domain experts through an expensive process that requires extensive literature review and a large amount of time, resulting in only a small fraction of the available information being processed and incorporated into the models. These models are crucial to predicting the vagaries in such complex systems. A timely and accurate assessment of the factors that affect such systems bears directly on the health of the nation and environment (Elliott et al., 2017; Porwollik et al., 2017). Automated machine reading can help create and validate hypotheses using these models, but there remains a disconnect – often relevant events are written in high-level language, yet the model requires specific quantities. In particular, when describing events, authors often make use of *gradable adjectives*, i.e., adjectives (such as *small*) that can take a range of magnitudes or degrees (e.g., something can be a little small, very small, extraordinarily small, etc.)

When used to describe important changes in model parameters, as with this snippet from a scientific publication: *"...doubling of the atmospheric carbon dioxide concentration will lead to only a **small decrease** in global crop production."* (Parry and Rosenzweig, 1993), the ability to quantify or ground such adjectives is a critical step for automated machine reading. For example, consider the difference between a *small increase in rainfall* versus a *severe increase in rainfall* when attempting to predict the funds needed for disaster relief or the potential impact on the expected yield of a given crop.

Here we propose a method for quickly and concretely grounding a large number of gradable adjectives such that their effect can be calculated for entities or events of interest. Specifically, we gather human intuitions about the effect of a particular gradable adjective on a given distribution (i.e., mean and standard deviation), independent of the item being modified, and then use this data to fit a linear model. Gradable adjectives are often classified in terms of scales, e.g., *hot, warm*, and *cool* are all in the temperature scale while *large* and *small* describe magnitude instead. Here we focus on these magnitude adjectives as a use case, though we suggest that the method can be straightforwardly extended to other scales.

The resulting resource we provide[1] consists of a linear model for each adjective that takes as input the typical distribution of the item being modified and in turn provides the predicted size of the change. Turning once again to the example above, for a region with a average rainfall of 40 inches/year (±6 in), if we need to ground a *small* increase, the model would return a predicted rainfall of 40.54 inches. This can be compared, for example, to 45.26, which is the model's prediction for a *large* increase.

Our specific contributions are:

- We provide a method for using human language intuitions about the semantic meaning of gradable adjectives to create a viable model for the adjective semantics. We decouple the meaning of the adjective from the noun being modified to get a truer semantic understanding of the adjective itself, and we show that the model predictions are well correlated with the human judgments.
- Using cross validation, i.e., testing on crowdsourced predictions not seen in training but whose adjectives were seen in training, we show that we can achieve a predictive R^2 of 0.632 when using all adjectives. When we use a smaller subset of higher frequency adjectives, we can fit a higher precision model that has a predictive R^2 of 0.677. One limitation of this model is that it is unable to make predictions on novel, or unseen, adjectives. We address this with an initial neu-

[1] All materials, data, and code used are available at `https://github.com/clulab/releases/tree/master/lrec2018-gradable`

ral network model based on word embeddings. On unseen adjectives this model has a predictive R^2 of 0.244.

- We release the resulting database of 98 adjectives and their corresponding linear models as a domain-agnostic resource. This resource could potentially facilitate the quantification of extracted entities and events that contain gradable adjectives for use in downstream tasks, such as modeling complex systems and predicting real-world events. To address a variety of future use cases, we break this resource down into several versions: the full version with all adjectives that emphasizes recall, a smaller subset that emphasizes precision, and a second version of each of these that depends only on the adjective (i.e., for when the typical distribution of the noun being modified is unknown).

2. Related Work

Gradable adjectives have been experimentally demonstrated to be interpreted in part based on the semantics of the nouns they modify (i.e., a *small* mouse versus a *small* building) (Bonini et al., 1999; Alxatib and Pelletier, 2011; Bylinina, 2014). For this reason, we test our adjectives using legal non-words (e.g. *mards*), for which we provide a typical size distribution, and we include the provided distribution in the linear model. This allows us to model the semantics of the adjective in context, while removing the need to test each adjective along with each possible noun it could modify.

Whitman et al. (2003) propose a model for finding perceptual groundings for adjectives. That is, they ground their adjectives using audio features and are then able to predict a lexical description of unheard music. This is quite similar to what we do in spirit, however our groundings are numerical rather than perceptual. Several works have attempted to link gradable adjectives with numerical quantities that co-occur in the context of the gradable adjective mention (Shivade et al., 2016; Narisawa et al., 2013). However, this dependence on corpus resources to find evidence for gradability requires complex information extraction and suffers greatly from sparsity, especially when attempting to ground adjectives in a new domain. Additionally, it results in a solution that is highly domain-specific (Shivade et al., 2016).

Kim and de Marneffe repurposed neural network language models by using word embeddings to rank gradable adjectives (2013). Bakhshandeh and Allen (2015) use bootstrapping to discover properties of adjectives, including what they can modify. However, rather than simply ranking adjectives or extracting their attributes, here we focus on determining a concrete, numerical grounding for each.

There have also been recent works that use crowdsourcing to determine gradability of adjectives. For example, Qing and Franke (2014) use crowdsourcing to gather intuitions about the interpretation of gradable adjectives, but they are testing whether or not adjective usage corresponds to optimal language use, whereas our resource grounds gradable adjectives. Accordingly, they test only four adjectives using visual cues, while we test 98 adjectives using numerical cues.

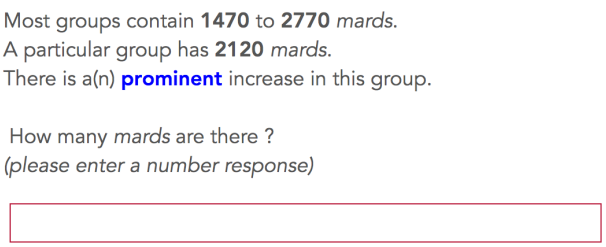


Figure 1: Example prompt given to Amazon Mechanical Turk workers to elicit the impact of gradable adjectives. Workers were given a specific distribution (of an imaginary item) and asked for the increase they perceived from the given adjective. The full set of prompts is included with our release.

The work by Wilkinson and Tim (2016) is closest to our work. They use crowdsourcing to create ranked lists of gradable adjectives that correspond to a variety of different scales (e.g., temperature, dimension, speed, and so on). Unlike Wilkinson and Tim, we use only one scale (magnitude) and again, we are interested in creating a concrete grounding for adjectives rather than a ranking.

3. Approach

While humans often use high-level language to describe events, models of the interactions between these events require specific quantitative information. To bridge the gap between gradable adjectives and this quantified representation, we use human language intuitions about an adjective's impact on a given distribution to fit a linear model. With this model, then, we can predict the impact of the adjective on an entity whose distribution is known.

Specifically our approach operates in two parts: gathering the human language intuitions for each adjective using crowdsourcing (described in Section 4.) and then fitting a linear mixed effects model to the data (described in Section 5.). The resulting model allows us to make predictions about the effects of one of our grounded adjectives on unseen nouns.

4. Data

We first gathered a set of 98 gradable adjectives from the Collins Birmingham University International Language Database (COBUILD) dictionary (Sinclair and others, 1987) that were determined to be particularly relevant to use cases focusing on national and global security (e.g., food insecurity) and able to be evaluated with a common methodology. We then used Amazon Mechanical Turk (MTurk) (Buhrmester et al., 2011) to gather our data for each of these adjectives. Our MTurk task was designed to test the amount by which a given adjective was perceived to change a known quantity. As discussed in Section 2., the perceived impact of a gradable adjective has been shown to be highly dependent on the typical distribution of the item it is modifying (Bonini et al., 1999; Alxatib and Pelletier, 2011; Bylinina, 2014). For example, an extra 2 inches of rain would be insignificant in a tropical location, but in the

desert it would be a large change. However, since we want a model that can be used in a range of contexts, we designed our experiment to decouple the adjective semantics from the noun semantics by using non-words for the items being modified (in the style of a "Wug Test" (Berko, 1958)) and providing MTurk workers (turkers) with a typical distribution of the item in question, as shown in the example prompt in Figure 1. We then ask the turkers to describe the effect of the adjective in question on the group size. This response forms the basis for the dependent variable used in our model building.

We required the turkers to be in the United States and informed them that they needed to be native speakers of English. To demonstrate this, they were required to correctly answer a language-based question in order to participate. They were given two attempts (i.e., a second question was shown to them if they did not correctly answer the first). For the task, each turker was asked to provide responses to 16 prompts and was compensated with $0.75, based on the average of 20 seconds per prompt.

5. Model Building

5.1. Model Factors

As interpretations of gradable adjectives are context-dependent (Section 2.), in addition to the factor of interest (i.e., adjective) we include in our model the shape of the distribution for the item being modified using two control factors: the provided mean (μ_p) and provided standard deviation (σ_p). We calculate σ_p directly from the typical range (e.g., 1470 to 2770 in the example in Figure 1), which we consider to be ± 2 standard deviations.[2] Thus, for the above example, $\sigma_p = (2770 - 1470)/4 = 325$. The value for the particular group given to turkers (i.e., 2120) lies in the middle of the range and so we use that directly for μ_p[3]. We chose not to add the interactions between these factors and adjective due to the large number of degrees of freedom (recall that adjective has 98 levels) in an effort to reduce the likelihood of overfitting.

In addition to differences based on context, gradable adjectives have also been shown to be interpreted differently by different individuals (Raffman, 1994; Raffman, 1996; Shapiro, 2006). To account for this, we elected to fit a linear mixed effects model to our data, as this allows us to include a random intercept for each turker. In effect, this means that while we are fitting a linear model with adjective, σ_p, and μ_p as fixed effects, we allow the fitted line to have a different intercept for each turker, thereby accounting for individual biases.[4] While it is possible in this framework to also have random slopes for each adjective, here we refrained so as to avoid a large increase in model complexity. The dependent variable (i.e., what the model is trying to predict) is the response given by the turkers, normalized by μ_p and σ_p into something very similar to a z-score:

$$respDev = \frac{|response - \mu_p|}{\sigma_p} \quad (1)$$

In this way, a $respDev$ of 0.5 indicates an increase of 0.5 standard deviations from the mean. Boxplots showing the responses for a subset of the adjectives are shown in Figure 2. The collected values for these adjectives align with human intuitions, and we also see something of a floor effect whereby the responses for the adjectives that indicate a smaller change seem to have much lower variance than the responses for the adjectives that indicate a larger change. For example, Figure 2 highlights that there is a small variance in responses for *conservative* and *slight*, but a the large variance in responses for *huge* and *major*.

Fixed Effects	χ^2	p-value
μ_p:σ_p	$\chi^2(1) = 1.98$	$p = 0.16$
μ_p	$\chi^2(1) = 5.59$	$p < 0.05$
σ_p	$\chi^2(97) = 151.46$	$p < 0.001$

Table 1: Results of the likelihood ratio tests (LRTs) used to determine the significance of the model's fixed effects. Significance was determined through a likelihood ratio test comparing a model with the predictor to a model without.

5.2. Data Cleaning

We initially gathered 50 data points for each of our 98 adjectives. However, responses generated using MTurk can be quite noisy, so to reduce the amount of noise in our data we excluded data based on several criteria. We excluded all responses from turkers we considered to be unreliable because more than 50% of their responses were outliers[5], 20% or more of their responses were identical to μ_p, or 50% or more of their responses were identical to one of the given range endpoints. We also removed responses that were less than or equal to the mean, as all prompts asked for an amount of increase. We then removed outliers by adjective. Finally, we removed responses from turkers who had 4 or fewer responses remaining (for the purposes of model fitting and evaluation). This left a total of 3309 responses for our 98 adjectives.

5.3. Model Fitting

Our model fitting was done using the `lme4` package in R (Bates et al., 2015; R Core Team, 2013). The residuals of the model (i.e., $respDev \sim 1 + adj + \sigma_p + \mu_p + \sigma_p : \mu_p + (1|turker)$) showed that the data was heteroskedastic. That is, as the predicted values from the model increased, so did the magnitudes of the error residuals. To adjust for this, we log-transformed $respDev$ to create $logRespDev$.

[2] The interpretation of *most* in terms of standard deviations is ambiguous, and may vary between domains. While we have chosen to build our model under the interpretation of *most* as ± 2 standard deviations, the resulting model could be calibrated to a specific domain by gathering a small set of adjective instances that are accompanied by a specific value. We leave this to future work.

[3] In a pilot study we found that neither the direction of the change (i.e., *increase* vs. *decrease*) nor the non-word used significantly affected the model, so in this study we did not include these as factors.

[4] Note that while random intercepts allow the model to be more robust to variance due to individual biases, the model included in our final resource are averaged across respondents, thus allowing predictions for novel instances.

[5] We considered any points more than 1.5 times the interquartile range below the first quartile or above the third quartile to be outliers.

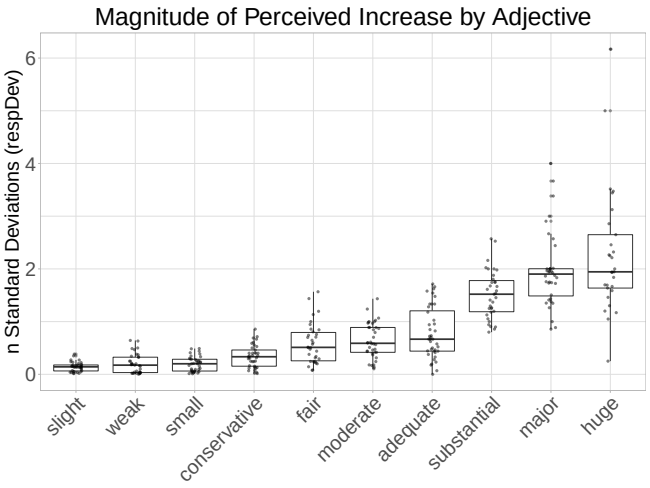


Figure 2: The magnitudes of the perceived increase for several adjectives. The magnitudes are measured as the absolute difference between the survey response and the given mean divided by the given standard deviation.

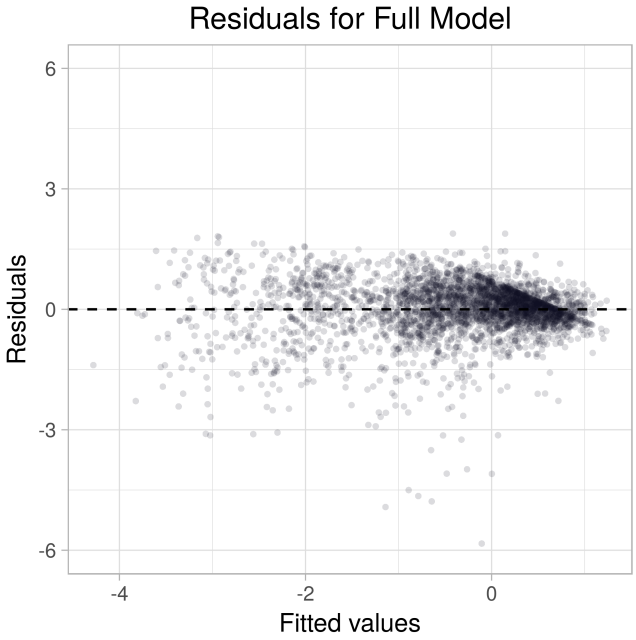


Figure 3: Residual plot for the model with all factors included: $logRespDev \sim 1 + adj + \sigma_p + \mu_p + \sigma_p : \mu_p + (1|turker)$. The residual plot for the final model (with $\sigma_p : \mu_p$ removed) is omitted for space, but it is nearly identical.

After verifying that the resulting residual plot showed homoskedasticity (see Figure 3), we used $logRespDev$ in all subsequent model-building.

We used likelihood ratio tests (LRTs) to determine the significance of the fixed effects (i.e., our three factors and their interactions) by first building a parent model that included all factors as well a daughter models that each had a factor removed. We then checked to see if the model with the factor removed was significantly different from its parent model. The resulting significances are shown in Table 1. As σ_p:μ_p was first determined to not be significant, it was removed and the model without this factor was used as the parent model for testing the significance of σ_p and μ_p, both of which were determined to be significant (see Table 1). The final model is given by:

$$logRespDev \sim 1 + adj + \sigma_p + \mu_p + (1|turker) \quad (2)$$

This fitted model itself is our resource. That is, for each adjective, we have a linear function, $f_{adj}(\mu_p, \sigma_p)$ that describes its predicted impact of the quantity in question. For example, the adjective *small* is represented as:

$$f_{small}(\mu_p, \sigma_p) = -1.77 + (1.034\text{e}{-5})\mu_p - (1.123\text{e}{-3})\sigma_p$$

The predicted new value implied by *small* can then be calculated from this as:

$$new = (e^{f_{small}(\mu_p, \sigma_p)} \times \sigma_p) + \mu_p$$

6. Alternative Models

6.1. Backoff Model

Though the standard deviation was significant, it is not always the case that this will be known. For this situation, we also provide a backoff model that does not include σ_p. The dependent variable used in this model is the absolute percent change in the mean (log-transformed):

$$logPercChange \sim 1 + adj + \mu_p + (1|turker) \quad (3)$$

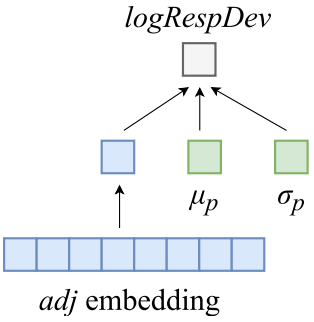


Figure 4: Shallow neural network architecture for predicting the impact of adjectives that were not seen during training using the word embedding of the adjective. As with the linear models, we include the provided mean (μ_p) and provided standard deviation (σ_p) as factors and predict the log-transformed response deviations ($logRespDev$).

6.2. High Frequency (HF) Models

Under the hypothesis that language intuitions will better align for more commonly used words, we additionally retrained our fitted model and backoff models on a smaller subset of the data that consists of the highest frequency words We sorted the adjectives based on their frequency in the English Gigaword corpus (Graff et al., 2003) and retained only the top 30 adjectives. We used this higher frequency subset of adjectives to train a regular as well as a backoff model.

6.3. Grounding Using Neural Networks

Even though the number of adjectives covered by these linear models is fairly large (98), they are unable to predict groundings for novel, or *unseen*, adjectives. To address this limitation, we propose a shallow neural network

	Full	Backoff	HF	HF-Backoff
marginal R^2	0.618	0.548	0.672	0.562
conditional R^2	0.670	0.589	0.725	0.596
predictive R^2	0.632	0.544	0.677	0.542

Table 2: Estimation from all models of how much of the variance in the data is accounted for by the model's fixed effects (marginal R^2), and both the fixed and random effects (conditional R^2). Also, an measure of how well the model predicts new data, predictive R^2.

(NN) model which builds upon pre-trained word embeddings (Mikolov et al., 2013). Intuitively, by using word embeddings trained over a large corpus, we already know some of the underlying semantics of the unseen adjectives. Therefore, to the extent that the embedding of a given adjective captures its implied magnitude, we can learn a mapping from this embedding to the specific, quantitative grounding for the adjective.

In this NN approach, shown in Figure 4, we use a fully-connected hidden layer of size one to compress the adjective's high-dimensional word embedding to a single value (shown in blue), the activation of which can be directly interpreted as the semantic impact of the adjective learned by the model. This value is then concatenated to the provided mean (μ_p) and provided standard deviation (σ_p) and passed to an output layer that predicts the the log transformed response deviation ($logRespDev$). In this framework we found that the features that uniquely identify the individual respondent from the crowdsourcing experiment were not needed (i.e., they did not greatly improve performance), and so we removed them to help prevent overfitting.

We trained the model using mean squared error as the loss function. The embeddings were initialized with the Glove (Pennington et al., 2014) 300-dimensional pre-trained word embeddings and they were not updated during training (again, to reduce overfitting). A tanh activation was used for the adjective node[6] and we use the RMSProp optimizer (Tieleman and Hinton, 2012) with a learning rate of 0.00001 and all other parameters with their default values.

7. Results

7.1. Linear Models

For the evaluation of our linear models, we report the marginal and conditional R^2 in Table 2. The marginal R^2 shows the amount of the variance that is explained by only the fixed effects and the conditional R^2 shows the amount of the variance that is explained by both the fixed and random effects. Both were calculated using the `r.squaredGLMM` function from the MuMIn (Barton, 2016) package, an implementation of the method of Nakagawa and Schielzeth (2013). As we are primarily interested in using this resource to make predictions about new instances of adjectives, the correlation of the model's predictions with real data is key. Thus, we also calculate the predictive R^2 with leave-one-out cross-validation, such that the residual error of each data point (i.e., individual response) is based on a model trained on all the data except for that point. Specifically, the predictive R^2 is the predicted residual error sum of squares (PRESS) statistic (Allen, 1974) divided by the total sum of squares (SS_{total}):

$$R^2_{pred} = 1 - \frac{PRESS}{SS_{total}} \quad (4)$$

$$PRESS = \sum_{i=1}^{n} (logRespDev_i - \widehat{logRespDev}_{i,D\setminus i})^2 \quad (5)$$

$$SS_{total} = \sum_{i=1}^{n} (logRespDev_i - \overline{logRespDev})^2 \quad (6)$$

That is, for each individual response $i \in D$, we sum the residual squared error between the true value, $logRespDev_i$ and the value predicted by a model trained on the rest of the data, $\widehat{logRespDev}_{i,D\setminus i}$. We then divide this by SS_{total} and subtract if from 1 to get the predictive R^2. For our full model, the predictive R^2 was 0.632 (also shown in Table 2). This result suggests that the quantities implied by these adjectives can be predicted with reasonable accuracy with simple, linear models trained on crowdsourced data.

We found that the backoff subset model had a slightly worse fit than the full model. This is expected as it does not contain the standard deviation as a factor, which was determined to be significant (Section 5.3.).

The high-frequency (HF) model shows a higher R^2 than the full model. This confirms that, indeed, language intuitions are more robust for high-frequency adjectives. However, this effect is only seen in the full model when standard deviation is known.

	Seen Adjs	Unseen Adjs
Linear Full Model	0.632	–
NN Model	0.540	0.244

Table 3: Comparison of how well the linear mixed effects full model and the neural network (NN) model predict new data (predictive R^2). Performance is shown for predictions both on adjectives that were present in the training data (seen) as well as on adjectives that were not (unseen).

7.2. Neural Network Model

We evaluate our neural network (NN) model on both seen and unseen adjectives. The predictive R^2 on *seen* adjectives (i.e., when data points for each adjective are split between training and test folds) can be compared to the performance of the linear models, while the performance on *unseen* adjectives (i.e., when adjectives appearing in test folds do *not* occur in training folds) indicates utility with novel adjectives. Due to time constraints rather than using leave-one-out cross-validation (as with the linear models) we instead use four-fold cross-validation with two folds for training, one for development, and one for testing.

The number of epochs for each fold was tuned on this split to avoid overfitting. The other hyperparameters (e.g., learning rate) were tuned on 10% of the data in early experiments and were not revisited. The resulting predictive R^2s are shown in Table 3.

[6] We tried using non-linear activations on all the nodes but did no see an improvement so we omitted them for model simplicity.

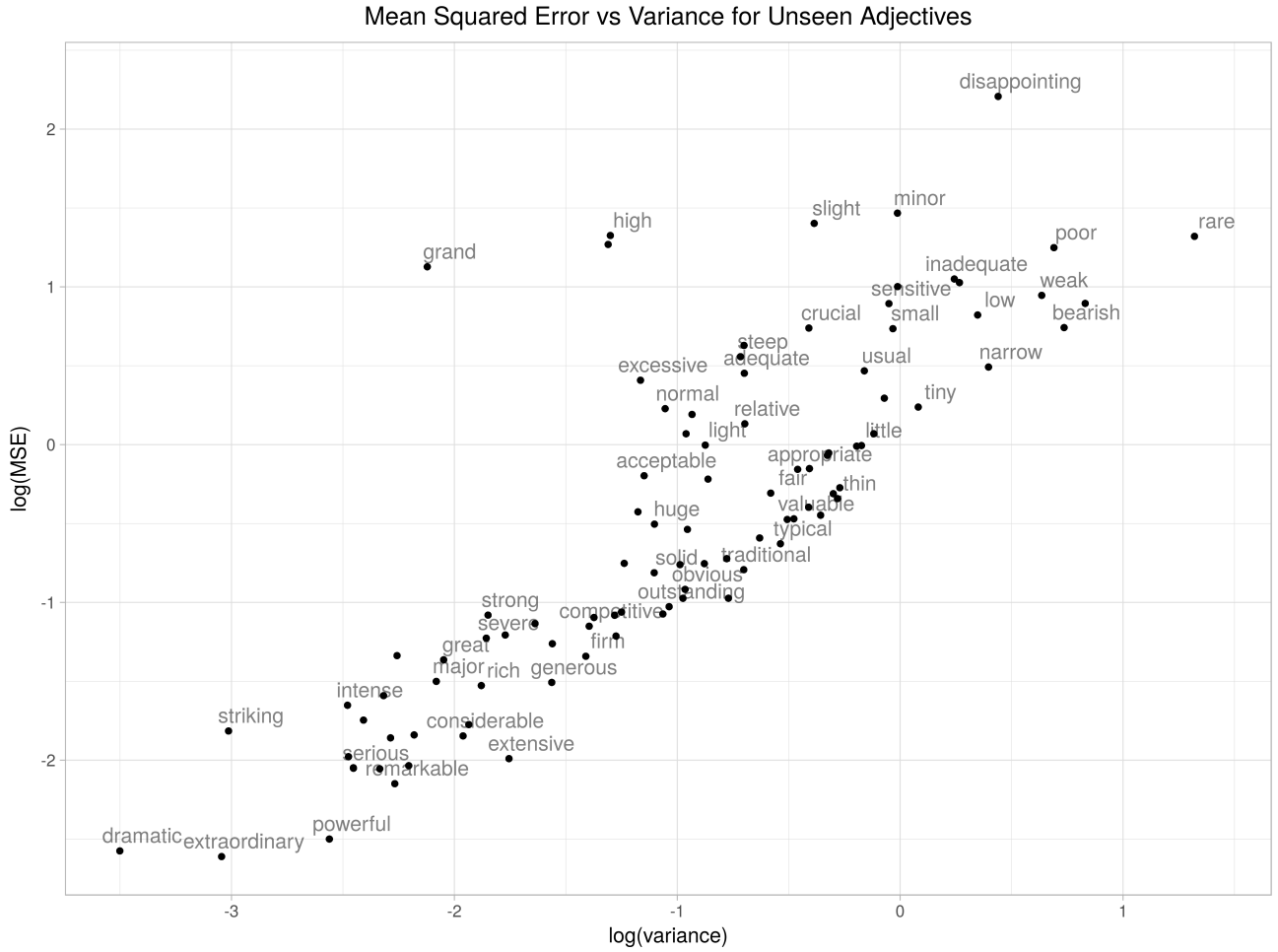


Figure 5: A comparison of the mean squared error (MSE) versus variance for each gradable adjective. The MSE is based on the neural network model predicting on unseen adjectives and the variance is from the original crowdsourced data. Note that the axes are presented in log scale.

On seen adjectives, the NN model performs almost as well as the linear model, and we suspect that the performance difference is primarily due to the larger number of parameters that need to be learned. Additionally, while the information about individual turkers was empirically found to not help the NN model, the linear model benefits from its inclusion.

The predictive R^2 for unseen adjectives is much lower than for seen adjectives, but recall that the linear model is unable to make any predictions for these adjectives at all. It is unclear exactly where the performance drop originates, though we hypothesize that it is primarily due to the reliance on pre-trained word-embeddings. While they allow us to estimate groundings for adjectives that we did not include in training, the estimates are only as good as our capacity to extract the necessary information from the embedding. That is, the embeddings were trained to capture distributional similarity, not relative magnitude of impact. Thus, this information, when present, is indirect and very likely noisy.

To better understand the performance of this model, we compared the mean squared error (MSE) of the model on these unseen adjectives with their variance in the original data from the crowdsourcing experiment. The plot is shown in Figure 5. Overall, as the variance in the original data increases, so does the MSE. This suggests that in general adjectives with higher variance are harder to predict. Further, some adjectives had much higher variance (e.g., *rare* and *disappointing*), suggesting that for some adjectives this task is difficult even for humans. Additionally, certain adjectives (such as *grand, high*, and *disappointing*) had a particularly high MSE. We suspect that this is, again, due to the reliance on the pre-trained emebeddings as many of these words have multiple senses (e.g., *disappointing*, as with *disappointing increase* versus *the play was disappointing*), and here the sense we are interested in is not the most frequent. For these words with multiple senses, the embeddings are confounded. To address these issues, dedicated embeddings that better model the semantics of interest could be explored (such as the embeddings proposed by Kim et al. (2016) that are dedicated for modeling adjectives). We leave this exploration to future work.

8. Conclusion

We proposed a method for quickly and efficiently generating groundings for a set of gradable adjectives. These groundings are modeled using a linear model conditioned on the typical distribution of the item being modified while

remaining otherwise independent of the item's identity. The model was trained on approximately 50 values collected through crowdsourcing for each of the adjectives in the set. The resulting model has a predictive R^2 of 0.632 on the whole dataset (measured through leave-one-out cross-validation), and a R^2 of 0.677 on a subset of high-frequency adjectives. We release all models created for these adjectives, which, we hope, brings us closer to developing technology that answers questions relevant to national and global security from texts containing qualitative statements.

9. Acknowledgements

This work was funded by the Defense Advanced Research Projects Agency (DARPA) World Modeling Seed program under ARO contract W911NF-17-1-0047.

10. Bibliographical References

Allen, D. M. (1974). The relationship between variable selection and data augmentation and a method for prediction. *Technometrics*, 16(1):125–127.

Alxatib, S. and Pelletier, F. J. (2011). The psychology of vagueness: Borderline cases and contradictions. *Mind & Language*, 26(3):287–326.

Bakhshandeh, O. and Allen, J. (2015). From adjective glosses to attribute concepts: Learning different aspects that an adjective can describe. In *IWCS*.

Barton, K. (2016). Mumin: Multi-model inference (r package version 1.15. 6.)[computer software].

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.

Berko, J. (1958). The child's learning of English morphology. *Word*, 14(2-3):150–177.

Bonini, N., Osherson, D., Viale, R., and Williamson, T. (1999). On the psychology of vague predicates. *Mind & language*, 14(4):377–393.

Buhrmester, M., Kwang, T., and Gosling, S. D. (2011). Amazon's mechanical turk. *Perspectives on Psychological Science*, 6(1):3–5.

Bylinina, E. (2014). *The Grammar of Standards. Judge-dependence, Purpose-relativity, and Comparison Classes in Degree Constructions*. Utrecht University.

Elliott, J., Glotter, M., Ruane, A. C., Boote, K. J., Hatfield, J. L., Jones, J. W., Rosenzweig, C., Smith, L. A., and Foster, I. (2017). Characterizing agricultural impacts of recent large-scale us droughts and changing technology and management. *Agricultural Systems*.

Graff, D., Kong, J., Chen, K., and Maeda, K. (2003). English gigaword, ldc2003t05. *Linguistic Data Consortium, Philadelphia*.

Jones, J., Hoogenboom, G., Porter, C., Boote, K., Batchelor, W., Hunt, L., Wilkens, P., Singh, U., Gijsman, A., and Ritchie, J. (2003). The dssat cropping system model. *European Journal of Agronomy*, 18(3):235–265. Modelling Cropping Systems: Science, Software and Applications.

Kim, J. and de Marneffe, M. (2013). Deriving adjectival scales from continuous space word representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1625–1630.

Kim, J.-K., de Marneffe, M.-C., and Fosler-Lussier, E. (2016). Adjusting word embeddings with semantic intensity orders. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 62–69.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Nakagawa, S. and Schielzeth, H. (2013). A general and simple method for obtaining r2 from generalized linear mixed-effects models. *Methods in Ecology and Evolution*, 4(2):133–142.

Narisawa, K., Watanabe, Y., Mizuno, J., Okazaki, N., and Inui, K. (2013). Is a 204 cm man tall or small? acquisition of numerical common sense from the web. In *ACL (1)*, pages 382–391. The Association for Computer Linguistics.

Parry, M. and Rosenzweig, C., (1993). *The Potential Effects of Climate Change on World Food Supply*, pages 1–26. Springer Berlin Heidelberg, Berlin, Heidelberg.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Porwollik, V., Müller, C., Elliott, J., Chryssanthacopoulos, J., Iizumi, T., Ray, D. K., Ruane, A. C., Arneth, A., Balkovič, J., Ciais, P., Deryng, D., Folberth, C., Izaurralde, R. C., Jones, C. D., Khabarov, N., Lawrence, P. J., Liu, W., Pugh, T. A., Reddy, A., Sakurai, G., Schmid, E., Wang, X., de Wit, A., and Wu, X. (2017). Spatial and temporal uncertainty of crop yield aggregations. *European Journal of Agronomy*, 88(Supplement C):10–21. Uncertainty in crop model predictions.

Qing, C. and Franke, M. (2014). Meaning and use of gradable adjectives: Formal modeling meets empirical data. In Paul Bello, et al., editors, *Proceedings of the 36th annual meeting of the Cognitive Science Society (CogSci-2014)*, pages 1204–1209, Austin, TX. Cognitive Science Society.

R Core Team, (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Raffman, D. (1994). Vagueness without paradox. *The Philosophical Review*, 103(1):41–74.

Raffman, D. (1996). Vagueness and context-relativity. *Philosophical Studies*, 81(2-3):175–192.

Rosenzweig, C., Jones, J. W., Hatfield, J. L., Ruane, A. C., Boote, K. J., Thorburn, P., Antle, J. M., Nelson, G. C., Porter, C., Janssen, S., et al. (2013). The agricultural model intercomparison and improvement project (agmip): protocols and pilot studies. *Agricultural and Forest Meteorology*, 170:166–182.

Shapiro, S. (2006). *Vagueness in context*. Oxford Univer-

sity Press on Demand.

Shivade, C., de Marneffe, M., Fosler-Lussier, E., and Lai, A. M. (2016). Identification, characterization, and grounding of gradable terms in clinical text. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing, BioNLP@ACL 2016, Berlin, Germany, August 12, 2016*, pages 17–26.

Sinclair, J. et al. (1987). *Collins COBUILD English language dictionary*. Harper Collins Publishers,.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning.

Whitman, B., Roy, D., and Vercoe, B. (2003). Learning word meanings and descriptive parameter spaces from music. In *Proceedings of the HLT-NAACL 2003 Workshop on Learning Word Meaning from Non-linguistic Data - Volume 6*, HLT-NAACL-LWM '04, pages 92–99, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wilkinson, B. and Tim, O. (2016). A gold standard for scalar adjectives. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Zhang, D. D., Lee, H. F., Wang, C., Li, B., Pei, Q., Zhang, J., and An, Y. (2011). The causality analysis of climate change and large-scale human crisis. *Proceedings of the National Academy of Sciences*, 108(42):17296–17301.

Zhao, C., Liu, B., Piao, S., Wang, X., Lobell, D. B., Huang, Y., Huang, M., Yao, Y., Bassu, S., Ciais, P., Durand, J.-L., Elliott, J., Ewert, F., Janssens, I. A., Li, T., Lin, E., Liu, Q., Martre, P., Müller, C., Peng, S., Peñuelas, J., Ruane, A. C., Wallach, D., Wang, T., Wu, D., Liu, Z., Zhu, Y., Zhu, Z., and Asseng, S. (2017). Temperature increase reduces global yields of major crops in four independent estimates. *Proceedings of the National Academy of Sciences*, 114(35):9326–9331.

Evaluating Phonemic Transcription of Low-Resource Tonal Languages for Language Documentation

Oliver Adams,[1] Trevor Cohn,[1] Graham Neubig,[2] Hilaria Cruz,[3] Steven Bird,[4,5] Alexis Michaud[6]
[1]Computing and Information Systems, The University of Melbourne, Australia
[2]Language Technologies Institute, Carnegie Mellon University, USA
[3]Linguistics Program, Dartmouth College, USA
[4]Northern Institute, Charles Darwin University, Australia
[5]International Computer Science Institute, University of California Berkeley, USA
[6]CNRS-LACITO, National Center for Scientific Research, France
oliver.adams@gmail.com, trevor.cohn@unimelb.edu.au, gneubig@cs.cmu.edu,
steven.bird@cdu.edu.au, hilaria.cruz@dartmouth.edu, alexis.michaud@cnrs.fr

Abstract

Transcribing speech is an important part of language documentation, yet speech recognition technology has not been widely harnessed to aid linguists. We explore the use of a neural network architecture with the connectionist temporal classification loss function for phonemic and tonal transcription in a language documentation setting. In this framework, we explore jointly modelling phonemes and tones versus modelling them separately, and assess the importance of pitch information versus phonemic context for tonal prediction. Experiments on two tonal languages, Yongning Na and Eastern Chatino, show the changes in recognition performance as training data is scaled from 10 minutes up to 50 minutes for Chatino, and up to 224 minutes for Na. We discuss the findings from incorporating this technology into the linguistic workflow for documenting Yongning Na, which show the method's promise in improving efficiency, minimizing typographical errors, and maintaining the transcription's faithfulness to the acoustic signal, while highlighting phonetic and phonemic facts for linguistic consideration.

Keywords: low-resource languages; Asian languages; Mesoamerican languages; speech recognition; language documentation.

1. Introduction

Language documentation involves recording the speech of native speakers. Transcribing these recordings, which are rich cultural and linguistic resources, is an integral part of the language documentation process. However, transcription is slow: it often takes a linguist between 30 minutes to 2 hours to transcribe and translate a minute of speech, depending on the transcriber's familiarity with the language and the difficulty of the content. This is a bottleneck in the documentary linguistics workflow: linguists accumulate considerable amounts of speech, but do not transcribe and translate it all, and there is a risk that untranscribed recordings could end up as "data graveyards" (Himmelmann, 2006, 4,12-13). There is clearly a need for "devising better ways for linguists to do their work" (Thieberger, 2016, 92).

There has been work on low-resource speech recognition (Besacier et al., 2014), with approaches using cross-lingual information for better acoustic modelling (Burget et al., 2010; Vu et al., 2014; Xu et al., 2016; Müller et al., 2017) and language modelling (Xu and Fung, 2013). However, speech recognition technology has largely been ineffective for endangered languages since architectures based on hidden Markov models (HMMs), which generate orthographic transcriptions, require a large pronunciation lexicon and a language model trained on text. These speech recognition systems are usually trained on a variety of speakers and hundreds of hours of data (Hinton et al., 2012, 92), with the goal of generalisation to new speakers. Since large amounts of text are used for language model training, such systems often do not incorporate pitch information for speech recognition of tonal languages (Metze et al., 2013), as they can instead rely on contextual information for tonal disambiguation via the language model (Le and Besacier, 2009; Feng et al., 2012) even though there is no computational burden in additionally using pitch features.

In contrast, language documentation contexts often have just a few speakers for model training and little text for language model training. However, there may be benefit even in a system that overfits to these speakers. If a *phonemic* recognition tool can provide a canvas transcription for manual correction and linguistic analysis, it may be possible to improve the workflow and leverage of linguists. The transcriptions collected in this semi-automated workflow can then be used for refinement of the acoustic model, leading to a snowball effect of better and faster transcription.

In this paper we investigate the application of neural speech recognition models to the task of phonemic and tonal transcription in a language documentation setting where limited resources are available for model training. We use the connectionist temporal classification (CTC) formulation (Graves et al., 2006) for the purposes of direct prediction of phonemes and tones given an acoustic signal, thus bypassing the need for a pronunciation lexicon, language model, and time alignments of phonemes in the training data. By reducing the data requirements we make automatic transcription technology more feasible in a language documentation setting.

We evaluate this approach on two tonal languages, Yongning Na and Eastern Chatino. Na is a Sino-Tibetan language spoken in southwest China with three tonal levels, High (H),

Mid (M) and Low (L), and a total of seven tone labels. Eastern Chatino, spoken in Oaxaca, Mexico, has a richer tone set but both languages have extensive morphotonology (Cruz and Woodbury, 2006; Cruz, 2011; Michaud, 2017). Overall estimates of numbers of speakers for Chatino and Na are similar, standing at about 40,000 for both (Simons and Fennig, 2017), but there is a high degree of dialect differentiation within the languages. The data used in the present study are from the Alawa dialect of Yongning Na, and the San Juan Quiahije (SJQ) dialect of Eastern Chatino; as a rule-of-thumb estimate, it is likely that these materials would be intelligible to a population of less than 10,000.[1]

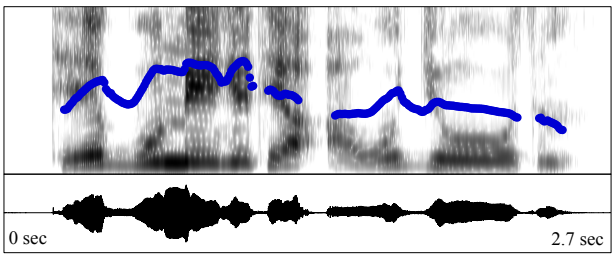
Though a significant amount of Chatino speech has been transcribed (Chatino Language Documentation Project, 2017), its rich tone system make it a useful point of comparison for our explorations of Na, the language for which automatic transcription is our primary practical concern. Though Na has previously had speech recognition applied in a pilot study (Do et al., 2014), phoneme error rates were not quantified and tone recognition was left as future work.

We perform experiments scaling the training data, comparing joint prediction of phonemes and tones with separate prediction, and assessing the influence of pitch information versus phonemic context on phonemic and tonal prediction in the CTC-based framework. Importantly, we qualitatively evaluate use of this automation in the transcription of Na. The effectiveness of the approach has resulted in its incorporation into the linguist's workflow. Our open-source implementation of this phonemic transcription tool, Persephone, is available online.[2]

A preliminary version of this work was presented at the Australasian Language Technology Association Workshop (Adams et al., 2017), which we extend and improve upon in this paper by exploring: the effect of including elicited wordlists in training; how effective the model is at predicting tone group boundaries in Na and how this influences the model's capacity to learn tonal rules; discussion of the potential of this approach for reviewing transcriptions; analysis of the Chatino output; refined results for Na involving data preprocessing improvements and more data; results for both random and story-wise cross validation; and presentation of example utterances.

2. Model

The underlying model used is a long short-term memory (LSTM) recurrent neural network (Hochreiter and Schmidhuber, 1997) in a bidirectional configuration (Schuster and Paliwal, 1997). The network is trained with the connectionist temporal classification (CTC) loss function (Graves et al., 2006). Critically, this alleviates the need for alignments between speech frames and labels in the transcription, which we do not have. This is achieved through the use of a dynamic programming algorithm that efficiently sums over the probability of neural network output label that correspond to the gold transcription sequence when repeated labels are collapsed.



/tʰiˊ | go˧miˊ-dʑoˀ | tʰiˊ, | ɑ˩ʁo˧ dʑo˩ tsɯ˩ | mɣ̩˩. |/
Quant à la sœur, elle demeurait à la maison, dit-on.
As for the sister, she stayed at home.
而妹妹呢，留在家里。

Target label sequence:

1.	tʰ i g o m i dʑ o tʰ i ɑ ʁ o dʑ o t s ɯ m ɣ
2.	ˊ ˧ ˧ ˀ ˊ ˩ ˧ ˩ ˩ ˩
3.	tʰ i ˊ g o ˧ m i ˧ dʑ o ˀ tʰ i ˊ ɑ ˩ ʁ o ˧ dʑ o ˩ t s ɯ ˩ m ɣ ˩
4.	tʰ i ˊ | g o ˧ m i ˧ dʑ o ˀ | tʰ i ˊ | ɑ ˩ ʁ o ˧ dʑ o ˩ t s ɯ ˩ | m ɣ ˩

Figure 1: A sentence from the Na corpus. Top to bottom: spectrogram with F_0 in blue; waveform; phonemic transcription; French, English and Chinese translations; target label sequences: (1) phonemes only, (2) tones only, (3) phonemes and tones together, and (4) phonemes and tones with tone group boundary markers, "|".

The use of an underlying recurrent neural network allows the model to implicitly model context via the parameters of the LSTM, despite the independent frame-wise label predictions of the CTC network. It is this feature of the architecture that makes it a promising tool for tonal prediction, since tonal information is suprasegmental, spanning many frames (Mortensen et al., 2016). Context beyond the immediate local signal is indispensable for tonal prediction, and long-ranging context is especially important in the case of morphotonologically rich languages such as Na and Chatino.

Past work distinguishes between *embedded* tonal modelling, where phoneme and tone labels are jointly predicted, and *explicit* tonal modelling, where they are predicted separately (Lee et al., 2002). We compare several training objectives for the purposes of phoneme and tone prediction. This includes separate prediction of 1. phonemes and 2. tones, as well as 3. jointly predict phonemes and tones using one label set. We additionally explore 4. joint prediction of phonemes, tones and the tone group boundaries (TGBs) which delimit tone groups. Figure 1 presents an example sentence from the Na corpus described in §3.1., along with an example of these four objectives.

3. Experimental Setup

We designed the experiments to answer these primary questions:

1. How do the error rates scale with respect to training data?
2. How effective is tonal modelling in a CTC framework?
3. To what extent does phoneme context play a role in tone prediction?

[1] For details on the situation for Eastern Chatino, see Cruz (2011, 18-23).

[2] `https://github.com/oadams/persephone`

4. Does joint prediction of phonemes and tones help minimize error rates?

We assess the performance of the systems as training data scales from 10 minutes to 224 minutes of spontaneous speech of a single Na speaker, and between 12 and 50 minutes for a single speaker of Chatino. Experimenting with this extremely limited training data gives us a sense of how much a linguist needs to transcribe before this technology can be profitably incorporated into their workflow.

We evaluate both the phoneme error rate (PER) and tone error rate (TER) of models based on the same neural architecture, but with varying input features and output objectives. Input features include log Filterbank features[3] (`fbank`), pitch features of Ghahremani et al. (2014) (`pitch`), and a combination of both (`fbank+pitch`). These input features vary in the amount of acoustic information relevant to tonal modelling that they include. The output objectives correspond to those discussed in §2.: tones only (`tone`), phonemes only (`phoneme`), or jointly modelling both (`joint`). We denote combinations of input features and target labellings as ⟨`input`⟩⇒⟨`output`⟩.

In case of tonal prediction we explore similar configurations to that of phoneme prediction, but with two additional points of comparison. The first is predicting tones given one-hot phoneme vectors (`phoneme`) of the gold phoneme transcription (`phoneme⇒tone`). The second predicts tones directly from pitch features (`pitch⇒tone`). These points of comparison serve to give us some understanding as to how much tonal information is being extracted directly from the acoustic signal versus the phoneme context.

In the `fbank+pitch⇒joint` configuration, we additionally explore the difference in performance between models that jointly predict tone group boundaries as well as phonemes and tones.

3.1. Data

We explore application of the model to the Na corpus (Michaud and Latami, 2017b) that is part of the Pangloss collection (Michailovsky et al., 2014). This corpus consists of around 100 narratives, constituting 11 hours of speech from one speaker in the form of traditional stories, and spontaneous narratives about life, family and customs (Michaud, 2017, 33). Several hours of recordings of spontaneous narratives have been phonemically transcribed, and we used up to 224 minutes of this for training, 24 minutes for validation and 23 minutes for testing. This represents an increase in training data from that used in preliminary reports on this work (Adams et al., 2017). Included in this additional data is 6 minutes and 30 seconds of semi-automatically transcribed speech of the narrative *Housebuilding 2* (Michaud and Latami, 2017a), where an automatic transcription of a model trained on less data was used as a canvas by the linguist during a field trip in 2017.

The total number of phoneme and tone labels used for automatic transcription was 90 and 7 respectively. This represents an increase in the number of phonemes from previously reported results, where 78 were used (Adams et al., 2017). This increase is the result of improved preprocessing of the linguist's transcriptions, taking place in parallel with improvements to the original transcriptions to arrive at full consistency. Concerning the preprocessing, (1) we represent fillers /əəə…/ and /mmm…/ with their own tokens; (2) onsetless syllables which are digraphs /wæ/, /wɑ/, /wɤ/, /jæ/, /jɤ/, /jo/ are also now represented as a single token because they constitute phonological units (syllable rhymes; Na syllables are composed of an onset, a rhyme, and a tone). Concerning improvements to the original transcriptions, we addressed cases where the same phoneme had inconsistent representation in the corpus, such as /wæ̃/ and /w̃æ/, as well as an instance where the unicode representation of a single phoneme was sometimes `v+nasality+syllabic diacritic` and sometimes `v+syllabic diacritic+nasality`. We computed the Na results of Tables 1-3 using the larger suite of 224 minutes and these preprocessing changes.

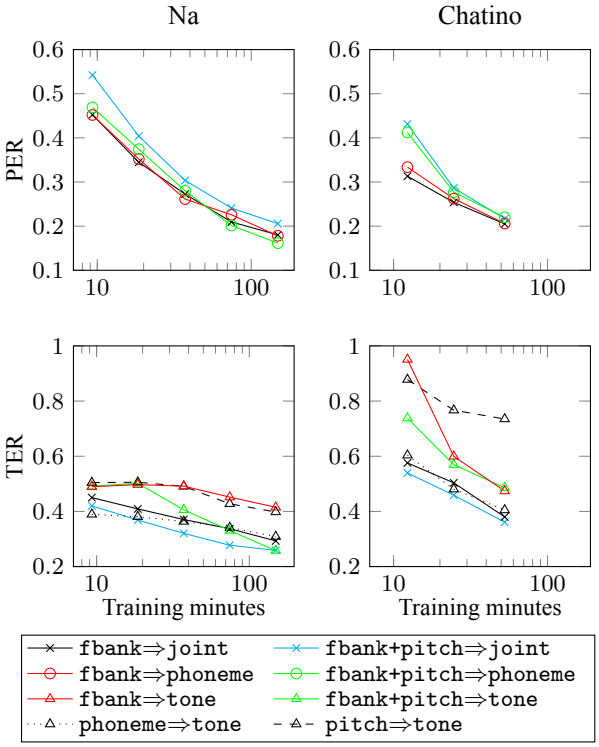


Figure 2: Phoneme error rate (PER) and tone error rate (TER) on test sets as training data is scaled from around 10 minutes up to 150 minutes for Na (left) and up to 50 minutes for Chatino (right). The legend entries are formatted as ⟨input⟩ ⇒ ⟨output⟩ to indicate model input and output.

For Chatino, we used data of Ćavar et al. (2016) from the GORILLA language archive for Eastern Chatino of San Juan Quiahije, Oaxaca, Mexico (Cavar et al., 2016) for the purposes of comparing phoneme and tone prediction with Na when data restriction is in place. We used up to 50 minutes of data for training, 6 minutes for validation and 6 minutes for testing. The phoneme inventory we used consists of 31 labels along with 14 tone labels. For both languages, preprocessing involved removing punctuation and any other symbols that are not phonemes, tones or the tone

[3] 41 log Filterbank features along with their first and second derivatives

		Chatino		Na		
Input	Output	PER ↓	TER ↓	PER ↓	TER ↓	TGB-F1 ↑
fbank	joint	0.205	0.380	0.119	0.219	0.845
fbank+pitch	joint	0.217	0.361	0.128	0.177	0.856
fbank	phoneme	0.206	-	0.112	-	-
fbank+pitch	phoneme	0.220	-	0.129	-	-
fbank	tone	-	0.474	-	0.394	0.830
pitch	tone	-	0.735	-	0.513	0.789
fbank+pitch	tone	-	0.486	-	0.240	0.847
phoneme	tone	-	0.405	-	0.267	0.850

Table 1: The phoneme error rate (PER) and tone error rate (TER) of a variety of models for transcription of Chatino and Na, along with tone group boundary F1 scores (TGB-F1) for Na. The Chatino models were trained on a total 50 minutes of training data, while the Na models were trained on 224 minutes.

group boundary (TGB) symbol, "|", such as hyphens connecting syllables within words.

3.2. Training and Tuning

We trained each configuration for a minimum of 30 epochs, stopping if no improvements on the validation set were found for 10 consecutive epochs thereafter. Occasionally ($< 10\%$ of the time) model training failed to converge, in which case training was restarted. Batch size varied with the training set, between 4 and 64 utterances as the total number of training utterances scaled from 128 to 2,048.

We tuned the hyperparameters on the Na validation set, settling on 3 hidden layers with 250 hidden units as a consistently solid performer across `fbank⇒phoneme`, `fbank+pitch⇒joint` and `phoneme⇒tone` for varying amounts of training data up to 150 minutes. For the full Na data set of 224 minutes, 400 hidden layers performed better and was used for the results in Tables 1-3. We averaged results across 3 training runs.

4. Quantitative Results

Figure 2 shows the phoneme and tone error rates for Na and Chatino as training is scaled from around 10 minutes.

Note that the results for Na reported in Tables 1-3 are substantially better than the best results reported in Figure 2, on account of preprocessing changes, increased data and use of hyperparameters that are more effective for the larger amount of data. The settings used for Figure 2 were tailored to a smaller data set, but could be further improved. The Chatino data and its preprocessing remain unchanged.

Error rate scaling Error rates decrease logarithmically with training data. The best methods have a lower than 30% PER with 30 minutes of training data. We believe it is reasonable to expect similar trends in other languages with similar data quality and a single speaker. These results thus suggest how much audio linguists might need to transcribe before semi-automation can become part of their workflow.

Tonal modelling TER is always higher than PER for the same amount of training data, despite there being only 7 tone labels versus 90 phoneme labels in our Na experiment. This is true even when pitch features are present. However, it is unsurprising since the tones have overlapping pitch ranges, and can be realized with vastly different pitch over the course of a single sentence. This suggests that context is more important for predicting tones than phonemes, which are more context-independent.

`fbank⇒tone` and `pitch⇒tone` are vastly inferior to other methods, all of which are privy to phonemic information via training labels or input. However, combining the fbank and pitch input features (`fbank+pitch⇒tone`) makes for a competitive approach for tonal prediction in Na at maximum training data. This indicates both that these features are complementary and that the model has learnt a representation useful for tonal prediction that is on par with explicit phonemic information.

Though tonal prediction is more challenging than phoneme prediction, these results suggest automatic tone transcription is feasible using this architecture, even without inclusion of explicit linguistic information such as constraints on valid tone sequences, which is a promising line of work.

In the case of phoneme-only prediction, the use of pitch information doesn't help reduce the PER, which differs from previous work (Metze et al., 2013), including our own preliminary results (Adams et al., 2017).

Phoneme context To assess the importance of context in tone prediction, `phoneme⇒tone` gives us a point of comparison where no acoustic information is available at all. It performs reasonably well for Na, and competitively for Chatino. One likely reason for its solid performance is that long-range context is modelled more effectively with phoneme input features, as there are vastly fewer phonemes per sentence than speech frames. The rich morphotonology of Na and Chatino means context is important in the realisation of tones, explaining why `phoneme⇒tone` can perform almost as well as methods using acoustic features.

Joint prediction Interestingly, joint prediction of phonemes and tones does not consistently outperform the best methods for phoneme-only prediction. In light of the celebrated successes of multitask learning in various domains (Collobert et al., 2011; Deng et al., 2013; Girshick, 2015; Ramsundar et al., 2015; Ruder, 2017), one might expect training with joint prediction of phonemes and tones to help, since it gives more relevant contextual information

to the model. The TER, on the other hand, is always at its lowest when tones are jointly predicted with phonemes.

Na versus Chatino The trends observed in the experimentation on Chatino were largely consistent with those of Na, but with higher error rates owing to less training data and a larger tone label set. There are two differences with the Na results worth noting. One is that `phoneme⇒tone` is somewhat more competitive in the case of Chatino, suggesting that phoneme context plays a more important role in tonal prediction in Chatino. The second is that `fbank⇒tone` outperforms `pitch⇒tone`, and that adding pitch features to Filterbank features offers less benefit than in Na. This may be because the phonological interpretation of the pitch features is less straightforward in Chatino than it is in Na.

Tone group boundary markers An important concept in the morphotonology of Na is the notion of a *tone group*. Tone groups (phonological phrases) describe segments of speech in which the realization of tones may have interdependent relationships. Since the rules by which tones influence neighbouring tones in a tone group are well described (Michaud, 2017), there is potential to harness these rules to improve tone transcription in a rule-based fashion by enforcing transcriptions to comport with these rules. While the specifics of the rules are language dependent, there is potential to enable linguists to describe these rules to the model in order to improve transcription in the language documentation setting. However, this relies on identifying tone groups since the tonal rules do not hold across the dividing lines between tone groups.

Tone group boundaries (TGBs) are the points that demarcate tone groups. While TGBs are a somewhat more abstract concept than that of phones or tones, there are acoustic features that may be harnessed to determine these tone group boundaries, including rhythm, duration, F_0, and details in the articulation of vowels and consonants. In light of the performance achieved when predicting tones from phonemes without acoustic information (`phoneme⇒tone`), and the potential value in harnessing tone group information for improved tonal prediction or rule-based methods, we additionally evaluated the performance of TGB prediction.

The first two rows of Table 2 show the performance of models trained without and with TGB prediction, respectively. TGB prediction is surprisingly accurate as per the F1 score. Prediction of TGBs decreases phoneme error rates somewhat, and decreases tone error rate moreso. TGBs have influence on the tones that precede and follow them, so this is unsurprising. While there is in principle no inherent link between tone group boundaries and phonemes (any syllable can occur before a tone-group boundary), some morphemes, such as the reported speech particle /tsɯ/, are frequent at the end of sentences, and the end of sentences also means the end of a tone group. Topic markers (the most frequent morphemes in the language) are also often found at TGBs and that can create a bias. This can potentially account for the decrease in PER when the model is trained to predict TGBs.

Elicited speech It is common practice in field linguistics to elicit clear non-spontaneous speech of interesting grammatical constructs to complement the collection of natural spontaneous speech. Such elicitation is useful for linguistic analysis, since some forms are unlikely to be found in a small corpus of spontaneous speech, and thus one cannot arrive at complete morphological paradigms. On the other hand, elicited speech tends to have bias and lacks many properties of spontaneous speech and so a balance of spontaneous and elicited speech is considered important (Cruz and Woodbury, 2014; Niebuhr and Michaud, 2015).

TGB	Wordlist	PER ↓	TER ↓	TGB-F1 ↑
No	No	0.131	0.184	-
Yes	No	0.128	0.177	0.856
No	Yes	0.129	0.178	-
Yes	Yes	0.135	0.179	0.858

Table 2: Results for Na with 224 minutes of spontaneous narrative speech and, where applicable, and 105 minutes of elicited wordlist speech. TGB-F1 is the F1 score of tone group boundary prediction.

Cross-validation	PER ↓	TER ↓	TGB-F1 ↑
Story-wise	0.163	0.205	0.842
Random	0.150	0.189	0.855
Default test set	0.128	0.177	0.856

Table 3: *Story-wise* cross-validation results for Na. Since a linguist will apply their model to utterances in narratives unseen in the training data, the first row is perhaps most representative of the results that can be expected in practice.

Supplementing the 224 minutes of Na training data is 104 minutes of elicited speech in the form of wordlists. For linguists interested in incorporating automation into their transcription workflow, one question is: what is the relative value of elicited speech versus spontaneous speech for improving the system? For insight into this we additionally include the elicited speech in the Na training set, constituting a 46% increase in the total duration of training data.

Table 2 shows the performance change when wordlists are additionally included in the training data. Using wordlists and not predicting TGBs yields a comparable improvement to adding prediction of TGBs without wordlists. However, when both TGBs are predicted and wordlists are used, the PER goes up, even though the TGB-F1 does not suffer. TGBs are easy to predict in the wordlists because there tend to be fewer of them and they tend to delimit repeats of the same word. As a result the phonemes TGBs co-occur with in that context is biased differently to those that they co-occur with in the spontaneous narratives, even though the TGB serves the same function. The nature of the interdependence between TGBs and phonemes is thus different in the narratives versus the wordlists. This observation illustrates how speech processing tools can help characterize how the function of a given sign differs across linguistic data sets, opening up new possibilities for linguistic-semiotic studies of speech corpora.

Our results are consistent with conventional machine learning understanding that training data should be similar to test

data. Thus, slower hyper-articulated training data may not necessarily help a model transcribe faster hypo-articulated speech that the linguist ultimately wants to transcribe.

***Story-wise* cross-validation** The quantitative results reported up to this point are based on training, validation and test sets randomly selected at the utterance level. However, this means the training set gets a fair representation of utterances from all the narratives present in the test set. Since the vocabulary of different narratives is different, it may be more insightful to evaluate the performance on a held-out narrative. We performed cross-validation where each fold involves testing on one of 27 held-out narratives (stories). Results are presented in Table 3, using the `fbank+pitch⇒joint` configuration with TGB prediction and without wordlists. Performance is substantially worse than the performance on the test set, with large variation between the stories for a minimum and maximum PER of 0.125 and 0.249, respectively and a minimum and maximum TER of 0.157 and 0.241. To ensure that this difference between story-wise cross-validation and the default test set reflects the challenge of encountering new story-specific content and isn't simply an artefact of the test set, we performed random cross-validation with held-out sets of the size of the average narrative. Performance for this task was worse than on the test set, but substantially better than story-wise cross-validation. The story-wise cross validation is thus most representative of the error rates to be expected in the transcription of subsequent Na data.

Implicit learning of tone rules In Na, a set of phonological tone rules govern how the underlying tones of words are converted to surface tones when realised in a tone group. To gain some insight into how prediction of TGBs can inform how tones should be predicted, we consider instances in the transcription where tone group boundaries influence the realisation of tones, and compare performance of the model that predicts TGBs (denoted as `TGB`) versus the one that doesn't (denoted as `¬TGB`).

Tone rule 6 (Michaud, 2017, 323) states *"In tone-group-final position, H and M are neutralized to H if they follow a L tone."* The Mid and High tones are acoustically identical in the tone group final position, and so the transcription is normalized to H. For example, /|dʑɤ˩ts^{h}o˦|/ becomes /|dʑɤ˩ts^{h}o˥|/.

In the test set, in 26.4% of the instances where this rule applies, it had not been applied by the human transcriber. Michaud hesitated on whether to transcribe according to the surface phonology or the tonal string prior to the application of this phonological rule. A decision was only made in 2015 and not all the narratives have yet been normalized. This means the training set used a mix of both tones and generally biased towards the high tone. "The transcription of spontaneous speech in little-known languages (...) is built on a more or less shaky foundation of linguistic hypotheses (...). It is certainly not raw or unchanging data." (Jacobson et al., 2001, 81).

The percentage of tones that were M in such positions was 27.7% and 23.0% for `TGB` and `¬TGB` respectively, which is close to the ratio in the reference transcriptions.

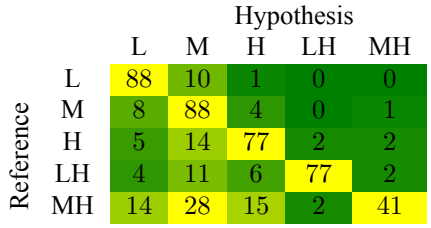


Figure 3: Confusion matrix showing the rates of substitution between Na tones (as a percentage, normalized per reference tone).

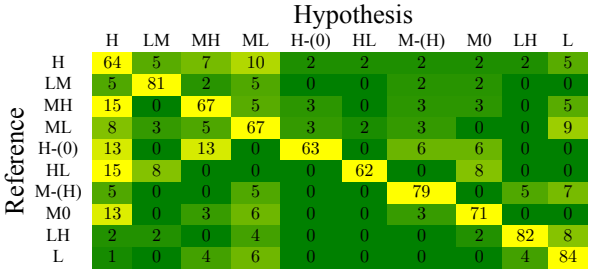


Figure 4: Confusion matrix showing the rates of substitution between Chatino tones (as a percentage, normalized per reference tone).

`¬TGB` transcribed with either an M or H tone 63.7% of the time in instances where the rule applies. However, `TGB` predicted M or H 80.4% of the time. Considering only instances when the latter model predicted a following TGB (TGB recall was 87.5%), this probability increased to 83.1%. This does suggest that predicting TGBs helps the model to learn that these tones should be transcribed as M or H in this situation.[4]

For tone rule 3 ("*In tone-group-initial position, H and M are neutralized to M*") the human annotator was 100% accurate. For instances where rule 3 applies, an M tone was predicted 88.6% by `¬TGB`, and 89.4% by `TGB`. Considering instances where the latter model accurately predicts the TGB (TGB recall was 83.2%), the accuracy increases to 93.1%.

There is thus some evidence to suggest that these tonal rules are learnt implicitly and benefit from TGB prediction.

Na tone errors Figures 3 and 4 show the most common tone substitution mistakes for Na and Chatino respectively using the `fbank+pitch⇒joint` configuration. The relative rates of substitution were similar for other methods. For Na, the most mis-recognized tone was the MH contour, which was frequently misclassified as M, H and L. These three tones are far more common, giving a bias to the training data. Moreover, in running speech the M and H tones have pitch ranges and phonetic contours that overlap substantially with the MH tone (due to tonal coarticulation as well as intonation: the conveyance of prominence and phrasing).

Chatino tone errors For Chatino, the most common errors were mislabelling tones as tone 1 (H) instead of 32 (MH), 0 ("super high"), 14 (HL-(0)) and 20 (M0). These tones generally have a similar pitch to tone 1. The speaker's

[4]These percentages were based on the test hypotheses across 4 different trained models.

Reference	tʰ i ꜔ k i ꜔ s e ꜓ | tʰ i ꜔ dz ɯ ꜓ s e ꜓ |
TGB	tʰ i ꜔ k i ꜓ s e ꜓ | tʰ i ꜔ dz i ꜓ s e ꜓|
¬TGB	tʰ i ꜔ k i ꜓ s e ꜓ tʰ i ꜗ dz i ꜓ s e ꜓

Table 4: An example Na transcription (from the narrative *Sister V1*, utterance #30), and the automatic transcriptions of two models from the joint phoneme and tone prediction task. TGB is a model that additionally predicts tone group boundary markers, while ¬TGB does not. The reference transcription has punctuation and syllable boundaries removed.

Reference	n d e 2 j y an 1 w a 42 n e 2
Hypothesis	n d e 2 j y o 14 w a 42 r e 2 n e 2
Revised	n d e 2 j y an 1 w a 42 r e 2 ne 2

Table 5: An example transcription of a Chatino sentence. There were common errors made in the model's hypothesis, such as confusing low and high back nasal vowels. The automatic transcription also highlighted errors in the reference transcription, leading to its revision.

tone 1 goes from 310-315HZ. Tone 32 runs roughly 270-290Hz, so it is possible the model is catching the top range of the contour. Similarly, tone 14 and 1 begin at the same level, with 14 going from 300-170Hz. This suggests that the model isn't considering the whole contour (see §5.2.). In isolation, it is difficult to distinguish the 0 and 1 tone, and tone 20 also starts roughly at 310Hz before going up to 370Hz and back down to 340Hz. In contrast, there was limited confusion between tones with substantially different pitch, even frequently occurring ones such as tone 4 (L).

5. Qualitative Discussion

5.1. Na

The error rates in the above quantitative analysis are promising, but is this system of practical use in a linguistic workflow? We discuss here the experience of a linguist (Michaud) in applying this model to Na data to aid in transcription of 9 minutes and 30 seconds of speech.

The phonemic errors typically make linguistic sense: they are not random added noise and often bring the linguist's attention to phonetic facts that are easily overlooked because they are not phonemically contrastive.

One set of such errors is due to differences in articulation between different morphosyntactic classes. For example, the noun 'person' /hĩ˥/ and the relativizer suffix /-hĩ˥/ are segmentally identical, but the latter is articulated much more weakly than the former and it is often recognized as /ĩ/ in automatic transcription, without an initial /h/. Likewise, in the demonstrative /ʈʂʰɯ˥/ the initial consonant /ʈʂʰ/ is often strongly hypo-articulated, resulting in its recognition as a fricative /ʂ/, /ʐ̩/, or /ʑ/ instead of an aspirated affricate. The extent to which a word's morphosyntactic category influences its pronunciation is known to be language-specific (Brunelle et al., 2015); the phonemic transcription tool indirectly reveals that this influence is considerable in Na.

A second set is due to loanwords containing combinations of phonemes that are unattested in the training set. For example /ʑɯ˩pe꜔/, from Mandarin *rìběn* (日本 , 'Japan'). /pe/ is otherwise unattested in Na, which only has /pi/; accordingly, the syllable was identified as /pi/. In documenting Na, Mandarin loanwords were initially transcribed with Chinese characters, and thus cast aside from analyses, instead of confronting the issue of how different phonological systems coexist and interact in language use.

A third set of errors made by the system result in an output that is not phonologically well formed, such as syllables without tones and sequences with consonant clusters such as /kgɣ̩/. These cases are easy for the linguist to identify and amend.

The recognition system currently makes tonal mistakes that are easy to correct on the basis of elementary phonological knowledge: it produces some impossible tone sequences such as M+L+M inside the same tone group. Very long-ranging tonal dependencies are not harnessed so well by the current tone identification tool. This is consistent with quantitative indications in §4. and is a case for including a tonal language model or refining the neural architecture to better harness long-range contextual information.

Table 4 exemplifies common errors and successes of the models. Erroneous replacement of the mid tone (/꜔/) with the low tone (/꜓/) was one of the most common mistakes for all models. In the second tone group, the absence of a tone group boundary following the /tʰ i ꜔/ precludes the use of the mid-high tone (/ꜗ/), even though phonetically there is a rise there. The model with tone group boundary prediction (TGB) halved the number of mis-transcriptions of /꜔/ as /ꜗ/, suggesting it used information about tone groups to learn a more phonological representation. In all models, misclassification of /ɯ / as /i/ was one of the most common errors.

5.2. Chatino

For tonal prediction in Chatino, the model has issues distinguishing between ascending and descending tones that have overlapping pitch ranges. There is additional trouble with distinguishing contours and floating tones. It was noted by the linguist (Hilaria Cruz) that in many of these cases it appears as though the model likes to pick up just one point in the tonal range. This is not inconsistent with typical behaviour of CTC-based networks, where label probabilities tend to spike in narrow windows of time (Graves et al., 2006). The model may be getting overconfident in the prediction of tone in a narrow part of the contour, but a more thorough investigation into the timing and cause of label probability peaks is required to be conclusive.

As for phonemes, the system had issues recognizing laminal sounds (eg. *ndyke32wan4* recognized as *ne32wan4*), prenasalized stops and glottal stops (eg. *ntyqya24qa1* recognized as *ya140qa1*). All of these are phonemically contrastive and are key sounds in SJQ Chatino.

There are also frequent issues with back nasal vowels. In the example in Table 5, there is a confusion between a low back and high back nasal vowel. There also tend to be is-

sues with double mora (eg. *ja4jlyo20ren2enq1* recognized as *ja4jlyo20ren1*).

6. Benefits for the Linguist

Using this automatic transcription as a starting point for manual correction was found to confer several benefits to the linguists.

Faithfulness to acoustic signal The model produces output that is faithful to the acoustic signal. In casual oral speech there are repetitions and hesitations that are sometimes overlooked by the transcribing linguist. When using an automatically generated transcription as a canvas, there can be full confidence in the *linearity* of transcription, and more attention can be placed on linguistically meaningful dialogue with the language consultant. There are also perceived benefits to this faithfulness even in the case of SJQ Chatino, where the linguist is a native speaker (Cruz).

Typographical errors and the transcriber's mindset Typographic errors are common, with a large number of phoneme labels and significant use of combinations of keys (Shift, Alternative Graph, etc). By providing a high-accuracy first-pass automatic transcription, much of this manual data entry is avoided. Enlisting the linguist solely for correction of errors also allows them to embrace a critical mindset, putting them in "proofreading mode," where focus can be entirely centred on assessing the correctness of the system output without the additional distracting burden of data entry.

In the Na documentation workflow, the importance of this effect is amplified, since the linguist is not a native speaker: transcriptions are made during fieldwork with a language consultant and are difficult to correct later on based only on auditory impression when the consultant is not available.

Although native speaker linguists have the great advantage of being able to carry out transcription independent of consultants (as in the case of Hilaria Cruz for SJQ Chatino), native language orthographies are for the most part very young and for this reason, there are few people who are trained to perform these tasks. The transcription is thus overwhelmingly handled by few overworked native linguists, which has led to repetitive stress injuries from excessive typing.

Speed Assessing automatic transcription's influence on the speed of the overall language documentation process will require time. Language documentation is a holistic process. Beyond phonemic transcription, documentation of Na involves other work that happens in parallel: translating, copying out new words into the Na dictionary, and being constantly on the lookout for new and unexpected linguistic phenomena.

In the case of Na, this all takes place in the context of discussions with a native speaker linguist. Further complicating this, Michaud's proficiency of the language and speed of transcription is dynamic, improving over time. This makes comparisons difficult.

From this preliminary experiment, the efficiency of the transcription in the Na workflow was perceived to be improved, but the benefits lie primarily in the advantages of providing a transcript faithful to the recording, and allowing the linguist to minimize manual entry, focusing on correction and enrichment of the transcribed document.

The snowball effect More data collection means more training data for better automatic transcription performance. The process of improving the acoustic model by training on such semi-automatic transcriptions has begun, with the freshly transcribed *Housebuilding2* used in this investigation having already been incorporated into subsequent Na acoustic modelling training. In the current set-up, this has involved sending new transcriptions between the linguist and computer-science for re-training, though it's conceivable this process could be automated at the linguist's end.

Reviewing transcriptions A goal of Michaud in the Na documentation process is carefully groomed transcriptions. As stated earlier, conventions for transcribing a newly documented language are not static but change.

The process of using cross-validation to review transcriptions for Na is now in its early stages. In this process, some errors in transcription have been noted that arose from the workflow: A form of respeaking that took place in the documentation has had some minor influence on the transcription. Sometimes the consultant would be requested to respeak a few seconds of speech for the greatest clarity, or Michaud would respeak. In both cases, substitutions of one acceptable variant for another can happen, such as replacing "this" for "that" when they are semantically equivalent. One instance is in *Buried Alive 2*, sentence #123 (Michaud and Latami, 2017b), where Persephone predicted /tʰɣ̣˧/ in the 5th tone group, while the manual transcription has /ʈʂʰɯ˧/ on the basis of a subtly distinct respeaking.

In 33 of 207 transcriptions in the Chatino validation set, comparison of the model hypothesis with the reference transcription helped the linguist to spot errors in the reference transcription (eg. Table 5). Since the Na narratives and Chatino read speech are substantially different, this suggests cross-language generality in the potential for automation to help highlight potential inconsistencies in the manual transcription, as well as aiding in the transcription of untranscribed resources.

7. Conclusion

We have presented the results of applying a CTC-based LSTM model to the task of phoneme and tone transcription in a resource-scarce context: that of a newly documented language. Beyond comparing the effects of a various training inputs and objectives on the phoneme and tone error rates, we reported on the application of this method to linguistic documentation of Yongning Na. Its applicability as a first-pass transcription is very encouraging, and it has now been incorporated into the workflow for transcribing hitherto untranscribed speech as well as reviewing existing transcriptions. Our results give an idea of the amount of speech other linguists might aspire to transcribe in order to bootstrap this process: as little as 30 minutes in order to obtain a sub-30% phoneme error rate as a starting point, with further improvements to come as more data is transcribed in the semi-automated workflow.

8. Acknowledgements

We are very grateful for support from NSF Award 1464553 *Language Induction Meets Language Documentation*.

9. Bibliographical References

Adams, O., Cohn, T., Neubig, G., and Michaud, A. (2017). Phonemic transcription of low-resource tonal languages. In *Australasian Language Technology Association Workshop 2017*, pages 53–60.

Besacier, L., Barnard, E., Karpov, A., and Schultz, T. (2014). Automatic speech recognition for under-resourced languages: A survey. *Speech Communication*, 56:85–100.

Brunelle, M., Chow, D., and Nguyễn, T. N. U. (2015). Effects of lexical frequency and lexical category on the duration of Vietnamese syllables. In The Scottish Consortium for ICPhS 2015, editor, *Proceedings of 18th International Congress of Phonetic Sciences*, pages 1–5, Glasgow. University of Glasgow.

Burget, L., Schwarz, P., Agarwal, M., Akyazi, P., Feng, K., Ghoshal, A., Glembek, O., Goel, N., Karafiát, M., Povey, D., and Others. (2010). Multilingual acoustic modeling for speech recognition based on subspace Gaussian mixture models. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 4334–4337. IEEE.

Ćavar, M. E., Cavar, D., and Cruz, H. (2016). Endangered Language Documentation: Bootstrapping a Chatino Speech Corpus, Forced Aligner, ASR. In *LREC*, pages 4004–4011.

Chatino Language Documentation Project. (2017). Chatino Language Documentation Project Collection.

Collobert, R., Weston, J., and Karlen, M. (2011). Natural Language Processing (almost) from Scratch. 1:1–34.

Cruz, E. and Woodbury, T. (2006). El sandhi de los tonos en el Chatino de Quiahije. In *Las memorias del Congreso de Idiomas Indígenas de Latinoamérica-II*. Archive of the Indigenous Languages of Latin America.

Cruz, E. and Woodbury, T. (2014). Finding a way into a family of tone languages: The story and methods of the Chatino Language Documentation Project. *Language Documentation and Conservation*, 8:490–524.

Cruz, E. (2011). *Phonology, tone and the functions of tone in San Juan Quiahije Chatino*. Ph.D., University of Texas at Austin, Austin.

Deng, L., Hinton, G., and Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: an overview. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8599–8603. IEEE, may.

Do, T.-N.-D., Michaud, A., and Castelli, E. (2014). Towards the automatic processing of Yongning Na (Sino-Tibetan): developing a 'light' acoustic model of the target language and testing 'heavyweight' models from five national languages. In *4th International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU 2014)*, pages 153–160, St Petersburg, Russia, may.

Feng, Y.-M., Xu, L., Zhou, N., Yang, G., and Yin, S.-K. (2012). Sine-wave speech recognition in a tonal language. *The Journal of the Acoustical Society of America*, 131(2):EL133–EL138.

Ghahremani, P., BabaAli, B., Povey, D., Riedhammer, K., Trmal, J., and Khudanpur, S. (2014). A pitch extraction algorithm tuned for automatic speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 2494–2498. IEEE.

Girshick, R. (2015). Fast R-CNN. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448. IEEE, dec.

Graves, A., Fernandez, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist Temporal Classification : Labelling Unsegmented Sequence Data with Recurrent Neural Networks. *Proceedings of the 23rd international conference on Machine Learning*, pages 369–376.

Himmelmann, N. (2006). Language documentation: what is it and what is it good for? In J. Gippert, et al., editors, *Essentials of language documentation*, pages 1–30. de Gruyter, Berlin/New York.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Others. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jacobson, M., Michailovsky, B., and Lowe, J. B. (2001). Linguistic documents synchronizing sound and text. *Speech Communication*, 33:79–96.

Le, V.-B. and Besacier, L. (2009). Automatic speech recognition for under-resourced languages: application to Vietnamese language. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(8):1471–1482.

Lee, T., Lau, W., Wong, Y. W., and Ching, P. C. (2002). Using tone information in Cantonese continuous speech recognition. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(1):83–102.

Metze, F., Sheikh, Z. A. W., Waibel, A., Gehring, J., Kilgour, K., Nguyen, Q. B., and Nguyen, V. H. (2013). Models of tone for tonal and non-tonal languages. *2013 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2013 - Proceedings*, pages 261–266.

Michailovsky, B., Mazaudon, M., Michaud, A., Guillaume, S., François, A., and Adamou, E. (2014). Documenting and researching endangered languages: the Pangloss Collection. *Language Documentation and Conservation*, 8:119–135.

Michaud, A. (2017). *Tone in Yongning Na: lexical tones and morphotonology*. Number 13 in Studies in Diversity Linguistics. Language Science Press, Berlin.

Mortensen, D. R., Littell, P., Bharadwaj, A., Goyal, K., Dyer, C., and Levin, L. (2016). PanPhon: A Resource for Mapping IPA Segments to Articulatory Feature Vectors. *Proceedings of COLING 2016, the 26th Interna-*

tional Conference on Computational Linguistics: Technical Papers, pages 3475–3484.

Müller, M., Stüker, S., and Waibel, A. (2017). Language Adaptive Multilingual CTC Speech Recognition. In Alexey Karpov, et al., editors, *Speech and Computer: 19th International Conference, SPECOM 2017, Hatfield, UK, September 12-16, 2017, Proceedings*, pages 473–482. Springer International Publishing, Cham.

Niebuhr, O. and Michaud, A. (2015). Speech data acquisition: the underestimated challenge. *KALIPHO - Kieler Arbeiten zur Linguistik und Phonetik*, 3:1–42.

Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., and Pande, V. (2015). Massively Multitask Networks for Drug Discovery. feb.

Ruder, S. (2017). An Overview of Multi-Task Learning in Deep Neural Networks. jun.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Gary F. Simons et al., editors. (2017). *Ethnologue: languages of the world*. SIL International, Dallas, twentieth edition edition.

Thieberger, N. (2016). Documentary linguistics: methodological challenges and innovatory responses. *Applied Linguistics*, 37(1):88–99.

Vu, N. T., Imseng, D., Povey, D., Motlicek, P., Schultz, T., and Bourlard, H. (2014). Multilingual deep neural network based acoustic modeling for rapid language adaptation. In *Proceedings of the 39th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7639–7643, Florence, Italy.

Xu, P. and Fung, P. (2013). Cross-lingual language modeling for low-resource speech recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 21(6):1134–1144.

Xu, H., Su, H., Ni, C., Xiao, X., Huang, H., Chng, E.-S., and Li, H. (2016). Semi-supervised and Cross-lingual Knowledge Transfer Learnings for DNN Hybrid Acoustic Models under Low-resource Conditions. In *Proceedings of the Annual Conference of the International Speech Communication Association, (INTERSPEECH)*, pages 1315–1319, San Francisco, USA.

10. Language Resource References

Cavar, D. and Cavar, M. and Cruz, H. (2016). *Chatino Speech Corpus Archive Dataset*. ELRA, ISLRN 557-415-504-956-6.

Michaud, A. and Latami, D. (2017a). *Housebuilding 2*. Pangloss Collection: http://lacito.vjf.cnrs.fr/pangloss/.

Michaud, A. and Latami, D. (2017b). *Yongning Na Corpus*. Pangloss Collection: http://lacito.vjf.cnrs.fr/pangloss/.

A Very Low Resource Language Speech Corpus for Computational Language Documentation Experiments

P. Godard*, G. Adda*, M. Adda-Decker†, J. Benjumea‡, L. Besacier ⋆, J. Cooper-Leavitt*, G-N. Kouarata†, L. Lamel*, H. Maynard*, M. Müller◇, A. Rialland†, S. Stüker◇, F. Yvon*, M. Zanon-Boito ⋆

*LIMSI, CNRS, Université Paris-Saclay, Orsay, France

†LPP, CNRS, Paris, France

‡ENS, EHESS, PSL Research University, CNRS, INRIA, Paris, France

⋆LIG, UGA, G-INP, CNRS, INRIA, Grenoble, France

◇KIT, Karlsruhe, Germany

contact: pierre.godard@limsi.fr, laurent.besacier@univ-grenoble-alpes.fr

Abstract

Most speech and language technologies are trained with massive amounts of speech and text information. However, most of the world languages do not have such resources and some even lack a stable orthography. Building systems under these almost zero resource conditions is not only promising for speech technology but also for computational language documentation. The goal of computational language documentation is to help field linguists to (semi-)automatically analyze and annotate audio recordings of endangered, unwritten languages. Example tasks are automatic phoneme discovery or lexicon discovery from the speech signal. This paper presents a speech corpus collected during a realistic language documentation process. It is made up of 5k speech utterances in Mboshi (Bantu C25) aligned to French text translations. Speech transcriptions are also made available: they correspond to a non-standard graphemic form close to the language phonology. We detail how the data was collected, cleaned and processed and we illustrate its use through a zero-resource task: spoken term discovery. The dataset is made available to the community for reproducible computational language documentation experiments and their evaluation.

Keywords: language documentation, field linguistics, spoken term discovery, word segmentation, zero resource technologies, unwritten languages.

1. Introduction

Many languages will face extinction in the coming decades. Half of the 7,000 languages spoken worldwide are expected to disappear by the end of this century (Austin and Sallabank, 2011), and there are too few field linguists to document all of these endangered languages. Innovative speech data collection methodologies (Bird et al., 2014; Blachon et al., 2016) as well as computational assistance (Adda et al., 2016; Stüker et al., 2016) were recently proposed to help them in their documentation and description work.

As more and more researches are related to computational language documentation (Duong et al., 2016; Franke et al., 2016a; Godard et al., 2016; Anastasopoulos and Chiang, 2017), there is a need of realistic corpora to fuel reproducible and replicable language studies at the phonetic, lexical and syntactic levels. To our knowledge, very few corpora are available for computational analysis of endangered languages.[1]

Our work follows this objective and presents a speech dataset collected following a real language documentation scenario. It is multilingual (Mboshi speech aligned to French text) and contains linguists' transcriptions in Mboshi (in the form of a non-standard graphemic form close to the language phonology). The corpus is also enriched with automatic forced-alignment between speech and transcriptions. The dataset is made available to the research community[2]. This dataset is part of a larger data collection conducted on Mboshi language and presented in a companion paper (Rialland et al., 2018).

Expected impact of this work is the evaluation of efficient and reproducible computational language documentation approaches in order to face fast and inflexible extinction of languages.

This paper is organized as follows: after presenting the language of interest (Mboshi) in section 2., we describe the data collection and processing in sections 3. and 4. respectively. Section 5. illustrates its first use for an unsupervised word discovery task. Our spoken term detection pipeline is also presented and evaluated in this section. Finally, section 6. concludes this work and gives some perspectives

2. Language Description

Mboshi (Bantu C25) is a typical Bantu language spoken in Congo-Brazzaville. It is one of the languages documented by the BULB (Breaking the Unwritten Language Barrier) project (Adda et al., 2016; Stüker et al., 2016).

Phonetics, phonology and transcription Mboshi has a seven vowel system (i, e, ɛ, a, ɔ, o, u) with an opposition between long and short vowels. Its consonantal system includes the following phonemes: p, t, k, b, d, β, l, r, m, n, ɲ, mb, nd, ndz, ng, mbv, f, s, ɣ , pf, bv, ts, dz, w, j. It has a set of prenasalized consonants (mb, nd, ndz, ng, mbv) which are common in Bantu languages (Embanga Aborobongui, 2013; Kouarata, 2014).

[1] We are only aware of a Griko-Italian corpus (Lekakou et al., 2013), and of a Basaa-French corpus (Hamlaoui et al., 2018).

[2] It will be made available for free from ELRA, but its current version is online on: `https://github.com/besacier/mboshi-french-parallel-corpus`

While the language can be considered as rarely written, linguists have nonetheless defined a non-standard graphemic form for it, considered to be close to the language phonology. Affricates and prenasalized plosives were coded using multiple symbols (e.g. two symbols for dz, three for mbv). Long and short vowels were coded respectively as V and as VV.

Mboshi displays a complex set of phonological rules. The deletion of a vowel before another vowel in particular, common in Bantu languages, occurs at 40% of word junctions (Rialland et al., 2015). This tends to obscure word segmentation and introduces an additional challenge for automatic processing.

Morphology Mboshi words are typically composed of roots and affixes, and almost always include at least one prefix, while the presence of several prefixes and one suffix is also very common. The suffix structure tends to consist of a single vowel V (e.g. -a or -i) whereas the prefix structure may be both CV and V. Most common syllable structures are V and CV, although CCV may arise due to the presence of affricates and prenasalized plosives mentioned above.

The noun class prefix system, another typical feature of Bantu languages, has an unusual rule of deletion targeting the consonant of prefixes[3]. The structure of the verbs, also characteristic of Bantu languages, follows: Subject Marker — Tense/Mood Marker — Root-derivative Extensions — Final Vowel. A verb can be very short or quite long, depending of the markers involved.

Prosody Mboshi prosodic system involves two tones and an intonational organization without downdrift (Rialland and Aborobongui, 2016). The high tone is coded using an acute accent on the vowel while low tone vowel has no special marker. Word root, prefix and suffix all bear specific tones which tend to be realized as such in their surface forms.[4] Tonal modifications may also arise from vowel deletion at word boundaries.

A productive combination of tonal contours in words can also take place due to the preceding and appended affixes. These tone combinations play an important grammatical role particularly in the differentiation of tenses. However, in Mboshi, the tones of the roots are not modified due to conjugations, unlike in many other Bantu languages.

3. Data Collection

We have recently introduced Lig_Aikuma[5], a mobile app specifically dedicated to fieldwork language documentation, which works both on android powered smartphones and tablets (Blachon et al., 2016). It relies on an initial smartphone application developed by (Bird et al., 2014) for the purpose of language documentation with an aim of long-term interpretability. We extended the initial app with the concept of retranslation (to produce oral translations of the initial recorded material) and speech elicitation from text or images (to collect speech utterances aligned to text or images). In that way, human annotation labels can be replaced by weaker signals in the form of parallel multimodal information (images or text in another language). Lig_Aikuma also implements the concept of *respeaking* initially introduced in (Woodbury, 2003). It involves listening to an original recording and repeating what was heard carefully and slowly. This results in a secondary recording that is much easier to analyze later on (analysis by a linguist or by a machine). So far, Lig_Aikuma was used to collect data in three unwritten African Bantu languages in close collaboration with three major European language documentation groups (LPP, LLACAN in France; ZAS in Germany).

Focusing on Mboshi data, our corpus was built both from translated reference sentences for oral language documentation (Bouquiaux and Thomas, 1976) and from a Mboshi dictionary (Beapami et al., 2000). Speech elicitation from text was performed by three speakers in Congo-Brazzaville and led to more than 5k spoken utterances. The corpus is split in two parts (*train* and *dev*) for which we give basic statistics in Table 1. We shuffled the corpus prior to splitting in order to have comparable distributions in terms of speakers and origin.[6] There is no text overlap for Mboshi transcriptions between the two parts.

language	split	#sent	#tokens	#types
Mboshi	train	4,616	27,563	6,196
	dev	514	2,993	1,146
French	train	4,616	38,843	4,927
	dev	514	4,283	1,175

Table 1: Corpus statistics for the Mboshi corpus

4. Data Processing

4.1. Cleaning, Pre-/Post-Processing

All the characters of the Mboshi transcription have been checked, in order to avoid multiple encodings of the same character. Some characters have also been transcoded so that a character with a diacritic effectively corresponds to the UTF-8 composition of the individual character with the diacritic. Incorrect sequences of tones (for instance tone on a consonant) have been corrected. It was also decided to remove the elision symbol in Mboshi.

On the French side, the translations were double-checked semi-automatically (using linux $aspell$ command followed by a manual process – 3.3% of initial sentences were corrected this way). The French translations were finally tokenized (using $tokenizer.perl$ from the Moses toolkit[7]) and lowercased. We propose an example of a sentence pair from our corpus in Figure 1.

4.2. Forced Alignment

As the linguists' transcriptions did not contain any timing information, the creation of timed alignments was necessary. The word and phoneme level alignments were

[3] A prefix consonant drops if the root begins with a consonant (Rialland et al., 2015).

[4] The distinction between high and low tones is phonological (see (Rialland and Aborobongui, 2016)).

[5] http://lig-aikuma.imag.fr

[6] Either (Bouquiaux and Thomas, 1976) or (Beapami et al., 2000).

[7] http://www.statmt.org/moses/

Mboshi	wáá ngá iwé léekundá ngá sá oyoá lendúma saa m ótéma
French	si je meurs enterrez-moi dans la forêt oyoa avec une guitare sur la poitrine

Figure 1: A tokenized and lowercased sentence pair example in our Mboshi-French corpus.

produced with what Cooper-Leavitt et al. refer to as 'A light-weight ASR tool' (Cooper-Leavitt et al., 2017a). The alignment tool is an ASR system that is used in a forced-alignment mode. That is, it is used to associate words in the provided orthographic level transcription with the corresponding audio segments making use of a pronunciation lexicon which represents each word with one or more pronunciations (phonemic forms). The word-position-independent GMM-HMM monophone models are trained using the STK tools at LIMSI (Lamel and Gauvain, 2015). A set of 68 phonemes are used to represent the pronunciation dictionary, with multiple symbols for each vowel representing different tones (Cooper-Leavitt et al., 2017b; Bedrosian, 1996) and a symbol for silence. The acoustic model is estimated iteratively, with 5 rounds of segmentation and parameter estimation, and the model resulting from the last iteration was used to resegment the data. Since vowel elision and morpheme deletion are known to be characteristic of the Mboshi language, variants explicitly allowing their presence or absence are included in the pronunciation lexicon. Details of how this was implemented can be found in (Cooper-Leavitt et al., 2017a).

5. Illustration: Unsupervised Word Discovery from Speech

In this section, we illustrate how our corpus can be used to evaluate unsupervised discovery of word units from raw speech. This task corresponds to *Track 2* in the *Zero Resource Speech Challenge 2017*.[8] We present below the evaluation metrics used as well as a monolingual baseline system which does not take advantage yet of the French translations aligned to the speech utterances (bilingual approaches may be also evaluated with this dataset but we leave that for future work).

5.1. Evaluation Metrics

Word discovery is evaluated using the *Boundary*, *Token* and *Type* metrics from the *Zero Resource Challenge 2017*, extensively described in (Ludusan et al., 2014) and (Dunbar et al., 2017). They measure the quality of a word segmentation and the discovered boundaries with respect to the gold corpus. For these metrics, the precision, recall and F-score are computed; the *Token* recall is defined as the probability for a gold word token to belong to a cluster of discovered tokens (averaging over all the gold tokens), while the *Token* precision represents the probability that a discovered token will match a gold token. The F-score is the harmonic mean between these two scores. Similar definitions are applied to the *Type* and *Boundary* metrics.

5.2. Unsupervised Word Discovery Pipeline

Our baseline for word discovery from speech involves the cascading of the following two modules:

- unsupervised phone discovery (UPD) from speech: find pseudo-phone units from the spoken input,
- unsupervised word discovery (UWD): find lexical units from the sequence of pseudo-phone units found in the previous step.

Unsupervised phone discovery from speech (UPD)
In order to discover a set of phone-like units, we use the KIT system which is a three step process. First, phoneme boundaries are found using BLSTM as described in (Franke et al., 2016b). For each speech segment, articulatory features are extracted (see (Müller et al., 2017a) for more details). Finally, segments are clustered based on these articulatory features (Müller et al., 2017b). So, the number of clusters (pseudo phones) can be controlled during this process.

Unsupervised word discovery (UWD)
To perform unsupervised word discovery, we use $dpseg$ (Goldwater et al., 2006; Goldwater et al., 2009).[9] It implements a Bayesian non-parametric approach, where (pseudo)-morphs are generated by a bigram model over a non-finite inventory, through the use of a Dirichlet-Process. Estimation is performed through Gibbs sampling.
Godard et al. (2016) compare this method to more complex models on a smaller Mboshi corpus, and demonstrate that it produces stable and competitive results, although it tends to oversegment the input, achieving very high recall and a lower precision.

5.3. Results

Word discovery results are given in Tables 2, 3 and 4 for *Boundary*, *Token* and *Type* metrics respectively.
We compare our results to a pure speech based baseline which does pair-matching using locally sensitive hashing applied to PLP features and then groups pairs using graph clustering (Jansen and Van Durme, 2011). The parameters of this baseline spoken term discovery system are the same[10] as the ones used for the *Zero Resource Challenge 2017* (Dunbar et al., 2017).
A topline where $dpseg$ is applied to the gold forced alignments (phone boundaries are considered to be perfect) is also evaluated.
For the pipeline, we experience with different granularities of the UPD system (5, 30 and 60 units obtained after the clustering step). For each granularity, we perform 3 runs and report the averaged results.
We note that the baseline provided by the system of (Jansen and Van Durme, 2011) has a low coverage (few matches). Given that our proposed pipeline provides an exhaustive

[8] http://zerospeech.com

[9] http://homepages.inf.ed.ac.uk/sgwater/resources.html

[10] Notably the DTW threshold – which controls the number of spoken clusters found – is set to 0.90 in our experiment

method	P	R	F
gold FA phones + dpseg	53.8	83.5	65.4
(Jansen and Van Durme, 2011)	31.9	13.8	19.3
UPD+dpseg pipeline (5 units)	27.4	46.5	34.5
UPD+dpseg pipeline (30 units)	24.6	58.9	34.7
UPD+dpseg pipeline (60 units)	24.4	60.2	34.8

Table 2: Precision, Recall and F-measure on word boundaries. Pipeline compared with an unsupervised system based on (Jansen and Van Durme, 2011).

method	P	R	F
gold FA phones + dpseg	20.8	35.2	26.2
(Jansen and Van Durme, 2011)	4.5	1.6	2.4
UPD+dpseg pipeline (5 units)	2.1	4.4	2.9
UPD+dpseg pipeline (30 units)	1.4	4.6	2.2
UPD+dpseg pipeline (60 units)	1.4	4.7	2.1

Table 3: Precision, Recall and F-measure on word tokens. Pipeline compared with an unsupervised system based on (Jansen and Van Durme, 2011).

parse of the speech signals, it is guaranteed to have full coverage and, thus, shows better performance according to the *Boundary* metric. The quality of segmentation, in terms of tokens and types is, however, still low for all systems. Increasing the number of pseudo phone units improves *Boundary* recall but reduces precision. For *Token* and *Type* metrics, a coarser granularity provides slightly better results.

6. Conclusion

We have presented a speech corpus in Mboshi made available to the research community for reproducible computational language documentation experiments. As an illustration, we presented the first unsupervised word discovery (UWD) experiments applied to a truly unwritten language (Mboshi).

The results obtained with our pipeline on pseudo-phones are still far behind those obtained with gold labels, which indicates that there is room for improvement for the UPD module of our pipeline. The UWD module should, in future work, make use of the bilingual data available to improve the quality of the segmentation.

Future work also includes enriching our dataset with some alignments at the word level, in order to evaluate a bilingual lexicon discovery task. This is possible with encoder-decoder approaches, as shown in (Zanon Boito et al., 2017). As we distribute this corpus, we hope that this will help the community to strengthen its effort to improve the technologies currently available to support linguists in documenting endangered languages.

7. Acknowledgements

This work was partly funded by the French ANR and the German DFG under grant ANR-14-CE35-0002.

Some experiments reported of this paper were performed during the Jelinek Summer Workshop on Speech and Language Technology (JSALT) 2017 in CMU, Pittsburgh, which was supported by JHU and CMU (via grants from Google, Microsoft, Amazon, Facebook, and Apple).

method	P	R	F
gold FA phones + dpseg	7.5	13.9	9.7
(Jansen and Van Durme, 2011)	4.6	4.8	4.7
UPD+dpseg pipeline (5 units)	2.5	6.7	3.6
UPD+dpseg pipeline (30 units)	2.0	4.5	2.8
UPD+dpseg pipeline (60 units)	2.0	4.4	2.7

Table 4: Precision, Recall and F-measure on word types. Pipeline compared with an unsupervised system based on (Jansen and Van Durme, 2011).

8. Bibliographical References

Adda, G., Stüker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon, F., and Zerbian, S. (2016). Breaking the unwritten language barrier: The Bulb project. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia.

Anastasopoulos, A. and Chiang, D. (2017). A case study on using speech-to-translation alignments for language documentation. pages 170–178. Association for Computational Linguistics.

Peter K. Austin et al., editors. (2011). *The Cambridge Handbook of Endangered Languages*. Cambridge University Press.

Beapami, R. P., Chatfield, R., Kouarata, G., and Embengue-Waldschmidt, A. (2000). *Dictionnaire Mbochi-Français*. SIL-Congo Publishers, Congo (Brazzaville).

Bedrosian, P. (1996). The Mboshi noun class system. *Journal of West African Languages*, 26(1):27–47.

Bird, S., Hanke, F. R., Adams, O., and Lee, H. (2014). Aikuma: A mobile app for collaborative language documentation. *ACL 2014*, page 1.

Blachon, D., Gauthier, E., Besacier, L., Kouarata, G.-N., Adda-Decker, M., and Rialland, A. (2016). Parallel speech collection for under-resourced language studies using the LIG-Aikuma mobile device app. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia, May.

Luc Bouquiaux et al., editors. (1976). *Enquête et description des langues à tradition orale*. SELAF, Paris.

Cooper-Leavitt, J., Lamel, L., Adda, G., Adda-Decker, M., and Rialland, A. (2017a). Corpus base linguistic exploration via forced alignments with a 'light-weight' ASR tool. In *Workshop on Language Technology for Less Resourced Languages (LT-LRL) at the 8th Language & Technology Conference*, Poznań, Poland, November.

Cooper-Leavitt, J., Lamel, L., Rialland, A., Adda-Decker, M., and Adda, G. (2017b). Developing an Embosi (Bantu C25) speech variant dictionary to model vowel elision and morpheme deletion. In *ISCA Interspeech*.

Dunbar, E., Nga Cao, X., Benjumea, J., Karadayi, J., Bernard, M., Besacier, L., Anguera, X., and Dupoux, E.

(2017). The zero resource speech challenge 2017. In *Automatic Speech Recognition and Understanding (ASRU), 2017 IEEE Workshop on*. IEEE.

Duong, L., Anastasopoulos, A., Chiang, D., Bird, S., and Cohn, T. (2016). An attentional model for speech translation without transcription. In *Proceedings of NAACL-HLT*, pages 949–959.

Embanga Aborobongui, G. M. (2013). *Processus segmentaux et tonals en Mbondzi – (variété de la langue embosi C25)*. Ph.D. thesis, Université Paris 3 Sorbonne Nouvelle.

Franke, J., Müller, M., Hamlaoui, F., Stüker, S., and Waibel, A. (2016a). Phoneme boundary detection using deep bidirectional LSTMs. In *Speech Communication; 12. ITG Symposium; Proceedings of*, pages 1–5. VDE.

Franke, J., Müller, M., Stüker, S., and Waibel, A. (2016b). Phoneme boundary detection using deep bidirectional lstms. In *Speech Communication; 12. ITG Symposium; Proceedings of*. VDE.

Godard, P., Adda, G., Adda-Decker, M., Allauzen, A., Besacier, L., Bonneau-Maynard, H., Kouarata, G.-N., Löser, K., Rialland, A., and Yvon, F. (2016). Preliminary Experiments on Unsupervised Word Discovery in Mboshi. In *Interspeech 2016*, San Francisco, California, USA.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2006). Contextual dependencies in unsupervised word segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 673–680, Sydney, Australia, July. Association for Computational Linguistics.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2009). A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54.

Hamlaoui, F., Makasso, E.-M., Müller, M., Engelmann, J., Adda, G., Waibel, A., and Stüker, S. (2018). BULBasaa: A bilingual Bàsàá-French speech corpus for the evaluation of language documentation tools. In *LREC 2018 (in press)*, Japan.

Jansen, A. and Van Durme, B. (2011). Efficient spoken term discovery using randomized algorithms. In *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, pages 401–406. IEEE.

Kouarata, G.-N. (2014). *Variations de formes dans la langue mbochi (Bantu C25)*. Ph.D. thesis, Université Lumière Lyon 2.

Lamel, L. and Gauvain, J.-L., (2015). *The Oxford Handbook of Computational Linguistics*, chapter Speech Recognition. Oxford University Press.

Lekakou, M., Baldissera, V., and Anastasopoulos, A. (2013). Documentation and analysis of an endangered language: aspects of the grammar of Griko.

Ludusan, B., Versteegh, M., Jansen, A., Gravier, G., Cao, X.-N., Johnson, M., and Dupoux, E. (2014). Bridging the gap between speech technology and natural language processing: an evaluation toolbox for term discovery systems. In *Proceedings of LREC*.

Müller, M., Franke, J., Stüker, S., and Waibel, A. (2017a). Improving phoneme set discovery for documenting unwritten languages. *Elektronische Sprachsignalverarbeitung (ESSV) 2017*.

Müller, M., Franke, J., Stüker, S., and Waibel, A. (2017b). Towards phoneme inventory discovery for documentation of unwritten languages. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE.

Rialland, A. and Aborobongui, M. E. (2016). How intonations interact with tones in Embosi (Bantu C25), a two-tone language without downdrift. In *Intonation in African Tone Languages*, volume 24. De Gruyter, Berlin, Boston.

Rialland, A., Embanga Aborobongui, G. M., Adda-Decker, M., and Lamel, L. (2015). Dropping of the class-prefix consonant, vowel elision and automatic phonological mining in Embosi. In *Proceedings of the 44th ACAL meeting*, pages 221–230, Somerville. Cascadilla.

Rialland, A., Adda-Decker, M., Kouarata, G.-N., Adda, G., Besacier, L., Lamel, L., Gauthier, E., Godard, P., and Cooper-Leavitt, J. (2018). Parallel Corpora in Mboshi (Bantu C25, Congo-Brazzaville). In *LREC 2018 (in press)*, Japan.

Stüker, S., Adda, G., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon, F., and Zerbian, S. (2016). Innovative technologies for under-resourced language documentation: The Bulb project. In *Proceedings of CCURL (Collaboration and Computing for Under-Resourced Languages : toward an Alliance for Digital Language Diversity)*, Portorož Slovenia.

Woodbury, A. C., (2003). *Defining documentary linguistics*, volume 1, pages 35–51. Language Documentation and Description, SOAS.

Zanon Boito, M., Berard, A., Villavicencio, A., and Besacier, L. (2017). Unwritten languages demand attention too! word discovery with encoder-decoder models. In *Automatic Speech Recognition and Understanding (ASRU), 2017 IEEE Workshop on*. IEEE.

Chahta Anumpa: A Multimodal Corpus of the Choctaw Language

Jacqueline Brixey, Eli Pincus, Ron Artstein
USC Institute for Creative Technologies
12015 E Waterfront Dr, Los Angeles, CA 90094
{brixey, pincus, artstein}@ict.usc.edu

Abstract

This paper presents a general use corpus for the Native American indigenous language Choctaw. The corpus contains audio, video, and text resources, with many texts also translated in English. The Oklahoma Choctaw and the Mississippi Choctaw variants of the language are represented in the corpus. The data set provides documentation support for the threatened language, and allows researchers and language teachers access to a diverse collection of resources.

Keywords: endangered languages, indigenous language, multimodal, Choctaw, Native American languages

1. Introduction

This paper introduces a general use corpus for Choctaw, an American indigenous language. The Choctaw language is spoken by the Choctaw tribe, who originally inhabited the southeastern United States. The tribe is the fourth largest indigenous group by population in the United States with 220,000 enrolled members.[1] The Choctaw language, however, is classified as "Threatened" by Ethnologue,[2] as there are only 10,400 fluent speakers and the language is losing users.

While sporadic written records of the language appear as early as 1715 (Sturtevant, 2005, page 16), systematic writing of Choctaw only began with the arrival of American missionaries, led by Rev. Cyrus Byington in 1819 (introduction to Byington, 1870). The first text published in Choctaw was the Bible, and this is still the longest published text in the language today. Efforts during the 1900s aimed to forcefully assimilate Native Americans and suppress indigenous languages (Battiste and Henderson, 2000). For this reason, many Native Americans did not learn their ancestral language, and few works are publicly published in these languages. As a result of this history, the Choctaw language has few published works and little text representation online.

While work has been undertaken to document the language and conduct linguistic studies, no digital corpus of the Choctaw language exists. The present work has gathered sparse text resources representing different variants of the language from teaching materials, books, and scholarly material. Video and transcribed audio clips where the Choctaw language is spoken in complete phrases are also included in this corpus. The majority of data in this work has an accompanying English translation, with a smaller portion in monolingual Choctaw.

For most of the world's languages, no data in a machine-readable format are available for human language technology applications. Compared to well-resourced languages such as English and French, natural language processing tools for Native American indigenous languages are still not abundant. The goal of this database is thus to first preserve a threatened language. The second contribution is to compile a comprehensive data set of existing resources for novel research opportunities in history, linguistics, and natural language processing. The final contribution is to provide documentation of the language for language learners and teachers, in order to assist in revitalization efforts.

2. Choctaw tribe and language

2.1. People and dialects

The Choctaw tribe originally resided in the southeast of the United States, in what today would be Alabama, Louisiana, and Mississippi. In the early 1830s the Choctaws were forcibly relocated to Oklahoma in the migration known as the "Trail of Tears", though some remained in Mississippi. Today there are three Federally recognized Choctaw tribes:[3] Jena Band of Choctaw Indians (in Louisiana), Mississippi Band of Choctaw Indians, and The Choctaw Nation of Oklahoma. Speakers are concentrated primarily in Mississippi and Oklahoma (Ulrich, 1993), but also in Alabama, Louisiana, Texas, and California. For many speakers in Oklahoma (Williams, 1999), Choctaw is their second language, and revitalization efforts have worked to establish language courses at local schools. Choctaw is spoken by all ages in Mississippi, but is losing speakers over time.

The language belongs to the Western Muskogean language family; it is most closely related to Chickasaw, and also related to the Creek and Seminole languages (Haas, 1979). Dialectal variation in Choctaw has been the matter of some debate (Nicklas, 1972; Broadwell, 2005; Broadwell, 2006): while sources agree that there are three dialect variants in Mississippi which reflect original settlement patterns, it is unclear whether and to what extent similar variation has been carried over to Oklahoma. Broadwell (2006) identifies four present-day regional variants: Mississippi Choctaw, Oklahoma Choctaw, Louisiana Choctaw, and Mississippi Choctaw of Oklahoma; the latter is spoken by Choctaws who live in The Chickasaw Nation in Oklahoma, and are believed to have been relocated there from Mississippi in

[1] https://web.archive.org/web/20120512040555/http://www.ok.gov/oiac/documents/2011.FINAL.WEB.pdf

[2] https://www.ethnologue.com/language/cho

[3] https://www.federalregister.gov/documents/2017/01/17/2017-00912/indian-entities-recognized-and-eligible-to-receive-services-from-the-united-states-bureau-of-indian

Consonants												
p	b	t	k	f	s	h	m	n	l	w	y [j]	
[tʃ]	ch, č				[ʃ]	sh, š			[ɬ]	hl, lh, ł		
Vowels												
[a]	a, ʊ, v				[i]	i			[o]	o, u		
[aː]	a, á, aa				[iː]	e, í, i, ii, ie			[oː]	o, ó, oo		
[ã]	a̲, an, am, ą				[ĩ]	i̲, in, im, į			[õ]	o̲, u̲, on, om, ǫ		

Figure 1: Choctaw sounds and orthographic variants.

the early 1900s. Overall, regional variation in Choctaw is fairly minor, with some variation in phonetic detail (Ulrich, 1993) and a small number of lexical differences (for example, the word for "onion" is typically hato̲fʊlaha in Oklahoma and shatshonna in Mississippi). A much larger source of variation in our corpus is differing orthographic conventions.

2.2. Sounds and orthography

The Choctaw language has 15 consonants, and 9 vowels in three series: short, long, and nasalized; the orthography uses the Latin script, but is not fully standardized (Figure 1). Broadwell (2006, section 1.2) identifies three broad writing systems: traditional, Mississippi, and modified traditional; our collected materials, however, show more variation. Generally, Oklahoma speakers write the short and long vowels as ʊ-a, i-e, and u-o (with v often substituting for ʊ), while Mississippi speakers use a-á, i-e, and o-ó. Mississippi speakers consistently write the lateral fricative [ɬ] as lh, while Oklahoma speakers use lh before a consonant and hl otherwise. In both variants, long vowels are sometimes but not always doubled, and nasalized vowels are sometimes represented with a macron below and sometimes with a following nasal consonant character. Broadwell (2006) characterizes the use of č, š, ł, ą, į, and ǫ as Mississippi Choctaw, but we have found it primarily in instructional materials rather than general Mississippi use.

In addition to the above conventions which are in wide use, our sources include idiosyncratic variants that are limited to a single work: The dictionary of Byington (1915), published nearly 50 years after his death, has unique symbols introduced by the editor: a̤ for short [a], and a superscript for nasalzed vowels a^{n}, i^{n}, o^{n}. Nicklas (1972) represents long vowels with a macron above ā, ī, ō.

Choctaw has additional sounds which are not indicated orthographically: a glottal stop whose phonemic status is unclear (Broadwell, 2006, section 2.6), and a lexically specified pitch accent in certain words and inflections (Nicklas, 1972, section 1.4). The glottal stop and pitch accent are rarely if ever contrastive and are not indicated in ordinary writing, but some scholarly or instructional materials in the corpus indicate them by an apostrophe or accent mark.

In our corpus, each entry retains the orthographic convention of its source.

2.3. Morphology and syntax

Choctaw word order is subject-object-verb, and adjectives follow the nouns they modify. Choctaw also has a complex morphology system, including infixation. The following examples illustrate some of the morphology features of Choctaw.

1. I̲kana ʊt aiittʊtoba ia tuk
   I̲-kana ʊt aiittʊtoba ia tuk
   his-friend the store go PAST
   His friend went to the store.

2. Ashekonopa ilʊppʊt tahakchi li
   Ashekonopa ilʊppʊt ta<ha>kchi li
   knot this tie<quickly> 1SG
   I tie this knot quickly.

3. Chik impo
   Chik imp-o
   not.2SG eat-NEG
   You are not eating.

Orthographic conventions vary regarding the segmentation of words and morphemes. Early works tended to segment words into shorter units than is customary now (Broadwell, 2006), but even today segmentation is not consistent: some texts show agglutination of all morphemes (Nicklas, 1979), while others specifically call for separation of some morphemes (The Choctaw Nation of Oklahoma Dictionary Committee, 2016).

2.4. Reference works

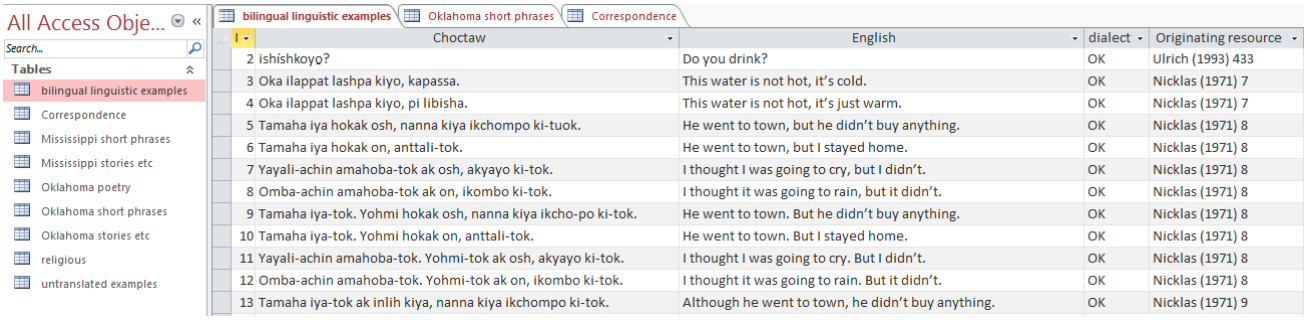
The language has three reference grammars, the first by the missionary Cyrus Byington (1870, published shortly after his death), the second by Thurston Dale Nicklas (1979), and the final by George Aaron Broadwell (2006). Several dictionaries also exist, the first written by Cyrus Byington (1915), while a more recent dictionary was released by The Choctaw Nation of Oklahoma (2016).

3. Data Collection

The long-term goal of this project is to continually add newly published and newly discovered data in the Choctaw language to this corpus, to encourage language research and preservation. All work was approved for study by the Choctaw Nation of Oklahoma, with whom the first author is an enrolled member.

The current repository was formed using printed written and oral teaching materials, examples from academic research sources, audio clips, and videos. In order to adhere to copyright standards, all text entries include the appropriate identifying source information, such as author and publication title. We are currently in the process of seeking permissions to make this corpus publicly available.

All text data was manually entered into a Microsoft Access database (Figure 2). Published teaching material was gathered from the Choctaw Nation of Oklahoma and the Mississippi Band of Choctaw Indians. Marcia Haag and Henry Willis's two books of teaching material, poetry, short stories, and correspondence formed a large portion of the Oklahoma Choctaw portion of the corpus (Haag and Willis, 2001; Haag and Willis, 2007). Archives from the Los Angeles Unified School District Indian Education Program provided teaching materials and scholarly resources for the Oklahoma and Mississippi variants.



	Choctaw	English	dialect	Originating resource
2	ishíshkoyo?	Do you drink?	OK	Ulrich (1993) 433
3	Oka ilappat lashpa kiyo, kapassa.	This water is not hot, it's cold.	OK	Nicklas (1971) 7
4	Oka ilappat lashpa kiyo, pi libisha.	This water is not hot, it's just warm.	OK	Nicklas (1971) 7
5	Tamaha iya hokak osh, nanna kiya ikchompo ki-tuok.	He went to town, but he didn't buy anything.	OK	Nicklas (1971) 8
6	Tamaha iya hokak on, anttali-tok.	He went to town, but I stayed home.	OK	Nicklas (1971) 8
7	Yayali-achin amahoba-tok ak osh, akyayo ki-tok.	I thought I was going to cry, but I didn't.	OK	Nicklas (1971) 8
8	Omba-achin amahoba-tok ak on, ikombo ki-tok.	I thought it was going to rain, but it didn't.	OK	Nicklas (1971) 8
9	Tamaha iya-tok. Yohmi hokak osh, nanna kiya ikcho-po ki-tok.	He went to town. But he didn't buy anything.	OK	Nicklas (1971) 8
10	Tamaha iya-tok. Yohmi hokak on, anttali-tok.	He went to town. But I stayed home.	OK	Nicklas (1971) 8
11	Yayali-achin amahoba-tok. Yohmi-tok ak osh, akyayo ki-tok.	I thought I was going to cry. But I didn't.	OK	Nicklas (1971) 8
12	Omba-achin amahoba-tok. Yohmi-tok ak on, ikombo ki-tok.	I thought it was going to rain. But it didn't.	OK	Nicklas (1971) 8
13	Tamaha iya-tok ak inlih kiya, nanna kiya ikchompo ki-tok.	Although he went to town, he didn't buy anything.	OK	Nicklas (1971) 9

Figure 2: Screenshot of how bilingual text entries are stored in the database. All entries cite the originating resource.

Published linguistic articles and grey literature – unpublished theses and manuscripts – form the section of the corpus called "scholarly sources". In total, content in the target language was pulled from sixteen published articles, one unpublished manuscript, and two reference grammar books.

Scholarly sources and religious texts, such as hymns, were the only texts gathered from the internet. To the best of our knowledge, no blogs or web pages in only the target language exist. English texts about the history and culture of the tribe were not included in the corpus, as the primary focus of this corpus is to provide resources of the target language.

Over 580 audio files created by the Choctaw Nation of Oklahoma's School of Choctaw Language distance learning program were compiled for this corpus. All audio clips are for learning the language, and are approximately thirty seconds long. The clips cover a variety of topics. These clips are accompanied by a transcript written by the School of Choctaw Language in both Choctaw and English. Also within the audio portion of the corpus is one five-minute-long religious clip from Global Recordings Network discovered without transcription. Finally, 13 audio files of interviews and oral traditions collected by William D. Davies were accessed from the American Philosophical Society (APS).

This corpus also includes 30 mp4 video files downloaded from YouTube that contain Choctaw speech. Video files were downloaded from YouTube using an open source downloading tool.[4] Terms such as "choctaw", "chahta anumpa" (the native name for the language), as well as compound terms such as "choctaw songs" were used in order to search for YouTube videos containing the Choctaw language. Each video was manually checked in order to ensure at least some Choctaw speech was present.

3.1. Future linguistic collection

As this is an ongoing effort, this initial corpus will continue to grow as more material is collected. There are numerous recordings that cannot be accessed online as well as undigitized documents in historical archives in Oklahoma and Mississippi. Due to the nature of these delicate, old, and irreplaceable items, in-person visits are required in order to access and view them. Examples of archives are those held by the Choctaw Nation of Oklahoma (Durant, OK), the Sam Noble Museum (Norman, OK), and the Mississippi Band of Choctaw Indians Tribal Archives (Choctaw, MS). As funding and permissions are attained, it is expected that these remote sources will be digitized and added to this corpus.

[4]MacX YouTube Downloader

Type	OK	MS
Short stories	5954	1693
Phrases	10911	331
Poetry	243	0
Correspondence	159	0
Religious texts	222	30010
Examples from scholarly sources	589	12
Total	18078	32046

Table 1: Word token counts for bilingual types of texts in two variants

4. Data Set

The data set comprises text, audio, and video. Texts in the data set are stored in plain text in a database. All data points list the source reference of the text with each entry in the database. Bilingual text data is separated by variant and type. The two orthographic variants – Oklahoma Choctaw (OK) and Mississippi Choctaw (MS) – are kept separate in order to make future processing easier. No Louisiana texts were found for this work. The types of text collected include short stories, single sentence phrases, and poetry (Table 1). Monolingual Choctaw data is stored separately.

4.1. Short stories

Stories and poems were all gathered from teaching materials: 29 stories from Oklahoma and 11 from Mississippi. The stories cover a range of topics, from cultural and oral traditions, to personal anecdotes. There is overlap between the MS and OK subsections. For example the story "How the Possum got its Tail" appears in both, however the story is not told identically. The MS section has 1693 word tokens, while the OK section has 5954 word tokens. The average length of an OK story is 249 words, with a minimum word count of and 53, a max count of 1277 words. The average length of a MS story is 153, with a max value of 241 words, and a minimum count of 94.

Word (English translation)	Number of tokens
OK	
tuk (recent past marker)	252
mʊt (the)	126
chi̲ (future marker)	107
ʊt (the)	103
li (I)	86
ho̲ (question marker)	86
yʊt (that)	80
ish (you)	78
hosh (subject marker)	58
hattak (man)	52
MS	
oklah (people)	1049
at (the)	878
alhíha (to act truly)	720
im (him/her, before vowel)	691
yat (the, who)	643
hicha (and)	615
Jesus	545
i̲ (him/her, before consonant)	457
áchittók (he/she said)	410
ma̲ (the, which)	399

Table 2: Top 10 most frequent words in the OK and MS phrase sections

4.2. Phrases

To form the phrases portion of the data set, only grammatically complete phrases were included in the repository. A phrase is one complete sentence, such as "Iti isht bʊshli mʊt haksobali" (That chainsaw is loud), and "Impah" (She is eating). Over 2700 Oklahoma Choctaw phrases and 69 Mississippi phrases form this section. The average number of words in a phrase for both the MS and OK sections is four, with a maximum length of 15. One word representing a complete phrase in the Mississippi Choctaw of Oklahoma was found and is stored in this section of the database.

4.3. Poetry

One teaching resource (Haag and Willis, 2007) published three poems. Poetry provides novel insights into both the culture and symbology of a language, thus poetry was included in the corpus. The poetry section contains 243 word tokens from three poems.

4.4. Correspondence

Only one teaching resource (Haag and Willis, 2001) published a translated letter, with a total of 159 tokens. We expect to add to this section over time, as many letters are available on paper only and require an in-person visit to view the collection.

4.5. Religious texts

Many Choctaws today are affiliated with the Methodist, Presbyterian, or Baptist faiths. Seven OK and one MS bilingual hymns and doxologies form this section of the corpus, with a total number of 234 word tokens. Part of Genesis[5] and all of the Gospel of Mark[6] were digitized by YouVersion and are in the MS orthography. The texts from YouVersion comprise 29998 tokens. Adding the remaining portions of the Old and New Testament of the Bible is ongoing.

4.6. Scholarly sources

Examples from linguistic resources are unique as they are the only input for the corpus that contain annotation for part of speech. This section contains 589 word tokens in Choctaw, and the average input is 2 words. When annotation is available, it is also included within an entry in the database, thus each entry can potentially have a Choctaw, Choctaw annotation, and English version, along with corresponding reference information.

4.7. Monolingual texts

A section of monolingual published Choctaw texts is also included in the corpus. These monolingual texts are either short stories or single phrases. This section serves as a potential test bed for future research, such as in translation or morphology detection. This section contains 1386 total word tokens, with 41 words from Mississippi, one word from Mississippi Choctaw of Oklahoma, and the remainder from Oklahoma.

4.8. Audio

All audio clips, with the exception of the religious audio retrieved from Global Recordings Network, are accompanied by a text document with the Choctaw transcription and English translation document. On average, audio clips from the Choctaw Nation of Oklahoma School of Choctaw Language are thirty seconds long, thus there are 193 minutes of audio from this resource. In each clip, the single, grammatically complete phrase is repeated twice.

Audio originally recorded by Davies for the American Philosophical Society range in length from one minute to ten minutes, for a total of 58 minutes. These recordings do have some overlap in content with the short stories section: for example, the traditional story "How the turtle got the pattern on its back" occurs in both. All recordings are accompanied by transcripts in English and Choctaw.

The final audio recording, from Global Recordings Network,[7] is 5:40 minutes long. No transcript is provided, and no information about the speaker or recording location was given by Global Recordings Network.

4.9. Video

The 30 videos were manually annotated with one of 4 categories: Songs – 12 videos primarily containing a Choctaw song; Cultural – 3 videos that describe aspects of Choctaw culture; Story – 1 video that relates a story in Choctaw from the Choctaw culture; and Instructional – 14 videos that attempt to teach Choctaw to English speakers. The average video length is 3:51 minutes, shortest video 1:04, and the longest video is 12:29. In total, there is 105 minutes and 38 seconds of video in the current corpus. Among the videos,

[5] https://www.bible.com/bible/1927/GEN.37.CHTW
[6] https://www.bible.com/bible/1927/MRK.1.CHTW
[7] https://globalrecordings.net/it/language/318

16 contain bilingual content (Choctaw and English speech) while the remaining 14 contain monolingual content (only Choctaw speech).

5. Use Cases

The proposed use cases for the data set are numerous given the variety in both content and language variant. The primary use cases are for academic research in linguistics, history, and natural language processing (NLP), as well as serving to preserve the language.

Our future work is to annotate a training set of the complex morphemes of the language for study and to build a supervised model for morpheme segmentation. As the data set includes both audio and their transcriptions, an automatic speech recognition system could be developed.

One potential NLP use case is machine translation (MT). The majority of the data in our corpus is translated in English, creating a well-formed parallel data set. The language presents interesting challenges in this domain, as morphologically rich languages pose problems for MT systems from errors in word-alignment and multiple affixes. Current alignment models at word-level do not distinguish words and morphemes, and produce low-quality end translation due to misalignment (Li et al., 2016). The small size of the data set would encourage novel approaches for a MT model, as there is not enough data to use many machine learning techniques. However, as no system yet exists, a MT system would assist in generating new texts in Choctaw from English.

Cultural immersion is another use case for the corpus. Storytelling is an important means for sharing cultural norms and beliefs. We built an interactive bilingual text-based conversational agent that shares stories and parables about animals from the corpus. The agent was built using NPCEditor, a response classifier and dialogue management system (Leuski and Traum, 2011). NPCEditor employs a statistical classifier that is trained on linked questions and responses: for our agent, the questions are requests for particular stories, and the responses are the stories themselves. For example, a user might ask *Do you know a story about foxes?* and the classifier will find the most appropriate response. The agent can also make statements about itself, as well as greetings and closings to maintain dialogue flow. Some previous applications using NPCEditor include virtual museum guides (Swartout et al., 2010), a system for coversation with Holocaust survivors (Traum et al., 2015), and a Facebook Messenger chatbot to answer sexual health questions (Brixey et al., 2017)

Finally, the corpus serves as a repository for teaching and learning the language. As nearly all of the text entries are bilingual, learners and teachers alike can benefit from the translations.

6. Discussion and Conclusions

This paper introduced a multimodal data set of the low resource American indigenous language Choctaw. This data set comprises more than 50,000 word tokens in text, many with English translation, and 400 minutes of oral examples from audio and video. Future work aims to continuously develop the data set as new publications are released. Future work will also include unpublished resources for better representation of the Mississippi Choctaws of Oklahoma, to represent the Louisiana Choctaw, and add the Bible for all available variants to the database.

7. Acknowledgements

The first author was supported by a National GEM Consortium fellowship. This work was supported in part by the U.S. Army; statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

8. Bibliographical References

Battiste, M. and Henderson, J. S. Y. (2000). *Protecting Indigenous Knowledge and Heritage: A Global Challenge*. UBC Press.

Brixey, J., Hoegen, R., Lan, W., Rusow, J., Singla, K., Yin, X., Artstein, R., and Leuski, A. (2017). SHIHbot: A Facebook chatbot for sexual health information on HIV/AIDS. In *Proceedings of the SIGDIAL 2017 Conference*, pages 370–373, Saarbrücken, Germany, August.

Broadwell, G. A. (2005). Choctaw. In Heather K. Hardy et al., editors, *Native Languages of the Southeastern United States*, pages 157–199. U of Nebraska Press.

Broadwell, G. A. (2006). *A Choctaw Reference Grammar*. U of Nebraska Press.

Byington, C. (1870). Grammar of the Choctaw language. *Proceedings of the American Philosophical Society*, 11:317–367. Edited by Daniel Garrison Brinton. Also published as a monograph by McCalla and Stavely, Philadelphia, 1870.

Byington, C. (1915). *A Dictionary of the Choctaw Language*. US Government Printing Office. Edited by John R. Swanton and Henry S. Halbert. Smithsonian Institution Bureau of American Ethnology Bulletin 46.

Haag, M. and Willis, H. (2001). *Choctaw Language and Culture: Chahta Anumpa*, volume 1. University of Oklahoma Press.

Haag, M. and Willis, H. (2007). *Choctaw Language and Culture: Chahta Anumpa*, volume 2. University of Oklahoma Press.

Haas, M. R. (1979). Southeastern languages. In Lyle Campbell et al., editors, *The Languages of Native America: Historical and Comparative Assessment*, pages 299–326. University of Texas Press.

Leuski, A. and Traum, D. (2011). NPCEditor: Creating virtual human dialogue using information retrieval techniques. *AI Magazine*, 32(2):42–56.

Li, X., Tracey, J., Grimes, S., and Strassel, S. (2016). Uzbek-English and Turkish-English morpheme alignment corpora. In *LREC*.

Nicklas, T. D. (1972). *The Elements of Choctaw*. Ph.d. dissertation, University of Michigan.

Nicklas, T. D. (1979). *Reference grammar to the Choctaw Language*. Choctaw Bilingual Education Program, Southeastern Oklahoma State University.

Sturtevant, W. C. (2005). History of research on the native languages of the Southeast. In Heather K. Hardy et al., editors, *Native Languages of the Southeastern United States*, pages 8–65. U of Nebraska Press.

Swartout, W., Traum, D., Artstein, R., Noren, D., Debevec, P., Bronnenkant, K., Williams, J., Leuski, A., Narayanan, S., Piepol, D., Lane, C., Morie, J., Aggarwal, P., Liewer, M., Chiang, J.-Y., Gerten, J., Chu, S., and White, K. (2010). Ada and Grace: Toward realistic and engaging virtual museum guides. In Jan Allbeck, et al., editors, *Intelligent Virtual Agents: 10th International Conference, IVA 2010, Philadelphia, PA, USA, September 20–22, 2010 Proceedings*, volume 6356 of *Lecture Notes in Artificial Intelligence*, pages 286–300, Heidelberg, September. Springer.

The Choctaw Nation of Oklahoma Dictionary Committee. (2016). *Chahta Anumpa Tosholi Himona: New Choctaw Dictionary*. Choctaw Print Services, 1st edition.

Traum, D., Jones, A., Hays, K., Maio, H., Alexander, O., Artstein, R., Debevec, P., Gainer, A., Georgila, K., Haase, K., et al. (2015). New Dimensions in Testimony: Digitally preserving a Holocaust survivor's interactive storytelling. In Henrik Schoenau-Fog, et al., editors, *Interactive Storytelling: 8th International Conference on Interactive Digital Storytelling, ICIDS 2015, Copenhagen, Denmark, November 30–December 4, 2015, Proceedings*, volume 9445 of *Lecture Notes in Computer Science*, pages 269–281, Heidelberg, December. Springer.

Ulrich, C. H. (1993). The glottal stop in Western Muskogean. *International Journal of American Linguistics*, 59(4):430–441.

Williams, R. S. (1999). Referential tracking in Oklahoma Choctaw: Language obsolescence and attrition. *Anthropological Linguistics*, pages 54–74.

BULBasaa: A Bilingual Bàsàá-French Speech Corpus for the Evaluation of Language Documentation Tools

Fatima Hamlaoui[◇‡], Emmanuel-Moselly Makasso[‡], Markus Müller[*], Jonas Engelmann[‡], Gilles Adda[†], Alex Waibel[**], Sebastian Stüker[*]

[◇]University of Toronto, Toronto, Canada
[‡]Leibniz-Zentrum Allgemeine Sprachwissenschaft, Berlin, Germany
[*]Karlsruhe Institute of Technology, Karlsruhe, Germany
[†]LIMSI, CNRS, Paris, France
[⋆]LTI, CMU, Pittsburgh, USA
contact: f.hamlaoui@utoronto.ca, m.mueller@kit.edu, sebastian.stueker@kit.edu

Abstract

Bàsàá is one of the three Bantu languages of BULB (*Breaking the Unwritten Language Barrier*), a project whose aim is to provide NLP-based tools to support linguists in documenting under-resourced and unwritten languages. To develop technologies such as automatic phone transcription or machine translation, a massive amount of speech data is needed. Approximately 50 hours of Bàsàá speech were thus collected and then carefully re-spoken and orally translated into French in a controlled environment by a few bilingual speakers. For a subset of ≈10 hours of the corpus, each utterance was additionally phonetically transcribed to establish a golden standard for the output of our NLP tools. The experiments described in this paper are meant to provide an automatic phonetic transcription using a set of derived phone-like units. As every language features a specific set of idiosyncrasies, automating the process of phonetic unit discovery in its entirety is a challenging task. Within BULB, we envision a workflow where linguists are able to refine the set of automatically discovered units and the system is then able to re-iterate on the data, providing a better approximation of the actual phone set.

Keywords: Bàsàá, Northwest Bantu, Computational linguistics, unsupervised phone discovery, under-resourced languages

1. Introduction

About two-thirds of the approximately 7,000 languages spoken today count less than 1,000 speakers and are in danger of disappearing (Simons and Fennig, 2017). In a world that is rapidly changing, some estimate that 70% to 90% of the languages spoken today will fall out of use by the end of this century. To provide communities with modern tools that will increase the vitality of their language and support its use in a variety of contexts, as well as to help linguists in their efforts to learn about human cognition though the study of language diversity, the BULB (Breaking the Unwritten Language Barrier) project has been developing Natural Language Processing-based tools to help and accelerate language documentation. To achieve this goal, a critical mass of speech data is necessary. In the present paper, we describe the corpus created for Bàsàá (Cameroon), one of the three under-resourced Bantu languages on which the project concentrates and outline a method for the unsupervised evaluation of the derived phone set. The other two languages part the of the project are Myene (Gabon) and Mboshi (Republic of Congo) (Godard et al., 2018). We first provide basic information about Bàsàá in Section 2. In Section 3., we describe the data collection process and provide a detailed overview of the corpus. In the remainder of the paper, we concentrate on the subset of our data that was used for evaluation. Data preparation steps are outlined in Section 4. and an example use of the *BULBasaa* corpus is shown in Section 5. Section 6. concludes the paper.

2. Bàsàá

Bàsàá (A43 in (Guthrie, 1948)) is spoken in the south of Cameroon, by approximately 300,000 speakers (Simons and Fennig, 2017). It is a two tone language (High and Low), that on the surface contrasts High, Low, Falling, Rising and downstepped High tones. On the segmental level, Bàsàá has a 7-vowel system (Figure 1), in which vowel length is contrastive.

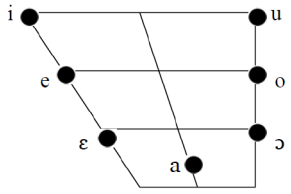


Figure 1: Bàsàá vowels (Makasso and Lee, 2015)

As discussed in detail by (Hyman, 2003), the phonology of Bàsàá is, in many ways, not very typical of Bantu languages. Both open and closed syllables are allowed, and surface syllable onsets are not required. Its consonant system is particularly complex (Table 1) and consonant oppositions depend on the position of the syllable within the prosodic stem. In a prosodic stem that contains up to three syllables (four consonants and three vowels), the total number of underlying consonant contrasts is only possible on the first consonant of the stem and progressively decreases as one reaches the end of the stem (Hyman, 2003; Makasso and Lee, 2015).

In the context of Northwest Bantu languages, a group that displays more diversity than in the rest of the Bantu family (Bearth, 2003), Bàsàá is a rather well described language. Just like most of the languages of its family (and proba-

	Bilabial	Alveolar	Palatal	Velar	Lab. velar	Uvular	Glottal
Plosive	p	t		k	k^w g^w		
Affricate			ʧ ʤ				
Implosive	ɓ						
Prenasalized	^{m}b	$^n d$	^{n}ʤ	^{ŋ}g			
Nasal	m	n	ɲ	ŋ	$ŋ^w$		
Tap/Flap		ɾ̥ ɾ					
Fricative	ɸ β	s		x ɣ		χ	h ɦ
Approximant	w		j				
Lateral approx.		l					

Table 1: Bàsàá consonants (Makasso and Lee, 2015)

bly beyond, of Africa), most linguistic studies are however based on a handful of speakers and on elicited data. This methodology has numerous advantages and has yielded significant results. Detailed descriptions of its phonology, its morphology and its syntax are available. Entire aspects of the language however remain inaccessible: those that can be better captured through the study of natural, uncontrolled speech. With *BULBasaa*, one of our longer term aims is also to provide the basis for a reference corpus of spoken Bàsàá.

A number of publications are available in Bàsàá (bibles, dictionaries, collections of idiomatic expressions, language learning methods). Some of them use an IPA-based transcription of the language, including a representation of tones, such as the *Dictionnaire Basaá-French* (Lemb and De Gastines, 1973). As already pointed out by (Lemb and De Gastines, 1973), and this is probably still true, many publications however use alphabets that follow the transcription conventions of European languages such as French and English, and also German, for the least recent ones, with the effect that a number of contrasts are not marked and tones are not represented. Instead, accents are often used to distinguish different sounds.

3. *BULBasaa*

3.1. Data Collection

As described in (Adda et al., 2016; Stüker et al., 2016), one central aspect of BULB is to use the resource-economic methodology developed by S. Bird and colleagues, (Bird et al., 2014) which consists in (i) collecting both elicited and natural speech, (ii) proceeding to the careful re-speaking of the data by selected native speakers, to ensure the best acoustic quality possible for the purpose of automatic phone transcription, and (iii) translating the data into a major language (here, French), to speed up the documentation process and provide a basis for machine translation.

Following this method, the *BULBasaa* corpus comprises two types of Bàsàá data. One half of the recordings (uncontrolled speech) was acquired from a local Bàsàá speaking radio station in the first phase of the project. The other half of the recordings (controlled speech) was made by one of the authors (E.-M. Makasso) in various locations in the Centre and Littoral regions (Yaoundé, Douala, Eseka, Edea, Messondo) shown on Figure 2, using the LIG-Aikuma application (Gauthier et al., 2016). The data collection process took place during several missions between the summer 2015 and the summer 2017. The recording equipment used consisted of four Android powered tablets (Samsung SM-T550/T800/T810).



Figure 2: Provinces of Cameroon (Wikipedia, 2006)

LIG-Aikuma is an improved version of AIKUMA, the Android application originally developed by Steven Bird and colleagues (Bird et al., 2014), which is also meant to facilitate data collection of the type of above-described parallel speech (original>re-speaking>translation). It offers four recording modes, shown in Figure 3.

We provide an overview of the type of data collected in the next Section. Our uncontrolled speech recordings consisted of radio shows of approximately 45 minutes, including musical opening and interludes. The files thus required some cleaning before re-speaking and translation could be achieved. Additionally, to facilitate their use on an early version of LIG-Aikuma, the files were segmented into small chunks of ≈1.5 minutes before being uploaded onto the tablets.

Re-speaking of the data was mostly done by one female speaker (in her early thirties) in Berlin (Germany), where the BULB team of linguists working on Bàsàá was based, and by one male speaker (in his mid-thirties) in Yaoundé. As to the oral translation into French, a consequent part of it was also done by one male speaker (in his early forties) in Berlin and by two speakers in Yaoundé, one female (in her forties) and one male (in his early thirties). Note that although the process of orally translating natural discourse data by bilingual speakers is probably much faster than in

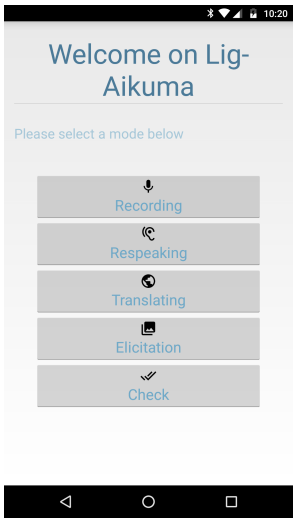


Figure 3: Screenshot LIG Aikuma (version July 2016)

more traditional methods of language documentation, it is also a cognitively demanding task for which we found it preferable to hire more than one speaker.

To have a baseline for comparison for the output of our ASR tools, approximately 10 hours of uncontrolled speech recordings were manually phonetically transcribed (against financial compensation) by a team of Bàsàá speaking linguistics students in Yaoundé.

3.2. Data Description

Table 2 provides an overview of the make-up of the *BULBasaa* corpus.

Data type	Amount
Original recordings (uncontrolled speech)	22,62 hours (1058 files)
Re-speaking (uncontrolled speech)	23,73 hours (1058 files)
Translation (uncontrolled speech)	33,55 hours (1142 files)
Picture-based elicitation	8,3 hours (2419 files)
Unprocessed bilingual controlled speech	32,10 hours (164 files)

Table 2: Overview of BULBasaa recordings

As already mentioned above, a significant part of the original Bàsàá speech data consists in radio recordings, where speakers discuss a variety of topics (every-day-life-related, local, social, cultural, historical etc). We also collected some more uncontrolled speech on the form of personal stories, tales and a traditional ceremony.

In addition, for our controlled data, we had one of our female speakers (from the Cenre region) record the entire *Dictionnaire Basaá-French* (Lemb and De Gastines, 1973), a total of 6,000 words together with their translation and example sentences to illustrate the various meanings of each word. The dictionary is a large part of the bilingual controlled speech recordings that we still need to process.

We also had a male speaker record sections of the *Enquête et description des langues orales* linguistic questionnaire (Bouquiaux and Thomas, 1976) dealing with the human being as a physical (anatomy) and as a social entity (family relations).

Finally, we used a subset of the pictures made by our colleague Guy-Noël Kouarata, a member of the BULB team working on Mboshi (Bantu C25), during one of his field trips in Republic of Congo, to elicit data parallel to the ones he collected (Godard et al., 2018).

4. Data Preparation

4.1. Pre-processing

The manual phonetic transcriptions were checked for consistency and occasional faulty symbols were removed or replaced by their correct Bàsàá counterpart. Some symbols were mapped so that, when compounded with diacritics representing tones, they correspond to existing symbols in UTF-8 encoding. The tone markers were checked as well, removing some erroneous sequences, and keeping only the 5 diacritics representing the different surface tones present in the language (see section 2). As French borrowings were transcribed using French orthography, we semi-automatically added a symbol to mark these words.

4.2. Forced Alignment

As the provided phonetic transcriptions did not contain any timing information and were not segmented based on individual recordings, we first split them to create segments for each recording. As this is a labor intensive process, we only prepared a total of 56 min / 814 speech segments of the re-spoken data in this manner. The alignment of the phones to the audio was performed using a speech recognition system. With a very limited amount of available data, we could not train an ASR system on Bàsàá data only, but instead used a system trained on multiple source languages (French, German, Italian, Russian, Turkish). The system was trained using data from the Euronews Corpus (Gretter, 2014). This corpus contains 70h of TV broadcast news per language. Trained jointly on this data, the system featured a multilingual phone set. It was adapted to Bàsàá by first manually mapping the phone set used for the phonetic transcription to the phones of the multilingual recognizer. Next, one iteration of Viterbi training was performed to adapt the acoustic model to the new language. Using this system, we force aligned the transcripts to the audio. Those alignments were used to measure against in the following experiments. As we did not have much data for adaptation, the alignments were not perfect and do not represent a gold standard.

5. Unsupervised Phone Discovery

One of the first steps in documenting a new language is to establish its phone inventory. This is a difficult and time-consuming process. We aim at supporting linguists during this step by inferring a set of phone-like units automatically. Given that unknown languages potentially feature a number

of idiosyncrasies, we want to provide an iterative approach, where linguists can make adaptions to the set of discovered phone-like units and the system is then able to incrementally derive a better set of discovered units. Our approach of unsupervised phone discovery (UPD) consists of three steps: (i) segmenting recordings into phone-like units, (ii) extracting articulatory features for each segment, and (iii) clustering segments based on the extracted features. In Sections 5.1. to 5.3., we describe each step of this pipeline. The experiments and the numbers reported were carried out using the part of the data for which the forced alignments were created.

5.1. Segmentation

A deep bi-directional long short-term memory network (DBLSTM) based approach was used to segment the recordings (Franke et al., 2016). The network was trained multilingually using the same data set as the multilingual ASR system in Section 4.2.: Data from 5 languages (English, French, German, Italian, Turkish) from the Euronews Corpus (Gretter, 2014).

5.2. Articulatory Feature Extraction

We trained detectors for 7 different types of articulatory features (AFs, see Table 5.2.) using data from French, German and Turkish from the same dataset (Müller et al., 2017). In order to create training data, we trained ASR systems for each language and used those systems to generate phonetic alignments. We then mapped the phones to AFs. As the pronunciation dictionaries were automatically built using MaryTTS (Schröder and Trouvain, 2003), we used the AF definitions embedded within MaryTTS' language definition files to establish the phone to AF mapping.

Type	# Classes	Description
cplace	8	Place of articulation
ctype	6	Type of articulation
cvox	2	Voiced
vfront	3	Tongue x position
vheight	3	Tongue y position
vlng	4	Type of vowel
vrnd	2	Lips rounded

Table 3: Overview of AF types used

The AF detectors were DBLSTM based and are trained using one-hot targets. Previous studies have shown, that using only the inner third of each segment does increase the classification performance as the articulators in the vocal tract have already reached their final position for producing the current sound and lesser co-articulation artifacts are being encountered. Figure 4 shows AFs extracted over an utterance. Co-articulation artifacts can be seen as the articulators transition from one position to the next, whereas the recognition at the center of each phone remains stable.

5.3. Clustering

As final step in our pipeline, we clustered the segments into a set of phone-like units using the extracted AFs. To determine AFs per segment, we extracted and averaged them for the inner third. A single AF vector for each segment was obtained this way. We used k-Means for clustering of the segments. This method required the number of classes as parameter. Here, we determined the class count supervised, based on the reference transcripts.

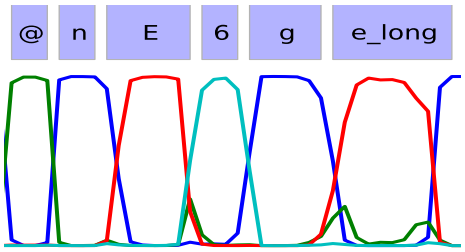


Figure 4: Example of extracted AFs

In order to evaluate the result of the clustering, we used an unsupervised evaluation method based on a TTS system (Baljekar et al., 2015). The inferred units were used to train a TTS system on the data set. This TTS system was then used to synthesize Bàsàá speech. To asses the performance of the TTS systems, the mean cepstral distortion (MCD) score (Mashimo et al., 2001) was used. It is a measure of the distortion between the synthesized and the real speech. The lower the quality of the TTS system is, the higher the distortion and the MCD score are.

The TTS was trained and tested on the full data set. Training and testing on the same data was possible, as we only want to assess the quality of the discovered units, and not the TTS system itself. As baseline, we computed the MCD score of a TTS system trained on the reference transcriptions. Table 4 shows the scores for clusterings. While the score for the automatically derived units increases, the clustered units could be used to synthesize Bàsàá speech.

# of units	MCD Score
Baseline	5.15
UPD Clustering	5.64

Table 4: Comparison of MCD Scores

6. Conclusion

We have presented the *BULBasaa* corpus, a Bàsàá-French bilingual corpus made up of both controlled (elicited) and uncontrolled (natural) speech. As Bàsàá was an under-resourced language, but its basic grammatical properties are fairly well documented, this dataset enables the evaluation of methods aimed at language documentation. From a computational linguistic perspective, our goal in the near future is to focus on improving the reference alignments, as well as to increase the performance of our setup for unsupervised phone discovery. From a linguistic perspective, one of our goals is to prepare the corpus so as to allow for the exploration of both well and under-studied grammatical aspects of the language, in particular through the study of natural speech, as well to allow for the study of aspects of

a variety of French (the variety spoken by our Bàsàá speakers) that has not received much attention so far. We plan to share these language resources through the ELDA agency after the end of the project.

7. Acknowledgments

This work was realized in the framework of the ANR-DFG project BULB (ANR-14-CE35-002). The research reported here was in part conducted at the 2017 Frederick Jelinek Memorial Summer Workshop on Speech and Language Technologies, hosted at Carnegie Mellon University and sponsored by Johns Hopkins University with unrestricted gifts from Amazon, Apple, Facebook, Google, and Microsoft. Many thanks go to the rest of the BULB@ZAS team, Carole Ngo Sohna, Chantale Ngo Ndindjock, Rodolphe Maah, Paul-Roger Bassong, and Henry Salfner, as well as to Ute Andersohn, Joerg Dreyer, Elke Dresler, Annerose Schroeder, Wolfgang Kreher and Manfred Krifka for their administrative and practical support at ZAS.

8. Bibliographical References

Adda, G., Stüker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon, F., and Zerbian, S. (2016). Breaking the unwritten language barrier: The Bulb project. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia.

Baljekar, P., Sitaram, S., Muthukumar, P. K., and Black, A. W. (2015). Using articulatory features and inferred phonological segments in zero resource speech processing. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Bearth, T., (2003). *The Bantu languages*, chapter Syntax, pages 121–142. Routledge.

Bird, S., Hanke, F. R., Adams, O., and Lee, H. (2014). Aikuma: A mobile app for collaborative language documentation. *ACL 2014*, page 1.

Luc Bouquiaux et al., editors. (1976). *Enquête et description des langues à tradition orale*. SELAF, Paris.

Franke, J., Müller, M., Hamlaoui, F., Stüker, S., and Waibel, A. (2016). Phoneme Boundary Detection using Deep Bidirectional LSTMs. In *Speech Communication; 12. ITG Symposium; Proceedings of*. VDE.

Gauthier, E., Blachon, D., Besacier, L., Kouarata, G.-N., Adda-Decker, M., Rialland, A., Adda, G., and Bachman, G. (2016). LIG-AIKUMA: a Mobile App to Collect Parallel Speech for Under-Resourced Language Studies. In *Interspeech 2016 (short demo paper)*, San-Francisco, France, September.

Godard, P., Adda, G., Adda-Decker, M., Benjumea, J., Besacier, L., Cooper-Leavitt, J., Kouarata, G.-N., Lamel, L., Maynard, H., Müller, M., Rialland, A., Stüker, S., Yvon, F., and Zanon-Boito, M. (2018). A very low resource language speech corpus for computational language documentation experiments. In *LREC 2018 (in press)*, Japan.

Gretter, R. (2014). Euronews: A Multilingual Benchmark for ASR and LID. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Guthrie, M. (1948). *The classification of the Bantu languages*. Oxford University Press for the International African Institute.

Hyman, L., (2003). *The Bantu languages*, chapter Bàsàa (A43), pages 257–282. Routledge.

Lemb, P. and De Gastines, F. (1973). *Dictionnaire basaá-français*. Collège Libermann.

Makasso, E.-M. and Lee, S. J. (2015). Basaá. *Journal of the International Phonetic Association*, 45(1):71–79.

Mashimo, M., Toda, T., Shikano, K., and Campbell, N. (2001). Evaluation of cross-language voice conversion based on gmm and straight.

Müller, M., Franke, J., Stüker, S., and Waibel, A. (2017). Improving Phoneme Set Discovery for Documenting Unwritten Languages. *Elektronische Sprachsignalverarbeitung (ESSV) 2017*.

Schröder, M. and Trouvain, J. (2003). The German text-to-speech synthesis system MARY: A tool for research, development and teaching. *International Journal of Speech Technology*, 6(4):365–377.

Gary F. Simons et al., editors. (2017). *Ethnologue: Languages of the World, Twentieth edition.* SIL International, Dallas, Texas. Online version: `http://www.ethnologue.com`.

Stüker, S., Adda, G., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon, F., and Zerbian, S. (2016). Innovative technologies for under-resourced language documentation: The Bulb project. In *Proceedings of CCURL (Collaboration and Computing for Under-Resourced Languages : toward an Alliance for Digital Language Diversity)*, Portorož Slovenia.

Wikipedia. (2006). Departments of cameroon — Wikipedia, the free encyclopedia. [Online; accessed 28-September-2017].

Researching Less-Resourced Languages – the DigiSami Corpus

Kristiina Jokinen
University of Helsinki, Finland and AIRC, AIST Tokyo Waterfront, Japan
kristiina.jokinen@helsinki.fi

Abstract

Increased use of digital devices and data repositories has enabled a digital revolution in data collection and language research, and has also led to important activities supporting speech and language technology research for less-resourced languages. This paper describes the DigiSami project and its research results, focussing on spoken corpus collection and speech technology for the Fenno-Ugric language North Sami. The paper also discusses multifaceted questions on ethics and privacy related to data collection for less-resourced languages and indigenous communities.

Keywords: corpus collection, under-resourced languages, North Sami

1. Introduction

Several projects and events have increased research activities for under-resourced languages during the past years. For instance, the DLDP-project (Digital Language Diversity Project) is to advance the sustainability of Europe's regional and minority languages, while the Flare-net network and the LRE Map (Calzolari et al. 2012) have had a big impact on sharing language resources and making speech corpora freely available. However, there are many challenges that researchers and developers face when aiming at the same technology standards for less-resourced and endangered languages as those for majority languages. The challenges do not only concern scarce data and non-optimal algorithms, but also issues inherently related to the cultural contexts of linguistic communities in general, and shared background of minority cultures in particular. It is thus important to pay attention to community-based techniques in data collection and technology development (crowdsourcing in a wide sense) as well as trying to connect the communities with other small language communities (Soria et al. 2013). An important aspect of such work is to empower minority language speakers with the knowledge and skills to create and share content for digital devices using their own language.

The DigiSami project (http://www.helsinki.fi/digisami/) is supported by the Academy of Finland in a wider context of a joint initiative with the Hungarian Academy of Sciences. The focus of the Digital Citizens framework project was to support collaborative research on endangered Fenno-Ugric Languages and to develop tools and resources for automatic language processing as well as to experiment with new technology applications. The main motivation was to improve digital visibility and viability of the target languages, and to explore different choices for encouraging and maintaining the use of less-resourced languages in the digitalized world. The goals of the DigiSami project are discussed in Jokinen (2014) and Jokinen et al. (2017).



Figure 1. The Sami language areas at the beginning of the 20th century. Abbreviations: So - South Sami, Um - Ume Sami, Pi - Pite Sami, Lu - Lule Sami, No - North Sami, In - Inari Sami, Sk - Skolt Sami, Ak - Akkala Sami (extinct), Ki - Kildin Sami, Tr - Ter Sami. From Jokinen et al. (2017), original source Sammallahti (1998).

The DigiSami project deals with the North Sami language (Davvisámegiela) which belongs to the Fenno-Ugric language family and is one of the nine Sami languages spoken in the northern part of Europe: Scandinavia, Finland and the Kola Peninsula in Russia (Seurujärvi-Kari et al. 1997). Figure 1 shows the Sami languages and their geographical distribution. There are about 40 000 speakers of the Sami languages, and of these about half speak North Sami, which has gained the status of a lingua franca. North Sami also enjoys official status in Norway and Finland, and has language technology tools so it is possible to develop online content analysis and interactive applications (see more in Jokinen et al. 2017).

2. The DigiSami Corpus

The project organised data collection in three locations in Finland and two locations in Norway, see Figure 2. The locations were selected to include representative locations of the variations of North Sami spoken in the Sápmi area: the border between the countries of Norway and Finland divides the area in two, and there is also a major language division between the Eastern and Western dialects of North Sami, so that the Sami language spoken in Kautokeino and Enontekiö belong to Western North Sami, while the others belong to Eastern North Sami. In Inari, Inari Sami is spoken as a separate language (Sammallahti 1998). See Jokinen (2014) and Jokinen and Wilcock (2014) for more about data collection.

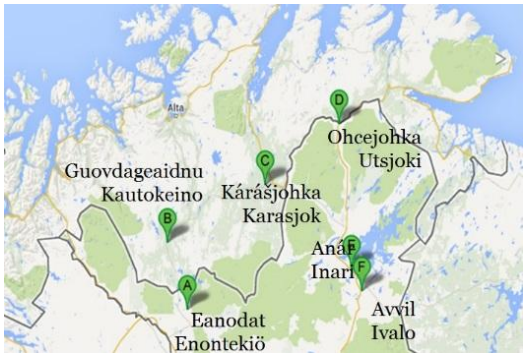


Figure 2. The data collection locations in Norway (Kautokeino, Karasjok) and in Finland (Ivalo, Inari, Utsjoki). No data was collected in Enontekiö although originally intended.

The DigiSami corpus contains two types of spoken data: video data on conversational spontaneous speech, and read speech from participants reading out Wikipedia articles. Participants were bilingual (North Sami and either Finnish or Norwegian), and had lived most of their life in the Sápmi area. Although local dialects in the participants' speech cannot be tracked due to the small number of participants, the corpus contains notable pronunciation variation between Norwegian and Finnish participants. This has been further studied by Jokinen et al. (2016a) using automatic recognition methods for speaker and dialect identification.

The dialogue corpus has been transcribed using Praat, and annotated following the MUMIN guidelines set out in Allwood et al. (2007) and Paggio et al. (2010). Multimodal annotation concerns especially laughing, body movements and topics for multimodal conversation studies.

Basic statistics of the corpus are given in Table 1, and demographic facts in Figure 3.

	Read speech		Conversational speech	
Dialects	#Spkr	Duration (hrs)	#Spkr	Duration (hrs)
Kautokeino	4	1.03	-	-
Karasjoki	7	0.72	6(1)	1.5
Ivalo	7	0.72	7(1)	0.72
Utsjoki	6	1.07	6(1)	1.03
Inari	4	0.73	-	-
Total	**28**	**3.36**	**19**	**4.28**

Table 1: Basic Statistics of the DigiSami Corpus

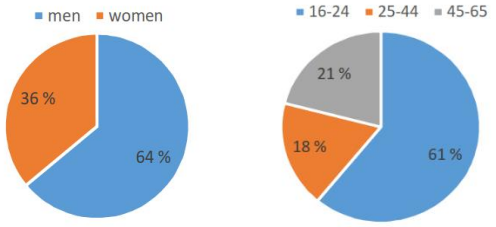


Figure 5 The participant's gender (left) and age (right).

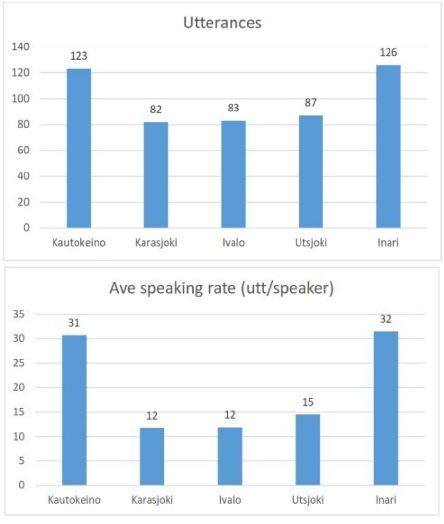


Figure 6. The number of utterances and the average speaking rate in the DigiSami corpus for each of the five locations. The Kautokeino and Inari numbers refer to read speech, while the Karasjok, Ivalo, and Utsjoki show conversational data.

The conversational corpora are comparable in that the speakers seem to talk at a similar rate and produce a similar number of utterances. Conversations take place among young adults at schools, and apparently reflect similarities among the young Sami people in Norway (Karasjok) and Finland (Ivalo and Utsjoki).

From the questionnaire that the participants filled in before the recordings, we can also find out that most participants used North Sami as the main communication means when interacting with other people, a third of the participants spoke North Sami when communicating with relatives on either mother's or father's side only, and only three (11%) of the participants used North Sami at official places and at work, but not at home (Figure 3). Communication context for speaking North Sami was predominantly at home, but also at work (or school), and to a lesser degree at public places like shops, offices, and restaurants (Figure 4). The percentages in this figure do not add up to 100%, since the participants could mark as many alternative locations as they wanted, and the categories are not mutually exclusive either. More descriptive data analysis can be found in Jokinen (2014).

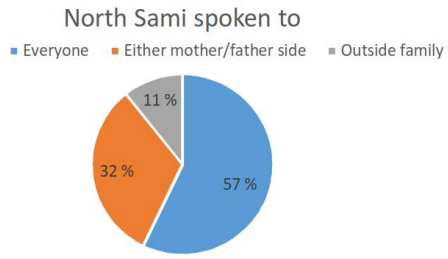


Figure 3 Reported interlocutors when speaking North Sami.

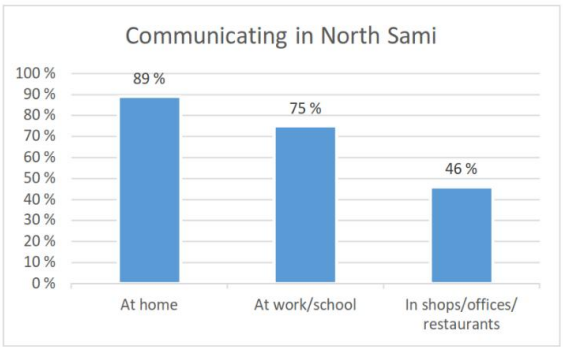


Figure 4 Communication contexts when speaking North Sami among the participants.

The corpus has been widely used in various studies on spoken interaction. The pronunciation differences are investigated in Jokinen et al. (2016a), while Grönroos et al. (2016) present morphological segmentation for the North Sami language using the active learning method. Trong et al. (2018) use an end-to-end dialect recognition system based on the deep learning method and discuss its use as enabling technology for building interactive applications. Hiovain and Jokinen (2016) discuss laughter types and Trong et al. (forthcoming) study relation between laughter, topics and body movement, and also compare the corpus with related corpora in Finnish and Estonian.

The papers are accessible through the project website, while the DigiSami corpus is available via the CSC website by contacting the author.

3. SamiTalk

The DigiSami project also aims at technology applications that would make the North Sami more visible in the digital world and also boost the North Sami status as a prestige language, usable for new digital applications. The project thus designed SamiTalk, a human-robot interactive application for accessing North Sami Wikipedia (Wilcock et al. 2017). SamiTalk is based on the WikiTalk and MoroTalk applications (Jokinen and Wilcock 2013, Wilcock 2012) which allow the users to search for interesting information from Wikipedia and interactively chat with the system. Although more work is needed to develop the Sami speech technology components further, it is expected that in the near future SamiTalk type applications can be effectively used for tasks such as interactive language learning and preserving cultural heritage via storytelling, besides the common information retrieval or question-answering tasks. However, these developments will also need effort and active participation from the Sami people to proceed in a co-design manner towards applications that also respect the fragile cultural context (see Section 4).

During the data collection events, also a small number of North Sami Wikipedia articles was produced to encourage the community to develop North Sami Wikipedia further. Wikipedia statistics (http://stats.wikimedia.org/) tells that the North Sami Wikipedia started in 2004, and ranks in the middle of all Wikipedias with over 7500 articles. Currently only North Sami has a Wikipedia, but during the data collection sessions, interest was also sparkled within Inari Sami community to start a Wikipedia of their own. The number of North Sami page requests is about half a million page requests per month, but it is not possible to know how many human readers there are. In general, however, an accelerating circle can be noted: the more readers, the more editors who create more content, and the more content, the more readers. As for the content, North Sami Wikipedia is 134th of about 300 in number of articles. Articles are fairly short, about 500 characters long on average, but as wikis grow, the average article length grows as well. However, it there are many articles about towns around the world, since such articles can be easily created following a regular pattern with the town specific numbers filled from a database. Some editors automate the creation of articles but no statistics exists about how many articles have been created by translating them from some other Wikipedia.

The aim in DigiSami was not only to create more articles for SamiTalk, but to encourage development of a new public space that indigenous people could use to make their own voice heard through the information that the people themselves have created for a wider audience. Wikipedia has a well-suited format for documenting indigenous people's life and important events, places and people, since it is a form of public and non-commercial information technology. It is collectively produced and jointly developed and can thus also help to make the language and culture more visible by creating interest in the topics that the indigenous writers choose. Since much of the language technology research and common information retrieval technology uses Wikipedia as a resource, extending the existing North Sami Wikipedia is not only useful for the North Sami speakers, but strengthens the North Sami presence in digital world.

4. Ethical Aspects

The DigiSami corpus follows the standard rules, principles, and guidelines for ethical and privacy issues in data collection provided by Finland's National Advisory Board on Research Ethics. Each participant, or the parent of an underage participant, signed a consent form to take part in the corpus collection, and the research papers on data collection were sent to the local coordinators. The names, ages, and other demographic information that would give away personal identification of the participants are not present in the transcriptions, and research examples are chosen so that they do not identify individual speakers (see discussion of the guidelines in Jokinen 2011). Local participants were also invited to the SamiTalk demo at the IWSDS 2016 conference held in Saariselkä in Lapland (http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=47125&copyownerid=66338), and the book edited on the basis of the presentations also acknowledges the DigiSami project and its goals (Jokinen and Wilcock 2017).

Moreover, the DigiSami project has paid attention to the questions concerning data gathering from less-resourced languages in general. Confidentiality, privacy and respect are relevant issues in data collection, and issues on ethics and privacy are particularly important when collecting data from indigenous communities. These are multifaceted questions related to sensitive issues of the rights and ownership of the indigenous people, and a major concern is the people's rights and ownership of their own cultural heritage which should be respected. Since the culture and traditions are carried by the language, there is always an inherent element of cultural knowledge in the topics and the use of the language, and although corpus collection can aim at a neutral goal of documenting a less-resourced language, in practice it may not be possible to collect a culturally "neutral" corpus. Thus attention should be paid to the culture and tradition. The following questions are relevant in data collection in general, with special emphasis in the context of less-resourced and indigenous languages:

- What kind of data is gathered and what kind of activities are included? Written and spoken discourse already available publicly in books, TV radio, and films, is of different nature than recordings of private, spontaneous, and everyday behaviours, or open data available in internet and social media channels. However, as discussed by Oskal (2008), social and cultural research is not an individualistic process, but includes interaction with community members who produce knowledge in collaboration with the other members.
- Who owns the data? The rights to use the data can vary according to the copyright legislation, but there are also sensitive and complex issues related to Civil and Political Rights. An overview of the issues in relation to Sami languages can be found in Kokko (2010).
- What purpose is the data used for? The collection of data may be used for studying characteristics of an individual, a group, or a culture, in order to preserve their characteristics or to better understand their activities. The question also concerns how to define the appropriate ways to use the data, who can use it and who has the right to access the data.
- What kind of personal information is included? Text, speech, and video include different degrees of personal information and thus identification of the individuals

who produced the data can vary (Jokinen 2011). However, given present-day techniques of data processing, individual characteristics may be retrieved easily. On the other hand, distinction between the categories of personal vs collective may not appear so clear-cut: if the individuals' experience of their own activity is seen as an inherent part of the group activity, it equates with a collective view of the community's life and culture. The question of privacy thus does not only concern a consent from an individual but from a whole community, as the activity transfers from a single participant to the whole community of which the individual is a member. An important question is what kind of impact the corpus has on the participants' life.

Concerning the Sami culture and Sami languages, these issues have been actively investigated by existing cultural and social institutions in the area. For instance, Sámi Parliament, Sámi Museum Siida, Giellagas Institute and the Saami Culture Archive at the University of Oulu, and the Law School of University of Lapland, are among the main actors studying legal issues and ethical questions concerning the Sami culture and heritage.

5. Conclusions

The DigiSami corpus is available and has already been used in various research activities (see Section 3). Future steps could include data and knowledge collection by the people themselves and documenting this in Wikipedia articles or in other common digital archive formats. This could be organised in community halls as part of community activities, or in schools as part of practising Sami language writing in mother tongue classes. An important part of the activities is that they should arise from the language community and from the speakers' willingness and concern to work on the language themselves, rather than being seen as an outside activity that meets resistance. As mentioned above, applications such as SamiTalk have potential to be used in language teaching as well as assisting story-telling for culture preservation. Examples of using technology to preserve the culture already exists (e.g. Rodil 2014), and a great example is the way game apps have been used successfully in Australian aboriginal languages (Aboriginal Australian language video game).

More information is also needed about the specific needs for digital information and applications that the indigenous users may find useful. In spring 2016 the DLDP consortium conducted a survey and gathered information about the personal digital use of the language and about any known digital resource and services that make use of the language.

Improving viability in the digital world as well as revitalization of the language use is more likely to succeed if there are local people actively involved in the process. Collection of corpora for further language technology studies requires that certain standards for the development for language technology tools and applications are taken into account. The DigiSami project has taken steps in this direction. It is hoped that the DigiSami corpus will prove useful as the first systematic collection of multimodal North Sami conversation for research in speech and interaction technology, as well as for cultural studies, and that it can also support revitalisation and digital viability of the Sami languages and Sami culture in general.

6. Acknowledgements

The work has been carried out in the Academy of Finland project Fenno-Ugric Digital Citizens (grant n°270082). The author would like to thank all the project workers, students, participants, teachers, and colleagues who have been involved in the project, assisted and contributed to the work in various stages of the project.

7. Bibliographical References

Allwood, J., Cerrato, L., Jokinen, K., Navarretta, C. and Paggio, P. (2007). The MUMIN Coding Scheme for the Annotation of Feedback, Turn Management and Sequencing. In Martin et al. (eds) *Multimodal Corpora for Modelling Human Multimodal Behaviour*.

Boersma, P. and D. Weenink (2009). Praat: doing phonetics by computer (version 5.1.05). Retrieved May 1, 2009, from http://www.praat.org/

Calzolari N., Del Gratta R., Fracopoulo G., Mariani J., Rubino F., Russo I. and Soria C. The LRE Map. Harmonising Community Description of Resources. *Proceedings of LREC 2012*, Istanbul, Turkey, pp. 1084—1089, 2012.

Grönroos, S.A., Hiovain, K., Smit, P., Rauhala, I., Jokinen, K., Kurimo, M., and Virpioja, S. (2016). Low-Resource Active Learning of Morphological Segmentation. Northern European Journal of Language Technology, Vol. 4, pp 47–72, DOI 10.3384/nejlt.2000-1533.1644

Grünthal, R., Siegl, F.: Uralilaisten kielten pensasmalli ja arvioidut puhujamäärät (The "bush model" of the Uralic languages and the estimated numbers of speakers) (2012). Department of Finno-Ugric Studies, University of Helsinki.

Hiovain, K. and Jokinen, K. (2016). Acoustic Features of Different Types of Laughter in North Sami Conversational Speech. LREC Workshop Just talking – casual talk among humans and machines, Portorož.

Jokinen, K. (2011) Multimodal Information - Collection and Analysis of Interactive Data. HCII 2011. Orlando, U.S.

Jokinen, K. (2014). Open-domain interaction and online content in the Sami language. In: Proceedings of Ninth International Conference on Language Resources and Evaluation (LREC 2014). Reykjavik.

Jokinen, K., Hiovain, K., Laxström, N., Rauhala, I., Wilcock, G. (2017). DigiSami and Digital Natives: Interaction Technology for the North Sami Language. In: Jokinen, K. and Wilcock, G. (Eds.). *Dialogues with Social Robots – Enablements, Analyses, and Evaluation.* pp. 3-19. Springer. http://www.springer.com/gb/book/9789811025846 DOI: 10.1007/978-981-10-2585-3.

Jokinen, K., Trong, T.N., Hautamäki, V. (2016). Variation in Spoken North Sami Language, Interspeech.

Jokinen, K. and Wilcock, G. (2017). *Dialogues with Social Robots – Enablements, Analyses, and Evaluation*. Lecture Notes in Electrical Engineering, Vol 427, Springer, DOI: 0.1007/978-981-10-2585-3

Jokinen, K. and Wilcock, G. (2014). Community-Based Resource Building and Data Collection. The 4th International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU'14). St.Petersburg, Russia.

Jokinen, K. and Wilcock, G. (2013). Multimodal Open-domain Conversations with the Nao Robot. In: Mariani,

J., Devillers, L., Garnier-Rizet, M. and Rosset, S. (eds.) *Natural Interaction with Robots, Knowbots and Smartphones - Putting Spoken Dialog Systems into Practice*. Springer Science+Business Media

Kokko, K.T. (Ed) (2010). Kysymyksiä saamelaisten oikeusasemasta. (”Issues Concerning the Legal Status of the Sámi people”) Lapin yliopiston oikeustieteellisiä julkaisuja Sarja B no 30 (University of Lapland Publications Series B Number 30), Rovaniemi. http://urn.fi/URN:NBN:fi:ula-201201161003

Oskal, N. (2008). The question of methodology in indigenous research. A philosophical exposition. In H. Minde, S. Jentoft, H. Gaski & G. Midré (eds.) *Indigenous peoples: self-determination, knowledge, indigeneity*. Eburon: Delft, pp. 331–345.

Paggio, P., J. Allwood, E. Ahlsén, K. Jokinen, C. Navarretta (2010). The NOMCO multimodal Nordic resource - goals and characteristics. In *Proceedings of LREC 2010*, 2968-2973.

Prószéky, G. (2011) Endangered Uralic Languages and Language Technologies. Proceedings of Language Technologies for Digital Humanities and Cultural Heritage Workshop, pages 1–2, Hissar, Bulgaria, 16 September 2011.

Rodil, K. (2014). A Participatory Perspective on Cross-Cultural Design. In Ebert, A., van der Veer G.C., Domik, G., Gershon, N.D., Scheler, I. (eds.). *Building Bridges: HCI, Visualization, and Non-formal Modeling*, Springer Lecture Notes in Computer Science, pp. 30-46.

Sammallahti, P. (1998). *The Saami Languages: An Introduction*. Davvi Girji, Kárášjohka.

Seurujärvi-Kari, I., Pedersen, S. and Hirvonen, V. (1997). The Sámi. The indigenous people of northernmost Europe. *European Languages 5*. Brussels: European Bureau for Lesser Used Languages.

Soria, C., Mariani, J. and Zoli, C. (2013). Dwarfs sitting on the giants' shoulders – how LTs for regional and minority languages can benefit from piggybacking major languages Proceeding of XVII FEL Conference 10/2013, Ottawa.

Trong, T.N., Jokinen, K., Hautamäki, V. (2018). Enabling Spoken Dialogue systems for low-resource languages – End-to-end dialect recognition for North Sami. Proceedings of the IWSDS 2018, Singapore.

Wilcock, G. (2012). WikiTalk: a Spoken Wikipedia-based Open-domain Knowledge Access System. Proceedings of the COLING-2012 Workshop on Question Answering for Complex Domains, Mumbai, pp. 57-69.

Wilcock, G., Laxström, N., Leinonen, J., Smit, P., Kurimo, M., Jokinen, K. (2017). Towards SamiTalk: a Sami-speaking Robot linked to Sami Wikipedia. In: Jokinen, K. and Wilcock, G. (Eds.). *Dialogues with Social Robots – Enablements, Analyses, and Evaluation*. pp. 343-351. Springer. http://www.springer.com/gb/book/9789811025846 DOI: 10.1007/978-981-10-2585-3.

8. Language Resource References

Aboriginal Australian language video game. An interview of the developers. The DLDP Newsletter #1 - December 2016. http://mpegmedia.abc.net.au/radio/local_canberra/audio/201609/abn-2016-09-27-video-game-offers-new-life-to-ancient-indigenous-language.mp3 [Online; accessed 25-September-2017]

DigiSami Project (2017). Academy of Finland project http://www.helsinki.fi/digisami/

DigiSami Corpus (2017). Available from the project leader, and accessible through Center for Scientific Computing. www.csc.fi

Digital Language Diversity Project. Erasmus+ Programme Grant Agreement No. 2015-1-IT02-KA204-015090 http://www.dldp.eu/ [Online; accessed 25-September-2017]

ESFRI. European Strategy Forum on Research Infrastructures. http://ec.europa.eu/research/infrastructures/index_en.cfm?pg=esfri [Online; accessed 25-September-2017]

Giellagas Institute. (2015). The Saami Culture Archive of University of Oulu. http://www.oulu.fi/giellagasinstitute/the_saami_culture_archive. [Online; accessed 25-September-2017]

Simple4All Consortium. (2015). Simple4All: developing automatic speech synthesis technology http://simple4all.org/ [Online; accessed 25-September-2017]

The MADAR Arabic Dialect Corpus and Lexicon

Houda Bouamor, Nizar Habash,[†] Mohammad Salameh, Wajdi Zaghouani,[♣] Owen Rambow,[‡] Dana Abdulrahim,[*] Ossama Obeid,[†] Salam Khalifa,[†] Fadhl Eryani,[†] Alexander Erdmann,[†] Kemal Oflazer

Carnegie Mellon University in Qatar, Qatar, [♣]Hamad Bin Khalifa University, Qatar
[†]New York University Abu Dhabi, UAE [‡]Columbia University, USA, [*]University of Bahrain, Bahrain
hbouamor@cmu.edu, nizar.habash@nyu.edu, wzaghouano@hbku.edu.qa,
rambow@cs.columbia.edu, darahim@uob.edu.bh

Abstract

In this paper, we present two resources that were created as part of the Multi Arabic Dialect Applications and Resources (MADAR) project. The first is a large parallel corpus of 25 Arabic city dialects in the travel domain. The second is a lexicon of 1,045 concepts with an average of 45 words from 25 cities per concept. These resources are the first of their kind in terms of the breadth of their coverage and the fine location granularity. The focus on cities, as opposed to regions in studying Arabic dialects, opens new avenues to many areas of research from dialectology to dialect identification and machine translation.

Keywords: Arabic Dialects, Parallel Corpus, Lexicon

1. Introduction

Dialectal Arabic (DA) is emerging nowadays as the primary written language of informal communication online in the Arab World: in emails, blogs, discussion forums, chats, SMS, etc. There has been a rising interest in research on computational models of Arabic dialects in the last decade (Meftouh et al., 2015). There have been several efforts on creating different resources to allow building models for several Natural Language Processing (NLP) applications. However, these efforts have been disjoint from each other, and most of them have focused on a small number of dialects that represent vast regions of the Arab World (Zaidan and Callison-Burch, 2011; Diab et al., 2014; Sajjad et al., 2013).

In this paper, we present two resources we created as part of the Multi Arabic Dialect Applications and Resources (MADAR) project.[1] The goal of MADAR is to create, for a large number of dialects, a unified framework with common annotation guidelines and decisions, and targeting applications of Dialect Identification (DID) and Machine Translation (MT).

The first resource is a large parallel corpus of 25 Arabic city dialects, in addition to the pre-existing parallel set for English, French and Modern Standard Arabic (MSA). The second resource is a 25-way lexicon of 1,045 entries in each city's dialect along with MSA, French, and English. These resources are the first of their kind in terms of the breadth of their coverage and their fine granularity. The kind of resources we present in this paper are useful not only for building computational systems but also for studying Arabic dialects from a linguistics perspective (e.g., computational dialectology).

[1]The project site is at http://nlp.qatar.cmu.edu/madar/. مَدَار *madAr* means 'orbit' in Arabic.

2. Arabic and its Dialects

The Arabic language is a family of varieties. Among these varieties, Modern Standard Arabic (MSA) is the shared language of culture, media, and education from Morocco to Oman. However, MSA is not the native language of any speaker of Arabic. In unscripted situations where spoken MSA would typically be required (such as talk shows on TV), speakers usually resort to repeated code-switching between their dialects and MSA (Abu-Melhim, 1991; Bassiouney, 2009). Arabic dialects are often classified regionally (as Egyptian, North African, Levantine, Gulf, Yemeni) (Habash, 2010) or sub-regionally (e.g., Tunisian, Algerian, Lebanese, Syrian, Jordanian, Kuwaiti, Qatari). DA varies phonologically, lexically, and morphologically from MSA, from region to region and to a lesser extent, from city to city in each region (Watson, 2007). In the rest of this section, we discuss these differences.

Phonology An example of phonological differences is in the pronunciation of dialectal words whose MSA cognate has the letter Qaf (ق *q*).[2] It is often observed that in Tunisian Arabic, this consonant appears as /q/ (similar to MSA), while in Egyptian and Levantine Arabic it is /ʔ/ (glottal stop) and in Gulf Arabic it is /g/ (Haeri, 1991; Habash, 2010).

Orthography While MSA has a standard orthography, the dialects do not. Often people write words reflecting the phonology or the etymology of these words. DA is sometimes written in the so-called Arabizi Romanization script (Darwish, 2014). In the context of NLP, a set of conventional orthography guidelines (CODA) has been proposed for a number of dialects (Habash et al., 2012a; Jar-

[2]Arabic transliteration is presented in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007): (in alphabetical order) *AbtθjHxdðrzsšSDTḎς*γ*fqklmnhwy* and the additional symbols: ' ء, Â أ, Ǎ إ, Ā آ, ŵ ؤ, ŷ ئ, ħ ة, ý ى.

Region	Maghreb				Nile Basin	Levant		Gulf		Yemen
Sub-region	**Morocco**	**Algeria**	**Tunisia**	**Libya**	**Egypt/Sudan**	**South Levant**	**North Levant**	**Iraq**	**Gulf**	**Yemen**
Cities	**Rabat** (*RAB*) Fes (*FES*)	Algiers (*ALG*)	**Tunis** (*TUN*) Sfax (*SFX*)	Tripoli (*TRI*) Benghazi (*BEN*)	**Cairo** (*CAI*) Alexandria (*ALX*) Aswan (*ASW*) Khartoum (*KHA*)	Jerusalem (*JER*) Amman (*AMM*) Salt (*SAL*)	**Beirut** (*BEI*) Damascus (*DAM*) Aleppo (*ALE*)	Mosul (*MOS*) Baghdad (*BAG*) Basra (*BAS*)	**Doha** (*DOH*) Muscat (*MUS*) Riyadh (*RIY*) Jeddah (*JED*)	Sana'a (*SAN*)

Table 1: Different region, sub-region, and city dialects considered in building the MADAR resources.

rar et al., 2014; Zribi et al., 2014; Saadane and Habash, 2015; Turki et al., 2016; Khalifa et al., 2016), and has been recently unified under the CODA* effort (Habash et al., 2018).

Morphology Morphological differences are quite common. One example is the future marker particle which appears as +س *s+* or سوف *swf* in MSA, +ح*H*+ or رح *rH* in Levantine dialects, +ه *h+* in Egyptian and باش *bAš* in Tunisian. This together with variation in the templatic morphology make the forms of some verbs rather different: e.g., 'I will write' is سأكتب *sÂktb* (MSA), حأكتب *HÂktub* (Palestinian), هكتب *hktb* (Egyptian) and باش نكتب *bAš nktb* (Tunisian).

Syntax Comparative studies of several Arabic dialects suggest that the syntactic differences between the dialects are relatively minor compared to morphological differences (Brustad, 2000). For example, negation may be realized differently using a combination of prefixes and suffixes (ما *mA*, مش *mish*, مو *muw*, لا *lA*, لم *lam*, etc.) but its syntactic distribution is to a large extent uniform across varieties (Benmamoun, 2012).

Lexicon The number of lexical differences among dialects is significant. The following are a few examples (Habash et al., 2012a): Egyptian بس *bas* 'only' and طربيزة *tarabayzaħ* 'table' correspond to MSA فقط *faqaT* and طاولة *TAwilaħ*, respectively. For comparison, the Levantine forms of the above words are بس *bas* and طاولة *TAwliħ*.

These differences pose serious challenges for Arabic NLP. The challenges are mainly related to the lack of resources and tools. The tools developed for MSA or for a specific dialect cannot effectively model DA which makes its direct use for handling dialects impractical (Habash et al., 2012b).

3. The MADAR Corpus

We built the MADAR Corpus, the first collection of parallel sentences covering the dialects of 25 cities from the Arab World, in addition to English, French, and MSA.[3] Table 1 shows the break up we follow in choosing these cities. This table relates the typical five-way regional break up of Arabic dialects (Habash, 2010) to a more refined ten-way sub-region division, and even further into 25 cities.

The corpus is created by translating selected sentences from the Basic Traveling Expression Corpus (BTEC) (Takezawa et al., 2007) in French and English to the different dialects.[4] BTEC is a multilingual spoken language corpus containing tourism-related sentences similar to those that are usually found in phrasebooks for tourists going abroad. This corpus is an attractive resource to use for different reasons: (i) it is conversational in nature (including questions and answers by tourists/guides) and this makes it closer to the genre dialects are used for primarily; (ii) it has short sentences (on the average 6.5 words), which makes it easy enough for the translators to translate; and (iii) the BTEC corpus has translations in several languages which allows the possibility to use this data in the future for training/testing machine translation models across these languages and Arabic dialects.

We selected 2,000 BTEC sentences and translated them to all 25 city dialects (each of these sentences has 25 corresponding parallel translations). Henceforth, we refer to this part of the corpus as CORPUS-25. Furthermore, we selected 10,000 additional sentences and translated them to the dialects of five selected cities: Doha, Beirut, Cairo, Tunis, and Rabat. We call this corpus CORPUS-5. Effectively, each of the five selected cities has 12,000 sentences that are five-way parallel translations, and that could be used to build several Dialectal Arabic NLP applications such as machine translation. An example of a 28-way parallel sentences extracted from CORPUS-25 is given in Figure 1.[5]

Translators, identified from each of the 25 cities specifically, were asked to read a set of sentences provided in English or French, and translate them into their dialects. The translators are all native speakers of the dialects of the cities they hail from. We did not choose MSA as a starting point to avoid biasing the translation (Bouamor et al., 2014). [6]

[3]The MADAR corpus will be made available to the research community. The English part will not be distributed due to copyright restriction. It can be acquired directly from the USTAR consortium (http://www.ustar-consortium.com/).

[4]The English, French and MSA versions we use are those provided in the IWSLT evaluation campaign (Eck and Hori, 2005).

[5]The MADAR Corpus is available for browsing online at http://nlp.qatar.cmu.edu/madar/.

[6]The translation was handled by Ramitechs (http://www.ramitechs.com/), a company that creates and annotates several types of corpora and lexicons using expert linguists.

English	This room is too small.
French	Cette chambre est trop petite.
MSA	هذه الغرفة صغيرة جدا . *hðh Alγrfħ Sγyrħ jdA .*
Beirut	هالأوضة كتير زغيرة . *hAlÂwDħ ktyr zγyrħ.*
Cairo	الأوضة دي صغيرة أوي . *AlÂwDħ dy Sγyrħ Âwy .*
Doha	هالغرفة واجد صغيرة . *hAlγrfħ wAjd Sγyrħ .*
Rabat	هاد الغرفة صغيرة بزاف . *hAd Alγrfħ Sγyrħ bzAf .*
Tunis	لبيت هذي صغيرة برشا. *lbyt hðy Sγyrħ baršA.*
Aleppo	هالغرفة كتير صغيرة . *hAlγrfħ ktyr Sγyrħ .*
Alexandria	الأوضة دى صغيرة جدا . *AlÂwDħ dý Sγyrħ jdA .*
Algiers	هاذ الغرفة صغيرة بزاف . *hAð Alγrfħ Sγyrħ bzAf .*
Amman	هاي الغرفة كتير صغيرة . *hAy Alγrfħ ktyr Sγyrħ .*
Aswan	الأوضة دي صغيرة خالص . *AlÂwDħ dy Sγyrħ xAlS .*
Baghdad	هاي الغرفة كولش صغيره . *hAy Alγrfħ kwlš Sγyrh .*
Basra	هاي الغرفة كلش صغيرة . *hAy Alγrfħ klš Sγyrħ.*
Benghazi	الدار صغيرة بكل . *AldAr Sγyrħ bkl.*
Damascus	هالغرفة صغيرة كتير . *hAlγrfħ Sγyrħ ktyr .*
Fes	هاد الغرفة صغيرة بزاف . *hAd Alγrfħ Sγyrħ bzAf.*
Jeddah	الغرفة دي مرا صغيرة . *Alγrfħ dy mrA Sγyrħ .*
Jerusalem	هاي الغرفة كتير صغيرة . *hAy Alγrfħ ktyr Sγyrħ .*
Khartoum	الغرفة دي صغيرة شديد . *Alγrfħ dy Sγyrħ šdyd .*
Mosul	الغرفه كلش صغيغي . *Alγrfh klš Sγyγy .*
Muscat	هالحجره وايد صغيرة . *hAlHjrh wAyd Sγyrħ .*
Riyadh	الغرفة صغيرة جدا . *Alγrfħ Sγyrħ jdA .*
Salt	هاي الغرفة كثير صغيرة . *hAy Alγrfħ kθyr Sγyrħ .*
Sanaa	الغرفة صغيرة قوي . *Alγrfħ Sγyrħ qwy.*
Sfax	البيت هذه ياسر صغيرة . *Albyt hðh yAsr Sγyrħ.*
Tripoli	الدار هادي صغيره هلبه . *AldAr hAdy Sγyrh hlbh .*

Figure 1: A sample of a 28-way parallel sentence extracted from CORPUS-25 including 5 sentences from CORPUS-5. The MSA and dialectal sentences are given along with their transliterations in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007).

3.1. Translation Guidelines

We provided the translators with the source sentences (out of context) and asked them to produce a translation that precisely reflects the source sentence without any assumption. We provided the translators with the following set of detailed guidelines:

- The translators were asked to use Arabic script, avoid any code-switching and to be internally consistent in spelling words. We did not provide them with any orthographic guidelines.
- Punctuation marks (such as periods, commas and question marks) that appear in the source sentence should remain in the Arabic dialect translation.
- Numbers written in letters should be translated into letters, while numbers written in digits should be kept as digits. For example, the translation of "six" is ستة *stħ*, while the translation of "6" is ٦.
- The translation of idioms should not be literal but reflect the meaning of the idioms instead.
- In the case where the gender (masculine vs. feminine) is not obvious in a source sentence, the masculine form should be used. For example, the English word *student* should be translated into Egyptian as طالب *TAlb* (masculine form in Arabic) not طالبة *TAlbħ* (feminine word in Arabic), unless the sentence contains a feminine form.[7] When the number (singular vs. plural) is not obvious in a sentence, the singular form should be used. For example, the English word *you* should be translated into Egyptian as انت *Ânt* not انتوا *ÂntwA*, as long as the sentence does not specify the plural form.
- Foreign words borrowed from English or French should be transliterated. For example, the French word *ordinateur (computer)* is commonly used in Tunisian Arabic, so it might be transliterated as اوردیناتور *Awrdynatwr*. If the word has an equivalent in MSA, that is widely used in a certain dialect; this word should be translated into its MSA alternative. For example, the English word *program* should be translated as برنامج *brnAmj* not transliterated to بروجرام *brwjrAm*.

3.2. Corpus Analysis

The example in Figure 1 highlights the many lexical and morphological differences among the dialects of different cities. For example, the MSA word الغرفة *Alγrfħ* 'the room' was translated into البيت *Albyt* in Sfax, الأوضة *AlÂwDħ* in the Cairo, Alexandria and Aswan dialects and الدار *AldAr* in Tripoli and Benghazi dialects. While it was translated into the MSA-like form in other city dialects. This example shows the difference between various dialects, commonly treated as one big class of dialects such as Algerian, Moroccan (الغرفة *Alγrfh)*, Tunisian (البيت *Albyt*) and Lybian (الدار *AldAr*).

In order to get an overall estimation of the similarity between the dialects of the cities covered in CORPUS-25 (in addition to MSA), we compute the Overlap Coefficient, representing the percentage of lexical overlap between the vocabularies for each dialect pair. The average pairwise similarity between the dialects in our dataset is 25.8% with

[7] We follow the choice made in producing the BTEC MSA version.

a standard deviation of 8.5% when MSA is included. When MSA is not included, the average similarity between the dialects is 26.3%. The most similar pair of dialects are those spoken in the cities of Amman and Jerusalem with an overlap of 54.4%. The least similar dialects are those of Sfax and Alexandria with a difference of 87.4%. The closest city dialect to MSA is Muscat dialect with an overlap score of 37.5%, and the most dissimilar one is the dialect of Sfax (lexical difference of 88.12%).

Overall, the lexical overlap between the dialects in our dataset is lower than the one reported in Bouamor et al. (2014). In the latter, the authors report high similarity scores between each dialect and Egyptian Arabic. This is explained by the fact that the translations were initially obtained from Egyptian which biased the lexical choices of the translators. This result justifies our decision to not use MSA as a starting point when building the MADAR corpus.

4. The MADAR Lexicon

In this section we present the structure of the MADAR lexicon and we describe the automatic and manual steps we followed in creating it.

4.1. Lexicon Structure

The MADAR lexicon is organized around *concept keys* which are defined in terms of triplets of words from English (En), French (Fr) and MSA. The multilingual triplets are intended to reduce ambiguity that comes from different senses of a particular word. For example, the English noun 'table' has a *furniture* sense and a *set of data* sense. But these two senses correspond to different MSA words طاولة *TAwlħ* and جدول *jdwl*, respectively. The latter of the MSA terms has other senses also, such as 'brook'. We plan to use these multilingual triplets to link to established large resources such as Wordnet (Fellbaum, 1998; Bentivogli et al., 2002) or Babelnet (Navigli and Ponzetto, 2012).

Each concept has a number of words associated with it. Each word is defined in terms of three aspects: its CODA orthography, its CAMEL Arabic Phonetic Inventory (CAPHI) phonology (Habash et al., 2018) and the various cities in which it is used. DA orthographic variations make it difficult for computational models to properly identify and reason about the words of a given dialect (Habash et al., 2012a). Hence, a conventional form for the orthographic notations is important to reduce sparsity and ambiguity. CODA is a set of guidelines and exception lists for Egyptian Arabic. Several efforts have extended them to cover other dialects (Jarrar et al., 2014; Zribi et al., 2014; Saadane and Habash, 2015; Turki et al., 2016; Khalifa et al., 2016). However, they focused on specific dialects and often made ad hoc decisions. In a recent effort, Habash et al. (2018) introduced a more unified set of guidelines and resources for DA orthography. They presented a common set of guidelines with enough specificity to help in creating dialect specific conventions as needed and applied them to the dialects of 25 Arab cities. In this work, we use these guidelines to build a CODAfied version of the MADAR Lexicon. [8]

Inspired by the International Phonetic Alphabet (IPA) and Arpabet (Shoup, 1980), CAPHI provides a system for transcribing all dialects of Arabic in a simple and user-friendly fashion, while still maintaining enough complexity to describe all meaningful phonological variation.[9] Figure 2 presents the full entry in the MADAR lexicon of the concept (very, très, جدّا *jdA*). The lexicon includes a small number of multi-word expressions, such as the Arabic multi-word expression representing the 'passport' concept in (passport, passeport, جواز سفر *jawAz safar*).

CODA	CAPHI	City Dialect
برشا *bršA*	b a r sh a	Tunis, Sfax
بزاف *bzAf*	b e z z aa f	Rabat, Fez, Algiers
بكل *bkl*	b i k k i l	Benghazi
جدا *jdA*	g i d d a n	Cairo, Alexandria
جدا *jdA*	j i d d a n	Jeddah, Khartoum, Riyadh
خالص *xAlS*	kh aa l i s.	Cairo, Alexandria, Aswan
شديد *šdyd*	sh a d ii d	Khartoum
قوي *qwy*	2 a w i	Cairo, Alexandria
قوي *qwy*	g a w i	Aswan, Sana'a
كثير *kθyr*	k i t ii r	Alexandria, Cairo
كثير *kθyr*	k t ii r	Beirut, Jerusalem, Damascus, Aleppo, Amman, Fez, Rabat
كثير *kθyr*	k th ii r	Amman, Salt
كثير *kθyr*	k th ii gh	Mosul
كثير *kθyr*	k a t ii r	Jeddah, Aswan, Khartoum
كثير *kθyr*	k i th ii r	Riyadh, Muscat
كلش *klš*	k u l l i sh	Basra, Baghadad
كلش *klš*	k e l l i sh	Mosul, Doha
مرة *mrħ*	m a r r a	Jeddah
هلبة *hlbħ*	h a l b a	Tripoli
عوم *ʕwm*	3 oo m	Muscat
هوايا *hwAyA*	h w aa y a	Basra, Baghadad
واجد *wAjd*	w aa y i d	Basra, Baghadad, Doha
واجد *wAjd*	w aa j i d	Benghazi, Tripoli, Doha
واجد *wAjd*	w aa g i d	Muscat

Figure 2: MADAR Lexicon entries for concept (very, très, جدّا *jdA*).

Besides, each *concept key* is represented in a lemma and phrasal form. The lemma form is supplemented with its part-of-speech tag (POS). For Arabic, the *POS* is provided for the segmented form of the word on a clitical level. The phrasal form is a frequently used inflected form of the concept. For example, the concept of 'thanks' has a lemma form of *(thanks_NOUN merci_NOUN, šukr_NOUN* شكر*)*, while the phrasal form represents the Arabic word in its frequently used form of *šukrAã* شكرا. Also, the Arabic lemma form of the 'zoo' concept is *Hadiyqaħ_NOUN Al+_DET HayawAn_NOUN* حديقة ال + حيوان, while its phrasal

[8] A detailed description of CODA guidelines are available at: http:\resources.camel-lab.com.

[9] The complete CAPHI inventory is available at: http:\resources.camel-lab.com.

form is *Hadiyqaħ_NOUN Al+_DET HayawAnAt_NOUN* حديقة ال + حيوانات.

The MADAR lexicon contains a total of 1,045 concepts, which cover 88.0%, 86.4% and 85.5% of the lemma tokens in the English, French and MSA BTEC corpora respectively. Almost three-quarters of the concepts are for open classes.

4.2. Lexicon Concept Identification

Concept key identification relies on an automatic process that extracts *(English, French, Arabic)* related tuples from the BTEC parallel corpus. Tuples are then clustered based on their semantic similarity, such that each cluster represents a concept. The automatic process is followed by manual validation and fixing of errors resulting from the automatic process.

4.2.1. Automatic Extraction of Concept Keys

Data Preprocessing Since the concept triplet words are represented in terms of lemmas, we pre-process the parallel data to map it into the lemma space. For English, we use the Stanford POS tagger (Toutanova et al., 2003) and for French, we use Treetagger (Schmid, 1994). For Arabic, we use MADAMIRA (Pasha et al., 2014) to tokenize words into the D3 scheme, which separates all clitics from the basewords. Arabic tokenization is required as the clitics attached to basewords in Arabic, are typically represented as separate words in English and French. The most common examples are the proclitic definite article +ال *Al+* 'the', and the enclitic possessive pronouns, such as ه+ *+h* 'his'. The goal here is to harmonize the forms of the three languages to encourage better word alignment and concept extraction.

Triplet Extraction Our trilingual concept extraction approach focuses on collecting frequently used triplets. We align French-English, English-MSA, MSA-French pairs with GIZA++(Och, 2002) using the intersection symmetrization heuristic. Each word in an English sentence is aligned to words in the corresponding French and MSA sentences.

We address the triplet extraction problem as a task of collecting connected components from an undirected graph. Given three parallel English, French and MSA sentences, we represent words as nodes and alignments as edges in the graph. Nodes in an extracted component have to belong to the three languages strictly. Connected components are collected from all sentence pairs in the parallel aligned sentences, and each unique triplet is provided with a count representing the number of times the triplet is extracted from all the different parallel sentences. The output of the extraction method is a set of triplets sorted by their count. In Figure 3, we show an example of an aligned English, French and MSA parallel sentence.

Among the eight extracted connected components, four components constitute the triplets spanning the three languages: *(ce, the,* ال *Al+) , (acteur, actor,* ممثّل *mumaθil), (vraiment, really,* فعل *fiçl)* and *(merveilleux, marvelous,* رائع *rAŷiç).*

Concept Extraction Since several extracted triplets share some semantic similarity, we need to group the triplets into clusters such that each cluster represents a shared concept among the triplets. For example, the triplets *(bag, sac,* حقيبة *Haqiybaħ), (bag, sac,* كيس *kiys)* and *(bag, baggage,* كيس *kiys)* represent the concept of a "bag" in the three languages. The concept can be represented by the triplet with the highest frequency. Our approach models this problem as a breadth-first traversal of an undirected graph where each triplet represents a node in the graph. An edge connects two triplets if they share two words from any of the three languages. For instance, we draw an edge between *(bag, sac,* حقيبة *Haqiybaħ)* and *(bag, sac,* كيس *kiys)* in Figure 4 since they share the English "bag" and the French "sac" constituents of the triplet.

We sort all triplets based on their frequency and apply a breadth-first traversal with a maximum depth of two, starting with the most frequent triplet. We iteratively repeat the breadth-first traversal starting with the next most frequent unvisited node, until all nodes are visited. The visited nodes in each traversal will constitute the cluster of a concept represented by the most frequent triplet.

Traversal with a depth of two (with respect to the starting node) was chosen empirically, as deeper levels showed some divergence from the main concept encompassed by triplets in the first two levels. In the undirected graph of Figure 4, we start with the highest frequency triplet (bag, sac, حقيبة *Haqiybaħ*) with a count of 134, and reach all neighboring triplets until a depth of two (shown in the left dotted square). The next most frequent triplet is (baggage, bagage, متاع *matAç*) with frequency of 102. We end up with two clusters representing the concepts of (bag, sac, حقيبة *Haqiybaħ*) and (baggage, bagage, متاع *matAç*).

4.2.2. Manual Validation of Concept Keys

The initial manual effort in building the lexicon involved carefully checking all the extracted concepts, correcting some cases and adding some missing entries. We identified four types of errors in the automatic lexicon construction approach we described above. First are preprocessing errors, mostly in the form of incorrect lemmatization. For example, يوقع *ywqç* 'to sign' was incorrectly lemmatized as وقع *waqaç* 'to fall' instead of وقّع *waq∼aç*. Second are missing alignment errors resulting from inherent linguistic differences. One example is the pronoun *I/je*, which is sometimes conjugated in Arabic as a verbal suffix. Since we use lemmas, the conjugated verbs are turned into their lemma form and that information is lost.

Third are multi-word expression (MWE) alignment errors. Since our approach did not address MWEs specifically, we had many cases of incomplete concept keys. For example, the English term 'really' in Figure 3 is incorrectly aligned to the Arabic term فعل *fiçl* 'act', while the correct align-

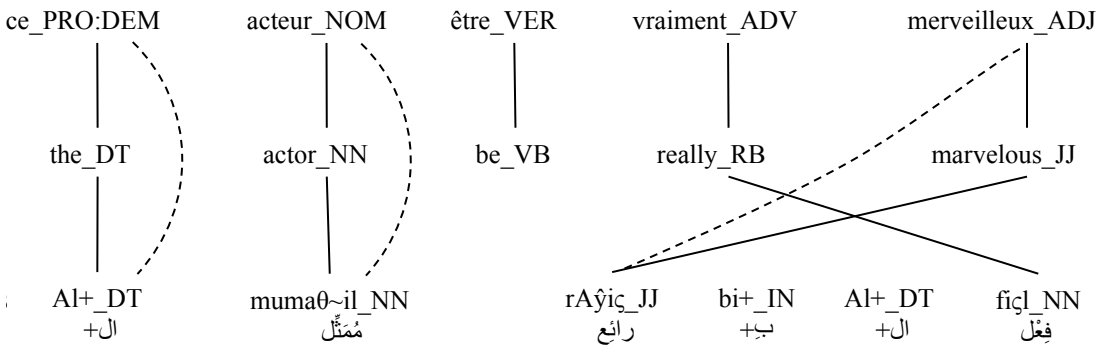


Figure 3: Alignment between French, English and MSA parallel sentences respectively. The non lemmatized forms of the three parallel sentences are: English: *The actor is really marvelous*; French: *Cet acteur est vraiment merveilleux*; MSA: *Almumaθ~il rAŷiç bAlfiçl* المثل رائع بالفعل

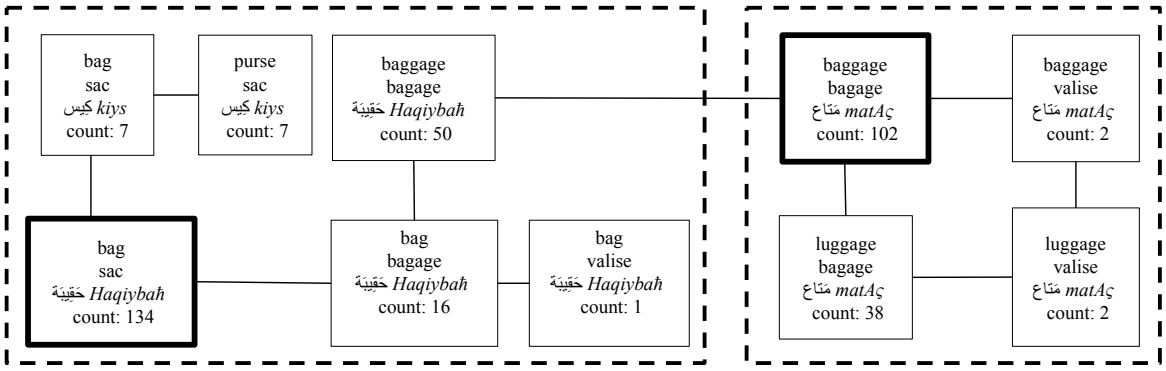


Figure 4: Concept extraction from aligned triplets. Each square represents a triplet with its English, French and Arabic terms and its count. Extracted concepts are indicated by the dotted square.

ment should be the MWE ب+ال+ فعل *b+ Al+ fiEl*. Fourth are errors from very ambiguous words such as the word قد *qd* which means both *may be* and *certainly* depending on the aspect of the verb that follows it. This particular case ended up not included as the head of any single cluster despite being very frequent. Since we could not manually fix all the clusters, we targeted the top 1,000 or so clusters ranked by cluster-head frequency and recovered additional high-frequency words that were not properly identified automatically.

4.3. Lexicon Population

The lexicon population with dialectal entries proceeded in two steps. First, we automatically inserted entries extracted from a number of existing dictionaries; and then we manually validated, and extended them.

4.3.1. Automatic Lexicon Population

We transcribed a number of dialectal dictionaries: (i) The Karmous dictionary for Tunisian Arabic (Abdelatif, 2010), including around 3,800 words and several expressions and proverbs in Tunisian; (ii) the Moroccan Arabic Dialect textbook (Morocco, 2011), written by a team of language instructors who shared their collective experience gained by training thousands of Americans who lived and worked in Morocco. We also use the Tharwa lexicon (Diab et al., 2014), a four-way large-scale lexicon for dialectal Arabic, covering Egyptian and Levantine in addition to MSA and English; and the Iraqi dictionary from the LDC (Graff and Maamouri, 2009).

We attempted to populate our lexicon with as many entries by pivoting on English or French. These entries were not always in CODA-compliant form or had phonological representations that we could easily convert to CAPHI. We tried our best in this step to create CODA and CAPHI forms that are easy to edit and extend in the manual annotation step.

4.3.2. Manual Lexicon Population

The automatic lexicon population is followed by a large annotation effort, which involved 13 linguists who are from different regions of the Arab World. The lexicon is presented in a Google Sheet where every concept and its associated dialectal word forms are listed as shown in Figure 5.

There are two sections for every concept: The first section (marked in yellow cells) specifies the concept definition. The second section (marked in green cells) specifies the various dialect words. The concept definition consists of six columns including the concept ID (Concept_ID), its category, and in addition to the French, English and MSA

Concept_ID	Category	English	French	Standard Arabic	En-POS	Fr-POS	Ar-POS
139	Concept	car	voiture	سَيَّارة	NOUN	NOUN	NOUN
#==========							
Concept_ID	**Category**	**Dialect**	**Arabic CODA**	**CAPHI**	**Comments/Questions**		
139	AUTO	EGY	عربية	3 a r a b i y y a			
139	AUTO	LEV, IRQ, YEM	سيارة	s a y y aa r a			
139	AUTO	TUN	كرهبة	k r h b a			
139	ADD						

Figure 5: An example of an automatically populated concept, as presented to the linguists.

lemmas triplets, their corresponding POS tags (Fr-POS, En-POS, and Ar-POS). The dialectal word list consists of five columns including an identifier of the content of the row, and a category. The category could be: (a) **AUTO** for a word proposed by the automatic lexicon population described in 4.3.1.. It is not validated by a human, or (b) **ADD**: is an open slot provided to allow editors to add entries without inserting a new row. These values must be changed to **VALID**.

The column **Dialect** specifies the dialect of the entry. One or more region or city codes are provided per entry. The region code is provided instead of the city one for entries for which we do not have a city dictionary. For instance, the entry سيارة *syArħ* in Figure 5 was extracted from dictionaries covering these regional dialects Levantine (LEV), Iraqi (IRQ) and Yemeni (YEM). The linguists were asked to update this column with the corresponding specific city code. The code of each city is given in Table 1.

The linguists were provided with detailed guidelines on the steps to follow when editing and populating the lexicon. Each linguist was asked to:

- Read the concept definition carefully, clarify in his/her mind the exact meaning (this includes being aware of the full meaning and sub-meanings), and use the different translations and POS to help with this task.
- Scan the various AUTO entries provided for all regions. This might help him remember words that are possible candidates to add for the cities he/she is responsible for.
- Delete all entries that are NOT relevant to the cities he/she is responsible for.
- Apply the necessary changes for some entries that may need some minor fixes.
- Add new words that are not on the AUTO list.
- Think of more than one translation into his/her dialect and carefully specify the city.
- Use external informants to get more information for cities in his/her area if it is not his original city.
- Enter the CODA and CAPHI versions of each entry, using the guidelines provided.
- Make sure the Arabic CODA and CAPHI are correct for all the entries for their cities.
- Add the code names of the cities he/she is responsible for.
- Change the category to VALID once a row is fully validated.

Weekly meetings by the project PIs and a consulting lead linguist reviewed the progress of the linguists. At the time of writing this paper, the MADAR lexicon contained 47,466 dialectal words (average 1,899 per dialect). The average number of words per concept per dialect is 1.8. We are continuously working on quality checking, expanding and improving the coverage of the lexicon. [10]

5. Related Work

In the context of work on NLP, MSA has received the bulk of attention. There are lots of parallel and monolingual data collections, richly annotated collections (e.g., treebanks), sophisticated tools for morphological analysis and disambiguation, syntactic parsing, etc. (Habash, 2010). Even for languages other than Arabic, the integration of dialectal variation in NLP applications is rather rare. One interesting exception is the work of Scherrer (2012) on Swiss German dialects.

Very recently, automatic DA processing has attracted a considerable amount of research in NLP (Shoufan and Alameri, 2015), facilitated by the newly developed monolingual and multilingual dialectal corpora and lexicons. Several mono-dialectal corpora covering different Arabic dialects were built and made available. Al-Badrashiny et al. (2014) compiled a large dialect-identified corpus of DA from several Egyptian sources, but with a large presence of MSA. In a related effort, McNeil and Faiza (2011) built a four-million-word corpus of Tunisian Spoken Arabic. Various other research work resulted in multidialectal non parallel corpora at different scales (Zaidan and Callison-Burch, 2011; Zbib et al., 2012; Cotterell and Callison-Burch, 2014; Salama et al., 2014; Jeblee et al., 2014; Al-Shargi et al., 2016; Zaghouani and Charfi, 2018).

[10]The latest version of the lexicon is available for browsing online at `http://nlp.qatar.cmu.edu/madar/`.

As for dialect-to-dialect parallel corpora, Bouamor et al. (2014) presented the first small-scale 7-way parallel corpus covering several dialects in addition to MSA, and English, all translated from Egyptian sentences. The fact that Egyptian was chosen as a starting point affected the quality of the translation. The sentences produced were biased by the use of some Egyptian expressions that might be accepted in other dialects, but a native would not produce naturally. The same concern applies to the 6-way parallel PADIC corpus used in Meftouh et al. (2015), as all translations were derived from DA or MSA. When developing CORPUS-5 and CORPUS-25, we avoided such priming effects by asking translators to produce translations starting from English or French based on their preferences. However, most of these efforts focus primarily on a number of varieties corresponding generally to those spoken in major cities (Cairo, Amman, Baghdad, etc.), or study different dialects independently.

Unlike MSA, DA has a small number of printed bilingual or monolingual dictionaries. Thus, building a DA lexicon with varying degrees of coverage and linguistic complexity has been the aim of several research efforts. The LDC built the Egyptian Colloquial Arabic Lexicon (ECAL) (Kilany et al., 2002) and the Iraqi Arabic Morphological Lexicon (IAML) (Graff and Maamouri, 2009), two mono-dialectal lexica, that were used in developing the Egyptian and Iraqi versions of the CALIMA morphological analyzer (Habash et al., 2012b).

A notable multi-dialectal lexicon is the one built in the *Arabic Variant Identification Aid (AVIA)* project. This lexicon covers the seven Arabic dialects spoken in the following cities: Al-Ain (United Arab Emirates), Baghdad (Iraq), Jeddah (Saudi Arabia), Jerusalem (Palestinian Arabic), Kuwait City, Doha (Qatar) and Sana'a (Yemen). Another significant effort is Tharwa (Diab et al., 2014), a 4-way English, MSA, Egyptian, Levantine lexicon with rich linguistic annotation. Our lexicon is similar to Tharwa in that we also use CODA compliant lemma forms. However, the MADAR lexicon includes phonetic modeling via the CAPHI representation. Also, our lexicon covers more regional and city dialects (25 city dialects) compared to Tharwa (two dialects only). *The Dialects of Arabic* project at the University of Manchester recently made publicly available a database of Arabic dialects that include a mix of words and sentences in their phonological forms covering samples from 15 countries in the Arab World (Matras and others, 2017).

To the best of our knowledge, our work is the first effort aiming at building large scale and fine-grained dialectal Arabic resources (corpora and lexicon) mapped to their English, French and MSA versions.

6. Conclusion and Future Work

We presented two resources: the MADAR Corpus and MADAR Lexicon. The first is a large scale parallel corpus created by translating selected sentences in the travel domain into 25 Arabic city dialects. The second is a lexicon of 1,045 entries covering the same 25 Arabic cities. These resources are the first of their kind in terms of the breadth of coverage and fine granularity.

In the future, we plan to extend both resources in terms of number of cities. We also plan to expand the lexicon with more entries. The MADAR Corpus and Lexicon will be used to create three enabling technologies and applications that are necessary to support future research in Arabic NLP: dialect identification, machine translation and morphological analysis.

Acknowledgements

We would like to thank our dedicated linguists who contributed in building the MADAR lexicon: Linda Alamir, Feryal Albrehi, Shumool Albuainain, Gazella Ben Sreiti, Jamila El-Gizuli, Dihia Gareche, Fatma Ghailan, Anissa Jrad, Reham Marzouk, Mohammad Abuodah, Salim Al-Mandhari and Aous Mansouri. We also would like to thank *Ramitechs* for the translation services, and the UStar Consortium for providing us with the English version of BTEC.

This publication was made possible by grant NPRP 7-290-1-047 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

7. Bibliographical References

Abdelatif, K. (2010). *Dictionnaire le Karmous du Tunisien*.

Abu-Melhim, A.-R. (1991). Code-switching and Linguistic Accommodation in Arabic. In *Perspectives on Arabic Linguistics III: Papers from the Third Annual Symposium on Arabic Linguistics*, volume 80, pages 231–250. John Benjamins Publishing.

Al-Badrashiny, M., Eskander, R., Habash, N., and Rambow, O. (2014). Automatic transliteration of romanized Dialectal Arabic. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 30–38, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Al-Shargi, F., Kaplan, A., Eskander, R., Habash, N., and Rambow, O. (2016). Morphologically annotated corpora and morphological analyzers for moroccan and sanaani yemeni arabic. In *10th Language Resources and Evaluation Conference (LREC 2016)*.

Bassiouney, R. (2009). *Arabic Sociolinguistics*. Edinburgh University Press.

Benmamoun, E. (2012). Agreement and Cliticization in Arabic Varieties from Diachronic and Synchronic Perspectives. In Reem Bassiouney, editor, *Al'Arabiyya: Journal of American Association of Teachers of Arabic*, volume 44-45, pages 137–150. Georgetown University Press.

Bentivogli, L., Pianta, E., and Girardi, C. (2002). MultiWordNet: developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*, Mysore, India, January.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A Multidialectal Parallel Corpus of Arabic. In *Proceedings*

of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pages 1240–1245, Reykjavik, Iceland.

Brustad, K. (2000). *The Syntax of Spoken Arabic: A Comparative Study of Moroccan, Egyptian, Syrian, and Kuwaiti Dialects*. Georgetown University Press.

Cotterell, R. and Callison-Burch, C. (2014). A Multi-Dialect, Multi-Genre Corpus of Informal Written Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 241–245.

Darwish, K. (2014). Arabizi Detection and Conversion to Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 217–224, Doha, Qatar.

Diab, M. T., Al-Badrashiny, M., Aminian, M., Attia, M., Elfardy, H., Habash, N., Hawwari, A., Salloum, W., Dasigi, P., and Eskander, R. (2014). Tharwa: A Large Scale Dialectal Arabic-Standard Arabic-English Lexicon. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3782–3789, Reykjavik, Iceland.

Eck, M. and Hori, C. (2005). Overview of the IWSLT 2005 Evaluation Campaign. In *International Workshop on Spoken Language Translation (IWSLT) 2005*.

Fellbaum, C. (1998). Wordnet: An electronic lexical database. *The MIT Press*.

Graff, D. and Maamouri, M. (2009). Iraqi Arabic Morphological Lexicon (IAML) Version 6.5. Linguistic Data Consortium.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic transliteration. In Abdelhadi Soudi, et al., editors, *Arabic Computational Morphology*, volume 38 of *Text, Speech and Language Technology*, chapter 2, pages 15–22. Springer.

Habash, N., Diab, M., and Rambow, O. (2012a). Conventional Orthography for Dialectal Arabic. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 711–718, Istanbul, Turkey.

Habash, N., Eskander, R., and Hawwari, A. (2012b). A Morphological Analyzer for Egyptian Arabic. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pages 1–9, Montréal, Canada.

Habash, N., Khalifa, S., Eryani, F., Rambow, O., Abdulrahim, D., Erdmann, A., Faraj, R., Zaghouani, W., Bouamor, H., Zalmout, N., Hassan, S., shargi, F. A., Alkhereyf, S., Abdulkareem, B., Eskander, R., Salameh, M., and Saddiki, H. (2018). Unified Guidelines and Resources for Arabic Dialect Orthography. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Habash, N. Y. (2010). *Introduction to Arabic natural language processing*, volume 3. Morgan & Claypool Publishers.

Haeri, N. (1991). Sociolinguistic Variation in Cairene Arabic: Palatalization and the qaf in the Speech of Men and Women.

Jarrar, M., Habash, N., Akra, D., and Zalmout, N. (2014). Building a corpus for Palestinian Arabic: a preliminary study. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 18–27.

Jeblee, S., Feely, W., Bouamor, H., Lavie, A., Habash, N., and Oflazer, K. (2014). Domain and Dialect Adaptation for Machine Translation into Egyptian Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 196–206, Doha, Qatar.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016). A Large Scale Corpus of Gulf Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Kilany, H., Gadalla, H., Arram, H., Yacoub, A., El-Habashi, A., and McLemore, C. (2002). Egyptian Colloquial Arabic Lexicon.

Matras, Y. et al. (2017). Database of Arabic Dialects. The University of Manchester.

McNeil, K. and Faiza, M. (2011). Tunisian Arabic corpus: Creating a written corpus of an unwritten language. In *Workshop on Arabic Corpus Linguistics (WACL)*.

Meftouh, K., Harrat, S., Jamoussi, S., Abbas, M., and Smaili, K. (2015). Machine translation experiments on PADIC: A parallel Arabic dialect corpus. In *The Proceedings of the 29th Pacific Asia conference on Language, Information and Computation*.

Morocco, P. C. (2011). *Moroccan Arabic*.

Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Och, F. (2002). *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. Ph.D. thesis, RWTH Aachen, Germany.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Saadane, H. and Habash, N. (2015). A Conventional Orthography for Algerian Arabic. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, page 69, Beijing, China.

Sajjad, H., Darwish, K., and Belinkov, Y. (2013). Translating Dialectal Arabic to English. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–6, Sofia, Bulgaria.

Salama, A., Bouamor, H., Mohit, B., and Oflazer, K. (2014). YouDACC: the Youtube Dialectal Arabic Comment Corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1246–1251, Reykjavik, Iceland.

Scherrer, Y. (2012). *Generating Swiss German Sentences from Standard German: a Multi-dialectal Approach*. Ph.D. thesis, University of Geneva, Switzerland.

Schmid, H. (1994). Probabilistic part-of-speech tagging

using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

Shoufan, A. and Alameri, S. (2015). Natural Language Processing for Dialectical Arabic: A Survey. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 36–48, Beijing, China.

Shoup, J. E. (1980). Phonological Aspects of Speech Recognition. *Trends in speech recognition*, pages 125–138.

Takezawa, T., Kikui, G., Mizushima, M., and Sumita, E. (2007). Multilingual Spoken Language Corpus Development for Communication Research. *Computational Linguistics and Chinese Language Processing*, 12(3):303–324.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180, Edmonton, Canada.

Turki, H., Adel, E., Daouda, T., and Regragui, N. (2016). A Conventional Orthography for Maghrebi Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portoroz, Slovenia.

Watson, J. C. (2007). *The Phonology and Morphology of Arabic*. Oxford University Press.

Zaghouani, W. and Charfi, A. (2018). ArapTweet: A Large Multi-Dialect Twitter Corpus for Gender, Age and Language Variety Identification. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic Online Commentary Dataset: an Annotated Dataset of Informal Arabic with High Dialectal Content. In *Proceedings of ACL 2011*, Portland, Oregon, USA.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O. F., and Callison-Burch, C. (2012). Machine translation of Arabic dialects. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 49–59. Association for Computational Linguistics.

Zribi, I., Boujelbane, R., Masmoudi, A., Ellouze Khmekhem, M., Hadrich Belguith, L., and Habash, N. (2014). A Conventional Orthography for Tunisian Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, Reykjavik, Iceland.

Designing a Collaborative Process to Create Bilingual Dictionaries of Indonesian Ethnic Languages

[1]Arbi Haza Nasution, [2]Yohei Murakami, [3]Toru Ishida
[1,3]Department of Social Informatics, Kyoto University, [2]Unit of Design, Kyoto University
Kyoto, Japan
[1]arbi@ai.soc.i.kyoto-u.ac.jp, [2]yohei@i.kyoto-u.ac.jp, [3]ishida@i.kyoto-u.ac.jp

Abstract

The constraint-based approach has been proven useful for inducing bilingual dictionary for closely-related low-resource languages. When we want to create multiple bilingual dictionaries linking several languages, we need to consider manual creation by a native speaker if there are no available machine-readable dictionaries are available as input. To overcome the difficulty in planning the creation of bilingual dictionaries, the consideration of various methods and costs, plan optimization is essential. Utilizing both constraint-based approach and plan optimizer, we design a collaborative process for creating 10 bilingual dictionaries from every combination of 5 languages, i.e., Indonesian, Malay, Minangkabau, Javanese, and Sundanese. We further design an online collaborative dictionary generation to bridge spatial gap between native speakers. We define a heuristic plan that only utilizes manual investment by the native speaker to evaluate our optimal plan with total cost as an evaluation metric. The optimal plan outperformed the heuristic plan with a 63.3% cost reduction.

Keywords: Bilingual Dictionary Creation, Low-resource Languages, Closely-related Languages

1. Introduction

Nowadays, machine-readable bilingual dictionaries are being utilized in actual services (Ishida, 2011) to support intercultural collaboration (Ishida, 2016; Nasution et al., 2017b), but low-resource languages lack such sources. Obviously bilingual lexicon extraction is highly problematic for low-resource languages due to the paucity or outright omission of parallel and comparable corpora. We introduced the promising approach of treating pivot-based bilingual dictionary induction for low-resource languages as an optimization problem (Nasution et al., 2016; Nasution et al., 2017c) where bilingual dictionaries are the only language resource required. Despite the high potential of our approach in enriching low-resource languages, it faces numerous issues when trying to create plans to implement multiple bilingual dictionaries for a set of low-resource languages like Indonesian ethnic languages. When actually implementing our constraint-based bilingual dictionary induction approach, we need to consider the inclusion of more traditional methods like manually creating the bilingual dictionaries by native speaker. In spite of the high cost, this will be unavoidable if no machine-readable dictionaries are available. Given the various methods and costs that may need to be considered, we recently introduced a plan optimizer to find the feasible optimal plan of creating multiple bilingual dictionaries with the least total cost (Nasution et al., 2017a). In this project, to create bilingual dictionary D_{A-B} between ethnic language L_A and ethnic language L_B, there is also a difficulty in finding a bilingual native speaker of two ethnic languages. To overcome this limitation, we can firstly create triple T_{A-ID-B} using the common language, Indonesian as pivot language L_{ID} where S_{ID-A}, a native bilingual speaker of Indonesian language L_{ID} - ethnic language L_A and S_{ID-B}, a native bilingual speaker of Indonesian language L_{ID} - ethnic language L_B collaborate by explaining the senses with Indonesian language. Then, the bilingual dictionary D_{A-B} can be induced from the triple T_{A-ID-B}. The native speakers need a tool that can bridge the spatial gap and help them collaborate. To actually implement our pivot-based bilingual dictionary induction following the optimal plan to create multiple Indonesian ethnic languages bilingual dictionaries, we address the following research goals:

- *Designing a Collaborative Process for Creating Bilingual Dictionaries of Indonesian Ethnic Languages*: Implementing plan optimization for creating bilingual dictionaries of low-resource languages and implementing a generalized constraint approach to bilingual dictionary induction for low-resource language families in creating 10 bilingual dictionaries with 2,000 translation pairs from every combination of 5 languages, i.e., Indonesian, Malay, Minangkabau, Javanese, and Sundanese.
- *Designing an Online Collaborative Dictionary Generation*: Bridging spatial gap between native speakers especially when doing a collaborative creation or evaluation of bilingual dictionary.

The rest of this paper is organized as follows: In Section 2 and Section 3, we will briefly discuss our constraint-based bilingual dictionary induction and plan optimizer, respectively. Section 4 details our collaborative process design. Finally, Section 5 concludes this paper.

2. Constraint-Based Bilingual Dictionary Induction

The traditional pivot-based approach is very suitable for low-resource languages (Tanaka and Umemura, 1994). Unfortunately, for some low-resource languages, it is often difficult to find machine-readable inverse dictionaries and corpora to identify and eliminate the erroneous translation

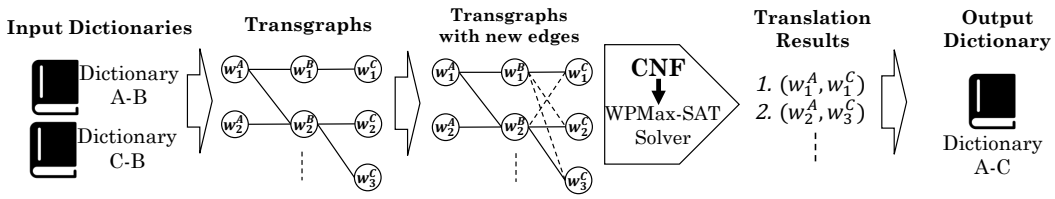


Figure 1: One-to-one constraint approach to pivot-based bilingual dictionary induction.

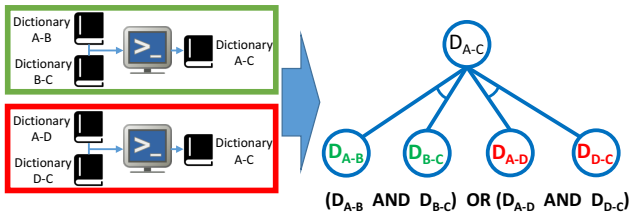


Figure 2: Modeling Bilingual Dictionary Induction Dependency.

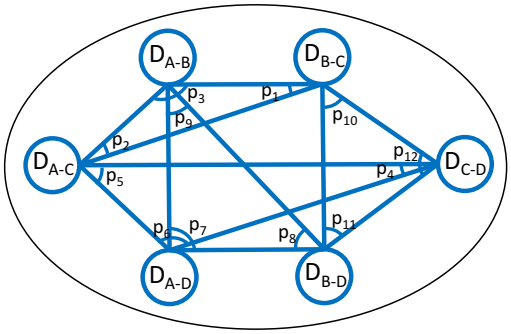


Figure 3: AND/OR Graph as an MDP State.

pair candidates. To overcome this limitation, our team (Wushouer et al., 2015) proposed to treat pivot-based bilingual lexicon induction as an optimization problem. The assumption was that lexicons of closely-related languages offer instances of one-to-one mapping and share a significant number of cognates (words with similar spelling/form and meaning originating from the same root language). The proposal uses a graph whose vertices represent words and edges indicate shared meanings; following (Soderland et al., 2009) it was called a transgraph. The proposal proceeds as follows: (1) use two bilingual dictionaries as input, (2) represent them as transgraphs where w_1^A and w_2^A are non-pivot words in language L_A, w_1^B and w_2^B are pivot words in language L_B, and w_1^C, w_2^C and w_3^C are non-pivot words in language L_C, (3) add some new edges represented by dashed edges based on the one-to-one assumption, (4) formalize the problem into conjunctive normal form (CNF) and use the Weighted Partial MaxSAT (WPMaxSAT) solver (Ansótegui et al., 2009) to return the optimized translation results, and (5) output the induced bilingual dictionary as the result. These steps are shown in Figure 1. However, the assumption of one-to-one mapping is too strong to induce the many-to-many translation pairs needed to offset resource paucity because few such pairs can be found. Therefore, we generalized the constraint-based bilingual dictionary induction framework by extending constraints and translation pair candidates from the one-to-one approach to attain more voluminous bilingual dictionary results with many-to-many translation pairs extracted from connected existing and new edges (Nasution et al., 2016). We further enhance our generalized method by setting two steps to obtaining translation pair results. First, we identify one-to-one cognates by incorporating more constraints and heuristics to improve the quality of the translation result. We then identify the cognates' synonyms to obtain many-to-many translation pairs. In each step, we can obtain more cognate and cognate synonym pair candidates by iterating the n-cycle symmetry assumption until all possible translation pair candidates have been reached (Nasution et al., 2017c).

3. Plan Optimizer

Our constraint-based bilingual dictionary induction approach has the potential to enrich low-resource languages with the only input being machine readable bilingual dictionaries. Unfortunately, the scarcity of such dictionaries for low-resource languages makes it difficult to plan which bilingual dictionary should be invested first or which bilingual dictionary should be induced right from the start in order to obtain all possible combination of bilingual dictionaries from the language set with the minimum total cost to be paid. We model the bilingual dictionary dependency with AND/OR graphs as shown in Figure 2, and employ the Markov Decision Process (MDP) for plan optimization where a state is defined by AND/OR graphs as shown in Figure 3. The exponential complexity of formulating the bilingual dictionary creation planning into a graph theory problem indicates a greater complexity of obtaining the optimal planning with the least total cost by only following the heuristic. Nevertheless, our algorithm greatly reduced the complexity, so that the MDP planning can find the feasible optimal plan with less total cost compared to heuristic planning (e.g., only use manual investment by native speaker). Our MDP model can calculate the cumulative cost while predicting and considering the probability of the pivot action to yield a satisfying output bilingual dictionary as utility for every state to better predict the most feasible optimal plan with the least total cost. Our formalization with MDP allow user to predict the feasible optimal plan with the least total cost before implementing the constraint-based bilingual dictionary induction framework in a big scale.

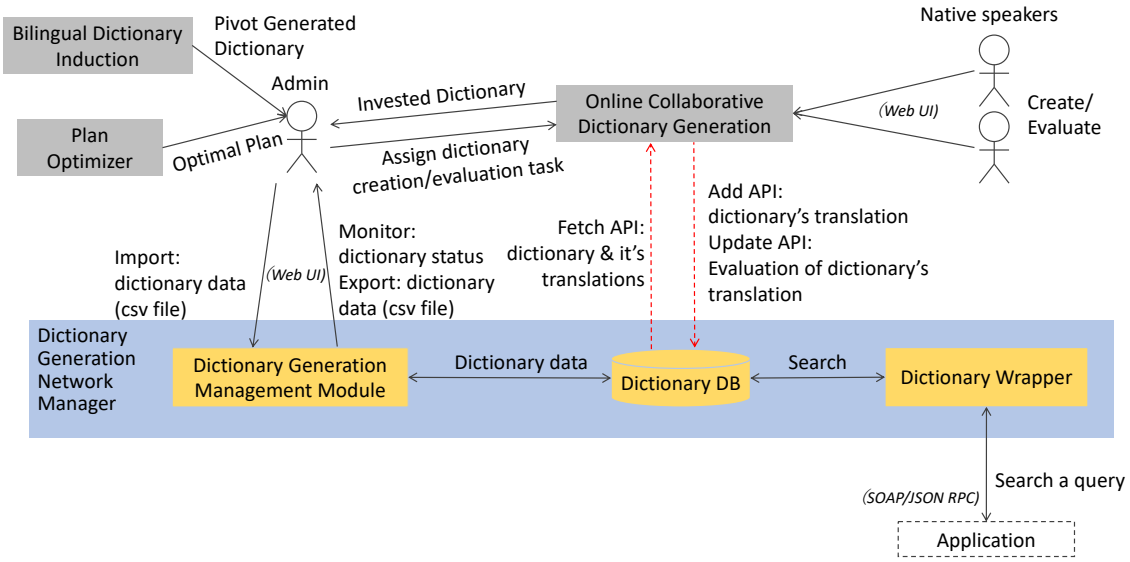


Figure 4: System Integration Overview.

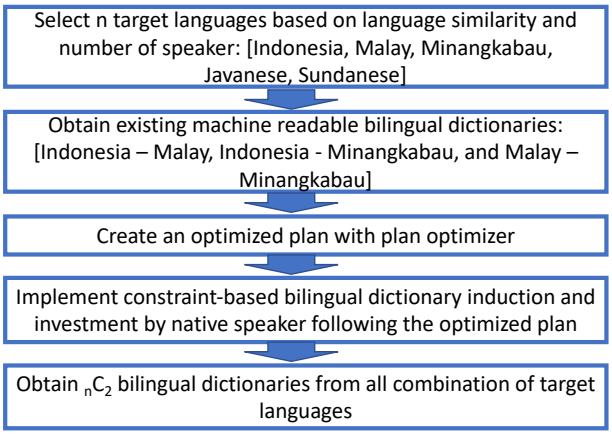


Figure 5: Overview of Bilingual Dictionaries Generation Process.

4. Designing a Collaborative Process

4.1. Overview

We integrate our Constraint-based Bilingual Dictionary Induction and Plan Optimizer with an Online Collaborative Dictionary Generation as a tool to bridge the spacial gap between native speakers and a Dictionary Generation Network Manager to manage the final dictionary so that it is accessible via API in the Language Grid (Ishida, 2011) as shown in Figure 4. The overview of bilingual dictionaries generation process is shown in Figure 5 while the detailed process is explained in Algorithm 1.

4.2. Selecting Target Languages

To select target languages in this paper, we use an Automatic Similarity Judgment Program (ASJP) (Holman et al., 2011) following our previous work (Nasution et al., 2017d). Indonesia has 707 low-resource ethnic languages (Lewis et al., 2015) that require our attention. There are two factors we consider in selecting the target languages: language similarity and number of speakers. In order to ensure that the induced bilingual dictionaries will be useful for many users, we listed the top 10 Indonesian ethnic languages ranked by the number of speakers. Since our constraint-based approach works better on closely related languages, we further generated the language similarity matrix by utilizing ASJP as shown in Table 1. Based on number of speaker, we select Javanese and Sundanese. To find and coordinate native speakers of those languages, we collaborate with Telkom University. Based on relatedness with Indonesian, we select Malay and Minangkabau. To find and coordinate native speakers of those language, we collaborate with Islamic University of Riau. Hence, we target 5 languages, i.e., Indonesian, Malay, Minangkabau, Javanese, and Sundanese. We want to enrich/create the following dictionaries: Indonesia-Malay, Indonesia-Minangkabau, Indonesia-Javanese, Indonesia-Sundanese, Malay-Minangkabau, Malay-Javanese, Malay-Sundanese, Minangkabau-Javanese, Minangkabau-Sundanese, and Javanese-Sundanese with 2,000 translation pairs each.

4.3. Modeling Task for Native Speaker

When actually implementing our constraint-based bilingual dictionary induction approach, we need native speakers for manual creation of bilingual dictionaries or evaluation of the output dictionaries. There are a lot of prior researches on modeling workflow management (Georgakopoulos et al., 1995; Hollingsworth and Hampshire, 1995; Kappel et al., 2000; Huang et al., 2000; Alexopoulos et al., 2011; Kulkarni et al., 2012). We define several rules of which native speaker can create/evaluate which dictionary.

A bilingual dictionary between ethnic language L_A and ethnic language L_B, D_{A-B} can be induced from a triple T_{A-ID-B}, while a triple T_{A-ID-B} can be induced from a bilingual dictionary D_{ID-A} and a bilingual dictionary D_{ID-B}. A bilingual dictionary between Indonesian language L_{ID} and ethnic language L_A, D_{ID-A} can be man-

Algorithm 1: Bilingual Dictionaries Generation

Input: targetLanguageInfo, existingDictionaries
/* In this project, targetLanguages: [Indonesia,Malay,Minangkabau,Javanese,Sundanese] */
/* targetLanguageInfo includes language similarities and expectedDictionarySize=2,000 */
/* existingDictionaries=[$D_{Indonesia-Malay}$, $D_{Indonesia-Minangkabau}$, $D_{Malay-Minangkabau}$] */
Output: dictionaryList /* all combination of bilingual dictionaries from the targetLanguages */

for *each* D_{A-B} *in existingDictionaries* **do**
  dictionaryList.add(D_{A-B});
end
optimizedPlan ← planOptimizer.create(targetLanguageInfo, dictionaryList);
for *each action to create bilingual dictionary* D_{A-B} *in optimizedPlan* **do**
  if *final state is reached* **then**
    return dictionaryList
  end
  else
    if *action type = investment* **then**
      /* CT1(L_{ID}, L_A): Creation and Evaluation of Indonesia-Ethnic Bilingual Dict */
      if L_A *or* L_B *is Indonesian language* L_{ID} **then**
        create and evaluate bilingual dictionary D_{A-B} by a native bilingual speaker S_{A-B};
        dictionaryList.add(D_{A-B});
      end
      /* CT2(L_A, L_B): Creation and Evaluation of Ethnic-Ethnic Bilingual Dict */
      else
        if *native bilingual speaker* S_{A-B} *is available* **then**
          create and evaluate bilingual dictionary D_{A-B} by a native bilingual speaker S_{A-B};
          dictionaryList.add(D_{A-B});
        end
        else
          create and evaluate triple T_{A-ID-B} by two native bilingual speakers S_{ID-A} and S_{ID-B};
          induce D_{A-B} from T_{A-ID-B}; dictionaryList.add(D_{A-B});
        end
      end
    end
    else if *action type = pivot* **then**
      use constraint-based approach to obtain triple T_{A-P-B} ;
      /* T4(L_A, L_P, L_B) */
      if *native bilingual speaker* S_{A-B} *is available* **then**
        evaluate triple T_{A-P-B} by a native bilingual speaker S_{A-B};
        induce D_{A-B} from T_{A-P-B}; dictionaryList.add(D_{A-B});
      end
      else
        evaluate triple T_{A-P-B} by two native bilingual speakers S_{ID-A} and S_{ID-B};
        induce D_{A-B} from T_{A-P-B}; dictionaryList.add(D_{A-B});
      end
    end
  end
end

ually created or evaluated by a native bilingual speaker S_{ID-A}. A bilingual dictionary D_{A-B} can be manually created or evaluated by a native bilingual speaker S_{ID-A} and a native bilingual speaker S_{ID-B} collaboratively or by a native bilingual speaker S_{A-B} alone.

There are some bilingual dictionaries between Indonesian and Indonesian ethnic languages exist in a printed format. We may be able to digitalized the printed Indonesian - ethnic language bilingual dictionaries to a machine readable format. Nevertheless, when we connect the digitalized bilingual dictionary D_{ID-A} and a bilingual dictionary D_{ID-B} via Indonesian language L_{ID} as a pivot, and further induced D_{A-B} with our constraint-based approach, we expect that there will be many unreachable translation pair candidates since some Indonesian words in one bilingual dictionary may not exist in the other bilingual dictionary. In order to maximize the use of our pivot-based approach, we prepare a list of 2,000 most commonly used Indonesian words to be translated to ethnic language L_A to create a bilingual dictionary D_{ID-A} by a native bilingual speaker S_{ID-A} as shown in Figure 6. Due to budget limitation, we only allow the native speaker to translate an Indonesian word to up to five words of ethnic language L_A. To ensure the quality of the manually created bilingual dictionary D_{ID-A}, another native bilingual speaker S_{ID-A} will evaluate the translation pairs as shown

Table 1: Similarity Matrix of Top 10 Indonesian Ethnic Languages Ranked by Number of Speakers

Language	**Indonesian**	Malang	Yogyakarta	**Javanese**	**Sundanese**	**Malay**	Palembang Malay	Madurese	**Minangkabau**
Malang	23.46%								
Yogyakarta	27.29%	87.36%							
Javanese	**24.09%**	47.50%	52.18%						
Sundanese	**39.43%**	18.55%	22.43%	**21.82%**					
Malay	**85.10%**	20.53%	24.35%	**21.36%**	**41.12%**				
Palembang Malay	68.24%	33.97%	37.97%	31.85%	38.90%	73.23%			
Madurese	34.45%	17.63%	14.15%	15.18%	19.86%	34.16%	34.32%		
Minangkabau	**61.59%**	26.59%	29.63%	**25.01%**	**30.81%**	**61.66%**	63.60%	34.32%	
Buginese	31.21%	12.76%	16.85%	18.33%	24.80%	32.04%	31.00%	17.94%	32.00%

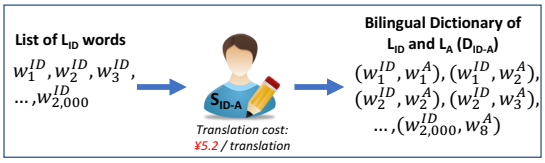


Figure 6: $T1(L_{ID}, L_A)$: Creation of Bilingual Dictionary D_{ID-A}.

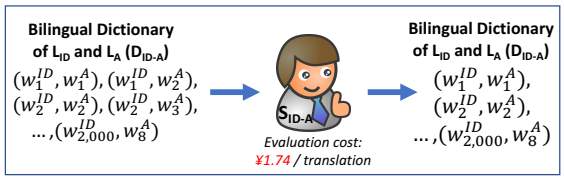


Figure 7: $T2(L_{ID}, L_A)$: Evaluation of Bilingual Dictionary D_{ID-A}.

in Figure 7. To overcome the limitation in finding native bilingual speakers of two ethnic languages for creation and evaluation of bilingual dictionary D_{A-B}, two native bilingual speakers S_{ID-A} and S_{ID-B} can collaborate as shown in Figure 8 and Figure 9 respectively. Finally, there are two composite tasks, which are $CT1(L_{ID}, L_A)$, a manual creation followed by evaluation of bilingual dictionary D_{ID-A} as shown in Figure 10a and $CT2(L_A, L_{ID}, L_B)$, a manual creation followed by evaluation of bilingual dictionary D_{A-B} as shown in Figure 10b.

4.4. Online Collaborative Dictionary Generation

The online collaborative dictionary generation has 6 modules: individual creation of Indonesia-Ethnic bilingual dictionary, individual evaluation of Indonesia-ethnic bilingual dictionary, individual creation of ethnic-ethnic bilingual dictionary, individual evaluation of ethnic-ethnic bilingual dictionary, collaborative creation of ethnic-ethnic bilingual dictionary, and collaborative evaluation of ethnic-ethnic bilingual dictionary. Each native speakers get his/her own user account. They can login to the system, read the user manual, update their profile, check their assigned task, and do their assigned task. For the individual task, the native speakers can do the task anywhere before the deadline as shown in Figure 11. However, for the collaborative task, a pair of native speakers need to login to the system at the same time in order to collaborate. The live chat is used to

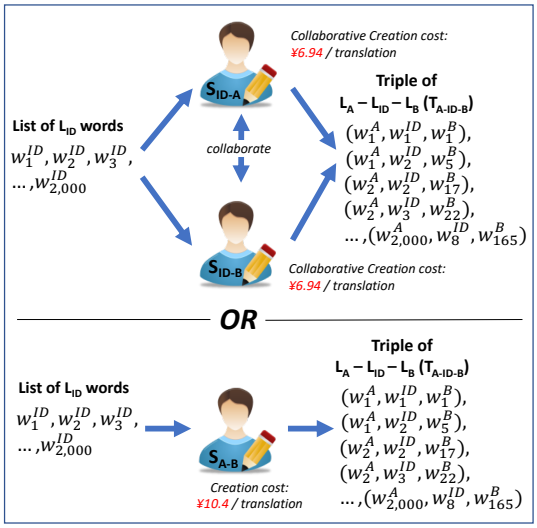


Figure 8: $T3(L_A, L_{ID}, L_B)$: (Individual/Collaborative) Creation of Triple T_{A-ID-B} to induce Bilingual Dictionary D_{A-B}.

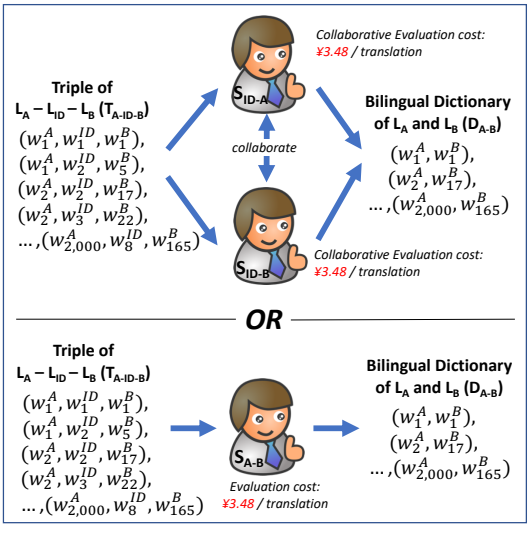


Figure 9: $T4(L_A, L_{ID}, L_B)$: (Individual/Collaborative) Evaluation of Triple T_{A-ID-B} to induce Bilingual Dictionary D_{A-B}.

ease communication and discussion during the collaborative creation / evaluation session as shown in Figure 12.

Table 2: Estimated Cost of Actions following MDP Optimal Plan

Task following MDP Plan	#Translation[1]	MDP Transition Probability[2]	Estimated Precision[2]	Unit Cost (JPY)	Total Cost (JPY)
T1(Indonesian, Malay)	1,480[3]			5.20	7,696.00
T2(Indonesian, Malay)	1,480			1.74	2,575.00
T1(Indonesian, Javanese)	2,000			5.20	10,400.00
T2(Indonesian, Javanese)	2,000			1.74	3,480.00
T1(Indonesian, Sundanese)	2,000			5.20	10,400.00
T2(Indonesian, Sundanese)	2,000			1.74	3,480.00
P(Malay, Indonesia, Minangkabau)	1,645[3]	0.983	0.4113	0.00	0.00
T4(Malay, Indonesian, Minangkabau)	754			6.96	5,248.00
P(Javanese, Indonesian, Sundanese)	1,027	0.972	0.2567	0.00	0.00
T4(Javanese, Indonesian, Sundanese)	1,027			6.96	7,147.00
T3(Javanese, Sundanese)	973			13.88	13,507.00
T4(Javanese, Sundanese)	973			6.96	6,773.00
P(Malay, Indonesia, Javanese)	1,094	0.943	0.2481	0.00	0.00
T4(Malay, Indonesia, Javanese)	1,094			6.96	7,615.00
T3(Malay, Javanese)	906			13.88	12,575.00
T4(Malay, Javanese)	906			6.96	6,305.00
P(Minangkabau, Indonesian, Sundanese)	1,157	0.949	0.289	0.00	0.00
T4(Minangkabau, Indonesian, Sundanese)	1,157			6.96	8,049.00
T3(Minangkabau, Sundanese)	844			13.88	11,708.00
T4(Minangkabau, Sundanese)	844			6.96	5,871.00
P(Malay, Indonesian, Sundanese)	1,356	0.826	0.3045	0.00	0.00
T4(Malay, Indonesian, Sundanese)	1,356			6.96	9,434.00
T3(Malay, Sundanese)	645			13.88	8,946.00
T4(Malay, Sundanese)	645			6.96	4,486.00
P(Minangkabau, Malay, Javanese)	1,148	0.929	0.2608	0.00	0.00
T4(Minangkabau, Malay, Javanese)	1,148			6.96	7,993.00
T3(Minangkabau, Javanese)	852			13.88	11,820.00
T4(Minangkabau, Javanese)	852			6.96	5,927.00
TOTAL					**171,435.00**

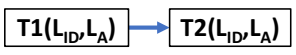
[1] *A number of translations is calculated from the number of translation pair candidates from the constraint-based approach × estimated precision with a high polysemy rate.*
[2] *Estimated from beta distribution based on language similarity and high polysemy pivot rate following our unpublished ACM TALLIP article entitled "Plan Optimization to Bilingual Dictionary Induction for Low-Resource Language Families".*
[3] *Excluding translation pairs from existing bilingual dictionaries: Indonesian-Malay (520 translation pairs) and Malay-Minangkabau (1,246 translation pairs).*

table

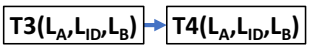
Table 3: Estimated Cost of Actions following Heuristic Plan

Task following Heuristic Plan	#Translation[1]	Unit Cost (JPY)	Total Cost (JPY)
T1(Indonesian, Javanese)	2,000	5.20	10,400.00
T2(Indonesian, Javanese)	2,000	1.74	3,480.00
T1(Indonesian, Sundanese)	2,000	5.20	10,400.00
T2(Indonesian, Sundanese)	2,000	1.74	3,480.00
T1(Indonesian, Malay)	1,480[1]	5.20	7,696.00
T2(Indonesian, Malay)	1,480	1.74	2,575.20
T3(Javanese, Sundanese)	2,000	13.88	27,760.00
T4(Javanese, Sundanese)	2,000	6.96	13,920.00
T3(Malay, Minangkabau)	754[1]	13.88	10,465.52
T4(Malay, Minangkabau)	2,000	6.96	13,920.00
T3(Malay, Javanese)	2,000	13.88	27,760.00
T4(Malay, Javanese)	2,000	6.96	13,920.00
T3(Minangkabau, Sundanese)	2,000	13.88	27,760.00
T4(Minangkabau, Sundanese)	2,000	6.96	13,920.00
T3(Malay, Sundanese)	2,000	13.88	27,760.00
T4(Malay, Sundanese)	2,000	6.96	13,920.00
T3(Minangkabau, Javanese)	2,000	13.88	27,760.00
T4(Minangkabau, Javanese)	2,000	6.96	13,920.00
TOTAL			**270,816.72**

[1] *Excluding translation pairs from existing bilingual dictionaries: Indonesian-Malay (520 translation pairs) and Malay-Minangkabau (1,246 translation pairs).*



(a) $CT1(L_{ID}, L_A)$: Composite Task Creation and Evaluation of Bilingual Dictionary D_{ID-A}.



(b) $CT2(L_A, L_{ID}, L_B)$: Composite Task Creation and Evaluation of Bilingual Dictionary D_{A-B}.

Figure 10: Composite Tasks.

4.5. Cost Estimation

We estimate the cost of each native speaker tasks as follows:

- $T1(L_{ID}, L_A)$: From an estimated duration of 30 seconds per translation and a daily wage of JPY5,000/8 hours, the estimated total translation per day is $1 \times 2 \times 60 \times 8 = 960$ and the estimated cost is JPY5.2 per correct translation.

- $T2(L_{ID}, L_A)$: From an estimated duration of 10 seconds per translation and a daily wage of JPY5,000/8 hours, the estimated total translation per day is $1 \times 6 \times 60 \times 8 = 2,880$ and the estimated cost is JPY1.74 per

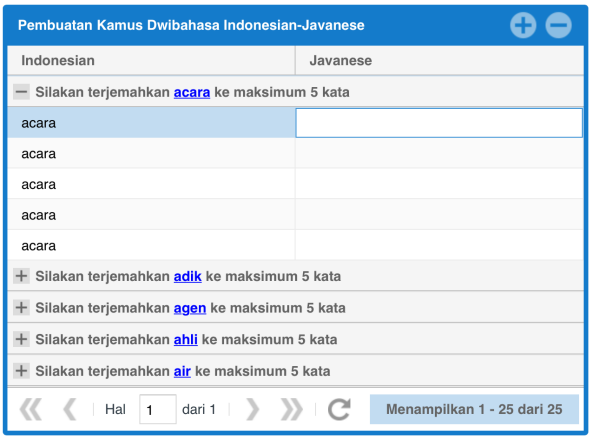


Figure 11: Individual Creation of Indonesia-Ethnic Bilingual Dictionary.

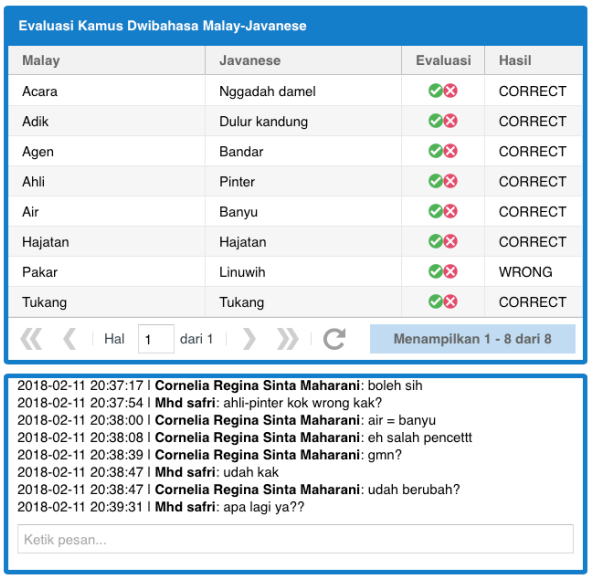


Figure 12: Collaborative Evaluation of Ethnic-Ethnic Bilingual Dictionary.

correct translation.

- $T3(L_A, L_{ID}, L_B)$: Following the cost of $T1(L_{ID}, L_A)$ and $T2(L_{ID}, L_A)$, for the individual task, from an estimated duration of 60 seconds per translation, the estimated cost is JPY5.2×2 = JPY10.4 per translation. For the collaborative task, from an estimated duration of 30 seconds to translate an Indonesian word to each ethnic language in parallel, and an extra 10 seconds for discussing the sense sharing between the two ethnic language translations, the estimated total cost is (JPY5.2 + JPY11.74)×2 workers = JPY13.88 per correct translation pair.

- $T4(L_A, L_{ID}, L_B)$: Following the cost of $T1(L_{ID}, L_A)$ and $T2(L_{ID}, L_A)$, for the individual task, from an estimated duration of 20 seconds per translation, the estimated cost is JPY1.74×2 = JPY3.48 per translation. For the collaborative task, from an estimated duration of 20 seconds to evaluate by discussing the sense sharing between the two ethnic language translations, the estimated total cost is (JPY1.74 + JPY1.74)×2 workers = JPY6.96 per correct translation pair.

- $CT1(L_{ID}, L_A)$: Following the cost of $T1(L_{ID}, L_A)$ and $T2(L_{ID}, L_A)$, the estimated cost is JPY5.2 + JPY1.74 = JPY6.94 per translation.

- $CT2(L_A, L_B)$: Following the cost of $T3(L_A, L_{ID}, L_B)$ and $T4(L_A, L_{ID}, L_B)$ and the combination of workers based on availability of native bilingual speakers ($S_{A-B} + S_{A-B}$, $S_{A-B} + S_{ID-A}\&S_{ID-B}$, $S_{ID-A}\&S_{ID-B} + S_{A-B}$, $S_{ID-A}\&S_{ID-B} + S_{ID-A}\&S_{ID-B}$), the variations of estimated total cost are (JPY10.4 + JPY3.48 = JPY13.88, JPY10.4 + JPY6.96 = JPY17.36, JPY13.88 + JPY3.48 = JPY17.36, JPY13.88 + JPY6.96 = JPY20.84) respectively.

We estimate the cost of actions following the optimized plan utilizing both constraint-based approach and manual investment by native speakers as shown in Table 2 and further compare them with cost of actions following the heuristic plan utilizing only manual investment by native speakers as shown in Table 3.

5. Conclusion

We design a collaborative process for creating 10 bilingual dictionaries with 2,000 translation pairs from every combination of 5 languages, i.e., Indonesian, Malay, Minangkabau, Javanese, and Sundanese. We implement our plan optimizer and our generalized constraint approach to bilingual dictionary induction in creating input dictionaries or evaluating the resulting bilingual dictionaries. We define a heuristic plan that only utilize manual investment by native speaker to evaluate our optimal plan with total cost as an evaluation metric. By following the optimal plan, we can reduce 63.3% cost of following the heuristic plan. We further design an online dictionary generation tool to bridge spatial gap between native speakers. We will analyze the native speakers' behavior and chat log for future improvement of the system.

6. Acknowledgment

This research was partially supported by a Grant-in-Aid for Scientific Research (A) (17H00759, 2017-2020) and a Grant-in-Aid for Young Scientists (A) (17H04706, 2017-2020) from Japan Society for the Promotion of Science (JSPS). The first author was supported by Indonesia Endowment Fund for Education (LPDP). We thank our collaborators, Department of Informatics Engineering, Universitas Islam Riau and Department of Computational Science, Telkom University for their effort in gathering the student workers.

7. Bibliographical References

Alexopoulos, K., Makris, S., Xanthakis, V., and Chryssolouris, G. (2011). A web-services oriented workflow

management system for integrated digital production engineering. *CIRP Journal of Manufacturing Science and Technology*, 4(3):290 – 295. Production Networks Sustainability.

Ansótegui, C., Bonet, M. L., and Levy, J. (2009). Solving (weighted) partial maxsat through satisfiability testing. In *Theory and Applications of Satisfiability Testing-SAT 2009*, pages 427–440. Springer.

Georgakopoulos, D., Hornick, M., and Sheth, A. (1995). An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3(2):119–153, Apr.

Hollingsworth, D. and Hampshire, U. (1995). Workflow management coalition: The workflow reference model. *Document Number TC00-1003*, 19.

Holman, E. W., Brown, C. H., Wichmann, S., Müller, A., Velupillai, V., Hammarström, H., Sauppe, S., Jung, H., Bakker, D., Brown, P., et al. (2011). Automated dating of the world's language families based on lexical similarity. *Current Anthropology*, 52(6):841–875.

Huang, G., Huang, J., and Mak, K. (2000). Agent-based workflow management in collaborative product development on the internet. *Computer-Aided Design*, 32(2):133 – 144.

Toru Ishida, editor. (2011). *The Language Grid: Service-Oriented Collective Intelligence for Language Resource Interoperability*. Springer Publishing Company, Incorporated.

Ishida, T. (2016). Intercultural collaboration and support systems: A brief history. In *International Conference on Principles and Practice of Multi-Agent Systems (PRIMA 2016)*, pages 3–19. Springer.

Kappel, G., Rausch-Schott, S., and Retschitzegger, W. (2000). A framework for workflow management systems based on objects, rules and roles. *ACM Comput. Surv.*, 32(1es), March.

Kulkarni, A., Can, M., and Hartmann, B. (2012). Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, pages 1003–1012, New York, NY, USA. ACM.

M. Paul Lewis, et al., editors. (2015). *Ethnologue: Languages of the World*. SIL International, Dallas, Texas, 18th edition.

Nasution, A. H., Murakami, Y., and Ishida, T. (2016). Constraint-based bilingual lexicon induction for closely related languages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3291–3298, Paris, France, May.

Nasution, A. H., Murakami, Y., and Ishida, T. (2017a). Plan optimization for creating bilingual dictionaries of low-resource languages. In *2017 International Conference on Culture and Computing (Culture and Computing)*, pages 35–41, Sept.

Nasution, A. H., Syafitri, N., Setiawan, P. R., and Suryani, D. (2017b). Pivot-based hybrid machine translation to support multilingual communication. In *2017 International Conference on Culture and Computing (Culture and Computing)*, pages 147–148, Sept.

Nasution, A. H., Murakami, Y., and Ishida, T. (2017c). A generalized constraint approach to bilingual dictionary induction for low-resource language families. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 17(2):9:1–9:29, November.

Nasution, A. H., Murakami, Y., and Ishida, T. (2017d). Similarity cluster of indonesian ethnic languages. In *Proceedings of the First International Conference on Science Engineering and Technology (ICoSET 2017)*, pages 12–27, Pekanbaru, Indonesia, November.

Soderland, S., Etzioni, O., Weld, D. S., Skinner, M., Bilmes, J., et al. (2009). Compiling a massive, multilingual dictionary via probabilistic inference. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 262–270. Association for Computational Linguistics.

Tanaka, K. and Umemura, K. (1994). Construction of a bilingual dictionary intermediated by a third language. In *Proceedings of the 15th conference on Computational linguistics-Volume 1*, pages 297–303. Association for Computational Linguistics.

Wushouer, M., Lin, D., Ishida, T., and Hirayama, K. (2015). A constraint approach to pivot-based bilingual dictionary induction. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 15(1):4:1–4:26, November.

Constructing a Lexicon of Relational Nouns

Edward Newell, Jackie C.K. Cheung
edward.newell@mail.mcgill.ca
jcheung@cs.mcgill.ca
McGill University

Abstract

Relational nouns refer to an entity by virtue of how it relates to another entity. Their identification in text is a prerequisite for the correct semantic interpretation of a sentence, and could be used to improve information extraction. Although various systems for extracting relations expressed using nouns have been developed, there are no dedicated lexical resources for relational nouns. We contribute a lexicon of 6,224 labeled nouns which includes 1,446 relational nouns. We describe the bootstrapped annotation of relational nouns, and develop a classifier that achieves 70.4% F1 when tested on held out nouns that are among the most common 2,500 word types in Gigaword. We make the lexicon and classifier available to the scientific community.

Keywords: relational noun, relation extraction, possessive, lexicon

1. Introduction

The extraction of relations is a fundamental aspect of natural language understanding, playing a central role in knowledge base construction, question answering, and recognizing textual entailment. Semantic relations can be expressed via a syntactically diverse set of constructions. Consider the following approximate paraphrases:

(1a) Jack **befriended** Jill 5 years ago.
(1b) Jack and Jill's **friendship** is now 5 years old.
(1c) Jack has been Jill's **friend** for 5 years.

Whereas **(1a)** establishes the relation by means of a verb, **(1b)** and **(1c)** do so by means of nouns. In example **(1c)**, *friend* is part of a semantically motivated class of nouns called *relational nouns*, which we focus on in this work.

Relational nouns refer to an entity by virtue of how it relates to something else (Barker, 2011). In the above example, *friend* establishes a relation between its referent, *Jack*, and the external entity, *Jill*, which could be depicted using a two-place predicate such as `friend(Jack, Jill)`. This differs from how a sortal (i.e. non-relational) noun like *person* is interpreted. The sentence *Jill is a person* could be depicted as a unary predicate, `person(Jill)`. In other words, there is an absolute set of people, but there is no absolute set of friends, without first specifying the person with whom they are friends.

Despite this semantic difference, relational nouns behave syntactically like other nouns. Semantic parsers such as Boxer that are trained on CCGBank do not currently distinguish between relational and non-relational nouns, leading to errors in sentences that contain them (Bos, 2008). Relation extraction systems such as RENOUN (Yahya et al., 2014) and RELNOUN (Pal and Mausam, 2016) rely on automatically extracted patterns, and also do not make a distinction between relational and non-relational nouns. Various information extraction systems could be improved by the identification of relational nouns. In this work, we create a high-quality lexicon of relational nouns using bootstrapped manual annotation.

Relational nouns pose a conceptually difficult annotation task, in part because the meaning of most nouns involves some relation at least indirectly. Part of our contribution is a series of decisions about what should be included or excluded from the class; decisions which are made with applications to relation extraction in mind. We describe these decisions and the approach to annotation.

Based on our annotation results, approximately one out of every 15 nouns, by type, is relational. This represents a substantial semantic class, but means that a significant amount of annotation effort will be expended on annotating the negative class. To more efficiently deploy annotation effort, we bootstrap annotation using a relational noun classifier.

We obtain a dataset of 6,224 labelled nouns, containing 1,446 relational nouns and 4,778 sortal nouns. We also develop a classifier that achieves 70.4% F1 when classifying held-out nouns from the top 2,500 most common nouns. We release the dataset[1] and classifier[2] to the scientific community so they can be used in relation extraction systems and other NLP applications.

2. Related work

There exists a line of work examining the properties of relational nouns using a formal theoretical framework (De Bruin and Scha, 1988; Partree, 2008; Laczkó et al., 2009; Barker, 2011). There, the focus is on developing a consistent theoretical treatment rather than lexical resources. Many of our decisions and heuristics used in annotation and classification are based on this literature.

Other work focuses on the development of resources and systems for nominal semantics, which is complementary to our work. This includes NomBank (Meyers et al., 2004), and other work on nominal semantic role labelling (Padó et al., 2008; Gerber and Chai, 2010). These resources and systems are concerned with the argument structure of nouns (relational nouns are often treated as taking arguments). But these resources do not focus on relational nouns in particular. NomBank, by its extension to NomLex (Macleod et al., 1998), includes a short list of 331 relational nouns. By focusing specifically on relational nouns,

[1] http://cgi.cs.mcgill.ca/~enewel3/publications/relational-nouns-lrec-2018

[2] https://github.com/enewe101/relational-nouns-lrec-2018

we provide a greater than fourfold increase in the number of labelled relational nouns (1,446). Within the context of Open Information Extraction, earlier work such as ReVerb focused on extracting relations from verbs (Fader et al., 2011). Mausam et al. (2012) examined the role of nouns and adjectives as bearers of predicates as well, showing that doing so increases coverage. Yahya et al. (2014) developed the RENOUN system, which focuses on extracting information about rarer attributes expressed using nouns.

The work most related to ours is the RELNOUN Open IE system, most recently augmented by Pal and Mausam (2016). This work extracts relations expressed using nouns, including relational nouns, using a combination of deterministic patterns and lexical resources, yielding 209 correct extractions from 2,000 newswire sentences.

Our current work complements automated extraction systems like RENOUN and RELNOUN by assembling a lexicon of relational nouns. Rather than only relying on automatically extracted patterns, we use manual annotation and bootstrapping to create a high-quality lexicon. This lexicon of relational nouns can be used as a semantic resource in relation extraction and other NLP tasks.

3. Operationalizing Relational Nouns

Our operationalization of relational nouns follows the theoretical treatment in the context of possessive constructs fairly closely. However, we deviate in certain cases (as will be noted) to prioritize relation extraction applications.

Applying the definition of relational nouns during annotation is quite difficult, in part because relational nouns carry no strong syntactic characteristics that distinguish them. It helps to decompose the definition of relational nouns into two criteria: (1) relational nouns must provide intrinsic lexical evidence that a relation is being expressed, and (2) relational nouns must refer to one of the members in the relationship expressed. This is equivalent to the prior definition in which a relational noun identifies an entity by virtue of how it relates to something else. But the decomposition makes it more obvious that there are two ways that a noun can fail to be relational.

Criterion 1 eliminates sortal nouns such as *car*. Consider, by way of example, the sentences *that is Jill's car* and *Jack is Jill's brother*. There we see that the referents of *car* and *brother* are both participating in relations. In contrast, if Jill utters the sentence *Jack is a brother* one would infer `brother(Jack, Jill)`, whereas uttering *that is a car* indicates no corresponding relation. Lacking specific context, we see that *car* does not provide intrinsic lexical evidence for a relation.

Criterion 2 eliminates nouns that indicate relationships yet which are not relational nouns, such as event and result nominals. Consider *disagreement* and *reconciliation*: although these both indicate a relation, notice that instead of referring to one of the participants of the relation, these nouns refer to the relation itself. So too do nouns like *friendship*. Although these nouns do indicate relations, they are not proper relational nouns and are correctly eliminated by criterion 2. (Note that agent and patient deverbal nouns, such as *employee* and *employer* adhere to criterion 2, and so are considered relational.)

In preparation for designing the annotation task, we collected many relational nouns given as examples in the literature, and organized them under the broad types of relations they expressed, shown in **Table 1**. In our experience, annotators perceive these classes as quite different, so it is useful to decompose the notion of relational nouns in terms of the subclasses. We found that sequentially introducing the subclasses simplified annotator training. We now review the subclasses:

Kinship. Kinship nouns like *brother*, describe family relations, and are the most common example in the literature.

Social non-kin. This includes informal roles, like *friend*, and formal or organizational roles like *mayor*, *CEO*, or *goalie*. Nouns that depict roles without providing lexical evidence for a relation, such as *butcher*, are excluded.

Operational. This includes non-social relations: purpose, cause/effect, function, representation, etc. Theories of possessive constructs dictate that relational nouns can occupy postnominal possessive constructions (Barker, 2011), but we include nouns like *cure* even though it is more natural to say *the cure **for** the disease*, than it is to say *the cure **of** the disease*.

Relative parts. This includes nouns designating a physical region based on a spatial or temporal relationship, such as *corner* or *intro*. These are typically reified by the relation itself: the corner of a desk exists by virtue of being the corner and cannot exist apart from the desk.

In creating these subclasses, we have specifically excluded two others containing nouns normally considered relational (Partee and Borschev, 2003; Cresswell, 1996; Laczkó et al., 2009; Barker, 2011; Lichtenberk et al., 2011; Partree, 2008; De Bruin and Scha, 1988):

Body parts. All body part nouns are traditionally considered to be relational (Laczkó et al., 2009). However, the relation between a body part and the whole body does not seem to be an essential part of the meaning of body part nouns. Supose one is practicing drawing ears, and is asked *what are you drawing?* In the response *that is an ear*, the word *ear* does not seem to indicate any sort of relation. Even in the gruesome case of a real disembodied ear, no relation is implied when one says *that is an ear*. The same cannot be said for other nouns we consider relational. With that said, in many cases a body part noun *can* have a relational meaning, when it specifically represents the relation that the body part has to the rest of the body, or a generalized analogical meaning, e.g. *head* as in *head of the department* or *leg* as in *leg of the journey*. We do consider these nouns relational (belonging either the operational or relative part subclasses), but do not admit all body part nouns wholesale.

Properties. Nouns such as *height*, or *colour* are also often treated as relational nouns, but we distinguish between nouns that identify the properties of individual entities from those establishing relationships *between* entities. We consider properties to warrant a separate annotation effort (in part undertaken in (De Bruin and Scha, 1988)).

Excluding body part nouns and properties, the operational

Subclass	Literature examples	Additional examples
Kinship	**Brother**[6], **sister**[5], **mother**[4], **child**[4†], **grandmother**[3], **husband**[2], **wife**, **spouse**, **father**, **daughter**, **aunt**, **uncle**, **cousin**, **family**, **relative**	**Stepdaughter**, **brother-in-law**, **kin**, **kindred**
Social non-kin	**Enemy**[4], **friend**[3], **pet**[2], **stranger**[2], **neighbour**[2], **mayor**, **governor**, **commander**, **co-author**, **employee**, **tutor**	**Captain**[†], **guitarist**[†], **investor**[†], **linebacker**, **lawyer**[†], **spokesperson**, **entourage**, **confidante**
Operational	**Name**[2], **birthday**[2], **picture**[2], **portrait**[2†], **tracks**[†], **sake**, rumor, **description**, **reputation**	**Passenger**, **solution**, **cure**, **hole**
Relative part	**Edge**[2], **mantel**[†], **side**, **corner**, **middle**	**Top**, **tip**[†], **base**[†], **leg**[†], **eye**[†], **body**[†], **face**[†],
Body part	**Leg**[3†], hand[2], **head**[†], **eye**[†], **body**[†], **face**[†], bone, blood, voice, ulcer, nose, sweat, hairdo, tears, finger	Epithelium, duodenum
Properties	Height[2], speed[2], distance, rating, length, readiness, color, weight, shape, temperature, gesture, habit, fear, posture, shadow	Happiness, badness, capacity, range, clarity, piety, extravagance

Table 1: relational noun examples drawn from the literature (Lichtenberk et al., 2011; Partree, 2008; Laczkó et al., 2009; Cresswell, 1996; De Bruin and Scha, 1988; Barker, 2011; Asudeh, 2005; Partee and Borschev, 2003), grouped by subclass, along with additional examples we provide. superscripts indicate the number of distinct authors that used the noun as an example. the nouns which we consider relational under our operationalization are shown in bold, with a dagger (†) indicating that the noun also has prominent non-relational meaning(s).

subclass catches any other non-social, non-spatio-temporal relational noun, making this classification exhaustive.

3.1. Annotation

We conducted annotation using 3 experts (natural language processing and linguistics researchers) and 13 non-expert annotators. We used 392 expert-annotated seed examples drawn from the literature and randomly sampled from Gigaword (Graff and Cieri., 2003) as a source of training examples and quality-control test questions. In an initial training phase, non-expert annotators learned about each relational noun subclass, and annotated examples from each with immediate feedback. Then, in a quiz phase, annotators completed 25 test (seed) examples before proceeding to the task. During the task, 2 of every 25 examples was randomly chosen to be a test example to track annotation quality. We only include data from annotators that sustained more than 70% correct test questions throughout the quiz and annotation.

Many nouns have both relational and non-relational meanings. To better serve as an input to NLP systems, we adopt a recall-oriented stance, and consider a noun to be relational if it has any relational meanings. However, during annotation, annotators were instructed to label a noun as *usually relational* if they judge relational sense(s) to be strictly more likely in general usage, and *occaisionally relational* if the noun has relational sense(s) that they judge not to be strictly more likely. Nouns lacking any a relational sense were to be labelled *almost never relational*.

Each noun was presented without context, and was annotated either by 1 expert or at least 3 non-experts (5 if initial non-expert annotations were not unanimous), except for a subset of 250 nouns which was annotated by all annotators (expert and non-expert) to test agreement. We reconcile multiple annotations by taking the *mode label* that received the most votes. Ties are resolved by applying the label *occasionally relational*. For nouns with expert and non-expert annotations, we only consider the expert annotations.

Consistent with our recall-oriented stance, we consider both the *occasionally relational* and *usually relational* as relational, but the distinction is preserved in the dataset so that analysts can use it as an indication of confidence that a given word in context is relational. Interested readers can view a complete reproduction of the annotation guidelines[3].

As mentioned, 250 nouns were annotated by all annotators to measure agreement. Agreement (Krippendorff's α, squared error metric) was 0.53 among expert annotators, 0.47 among non-expert participants, and 0.43 among all annotators. These modest agreement levels reflect the fact that the task is conceptually extremely challenging. They were the highest agreement levels we achieved after several iterations of task design and pilot studies. Reviewing disagreements, in many cases, annotators failed to recall a given noun's alternative relational meaning(s). In part this contributed to our decision to default to *occasionally relational* to resolve ties.

3.2. Bootstrapping

In choosing nouns to annotate, we sought to balance the desire to annotate common words, the desire to cover rare phenomena, and the desire limit the amount annotation effort expended on labelling negative examples.

To focus on examples likely to be positive, we bootstraped the sampling of nouns using a relational noun classifier trained on partial data, corresponding to the *feature-rich* model to be described in §4.2. We began with the expert-annotated seed set of 392 examples. During each round of bootstrapped annotation, we trained a model based on the existing annotations, and use the model to select new nouns for the next round of annotation, while oversampling those predicted by the classifier to be relational. We still included some nouns predicted by the classifier to be negative, to

[3] http://cgi.cs.mcgill.ca/~enewel3/publications/relational-nouns-lrec-2018

avoid excessive drift or bias. Sampling nouns for annotation was done in three rounds, with approximately 2,000 nouns sampled in each round. Nouns were sampled by taking, among nouns not yet annotated: 800 nouns scoring highest in the classifier's decision function, 200 nouns scoring lowest, 800 most common nouns in Gigaword, and 800 nouns uniformly randomly selected from Gigaword (but occurring at least 5 times). These sets overlap to some extent, resulting in each round contributing somewhat fewer than 2,000 nouns (1944 on average). The classifier's positive samples were enriched in relational nouns by factor of 8, being 49.8% relational.

4. Relational noun detection

As mentioned, the annotations provide three-class labels: *usually relational*, *occasionally relational*, and *almost never relatoinal*. But, motivated to create a more recall-oriented model, we collapse *usually relational* and *occasionally relational* into a single positive class.

Automatically detecting relational nouns is quite difficult, given (a) relational nouns behave grammatically the same way as sortal nouns, (b) the annotation of relational nouns is quite difficult even for people, and (c) even after merging the *usually relational* and *occasionally relational* classes, there is considerable class imbalance.

4.1. Baseline

Our baseline classifier draws on the observation of Barker (2011) that relational nouns characteristically arise under possessive constructions, specifically the prenominal (*Jill's brother*) and the postnominal (*the brother of Jill*) constructions. We collected counts for the number of times that nouns arose in each grammatical context in Gigaword (Graff and Cieri., 2003), as a fraction of total occurrences, and used these as features in shallow learners described below.

4.2. Feature-rich model

To create a more performant classifier, we added the following features:

Dependency tree features. We note the frequency with which a noun arises in particular dependency tree contexts, according to Stanford CoreNLP's dependency parse (Manning et al., 2014) of Gigaword. The dependency tree context was encoded by starting from the noun, and following a non-intersecting path along dependency tree relations for up to three hops, while noting the sequence (and direction) of the relations, and the part of speech (POS) tags of the nodes.

Sequence features. We record counts for the POS, lemma, and surface form, and bigrams thereof, of the surrounding 10 tokens along with their relative position.

Morphological features. We record the suffix of the noun, based on a list of common suffixes (see supplementary material[4]).

Derivational features. We record whether a noun is derivationally related to words of other POSs, based on WordNet's `derivationally_related_forms` (Fellbaum, 1998).

Semantic features. 300-dimensional Google word embeddings (Mikolov et al., 2013).

Hand-crafted features. 31 theoretically and empirically motivated features based on surrounding lemmas, POS-tags, named entities, and dependency-tree relations in Gigaword (Graff and Cieri., 2003) (see supplementary material[4]).

4.3. Feature transformations

We tried alternative encodings of dependency tree and sequence features: raw counts, frequencies (i.e. counts normalized by the number of occurrences of the noun), log-frequencies, and binarized features based on a threshold of the pth percentile for feature's frequency, $p \in \{25, 50, 75\}$. We pruned these features to the top k with highest mutual information, for $k \times 10^{-3} \in \{10, 20, 40, 80\}$. **Table 2** shows the performance of classifiers using each representation, showing that the frequency representation performed best.

4.4. Learning algorithms

Using the scikit-learn package (Pedregosa et al., 2011), we optimized over several learners for both the baseline and feature rich models: support vector machine (SVM), logistic regression (LR), naive Bayes (NB), and random forest (RF). For learners with linear decision surfaces (SVM, LR, and NB), we shifted the surface to optimize F1 on the training set due to class imbalance. As shown in **Table 3** SVM performed best.

4.5. Feature Ablation

The various features mentioned above were ablated to determine the best-performing combination of features. As shown in **Table 4**, The best performance is achieved by including all except the morphological features.

4.6. Classifier performance

Because we are interested in using the classifier both for bootstrapping and for classification of unannotated nouns, we report two evaluation metrics: the average precision, and the F1 score. Average precision considers the ranking of nouns by order of decreasing likelihood of being relational according to the classifier's decision function, rather than the actual classification, which indicates the usefulness of the classifier in bootstrapping. F1 provides a measure of the binary prediction recall and precision, which is more useful in NLP applications when classifying un-annotated nouns.

Optimizing F1 over the space of learners, hyperparameters, features, and feature representations, the best performance on the dev set was achieved by excluding morphological features, using frequencies, $k = 80,000$, while using an SVM with radial basis function, and $C = 100$ and $\gamma = 0.001$[5].

The best-performing classifier was one built using al features except morphological (suffix) features, and using

[4] http://cgi.cs.mcgill.ca/~enewel3/publications/relational-nouns-lrec-2018

[5] see Pedregosa et al. (2011) for the meaning of γ and C.

Feature representation	AP (%)	F1 (%)
raw counts	56.3	58.6
frequency	61.1	63.2
log-frequency	53.2	56.0
threshold-25%	56.2	59.3
threshold-50%	58.4	60.3
threshold-75%	54.8	58.6

Table 2: Performance of relational noun classifiers using various feature representations. Support vector machine is the learner for each model shown. AP = Average Precision.

classifier	AP (%)	F1 (%)
Support Vector Machine	79.8	76.8
Random Forest	75.1	69.9
Logistic Regression	79.6	74.8
Naive Bayes	53.8	67.6

Table 3: Performance of relational noun classifiers using a variety of learners. All models shown use all features except the morphological (suffix-based) features and use the frequency encoding. AP = Average Precision.

SVM as the learner. This classifier achieved average precision $AP = 76.1\%$ and $F1 = 70.4\%$ on the 2500 most common nouns in Gigaword (Graff and Cieri., 2003) and $AP = 49.5\%$ and $F1 = 46.7\%$ on randomly sampled nouns occurring at least 5 times in Gigaword.

While these performances are modest, for the purposes of bootstrapping, the set of nouns predicted to be relational turned out to be enriched by eight fold over the baseline rate of occurrence, from approximately 1 in 15 words to 1 in 2. This was crucial to increase the number of relational nouns discovered during the study.

As previously mentioned, the subclasses of relational nouns we identified were perceived very differently by annotators, and seemed to differ greatly in difficulty. Based on post-hoc manual classification of 300 nouns from the held-out test set, according to relational noun subclass, we can see that classifier performance also differs greatly by subclass (**Table 5**). Whereas classifier recall is high for kinship and social non-kin nouns, it is very low for operational and relative part nouns.

Ablated feature	AP (%)	F1 (%)
baseline	80.2	75.2
dependency	80.3	76.1
hand-picked	79.9	75.2
lemma	80.2	75.2
surface	80.3	74.9
POS	80.1	75.2
derivational	80.3	75.5
google-vectors	80.2	75.2
suffix	79.8	76.8

Table 4: Performance of relational noun classifiers using various feature sets. Each classifier is built using all features except the ablated feature; lower scores indicate greater importance of the feature. All models used support vector machine with the frequency representation of count-based features. AP = Average Precision.

Subclass	Recall (%)	Fraction of unique token types (%)
Kinship	80.0	5.1
Social non-kin	85.5	46.9
Operational	37.0	33.0
Relative part	34.3	15.0

Table 5: Classifier recall for various subclasses of relational noun, and the size of the subclasses in terms of the fraction of unique tokens.

The kinship nouns constitute the smallest subclass by number of unique token types, so it is at first surprising that recall was very high. However, this stands to reason when one considers that they tend to have fewer alternate non-relational meanings, and are the more common class by number of occurrences (rather than by token type). These facts mean that their corpus-derived features are based on a larger number of occurrences in Gigaword and may have less diversity in their contexts (and hence features).

On the other hand, the operational and relative part nouns had very low recall, despite acounting for an intermediate number of examples (by number of unique token types). This may be due to the fact that these words tend to labelled *occasionally relational* more frequently than the other relational noun subclasses. This results both from polysemy and annotator disagreement, both of which lead to greater difficulty for the classifier—in the first case due to more diverse contexts in Gigaword (and presumably poor clustering in feature space), and in the second case due to noisier training data.

Overall, recall was highest for the social non-kin nouns, which may be due to a combination of having fewer non-relational meanings and being a much larger class compared to the other three (by number of unique token types).

5. Conclusions

In this paper we presented the first effort dedicated to building a high-quality lexicon of relational nouns, and contribute the largest lexicon of relational nouns to date.

The lexicon is based on an operationalization of relational nouns designed with applications to relation extraction in mind. We provide the dataset of 6,224 nouns, including 1,446 relational nouns, along with a relational noun classifier to the research community for inclusion in relation extraction systems.

Delineating the class of relational nouns, whether automatically or manually, is a very challenging task. We hope this first effort will provide a starting point for future work to develop additional lexical resources for relational nouns and relation extraction.

6. Acknowledgements

We wish to acknowledge many helpful conversations with Professor Brendan Gillion of the Linguistics Department at McGill University. Funding for this research was provided by the Natural Sciences and Engineering Research Council of Canada.

7. Bibliographical References

Asudeh, A. (2005). Relational nouns, pronouns, and resumption. *Linguistics and Philosophy*, 28(4):375–446.

Barker, C. (2011). Possessives and relational nouns. *Semantics: An international handbook of natural language meaning*, 2:1109–1130.

Bos, J. (2008). Wide-coverage semantic analysis with Boxer. In *Proceedings of the 2008 Conference on Semantics in Text Processing*, pages 277–286. Association for Computational Linguistics.

Cresswell, M. J., (1996). *Relational Nouns*, pages 21–36. Springer Netherlands, Dordrecht.

De Bruin, J. and Scha, R. (1988). The interpretation of relational nouns. In *Proceedings of the 26th annual meeting on Association for Computational Linguistics*, pages 25–32. Association for Computational Linguistics.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Gerber, M. and Chai, J. Y. (2010). Beyond NomBank: A study of implicit arguments for nominal predicates. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1583–1592. Association for Computational Linguistics.

Graff, D. and Cieri., C. (2003). English Gigaword LDC2003T05.

Laczkó, T., Butt, M., and King, T. H. (2009). Relational nouns and argument structure: Evidence from hungarian.

Lichtenberk, F., Vaid, J., and Chen, H.-C. (2011). On the interpretation of alienable vs. inalienable possession: A psycholinguistic investigation. *Cognitive Linguistics*, 22(4):659–689.

Macleod, C., Grishman, R., Meyers, A., Barrett, L., and Reeves, R. (1998). Nomlex: A lexicon of nominalizations. In *Proceedings of EURALEX*, volume 98, pages 187–193.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mausam, Schmitz, M., Bart, R., Soderland, S., and Etzioni, O. (2012). Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). The nombank project: An interim report. In *HLT-NAACL 2004 workshop: Frontiers in corpus annotation*, volume 24, page 31.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Padó, S., Pennacchiotti, M., and Sporleder, C. (2008). Semantic role assignment for event nominalisations by leveraging verbal data. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 665–672. Association for Computational Linguistics.

Pal, H. and Mausam. (2016). Demonyms and compound relational nouns in nominal open ie. *Proceedings of AKBC*, pages 35–39.

Partee, B. H. and Borschev, V. (2003). Genitives, relational nouns, and argument-modifier ambiguity. *Modifying adjuncts*, 4:67–112.

Partree, B., (2008). *Genitives: A Case Study*, pages 182–189. Blackwell Publishing Ltd.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Yahya, M., Whang, S., Gupta, R., and Halevy, A. Y. (2014). Renoun: Fact extraction for nominal attributes. In *EMNLP*, pages 325–335.

Creating Large-Scale Multilingual Cognate Tables

Winston Wu, David Yarowsky
Department of Computer Science, Center for Language and Speech Processing
Johns Hopkins University
{wswu,yarowsky}@jhu.edu

Abstract

Low-resource languages often suffer from a lack of high-coverage lexical resources. In this paper, we propose a method to generate cognate tables by clustering words from existing lexical resources. We then employ character-based machine translation methods in solving the task of cognate chain completion by inducing missing word translations from lower-coverage dictionaries to fill gaps in the cognate chain, finding improvements over single language pair baselines when employing simple but novel multi-language system combination on the Romance and Turkic language families. For the Romance family, we show that system combination using the results of clustering outperforms weights derived from the historical-linguistic scholarship on language phylogenies. Our approach is applicable to any language family and has not been previously performed at such scale. The cognate tables are released to the research community.

Keywords: cognates, clustering, transliteration

eng	lat	fra	ita	spa	por	cat
table	mensa	?	mensa	mesa	mesa	?
table	?	table tableau	tabella tavolo tavola	tabla	tabela	taula

eng	azj	tat	tuk	tur	uig	uzn
table	stol	östäl	stol	?	üstel	stol
table	?	tablis	tablisa	tablo	?	tablitsa

Figure 1: Each row in the table is a *cognate chain*. The task of *cognate chain completion* is to fill in missing cells in the table.

1. Introduction

Cognates are words in related languages that share a common origin. For example, the Italian *cavallo* and French *cheval* both originated from the Latin *caballus*. Besides being instrumental in historical linguistics, cognates find uses in many areas of NLP, including machine translation (Kondrak et al., 2003; Nakov and Tiedemann, 2012) and lexicon induction (Mann and Yarowsky, 2001).

We define the task of cognate chain completion, shown in Figure 1. Given multi-way aligned cognate table, a cognate "chain" is a group of cognates across a language family (represented as a single row). Chains may have empty cells due to dictionary gaps, denoted by a ?, and the task is to predict these missing entries. Cognate chain completion is related to the task of cognate transliteration, except that words in related languages (within the same row) can contribute to the hypothesis of a cell. For low-resource languages, generating hypotheses for missing cognates has applications in alignment and resolving unknown words in machine translation. In the field of linguistics, examining cognates across multiple related languages can shed light on how words are borrowed between languages.

Cognate lists are not widely available for many languages and are time-consuming to create by hand. In many NLP-related applications, including the translating out-of-vocabulary words in machine translation, it is often not necessary that these words be true cognates in the linguistic sense, i.e. they are descendants of a common ancestor (Ciobanu and Dinu, 2014). For example, names and loanwords are not technically considered cognates, though they behave as such. Rather, "cognates" only need to meet certain established criteria for cognacy (Kondrak, 2001; Inkpen et al., 2005; Ciobanu and Dinu, 2014), which include individually or a combination of orthographic, phonetic, and semantic similarity between words.

Previous approaches to cognate transliteration (Mulloni, 2007; Beinborn et al., 2013) suffer from the drawback that they require an existing list of cognates, which is infeasible for low-resource languages. In contrast, we automatically generate cognate tables by clustering words from existing lexical resources using a combination of similarity measures. Our produced cognate tables for Romance and Turkic languages are available for research purposes [1]. Using these cognate tables, we construct multi-way bitext and train character-based machine translation systems to transliterate cognates to fill in missing entries in the cognate chains. Finally, we evaluate multiple methods of system combination on the cognate chain completion task, showing improvements over single language-pair systems. For the Romance languages, we find that performance-based weight outperforms combining weights derived from a linguistic phylogeny.

2. Data

We begin with lemmas from two free lexical resources, PanLex (Baldwin et al., 2010) and Wiktionary[2]. From PanLex, we pivot words on English and extract foreign-English translation pairs, retaining each word's Meaning IDs,[3] and its most common backtranslation in PanLex[4]. From Wiktionary, we use translation pairs mined from the info boxes on the English version of the site (Sylak-Glassman et al.,

[1] github.com/wswu/coglust

[2] wiktionary.org

[3] An identifier indicating semantic relatedness. A single word may have multiple Meaning IDs, and words in different languages may have the same meaning ID.

[4] The most common backtranslation is the most frequent English translation of the foreign word.

Foreign	accado
English	Akkadian
Language	ita
Backtranslation	Akkadian
POS	NOUN, ADJ
Meaning IDs	4444597, 32087717

Table 1: A translation pair extracted from Panlex and Wiktionary.

2015). In addition, we retain a word's part of speech[5]. We preprocess the data by removing words in all caps (abbreviations) and words with spaces and symbols. Table 1 illustrates an example of a single translation pair extracted from the combination of PanLex and Wiktionary.

3. Cognate Clustering

To generate multilingual cognate tables, we employ an automatic method of clustering words from our lexical resources. In contrast to Scherrer and Sagot (2014), who compare entire word lists to find possible cognates, we only consider two words to be cognates if they have the same English translation. Pivoting through English removes the need to compute a similarity score between every pair of words in every list, thus reducing the time complexity required to perform alignment. In addition, by introducing a strict semantic similarity constraint, we avoid clustering false cognates, which are orthographically similar by semantically distant.

On each group of words with the same English translation, we perform single-linkage clustering, an agglomerative clustering method where the distance between two clusters X and Y is defined as $D(X, Y) = \min_{x \in X, y \in Y} d(x, y)$for some distance metric d between two points (in our case, words) x and y. While clusters formed using this linkage method tend to be thin, we found that this method works well for cognates spread out across a language family compared to other linkage methods. We examine different linkage methods in Section 3.1.

First, we construct lists of plausible cognates from our data by running an initial clustering step on each group of words. In this step, the distance function is the unweighted normalized Levenshtein distance $\frac{LD(x,y)}{\frac{|x|+|y|}{2}}$, and clusters are merged if the distance falls under a generous threshold of 0.5.

Treating these clusters as multi-way aligned bitext, we run GIZA++ (Och and Ney, 2000) to extract character-to-character substitution probabilities, which are used in a second clustering step. The idea is that a second iteration of clustering should produce better results than a single iteration. This is similar to the two-pass approach employed by (Hauer and Kondrak, 2011).

For the second iteration of clustering, we define the distance function d between two words x and y as a linear combination of the following features, chosen specifically to model both the orthographic and semantic relatedness of cognates.

[5]A word may have multiple or no POS tags. PanLex also contains POS tags, but they are noisy, so we only use those from Wiktionary

Inter-Language Distance A normalized weighted Levenshtein distance, where the insertion, deletion, and substitution costs are specific to the language pair (A, B) and the characters being compared (a, b).

$$\text{Ins}(a) = 1 - p_{A \to B}(\text{NULL} \to a) \quad (1)$$

$$\text{Del}(a) = 1 - p_{A \to B}(a \to \text{NULL}) \quad (2)$$

$$\text{Sub}(a, b) = 1 - p_{A \to B}(a \to b) \quad (3)$$

The character transition probabilities are obtained from alignment using GIZA++. They are subtracted from 1 to convert them to costs used in the edit distance calculation. We added an addition rule such that the distance between identical characters is zero to account for the noisy nature of alignment.

Intra-Family Distance Same as the inter-language distance, except that the probabilities are obtained by running an aligner on the concatenation of all bitexts of every language pair. This is a more universal, non-language-specific distance, and we expect it to smooth or counter-balance the inter-language distance if there is not enough data for an accurate measure of inter-language distance. The intra-family distance is also used as a fallback distance in place of the Inter-Language Distance when comparing words of the same language. In practice, we observed that the intra-family distances are very close to the inter-language distance.

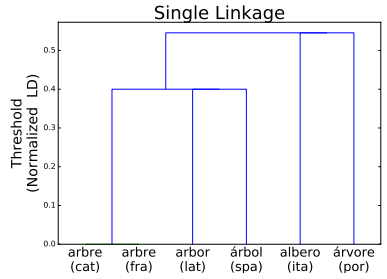
Same Backtranslation A word's backtranslation is the most frequent English translation of that word in PanLex. If a word is in Wiktionary but not in PanLex, we assign the backtranslation to be that word's English translation. This feature is 0 if two words' most common backtranslation is the same, or 1 if they are different.

Same POS Part of speech is obtained from the English edition of Wiktionary. Polysemous words may have multiple parts-of-speech. If a word is in Panlex but not in Wiktionary, the word will not have a POS. PanLex also contains POS tags for words, but we choose not to use them because they are often incorrect (e.g. due to OCR errors), and words seem to be marked as nouns by default. This feature is 0 if two words share a common part of speech, and 1 otherwise.

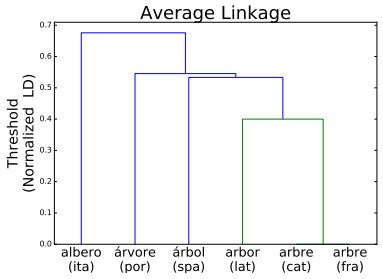
Same MeaningID A word from PanLex has a set of possible Meaning IDs that link it to semantically equivalent words in other languages. If a word exists in PanLex, we use all Meaning IDs that occur with this word. A word in Wiktionary but not in PanLex will not have a Meaning ID. This feature is 0 if two words share a common Meaning ID and 1 otherwise.

3.1. Evaluation of Linkage Methods

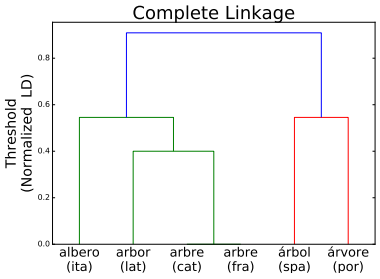
We motivate our choice of clustering linkage method by illustrating the results of our multiple-iteration clustering approach using hierarchical clustering with different linkage methods: single-linkage, complete-linkage, and average-linkage. These methods differ only in the metric used to



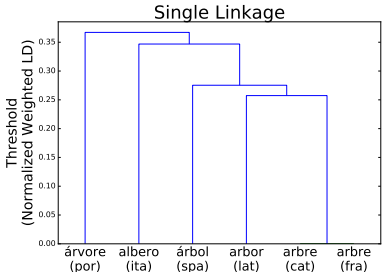
(a) Single Linkage Clustering using Unweighted Distance



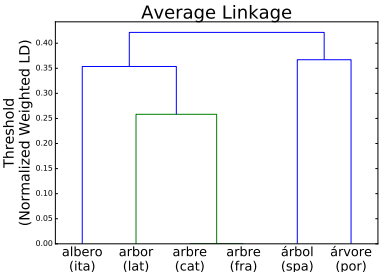
(b) Average Linkage Clustering using Unweighted Distance



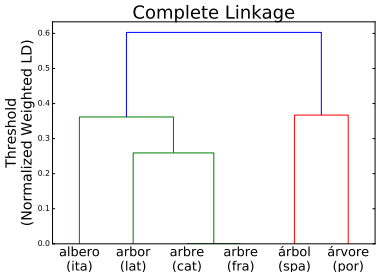
(c) Complete Linkage Clustering using Unweighted Distance



(d) Single Linkage Clustering using Weighted Distance



(e) Average Linkage Clustering using Weighted Distance



(f) Complete Linkage Clustering using Weighted Distance

Figure 2: Results of different linkage methods with unweighted and weighted distances

merge clusters:

$$\text{Single(X, Y)} = \min_{x \in X, y \in Y} d(x, y) \quad (4)$$

$$\text{Complete(X, Y)} = \max_{x \in X, y \in Y} d(x, y) \quad (5)$$

$$\text{Average(X, Y)} = \frac{1}{|X||Y|} \sum_{x \in X, y \in Y} d(x, y) \quad (6)$$

for some distance function d.

In Figures 2a to 2c, using an unweighted normalized Levenshtein distance, *arbre* in Catalan and *arbre* in French are immediately grouped into the same cluster because they have a distance of zero. Ideally, we would like all of these words to be put into the same cluster, since they are true cognates. Single linkage seems to fulfill our needs the best, because the range of distances for merging clusters is the smallest.

When performing a second iteration of clustering using the weighted distances, the dendrograms in Figures 2d to 2f show similar results. Notably, the range of distances between clusters shrinks, which supports our hypothesis that multiple iterations of clustering are beneficial.

4. Experiments and Results

We experiment on the Romance and Turkic families to illustrate our method on both high-resource and lower-resource languages. From the Romance languages, we utilize Latin, Italian, French, Spanish, Portuguese, Romanian, and Catalan. For Turkic, we use Azerbaijani, Kazakh, Turkish, Uyghur, Turkmen, and Uzbek.

Our data contains over 1M words for the Romance languages and 130K words for Turkic languages. The specific breakdown per language is shown in Table 2. Performing the cognate clustering results in a total of 204,065 non-singleton clusters for Romance and 16,931 for Turkic, both substantially larger than prior cognate studies.

Romance		Turkic	
fra	286,002	tur	80,063
ita	281,015	tuk	17,028
spa	239,360	kaz	16,048
por	189,105	azj	10,195
cat	93,442	tat	5,303
lat	88,602	uzn	4,375
rom	1,119	uig	2,118

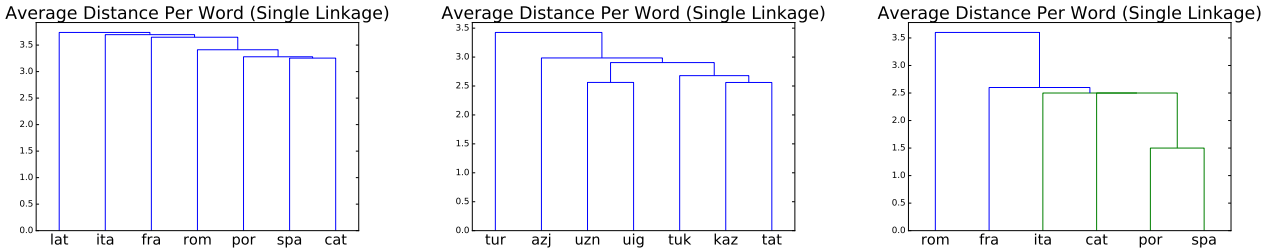
Table 2: Total number of words per language

4.1. Character-Based Machine Translation for Transliteration

Although we might ideally evaluate the quality of the cognate clusters against a gold list of cognate pairs (e.g. Beinborn et al. (2013)), an alternative is to evaluate on a downstream task, namely cognate chain completion.[6] To do this, we consider all cognate pairs within each cluster as translations of each other and construct bitext for each language pair, where characters are separated with spaces. Intuitively, if a machine translation system can translate well using this data, then the cognate chains have been correctly constructed.

We train character-based Moses (Koehn et al., 2007) SMT systems for each language pair, using a standard setup of GIZA++ (Och and Ney, 2000), a 5-gram KenLM (Heafield, 2011) trained with the `--discount-fallback` option, and

[6]This is similar to the task of Scherrer and Sagot (2014). Since we use a different set of languages from a different data source, we cannot directly compare to this work. However, we emphasize that since Scherrer and Sagot (2014) computes a distance between all pairs of words to determine cognacy, our approach of pivoting through English is computationally more efficient.



(a) Romance Language Distance

(b) Turkic Language Distance

(c) Distances from Gray and Atkinson (2003)

Figure 3: Distance between languages

no distortion, since reordering should not occur during transliteration (Karimi et al., 2011). For each language pair, We generate a 10-best list of distinct hypotheses. While MT systems are generally evaluated on BLEU score (Papineni et al., 2002), it is not clear that BLEU is the best metric for evaluating transliterations: Beinborn et al. (2013) find that tuning on BLEU score made almost no difference in their system's performance. Nevertheless, we tune using MERT (Och, 2003) with the standard Moses scripts. For each experiment in Figure 4, we report 1-best accuracy, 10-best accuracy (is the truth in the top 10 hypotheses?), and mean reciprocal rank (MRR): MRR $= \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\text{rank}_i}$
Due to the way the bitext is constructed (i.e. the cross product of all words in a cognate cluster), the same source word often maps to different output words, e.g.

src (por)	tgt (ita)
associação	associamento
associação	associazione

which makes this an inherently hard task for machine translation systems. Thus, to compute accuracy, we consider a hypothesis to be correct if it matches any of the words in the set of gold words.

4.2. System Combination

While the results of single-language-based systems are indicative of the missing word translation prediction performance achieved via a single related language, we seek to improve performance by combining prediction models from multiple related languages. For a given target language, the hypotheses from all systems transliterating into that target language are combined using performance-based weighting, where the weight of a system is proportional to its performance relative to the other systems for a given target language. For example, when transliterating into French, the normalized scores for the cat-fra, ita-fra, lat-fra, etc. systems become those systems' respective weights). Within the hypotheses of each system, we employ a simple rank-based scoring Within each language, where the score of the first-ranked hypothesis is 1.0, the second-ranked is 0.9, the third-ranked is 0.8, etc. These are multiplied by the performance-based system scores, resulting in a ranked list of hypotheses. We retain the top 10 best hypotheses in order to compare with the single (non-combined) systems. System combination results in Figure 4 are labeled as "SC" and the metric used for the systems' weights (1-best, 10-best, and MRR).

4.3. Analysis

One overall result that we noticed was that a key parameter predicting performance was the relative amount of source dictionary data available for a given language, in addition to phylogentic similarity and degree of cultural contact. For example, in the Romance family, translations into Romanian scored well below translations into other languages, likely due to the small amount of available data compared to its sister languages.

For the Romance languages, we found improvements using system combination for all target languages when evaluating on 1-best accuracy (Figure 4a), which is is the system's best guess for transliterating into a language if it was forced to output only a single answer. When evaluating the percentage of words that occurred in the top 10 hypotheses (Figure 4b), system combination performed close to the best performing language pair. For MRR (Figure 4c), system combination also performed better than all language pairs. Because of how MRR is calculated, increases in the ranking of the truth in the hypothesis list results in an increase in MRR.

For the Turkic languages, system combination performed similar to the single language systems in the 1-best (Figure 4d) and 10-best (Figure 4e) experiments. The major exception is the pair Kazakh-Tatar, which are each other's best single-language source and are quite close in edit distance. We believe this was due to the small amount of data for most Turkic language pairs, which may also explain the higher performance on Romance languages compared to Turkic languages. Using a fixed-weight weighting scheme where the system that performed the best on the dev set received a weight of 0.8 and the others 0.1 resulted in large improvements in 10-best accuracy.

We also compare against an established theory of language evolution (Gray and Atkinson, 2003), using their notion of inferred maximum-likelihood estimates of evolutionary change per cognate as a measure of distance between languages. They analyzed a large set of Indo-European languages, excluding Latin. We find that using performance-based weights outperforms using weights derived from well-established phylogentic trees and distances published in the historical linguistic literature, possibly due to the phenomenon of borrowing. For example, while Spanish

English	Gold	Lang	Hypotheses
Zero	Zero	ita	Cero, Caro, Zero, Cereo, Zeno, Zereo, Chiro, Zairo, Coro, Sero
you are	stai, state, siete	ita	statis, estas, esto, restai, estais, ista, istas, estes, restatis, estatis
frambesia	framboesia	ita	frambesia, framboisia, frambonesia, framboesia
colloquium	colloquium	lat	colloquium, colloquius, coloquium, coloquius, colloquio
host	hostis, hospes	lat	hostis, hosti, hosped, hostie, hostes, hospet, hospide, conviva, hospe, suspes
cave	covo,cavea	lat	cavus, cavum, cavo, cau, caverna, caberna, copus, cavernus, cave, cubus
chopper	chóper	spa	chuparse, chofer, chupar, compar, copiar, chupatorio, copar, chaparse, chopper, compararse
apple	manzana	spa	mansana, mandana, mansada, mansanna, mansiana, amansana, manosana, manillana, mansiano, manisana
elision	elisión	spa	elisión, alisión, delisión, emisión, elección, elimiento, enlisión, elisiones, olisión, adolisión
vagolytic	vagolytique	fra	vagolithique, vagolitique, vagabolithique
mixture	mixer, mixage, ...	fra	mixture, mixturer, mêler, masculer, mixtion, mixer, mixtior, masculaire, mesquiller, miscer
bigos	bigos	fra	bigos, bigus, bigous, bigues, bios, bige, Bigos, bingous, begues, vigos
butterfly	paparuga, peperuga	rom	papillos, purbolekto, papollono, ûrbolekto, papiillon, Borboleto, papallono, papollos, palomi, purboleto
Croatia	Croazia, Kroatiya	rom	Croakia, Croakiia, Croaatiia, Crroakia, Croatia, Croakiya, Croagia, krroato, skroato, Croaatiio
dove	gulumbo, kolombo	rom	kolombo, skulombo, limbo, pombo, posmva, pompa, koloma, lomba, koaoma, ppombo
divorce	divorzio, divorzile, divórcio, ...	por	divorciar, divorcio, divorcista, divorzile, divórcista, divorcissarse, divórciar, divorcia, divorcístico, adivorcio
cybernaut	cibernauta	por	cibernauta, cybernauta, Cibernauta, internauta
patio	pátio	por	patia, patino, partido, pátio, partio, patio, patrio, patiano, pastio, pato

(a) Examples of system combination results using 1-best weights for Romance languages

English	Gold	Lang	Hypotheses
skirt	yubka	azj	yubka, yubqa, jubka, jubqa, yubkə, Yubka, yubxa, übka, yübka, yubqə
fluorine	fluor, flüor	azj	ftor, ftar, flüor, ftər, faor, fdor, fluor, fdar, vlor, lor
food	gıda, qida	azj	gı, qı, qıda, gıda, gida, qışa, ğıda, ğı, kida, qada
Greece	Юнанстан	kaz	Жунанастан, жунанастан, жананастан, Жананастан, Жунаныстан, жунаныстан, Жүнанастан, жананыстан, жүнанастан, Жананыстан
where	қайда	kaz	қайда, кайда, қажда, кажда, қайта, қада, қайға, кәйда, кайта, шайда
cheese	сыр	kaz	сыр, сыл, сырт, сұр, сшыр, сын, сір, сур, сырш, сір
wall	дивар	tat	дуал, двал, дуәл, дугал, дул, дуаль, дгал, дваль, дуар, дүал
letter	harf, xäref, xat	tat	hät, tarf, xärf, xarf, Qät, hirf, harp, härp, harş, kärf
dove	yeni, yaña	tat	yaña, yaNa, yaNi, yañı, yañi, yene, yañge, yaNe, yeni, yange
weaving	dokuma, dokma	tuk	dokuma, dokamak, dokumak, dokuşmak, dokama, dokumaklyk, dokume, dokumamak, dokulamak, dokuşma
cop	polis, politsiýa	tuk	polisiýa, politsiýa, polis, milisiýa, militsiýa, pilisiýa, polits, poliz, polisi, polys
shaman	şaman, şaman	tuk	şaman, shaman, saman, sheman, naman, kaman, şeman, şamen, sharman, haman
microbe	mikrop	tur	mikrob, mikrop, mikroB, mikros, mikrobit, mikrap, mikrod, mikrok, mikrab, mikrov
professor	profesör	tur	professoğur, profestor, profesur, profesor, professor, profesör
function	işlemek	tur	işlemek, işletmek, işlenmek, inlemek, işleme, işleştirmek, işlamak
enemy	düşman, düshmen	uig	dushman, düshman, düshmen, dushmen, tushman, tüshman, tüshmen, duSman, doshman, Tushman
gymnastics	gimnastika	uig	gimnastika, qimnastika, ximnastika, gimnastiqa, Qimnastika, yimnastik, gimnastik, gimnestika, yimnastiq, gimnastiq
one-ness	birlik	uig	birliq, birlik, bir, birlük, jirlik, biriq, birlikk, birik, bhirlik, pirlik
crocodile	timsah, timsoh	uzn	timsoh, timsah, timsax, timshoh, timdal, timsog, timsox, timshah, timsoq, timmayd
Tuesday	seshanba	uzn	sishanba, says'anba, soyshanba, says'anba, tsishanba, siyshanba, sayrshanba, shoshanba, seshanba, Seshamb
selenium	selenyum, selen	uzn	selen, tselen, selan, salen, sselen, selleniy, seleniy, seleyn, soleniy, salaniy

(b) Examples of system combination results using 1-best weights for Turkic languages

Table 3: Example hypotheses from system combination.

(a) Romance, 1-best

src \ tgt	cat	fra	ita	lat	por	rom	spa
cat	—	.50	.41	.29	.46	.09	.55
fra	.38	—	.42	.28	.45	.16	.43
ita	.49	.42	—	.22	.62	.12	.44
lat	.33	.32	.38	—	.30	.04	.35
por	.55	.43	.46	.28	—	.05	.55
rom	.00	.09	.21	.03	.13	—	.13
spa	.62	.43	.45	.29	.54	.05	—
SC 1-best	**.65**	**.52**	**.52**	.37	**.66**	**.19**	**.58**
SC 10-best	.65	.52	.52	**.38**	.66	.19	.58
SC MRR	.65	.52	.52	.37	.66	.19	.58
G&A '03	.64	.51	.52	—	.63	.17	.56

(b) Romance, 10-best

src \ tgt	cat	fra	ita	lat	por	rom	spa
cat	—	**.86**	.73	.68	.77	.23	.83
fra	.72	—	.76	.69	.77	.42	.75
ita	.82	.75	—	.58	**.89**	**.45**	.76
lat	.71	.74	**.80**	—	.75	.28	.78
por	.88	.77	.77	**.71**	—	.26	**.84**
rom	.06	.32	.46	.13	.26	—	.31
spa	**.91**	.75	.75	.70	.83	.35	—
SC 1-best	.90	.81	**.80**	.70	.88	.36	**.84**
SC 10-best	.90	.82	.80	.70	.88	.41	.84
SC MRR	.90	.81	.80	.70	.88	.36	.84
G&A '03	.90	.81	.80	—	.87	.33	.83

(c) Romance, MRR

src \ tgt	cat	fra	ita	lat	por	rom	spa
cat	—	.61	.51	.40	.56	.12	.64
fra	.49	—	.53	.39	.55	.22	.53
ita	.59	.52	—	.32	.70	.19	.54
lat	.43	.43	.50	—	.42	.10	.47
por	.65	.54	.56	.40	—	.09	.65
rom	.01	.13	.29	.05	.16	—	.18
spa	.71	.53	.55	.41	.64	.13	—
SC 1-best	**.74**	**.62**	**.62**	**.49**	**.74**	**.25**	**.68**
SC 10-best	.74	.62	.62	.49	.74	.26	.68
SC MRR	.74	.62	.62	.49	.74	.25	.68
G&A '03	.73	.61	.61	—	.72	.23	.65

(d) Turkic, 1-best

src \ tgt	azj	kaz	tat	tuk	tur	uig	uzn
azj	—	.14	.18	.39	.40	.29	.45
kaz	.10	—	**.40**	.07	**.50**	.14	.33
tat	.21	**.41**	—	.24	.23	.21	.19
tuk	.39	.07	.17	—	.34	.30	.32
tur	.34	.21	.23	.22	—	.20	.27
uig	.26	.14	.14	.29	.21	—	.33
uzn	.38	.00	.18	.38	.28	.27	—
SC 1-best	**.43**	.39	.32	.36	.40	.36	.45
SC 10-best	.43	.39	.32	.37	.40	**.37**	**.46**
SC MRR	.43	.40	.32	.36	.40	.36	.46
Fixed Wt.	.40	.39	.33	**.40**	.39	.32	.44

(e) Turkic, 10-best

src \ tgt	azj	kaz	tat	tuk	tur	uig	uzn
azj	—	.39	.53	.73	**.75**	.58	.76
kaz	.46	—	**.74**	.27	.64	.29	.33
tat	.53	**.77**	—	.55	.60	.47	.64
tuk	.72	.07	.56	—	.66	.50	.72
tur	.72	.64	.58	.58	—	.49	.63
uig	.56	.14	.45	.57	.50	—	.67
uzn	.75	.17	.58	**.74**	.64	.62	—
SC 1-best	.71	.73	.61	.63	.69	.61	.75
SC 10-best	.71	.74	.61	.63	.69	.60	.75
SC MRR	.71	.74	.61	.63	.69	.61	.76
Fixed Wt.	**.78**	.75	.71	.70	.74	**.70**	**.83**

(f) Turkic, MRR

src \ tgt	azj	kaz	tat	tuk	tur	uig	uzn
azj	—	.17	.28	**.48**	.50	.36	.54
kaz	.20	—	**.50**	.15	**.55**	.21	.33
tat	.29	**.53**	—	.32	.33	.28	.31
tuk	.48	.07	.27	—	.43	.35	.43
tur	.44	.35	.32	.32	—	.28	.36
uig	.34	.14	.22	.37	.29	—	.43
uzn	.49	.06	.29	**.48**	.37	.38	—
SC 1-best	.52	.50	.42	.45	.49	**.44**	.55
SC 10-best	**.53**	.50	.42	.45	.49	.44	**.56**
SC MRR	.52	.51	.42	.45	.49	.44	.56
Fixed Wt.	.49	.50	.43	.42	.48	.40	.53

Figure 4: Results for cognate chain completion. Valid comparisons are within a column (target language).

and Portuguese are evolutionarily closer than Spanish and Catalan, our analysis, which accounts for borrowed words, places Spanish and Catalan closer than Spanish and Portuguese, likely due to external factors such as trade, migration, or the fact that Catalonia is an autonomous community of Spain. By computing the average edit distance per word in a cognate chain, we can construct phylogenic tress Figure 3 to illustrate closeness between languages.

Examples of results given by the combination of multiple systems are presented in Tables 3a and 3b. We observed that even if the truth is not the 1-best hypothesis, it is often in the top 10 hypotheses, and the top 10 hypotheses have a low edit distance to the truth. Having a top 10 list is useful for applications such as translating into a foreign language when conversing with a native speaker. In such cases, it is often not necessary to use the exact words; one only needs to produce a word that is close enough that the speaker will understand the meaning. When translating in the opposite direction, unknown words can be easily checked against entries in the top 10 to obtain a translation.

Several errors in transliteration seem to stem from inaccurate clustering, with words clustered due to the strong orthographic similarity feature. For example, Latin *hostis* 'enemy' is incorrectly clustered together with *hospes* 'host/guest', which causes some noise in the hypotheses. Similar phenomena can be observed for French *mixage*. Refining the clustering process may lead to improvements in our missing-word prediction models.

5. Related Work

Cognates have been used in the task of lexicon induction, with Mann and Yarowsky (2001) inducing translation lexicons between cross-family languages via bridge languages. They make extensive use of Levenshtein distance (Levenshtein, 1966) in determining the distance between two cognates. In our work, we employed a weighted edit distance as a major component in determining cognate clusters.

Clustering cognates has also been recently explored using different approaches to determine cognacy, e.g. using an SVM which trained to determine cognacy (Hauer and Kondrak, 2011) and accounting for a language family's phylogeny when constructing cognate groups (Hall and Klein, 2010). We experiment with using phylogenetic information in our system combination. Recently, Bloodgood and Strauss (2017) experimented with global constraints to improve cognate detection. This approach is complementary to ours and could be used to improve our cognate tables.

Several methods have also been proposed to generating cognates, e.g. using a POS tagging framework where the tags are actually target language n-grams (Mulloni, 2007). Recently, several approaches to character-based machine translation using cognates have been investigated, although on a small set of language pairs. Beinborn et al. (2013) experiment on English-Spanish with a manual list of cognates. Scherrer and Sagot (2014) perform a task similar to our own; they start with a word list and find plausible cognates using the BI-SIM metric (Kondrak and Dorr, 2004),

then perform character-based machine translation on cognates. They experiment with translating cognates from a high-resource language to a low-resource language. Our work differs in that our experiments are on a much larger scale, and we realize improvements by combining the results of multiple MT systems.

6. Conclusion

We have presented an automatic clustering method to generate cognate tables from Panlex- and Wiktionary- derived dictionary data, which we release as a resource. Based on these cognate clusters, we then trained multiple Moses-based models to complete cognate chains by generating hypotheses for missing translations, which often occur due to sparse dictionary coverage in lower-resource languages. Via several novel methods of system and model combination over multiple related languages, we realized improvements over single language-pair baselines for the Romance and Turkic language families. In addition, we also observed that our performance-based weighting of related languages in system combination outperformed language-similarity weights derived from phylogenetic trees from widely-cited historical linguistics literature, suggesting that other latent factors such as the degree of political and cultural interaction are impactful as well.

7. Acknowledgments

This work was supported in part by the DARPA LORELEI program. The findings, conclusions, and opinions found in this work are those of the authors and do not necessarily represent the views of the funding agency.

8. Bibliographical References

Baldwin, T., Pool, J., and Colowick, S. M. (2010). Panlex and lextract: Translating all words of all languages of the world. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, pages 37–40. Association for Computational Linguistics.

Beinborn, L., Zesch, T., and Gurevych, I. (2013). Cognate production using character-based machine translation. In *IJCNLP*, pages 883–891.

Bloodgood, M. and Strauss, B. (2017). Using global constraints and reranking to improve cognates detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1983–1992. Association for Computational Linguistics.

Ciobanu, M. A. and Dinu, P. L. (2014). Automatic detection of cognates using orthographic alignment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 99–105. Association for Computational Linguistics.

Gray, R. D. and Atkinson, Q. D. (2003). Language-tree divergence times support the anatolian theory of indo-european origin. *Nature*, 426(6965):435–439.

Hall, D. and Klein, D. (2010). Finding cognate groups using phylogenies. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1030–1039. Association for Computational Linguistics.

Hauer, B. and Kondrak, G. (2011). Clustering semantically equivalent words into cognate sets in multilingual lists. In *IJCNLP*, pages 865–873.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197. Association for Computational Linguistics.

Inkpen, D., Frunza, O., and Kondrak, G. (2005). Automatic identification of cognates and false friends in french and english. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 251–257.

Karimi, S., Scholer, F., and Turpin, A. (2011). Machine transliteration survey. *ACM Computing Surveys (CSUR)*, 43(3):17.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Kondrak, G. and Dorr, B. (2004). Identification of confusable drug names: A new approach and evaluation methodology. In *Proceedings of the 20th international conference on Computational Linguistics*, page 952. Association for Computational Linguistics.

Kondrak, G., Marcu, D., and Knight, K. (2003). Cognates can improve statistical translation models. In *Companion Volume of the Proceedings of HLT-NAACL 2003 - Short Papers*.

Kondrak, G. (2001). Identifying cognates by phonetic and semantic similarity. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

Mann, G. S. and Yarowsky, D. (2001). Multipath translation lexicon induction via bridge languages. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–8. Association for Computational Linguistics.

Mulloni, A. (2007). Automatic prediction of cognate orthography using support vector machines. In *Proceedings of the 45th Annual Meeting of the ACL: Student Research Workshop*, pages 25–30. Association for Computational Linguistics.

Nakov, P. and Tiedemann, J. (2012). Combining word-level and character-level models for machine translation between closely-related languages. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 301–305. Association for Computational Linguistics.

Och, F. J. and Ney, H. (2000). Giza++: Training of statis-

tical translation models.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Scherrer, Y. and Sagot, B. (2014). A language-independent and fully unsupervised approach to lexicon induction and part-of-speech tagging for closely related languages. In *Language Resources and Evaluation Conference*.

Sylak-Glassman, J., Kirov, C., Yarowsky, D., and Que, R. (2015). A language-independent feature schema for inflectional morphology. In *ACL (2)*, pages 674–680.

Lexical Profiling of Environmental Corpora

Patrick Drouin, Marie-Claude L'Homme, Benoît Robichaud
Observatoire de linguistique Sens-Texte (OLST)
Université de Montréal
C.P. 6128, succ. Centre-ville
Montréal (Québec) H3C 3J7 CANADA
{patrick.drouin, mc.lhomme, benoit.robichaud}@umontreal.ca

Abstract

This paper describes a method for distinguishing lexical layers in environmental corpora (i.e. the general lexicon, the transdisciplinary lexicon and two sets of lexical items related to the domain). More specifically we aim to identify the general environmental lexicon (GEL) and assess the extent to which we can set it apart from the others. The general intuition on which this research is based is that the GEL is both well-distributed in a specialized corpus (criterion 1) and specific to this type of corpora (criterion 2). The corpus used in the current experiment, made of 6 subcorpora that amount to 4.6 tokens, was compiled manually by terminologists for different projects designed to enrich a terminological resource. In order to meet criterion 1, the distribution of the GEL candidates is evaluated using a simple and well-known measure called *inverse document frequency*. As for criterion 2, GEL candidates are extracted using a term extractor, which provides a measure of their specificity relative to a corpus. Our study focuses on single-word lexical items including nouns, verbs and adjectives. The results were validated by a team of 4 annotators who are all familiar with the environmental lexicon and they show that using a high specificity threshold and a low idf threshold constitutes a good starting point to identify the GEL layer in our corpora.

Keywords: terminology, lexical layers, term extraction, corpora, environment

1. Introduction

It is generally recognized that specialized texts comprise three main lexical layers: 1. terminology (the lexicon used to express domain-specific knowledge); 2. general language (the lexicon used by all speakers of a language and that is likely to be found in any kind of texts); and 3. a layer that lies in-between that will be called herein the *transdisciplinary lexicon* (Drouin, 2007; Tutin, 2008; Hatier, 2016)[1]. We believe that in very large domains, such as the environment that encompasses a broad variety of topics (climate change, sustainable development, renewable energy, water pollution, etc.), the terminology (defined above as the "domain-specific lexicon") further divides into two layers. The first layer of the lexicon is topic specific. For instance, terms such as *chlorination* or *marine turbine* are specific to water pollution and renewable energy respectively. The second layer of the lexicon cuts across the entire field of the environment: e.g. *ecosystem*, *sustainable*, *energy*, *development*, etc. We would thus obtain four different lexical layers in specialized texts, as shown in Figure 1.

In given applications (such as terminology resource compilation for which the method proposed in the paper is investigated)[2], identifying items that belong to one layer or the other can be quite difficult. For example, when working with a general environment corpus (such as PANACEA[3] that covers a wide range of topics), some topic specific terminology might be difficult to spot since the corpus will cover several specialized topics related to the overall domain. In contrast, when working with topic specific corpora, some general domain terminology might not be perceived as such since the corpus does not offer broad view of the subject.

For the time being, compilers of resources make decisions based on their intuition, but this can lead to choices that differ from one compiler to another. Furthermore, specialized resources are not necessarily enriched by experts of a domain (in fact, they seldom are). So making fine-grained distinctions between topic specific, general specialized or transdisciplinary lexica can soon become a quite challenging task.

This paper proposes a method for identifying one of the layers mentioned above, i.e. the general environmental lexicon (GEL). In the process, however, we will need to distinguish this lexicon from topic specific lexica, on the one hand, and from the transdisciplinary lexicon, on the other hand.

The general intuition on which this research is based is that the GEL is both well-distributed in a specialized corpus (criterion 1) and specific to this type of corpora (criterion 2). In order to meet criterion 1, distribution of the GEL candidates is evaluated using a simple and well-known measure called inverse document frequency. As for criteria 2, GEL candidates are extracted using a term extractor, which provides a measure of their specificity relative to a corpus.

2. Related Work

Different methods were devised to identify terminology and the transdisciplinary lexicon. Regarding term extraction, methods are now well established and used for different applications (Indurkhya and Damerau, 2010). An efficient method consists of comparing a domain-specific

[1] Other names for this specific layer can be found in the literature: e.g., academic vocabulary (Coxhead, 2000; Paquot, 2014)

[2] There are other applications for which distinguishing lexical layers is important: specialized translation, language teaching, for instance.

[3] `http://catalog.elra.info/productinfo.php?products_id=1184`, ELRA-W0063

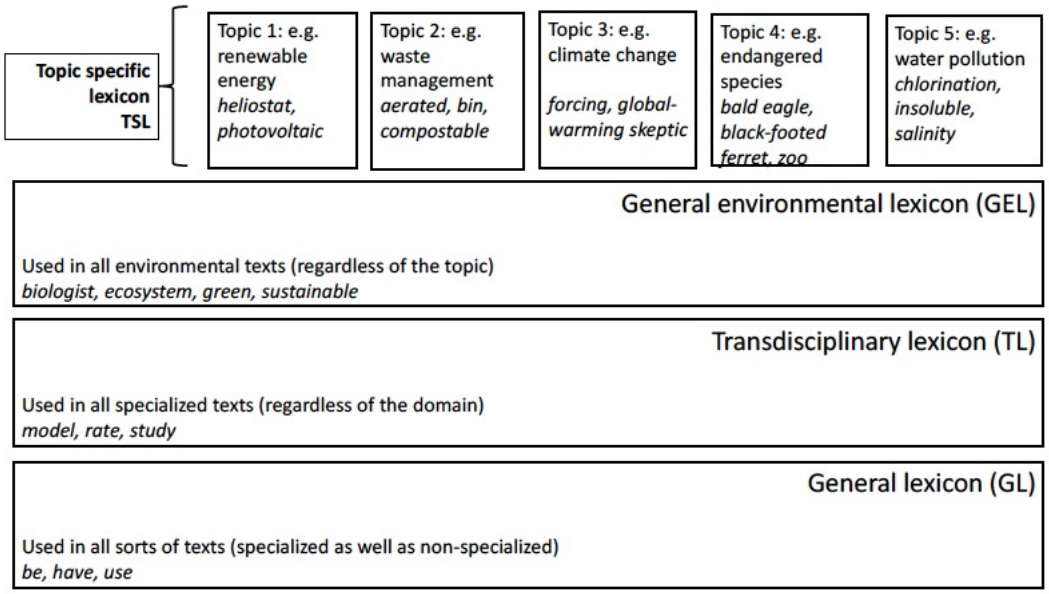


Figure 1: Lexical layers in environmental texts.

corpus to a general one and computing a specificity score[4] of lemmas. For instance, a corpus of environmental texts can be compared to a general balanced corpus such as the British National Corpus. This method was implemented in TermoStat developed by (Drouin, 2003). It was evaluated for the extraction of single-word terms with satisfactory results (Lemay et al., 2005) and supports multiple languages[5]. The concept of "specificity" aims to capture the potential of term candidates to behave like terms (termhood, see (Kageura and Umino, 1996)). In most cases, termhood is linked to a higher than expected frequency in a specialized corpus based on a theoretical frequency computed from a general corpus. Various statistical measures can be used to compute specificity. Such an approach gives us access to the topic specific layer (TSL).

Over the years, methods have also been developed for the identification of the transdisciplinary lexicon (TL) (Drouin, 2007; Tutin, 2008; Hatier, 2016). This second set of lexical items can also be identified with corpus comparison with a general corpus in order to identify this lexical layer. In such a case, however, the corpus that is analyzed should cover various disciplines and be composed of several topic specific corpora such as physics, chemistry and linguistics. Previous work has shown that identifying the transdisciplinary lexicon raises challenges due to different factors such as polysemy of lexical items, interference with other layers.

Since term extraction techniques are targeted at the identification topic specific lexical items solely, they cannot be used as-is and they have to be slightly modified. For our proposed task, the strategy used to identify TL lexical items cannot be considered either as we have a corpus covering one domain, namely the environment. What we need is a technique that can capture that fact that the GEL lexical items are both related to the overall topic of a corpus (thus semantically close to TSL items) and transdisciplinary as far as the overall topic of the corpus is concerned (from this point of view, their behavior bears some similarities with TL items).

Our hypotheses for the current task are that:

1. Lexical items of the TSL should be associated with high specificity measures when compared to a balanced general reference corpus as they are characteristic of the overall subject area. Furthermore, TSL members should have a low distribution across the corpus since topics are addressed in subcorpora.

2. Lexical items that belong to the GEL should also be associated with a high specificity measure when compared to a balanced general reference corpus on the one hand. On the other hand, they should have a large distribution across different subcorpora since they are associated with the environment as an overall domain.

3. Members of the TL should have lower specificity levels as the TLS items since they also occur on a regular basis in a balanced general reference corpus. We expect them, as demonstrated in prior studies, to be highly distributed across the corpus.

4. Common words, or lexical units of the General Lexicon should be both distributed in the corpus and have low specificity values.

3. Method

Our method aims to identify the GEL (2. above). In order to do so, we will apply two criteria designed to model our hypotheses: the first, criterion 1, aims to capture distribution; the second, criterion 2, captures specificity. Distribution is evaluated on the specialized corpus while specificity computation requires that we use two corpora: a general bal-

[4]The concept of *specificity* used in this paper differs from the usage of the nearby concept in the medical context where it is used as a measure of false positive rate.

[5]`http://termostat.ling.umontreal.ca`

anced corpus and a specialized corpus. Figure 2 illustrates the overall process used to reach our goals.

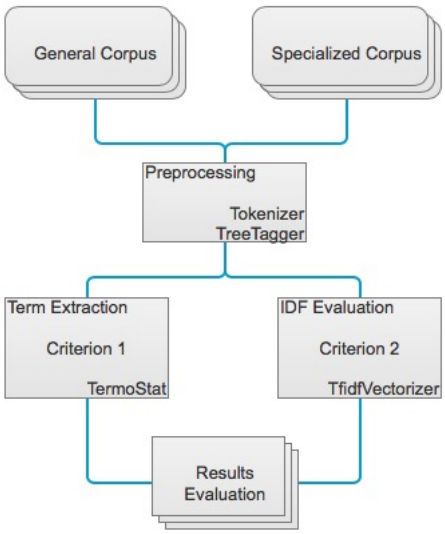


Figure 2: Overview of the method to identify the general environmental lexicon (GEL).

The following sections detail our experimental setup, including measures, tools and the annotation process and scheme.

4. Experimental Setup

4.1. Corpus Data

4.1.1. Specialized Corpora

The specialized corpora used in the current experiment were compiled manually by terminologists for different projects designed to enrich a terminological resource (DiCoEnviro) (L'Homme, 2018). Table 1 gives an overview of the subcorpora combined in order to build our specialized corpus.

Subcorpora	Number of tokens
Climate Change	607,233
Endangered Species	1,276,304
Renewable energy	776,838
Transportation Electrification	747,389
Waste management	626,039
Water pollution	586,849
Total	4,620,652

Table 1: Size of the subcorpora.

4.1.2. General Corpus

The general reference corpus used was built from subsets of two large corpora: the British National Corpus (BNC) (Consortium, 2007) and the American National Corpus (ANC) (Reppen et al., 2005). We extracted 4M tokens from each of these corpora in order to compile our 8M tokens reference corpus.

4.2. Corpus Preprocessing

Basic preprocessing was applied to both the specialized and the reference corpora, which included extracting the text from the XML files that comprise the corpus, replacing non-ASCII characters with ASCII equivalents and tokenizing. The corpora are then tagged and lemmatized using TreeTagger (Schmid, 1994).

4.3. Term Extraction and Specificity Evaluation

Terms[6] were extracted using a modified version of TermoStat (Drouin, 2018) in order to use a general reference corpus designed for this specific experiment. The extraction process was limited to single-word lexical items including nouns, verbs and adjectives.

As mentioned, TermoStat computes a *Specificity* score to represent how far the frequency in the specialized corpus deviates from a theoretical frequency. In order to do so, a measure proposed by Lafon (1980) is used.

	Reference Corpus	Specialized Corpus	Total
Freq. term	a	b	a+b
Freq. of other words	c	d	c+d
Total	a+c	b+d	N=a+b+c+d

Table 2: Contingency table of frequencies.

Using values from Table 2, specificity can be calculated as follows:

$$\log P(X=b) = \log (a+b)! + \log (N-(a+b))! + \log (b+d)! + \log (N-(b+d))! - \log N! - \log b! - \log ((a+b)-b)! - \log ((b+d)-b)! - \log (N-(a+b)-(b+d)+b)!$$

This measure has been tested in previous studies (Lemay et al., 2005; Drouin and Langlais, 2006; Drouin, 2006; Drouin and Doll, 2008) and leads to excellent results for both the extraction of single-word terms and multi-word terms. Specificity allows identifying forms that are both over- and under- represented in a corpus. In the case of terminology, a domain- and genre-oriented lexicon, we are solely interested in positive specificities which correspond to forms that over-represented.

Although it is a common practice when dealing with domain-specific units to extract multi-word terms and especially multi-word nouns, we apply criteria that are more compatible with lexicography. Hence, items such as *climate, pollute, green* and *greenhouse effect* are considered as terms; expressions such as *climatic impact* and *renewable energy* are considered as compositional collocations. Since most multi-word expressions are compositional in specialized corpora, it is much more productive for terminologists in our projects to work with lists of single-word lexical items. The drawback of this method is, of course, to

[6]We are using *term* here to describe the output of the term extractor. In fact, this output will encompass both topic-specific lexical items and GEL members.

potentially raise more difficulties when trying to separate the lexical layers to which we refer in the present paper.

Since the specificity scores cannot be represented on a predefined scale, we expressed them on a scale ranging from 0 to 100 where the max specificity score is mapped to 100. This mapping leads to a less granular representation of the scores and a more flexible set of scores to assess.

4.4. Inverse Document Frequency Evaluation

In order to evaluate the distribution of the GEL candidates we used the simple and well-known measure called *inverse document frequency* (Sparck Jones, 1972). This measure returns lower scores for tokens that occur very frequently in a document set, and contrariwise higher scores for tokens that occur rarely. To compute idf, we used its Python implementation (TfidfVectorizer) from the Python scikit-learn library. For our study, default values were used and sentences were considered as documents. As with the previous measure, idf scores were also mapped on a scale of 0 to 100. However, in the case of idf, we reverse the score so that the most "interesting" GEL candidates for our study receive a higher idf. This modification was applied to make the scoring results more intuitive for the team of annotators.

4.5. Annotation of results

4.5.1. Result sampling

Since the volume of GEL candidates identified was too large for our team to proceed to a complete validation, we resorted to a sampling mechanism. In order to do so, we broke down both the idf and the specificity scores in groups of 10 ranging from 0 to 100. The results were then sorted by decreasing order of idf and decreasing specificity scores providing us with a matrix of results of size 10x10. The lower left corner corresponds to a mapped idf score of 0-9 and a mapped specificity score in the same range. At the opposite side, the upper right corner of the matrix contains GEL candidates with mapped idf and specificity scores of 90-100. From each of the cell of the matrix, we sampled a maximum of 15 GEL candidates, which means we could evaluate a theoretical maximum number of 1,500 GEL candidates. In fact, since not all cells contain 15 candidates, our process led to a total of 522 GEL candidates to be evaluated.

4.5.2. Annotation team

A team of 4 annotators who are all familiar with the environmental lexicon were responsible for carrying out the annotation process. They have varying experience in enriching a terminological resource that contains terms related to the different topics mentioned in Table 1.

4.5.3. Annotation guidelines

Since the task given to annotators was to single out the GEL – and thereby distinguish it from the TSL, on the one hand, and from the TL, on the other – annotators held a discussion to agree on a definition for each lexical level. They also defined very broad classes of terms that in their opinion are relevant for characterizing the GEL:

- Related to nature (*ecosystems, species*)
- Related to Earth and to its subdivisions (*ocean, continent, hemisphere*)
- Human impact on nature and human activities (*agriculture, activity, defiorest*)
- Products made by humans; things produced by humans (*chemical, waste*)
- Greenhouse gases and related concepts (*carbon, methane, emit*)
- Pollution and contamination (*contaminated, pollutant*)
- Climate/weather and meteorological events (*cyclone, extreme*)
- Protection and conservation (*endangered, protect*)
- General scientific domains (*biology, chemistry*) and experts (*biologist*)

Afterwards each annotator proceeded to validate the list of candidates separately. They could use different resources (terminological databases and corpora), but they could not consult each other during the validation process.

4.5.4. Annotation scheme

In order to obtain optimal results, we decided to use a simple annotation scheme where annotators classified GEL candidates in four different categories represented by a single letter. Keeping in mind that the "good" candidates are those that belong to the GEL), our scheme includes:

- B: the candidate is part of the GEL. *energy, emission, temperature, water, waste*
- M: the candidate is not part of the GEL. *cell, high, include, show, year*
- I: the candidate is part of the vocabulary of the environment; however, the annotator hesitates to classify it as topic specific or as part of the GEL. *model, range, turbine, wave, wind*
- P: the candidate is not valid. *bacterium, recharg, semi, specie, trolleybuses*

All GEL candidates proposed to the annotators had to be classified using the 4 previous codes. The P code is used to classify all forms that are mainly related to tokenizing errors and NLP errors (for example, erroneous part-of-speech tagging). Items classified using the M code could either be members of TSL, TL or general language (GL) and relevant for the current study which is solely focused on GEL.

5. Results and evaluation

5.1. Results

The extraction process on our 4.6M word specialized corpus led an impressive amount of GEL candidates. Table 3 gives an overview of the results broken down by part-of-speech.

Part of speech	Number of GEL candidates
Nouns	11,725
Adjectives	4,817
Verbs	1,722
Total	18,265

Table 3: Number of GEL candidates by part-of-speech.

5.2. Inter-annotator agreement results

The inter-annotator agreement was evaluated using a free online tool (Geertzen, 2012), which provides both the Fleiss kappa (Fleiss and others, 1971) and the Krippendorff's alpha (Krippendorff, 2004) scores (See Table 4). Detailing these measures is beyond the scope of this paper, but both measures consider pairwise agreement of the annotators.

Fleiss	Krippendorff
A_obs = 0.797	D_obs = 0.203
A_exp = 0.471	D_exp = 0.529
Kappa = 0.616	Alpha = 0.616

Table 4: Inter-annotator agreement.

Although both scores indicate that our annotators are not in total agreement, they lead us to believe that the agreement level is nevertheless fairly high.

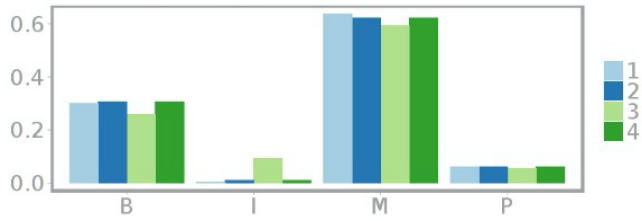


Figure 3: Inter-rater agreement evaluation.

Figure 3 clearly shows that agreement is higher for items that are not part of the GEL (M in Figure 3). We can also note that one of the annotators (the more experienced one) had more problems classifying some candidates than others ("I" in Figure 3). This is an interesting fact and it leads us to believe that more experienced annotators might be more cautious in their classification process.

In order to assess the suitability of the indices to identify the lexical items that interested us, we measured the accuracy of each index for a group of specificity and idf scores. Precision is usually defined as the fraction of relevant instances among the retrieved instances. In other others words, in our case, it corresponds to the number of GEL entries in each group compared to the total of entries in each group.

Figure 4 indicates that the specificity scores are useful to identify terminologically interesting lexical items. However, for our current goal, which is to identify GEL entries, the usefulness of this measure is mitigated by the fact that valid candidates are scattered throughout the score range. This is in line with our hypotheses that specificity scoring

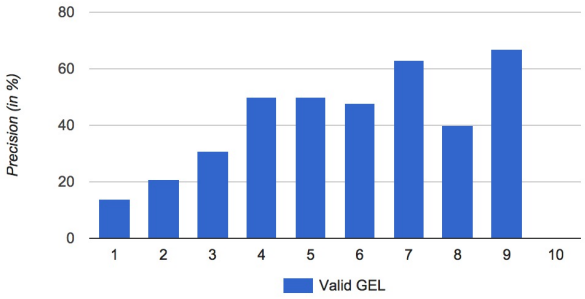


Figure 4: Precision for each group of specificity scores.

cannot, by itself, allow to identify precisely GEL entries from a list of candidates.

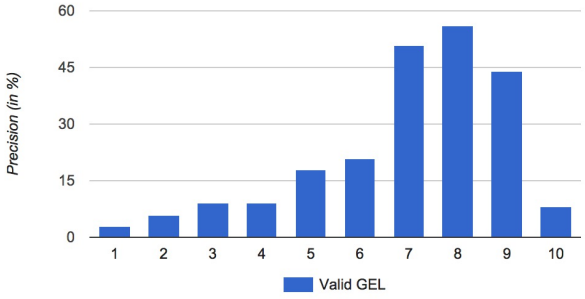


Figure 5: Precision for each group of idf scores.

In order to complete the information provided by the specificity scores, we resorted to using idf. As Figure 5 shows, higher distribution (higher values in our figure correspond to lower original idf scores) is obviously linked to the identification of valid GEL entries in a list of candidates. This observation is also in line with our initial hypotheses. The heatmap in Figure 6 combines both scores in the 10x10 matrix used for the sampling and evaluation. Some of the cells of the matrix contained no candidate and are thus empty (light green). All non-empty cells contain a precision score and are color-coded: red cells have a precision of 0 while green cells have various levels of precision with higher precision levels being darker.

As one can see in Figure 6, most of our candidates are distributed in cells 1-7 for the specificity score and 2-9 for the idf score. Our results show that our valid GEL items are mainly located in the range of specificity 4-9 and the idf 7-10. The relation between higher specificity scores and idf scores can be clearly seen as higher idf scores[7] allow to complete the information provided by specificity.

Figure 7 contains the details of the precision measures for Figure 6. Each cell where data was retrieved shows a ratio of the number of valid GEL items over the number of items in the same group. As can be observed, higher specificity leads to a lower number of candidates while the same observation cannot be made about idf. Restricting our results to high specificity (4-9) and high idf (7-10) values would mean discarding quite a few valid GEL items (88 total). On

[7] We need to remind our readers that our idf scores are reversed from the original idf measure. See section 3.3.2

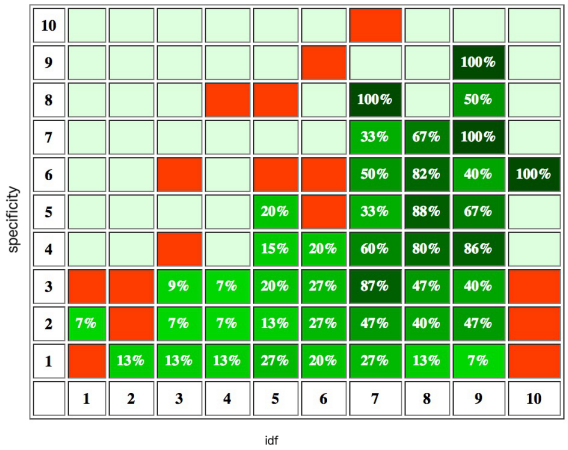


Figure 6: Specificity - idf heatmap - precision.

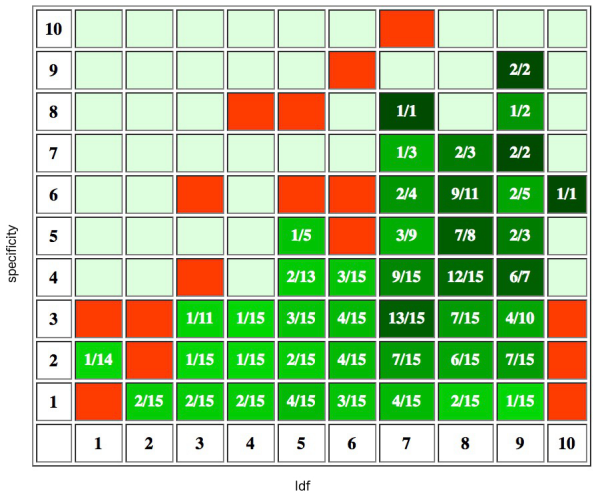


Figure 7: Specificity - idf heatmap - ratio.

the other hand, this would mean that we can obtain a precision of 68% for the same area of the matrix above, which is an interesting performance. Our specificity measure seems to consider far too few GEL items as being specific to our environment corpus.

6. Future Work

Although we limited our investigation to single-word lexical items for the current project, it could be easily applied to multi-word lexical items. In fact, this is not in itself a limitation our approach as much as a methodological decision on our part based on the terminological work being done in our research group. One avenue that could be explored is to measure the impact or the benefit of taking into consideration multi-word lexical items on the validation process.

As could be seen from the inter-annotator evaluation, the annotators seem to strongly agree on what is and what is not a valid GEL item. This was a surprising result given the difficulty of the task and the overlap between lexical that is often assumed by researchers. We would like to investigate what led to that strong agreement in order to see if an algorithm could somehow capture this knowledge. If so, it could be built into further experiments so as to increase precision and complement the method reported in this paper. Idf scores allow us to capture the behaviour of the GEL items adequately while the specificity scores do not seem to be a good indicator as valid forms are scattered throughout the specificity groups. Using a different measure to model the concept of specificity might lead to better results.

Our validation process was carried out using a sample of 15 GEL candidates taken from each cell of our 10x10 matrix. Using a larger number of candidates from each cell might allow us to observe more accurate precision levels. The method was tested on corpora linked to the domain of the environment, a domain that is quite unique since it encompasses a wide variety of topics. An interesting extension would be to test our method with corpora from other domains and see if we can obtain similar results. We will also devise a methodology for implementing this method (in this form or in a modified version) in the compilation process of terminological resources.

7. Conclusion

In this paper we proposed a method to automatically distinguish terminologically relevant lexica in the subject area of the environment. More specifically, we devised a technique to identify the general environmental lexicon (GEL) and distinguish it from other lexical layers that co-exist in specialized corpora. Our basic hypotheses were that lexical items from the GEL were both very specific to our environmental corpus and distributed evenly throughout the same corpus. In order to verify these hypotheses, we used a term extractor relying on the specificity score proposed by Lafon (1980) (criterion 1), and a reversed standard idf measure to quantify the distribution of GEL candidates (criterion 2). Our results validated our hypotheses to a large extent and that candidates with both a higher specificity level and a higher distribution tend to be lexical items of the GEL.

8. Acknowledgements

This work was supported by the Social Sciences and Humanities Research Council (SSHRC) of Canada.

We also would like to thank the annotators who contributed to the validation of the GEL candidates.

9. Bibliographical References

Coxhead, A. (2000). A new academic word list. *TESOL Quarterly*, 34(2):213–238.

Drouin, P. and Doll, F. (2008). Quantifying termhood through corpus comparison. In *Terminology and Knowledge Engineering (TKE-2008)*, pages 191–206, Copenhague, Danemark, Août. Copenhagen Business School, Copenhagen, Copenhagen Business School, Copenhagen.

Drouin, P. and Langlais, P. (2006). Évaluation du potentiel terminologique de candidats termes. In *Actes des 8es Journées internationales d'Analyse statistique des Données Textuelles. (JADT 2006)*, pages 389–400, Besançon, France.

Drouin, P. (2003). Term extraction using non-technical corpora as a point of leverage. *Terminology*, 9(1):99–115.

Drouin, P. (2006). Termhood experiments: quantifying the relevance of candidate terms. *Modern Approaches to Terminological Theories and Applications*, 36:375–391.

Drouin, P. (2007). Identification automatique du lexique scientifique transdisciplinaire. *Revue française de linguistique appliquée*, 12(2):45–64.

Fleiss, J. et al. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Geertzen, J. (2012). Inter-rater agreement with multiple raters and variables. `https://nlp-ml.io/jg/software/ira/`. September 28, 2017.

Hatier, S. (2016). *Identification et analyse linguistique du lexique scientifique transdisciplinaire. Approche outillée sur un corpus d'articles de recherche en SHS*. Ph.D. thesis. Thèse de doctorat dirigée par Tutin, Agnès Sciences du langage Spécialité Informatique et sciences du langage Grenoble Alpes 2016.

Indurkhya, N. and Damerau, F. (2010). *Handbook of Natural Language Processing, Second Edition*. Chapman & Hall/CRC machine learning & pattern recognition series. CRC Press.

Kageura, K. and Umino, B. (1996). Methods of automatic term recognition: A review. *Terminology*, 3(2):259–289.

Krippendorff, K. (2004). *Content Analysis: An Introduction to Its Methodology*. Sage.

Lemay, C., L'Homme, M.-C., and Drouin, P. (2005). Two methods for extracting specific single-word terms from specialized corpora: Experimentation and evaluation. *International Journal of Corpus Linguistics*, 10(2):227–255.

Paquot, M. (2014). *Academic Vocabulary in Learner Writing: From Extraction to Analysis*. Corpus and Discourse. Bloomsbury Publishing.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Tutin, A. (2008). Sémantique lexicale et corpus : l'étude du lexique transdisciplinaire des écrits scientifiques. *Lublin Studies in Modern Languages and Literature*, (32):242–260.

10. Language Resource References

BNC Consortium. (2007). *British National Corpus, version 3 BNC XML edition*. British National Corpus Consortium, ISLRN 143-765-223-127-3.

Drouin, Patrick. (2018). *TermoStat 3.0*. http://termostat.ling.umontreal.ca.

L'Homme, Marie-Claude. (2018). *DiCoEnviro : Le dictionnaire fondamental de l'environnement*. http://olst.ling.umontreal.ca/dicoenviro.

Reppen, Randi and Ide, Nancy and Suderman, Keith. (2005). *American National Corpus (ANC) Second Release*. Linguistic Data Consortium, ISLRN 797-978-576-065-6.

Linking, Searching, and Visualizing Entities in Wikipedia

Marcus Klang, Pierre Nugues
Lund University, Department of Computer science,
Lund, Sweden
marcus.klang@cs.lth.se, pierre.nugues@cs.lth.se

Abstract

In this paper, we describe a new system to extract, index, search, and visualize entities in Wikipedia. To carry out the entity extraction, we designed a high-performance, multilingual, entity linker and we used a document model to store the resulting linguistic annotations. The entity linker, HEDWIG, extracts the mentions from text using a string matching engine and links them to entities with a combination of statistical rules and PageRank. The document model, Docforia (Klang and Nugues, 2017), consists of layers, where each layer is a sequence of ranges describing a specific annotation, here the entities. We evaluated HEDWIG with the TAC 2016 data and protocol (Ji and Nothman, 2016) and we reached the CEAFm scores of 70.0 on English, on 64.4 on Chinese, and 66.5 on Spanish.

We applied the entity linker to the whole collection of English and Swedish articles of Wikipedia and we used Lucene to index the layers and a search module to interactively retrieve all the concordances of an entity in Wikipedia. The user can select and visualize the concordances in the articles or paragraphs. Contrary to classic text indexing, this system does not use strings to identify the entities but unique identifiers from Wikidata. A demonstration of the entity search and visualization will be available for English at this address http://vilde.cs.lth.se:9001/en-hedwig/ and for Swedish at: http://vilde.cs.lth.se:9001/sv-hedwig/.

Keywords: named entity recognition, entity linker, wikipedia

1. Introduction

Wikipedia has become a popular NLP resource used in many projects such as text categorization (Wang et al., 2009), information extraction, question answering (Ferrucci, 2012), or translation (Smith et al., 2010). In addition to its size and diversity, Wikipedia, through its links, also enables to create a graph that associates concepts, entities, and their mentions in text. Wu and Weld (2010), for instance, used the "wikilinks", the Wikipedia hyperlinks, to collect the mentions of an entity and build sets of synonyms for an open information extraction system.

However, according to the edition rules of Wikipedia, only the first mention of an entity should be linked in an article. An automatic wikification is then necessary to associate the subsequent mentions with an entity (Mihalcea and Csomai, 2007). In addition, searching entities using names in the form of strings can be tricky as names are sometimes ambiguous and entities may have more than one name. Finding all the occurrences of an organization like the United Nations would require five or more queries as they can be mentioned not only as *the United Nations*, but also as: *UN*, *U.N.*, *United Nations Organization*, *UNO*, etc.

In this paper, we describe a novel multilingual system to process, index, search, and visualize all the mentions of an entity in Wikipedia. This system consists of an entity linker, HEDWIG, that extracts the mentions from text using a named-entity recognition engine and links them to entities with a combination of statistical rules and PageRank. We applied HEDWIG to the whole collection of English and Swedish articles of Wikipedia. We then used Lucene to index the layers and a search module to interactively retrieve all the concordances of an entity in the articles, paragraphs and metadata. The user can then select a concordance s/he wants to visualize. As opposed to the Wikipedia index, the system uses unique identifiers to index the entities and not their mentions, which enables the users to carry out more easily exhaustive searches.

2. Previous Work

Most named entity linkers adopt a two-step procedure, where they first identify the mentions and then link them to a unique identifier.

2.1. Mention Detection

The mention detection step, or spotting, has been addressed by a variety of techniques. Mihalcea and Csomai (2007) used a dictionary associating the entities with their surface forms, where the surface forms are simply n-grams. They extracted all the strings in a text that matched any of the surface forms in the dictionary to produce the set of mention candidates. As the candidates may overlap, the authors ranked them using a *keyphraseness* metric defined as the number of documents, where the mention was linked divided by the number of documents, where the mention occurred. They set the number of mentions to keep to 6% of the total number of words in the document following figures they observed in Wikipedia.

Milne and Witten (2008) also used a dictionary of surface forms as well a classifier to decide on the mentions to keep. They trained the classifier on Wikipedia mentions, either linked, the positive examples, or nonlinked, the negative ones. As features, they used the link probability (keyphraseness), relatedness, disambiguation confidence, generality, location, and spread.

Lipczak et al. (2014) used the Lucene's finite state transducers and Solr Text Tagger to detect the mentions. They collected the surface form dictionary from Wikipedia as well as Freebase and Google's wikilinks. The tagging step results in an overdetection that is pruned using lexical filters. The final selection of mentions is carried out in the linking step.

Cucerzan (2014) used a dictionary of surface forms collected from Wikipedia, anchor text, page titles, redirection pages, etc, and a set of rules to identify the mentions in the text. As in Lipczak et al. (2014), the overgeneration is

solved at the linking stage.
Piccinno and Ferragina (2014) used a dictionary of surface forms similar to Cucerzan (2014) to spot the mentions. They also used a pruner to discard unlikely annotations based on a classifier and a coherence metric with the set of neighboring entities. This final selection is done at linking time.
Sil et al. (2015) used classifiers based on neural nets and conditional random fields trained on three languages.
Some annotators also used an external named entity recognition module to carry out this mention detection as AIDA (Hoffart et al., 2011) and Tan et al. (2015) that used Stanford NER (Finkel et al., 2005).

2.2. Entity Linking

Bagga and Baldwin (1998) is one of the earliest works that introduced the notion of linkage to unique things through the task of cross-document coreference. The main difference with entity linking is that predefined lists of entities do not exist but have to be found. Bagga and Baldwin (1998) created summary vectors and tried to cluster them to form linkages. These summary vectors were created from noun phrases contained within coreference chains in documents. Using cosine similarity with a predefined threshold, they were able to cluster coreferences that crossed the document boundaries.
Bunescu and Pasca (2006) first explored entity linking using Wikipedia as knowledge base. They used hyperlinks, redirects, disambiguation pages, and the category hierarchy, which would be used by almost every major paper since. Using context article similarity based on 55-word window vector space model (VSM) cosine similarity and a taxonomy kernel, they trained SVM models to recast the disambiguation problem as a classification. They reported accuracies between 55.4% and 84.8% depending on which model they used.
Cucerzan (2007a) introduced clearly defined end-to-end pipelines – starting with text and ending with linked entities – as well as a notion of collective agreement in the disambiguation component. Using a *document vector* comprised of surface form context, entity context, and categories, he could maximize an agreement between the proposed entity candidates. Using the top two stories from 10 MSNBC news categories in January 2, 2007, he reported an accuracy of 91.4% versus 88.3% from 350 random Wikipedia pages.
Milne and Witten (2008) introduced important concepts such as relatedness and commonness which still defines a strong baseline used by many following papers in one form or another.
Hoffart et al. (2011) used an ensemble system to compute a linear combination of entity probabilities, context similarities, and entity coherences, where the *popularity prior* corresponds to the number of in-links to a Wikipedia entity; the *context similarity* compares the context of the input by computing a similarity between all the tokens in the input against a key phrase defined for entities they extracted from YAGO. A key phrase is a phrase that is derived from link texts, category names, citation titles, and other references; finally, *coherence* provides a way of comparing different entity candidates in a text in order to measure how compatible they are.
Lipczak et al. (2014) built a set of all the entity candidates for all the mentions in a document. They started from an entity core corresponding to the default senses. Using this core, they built a topic centroid from Wikipedia categories and discard entities from the core that are outside the topic. They finally refined the core and rank the remaining entities using a cosine similarity.
Eckhardt et al. (2014) built a graph of entity-mention pairs, where they weighted the edges with $P(E|M)$ probabilities. They applied a variant to PageRank to rank the entities.
Sil et al. (2015) described a trilingual system that uses a classifier with features such as the number of mention–entity matches in Wikipedia, acronym match, pointwise mutual information between entities and categories, etc.
Tan et al. (2015) used a graph of entity-mention and entity-entity edges. The edges are weighted by a function of the context similarity between a mention and an entity description in Freebase and functions of relatedness and context similarities. The entity ranking is eventually determined by a random walk in the graph.
Cucerzan (2007b) and Han and Zhao (2009) described other algorithms for NERL. In contrast to most of these previous works, multilingual support is at the core of HEDWIG.

3. Extraction of the Wikipedia Structure

Before we apply the linker to Wikipedia, we convert the HTML pages into a multilayer document model; see Sect. 5. This preprocessing step parses the HTML documents into DOM trees and extracts the original page structure, text styles, links, lists, and tables. We then resolve all the Wikipedia links to unique Wikidata identifiers, where Wikidata is an entity database, which assigns unique identifiers across all the language editions of Wikipedia.
The United Nations, for instance, has the unique id: Q1065, which enables to retrieve the article pages in English, French, Swedish, or Russian. Figure 1 shows examples of these ids in the *United Nations* article from the English Wikipedia, where we have replaced the manually set Wikipedia anchors (the wikilinks) with their Wikidata numbers: Q245065 for *intergovernmental organization* and Q60 for *New York City*. Figure 2 shows the first paragraph of the corresponding article from the Swedish Wikipedia, *Förenta nationerna* 'United Nations', where *mellanstatlig organisation*, the Swedish word for intergovernmental organization, has also the Q245065 number.

4. Entity Linking

Once we have collected and structured the text, we apply the entity linking module to find all the mentions of an entity in text and link these mentions to a unique identifier.

4.1. Set of Entities

We used the wikilinks to build a repository of (mention, entity) pairs and Wikidata as the nomenclature for the unique entity identifiers. We collected all the wikilinks in the Wikipedia articles, where each link consists of a label and the name of the destination page:

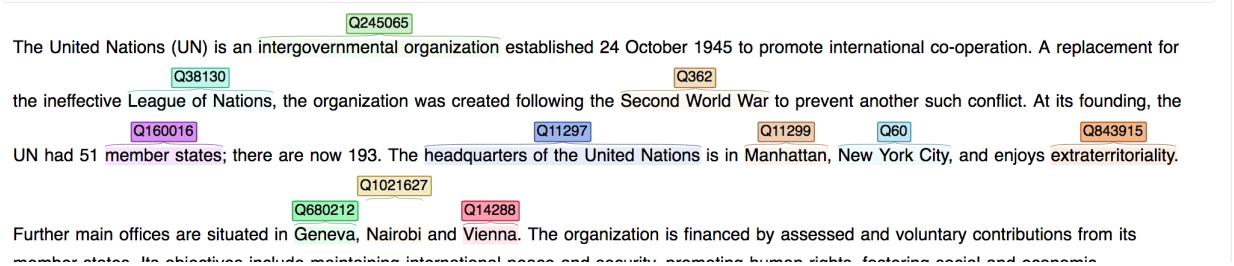


Figure 1: Visualization of anchors with Wikidata Q-numbers. The first lines of the *United Nations* article in the English Wikipedia

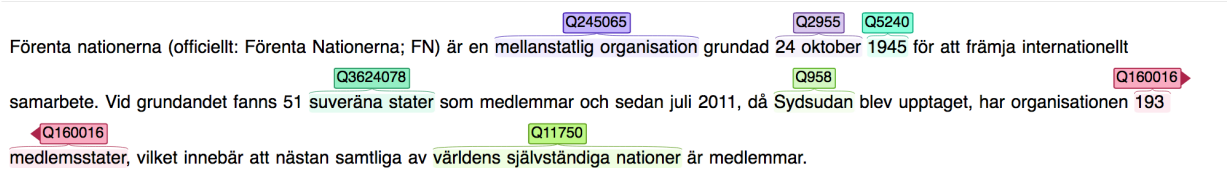


Figure 2: Visualization of anchors with Wikidata Q-numbers. First paragraph of the *Förenta nationerna* 'United Nations' article in the Swedish Wikipedia

`[[destination|label]]`. We parsed these links into (mention, entity page) pairs and we translated the entity pages into Wikidata Q-numbers.

We annotated each mention-entity pair with a set of properties: its frequency, its frequency relative to the mention, $P(E|M)$, if the mention is in a dictionary, if the mention consists of stop words. We then pruned the knowledge base from unique mentions for entities with a high frequency, mentions that are only stop words, etc.

During the mention gathering, we also derived statistics for a given language. Before we computed these statistics, we applied a procedure that we called *autolinking*. In an article, the Wikipedia guidelines advise to link only one instance of an entity mention[1]: Normally the first one in the text. With the autolinking procedure, we link all the remaining mentions provided that we have sequences of exactly matching tokens.

The statistics we collect are:

- The frequency of the mention string over the whole Wikipedia collection (restricted to one language);
- The frequency of the pair (mention, entity) that we derive from the links without autolinking (only manually linked mentions);
- The count of (entity1, entity2) pairs in a window corresponding to a paragraph and limited to 20 linked mentions. This is carried out after autolinking;
- Capitalization statistics for all the tokens: We extract token counts for all tokens with a frequency greater than 100 and we break them down by case properties: uppercased, lowercased, titlecased, and camelcase;

[1] `https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Linking#Overlinking_and_underlinking`

4.2. Mention Recognition

To detect the mentions in an unannotated text, we use a two-step procedure: We first generate the mention candidates using a finite-state transducer; this results in a very large overgeneration. We then apply a mention segmenter that identifies the mentions to keep for the linking phase.

Following Lipczak et al. (2014) and Södergren and Nugues (2017), we used an automaton to spot the mentions. This automaton uses Lucene's finite-state transducers and is efficient in terms of memory usage and execution time. Depending on the language and the availability of manually-annotated data, we can complement this candidate generation with two named-entity recognition systems trained on the annotated data: The first one being based on an extension of the fixed-size ordinally forgetting encoding (FOFE) technique (Xu et al., 2017; Zhang et al., 2015) and the second one being CoreNLP (Manning et al., 2014).

The overgeneration of mention candidates impairs the quality of the downstream linker. To discard the very unlikely ones, we introduced rules based on the frequency of the manual links applied to mention M and its link probability lp. We denote M_{linked} a mention with a manual hyperlink; this would correspond to the wiki markup: `[[link|mention]]`, and $M_{autolinked}$, an autolinked mention. We define:

$$\begin{aligned} lp(M) &= P(M_{autolinked}|M) \\ &= \frac{\#M_{autolinked}}{\#M_{autolinked} + \#M_{unlinked}}, \end{aligned}$$

where $\#M_{autolinked}$ is the number of times a mention is linked in the Wikipedia collection and $\#M_{unlinked}$, its frequency when unlinked.

The rules are:

1. Remove the mentions M where $lp(M) < D_{lp}$, for instance with $D_{lp} = 0.01$;

2. Keep the mentions where $lp(M) > K_{lp}$, and $\#M_{linked} > K_f$, with, for instance, $K_{lp} = 0.15$ and $K_f = 25$. All these mentions are candidates for the linking step;

3. Set the rest in a dubious set.

4.3. The Linking Step

We applied the JUNG implementation of PageRank (Brin and Page, 1998; O'Madadhain et al., 2003) to the tagged mentions. Following Eckhardt et al. (2014) and Södergren and Nugues (2017), we created a node for every mention-entity pair that is detected in the text and we ran PageRank on this graph; we used the JUNG default settings.

We analyzed the internal links of Wikipedia to determine the entities that appear in the same context. Two entities are linked if the article of Entity A links to the article of Entity B or there exist at least one link to the article of Entity A and another one to the article of Entity B occurring in the same paragraph.

We then re-ranked the PageRank candidates using a feed forward neural network consisting of three layers with RELU activations, a crossentropy loss, and a sigmoid output. We trained the model on the output of the PageRank disambiguator applied to the TAC 2015 dataset. The features we used consist of the mention tokens, candidate title tokens, both as word embeddings on 256 dimensions, the Jaccard distance between the mention and candidate title, the commonness and pagerank weights.

We evaluated the system with the same method as used in the TAC 2016 competition (Ji and Nothman, 2016) and we reached the `CEAFm` scores of 70.0 on English, on 64.4 on Chinese, and 66.5 on Spanish. We applied our linker to Swedish without any language adaptation.

We deployed the entity linker on our cluster and we used HDFS to spread the Wikipedia dump across the nodes as well as to save the final result.

5. The Document Model

We represented Wikipedia and the entity annotations using the Docforia document model[2] (Klang and Nugues, 2016b; Klang and Nugues, 2016a; Klang and Nugues, 2017). Docforia is designed it so that we can store the original markup, as well as any subsequent linguistic annotation. It consists of multiple layers, where each layer is dedicated to a specific type of annotation.

The annotations are encoded in the form of graph nodes, where a node represents a piece of data: a token, a sentence, a named entity, etc., delimited by ranges. These nodes are possibly connected by edges as in dependency graphs. The data structure used is similar to a property graph.

6. Indexing

We created an indexing tool, Panforia, to retrieve the entities from the annotated documents. As input, Panforia uses the output of the entity annotation in the form of Parquet files. Panforia is based on the Lucene search and indexing library. Each Docforia record is converted into a Lucene document by mapping record properties and documents to Lucene fields. In addition, a binary copy of the Docforia record is embedded with the indexed fields, which provides the ranges and relationships between nodes needed for the visualization.

Building directly on the Lucene library, instead of existing packages such as Solr or ElasticSearch, allowed us to optimize the insertion into an index. One key advantage of the Panforia indexer is that it can read the output from the Wikipedia pipeline, Parquet files, without a conversion step.

7. Visualization

The front-end of Panforia is a web server that embeds the Docforia library, Lucene, and a client-side web application. To search an entity, we enter a Wikidata Q-number, for instance, `urn:wikidata:Q168756`, corresponding to the entity identifier, here the University of California, Berkeley. Figure 3 shows the results of this search, where in each row, the entity is listed by its mention together with its left and right contexts. The document that contains the source of the concordance is listed in the leftmost column and the offset from the beginning in the last column.

In the figure, we can see that the entity has many possible mentions: *University of California, Berkeley*, *Berkeley*, *UC Berkeley*, etc. All these mentions and concordances are automatically retrieved through the entity index. We can visualize the document by clicking on a link in the left column.

For each document, the interactive visualization tool also enables the user to examine the annotated layers resulting from the HTML parsing (Sect. 3.). These layers include the manually set anchors, the automatically detected entities, and text enrichment. These layers are selectable from the dropdown menu to the right. Figure 4 shows an example with the automatically linked entities, the text in bold (strong) and in italics.

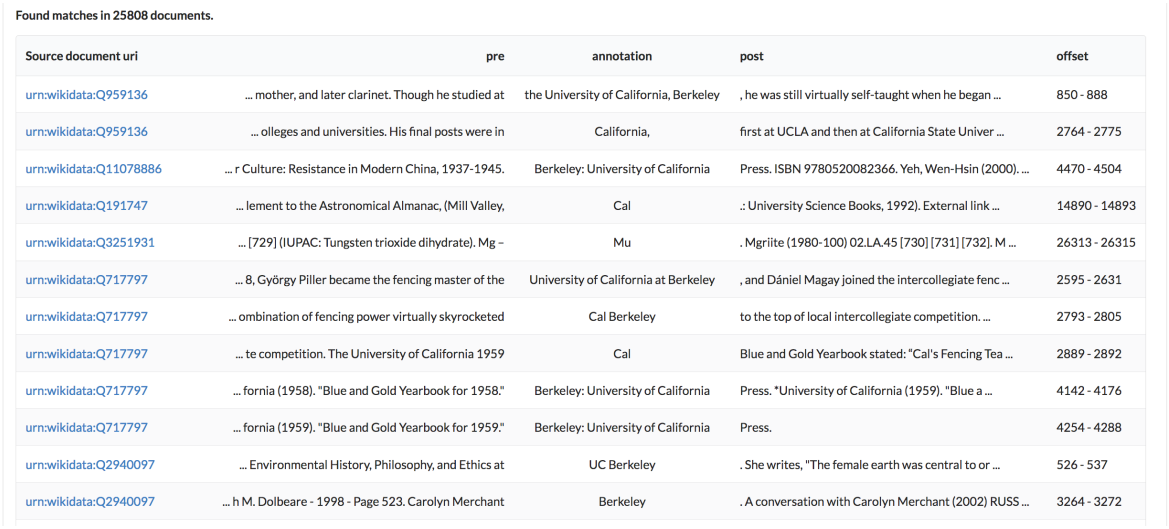
Figure 5 shows an example of results we obtained in the Swedish Wikipedia when we searched the entity Göran Persson, the former Swedish Prime Minister, using his Q-number: Q53747. This mention, *Göran Persson*, is ambiguous and Wikipedia lists as many as four different entities with this name: The former Swedish Prime Minister, a progressive musician (Q6042900), a Swedish social democratic politician, former member of the Riksdag (Q5626648), and a Swedish statesman from the 16th century (Q2625684). The latter is also spelled Jöran Person.

Searching the mention *Göran Persson* would return articles or concordances with any of these entities, while searching the entity through its Q-number only returns the intended person, either with her/his name or with other mentions such as *Persson* or *Göran*. The results are given in the forms of concordances with left and right contexts (Fig. 5).

8. Conclusion and Future Work

We have described a system to extract, index, search, and visualize entities on the English and Swedish Wikipedia. Given a Wikidata Q-number, a user can interactively retrieve all the concordances of an entity in the articles, para-

[2] `https://github.com/marcusklang/docforia/`



Found matches in 25808 documents.

Source document uri	pre	annotation	post	offset
urn:wikidata:Q959136	... mother, and later clarinet. Though he studied at	the University of California, Berkeley	, he was still virtually self-taught when he began ...	850 - 888
urn:wikidata:Q959136	... olleges and universities. His final posts were in	California,	first at UCLA and then at California State Univer ...	2764 - 2775
urn:wikidata:Q11078886	... r Culture: Resistance in Modern China, 1937-1945.	Berkeley: University of California	Press. ISBN 9780520082366. Yeh, Wen-Hsin (2000). ...	4470 - 4504
urn:wikidata:Q191747	... lement to the Astronomical Almanac, (Mill Valley,	Cal	.: University Science Books, 1992). External link ...	14890 - 14893
urn:wikidata:Q3251931	... [729] (IUPAC: Tungsten trioxide dihydrate). Mg –	Mu	. Mgriite (1980-100) 02.LA.45 [730] [731] [732]. M ...	26313 - 26315
urn:wikidata:Q717797	... 8, György Piller became the fencing master of the	University of California at Berkeley	, and Dániel Magay joined the intercollegiate fenc ...	2595 - 2631
urn:wikidata:Q717797	... ombination of fencing power virtually skyrocketed	Cal Berkeley	to the top of local intercollegiate competition. ...	2793 - 2805
urn:wikidata:Q717797	... te competition. The University of California 1959	Cal	Blue and Gold Yearbook stated: "Cal's Fencing Tea ...	2889 - 2892
urn:wikidata:Q717797	... fornia (1958). "Blue and Gold Yearbook for 1958."	Berkeley: University of California	Press. *University of California (1959). "Blue a ...	4142 - 4176
urn:wikidata:Q717797	... fornia (1959). "Blue and Gold Yearbook for 1959."	Berkeley: University of California	Press.	4254 - 4288
urn:wikidata:Q2940097	... Environmental History, Philosophy, and Ethics at	UC Berkeley	. She writes, "The female earth was central to or ...	526 - 537
urn:wikidata:Q2940097	... h M. Dolbeare - 1998 - Page 523. Carolyn Merchant	Berkeley	. A conversation with Carolyn Merchant (2002) RUSS ...	3264 - 3272

Figure 3: Searching an entity in the Wikipedia pages, where Q168756 is the Wikidata identifier of the University of California, Berkeley. The entity concordances, where each concordance is listed with its source, mention in the text, left and right contexts, and position in the text

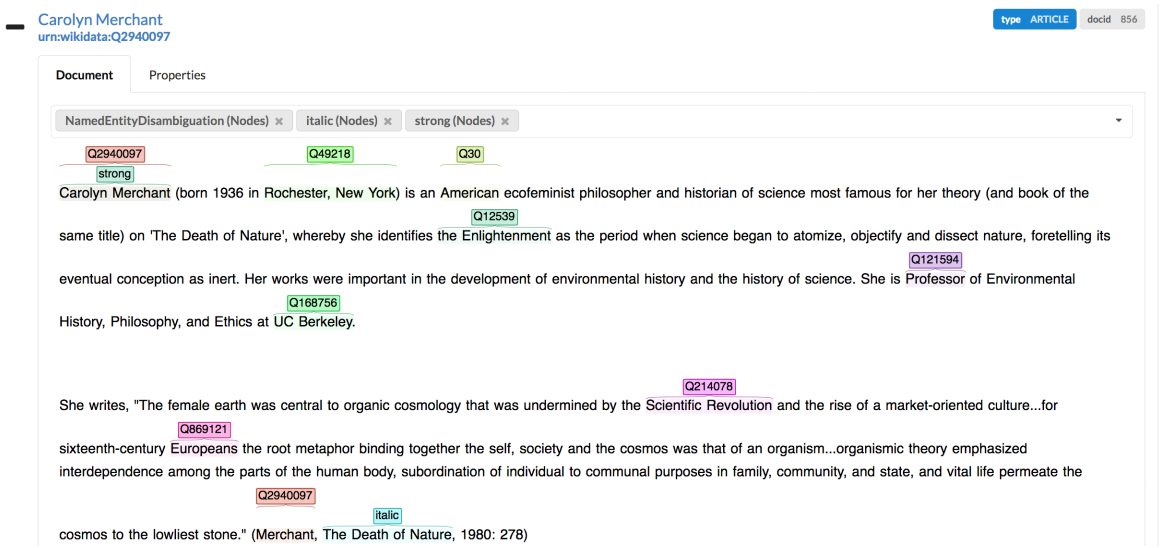
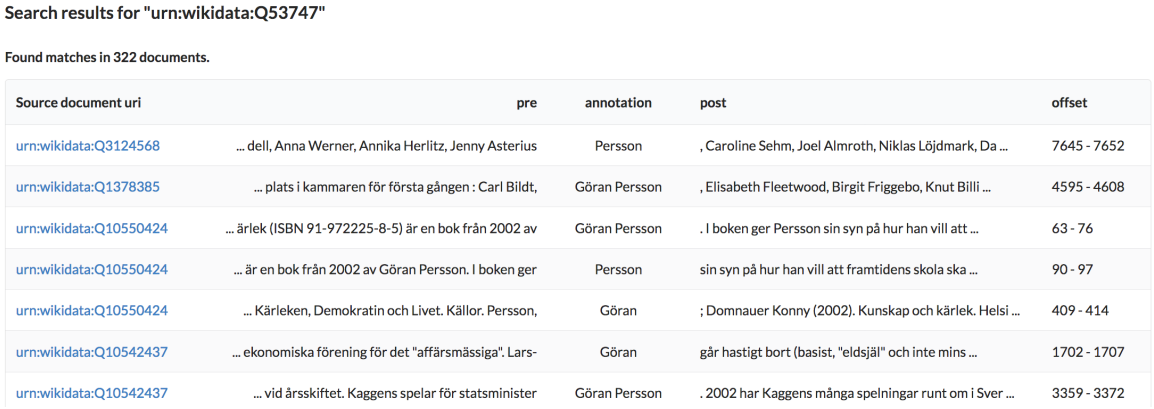


Figure 4: Visualization of annotated layers: The automatically linked entities, text in bold and in italics



Search results for "urn:wikidata:Q53747"

Found matches in 322 documents.

Source document uri	pre	annotation	post	offset
urn:wikidata:Q3124568	... dell, Anna Werner, Annika Herlitz, Jenny Asterius	Persson	, Caroline Sehm, Joel Almroth, Niklas Löjdmark, Da ...	7645 - 7652
urn:wikidata:Q1378385	... plats i kammaren för första gången : Carl Bildt,	Göran Persson	, Elisabeth Fleetwood, Birgit Friggebo, Knut Billi ...	4595 - 4608
urn:wikidata:Q10550424	... ärlek (ISBN 91-972225-8-5) är en bok från 2002 av	Göran Persson	. I boken ger Persson sin syn på hur han vill att ...	63 - 76
urn:wikidata:Q10550424	... är en bok från 2002 av Göran Persson. I boken ger	Persson	sin syn på hur han vill att framtidens skola ska ...	90 - 97
urn:wikidata:Q10550424	... Kärleken, Demokratin och Livet. Källor. Persson,	Göran	; Domnauer Konny (2002). Kunskap och kärlek. Helsi ...	409 - 414
urn:wikidata:Q10542437	... ekonomiska förening för det "affärsmässiga". Lars-	Göran	går hastigt bort (basist, "eldsjäl" och inte mins ...	1702 - 1707
urn:wikidata:Q10542437	... vid årsskiftet. Kaggens spelar för statsminister	Göran Persson	. 2002 har Kaggens många spelningar runt om i Sver ...	3359 - 3372

Figure 5: Concordances of the entity *Göran Person*, Q53747. The results are given in the form of concordances with a left and right contexts.

graphs, and metadata. The user can then select a concordance and the annotations s/he wants to visualize.

This system could be improved in many ways. The entity linker makes no assumption on the language and could easily be applied to other Wikipedias. We plan to extend this demonstration to four other languages: French, German, Spanish, and Russian and for one entity, show the concordances in the six languages.

Finally, we plan to introduce a coreference resolution for the languages where a coreference-annotated corpus exists or where a solver is available.

The demonstrations will be available at: `http://vilde.cs.lth.se:9001/en-hedwig/` for English and `http://vilde.cs.lth.se:9001/sv-hedwig/` for Swedish.

9. Acknowledgements

This research was supported by Vetenskapsrådet, the Swedish research council, under the *Det digitaliserade samhället* program.

10. Bibliographical References

Bagga, A. and Baldwin, B. (1998). Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 79–85. Association for Computational Linguistics.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, April.

Bunescu, R. C. and Pasca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *European Chapter of the Association for Computational Linguistics*, volume 6, pages 9–16.

Cucerzan, S. (2007a). Large-scale named entity disambiguation based on wikipedia data. In *Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, volume 7, pages 708–716.

Cucerzan, S. (2007b). Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 708–716, Prague, June. Association for Computational Linguistics.

Cucerzan, S. (2014). Name entities made obvious: The participation in the ERD 2014 evaluation. In *Proceedings of Entity Recognition and Disambiguation, ERD'14*, Gold Coast.

Eckhardt, A., Hreško, J., Procházka, J., and Smrž, O. (2014). Entity linking based on the co-occurrence graph and entity probability. In *Proceedings of the First International Workshop on Entity Recognition & Disambiguation*, ERD '14, pages 37–44.

Ferrucci, D. A. (2012). Introduction to "This is Watson". *IBM Journal of Research and Development*, 56(3.4):1:1 –1:15, May-June.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370, Ann Arbor.

Han, X. and Zhao, J. (2009). Named entity disambiguation by leveraging wikipedia semantic knowledge. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 215 – 224.

Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, Edinburgh.

Ji, H. and Nothman, J. (2016). Overview of TAC-KBP2016 Tri-lingual EDL and Its Impact on End-to-End KBP. In *Proceedings of the Ninth Text Analysis Conference (TAC 2016)*, Gaithersburg, Maryland. National Institute of Standards and Technology.

Klang, M. and Nugues, P. (2016a). Docforia: A multilayer document model. In *Proceedings of SLTC 2016*, Umeå, November.

Klang, M. and Nugues, P. (2016b). WIKIPARQ: A tabulated Wikipedia resource using the Parquet format. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4141–4148, Portorož, Slovenia, may.

Klang, M. and Nugues, P. (2017). Docforia: A multilayer document model. In *Proceedings of the 21st Nordic Conference of Computational Linguistics*, page 226–230, Gothenburg, May.

Lipczak, M., Koushkestani, A., and Milios, E. (2014). Tulip: Lightweight entity recognition and disambiguation using wikipedia-based topic centroids. In *Proceedings of the First International Workshop on Entity Recognition & Disambiguation*, ERD '14, pages 31–36.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland.

Mihalcea, R. and Csomai, A. (2007). Wikify!: Linking documents to encyclopedic knowledge. In *Proceedings of the Sixteenth ACM Conference on CIKM*, CIKM '07, pages 233–242, Lisbon, Portugal.

Milne, D. and Witten, I. H. (2008). Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 509–518.

O'Madadhain, J., Fisher, D., White, S., and Boey, Y.-B. (2003). The JUNG (Java Universal Network/Graph) framework. Technical Report UCI-ICS 03-17, School of Information and Computer Science, University of California, Irvine.

Piccinno, F. and Ferragina, P. (2014). From tagme to wat: A new entity annotator. In *Proceedings of the First International Workshop on Entity Recognition & Disambiguation*, ERD '14, pages 55–62, New York, NY, USA. ACM.

Sil, A., Dinu, G., and Florian, R. (2015). The IBM system for trilingual entity discovery and linking at TAC 2015. In *Proceedings of the Eighth Text Analysis Conference (TAC2015)*.

Smith, J. R., Quirk, C., and Toutanova, K. (2010). Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, pages 403–411.

Södergren, A. and Nugues, P. (2017). A multilingual entity linker using PageRank and semantic graphs. In *Proceedings of the 21st Nordic Conference of Computational Linguistics*, page 87–95, Gothenburg, May.

Tan, Y., Zheng, D., Li, M., and Wang, X. (2015). BUPT-Team participation at TAC 2015 knowledge base population. In *Proceedings of the Eighth Text Analysis Conference (TAC2015)*.

Wang, P., Hu, J., Zeng, H.-J., and Chen, Z. (2009). Using wikipedia knowledge to improve text classification. *Knowledge and Information Systems*, 19(3):265–281.

Wu, F. and Weld, D. S. (2010). Open information extraction using Wikipedia. In *Proceedings of the 48th Annual Meeting of the ACL*, page 118–127, Uppsala, Sweden.

Xu, M., Jiang, H., and Watcharawittayakul, S. (2017). A local detection approach for named entity recognition and mention detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1237–1247, Vancouver, Canada, July. Association for Computational Linguistics.

Zhang, S., Jiang, H., Xu, M., Hou, J., and Dai, L. (2015). The fixed-size ordinally-forgetting encoding method for neural network language models. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 495–500, Beijing, China, July. Association for Computational Linguistics.

Learning to Map Natural Language Statements into Knowledge Base Representations for Knowledge Base Construction

Chin-Ho Lin, Hen-Hsen Huang, and Hsin-Hsi Chen
Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan
{chlin, hhhuang}@nlg.csie.ntu.edu.tw, hhchen@ntu.edu.tw

Abstract

Directly adding the knowledge triples obtained from open information extraction systems into a knowledge base is often impractical due to a vocabulary gap between natural language (NL) expressions and knowledge base (KB) representation. This paper aims at learning to map relational phrases in triples from natural-language-like statement to knowledge base predicate format. We train a word representation model on a vector space and link each NL relational pattern to the semantically equivalent KB predicate. Our mapping result shows not only high quality, but also promising coverage on relational phrases compared to previous research.

Keywords: knowledge base, knolwedge base construction, relation mapping

1. Introduction

Knowledge bases (KBs) such as Freebase (Bollacker et al., 2008) and DBpedia (Auer et al., 2007) are fundamental resources for many intelligent applications. Currently, the construction and updating of KBs, directly or indirectly, rely on human labor. Keeping KBs up-to-date by humans is cost intensive and impractical. To reduce the cost and to minimize the updating latency, automatically updating a KB with knowledge extracted from natural language (NL) content is a feasible strategy.

In KBs, a fact is represented by a triple (*subject*, *predicate*, *object*), where *subject* and *object* are two entities in the KB, and *predicate* describes their relation. With an information extraction system (Carlson et al., 2010; Fader et al., 2011), we can also extract facts from NL text in the format (np_1, *pattern*, np_2), where np_1 and np_2 are two entities and *pattern* is the relational phrase between them. However, the extracted triples from NL text do not always follow the paradigm of KB, and that becomes a challenging issue for KB construction. For example, the NL triple (Garnett, was born in, Mauldin) extracted by ReVerb (Fader et al., 2011) shows a fact identical to the KB triple (Kevin Garnett, birthPlace, Mauldin (South Carolina)). Although these two triples state the same fact, there is a vocabulary gap between them. In the former triple, "was born in" is an NL-like expression and "birthPlace" in the latter triple is a formatted predicate used in KB. These two relational phrases are different in surface forms. They cannot be mapped by string matching directly. In addition, a KB predicate may be described in multiple NL statements. A number of ReVerb patterns such as "is the hometown of", "was raised in", and "grew up in" are related to the predicate "hometown" in DBpedia. That makes the mapping between KB and NL even more challenging.

Recently, more and more works show their interests in the issue of KB construction. Knowledge graph embedding models (Bordes et al., 2013; Yang et al., 2015; Xie et al., 2016a; Xie et al., 2016b) focus on learning the vector representation on the KB side only. Previous works (Nakashole et al., 2012; Riedel et al., 2013; Dutta et al., 2015) aim to solve similar problems as ours. Nakashole et al. (2012) in their PATTY approach try to learn paraphrases to define the predicates of DBpedia, and Dutta et al. (2015) propose clustering-based approaches to transform the knowledge extracted by an open information extraction system into DBpedia paradigm. Riedel et al. (2013) propose universal schemas, which are the mapping between NL surface forms to the KB predicates, by using matrix factorization. However, all of them suffer from low coverage on relational phrases. In this work, we aim to propose a more general framework that maps relational phrases extracted from an NL resource to DBpedia predicates. Our method is capable of covering most NL patterns and KB predicates. The relational mappings can be used for a range of applications. For KB construction, the mappings can be consulted for mining the new facts from textual data written in NL. For question answering over the KB, the mappings can be used for looking up the facts in KB that are candidates for the answer.

The rest of this paper is organized as follows. Section 2 describes the corpora used as NL data for learning the relational mapping. Section 3 presents our learning to map approach. In Section 4, we conduct experiments for evaluating the results. The challenging issues of this work are discussed in Section 5. Finally, Section 6 concludes this work.

2. Linguistic Resource

English Wikipedia is regarded as the NL text resource in this study. We obtain the NL relational triples from ReVerb (Fader et al., 2011), a dataset of relational triples extracted from Wikipedia. Let an NL dataset D_{NL} be a 3-tuple (P, N, I_{NL}), where P is a set of NL patterns, N is a set of entities, and I_{NL} is a set of NL triples. For example, (Garnett, was born in, Mauldin) $\in I_{NL} = \{(n_i, p_k, n_j) | n_i, n_j \in N, p_k \in P\}$.

On the other hand, DBpedia (Auer et al., 2007) serves as our target KB. We define a KB dataset D_{KB} as a 5-tuple (R, E, T, E_T, I_{KB}), where R is a set of KB predicates, E is a set of entities in KB triples, T is a set of KB entity types, E_T is a set of entity-type pairs in KB, and I_{KB} is a set of KB triples. For example, (Kevin Garnett, birthPlace, Mauldin (South Carolina)) $\in I_{KB} = \{(e_m, r_o, e_n) | e_m, e_n \in E, t_s, t_u \in T, (e_m, t_s), (e_n, t_u) \in E_T, r_o \in R\}$.

To resolve the entity disambiguation problem, a lexicalization dataset[1] released on the DBpedia Spotlight (Mendes et al., 2011) official website is consulted. We extract an entity-alias list from the dataset and let the alias list be

$$Alias = \{(a_{m1}, a_{m2}, \dots, a_{m|A(e_m)|}) | a_{m1}, a_{m2}, \dots, a_{m|A(e_m)|} \in A(e_m)\}$$

where $e_m \in E$ and $A(e_m)$ is a set of entity aliases corresponding to the KB entity e_m.

We further randomly sampled 10 million sentences Sen_{Clue} from ClueWeb09 dataset[2] as an NL resource, which is considered as an auxiliary dataset for training word embedding models.

3. Relational Mapping

We propose an approach inspired by word2vec model (Mikolov et al., 2013), i.e., Skip-gram and CBOW, to build a relational mapping. Our method projects all relational phrases, i.e., NL patterns and KB predicates, to a vector space and measures cosine similarity between relational phrases on this space.

As illustrated in Figure 1, our method consists of three components, EB (Entity Bridging with Alias Resolution), DR (Decompose Relational Phrases and Introduce Additional NL Text), and TF (Filter Relational Mapping by Argument Types constraint). The first workflow builds a relational mapping by conducting entity bridging on training triples with consultation of entity alias dictionary. Our second workflow is based on EB and further performs DR, which considers the information of words decomposed from relational phrases and adds auxiliary natural language sentences into training data. Finally, TF is used to filter the mappings built by EB or EB+DR. The descriptions of EB, DR, and TF are presented in Sections 3.1, 3.2 and 3.3, respectively.

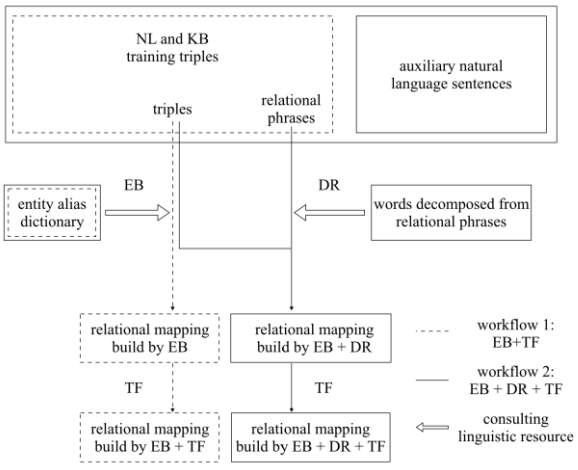


Figure 1: Overview of our approach to relational mapping.

3.1 EB : Entity Bridging with Alias Resolution

Unlike KB predicates that are formatted, patterns in ReVerb are NL-like expressions. As the examples shown in Table 1, KB predicates and semantically related NL patterns might be different in surface form such as the predicate "headquarter" and the pattern "is based in". Another example in Table 1 is the KB predicate "spouse" and NL patterns "is the wife of", "is the husband of", and "is the first wife of". The NL patterns might contain more specific details but are still mapped to the KB predicate. To build a relational mapping of NL patterns and KB predicates, our model should learn the connection of information from KB and NL data resources.

KB Predicate	NL Patterns
headquarter	is headquartered in / is based in
spouse	is the wife of / is the husband of / is the first wife of
hometown	grew up in / was raised in
writer	was written by / is a novel by

Table 1: Examples of KB predicates and their semantically related NL patterns.

EB aims to capture the structural information between entities and relations, and then links relational phrases through entity bridging. Figure 2 shows how EB works. For instance, (Kobe Bryant, was born in, Philadelphia) and (Kobe Bryant, birthPlace, Philadelphia) are triples from NL and KB datasets, respectively. Through the co-occurrence of entity pairs and relational phrases, the model gradually learns the connection between "was born in" and "birthPlace".

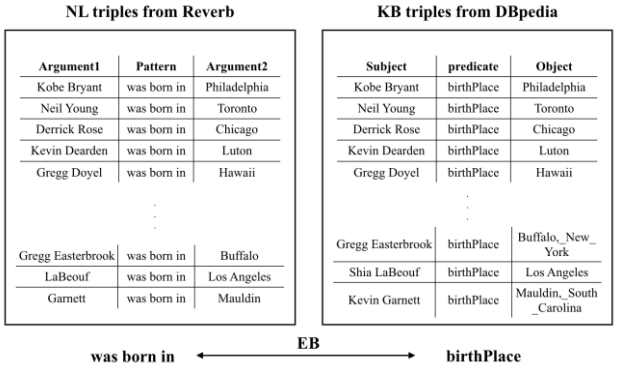


Figure 2: Examples of Entity Bridging (EB).

We train a word embedding model with the KB and NL triples. Through the update of entity pairs, the connection of a pattern p_k and a predicate r_o are captured in the model training process. More precisely, given a KB training triple (e_m, r_o, e_n), a training sequence W for the model will be w_1, w_2, w_3, where w_1=e_m, w_2=r_o, and w_3=e_n. The model maximizes the average log probability θ_{triple} as shown in Equation (1).

$$\theta_{triple} = \frac{1}{|W|} \sum_{x=1}^{|W|} \sum_{-c \le y \le c, y \ne 0} \log p(w_{x+y} | w_x) \qquad (1)$$

where c is the size of context windows, w_x is the central word, and w_{x+y} denotes one of the context words. Probability $p(w_{x+y}|w_x)$ is calculated using the softmax function as shown in Equation (2).

[1]http://spotlight.sztaki.hu/downloads/latest_data/en.tar.gz

[2] http://lemurproject.org/clueweb09/

$$p(w_{x+y}|w_x) = \frac{\exp({v'_{w_{x+y}}}^T v_{w_x})}{\sum_{z=1}^{|V|} \exp({v'_z}^T v_{w_x})} \quad (2)$$

where v_w and v'_w are the "input" and "output" embedding of word w, and $|V|$ is the vocabulary size of the model. In addition, the $Alias$ is consulted for alias resolution. For an entity e_m in KB training triple, its alias $a_{mx} \in A(e_m)$ is updated by maximizing average log likelihood θ_{alias} in Equation (3).

$$\theta_{alias} = \frac{1}{|A(e_m)|} \sum_{a_{mx} \in A(e_m)} \log p(a_{mx}|e_m) \quad (3)$$

where $p(a_{mx}|e_m)$ is computed by using the softmax function (2).

The trained model results in an embedding space, where NL patterns and KB predicates are represented as vectors in this space. Thus, the similarity between an NL pattern and a KB predicate can be measured by their cosine similarity. On the one hand, most similar KB predicates of an NL pattern can be considered as its mapping targets. On the other hand, most similar NL patterns of a KB predicate can be regarded as its mapping patterns. In Section 4, triple linking task and human verification task will evaluate the results from these two aspects, respectively.

3.2 DR: Decomposing Relational Phrases and Introducing Additional NL Text

EB may suffer from data sparseness because each relational phrase is treated as a distinct symbol, and information from the words that compose a relational phrase is completely ignored. For example, the meaning of predicate "birthPlace" can be captured from words "birth" and "place". Thus, DR is proposed and integrated with EB for leveraging the words decomposed from relational phrases. More clearly, given a KB predicate r_o, the word semantics of r_o will be jointly learned by maximizing the average log probability $\theta_{compose}$ as Equation (4).

$$\theta_{compose} = \log p(r_o|c_{o1}, c_{o2}, \dots, c_{on}) \quad (4)$$

where $c_{o1}, c_{o2}, \dots, c_{on}$ are words decomposed from r_o and n is the number of the words. The computation of the probability is done by the softmax function (2) and a composition function that simply averages the vectors of words $c_{o1}, c_{o2}, \dots, c_{on}$.

Figure 3 illustrates how DR encodes words decomposed from relational phrases and jointly learns the meaning of relational phrases. The advantage of DR is that the semantics of a predicate is expanded by its compositional words, which are written in NL. For example, the word meaning of KB predicate "birthPlace" will also be considered from words "birth" and "place" and the word meaning of NL pattern "was born in" will similarly be viewed from "was", "born" and "in". Relational phrases that are semantically related might consist of words with similar meaning such as "birth" and "born" in this example. In other words, we reduce the data sparseness of relational phrases by connecting patterns and predicates through the decomposed words.

In addition to the information from the KB, we add additional NL statements Sen_{Clue} to the training set. These statements serve as an auxiliary resource that aids to model the meaning of the decomposed words. In other words, we co-train a distributed word model in the same vector space. In the training process, the decomposed words play a role of bridges that connect similar relational phrases. The natural language statements provide semantic information for those words.

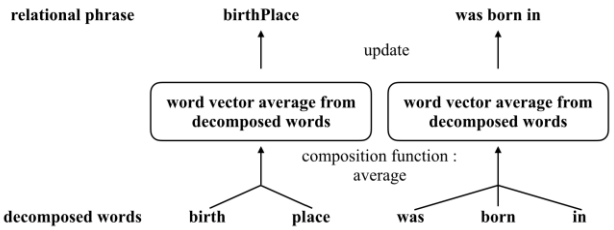


Figure 3: Example of Decomposing Relational Phrases (DR).

3.3 TF: Filter Relational Mapping by Argument Types of Relational Phrases

We further filter the mapping results with argument type constraints. That is, the argument type of an NL pattern should be consistent with those of its corresponding KB predicates. The mappings with inconsistent argument types are removed from the relational mapping list.

We obtain the argument type constraint for each relational phrase by voting with training triples. More formally, for each KB triple $(e_m, r_o, e_n) \in I_{KB}$, $e_m \in t_s$, $e_n \in t_u$ the argument type for r_o is voted as (t_s, t_u). We count the majority argument type for each predicate and define the argument type constraint of r_o as $Arg_{type(r_o)} = (t_s, t_u)$.

In NL side, unlike the entity types defined in KB datasets, we have to find the types of entities extracted from natural language sentences. We obtain the type of each NL entity by matching it with KB entity aliases. Then, we vote argument types of NL patterns with NL triples in the similar way as above. If an entity n_i has c possible types, each type of the entity n_i will be weighted equally by $\frac{1}{c}$. If $c = 0$, the triple will not contribute to argument type determination. In this way, we generate argument type constraints for NL patterns. Let $Arg_{type(p_k)} = (t_d, t_f)$ be the type constraint of the corresponding pattern p_k for some $t_d, t_f \in T$.

4. Experiments

Dataset and Experiment Setting: The statistics of NL and KB datasets show that $|I_{NL}| = 407{,}239$, $|P| = 100{,}264$, $|I_{KB}| = 14{,}408{,}940$, and $|R| = 662$. Comparatively, even though the size of KB triple set is much larger than NL triple set, KB predicates are formatted and there are only 662 predicates in KB dataset. We randomly split our dataset into five folds and conduct five-fold cross validation for our experiment. We train 300-dimensional vector models with the proposed methods and analyze the mapping results. The training parameters of our model, i.e., (window size, negative sample, min count), are set to (5, 10, 0), respectively. The relational phrases with less than 5 occurrences in ReVerb and DBpedia are excluded from the mapping. After filtering, our approach builds a relational mapping that covers 7,361 of 9,171 frequent ReVerb patterns with 629 of 634 frequent DBpedia predicates.

Compared to previous works, Dutta et al. (2015) cover 212 ReVerb patterns and 41 DBpedia predicates, Riedel et al. (2013) cover 100 Freebase facts in training and test data, and Nakashole et al. (2012) cover 225 predicates, we generate a higher coverage of relational mapping.

Evaluation: Because of lack of ground truth, most previous work evaluates their relational mapping result by human annotation only. In this work, we try to evaluate our approach from two perspectives. Firstly, we conduct a triple linking task that aims to link NL triples to the KB triples sharing the same knowledge. It reflects the ability of our mapping approach to translate relational phrases from NL side to KB paradigm. Secondly, we further evaluate the performance of our mapping by human verification. It demonstrates the result and the accuracy of NL patterns that link to each KB predicate.

Triple linking task: This task simulates knowledge base construction. The KB test set is considered as new facts. The word embedding model learned from the KB training set builds a relational mapping between NL patterns and KB predicates. Through the mapping, we can add new knowledge into the KB by translating NL triples to KB triples. We judge the correctness of translation by checking if the translated triple is actually in the test data. Formally, given an NL triple (n_i, p_k, n_j), $n_i \in A(e_m), n_j \in A(e_n)$, we have to predict a KB predicate r_o such that (e_m, r_o, e_n) is the corresponding fact in KB test set. Thus, we generate NL test triples Λ_{NL} and ground truth Λ_{KB} as follows. Λ_{NL} is a set of triples $(n_i, p_k, n_j) \in I_{NL}$ and Λ_{KB} is a set of triples $(e_m, r_o, e_n) \in I_{KB}$, where $n_i \in A(e_m), n_j \in A(e_n)$. In this way, we derive $|\Lambda_{NL}| = 54{,}752$ and $|\Lambda_{KB}| = 58{,}504$.

Due to the low coverage on relational phrases, the results of Riedel et al. (2013) and Dutta et al. (2015) cannot be directly compared with ours under this task. Thus, we built a baseline model through a counting-based approach that counts the co-occurrence between each pattern and each predicate found in the same or alias entity pairs. The baseline model always selects the majority. TransE (Bordes et al., 2013) focuses on learning the vector representation in KB side only. Although it solves a problem different from ours, we also adapt it to this task. We train a model with NL and KB triples. Given an NL triple (n_i, p_k, n_j), the TransE model is trained to optimize the equation $n_i + p_k \approx n_j$, so we can predict a KB predicate r_o through the entity operation $n_j - n_i$. We denote this operation as TransE(entity). We can also rank mapping candidates of p_k by cosine similarity of NL patterns and KB predicates. We denote it as TransE(rel). We calculate hit@k and MRR to measure the performance. Hit@k indicates the percentage of NL triples where correct relations can be found in the top k positions. In the special case where k=1, hit@1 is equivalent to P@1 (Precision at 1). MRR is the mean reciprocal rank of correct mapping and is calculated to the 100 position.

As shown in Table 2, counting-based baseline suffers from the low translation rate on relational phrases, i.e., no suitable mapping can be applied, and its hit@k shows no difference when k is larger than 5. By contrast, though TransE(entity) and TransE(rel) do not have such a problem, all our methods outperform them. Besides, DR expands the meaning of relational phrases by decomposed words. Although hit@k drops at k=1, the performance shows large improvement when k is larger than 5. That indicates the effectiveness of semantic information provided by decomposed words and the additional natural language statements. TF shows the strength of type filter. EB+DR+TF even achieves 0.800 and 0.327 of hit@20 and MRR, respectively, in this task.

	hit@1	**hit@5**	**hit@10**	**hit@20**	**MRR**
baseline	0.113	0.151	0.151	0.151	0.129
TransE(entity)	0.042	0.116	0.174	0.250	0.086
TransE(rel)	0.035	0.122	0.202	0.308	0.087
EB	0.189	0.294	0.345	0.392	0.242
EB+DR	0.117	0.273	0.361	0.457	0.205
EB+TF	0.222	0.358	0.408	0.530	0.299
EB+DR+TF	0.158	0.538	0.709	0.800	0.327

Table 2: Evaluation results of triple linking task.

Human verification task: We select top 100 frequent predicates from KB triples and manually annotate their top 5 mapped NL patterns. Because human annotation is cost intensive, we only verify the mapping results by EB+DR+TF, the best performing method in the triple linking task.

The results of human verification are shown in Table 3. The hit@1, hit@3, and hit@5 of the most frequent 50 and 100 predicates, respectively, are reported. This results further confirm the quality of our mapping from another aspect. Some NL patterns are mapped to KB predicates that have exactly the same meaning, e.g., "birthplace" and "was born in" is counted a correct mapping. Some NL patterns do not have exactly the same meaning with the KB predicates, but they can be inferred. For example, the pattern "is a fantasy novel by" infers the predicate "author". Thus, we regard them as correct mapping. Besides, there are some incorrect mapping examples, such as pattern "was born in" and predicate "residence". The predicate "residence" indicates a place a person live in.

	hit@1	**hit@3**	**hit@5**
Top 50 Predicates	0.352	0.480	0.528
Top 100 Predicates	0.326	0.456	0.510

Table 3: Evaluation results of human verification task.

5. Discussion

We find four major types of errors in our mapping:

Complex concept: Some KB predicates containing complex concept are difficult to map accurately. For instance, the KB predicate "leftTributary" contains not only the relational expression "tributary", but also the concept "left". In this case, NL patterns such as "is a tributary of", "is a tributary to", and "is a river in" can only capture partial phrase meaning, and they are regarded as inaccurate mapping. The complex concept would be better modeled by decomposing it to multiple simple concepts.

Uncommon in NL sentences: Some KB predicates such as "youthWing" and "varietals" are uncommon in NL sentences, so that there are insufficient instances available for training. As a result, DR may not perform well since it aims to capture semantic information from NL sentences. Fortunately, most uncommon predicates are less important in general domain. For a specific-purpose application, the

in-domain corpus can be used to train a dedicated relational mapping.

Similar context: Word embedding models learn word meaning through NL context. However, some antonyms share similar context in NL, and they may have small cosine distance in the vector space. For instance, the NL patterns "is a large town in" and "is a small town in" are hard to distinguish. This issue has been addressed in other applications that use word embedding models.

Multiple meanings: Some relational phrases have multiple meanings and only parts of the meanings are used in KB. This leads to wrong relational mapping. For example, the word "billed" has several meanings, but the predicate "billed" is only applied to the state that a wrestler comes from, such as (A-1(wrestler), billed, Niagara Falls (Ontari)). One of possible solutions to this issue is performing word sense disambiguation (WSD) on the corpus, and learning the relational mapping at the sense level, instead of at the word level.

6. Conclusion

Relational mapping is a challenging problem. This paper proposes an approach that provides a quality mapping in terms of coverage and correctness. This method is also unsupervised and is not restricted to a specific KB. It is easy to apply to different data resources for various applications such as KB construction and question-answering. Because the ground truth is not available, we also propose a triple linking task. The trask provides an automatic and scalable evaluation for relational mapping.

In the end, we suggest some research directions for improving the proposed approach in the future. DR encodes words decomposed from relational phrases by a compositional function that simply averages the embeddings of these words. The joint learning model may handle the compositionality better in phrase embedding learning. TF considers entity type information by filtering the relational mapping with argument type constraints. Embedding model learns type information while training knowledge embedding may be explored further.

7. Acknowledgements

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST-105-2221-E-002 -154 -MY3, MOST-106-2923-E-002 -012 -MY3 and MOST-107-2634-F-002-011-, and Academia Sinica under grant AS-107-TP-A05.

8. Bibliographical References

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD 2008), pages 1247–1250, Vancouver, Canada. ACM.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J. and Yakhnenko, O. (2013). Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS 2013), pages 2787–2795, Lake Tahoe, Nevada, USA. Curran Associates Inc.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr., E. R., and Mitchell, T. M. (2010). Toward an Architecture for Never-ending Language Learning. In Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2010), pages 1306–1313, Atlanta, Georgia, USA. AAAI Press.

Dutta, A., Meilicke, C., and Stuck-enschmidt, H. (2015). Enriching Structured Knowledge with Open Information. In Proceedings of the 24th International Conference on World Wide Web (WWW 2015), pages 267–277, Florence, Italy. International World Wide Web Conferences Steering Committee (IW3C2).

Mendes, P. N., Jakob, M., Garcı́a-Silva, A., and Bizer, C. (2011). Dbpedia Spotlight: Shedding Light on the Web of Documents. In Proceedings of the 7th International Conference on Semantic Systems (I-Semantics 2011), pages 1–8, Graz, Austria. ACM.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In Proceedings of International Conference on Learning Representations 2013 (ICLR 2013).

Nakashole, N., Weikum, G., and Suchanek, F. (2012). Patty: A Taxonomy of Relational Patterns with Semantic Types. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP 2012), pages 1135–1145, Jeju Island, Korea. Association for Computational Linguistics (ACL).

Riedel, S., Yao, L., McCallum, A., and Marlin, B. M. (2013). In Proceedings of NAACL-HLT 2013, pages 74–84, Atlanta, Georgia, USA. Association for Computational Linguistics (ACL).

Xie, R., Liu, Z., Jia, J., Luan, H., and Sun, M. (2016a). Representation Learning of Knowledge Graphs with Entity Descriptions. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016), pages 2659–2665, Phoenix, Arizona, USA. AAAI Press.

Xie, R., Liu, Z., and Sun, M. (2016b). Representation Learning of Knowledge Graphs with Hierarchical Types. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI 2016), pages 2965–2971, New York, New York, USA. AAAI Press.

Yang, B., Yih, W.-T., He, X., Gao, J., Li Deng. (2015). Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of International Conference on Learning Representations 2015 (ICLR 2015).

9. Language Resource References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A Nucleus for a Web of Open Data. In Proceedings of the 6th International the Semantic Web and 2nd Asian Conference on Asian Semantic Web Conference (ISWC 2007/ASWC 2007), pages 722–735, Busan, Korea. Springer-Verlag.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying Relations for Open Information Extraction. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011), pages 1535–1545, Edinburgh, Scotland, UK. Association for Computational Linguistics (ACL).

Building a Knowledge Graph from Natural Language Definitions for Interpretable Text Entailment Recognition

Vivian S. Silva[1], André Freitas[2], Siegfried Handschuh[1]
[1]Department of Computer Science and Mathematics, University of Passau, Innstraße 43, 94032, Passau, Germany
[2]School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, M13 9PL, UK
vivian.santossilva@uni-passau.de, andre.freitas@manchester.ac.uk, siegfried.handschuh@uni-passau.de

Abstract

Natural language definitions of terms can serve as a rich source of knowledge, but structuring them into a comprehensible semantic model is essential to enable them to be used in semantic interpretation tasks. We propose a method and provide a set of tools for automatically building a graph world knowledge base from natural language definitions. Adopting a conceptual model composed of a set of semantic roles for dictionary definitions, we trained a classifier for automatically labeling definitions, preparing the data to be later converted to a graph representation. WordNetGraph, a knowledge graph built out of noun and verb WordNet definitions according to this methodology, was successfully used in an interpretable text entailment recognition approach which uses paths in this graph to provide clear justifications for entailment decisions.

Keywords: lexical definitions, knowledge graph, text entailment

1. Introduction

Natural language lexical definitions of terms can be used as a source of knowledge in a number of semantic tasks, such as Question Answering, Information Extraction and Text Entailment. While formal, structured resources such as ontologies are still scarce and usually target a very specific domain, a large number of linguistic resources gathering dictionary definitions is available not only for particular domains, but also addressing wide-coverage commonsense knowledge.

However, in order to make the most of those resources, it is necessary to capture the semantic shape of natural language definitions and structure them in a way that favors both the information extraction process and the subsequent information retrieval, allowing the effective construction of semantic models from these data sources while keeping the resulting model easily searchable and interpretable. Furthermore, by using these models, systems can increase their own interpretability, benefiting from the structured data for performing traceable reasoning and generating explanations – features which are becoming even more valuable given the growing importance of Explainable AI (Gunning, 2017).

In this work, we propose a method for automatically building commonsense knowledge bases out of natural language dictionary definitions, which is easily extensible to any domain where natural language glossaries are available. Building upon a conceptual model based on a set of semantic roles for definitions, we classify each segment in a definition according to its relation to the entity being defined, and convert the classified data into a knowledge graph where each node is a meaningful phrase which contains a piece of self-contained information about the definiendum. Following this methodology, we processed the whole noun and verb databases of WordNet (Fellbaum, 1998) and built the WordNetGraph, and then used this knowledge graph to recognize text entailments in an interpretable way, providing concise justifications for the entailment decisions.

2. Related Work

The construction of structured databases from dictionary definitions has been largely explored, and most approaches rely on syntactic parsers for the identification of patterns that point to relationships between words (Calzolari, 1991; Vossen, 1991; Vossen, 1992; Vossen and Copestake, 1994). Among early efforts, it is remarkable the creation of LKB, a Lexical Knowledge Base (Copestake, 1991) based on typed-feature structures that can be seen as a set of attributes for a given concept, such as "origin", "color", "smell", "taste" and "temperature" for the concept *drink*, for example. The definitions from a machine-readable dictionary are parsed to extract the definiendum's genus and differentiae, and the values represented by the differentiae will fill in the feature structures for that genus. Since the features, that is, the relevant attributes for a given entity, must be defined in advance, only a restricted domain was considered in their approach.

Dolan et al. (1993) also describe an automated strategy to build a structured lexical knowledge base but, instead of the entity-attributes structure, they use syntactic parsing to identify semantic relations such as *is-a*, *part-of*, etc., to build a directed graph. Recski (2016) also derives a graph representation from dictionary definitions, but in the adopted conceptual model there are only three types of edges, numbered from 0 to 2: the 0-edge represents unary predicates and the 1 and 2-edges connects binary predicates to their arguments. In common, most approaches work at the word-level, converting each single word in the definition into a different attribute or node. In the graph knowledge base scenario, this can increase the information retrieval complexity, given that it may be necessary to concatenate the contents of several nodes to obtain meaningful enough information about an entity.

The work proposed by (Bovi et al., 2015) go beyond the word-level representation, being able to identify multiword expressions. They perform a syntactic-semantic analysis of textual definitions for Open Information Extraction

(OIE). Although they generate a syntactic-semantic graph representation of the definitions, the resulting graphs are used only as an intermediary resource for the final goal of extracting semantic relations between the entities present in the definition.

3. Graph Conceptual Model

To build the definition graph, we adopted the conceptual model proposed by Silva et al. (2016). This model extends the genus-differentia definition pattern from Aristotle's classic theory of definition (Berg, 1982; Lloyd, 1962; Granger, 1984) by defining a set of entity-centered *semantic roles* for lexical definitions. Differently from the commonly used event-centered semantic roles, which define the semantic relations holding among a predicate (the main verb in a clause) and its associated participants and properties (Màrquez et al., 2008), definition's semantic roles express the part played by an expression in a definition, showing how it relates to the *definiendum*, that is, the entity being defined.

In this model, the *genus* concept was replaced by the more general role *supertype*, which can be not only the definiendum's immediate superclass but also an ancestor higher in the concepts hierarchy. The *differentia* component was split into two roles: *differentia quality* and *differentia event*. These three roles can be seen as the representatives of an entity's essential properties, while other roles, such as *associated fact*, *purpose* or *accessory quality*, for example, define non-essential properties. The conceptual model is depicted in Figure 1, and Table 1 presents a summarized description for each of the roles defined in this model.

This set of semantic roles captures the semantic "shape" of natural language definitions and allows the extraction of structured representations from linguistic resources, enabling them to be used as knowledge sources in a wide range of semantic tasks.

Role	Description
Supertype	the immediate or ancestral entity's superclass
Differentia quality	a quality that distinguishes the entity from the others under the same supertype
Differentia event	an event (action, state or process) in which the entity participates and that is mandatory to distinguish it from the others under the same supertype
Event location	the location of a differentia event
Event time	the time in which a differentia event happens
Origin location	the entity's location of origin
Quality modifier	degree, frequency or manner modifiers that constrain a differentia quality
Purpose	the main goal of the entity's existence or occurrence
Associated fact	a fact whose occurrence is/was linked to the entity's existence or occurrence
Accessory determiner	a determiner expression that doesn't constrain the supertype-differentia scope
Accessory quality	a quality that is not essential to characterize the entity
[*Role*] particle	a particle, such as a phrasal verb complement, non-contiguous to the other role components

Table 1: Semantic roles for dictionary definitions

4. Construction Methodology

Structuring natural language definitions as a graph allows us to select the portions of information regarding an entity's description that are relevant for a certain reasoning task. For example, consider the definition (from WordNet) for the concept "lake poets", which was classified according to the model described in Section 3., illustrated in Figure 2. When retrieving data related to this concept, we could be interested only in origin- (*lake poets are English poets*), time- (*lake poets are poets at the beginning of the 19th century*) or space- (*lake poets are poets who lived in the Lake District*) related information. When each of those roles is represented as a node in a graph we can focus only on the path containing the nodes of interest. Moreover, since the definition is split into segments rather than single words, each node contains a comprehensible amount of information, avoiding the need to visit several nodes to gather intelligible phrases.

To generate the *WordNetGraph*[1] – a knowledge graph following the RDF data model – from WordNet's noun and verb glosses, we adopted the following methodology for classifying and structuring the definitions:

Synsets sample selection: in order to use a supervised machine learning model to classify the data, we needed a initial set of annotated definitions. To build this set, we randomly selected 2,000 WordNet synsets, being 1,732 noun synsets and 268 verb synsets (the verb database size is around 17% of the noun database size).

Automatic pre-annotation: the set of 2,000 definitions was automatically pre-annotated according to a rule-based heuristic that takes into account the syntactic patterns identified by statistical analysis as described by Silva et al. (2016). Using the Stanford parser (Manning et al., 2014), we generated the syntactic parse tree for each definition, identified the relevant phrasal nodes and then assigned the semantic roles more often associated to them. For example: the *supertype* for a noun definition is usually the innermost and leftmost noun phrase (NP) that contains at least one noun (NN); a *differentia event* is usually either a subordinate clause (SBAR) or a verb phrase (VP); an *event location* is normally a prepositional phrase (PP) inside a SBAR or VP and possibly containing a location named entity, and so on. Figure 3 shows the parse tree generated for the definition of the term "Scotch" – *whiskey distilled in*

[1] https://github.com/Lambda-3/WordnetGraph

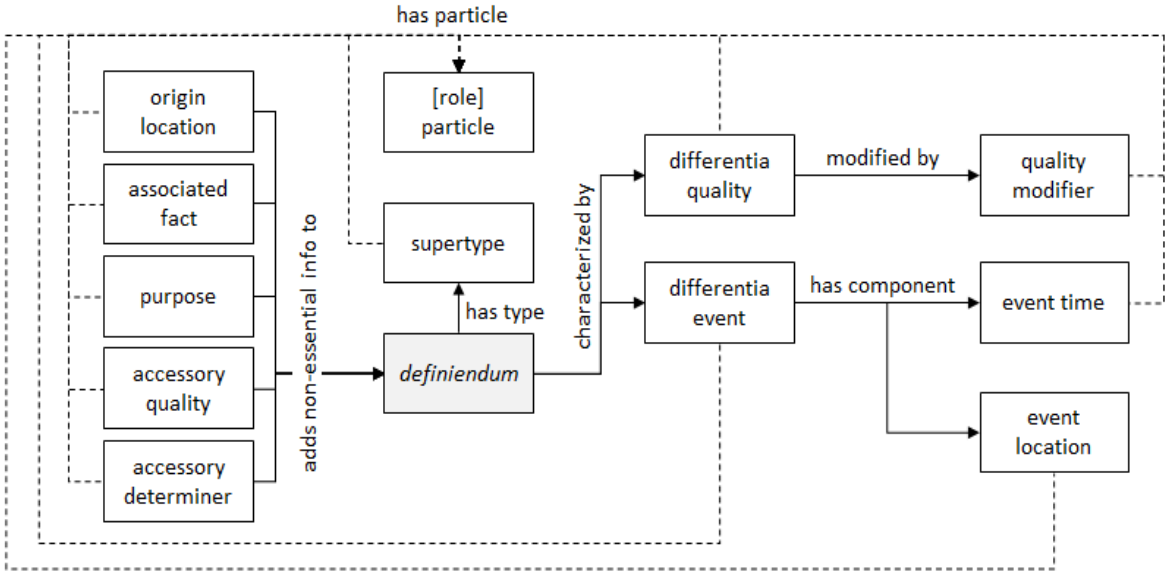


Figure 1: Conceptual model for the semantic roles for lexical definitions. Relationships between *[role] particle* and every other role in the model are expressed as dashed lines for readability.

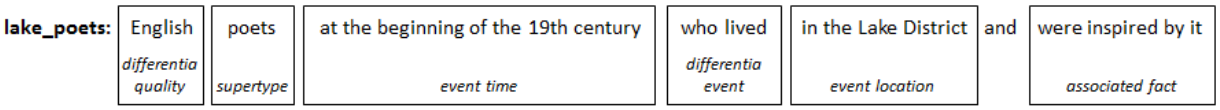


Figure 2: Example of role labeling for the definition of the "lake_poets" synset.

Scotland – and the semantic roles automatically assigned to each phrasal node.



Figure 3: Syntactic parse tree for the definition of the concept "Scotch" and assigned semantic role labels. After being classified as a differentia event, the VP is further analyzed and a PP containing an event location is identified and assigned its own role label.

Data curation: after the automatic pre-annotation, the definitions were manually curated with the aid of the Brat[2] annotation tool. Misclassifications were fixed and segments missing a role were assigned the appropriate one. Misclassifications and missing roles are due to parser errors or insufficient information (for instance, a PP inside a VP may not contain any named entity, making it hard to correctly distinguish between an event time and an event location). The manual data curation ensured that every segment in each definition, apart of leading determiners and conjunctions between roles (as opposed to conjunctions inside roles), was associated with a semantic role label.

[2]http://brat.nlplab.org/

Classifier training: the curated data was then used to train a Recurrent Neural Network (RNN) machine learning model designed for sequence labeling. We used the RNN implementation provided by Mesnil et al. (2015), which reports state-of-the-art results for the slot filling task. The dataset was split into training (68%), validation (17%) and test (15%) sets. The best accuracy reached during training was of 80.35%.

Database classification: the trained classifier was then used to label all WordNet's noun and verb definitions. For simplicity, example sentences and parentheses were excluded from the original glosses. The classification was performed over WordNet 3.0; 82,112 noun definitions and 13,761 verb definitions were labeled.

Data post-processing: since some of the classified definitions lacked the supertype role, the labeled data had to pass through a post-processing phase. The supertype is a mandatory component in a well-formed definition and, as will be detailed later, the RDF model is structured around it. Following the same syntactic rules adopted for pre-annotation, missing supertypes were identified and the roles around it had its limits adjusted, while the remaining classification was kept unchanged. Figure 4 shows an example of definition (for the term "spur") fixed in the post-processing phase.

RDF conversion: finally, the labeled definitions were serialized in RDF format. In the final graph, a synset is a node and each role in its definition is another node. The synset node is linked to the supertype role, which is, in turn, linked



Figure 4: Classified definition missing a supertype fixed in the post-processing phase.

to all the other roles. More specifically, a supertype linked to a role is a *reified* node, and this reified node is linked to the synset node. Reification is also used when a role has components, such as event time and/or location for a differentia event and quality modifier for a differentia quality. In this case, the component is linked to its main role, composing a reified node which is linked to the supertype, creating another reified node which is eventually linked to the synset node. This structure allows the relationships to be fully contextualized. As an example, consider the definition depicted in Figure 2. The node defined by the concept "poet" may be linked to several other nodes in the graph, but it is linked to the differentia quality node "English" only in the context of this definition. Supertype nodes are always represented as resources. The differentia quality and differentia event nodes can be represented as either resources, when they have components (event times and/or locations, or quality modifiers) to be linked to, or literals otherwise. All the other roles are represented as literals, and properties are named after role names[3]. Figure 5 shows the simplified (without reification) RDF representation for the definition in Figure 2.

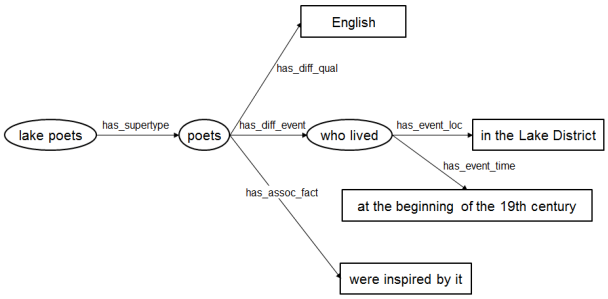


Figure 5: RDF representation for the definition of the "lake poets" synset.

Besides WordNetGraph, which is available in both XML and N-Triples format, we provide a set of tools[4] that implement the methodology described above. Routines for pre-processing definitions to generate sample data for manual curation, post-processing data returned by a machine learning classifier, and generating the RDF model from the classified data are freely available, along with some auxiliary routines to prepare the data for external tools, such as converting to the *standoff* file format required by the Brat annotation tool and generating a python script that will create the dataset for the RNN classifier.

[3]Complete list of the model's properties and namespaces at https://github.com/Lambda-3/WordnetGraph

[4]https://github.com/ssvivian/DefRelExtractor

5. Application

WordNetGraph is one of the main components in a text entailment recognition approach aimed at justifying entailment decisions where reasoning over world knowledge is required. Text entailment is defined as a directional relationship between an entailing text *T* and a entailed hypothesis *H*, holding true whenever a human reading T would infer that H is most likely true (Dagan et al., 2006). Using WordNetGraph as the world knowledge base, we implemented a navigation algorithm based on distributional semantics (Freitas et al., 2014) to find a path in this graph linking T to H, and used the contents of the nodes in this path to build a human-readable justification for the entailment decision. The entailment is rejected if no path is found.
Consider, as an example the entailment pair 39.3 from the BPI dataset[5]:

39.3 T: Many cellphones have built-in digital cameras.
39.3 H: Many cellphones can take pictures.

First, we look for pairs of terms that have a strong semantic relationship and that can prove this entailment to be true, and then send these pairs as input for the graph navigation algorithm. In this example, the best pair is composed by the terms "digital camera", which is our *source*, and "pictures", our *target*. Starting from the source, we retrieve all the nodes in WordNetGraph linked to it, compute the semantic similarity between each node and the target and choose the one having the highest value as the next node to be visited, and do this recursively until we reach the target. The following segments (triples) are found by the navigation algorithm:

<digital camera *has_supertype* camera>
<camera *has_supertype* equipment>
<equipment *has_diff_qual* for taking photographs>

Since "photograph" and "picture" are in the same synset node, the search stops at this point, confirming the entailment and producing the following justification, built from the path segments:

A digital camera is a kind of camera
A camera is an equipment for taking photographs
Photograph is synonym of picture

Experiments with the BPI dataset and a sample of the Guardian Headlines dataset[6] show the results are comparable to those of well-established text entailment algorithms, such as tree edit-distance based (Kouylekov and Magnini, 2005) and classification based (Wang and Neumann, 2008), while providing clear human-like explanations, an important feature still missing in most text entailment recognition approaches. A detailed description of the entailment recognition application, including experiment results and further justification examples can be found in (Silva et al., 2018).

[5]http://www.cs.utexas.edu/users/∼pclark/bpi-test-suite/

[6]https://goo.gl/4iHdbX

6. Conclusion

We presented a method for automatically building a graph world knowledge base from natural language dictionary definitions. Adopting a conceptual model based on entity-centered semantic roles, we trained a supervised machine learning classifier for automatic role labeling and then converted the labeled data into an RDF graph representation. Following this methodology, we created the WordNetGraph, a graph built from the definitions of nouns and verbs in WordNet. A set of tools implementing the methodology is also freely available.

WordNetGraph was successfully used in a text entailment recognition approach based on distributional navigation over definition graphs. Besides using paths in this graph to recognize the entailment, this approach also provides a human-readable justification for the entailment decision. Since each graph node encloses a self-contained amount of information rather than always representing single words, an intelligible justification can be built from a path made up by only a few nodes. As future work, we intend to apply this methodology to other language resources, such as Wiktionary.

7. Acknowledgments

Vivian S. Silva is a CNPq Fellow – Brazil.

8. Bibliographical References

Berg, J. (1982). Aristotle's theory of definition. *ATTI del Convegno Internazionale di Storia della Logica*, pages 19–30.

Bovi, C. D., Telesca, L., and Navigli, R. (2015). Large-scale information extraction from textual definitions through deep syntactic and semantic analysis. *Transactions of the Association for Computational Linguistics*, 3:529–543.

Calzolari, N. (1991). Acquiring and representing semantic information in a lexical knowledge base. In *Workshop of SIGLEX (Special Interest Group within ACL on the Lexicon)*, pages 235–243. Springer.

Copestake, A. (1991). The LKB: a system for representing lexical information extracted from machine-readable dictionaries. In *Proceedings of the ACQUILEX Workshop on Default Inheritance in the Lexicon, Cambridge*.

Dagan, I., Glickman, O., and Magnini, B. (2006). The pascal recognising textual entailment challenge. In *Machine learning challenges: evaluating predictive uncertainty, visual object classification, and recognising textual entailment*, pages 177–190. Springer.

Dolan, W., Vanderwende, L., and Richardson, S. D. (1993). Automatically deriving structured knowledge bases from on-line dictionaries. In *Proceedings of the First Conference of the Pacific Association for Computational Linguistics*, pages 5–14. Pacific Association for Computational Linguistics Vancouver.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Freitas, A., da Silva, J. a. C. P., Curry, E., and Buitelaar, P. (2014). A distributional semantics approach for selective reasoning on commonsense graph knowledge bases. In *International Conference on Applications of Natural Language to Data Bases/Information Systems*, pages 21–32. Springer.

Granger, E. H. (1984). Aristotle on genus and differentia. *Journal of the History of Philosophy*, 22(1):1–23.

Gunning, D. (2017). Explainable artificial intelligence (XAI). *Defense Advanced Research Projects Agency (DARPA)*.

Kouylekov, M. and Magnini, B. (2005). Recognizing textual entailment with tree edit distance algorithms. In *Proceedings of the First Challenge Workshop Recognising Textual Entailment*, pages 17–20.

Lloyd, A. C. (1962). Genus, species and ordered series in Aristotle. *Phronesis*, pages 67–90.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The Stanford Corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Màrquez, L., Carreras, X., Litkowski, K. C., and Stevenson, S. (2008). Semantic role labeling: an introduction to the special issue. *Computational linguistics*, 34(2):145–159.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 23(3):530–539.

Recski, G. (2016). Building concept graphs from monolingual dictionary entries. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA).

Silva, V. S., Handschuh, S., and Freitas, A. (2016). Categorization of semantic roles for dictionary definitions. In *Cognitive Aspects of the Lexicon (CogALex-V), Workshop at COLING 2016*, pages 176–184.

Silva, V. S., Freitas, A., and Handschuh, S. (2018). Recognizing and justifying text entailment through distributional navigation on definition graphs. In *AAAI*.

Vossen, P. and Copestake, A. (1994). Untangling definition structure into knowledge representation. In *Inheritance, defaults and the lexicon*, pages 246–274. Cambridge University Press.

Vossen, P. (1991). Converting data from a lexical database to a knowledge base. *Esprit BRA-3030 ACQUILEX Working Paper No 27*.

Vossen, P. (1992). The automatic construction of a knowledge base from dictionaries: a combination of techniques. In *EURALEX*, volume 92, pages 311–326.

Wang, R. and Neumann, G. (2008). An divide-and-conquer strategy for recognizing textual entailment. In *Proceedings of the Text Analysis Conference, Gaithersburg, MD*.

Combining rule-based and embedding-based approaches to normalize textual entities with an ontology

Arnaud Ferré[1,2], Louise Deléger[1], Pierre Zweigenbaum[2], Claire Nédellec[1]
[1]MaIAGE, INRA, Université Paris-Saclay, 78350 Jouy-en-Josas, France
[2]LIMSI, CNRS, Université Paris-Saclay, 91405 Orsay, France
arnaud.ferre@universite-paris-saclay.fr, louise.deleger@inra.fr, pierre.zweigenbaum@limsi.fr, claire.nedellec@inra.fr

Abstract

In this paper, we propose a two-step method to normalize multi-word terms with concepts from a domain-specific ontology. Normalization is a critical step of information extraction. The method uses vector representations of terms computed with word embedding information and hierarchical information among ontology concepts. A training dataset and a first result dataset with high precision and low recall are generated by using the ToMap unsupervised normalization method. It is based on the similarities between the form of the term to normalize and the form of concept labels. Then, a projection of the space of terms towards the space of concepts is learned by globally minimizing the distances between vectors of terms and vectors of concepts. It applies multivariate linear regression using the previously generated training dataset. Finally, a distance calculation is carried out between the projections of term vectors and the concept vectors, providing a prediction of normalization by a concept for each term. This method was evaluated through the categorization task of bacterial habitats of BioNLP Shared Task 2016. Our results largely outperform all existing systems on this task, opening up very encouraging prospects.

Keywords: ontology-based entity normalization, vector-based model, distributional semantics, multivariate linear model

1. Introduction

An important part of knowledge is expressed in textual form, such as in scientific articles. Specialized literature is characterized by the presence of terms of interest which are often complex nominal groups (e.g. "epithelial cells of the intestine") and display a high variability in their forms. At the same time, life sciences are a field with many structured, albeit incomplete, representations of knowledge: ontologies. These representations have the advantage of being machine interpretable, which can greatly improve the ability of programs to extract and reuse information from texts. The biomedical/biological field is therefore a good candidate for the development of more efficient generic methods that make use of these structured representations.

To extract entity information from texts, two steps are commonly applied: recognition of named entities and normalization of these entities (also called entity linking in the general domain). The recognition step detects terms of interest (e.g. bacterial habitat references) while normalization identifies them precisely by linking them to specific concepts or categories of an ontology (e.g. "T-cells" is a bacterial habitat mention that can be identified by the "lymphocyte" labeled concept). We are focusing on this particular task.

Today's best-performing normalization methods commonly rely on supervised learning from data that is manually annotated by experts of the field. However in specialized fields, these annotations are rare because they are difficult to obtain. Moreover, given the large number of target concepts in these fields (e.g. the biomedical metathesaurus UMLS contains more than 3 million concepts, the Ontobiotope ontology contains more than 2 thousand concepts, etc.), it seems unlikely to obtain sufficient data to cover all the possibilities of learning. An efficient and distant supervised method (i.e. with learning based on the results of an unsupervised method) would therefore be of great interest.

In this work, we propose several methods, notably a distant supervised normalization method, which we evaluated through the BioNLP 2016 Shared-Task (Deléger et al., 2016) and its task of bacterial habitat categorization. Habitat entities are often designated by complex nominal groups with variable forms, offering a relevant case study.

1.1 Related work

In the biomedical domain, normalization approaches relying on dictionaries and similarities of form between terms and labels from a knowledge source (Hanisch et al., 2005; Schuemie et al., 2007) are historically the oldest approaches and are still much used. They often provide good precision but a low recall because they have difficulty dealing with important variations in the form of terms/labels (e.g. synonymy such as "T-cells" / "lymphocyte", hyperonymy such as "Chondrus crispus" / "algae", etc.). These types of approach also often combine dictionaries with manually defined heuristic rules (Gerner et al., 2010; Kang et al., 2013). However rules are time-consuming to implement and highly dependent on the task and domain.

Statistical approaches aimed at learning directly the associations between terms and labels from corpora. Deep neural networks and word embeddings are those that have achieved the best performance at present (Mehryary et al., 2017), but the lack of training data and the large number of concepts seem to be a limitation on their potential for improvement.

Another recent method, CONTES (Ferré et al., 2017) is based on an approach that does not take into account the form of terms and concept labels, but only distributional information for terms, and hierarchical ontological information for concepts. It is based on the ability of the latest word embedding methods to generate relevant semantic spaces, as well as on building "concept embeddings" that preserves hierarchical information between concepts. This kind of method aims to overcome the problem of variability of forms of associated terms/labels, but does not otherwise take into account relevant morphological information.

In this work, we propose to improve the CONTES method by combining it with a rule-based approach, the ToMap method (Golik et al., 2011).

2. Material

Part of the data used is from the Bacteria Biotope categorization task of BioNLP Shared Task in 2016. The documents are MEDLINE references, consisting of titles and abstracts of scientific articles in the field of biology. The task is to assign concepts from the OntoBiotope ontology[1] to textual entities denoting bacterial habitats (entities are provided and do not have to be detected beforehand). The corpus is divided into two parts: a development corpus (combining the initial training and development corpora of the shared task) and a test corpus which we used to evaluate our method for the normalization task. The entities of each of these corpora have been annotated manually (see Table 1).

	BB		
	Dev.	**Test**	**Total**
Documents	107	54	161
Words	25,185	13,797	38,982
Entities	1,201	720	1,921
Distinct entities	743	478	1,125
Semantic categories	1,360	861	2,221
Distinct sem. categories	332	177	329

Table 1: Descriptive statistics for the Bacteria Biotope corpus ("Dev." = development corpus)

We used an expanded corpus to generate embeddings. It consists of all titles and abstracts of MEDLINE articles matching the MeshTerm "bacteria" from 2016 (see Table 2). The selection of this corpus was motivated by the need to use a corpus that is representative of the specific field of interest for this normalization task (i.e., habitats of bacteria). Considering that a high quantity of data for computing word embeddings does not guarantee a high quality in the biomedical field (Chiu et al., 2016), we chose to use this smaller targeted corpus as opposed to a larger and less relevant corpus.

sentences	7,714,841
raw words	154,749,541
raw words without stopwords	74,808,541
unique word (stopwords included)	1,565,740

Table 2: Descriptive statistics of extended corpus

3. Method

3.1 Rule-based approach: ToMap

As first approach to normalize entities, we used the ToMap method (Golik et al., 2011). ToMap relies on the internal morpho-syntactic structure of entities and maps them based on their syntactic heads, the underlying assumption being that the head is often the most informative component. The system first looks for a match between the syntactic head of an entity and the syntactic heads of the ontology concepts. Then, the entity is assigned the concept(s) with matching head(s). As there may be multiple candidates for a given entity, a Jaccard index is also computed between the entity and each of the concepts, and the concept with the highest score is selected.

The core algorithm is complemented by a set of heuristics designed to handle specific cases. For instance, a list of uninformative syntactic heads is provided so that if a head belongs to this list, then the algorithm tries to match the modifiers instead (e.g., in "water sample", "sample" is not very informative so "water" will be chosen to perform the mapping). Other heuristics include disambiguation rules targeted at particularly ambiguous terms (e.g., "plant" which can designate either a processing factory or living things such as trees, flowers, etc.). These heuristics are dependent on the type of entities (in our experiments, heuristics are designed for habitat entities). In the remainder of the article, we refer to the core algorithm alone (without the specific heuristics) as "simple ToMap".

Not all entities can be matched to an ontology concept with the ToMap method. When the syntactic head of an entity has no equivalent in the ontology, the entity cannot be normalized to a precise concept and is simply assigned the root concept of the ontology (i.e., "Bacteria habitat").

3.2 Embedding-based approach: CONTES (CONcept-TErm System)

3.2.1 Word embeddings with Word2Vec and Skip-Gram architecture

To train the word embedding method on the expanded corpus, the sentences were randomly shuffled and converted to lower case. Next, we applied the Word2Vec method with the Skip-Gram architecture (Mikolov et al., 2013). To evaluate our different methods, all based on embeddings, we chose to adopt the optimal parameters given in similar work on biomedical literature, which does not seem to drastically impact results (Chiu et al., 2016).

alpha	0.05
min-count	0
negative	5
sample	0.001
vector size	200
window size	2

Table 3: Main Word2Vec/Skip-Gram parameters used to calculate word embeddings for this work

We chose to keep all words, even those appearing once in the whole corpus, because we want to normalize entities that contain words with a low frequency and it does not seem to have a real impact on the global vector space. The most impacting parameters are the size of the vectors and the size of the contextual windows. We chose the vector size following the initial work on CONTES (Ferré et al.,

[1]http://2016.bionlp-st.org/tasks/bb2/OntoBiotope_BioNLP-ST-2016.obo?attredirects=0

2017), which obtained the best results with an output vector size of 200. However, we choose a smaller size of the context window, because of their possible potential to affect the nature of semantic proximity in the embedding space (Turney, 2012). Indeed, a common hypothesis is that larger window emphasizes the learning of domain similarity between words, while a narrow context window leads to hyponymic gathering. Thus, a smaller size of context window could be more preferable in our task.

All remaining parameters are those by default. Table 3 gives the main parameters.

3.2.2 Term and concept embeddings

The CONTES method computes embeddings for each word of the expanded corpus as described in the previous subsection. Then, for each entity of the training corpus and of the test corpus, a term vector is calculated by computing the barycenter of the vectors of the words that compose the entity. In parallel, concept vectors are calculated for all concepts of the ontology of interest, thus generating an ontological space. Each concept is associated with a vector of equivalent size to the number of concepts in the ontology. Each dimension is associated with a fixed concept. The vector is initialized as a one-hot (i.e. all weights are set to zero except the weight associated with the current concept, which is set to one), then each of the weights of the dimensions associated with the ancestors of the current concept are also set to one. The method encodes the hierarchical information of the ontology: if we estimate the cosine similarity between ontology concepts, the parent/child concepts are always the closest. This hierarchical information has the advantage of being the most frequent semantic relation (*is_a* relation) in ontologies.

After the generation of the term vectors (that represent the textual entities) and of concept vectors (that represent the concepts from the ontology), the method performs a projection of the term vectors into the ontological space. We use the training corpus to learn the optimal projection. That is, the projection that globally minimizes the distance between the projection of terms in the ontological space and the vectors of the associated concepts is determined. The learning method we used is a multivariate linear regression in order to limit the overfitting risk, particularly with regard to a relatively small training corpus. This projection is then applied to the term vectors of the test corpus, allowing to obtain new term vectors in the ontological space, and to calculate a cosine similarity with the concept vectors. The closest concept is chosen to normalize a term. The method has been designed to address the problem of term variability, because it does not rely on the similarity of form between terms and concept labels.

3.3 Combining the two approaches: HONOR (Hierarchical Ontological NORmalization)

The aim of the HONOR method is to take advantage of the precision of ToMap and to complement it with the CONTES method which has the potential to address the problem of form variability. More specifically, cases that are not handled by ToMap will be normalized by CONTES. Our hypothesis is that CONTES, which is not based on form similarity, should have the potential to propose relevant normalization predictions in cases where ToMap cannot. The overall scheme of the method is shown in Figure 1.

3.4 Distant supervision version

Additionally, as ToMap is not supervised and potentially yields good precision results, we can also use it to generate a first prediction on a larger corpus, which will be used to train CONTES rather than using the gold standard. In this setting, the method becomes a distant supervised method, as it would not rely on manual annotation anymore to train its learning model. We chose to use the same corpus as the one used for the training of Word2Vec. We tested for many random subsamples of the full predictions of ToMap on this corpus and analyzed the impact of the number of selected examples on the global performance of these methods.

We tested four different versions of the distant supervised method: two versions are based on the unsupervised predictions of ToMap without using its supplementary heuristics (simple ToMap), and the two others are based on the predictions of the complete version of ToMap. For each configuration (simple ToMap vs. complete ToMap), there is one version which only uses the CONTES method, and another version which uses the combined approach of HONOR. For both versions of the HONOR method, we studied the impact of the number of examples (i.e. the predictions of ToMap on the expanded corpus). The smallest batches have the same order of magnitude as the gold standard and the biggest batch is a hundred times bigger.

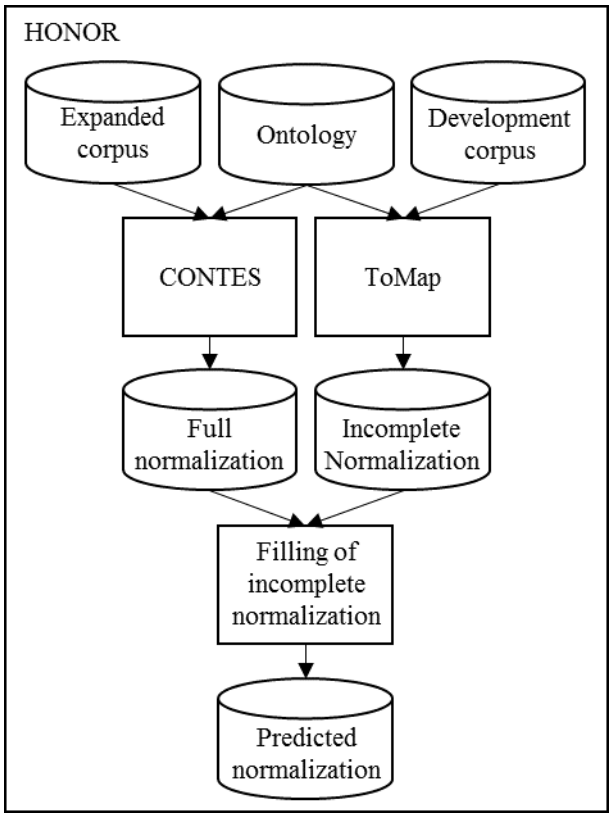


Figure 1: Global schema of the method HONOR

An implementation of the ToMap and CONTES methods is available via the AlvisNLP/ML[2] engine (Ba et al., 2016).

[2] https://bibliome.github.io/alvisnlp/

You need to download the module for the CONTES method[3]. AlvisNLP/ML allows to combine the two methods and to emulate HONOR.

4. Results

4.1 Evaluation

We evaluated the performance of ToMap and CONTES individually and of our combined approach HONOR (both in its supervised and distant supervised settings), on the Bacteria Biotope normalization task of the BioNLP Shared Task 2016. The predicted concept identifiers are compared to the gold standard concepts according to the similarity measure of (Wang et al., 2007), with the weight parameter set to 0.65. The evaluation was performed by submitting our results to the evaluation server of the BioNLP-ST 2016 challenge site. We computed baseline results by assigning all terms to the concept "bacteria habitat", which is the root of the OntoBiotope ontology hierarchy. The baseline obtained a score of 32% (Table 6). For comparison, we also included results from other systems that proposed methods to normalize bacterial habitats and evaluated them on the shared task corpus (Grouin, 2016; Tiftikci et al., 2016; Mehryary et al., 2017).

4.2 ToMap and CONTES individually

ToMap can provide a normalization prediction for 54% of the terms. Even if there are still problems of ambiguity, it enables us to estimate the proportion of terms that have a form similar to an ontology label at least at the level of their syntactic head. Despite its limitation, ToMap obtained good performances on the test corpus and ranked above existing methods (Table 6). This shows that the method has a really good precision when it can provide a prediction.

As long as terms are composed with tokens from the initial vocabulary, which allows to calculate an embedding for the terms, CONTES can provide a normalization prediction for all terms to normalize. The method obtained a score of 61% (Table 6). Compared to previous work, this version of CONTES includes a few improvements (refined embedding parameters and extended training corpus). We also tested the impact of the hierarchical information encoded in concept vectors to evaluate the gain when using this information compared to a simple one-hot representation (Table 4). Indeed, a one-hot encoding does not use hierarchical information because all vectors are equidistant from each other. The improvement of 7% validates the usefulness of this information to improve the matching between the two vector spaces.

Taking into account hierarchical information	0.61
One-hot encoding	0.54

Table 4: Comparison between the CONTES method encoding of concepts and an approach with one-hot encoding of concepts

4.3 HONOR

We compared the results of HONOR to those of existing systems. We report all results in Table 5 and Table 6. In 2016, two teams participated in this task. The best method was the BOUN method (Tiftikci et al., 2016) which combined a form similarity approach and an information-retrieval based approach (based on tf-idf) and obtained a 0.61 score. The method of LIMSI (Grouin, 2016), which is based on a form similarity approach, had the lowest performance (0.43). Since the 2016 shared task, the University of Turku has proposed an end-to-end neural network method (Mehryary et al., 2017) which has outperformed these methods with a 0.63 score.

Compared to these systems, our method performs well above by a 10 points increase compared to the Turku system. It also brings a significant gain to the ToMap method (+7 points in the supervised version and +6 points in the distant supervised version). There is only a one-point difference between the supervised and distant supervised HONOR methods, which shows that our method could enable to perform without manual annotated data and without a significant loss of performance.

Method	Score
Unsupervised	
ToMap	0.66
Simple ToMap	0.62
Distant Supervised	
HONOR (ToMap)	0.72
HONOR (simple ToMap)	0.66
CONTES (ToMap)	0.61
CONTES (simple ToMap)	0.59
Supervised	
HONOR (ToMap)	**0.73**
CONTES (improved)	0.61

Table 5: Results of all the methods described in this article on the normalization task of BioNLP-ST 2016

System	Score
Supervised HONOR	**0.73**
Distant supervised HONOR	**0.72**
Turku	0.63
BOUN	0.62
CONTES (2017)	0.60
LIMSI	0.43
Baseline	0.32

Table 6: Results on the normalization task of BioNLP-ST 2016

4.4 Impact of the number of examples in distant supervision

For the two distant supervised versions of HONOR, we evaluated performances obtained with three different sizes of data batches. These batches have been constituted by randomly choosing a variable number of examples in the predictions of ToMap on the expanded corpus. For each batch size, variations on the score have been estimated over many executions. The results remain relatively stable across different batches of the same size: less than 0.2% variation for the biggest batch and less than 1.5% for the smallest. It seems that there is a small gain for both versions

[3] https://github.com/ArnaudFerre/CONTES

to use 10^5 rather 10^4 examples. The means of these results are reported in Table 7:

Number of predictions from ToMap:	**10^3**	**10^4**	**10^5**
HONOR with simple ToMap:	0.64	0.66	0.66
HONOR with ToMap:	0.70	0.72	0.72

Table 7: Results for the two distant supervised version of HONOR.

5. Discussion and future work

The key hypothesis behind the efficiency of the CONTES and HONOR methods is that the semantic space of terms (based on distributional semantics) and the semantic space of the ontology (based on the specific hierarchical information of the current ontology) are homologous. Even if this work seems to indicate that there are at least some similarities between these two vector spaces, this hypothesis is most likely too strong. It would certainly be interesting to alter the space of embeddings to increase this similarity and then use a non-linear algorithm to find a better projection from the embedding space to the ontological space.

With this kind of word embedding-based approach, there is a problem with computing words that have not been met before (i.e. out-of-vocabulary words). Consequently, any new word encountered should require a complete recalculation of word embeddings to be taken into account. Recently, a new method to calculate embeddings seems to overcome this difficulty (Bojanowski et al., 2016) and we plan to estimate its performance on our method in further work.

Like the CONTES method, a problem of high dimensional representation of the concept vectors persists. We plan to address this issue in further work.

The bacterial habitat normalization task includes the normalization of an entity by multiple concepts. Currently, no system has successfully addressed this issue. We would like to investigate this problem in the future.

Finally, beyond the benefits of using a multivariate linear regression, work is underway to explore the possibility that a non-linear learning method can provide a better projection between vector space of terms and vector space of concepts.

6. Conclusion

Our method seems to open up interesting perspectives for joint use of word embeddings, ontologies and form similarity based methods, particularly for the domain specific literature. This kind of synergy could also address challenges in fields where sufficient manual annotations are difficult to obtain.

7. Acknowledgment

This work is supported by the "IDI 2015" project funded by the IDEX Paris-Saclay, ANR-11-IDEX-0003-02.

8. Bibliographical References

BA MOUHAMADOU, BOSSY ROBERT (2016), Interoperability of Corpus Processing Workflow Engines: The Case of. AlvisNLP/ML in OpenMinTeD, np, in: *Meeting of working Group Medicago sativa*.

BOJANOWSKI PIOTR, GRAVE EDOUARD, JOULIN ARMAND, MIKOLOV TOMAS (2016), Enriching Word Vectors with Subword Information. *ACL 2017*.

CHIU BILLY, CRICHTON GAMAL, KORHONEN ANNA, PYYSALO SAMPO (2016), How to Train Good Word Embeddings for Biomedical NLP. *Proceedings of BioNLP16*: 166.

DELÉGER LOUISE, CHAIX ESTELLE, BA MOUHAMADOU, FERRÉ ARNAUD, BESSIÈRES PHILIPPE, NÉDELLEC CLAIRE (2016), Overview of the Bacteria Biotope Task at BioNLP Shared Task 2016.

FERRÉ ARNAUD, ZWEIGENBAUM PIERRE, NÉDELLEC CLAIRE (2017), Representation of Complex Terms in a Vector Space Structured by an Ontology for a Normalization Task. *BioNLP 2017*: 99–106.

GERNER MARTIN, NENADIC GORAN, BERGMAN CASEY M. (2010), LINNAEUS: A Species Name Identification System for Biomedical Literature. *BMC bioinformatics*, 11(1): 85.

GOLIK WIKTORIA, WARNIER PIERRE, NÉDELLEC CLAIRE (2011), Corpus-Based Extension of Termino-Ontology by Linguistic Analysis: A Use Case in Biomedical Event Extraction, 37–39, in: *WS 2 Workshop Extended Abstracts, 9th International Conference on Terminology and Artificial Intelligence*.

GROUIN CYRIL (2016), Identification of Mentions and Relations between Bacteria and Biotope from PubMed Abstracts. *ACL 2016*: 64.

HANISCH DANIEL, FUNDEL KATRIN, MEVISSEN HEINZ-THEODOR, ZIMMER RALF, FLUCK JULIANE (2005), ProMiner: Rule-Based Protein and Gene Entity Recognition. *BMC Bioinformatics*, 6(Suppl 1): S14.

KANG NING, SINGH BHARAT, AFZAL ZUBAIR, VAN MULLIGEN ERIK M, KORS JAN A (2013), Using Rule-Based Natural Language Processing to Improve Disease Normalization in Biomedical Text. *Journal of the American Medical Informatics Association*, 20(5): 876–881.

MEHRYARY FARROKH, HAKALA KAI, KAEWPHAN SUWISA, BJÖRNE JARI, SALAKOSKI TAPIO, GINTER FILIP (2017), End-to-End System for Bacteria Habitat Extraction. *BioNLP 2017*: 80.

MIKOLOV TOMAS, CHEN KAI, CORRADO GREG, DEAN JEFFREY (2013), Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.

SCHUEMIE MARTIJN J., JELIER ROB, KORS JAN A. (2007), Peregrine: Lightweight Gene Name Normalization by Dictionary Lookup, 131–133, in: *Proc of the Second BioCreative Challenge Evaluation Workshop*.

TIFTIKCI MERT, SAHIN HAKAN, BÜYÜKÖZ BERFU, YAYIKÇI ALPER, OZGÜR ARZUCAN (2016), Ontology-Based Categorization of Bacteria and Habitat Entities Using Information Retrieval Techniques. *ACL 2016*: 56.

TURNEY PETER D. (2012), Domain and Function: A Dual-Space Model of Semantic Relations and Compositions. *Journal of Artificial Intelligence Research*.

WANG JAMES Z., DU ZHIDIAN, PAYATTAKOOL RAPEEPORN, YU PHILIP S., CHEN CHIN-FU (2007), A New Method to Measure the Semantic Similarity of GO Terms. *Bioinformatics*, 23(10): 1274–1281.

T-REx: A Large Scale Alignment of Natural Language with Knowledge Base Triples

Hady Elsahar*, Pavlos Vougiouklis†, Arslen Remaci*,
Christophe Gravier*, Jonathon Hare†, Elena Simperl†, Frederique Laforest*
*Université de Lyon, Laboratoire Hubert Curien, Saint-Étienne, France
†Electronics and Computer Science, University of Southampton, UK
{hady.elsahar, christophe.gravier, frederique.laforest}@univ-st-etienne.fr
arslen.remaci@etu.univ-st-etienne.fr
{pv1e13, jsh2, e.simperl}@ecs.soton.ac.uk

Abstract

Alignments between natural language and Knowledge Base (KB) triples are an essential prerequisite for training machine learning approaches employed in a variety of Natural Language Processing problems. These include Relation Extraction, KB Population, Question Answering and Natural Language Generation from KB triples. Available datasets that provide those alignments are plagued by significant shortcomings – they are of limited size, they exhibit a restricted predicate coverage, and/or they are of unreported quality. To alleviate these shortcomings, we present *T-REx*, a dataset of large scale alignments between Wikipedia abstracts and Wikidata triples. *T-REx* consists of 11 million triples aligned with 3.09 million Wikipedia abstracts (6.2 million sentences). *T-REx* is two orders of magnitude larger than the largest available alignments dataset and covers 2.5 times more predicates. Additionally, we stress the quality of this language resource thanks to an extensive crowdsourcing evaluation. *T-REx* is publicly available at `https://w3id.org/t-rex`.

Keywords: Knowledge Base Population, Relation Extraction, Distant Supervision, Wikidata

1. Introduction

Reducing the gap between Natural Language and structured knowledge bases (KB) has been the concern of many research tasks such as: Relation Extraction [Mintz et al.2009], KB Population [Ji and Grishman2011], KB-driven Natural Language Generation [Lebret et al.2016] and Question Answering [Xu et al.2016]. Models built for these tasks rely on training datasets containing alignments between sentences in free text and KB triples. Previous works [Mintz et al.2009, Yao et al.2011] have created alignments either manually or automatically for the purpose of training and evaluating their models. [Augenstein et al.2016, Martin et al.2016] have pointed out shortcomings of existing alignments language resources, and showed the importance of building a dataset of high quality large scale alignments. Such shortcomings include: 1) their limited size in terms of the number of alignments, 2) their limited coverage as the number of represented predicates is not enough to generalize to larger domains (usually such datasets are very biased towards few predicates), and/or 3) their either low or unreported quality.

In this work, we build *T-REx*, a large scale alignment dataset between free text documents and KB triples. *T-REx* consists of 3.09 million Wikipedia abstracts aligned with 11 million Wikidata triples, covering more than 600 unique Wikidata predicates. T-REx is two orders of magnitude larger than the largest available alignments dataset, and covers 2.5 times more predicates. In this paper, we define the customizable architecture of the alignment pipeline which uses three different automatic alignment techniques. We evaluate the quality of *T-REx* by running a crowdsourcing experiment over 2,600 created alignments. The best automatic alignment technique in *T-REx* achieved an accuracy of 97.8% over the evaluated subset of the dataset.

2. Related Work

A considerable body of work has created alignments between free text and KB triples. The *TAC-KBP* dataset [Li et al.2012] is built from news wire and web forums. The dataset is generated as a bi-product of the evaluation process of the TAC KB population competition[1], where human annotators evaluate the output of each competing system. The dataset is limited in size as it consists of only 5 classes and 41 predicates. Several works [Mintz et al.2009, Yao et al.2011] have aligned the New York Times corpus with Freebase triples, resulting in several variations of the same dataset, *NYT-FB*. This dataset is prone to bias and coverage issue since the Named Entity linking used for its construction is based on keyword matching against Freebase labels. For example, 30.7% of the alignments are for the sole predicate "/location/country". The *FB15K-237* [2] dataset [Toutanova et al.2015] contains alignments of the Clueweb dataset with Freebase-named entities [Gabrilovich et al.2013] and Freebase triples. The dataset is of relatively large size (2.7 million alignments); however, it lacks the original text from which the alignments are derived – This makes it unsuitable for some applications such as natural language generation. *Google-RE*[3] is a Google dataset with 60K sentences from Wikipedia, manually aligned with Freebase. Despite its high quality, the dataset is labeled for only five Freebase relations. *WikiReadings* [Hewlett et al.2016] is another dataset containing rough alignments created by replacing each subject of a Wikidata triple by the

[1] `http://bit.ly/tackbpcompetition`
[2] `https://www.microsoft.com/en-us/download/details.aspx?id=52312`
[3] `https://code.google.com/archive/p/relation-extraction-corpus`

Dataset	Documents / Format	Unique predicates	Aligned Triples	Available
NYT-FB	1.8M sent.	258	39K	partially
TAC KBP	90K sent.	41	122K	closed
Google-RE	60K sent.	5	60K	publicly
FB15K-237	2.7 M patterns	237	2.7M	publicly
Wikireadings	4.7M articles	884	n.a.	publicly

Table 1: Statistics over existing alignments from previous work.

whole text of its Wikipedia article. Despite its large size, the dataset does not contain actual alignments between text and KB triples as there is no way to tell whether all the mentioned triples appear in the text, nor, if applicable, their location in the original text. Table 1 lists different alignments along with their size and coverage.

3. *T-REx* Creation

T-REx creation pipeline (Figure 1) contains components for document reading, entity extraction, and dataset exportation into different formats are described at Section 3.1. while triple aligners – key components of the system – are presented in Section (3.2.).

3.1. Alignment pipeline

Document Reader: It gets documents from a dump and outputs in an format readable by all components. Also, it includes sentence and word tokenizers to extract the start and end positions of sentences and words in documents.

Entity Extraction: For each document, we extract named entities in the text and link them to their URI with the DBpedia Spotlight [Mendes et al.2011] entity linker.

Date and Time Extraction: We use the Stanford temporal tagger SUtime [Chang and Manning2012] to extract temporal expressions and their locations in documents. We normalize them to the XSD Date and Time Data Type format as expressed in most KB.

Predicate Linking: A sentence is more likely to express a KB triple if the label of the predicate forming this triple matches with any sequence of words in that sentence. A predicate linker links a sequence of words in a paragraph to its equivalent KB predicate URI if it matches the predicate label or any of its aliases in the KB.

Coreference Resolution: We use the Stanford CoreNLP co-reference resolution component [Manning et al.2014], coupled with a robust heuristic inspired from [Augenstein et al.2016]. We map a list of possible pronouns to each KB entity according to values of specific predicates such as "gender" and "instance of". Then, we link each pronoun in a sentence to its document main entity if they map.

Triple Aligners: Triple aligners are the main components of our pipeline: each provided document is aligned with a set of KB triples expressed in the document alongside with their locations. They are described in the next subsection.

Document Writers: They export documents with annotation in standard formats. We propose a plain JSON format and NIF 2.0 [Hellmann et al.2013], a RDF/OWL-based standard annotation format for natural language processing.

3.2. Triple Aligners

Let $t_{xyz} = (e_x, e_y, e_z) \subset \mathcal{E}X\mathcal{P}X\mathcal{E}$ be one of all possible triples in a KB where $\mathcal{E} = \{e_i, ...e_n\}$ and $\mathcal{P} = \{p_i, ...p_n\}$ be the sets of all entities and properties represented in the KB respectively. Given a corpus of text documents, each document d contains a set of sentences $d = \{s_i, ..s_n\}$, a main entity e_{doc} and a set of linked entities $\mathcal{E}_{doc} = \{\mathcal{E}_i, ..., \mathcal{E}_n\}$ where $\mathcal{E}_i$ is the set of entities linked in sentence s_i.
Following [Augenstein et al.2016], we explore different methodologies to create those alignments using the distant supervision assumption. Distant supervision creates a set of alignments A between all triples whose subject and object entities are in the set of tagged entities in this sentence, i.e. $A = \{(s_i, t_{xyz}) |\ e_x \in \mathcal{E}_i \wedge e_z \in \mathcal{E}_i\}$.

NoSub Aligner: In practice the subject entity is usually mentioned once at the beginning of the paragraph and is often referred implicitly or using pronouns. These implicit lexicalizations can hardly be detected by entity linkers, and lead to a coverage issue. The NoSub aligner relaxes the distant supervision assumption and assumes that sentences in one paragraph often have the same subject. It extracts a set of alignments $A = \{(s_i, t_{xyz}) | (e_x = e_{doc} \wedge e_z \in \mathcal{E}_i) \vee (e_z = e_{doc} \wedge e_x \in \mathcal{E}_i)\}$. This relaxation comes at a price: the position of the subject entity in each aligned triple is not known as the aligner assumes it is implicitly mentioned.

AllEnt Aligner: Every pair of entities in a sentence is considered in alignment and mapped to their equivalent KB relations. For implicit mentions of entities, we use co-reference resolution to extract all mentions of the main entity of the paragraph. Given $\mathcal{E}' = \mathcal{E}_i \cup \mathcal{E}_i^{coref}$ the union of the sets of entities in the sentence through named entity linking and co-reference resolution, AllEnt extracts a set of alignments $A = \{(s_i, t_{xyz}) | e_x \in \mathcal{E}' \wedge e_z \in \mathcal{E}'\}$.

SPO Aligner: The alignment of every pair of entities as shown in Table 2 Examples 8 & 9 can sometimes be noisy:

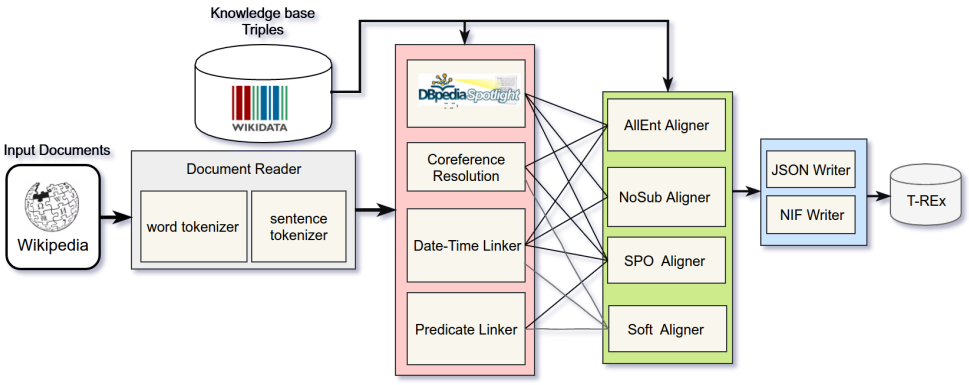


Figure 1: Overview of the alignment pipeline and its components

it aligns triples that are not necessarily mentioned in the sentence. For that, the SPO Aligner aligns triples not only when the subject and object of a triple are mentioned in a sentence but also when the predicate of the triples has been extracted. Given $\mathcal{P}_i \subset \mathcal{P}$ the set of predicates tagged in the sentence s_i using the predicate linker, the SPO Aligner creates a set of alignments $A = \{(s_i, t_{xyz}) | e_x \in \mathcal{E}_i \wedge p_y \in \mathcal{P}_i \wedge e_z \in \mathcal{E}_i\}$.

4. *T-REx* Dataset

We feed the pipeline with documents from the DBpedia Abstracts dataset [Brümmer et al.2016], an open corpus of annotated Wikipedia texts. We use its English section, containing $4.6M$ text documents. As a source of triples, we use the Wikidata truthy dump[4] containing 144M triples. The result of the alignment process is *T-REx*, a large dataset with alignments of KB with free text, provided from the three alignment techniques previously presented.

4.1. Size and Coverage

In Table 3, we compare the number of alignments in the *T-REx* dataset with the largest datasets of the literature NYT-FB and TAC-KBP. All of the 3 alignment techniques proposed in *T-REx* have reported a substantial larger number of alignments than the two other datasets. The largest number of alignments was achieved by the AllEnt aligner with 11.1M alignments. In terms of coverage, the NoSub Aligner recorded 642 predicates. This makes *T-REx* two orders of magnitude larger than the largest available alignments, representing 2.5 times more predicates. Moreover, having a significant number of examples for each predicate is of the utmost practical interest for training high coverage models, regardless the NLP task at hand. In Figure 2, we illustrate the gap between *T-REx* and prior datasets on the predicate coverage criteria by plotting the distribution of the number of alignments created for each predicate. *T-REx* has substantially more examples than the other datasets, not only for the most common predicates but also for the long tail ones, which is of the utmost practical interest for the NLP practitioner.

[4] https://dumps.wikimedia.org/wikidatawiki/entities/20170503/

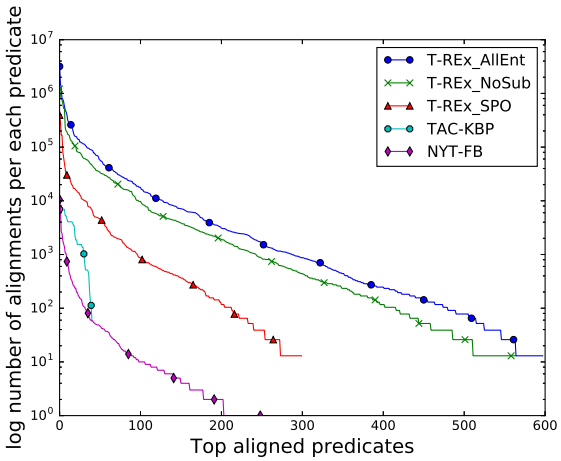


Figure 2: Distribution of the number of alignments created for each predicate

4.2. Availability and Licensing

T-REx is publicly available under a Creative Commons Attribution-ShareAlike 4.0 International License on the following persistent address https://w3id.org/t-rex and registered at Datahub https://datahub.io/dataset/t-rex. Its alignment pipeline code is available under the MIT License[5].

5. Evaluation

In order to evaluate the quality of *T-REx* we have led a crowdsourcing experiment on a subset of the alignments comprised of 2,600 aligned triples distributed over our three alignment techniques from 700 Wikipedia abstracts. In order to make sure that this sample is not biased towards one type of documents or a predicate, we made sure that the randomly selected evaluation sample has the same mean and median values of aligned triples per document and number of words per document, as the whole dataset. We asked contributors[6] to read each document carefully and annotate each alignment to be true only if the triple is explicitly mentioned in the given document. Each alignment is being annotated at least 5 times. For example, given the sentence

[5] https://github.com/hadyelsahar/RE-NLG-Dataset

[6] Instruction page: http://bit.ly/2pBOZpx

David Bowie was an English singer, who later on [worked as] an actor. He was [born in] Brixton, London to his [mother] Margaret Mary and his [father] Haywood Stenton.			
# Triples	**NoSub**	**AllEnt**	**SPO**
1) wd:David_Bowie wdt:nationality wd:England .	x	x	
2) wd:David_Bowie wdt:occupation wd:singer .	x	x	
3) wd:David_Bowie wdt:occupation wd:Actor .	x	x	x
4) wd:David_Bowie wdt:birthPlace wd:Brixton .	x	x	
5) wd:Brixton wdt:region wd:London .		x	
6) wd:David_Bowie is wdt:child_of wd:Margaret_Mary .	x	x	x
7) wd:David_Bowie is wdt:child_ of wd:Haywood_Stenson .	x	x	x
8) *wd:Margaret_Mary wdt:Divorce wd:Haywood_Stenson .*		x	
9) *wd:Margaret_Mary wdt:deathPlace wd:London .*		x	

Table 2: Comparison between different extractions of three alignment schemes for a sample paragraph of two sentences. The detected properties in the paragraph are put between square brackets. Wrong alignments are in italic.

Annotator	Documents covered	Alignments	Numerical Alignments	Uniq predicates
NYT-FB	1.8M	39K	None	258
TAC-KBP	0.09M	122K	n.a.	41
*T-REx*_SPO	0.79M	1.2M	21K	336
*T-REx*_NoSub	2.85M	5.2M	561K	642
*T-REx*_AllEnt	**3.09M**	**11.1M**	**350K**	**633**

Table 3: Number of alignments in different datasets

"Jonathan Swift was born in Dublin, Ireland", the triple `"Ireland, Capital of, Dublin"` should be annotated as False as it is not directly implied from the sentence. To guarantee high quality annotations, we manually annotated 100 documents and used them to filter out spammers and non-qualified contributors. One of each 4 questions given to a contributor contains a test question, contributors who score less than 80% accuracy on these questions were disqualified from the crowdsourcing experiment. Table 4 shows the accuracy of each alignment methodology and its corresponding inter annotator agreement I, calculated through the following formula:

$$I = 1 - \frac{\sum_{i=0}^{N} |\frac{f_i}{a_i} - t_i|}{N} \quad (1)$$

where $t_i \in \{0, 1\}$ is the value of the majority vote for the alignment i, $f_i \in [0, a_i]$ is the number of times the alignment was labeled as True and a_i is the number of manual annotators for it. N is the total number of alignments being annotated. The NoSub Aligner has scored the top accuracy scoring 97.8%, compared to 95.7% for the SPO Aligner, let alone that the Nosub Aligner has almost 4 times more extractions. However, the SPO Aligner has the advantage of extracting the positions of the subject, predicate and the object in the text, which makes it more suitable for training extractive models for Relation Extraction and Question Answering [Rajpurkar et al.2016]. Table 5 shows the alignment accuracy of top occurring predicates along side with inter annotator agreement.

	AllEnt	SPO	NoSub
Accuracy	0.88	0.96	0.98
Inter-Annotator	0.85	0.93	0.96

Table 4: Accuracy of each alignment methodology.

Property Label	AllEnt	SPO	NoSub	Inter ann.
located in	0.95	1.00	1.00	0.90
member of sports team	1.00	0.99	0.99	0.97
date of birth	1.00	1.00	1.00	0.97
date of death	1.00	1.00	0.99	0.98
country of citizenship	0.91	1.00	0.95	0.92
educated at	0.88	0.92	1.00	0.92
occupation	0.90	0.94	1.00	0.93
spouse	0.75	0.94	1.00	0.92
capital	**0.40**	**1.00**	n.a.	0.82
shares border with	**0.14**	**1.00**	n.a.	**0.69**

Table 5: Accuracy of top properties for each annotation methodology in T-REx

6. Conclusion & Future Work

In this paper, we present *T-REx* a dataset of large scale alignments of Wikipedia Abstracts with KB Facts represented in Wikidata Triples. *T-REx* consists of three types of alignments made by three automatic alignment hypotheses. T-REx is unmatched in size and in the number of represented predicates. Moreover, although its significant size with respect to its counterparts, T-REx offers a very high quality of its alignments – a crowdsourcing experiment on 2,600 alignments exhibits a 97.8% accuracy with a high inter-annotator agreement (ranging from 0.854 to 0.962 depending on the triple aligners used in the process). T-REx also provides an extensive evaluation through a crowdsourcing experiment, through which T-REx showed to have high quality alignments reaching 97.8% accuracy.

To plan our future work, we handpicked a sample of wrong align-

	Alignment	Cause of error
1)	He was the son of Ekoji I as well as the younger brother of Serfoji I. (Ekoji I, child(ren), Serfoji I)	Nested Relations
2)	Ernst Gustav Kuhnert was born in Tallinn, Estonia (Tallinn, Capital of, Estonia)	Wrong Entailment
3)	Carolyn Virginia Wood (born December 18, 1945) is an American.. (Virginia, country, American)	Entity Linking

Table 6: Causes of error in alignments

ments to analyze their main causes. We noticed three main causes of alignment errors: 1) Nested relations errors, where multiple relations in a short sentence share the same entities e.g. Table 6 example 1. This can be alleviated by creating aligners who take into consideration the linguistic structure of the sentence such as dependency paths. 2) Wrong entailment, where the aligners aligns triples that do not imply the sentence, as shown on example 2 in Table 6. This can be alleviated through incorporating implication rules in the alignment process [Demeester et al.2016]. 3) Entity linking errors like in example 3 of Table 6. Alleviating these three main types of errors is the main future directions of T-REx enhancement.

7. Bibliographical References

Augenstein, I., Maynard, D., and Ciravegna, F. (2016). Distantly supervised web relation extraction for knowledge base population. *Semantic Web*, 7(4):335–349.

Brümmer, M., Dojchinovski, M., and Hellmann, S. (2016). Dbpedia abstracts: A large-scale, open, multilingual nlp training corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France.

Chang, A. X. and Manning, C. (2012). Sutime: A library for recognizing and normalizing time expressions. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may.

Demeester, T., Rocktäschel, T., and Riedel, S. (2016). Lifted rule injection for relation embeddings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1389–1399.

Gabrilovich, E., Ringgaard, M., and Subramanya, A. (2013). FACC1: Freebase annotation of ClueWeb corpora, Version 1, June.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, pages 98–113.

Hewlett, D., Lacoste, A., Jones, L., Polosukhin, I., Fandrianto, A., Han, J., Kelcey, M., and Berthelot, D. (2016). Wikireading: A novel large-scale language understanding task over wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Ji, H. and Grishman, R. (2011). Knowledge Base Population: Successful Approaches and Challenges. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 1148–1158.

Lebret, R., Grangier, D., and Auli, M. (2016). Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1203–1213.

Li, X., Strassel, S., Ji, H., Griffitt, K., and Ellis, J. (2012). Linguistic resources for entity linking evaluation: from monolingual to cross-lingual. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, System Demonstrations*, pages 55–60.

Martin, T., Botschen, F., Nagesh, A., and McCallum, A. (2016). Call for discussion: Building a new standard dataset for relation extraction tasks. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction, AKBC@NAACL-HLT 2016, San Diego, CA, USA, June 17, 2016*, pages 92–96.

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011). DBpedia Spotlight: Shedding Light on the Web of Documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8. ACM.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proc. of ACL 2009*, pages 1003–1011.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392.

Toutanova, K., Chen, D., Pantel, P., Poon, H., Choudhury, P., and Gamon, M. (2015). Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1499–1509.

Xu, K., Reddy, S., Feng, Y., Huang, S., and Zhao, D. (2016). Question answering on freebase via relation extraction and textual evidence. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Yao, L., Haghighi, A., Riedel, S., and McCallum, A. (2011). Structured relation discovery using generative models. In *Proc. of EMNLP 2011*.

Multilingual Parallel Corpus for Global Communication Plan

Kenji Imamura and Eiichiro Sumita
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0289, Japan
{kenji.imamura,eiichiro.sumita}@nict.go.jp

Abstract

In this paper, we introduce the Global Communication Plan (GCP) Corpus, a multilingual parallel corpus being developed as part of the GCP. The GCP Corpus is intended to be develop speech translation systems; thus, it primarily consists of pseudo-dialogues between foreign visitors and local Japanese people. The GCP Corpus is sentence-aligned and covers four domains and ten languages, including many Asian languages. In this paper, we summarize the GCP and the current status of the GCP Corpus. Then, we describe some of the corpus' basic characteristics from the perspective of multilingual machine translation and compare direct, pivot, and zero-shot translation techniques.

Keywords: Multilingual Parallel Corpus, Global Communication Plan, Asian Languages, Pivot/Zero-shot Translation

1. Introduction

In 2014, the Ministry of Internal Affairs and Communications in Japan implemented the Global Communication Plan (GCP; c.f., Sec. 2.), whose mission is to eliminate global "language barriers." An important aspect of this plan is the public demonstration of multilingual speech translation technologies. With this in mind, we are improving the usability of the multilingual speech translation system by improving translation quality and developing user interfaces that enable it to be used in various places, such as hospitals, malls/stores, and tourist spots.

Multilingual translation corpora are required to develop multilingual speech translation systems, and the GCP includes a corpus development program for machine translation (MT).

In this paper, we summarize the GCP and the current status of the GCP Corpus, which is a multilingual parallel corpus used for public demonstration [1]. This corpus primarily consists of pseudo-dialogues between foreign visitors and local Japanese people. (It also includes some isolated utterances, such as phrases frequently associated with travelling.) The utterances in the dialogues are translated into ten languages (including Japanese) targeted by the GCP. Therefore, the corpus is sentence-aligned. The target domains are medical care, disaster prevention, shopping, and tourism.

Although the GCP Corpus is being developed for use in speech translation systems, it could also be used in various other research fields because it has the following characteristics.

1. It is a multilingual sentence-aligned corpus that covers ten languages, including Asian languages. Therefore, this allows 90 different MT systems to be constructed. Furthermore, it can also be applied in comparative studies of pivot translation (Utiyama and Isahara, 2007; Cohn and Lapata, 2007) and zero-shot translation (Johnson et al., 2016).
2. It covers four domains, namely, medical care, disaster prevention, shopping, and tourism. It can be applied to domain adaptation studies (e.g., (Imamura and Sumita, 2016)).
3. It consists of pseudo-dialogues. Therefore, it can also be applied to discourse studies that consider long-distance contexts. Note that such contexts are simpler than those of real dialogues because the dialogue never breaks down (Higashinaka et al., 2016).

In this paper, we focus on the first characteristic. We use the GCP Corpus to confirm MT qualities between Japanese and other languages. Furthermore, we compare the qualities of direct, pivot, and zero-shot translations.

Europarl (Koehn, 2005), a collection of European Parliament proceedings, is a well-known multilingual parallel corpus. The characteristics of the GCP Corpus are similar to those of the Europarl. However, the GCP Corpus has different applications because it includes Asian languages and pseudo-dialogues that are being developed for use in speech translation systems.

The reminder of this paper is organized as follows. Sections 2. and 3. summarize the GCP and discuss the current status of the GCP Corpus, respectively. In Section 4., we construct a neural machine translation system using the GCP Corpus and evaluate the quality of its translations. In Section 5., we compare direct, pivot, and zero-shot translations. We discuss current problems and future directions in Section 6. and present our conclusions in Section 7.

2. Global Communication Plan

Global Communication Plan was proposed in 2014 by Yoshitaka Shindo, who was Minister of Internal Affairs and Communications of Japan. The mission of the GCP is to eliminate global "language barriers" by targeting the following goals.

1. Realizing global and open communications
2. Enhancing Japanese presence in the world
3. Promoting "O-mo-te-na-shi" (hospitality) at the Tokyo 2020 Olympic and Paralympic Games

[1] We are also developing speech corpora for speech recognition and synthesis tasks; however, we only describe the parallel corpus in this paper.

Category				
Lv. 1	Lv. 2	Lv. 3	Speaker Type	Utterance
Medical Care	Illness or Injury	Response to Urgency	Foreigner	Is there a hospital nearby?
			Japanese	There is an internal medicine hospital after you turn right at that convenience store.
			Japanese	Aren't you feeling well?
			Foreigner	I feel dizzy.
			Japanese	Aren't you good at Japanese?
			Foreigner	I am not good at it.
			Japanese	It may be impossible to communicate in English in the hospital there.
			Japanese	There are staffs who can speak English in the comprehensive hospital in front of the station nearby.

Table 1: Example Pseudo-dialogue for Medical Care Domain

Based on this plan, the Global Communication Developers' Group [2] was formed in collaboration with industry, academia, and the government. This group performs the following tasks: 1) research and development of multilingual speech translation to extend target languages and domains, and 2) public demonstrations of speech translation at hospitals, malls/stores, and tourist spots.
The GCP focuses on the following four domains.

- **Medical care**
  This domain includes dialogues between patients and medical staff (doctors, nurses, etc.) at hospitals.
- **Disaster prevention**
  This domain involves local governments dealing with disasters and providing information to foreigners.
- **Shopping**
  This domain includes dialogues between store clerks and foreign visitors who are shopping.
- **Tourism**
  This domain includes dialogues that provide travel information to visitors, e.g., information about accommodations, transportation, and tourist spots.

The ten target languages are as follows: Japanese (Ja), English (En), Simplified Chinese (Zh), Korean (Ko), Thai (Th), Vietnamese (Vi), Indonesian (Id), Myanmar (My), Spanish (Es), and French (Fr). These languages were selected taking into account the number of visitors to Japan who spoke these languages.

3. GCP Corpus

The GCP Corpus is being developed to help create speech translation systems for the GCP. In this paper, we focus on the translation component rather than the speech component.
The corpus targets four domains defined by the GCP. We assume situations where foreign visitors are speaking with local Japanese people because the goal of the corpus is to realize speech translation in such situations. Therefore, the corpus primarily consists of dialogues. However, these are not real dialogues that have been recorded and transcribed but pseudo-dialogues written by scenario writers imagining possible situations. Pseudo-dialogues are more suitable from the perspective of early development of speech translation systems because actual dialogues contain many ungrammatical utterances that require cleaning during transcription and are difficult for human translators to understand.
Table 1 shows an example pseudo-dialogue in the medical care domain. Here, each dialogue is categorized into three levels, and each utterance contains speaker type information (Japanese or foreigner). We also include isolated utterances, such as greetings and common expressions.
Since the GCP Corpus involves conversations between Japanese people and foreign visitors, we first created pseudo-dialogues in Japanese and then translated them into the other nine target languages. Table 2 shows the size of the corpora for each language at the end of 2017. The Japanese, English, Chinese, and Korean corpora are larger than the other languages because these languages are given first preference.

4. Quality of MTs Based on GCP Corpus

The GCP Corpus is being developed to help realize speech translation systems. In this section, we evaluate MT quality by training a neural MT (NMT) system using the GCP Corpus.

4.1. Experimental Settings

Language Pairs The GCP Corpus is a parallel corpus and includes ten languages; therefore, we can construct up to $10 \times 9 = 90$ different MT systems. Here, we only evaluate the quality of MTs between Japanese and the other languages (Ja $\leftrightarrow$ X; a total of 18 systems) due to resource limitations.

Datasets The corpora (Table 2) were divided into training, development, and test sets. Initially, we set aside some sentences from each corpus (held-out data) and used the remaining sentences as the training set. From the held-out data, we uniformly selected two 2,000 sentence sets as development and test sets.

MT System The training, development, and test sets were segmented into words using in-house word segmenters, and the words were further segmented into subwords using a byte-pair encoder (Sennrich et al., 2016).

[2] http://gcp.nict.go.jp/about/index.html

Language	Abbr.	No. of Sentences (Utterances) Total	Medical Care	Disaster Prevention	Shopping	Tourism	Other
Japanese	Ja	2,029,111 (25.2 chars. / sent.)	420,270	249,495	355,429	527,056	476,861
English	En	2,029,111 (11.2 words / sent.)	420,270	249,495	355,429	527,056	476,861
Chinese	Zh	2,026,608	420,270	249,495	355,429	527,056	474,358
Korean	Ko	2,026,608	420,270	249,495	355,429	527,056	474,358
Thai	Th	1,150,070	145,054	117,636	180,843	232,179	474,358
Vietnamese	Vi	1,150,070	145,054	117,636	180,843	232,179	474,358
Indonesian	Id	1,150,070	145,054	117,636	180,843	232,179	474,358
Myanmar	My	1,150,070	145,054	117,636	180,843	232,179	474,358
Spanish	Es	337,654	145,054	117,636	9,512	18,944	46,508
French	Fr	340,499	145,054	117,636	9,867	19,593	48,349

Table 2: GCP Corpora Sizes as of Summer 2017

The number of sub-word types (corresponding to the vocabulary size) was approximately 16 thousand per language.

We used OpenNMT (Klein et al., 2017) [3] as the neural translation system with the following settings.

- We used a two-layer bi-directional LSTM (long short-term memory) encoder with $500+500$ units. The word embedding was 500 units.
- We used a two-layer LSTM decoder with 1,000 units. The word embedding was 500 units.
- The stochastic gradient descent (SGD) was used for optimization with a learning rate of 1.0 for the first fourteen epochs, followed by annealing of six epochs while decreasing the learning rate by half. The mini-batch size was 64.
- During translation, 10-best translation was performed with a beam width of 10. Furthermore, we applied reranking using the following formula and selected the best translation (Morishita et al., 2017; Oda et al., 2017).

$$ll_{\mathrm{len}}(\boldsymbol{y}|\boldsymbol{x}) = \sum_t \log Pr(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t}) + WP \cdot T, \quad (1)$$

where ll_{len} denotes the log-likelihood that considers translation length, the first term of the right side denotes the log-likelihood, WP denotes a word penalty ($WP \geq 0$), and T denotes the word number of the translation.

Equation 1 corrects a translation length using the word penalty because NMTs typically generate short translations. The word penalty is optimized using a development set to make the translation length and reference length nearly equal. By correcting the translation length, we can compute the BLEU scores regardless of the brevity penalty.

4.2. Translation Quality

Table 3 shows the quality of the MTs as measured by BLEU (Papineni et al., 2002).

[3] http://opennmt.net/

Language	No. of Training Sentences	BLEU Score from Japanese	BLEU Score to Japanese
English	1,954,477	27.20	31.01
Chinese	1,952,475	35.82	42.34
Korean	1,952,475	52.87	58.13
Thai	1,110,232	25.64	27.64
Vietnamese	1,110,232	30.32	30.64
Indonesian	1,110,232	22.16	25.94
Myanmar	1,110,232	23.90	30.82
Spanish	326,433	22.28	24.82
French	329,160	22.05	23.39

Table 3: Quality of MTs between Japanese and other Languages

Languages	Ja → En	En → Ja
General	27.20	31.01
Medical Care	26.53	30.69
Disaster Prevention	27.23	33.01
Shopping	26.09	29.96
Tourism	32.47	33.22
Other	23.89	27.05

Table 4: Translation Quality of Each Domain

First, the BLEU scores are significantly different for each language, ranging from 22.05 to 52.87 for translation from Japanese and from 23.39 to 58.13 for translation to Japanese. However, the score tended to increase with an increasing number of training sentences.

Next, by comparing translations from Japanese with translations to Japanese, it was found that the scores when translating to Japanese were higher than those when translating from Japanese for all language pairs. This phenomenon shows that translating from Japanese was more difficult than translating to Japanese. For example, the subjects of Japanese sentences are sometimes not present, and MT systems translating to other languages must generate such subjects.

In addition, only the BLEU scores from/to Korean were greater than 50. It is known that conventional statistical MTs between Japanese and Korean tend to be high quality because Korean grammar is similar to Japanese grammar (e.g., SVO order). We observed similar results using NMT.

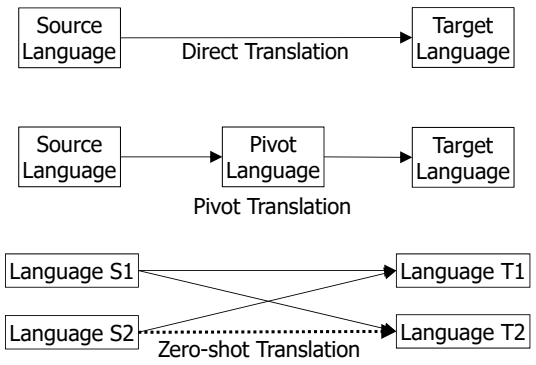


Figure 1: Direct, Pivot, and Zero-shot Translations

Table 4 shows the translation quality of each domain between Japanese and English. 'General' represents the scores of the general test sets described in Section 4.1. The test sets for the domains were extracted by selecting 1,000 sentences from the held-out data described in Section 4.1. The same models used in Table 3 were used for all domains (i.e., no domain adaptation was employed). This table shows that BLEU scores that are comparable to the general test sets were obtained in the target domains of GCP, although there are some variations.

5. Pivot and Zero-shot Translation

We can directly construct translators between all possible language pairs from multilingual parallel corpora (i.e., direct translation). However, if we do not have such parallel corpora, we use pivot translation, which involves translating source sentences into target sentences via a resource-rich language known as a pivot (Utiyama and Isahara, 2007; Cohn and Lapata, 2007).

In NMT, another approach known as the zero-shot translation can be used to construct MT systems without using directly-translated corpora (Johnson et al., 2016). Here, an encoder that recognizes the source language and a decoder that generates the target language are trained using the corpora of indirect language pairs. Figure 1 illustrates the relationship among these three approaches.

Both the pivot and zero-shot translation generally assume that bilingual corpora covering the source language and those that cover the target language are obtained from the different texts. However, comparative analysis is difficult to perform under this setting because the vocabulary differs and we cannot construct a direct translator.

Using multilingual parallel corpora, we can compare these three translation methods. By regarding the quality of the direct translation as the upper bound, we can evaluate improvements in pivot and zero-shot translations by comparing them to the direct translation. Note that the experimental settings can be shared for all three methods; for example, identical vocabularies can be used.

In this section, we compare direct, pivot, and zero-shot translation using four languages, i.e., Japanese, English, Chinese, and Korean.



Figure 2: Training Data for Zero-shot Translation (Ja-En)

5.1. Experimental Settings

Data From approximately two million sentences that were common among the four languages, we created training, development, and test sets using the same method in Section 4.1. The sentences were segmented into fifty thousand sub-word types using byte-pair encoding trained from all training sentences (i.e., joint encoding). Note that the same vocabulary set was used for all experiments discussed in this section.

Direct Translation Models for the 12 language pairs were trained using the system described in Section 4.

Pivot Translation The qualities of six translations among English, Chinese, and Korean were measured using Japanese as the pivot language. In other words, we measured the BLEU scores for (En | Zh | Ko) $\rightarrow$ Ja $\rightarrow$ (En | Zh | Ko) using the Ja $\leftrightarrow$ (En | Zh | Ko) models trained for direct translation tasks. In each case we used the 1-best translations from the source language to the pivot languages.

Zero-shot Translation We constructed (En | Zh | Ko) $\rightarrow$ (En | Zh | Ko) translators using the Ja $\leftrightarrow$ (En | Zh | Ko) corpora similar to the pivot translation.

First, we added a target language tag at the beginning of each source sentence in the Ja $\leftrightarrow$ (En | Zh | Ko) corpora (Figure 2). Then, a unified model was trained using a combined corpus containing all these language pairs. In this experiment, 12 million sentences were used for training.

During testing, the target language tags were added to the source sides of the test sentences, and translation was performed using the unified model. Then, appropriate target sentences were generated based on the tags even though that particular language pair had not been learned.

5.2. Results

Table 5 shows the results of the (a) direct, (b) pivot, and (c) zero-shot translations. With the pivot translations, the BLEU scores for most language pairs were worse than those for direct translation. However, the score for the Ko $\rightarrow$ Zh pair improved; thus, we can conclude that the pivot translation can achieve quality close to that of direct translation.

The zero-shot translations, on the other hand, showed very low scores for the unlearned language pairs. The scores were higher for the learned language pairs (Ja $\leftrightarrow$ X), although they were lower than those for direct translation. This means that the multilingual (unified) model was learned reasonably well; however, further study is required because zero-shot translation has only been researched for a few years.

		Target Language			
		Ja	En	Zh	Ko
Source Language	Ja	–	27.31	35.60	52.81
	En	30.84	–	22.21	26.27
	Zh	42.33	24.14	–	34.85
	Ko	57.92	24.85	30.66	–

(a) Direct Translation

		Target Language			
		Ja	En	Zh	Ko
Source Language	Ja	–	(27.31)	(35.60)	(52.81)
	En	(30.84)	–	20.82	25.49
	Zh	(42.33)	23.85	–	33.87
	Ko	(57.92)	24.55	30.82	–

(b) Pivot Translation
The brackets denote scores of the direct translation.

		Target Language			
		Ja	En	Zh	Ko
Source Language	Ja	–	26.32	34.17	51.73
	En	29.49	–	2.90	5.11
	Zh	40.71	8.87	–	10.00
	Ko	56.66	10.63	6.44	–

(c) Zero-shot Translation (En, Zh, and Ko)

Table 5: Comparison among Direct, Pivot, and Zero-shot Translation

6. Future Directions

First, we intend to further increase the size of the corpus to improve translation quality. Because the amount of the current corpus spreads various range among languages, we especially complement Asian languages, which is a feature of our corpus.

One current problem is context-dependent translation. We consider that there are two types of context-dependent translation.

- One type of context-dependent translation depends on external knowledge, such as domain knowledge. Such translations use special words and expressions depending on the domain. For example, there are two English translations of the following Japanese sentence in the disaster prevention domain.

  Ja1 *Youyaku yure-ga osamari-mashita.*
  finally shaking-SUBJ finish-POLITE
  En1-1 *The shaking finally has calmed down.*
  En1-2 *The earthquake is now stopped.*

  En1-1 is a literal translation of **Ja1**. **En1-2** can only be used when shaking is caused by an earthquake, and it is a particular translation within the disaster prevention domain.

- The other type of context-dependent translation depends on previous utterances. Here, the information of a source sentence is complemented or eliminated in a translation. For example, one translation of **Ja2** is **En2-2** in the GCP Corpus.

  Ja2 *Moyori eki-wa doko-desuka*?
  nearest station-TOP where-POLITE
  En2-1 *Where is the nearest station?*
  En2-2 *What is the station closest to the park?*

  "*The park*" in **En2-2** are extra words of **Ja-2**. However, the speaker also spoke "*I've never heard Tetsugakudo Park.*" in the preceding utterance, so it is correct in this context.

Human translators tend to translate literally between languages of the same family, such as English and French. In contrast, with language pairs for which it is difficult to make literal translations, such as English and Japanese, professional translators elaborately generate context-dependent translations to make the translations natural and the meaning of the dialogue identical.

Most MTs assume that the translation unit is a sentence. Therefore, it is harmful if a training corpus contains bilingual sentences with extra or missing words. The first type of context-dependent translation, which refers to external knowledge, is being solved by domain adaptation techniques. The second type, which depends on previous utterances, has not been solved using a sentence as the translation unit.

In the GCP Corpus, we attempt to apply the following counterapproach to maintain fluency and reduce extra/missing words. [4] First, utterances in a dialogue are shuffled and human translators translate them. This process breaks context in a dialogue; however, translators can refer to external knowledge if they read the entire dialogue. Then, the utterance order is restored, and the translations are checked to maintain the fluency of the dialogue.

7. Conclusion

In this paper, we have introduced the GCP Corpus, which is being developed as part of Global Communication Plan. The GCP Corpus is a multilingual sentence-aligned corpus that is being developed to help realize speech translation. The GCP Corpus has the following characteristics.

- It covers ten languages: Japanese, English, Chinese, Korean, Thai, Vietnamese, Indonesian, Myanmar, Spanish, and French. Notably, it includes Asian languages.
- It supports four target domains, i.e., medical care, disaster prevention, shopping, and tourism.
- It primarily consists of pseudo-dialogues between foreign visitors and local Japanese people.

Here, we have focused on the first of these characteristics and have investigated the corpus by evaluating the quality of different MT systems. In addition, we have compared direct, pivot, and zero-shot translations by taking advantage of the fact that the GCP Corpus is a parallel corpus.

[4] We started this process in the middle of the construction.

As we work toward the 2020 Tokyo Olympic and Paralympic Games, we will increase the size of the GCP Corpus, use it to develop speech translation systems, and distribute these systems widely [5].

8. Acknowledgments

This work was supported by "Promotion of Global Communications Plan — Research and Development and Social Demonstration of Multilingual Speech Translation Technology," a program of the Ministry of Internal Affairs and Communications, Japan.

9. Bibliographical References

Cohn, T. and Lapata, M. (2007). Machine translation by triangulation: Making effective use of multi-parallel corpora. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 728–735, Prague, Czech Republic, June.

Imamura, K. and Sumita, E. (2016). Multi-domain adaptation for statistical machine translation based on feature augmentation. In *Proceedings of the 12th Conference of the Association for Machine Translation in the Americas (AMTA-2016): Volume 1' MT Researchers' Track*, pages 79–92, Austin, Texas, USA, October.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2016). Google's multilingual neural machine translation system: Enabling zero-shot translation. ArXiv e-prints. 1611.04558.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2017). OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, July.

Morishita, M., Suzuki, J., and Nagata, M. (2017). NTT neural machine translation systems at WAT 2017. In *Proc. of WAT2017*, pages 89–94.

Oda, Y., Sudoh, K., Nakamura, S., Utiyama, M., and Sumita, E. (2017). A simple and strong baseline: NAIST-NICT neural machine translation system for WAT2017 English-Japanese translation task. In *Proc. of WAT2017*, pages 135–139.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, Pennsylvania, USA.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany.

Utiyama, M. and Isahara, H. (2007). A comparison of pivot methods for phrase-based statistical machine translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 484–491, Rochester, New York, April.

10. Language Resource References

Higashinaka, R., Funakoshi, K., Kobayashi, Y., and Inaba, M. (2016). The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X*, pages 79–86, Phuket, Thailand.

[5] We are providing the VoiceTra system.
`http://voicetra.nict.go.jp/`

A Large Parallel Corpus of Full-Text Scientific Articles

Felipe Soares, Viviane Pereira Moreira, Karin Becker
Institute of Informatics, Universidade Federal do Rio Grande do Sul
Porto Alegre - RS, Brazil
{felipe.soares, viviane, karin.becker}@inf.ufrgs.br

Abstract

The Scielo database is an important source of scientific information in Latin America, containing articles from several research domains. A striking characteristic of Scielo is that many of its full-text contents are presented in more than one language, thus being a potential source of parallel corpora. In this article, we present the development of a parallel corpus from Scielo in three languages: English, Portuguese, and Spanish. Sentences were automatically aligned using the Hunalign algorithm for all language pairs, and for a subset of trilingual articles also. We demonstrate the capabilities of our corpus by training a Statistical Machine Translation system (Moses) for each language pair, which outperformed related works on scientific articles. Sentence alignment was also manually evaluated, presenting an average of 98.8% correctly aligned sentences across all languages. Our parallel corpus is freely available in the TMX format, with complementary information regarding article metadata.

Keywords: parallel corpus, scientific articles, Scielo

1. Introduction

Cross-language corpora is one of the basis of Statistical Machine Translation (SMT) systems. The acquisition of quality corpora is not a trivial task, as it may demand considerable use of expert human curating, especially for parallel corpora. In that sense, the automated building of parallel corpora from open resources is of great interest in Natural Language Processing (NLP). Europarl (Koehn, 2005), is one of the largest parallel corpora availabe, including up to 21 European languages. Similarly, the United Nations (UN) parallel corpus (Ziemski et al., 2016) makes use of UN's official documents and records, providing aligned sentenced in six languages. Other parallel corpora initiatives have been reported in distinct domains, such as patents (Utiyama and Isahara, 2007), movie subtitles (Zhang et al., 2014), and books (Skadins et al., 2014).

Parallel corpora based on scientific articles can be a valuable language resource. Several text mining tasks may benefit from the availability of parallel corpora of scientific articles. The most straightforward example is cross language plagiarism detection, when an original text is translated in another language and presented as novel. Other possible applications are related to article indexing or classification, as well as the development and extension of Named Entity Recognition (NER) tools for multiple languages. The latter is of particular interest to the increasingly active biomedical sciences field, which has already standardized vocabularies and ontologies, thus favoring NER initiatives.

The construction of parallel corpora of scientific texts has been addressed by different authors. Wu et al. (2011) constructed a parallel corpus of biomedical article titles from PUBMED in six languages. Based on this corpus, the authors built SMT systems and achieved higher BLEU scores than Google Translator. With a different intent, Kors et al. (2015) produced a gold-standard annotated parallel corpus for biomedical concept recognition in five languages. They used Medline abstracts, drug labels, and patent claims as sources. Recently, Neves et al. (2016) used the Scielo scientific database to produce a parallel corpus of biomedical abstracts in three language pairs: Portuguese-English, Spanish-English, and French-English.

The Scielo database is a Latin American and Caribbean initiative developed to meet the needs of developing countries regarding scientific communications, increasing the visibility and access to scientific literature (Packer, 2000). Another interesting aspect of Scielo is that several journals publish full-text of scientific articles in more than one language, a feature commonly limited to the abstracts. Therefore, the Scielo database can be a valuable source for parallel corpora for various scientific domains.

In this work, we developed a parallel corpus of full-text scientific articles collected from the Scielo database in English (EN), Portuguese (PT) and, Spanish (ES). The corpus is sentence aligned for all language pairs. We also made available trilingual alignments for a subset of sentences. The main differences with regard to a previous initiative of Neves et al. (2016) are: (i) our corpus contains full-text articles, providing a larger resource; (ii) we collected data from various domains, not just biomedical; (iii) whenever possible, we presented the articles in a structured way using sections and paragraphs, favoring other NLP tasks such as summarization; and (iv) we included metadata, such as journal name and subject category, which can be used for text classification.

2. License and Availability

Most articles in the Scielo database are licensed under the Creative Commons copyright, with different types of licenses. In order to be able to distribute the contents of the gathered articles, we filtered only those licensed under terms that allow derivatives, since ND (No Derivatives) licenses require the content to be distributed without any modification. As we removed some parts from the articles (e.g. images, tables, references), we would be infringing such copyright rules. All articles distributed in our dataset contain the corresponding license, authorship, and unique identifiers of original sources, as detailed in Section 4.1.

3. Material and Methods

In this section, we detail the information gathered from Scielo, the filtering process, as well as our method for article parsing and alignment. An overview of the workflow employed in this article is shown in Figure 1

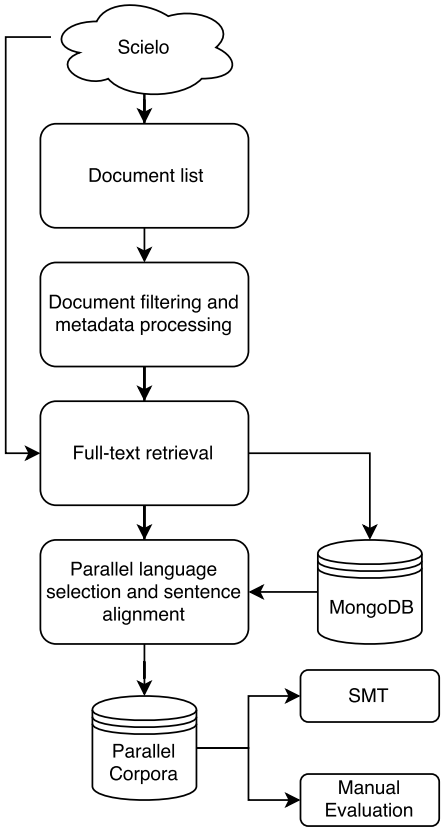


Figure 1: Steps employed in the development of the parallel corpora.

3.1. Document retrieval

Scielo's website[1] provides unified access to a series of regional databases (such as from Argentina, Brazil, South Africa), offering simple and advanced search capabilities. We iteratively queried the database to retrieve all lists of results, which were then parsed and all relevant contents stored, such as URLs for all available languages of each article, authorship, licensing, title, and abstract. We adopted the MongoDB database system, as it is document-oriented, and allows for the easy querying and storage of this type of data.

We queried the results in MongoDB to filter only the articles meeting the following constraints: a) articles with full-text available in at least two of three languages of interest (i.e. English, Portuguese, and Spanish); and b) type of licensing is non ND terms. The full-text of all articles meeting these two criteria were downloaded from the Scielo database in HTML format.

[1] `http://www.scielo.org`

3.2. Document parsing

The HTML contents of all articles were parsed using an in-house Python script tailored to the Scielo format. First, all non-textual elements, such as images, tables, references, citations, and footnotes were removed. Our algorithm was designed to preserve the hierarchical and paragraph structure of the article across the different languages in order to produce results aligned at paragraph and section levels. This could help achieving good sentence level alignment.

The main challenges in parsing Scielo HTML contents are heterogeneity issues concerning HTML structure and formatting over different years. More recent articles are well-formated and contain specific tags for paragraphs, sections, subsections, and titles. We concentrated efforts in developing rules to tackle all ill-formated HTML issues identified, so as to cover as much content as possible, but to reduce the risk of misalignment, we discarded all documents that presented very different structures across the languages.

Each parsed full-text translation was stored in MongoDB aiming at preserving the structure of the articles. When our parsing algorithm failed at identifying the document structure, its content was stored as a unstructured list of paragraphs, as we assume that if two translations of the same article present the same number of parsed paragraphs, it is likely they can be simply aligned according to their order.

3.3. Sentence alignment

Once all articles were parsed, we employed a pre-processing step to ensure a better alignment. We deleted all parentheses from the texts (mainly used for citations), as well as newline/carriage return characters (i.e `\n` and `\r`), as they would interfere with the sentence alignment tool.

For sentence alignment, we used the LF aligner tool[2], a wrapper around the Hunalign algorithm (Varga et al., 2007), which provides an easy to use and complete solution for sentence alignment, including pre-loaded dictionaries for several languages.

Hunalign uses Gale-Church sentence-length information to first automatically build a dictionary based on this alignment. Once the dictionary is built, the algorithm realigns the input text in a second iteration, this time combining sentence-length information with the dictionary. When a dictionary is supplied to the algorithm, the first step is skipped. A drawback of Hunalign is that it is not designed to handle large corpora (above 10 thousand sentences), causing large memory consumption. In these cases, the algorithm cuts the large corpus in smaller manageable chunks, which may affect dictionary building.

For articles with the same structure across the languages, pairs of parallel paragraphs were input to the sentence aligner at a time, aiming at reducing the risk of misalignment. For the other cases, all paragraphs were passed to the aligner together. Aligned sentences were stored as text files for post-processing.

After sentence alignment, the following post-processing steps were performed: (i) removal of all non-aligned sentences; (ii) removal of all sentences with fewer than three

[2] `https://sourceforge.net/projects/aligner/`

characters, since they are likely to be noise from ill-formatted HTML; (iii) removal of all sentences written in the same language using a language detector[3]. This last step was performed since abstracts in different languages could be present in a full-text HTML, which could produce same-language alignments.

3.4. Machine translation evaluation

To evaluate the usefulness of our corpus for SMT purposes, we used it to train an automatic translator with Moses (Koehn et al., 2007). The produced translations were evaluated according to the BLEU score(Papineni et al., 2002), using the evaluation tool `multi-bleu` in Moses.

3.5. Manual evaluation

Although the Hunalign algorithm usually presents a good alignment between sentences, we also conducted a manual validation to evaluate the quality of the aligned sentences. We randomly selected 300 pairs of sentences, 100 for each language pair, and 100 trilingual sentences. If the pair was correctly aligned, we marked it as "correct", otherwise, as "no alignment".

4. Results and Discussion

In this section, we present statistics about the corpus and a quality evaluation in terms of SMT and sentence alignment.

4.1. Corpus statistics

Table 1 shows the overall corpus statistics for all language pairs and for the set of trilingual aligned documents. One may notice that EN-PT documents are predominant over other language pairs. This may be explained by the fact that almost all Brazilian journals are published through Scielo, thus favoring Portuguese-English translations.

The datasets are available[4] in the TMX format (Rawat et al., 2016), since it is the standard format for translation memories. Besides the aligned sentences, we included the following metadata for each document: aligned title, authors, copyright license, DOI (if available), journal name, Scielo's unique identifier, and subject area. This information was included either to fully comply with Creative Commons requisites, or to provide additional information for other possible applications, such as text classification or clustering.

An example of trilingual sentence is shown below:

> English: *Among its objectives, it aims to defend the interests of society and Nursing in the context of Public Policies and the Unified Health System with emphasis on Mental Health.*
>
> Spanish: *Entre sus objetivos está defender los intereses de la sociedad y de la Enfermería en el contexto de las Políticas Públicas y del Sistema Único de Salud con énfasis en el área de la Salud Mental.*
>
> Portuguse: *Entre seus objetivos, visa defender os interesses da sociedade e da Enfermagem no contexto das Políticas Públicas e do Sistema Único de Saúde com ênfase na área de Saúde Mental.*

Table 1: Corpus statistics for all language pairs and the trilingual set. "Docs" refers to the number of documents, "Sents" to the number of aligned sentences, and "Tokens" is the number of tokens in each language.

Languages	Docs	Sents	Tokens
EN-ES	2,029	177,781	5.2M
			5.7M
PT-ES	76	4,987	140,434
			151,148
EN-PT	29,609	2.9M	76.0M
			77.3M
EN-PT-ES	3,142	255,914	7.0M
			7.8M
			7.2M

Table 2: BLEU scores for translation using Moses. Previous related work by Neves et al.(2016) is also presented for comparison in the right-hand column.

Language Pairs		BLEU Current Work	BLEU (Neves et al.2016)
EN-ES	EN→ES	36.88	32.75
	ES→EN	37.93	30.53
PT-ES	PT→ES	62.63	-
	ES→PT	62.96	-
EN-PT	EN→PT	48.51	33.37
	PT→EN	49.24	31.78

4.2. SMT experiments

Prior to the SMT experiments, all sentences were randomly split in three disjoint datasets for each language pair: training, tuning and test. Approximately 85% of the aligned sentences were kept for training, 5% for tuning and 10% for test. The translation models were built following Moses' baseline system steps[5].

Table 2 presents the BLEU scores for each language pair for the test set. We included the best results from related work by Neves et al. (2016) for the sake of comparison. We highlight that their study was focused on titles and abstracts from biomedical articles, while our corpus is focused on full-text content of scientific articles in general.

Our EN-ES models presented considerably higher BLEU scores, despite dealing with several domains. Regarding EN→ES translation, our results are approximately 4.13 percentage points (pp) higher, while about 7.4 pp better for ES→EN. Considering EN-PT, our results are more expressive. BLEU score for EN→PT is 15.48 pp higher, and 17.46 pp for PT→EN.

Another source of comparison is the EuroMatrix project[6], which is an European funded initiative to promote ma-

[3] `https://github.com/Mimino666/langdetect`

[4] `https://figshare.com/s/091fcaf8ad66a3304e90`

[5] `http://www.statmt.org/moses/?n=moses.baseline`

[6] `http://matrix.statmt.org/matrix`

chine translation research for all pairs of the European Union (EU) languages. When comparing our results to the reported benchmarking BLEU scores available on the project's website, our results are similar to the reported BLEU scores for the Europarl corpus (scores between 29.4 and 39.4) and overall lower than the ones for the JRC-Acquis corpus (between 55.1 and 60.7). The main differences among these three corpora are domain related, since JRC-Acquis is comprised of EU laws applicable in its state members, Europarl is derived from the EU Parliament proceedings, and our corpus is from scientific articles. Thus, a variation in achieved BLEU scores is expected. As stated by Koehn et al. (2009), JRC-Acquis corpus is of considerable size within its very specific and well-defined domain of legal text, therefore presenting good translation performance. On the other hand, Europarl corpus is a transcription of speeches, thus inducing a greater linguistic variability.

4.3. Sentence alignment quality

We manually validated the alignment quality for 400 sentences randomly sampled from the parsed corpus. Figure 2 depicts the rate of correct alignments for each subset of parallel languages. All language combinations presented at least 98% of correct alignments, with the language pair ES - PT achieving 100%. Different factors may have contributed to this high alignment quality. The use of Hunalign (Varga et al., 2007) with a dictionary is perhaps the most probable reason, as it combines a dictionary with sentence-length information to boost alignments. The input of articles segmented by parallel paragraphs also contributed to quality enhancement, since this can reduce the probability of misalignment.

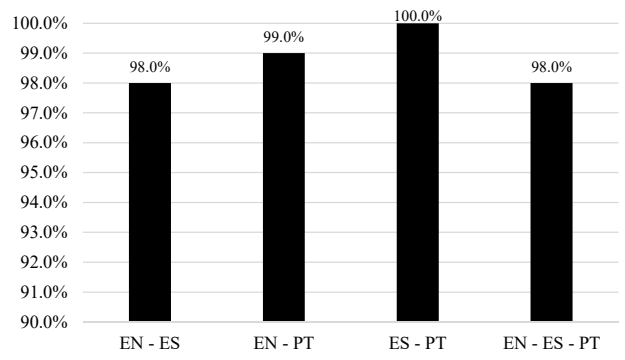


Figure 2: Alignment accuracy for the four language subsets.

During alignment validation, some translation incoherences were identified, specially regarding nomenclatures, which may have affected the SMT performance. An example of incorrect scientific nomenclature translation is:

> Portuguese: *[...]movimentos coreoatetosicos nos membros ipsilaterais ao lado comprometido*

which was translated to:

> English: *[...] and ipsilateral coreoatetosis*

while the correct translation would be using "choreoathetosis":

> English: *[...] and ipsilateral choreoathetosis*.

Another nomenclature translation problem, but related to context, is:

> Spanish: *[...]estudiantes en el internado de la Escuela Superior de Medicina*

which was translated to:

> English: *[...]boarding school students at the School of Medicine*

while a better translation would be:

> English: *[...]medical intern students at the School of Medicine*

since boarding schools are mainly associated to primary and secondary education.

5. Conclusion and Future Work

We developed a parallel corpus of scientific articles in three languages: English, Portuguese and Spanish. Additionally to the language pairs, we provided a subset of trilingual aligned sentences. Our corpus is based on full-text contents from the Scielo database, which is available under open-access licenses, thus favoring distribution.

We evaluated our corpus using an SMT experiment with Moses and by manual evaluation of sentence alignment. Our translation experiment presented superior performance regarding BLEU score than a previous related work on Scielo database. We highlight the high translation scores achieved for PT-EN language pairs, boosted by the large number of sentences (almost 3M). Hunalign also presented remarkable alignment quality, with over 98% sentences correctly aligned. Other important features of our corpus are the availability of trilingual sentences, and the additional subset of articles aligned according their hierarchical structures, which can be useful for automatic building of structured abstracts.

Regarding future work, we foresee the use of this corpus in text mining applications, such as classification and clustering. In addition, the corpus could be used in cross-language plagiarism detection, as we provide at least two versions of the same article in multiple languages. New Neural Machine Translation (NMT) systems could be trained to provide a comparison against the tested SMT system. An interesting approach would be use an approach similar to Google's multilingual NMT (Johnson et al., 2017) to perform zero-shot translation based on an additional parallel corpus.

6. Bibliographical References

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al. (2017). Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Koehn, P., Birch, A., and Steinberger, R., (2009). *462 Machine Translation Systems for Europe*, pages 65–72.

Association for Machine Translation in the Americas, AMTA.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Kors, J. A., Clematide, S., Akhondi, S. A., Van Mulligen, E. M., and Rebholz-Schuhmann, D. (2015). A multilingual gold-standard corpus for biomedical concept recognition: the mantra gsc. *Journal of the American Medical Informatics Association*, 22(5):948–956.

Neves, M., Yepes, A. J., and Névéol, A. (2016). The scielo corpus: a parallel corpus of scientific publications for biomedicine. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Packer, A. L. (2000). Scielo-a model for cooperative electronic publishing in developing countries. *D-Lib Magazine*, 6(10).

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Rawat, S., Chandak, M. B., and Chauhan, N., (2016). *An Approach for Efficient Machine Translation Using Translation Memory*, pages 285–291. Springer Singapore, Singapore.

Skadins, R., Tiedemann, J., Rozis, R., and Deksne, D. (2014). Billions of parallel words for free: Building and using the eu bookshop corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1850–1855.

Utiyama, M. and Isahara, H. (2007). A japanese-english patent parallel corpus. *MT summit XI*, pages 475–482.

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., and Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam studies in the theory and history of Linguistic Science series 4*, 292:247.

Wu, C., Xia, F., Deleger, L., and Solti, I. (2011). Statistical machine translation for biomedical text: are we there yet? In *AMIA Annual Symposium Proceedings*, volume 2011, page 1290. American Medical Informatics Association.

Zhang, S., Ling, W., and Dyer, C. (2014). Dual subtitles as parallel corpora.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The united nations parallel corpus v1.0. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

7. Language Resource References

Koehn, Philipp and Hoang, Hieu and Birch, Alexandra and Callison-Burch, Chris and Federico, Marcello and Bertoldi, Nicola and Cowan, Brooke and Shen, Wade and Moran, Christine and Zens, Richard and others. (2007). *Moses: Open source toolkit for statistical machine translation*.

NegPar: a parallel corpus annotated for negation

Qianchu Liu[1] **Federico Fancellu**[2] **Bonnie Webber**[2]
[1] LTL,University of Cambridge
[2] ILCC, School of Informatics, University of Edinburgh
ql261@cam.ac.uk f.fancellu@sms.ed.ac.uk bonnie@inf.ed.ac.uk

Abstract

Although the existence of English corpora annotated for negation has allowed for extensive work on monolingual negation detection, little is understood on how negation-related phenomena translate across languages. The current study fills this gap by presenting *NegPar*, the first English-Chinese parallel corpus annotated for negation in the narrative domain (a collection of stories from Conan Doyle's Sherlock Holmes). While we followed the annotation guidelines in the CONANDOYLE-NEG corpus (Morante and Daelemans, 2012), we reannotated certain scope-related phenomena to ensure more consistent and interpretable semantic representation.
To both ease the annotation process and analyze how similar negation is signaled in the two languages, we experimented with first projecting the annotations from English and then manually correcting the projection output in Chinese. Results show that projecting negation via word-alignment offers limited help to the annotation process, as negation can be rendered in different ways across languages.

Keywords: negation, annotation projection, English, Chinese, parallel corpora, cross-lingual semantics

1. Introduction

Considering negation when extracting information from text is crucial for a range of natural language processing (NLP) tasks, especially when one is interested in whether an action or a state described in a statement is factual or counter-factual.
To date, most work has focused on automatically identifying which tokens in a sentence constitute the three key elements of negation: *cue, event and scope*. As shown in (1), 'not' is marked as the presence of negation, the cue; 'eat' is a negated event; and ['I', 'do', 'eat', 'pizza'] are part of the scope.[1] If not otherwise stated, here and in the remainder of the paper, the cue is marked in **bold** with the scope underlined and the event marked in a box.

(1) I do **not** eat pizza

Corpora annotated for negation have been developed for both English and Chinese and for a variety of domains (§ 2.). However, these are monolingual corpora, and there is little empirical work on how negation is represented across languages.
The current study aims at addressing this shortcoming by presenting the first *parallel* corpus annotated for negation. We created it by leveraging the CONANDOYLE-NEG corpus (Morante and Daelemans, 2012), a collection of four Sherlock Holmes stories. Before extending the English annotations onto the sentence-aligned Chinese translation, we edited the English annotations to better capture phenomena related to negation scope. (§ 3.2.).
To ease the annotation task and to investigate differences in the way negation is represented in the two languages, we experimented with first automatically projecting the annotations and then manually correcting them (§ 3.3. and § 4.).
We experimented with projecting onto both Chinese characters and words via character or word alignment extracted from a large parallel corpus.
In general, the performance of automatic projection yields relatively low results with a large number of false negatives, hence lower recall. In order to ease the annotation task, more effort is required to find those instances where projection failed to predict mostly because negation is present in the Chinese translation but not in English, than to filter over-predicted instances. .
We hope that the resource and insights provided in this study will foster work in a variety of NLP tasks such as machine translation, information extraction, etc.

[1]The negated event is usually considered as part of the scope of negation

2. Related Work

2.1. Corpora annotated for negation

The importance of detecting negation is testified by the different domains corpora were annotated for. BIOSCOPE (Vincze et al., 2008), a collection of medical papers, abstract and clinical reports, stresses the importance of recognizing negation for information extraction from medical records; the SFU PRODUCT REVIEW CORPUS (Konstantinova et al., 2012) annotates negation on top of product reviews, acknowledging its importance for sentiment analysis tasks. Finally, CONANDOYLE-NEG (Morante and Daelemans, 2012) annotates negation in narrative texts – a collection of four stories from Conan Doyle's Sherlock Holmes. There have also been some attempts in developing corpora annotated for negation in other languages as demonstrated by CNESP (Chinese Negation and Speculation corpus) (Zou et al., 2015), which closely follows the annotation style of the BIOSCOPE corpus.
However, tailoring the annotation style to a specific domain leads these corpora to differ in what was annotated and how. For instance, the SFU and BIOSCOPE corpora consider negation scope in purely syntactic terms (that we infer

is 'the maximum constituent c-commanded by the negation cue'), including only the tokens to the right of the cue and excluding the subject of the clause except in passive constructions. (2) exemplifies this.

(2) a. BIOSCOPE: It helps activation, **not** inhibition of the cell.

b. SFU: This book **wasn't** published before the year 2000

The CNESP corpus follows a similar notion of scope but also includes the subject, unless this creates a discontinuity in the scope; the subject 'it' in (2.a) would therefore *not* be annotated because it would break the scope into two discontinuous spans.

CONANDOYLE-NEG considers a more semantic notion of scope which can be a discontinuous span of text and is therefore able to capture long range dependencies that other corpora do not take into account. Unlike other corpora, CONANDOYLE-NEG also annotates affixal negation (e.g. '**im**patient') as well as the negated event, which allows for better coverage of the negated elements in a sentence.

2.2. Annotation projection

To our knowledge, there has not been any previous work on projecting negation across languages. However previous studies have experimented with projecting semantic annotations via word-alignment information extracted from large parallel corpora.Hwa et al. (2002) used word alignment to project parses from English to Chinese and later improved the performance by implementing a set of linguistically-informed post-processing rules. Padó and Lapata (2009) used word alignment information in their constituent-based projection algorithm to transfer semantic role labels from English to German.

3. The Annotation Task

3.1. Creating the corpus

We built our annotated parallel corpus by aligning the four stories in CONANDOYLE-NEG ('The Hound of the Baskervilles', 'The Adventure of Wisteria Lodge', 'The Adventure of the Cardboard Box' and 'The Adventure of the Red Circle') to their Chinese translation by Mengyuan Lin.[2]

We annotate negation as a tuple of three elements: the **cue**, which is the word (e.g. 'not'), multiword unit (e.g. 'neither...nor') or morpheme (e.g. 'im'-patient) that are inherently expressing negation; the scope, which is the sentence span affected by negation; and the negated event, which is the part of the scope that is directly or most prominently negated. The current annotations closely follow the same CONLL format of CONANDOYLE-NEG, marking each negation instance in a sentence as a set of three columns, for cue, event and scope respectively.

The annotation and the guidelines for Chinese were created by one of the current authors, who is a native Mandarin speaker with linguistics background. Issues arising during the annotation process were discussed with the other two authors. All authors collaborated equally in reannotating the English side.

The annotations were carried out on the basis of the projection from CONANDOLYE-NEG. When a negation instance in English is translated as such into Chinese, we correct the projection if the projection is wrong. If a negation instance in English is not translated as negation in Chinese, the extra negation projection is deleted from the Chinese corpus. Finally, we add those annotations for instances of negation in Chinese that are not present in the corresponding English text. The annotation process took approximately one month to complete.

The corpus with the full annotation guidelines can be found at `https://github.com/qianchu/NegPar`.

A quantitative comparison between the English and the Chinese sides of our corpus is presented in Table 1. Overall, there are more negation instances in the Chinese translation than in English as the percentage of negated sentences, the number of cues, scopes and events in Chinese are consistently higher than in English.

	English	Chinese
# sents	5520	5005
# neg. sents	1227(22.22%)	1442(28.81%)
# cues	1421	1782
# events	911	1168
# scopes	1304	1762

Table 1: Comparison between the English and the Chinese text of NEGPAR in terms of total number of sentences (# sents.), negated sentences (# neg.sents.), number of cues (#cues), events (#events) and negated scope spans (#scopes).

3.2. Reannotating the English side

Before projecting the CONANDOLYE-NEG onto Chinese, we reread the original guidelines of Morante and Daelemans (2012) to verify how phenomena related to negation scope are handled.

We divide this discussion in two parts, the first addressing the phenomena we think need improvement and the second addressing those that were not considered by the original guidelines.

3.2.1. Alternative annotation choices

Morphological negation. Some cases of morphological negation (as in '**un**happy', '**im**patient', etc.) do not have scope in the traditional semantic sense of the word. The statements 'I am happy' and 'I am unhappy' can in fact be false at the same time ('I am neither happy nor unhappy' is a valid statement), which is why they are contraries rather than real contradictions. Although keeping them annotated, we do not consider them any further.

However, as a matter of consistency, we reconsider the status of negation affixes in adverbs. Whereas the original English guidelines state that 'If the negated affix is attached to an adverb that modifies a verb, the negation scopes over the entire clause'(Morante et al., 2011), we found that in many

[2]The Chinese translation can be found at http://www.kanunu8.com/book3/8105/.

such cases, only the adverb is in the scope. Such inconsistencies are shown in (3).

(3) a. [...] tossing rest**lessly** from side to side
b. [...] glaring help**lessly** at the frightful thing which was hunting him down

In a case like (3.b) we felt that the wide scope reading does not lead to a correct interpretation because the event actually takes place and it is just the manner it takes place that is negated. For this reason, in such cases, we just **annotate the adverb as being in the scope of negation**; therefore, (3.b) will be reannotated in the same way as (3.a).

Except/save/no...but. By stating that 'often exception items function as neutralizers of the polarity of the statement expressed in the main sentence where they occur', Morante et al. (2011) annotated these as part of the scope, as well as the material they introduce. This is exemplified in (4):

(4) [...] Marx knew **nothing** of his customer save that he was a good payer

This annotation style does not reflect the fact that 'save' excludes from the set of things negated in the matrix clause, which should be interpreted as positive. (4) can be in fact paraphrased as '*It is not the case that* (Marx knew nothing about his customer). He knew that he was a good payer'.
This use of 'save' contrasts from its use as a negation cue as shown in (5)

(5) Mr. Sherlock Holmes, who was usually very late in the mornings, **save** upon those not infrequent occasions when he was up all night, was seated at the breakfast table.

Here, 'save' is used to neutralize positive polarity to highlight the set of instances where an event did not take place; (5) implies in fact that 'he was usually very late but he wasn't late on those not infrequent occasions when he was up all night'.
For this reason, we distinguish two types of exceptions: **The 'exception to nothing', which is positive, as shown in (4), and is excluded from the scope of negation; and the 'exception to usually', with a negative meaning as shown in (5), where we include only the 'except' phrase in the scope of negation, as the original guidelines already do**.

3.2.2. Additional phenomena

Neg raising. Neg raising, i.e. the phenomenon that a negation in the matrix clause of a sentence is interpreted as negating the complement clause, is not covered by the annotation guidelines. Neg raising is encountered with verbs expressing the speaker's opinion, such as 'think', 'believe', 'want', 'seem', etc.. In cases like (6), the original annotations consider the entire sentence under the scope of negation; however, it is not the thinking that should be negated but the object of the thought.

(6) I do **not** think it is likely = I think that it is not likely.

In cases of neg raising we annotate as part of the scope the subordinate only. (6) is therefore reannotated as(7)

(7) I do **not** think it is likely

Quantifiers. The interaction between quantification and negation scope at a string level is not considered at all in the original CONANDOYLE-NEG guidelines.
Cases where 'not' directly precedes lexical items like 'all' and 'every' are correctly annotated, as in the following.

(8) Money is **not** everything.
(= *It is not the case that* money is everything).

However, let us consider the following example annotated according to the original guidelines.

(9) The fellow might have had other reasons for thinking that all was **not** well

The original guidelines paraphrased the construction under the scope as 'It was not the case that the fellow was thinking that all was well', which is why 'thinking' as well is in the scope. Besides the problem with neg raising, from a logical perspective, the universal quantifier should scope over negation and not vice versa, given that $\forall thing(x) \rightarrow \neg\exists e.well(e) \wedge Topic(e, x)$
In cases like (10), we therefore **exclude the lexical item representing universal quantification** from the scope to yield the following annotation.

(10) The fellow might have had other reasons for thinking all was **not** well

Modals. The interaction between the scope of negation and modality is another phenomenon the guidelines do not mention. Some cases, as the one shown in (11), are correctly handled: negation correctly scopes over the modal.

(11) You need **not** to fear to speak the truth. = *It is not the case that* you need to fear to speak the truth.

We however found two cases of deontic modality where the annotations fail to capture this interaction as shown below.

(12) You certainly must **not** go alone $\neq$ *It is not the case that* you certainly must go alone.

Having negation take scope over 'must' leads to the incorrect interpretation that the person could go alone.
In cases like (12), we adopt a strategy similar to the one used for quantifiers and **exclude the lexical item representing modality from the scope**.

3.3. Annotating the Chinese side

We include here a brief summary of the annotation guidelines in Chinese, where we report annotation examples subdivided into the three components we considered: cue, scope and event. The full guidelines can be found at the same link presented in §3.1..

3.3.1. Cue

We annotated a total of 45 negation cue *types* in Chinese including adverbs, auxiliary verbs and prefixes. Amongst these, we found 10 *core* negation cues[3]; these negation cues can stand independently to function as adverbs (similarly to the English 'not') but they can also be compounded with either adjectives or adverbs to form multiword cues eg.并不, meaning 'not' with a contrast emotion.

The most common cues in Chinese are 不，没有，没and 无as shown in Figure 1. 不is an adverb cue equivalent to 'not', and it is the most widely used cue in Chinese. 没(有) is a negated auxiliary verb that indicates non-completion, roughly equivalent to English 'did not/have not'. 无is the classical form of 没有. In modern Chinese, 无is also used as a negation prefix that translates the English suffix '-less'. e.g. 无线('no wire=wireless').

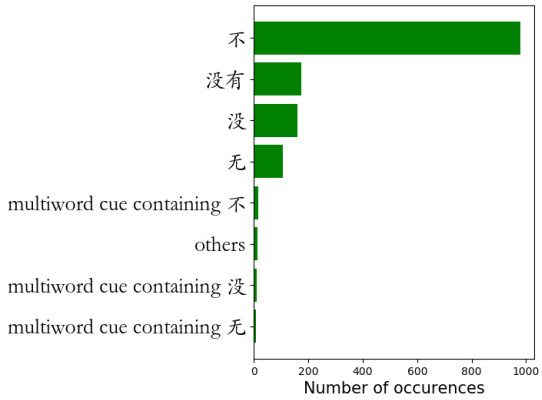


Figure 1: The distribution of Chinese cues in NegPar

Implicit negated verbs. English verbs such as 'refuse' and 'fail' implicitly express negation and are treated as negation cues in the English guidelines. However, since they are not annotated consistently in the English side, we decide *not* to mark these verbs as cues in Chinese.

Infix cue in verb-complement constructions. In Chinese, the negation cue 不 can appear as an infix in verb-complement construction. The complement usually indicates the result or the direction of an action expressed by the verb, as well as expresses a potential form. According to Li and Thompson (1989), one can interpret infixal negation in Chinese as introducing a result that is 'unachievable', roughly equivalent to English 'cannot'. (13) exemplifies this construction alongside its annotation, where only the infix 不 is considered the cue.

(13) 他 说 得 清楚
He speaks can clear
'He can speak clearly.'

他 说 **不** 清楚
He speaks cannot clear
'He can**not** speak clearly.'

Non-functional negation cue. We do *not* annotate as cues any instances of non-functional negation, i.e. expressions that include markers of negation but have positive meaning.

Certain fixed expressions belong to this category; as shown in (14), the expression 'can't help' and the Chinese counterpart '不得不/不能不' , despite including a negation marker, have a positive meaning (i.e. the action they introduce has taken place or will take place).

(14) 我 不得不 讨厌 他
I not-should-not hate he
'I couldn't help hating him'

Chinese displays a wide array of double negative constructions that have positive meaning. Some of these are *chengyu*, idioms of four characters, where the first and the third characters are sometimes negation markers; For example, the idiom 无往不利, literally meaning 'there are no places where victory is not achieved', has a positive meaning that is translated into 'always successful'.

Similar to English, another issue arising from identifying non-functional negation in Chinese is that negation affixes sometimes do not introduce negation. For instance, in the word 'disgrace', the affix 'dis-' is *not* considered a cue because the meaning of the whole word is not 'no grace', whereas 'impatient' is opposed to 'patient'.

This problem is mostly related and can be solved through *semantic transparency*; that is, if the meaning of the whole cannot be analyzed from the meaning of the parts, we do not annotate part of the word as a negation cue. In the case of 无聊'boring', literally meaning'no chatting', we do not annotate the negation marker 无 as a cue.

Another criterion that we used is *obsolescence*: if part of a word that is modified by a negation affix is now obsolete, we do not annotate the affix as a cue. This is the case of 不然'otherwise', where the second character possessed the meaning of 'like this' in classical Chinese, but not in modern Chinese.

Finally, we exclude negation cues used in rhetorical and yes-no questions which often take the form of 'modal + cue + modal'. Some of these, as in (15), are roughly equivalent to the English 'shall we...?'

(15) 咱们 要 不 要 向 后 退 ?
we want not want towards back retreat ?
'Shall we move further back?'

Discontinuous cues. The Chinese equivalent of 'neither...nor', '既不...也不' , is also a type of discontinuous cues as shown in (16). It is worth mentioning however that omission is a feature of this construction which can be reduced further to '不...不', therefore preserving only the core cues.

(16) 对 他 **(既)不** 应该 可怜 , **(也)不** 应该 原谅
towards him not should pity , not should excuse
'[...] for whom there was neither pity nor excuse'

[3]To see a list of the core negation cues with examples, please go to the link provided in §3.1.

3.3.2. Scope

Sentential negation. If negation is sentential, i.e. the predicate of a simple clause is negated, we annotate the entire clause under the scope of negation. In the case of two or more coordinated clauses where only one is negated, we annotate as inside the scope only the negated clause. If there is any material missing from the negated clause but retrievable from other parts of the sentence, this retrievable part is annotated as well. (17) exemplifies the annotation of coordination.

(17) 我把 他 弃 而 不 顾 了
I BA him abandon and not care ASP
'I abandoned and did not care about him'

Subordination. If the negated event is only in the subordinate clause, we only consider the subordinate clause for scope annotation but not the matrix clause. If the negated event is only in the matrix clause, subordinates are usually excluded from the scope of negation.
However, Chinese allows for it-cleft constructions like the one in (18), where only the subordinate clause, which appears before the event of the main clause, is in the scope of negation.

(18) 您 不 是 因为 知道 了 这 一点 才 感到 高兴
you not BE because know ASP this point then feel happy
It is not because you know this that you feel happy.

Relative clauses. If a negation appears in a relative clause, we annotate only the relative clause in the scope of negation but not the head noun it modifies. Unlike English, where the clause follows the head, Chinese displays the opposite order with the particle '的' in between the head and the relative clause. This is exemplified in (19), where the adjective marker '的' separates the relative clause '不爱出风头' and the head '人' .

(19) 他 是 个 不 爱 出风头 的 人
he is CL not like show-off DE person
'He is a person who does not like to show off.'

Nominal and adjectival predicates. When negation directly denies a state which is also the main predicate of a clause, the entire clause is under the scope. Whereas in English, these constructions are formed by copula followed by an adjective ("He is impatient"), Chinese do not require a copula. This is shown in (20).

(20) 这样 不 公正
This not fair
'This is unfair.'

With respect to negated adjectives, one important difference between Chinese and English is the status of affixal negation. While affixal negation in English often creates contraries rather than contradictions, hence not forming a scope, in Chinese, an adjective and its negated counterpart usually cannot be false at the same time, therefore abiding by the 'Law of the Excluded Middle'[4].

[4]Notice however the definition of affixal negation in Chinese is less clear.

(21) I am neither patient nor impatient
* 我 既不 耐心 也不 不耐心
I neither patient nor impatient

As to nominal predicates, where a noun phrase follows a copula (similar to the English "He is not a patient man"), we also annotate the entire clause in the scope, as shown in (22)

(22) 他 不 是 一个 耐心 的 人
He not BE a CL patient DE person
'He is not a patient man.'

Sentence final particles. Chinese is characterized by sentence-final mood particles that serve to express the attitude or mood of the speaker towards the whole sentence. Given that these particles are not affected by the presence of a negation cue, they are *not* included in the scope of negation; this decision is also supported by theories that define these particles as complementisers out of the IP (Paul, 2014). For example in (23), the emphatic mood introduced by the final '呀' is not affected by the negation in the sentence.

(23) 不 要 等 他过 了 山 呀!
not need wait he past ASP mountain MOOD
'Do not wait until he has past the mountain!'

Comparative constructions. In Chinese, comparison is expressed in most cases through the co-verb '比' , which takes as subject and object the two things compared, followed by the dimension they are compared along. This is the case in (24), where the subject and the object are compared for their age; In such cases where the negation scopes over the co-verb '比' , we annotate as scope the entire clause.

(24) 约翰森 先生 年纪 不 比 你 大
Johnson Mr. age not compare you old
'Mr. Johnson is not older than you.'

However, when negation scopes over only the dimension being compared (25), we distinguish this from the previous case by excluding from the scope the object of the comparison. This is in line with the English annotation for 'Compared to usual, my sleep hasn't been deep' where the phrase 'compared to usual' is excluded from the scope.

(25) 我 的 觉 睡的 比 平常 还要 不 踏实
I of sleep sleep compare normal even not sound
'Compared to usual, my sleep has not been deep'

3.3.3. Event

We annotate an event as being negated if it is factual and if it did not happen; 'eventuality' includes here both events and states. What the annotation considers as an event is a minimal unit in a negated phrase, usually corresponding to its head. An example of a negated event in verbal predicate is shown inside a box in (26)(we omit the scope just for presentational purposes). Although one could consider 吃羊肉, 'eat mutton', as the entire event, the event actually annotated is just its minimal unit, that is, the head verb 吃 'to eat'

(26) 我 不 [吃] 羊肉
I not eat mutton
'I do not eat mutton.'

Existential and copulative constructions. In existential constructions, we do *not* mark the verb 有, 'there is/are', as an event; instead, we mark as the event the head of the nominal phrase following the existential verb as shown in (27). This treatment is consistent with the English guidelines in Morante et al. (2011).

(27) 这里 没 有 [人]
here not there-is people
'There is nobody here'

As shown in (27), the existential construction in Chinese also encodes universal quantification (i.e. 'nobody'). When universal quantification applies to the subject of the clause, we annotate as event only the head of the verbal predicate. In (28), we therefore mark as event '动', 'to move' but not the aspect marker '在' (continuous action).

(28) 没 有 人 在 [动]
not there-is person ASP-cont. move
'Nobody is moving'

Modality. Given that we annotate factuality, we do *not* annotate as events those verbs in the scope of certain modals, in particular where the speaker is uncertain about the happening of an event. In English, this excludes most of epistemic and deontic modality (i.e. verbs introduced by auxiliaries such as 'should', 'would', etc.). In both English and Chinese, we do *not* annotate negated events in the scope of modals except for modality expressing the subject internal ability. This is the case of the modal '能' which is annotated only when expressing participant internal abilities (29.1) but not when expressing conjecture about a non-factual event (29.2).

(29) 1. 我 不 [能] 打 篮球
I not can play volleyball
'I cannot play volleyball'

2. 我 不 能 再 这样 了
I not can again like-this ASP
'I couldn't be like this anymore.'

Supposition or presumption. We also examine the semantics of the verb that is directly negated by the cue, when annotating events. If the verb suggests that the speaker is certain about the content of the predicate, we treat the head of the predicate as factual and annotate the negated event in the clause. If the verb suggests that the predicate is only supposed or presumed by the speaker, we do *not* annotate events in the predicate. This contrast is exemplified in (30.1) and (30.2), through the verbs '知道' , 'to know', where we annotate the event, and '相信' , 'to believe', where we do not.

(30) 1. 我 知道 您 决不 [愿意] [...]
I know you not want [...]
'I know that you do not wish [...'

2. 我 相信 您 决不 愿意 [...]
I believe you not want [...]
'I believe that you do not wish [...]'

4. Annotation Projection

4.1. Methodology

The goal of the annotation projection is to investigate whether we can ease the burden of annotating from scratch in the presence of parallel text.

Annotations are projected using word alignment information computed by the IBM model 2, as implemented in the fast_align toolkit (Dyer et al., 2013).

The training data for the alignment model consists of the aligned sentence pairs in NegPar and the English-Chinese UN parallel corpus (Ziemski et al., 2016); the Chinese side of the corpus was also tokenized using the Stanford Word Segmenter (Tseng et al., 2005). We used the symmetrical two-way alignment results as the basis for projection. In our work, we experimented with two types of alignment models: (1).English word to Chinese word (word-level projection); (2).English word to Chinese character (character-level projection). An example of the former is shown in Fig. 2 where all elements are projected correctly, except for the scope projection of the subject '我'('I').

For both levels of projection, we report precision, recall, F1 measure and number of gold (both English and Chinese) and projected spans for cue, event and scope independently, as they were projected as such.

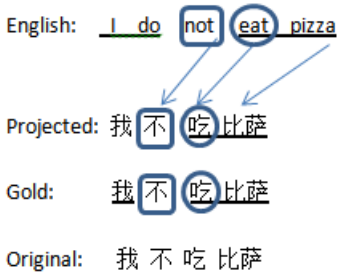


Figure 2: An example of projection from an English sentence to its Chinese translation: cue is indicated with a box around, event is circled and the scope is underlined. Word alignment (English word to Chinese word) is indicated by arrows.

4.2. Results in the projection task

The results for cue, scope and event projection on the development set are summarized in Table 2.

Considering F1 alone, we found word-level projection to yield better results for event and scope but not for cue projection. This can be explained by the fact that cues often span subword units (as in the case of morphological negation) and word-level projection might end up over-predicting cues in Chinese (hence the relatively lower precision). However, in terms of easing the annotation process, a relatively lower recall means that more work is required to find elements that the projection has missed, rather than

	Word-level			Character-level					
	precision	recall	F1	precision	recall	F1	gold-en	gold-zh	projection (% gold-zh)
cue	0.39	0.47	0.43	0.49	0.42	**0.45**	175	230	169(73%)
event	0.37	0.32	**0.34**	0.40	0.27	0.32	123	153	103(67%)
scope	0.64	0.48	**0.55**	0.64	0.44	0.50	170	226	168(73%)

Table 2: Performance of the annotation projection on the *dev* set both at word and character level for cue, event and scope respectively.

filtering out over-predicted elements (reflected in lower precision) which might be done with some post-processing heuristics.

In general, even in the presence of parallel data, detecting negation using annotation projection does not lead to good results. The number of projected spans vs. gold spans in Chinese and English suggests that this is in part due to differences in how negation is translated.

4.3. Error Analysis

To delve deeper into the errors in projecting the annotations from English, we carry out an error analysis. In doing so, we consider character-based projection for the cue and word-based projection for event and scope spans.

Cue. we first break down the performance of cue projection according to different target and source cues. First of all we found performance to vary across different cues as shown in Fig. 3. Compared to precision, recall is lower for two-character cues where projection often seems to miss either of the characters; this is the case of both 没有and 并不.

The low performance of 无and 未might be caused by 无and 未being common components in Chinese chengyu (idioms) that are frequently used to translate positive phrases in English as in (32) (8 cases).

Furthermore, we analyzed projection errors based on source cues as shown in Fig 3.(b). The relatively higher performance compared with Fig 3.(a) is not surprising as we exclude here a large number of errors from cases where negation expressed in Chinese but not in English. In general, when the English cues are correctly projected onto the Chinese cues, the projection might also include additional surrounding characters in Chinese as cues; therefore the recall is higher than precision. It is especially the case for the negation pronoun 'nothing' which maps to the negative polarity item (NPI) 什么都/一点也/一点/一, 'any/anything'. This accounts for 5 errors, one of which we report below (31).

(31) projected.: 从他那里**什么 都** 得不到
gold-zh: 从 他 那里 什么 都 得 **不** 到
from him there anything DOU get not POT
'gold-en: can get **nothing** from him'

Finally, we wanted to analyze those cases where projection fails due to the fact that negation is present in Chinese but not in English. we found this to happen for 2 different reasons:

1. A negation instance in Chinese is paraphrased in positive terms in English; this often concerns just a pair of contrary adjectives or adverbs as shown in (32), but also extends to entire clauses as shown in (33)

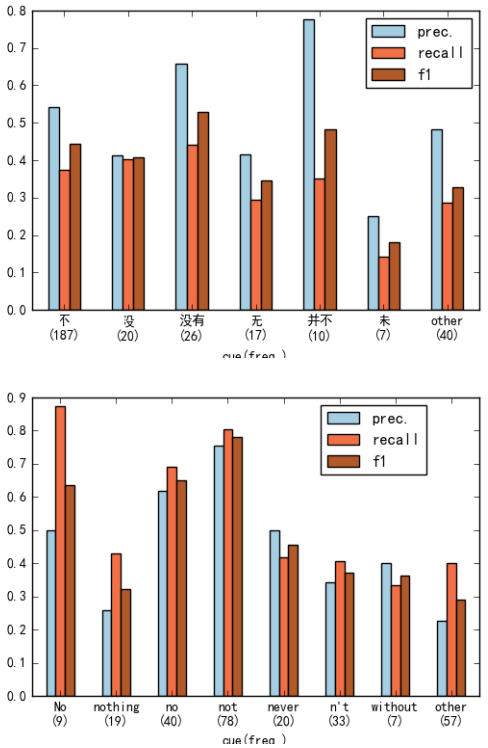


Figure 3: Performance of annotation projection plotted against Chinese (a) and English (b) cues

(32) gold-zh: 他 安然 **无** 恙
he safe not sick
gold-en: 'He is safe and sound.'

(33) gold-zh: 惊慌 的 脸上 **没** 有 一点 血色
panicked DE face not there-is one-bit blood-color
gold-en: 'Every tinge of colour struck from his astonished face'

2. Some lexical items in English can be interpreted as inherently expressing negation and thus can be translated as cues in Chinese, but they are usually not annotated as cues on the English side. This is the case in (34); along with 'hardly', we found other expressions such as 'rather than', 'absence', 'out of question' and 'refused' that can be translated into negation in Chinese.

(34) gold-zh: 这 件 事 的 前前后后 **不** 可能 是 为了 [...]
This CL thing of everything not can be for [...]
gold-en: 'The whole proceeding could hardly be for [...]'

Event. Out of the 153 gold events we found that only 13.3% were correctly projected from English, with 38.8%

of the cases where the projection does not detect an event at all. These false negatives are caused by the fact that Chinese translates positive terms in English as negation (same as the cases in (32)∼(34)) but in some cases are just due to English words aligning to a null token.
On the other hand, in 16% of the cases we observed that the event from English is projected onto a completely different span of the sentence. Some of these cases are however not due to alignment errors but because the Chinese side uses different constructions with emphasis on different events from English. For instance, in (35), the English guidelines annotate as event the nominal predicate 'colour' where this is translated in Chinese as a verb 说上 'say'.

(35) gold-zh: 那 张 脸 既 不 黑 [...] 说 不 上 是 什么 颜色
That CL face either not black [...] say not up be what colour
gold-en: It was **n't** black [...] **nor** any **colour**

Finally, in 13.3% of the cases, the projection only partially matches the gold annotation for an event. 13 of these 25 cases we found that the projection includes the negation cue inside the event. These are often cases such as (36), where a word containing morphological negation in English is projected onto both the cue and the event in Chinese.

(36) projected: '[...] 我们还弄 不清楚 的罪行'
gold-zh: [...] 我们 还 弄 不 清楚 的 罪行
[...] we still manage not clear DE crime
gold-en: 'They were all confederates in the same **un** known crime .'

Scope. Out of the 226 instances of negation scope we found that only 3 (0.01%) were fully and correctly projected, with 39% of the cases where the projection returns nothing. As discussed above, this is probably due to positive phrases being translated into negation in Chinese.
We found only 5 cases (0.02%) where the scope in English is projected to a completely different span from the gold span in the Chinese sentence. The majority of the errors (145/226 – 77%) concerns partial overlap, where the projection covers the gold scope only in part. A closer analysis shows that the projection tends to often miss the NPI 什么 (12 cases). 什么 corresponds to the English 'any' when in the scope of negation. In all the cases where projection fails to include this element in the scope, English uses the determiner 'no' or pronoun 'nothing' instead of an overt NPI eg. 'any'; therefore什么 is usually mapped to the negation cue rather than being marked as a scope element; this is exemplified in (37).

(37) projected: 这里面**没有什么**
gold-zh: 这里面 **没** 有 什么
Here not there-is anything
gold-en: **Nothing** in all this

Finally, to allow comparison with future work, Table 3 and 4 report the performance of annotation projection on the test set as well.

	precision	recall	F1
cue	0.372	0.428	0.398
scope	0.574	0.381	**0.458**
event	0.299	0.209	**0.246**

Table 3: Results from word-level projection of negation on the test data

	precision	recall	F1
cue	0.478	0.382	**0.425**
scope	0.583	0.312	0.406
event	0.338	0.180	0.235

Table 4: Results from character-level projection of negation on the test data

5. Conclusion

We have introduced *NegPar*, the first English-Chinese parallel corpus annotated for negation. The corpus is based on a pre-existing negation corpus for English, CONANDOYLE-NEG, whose annotation we have amended and extended onto its Chinese translation. To ease the burden of annotating from scratch, we have experimented with projecting the English annotations onto Chinese via word-alignments. Results have shown that the projection offers limited help as it would not deal with the different ways negation can be translated across languages.
Future work could investigate whether differences in translating negation vary according to languages and domains. The annotation projection algorithm could also be improved for example by considering syntactic mapping.

6. Appendix A: Abbreviations

ASP: aspect marker
BA: object marker
CL: classifier
DE: adjective marker
MOOD: mood particle
ASP-cont: Present tense marker
DOU: particle (all)
POT: verb complement expressing the potential form

7. Acknowledgements

We would like to thank the three anonymous reviewers for their useful comments. This project was also founded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 644402 (HimL). We would also like to thank Peterhouse College Cambridge for funding for the conference trip.

8. Bibliographical References

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics.
Hwa, R., Resnik, P., Weinberg, A., and Kolak, O. (2002). Evaluating translational correspondence using annotation projection. In Proceedings of the 40th Annual

Meeting on Association for Computational Linguistics, pages 392–399. Association for Computational Linguistics.

Konstantinova, N., De Sousa, S. C., Díaz, N. P. C., López, M. J. M., Taboada, M., and Mitkov, R. (2012). A review corpus annotated for negation, speculation and their scope. In LREC, pages 3190–3195.

Li, C. N. and Thompson, S. A. (1989). Mandarin Chinese: A functional reference grammar. Univ of California Press.

Morante, Roser and Daelemans, Walter. (2012). ConanDoyle-neg: Annotation of negation in Conan Doyle stories.

Morante, R., Schrauwen, S., and Daelemans, W. (2011). Annotation of negation cues and their scope: Guidelines v1. Computational linguistics and psycholinguistics technical report series, CTRS-003.

Padó, S. and Lapata, M. (2009). Cross-lingual annotation projection for semantic roles. Journal of Artificial Intelligence Research, 36(1):307–340.

Paul, W. (2014). Why particles are not particular: Sentence-final particles in chinese as heads of a split cp. Studia Linguistica, 68(1):77–115.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D., and Manning, C. (2005). A conditional random field word segmenter for sighan bakeoff 2005. In Proceedings of the fourth SIGHAN workshop on Chinese language Processing.

Vincze, V., Szarvas, G., Farkas, R., Móra, G., and Csirik, J. (2008). The bioscope corpus: biomedical texts annotated for uncertainty, negation and their scopes. BMC bioinformatics, 9(11):S9.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The united nations parallel corpus v1. 0. In LREC.

Zou, B., Zhu, Q., and Zhou, G. (2015). Negation and speculation identification in chinese language. In ACL (1), pages 656–665.

9. Language Resource References

Morante, Roser and Daelemans, Walter. (2012). ConanDoyle-neg: Annotation of negation in Conan Doyle stories.

The IIT Bombay English-Hindi Parallel Corpus

Anoop Kunchukuttan[†], Pratik Mehta[1‡], Pushpak Bhattacharyya[†]
[†]Center for Indian Language Technology,
Department of Computer Science and Technology,
Indian Institute of Technology Bombay.
[‡] College of Information and Computer Sciences,
University of Massachusetts Amherst.
{anoopk,pb}@cse.iitb.ac.in, psmehta@cs.umass.edu

Abstract

We present the IIT Bombay English-Hindi Parallel Corpus. The corpus is a compilation of parallel corpora previously available in the public domain as well as new parallel corpora we collected. The corpus contains 1.49 million parallel segments, of which 694k segments were not previously available in the public domain. The corpus has been pre-processed for machine translation, and we report baseline phrase-based SMT and NMT translation results on this corpus. This corpus has been used in two editions of shared tasks at the Workshop on Asian Language Translation (2016 and 2017). The corpus is freely available for non-commercial research. To the best of our knowledge, this is the largest publicly available English-Hindi parallel corpus.

Keywords: machine translation, parallel corpus, Indian languages

1. Introduction

Hindi is one of the major languages of the world, spoken primarily in the Indian subcontinent, and is a recognised regional language in Mauritius, Trinidad and Tobago, Guyana, and Suriname. In addition, it serves as a major *lingua franca* in India. According to the 2001 Census of India, Hindi has 422 million native speakers and more than 500 million total speakers (Wikipedia, 2017). It is also an official language of the Union Government of India as well as major Indian states like Uttar Pradesh, Bihar, Rajasthan, *etc.* and is used for conducting business and administrative tasks. Many languages and dialects in the Gangetic plains are closely related to Hindi *e.g.* Bhojpuri, Awadhi, Maithili, *etc.* Hindi is the fourth-most spoken language in the world, and third-most spoken language along with Urdu (both are registers of the Hindustani language). In contrast, English is spoken by just around 125 million people in India, of which a very small fraction are native speakers.

Hence, there is a large requirement for digital communication in Hindi and interfacing with the rest of the word via English. Hence, there is immense potential for English-Hindi machine translation. However, the parallel corpora available in the public domain is quite limited. This work is an effort to consolidate all publicly available parallel corpora for English-Hindi as well as significantly add to the available parallel corpus through corpora collected in the course of this work.

2. Dataset

The parallel corpus has been compiled from a variety of existing sources (primarily OPUS (Tiedemann, 2012), HindEn (Bojar et al., 2014b) and TED (Abdelali et al., 2014)) as well as corpora developed at the *Center for Indian Language Technology*[2] (CFILT), IIT Bombay over the years. The training corpus consists of sentences, phrases as well as dictionary entries, spanning many applications and domains.

2.1. Corpus Details

The details of the training corpus are shown in Table 1. We briefly describe the new sub-corpora we have added to the collection. For the corpora compiled from existing sources, please refer to the papers mentioned in the table.

Judicial domain corpus - I contains translations of legal judgements by in-house translators with many years of experience, though not with a legal background.

Judicial domain corpus - II contains translation done by graduate students taking a graduate course on natural language processing as part of a course project. This was part of an exercise of collecting translations in complex domain by non-expert translators. The translations included in the corpus were determined to be of good quality by annotators.

Mahashabdkosh is an online official terminology dictionary website[3] which is hosted by Department of Official Language, India. It contains Hindi as well as English terms along with definitions and example usage which are translations. The translation pairs were crawled from the website.

Indian Government corpora has been manually collected by CFILT staff from various websites related to the Indian government like the National Portal of India, Reserve Bank of India, Ministry of Human Resource Development, NABARD, *etc.*

Hindi-English Linked Wordnet contains bilingual dictionary entries created from the linked Hindi and English wordnets.

Gyaan-Nidhi Corpus is a multilingual parallel corpus between English and multiple Indian languages. The data is available in HTML format, hence it is not sentence aligned.

[1] work done at IIT Bombay

[2] www.cfilt.iitb.ac.in

[3] e-mahashabdkosh.rb-aai.in

Corpus Id	Source	Number of segments
1	GNOME (OPUS) (Tiedemann, 2012)	145,706
2	KDE4 (OPUS)	97,227
3	Tanzil (OPUS)	187,080
4	Tatoeba (OPUS)	4,698
5	OpenSubs2013 (OPUS)	4,222
6	HindEnCorp (Bojar et al., 2014b)	273,885
7	Hindi-English Linked Wordnets (Bhattacharyya, 2010)	175,175
8	Mahashabdkosh: Administrative Domain Dictionary* (Kunchukuttan et al., 2013)	66,474
9	Mahashabdkosh: Administrative Domain Examples*	46,825
10	Mahashabdkosh: Administrative Domain Definitions*	46,523
11	TED talks (Abdelali et al., 2014)	42,583
12	Indic Multi-parallel corpus (Alexandra Birch and Post, 2011)	10,349
13	Judicial domain corpus - I* (Kunchukuttan et al., 2013)	5,007
14	Judicial domain corpus - II* (Kunchukuttan et al., 2012)	3,727
15	Indian Government corpora*	123,360
16	Wiki Headlines (Provided by CMU: `www.statmt.org/wmt14/wiki-titles.tgz`)	32,863
17	Gyaan-Nidhi Corpus * (`tdil-dc.in/index.php?option=com_download&task=showresourceDetails&toolid=281`)	227,123
Total		**1,492,827**

Table 1: Details of the IITB English-Hindi Parallel Corpus (training set). * indicates new corpora not in the public domain previously.

	Language	Train	Test	Dev
#Sentences		1,492,827	2,507	520
#Tokens	eng	20,667,259	57,803	10,656
	hin	22,171,543	63,853	10,174
#Types	eng	250,782	8,957	2,569
	hin	343,601	8,489	2,625

Table 2: Statistics of data sets

We used the sentence alignment technique proposed by Moore (2002) to extract parallel corpora from this comparable corpus. This method combines sentence-length models and word-correspondence based models, and requires no language or corpus specific knowledge. We manually checked a small sample of 300 sentences from the parallel sentences extracted. We found that the precision of extraction of parallel sentences was 88.6%.

2.2. Corpus Statistics

The test and dev corpora consist of newswire sentences, which are the same ones as used in the WMT 2014 English-Hindi shared task (Bojar et al., 2014a). The training, dev and test corpora consist of 1,492,827 and 520 and 2507 segments respectively. Detailed Statistics are shown in Table 2. The Hindi and English OOV rate (for word types) is 11.4% and 6.7%.

3. Baseline Systems

We trained baseline machine translation models using the parallel corpus with popular off-the-shelf machine translation toolkits to provide benchmark translation accuracies for comparison. We trained phrase-based Statistical Machine Translation (PBSMT) systems as well as Neural Machine Translation systems for English-Hindi and Hindi-English translation.

3.1. Data Preparation

Text Normalization: For Hindi, characters with *nukta* can have two Unicode representations. In one case, the character and nukta are represented as two Unicode characters. In the other case, a single Unicode character represents the composite character. We choose the former representation. The normalization script is part of the *IndicNLP*[4] library .
For English, we used true-cased representation for our experiments. However, the parallel corpus being distributed is available in the original case.
Tokenization: We use the *Moses* tokenizer for English and the *IndicNLP* tokenizer for Hindi.

3.2. SMT Setup

We trained PBSMT systems with *Moses*[5] (Koehn et al., 2007). We used the *grow-diag-final-and* heuristic for extracting phrases, lexicalised reordering and Batch MIRA (Cherry and Foster, 2012) for tuning (default parameters). We trained 5-gram language models with Kneser-Ney smoothing using *KenLM* (Heafield, 2011). We used the *HindMono* (Bojar et al., 2014b) corpus for Hindi and the WMT NEWS Crawl 2015 corpus for English as additional monolingual corpora to train language models. These contain roughly 44 million and 23 million sentence for Hindi and English respectively.

3.3. NMT Setup

We trained a subword-level encoder-decoder architecture based NMT system with attention (Bahdanau et al., 2015).

[4] `anoopkunchukuttan.github.io/indic_nlp_library`
[5] `www.statmt.org/moses`

System	eng-hin		hin-eng	
	BLEU	*METEOR*	*BLEU*	*METEOR*
SMT	11.75	0.313	14.49	0.266
NMT	12.23	0.308	12.83	0.219

Table 3: Results for Baseline Systems

We used *Nematus* [6] (Sennrich et al., 2017) for training our NMT systems.

Vocabulary: We used Byte Pair Encoding (BPE) to learn the vocabulary (with 15500 merge operations) (Sennrich et al., 2016b). We used the *subword-nmt* [7] tool for learning the BPE vocabulary. Since the writing systems and vocabularies of English and Hindi are separate, BPE models are trained separately.

Network parameters: The network contains a single hidden encoder and decoder RNN layer, containing 512 GRU units each. The dimension of input and output embedding layers is 256 units.

Training details: The model is trained with a batch size of 50 sentences and maximum sentence length of 100 using Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.0001. The output parameters were saved after every 10,000 iterations. We used early-stopping based on validation loss with *patience=10*.

Decoding: We used a beam size of 12. We decoded the test set with an ensemble of four models (best model and the last three saved models).

3.4. Results

We evaluated our system using BLEU (Papineni et al., 2002) and METEOR (Banerjee and Lavie, 2005). We used a *METEOR-Indic*[8], a customized version of METEOR Indic, for evaluation of Hindi as target language. *METEOR-Indic* can perform synonym matches for Indian languages using synsets from IndoWordNet (Bhattacharyya, 2010). It can also perform stem matches for Indian languages using a trie-based stemmer (Bhattacharyya et al., 2014). This is useful for a morphologically rich language like Hindi.

Table 3 shows the results of our experiments.

4. Availability

The homepage for the dataset can be accessed here: `http://www.cfilt.iitb.ac.in/iitb_parallel`.

The new corpora we release are available for research and non-commercial use under a Creative Commons Attribution-NonCommercial-ShareAlike License [9]. The corpora we compiled from other sources are available under their respective licenses. The sub-corpora (in the corpus distribution that we make available) are in the same order as listed in the Table 1, so they can be separately extracted, if required (*e.g.* for domain adaptation).

[6] `github.com/EdinburghNLP/nematus`

[7] `github.com/rsennrich/subword-nmt`

[8] `github.com/anoopkunchukuttan/meteor_indic`

[9] `http://creativecommons.org/licenses/by-nc-sa/4.0`

5. Conclusion and Future Work

We presented the IIT Bombay English-Hindi Parallel corpus version 1.0, and provided benchmark baseline SMT and NMT results on this corpus. This corpus has been used for the two shared tasks (Workshop on Asian Language Translation 2016 and 2017). The *HindiEn* component of the corpus has also been used for the WMT 2014 shared task. The corpus is available under a Creative Commons Licence.

In future, we plan to enhance the corpus from additional sources, mostly websites of the Government of India which is still a largely untapped source of parallel corpora. We also plan to build stronger baselines like pre-ordering with PBSMT (Ramanathan et al., 2008) for English-Hindi translation, and use of synthetic corpora generated via back-translation for NMT systems (Sennrich et al., 2016a).

6. Acknowledgements

We thank past and present members of the Center for Indian Language Technology for their efforts in creating various parts of the corpora over the years: Pallabh Bhattacharjee, Kashyap Popat, Rahul Sharnagat, Mitesh Khapra, Jaya Jha, Rajita Shukla, Laxmi Kashyap, Gajanan Rane and many members of the Hindi WordNet team. We also thank the Technology Development for Indian Languages (TDIL) Programme and the Department of Electronics & Information Technology, Government of India for their support.

7. Bibliographical References

Abdelali, A., Guzman, F., Sajjad, H., and Vogel, S. (2014). The AMARA Corpus: Building Parallel Language Resources for the Educational Domain. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Alexandra Birch, Chris Callison-Burch, M. O. and Post, M. (2011). The Indic multi-parallel corpus. `http://homepages.inf.ed.ac.uk/miles/babel.html`.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*.

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*.

Bhattacharyya, P., Bahuguna, A., Talukdar, L., and Phukan, B. (2014). Facilitating multi-lingual sense annotation: Human mediated lemmatizer. In *Global WordNet Conference*.

Bhattacharyya, P. (2010). IndoWordNet. In *Language Resources and Evaluation Conference*.

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., et al. (2014a). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*.

Bojar, O., Diatka, V., Rychlỳ, P., Stranák, P., Suchomel, V., Tamchyna, A., and Zeman, D. (2014b). HindEnCorp-Hindi-English and Hindi-only Corpus for Machine Translation. In *Language Resources and Evaluation Conference*.

Cherry, C. and Foster, G. (2012). Batch tuning strategies for Statistical Machine Translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*.

Kunchukuttan, A., Roy, S., Patel, P., Ladha, K., Gupta, S., Khapra, M. M., and Bhattacharyya, P. (2012). Experiences in Resource Generation for Machine Translation through Crowdsourcing. In *Language Resources and Evaluation Conference*.

Kunchukuttan, A., Chatterjee, R., Roy, S., Mishra, A., and Bhattacharyya, P. (2013). TransDoop: A Map-Reduce based Crowdsourced Translation for Complex Domain. In *Conference of the Association of Computational Linguistics (System Demonstrations)*.

Moore, R. (2002). Fast and accurate sentence alignment of bilingual corpora. *Machine Translation: From Research to Real Users*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Association for Computational Linguistics*.

Ramanathan, A., Hegde, J., Shah, R., Bhattacharyya, P., and Sasikumar, M. (2008). Simple Syntactic and Morphological Processing Can Help English-Hindi Statistical Machine Translation. In *International Joint Conference on Natural Language Processing*.

Sennrich, R., Haddow, B., and Birch, A. (2016a). Edinburgh Neural Machine Translation Systems for WMT 16. In *Workshop on Machine Translation (Shared Task)*.

Sennrich, R., Haddow, B., and Birch, A. (2016b). Neural machine translation of rare words with subword units. In *Conference of the Association of Computational Linguistics*.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017). Nematus: a Toolkit for Neural Machine Translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In *Language Resources and Evaluation Conference*.

Wikipedia. (2017). Hindi. `https://en.wikipedia.org/w/index.php?title=Hindi&oldid=802224343`. [Online; accessed 02-October-2017].

Extracting an English-Persian Parallel Corpus from Comparable Corpora

Akbar Karimi[*], Ebrahim Ansari[*], Bahram Sadeghi Bigham[*]
[*] Department of Computer Sciences and Information Technology, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran
{ak.karimi, ansari, b_sadeghi_b}@iasbs.ac.ir

Abstract
Parallel data are an important part of a reliable Statistical Machine Translation (SMT) system. The more of these data are available, the better the quality of the SMT system. However, for some language pairs such as Persian-English, parallel sources of this kind are scarce. In this paper, a bidirectional method is proposed to extract parallel sentences from English and Persian document aligned Wikipedia. Two machine translation systems are employed to translate from Persian to English and the reverse after which an IR system is used to measure the similarity of the translated sentences. Adding the extracted sentences to the training data of the existing SMT systems is shown to improve the quality of the translation. Furthermore, the proposed method slightly outperforms the one-directional approach. The extracted corpus consists of about 200,000 sentences which have been sorted by their degree of similarity calculated by the IR system and is freely available for public access on the Web[1].

Keywords: Parallel Sentence Extraction, Comparable Corpora, Statistical Machine Translation, Wikipedia, English-Persian Corpus

1. Introduction

Due to the abundance of data on the Internet, statistical machine translation (SMT) has gained more popularity. In order to build an SMT system, parallel corpora are of high importance. These parallel resources which have been aligned on the sentence level in two languages (source and target), are used in the training phase of the SMT system. Therefore, the larger the parallel corpora are, the better the performance of the SMT system is. However, for some language pairs such as Persian-English not much data of this type is available. This lack of parallel data has led researchers to make use of other available data called comparable corpora which contain a mixture of parallel and partially parallel sentences. They can be given a certain degree of comparability which ranges from lowly comparable to highly comparable (Li and Gaussier, 2010). Research shows that using these corpora can help improve the performance of the SMT system.

There are several sources such as news articles, company manuals, Wikipedia articles, and so forth which can be considered as comparable corpora. In this work, our aim is to extract parallel sentences for Persian-English language pair from Wikipedia documents using a new approach to improve the Persian-English SMT system.

Our method consists of two main parts: translation and information retrieval. For the translation part (Persian to English and English to Persian), we employed Moses Toolkit developed by Koehn et al., (2007) which is an open-source toolkit developed for phrase-based translation and for the IR step we utilized the Lucene IR system[2]. Lucene has been designed to work with queries which are fed into the system one by one and the results shown by the IR system for a query are numbers representing the degree of the documents' relevance to the query. To compute the similarity of two sentences, Lucene's original source code was modified so that the queries could be read from a text file and the most relevant sentences from another file could be given as the result of each query by the IR system. To carry out our experiments, we needed documents in Persian and English whose topics were the same. Therefore, we downloaded the document aligned Persian-English Wikipedia from Linguatools[3]. It is an XML file that contains the English documents for each of which there is a Persian entry. There are 363183 document pairs in this file.

The rest of the paper is as follows: In Section 2, a review of some of the related work is presented. Section 3 is dedicated to describing our method. Then, a detailed explanation of our experiments and their results is given in Section 4 and the final section concludes the paper.

2. Related Word

There have been several papers written on the use of comparable corpora. Due to the lack of enough parallel data for many language pairs, some have also proposed using English as a pivot language to extract parallel resources and for translation purposes.

Resnik et al., (2003), working on web pages, use STRAND, which is their structural filtering system, to recognize parallel pairs. In order to do so, they specify a set of pair-specific values and experiment on English-Chinese corpus, reporting precision and recall of 98 percent and 61 percent, respectively.

Koehn et al., (2005) extract parallel texts for 11 languages from the proceedings of the European Parliament to be used as the training data for building SMT systems. Smith et al., (2010) work on the document level aligned Wikipedia data for three language pairs, Spanish-English, Bulgarian-English, and German-English, and using Hidden Markov Model for word alignment, they extract parallel sentences for the aforementioned language pairs and build improved SMT systems.

Using a very small parallel corpus which contains only 100 thousand words and a bilingual dictionary, Munteanu and Marcu (2005) train a maximum entropy classifier to extract parallel sentences from large comparable corpora. They work with Arabic-English and French-English language pairs to carry out their experiments. In another work (Munteanu and Marcu, 2006), they extract sub-sentential

[1] https://iasbs.ac.ir/~ansari/nlp/pepc.html
[2] https://lucene.apache.org
[3] http://linguatools.org

fragments from non-parallel corpora that do not contain any parallelism on the sentence level.

Stefanescu and Ion (2013) work on Wikipedia to extract parallel sentences for English-German, English-Romanian, and English-Spanish language pairs. In order to find the parallel sentences from the comparable documents for each language pair, they make use of LEXACC, a tool developed by ACCURAT project for extraction of parallel sentences.

Do et al., (2010) propose a fully unsupervised method for parallel sentence extraction in which they build an SMT system using not parallel data but comparable data, and with this system, they translate the sentences from the source side of another comparable corpus to the target language. Then, they evaluate the translations by BLEU, NIST, and TER evaluation metrics, refeeding the ones that have been recognized as parallel into the SMT system and repeating the process. They claim that first few iterations of this process helps increase the number of parallel sentences resulting in improvements in the quality of the SMT system. In another work, using the aforementioned method and the English as the pivot language and a method called triangulation, Do et al., (2010) make an attempt to translate from Vietnamese to French.

Ansari et al., (2017) work on Persian-Italian languages using English as the pivot language. Sentences from Persian and Italian are translated into English and compared with each other by a new similarity metric which is based on Normalized Google Distance (NGD).

Linard et al., (2015) propose two approaches to bilingual lexicon extraction using English as the pivot language. One is to translate the source language to pivot and from that to the target language. The second approach is to translate both of the source and target languages into pivot language and then extract bilingual vocabulary.

Bakhshaei et al., (2015) introduce a generative model based on LDA concept to extract fragments and show that the baseline system with the additional fragments perform better than the baseline system alone.

Aker et al., (2013) use an SVM binary classifier for the extraction of bilingual terminology and they claim to have achieved an accuracy of 100% for the classifier. In another attempt to extract bilingual lexica, Seo et al., (2015) use self-organizing maps on comparable corpora for Korean-French and Korean-Spanish language pairs.

Using bootstrapping, Fung et al., (2004) work on very-non-parallel corpora and present a method for parallel sentence extraction, claiming that their method is 50% more effective than the baseline system. In their work, after matching the documents and extracting some parallel sentences, they rematch them based on the number of extracted parallel sentences and then carry out bootstrapping. The reason why they do this is due to a principle that they call "find-one-get-more" which means that if a sentence pair can be found in a document, more sentences are likely to exist in the same documents.

Rauf and Schwenk (2009, 2011) build an SMT system to translate one side of their bilingual corpus to be used as queries in an IR system to find their equivalents in the target language. To filter out the candidate sentences for each query, they use evaluation metrics such as word error rate (WER), translation error rate (TER), and translation error rate plus (TERp). They work with Arabic-English and French-English language pairs and report significant improvements in BLEU score.

3. Our Approach

When using a translation-based method to extract parallel sentences, the quality of the machine used for translation plays an important role. Since translating only one side of the corpus into another, which we call one-directional method, is not done flawlessly, it seems that if both sides were to be translated and used as queries, it would result in extracting better equivalents from the comparable corpus. We call this a bidirectional approach whose architecture is shown in Figure 1.

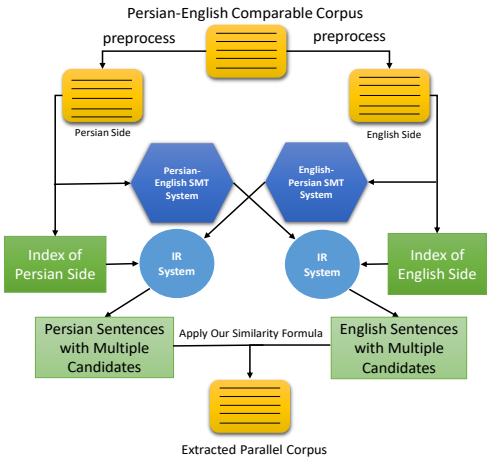


Figure 1: The architecture of the proposed bidirectional method for parallel sentence extraction.

As can be seen in the diagram, we first obtain the data from the Internet after which the data is translated in both directions and then parallel sentences are extracted from the candidate sentences found by Lucene IR system. Each of these steps is discussed in depth in the following sections.

3.1 Data Preparation

Wikipedia articles from which we want to extract parallel sentence have been made available by Linguatools website. The files are in XML file format containing all the documents in Wikipedia for many language pairs. The Persian-English corpus was downloaded for extracting parallel sentences. Since it is an XML file, it contains markup language, links, tables, figures, and so forth. Therefore, all the unnecessary characters in the file need to be removed first. This was carried out by writing a python script, and as a result, we obtained two plain texts containing only Persian and English sentences. In the process of obtaining plain texts from them, we ignored some documents and some sentences. If the number of sentences in a document was lower than 0.3 times the other's, both documents were ignored. In addition, when choosing the sentences from the selected documents, the sentences with the length of lower than 8 words did not make it to the final plain texts. With these limitations, the English text contained about 1.4 million sentences and the Persian text one million sentences.

Documents		Sentences	
English	Persian	English	Persian
363,183	363,183	9,933,618	1,789,632
145,479	145,479	1,391,214	1,021,103

Table 1: Number of documents and sentences, before (first row) and after (second row) preprocessing

These two plain texts were translated by the initial SMT systems which were trained on Open Parallel Corpus (OPUS) (Lison and Tiedemann, 2016).

3.2 Our Method

Our method consists of two main steps: translation and extraction (information retrieval). In translation step we utilize a bidirectional translation approach to extracting parallel sentences from comparable corpora. Two SMT systems are built, one translating from Persian to English and another doing translation from English to Persian, using Moses translation toolkit. Then Lucene IR system is utilized to measure the similarity of sentence pairs. Two similarity scores for a sentence pair are produced by the IR system, one, Sim_{en-fa}, for the original Persian sentence and the sentence translated by English-Persian SMT system and another, Sim_{fa-en}, for the original English sentence and the one translated by Persian-English SMT system. Based on these two scores, we develop a formula to calculate one similarity score for each sentence pair which is as follows:

$$BiSimilarity = \frac{\alpha}{\alpha + Penalty} \times \frac{\beta \times Sim_{fa-en} + Sim_{en-fa}}{\beta+1} \quad (1)$$

The coefficient β in the formula represents the logarithm of relative translation quality of one machine against the other. Since the quality of the two SMT systems were different, we decided to assign them different weights in the formula. The quality of the Persian-English system, which is 19.78 by BLEU evaluation metric (Papineni et al., 2002), is almost triple that of English-Persian system (7.80). However, assigning the weight 3 to Persian-English system against 1 to the other one would make English-Persian system almost uninfluential in the process of parallel sentence extraction, hence the weight 1.5 as opposed to 1 which, relatively, are the logarithmic values of the qualities of the two SMT systems. In addition, we penalize the sentences that have more or less number of words than their translated equivalents. Therefore, the Penalty variable, which is the difference in the word number of the two sentences, makes the similarity score smaller when the difference is too large. The average number of words in each sentence in the corpus is α which in our experiments was 22.

3.2.1 Translation

Moses toolkit has been widely used for translation in recent years. Therefore, we chose this toolkit for translating our plain texts. The initial systems for translating Wikipedia articles were built on OPUS collection which is a parallel collection of movie subtitles in many languages and is available online for public access. The Persian-English corpus we downloaded consisted of more than 3.7 million sentences in both languages. Three and a half million sentences were used for training, 200 for tuning, and 200,000 for testing. The BLEU scores of the baseline systems were 19.78 and 7.80 for Persian to English and English to Persian, respectively. To build the translation systems, the default settings for Giza++ and SRILM toolkit were used.

3.2.2 Information Retrieval

We employed Lucene IR system for extracting parallel sentences. Lucene is a java program which can be used for indexing all the documents in a directory and performing queries on the indexed files. The queries can consist of several words and the results shown by the IR system are the most relevant documents to a given query with a score representing the degree of their relevance. The formula with which Lucene measures the relevance of a document is based on term frequency and inverse document frequency. The documents are ranked with the most relevant as number one and the least relevant at the end. We made use of Lucene to measure the similarity of the translated sentences and the original ones. For each English sentence, 10 Persian candidate sentences were recognized with their similarity scores calculated by the IR system and the same was done for each Persian sentence. Then, using the score for each sentence pair in Formula 1, we chose the candidates that scored the highest. Also we allowed two candidates to be chosen for one sentence when it was possible.

4. Experiments

In this section, first some detailed information about the extracted corpus is given, and then the results of several experiments which were conducted on the extracted sentences are presented.

In order to compare the bidirectional method with one-directional method, both methods were implemented which resulted in the extraction of 158339 sentences by one-directional method and 199936 sentences by bidirectional one. The extracted sentences have been sorted by their degree of similarity score calculated by the IR system and Formula 1. The produced scores were divided into 6 intervals to determine the number of sentence pairs that belong to each interval. The result is presented in Figure 2.

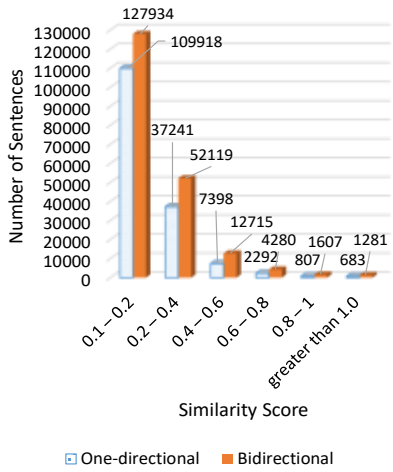


Figure 2: The number of extracted sentences by each method with their similarity scores

Similarity Score	Extracted Sentences
3.6	On the eve of his execution he talked of suicide در آستانه اعدام او از قصد خودکشی صحبت کرد
1.5	Mycobacteria infect many different animals including birds rodents and reptiles میکوباکتری‌ها بسیاری از حیوانات مختلف را آلوده می‌کنند، از جمله پرندگان،جوندگان، و خزندگان
1.0	The two kinds of variables commonly used in Smalltalk are instance variables and temporary دو نوع متغیر معمول استفاده شده در اسمالتاک متغیرهای نمونه و متغیرهای موقت هستند
0.9	Hatem was born in Burbank California and grew up in Monterey Park California هتم در بربنک کالیفرنیا بدنیا آمد و در مانتری پارک کالیفرنیا بزرگ شد
0.7	Today their fragmented and partly degraded range extends from India in the west to China and Southeast Asia امروزه باقی مانده زیستگاه ببرها از هند در غرب تا چین و آسیای جنوب شرقی کشیده شده است
0.5	It is bordered to the north by the Gulf of Finland to the west by the Baltic Sea to the south by Latvia and to the east by Lake Peipus and Russia این کشور از غرب به دریای بالتیک و از شمال به خلیج فنلاند می‌رسد و مرز شرقی آن با روسیه و مرز جنوبی آن با لتونی مشترک است
0.3	The difficulty is getting enough data of the right kind to support the particular method دشواری کار ترجمه خودکار، بدست آوردن اطلاعات کافی از نوع صحیح آن برای پشتیبانی روشی خاص است
0.1	The role of the passive audience therefore has shifted since the birth of New Media and an ever-growing number of participatory users are taking advantage of the interactive opportunities especially on the Internet to create independent content تولید محتوا همچنان در حال رشد است و مصرف محتوا ناشی از رشد موبایل و وسایل دیگر هم در حال رشد و گسترش است

Table 2: A sample of extracted sentence pairs. In each entry, first sentence is an English sample and the second line is corresponding extracted Persian one

By Looking at Figure 2, it can be observed that there are few sentence pairs with a higher similarity score than 0.4. They make up almost 10 percent of the corpus. Yet, these are the ones that contribute much to the performance of the systems built on this corpus (Figure 3). Although the contribution of the other 90 percent is small, it is still noticeable. In Table 2, a sample of sentences extracted by the proposed method is presented.

We checked the quality of the extracted sentences in two ways: (1) by building an SMT system using only the extracted sentences (Figure 3) and (2) by building a baseline SMT system using 500,000 sentences from OPUS collection and then adding the extracted sentences to the baseline system (Table 3). In both ways, our method performed slightly better than one-directional method.

To tune and test the SMT systems trained on the extracted sentences, we collected 200 sentences (ak-tune-200) for the tuning part and 1000 sentences (ak-test-1k) for testing. Five hundred sentences from the test collection are the ones translated by some colleagues of ours at an English institute. We collected the other 500 and also the 200 sentences of the tuning collection from some websites which offered free parallel sentences. These sentences are all taken from paper abstracts. We proofread all of them one by one to make sure that they have been translated correctly and also made sure that none of them was taken from Wikipedia or movie subtitles. The language model was built by combining Wikipedia documents with OPUS collection.

The first 100,000 sentences extracted by one-directional method have been named ‘one-directional-100k’ and the ones extracted by our method have been named ‘Bidirectional-100k’. Because of the randomness hidden in the tuning phase of an SMT system, every time it is implemented, the result of the translation can be slightly different. In order to obtain more reliable results, we implemented the SMT systems 3 times for each test. Therefore, the BLEU scores shown in the Table 3 are the average of the three scores.

In order to compare the quality of the extracted sentences with the OPUS collection, we built another system using 3.5 million sentences from the OPUS collection. As can be seen in Table 3, its quality was lower than that of the system built by our extracted data although the number of sentences in OPUS3.5M was 35 times higher than that of ours. This can be attributed to the nature of the OPUS collection which is a collection of movie subtitles, making it unable to translate formal sentences with good quality. The number of sentences in the test set can also affect the BLEU score. To show this, we added 4000 more sentences

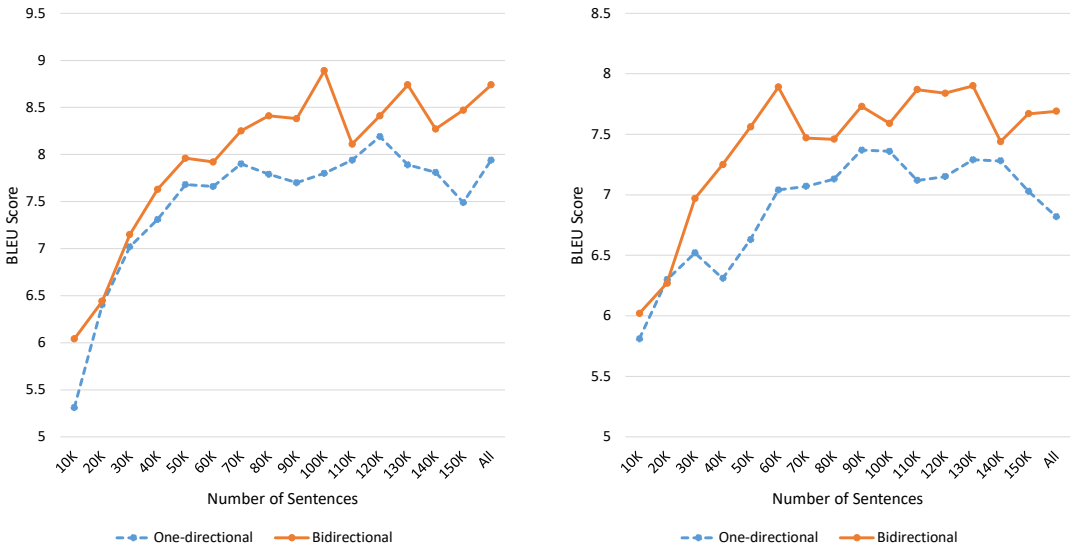


Figure 3: The performances of SMT systems built on various numbers of extracted sentences by both methods. Left: Persian-English system. Right: English-Persian system

to the test set and as is shown for the third test set in Table 3, the BLEU score went up by almost 1.5 points. In addition to Persian-English SMT systems, experiments with English-Persian SMT systems were carried out whose results can be seen in Table 3. As can be seen, the trend for the latter is similar to the former, indicating that the proposed method outperforms the one-directional method in both directions.

Test Sets	**Corpus**	BLEU Fa-En	BLEU En-Fa
ak-test-1k	OPUS3.5M	6.08	1.57
	One-directional-100k	7.94	7.05
	Bidirectional-100k	**8.39**	**7.58**
ak-test-1k + OPUS1k	OPUS500k	6.22	3.54
	OPUS500k + One-directional-100k	9.92	7.86
	OPUS500k + Bidirectional-100k	**10.21**	**8.23**
ak-test-1k + OPUS5k	OPUS500k	9.11	5.26
	OPUS500k + One-directional-100k	11.45	7.02
	OPUS500k + Bidirectional-100k	**11.70**	**7.30**

Table 3: Results of experiments with Persian-English and English-Persian SMT systems using 500k sentences of OPUS collection and 100k sentences of the extracted corpora by one-directional and bidirectional methods

It is worth noting that if the number of sentences from OPUS collection in the test set increases, the quality of English-Persian system is not guaranteed to improve as is the case with our test sentences combined with 5000 thousand sentences from OPUS. In this case, although the Persian-English system translates better than previous ones, the quality of English-Persian system drops. One way for this anomaly to be explained is by looking at the Persian side of our extracted corpus and that of OPUS corpus. Since the former is used for language modeling and is significantly different from the latter in terms of grammatical structure and the use of words, not to mention the inconsistencies prevalent in the typesetting of the OPUS collection, when more sentences from OPUS are added to the test set, the system's quality deteriorates. This is not the case regarding Persian-English system due to the fact that English side contains less problematic typesetting.

5. Conclusion

Parallel corpora are an important part of a statistical machine translation system. However, there is a lack of such data available for everyone. In this paper, a bidirectional method to extract parallel sentences from Wikipedia documents was proposed. The documents were translated from Persian to English and also in the reverse direction in order to find equivalent sentences. Furthermore, a similarity score was proposed to choose the best equivalents. Several different experiments with Persian-English and English-Persian SMT systems were carried out to show the quality of the extracted corpus. It was shown that existing SMT systems performed better when the extracted sentences were added to the systems. It was also demonstrated that the corpus extracted by bidirectional method performs better than the corpus extracted by one-directional approach by approximately 0.5 points in BLEU score. The sentences extracted by both methods have been made available online. As future work, instead of translating the documents by a statistical machine translation system, deep learning models such as word2vec, which are becoming more popular due to their high performance compared to statistical models, can be used for translation.

6. Acknowledgements

We would like to thank our colleagues, Zahra Sepehri and Ailar Qaraie, at Iranzamin Language School for providing us with 500 sentences used in our test set.

7. Bibliographical References

AbduI-Rauf, S., & Schwenk, H. (2009). On the use of comparable corpora to improve SMT performance. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (pp. 16-23). Association for Computational Linguistics.

Aker, A., Paramita, M., & Gaizauskas, R. (2013). Extracting bilingual terminologies from comparable corpora. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 402-411).

Ansari, E., Sadreddini, M. H., Sheikhalishahi, M., Wallace, R., & Alimardani, F. (2017). Using English as Pivot to Extract Persian-Italian Parallel Sentences from Non-Parallel Corpora. *arXiv preprint arXiv:1701.08339*.

Bakhshaei, S., Khadivi, S., & Safabakhsh, R. (2015). A Generative Model for Extracting Parallel Fragments from Comparable Documents. ACL-IJCNLP 2015, 43.

Do, T. N. D., Besacier, L., & Castelli, E. (2010). A fully unsupervised approach for mining parallel data from comparable corpora. In European COnference on Machine Translation (EAMT) 2010 (p. xx).

Do Thi Ngoc Diep, L. B., & Castelli, E. Improved Vietnamese-French Parallel Corpus Mining Using English Language.

Fung, P., & Cheung, P. (2004). Mining Very-Non-Parallel Corpora: Parallel Sentence and Lexicon Extraction via Bootstrapping and E. In EMNLP (pp. 57-63).

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In MT summit (Vol. 5, pp. 79-86).

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., ... & Dyer, C. (2007). Moses: Open source toolkit for statistical machine translation. In Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions (pp. 177-180). Association for Computational Linguistics.

Li, B., & Gaussier, E. (2010). Improving corpus comparability for bilingual lexicon extraction from comparable corpora. In Proceedings of the 23rd International Conference on Computational Linguistics (pp. 644-652). Association for Computational Linguistics.

Linard, A., Daille, B., & Morin, E. (2015). Attempting to bypass alignment from comparable corpora via pivot language. ACL-IJCNLP 2015, 32.

Lison, P., & Tiedemann, J. (2016). OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In LREC.

Munteanu, D. S., & Marcu, D. (2005). Improving machine translation performance by exploiting non-parallel corpora. Computational Linguistics, 31(4), 477-504.

Munteanu, D. S., & Marcu, D. (2006, July). Extracting parallel sub-sentential fragments from non-parallel corpora. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (pp. 81-88). Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics (pp. 311-318). Association for Computational Linguistics.

Rauf, S. A., & Schwenk, H. (2011). Parallel sentence generation from comparable corpora for improved SMT. Machine translation, 25(4), 341-375.

Resnik, P., & Smith, N. A. (2003). The web as a parallel corpus. Computational Linguistics, 29(3), 349-380.

Seo, H. W., Cheon, M. A., & Kim, J. H. (2015). Extracting Bilingual Lexica from Comparable Corpora Using Self-Organizing Maps. ACL-IJCNLP 2015, 62.

Smith, J. R., Quirk, C., & Toutanova, K. (2010). Extracting parallel sentences from comparable corpora using document level alignment. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (pp. 403-411). Association for Computational Linguistics.

Ștefănescu, D., & Ion, R. (2013). Parallel-Wiki: A collection of parallel sentences extracted from Wikipedia. In Proceedings of the 14th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2013) (pp. 24-30).

Learning Word Vectors for 157 Languages

Edouard Grave[1,*] **Piotr Bojanowski**[1,*] **Prakhar Gupta**[1,2] **Armand Joulin**[1] **Tomas Mikolov**[1]
[1]Facebook AI Research [2]EPFL
{egrave,bojanowski,ajoulin,tmikolov}@fb.com, prakhar.gupta@epfl.ch

Abstract

Distributed word representations, or word vectors, have recently been applied to many tasks in natural language processing, leading to state-of-the-art performance. A key ingredient to the successful application of these representations is to train them on very large corpora, and use these pre-trained models in downstream tasks. In this paper, we describe how we trained such high quality word representations for 157 languages. We used two sources of data to train these models: the free online encyclopedia Wikipedia and data from the common crawl project. We also introduce three new word analogy datasets to evaluate these word vectors, for French, Hindi and Polish. Finally, we evaluate our pre-trained word vectors on 10 languages for which evaluation datasets exists, showing very strong performance compared to previous models.

Keywords: word vectors, word analogies, fasttext

1. Introduction

Distributed word representations, also known as word vectors, have been widely used in natural language processing, leading to state of the art results for many tasks. Publicly available models, which are pre-trained on large amounts of data, have become a standard tool for many NLP applications, but are mostly available for English. While different techniques have been proposed to learn such representations (Collobert and Weston, 2008; Mikolov et al., 2013b; Pennington et al., 2014), all rely on the *distributional hypothesis* – the idea that the meaning of a word is captured by the contexts in which it appears. Thus, the quality of word vectors directly depends on the amount and quality of data they were trained on.

A common source of data to learn word representations, available in many languages, is the online encyclopedia Wikipedia (Al-Rfou et al., 2013). This provides high quality data which is comparable across languages. Unfortunately, for many languages, the size of Wikipedia is relatively small, and often not enough to learn high quality word vectors with wide coverage. An alternative source of large scale text data is the web and resources such as the common crawl. While they provide noisier data than Wikipedia articles, they come in larger amounts and with a broader coverage.

In this work, we contribute high quality word vectors trained on Wikipedia and the Common Crawl corpus, as well as three new word analogy datasets. We collected training corpora for 157 languages, using Wikipedia and Common Crawl. We describe in details the procedure for splitting the data by language and pre-processing it in Section 2. Using this data, we trained word vectors using an extension of the fastText model with subword information (Bojanowski et al., 2017), as described in Section 3. In Section 4, we introduce three new word analogy datasets for French, Hindi and Polish and evaluate our word representations on word analogy tasks. Overall, we evaluate our word vectors on 10 languages: Czech, German, Spanish, Finnish, French, Hindi, Italian, Polish, Portuguese and Chinese. We plan to release models for 157 languages other than English, which were not all evaluated in this paper.

Related work. In previous work, word vectors pre-trained on large text corpora have been released alongside open source implementation of word embedding models. English word vectors trained on a part of the Google News dataset (100B tokens) were published with `word2vec` (Mikolov et al., 2013b). Pennington et al. (2014) released `GloVe` models trained on Wikipedia, Gigaword and Common Crawl (840B tokens). A notable effort is the work of Al-Rfou et al. (2013), in which word vectors have been trained for 100 languages using Wikipedia data.

2. Training Data

We train our word vectors using datasets composed of a mixture of Wikipedia and Common Crawl.

2.1. Wikipedia

Wikipedia is the largest free online encyclopedia, available in more than 200 different languages. Because the articles are curated, the corresponding text is of high quality, making Wikipedia a great resource for (multilingual) natural language processing. It has been applied to many different tasks, such as information extraction (Wu and Weld, 2010), or word sense disambiguation (Mihalcea, 2007). We downloaded the XML Wikipedia dumps from September 11, 2017. The first preprocessing step is to extract the text content from the XML dumps. For this purpose, we used a modified version of the `wikifil.pl` script[1] from Matt Mahoney.

Even if Wikipedia is available for more than 200 languages, many dumps are relatively small in size (compared to the English one). As an example, some widely spoken languages such as Hindi, have relatively small Wikipedia data (39 millions tokens). Overall, 28 languages contain more than 100 millions tokens, and 82 languages contain more than 10 millions tokens. We give the number of tokens for the 10 largest Wikipedia in Table 1. For these reasons (and the fact that Wikipedia is restricted to encyclopedic

[*] The two first authors contributed equally.

[1] `http://mattmahoney.net/dc/textdata.html`

Language	# tokens	# words
German	1,384,170,636	3,005,294
French	1,107,636,871	1,668,310
Japanese	998,774,138	916,262
Russian	823,849,081	2,230,231
Spanish	797,362,600	1,337,109
Italian	702,638,442	1,169,177
Polish	386,874,622	1,298,250
Portuguese	386,107,589	815,284
Chinese	374,650,371	1,486,735
Czech	178,516,890	784,896
Finnish	127,176,620	880,713
Hindi	39,733,591	183,211

Table 1: Comparison of the size of the Wikipedia corpora for selected languages. The second column indicates the number of words which appear at least five times in the corpus.

	TCL		Wikipedia		EuroGov	
Model	Acc.	Time	Acc.	Time	Acc.	Time
`langid.py`	93.1	8.8	91.3	9.4	98.7	13.1
`fastText`	94.7	1.3	93.0	1.3	98.7	2.9

Table 2: Accuracy and processing time of our language detector and `langid.py` on three publicly available datasets. The TCL dataset was converted to UTF-8.

domains), we decided to also use data from the common crawl to train our word vectors.

2.2. Common Crawl

The common crawl is a non profit organization which crawls the web and makes the resulting data publicly available. This large scale corpus was previously used to estimate n-gram language models (Buck et al., 2014) or to learn English word vectors (Pennington et al., 2014). To the best of our knowledge, it was not used yet to learn word vectors for a large set of languages. The data is distributed either as raw HTML pages, or as WET files which contain the extracted text data, converted to UTF-8. We decided to use the extracted text data, as it is much smaller in size, and easier to process (no need to remove HTML). We downloaded the May 2017 crawl, corresponding to roughly 24 terabytes of raw text data.

Language Identification. The first preprocessing step consists in splitting the data based on the language. As noted by Buck et al. (2014), some pages contain text in different languages. We thus decided to detect the language of each line independently. For this purpose, we built a fast language detector using the `fastText` linear classifier (Joulin et al., 2017), which can recognize 176 languages. We used 400 millions tokens from Wikipedia (described in the previous section) as well as sentences from the Tatoeba website[2] to train our language detector. The model uses character ngrams of length 2, 3 and 4 as features, and a hierarchical softmax for efficiency. We evaluate our model on publicly available datasets from Baldwin and Lui (2010) and report results in Table 2. Our approach compares favorably to existing methods such as `langid.py` (Lui and Baldwin, 2012), while being much faster. This language detector will be released along the other resources described in this article. After language identification, we only keep lines of more than 100 characters and with a high confidence score (≥ 0.8).

[2] `www.tatoeba.org`

Language	# tokens	# words
Russian	102,825,040,945	14,679,750
Japanese	92,827,457,545	9,073,245
Spanish	72,493,240,645	10,614,696
French	68,358,270,953	12,488,607
German	65,648,657,780	19,767,924
Italian	36,237,951,419	10,404,913
Portuguese	35,841,247,814	8,370,569
Chinese	30,176,342,544	17,599,492
Polish	21,859,939,298	10,209,556
Czech	13,070,585,221	8,694,576
Finnish	6,059,887,126	9,782,381
Hindi	1,885,189,625	1,876,665

Table 3: Comparison accross languages of the size of the datasets obtained using the Common Crawl. The second column indicates the vocabulary size of the models trained on this data.

2.3. Deduplication and Tokenization

The second step of our pipeline is to remove duplicate lines from the data. We used a very simple method for this, computing the hash of each line, and removing lines with identical hashes (we used the default hash function of java String objects). While this could potentially remove unique lines (which do not have a unique hash), we observed very little collision in practice (since each language is processed independently). Removing duplicates is important for the crawl data, since it contains large amounts of boilerplate, as previously noted by Buck et al. (2014). Overall, 37% of the crawl data is removed by deduplication, while 21% of the Wikipedia data is removed by this operation.

The final step of our preprocessing is to tokenize the raw data. We used the Stanford word segmenter (Chang et al., 2008) for Chinese, Mecab (Kudo, 2005) for Japanese and UETsegmenter (Nguyen and Le, 2016) for Vietnamese. For languages written using the Latin, Cyrillic, Hebrew or Greek scripts, we used the tokenizer from the Europarl preprocessing tools (Koehn, 2005). For the remaining languages, we used the ICU tokenizer. We give statistics for the most common languages in Table 1 and 3.

3. Models

In this section, we briefly describe the two methods that we compare to train our word vectors.

Skipgram. The first model that we consider is the skipgram model with subword information, introduced by Bojanowski et al. (2017). This model, available as part of

the fastText[3] software, is an extension of the skipgram model, where word representations are augmented using character ngrams. A vector representation is associated to each character ngram, and the vector representation of a word is obtained by taking the sum of the vectors of the character ngrams appearing in the word. The full word is always included as part of the character ngrams, so that the model still learns one vector for each word. We refer the reader to Bojanowski et al. (2017) for a more thorough description of this model.

CBOW. The second model that we consider is an extension of the CBOW model (Mikolov et al., 2013b), with position weights and subword information. Similar to the model described in the previous paragraph, this model represents words as bags of character ngrams. The second difference with the original CBOW model is the addition of position dependent weights, in order to better capture positional information. In the CBOW model, the objective is to predict a given word w_0 based on context words $w_{-n}, ..., w_{-1}, w_1, ..., w_n$. A vector representation $\mathbf{h}$ of this context is obtained by averaging the corresponding word vectors:

$$\mathbf{h} = \sum_{\substack{i=-n \\ i \neq 0}}^{n} \mathbf{u}_{w_i}$$

Here, we propose to use the model with position weights introduced by Mnih and Kavukcuoglu (2013). Before taking the sum, each word vector is multiplied (element wise) by a position dependent vector. More formally, the vector representation $\mathbf{h}$ of the context is obtained using:

$$\mathbf{h} = \sum_{\substack{i=-n \\ i \neq 0}}^{n} \mathbf{c}_i \odot \mathbf{u}_{w_i},$$

where $\mathbf{c}_i$ are vectors corresponding to each position in the window, $\odot$ is the element-wise multiplication and $\mathbf{u}_{w_i}$ are the word vectors. We remind the reader that the word vectors $\mathbf{u}_{w_i}$ are themselves sums over the character ngrams. We refer the reader to Mikolov et al. (2017) for a study of the effect of deduplication and model variants (such as position-weighted CBOW) on the quality of the word representations.

4. Evaluations

In this work, we evaluate our word vectors on the word analogy task. Given a triplet of words *A : B :: C*, the goal is to guess the word *D* such that *A : B* and *C : D* share the same relation. An example of such analogy question is *Paris : France :: Berlin : ?*, where the corresponding answer is *Germany*. Word vectors can be evaluated at this task by computing the expected representation of the answer word *D*. Given word vectors x_A, x_B and x_C respectively for words *A*, *B* and *C*, the answer vector can be computed as $x_B - x_A + x_C$. In order to evaluate, the closest word vector x in the dictionary is retrieved (omitting the vectors x_A, x_B and x_C) and the corresponding word is returned. Performance is measured using average accuracy over the whole corpus.

[3] https://fasttext.cc/

4.1. Evaluation Datasets

Analogy datasets are composed of word 4–uplets, of the form *Paris : France :: Rome : Italy*. Such datasets are usually composed of all the possible combinations of pairs such as *Paris : France*, *Berlin : Germany* or *Beijing : China*. In our evaluation, we use the dataset of Svoboda and Brychcin (2016) for Czech, that of Köper et al. (2015) for German, that of Cardellino (2016) for Spanish, that of Venekoski and Vankka (2017) for Finnish, that of Berardi et al. (2015) for Italian, the European variant of the dataset proposed by Hartmann et al. (2017) for Portuguese and that of Jin and Wu (2012) for Chinese.
One of the contributions of this work is the introduction of word analogy datasets for French, Hindi and Polish. To build these datasets, we use the English analogies introduced by Mikolov et al. (2013a) as a starting point. Most of the word pairs are directly translated, and we introduced some modifications, which are specific for each language.

French. We directly translated all the word pairs in the capital-common-countries, capital-world and currency analogies. For family we translated most pairs, but got rid of ambiguous ones (singular and plural for *fils*) or those that translate into nominal phrases. We replaced the city-in-state category by capitals of French *départements*, removing those where either the *département* or capital name is a phrase. We also added a category named antonyms-adjectives composed of antinomic adjectives such as *chaud / froid* (hot and cold). For syntactic analogies, we translated word pairs in all categories, except for comparative and superlative, which in french are trivial: for example *fort*, *plus fort*, *le plus fort* (strong, stronger, strongest). When the word pair was ambiguous we either removed it or replaced with another one. Finally, we added a new past-participle category with pairs such as *pouvoir* and *pu*. In total, this dataset is composed of 31,688 questions.

Hindi. All the word pairs in the categories capital-common-countries, capital-world and currency were translated directly. For the family category, most of the pairs were translated. However, we got rid of word pairs like stepbrother and stepsister which translate into two-word phrases. Also, word-pairs which differentiate in the maternal or paternal origin of the relationship like 'dādā - dādī' (paternal grandparents) and 'nānā - nānī' (maternal grandparents) were added. For the city-in-state category, city-state pairs from India were added, removing pairs in which the city or the state name is a phrase. We had to remove adjective-to-adverb, comparative, superlative, present-participle and past-tense categories as in these cases, we are left with phrases rather than words. We also added a new category adjective-to-noun, where an adjective is mapped to the corresponding abstract noun: for example 'mīṭhā'(sweet)' is mapped to 'miṭhās'(sweetness).

Polish. As for the other languages, we translated all the word pairs in the capital-common-countries, capital-world, currency and family categories.

	CS	DE	ES	FI	FR	HI	IT	PL	PT	ZH	Average
Baseline	63.1	61.0	57.4	35.9	64.2	10.6	56.3	53.4	54.0	60.2	51.0
n-gram 5-5	57.7	61.8	57.5	39.4	65.9	8.3	57.2	54.5	54.8	59.3	50.9
CBOW	63.9	71.7	64.4	42.8	71.6	14.1	66.2	56.0	60.6	51.5	55.5
+negatives	64.8	73.7	65.0	45.0	73.5	14.5	68.0	58.3	62.9	56.0	57.4
+epochs	64.6	73.9	67.1	46.8	74.9	16.1	69.3	58.2	64.7	60.6	58.8
Using Crawl	69.9	72.9	65.4	70.3	73.6	32.1	69.8	67.9	66.7	78.4	66.7

Table 4: Performance of the various word vectors on the word analogy tasks. We restrict the vocabulary for the analogy tasks to the 200,000 most frequent words from the training data.

	CS	DE	ES	FI	FR	HI	IT	PL	PT	ZH
Wikipedia	76.9	79.1	93.9	94.6	88.1	70.8	80.9	69.5	79.2	100.0
Common Crawl	78.6	81.1	90.4	92.2	92.5	70.7	82.6	63.4	75.7	100.0

Table 5: Coverage of models trained on Wikipedia and Common Crawl on the word analogy tasks.

For the `city-in-state` category, we used the capital of Polish regions (*województwo*). For the syntactic analogies, we translated word pairs in all categories except for `plural-verbs`, which we replaced with `verb-aspect`. One example with two aspects is *iść* and *chodzić* which are both imperfective verbs, but the second one expresses an aimless motion. For the `past-tense` category, we use a mixture of perfective and imperfective aspects. Overall, by taking all possible combinations, we come up with 24,570 analogies.

4.2. Model Variants

In all our experiments, we compare our word vectors with the ones obtained by running the `fastText` skipgram model with default parameters – we refer to this variant as "Baseline". Additionally, we perform an ablation study showing the importance of all design choices. We successively add features as follows:

- n-gram 5–5: getting word vectors with character n-grams of length 5 only. By default, the `fastText` library uses all character n-grams from length 3 to 6. One motivation for using fewer n-grams is that the corresponding models are much more efficient to learn.
- CBOW: using the model described in Sec. 3. instead of the skipgram variant from Bojanowski et al. (2017).
- +negatives: using more negative examples. By default, the `fastText` library samples 5 negative examples. Here, we propose to use 10 negatives.
- +epochs: using more epochs to train the models. By default, the `fastText` library trains models for 5 epochs. Here, we propose to train for 10 epochs.
- Using Crawl: instead of only training on Wikipedia, we also use the crawl data. For many languages, this corresponds to a large increase of training data size.

4.3. Results

We evaluate all the model variants on word analogies in ten languages and report the accuracy in Table 4. We restrict the vocabulary for the analogy tasks to the 200,000 most frequent words from the training data. Therefore, the models trained on Wikipedia and Wikipedia+Crawl do not share the exact same vocabulary (see coverage in Table 5).

Influence of models and parameters. First, we observe that on average, all the modifications discussed in Section 4.2. lead to improved accuracy on the word analogy tasks compared to the baseline `fastText` model. First, using character n-grams of size 5, instead of using the default range of 3–6, does not significantly decrease the accuracy (except for Czech). However, using a smaller number of character n-grams leads to faster training, especially when using the CBOW model. Second, we note that using the CBOW model with position weights, described in Section 3., gives the biggest improvement overall. Finally, using more negative examples and more epochs, while making the models slower to train, also leads to significant improvement in accuracy.

Influence of training data. One of the contributions of this work is to train word vectors in multiple languages on large scale noisy data from the web. We now compare the quality of the obtained models to the ones trained on Wikipedia data. Unsurprisingly, we observe that for high resources languages, such as German, Spanish or French, using the crawl data does not increase (or even slightly decreases) the accuracy. This is partly explained by the domain of the analogy datasets, which corresponds well to Wikipedia. However, it is important to keep in mind that the models trained on the crawl data have a larger coverage, and might have better performance on other domains. Second, we observe that for languages with small Wikipedia, such as Finnish or Hindi, using the crawl data leads to great improvement in performance: +23.5 for Finnish, +9.7 for Polish, +16.0 for Hindi, +17.8 for Chinese.

5. Conclusion

In this work, we contribute word vectors trained on Wikipedia and the Common Crawl, as well as three new analogy datasets to evaluate these models, and a fast language identifier which can recognize 176 languages. We study the effect of various hyper parameters on the performance of the trained models, showing how to obtain high quality word vectors. We also show that using the common crawl data, while being noisy, can lead to models with larger coverage, and better models for languages with small Wikipedia. Finally, we observe that for low resource languages, such as Hindi, the quality of the obtained word vectors is much lower than for other languages. As future work, we would like to explore more techniques to improve the quality of models for such languages.

6. Bibliographical References

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual nlp. *Proc. CoNLL*.

Baldwin, T. and Lui, M. (2010). Language identification: The long and the short of the matter. In *Proc. NAACL*.

Berardi, G., Esuli, A., and Marcheggiani, D. (2015). Word embeddings go to Italy: a comparison of models and training datasets. *Italian Information Retrieval Workshop*.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5.

Buck, C., Heafield, K., and Van Ooyen, B. (2014). N-gram counts and language models from the common crawl. In *Proc. LREC*, volume 2.

Cardellino, C. (2016). Spanish Billion Words Corpus and Embeddings, March.

Chang, P.-C., Galley, M., and Manning, C. D. (2008). Optimizing chinese word segmentation for machine translation performance. In *Proceedings of the third workshop on statistical machine translation*.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proc. ICML*.

Hartmann, N., Fonseca, E., Shulby, C., Treviso, M., Rodrigues, J., and Aluisio, S. (2017). Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *arXiv preprint arXiv:1708.06025*.

Jin, P. and Wu, Y. (2012). Semeval-2012 task 4: evaluating chinese word similarity. In *Proc. *SEM*.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2017). Bag of tricks for efficient text classification. In *Proc. EACL*.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5.

Köper, M., Scheible, C., and im Walde, S. S. (2015). Multilingual reliability and "semantic" structure of continuous word spaces. *Proc. IWCS 2015*.

Kudo, T. (2005). Mecab: Yet another part-of-speech and morphological analyzer. *http://mecab. sourceforge. net/*.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proc. ACL (system demonstrations)*.

Mihalcea, R. (2007). Using wikipedia for automatic word sense disambiguation. In *Proc. NAACL*.

Mikolov, T., Chen, K., Corrado, G. D., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Adv. NIPS*.

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2017). Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*.

Mnih, A. and Kavukcuoglu, K. (2013). Learning word embeddings efficiently with noise-contrastive estimation. In *Adv. NIPS*.

Nguyen, T.-P. and Le, A.-C. (2016). A hybrid approach to vietnamese word segmentation. In *Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2016 IEEE RIVF International Conference on*. IEEE.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proc. EMNLP*.

Svoboda, L. and Brychcin, T. (2016). New word analogy corpus for exploring embeddings of Czech words. In *Proc. CICLING*.

Venekoski, V. and Vankka, J. (2017). Finnish resources for evaluating language model semantics. In *Proc. NoDaLiDa*.

Wu, F. and Weld, D. S. (2010). Open information extraction using wikipedia. In *Proc. ACL*.

SumeCzech: Large Czech News-Based Summarization Dataset

Milan Straka, Nikita Mediankin, Tom Kocmi, Zdeněk Žabokrtský, Vojtěch Hudeček, Jan Hajič
Institute of Formal and Applied Linguistics
Charles University, Faculty of Mathematics and Physics
Malostranské náměstí 25, Prague, Czech Republic
{straka,mediankin,kocmi,zabokrtsky,hudecek,hajic}@ufal.mff.cuni.cz

Abstract

Document summarization is a well-studied NLP task. With the emergence of artificial neural network models, the summarization performance is increasing, as are the requirements on training data. However, only a few datasets are available for Czech, none of them particularly large. Additionally, summarization has been evaluated predominantly on English, with the commonly used ROUGE metric being English-specific. In this paper, we try to address both issues. We present SumeCzech, a Czech news-based summarization dataset. It contains more than a million documents, each consisting of a headline, a several sentences long abstract and a full text. The dataset can be downloaded using the provided scripts available at http://hdl.handle.net/11234/1-2615. We evaluate several summarization baselines on the dataset, including a strong abstractive approach based on Transformer neural network architecture. The evaluation is performed using a language-agnostic variant of ROUGE.

Keywords: SumeCzech, summarization dataset, document summarization, ROUGE, Czech

1. Introduction

Similarly to many other NLP tasks, performance of automatic document summarization has been improving with the recent rise of neural network methods. While deep neural network models can leverage large datasets, only a few moderately-sized datasets are available for document summarization when compared to, e.g., machine translation.

Additionally, document summarization has been explored mostly on English, with the dominant ROUGE metric (Lin, 2004) being English-specific (utilizing English stemmer, stop words and synonyms).

In order to provide more data for document summarization in Czech, this paper introduces SumeCzech – a collection of one million Czech news articles, each consisting of a headline, a several sentence abstract and a full text. The documents originate from five Czech Internet news sites. The dataset can be downloaded using our provided scripts.

Headline-abstract-text structure of the documents allows the dataset to be used for multiple summarization setups: headline generation either from an abstract or a full text, or generation of a multi-sentence abstract from a full text.

To enable automatic evaluation of summarization for Czech, we also propose a straightforward language-agnostic variant of the ROUGE metric, which we call $\text{ROUGE}_{\text{RAW}}$.

We evaluate several baselines for all selected summarization settings. Apart from several unsupervised methods, we evaluate two supervised methods: an extractive one inspired by approach by Kupiec et al. (1995), and an abstractive baseline based on Transformers neural network architecture (Vaswani et al., 2017).

2. Related Work

2.1. Datasets

Sentence summarization has been traditionally connected with the task of headline generation. The task was standardized around the DUC-2003 and DUC-2004 competitions (Over et al., 2007), which provided a standard evaluation set consisting of 500 news articles from New York Times and Associated Press Wire, each paired with 4 different human-generated reference summaries. For training, the Gigaword dataset (Graff et al., 2003) has been used frequently, offering 4 million news articles including their headlines.

Recently, Nallapati et al. (2016a) modified the CNN/Daily Mail corpus constructed by Hermann et al. (2015) to serve for multi-sentence summarization. The corpus consists of approximately 300 000 documents. Additionally, Filippova and Altun (2013) proposed a method for constructing datasets for extractive sentence summarization.[1]

To our best knowledge, only small summarization datasets exist for Czech: Czech part of the MultiLing dataset (Giannakopoulos et al., 2015; Li et al., 2013; Elhadad et al., 2013) containing 40 Wikipedia articles, and SummEC (Rott and Červa, 2013) containing 50 news articles.

2.2. Metrics

ROUGE (Lin, 2004) is the most commonly used metric, proposed as an English-specific recall-based metric. It utilizes English stemmer, stop words and synonyms.

Recently, the METEOR metric (Denkowski and Lavie, 2014) has been used by See et al. (2017) to evaluate multi-sentence summarization.

2.3. Summarization Methods

Summarization methods are generally either *extractive* or *abstractive*. Extractive methods only select suitable parts (sentences, words or phrases) from the document, while abstractive methods can produce an arbitrary text as the summary.

The extractive summarization methods are typically unsupervised, for example Luhn (Luhn, 1958), Latent Se-

[1] The dataset has been recently released at https://github.com/google-research-datasets/sentence-compression.

mantic Analysis (Steinberger and Ježek, 2004), LexRank (Erkan and Radev, 2004), TextRank (Mihalcea and Tarau, 2004), SumBasic (Vanderwende et al., 2007) or KL-Sum (Haghighi and Vanderwende, 2009). However, very good results in extractive summarization were achieved recently with recurrent neural networks (Filippova et al., 2015; Filippova and Alfonseca, 2015; Nallapati et al., 2016b; Nallapati et al., 2017).

Abstractive approach relies predominantly on the machine translation paradigm, also boosted by the recent success of neural machine translation (Rush et al., 2015; Nallapati et al., 2016a; Gülçehre et al., 2016; See et al., 2017).

3. The Dataset

3.1. Choice of Data Sources

When designing the dataset, we considered two main requirements. First, and most importantly, we wanted to produce a dataset that would be sufficiently large for deep learning methods to be applicable to it. However, we possessed limited human and time resources making it impossible to accomplish this task by creating summaries manually. This implied an automatic or a semi-automatic method of collecting the data, facilitating the need for a data source consisting of documents that would already have some kind of easily identifiable human-produced summary. Second, we wanted the data to be more or less domain-neutral, i.e., without much domain-specific terminology.

Collecting a dataset of scientific articles using their abstracts as summaries was considered, but promptly rejected. The next choice was to use electronic newspapers as they seemed to be able to provide a reasonable amount of data with reasonably well separated short abstracts preceding the articles.

The raw data for the dataset was collected from the Common Crawl project[2] using the Common Crawl API. Initially, five Czech news websites were selected to create the dataset: `novinky.cz`, `lidovky.cz`, `denik.cz`, `idnes.cz`, and `ihned.cz`. However, during the cleanup of the data, we decided to drop `ihned.cz` from the dataset, because too many of its pages turned out to be just abridged versions of the actual articles with links to paid content. Instead, `ceskenoviny.cz`, which provides mostly high-quality articles, was added to the collection.

3.2. Data Preparation

The data was prepared in the following steps:

1. Dumps of the relevant websites' pages from 10 Common Crawl collections were downloaded.
2. Irrelevant entries such as advertisement pages, article listings and photo galleries were filtered out based on a set of simple heuristics.
3. From each seemingly relevant entry, its headline, abstract and full text were extracted based on the HTML structure of the webpage, cleaned from HTML markup, embedded javascript and irrelevant information such as:
   - advertisement links;
   - links to other news;
   - leftover captions of embedded photo and video materials;
   - low-level headers embedded in the text, which are used as paragraph titles in some texts but should be removed because they are not really part of the text.
4. Frequently seen leading tags such as FOTO, VIDEO, country, city were removed from headlines and abstracts. These tags were usually separated from the rest of the headline or abstract by a dash or a colon (e.g., "Praha: ..."). For the purpose of cleaning these up, lists of most frequent tokens seen at the start of headlines and abstracts before dash or colon were created and manually checked. Names of persons with the following colon (indicating direct speech) were deliberately left in place.
5. The following documents were dropped:
   - with empty headline;
   - with abstract shorter than 10 words;
   - with full text shorter than 100 words;
   - with text-to-abstract ratio less than 4.
6. Language recognition was performed with langdetect,[3] Python port of Google's language-detection library,[4] and non-Czech documents were dropped.
7. A number of documents was dropped based on the headline and/or abstract text (e.g., some headlines clearly indicated that the page is an advertisement, not a news article, some abstracts were disclaimers that the page belongs to a series of culinary recipes with no other information in the abstract).
8. A number of documents was dropped based on the presence of certain keywords in the headline or abstract, e.g., some abstracts were starting with the word 'aktualizováno' ('updated'), a metainformation not directly connected with the content of the article that could not be reliably removed.
9. From the sets of documents with either duplicate headlines, duplicate abstracts or duplicate texts, only one document was retained. Therefore, headlines in the dataset are unique, as well as abstracts and texts.
10. Some inexact news duplicates were filtered out based on several heuristics. Specifically for `denik.cz`, all regional pages were dropped based on their URLs, since they were mostly either reprints of central news or very specific entries such as "Where to play football this weekend".
11. Date of each article's publication was extracted wherever possible either from the page's metadata or from its body based on HTML markup. All dates were then converted into standardized format.

3.3. Structure of Dataset Entries

The dataset is produced in the JSON Lines format,[5] where each document is represented on a single line as a JSON object with the following fields:

[2] `http://commoncrawl.org`

[3] `https://pypi.python.org/pypi/langdetect`

[4] `https://github.com/shuyo/language-detection`

[5] `http://jsonlines.org`

Website	Documents	
	Number	Percentage
ceskenoviny.cz	4 854	0.5%
denik.cz	157 581	15.7%
idnes.cz	463 192	46.2%
lidovky.cz	136 899	13.7%
novinky.cz	239 067	23.9%
Total	1 001 593	

Table 1: Number of documents from individual websites.

- `url`: the URL of the article webpage from where it was crawled by the Common Crawl;
- `headline`;
- `abstract`;
- `text`;
- `subdomain`: some of the source websites have clear-cut subdomains for different broad topics, e.g., `lidovky.cz` has `sport.lidovky.cz` for sport news; these were extracted from article URLs for possible future use as a surrogate means of identificating a human-assigned article topic;
- `section`: another option for topic identification: sometimes, a broad topic can be extracted from the part of the URL that follows the domain name part;
- `published`: date of publication in RFC 3339 format,[6] with all dates stored in CET and CEST as appropriate (i.e., utilizing the timezone in which the article was published).

Headlines and abstracts are stored without any line breaks. The former mostly did not have them originally, while the latter either had none or had each sentence separated by a line break, depending on the website formatting, making line breaks in abstracts non-indicative. Line breaks in full text are used as the means to separate the paragraphs of the original text.

We put the emphasis on maximum human-readability of the resulting data without sacrificing the ease of processing. Both the former and the latter were the reasons behind choosing JSON Lines format. First, Czech uses significant amount of accented characters, therefore it was important to be able to save the data in UTF-8 character encoding as is, i.e., without escaping non-ASCII characters, which is permitted in JSON Lines. Second, we wanted to keep individual entries contained within single lines to facilitate the ease of use of the dataset with Unix-style text processing tools.

3.4. Dataset Size Statistics

In total, the dataset contains approximately one million documents, with the distribution across websites shown in Table 1.

The quantitative statistics of headline, abstract and full text length are displayed in Table 2. The headlines are approximately 9 words long on average, with the abstracts being nearly five times the size and the full texts being nearly ten times the size of abstracts.

[6] `https://www.ietf.org/rfc/rfc3339.txt`

	Q1	Median	Q3	Mean	Stddev
Headlines	7	9	11	9.4	2.9
Abstracts	33	42	51	42.2	14.8
Texts	265	378	553	470.1	365.3

Table 2: Quantitative statistics of lengths of headlines, abstracts and texts in words. Q1 and Q3 denote the first and the third quartile, respectively.

3.5. Dataset Split

Before splitting the data into train, dev and test sets, we theorized that having too similar documents in the train and the test sets could lead to a skewed (too optimistic) evaluation of any supervised summarization methods. Therefore, we wanted the documents that are close to each other in some sense to be put into the same part of the split. At the same time, we did not want to end up with all the documents from one domain in the same part of the split, as it would introduce even stronger bias to the evaluation. To elaborate, this can be imagined as a situation when a model is trained on the data from one domain and then evaluated on the data from another. However, it appeared to us that the possibility of evaluation on an out-of-domain test set would be an interesting option. This, again, can be thought of as a common real-life situation when a model is trained on the data from one domain, then used on real data from other domain. In this use case having an out-of-domain test set could provide some insight into the model's possible behavior on real-world data.

Taking into account the above considerations, we devised the following procedure. The documents were first clustered into 25 clusters by K-Means algorithm, based on normalized L2 similarity of their abstracts. A cluster of size approximately 4.5% of the whole dataset size was taken as the out-of-domain test set. The rest of the data was then clustered again into 5000 clusters by K-Means algorithm, again based on L2 similarity of their abstracts. Consequently, the clusters were randomly divided in rougly 86.5:4.5:4.5 ratio to form the standard train/dev/test split.

The sizes of the individual dataset parts, along with distribution of articles across websites in each part, are presented in Table 3.

When inspected, the out-of-domain test set turned out to contain news about concerts and festivals, which is indeed out of domain when related to other topics, albeit not radically, because it is still news articles.

4. Obtaining the SumeCzech Dataset

Instead of distributing the produced dataset, we provide the two components for an end user to recreate it: the document listings and the extractor script.

The document listings contain the following values for each documents of the dataset:

- name of the Common Crawl file that contains the raw data for the document;
- its offset in the Common Crawl file;
- its length in the Common Crawl file;

Website	Documents	
	Number	Percentage
train		
ceskenoviny.cz	4318	0.5%
denik.cz	137926	15.9%
idnes.cz	404367	46.6%
lidovky.cz	118761	13.7%
novinky.cz	202224	23.3%
Total	867596	
dev		
ceskenoviny.cz	229	0.5%
denik.cz	7559	17.0%
idnes.cz	21163	47.5%
lidovky.cz	5755	12.9%
novinky.cz	9861	22.1%
Total	44567	
test		
ceskenoviny.cz	168	0.4%
denik.cz	6854	15.4%
idnes.cz	19960	44.9%
lidovky.cz	6462	14.5%
novinky.cz	11010	24.8%
Total	44454	
out-of-domain test		
ceskenoviny.cz	139	0.3%
denik.cz	5242	11.7%
idnes.cz	17702	39.4%
lidovky.cz	5921	13.2%
novinky.cz	15972	35.5%
Total	44976	

Table 3: The train/dev/test/out-of-domain test split of SumeCzech.

- which set (train/dev/test/out-of-domain test) this document belongs to;
- MD5 sum of the corresponding entry in the dataset.

The first three values deterministically define the place of the raw data for the document in the Common Crawl data and allow for its retrieval via Common Crawl API. The last value allows to check if the extraction procedure have successfully recreated the document from the raw data.

The extractor script is written in Python 3 and recreates the dataset using the document listings and the Common Crawl data by downloading the raw data and applying the original steps described in 3.2. that are required to extract headlines, abstracts, full texts and metadata and clean them up (but not the steps involved in filtering out undesirable documents, because those documents are already absent from the listings). The script then checks each recreated entry against the corresponding MD5 sum provided in the listings.

The document listings and the extraction script are available for download at `http://hdl.handle.net/11234/1-2615` under Mozilla Public License 2.0.[7]

We do not impose any additional licensing restrictions on the recreated dataset, however, it is subject to the Common Crawl terms of use,[8] and, by extension, local legislations regulating authors' rights that are in effect in the end user's country.

5. Evaluation Metrics

A standard way to evaluate summarization task is to use the ROUGE metric (Lin, 2004). ROUGE is an English-specific metric (employing English stemmer, stop words and synonyms), and was originally recall-based. In the DUC task, both the gold summary and the system summary is capped at 75 bytes and the recall of the non-stop words is evaluated, taking synonyms into account.

However, with the appearance of other datasets and more powerful abstractive methods, a fixed limit on the summary length became neither desirable nor needed, and, therefore, full-length F1 ROUGE is also being used recently (Nallapati et al., 2016a; Chopra et al., 2016; See et al., 2017).

Therefore, we propose to evaluate summarization methods trained on the SumeCzech dataset using full-length F1-score of a language-agnostic variant of ROUGE, which utilizes no stemmer, no stop words and no synonyms. We denote this variant $\text{ROUGE}_{\text{RAW}}$ and report $\text{ROUGE}_{\text{RAW}}$-1 (unigrams), $\text{ROUGE}_{\text{RAW}}$-2 (bigrams) and $\text{ROUGE}_{\text{RAW}}$-L (longest common subsequence). The Python 3 implementation of language-agnostic $\text{ROUGE}_{\text{RAW}}$ is provided alongside the SumeCzech dataset.

6. Experiments

The dataset allows for three summarization task setups:

- abstract→headline: generate one-sentence summary using a paragraph of approximately 3 sentences; similar to the DUC (Over et al., 2007) and Gigaword (Graff et al., 2003) tasks;
- full text→headline: generate one-sentence summary using a full text of several dozen sentences; also similar to the DUC (Over et al., 2007) and Gigaword (Graff et al., 2003) tasks;
- full text→abstract: generate multi-sentence summary using a full text consisting of several dozen sentences; similar to the CNN/Daily Mail (Nallapati et al., 2016a) task.

6.1. Extractive Methods

6.1.1. Unsupervised

We evaluate several unsupervised extractive methods for all three summarization setups. All methods extract either 1 or 3 sentences, depending on whether they are generating a headline or an abstract, respectively. We employed the following methods:

- `first`: return given number (1 or 3) of initial sentences. Such baseline, while seemingly trivial, usually achieves high performance on news articles and is very hard to beat, because authors tend to summarize the most prominent information in the first few sentences.
- `random`: return randomly chosen sentences.
- `textrank`: TextRank (Mihalcea and Tarau, 2004), a classic unsupervised method based on the representation of the text as a network of sentences based on their similarity.

[7] `http://www.mozilla.org/MPL/2.0/`

[8] `http://commoncrawl.org/terms-of-use/full`

For the above methods, we use our own Python 3 implementation. TextRank utilizes a list of Czech stop words for the purposes of calculating sentence similarity.

6.1.2. Supervised

In order to evaluate supervised approach, we include an extractive machine learning method inspired by the work of Kupiec et al. (1995). In this method, we first transform each sentence to a vector of features that are listed below:

- `TF-IDF` (Ramos and others, 2003): sum of `TF-IDF` measured for each word normalized by the sentence length. In the inference phase, we rely on the frequency values obtained during training.
- `Length`: length of the sentence.
- `Cohesion`: total distance from the sentence to the other ones in terms of edit distance.
- `Proper names`: count of capitalized words in the sentence.
- `Numbers`: count of tokens that consist of digits.
- `Non-essential words`: count of common words that indicates that the sentence relates to some other one.

In the training phase, the vectors are labeled by binary values. First, the sentences are sorted based on their similarity to the sentences from the gold abstract (or headline, respectively). Then, top sentences are picked and corresponding feature vectors are marked positive, the rest is considered negative. This way we obtain a classification task and we can train a classifier. We consider two classification algorithms: logistic regression and random forests. In the inference phase, the sentences are transformed into vectors again, and the classifier assigns each one the probability of being picked. Finally, a fixed number of sentences with the best scores is picked.

Depending on the employed classifier, the method is dubbed either `clf-lr` (when classifier used is logistic regression) or `clf-rf` (when random forests are employed).

6.2. Abstractive Summarization

Following the recent success in abstractive summarization (See et al., 2017), we also evaluated an abstractive summarization method. We utilized the tensor2tensor framework,[9] namely version 1.2.9. We used a neural machine translation model of Vaswani et al. (2017) with hyperparameters set as in model called `base` in the paper.[10] We evaluated the abstractive summarization method, dubbed `t2t`, on all three tasks.

We trained the model on the lowercased data and vocabulary of 32 000 word-pieces (Wu et al., 2016). We utilized GeForce GTX 1080 Ti GPU for training. The batch sizes differed for each task, batch size of 1700 was used for abstract→headline, batch size of 6500 for text→abstract and batch size of 7500 for text→headline. The final models utilize averaging over last 8 consecutive checkpoints (one hour from each other). For the abstract→headline task, we trained the model for 15 days and for the final evaluation we use beam size 4. The tasks text→headline/abstract were trained for 8 days, use beam size 3 and clip all inputs to maximal length of 400 words in order to fit in GPU memory.

[9] `https://github.com/tensorflow/tensor2tensor`

[10] The `big` model as described in the paper exhibited worse results, possibly due to a small maximal batch size.

Method	$ROUGE_{RAW}$-1			$ROUGE_{RAW}$-2			$ROUGE_{RAW}$-L		
	P	R	F	P	R	F	P	R	F
test									
`first`	13.3	22.9	15.9	4.3	7.6	**5.2**	11.9	20.5	14.3
`random`	10.0	16.6	11.6	2.7	4.7	3.2	9.0	14.8	10.4
`textrank`	12.9	22.4	15.5	4.1	7.3	4.9	11.6	20.0	13.9
`clf-lr`	11.5	**29.6**	15.9	3.4	**9.3**	4.7	9.8	**25.4**	13.7
`t2t`	**19.3**	15.4	**16.6**	**6.2**	4.8	**5.2**	**17.9**	14.3	**15.4**
out-of-domain test									
`first`	13.5	25.1	**16.6**	4.8	9.3	**5.9**	12.1	22.4	14.8
`random`	10.2	18.7	12.4	3.1	6.2	3.9	9.2	16.7	11.1
`textrank`	13.2	24.7	16.2	4.6	9.0	5.7	11.8	21.9	14.5
`clf-lr`	11.5	**28.6**	15.3	3.9	**10.7**	5.4	9.9	**24.5**	13.1
`t2t`	**18.9**	14.8	16.0	**6.8**	5.0	5.5	**17.7**	13.9	**15.0**

Table 4: Abstract→headline summarization results.

6.3. Results and Discussion

We evaluated the above extractive and abstractive methods on both the test and out-of-domain test portions of SumeCzech, utilizing the $ROUGE_{RAW}$-1, $ROUGE_{RAW}$-2 and $ROUGE_{RAW}$-L metrics. To allow for more detailed interpretation of the results, we present not only F1-score, but also precision and recall.

Before we present the results, it is worth mentioning that the `first` baseline is usually very difficult to overcome, especially in the domain of news articles (Nallapati et al., 2016a; See et al., 2017).

First, we present the evaluation of extractive and abstractive methods in the abstract→headline setting in Table 4. The extractive methods perform similarly to `first` baseline, but the `first` baseline has slightly higher F-scores. The abstractive `t2t` method performs the best, achieving the highest F-scores in all three $ROUGE_{RAW}$ variants.

Note that the abstractive method has very high precision, but lacks in recall. We found out that this is a consequence of generating too short headlines. While the gold headlines have an average length of 9.7 words, the headlines generated by the `t2t` method consist of 7.7 words on average. We therefore conclude that a higher performance could be achieved by better matching the length distribution of the headlines.

On the out-of-domain test set, the results of the `t2t` method are lower relative to the performance of other algorithms. Notably, the F-score of the `first` baseline is the highest for $ROUGE_{RAW}$-1 and $ROUGE_{RAW}$-2 metrics, while being only slightly behind the best $ROUGE_{RAW}$-L F-score, which was achieved by `t2t`. We hypothesise that this drop is caused by the `t2t` method not being able to generalize well enough for the out-of-domain test set.

The results of summarization of full texts into headlines are presented in Table 5. Both supervised algorithms `clf-rl` and `t2t` demonstrate lower F-score performance than the unsupervised `first` and `textrank` methods. However, the precision of `t2t` approach still surpasses all other meth-

Method	ROUGE$_{RAW}$-1			ROUGE$_{RAW}$-2			ROUGE$_{RAW}$-L		
	P	R	F	P	R	F	P	R	F
test									
first	6.3	11.8	**7.6**	**1.1**	2.2	**1.4**	5.7	10.6	**6.8**
random	4.3	8.0	5.2	0.5	1.0	0.6	4.0	7.2	4.7
textrank	5.6	**15.3**	**7.6**	0.9	**2.6**	1.2	4.9	**13.3**	6.6
clf-rf	5.0	9.4	6.3	0.7	1.3	0.8	4.5	8.4	5.6
t2t	**7.4**	5.9	6.4	0.7	0.5	0.6	**7.0**	5.6	6.0
out-of-domain test									
first	**6.2**	12.2	7.6	**1.3**	2.6	**1.6**	**5.6**	10.9	6.8
random	4.3	8.3	5.2	0.6	1.2	0.7	3.9	7.5	4.7
textrank	5.7	**15.9**	**7.8**	1.1	**3.3**	1.5	5.0	**13.9**	**6.9**
clf-rf	5.3	10.1	6.7	1.0	2.1	1.3	4.9	9.3	6.1
t2t	5.3	4.3	4.6	0.4	0.3	0.4	5.0	4.1	4.3

Table 5: Text→headline summarization results.

Method	ROUGE$_{RAW}$-1			ROUGE$_{RAW}$-2			ROUGE$_{RAW}$-L		
	P	R	F	P	R	F	P	R	F
test									
first	**13.3**	18.4	**14.6**	**2.3**	3.4	**2.6**	8.9	12.4	**9.8**
random	11.6	15.5	12.5	1.4	2.1	1.6	7.7	10.4	8.3
textrank	11.6	21.5	14.3	1.9	3.8	2.4	7.6	14.1	9.3
clf-rf	10.5	**23.3**	13.8	1.6	**3.9**	2.2	6.7	**15.0**	8.8
t2t	12.2	9.4	10.2	1.1	0.8	0.9	**9.6**	7.4	8.0
out-of-domain test									
first	**12.2**	18.1	**13.8**	**2.1**	3.4	**2.5**	8.3	12.4	**9.3**
random	10.7	15.4	11.9	1.4	2.2	1.6	7.3	10.5	8.1
textrank	11.0	**21.2**	13.7	2.0	**4.0**	**2.5**	7.3	**14.2**	9.1
clf-rf	9.1	20.2	11.9	1.4	3.3	1.8	6.3	13.5	7.9
t2t	11.7	8.4	9.3	0.8	0.6	0.7	**9.6**	7.0	7.7

Table 6: Text→abstract summarization results.

ods in two ROUGE variants.

Similarly to the previous settings, the performance of t2t deteriorates on the out-of-domain test set, while other methods are mostly unaffected.

The last considered setup of text→abstract summarization is evaluated in Table 6, yielding results similar to the previous setup. The first baseline is performing the best, followed by the textrank approach. The relative performance of the t2t abstractive summarization is the lowest, being inferior even to the random baseline on both test and out-of-domain test sets.

In order to compare quality of documents from different websites, we also analyse the first baseline in the abstract→headline setup for every website separately. The results are presented in Table 7. The ROUGE$_{RAW}$ metric shows that all websites provide headlines of similar quality, with the exception of ceskenoviny.cz, which provides headlines that are much more similar to the first sentences of their articles' abstracts.

6.4. Examples

We illustrate three test set examples of first and t2t baselines in abstract→headline setup in Figure 1. In order to make the examples accessible to non-Czech speaking audience, we translated the examples to English, preserving the original phrase structure and vocabulary as much as possible.

In all examples, the first method produces a good summary, even though quite large. The t2f method generates fluent summaries of suitable length, but while in the first case the headline is identical to the gold one, in the second case it is slightly paraphrased, and in the third case the produced headline uses completely different words than the gold one. Even while the headline produced by the t2t method is of high quality in all three cases, it receives lower ROUGE$_{RAW}$ score in the second case and zero score in the third case.

Website	ROUGE$_{RAW}$-1 F-score	ROUGE$_{RAW}$-2 F-score	ROUGE$_{RAW}$-L F-score
test			
ceskenoviny.cz	29.8	13.8	27.9
denik.cz	16.6	6.1	15.1
idnes.cz	14.2	4.2	12.6
lidovky.cz	16.5	5.3	14.8
novinky.cz	18.2	6.1	16.2
All websites	15.9	5.2	14.3
out-of-domain test			
ceskenoviny.cz	30.7	14.4	27.6
denik.cz	16.2	6.1	14.6
idnes.cz	14.8	4.8	13.1
lidovky.cz	17.8	6.3	15.7
novinky.cz	18.7	7.2	16.8
All websites	16.6	5.9	14.8

Table 7: The first baseline for abstract→headline task computed per website.

7. Conclusions

We have presented SumeCzech, a new large news summarization dataset for Czech. Every document in the dataset is composed of a short headline, an abstract comprising a few sentences, and a full text, allowing for several summarization setups. The scripts for downloading the dataset are available at http://hdl.handle.net/11234/1-2615.

We use language-agnostic variant of ROUGE metric ROUGE$_{RAW}$ for evaluation.

Finally, we have evaluated several baseline extractive summarization methods, both unsupervised and supervised, as well as an abstractive method based on neural machine translation Transformer architecture with subword units (Vaswani et al., 2017).

Acknowledgments

This work has been partially supported by grants GAUK 1114217/2017, GAUK 8502/2016, and SVV 260 453 grant of Charles University.

This work has also been partially supported by, and data has been stored into, the LINDAT/CLARIN repository, a large research infrastructure supported by the Ministry of Education, Youth and Sports of the Czech Republic under projects LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781.

Method	Headline
gold	Žalobce navrhl pro Sisáka a Halu vazbu *The prosecutor proposed remand for Sisak and Hal*
first	Státní zástupce Adam Borgula navrhl poslat finančníka Petra Sisáka a jeho pravou ruku, advokáta Iva Halu, do vazby. *State attorney Adam Borgula proposed to send financier Peter Sisak and his right hand, lawyer Iva Hal, to remand.*
t2t	Žalobce navrhl pro sisáka a halu vazbu *The prosecutor proposed remand for sisak and hal*
gold	Sněmovna dala šanci úplnému zákazu kouření v restauracích *The parliament gave a chance to a complete smoking ban in restaurants*
first	Sněmovna dala šanci v dnešním úvodním kole úplnému zákazu kouření cigaret v restauracích, barech, vinárnách nebo v kavárnách a čajovnách. *In today's opening round, the parliament gave a chance to a complete smoking ban of cigarettes in restaurants, bars, wine bars, cafes and tearooms.*
t2t	Poslanci dali šanci zákazu kouření v restauracích *The deputies gave a chance to a smoking ban in restaurants*
gold	Rumunsko přijme prvky amerického raketového štítu *Romania will accept elements of American rocket shield*
first	Rumunská nejvyšší rada obrany (CSAT) ve čtvrtek schválila plán Spojených států rozmístit v Rumunsku pozemní prvky nového systému protiraketové obrany. *On Thursday, the Romanian Supreme Defense Council (CSAT) approved the United States' plan to distribute in Romania the ground elements of the new anti-missile defense system.*
t2t	Rumuni schválili nový protiraketový systém *The Romanians approved a new anti-missile system*

Figure 1: Examples of `first` and `t2t` methods in the abstract→headline setup taken from the test set. The English translations are in italics.

8. Bibliographical References

Chopra, S., Auli, M., and Rush, A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 93–98. The Association for Computational Linguistics.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *EACL 2014 Workshop on Statistical Machine Translation*.

Elhadad, M., Miranda-Jiménez, S., Steinberger, J., and Giannakopoulos, G. (2013). Multi-document multilingual summarization corpus preparation, part 2: Czech, hebrew and spanish. *MultiLing 2013*, page 13.

Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Filippova, K. and Alfonseca, E. (2015). Fast k-best sentence compression. *CoRR*, abs/1510.08418.

Filippova, K. and Altun, Y. (2013). Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1481–1491. The Association for Computational Linguistics.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence Compression by Deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 360–368. The Association for Computational Linguistics.

Giannakopoulos, G., Kubina, J., Conroy, J. M., Steinberger, J., Favre, B., Kabadjov, M. A., Kruschwitz, U., and Poesio, M. (2015). Multiling 2015: Multilingual summarization of single and multi-documents, on-line fora, and call-center conversations. In *SIGDIAL Conference*, pages 270–274.

Graff, D., Kong, J., Chen, K., and Maeda, K. (2003). English Gigaword. Philadelphia. Linguistic Data Consortium.

Gülçehre, Ç., Ahn, S., Nallapati, R., Zhou, B., and Bengio, Y. (2016). Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Haghighi, A. and Vanderwende, L. (2009). Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics.

Hermann, K. M., Kociský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM.

Li, L., Forascu, C., El-Haj, M., and Giannakopoulos, G. (2013). Multi-document multilingual summarization corpus preparation, part 1: Arabic, English, Greek, Chinese, Romanian. Association for Computational Linguistics.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *In Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–

81.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text. In *EMNLP*, volume 4, pages 404–411.

Nallapati, R., Zhou, B., dos Santos, C. N., Gülçehre, Ç., and Xiang, B. (2016a). Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290. The Association for Computer Linguistics.

Nallapati, R., Zhou, B., and Ma, M. (2016b). Classify or select: Neural architectures for extractive document summarization. *CoRR*, abs/1611.04244.

Nallapati, R., Zhai, F., and Zhou, B. (2017). Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3075–3081. AAAI Press.

Over, P., Dang, H., and Harman, D. (2007). DUC in Context. *Inf. Process. Manage.*, 43(6):1506–1520, November.

Ramos, J. et al. (2003). Using TF-IDF to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 133–142.

Rott, M. and Červa, P. (2013). SummEC: A Summarization Engine for Czech. In *International Conference on Text, Speech and Dialogue*, pages 527–535. Springer.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389. The Association for Computational Linguistics.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics.

Steinberger, J. and Ježek, K. (2004). Using latent semantic analysis in text summarization and summary evaluation.

Vanderwende, L., Suzuki, H., Brockett, C., and Nenkova, A. (2007). Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

A Diachronic Corpus for Literary Style Analysis

Carmen Klaussner, Carl Vogel
Computational Linguistics Group, Trinity Centre for Computing and Language
School of Computer Science and Statistics
Trinity College Dublin
klaussnc@tcd.ie, vogel@tcd.ie

Abstract

This research presents a resource for diachronic style analysis in particular the analysis of literary authors over time. Temporal style analysis has received comparatively little attention over the past years in spite of the possibility of an author's style frequently changing over time, a property that is not only interesting in its own right, but which also has implications for synchronic analyses of style. The corpus contains 22 American literary authors from the mid 19th to early 20th century that wrote largely in parallel. After describing the resource, we show how the corpus can be used to detect changing features in literary style.

Keywords: Diachronic literary corpus, style analysis, linear regression

1. Introduction

The analysis of the authorial *fingerprint* through computational means, referred to as stylometry, is to be distinguished from manual analyses relying on the verdict of literary scholars particularly acquainted with the author(s) in question. A matter complicating this type of analysis is the fact that an author's style only takes shape through the comparison with other contemporaneous authors, the exact selection of which determines how close one comes to the "actual" fingerprint. For instance, if Mark Twain uses the word *'but'* more frequently than other conjunctions then this is only *interesting* or useful in telling his style apart from others if comparable American authors of his time use the same feature at different rates.

Additionally, one may consider this type of analysis to hold an intrinsic flaw: despite the fact that most authors compose their published writings over a period of 20-40 years, this diachronic property is not widely taken into account. This neglect poses an issue for synchronic style analyses in at least one way. Unless style is found to be invariant for an author and does not change with age and experience, temporality can be a confounding factor in stylometry and authorship attribution (Daelemans, 2013). In the best case, the (synchronic) stylistic analysis might merely select those elements *stable* over time, as this renders them also stable and consistent over the author's corpus, heedlessly discarding those elements that show temporal variation and incidentally possibly style *development*. In the worst case, all features considered are affected by time and the most stable features over the corpus are too variable to discriminate well.

Analysis of diachronic elements of style requires accurately time-stamped data, i.e. either reflecting the time of composition or publication year. This paper describes the development of a parallel literary corpus that allows for comparison among authors as these are both temporally annotated and partially aligned. To the best of our knowledge, this is the first open-source derived literary corpus containing time-stamped texts, which have been collected with the first publication date in mind. Hence, this paper is aimed at presenting this corpus and exemplifying how it can be used for diachronic literary style analysis. More specifically, section 2. considers other diachronic corpora and analyses that have been conducted based thereon. Section 3. describes our corpus specifically and section 4. shows example diachronic style analyses based on this corpus. Section 5. discusses the results and section 6. concludes this work.

2. Previous Research

One of the earlier studies of changes in an author's writing style was the study of the poet William Butler Yeats (Forsyth, 1999). Although using dated texts as a means to develop stable methods for chronological prediction is presented as a main motivation for the study, the question of change in Yeats' style is also mentioned given that scholars do not seem to agree on what change his style is supposed to have undergone. The analysis is based on distinctive marker substrings that are extracted from 142 poems using a modified version of *Monte-Carlo Feature Finding* (a quasi-random search algorithm), which are then ranked according to distinctiveness as measured by χ^2 in separating the categories *Young Yeats* and *Old Yeats*. Poems were divided into these categories based on being written either before or after 1915. Forsyth (1999) reports identifying clear markers of 'young' and 'old' Yeats based on 20 substring markers: for nine out of ten test poems their count is higher in the appropriate age category. In order to be able to assign dates to texts 'a youthful Yeatsian index' is defined as: $YYIX = (YY - OY)/(YY + OY)$, where *YY* refers to the number of younger Yeats markers and *OY* to the number of older Yeats markers found (Forsyth, 1999, p.474). A correlation of $YYIX$ and composition year yields an r of -0.84. When examining two poems that had been revised by Yeats some 30 years later, it is observable that the number of *YY* markers decreased in the revised version, while the number of *OY* markers increased.

Another temporal study was focused on the late 19th century American author Henry James (Hoover, 2007), who is deemed to have changed his style over his creative lifespan (Beach, 1918). Considering the most frequent word unigrams and a variety of different methods, such as Cluster Analysis, Burrows' *Delta*, Principal Component Anal-

Author	Timeline	Gender	Works	Size(MB)
Alice Brown	1884–1922	F	12	5.7
Amanda Minnie Douglas	1866–1914	F	51	24.5
Constance Fenimore Woolson	1873–1895	F	12	6.7
Edith Wharton	1897–1920	F	10	3.5
Elizabeth Stuart Phelps Ward	1866–1907	F	21	5.8
Gertrude Atherton	1888–1923	F	19	9.1
Harriet Beecher Stowe	1852–1886	F	18	11.2
Louisa May Alcott	1854–1893	F	16	5.6
Marion Harland	1854–1914	F	15	9.0
Susan Warner	1850–1884	F	29	18.6
Charles Dudley Warner	1872–1899	M	14	6.1
Edgar Saltus	1884–1919	M	17	3.6
Francis Marion Crawford	1882–1908	M	41	23.3
Harold McGrath	1903–1922	M	15	5.3
Henry James	1877–1917	M	32	17.3
Horatio Alger jr	1866–1906	M	37	10.3
Mark Twain	1869–1916	M	23	11
Robert W. Chambers	1894–1922	M	38	20
Timothy Shay Arthur	1847–1890	M	30	10.7
Upton Sinclair	1898–1922	M	17	8.6
William Dean Howells	1867–1916	M	38	16.7
William Taylor Adams	1855–1896	M	49	17.5

Table 1: Corpus of literary authors, indicating timeline, gender, number of works and their size in megabytes.

ysis and Distinctiveness Ratio, Hoover investigates natural partitions of James' style into three different temporal divisions of early (1877–1881), intermediate (1886–1890) and late style (1897–1917).[1] These three divisions have also been identified by literary scholars (Beach, 1918). Furthermore, Hoover notes the existence of transition periods in between which, for instance, the first novels of the late period being somewhat different from the rest of them. Analysis of the 100 words with the largest Distinctiveness Ratio that are either increasing or decreasing over time show that James appears to have increased in his use of *-ly* adverbs and also in his use of more abstract diction, preferring more abstract terms over concrete ones.

The work on temporal prediction by Klaussner and Vogel (2015) considered the task of accurately predicting the publication year of a text through the relative frequencies of syntactic word features.[2] They used multiple linear regression models to predict the year a text was published in for three data sets, the first containing Mark Twain and Henry James' texts, the second a mid 19th to early 20th reference corpus and a third one combining all data from the previous two sets. Although the two authors' data had been kept separate considering possibly different levels for them, the models disregarding authorial source tended to be more accurate (Root-mean-square-error (RMSE) of 7.2 vs. 8.0 on unseen data).[3] This indicates that Twain and James used their shared features with similar rates. Klaussner and Vogel (2015) also used a reference corpus to examine background language change, specifically *The Corpus of Historical American English (COHA)* (Davies, 2010).[4] They built an accurate model based on this corpus to approximate the general language change over time (RMSE of 4 on unseen data). However, using the same model to predict change in James and Twain was rather inaccurate for both authors (RMSE: 15.4 (Twain) / 20.3 (James)), suggesting that the two authors were rather different to the general language in terms of the stylistic features examined, Twain being somewhat more similar to it than James. Combining all data without reference to authorial source leads to more accurate results (RMSE: 1.8) and model features and estimates suggest a marked influence of Twain and James on the model through change in predictors and their associated weights. Conceptually, this set can be thought of representing a style (change) of a community, where a large proportion of people has a similar style to Twain and James.

All previously described studies have the same serious interpretative issue, i.e. from just examining one to two authors it is not obvious what elements of style change can be attributed to the individual and which would be shared by the larger community of writers to which he or she belongs. The corpus presented in the next section (section 3.) offers the possibility to draw from a set of 22 different authors, thus offering more interpretative background to what is individual and was is general with respect to literary style

[1] Distinctiveness Ratio: Measure of variability defined by the rate of occurrence of a word in a text divided by its rate of occurrence in another. Principal Component Analysis (PCA) is an unsupervised statistical technique to convert a set of possibly related variables to a new uncorrelated representation or principal components.

[2] Syntactic word features are words marked for their syntactic category within context.

[3] Hereafter, when we report RMSE we take the units to be years and do not repeat the unit. This is to be understood with respect to the caveat that the data is processed using only integer values of years. It is not the case, that temporal prediction for any text can be wrong by "7.2 years" - rather by seven years or eight years. The RMSE is an aggregate.

[4] A free sample version is accessible on: `http://corpus.byu.edu/coha` – last verified February 2018.

change.

3. Diachronic corpus

Table 1 shows the set of literary authors, comprising twenty women and twenty-two men, all of whom composed work between 1847–1923.[5] The corpus was populated in the following way: Henry James was chosen based on analyses in the literature that suggested that his style had undergone notable change over time (Hoover, 2007; Beach, 1918). Mark Twain, somewhat of a rival author, presented an interesting contrast to James (Beach, 1918; Canby, 1951), especially since Mark Twain and William James (Henry's brother) maintained an active friendship throughout their lives, both being interested in the Psychical Research and paranormal phenomena.[6]

The remaining authors were chosen by first assembling a list of male and female American authors of the 19th–20th century using *Wikipedia*[7] and then choosing those who had a few works publicly available and spread out over at least twenty years. Also, for the purpose of estimating stable word distributions, it was decided that works had to be at least 150 kilobytes in length thus discarding authors with multiple shorter works. Thus, there might be a bias towards more prominent writers, as there could be more incentive to make their data publicly available. For instance, this may result in a shift towards only certain words or expressions being used more frequently throughout. Also, there is little to no racial diversity in the data set as all authors were white, and even though individuals, such as Harriet Beecher Stowe's writings describe African Americans' conflicts, most authors probably remained in their sphere and wrote predominantly about the type of society they were exposed to themselves. Therefore, any inferences based on this set of literary authors does not necessarily extend to the population of American literary authors at large.

Apart from the apparent dislike James and Twain harboured for each other, there were also more positive connections and collaborations between authors of this corpus. Mark Twain and Charles Dudley Warner wrote *The Gilded Age* together.[8] Elizabeth Stuart Phelps Ward seems to have been an admirer of Harriet Beecher Stowe and referred to her in 1896's *"Chapters from a Life"* as the 'greatest of American women'. Constance Fenimore Woolson, a grandniece of James Fenimore Cooper, quoted William Dean Howells in one of her works and established a friendship with Henry James. Her 1884 'East Angels' is seen as a response to James' 'Portrait of a Lady' (Kreiger, 2005). Susan Warner's 1850's 'Wide, Wide World' has been described as a Feminist *Huckleberry Finn*.[9]

In terms of temporal alignment, a fair subset of the authors wrote largely in parallel. For instance, Harriet Beecher Stowe, Louisa May Alcott, Marion Harland and Susan Warner all have their first work in this corpus within four years of each other (1850–1854).[10] Elizabeth Stuart Phelps Ward and Amanda Minnie Douglas both began writing about 15 years later in 1866. The remainder of the female authors' first contribution is somewhat spread out: Constance Fenimore Woolson (1873), Alice Brown (1884), Gertrude Atherton (1888) and lastly Edith Wharton (1897). As for the male authors, Charles Dudley Warner, Mark Twain, William Dean Howells and Horatio Alger jr also made their first appearance within a few years of each other (1866–1872). The second big wave of male authors' first publication clusters around the 1880s: Henry James (1877), Francis Marion Crawford (1882), and Edgar Saltus (1884). Timothy Shay Arthur and William Taylor Adams started publishing slightly earlier than the rest, 1847 and 1855, respectively, and both remained active for about 40 years. Thus, these earlier time lines still have considerable overlap with most of the other writers in the corpus. An exception to this are Upton Sinclair and Robert W. Chambers, Harold McGrath and Edith Wharton, who only started their career in the 1890s or beginning of the 20th century. However, most authors in this corpus should be comparable in that they composed work over at least 20 years in parallel.

The set of literary authors was mainly collected from *Project Gutenberg (PG)*[11] and supplemented with works from the *Internet Archive (IA)*.[12] Project Gutenberg is the more desirable source given that the data is hand transcribed rather than scanned automatically. However, in this case acquiring data with a time stamp close to the first publication date was essential and for this reason and especially when the equivalent Gutenberg version did not have a time stamp, the Internet Archive version was chosen instead if available. The Internet Archive contains scanned version of books using Optical Character Recognition (OCR), and the quality of the processing varied considerably across books and sponsors. In this a trade-off had to be found, balancing accurate time stamp and quality of processing. Occasionally, when content was very noisy due to OCR errors, files were not included at all. In all cases, the date of a file was decided by taking the first available date, e.g. first copyright or publication date, unless a preface clearly stated that the work had been subjected to explicit revisions. The issue with dating in this case is that both dating a work too early or too late would distort the results.

All data was prepared for processing by manually removing parts that were written at a different time from the main work or introductions or comments not by the author, such as notes or introductions by editors. Additionally, table of contents were also removed, as these do not usually follow a normal sentence structure. Minimal preprocessing was needed for PG files, but the books sourced from the IA could be rather noisy, and as upon inspection each file ap-

[5] The data set is available at `www.scss.tcd.ie/clg/DCLSA/` – last verified February 2018.

[6] `http://www.apa.org/monitor/2010/04/twain.aspx` – last verified February 2018.

[7] `https://en.wikipedia.org/wiki/Category:19th-century_American_writers`– last verified February 2018.

[8] As Twain is listed as first author, it is assigned to his corpus.

[9] Usually, described this way in the book's synopsis.

[10] When using descriptions, such as *first* or *last* with respect to authors' works, this is generally to be understood with respect to this corpus; there might be cases where an earlier or later work for an author exists, but could not be included in this corpus.

[11] `http://www.gutenberg.org/` – last verified February 2018.

[12] `https://archive.org/` – last verified February 2018.

Items Found		Example Context		Occurrence	
Incorrect	*correct*	*incorrect*	*correct*	*raw*	*%*
’11	’ll	you’11	you’ll	15275	0.1
lv	ly	onlv you	only you	154	0.001
n t	n’t	could n t	couldn’t	99465	0.7
}’ / }-	y	exactl}- / exactl}’	exactly	6417	0.05
3011/ 3 r ou / 3ôu	you	3 r ou go home	you go home	2351	0.02
011	no / on	011 the table/ 011 way	on the table / no way	1474	0.01
U	ll / il / li	wiU / wUl	will / will	15895	0.1
/	I / 1 / ! /,	/ will / /”	I will / !”	10067	0.07
AV	W	AVhat	What	4508	0.03

Table 2: Common OCR errors and their correct possible realizations, their raw counts and % of processed IA tokens.

peared to have different types of OCR errors, it was deemed best to correct each file manually to correct scanning errors and remove unwanted formatting sequences. One of the issues with automatically correcting these errors was that even within one file, a misread character could refer to multiple different correct character realizations and only manual examination of the context could accurately determine the correct realization.[13] Errors that have only one possible correct version could be corrected using regular expressions, but manual correction was necessarily in cases where there was more than one possible correct version, e.g. the error ‘011’ could correspond to both ‘no’ and ‘on’ even within the same file. Table 2 shows some of the most common OCR errors and their possible correct realizations as well as occurrence of these and their rates as percentages of the raw corrected tokens in IA texts. All whitespace-separated items in the raw texts add up to 14140296 tokens, which reduces to 13614013 tokens in the manually processed version (a reduction of 4%). We estimated the number of broad differences between the two versions by considering the lines changed compared to all lines in the processed version, i.e. 137594/2146720=0.064 (6.4%).[14] It is important to note that there could be multiple changes per line and simple deletion of superfluous headings or page numbers would not be as time-intensive as manual correction of OCR errors. All processed works add up to 554 files in total, 400 (176.9 MB) from Project Gutenberg and 154 (73.7 MB) from the Internet Archive. When reducing the set to unique *author-publication year* combinations, 409 cases are left.

4. Diachronic Style Analyis

In this section, we present an example of the type of analysis that can be done based on this corpus. Specifically, we examine stylistic change in Henry James and Mark Twain and consider to what extent salient features change in the other literary authors. Section 4.1. briefly introduces the regression technique used to discover linearly changing aspects of style. Section 4.2. then reports on the results.

[13] ‘Character’ here refers to alphanumeric letter.

[14] First, all files were compared pairwise using *diff* in linux, followed by counting changed lines in the resulting output and comparing this to the overall line count in the processed data.

Feature	Model weight
beyond.IN	23515.5
broad.JJ	−37652.9
case.NN	13258.1
feet.NNS	−2693.2
joy.NN	25044.9
other.JJ	7966.1
real.JJ	13550.6
things.NNS	13562.0
usual.JJ	−19171.8
ways.NNS	1468.7
word.NN	7535.7
wore.VBD	−10892.7
since.RB	30815.1
sort.NN	−4496.1

Table 3: Syntactic word features included in the best out of four James-Twain models.

4.1. Methods

For the stylistic feature experiments, we consider ‘syntactic word’ sequences, meaning words that have been marked for their syntactic class, and thus each word is augmented with its respective part-of-speech tag.[15] In cases where an author had more than one work per year, the respective feature token count is collapsed to form a single entry for that year. The following experimental paradigm was first introduced by Klaussner and Vogel (2015) and then further developed by Klaussner and Vogel (2017) to its current state. Thereby, a set of features is selected based on its accuracy in predicting the publication year of a text. Thus, the prediction of a variable y using explanatory models is based on a function over a set of distinct variables: $\{x_1, x_2, \dots x_{p-1}, x_p\} = X$ with $y \notin X$, at the same time point $t : \{t \in 1, ...n\}$ and some error term: $y_t = f(x_{1t} \dots x_{2t}, \dots x_{p-1t} \dots x_{pt}, error)$. The general model for this is shown in eq. 1, predicting variable y, where $\hat{y}_t$ refers to the estimate of that variable at a particular time instance $t : \{t \in 1, ...n\}$, β_0 refers to the intercept

[15] To extract part-of-speech features needed for syntactic word features, the TreeTagger POS tagger (Michalke, 2014; Schmid, 1994) was used.

and β_p to the p-th coefficient of the p-th predictor x_{pt}.

$$\hat{y}_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \ldots \beta_p x_{pt}. \quad (1)$$

In the present case, the 'year of publication' is always set as the response variable, e.g. a model based on syntactic word unigrams (relative frequencies) for the year 1880 could be defined in the following way: $\hat{y}_{1880} = \beta_0 + \beta_1(I.pp_{1880}) + \beta_2(he.pp_{1880}) + \beta_3(a.det_{1880})$.

For this work, *'shrinkage'* models and specifically 'lasso' and 'ridge' as part of the 'elastic net' regression were used (Zou and Hastie, 2005).[16] These models offer an extension to the regular 'ordinary least squares' (OLS) models by additionally penalizing the magnitude of the model coefficients thus aiming to keep the model from overfitting to the data. The elastic net penalizes both the L_1 and L_2 norms.[17]

For evaluation, we used the *Root-mean-square-error* (RMSE): it is defined as the square root of the variance of the residuals between outcome and predicted value and provides the standard deviation around the predicted value, as shown in eq. 2. In the present case, RMSE units would correspond to deviations in years, e.g. a RMSE of 2 translates to an error of 2 years around the actual value.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{n}} \quad (2)$$

4.2. Experiments

For the experiments, the Twain and James data (40 cases) was separated into training and test sets by using a 75/25 split on the response variable 'year of publication'.[18] Then, the features appearing in all training instances over both their data points were extracted.[19] Using the above regression models in five-fold cross-validation, the best model within 1 standard error (SE) of the model with the lowest error, as defined by the mean-square-error MSE was selected. We ran this configuration four times, constructing different training and test splits each time in order to identify salient features over different divisions. Mean training and test set accuracy over all four iteration are 9.7 and 9.9 RMSE respectively. The best performing model in terms of accuracy has 14 features, shown in table 3, and achieves a RMSE of 9.8 on the training and 5.6 on the test set. Prevalent features over all four iterations are: *"broad.JJ"*, *"case.NN"*, *"other.JJ"*, *"things.NNS"*, *"usual.JJ"*, *"word.NN"* and *"wore.VBD"*.

Figure 1 shows two salient features for James and Twain: the adjectives *broad* and *usual*, where both decrease in usage over time. However, without having examined other authors for the same features, it is not clear whether James' and Twain's common trend is remarkable and indicates agreement among them or is fairly common for American writers at this time. Thus, one of the questions arising from this analysis is whether other contemporaneous authors show a similar trend for these features.

For this purpose, we use our predictive James-Twain model to predict the publication year for each author in the corpus separately. If James and Twain's sharing common trends for these syntactic word features is truly unique, publication dates of other authors' works in the set should not be predicted accurately on average. Table 4 shows the results of this prediction task for each author separately in the second column and the results for only those works within the same year range as the training data in the third column. The final column shows the drop in RMSE with respect to the number of test items left out.[20] Both the works of Twain and James are predicted with a similar accuracy, i.e. a RMSE of 4.4 and 4.5 respectively. Considering now prediction accuracy for the remaining authors in the set using the James-Twain model (first column in table 4) shows that most authors' prediction accuracy is far below that of Twain and James. This suggests that their trend for these features may not have followed the same pattern over time. However, some of the authors composed work some time before or after James and Twain and extrapolation may have caused a drop in prediction accuracy. The third column therefore shows what happens, when works outside of Twain and James' combined timeline (1869–1917) are left out for each author. While some scores stay exactly the same, e.g.

Author	RMSE	RMSE (−ext)	−(RMSE/item)
Twain	4.4	4.4	NaN
James	4.5	4.1	−0.4
Arthur	28.8	17.2	−1.0
S. Warner	24.8	19.0	−0.7
Stowe	26.4	16.0	−1.5
Alcott	**11.9**	**5.9**	**−1.5**
Harland	16.7	14.8	−0.9
Adams	13.4	12.0	−0.1
Douglas	10.3	10.0	−0.1
Ward	17.1	14.4	−0.9
Alger	15.2	15.5	+0.1
Howells	11.8	11.8	+0.0
C.D. Warner	13.8	13.8	NaN
Woolson	13.5	13.1	NaN
Crawford	11.9	11.9	NaN
Brown	21.3	14.7	−2.2
Saltus	14.0	11.6	−1.0
Artherton	28.1	20.8	−1.5
Chambers	**32.1**	**29.4**	**−0.5**
Wharton	25.8	23.7	−0.7
Sinclair	20.5	13.0	−1.5
McGrath	27.5	27.0	−0.2

Table 4: Test RMSE for all authors: showing RMSE for all works (RMSE), only works within model range (RMSE (−ext)) and the drop in RMSE per excluded test item (−(RMSE/item)). Bold printed authors are graphically compared to James and Twain in figure 2.

[16] All regression models were computed using the *glmnet* package in R (Friedman et al., 2010), which in our opinion currently offers the most transparent and flexible implementation.

[17] L1: $\|\beta\|_1$: $\sum_i |\beta_i|$ and L2: $\|\beta\|_2^2$: $\sum_i \beta_i^2$

[18] Using the *caret* package in R (Kuhn et al., 2014).

[19] In further studies, this constraint could be relaxed to 'present in most instances'.

[20] 'NaN' indicates that no removal of test items outside the range was necessary, hence no change in RMSE.

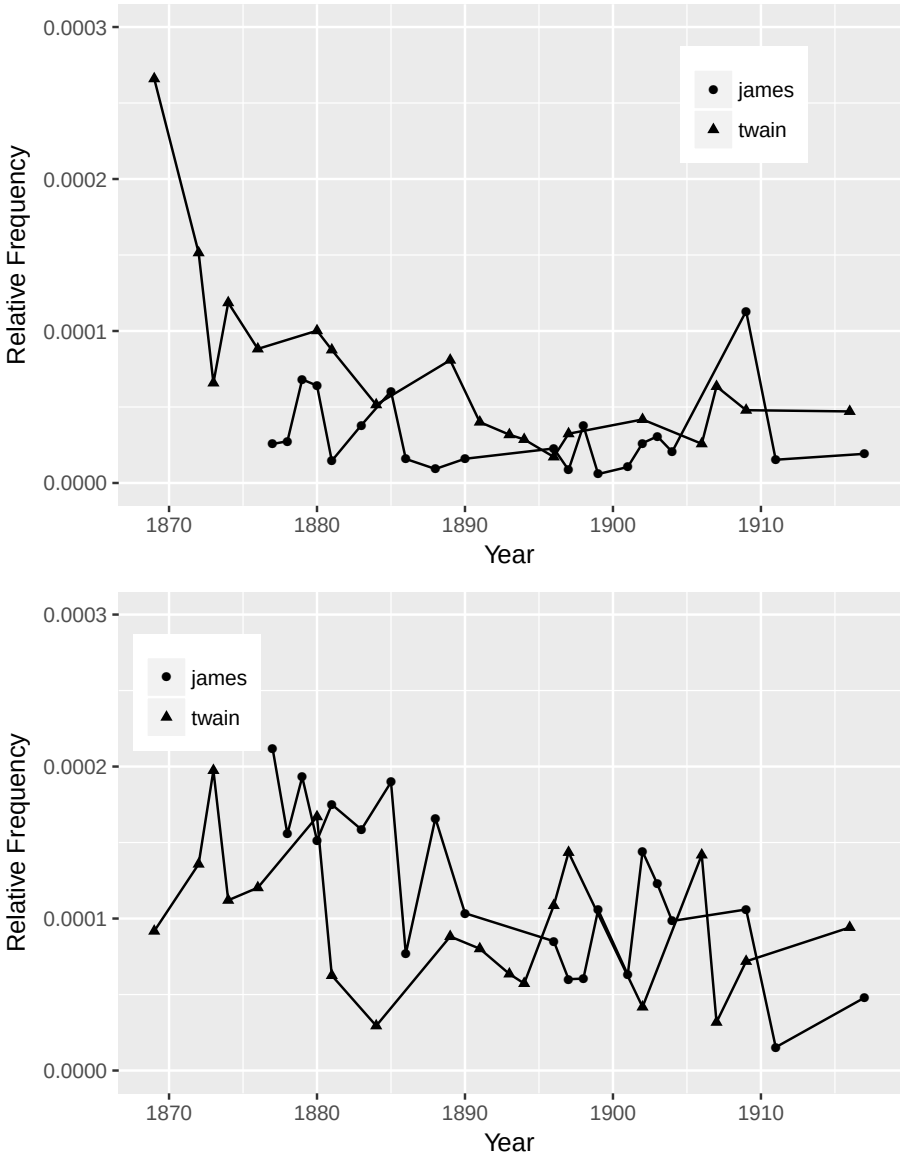


Figure 1: Two salient features for Twain and James. The adjectives 'broad' (top) and 'usual' (bottom).

Adams and Howells, some of them rather improve in accuracy. For instance, leaving out Louisa May Alcott's extra works results in a drop from 11.9 to merely 5.9 RMSE. This actually makes her works fit almost as well with the model as Twain and James' data, even though it was not trained on it. Figure 2 shows pairwise differences between Alcott, Twain and James for the adjective *usual*. Excluding her earlier works before 1865 renders her change in these features a lot closer to the other two authors. In comparison, we examine the author with the highest error, even after removing difficult test pieces, Robert W. Chambers. From the pairwise differences in the bottom plot in figure 2, it can be observed that his trend for the same feature appears to be on a different frequency level than James and Twain, explaining why the model may not be able to accurately date his works based on this high ranking feature among others.

5. Discussion

Section 3. introduced and described a new parallel diachronic literary corpus that can be used to compare among female and male American literary authors from the 19th century. The analysis in the previous section has shown how one can detect salient features based on a two-author set that are discriminatory as to the publication date of texts of these authors. Highly salient model features could be interpreted as being interesting in terms of what these authors have in common when considering stylistic change over time.[21] Yet to what extent these features are changing in this fashion exclusively for the authors considered can only be decided by examining contemporaneous authors that composed works in parallel. Our analysis of works of both Mark Twain and Henry James returned a model with a few highly salient features that when examined visually showed development over time. Trying to use this model to predict other authors' works generally returned much higher error rates than for the two authors on their entire sets. However, the average training and test set error of 9.7 and 9.9 are extremely close to the RMSE scores of

[21] Here, we only considered features that exhibit linear change.

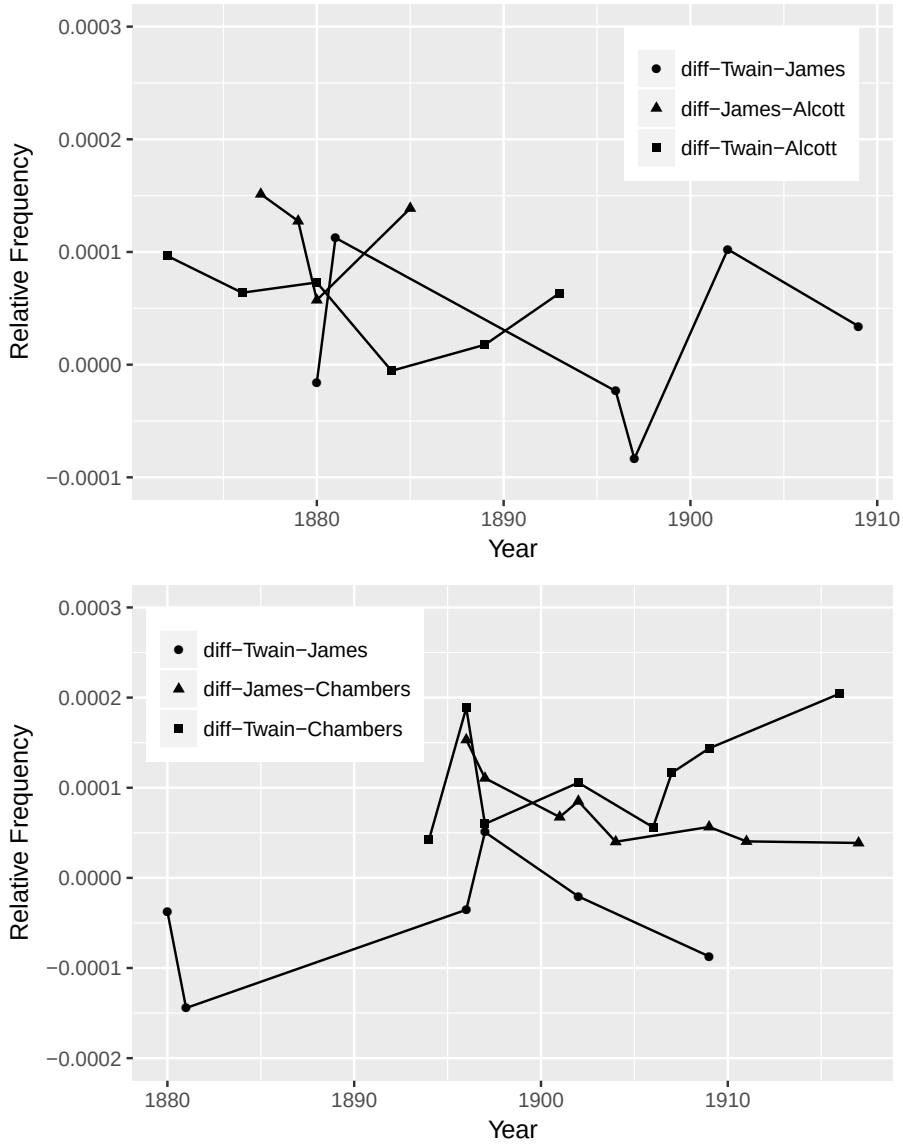


Figure 2: Pairwise differences between Twain and James and Alcott (top) for the adjective 'usual' and Chambers for the noun 'joy' (bottom).

Douglas (10.0), Adams (12.0), Howells (11.8), Crawford (11.9) and Saltus (11.6), especially taking into account that their data would be truly *unseen*. Thus, without also having examined combined models of or with other authors, it is not clear how close James and Twain are in terms of stylistic change regarding the features examined here. This analysis has certainly cast doubt on the *extraordinariness* of shared trends of features, suggesting that parallel analysis of other authors may very well return even a stronger agreement between authors than was witnessed in James and Twain's case.

This analysis has been inclusive with respect to showing that there are common James-Twain features unique in style development to these two authors. Lousia May Alcott gets arguably too close in terms of temporal development to render the discovered features true James and Twain markers. This finding does not necessarily extend to other feature types, for instance they could share unique similarities on stem or syntactic features. What this analysis has shown is that, even for pairwise comparisons, contemporaneous authors need to be examined in parallel in order to give meaning to the individual analyses.

6. Conclusion

In order to analyze style change accurately and determine what features are likely to be more unique in the particular author's case, other contemporary authors have to be examined in parallel. This paper has presented a corpus that can be used for just this purpose, specifically to analyze an author's style with respect to other authors that have composed works during the same time span.

7. Acknowledgements

We would like to thank our anonymous reviewers for their helpful suggestions on how to improve the earlier version of this paper. This research is supported by Science Foundation Ireland (SFI) through the CNGL Programme (Grant 12/CE/I2267 and 13/RC/2106) in the ADAPT Centre (www.adaptcentre.ie).

8. Bibliographical References

Beach, J. W. (1918). *The Method of Henry James*. Yale University Press.

Canby, H. S. (1951). *Turn West, Turn East: Mark Twain and Henry James*. Biblo & Tannen Publishers.

Daelemans, W. (2013). Explanation in computational stylometry. In *Computational Linguistics and Intelligent Text Processing*, pages 451–462. Springer.

Davies, M. (2010). The corpus of historical American English: 400 million words, 1810-2009. *http://corpus. byu. edu/coha/.*, 24:2011. (last verified: 21.02.2018).

Forsyth, R. (1999). Stylochronometry with substrings, or: a poet young and old. *Literary and Linguistic Computing*, 14(4):467–478.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22.

Hoover, D. L. (2007). Corpus stylistics, stylometry, and the styles of Henry James. *Style*, 41(2):174–203.

Klaussner, C. and Vogel, C. (2015). Stylochronometry: Timeline prediction in stylometric analysis. In *Research and Development in Intelligent Systems XXXII*, pages 91–106. Springer.

Klaussner, C. and Vogel, C. (2017). Temporal predictive regression models for linguistic style analysis. *Journal of Language Modeling*, 5(3):To appear.

Kreiger, G. (2005). East Angels: Constance Fenimore Woolson's revision of Henry James's the Portrait of a Lady. *Legacy*, 22(1):18–29.

Kuhn, M., Contributions from Wing, J., Weston, S., Williams, A., Keefer, C., Engelhardt, A., Cooper, T., Mayer, Z., and the R Core Team, (2014). *caret: Classification and Regression Training*. R package version 6.0-30.

Michalke, M., (2014). *koRpus: An R Package for Text Analysis*. (Version 0.05-4).

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of international conference on new methods in language processing*, volume 12, pages 44–49. Manchester, UK.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(2):301–320.

Text Simplification from Professionally Produced Corpora

Carolina Scarton, Gustavo Henrique Paetzold, Lucia Specia
Department of Computer Science, University of Sheffield
Regent Court, 211 Portobello, Sheffield, S1 4DP, UK
{g.h.paetzold, c.scarton, l.specia}@sheffield.ac.uk

Abstract

The lack of large and reliable datasets has been hindering progress in Text Simplification (TS). We investigate the application of the recently created Newsela corpus, the largest collection of professionally written simplifications available, in TS tasks. Using new alignment algorithms, we extract $550,644$ complex-simple sentence pairs from the corpus. This data is explored in different ways: (i) we show that traditional readability metrics capture surprisingly well the different complexity levels in this corpus, (ii) we build machine learning models to classify sentences into complex vs. simple and to predict complexity levels that outperform their respective baselines, (iii) we introduce a lexical simplifier that uses the corpus to generate candidate simplifications and outperforms the state of the art approaches, and (iv) we show that the corpus can be used to learn sentence simplification patterns in more effective ways than corpora used in previous work.

Keywords: text Simplification, simplification corpora, Newsela

1. Introduction

Text Simplification (TS) consists in making texts more easily comprehensible. It can take many forms: Lexical Simplification (LS), in which complex words are replaced by simpler alternatives (Devlin, 1999), Syntactic Simplification (SS), which consists in changing the syntactic structure of a sentence (Siddharthan, 2006), and Semantic Simplification, in which portions of the text are paraphrased (Kandula et al., 2010).

Current empirical approaches rely mostly on the Wikipedia-Simple Wikipedia parallel corpus (Coster and Kauchak, 2011). This resource has been used by Machine Translation (MT) approaches (Zhu et al., 2010), tree transductors (Paetzold and Specia, 2013; Feblowitz and Kauchak, 2013), integer programming techniques (Woodsend and Lapata, 2011), and discriminative linear models (Bach et al., 2011). In LS, Yatskar et al. (2010) extract candidate simplifications from Wikipedia and Simple Wikipedia edit histories, and Horn et al. (2014) extract word correspondences from word alignments between the complex-simple segments in the corpus.

Even though the Simple Wikipedia corpus has been a valuable resource for modern TS, as discussed in Yasseri et al. (2012), Amancio and Specia (2014) and Xu et al. (2015), it is very small ($167,689$ parallel sentence pairs (Coster and Kauchak, 2011)) in comparison to bilingual corpora used with similar modelling techniques in MT and, more critically, covers a limited range of simplification operations, which are applied in *ad hoc* ways by volunteer editors. Xu et al. (2015) introduce a new resource that allegedly addresses these limitations: the **Newsela corpus** (Newsela, 2016). Unlike Simple Wikipedia, the Newsela corpus was created by professional editors and targets a specific audience (students), which should make it a more reliable resource for TS. However, the Newsela corpus has only recently started to be exploited for this task and not enough work has been done to understand its potential.

In this paper, we investigate whether (Xu et al., 2015)'s claims hold in practice. We produce sentence alignments for the Newsela corpus (Section 2.) and conduct experiments to evaluate its effectiveness in TS tasks, namely: readability analysis (Section 4.), complex vs. simple classification, complexity level prediction, lexical simplification and MT-based sentence simplification (Section 5.).

2. Aligning the Newsela Corpus

The Newsela corpus (version 2016-01-29.1) is composed of $10,787$ news articles in English, which includes $1,911$ articles in their original form as well as in 4 or 5 versions rewritten by humans to suit different reading reading levels. Each document is characterised by a unique identifier, a version identifier (from 0 – most complex to 5 – simplest), and a reading level from 2 to 12, where 2 represents the lowest and 12 the highest level. Version identifiers capture the relationship between the reading levels of a pair of documents: e.g. version 1 has a lower reading level than version 0, version 2 has a lower reading level than version 1.

Articles are only aligned at document level and there is no guarantee that different versions of an article will have the same number of sentences, nor that they will be aligned in 1-to-1 fashion. The absence of paragraph and sentence alignments limits the use of the data.

To produce such alignments, we use the algorithms in (Paetzold and Specia, 2016d), which employ a vicinity-driven search approach. These algorithms address the limitations of previous strategies (Barzilay and Elhadad, 2003; Coster and Kauchak, 2011; Smith et al., 2010; Xu et al., 2015; Bott and Saggion, 2011) by disregarding the need for supervised or semi-supervised training, allowing long-distance alignment skips, capturing 1-N and N-1 alignments, and exploiting the fact that the order in which information is presented is constant between pairs of aligned Newsela articles. Because the vicinity-driven approach of Paetzold and Specia (2016d) exploits a series of assumptions that can be made about the Newsela corpus, it is more efficient than more sophisticated approaches that perform exhaustive search over all possible

paragraph/sentence alignments (Štajner et al., 2017), while still offering comparable alignment accuracy.

The result of the alignment is a corpus with 19,198 pairs of articles aligned at both paragraph (300,475 pairs) and sentence (550,644 pairs) levels. This is over three times larger than the Wikipedia–Simple Wikipedia corpus (Coster and Kauchak, 2011), making it the largest corpus of its kind.

Columns 2 to 4 in Table 1 illustrate the number of paragraph and sentence alignments for all version pairs in the corpus. We categorise the sentence alignments according to four types of simplification:

- **None:** Complex and simple sentences are identical (146,251 pairs).
- **Compression:** Multiple complex sentences are aligned to fewer simple sentences (24,661 pairs).
- **Splitting:** Multiple simple sentences are aligned to fewer complex sentences: (121,582 pairs).
- **Rewriting:** Same number of complex and simple sentences, but with different content: (258,150 pairs).

Pair	# Doc.	# Parag.	# Sent.	% None	Avg. TER
0-1	1,910	39,414	69,443	42.1	**0.193**
0-2	1,910	35,720	60,725	26.5	0.316
0-3	1,910	27,752	44,168	16.7	0.449
0-4	1,882	19,369	28,499	12.3	0.537
0-5	42	261	346	5.5	0.647
1-2	1,910	38,497	75,953	37.8	**0.222**
1-3	1,910	30,824	55,572	19.1	0.400
1-4	1,882	22,163	36,089	12.3	0.511
1-5	42	300	417	5.3	0.651
2-3	1,910	33,033	69,416	29.4	**0.308**
2-4	1,882	24,363	45,392	15.8	0.455
2-5	42	329	523	6.5	0.605
3-4	1,882	27,635	62,413	29.6	**0.325**
3-5	42	386	706	9.5	0.554
4-5	42	429	982	20.5	**0.423**
Total	19,198	300,475	550,644	26.6	0.440

Table 1: Documents, aligned paragraphs and sentences at all levels (columns 2-4), % of "none" alignments, and average TER (columns 5-6)

On average, sentence lengths remain close for adjacent levels (e.g. 25.0 & 24.4 for levels 0-1), but sentences become shorter for higher levels (12.5 & 11.1 at levels 4-5). This shows that editors significantly compress text while simplifying.

3. Related Work

Xu et al. (2015) are the first to present an analysis of the Newsela corpus. They compare the Newsela and Wikipedia-Simple Wikipedia data using several metrics, showing that the Newsela corpus appears to be more useful. However, they do not use it in any tasks (e.g. lexical simplification) like we propose in this paper.

Besides proposing alignment algorithms for the Newsela corpus, Štajner et al. (2017) also build MT-based models with the aligned data. As test set, instead of using part of Newsela data, the Wikipedia-Simple Wikipedia dataset proposed by Xu et al. (2016) is used. Two out of three systems trained with Newsela aligned data perform better in terms of simplicity than state-of-the-art systems for the same corpus.

Zhang and Lapata (2017) train an attention-based encoder-decoder model (Bahdanau et al., 2014) and use reinforcement learning with a reward policy combining SARI (to measure simplicity) (Xu et al., 2016), BLEU (to measure grammaticality) (Papineni et al., 2002) and cosine similarity (to measure meaning preservation). This approach shows improvements over a model trained using a phrase-based MT approach in terms of BLEU and SARI.

Alva-Manchego et al. (2017) propose a TS model that uses predicted simplification operations. Simplification operations automatically annotated in source sentences are predicted as a first step, using sequence labelling techniques. As operations, they model only *replace* and *delete*. Their TS model produces better results according to human judgements for simplicity than general-purpose MT-based models.

In general, the aforementioned contributions explore MT-based techniques and train systems using the Newsela data in similar ways as it was done previously for Wikipedia-Simple Wikipedia data. However, none of them provide an analysis of the Newsela data in terms of readability of the aligned data, the use of the data for complex vs. simple classification or complexity level prediction, or the impact of the data in state-of-the-art LS approaches.

4. Corpus Analysis

We analyse the sentence-aligned Newsela corpus to (i) understand the differences between its various levels of simplification, and (ii) investigate how existing readability and psycholinguistic metrics fair in distinguishing these levels.

Edit Rate This analysis focuses on the differences in edits between the various simplified versions. We use TER[1] as a metric of edit distance, as it is widely used for this purpose in MT evaluation.

Columns 5 and 6 of Table 1 show the percentage of alignments with TER = 0 ("% None") and averaged TER. As expected, the non-adjacent versions have higher TER values, e.g. the distance between levels '0' and '1' is much smaller than the distance between levels '0' and '5'. The percentage of sentences with no edits decreases as we move from adjacent to non-adjacent levels. Interestingly, between the adjacent levels, the closer to the original level, the lower the TER, e.g. there are fewer edits between '0' and '1' than between '1' and '2'.

Readability Metrics Here we evaluate standard readability metrics that aggregate shallow text information (such as number of syllables and words): Flesch Reading Ease, Flesch-Kincaid Grade Level, SMOG Index, Gunning Fog Index, Automated Readability Index, Coleman-Liau Index, Linsear Write Formula and Dale-Chall Readability Score from the TEXTSTAT toolkit[2]. Figure 1 shows the Flesch Reading Ease box plot for pairs of original and simplified sentences with level '0' as original version. Flesch varies from 0 (most complex) to 100 (simplest). The Flesch index for the simplified versions is higher than for the original version in all cases, which is an expected behaviour. Therefore, although the simplified versions of the Newsela

[1] http://www.cs.umd.edu/~snover/tercom

[2] https://pypi.python.org/pypi/textstat

corpus can be composed of more and/or longer sentences than the original, the information encoded in them is still simpler.

A similar trend is observed for all other readability metrics and between all levels. For completion, we also show in Figure 2 the box plot for the Flesch-Kincaid Grade Level metric for 0-to-n original/simplified pairs. As expected, simplified versions have lower Flesch-Kincaid scores.

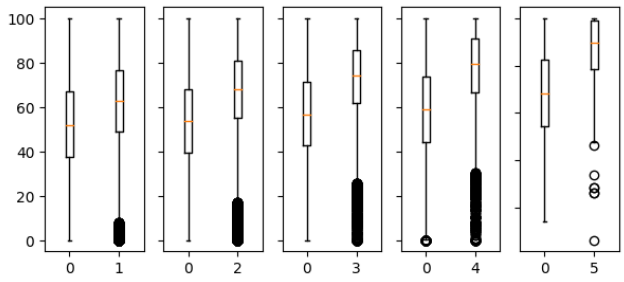


Figure 1: Flesch Reading Ease for simplified versions from level '0'

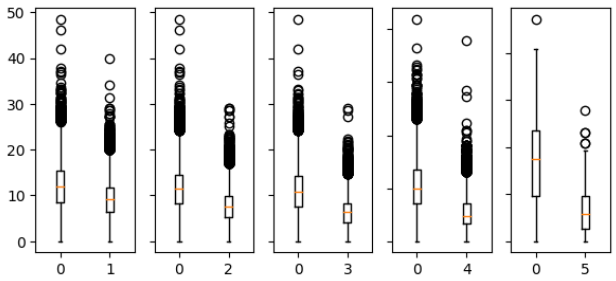


Figure 2: Flesch-Kincaid Grade Level for simplified versions from level '0'

Psycholinguistic Metrics We also analysed the data using 12 psycholinguistic features, motivated by those in the Coh-Metrix tool (Graesser et al., 2004):[3]

- Number of words / tokens / letters/ syllables
- Type/token ratio
- Ratio betw. numbers of letters and words
- Ratio betw. numbers of syllables and words
- Number of content words
- Mean age of acquisition / familiarity / imageability / concreteness score of content words

For sentences without content words, the correspondent features were assigned zero. We extracted age of acquisition, familiarity, imageability and concreteness features from the bootstrapped MRC database (Paetzold and Specia, 2016b), which is an extended version of the original MRC database (Coltheart, 1981).

Original sentences have a higher number of words, tokens, letters and syllables, as expected. Type/token ratio and number of content words are not considerably different between original and simplified versions. Figures 3 and 4 show the box plots for the ratio between number of letter and number words and the ration between number of syllables and number words, respectively, when '0' is the original level. Simplified versions have a higher ratio of letters per words and a lower ratio of syllables per words, when compared to original versions. It appears that even though simplified sentences have slightly longer words, such words have fewer syllables, which is often a sign of simplicity.

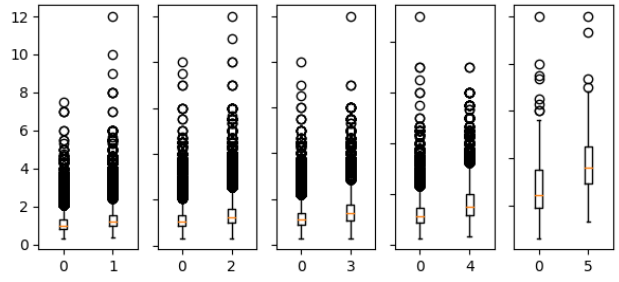


Figure 3: Ratio between number of letters and number of words for simplified versions from level '0'

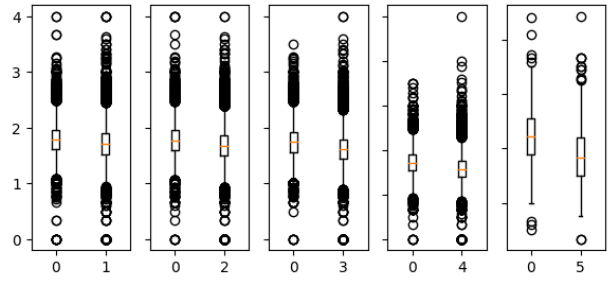


Figure 4: Ratio between number of syllables and number of words for simplified versions from level '0'

Figures 5 and 6 show the box plots for age of acquisition and imageability metrics, when '0' is the original. Age of acquisition aims to define the age at which a given word is learned, whilst imageability refers to the mental capability of retrieving an image, given a word. As expected, simplified sentences show lower values for age of acquisition and higher values for imageability than their original counterparts. Familiarity (the frequency to which a word is seen, heard or spoken daily) and concreteness did not show differences between simplified and original sentences.

5. Using the Corpus in TS Tasks

5.1. Complex vs. Simple Classification

Here we present sentence-level binary classifiers created for all possible combinations of levels of simplification. Sentences from the more complex version were assigned the label "complex", while their simpler counterpart (at any level), the label "simple". For sentences pairs whose TER is 0 (no simplification made), both original and simplified sentences were considered "simple".

We trained Stochastic Gradient Descent (SGD) classifiers (with hinge loss function) using the scikit-learn toolkit (Pe-

[3] http://cohmetrix.com

	Majority			Readability			Psycholinguistic			All		
	P	R	*F*	P	R	*F*	P	R	*F*	P	R	*F*
0-1	0.634	1.0	0.491	0.684	0.929	0.641	0.651	0.889	0.576	0.696	0.936	**0.661**
0-2	0.576	1.0	0.421	0.715	0.857	0.711	0.660	0.826	0.633	0.724	0.860	**0.724**
0-3	0.545	1.0	0.385	0.740	0.875	0.759	0.687	0.782	0.684	0.753	0.872	**0.770**
0-4	0.533	1.0	0.370	0.756	0.917	0.794	0.708	0.789	0.717	0.776	0.909	**0.808**
0-5	0.514	1.0	0.349	0.758	0.951	0.809	0.718	0.837	0.740	0.767	0.953	**0.823**
1-2	0.616	1.0	0.470	0.681	0.881	0.652	0.643	0.881	0.555	0.696	0.868	**0.664**
1-3	0.553	1.0	0.393	0.715	0.853	0.725	0.674	0.752	0.664	0.727	0.845	**0.734**
1-4	0.533	1.0	0.370	0.744	0.891	0.778	0.713	0.773	0.704	0.765	0.890	**0.791**
1-5	0.514	1.0	0.349	0.798	0.954	0.849	0.742	0.781	0.750	0.801	0.957	**0.853**
2-3	0.586	1.0	0.433	0.662	0.861	0.638	0.633	0.896	0.584	0.674	0.848	**0.653**
2-4	0.543	1.0	0.382	0.704	0.851	0.717	0.677	0.743	0.664	0.722	0.847	**0.732**
2-5	0.517	1.0	0.352	0.761	0.929	0.800	0.726	0.776	0.730	0.772	0.922	**0.810**
3-4	0.587	1.0	0.434	0.649	0.883	0.611	0.622	0.828	0.551	0.655	0.869	**0.628**
3-5	0.525	1.0	0.361	0.709	0.875	0.727	0.685	0.729	0.673	0.711	0.888	**0.745**
4-5	0.557	1.0	0.399	0.664	0.837	0.656	0.644	0.753	0.626	0.644	0.789	**0.622**

Table 2: Precision, recall and F-measure of classifiers for different simplification levels

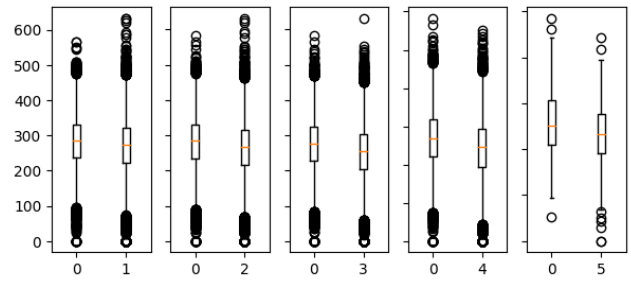


Figure 5: Age of acquisition for simplified versions from level '0'

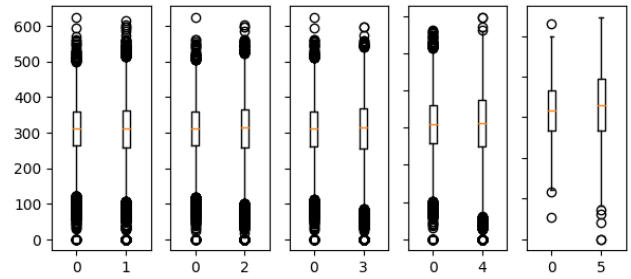


Figure 6: Imageability for simplified versions from level '0'

dregosa et al., 2011) with hyperparameters optmised using grid search. As features we used the nine readability metrics from the TEXTSTAT toolkit and the 12 psycholinguistic features, mentioned in Section 4..

Three models were built: one with the readability metrics only, one with the psycholinguistic metrics only and another with both. The classifiers were evaluated by using 10-fold cross-validation. As a baseline, we used a majority class classifier (Majority).

Table 2 shows the results for each complex-simple level. As expected, the classifiers built for non-adjacent levels achieve better performance than those for adjacent levels (in terms of F-measure). The opposite behaviour is observed for the majority class models. This is expected, however, since the number original/simplified pairs whose TER is 0 is much larger in adjacent levels than in non-adjacent levels, leading to more biased (and hence easier to predict) instances. It can also be noticed that the precision and F-measure of the classifiers follow the degree of difference between the complex and simple levels, as shown by TER (Table 1). Recall, on the other hand, is higher for adjacent levels. This is most likely caused also by the large number of sentences considered simple because of no changes in TER (% None in Table 1). All classifiers outperform the majority class baseline and the best classifiers use the combination of both types of metrics as feature.

5.2. Complexity Level Prediction

Here we directly predict the level of complexity of a sentence. These are defined as 2-12 reading proficiency levels, as explained in Section 2.: the higher the level, more complex the text. We use the same feature sets as in Section 5.1., but mix all sentences to build a single model and use the Ridge Regression algorithm.

We evaluate the model in terms of Mean Absolute Error (MAE). As baseline, we considered the MAE obtained from applying the mean complexity level of the training set as the "prediction" for all instances in the test set. MAE values of 1.793, 1.962 and 1.715 were obtained for models built with readability, psycholinguistic and all features, respectively. The mean baseline MAE was of 2.247, and therefore all models outperformed the baseline, with the best model using all features.

5.3. Lexical Simplification

Generator	Pot.	Prec.	Rec.	F1
Horn	0.569	0.235	0.131	0.168
Devlin	0.647	0.133	0.153	0.143
Biran	0.610	0.130	0.144	0.136
Glavas	0.724	0.142	0.191	0.163
Paetzold	**0.856**	0.180	**0.252**	**0.210**
Newsela	0.602	**0.304**	0.128	0.180

Table 3: Substitution Generation results

	Famil.	Glavas	Paetzold
Horn	0.334	0.332	0.352
Devlin	0.291	0.265	0.341
Biran	0.230	0.267	0.238
Glavas	0.186	0.251	0.231
Paetzold	0.350	0.325	0.378
Newsela	**0.372**	**0.344**	**0.400**

Table 4: Accuracy in the full pipeline evaluation

Here we assess the potential of our corpus in LS. LS is commonly addressed as a pipeline of steps: candidates for a target complex word are produced via a **Substitution Generation** (SG) method, filtered with respect to the context of the complex word via a **Substitution Selection** (SS) method, and finally ordered for simplicity by a **Substitution Ranking** (SR) method.

We use our aligned corpus for SG following the state of the art approach in (Horn et al., 2014). First, we produce word alignments using Meteor (Denkowski and Lavie, 2011) and extract complex-to-simple word correspondences. Then we filter word pairs with different POS tags, where the complex word is a stop word, or either word is a proper noun. Finally, we generate all possible inflections for nouns and verbs (Burns, 2013).

We compare this approach to six other generators from a recent benchmark (Paetzold and Specia, 2016a): the Horn generator (Horn et al., 2014), which employs the approach described above over Wikipedia-Simple Wikipedia data, the Devlin (Devlin and Tait, 1998), Biran (Biran et al., 2011), Glavas (Glavaš and Štajner, 2015) and Paetzold (Paetzold and Specia, 2016c) generators, which exploit WordNet, comparable complex-to-simple documents, typical word embeddings and context-aware word embeddings, respectively. All generators were implemented with the LEXenstein framework (Paetzold and Specia, 2015).

We use the BenchLS dataset as our gold-standard dataset (Paetzold and Specia, 2016a). It is the largest dataset of its kind, with 929 instances, each composed by a sentence, a target complex word, and a set of gold substitutions given by humans. To compare the generators, we use standard metrics: **Potential** – the proportion of instances in which at least one of the candidates generated is in the gold-standard, **Precision** – the proportion of generated substitutions that are in the gold-standard, **Recall** – the proportion of gold-standard substitutions that are among the generated substitutions, and **F1**. Table 3 reveals that our approach achieves the highest Precision overall, as well as higher Potential and F1 scores.

We also evaluated our generator in practice through a full pipeline evaluation, where the output is the best lexical simplification for each complex word. To do so, we paired all aforementioned generators with three state of the art SR strategies:

- **Familiarity** (Paetzold and Specia, 2016b): Ranks candidates according to their word familiarity scores, as extracted from the bootstrapped MRC database.
- **Glavas** (Glavaš and Štajner, 2015): Ranks candidates according to various features, then obtains a final ranking for a candidate by averaging the ranks of said features.
- **Paetzold** (Paetzold and Specia, 2015): Learns a ranking model from a binary classification setup.

All rankers were implemented with the LEXenstein framework with features and settings as in (Paetzold and Specia, 2016a). The gold-standard test set used is also BenchLS, and the metric is **Accuracy**: the ratio with which the highest ranking candidate is not the target word itself and is among the gold-standard candidates. Table 4 shows that our generator outperformed all others with any ranking method, highlighting the potential of the Newsela corpus for LS.

5.4. Sentence Simplification

Simplification can be addressed as a "translation approach" (Shardlow, 2014). This approach requires a large enough sentence-aligned complex-simple corpus and a method to learn simplification rules, such as off-the-shelf Statistical Machine Translation (SMT) toolkits.

We experiment with the aligned Newsela corpus following an SMT-like pipeline, using only the adjacent levels of simplification (0-1, 1-2, 2-3, 3-4 and 4-5) (278.184 sentences). For comparison, we build a model using the Wikipedia-Simple Wikipedia sentence-aligned corpus ($167,689$ sentences). Both datasets were divided in approximately 70% for training, 10% for development and 20% for test. Additionally, we also built a model using a subset of the Newsela dataset containing the same number of sentence pairs as the Wikipedia-Simple Wikipedia dataset, i.e. the training and development sets were reduced via random sampling to the size of the Wikipedia-Simple Wikipedia dataset.

We train standard MOSES toolkit (Koehn et al., 2007) with default configurations. Table 5 shows the evaluation of the simplification systems built in terms of BLEU, SARI and Flesch Ease Index. In this table "full" refers to the model trained with the entire Newsela dataset and "wiki size" refers to the model trained with the portion of the Newsela dataset with the same size as the Wikipedia-Simple Wikipedia dataset.

BLEU scores are higher for the Newsela trained system in both "full" and "wiki size" settings, which can indicate that the models trained with Newsela data are producing outputs more grammatically correct than the model trained with Wikipedia-Simple Wikipedia data. SARI, however, shows that models built with Wikipedia-Simple Wikipedia data seems to be producing slightly simpler outputs. The Flesch index for simplified sentences (FLESCH-S) is lower than that of the reference sentences (FLESCH-R) for both corpora. This seems to reflect the fact that automatic simplifications are closer to the original (FLESCH-O) than to the reference.

We also experiment with variants of the test set where only sentences with at least one edit (TER > 0) or with no edits at all (TER $= 0$) are used. Sentences with TER $= 0$ should not be modified as they are already simple, and thus the MT output should be exactly the same as the reference (and the original). This is often a problem in SMT-based simplification approaches, which tend to over simplify and introduce noise. The Newsela trained models are still the best in terms of BLEU, while show slightly smaller SARI. However, as the results for all TER $= 0$ suggest, SARI is

	BLEU	SARI	FLESCH-S	FLESCH-O	FLESCH-R
Wikipedia	0.569	0.302	66.93	66.21	74.32
Wikipedia: TER = 0	0.918	0.330	71.12	69.85	69.85
Wikipedia: TER > 0	0.454	0.292	67.29	65.34	79.98
Newsela (full)	0.692	0.270	75.04	74.89	80.62
Newsela (full): TER = 0	0.992	0.330	87.49	87.33	87.33
Newsela (full): TER > 0	0.575	0.238	70.34	70.19	78.91
Newsela (wiki size)	0.691	0.272	75.10	74.89	80.62
Newsela (wiki size): TER = 0	0.991	0.330	87.52	87.33	87.33
Newsela (wiki size): TER > 0	0.574	0.240	70.41	70.19	78.91

Table 5: Results for SMT-based simplifiers

not a reliable metric when original, reference and simplified sentences are the same. For all cases where TER = 0, the SARI value was 0.330, which can be seem as a low value if the systems are producing an output equal to the reference. Since this metric was designed for cases where sentences should also be simplified (as explained in Xu et al. (2016)), the use of SARI for cases where the original sentences are already simple is not reliable.

6. Conclusions

Upon studying the sentence-aligned Newsela corpus we found that: (i) it follows an expected TER distribution, with the lowest TER being between adjacent levels; (ii) the simplified sentences score as more readable than their original counterparts according to traditional readability metrics, and (iii) the corpus proved a more reliable source of complex-simple correspondences for LS and MT-based simplification than the Wikipedia-Simple Wikipedia corpus. We achieve some the highest performance to date when generating candidate substitutions for complex words as well as when applying these into a full LS pipeline. Improvements for MT-based simplification using the Newsela corpus are also observed but more in depth (manual) evaluation is needed for these experiments.

In the future, we hope that the aligned corpus will lead to better data-driven approaches to TS. We cannot release the aligned Newsela corpus, but it can be recreated using MASSAlign[4] (Paetzold et al., 2017), which provides the alignment algorithm used.

7. Acknowledgements

This work was supported by the EC project SIMPATICO (H2020-EURO-6-2015, grant number 692819).

8. Bibliographical References

Alva-Manchego, F., Bingel, J., Paetzold, G. H., Scarton, C., and Specia, L. (2017). Learning how to simplify from explicit labeling of complex-simplified text pairs. In *Proceedings of IJCNP*, pages 295–305.

Amancio, M. A. and Specia, L. (2014). An analysis of crowdsourced text simplifications. In *Proceedings of the 3rd PITR*, pages 123–130.

Bach, N., Gao, Q., Vogel, S., and Waibel, A. (2011). Tris: A statistical sentence simplifier with log-linear models and margin-based discriminative training. In *IJCNLP*.

[4] https://github.com/ghpaetzold/massalign

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Barzilay, R. and Elhadad, N. (2003). Sentence alignment for monolingual comparable corpora. In *Proceedings of the 2003 EMNLP*, pages 25–32.

Biran, O., Brody, S., and Elhadad, N. (2011). Putting it simply: a context-aware approach to lexical simplification. In *Proceedings of the 49th ACL*, pages 496–501.

Bott, S. and Saggion, H. (2011). An unsupervised alignment algorithm for text simplification corpus construction. In *Proceedings of the 2011 MTTG*, pages 20–26.

Burns, P. R. (2013). Morphadorner v2: A java library for the morphological adornment of english language texts. *Northwestern University, Evanston, IL*.

Coltheart, M. (1981). The mrc psycholinguistic database. *The Quarterly Journal of Experimental Psychology*, 33(4):497–505.

Coster, W. and Kauchak, D. (2011). Simple english wikipedia: A new text simplification task. In *Proceedings of the 49th ACL*, pages 665–669.

Denkowski, M. and Lavie, A. (2011). Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the 6th WMT*, pages 85–91. Association for Computational Linguistics.

Devlin, S. and Tait, J. (1998). The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases*, pages 161–173.

Devlin, S. (1999). *Simplifying Natural Language for Aphasic Readers*. Ph.D. thesis, University of Sunderland.

Feblowitz, D. and Kauchak, D. (2013). Sentence simplification as tree transduction. In *Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 1–10.

Glavaš, G. and Štajner, S. (2015). Simplifying lexical simplification: Do we need simplified corpora? In *Proceedings of the 53rd ACL*, pages 63–69.

Graesser, A. C., McNamara, D. S., Louwerse, M. M., and Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments and Computers*, 36(2):193–202.

Horn, C., Manduca, C., and Kauchak, D. (2014). Learning a lexical simplifier using wikipedia. In *Proceedings of the 52nd ACL*, pages 458–463.

Kandula, S., Curtis, D., and Zeng-Treitler, Q. (2010). A

semantic and syntactic text simplification tool for health content. In *Proceedings of the 2010 AMIA*, pages 366–70.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, demonstration session*, Prague, Czech Republic.

Paetzold, G. H. and Specia, L. (2013). Text simplification as tree transduction. In *Proceedings of the 9th STIL*, pages 116–125.

Paetzold, G. H. and Specia, L. (2015). Lexenstein: A framework for lexical simplification. In *Proceedings of The 53rd ACL*, pages 85–90.

Paetzold, G. H. and Specia, L. (2016a). Benchmarking lexical simplification systems. In *Proceedings of the 10th LREC*.

Paetzold, G. H. and Specia, L. (2016b). Inferring psycholinguistic properties of words. In *Proceedings of the 2016 NAACL*, pages 435–440.

Paetzold, G. H. and Specia, L. (2016c). Unsupervised lexical simplification for non-native speakers. In *Proceedings of The 30th AAAI*, pages 3761–3767.

Paetzold, G. H. and Specia, L. (2016d). Vicinity-driven paragraph and sentence alignment for comparable corpora. *arXiv preprint arXiv:1612.04113*.

Paetzold, G., Alva-Manchego, F., and Specia, L. (2017). Massalign: Alignment and annotation of comparable documents. In *Proceedings of the 8th IJCNLP*, pages 1–4.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th ACL*, pages 311–318.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Shardlow, M. (2014). A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications*.

Siddharthan, A. (2006). Syntactic simplification and text cohesion. *Research on Language and Computation*, 4(1):77–109, March.

Smith, J. R., Quirk, C., and Toutanova, K. (2010). Extracting parallel sentences from comparable corpora using document level alignment. In *Proceedings of the 2010 HLT*, pages 403–411.

Štajner, S., Franco-Salvador, M., Ponzetto, S. P., Rosso, P., and Stuckenschmidt, H. (2017). Sentence alignment methods for improving text simplification systems. In *Proceedings of ACL*, pages 97–102.

Woodsend, K. and Lapata, M. (2011). Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 EMNLP*, pages 409–420.

Xu, W., Callison-Burch, C., and Napoles, C. (2015). Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing statistical machine translation for text simplification. *TACL*, 4:401–415.

Yasseri, T., Kornai, A., and Kertész, J. (2012). A practical approach to language complexity: a wikipedia case study. *PloS one*, 7(11):e48386.

Yatskar, M., Pang, B., Danescu-Niculescu-Mizil, C., and Lee, L. (2010). For the sake of simplicity: Unsupervised extraction of lexical simplifications from wikipedia. In *NAACL*, pages 365–368, Los Angeles, California.

Zhang, X. and Lapata, M. (2017). Sentence simplification with deep reinforcement learning. In *Proceedings of EMNLP*, pages 595–605.

Zhu, Z., Bernhard, D., and Gurevych, I. (2010). A monolingual tree-based translation model for sentence simplification. *Computational Linguistics*, (August):1353–1361.

9. Language Resource References

Newsela. (2016). *Newsela Article Corpus*. Version: 2016-01-29, `https://newsela.com/data`.

Intertextual Correspondence for Integrating Corpora

Jacky Visser, Rory Duthie, John Lawrence, Chris Reed
Centre for Argument Technology
University of Dundee, Dundee DD1 4HN, United Kingdom
{j.visser, r.w.duthie, j.lawrence, c.a.reed}@dundee.ac.uk

Abstract

We present intertextual correspondence (ITC) as an integrative technique for combining annotated text corpora. The topical correspondence between different texts can be exploited to establish new annotation connections between existing corpora. Although the general idea should not be restricted to one particular theoretical framework, we explain how the annotation of intertextual correspondence works for two corpora annotated with argumentative notions on the basis of Inference Anchoring Theory. The annotated corpora we take as examples are topically and temporally related: the first corpus comprises television debates leading up to the 2016 presidential elections in the United States, the second corpus consists of commentary on and discussion of those debates on the social media platform Reddit. The integrative combination enriches the existing corpora in terms of the argumentative density, conceived of as the number of inference, conflict and rephrase relations relative to the word count of the (sub-)corpus. ITC also affects the global properties of the corpus, such as the most divisive issue. Moreover, the ability to extend existing corpora whilst maintaining the level of internal cohesion is beneficial to the use of the integrated corpus as resource for text and argument mining based on machine learning.

Keywords: argument, corpus, debate, dialogue, intertextuality, Reddit, US presidential elections

1. Introduction

In many regards, a larger corpus is a better corpus (all other things being equal). Additionally, for some purposes, a diverse corpus covering various communicative genres or data from different modalities can be of great value. In a study of argumentative conduct in television debates leading up to the 2016 presidential elections in the United States, we are looking for both. We need a large corpus for the purpose of training machine learning algorithms for argument mining (Palau and Moens, 2009; Feng and Hirst, 2011; Stab and Gurevych, 2014) – with a size of 97,999 words (tokens), our US2016 corpus is the largest corpus of dialogical argumentation annotated for both inferential and discursive structure. We also want the corpus to be amenable to automated comparison between the characteristics of argumentative discussions in different communicative genres, as a means of quantifying the uptake certain topics from the television debates find in online social media reactions.

To construct a large corpus, it might be sufficient to collate various (preferably consistently annotated) corpora into one super-corpus. However, if any of the studies for which the text corpus is used relies upon the relationships between different parts of the corpus, then mere collation will not suffice. To properly integrate corpora, we propose that intertextual correspondence can be used as a method to preserve dialogical cohesion by functionally connecting content from two or more corpora on the basis of their topical correspondence.

In the present paper, we introduce the notion of 'intertextual correspondence' (ITC) in relation to existing work (Section 2.), we outline the general idea behind ITC and explain how we employ ITC in the creation of the US2016 corpus of election debate material (3.). The integration of both transcripts of television debates between the presidential candidates, and social media reactions to the debates, allows us to exploit the large-scale properties of the corpus to create Argument Analytics that provide new insight into the dynamics of argumentative reality (4.).

2. Related Work

The notion of 'intertextuality', introduced by Kristeva (1977), is one of the cornerstones of the postmodern tradition particularly in literary studies and religion studies (Alfaro, 1996). Intertextuality is used to explain that any text can only be properly understood in its relation to the larger body of other texts. To put it differently: context – broadly conceived – should be taken into account when interpreting (the meaning of) a text.

In our repurposing of the term, we divorce it from its associated philosophical connotations, to refer to the topical correspondence that exists between the contents of independent text corpora. While our focus is exclusively on corpora of argumentative discourse and on the intertextual inference dependencies that exist between them, we do not intend to constrain ITC to such corpora. The corpus which we consider in Section 4. comprises argumentation in the political domain. In her Critical Discourse Analysis studies of political communication, Wodak (2009) makes a distinction between 'intertextuality' and 'interdiscursivity'. Because of the shared focus on topical interconnection, the latter notion is closer to what we mean here with 'intertextual correspondence'.

Connecting various topically related parts of analysed argumentative discourse is directly related to the (World Wide) Argument Web: a vision of "a large-scale Web of interconnected arguments posted by individuals to express their opinions in a structured manner" (Rahwan et al., 2007). The foundation underpinning the Argument Web is the Argument Interchange Format (AIF) (Chesñevar et al., 2006). The AIF is an ontology designed to capture the variety of theory-dependent notions, properties and constructs used in the study of argumentation. As such, the AIF is intended to serve as an interlingua in which the different theoretical

conceptions of argumentatively relevant terms can be translated. Reed et al. (2010) extended the AIF to account for the dialogical dimension of argumentation in the AIF ontology.

3. Intertextual Correspondence

3.1. Integrating Corpora

We propose a method for integrating annotated text corpora on the basis of the correspondences that exist between the topics at issue in each corpus. Of course, this is not to imply that any two corpora can always be integratively merged. The merging is not the issue, two corpora can just be collated into one. Integrating them, however is where the challenge lies. This requires identification of correspondences between elements of the two corpora. Such correspondences can be weaker, e.g., when the same topic is being discussed without further direct communicative relation, or stronger, e.g., when a direct reference is made by means of reported speech to some statement made elsewhere.

The underlying principles of ITC can be used to integratively combine corpora annotated with a variety of theoretical approaches. A large, but partially disjoint corpus such as Abbott et al. (2016)'s Internet Argument Corpus (version 2) which contains content from different online sources could be amenable to the techniques we discuss to strengthen the internal cohesion. In the remainder of this paper, however, we build on the theoretical foundation upon which our corpus annotation is based, with application to the US2016 corpus.

3.2. Corpus Annotation Guidelines

Four annotators were extensively trained in the use of Inference Anchoring Theory (IAT) (Budzynska and Reed, 2011) to analyse the television debates and Reddit discussions constituting the US2016 corpus that we take as a case in point in the current paper. Building on insights from discourse analysis and argumentation studies (van Eemeren et al., 2014), IAT explains argumentative conduct in terms of the anchoring of argumentative reasoning in dialogical interaction. Drawing on Speech Act Theory (Austin, 1962; Searle, 1969), the anchoring is theoretically conceptualised by means of the 'illocutionary connection' between locutions in dialogue and their propositional content. Importantly, IAT complies with the standards of the AIF ontology discussed in Section 2.).

The annotation guidelines, summarised below, are based on IAT. The full version of the guidelines (available online at `arg.tech/US2016-guidelines`) deals with, among others: anaphoric references, epistemic modalities, repetitions, punctuation, discourse indicators, interposed text, reported speech, and how to deal with context-specific peculiarities.

Segmentation divides the (transcribed) text into locutions. A locution consists of a speaker designation and an 'argumentative discourse unit' (a text span with discrete argumentative function) (Peldszus and Stede, 2013).

Transitions capture the functional relationships between locutions, reflecting the dialogue protocol – a high level specification of the set of transition types that are available in a particular communicative activity.

Illocutionary connections embody the intended communicative functions of locutions or transitions, such as: *Agreeing*, *Arguing*, *Asserting*, (three sub-types of) *Challenging*, *Disagreeing*, (three sub-types of) *Questioning*, *Restating*, and *Default Illocuting* (when none of the other types suffice). Some types of illocutionary connection lead to the reconstruction of a propositional content.

Inferences are directed relations between propositions, reflecting that a proposition is meant to supply a reason for accepting another proposition. A specific argument scheme (e.g., *Argument from Example* or *Argument from Expert Opinion*) can be specified; failing that, it is labelled as *Default Inference*.

Conflicts are directed relations between propositions, reflecting that a proposition is meant to be incompatible with another proposition or relation. Such incompatibility may depend on, e.g., logical contradiction or pragmatic contrariness, or the annotated relation may default to *Default Conflict*.

Rephrases are directed relations between propositions, reflecting that a proposition is meant to be a reformulation of another proposition. Such reformulation may involve, e.g., *Specialisation*, *Generalisation* or *Instantiation*, or the relation defaults to *Default Rephrase*.

3.3. Intertextual Correspondence Guidelines

The ITC between the comments on Reddit and the television debates to which they react were annotated by two annotators trained in the application of the instruction described in this section. Because the original US2016 corpus is annotated on the basis of IAT, the annotation guidelines for ITC are generally the same as those described in Section 3.2.. However, the contextual characteristics of the two genres of television debates and social media discussion lead to five annotation patterns that can be expected to typically occur in ITC in particular (although variations are possible). In the explanation of these patterns, we will use the suffixes '-tv', '-reddit', and '-itc' when we refer to the elements of the annotations that are part of, respectively, the television debate sub-corpus (US2016tv), the Reddit discussion sub-corpus (US2016reddit), and the intertextual correspondence sub-corpus that serves to bridge between them (US2016itc).

As part of the annotation of ITC, no new propositional content is annotated. Rather, new relations are introduced that connect content (both propositions and locutions) in the US2016reddit corpus to their counterparts in the television corpus. Due to the temporal ordering of the Reddit commentary following the proceedings in the television debate, by necessity, the transitions and most of the other relations will be directed from US2016reddit to US2016tv (one notable exception will be introduced below).

The first common pattern reflects rephrases of what is said in the television debates. Although propositions can be quoted literally, in most cases there is some degree of reformulation. In these cases, an ITC rephrase relation is introduced. An example of such a rephrase relation is visualised in Figure 1. In this example, Reddit user Mr_Jensen reformulates Bernie Sanders' claim about college affordability in order to explain what the then candidate for the

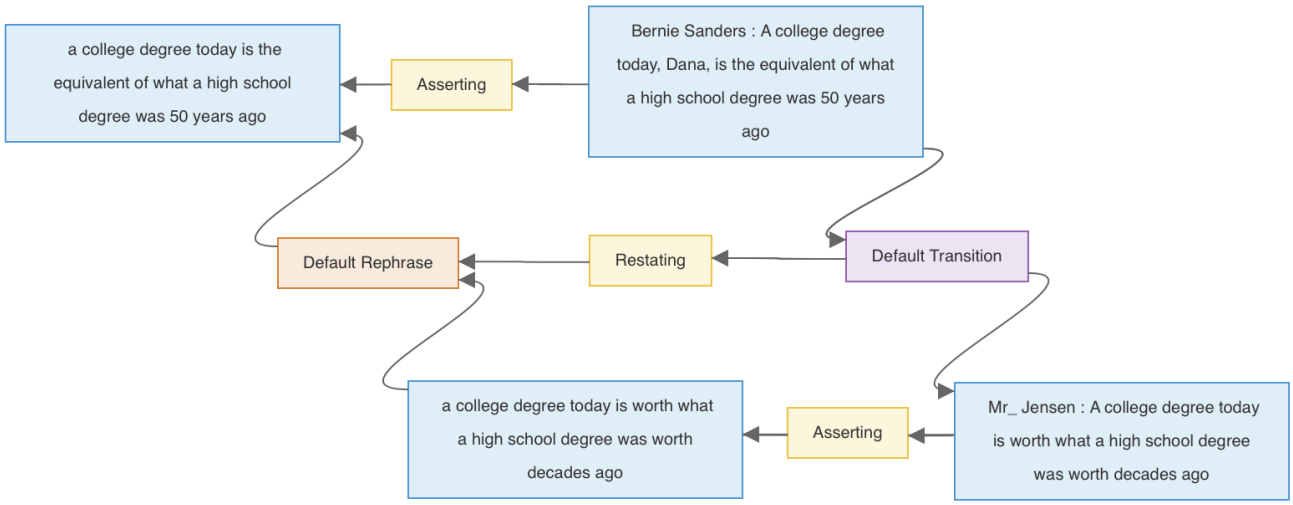


Figure 1: Diagrammatic visualisation of common ITC rephrase pattern.

Democrat's nomination meant. The top row of Figure 1 shows the connections between proposition-tv, illocution-tv and locution-tv; the bottom row shows proposition-reddit, illocution-reddit and locution-reddit; the middle row contains the connecting relations introduced as part of the annotation of ITC. The rephrase-itc relation from proposition-reddit to proposition-tv reflects the intertextual rephrase. The illocution-itc of *Restating* anchors the *Default Rephrase* relation between the two propositions in the *Default Transition* between the two locutions.

The next three common patterns are structurally equivalent to the first, although they introduce different relations between the proposition-reddit and the proposition-tv, and hence employ different illocutionary connections. A Reddit user can engage in an argumentative interaction with the content of the television debate by either supplying (additional) reasons for a claim made during the television debate, or by drawing a conclusion on the basis of what was said on television. In both cases the transition-itc connects locution-reddit to locution-tv, and in both cases the illocution-itc of *Arguing* anchors the inference-itc. In the first case (providing additional reasons), the inference-itc goes from proposition-reddit to proposition-tv. In the second case (drawing a conclusion), this direction is reversed. Similarly to arguing, a Reddit user can express disagreement with what is said on television. In this case, the illocution-itc of *Disagreeing* anchors the conflict-itc in the transition-itc.

Instead of disagreeing, a Reddit user can agree with what is said during the television debate. The resulting structure of the ITC is different from that in Figure 1, because the associated illocution-itc of *Agreeing* does not anchor any propositional relation (of rephrase, inference or conflict). Rather, the annotated ITC only introduces two relations: the by now expected transition-itc from locution-reddit to locution-tv, and an illocution-itc from locution-reddit to proposition-tv.

3.4. Validation

To validate the ITC annotation guidelines, the two annotators both annotated an 11% subset of the corpus. This subset was sampled by taking every second US2016reddit excerpt until the number of propositions was greater than 10% of the overall corpus. The same process was used for the validation of the US2016tv and US2016reddit corpora yielding a Cohen's κ (Cohen, 1960) of 0.73 for the IAT annotation. In the case of ITC annotation, validating using percentage agreement becomes difficult. The high proportion of agreed negatives, due to a low amount of overall connections made, and a high number of possible connections of 178,940 (overall locutions and propositions in US2016tv $\times$ possible illocutionary relations $\times$ possible propositional relations) make validation with this metric impractical as a result of the weight of negative examples (see Table 1).

Annotation	% Agreement	Cohen's κ
Relations	0.99	0.50
Excerpts	0.99	0.83
Combined	0.98	0.62

Table 1: Percentage agreement and Cohen's κ for annotation of ITC-relations, ITC-excerpts and both combined.

Cohen's κ was used for the inter-annotator agreement calculation to take into account chance agreement and lower the weight of the agreed negatives. Although this can effectively give an agreement score for relations, it undermines the complexity of the ITC task due to the need for exact matches between both annotators. In ITC this is particularly difficult due to the high number of possible relations that can be made by an annotator. A more effective measure is employed which first considers that two excerpts, one from television and one from Reddit, are agreed to have an ITC relation connecting them by both annotators (see excerpts in Table 1). By using κ we account for chance agreement between the annotators and therefore the evalu-

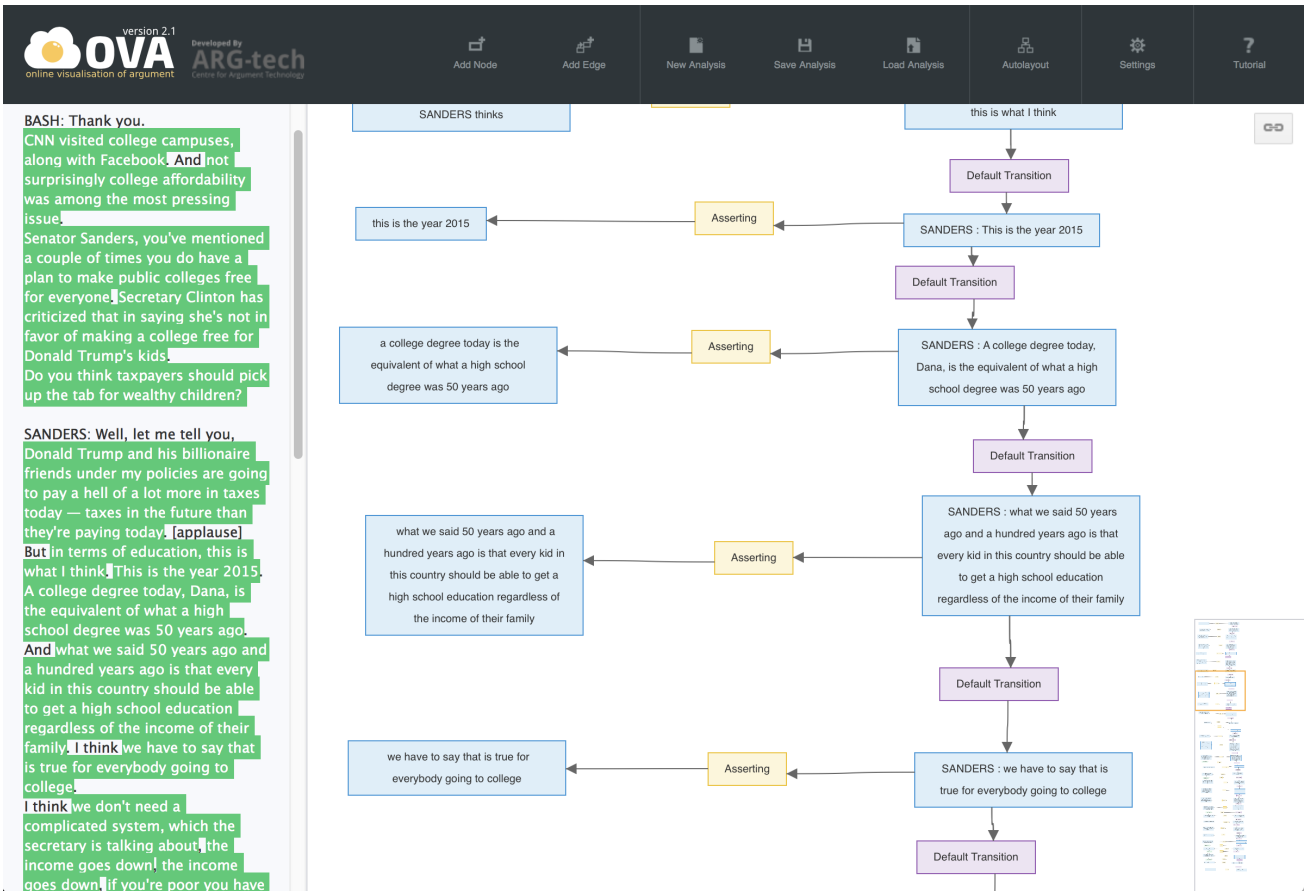


Figure 2: IAT analysis in OVA of an excerpt of US2016tv.

ation is not skewed by false negatives like in the percentage agreement calculation. Following this, κ is used to evaluate the relations made between two excerpts with a more harsh agreement policy where all locutions, propositions, and ITC relations must be identical (see relations in Table 1). Finally, to give an overall κ value we combine the results from the excerpt linking task and the ITC labelling task using the harmonic mean which gives a κ score of 0.62 ((see combined in Table 1)) seen as "good" agreement by (Landis and Koch, 1977).

3.5. Software tools

In the annotation of the US2016 corpus, annotators use an infrastructure of software tools for the manipulation and archiving of argumentation. For the IAT annotation of the television debates and the Reddit discussions, annotators use the Online Visualisation of Argument (OVA) tool (see Figure 2 and available online at `ova.arg.tech`) (Janier et al., 2014). OVA allows raw natural language text to be highlighted (left of Figure 2) to create locutions, propositions and illocutionary connections in an IAT diagram when the canvas on the right is clicked. Relations between locutions or propositions can be created by toggling the add edge button on the OVA menu and can be edited by double clicking. The full IAT diagram is displayed in the canvas and can be navigated through a small overview box (see bottom right of Figure 2).
The annotation of ITC is facilitated by the ITClinker tool available at `arg.tech/ITClinker`. Figure 3 shows a screenshot of the online tool, which is still in the academic prototype stage. Two IAT diagrams are loaded side by side, and locutions and propositions can be selected to create ITC relations. After selection, the user is prompted to decide on the relation type which is displayed below the two IAT diagrams. There is also an option to delete ITC relations.
The finished annotations are stored in AIFdb (Lawrence et al., 2012), an online repository of analysed arguments (available at `aifdb.org`). AIFdb Corpora (Lawrence and Reed, 2014) is used to manage the annotated corpora, and now includes more than 130 corpora of varying size and purpose (e.g. to support education, for particular academic studies, or for our collaboration with the BBC) (`corpora.aifdb.org`). The global argumentative and discursive properties of a corpus can be automatically extracted by using Argument Analytics (`analytics.arg.tech`) (Lawrence et al., 2016).

4. Intertextual Correspondence between 2016 US Election TV Debates and Reactions on Reddit

4.1. The US2016 Election Debate Corpus

The US2016 corpus comprises annotated texts from two genres: transcripts of television debates leading up to the 2016 US presidential elections, and online reactions to those debates on the Reddit social media plat-

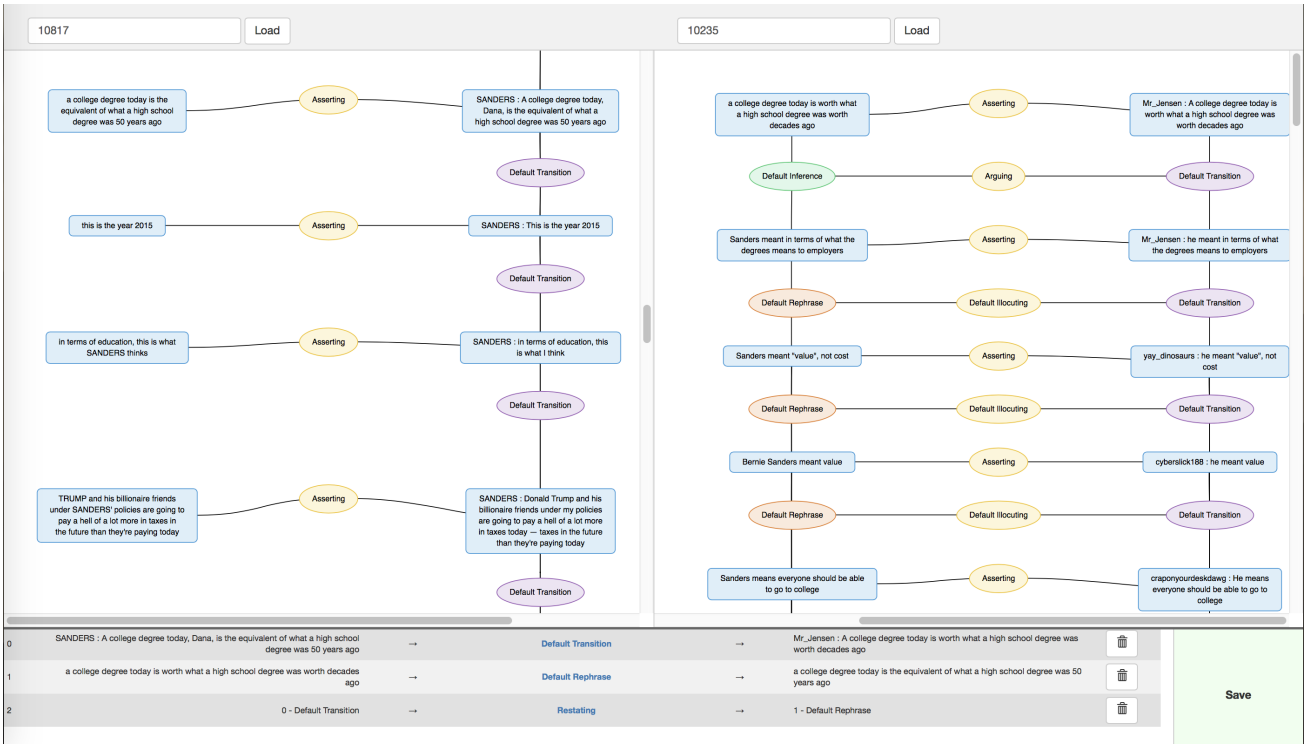


Figure 3: ITC annotation with ITClinker between graph-based IAT annotations from US2016tv and US2016reddit, showing the ITC relations connecting them underneath.

form (`reddit.com`). The full US2016 corpus is freely available online at `corpora.aifdb.org/US2016`, and contains a television debate sub-corpus (US2016tv), and a Reddit sub-corpus (US2016reddit) – both independently available online at respectively `corpora.aifdb.org/US2016tv` and `corpora.aifdb.org/US2016reddit`.

The US2016tv sub-corpus comprises annotated transcripts of three television debates leading up to the 2016 US presidential elections: the first Republican primaries debate (Peters and Woolley, 2015b), the first Democrat primaries debate (Peters and Woolley, 2015a), and the first general election debate between Hilary Clinton and Donald Trump (Peters and Woolley, 2016). The US2016reddit sub-corpus comprises manually retrieved excerpts from the Reddit mega-threads on the topic of the respective television debates (a new mega-thread being opened during the debates every 30 minutes), selecting sub-threads corresponding to time windows of increased dialogical interaction in the television debate while excluding topically irrelevant sub-threads (e.g. those about practical issues of a technical nature).

The IAT annotation of the US2016tv and sub-corpora (see Section 3.2.) means that the corpus is annotated with dialogical structures consisting of locutions and illocutionary connections, as well as with argumentative relations between the propositional contents of the locutions. Table 2 shows the total counts of annotated illocutionary connections and propositional relations (of inference, conflict and rephrase) in the various (sub-)corpora.

Corpus	Inference	Conflict	Rephrase	Illocution
US2016tv	1551	194	333	6617
US2016reddit	1203	629	287	6348
US2016itc	76	119	144	366
US2016	2830	942	764	13331

Table 2: Propositional relation and illocutionary connection counts for the US2016 corpus.

Corpus	Most divisive proposition
US2016D1tv	Bernie Sanders is tough enough on guns.
US2016D1reddit	Bernie Sanders is really doing that "bad" as the comments suggest.
US2016D1	Bernie Sanders wants Colleges to get their costs down.

Table 3: Most divisive issues for US2016D1tv, US2016D1reddit, and US2016D1 including ITC.

4.2. Intertextual Correspondence in US2016

The election debates and the online reactions are in many ways entirely different argumentative activities: one is broadcast on live television, the other is a social media platform with user-generated content, there is no overlap in participants, and both have their own genre conventions. The two constitutive parts of our corpus are however re-

lated both topically and temporally: the raison d'être of the Reddit threads is to discuss what happens in the television debate, and this online commentary is happening while the election debate is being broadcast (in a typical second-screen fashion). This makes the US2016 corpus very suitable for integrative annotation with ITC.

Before ITC annotation, the US2016 corpus contains a total of 4,197 argumentative relations between propositions (2,754 inferences, 823 conflicts, 620 rephrases) and 12,965 illocutionary connections. Table 2 shows the increase of interconnectivity between the television and Reddit subcorpora of US2016.

As well as increasing the overall counts of propositional relations within the US2016 corpus, ITC also provides a more complete overview of a debate and unifies the issues discussed. This effect on the corpus' global properties becomes evident when calculating, for example, the 'divisiveness' measure. This graph-based Argument Analytic allows the automatic calculation of the most divisive issues within a corpus (Konat et al., 2016). In Table 3, the most divisive issue is shown for the first Democratic primaries debate (US2016D1tv), for the corresponding reactions on Reddit (US2016D1reddit), and for the compilation of all material related to the first Democratic primaries debate (US2016D1) – television debate, Reddit discussion and ITC. While in all three cases the most divisive issue is related to Bernie Sanders, it is clear that the addition of cross-corpus relations of inference, conflict and rephrase by means of ITC results in a shift; in this case moving from opinions about Bernie Sanders himself to a more general divisive issue.

5. Conclusion

Intertextual correspondence (ITC) provides a powerful method of increasing the value of existing annotated corpora. We have demonstrated how ITC is used in our US2016 corpus of US presidential election debates and associated reactions on Reddit. By integrating previously independent corpora, ITC enables the connecting of constellations of (argumentative) content in different corpora to get a better perspective of the interplay between reasons for and against claims in general, regardless of the genre, context or time period in which the claim was made. ITC can be used to connect discourses, establishing detailed, specific and computational meaning to the notions of intertextuality and context. Our initial work has shown how three televised debates and the live reaction to them on Reddit can be connected together not only to build a much larger corpus of dialogue on the topic, but more importantly, to allow investigation of phenomena such as popularity and divisiveness that hinge upon this connection being well established. In the longer term, ITC will allow connections between more distant corpora that are thematically or contextually related, enabling more thorough and wide-ranging computational modelling of complex debates in which discussion takes place across a diverse set of fora: climate change discussions, for example, that take place partly in the scientific literature, partly in popular science media, partly in the political sphere, partly on social media, and so on. ITC offers a key step towards being able to model these debates in their entirety allowing audiences and contributors (whether, in this case, scientists, politicians or the public) access to the entire debate in a structured and navigable format. In this way, ITC forms a critical part in the current effort to realise the vision of the Argument Web (Rahwan et al., 2007).

6. Acknowledgements

This research was supported by the Engineering and Physical Sciences Research Council in the UK under grants EP/M506497/1 and EP/N014871/1.

7. Bibliographical References

Abbott, R., Ecker, B., Anand, P., and Walker, M. (2016). Internet Argument Corpus 2.0: An SQL schema for dialogic social media and the corpora to go with it. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4445–4452, Paris, France. European Language Resources Association (ELRA).

Alfaro, M. J. M. (1996). Intertextuality: Origins and development of the concept. *Atlantis*, 18(1/2):268–285.

Austin, J. L. (1962). *How to Do Things with Words*. Clarendon Press.

Budzynska, K. and Reed, C. (2011). Whence inference. Technical report, University of Dundee.

Chesñevar, C., Modgil, S., Rahwan, I., Reed, C., Simari, G., South, M., Vreeswijk, G., Willmott, S., et al. (2006). Towards an argument interchange format. *The Knowledge Engineering Review*, 21(04):293–316.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Feng, V. W. and Hirst, G. (2011). Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 987–996. Association for Computational Linguistics.

Janier, M., Lawrence, J., and Reed, C. (2014). OVA+: An argument analysis interface. In S. Parsons, et al., editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, pages 463–464, Pitlochry. IOS Press.

Konat, B., Lawrence, J., Park, J., Budzynska, K., and Reed, C. (2016). A corpus of argument networks: Using graph properties to analyse divisive issues. In *LREC 2016*, pages 3899–3906. European Language Resources Association.

Kristeva, J. (1977). Word, dialogue and novel. In Leon S. Roudiez, editor, *Desire in language: A semiotic Approach to Literature and Art*, pages 64–91. Columbia University Press.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Lawrence, J. and Reed, C. (2014). AIFdb corpora. In S. Parsons, et al., editors, *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, pages 465–466, Pitlochry. IOS Press.

Lawrence, J., Bex, F., Reed, C., and Snaith, M. (2012). AIFdb: Infrastructure for the Argument Web. In *Proceedings of the Fourth International Conference on Computational Models of Argument (COMMA 2012)*, pages 515–516.

Lawrence, J., Duthie, R., Budzynska, K., and Reed, C. (2016). Argument Analytics. In *the Sixth International Conference on Computational Models of Argument (COMMA 2016)*, pages 371–378.

Palau, R. M. and Moens, M.-F. (2009). Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM.

Peldszus, A. and Stede, M. (2013). From argument diagrams to argumentation mining in texts: a survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 7(1):1–31.

Peters, G. and Woolley, J. T. (2015a). Democratic candidates debate in Las Vegas, Nevada, October 13, 2015. Accessed 11 Aug. 2017.

Peters, G. and Woolley, J. T. (2015b). Republican candidates debate in Cleveland, Ohio, August 6, 2015. Accessed 11 Aug. 2017.

Peters, G. and Woolley, J. T. (2016). Presidential debate at Hofstra University in Hempstead, New York, September 26, 2016. Accessed 11 Aug. 2017.

Rahwan, I., Zablith, F., and Reed, C. (2007). Laying the foundations for a world wide argument web. *Artificial Intelligence*, 171:897–921.

Reed, C., Wells, S., Budzynska, K., and Devereux, J. (2010). Building arguments with argumentation: the role of illocutionary force in computational models of argument. In P. Baroni, et al., editors, *Proceedings of the 3rd International Conference on Computational Models of Argument (COMMA 2010)*, pages 415–426. IOS Press.

Searle, J. R. (1969). *Speech acts: An essay in the philosophy of language*. Cambridge University Press.

Stab, C. and Gurevych, I. (2014). Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 46–56, Doha, Qatar. Association for Computational Linguistics.

van Eemeren, F. H., Garssen, B., Krabbe, E. C. W., Snoeck Henkemans, A. F., Verheij, B., and Wagemans, J. H. M. (2014). *Handbook of argumentation theory.* Springer.

Wodak, R. (2009). *The discourse of politics in action. Politics as usual.* Palgrave Macmillan.

A Gold Anaphora Annotation Layer on an Eye Movement Corpus

Olga Seminck, Pascal Amsili
Laboratoire de Linguistique Formelle (Université Paris Diderot & CNRS)
olga.seminck@cri-paris.org, pascal.amsili@linguist.univ-paris-diderot.fr

Abstract

Anaphora resolution is a complex process in which multiple linguistic factors play a role, and this is witnessed by a large psycholinguistic literature. This literature is based on experiments with hand-constructed items, which have the advantage to filter influences outside the scope of the study, but, as a downside, make the experimental data artificial. Our goal is to provide a first resource allowing to study human anaphora resolution on natural data. We annotated anaphorical pronouns in the Dundee Corpus: a corpus of $\sim$ 50k words coming from newspaper articles read by humans of whom all eye movements were recorded. We identified all anaphoric pronouns — in opposition to non-referential, cataphoric and deictic uses — and identified the closest antecedent for each of them. Both the identification of the anaphoricity and the antecedents of the pronouns showed a high inter-annotator agreement. We used our resource to model reading time of pronouns to study simultaneously various factors of influence on anaphora resolution. Whereas the influence of the anaphoric relation on the reading time of the pronoun is subtle, psycholinguistic findings from settings using experimental items were confirmed. In this way our resource provides a new means to study anaphora.

Keywords: Dundee Corpus, Personal Pronouns, Mixed Effects Modeling, Eye-Tracking

1. Introduction

How the human mind interprets pronouns has been a subject studied in psycholinguistics for a long time now. Many linguistic features have been shown to be of influence. They were discovered using experimental setup, for example by using ambiguous sentences and asking participants to choose an interpretation:

(1) Fred loves Pete, because he is always smiling.

In (1), people could be asked who the pronoun *he* refers to, and most would answer *Pete*, whereas in fact both *Fred* and *Pete* are possible answers. The preference for *Pete* depends mostly in this case on the verb *loves*, which belongs to the family of *implicit causality verbs* (Garvey and Caramazza, 1974). When reading (1), people automatically wonder what is so lovely about Pete. By manipulating different features in experiments (e.g. syntactic role, semantic role, information structure, implicit causality), many factors of influence were discovered. The effect of some factors is quite robust and was confirmed by studies using various experimental methods. However, we are not aware of any study on these factors that did not used hand-constructed experimental items. This is a reason why we wanted to test whether factors of influence on anaphora resolution would also show in other settings, like natural text reading. Finding an influence of these factors would strengthen the proof of their robustness, because —unlike in an experimental setting— unwanted properties cannot be filtered out. As natural data we used the English part of the Dundee Corpus: a resource of eye movement data from ten participants reading newspaper articles. We annotated all anaphoric personal pronouns in this corpus by identifying their antecedent. We then used the reading times from the corpus combined with our anaphora annotation to study the time course of the resolution of anaphorical pronouns. With a first series of models we focused particularly on the influence of distance between the anaphor and the antecedent, frequency of the antecedent and grammatical function of the antecedent and of the anaphor. In the remainder of this introduction we say a few words about these three factors; in section 2., we present our annotation layer and various measures of its quality; in section 3., we present a series of linear models we used, that show the benefits that can be drawn from our resource.

1.1. Distance

The distance between the pronoun and the antecedent is assumed to play an important role in pronoun resolution. Theories providing a saliency account for pronoun resolution, such as *Centering Theory* (Grosz et al., 1983; Grosz et al., 1995) or *Accessibility Theory* (Ariel, 1988; Ariel, 1991) state that (all other things being equal) a shorter distance between the pronoun and the antecedent makes the latter more *salient*. This means that the antecedent is easier to retrieve from memory. The influence of distance was studied in various psycholinguistic experiments (Clark and Sengul, 1979; Ehrlich and Rayner, 1983, e.g.) that confirmed that a longer distance increases comprehension, or reading times.

1.2. Frequency

When a pronoun is processed, the antecedent has to be recovered and its lexical frequency is of influence. However, several psycholinguistic studies found different effects of the frequency of the antecedent, in line with competing theories on the effect of frequency (Van Gompel and Majid, 2004). One, called *Full Reaccess Hypothesis*, predicts that the reactivation of the antecedent when the pronoun is read is very similar to normal lexical access during reading (Shillcock, 1982). Hence, an infrequent antecedent will provoke longer reading times. The other is called the saliency account and predicts that antecedents with lower lexical frequency are more salient (better marked) and therefore easier to recover, evoking shorter reading times (Pynte and Colonna, 2000).

1.3. Grammatical Function

Grammatical functions seems to play an important role in anaphoric resolution: on one hand, the ease of retrieving an antecedent seems to depend on its grammatical function. Many experiments —at least for English— found faster processing when the antecedent was in the subject position (Broadbent, 1973; Clancy, 1980; Frederiksen, 1981; Hobbs, 1976). On the other hand, the comparison of the function of the pronoun and its antecedent is also influential: an effect that has frequently been found is a faster processing of the pronoun when the anaphor and the antecedent have the same syntactic function (Maratsos, 1973; Sheldon, 1974; Smyth, 1994). We leave aside this last factor in this paper.

2. Language Resource

Our resource's name is APADEC: *Anaphorical Pronouns and their Antecedents in the Dundee Eye-Tracking Corpus* (Seminck and Amsili, 2017)[1]. It is an annotation layer on the English part of the Dundee Corpus (Kennedy and Pynte, 2005) which records eye-movement measures for English and French. Ten participants read 20 articles[2] of *The Independent* for a total of 51 502 tokens, 9 776 types and 2 368 sentences (Barrett et al., 2015). In total there are 2 123 personal pronouns in the corpus, from which 1 109 were labelled anaphoric (the other 1 014 being either deictic, non-referential, cataphoric, or having a split antecedent).

2.1. Annotation

We chose to have the corpus manually annotated by two annotators. We searched for instances of personal pronouns using a part-of-speech tag annotation of the corpus provided by Frank and others (2009) and Barrett et al. (2015). We then selected the pronouns that were used anaphorically, and annotated for each of them the closest mention of the antecedent, by marking its span of words (Table 1).

Table 1: An antecedent annotation example from the corpus

word nb	word form	POS	antecedent
1800	if	[IN]	
1801	the	[DT]	
1802	voters	[NNS]	
1803	did	[VBD]	
1804	not	[RB]	
1805	care	[VB]	
1806	about	[IN]	
1807	that,	[DT, ,]	
1808	they	[PRP]	1801-1802
...	...	...	...

First and second person pronouns were considered as non-anaphoric, as they have a deictic function. We did not annotate the referents of cataphoric pronouns and we also excluded split antecedent anaphoric pronouns from our data. We used the following procedure for annotation:

- Annotator 1 annotated the entire corpus.
- Annotator 2 was instructed by Annotator 1 and annotated separately 36 232 words of the corpus.
- Annotator 1 and 2 compared their annotations and decided upon all cases they did not agree on.
- Annotator 1 corrected the $\sim$ 15 000 remaining words of the corpus for mistakes.

2.2. Evaluation

To evaluate the inter-annotator agreement for distinguishing anaphoric from non-anaphoric pronouns, we used Cohen's κ, a measure of agreement adjusted for chance (Cohen, 1960; Artstein and Poesio, 2008). We found $\kappa = 0.88$ (Pedregosa et al., 2011) for the 36 232 words Annotator 1 and 2 annotated separately. This indicates a very good agreement. Second, we evaluated the identification of antecedents. Because this task consisted in giving the span of words that corresponds to the antecedent, there is no obvious set of labels available for this task, and therefore Cohen's κ is an inadequate measure. Even, if it were possible to consider every possible span of words in a text as a potential label, this does not resolve the problem that the spans two annotators identify can overlap, without being exactly the same. A metric that can handle non-categorical data is Krippendorff's α (Krippendorff, 1980). α is 1 minus the ratio between observed and expected agreement. Passonneau (2004), Passonneau (2006) and Artstein and Poesio (2008) propose various ways to adapt Krippendorff's α to the situation in which labels are sets. Disagreement can be quantified by various distance metrics to account for set similarity. See Table 2. We applied the α-metric to our data using the implementation provided in the NLTK-library (Bird et al., 2009), considering antecedent spans as sets of words. The scores are given in Table 2.

			α
binary distance	d_b =	0 if $s_1 = s_2$ 1 if $s_1 \neq s_2$	0.71
Jaccard distance	d_j=	$1 - \frac{\lvert s_1 \cap s_2 \rvert}{\lvert s_1 \cup s_2 \rvert}$	0.78
MASI-distance	d_M =	$d_j * M$, with $M =$ 0 if $s_1 = s_2$; $\frac{1}{3}$ if $s_1 \subset s_2$ or $s_2 \subset s_1$; $\frac{2}{3}$ if $s_1 \cap s_2 \neq \emptyset$ but $s_1 \not\subset s_2$ & $s_2 \not\subset s_1$; 1 if $s_1 \cap s_2 = \emptyset$	0.75

Table 2: Values of Krippendorff's α given various distances. s_1 and s_2 are sets of words

It is often assumed that $\alpha > 0{,}67$ is enough to support "cautious conclusions". In that light our annotation seems rather reliable. However, Passonneau (2006) and Artstein and Poesio (2008) warn that this is not a hard value and that it is heavily dependent on the data. We therefore also measured the reliability of the annotations by comparing both annotations to the final gold standard of our corpus. In the field of anaphora resolution, the information retrieval metrics of precision and recall (see Table 3) are often used to measure the quality of coreference chains (Vilain et al., 1995; Artstein and Poesio, 2008). For every anaphoric pronoun in our corpus, we calculated the precision and the recall by comparing the annotations of both annotators to the

[1] Freely available: `http://www.llf.cnrs.fr/apadec`

[2] It seems rather to be 20 sessions of reading, each session containing multiple articles.

gold standard. Every word in the annotated span that was also in the gold span was counted as a *true positive (tp)*. If a word occurred in the annotated span, but not in the gold span, it was treated as a *false positive (fp)*. A word that was in the gold span, but not in the annotated span was counted as a *false negative (fn)*. Table 3 gives the mean precision and recall of all the anaphoric pronouns in the corpus. Both annotators seem to obtain good scores. The differences between the scores can be explained by the fact that Annotator 2 sometimes annotated the right antecedent, but not its closest mention. Another reason is that Annotator 2 is an undergrad student, whereas Annotator 1 is a PhD student in linguistics.

	precision $\frac{tp}{tp+fp}$	recall $\frac{tp}{tp+fn}$	F_1
Annotator 1	0,93	0,92	0,93
Annotator 2	0,83	0,81	0,82

Table 3: precision/recall, annotators *vs.* Gold Standard.

3. Experiment

In this experiment we show how reading times for anaphoric pronouns can be modeled using our resource. Modeling reading times for pronouns is not straightforward, because they are often not fixated (Rayner, 1998). But this does not necessarily mean they are not read and processed. Kennedy and Pynte (2005) found evidence that unfixated words can be processed — at least on a lexical level — when they occur in the parafoveal vision. Another question is whether pronominal anaphora are processed the moment they are fixated in foveal (or parafoveal) vision, or whether the processing takes place later on in time. In their studies on pronoun reading, Ehrlich and Rayner (1983) and Van Gompel and Majid (2004) concluded that the retrieving process of the antecedent is initiated where the pronoun is encoded (a fixation on the pronoun itself, or a fixation very near to the pronoun on an adjacent word), but that the processing can be continued later.

3.1. Regions, Reading Time and Preprocessing

We used six regions to study the processing of the anaphoric pronoun: the word before, the pronoun itself and four words following. The word before the pronoun was included, because the pronouns we studied are often so short that they can be read in parafoveal vision when the word before the pronoun is fixated. For this first exploration of our corpus, we studied *first pass reading time*, a very commonly used measure. First pass reading time is the sum of all fixations from the moment a region is entered for the first time, until the eyes move to another region. To prepare for the modeling, we followed previous studies on the Dundee Corpus (Demberg and Keller, 2008; Frank and Bod, 2011) to clean the data. Fixations on words that had punctuation, or clitics attached were eliminated. We also eliminated fixations on words with capital letters. For our six regions we had between 2 593 and 4 173 data points left per region. Two other preprocessing steps we applied were a log-transformation on the reading times and scaling on numeric variables.

3.2. Mixed Effects Model

We used mixed effect models from the lme4 R-package (Bates et al., 2015) to study the factors having an influence on pronoun resolution. For each region a new model was made.

3.3. Factors

We can distinguish three types of factors in our study: random factors, control factors and the factors special to anaphoric relationships we introduced before (distance, frequency and grammatical role). Participants and instances of anaphoric pronouns are modeled with a random intercept. Our control factors are factors that are known to be of influence on reading times, such as word frequency and word length. Following other studies (Demberg and Keller, 2008; Frank and Bod, 2011) on the Dundee Corpus, we used the following control factors: for forward and backward probability from an n-gram model (Frank and Bod, 2011), length in characters of the region, log-frequency of the word in the region in the corpus itself and in the British National Corpus[3], log-frequency of the word previous to the region in the same two corpora and finally the launch and the landing position of the fixation with repect to the number of characters in the word in the region. The control factors are local for each region, in opposition to the factors of distance, frequency and grammatical function. These factors are constant over the six regions, because they consider the anaphoric relation that is not marked specifically on one of the words of the six regions. These factors are briefly presented below:

- dist-ant-begin: the distance in words between the antecedent and the beginning of the text
- same-sent-as-ana: the anaphor and the antecedent are in the same sentence
- dist-ant-ana-words: distance in words between the anaphor and its antecedent
- log-freq-dundee-head-ant: the log frequency of the syntactic head of the antecedent in the Dundee Corpus
- log-freq-bnc-head-ant: the log frequency of the syntactic head of the antecedent in the British National Corpus
- syntactic-role-ana: the grammatical function of the pronoun (subject, direct object or other)
- syntactic-role-head-of-antecedent: the grammatical function of the head of the antecedent (subject, direct object or other)

3.4. Results and Discussion

The results of the models can be found in Table 4. Each column under a region — 0 for the pronoun — represents a model for that region. The numbers for each factor are the model's coefficient estimate for that factor. A positive estimate indicates that a higher score for the factor increases the reading time. Collinearity did not play a major role within our pronoun-related factors. We will shortly discuss the distance, frequency and grammatical function factor below. Due to limited space here we will not discuss the

[3]The BNC counts were taken from: `https://www.kilgarriff.co.uk/bnc-readme.html`

Table 4: Models for the six regions for prediction first pass reading time.

Region	-1		0		1		2		3		4	
(Intercept)	5.481	***	5.350	***	5.385	***	5.399	***	5.411	***	5.412	***
forward probability	0.001		0.018		-0.009		0.002		-0.001		-0.004	
backward probability	-0.003		0.007		0.044	***	0.010		0.011		-0.008	
length in characters	-0.008		0.000		-0.021	**	-0.005		0.018	*	-0.005	
log-freq-dundee	-0.025	**	-0.014		-0.008		-0.002		0.005		-0.011	
log-freq-dundee previous word	-0.009		0.003		-0.015		0.024		-0.004		0.002	
log-freq-bnc	0.010		-0.011		-0.006		-0.019	*	0.010		0.001	
log-freq-bnc previous word	-0.110	**	-0.006		-0.013		-0.025	*	-0.001		0.002	
launch-pos-first-fix	-0.013	*	-0.011	*	-0.016	***	-0.012	*	-0.009		-0.008	
land-pos-first-fix	0.004		0.004		-0.002		0.005		-0.002		-0.009	
Distance Factors												
dist-ant-begin	-0.009		-0.001		0.011	*	0.004		-0.001		-0.001	
same-sent-as-ana True	-0.023		0.009		-0.013		-0.013		-0.015		-0.019	
dist-ana-ant-words	-0.010	*	0.005		0.002		0.003		0.009	*	0.005	
Frequency Factors												
log-freq-dundee-head-ant	0.002		0.005		0.006		0.006		-0.014		-0.022	*
log-freq-bnc-head-ant	0.001		0.003		-0.002		-0.000		0.010		0.016	
Grammatical Function Factors												
syntactic-role-ana subj	0.022		-0.007		-0.021		-0.049	*	-0.022		-0.014	
syntactic-role-ana other	0.005		0.006		-0.002		-0.006		-0.031		-0.026	
syntactic-role-head-of-antecedent subj	-0.008		-0.007		0.030		0.012		0.019		0.007	
syntactic-role-head-of-antecedent other	0.002		0.002		0.047	*	0.003		0.028		-0.015	

Significance : *** for $p < 0.001$; ** for $p < 0.01$ and * for $p < 0.05$.

control factors. Let's first have a look at the distance factors. It seems that a higher distance between the antecedent and the beginning of the text matters: antecedents early in the text are retrieved faster than those further in the text (see Table 4, region 1, $p < 0.05$). This could be explained by mechanisms such as the *first mention preference* that attributes greater saliency to early mentioned antecedents. There is also an effect of the distance between the pronoun and its antecedent. We observe that the word before the pronoun is read faster when the distance is longer (distance antecedent & pronoun, region -1, $p < 0.05$) and that the reading slows down later (region 3, $p < 0.05$). We think this means that the resolution of pronouns at a longer distance is delayed. Pronouns that are close to their antecedent could be resolved more immediately. When we have a look at our two frequency factors, we see that only the factor that takes into account the frequency of head of the antecedent in the Dundee Corpus is significant (region 4, $p < 0.05$) and the frequency of the head of the antecedent in the British National Corpus does not show significant effects. This result has a negative estimate, thus indicates that a higher frequency in the Dundee Corpus leads to faster reading times. However, we think that this finding cannot be interpreted as a result in favor of a theory that states that memory retrieval of the antecedent is similar to normal lexical retrieval. The frequencies in the Dundee Corpus are not very representative of general word-frequencies of the English language, because of the small size of the corpus. Therefore, we suggest that the effect is driven rather by the presence of the antecedent in a particular text. When we consider grammatical function we see that when the pronoun has a subject function, there is a speed up in region 2 ($p < 0.05$), in line with the saliency account, suggesting that pronouns are expected in the subject position. Finally, when looking at the grammatical function of the antecedent, we see that antecedents in a position other than the subject or object are processed slower (region 1, $p < 0.05$). This last result is also in line with the saliency account that states that subjects and direct objects are easier to retrieve from memory than other grammatical functions. With our first series of models we demonstrate how our resource APADEC can be used to study anaphoric pronoun resolution in a natural setting. The patterns we find in the results may serve to tease apart various psycholinguistic theories. We plan to use our resource to test more factors of influence, including also less-well studied factors, in the near future.

4. Conclusion

Many factors play a role in pronoun interpretation. The field of psycholinguistics revealed these factors in an experimental setting. Yet, the study of the interaction of these factors in a natural setting is new. Our resource APADEC — an annotation layer of anaphorical pronouns in the Dundee Eye-Tracking Corpus — provides a means to take a first step in this direction. Using the data from APADEC, we built mixed effect models to predict the first pass reading time. In our data we find significant effects of Distance, Frequency and Grammatical Role. The effects of the anaphoric relation on reading times are subtle, but despite this fact we were able to obtain significant results, showing the robustness of the effects and confirming psycholinguistic literature.

5. Acknowledgements

We thank Amandine Martinez for her help with the annotation. We thank Saveria Colonna, Barbara Hemforth and Yair Haendler for their helpful suggestions and comments. This work was supported by the Labex EFL (ANR-10- LABX-0083) and the Ecole Doctorale Frontières du Vivant (FdV) — Programme Bettencourt.

6. Bibliographical References

Ariel, M. (1988). Referring and accessibility. *Journal of linguistics*, 24(01):65–87.

Ariel, M. (1991). The function of accessibility in a theory of grammar. *Journal of Pragmatics*, 16(5):443–463.

Artstein, R. and Poesio, M. (2008). Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Barrett, M., Agić, Ž., and Søgaard, A. (2015). The Dundee treebank. In *The 14th International Workshop on Treebanks and Linguistic Theories (TLT 14)*.

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.

Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Broadbent, D. E. (1973). *In defence of empirical psychology*. Methuen.

Clancy, P. M. (1980). Referential choice in english and japanese narrative discourse. *The pear stories: Cognitive, cultural, and linguistic aspects of narrative production*, 3:127–201.

Clark, H. H. and Sengul, C. (1979). In search of referents for nouns and pronouns. *Memory & Cognition*, 7(1):35–41.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Demberg, V. and Keller, F. (2008). Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210.

Ehrlich, K. and Rayner, K. (1983). Pronoun assignment and semantic integration during reading: Eye movements and immediacy of processing. *Journal of Verbal Learning and Verbal Behavior*, 22(1):75–87.

Frank, S. L. and Bod, R. (2011). Insensitivity of the human sentence-processing system to hierarchical structure. *Psychological science*, 22(6):829–834.

Frank, S. L. et al. (2009). Surprisal-based comparison between a symbolic and a connectionist model of sentence processing. In *Proceedings of the 31st annual conference of the cognitive science society*, pages 1139–1144. Cognitive Science Society Austin, TX.

Frederiksen, J. R. (1981). Understanding anaphora: Rules used by readers in assigning pronominal referents. *Discourse Processes*, 4(4):323–347.

Garvey, C. and Caramazza, A. (1974). Implicit causality in verbs. *Linguistic inquiry*, pages 459–464.

Grosz, B. J., Joshi, A. K., and Weinstein, S. (1983). Providing a unified account of definite noun phrases in discourse. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics*, pages 44–50.

Grosz, B. J., Weinstein, S., and Joshi, A. K. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational linguistics*, 21(2):203–225.

Hobbs, J. (1976). Pronoun resolution. research report76-1. New York: Department of computer science. *City University of New York*.

Kennedy, A. and Pynte, J. (2005). Parafoveal-on-foveal effects in normal reading. *Vision research*, 45(2):153–168.

Krippendorff, K. (1980). *Content analysis: An introduction to its methodology*. Sage Publications.

Maratsos, M. P. (1973). The effects of stress on the understanding of pronominal co-reference in children. *Journal of Psycholinguistic Research*, 2(1):1–8.

Passonneau, R. J. (2004). Computing reliability for coreference annotation. In *Proceedings of the 4th international conference on language resources and evaluation (LREC)*.

Passonneau, R. (2006). Measuring agreement on set-valued items (masi) for semantic and pragmatic annotation. *Proceedings of the 5th international conference on language resources and evaluation (LREC)*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pynte, J. and Colonna, S. (2000). Decoupling syntactic parsing from visual inspection: The case of relative clause attachment in french. In Alan Kennedy, et al., editors, *Reading as a perceptual process*, pages 529–547. Elsevier, Amsterdam.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372.

Sheldon, A. (1974). The role of parallel function in the acquisition of relative clauses in english. *Journal of verbal learning and verbal behavior*, 13(3):272–281.

Shillcock, R. (1982). The on-line resolution of pronominal anaphora. *Language and Speech*, 25(4):385–401.

Smyth, R. (1994). Grammatical determinants of ambiguous pronoun resolution. *Journal of Psycholinguistic Research*, 23(3):197–229.

Van Gompel, R. P. and Majid, A. (2004). Antecedent frequency effects during the processing of pronouns. *Cognition*, 90(3):255–264.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D., and Hirschman, L. (1995). A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics.

7. Language Resource References

Seminck, Olga and Pascal Amsili. (2017). *APADEC: Anaphorical Pronouns and their Antecedents in the Dundee Eye-Tracking Corpus*. Laboratoire de Linguistique Formelle, Dundee Corpus, 1.0, ISLRN 186-137-103-148-0.

Annotating Zero Anaphora for Question Answering

Yoshihiko Asao, Ryu Iida, Kentaro Torisawa
National Institute of Information and Communications Technology
Kyoto 619-0289, Japan
{asao, ryu.iida, torisawa}@nict.go.jp

Abstract

We constructed a large annotated dataset of zero pronouns that correspond to adjuncts marked by *-de* (translated to English as *in*, *at*, *by* or *with*) in Japanese. Adjunct zero anaphora resolution plays an important role in extracting information such as location and means from a text. To our knowledge, however, there have been no large-scale dataset covering them. In this paper, focusing on the application of zero anaphora resolution to question answering (QA), we proposed two annotation schemes. The first scheme was designed to efficiently collect zero anaphora instances that are useful in QA. Instead of directly annotating zero anaphora, annotators evaluated QA instances whose correctness hinges on zero anaphora resolution. Over 20,000 instances of zero anaphora were collected with this scheme. We trained a multi-column convolutional neural network with the annotated data, achieving an average precision of 0.519 in predicting the correctness of QA instances of the same type. In the second scheme, zero anaphora is annotated in a more direct manner. A model trained with the results of the second annotation scheme performed better than the first scheme in identifying zero anaphora for sentences randomly sampled from a corpus, suggesting a tradeoff between application-specific and general-purpose annotation schemes.

Keywords: zero anaphora, question answering, convolutional neural networks

1. Introduction

Zero anaphora refers to anaphora in which the anaphor has a phonetically null form. Zero anaphora resolution is an important sub-problem of many NLP tasks such as question answering (QA). To successfully apply machine learning algorithms to zero anaphora, it is crucial to construct a large and consistent annotated dataset.

In this paper, we focus on zero pronouns that correspond to non-obligatory adjuncts marked by postposition *-de* in Japanese, which mark the location, means, reason or manner of an event and can be translated to English prepositions such as *in*, *at*, *by* or *with*. Although the identification of adjunct zero pronouns plays an important role in extracting such information as location and means from a text, to our knowledge, there have been no attempts to create a large-scale dataset covering them.

While it is possible to exhaustively annotate zero anaphora in a corpus to achieve our goal, such an approach might be inefficient when we are interested only in instances that are useful in a specific application. Given this, we propose two different annotation schemes in this paper. In the first scheme, annotators annotate QA instances that potentially involve zero anaphora. In the second scheme, annotators directly annotate noun-predicate pairs with regard to whether they are in a zero anaphora relationship. We evaluated the performance of these two schemes using an existing neural network-based zero anaphora resolution method (Iida et al., 2016). Our experimental results show that the first scheme achieved better performance when it is used to train a module for a QA system, suggesting that the effectiveness of an annotation scheme depends on applications even in a relatively well studied task like zero anaphora resolution. Conversely, the model trained with annotation results of the second scheme achieved better performance in identifying zero anaphora for sentences randomly sampled from a corpus.

We collected 20,830 instances of *-de* zero anaphora with our first annotation scheme alone, while the Kyoto University Text Corpus (Kawahara et al., 2002), the largest existing resource that we are aware of, has only 333 instances of *-de* zero anaphora of the equivalent type. Table 1 shows a few illustrative examples of zero anaphora successfully collected in our work.[1]

レンタルバイクを借りて島をまわる。I rent a motorcycle$_i$ and *travel* the island [$^{\text{by}}$ $\emptyset_i$].
ソーラー発電が拡大すると環境はどう変化するのか。 How does our environment *change* [$^{\text{by}}$ $\emptyset_i$] if solar electricity$_i$ expands?
名古屋市立大学は、市民と科学者が喫茶店でコーヒーを飲みながら科学について話し合う「サイエンスカフェ」を名古屋市内で開催する。Nagoya City University will hold a 'science café' in the city of Nagoya$_i$, in which citizens and scientists *discuss* science [$^{\text{in}}$ $\emptyset_i$] while drinking coffee at a café.

Table 1: Examples of *-de* zero anaphora in our data

2. Related work

Anaphora or coreference has been annotated in several projects including Message Understanding Conference (MUC) (Hirschman and Chinchor, 1997), Automatic Context Extraction (ACE) (Doddington et al., 2004) and OntoNotes (Hovy et al., 2006), but zero anaphora is not annotated in their English corpora. The OntoNotes corpora for pro-drop languages like Chinese and Arabic contain coreference annotations for certain types of zero pronouns. They do not, however, include adjunct zero pronouns, which we deal with in this paper.

Another kind of resources that are relevant to our work is annotated corpora of semantic roles or frame elements such as PropBank (Palmer et al., 2005) and FrameNet

[1] We only deal with intra-sentential anaphora in this paper.

(Baker et al., 1998). In FrameNet, for example, frame elements that are not overtly encoded are annotated as Null Instantiation, some of which can be regarded as adjunct zero anaphors, although their antecedents are not annotated.

As for Japanese resources, zero anaphora was annotated in 5,000 sentences of the Kyoto University Text Corpus (Kawahara et al., 2002), including *-de* zero anaphora, although its size is small; it has only 333 instances of intra-sentential *-de* zero anaphora. Zero anaphora is also annotated for 20,000 sentences in the NAIST corpus (Iida et al., 2007), but only for *-ga* (nominative), *-o* (accusative) and *-ni* (dative).

While zero anaphora resolution has been recognized as an important task in pro-drop languages such as Chinese and Japanese, the task has been less prominent in languages like English, in which core arguments are usually realized as overt forms. However, adjuncts can be omitted in any language, and in such cases, they must be inferred from the context. For example, consider the following English sentence: *Shortly after her arrival in Tokyo, she began her career as a journalist*. Given this sentence, we can infer that she began her career as a journalist *in Tokyo*, but it is not a trivial task to identify this implicit relation. Thus, although the target language of our current work is Japanese, it has relevance to other types of languages as well.

3. *-de* zero anaphora of Japanese

In this paper, we focus on adjuncts marked by *-de* in Japanese. The postposition *-de* in Japanese marks the location, means, reason, or manner of an event or action denoted by the predicate it modifies; it can be translated into a number of English prepositions including *in*, *at*, *by* and *with* depending on the context. Examples of the usage of *-de* are shown below. In sentence (1), *-de* marks the location of the action, whereas in sentence (2), *-de* marks the means by which the action is performed.

(1) *kōen-de* *hashiru*
park-LOC run
'run in a park'

(2) *supūn-de* *taberu*
spoon-INSTR eat
'eat with a spoon'

In some cases, a *-de* phrase is not overt, but can be recovered from the context. Such cases constitute examples of *-de* zero anaphora:

(3) *Kare-wa* *gakkō$_i$-ni* *itte* $\emptyset_i^{de}$ *benkyōshita*
he-TOP school-to go.and studied
'He went to school$_i$ and studied $^{\text{at}}\emptyset_i$'

(4) *shizen-gengo-shori$_i$-o* *katsuyōshite*
natural-language-processing-ACC utilize.and
$\emptyset_i^{de}$ *iryō-o* *kaizendekiru*
medicine-ACC improve.can
'We can improve medicine $^{\text{with}}\emptyset_i$ utilizing natural language processing$_i$'

In sentence (3), the location of studying is not directly expressed, but can be inferred from the preceding part, *he went to school*. In the same vein, in sentence (4), *natural language processing* does not directly modify *improve*, but we can clearly see that it is the means of the action denoted by the predicate.

Solving *-de* zero anaphora is useful in a variety of applications including question answering. For example, given sentence (4), to build a QA system that can correctly answer to the question 'What can we improve medicine with?', the system must resolve *-de* zero anaphora. More generally, *-de* anaphora resolution plays a crucial role when we would like to extract information like location and means that must be inferred from context.[2]

4. Data construction

To generate datasets for annotation, we used WISDOM X (Mizuno et al., 2016), a question-answering system that our team has been developing.[3] The factoid QA module of WISDOM X accepts a question in Japanese and returns nouns as answers, as well as original sentences from the web corpus that support the answers. An example is below.

(5) **Question** *AI-de* *nani-ga* *jitsugensuru*
AI-INSTR what-NOM be.realized
'What will be realized by AI?'

Answer *kaji-robotto* 'housekeeping robot'

Sentence *AI-ga* *sarani* *hattensureba,*
AI-NOM further develop.if
***kaji-robotto**-ga* *jitsugensuru* *daroo.*
housekeeping-robot-NOM be.realized will
'If AI develops further, housekeeping robots will be brought into reality.'

Human annotators judge whether the answer is correct, or in other words, whether the sentence supports the answer given the question. In this example, we expect that the annotators' judgment is positive. When the judgment is positive, we can suppose that the sentence entails *housekeeping robots will be realized by AI*, although the *-de* phrase (translated as *by AI*) is not overtly expressed. This indicates that the original sentence involves zero anaphora, as shown in sentence (6).

(6) *AI$_i$-ga* *sarani* *hattensureba,*
AI-NOM further develop.if
***kaji-robotto**-ga* $\emptyset_i^{de}$ *jitsugensuru* *daroo.*
housekeeping-robot-NOM be.realized will
'If AI$_i$ develops further, housekeeping robots will be brought into reality $^{\text{by}}\emptyset_i$.'

Thus, this annotation task can be regarded as an indirect way to discover zero anaphora instances. With this scheme,

[2] Assuming zero pronouns for adjuncts may be non-conventional, but this is just one way of capturing covert semantic relationships between nouns and predicates in text. Instead of using the notion of adjunct zero anaphora, we can say that there are implicit semantic links between 'school' and 'study' in sentence (3) and between 'natural language processing' and 'improve' in sentence (4) that can be expressed by postposition *-de*. This alternative view does not affect our discussion.

[3] http://wisdom-nict.jp/

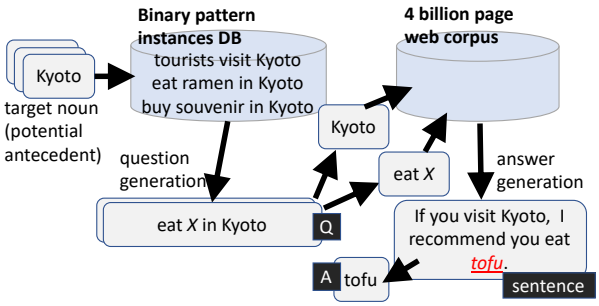


Figure 1: Construction of QA instances

we can focus on zero anaphora instances that are likely to be useful in QA, rather than zero anaphora in general. Another advantage of this annotation scheme is that annotators do not have to know what zero anaphora is, and therefore non-experts can more readily work on it.

In our approach, QA instances for annotation are automatically generated by combining the question suggestion module and the factoid QA module of WISDOM X. Figure 1 illustrates our workflow, which we also summarize below.

1. Choose a *-de candidate* (e.g., *Kyoto*)
2. Generate questions:
   (a) Extract binary pattern instances that contain the *-de* candidate (e.g., *eat ramen in Kyoto*)
   (b) Generate questions based on these binary pattern instances (e.g., *What is there to eat in Kyoto?*)
3. Generate answers:
   (a) Decompose the questions into a *-de* candidate and a unary pattern
   (b) Search for sentences with both the *-de* candidate and the unary pattern

In the first step, we choose a noun that is frequently used with *-de*, which we call the *-de candidate*. The following steps are designed to find a zero anaphor which has this *-de* candidate as the antecedent.

In the second step, we generate questions using the question suggestion module. The module works in the following way. First, it searches for binary pattern instances in our database that contain the *-de* candidate.[4] Here a *binary pattern instance* refers to a dependency tree fragment that consists of a predicate and two nouns that depend on it. For example, when the *-de* candidate is *Kyoto*, examples of binary pattern instances include ⟨*kankōkyaku-ga Kyōto-o otozureru*⟩ '*tourists visit Kyoto*', ⟨*Kyōto-de rāmen-o taberu*⟩ '*eat ramen in Kyoto*' and ⟨*Kyōto-de kaigi-ga hirakareru*⟩ '*a conference is held in Kyoto*'. While the *-de* candidate *Kyoto* may appear in a variety of syntactic positions, we only use questions in which it appears with *-de*. Next, questions are generated by replacing the non-*de*-candidate noun with an interrogative word. For example, the question *Kyōto-de nani-o taberu* (literally '*eat what in Kyoto*', i.e., '*what is there to eat in Kyoto?*') is generated based on such instances as ⟨*Kyōto-de rāmen-o taberu*⟩ '*eat ramen in Kyoto*'.

In the third step, the generated questions are input into our QA system, WISDOM X. While WISDOM X uses multiple methods to find answers, for our present work, we only use answers obtained via the method that we describe here. WISDOM X decomposes the question into the *-de* candidate and the *unary pattern* that corresponds to the question minus the *-de* candidate. Here a *unary pattern* refers to a dependency tree fragment that consists of a predicate and a slot for a noun, such as ⟨*X-ni iku*⟩ '*go to X*' or ⟨*X-ga aku*⟩ '*X opens*'. In our '*eat what in Kyoto*' example, the *-de* candidate is *Kyoto* and the unary pattern is *'eat X'*. The system then searches for sentences that contain *both* the unary pattern and *-de* candidate. We will obtain, for example, the following sentence from the corpus: '*If you visit Kyoto, I recommend you eat tofu*'. Here, the noun *tofu* fills *X* of the *'eat X'* pattern; therefore, *tofu* is presented as an answer along with the original sentence, which the answer is based on. As is the case in this example, we only use sentences in which the *-de* candidate ('*Kyoto*' in this example) and the target predicate ('*eat*') are *not* in a dependency relationship. We expect sentences collected with this method to have a higher-than-average chance of involving *-de* zero anaphora. This is because our procedure guarantees that the *-de* candidate is a noun that frequently occurs in the *-de* position. Note that, because WISDOM X does not search for contexts beyond a single sentence, our target is limited to intra-sentential anaphora.

We created two datasets for annotation, **QAAnnot** and **AllNouns**, based on the QA instances generated by the procedure above.

QAAnnot Annotators directly evaluate QA instances that potentially involve zero anaphora. This task can be simultaneously interpreted as both a QA evaluation task and a zero anaphora annotation task. For this task, we obtained 100,000 QA instances in the following manner. First, we generated questions for 10,000 nouns randomly sampled from the nouns that most frequently appear in the *-de* position in the TSUBAKI corpus (Shinzato et al., 2008) of 600 million web pages. Next, we randomly sampled questions according to the frequency distribution of the predicates; these questions were then input into WISDOM X until we obtained 100,000 QA instances. Finally, human annotators judge each QA instance for its correctness.

AllNouns While **QAAnnot** may be optimized for QA, its special annotation scheme may have a negative impact on performance when it is used to train a model for identifying *-de* zero anaphora in general. To investigate this, we created the second dataset called **AllNouns**; for this dataset, we obtained 10,000 QA instances using the same procedure as **QAAnnot**. Unlike **QAAnnot**, however, annotators do not see questions or answers, but instead judge whether each noun in the original sentence is in a *-de* anaphora relationship with the target predicate. On average there were

[4] Our binary pattern database is based on the four-billion-page web corpus that we constructed, which is also used by the factoid QA module.

	#sentences	#predicates	#pairs	#-*de* zero anaphora		#annotators	approx. person days	Fleiss' κ
QAAnnot	100,000	100,000	100,000	20,830	(20.8%)	24	500	0.556
AllNouns	10,000	10,000	70,971	7,749	(10.9%)	8	100	0.539
General	1,433	3,790	17,392	1,126	(6.5%)	4	20	0.495
KTC	5,127	14,987	90,731	333	(0.4%)	–	–	–

KTC: Kyoto University Text Corpus (Kawaraha et al. 2002)

Table 2: Annotation results

seven nouns to annotate per sentence. More than one noun can be simultaneously in a *-de* anaphora relation with the same target predicate.

We also created a small third dataset for evaluation, which we refer to as **General**. For this dataset, sentences were randomly sampled from the four-billion-page web corpus. For each noun-predicate pair that was identified as not being in a dependency relationship, annotators annotated whether it was in a *-de* anaphora relationship or not. In order to restrict our data to Japanese body texts, we only used sentences that (i) have at least two postpositions and (ii) end with the Japanese full stop (。).

To identify nouns and predicates to annotate, we used the morphological analyzer MeCab (Kudo et al., 2004), as well as the dependency parser J.DepP (Yoshinaga and Kitsuregawa, 2009).

5. Annotation results

In this section, we describe annotation results obtained from the annotation tasks described above. Table 2 summarizes the sizes of our datasets as well as our annotation results.

For each dataset, three annotators independently evaluate each instance, and the final judgment was determined by a majority vote. An annotator was sometimes replaced by another person after completing a set of 1,000 instances. The total numbers of annotators participated in our work, as well as the time required to create each dataset, are included in Table 2.

While 20.8% of the instances were judged to be positive in **QAAnnot**, the percentage was only 10.9% in **AllNouns** and 6.5% in **General**. This difference is expected given the way each dataset was built; **QAAnnot** has the highest proportion of positive instances because it is designed to annotate only likely candidates of zero anaphora. **AllNouns** has a smaller proportion of positive instances because not only the original *-de* candidates but also the other nouns are annotated. The third dataset, **General**, has the lowest proportion of positive instances, as it consists of random sentences in the corpus.

Table 2 also shows a comparison with the Kyoto University Text Corpus (Kawahara et al., 2002). In the Kyoto University Text Corpus, the number of noun-predicate pairs that are targets of zero anaphora annotation based on our criteria is 90,731. Among them, only 333 pairs (0.3%) are annotated as *-de* zero anaphora. The number of *-de* zero anaphora that we collected is substantially larger, in terms of both absolute numbers and the frequency relative to the size of the corpus.

6. Experiments

	train	valid	devel	test
QAAnnot	62,527	20,805	10,401	10,409
QAAnnot-Small	44,603	14,753	-	-
AllNouns	44,603	14,753	7,231	7,253
General	-	-	-	18,265

Table 3: Dataset sizes for our experiments

To see how our annotation results could be generalized to new data, we employed the multi-column convolutional neural network (MCNN) (Cireşan et al., 2012). An MCNN is a variant of a convolutional neural network and has multiple independent columns, each of which has its convolutional and pooling layers. MCNNs have recently been successfully used to model subject zero anaphora in Japanese (Iida et al., 2016). In this work, Iida et al. extracted eleven distinct column inputs from the target predicate (*pred*), the *-de* candidate (*cand*) and their context. The column inputs consist of (a) the word sequence of *cand* and *pred*, (b) the surface word sequences before *cand*, between *cand* and *pred* and after *pred*, (c) the four word sequences extracted from the dependency tree, and (d) *pred* and the word sequences before and after *pred*. We used the same eleven column inputs for our experiments; our architecture is illustrated in Figure 2. More details on the definition of each column are given in (Iida et al., 2016).

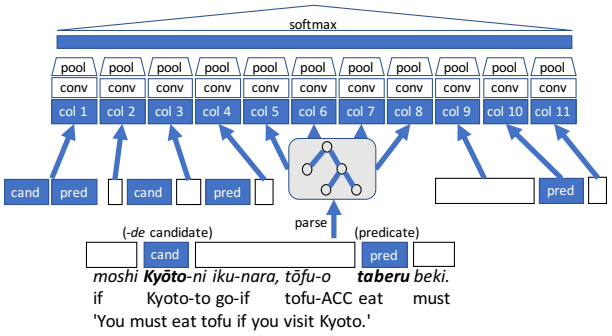


Figure 2: Our multi-column convolutional network architecture

Our MCNN was implemented with Theano (Bastien et al., 2012). We used 300-dimensional word embedding vectors pre-trained with Wikipedia articles using Skip-gram with a negative-sampling algorithm (Mikolov et al., 2013). We treated all the words that only appeared once as unknown words and assigned them a random vector. We used an SGD with mini-batches of

test data	training data	R	P	F	avg. P
QAAnnot	**QAAnnot**	0.263	0.629	0.371	**0.519**
	QAAnnot-Small	0.199	**0.677**	0.307	0.510
	AllNouns	**0.335**	0.523	**0.408**	0.452
AllNouns	**QAAnnot**	0.198	0.563	0.293	0.402
	QAAnnot-Small	0.124	**0.589**	0.204	0.369
	AllNouns	**0.288**	0.569	**0.382**	**0.451**
General	**QAAnnot**	0.125	0.379	0.188	0.218
	QAAnnot-Small	0.081	**0.414**	0.135	0.197
	AllNouns	**0.186**	0.413	**0.257**	**0.269**

Table 4: Performance comparison for different combinations of training and test data

100 and a learning rate decay of 0.95. We used 3-, 4- and 5-grams with 100 filters each. Average precision was used as our evaluation metric. Tuning embeddings in training was turned off, as we found no performance improvements. **QAAnnot** and **AllNouns** were divided into training, validation, development and test data as summarized in Table 3, such that noun-predicate pairs from the same sentence were included in the same bin.[5] For **General**, all instances were used as test data. Further, because **QAAnnot** is larger than **AllNouns**, we could not easily determine whether the performance differences between these two datasets were due to differences in annotation methods or size. To avoid this complication, we constructed **QAAnnot-Small** by randomly sampling **QAAnnot** such that the sizes of training and validation data exactly matched those of **AllNouns**; note that **QAAnnot-Small** was only used for training.

Table 4 summarizes our results. Regardless of the type of training data, the average precision was highest when the test data was **QAAnnot**, and lowest when the test data was **General**. This is probably because **QAAnnot** has the largest proportion of positive instances, whereas **General** has the smallest.

The model trained with **QAAnnot** outperformed that of **AllNouns** when the test data was **QAAnnot**, whereas the model trained with **AllNouns** outperformed that of **QAAnnot** when it was **AllNouns** or **General**, in terms of average precision. As expected, the model trained with **QAAnnot-Small** performed worse than that of **QAAnnot** for all test data, but not to the extent that the gap between the two annotation schemes was reversed, showing that the performance differences between **QAAnnot** and **AllNouns** were not caused by their size differences. Our results suggest that **QAAnnot** has an advantage when our goal is to improve a module of a QA system, while **AllNouns** is better for identifying *-de* zero anaphora in randomly sampled sentences.

Figure 3, 4 and 5 show the precision-recall curves corresponding to different training data when **QAAnnot**, **AllNouns** and **General** were used as test data respectively. To improve QA, we would be able to set a threshold such that only relatively reliable answers are returned.

[5]The total number of instances does not exactly match the total number of annotated noun-predicate pairs because sometimes the same noun occurs more than once in a sentence. In these cases, annotations were made only once without distinguishing different occurrences of the same noun, but each occurrence of the noun was used as a different instance for experiments.

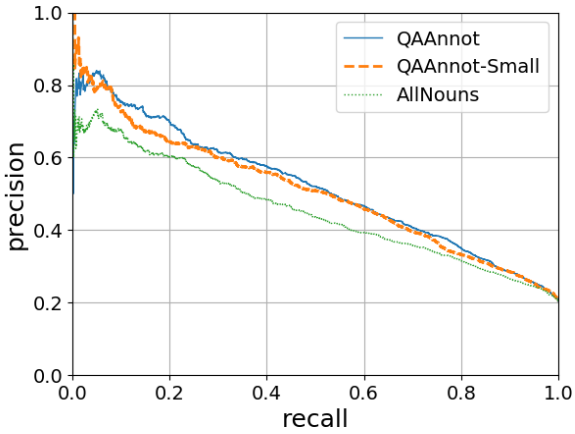


Figure 3: The precision-recall curves for experiments in which **QAAnnot** was used as test data

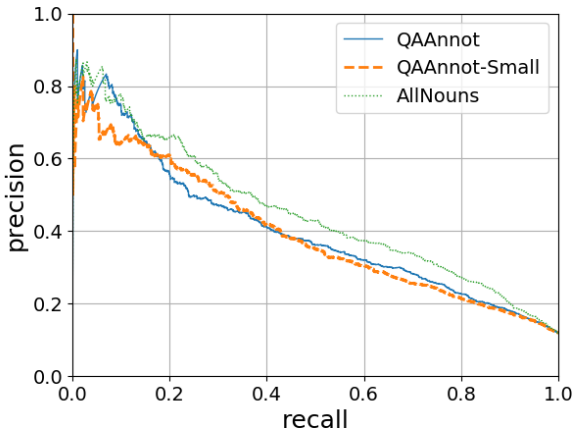


Figure 4: The precision-recall curves for experiments in which **AllNouns** was used as test data

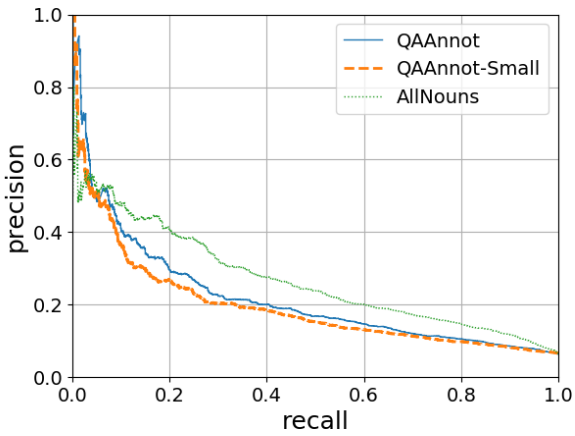


Figure 5: The precision-recall curves for experiments in which **General** was used as test data

To the best of our knowledge, this work is the first attempt to identify *-de* zero anaphora on a large scale, and thus a direct performance comparison with existing work is not possible. For reference, Ouchi et al. (2017) report zero anaphora resolution experiments for *-ga* (nominative), *-o* (accusative) and *-ni* (dative); their best F-measures were 50.65%, 35.07% and 9.83% respectively. Our best F-measure on **General** is 25.7%, which surpass the best score for dative, but not for nominative or accusative. This suggests that *-de* zero anaphora is more difficult to solve than *-ga* and *-o* zero anaphora.

7. Conclusions

In this paper, we described our work in constructing a large annotated dataset of zero pronouns that correspond to adjuncts marked by *-de* in Japanese. The contrast between **QAAnnot** and **AllNouns** can be equated to the contrast between an end-to-end approach and a component-based approach. While **QAAnnot** performs better for QA, it does not generalize well to sentences randomly sampled from a corpus, suggesting that there is a tradeoff between application-specific and general-purpose annotation methods.

8. Bibliographical References

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley Framenet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., and Bengio, Y. (2012). Theano: new features and speed improvements. In *In Proceedings of the NIPS 2012 Workshop: Deep Learning and Unsupervised Feature Learning.*

Cireşan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. *International Conference of Pattern Recognition*, pages 3642–3649.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The Automatic Content Extraction (ACE) program-tasks, data, and evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, volume 2, pages 837–840.

Hirschman, L. and Chinchor, N. (1997). MUC-7 coreference task definition.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). Ontonotes: the 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.

Iida, R., Komachi, M., Inui, K., and Matsumoto, Y. (2007). Annotating a Japanese text corpus with predicate-argument and coreference relations. In *Proceedings of the Linguistic Annotation Workshop*, pages 132–139. Association for Computational Linguistics.

Iida, R., Torisawa, K., Oh, J.-H., Kruengkrai, C., and Kloetzer, J. (2016). Intra-sentential subject zero anaphora resolution using multi-column convolutional neural network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-2016)*, pages 1244–1254.

Kawahara, D., Kurohashi, S., and Hashida, K. (2002). Construction of a Japanese relevance-tagged corpus. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*, pages 2008–2013.

Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). Applying conditional random fields to Japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004)*, pages 230–237.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of Advances in Neural Information Processing Systems 26 (NIPS 2013)*, pages 3111–3119.

Mizuno, J., Tanaka, M., Ohtake, K., Oh, J.-H., Kloetzer, J., Hashimoto, C., and Torisawa, K. (2016). WISDOM X, DISAANA and D-SUMM: Large-scale NLP systems for analyzing textual big data. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016) (Demo Track)*.

Ouchi, H., Shindo, H., and Matsumoto, Y. (2017). Neural modeling of multi-predicate interactions for Japanese predicate argument structure analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1591–1600.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Shinzato, K., Shibata, T., Kawahara, D., Hashimoto, C., and Kurohashi, S. (2008). TSUBAKI: An open search engine infrastructure for developing new information access methodology. In *Processings of the 3rd International Joint Conference on Natural Language Processing*, pages 189–196.

Yoshinaga, N. and Kitsuregawa, M. (2009). Polynomial to linear: Efficient classification with conjunctive features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP-2009)*, pages 1542–1551.

Building Named Entity Recognition Taggers via Parallel Corpora

Rodrigo Agerri, Yiling Chung, Itziar Aldabe, Nora Aranberri, Gorka Labaka, German Rigau
IXA NLP Group, University of the Basque Country UPV/EHU
rodrigo.agerri@ehu.eus,lydia193@gmail.com,{itziar.aldabe,nora.aranberri,gorka.labaka,german.rigau@ehu.eus}

Abstract

The lack of hand curated data is a major impediment to developing statistical semantic processors for many of the world languages. Our paper aims to bridge this gap by leveraging existing annotations and semantic processors from multiple source languages by projecting their annotations via the statistical word alignments traditionally used in Machine Translation. Taking the Named Entity Recognition (NER) task as a use case, this work presents a method to automatically induce Named Entity annotated data using parallel corpora without any manual intervention. The projected annotations can then be used to automatically generate semantic processors for the target language helping to overcome the lack of training data for a given language. The experiments are focused on 4 languages: German, English, Spanish and Italian, and our empirical evaluation results show that our method obtains competitive results when compared with models trained on gold-standard, albeit out-of-domain, data. The results point out that our projection algorithm is effective to transport NER annotations across languages thus providing a fully automatic method to obtain NER taggers for as many as the number of languages aligned in parallel corpora. Every resource generated (training data, manually annotated test set and NER models) is made publicly available for its use and to facilitate reproducibility of results.

Keywords: Named Entity Recognition, Information Extraction, Multilingual Language Resources

1. Introduction

The best results for every type of semantic processing task are currently obtained by supervised corpus-based approaches. This means that manually annotated data is required to learn probabilistic models from the data. This poses a major obstacle to developing semantic processors whenever there is not manually annotated data for a semantic task in a given language. In most cases, manually annotating text for every single specific need is generally inefficiently slow and, in most cases, unaffordable in terms of human resources and economic costs. Instead, we would like to be able to use already available semantic processors and texts in other languages to get a good statistical model for a new target language.

Our method leverages existing semantic processors and annotations to overcome the lack of annotation data for a given language. The intuition is to transfer or project semantic annotations, from multiple sources to a target language, by statistical word alignment methods applied to parallel texts (Och and Ney, 2000; Liang et al., 2006). The projected annotations could then be used to automatically generate semantic processors for the target language. In this way we would be able to provide semantic processors without training data for a given language.

Thus, this means that the problem can be decomposed into two smaller inter-related ones: (i) How to project semantic annotations across languages via parallel texts with a sufficient acceptable quality to train semi- or weakly-supervised semantic processors and (ii) how to effectively leverage the (potentially noisy) projected annotations to induce robust statistical models to perform semantic tasks such as Named Entity Recognition (NER), Word Sense Disambiguation or Semantic Role Labelling, to name but a few.

In this paper we focus on the first problem. We propose using parallel data from multiple languages as source to project the semantic annotations to a target language. Our hypothesis is that in the combination of multiple sources lies the possibility of improving the quality of the projections that will be used to train the semantic processors. For the purpose of this work, we take the NER task as a use case to test our hypothesis. Furthermore, four languages are considered in our study: English, Spanish, German, and Italian, although any language aligned in a a parallel corpus is a possible candidate. Our method can be illustrated by the following example provided in Figure 1, which takes English as a target language.

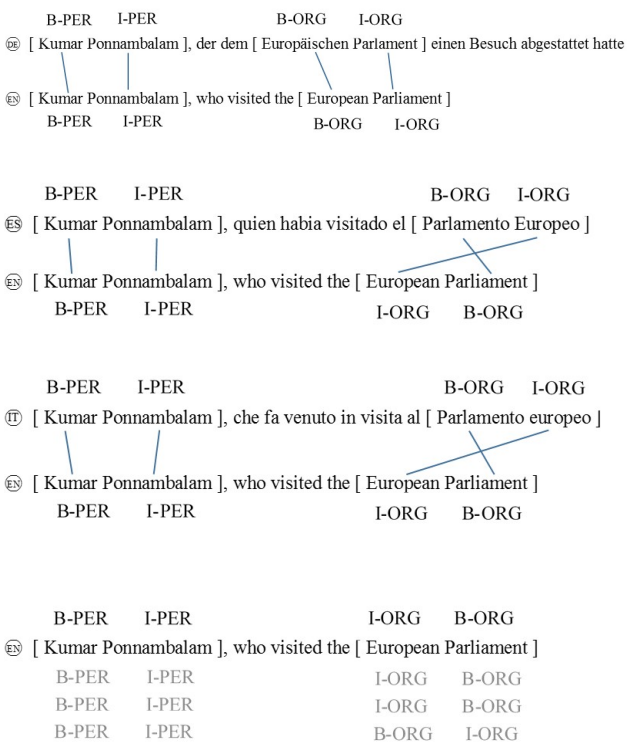


Figure 1: Projecting Named Entity annotations via word alignments to English as target language.

Alignments	Tokens in source languages es; de; it	Target language (en)
1-1 alignment	Europeos; Europas; europeo	European
Multiple alignments	del, Parlamento; Parlamentsgebäude; Parlamento	Parliament
Misalignments	del; Parlamentsgebäude; Parlamento	Parliament
No alignments	Los; Beschäftigungspakten; NONE	European

Table 1: Examples of various alignments from Spanish, German, Italian to English.

2. Methodology

In order to develop our system we need: (i) a Named Entity Recognition (NER) tagger; (ii) a parallel corpus to project the semantic annotations in order to create the training data; (iii) NER datasets for training the initial models to be deployed to tag the parallel corpus, and (iv) a gold-standard test set to evaluate our approach.
As NER tagger we choose *ixa-pipe-nerc* (Agerri and Rigau, 2016). It is designed to work robustly across languages and datasets and it obtains state of the art results for the languages used in this study. We also use the following corpora:

1. Gold standard data for training the initial *ixa-pipe-nerc* models for the source languages. CoNLL 2002 and 2003 for German, English and Spanish, and Evalita 2009 for Italian. Both CoNLL and Evalita annotate the three entity types (location, organization and person) that we will use to induce our training data.

2. The Europarl parallel corpus on which to perform the cross-lingual projections (Koehn, 2005), word-aligned using Giza++ (Och and Ney, 2000) and divided into a training and a test set.

3. The Europarl gold-standard test set is a new manually-annotated evaluation set taken from the Europarl. The test set contains 800 sentences manually annotated using the three entity types and following the CoNLL 2002 and 2003 guidelines for the 4 languages used in this paper.

4. Back-off corpora to resolve ties in the projection step. The idea is to compute the most frequent tag of a token in a large NER annotated resource. Thus, in case of ties during the annotation projection the most frequent entity tag will be assigned. We use Wikiner, a silver standard built from the Wikipedia for several languages (Nothman et al., 2013).

3. Projecting Annotations

There can be various types of word alignments between the tokens and Named Entity (NE) classes for any two languages: 1-1 alignments, multiple alignments, misalignments and no alignments. Examples of such cases can be found in Table 1:

- 1-1 alignments occur when the NE class of a token in the source is cleanly aligned to one token and NE class in the target.
- Multiple alignments means that two or more tokens and NE classes are aligned from the source to one token and NE class in the target (e.g., *del* and *Parlamento* in Spanish aligned to one token *Parliament* in English).
- Misalignments happen when a token is wrongly aligned to a token in the target. Our algorithm considers different strategies for these different types of (mis-)alignments.
- No alignments occur when the token containing a NE class in the source is not aligned to any token in the target.

Taking this into account, for this work we have developed two projection algorithms: (1) an *upper bound* designed to evaluate the quality of the word alignments to transport gold-standard Named Entity Recognition (NER) annotations; (2) *strict match* projection algorithm to project automatically annotated NER annotations over the training set. The automatically projected annotations will then be used to train new NER models for the target language.

3.1. Upper-Bound

In order to establish the quality of the word alignments to project semantic annotation, we designed an *upper bound* projection method. In order to do so, we project the manually annotated Named Entities using the Europarl test set described in Section 2. Furthermore, we only transport a NE tag to the target language whenever all three source annotations coincide. Thus, no back-off will be used for the upper-bound. The resulting projected data will be evaluated with respect to the test gold-standard of the target language using the CoNLL script for NER evaluation. Intuitively, the results should establish how much noise is created by the projection via word alignments whenever the alignments and tags for all the three source languages agree. For each language, the upper-bound project algorithm performs the following steps:

1. We obtain the word alignments for all four languages and order them following the alignment types presented in Table 1.

2. The semantic tags are projected via word alignments from the three source languages to a given fourth, target language. In this step a NE tag is projected only if there are three agreements between the alignments in the source. This may happen whenever there is a 1-1 alignment, or if there are multiple alignments but three of the tags coincide across the three source languages. Otherwise, the projected tag is 'O'. This is illustrated by Table 2.

Alignments	Tokens	Tags in projection es; de; it	Projected tag
1-1 alignment	European	ORG; ORG; ORG	ORG
Multiple alignments	Parliament	ORG; O, ORG; ORG	ORG

Table 2: Examples of upper-bound projection for English as target language.

3. Assign the span to the projected NE tag: It should be noted that the projections are performed at token level, so in order to annotate a Named Entity in the target language, the projected NE tags must be contiguous.

3.2. Strict Match

The aim of our work is to project Named Entity (NE) annotations from several source languages into a target language for which there is not training data. Thus, the result of the projection will be an automatically created training corpus on which we could train a Named Entity Recognition (NER) tagger. As there are not large parallel corpora manually annotated with Named Entities on which to perform the projections, we annotate a large parallel corpus with already existing Named Entity taggers so that we can use the automatic Named Entity annotations and word alignments to undertake the projection across languages. Specifically, in order to automatically generate NER taggers without manual intervention via the *strict-match* algorithm our method goes through the following four steps:

1. We train ixa-pipe-nerc (Agerri and Rigau, 2016) on the gold-standard training data from CoNLL (Tjong Kim Sang and De Meulder, 2003; Tjong Kim Sang, 2002) and Evalita (Speranza, 2009).

2. The Europarl training set for each language is tagged with the gold-standard trained models.

3. We project the automatic tagged Named Entities from three source languages to a fourth target language, performing all 4 permutations.

4. ixa-pipe-nerc is then trained on the induced training data via projection across languages obtaining a NER tagger which is fully automatically generated.

For *strict match*, given a word in a sentence of the target language, we obtain the aligned words and its NE classes across the three source languages. Next, the NE tags of target language are projected based on the candidates collected from the three source languages. For the first version, our *strict-match* projection algorithm considers at least two or three alignment agreements among three source languages to determine the final tag for target language. If that agreement is not reached, we use a back-off Named Entity tag obtained from computing the most frequent tag for that token in Wikiner (Nothman et al., 2013). *Strict match* requires at least a two alignment agreement (the same NE class for an aligned token in two of three source languages) in order to project a NE tag. If there is a tie (usually when multiple alignments occur), then we back-off and project the most frequent NE class in Wikiner for the given token. More specifically:

1. Following the same procedure as for the upper-bound method, we obtain the word alignments for all four languages and order them following the alignment types presented in Table 1.

2. The semantic tags are projected via word alignments from the three source languages to a given fourth, target language. In this step a NE tag is projected following three criteria: (i) if there are at least two agreements between the alignments in the source; (ii) if there is a tie, e.g., if more than two agreements occur, then via back-off; and (iii) if there is not agreement in the alignments, via back-off. This is illustrated by Table 3.

3. This step is the same for the upper-bound method: we assign the span to the projected NE tags in the target assuming that they must be contiguous.

As mentioned in Section 2., we use the corpus Wikiner (Nothman et al., 2013) to back-off whenever *strict match* cannot decide which tag to project. The back-off strategy is fairly simple. Given a large corpus annotated with Named Entities, we calculate the most frequent tag for each token in the corpus. Thus, whenever we need to back-off, we simply consult the frequencies table obtained from Wikiner for the candidate token and assign to that token the most frequent one. For example, the token *European* mentioned in Table 3 could conceivably be LOC, ORG or PER. As in Wikiner the most frequent tag is ORG, then when backing-off that is the tag that will be assigned to that specific token.

4. Experiments

First we present the results of the *upper-bound* projections using the Europarl gold-standard described in section 2. Table 4 displays the overview results of projecting with both 1-1 and multiple alignments. It is clear that the results obtained projecting multiple alignments are better than those with 1-1 alignment for all four languages. This is probably due to the fact that in 1-1 alignments many projections are not performed because no agreement is found.

With respect to the results of the *strict match* projections, the models obtained from the gold-standard data are those described in (Agerri and Rigau, 2016). For training the models on the projected data we induced the same clustering features described in (Agerri and Rigau, 2016) but using the unlabelled Europarl training set instead of the datasets originally used. These clustering features replaced the features used in the original, gold-standard trained models. Thus, the difference between the gold-standard and the projected models mentioned in Tables 5, 6 and 7 is the training corpus (CoNLL-Evalita vs Europarl) and the clustering lexicons used (Wikipedia-Gigaword, etc. vs Europarl).

As we have already mentioned, we compare the gold-trained models with the automatically induced ones via

Alignments	Tags in projection es; de; it	Projected tag
3-agreement	ORG; ORG; ORG	ORG
2-agreement	ORG; ORG; PER	ORG
no agreement	ORG; LOC; PER	back-off
$>$ 2-agreement	ORG; ORG, LOC; LOC	back-off

Table 3: Strict match projection for the token *European* for English as target language.

Alignment	en	de	it	es
1-1	91.47	75.52	91.75	96.32
Multiple	96.01	94.21	93.50	97.34

Table 4: F1 results on upper-bound projection.

strict match projections. This evaluation allows to understand if our method produces as good results as the models trained on gold standard, albeit out-of-domain, data. The F1 results in Table 5 show that the automatically trained models outperform the models trained on gold-standard data except for Italian.

Training	en	de	it	es
Gold	65.08	49.87	65.82	58.75
Projected	69.14	70.62	62.44	64.16
upper-bound	96.01	94.21	93.50	97.34

Table 5: Comparing F1 results training ixa-pipe-nerc on projected and gold-standard data.

Furthermore, our automatically obtained models are particularly good in terms of precision, which means that our strict match projection algorithm is very strict, and only projects Named Entities when it is quite confident. Thus, for Italian the precision results are 7 points higher, 25 points for Spanish and 10 points for German, as it can be seen Tables 6 and 7.

	Precision	Recall	F1
en	70.00	60.34	64.81
de	68.40	39.24	49.87
it	67.03	62.45	64.66
es	55.66	59.69	57.60

Table 6: Evaluating Gold-standard trained CoNLL and Evalita models on Europarl test.

Still, and even though our first results are quite promising, we should note that the results of the automatically generated models are much lower than those established by the upper-bound.

5. Related Work

Traditionally, there are many studies and works exploring the contribution of semantic information or features with the aim of improving Machine Translation (Koehn, 2010; Artetxe et al., 2015) but the reverse has been rather uncommon. Among previous works using parallel texts to automatically induce linguistic processors, most of them focus on inducing Part of Speech taggers (Yarowsky et al., 2001; Ganchev and Das, 2013; Täckström et al., 2012; Fossum and Abney, 2005) although a very few of them worked on semantic tasks such as Named Entity Recognition (NER) (Yarowsky et al., 2001; Zhang et al., 2016) and Semantic Role Labelling (SRL) (Padó and Lapata, 2009).

	Precision	Recall	F1
en	70.30	68.01	69.14
de	78.87	63.94	70.62
it	75.14	53.41	62.44
es	80.29	53.42	64.16

Table 7: Evaluating models trained on automatically projected data.

Furthermore, almost every previous approach is based on one-to-one projections using only one language pair to induce the linguistic processors. As far as we know, there are only two exceptions: Yarowsky et al. (2001) use bridging between two languages to perform lemmatization in a third target language, and Fossum and Abney (2005) train multiple POS taggers from monolingual source data and combine their annotations to project them to a given target language. Therefore, to our knowledge, no previous approach aims at doing transfer of semantic annotations as we propose in this paper.

These previous works based on projection of annotations have shown that the projected labels can result in a very noisy training set in the target language. Various methods have been applied to address this problem, including smoothing techniques (Yarowsky et al., 2001) and the combination of token-level and type-level constraints to recalculate the probability distribution of the labels in a CRF for Part of Speech tagging (Täckström et al., 2012). (Das and Petrov, 2011) use the projected labels as contraints in a Posterior Regularization framework and (Ganchev and Das, 2013) extend this work by training directly discriminative models via cross lingual projection with Posterior Regularization. Finally, instead of using total counts of labels of a class to enforce the constraints, (Wang and Manning, 2014) define expectation constraints at token level for NERC.

Closest to our work, Zhang et al. (2016) generate a high-confidence annotation set using strict rules on parallel corpora in order to project the Named Entity information from the source to the target. The resulting annotated bitext is then used to train a LSTM model. They evaluate their work with respect to a baseline consisting of the projected tags via automatic word alignments. The results show that the word alignment method is much worse than the bitext trained LSTM model. It should be noted that they do not explain how the annotations are projected via word alignments. Furthermore, we believe that using only one source

language may be detrimental to the quality of the projections.

6. Concluding Remarks

We train the same tagger on the automatically projected training data and on out-of-domain gold-standard annotated data. Our evaluation shows that the automatic generated model outperforms the gold-standard trained model in an in-domain evaluation. In this paper we have demonstrated that it is feasible to automatically induce training data using parallel data without manual intervention. This method allows to generate Named Entity Recognition (NER) taggers for a given language when no manually data is available. Furthermore, our method may be applied to generate annotations for other semantic tasks, such as Semantic Role Labeling or Supersense tagging.

We believe that the reported results could be improved by using several strategies: First, it could be worth to perform another iteration of the *strict match* projection. After all, the projected NE tags are automatically obtained by applying the gold-standard trained models, which, as shown by the evaluation, are much worse than the models obtained from the Europarl. Thus, tagging the Europarl training set with the automatically obtained models and project those annotations could improve the quality of the projections.

Second, we may include more languages to improve the quality of projections. In our experiments we have considered four languages, three source and one target, but it might be worth to investigate if integrating more source annotations is likely to substantially cancel out projection errors.

Third, we could focus the projection via word alignments by language groups, namely, grouping Romance languages, Slavic languages, under the assumption that word alignments for closely related languages may be of higher quality.

Future work also includes evaluating both gold-trained and projected models on out-of-domain data. After all, NER taggers are usually used to tag out-of-domain data, so if our automatically generated models were to be at least as good as the models trained on gold-standard out-of-domain data, that would mean that for out-of-domain use our method would be a convenient solution to obtain general semantic processors without manual intervention.

The gold-standard and automatically generated models[1] as well as the the manually-annotated test set from the Europarl[2] are made publicly available for its use and to facilitate reproducibility of results.

7. Acknowledgements

This work has been funded by the Spanish Ministry of Economy and Competitiveness (MINECO/FEDER, UE), under the projects CROSSTEXT (TIN2015-72646-EXP) and TUNER (TIN2015-65308-C5-1-R).

[1] https://github.com/ixa-ehu/ixa-pipe-nerc

[2] https://github.com/ixa-ehu/ner-evaluation-corpus-europarl

8. Bibliographical References

Agerri, R. and Rigau, G. (2016). Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

Artetxe, M., Agirre, E., Alegria, I., and Labaka, G. (2015). Analyzing english-spanish named-entity enhanced machine translation. In *SSST@ NAACL-HLT*, pages 52–54.

Das, D. and Petrov, S. (2011). Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 600–609. Association for Computational Linguistics.

Fossum, V. and Abney, S. (2005). Automatically inducing a part-of-speech tagger by projecting from multiple source languages across aligned corpora. *Lecture notes in computer science*, 3651:862.

Ganchev, K. and Das, D. (2013). Cross-lingual discriminative learning of sequence models with posterior regularization. In *EMNLP*, pages 1996–2006.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5.

Koehn, P. (2010). *Statistical Machine Translation*. Cambridge University Press.

Liang, P., Taskar, B., and Klein, D. (2006). Alignment by agreement. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 104–111.

Nothman, J., Ringland, N., Radford, W., Murphy, T., and Curran, J. R. (2013). Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175.

Och, F. J. and Ney, H. (2000). *Giza++: Training of statistical translation models*.

Padó, S. and Lapata, M. (2009). Cross-lingual annotation projection for semantic roles. *Journal of Artificial Intelligence Research*, 36(1):307–340.

Speranza, M. (2009). The named entity recognition task at evalita 2009. In *Proceedings of the Workshop Evalita*.

Täckström, O., McDonald, R., and Uszkoreit, J. (2012). Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 477–487. Association for Computational Linguistics.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147.

Tjong Kim Sang, E. F. (2002). Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *Proceedings of CoNLL-2002*, pages 155–158. Taipei, Taiwan.

Wang, M. and Manning, C. D. (2014). Cross-lingual projected expectation regularization for weakly supervised

learning. *Transactions of the Association of Computational Linguistics*, 2:55–66.

Yarowsky, D., Ngai, G., and Wicentowski, R. (2001). Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.

Zhang, D., Zhang, B., Pan, X., Feng, X., Ji, H., and Weiran, X. (2016). Bitext name tagging for cross-lingual entity annotation projection. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 461–470.

Cross-Document, Cross-Language Event Coreference Annotation Using Event Hoppers

Zhiyi Song, Ann Bies, Justin Mott, Xuansong Li, Stephanie Strassel, Christopher Caruso
Linguistic Data Consortium
Suite 810, 3600 Market Street, Philadelphia, Pennsylvania, 19104, USA
{zhiyi, bies, jmott, xuansong, strassel, ccaruso}@ldc.upenn.edu

Abstract

We discuss the development and implementation of an approach for cross-document, cross-lingual event coreference for the DEFT Rich Entities, Relations and Events (Rich ERE) annotation task. Rich ERE defined the notion of event hoppers to enable intuitive within-document coreference for the DEFT event ontology, and the expansion of coreference to cross-document, cross-lingual event mentions relies crucially on this same construct. We created new annotation guidelines, data processes and user interfaces to enable annotation of 505 documents in three languages selected from data already labeled for Rich ERE, yielding 389 cross-document event hoppers. We discuss the data creation process and the central role of event hoppers in making cross-document, cross-lingual coreference decisions. We present the challenges encountered during annotation along with three directions for future work.

Keywords: Events, Information Extraction, Coreference, Linguistic Resources

1. Introduction

This paper discusses the development and implementation of annotation for cross-document and cross-lingual event coreference as an expansion to the Rich Entities, Relations, and Events (ERE) annotation task first defined as part of DARPA's Deep Exploration and Filtering of Text (DEFT) program (DARPA, 2012). The goal of the DEFT program is to develop technologies capable of extracting knowledge from unstructured text in multiple languages and genres. ERE annotation was developed at Linguistic Data Consortium (LDC) to support multiple research directions and evaluations in DEFT. ERE builds on the approach to labeling entities, relations, events and their attributes under a pre-defined taxonomy, following the approach used in Automatic Content Extraction (ACE) (LDC, 2005; Walker et al., 2006; Song et al., 2015; Mott et al., 2016).

One important DEFT use case is automatically building a structured Knowledge Base (KB) from scratch. This task, known as Cold Start, is one of several tasks relevant to DEFT that are evaluated in the NIST Knowledge Base Population evaluation series (NIST 2017; Getman et al, 2018). Given DEFT's focus on whole-corpus understanding culminating in the Cold Start task, Rich ERE annotation has evolved over the course of the program to emphasize cross-document and cross-lingual approaches. Rich ERE event annotation includes 9 event types and 38 subtypes (e.g. Conflict.Attack, Contact.Meet, Movement.TransportPerson). For each event mention annotators label the most salient word evoking the event (the "event trigger"), the event type and subtype, the realis status (Actual, Other or Generic), all of the event's arguments (e.g. agent, instrument) and several attributes like temporal information (Aguilar et al., 2014; Song et al., 2015). Multiple mentions of the same event in the document are labeled for coreference, utilizing the notion of event hoppers. While stricter approaches to event coreference typically require all event features (including arguments and attributes) to be identical, event hoppers relax this requirement to enable coreference of two events that are intuitively the same although certain features may differ. For instance, two reports about the same terrorist incident may differ with respect to the number of perpetrators, especially in the immediate aftermath of the event when facts are still being uncovered. The event hopper approach allows for coreference of such event mentions where other event coreference schemas may not. While event hoppers were originally introduced to support within-document Rich ERE annotation, extending them to cross-document and cross-lingual event coreference was a natural progression since the approach results in a more complete Knowledge Base, particularly when the KB reflects information extracted from multiple languages, sources and genres. In the sections that follow we present the results of our effort to define an approach to cross-document, cross-lingual event coreference using event hoppers as part of the Rich ERE annotation task in DEFT.

2. Related Work

There have been other efforts that have captured some variety of cross-document event-event coreference, which have informed our design of the cross-document event coreference annotation task. These include topic-clustering of documents and pair-wise comparison of event mentions.

The EventCorefBank (ECB) (Bejan and Harabagiu, 2008) contains 482 documents clustered into 43 topics that were annotated for within-document and cross-document coreference according to the TimeML specification (Pustejovsky et al., 2013). Lee et al. (2012) extended ECB annotation, following the OntoNotes guidelines (Pradhan et al., 2007). These studies required both matching predicates and matching arguments for event coreference. The ECB+ corpus (Cybluska and Vossen, 2014a) is an extension of ECB with the addition of source documents as well as event components, using the CROMER (CROss-document Main Events and entities Recognition) tool (Girardi et al., 2014). Event coreference in ECB+ required matching time, place, and participants (Cybluska and Vossen, 2014b).

A richer event-event relation annotation scheme (Hong et al., 2016) was developed to capture cross-document event-event relations, including coreference. This data was constructed using ACE 2005 data and supplemented with data collected by researchers; event relations including coreference were annotated by pairwise comparison of events from documents within a given topic. Event coreference required event arguments in the pair to match.

To support the cross-document component of Event Argument evaluation in TAC-KBP 2016 (NIST, 2016),

LDC manually identified strings of text that contained the event corresponding to a series of manually curated queries (Ellis et al., 2016). The evaluated systems, however, failed to produce any entry points for those queries.

However, we are not aware of previous attempts at labeling cross-document coreference using the event hopper framework, which focuses on annotators' intuitions about event reference and allows event argument mismatches as well as event mention realis mismatches. This approach allows annotators to group together more event mentions and arguments than previous stricter coreference approaches, on both the within-document and cross-document level.

3. Annotation Approach

Cross-document and cross-lingual event coreference annotation for the current effort starts with within-document Rich ERE event hopper annotation as input, and coreferences event hoppers from different documents or languages. In Rich ERE event annotation, every tagged event mention is put into an event hopper. All event mentions that refer to the same event occurrence are grouped into the same event hopper, with the result that each event hopper consists of one or more event mentions. The criteria for judging whether hoppers are coreferential or not are the same as those outlined in the description of the event hoppers in Song et al. (2015):

- They are intuitively the same event
- They have the same event type and subtype
- Temporal and place arguments don't need to match, but need to be the same general scope
- Event arguments may be non-coreferential or conflicting
- Realis status may be different

The event hopper concept is a more inclusive, less strict notion of event coreference than that used in ACE and other schemes, handling within-document event coreference using the notion of event hoppers (as above) which permits intuitive coreference and allows non-matching arguments or realis (see the following subsections for examples). Cross-document coreference is inherently more difficult for annotators than within-document, because the cross-document and cross-language aspect of the task requires annotators to fully digest multiple documents being compared and to develop an understanding of the overall topic as context. When comparing pairs of event hoppers from individual documents, it can be difficult to understand how the individual hoppers fit into the larger picture of the topic as a whole. Moving from annotating within-document event hoppers to cross-document coreference highlights several points where the less strict event hopper concept is necessary, such as differing granularity or realis, and multiple occurrences.

3.1 Event Argument Granularity

Perhaps the most common reason that event mentions are not strictly coreferential is event arguments that differ in granularity or sometimes conflict. For example,

> S1: **Attack** in Baghdad on Thursday
>
> S2: **Bombing** in the Green Zone last week

The journalistic nature of newswire documents lends itself to repetitive language that varies in levels of specificity. Annotators would that these two Conflict.Attack events are coreferential when looking at the context of the documents. Annotators should treat two event hoppers as coreferential even when their arguments are not identical, if the events are intuitively the same.

In the cross-document task, it can be particularly difficult to decide whether the same event is being reported with differing arguments over time, or if it is a different event.

> S1: Policía china **detiene** a 118 sospechosos en un caso de contaminación ambiental.
> *Chinese police are arresting 118 suspects in a case of environmental pollution.*
>
> S2: Ocho personas han sido **detenidas** por haber realizado vertidos ilegales en un río local de la provincia suroccidental china de Yunnan.
> *Eight people have been arrested for illegally dumping in a local river in the southwestern Chinese province of Yunnan.*

Here, it is possible that the 118 suspects mentioned in the first sentence are a superset containing the 8 suspects in the second sentence from a different document. In such cases, the context of the entire document is taken into account. If doing so still does not resolve the ambiguity, then the default is to not coreference the events.

3.2 Factually Conflicting Event Arguments

Event hopper annotation also addresses factually conflicting Event Arguments. Unstructured text such as discussion forum documents, commonly contains misinformation or conflicting assertions. Newswire data covering breaking events may also present seemingly conflicting reports based on what it known at the time of publication. Annotators need to use general information or real knowledge or their own judgment for these coreference decisions.

In the following example, the first document contains an assertion, but in the second document, the information relayed contradicts that in the first document:

> S1: John Smith was **killed** in Canada.
>
> S2: Many people believe that the man was actually shot **dead** in Iowa.

These two event mentions have conflicting place argument (Canada vs. Iowa), but they are interpreted as coreferential, because both mentions refer to the Life.Die event of "John Smith" (also mentioned as "the man").

3.3 Event Hoppers with Different Realis State

Cross-document event coreference annotation includes realis states from the Rich ERE event annotation. A future or hypothetical event mention would be marked "Other", while a similar event mention that refers to an asserted event would be marked "Actual". Different temporal states of the same event should be coreferred. For instance,

S1: Barack Obama will **visit** [OTHER] London next month, the White House has confirmed.

S2: US President Barack Obama **arrived** [ACTUAL] at the Globe Theatre on April 23, 2016 in London, England

When approaching two mentions in cross-document cross-lingual event coreference annotation, "Other" and "Actual" event hoppers can be coreferred if they are intuitively the same event. The event hopper pair in the above example would be judged as coreferential.

3.4 Multiple Occurrences of Events of the Same Type and Subtype

Multiple instances of the same type/subtype of event can occur within documents or within a topic. For example,

S1: Angela Merkel was first **elected** in 2005 and re-elected in 2009 and 2013.

S2: Merkel's re-**election** was never in doubt.

The first sentence contains three separate Personnel.Elect events; it is easy to distinguish them because of the explicit mention of time arguments. However, the second sentence could refer to either 2009 or 2013. The broader context of the document can be used to resolve this ambiguity. However, if it is not possible to resolve in context, then the mentions will not be coreferred.

3.5 Event Hoppers with Conflicting Volition

Event mentions can be coreferred when the arguments of the event are referred to with differing levels of volition in different documents. For example,

S1: The defendant was **dragged** kicking and screaming from the courtroom.

S2: The defendant **left** the courtroom.

Here, the two Movement.TransportPerson event mentions can be included in the same hopper, even though in the first instance the it is clear the entity argument is being transported involuntarily, whereas in the second the same argument seems to be a voluntary participant.

4. Methodology

To perform the cross-document event hopper coreference task, annotators were presented with a pair of event hoppers from two different documents to compare and judge as coreferential or not. The presented event hoppers already included all event mentions that were judged as coreferential from within-document annotation (as well as event hoppers containing only singleton event mentions).

4.1 Source Data

The data consisted of 505 "core" source documents which were annotated with Rich ERE for the TAC KBP 2016 evaluations (Ellis et al., 2016). All of these documents (254 Newswire (NW), and 251 Discussion Forum (DF) threads) were manually selected using a topic-driven approach to ensure appropriate coverage of event types, ambiguous entities, and entities referenced only by nominal mentions (Ellis et al., 2016). Table 1 shows the document and token count distribution by language and genre for the TAC KBP 2016 evaluation "core" set.

Language	Genre	Documents	Tokens
Chinese	NW	85	43,338
Chinese	DF	82	78,675
English	NW	85	41,622
English	DF	84	46,282
Spanish	NW	84	26,228
Spanish	DF	85	40,703
Total	--	505	276,848

Table 1: TAC KBP 2016 "core" set

A wide variety of newsworthy topics were included, with a total of 61 topics, such as: South China Sea Tension, Spanish Train Derailment, Syrian Revolution, Typhoon Haiyan, etc. Some topics included documents in two languages, some in only one language, and ten of the topics included documents from all three languages.

ERE annotation for the "core" set followed the Rich ERE annotation guidelines (Song et al., 2015), with the exception that the inventory of Rich ERE event types and subtypes was reduced to 18 types and subtypes.

4.2 Data Pruning

Exhaustive manual pairwise comparison of all event hoppers in the corpus for coreference was infeasible. We therefore limited the annotation scope by (1) topic, (2) event type/subtype, and (3) realis, in order to make the annotation task more tractable.

Within each topic we identified documents with event hoppers of the same type and subtype. Within that pool we then selected a "seed hopper", while hoppers from all other documents in the pool became "candidate hoppers". Annotators judged each candidate hopper against the seed hopper and made a coreference decision. Candidate hoppers that were not co-referred with the current seed hopper then became available as candidates or seeds for the next iteration of coreference. The iterations continued until the annotators had judged all hoppers in the pool against one another.

Table 2 shows the total number of pools, event hoppers and hopper pairs for cross-document event coreference annotation. The actual number of pairs annotated is smaller than the total number of event hopper pairs due to the exclusion of the non-seed hoppers from the seed document in each iteration and hoppers that were judged as coreferential from previous iterations.

Language	Pools	Hoppers	Max. hopper pairs	Annotated pairs
Chinese	140	1643	21761	14217
English	166	2454	49124	27946
Spanish	143	1234	11613	7265
Total	449	5331	82398	49428

Table 2: Pool, hopper and hopper pair counts for cross-document annotation

4.3 Cross-lingual Annotation

After cross-document coreference annotation within each language was completed, we clustered the coreferential event hoppers into cross-document event hopper clusters.

An event hopper cluster contains two or more event hoppers from different documents. These event hopper clusters were candidates for cross-lingual linking annotation. Event hoppers that were not included in event hopper clusters were considered as singleton event hoppers and not fed into cross-lingual linking annotation. We pivoted cross-lingual coreference though English by linking Chinese and Spanish hopper clusters to English hopper clusters. No additional direct linking of Chinese and Spanish hoppers was performed due to the scarcity of Chinese-Spanish bilingual specialists. Clusters were linked if they were

- From documents of the same topic
- The same event type and subtype
- Coreferential according to event hopper criteria

Table 3 shows the total number of pools, event hoppers and hopper pairs for cross-lingual event coreference annotation.

Language	Pools	Hopper clusters	Annotated pairs
Cmn_eng	29	102	237
Spa_eng	25	87	217
Total	54	189	454

Table 3: Pool, hopper and hopper pair counts for cross-lingual annotation

5. Results and Challenges

This cross-document annotation effort resulted in 892 coreference pairs and 389 event hopper clusters, as shown in Table 3.

Language	Total pairs	Coreferen -tial pairs	Total hoppers	Hopper clusters
Chinese	14217	256	1643	108
English	33527	423	2454	195
Spanish	7265	213	1234	86
Total	55009	892	5329	389

Table 4: Annotation results for cross-document coreference annotation

Out of a total of 55,009 hopper pairs, only 892 pairs were judged as coreferential, with a coreference ratio of 1.6%. Certain event types and subtypes have a higher coreference ratio – for example, the subtypes of the Personnel type: Elect 32%, StartPosition 16%, and EndPosition 24%.

The coreferenced event hoppers resulted in a total of 389 hopper clusters, with most hopper clusters containing 2-3 coreferenced event hoppers. The largest hopper cluster in Chinese had 15 coreferenced event hoppers, in Spanish 12, and in English 16.

The cross-lingual event hopper linking annotation based on the 189 cross-document hoppers resulted in a total of 28 hopper clusters, with 13 tri-lingual clusters and 15 bi-lingual clusters.

Although we selected a dataset of documents sharing the same topics to increase the chance of cross-document event hopper coreference in the data, there were still relatively few coreferenced event hoppers in this data. We took advantage of an existing topic-annotated corpus for this work, but the topics were not originally designed for this coreference task. A more suitable corpus for future work would have more specifically targeted topics that are more aligned to the event ontology that we are working with. The Personnel type had a high ratio of coreference in this corpus in part because this particular event type aligned well with the pre-existing topics (e.g., Angela Merkel's Third Electoral Win, Presidential Election of Nicholas Maduro, Taiwan Presidential Election).

Resource limitations did not permit dual annotation and calculation of inter-annotator agreement numbers during the initial pilot, but this is a necessary part of any future work.

5.1 Annotation Efficiency

During annotation, annotators judged one pair of event hoppers at a time. Some of the pools had many more event hoppers than others, which resulted in a substantial number of pairs to be judged. Such pools can also have a very long tail of judgments to be made through the iterations, and this is time consuming for annotation. Figure 2 illustrates the long tails of some event hopper pools. The reason that

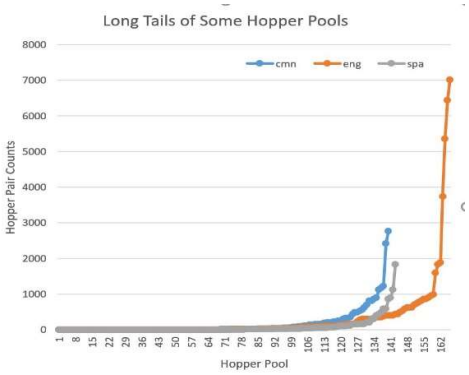


Figure 2: Some hopper pools have very long tails

English had more pools with long tails is that the English pools tend to be larger in this corpus, the largest being the Conflict.Attack pools for the "Syrian Revolution" and "Egypt Protests" topics, with 114 and 119 event hoppers.

5.2 Aggregate Events and Subevents

Drawing the distinction between subevents of aggregate events and events that are intuitively the same but with arguments of differing granularity remains a challenge for annotation, more so for cross-lingual annotation. An aggregate event is an event that may be composed of two or more subevents, which are only partially coreferential. Aggregated events and subevents have a parent-child relation, and the subevents themselves have a sister-like relation (Araki et al., 2014).

In the example below, event hopper 1 is the aggregate of protests (Conflict.Demonstration) occurring throughout the country. Each of the subsequent hoppers refers to a separate subevent of the aggregate event in eh1.

> S1: **Protests** (eh1) broke out throughout Bolivia over an increase in gas prices.
>
> S2: The **march** (eh2) in the capital began peacefully but clashes with police erupted near the main plaza where the government palace is located.

S3: Taxi drivers held a **strike** (eh3) that largely paralyzed La Paz on Thursday.

Often the challenge of aggregated events and subevents arises when the same aggregate event occurs at different times or places. For example, the Conflict.Demonstrate events in S1, S2 and S3 are co-referred, as they occur in about the same time period and same place, but the event in S4 doesn't belong to this event hopper cluster as it happened at a different time, although it was a subevent of the same aggregate event "protest in Egypt".

S1: Morsi's supporters, who have been holding sit-ins and **demonstrations** (eh1) since the president's ouster. (2013-07-16)

S2: 穆尔西和穆兄会的支持者在多座城市发起大规模**游行** (eh2). (2013-07-19)
*Supporters of Morsi and the Muslim Brotherhood launched big **protests** in multiple cities.*

S3: 土耳其民众举行集会声援埃及**示威** (eh3)者 (2013-08-17)
*People in Turkey gathered to support Egyptian **protesters**.*

S4: 当天，在开罗、亚历山大、苏伊士、法尤姆等省份爆发了不同规模的**示威游行** (eh4) (2013-12-14)
*On that same day, **protests** erupted in Cairo, Alexandria, Suiz, Faiyum and other provinces.*

Annotators should not place the aggregate event in the same hopper as any of its subevents, and likewise should not place the subevents in a hopper with each other. However, it is sometimes difficult to distinguish subevents as opposed to the argument granularity examples in Section 3.1, and this is an area that we will continue to investigate.

6. Future Directions

The work reported here has suggested three directions for future work: First, the annotation pipeline for cross-document/cross-lingual event coreference could be further optimized by leveraging existing entity linking annotation of event arguments (Ellis et al., 2016). It is expected that event hoppers sharing arguments that are linked to the same node of an entity knowledge base would have a greater likelihood of being coreferent.

Second, developing a KB of events, or events that occur within each topic, would allow document-level event hoppers to be linked to the KB. Similar to work that has been done for Entity Detection and Linking (EDL) (Ji et al., 2010), linking to such a KB of events would reduce the need to compare every relevant document-level event hopper to every potentially coreferent hopper, since many document-level hoppers could be linked directly to the event KB. The remaining document-level event hoppers that are not found in the KB would still need to be coreferenced via a pairwise comparison as in this paper (as NIL clusters are created for EDL entities). This direction, however, would require building such a KB before annotation. Giraldi et al. (2014) adopted this approach and demonstrated feasibility, but the event and coreference definitions used were quite different from the ERE framework.

Third, the cross-document/cross-lingual coreferenced event hoppers that are the result of this process can now be used in other event-event relations, such as part-whole, causation, or event sequencing. Using corpus-wide coreferenced event hoppers (rather than individual event mentions) as the arguments of event-event relations would allow for a corpus-wide view of events and event relations, which is critical for corpus-wide evaluation and understanding (Hong et al., 2016).

7. Conclusion

We created a small corpus annotated for cross-document and cross-lingual event coreference in 505 documents in three languages. Although we leveraged existing ERE annotation as input, this task required the development of new annotation guidelines, new data processes and user interfaces, and the creation of new cross-document and cross-lingual annotation. The more intuitive, coarser grained event hopper concept that was originally developed as part of within-document Rich ERE annotation (Song et al., 2015) has proven to be well suited for the type of event coreference that is possible across documents and across languages. We plan to continue with corpus-wide event coreference using the event hopper concept.

Although this corpus is relatively small so far, it does provide data in support of developing a corpus-wide understanding of events and the entities participating in those events. The corpus includes both positive examples of corpus-wide event hopper coreference, both cross-document and cross-lingual, and also negative coreference judgements of many more potential event pairs. Positive and negative judgements both provide useful training data for identifying corpus-wide events and event argument clusters.

The cross-document event coreference annotation described in this paper has been distributed to DEFT performers and to participants in the 2017 NIST TAC KBP evaluations. Following its use in these evaluations, the data and annotations will be published in LDC's public catalog, making the resource broadly available for language-related research, education and technology development.

8. Acknowledgements

This material is based on research sponsored by Air Force Research Laboratory and Defense Advanced Research Projects Agency under agreement number FA8750-13-2-0045. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory and Defense Advanced Research Projects Agency or the U.S. Government.

We also thank Jonathan Wright for building the cross-document annotation tool that makes the annotation possible.

9. Bibliographical References

Aguilar, J., Beller, C., McNamee, P., Van Durme, B., Strassel, S., Song, Z., & Ellis, J. (2014). A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. In Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation (pp. 45-53).

Araki, J., Liu, Z., Hovy, E. H., & Mitamura, T. (2014). Detecting Subevent Structure for Event Coreference Resolution. In Proceedings of LREC 2014.

Bejan, AC. and Harabagiu, S. (2008). A Linguistic Resource for Discovering Event Structures and Resolving Event Coreference. In Proceedings of LREC 2008.

Bies, A., Song, Z., Getman, J., Ellis, J., Mott, J., Strassel, S., Palmer, M., Mitamura, T., Freedman, M., Ji, H., and O'Gorman, T. (2016). A Comparison of Event Representations in DEFT. In Proceedings of the Fourth Workshop on Events (pp. 27-36).

Cybulska, A., & Vossen, P. (2014a). Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In Proceedings of LREC 2014.

Cybulska, A., & Vossen, P. (2014b). Guidelines for ECB+ annotation of events and their coreference. Technical Report NWR-2014-1, VU University Amsterdam.

DARPA. 2012. *Broad Agency Announcement: Deep Exploration and Filtering of Text (DEFT)*. Defense Advanced Research Projects Agency, DARPA-BAA-12-47.

Ellis, J., Getman, J., Fore, D., Kuster, N., Song, Z., Bies, A., & Strassel, S. (2016). Overview of Linguistic Resources for the TAC KBP 2016 Evaluations: Methodologies and Results. In Proceedings of TAC KBP 2016 Workshop, National Institute of Standards and Technology.

Getman, J., Ellis, J., Strassel, S., Song, Z., & Tracey, J. (2018). Laying the Groundwork for Knowledge Base Population: Nine Years of Linguistic Resources for TAC KBP. In Proceedings of LREC 2018.

Girardi, C., Speranza, M., Sprugnoli, R., & Tonelli, S. (2014). CROMER: a Tool for Cross-Document Event and Entity Coreference. In Proceedings of LREC 2016.

Hong, Y., Zhang, T., O'Gorman, T., Horowit-Hendler, S., Ji, H., & Palmer, M. (2016). Building a Cross-document Event-Event Relation Corpus. LAW X, 1.

Ji, H., Grishman, R., Dang, H. T., Griffitt, K., & Ellis, J. (2010). Overview of the TAC 2010 Knowledge Base Population Track. In Third Text Analysis Conference (TAC 2010).

Lee, H., Recasens, M., Chang, A., Surdeanu, M., & Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 489-500). Association for Computational Linguistics.

Linguistic Data Consortium. (2005). *ACE (Automatic Content Extraction) English Annotation Guidelines for Events Version 5.4.3*.

Mott, J., Bies, A., Song, Z., & Strassel, S. (2016). Parallel Chinese-English Entities, Relations and Events Corpora. In Proceedings of LREC 2016.

NIST. (2016). TAC 2016 Cold Start KB Track. https://tac.nist.gov/2016/KBP/ColdStart/index.html. March 2017. Accessed September, 2017.

NIST. (2017). TAC 2017 Cold Start KB Track. https://tac.nist.gov/2017/KBP/ColdStart/index.html. May 2017. Accessed September, 2017.

Pradhan, S. S., Ramshaw, L., Weischedel, R., MacBride, J., & Micciulla, L. (2007). Unrestricted coreference: Identifying entities and events in OntoNotes. In Semantic Computing, 2007. ICSC 2007. International Conference on (pp. 446-453). IEEE.

Pustejovsky, J., Castano, J.M., Ingria, R., Sauri, R., Gaizauskas, R.J., Setzer, A., Katz, G. and Radev, D.R. (2003). TimeML: Robust specification of event and temporal expressions in text. New directions in question answering, 3, 28-34.

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In Proceedings of the 3rd Workshop on Events at the NAACL-HLT.

10. Language Resource References

Walker, C., Strassel, S., Medero, J., and Maeda, K. (2006). *ACE 2005 Multilingual Training Corpus*. Linguistic Data Consortium, LDC Catalog No.: LDC2006T06.

TAP-DLND 1.0 : A Corpus for Document Level Novelty Detection

Tirthankar Ghosal, Amitra Salam, Swati Tiwari, Asif Ekbal, Pushpak Bhattacharyya
Indian Institute of Technology Patna
Bihta, Bihar-801106, India
(tirthankar.pcs16, salam.recs, stlex.re, asif, pb)@iitp.ac.in

Abstract

Detecting novelty of an entire document is an Artificial Intelligence (AI) frontier problem. This has immense importance in widespread Natural Language Processing (NLP) applications ranging from extractive text document summarization to tracking development of news events to predicting impact of scholarly articles. Although a very relevant problem in the present context of exponential data duplication, we are unaware of any document level dataset that correctly addresses the evaluation of automatic novelty detection techniques in a classification framework. To bridge this relative gap, here in this work, we present a resource for benchmarking the techniques for *document level novelty detection*. We create the resource via *topic-specific crawling* of news documents across several domains in a periodic manner. We release the annotated corpus with necessary statistics and show its use with a developed system for the problem in concern.

Keywords: Document level novelty detection, Classification, Web crawling, Corpus

1. Introduction

Novelty detection implies finding elements that have not appeared before, or new, or original with respect to relevant references. The explosive growth of documents across the web has resulted in the accumulation of redundant ones, thereby consuming space as well as precious time of readers seeking new information. This necessitates finding means for discarding redundant document(s) and retaining ones containing novel information. The level of information duplication is not just limited to the lexical surface form of texts but has encroached the barriers of semantics and pragmatics too. Paraphrasing, semantic level plagiarism etc. are instances of such practices. Intelligent text reuse, synonym replacement and careful alignment may lead to a surface form which is very different from the originating source yet convey the same meaning. Present state-of-the-art text matching techniques are unable to process such redundancy. The quest of new information is an eternal human need and urges attention in this very age of exploratory data redundancy. One major objective of this work is to provide a benchmark setup for experiments to filter out superfluous information across the web. With this work we introduce a simplistic dataset to the research community to inculcate efficient methods to detect *document level novelty* or on the contrary document level redundancy. We create the resource by crawling news articles of different categories by various agencies and coin it as *TAP-DLND 1.0*[1] (after the initial names of the principal investigators *Tirthankar-Asif-Pushpak*) which also stands for *Explore Document Level Novelty Detection (DLND)*. In this work we view the problem of novelty detection as a two-class classification problem with the judgment that whether an incoming document bears sufficiently new information to be labeled as novel with respect to a set of source documents. The source document set could be seen as the memory of the reader which stores known information. We extract features from target documents with respect to their corresponding source documents and develop a classification system. Our results are promising and can serve as robust baseline to further research in this topic.

[1] http://www.iitp.ac.in/∼ai-nlp-ml/resources.html

2. Related Works

Although sentence level novelty detection is a well studied problem in information retrieval literature, very little has been done to address the problem at the document level. To begin with (Li and Croft, 2005) rightly pointed out that, research in novelty detection from texts has been carried out at three levels : event level, sentence level and document level. Research in novelty mining could be traced back to the Topic Detection and Tracking (TDT) (Allan, 2002) evaluation campaigns where the concern was to detect new event from online news streams. Although the intention was to detect the *first story* or reporting of a new event from a series of news stories, the notion of *novelty detection* from texts came into light for the research community. Some notable approaches for New Event Detection with the TDT corpus are by (Allan et al., 1998; Yang et al., 2002; Stokes and Carthy, 2001; Franz et al., 2001; Yang et al., 1998; Allan et al., 2000; Brants et al., 2003). However, the Novelty track in Text Retrieval and Evaluation Conferences (TREC) (Soboroff and Harman, 2005) were the first to explicitly explore the concept of Novelty Detection from texts. Under the paradigm of information retrieval, given a query, the TREC experiments were designed to retrieve relevant and novel sentences from a given collection. Some notable approaches for sentence level novelty detection from the TREC exercises are by (Allan et al., 2003; Kwee et al., 2009; Li and Croft, 2005; Zhang et al., 2003; Collins-Thompson et al., 2002; Gabrilovich et al., 2004; Ru et al., 2004). Textual Entailment based sentence level novelty mining was explored in the novelty subtask of Recognizing Textual Entailment-Text Analytics Conference (RTE-TAC) 6 and 7 (Bentivogli et al., 2011). The datasets made available from these tasks are for sentence level novelty mining and were created from an information retrieval perspective. At the document level the

problem is attempted by a few like (Zhang et al., 2002; Tsai and Zhang, 2011; Karkali et al., 2013; Dasgupta and Dey, 2016). However none of the datasets used in these works are publicly available. Hence we find that there is a dearth of a proper evaluation setup (e.g. corpus, baseline and evaluation methods) for document level novelty detection. This inspired us to create one and establish a benchmark for the same.

3. Motivation and Contribution

Our understanding and survey revealed that in spite of having several applications in NLP tasks, novelty detection at the document level has not attracted the coveted attention. Hence, we deem that novelty at the document level needs to be understood first, investigated in-depth, and benchmark setup (gold standard resources) be created to validate the investigations. We hope that the knowledge gained from this dataset and experiments would be a step towards our more ambitious vision of semantic level plagiarism detection in scholarly articles. We briefly outline the contributions of this work :

- Proposing a benchmark dataset for document level novelty detection. We are unaware of the availability of any such corpus; and
- Presenting a supervised machine learning model for document level novelty detection. This can be treated as a baseline model for further research along this line.

4. Document Level Novelty

Novelty detection from texts implies search for new information with respect to whatever is already known or seen. Hence, the problem of novelty detection from texts is very subjective and depends upon the view of the intended reader. The knowledge of the reader regarding a particular event serves as the reference against which s/he decides the novelty of an incoming information. Careful observation of data characteristics led us to believe that there exists at least four properties that characterizes novelty detection from texts :

- *Relevance*
- *Diversity*
- *Relativity* and
- *Temporality*

4.1. Relevance

The target document should be relevant to prior knowledge. For example, seeking novelty between two documents, one talking about *jaguar*, the animal and the other about *jaguar*, the car is futile as one is not relevant to the other. Quite obvious that each one would contain different information than the other. So *Relevance* should hold.

4.2. Diversity

Diversity correlates with the new information content. More the new information in a document, diverse would be the content. Hence novel information should be relevant yet *diverse* from existing information.
For example, let us consider, on a given date a certain newswire document X reports about an accident at a certain place. On the subsequent date another reporting X' surfaces which details about the investigation being carried out by police. Now X' will contain new information with respect to X. That is to say, given a reader has already read about the first reporting (facts of the accident) X, the second reporting X' having significant different content as well as different direction of reporting (or intent) would appear novel to the reader. So X' is relevant to X yet divergent i.e. containing new information.

4.3. Relativity

The amount of new information content is important while deciding the novelty of an entire document. When we talk about a document being novel it is always with respect to a reference set of documents already seen (information already gained from those seen documents) or what we say as the knowledge base of the reader. So the quantity of *relative* new information plays a role for deciding document novelty.

4.4. Temporality

Finally novel information is usually a temporal update over existing knowledge. The previous example justifies the view.

With these notion of novelty we went on to create a resource that effectively taps these properties, *viz.,* **Relevance**, **Diversity**, **Relativity** and **Temporality**. Our resource not only encompasses the lexical form of redundancy (a straight forward form of *non-novelty*) but also delves deep into semantic textual redundancy (a more complex form of *non-novelty*). To understand it better let us consider the following example :

- *d1 : Singapore is an island city-state located at the southern tip of the Malay Peninsula. It lies 137 kilometers north of the equator.*
- *d2 : Singapore's territory consists of one main island along with 62 other islets. The population in Singapore is approximately 5.6 million.*
- *d3 : Singapore is a global commerce, finance and transport hub. Singapore has a tropical rain forest climate with no distinctive seasons, uniform temperature and pressure, high humidity, and abundant rainfall.*
- *d4 : Singapore, an island city-state off southern Malaysia, lies one degree north of the equator. As of June 2017, the island's population stood at 5.61 million.*

It is fairly easy to conclude that *d4* follows from *d1* and *d2*. However, considering only *d3* as the source, although related but *d4* has entirely diverse information. Thus, *d4*

would be non-novel with respect to *d1* and *d2* but would appear novel with respect to *d3* only. We build our corpus along this line and the annotations too followed this kind of judgments.

5. Benchmark Setup : TAP-DLND 1.0

To address the issues pointed out in the previous section we develop a benchmark setup as discussed below.

5.1. Data Collection

We design a web crawler[2] to perform systematic, unbiased, *event-specific* crawling of news articles, mostly from the online versions of Indian English newspapers. The news domains we looked into are : *Accident (ACC), Politics (PLT), Business (BUS), Arts and Entertainment (ART), Crime (CRM), Nature (NAT), Terrorism (TER), Government (GOV), Sports (SPT), and Society (SOC)*. To ensure that *Temporality* criteria is preserved, our web crawler is designed to fetch web documents for a certain event in a timely manner i.e. the crawled documents are grouped as per their dates of publications in different forums (See Figure 1). Event wise statistics of the corpus are in Table 2.

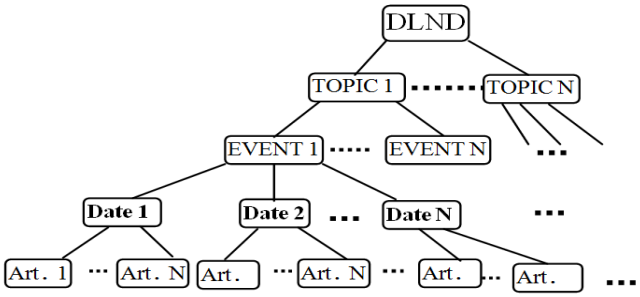


Figure 1: Temporal Crawling

Features	Statistics
Crawling period	Nov'16 - Nov'17
Number of events	223
Number of sources per event	3
Total novel documents	2736
Total non-novel documents	2704
Total documents in TAP-DLND 1.0	6109
Average number of sentences	15
Average number of words	353

Table 1: Statistics of TAP-DLND 1.0 corpus. Here, average number of sentences and words is per document.

[2]using the www.webhose.io API

Category	# Events	# N	# NN
ACC	10	231	272
PLT	97	669	685
BUS	35	202	264
ART	21	397	258
CRM	10	237	174
NAT	10	87	250
TER	18	255	468
GOV	15	405	219
SPT	2	39	51
SOC	5	214	63

Table 2: Event wise statistics of TAP-DLND 1.0, $\#N \rightarrow$ Number of Novel documents, $\#NN \rightarrow$ Number of Non-Novel documents

5.2. Preprocessing

As the data were crawled from various web sources[3] we perform some manual preprocessing works like removal of headlines, news source, date, time, noises (advertisements, images, hyperlinks) and convert the data into desired shape.

5.3. Source Document Selection

To mandate the *Relevance* and *Relativity* criteria, we select three documents for each event as the seed source documents. They are usually selected from the initial dates of reporting of a particular event. Also so chosen that they represent different facets of information regarding that particular event (*information coverage*). These source documents serve as the reference against which we asked the annotators to tag a target document (chosen from the remaining crawled documents for that event) as *novel* or *non-novel*. The source documents could be perceived as the memory of the reader or information already known against which it is to be determined with reasonable level of certainty that whether a target document contains sufficient new information to be labeled as *novel*.

5.4. Renaming files

For ease of information retrieval we rename each document in the corpus. A certain document bearing **ACCE005SRC003.txt** as file name indicates that it is the 3rd source document of the 5th event in the *accident category*. For target documents SRC is replaced by TGT.

5.5. Meta files

We generate meta files (.xml) for each document in the corpus. These meta files contain background information regarding a source/target document within structured XML tags and have the same file name as that of the corresponding document. The information content of the meta files are : *date of publishing, publisher, title of reporting, source*

[3]List of few news sources : www.ndtv.com, indianexpress.com, timesofindia.indiatimes.com, indiatoday.intoday.in, thehindu.com, news18.com, firstpost.com, dnaindia.com, deccanchronicle.com, financialexpress.com, business-standard.com, sify.com, newskerala.com, mid-day.com, thedailystar.net, theweek.in, tribuneindia.com

id, event id, event name, category, Document Level Annotation (DLA), number of words and sentences. We develop a semi-automatic *meta file generator interface* where attribute values are automatically captured from the hierarchically organized data (See Figure 2). Stanford CoreNLP (Manning et al., 2014) integrated with our interface gave us the field values for *sentence* and *word* count. We asked our annotators to provide their judgments for the *DLA* attribute based on the guidelines specified in the next section.

5.6. Annotation

Three annotators with post-graduate level knowledge in English were involved in labeling the TAP-DLND 1.0 target documents. Having read the source document(s) we asked the annotators to annotate an incoming on-event document as *non-novel* or *novel* solely based on the information coverage in the source documents. The **annotation guidelines** were simple:

1. To annotate a document as *non-novel* whose semantic content significantly overlaps with the source document(s) (maximum redundant information).

2. To annotate a document as *novel* if its semantic content as well as intent (direction of reporting) significantly differs from the source document(s)(minimum or no information overlap). It could be an update on the same event or describing a post-event situation.

3. To leave out the ambiguous cases (for which the human annotators were not sure about the label).

Two annotators independently labeled the target documents. The third annotator resolved the differences via majority voting. We found that novel items with respect to the source documents were mostly found in the reporting published in subsequent dates. Whereas non-novel items we found in the reporting published by different agencies in the same date as that of the source documents. This is in line with the *Temporality* criteria we discussed earlier. The inter-annotator agreement ratio was found to be **0.82** in terms of **Kappa coefficient** (Fleiss, 1971) which is assumed to be good as per (Landis and Koch, 1977). The final structure of DLND is in Figure 2.

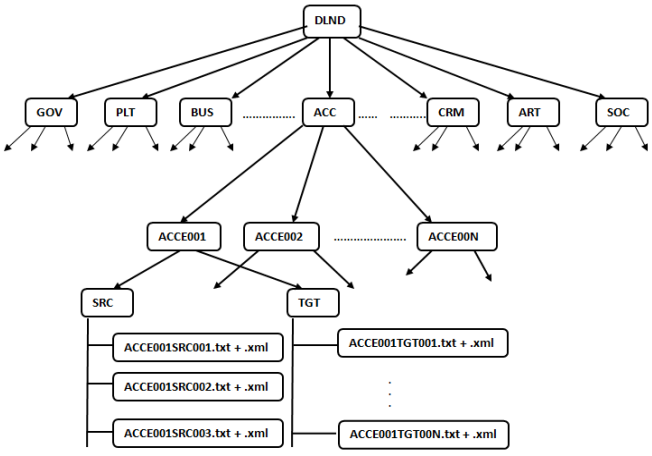


Figure 2: The DLND corpus structure

6. Evaluation

We frame document level novelty detection as a binary classification problem and choose the features in parlance with the objective nature of texts that we consider for our experiments. We develop a binary classifier based on Random Forest[9] (RF) (Breiman, 2001) algorithm that classifies a document into either *novel* or *non-novel*. Our key focus is on extracting features that contribute to the semantics of a document. The set of features that we use for training and/or testing RF is listed in Table 3. As is evident from the discussion in Section 3, TAP-DLND 1.0 consists a fair share of different levels (lexical as well as semantic) of text representations.

6.1. Results and Discussions

We first take a simple yet popular lexical baseline: Jaccard similarity with unigrams between the source document and the target (Zhang et al., 2003). We train a Logistic Regression (LR) classifier with the Jaccard score to classify a document based on its overlap with the source document. Table 4 clearly indicates that the lexical baseline fails miserably in identifying *non-novel* documents. Next we went ahead with three approaches by (Zhang et al., 2002) for novelty detection at the document level. The first one i.e. the Set Difference is essentially the count of new words in the target document with respect to the set of source document(s). For this we concatenate the source document(s) of each event to form one source against each target. The Geometric Distance measures the cosine similarity between two document vectors represented as *tf-idf* vectors. For three source documents against one target document in TAP-DLND 1.0, we take the maximum of the cosine similarity score. The third approach measures the Kullback-Leibler divergence between the concatenated source document(s) and the prospective target document where a document d is represented as a probabilistic unigram word distribution (language model θ_d). Instead of setting a fixed threshold as (Zhang et al., 2003), we train a Logistic Regression classifier based on those measures to automatically determine the decision boundary. Another approach by (Karkali et al., 2013) based on Novelty Scoring via Inverse Document Frequency (IDF) performed poorly in recognizing novel/non-novel documents in TAP-DLND 1.0. We also compare our method with a more recent approach of (Dasgupta and Dey, 2016) on our data. This particular entropy-based approach produces novelty score (NS) of a document d with respect to a collection C. We adapt their respective threshold criteria and infer that documents with novelty score above (*average+standard deviation*) are *novel* and that with novelty score below (*average-standard deviation*) are *non-novel*.

[4]Distributed Bag-Of-Words (DBOW) paragraph vector model trained on Wikipedia articles.

[5]Trained on Google News Corpus of 100 billion words. 300 dimension vectors using CBOW model

[6]Entities were extracted using the Stanford Tagger.

[7]Using the Rapid Automatic Keyword Extraction (RAKE) algorithm.

[8]Obtained from English WordNet (Fellbaum, 1998)

[9]RF of 100 trees with minimum number of instances per leaf set to 1 implemented in WEKA machine learning toolkit

Type	Features	Description
Semantic	Paragraph Vector (pv) + Cosine	We represent the source and target documents in terms of *paragraph vectors*[4](Le and Mikolov, 2014). Then we take the maximum of the cosine similarity between the source-target pairs.
Semantic	Concept Centrality	To identify the central theme of a document we use the *TextRank* summarization algorithm by (Mihalcea and Tarau, 2004). Thereafter we vectorize the ranked summary for each source and target document by simple *word2vec*[5](Mikolov et al., 2013) concatenation. Finally we take the maximum of the cosine similarity between the source and target vectors.
Lexical	n-gram similarity	We compute lexical overlap of target *n-gram*'s with respect to source documents for n = 2,3 and 8. Octagrams we use to put emphasis on phrase overlap.
Lexical	Named Entities and Keywords match (kw-ner)	As Named Entities[6] and Keywords[7] play a significant role in determining *relevance*, we put additional weightage to them by considering their match (target w.r.t. sources) as a separate feature.
Lexico-Semantic	New Word Count (nwc)	The number of new words could be an effective indicator of the amount of novel information content in the target document w.r.t. the source(s) given. Here, for calculating new words, along with the surface forms, we consider their synonyms[8] as well to establish semantic relatedness.
Language Model	Divergence (kld)	We use this feature to measure the dissimilarity between two documents represented as language models. We concatenate all the source documents into one and then measure the Kullback-Leibler Divergence with the target.

Table 3: Feature Set

Systems	P(N)	R(N)	F_1(N)	P(NN)	R(NN)	F_1(NN)	Accuracy
Jaccard+LR (Baseline)	52.2	96.1	67.6	74.0	10.9	19.0	53.8
Set Difference+LR (Zhang et al., 2002)	74.3	71.5	72.8	72.2	74.9	73.5	73.2
Geometric Distance+LR (Zhang et al., 2002)	65.6	84.3	73.7	84.2	55.3	66.7	69.8
Language Model (KLD)+LR (Zhang et al., 2002)	73.2	74.9	74.1	74.0	72.3	73.1	73.6
Novelty (IDF)+LR (Karkali et al., 2013)	52.5	92.1	66.9	66.5	15.9	25.6	54.2
(Dasgupta and Dey, 2016)	65.1	63.8	64.4	64.1	65.3	64.6	64.5
Proposed Approach (RF)	**77.6**	82.3	**79.8**	80.9	**76.1**	**78.4**	79.2

Table 4: *10-fold* cross-validation results on TAP-DLND 1.0 (in %), $P \rightarrow Precision$, $R \rightarrow Recall$, $N \rightarrow Novel$, $NN \rightarrow Non-Novel$, $LR \rightarrow$ Logistic Regression, $IDF \rightarrow$ Inverse Document Frequency, $KLD \rightarrow$ Kullback- Leibler Divergence

We left out the remaining (average novelty class) cases for our experiments. Table 4 numbers clearly show that our method superseded the baselines and purported *state-of-the-art* by a substantial margin.

6.2. Feature Significance

We investigated the significance of each feature by measuring the Information Gain (See Figure 3). The information gain for a feature x_k is the expected reduction in entropy-i.e., uncertainty achieved by learning the state of that feature. We attribute the better performance of our approach to the choice of semantic features for our experiments (see Figure 3). Lexico-Semantic feature new word count has the maximum contribution, for which we argue that novel events in context to newspaper articles would contain new entities, concepts, numbers whereas non-novel documents would consist identical or synonymous entities. Semantic features play a vital role which indicates that detection of novelty extends beyond lexical characteristics of text.

7. Conclusion

In this work we put forward a benchmark resource for *document level novelty detection* and an evaluation scheme for the same. Our resource has an extensive coverage of ten different news categories and also includes the *relevance, diversity, relativity,* and *temporality* criteria inherently within its schema. Along with straightforward lexical characteristics it also manifests the high level semantic understanding of human annotators in its gold labels which is very essential for detecting semantic level redundancy. We hope that TAP-DLND 1.0 would evolve as a benchmark resource for experiments on document level novelty detection and provide valuable insights into the problem. In future we plan to annotate the TAP-DLND 1.0 corpus at the sentence level

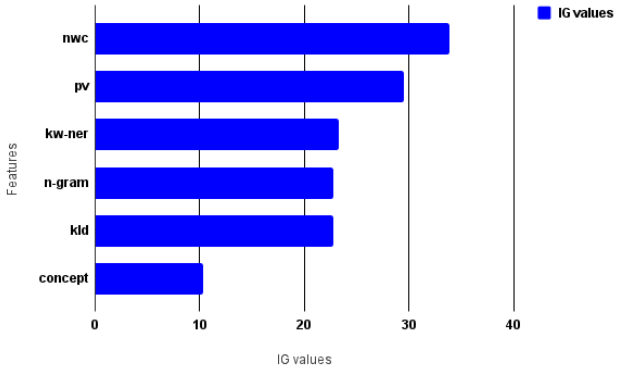


Figure 3: Significance of features *based on Information Gain (IG)*. The length of the bar corresponds to the average merit (X : IG) of the feature (: Y).

to have more fine perception regarding the amount of new information required to deem a document as *novel*. Also we intend to include more target documents in data scarce categories.

8. Availability

The dataset is submitted to European Language Resource Association (ELRA) for hosting. Also would be available at http://www.iitp.ac.in/∼ai-nlp-ml/resources.html. Researchers can also contact the first author for the resource.

9. Acknowledgements

Tirthankar Ghosal and Asif Ekbal gratefully acknowledge Visvesvaraya PhD scheme for Electronics and Information Technology, an initiative of Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia). Tirthankar is a PhD scholar under Visvesvaraya PhD scheme and Asif is the recipient of Sir Visvesvaraya Young Faculty Research Fellow Award. The work is also generously supported by Elsevier Centre of Excellence for Natural Language Processing, Department of Computer Science and Engineering, Indian Institute of Technology Patna. We would also like to thank the anonymous reviewers for their valuable suggestions in improving this work.

10. Bibliographical References

Allan, J., Papka, R., and Lavrenko, V. (1998). On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 37–45. ACM.

Allan, J., Lavrenko, V., and Jin, H. (2000). First story detection in tdt is hard. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 374–381. ACM.

Allan, J., Wade, C., and Bolivar, A. (2003). Retrieval and novelty detection at the sentence level. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 314–321. ACM.

Allan, J. (2002). Introduction to topic detection and tracking. *Topic detection and tracking*, pages 1–16.

Bentivogli, L., Clark, P., Dagan, I., and Giampiccolo, D. (2011). The seventh pascal recognizing textual entailment challenge. In *TAC*.

Brants, T., Chen, F., and Farahat, A. (2003). A system for new event detection. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 330–337. ACM.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Collins-Thompson, K., Ogilvie, P., Zhang, Y., and Callan, J. (2002). Information filtering, novelty detection, and named-page finding. In *TREC*.

Dasgupta, T. and Dey, L. (2016). Automatic scoring for innovativeness of textual ideas. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Franz, M., Ittycheriah, A., McCarley, J. S., and Ward, T. (2001). First story detection: Combining similarity and novelty based approaches. In *Topic Detection and Tracking Workshop Report*, pages 193–206.

Gabrilovich, E., Dumais, S., and Horvitz, E. (2004). Newsjunkie: providing personalized newsfeeds via analysis of information novelty. In *Proceedings of the 13th international conference on World Wide Web*, pages 482–490. ACM.

Karkali, M., Rousseau, F., Ntoulas, A., and Vazirgiannis, M. (2013). Efficient online novelty detection in news streams. In *WISE (1)*, pages 57–71.

Kwee, A. T., Tsai, F. S., and Tang, W. (2009). Sentence-level novelty detection in english and malay. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 40–51. Springer.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Li, X. and Croft, W. B. (2005). Novelty detection based on sentence level patterns. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 744–751. ACM.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. Association for Computational Linguistics.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Ru, L., Zhao, L., Zhang, M., and Ma, S. (2004). Improved feature selection and redundance computing-thuir at trec 2004 novelty track. In *TREC*.

Soboroff, I. and Harman, D. (2005). Novelty detection: the trec experience. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 105–112. Association for Computational Linguistics.

Stokes, N. and Carthy, J. (2001). First story detection using a composite document representation. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.

Tsai, F. S. and Zhang, Y. (2011). D2s: Document-to-sentence framework for novelty detection. *Knowledge and information systems*, 29(2):419–433.

Yang, Y., Pierce, T., and Carbonell, J. (1998). A study of retrospective and on-line event detection. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 28–36. ACM.

Yang, Y., Zhang, J., Carbonell, J., and Jin, C. (2002). Topic-conditioned novelty detection. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 688–693. ACM.

Zhang, Y., Callan, J., and Minka, T. (2002). Novelty and redundancy detection in adaptive filtering. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 81–88. ACM.

Zhang, M., Song, R., Lin, C., Ma, S., Jiang, Z., Jin, Y., Liu, Y., Zhao, L., and Ma, S. (2003). Expansion-based technologies in finding relevant and new information: Thu trec 2002: Novelty track experiments. *NIST SPECIAL PUBLICATION SP*, (251):586–590.

A Corpus for Multilingual Document Classification in Eight Languages

Holger Schwenk, Xian Li
Facebook AI Research, Facebook AML
{schwenk,xianl}@fb.com

Abstract

Cross-lingual document classification aims at training a document classifier on resources in one language and transferring it to a different language without any additional resources. Several approaches have been proposed in the literature and the current best practice is to evaluate them on a subset of the Reuters Corpus Volume 2. However, this subset covers only few languages (English, German, French and Spanish) and almost all published works focus on the the transfer between English and German. In addition, we have observed that the class prior distributions differ significantly between the languages. We argue that this complicates the evaluation of the multilinguality.

In this paper, we propose a new subset of the Reuters corpus with balanced class priors for eight languages. By adding Italian, Russian, Japanese and Chinese, we cover languages which are very different with respect to syntax, morphology, etc. We provide strong baselines for all language transfer directions using multilingual word and sentence embeddings respectively. Our goal is to offer a freely available framework to evaluate cross-lingual document classification, and we hope to foster by these means, research in this important area.

Keywords: cross-lingual document classification, multilinguality, Reuters Corpus Volume 2 (RCV2), evaluation framework

1. Introduction

There are many tasks in natural language processing which require the classification of sentences or longer paragraphs into a set of predefined categories. Typical applications are for instance topic identification (e.g. sports, news, ...) or product reviews (positive or negative). There is a large body of research on approaches for document classification. An important aspect to compare these different approaches is the availability of high quality corpora to train and evaluate them. Unfortunately, most of these evaluation tasks focus on the English language only, while there is an ever increasing need to perform document classification in many other languages. One could of course collect and label training data for other languages, but this would be costly and time consuming. An interesting alternative is *"cross-lingual document classification"*. The underlying idea is to use a representation of the words or whole documents which is independent of the language. By these means, a classifier trained on one language can be transferred to a different one, without the need of resources in that transfer language. Ideally, the performance obtained by cross-lingual transfer should be as close as possible to training the entire system on language specific resources. Such a task was first proposed by (Klementiev et al., 2012) using the Reuters Corpus Volume 2. The aim was to first train a classifier on English and then to transfer it to German, and vice versa. An extension to the transfer between English and French and Spanish respectively was proposed by (Mogadala and Rettinger, 2016). However, only few comparative results are available for these transfer directions.

The contributions of this work are as follows. We extend previous works and use the data in the Reuters Corpus Volume 2 to define new cross-lingual document classification tasks for eight very different languages, namely English, French, Spanish, Italian, German, Russian, Chinese and Japanese. For each language, we define a train, development and test corpus. We also provide strong reference results for all transfer directions between the eight languages, e.g. not limited to the transfer between a foreign language and English. We compare two approaches, based either on multilingual word or sentence embeddings respectively. By these means, we hope to define a clear evaluation environment for highly multilingual document classification.

2. Corpus description

The Reuters Corpus Volume 2 (Lewis et al., 2004), in short RCV2[1], is a multilingual corpus with a collection of 487,000 news stories. Each news story was manually classified into four hierarchical groups: CCAT (Corporate/Industrial), ECAT (Economics), GCAT (Government/Social) and MCAT (Markets). Topic codes were assigned to capture the major subject of the news story. The entire corpus covers thirteen languages, i.e. Dutch, French, German, Chinese, Japanese, Russian, Portuguese, Spanish, Latin American Spanish, Italian, Danish, Norwegian, and Swedish, written by local reporters in each language. The news stories are not parallel. Single-label stories, i.e. those labeled with only one topic out of the four top categories, are often used for evaluations. However, the class distributions vary significantly across all the thirteen languages (see Table 1). Therefore, using random samples to extract evaluation corpora may lead to very imbalanced test sets, i.e. undesired and misleading variability among the languages when the main focus is to evaluate cross-lingual transfer.

2.1. Cross-lingual document classification

A subset of the English and German sections of RCV2 was defined by (Klementiev et al., 2012) to evaluate cross-lingual document classification. This subset was used in several follow-up works and many comparative results are available for the transfer between German and English. (Mogadala and Rettinger, 2016) extended the use of RCV2 for cross-lingual document classification to the French and Spanish language (transfer from and to English). An analysis of these evaluation corpora has shown that the class prior

[1] http://trec.nist.gov/data/reuters/reuters.html

Language	Category ECAT	CCAT	GCAT	MCAT
English	6.2%	39.8%	29.5%	24.5%
German	6.4%	30.1%	40.9%	22.6%
French	6.3%	21.6%	60.2%	11.8%
Spanish	8.6%	15.0%	9.0%	67.3%
Chinese	19.7%	18.2%	2.8%	59.4%
Italian	18.0%	35.7%	9.5%	36.8%
Japanese	14.8%	42.1%	7.0%	36.1%
Russian	26.9%	27.6%	13.4%	32.2%
Danish	7.5%	56.6%	5.3%	30.6%

Table 1: Class distribution of all single-label stories per language of the entire Reuters Corpus Volume 2.

distributions vary significantly between the classes (see Table 2). For German and English, more than 80% of the examples in the test set belong to the classes GCAT and MCAT and at most 2% to the class CCAT. These class prior distributions are very different for French and Spanish: the class CCAT is quite frequent with 21% and 15% of the French and Spanish test set respectively. One may of course argue that variability in the class prior distribution is typical for real-world problems, but this shifts the focus from a high quality cross-lingual transfer to *"tricks"* for how to best handle the class imbalance. Indeed, in previous research the transfer between English and German achieves accuracies higher than 90%, while the performance is below 80% for EN/FR or even 70% EN/ES. We have seen experimental evidence that these important differences are likely to be caused by the discrepancy in the class priors of the test sets.

2.2. Multilingual document classification

In this work, we propose a new evaluation framework for highly multilingual document classification which significantly extends the current state. We continue to use Reuters Corpus Volume 2, but based on the above mentioned limitations of the current subset of RCV2, we propose new tasks for cross-lingual document classification. The design choices are as follow:

- **Uniform class coverage:** we sample from RCV2 the same number of examples for each class and language;
- **Split the data into train, development and test corpus:** for each languages, we provide training data of different sizes (1k, 2k, 5k and 10k stories), a development (1k) and a test corpus (4k);
- **Support more languages:** German (DE), English (EN), Spanish (ES), French (FR), Italian (IT), Japanese (JA), Russian (RU) and Chinese (ZH). Reference baseline results are available for all languages.

Most works in the literature use only 1 000 examples to train the document classifier. To invest the impact of more training data, we also provide training corpora of 2 000, 5 000 and 10 000 documents.[2] The development corpus for each language is composed of 1 000 and the test set of 4 000 documents respectively. All have uniform class distributions. An important aspect of this work is to provide a framework to study and evaluate cross-lingual document classification for many language pairs. In that spirit, we will name this corpus *"Multilingual Document Classification Corpus"*, abbreviated as MLDoc. The full Reuters Corpus Volume 2 has a special license and we can not distribute it ourselves. Instead, we provide tools to extract all the subsets of MLDoc at https://github.com/facebookresearch/MLDoc.

[2]With the exception of Spanish (9 458 documents) and Russian (5 216 documents) for which not enough data is available.

Language	Category ECAT	CCAT	GCAT	MCAT
English	18.6%	1.5%	33.0%	46.6%
German	11.9%	0.6%	40.6%	46.8%
French	6.0%	21.4%	60.8%	12.8%
Spanish	9.2%	14.8%	9.1%	66.8%

Table 2: Class distribution of the test set of the RCV2 subsets as used in previous publications on cross-lingual document classification.

3. Baseline results

In this section, we provide comparative results on our new Multilingual Document Classification Corpus. Since the initial work by (Klementiev et al., 2012) many alternative approaches to cross-lingual document classification have been developed. We will encourage the respective authors to evaluate their systems on MLDoc. We believe that a large variety of transfer language pairs will give valuable insights on the performance of the various approaches.

In this paper, we propose initial strong baselines which represent two complementary directions of research: one based on the aggregation of multilingual word embeddings, and another one, which directly learns multilingual sentence representations. Details on each approach are given in section 3.1. and 3.2. respectively. In contrast to previous works on cross-lingual document classification with RVC2, we explore training the classifier on all languages and transfer it to all others, ie. we do not limit our study to the transfer between English and a foreign language.

One can envision several ways to define cross-lingual document classification, in function of the resources which are used in the source and transfer language (see Table 3). The first scheme assumes that we have no resources in the transfer language at all, neither labeled nor unlabeled. We will name this case **"zero-shot cross-lingual document classification"**. To simplify the presentation, we will assume that we transfer from English to German. The training and

Transfer type	Model training	Model selection	Evaluation
Zero short	Train L_1	Dev L_1	Test L_i
Targeted	Train L_1	Dev L_2	Test L_2
Joint	Train L_i	Dev L_i	Test L_i

Table 3: Different schemes of cross- and multilingual document classification.

Training Language	Develop Accuracy	Accuracy on test languages								Average	
		DE_1	$EN_{1,2}$	$ES_{1,2}$	$FR_{1,2}$	IT_1	RU_2	ZH_2	JA	$lang_1$	$lang_2$
MultiCCA word embeddings, aggregation by convolutional network											
German	92.2	(93.7)	55.95	*73.23*	71.55	*63.98*	44.83	55.45	60.18	71.68	60.20
English	93.9	**81.2**	(92.2)	*72.50*	72.38	**69.38**	*60.80*	**74.73**	**67.63**	**77.52**	**73.44**
Spanish	95.3	55.8	74.0	(94.45)	65.63	58.35	45.53	41.63	43.40	69.63	67.58
French	91.5	53.7	64.8	65.40	(92.05)	61.15	40.75	38.35	37.75	67.43	64.83
Italian	85.6	49.2	53.7	58.68	62.25	(85.55)	35.58	32.13	45.30	61.87	64.83
Russian	86.8	40.3	72.5	41.03	44.60	42.70	(85.65)	42.38	39.68	48.22	57.30
Chinese	90.8	48.7	56.0	35.53	53.58	47.18	40.45	(87.30)	50.63	48.19	46.55
Japanese	87.3	52.7	54.9	54.28	48.30	44.33	40.85	44.78	(85.35)	50.89	48.52
Joint sentence embeddings BiLSTM + max pooling, trained on Europarl											
German	94.3	(92.03)	71.52	**75.50**	**75.45**	56.45	-	-	-	74.15	-
English	90.7	71.83	(88.40)	66.65	72.83	60.73	-	-	-	72.09	-
Spanish	88.2	71.05	62.70	(88.28)	62.67	57.93	-	-	-	68.53	-
French	90.6	*78.42*	**76.00**	70.70	(89.75)	63.70	-	-	-	*75.71*	-
Italian	83.1	66.22	67.15	67.07	65.07	(82.88)	-	-	-	69.68	-
Joint sentence embeddings BiLSTM + max pooling, trained on United Nations											
English	91.3	-	(88.83)	69.50	*74.52*	-	**61.42**	*71.97*	-	-	**73.25**
Spanish	86.8	-	61.65	(87.67)	61.62	-	45.10	59.88	-	-	63.18
French	90.5	-	*75.35*	71.80	(89.55)	-	59.55	69.08	-	-	73.07
Russian	83.8	-	68.53	65.18	65.90	-	(81.60)	59.65	-	-	68.17
Chinese	90.4	-	66.30	64.78	63.82	-	54.57	87.10	-	-	67.31

Table 4: Baseline classification accuracies for **zero-shot transfer** on the test set of the proposed Multilingual Document Classification Corpus. All classifiers were trained on 1 000 news stories and model selection is performed on the Dev corpus of the training language. The same system is then applied to all test languages. Underlined scores indicate the best result on each transfer language for each group, bold scores the overall best accuracy, and italic ones the second best results.

evaluation protocol is as follows. First, train a classifier using resources in the source language only, eg. the training and development corpus are in English. All meta parameters and model choices are performed using the English development corpus. Once the best performing model is selected, it is applied to the transfer language, eg. the German test set. Since no resources of the transfer language are used, the same system can be applied to many different transfer languages. This type of cross-lingual document classification needs a very strong multilingual representation since no knowledge on the target language was used during the development of the classifier.

In a second class of cross-lingual document classification, we may aim in improving the transfer performance by using a limited amount of resources in the target language. In the framework of the proposed MLDoc we will use the development corpus of target language for model selection. We will name this method **"targeted cross-lingual document classification"** since the system is tailored to one particular transfer language. It is unlikely that this system will perform well on other languages than the ones used for training or model selection.

If the goal is to build one document classification system for many languages, it may be interesting to use already several languages during training and model selection. To allow a fair comparison, we will assume that these multilingual resources have the same size than the ones used for zero-shot or targeted cross-language document classification, e.g. a training set composed of five languages with 200 examples each. This type of training is not a cross-lingual approach any more. Consequently, we will refer to this method as **"joint multilingual document classification"**.

3.1. Multilingual word representations

Several works have been proposed to learn multilingual word embeddings, which are then combined to perform cross-lingual document classifications. These word embeddings are trained on either word alignments or sentence-aligned parallel corpora. To provide reproducible benchmark results, we use MultiCCA word embeddings published by (Ammar et al., 2016).

There are multiple ways to combine these word embeddings for classification. We train a simple one-layer convolutional neural network (CNN) on top of the word embeddings, which has shown to perform well on text classification tasks regardless of training data size (Kim, 2014). Specifically, convolutional filters are applied to windows of word embeddings, with a max-over-time pooling on top of them. We freeze the multilingual word embeddings while only training the classifier. Hyper-parameters such as convolutional output dimension, window sizes are done by grid search over the Dev set of the same language as the train set.

3.2. Multilingual sentence representations

A second direction of research is to directly learn multilingual sentence representations. In this paper, we evaluate a recently proposed technique to learn joint multilingual sentence representations (Schwenk and Douze, 2017). The underlying idea is to use multiple sequence encoders and decoders and to train them with aligned corpora from

Train	Accuracy on test languages					Avg
	DE	EN	ES	FR	IT	
Joint sentence embeddings (Europarl)						
DE	(92.03)	76.48	76.95	76.72	66.27	77.69
EN	81.17	(88.40)	70.75	77.80	62.35	76.09
ES	77.38	67.58	(88.28)	67.92	64.07	73.05
FR	82.78	76.72	76.97	(89.75)	64.07	78.06
IT	77.10	72.70	72.60	76.97	(82.88)	76.45

Table 5: Baseline classification accuracies for **targeted transfer** on the test set of the proposed MLDoc. All classifiers were trained on 1 000 news stories and model selection is performed on the Dev corpus of the target language. Each entry corresponds to a specifically optimized system.

the machine translation community. The goal is that all encoders share the same sentence representation, i.e. we map all languages into one common space. A detailed description of this approach can be found in (Schwenk and Douze, 2017). We have developed two versions of the system: one trained on the Europarl corpus (Koehn, 2005) to cover the languages English, German, French, Spanish and Italian, and another one trained on the United Nations corpus (Ziemski et al., 2016) which allows to learn a joint sentence embedding for English, French, Spanish, Russian and Chinese. We use a one hidden-layer MLP as classifier. For comparison, we have evaluated its performance on the original subset of RCV2 as used in previous publications on cross-lingual document classification: we are able to outperform the current state-of-the-art in three out of six transfer directions.

3.3. Zero-short cross-lingual document classification

The classification accuracy for **zero-shot transfer** on the test set of our Multilingual Document Classification Corpus are summarized in Table 4. The classifiers based on the MultiCCA embeddings perform very well on the development corpus (accuracies close or exceeding 90%). The system trained on English also achieves excellent results when transfered to a different languages, it scores best for three out of seven languages (DE, IT and ZH).[3] However, the transfer accuracies are quite low when training the classifiers on other languages than English, in particular for Russian, Chinese and Japanese.

The systems using multilingual sentence embeddings seem to be overall more robust and less language specific. They score best for four out of seven languages (EN, ES, FR and RU). Training on German or French actually leads to better transfer performance than training on English. Cross-lingual transfer between very different languages like Chinese and Russian also achieves remarkable results.

3.4. Targeted cross-lingual document classification

The classification accuracy for **targeted transfer** are summarized in Table 5. Due to space constraints, we provide only the results for multilingual sentence embeddings and five target languages. Not surprisingly, targeting the classifier to the transfer language can lead to important improvements, in particular when training on Italian.

[3]We exclude Japanese from the comparison since we do not have joint sentence embeddings for that language yet.

Train Size	Accuracy on test languages					Average
	DE	EN	ES	FR	IT	
MultiCCA word embeddings						
1k	91.23	79.08	86.95	81.70	77.58	83.31
Joint sentence embeddings (Europarl)						
1k	88.02	82.42	80.12	84.55	75.08	82.04

Table 6: Baseline classification accuracies on the test set of the proposed MLDoc for **joint multilingual training**. Train and test sets are composed of 200 examples form each of the five languages.

3.5. Joint multilingual document classification

The classification accuracies for **joint multilingual training** are given in Table 6. We use a multilingual train and Dev corpus composed of 200 examples of each of the five languages. One could argue that the data collection and annotation cost for such a corpus would be the same than producing a corpus of the same size in one language only. This leads to important improvement for all languages, in comparison to zero-shot or targeted transfer learning.

4. Conclusion

We have defined a new evaluation framework for cross-lingual document classification in eight languages. This corpus largely extends previous corpora which were also based on the Reuters Corpus Volume 2, but mainly considered the transfer between English and German. We also provide detailed baseline results using two competitive approaches (multilingual word and sentence embeddings, respectively), for cross-lingual document classification between all eight languages. This new evaluation framework is freely available at `https://github.com/facebookresearch/MLDoc`.

5. Bibliographical References

Ammar, W., Mulcaire, G., Tsvetkov, Y., Lample, G., Dyer, C., and Smith, N. A. (2016). Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Klementiev, A., Titov, I., and Bhattarai, B. (2012). Inducing crosslingual distributed representations of words. In *Coling*.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.

Lewis, D. D., yang, Y., Rose, T. G., and Li, F. (2004). Rcv1: A new benchmark collection for text categorization research. *JMLR*, 5:361–397.

Mogadala, A. and Rettinger, A. (2016). Bilingual word embeddings from parallel and non-parallel corpora for cross-language classification. In *NAACL*, pages 692–702.

Schwenk, H. and Douze, M. (2017). Learning joint multilingual sentence representations with neural machine translation. In *ACL workshop on Representation Learning for NLP*.

Ziemski, M., Juncys-Dowmunt, M., and Pouliquen, B. (2016). The United Nations parallel corpus v1.0. In *LREC, Portorož*.

Analyzing Citation-Distance Networks for Evaluating Publication Impact

Drahomira Herrmannova, Petr Knoth, Robert Patton
Oak Ridge National Laboratory, The Open University, Oak Ridge National Laboratory
Oak Ridge, TN, USA; Milton Keynes, UK; Oak Ridge, TN, USA
herrmannovad@ornl.gov, petr.knoth@open.ac.uk, pattonrm@ornl.gov

Abstract

Studying citation patterns of scholarly articles has been of interest to many researchers from various disciplines. While the relationship of citations and scientific impact has been widely studied in the literature, in this paper we develop the idea of analyzing the semantic distance of scholarly articles in a citation network (citation-distance network) to uncover patterns that reflect scientific impact. More specifically, we compare two types of publications in terms of their citation-distance patterns, seminal publications and literature reviews, and focus on their referencing patterns as well as on publications which cite them. We show that seminal publications are associated with a larger semantic distance, measured using the content of the articles, between their references and the citing publications, while literature reviews tend to cite publications from a wider range of topics. Our motivation is to understand and utilize this information to create new research evaluation metrics which would better reflect scientific impact.

Keywords: citation networks, publication impact, semantic similarity

Acknowledgments

This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan[1].

1. Introduction

With the enormous and ever-growing number of research articles being published every year (Jinha, 2010; Laakso and Björk, 2012), researchers more and more often resort to using research evaluation metrics such as journal and publication citation indexes, as a proxy to quality and importance. Research metrics have been applied in various scenarios, from search and recommendation (Carevic and Schaer, 2014; Belter, 2016), to grant and tenure awards (Meho, 2007). Due to many drawbacks and limitations of the purely citation-based methods (Seglen, 1992; Seglen, 1997; Priem et al., 2010), recent years have seen the emergence of many new approaches and alternatives to the traditional *bibliometrics*, most notably metrics often referred to collectively as *altmetrics* (Priem et al., 2010) and *webometrics* (Almind and Ingwersen, 1997), which rely on data collected from the Web, such as download counts and social and news media mentions, and *semantometrics* (Knoth and Herrmannova, 2014), which measure how far each discovery takes us by utilizing publication full texts.

While citation networks have been widely studied in the literature, a number of works have recently developed the idea of studying citation patterns in combination with content similarity. In this paper we further explore this area and study how these citation-distance patterns reflect scientific impact. Our motivation is to explore whether certain patterns could be utilized in research evaluation to create new research evaluation metrics which reflect scientific impact more accurately than the widely used citation counts. We argue that measuring just the number of interactions in the scholarly communication network does not provide enough information for a sufficient understanding of the contributions a publication had, and posit that publication manuscript, in addition to the number of interactions, is needed to asses the value of a publication. For example, literature review publications are known to be highly cited, yet their main aim is to educate rather than influence a research area the way seminal publications do.

To this end we study the differences between citation-distance patterns of seminal publications and literature reviews. In this sense, these two types of papers represent extreme cases, as literature reviews, by definition, do not provide new ideas, while seminal publications greatly influence later developments. We believe research evaluation metrics, especially those focused on research impact, should be able to distinguish between these publication types. Our motivation is to understand and utilize information about the citation-distance patterns of these publications to create new research evaluation metrics which would better reflect scientific impact.

2. Related Work

A number of researchers have recently explored the idea of studying citation patterns in terms of content similarity, and utilizing these patterns for various tasks related to research evaluation. (Gerrish and Blei, 2010) have used a dynamic topic model to model thematic changes of content of documents, which was then used to create a Document Influence Model for measuring the importance of individual documents within a collection. (Yan et al., 2012) have used similarity between a publication and its references, which was calculated using Kullback-Leibler divergence of the content, to assess novelty. (Knoth and Herrmannova, 2014) have used semantic distance between publications

[1] http://energy.gov/downloads/doe-public-access-plan

which have cited a given publication and the publications cited by the publication to assess research contribution. In their case, distance was calculated using cosine similarity between $tf-idf$ term-document vectors. (Whalen et al., 2015) have used distance between a publication and publications that cite it, which was also calculated as cosine similarity, to predict future citations. In this paper we further explore the idea of studying citation patterns in terms of content similarity. To do this, we use the recently released TrueImpactDataset (Herrmannova et al., 2017) which contains publications of two types, seminal publications and literature reviews, and compare the citation patters of these two types of publications in terms of content distance.

3. Methodology

Our greatest interest lies in understanding how can we take into account publication content to improve automated research evaluation. We investigate citation networks in terms of content distance and study how the uncovered patterns can be used for identifying highly influential publications. Specifically, we investigate the relations in a citation network studied in (Yan et al., 2012), (Knoth and Herrmannova, 2014) and (Whalen et al., 2015), which are depicted in Figure 1.

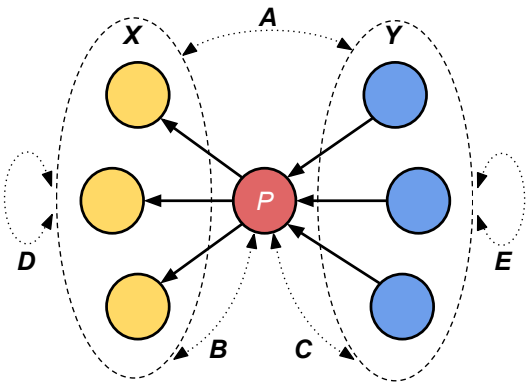


Figure 1: Neighborhood of a single publication P and relations between publications in the neighborhood which we investigate.

In the figure, node P represents the publication of interest, the yellow nodes (set X) represent publications cited by P (its references), and the blue nodes (set Y) represent publications that cited P. The relations between these publications, which are studied in this paper, are labeled A, B and C in the figure. In addition to the relations mentioned above, we also study the relations between the cited papers (regardless of whether there is a direct citation edge between them, these relations are labeled D), and between the citing papers (E).

The idea of studying the citing and cited publications is based on the process of how research builds on the existing knowledge in order to create new knowledge. In citation networks, the nodes, which are scholarly publications, are connected by citation relations which represent knowledge flows between the publications. While the cited papers are representative of the state-of-the-art in the domain of the publication in question (the publication itself contains only a fraction of the knowledge on which it is built, while the cited publications represent this knowledge more completely), the citing publications represent areas of application of the publication in question. Incorporating content into the analysis of citation behavior enables us to distinguish articles which draw on or influence diverse research areas from those which work within one specific discipline. Furthermore, the assumption, and a hypothesis made by both (Knoth and Herrmannova, 2014) and (Whalen et al., 2015), is that useful innovation will propagate in the form of new knowledge to the citing publications leading to a higher distance between the cited and citing publications.

To measure the distance we use the cosine similarity measure on $tf-idf$ term-document vectors created from the publications' abstracts. We then calculate the distance of two publications as $dist(p_1, p_2) = 1 - sim(p_1, p_2)$, where $sim(p_1, p_2)$ is the cosine similarity between the $tf-idf$ term vectors. Each set of relations A-E described above is represented as a set of distances (for example a set of distances between a publication and each of its references). We define a set of metrics applied on the distributions induced by the distances. An example of what characteristics we aim to distinguish is whether literature review publications typically cite a wider range of topics than seminal publications and whether seminal publications tend to work within a narrower area. The metrics we use to describe the distance distributions are: (1) minimum, (2) maximum, (3) range (difference between maximum and minimum), (4) sum of the distances, (5) mean distance, (6) standard deviation, (7) variance of the distances, (8) 25th percentile, (9) 50th percentile (median), (10) 75th percentile, (11) skewness, which is a measure of the asymmetry of the distribution, negative skew means the left tail is longer, positive skew means the right tail is longer, and (12) kurtosis, which is a measure of whether the data are heavy- or light-tailed, higher value means sharper peak. Because we describe each of the 5 distance distributions with 12 metrics, we have 60 features (features F1-F60) describing a publication's neighborhood.

4. Data

To collect all data needed for studying the relations introduced in the previous section, we have used three data sources:

1. TrueImpactDataset[2] (Herrmannova et. al., 2017) (Herrmannova et al., 2017), which provides us with seminal publications and literature reviews (i.e. the P node in Figure 1),
2. Microsoft Academic (MA) API[3] (Sinha et al., 2015) which we use to collect metadata (authors, year, venue, DOI, etc.) of the citing and cited publications (blue and yellow nodes in Figure 1),
3. Mendeley API[4] which we use to collect abstracts (since MA does not contain abstracts).

[2] http://trueimpactdataset.semantometrics.org/
[3] http://aka.ms/academicgraph/
[4] http://dev.mendeley.com/

Table 1 shows the size of the dataset. After collecting all needed data the size of the dataset was reduced to 276 publications (i.e. publications with at least one reference or at least one citation) – 126 literature reviews and 150 seminal publications.

Publications in TrueImpactDataset	314
TrueImpactDataset publications in MA	298
Pubs with at least one citation in MA	269
Pubs with at least one reference in MA	215
At least one cit. and one ref. in MA	209
Total number of citing papers	154,056
Total number of references	13,599

Table 1: Dataset size. The table shows for how many of the TrueImpactDataset publications we managed to get the needed metadata and how many additional publications we collected.

5. Experiments and Results

We begin by comparing the properties of seminal publications and literature reviews. We investigate how these two types of papers are situated with regard to the extracted features. To understand which features might assist with the task we calculate an independent one-tailed t-test for each feature. The t-test is a measure commonly used to assess whether two sets of data are statistically different from each other. In other words, it helps to determine the features that can distinguish literature reviews from seminal papers. To test the significance, we set the significance threshold at 0.05. Out of the 60 features, 27 result in p-value higher than 0.05. In this case we accept the null hypothesis of equal means. As the t-test tells us the values of these features are not significantly different for the two sets of papers, we remove these features from further analysis. The removed features are crossed out in Table 2.

From Table 2 it is obvious that there is not a single type of feature which describes well all five distributions. Furthermore, as most of the features describing the distribution E (distances between papers citing a publication) were removed, it seems this distribution does not offer much information for this task.

Next, we create a histogram for each feature and by comparing these histograms for the two publication types we gain insight into norms and placement of seminal publications and literature reviews in terms of their citation patterns. Figure 2 shows histograms of the remaining features, with seminal publications and literature reviews distinguished by color. In all of the histograms literature reviews are represented with dashed lines with circle points, while seminal publications with full lines with square points. The numbers in the legend of each plot show how many publications were used to produce each histogram (the numbers differ in case our data was incomplete and we could not calculate the given feature for all publications). To preserve space we do not show here histograms of all of the remaining features F1-F60, but instead we select 15 features with interesting properties.

	A (C-R)	**B (P-R)**	**C (C-P)**	**D (R-R)**	**E (C-C)**
min	~~F1~~	**F13**	~~F25~~	**F37**	**F49**
max	~~F2~~	**F14**	~~F26~~	~~F38~~	~~F50~~
range	~~F3~~	**F15**	~~F27~~	**F39**	**F51**
sum	~~F4~~	**F16**	**F28**	**F40**	**F52**
mean	**F5**	**F17**	**F29**	**F41**	~~F53~~
std	**F6**	**F18**	**F30**	**F42**	~~F54~~
variance	**F7**	**F19**	**F31**	**F43**	~~F55~~
p25	**F8**	**F20**	**F32**	**F44**	~~F56~~
p50	~~F9~~	**F21**	~~F33~~	~~F45~~	~~F57~~
p75	~~F10~~	~~F22~~	~~F34~~	~~F46~~	~~F58~~
skewness	**F11**	**F23**	~~F35~~	**F47**	~~F59~~
kurtosis	~~F12~~	~~F24~~	**F36**	**F48**	~~F60~~

Table 2: The columns in the table represent the five distance distributions studied here, the rows represent the 12 metrics used to describe each of the distance distributions, and the cells represent individual features. The second row of the header provides an explanation for which distance each column represents, e.g. *C-R* means the distance between citing papers and references, *P-R* means the distance between the publication and its references, *C-P* means the distance between the citing papers and the publication in question, and so forth. The crossed-out features are those which we removed from further analysis.

In general, various metrics seem quite consistent across both groups. However, these metrics also reveal some important differences in citation patterns of seminal publications and literature reviews. First, one of our expectations and a hypothesis made by both (Knoth and Herrmannova, 2014) and (Whalen et al., 2015) is that useful innovation introduced by a publication will propagate in the form of new knowledge to the citing publications, leading to a higher distance between the publication and the citing publications (distance C) as well as between the references and citing publications (distance A). This is confirmed by higher average distances of both distributions in case of seminal publications (features F5 and F29). This is further supported by a lower standard deviation of the A and C distance distributions for seminal papers (features F6 and F30).

Secondly, the distribution of distances between a publication and its references seems consistent with our expectations. In the case of literature reviews, the minimal distance between the publication and its references is on average smaller than for seminal papers (F13). At the same time, the difference between the most similar and most dissimilar reference is higher for literature reviews (F15). Furthermore the sum of distances between the publication and its references is higher for literature reviews than for seminal papers (F16), which is likely because reference lists of literature reviews are typically long.

5.1. Citation Patterns and Publication Importance

In this section we explore the relation between the perceived impact of publications and the different metrics used to measure it. Although the above analysis of the separate

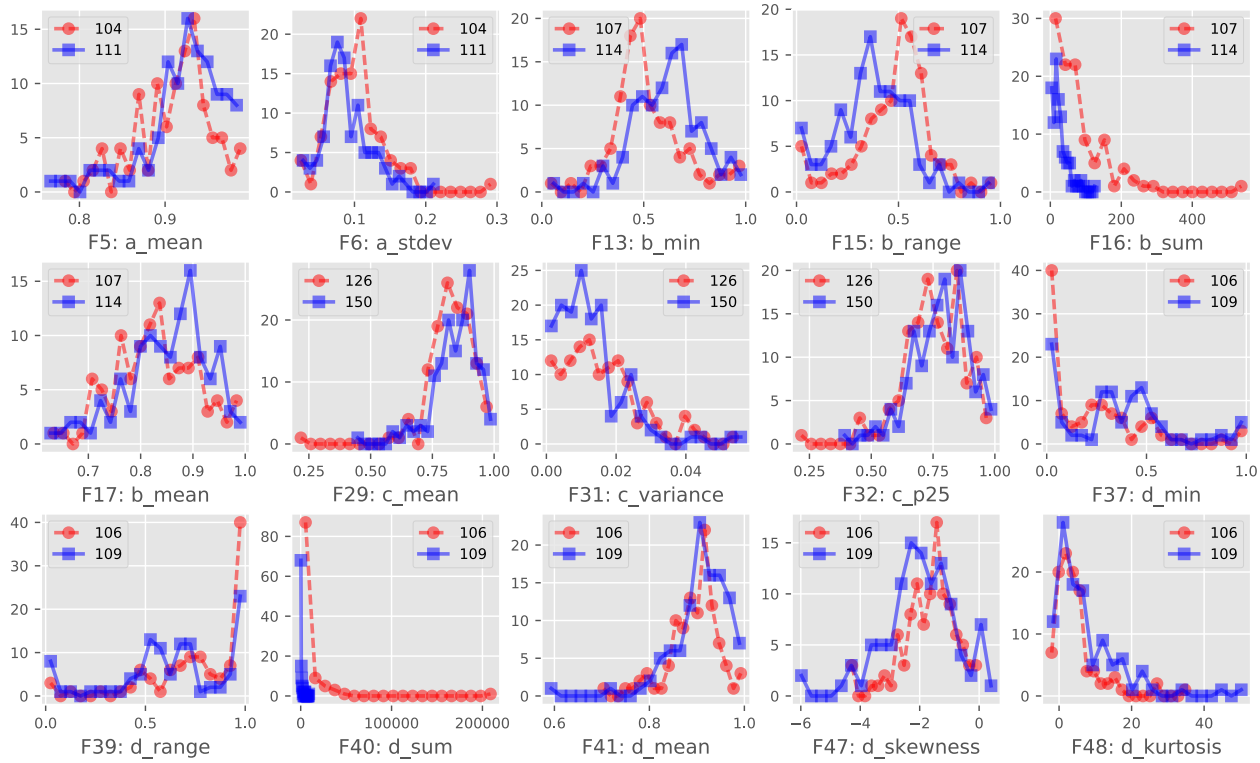


Figure 2: Histograms of selected features describing distance distributions A-E from Figure 1. In the figures, literature reviews are represented with dashed red lines with circle points and seminal publications with full blue lines with square points. The numbers in the legend of each subplot show how many publications were used to plot each line (the numbers differ across subplots in case we didn't have all data needed to calculate a given feature). Each subplot then shows how literature reviews and seminal publications are positioned with respect to that feature.

features revealed distinct differences between the citation behavior of seminal and literature reviews, we are interested in analyzing whether the revealed patterns help in distinguishing important seminal publications from literature reviews better than current research evaluation methods. To be able to compare features in terms of accuracy we approach this question as a classification task.

After testing different classifiers (specifically SVM, logistic regression, decision tree and Naïve Bayes), we have selected Naïve Bayes (NB) classifier as a classifier which works well on our dataset and our task across different setups. In our classification experiment we use a leave-one-out cross-validation setup, that is we repeatedly train on all but one publication and then test the performance of the model on the publication we left out of the training. The performance is evaluated using accuracy, considering seminal papers as the positive class. We compare the results against a baseline which always predicts the most frequent label (a seminal publication).

To understand the contribution of each feature, we trained and tested the NB classifier using a single feature at a time and calculated accuracy using each feature. This is again done in a leave-one-out cross-validation setup. For comparison we also train a classifier using citation counts as a single feature (specifically citation counts of each seminal and literature review publication obtained from MA). We are interested in analyzing whether some content-based features distinguish between these two types of publications better than citation counts. Table 3 shows results for top 10 features according to classification accuracy, as well as for citation counts, which were the 22. best feature out of the 27 features used in this experiment. All 27 features achieve better performance than the baseline.

#	Feature	Accuracy
1	F16: B sum	0.6897
2	F15: B range	0.6502
3	F13: B min	0.6453
4	F40: D sum	0.6355
5	F37: D min	0.6059
6	F31: C variance	0.6010
7	F39: D range	0.5911
8	F47: D skewness	0.5911
9	F32: C p25	0.5911
10	F48: D kurtosis	0.5813
22	Citations	0.5616
	Baseline	0.5025

Table 3: Classification performance when using individual features. The features are listed in descending order of accuracy.

The classification accuracy using the best performing feature F16 is $\sim$69%, while our baseline classifier achieves the accuracy of $\sim$50%. This means that by using F16 alone, it

is possible to achieve 11% improvement over the widely used citation counts on this task. While in this study we only trained the classifier using a single feature at a time, in the future we will evaluate the performance of classifiers trained using a variety of well performing feature combinations.

It can be seen there are a number of features which work particularly well in distinguishing these two types of publications. These are particularly features describing the distance distributions B, C and D. The three best performing features are all related to the distance between a publication and its references. In particular, this experiment confirmed the features describing the distance between a publication and its references distinguish between the two types of papers (F13, F15, F16). Similarly, features describing the distribution of distances between a publication's references work well in this task, particularly features describing the "width" of the distribution and the shape of its peak (skew and kurtosis). One particularly interesting feature which outperforms simple citation counts by a significant margin is feature F31, which describes variance of the distance distribution C. This is interesting as it shows citations to literature reviews tend to come from broader mix of more and less distant citing publications.

6. Conclusion and Future Work

This paper studied the relationship between semantic distance of scholarly articles in a citation network and their impact. More specifically, following on the work of (Knoth and Herrmannova, 2014) and (Whalen et al., 2015) we investigated the novelty assumption, i.e. the idea that new useful ideas tend to propagate to the work of others, which in turn influences the semantic distance patterns in a citation network. To validate this assumption, we have used the new TrueImpactDataset (Herrmannova et al., 2017) to systematically evaluate a range of distance features characterizing the relationship between seminal and review publications, their references and citing publications. Our results show that there a number of features describing citation-distance patterns which significantly outperform widely used citation counts in distinguishing seminal publications from literature reviews on our dataset. This demonstrates content analysis might provide valuable information for research evaluation. While in this study we have focused on individual features, as future work, we are planning to evaluate the performance of a variety of well performing feature combinations. We also plan on experimenting with other semantic similarity measures, such as similarity computed on word2vec vectors, as well as investigating the effect of using abstracts compared to fulltext.

7. Bibliographical References

Almind, T. C. and Ingwersen, P. (1997). Informetric Analyses on the World Wide Web: Methodological Approaches to 'Webometrics'. *Journal of Documentation*, 53(4):404–426.

Belter, C. W. (2016). Citation Analysis as a Literature Search Method for Systematic Reviews. *Journal of the Association for Information Science and Technology*, 67(11):2766–2777.

Carevic, Z. and Schaer, P. (2014). On the Connection Between Citation-based and Topical Relevance Ranking: Results of a Pretest using iSearch. In *BIR@ ECIR*, pages 37–44.

Gerrish, S. and Blei, D. M. (2010). A Language-Based Approach to Measuring Scholarly Impact. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 375–382.

Herrmannova, D., Patton, R. M., Knoth, P., and Stahl, C. G. (2017). Citations and Readership are Poor Indicators of Research Excellence: Introducing TrueImpactDataset, a New Dataset for Validating Research Evaluation Metrics. In *Proceedings of the 1st Workshop on Scholarly Web Mining*, pages 41–48. ACM.

Jinha, A. E. (2010). Article 50 Million: An Estimate of the Number of Scholarly Articles in Existence. *Learned Publishing*, 23(3):258–263.

Knoth, P. and Herrmannova, D. (2014). Towards Semantometrics: A New Semantic Similarity Based Measure for Assessing a Research Publication's Contribution. *D-Lib Magazine*, 20(11/12).

Laakso, M. and Björk, B.-C. (2012). Anatomy of Open Access Publishing: A Study of Longitudinal Development and Internal Structure. *BMC medicine*, 10(1):1. DOI: 10.1186/1741-7015-10-124.

Meho, L. I. (2007). The Rise and Rise of Citation Analysis. *Physics World*, 20(1):32.

Priem, J., Taraborelli, D., Groth, P., and Neylon, C. (2010). Altmetrics: A Manifesto.

Seglen, P. O. (1992). The Skewness of Science. *Journal of the American Society for Information Science*, 43(9):628.

Seglen, P. O. (1997). Why the Impact Factor of Journals Should Not be Used for Evaluating Research. *BMJ: British Medical Journal*, 314(7079):498.

Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-j. P., and Wang, K. (2015). An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web*, pages 243–246. ACM.

Whalen, R., Huang, Y., Sawant, A., Uzzi, B., and Contractor, N. (2015). Natural Language Processing, Article Content & Bibliometrics: Predicting High Impact Science. *Quantifying and Analyzing Scholarly Communication on the Web (ASCW'15)*.

Yan, R., Huang, C., Tang, J., Zhang, Y., and Li, X. (2012). To Better Stand on the Shoulder of Giants. In *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 51–60. ACM.

8. Language Resource References

Herrmannova et. al. (2017). *TrueImpactDataset*. Distributed via `http://trueimpactdataset.semantometrics.org/`, 1.0, ISLRN 197-407-228-291-9.

Annotating Educational Questions for Student Response Analysis

Andreea Godea and Rodney Nielsen
University of North Texas
1155 Union Circle, Denton, TX, USA
AndreeaGodea@my.unt.edu
Rodney.Nielsen@unt.edu

Abstract

Questions play an important role in the educational domain, representing the main form of interaction between instructors and students. In this paper, we introduce the first taxonomy and annotated educational corpus of questions that aims to help with the analysis of student responses. The dataset can be employed in approaches that classify questions based on the expected answer types. This can be an important component in applications that require prior knowledge about the desired answer to a given question, such as educational and question answering systems. To demonstrate the applicability and the effectiveness of the data within approaches to classify questions based on expected answer types, we performed extensive experiments on our dataset using a neural network with word embeddings as features. The approach achieved a weighted F_1-score of 0.511, overcoming the baseline by 12%. This demonstrates that our corpus can be effectively integrated in simple approaches that classify questions based on the response type.

Keywords: question dataset, taxonomy, expected answer types, educational environment

1. Introduction

Questions represent natural language sentences that express the information need of the inquirer. The analysis of questions is an important part in educational systems, since questioning is the main form of interaction between instructors and students. In this domain, the automatic classification of questions has multiple potential applications. For instance, it can help in the assessment process, in developing effective teaching strategies or in the analysis of questions asked by the students. Question datasets and taxonomies play a very important part in any automatic approach, being used as a base to identify patterns in annotated data that will be applied further to unseen examples. In the educational field, the existence of approaches using such data will provide important information to instructors regarding their interaction with students and will allow them to adapt their teaching strategies to the classroom's needs.

Various question taxonomies and classification approaches for educational systems have been proposed to help in the analysis of data from this domain. However, the majority of question datasets and taxonomies that consider the expected answer type were designed for question answering (QA) and less for educational systems. This is because question classification is a very important component in QA systems, since the main goal of such systems is to identify the best possible answer among a collection of candidate answers, given a question asked by the user. On the other hand, in the educational domain, the questions have multiple potential applications and can be analyzed from different perspectives. Based on the objective being sought, researchers focused their attention on classifying questions based on their subject (Conner, 1927), the educational objective (Bloom, 1956), the difficulty level (İnce, 2008) or the question goal (Lehnert, 1977). However, we found that this domain lacks datasets and taxonomies that aim to analyze questions with respect to *expected answer types*.

In this paper, we introduce the first taxonomy and annotated corpus that aims to analyze educational questions based on expected answer types. More specific, the taxonomy characterizes the questions based on the type of information that is expected to appear in correct answers. Our analysis of questions provides valuable information regarding the expected answer, which will help in identifying if *the correct answer is expected to provide the solution to a given problem, an equation or a drawing, or if it is expected to be a short or a constructed response*, among others.

The dataset can be integrated as an important component in various systems. For example, it can be employed within question answering systems to automatically identify the types of answers elicited by users' questions. This can help in narrowing the space of candidate answers and ensure more accurate recommendations to users. In addition to this, the corpus can be leveraged in educational systems that aim to analyze questions based on the types of the expected answers. The task has multiple potential applications, such as facilitating the assessment process by comparing student responses with the expected answers or identifying which concepts were understood, misunderstood or omitted by the student. This information can help teachers to draw important conclusions regarding the students' conceptual understanding, and allow them to develop teaching strategies based on the students' needs.

Finally, if incorporated in an educational system, the dataset we introduce can also be used to automatically generate questions depending on the type of answers the instructor wants to elicit. For example, the teacher can choose to ask short/direct questions or elicit constructed responses, drawings, equations or it can ask the student to provide the solution to a given problem.

The main contributions of this paper are: (1) We present the first question taxonomy based on expected answer types for educational applications, comprising 16 categories, (2) We collect a dataset of questions from real middle school science classrooms and construct thorough anno-

tation guidelines based on the analysis of questions, which will both be released for research purposes and (3) We provide evidence that our dataset can be effectively integrated in supervised approaches within educational systems.

In the remainder of this paper, we present an overview of related work, describe our dataset and present additional evidence that our data can be effectively utilized by supervised approaches to determine what to expect in student responses.

2. Related Work

Over time, researchers have proposed various datasets and taxonomies for question classification based on their interests. One widely used way to classify questions is based on the expected answer types. Although multiple applications can benefit from analyzing questions based on this criterion, the majority of datasets and taxonomies were designed for question answering systems. However, this type of classification can have multiple potential applications in educational systems as well, from facilitating student assessment to identifying the students' knowledge gaps in order to initiate classrooms discussions. Since educational systems currently lack such data, we present the first question dataset and taxonomy based on expected answer types which can help with the analysis of student responses.

Question classification is an important component in QA systems, which use question datasets and taxonomies to learn patterns in question-answer pairs. In the QA area, the most well known datasets were developed within the Text Retrieval Conference (TREC) QA tracks, which published large amounts of data each year to support competitive research toward developing systems capable of answering open-domain, closed-class questions. Starting with TREC-8 (Voorhees and Tice, 1999), new subsets of questions were included each year, extracted from different sources (e.g., Encarta, Excite, MSNSearch, AskJeeves logs).

Subsets of the TREC datasets have been used by various researchers in their approaches. For example, Hovy et al. (2001) proposed the USC dataset, containing question-answer pairs from TREC-8, TREC-9 and $answers.com$. They created a question taxonomy that reflects the user's intention, such as *veracity (yes-no, true-false), entity (e.g., agent, quantity, location)* and *narative (e.g., history, evaluation, cause-effect)*. Later, Li and Roth (2002) introduced the UIUC dataset based on the USC dataset and TREC-10. They proposed a two layer taxonomy containing 6 coarse classes (abbreviation, entity, description, human, location and numeric value) and 50 fine classes. This hierarchical taxonomy allows the classification of questions at various degrees of granularity and allows more flexibility than the flat one proposed by Hovy et al. (2001).

More recently, Yang et al. (2015) proposed a dataset for open-domain question answering, named WIKIQA. The dataset contains questions collected from Bing query logs and each question is associated with a Wikipedia page assumed to be the topic of the question. The candidate answers for a given question are considered to be all the sentences in the summary paragraph of the Wikipedia page. In contrast with the TREC datasets, WIKIQA is more challenging because it includes questions with no correct answers. The questions were labeled in a similar manner with previous works, based on the following answer types: *location, human, numeric, abbreviation, entity, description.*

In this paper, we introduce the first dataset with questions from the educational domain, annotated with expected answer types. In contrast with previous datasets with questions primarily used for question answering systems, we propose the first dataset that can facilitate the analysis of student responses in the educational environment.

3. The Corpus

The corpus presented in this paper contains questions from the educational environment. Specifically, we collected science questions asked by teachers in real middle school classrooms. The teachers entered their questions on a web-enabled device and presented them to the students in order to initiate discussions and identify potential gaps in the conceptual understanding. After a thorough analysis of the questions in our data, we propose a novel taxonomy containing 16 categories of questions based on expected answer types. Specifically, the taxonomy was created considering previously proposed schemes and the particularities of questions in our data, with the focus on the features that would facilitate the identification of what is expected from a correct student response. Since our questions can contain one or more sentences and each sentence can elicit different information, we created our taxonomy based on each unique question sentence in our data.

In this section, we present our taxonomy, the inter-class correlations, the annotation process and data distribution.

3.1. Question Taxonomy

We propose a taxonomy with 16 question categories based on expected answer types, as follows:

Clarification – the question elicits a response, but only clarifies, details or paraphrases information already requested in preceding sentences. Example: "Describe mammals. *Discuss the brain, dermis, and child rearing.*"

SubjectiveConcept – the question asks for feelings or opinions rather than facts. There is no wrong answer assuming the response is honest and on task and that any background information or supporting claims are accurate. Example: "*Describe your favorite activity or lab.*"

Select1 – the question provides two or more possible answers and only one of them is correct. The option labels must be included in the text of the question. Example: "*Which would you expect to have a higher density, hot water or cold water?*"

SelectN – the question provides a list of possible answers and elicits the selection of all responses that apply. The option labels must be included in the text of the question. Example: "*Which of the following sources of energy are considered clean?* Coal, Solar, Wind, Oil, Gas, Nuclear."

TrueFalse – the question requires a positive (e.g., true, yes) or negative response (e.g., false, no). Example: "*Does energy have anything to do with physical movement?*"

List – the question elicits a list of items. Example: "*Explain at least two differences between longitudinal and transverse waves.*"

Data	Sents	Clarif	SubjC	Sel1	SelN	T/F	List	MultiP	ShrtAns	OthCR	Proc	Eq	Soln	Draw	CntxtS	AnsVry	Order
Train	1085	25	21	24	0	29	145	102	272	633	62	14	21	6	252	207	112
Test	569	17	13	20	1	15	75	52	151	313	38	8	16	4	125	99	58
Total	1654	42	34	44	1	44	220	154	423	946	100	22	37	10	377	306	170

Table 1: Number of Instances per Class.

Multi-Part – the question asks for a number of various items, differentiated in text. Example: "*What is mass and how do we measure it?*"
VeryShortAnswer – the question elicits an extremely short phrase or a single word. Example: "*Where was Helium discovered?*"
OtherConstructedResponse – the question seeks a constructed response that can have up to several sentences. The answer should contain at least a verb phrase. Example: "*Compare and contrast microwaves with gamma waves.*"
ProcessProcedure – the question requests the process by which something happens (e.g., a natural/involuntary process of change) or the procedure for accomplishing a task. Example: "*How did Marie Currie discover radioactivity?*"
Equation – the question elicits an equation/formula. Example: "*Write the equation for calculating an object's speed.*"
Solution – the question asks the student to solve a computational or mathematical problem. Example: "*What is the volume of 103 g of water?*"
Drawing – the question asks for a drawing. Example: "*Sketch the atomic structure for nitrogen and boron using the Bohr Model.*"
ContextSensitive – the question is referring to a picture, video, image, or previously-conducted lab, etc. The answer should refer to material that is not explicitly included in the text of the question and that is not based on the general subject matter the course is teaching. Example: "*Summarize what you learned yesterday by using the simulation.*"
AnswersWillVary – the question has more than one correct answer. Hence, the student responses can vary. Example: "*Think of a chemical reaction you are familiar with and list the reactants and products.*"
Ordered – the question expects a specific sequence within the response. Example: "*Name the planets in order from the closest to the sun to the furthest from the sun.*"

3.2. Inter-class Correlations

From the description of our taxonomy, it can be noticed that only a subset of types are in general mutually exclusive – a sentence cannot be considered both a *VeryShortAnswer* and an *OtherConstructedResponse* unless we deal with a *Multi-Part* question. It can be observed that several question types are ***not*** mutually exclusive. For example, the question "*Name one important lab safety procedure and explain why it is important.*" will be considered *Multi-Part, VeryShortAnswer and OtherConstructedResponse*, since it elicits two different parts – a name (a short answer) and an explanation (a constructed response). In fact, we identified several class pairs that are highly correlated in our data.

- A question annotated as a *SubjectiveConcept* is always also considered an *AnswersWillVary* ("*What interested you most about elements in periodic table?*"), but the reverse does not always apply ("*What scientific practice could you reflect upon for the Atoms?*"). Based on the context, *SubjectiveConcept* questions are often considered *OtherConstructedResponse*, but could be annotated as *VeryShortAnswer*, among others.
- A *List* is very frequently also annotated as *AnswersWillVary* ("*Describe some of the unique qualities of the water molecule.*"), but there are also exceptions ("*Name **the three** parts of an atom.*").
- A *ProcessProcedure* question is very often considered an *OtherConstructedResponse* and *Ordered*, since it implies the description of a sequence of steps.
- A *Solution* is in general assumed to also be a *VeryShortAnswer*, unless it asks the students to show their work (where it is *OtherConstructedResponse*).
- A *Multi-Part* question is expected to also be *Ordered*.

3.3. Data Annotation

The dataset presented in this paper comprises 1155 questions, with 1654 sentences in total. Each question is composed of one or more sentences, with the largest question having 18 sentences (only six of which elicited a response). Since our questions can contain multiple sentences and each sentence can elicit different information, we created the taxonomy based on the question sentences in our data. Hence, in the annotation process, each question was first split into sentences. Then, each sentence was independently annotated with class types by two graduate students and adjudicated by a third. Sentences were tagged with one or more labels, since the categories are not mutually exclusive and each sentence can elicit multiple response types.
An analysis of our data revealed that the maximum number of labels attached to a sentence is 6. This applies to the italicized sentence from the following question: "What was his apparatus of choice? *Draw and label its components.*", which is annotated as: (1) *Multi-Part* (it elicits two different things – a drawing and a list of components), (2) *List* (the answer should contain a list), (3) *VeryShortAnswer* (the list should contain short answers), (4) *Drawing* (the sentence elicits a drawing), (5) *ContextSensitive* (it refers to context sensitive material – "his apparatus of choice") and (6) *AnswersWillVary* (the list length was not specified).
The inter-annotator agreement over each class is Kappa = 0.75, which is a substantial agreement according to Landis and Koch (1977). Further, we analyzed the independent labels of the first two annotators and we observed that most disagreements were in labeling the *ContextSensitive* class. More specifically, 24% of disagreements were for the *ContextSensitive* class, followed by *VeryShortAnswer* and *OtherConstructedResponse* with 18% each. On the other hand, the annotators agreed in all cases when labeling *Drawing*

Clarification	SubjCon	Select1	T/F	List	MultiP	ShrtAns	OthConRrsp	Process	Equation	Solution	CntxtSens	AnsWillVry	Order
0.000	0.273	0.125	0.100	0.483	0.364	0.51	0.75	0.154	0.571	0.522	0.50	0.472	0.174

Table 2: Test Set F_1-score per Class.

and *Ordered* questions. This can be explained by the fact that these classes have clearer patterns in data and can be easily separated from the other class types. However, identifying if a question elicits context sensitive information or a short versus a longer response appears to be more subjective, based on each annotator's interpretation.

3.4. Data Distribution

To assess the applicability of our dataset, we also tested it within a supervised approach. For this purpose, we split the *questions* in our dataset into two separate sets – 66% for *train* and 34% for *test*. We performed the split at the question level instead of sentence level, to ensure that all the information comprised in a sentence is located in a single subset of data - either train or test. This is an important aspect in the classification approach, since sometimes the labels are preserved between sentences within the same question. For instance, if a sentence comprises *ContextSensitive* information, the next sentences within the same question will also have this label if they refer to the same concepts. Similarly, a *Clarification* sentence always refers to previous sentences within the question, since its goal is to detail or paraphrase what was previously elicited. We provide the data distribution based on adjudicated labels in Table 1.

The distribution of question types in the data reveals that the dominant class is *OtherConstructedResponse* with 946 instances, followed by *VeryShortAnswer* with 423 instances. This implies that these instructors generally focused on asking deeper questions in this dataset. This is an important observation regarding our data, since it was shown that involving deep questioning during tutoring can improve knowledge learning (Chi et al., 1994). On the other hand, the least frequent question types in the data collected during tutoring are *SelectN, Drawing* and *Equation*. These question types require direct answers – the selection of all options that apply from the question text, a drawing or an equation.

4. Methodology for Question Classification

We propose a supervised approach in order to validate our data and demonstrate that it can be effectively learned. More specifically, we trained separate artificial neural networks using the one-vs.-all strategy for each class and used pre-trained word embeddings (Pennington et al., 2014) to classify questions based on their expected answer types. To this end, we split the training data into two subsets – 66% for training and 34% for validation, and used the *validation* set to run experiments in order to tune the parameters for each class.

We experimented with various word embeddings dimensions (50, 100 or 200) for each class type and finally set the dimension to 100 for the *ContextSensitive, Equation, List, Multi-Part, ProcessProcedure, Select1* and *SubjectiveConcept* classes and 200 for the remaining classes. The word embeddings corresponding to a question were combined into a single feature vector by computing their normalized sum, as in the following equation:

$$v(Q, C) = || \sum_{w \in Q} glove(w_C)||$$

where $v(Q, C)$ represents the features for a given question-class pairing, w iterates over all the words in the question, and the function $glove(w_C)$ retrieves the Glove word embedding for w with the dimension specific to class C.

As a result of tuning the network's parameters on the validation data, the number of iterations was set to 3000, the learning rate had values between 0.01 and 0.1, and the number of hidden layers was set to 2, with the number of nodes ranging from 3 to 10 in a layer. The results obtained for our classes using these parameters are reported in Table 2. As it can be observed, we did not include the *Drawing* and *SelectN* classes, because they have fewer than 10 examples in the entire dataset (see Table 1 for the distribution).

Our results show that the best performing class is *OtherConstructedResponse* with F_1-score = 0.75, followed by *Equation, Solution, VeryShortAnswer* and *ContextSensitive*, each achieving an F_1-score higher than 0.5. In case of *OtherConstructedResponse* and *VeryShortAnswer* classes, the results can be explained by the large number of examples in the training set, which helped in identifying patterns for these question types. Although *Equation* and *Solution* have less training examples, these classes possess clearer patterns in the data. On the other hand, the worst performing class is *Clarification*, for which the simple classifier (strawman) was not able to capture patterns. This is because the classifier employs only the sentence's context and does not take into account what was requested in previous sentences. A *Clarification* can be easier identified if the information from previous sentences is taken into account, since the goal of this class type is to clarify, detail or paraphrase other sentence.

It can be seen that *TrueFalse* and *Select1* are among the worst performing classes, although they intuitively follow specific patterns. We analyzed our data and found that there are no specific keywords in the question sentences associated with these types of categories. For *TrueFalse*, our questions do not explicitly contain terms such as *true* or *false*, but generally start with auxiliary verbs and require yes/no responses – *"Does energy have anything to do with physical movement?"*. We experimented with adding a new binary feature to the word embeddings' feature vector, which checks if the question starts with an auxiliary verb. The performance reached an F_1-score = 0.65, with an increase of 55% compared with using only word embeddings.

With respect to Select1, we found that the corresponding question sentences do not contain keywords such as *"select"* and there are cases when the options given are part of other question sentences. However, we found that frequent words associated with *Select1* within a sentence are the wh-word *"which"* and the conjunction *"or"* – *"Which would you expect to have a higher density, hot water or cold wa-*

ter?". We experimented with adding two binary features to the word embeddings to check if the sentence contains any of these words and the performance reached an F_1-score = 0.26, doubling the performance obtained by using only word embeddings. These experiments show that our data can be learned by simple approaches and the performance can be improved by leveraging more of the question types' patterns.

The overall performance using word embeddings was computed in terms of weighted F_1-score over all classes at 0.511. As a baseline, we considered the *Majority Class* approach (all instances were labeled *OtherConstructedResponse*), which achieved a weighted F_1-score = 0.39. As can be observed, our approach surpassed the baseline by 12%. This suggests that our dataset can be effectively learned by a more complex approach by leveraging features specific to each question type, in addition to word embeddings.

5. Conclusion

In this paper, we presented an educational corpus collected from middle school science classrooms. The data contains questions asked by instructors during tutoring, which are annotated based on expected answer types. The goal of the proposed corpus is to enable the classification of instructors' questions in order to help with the analysis of student responses. This has the potential to improve the assessment process by facilitating the interpretation, comparison and contrasting of student responses. This will, in turn, provide the instructor with better formative assessment regarding the concepts understood, misunderstood or omitted by students in their answers. The corpus can also be leveraged in the question answering domain, where systems need information about the expected answer to a user's question in order to make accurate recommendations of answers.

The quality of the annotation is attested by high inter-rater reliability, K=0.75 (substantial agreement). We also tested our data on a supervised approach employing artificial neural networks and word embeddings, achieving a weighted F_1-score of 0.51, outperforming the baseline by 12%. This demonstrates that our annotations of a question's expected answer type are learnable, even by a relatively simple approach. The dataset will be released to the research community to improve and extend these findings.

6. Acknowledgements

This research was supported by the Institute of Education Sciences, U.S. Department of Education, Grant R305A120808 to University of North Texas. Opinions expressed are those of the authors.

7. Language Resource References

Bloom, B. S. (1956). *Taxonomy of educational objectives: The classification of educational goals: Cognitive Domain*. Longman.

Chi, M. T., Leeuw, N., Chiu, M.-H., and LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive science*, 18(3):439–477.

Conner, M. (1927). What a reference librarian should know. *Library Journal*, 52(8):415–418.

Hovy, E., Gerber, L., Hermjakob, U., Lin, C.-Y., and Ravichandran, D. (2001). Toward semantics-based answer pinpointing. In *Proceedings of the first international conference on Human language technology research*, pages 1–7. Association for Computational Linguistics.

İnce, İ. F. (2008). *Intellegent question classification for e-learning environments by data mining techniques*. Ph.D. thesis, Institute of Science.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Lehnert, W. G. (1977). The process of question answering. Technical report, DTIC Document.

Li, X. and Roth, D. (2002). Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Voorhees, E. M. and Tice, D. M. (1999). The trec-8 question answering track evaluation. In *TREC*, volume 1999, page 82.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, pages 2013–2018.

Incorporating Global Contexts into Sentence Embedding for Relational Extraction at the Paragraph Level with Distant Supervision

Eun-kyung Kim, Key-Sun Choi
Semantic Web Research Center, School of Computing, KAIST
291 Daehak-ro, Yuseong-gu, Daejeon, South Korea
{kekeeo, kschoi}@kaist.ac.kr

Abstract

The increased demand for structured knowledge has created considerable interest in relation extraction (RE) from large collections of documents. In particular, distant supervision can be used for RE without manual annotation costs. Nevertheless, this paradigm only extracts relations from individual sentences that contain two target entities. This paper explores the incorporation of global contexts derived from paragraph-into-sentence embedding as a means of compensating for the shortage of training data in distantly supervised RE. Experiments on RE from Korean Wikipedia show that the presented approach can learn an exact RE from sentences (including grammatically incoherent sentences) without syntactic parsing.

Keywords: Relation Extraction, Sentence Embedding, Pro-drop Languages

1. Introduction

As the demand for structured knowledge has increased, considerable interest has emerged in relation extraction (RE) from large collections of documents written in natural language. In particular, with "distant supervision" (DS) (Mintz et al., 2009; Hoffmann et al., 2011; Riedel et al., 2013), it is possible to extract the relationships between pairs of entities without human manual annotation using a knowledge base (KB); this heuristically aligns entities in texts to a given KB and then uses this alignment to train an RE system.

Although the DS strategy is a more effective method of automatically labeling training data than directly supervised labeling, DS-based approaches can extract only relations that are limited to a "single complete sentence" that contains two target entities. This makes it difficult to obtain both the subjects and object entities that participate in the KB in a single sentence, particularly in *null subject (or object) languages* such as Korean, Japanese, Arabic, and Swedish, that can leave the subject of a sentence unexpressed, unlike English which allows neither. It is also difficult to utilize DS-based approaches for English data when sentences have an informal, grammatically incoherent style, such as the style popularly used on Twitter, in discharge summaries of clinical texts (Marsh and Sager, 1982), or in a text shortened to bulleted lists in a Wikipedia article.

This point can be illustrated by considering the examples in Figure 1. S_1 contains a subject, object, and predicate, whereas the subject is omitted in S_2 because it is obvious in adjacent sentences in Korean, resulting in differences between the same sentence written in Korean and in English. Therein, we know S_2 is obviously a positive example for tuple `founderOf`(*Steve Jobs*, *Apple Inc.*), but we cannot label the training instance S_2 according to the traditional paradigm of an existing DS-based approach.

We propose a novel approach that performs RE across sentences, at the paragraph-level, and does not require labeled data. The proposed method builds upon sentence embedding with global context constraints by spanning multiple

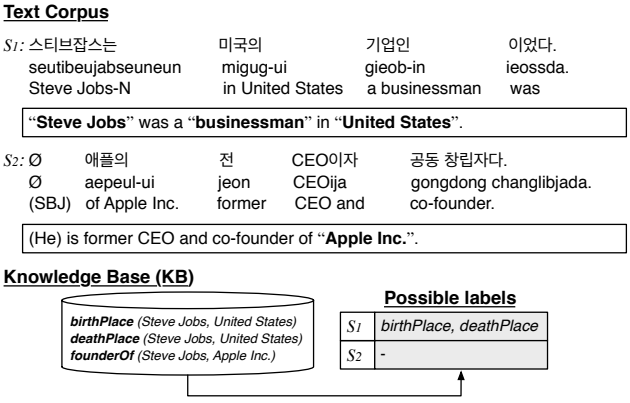


Figure 1: The English sentences are both correct translations of each Korean sentence. The entities in the sentence are marked in boldface with parenthesized boundaries.

sentences, which is useful for estimating omitted subjects and predicting relations. First, we specifically perform novel zero subject resolution with the entity-relation-based graph analysis by applying the centrality measure. This allows us to learn RE models for informal sentences and has the advantage of compensating for a shortage of training data in the DS-based approach in a DS-based approach to null subject languages. Then, we try to capture the discriminative context features of each document type, such as the specific logical pattern to the relational flow of text within a paragraph, to support sentence embedding.

Our work differs from previous related works in two ways: (1) we propose a method of RE at the paragraph-level—i.e., from a collection of multiple sentences— rather than extracting information from an independent single sentence; (2) our approach, which builds upon the sentence embedding, is more effective for language-independent extraction because it avoids high-level natural language processing (NLP) tools. Therefore, the present approach can be generally used for RE, even in languages for which NLP tools are lacking.

2. Related Work

We often encounter a lack of explicitly annotated text in RE, instead finding richly structured KBs such as DBpedia (Bizer et al., 2009) or Freebase (Bollacker et al., 2008), which has raised significant interest in learning RE using DS. Many DS-based approaches (Hoffmann et al., 2011; Roller and Stevenson, 2014; Tsai and Roth, 2016; Craven and Kumlien, 1999; Mintz et al., 2009) use simple but effective heuristics to align existing facts with unlabeled text. This automatically generated labeled text can be used as training data for supervised learners. Our work was inspired by Mintz et al. (2009), who adopted the Freebase for the distant supervision of the Wikipedia corpus. Unlike existing methods, we performed RE across sentences at the paragraph-level by extending the possibility of labeling incomplete sentences that were unavailable in the traditional DS-based approach. To the best of our knowledge, ours is the first DS-based approach to solve the problem of data sparseness by applying DS to the RE of informal sentences and alleviating DS assumptions.

Quirk and Poon (2016) introduced the RE method in two adjacent sentences using the DS approach. Peng et al. (2017) explored a framework for cross-sentence n-ary RE based on graph long short-term memory networks; they used a graph formula to provide a unified method of integrating various intra- and inter-sentential dependencies such as sequential, syntactic, and discourse relationships. According to their experiments on biomedical domains, use of RE beyond sentence boundaries can yield much more knowledge. In this context, we intend to find more information by spanning multiple sentences. While they are based on the various linguistic analyses, our proposed method can be differentiated by using contexts without syntactic information.

3. Relation Extraction at the Paragraph-level

We define our task as follows: Given a sentence s' that is a complemented form of an informal (e.g. subject-less) sentence s with marked entities e_1 and e_2 and a set of relations $R = \{r_1, ..., r_n\}$, we formulate the task of identifying the semantic relation as a standard classification problem as follows:

$$f : (P, E, L) \rightarrow R, \quad (1)$$

where P is the set of all paragraphs, a paragraph $p \in P$ is the set of contiguous sentences $\{s'_1, s'_2, \ldots, s'_m\}$, E is the set of entity pairs, and L denotes the set of relation flows. A relation flow $l \in l$ is a tuple $(\overleftarrow{s'}, \overrightarrow{s'})$ in which $\overleftarrow{s'} = \{r_1, r_2, ...\}$ is the set of labeled relation mentions in which the preceding sentences are $(s'_1, ..., s'_{i-1})$ and $\overrightarrow{s'}$ is the set of labeled relation mentions in which the succeeding sentences are $(s'_{i+1}, ..., s'_m)$ with a given target sentence s'_i. Our training objective is to learn a joint representation of the sentences and the logical pattern of the relation flow of text within paragraphs such that a regression layer can predict the correct label. We propose an architecture that learns sentence embedding after compensating sentences with zero subject resolution.

S_1	[Steve Jobs]$_{e_1}$ was a [Businessman]$_{e_2}$ in [United States]$_{e_3}$.
S_2	ø$_1$ Former [CEO]$_{e_4}$ and co-founder of [Apple Inc.]$_{e_5}$.
S_3	On October 5, 2011, ø$_2$ died of [Pancreatic cancer]$_{e_6}$.

Table 1: Entity-tagged sentences taken from the first paragraph of the "Steve Jobs" article in the Korean edition of Wikipedia.

3.1. Zero Subject Resolution using Graph Analysis of a Paragraph

The basic idea of our zero subject (entity) prediction is to perform tasks by finding the central entity being described within a paragraph without parsing. This prediction task allows us to apply our method to many languages in which NLP tools are lacking. We hypothesize that the paragraph consists of contiguous sentences that describe the central entity. Given an unlabeled textual training corpus (Φ = Wikipedia) and the supervision KB (Ψ = DBpedia), we first identify all paragraphs in Φ and entities ($\in \Psi$) in the sentence. For example, for S_1, S_2, and S_3 in Table 1, we use WikiLinks[1] to identify six DBpedia entities in total.

When entities in every sentence of a given paragraph are identified, the entity graph $\mathbb{G}$ in the paragraph is constructed based on the relation tuple in Ψ between a pair of entities that appear in the paragraph. Then, the center node is selected in $\mathbb{G}$ based on the degree centrality (Wasserman and Faust, 1994) for assigning the latent subject entity beyond the sentence boundary. Centrality is important if the entity links to many other entities with one or multiple links to other entities in $\mathbb{G}$. The example entity graph G generated with e_1–e_6 is shown in Figure 2; (a) represents a tuple in the given Ψ between a pair of entities that appear together in the paragraph and (b) represents a digraph of the tuples shown in (a), where "Steve Jobs" is selected as the pivot by the out-degree centrality measure.

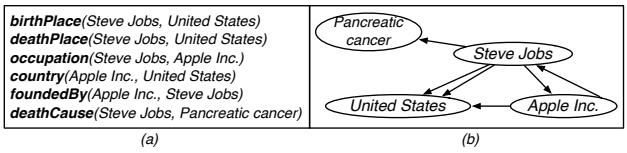


Figure 2: Example graph with given relation tuples between a pair of entities in the paragraph.

In this paper, the selected center entity is used to resolve zero subjects. Accordingly, a pair of entities that appear together in a single sentence, or head a pivoting entity and appear within a paragraph, is considered a potential relation instance. In the case of sentences S_2 and S_3 in Table 1, the concealed subjects, ø$_1$ and ø$_2$, both becomes "Steve Jobs" and provide an opportunity for acquiring possible labeled instances via heuristic alignments such as `founderOf`(*Steve Jobs*, *Apple Inc.*) and

[1] https://en.wikipedia.org/wiki/Help:Link

deathCause(*Steve Jobs*, *Pancreatic cancer*). Neither sentence explicitly states that "Steve Jobs" has such relationships, but have become useful for learning at training time by our extended model. For example, a sentence compensated by a pivot entity is syntactically incomplete, but we may derive a relatively large weight for the context feature associated with `founderOf` such as "former CEO and co-founder of."

At this stage, the only context features we use from s' are the words themselves. The vector representation of these words can be obtained using the Paragraph2Vec framework proposed by (Le and Mikolov, 2014), which maps each word to a vector and then uses a vector to represent all the words in the context window and thus predict the vector representation of the next word. The basic idea behind this method is to use an additional paragraph token (that maps to a vector space using a different matrix from that used to map the word) from the previous sentence in the document in the context window. Then, using the embedding matrix $\mathbf{E_{sen}} \in R^{D \times |V|}$ where D is the dimension of embedded words and $|V|$ is the dimension of the word vocabulary, we can obtain the embedding of the word. All words were randomly initialized and then updated during training.

3.2. Relational Flow Generation

Through the background of incorporating a global context into sentence embedding, the important intuition in our proposed model is understanding the whole paragraph as a single flow document. In this paper, we use the intuitive concept that if the semantic flow of a paragraph can be grasped, the relation type with which to classify the target sentence can be more clearly determined by the relation type with the preceding and succeeding sentences. For this, our auxiliary task is to determine the sequence of how preceding and succeeding sentences are classified into their respective relation types. Figure 3 shows an example of each paragraph that consists of contiguous sentences for two different types of entity. When there are two types of entities—in this case "baseball player" and "president"—it is possible to use a pattern in which there is no relation "party" (a type of relationship that points to a group of politically organized people) in the baseball player paragraph, and "team" and "position" relations are found that are very close to one another. According to this, for all three sentences (S_1–S_3 in Table 1), relationship flow of S_1 is ($\emptyset$, {*founderOf*, *deathCause*}), that of S_2 is ({*birthPlace*, *deathPlace*}, {*deathCause*}), and that of S_3 is ({*birthPlace*, *deathPlace*, *founderOf*}, $\emptyset$). We embed this relational flow, thereby aiming to learn continuous representations of it in vector space, similarly to embedding of words. Thus, we can also represent each element, i.e. the preceding and succeeding sequences of the relational flow, as two one-hot vectors of the K-dimension, where K is equal to the amount of relational flow. We then use the matrixes $\mathbf{E}^{\leftarrow}_{\mathbf{flow}} \in R^{D \times K}$ and $\mathbf{E}^{\rightarrow}_{\mathbf{flow}} \in R^{D \times K}$ to obtain its embedding.

In succession, we directly concatenate the sentence vector $\mathbf{E_{sen}}$ and the relational flow vectors $\mathbf{E}^{\leftarrow}_{\mathbf{flow}}$ and $\mathbf{E}^{\rightarrow}_{\mathbf{flow}}$ to form the final feature vector. This results in low-dimensional sentence embedding where semantically woven sentences and the relation flows of paragraphs reside in the same part of the space that presents the semantic relationship. We use this vector to train the machine learning algorithm and classify relationships.

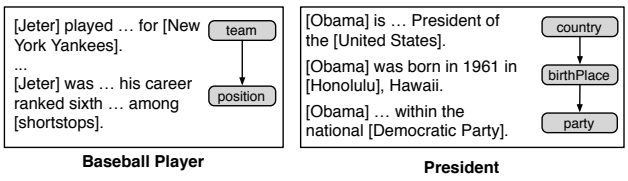


Figure 3: Relation sequence within a paragraph.

4. Experiments

We evaluated the performance of our proposed method by performing training and testing using the Korean version of Wikipedia as the textual corpus, specifically a snapshot from December 2016[2]. We used DBpedia to supervise background knowledge, which was a large KB of entities and relationships. As DBpedia provides tuple downloads in multiple languages[3], it was advantageous to build an efficient RE model for Koreans. KBs in non-Latin languages are relatively smaller than the English Freebase and DBpedia; our procedure used entities and tuples from DBpedia to provide relationship instances.

4.1. Implementation details

Distantly supervised RE can be viewed as a two-step process. This process (A) detects entities of interest and (B) determines the relationship between the possible set of entities. In this paper, we concentrate on the relationships between two entities, i.e., Step B. We processed the Wikipedia text using the following steps. (1) First, paragraphs are extracted from the article where a paragraph consists of two or more consecutive sentences that are separated by blank lines or different section names. (2) Second, the entities of sentences are identified using WikiLinks. In practice, an alternative entity recognition system may be required because the amount of text linked by WikiLinks is relatively small; however, that endeavor is beyond the scope of this study. (3) Third, central entities are selected from each paragraph by calculating the out-degree centrality based on the network model of the entity graph using the DBpedia tuple. (4) Fourth, sentences whose entity scope is recognized are tokenized. (5) Fifth, the pivot entity is employed to supplement the sentence and collect heuristically aligned data for the RE based on distant supervision. (6) Sixth, these labeled data are leveraged to construct sentence embedding, relation flow embedding, and finally to generate a single concentrated feature vector. (7) Finally, the RE model is trained with the feature vector to maximize the log probability of the correct relationship type. We converted each sentence into a word-level matrix in which each row was a sentence vector extracted from our model. Sentence vectors were learned from the Distributed Memory version of the Paragraph Vector (PV-DM) algorithm using training data

[2]https://dumps.wikimedia.org/kowiki/

[3]http://wiki.dbpedia.org/downloads-2016-04

to automatically learn and classify relationships into one of the 240 relation types in our evaluation dataset. PV-DM is an extension to Word2Vec (Mikolov et al., 2013) for learning document embeddings that was first applied to train using the entire corpus completely unsupervised.

We did not tune the initial learning rate (α) and minimum learning rate (α_{min}), and used the following values for all experiments: α = .025 and α_{min} = .002. The learning rate decreased linearly in each epoch from the initial rate to the minimum rate. We used the unchanged parameter min count (β) that represents the minimum frequency for times that a token must appear to be included in the Paragraph2Vec model's vocabulary. Our model set this as β = 1 to ensure that we treated all tokens in the context as meaningful and used them to train. We have optimized the embedding vector size (=400) and we used window sizes (=5) for the left and right fixed context windows. We ran an experiment with 10 epochs as the number of training iterations. All PV-DM training was carried out using the Gensim[4] library in Python. The next step was using a multiclass logistic regression classifier that was optimized using L-BFGS given the sentence embeddings inferred from the PV-DM model. Once the model had been trained, each sentence in the test dataset could be directly inferred.

4.2. Results Analysis of Extended Labeling

The original DS-based RE corresponds to a single sentence that contains two entities, but we extended this in this paper to tasks for two entities in a paragraph. We have made two extensions to the automated labeling schema, as described in Table 2. **Non-Extended** denotes the labeling results of two entities in a sentence according to the existing distant supervision paradigm. **Extended:Title** and **Extended:Pivot** are extensions of the label rather than Non-Extended. **Extended:Title** interprets the title of the Wikipedia document as the head entity because the title is the protagonist in the document, whereas **Extended:Pivot** represents the extension of the central pivoting entity in the paragraph as the subject entity, i.e. the proposed approach. Table 2 shows the proportion of judged documents for 50 sample documents and the precision—the proportion of relevant labeled sentences for RE—among that set.

It is clear from this table that the **Extended:Pivot** run achieved a higher or similar precision for the judged documents that it returned, but returned larger relevant labeled sentences (i.e. Positive Labels), and hence achieved a higher recall@R score, where R is the number of relevant documents in the collection. The **Extended:Title** method can also raise the precision and recall compared to the default DS paradigm in Wikipedia, but this is difficult to scale to a web-scale without a document title.

4.3. Held-out Evaluation for RE

We evaluated our RE model as a "held-out" evaluation. Such an evaluation is conducted automatically by withholding half of the DBpedia relationship knowledge during training and comparing the newly discovered relationship instances against the withheld data. The goal of automatic evaluation focuses on the accuracies of relation labels for each entity pair instead of the accuracies of the relation labels for each instance. We compared our model with the part-of-speech (POS) tag feature as a baseline that relies on the POS tag sequences of sentences for classification. Table 3 shows the results for the baseline for comparison with our algorithm. The best result was achieved using sentence embedding with relational flow, which led to an F1-measure of 59%. Although there is much room for improvement in precision and recall, our results indicated that it could be useful for extracting the relationship with small amounts of labeled data without advanced NLP tools such as a parser.

[4]https://radimrehurek.com/gensim/

	Total Labels	Sample Analysis (50 Doc.)	
		Positive Labels	Precision
Non-Extended	159,148	172	0.7257
Extended:Title	413,104	331	0.7405
Extended:Pivot	582,596	481	0.7527

Table 2: The corpus statistics before and after labeling extensions based on the distant supervision paradigm.

Features		Precision	Recall	F1-score
	Baseline	0.41	0.14	0.2087
$\mathcal{S}$	Non-Extended	0.45	0.39	0.4179
	Extended	0.53	0.50	0.5146
$\mathcal{S}+\mathcal{F}$	Non-Extended	0.59	0.54	0.5639
	Extended	0.60	0.59	0.5950

Table 3: Best F1-score measures with Precision and Recall for different feature sets, where $\mathcal{S}$ denotes the "sentence embedding" and $\mathcal{F}$ denotes the "relation flow embedding," by the incremental embedding of features compared with the POS-baseline.

5. Conclusion

In this paper, we focused on the distant supervision paradigm and proceeded to RE from passages that did not contain both of the entities that are expected to participate in a relation. We showed that it was possible to use a DS-based model that does not require labeling to represent the contexts of sentences and the surrounding relationship mentions to enable relation classification at the paragraph level. Experiments on Korean Wikipedia were conducted and showed the model's effectiveness in practical use. In future research, we intend to implement our technique on a much larger scale and with a more refined set of relation classifications. Alternatively, we may leverage cross-lingual joint techniques to transfer knowledge from other languages and to include joint learning with entity linking.

Acknowledgments

This work was supported by the Industrial Strategic technology development program (10072064, Development of Novel Artificial Intelligence Technologies To Assist Imaging Diagnosis of Pulmonary, Hepatic, and Cardiac Diseases and Their Integration into Commercial Clinical PACS Platforms) funded by the Ministry of Trade Industry and Energy (MI, Korea) and supported by the Bio & Medical

Technology Development Program of the NRF funded by the Korean government, Ministry of Science and ICT(NRF-2015M3A9A7029725)

6. Bibliographical References

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia – a crystallization point for the web of data.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *In SIGMOD Conference*, pages 1247–1250.

Craven, M. and Kumlien, J. (1999). Constructing biological knowledge bases by extracting information from text sources. In *ISMB-99*.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L. S., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In Dekang Lin, et al., editors, *ACL*, pages 541–550. The Association for Computer Linguistics.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In Tony Jebara et al., editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196. JMLR Workshop and Conference Proceedings.

Marsh, E. and Sager, N. (1982). Analysis and processing of compact text. In *Proceedings of the 9th Conference on Computational Linguistics - Volume 1*, COLING '82, pages 201–206, Czechoslovakia. Academia Praha.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peng, N., Poon, H., Quirk, C., Toutanova, K., and Yih, W.-t. (2017). Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Quirk, C. and Poon, H. (2016). Distant supervision for relation extraction beyond the sentence boundary. *CoRR*, abs/1609.04873.

Riedel, S., Yao, L., McCallum, A., and Marlin, B. M. (2013). Relation extraction with matrix factorization and universal schemas. In Lucy Vanderwende, et al., editors, *HLT-NAACL*, pages 74–84. The Association for Computational Linguistics.

Roller, R. and Stevenson, M., (2014). *Self-supervised Relation Extraction Using UMLS*, pages 116–127. Springer International Publishing, Cham.

Tsai, C.-T. and Roth, D. (2016). Concept grounding to multiple knowledge bases via indirect supervision. *TACL*, 4:141–154.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Number 8 in Structural analysis in the social sciences. Cambridge University Press, 1 edition.

MCScript: A Novel Dataset for Assessing Machine Comprehension Using Script Knowledge

Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, Manfred Pinkal
Saarland University
Saarbrücken, Germany
{simono|ashutosh|mroth|stth|pinkal}@coli.uni-saarland.de

Abstract

We introduce a large dataset of narrative texts and questions about these texts, intended to be used in a machine comprehension task that requires reasoning using commonsense knowledge. Our dataset complements similar datasets in that we focus on stories about everyday activities, such as going to the movies or working in the garden, and that the questions require commonsense knowledge, or more specifically, *script knowledge*, to be answered. We show that our mode of data collection via crowdsourcing results in a substantial amount of such inference questions. The dataset forms the basis of a shared task on commonsense and script knowledge organized at SemEval 2018 and provides challenging test cases for the broader natural language understanding community.

Keywords: machine comprehension, reading comprehension, commonsense knowledge, script knowledge

1. Introduction

Ambiguity and implicitness are inherent properties of natural language that cause challenges for computational models of language understanding. In everyday communication, people assume a shared common ground which forms a basis for efficiently resolving ambiguities and for inferring implicit information. Thus, recoverable information is often left unmentioned or underspecified. Such information may include encyclopedic and commonsense knowledge. This work focuses on commonsense knowledge about everyday activities, so-called *scripts*.

This paper introduces a dataset to evaluate natural language understanding approaches with a focus on interpretation processes requiring inference based on commonsense knowledge. In particular, we present *MCScript*, a dataset for assessing the contribution of script knowledge to machine comprehension. Scripts are sequences of events describing stereotypical human activities (also called scenarios), for example baking a cake or taking a bus (Schank and Abelson, 1975). To illustrate the importance of script knowledge, consider Example (1):

(1) The waitress brought Rachel's order. She ate the food with great pleasure.

Without using commonsense knowledge, it may be difficult to tell who ate the food: Rachel or the waitress. In contrast, if we utilize commonsense knowledge, in particular, script knowledge about the EATING IN A RESTAURANT scenario, we can make the following inferences: Rachel is most likely a customer, since she received an order. It is usually the customer, and not the waitress, who eats the ordered food. So *She* most likely refers to Rachel.

Various approaches for script knowledge extraction and processing have been proposed in recent years. However, systems have been evaluated for specific aspects of script knowledge only, such as event ordering (Modi and Titov, 2014a; Modi and Titov, 2014b), event paraphrasing (Regneri et al., 2010; Wanzare et al., 2017) or event prediction (namely, the narrative cloze task (Chambers and Jurafsky, 2008; Chambers and Jurafsky, 2009; Pichotta and Mooney, 2014; Pichotta and Mooney, 2016; Modi, 2016)). These evaluation methods lack a clear connection to real-world tasks. Our MCScript dataset provides an extrinsic evaluation framework, based on text comprehension involving commonsense knowledge. This framework makes it possible to assess system performance in a multiple-choice question answering setting, without imposing any specific structural or methodical requirements.

T I wanted to plant a tree. I went to the home and garden store and picked a nice oak. Afterwards, I planted it in my garden.

Q1 What was used to dig the hole?
a. a shovel b. his bare hands

Q2 When did he plant the tree?
a. after watering it b. after taking it home

Figure 1: An example for a text snippet with two reading comprehension questions.

MCScript is a collection of (1) narrative texts, (2) questions of various types referring to these texts, and (3) pairs of answer candidates for each question. It comprises approx. 2,100 texts and a total of approx. 14,000 questions. Answering a substantial subset of questions requires knowledge beyond the facts mentioned in the text, i.e. it requires inference using commonsense knowledge about everyday activities. An example is given in Figure 1. For both questions, the correct choice for an answer requires commonsense knowledge about the activity of planting a tree, which goes beyond what is mentioned in the text. Texts, questions, and answers were obtained through crowdsourcing. In order to ensure high quality, we manually validated and filtered the dataset. Due to our design of the data ac-

quisition process, we ended up with a substantial subset of questions that require commonsense inference (27.4%).

2. Corpus

Machine comprehension datasets consist of three main components: texts, questions and answers. In this section, we describe our data collection for these 3 components. We first describe a series of pilot studies that we conducted in order to collect commonsense inference questions (Section 2.1.). In Section 2.2., we discuss the resulting data collection of questions, texts and answers via crowdsourcing on Amazon Mechanical Turk[1] (henceforth *MTurk*). Section 2.3. gives information about some necessary postprocessing steps and the dataset validation. Lastly, Section 2.4. gives statistics about the final dataset.

2.1. Pilot Study

As a starting point for our pilots, we made use of texts from the *InScript* corpus (Modi et al., 2016), which provides stories centered around everyday situations (see Section 2.2.2.). We conducted three different pilot studies to determine the best way of collecting questions that require inference over commonsense knowledge:

The most intuitive way of collecting reading comprehension questions is to show texts to workers and let them formulate questions and answers on the texts, which is what we tried internally in a *first pilot*. Since our focus is to provide an evaluation framework for inference over commonsense knowledge, we manually assessed the number of questions that indeed require common sense knowledge. We found too many questions and answers collected in this manner to be lexically close to the text.

In a *second pilot*, we investigated the option to take the questions collected for one text and show them as questions for another text of the same scenario. While this method resulted in a larger number of questions that required inference, we found the majority of questions to not make sense at all when paired with another text. Many questions were specific to a text (and not to a scenario), requiring details that could not be answered from other texts.

Since the two previous pilot setups resulted in questions that centered around the texts themselves, we decided for a *third pilot* to not show workers any specific texts at all. Instead, we asked for questions that centered around a specific script scenario (e.g. EATING IN A RESTAURANT). We found this mode of collection to result in questions that have the right level of specificity for our purposes: namely, questions that are related to a scenario and that can be answered from different texts (about that scenario), but for which a text does not need to provide the answer explicitly. The next section will describe the mode of collection chosen for the final dataset, based on the third pilot, in more detail.

2.2. Data Collection

2.2.1. Scenario Selection

As mentioned in the previous section, we decided to base the question collection on script scenarios rather than specific texts. As a starting point for our data collection, we use scenarios from three script data collections (Regneri et al., 2010; Singh et al., 2002; Wanzare et al., 2016). Together, these resources contain more than 200 scenarios. To make sure that scenarios have different complexity and content, we selected 80 of them and came up with 20 new scenarios. Together with the 10 scenarios from InScript, we end up with a total of 110 scenarios.

2.2.2. Texts

For the collection of texts, we followed Modi et al. (2016), where workers were asked to write a story about a given activity "as if explaining it to a child". This results in elaborate and explicit texts that are centered around a single scenario. Consequently, the texts are syntactically simple, facilitating machine comprehension models to focus on semantic challenges and inference. We collected 20 texts for each scenario. Each participant was allowed to write only one story per scenario, but work on as many scenarios as they liked. For each of the 10 scenarios from InScript, we randomly selected 20 existing texts from that resource.

2.2.3. Questions

For collecting questions, workers were instructed to "imagine they told a story about a certain scenario to a child and want to test if the child understood everything correctly". This instruction also ensured that questions are linguistically simple, elaborate and explicit. Workers were asked to formulate questions about details of such a situation, i.e. independent of a concrete narrative. This resulted in questions, the answer to which is not literally mentioned in the text.

To cover a broad range of question types, we asked participants to write 3 temporal questions (asking about time points and event order), 3 content questions (asking about persons or details in the scenario) and 3 reasoning questions (asking how or why something happened). They were also asked to formulate 6 free questions, which resulted in a total of 15 questions. Asking each worker for a high number of questions enforced that more creative questions were formulated, which go beyond obvious questions for a scenario.

Since participants were not shown a concrete story, we asked them to use the neutral pronoun "they" to address the protagonist of the story. We permitted participants to work on as many scenarios as desired and we collected questions from 10 participants per scenario.

2.2.4. Answers

Our mode of question collection results in questions that are not associated with specific texts. For each text, we collected answers for 15 questions that were randomly selected from the same scenario. Since questions and texts were collected independently, answering a random question is not always possible for a given text. Therefore, we carried out answer collection in two steps. In the first step, we asked participants to assign a category to each text–question pair. We distinguish two categories of answerable questions: The category *text-based* was assigned to questions that can be answered from the text directly. If the answer could only be inferred by using commonsense knowledge, the category

[1] www.mturk.com

answerable		not answerable	
text-based	**script-based**	**unknown**	**unfitting**
10,160	3,914	9,974	3,172
14,074		13,246	

Table 1: Distribution of question categories

script-based was assigned. Making this distinction is interesting for evaluation purposes, since it enables us to estimate the number of commonsense inference questions. For questions that did not make sense at all given a text, *unfitting* was assigned. If a question made sense for a text, but it was impossible to find an answer, the label *unknown* was used.

In a second step, we told participants to formulate a plausible correct and a plausible incorrect answer candidate to answerable questions (*text-based* or *script-based*). To level out the effort between answerable and non-answerable questions, participants had to write a new question when selecting *unknown* or *unfitting*.

In order to get reliable judgments about whether or not a question can be answered, we collected data from 5 participants for each question and decided on the final category via majority vote (at least 3 out of 5). Consequently, for each question with a majority vote on either *text-based* or *script-based*, there are 3 to 5 correct and incorrect answer candidates, one from each participant who agreed on the category. Questions without a clear majority vote or with ties were not included in the dataset.

2.2.5. Data Post-Processing

We performed four post-processing steps on the collected data.

- We manually filtered out texts that were instructional rather than narrative.
- All texts, questions and answers were spellchecked by running aSpell[2] and manually inspecting all corrections proposed by the spellchecker.
- We found that some participants did not use "they" when referring to the protagonist. We identified "I", "you", "he", "she", "my", "your", "his", "her" and "the person" as most common alternatives and replaced each appearance manually with "they" or "their", if appropriate.
- We manually filtered out invalid questions, e.g. questions that are suggestive ("Should you ask an adult before using a knife?") or that ask for the personal opinion of the reader ("Do you think going to the museum was a good idea?").

2.3. Answer Selection and Validation

We finalized the dataset by selecting one correct and one incorrect answer for each question–text pair. To increase the proportion of non-trivial inference cases, we chose the candidate with the *lowest* lexical overlap with the text from the set of correct answer candidates as *correct* answer. Using this principle also for incorrect answers leads to problems. We found that many incorrect candidates were not plausible answers to a given question. Instead of selecting a candidate based on overlap, we hence decided to rely on majority vote and selected the candidate from the set of incorrect answers that was most often mentioned.

For this step, we normalized each candidate by lowercasing, deleting punctuation and stop words (articles, *and*, *to* and *or*), and transforming all number words into digits, using *text2num*[3]. We merged all answers that were string-identical, contained another answer, or had a Levenshtein distance (Levenshtein, 1966) of 3 or less to another answer. The "most frequent answer" was then selected based on how many other answers it was merged with. Only if there was no majority, we selected the candidate with the highest overlap with the text as a fallback.

Due to annotation mistakes, we found a small number of chosen correct and incorrect answers to be inappropriate, that is, some "correct" answers were actually incorrect and vice versa. Therefore, we manually validated the complete dataset in a final step. We asked annotators to read all texts, questions, and answers, and to mark for each question whether the correct and incorrect answers were appropriate. If an answer was inappropriate or contained any errors, they selected a different answer from the set of collected candidates. For approximately 11.5% of the questions, at least one answer was replaced. 135 questions (approx. 1%) were excluded from the dataset because no appropriate correct or incorrect answer could be found.

2.4. Data Statistics

For all experiments, we admitted only experienced MTurk workers who are based in the US. One HIT[4] consisted of writing one text for the text collection, formulating 15 questions for the question collection, or finding 15 pairs of answers for the answer collection. We paid $0.50 per HIT for the text and question collection, and $0.60 per HIT for the answer collection.

More than 2,100 texts were paired with 15 questions each, resulting in a total number of approx. 32,000 annotated questions. For 13% of the questions, the workers did not agree on one of the 4 categories with a 3 out of 5 majority, so we did not include these questions in our dataset.

The distribution of category labels on the remaining 87% is shown in Table 1. 14,074 (52%) questions could be answered. Out of the answerable questions, 10,160 could be answered from the text directly (*text-based*) and 3,914 questions required the use of commonsense knowledge (*script-based*). After removing 135 questions during the validation, the final dataset comprises 13,939 questions, 3,827 of which require commonsense knowledge (i.e. 27.4%). This ratio was manually verified based on a random sample of questions.

We split the dataset into training (9,731 questions on 1,470 texts), development (1,411 questions on 219 texts), and test set (2,797 questions on 430 texts). Each text appears only

[2] `http://aspell.net/`

[3] `https://github.com/ghewgill/text2num`

[4] A *Human Intelligence Task* (HIT) is one single experimental item in MTurk.

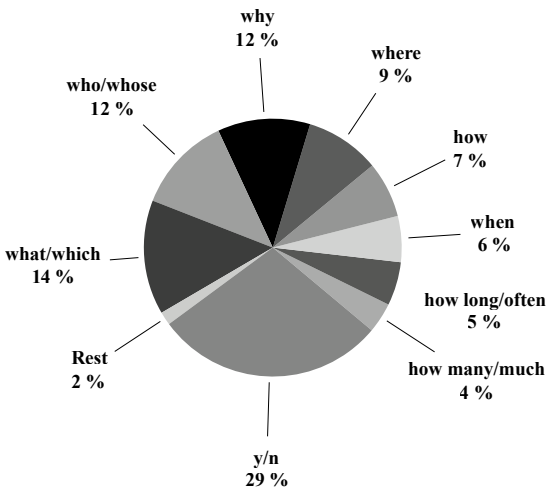


Figure 2: Distribution of question types in the data.

in one of the three sets. The complete set of texts for 5 scenarios was held out for the test set.

The average text, question, and answer length is 196.0 words, 7.8 words, and 3.6 words, respectively. On average, there are 6.7 questions per text.

Figure 2 shows the distribution of question types in the dataset, which we identified using simple heuristics based on the first words of a question: Yes/no questions were identified as questions starting with an auxiliary or modal verb, all other question types were determined based on the question word.

We found that 29% of all questions are yes/no questions. Questions about details of a situation (such as *what/ which* and *who*) form the second most frequent question category. Temporal questions (*when* and *how long/often*) form approx. 11% of all questions. We leave a more detailed analysis of question types for future work.

3. Data Analysis

As can be seen from the data statistics, our mode of collection leads to a substantial proportion of questions that require inference using commonsense knowledge. Still, the dataset contains a large number of questions in which the answer is explicitly contained or implied by the text: Figure 3 shows passages from an example text of the dataset together with two such questions. For question Q1, the answer is given literally in the text. Answering question Q2 is not as simple; it can be solved, however, via standard semantic relatedness information (chicken and hotdogs are meat; water, soda and juice are drinks).

The following cases require commonsense inference to be decided. In all these cases, the answers are not overtly contained nor easily derivable from the respective texts. We do not show the full texts, but only the scenario names for each question.

(2) BORROWING A BOOK FROM THE LIBRARY
Did they have to pay anything to borrow the book?
a. yes
b. no

(3) CHANGING A BABY DIAPER
Did they throw away the old diaper?
a. Yes, they put it into the bin.
b. No, they kept it for a while.

(4) CLEANING THE TABLE
When did they clean the table?
a. After a meal
b. Before they ate

(5) PREPARING A PICNIC
Who is packing the picnic?
a. the children
b. the parents

(6) TAKING A SHOWER
How long did the shower take?
a. a few hours
b. a few minutes

T It was time to prepare for the picnic that we had plans for the last couple weeks. ...I needed to set up the cooler, which included bottles of water, soda and juice to keep everyone hydrated. Then I needed to ensure that we had all the food we intended to bring or cook. So at home, I prepared baked beans, green beans and macaroni and cheese. ...But in a cooler, I packed chicken, hotdogs, hamburgers and rots that were to be cooked on the grill once we were at the picnic location.

Q1 What did they bring to drink?
a. Water, soda and juice.
b. Water, wine coolers and sports drinks.

Q2 What type of food did they pack?
a. Meat, drinks and side dishes.
b. Pasta salad only.

Figure 3: An example text with 2 questions from MCScript

Example 2 refers to a library setting. Script knowledge helps in assessing that usually, *paying* is not an event when borrowing a book, which answers the question. Similarly, event information helps in answering the questions in Examples 3 and 4. In Example 5, knowledge about the typical role of parents in the preparation of a picnic will enable a plausibility decision. Similarly, in Example 6, it is commonsense knowledge that showers usually take a few minutes rather than hours.

(7) MAKING BREAKFAST
What time of the day is breakfast eaten?
a. at night
b. in the morning

There are also cases in which the answer can be inferred from the text, but where commonsense knowledge is still beneficial: The text for example 7 does not contain the information that breakfast is eaten in the morning, but it could still be inferred from many pointers in the text (e.g. phrases such as *I woke up*), or from commonsense knowledge.
These few examples illustrate that our dataset covers questions with a wide spectrum of difficulty, from rather simple questions that can be answered from the text to challenging inference problems.

4. Experiments

In this section, we assess the performance of baseline models on MCScript, using accuracy as the evaluation measure. We employ models of differing complexity: two unsupervised models using only word information and distributional information, respectively, and two supervised neural models. We assess performance on two dimensions: One, we show how well the models perform on text-based questions as compared to questions that require common sense for finding the correct answer. Two, we evaluate each model for each different question type.

4.1. Models

Word Matching Baseline
We first use a simple word matching baseline, by selecting the answer that has the highest literal overlap with the text. In case of a tie, we randomly select one of the answers.

Sliding Window
The second baseline is a sliding window approach that looks at windows of w tokens on the text. Each text and each answer are represented as a sequence of word embeddings. The embeddings for each window of size w and each answer are then averaged to derive window and answer representations, respectively. The answer with the lowest cosine distance to one of the windows of the text is then selected as correct.

Bilinear Model
We employ a simple neural model as a third baseline. In this model, each text, question, and answer is represented by a vector. For a given sequence of words $w_1 \dots w_n$, we compute this representation by averaging over the components of the word embeddings $\mathbf{w}_i$ that correspond to a word w_i, and then apply a linear transformation using a weight matrix. This procedure is applied to each answer a to derive an answer representation $\mathbf{a}$. The representation of a text $\mathbf{t}$ and of a question $\mathbf{q}$ are computed in the same way. We use different weight matrices for $\mathbf{a}$, $\mathbf{t}$ and $\mathbf{q}$, respectively. A combined representation $\mathbf{p}$ for the text–question pair is then constructed using a bilinear transformation matrix $\mathbf{W}$:

$$\mathbf{p} = \mathbf{t}^\top \mathbf{W} \mathbf{q} \tag{1}$$

We compute a score for each answer by using the dot product and pass the scores for both answers through a softmax layer for prediction. The probability p for an answer a to be correct is thus defined as:

$$p(a|t,q) = softmax(\mathbf{p}^\top \mathbf{a}) \tag{2}$$

Attentive Reader
The attentive reader is a well-established machine comprehension model that reaches good performance e.g. on the *CNN/Daily Mail* corpus (Hermann et al., 2015; Chen et al., 2016). We use the model formulation by Chen et al. (2016) and Lai et al. (2017), who employ bilinear weight functions to compute both attention and answer-text fit. Bidirectional GRUs are used to encode questions, texts and answers into hidden representations. For a question q and an answer a, the last state of the GRUs, $\mathbf{q}$ and $\mathbf{a}$, are used as representations, while the text is encoded as a sequence of hidden states $\mathbf{t}_1 ... \mathbf{t}_n$. We then compute an attention score s_j for each hidden state $\mathbf{t}_j$ using the question representation $\mathbf{q}$, a weight matrix $\mathbf{W}_a$, and an attention bias b. Last, a text representation $\mathbf{t}$ is computed as a weighted average of the hidden representations:

$$\begin{aligned} s_j &= softmax_j(\mathbf{t}_j^\top \mathbf{W}_a \mathbf{q} + b) \\ \mathbf{t} &= \sum_j s_j \mathbf{t}_j \end{aligned} \tag{3}$$

The probability p of answer a being correct is then predicted using another bilinear weight matrix $\mathbf{W}_s$, followed by an application of the softmax function over both answer options for the question:

$$p(a|t,q) = softmax(\mathbf{t}^\top \mathbf{W}_s \mathbf{a}) \tag{4}$$

4.2. Implementation Details

Texts, questions and answers were tokenized using NLTK[5] and lowercased. We used 100-dimensional GloVe vectors[6] (Pennington et al., 2014) to embed each token. For the neural models, the embeddings are used to initialize the token representations, and are refined during training. For the sliding similarity window approach, we set $w = 10$.
The vocabulary of the neural models was extracted from training and development data. For optimizing the bilinear model and the attentive reader, we used vanilla stochastic gradient descent with gradient clipping, if the norm of gradients exceeds 10. The size of the hidden layers was tuned to 64, with a learning rate of 0.2, for both models. We apply a dropout of 0.5 to the word embeddings. Batch size was set to 25 and all models were trained for 150 epochs. During training, we measured performance on the development set, and we selected the model from the best performing epoch for testing.

4.3. Results and Evaluation

Human Upper Bound
As an upper bound for model performance, we assess how well humans can solve our task. Two trained annotators labeled the correct answer on all instances of the test set. They agreed with the gold standard in 98.2 % of cases. This result shows that humans have no difficulty in finding the correct answer, irrespective of the question type.

[5] http://www.nltk.org/
[6] https://nlp.stanford.edu/projects/glove/

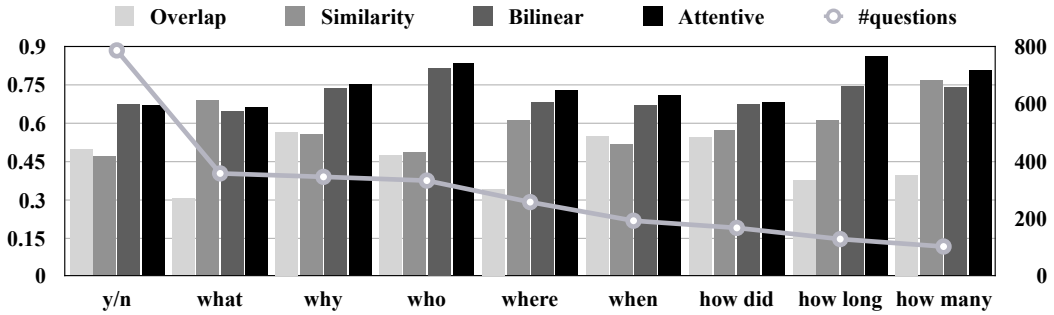


Figure 4: Accuracy values of the baseline models on question types appearing > 25 times.

Model	*Text*	*CS*	*Total*
Chance	50.0	50.0	50.0
Word Overlap	41.8	59.0	54.4
Sliding Window	55.7	53.1	55.0
Bilinear Model	69.8	71.4	70.2
Attentive Reader	**70.9**	**75.2**	**72.0**
Human Performance			98.2

Table 2: Accuracy of the baseline systems on text-based (*Text*), on commonsense-based questions (*CS*), and on the whole test set (*Total*). All numbers are percentages.

Performance of the Baseline Models

Table 2 shows the performance of the baseline models as compared to the human upper bound and a random baseline. As can be seen, neural models have a clear advantage over the pure word overlap baseline, which performs worst, with an accuracy of 54.4%.

The low accuracy is mostly due to the nature of correct answers in our data: Each correct answer has a low overlap with the text by design. Since the overlap model selects the answer with a high overlap to the text, it does not perform well. In particular, this also explains the very bad result on text-based questions. The sliding similarity window model does not outperform the simple word overlap model by a large margin: Distributional information alone is insufficient to handle complex questions in the dataset.

Both neural models outperform the unsupervised baselines by a large margin. When comparing the two models, the attentive reader is able to beat the bilinear model by only 1.8%. A possible explanation for this is that the attentive reader only attends to the text. Since many questions cannot be directly answered from the text, the attentive reader is not able to perform significantly better than a simpler neural model.

What is surprising is that the attentive reader works better on commonsense-based questions than on text questions. This can be explained by the fact that many commonsense questions do have prototypical answers within a scenario, irrespective of the text. The attentive reader is apparently able to just memorize these prototypical answers, thus achieving higher accuracy.

Inspecting attention values of the attentive reader, we found that in most cases, the model is unable to properly attend to the relevant parts of the text, even when the answer is literally given in the text. A possible explanation is that the model is confused by the large amount of questions that cannot be answered from the text directly, which might confound the computation of attention values.

Also, the attentive reader was originally constructed for reconstructing literal text spans as answers. Our mode of answer collection, however, results in many correct answers that cannot be found verbatim in the text. This presents difficulties for the attention mechanism.

The fact that an attention model outperforms a simple bilinear baseline only marginally shows that MCScript poses a new challenge to machine comprehension systems. Models concentrating solely on the text are insufficient to perform well on the data.

Performance on Question Types

Figure 4 gives accuracy values of all baseline systems on the most frequent question types (appearing $>$25 times in the test data), as determined based on the question words (see Section 2.4.). The numbers depicted on the left-hand side of the y-axis represent model accuracy. The right-hand side of the y-axis indicates the number of times a question type appears in the test data.

The neural models unsurprisingly outperform the other models in most cases, and the difference for *who* questions is largest. A large number of these questions ask for the narrator of the story, who is usually not mentioned literally in the text, since most stories are written in the first person.

It is also apparent that all models perform rather badly on *yes/no* questions. Each model basically compares the answer to some representation of the text. For yes/no questions, this makes sense for less than half of all cases. For the majority of yes/no questions, however, answers consist only of *yes* or *no*, without further content words.

5. Related Work

In recent years, a number of reading comprehension datasets have been proposed, including *MCTest* (Richardson et al., 2013), *BAbI* (Weston et al., 2015), the *Children's Book Test* (*CBT*, Hill et al. (2015)), *CNN/Daily Mail* (Hermann et al., 2015), the *Stanford Question Answering Dataset* (*SQuAD*, Rajpurkar et al. (2016)), and *RACE* (Lai et al., 2017). These datasets differ with respect to text type (Wikipedia texts, examination texts, etc.), mode of answer selection (span-based, multiple choice, etc.) and test systems regarding different aspects of language understand-

ing, but they do not explicitly address commonsense knowledge.

Two notable exceptions are the *NewsQA* and *TriviaQA* datasets. *NewsQA* (Trischler et al., 2017) is a dataset of newswire texts from CNN with questions and answers written by crowdsourcing workers. *NewsQA* closely resembles our own data collection with respect to the method of data acquisition. As for our data collection, full texts were not shown to workers as a basis for question formulation, but only the text's title and a short summary, to avoid literal repetitions and support the generation of non-trivial questions requiring background knowledge. The NewsQA text collection differs from ours in domain and genre (newswire texts vs. narrative stories about everyday events). Knowledge required to answer the questions is mostly factual knowledge and script knowledge is only marginally relevant. Also, the task is not exactly question answering, but identification of document passages containing the answer. *TriviaQA* (Joshi, Mandar and Choi, Eunsol and Weld, Daniel S. and Zettlemoyer, Luke, 2017) is a corpus that contains automatically collected question-answer pairs from 14 trivia and quiz-league websites, together with web-crawled evidence documents from *Wikipedia* and *Bing*. While a majority of questions require world knowledge for finding the correct answer, it is mostly factual knowledge.

6. Summary

We present a new dataset for the task of machine comprehension focussing on commonsense knowledge. Questions were collected based on script scenarios, rather than individual texts, which resulted in question–answer pairs that explicitly involve commonsense knowledge. In contrast to previous evaluation tasks, this setup allows us for the first time to assess the contribution of script knowledge for computational models of language understanding in a real-world evaluation scenario.

We expect our dataset to become a standard benchmark for testing models of commonsense and script knowledge. Human performance shows that the dataset is highly reliable. The results of several baselines, in contrast, illustrate that our task provides challenging test cases for the broader natural language processing community. MCScript forms the basis of a shared task organized at SemEval 2018. The dataset is available at `http://www.sfb1102.uni-saarland.de/?page_id=2582`.

Acknowledgements

We thank the reviewers for their helpful comments. We also thank Florian Pusse for the help with the MTurk experiments and our student assistants Christine Schäfer, Damyana Gateva, Leonie Harter, Sarah Mameche, Stefan Grünewald and Tatiana Anikina for help with the annotations. This research was funded by the German Research Foundation (DFG) as part of SFB 1102 'Information Density and Linguistic Encoding' and EXC 284 'Multimodal Computing and Interaction'.

7. Bibliographical References

Chambers, N. and Jurafsky, D. (2008). Unsupervised Learning of Narrative Event Chains. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 789–797, Columbus, Ohio, June. Association for Computational Linguistics.

Chambers, N. and Jurafsky, D. (2009). Unsupervised Learning of Narrative Schemas and their Participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 602–610, Singapore, August. Association for Computational Linguistics.

Chen, D., Bolton, J., and Manning, C. D. (2016). A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2358–2367.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., and Blunsom, P. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Hill, F., Bordes, A., Chopra, S., and Weston, J. (2015). The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. *arXiv preprint arXiv:1511.02301*.

Joshi, Mandar and Choi, Eunsol and Weld, Daniel S. and Zettlemoyer, Luke. (2017). TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July. Association for Computational Linguistics.

Lai, G., Xie, Q., Liu, H., Yang, Y., and Hovy, E. (2017). RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady*, volume 10, pages 707–710.

Modi, A. and Titov, I. (2014a). Inducing Neural Models of Script Knowledge. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*, Baltimore, MD, USA.

Modi, A. and Titov, I. (2014b). Learning Semantic Script Knowledge with Event Embeddings. In *Proceedings of the 2nd International Conference on Learning Representations (Workshop track)*, Banff, Canada.

Modi, A., Anikina, T., Ostermann, S., and Pinkal, M. (2016). InScript: Narrative texts annotated with script information. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, may. European Language Resources Association (ELRA).

Modi, A. (2016). Event Embeddings for Semantic Script Modeling. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 75–83, Berlin, Germany, August. Association for

Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Pichotta, K. and Mooney, R. J. (2014). Statistical Script Learning with Multi-Argument Events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 220–229, Gothenburg, Sweden, April.

Pichotta, K. and Mooney, R. J. (2016). Using Sentence-Level LSTM Language Models for Script Inference. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL-16)*, Berlin, Germany.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Regneri, M., Koller, A., and Pinkal, M. (2010). Learning Script Knowledge with Web Experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 979–988, Uppsala, Sweden, July. Association for Computational Linguistics.

Richardson, M., Burges, C. J., and Renshaw, E. (2013). MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Seattle, Washington, USA, October. Association for Computational Linguistics.

Schank, R. C. and Abelson, R. P. (1975). Scripts, Plans, and Knowledge. In *Proceedings of the 4th international joint conference on Artificial intelligence-Volume 1*, pages 151–157. Morgan Kaufmann Publishers Inc.

Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., and Zhu, W. L. (2002). Open Mind Common Sense: Knowledge Acquisition from the General Public. In *On the move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pages 1223–1237. Springer.

Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., and Suleman, K. (2017). NewsQA: A Machine Comprehension Dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200.

Wanzare, L. D. A., Zarcone, A., Thater, S., and Pinkal, M. (2016). DeScript: A Crowdsourced Database for the Acquisition of High-quality Script Knowledge. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, may. European Language Resources Association (ELRA).

Wanzare, L., Zarcone, A., Thater, S., and Pinkal, M. (2017). Inducing Script Structure from Crowdsourced Event Descriptions via Semi-Supervised Clustering. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 1–11, Valencia, Spain, April. Association for Computational Linguistics.

Weston, J., Bordes, A., Chopra, S., Rush, A. M., van Merriënboer, B., Joulin, A., and Mikolov, T. (2015). Towards AI-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

A Neural Network Based Model for Loanword Identification in Uyghur

Chenggang Mi[†‡], Yating Yang[†‡], Lei Wang[†‡], Xi Zhou[†‡], Tonghai Jiang[†‡]
[†]The Xinjiang Technical Institute of Physics & Chemistry of Chinese Academy of Sciences, Urumqi, China
[‡]Key laboratory of speech language information processing of Xinjiang, Urumqi, China
{micg,yangyt,wanglei,zhouxi,jth}@ms.xjb.ac.cn

Abstract

Lexical borrowing happens in almost all languages. To obtain more bilingual knowledge from monolingual corpora, we propose a neural network based loanword identification model for Uyghur. We build our model on a bidirectional LSTM - CNN framework, which can capture past and future information effectively and learn both word level and character level features from training data automatically. To overcome data sparsity that exists in model training, we also suggest three additional features , such as hybrid language model feature, pronunciation similarity feature and part-of-speech tagging feature to further improve the performance of our proposed approach. We conduct experiments on Chinese, Arabic and Russian loanword detection in Uyghur. Experimental results show that our proposed method outperforms several baseline models.

Keywords: Loanword Identification, Uyghur Language, BiLSTM-CNN, Language Model, Pronunciation Similarity

1. Introduction

Lexical borrowing is very common between languages (Taylor and Grant, 2015). It is a phenomenon of cross-linguistic influence. If loanwords in resource-poor languages (e.g. Uyghur) can be identified effectively, we can use the donor-receipt word pairs to extend bilingual dictionary. And the bilingual dictionary plays a very important role in many cross-lingual areas in natural language processing (NLP), such as machine translation (Tsvetkov and Dyer, 2015).

In this paper, we describe a novel method to identify loanwords in Uyghur texts to alleviate the data sparsity that exists in Uyghur related NLP tasks. Our loanword identification model is based on a bidirectional long-short term memory (BiLSTM) - convolutional neural networks (CNNs) framework (Chiu and Nichols, 2016). The BiLSTM have achieved state-of-the-art performance in varies sequence-to-sequence learning tasks, a very important reason is that it can capture past (from previous tagged words) and future (from next untagged words) information effectively. We use it to model word level features. CNNs have been used in several character level natural language processing (NLP) tasks; we use it to model character level features. Therefore, our model can learn both word level and character level feature from training data, automatically. Additionally, we also propose three features (hybrid language model, pronunciation similarity and part-of-speech tagging) to argument the BiLSTM-CNN model by exploit knowledge learned from monolingual corpus. We conduct experiments on Chinese, Arabic and Russian loanwords detection in Uyghur, respectively. Experimental results show that our model outperforms several baseline models.

2. Loanwords in Uyghur Language

2.1. An Introduction of Uyghur Language

Uyghur is an official language of the Xinjiang Uyghur Autonomous Region in China, and is widely used in both social and official spheres, as well as in print, radio and television, and is mostly used as a lingua franca by other ethnic minorities in Xinjiang. Uyghur belongs to the Turkic language family. Like other Turkic languages, Uyghur displays vowel harmony and agglutination, lacks noun classes or grammatical gender, and is a left-branching language with subject (S) - object (O) - verb (V) word order.

As an agglutinative language, nouns in Uyghur are inflected for number and case, but not gender and definiteness like in many other languages(Table 1)[1]. There are two numbers (singular, plural) and six different cases in nouns of Uyghur language. Verbs are inflected for tense, voice, aspect and mood.

Uyghur(in English)	Uyghur(stem+suffix)
aliqanimda(In my hands)	**aliqan**+im+da
etrapidikilerni(People around)	**etrap**+i+diki+ler+ni
qurulmasining(Structured)	**qurulma**+si+ning
qalduridu(Stay)	**qal**+dur+i+d+u

Table 1: Examples of Uyghur word formation.

2.2. Linguistic Issues of Loanwords in Uyghur

Due to different kinds of language contact through the history, Uyghur has adopted many loanwords(Kamalov, 2006). Some studies show that larger than 20% of the vocabulary is from other languages. Kazakh, Uzbek, and Chagatai are all Turkic languages which had a strong influence on Uyghur. Arabic words have also entered Uyghur through Islamic literature after the introduction of the Islamic religion around the 10th century.

Recently, Chinese and Russian had the greatest influence on Uyghur language. In particular, loanwords from these two languages are all quite recent. Below are some examples of loanwords in Uyghur (Table 2):

3. Method

In this section, we describe the details of our proposed loanword identification model. First, we present the BiL-

[1]In this paper, we write Uyghur with the Latin alphabet.

Uyghur(Chinese)[In English]	Uyghur(Russian)[In English]
shinjang(新疆) [one province in China]	tElEfon(телефон) [telephone]
laza(辣子) [hot pepper]	uniwErsitEt(университет) [university]
shuji(书记) [secretary]	radiyo(радио) [radio]
koi(块) [Chinese currency]	pochta(почта) [post office]
lengpung(凉粉) [agar-agar jelly]	wElsipit(велосипед) [bicycle]
dufu(豆腐) [tofu]	oblast(область) [region]

Table 2: Examples of Chinese and Russian Loanwords in Uyghur.

STM model, which can model word-level features from both forward and backward directions. Next, we introduce the CNN based character-level feature extraction. Then, we detail two core features (word embedding feature and character embedding feature) and three additional features (pronunciation similarity feature, POS tagging feature and hybrid language model feature) used in our model. Finally, we present the training and optimization of our proposed neural network.

3.1. Word-level Tagging with BiLSTM

Similar to previous work used the BiLSTM in other areas in NLP (Such as NER and speech recognition), we explored a stacked bi-directional recurrent neural network with LSTM units to transform word features into loanword tag scores(Figure 1).

In our paper, we fed extracted features of each word into

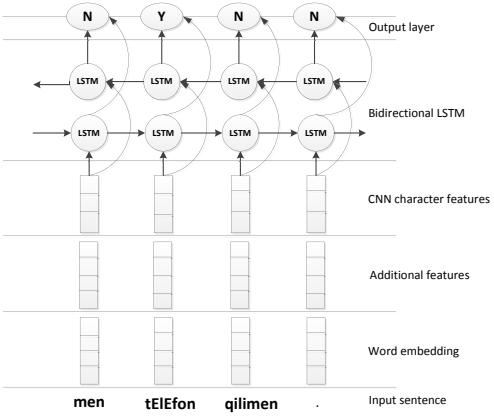


Figure 1: BiLSTM Model for Loanwords Identification in Uyghur.

a forward and backward LSTM network. The output of each network at each time step is decoded by a linear layer and a log-softmax layer into log-probabilities for each tag category. The two vectors produced by a linear and a log-softmax are added to produce the final output.

3.2. Character-level Features Extraction Based on CNN

Convolutional neural networks (CNN) have shown great success in character level features extraction in NER and POS tagging. However, the character level CNN has not been applied in loanwords identification. In this paper, we employ a convolution and a max layer to extract a new feature vector from the character embeddings(Figure 2).

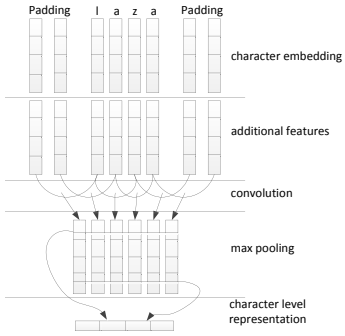


Figure 2: Character-level Features Extraction Model Based on CNN.

3.3. Features Definition

3.3.1. Main Features

Word Embedding Feature

Word embedding is a collective name for techniques in NLP where words or phrases from the vocabulary are mapped to vectors represent by real numbers. It aims to quantify and categorize semantic similarities between words based on their distributional properties in large samples of language data. In this study, we use the word embedding as input to the neural network.

Currently, there is no publicly available Uyghur word embeddings, so we trained them by ourselves. In this paper, we experimented with three sets of word embeddings: 1) NEWS word2vec embeddings trained on 2 billion words from news and government documents; 2) ORAL word2vec embeddings trained on 1.5 billion words from short messages and weinxin, 3) HYBRID word2vec embeddings trained on 3.5 billion words from both 1) and 2). We used the open source toolkit Glove[2] to train above word embeddings.

Character Embedding Feature

In this paper, the character embeddings are uniformly sampled from range $[-\sqrt{\frac{3}{DIM}}, \frac{3}{DIM}]$. DIM is the dimension of embeddings. The character set includes all unique characters in Uyghur language. Besides, there are two more tokens are also containing in above set: UNKNOWN and PADDING. The UNKNOWN is indicates all other characters and PADDING is used for CNN.

[2]https://nlp.stanford.edu/projects/glove/

3.3.2. Additional Features

Besides the main features described in previous section, we also proposed four features to further improve the performance of our approach. Intuitively, we may think that the loanword in Uyghur should have a similar pronunciation with its corresponding word in donor language. Therefore, we take the pronunciation similarity as an important feature in our model. Due to most loanwords are nouns, we constraint the output of our approach by a part-of-speech tag feature. A loanword may adapt the pronunciation system of receipt language when borrowing from the donor language. So the pronunciation of a loanword has the features of pronunciation systems of both receipt and donor languages. We use a hybrid language model to represent this feature. In this paper, we focus on Chinese, Arabic and Russian loanwords in Uyghur.

Pronunciation Similarity (ps)

Loanword (LW) usually have a similar pronunciation to their corresponding donor language (DW) word. Previous work detected loanwords in Uyghur according to string similarity between LW and DW, which first transfer the pronunciation similarity as string similarity. However, we found that there exist many differences among different writing systems, which have negative effects on the loanwords detection. To overcome this, we proposed a method that transfers both words into strings according to the International Phonetic Alphabet (IPA) . After that, we compute string similarity based on the Edit Distance algorithm (aka Levenshtein Distance).

Part-Of-Speech Tagging (pos)

Most loanwords are nouns. Therefore, we use part-of-speech information as one of the additional feature to constraint the output of our model. The POS tags are obtained by an in-house Uyghur POS tagger which was developed by us.

Hybrid Language Model (hlm)

Usually, there are different pronunciation systems between recipient language and donor language. Each pronunciation system can be represented by a certain character-level language model. When lexical borrowing, the pronunciation of a word in donor language may adapts the pronunciation system of the recipient language. This inspired us to combine these two language models to feature the pronunciation of loanwords.

$$p_{hlm} = (1 - \lambda_1 p_{uyg}) + \lambda_2 p_{dnr} \quad (1)$$

Where p_{uyg} is the language model probability of a given character sequence in Uyghur, p_{dnr} is the language model probability of above sequence in donor languages. λ_1 and λ_2 are weights which can be obtained during model optimization.

3.4. Neural Network Training

3.4.1. Tagging Scheme

Similar to the scheme used in named entities recognition (NER), we used the BIESO tag set in this paper. BIOES, which stand for Begin (B), Inside (I), End (E), Single (S) and Other (O), indicating the position of the character in a certain loanword. For example: With the BIESO tagging scheme, more information can be considered when neural networks training.

3.4.2. Network Implementation

We implement the BiLSTM-CNN model used in our paper based on Theano[3], a widely used deep learning Python library. We trained the loanwords identification model based on sentence-level corpus. We initialized the word embedding and character embedding as previous description. Other lookup tables used in our model are randomly initialized with values drawn from a standard normal distribution.

3.4.3. Parameters

We tune the hyper-parameters as follows: for CNN, we set the window size as 5 and use 40 filters; for bidirectional LSTM network, we set the initial state as 0.0 and state size as 300. As mentioned above, we use dropout to regularize our model to alleviate overfitting when neural network training and we set the dropout rate as 0.5.

3.4.4. Optimization Algorithm

We use the min-batch stochastic gradient descent (SGD) to train our loanwords identification model. Each minibath includes multiple sentences with the same number of Uyghur words. We find that training neural network with dropout is very effective in alleviate the overfitting. To achieve a better performance on development sets, we use early stopping method in our experiments.

4. Experiments

4.1. Datasets and Settings

To evaluate the proposed method effectively, we train the neural network with three groups of corpora (tokens)(UYGLWChn:20M/10K/20K*2 [training/develop/test], UYGLWArab:15M/10K/20K*2 and UYGLWRus:12M/10K/20K*2). Because there are relatively few loanwords (compared with other words) in Uyghur, we also annotated person names in these three donor languages as loanwords. These corpora are collected from government websites and newspapers. Test data for cross-domain experiments are selected from social medias such as Weixin and Twitter. We train three donor language models with corpus selected from previous corpora. The develop sets and test sets used in our experiments are all selected from the same domain.

We built the bi-directional LSTM-CNN neural network on the Theano library. The computations for a model are run on a GPU. We extract the word embedding and character embedding based on the open source toolkit Glove[4]. We use SRILM[5] to obtain the character level language model for four languages. The POS tagging features are extracted based on a Uyghur POS tagger, which was developed by us.We use Precision (P), Recall (R) and F1 score to evaluate the performance of loanword identification models.

[3] http://deeplearning.net/software/theano/

[4] https://nlp.stanford.edu/projects/glove/

[5] http://www.speech.sri.com/projects/srilm/

Model	Pchn	Rchn	F1chn	Prus	Rrus	F1rus	Parab	Rarab	F1arab
CRFs	69.78	62.33	66.35	71.64	63.25	67.18	72.50	65.32	68.72
SSIM	66.32	77.28	71.38	75.39	70.02	72.61	73.76	67.51	70.50
CIBM	78.82	68.30	73.18	81.03	73.22	76.93	75.22	70.71	72.90
RNN	78.97	79.20	79.08	82.55	75.93	79.10	83.26	77.58	80.32
Ours	**80.24**	**81.02**	**80.63**	**82.95**	**76.30**	**79.49**	**84.09**	**78.28**	**81.08**

Table 3: Experimental Results on Loanword Identification Models.

4.2. Experimental Results

4.2.1. Effects on Different Methods

Table 3 presents results on different methods, including CRFs-based model (CRFs), the string similarity based loanword identification model (SSIM) (Mi et al., 2013), identification model based on classification (CBIM) (Mi et al., 2014), a RNN based identification model (Mi et al., 2016) and the model proposed in this paper. Due to lack sufficient annotated corpus, CRFs based model cannot outperform other models. After analysis the output of CRFs based model, we found that only a small number of non-person name loanwords are identified, which means the CRFs based method rely on annotated corpus heavily. The performance of SSIM outperforms CRFs based model, an important reason is that the SSIM method based on pronunciation similarity between two words (donor words and receipt word). Because it combines the advantages of statistical based and rule based model, the CIBM outperforms CRFs and SSIM models. Like CRFs, RNN based model also rely on annotated corpus heavily. However, we found that the RNN based model can identify more loanwords (more than person names) than CRFs based model, a possible reason is that the RNN encoder-decoder framework can learn features automatically and use its internal memory to process arbitrary sequences of inputs. To model the word-level and character-level features, we proposed a BiLSTM-CNN neural network to identify loanwords from Uyghur texts. Moreover, three important additional features were also suggested to overcome data sparseness. Our proposed method achieved the best score among these approaches.

4.2.2. Experimental Results on Different Features

In Table 4, we present results with different additional features. We can found that our model (ours) achieves best performance among all these models. We observe that our model benefit most from hlm (**h**ybrid **l**anguage **m**odel) feature. Compare with other two models (BiLSTM-CNN+ps&pos), model with hlm feature achieve the best improvements. An important reason is that the hlm feature combines both pronunciation systems of donor language and receipt language. Models with string similarity and POS tag features only reflect the shallow semantic information. Due to lack of annotated corpus, the model without any additional features performs worst in all experiments.

4.2.3. Evaluation on Different Domains

Table 5 shows our results on different domains. To show the capability of our loanwords identification model, we evaluate our model on different domains, such as news(News) and social network(socialNet). We can found that the experimental results on formal corpus (news domain) which have the same domain with our training corpus are outperforming the performance on informal domain (social network). We observed the same situation in all donor languages. We also found that experimental results on informal domain are just a little worse compared with results on formal domain. One possible reason is that our BiLSTM+CNN model can learn representation of knowledge beyond given training examples.

5. Related work

In general, word borrowing is often concerned by linguists (Chen, 2011)(Chen and Chen, 2011). There are relatively few studies about loanwords in NLP area.(Tsvetkov et al., 2015) and (Tsvetkov and Dyer, 2016) proposed a morphphonological transformation model, features used in this model are based on optimality theory; experiment has been proved that with a few training examples, this model can obtain good performance at predicting donor forms from borrowed forms.(Tsvetkov and Dyer, 2015) suggest an approach that uses the lexical borrowing as a model in SMT framework to translate OOV words in a low-resource language. For loanwords detection in Uyghur, string similarity based methods were often used at the early stage(Mi et al., 2014).(Mi et al., 2016) propose a loanword detection method based on the perceptron model, several features are used in model training.

6. Conclusion

In this paper, we have presented a novel model to detect loanwords (mainly Chinese, Arabic and Russian loanwords) in Uyghur by using a BiLSTM-CNN framework. Except two main features such word embeddings and character embeddings, two additional features like donor language model feature and hybrid language model feature are also proposed and integrated the framework to further improve the performance. In the proposed model, the character-level feature of each word is extracted by the CNN model based on character embedding and our proposed two features. For each word, the character-level feature vector is concatenated with the word embedding feature vector and fed into the BiLSTM model. After that, these feature vectors are fed to output layers. Experiment results on loanwords identification in Uyghur have presented that the proposed model can significantly improve the identification performance.

Although our model achieves the best results on loanwords identification in Uyghur, we only use loanword taggers in our training set. In our future work, we plan to integrate

Feature	Pchn	Rchn	F1chn	Prus	Rrus	F1rus	Parab	Rarab	F1arab
BiLSTM-CNN	77.65	67.89	72.44	78.02	68.33	72.85	78.38	70.96	74.49
BiLSTM-CNN+ps	78.86	70.32	74.35	81.94	70.65	75.88	81.12	71.52	76.02
BiLSTM-CNN+pos	78.79	69.54	73.88	81.35	71.28	75.98	80.76	70.20	75.11
BiLSTM-CNN+hlm	79.42	70.37	74.62	82.29	73.50	77.65	82.14	73.59	77.63
BiLSTM-CNN+all	**80.24**	**81.02**	**80.63**	**82.95**	**76.30**	**79.49**	**84.09**	**78.28**	**81.08**

Table 4: Evaluation on Features Used in RNN-based Model.

Domain	Pchn	Rchn	F1chn	Prus	Rrus	F1rus	Parab	Rarab	F1arab
socialNet	79.63	80.51	80.07	81.05	75.22	78.03	83.63	77.45	80.42
News	**80.24**	**81.02**	**80.63**	**82.95**	**76.30**	**79.49**	**84.09**	**78.28**	**81.08**

Table 5: Evaluation on Cross-Domain Corpora.

more linguistic knowledge to further optimize the performance of our proposed model.

7. Acknowledgements

We sincerely thank the anonymous reviewers for their thorough reviewing and valuable suggestions. This work is supported by the West Light Foundation of The Chinese Academy of Sciences under Grant No.2015-XBQN-B-10, the Xinjiang Key Laboratory Fund under Grant No.2015KL031, the Xinjiang Science and Technology Major Project under Grant No.2016A03007-3 and the Natural Science Foundation of Xinjiang under Grant No.2015211B034.c

8. Bibliographical References

Chen, Y. and Chen, P. (2011). A comparison on the methods of uyghur and chinese loan words. *Journal of Kashgar Teachers College*, 32(2):51–55.

Chen, S. (2011). New research on chinese loanwords in the uyghur language. *N.W.Journal of Ethnology*, 28(1):176–180.

Chiu, J. and Nichols, E. (2016). Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Kamalov, A. (2006). *Uyghur Studies in Central Asia: A Historical Review*.

Mi, C., Yang, Y., Zhou, X., Li, X., and Yang, M. (2013). Recognition of chinese loan words in uyghur based on string similarity. *Journal of Chinese Information Processing*, 27(5):173–179.

Mi, C., Yang, Y., Wang, L., Li, X., and Dalielihan, K. (2014). Detection of loan words in uyghur texts. In *Natural Language Processing and Chinese Computing: Third CCF Conference, NLPCC 2014, Shenzhen, China, December 5-9, 2014. Proceedings*, pages 103–112, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mi, C., Yang, Y., Zhou, X., Wang, L., Li, X., and Jiang, T. (2016). Recurrent neural network based loanwords identification in uyghur. In *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers*, pages 209–217.

Taylor, J. R. and Grant, A. P. (2015). Lexical borrowing.

Tsvetkov, Y. and Dyer, C. (2015). Lexicon stratification for translating out-of-vocabulary words. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 125–131. Association for Computational Linguistics.

Tsvetkov, Y. and Dyer, C. (2016). Cross-lingual bridges with models of lexical borrowing. *Journal of Artificial Intelligence Research*, 55(1):63–93, January.

Tsvetkov, Y., Ammar, W., and Dyer, C. (2015). Constraint-based models of lexical borrowing. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 598–608. Association for Computational Linguistics.

Revisiting Distant Supervision for Relation Extraction

Tingsong Jiang[†], **Jing Liu**[‡], **Chin-Yew Lin**[§], **Zhifang Sui**[†]
†Key Laboratory of Computational Linguistics, Ministry of Education
School of Electronics Engineering and Computer Science, Peking University
‡Baidu Inc., Beijing, China
§Microsoft Research
tingsong@pku.edu.cn, liujing46@baidu.com, cyl@microsoft.com,szf@pku.edu.cn

Abstract

Distant supervision has been widely used in the task of relation extraction (RE). However, when we carefully examine the experimental settings of previous work, we find two issues: (i) The compared models were trained on different training datasets. (ii) The existing testing data contains noise and bias issues. These issues may affect the conclusions in previous work. In this paper, our primary aim is to re-examine the distant supervision-based approaches under the experimental settings without the above issues. We approach this by training models on the same dataset and creating a new testing dataset annotated by the workers on Amzaon Mechanical Turk. We draw new conclusions based on the new testing dataset. The new testing data can be obtained from *http://aka.ms/relationie*.
Keywords: relation extraction, distant supervision

1. Introduction

In recent years, knowledge bases (KBs) like Freebase (Bollacker et al., 2008), DBpedia (Lehmann et al., 2015) and NELL (Carlson et al., 2010) have become extremely useful resources for many natural language processing (NLP) tasks. These KBs are mostly composed of relational facts between entities, which are typically represented as triples with the format (head entity, relation, tail entity), e.g., (`Paris`, `capitalOf`, `France`). Although existing KBs may contain billions of relational facts, they are still far from complete and missing many crucial facts. To enrich KBs, relation extraction (RE), *i.e.*, the task of extracting relations between entities from plain texts, has thus attracted increasing attention. For example, here is a sentence: `Paris is the capital city of France`.
where `Paris` and `France` are two entity mentions. A relation extractor or classifier takes the sentence and the two entity mentions as inputs, and determines the semantic relation that it expresses, if any. In the above example, a correct prediction may be `capitalOf` relation.

Most existing approaches formulate RE as a classification task and use supervised learning on relation-specific training data, which is very expensive to acquire. To address this issue, distant supervision is proposed to automatically generate training data via aligning facts in KBs and texts (Wu and Weld, 2007; Mintz et al., 2009). The *distant supervision assumption* is that if two entities preserve a relation in a KB, then *all sentences* that mention the two entities express the relation. Figure 1 shows an example of the automatic labeling of data via distant supervision. In this example, `Paris` and `France` are two entities with a relation type `capitalOf` in a KB. All sentences contain these two entities are labeled with `capitalOf`. Although distant supervision provides a cheap way to automatically label training data, it leads to a noise problem with the data. The noisy data can be mainly classified into two categories:

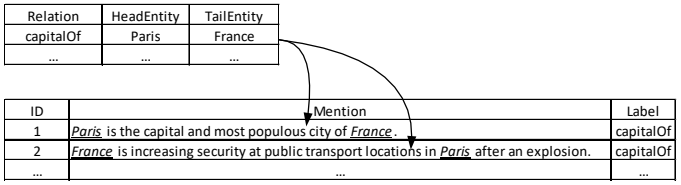


Figure 1: Training instances generated via distant supervision. The first sentence has a correct label, but the second sentence has a wrong label.

(i) False positive instances. Not necessarily all sentences that mention an entity pair express the target relation. As shown in Figure 1, the second sentence is a false positive instance. (ii) Multiple labels instances. An entity pair may preserve multiple relation types in a KB. For example, (`Bill Gates`, `founderOf`, `Microsoft`) and (`Bill Gates`, `ceoOf`, `Microsoft`) are clearly true.

To deal with the two major issues, multi-instance multi-label learning (MIML) was proposed for RE by relaxing the distant supervision assumption and making the *at-least-one assumption*: if two entities preserve a relation in a KB, *at least one sentence* that mentions the entity pair expresses the relation (Riedel et al., 2010; Hoffmann et al., 2011; Surdeanu et al., 2012). The previous work of MIML-based approaches can be mainly categorized into two folds: (i) **feature-based approaches** (Hoffmann et al., 2011; Surdeanu et al., 2012) and (ii) **neural network-based approaches** (Zeng et al., 2015; Lin et al., 2016).

However, when we carefully examine the experimental settings of the previous MIML-based work (Hoffmann et al., 2011; Zeng et al., 2015; Lin et al., 2016), we find the following issues which may affect the existing conclusions: (i) When the model comparison experiments were conducted by Zeng et al. (2015; Lin et al. (2016), the compared models were trained on the datasets with different size. Particularly, the neural network-based models (Lin et al., 2016) actually used a large training dataset containing $522,611$ sentences, while feature-based models (Hoffmann et al., 2011)

This work was done when Tingsong Jiang and Jing Liu were working at Microsoft Research.

were trained on a small dataset containing $126,184$ sentences. We will give the details of the two training datasets in Section 2.. It is important to re-examine the performances of the models that are trained on the same training dataset. The new experimental results will be shown in Section 3.4.. (ii) Most MIML-based approaches (Hoffmann et al., 2011; Zeng et al., 2015; Lin et al., 2016) were evaluated on the testing data generated by distant supervision. However, the automatically generated labels in the testing data could be wrong due to the limitation of distant supervision assumption. The quality of the testing data may affect the experimental results. Although Hoffmann et al. (2011) released a testing dataset which was manually annotated, the dataset was sampled from the union of the extraction results by the model of (Hoffmann et al., 2011) and the data generated by distant supervision. If the experiments are conducted on this testing data, the results may be biased towards to the model of (Hoffmann et al., 2011). To address these issues, similar to the slot filling task of TAC KBP, we develop a new testing dataset which is sampled from a set by pooling the extraction results from all compared models and the data generated by distant supervision. We will show our new observations based on the new testing data in Section 3.4..

The above issues may affect the conclusions in previous work. In this paper, we revisit the distant supervision for relation extraction. Specifically, our contributions include:

- We carefully re-examine the experimental settings of previous MIML-based work for RE. We find the issues with the training data size and the testing data used in the experiments.
- We create a new testing dataset by pooling the extraction results from all compared models (Hoffmann et al., 2011; Zeng et al., 2015; Lin et al., 2016) and the data generated by distant supervision on NYT corpus, and using Amazon Mechanical Turk[1] (MTurk) to annotate the data in a crowdsourcing way.
- We conduct extensive experiments to examine the MIML-based approaches (Hoffmann et al., 2011; Zeng et al., 2015; Lin et al., 2016). All models are trained on the same training dataset and evaluated on our new testing dataset. We draw new conclusions based on the new testing dataset.

2. Datasets

As we discussed in Section 1., we find two issues of previous MIML-based approaches to relation extraction: (i) The compared models were trained on different training datasets. (ii) The existing testing data contains noise and bias issues. In this section, we will give the details of the datasets used in previous work, and create a new testing dataset.

Data Source. Most previous work uses New York Times (NYT) dataset [2] developed by (Riedel et al., 2010)[3]. The NYT corpus contains about 1.8 million news articles.

Dataset	#sentences	#pairs	#facts
DSTrainSmall	126,184	67,946	4,700
DSTrainLarge	522,611	279,786	18,252
DSTest	172,448	96,678	1,950
HoffmannTest	881	565	259
OurTest	2,040	1,666	547

Table 1: Statistics about the datasets.

When constructing the dataset, named entity mentions were first extracted from the text of NYT articles by using Stanford Named Entity Tagger (Finkel et al., 2005). Then, the named entity mentions were linked to the entities in Freebase by using exact string matching. If a sentence mentions two entities that have a relation in Freebase, then a corresponding instance will be generated and labeled as the relation type. Otherwise, an instance with a label *NA* which indicates that there is no relation between the entity pair, will be generated. Riedel et al. (2010) mainly focus on the relations related to "people", "business", "person" and "location". There are 53 relation labels including the special label *NA* in the corpus. The Freebase relations were divided into two parts, one for training and one for testing. The former is aligned to the $2005-2006$ articles of NYT corpus, and the latter to the 2007 articles.

Training Data. In previous work, there are two training datasets sampled from the aligned sentences of $2005-2006$ NYT articles. (i) Riedel et al. (2010) sampled a small training dataset containing $126,184$ sentences, $67,946$ entity pairs and $4,700$ facts. We denote this dataset as *DSTrainSmall*. (ii) Zeng et al. (2015; Lin et al. (2016) sampled a large training dataset containing $522,611$ sentences, $279,786$ entity pairs and $18,252$ facts which covers all sentences in *DSTrainSmall*. We denote this dataset as *DSTrainLarge*. The details of these two training datasets have been given in Table 1. In the experiments of Zeng et al. (2015; Lin et al. (2016), the neural network-based models were trained on *DSTrainLarge*, while the feature-based models were trained on *DSTrainSmall*. The comparison might be not fair. We will train and compare these models on the two training datasets respectively.

The Existing Testing Data. In previous work, there are two popular testing datasets. (i) One is the dataset generated by distant supervision. We denote this dataset as *DSTest*. However, as we discussed previously, the automatically generated labels in the testing data could be wrong due to the limitation of distant supervision assumption. The quality of the testing data may affect the experimental results. (ii) Although Hoffmann et al. (2011) released a testing dataset which was manually annotated, the dataset was sampled from the union of the extraction results from MultiR (Hoffmann et al., 2011) and the data generated from distant supervision. We denote this dataset as *HoffmannTest*. The evaluation conducted on this dataset may be biased to-

[1] https://https://www.mturk.com/

[2] http://iesl.cs.umass.edu/riedel/ecml/

[3] Apart from the NYT dataset released by (Riedel et al., 2010), Surdeanu et al. (2012) has released a KBP dataset with a manually labeled testing data. Unfortunately, KBP dataset only contains feature information for each sentence while lack of original plain texts. Neural network-based approaches cannot be compared on the KBP dataset due to lack of plain texts.

Window size	Word dim.	Position dim.	Batch size	Learning rate	Dropout prob.	Sentence dim.
l=3	d_w=50	d_p=5	160	0.001	0.5	d_c=230

Table 2: The parameters of neural networks-based approaches used in our experiments.

wards MultiR.

A New Testing Data. Since the existing testing datasets have noise and bias issues, we develop a new testing dataset. Our aim is to guarantee the quality of the data and make it not biased towards any of the compared models. Because the instances with non-"NA" labels are quite sparse in the data, it is difficult for us to directly sample enough non-"NA" instances from the corpus to annotate. Similar to the slot filling task of TAC KBP, the key idea of creating the dataset is pooling the top predicted results from all compared models and the data generated from distant supervision.

In the testing corpus *DSTest*, distant supervision labels $172,448$ sentences and only $6,444$ sentences are labeled as non-"NA". We first pool the $6,444$ non-"NA" sentences given by distant supervision and the top $10,000$ non-"NA" sentences predicted by each of the compared systems (including MultiR, CNNONE, PCNNONE, CNNATT and PCNNATT that will be described in Section 3.1.). The pooling results contain $17,147$ sentences. Then we randomly sample $2,040$ sentences from the pooling results, and utilize Amazon Mechanical Turk to annotate the dataset in a crowdsourcing way. We divide the $2,040$ sentences into 120 tasks, and each task contains 17 sentences to be labeled and 3 controls. Each control is a sentence that we already know its label. All the controlled sentences are sampled from the set of *HoffmannTest* (Hoffmann et al., 2011). Since it is important to control the annotation quality, we use the controls to detect the unqualified workers. If a worker fails on more than one control in a task, we will discard all his annotations for the task. Then the task will be automatically re-assigned to a new worker to complete. Besides, we request 5 workers to annotate each sentence, and use the majority votes to get the ground truth label. The agreements between workers are high. There are 99.7% sentences to which $3/5$ or more workers give the same label. If there is a tie, we will ask another annotator to break it. There are only 6 tie sentences. The details of the three testing datasets have been shown in Table 1.

3. Experiments

In this section, we will examine that how the two data issues will affect the conclusions in previous work, and we will give our new observations based on the new testing dataset.

3.1. Systems

In our experiments, we revisit and compare the following feature-based and neural network-based MIML systems:

- **MultiR** (Hoffmann et al., 2011) which is a feature-based MIML approach.
- **CNNONE** (Zeng et al., 2015) which is a convolutional neural networks (CNN) based MIML model. ONE means that only one sentence is active in each bag (for one entity pair).
- **PCNNONE** (Zeng et al., 2015) which is a picewise convolutional neural networks (PCNN) based MIML model.
- **CNNATT** (Lin et al., 2016) which extends the model of CNNONE by introducing sentence-level attention over multiple instances.
- **PCNNATT** (Lin et al., 2016) which extends the model of PCNNONE by introducing sentence-level attention over multiple instances.

We use the implementations of these systems shared by the authors (Hoffmann et al., 2011; Lin et al., 2016)[4].

3.2. Parameters Settings

In this section, we describe the parameters settings of the neural network-based approaches.

Word Embeddings. In this paper, we follow Lin et al. (2016) and use the word2vec tool[5] to train the word embeddings to on NYT corpus. We keep the words which appear more than 100 times in the corpus as vocabulary. Besides, when training the word embeddings, an entity mention will be considered as one token if it has multiple words.

Model Parameters. Following (Surdeanu et al., 2012), we use three-fold validation on the training set to tune the parameters of all models. We use grid search to determine the optimal parameters and manually specify spaces of the following parameters: the sliding window size $l \in \{1,2,...,8\}$, the size of sentence embedding $n \in \{50,60,..., 300\}$ and the batch size B among $\{40, 160, 640, 1280\}$. For other parameters, we follow the settings used in (Zeng et al., 2015; Lin et al., 2016). Table 2 summarizes the parameters of neural networks-based approaches used in our experiments.

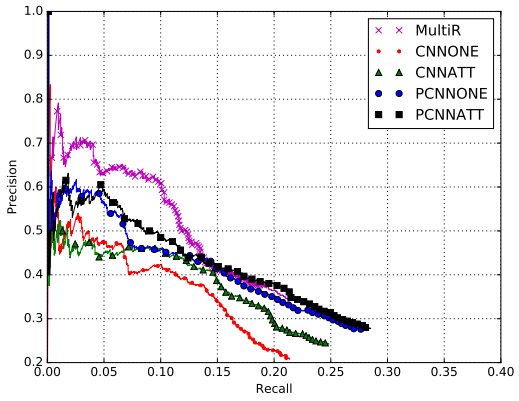
3.3. Evaluation Metrics

In the experiments, we compare the precision and recall curve of each system. The curve of each system is drawn by (i) ranking all predicted instances according to their confidence scores given by the system, and (ii) traversing the ranking list from the high score to low score to measure the precision and recall at each position.

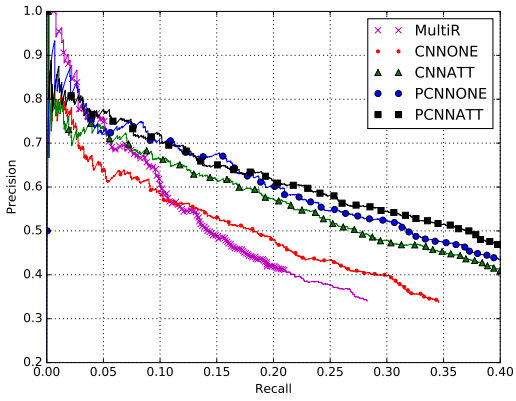
Additionally, in previous work, the evaluation were usually conducted in two levels: **entity pair level** and **sentence level**. By entity pair level, we mean that the system should determine the relation of one bag (i.e., a set of sentences that mention the same entity pair). When using the testing data generated by distant supervision (*DSTest*), we use entity pair level evaluation. Because *DSTest* has less noise at

[4]http://www.cs.washington.edu/ai/raphaelh/mr and https://github.com/thunlp/NRE

[5]https://code.google.com/p/word2vec/

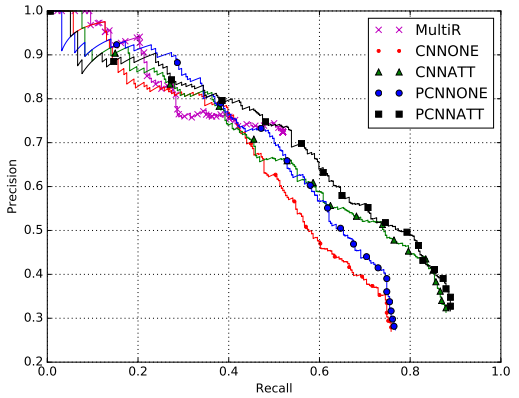


(a) The results of models trained on *DSTrainSmall*.

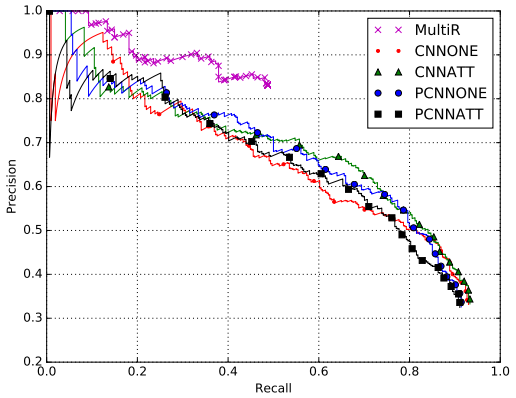


(b) The results of models trained on *DSTrainLarge*.

Figure 2: The experimental results on the testing data generated by distant supervision (i.e. *DSTest*).

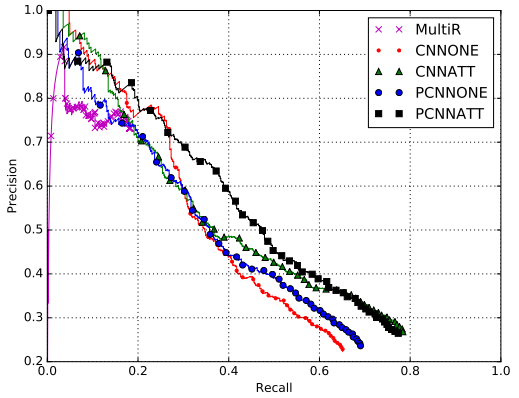


(a) The results of models trained on *DSTrainSmall*.

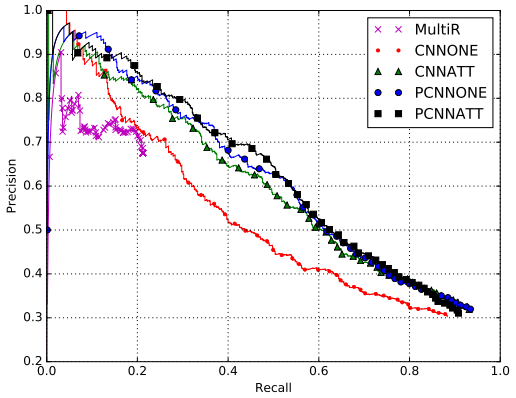


(b) The results of models trained on *DSTrainLarge*.

Figure 3: The experimental results on the manual testing data created by Hoffmann (i.e. *HoffmannTest*).



(a) The results of models trained on *DSTrainSmall*.



(b) The results of models trained on *DSTrainLarge*.

Figure 4: The experimental results on the manual testing data created in this paper (i.e. *Ours*).

entity pair level while more noise at sentence level, it is better to use the bag level label under the at-least-one assumption. By sentence level, we mean that the system should determine the relation of one instance (i.e., a sentence that mentions an entity pair). When using the testing data that contains the manually labeled sentences (*HoffmannTest* and *Ours*), we use sentence level evaluation.

3.4. Experimental Results

As we discussed in the Section 1., there are two issues with the experimental settings of previous work (Zeng et al., 2015; Lin et al., 2016): (i) the compared models were trained on the data with different size. (ii) the quality of the existing testing data is not good. In this paper, we conduct three experiments. In the first two experiments, we try to examine that how the two issues may affect the conclusions in previous work (Zeng et al., 2015; Lin et al., 2016). The third experiment is conducted on our new testing dataset, and we will give our new observations based on the results.

Experiment 1. In the experiments of (Zeng et al., 2015; Lin et al., 2016), the main evaluations were conducted on the testing data of *DSTest*. When Zeng et al. (2015; Lin et al. (2016) compare different models, the feature-based model (MultiR) was trained on the small training dataset *DSTrainSmall*, while the neural network-based approaches (including CNNONE, PCNNONE, CNNATT and PCNNATT) were trained on the large training dataset *DSTrainLarge*. Their major conclusion is that neural network-based approaches significantly outperform the feature-based approaches. However, it might be not fair to compare these models that were trained on the datasets with different size. In Experiment 1, we train all models on two training datasets (*DSTrainSmall, DSTrainLarge*) respectively and compare their performance on testing dataset *DSTest*. Figure 2 shows the experimental results. We can observe that (i) In Figure 2a, the feature-based approach MultiR outperforms the neural network-based approaches, when all models are trained on the small training data *DSTrainSmall*. (ii) In Figure 2b, when all models are trained on *DSTrainLarge*, MultiR outperforms the neural network-based approaches at the low recall positions. While MultiR performs worse at the high recall positions. (iii) All models benefit from enlarging the training data.

Experiment 2. Since there is a noise problem with the testing dataset *DSTest*, we further conduct the evaluation based on the testing data *HoffmannTest* which was manually annotated by Hoffmann et al. (2011). Figure 3 shows the experimental results. From Figure 3, we can observe that (i) MultiR is comparable to the the neural network-based approaches, when all models are trained on *DSTrainSmall*. (ii) MultiR significantly outperforms the neural network-based approaches when all models are trained on *DSTrainLarge*. (iii) Only MultiR benefits from enlarging the training data. The reason might be that the sampling strategy of the testing data *HoffmanTest* is biased towards MultiR.

Experiment 3. In this experiment, we conduct the evaluation on our new manual testing dataset, which tries to avoid the bias issue. Figure 4 shows the experimental results. From Figure 4, we have the following observations:

- Comparing to Figure 4a and Figure 4b, neural network-based approaches benefit more when the size of training data increases. The gap between MultiR and neural network-based approaches becomes larger when increasing the training data.

- According to Figure 4b, neural network-based approaches outperforms the feature-based approaches, but the gap is much smaller as compared to the observations in previous work. In the experimental results of (Lin et al., 2016), the precision gap between MultiR and PCNNATT is more than 0.3 given the recall 0.2. In contract, the precision gap is around 0.1 at the same recall position in our experiments according to Figure 4b.

- Lin et al. (2016) concludes that sentence-level attention brings performance gains for both CNN and PCNN. However, according to Figure 4b, sentence-level attention brings significant gains for CNN only. We cannot observe significantly improvements on PCNN.

4. Conclusions

In this paper, we carefully re-examine the experimental settings of previous work, and we find two issues: (i) the compared models were trained on the data with different size. (ii) the quality of the existing testing data is not good. We conduct experiments by training models on the same dataset and creating a new manual testing dataset annotated by the workers on Amzaon Mechanical Turk. Our major new observations include: (i) Neural network-based approaches benefit more when the size of training data increases. (ii) The performance gap between feature-based approaches and neural network-based approaches is much smaller as compared to the observations in previous work. (iii) Sentence-level attention brings significant improvement for CNN but not for PCNN. We also share the new testing data with the research community.

5. Acknowledgments

We thank the anonymous reviewers for their valuable comments. The research work is supported by the National Key Research and Development Program of China under Grant No. 2017YFB1002101 and the National Science Foundation of China under Grant No. 61772040. The contact author is Zhifang Sui.

6. Bibliographical References

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of SIGMOD*, pages 1247–1250. AcM.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E. R., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Proceedings of AAAI*, page 3.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language*

Technologies-Volume 1, pages 541–550. Association for Computational Linguistics.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Lin, Y., Shen, S., Liu, Z., Luan, H., and Sun, M. (2016). Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2124–2133.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Riedel, S., Yao, L., and McCallum, A. (2010). Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Surdeanu, M., Tibshirani, J., Nallapati, R., and Manning, C. D. (2012). Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics.

Wu, F. and Weld, D. S. (2007). Autonomously semantifying wikipedia. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50. ACM.

Zeng, D., Liu, K., Chen, Y., and Zhao, J. (2015). Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP), Lisbon, Portugal*, pages 17–21.

Incorporating Contextual Information for Language-Independent, Dynamic Disambiguation Tasks

Tobias Staron, Özge Alaçam, Wolfgang Menzel
Department of Informatics
University of Hamburg
{staron, alacam, menzel}@informatik.uni-hamburg.de

Abstract

Humans resolve various kinds of linguistic ambiguities by exploiting available external evidence that has been acquired from modalities besides the linguistic one. This behavior can be observed for several languages, for English or German for example. In contrast, most natural language processing systems, parsers for example, rely on linguistic information only without taking further knowledge into account. While those systems are expected to correctly handle syntactically unambiguous cases, they cannot resolve syntactic ambiguities reliably. This paper hypothesizes that parsers would be able to find non-canonical interpretations of ambiguous sentences, if they exploited external, contextual information. The proposed multi-modal system, which combines data-driven and grammar-based approaches, confirmed this hypothesis in experiments on syntactically ambiguous sentences. This work focuses on the scarcely investigated relative clause attachment ambiguity instead of prepositional phrase attachment ambiguities, which are already well known in the literature. Experiments were conducted for English, German and Turkish and dynamic, i. e. highly dissimilar, contexts.

Keywords: disambiguation, context, language-independence

1. Introduction

In psycholinguistics, there is substantial empirical evidence suggesting that human language processing successfully integrates available information acquired from different modalities in order to resolve fully as well as temporally ambiguous linguistic input, e. g. on the syntactic level, and predict what will be revealed next in the unfolding sentence (Tanenhaus et al., 1995; Altmann and Kamide, 1999; Knoeferle, 2005). During spoken communication, disambiguation and prediction processes allow for more accurate understanding. In contrast, natural language processing (NLP) systems are still not able to achieve that level of accuracy concerning challenging linguistic situations.

For example, a parser that processes linguistic information is expected to successfully handle syntactically unambiguous sentences by applying knowledge derived from training data or linguistic rules. However, neither parsers nor humans can resolve syntactic ambiguities without additional information. They may only have preferences. But, humans will use external information from other modalities for disambiguation successfully if it becomes available. Therefore, we expect to resolve syntactic ambiguities via multi-modal disambiguation by exploiting external knowledge, i. e. contextual information, that is derived from another modality, e. g. from visual scenes of the described events. This work proposes two hypotheses.

H1) A parser will resolve linguistic ambiguities reliably and reach correct interpretations if contextual information derived from additional modalities besides the linguistic one, i. e. visual scenes, are exploited.

H2) In addition, this behavior is expected to be observed independent of the language.

The contributions of this work are twofold. First, a language independent, data-driven parser has been modified to employ a grammar-based approach that will incorporate the contextual information even if it is previously unseen. Secondly, that system is used to validate the hypotheses for multiple languages: English, German and Turkish.

One of the most frequently investigated cases of syntactic ambiguity are prepositional phrase (PP) attachment ambiguities, where different semantic and syntactic interpretations are possible depending on assigning different thematic roles (Tanenhaus et al., 1995). The example *"the woman shoots the man with the pistol"* can be interpreted in different ways. Either, the woman is using the pistol to shoot the man or the man is holding the pistol. Instead, this work investigates the attachment ambiguity concerning relative clauses (Alaçam et al., 2018). In both cases, a multi-modal setting where the visual information constrains the referential choices helps the disambiguation process.

This paper is structured as follows. Section 2. describes the multi-modal data-set that contains syntactically ambiguous sentences and respective, disambiguating contextual information and that has been used to validate our hypotheses. Section 3. proposes a multi-modal disambiguation scheme, which has been used for the experiments, the results of which are shown in Section 4. and analyzed in Section 5.. Next, Section 6. describes related work followed by the discussion of the results.

2. Multi-Modal Data-Set

There are only few data-sets available that address complex linguistic ambiguities. The corpus of language and vision ambiguities (LAVA) (Berzak et al., 2016) contains 237 ambiguous sentences for English, which can only be disambiguated using respective external knowledge provided as short videos or static visual images with real world complexity. The LAVA corpus addresses a wide range of syntactic ambiguities including prepositional as well as verb phrase attachments and ambiguous interpretations of con-

Language	Voice	Exemplary Sentence	PoS Template
English	active	The woman carves the head of the bed, which the man paints.	$NP1_{nom}$ VP1 $NP2_{acc}$ PP1, WDT_{acc} $NP3_{nom}$ VP2.
	passive	The woman carves the head of the bed, which is painted by the man.	$NP1_{nom}$ VP1 $NP2_{acc}$ PP1, WDT_{nom} VP2 PP2.
German	active	Die Frau schnitzt das Kopfende des Bettes, das der Mann bemalt.	$NP1_{nom}$ VP1 $NP2_{acc}$ $NP3_{gen}$, WDT_{acc} $NP4_{nom}$ VP2.
	passive	Die Frau schnitzt das Kopfende des Bettes, das von dem Mann bemalt wird.	$NP1_{nom}$ VP1 $NP2_{acc}$ $NP3_{gen}$, WDT_{nom} PP1 VP2.
Turkish	active	Kadın adamın boyadığı yatağın başını oyuyor.	$NP1_{nom}$ $NP2_{gen}$ VP1(verb+adj/relativiser) $NP3_{gen}$ $NP4_{acc}$ VP2.
	passive	Kadın adam tarafından boyanan yatağın başını oyuyor.	$NP1_{nom}$ $NP2_{ablative}$ VP1(verb+verb+adj/relativiser) $NP3_{gen}$ $NP4_{acc}$ VP2.

Table 1: Exemplary sentence for ambiguity **A1)** for different languages in active as well as passive voice, including the respective part-of-speech (PoS) templates.

junctions. However, it does not take relative clause attachment ambiguities, which we are concerned with, into account. Also, it addresses only English. For human language processing, a recent study on the resolution of relative clause attachment ambiguities for different languages, e. g. English and German, can be found in (Hemforth et al., 2015), but, in general, the reference resolution in this case and the effect of its complexity in visually disambiguated situations addressing various languages have been scarcely investigated although ambiguities concerning relative clause attachments are quite common.

In the Hamburg Dependency Treebank (HDT) (Foth et al., 2014) part A, which contains $\approx$ 100k German sentences that were collected from the news website heise online, there are $13,256$ relative clauses that contain a relative pronoun that is supposed to have an antecedent and its reference resolution is ambiguous in $2,418$ cases (18.24%). While the nearest attachment has been chosen $1,907$ times (78.87%) in the HDT, an alternative respectively farther attachment occurs in 511 cases (21.13%). Therefore, we created a multi-modal data-set addressing these kinds of ambiguities among other things: the Linguistic Ambiguities in Situated Contexts (LASC) data-set (Alaçam et al., 2017; Alaçam et al., 2018). It contains challenging linguistic cases including ambiguous relative clause attachments and scope ambiguities for conjunctions as well as negations, which become fully unambiguous in the presence of visual stimuli. This work focuses on the relative clause attachment ambiguities. The multi-modal data, i. e. the syntactically ambiguous sentences and the corresponding scenes, are discussed in this section.

2.1. Linguistic Input

The LASC data-set (Alaçam et al., 2018) provides three types of fully ambiguous relative clause attachments, which are listed below.

A1) RPA[1] **- a Genitive Modifier** ***(English, German, Turkish - active & passive voice)***
The woman carves the head of the bed, which the man paints.
Int. 1[2]*:* The man paints the bed. *(low attachment)*
Int. 2: The man paints the head of the bed. *(high attachment)*

A2) RPA - a Prepositional Phrase ***(English, German)***
It is a mug on a coffee table, which she damages carelessly.
Int. 1: She damages the coffee table. *(low attachment)*
Int. 2: She damages the mug. *(high attachment)*

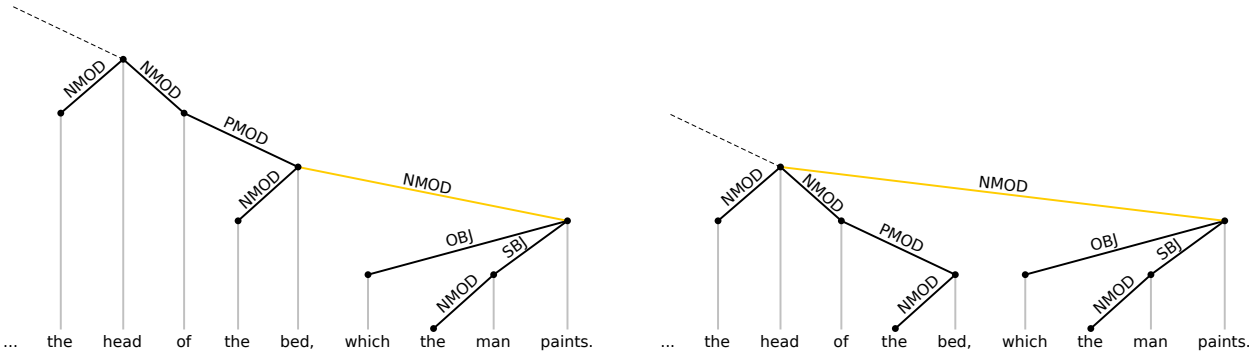
A3) RPA - Scope Ambiguities ***(English, German)***
I see apples and bananas, which lie on the table.
Int. 1: Only bananas lie on the table. (low attachment)
Int. 2: Both apples and bananas lie on the table. (high attachment)

In ambiguity **A1)** and **A2)**, the relative clause is either attached to a preceding nominal phrase (NP) (high attachment) or to a genitive modifier respectively a PP of that NP (low attachment). In **A3)**, the relative clause either refers to the preceding conjunction of two NPs (high attachment) or to the latter NP only (low attachment). Syntactically, both the low, i. e. the nearest, and high, i. e. the farther, attachment are always possible in all examples. Hence, all sentences of this LASC subset are ambiguous. The contrary interpretations are equally distributed in this subset. Figure 1 shows the two plausible interpretations of the exemplary sentence of **A1)** by depicting parts of the respective dependency trees. In Figure 1a, the relative clause is low-attached to *bed* while it is high-attached to *head* in Figure 1b.

All examples are provided in English as well as German and with the relative clause in active voice. **A1)** is also available in Turkish and with passive instead of active voice. Table 1 shows the different configurations for the example from ambiguity **A1)** and, in addition, the part-of-speech (PoS) templates that are used to generate the sentences. There are corresponding templates for each type of ambiguity for each language respectively voice. The number of sentences per test set, depending on ambiguity type, language, voice and target interpretation, can be found in Table 2. Overall, this LASC subset contains 458 examples.

[1] RPA = Relative Pronoun Ambiguity

[2] Int. = Interpretation



(a) low attachment

(b) high attachment

Figure 1: Partial dependency trees for the two interpretations, i. e. the different attachments of the relative clause, of the exemplary sentence of ambiguity **A1)**.

2.2. Semantically Annotated Visual Scenes

NLP systems are often not able to correctly establish reference resolution in case of linguistic ambiguities like the ones that are described before in this section because they are based on linguistic information alone. Hence, they choose interpretations with respect to statistical distributions in their training data or explicitly stated rules, e. g. the data-driven parser that our system is based on (see Section 3.) originally prefers the low attachment for the German examples of ambiguity **A1)** (see Table 2) even if the high attachment is supposed to be chosen. But, respective visual scenes eliminate interpretations and favor the target one, assuming the scenes themselves are unambiguous.

There are different scenarios in the multi-modal LASC data-set, which involve several people, objects and actions. Each interpretation of a sentence belongs to one scenario and has a corresponding visual scene that visualizes the relations between agents and objects mentioned in the sentence and that supports the target interpretation. Figure 2 shows the images that belong to the different interpretations of the example of ambiguity **A1)**. In Figure 2a, the man paints the bed, which supports the low attachment of the relative clause seen in Figure 1a, while he paints the head of the bed in Figure 2b, which corresponds to the high attachment (Figure 1b).

Since we investigate the effect of external knowledge like visual scenes on language processing, information are not derived from the images automatically. Instead, the semantic annotations that are provided for each image from the LASC data-set are taken as external, contextual information. Those annotations were created with clear knowledge of the scenes, their Agents and Patients (Alaçam et al., 2018) in order to determine the upper bound of the performance of our computational model. Nevertheless, the images are part of the LASC data-set in order to conduct comparable studies with humans, which is not addressed here because it would exceed the scope of this work.

In parts, people, objects and actions in the images are manually annotated with semantic roles, similar to the approach of McRae et al. (2001), see also (Mayberry et al., 2006). Semantic roles are linguistic abstractions to distinguish and classify different functions of a predicate in a sentence, so they specify *"who did what to whom"*, and they establish a relation between the semantic and syntactic level of an analysis. The most common semantic roles include Agent, Theme, Patient, Instrument, Location, Source and Goal (Palmer et al., 2010). Figure 3 exemplarily shows some semantic roles for the images in Figure 2. There, the *man* is the Agent, who paints something, in both images. In Figure 3a, the *bed* is the Patient, the entity undergoing a change of state caused by the painting action, which supports the low attachment of the relative clause in Figure 1a, while the *head* is the Patient in Figure 3b reinforcing the high attachment in Figure 1b.

Both the sentences and the semantic annotations of the corresponding images serve as input to the system that is introduced in Section 3. and that enables multi-modal disambiguation based on sentences as linguistic and visual scenes as contextual information.

3. Multi-Modal Disambiguation

The previous section describes the multi-modal data-set that contains exemplary linguistic ambiguities whose disambiguation is investigated in this paper. Each example consists of a sentence and a respective visual scene, which is incompletely annotated with semantic roles, as contextual information. While most state-of-the-art parsers, which rely on the linguistic input only, are not able to resolve ambiguities reliably, this section introduces a parsing scheme that reaches correct disambiguation results by exploiting the contexts.

The final system has to meet the following requirements:

R1) Examples with sentences from several languages, i. e. English, German and Turkish, are part of the data-set. Thus, the system is supposed to be language-independent instead of -specific.

R2) The context is dynamic, i. e. the scenarios, which are displayed by the visual scenes, take place in different environments. Therefore, the contexts are highly dissimilar and the system has to account for this.

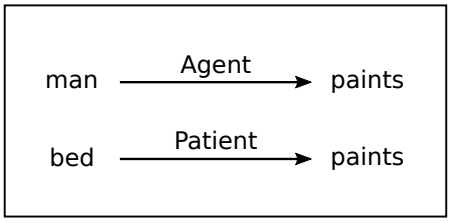


(a) The man paints the entire bed - corresponding to 1a.

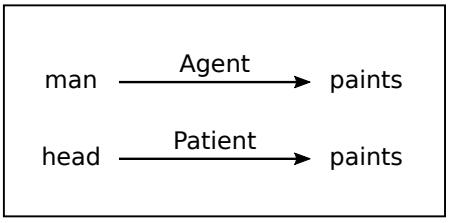


(b) The man paints only the head of the bed - corresponding to 1b.

Figure 2: The visual scenes that correspond to the two interpretations of the exemplary sentence of ambiguity **A1)**, the dependency trees of which are partially displayed in Figure 1.



(a) corresponding to 2a



(b) corresponding to 2b

Figure 3: Some semantic role annotations for the scenes in Figure 2.

R3) The parsing results for the overall sentences, besides the investigated ambiguities, are supposed to be state-of-the-art results.

Subsection 3.1. addresses requirements R1) and R3) by basing the system on a state-of-the-art, data-driven parser. To deal with the dynamic nature of the context, requirement R2), a grammar-based approach is taken, which is being outlined in Subsection 3.2..

3.1. Language-Independent Parsing

Our system is based on the data-driven, syntactic RBG-Parser (RBG) (Zhang et al., 2014), which performs graph-based dependency parsing. It possesses a scoring function to evaluate entire dependency trees respectively their edges and learns that function based on training data that are annotated dependency trees from treebanks, which exist for various languages. Furthermore, RBG does not require language specific knowledge like hand-written grammars. Therefore, it fulfills the requirement of language-independence, i. e. R1). Also, requirement R3) is met because RBG achieves state-of-the-art results for several languages (Zhang et al., 2014).

RBG extracts up to third-order local features, like sibling or grand-grandparent structures, as well as global features, e. g. span lengths, from input sentences. For a complete list of all possible features, see (Zhang et al., 2014). Hill climbing is applied if all features are exploited to approximate the most plausible dependency tree. First, a random tree is uniformly sampled. Next, the heads of all dependents are exchanged so the edges of the tree are changed until a local optimum is reached. To increase the likelihood of finding the global optimum, hill climbing repeatedly restarts, always with random trees that are sampled independently of previous solutions, until the best solution converges. During hill climbing, edges as well as entire trees are scored by RBG's scoring function, see (Zhang et al., 2014; Lei et al., 2014) for details.

3.2. Inclusion of Dynamic Contexts

While RBG fulfills the requirements of language-independence and state-of-the-art results, it is not able to deal with dynamic contexts, i. e. requirement R2). In theory, features could be extracted from the contexts, for example features capturing the relations the semantic roles, which the visual scenes are annotated with in our data-set, express. Either an RBG model might be trained on those features combined with the ones extracted from the input sentences or separate models might be trained. In both cases, the disambiguation results for the unseen test data might deteriorate and the linguistic ambiguities, which are investigated in this paper, might not be resolved because

the contexts of the test data would differ highly from the contexts of the training data. Instead, a grammar-based approach utilizes the contextual information to improve the disambiguation results.

First, the semantic roles the visual scenes are annotated with are grounded. So, the contextual information is aligned with the linguistic one. In Figure 3a, *man* is the Agent of *paints* while *bed* is its Patient. All three instances are grounded by connecting them to the respective words of the input sentence. To focus the effects of multi-modal disambiguation and to determine its upper bounds, grounding is simply based on lexical agreement to exclude possible errors. In future experiments, the lexical-based grounding will be replaced with a concept-based one. An ontology will contain a conceptional hierarchy and each concept will provide a multitude of possible lexicalizations, e. g. singular and plural forms for different cases. This way, instances of the semantic roles and words of the input sentences can differ conceptually, e. g. *chair* occurring in a sentence may refer to *piece of furniture* in the context.

In order to incorporate contextual information into the disambiguation process, our system employs jwcdg (Beuck et al., 2013), which is a graph-based dependency parser. As opposed to RBG, it is grammar-based. It possess the ability to only evaluate dependency trees respectively their edges with respect to its grammar in addition to actually parsing sentences. Possible grammars contain two types of constraints, hard ones, which are not allowed to be violated, and soft constraints, which may be violated. The result of the evaluation is a score between 0, at least one hard constraint is violated, and 1, no constraints are violated at all.

Originally, there is a jwcdg grammar available for German only. Thus, jwcdg does not fulfill the language-independence requirement. But, our system uses jwcdg only for evaluating dependency trees with respect to the context so that the original grammar is not required. Instead, a new grammar has been constructed that links semantic roles with their respective syntactic structures. For example, if the predicate of a semantic role refers to the main verb of a relative clause, which is in active voice, and if the Patient of that predicate is known and grounded as well, that relative clause is supposed to be attached to that Patient. In the example of Figures 1a and 3a, both *paints* and *bed* can be grounded, the former referring to the verb of the relative clause, which is in active voice. Thus, the relative clause is supposed to be attached to *bed*. Our linking grammar only covers semantic roles and their respective syntactic structures relevant with respect to the data-set (Section 2.) used in the experiments (Section 4.). While individual grammars have been used for the different languages in those experiments, those grammars are not lexicalized and there is an overlap of the constraints, e. g. comparable rules have been used for German and Turkish or the rules for English and German do not contradict each other, which suggests that a single grammar is able to cover several languages. Therefore, our system maintains its language-independence.

The score jwcdg determines by evaluating a dependency tree that is passed on to it by RBG is combined with the respective RBG score, which is normalized first because it has a different domain than jwcdg. Min-Max normalization (Priddy and Keller, 2005) is applied to map all RBG scores to the range of 0 to 1. After the normalization, the inverse jwcdg score, interpreted as a penalty, is subtracted from the RBG score. In case neither the dependent nor the head of an edge of a dependency tree refer to any instance of the context, that edge will not be able to violate any constraints. The same will hold true for an entire tree if none of its dependents or heads refer to the context. In those cases as well as in case references are established between the input sentence and the context but no constraints are violated, a score of 1 will be returned by jwcdg, with its inverse score being 0, and the RBG score remains unchanged. Otherwise it is decreased by the inverse jwcdg score.

Figure 4 visualizes this process. While RBG performs hill climbing for the linguistic input $x_{linguistic}$, its scoring function is repeatedly called to evaluate entire trees $t_{current}$ respectively their individual edges. jwcdg is called to evaluate whether the semantic roles $x_{semanticAnnotations}$ the corresponding visual scenes are annotated with can be linked to $t_{current}$ via the $linking\ grammar$, which links semantic roles to syntactic structures. The jwcdg score s_{jwcdg} is converted into a penalty by taking its inverse and this penalty is subtracted from the normalized RBG score $norm(s_{RBG})$. The combined scores $s_{RBG\&jwcdg}$ are returned so that the external knowledge guides the hill climbing. The best dependency tree that is found during hill climbing is returned as final solution t_{best}.

Instead of being trained on the context, our system employs a grammar-based approach to link linguistic with contextual information in order to incorporate previously unseen, i. e. dynamic, context. Therefore, our system meets requirement R2). The following sections evaluate the experiments and analyze the results.

4. Evaluation

This section presents the evaluation results of parsing linguistically ambiguous sentences both without and with exploiting contextual information. The original RBG parser, which relies on linguistic input only, is compared to our system, which performs multi-modal disambiguation and is outlined in Section 3.. The exemplary linguistic ambiguities are described in Section 2.. Three cases of relative clause attachments have been selected from the LASC data-set (Alaçam et al., 2018) that are ambiguous both in English and German. The case of attaching a relative clause to an NP or its genitive object is also ambiguous in Turkish. On the syntactic level, there are two possible antecedents for each relative clause and its relative pronoun agrees with both in number and gender. Therefore, a disambiguation is not possible without further evidence.

For the experiments, full RBG models, which use all available RBG features, i. e. global as well as up to third-order local features, have been trained for English, German and Turkish, respectively. For English, sections $0 - 22$ and 24 of the Wall Street Journal (WSJ) corpus of the Penn Treebank (Marcus et al., 1994) constitute the training set ($\approx$ 46k sentences, duplicates excluded). They have been converted to dependency structure using the LTH converter (Johansson and Nugues, 2007). For German, the first $\approx$ 98k sen-

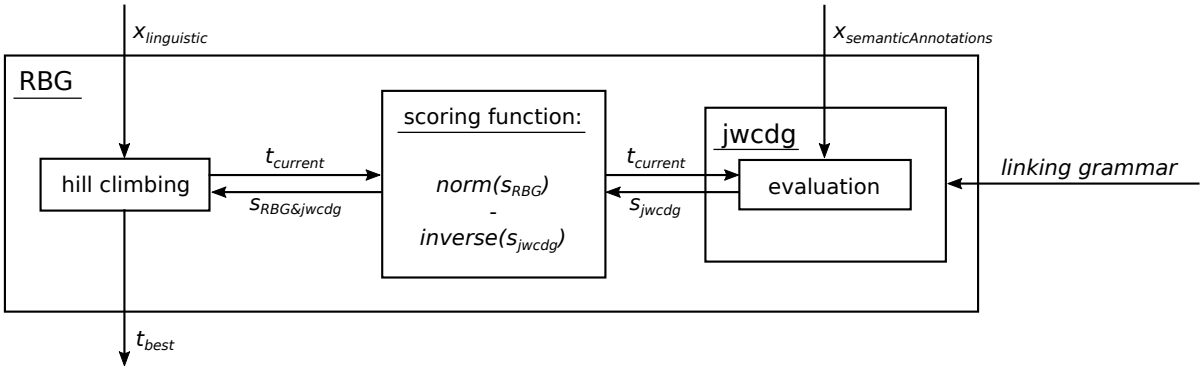


Figure 4: RBG applies hill climbing for the linguistic input $x_{linguistic}$, its scoring function is repeatedly called to evaluate entire trees $t_{current}$ or their edges. jwcdg is called to evaluate whether the semantic roles $x_{semanticAnnotations}$ from the context can be linked to $t_{current}$ via the $linking\ grammar$. The normalized RBG score $norm(s_{RBG})$ is penalized by the inverse jwcdg score s_{jwcdg} and the combined score $s_{RBG+jwcdg}$ is returned. The best dependency tree found during hill climbing is returned as final solution t_{best}.

tences (duplicates excluded) of the HDT part A are used for training. For Turkish, the training data are the first 5k sentences of the METU-Sabanci Turkish Dependency Treebank (Oflazer et al., 2003). If a word from that corpus is separated into different inflectional groups, they will be concatenated as described in (Nivre et al., 2007) so that parsing is performed on word basis. All three corpora provide word forms, gold PoS tags and gold standard annotations. While the gold PoS tags have been used for Turkish, all German and English sentences have been PoS tagged by the TurboTagger (Martins et al., 2013). Ten-way jackknifing has been performed for the training sets, i. e. each is split into ten partitions and each partition is tagged by a model trained on the other nine partitions. The test sets have been PoS tagged by models trained on the entire training set of the respective corpus.

Table 2 lists all experimental results. The accuracies of the relative clause attachments predicted by the original RBG are compared to the multi-modal disambiguations of our system. In general, RBG always favors one attachment, mostly the low one, irrespective of the supposed attachment because it does not have access to the contextual information. In contrast, our system, which exploits that external knowledge, always predicts the correct attachment.

Several observations can be made. First, while RBG mostly favors low attachments, it will predict high ones instead for ambiguities for some English examples if the relative clause is in active voice. Secondly, RBG makes some unusual predictions, sometimes attaching relative clauses to improbable antecedents, for English and Turkish. Section 5. discusses these results.

5. Analysis

This section analyzes the evaluation results, which are described in Section 4. and listed in Table 2. First, the fact that RBG always prefers either the low or the high attachment in case there are two possible antecedents for a relative pronoun is due to statistical distributions in the data the data-driven parser has been trained on. In German sentences, relative clauses are more frequently low-attached than high-, although there are examples for both. In the HDT part A, the nearest plausible attachment is chosen in 78.87 percent of all cases that resemble the examples of our experimental data, i. e. relative pronouns with ambiguous antecedents (see Section 2.). Comparable observations can be made for Turkish and it will also hold for English in case of ambiguity **A1)**. For **A2)** and **A3)**, the opposite can be observed. They are more often high- than low-attached in English. In those cases, RBG prefers the high attachment.

The disambiguation results of RBG do not match the intended interpretations, with respect to the corresponding visual scenes, because it does not take any external knowledge into account. In contrast, our system, which incorporates contextual information, always resolves the linguistic ambiguities in the experimental data correctly without exception. This result proves our hypothesis H1) that the disambiguation of linguistic ambiguities will improve if it is not only based on analyzing the linguistic input but also takes external knowledge respectively contextual information, like the semantically annotated visual scenes in this work, into account.

Furthermore, the evaluation shows that our system fulfills the requirements R1) - R3). It correctly resolves ambiguities independent of the languages although those show some major differences. As discussed before, in case of ambiguity **A2)** and **A3)**, the high attachment of the relative clause is preferred in English compared to the low one in German. Also, relative clauses usually appear after their possible antecedents in English as well as German while they precede them in Turkish. Neither of those properties influence the ability of our system to correctly resolve linguistic ambiguities. Therefore, it is language-independent, which validates hypothesis H2). Furthermore, the contexts respectively the visual scenes differ significantly, i. e. they are dynamic. Nevertheless, our system achieves correct disambiguations, which proves that it is able to deal with dynamic contexts. In addition, it is based on RBG, which is a parser that enables state-of-the-art results. Therefore, all

Ambiguity Type	Language	Voice	Supposed Attachment	# Examples	# Relative Clause Attachments					
					RBG			Multi-Modal Disambiguation		
					low	high	misc	low	high	misc
A1) RPA - a Genitive Modifier	English	active	low	24	8	16	-	24	-	-
			high	24	8	16	-	-	24	-
		passive	low	24	24	-	-	24	-	-
			high	24	24	-	-	-	24	-
	German	active	low	24	24	-	-	24	-	-
			high	24	24	-	-	-	24	-
		passive	low	24	24	-	-	24	-	-
			high	24	24	-	-	-	24	-
	Turkish	active	low	24	19	-	5	24	-	-
			high	24	19	-	5	-	24	-
		passive	low	24	19	-	5	24	-	-
			high	24	19	-	5	-	24	-
A2) RPA - a Prepositional Phrase	English	active	low	20	5	14	1	20	-	-
			high	17	3	13	1	-	17	-
	German	active	low	20	18	-	2	20	-	-
			high	17	16	-	1	-	17	-
A3) RPA - Scope Ambiguities	English	active	low	24	1	23	-	24	-	-
			high	24	1	23	-	-	24	-
	German	active	low	24	24	-	-	24	-	-
			high	24	24	-	-	-	24	-

Table 2: Comparison of the evaluation results of the original RBG and our multi-modal disambiguation regarding the ambiguous attachment of the relative clauses for the ambiguities **A1)** - **A3)** in different configurations (language, voice of the finite verb of the relative clause).

requirements are fulfilled.

RBG sometimes predicts attachments that do not correspond to the ones it generally favors. For example, it predicts the high attachment 16 times for the English version of **A1)** with the relative clause in active voice although it favors the low attachment. This is due to wrongly assigned PoS tags. If the gold PoS tags are used, the low attachment is consistently predicted. The same holds true for the German examples of **A2)**, in which RBG assigns alternative, improbable heads to the relative clauses. For Turkish, RBG always chooses the low attachment, but, in case of compound nouns, it refers to their first part although it is supposed to choose the second part.

6. Related Work

In this work, the contextual information stems from visual scenes and helps to disambiguate linguistic ambiguities. A similar effect can be observed in human language processing. Tanenhaus and his colleagues (Tanenhaus et al., 1995) showed that visual information influences how humans disambiguate linguistic input. While the authors of that study focused on PP attachment ambiguities, we used the problem of attaching relative clauses because it has been less frequently investigated than the former. Further evidence that supports the conclusion of (Tanenhaus et al., 1995) was provided by Knoeferle (Knoeferle, 2005), whose work also indicates that visual information influences language processing independent from the experiment language, she conducted experiments for English and German. In addition to those languages, some of our tests were also carried out for Turkish. Furthermore, Coco and Keller (Coco and Keller, 2015) investigated the interaction between language and vision and its influences on the interpretation of syntactically ambiguous sentences in a simple real-world setting. Their study provided further evidence that not only linguistic but also visual information influences the interpretation of a sentence. While the aforementioned empirical studies were psycho-linguistically motivated and provided insights how humans resolve linguistic ambiguities by exploiting visual cues, our work provides evidence that similar effects can be observed for automatically parsing syntactically ambiguous sentences in the presence of contextual information.

An early approach to exploit external knowledge during parsing was proposed by (McCrae and Menzel, 2007; McCrae, 2009; Baumgärtner et al., 2012). They suggested to utilize high-level representations of visual information from related scenes to resolve linguistic ambiguities in German, e.g. Genitive-Dative ambiguity of feminine nouns or PP attachment, and developed a syntactic parsing architecture for the integration of cross-modal information. Moreover, McCrae (2009) hypothesized multiple requirements for such a system. Like ours, their system uses visual scenes as context and the author did not discuss the auto-

matic derivation of information from those scenes. Also, the instances of the context are mapped onto the words of the corresponding sentence and linked via semantic relations to the syntactic structure by applying constraints. But, that system is solely constraint-based and does no longer produce state-of-the-art results. Furthermore, it requires a complete grammar of the respective language it is applied to. Thus, that approach is language specific. In contrast, our system is based on a data-driven parser, which achieves state-of-the-art results and whose model is trained on annotated data that is provided by treebanks for various languages and which is, thus, language-independent.

In a more recent work, Christie et al. (2016) proposed to jointly segment an input image semantically and resolve PP attachment ambiguities in its caption. Their approach generates several hypotheses for both segmentation and disambiguation and a model scores pairs of hypotheses for both tasks in order to find the most plausible hypotheses for both the visual and linguistic task. Compared to our work, they tried to improve not only the disambiguation but also the processing of the visual input. Furthermore, they do not assume the visual input to be perfect. Instead, we are interested in ambiguous attachments of relative clauses and, while Christie et al. (2016) employed a two-staged approach, i. e. reranking, the visual information guides the disambiguation process in this work, so the different modalities are employed at the same time, not sequentially.

7. Conclusion

This work addresses the hypotheses that linguistic ambiguities can be resolved if contextual information derived from additional modalities besides the linguistic one are exploited, independent of the language. In order to test these hypotheses, a data-driven, syntactic parser has been modified by repeatedly calling a grammar-based parser that evaluates current analyses with respect to the context, which consists of semantically annotated visual scenes. A grammar that contains constraints to link those semantic roles to the respective syntactic structures has been constructed so that the context guides the data-driven parser towards the most plausible solution given both the input sentence and the corresponding visual scene, i. e. the context.

Our hypotheses have been evaluated for several examples of relative clause attachment ambiguities under varying conditions, e. g. active versus passive voice in the relative clause, and for multiple languages, namely English, German and Turkish. While the original data-driven parser, which does not use any external knowledge, has not reached the intended interpretations consistently, instead it always preferred either low or high attachments, our system disambiguates the examples correctly, which is evidence of the hypotheses being true. Furthermore, the resulting system fulfills several requirements it is subjected to. It proofed to be language-independent and able to deal with dynamic context, e. g. highly dissimilar visual scenes.

For the future, our hypotheses will be verified for Chinese, too. Also, the influence of possible error sources will be evaluated, e. g. spelling mistakes or erroneous context that match none of the possible interpretations of the respective sentence. In addition, these images together with the corresponding sentences are going to be employed for comparative studies with humans to enable a comparison between human and machine disambiguation abilities.

Acknowledgments

This research was funded by the German Research Foundation (DFG) in the project Crossmodal Learning, TRR-169.

8. Bibliographical References

Alaçam, Ö., Staron, T., and Menzel, W. (2017). Towards a systematic analysis of linguistic and visual complexity in disambiguation and structural prediction. In *Proceedings of the 5th International Conference on Information Management and Big Data*, Lima, PERU, September.

Alaçam, Ö., Staron, T., and Menzel, W. (2018). A multimodal data-set for systematic analyses of linguistic ambiguities in situated contexts. In Juan Antonio Lossio-Ventura et al., editors, *Information Management and Big Data*, Communications in Computer and Information Science. Springer. In press.

Altmann, G. T. and Kamide, Y. (1999). Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition*, 73(3):247–264.

Baumgärtner, C., Beuck, N., and Menzel, W. (2012). An architecture for incremental information fusion of cross-modal representations. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2012 IEEE Conference on*, pages 498–503, Hamburg, Germany, September. IEEE.

Berzak, Y., Barbu, A., Harari, D., Katz, B., and Ullman, S. (2016). Do you see what i mean? visual resolution of linguistic ambiguities. *arXiv preprint arXiv:1603.08079*.

Beuck, N., Köhn, A., and Menzel, W. (2013). Predictive incremental parsing and its evaluation. In *Computational Dependency Theory*, volume 258 of *Frontiers in Artificial Intelligence and Applications*, pages 186 – 206. IOS press.

Christie, G., Laddha, A., Agrawal, A., Antol, S., Goyal, Y., Kochersberger, K., and Batra, D. (2016). Resolving language and vision ambiguities together: Joint segmentation & prepositional attachment resolution in captioned scenes. arXiv preprint arXiv:1604.02125, September.

Coco, M. I. and Keller, F. (2015). The interaction of visual and linguistic saliency during syntactic ambiguity resolution. *The Quarterly Journal of Experimental Psychology*, 68(1):46–74.

Foth, K. A., Köhn, A., Beuck, N., and Menzel, W. (2014). Because size does matter: The Hamburg Dependency Treebank. In *Proceedings of the Language Resources and Evaluation Conference 2014*, Reykjavik, Iceland, may. LREC.

Hemforth, B., Fernandez, S., Clifton, C., Frazier, L., Konieczny, L., and Walter, M. (2015). Relative clause attachment in german, english, spanish and french: Effects of position and length. *Lingua*, 166:43–64.

Johansson, R. and Nugues, P. (2007). Extended constituent-to-dependency conversion for english. In *Proceedings of NODALIDA 2007*, pages 105–112, Tartu, Estonia, May.

Knoeferle, P. S. (2005). *The role of visual scenes in spoken language comprehension: Evidence from eye-tracking*. Ph.D. thesis, Universitätsbibliothek.

Lei, T., Xin, Y., Zhang, Y., Barzilay, R., and Jaakkola, T. (2014). Low-rank tensors for scoring dependency structures. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1381–1391, Baltimore, Maryland, June. Association for Computational Linguistics.

Marcus, M., Kim, G., Marcinkiewicz, M. A., MacIntyre, R., Bies, A., Ferguson, M., Katz, K., and Schasberger, B. (1994). The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the workshop on Human Language Technology*, pages 114–119. Association for Computational Linguistics, March.

Martins, A., Almeida, M., and Smith, N. A. (2013). Turning on the turbo: Fast third-order non-projective turbo parsers. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 617–622, Sofia, Bulgaria, August.

Mayberry, M., Crocker, M. W., and Knoeferle, P. (2006). A connectionist model of the coordinated interplay of scene, utterance, and world knowledge. In *Proceedings of the 28th annual conference of the Cognitive Science Society*, pages 567–572.

McCrae, P. and Menzel, W. (2007). Towards a system architecture for integrating cross-modal context in syntactic disambiguation. In *Proceedings of the 4th International Workshop on Natural Language Processing and Cognitive Science, Madeira (NLPCS 2007)*, pages 228–237.

McCrae, P. (2009). A model for the cross-modal influence of visual context upon language procesing. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2009)*, pages 230–235, Borovets, Bulgaria, September.

McRae, K., Hare, M., Ferretti, T., and Elman, J. L. (2001). Activating verbs from typical agents, patients, instruments, and locations via event schemas. In *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society*, pages 617–622. Erlbaum Mahwah, NJ.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Oflazer, K., Say, B., Hakkani-Tür, D. Z., and Tür, G. (2003). Building a Turkish treebank. *Treebanks*, pages 261–277.

Palmer, M., Gildea, D., and Xue, N. (2010). Semantic role labeling. *Synthesis Lectures on Human Language Technologies*, 3(1):1–103.

Priddy, K. L. and Keller, P. E. (2005). *Artificial Neural Networks: An Introduction*, volume 68. SPIE press.

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., and Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, 268(5217):1632.

Zhang, Y., Lei, T., Barzilay, R., Jaakkola, T., and Globerson, A. (2014). Steps to excellence: Simple inference with refined scoring of dependency trees. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Baltimore, Maryland, June. Association for Computational Linguistics.

Overcoming the Long Tail Problem: A Case Study on CO2-Footprint Estimation of Recipes using Information Retrieval

Melanie Geiger[1,2], **Martin Braschler**[2]
[1] Université de Neuchâtel, Neuchâtel, Switzerland
[2] Zurich University of Applied Sciences, Winterthur, Switzerland
{imhf, bram}@zhaw.ch

Abstract

We propose approaches that use information retrieval methods for the automatic calculation of CO2-footprints of cooking recipes. A particular challenge is the "long tail problem" that arises with the large diversity of possible ingredients. The proposed approaches are generalizable to other use cases in which a numerical value for semi-structured items has to be calculated, for example, the calculation of the insurance value of a property based on a real estate listing. Our first approach, ingredient matching, calculates the CO2-footprint based on the ingredient descriptions that are matched to food products in a language resource and therefore suffers from the long tail problem. On the other hand, our second approach directly uses the recipe to estimate the CO2-value based on its closest neighbor using an adapted version of the BM25 weighting scheme. Furthermore, we combine these two approaches in order to achieve a more reliable estimate. Our experiments show that the automatically calculated CO2-value estimates lie within an acceptable range compared to the manually calculated values. Therefore, the costs of the calculation of the CO2-footprints can be reduced dramatically by using the automatic approaches. This helps to make the information available to a large audience in order to increase the awareness and transparency of the environmental impact of food consumption.

Keywords: BM25 weighting scheme adaptation, cooking recipe retrieval, CO2-footprint estimation

1. Introduction

One easily measurable quantitative quality criterion of a language resource (LR) is its coverage. However, achieving a high coverage usually requires a lot of human effort. One of the reasons is that often the frequencies of potential LR entries; e.g. words in human language or food products in cooking recipes (Müller et al., 2012), arrange themselves according to the so-called Zipf's law (Zipf, 1949), meaning that most entries relate to entities that occur very infrequently. Therefore, LRs are most likely never complete. This long tail problem is relevant for most applications that rely on LRs, however, it is particularly severe for information retrieval (IR) applications that not only use the LR to enhance their effectiveness (e.g. expanding queries with synonyms) but directly use the LR entries to compile their output.

To illustrate, consider a new class of retrieval applications that require the calculation of a single numerical value from a semi-structured item that consists of a list of textual elements. For example, such a semi-structured item may be a cooking recipe in which the elements are the instruction lines, such as "100g carrots, sliced" and "1 pizza dough", or a real estate listing in which the elements are the components, such as "Bedrooms: 4" and "Heating: Oil-Fired Central Heating". In those examples the numerical values to be calculated can be the nutrition value or the CO2-footprint of a recipe or for the real estate example the insurance value of a property.

An *element-wise* approach to calculating such values splits the problem into sub-problems by first calculating the value for each element individually and then computing the value of the complete item by aggregating the values of the individual elements. For most use cases, this means that the individual elements are matched to an LR, which then helps to estimate their values. In the case of recipes, the LR (Eaternity AG, 2017) we use contains the nutrition value and the CO2-value for each food product. For the estimation of real estate insurance values, a suitable LR contains the costs of the corresponding components; e.g. the average costs of a bathroom with a shower and a double washbasin. This *element-wise* approach, however, heavily relies on the completeness of the LR and has to use a fallback strategy if elements are not found in the LR. In practice, the fallback usually means that additional entries need to be added manually, an excessively costly option.

In the real estate example, an alternative human line of action is often to estimate the value of a property based on the values of other similar properties for which the value is already known. Hence, the value is estimated based on the whole item rather than the individual elements and thus the problem of the incompleteness of the LR can be circumvented. Gonzalez and Laureano-Ortiz (1992) replicate this process for automatic property appraisal. We propose an *item-based* approach using IR technology. We claim that this approach is applicable to many scenarios that include the calculation of a value for a semi-structured item whenever a similarity between the items can be defined.

In this paper, we focus on the use case of the automatic calculation of CO2-footprints of cooking recipes. The motivation for such a use case is that about one-third of CO2-emissions produced by the final household demand in Europe is caused by the consumption of food (Tukker and Jansen, 2006) and that the calculation of CO2-footprints for cooking recipes helps to increase the awareness and transparency of the environmental impact of food consumption. However, so far the footprint of a recipe was calculated with

a manual process (O'Connor et al., 2018) which is time-consuming and therefore too costly to be applied to a wide range of cooking recipes.

We describe and evaluate an *element-wise*, an *item-based*, and a *hybrid* approach, combining the two, to automatically calculate the CO2-footprints of recipes. In the context of our CO2 use case, we call the *element-wise* approach "ingredient matching" and the *item-based* approach "recipe matching". The ingredient matching approach uses an IR pipeline to match the instruction lines to the corresponding entries in the LR through retrieval from an index. The recipe matching approach finds the most similar recipe in a corpus of indexed recipes for which the CO2-footprints are already assessed. A novelty is our proposal of an adapted version of the BM25 weighting scheme which also considers the amounts of the individual ingredients in the recipes. Finally, the hybrid approach combines the two other approaches so that a higher accuracy and stability of the CO2-value estimates can be achieved.

In our experiments, we compare the automatic approaches to the manual process as well as to each other. Both the ingredient as well as the recipe-based approaches perform similarly, while our hybrid approach outperforms the individual approaches. We show that the automatic approaches lie within an acceptable range to the CO2-values calculated manually and therefore are serious alternatives. Using the approaches suggested in this paper, the cost of calculating CO2-footprints of recipes can be reduced dramatically, which makes it possible to make this information available to a large audience. The company Eaternity, which has commercialized a CO2-calculation service based on the approaches we describe, reports that it realizes a reduction in the calculation effort of 50-60% and an overall cost reduction of 80% compared to their old, manual process.

2. Related Work

Processing and more specifically choosing, designing, adapting and comparing cooking recipes has proven popular with case-based reasoning (CBR) researchers ever since the two automated meal recommendation systems CHEF (Hammond, 1986) and Julia (Hinrichs, 1989) have been presented. Many efforts are related to the Computer Cooking Contest, which runs since 2007. We distinguish between work to automatically process the ingredients of cooking recipes and work that deals with the similarity of recipes.

Several publications deal with automatically constructing a process flow graph of a given recipe (Hamada et al., 2000), (Walter et al., 2011). Hamada et al. (2000) create domain-specific dictionaries and match the keywords in the recipe to the words in the dictionaries. Based on the structure of the sentences they then construct the process flow graph. Walter et al. (2011) preprocess and annotate the recipes with GATE, a natural language processing (NLP) framework. Based on rules created from a domain expert the ingredients, as well as the actions, are linked to a workflow. Moreover, Müller et al. (2012) automatically match the ingredients of a recipe to a nutrition database in order to estimate the nutritional value of the recipe. The similarity of recipes is mostly investigated for content-based recommender systems (Teng et al., 2012), (van Pinxteren et al., 2011).

The CO2-database that we use in our experiments as well as the whole CO2-application is described by O'Connor et al. (2018), while other CO2-reduction experiments that are conducted using the automatic ingredient matching approach are described by Itten et al. (2018).

Gonzalez and Laureano-Ortiz (1992) propose a CBR system that automatically estimates the value of a property based on similar real estates handled in past experiences. If the markets for particular properties are too sparse, they use heuristic knowledge.

The K-nearest neighbor (kNN) approach is usually applied to solve classification problems where the only prerequisite is the definition of a similarity of feature vectors. It was first mentioned in a technical report in 1951 (Fix and Hodges Jr, 1951). Since then, kNN is also used for text classification amongst others by Yang (1999) and Sebastiani (2002). In this paper, we do not classify the recipe but only use the idea of nearest neighbors in order to estimate the CO2-value based on them.

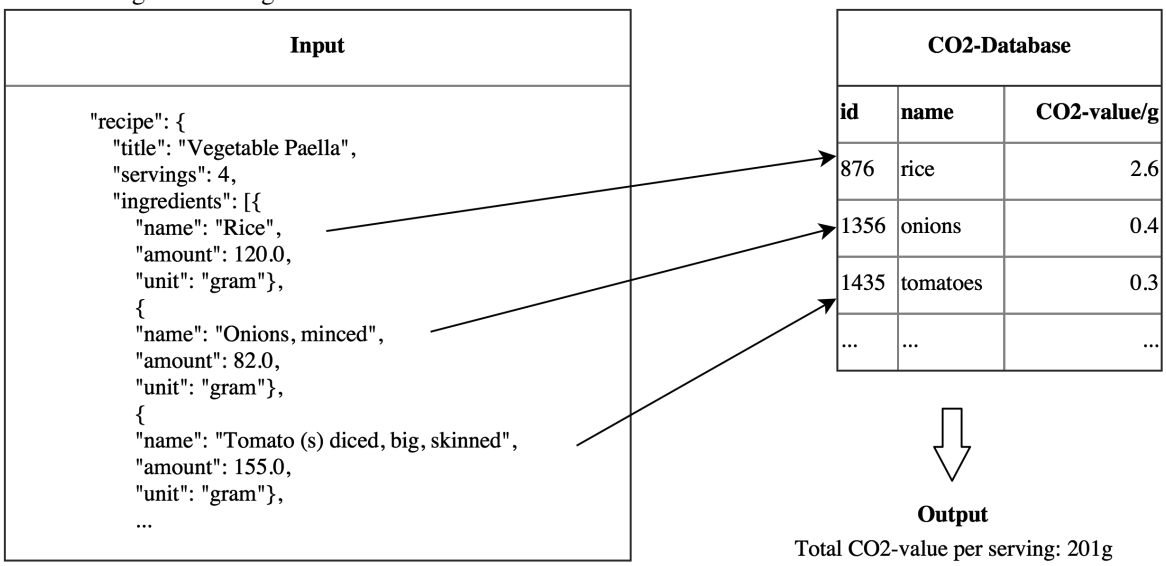
3. Methods

The case of calculating CO2-footprints is interesting for IR research on multiple fronts. As described above, in an *element-wise* approach the value is retrieved by either manually or automatically matching all the ingredient descriptions in the recipe to the appropriate food products in an LR. However, this matching is more challenging than it may appear at first glance. The difficulty stems from the fact that recipes are usually written in natural language and are therefore not restricted to use the fixed vocabulary used in the food product database. The following challenges are all very well known in NLP and IR.

The first challenge is that the authors might use synonyms in order to describe the ingredient; e.g. the ingredient description for "chard" in German might be either "Mangold" or "Krautstiel". In the cooking domain, synonyms are frequently used due to regional differences. The second challenge is the specificity of the ingredient description; both cases, very unspecific descriptions and overly specific descriptions are hard to handle correctly. For example, the unspecific description "fillet of fish" has a lot of different options to be interpreted by the cook and therefore the correct assignment to a food product in the database is a non-trivial choice. On the other hand, the description "Pinot Noir" may be too specific and in order to correctly match it to a food product in the database, the fact that this is a red wine has to be known. The third challenge is the handling of combined products, such as a pizza dough, which themselves consist of several other products.

In order to match the combined products to the food products in the database, the database either needs to contain them as well or a process to recursively split them into their base food products has to be defined. In addition to the three challenges described, we have to handle different word forms, word compounds, special characters, etc. Moreover, the food products used in recipes worldwide are manifold and it has been shown that most of them only appear in very few recipes (Müller et al., 2012). Also, new

Figure 1: Visualization of the manual as well as the automatic ingredient matching approach on an explanatory excerpt of a recipe. Note that the CO2-values are simplified for this example. In the real application, they differ depending on the season and the origin of the ingredients.



products are continuously introduced to the marketplace. According to these facts, a food product database is basically never complete.

In the following sections, we propose three approaches to automatically calculate the CO2-value of cooking recipes using NLP and IR methods. Section 3.1. describes our first approach, which we call *ingredient matching*. It reproduces an automatic version of the traditional manual process of assigning CO2-values to ingredient descriptions. Section 3.2. describes the *recipe matching* approach, which estimates the CO2-value of a recipe based on similar recipes rather than the individual ingredients. Therefore, it does not depend on the completeness of the food product database. Our third approach, the *hybrid approach*, combines the ingredient and the recipe matching approaches and therefore benefits from the advantages of both. It is described in Section 3.3.

3.1. Ingredient Matching

The ingredient matching approach matches all ingredient descriptions individually to the corresponding food product in the CO2-database (Eaternity AG, 2017) and the estimate is computed by the sum of the CO2-values of each food product multiplied with the corresponding amount. As shown in Figure 1, the input is a semi-structured recipe and the output is a mapping of the ingredients to the food products in the CO2-database as well as the total CO2-value of the recipe per serving.

For each ingredient description in the German input recipes, we find the best matching food product in the database. Therefore, we create an index of food products using a traditional IR pipeline including stemming, stopword removal, decompounding and synonym handling. Apart from the commonly used stopwords, we also use several domain-specific stopwords such as pasteurized, portion and minced. To ensure a high precision for the matching, we use a light stemmer (Savoy, 2002). Since the experiments are based on German recipes, we employ a decompounding component that splits compound words such as "Zitronensaft" (lemon juice) into their components "Zitrone" (lemon) and "Saft" (juice) using a dictionary-based n-gram decompounder with a minimum word size of 10 and a minimum and maximum word constituent size of 4 respectively 12. Along with the original food products, we also index their synonyms with the same CO2-information. Moreover, we also add combined products to the index, for which the CO2-values are manually pre-calculated.

Table 1: Notation used for BM25 and our adaptations thereof.

d_j	single document
q	single query
Φ	indexing vocabulary
φ_k	single indexing feature
l_j	length of document
Δ	average document length
$\Phi(d_j)$	set of features representing document d_j
$\Phi(q)$	set of features representing query q
$w(\varphi_k, d_j)$	weight of feature φ_k for document d_j
$w(\varphi_k, q, d_j)$	weight of feature φ_k for query q and d_j
$\text{ff}(\varphi_k, d_j)$	feature frequency of feature φ_k for d_j
$\text{df}(\varphi_k)$	document frequency of feature φ_k

The search in the index is performed using an adaptation of the BM25 weighting scheme (Robertson and Zaragoza, 2009) that ignores the inverse document frequency. Unlike in most other IR applications, the fact that a term appears often in the collection does *not* mean that it is less important. For example, the database might contain several products containing the term "apple", such as apple, apple juice, and apple puree. However, the terms juice and puree should not be weighted heavier than apple, since a match to one of the three food products containing apple is already a much

better fit than a match to for example orange juice. On the other hand, the term frequency is needed since some ingredient descriptions contain the same stemmed term multiple times and thus we assume that it is indeed more important than others. Since the number of terms in the ingredient description varies, we apply a document length normalization. Hence, we use the retrieval status value (RSV) of document d_j w.r.t. query q according to BM25

$$w(\varphi_k, d_j) := \frac{\text{ff}(\varphi_k, d_j)}{k_1((1-b) + b\frac{l_j}{\Delta}) + \text{ff}(\varphi_k, d_j)} \cdot \log\left(\frac{0.5 + N - \text{df}(\varphi_k)}{0.5 + \text{df}(\varphi_k)}\right) \quad (1)$$

$$w(\varphi_k, q) := \text{ff}(\varphi_k, q) \quad (2)$$

$$\text{RSV}(q, d_j) := \sum_{\varphi_k \in \Phi(q) \cap \Phi(d_j)} w(\varphi_k, d_j) \cdot w(\varphi_k, q), \quad (3)$$

where we set $\text{df}(\varphi_k) = 1$ for all features φ_k. Table 1 shows an overview of the notation used. Apart from the ignored inverse document frequency, we employ BM25 with the commonly used term frequency saturation parameter $k_1 = 1.2$ and document length normalization parameter $b = 0.75$. The default parameters are used due to the lack of suitable training data and to avoid overfitting. For the ingredient descriptions for which no food product can be retrieved from the index, we assign an artificial food product that has an average CO2-value.

3.2. Recipe Matching

The goal of our recipe matching approach is to estimate the CO2-footprint of an arbitrary recipe from the most similar recipe in a database of recipes for which the CO2-footprints are already known. Hence, we exploit the knowledge we already gathered with either a manual or a semi-automatic process that allocates the CO2-values. Unlike the ingredient matching, the recipe matching does not rely on assigning the individual ingredients to a database entry. Therefore, this nearest neighbor approach overcomes the long tail problem introduced by the incompleteness of the ingredient database.

An approach using IR techniques to find the most similar recipe is to run a textual search with the description of the ingredients in the query recipe against an index in which the recipes are indexed with all their ingredient descriptions. However, the similarity used in this approach does not reflect the amounts of the ingredients in the recipes, e.g. a recipe with 500g flour and 3g salt would be similar to a recipe with 500g salt and 3g flour.

Therefore, we suggest a method that also considers the amounts as an additional information, so that recipes have a higher similarity if the difference between the amounts of their respective ingredient descriptions is small. Our approach is based on an adjustment of the BM25 weighting scheme although other popular weighting schemes such as language models or divergences from randomness could be used. We adapt the weight of the query terms so that the difference between amounts of the ingredients in the query recipe and the document recipes is considered. A query term, i.e. an ingredient description, is weighted with the reciprocal difference between the amounts of the two the ingredients. Hereby, we choose the formula so that a difference of zero leads to a weight of one. Therefore, we also store the amounts of each term in each recipe, so that we can quickly retrieve the amount of a term in a given recipe when comparing recipes. In case an ingredient description consists of several terms, the amount of the ingredient will be assigned to each of its terms.

The retrieval status value (RSV) of document d_j w.r.t. query q of the adjusted BM25 that considers the amounts of the ingredients is therefore defined as:

$$w(\varphi_k, d_j) := \frac{\text{ff}(\varphi_k, d_j)}{k_1((1-b) + b\frac{l_j}{\Delta}) + \text{ff}(\varphi_k, d_j)} \cdot \log\left(\frac{0.5 + N - \text{df}(\varphi_k)}{0.5 + \text{df}(\varphi_k)}\right) \quad (4)$$

$$w(\varphi_k, q, d_j) := \frac{\text{ff}(\varphi_k, q)}{|\text{a}(\varphi_k, q) - \text{a}(\varphi_k, d_j)| \cdot \alpha + 1} \quad (5)$$

$$\text{RSV}_{\text{BM25}}(q, d_j) := \sum_{\varphi_k \in \Phi(q) \cap \Phi(d_j)} w(\varphi_k, d_j) \cdot w(\varphi_k, q, d_j), \quad (6)$$

where $\text{a}(\varphi_k, r)$ is the amount of the term k in the recipe r and α is a tuning parameter to weight the difference between the amounts. The *tf* saturation parameter k_1 and the document length normalization parameter b are used as in the original definition of BM25. Note, that only the definition of $w(\varphi_k, q, d_j)$ is different to the one in the original BM25, where it is equal to $\text{ff}(\varphi_k, q)$.

Once the most similar recipe is known, we can use its CO2-value as an approximation of the CO2-value of the input recipe.

3.3. Hybrid Matching

The use of a hybrid approach is motivated by the failure analysis of the two individual approaches. Our goal is to obtain a more robust estimate that reduces the number of outliers, where the automatically generated value is far from the correct, manual assessment. Table 2 summarizes the reasons why the ingredient matching and recipe matching approaches produce inaccurate estimates which are outside of an acceptable range with respect to the manually computed value. The ingredient matching results in a bad estimate when one or several ingredient descriptions can not be matched to the correct food product in the database. The reason is either that the correct food product does not exist in the database (long tail problem) or that the IR pipeline fails to retrieve the correct food product. Generally, the estimates do not lie within an acceptable range either if many ingredient descriptions are not correctly matched; i.e. the error accumulates; or if a few ingredient descriptions with a high CO2-impact are matched to food products with a low CO2-impact or vice versa.

There are three main reasons for the recipe matching to produce an estimate that does not lie within an acceptable range. The first reason is that the search space in which the recipe matching approach finds the nearest neighbor often has regions in the vector space in which it is not dense. In these regions, the distance between the recipe and its nearest neighbor is bigger than in other regions where the search

Table 2: Summarized reasons for estimation errors.

Ingredient Matching	Recipe Matching
Long Tail Problem	Sparse Space Problem
IR Pipeline Problem	IR Pipeline Problem
	Granularity Problem

space is less sparse. In the recipe domain, the different regions in the vector space might also correspond to cultural differences. For example, our test collection contains a lot of Swiss menus and not so many Asian recipes; therefore, in general, the estimates for Asian menus are less accurate than for Swiss menus. The second reason for bad estimates is that the true nearest neighbor can not be retrieved since it uses a different vocabulary. The third reason for estimates that are far from the manually calculated value can be summarized as granularity problems. This means that the recipe matching, which operates on the whole recipe rather than on the individual elements, fails to produce a good estimate if the nearest neighbor recipe is similar to the input recipe, although there are small but decisive differences in the ingredients that lead to a completely different CO2-footprint. The different kinds of failures of the two approaches lead to situations where either only one of the approaches produces an estimate that is rather far from the ground truth or that one overestimates and the other underestimates the CO2-value. Therefore, we propose a hybrid matching approach that uses the average of the two CO2-estimates of the ingredient matching and the recipe matching as a new estimate, as shown in equation 7.

$$\text{estimate}_{\text{hybrid}} = \frac{\text{estimate}_{\text{ingredient}} + \text{estimate}_{\text{recipe}}}{2} \quad (7)$$

This flattens the outliers produced by the individual approaches and makes sure the system can provide a CO2-estimate for more recipes.

4. Test Collections and Language Resources

For the experiments, we use two collections of recipes that were created specifically for this task. The first collection, the so-called *hobby collection*, consists of 243 vegetarian and vegan recipes from chefkoch[1], an online platform for recipes. The second collection, the *catering collection*, contains 600 recipes from the catering company "Compass Group (Schweiz) AG", a subsidiary of Compass Group PLC, the largest caterer worldwide. The recipes in both collections are in German and are in a semi-structured form given by either chefkoch or the catering company's enterprise resource planning system. This means, each instruction line is provided with separate fields for amount, unit and ingredient description, hence no information extraction is needed. Different units, such as the number of teaspoons, are converted to grams using a simple set of rules.

The ingredient matching approach matches the ingredient descriptions to a product index that contains 3,121 food products that was generated from the LR (Eaternity AG, 2017). The LR contains base food products with their CO2-values as well as synonyms that are linked to the base food products. The LR contains a lot of very region specific food products, such as "Cervelat" and "Roesti" which are frequently used in cooking recipes in Switzerland. The recipe matching approach searches for the nearest neighbor recipes in a recipe index that contains approximately 50,000 recipes. Most of the recipes are from catering companies others are from chefkoch and various other sources. Both the product index and the recipe index are primarily in German.

We manually built a ground truth for both the hobby and the catering collections. That means that for each ingredient in each recipe we manually assigned the best matching food product in the product index. Based on this ground truth it is possible to calculate the CO2-footprint of each recipe in the collection. For example, "spaghetti carbonara" has an expected CO2-value of 774g. There are also recipes with a much larger CO2-value such as "schnitzel with french fries" which has a CO2-value of 2,366g. Table 3 shows the range of the CO2-values of the recipes in these collections. The hobby collection has a significantly smaller average CO2-value per recipe (1,100g) than the catering collection (1,700g) since the hobby collection only contains vegan and vegetarian recipes.

Table 3: Statistics of the test collections.

Collection	Catering	Hobby
Number of recipes	600	243
Minimum CO2-value	32g	113g
Maximum CO2-value	13,513g	1,732g
Average CO2-value	1,700g	1,100g

5. Experiments

5.1. Ingredient Matching

The ingredient matching approach matches the ingredient descriptions to the food products in the database. Table 4 shows the precision, the fraction of correctly matched ingredient descriptions, as well as the mean absolute error (MAE) and the Pearson correlation between the CO2-value estimate from the ingredient matching and the CO2-values from the manual matching. Hereby, correctly matched means that the automatic matching is strictly equal to the manual matching. The mean absolute error is the average of all the absolute errors in the test collection, where the absolute error of a recipe is the difference between the expected CO2-value and our estimate. For example, "spaghetti carbonara" has an expected CO2-value of 774g and an estimate of 684g which results in an absolute error of 90g. The Pearson correlation measures the linear dependence between two variables, in our case the manually assessed CO2-values and the estimates from the automatic process. The possible values are between 1 and -1, where 1 is the maximal positive correlation, 0 means no correlation and -1 is the maximal negative correlation.

The precision for the catering collection is slightly higher than for the hobby collection, mostly since the food product database was mainly designed for catering recipes. At

[1] http://www.chefkoch.de/

Table 4: Matching results using the ingredient matching approach on the two test collections *catering* and *hobby*.

Collection	Catering	Hobby
Precision	0.72	0.68
Mean absolute error	336g	163g
Pearson correlation	0.81	0.73

first glance, the achieved precisions of 0.72 respectively 0.68 are not that encouraging. However, given that other studies show that even the consensus of human assessors is smaller than 75% for 23% of the recipes (Müller et al., 2012), the achieved precision is at least acceptable. Having a closer look at some of the wrongly matched ingredients descriptions, we indeed find many examples which are within the margin of human disagreement. For example, "celery large" is wrongly matched to "celery root" instead of "celery stalks" as denoted in the ground truth, although both seem to be valid options. There are however also some IR specific issues. For example, "red trout fillet (breed)" is wrongly matched to "salmon trout (breed, fillet)" rather than "trout".
The significantly smaller average CO2-value per recipe in the hobby collection, as shown in Table 3, is the main reason why the MAE of the hobby collection is smaller than the MAE of the catering collection.

5.2. Recipe Matching

The recipe matching approach, in which we estimate the CO2-value of an input recipe by its most similar recipe, heavily relies on the size of the recipe corpus from which the similar recipes are retrieved. Our retrieval system is built on top of Lucene and is using the built-in BM25 weighting scheme with the default saturation parameter $k_1 = 1.2$ and the document length normalization parameter $b = 0.75$.
Table 5 shows the MAE and the correlation between the CO2-value estimate from the recipe matching and the CO2-values from the manual matching. For the experiments, we use $\alpha = 0.02$ as the tuning parameter of the weight of the difference between the amounts. Note that our ground truth does not include the closest neighbor of the recipes, but only the manually assigned food products and the total CO2-value of each recipe, thus we do not specify the precision for the recipe matching approach.

Table 5: Matching results using the recipe matching approach on the two test collections *catering* and *hobby*.

Collection	Catering	Hobby
Mean absolute error	310g	360g
Pearson correlation	0.83	0.14

In order to explain the different performances of the algorithm on the two datasets, we first have a look at the two collections. As already stated previously the hobby collection only contains vegan and vegetarian recipes from a hobby cooking platform, while the catering collection contains recipes from several canteens in Switzerland. Having a closer look shows that the two collections are quite different regarding the number of ingredients used in each recipe. An average recipe in the hobby collection consists of 12.7 ingredients and an average recipe in the catering collection has 20.5 ingredients. However, not only the number of ingredients is different but also the ingredients themselves. Therefore the most similar recipe from which the CO2-value is used as an estimate is most likely a recipe from the same category (hobby or catering) as the input recipe. The corpus used to retrieve the recipes with already allocated CO2-values consists of approximately 50,000 recipes from which only around 1% are recipes from the hobby domain, while all the others stem from the catering domain. The lack of close neighbors; i.e. too few recipes from the hobby domain, therefore explains the small correlation (0.14) of estimates in the hobby collection. Even though the performance of the recipe matching for the hobby collection is not as good as for the catering collection, the MAE for the hobby collection (360g) is still in the same range as for the catering collection (310g) due to the smaller average CO2-value of the vegan and vegetarian hobby recipes.

5.3. Hybrid Matching

The hybrid matching approach combines the ingredient matching and recipe matching by averaging the two estimates and therefore is able to account for their individual shortcomings. Table 6 shows the MAE and the correlation between the CO2-value estimate from the hybrid matching and the CO2-values from the manual matching.

Table 6: Matching results using the three matching approaches on the two test collections *catering* and *hobby*.

Method	Measure	Catering	Hobby
Ingredient	Precision	0.72	0.68
	Mean absolute error	336g	163g
	Pearson correlation	0.81	0.73
Recipe	Mean absolute error	310g	360g
	Pearson correlation	0.83	0.14
Hybrid	Mean absolute error	279g	206g
	Pearson correlation	0.90	0.55

For the catering collection the hybrid matching approach achieves a better result for both measures (MAE and correlation) than the other approaches individually. In spite of the significantly worse performance of the recipe matching for the hobby collection the hybrid matching only achieves a slightly worse result as the ingredient matching. These results show that in the case in which both individual approaches achieve an acceptable performance the hybrid matching results in more reliable estimates.

6. Conclusions

We proposed three approaches using IR to automatically compute a single numerical value of a semi-structured item that consists of a list of textual elements based on the use case of calculating CO2-footprints of cooking recipes,
The first approach, ingredient matching, calculates the CO2-footprint on an element-basis; i.e. the ingredients.

Our experiments show that the CO2-value estimates of the ingredient matching lie within an acceptable range compared to the estimate of the manual calculation. The second approach estimates the CO2-values based on similar recipes rather than individual ingredients. Since the estimate is no longer based on the individual ingredients, this recipe matching approach overcomes the long tail problem of the ingredient matching, i.e. that the food product LR is most likely not complete.

For the similarity of recipes, we proposed an adaptation of the BM25 weighting scheme that takes the different amounts of the ingredients into account. We showed that the recipe matching slightly outperforms the ingredient matching, if the recipe corpus is large enough. We have reason to believe, that the effectiveness of matching would increase as the size of the collection of recipes that is searched against increases.

Combining both the ingredient matching and the recipe matching with our hybrid approach allows us to estimate the CO2-value even more accurately. It is therefore able to balance the shortcomings of the individual approaches. The achieved correlation of 0.9 between the CO2-value estimates of the hybrid matching and the CO2-value estimates of the manual matching shows that the automatic calculation is a serious alternative to the manual calculation and therefore the costs of a manual calculation can be reduced dramatically by instantiating the automatic calculation. Indeed, first experiences from using the approaches in the commercial CO2-calculation service of our partner Eaternity indicate a reduction in effort for the calculations in the range of 50-60%, with an even higher overall cost reduction of 80%.

As a next step, the accuracy of the estimates of the hybrid matching approach could possibly be further improved by weighting the estimates of the ingredient and the recipe matching based on an estimate of their reliability. The reliability of the CO2-estimates of the recipe matching could for example be predicted using the distance between the input recipe and its nearest neighbor.

7. Acknowledgments

We thank Compass Group (Schweiz) for the recipe dataset and Eaternity AG for the project work in which most of the approaches presented in this paper were developed.

8. Bibliographical References

Fix, E. and Hodges Jr, J. L. (1951). Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, DTIC Document.

Gonzalez, A. J. and Laureano-Ortiz, R. (1992). A case-based reasoning approach to real estate property appraisal. *Expert Systems with Applications*, 4(2):229 – 246.

Hamada, R., Ide, I., Sakai, S., and Tanaka, H. (2000). Structural analysis of cooking preparation steps in japanese. In *Proceedings of the fifth international workshop on on Information retrieval with Asian languages*, pages 157–164. ACM.

Hammond, K. J. (1986). Chef: A model of case-based planning. In *AAAI*, pages 267–271.

Hinrichs, T. R. (1989). Strategies for adaptation and recovery in a design problem solver. In *Proc. of the 2 nd Workshop on Case-Based Reasoning*, pages 115–118.

Itten, R., Imhof, M., Jaggi, A., and Stucki, M. (2018). *Combining Natural Language Processing and Life Cycle Assessment for Computer-Aided Optimisation of Greenhouse Gas Emissions in System Catering*. unpublished manuscript.

Müller, M., Harvey, M., Elsweiler, D., and Mika, S. (2012). Ingredient matching to determine the nutritional properties of internet-sourced recipes. In *2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, pages 73–80. IEEE.

O'Connor, I., Aleksandrowicz, A., Scheuss, A., Jaggi, A., Klarmann, M., and Ellens, J. (2018). *Description of the Eaternity Database*. unpublished manuscript.

Robertson, S. and Zaragoza, H. (2009). *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc.

Savoy, J. (2002). *Morphologie et recherche d'information*. Université de Neuchâtel, Faculté de droit et des sciences économiques, Division économique et sociale.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.

Teng, C.-Y., Lin, Y.-R., and Adamic, L. A. (2012). Recipe recommendation using ingredient networks. In *Proceedings of the 4th Annual ACM Web Science Conference*, pages 298–307. ACM.

Tukker, A. and Jansen, B. (2006). Environmental impacts of products: A detailed review of studies. *Journal of Industrial Ecology*, 10(3):159–182.

van Pinxteren, Y., Geleijnse, G., and Kamsteeg, P. (2011). Deriving a recipe similarity measure for recommending healthful meals. In *Proceedings of the 16th international conference on Intelligent user interfaces*, pages 105–114. ACM.

Walter, K., Minor, M., and Bergmann, R. (2011). Workflow extraction from cooking recipes. In *Proceedings of the ICCBR 2011 Workshops*, pages 207–216.

Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1-2):69–90.

Zipf, G. K. (1949). Human behavior and the principle of least effort.

9. Language Resource References

Eaternity AG. (2017). *Eaternity Food Product Database*. Eaternity AG, not publicly available, 1.0.

Comparison of Pun Detection Methods Using Japanese Pun Corpus

Motoki Yatsu* **and Kenji Araki****
*Aoyama Gakuin University,
5-10-1 Fuchinobe, Chuo-ku, Sagamihara, Kanagawa, 252-5258, Japan
yatsu@it.aoyama.ac.jp
**Hokkaido University
Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido, 060-0814, Japan
araki@ist.hokudai.ac.jp

Abstract
A sampling survey of typology and component ratio analysis in Japanese puns revealed that the type of Japanese pun that had the largest proportion was a pun type with two sound sequences, whose consonants are phonetically close to each other in the same sentence which includes the pun. Based on this finding, we constructed rules to detect pairs of phonetically similar sequences as features for a supervised machine learning classifier. Using these features in addition to Bag-of-Words features, an evaluation experiment confirmed the effectiveness of adding the rule-based features to the baseline.

Keywords: corpus, Japanese, humour, puns, detection

1. Introduction

In recent years, there has been growing interest in machines expressing and understanding humour. There is an interest in computational humour improving the quality of life (QOL) of the user, such as alleviating the psychological stress of the person expressing himself (Yamada et al., 2012), and improving depressive symptoms (Tsukawaki et al., 2011).

Research to obtain such effects in artificial agent and robot communication is under way. For example, Dybala et al. (2012) showed that the user feels unexpected to the system and the desire to continue dialogue with the system is increased under the condition that the dialog agent generates humour. Miyazawa et al. (2012) conducted a survey of factors that enhance the continuity of the dialogue between the dialogue system and the user. As a result, it was shown that a humourous dialogue would get enhanced its continuity.

In this paper, first we take a look at what types of Japanese pun are prominent from a categorization survey. Subsequently, we describe the rule-based and machine learning-oriented features as training data for a supervised machine learning method, reflecting the phonological characteristics that Japanese puns share.

(Seed) *Futon-ga futton-da.* (Stc. 1)
(Transformed)
The *futon* blew away.

Kiri-no naka-de kiri-o kiri taoshi-te mo
kiri-ga nai. (Stc. 2)
Even if you cut down paulownia in the fog, you will not reach the end.

Suimasenbazuru! (Stc. 3)
suimasen *senbazuru*
I apologize (with a thousand of paper cranes)!

Figure 1: Component ratios of the types of Japanese puns (*italic*) found in the pun corpus noted in Section 2.2.

2. Japanese Puns as Text Humour

2.1 Types of Japanese Puns

According to (Takizawa, 1995), Japanese puns can be classified mainly into two classes: **juxtaposed puns** and **superposed puns**.

In a **juxtaposed pun**, a **seed** expression refers to one or more independent (noun, verb, adjective, or emoticon) morpheme(s) or phrase(s) in the sentence. A **transformed** expression refers to a phoneme string in an arbitrary section in a sentence having phonological similarity with seed expression(s). For example, the first sentence (Stc. 1) shown in Figure 1 have a pair of phoneme sequences phonologically similar, one of which has an extra 't' character. As another example, Stc. 2 has more than one pair of similar phoneme sequences. These kinds of puns are classified into juxtaposed puns.

Kawahara and Shinohara (2012) define **perfect** puns as juxtaposed puns whose seed and transformed expressions have no phonological difference (as seen in stc. 2). Similarly, those that do not match due to changes or dropout of phonemes (stc. 1) are referred to as **imperfect** puns.

Superposed puns, on the other hand, have only one surface part with two or more semantic interpretation. An example is shown as Stc. 3.

2.2 Pun Database

We created a pun database, which consists of sentences as the source for Japanese juxtaposed pun humour texts, with a scale that has never been achieved. A web crawler was

Pun Type		Frequency
Juxtaposed puns	Perfect	42 (7.0%)
	Imperfect	521 (86.8%)
	Total	563 (93.8%)
Superposed puns		33 (5.5%)
Other		4 (0.7%)

Table 1: Component ratios of the types of puns in a sample of 600 sentences from pun corpus.

Website Name	URL	# of Sentences
Dajare Nabi	http://www.dajarenavi.net/pc/i_today_index.htm	39,120
Dajare Suteshon	http://dajare.jp	8,795
Dajare Netto	http://www.dajare.net	1,621
Hitokuchi Dajare Daishuugou	http://www.biwa.ne.jp/~aki-ina/gyagu.html	1,067
Dajare Shu Dajare Jiten	http://dajareshuu.web.fc2.com	982
Dajare No Kanzume	http://www.geocities.jp/pikumin_hiroba/dajare.html	572
Dajare Kurabu	http://with2.net/dajakura/	428
Dajare Hiroba	http://www1.ocn.ne.jp/~origo/dazyare	303
Dajare o Itta no ha Dareja?	http://wtpage.info/dajare/	107
Total number of extracted pun sentences		52,995
Number of pun sentences after removing dupicates & near-duplicates		45,970

Table 2 : The source websites from which pun sentences were obtained.

run to gather pun sentences from specified websites (shown in Table 2) that contained humourous sentences. Over 95% of the data obtained here is shared with the Japanese pun database created by Araki et al. (2017).

We obtained Japanese puns as positive data of a classification problem. Originally, in analyzing puns, as with many other linguistic phenomena, it is necessary to analyze them reflecting the frequency of occurrence in the real world and conditions for occurrence (such as topics with high possibility of occurrence of conversation including puns). However, with regard to the use of puns in everyday conversation, actual corpus and its statistical analysis result are not available so far and it is still difficult to find concrete and practical examples.

Therefore, we obtained the positive data from the Web using a spider. Duplicate and near-duplicates (with the difference of 3 characters or less) were removed at the time of acquisition. Table 2 shows the websites from which the data was obtained, the total number of extracted sentences, and the final number of obtained pun sentences. From the results of sampling survey we made against this pun corpus (shown in Table 1), we can see that Juxtaposed and imperfect puns have a large portion (> 85.0%).

3. Pun Detection Methods

In this research, we refer juxtaposed puns as the main object of detection. The reason is that, firstly, the number of sentences with juxtaposed puns has a large proportion of all the pun sentences, as we noted shortly in Section 2.1. We use a Support Vector Machine (SVM) as a supervised machine learning method, as a two-class classifier using vector space model with input as vector. We used *scikit-learn* [1] to implement SVM. Using machine learning methods allows us to implement multiple detection methods by choosing different sets of features as training data. The features used in the experiment were:

Rule-based features: **(a)** identical match of phonemes in seed and transformed expressions, **(b)** handling alternation of phonemes with phonological similarity, **(c)** allowance of deletion/addition of repetitive consonant phoneme added to (b).

Machine learning-oriented features: **(d)** Bag-of-Words features.

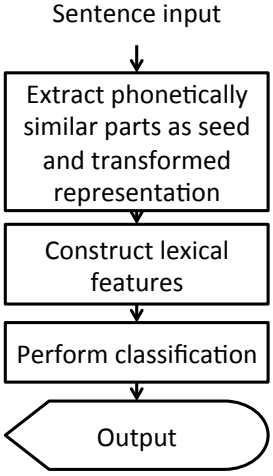


Figure 2: The outline of the proposed pun detection method.

3.1 Rule-based Features

Feature (a): A reading *Kana* sequence is extracted using a morphological analyzer *MeCab*[2] and *JUMAN*[3]. Here, each token or sequence of them is treated as a **seed** expression. The feature value is set to 1.0 when the pronunciation of the reading *Kana* is identically reproduced as transformed expression. However, in order to prevent erroneous detection due to a large number of matching sections for a reading *Kana* of one character, the target reading *Kana* shall be two or more characters.

Feature (b) and (c): In addition, since imperfect juxtaposed pun with a part of the vocal sounds transformed occupies more than 85.0% of targeted puns, we consider that absorption of different pronunciations is necessary for detecting phonemic similar parts. Therefore, we constructed features **(b)** and **(c)** using phonological similarity of consonants described later. Specifically, the following processing is added to the reading *Kana* sequence extracted in the feature (a). First, in order to avoid the consistency of single words, we divide the reading *Kana* into mora units and arrange it. At this time the method converts the long tone to diphthong moras and give the mora of geminate and repellent unique signs. The consonant phonemes and vowel phonemes in the mora are separated and stored in one array.

[1] http://scikit-learn.org
[2] http://taku910.github.io/mecab/
[3] http: //nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN

For example, *intaashoppu* (intershop) is converted into an array such as `{/*,i/, /*,N/, /t,a/, /*,a/, /sh,o/, /*,Q/, /p,u/}`. Here, `N` represents sound absorption, and `Q` represents geminate. The notation of phonemes of other consonant parts is the same as the phonetic expression by Hepburn type *Romaji*.

The system calculates consonant similarity for different consonants p and q using the function

$$s_c(p,q) = \frac{N_{\text{pair}}\, N_{p,q}}{N_p\, N_q}. \quad (1)$$

Here N_p and N_q are the concurrence frequency of the two consonants in a seed and transformed expressions respectively, or vice versa, and N_{pair} the number of all the consonant pairs.

For example, when calculating phonological similarity of a pair of expression `{/n,i/, /*,N/, /gy,o/}` and `{/n,i/, /*,N/, /j,o/}`, the consonant pair, `/j/` is used. If using the development data constructed in Section 4, $s_c(/\text{gy}/,/\text{j}/) = 0.1$. And if there are multiple transformed expression candidates for one type expression candidate, we treat all transformed expression candidates as equivalents rather than the one with a higher value.

In addition, even when the pluralities of transformed expression candidates become highest, all the transformed expression candidates are detected accordingly. In the condition of detection of seed / transformed expression, the number of detections within one sentence of the seed / transformed expression and the magnitude of the similarity value is not taken into consideration. However, we consider that these indicators give superficial information indispensable for automatic evaluation of recognition of the meaning of a pun and how fun it is.

3.2 Machine Learning-oriented Features

Features (d): The proposed method creates Bag-of-Words features by extracting the stem of a content word (noun, proper noun, general verb, adjective, adverb, emotive verb) using MeCab as the morphological analyzer, and IPAdic (Asahara and Matsumoto, 2003) the morpheme dictionary.

The values of Bag-of-Words features are determined as follows. We associate Bag-of-Words acquired from all input sentences of training data in advance with feature numbers. At the time of learning and detection, similarly the method obtains a Bag-of-Words from input sentences. Finally, the method lets the value of the feature corresponding to the content word to be 1.0.

4. Detection Experiment and Results

For the experiment, negative example data consisting of sentences not including puns was prepared. We extracted the sentences for the data from the YACIS corpus (Ptaszynski et al., 2012), which collected blog articles in Japanese posted between 2011 and 2012. We randomly extracted the sentences

In actual data such as chat dialogue corpus, the number of positive examples is considered to be extremely small compared to the number of negative examples. In such a case, it is necessary to be able to suppress deterioration of detection performance against robustness reduction (robustness). In order to compare the robustness of each feature set, we used the feature sets (shown in Table 3) F8, F9, F10, F11, F12, F13 and F14. These feature sets showed particularly high detection performance in preliminary experiments. We performed a 20-fold cross validation for evaluation data, changing the ratio of positive data number to negative one. The ratio was repeatedly increased by 0.01, from 0.01 to 1.0. Figure 3 shows the plot of the positive example negative example ratio in this result on the horizontal axis and the F-measure showing the detection performance on the vertical axis.

Due to constraints of computing resources, we used linear kernel as a kernel function for this observation.

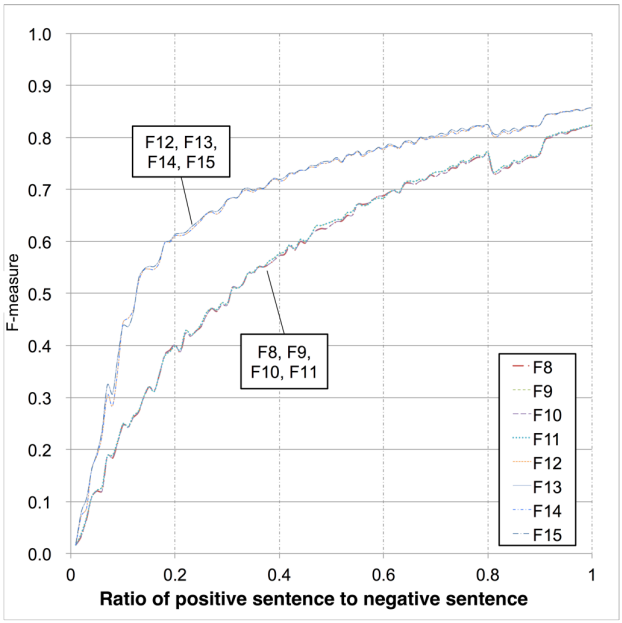


Figure 3: A plot of F-measure scores to varied sizes of positive sentences for each feature set.

FS	(a)	(b)	(c)	(d)	FS	(a)	(b)	(c)	(d)
F1	-	-	✓	-	F9	-		✓	✓
F2	-	✓	-	-	F10	-	✓	-	✓
F3	-	✓	✓	-	F11	-	✓	✓	✓
F4	✓	-	-	-	F12	✓	-	-	✓
F5	✓	-	✓	-	F13	✓	-	✓	✓
F6	✓	✓	-	-	F14	✓	✓	-	✓
F7	✓	✓	✓	-	F15	✓	✓	✓	✓
F8	-	-	-	✓					

Table 3: Feature sets and features for each set.

Kernel	FS	F-measure	Accuracy
Linear	F15	0.858	0.857
RBF	F15	0.908	0.909

Table 4: The feature set that showed highest F-measure score for RBF and Linear kernels.

Using the same data as the sample for categorization shown in Table 1 and Section 1, we tuned parameters of the SVM and its kernels.

5. Discussion

5.1 Effectiveness of SVM with RBF Kernel

In the experiment comparing performance between RBF and Linear kernels, the results in Table 4 show that the RBF kernel has higher performance. However, the RBF kernel required a lot of time for learning. In the experiment we have conducted, using a Xeon 2.0GHz CPU with around 64GB free memory, one learning division ended in 3,753 and 2,420 seconds, respectively. On the other hand, the learning using the linear kernel took only 6.54 seconds.

Hence, it can be said that the performance with respect to the unit requirement of the computational resource becomes maximum in the case of the linear kernel. However, there still remains a strong need for a faster learning algorithm as accurate as an SVM with RBF Kernel.

5.2 Adding Rule-based Features

As shown in Figure 3, in the experiment using the linear kernel, there was no significant difference between the feature sets F12, F13, F14, F15 and F8, F9, F10, F11. The difference between F8 vs. F10, F9 vs. F11, F12 vs. F14 and F13 vs. F15 is the addition of phonological similarity feature (b). While there was a factor of the significant difference, but no significant difference was observed. The same applies to the other phonological similarity feature (c) that allows insertion of acoustic long sounds. (Compare F8 to F9, F10 to F11, F12 to F13, and F14 to F15.) In the feature (c), a significant difference was confirmed in two sets of F4 to F5 and F6 to F7, though this was not tested in the experiment in this experiment's scale. It is understood that the effects of the features (b) and (c) were not remarkable in this experiment.

On the contrary, the features showing significant differences by addition of features are (a) perfect match features (F8 to F12, F9 to F13, F10 to F14, and F11 to F15). From this fact, although there is room for improvement in the features (b) and (c), compared with the case of only lexical features (d) (feature set F8) due to addition of rule base feature (a). It can be said that the robustness against reduction of the number of positive data is improved.

6. Conclusion and Future Work

In this research, we proposed a method to detection of similar parts using phonological similarity and insertion / omission of prolonged sounds in addition to lexical feature in detection of sentences including pun, based on supervised learning by SVM. In the detection performance evaluation experiments, we confirmed the effectiveness of lexical feature, the validity of rule base feature and each case of successful detection, respectively, and showed the overall effectiveness of the proposed method alongside the phonological nature of sentences. Moreover, by using lexical feature, it was shown that the proposed method obtains constant detection performance against overlapping pun in addition to collocation type puns. However, in considering the results, there is room for improvement in rule base identification, and the dependence of lexical feature on learning data was suggested. Also, we have found that the scale around 50,000 sentences of Japanese puns is suitable for construction of its detection method.

As one of current work, we are doing precise research of more optimized detection method including the use of neural language models. We are going to do investigations of the dataset itself in order to comprehend the linguistic nature of puns quantitatively.

7. Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP17K00294.

8. Bibliographical References

Yamada, K., Asano, S., and Mononobe, H. (2012). A Study on the Relationships among Self-Efficacy, Social Skills, Assertive Communication Skill, and Coping Humor in Interpersonal Conflict. *The Japanese Journal of School Health*, 54:203--210. (In Japanese)

Miyazawa, K., Tokoyo T., Masui Y., Matsuo, N., and Kikuchi, H. (2012). Factors of Interaction in the Spoken Dialogue System with High Desire of Sustainability. *The journal of the Institute of Electronics, Information and Communication Engineers*. J95-A(1): 27-36. (In Japanese)

Takizawa, O., (1995), Several Phonemic Features of Written Puns. *Journal of Natural Language Processing* (ANLP), 2(2):3-22. (In Japanese)

Kawahara, S. and Shinohara K., (2009). The role of psychoacoustic similarity in Japanese puns: A corpus study. *Journal of Linguistics*, 45(1):111-138.

Tsukawaki, R., Fukada, H., Higuchi, M., (2011). Process effects of expression of humor on anxiety and depression. *The Japanese Journal of Experimental Social Psychology*, 51(1):43--51. (In Japanese)

Dybala, P., Ptaszynski, M., Jacek, M., Takahashi, M., Rzepka, R., and Araki, K., (2012). Multiagent system for joke generation: Humor and emotions combined in human-agent. *Journal of Ambient Intelligence and Smart Environments*, 2(1):31-38.

9. Language Resource References

Ptaszynski, M., Dybala P., Rzepka R., Araki, K., and Momouchi, Y. (2012). A Five-Billion-Word Corpus of Japanese Blogs Fully Annotated with Syntactic and Affective Information. In Proc. of LaCATODA, a workshop of the AISB/IACAP World Congress in Honour of Alan Turing, pp. 40-49, Birmingham, UK.

Araki, K., Uchida, Y., Sayama, K., and Yatsu., M. (2017). Construction and Analysis of Pun Database in Japanese. In Proc. of the 56th SIG. Language Sense processing and Engineering (SIG-LSE-B702-3), pp. 13-24, Iwate, Japan. (In Japanese)

Asahara, M., and Matsumoto, Y., (2003). IPADIC User Manual. Nara Institute of Science and Technology, Japan.

A vision-grounded dataset for predicting typical locations for verbs

Nelson Mukuze[1], **Anna Rohrbach**[3,4], **Vera Demberg**[2], **Bernt Schiele**[3]
[1]Visual Meta GmbH
[2]Saarland University
[3]Max Planck Institute for Informatics, Saarland Informatics Campus
[4]EECS, UC Berkeley
nelson.mukuze@visual-meta.com, vera@coli.uni-saarland.de, {arohrbach, schiele}@mpi-inf.mpg.de

Abstract

Information about the location of an action is often implicit in text, as humans can infer it based on common sense knowledge. Today's NLP systems however struggle with inferring information that goes beyond what is explicit in text. Selectional preference estimation based on large amounts of data provides a way to infer prototypical role fillers, but text-based systems tend to underestimate the probability of the most typical role fillers. We here present a new dataset containing thematic fit judgments for 2,000 verb/location pairs. This dataset can be used for evaluating text-based, vision-based or multimodal inference systems for the typicality of an event's location. We additionally provide three thematic fit baselines for this dataset: a state-of-the-art neural networks based thematic fit model learned from linguistic data, a model estimating typical locations based on the MSCOCO dataset and a simple combination of the systems.

Keywords: human judgments, thematic fit, vision

1. Introduction

Most of automatic language understanding today is based on information that is explicitly stated in text. However, tasks that require the ability to make additional common sense inferences are increasingly coming into focus (Levesque et al., 2011; Roemmele et al., 2011; Mostafazadeh et al., 2016). We here propose a dataset which addresses a sub-task towards this goal, i.e. inferring typical locations for a given verb. This task is related to selectional preference tasks, and taps into the kind of inferences humans make when comprehending language. The task consists of predicting a location, given other information from the text (here: the predicate), see also Baroni and Lenci (2010; Sayeed et al. (2015; Tilk et al. (2016). Given the verb *to eat*, a good thematic role filling model would for instance prefer the location *restaurant* over *office*. Previous work has found that locations are a particularly difficult role to predict (Sayeed et al., 2016). This could be due to the fact that location modifiers are often omitted in text when the location is inferable from common sense knowledge, known as reporting bias (Gordon and Van Durme, 2013; Misra et al., 2016). It is hence a logical next step to use multimodal data, and in particular, data from vision. However, there are to date no datasets which can be used to evaluate the contribution of a vision model, as the only existing location fit dataset is very small (Ferretti et al., 2001), comprising only 277 verb/location pairs, and includes many rare verbs.

We here present a new dataset based on 20,000 human judgments for a total of 2,000 verb/location pairs. The dataset was specifically constructed to evaluate the contribution of both visual and linguistic information for learning common sense knowledge about locations. It builds on MS COCO (Lin et al., 2014), a dataset of images with captions; locations for these pictures were labeled using a scene classifier trained to distinguish 365 locations of the Places365 dataset (Zhou et al., 2016). Along with the dataset, we provide performance baselines for a language-based, a vision-based and a simple multimodal model for a common sense inference task using this dataset.

2. Related work

Datasets. Datasets with human ratings on a scale of 1 (least common) to 7 (most common) for *agent* and *patient* roles were made available as part of McRae et al. (1997), Padó et al. (2009), Vandekerckhove et al. (2009). Ferretti et al. (2001) created a dataset of 277 *location* ratings (**Ferretti-Loc**) using questions like "How common is it for someone to *eat* at the following locations"? 40 participants provided ratings on a 7 point scale, e.g.: eat/restaurant: 7. For their study, Ferretti et al. (2001) chose 40 transitive verbs. Many highly frequent verbs like *move* were not included in their study due to the selection criterion they used (the verb should activate a "distinct prototype"), and hence performance on such verbs was not evaluated. Our proposed dataset includes the 100 most frequent verbs in MS COCO image captions (Lin et al., 2014), and is hence intended to be more representative, and makes it possible to evaluate multimodal systems on the task.

Thematic role filling. Many thematic role filling models have been proposed (Baroni and Lenci, 2010; Baroni et al., 2014; Greenberg et al., 2015; Lenci, 2011; Sayeed and Demberg, 2014). Typically, they are evaluated by computing a correlation to human judgments, as motivated by Padó et al. (2009). A related line of work focuses on selectional preference estimation (Erk, 2007; Van de Cruys, 2014). Recently, Tilk et al. (2016) proposed a neural network model for thematic role filling. The system distributes probability over the possible role fillers of specific missing roles. Their model learns the interactions between different roles and achieves state-of-the-art performance on multiple datasets. As a baseline for this dataset, we use their best model, denoted as **Language baseline**. The **Vision baseline** system presented in this work leverages visual information from the captioned images as opposed to purely relying on text.

Vision helps language. Language and vision are the two primary human communication channels. They have a lot of complementary information, which was successfully used by prior work. Regneri et al. (2013) showed that video features help to improve the similarity estimation for sentences describing actions. Yatskar et al. (2016a) relied on images to extract common sense knowledge about objects and spatial relations between them. Tandon et al. (2016) used image tags to learn the *part-of* relation between objects. Yatskar et al. (2016b) proposed a dataset of images and "situations" (a verb with e.g. agent, tool), based on FrameNet (Baker et al., 1998). Prior work has shown that vision can benefit various linguistic similarity tasks (Bruni et al., 2014; Silberer and Lapata, 2014). For such work, it is however a crucial precondition to have a dataset on which system performance can be meaningfully evaluated. Hence, the dataset proposed in this work allows to study the impact of vision for the task of thematic role filling.

3. Dataset

In order to be able to measure progress on the task of learning typical locations of a given event (here approximated via a verb), it is necessary to have a dataset that contains a representative set of verbs and locations together with judgments of how typical these locations are. In order to enable evaluations that correlate automatic fit estimates with judgments, it is particularly important to create the dataset such that there is a good range of well fitting to badly fitting verb/location pairs.

Verb/Location pairs selection. As the existing evaluation set by (Ferretti et al., 2001) is quite heavily biased against verbs that can be grounded in pictures, we here chose to base our corpus on the 100 most frequent verbs found in the captions of the MS COCO (Common Objects in Context) (Lin et al., 2014) dataset (training/validation sets). The MS COCO dataset contains 123,287 images of people, animals and other objects "in context", i.e. in realistic environments. For each image, five captions are provided (see Figure 1).

As verbs are hard to recognize automatically from an image, the availability of captions, together with the situational context of the images, makes this corpus a good choice for studying relations between verbs and locations. We processed the available captions with the Natural Language Toolkit (NLTK) (Bird, 2006) to extract the verbs. Each of the extracted 100 most frequent verbs occurred in the captions of at least 100 images. For each of those images, we ran a state-of-the-art neural network based scene classifier, ResNet (He et al., 2016), trained to distinguish 365 locations of the Places365 dataset (Zhou et al., 2016)[1]. The classifier applies a Softmax function to the last fully connected layer representation distributing a probability of 1 over all 365 locations. Figure 1 provides an example MSCOCO image with associated captions, that e.g. mention the verbs *to sit* and *to work*, as well as the top 5 predicted locations from the ResNet classifier. In order to identify common likely locations for each verb, for each image,



Top-5 predicted locations:
Office
Home office
Waiting room
Reception
Computer room

MSCOCO Captions:
1. A man in a tuxedo working on a laptop.
2. There is a man sitting on a two seat bench in a tuxedo with a laptop in front of him.
3. A man in a suit using his laptop.
4. A man in formal attire sitting on bench using laptop computer.
5. A man in a tuxedo sits at a table and uses a laptop.

Figure 1: Example MSCOCO image with 5 captions and top-5 predicted locations from the ResNet classifier.

with at least two captions containing the verb, we extracted the top 5 predicted locations. This resulted for each verb in a list of locations, out of which some occur very frequently, while others are only seen once.

In order to obtain a balanced dataset, from the list of predicted locations per verb, we selected a subset of these by randomly choosing 4 top ranked (from the top 10), 2 middle ranked (from rank 11 to 20), and 4 low ranked (from rank 21 onwards) locations. The ranking of a location for a particular verb was based on increasing average probability (predicted score) of the location across all images relevant for that verb (see Equation 1). Next we selected a set of locations present in language data for the same set of 100 verbs as follows. From a chosen vocabulary of 50,000 most frequent words in the ukWaC corpus (Ferraresi et al., 2008), we selected candidate words, labeled as locations by the SENNA role labeler (Collobert and Weston, 2007), similar to Tilk et al. (2016), and excluded proper names using the Stanford Named Entity Recognizer (Finkel et al., 2005). We manually removed all non-physical locations (e.g. the Web) from the resulting set, which resulted in a total set of 423 candidate physical locations. We again selected for each verb 4 likely, 2 middle ranked and 4 unlikely locations, relying on the probabilities from the best model of Tilk et al. (2016). Overall, this resulted in a total of 2,000 verb/location pairs (100 verbs, with about 10 locations selected based on the visual domain and about 10 locations selected based on the language system).

Some of the most frequent verbs from MS COCO, which were included in our proposed dataset, are: *move*, *open*, *ride*, *walk*, *dress*, *eat*, *wait*, *serve* etc. Compared to FerrettiLoc, the verbs in the proposed dataset activate locations

[1]The ResNet classifier achieves a top-5 accuracy of 85% on the test set of Places365 dataset.

with little overlap in features. For example one could *walk* at various places which have little in common (e.g. at a beach, at home, in an office, in a restaurant, at a plaza). Additionally there was no assumption made about the sense of the verb for polysemous verbs like *serve* (*serving food*, *serving the ball during a tennis or volley ball match*). This choice of verbs made the locations per verb in our dataset more varied than those in Ferretti-Loc. Last but not the least, the locations in our dataset include general locations like *office*, *house*, *park* as well as more specific locations like *home office*, *courthouse*, *amusement park*. The vision-based subset has more fine-grained (specific) locations (from Places365 categories) and fewer general locations while the language-based subset has mainly general locations (all vocabulary words used by the language system are single words).

Human ratings. We collected 10 human judgments for each verb/location pair (20,000 judgments in total) via crowd-sourcing on Amazon Mechanical Turk using the LingoTurk software (Pusse et al., 2016). We provided Turkers with a verb V and location L and asked them: *Assume "X" (X can be a human, an animal or an object) is doing V, how common is it to happen at location L?* We followed the rating strategy of Ferretti et al. (2001), so the Turkers could rate on a scale from 1 ("extremely uncommon") to 7 ("extremely common"), and additionally introduced option 0 for "impossible". The rating "impossible" was chosen 1,227 times out of the 20,000 judgments, with the majority of these ratings falling on a set of 130 verb/location pairs including e.g. *graze/flat*, *dock/airplane cabin*. As some locations are rare words that may not be familiar to all Turkers, we allowed them to indicate whether they were unsure about any words in the question. This option was selected for 2% of ratings; exclusion of these ratings did not change results. We found that on average 46% of ratings (around 5 out of 10) for a given verb/location pair were identical. Standard deviations for verb/location ratings varied from 0.0 to 3.14, with the mean standard deviation being 1.55. Example ratings, averaged over 10 humans, for the verb *play* are shown in Table 1. As expected, the locations such as a *soccer field* or a *playground* are rated higher than e.g. a *repair shop* or a *hangar*.

Dataset statistics. Figure 2 presents a distribution of obtained human ratings over 2,000 verb/location pairs. We first average 10 human ratings for each pair, and then show how many verb/location pairs fall into each rating "category". We see that all levels of "fitness" are covered, from 0 ("impossible) to 7 ("extremely common"), as well as the intermediate levels. We also analyze agreement within the human ratings. Human judgments correlate with each other at Spearman's $\rho = 0.63$, which reflects the difficulty of the task. This number was obtained by computing the correlation between an individual rater vs. the average of the other 9. We repeated this for each of 10 raters and averaged the obtained correlations; the standard deviation was 0.02.

Table 2 highlights the key differences between our proposed dataset and Ferretti-Loc (Ferretti et al., 2001): the size (the higher number of verbs and verb/location pairs) and the fact that our verbs can be grounded in images. Our dataset is publicly available[2].

Location	Rating
Cockpit	0.6
Repair shop	1.8
Hangar	1.9
Landing deck	2.2
Market	2.3
Fishpond	3.6
Hall	4.9
Porch	5.4
Room	5.7
Street	5.8
Martial arts gym	6.0
School	6.0
Music studio	6.3
House	6.6
Field	6.7
Playground	6.9
Basketball court	6.9
Soccer stadium	7.0
Soccer field	7.0
Football stadium	7.0

Table 1: Example ratings, averaged over 10 humans, for the verb *play* and respective 20 locations, on a scale from 0 ("impossible") to 7 ("extremely common").

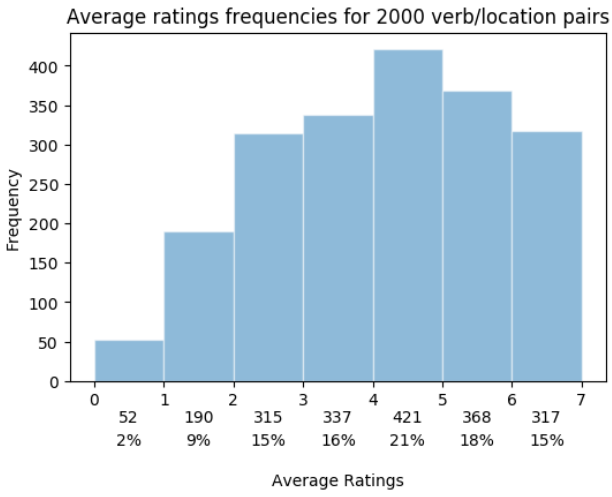


Figure 2: Histogram of the distribution of ratings, averaged over 10 humans, for all the 2,000 verb/location pairs in our dataset.

Use cases. In Section 2. we stated that our dataset allows to study the impact of vision for the task of thematic role filling, or typicality estimation. Here we discuss some of the cases where verb/location typicality can in its turn aid vision tasks. Specifically, consider the tasks of activity and scene recognition. If the visual classifiers are noisy, typicality ratings can serve as a post hoc verification to filter out untrustworthy predictions (e.g. it is unlikely to *ski* on an *ice skating rink*). If the visual classifiers are reliable

[2] *http://datasets.d2.mpi-inf.mpg.de/arohrbach/datasetV1.csv* The columns in the .csv correspond to id, verb, location and an average human rating.

Dataset	#Verbs	#Verb/Location Pairs	Vision Grounded
Ferretti-Loc	40	277	no
Our	100	2,000	yes

Table 2: Comparison of our dataset and Ferretti-Loc.

and we trust their predictions, typicality estimates could be used for anomaly detection. For instance, if we confidently recognize an action "eat" at the location "synagogue", we may decide that it is worth reporting when generating a textual description of the scene. Another way of using typicality ratings would be to incorporate them in the model at training time when learning to predict activities and locations jointly (e.g., for situational recognition; (Yatskar et al., 2016b)). We leave the experimental validation of these use cases to future work.

4. Baseline systems for location typicality estimation

We have built our dataset in a way that allows us to study whether visual information can help to improve location typicality estimates over language-only models. In this section we introduce three baseline systems which we benchmark on our proposed dataset.

Vision baseline. To estimate the probability of a location *L* given a verb *V* we rely on the scene classifier predictions (see also Section 3.). For each image where *V* occurs in at least two of its captions, we obtain the probability for the location *L* from the scene classifier, and then average probabilities across all images relevant for that verb, to obtain an estimate of how prototypical *L* is given *V*:

$$P(L|V) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\{cat=L\}} \circ score_i, \qquad (1)$$

where cat is a category in Places365, N is the number of images where *V* occurs in at least two captions, $\circ$ is the element-wise vector product, and $score_i$ is the vector output of the Softmax function.

Language baseline. As a baseline for a system trained on language data, we use the current state of the art model for this task, Tilk et al. (2016). To mitigate the domain shift between MS COCO captions and the ukWaC corpus, we used the domain adaptation method of Chelba and Acero (2006). The model gained 2% in performance from this domain adaptation. Domain adaptation benefited in particular animal related locations which are highly frequent in the MS COCO dataset (e.g., *pasture*, *corral*).

Simple multimodal baseline. We also build a **"Vision+Language"** system, which averages the probabilities from both systems, to study their complementarity.

Results. Table 3 shows the correlations between average human judgments and each of the three baseline systems' predictions. Whenever a system failed to make an estimate on the full dataset (because the relevant target location was not part of its vocabulary), we used the average probability as a generic value. This allowed us to estimate performance on the complete dataset. To make a detailed analysis of the baseline systems, we divided the dataset into three subsets: a subset with 1,353 verb/location pairs for which a language system can estimate fit (**"Ratings-language"**), a subset with 1,311 pairs for which a vision system can make predictions (**"Ratings-vision"**), and an overlapping set of 664 verb/location pairs (**"Ratings-overlap"**) for which both systems can make predictions. We can see that the multimodal system consistently improves over the unimodal systems (on **"Ratings-all"** and **"Ratings-overlap"**). A substantial gap to human performance remains.

Dataset **Ratings**	# ratings	Vis. sys.	Lang. sys.	Vis.+Lang. sys.	Human corr.
-all	2,000	0.28	0.25	0.32	0.63
-vision	1,311	0.36	-	-	0.63
-language	1,353	-	0.30	-	0.63
-overlap	664	0.41	0.27	0.43	0.64

Table 3: Spearman's ρ correlation between mean human ratings and systems' estimates, on our dataset, see text for details.

System	all	overlap-vis
Baroni & Lenci, 2010	0.23	
Greenberg et al., 2015	0.29	
Language	0.44	0.50
Vision	0.30	0.42
Vision+Lang	0.47	0.54

Table 4: Spearman's ρ correlation between mean human ratings and systems' probabilities on Ferretti-Loc dataset. The language system used here is identical to Tilk et al., 2016.

Error analysis. We further analyzed our results to see where the complementarity between the language and vision systems comes from. A typical pattern we observed was that there are cases where human ratings are high but language predictions are low, confirming our initial hypothesis that language-only models may down-rate common locations. Some examples of such cases (shown here with the average human rating) are: *cook/delicatessen: 5.7*, *ride/street: 6.4*, *display/supermarket: 6.5* or *graze/farm: 6.9*. In these cases, the vision system provides valuable complementary information.

A common failure case for the vision system is related to a lack in abstraction: more general locations, e.g. *house* are underrated compared to more specific ones which are present in the set of target locations, e.g., *beach house*. In the joint system, the language system can sometimes help to recover from such errors, e.g. for *sleep/house: 6.0*.

Another type of error is when the "Vision+Language" system predicts a high score, but human ratings are low. E.g. *fly/soccer field* is rated low by humans, while the "Vision+Language" rates it high due to a large support in images and captions mentioning the *soccer ball flying*. It appears that the human raters are biased towards animate sub-

jects (or a more agentive interpretation of flying) so they assigned a low rating to these cases.

Results on Ferretti-Loc dataset. While the previous experiment on our own dataset has served as a proof of concept, providing evidence that the vision system is complementary to the language system, and that the combination of the two may improve performance, we now proceed to test whether this finding holds up for a previously used dataset for this task. Table 4 presents an evaluation on the Ferretti-Loc dataset. We see that the "Vision system" performs well on this general dataset, and that the joint "Vision+Language" system consistently improves over the language-only baseline. Note that only 29 (overlap-vis) out of 40 (all) verbs in this dataset occur in MS COCO captions, and only 9 of them occur frequently. We again used average location probability on MS COCO for out of vocabulary pairs. The "Vision+Language" system relies on averaged probabilities when vision predictions are available, otherwise it uses the language probabilities. On the "overlap-vis", both systems can make predictions. These results provide further support for the idea that the vision and language systems are complementary.

5. Conclusions

We have presented a new dataset for thematic role filling, targeting the *location* role, which is significantly larger than the prior work (Ferretti et al., 2001). Our data collection relies on image captions and visual scene classifiers, and allows for different types of approaches to be evaluated and compared. We show three different baselines for the task of predicting typical locations for a verb. Our experiments support the hypothesis that the visual scene probabilities provide useful cues for typical location prediction, and are complementary to the language estimates. The multimodal baseline performs substantially better than the unimodal baselines.

6. Bibliographical References

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The berkeley framenet project. In *ACL*, pages 86–90.

Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL*, pages 238–247.

Bird, S. (2006). Nltk: the natural language toolkit. In *COLING/ACL on Interactive presentation sessions*, pages 69–72.

Bruni, E., Tran, N.-K., and Baroni, M. (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research (JAIR)*, 49(2014):1–47.

Chelba, C. and Acero, A. (2006). Adaptation of maximum entropy capitalizer: Little data can help a lot. *Computer Speech and Language*, 20(4):382–399.

Collobert, R. and Weston, J. (2007). Fast semantic extraction using a novel neural network architecture. In *ACL*, volume 45, page 560.

Erk, K. (2007). A simple, similarity-based model for selectional preferences. In *ACL*, volume 45, page 216.

Ferretti, T. R., McRae, K., and Hatherell, A. (2001). Integrating verbs, situation schemas, and thematic role concepts. *Journal of Memory and Language*, 44(4):516–547.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*, pages 363–370.

Gordon, J. and Van Durme, B. (2013). Reporting bias and knowledge acquisition. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 25–30. ACM.

Greenberg, C., Sayeed, A. B., and Demberg, V. (2015). Improving unsupervised vector-space thematic fit evaluation via role-filler prototype clustering. In *HLT-NAACL*, pages 21–31.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE CVPR*.

Lenci, A. (2011). Composing and updating verb argument expectations: A distributional semantic model. In *2nd Workshop on Cognitive Modeling and Computational Linguistics*, pages 58–66.

Levesque, H. J., Davis, E., and Morgenstern, L. (2011). The winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, volume 46, page 47.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *ECCV*, pages 740–755.

McRae, K., Ferretti, and Liane Amyote, T. R. (1997). Thematic roles as verb-specific concepts. *Language and cognitive processes*, 12(2-3):137–176.

Misra, I., Lawrence Zitnick, C., Mitchell, M., and Girshick, R. (2016). Seeing through the human reporting bias: Visual classifiers from noisy human-centric labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2939.

Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., Kohli, P., and Allen, J. (2016). A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of NAACL-HLT*, pages 839–849.

Padó, U., Crocker, M. W., and Keller, F. (2009). A probabilistic model of semantic plausibility in sentence processing. *Cognitive Science*, 33(5):794–838.

Pusse, F., Sayeed, A., and Demberg, V. (2016). Lingoturk: managing crowdsourced tasks for psycholinguistics. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 57–61.

Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., and Pinkal, M. (2013). Grounding action descriptions in videos. *TACL*, 1:25–36.

Roemmele, M., Bejan, C. A., and Gordon, A. S. (2011). Choice of plausible alternatives: An evaluation of com-

monsense causal reasoning. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Sayeed, A. and Demberg, V. (2014). Combining unsupervised syntactic and semantic models of thematic fit. In *First Italian Conference on Computational Linguistics (CLiC-it 2014)*.

Sayeed, A., Demberg, V., and Shkadzko, P. (2015). An exploration of semantic features in an unsupervised thematic fit evaluation framework. *Italian Journal of Computational Linguistics*, 1(1).

Sayeed, A., Greenberg, C., and Demberg, V. (2016). Thematic fit evaluation: an aspect of selectional preferences. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 99–105.

Silberer, C. and Lapata, M. (2014). Learning grounded meaning representations with autoencoders. In *ACL*, pages 721–732.

Tandon, N., Hariman, C., Urbani, J., Rohrbach, A., Rohrbach, M., and Weikum, G. (2016). Commonsense in parts: Mining part-whole relations from the web and image tags. In *AAAI*, pages 243–250.

Tilk, O., Demberg, V., Sayeed, A. B., Klakow, D., and Thater, S. (2016). Event participant modelling with neural networks. In *EMNLP*.

Van de Cruys, T. (2014). A neural network approach to selectional preference acquisition. In *EMNLP*, pages 26–35.

Vandekerckhove, B., Sandra, D., and Daelemans, W. (2009). A robust and extensible exemplar-based model of thematic fit. In *EACL*, pages 826–834.

Yatskar, M., Ordonez, V., and Farhadi, A. (2016a). Stating the obvious: Extracting visual common sense knowledge. In *NAACL-HLT*, pages 193–198.

Yatskar, M., Zettlemoyer, L., and Farhadi, A. (2016b). Situation recognition: Visual semantic role labeling for image understanding. In *CVPR*, pages 5534–5542.

Zhou, B., Khosla, A., Lapedriza, A., Torralba, A., and Oliva, A. (2016). Places2: A large-scale database for scene understanding. *arXiv:1610.02055*.

7. Language Resource References

Ferraresi, A., Zanchetta, E., Baroni, M., and Bernardini, S. (2008). Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Web as Corpus Workshop (WAC-4) Can we beat Google*, pages 47–54.

Ferretti, T. R., McRae, K., and Hatherell, A. (2001). Integrating verbs, situation schemas, and thematic role concepts. *Journal of Memory and Language*, 44(4):516–547.

Creating Dialect Sub-corpora by Clustering: a case in Japanese for an adaptive method

Yo Sato, Kevin Heffernan
Satoama Language Services, Kwansei Gakuin University
Kingston-upon-Thames, U.K., Sanda, Hyogo, Japan
yo@satoama.co.uk, kevin@kwansei.ac.uk

Abstract

We propose a pipeline through which to derive clusters of dialects, given a mixed corpus composed of different dialects, when their standard counterpart is sufficiently resourced. The test case is Japanese, where the written standard language is sufficiently equipped with adequate resources. Our method starts by detecting non-standard contents first, and then clusters what is deemed dialectal. We report the results on the clustering of mixed Twitter corpus into four dialects (Kansai, Tohoku, Chugoku and Kyushu).

Keywords: dialect, Japanese, clustering, adaptation of language model

1. Introduction

In what follows we propose a pipeline through which to derive clusters for dialects, given a body of 'mixed' corpus composed of different dialects, when one of them is sufficiently resourced. The main component of the proposal is an unsupervised clustering method to generate dialectal sub-corpora. The test case we experiment on is Japanese, where the variety deemed 'standard', the Tokyo dialect, is equipped with an adequate language model. We call such a dominant and well-resourced language a 'pivot' language, and its model a pivot model. Our main target data are Twitter utterances, coming from four broad dialect-speaking regions of Japan outside the Tokyo area. We will show how a) the dialectal content can be identified in this dataset and b) the identified dialectal content can be classified into the four dialects.

While our experiments are specifically on Japanese, we believe the work will have a wider implication, since a similar situation exists in many linguistic communities, where a well-resourced pivot exists but dialects lack mature enough language models. Ironically enough this is despite the fact that the availability of dialect data has increased in the cyberspace, with the rise of social media and interactive message boards. It is principally the lack of classified data that prevents the data from being utilised.

Japanese poses an additional challenge since there is no word segmentation in its orthography. This challenge however is a *general* problem. Word segmentation is an artefact of orthography, which is not present in speech, and hence requires lexical knowledge to perform. Thus the challenge is similar to the situation faced by a dialect monolingual who is exposed to other dialects in a *spoken* form: the difficulty is not so much with the unknown 'words' as the incomprehensible 'chunks' that may or may not correspond to words.

Our proposed procedure consists of three main stages. At the first stage we set apart the portion that is deviant in the view of the word pivot model. We find that this separated part not only contains dialects but non-dialectal non-standard utterances (e.g. internet jargon), so a second filtering follows, to set apart genuinely dialectal sentences. Here, in contrast to the prior stage, the *character* pivot model is used for filtering. The final stage consists of clustering, performed on what remains, where the technique employed is a modified form of divisive hierarchical clustering.

2. Related work

The present work may be categorised into the domain of 'discrimination of similar languages', which has attracted attention in recent years. Discrimination tasks have been tackled for variant sets such as South-Asian languages (Ranaivo-Malançon, 2006), English varieties (Lui and Cook, 2013) and Arabic (Zaidan and Callison-Burch, 2013). Work targeting dialects in the usual sense also exists, and Vergez-Couret and Urieli (2014) use what we call a pivot model in their study. Predominantly the techniques used in this domain are supervised, but Scherrer (2014) is a notable exception we find particularly relevant, in its unsupervised approach with pivot language models.

Another genre of studies from which we draw inspiration is unsupervised clustering that uses a distance metric. Nagata (2014) uses the distance *between* language models for clustering various types of English, though our notion pertains to the distance of a sentence *from* a language model. This notion, distance of a sentence from a certain model, has come to be used in corpus linguistics, where one might want to quantify the degree of difference from some existent model e.g. Collins et al. (2014) as a 'colloquiality' measure, Chen (2016) for proximity of textbooks to naturally occurring corpora.

Another line of research worth mentioning in the context of 'deviation from the standard' concerns normalisation. Research of this kind tries to deal with 'noisiness' by trying to find the 'standard' forms, faced with noisy corpora like Twitter. From a purely technical perspective our work also addresses the issue of 'deviation from the standard' faced with nonstandard data.

Like the present work Han et al. (2011) propose a 'pipeline' of three stages, and their initial stages also pertain to filtering non-standard content. With a similar goal (normalisation), Saito et al. (2014) introduce additionally a character-lattice based learning for non-segmented (Japanese Twitter) data, so this approach can be compared to ours, though theirs is a supervised method.

3. Data

We use two sets of data, controlled and natural. The first is the parallel dialect corpus recently published (Parallel Speech Corpora of Japanese Dialects, Yoshino et al. (2016) henceforth PCJD), with four sets of sentences that each represent a dialect (Tohoku, Kansai, Chugoku and Kyushu). Each set consists in turn of five sets of 100 sentences, the translations by five native speakers of the dialects, of the Tokyo dialect equivalents. Although the quantity of data is insufficient for training purposes, this dataset provides useful sources for evaluation.

Our main dataset is crawled Twitter data, obtained from the social media's public API (Twitter Inc., present) for the four-month period between February and May 2017, which amount to about 280 thousand sentences altogether after cleaning.[1] They were collected by setting geographical locations to the five areas represented in PCJD. Presumably due to the tweeting population difference, we have a slight disproportion: Tokyo roughly accounts for 31%, Kansai 28%, Kyushu 21%, Chugoku 16% and Tohoku 14%.

The data were then processed as follows. First, the whole corpora were processed with MeCab, a Japanese morphological analyser (Kudo et al., 2004). MeCab assigns each sentence a probability score, based on a CRF-trained model. With its training utility, we created our own model based on the written newspaper corpus (Kawahara et al., 2002; Mainichi Shimbun, 1995). We used the default dictionary that comes with the tool with nine features (Japan Information-Technology Promotion Agency, 1995). This constitutes our word pivot model, the basis for the first stage filtering.

Using roughly the same training method, we also built a pivot *character* model, this time using the Tokyo-area part of our crawled Twitter. Unlike the word model, of which it would be difficult to create reliable annotation, a character model can be built without much manual effort, and hence could be made easily available.

In short, we have made available the pivot models of standard Japanese in two types, word (newspaper) and character (Twitter). The target data is the Twitter data coming from four dialect-speaking regions. The goal is to create four sub-corpora of these dialects, by first detecting dialectal contents, and then, clustering them.

4. Method

In the following subsections we describe our three stages to separate and cluster dialectal utterances the Twitter data in more details.

4.1. Stage 1: Separation of non-conformant sentences

At this initial stage we separate the sentences which do not conform to the word pivot model. Since this CRF model outputs a probability given its dictionary items and their feature constellation in context, non-congruant items usually manifest themselves as their low probability, in one of the following two ways, faced with out-of-vocabulary (OOV) items. First, it could force some alternative word assignment in what could be called the forced-alignment strategy, finding an alternative segmentation path, at the likely cost of contextual probabilities. For example, for the sentence

そんなんありえへん家計ピンチやし

our word pivot model finds for the position of へん, in place of the correct label, dialectal negative auxiliary, an adjectival stem meaning 'strange'. It also labels やし, correctly two dialectal auxiliaries, as a noun (name of a fruit). As a result of forcing out-of-context words, the contextual probability suffers, in relation to our 'expected' probability.

Second, it might choose to, or be forced to, abandon the dictionary search, leaving the OOVs as unknown. In this case, the model invokes a user-defined OOV model. We adopt a scheme dependent primarily on the length,[2] giving more weights to the average length of (known) word tokens (approximately 3.2 syllables). We gave apriori initial values to OOV items, depending on its syntactic categories and adjusted experimentally, starting from the default values of the parameterised dictionary that came with the tool.

We use the concept of *distance from the pivot model* in this, as well as the following, stage as the filtering criteria, which is the difference in probability from the mean probability of the training data, that is, given the model $\mathbb{M}$ and the mean probability of the training data of $\mathbb{M}$, $E(P(s_{\mathbb{M}}))$

$$Dist(s, \mathbb{M}) = P(s)^{\mathbb{M}} - E(P(s_{\mathbb{M}})) \quad (1)$$

To determine what 'low' probability is low enough to be marked as non-conformant, we started from the mean distance we obtained for the Twitter corpus against the pivot model, and gradually lower it to optimise.

[1] We excluded utterances unsuitable for language model building, such as sentences that are heavily duplicated, are short (we set the minimum of 10 characters), consist only of punctuations, emojis and onomatopoeia.

[2] We also used the character type for assigning weights, given three types of alphabets (hiragana, katakana and kanji) in Japanese, preferring single-type sequences over mixed-type sequences.

With the threshold thus set, 41% of the data are classified as non-conformant overall.[3] For evaluation, with 100 sentences randomly extracted from the data coming from each region (i.e. 400 in total), we asked four native speakers, one each from our four regions, to check all the 400 sentences.[4] Each volunteer judged whether each sentence belongs to their dialect or not. The result is shown in the table below.

		Regional breakdown			
	All	Tohoku	Kansai	Chugoku	Kyushu
Utt. count	400	100	100	100	100
Matches	323	68	91	83	81
Other dialect	19	2	5	8	4
Nondialect	58	29	3	11	15
Precision	.808	.780	.940	.890	.910
Recall	.946	.994	.973	.978	.988

Table 1: Evaluation of non-conformity separation

The third row in the table shows two types of non-matches, ones belonging to one of our other three dialects and non-dialect. As can be seen, while we have achieved a reasonable success in setting apart the nonstandard portion, an obvious remaining problem is that this portion still contains a large proportion of non-dialect sentences. This is likely due to the fact that our pivot model is based on newspaper data. Therefore typical 'internet-speak' with jargon and contracted forms, such as 'ワロタ' or 'mjd', are frequently observed but mark unsurprisingly a low score, on par with the level of distinctly dialectal utterances.

4.2. Stage 2: Differentiating dialectal and non-dialectal deviations

At the second stage, we attempt to further filter the non-conformant sentences down to genuinely dialectal sentences. In order to do this, we use the same notion as the first stage, i.e. distance from the pivot model, but used the character model as a pivot. We also limit our target to substrings that are low in probability, or *anomalous segments* (ASs). By this we mean the segments, classified as either OOVs or known words, which are assigned a low probability. Again we adopted the simple threshold of the mean distance against the pivot model. The rationale behind this strategy is this: albeit both 'unknown' with respect to the pivot dictionary, the difference between dialectal and non-dialectal strings should manifest itself in the character-level patterns.

Our character pivot model is a three-dimensionality CRF model. The features used are the vowel/consonant types and 'voicedness', important elements that determine the character constellation of Japanese.[5] The distance from the pivot model is computed as in Equation 1, though now on the character model. We take the average probability of all the strings to be the 'expected' value, and consider the difference of the target substring to this expected value to be the distance from the pivot (character) model. The threshold for non-conformity has been determined experimentally.

Evaluation of the results at this stage has been done in two ways, without further assistance of our volunteers. First, to see its effect directly and allow comparison, we first inspected how many of the remaining (non-dialectal) sentences have been removed, with the same set of 400 sentences as before and recalculated the precision. The improvement figures are shown in Table 2. As can be seen, now approximately half of the non-dialectal sentences go away, bringing the precisions to a respectable level. Additionally, given the small size of this part of the results, one of the authors, a Tokyo dialect speaker, inspected results of 200 randomly picked sentences to see how many non-dialectal, or pivot dialect, sentences remain. We find 18, which is comparable to the results we obtained for the 400 set.

	Reduction nondialect	Improvement precision
Tohoku	$29 \rightarrow 19$	$0.68 \rightarrow 0.78$
Kansai	$3 \rightarrow 1$	$0.91 \rightarrow 0.98$
Chugoku	$11 \rightarrow 4$	$0.91 \rightarrow 0.95$
Kyushu	$15 \rightarrow 8$	$0.81 \rightarrow 0.91$
Overall	$58 \rightarrow 32$	$0.81 \rightarrow 0.91$

Table 2: Improvements at second filtering for dialect

4.3. Stage 3: Clustering

In our third and final stage, we cluster our likely 'dialectal' portion. We use a hierarchical clustering method, a top-down variety called DIANA (Kaufmann and Rousseeuw, 1990). We again require a character-based distance given the lack of dialect lexicon, though this time between sentences. The basis for our metric is Levenshtein distance (LD), which captures a major characteristic of the membership to the same dialect, i.e. sharing of non-standard lexical items, by restricting the target, as in 4.2., to the anomalous segments (ASs). We use two sets of ASs here, both character- and word-levels, in order to capture the anomaly of syntactic/contextual type on the first level, and that of phonotactic type on the second. Our *per sentence Levenshtein distance (LD_{sent})*, then, is as follows. Let AS_s and AS_l, be anomalous segment sets, then:

$$LD_{sent} = \sum_{i \in AS_s, j \in AS_l} \underset{i}{argmax} LD(i,j)/|AS_s| \quad (2)$$

[3] There was a significant difference in terms of the proportion of non-conformity across regions, however, with 34% for Tohoku, 52% for Kansai, 45% for Kyushu and 40% for Chugoku. See Sato and Heffernan (2017) for details.

[4] Later another native speaker of each region was asked to validate the prior volunteer's judgements and excluded the ones with disagreement.

[5] The surface N-gram model does not suffice for Japanese because Japanese characters mostly consist of a combination of a consonant and a vowel and surface forms may obscure a similarity in sound between two characters.

	Tohoku	Kansai	Chugoku	Kyushu	Recall
Tohoku	55	2	5	3	.846
Kansai	3	67	8	6	.798
Chugoku	2	10	59	11	.719
Kyushu	9	7	8	57	.703
Precision	.797	.779	.738	.740	

	Tohoku	Kansai	Chugoku	Kyushu	Recall
Tohoku	86	2	6	5	.868
Kansai	0	68	13	9	.756
Chugoku	9	5	66	12	.717
Kyushu	4	8	9	73	.776
Precision	.869	.819	.702	.759	

Table 3: Clustering performance, Twitter and PCJD

which is essentially the average of the LDs between the segments in the smaller set and their closest counterparts in the larger set.

However, this proves insufficient due to the fact that there are lexical items which, while not found in the pivot lexicon, belong to more than one dialect, particularly (not surprisingly) between neighbouring dialects. For example, the auxiliary よる and じゃ are both used frequently in the Chugoku and Kyushu dialects. Therefore, an entirely bottom-up procedure of clustering (such as agglomerative clustering) could lead to wrong groupings at the initial stages and cannot recover from these mistakes.

We have therefore opted to have a top-down constraint incorporated, the *sequence sharing rate* or SSR. The intuition is that a dialect will have a *consistent* shared vocabulary, and hence, even if some words can happen to be shared across dialects, the substring sharing *as a whole* inside a dialect should be higher than across dialects. By shared sequence we mean a contiguous substring that is found in the target strings. We take the longest match. Therefore for example between *abcde* and *ijbcdk* it is *bcd*. We also take multiple matches if they exist but not repetitions in the same string, so for *abcdef* and *efabcef* we will have two shared sequences, *ef* and *abc*. Given a set of utterances U and a set of shared sequences that a set of shared sequences S, SSR is defined as follows:

$$SSR(U) = \sum_{s \in S} (len(s) \times 2)/|U| \quad (3)$$

where $len(s)$ refers to the number of characters in shared sequence s. Notice we give more weights, proportionate to two, to longer shared sequences, given the likelihood that longer sequences contain words and phrases, which we are implicitly modelling.

Now, DIANA essentially splits the chosen cluster into two subsets iteratively. The choice of which cluster to be split is made between iterations.[6] The binary split proceeds in such a way that the original cluster, below A, 'transfers' its 'most dissimilar' of the remaining members to a new cluster, B, with the following function for *dissimilarity distance*:

$$D_{dissim} = \frac{1}{|A|-1} \sum_{j \in A} d(i,j) - \frac{1}{|B|} \sum_{k \in B} d(i,k) \quad (4)$$

where $i, j \in A$ and $k \in B$ and d is the distance metric. $argmax_i D_{dissim}$ is the most dissimilar member to transfer, and this continues until $D_{dissim} \leq 0$.

With per-sentence Levenshtein distance as the distance metric, an iteration proceeds generally as described above. The split is then adjusted after each iteration in such a way that the two metrics, LD and SSR, are balanced. We do this by reversing some of the decisions made in the iteration. We start from the least dissimilar, or the most recent, item 'moved' from the original cluster to the new, and conduct a check towards the less dissimilar ones iteratively. The check is about whether taking the item back to the original cluster improves the SSR. If so, the previous decision is reversed and the item goes back to the original. We repeat this procedure until the point where no further improvement is likely.

In Table 3 we report the performances, for the Twitter data (filtered by our native volunteers to what they judged dialectal as the Gold Standard) and PCJD, in the form of confusion matrices. For PCJD, we used a set of 100 sentences for each of the four dialects, though we excluded a few sentences[7]. For this dataset we obtained on average precision and recall around 80% level, but have a large variance between dialects. In particular precision remains rather low for Chugoku and Kyushu, presumably due to the confusability between the dialects. The Twitter results follow roughly the same pattern, with a slightly lower average generally than PCJD. The aforementioned inconsistency between regions is less pronounced however. This can be due to the effectiveness of the top-down control, which can only kick in when the data reaches a critical mass.

5. Final remarks and future tasks

We have presented a pipeline of methods that generates clusters of dialects from a mixed corpus, on the basis of a pivot language model.

As has been said in Introduction, our method is general enough to be replicated with other languages. Although we did not emphasise it (since no effective test has been feasible), the fact that there is no restriction on the number of clusters is a great advantage for generality.

[6] Amonst the several criteria for calculating the distance, we use the 'average linkage' criterion.

[7] We excluded 19 sentences altogether out of total 400, for the following two reasons: a) ones that are close to the pivot model, and hence would have been isolated in our step 2, b) ones that are identical between regions.

A shortcoming on the other hand, related to generality of application too, is that it depends heavily on the pivot model. In our case, the pivot model was a CRF-based model, which in turn depends heavily on feature configurations. We also have had to resort to some feature engineering to make clustering work. There will likely be a trade-off between performance and amount of such feature manipulations. Future research therefore should address a more general training method.

Another major issue is evaluation, which can be labour-intensive if manually done, like we did. A small test data also means less reliability and generality in the results. It would therefore be desirable to handle Gold Standard creation more efficiently, e.g. through crowd sourcing, or devise a way for intrinsic evaluation.

The most important further goal we envisage is to create automatically the language model for each new subcorpus. We believe that as long as lexicons are similar, it is possible to Scherrer (2014)'s method, i.e. generating sub-corpus lexicons by finding equivalent words. This will enhance greatly the usefulness of the dialect corpora, and also render intrinsic evaluation feasible.

6. Bibliographical References

Chen, A. C.-H. (2016). A critical evaluation of text difficulty development in elt textbook series: A corpus-based approach using variability neighbor clustering. system. *System*.

Collins, P. A. M., Borlongan, and Yao, X. (2014). Modality in philippine English: A diachronic study. *Journal of English Linguistics*.

Han, B., Cook, P., and Baldwin, T. (2011). Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 49th Annual Meeting of ACL*.

Kaufmann, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data - An Introduction to Cluster Analysis*. John Wiley and Sons.

Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). Applying Conditional Random Field to Japanese morphological analysis. In *Proceedings of the conference on Empirical Method in Natural Language Processing*.

Lui, M. and Cook, P. (2013). Classifying English documents by national dialect. In *InProceedings of Australasian Language Tchnology Workshop*.

Nagata, R. (2014). Language family relationship preserved in non-native English. In *Proceedings of COLING 2014*.

Ranaivo-Malançon, B. (2006). Automatic identification of close languages - case study: Malay and Indonesian. *ECTI Transactions on Computer and Information Technology*.

Saito, I., Kugatsu, S., Asano, H., and Matsuo, Y. (2014). Morphological analysis for Japanese noisy text based on character-level and word-level normalization. In *Proceedings of COLING 2014*.

Sato, Y. and Heffernan, K. (2017). Quantifying 'standardness' of the language use in locality: a study with twitter data. In *Extended abstract, Corpus Linguistics International Conference 2017*.

Scherrer, Y. (2014). Unsupervised adaptation of supervised part-of-speech taggers for closely related languages. In *Proceedings of the 1st Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*.

Vergez-Couret, M. and Urieli, A. (2014). Pos-tagging different varieties of occitan with single-dialect resources. In *Proceedings of the 1st Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*.

Yoshino, K., Hirayama, N., Mori, S., Takahashi, F., Itoyama, K., and Okuno, H. G. (2016). Parallel speech corpora of Japanese dialects. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages –, Paris, France, may. European Language Resources Association (ELRA).

Zaidan, O. F. and Callison-Burch, C. (2013). Arabic dialect identification. *Computational Linguistics*.

7. Language Resource References

Japan Information-Technology Promotion Agency. (1995). *IPA dictionary*. Japan Information Technology Promotion Agency.

Daisuke Kawahara and Sadao Kurohashi and Koiti Hasida. (2002). *Kyoto University Text Corpus*. Kyoto University.

Mainichi Shimbun. (1995). *Mainichi Newspaper CD-ROM 1995*. Mainichi Newspaper.

Twitter Inc. (present). *Twitter Stream API*. Twitter Inc.

A Fast and Flexible Webinterface for Dialect Research in the Low Countries

Roeland van Hout, Nicoline van der Sijs, Erwin Komen, Henk van den Heuvel
CLS, Radboud University
Erasmusplein 1, Nijmegen, the Netherlands
{r.vanhout, n.vandersijs, h.vandenheuvel, e.komen}@let.ru.nl

Abstract

This paper describes the development of webportals with search applications built in order to make the data from the 33 volumes of the Dictionary of the Brabantic dialects (1967-2005) and the 39 volumes of the Dictionary of the Limburgian dialects (1983-2008) accessible and retrievable for both the research community and the general audience. Part of the data was available in a digital format, a larger part only in print. The printed data was semi-automatically converted from paper to structured text (database). This process allowed for streamlining information, applying (semi-)automatic data checks and manually correcting the input. Next, the resulting database was the backbone of a webportal for faceted search requests on the full collection, including filtering and splitting the results on metadata. The design and implementation of the webportals, called e-WBD and e-WLD, are being defined in more detail. The URLs of the portals are: http://e-wbd.nl/ and http://www.e-wld.nl/.

Keywords: web services; data curation; dialects

1. Introduction

The 33 volumes of the Dictionary of the Brabantic dialects (*Woordenboek van de Brabantse Dialecten*, WBD) have appeared in press between 1967 and 2005, while the 39 volumes of the Dictionary of the Limburgian dialects (*Woordenboek van de Limburgse Dialecten*, WLD) were published between 1983 and 2008. The WBD and WLD have been compiled at the Radboud University Nijmegen and at the University of Leuven and both dictionaries started under the guidance of the famous Dutch dialectologist A.A.Weijnen.[1] The Dictionary of the Flemish Dialects (*Woordenboek van de Vlaamse Dialecten*, WVD) was set up according to the same semantic principles, but started later.

The Limburgian dialects are spoken in the provinces of Limburg in the Netherlands and Belgium. The dialects are separated into six dialect areas, as shown in Figure 1.

The same goes for the Brabantic dialects: the dialects are spoken in the Dutch province Northern Brabant, the Belgian provinces Antwerp and Flemish Brabant, and the Brussels-Capital Region, as can be seen on Figure 2. More details can be found in WBD, *part III*, volume *Inleiding & Klankgeografie* (2000).

The WBD and WLD are onomasiologically organized. The main entries are semantic concepts, represented in Standard Dutch (the lemmas). These entries contain the keywords that are dutchified transliterations (normalized spellings) of the dialect forms. The next level gives the phonetic transcription forms (the dialect forms). For each transcription form the location or locations are given where they are used.

Overall, the dictionaries consist of three parts. The first part (WBD 9 volumes, including one devoted to introductory matters; WLD 13 volumes) contains the agricultural terminology. The second part (WBD 9 volumes; WLD 12 volumes), concerns the technical terminology (industries, trades), and finally the third part (WBD 15 volumes, including one devoted to introductory matters; WLD 14 volumes) enumerates the general vocabulary.

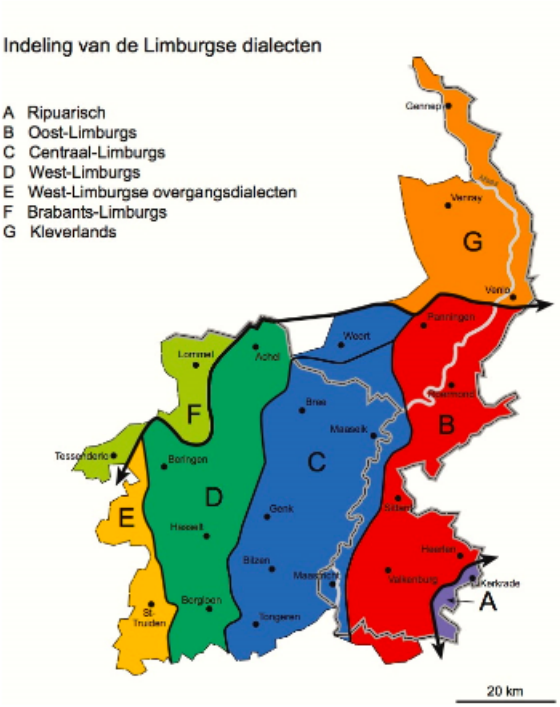


Figure 1. The six Limburgian dialect areas

Unique for Limburg was the mining industry. WLD dedicated a special volume to the vocabulary used in the 19 coal mines in the two Limburgian provinces. The 12 Dutch mines were closed in the 1960s and 1970s, and in the 1990s the 7 Flemish mines followed, making the mine jargon obsolete. In the mines a special, mixed form of dialects was

[1] A complete overview of all volumes can be found at http://www.e-wld.nl/delen and http://e-wbd.nl/delen

spoken: the mine workers came from far and wide and spoke different Limburgian dialects, but also other Dutch, Flemish, German or Walloon dialects, and there were even workers from, among others, Italy, Poland and Brazil. In Dutch mining terminology, German had the greatest influence because many workers came from Aachen and the surrounding area, while French borrowings were frequently occurring in Flemish mines due to the Walloon workers and staff.

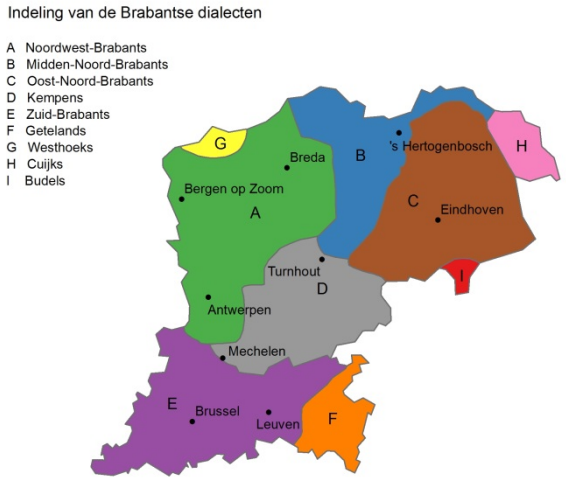


Figure 2. The nine Brabantic dialect areas

The third parts of the dictionaries were prepared by the editors on the computer, using a tailor-made version of FileMaker Pro. These parts have been made digitally available on a website in 2004, as a result of the NWO project D-Square. To this website a cartographic tool was added (Van den Heuvel et al., 2015; de Vriend et al., 2006). No cross-checks were performed on data consistency in D-square and the database only had limited search functions, accessible through a meanwhile fairly obsolete interface.

In 2015 the CLARIN-NL program granted a project called CARE: CurAtion and integration of REgional dictionaries, under supervision of Nicoline van der Sijs and Roeland van Hout. The goal of CARE was to semi-automatically convert text documents of the first two parts of the two dictionaries to structured text (database), and to combine these data with those of the third part. The input format was scanned text, read by optical character recognition (OCR). As output format the Lexical Markup Framework LMF was chosen, as it is an accepted standard in CLARIN for lexical data. In Van den Heuvel, Sanders & Van der Sijs (2016) the methodology used has been described in detail.

The resulting database made it possible to uniform all data and to check information consistency. For instance, there were quite a number of inaccuracies in the so-called Kloeke codes, either in the printed volumes or because of the OCR process. These Kloeke codes, named after the dialectologist G. Kloeke who constructed the system (Kruijsen & Van der Sijs, 2010), refer to locations in the Netherlands and Belgium in a unique way.

The special phonetic symbols turned out to be the hardest problem, particularly for the Limburgian dialects. The editors decided that extremely fine-grained phonetic distinctions between the Limburgian dialects had, for scientific reasons, to be expressed, see Figure 3. Instead of using the IPA for this purpose, they developed an exclusive phonetic script containing many diacritics. The OCR programme could not make head or tail of it, so the resulting text was gibberish. Once the text was converted into a database and all phonetic symbols were put into a single field or column, systematic semi-automatic correction became feasible. This laborious task was done by retired editor Joep Kruijsen.[2] The editors of the Brabantic dialects chose a less elaborate phonetic transcription. This too hampered computer reading, but in a less interfering way.

Along these lines we managed to greatly improve the process of data curation.

1.8.9 VARKENS FOKKEN

(N 76, 37b; monogr.)
[Zich toeleggen op de teelt van varkens.]
fokken: *fǫkə* L 266, 267, 288a, 295, 318d, 328, 371, 376, 0426, Q 16, 27, 32, 112, 117; *fokə* L 320a; **kweken:** *kwēkə* L 381, 423, Q 9; *kwīkə* L 414; *kwekə* P 219, 219a; **baggen kweken:** *bagə kwēkə* L 415, Q 11; **telen:** *tǭlə* L 318d; **vermeerderen:** *vərmiərdərə* L 295; **varkens optrekken:** *vɛrkəs optrɛkə* L 265; **trekken:** *trękə* Q 197; **tuchten:** *tsøxtə* Q 117; *tsyxtə* Q 121; **varken houden:** *vɛrkən hǭgən* Q 1; *vɛrkə hā* P 188.

Figure 3. An example of the phonetic symbols used in WLD

All resulting LMF-files were stored in open access at the Meertens Institute in Amsterdam (which is a CLARIN Data Centre).

The next step was to design and implement a webportal. The idea was that a complete, curated database with a transparent webinterface providing precise and powerful selection and search tools would open up new avenues for dialect research in the Brabant and Limburg areas, in particular because the data are there for everyone on the internet. In the following sections we describe how we dealt with the design and implementation of the

[2] These corrected data have not yet been made available through the webportal, but this will be done in the future.

webportals, now accessible at http://e-wbd.nl/ and http://www.e-wld.nl.

2. Set-up of the Webportal

Figure 4 shows the main elements of a dictionary entry as presented in the volumes.

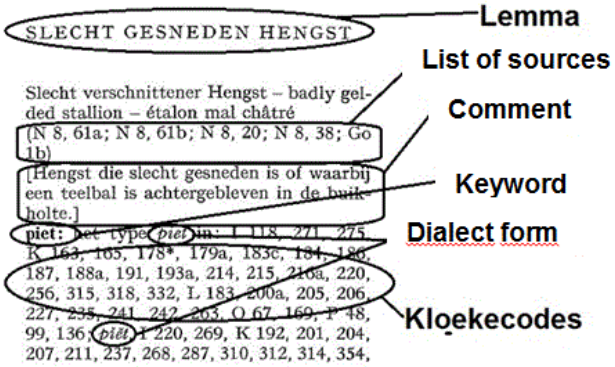


Figure 4. Example from the dialect dictionary with explanation of how it is structured.

The main components contained in the database are therefore:

- Lemma (= concept)
- Lemma comment
- Source list
- Keyword (= dialect entry in normalized spelling)
- Dialect form (= phonetic spelling of dialect form)
- Kloeke code and associated place name

At the start of the webportal project we set up a list of requirements which the portals should fulfill. The webportals should contain:

- Information about the digitization process of the paper books into a database
- A map of the Limburgian and Brabantic areas or dialects (see Fig. 1 and 2)
- Full overview of the volumes
- Access to the PDF versions of all volumes
- Overviews of all lemmas, keywords, places
- Search options at the level of lemma, keyword, place, allowing
    - Wildcards in the search terms
    - Filter options for place, Kloeke code, volume
- Hyperlinks between retrieved lemmas and corresponding keywords
- Access to the PDF files of the books via the results of the query.

3. Implementation of the Webportal

The e-WLD and e-WBD webportals are written in Python 3.5, building on the freely available Django web application framework.[3] Django is a Python-based framework that makes it easier to build web applications quickly and with less code. The code of the portal has been developed using the PTVS facilities of Microsoft Visual Studio 2013.[4] The site is served as an uwsgi application that runs under Apache on one of the webservers at Radboud University Nijmegen. All entries (close to 2 million dialect words) are stored in a Sqlite database.[5]

The map on the site's homepage, is based on the most recently available geographical data, and it shows the Belgian and Dutch dialect areas that are covered by the WLD.

The webportal allows listing the available dialect forms based on their lemma (*begrippen* in Figure 5), or on their keyword (*trefwoorden* in Figure 4). It is also possible to obtain a list of the locations (*plaatsen* in Figure 5) or (for WLD) a list of the coal mines where dialect forms have been collected.

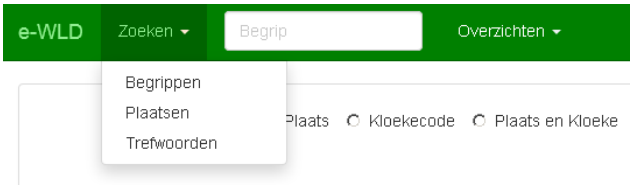


Figure 5. The menu bar's basic search functions

The web application's menu bar (Figure 5) provides access to three basic search functions: search a lemma or concept (e.g. "aard*"), search a keyword (e.g. "*appel") or look for a location (e.g. "kern*"). The result of a lemma search is a list of all the keywords, dialect forms and locations where the lemma is found; searching for a keyword lists all the lemmas, dialect forms and places where the keyword is found. A search for a location yields all the different locations that match the query, and a list of lemmas, keywords and dialect forms for each of the matching locations. Lemma's and keywords that are part of a search's result list are clickable: they allow fast access to the respective lemma and keyword entries.

Two of the search functions allow making use of additional filtering. The search for a lemma can be extended by filtering for: Kloeke code (a location code for the Netherlands and Belgium; see above), name of the location, dialect form, publication volume and, when applicable and only for WLD, the name of the mine where words have been elicited. Searching for a keyword adds a filter for the lemma and one for keyword comments. Search results can be downloaded as tab-separated text, as an Excel file or as an HTML file.

Figure 6 illustrates one type of search result. Looking for the keyword 'inktappel' yields two lemmas that contain this keyword: *dennentakje met een harsknopje* 'little pine branch that has a resin bud' and *galnoot* 'gall apple'. Each of these lemma results is accompanied by a list of dialect forms, and each dialect form contains a list of locations (and possibly mines) where this form has been

[3] See http://www.djangoproject.com.

[4] PTVS is the Python toolkit for Visual Studio. The sources of the webportal are available at https://github.com/ErwinKomen/RU-wld.

[5] The development process required a number of attempts to import all the data, which is why a special asynchronously running admin component was added, allowing the import process to be monitored. Importing the data from just one of the published volumes requires one to two hours.

attested. The locations can be accompanied by location-specific comments. The list of results is finished by a list of volumes where these lemmas occur.[6]



Figure 6. Results obtained for the keyword 'inktappel'

An interested researcher can, on the basis of the search results, access the PDF version of the original publication and check the text over there.

4. Data model

The data model that is being used to represent the information in the dialect dictionary is shown in Figure 7.

The basic unit in the data model is the `Entry`. Each entry contains the information pertaining to a dialect word: the word itself ('woord' in the model), additional information to this particular word ('toelichting') and a number of links to other parts of the database:

- A link to the `Lemma` of which this word is part
- A link to one of the `Dialect`'s where words have been elicited
- A link to the keyword (`Trefwoord`) it belongs to
- An optional link to a number of mines (`Mijn`) this word has been found (only for WLD)
- A link to the published edition (`Aflevering`) where this particular combination of dialect-word/keyword/lemma can be found.
- Each edition links to one of the three parts (`Deel`) to which the editions belong.

The data model chosen has direct consequences for processing and using the database, since the distance between which any two elements in a relational database can be found determines the complexity of a search and, consequently, the speed of searching. All `Entry` elements of one `Lemma` are found fast enough, but retrieving and sorting the related keywords (`Trefwoord`) is more complex. The use of Django, however, takes care of optimalizations behind the scene, arriving at a workable database.

5. Conclusions and Future work

The e-WLD website was launched on December 17, 2017 (Van der Sijs 2017) and was positively reviewed in local and national Dutch news papers. The same holds for the e-WBD website, which was launched on 14 December 2017.

For the first time we can count how much material the dictionaries actually contain: in the e-WBD there appear to be 15,794 concepts, 140,091 keywords and 1,704,116 dialect forms, collected in more than 1000 dialects (each place/location representing its own dialect). In the e-WLD there appear to be 17,539 concepts, 137,231 keywords and 1,759,090 dialect forms, collected in more than 1000 dialects (each place/location representing its own dialect).

Answers on many questions can now for the first time be given, for instance whether a specific word for a certain concept is really unique or not and where particular word forms can be found in the area. Another relevant research question is the relationship between word form distributions and semantic concepts. Word lists per location can be made now quite easily, which is highly supportive for writers of local dialect dictionaries. People can easily check whether a specific word form with a specific meaning was previously documented for a Limburgian or Brabantic dialect, or perhaps in another Limburgian or Brabantic dialect. Both webportals seem to meet both professional and popular needs of people interested in dialects.

Several collaborations have been set up in order to make sure that the same uniform data base model is used among various projects dealing with these dictionaries, including the Dictionary of the Flemish dialects (*Woordenboek van de Vlaamse Dialecten*, WVD).

The University of Ghent now hosts the overarching project 'Dictionary of the Southern Dutch dialects. An integrated lexicological infrastructure for the Southern Dutch dialects' (DSDD). Aim of this project is to integrate and standardize all three southern comprehensive dialect lexicographic databases (Limburgian, Brabantic, Flemish). The consortium involved includes linguists, ICT support staff, digital humanities experts and geographers. This project will be carried out in close co-operation with the INT, the Institute for the Dutch Language in Leyden.

Finally, the overarching data model is designed in such a way that not only other regional onomasiological dictionaries can be added, but also local semasiological dictionaries, i.e. dictionaries that contain a description of the dialect vocabulary of a specific place or small region. For the Netherlands, these dictionaries are collected, digitized and curated at the Meertens Institute, and made available through http://www.meertens.knaw.nl/ewnd/. The Flanders counterpart can be found at https://www.woordenbank.be.

[6] The abbreviated publication denotation consists of the part number (running from I to III) followed by the volume number. Part III deviates from this scheme, having each publication denotation consisting of: part number (that is: III), division number (ranging from 1-4) and then the volume number.

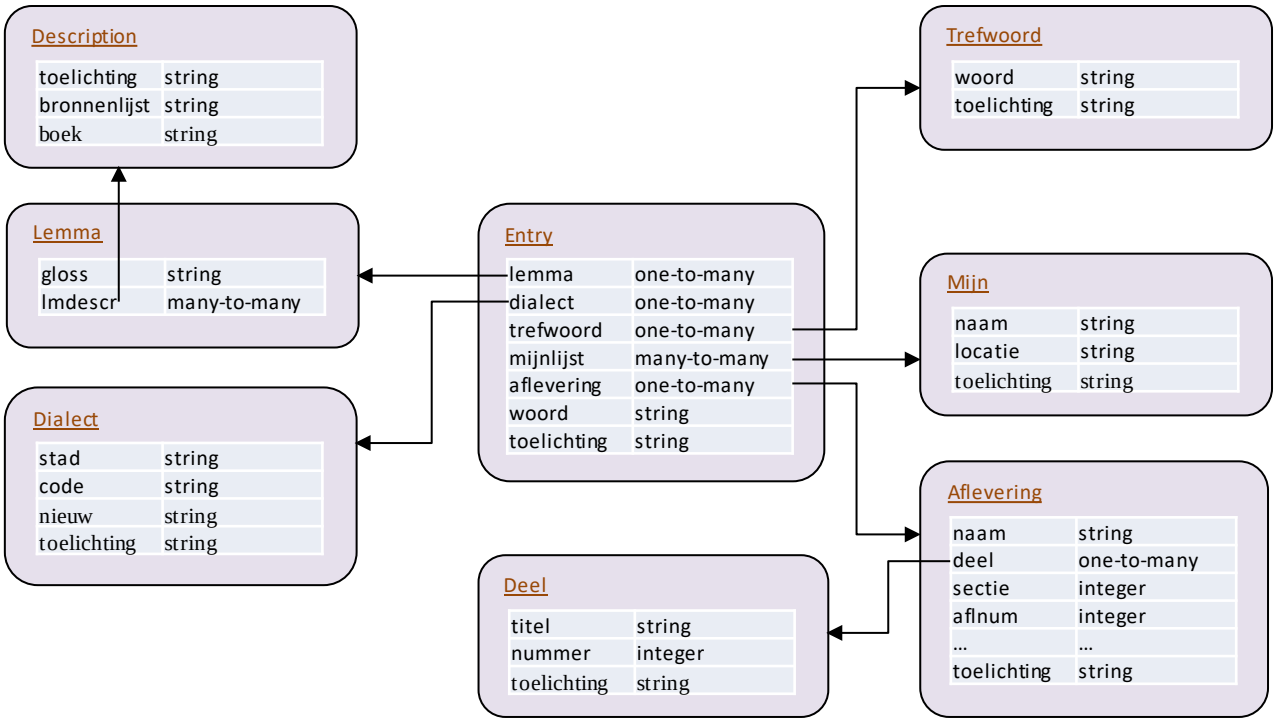


Figure 7. The data model used for the webportals of the dialect dictionaries

6. Acknowledgements

The data form the core of the web application. The data curation has been executed by former editor Joep Kruijsen, research assistants Linda van Meel and Aukje Borkent, student assistants Jorik van Engeland, Inge Otto, Lisette van der Heijde and Hanna van den Heuvel, interns Maaike Borst and Eline Dimmendaal. We greatly thank them for their meticulous and diligent work on the manual text processing. We also thank volunteers Jantien Kettenes-Van den Bosch and Herman Wiltink for their work on the preprocessing of the text documents. We thank the *Raod voor 't Limburgs* and the chair of *Diversity in Language and Culture* of Tilburg University for their moral and financial support, and Thijs Hermsen for drawing the website's logos showing the Limburgian and Brabantic dialect areas.

7. Bibliographical References

Heuvel, H. van den, Oostdijk, N., Sanders, E., and Lint, V. de (2015). Data curations by the Dutch data curation service. Overview and future perspective. In: Odijk, J. (2015). *Selected Papers from the CLARIN 2014 Conference*, October 24-25, 2014, Soesterberg, The Netherlands., pages 54–62. Linkoping Electronic Conference Proceedings, 116. http://www.ep.liu.se/ecp_article/index.en.aspx?issue=116;article=005.

Heuvel, H. van den, Sanders, E. and Sijs. N. van der (2016). Curation of Dutch Regional Dictionaries. *Proceedings LREC 2016* (10th edition of the Language Resources and Evaluation Conference, 23-28 May 2016, Portorož (Slovenia).

Kruijsen, J. and Sijs, N. van der (2010). Mapping Dutch and Flemish. In: Lameli, Alfred, Kehrein, Roland and Rabanus, Stefan (eds). *Language and Space: An International Handbook of Linguistic Variation: Language Mapping, Handbooks of linguistics and communication science*; 30.2, pages 180–202. De Gruyter Mouton.

Sijs, Nicoline van der (2017). Limburg heeft het! Het Woordenboek van de Limburgse Dialecten digitaal doorzoekbaar. In: *Neerlandia* 1, pages 41-43.

Sijs, Nicoline van der (2018). Onverwoestbaar Brabants. Over het elektronische Woordenboek van de Brabantse Dialecten. In: *Neerlandia* 1.

Vriend, F. de, Boves, L., Heuvel, H. van den, Hout, R. van, Kruijsen, J., and Swanenberg, J. (2006). A unified structure for Dutch dialect dictionary data. In : *Proceedings LREC 2006*, Language Resources and Evaluation Conference, Genova, Italy 2006, pages 1660–1665.

WBD = *Woordenboek van de Brabantse Dialecten.* Assen/Maastricht/Groningen/Utrecht, 1967-2005.

WLD = *Woordenboek van de Limburgse Dialecten* (WLD). Assen/Maastricht/Groningen, 1982-2008

WVD = *Woordenboek van de Vlaamse Dialecten*, Tongeren, 1979-.

Wijngaard, T. van de, and Keulen, R. (2007). De indeling van de Limburgse dialecten. In: Keulen, R. Wijngaard, T. van de, Crompvoets, H. and Walraven, F. (ed.). Riek van klank. *Inleiding in de Limburgse dialecten*, Sittard, pages 15-22.

Arabic Dialect Identification in the Context of Bivalency and Code-Switching

Mahmoud El-Haj[1], **Paul Rayson**[1] **and Mariam Aboelezz**[2]
[1]School of Computing and Communications, Lancaster University, UK and [2]British Library, UK
{m.el-haj, p.rayson}@lancaster.ac.uk, mariam.aboelezz@bl.uk

Abstract

In this paper we use a novel approach towards Arabic dialect identification using language bivalency and written code-switching. Bivalency between languages or dialects is where a word or element is treated by language users as having a fundamentally similar semantic content in more than one language or dialect. Arabic dialect identification in writing is a difficult task even for humans due to the fact that words are used interchangeably between dialects. The task of automatically identifying dialect is harder and classifiers trained using only n-grams will perform poorly when tested on unseen data. Such approaches require significant amounts of annotated training data which is costly and time consuming to produce. Currently available Arabic dialect datasets do not exceed a few hundred thousand sentences, thus we need to extract features other than word and character n-grams. In our work we present experimental results from automatically identifying dialects from the four main Arabic dialect regions (Egypt, North Africa, Gulf and Levant) in addition to Standard Arabic. We extend previous work by incorporating additional grammatical and stylistic features and define a *subtractive bivalency profiling* approach to address issues of bivalent words across the examined Arabic dialects. The results show that our new methods classification accuracy can reach more than 76% and score well (66%) when tested on completely unseen data.

Keywords: Arabic, bivalency, language identification, dialects, machine learning, NLP

1. Introduction

In natural language processing, the problem of detecting the language of a given text is called language identification or language guessing. In early work, relatively simple approaches employing character n-grams proved to be successful (Cavnar and Trenkle, 1994; Dunning, 1994; Souter et al., 1994). More recently, this has been seen as a classification problem where machine learning is used to distinguish between languages (Gupta et al., 2015). The vast majority of language identification research has focused on differentiating between languages. In this paper, we instead focus on differentiating regional varieties of the same language (i.e. dialects), taking Arabic as our case study.

Automatically identifying dialects could prove fruitful in fields such as natural language processing, corpus linguistics and machine translation (Zaidan and Callison-Burch, 2014). The process of hiring human participants to identify dialects is very costly and it is a time consuming job. Therefore, machine automation could work as a quick and cheap alternative provided we can create an effective mix of new methods with the appropriate dataset.

In this study, we tackle the problem of automatically identifying Arabic dialects using a variety of approaches in order to address bivalency and dialectal written code-switching (Habash et al., 2008; Biadsy et al., 2009) which pose significant challenges for existing approaches. We apply our novel *Subtractive Bivalency Profiling (SBP)* approach to address the issue of bivalent words across the Arabic dialects examined here. The results show that our new methods can achieve good levels of accuracy on unseen data.

Bivalency is defined by Woolard and Genovese (2007) as the "simultaneous membership of a given linguistic segment in more than one linguistic system in a contact setting". It is typically a feature of linguistic codes that are closely related to each other, like Standard Arabic and the various Arabic colloquial varieties. Woolard uses the term 'strategic bivalency' to refer to deliberate linguistic manipulation that makes it nearly impossible to classify a segment of speech as belonging to one code or the other. She originally introduced the term bivalency to talk about spoken language use (namely Spanish and Catalan), but it has since been extended to writing. For example, Mejdell (2011) later studied strategic bivalency in written Arabic with respect to Standard Arabic and Egyptian Arabic.

Bivalency is a hallmark feature of written Arabic, especially in the case of single-word or short utterances such as "القلم علَى المكتب" (the pen is on the table). The three words in this utterance can be found - with the same semantic content - in all major Arabic dialects. What makes bivalency more common in writing than speech is that regional variants such as 'qalam', 'galam' and 'alam' - which are easily distinguished in speech - are all likely to be represented using the Arabic writing system as "قلم". Hence, written bivalency (bivalency hereafter) is not simply the result of overlap in vocabulary but also the loss of important linguistic information when different pronunciations are encoded using the same standard representation in Arabic script.

2. Related Work

For differentiating texts at the language level, comparing the relative ranks of character n-grams has proved to be a very successful approach (Cavnar and Trenkle, 1994). Recent research has tackled language identification in noisy settings such as online forums and social media using ensemble methods (Lui and Baldwin, 2014) or more complex statistical approaches (Abainia et al., 2016). Much other research focuses on language identification or recognition from speech signals but that is out of scope for this paper. In the area of corpus linguistics, language identification is not studied directly, but the field has a long history of comparing language varieties and has developed a number of approaches to explore this issue e.g. keywords used in an American versus British English study (Hofland and Johansson, 1982) and multidimensional approaches (Biber, 1988).

Previous work on the more fine-grained task of dialect identification itself is much more scarce with some reported research on African-American English (Blodgett et al., 2016) and European versus Brazilian Portuguese (Laboreiro et al., 2013).

Zaidan (2014) created an Arabic resource of dialect annotation using Mechanical Turk crowdsourcing. The annotators labeled 100,000 sentences defining the Arabic dialect used in writing. They trained a classifier to identify dialectal Arabic in text harvested from online social media. The dialects used to train their classifier were Egyptian, Gulf, Levant, Iraqi and Maghrebi (also known as North African Arabic). The probabilistic classification models used words and character n-gram features. Considering the overlap between dialects the training accuracy did not perform better than 88%. In our work, we hypothesise that using additional grammatical and stylistic features would outperform relying on n-gram features alone to classify Arabic dialects.

Elfardy (2014) trained a supervised classifier to distinguish between Modern Standard Arabic and Egyptian dialect extracting n-gram and token based features to class each word in a sentence whether it is MSA, Egyptian or out-of-vocabulary (OOV). The best configuration classifier achieved an accuracy of just above 80%. For a binary classification task we expect the classifier to perform better. For example, we ran a simple n-gram binary classifier using just MSA and Egyptian, and the classifier achieved near 99% accuracy on training and above 95% when tested on unseen data.

3. Dataset

The dataset used in our research covers four major Arabic dialect groups: Egyptian (EGY), Levant (LAV), Gulf (GLF), and North African (NOR). The dataset also includes Modern Standard Arabic (MSA)[1]. Apart from NOR, all the other dialects were collected from the Arabic Commentary Dataset (AOC) (Zaidan and Callison-Burch, 2014). We randomly selected commentaries written in MSA, EGY, GLF and LAV. Iraqi and Maghrebi (NOR) sections from that corpus do not provide enough sentences for our experiments.

In their preparation of the AOC dataset, it was annotated through hiring online participants on Mechanical Turk[2]. The participants annotated the dataset through answering two questions a) how much dialect is in the sentence, and b) which Arabic dialect the writer intends. It is worth noting that we only selected sentences where the answers to the first question is 'mostly dialect' and where either EGY, GLF, MSA or LAV is the answer to the second question. We did not include NOR (called Maghrebi in AOC) as there were not enough NOR sentences where the answer to the first question is 'mostly dialect'. Instead we supplemented the collection with NOR dialect from Tunisian Arabic which is a free online corpus of Tunisian (North African) Arabic[3]. We randomly selected sentences from the *Internet Forums* category so that it is consistent with the AOC dataset.

During our initial examination of the corpus data, we noticed bivalency between the dialects in the dataset. We also noticed that many writers used a combination of dialect and MSA. This is common in political debates where readers comment on a political news article and others respond to them.

The dataset is rich with bivalent words such as "ترجمة" [translation] and "النّاس" [people]. There is also frequent code-switching to MSA, with phrases such as "حتّى هذه اللّحظة" [Until this moment] and "لم يسبق ان" [never before] which are rarely used in dialect conversations. We later describe how this played a vital role in our dialect identification process. Table 1 shows the count of sentences (instances/samples) and words for each class.

Dialect Label	Sentences	Words
GLF	2,546	65,752
LAV	2,463	67,976
MSA	3,731	49,985
NOR	3,693	53,204
EGY	4,061	118,152
Total	16,494	355,069

Table 1: Training data size

4. Automatic Dialect Identification (ADID)

For the purpose of this task we trained different text classifiers using four algorithms: Naïve Bayes, Support Vector Machine (SVM), k–Nearest Neighbor (KNN) and Decision Trees (J48).

4.1. Baselines

For the first baseline, we created a simple classifier that always selects the most frequent class (EGY in this case). As a more intelligent baseline, we extracted simple word-level n-gram features selecting unigram, bigram and trigram contiguous words using a Naïve Bayes classifier. The second baseline classifiers' accuracy was expected to be an improvement over the most frequent class approach.

4.2. Feature Extraction

To help the classifier distinguish between the dialects more accurately, we extracted more linguistically informed features in addition to our subtractive bivalency profiling method. The selected features fall into two groups: grammatical and stylistic.

4.2.1. Grammatical Features

In order to extract grammatical knowledge from the training data we used the Stanford Part of Speech (POS) Tagger[4] to annotate the text with part-of-speech tags. The POS tagger was trained on an MSA dataset but we judged it to be appropriate enough for our experiments. Key differences

[1]The Arabic Dialects Dataset is freely available for research purposes and can be directly downloaded from http://www.lancaster.ac.uk/staff/elhaj/corpora.htm

[2]http://www.mturk.com

[3]http://www.tunisiya.org/

[4]http://nlp.stanford.edu/software/tagger.shtml

will be in the sentence structure and the introduction of dialect words that may not have appeared in the MSA training data. We expect this may make the tagger more error-prone, but we wish to understand whether it can still help in distinguishing between dialects.

Using the annotated training data introduced in section 3., we extracted a number of grammatical features. *Tag frequency* refers to the frequency of each tag found in the POS tagset while *uniqueness* refers to the number of tag types introduced in the text. In addition to the tag frequencies, we also extracted features which are counts of function words of the following types: adverbs, adverbials, conjunctions, demonstratives, modals, negations, particles, prepositional, prepositions, pronouns, quantifiers, interrogatives and comparatives. Each list contains function words/tags related to that category (Garcia-Barrero et al., 2013; Ryding, 2014).

4.2.2. Stylistic Features

In addition to grammatical features we also extracted two stylistic features, namely a readability metric and Type-Token-Ratio (TTR) which are used elsewhere in authorship identification (Holmes, 1994). TTR is the ratio obtained by dividing the total number of different words (i.e. types) occurring in a text by the total number of words (tokens). Higher TTR indicates a high degree of lexical variation. We calculated TTR by simply dividing the number of types by the number of tokens in each instance (Holmes, 1994):

$$TTR = \frac{types}{tokens}$$

We normalised the output by dividing by the number of sentences in each instance, this was achieved by using the Stanford Arabic sentence splitter[5].

Furthermore, we measured the readability of the text using the OSMAN readability metric (El-Haj and Rayson, 2016). In addition to providing a readability score between 0 (hard to read) and 100 (easy to read), OSMAN also provides information about the number of syllables, hard words (words with more than 5 letters), complex words (>4 syllables) and Faseeh (aspects of script usually dropped in informal Arabic writing).

4.2.3. Subtractive Bivalency Profiling

As mentioned earlier the dataset contains a high level of language bivalency, which is typical when speakers switch between closely related language varieties. We used an approach influenced by earlier work in corpus linguistics in order to select features to study the closeness and homogeneity between the texts in the different classes. We have therefore devised a new method which we have dubbed *Subtractive Bivalency Profiling* (SBP). When examining the frequency lists for each dialect, we noticed that writers occasionally switch to MSA in an apparent bid to invoke formality and/or authority. In order to use bivalency and written code-switching as features in the classification process, we created dialect-specific frequency lists to distinguish the vocabularies spoken in each dialect compared to MSA. The frequency lists we created are of two types: a) dialect SBP list, and b) MSA written code-switching list. In the former list we worked on identifying and removing bivalent words between dialects aside from MSA leaving us with more fine-grained dialectal lists. We then found bivalent words between dialects and MSA, which we refer to as MSA written code-switching. Dialect SBP lists were created using an independent dialectal dataset that has not been exposed to the training process. The independent dataset called DART[6] contains more than 24,000 Arabic sentences labeled into 5 Arabic dialects (Egyptian, Gulf, Levantine, Iraqi and Maghrebi (North African)) matching the dialects we are working with for this paper (we have taken Iraqi out). We created a dialect SBP list for each dialect (in addition to MSA). This was done by creating a unique (with no duplicates) frequency list for each dialect in DARTS removing from that list any bivalent words between that dialect and any of the remaining dialects.

The MSA written code-switching list was created using an independent list of MSA sentences from the United Nations (UN) Corpus[7]. We created a frequency list for the UN corpus and detected bivalency with each of the other four dialects. Here we keep the bivalent words between MSA and each of the other dialects creating a MSA written code-switching list for each dialect.

5. Feature Selection

The count of the selected features including each entry of the frequency and grammatical lists was 50 divided into 3 categories as in Subsection 4.2.. Training the algorithms using this number of features took a significant amount of time thereby making it difficult to attempt many different algorithms. In order to simplify the model, shorten the training time and enhance generalisation to reduce overfitting, we reduced the number of features using machine learning feature selection technique as explained below.

5.1. Classifier Subset Evaluator

To reduce the number of features we used WEKA[8] Classifier Subset Evaluator which evaluates attribute subsets on training data and uses a SVM classifier to estimate the merit of a set of attributes. This helps evaluate the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. This was combined with a Best First Search (BFS) which searches the space of attribute subsets by greedy hill climbing augmented with a backtracking facility. The attribute selection process selected 8 features as the most predictive ones. The selected features are: SBP (MSA_SBP, EGY_SBP, GLF_SBP, LAV_SBP, NOR_SBP), Grammatical (Conjunctions) and Stylistic (Osman-readability and type-Token-Ratio).

Using the complete set of features, J48 algorithm achieved an accuracy of 75.66%. After reducing the set of features we found that the top 8 features can achieve an accuracy of 75.02% using J48. This means that 16% of the features are enough to achieve a similar accuracy to that of using

[5] http://nlp.stanford.edu/projects/arabic.shtml

[6] Dialectal Arabic Tweets (DARTS) http://qufaculty.qu.edu.qa/telsayed/datasets/

[7] http://www.uncorpora.org/

[8] https://www.cs.waikato.ac.nz/ml/weka/

all the features combined. This 84% reduction helped in simplifying the model and shortening the training time.

5.2. Feature-Group Filtering

In order to examine the effect of each feature group (section 4.2.), we ran the classifiers using all of the training data testing on each feature-group individually and combined. This process helped us determine which set of features (feature-group) contribute most to identifying dialects. It also helped demonstrate which feature-groups are not meant to be used together, as such combination may increase complexity.

6. Results and Discussion

Overall, the best machine learning algorithm correctly distinguished dialects and MSA with more than 76% accuracy. This was calculated by training a SVM classifier using all set of features. We later show the detailed classification results after reducing the number of features as described in Section 5.. We compare that to our two baselines below.

6.1. Baselines

We used the most frequent class as our first baseline model. We also used a unigram, bigram and trigram word Naïve Bayes n-gram classifier as our second more intelligent baseline. Table 2 shows the accuracy scores of each baseline. The most frequent class (EGY) achieved an accuracy of 24% for the first baseline and reached 52% when using the second, more sensible baseline. The second score is still quite low due to the high bivalency between the dialects, especially in relation to short sentences, which in some cases were only one word long. On average, each instance contains 40 words with more than 3,000 instances containing less than 20 words. It is very difficult even for humans to guess the dialect for instances with one bivalent word such as "نعم" [yes], "رياضة" [sport] and "تعليم" [education], a task that is deemed impossible for a machine since these words are bivalent. Therefore, we expect our more refined, linguistically informed features will perform better to help the machine distinguish between the dialects.

Baseline	Accuracy
Most frequent class	24.62%
Word n-gram	52.07%

Table 2: Baseline Results

6.2. Training Results

We used a 10-fold cross validation to evaluate the models and avoid over-fitting. As shown in Section 4. we used four classifiers: J48, SVM, Naïve Bayes and KNN.

Table 3 shows the accuracy of the models on our training dataset. SVM achieved the highest accuracy with higher recall and precision compared to the other algorithms.

With more than 76% accuracy, our model performed better than a previous approach (Ali et al., 2015) which used a set of lexical and acoustic features to train a classifier using a dialectal dataset generated using an Arabic Automatic Speech Recognition (ASR) system.

M	A%	R	P	F
SVM	**76.29**	**0.76**	**0.79**	**0.78**
J48	75.66	0.76	0.75	0.76
KNN	62.72	0.63	0.62	0.63
NB	56.96	0.57	0.75	0.65

Table 3: Training results using all features
M: Model, A%: Accuracy, R: Recall, P: Precision, F: F-Measure

Table 4 shows the results of using our group-feature in addition to the combination of the 8 features (AttSel) selected using a Classifier Subset Evaluator as explained in Section 5.. As shown in the table, our SBP features play a vital role in helping the classifier identify dialects.

Feature	J48	SVM	NB	KNN
SBP + Gram	75.95	75.32	56.61	64.09
AttSel	75.02	69.96	65.45	69.26
SBP	74.97	71.01	59.19	72.49
Sty + SBP	74.55	69.55	59.66	69.03
Sty + Gram	51.50	54.21	41.47	47.02
Gram	50.56	52.56	40.47	46.39
Sty	45.59	31.22	33.82	42.41

Table 4: Examining Feature Groups (training)
Sty: Stylistics, SBP: Subtractive Bivalency Profiling, Gram: Grammatical, AttSel: Attribute Selection

The results show that our SBP method alone outperformed all the other features and that combining SBP with other features such as Grammatical and Stylistic features provides a small boost to accuracy. The selected features helped the classifier to distinguish between the dialects with a minimised error rate.

However, the results show that the Grammatical and Stylistic features alone did not perform better than the intelligent baseline. Comparing tables 3 and 4 shows that Naïve Bayes and KNN as well slightly improve the results when combining SBP with Stylistic features.

Table 5 shows the confusion matrix for the SVM classifier which achieved the highest scores as shown in Table 3. The confusion matrix shows that some dialects are harder to classify than others due to overlap with other dialects which reinforces our earlier informal observation that dialects use words interchangeably. The table shows the classifier to miss-classify EGY near equally between the other dialects when considering the number of pairwise mis-classified items. The table shows less confusion between GLF, LAV and NOR dialects. We believe this is due to the use of the SBP method which made it easier for the classifier to distinguish between dialects. It is not quite clear which dialect is closest to MSA but from the table we can observe that LAV is most likely to be miss-classified as MSA more than the other dialects in this corpus. Finally, it is important to note that our classifier has managed to equally distinguish between dialects whereas we can see in the confusion matrix that none of the dialects has been individually highly mis-classified except for MSA and EGY. This shows that the selected features are of good quality but that there is still scope for further improvement.

	EGY	GLF	LAV	MSA	NOR	Total
EGY	**3,371**	62	37	518	73	4,061
GLF	403	**1,573**	17	535	18	2,546
LAV	509	67	**1,231**	601	55	2,463
MSA	393	86	67	**3,047**	138	3,731
NOR	101	17	7	206	**3,362**	3,693
					Total	16,494

Table 5: SVM Classifier Confusion Matrix (based on Table 3)

We believe that our model can perform better with more refinement to the training data which could help decrease the impurity of the instances. To show how further refinement could help increase the accuracy of a classifier we trained a model using the same set of reduced features but this time only considering instances with more than or equal to 20 words. We reached this threshold by training a machine to increase the threshold with an increment of 1 for each classification iteration and to stop when the accuracy stops increasing or stalls. This has helped increase the accuracy of the SVM classifier to around 78%. This clearly shows how short sentences affect the classifier's accuracy. We will not use this classifier for testing as we may end up penalising the classifier when tested over short sentences. By refining the training data to only select instances with more than 20 words we intend to show how words overlap between dialects. Having these words out of context makes it difficult for a prediction model to infer the correct dialect even with more instances than what we have in our training data.

6.3. Unseen Testing Results

To further test the classifiers we used a separate unseen dataset. The source of the unseen testing data is similar to those of the training instances shown in Section 3. The randomly selected unseen data has never been used in training the classifiers and we use it to demonstrate how the classifier performs when tested on new data. Table 6 shows the distribution of the testing data.

Label	Sentences	Words
EGY	1,741	40,768
GLF	1,092	17,070
LAV	1,056	18,215
MSA	1,600	29,759
NOR	1,584	33,066
Total	7,073	138,878

Table 6: Testing data count

Table 7 shows the testing results using each classifier and set of features. In line with the training results (Section 6.) the testing results show that the SBP features alone outperformed all the other features. Moreover, combining SBP with other features such as Grammatical and Stylistic features slightly boosts classification accuracy. The testing results outperformed the two baselines in Section 4.1. which is also shown in the table when using n-gram features (c42%). The results show using n-gram features did not perform well on new unseen dataset. This could be due to the presence of new vocabulary items that the classifier has not encountered before. This suggests that our SBP feature group in addition to Grammatical and Stylistic features can still identify dialects fairly well even in the presence of vocabulary that the classifier has not seen before.

Feature	J48	SVM	NB	KNN
All	66.12	64.43	47.39	53.36
SBP + Gram	63.66	63.58	46.58	52.38
AttSel	60.61	63.96	56.47	56.78
SBP	57.06	61.73	47.65	57.83
Sty + SBP	57.04	60.17	49.39	54.47
Sty + Gram	55.04	50.52	40.63	44.19
Gram	52.49	50.38	38.92	42.17
Sty	44.60	45.08	37.30	40.19
n-gram	42.78	31.02	32.36	38.86

Table 7: Testing results
Sty: Stylistics, SBP: Subtractive Bivalency Profiling, Gram: Grammatical, AttSel: Attribute Selection, All: All features as in Table 3.

7. Conclusion

In this paper, we used machine learning to automatically detect dialects in a dataset comprising four Arabic dialects groups (Egyptian, Gulf, Levant and North African) in addition to Standard Arabic by applying a new method termed Subtractive Bivalency Profiling (SBP). The results showed that our SBP method alone outperformed all the other individual features and that the results improve slightly when combining SBP with other features. Code and other resources used in this paper are released freely on our GitHub repository.[9]

8. Acknowledgment

We would like to thank the University Centre for Computer Corpus Research on Language (UCREL) at Lancaster University for funding this work.

9. Bibliographical References

Abainia, K., Ouamour, S., and Sayoud, H. (2016). Effective language identification of forum texts based on statistical approaches. *Information Processing & Management*, 52(4):491 – 512.

Ali, A. M., Bell, P., and Renals, S. (2015). Automatic dialect detection in arabic broadcast speech. *Computing Research Repository*, abs/1509.06928.

Biadsy, F., Hirschberg, J., and Habash, N. (2009). Spoken arabic dialect identification using phonotactic modeling. In *Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages*, Semitic '09, pages 53–61, Stroudsburg, PA, USA. Association for Computational Linguistics.

Biber, D. (1988). *Variation across speech and writing*. Cambridge University Press, Cambridge.

Blodgett, L. S., Green, L., and O'Connor, B. (2016). Demographic dialectal variation in social media: A case

[9]https://github.com/drelhaj/ArabicDialects

study of african-american english. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130. Association for Computational Linguistics.

Cavnar, W. B. and Trenkle, J. M. (1994). N-gram-based text categorization. In *In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175.

Dunning, T. (1994). Statistical identification of language. Technical report, New Mexico State University.

El-Haj, M. and Rayson, P. (2016). Osman – a novel arabic readability metric. In *10th Language Resources and Evaluation Conference (LREC 2016)*, pages 250–255, Portoroz, Slovenia.

Elfardy, H., Al-Badrashiny, M., and Diab, M. (2014). Aida: Identifying code switching in informal arabic text. In *First Workshop on Computational Approaches to Code Switching. EMNLP 2014*, pages 94–101, Doha, Qatar.

Garcia-Barrero, D., Feria, M., and Turell, M. T. (2013). Using function words and punctuation marks in arabic forensic authorship attribution. In *Proceedings of the 3rd European Conference of the International Association of Forensic Linguists*, pages 42–56, Porto, Portugal.

Gupta, Kumar, D., Kumar, S., and Ekbal, A. (2015). Machine learning approach for language identification & transliteration. In *Proceedings of the Forum for Information Retrieval Evaluation*, FIRE '14, pages 60–64, New York, NY, USA. ACM.

Habash, N., Rambow, O., Diab, M., and Kanjawi-Faraj, R. (2008). Guidelines for annotation of arabic dialectness. In *LREC Workshop on HLT and NLP within the Arabic World*, pages 49–53, Marrakesh, Morocco.

Hofland, K. and Johansson, S. (1982). *Word frequencies in British and American English*. The Norwegian Computing Centre for the Humanities, Bergen, Norway.

Holmes, D. I. (1994). Authorship attribution. *Computers and the Humanities*, 28(2):87–106.

Laboreiro, G., Bošnjak, M., Sarmento, L., Rodrigues, E. M., and Oliveira, E. (2013). Determining language variant in microblog messages. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, pages 902–907, New York, NY, USA. ACM.

Lui, M. and Baldwin, T. (2014). Proceedings of the 5th workshop on language analysis for social media (lasm). pages 17–25. Association for Computational Linguistics.

Mejdell, G. (2011). Strategic bivalency in written 'mixed style'? a reading of ibrahim isa in al-dustur. In *Proceedings of the 9th International Arabic Dialectology Association (Aida) Conference*, Aida 9th, pages 273–278.

Ryding, K. (2014). *Arabic: A Linguistic Introduction*. Cambridge University Press.

Souter, C., Churcher, G., Hayes, J., Hughes, J., and Johnson, S. (1994). N-gram-based text categorization. In *Natural Language Identification using Corpus-based Models*, pages 161–175. Hermes Journal of Linguistics.

Woolard, K. A. and Genovese, E. N. (2007). Strategic bivalency in latin and spanish in early modern spain. *Language in Society*, 36(4):487–509.

Zaidan, O. F. and Callison-Burch, C. (2014). Arabic dialect identification. *Comput. Linguist.*, 40(1):171–202, March.

Unified Guidelines and Resources for Arabic Dialect Orthography

Nizar Habash, Fadhl Eryani, Salam Khalifa, Owen Rambow,† Dana Abdulrahim,‡
Alexander Erdmann, Reem Faraj,† Wajdi Zaghouani,* Houda Bouamor,♣
Nasser Zalmout, Sara Hassan, Faisal Al-Shargi,⋆ Sakhar Alkhereyf,†
Basma Abdulkareem,† Ramy Eskander,† Mohammad Salameh,♣ Hind Saddiki
New York University Abu Dhabi, UAE, † Columbia University, USA, ‡University of Bahrain, Bahrain,
*Hamad Bin Khalifa University, Qatar, ♣Carnegie Mellon University in Qatar, Qatar, ⋆Univisität Leipzig, Germany
nizar.habash@nyu.edu, rambow@cs.columbia.edu, darahim@uob.edu.bh,
wzaghouani@hbku.edu.qa, hbouamor@cmu.edu, alshargi@informatik.uni-leipzig.de

Abstract

We present a unified set of guidelines and resources for conventional orthography of dialectal Arabic. While Standard Arabic has well defined orthographic standards, none of the Arabic dialects do today. Previous efforts on conventionalizing the dialectal orthography have focused on specific dialects and made often ad hoc decisions. In this work, we present a common set of guidelines and meta-guidelines and apply them to 28 Arab city dialects from Rabat to Muscat. These guidelines and their connected resources are being used by three large Arabic dialect processing projects in three universities.

Keywords: Dialectal Arabic, Orthography, Phonology, Morphology, Conventions

1. Introduction

Arabic dialects are linguistic varieties that are historically related to classical Arabic and co-exist with Modern Standard Arabic (MSA) in a diglossic relationship. While MSA, the official language of all Arab countries, has well-defined orthographic standards, Arabic dialects have no official orthographies. As such, besides unintentional typographic errors, no spelling of a dialectal word can be considered "incorrect." And since Arabic dialects vary from MSA and from each other in terms of phonology, morphology, lexicon and syntax (Watson, 2007), using MSA orthographic standards cannot fully address the needs of the dialects. As an example of the degree of variety in dialectal spelling, Figure 1. presents the 27 actually attested spellings of one Egyptian Arabic word online. The large number of possibilities results from independent decisions such as whether the proclitic /ma/ should be written attached or separated (+م *m*+[1] or ما *mA*), or whether to write the stem in a way that reflects its phonology (ؤ ŵ), or etymology (ق *q*).

Arabic Orthography	Arabic Transliteration	Frequency
مبيقولهاش	*mbyqwlhAš*	≈ 26,000
ما بيقولهاش	*mA byqwlhAš*	≈ 13,000
مابقلهاش، مبقولهاش، مبقلهاش، ما بقلهاش، مابيقولهاش	*mAbqlhAš, mbqwlhAš, mbqlhAš, mA bqlhAš, mAbyqwlhAš*	≤ 10,000
مابقولهاش، ما بقولهاش، مبيقلهاش، ما بيقلهاش	*mAbqwlhAš, mA bqwlhAš, mbyqlhAš, mA byqlhAš*	≤ 1,000
مبئلهاش، مابيئولهاش، ما بيئولهاش، مابيؤلهاش	*mbŷlhAš, mAbyŷwlhAš, mA byŷwlhAš, mAbywŵlhAš*	≤ 100
ما بيؤلهاش، مابئلهاش، مبيئولهاش، ما بيئلهاش، مابئولهاش، ما بئلهاش، ما بؤلهاش، مبئولهاش، مبيؤلهاش، مابؤلهاش، مبؤلهاش	*mA bywŵlhAš, mAbŷlhAš, mbyŷwlhAš, mA byŷlhAš, mAbŷwlhAš, mA bŷlhAš, mA bwŵlhAš, mbŷwlhAš, mbywŵlhAš, mAbwŵlhAš, mbwŵlhAš*	≤ 10

Figure 1: 27 encountered ways to write the Egyptian Arabic word /mabiʔulha:ʃ/ 'he does not say it' and their frequencies from Google Search (September 29, 2017).

Habash et al. (2012) introduced the concept of Conventional Orthography for Dialectal Arabic (CODA); and they proposed a set of guidelines and exception lists for Egyptian Arabic. Their conventions were used in the Linguistic Data Consortium for annotating Egyptian Arabic (Maamouri et al., 2014). Since then, a number of additional efforts followed suit for other dialects (Zribi et al., 2014; Saadane and Habash, 2015; Jarrar et al., 2016; Khalifa et al., 2016). While the original CODA guidelines aimed at being easy to adjust to new dialects and contained some dialect-independent components, the guidelines were not specific enough, and often open to interpretation. Furthermore, the resources supporting the process of extending CODA to new dialects were non-existent.

Previous CODA efforts approached the conventionalization problem with a focus processing Arabic dialect text as input only. They did not address the challenge of generating Arabic dialect text for human readability (e.g., as output of speech recognition, machine translation or chatbots). Considering both aspects (input and output) highlights the need of conventions that are accessible to Arabic readers.[2]

In this work, we present a common set of guidelines with enough specificity to help in creating dialect-specific conventions, and we apply them to 28 Arab city dialects.

[1] Arabic script transliteration is presented in the Habash-Soudi-Buckwalter transliteration scheme (Habash et al., 2007):
أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي
Â b t θ j H x dð r z s š S D T Ď ς γ f q k l m n h w y
and the additional symbols: ' ء, Â أ, Ǎ إ, Ā آ, ŵ ؤ, ŷ ئ, ى ý, ة ħ. Phonological forms are presented in IPA or in the CAPHI scheme, which is discussed in Section 5.

[2] Most recently, the Palestinian CODA conventions have been adopted by a website for teaching Colloquial Arabic: http://www.learnpalestinianarabic.com.

We call our new version of CODA: CODA* (pronounced *CODA Star*, as in, for *any dialect*). The contributions of this paper include: (a) the definition of a phonological representation inspired by Arpabet (Shoup, 1980) for Arabic and its dialects to be used for specifying the pronunciation in computational resources; (b) a clear separation between CODA general dialect-independent rules and specification rules for organizing and managing the numerous exceptional cases presented in previous work; (c) the introduction of the concept of a multidialectal *Seed Lexicon* that is used to allow users of CODA* to have access to previous decisions when identifying spellings for new words in new dialects; and finally, (d) a set of online pages that give users easy public access to all of these resources.

The CODA* guidelines and their connected resources are being used by three large Arabic dialect processing projects in three universities: The Multi-Arabic Dialect Applications and Resources at Carnegie Mellon University Qatar and New York University Abu Dhabi (NYUAD) (Bouamor et al., 2018), The Gulf Arabic Annotated Corpus (NYUAD) (Khalifa et al., 2018), and The Columbia Arabic Dialect Annotation project (Columbia University and NYUAD). The CODA* effort is large and ongoing; the goal of this paper is to introduce the effort and some of its important contributions on how to conceptualize and address the question of orthographic decisions in dialectal Arabic computational processing.

The rest of the paper is structured as follows. We present common challenges to Arabic processing in Section 2. This is followed by related work in Section 3. We introduce CODA* in Section 4., and discuss its components in Section 5. (CAPHI), Section 6. (General Rules and Specifications), and Section 7. (Seed Lexicon).

2. Challenges to Arabic Processing

There are four distinct and orthogonal challenges to working on written Arabic natural language processing (NLP): morphological richness, orthographic ambiguity, dialectal variations, and orthographic inconsistency.

Morphological Richness Arabic words have a large number of forms. This results from a rich inflectional morphology that models gender, number, person, aspect, mood, case, state and voice, in addition to a large number of clitics such as conjunctions, negative particles, future particles, etc. The word featured in Figure 1. is only one of a few thousand forms (inflections and cliticizations) of the verbal lemma قال *qAl* 'to say'.

Orthographic Ambiguity Arabic orthography using the Arabic script employs optional diacritical marks for short vowels and consonantal gemination. The missing diacritics are not a major challenge to literate native adults. However, their absence is the main source of ambiguity in Arabic NLP. In MSA, the average ambiguity is 2.7 lemmas per word (Habash, 2010). For example, the MSA word عقد ʕqd can be diacritized as عَقَد ʕaqad 'contract' or عُقُد ʕuqud 'necklace', among other readings.

Dialectal Variations Arabic dialects are often classified regionally (such as Egyptian, Levantine, Gulf, etc.) or subregionally (e.g., Lebanese, Syrian, Jordanian, etc.). These classifications are generally problematic because of the continuous nature of language variation. In our work, we have opted to focus on specific cities that represent the different regions and sub-region, in an effort to control the degree of variation we study. Table 1 lists the names of the cities we cover in the work presented in this paper. Arabic dialects differ significantly in terms of their phonology, morphology and lexicon from one another and from MSA (Watson, 2007).

Orthographic Inconsistency Noise in written text is a common problem for NLP when working in social media and non-edited text (see Figure 1.). For MSA, Zaghouani et al. (2014) report that 32% of words in MSA comments online have spelling errors. Eskander et al. (2013) also report close to 24% of Egyptian Arabic words having non-CODA-compliant spelling. Dialectal Arabic text is also known to appear on social media in a non-standard romanization, often called Arabizi (Darwish, 2013).

The work presented in this paper focuses primarily on the issue of orthographic inconsistency although it is inseparable from all of the other challenges.

3. Related Work

Before Habash et al. (2012) introduced their Egyptian Arabic conventional orthography (CODA-Egyptian), there were many proposals such as the Asaakir system ('Asaakir, 1950) and Akl's system (Arkadiusz, 2006), neither of which are broadly used today. Various DA dictionaries used Arabic, Latin or mixed script orthographies (Badawi and Hinds, 1986). In the context of NLP, the Linguistic Data Consortium (LDC) guidelines for transcribing Levantine Arabic (Maamouri et al., 2004) and the COLABA project at Columbia University (Diab et al., 2010) were precursors to the work of Habash et al. (2012).

After the CODA-Egyptian guidelines were created and used for the creation of Egyptian Arabic resources (Maamouri et al., 2014; Diab et al., 2014; Pasha et al., 2014; Eskander et al., 2013; Al-Badrashiny et al., 2014), two additional sets of guidelines were created for CODA-Tunisian (Zribi et al., 2014) and CODA-Palestinian (Jarrar et al., 2014). These were part of projects involving morphology annotation (Palestinian) or speech recognition (Tunisian). A variant on CODA was proposed for speech recognition by Ali et al. (2014) and was shown to reduce OOV and perplexity. Since then, four more dialects followed: CODA-Algerian (Saadane and Habash, 2015), CODA-Gulf (Khalifa et al., 2016), CODA-Moroccan and CODA-Yemeni (Al-Shargi et al., 2016). The latter efforts were heavily based on earlier versions, modifying/extending the exception lists of the Egyptian and Palestinian versions while preserving the general CODA rules. These efforts focused on one dialect at a time, and a number of them were only interested in processing Arabic input – not considering the challenges of dialectal Arabic output. Some recent efforts have highlighted the value of generating Arabic dialect text in the context of speech recognition, chatbots, and machine translation (Ali et al., 2014; Meftouh et al., 2015; Abu Ali and Habash, 2016). Erdmann et al. (2017), for instance, evaluated translation output in DA, finding that 10% of tokens not found in the reference but

Maghreb				Nile Basin		Fertile Crescent			Arabian Peninsula				
Morocco	**Algeria**	**Tunisia**	**Libya**	**Egypt**	**Sudan**	**South Levant**	**North Levant**	**Iraq**	**Hijaz**	**Najd**	**Gulf**	**Oman**	**Yemen**
Rabat	Algiers	Tunis	Tripoli	Cairo	Khartoum	Jerusalem	Beirut	Mosul	Jeddah	Riyadh	Abu Dhabi	Muscat	Sana'a
Fes		Sfax	Benghazi	Alexandria		Amman	Damascus	Baghdad			Manama		Taiz
				Aswan		Salt	Aleppo	Basra			Doha		

Table 1: The different region, sub-region, and city dialects covered in CODA*.

manually judged to be correct, were in fact orthographic variants of a corresponding reference token. Zalmout et al. (2018) trained a morphological disambiguator on a CODA-based version of the data to show the upper accuracy limit noise-wise.

4. CODA* : Conventional Orthography for Multiple Arabic Dialects

In this section, we review the design goals and principles of CODA and discuss how we extend the CODA guidelines to CODA*.

4.1. CODA Goals and Design Principles

The original CODA goals as outlined by (Habash et al., 2012) in their paper on CODA-Egyptian were that: (i) CODA is an internally consistent and coherent convention; (ii) CODA is created for computational purposes; (iii) CODA uses the Arabic script; (iv) CODA is intended as a unified framework for writing all DAs; and finally, (v) CODA aims to strike an optimal balance between maintaining a level of dialectal uniqueness and establishing conventions based on MSA-DA similarities. The authors also describe CODA design as a consistent *ad hoc* convention that balances being MSA-like with being generally phonemic, and morphologically and syntactically faithful to the dialect. Furthermore, it aims to be easily learnable and readable.

4.2. Extending CODA to other Dialects

Despite the stated goals and design principles, the final guidelines Habash et al. (2012) presented were for Egyptian Arabic only. They included general rules and an exception list, which is to be consulted before applying any rules. The contents of the exception list are a mix of frequent closed class words (e.g., pronouns, demonstratives), frequent adverbial expressions (e.g., *today*, *tomorrow*, etc.), number words and days of the week, with the occasional odd case or not-so obvious example that applies the general rules. The protocol of consulting the exception list before applying the general rules is clear; however, there were no guidelines on what to put on the exception list and how to write words in that list. For one dialect, this perhaps is not a problem, but as we expand to other dialects, we have no guidance for how to proceed. As such, different efforts since Habash et al. (2012) interpreted the general rules as dialect independent, and the exception list as dialect specific and *translated* the exception list, which only added more ad hoc decisions.

4.3. CODA*

What we propose in this paper is an extension of CODA, which we call CODA* (as in, for *any dialect*). CODA* includes minor adjustments to the CODA general dialect-independent rules, and major clarifications of the structure of the exception list. CODA* replaces said list with a detailed set of dialect-independent *specifications* to guide the creation of all closed classes and other categories of expressions. Additionally, CODA* introduces the concept of a *seed lexicon* which contains dialect-specific examples of frequent words, closed classes, examples, etc. The seed lexicon can grow as more work on a dialect is done to become eventually a dictionary of the dialect. As part of the effort to make the rules and lexicons easily definable and usable across all dialects in CODA*, we introduce a phonological representation that can be used to discuss and compare different dialectal entries in their respective lexicons, as phonological information is often obscured or generalized by CODA orthography. We also created a website (see Section 8.) for listing the rules and specification, and an easy-to-use interface for the seed lexicon, which currently includes multiple dialects.

In the next three sections, we present a summary of the phonological representation, the general rules and specifications, and the seed lexicon.

5. CODA* Phonological Representation

In discussing the many dialects included in CODA*, it is necessary to distinguish CODA orthography from the actual phonological properties of an utterance. Because CODA may conflate some phonological variation for the sake of pan-Arabic consistency, it is not ideal for describing dialectal phonological variation. For this purpose, we present the CAMEL[3] Arabic Phonetic Inventory (CAPHI).[4] Inspired by the International Phonetic Alphabet (IPA) and Arpabet (Shoup, 1980), CAPHI is an objective system for transcribing utterances in all dialects of Arabic in a simple, user-friendly fashion. The CAPHI inventory contains only ASCII symbols–avoiding the need to switch between multiple keyboard layouts–such that symbols intuitively correspond to the representative phone and are easily memorized. This makes it a far more attractive representation for our purposes than the otherwise popular IPA. Additionally, each transcribed phone is white space-separated with optional markers available for distinguishing morpheme (+) and word boundaries (#). This enables single phones to be written with multiple characters when it is intuitive to do so, as is the case for long vowels or affricates. The infrastructure is easily extendable to cover yet undiscovered phenomena.

[3]Computational Approaches to Modeling Language (CAMEL) Lab at New York University Abu Dhabi.

[4]The Arabic word كافي *kAfy* means 'sufficient'.

			Example			
CAPHI	IPA	Letter	CODA	CAPHI	Gloss	Dialect
p	**p**	**ب *b***	بري	p r i	price	Algiers
p.	**p$^{\varsigma}$**	**ب *b***	بمب	p. a m. p.	pump	Baghdad
b	**b**	**ب *b***	باب	b ee b	door	Sfax
b.	**b$^{\varsigma}$**	**ب *b***	يباباني	y aa b. aa n i	Japanese	Muscat
f	**f**	**ف *f***	فردي	f a r d i	single	Tripoli
f.	**f$^{\varsigma}$**	**ف *f***	فَكّ	f. a k k	he opened	Baghdad
v	**v**	**ف *f***	روندیفو	r o o n d ii v uu	appointment	Algiers
t	**t**	**ت *t***	تفاح	t i ff ee 7	apple	Beirut
t	t	ث *θ*	ثامن	t aa m i n	eighth	Jeddah
t.	**t$^{\varsigma}$**	**ط *T***	طبق	t. a b a g	dish	Riyadh
t.	t$^{\varsigma}$	ت *t*	فستان	f u s t. aa n	dress	Amman
d	**d**	**د *d***	جديد	j d ii d	new	Algiers
d	d	ت *t*	اتدشدش	2 i d d a sh d a sh	he got smashed	Cairo
d	d	ذ *ð*	ذيل	d ee l	tail	Beirut
d	d	ض *D*	ضحك	d i 7 i k	he laughed	Cairo
d.	**d$^{\varsigma}$**	**ض *D***	ضرس	d. i r s	tooth	Jeddah
d.	d$^{\varsigma}$	ت *t*	اتضرب	2 i d. d. a r a b	he got hit	Cairo
d.	d$^{\varsigma}$	د *d*	ضفدعة	d. u f d. a 3 a	frog	Cairo
d.	d$^{\varsigma}$	ظ *Ď*	ظهرية	d. u h r i y y a	shaddow	Jeddah
th	**θ**	**ث *θ***	ثياب	th y aa b	clothes	Salt
dh	**ð**	**ذ *ð***	ذراع	dh r aa 3	arm	Muscat
dh.	**ð$^{\varsigma}$**	**ظ *Ď***	ظاهر	dh. aa h i r	appearing	Abu Dhabi
dh.	ð$^{\varsigma}$	ذ *ð*	ذوق	dh. oo g	tasting	Baghdad
dh.	ð$^{\varsigma}$	ض *D*	ضغط	dh. a gh a t.	he pressed	Jerusalem
s	**s**	**س *s***	اساسي	2 a s aa s i	main	Rabat
s	s	ث *θ*	ثورة	s a w r a	revolution	Cairo
s	s	ز *z*	الزعيم	2 a s s e 3 ii m	the boss	Sana'a
s	s	ص *S*	صايغ	s aa y e gh	jeweler	Cairo
s.	**s$^{\varsigma}$**	**ص *S***	قصّة	q i s. s. a	story	Alexandria
s.	s$^{\varsigma}$	س *s*	سلطة	s. a l. a t. a	salad	Khartoum
z	**z**	**ز *z***	زورق	z a w r a 2	boat	Aleppo
z	z	ذ *ð*	ذبيحة	z a b ii 7 a	meat	Muscat
z	z	س *s*	اسبوع	2 i z b uu 3	week	Khartoum
z	z	ص *S*	صغير	z gh ii r	small	Beirut
z.	**z$^{\varsigma}$**	**ز *z***	جزمة	tsh i z. m. a	boot	Baghdad
z.	z$^{\varsigma}$	ض *D*	مضبوط	m a z. b uu t.	correct	Jeddah
z.	z$^{\varsigma}$	ظ *Ď*	عظيم	3 a z. ii m	great	Damascus
sh	**ʃ**	**ش *š***	شاذر	sh aa dh i r	blanket	Muscat
j	**ʒ**	**ج *j***	جاب	j ee b	he brought	Beirut
j	ʒ	ق *q*	طريق	t. a r ii j	road	Baghdad
ts	**ts**	**تس *ts***	فرتس	f r i ts	Fritz	MSA
ts	ts	ك *k*	سمك	s i m a ts	fish	Riyadh(B)
ts	ts	ج *j*	كتابج	k i t aa b i ts	your [2fs] book	Riyadh(B)
dz	**dz̄**	**دز *dz***	ايدز	2 ee dz	AIDS	Beirut
dz	dz̄	ج *j*	جزاير	dz aa y i r	Algeria	Algiers
dz	dz̄	ق *q*	طريق	t. a r ii dz	road	Riyadh(B)
tsh	**tʃ**	**تش *tš***	كتشب	k a tsh tsh a p	ketchup	Sana'a
tsh	tʃ	ج *j*	عيونج	3 y uu n i tsh	your eyes [2fs]	Doha
tsh	tʃ	ش *š*	شاف	tsh aa f	he saw	Doha
tsh	tʃ	ك *k*	سمك	s i m a tsh	fish	Basra
dj	**dʒ**	**ج *j***	موجود	m a w dj uu d	available	Khartoum
dj	dʒ	ق *q*	طريق	t. i r ii dj	road	Abu Dhabi
k	**k**	**ك *k***	فاكهة	f aa k h a	fruit	Doha
k	k	ق *q*	رقم	r a k a m	number	Jerusalem(R)
g	**g**	**ج *j***	جميل	g a m ii l	beautiful	Cairo
g	g	ق *q*	قال	g aa l	he said	Aswan
g	g	ك *k*	بنك	b. a n g	bank	Baghdad

			Example			
CAPHI	IPA	Letter	CODA	CAPHI	Gloss	Dialect
gy	**g^{j}**	**ج *j***	جليز	gy aa y i z	possible	Khartoum
q	**q**	**ق *q***	قطار	q i t. aa r	train	Fes
qh	**ɢ**	**ق *q***	رقم	r a qh a m	number	Khartoum
kh	**x**	**خ *x***	تَخت	t a kh i t	bed	Aleppo
kh	x	غ *ɣ*	غسالة	kh a s s ee l a	washer	Tunis
gh	**ɣ**	**غ *ɣ***	غاوي	gh aa w i	beautiful	Muscat
gh	ɣ	ر *r*	اريد	2 a gh ii d	I want	Mosul
7	**ħ**	**ح *H***	حبّ	7 a b b	he loved	Sfax
3	**ʕ**	**ع ـ *ـ***	عمّر	3 a m m a r	he filled	Rabat
2	**ʔ**	**ء *'***	غطاء	gh t. aa 2	blanket	Tunis
2	**ʔ**	**آ *Ā***	آمين	2 aa m ii n	amen	Abu Dhabi
2	**ʔ**	**أ *Â***	متأكد	m i t 2 a k k i d	sure	Amman
2	**ʔ**	**إ *Ǎ***	إجازة	2 i dj aa z e	vacation	Sana'a
2	**ʔ**	**ؤ *ŵ***	تساؤل	t a s aa 2 u l	question	Muscat
2	**ʔ**	**ئ *ŷ***	جائز	j aa 2 i z	possible	Mosul
2	ʔ	ق *q*	طريق	t. a r ii 2	road	Damascus
h	**h**	**ه *h***	مهم	m u h i m m	important	Aswan
m	**m**	**م *m***	مكن	m a k k a n	he gave	Sana'a
m	m	ن *n*	جنب	j a m b	besides	Jeddah
m.	**m$^{\varsigma}$**	**م *m***	ميّ	m. a y y	water	Damascus
n	**n**	**ن *n***	فنّ	f a n n	art	Baghdad
n	n	ـً *ã*	فعلاً	f i 3 l a n	indeed	Cairo
n	n	ـٌ *ũ*	جميلٌ	dj a m ii l u n	beautiful	MSA
n	n	ـٍ *ĩ*	غصبٍ	gh a s. b i n	forcefly	Abu Dhabi
n.	**n$^{\varsigma}$**	**ن *n***	ناي	n. aa y	flute	Damascus
r	**r**	**ر *r***	روج	r uu j	red	Tunis
r.	**r$^{\varsigma}$**	**ر *r***	فرنسي	f a r. a n s i	french	Khartoum
l	**l**	**ل *l***	لازم	l aa z i m	necessary	Riyadh
l.	**l$^{\varsigma}$**	**ل *l***	لطش	l. a t. a sh	he stole	Khartoum
w	**w**	**و *w***	ولد	w a l d	boy	Sana'a
y	**j**	**ي *y***	هايل	h aa y i l	great	Alexandria
y	j	ج *j*	جلس	y a l a s	he sat down	Abu Dhabi
i	**i**	ـِ *i*	مرشد	m u r sh i d	guide	Mosul
i	i	ة *ħ*	سمكة	s a m a k i	fish	Beirut
i	i	ه *h*	فواكه	f w ee k i	fruits	Beirut
i	i	ي *y*	نسي	n i s i	he forgot	Sana'a
ii	**ī**	**ي *y***	مزيكة	m a zz ii k a	music	Alexandria
e	**e**	ـِ *i*	باكر	b aa k e r	tomorrow	Muscat
e	e	ة *ħ*	ستّة	s i t t e	six	Damascus
e	e	ـَي *ay*	غيرهم	gh e r h o m	other than them	Cairo
ee	**ē**	**ـَي *ay***	ستّيشن	s t ee sh a n	station	Muscat
ee	ē	ـا *iA*	حاسب	7 ee s a b	he paid	Beirut
a	**a, æ**	ـَ *a*	جَرَّب	dj a r r a b	he tried	Sana'a
a	a, æ	ا *A*	سما	s a m a	sky	Cairo
a	a, æ	ة *ħ*	شيبة	sh ee b a	old man	Jeddah
a	a, æ	ه *h*	شافته	sh aa f i t a	she saw him	Doha
a	a, æ	ى *ý*	حمّى	7 o mm a	fever	Muscat
aa	**ā, æ:**	**ا *A***	دار	d aa r	house	Abu Dhabi
o	**o**	ـُ *u*	قُلت	2 o l t	I said	Cairo
o	o	ه *h*	جيبته	j i b t o	I brought it	Damascus
o	o	و *w*	الو	2 a l o	hello	Amman
oo	**ō**	**و *w***	دور	d oo r	turn	Damascus
u	**u**	ـُ *u*	جُبن	dj u b i n	cheese	Doha
u	u	ه *h*	جيبته	g i b t u	I brought it	Cairo
u	u	و *w*	كتابكو	k i t a b k u	your [2p] book	Cairo
u	u	وا *wA*	كنتوا	k i n t u	you [2p] were	Beirut
uu	**ū**	**و *w***	بلوزة	b l uu z a	blouse	Jeddah

Figure 2: A detailed listing of possible pairings of CAPHI and IPA sounds with CODA letters. Each pairing is accompanied with an example in terms of CODA, CAPHI, English gloss and city dialect. The dialect chosen is not meant to be exclusionary of other city dialects. The order of presentation is based on phonological features, and it positions consonants > vowels, stops > fricatives > affricates > nasals > other, labials >> glottals, voiceless > voiced, and non-emphatic > emphatic. Highlighted cells indicate CAPHI-CODA default pairing. (B) and (R) refer to *Bedouin* and *Rural* sub-dialects.

CAPHI phones include all phonemes in any dialect and any allophones which are confusable with phonemes of another dialect. For example, the voiceless alveolar stop [t] and its pharyngealized counterpart [t$^{\varsigma}$], are both phonemes because they can be used to distinguish meaning, as demonstrated by the minimal pair, تيّار [taj:aːr] 'current' and طيّار [t$^{\varsigma}$aj:aːr] 'pilot'. Thus both phones exist in CAPHI as /t/ and /t./ respectively. Conversely, [a$^{\varsigma}$] and [a] are not separate phonemes despite the fact that pharyngealized vowels do occur near pharyngealized consonants. Pharyngealized vowels, however, do not distinguish meaning and are not confusable with any phonemes in any known dialect. Therefore, they are not included in CAPHI, as little descriptive power would be gained from their inclusion and annotators would be required to make a challenging, error-prone distinction. Some allophones though, are included in CAPHI, like the Iraqi [p$^{\varsigma}$] from phoneme [p]. The pharyngealization on the voiceless bilabial stop causes this allophone to sound similar to the voiced bilabial stop [b] with

which it can be confused by speakers of dialects which do not have the phoneme [p]. Thus, p^{ς} is included in CAPHI as /p./ as it is useful in describing the dialectal differences between Iraqi and other dialects. The complete CAPHI inventory is listed in Figure 2.

6. CODA* General Rules and Specifications

While the goals of the CODA* guidelines is to precisely define the CODA choices, it is unavoidable that different versions of the guidelines will need to be presented differently for specific annotators on specific tasks for specific dialects: e.g., conversion form Arabizi to Arabic script (Bies et al., 2014), or lexicon construction (Diab et al., 2014).

In this paper, we summarize and highlight specific contributions of the effort; but the full set of CODA* guidelines is described on its online page (See Section 8.). We start with a description of the technical terminology we use; then we discuss the various rules and how to use them. The border between the general rules and the specification rules is broadly drawn along the lines that general rules do not refer to any specific lexical items (morphemes or words) and pertain to the meta-mechanics of CODA; while the specification rules are lexically specific, and at times ad hoc.

6.1. Terminology

We define the various technical terms we use in the rest of this section. For more information on Arabic morphology, see (Habash et al., 2012).

Sounds, Letters and Diacritics The term *sounds* is used in the context of pronunciation (phonology), while letters and diacritics are used in the context of writing (orthography). Sounds can be consonants or vowels. They are represented using the CAPHI representation and are bounded by forward slashes. *Letters and diacritics* are symbols used in the Arabic script to write words. Letters in the Arabic language are always required to be written; while diacritics are optional.[5]

Roots, Patterns, and other Morphemes Arabic's templatic morphology makes common reference to the concept of the *root*, a typically tri-consonantal abstraction capturing a general meaning about the word. For example, the root ك.ت.ب *k.t.b* 'writing-related' appears in words like مكتب *maktab* 'office' and كتاب *kitAb* 'book'. Each sound in the root is referred to as a *radical*. The general complement of the root is the pattern, which in the examples above are *ma12a3* and *1i2A3* (here, 1, 2, 3 are slots for the root radicals). In addition to the root and pattern templatic morphemes, Arabic uses numerous other concatenative morphemes.

Words, Basewords, and Clitics We define an Arabic *baseword* to consist of a stem and the minimal number of concatenative affixes needed to specify the obligatory features for its POS. A stem can be non-templatic or it can be composed from the interdigitation of a root and a pattern. The pattern may specify the features fully, as in broken plurals. Basewords are as such the smallest fully formed words. Examples include: كتابين *kitAb+yn* 'two books' and يكتبون *y+ktb+wn* 'they write'. Clitics are syntactically independent but phonologically dependent morphemes that are attached to the word phonologically. Words can be basewords or basewords with added clitics. We use the term *word* to refer to the phonological utterance or the orthographic string, and we specify as needed. In CODA, phonological words typically map one-to-one to orthographic words; but there are many exceptions, pertaining mostly to clitics that are spelled as separate orthographic words.

[5]While sounds are represented in CAPHI, letters and diacritics will be represented in Arabic script and supplemented with a romanized transliteration (Habash et al., 2007) for non-Arabic readers.

6.2. Determining CODA-Compliant Orthography

To construct the spelling of a word, one must first identify all of its components: from sounds to morphemes, basewords and clitics. The morphemes should not just be identified in terms of their form, but also in terms of their meaning and POS. Different rules, general and specific, will often apply to different parts of the word. In practice, we expect some users to try to identify the baseword and look it up in the seed lexicon first; otherwise, they should form the baseword then add the clitics and follow the rules of clitic attachment.

6.3. General Rules

The general CODA* rules are for the most part a subset of the original CODA rules (Habash et al., 2012) with minor simplifications. We summarize below some of the most important general rules.

6.3.1. Basic Phonology to Orthography Mapping

These rules cover the mapping from sounds to letters. The default mapping is indicated in Figure 2 (bolded sections). All other pairings in that table follow from other general and specification rules, some of which are discussed below.

Hamza Spelling Hamza (Glottal Stop) spelling follows from the same rules as those of MSA and the original CODA. The Hamza is represented in six letters that are conditioned on its phonological context. In previous versions of CODA, and in MSA spelling, baseword initial Hamza had complex rules for deciding its form. The rule is now simplified to ا *A* and considers the Hamzation (أ *Â*, إ *Ǎ*) optional.

Diacritic Spelling While Arabic diacritics are optional in general, they can be required in certain contexts, e.g., lemmas in the work of Khalifa et al. (2018) are diacritized. In this paper, we generally omit the diacritics unless needed. Arabic diacritics are primarily used for representing short vowels, or absence of vowels. However, the Shadda diacritic is used to represent consonantal gemination, e.g., كَتَّب *kat∼ab* /k a t t a b/ 'he dictated'. As such, using the Shadda interacts with the number of letters in a word. The Shadda general rule states that it is used within the baseword, but not across word-clitic boundaries. Any exceptions must be specified in the specification rules.

Long-Short Vowel Spelling In many dialects, baseword long vowels may be shortened in certain contexts. Gener-

ally, the rule is to prefer the *long* letter-based spelling over the shortened diacritic spelling.

6.3.2. Baseword Spelling

Unlike the first set of general rules discussed above, the next three rules make reference to the root and pattern morphemes.

Root Radical Spelling We expand the rule on etymologically spelled consonants discussed by Habash et al. (2012): dialectal word root radicals which have MSA cognates will be spelled using the MSA cognate radical if the dialectal radical sound and the MSA radical sounds are paired according to a specific set of common sound changes. Our expanded list of pairings is presented in Figure 3. Examples of specific words can be found in Figure 2.

		CAPHI	
CODA		MSA Sound	Cognate Dialect Variant Sound
ت	*t*	t	t.
ث	*θ*	th	t, t., s
ج	*j*	dj, g	j, y, gy, tsh
د	*d*	d	t., d.
ذ	*ð*	dh	d, dh., z
ر	*r*	r	gh
ز	*z*	z	s, s.
س	*s*	s	s., z
ش	*š*	sh	tsh
ص	*S*	s.	s, z
ض	*D*	d.	d, dh., z.
ط	*T*	t.	t
ظ	*Ď*	dh.	d., z.
ق	*q*	q	j, dz, dj, k, g, qh, 2
ك	*k*	k	ts, tsh, g
ن	*n*	n	m

Figure 3: Root Radical Map

Pattern Spelling Dialectal words with patterns that are cognates of MSA patterns will retain the spelling choice of the MSA pattern if the difference in pronunciation can be expressed using diacritics (for vowel change or absence), or if the pronunciation is a shortened form of the MSA pattern vowels.

Alif Maqsura The MSA rules for spelling the Alif-Maqsura (ى *ý*), which are sometimes based on roots and sometimes on patterns, apply in CODA*.

6.3.3. Clitic Spelling

The general rule on phonological clitic spelling is that clitics that are mapped into single letters (with possible diacritics) will be spelled attached to the word, and will not interact with the spelling of the word. The specification rules identify the exceptions to this rule.

6.4. Specification Rules

The CODA* specification rules are organized along the different POS tags, such as pronouns, conjunctions, demonstratives, etc.; and other word classes, such as number words and vocative familial expressions. We present next a few iconic examples of such specification rules. The full listing is part of the online CODA* guidelines. While in this section we use examples from specific dialects, the rules are dialect-independent. They, however, make specific reference to the morpheme POS, meaning, and pronunciation as the main determinants of how it is written in CODA.

6.4.1. The Definite Article

The Arabic definite article is always written as a proclitic +ال *Al+*, regardless of how it is pronounced. Table 2 presents a number of example cases with the definite article pronunciations bolded. As with MSA spelling, general cliticization rules apply except when following the proclitic +ل *l+*, where the article is spelled without its ا *A*. The Shadda rule is overridden in the specific context of +ال+ل *l+Al+* followed by an *l*-initial baseword (see last row in Table 2).

CODA	CAPHI	Gloss	Dialect
القمر *Alqmr*	**2 i l** 2 a m a r	'the moon'	Cairo
الشمس *Alšms*	**2 i sh** sh a m e s	'the sun'	Jerusalem
الكتاب *AlktAb*	**2 i k** k i t aa b	'the book'	Cairo
البيت *Albyt*	**l** b ee t	'the house'	Jerusalem
البيوت *Albyt*	**l e** b y uu t	'the houses'	Jerusalem
بالبيت *bAlbyt*	b e **l** b ee t	'at home'	Jerusalem
بالبيوت *bAlbywt*	b **l** e b y uu t	'at the houses'	Jerusalem
للبيت *llbyt*	l a **l** b ee t	'for the house'	Jerusalem
للبيوت *llbywt*	l a **l e** b y uu t	'for the houses'	Jerusalem
للشمس *llšms*	l a **sh** sh a m e s	'to the sun'	Jerusalem
للشموس *llšmws*	l a **l e** sh m uu s	'to the suns'	Jerusalem
اللجنة *Alljnħ*	**2 e l** l a g n a	'the committee'	Cairo
للجنة *lljnħ*	l e **l** l a g n a	'for the committee'	Cairo

Table 2: Definite Article examples.

6.4.2. The Ta-Marbuta

The *Ta-Marbuta* (ة *ħ*) is a secondary letter of the Arabic alphabet used to represent a particular suffix morpheme that is often (but not exclusively) associated with the feminine-singular feature (Alkuhlani and Habash, 2011). This morpheme has a number of allomorphs with differing pronunciations. Most notably, it appears as a vowel at the end of nominals, and changes to a ∼ /t/ when followed by possessive pronominal enclitics. The Ta-Marbuta should be written as ة *ħ* in word-final positions, regardless of its pronunciation, and following general CODA rules in non-word-final positions. See Table 3 for example cases.

CODA	CAPHI	Gloss	Dialect
حاجة *HAjħ*	7 aa g **a**	something	Cairo
حاجتي *HAjty*	7 aa g **t** i	my thing	Cairo
حاجتها *HAjthA*	7 aa g **i t** h a	her thing	Cairo
طاولة *TAwlħ*	t. aa w l **e**	table	Jerusalem
غزالة *γzAlħ*	gh a z ee l **i**	gazelle	Beirut
معلمة مدرسة	m 3 a l m **i t** # m a d r a s **e**	school teacher	Jerusalem
mςlmħ mdrsħ	m 3 a l m **e** # m a d r a s **e**	she taught a school	Jerusalem
معلمتهم *mςlmthm*	m 3 a l m **i t** h u m	their teacher	Jerusalem
معلماهم *mςlmAhm*	m 3 a l m **aa** h u m	she taught them	Jerusalem

Table 3: Ta-Marbuta examples.

6.4.3. The Plural Waw

Verbal suffixes that indicate the feature *plural subject* and end with the sounds (/u/, /uu/, /o/, /oo/, and /aw/) will represent those sounds as وا *wA* (واو الجماعة 'Waw of Plurality') in word-final positions, and as و *w* when followed by other attached clitics. This rule is similar to the MSA rule, except for expanding the phonetic definition. See Table 4 for example cases.

CODA	CAPHI	Gloss	Dialect
قالوا *qAlwA*	2 aa l **u**	they said	Cairo
بيقولوا *byqwlwA*	b i y 2 uu l **u**	they say	Cairo
نقولوا *nqwlwA*	n q uu l **u**	we say	Tunis
قالوا *qAlwA*	g aa l **a w**	they said	Abu Dhabi
قالوها *qAlwhA*	2 a l **uu** h a	they said it	Cairo
ما قالوش *mA qAlwš*	m a # 2 a l **uu** sh	they did not say	Cairo
قالوا له *qAlwA lh*	2 a l **uu** # l u	they said to him	Cairo

Table 4: The Plural Waw examples.

6.4.4. Negation Clitics

The negation particle (/m a/, /m aa/) has phonologically become a proclitic in many dialects. However, it is always written as a separate particle ما *mA* except when overridden by another specification rule. One example of such a rule is the case of negated pronouns, which require the ما *mA* to be attached to the pronoun stem and does not allow repeated *Alif* letters. Table 5 presents a number of example cases.

CODA	CAPHI	Gloss	Dialect
ما قال	**m aa** # 2 aa l	he did not say	Damascus
ما قالش	**m a** # 2 a l sh	he did not say	Cairo
ما بدناش	**m a** # b i d d n aa sh	we do not want	Amman
مانيش	**m a** n ii sh	I am not	Cairo
ماهياش	**m a** h i y y aa sh	she is not	Cairo

Table 5: Negation clitic examples.

6.4.5. Prepositional Enclitics

Post-verbal and post-nominal prepositions that have phonologically become enclitics will nonetheless be spelled separately from the words they follow. The most prominent such case is the preposition +ل *l+* 'to, for' which introduces indirect verb objects in a number of dialects. Table 6 shows are a number of example cases from the dialect of Cairo.

CODA	CAPHI	Gloss
قالوها لي *qAlwhA ly*	2 a l u h aa # **l** i	they said it to me
ما قالوا ليش *mA qAlwA lyš*	m a # 2 a l u # **l** ii sh	they did not say it to me
بالنسبة له *bAlnsbħ lh*	b i n n i s b aa # **l** u	as for him

Table 6: Prepositional enclitic examples.

6.4.6. Numbers

The words for numbers in Arabic dialects are amongst the most rich in phonological variety. The rules of writing number words in CODA* add the following exceptions to the general rules:

- The sometimes reduced historical Ta-Marbuta in the middle of the teens (11-19) is always written as ت *t* regardless of its pronunciation as /t/ or /t./. It is never reduced to a Shadda diacritic.
- The sometimes reduced historical ع ς /3/ in numbers such as عشر ςšr 'ten, -teen', and تسع *tsς* 'nine' will always be spelled as ع ς even if completely elided or turned into a vowel.
- The sometimes reduced or altered final letter of عشر ςšr 'ten, -teen' will be written as pronounced. The variation in this form marks different syntactic construction in some dialects.
- The hundreds will be written as a single word only if the *hundred* part is singular in form.
- The remnant /t/ of the historical Ta-Marbuta appearing only before Alif-initial words after number words will not be written.

The above rules apply to all number words, whether ordinal, cardinal, or fractions. Number words sometimes have different masculine and feminine forms that are used according to different dialect-specific rules. CODA guidelines do not interact with these dialect-specific decisions. Table 7 presents example cases, some of which involve interactions with other specification rules and general rules.

CODA	CAPHI	Gloss	Dialect
ثمانية *θmAnyħ*	th a m a n y e	eight	Salt
ثمانية *θmAnyħ*	t a m a n y e	eight	Amman
ثمانة *θmAnħ*	t a m aa n e	eight	Damascus
ثمان *θmAn*	t a m a n	eight X	Amman
ثمان *θmAn*	t m aa n	eight X	Damascus
ثمانتعش *θmAntςš*	t a m a n t a 3 sh	eighteen	Amman
ثمانتعش *θmAntςš*	t m a n t a 3 sh	eighteen	Damascus
ثمنتعش *θmntςš*	th m u n t. a 3 i sh	eighteen	Baghdad
ثمانتعشر *θmAntςšr*	t a m a n t aa sh a r	eighteen	Cairo
ثمانتعشر *θmAntςšr*	t a m a n t a 3 sh a r	eighteen X	Amman
ثمانتعشن *θmAntςšn*	th m a n t. aa sh e n	eighteen X	Tunis
اربعمية *Arbςmyħ*	2 a r b a 3 m i y y e	400	Amman
اربعمية *Arbςmyħ*	2 a r b a 3 m ii t	400 X	Amman
ربعمية *rbςmyħ*	r u b 3 u m i y y a	400	Cairo
اربع الاف *Arbς AlAf*	2 a r b a 3 # t a l aa f	4,000	Cairo
خمس ارباع *xms ArbAς*	kh a m a s # t i r b aa 3	five-fourths	Cairo

Table 7: Number examples.

6.4.7. Pronominal Enclitics

The set of specifications for the pronominal clitics which can serve as possessive pronouns, direct objects or indirect objects are presented in Table 8. Some of the decisions follow from the general rules, but for the most part they are intended to normalize the spelling as close as possible to the MSA variety without adding unnecessary and unresolvable ambiguity (e.g., using diacritics). It is important to point out again that this list is not dialect specific, but rather, it lists all the phonological forms of the pronominal morphemes in all dialects. The CODA* spelling for a dialect will depend on the phonology-morphology pair it corresponds to. Some of these pronouns have a large number of variants that can be ambiguous cross-dialectally. An interesting example is the case of the morpheme pronunciation /a/ which can be 3rd masculine singular in Gulf Arabic, but 3rd feminine singular in North Levantine: / k t aa b + a / can correspond to كتابه *ktAb+h* 'his book' (Abu Dhabi) or to كتابها *ktAb+hA* 'her book' (Damascus). The CODA* specification does not address how a particular dialect may organize the use of the different forms in terms of morphotactics, e.g., the possessive 2nd person singular feminine pronominal clitic is always ك+ *+k* in Tunis, and always كي+ *+ky* in Mosul; however, in Amman, it is كي+ *+ky*

post-vocalically, and ك+ *+k* otherwise. The underspecification of some features is intentional as some pronominal clitics may be used with different associated genders in different dialects, e.g., كن+ *+kn* is 2nd person plural feminine in Doha, but its is gender ambiguous in Beirut.

CODA	CAPHI	Morpheme Features
ني *ny*	n i, n e, n ii, n ee	1st Person Singular
ي *y*	i, ii, e, ee, y, y a, y e	1st Person Singular
ك *k*	k, i k, e k, k a	2nd Person Singular
كي *ky*	k i, k e, k ii, k ee	2nd Person Singular Feminine
ج *j*	tsh, i tsh, ts, i ts	2nd Person Singular Feminine
ه *h*	h, h u, u, o, a, a h, u h, *length*	3rd Person Singular Masculine
ها *hA*	h a, h aa, a, aa, h e, h ee	3rd Person Singular Feminine
نا *nA*	n a, n aa, n e, n ee	1st Person Plural
كم *km*	k u m, k o m	2nd Person Plural
كن *kn*	k u n, k o n, tsh i n	2nd Person Plural
هم *hm*	h u m, h o m, u m, o m	3rd Person Plural
هن *hn*	h u n, h o n, u n, o n	3rd Person Plural

Table 8: CODA* specifications for pronominal clitics.

6.4.8. Vocative Familial Expressions

Some of the vocative expressions used primarily for familial reference have vocalic endings that are homophonous with pronominal suffixes. These endings are spelled following the general phonology-to-orthography rules. For example, the word /3 a m m o/ in the dialect of Amman can mean 'uncle!' (spelled in CODA as عمو ʿ*mw*) or 'his uncle' (spelled in CODA as عمه ʿ*mh*).

7. CODA* Seed Lexicon

The CODA* seed lexicon is a large and growing database containing verified examples of CODA* spelling for dialectal words. The seed lexicon started, as per its namesake, as a starter kit for defining CODA for new dialects by considering earlier decisions. It minimally contains the closed class words from any dialect in it, in addition to any number of examples that come out of the specification rules (e.g., numbers, familial expressions, etc.). The current CODA* lexicon has 4,819 entries from 19 cities (average 253 per city). Some city dialects have more entries than others. As part of the work on the MADAR project, we are adding all the entries from MADAR's 25 cities, which are over 47,000 entries. Table 9 shows a few examples of the CODA* seed lexicon. The different columns in the table are as follows.

- **Category** identifies the type of the entry, as phrase, word, prefix, suffix, proclitic, or enclitic.
- **Lemma** is a dialect specific lemma that abstracts over the inflectional variants of the word.
- **CODA** is the spelling of the entry according to the CODA spelling guidelines.
- **CAPHI** is the phonological transliteration of the entry following the CAPHI guidelines in Section 5.
- **English** provides the lemmatized form of the English gloss for each entry.
- **POS** identifies the entry's part-of-speech tag following the CAMEL POS guidelines (Khalifa et al., 2018).
- **Dialect** identifies the city-based dialect for each entry.

Category	Lemma	CODA	CAPHI	English	POS	Dialect
WORD	برشا *bršA*	برشا *bršA*	b a r sh a	very	ADV	Tunis
WORD	قوي *qwy*	قوي *qwy*	2 a w i	very	ADV	Alexandria
WORD	قوي *qwy*	قوي *qwy*	2 a w i	very	ADV	Cairo
WORD	قوي *qwy*	قوي *qwy*	g a w i	very	ADV	Sanaa
WORD	كثير *kθyr*	كثير *kθyr*	k a t ii r	very	ADV	Aswan
WORD	كثير *kθyr*	كثير *kθyr*	k i t ii r	very	ADV	Cairo
WORD	كثير *kθyr*	كثير *kθyr*	k th ii gh	very	ADV	Mosul
WORD	كثير *kθyr*	كثير *kθyr*	k th ii r	very	ADV	Salt
WORD	كثير *kθyr*	كثير *kθyr*	k t ii r	very	ADV	Beirut
WORD	واجد *wAjd*	واجد *wAjd*	w aa y i d	very	ADV	Abu Dhabi
WORD	واجد *wAjd*	واجد *wAjd*	w aa g i d	very	ADV	Muscat
WORD	واجد *wAjd*	واجد *wAjd*	w aa j i d	very	ADV	Benghazi
WORD	قال *qAl*	قال *qAl*	g aa l	he said	VERB.P3MS	Abu Dhabi
WORD	قال *qAl*	نقول *nqwl*	n g uu l	we say	VERB.I1P	Abu Dhabi
WORD	قال *qAl*	نقول *nqwl*	n q uu l	I say	VERB.I1S	Tunis
ENC	ك+ *+k*	ك+ *+k*	i k	you	PRON.2FS	Jerusalem
ENC	ك+ *+k*	كي+ *+ky*	k i	you	PRON.2FS	Jerusalem
ENC	ج+ *+j*	ج+ *+j*	tsh	you	PRON.2FS	Abu Dhabi
PROC	+ش *š+*	+ش *š+*	sh a	will	PART_FUT	Sanaa
PROC	+ح *H+*	+ح *H+*	7 a	will	PART_FUT	Amman
PROC	+ه *h+*	+ه *h+*	h a	will	PART_FUT	Cairo
PROC	+غ *γ+*	+غ *γ+*	gh a	will	PART_FUT	Rabat
PROC	+ب *b+*	+ب *b+*	b	will	PART_FUT	Manama

Table 9: Examples from the CODA* seed lexicon.

The table also shows how the same word, with the same lemma and CODA representation, can be mapped to multiple phonological representations from different dialects. For example, the word كثير *kθyr* 'very' is mapped to five different phonological representations. In the online browsable version of the seed lexicon there are extra comments and notes indicating which rules were used.

8. Conclusion and Outlook

We presented a unified set of guidelines and resources for conventional orthography of dialectal Arabic. These guidelines and their connected resources are being used by three large Arabic dialect processing projects in three universities working on dialects from 28 Arab cities.

The resources are all available online at the project website: `http://resources.camel-lab.com/`.

In the future, we plan to continue expanding our guidelines and resources for the Arabic dialects we are working with and add new dialects. We also plan to annotate collections of text in their CODA forms to train and benchmark systems for automatic spelling conventionalization.

Acknowledgments

This effort has been supported by the Gulf Arabic Morphological Annotation project (New York University Abu Dhabi – research enhancement fund), and the Multi-Arabic Dialect Applications and Resources (MADAR) project (grant NPRP 7-290-1-047 from the Qatar National Research Fund – a member of Qatar Foundation). All statements made herein are solely the responsibility of the authors.

Bibliographical References

Abu Ali, D. and Habash, N. (2016). Botta: An arabic dialect chatbot. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 208–212, Osaka, Japan, December.

Al-Badrashiny, M., Eskander, R., Habash, N., and Rambow, O. (2014). Automatic Transliteration of Romanized Dialectal Arabic. *CoNLL-2014*, page 30.

Al-Shargi, F., Kaplan, A., Eskander, R., Habash, N., and Rambow, O. (2016). A Morphologically Annotated Corpus and a Morphological Analyzer for Moroccan and Sanaani Yemeni Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Ali, A., Mubarak, H., and Vogel, S. (2014). Advances in dialectal arabic speech recognition: A study using twitter to. In *Improve Egyptian ASR. International Workshop on Spoken Language Translation (IWSLT*. Citeseer.

Alkuhlani, S. and Habash, N. (2011). A Corpus for Modeling Morpho-Syntactic Agreement in Arabic: Gender, Number and Rationality. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL'11)*, Portland, Oregon, USA.

Arkadiusz, P. (2006). Le nationalisme linguistique au liban autour de sa'id 'aql et l'idée de langue libanaise dans la revue "lebnaan" en nouvel alphabet. *Arabica*, 53(4):423–471.

'Asaakir, K. (1950). A Method for Writing Modern Arabic Dialects with Arabic Letters. (in Arabic). *The Arab Academy Magazine*, 8(181–192), January.

Badawi, E.-S. and Hinds, M. (1986). *A Dictionary of Egyptian Arabic*. Librairie du Liban.

Bies, A., Song, Z., Maamouri, M., Grimes, S., Lee, H., Wright, J., Strassel, S., Habash, N., Eskander, R., and Rambow, O. (2014). Transliteration of arabizi into arabic orthography: Developing a parallel annotated arabizi-arabic script sms/chat corpus. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 93–103.

Bouamor, H., Habash, N., Salameh, M., Zaghouani, W., Rambow, O., Abdulrahim, D., Obeid, O., Khalifa, S., Eryani, F., Erdmann, A., and Oflazer, K. (2018). The MADAR Arabic Dialect Corpus and Lexicon. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, May.

Darwish, K. (2013). Arabizi Detection and Conversion to Arabic. *CoRR*.

Diab, M., Habash, N., Rambow, O., AlTantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *Proceedings of the LREC Workshop for Language Resources (LRs) and Human Language Technologies (HLT) for Semitic Languages: Status, Updates, and Prospects*.

Diab, M. T., Al-Badrashiny, M., Aminian, M., Attia, M., Elfardy, H., Habash, N., Hawwari, A., Salloum, W., Dasigi, P., and Eskander, R. (2014). Tharwa: A large scale dialectal arabic-standard arabic-english lexicon. In *LREC*, pages 3782–3789.

Erdmann, A., Habash, N., Taji, D., and Bouamor, H. (2017). Low Resourced Machine Translation via Morpho-syntactic Modeling: The Case of Dialectal Arabic. In *MT Summit*.

Eskander, R., Habash, N., Rambow, O., and Tomeh, N. (2013). Processing Spontaneous Orthography. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, Atlanta, GA.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic Transliteration. In A. van den Bosch et al., editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, N., Diab, M., and Rabmow, O. (2012). Conventional Orthography for Dialectal Arabic. In *Proceedings of LREC*, Istanbul, Turkey.

Habash, N. Y. (2010). *Introduction to Arabic natural language processing*, volume 3. Morgan & Claypool Publishers.

Jarrar, M., Habash, N., Akra, D., and Zalmout, N. (2014). Building a Corpus for Palestinian Arabic: a Preliminary Study. *ANLP 2014*, page 18.

Jarrar, M., Habash, N., Alrimawi, F., Akra, D., and Zalmout, N. (2016). Curras: an annotated corpus for the Palestinian Arabic dialect. *Language Resources and Evaluation*, pages 1–31.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016). A Large Scale Corpus of Gulf Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Khalifa, S., Habash, N., Eryani, F., Obeid, O., Abdulrahim, D., and Kaabi, M. A. (2018). A morphologically annotated corpus of emirati arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, May.

Maamouri, M., Buckwalter, T., and Cieri, C. (2004). Dialectal Arabic Telephone Speech Corpus: Principles, Tool design, and Transcription Conventions. In *NEMLAR International Conference on Arabic Language Resources and Tools*.

Maamouri, M., Bies, A., Kulick, S., Ciul, M., Habash, N., and Eskander, R. (2014). Developing an egyptian arabic treebank: Impact of dialectal morphology on annotation and tool development. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Meftouh, K., Harrat, S., Jamoussi, S., Abbas, M., and Smaili, K. (2015). Machine translation experiments on PADIC: A parallel Arabic dialect corpus. In *The 29th Pacific Asia conference on language, information and computation*.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Saadane, H. and Habash, N. (2015). A Conventional Orthography for Algerian Arabic. In *ANLP Workshop 2015*, page 69.

Shoup, J. E. (1980). Phonological aspects of speech recognition. *Trends in speech recognition*, pages 125–138.

Watson, J. C. (2007). *The Phonology and Morphology of Arabic*. Oxford University Press.

Zaghouani, W., Mohit, B., Habash, N., Obeid, O., Tomeh, N., Rozovskaya, A., Farra, N., Alkuhlani, S., and Oflazer, K. (2014). Large scale Arabic error annotation: Guidelines and framework. In *International Conference on Language Resources and Evaluation (LREC 2014)*.

Zalmout, N., Erdmann, A., and Habash, N. (2018). Noise-robust morphological disambiguation for dialectal arabic. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2018)*.

Zribi, I., Boujelbane, R., Masmoudi, A., Ellouze Khmekhem, M., Hadrich Belguith, L., and Habash, N. (2014). A Conventional Orthography for Tunisian Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Automatic Identification of Maghreb Dialects Using a Dictionary-Based Approach

Houda Saâdane*, Hosni Seffih+, Christian Fluhr*, Khalid Choukri∓, Nasredine Semmar±
*GEOLSemantics, 12 Avenue Raspail, 94250 Gentilly, France
+LIASD, Université Paris 8, 2 rue de la liberté 93526 Saint-Denis, France
∓ELDA / ELRA , 9 Rue des Cordelières, 75013 Paris, France
±CEA, LIST, Vision and Content Engineering Laboratory, F-91191, Gif-sur-Yvette, France
{houda.saadane,hosni.seffih,christian.fluhr}@geolsemantics.com
choukri@elda.org, nasredine.semmar@cea.fr

Abstract

Automatic identification of Arabic dialects in a text is a difficult task, especially for Maghreb languages and when they are written in Arabic or Latin characters (Arabizi). These texts are characterized by the use of code-switching between the Modern Standard Arabic (MSA) and the Arabic Dialect (AD) in the texts written in Arabic, or between Arabizi and foreign languages for those written in Latin. This paper presents the specific resources and tools we have developed for this purpose, with a focus on the transliteration of Arabizi into Arabic (using the dedicated tools for Arabic dialects). A dictionary-based approach to detect the dialectal origin of a text is described, it exhibits satisfactory results.

Keywords: Arabic Dialects, Arabizi, Transliteration, Identification of Dialects

1. Introduction

The sociolinguistic situation for the Arabic language is characterized by the use of two varieties of one language, which contributes to having a diaglossic conception for this language. This is evidenced by the use of modern standard Arabic (MSA) in the educational, religious and literary fields and in some medias. On the other hand, a large number of dialects are used as mother tongues for many Arabic-speaking populations. In fact, they are the main communication tool spoken in everyday life through informal conversations, exchanges on SMS, forums and social networks, even in e-mails. These dialects vary from one country to another, one region to another, or even from one city to another. In addition, they differ from each other by important phonological, morphological, lexical and syntactic characteristics.

The processing of informal texts has become an extremely popular field of research among researchers (Yang and Eisenstein, 2013). For Arabic NLP, the identification of dialects is very important and considered as a prepossessing step for any natural language application dealing with Arabic language, such as machine translation, information retrieval for social media. It is sometimes considered as a difficult case of language identification where, according to (Zaidan and Callison-Burch, 2011) it is applied to a group of closely related languages that share a common character set. This identification is made even more complex by the absence of orthographic conventions, by the transliteration of Arabic dialects into Latin script (Arabizi) and also the use of code-switching.

Recent works have proposed both supervised and unsupervised statistical approaches to language identification. However, current methods are based on the assumption that dedicated resources exist, such as large corpora and dictionaries. Unfortunately, these resources are rarely available for certain languages and dialects, especially for Maghreb dialects.

Dialect processing cannot reuse the generic tools and techniques that are employed for processing and analyzing MSA texts. Detecting dialects requires to develop techniques and approaches to classify texts written with Arabic or Arabizi.

This paper's main focus is the automatic identification of Algerian, Tunisian, Moroccan and Egyptian Arabic dialects, written in Arabic and latin scripts, in the context of online comments and social media platforms. The paper is organized as follows: Section 2 presents main differences between MSA and dialects. Section 3 summarizes the related work in the domain of Arabic dialects identification. We describe in Section 4 our linguistic resources used to create the corpora and lexicon. In Section 5 we present the developed linguistic method for classifying Arabic dialects. We describe in Sections 6 a set of experiments conducted to evaluate our system, followed by a discussion about the obtained results. Finally, we present in Section 8 the conclusion and future work.

2. MSA and Dialects Differences

In the latest decades, many works focused on proposing new stratifications of the Arabic dialects. Hence, many classifications were proposed based on several criteria, like geography or social specificities. These works constitute a considerable part of the Arabic dialectology domain which considers that a dialect is a part of one of the following families:

- ***Western zone (Maghreb):*** North African group, with dialects of Algeria, Morocco, Tunisia, Libya and Mauritania.
- ***Eastern zone (Mashriq):*** with dialects from Egypt, Syria and the others middle-east countries like Iraq, Golf states, Yemen, Oman, Jordan, etc.

However, this classification was refined, giving a new typology which was accepted by many researchers, such as (Versteegh and Versteegh, 2001; Habash, 2010). This typology divides Arabic dialects into five major dialectal areas (also called geolects) from Eastern to Western as follows: (i) dialects of the Arabian Peninsula (Golf), (ii) Mesopotamian dialects (Iraqi), (iii) The Levantine dialects, (iv) the Egyptian dialects, and (v) the Maghreb dialects.
We note that these dialects are declined into variants, which have some features characterizing national dialects (natiolectes), or more granular regional features for regional dialects (regiolectes) or local features for local dialects (topolectes). The considered granularity is entirely geographical ranging from the nation to the village passing through the region and the city (Saâdane, 2011).
Whatever the dialect, it presents striking differences with MSA. In the following examples we describe these differences according to four levels: phonological, orthographic, morphological and lexical.

- **Phonology:** The consonant ق [q] in MSA deserves special attention. This sound has many varieties of pronunciation in dialects: it is pronounced (i) ق in Algerian, Moroccan and Tunisian dialect, (ii) ڤ [g] in Maghreb and in some Eastern Bedouin dialects, (iii) glottal stop in Egyptian, Levantine, Gulf states Arabic. There is also the glottal stop phoneme, which appears in many words in MSA as opposed to dialects. As an example, we give the following words: فأس*faAŝ* becomes فاس *faAs*[1] "pick", ذئب *Diŷb* becomes ذيب *Diyb* "wolf".

- **Orthographic:** Unlike MSA, dialects do not have an orthographic standard. We find many orthographic variations in the writing of words. These variations are mainly due to phonological differences between MSA and Arabic dialects . In some cases, phonology or underlying morphology results in regular phonological assimilation writing, for example, من بعد *man baçd* "after" also written مم بعد *mam baçd*. To remedy this lack of norm, work has been carried out to propose a Conventional Orthography for Dialectal Arabic (CODA), first proposed for Egyptian (Habash et al., 2012), then extended to other dialects such as Tunisian (Zribi et al., 2014), Algerian (Saâdane and Habash, 2015), Maghrebi Arabic (Turki et al., 2016) and Palestinian (Habash et al., 2016).

- **Morphology:** There are many morphological differences between dialects and MSA. These differences can be seen through several aspects. One of these aspects is the future particle which appears as + س *sa* + or سوف *sawfa* in MSA, which is expressed in: (i) + ح+ *Ha* + or رح *raH* in the Levantine dialects, (ii) ه+ *ha* + in Egyptian dialect, (iii) ك+ *ka* + in Moroccan dialect and; (iv) باش *baAŝ* in the Tunisian dialect. We also note that many dialects add new clitics that do not exist in MSA, such as the negation particle enclitic ما +*mA*+ ... +ش +*ŝ* which is expressed in MSA with various particles such as ما *mA*, لم *lam*, لن *lan* "do ... not". For instance the sentence ما قريتش *mA qriytiŝ* means "I have not read".

- **Lexical:** In this respect we can find a significant number of differences between dialects and Arabic MSA. For example, the MSA word الآن *AlÃn* "now" is expressed دلوقتي *dilwaqtiy*, توى *tawaý*, دابا *daAbaA* and دركا *durkaA* in the Egyptian, Tunisian, Moroccan and Algerian dialects respectively. There are also two other lexical aspects characterizing dialects: derivation and borrowings (Saâdane and Habash, 2015). The Algerian dialect, like other Arabic dialects, has been influenced over the centuries by other languages such as Amazigh, Turkish, Italian, Spanish and French. For example, let us consider the following words: قرجومة *Qarjuwma* "gorge" borrowed from the Berber, سكارجي *sukaArjiy* "drunk" borrowed from the Turkish, زبلة *Zablah* "fault" borrowed from the Italian, سبردينة *Spardiynah* "Espadrille", تيليفون *Tiyliyfuwn* "Telephone" from French.

  Arabic dialects differ greatly from MSA on the phonological, orthographic, morphological and lexical levels. Thus, to determine which dialect should be processed in a text, it is unavoidable to use criteria related to these levels of analysis. In particular, we mention that there are important vocabulary differences from one dialect to another. In fact, this difference is the basis of many methods of dialect detection.

3. Related Work

The creation of resources and development of methods to deal with Arabic dialects have attracted the attention of many researchers in the last few years. The aim is to compensate the lack of resources for dialectal Arabic, which are crucial for the development of adequate NLP tools. In (Zaidan and Callison-Burch, 2011) the authors collected a corpus based on texts available on the web, from three Arabic newspapers of Levantine, Gulf and Egyptian dialects. Articles and their comments were extracted to build the corpus.
We can find several other corpora for Arabic dialects, such as (Al-Sabbagh and Girju, 2012) who created an annotated corpus of Egyptian, but only a small subset of it was manually annotated to build a classifier, the rest of the corpus being automatically annotated. Other initiatives aimed to create a dialectal Arabic dataset to address the lack of dedicated resources such as (Cotterell and Callison-Burch, 2014), where the authors collected a significant amount of dialectal data from comments, online journals and Twitter for Egyptian, Gulf, Levantine, Algerian and Iraqi.

[1]Arabic transliteration is presented in the Habash-Soudi-Buckwalter (HSB) scheme (Habash et al., 2007).

The work presented in (Elfardy and Diab, 2012b) suggests guidelines for the foundation of large corpora of mixed Arabic resources with switching code. In addition to the former work, an identification, interpretation and classification system for dialects was introduced in (Elfardy and Diab, 2012a) called AIDA (Automatic Identification and Glossing of Dialectal Arabic). In the continuation of AIDA, the authors of (Elfardy and Diab, 2013) presented a supervised approach for the identification of dialectal sentences. They also studied the effects of preprocessing techniques on the accuracy of the developed classifiers.

As far as we know, there are few tools for automatically processing Arabizi. Works presented in (Darwish, 2014) and (Eskander et al., 2014), aimed at distinguishing English from Arabizi, resulting in a transliteration of texts from Arabizi to Arabic, which allows to process these texts with NLP systems dedicated to Arabic. (Adouane et al., 2016) considered the task of automatic identification of dialects as a classification problem and used supervised machine learning techniques to recognize Arabized Berber and Arabic dialects.

A review of methods and obtained results for the processing of Arabic dialects was presented in (Shoufan and Al-Ameri, 2015). Four types of tasks are described: basic analyzes, resource building, semantic analysis and identification of dialects. We can see that the approaches are generally divided into two main categories: dialectal systems built from dedicated resources and systems made by adaptation of available resources for MSA.We note that most works focus on Egyptian and Levantine. For Maghreb dialects, there is a significant lack of resources. We also note that the resources presented above are generally not available, which makes it difficult to reproduce experiments and compare results. In our work, we are primarily interested in the creation of resources for the Maghrebi dialects and more specifically in the automatic identification of the dialects written in Arabic and Latin characters: these are real problems that call for concrete solutions.

4. Dictionary and Corpus Creation

Our corpus used in this paper was created using essentially two sources: i) comments of reader extracted from diffrent online Arabic newspapers, and ii) exchanges extracted from various social media platforms. The choice of the online newspapers and social media was based on the results of web queries on some keywords that are dialect words written in Arabic and Arabizi. The retained results are those where the comments are more expressed in dialects. Finally, we note that the used keywords are provided by native speakers from the countries of the considered dialects: algerian, tunisian, morrocan and egyptian. This technique of corpus extraction allows us to consider various topics and subjects like sport, health, etc . Similar to (Zaidan and Callison-Burch, 2011; Saâdane et al., 2013), we also extract the following information for each comment, whenever available:

- The URL of the newspaper article.
- The author ID associated with the comment.
- The subtitle header.
- The author e-mail address.
- The date and time of the comment.
- The Commentary Contents.

The structure of the extracted information for each comment is presented as follows:

```
<doc docid="elkhabar_comment1" articleURL="http://www.elkhabar.com/ar/autres/athman_snadjki/240186.html" author="1-RABIE" pays="MARSEILLE" date="2011-01-01" time="13:31" >
<comment> ALLAH YARHMEK. INA LILLAH WA INA ILAYHI RAJI3OUN </comment>
</doc>
```

The size and the amount of the used copus per dialect are depicted in Table 1.

To deal with the problem of the lack of ressources for dialects, we adopt an approach of constructing the ressources by exploiting MSA/dialect similarities and addressing known differences. Indeed, we first study the phonological, morphological and lexical differences using the MSA. This step is realized after constructing a first set of lemmas by asking diffrent native speakers to give the equivalent MSA lemmas in dialects. Then we develop rules and build dialectal concepts (lemmas, patterns and roots) using the identified differences. Finally, based on the developed rules and concepts we construct automatically our dictionaries by using flexion. In order to give an example for the classification of lemmas that we have built, we focused our explanation on the creation of verbal patterns for dialectal verbs. We based this example on three criteria for classification the verbs, as following:

- **Verb model oriented classification:** We first started by identifying the dialectal verbs that still unchanged during the transition MSA / dialect and keep the same model. This is the case of the following word MSA: سَافَر *saAfar* "travel" with a Pattern-MSA: CaACaC following the model: CVACVC, the corresponding EGY word is سَافِر *saAfir* which follows the model: CVACVC and the EGY-pattern: CaACiC. The second step consists on seeking the verbs that completely change their form when transiting from MSA to dialect. For instance: MSA: بَحَثَ *baHaT* "search", the corresponding word in TUN: لَوَّجْ *law∼aj* which follows the model: CVC∼VL. This model looks like the one of the word كَسَّرْ *kas∼ar* "break": CVC∼VL . We have already assigned to كَسَّرْ *kas∼ar* the pattern TUN- II: CaC∼aC. Therefore, the verb لَوَّجْ *law∼aj* will have the same pattern. In the last step, we define forms for those verbs whose associated patterns are not identified by the previous steps. For example, for the word MSA: إِحْمَرَّ *(?i)hmar∼(-a)* "to blush"

	Dialects			
	DZ	MA	TN	EG
#comments	326K	120K	102K	599K
#sentences	794K	286K	201K	1.4M
#words	10.7M	6.4M	4.8M	32.4M

Table 1: Developed Corpora Features

the corresponding word in ALG is : إحْمَارْ *(?i)hmaAr* following the model: AiCVCVAC. We associated to this type of verbs the patterns named **"exception patterns"**.

- **Pattern's second consonant vowel oriented classification:** In Arabic, the vowel of the pattern's second consonant is one of the basic deterministic elements of the verbal morphology. According to (Ouerhani, 2009), this vowel is considered as the first criterion for classifying a verb in MSA, both in past and present. In dialect, this variation is very common and it is marked not only in the MSA pattern I but in all patterns (Boujelbane et al., 2013). For example, for the Pattern-TUN II three new sub-patterns emerged: II-aa: CaC∼aC/yiCaC∼aC ; II-ai: CaC∼aC/yiCaC∼iC ; TUN II-ii: CaC∼iC/yiCaC∼iC.

- **Imperfect mark oriented classification:** In the Imperfect form of Arabic grammar, the mark of the verbs inside the same scheme is stable and stills unchanged. For instance, at the level of the scheme I, the verbs in the Imperfect form begin always with the prefix يـ"ya" as in the verbs: يَضْرِبُ *ya-Drib(-u)* "hit", يَخْرُجُ *ya-khruj(-u)* "go out" and يَحْزَنُ *yahzan(-u)* 'be sad'. This specificity is not valid for dialects because this mark varies, even inside the same schema. For example, the word يِرْبَحْ - رْبَحْ *rbaH - yirbaH* "to win" follows the ALG –pattern-I-aa; the word كْتَبْ - يَكْتَبْ *ktab- yaktab*, "to write" follows the ALG-I-aa.We remark that the two verbs came from the same class but they have not the same imperfect marks.

  This is why we propose to extend the ALG pattern-I-au in order to define for the pattern-I more *sub-patterns*. To get this goal, we attribute to رْبَحْ - يِرْبَحْ *rbaH - **yi**-rbaH* the scheme ALG-I-aa-**i** and to كْتَبْ - يَكْتَبْ *ktab- **ya**-ktab* the scheme ALG-I-aa-**a**.

This approach was also followed by (Shaalan et al., 2007) for Egyptian dialect and (Boujelbane et al., 2013) for Tunisian dialect. Table 2 gives some statistics about the final lexicons per dialect.

5. Dictionary Based Dialect Detection System

5.1. Dialectal Data Annotation Format

We developed a system that automatically assign, for each word of the text, the following labels:

MSA	DZ	TN	MA	EG
2245136	58237	43690	42282	34859

Table 2: Statistics about the lexicons

- **Lang1:** Word in MSA written in the Arabic script. It also annotates in Arabizi texts the foreign words which keep their orthographic forms as the words "normal" of French or "good" of English.

- **Lang2:** Dialect word <AD> in Arabic or Arabizi script and information for Arabic text: <DZ> *(Algerian)*,<TN> *(Tunisian)*, <MA> *(Moroccan)* or <EG> *(Egyptian)*. This information is added in the arabizi script after their transliteration.

- **Named Entity :** A named entity, such as الجزاير *Al-jazaAyir* "Algeria".

- **Other:** Punctuations, numbers, sounds and emoticons, URL, etc.

- **Ambiguity:** Word where the class (lang1 or lang2) cannot be determined according to the current context (e.g. طيب *Tyb* can be used in MSA "good" and in Egyptian dialect "ok").

5.2. Arabizi to Arabic Transliteration

After annotating a dialectal text written in Arabizi, it is automatically converted into Arabic script, following the Conventional Orthography for Dialectal Arabic (CODA) (Habash et al., 2012). This Arabizi-to-Arabic script transliteration focuses first on the named entities using a system (Saâdane et al., 2012) which converts texts from Arabizi to Arabic, and vice versa, using finite-state transducers. The possible transliterations are filtered using a morphological analyzer of Arabic. For words that are not named entities, a transliterator using a rule-based approach and specific dictionaries for each dialect is used. The dictionaries contain 24,451 pairs of Arabizi-to-Arabic correspondences (words, but also bigrams or trigrams) distributed among the studied dialects. These correspondences were established by Arabic native speakers who worked on patterns exhibiting the greatest number of occurrences in corpora. The proposed rules allow to define the boundaries of the letters in an Arabic word and to apply exception rules for each word in order to remove the spelling variants that are not attested in use. For example, the automatic transliteration from Arabizi to Arabic script of the following sentence:

<**Arabizi**>Hadi 3afsa chaba bezzef fi dzayer</**Arabizi**>
<**Arabic**>هَادِي|هَاذِي عَفسَة | عَفسَا | عَفسَى|عَفصَة|عَفصَا دزَاير فِي بَزَّاف شَبَة|شَابَة عَفصَى |</**Arabic**>
<**Meaning**>This is a very beautiful thing in Algeria</**Meaning**>

Arabizi	Hadi	3afsa	chaba	bezzef
Arabic	هَاذِي	عَفسَة	شَابَة	بَزَّاف
Dialect	DZ,MA,TN	DZ	MSA,DZ	DZ

Table 3: Filtering of the best candidate using a morphological analyze

The generated list is then filtered using a morphological analyzer to predict whether the word belongs to one of the studied dialects (Table 3).
The conversion of Arabizi into Arabic script is an important step, but this article focuses on the identification of dialects and due to lack of space we cannot detail this part of the processing. Note, however, that this process adds a crucial information to identify the dialect: the presence of vowels in Arabizi makes it possible to indicate diacritics in Arabic (which are generally omitted) thus providing decisive clues to determine the dialect of a text.

5.3. Ambiguous Annotation System for Dictionary-based Words

The originality of our annotation approach is in the production of diacritical morphological analysis. The Maghreb dialects produce indeed many morphemes and lexical elements which are quite similar, and often differ only in short vowels. If this information is not written in arabic script, it is present in Arabizi, where diacritics are written, which helps us in the recognition of a dialect. Our annotation approach is summarized as follows:

- **Preprocessing:** The text is cleaned to separate punctuation and numbers attached to the words, to normalize the effects of the lengthening of the letters, to detect URLs and numbers and finally to tag punctuation, emoticons and sounds.

- **Morphological analysis:** Implements segmentation, lemmatization and labeling to determine morphological features and whether a word belongs to MSA, a dialect (DZ, TN, MA or EG), French or English. We use the system proposed in (Fluhr et al., 2012) for the identification of foreign words, named entities and dialect words (Out-of-vocabulary) in Arabizi texts. After a step of transliteration and labeling of Arabizi words and proper nouns, we apply a morphological analysis on the obtained Arabic text, which is annotated via the system presented in (Saâdane, 2015)(Saâdane, 2013).

- **List of named entities:** We use ANERGazet (Benajiba and Rosso, 2007) and GeoNames resources to identify the named entity in Arabic, French and English. Our resources are divided likewise:

  - **Persons:** 7,387 entries for person names,
  - **Locations:** 73,892 entries for geographical entities (countries, cities, continents names, etc.),
  - **Organizations:** 22,772 entries for names of organizations (companies, football teams, etc.).

- **Combination:** The combining step is used to aggregate multiple components, including dictionaries of named entities and language templates, in order to perform the recognition and the identification of named entities and language. Each word of the input sentence can be tagged with different labels from the previous steps. Thereby the combining step, based on the generated labels, uses a set of decision rules to assign the final tag to each word in the analyzed sentence. The decision rules used are presented in (Elfardy et al., 2014; Saâdane, 2015), and summarized as follows:

  - If the word contains only numbers or punctuation, it is associated with the *other tag* (Punct, Num, etc);
  - If the word is present in one of the dictionaries or if the GEOL parser assigns the named entity tag, then the word is labeled as *Named Entity NE*;
  - If the word is identified by the Morphological Analyzer to be tagged with *Lang1* or *Lang2*, the word is then associated with the corresponding label;
  - If the word identified is associated with both Lang1 and Lang2, then we assign to the word the tags *Lang1* and *Lang2*. However, this case adds ambiguity;
  - If the Morphological Analyzer did not label the word, then we assign the tag *UNK*. We find this situation in the case of a word considered out of the vocabulary.

5.4. Disambiguation of Dialects

After annotating the words of the analyzed messages, dialect detection system has now to determine dialect of the analyzed corpora or texts. For this, the only available indicators are the ambiguous dialect annotations presented previously. Two methods are proposed.
The first is based on the number of discriminant words. Its principle is that the presence of a discriminating word in a short text gives a good idea of the dialect. This approach allows returning the number of occurrences of each tag as well as the detected dialect. The following example: <**EG**> علشان ***(çalaŝAn)*** </**EG**> <**EG**> النهاردة ***(AlnahArdh)*** </**EG**> <**MSA**> الناس ***(AlnaAs)*** </**MSA**> <**DZEGMATN**> رايحين ***(rayHyn)*** </**DZEGMATN**>, shows that there are two discriminating words belonging only to the Egyptian dialect, as well as a word in the MSA which is the last common word to several dialects, therefore the sentence is tagged as Egyptian. The major disadvantage of this method is that detection is impossible if the message contains no specific word for a given dialect, or equal counts for multiple dialects.
The second method is based on a notation system, which sums the weights associated with the detected dialects of all text words. For a given word, the weight of a dialect is inversely proportional to the number of dialects detected. This method returns the most important dialect

Dialect (#ar/#lat)	Dictionary approach
	LER (#err-ar/#err-lat)
ALG	18.6% (#104/#82)
EGY	04.9% (#30/#19)
MAR	22.8% (#130/#98)
TUN	20.1% (#94/#107)
All	16.6% (#358/#306)

Table 4: Error rate for the linguistic method

	Dictionary Approach (#ar#lat)			
	Hypothesis			
	ALG	**EGY**	**MAR**	**TUN**
ALG	396/418	25/16	47/29	32/37
EGY	5/12	470/481	6/3	19/4
MAR	67/38	42/28	370/402	21/32
TUN	60/66	15/11	19/30	406/393

Table 5: Confusion for the linguistic method

that can be associated to the text. In the following example : <**DZTN**> كراع ***(krAç)*** </**DZTN**> <**DZTNMAEG**> كبير ***(kbyr)*** </**DZTNMAEG**> <**DZMA**> بزاف ***(bzAf)*** </**DZMA**>, the weights are assigned as follows:

- The weights for the sentence results in DZ for the Algerian represent the sum of the following values: 0.5 (only two dialects), 0.25 (4 dialects) and 0.5 which gives for the Algerian the weight : 1.25. We obtain the scores of 0.75, 0.75 and 0.25 for the Tunisian, Moroccan and Egyptian dialects respectively.
- The weight of the Algerian is greater than the one of the other dialects, so it is retained as the detected dialect.

6. Experiments

We have selected 4000 messages from the dialectal corpus (2000 in Arabic and 2000 Arabizi). These messages are not extracted from the training corpus and validated by experts. We calculated the error rate of dialect identification (LER), i.e. the proportion of messages for which the dialect is incorrectly detected.

For our dictionary-based system, on both types of writing in Latin and Arabic characters, we obtain an average error rate of 16.6%. On the Latin written corpus, this error rate is even less than 15.3%. This is due to the presence of vowels as an additional discriminating clue in Arabic writing. There are very few errors for Egyptian.

The confusion table shows also that Algerian presents difficulties: confusions are mainly due to the neighboring dialects: Tunisian and Moroccan.

The degree of ambiguity of the dictionary-based method can be explained by the choice of several dialects with whom dictionary-based method must deal before making a decision. The same observation is made: the confusion concerns Algerian, Moroccan and Tunisian. A finer method would be welcome to distinguish these dialects.

7. Discussion

This study shows that the detection of dialects is a difficult task when comparing closer Arabic dialects, as has been observed for the Algerian, Moroccan and Tunisian dialects. A more detailed analysis of the results allowed us to show some difficulties in building an accurate corpus like the presence of English or French words identical to the transliteration of dialect words. Even if these phenomena remain marginal, they should be addressed to improve the quality of the corpus and correctly evaluate the results. An important challenge addressed by our work is the processing of Arabizi. In addition to the fact that this script can be transliterated into Arabic, we also find that the addition of short vowels to the Arabic script helps to better distinguish dialects. However, the Arabizi can lead to errors; in particular due to the possible ambiguities we can obtain using Latin languages (like English or French).

Finally, we highlight the fact that the dictionary method allows to assign several dialects to the texts, which is an important advantage. Indeed, we checked that, in many cases, the content of messages did not make the dialect distinguishable, even for humans. In addition, in the case of messages may have been written, voluntarily, in several dialects. In this case, the dictionary-based method explicitly maintains this ambiguity in a way that is better controlled.

8. Conclusion and Future Work

In this paper, we have described a system for the identification and classification of the dialectal origin of texts written in Arabic or Arabizi characters. We have created lexicon and corpora for four dialects: Algerian, Tunisian, Moroccan and Egyptian. Morphological analyzers and transliterators from Arabizi to Arabic have been developed, with the aim of processing these texts with the same system used for Arabic MSA. Experiments show that a controlled approach based on dictionaries obtains good results.

To further develop and enhance this work, we first plan to extend our corpus then annotate it with a supervised and unsupervised statistical approaches. Second, we want to devise tools and resources for other Arabic dialects. Finally, we plan with ELRA to make the corpus available freely for research.

9. Acknowledgements

The authors acknowledges the support of the DGE (Ministry of Economy) and DGA (Ministry of Defense): RAPID Project “DRIRS”, referenced by number 172906108. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of DGE or DGA. We would like to thank Billel Gueni and Damien Nouvel for helpful feedback.

10. Bibliographical References

Adouane, W., Semmar, N., Johansson, R., and Bobicev, V. (2016). Automatic detection of arabicized berber and arabic varieties. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 63–72.

Al-Sabbagh, R. and Girju, R. (2012). Yadac: Yet another dialectal arabic corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2882–2889, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Benajiba, Y. and Rosso, P. (2007). Conquering the ner task for the arabic language by combining the maximum entropy with pos-tag information. In *Proceedings of Workshop on Natural Language-Independant Engineering*.

Boujelbane, R., Khemakhem, M. E., and Belguith, L. H. (2013). Mapping rules for building a tunisian dialect lexicon and generating corpora. In *IJCNLP*, pages 419–428.

Cotterell, R. and Callison-Burch, C. (2014). A multi-dialect, multi-genre corpus of informal written arabic. In *LREC*, pages 241–245.

Darwish, K. (2014). Arabizi detection and conversion to arabic. *ANLP 2014*, page 217.

Elfardy, H. and Diab, M. (2012a). Aida: Automatic identification and glossing of dialectal arabic. In *Proceedings of the 16th eamt conference (project papers)*, pages 83–83.

Elfardy, H. and Diab, M. T. (2012b). Simplified guidelines for the creation of large scale dialectal arabic annotations. In *LREC*, pages 371–378. Citeseer.

Elfardy, H. and Diab, M. T. (2013). Sentence level dialect identification in arabic. In *ACL (2)*, pages 456–461.

Elfardy, H., Al-Badrashiny, M., and Diab, M. (2014). Aida: Identifying code switching in informal arabic text. page 94. Citeseer.

Eskander, R., Al-Badrashiny, M., Habash, N., and Rambow, O. (2014). Foreign words and the automatic processing of arabic social media text written in roman script. In *Proceedings of The First Workshop on Computational Approaches to Code Switching*, pages 1—12, Doha, Qatar. Association for Computational Linguistics.

Fluhr, C., Rossi, A., Boucheseche, L., and Kerdjoudj, F. (2012). Extraction of information on activities of persons suspected of illegal activities from web open sources. page 19.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On arabic transliteration. In *Arabic computational morphology*, pages 15–22. Springer.

Habash, N., Diab, M. T., and Rambow, O. (2012). Conventional orthography for dialectal arabic. In *LREC*, pages 711—-718, Istanbul, Turkey.

Habash, N., Jarrar, M., Alrimawi, F., Akra, D. F., Zalmout, N., Bartolotti, E., and Arar, M. (2016). Palestinian arabic conventional orthography guidelines. Technical report.

Habash, N. (2010). Introduction to arabic natural language processing. In *Synthesis Lectures on Human Language Technologies*.

Ouerhani, B. (2009). Interférence entre le dialectal et le littéral en tunisie: Le cas de la morphologie verbale. *Synergies Tunisie n*, 1:75–84.

Saâdane, H. and Habash, N. (2015). A conventional orthography for algerian arabic. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 69–79, Beijing, China. Association for Computational Linguistics.

Saâdane, H., Rossi, A., Fluhr, C., and Guidere, M. (2012). Transcription of arabic names into latin. In *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, pages 857–866.

Saâdane, H., Guidere, M., and Fluhr, C. (2013). La reconnaissance automatique des dialectes arabes à l'écrit. In *colloque international Quelle place pour la langue arabe aujourd'hui*, pages 18–20.

Saâdane, H. (2011). Dialectologie arabe et transcription automatique des noms. In *In Actes des IXe Rencontres des Jeunes Chercheurs en Parole*, pages 91–93. Université de Grenoble.

Saâdane, H. (2013). Une approche linguistique pour l'extraction des connaissances dans un texte arabe. Les Sables d'Olonne, France. TALN-Récital.

Saâdane, H. (2015). Traitement automatique de l'arabe dialectalisé: aspects méthodologiques et algorithmiques. In *PhD thesis, Université Grenoble Alpes*.

Shaalan, K., Bakr, H., and Ziedan, I. (2007). Transferring egyptian colloquial dialect into modern standard arabic. In *International Conference on Recent Advances in Natural Language Processing (RANLP–2007), Borovets, Bulgaria*, pages 525–529.

Shoufan, A. and Al-Ameri, S. (2015). Natural language processing for dialectical arabic: A survey. page 36–48.

Turki, H., Adel, E., Daouda, T., and Regragui, N. (2016). A conventional orthography for maghrebi arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC), Portoroz, Slovenia*.

Versteegh, C. and Versteegh, K. (2001). The arabic language. Edinburgh University Press Series. Edinburgh University Press.

Yang, Y. and Eisenstein, J. (2013). A log-linear model for unsupervised text normalization. In *EMNLP*, pages 61–72.

Zaidan, O. F. and Callison-Burch, C. (2011). The arabic online commentary dataset: An annotated dataset of informal arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 37–41, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zribi, I., Boujelbane, R., Masmoudi, A., Ellouze, M., Belguith, L. H., and Habash, N. (2014). A conventional orthography for tunisian arabic. In *LREC*, pages 2355—-2361, Reykjavik, Iceland.

Shami: A Corpus of Levantine Arabic Dialects

Kathrein Abu Kwaik[*], [†]**Motaz Saad,** [*]**Stergios Chatzikyriakidis,** [*]**Simon Dobnik**
[*]CLASP, Department of Philosophy, Linguistics and Theory of Science, Gothenburg University, Sweden
[†]The Islamic University of Gaza, Palestine
kathrein.abu.kwaik@gu.se, motaz.saad@gmail.com, stergios.chatzikyriakidis@gu.se, simon.dobnik@gu.se

Abstract

Modern Standard Arabic (MSA) is the official language used in education and media across the Arab world both in writing and formal speech. However, in daily communication several dialects depending on the country, region as well as other social factors, are used. With the emergence of social media, the dialectal amount of data on the Internet have increased and the NLP tools that support MSA are not well-suited to process this data due to the difference between the dialects and MSA. In this paper, we construct the Shami corpus, the first Levantine Dialect Corpus (SDC) covering data from the four dialects spoken in Palestine, Jordan, Lebanon and Syria. We also describe rules for pre-processing without affecting the meaning so that it is processable by NLP tools. We choose Dialect Identification as the task to evaluate SDC and compare it with two other corpora. In this respect, experiments are conducted using different parameters based on n-gram models and Naive Bayes classifiers. SDC is larger than the existing corpora in terms of size, words and vocabularies. In addition, we use the performance on the Language Identification task to exemplify the similarities and differences in the individual dialects.

Keywords: Dialectal Arabic, Levantine Dialect Corpus, Dialect Identification.

1. Introduction

Arabic is one of the five most spoken languages in the world; it is spoken by more than 422 million native speakers and used by more than 1.5 billion Muslims.[1] The situation in Arabic is a classic case of diglossia, in which the written formal language differs substantially from the spoken vernacular. Modern standard Arabic (MSA), which is heavily based on Classical Arabic, is the official written language used in government affairs, news, broadcast media, books and education. MSA is the lingua franca amongst Arabic native speakers. However, the spoken language (collectively referred to as Dialectal Arabic) varies widely across the Arab world.

The rapid proliferation of social media resulted in these dialects finding their way to written online social interactions. Dialects of Arabic differ widely among each other and depend on the geographic location and the socioeconomic conditions of the speakers. They are commonly categorized in five dominant groups: Maghreb (Libya, Tunisia, Algeria, Morocco and western Sahara), Egyptian (Egypt and some parts of Sudan), Levantine (Palestine, Syria, Lebanon and Jordan), Iraqi (Iraq) and Gulf (Qatar, Kuwait, Bahrain, United Arab Emirates, Saudi Arabia, Oman, Yemen) (Zbib et al., 2012). Further varieties can be identified, for example the Jordanian dialect can be split to Urban, Rural and Bedouin.

MSA and Dialectal Arabic share a considerable number of lexical, semantic, syntactic and morphological features, but there are several differences as well. For example, the word (ايش *āyš*) means "what" in Palestinian but in MSA (اي + شيء *āy + šyʾ*) means "what thing". The word (زرابي *zrāby*) in Moroccan means "carpets" and is a synonym of (سجاد *sǧād*) or (بساط *bsāṭ*) in MSA. On the other hand, the word (وقية *wqyh*) in Algerian means "3 kg" while it means "1/4 kg" in MSA and Levantine, i.e. the same word has different meanings in Algerian, Levantine, and MSA.

Most of the Natural language processing (NLP) resources for Arabic concern Modern Standard Arabic (MSA) (Diab, 2009; Maamouri and Cieri, 2002; Manning et al., 2014; Habash et al., 2009; Habash, 2010). However, using these resources for Dialectal Arabic (DA) is considered a very challenging task given the differences between them.

It should be noted that, even though the Levantine dialects look similar to each other when only written form rather than spoken form is taken into consideration, there are a number of differences. Some of these are summarized in Table 1. There are additional reasons why discriminating Levantine dialects in text is challenging. Some of these include:

- Lack of accent in writing. For example, a word like (كيفك *kyfk* / *how are you*) is used in all dialects, but its pronunciation varies by country.
- High similarity between the Palestinian and Jordanian dialects, **except in some key words.**
- The political situation. For example, we find many Palestinians leveling their dialect closer to Syrian and Lebanese, which makes Palestinian appear as an intermediate dialect.
- Prestige. For example, many Bedouins or Rural people adapt their dialect to Urban to clarify the conversation.
- Lack of linguistic description and resources. For example, Levantine dialects are frequently treated as one dialect despite their differences.

In this paper, we focus on Levantine dialects which are spoken in Palestine, Syria, Jordan and Lebanon and collect a corpus of their usage from social media which we label as *Shami*. This is the first joint corpus of these four dialects. The name *Shami* derives from شامي *šāmy*, which in MSA means *Levantine*. We evaluate the corpus through a Language Identification task.

[1] http://www.unesco.org/new/en/unesco/events/prizes-and-celebrations/celebrations/international-days/world-arabic-language-day-2013/

الآنَ أَفْضَلُ شَيءٍ أَنَ الشَبابُ كُلهمْ *ālʿāna ʾafḍalu šayʾ ʾana ālšabaābu kulahum* ظَهَرُوا عَلَى السَاْحَةِ *ẓaharuwā ʿalaā ālsaāḥati*	MSA
Now the best thing is that guys have appeared on the scene	English
هَلَأْ أَحْلَى شِي إِنْوْ الشَبابَ كِلْلون ضَهَرُوا عسّاحِه *halaʿa ʾaḥlaā šiy ʾinw ālšabaāba killuwn ḍaharuwā ʿsāḥih*	Lebanon
هسا أَحْلَى اِشِي إِنْوْ الشَّبَابْ هَلَّأْ *hsā ʾaḥlaā āišiy ʾinw ālšabaāb halaʿa* كُلُّهُمْ ظَهَرُوا عَسَّاحَة *kuluhum ẓaharuwā ʿasaāḥah*	Jordanian
هَلْجِيت أَحْلَى اِشِي إِنْوْ الشَّبَابْ *halǧiyt ʾaḥlaā āišiy ʾinw ālšabaāb* كُلُّهُمْ طِلْعُوا عَسَّاحَة *kulahum ṭilʿuwā ʿasaāḥah*	Palestinian
هَلَّأْ أَحْلَى شِي إِنْوْ الشَّبَابْ كِلْلُون بَيَّنُوا عسّاحِه *halaʿa ʾaḥlaā šiy ʾinw ālšabaāb killuwn bayanuwā ʿsāḥih*	Syrian

Table 1: little differences between the parallel sentence among Levantine Dialects

2. Related Works

There are two kinds of Arabic dialectal resources: (i) resources that group all Levantine dialects in the same group; (ii) resources where individual Levantine dialects are represented but none of them contains all of the dialects that we are interested in.

The Arabic Online Commentary (AOC) dataset presents a monolingual dataset rich with dialectal content from three dialects (Gulf, Egyptian, Levantine) (Zaidan and Callison-Burch, 2011). In the case of Levantine, data are extracted from Jordanian newspapers only. Zbib et al. (2012) build Parallel Levantine-English and Egyptian-English parallel corpora, consisting of 1.1M words and 380k words, respectively.

Meftouh et al. (2015) present PADIC (Parallel Arabic Dialect Corpus). It includes five dialects: two Algerian, one Tunisia and two Levantine (Palestinian and Syrian). Bouamor et al. (2014) present a multi-dialectal Arabic parallel corpus. The dataset consists of 2,000 MSA sentences in Egyptian, Syrian, Palestinian, Tunisian, Jordanian and English. Table 2 in section 3 illustrate the size of these corpora.

A preliminary work on a Palestinian dialect corpus is presented by Jarrar et al. (2014). It includes 5,836 Palestinian sentences with 43K words.

Diab et al. (2010) designed a set of search queries for each dialect (Egyptian, Iraqi, Maghrebi and Levantine) to harvest automatically dialectal content from large online resources like weblogs and forums. Levantine dialects (Palestinian, Syrian, Jordanian, Lebanese) were assumed to comprise a single Arabic dialect. The data cover three domains only: social issues, politics and religion.

A similar corpus to AOC is presented by Cotterell and Callison-Burch (2014). However, again, most of the Levantine data are from one Levantine dialect only, namely Jordanian (6k sentences).

Almeman and Lee (2013) build a multi-dialect text corpus by bootstrapping dialectal words. They then categorize the dialectal text into four main categories depending on geographical distribution (Gulf, Levantine, Egyptian and North Africa), giving 14.5M, 10.4M, 13M and 10.1M tokens respectively.

3. Shami Dialect Corpus (SDC)

In this section, we present our *Shami* Dialect Corpus (SDC) which contains only Levantine dialects.

Its most important characteristics are: a) it is the first Levantine dialect corpus that contains the largest volume of data separated as individual Levantine dialects compared to the previous corpora; b) it is not a crafted and also not a parallel corpus; it contains real conversations as written in social media and blogs; c) it is not confined to a specific domain; it includes several topics from regular conversations such as politics, education, society, health care, house keeping and others; d) unlike previous corpora, SDC has been created from scratch by collecting Levantine data through automatic and manual approaches.

SDC is organized in text files, where each file represents one dialect. We have another structure for the Language Identification task, where we have a sub folder for each dialect and each sentence is represented as a separate text file. The corpus and the associated code can be found here: `https://github.com/GU-CLASP/shami-corpus`.

3.1. Data Collection

3.1.1. Automatic Collection

We use the Twitter API streaming library (*Tweepy*)[2] to collect as many relevant tweets as possible. Firstly, we gather twitter IDs from public Levantine figures and hence their linguistic background is known. Secondly, we use *tweepy* to collect tweets and replies from these IDs. Each streaming was run until 9,999 tweets are reached each time. We further use *tweepy* to extract data according to geographical location.

3.1.2. Manual Collection

Given that various domains and topics are needed for our corpus, we also collected a part of SDC manually. We harvest the web and choose online dialectal blogs for public figures in Levantine countries. We also extract discussions and stories from forums. Overall, this gives us sentences of various lengths.

[2] http://www.tweepy.org/

3.2. Data Pre-processing

Special treatment is required in order to pre-process dialectal text in order to standardize it and to make it useful for NLP applications in order to avoid a large number of single instance tokens[3]. For this reason, We employ the following processing steps:

- Remove diacritics: Arabic text has several diacritics which mark the pronunciation of the words and sometimes the meaning. We remove these diacritics from the corpus, for example: ّ Tashdid, َ *a* Fatha, ً *an* Tanwin Fath.
- Automatic data collection extracts many words and letters that do not belong to Arabic dialects. For example, Lebanese texts contain a lot of French and Arabic text in Latin characters as well as special characters like (@, !!, ??), number and dates, emoticons, and symbols.
- Normalization: there is no standard orthography for Arabic dialects. We try to unify the writing style by normalizing the spelling. Unlike previous work that applies across the board normalization, which sometimes unintentionally changes the meaning of the words, we implement finer rules that work more reliably and preserve the semantic meaning of the text, for example:
  a) Aleph: we only convert Aleph with an accent أ *ʔa* to Aleph without an accent ا *ā* if it appears at the beginning of the word. This is because we want to mark the accent in other contexts in order to preserve the meaning of dialectal words. For example, this allows us to distinguish (’هلأ *hlʔa*’ / now) from (’هلا *hlā*’/ Hello), which otherwise would be indistinguishable in meaning.
  b) Alef Maqsora (ى *ā*) at the end of the word: in most processing steps the letter (ى *ā*) is converted to a (ي *y*), but we did not do so because a lot of words would change the meaning if we unified the characters. For example: (على *ʕlā* / on preposition) and (علي *ʕly* Ali / a personal name). If we change the letter (ى *ā*), this will affect the context of the sentence.
- Remove repeated characters: In colloquial written speech as well as in social networks, some letters are frequently repeated to indicate length (for example Waaaaaaw). In previous works, all duplicate letters are removed to one or two letters. Again, we have specified some finer criteria to specify the repetition based on its origin. Below is our steps to remove repeated letters:
  1. We extract all words containing repeated characters in MSA texts and keep them in a list.
  2. All words containing duplicate characters from the previous list are abbreviated to two characters.
  3. The rest of the characters are reduced to only one character, for example the repeating character و *w* in (مبروووووك *mbrwwwwwk* / congratulation) is converted to (مبروك *mbrwk*/ congratulation). These are all alternative spellings imitating spoken language.
  4. The conjunction letter (و *w* / and) is a special case in Arabic. Some people in colloquial dialects connect it with the next word without a space. We postulated that if the given word begins with more than one (و *w*), the first (و *w*) and the rest of the word are separated and the original word is processed according to the previous algorithm. Figure 1 describes the algorithm for reducing repeated characters.
- Finally, we homogenize individual parts of the corpus to make sure they are free from MSA sentences as well as other non-Levantine dialects. This is done manually by native speakers of individual dialects.

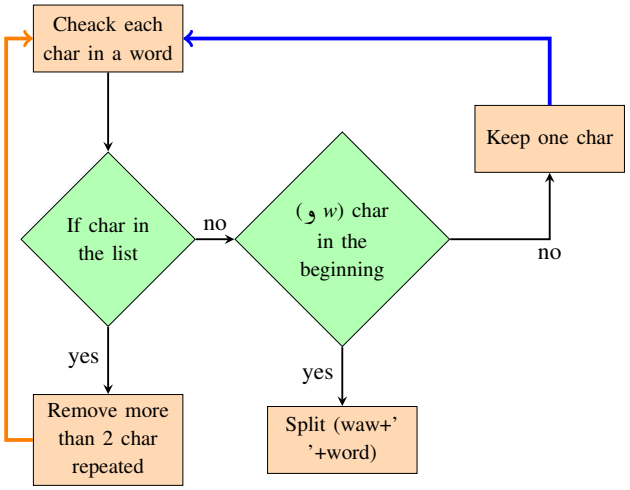


Figure 1: Algorithm for repeated characters in dialectal words

Compared with the PADIC (Meftouh et al., 2015) and the Multi-dialect Corpus (Bouamor et al., 2014), SDC is larger, more diverse, and more comprehensive. Table 2 reports the sentences, words, and vocabulary counts for SDC, PADIC and the Multi-dialect Corpus corpora.
In the next section, we use Arabic Dialect Identification as a test task for SDC in order to validate the corpus and test its usefulness. We then compare the results with PADIC and the Multi-dialect Corpus.

4. Arabic Dialect Identification

Arabic Dialect Identification can be performed at two levels: a) a coarse-grained level that builds a learner that given a specific Arabic sentence, measures the percentage of its dialectal content; b) a fine-grained level that classifies a sentence to the dialect in which it belongs.

[3]The original version of the corpus before pre-processing is also available if needed for other tasks

Shami Corpus			
	sentences	tokens	types
Jordanian	32078	3684369	85383
Palestinian	21264	2789103	69378
Syrian	48159	5268065	77918
Lebanese	16304	1409952	44418
Total	117805	13151489	227097
PADIC			
	sentences	tokens	types
Palestinian	6418	50827	22896
Syrian	6418	48701	27032
Total	12836	99528	49820
Multi-dialect Corpus			
	sentences	tokens	types
Jordanian	1000	9866	8905
Palestinian	1000	10315	8874
Syrian	1000	11586	9145
Total	3000	31767	26924

Table 2: Statistics for SDC, PADIC, and multi-dialects corpora

Zaidan and Callison-Burch (2014) train and evaluate an automatic classifier for Dialect Identification task on Maghrebi, Egyptian, Levantine, Iraqi and Gulf based on the corpus from Zaidan and Callison-Burch (2011) and they achieve an accuracy of 85.7% using a word-gram model. They conclude that using an n-gram word and character model is the most suitable method to distinguish among these dialects.

Elfardy and Diab (2013) propose sentence level identification and use words as tokens. They present a supervised classification method using Naive Bayes classifier to classify between MSA and Egyptian. They work on two parallel corpora. The first one is an Egyptian - Levantine - English corpus of 5M tokenized words of Egyptian (3.5M) and Levantine (1.5M). The second one is an MSA-English corpus with 57M tokenized words obtained from several LDC corpora. The system achieves different accuracy depending on the preprocessing steps and extracted features like percentage of dialect content, perplexity and metadata. The highest accuracy is 85% on the Arabic Online Commentary dataset AOC. In (Salloum et al., 2014), this work is extended to include the Iraqi, Levantine and Moroccan dialects.

Sadat et al. (2014) experiment with character n-grams using Markov Model and Naive-Bayes classifiers. The experiments are conducted on 6 main dialects defined by geographical area and 98% accuracy is achieved. However, the Levantine data constitutes the smallest dialect set and receives low overall accuracy.

Adouane et al. (2016) focus on Berber and various Arabic dialects (Algerian, Egyptian, Levantine, Gulf, Mesopotamian (Iraqi), Moroccan, Tunisian). They show that machine leaning (ML) models combined with lexicons are well suited for Dialect Identification as they achieve 93% accuracy when employed on 9 dialects whereby all Levantine dialects are grouped together.

A common characteristic of the previous Arabic Language Identification systems is that: (i) the data on which they are trained mostly comes from the same domain; (ii) most datasets and corpora are small in size; (iii) systems are trained and tested on different datasets with different parameters (size, domain, preprocessing).

Language Identification is a well-known task and given a sufficient amount of resources it can be considered a solved task. However, this does not hold for Arabic Dialect Identification. A lot of tools for Language Identification are available as open source, for example : *langid.py* (Lui and Baldwin, 2012) and *langdetect* (Shuyo, 2010). Given that the dialects that we are focusing on are very similar, our experiments may also give new insights with respect to Language Identification.

4.1. Langid.py for language identification

Lui and Baldwin (2012) present a tool for Language Identification called *langid.py*. In their tool they use Naive Bayes classifier with various n-gram character sequences for training purposes. The tool has been trained to identify 97 languages in the multi-domain Language Identification corpus of (Lui and Baldwin, 2011). The tool supports many modules so developers can easily train and build their own language models.

Comparing *Langid.py* to other Language Identification tools like *langdetect*, *TextCat* (Cavnar et al., 1994), and *CDL* (McCandless, 2011), Lui and Baldwin (2012) found that it is faster and gives better accuracy. This is why we also use *Langid.py* to conduct our corpus evaluation.

4.2. Scikit learn machine learning toolkit

Scikit learn (Pedregosa et al., 2011) is an open source python library that is a simple and efficient tool for data mining, data analysis and machine learning. It implements many machine learning methods like classification, regression and clustering and includes modules for preprocessing and feature selection. We use it to build Language Identification classifiers with word-gram models as *langid.py* does not support this.

5. Evaluation using Language Identification

Two techniques are commonly used in the literature for Language Identification. The first one is based on compiling lists of keywords for each language and scoring the text based on these lists (Richardson and Campbell, 2008). The second one uses Machine Learning such as Artificial Neural Networks (Al-Dubaee et al., 2010; Gonzalez-Dominguez et al., 2014), Support Vector Machines (Botha and others, 2008), Hidden Markov Models (Dunning, 1994) and N-gram models (Yang and Liang, 2010; Selamat, 2011) to identify languages.

We train two Dialect Identification systems a) a character based n-gram model with Naive Bayes classifier (*langid.py*); b) word based n-gram model with Naive Bayes classifier (*scikit learn*). We use n-gram based approaches because most of the variation between dialects can be identified by considering sequences of characters, e.g. affixation, and words, in addition to word features which can be provided by the lexicon. We conduct several experiments, which vary w.r.t the size of the data, the libraries

used (*langid.py* (Lui and Baldwin, 2012), *scikit learn* (Pedregosa et al., 2011)) and the classification techniques. For evaluation purposes, we measure the accuracy of the correctly identified test instances, while the F-measure gives us the balance between Precision and Recall.

5.1. Baseline system

To properly evaluate SDC's suitability as a corpus, we compare it to two Dialectal Corpora, i.e. PADIC (Meftouh et al., 2015) and Multi-dialect Corpus (Bouamor et al., 2014) using Language Identification. Firstly, we split the data and use 90% for training data and 10% for testing data. The first experiment was run with *Langid.py* with 4,5,6 and 7 n-character grams language model. Then, we evaluate these models using the test set. The second experiment was run with *scikit-learn* using uni-gram and bi-gram word models. The results for the two corpora are shown in Table 3.

PADIC			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.61	0.75
	5-gram char	0.64	0.78
	6-gram char	0.68	0.81
	7-gram char	0.68	0.81
Scikit-learn	uni-gram word	0.83	0.83
	bi-gram word	0.84	0.83
Multi-dialects Corpus			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.63	0.77
	5-gram char	0.68	0.81
	6-gram char	0.70	0.83
	7-gram char	0.69	0.82
Scikit-learn	uni-gram word	0.69	0.68
	bi-gram word	0.69	0.69

Table 3: Evaluation on PADIC and Multi-dialects Corpora

In general, the table shows that 6-gram models work best for Language Identification on the two corpora; it appears they are picking out particular phrases. In PADIC, the *scikit-learn* library with word-gram model outperforms *langid.py*. This is because PADIC is a parallel corpus of translated sentences where the differences are specifically emphasized when the corpus was built, and therefore more differences can be observed between Palestinian and Syrian. The Multi-dialect Corpus includes three Levantine dialects (Jordanian, Palestinian and Syrian), which makes the distinction between words harder, especially between Palestinian and Jordanian, which are very similar.

5.2. Dialect Identification with SDC

We carried out an experiment to determine the data size that will give us the highest performance. When we used all the data (Table 2), we get a low Accuracy ranging from 36% to 39% in all n-grams model as shown in Table 4.

As SDC is neither a parallel corpus nor a crafted corpus, many sentences are difficult to classify to a particular language. We commissioned four native speakers, one per each Levantine dialect, to create a subset of the data in order to reduce its heterogeneousness using the native knowledge of these dialect as shown in Table 5. Table 6 shows the results from training and testing on the filtered data from SDC. In comparison to raw data as shown in Table 4, the reduction of heterogeneousness improved the performance, thus, the classes of documents become more homogeneous in terms of the dialect.

	Language model	Accuracy	F-measure
langid.py	4-gram char	0.36	0.52
	5-gram char	0.38	0.53
	6-gram char	0.38	0.55
	7-gram char	0.39	0.55

Table 4: Evaluation on Sampling from SDC

Dialect	Train	Test	Total
Palestinian	9577	1065	10642
Syrian	33983	3776	37759
Jordanian	6316	702	7018
Lebanese	9747	1083	10830

Table 5: Train and test set for SDC after filtering

	Language model	Accuracy	F-measure
langid.py	4-gram char	0.54	0.70
	5-gram char	0.65	0.71
	6-gram char	0.55	0.71
	7-gram char	0.55	0.71
Scikit-learn	uni-gram word	0.70	0.71
	bi-gram word	0.70	0.70

Table 6: Evaluation on Sampling from SDC

However, the systems do not perform as well on SDC as on PADIC and Multi-dialect Corpus. This shows that Dialects Identification is more difficult in the case of SDC than the other two corpora. This is presumably because SDC is a more natural corpus than the other two which were translated with a focus on differences.

To confirm this, we have done a survey of several sentences without typical dialectal keywords and asked Levantine native speakers to classify each sentence to the dialect it belongs. For example, one of the sentences was الدرس اليوم كان كتير حلو وما حسينا بملل بالمرة. *āldrs ālywm kān ktyr ḥlw wmā ḥsynā bmll bālmrt.* ياريت كل يوم يكون هيك *yāryt kl ywm ykwn hyk* (The class today was very nice and interesting and we never felt bored. Hopefully every day will be like that). None of the participants could classify the Levantine dialect of this sentence with full certainty.

5.2.1. Comparing groups of dialects

Due to the great similarity between the Levantine dialects, we have conducted further experiments to compare subsets of dialects in the classification. We first used two-way classification between Palestinian and Syrian (as in the PADIC corpus) and between Jordanian and Lebanese (Table 7), and a three-way classification between Palestinian, Jordanian

and Syrian as in the Multi-dialect Corpus. Due to the similarity of Palestinian to the other dialects, we excluded it in the final classification, which included Jordanian, Syrian and Lebanese only (Table 8). Figures 2, 3 show the F-measure for the classification task for two and three dialects comparing with the PADIC and the Multi-dialect corpus.

Palestinian - Syrian Classification			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.73	0.83
	5-gram char	0.72	0.83
	6-gram char	0.72	0.84
	7-gram char	0.72	0.83
Scikit-learn	uni-gram word	0.87	0.85
	bi-gram word	0.80	0.74
Jordanian - Lebanese Classification			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.87	0.88
	5-gram char	0.89	0.89
	6-gram char	0.89	0.89
	7-gram char	0.89	0.89
Scikit-learn	uni-gram word	0.90	0.90
	bi-gram word	0.88	0.88

Table 7: Evaluation on two dialects classification

Palestinian - Jordanian - Syrian Classification			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.65	0.78
	5-gram char	0.65	0.79
	6-gram char	0.65	0.78
	7-gram char	0.64	0.78
Scikit-learn	uni-gram word	0.77	0.71
	bi-gram word	0.70	0.60
Jordanian - Syrian - Lebanese Classification			
	Language model	Accuracy	F-measure
langid.py	4-gram char	0.64	0.78
	5-gram char	0.81	0.82
	6-gram char	0.66	0.79
	7-gram char	0.65	0.79
Scikit-learn	uni-gram word	0.75	0.70
	bi-gram word	0.70	0.60

Table 8: Evaluation on three dialects classification

We get the highest performance when we classify Jordanian and Lebanese as they are to some extent different, and thus can be distinguished by text. Overall, this suggests that the Levantine dialects in SDC are very similar and therefore difficult to differentiate.

This is further emphasized by the fact that SDC is not a parallel corpus where textual differences between the sentences are specifically introduced as in the vase of PADIC and Multi-dialect Corpus. This way, the SDC highlights both similarities and differences between the four dialects. As it contains real data from social media and other sources as used by native speakers in everyday domains, it is a valuable resource for building NLP systems dealing with such data.

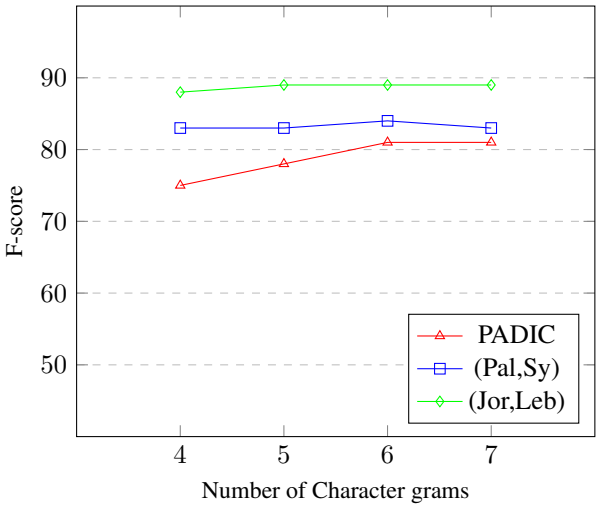


Figure 2: F-score on two dialect classification

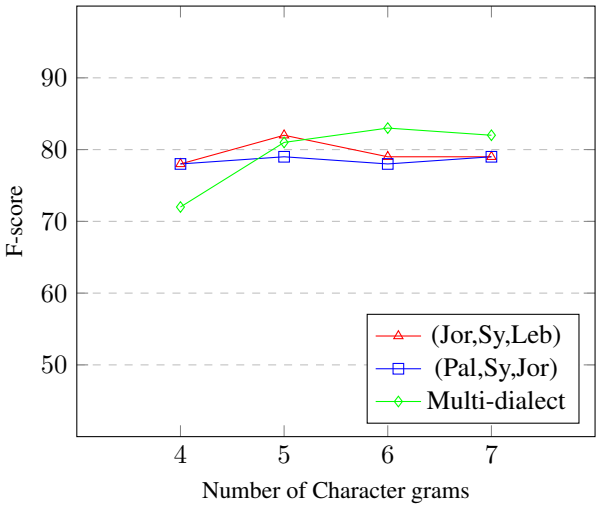


Figure 3: F-score on 3 dialect classification

6. Conclusions and Future Work

In this paper we present SDC, i.e. the first Levantine Dialect corpus which contains Palestinian, Jordanian, Syrian and Lebanese data. SDC contains natural data from new domains not available in the previous corpora. We have adopted a combination of manual and automatic methods to collect the documents. We then performed some preprocessing to standardize the spelling. Finally, we filtered the corpus to make it more homogeneous in terms of the dialects. We tested the usefulness of language models for Dialect Identification task by applying various n-gram models in two different classification approaches. At the same time, we compared the performance of Dialect Identification on SDC with that of PADIC and the Multi-Dialect Corpus corpora. The best results were achieved for classifying Jordanian and Lebanese only using a uni-gram word model (90% accuracy). This result is not surprising given a little similarity between the two dialects on the lexical level. The worst results were obtained when classifying for all four dialects on the whole SDC (52% accuracy). This is due to the great overlap between the dialects and dispersion of lexical items between categories.

We found that Language Identification on SDC outperforms other corpora when exactly the same dialects are considered. For example, Table 7 shows better performance than Table 3 for PADIC, and Table 8 shows better performance than Table 3 for Multi-Dialect corpus.

Our future works consist in extending the coverage of SDC and performing a corpus analysis of its linguistic features which will give us a more consistence picture of the differences between these dialects. The corpus will also be used to create lists of dialectal keywords. Finally, techniques other than Language Identification will also be tested on SDC.

7. References

Adouane, W., Semmar, N., Johansson, R., and Bobicev, V. (2016). Automatic Detection of Arabicized Berber and Arabic Varieties. *VarDial 3*, page 63.

Al-Dubaee, S. A., Ahmad, N., Martinovic, J., and Snasel, V. (2010). Language Identification Using Wavelet Transform and Artificial Neural Network. In *Computational Aspects of Social Networks (CASoN), 2010 International Conference on*, pages 515–520. IEEE.

Almeman, K. and Lee, M. (2013). Automatic Building of Arabic Multi Dialect Text Ccorpora by Bootstrapping Dialect Words. In *Communications, Signal processing, and their Applications (ICCSPA), 2013 1st international conference on*, pages 1–6. IEEE.

Botha, G. R. et al. (2008). *Text-based Language Identification for the South African Languages*. Ph.D. thesis, University of Pretoria.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A Multidialectal Parallel Corpus of Arabic. In *LREC*, pages 1240–1245.

Cavnar, W. B., Trenkle, J. M., et al. (1994). N-gram-based Text Categorization. *Ann Arbor MI*, 48113(2):161–175.

Cotterell, R. and Callison-Burch, C. (2014). A Multi-dialect, Multi-Genre Corpus of Informal Written Arabic. In *LREC*, pages 241–245.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). Colaba: Arabic Dialect Annotation and Processing. In *Lrec workshop on semitic language processing*, pages 66–74.

Diab, M. (2009). Second Generation AMIRA Tools for Arabic Processing: Fast and Robust Tokenization, POS Tagging, and Base Phrase Chunking. In *2nd International Conference on Arabic Language Resources and Tools*, volume 110.

Dunning, T. (1994). *Statistical identification of Language*. Computing Research Laboratory, New Mexico State University.

Elfardy, H. and Diab, M. T. (2013). Sentence Level Dialect Identification in Arabic. In *ACL (2)*, pages 456–461.

Gonzalez-Dominguez, J., Lopez-Moreno, I., Sak, H., Gonzalez-Rodriguez, J., and Moreno, P. J. (2014). Automatic Language Identification Using Long Short-term Memory Recurrent Neural Networks. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Habash, N., Rambow, O., and Roth, R. (2009). MADA+ TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization. In *Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt*, volume 41, page 62.

Habash, N. Y. (2010). Introduction to Arabic Natural Language Processing. *Synthesis Lectures on Human Language Technologies*, 3(1):1–187.

Jarrar, M., Habash, N., Akra, D., and Zalmout, N. (2014). Building a Corpus for Palestinian Arabic: A Preliminary Study. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 18–27.

Lui, M. and Baldwin, T. (2011). Cross-domain Feature Selection for Language Identification. In *In Proceedings of 5th International Joint Conference on Natural Language Processing*. Citeseer.

Lui, M. and Baldwin, T. (2012). Langid.py: An Off-the-shelf Language Identification Tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30. Association for Computational Linguistics.

Maamouri, M. and Cieri, C. (2002). Resources for Arabic Natural Language Processing. In *International Symposium on Processing Arabic*, volume 1.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford Corenlp Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

McCandless, M. (2011). Accuracy and Performance of Google's Compact Language Detector.

Meftouh, K., Harrat, S., Jamoussi, S., Abbas, M., and Smaili, K. (2015). Machine Translation Experiments on PADIC: A Parallel Arabic Dialect Corpus. In *The 29th Pacific Asia conference on language, information and computation*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Richardson, F. S. and Campbell, W. M. (2008). Language Recognition With Discriminative Keyword Selection. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 4145–4148. IEEE.

Sadat, F., Kazemi, F., and Farzindar, A. (2014). Automatic Identification of Arabic Dialects in Social Media. In *Proceedings of the First International Workshop on Social Media Retrieval and Analysis*, pages 35–40. ACM.

Salloum, W., Elfardy, H., Alamir-Salloum, L., Habash, N., and Diab, M. (2014). Sentence Level Dialect Identification for Machine Translation System Selection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 772–778.

Selamat, A. (2011). Improved N-grams Approach for Web Page Language Identification. In *Transactions on*

Computational Collective Intelligence V, pages 1–26. Springer.

Shuyo, N. (2010). Language Detection Library for Java. *Retrieved Jul*, 7:2016.

Yang, X. and Liang, W. (2010). An N-gram-and-Wikipedia Joint Approach to Natural Language Identification. In *Universal Communication Symposium (IUCS), 2010 4th International*, pages 332–339. IEEE.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic Online Commentary Dataset: An Annotated Dataset of Informal Arabic with High Dialectal Content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41. Association for Computational Linguistics.

Zaidan, O. F. and Callison-Burch, C. (2014). Arabic Dialect Identification. *Computational Linguistics*, 40(1):171–202.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O. F., and Callison-Burch, C. (2012). Machine Translation of Arabic Dialects. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 49–59. Association for Computational Linguistics.

You Tweet What You Speak: A City-Level Dataset of Arabic Dialects

Muhammad Abdul-Mageed, Hassan Alhuzali, Mohamed Elaraby
Natural Language Processing Lab
University of British Columbia
muhammad.mageeed@ubc.ca, {halhuzali,mohamed.elaraby}@alumni.ubc.ca

Abstract

Arabic has a wide range of *varieties* or *dialects*. Although a number of pioneering works have targeted some Arabic dialects, other dialects remain largely without investigation. A serious bottleneck for studying these dialects is lack of any data that can be exploited in computational models. In this work, we aim to bridge this gap: We present a considerably large dataset of $> 1/4$ billion tweets representing a wide range of dialects. Our dataset is more nuanced than previously reported work in that it is labeled at the fine-grained level of city. More specifically, the data represent 29 major Arab cities from 10 Arab countries with varying dialects (e.g., Egyptian, Gulf, KSA, Levantine, Yemeni).

1. Introduction

The Arab world covers a vast region across the two contents, Africa and Asia. The term *Arabic* itself refers to a collection of varieties, possibly comprised by three major categories: (1) Modern Standard Arabic (MSA), (2) Classical Arabic (CA), and (3) Dialectal Arabic (DA). MSA (Badawi, 1973) is the modern variety of the language used in educational settings and some pan-Arab networks like AlJazeera (Abdul-Mageed, 2008; Abdul-Mageed and Herring, 2008). CA is the language of the Qura'an (the Holy Book of Islam) that is employed in religious and elite literary works. MSA and CA differ mainly lexically and morphologically, with fewer structural and syntactic differences (Bateson, 1967; Ryding, 2005). DA is a collection of arbitrarily defined (Versteegh, 2001; Habash, 2010) variations, although geography does play a role in the classification of DA.

Most Arabic varieties remained primarily spoken for a long time. With the advent of the internet and the proliferation of social media, Arabic dialects found their way to online written form (Abdul-Mageed, 2015). Early computational studies of Arabic dialects have depended on data collected from blogs and comments on online news sites, e.g., (Diab et al., 2010; Elfardy and Diab, 2012). Due to the costly efforts associated with labeling the data with dialect tags, these pioneering works have focused on a few varieties like Egyptian or Levantine (Elfardy and Diab, 2013; Zaidan and Callison-Burch, 2011; Zaidan and Callison-Burch, 2014). User-provided location information in the Twitter microblogging platform have made it possible to collect data with nuanced geographical labels in ways not previously possible, see e.g., (Jurgens et al., 2017). We depend on these cues to label our dataset with location tags as a proxy for the relevant dialects.

Although no agreement exists as to where dialectal boundaries should be drawn, some proposals have been made. Figure 1 shows only one such classifications of Arabic dialects. The vast geographic extension the Arab world constitutes naturally translates into rich and varied linguistic tradition, thus making nuanced study of Arabic dialects an attractive object of scientific investigation. In this work, we take a step in this direction by collecting a large dataset of Arabic dialects. We focus on the Eastern part of the region,

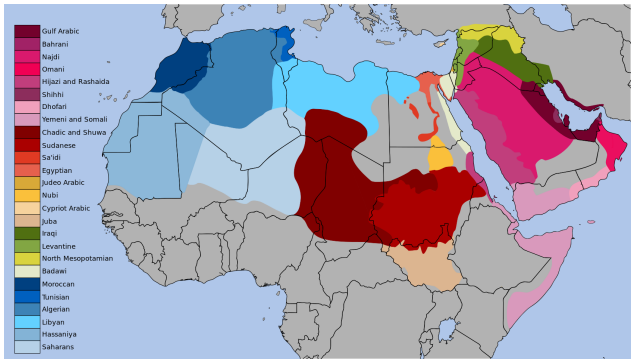


Figure 1: A classification of Arabic varieties. [1]

covering 10 different countries. These includes countries for which no (large) datasets are available (e.g., KSA, Jordan, Qatar, Yemen, and UAE).

Overall, we offer the following contributions: **(1)** We build a large-scale dataset for a variety of Arabic dialects with automatically labeled city- and country-level tags, as a proxy for respective dialects, **(2)** we manually verify the dialect labels on a pilot section of our data, **(3)** to show the utility of our data, we build a word-level embeddings model exploiting the data. We illustrate the capacity of this distributed representation model of words by both comparing its coverage to a standard publicly available model built on MSA and providing examples of word relationships it is able to capture.

The rest of the paper is organized as follows: Section 2. reviews related work. In Section 3., we describe our dataset. Section 4. is a description of our pilot dialect annotation study. In Section 5., we describe a distributed representation of words model we built exploiting our data. Section 7. is where we conclude and discuss future directions.

2. Related Works

2.1. Arabic Dialects.

Arabic dialects differ in various ways from MSA. These include phonological, morphological, lexical, and syntactic differences (Bassiouney, 2009; Holes, 2004; Palva, 2006). Although, in theory, Arabic dialects can be classified in various ways (Palva, 2006), categorizations of Arabic dialects

remain arbitrary and primarily based on geographical divisions, e.g., (Habash, 2010; Versteegh, 2014). Habash (2010) provides the following classification of Arabic dialects, indicating it is only one of many categorizations:

- **Egyptian (EGY):** Includes dialects of Egypt and the Sudan (Nile valley);
- **Gulf (GLF):** Covers dialects of Bahrain, Kuwait, Oman, Qatar, Kingdom of Saudi Arabia, and United Arab Emirates;
- **Iraqi (IRQ):** Includes elements from both Levantine and Gulf;
- **Levantine (LEV):** Includes dialects of Lebanon, Syria, Jordan, and Palestine;
- **Maltese (MLT):** Is a variant that is not always considered an Arabic dialect, but rather a separate language, and is written in the Roman script;
- **North African (MAG):** Encompasses dialects of Algeria, Libya, Mauritania, Morocco, and Tunisia;
- **Yemenite (YEM):** often considered its own class (i.e., does not include other varieties and hence stands as a category by itself).

The classification above is perhaps the most common in the literature[2]. Differences between dialects within the categories above are most pronounced. The following tweet illustrates a morphological characteristic of the the Iraqi dialect, for example. In specific, it is focused at the practice of using the final "ج" for "second person singular" in the word "طبيعتج" in Iraqi Arabic:

- (1) سخافة اتصرفي ع طبيعتج دادة، حتى لو تصرفات عبوسي وبعفويتج احلى مـن تصرفات مصطنعة وكلها فوطوشوب، نصيحتي كيوت لاتكلمني البنت لماتدلع

  Eng. "Enough with fake behaviour, girl! Be yourself, even if you think it's not attractive, being natural is way better than the "I'm cute, artificial behaviour. My advice is to ignore girls that are artificial."

This contrasts with use of "ك" for "second person singular" in the word "طبيعتك" in Gulf (example from Palestinian) below:

- (2) بصي ياحبي العيب منك و منهم بمعني انك خليكي ع طبيعتك و طيبة قلبك بس بلاش الثقه ف اي حد او الحب ال بنوزعه عمال ع بطال ع الناس لان اقسم بالله بنتشنير شنيره بنت لذين بسبب طيبتنا دي..خليكي فاللون الرمادي حبي واكتمي ولو حسيتي الحب متبادل يبقا اتكلي ع الله

  Eng. "Listen, [female] dear, the problem lies both in you and them. In other words, be yourself, as kind as you are, but don't put extra trust in anyone. So forget about the overflowing love we distribute right an left for this naive kindness ends up causing us to suffer tons... So just stick to the grey area, dear, and keep your emotions to yourself. If you start feeling it's mutual love, then go ahead [and express your feelings]!"

The following two examples illustrate lexical differences between the Gulf (example #3, from the Qatari variety) and Egyptian (example # 4). In example (3), the equivalent of the English word "I want" is "ابي", whereas in example (4), the equivalent is "عايز":

- (3) وحده داقه علي موظف بنك : ممكن استفسر : تفضلي عن شنو اختي :ابي اشوف كم برصيدي : اختي شنو اخر عملية سويتيها :ولاده. يقولك الموظف تقاعد

  Eng. "One time, a lady called a bank employee: "Hello, can I inquire about my balance?" "Of course sister, what was the last 'operation' you made?" "A delivery!" People say that he [the employee] quit his job after this call."

- (4) انا بس عايز فرصة كمان وانا هاثبتلك انى مش هتغير وانك تروح تضرب دماغك فى الحيط اسهل

  Eng. "I only need another chance to prove to you I won't change, and that you'd better go hit your head on the wall."

Classifications of dialects in general can easily gloss over distinctions between language variants. Many classifications of Arabic dialects, the one classification used by Habash (2010) being no exception, does. The limitations of some of these classifications include that differences between variants, especially across countries or regions of countries, can be significant. For example, varieties of Arabic in Egypt and the Sudan can be very different at various linguistic level (e.g., lexical, morphological, syntactic). These types of regional variations (Gonçalves and Sánchez, 2014) are pervasive, and even within each of these countries there exists further, more nuanced variations. The Arabic of Egypt's Sinai (north east of the country) is different from that of Cairo (the capital, which is situated toward the North), which is still different from that of Alexandria (north west, on the Mediterranean). Indeed, the linguistic literature shows how language can vary even within different parts of the the same city (Labov, 1964; Orton et al., 1998; Trudgill, 1974), thus creating micro-dialects within the same dialect. To the best of our knowledge, these fine-grained variations within countries has not been studied in Arabic NLP. Our work seeks to take a first step toward enabling the bridging of this important gap.

2.2. Computational Treatment of Arabic Dialects.

Early NLP work on Arabic dialects focused on collecting datasets that would enable the investigation of these

[2]Nizar Habash (personal communication, December, 2011) also points out he found this classification to be the 'most common' in the literature and hence he opted for it in his book.

dialects. A number of these pioneering studies focused on collecting data from blogs (Diab et al., 2010; Elfardy and Diab, 2012; Al-Sabbagh and Girju, 2012; Sadat et al., 2014), the general Web (Al-Sabbagh and Girju, 2012), comments on online news sites (Zaidan and Callison-Burch, 2011), or building dialectal lexica (Diab et al., 2014). Other works have dealt with detecting one or more of the Levantine, Gulf, and Egyptian dialects (Elfardy and Diab, 2013; Zaidan and Callison-Burch, 2011; Zaidan and Callison-Burch, 2014; Cotterell and Callison-Burch, 2014). Works covering other dialects (e.g., from countries like Tunisia, the Sudan, Qatar, Bahrain) include (Sadat et al., 2014), although they exploited small datasets (mostly $< 5K$ sentences from each country). Closer to our work is (Mubarak and Darwish, 2014) who report collecting a dataset of 123 million tweets covering Egyptian, Levantine, Iraqi, Maghrebi dialects. Our work follows (Mubarak and Darwish, 2014)'s lead, while developing a dataset almost twice the size (and from as twice countries). In addition, our work compares preferably to (Mubarak and Darwish, 2014) in that our data has more nuanced labels (i.e., at the city level).

Also related to our research is recent work on discriminating similar languages, e.g., via the VarDial workshop (Malmasi et al., 2016; Zampieri et al., 2017) where some works focused on Arabic (Malmasi and Zampieri, 2017; Ionescu and Butnaru, 2017). Some works also focus on Arabic dialect identification in speech transcripts, e.g., (Malmasi and Zampieri, 2016). Again, our work has wider scope and coverage. We now turn to describing our dataset.

3. Dataset

In order to develop our dataset, we exploit several in-house corpora (i.e., a total of > 1 billion tweets) covering the 10 Arab countries from the set {*Oman, Egypt, Iraq, Jordan, Kuwait, Palestine, Qatar, KSA, UAE, and Yemen*}. Our in-house data were collected using Twitter API using several bounding boxes over multiple Arab countries. As such, the data are diverse as we do not use any specific seeds to crawl. In addition, the data cover $\sim$ the last 5 years (i.e., $2013 - 2018$). To acquire location labels on the data, we use the Python geocoding library geopy [3], which helps locate the coordinates of addresses (e.g., 2103 Charleston Rd, Mountain View, CA 55321, USA), cities (e.g., Seattle), countries (e.g., Yemen), and landmarks (in the form of a co-ordinates, e.g., $49.264031, -123.246179$) based on third-party geocoders and a number of other data sources [4]. More specifically, we use "OpenStreetMap Nominatim" [5] as a third party tool. We acquire a total of $234,801,907$ tweets from 29 Arab cities, representing 10 Arab countries. The 29 cities in our data are shown in Figure 2. In addition, statistics of the dataset are provided in Table 1.

[3] https://github.com/geopy/geopy.

[4] A list of these third-party geocoders and other sources can be found at: https://github.com/geopy/geopy/tree/master/geopy/geocoders.

[5] https://nominatim.openstreetmap.org/

Country	City	# Users	# Tweets
Egypt	Alexandria	6,613	9,839,453
	Cairo	20,544	29,597,031
	Giza	2,499	3,252,507
Iraq	Baghdad	2,447	2,617,790
	Karbala	235	223,885
	Zubair	238	266,777
Jordan	Amman	2,943	4,132,434
	Aqaba	53	57,066
	Irbid	332	431,016
Kuwait	Ahmadi	396	678,050
	Hawally	142	200,757
	Kuwait City	2,827	5,071,420
Oman	Muscat	2,247	2,883,711
	Salalah	298	339,296
	Sohar	256	340,806
Palestine	Gaza	1,931	2,754,851
	Nablus	113	146,967
	Ramallah	167	216,245
Qatar	Al-Rayyan	466	694,715
	Doha	4,025	6,394,218
KSA	Dammam	5,560	8,483,462
	Jeddah	29,045	42,840,379
	Riyadh	61,697	90,410,407
UAE	Abu Dhabi	5,074	8,093,556
	Al Ain	497	822,870
	Dubai	7,050	11,436,814
Yemen	Aden	481	674,345
	Sana	1,200	1,610,875
	Taiz	229	290,204
All	–	**159,605**	**234,801,907**

Table 1: Data statistics: Number of users and tweets per city, for 29 cities covering 10 different countries representing the Eastern part of the Arab world.

4. Dialect Annotation

We perform a pilot dialect annotation task with varieties from the 10 Arab countries in our data. We provide each annotator with data representing a single country at a time, after explaining how the data were collected and the goal of our work. We then cast the dialect annotation task as a 3-way decision where judges choose whether a tweet (1) represents the dialect of the given country **(DA)**, (2) **(MSA)**, or (3) any other dialect **(OTHER)**. In the case of **(OTHER)**, we do not ask annotators to specify what other dialect the tweet exactly belongs to, thus reducing cognitive overload and keeping the task simple. Judges labeling the data are college-educated native Arabic speakers. A total of 5 annotators performed the task and we ensured each annotator is fluent with the variety they worked on. In almost all cases, the annotator either comes from the country from which the data are derived or from a directly neighboring country. We asked annotators to follow a number of steps for labeling tweets with language they cannot understand, including consulting with one another and online. To ensure quality, any tweet whose language

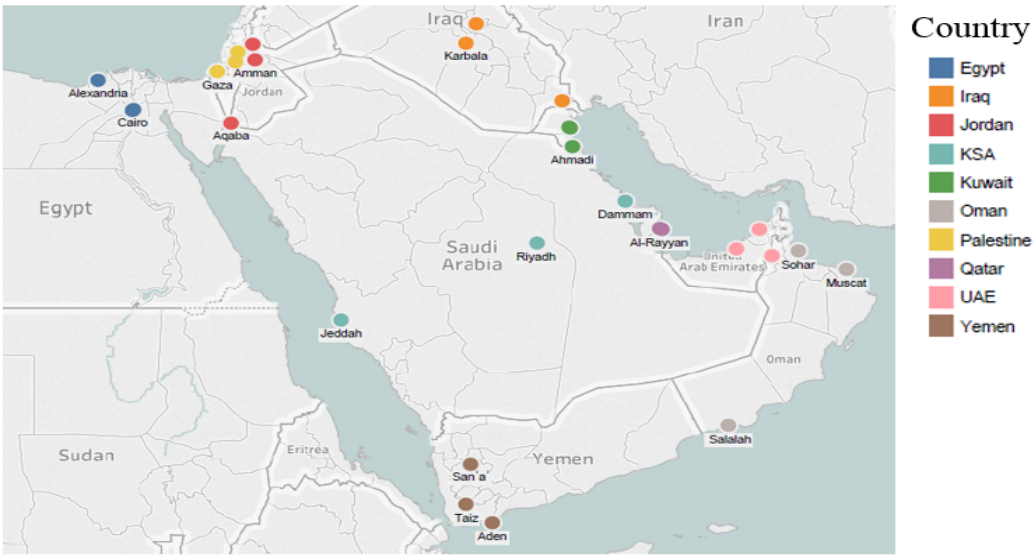


Figure 2: Cities represented in our data. Each city is shown as a dot; cities belonging to the same country are shown in similar color.

was still judged unintelligible after following these steps was excluded from the data. All non-Arabic tweets were removed from the data automatically before annotation using a simple character count method. In addition, we asked annotators to manually remove any non-Arabic tweets that may have remained after automatic filtering.

For this pilot annotation, we select a sample of 250 tweets per country (a total of $2,500$ tweets). Each tweet was labeled by two judges. Table 2 shows inter-annotator agreement on the task. As table 2 shows, annotators agree with a Cohen's Kappa $(K) = 67\%$, on average. This reflects 'substantial' agreement (Landis and Koch, 1977). Table 2 also shows that annotators agreed less on the cases of Yemen ($(K) = 40\%$), Oman ($(K) = 55\%$), and Qatar ($(K) = 58\%$), and Jordan ($(K) = 60\%$). This may be due to one or more of several factors. For example, annotators reported not being able to distinguish the dialects coming from some cities that closely neighbor other countries. For example, annotators had difficulty distinguishing tweets from Al Ain (UAE) and Sohar (Oman). In addition, annotators reported less acute difficulty working on data from countries in which there seems to be users originally from other countries. For example, users with Egyptian dialect tweeted from Qatar. For these reasons, we believe the political situation and immigration waves in the Arab world are important factors for dialect data collection. Conceivably, there would be cases where there are cities near borders where more than one dialect are used. Our research did not investigate these cases. However, we note this as an important direction for future research.

Country	Cohen's Kappa *(K)*
Egypt	0.73
Iraq	0.71
Jordan	0.60
KSA	0.89
Kuwait	0.71
Oman	0.55
Plastine	0.88
Qatar	0.58
UAE	0.60
Yemen	0.40
Avg.	**0.67**

Table 2: Inter-annotator agreement for the human dialect identification task.

5. Distributed Representations of Dialects

5.1. Building Word Vectors Model

Distributed representations of language at various levels of granularity, e.g., words and phrases (Mikolov et al., 2013; Pennington et al., 2014) or sentences (Kiros et al., 2015) boost performance on various NLP tasks. Zahran et al. (2015) pioneered efforts to build word embedding models for Arabic. In spite of the usefulness of a model built on MSA, it is expected to suffer coverage issues (i.e., sparsity) when applied to dialectal data. To alleviate this problem, we build a word vectors model exploiting our data. We adopt the pre-processing pipeline of (Zahran et al., 2015). Namely, we remove any non-unicode characters, normalize *Alif maksura* to *Ya*, reduce all *hamzated Alif* to plain *Alif*, and remove all non-Arabic characters. Additionally, to clean noise associated with social media non-standard typography, we reduce all letter repetition of > 2 characters to only 2. We build a skip-gram model with 300 dimensions, with a minimal word count $= 100$ words, and a window size of 5 words on each side of a target word. We use the gensim [6] implementation for the word2vec tool [7].

5.2. Hand-Picked Examples

In order to demonstrate the capacity and richness of our word vectors model, particularly in terms of dialectal word coverage, we ask our annotators to identify a list of dialectal words from the data

[6] https://radimrehurek.com/gensim/models/word2vec.html.

[7] https://code.google.com/archive/p/word2vec.

Country	Arabic	English	Most similar words
Egypt	ازاي	how	كده، بردو، دلوقتى، علشان، دلوقت
	معرفش	I don't know	مشعارف، دلوقتى، معرفوش، بردو، بجد
	جدعان	men	جماعه، خونا، شوباب، مسريين، كماعه
	بتعدي	pass by	هتعدى، بتمر، بتيجى، بتمشى، بتجرى، بتبقى
	مفيش	nothing	ومافيش، مفهاش، معندناش، مقيش، معنديش
KSA	ماننام	don't we sleep	ننام، ننوم، نرقد، مانموت، ننخمد
	هالكلام	this talk	هالحكى، هالخبر، هالحجه، الشئ، هالتصريح
	سم	name it	تسم، سمى، ذبو، آسلوب، مرب
	ورع	boy	بزر، هطف، مهابطى، طعس، شايب، زلابه
	ايوه	yes	ايوا، ايون، ايبه، اهوه، ليه
UAE	يالس	sitting	قاعد، جالس، داش، مستعيل، مجابل
	شحالك	how are you?	شلونك، شخبارك، شحالج، شلونج، شالاخبار
	سير	go	معرقل، وعرقله، وتسيير، سيرها، حادثى
	غرشه	sip	قلاص، قوطى، قاروره، علبه، قنينه
	يرمسون	they talk	يتكلمون، يرمس، يحوفون، شيقولون، يتحرطمون

Table 3: A set of hand-picked words from the top 3 countries (Egypt, KSA, and UAE) and their most similar words in our word embeddings model.

(10 words from each of the 10 countries for a total of 100 words). We then pick a random sample of 5 words from each of the top 3 countries, i.e., {*Egypt, KSA, UAE*}, in our data. Next, we use each item in this list to query the model for the 5 most similar words. Table 3 shows some examples. As Table 3 illustrates, for each dialectal word, the model captures not only morphological variants of the word: e.g., "ومافيش" and "مفهاش" (Eng. "there is nothing") for EGY, but also its orthographic variants: e.g., "شحالك" and "شحالج" (Eng. "how are you?"), for UAE. The model also captures similarity inter-dialectaly. For example, given the query "هالكلام" (Eng. "this argument) in KSA, the model returns "هالحكى" which belongs to the Levantine dialect. Moreover, given a query word, the model identifies syntactically and semantically related words. For example, the words "جدعان" (Eng. "men") and "مسريين" (Eng. sarcastic in a political context for "Egyptians") frequently occur after the vocative particle "يا" (Eng. "oh, you") and hence share syntactic context. These retrieved words also often times occur in humorous/sarcastic contexts, which implies the model may be capturing some pragmatic relationships in the data.

5.3. Lexical Coverage

Under the training parameters listed earlier, our word embeddings model ended with about a total of $500K$ words. Clearly, there is lexical overlap between different Arabic varieties and each dialect would employ words that are functional in MSA, with varying degrees of semantic relatedness. Our interest here is investigating this space, simplistically as a starting point: We query an embeddings model built with corpora that are overridingly MSA, to check how much coverage it affords for entries in our model. Our intuition is that the more frequent a word is in our data, the higher the likelihood it will be covered in an MSA-based model, and *vice versa*. This also implies that a list of words randomly sampled from our data should have a coverage in an MSA model that is neither as high as these most frequent in our data nor as low as those least frequent in our data. We test this intuition, finding it holding true as shown in Figure 3. In other words, our data have wider lexical coverage than is captured in the (Zahran et al., 2015) model.

Figure 3 shows distribution of coverage in the MSA model (i.e., (Zahran et al., 2015)) with highest, lowest, and random word frequencies as extracted from our data. In each case, we limit to frequencies from the set {*1k,2k,3k,4k,5k*}. Interestingly, this simple test also shows how certain dialects overlap with MSA more than others. For example, while coverage is $< 50\%$ as is clearly shown by the least common line in the Yemen data (Figure 4), coverage is $> 50\%$ for both EGY and KSA. Since the data representing each dialect are not of equal size, we cannot make claims as to semantic distance between dialects at this point, although this is one question we would like to eventually be able to answer.

6. Conclusion

In this paper, we reported the development of a new large-scale dataset for a number of Arabic dialects. The data are tagged at the city level. We also reported a pilot annotation study, identifying some of the challenges associated with fine-grained 'country-based' dialect annotation. Finally, we investigated the dialectal coverage of our data using a word vectors model. The distributed representations enabled by the model, as we show, have richer coverage than available models. Together with similar works, we believe this line of research opens up interesting frontiers for dialectal Arabic NLP. In the future, we will perform a wider scale annotation of the data and evaluate the distributed representations models in a number of downstream tasks.

7. Acknowledgement

This work was partially funded by UBC Hampton Grant #12R74395 and UBC Work Learn Award to Muhammad Abdul-Mageed. The research was also enabled in part by support provided by WestGrid (`https://www.westgrid.ca/`) and Compute Canada (`www.computecanada.ca`).

8. Bibliographical References

Abdul-Mageed, M. M. and Herring, S. C. (2008). Arabic and english news coverage on aljazeera. net.

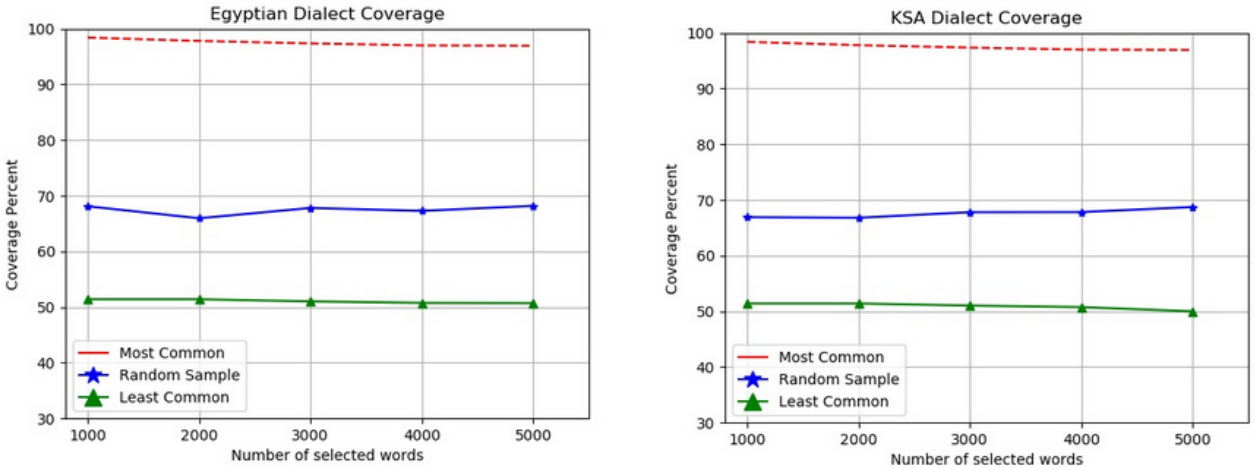


Figure 3: Distribution of dialectal lexical coverage in an MSA word vectors model (Zahran et al., 2015) on words from 2 countries (Egypt and KSA) in our data. The more frequent a word is in our data, the higher the likelihood it is covered in the MSA model: **`Red, dotted line`**: most common words in our data are covered best (i.e., close to 100%) in MSA model. **`Blue, starred line`**: Random frequency words are covered at values between 60% and 70% in MSA model. **`Green, triangled line`**: Least frequent words are sparse in MSA model (i.e., around, *or* $< 50\%$).

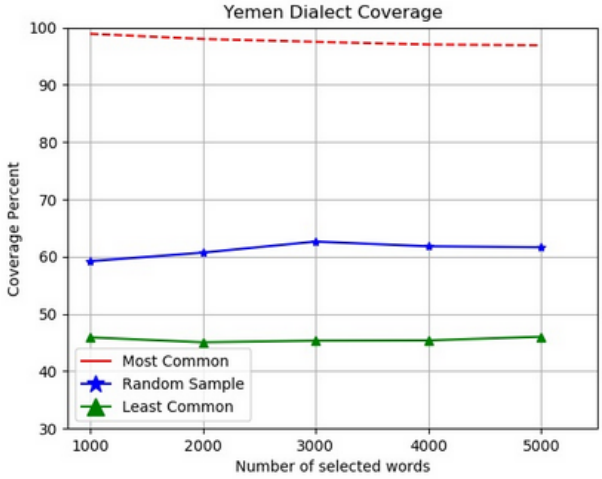


Figure 4: Distribution of dialectal lexical coverage in an MSA word vectors model (Zahran et al., 2015) on words from Yemen in our data.

Abdul-Mageed, M. M. (2008). Online news sites and journalism 2.0: Reader comments on al jazeera arabic. *tripleC: Communication, Capitalism & Critique. Open Access Journal for a Global Sustainable Information Society*, 6(2):59–76.

Abdul-Mageed, M. (2015). *Subjectivity and sentiment analysis of Arabic as a morophologically-rich language*. Ph.D. thesis, Indiana University.

Al-Sabbagh, R. and Girju, R. (2012). Yadac: Yet another dialectal arabic corpus. In *LREC*, pages 2882–2889.

Badawi, M. (1973). Levels of contemporary arabic in egypt. *Cairo: Dâr al Maârif*.

Bassiouney, R. (2009). *Arabic sociolinguistics*. Edinburgh University Press.

Bateson, M. C. (1967). *Arabic language handbook*, volume 3. Georgetown University Press.

Cotterell, R. and Callison-Burch, C. (2014). A multi-dialect, multi-genre corpus of informal written arabic. In *LREC*, pages 241–245.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *Lrec workshop on semitic language processing*, pages 66–74.

Diab, M. T., Al-Badrashiny, M., Aminian, M., Attia, M., Elfardy, H., Habash, N., Hawwari, A., Salloum, W., Dasigi, P., and Eskander, R. (2014). Tharwa: A large scale dialectal arabic-standard arabic-english lexicon. In *LREC*, pages 3782–3789.

Elfardy, H. and Diab, M. T. (2012). Simplified guidelines for the creation of large scale dialectal arabic annotations. In *LREC*, pages 371–378.

Elfardy, H. and Diab, M. T. (2013). Sentence level dialect identification in arabic. In *ACL (2)*, pages 456–461.

Gonçalves, B. and Sánchez, D. (2014). Crowdsourcing dialect characterization through twitter. *PloS one*, 9(11):e112074.

Habash, N. Y. (2010). Introduction to arabic natural language processing. *Synthesis Lectures on Human Language Technologies*, 3(1):1–187.

Holes, C. (2004). *Modern Arabic: Structures, functions, and varieties*. Georgetown University Press.

Ionescu, R. T. and Butnaru, A. M. (2017). Learning to identify arabic and german dialects using multiple kernels. *VarDial 2017*, page 200.

Jurgens, D., Tsvetkov, Y., and Jurafsky, D. (2017). Incorporating dialectal variability for socially equitable language identification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 51–57.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Labov, W. (1964). *he social stratification of En- glish in New York City*. Ph.D. thesis, Columbia uni- versity.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Malmasi, S. and Zampieri, M. (2016). Arabic dialect identification in speech transcripts. *VarDial 3*, page 106.

Malmasi, S. and Zampieri, M. (2017). Arabic dialect identification using ivectors and asr transcripts. *VarDial 2017*, page 178.

Malmasi, S., Zampieri, M., Ljubešic, N., Nakov, P., Ali, A., and Tiedemann, J. (2016). Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task. *VarDial 3*, page 1.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mubarak, H. and Darwish, K. (2014). Using twitter to collect a multi-dialectal corpus of arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 1–7.

Orton, H., Sanderson, S., and Widdowson, J. (1998). *The linguistic atlas of England*. Psychology Press.

Palva, H. (2006). Dialects: classification. *Encyclopedia of Arabic Language and Linguistics*, 1:604–613.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Ryding, K. C. (2005). *A reference grammar of modern standard Arabic*. Cambridge university press.

Sadat, F., Kazemi, F., and Farzindar, A. (2014). Automatic identification of arabic language varieties and dialects in social media. *Proceedings of SocialNLP*, page 22.

Trudgill, P. (1974). Linguistic change and diffusion: Description and explanation in sociolinguistic dialect geography. *Language in society*, 3(2):215–246.

Versteegh, K. (2001). Linguistic contacts between arabic and other languages. *Arabica*, 48(4):470–508.

Versteegh, K. (2014). *The arabic language*. Edinburgh University Press.

Zahran, M. A., Magooda, A., Mahgoub, A. Y., Raafat, H. M., Rashwan, M., and Atyia, A. (2015). Word representations in vector space and their applications for arabic. In *CICLing (1)*, pages 430–443.

Zaidan, O. F. and Callison-Burch, C. (2011). The arabic online commentary dataset: an annotated dataset of informal arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41. Association for Computational Linguistics.

Zaidan, O. F. and Callison-Burch, C. (2014). Arabic dialect identification. *Computational Linguistics*, 40(1):171–202.

Zampieri, M., Malmasi, S., Ljubešić, N., Nakov, P., Ali, A., Tiedemann, J., Scherrer, Y., and Aepli, N. (2017). Findings of the vardial evaluation campaign 2017.

Visualizing the *Dictionnaire des Régionalismes de France* (DRF)

Ada Wan
University of Zurich
Zurich, Switzerland
ada.wan@uzh.ch

Abstract

This paper presents CorpusDRF, an open-source, digitized collection of regionalisms, their parts of speech and recognition rates, published in *Dictionnaire des Régionalismes de France* (DRF, "Dictionary of Regionalisms of France") (Rézeau, 2001), enabling the visualization and analyses of the largest-scale study of French regionalisms in the 20th century using publicly available data. CorpusDRF was curated and checked manually against the entirety of the printed volume of more than 1000 pages. It contains all the entries in the DRF for which recognition rates in continental France were recorded from the surveys carried out from 1994 to 1996 and from 1999 to 2000. In this paper, in addition to introducing the corpus, we also offer some exploratory visualizations using an easy-to-use, freely available web application and compare the patterns in our analysis with that by (Goebl, 2005a) and (Goebl, 2007).

Keywords: digitization and release of heritage data, visual dialectometry, data visualization

1. Introduction

The DRF comprises data from the last large-scale study of lexical regionalisms in continental France in the 20th century, which took place from 1994 to 1996 and from 1999 to 2000. This paper describes a project carried out in 2016 with the following goals:

i. to curate the data on recognition rates published in the DRF as CorpusDRF,

ii. to provide preliminary analysis of the DRF data through hierarchical clustering, including exploratory visualization of some potential "regions" of lexical regionalisms based on the DRF, and

iii. to evaluate the result from [ii.] by comparing with previous work performed by (Goebl, 2005a) and (Goebl, 2007) and by examination of the nature of our approach in relation to that of the survey design.

2. Data

The DRF is the first comprehensive book presenting a careful description of regionalisms of mainland France with up to 1,100 headword entries, furnished with definitions, examples, comments, citations and quotes, along with results from a survey of regionalisms carried out between 1994 and 1996 in France (EnqDRF), with a supplementary survey between 1999 and 2000, in form of recognition rates for the regionalisms in their relevant departments/regions as well as maps depicting the diatopic distribution of 330 regionalisms throughout France on the departmental level. Linguists from different regions compiled a list of known lexical facts characteristic of their regions which was then turned into a questionnaire – this resulted in about 4,500 facts being tested.

A few years ago, much of the DRF data fell victim to an administrative housekeeping effort. The corpus we curated manually from the printed version of this heritage volume is the only open-source digital version of the DRF recognition rates mapping entries (annotated with their parts of speech) with the corresponding departments.

2.1. Data documentation and profile

Entries (indicating a headword, a variant of a headword, or a multi-word expression involving the headword) with recognition rates published in the DRF were transcribed manually into a plain text file which was then processed into a data matrix in a TSV file with entries as rows and the 94 departments of mainland France as columns. (The department *Mayenne* was not included in the survey, and this did indeed result in an empty column in the matrix.) Most[1] entries in the DRF report a set of recognition rates for a certain number of departments/regions in France. Note that these departments/regions are pre-selected by specialists and that not the same set of words were surveyed in each department. Since our objective is to map entries by departments, names of regions such as *Champagne* and *Brittany*, had to be resolved into department names, leveraging cues from the legend and individual maps in the DRF wherever necessary (see Appendix A for the list of region-department correspondences used here). In the DRF, words or senses that were not surveyed are published with a "Ø", whereas words or senses which were asked but not recognized by anybody have rates of "0" – we did not record the former (i.e. we represented such absence of data as empty cells in our matrix) and documented the latter with an actual "0" (zero). For some words, there are multiple sets of recognition rates recorded for their multiple senses, while for other words, sets of recognition rates for all the senses were merged and published as one set. Whenever there is a different set of departments for a distinct sense or expression, we tried to list it as a separate entry (with a sense number preceded by an underscore following the headword). In more complicated cases (e.g. a sense of a multi-sense headword having multiple senses/usages/variants), or cases where the merging of recognition rates seemed sensible, we merged the different rate sets and took the highest rate for each department reported. Words were surveyed based on the special senses they have taken on in a particular region,

[1] Please note that not all entries had recognition rates published in the DRF.

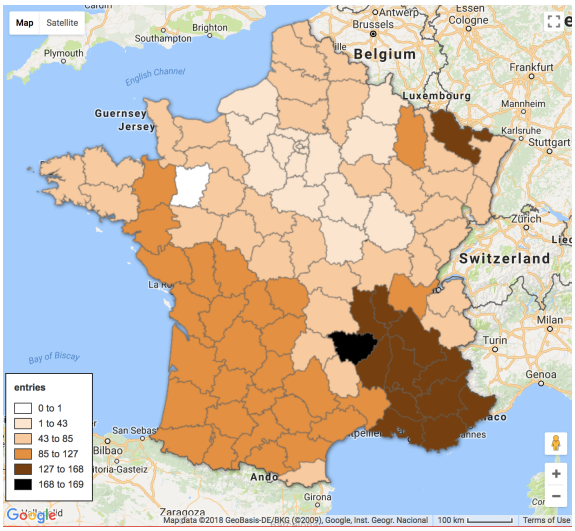


Figure 1: Entry distribution across departments – *Haute-Loire* (in black) has the highest number of entries at 168 while in *Mayenne* (white) no words were surveyed. (The intervals in the legend are half-closed with end values excluded.)

we therefore ask the users to refer to the DRF for further context.

All in all, we obtained 936 entries for CorpusDRF. Including the empty column for *Mayenne*, we have a data matrix with 936 rows and 94 columns. Of these 87984 ($936 \times 94 = 87984$) cells, however, only 7248 were not empty, i.e. only 8.24%[2] of the matrix are filled. (Disregarding *Mayenne* would yield a matrix density rate of 8.33%.)

The distribution of the entries with recognition rates is found to be rather unbalanced – more entries were asked in some departments than others (as shown in Fig. 1)[3]. For the 94 departments, we have a range of counts from 0 (*Mayenne*) to 168 (*Haute-Loire*) with a mean of 77.11, a median of 79.50, and a standard deviation of 38.07. (Discounting *Mayenne*, the figures remain similar with minimum at 5, maximum 168, mean 77.94, median 80, standard deviation 37.42.) The dichotomy between Northern/Central France and Southern/Southeastern France is clearly visible: there is almost a 50% gap between the two adjacent departments of *Allier* (count: 44) in the north and *Puy-de-Dôme* (80) in the south (the "southern department" with the lowest count on this "border").

CorpusDRF will be made freely available without warranty on the ELRA/LREC website through the "Share Your LRs" initiative as well as through *Google Fusion Tables*[4] as "CorpusDRF".

3. Related Work

Our approach is rather similar to most dialectometrical procedures. Dialectometry is a data-driven approach to identify dialectal regions using statistical techniques on quantitative data. It is also sometimes known as "data-driven dialectology" or "quantitative dialectology". Hans Goebl was a pioneering figure in this field as he was the first to use a computer for the calculation of linguistic distances and was hence able to process data on a bigger scale. He also introduced cluster analysis as a means of numerical taxonomy in dialectometry (Pickl and Rumpf, 2012). His collaboration with Edgar Haimerl in the Dialectometry Project at the University of Salzburg since 1998 led to the development of the software *VisualDialectoMetry* (VDM)[5] which pioneered in the automatic combination of cartography and dialectometry. In the 2000s, visualization in linguistic analyses were also enhanced through the popularization of multidimensional scaling (MDS) by Wilbert Heeringa (Heeringa, 2004) and factor analysis by John Nerbonne (Nerbonne, 2006). Nerbonne et al. (2011) also released *Gabmap* as an online application for dialectology which allows researchers to inspect their data and visualize through MDS. For a more detailed overview of the historical development in (visual) dialectometry and a systematic overview in dialectology, please refer to Bauer (2009) and Boberg et al. (2018) respectively. For a comparison between VDM and *Gabmap*, please refer to Kellerhals (2014). In this paper, we report our methods and results in exploratory data analysis and visualization using *R* (R Core Team, 2015) and *Google Fusion Tables* (Gonzalez et al., 2010).

4. Data Visualization

Google Fusion Tables is a freely available web application that integrates seamlessly with *Google Maps*[6]. Producing a map visualization of data that contain geo information, in form of place names or Keyhole Markup Language (KML) data which define polygons on the map, is as easy as clicking a button. One can easily upload custom data files in tabular format and combine these with pre-existing public data resources. In our case, we imported our data matrix as a CSV file. To visualize the data in terms of their corresponding departments, we joined it with an already existing Fusion Table of KML shapes, made publicly available through *Fusion Tables* by GEOFLA®[7].

To view a map for an individual entry, select *Change feature styles* under *Feature map*, then under *Polygons/Fill color/Buckets*, select entry under *Column*. Fig. 2 shows how our map for the entry *verrine* compares to the one in the DRF in Fig. 3[8].

We also uploaded the results of our clustering experiments as CSV files with departments as row names and their respective cluster IDs as columns. These files were then merged with the already prepared Fusion Table.

[2] All numbers reported in this paper are rounded to 2 decimal places.

[3] See Appendix B for a list of these counts.

[4] `https://sites.google.com/site/fusiontablestalks/home`

[5] `http://ald.sbg.ac.at/dm/germ/VDM/`

[6] `https://maps.google.com/`. All colored maps we created for this paper have been made using Map data ©2018 GeoBasis-DE/BKG (©2009), Google, Inst. Geogr. Nacional.

[7] `https://fusiontables.google.com/data?docid=1g_ydg74ooUSBzNfQBHOIdgrOKhxZD_92In8xTDg#rows:id=1`

[8] Figure taken from `http://www.atilf.fr/spip.php?rubrique86`.

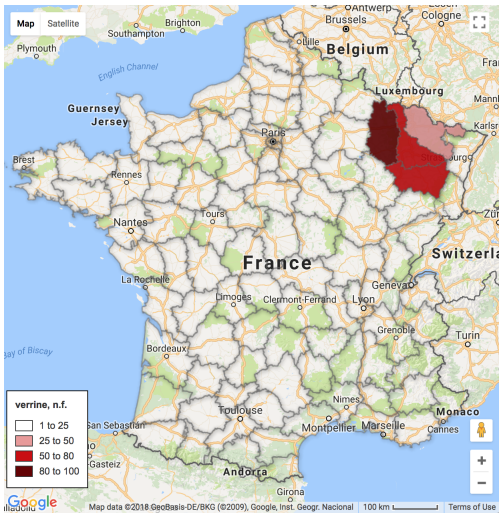


Figure 2: Our visualization of the entry *verrine*

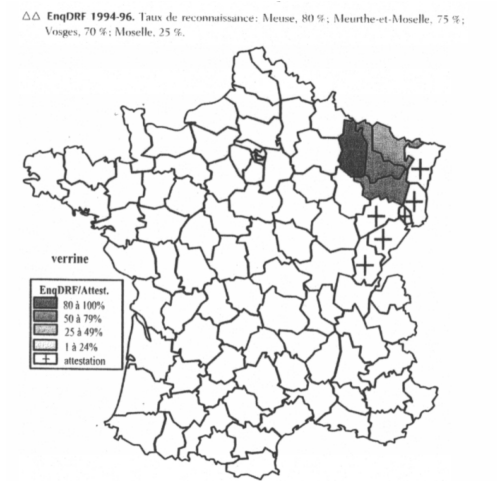


Figure 3: The entry *verrine* as printed in the DRF. (Note that "attestation" is only graphed in some of the maps in the DRF, not documented in text.)

5. Clustering

In order to detect potential regional groupings based on the recognition rates, we computed a distance matrix for our clustering with the continuous values we recorded as recognition rates by means of the Euclidean distance measure. The distance (d) between points x and y in a Euclidean n-dimensional space is defined as follows:

$$d(x, y) = d(y, x) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \qquad (1)$$

5.1. Treatment of empty values: NAs vs. 0s

In our data documentation phase, we made a distinction in coding "missing values" (i.e. the word was not asked in the survey for the department) as NAs as opposed to 0s (meaning 0% recognition rate). When NAs are treated as NAs, only words for which recognition rates are available for both will be considered in the distance/similarity computation between two departments. This can be rather infeasible with our very sparse dataset – e.g., if 2 departments differ a lot in the distribution of their entries (i.e. which words got surveyed), but have 2 out of 5 common entries the same, then their dissimilarity score would be 0, whereas for 2 other departments with the same entry distribution, i.e. they were expected by the experts to share a common vocabulary of regionalisms, if all values of their recognition rates are different, even if slightly, they would have a larger dissimilarity score than the former pair. With a larger and denser dataset, disregarding a few empty values here and there may not be as problematic, yet not in our case. There is opportunity for leveraging various imputation techniques here, but we will leave that for future work. Instead we return to the context of the EnqDRF – if a word is not expected (by the specialists) to be known much or at all in a department, it is not surveyed there. This is an assumption that may or may not hold in reality and can only be proven with future empirical research with an extended survey.

In CorpusDRF, three versions of the data are provided – one with NAs left as empty, one with NAs imputed with 0s, and one imputed with -1s. The first two versions are for processing in *R*, the last is useful for an easy visualization in *Fusion Tables*.

5.2. Results and comparison with Goebl's work on DRF

Hierarchical clustering algorithms were used as these are standard for taxonomical tasks similar to ours. Especially since we expected the distribution of our data to be a bit skewed, we performed hierarchical agglomerative clustering (HAC), i.e. we clustered agglomeratively (bottom- up), as opposed to divisively (top-down) so to minimize effect of the global distribution. This is also analogous to Goebl's method. We first tried to replicate Goebl's analysis of 4 clusters (Goebl, 2007) using Ward's method with *R*'s built-in `hclust` function. Fig. 4 shows the analysis by Goebl while Fig. 5 shows ours with `hclust` (since `hclust` does not support missing values, we imputed the NAs as 0s).

A clustering algorithm that does handle missing data in *R* is Eisen's algorithm, available via the `eisenCluster` method from the `hybridHclust` package. The resulting map using this algorithm is presented in Fig. 6. Note how the handling of NAs could have a huge impact on clustering results.

We also experimented with the unweighted average linkage (UPGMA) (Sokal and Michener, 1958). With UPGMA, the distance between two clusters is the average distance between all pairs of observations. UPGMA tends to join clusters with small variances and has slight bias toward producing clusters with the same variance (SAS Institute Inc., 2009) – this would be the preferred behavior for our data as we'd want the pairwise distances of all points to be small. Fig. 7 shows our results using UPGMA and `hclust`.

Goebl provided dialectometrical analysis of the DRF first by employing the 342 maps of individual word entries with recognition rates (2005a) and in (2007) the entire DRF data. The recognition rates presented in his analyses followed the format in the DRF: discretized into 6 bins. Although our methods differ – he used ordinal variables for recognition

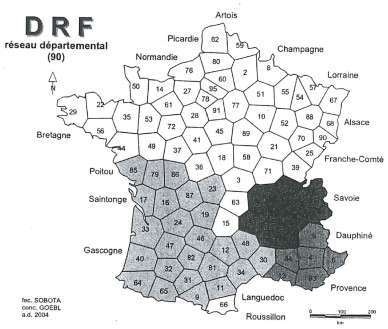


Figure 4: Visualization of 4 clusters from Goebl's analysis; map 22 from (Goebl, 2007).

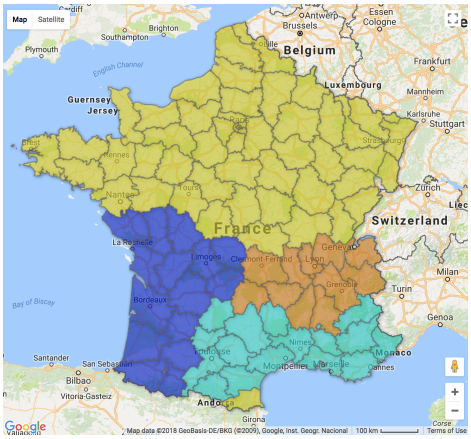


Figure 5: HAC with WARD (4 clusters) with NAs as 0s

rates and relative identity value ("Relativer Identitätswert", RIW) as similarity measure, whereas we recorded recognition rates from the DRF text as continuous values and used Euclidean distance, there are parallels in our analyses. Fig. 7 exhibits overlap with his analysis (Fig. 4) in that the 9 departments in the Southeast (*Haute-Loire*, *Loire*, *Rhône*, *Ain*, *Haute-Savoie*, *Savoie*, *Isère*, *Drôme*, and

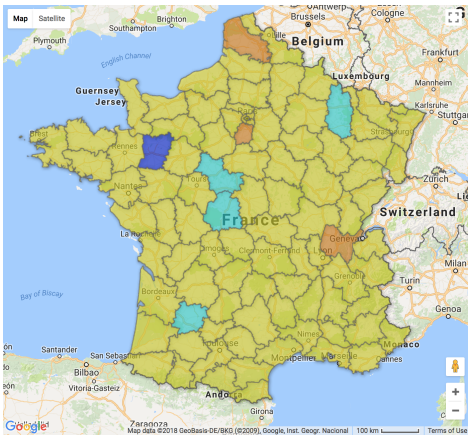


Figure 6: HAC using Eisen's algorithm (4 clusters) with NAs as NAs

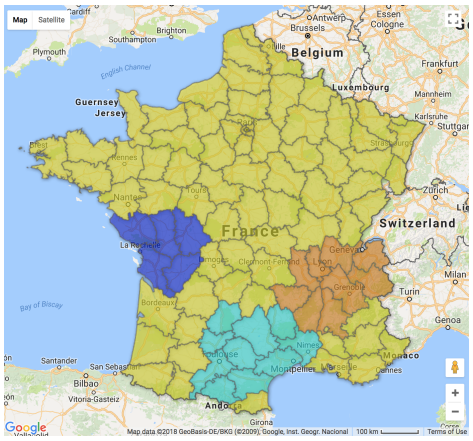


Figure 7: HAC with UPGMA (4 clusters) with NAs as 0s

Ardèche) form a cluster of their own. This cluster differs from the area with denser entry distribution in the Southeast in Fig. 1, suggesting that the pattern may not merely be contingent on the sheer counts of observations. Yet, the department *Pyrénées-Orientales* in southern France by the Spanish border shows the tendency to be clustered with the relatively homogenous northern half of France in all of our analyses shown here, mirroring its outlier status in entry distribution.

In general, we concur with Goebl's observation that the DRF data might be a bit too sparse and too unsystematic for proper dialectometrical analysis. But precisely because of this, we tried to map out some of these preliminary analyses so to inspire other researchers to further investigate.

6. Discussion

At the onset of this project, we had hoped to be able to identify French dialectal regions on the basis of the recognition rate data in a data-driven manner. That, however, did not pan out as expected not only due to data sparsity, but also because we have come to realize that it was not the intention of the editor/researchers of the DRF for the recognition rates to be evaluated quantitatively on a national level. No rigorous quantitative methods were applied in the survey design, as the main concern of the DRF was one of (qualitative) lexicography for humans – to investigate and compile a collection of regionalisms that were of questionable status in comparison to Standard French at the time. Entries with accompanying rate information were certainly asked, counts were tallied but the dataset was biased by the preconception of the experts who decided which words would be surveyed in which departments in the first place. Such a dataset could be more representative of the mental maps of the specialists than of the survey participants. That said, the publication of CorpusDRF could help effect an easy, clear, and systematic evaluation of the DRF. It could help us understand the perceptual dialectological situation of the researchers in the past so to design more comprehensive and quantitatively rigorous experiments and surveys to examine new regionalisms and to (re-)investigate the status and development of the old. (Many researchers in the past have advocated for more rigor and standard in experiment

design, readers may find sections on methodology and research design in Boberg et al. (2018), Schütze (2015), and Goebl (1993), among others, a worthy primer.) Data-driven methods can help us detect the hidden structures in the object(s) of investigation and, perhaps even more importantly, the hidden structures in the thought processes and values of the human investigators. That can in turn help much of (language) science progress in ways hitherto unanticipated and beyond a corpus-based approach of validation and refutation of hypotheses. Last but not least, CorpusDRF, in the absence of a digital version of DRF, can also function as a convenient reference facilitating studies in French lexicography, as it now enables an unprecedented production of maps for all and any of the 936 entries with recognition rate information.

7. Conclusion

In this paper, we described the DRF data profile and processing procedures and performed cartographic evaluation of some preliminary clustering results. Whereas it is clear that the nature of the EnqDRF and the DRF is different from that of a systematic dialectal survey and atlas (Goebl, 2005b), and that it would be presumptuous to generalize much from this dataset on a national level, it is nonetheless possible to gain insights on a smaller scale to better our understanding of the regions and of the language from the data as well as to use this as a stepping stone for future research on linguistic varieties and regionalisms. We hope that the accessibility of the CorpusDRF will inspire many more systematic and substantial surveys and dialectometrical analyses in the future.

8. Acknowledgements

We would like to thank Pierre Rézeau and l'ATILF (l'Analyse et Traitement Informatique de la Langue Française) for their support of the release of the curated data as a freely available language resource, as well as Hans Goebl, whose dedication in imbuing the field of dialectology and (Romance) linguistics with quantitive rigor and computational finesse (which we can imagine was a force a bit ahead of his time back in the days) shone through in our correspondences. Our thanks also go out to Mathieu Avanzi and Yves Scherrer for our valuable exchanges. All errors and misinterpretation remain solely the responsibility of the author.

9. Bibliographical References

Bauer, R. (2009). *Dialektometrische Einsichten: Sprachklassifikatorische Oberflächenmuster und Tiefenstrukturen im lombardo-venedischen Dialektraum und in der Rätoromania*. Ladinia monografica. Istitut ladin Micurà de Rü.

Boberg, C., Nerbonne, J., and Watt, D. (2018). *The Handbook of Dialectology*. Blackwell Handbooks in Linguistics. Wiley.

Goebl, H. (1993). Die dialektale Gliederung Ladiniens aus der Sicht der Ladiner. Eine Pilotstudie zum Problem der geolinguistischen "Mental Maps". *Ladinia*, 17:59–95.

Goebl, H. (2005a). Comparaison dialectométrique des structures de profondeur des cartes linguistiques du Dictionnaire des régionalismes de France (DRF) et de l'Atlas linguistique de la France (ALF). In M-D. Glessgen et al., editors, *La lexicographie différentielle du français et le Dictionnaire des régionalismes de France. Actes du Colloque en l'honneur de Pierre RÉZEAU*, pages 153–193. Presses Universitaires de Strasbourg, Strasbourg.

Goebl, H. (2005b). L'interprétation de cartes dialectométriques (débat GOEBL-RÉZEAU). *Revue de linguistique romane*, 69:299–306.

Goebl, H. (2007). La distribution spatiale des régionalismes du DRF comparée avec celle des données de l'ALF: un calibrage dialectométrique. In David Trotter, editor, *Actes du XXIVe Congrès International de Linguistique et Philologie Romanes (Aberystwyth 2004)*, volume I, pages 381–404. Niemeyer, Tübingen.

Gonzalez, H., Halevy, A., Jensen, C., Langen, A., Madhavan, J., Shapley, R., and Shen, W. (2010). Google Fusion Tables: Data management, integration, and collaboration in the cloud. In *Proceedings of the ACM Symposium on Cloud Computing (SOCC)*.

Heeringa, W. J. (2004). *Measuring dialect pronunciation differences using Levenshtein distance*. Ph.D. thesis, Rijksuniv. Groningen.

Kellerhals, S. (2014). Dialektometrische Analyse und Visualisierung von schweizerdeutschen Dialekten auf verschiedenen linguistischen Ebenen. Master's thesis, University of Zurich.

Nerbonne, J., Colen, R., Gooskens, C., Kleiweg, P., and Leinonen, T. (2011). Gabmap – a web application for dialectology. *Dialectologia: revista electrònica*, pages 65–89.

Nerbonne, J. (2006). Identifying linguistic structure in aggregate comparison. *Literary and Linguistic Computing*, 21(4):463–475.

Pickl, S. and Rumpf, J. (2012). Dialectometric concepts of space: Towards a variant-based dialectometry. In *Dialectological and folk dialectological concepts of space*, volume Linguae and Litterae, Band 17, pages 199–214. Walter de Gruyter, Berlin/New York.

R Core Team, (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Pierre Rézeau, editor. (2001). *Dictionnaire des Régionalismes de France*. De Boeck et Larcier s.a., Brussels, 1 edition.

SAS Institute Inc. (2009). SAS/STAT®9.2 User's Guide, Second Edition. `https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_cluster_sect012.htm`.

Schütze, C. (2015). *The empirical base of linguistics: Grammaticality judgments and linguistic methodology*. Classics in Linguistics. Language Science Press.

Sokal, R. R. and Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438.

A Region-department correspondences used

The following list contains the correspondences between *regional/departmental variants used in DRF → corresponding departments used in present work*:
Alsace → Bas-Rhin, Haut-Rhin
Aquitaine → Dordogne, Lot-et-Garonne, Gers, Hautes-Pyrénées, Pyrénées-Atlantiques, Landes, Gironde
Argonne → Marne, Ardennes, Meuse
Auvergne → Haute-Loire, Cantal. Puy-de-Dôme
Basse-Normandie → Calvados, Orne, Manche
Basse-Bretagne → Côtes-d'Armor, Morbihan, Finistre
Bourgogne → Saône-et-Loire, Côte-d'Or, Yonne
Bretagne → Ille-et-Vilaine, Côtes-d'Armor, Morbihan, Finistère, Loire-Atlantique
Centre-Ouest → Deux-Sèvres, Vienne, Charente-Maritime, Charente
Champagne → Aube, Haute-Marne, Marne, Ardennes
Dordogne (nord) → Dordogne
Franche-Comté → Haute-Saône, Territoire-de-Belfort, Doubs, Jura
Haute-Bretagne → Ille-et-Vilaine, Loire-Atlantique, Morbihan, Côtes-d'Armor
Haute-Loire (Velay) / Haute-Loire (nord-ouest) → Haute-Loire
Haute-Saône (sud)/(nord)/nord → Haute-Saône
Ile-de-France/Île-de-France → Paris, Seine-et-Marne, Yvelines, Essonne, Hauts-de-Seine, Seine-Saint-Denis, Val-de-Marne, Val-d'Oise
Languedoc oriental → Gard, Hérault, Aude
Languedoc occidental/Occidental → Ariège, Haute-Garonne, Tarn, Tarn-et-Garonne, Lot, Aveyron
Limousin → Creuse, Corrèze, Haute-Vienne
Loir-et-Cher sud / Loir-et-Cher (sud) → Loir-et-Cher
Lorraine → Meuse, Meurthe-et-Moselle, Moselle, Vosges
Meuse (nord) → Meuse
Moselle (est) / Moselle (sauf est) / Moselle romane → Moselle
Nord-Picardie → Somme, Aisne
Normandie → Seine-Maritime, Eure, Calvados, Orne, Manche
Picardie → Somme, Aisne, Oise
Provence → Alpes-de-Haute-Provence, Alpes-Maritimes, Var, Bouches-du-Rhne, Vaucluse
région lyonnaise → Rhñoe, Loire, Isère, Ain, Ardèche, Drôme, Saône-et-Loire
Roussillon → Pyrénées-Orientales

B Number of entries[9] by department sorted in ascending order:

Mayenne: 0; Eure: 5; Seine-Maritime: 5; Paris: 10; Val-de-Marne: 10; Yvelines: 10; Hauts-de-Seine: 10; Seine-Saint-Denis: 10; Essonne: 21; Eure-et-Loir: 27; Loiret: 27; Seine-et-Marne: 29; Val-d'Oise: 31; Nièvre: 34; Yonne: 36; Côte-d'Or: 36; Ardennes: 40; Marne: 42; Haute-Marne: 43; Indre: 43; Allier: 44; Aube: 44; Cher: 44; Saône-et-Loire: 47; Indre-et-Loire: 50; Oise: 53; Pas-de-Calais: 54; Aisne: 54; Nord: 54; Somme: 55; Pyrénées-Orientales: 60; Loir-et-Cher: 63; Vosges: 64; Meurthe-et-Moselle: 66; Doubs: 68; Haute-Saône: 68; Calvados: 69; Jura: 69; Territoire-de-Belfort: 69; Orne: 69; Manche: 69; Finistère: 71; Côtes-d'Armor: 73; Morbihan: 73; Savoie: 73; Haute-Savoie: 76; Cantal: 79; Puy-de-Dôme: 80; Maine-et-Loire: 81; Haut-Rhin: 83; Sarthe: 83; Bas-Rhin: 83; Lozère: 84; Loire-Atlantique: 87; Ille-et-Vilaine: 88; Aude: 89; Gard: 90; Hérault: 90; Lot: 91; Meuse: 95; Haute-Vienne: 97; Corrèze: 98; Gers: 98; Creuse: 100; Vendée: 102; Dordogne: 103; Lot-et-Garonne: 104; Pyrénées-Atlantiques: 104; Hautes-Pyrénées: 104; Tarn-et-Garonne: 105; Charente-Maritime: 105; Vienne: 105; Landes: 105; Gironde: 106; Ariége: 106; Deux-Sèvres: 106; Charente: 106; Haute-Garonne: 106; Tarn: 107; Aveyron: 107; Ain: 121; Rhône: 129; Isère: 130; Drôme: 130; Ardèche: 131; Loire: 131; Hautes-Alpes: 134; Alpes-Maritimes: 136; Var: 137; Bouches-du-Rhône: 138; Vaucluse: 138; Alpes-de-Haute-Provence: 138; Moselle: 142; Haute-Loire: 168.

[9] entries with 0 included, with NA not

DART: A Large Dataset of Dialectal Arabic Tweets

Israa Alsarsour, Esraa Mohamed, Reem Suwaileh, Tamer Elsayed
Computer Science and Engineering Department, Qatar University, Doha, Qatar
{ia1205702, em1205267, reem.suwaileh, telsayed}@qu.edu.qa

Abstract

In this paper, we present a new large manually-annotated multi-dialect dataset of Arabic tweets that is publicly available. The Dialectal ARabic Tweets (DART) dataset has about 25K tweets that are annotated via crowdsourcing and it is well-balanced over five main groups of Arabic dialects: Egyptian, Maghrebi, Levantine, Gulf, and Iraqi. The paper outlines the pipeline of constructing the dataset from crawling tweets that match a list of dialect phrases to annotating the tweets by the crowd. We also touch some challenges that we face during the process. We evaluate the quality of the dataset from two perspectives: the inter-annotator agreement and the accuracy of the final labels. Results show that both measures were substantially high for the Egyptian, Gulf, and Levantine dialect groups, but lower for the Iraqi and Maghrebi dialects, which indicates the difficulty of identifying those two dialects manually and hence automatically.

Keywords: Arabic, Multi-Dialect, Twitter, Crowdsourcing, Annotations, Corpus

1. Introduction

The Arabic language is the fifth most widely spoken language in the world; more than 380 million people speak and write in Arabic(Darwish et al., 2014). Additionally, approximately 41.7% of Arabic speakers are using the Internet[1] which necessitates the need for developing language-specific tools for Arabic. The Arabic language has many dialects (varieties), besides the Modern standard Arabic (MSA), that are broadly used in daily life(Huang, 2015). Although dialects have common linguistic uses, they greatly differ making Arab people themselves face difficulty in understanding each other. The variations and similarities of Arabic dialects stem from different factors, e.g., social class, education level, religion, gender, and geographical region (Benajiba and Diab, 2010).

Arab users write in dialects over the Internet and extensively in social media. This introduces many challenges to researchers in areas such as Natural Language Programming, Information Retrieval, and Machine Learning, who deal with the spoken and/or written language one way or the other. The advancement in these areas is remarkably restricted by the shortage of high-quality Arabic language resources.

In this paper, we tackle the problem of building a large dialectal Arabic tweets dataset that somewhat remedies the lack of Arabic resources and opens the door of support to tackle various research problems such as dialect detection, words segmentation, translation, cross-dialect search, and speech recognition. Our contribution in this work is two folds:

1. We introduce ***DART*** dataset; a large well-balanced publicly-available[2] **D**ialectal **AR**abic **T**weets dataset that we believe will enable research on different areas.
2. We provide an analysis on the quality of the dataset in terms of inter-annotator agreement (measured in Kappa), and accuracy of final labels (measured by in-house annotators).

The rest of this paper is organized as follows. Section 2 reviews the literature and describes the publicly-available datasets. We layout the process of collecting and annotating DART dataset in Sections 3 and 4. Section 5 provides a comparison between DART and the other similar datasets. Finally, we conclude and discuss possible future directions in Section 6.

2. Related Work

In this section, we review the available dialectal Arabic datasets and discuss their properties.

The Arabic Online news Commentary (AOC) (Zaidan and Callison-Burch, 2011) is the first available dialectal dataset that contains 3.1M comments gathered from Egyptian, Gulf, and Levantine news websites. The authors initially labeled only around 0.05% of the dataset by Amazon's Mechanical Turk (MTurk) crowdsourcing platform [3]. Many researchers used AOC dataset by either extending the annotations (Cotterell and Callison-Burch, 2014) or directly using it for different purposes such as extracting dialectal n-grams to automatically label tweets by their dialect (Mubarak and Darwish, 2014). (Cotterell and Callison-Burch, 2014) extended the AOC dataset to cover Maghrebi (MG) and Iraqi dialects. They also crawled tweets using Twitter API and labeled them using MTurk. The dataset is not balanced across dialect groups. Moreover, it contains lots of noise such as Arabizi and French tweets.

(Bouamor et al., 2014) used an Egyptian-English corpus (Zbib et al., 2012) as seed corpus and asked four in-house Arabic native speakers from Palestine, Syria, Jordan, and Tunisia to translate 2,000 Egyptian sentences into their dialects. A major issue of this dataset is the approach that generated sentences that do not reflect the natural way of writing and speaking in dialects. Additionally, as annotators were selected from few countries, the dataset provides biased labeled data for only two dialectal groups (Levantine and Maghrebi) besides Egyptian group.

Thus far, we discussed datasets that were manually labeled (either by in-house or crowdsourcing annotators). We fur-

[1] http://www.internetworldstats.com/stats19.htm
[2] http://qufaculty.qu.edu.qa/telsayed/datasets/
[3] www.mturk.com

ther discuss datasets that are collected and labeled automatically. (Mubarak and Darwish, 2014) used the geo-location attribute of tweets to automatically label them by their corresponding dialect. (Eldesouki et al., 2017) selected 350 tweets from this corpus for the five-dialectal groups and labeled them manually. Similarly, (Huang, 2015) also used the geographical location of Facebook posts to label them and create a week classifier to detect the dialect of posts. The classifier was trained to detect the five dialects groups. All these datasets are rather small with respect to the current standards. (Salama et al., 2015) also labeled Youtube comments and videos description using their geographic location. They randomly selected 1,000 sentences from each dialectal corpus and asked two native speakers to judge them. Differently, (Almeman and Lee, 2013) proposed an automatic approach to collect dialectal Arabic web pages. They covered only four of the common dialectal groups (they combine Iraqi and Gulf dialects). Their approach has a pipeline of gathering and filtering steps. The major issue with this dataset is that the pages might also contain MSA sentences which are hard to separate.

3. Collecting Dialectal Data

In this section, we describe the pipeline of collecting *DART* dataset. We started by manually collecting popular dialectal phrases for each Arabic dialect group as listed in table 1. After filtering out inappropriate and common phrases (i.e., those used in more than one dialect group), we tracked the unique phrases over Twitter stream. As the stream is flooded by spam and retweets, we cleaned the collected data annotating it. We elaborate thoroughly on each step in the following subsections.

3.1. Collecting Dialectal Phrases

For each dialect[4], we target distinct phrases that are spoken by only the native speakers of that dialect. We collected a list of dialectal phrases from two sources. We first acquired a list of 1,000 dialectal words collected by (Almeman and Lee, 2013). The list covers only four dialects: EGY, GLF, LEV, MGH. To diversify the sources from where we collected the dialectal phrases, we extended the list with phrases from *Mo3jam* website[5], which allows Arab users to contribute with dialectal phrases spoken in their countries. For each dialect group, we randomly selected phrases from the list of phrases of each country under that dialect from that website.

We performed several filtering steps on both lists of dialectal phrases. We first manually dropped inappropriate phrases. We then issued each phrase against Twitter Live search interface [6] and excluded any phrase that returns inappropriate or no results. We also filtered out phrases that returned tweets in different dialects. We ended up with 232 phrases on average for each dialect: 278 for EGY, 246 for GLF, 244 for LEV, 121 for IRQ, and 273 for MGH. We share all of the final lists of phrases in our released dataset.

[4]We will use "dialect" to denote "dialect group" from now on.

[5]ar.mo3jam.com/

[6]twitter.com/search-home

3.2. Tracking Tweets

To construct a potential dialectal dataset, we tracked the list of phrases using Twitter streaming API for tracking[7]. The tracking period spanned about two months sporadicly (from 25 of February to 5 of May 2017). Table 1 shows (in the third column) examples of tracked tweets and their corresponding tracked phrases (in bold face). We also report the number of tweets crawled for each dialect group in the fourth column of the table.

3.3. Cleaning Tweets

Although we crawled the potential dialectal dataset via tracking dialectal phrases, the dataset might still contain lots of noise such as multilingual tweets (i.e., tweets written in other languages besides Arabic), inappropriate tweets, etc. Therefore, to have a better-quality potential dataset for labeling, we cleaned the dataset as follows:

- *Filtering out non-Arabic tweets*: Many Arab users post tweets written in multiple languages. For example, users from North Africa tend to write in the French language besides Arabic. Moreover, many other Arab users prefer to communicate in Arabizi [8] (Arabic phrases written in English alphabet) on Twitter. Although foreign words and phrases might be good indicator of the dialect of the text, we opted to drop the tweets that are mostly written in non-Arabic language.
- *Filtering out inappropriate tweets*: Having a list of around 300 manually-collected Arabic inappropriate phrases, we cleaned the dataset by dropping tweets that contain any phrase from that list.
- *Filtering out short tweets*: To avoid ambiguity in very short tweets and hence difficulty and confusion in annotations, we also eliminated tweets that have less than three words.
- *Filtering out duplicates*: To save time of annotating duplicate tweets, we also excluded the retweets.

Applying the above cleaning steps, we finally obtained a cleaned version of the potential dataset that contains around 145K tweets, as shown in the right-most column of table 1.

4. Annotations via Crowdsourcing

Our objective is to create a good quality dialectal dataset that is of lasting value to researchers interested in working on Arabic dialects. Having a clean Arabic tweets dataset with pseudo-labels (i.e., tweets labeled by the dialect of the corresponding tracking phrase), the next step is to annotate the tweets in a more reliable way.

To accurately and reliably annotate the tweets with their corresponding dialects, we need multiple Arabic native speakers for each dialect. That is indeed challenging to find in a surrounding community. More importantly, the dataset size we want to annotate is quite large, hence it is too expensive to recruit in-house annotators for that task. Therefore, we used CrowdFlower crowdsourcing platform[9] to acquire

[7]developer.twitter.com/en/docs/tutorials/consuming-streaming-data#track

[8]en.wikipedia.org/wiki/Arabic_chat_alphabet

[9]www.crowdflower.com

Dialect	Countries	Example Tweet	Collected	Clean
EGY	Egypt	انا **عاوزه** اعدي التيرم ده يا عاطف	89,424	37,834
GLF	UAE, KSA, Bahrain, Kuwait, Qatar	ودي اشوفكم **دحين** تعالو بسرعة	89,709	35,137
LEV	Palestine, Jordan, Syria, Lebanon	فوتو احكوا شو **بدكن**	75,549	23,039
IRQ	Iraq	**شكد** حلوين شلون كاعدين	55,464	23,236
MGH	Morocco, Algeria, Tunisia	**واش بيك** واش راك تحكي مافهمتش عليك	36,991	16,350
Total	-	-	347,137	144,596

Table 1: Dialect groups and corresponding collected/cleaned dialectal data.

Dialect	Test Qs	Dial. Tweets (%)	Kappa	Acc.
EGY	338	5,265 (75%)	0.71	97%
GLF	340	5,893 (84%)	0.71	100%
LEV	347	3,939 (55%)	0.62	96%
IRQ	234	5,253 (75%)	0.42	78%
MGH	398	3,930 (55%)	0.28	88%
Total	1,657	24,280 (69%)	-	92%

Table 2: Different statistics about DART.

annotations. In the following subsections, we describe our annotation process.

4.1. Task Design

For each dialectal group, we randomly selected about 7K tweets from the potential dialect dataset to be annotated by contributors on CrowdFlower. In order to increase both accuracy and reliability, we designed one job (with relatively simple instructions) for each dialectal group, aiming at finding native speakers for that group. For each job, we allow only contributors from the corresponding countries of the dialect group to work on the job. For example, for Gulf dialect job, only participants from Bahrain, Oman, Kuwait, Qatar, Saudi Arabia, and United Arab Emirates are eligible to work. We designed the annotation task to show contributors ten tweets per page, and asked them to label each tweet by either the corresponding dialect (indicated by the pseudo-label of the tweet), MSA, or other (in case the tweet is written in other dialect or annotators could not identify its dialect).

4.2. Quality Control

We adopted a common quality control method to ensure high-quality labels. We sought five native speakers, one for each dialectal group, to label around 300 to 400 tweets (1,657 tweets in total) and use those as the source of quiz/test questions in our crowdsourcing jobs. We required the contributors to attain at least level 2 (moderate) according to CrowdFlower rating. We randomly selected 10 tweets from that set as quiz questions to examine the contributors before they start the job. A minimum accuracy of 90% was required to pass the quiz.

We also used the full set of labeled tweets as "gold" questions to ensure a consistent performance of the contributors throughout the job. When contributors accuracy fell under the predefined accuracy level, they were excluded. Finally, we collected 3 annotations for each tweet from different contributors to increase confidence in labeling.

4.3. Pilot Studies

Before we launch the actual jobs, we conducted several pilot studies for each dialectal group separately (using 100 tweets for each). We aimed at estimating the required budget (e.g., cost and time) and improving the instructions, design, and setup of the jobs. We list here the major challenges we encountered while we ran these small-scale studies.

- Inaccessible Tweets: We used Twitter widget to display the tweets on the task interface using the tweet ID. However, when tweets are deleted or the author make his profile private, the annotators are no longer able to label them. This is critical especially if the tweets are used as gold questions. In such case, the annotator would arbitrarily choose a label, which in turn affects their performance. To resolve this issue, we periodically checked the test questions during the period when the tasks were running and removed the inaccessible ones.

- Lack of country-specific contributors: Some Arab countries do not have contributors on CrowdFlower. This was evident specifically for the Iraqi dialect, hence, we opted to disable the geographical constraint on the contributors of the Iraqi job. As this decision has the potential to affect the quality of the labels, we limited the contributors to be only from the Gulf countries for that job, as they are the closest group to the Iraqi dialect.

4.4. Aggregation and Agreement

To aggregate the multiple labels per tweet, we opted to use majority voting which requires at least two annotators to agree on the label. This resulted in 24,280 dialectal tweets. To asses the reliability of the agreement, we measured the inter-annotator agreement using Fleiss Kappa (Fleiss, 1971). Fleiss Kappa is used when more than two annotators labeled a data item (a tweet in our case) to measure the degree of agreement over what would be expected by chance. We found the degree of agreements substantial for three dialects EGY, GLF, and LEV, moderate for IRQ, and fair for MGH. We show the exact kappa values in Table 2.

4.5. Accuracy

To evaluate the accuracy of the crowdsourcing labels, we randomly selected 100 tweets per dialectal group and asked

Dataset	Source	Size	Dial. Groups	Labels	Public?
(Cotterell and Callison-Burch, 2014)	Twitter and AOC	67,468	5G	Manual	✓
(Zaidan and Callison-Burch, 2011)	News Comments	44,618	5G-{IRQ, MGH}	Manual	✓
DART	Twitter	24,280	5G	Manual	✓
(Bouamor et al., 2014)	Egy-Eng Corpus	5,000	5G-{GLF, IRQ}	Manual	✓
(Eldesouki et al., 2017)	Twitter	1,400	5G-{IRQ}	Manual	✓
(Huang, 2015)	FaceBook	66M	5G	Auto	✗
(Mubarak and Darwish, 2014)	Twitter	6.5M	5G	Auto	✗
(Almeman and Lee, 2013)	Web Corpus	2M	5G-{IRQ}	Auto	✓
(Salama et al., 2015)	YouTube	640,817	5G	Auto	✗

Table 3: A comparison between DART and datasets used in literature.

one native speaker from each group to *re-label* the corresponding tweets. The last column in Table 2 shows the accuracy for each group. It indicates that accuracy for GLF, EGY, and LEV is high (ranges between 100% and 96%), a little lower for MGH (88%), and much lower for IRQ (78%). This is somewhat aligned with the inter-annotator agreement values. In fact, both indicate that manually-identifying Iraqi and Maghrebi tweets is very challenging, which in turn hints about the difficulty dialect identification systems would face in identifying them too.

5. DART among Others

Table 3 illustrates a comparison between DART and the Arabic dialectal datasets used in the literature. For each dataset, the table indicates the data source, the size of the dataset (in sentences or tweets), the dialectal groups covered (5G denotes the five groups we covered), the type of annotations (manually or automatically), and whether it is publicly available or not. While the table shows that DART is the third largest dataset among the manually-annotated ones, the largest two have balancing and coverage limitations that make them less usable. The first is not well balanced over the five groups as it has too many GLF (63%) but very few IRQ ($<$1%) and MGH (10%) sentences or tweets, while the second covers only three dialectal groups. On the contrary, DART is well-balanced over the five groups it covers. Moreover, DART is exclusively composed of tweets, which makes it more homogeneous and thus suitable for training Twitter-specific dialect identification systems.

6. Conclusion and Future Work

We introduced DART, a large multi-dialect dataset of Arabic tweets that is publicly-available. The dataset is composed of about 25k labeled tweets and is well balanced over five common dialect groups. DART is constructed over a well-planned pipeline and was annotated via crowdsourcing. Measures of inter-annotator agreement as well as accuracy of final labels showed high quality and hence high potential of utilizing the dataset as a rich resource for the community.

DART opens several possible future research directions. It can be used as a gold-standard for training and evaluating Arabic dialect detection systems. It can also be extended to a more fine-grained level of annotations per country. It can even enable further studies on the differences and commonalities between Arabic dialects.

7. Acknowledgments

This work was made possible by NPRP grant# NPRP 7-1313-1-245 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors. The work was also supported by grant QUST-CENG-SPR-2017-21 from College of Engineering at Qatar University.

8. Bibliographical References

Al-Mannai, K., Sajjad, H., Khader, A., Al Obaidli, F., Nakov, P., and Vogel, S. (2014). Unsupervised word segmentation improves dialectal Arabic to English machine translation. *ANLP 2014*, page 207.

Almeman, K. and Lee, M. (2013). Automatic building of Arabic multi dialect text corpora by bootstrapping dialect words. In *Communications, signal Processing, and their Applications (ICCSPA), 2013 1st international conference on*, pages 1–6. IEEE.

Benajiba, Y. and Diab, M. (2010). A Web application for dialectal arabic text annotation. In *Proceedings of the LREC workshop for language resources (LRS) and human language technologies (HLT) for semitic languages: Status, updates, and prospects*.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A multidialectal parallel corpus of Arabic. In *LREC*, pages 1240–1245.

Cotterell, R. and Callison-Burch, C. (2014). A multi-dialect, multi-genre corpus of informal written Arabic. In *LREC*, pages 241–245.

Darwish, K., Sajjad, H., and Mubarak, H. (2014). Verifiably effective Arabic dialect identification. In *EMNLP*, pages 1465–1468.

Eldesouki, M., Samih, Y., Abdelali, A., Attia, M., Mubarak, H., Darwish, K., and Laura, K. (2017). Arabic multi-dialect segmentation: bi-lstm-crf vs. svm. *arXiv preprint arXiv:1708.05891*.

Elfardy, H. and Diab, M. T. (2013). Sentence level dialect identification in Arabic. In *ACL (2)*, pages 456–461.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Huang, F. (2015). Improved Arabic dialect classification with social media data. In *EMNLP*, pages 2118–2126.

Mubarak, H. and Darwish, K. (2014). Using Twitter to collect a multi-dialectal corpus of arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 1–7.

Salama, A., Bouamor, H., Mohit, B., and Oflazer, K. (2015). Youdacc: the youtube dialectal Arabic commentary corpus.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic online commentary dataset: an annotated dataset of informal arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41. Association for Computational Linguistics.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O. F., and Callison-Burch, C. (2012). Machine translation of arabic dialects. In *Proceedings of the 2012 conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 49–59. Association for Computational Linguistics.

Classification of Closely Related Sub-dialects of Arabic Using Support-Vector Machines

Samantha Wray
New York University Abu Dhabi
United Arab Emirates
samantha.wray@nyu.edu

Abstract

Colloquial dialects of Arabic can be roughly categorized into five groups based on relatedness and geographic location (Egyptian, North African/Maghrebi, Gulf, Iraqi, and Levantine), but given that all dialects utilize much of the same writing system and share overlapping features and vocabulary, dialect identification and text classification is no trivial task. Furthermore, text classification by dialect is often performed at a coarse-grained level into these five groups or a subset thereof, and there is little work on sub-dialectal classification. The current study utilizes an n-gram based SVM to classify on a fine-grained sub-dialectal level, and compares it to methods used in dialect classification such as vocabulary pruning of shared items across dialects. A test case of the dialect Levantine is presented here, and results of 65% accuracy on a four-way classification experiment to sub-dialects of Levantine (Jordanian, Lebanese, Palestinian and Syrian) are presented and discussed. This paper also examines the possibility of leveraging existing mixed-dialectal resources to determine their sub-dialectal makeup by automatic classification.

Keywords: text classification, validation of language resources, language identification

1. Introduction

Arabic is a language rich in dialectal variety, with Modern Standard Arabic (MSA) used for official purposes alongside a colloquial dialect local to a speaker's given region. Although these dialects are typically written throughout informal contexts such as social media, they are nonstandardized and exhibit spontaneous spelling when compared to the standardized variety.

In the text domain, previous work on dialect identification has been performed for purposes ranging from training automatic speech recognition to bootstrapping corpus building efforts. For some of these studies, the focus is on a binary classification between MSA and a colloquial dialect, such as the Egyptian-MSA classification performed by Tillmann et al. (2014), Mansour et al. (2014) and Elfardy and Diab (2013).

A more fine-grained classification is performed by (Zaidan and Callison-Burch, 2011), as they crowdsource the human classification of 100,000 sentences of dialectal Arabic, and train language models to classify additional dialectal material scraped from online newspaper comment sections. The classification categories for this study were MSA, Levantine, Egyptian, and Gulf. This 4-way classification achieves an accuracy of 69.4% (Zaidan and Callison-Burch, 2011). However, their accuracy when classifying between just the non-standard dialects Levantine, Egyptian, and Gulf jumps to 83.5%.

Typically, dialect classification is performed into these coarse-grained categories. However, work by Malmasi et al. (2015) attempted classification with a finer-grained distinction. They classify text into seven categories: MSA, Tunisian, Egyptian, Jordanian, Syrian, and Palestinian, achieving a 74.3% accuracy for this 6-way classification.

However, the training set used to construct the language models was based on materials created by human translation from one dialect to another. This parallel corpus is composed of the above 5 non-standard dialects in addition to MSA from (Bouamor et al., 2014), but the corpus is not spontaneous and was created under prompts in such a way that it is not entirely parallel. For example, when speakers who contributed to the corpus were prompted to translate a base phrase into their native dialects, some speakers inserted additional material (such as /maʃa al:a/ 'wow!') while others did not, despite these phrases being pan-Arabic and having no likely preponderance to occur more frequently in one dialect over another. It's likely then that these phrases emerge in the corpus as an artifact of a particular contributor's inclination to use them at that time. Finally, Arabic dialect identification has been the subject of shared tasks focusing on discriminating between similar languages (Malmasi et al., 2016), again, at a coarse-grained level between MSA and the colloquial dialects North African, Egyptian, Levantine, and Gulf. The methodology that has emerged from this line of research is the use of *support vector machines* (SVM) utilizing character n-grams. Given the relatively small size of training data utilized in these tasks, SVM has outperformed other methods such as deep neural networks (Malmasi et al., 2016).

2. Materials

To focus on training and testing materials that were annotated at the sub-dialectal level and produced spontaneously, I utilized 10,000 tweets harvested from each of the four countries that make up the speakers of the Levantine dialect (Jordan, Palestine, Syria, and Lebanon). These 10,000 tweets contained a total of approximately 100,000 words.

I drew partially from annotated material from (Mubarak and Darwish, 2014) which contains Twitter data from Jordan, Palestine, Syria, and Lebanon and was collected by exploiting user-provided location and geo-tagging info. I supplemented this with additional tweets from each country, which were harvested in the same manner throughout 2014-2015.

There are approximately 15 million tweets produced in Arabic per day (Mubarak and Darwish, 2014). Each tweet produced has associated metadata that includes the time of publication, the username of the author, the language the tweet is written in, as well as location information of where the tweet was written from. This metadata is searchable and made available for harvesting via the Twitter API[1]. Given that all colloquial dialects of Arabic utilize the same script for encoding language, and because the writing of colloquial language is not a phenomenon unique to any given dialect, the approach used in previous studies pioneering tweet harvesting such as (Ljubešic et al., 2014) will not provide fine-grained dialect information given that the metadata included for Twitter does not include Arabic dialect info. A selection of 'Arabic' would produce all colloquial dialects, as well as material written in MSA.

As for location information, it comes in two forms. The first is a raw latitude longitude geo-tag which can give precise indication to region. However, this is an optional feature most users do not use: only about 2% of tweets are geo-tagged for location (Huck et al., 2012). Another optional feature is a user-provided location, which approximately 70% of tweets include (Mubarak and Darwish, 2014). These are locations which the user has custom written, and may or may not be informative as to actual location. For example, a user can write they are located in 'Amman, Jordan' or simply write 'My house'. I used the Twitter API and engaged with it via the Twython library[2] for Python. Using these tools, I requested the Twitter API search through all tweets in the Arabic language which met either of the following criteria:

- Geo-tag: tweet was tagged within the geographic borders of the regions of interest
- User location: tweet was written by user with location matching major cities in Levantine countries, as well as the country names. Following Mubarak and Darwish (2014), possible user location names associated with each country were drawn from the geographic name database GeoNames[3]. In addition, each time a user from a specific country was identified, manual inspection of user locations of their followers was performed to collect new permutations of possible ways to indicate residence in that country

This approach operated under a couple of assumptions. First, that users tweet from a location where speakers speak the same dialect as the user. Given patterns of immigration in the Levant, users who produce geo-coded tweets in a particular country may have origins, and therefore native dialects, associated with a different region. Furthermore, a user may be temporarily producing tweets with a different geo location other than their default location if they are traveling. To partially circumvent this potential confound, tweets which exhibited mismatch between the user-provided location and the geocode were not retained in the dataset. Furthermore, a potential confound arises from the fact that a particular author's features may be easily identified across testing and training sets (Rangel et al., 2017). However, the data has been completely anonymized and therefore further analysis regarding the influence of authorship is unavailable.

Tweets were collected for the greater part of a year to help reduce the possibility of cyclic effects of frequency, such as the tendency for holiday greetings to spike during Ramadan, for example (Eisenstein, 2013; Refaee and Rieser, 2014). To help maximize the likelihood that the collected tweets contained colloquial material, I discarded tweets which contained vowel diacritics, as this was typically indicative of a user tweeting a verse from the Quran, and not likely to contain colloquial data.

Data was prepared for study by removing additional non-textual information included in the tweets such as emojis, other usernames (mentions), punctuation, and links.

In addition to spontaneously-produced tweets, two additional language resources were utilized: two different spoken Arabic telephone corpora in which the speaker's country of origin is annotated (Appen, 2007; Maamouri et al., 2007). The transcribed speech corpora differed from the tweets as they were professionally transcribed, thus losing any character-level features that may emerge as a result of being written by the speaker. The telephone transcripts were used to generate words unique to each dialect; that is, words which appeared in telephone conversations in one dialect but not in the other three. These vocabularies were thus pruned to remove overlapping dialectal items.

3. Classification Experiments

Following recent work on discriminating between dialects of Arabic for transcriptions of speech (Malmasi et al., 2016), a state-of-art method of Support Vector Machines including features of character n-grams (from unigrams to 5-grams) was explored (Eldesouki et al., 2016). In addition to this, word n-grams (from unigrams to trigrams) were included in the models.

In addition to the inclusion of n-gram based feature models, models which focused on unique words for each dialect were included to prune vocabulary that overlaps across dialects. This method has been demonstrated to be successful in large-scale text identification and classification tasks (Madsen et al., 2004, among others), including discrimination tasks for other closely-related languages with low resources and a large lexical overlap, such as Tagalog, Cebuano, and Bicolano (Dimalen and Roxas, 2007).

3.1. Procedure

A multiclass linear kernel Support Vector Machine (Joachims, 1999) was used to perform a four-way classification into the sub-dialects Jordanian, Palestinian, Lebanese, and Syrian, on the Twitter data described above. The model included features for character unigrams up to 5-grams, and word unigrams up to trigrams. Furthermore, a model incorporating features of the pruned vocabulary lists built from the telephone corpora were explored. Classification was performed on each tweet individually.

To generate n-gram probabilities used as features for the SVM, I used the SRI Language Modeling (SRILM) toolkit

[1] https://dev.twitter.com/

[2] https://github.com/ryanmcgrath/twython

[3] geonames.org

(Stolcke and others, 2002) encompassing the following:

- *Word n-grams:* Modeled the probability of occurrences of contiguous word strings in a text. For the current study, three word n-gram models were built: a unigram model determining the likelihood of a word occurring in a text, a bigram model determining the likelihood of a word occurring given the previous word, and a trigram model determining the likelihood of a word occurring given the previous two words. Unknown words encountered by the model were mapped to <unk>, and sentence start and end markers were considered in the models.
- *Character n-grams:* Modeled the probability of occurrences of contiguous character strings in a text. Five character n-gram models were built: a unigram model determining the likelihood of a character appearing in a text, a bigram model determining the likelihood of a character occurring given a previous character, and so on, until the 5-gram model.[4]

An additional feature type considered was the inclusion of dialect-specific words generated by the unique words list from the telephone corpora. These features were binary: either the word was present, or not.

3.2. Results

I used k-fold cross-validation (k = 10), in which a sampling of 9/10 of the data was used as the training data with the remaining 1/10 being used as testing data, and this process was performed 10 times. The accuracy reported in 1 is calculated as an average across all 10 folds.

Method	Accuracy
N-gram model:	
1-5 character; 1-3 word	**65%**
Vocabulary pruning	54%
N-gram + vocabulary pruning	46%

Table 1: Classification accuracy for training and testing sets. Best results are in bold.

Table 1, shows the performance of three models: (i) The n-gram based SVM trained on the Twitter data which did not incorporate unique words from the telephone corpora, (ii) SVM trained on only the unique words per dialect from the telephone corpora and (iii) a combination SVM using both of the above feature sets. The first model, the n-gram based SVM trained on the Twitter data, had the highest accuracy at 65%.

4. Classification Estimates for Existing Resources

Given the SVM n-gram model presented in the previous section achieves an accuracy of 65% when classifying tweets, it was utilized to perform further classification on existing Levantine resources in order to provide estimates on their sub-dialectal makeup. I used the model to classify the mixed Levantine corpora from Almeman and Lee (2013) and Zaidan and Callison-Burch (2011) and performed classification on each sentence in the corpus. These estimates are shown in Table 2.

Existing mixed-dialect resource	Classification
(Zaidan and Callison-Burch, 2011)	JOR 42%
	LEB 19%
	PAL 9%
	SYR 30%
(Almeman and Lee, 2013)	JOR 70%
	LEB 9%
	PAL 17%
	SYR 4%

Table 2: Classification estimates for existing mixed-dialect resources using the best performing model (n-gram SVM)

As shown in Table 2, a majority of the data was classified as Jordanian. The high estimates for Jordanian are not surprising, especially for classification of (Zaidan and Callison-Burch, 2011) given that the resource is composed of newspaper commentary and the newspaper selected to represent Levantine for that multidialectal dataset was indeed a Jordanian newspaper.

5. Conclusions and Future Work

Colloquial/Spoken Arabic dialect identification and discrimination in text is a nontrivial task due to several factors, including the fact that the dialects are closely related and thus exhibit overlapping lexicons and a shared writing system. For fine-grained distinctions at the sub-dialectal level, this problem is exacerbated by the lack of data annotated by sub-dialect and a lack of previous research into discrimination at the sub-dialectal level. In this paper, the state-of-the-art methodology for discriminating between colloquial/spoken dialects of Arabic in text (n-gram based SVMs) (Malmasi et al., 2016) was applied at a more fine-grained level. I also explored utilizing this methodology to leverage existing coarse-grained mixed dialectal resources with the intention of repurposing them as fine-grained resources with classification at the sub-dialectal level. Further improvements are currently being explored, including the utilization of tf-idf for features. Furthermore, vocabulary pruning from different sources than the telephone corpora should be explored, given that the telephone corpora are quite sparse at only approximately 200,000 words. For example, Darwish et al. (2014) demonstrated success with dialect-specific words in text-based classification, specifically, manually-identified Egyptian words at the high end of the frequency range from (Zbib, Rabih and Malchiodi, Erika and Devlin, Jacob and Stallard, David and Matsoukas, Spyros and Schwartz, Richard and Makhoul, John and Zaidan, Omar F and Callison-Burch, Chris, 2012). The current study focuses on a test case of the Arabic dialect Levantine, which is broadly made up of four geopolitical entities that can serve as a more fine-grained classification (Jordanian, Lebanese, Palestinian, and Syrian.)

[4]Models were also run utilizing a binary system in which presence or absence of an n-gram was indicated. Results did not differ with regard to predictive performance, but did benefit a slight time cost and would be therefore lighter weight to deploy if scaled up.

However, this methodology and improvements upon it can be easily adapted to other closely-related dialects, such as those spoken throughout the Arab Gulf, and to other non-geopolitical dialectal distinctions such as the urban vs. rural varieties.

6. Acknowledgements

Preliminary work on this research appeared as part of a doctoral dissertation (Wray, 2016). Funding for this research is gratefully acknowledged from the National Science Foundation under BCS-1533780, and the University of Arizona Graduate and Professional Student Council. An allocation of computer time is also gratefully acknowledged from the UofA Research Computing HPC and HTC, as well as HPC at New York University Abu Dhabi.

7. Bibliographical References

Almeman, K. and Lee, M. (2013). Automatic building of Arabic Multi Dialect text corpora by bootstrapping dialect words. In *Communications, Signal Processing, and their Applications (ICCSPA), 2013 1st International Conference on*, pages 1–6. IEEE.

Darwish, K., Sajjad, H., and Mubarak, H. (2014). Verifiably Effective Arabic Dialect Identification. *EMNLP-2014*.

Dimalen, D. M. and Roxas, R. E. O. (2007). AutoCor: A Query Based Automatic Acquisition of Corpora of Closely-related Languages. In *PACLIC*.

Eisenstein, J. (2013). What to do about bad language on the internet. In *HLT-NAACL*, pages 359–369.

Eldesouki, M., Dalvi, F., Sajjad, H., and Darwish, K. (2016). QCRI@ DSL 2016: Spoken Arabic Dialect Identification Using Textual. *VarDial 3*, page 221.

Elfardy, H. and Diab, M. T. (2013). Sentence Level Dialect Identification in Arabic. In *ACL (2)*, pages 456–461.

Huck, J., Whyatt, D., and Coulton, P. (2012). Challenges in geocoding socially-generated data.

Joachims, T. (1999). Svmlight: Support vector machine. *SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund*, 19(4).

Ljubešic, N., Fišer, D., and Erjavec, T. (2014). TweetCaT: a Tool for Building Twitter Corpora of Smaller Languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland. European Language Resources Association (ELRA)*.

Madsen, R. E., Sigurdsson, S., Hansen, L. K., and Larsen, J. (2004). Pruning the vocabulary for better context recognition. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 483–488. IEEE.

Malmasi, S., Refaee, E., and Dras, M. (2015). Arabic dialect identification using a parallel multidialectal corpus. In *Proceedings of the 14th Conference of the Pacific Association for Computational Linguistics (PACLING 2015), Bali, Indonesia*, pages 209–217.

Malmasi, S., Zampieri, M., Ljubešic, N., Nakov, P., Ali, A., and Tiedemann, J. (2016). Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task. *VarDial 3*, page 1.

Mansour, S., Al-Onaizan, Y., Blackwood, G., and Tillmann, C. (2014). Automatic Dialect Classification for Statistical Machine Translation. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA), Vancouver, BC, Canada*.

Mubarak, H. and Darwish, K. (2014). Using Twitter to collect a multi-dialectal corpus of Arabic. *ANLP 2014*.

Rangel, F., Rosso, P., Potthast, M., and Stein, B. (2017). Overview of the 5th author profiling task at pan 2017: Gender and language variety identification in twitter. *Working Notes Papers of the CLEF*.

Refaee, E. and Rieser, V. (2014). Subjectivity and sentiment analysis of Arabic twitter feeds with limited resources. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools*.

Stolcke, A. et al. (2002). SRILM - An extensible language modeling toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, volume 2002, pages 901–904.

Tillmann, C., Al-Onaizan, Y., and Mansour, S. (2014). Improved sentence-level Arabic dialect classification. In *Proceedings of the VarDial Workshop*, pages 110–119.

Wray, S. (2016). Decomposability and the effects of morpheme frequency in lexical access. *University of Arizona*.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic Online Commentary Dataset: an annotated dataset of informal Arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

8. Language Resource References

Almeman, K. and Lee, M. (2013). Automatic building of Arabic Multi Dialect text corpora by bootstrapping dialect words. In *Communications, Signal Processing, and their Applications (ICCSPA), 2013 1st International Conference on*, pages 1–6. IEEE.

Appen. (2007). *Levantine Arabic Conversational Telephone Speech LDC2007T01*. Linguistics Data Consortium.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A Multidialectal Parallel Corpus of Arabic. In *LREC*, pages 1240–1245.

Maamouri, M., Buckwalter, T., Graff, D., and Jin, H. (2007). *Fisher Levantine Arabic Conversational Telephone Speech LDC2007S02*. LDC.

Mubarak, H. and Darwish, K. (2014). Using Twitter to collect a multi-dialectal corpus of Arabic. *ANLP 2014*.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic Online Commentary Dataset: an annotated dataset of informal Arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Zbib, Rabih and Malchiodi, Erika and Devlin, Jacob and Stallard, David and Matsoukas, Spyros and Schwartz, Richard and Makhoul, John and Zaidan, Omar F and Callison-Burch, Chris. (2012). *Machine translation of Arabic dialects*.

Page Stream Segmentation with Convolutional Neural Nets Combining Textual and Visual Features

Gregor Wiedemann, Gerhard Heyer
Department of Computer Science
Leipzig University, Germany
gregor.wiedemann@uni-leipzig.de, heyer@informatik.uni-leipzig.de

Abstract

In recent years, (retro-)digitizing paper-based files became a major undertaking for private and public archives as well as an important task in electronic mailroom applications. As a first step, the workflow involves scanning and Optical Character Recognition (OCR) of documents. Preservation of document contexts of single page scans is a major requirement in this context. To facilitate workflows involving very large amounts of paper scans, page stream segmentation (PSS) is the task to automatically separate a stream of scanned images into multi-page documents. In a digitization project together with a German federal archive, we developed a novel approach based on convolutional neural networks (CNN) combining image and text features to achieve optimal document separation results. Evaluation shows that our PSS architecture achieves an accuracy up to 93 % which can be regarded as a new state-of-the-art for this task.

Keywords: page stream segmentation, convolutional neural nets, document image classification, document management, text classification

1. Introduction

For digitization of incoming mails in business contexts as well as for retro-digitizing archives, batch scanning of documents can be a major simplification of the processing workflow. In this scenario, scanned images of multi-page documents arrive at a document management system as an ordered stream of single pages lacking information on document boundaries. Page stream segmentation (PSS) then is the task of dividing the continuous document stream into sequences of pages that represent single physical documents.[1]

Applying a fully automated approach of document page segmentation can be favorable over manually separating and scanning documents, especially in contexts of very large data sets which need to be separated (Gallo et al., 2016). In a joint research project together with a German research archive, we supported the task of retro-digitization of a paper archive consisting of circa one million pages put on file between 1922 and 2010 (Isemann et al., 2014). The collection contains documents of varying content, types and lengths around the topic of ultimate disposal of nuclear waste, mostly administrative letter correspondence and research reports, but also stock lists, meeting minutes and email printouts. The 1M pages were archived in roughly 20.000 binders which were batch-scanned due to limited manual capacities for separating individual documents. The long time range of archived material affects document quality, proliferation of layout standards, different fonts and the use of hand-written texts. All these circumstances pose severe challenges to OCR as well as to page stream segmentation (PSS).

In this article, we introduce our approach to PSS comparing (linear) support vector machines (SVM) and convolutional neural networks (CNN). For the first time for this task, we combine textual and visual features into one network to achieve most-accurate results. The upcoming section 2. elaborates on related work. In section 3. we describe our dataset together with one reference dataset for this task. In section 4. we introduce our neural network based architecture for PSS. As a baseline, we introduce an SVM-based model solely operating on text features. Then, we introduce CNN for PSS on text and image data separately as well as in a combined architecture. Section 5. presents a quantitative and a qualitative evaluation of the approach on the two datasets.

[1] The task is also referred to as Document Flow Segmentation or Document Separation.

2. Related work

Page stream segmentation is related to a series of other tasks concerned with digital document management workflows. Table 1 summarizes important characteristics of recent works in this field. A common task related to PSS is document image classification (DIC) in which typically visual features (pixels) are utilized to classify scanned document representations into categories such as “invoice”, “letter”, “certificate” etc. Category systems can become quite large and complex. (Gordo et al., 2013) summarize different approaches in a survey article on PSS and DIC.

In (Gallo et al., 2016), PSS is performed on top of the results from a DIC process. Pages from the stream are segmented each time the DIC system detects a change of class labels between consecutive page images. This approach can only be successful in case there are alternating types of documents in the sequential stream. Often, this cannot be guaranteed, especially in case of small document category systems.

Alternative approaches seek to identify document boundaries explicitly. Such approaches are proposed in (Daher and Abdel, 2014) and (Agin et al., 2015) where each individual image of the sequence is classified as either continuity of the same document (SD) or beginning of a new document (ND). For this binary classification, (Daher and Abdel, 2014) rely on textual features extracted from OCR-results and classify pages with SVM and multi-layer per-

ceptrons (MLP). (Agin et al., 2015) employ bag of visual words (BoVW) and font information obtained from OCR as features, and test performance with three binary classifiers (SVM, Random Forest, and MLP).

The recent state-of-the-art for DIC is achieved by (Gallo et al., 2016), (Harley et al., 2015) and (Noce et al., 2016) who employ Deep Learning with Convolutional Neural Networks to identify document classes. While the former two employ only visual features, the latter study uses both, visual and text features for DIC. For this, class-specific key terms are extracted from the OCR-ed training documents and highlighted with correspondingly colored boxes in the document images. Then, a CNN is applied to learn document classes from these images augmented with textual information highlighting.

Although with (Gallo et al., 2016) there is already one study employing neural network technology not only for DIC but also for PSS, their approach was not applicable to our project for two reasons. First, as mentioned earlier, they perform PSS only indirectly based on changing class labels of consecutive pages. Since we only have 17 document categories and a majority of them belong to one category (”letter”), we need to perform direct separation of the page stream by classifying each page into either SD or ND. Second, quality and layout of our data is extremely heterogeneous due to the long time period of document creation. We expect a lowered performance by solely relying on visual features for separation. Therefore, taking the previous work of (Gallo et al., 2016) as a starting point, we propose our approach for direct PSS as a binary classification task combining textual features and visual features using deep neural networks. We compare this architecture against a baseline comprising an SVM classifier solely relying on textual features.

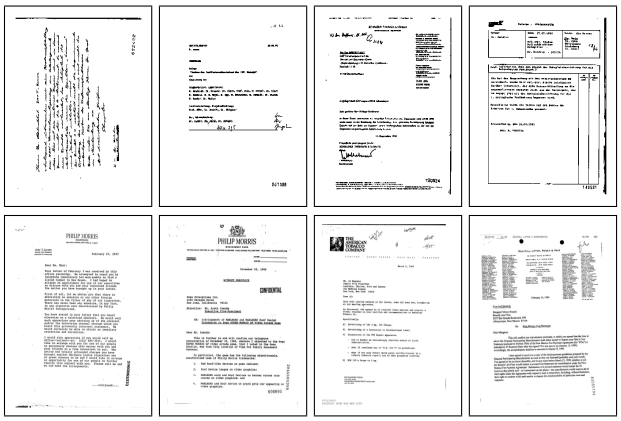


Figure 1: Examples for first pages (class *new document*); from Archive22k (above) and Tobacco800 (below).

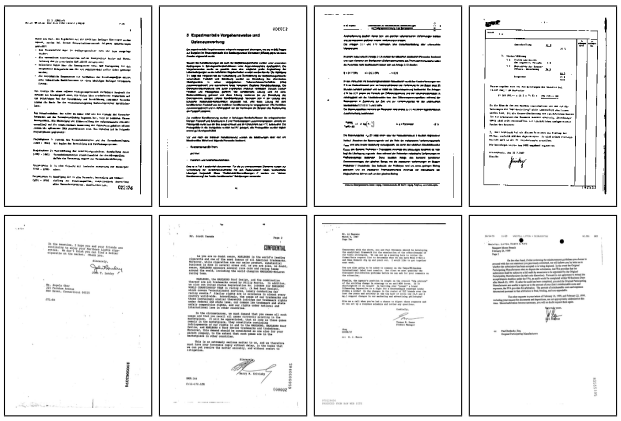


Figure 2: Examples for subsequent pages (class *same document*); from Archive22k (above) and Tobacco800 (below).

3. Datasets

We evaluate our approach on two datasets, one sample from the German archive data of our project context, and one public resource of annotated document scans from U.S. tobacco companies.

3.1. German archive data

The German dataset consists of a variety of document classes from a very long time frame. Most of the documents were archived between the mid-1960s and 2010. Due to this, OCR-quality, document lengths, layout standards as well as used fonts differ widely.

After batch scanning, about 40 % of all binders from the German research archive have been manually separated into documents and annotated with document categories. The manually separated documents can serve as a ground truth for our experiments on model selection and feature engineering for automatic page stream segmentation. For these experiments, we randomly selected 100 binders from the set of all manually separated binders. The binders represent 100 ordered streams of scanned pages, in total consisting of 22,741 pages. 80 of the selected binders containing 17,376 pages were taken as a training set, 20 binders with 5095 pages were taken as test set. Scanned pages were resampled to a size of 224 $\times$ 224 pixels and color-converted to black and white with the OTSU binarization method (Otsu, 1979). The upper lines in Fig. 1 and 2 show examples of first pages, resp. subsequent pages from documents.

From original document scans, text information was extracted by optical character recognition (OCR). In the following, this dataset is referred to as *Archive22k*.

3.2. Tobacco800

As a second evaluation set, we run our classification process on the *Tobacco800* document image database (Lewis et al., 2006). The dataset allows comparing the performance of our approach to other recent studies.

The Tobacco800 dataset is a small annotated subset of the Truth Tobacco Industry Documents, a collection of more than 14 million documents originating from seven major U.S. tobacco industry organizations dealing with their research, manufacturing, and marketing during the last decades. The documents had to be publicly released due to lawsuits in the United States.

The annotated subset for our experiments is composed of 1,290 document images sampled from the original corpus. Similar to the German dataset, it contains multi-page documents of different types (e.g. letters, invoices, hand-written documents) and thus is well suited for evaluation of our

Table 1: Recent work on page stream segmentation

Authors	PSS	DIC	Visual Features	Text Features	Architecture	Accuracy
Daher; Belaid (2014)	X			X	SVM, MLP	F = 0.8 - 0.9
Agin et al. (2015)	X		X	(X, fonts)	SVM, RF, MLP	F = 0.89
Harley et al. (2015)		X	X		CNN	A = 0.76 - 0.90
Noce et al. (2016)		X	X	X	CNN	A = 0.8 - 0.9
Gallo et al. (2016)	(X, indirect)	X	X		CNN+MLP	A = 0.88
Our approach	X		X	X	SVM	
					CNN+MLP	see Section 5.

task. Samples from the Tobacco dataset were also used in (Harley et al., 2015) and (Noce et al., 2016). Again, we extract text information for each page via OCR from the original page scans, OTSU-binarize them to a black/white color palette and resize them to a 224 $\times$ 224 pixel resolution. The lower lines in Fig. 1 and 2 show examples of first pages, resp. subsequent pages from Tobacco800 documents.

As the example pages show, both collections share similarities in their visual appearance. First pages compared to subsequent ones may contain distinct header elements. But in general, the human observer has difficulties to identify clear layout patterns discriminating between both classes, especially for the Archive22k documents. Therefore, visual features alone may not be sufficient for accurate PSS.
Regarding their textual content, the two datasets share certain similarities but also differ with respect to language, size, and creatorship. Both have in common that they cover long time periods and are thematically located within a rather narrow domain (nuclear waste disposal, tobacco industry). Nonetheless, they largely differ with respect to characteristics of content creators. On the one hand, there is a state-run research library archiving material from a wide variety of actors, while on the other hand there are internal documents from a rather small set of business actors with corporate design standards. Due to this, we expect different performance from textual and visual features for PSS on both datasets.

4. Binary classification for PSS

Analogue to (Daher and Abdel, 2014) and (Agin et al., 2015), we approach PSS as a binary classification task on single pages from a data stream. Pages are classified into either continuity of the *same document* (SD) or beginning of a *new document* (ND). For classification, we compare two architectures: SVM with specifically engineered text features (4.1.) and a combination of CNN and MLP with both, textual and visual features (4.2.).

4.1. Baseline: SVM on text features

As a baseline, we use linear text classification together with specifically engineered features for PSS. For this first step, we rely on SVM with a linear kernel[2]. This learning algorithm has proven to be very efficient for binary classification problems with sparse and large feature spaces (Joachims, 1998) and is computationally much faster than neural network architectures.[3]
We extract four types of features from the OCR-ed text data of the single pages.

Unigrams: Page texts were tokenized and resulting tokens reduced to their word stem. We further replaced digits in tokens with a #-character and pruned types from the vocabulary which occurred less than 3 times (Tobacco800), resp. 10 times (Archive22k). Pruning was applied to maintain manageable vocabulary sizes and reduce noise from infrequent events in the data. Different thresholds for feature pruning were chosen with respect to different collection sizes. This step resulted in 6,849 (Tobacco800), resp. 18,917 (Archive22k) features encoding raw frequency counts of all word types on each page.

Topic composition: In a second step, we obtained features of topical composition for each page from an unsupervised machine learning process. For this, we rely on Latent Dirichlet Allocation (LDA), also referred to as topic modeling (Blei et al., 2003).[4] Topic proportions based on multinomial posterior probability distributions θ from a topic model can be used as a dense feature vector comprising latent semantics of the modeled documents. In addition to highly sparse n-gram features, they can provide useful information to any text classifier. Following a method proposed by (Phan et al., 2011), we presented single page texts as pseudo-documents to the process and compute a model with $K = 50$ (Tobacco800), resp. $K = 100$ (Archive22k) topics. Different topic resolutions were chosen again with respect to different collection sizes. For each page p, we then use the resulting topic-page distribution θ_p as feature vector supplementary to the previously extracted vector of unigram counts.

Topic difference: We expect multi-page documents to comprise a rather coherent topic structure. Thus, for each

[2]We use the Liblinear library by (Fan et al., 2008)

[3]We refrain from using image features in this architecture, because pixel features are not supposed to be linearly separable. First experiments confirmed that pixel features do not contribute discriminative information on top of text features to the linear SVM for our task. Of course, we could use a different SVM kernel for image classification. But, very likely we would lose the advantage of computational speed. Due to this, we stick to text features for our baseline method.

[4]Actually, there is a large variety of unsupervised topic models as well as many other methods to reduce sparse, high-dimensional text data to a dense, lower-dimensional space (e.g. latent semantic analysis). For our baseline system, we stick to LDA as the seminal and most widely-used topic model.

page p, we determine the difference between its topic composition θ_p and its predecessor θ_{p-1} as a third feature type for PSS. We utilize two measures, Hellinger distance and Cosine distance, to create two additional features. While the former is a common metric to compare two probability distributions, the latter also has been adopted successfully to compare topic model results (Niekler and Jähnichen, 2012). Distance values near zero indicate a high similarity of topic composition compared to the predecessor page. Values near one indicate a significant change of topic composition which could indicate the beginning of a new document.

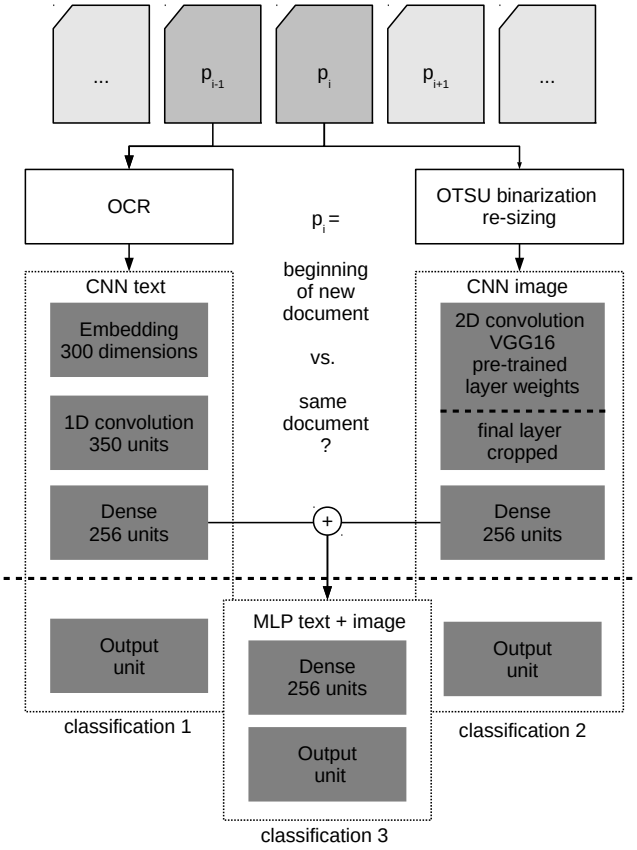
Predecessor pages: As a last feature type, we add a copy of features extracted in the previous three steps belonging to the predecessor page as new features to each current page. This can be achieved easily by appending their values together with a new unique feature identifier. For this, we simply concatenate existing feature identifiers with a prefix, e.g. 'PREV#'. This is necessary to allow for the distinction between feature values for the current page and copied values from the predecessor page. By this, any classifier not only can rely on the information about characteristics of the current page for its decision but also may learn from information contained on the previous page. For instance, the presence of a salutation phrase such as "With kind regards" on a predecessor page highly increases the probability for the beginning of a new document on the current page.
The performance of *SVM classification* to determine for each page whether it is the beginning of a new document or the continuation of the current document is tested in different steps. In each step, one of the four just introduced feature types is added to the feature set. Step-wise addition of the feature types to the linear SVM allows controlling whether each type effectively provides valuable information for the process.

4.2. Neural networks on text and image features

For our new PSS approach (cp. Fig. 3 for a schematic representation of the architecture), we first create two separate convolutional neural networks (CNN) for binary classification of pages into either SD or ND, one based on text data and another based on image scans. In a third step, we combine the learned parameters from the two final hidden layers of both CNN to an input vector of features for a multi-layer perceptron. This MLP delivers a third and final classification result based on both feature types.

CNN for text data: (Kim, 2014) proposed a simple but effective CNN-architecture for text classification which achieved high performance for sentiment analysis tasks on standard data sets. He uses 1-dimensional convolution over word sequences encoded as embedding vectors. We adopt a slightly simpler version of this network architecture by relying on only one kernel size instead of combining convolution layers with three different kernel sizes. Our network starts with an embedding layer with 300 dimensions, followed by a convolution layer with 350 filters and a kernel size of 3. On the resulting convolution filters, global max pooling is applied, followed by a dense layer of 256 neurons with "ReLU" activation, a dropout layer ($dr = .5$) and a final prediction layer for the binary class (sigmoid activation). The embedding layer is randomly initialized.[5] Learning for this network was performed using RMSProp optimization with learning rate 0.0002 and mini-batches of size 32.

Figure 3: CNN + MLP architecture for PSS



CNN for image data: Following the works in (Noce et al., 2016) and (Gallo et al., 2016), we use a very deep CNN architecture to classify scanned pages based on their binarized and resized representation as 224×224 pixels. We employ a network of 16 weight layers with very small convolution filters (3×3) and max pooling as introduced by (Simonyan and Zisserman, 2014). The network is initialized with pretrained weights based on the 'imagenet' dataset (VGG-16). Actually, 'imagenet' provides manually labeled photographs for object recognition tasks. But, earlier work has shown that CNN weights pre-trained on imagenet, although not specifically intended for the task of document image classification, can significantly improve DIC results for small datasets, too (Harley et al., 2015). Hence, we expect them to be beneficial also for our PSS task.
To allow the network to adapt to our specific data and clas-

[5]High performance for sentiment analysis in (Kim, 2014) is achieved by initializing the embedding layer with pre-trained word2vec embeddings (Mikolov et al., 2013) obtained from very large empirical data sets. Since we operate with data from two different languages, do not classify for semantic categories such as sentiments and also have a situation of rather noisy OCR data, we refrained from using pre-trained word embeddings in our setup.

sification task, we applied a common technique of fine-tuning pretrained deep CNN. For this, we removed the final prediction layer and flattened the output of the last fully connected layer. Then, we fixate all weights of the original model layers. On top of this architecture, we added a new trainable, fully connected layer with 256 units and dropout regularization ($dr = .5$), and a new final prediction layer (sigmoid activation) for our binary classification task. Learning for this network was performed using the Adam optimizer with a small learning rate ($lr = 0.0001$) and mini-batches of size 32.

Combining text and visual features: Each of the two previously introduced CNN are capable of classifying pages into either SD or ND on their own. But, since different information is utilized in each approach, we expect a performance gain from combining textual and visual information. For this, we modify the two previously introduced models in the same way. First, each model is trained on the training data individually. Then, we remove the final prediction layer from each model. In a next step, each example from the training and test data is fed into the networks again, to receive prediction values from the last fully connected layers of the two pruned networks. The output values from these last layers can be interpreted as new feature vectors for each data instance which encode dimensionality-reduced information from both, text and images.

From the text-based CNN, we receive a feature vector of 256 dimensions for each page according to the last dense layer of the model. To this vector, we concatenate the K inferred topic proportion features and the two topic distance features from our baseline approach. Since text features from predecessor pages proved to be very useful in SVM baseline classification, we also use features from neighbor pages in our final model. For this, we concatenate the vector of the current page with the vectors from its two predecessor pages to one text feature vector of length $3 \times (256 + K + 2)$. In a last step, we concatenate the 256 image features from the image-based CNN to receive a final vector of 1,180 (Tobacco800), resp. 1,330 (Archive22k) dimensions.[6]

These final feature vectors now encode both, text and visual information from each page. They serve as input for a new MLP network consisting of 256 fully connected nodes with "ReLU" activation and l2-regularization ($factor = 0.01$), followed by dropout regularization ($dr = .5$) and a final, fully connected prediction layer with sigmoid activation. Learning is performed using the Adam optimizer ($lr = 0.0005$) and a batch size of 16.

5. Evaluation

Quantitative evaluation: Table 2 displays the results of all tested model architectures and features types for PSS on our two investigated data sets. Performance is measured by the accuracy of identification of a new document beginning vs. continuity of the same document. Since the distribution of both classes is fairly uneven due to different document length (there are a lot more pages in the SD class), we additionally use kappa statistics to report a chance-corrected agreement between human and machine separations of page streams.

Table 2: Evaluation of page stream segmentation

Approach/dataset	**Archive22k**		**Tobacco800**	
	Acc.	kappa	Acc.	kappa
SVM unigrams	0.840	0.421	0.829	0.640
+ topic composition	0.839	0.419	0.829	0.640
+ topic difference	0.847	0.446	0.837	0.657
+ predecessor page	0.855	0.446	0.822	0.624
CNN Text	0.904	0.594	0.760	0.493
CNN Image	0.884	0.515	0.837	0.654
MLP Image + Text	**0.929**	**0.691**	**0.911**	**0.816**

The text features specifically engineered for PSS based on LDA topic composition and difference between consecutive pages improve the SVM results for text-based classification. Adding features from the predecessor page improves results for one dataset (Archive22k), but not for the other (Tobacco800).

For the German dataset, we can observe that document boundaries can be identified more accurately with the CNN architectures than with linear SVM classification. For the English dataset, SVM constantly beats convolutional neural net classification on text features, but not on image features. One potential reason might be the rather small size of the dataset which does not contain enough examples for the complex CNN architecture to learn from.

For both datasets, accuracy and kappa statistics improve significantly when image and text feature types are combined in one MLP architecture. The classifier achieves circa 93 % accuracy on the German dataset and more than 91 % on the English data. Compared to the results reported by the studies in section 2., this can be regarded as a new state-of-the-art for page stream segmentation.

Qualitative evaluation: Although first pages and subsequent pages of documents can be distinguished with high accuracy, our improved PSS approach still makes a considerable number of errors. There are two types of errors for the binary classification of pages: False positives (FP) and false negatives (FN). According to the manually separated pages in the gold standard, FP are subsequent pages (class SD) that are recognized by the classifier as first page (class ND). FN are defined the other way around. For SVM classification in the German dataset, FP account for about three quarters of all errors in the test set. FN make up about one quarter of all errors. This mismatch means that automatic PSS potentially splits the page stream into more documents than there are actually in the gold standard.

For the final MLP architecture, we observe not only an increased accuracy but also a more balanced ratio between FP and FN. Apparently, the architecture is able to avoid more FP errors that FN errors resulting in less (incorrect) document splits. On closer inspection, the remaining FPs often prove to contain characteristics of valid first pages, e.g. the beginning of a sub-document attached to one main docu-

[6] Concatenating text features from a window size 3 has been decided experimentally. We also found that concatenating image features from predecessor pages did not improve the final performance.

ment. This means although an automatic split is counted as an error in the quantitative evaluation, it nevertheless can represent a meaningful, content-related split for our application of retro-digitizing a large paper archive.

6. Discussion

We presented a new approach for page stream segmentation based on binary classification of pages. Our approach combines two convolutional neural networks to create features from image and text data which are used as input for a third MLP network. Our approach achieves very high accuracy for the task to identify the beginning of a new document in a flow of scanned document pages. An accuracy above 91 % for the Tobacco800 dataset which has been used in previous studies on this task, and accuracy of 93 % on our own dataset can be regarded as a new state-of-the-art for this task. The approach allowed us to drastically reduce costs for separating batch-scanned pages into document units in our project of retro-digitizing a research archive of around one million pages.

7. Acknowledgements

This work has been realized at the Leipzig University in the joint research project "Knowledge Management of Legacy Documents in Science, Administration and Industry" together with the Helmholtz Research Centre for Environmental Health in Munich and the CID GmbH, Freigericht. The authors thank colleagues at Helmholtz and CID for their valuable support.

8. Bibliographical References

Agin, O., Ulas, C., Ahat, M., and Bekar, C. (2015). An approach to the segmentation of multi-page document flow using binary classification. In *6th International Conference on Graphic and Image Processing (ICGIP 2014)*.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Daher, H. and Abdel, B. (2014). Document flow segmentation for business applications. In *Document Recognition and Retrieval XXI*, pages 9201–9215, San Francisco, France.

Fan, R.-e., Chang, K.-w., Hsieh, C.-j., Wang, X.-r., and Lin, C.-j. (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Gallo, I., Noce, L., Zamberletti, A., and Calefati, A. (2016). Deep neural networks for page stream segmentation and classification. In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7.

Gordo, A., Rusiñol, M., Karatzas, D., and Bagdanov, A. D. (2013). Document classification and page stream segmentation for digital mailroom applications. In *12th International Conference on Document Analysis and Recognition*, pages 621–625.

Harley, A., Ufkes, A., and Derpanis, K. (2015). Evaluation of deep convolutional nets for document image classification and retrieval. In *13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 991–995.

Isemann, D., Niekler, A., Preßler, B., Viereck, F., and Heyer, G. (2014). OCR of legacy documents as a building block in industrial disaster prevention. In *DIMPLE Workshop on DIsaster Management and Principled Large-scale information Extraction for and post emergency logistics, LREC 2014*.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *EMNLP 2014*.

Lewis, D., Agam, G., Argamon, S., Frieder, O., Grossman, D., and Heard, J. (2006). Building a test collection for complex document information processing. In *Proc. 29th Annual Int. ACM SIGIR Conference*, pages 665–666.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality.

Niekler, A. and Jähnichen, P. (2012). Matching results of latent dirichlet allocation for text. In *Proceedings of the 11th International Conference on Cognitive Modeling (ICCM 2012)*, pages 317–322. Universitätsverlag der TU Berlin.

Noce, L., Gallo, I., Zamberletti, A., and Calefati, A. (2016). Embedded textual content for document image classification with convolutional neural networks. In *Proceedings of the 2016 ACM Symposium on Document Engineering (DocEng)*, pages 165–173, New York. ACM.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, jan.

Phan, X.-H., Nguyen, C.-T., Le, D.-T., Nguyen, L.-M., Horiguchi, S., and Ha, Q.-T. (2011). A hidden topic-based framework toward building applications with short web documents. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):961–976.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Automating Document Discovery in the Systematic Review Process: How to Use Chaff to Extract Wheat

Christopher Norman,[1,2] Mariska Leeflang,[2] Pierre Zweigenbaum,[1] Aurélie Névéol[1]
[1] LIMSI, CNRS, Université Paris Saclay, F-91405 Orsay
[1] Academic Medical Center, University of Amsterdam, Amsterdam, the Netherlands
{christopher.norman, pierre.zweigenbaum, aurelie.neveol}@limsi.fr
m.m.leeflang@amc.uva.nl

Abstract

Systematic reviews in e.g. empirical medicine address research questions by comprehensively examining the entire published literature. Conventionally, manual literature surveys decide inclusion in two steps, first based on abstracts and title, then by full text, yet current methods to automate the process make no distinction between gold data from these two stages. In this work we compare the impact different schemes for choosing positive and negative examples from the different screening stages have on the training of automated systems. We train a ranker using logistic regression and evaluate it on a new gold standard dataset for clinical NLP, and on an existing gold standard dataset for drug class efficacy. The classification and ranking achieves an average AUC of 0.803 and 0.768 when relying on gold standard decisions based on title and abstracts of articles, and an AUC of 0.625 and 0.839 when relying on gold standard decisions based on full text. Our results suggest that it makes little difference which screening stage the gold standard decisions are drawn from, and that the decisions need not be based on the full text. The results further suggest that common-off-the-shelf algorithms can reduce the amount of work required to retrieve relevant literature.

Keywords: Evidence Based Medicine, Information Storage and Retrieval, Review Literature as Topic

1. Introduction

Systematic reviews seek to systematically gather all published evidence addressing a given research question and analyze the aggregate results. Systematic reviews constitute some of the strongest forms of scientific evidence, are an integral part of evidence based medicine, and serve a key role in informing and guiding public and institutional decision-making (Wright et al., 2007).

One limiting factor of systematic reviews is that they tend to be prohibitively costly to produce.[1] The number of references needed to be manually screened in order to satisfy the requirement that virtually all relevant articles have been identified can number in the tens of thousands. Often only some dozens of these references are selected for the final meta-analysis, and the selection process may require months of work for several reviewers (O'Mara-Eves et al., 2015).

The screening process starts with identifying an initial set of candidate references, typically by searching databases using boolean queries handcrafted by experts. From this initial set of references, reviewers first screen for inclusion based on titles and abstracts, and then based on the full text (O'Mara-Eves et al., 2015) as illustrated in figure 1. In this paper we will call the references excluded in the first screening stage No ('N'), references excluded in the second screening stage Maybe ('M'), and references included in the final analysis Yes ('Y').

This selection is divided into two stages because while final decisions can only be based on the full text of articles, many references can be rejected based only on title and abstract. Retrieving the full text articles, which often needs to be done manually, is generally only feasible for a fraction of the articles in large systematic reviews (Tsafnat et al., 2014). However, even though humans approach screening as a two-step process, automation methods to date have generally approached the problem as a one-step process to find the relevant articles.

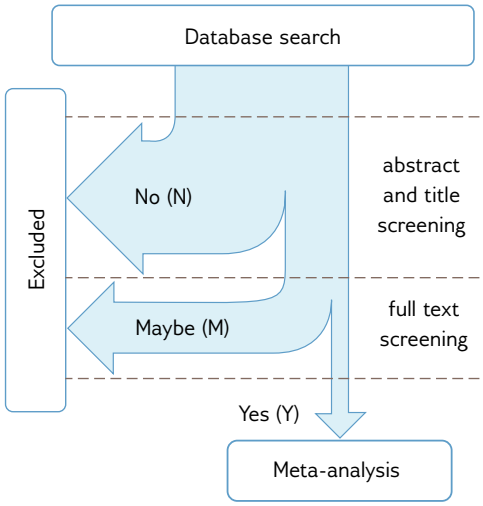


Figure 1: Overview of the data flow during the screening process in systematic reviews.

In this paper we ask if there is value in recognizing the distinction between each successive stage of the process. Our contribution is two-fold: First, we conduct experiments to inform methodology choices for automating the literature screening, and to find ways to improve the quality of constructing datasets used to train such retrieval methods. Second, we experiment on an existing reference dataset and introduce a new, complementary dataset.

[1] Although primary clinical research is often more expensive.

2. Related Work

Methods for automation have been attempted with varying degrees of success in technology assisted review in several topics in biomedicine (O'Mara-Eves et al., 2015). Technology assisted review has also been implemented in other fields with similarly stringent recall requirements, such as patent search (Stein et al., 2012), and electronic discovery (Grossman and Cormack, 2011). Automated document discovery is typically cast as a ranking or classification problem (O'Mara-Eves et al., 2015).

Common methods for automation include Support Vector Machines and variants of Naive Bayes, including Complement Naive Bayes (Matwin et al., 2010), and Multinomial Naive Bayes (Matwin and Sazonova, 2012). Other methods have been tried, including Voting Perceptrons (Cohen et al., 2006), Decision Trees (Bekhuis and Demner-Fushman, 2010), Evolutional SVM (Bekhuis and Demner-Fushman, 2010), WAODE (Bekhuis and Demner-Fushman, 2010), kNN (García Adeva et al., 2014), Rocchia (García Adeva et al., 2014), hypernym relations (Fiszman et al., 2010), ontologies (Sun et al., 2012), Generalized Linear Models (Shekelle et al., 2012), Gradient Boosting Machines (Shekelle et al., 2012), Random Indexing (Jonnalagadda and Petitti, 2014), and Random Forests (Khabsa et al., 2016). Few of the methods proposed have been evaluated on common datasets however, and it is therefore difficult to draw conclusions about relative performance (O'Mara-Eves et al., 2015).

Recently, Khabsa et al. (2016) proposed using random forests, and compared the performance of their system with the reported performance of earlier systems on Cohen's 15 reviews (see section 4.). Other methods have also been evaluated on the same dataset (Jonnalagadda and Petitti, 2014). For these reasons, and because the dataset is publicly available we will use this dataset as our baseline.

However, even though humans approach screening as a series of filters of increasingly fine granularity, all methods we have reviewed in previous literature approach the problem as a one stage process.

3. Objective

We construct an automatic screening system using a standard, off-the-shelf classifier. We describe our implementation and compare it with the state of the art to show that it functions as intended. We then apply our implementation on two datasets for systematic reviews, one of which is novel, in order to answer the following questions:

1. Can we separate the screening into two stages?
2. Do we need examples from all stages of screening (Y, M, N)?
3. Should the positive labels match the decisions in the first or second stage of the screening?

To our knowledge, these questions have not yet been considered by existing literature.

Note that the aim of this study is not to improve upon the state of the art, but to investigate how different labeling schemes affect datasets for literature screening.

Dataset	Topic	Y	M	N
Yearbook	ClinicalNLP (2017)	11	70	244 (177)
	ClinicalNLP (2016)	23	60	267 (191)
Cohen	CalciumChannelBlockers	100	180	938
	ACEInhibitors	41	142	2361
	BetaBlockers	42	260	1770
	Opiods	15	33	1867
	OralHypoglycemics	136	3	364
	Statins	85	88	3292
	SkeletalMuscleRelaxants	9	25	1609
	Antihistamines	16	76	218
	ProtonPumpInhibitors	51	187	1095
	Triptans	24	194	453
	NSAIDS	41	47	305
	ADHD	20	64	767
	AtypicalAntipsychotics	146	218	756
	UrinaryIncontinence	40	38	249
	Estrogens	80	0	288

Table 1: The distribution of class labels in each dataset. The Yearbook makes an additional separation of N into references that are off-topic and those that are on-topic but does not fit the research question of the review. The number of off-topic references is given in parentheses.

4. Datasets

To address our research questions, we use two datasets that label not only Y and N judgments, but explicitly mark the M subset.

The datasets each consist of references in the form of PubMed® identifiers (PMID) with corresponding inclusion labels (i.e. Y, M, or N) and topic labels. Article metadata, as well as titles and abstracts, are not included in either dataset, but can be downloaded from Medline® using the Entrez API.[2] The distribution of references from each review stage is reported in Table 1.

Like in the majority of previous literature, we assume that labeled training data is available, which is generally not true for new reviews.

Training data might however exist from past reviews on the same or similar topics. We call such cases where the training data is drawn from similar, but not exactly the same topic, *inter-topic* training.

It may also be possible to have reviewers label small batches of references, and use these as training data for the remainder of the process. Furthermore, systematic reviews sometimes need to be updated, in which case we can use the data from previous iterations for training. We call such cases where the training data is drawn from exactly the same topic *intra-topic* training.

4.1. The Yearbook Dataset

We construct this dataset by using the references that were considered on topic in the review on clinical NLP done by Névéol and Zweigenbaum (2016; 2017) for the IMIA Yearbook of Medical Informatics.

This review is updated annually, and the resulting dataset illustrates systematic reviews updates. In each iteration, previous data can be leveraged to train an *intra-topic* classifier. This dataset is made available in CSV and JSON format,[3] and is planned to be updated to incorporate future iterations of the review.

[2] https://www.ncbi.nlm.nih.gov/home/develop/api/

[3] Available from DOI: 10.5281/zenodo.1173076

Topic \ Measure	Intertopic			Intratopic			
	WSS@95	AUC		WSS@95		AUC	
			(Cohen)		(Khabsa)		(Khabsa)
CalciumChannelBlockers	.129	**.759**	.712	**.398**	.287 (RF)	.825	**.873** (SVM)
ACEInhibitors	.566	**.817**	.806	**.629**	.523 (CNB)	.917	**.951** (RF)
BetaBlockers	.400	**.837**	.801	**.511**	.367 (CNB)	.863	**.893** (RF)
Opiods	.301	**.885**	.856	**.590**	.554 (CNB)	.905	**.913** (RF)
OralHypoglycemics	.072	**.657**	.573	**.111**	.080 (CNB)	.568	**.781** (SVM)
Statins	.266	**.826**	.773	**.436**	.400 (RF)	.873	**.915** (RF)
SkeletalMuscleRelaxants	.241	.828	**.836**	**.429**	.371 (RF)	.740	**.794** (RF)
Antihistamines	.073	**.652**	.620	**.149**	.148 (CNB)	.650	**.722** (SVM)
ProtonPumpInhibitors	.377	**.823**	.793	**.307**	.288 (RF)	.826	**.880** (RF)
Triptans	.464	.819	**.823**	.303	**.312** (RF)	.792	**.909** (SVM)
NSAIDS	.671	**.912**	.899	**.537**	.528 (CNB)	.861	**.951** (SVM)
ADHD	.128	**.591**	.469	.616	**.668** (VP)	.908	**.951** (RF)
AtypicalAntipsychotics	.162	**.759**	.653	**.210**	.206 (CNB)	.779	**.835** (RF)
UrinaryIncontinence	.374	**.887**	.851	**.422**	.411 (RF)	.784	**.890** (SVM)
Estrogens	.176	**.693**	.588	.292	**.375** (CNB)	.689	**.887** (SVM)

Table 2: Results comparing our implementation to the state of the art. Intertopic results report the average over 5 runs. Intratopic results report the average over 10 runs (5×2 cross validation). Both cases use (Y||MN). Intertopic state of the art results are taken from Cohen (2008). Intratopic state of the art results are taken from Khabsa et al. (2016), who also report results on Complement Naive Bayes (CNB) by Matwin et al. (2010), Voting Perceptrons (VP) by Cohen et al. (2006), and Support Vector Machines (SVM) by Cohen (2008). Exact intertopic AUC scores are not explicitly reported by Cohen (2008) and have instead been extracted from Figure 1 in his paper.

4.2. The Cohen Dataset

In one of the early papers on screening automation, Cohen et al. (2006) constructed a dataset from 15 systematic reviews on drug efficacy. This dataset was later extended to 18 (Cohen et al., 2010), then to 24 reviews (Cohen et al., 2009). The smaller dataset comprising 15 reviews has been made available (Cohen et al., 2006).[4] Several methods, including Voting Perceptrons (Cohen et al., 2006), Complement Naive Bayes (Matwin and Sazonova, 2012), SVM (Cohen, 2006; Cohen et al., 2009; Cohen, 2008), Random Indexing (Jonnalagadda and Petitti, 2014), and Random Forests (Khabsa et al., 2016) have been tested on this dataset, and we can therefore use this dataset to compare our performance against previous work.

This dataset illustrates leveraging training data from similar topics. For each subtopic, data from the other subtopics may be leveraged to build an *inter-topic* classifier.

5. Document Ranking Method

We construct a ranker by extracting bag-of-n-grams ($n \leq 3$) over words in the titles and abstracts. We use both tf-idf scores and binary features, and both stemmed and unstemmed versions. The n-grams from the background, method, results, and conclusion of the abstract are also each considered in separation. We also extract article metadata, namely author-assigned keywords, journal names, and publication types. For Cohen we also extract MeSH terms, but omit these for Yearbook since MeSH terms are generally not yet available when reviews are updated.

We use a ranking approach only. In practice we ignore the decision boundary used by the logistic regression, and instead leave the decision as to where to stop the search entirely to the reviewer(s). Point measures, such as recall, can therefore only be computed as a function of the position in the ranked list.

We use the implementation of logistic regression in sklearn (Pedregosa et al., 2011) trained using stochastic gradient descent, i.e. the SGDClassifier trained using log loss. We train the ranker for a maximum of 100,000 iterations.

We generally follow the setup of Cohen et al. (2006), and Khabsa et al. (2016). For intra-topic cross validation we use 2-fold cross validation on each topic and repeat this 5 times. For intertopic training we report the average of 5 repetitions. In each experiment we report the average and standard deviation over all folds and repetitions. All hyperparameters remain constant throughout each experiment.

Unless otherwise stated, we use the default settings for all parameters. We train the ranker and calculate the AUC similarly to Cohen et al. (2009; 2008). Cross validation was done both inter-topic and intra-topic similarly to the later work of Cohen et al. (2009), and results are reported for each case. We also report the WSS@95 scores (Cohen et al., 2006) in order to compare our results against the naive bayes methods of Matwin et al. (2011). We handle class imbalance by (pseudo)randomly undersampling the majority class to have the same number of instances as the minority class. We however observe that this yields poor results when the number of examples in the majority class is low, and therefore include a minimum of 500 majority class examples.

[4] The old link has however expired. The data can now be found at `http://skynet.ohsu.edu/~cohenaa/systematic-drug-class-review-data.html`

We increase the weights on the relevant references to 80 to emulate differing costs of misclassification. We also chose $\alpha = 10^{-4}$ as a reasonable value for the regularization term for the Cohen dataset, and $\alpha = 0.05$ for Yearbook. We selected these values through experimentation on one of the topics in Cohen (CalciumChannelBlockers), and the first iteration of the Yearbook dataset (2016).

5.1. Experimental Setup

We perform two types of experiments;
First, we run our implementation on the Cohen dataset and compare it with the reported performance of previous work. We do this in order to verify the correctness of our algorithm.
Second, we perform experiments where we enumerate different ways to treat Y, M, and N labels as positive and negative examples.
We test if it is feasible to emulate the way humans conduct systematic reviews by considering a two-stage approach where we first separate YM from N, and then Y from M.
We test whether treating the M subset as positive or negative labels impacts the performance by comparing the performance when separating YM from N with the performance when separating Y from MN.
And finally, we evaluate models where we treat the M subset as positive examples during training but negative during testing in order to test whether classification in earlier stages generalize to classification in later stages.
We report the work saved over sampling at 95% recall (WSS@95) (Cohen et al., 2006) and the area under the receiver operator characteristic curve (AUC) (Cohen, 2008) in order to bring our results in line with previous literature (Khabsa et al., 2016). The WSS@95 metric measures the theoretical work saved when using the model to retrieve 95% of the relevant articles.

6. Results

We present our comparison with the state of the art in Table 2. In Tables 3a–3c we present the results of our experiments using data with different compositions of examples in terms of Y, M, and N.

7. Discussion

In this section we discuss the results, in order to verify that our system works as intended, and to address the questions we set out in Section 3. Objective.

7.1. Performance of our System

Intuitively: based on the WSS@95 scores (Tables 3a, 3b), our method could save the reviewers from having to look at 46 (Antihistamines) to 1058 (BetaBlockers) references depending on the topic, or about 605 references on average. The results of our implementation are comparable to state of the art results across the board (Table 2). Our implementation exhibits equal or better results for intertopic training (Table 2). For intratopic training, our implementation exhibit worse results in terms of AUC, but better scores in terms of WSS@95. Our implementation seems to perform worse than the state of the art mainly on the topics where there are no or very few M (OralHypoGlycemics, Estrogens). It is also possible that the additional features used by Khabsa et al. (references cited) can explain some of the difference in results.

7.2. Can We Separate the Screening into Two Stages?

Separating the screening into two stages would entail first screening in terms of (YM||N) followed by (Y||M). However, from Tables 3a–3c it is clear that while (Y||MN) is feasible, (Y||M) is considerably more difficult than (Y||N) or (Y||MN) (Tables 3a–3c). The ranker is however doing a slightly better job on BetaBlockers and Triptans (Tables 3b and 3c).

In particular, when separating Y from M, the ranker is not performing much better than chance on many of the topics. This is to be expected, since M represent those references the human annotators required the full text to judge, and it would be unreasonable to expect the ranker to be able to judge these based only on title and abstract.

Consequently, we can certainly perform (YM||N) as an initial step, but (Y||M) would at the very least require ranking the full text articles.

7.3. Do We Need Examples from All Stages of Screening (Y, M, N)?

We observe similar results for (Y|M|N) and (Y||MN) on Cohen, i.e. we can train a ranker using positive examples that were included based on title and abstract (Y+M), even if these were to turn out to be non-relevant upon inspecting the full text (M). On the Yearbook dataset we observe better scores for (Y|M|N) than (Y||MN), likely due to the number of Y available for training (23) being much smaller. In Table 3b we can generally observe similar results for (Y|M|N) and (Y||MN), the exceptions being Triptans and NSAIDS where we observe better results for (Y||MN). We also observe similar results for (Y|M|N) and (Y||MN) on the Yearbook data. On some topics we observe better results for (Y|M|N), but the difference is small.

Furthermore, both (Y||N) and (M||N) seem to give reasonable results, although these results are not directly comparable to the results for (Y|M|N). We can also observe that (Y||N) is generally easier than (M||N). This could be due to Y containing fewer borderline cases.

Consequently, we do need positive examples drawn from Y or M, as well as negative examples drawn from N. It seems to make less difference whether we consider M to be positive or negative examples and we may be able to exclude either Y or M in training.

Interestingly it seems from Table 3a that it is more difficult to classify in terms of (YM||N) than (Y||MN) on Cohen, but the inverse is true on Yearbook. This might be explained by the small number of Y on Yearbook (11), and we can observe the same on the topics in Cohen with few Y (SkeletalMuscleRelaxants, ADHD). OralHypoglycemics have only 3 M and Estrogens no M at all, and we therefore exclude these topics from the results.

	(Y||MN)		(YM||N)		(Y|M|N)		(Y||M)		(Y||N)		(M||N)	
	WSS	AUC	WSS	AUC	WSS	AUC	WSS	AUC	WSS	AUC	WSS	AUC
Yearbook	.003	.625	.229	.803	.189	.808	.012	.481	.020	.738	.256	.785
Cohen	.449	.839	.265	.768	.472	.814	.163	.557	.423	.832	.239	.714

(a) Intra-topic results averaged over 10 runs (5 × 2 cross validation) for different dataset compositions. The averages were computed using weights proportional to the number of articles in each topic (Y+M+N, Y+M, Y+N, or M+N).

	(Y||MN)				(YM||N)				(Y|M|N)			
	WSS@95		AUC		WSS@95		AUC		WSS@95		AUC	
Topic	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
ClinicalNLP (Yearbook)	.003	.000	.625	.005	.229	.011	.803	.001	.189	.008	.808	.002
CalciumChannelBlockers	.398	.098	.825	.024	.218	.056	.764	.030	.338	.073	.790	.012
ACEInhibitors	.629	.158	.917	.020	.277	.050	.800	.021	.598	.126	.879	.027
BetaBlockers	.511	.157	.863	.030	.187	.047	.730	.025	.476	.210	.831	.021
Opiods	.590	.193	.905	.052	.366	.096	.817	.033	.705	.063	.881	.035
OralHypoglycemics	.111	.048	.568	.026	.138	.068	.579	.036	.089	.020	.583	.026
Statins	.436	.176	.873	.021	.254	.094	.779	.025	.421	.101	.864	.015
SkeletalMuscleRelaxants	.429	.221	.740	.113	.264	.180	.826	.064	.445	.116	.746	.057
Antihistamines	.149	.089	.650	.089	.126	.038	.566	.026	.239	.092	.596	.013
ProtonPumpInhibitors	.307	.191	.826	.044	.167	.043	.731	.023	.378	.058	.770	.037
Triptans	.303	.237	.792	.075	.300	.039	.746	.030	.412	.067	.691	.026
NSAIDS	.537	.184	.861	.022	.402	.072	.755	.042	.458	.057	.727	.024
ADHD	.616	.148	.908	.026	.697	.096	.910	.017	.828	.057	.906	.011
AtypicalAntipsychotics	.210	.044	.779	.012	.123	.024	.714	.027	.284	.057	.803	.022
UrinaryIncontinence	.422	.144	.784	.032	.207	.089	.660	.040	.475	.072	.750	.038
Estrogens	.292	.089	.689	.026	.266	.093	.715	.040	.319	.056	.693	.026

(b) Intratopic results averaged over 10 runs (5 × 2 cross validation) for different dataset compositions.

	(Y||M)				(Y||N)				(M||N)			
	WSS@95		AUC		WSS@95		AUC		WSS@95		AUC	
Topic	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
ClinicalNLP (Yearbook)	.012	.000	.481	.005	.020	.002	.738	.003	.256	.004	.785	.001
CalciumChannelBlockers	.141	.039	.590	.030	.421	.106	.852	.024	.208	.069	.743	.032
ACEInhibitors	.165	.083	.631	.059	.410	.370	.918	.032	.256	.063	.771	.020
BetaBlockers	.383	.096	.737	.021	.515	.135	.870	.034	.190	.031	.713	.018
Opiods	.131	.096	.526	.006	.592	.205	.906	.064	.249	.177	.762	.045
OralHypoglycemics	.058	.000	.387	.167	.105	.039	.579	.030	.754	.194	.826	.112
Statins	.125	.052	.560	.037	.439	.184	.879	.047	.240	.086	.708	.028
SkeletalMuscleRelaxants	.240	.143	.547	.017	.297	.149	.668	.078	.226	.163	.800	.067
Antihistamines	.204	.165	.554	.062	.161	.090	.700	.033	.128	.073	.583	.036
ProtonPumpInhibitors	.159	.052	.584	.022	.421	.168	.852	.026	.122	.046	.694	.032
Triptans	.199	.130	.695	.072	.437	.244	.880	.042	.272	.064	.746	.028
NSAIDS	.129	.050	.576	.056	.479	.185	.851	.017	.316	.094	.723	.027
ADHD	.193	.138	.588	.093	.707	.169	.938	.021	.639	.170	.916	.013
AtypicalAntipsychotics	.112	.023	.548	.017	.259	.114	.792	.030	.113	025	.629	.031
UrinaryIncontinence	.090	.038	.550	.024	.433	.159	.792	.033	.121	.103	.591	046
Estrogens	-	-	-	-	.233	.034	.686	.038	-	-	-	-

(c) Intratopic results averaged over 10 runs (5 × 2 cross validation) for different dataset compositions.

Table 3: (Y||MN) denotes results using Y as the positive class. (YM||N) denotes results using Y and M as the positive class. (Y|M|N) denotes results using Y and M as the positive class in training, and Y as the positive class in evaluation. (Y||M) denotes results using Y as the positive, and M as the negative class. (Y||N) denotes results using Y as the positive, and N as the negative class. (M||N) denotes results using M as the positive, and N as the negative class. Estrogens has no M, and is consequently excluded from the calculations of the results for (Y||M) and (M||N).

7.4. Can We Use M as Positive Examples for Training?

Cohen et al. previously discovered that while intratopic data is generally better than intertopic data (Cohen et al., 2006), the less targeted intertopic data can complement the intratopic data if the intratopic data is scarce (Cohen et al., 2006; Cohen et al., 2009). Our results suggest the same (Table 2), but also that we can generally use M as training examples to complement the Y. The intuition behind these ideas is similar: while it is generally important to have training data targeted for the particular problem, it is also important to have sufficient amounts of data, and less targeted training data can provide a supplement if only scarce amounts of data is available.

We can further compare the results for intratopic (Y|M|N) versus the results for intertopic (Y||MN) in Tables 3b and 2 to get a sense of whether complementing our training data by using M as positive examples works better than complementing our training data with less targeted data from similar topics.

We observe better results for intertopic (Y||MN) for OralHypoglycemics, SkeletalMuscleRelaxants, Antihistamines, Triptans, NSAIDS, and UrinaryIncontinence. This might in part be explained by OralHypoglycemics, SkeletalMuscleRelaxants and Antihistamines having few Y. We observe better results for intratopic (Y|M|N) on ACEInhibitors, ProtonPumpInhibitors, and ADHD. It is not clear why we observe this difference on these topics.

7.5. Strategies for Ranking Articles

From Tables 3a–3c it seems that there is no single approach that is clearly better for any kind of data. Which approach works best depends on the number of articles in each class, as well as the exact nature of articles in each stage. What parts of the data to e.g. use for training must therefore be decided based on the characteristics of the dataset, or by testing multiple approaches.

The results and conclusions of this study guided the strategic choices we made for the system submitted to the CLEF eHealth shared task *Technology Assisted Reviews in Empirical Medicine* (Norman et al., 2017; Kanoulas et al., 2017). We submitted four runs using different machine learning methods: 1) the (YM||N) approach described here 2) an (YM||N) approach using standard logistic regression (i.e. not trained using SGD), and 3) two variations of logistic regression with active learning, where the system starts using the (Y|M|N) approach and later switches to using the (Y||MN) approach once a sufficient number of Y have been discovered.

On the Cohen dataset approach 2 worked better than approach 1 for intratopic training and vice-versa, and we could reliably see improvements over either of these by using active learning. On the CLEF data however, approach 1 achieved much better results than either approach 2 or 3. We believe that this was at least partly due to the small number of relevant articles per topic in the CLEF dataset (Norman et al., 2017).

Our participation placed third to fifth in the evaluation overall, depending on metric used, and placed first among the systems not using active learning.

7.6. Limitations

This work relied on two datasets and a ranker developed in-house. It is not clear how the results generalize to other domains and datasets, or to other machine learning methods.

We observe fairly large variance for many of the runs (Tables 3b, 3c), and on many topics. This is particularly problematic for the WSS metric, but it also affects the AUC metric even averaged over ten repetitions. For instance, Estrogens has no M, and we should therefore expect the same results for (Y||MN) and (YM||N), yet we observe differences roughly equal to the standard deviation for the AUC. Previous literature generally do not report their variance, which complicates the comparison with previous results.

7.7. Future work

We are working on extending the system to use additional machine learning methods, including deep artificial neural networks, and to complement the system with information retrieval methods.

8. Conclusion

We find that in order to train rankers to automate the screening process we need to use 1) examples of excluded references (N), *and* 2) references included in either the first (M) or second stage of the screening (Y). In the systematic reviews, the M are those articles that were excluded after reading the full text, and so are in reality negative examples. However, our results suggest that these can still be used as positive examples for training. It may well be possible to construct an accurate ranker using only the M as the positive examples, without any real positive examples (i.e. Y) at all.

Our best results are achieved with (Y||MN) on the Cohen dataset, whereas our best results are achieved with (Y|M|N) on the Yearbook dataset. Given that the distribution of the labels is similar in both datasets it is likely that greater contribution of the M on the Yearbook dataset is due to its smaller size. For any new systematic review we only have whatever training data we label ourselves, and data scarcity is therefore one of the major issues we need to overcome. Even for systematic review updates the amount of positive training data available is typically modest since the number of included articles in any systematic review tends to be small (the Y column in Table 1).

Since the number of references that are provisionally included based on title and abstract (Y+M) can outnumber the final includes (Y) by almost ten to one (Table 1), using examples of M in addition to Y suggests a straightforward way to increase the amount of training data available (i.e. the Y|M|N approach), and thus potentially overcome the data scarcity problem, particularly if we do not have access to inter-topic training data. This does not seem to have been considered in previous work.

Our results also agree with the state of the art and suggest that common-off-the-shelf machine learning algorithms can accurately predict topical relevance of candidate articles for inclusion in systematic reviews.

In light of the results, we recommend that future datasets intended to be used either for training or for evaluation of

document screening should include a tripartite labeling reflecting the two filtering stages in manual systematic reviews. Strictly, only the distinction between YM and N is necessary for training, but we still likely want to only treat Y as positive during evaluation, since only these would be considered relevant for the purposes of the systematic review.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 676207.

9. Bibliographical References

Bekhuis, T. and Demner-Fushman, D. (2010). Towards automating the initial screening phase of a systematic review. *Studies in Health Technology and Informatics*, 160(PART 1):146–150.

Cohen, A. M., Hersh, W. R., Peterson, K., and Yen, P. (2006). Reducing Workload in Systematic Review Preparation Using Automated Citation Classification. pages 206–219.

Cohen, A. M., Ambert, K., and McDonagh, M. (2009). Cross-Topic Learning for Work Prioritization in Systematic Review Creation and Update. *Journal of the American Medical Informatics Association*, 16(5):690–704.

Cohen, A. M., Ambert, K., and McDonagh, M. (2010). A Prospective Evaluation of an Automated Classification System to Support Evidence-based Medicine and Systematic Review. *AMIA Annual Symposium Proceedings*, 2010:121–125.

Cohen, A. M. (2006). An effective general purpose approach for automated biomedical document classification. *AMIA Annual Symposium proceedings*, pages 161–165.

Cohen, A. M. (2008). Optimizing feature representation for automated systematic review work prioritization. *AMIA Annual Symposium proceedings*, pages 121–5.

Fiszman, M., Bray, B., Shin, D., Kilicoglu, H., Bennett, G., Bodenreider, O., and Rindflesch, T. (2010). Combining Relevance Assignment with Quality of the Evidence to Support Guideline Development. *Stud Health Technol Inform*, 160(1):709—-713.

García Adeva, J. J., Pikatza Atxa, J. M., Ubeda Carrillo, M., and Ansuategi Zengotitabengoa, E. (2014). Automatic text classification to support systematic reviews in medicine. *Expert Systems with Applications*, 41(4 PART 1):1498–1508.

Grossman, M. and Cormack, G. (2011). Technology-assisted review in e-discovery can be more effective and more efficient than exhaustive manual review. *Richmond Journal of Law and Technology*, 17(3).

Jonnalagadda, S. R. and Petitti, D. (2014). A new iterative method to reduce workload in the systematic review process. *Int J Comput Biol Drug Des*, 6(0):5–17.

Kanoulas, E., Li, D., Azzopardi, L., and Spijker, R. (2017). Overview of the CLEF technologically assisted reviews in empirical medicine. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation forum, Dublin, Ireland, September 11-14, 2017.*, CEUR Workshop Proceedings. CEUR-WS.org.

Khabsa, M., Elmagarmid, A., Ilyas, I., Hammady, H., and Ouzzani, M. (2016). Learning to identify relevant studies for systematic reviews using random forest and external information. *Machine Learning*, 102(3):465–482.

Matwin, S. and Sazonova, V. (2012). Direct comparison between support vector machine and multinomial naive Bayes algorithms for medical abstract classification. *Journal of the American Medical Informatics Association*, 19(5):917–917.

Matwin, S., Kouznetsov, A., Inkpen, D., Frunza, O., and O'Blenis, P. (2010). A new algorithm for reducing the workload of experts in performing systematic reviews. *Journal of the American Medical Informatics Association*, 17(4):446–53.

Matwin, S., Kouznetsov, A., Inkpen, D., and O'Blenis, P. (2011). Performance of support-vector-machine-based classification on 15 systematic review topics evaluated with the WSS@95 measure. *Journal of the American Medical Informatics Association*, 1(18):author reply 105.

Névéol, A. and Zweigenbaum, P. (2016). Clinical natural language processing in 2015: Leveraging the variety of texts of clinical interest. *IMIA Yearbook*, pages 234–239.

Névéol, A. and Zweigenbaum, P. (2017). Making sense of big textual data for health care: Findings from the section on clinical natural language processing. *Yearbook of medical informatics*, 26(01):228–233.

Norman, C., Leeflang, M., and Névéol, A. (2017). LIMSI@CLEF eHealth 2017 task 2: Logistic regression for automatic article ranking.

O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., and Ananiadou, S. (2015). Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Systematic reviews*, 4(1):5.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Shekelle, P. G., Dalal, S. R., and Shetty, K. D. (2012). A Pilot Study Using Machine Learning and Domain Knowledge To Facilitate Comparative Effectiveness Review Updating. *AHRQ*.

Stein, B., Hoppe, D., and Gollub, T. (2012). The impact of spelling errors on patent search. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 570–579. Association for Computational Linguistics.

Sun, Y. B., Yang, Y., Zhang, H., Zhang, W., and Wang, Q. (2012). Towards evidence-based ontology for supporting systematic literature review. *Evaluation and Assessment in Software Engineering*, 2012(1):171–175.

Tsafnat, G., Glasziou, P., Choong, M. K., Dunn, A., Galgani, F., and Coiera, E. (2014). Systematic review automation technologies. *Systematic reviews*, 3(1):74.

Wright, R. W., Brand, R. A., Dunn, W., and Spindler, K. P. (2007). How to write a systematic review. *Clinical orthopaedics and related research*, 455:23–29.

Two Multilingual Corpora Extracted from the Tenders Electronic Daily for Machine Learning and Machine Translation Applications

Oussama Ahmia[1,2], **Nicolas Béchet**[1], **Pierre-François Marteau**[1]
[1]IRISA, Université Bretagne Sud
Campus de Tohannic, 56000 Vannes FRANCE
[2] Jurismarchés, 2 Place Saint-Pierre, Nantes, 44000, FRANCE
{firstname.name}@irisa.fr

Abstract

The European "Tenders Electronic Daily" (TED) is a large source of semi-structured and multilingual data that is very valuable to the Natural Language Processing community. This data sets can effectively be used to address complex machine translation, multilingual terminology extraction, text-mining, or to benchmark information retrieval systems. Despite of the services offered by the user-friendliness of the web site that is made available to the public to access the publishing of the EU call for tenders, collecting and managing such kind of data is a great burden and consumes a lot of time and computing resources. This could explain why such a resource is not very (if any) exploited today by computer scientists or engineers in NLP. The aim of this paper is to describe two documented and easy-to-use multilingual corpora (one of them is a parallel corpus), extracted from the TED web source that we will release for the benefit of the NLP community.

Keywords: Multilingual corpora, Parallel Corpus, Call for Tender, European Languages, Natural Language Resource

1. Introduction

The world procurement generates daily large amounts of data, that represent useful knowledge for business intelligence tasks.
Among other sources, the European "Tenders Electronic Daily" (TED) system, publishes approximately 1,700 tenders five times a week in the TED servers[1]. More precisely, the publication office of the TED publishes about 460,000 contract notices per year, among which 175,000 tenders worth around 420 billion euros. These raw data are available as bulk downloads of these data that contain contract notices for tender data in XML format with a complex structure. Unfortunately, different versions of the XML data structure from year to year have been used, making the aggregation of the different bulks of data difficult. Furthermore, the collected documents are associated to a variable number of translations as well as variable sets of meta data that is used for indexing. Consequently, the collecting and managing of such data is a great burden and consumes a lot of time and computing resources.

The aim of this paper is to describe a processed version of this database in a raw text format that can be directly and easily used for text mining and natural language processing tasks. We also aim at making this processed dataset available to the scientific community and can be downloaded[2] along with a simple Python API for easier manipulation.

The provided dataset is declined into two sub-datasets created from the TED's documents that have been published between January 2011 and August 2017.

1. The first sub-dataset, fd-TED, is a (multilingual) corpus or aligned translated documents. It contains around 3 millions of documents translated to 24 languages (DA, DE, EN, ES, FI, FR, EL, IT, NL, PT, SV, CS, ET, HU, LT, LV, MT, PL, SK, SL, GA, BG, RO, HR). This dataset can be used as a benchmark for supervised classification or for training machine learning models applied to business intelligence application.

2. The second sub-dataset, par-TED, consists of the aligned sentences of translated texts extracted from the fd-TED corpus. It can be used for machine assisted translation of juridical and technical documents, or the extraction of multilingual terminology. This corpus is composed with 4 millions of unique sentences translated to at least 23 languages.

The two sub-datasets, fd-TED and par-TED, will be updated in a regular basis to keep tracks of the new calls for tender published by the EU states.

We also provide an API, to download the new updates and to support an easy access to the data. This is done through the use of filters that can be applied on the meta data, basically the language(s), the hierarchical level(s) of the Common Procurement Vocabulary (CPV) codes, the type of processed texts, and so on.

2. The (full-document) fd-TED corpus

The fd-TED corpus is built from the full content of the documents extracted from the TED platform. Each document of the corpus belongs to a hierarchy that is succinctly described below.

[1]http://ted.europa.eu/TED/main/HomePage.do
[2]`https://github.com/oussamaahmia/TED-dataset`

Level in the hierarchy	1	2	3	4	5	6	7	8	9
Count	1,868,420	433,111	231,167	144,393	115,487	45,656	30,792	21,694	16,727
Cumulative	2,907,447	1,039,027	605,916	374,749	230,356	114,869	69,213	38,421	16,727

Table 1: Number of documents for each level of the CPV code

Language	DE-ES	DE-IT	EN-DE	EN-ES	EN-FR	EN-IT	FR-DE	FR-ES	FR-IT	IT-ES
Count	425,797	428,097	425,893	425,808	426,027	425,856	429,039	425,797	425,803	425,797

Table 2: Number of documents fully translated for some pairs of languages.

2.1. Common Procurement Vocabulary

Common Procurement Vocabulary (CPV) [3] is the "thesaurus" that defines the subject matter of public contracts, allowing companies to easily find public procurement notices according to their areas of expertise. The main CPV vocabulary is based on a hierarchical structure (a tree structure) comprising codes of up to 9 digits (the ninth digit serves to check the previous digits). The CPV code consists of 8 digits that encodes 5 hierarchical subdivisions as follows:

1. The first two digits identify the divisions (XX000000-Y), e.g. "industrial machinery".
2. The first three digits identify the groups (XXX00000-Y), e.g. "Machine tools".
3. The first four digits identify the classes (XXXX0000-Y), e.g. Metal-working machine tools.
4. The first five digits identify the categories (XXXXX000-Y), e.g. "Hydraulic presses".
5. Each of the last three digits gives a greater degree of precision within each category.

For Example:
42000000 is the code for "industrial machinery", 42600000 is for "Machine tools", 42630000 for "Metal-working machine tools" and 4263600 is for "Hydraulic presses".
Table 1 presents the number of documents for each level of the CPV codes by taking into account the last hierarchical level (the 8 digits of the CPV code)

2.2. The documents

The documents are published in 24 languages of the EU. They can be fully translated to the 24 languages (Table 2 and 3.) or partially translated (in most of the cases the object of the document and the lots [4] are translated).
The dataset that we provide is presented as a multilingual corpus that can be exploited for supervised hierarchical classification or Cross-Language Text Classification (Olsson et al., 2005).
The XML schema comes in different versions (R2.0.9 and R2.0.8), hence the needed fields are extracted using the parser corresponding to each version. Then the CPV codes are corrected if additional characters are found. The descriptions fields (named "desc") are created from the aggregation of several XML elements that are checked for administrative sentences using a classifier trained with manually tagged dataset by Jurismarches[5] experts (the accuracy of the classifier is 98%). The raw text is created for each available language by converting the XML into text records. In case of any error in this step, the text is downloaded directly from the TED's website or converted using TED's online API. The filtered text is created by ignoring all the XML entities dealing with administrative information (some XML elements will always contain only administrative content) and filtering the mixed elements using the classifier to get rid of the administrative content.
Knowing that the procurement notices contain legal and administrative information that are not fundamental for understanding the core business of the consultation, as filtered sentences tends to introduce a lot of noise if the interest is upon valuable business informations present in call for tenders (conditions relating to the contract, deposits and guarantees required...).
With the help of experts in public markets (Jurismarches, we provide a filtered version of each document that only contains the description of the supplies.
Example of core business information:

- Installation of doors and windows and related components.

Example of legal and administrative information that has been filtered out:

- Candidates (all partners in the case of a consortium) shall prove that they have the legal capacity to perform the contract by providing (...)

The data structure of the documents contained in the fd-TED corpus is presented in Figure 1.

2.3. Classification Example

As an example of supervised classification, Table 3 shows the results of a classification using Linear Support Vector Machine (SVM) (Cortes and Vapnik, 1995) and a bag of words representation. From a random sub-sample of 200K English and French documents extracted from the fd-TED corpus, we randomly split our data into into 75%

[3] COMMISSION REGULATION (EC) No 213/2008 of 28 November 2007

[4] Tenders are generally advertised with a global title, the object, and some of them are divided into lots, each having it's own title

[5] https://www.jurismarches.com/

```
{"ref":0000-0000 #The document ID in the TED database.
 "origin_ln":"" #The original language of the document.
 "list_ln": [] #the list of languages in which the document is translated.

 "document":{
     "EN":
         {"title":"Document Title" #The title of the document.
          "CPV":['00000000'] #The list of CPVs codes of the document
          "desc": "description of the project"
                       #additional information about the project.
          "lots": [ #list of the parts of the project.
                    "title":"Title of the lot"
                    "CPV": ['00000000'] #the CPVs codes of the lot.
                    "desc": "description of the lot"
                         #additional information about the lot.
          ]
          "raw": "the raw text of the document" # full text
          "filtered": "the processed document" # the text without the
          administrative information.

         }
 }}
```

Figure 1: Format of the documents for the processed dataset fd-TED.

for training and 25% for testing. We have used for this experiment the first hierarchical levels of the CPV codes, namely the two first digits.

The combination of the model trained on the English version of the documents and the French one using a max rule (Kittler et al., 1998) increases significantly the accuracy of this classification task.

Language	Accuracy
FR	59%
EN	65%
EN+FR	68%

Table 3: SVM classification results

3. The (parallel) par-TED corpus

Alongside with the fd-TED corpus, we provide a multilingual aligned corpus in the form of a set of parallel sentences with at least 1.2 million unique sentences translated to at least 23 languages. This corpus is created by aligning the XML trees for each language. Some XML elements are ignored (such as Phone numbers, email, addresses, etc). Then the repeated sentences are deleted.
Below is an example of aligned sentences for the EN,FR,ES,and IT languages.

- FR: Travaux de finition et de rénovation pour le complexe tokamak, le bâtiment d'assemblage et tous les bâtiments voisins.
- EN: Finishing and retrofit works for the Tokamak complex, assembly hall and all surrounding buildings.
- ES: Obras de modernización y finalización del complejo y taller de montaje del Tokamak y de los edificios colindantes.
- IT: Lavori di rifinitura e di ammodernamento per il complesso Tokamak, il reparto di assemblaggio e tutti gli edifici circostanti.

The data structure for the par-TED corpus is presented in Figure 2.
As an example, we have built word embeddings for the EN and FR languages to show the potentiality of this corpus in a multilingual terminology extraction application.
From Table 4 and Table 5 we can see that using a cosine similarity on Word2Vec representations (Mikolov et al., 2013) built on this corpus, we get comparable results regarding the word similarity on excerpts of common and proper nouns for the two tested languages.

4. Conclusion

The "Tenders Electronic Daily" (TED) is a large source of semi-structured and multilingual document widely underused by the NLP community, mainly due to the burden and costs associated to the collecting and formatting of the data. To ease the exploitation of this resource either for text mining or machine translation tasks, we have presented a packaging of these data (along with a Python API to access it) that can be freely downloaded and used by scientists and engineers to benchmark or solve some of their NLP problems. Few toy application examples have been also detailed to highlight the usefulness of this this resource and the type of services it can provide.

```
{"ref":0000-0000 #The document ID in the TED database.
 "origin_ln':"" #The language of the source document.
 "sent_id": #Sentence id in the document.

 "sentences":{ #List of the translations.
     "EN":"...",
     "FR":"...",
     "ES":"...",
     ...
          }
 }}
```

Figure 2: Format of the documents in the par-TED corpus.

Word	Similar Words	Similarity	Word	Similar Words	Similarity	Word	Similar Words	Similarity
Linux	citrix	0.81	Twitter	facebook	0.85	Lawyer + Advice	legal	0.70
	unix	0.81		social media	0.82		matters	0.69
	server	0.80		blogs	0.78		legal matters	0.68
	vmware	0.80		web chat	0.69		advisers	0.66
	microsoft	0.78		press releases	0.68		disputes	0.66
	windows server	0.77		youtube	0.67		matters arising	0.65
	weblogic	0.77		newsletter	0.66		legal advice	0.65
	oracle	0.77		text messaging	0.65		specific issues	0.65
	ms sql	0.77		direct mail	0.65		lawyers	0.65
	red hat	0.77		google	0.65		advice guidance	0.65

Table 4: Example of some most similar words using Word2Vec embedding and cosine similarity on English corpus

Word	Similar Words	Similarity	Word	Similar Words	Similarity	Word	Similar Words	Similarity
Linux	windows	0.85	Twitter	facebook	0.90	avocat + conseil	representation	0.75
	redhat	0.83		instagram	0.86		conseils	0.74
	unix	0.83		netflix	0.84		droit social	0.74
	mac os	0.82		snapchat	0.82		assistance juridique	0.73
	citrix	0.81		google	0.81		avocats	0.73
	serveurs	0.80		tweets	0.80		conseils juridique	0.72
	microsoft	0.79		youtube	0.80		representation juridique	0.72
	ibm	0.79		linkedin	0.77		contentieux	0.72
	windows server	0.79		maddyness	0.77		representation devant	0.71
	env windows	0.79		tweet	0.77		conseil juridique	0.71

Table 5: Example of some most similar words using Word2Vec embedding and cosine similarity on French corpus

SL	SK	DE	EL
451.1K/1.2M/433.0K	452.0K/1.4M/491.6K	849.4K/6.4M/1.5M	461.5K/1.7M/526.7K
LT	GA	PT	FI
457.3K/1.6M/496.3K	425.8K/868.6K/359.9K	450.7K/1.0M/371.1K	472.2K/1.3M/696.6K
MT	SV	LV	HU
425.8K/913.8K/352.1K	499.8K/1.3M/592.8K	443.3K/1.1M/420.9K	457.4K/2.5M/691.8K
EN	DA	FR	ES
674.8K/4.2M/824.9K	461.3K/1.4M/545.6K	1.1M/11.4M/1.2M	560.3K/2.1M/510.0K
RO	PL	HR	IT
483.1K/3.5M/567.8K	739.8K/11.2M/988.6K	288.3K/765.3K/314.9K	544.9K/2.9M/677.3K
NL	CS	ET	BG
525.1K/2.0M/613.5K	527.3K/1.8M/563.4K	441.5K/1.1M/510.0K	485.3K/2.4M/540.6K

Table 6: Number of (fully translated documents/ unique sentences/ unique words) per language.

5. Bibliographical References

Cortes, C. and Vapnik, V. (1995). Support vector machine. *Machine learning*, 20(3):273–297.

Kittler, J., Hatef, M., Duin, R. P., and Matas, J. (1998). On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence*, 20(3):226–239.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Olsson, J. S., Oard, D. W., and Hajič, J. (2005). Cross-language text classification. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 645–646. ACM.

Using Adversarial Examples in Natural Language Processing

Petr Bělohlávek, Ondřej Plátek, Zdeněk Žabokrtský, Milan Straka
Institute of Formal and Applied Linguistics
Charles University, Faculty of Mathematics and Physics
Malostranské náměstí 25, Prague, Czech Republic
{belohlavek,oplatek,zabokrtsky,straka}@ufal.mff.cuni.cz

Abstract

Machine learning models have been providing promising results in many fields including natural language processing. These models are, nevertheless, prone to *adversarial examples*. These are artificially constructed examples which evince two main features: they resemble the real training data but they deceive already trained model. This paper investigates the effect of using adversarial examples during the training of recurrent neural networks whose text input is in the form of a sequence of word/character embeddings. The effects are studied on a compilation of eight NLP datasets whose interface was unified for quick experimenting. Based on the experiments and the dataset characteristics, we conclude that using the adversarial examples for NLP tasks that are modeled by recurrent neural networks provides a regularization effect and enables the training of models with greater number of parameters without overfitting. In addition, we discuss which combinations of datasets and model settings might benefit from the adversarial training the most.

Keywords: Neural networks, Adversarial examples, Natural Language Processing, Regularization, Evaluation

1. Introduction

In recent years, deep learning has outperformed many of other machine learning models in various tasks of natural language processing (Amodei et al., 2016; Wang et al., 2017; Yin et al., 2015; Collobert and Weston, 2008). Many of these models have been completely trained in end-to-end manner without any need for hand-crafting.

These models are usually very complex and tend to overfit easily, especially in cases of small datasets. For that reason, *regularization techniques* are often employed in order to prevent overfitting. A popular solution to the lack of data and to model overfitting is *dataset augmentation* (Simard et al., 2003). This technique automatically generates similar training instances to the ones already present in the dataset, which effectively results in the dataset size increase. While augmenting the visual data is straightforward, the augmentation of text data is non-trivial.

Lately, a novel method for creating so called adversarial examples was introduced Goodfellow et al. (2014b; Szegedy et al. (2013). These examples *reveal* that the models do not fulfil the smoothness assumption, i.e. the adversarial examples are very similar to the examples in the training dataset, but the already trained models classify them differently than the very similar ones in the training data. When generating such examples during training and employing them in the process of model parameters update, they function as strong regularization. Generating adversarial examples can be understood as a form of dataset augmentation.

The aim of this paper is to evaluate the effect of using the adversarial examples during training of the deep recurrent neural networks which process natural language in the form of written text. More specifically, we intend to perturb trainable word/character embeddings[1] (Mikolov et al., 2013) in a way the network is confused, although the new embeddings are extremely close to the original ones.

[1]The vector representations of the text tokens which provide a unified input structure for machine learning models.

Our contribution is threefold:

- We prepared, selected and preprocessed a collection of eight distinct NLP datasets.
- Employing the collected datasets, we evaluate the effects of using the adversarial examples during the deep learning model training. We include significance tests in order to support our claims.
- We discuss which datasets in general can benefit most from the adversarial examples.

2. Related Work

Neural networks are commonly regularized by a handful of standard techniques. Apart from traditional *shrinkage techniques* such as L_2 and L_1 regularization, the most used regularization technique is *dropout* (Hinton et al., 2012) which randomly selects a subset of neurons that are disabled in a following training iteration.

Additional techniques include *batch normalization* (Ioffe and Szegedy, 2015) and *layer normalization* (Ba et al., 2016) which normalize the layer activation. Notably, recurrent neural networks greatly benefit from the latter approach. Further techniques include *weight normalization* (Salimans and Kingma, 2016), *batch renormalization* (Ioffe, 2017) and *self normalizing networks* (Klambauer et al., 2017).

Adversarial examples have been mostly studied in the context of image processing, especially for image classification. Goodfellow et al. (2014b), by following the work of Szegedy et al. (2013), show that not only highly non-linear models such as deep neural networks have adversarial examples, but also much more linear and simpler models such as logistic regression do too.

Moosavi-Dezfooli et al. (2016a) introduce a heuristic technique which develops such a perturbation that, when applied to almost any image in the dataset, misleads the model. They name this established perturbation as *universal*. In addition, the authors discover that this perturbation

affects various models trained on the same dataset. Therefore, the universality is *twofold*.

Nguyen et al. (2015) use a different approach by generating the adversarial examples by an *evolution algorithm* instead of by using gradient descent. This approach provides adversarial examples even in the environments in which the neural network cannot be trained via back-propagation, since the target variables remain unknown.

Vidnerová and Neruda (2016) propose a *genetic algorithm* for adversarial example creation. They study robustness and generalization power of various models including both deep and shallow neural networks, support vector machines (SVM) even with radial basis function kernel (RBF) and decision trees.

Other noteworthy works regarding the adversarial examples in image processing include the work of Fawzi et al. (2016), Moosavi-Dezfooli et al. (2016b), Papernot et al. (2016b) and Papernot et al. (2016a).

Regarding adversarial examples used in the context of NLP, there is much less published research. Caswell et al. (2016) have been, to our knowledge, the first who experimented with using *adversarial embedding perturbations*. The IMDB dataset (Maas et al., 2011) was used as a binary sentiment classification task. The authors introduced visualization techniques in order to understand the effect of constructed adversarial embeddings. Nonetheless, the results of their experiments did not support any hypothesis of the method usefulness.

Miyato et al. (2016b) were the first who constructed adversarial perturbations of word embeddings. In addition, the authors employed a so-called *virtual adversarial training* which introduces a new loss term based on KL-divergence (Miyato et al., 2016b, Eq. 3,4). The authors evaluated five data sources and both adversarial and virtual adversarial learning outperformed the state-of-the-art models.[2]

Jia and Liang (2017) used *adversarial generation* of the source text instead of embeddings. They introduced various methods fooling the models for Standford Question Answering Dataset (Rajpurkar et al., 2016) by generating additional sentences to the original source text.

Lu et al. (2017) aimed for *adversarial examples detection* in the context of Generative Adversarial Networks (GANs) (Goodfellow et al., 2014a). In addition, Miyato et al. (2016a) employed RNN-based GANs to semi-supervised text classification.

Adversarial examples are often created as a perturbation of the original input. Goodfellow et al. (2014b) and Szegedy et al. (2013) work with a deterministic perturbation which "damages" the current model the most. Contrary to them, using random perturbations has various theoretical implications and is commonly employed (Matsuoka, 1992; Bishop, 1995; Grandvalet et al., 1997).

3. Adversarial Examples

Adversarial examples are artificially constructed inputs that are very similar to some example from the training set, but they fool an already trained model. Adversarial examples can be employed during the model training in order to supply new training data from which the model might benefit.

Let us assume a classification task. Given a trained model representing function f and a training example $(\boldsymbol{x}, y)$, the adversarial example is such $\boldsymbol{x} + \boldsymbol{\Delta}$ which fools the model, i.e. $f(\boldsymbol{x}) = y$ while $f(\boldsymbol{x} + \boldsymbol{\Delta}) \neq y$. The difference $\boldsymbol{\Delta}$ is called the *perturbation* and its magnitude can be limited subject to some metric. In our case, $\boldsymbol{\Delta}$ is a deterministic perturbation which is constructed subject to the gradient updates of the network.

We focus on NLP tasks in which the input is encoded as a sequence of word/character embeddings. The perturbation $\boldsymbol{\Delta}$ slightly modifies the embeddings of the input sequence.

The described process can be understood as a modification of the original example text, however, the final perturbed embedding does not correspond to a particular word (besides exceptional cases), since the perturbations are arbitrary. Nevertheless, it is easy to find a nearest neighbor embedding and its corresponding word. Understandably, such nearest word does not necessarily behave the same as the perturbed embedding.

Since it is expected that the nearest neighbor is the original embedding without any perturbation applied, the adversarial examples in the context of NLP are difficult to interpret. One *possible intuition* is that the adversarial example replaces each word with a "new word" which is supposed to have a similar semantics to the original one, even though it is uninterpretable to human.

Apart from the proof of various machine learning models instability, Goodfellow et al. (2014b) demonstrated that employing the adversarial examples during the training of gradient-based models functions as a regularization technique whose effect is comparable to *dropout* (Hinton et al., 2012).

The main principle lies in *online* generation of adversarial examples throughout the training of the model. These generated examples are similar to the training instances, which might improve model smoothness and generalization, by enforcing the model to behave similarly on a similar input (Goodfellow et al., 2014b).

This modification of the training effectively extends the training data, hence it can be understood as an augmentation technique.

Given a training pair $(\boldsymbol{x}, y)$ and current model parameters $\boldsymbol{\Theta}$, the cost function J might be linearly approximated on the neighborhood of $\boldsymbol{\Theta}$.

Assuming the prior conditions, gradient $\nabla_{\boldsymbol{\Theta}} J(\boldsymbol{x}, y; \boldsymbol{\Theta})$ can be easily estimated (e.g. by back-propagation). Based on the explanation of (Goodfellow et al., 2014b), a possible way of perturbation computation is using the estimated gradients. Considering the fact that the gradient might be arbitrarily large and the perturbation is thought to be small, the authors approximate the gradient by a vector of $\{-1, 0, +1\}$ by applying the sign function. The total magnitude of the perturbation might be then controlled by a (small) multiplicative constant $\varepsilon > 0$. This approach was named *fast gradient sign method* and is defined by Equa-

[2] All experiments benefited from using a pretrained language model, which is a substantial difference from models used in our experiments.

tion 1.

$$\boldsymbol{\Delta} = \varepsilon \cdot \text{sign}\left(\nabla_{\boldsymbol{\Theta}} J(\boldsymbol{x}, y; \boldsymbol{\Theta})\right) \quad (1)$$

Goodfellow et al. (2014b) proposed a modification of the optimalization algorithm based on back-propagation which employs the adversarial examples.
The authors define a new loss function $\tilde{J}$ which regards both the original loss and the loss computed on an adversarial example (based on the currently processed pattern). The definition of $\tilde{J}$ for training pair $(\boldsymbol{x}, y)$ and parameters $\boldsymbol{\Theta}$ is given by Equation 2.

$$\begin{aligned}\tilde{J}(\boldsymbol{x}, y; \boldsymbol{\Theta}) &= \alpha J(\boldsymbol{x}, y; \boldsymbol{\Theta}) + (1-\alpha) J(\boldsymbol{x} + \boldsymbol{\Delta}, y; \boldsymbol{\Theta}) \\ &= \alpha J(\boldsymbol{x}, y; \boldsymbol{\Theta}) + (1-\alpha) \cdot \\ &\quad \cdot J(\boldsymbol{x} + \varepsilon \cdot \text{sign}(\nabla_{\boldsymbol{\Theta}} J(\boldsymbol{x}, y; \boldsymbol{\Theta})), y; \boldsymbol{\Theta})\end{aligned} \quad (2)$$

4. Datasets

A collection of eight datasets in total was selected in order to evaluate the adversarial training in the context of NLP. In order to provide further complexity of the evaluation, we selected datasets that differ in multiple characteristics. However, due to limited resources, the selected datasets do not cover the whole characteristic space.
All datasets were provided with a unified interface so that they could be easily experimented with.[3]
At first, we selected three of the Facebook bAbI datasets (Weston et al., 2015; Sukhbaatar et al., 2015). Each of the datasets models a different aspect of natural language understanding and reasoning. The typical dataset example contains a sequence of facts regarding position of various objects and a question testing the understanding and reasoning based on these facts. To be specific, we decided to choose the English versions of bAbI tasks #1, #2 and #6, because they represent both simple and advanced problems.
Secondly, we focused on subjectivity detection, i.e. distinguishing between subjective and objective texts. For this reason, we selected Movie Review Data (originally evaluated by Pang and Lee (2004)). This dataset contains short texts (typically single sentences), which were automatically mined either from the official movie description provided by the distributor or from the user reviews. It is assumed that the prior texts are objective and the latter subjective. This process of automatic mining naturally introduces a noise in the generated annotations, i.e. some objective sentences are probably classified as subjective and vise-versa.
In order to evaluate also non-English datasets, we selected three Czech/Slovak datasets. All of them aim to test sentiment analysis, i.e. the goal of the model is to distinguish among positive, negative, bipolar and neutral texts. The three selected datasets (Social Media Dataset, Movie Review Dataset and Product Review Dataset, all described by Habernal et al. (2013)) represent texts regarding various topics and were automatically collected from the internet. Their sentiment was automatically derived based on the rating assigned to the reviewed item (number of stars).
The final selected dataset is the one introduced as Discriminating between Similar Language (DSL) Shared Task 2015 at LT4VarDial - Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects in 2015 (Zampieri et al., 2015). The goal of the dataset is to demonstrate the language discrimination among 13 different languages and dialects and one additional category. The provided dataset contains approximately 200 thousand examples, which enable us to restrict the training set to a specified number of examples. We use the restricted versions of this dataset in order to evaluate the effects of various dataset sizes on adversarial training.
In order to evaluate the effect of adversarial examples, we consider the following dataset features that aim to cover distinct types of NLP datasets. We use the characteristics for discussion of the relation between the regularization effect of adversarial examples and the dataset features.

- *Dataset size*: number of the training examples.
- *Dataset language*: input text language.
- *Vocabulary size*: number of distinct tokens.
- *Text length*: distribution of the text lengths in the training dataset.
- *Annotation quality*: noise amount in the gold labels.
- *Input text structure*: distribution of the token frequency; number of sentences.
- *Dataset artificiality*: real-world dataset vs. artificially constructed one.

5. Evaluation

Each dataset was split into three distinct parts: training, validation and testing. The training part was used for the model parameter estimation, the validation for the best model selection and the testing for the final result reports.
We employ the testing losses in order to perform significance of the performance improvement.

5.1. Model Selection

The neural networks are typically trained in several epochs. After each epoch, the model loss is evaluated both on *training* and *validation* data. While the validation loss represents an independent estimation of the model generalization capability, both losses are used together for overfitting detection.
In order to select the epoch in which the model achieved the best performance, we select the one in which the *loss was the least* on the validation data. The final performance is then reported on *testing* dataset, which is completely held-out until the final model is selected.
Since the validation and test datasets were constructed by randomly choosing examples, the test performance on the model is *conditionally independent* on the validation performance given the model parameters. However, the test performance is expected to be slightly worse in comparison to the validation performance since the model was chosen to achieve the best validation performance regardless the test evaluation.
The use of *loss* instead of other metrics such as accuracy or F-measure is supported by the fact that the loss itself

[3] We implemented and employed deep learning framework cxflow (v0.4) for all our experiments (`https://cxflow.org`) as it supports quick experiment creation, logging and result management.

models the actual behavior of the model. In contrast, in a classification problem, the accuracy of the model does not express its behavior in a sufficient detail.

5.2. Testing Significance

In order to evaluate whether the adversarial training has a significantly positive effect on model performance, we perform the *paired t-test* on the testing losses as follows.

Comparing two experiments given a single dataset, we train appropriate network architectures independently with the same train/valid/test data split. Then, we select the best models using the validation dataset. Finally, we evaluate and compare the testing losses between the corresponding testing examples. This leads to a paired t-test using these two sets of losses pairs.

5.3. Experiment Setup

A model processes tokenized[4] text with special tokens indicating the beginning and the ending of a sequence. In case there is a token in the validation or testing dataset which was not present in the training dataset, it is replaced with the *unknown token* symbol. In order to adapt to texts containing such unknown tokens, the tokens used for training are usually uniformly randomly replaced with the unknown token during each training epoch. The uniformity is employed since many of the datasets were collected automatically and contain various typing errors, which are believed to be distributed uniformly.

Afterwards, each token is translated into its corresponding embedding. The embeddings are uniformly randomly initialized without any pretraining and being updated during the model training. In some experiments, we apply dropout with 50% keep probability to the embeddings.

Once the sequence of embeddings is constructed, a recurrent cell is applied to it. We employ either GRU (Cho et al., 2014) or LSTM (Hochreiter and Schmidhuber, 1997) cells in the following experiments. We use the last output of the cell (the concatenation of both last outputs in the case of bi-directional RNN) as the input to a multilayer perceptron (MLP) which contains a single hidden layer. In some experiments, we apply apply dropout with 50% keep probability to the hidden layer of the MLP.

By using a fully connected layer, we transform the output vector cell so that the final dimension matches the desired number of output neurons.

5.4. Model Training

The final output vector is trained to minimize *categorical cross-entropy* error function, in which the index of the target class is provided as the ground truth. The models are trained via mini-batch gradient descent with Adam gradient update (Kingma and Ba, 2014). The learning rates have been chosen from the interval 0.001–0.003 without major performance influence.

We experimented also with additional loss functions such as mean-square error applied to the sigmoid of the final output vector and one-hot encoded ground truth, however, we observed no major difference.

[4]The text is tokenized by the popular tokenized *twokenize* (`https://github.com/brendano/tweetmotif`).

5.5. Results

During the evaluation, we focus on the testing loss improvement in comparison with the original vanilla model. We do not aim to compare our results with state-of-the-art results that usually employ additional (possibly hand-crafted) techniques during the training. All our results are presented in Table 1.

At least two models differing in the number of parameters (referred to as the big and small models) were trained for each experiment. Throughout our experiments, the smaller models tend to underfit the given task and the use of adversarial examples had no effect. In contrast, the bigger models tend to overfit massively which made regularization methods such as dropout or the adversarial examples effective. In the following text, we report only the results of the big models.

In addition, we experimented with both LSTM and GRU recurrent cells which, to our knowledge, did not affect the overall model performance. The presented results represent the models which employ bidirectional GRU recurrent cells.

All experiments compare four models (see Table 1):

1. the vanilla model (without any regularization),
2. the same model trained with dropout (embedding layer, MLP hidden layer),
3. the same model trained with adversarial examples,
4. the same model trained with both adversarial examples and dropout.

We start by evaluating the bAbI datasets, namely bAbI #1, #2 and #6. In all bAbI experiments we use a model which employs embeddings of dimension 100, which we experimentally discovered to be sufficient (increasing the dimension does not affect training). The GRU dimension is set to 400 and MLP hidden layer dimension is set to 512.

Focusing mainly on the test performance which represents the actual model generalization, we observe that the employment of the adversarial perturbations together with dropout provides approximately 22% accuracy improvement[5] in the case of bAbI #6. On the contrary, the employment of dropout provides only approximately 10% accuracy gain.

In contrast with the previous experiment, the employment of the adversarial examples damaged the model performance in the case of bAbI #1. However, the dataset accuracy benefited from using both adversarial training and dropout by approximately 4%.

For both experiments, we tested the testing loss difference between the vanilla model and the one employing the adversarial training. All measurements indicate that the testing losses significantly differ at the significance level $\alpha = 0.05$, which was set in advance (p-values 0.003 and 0.001, respectively).

Both datasets outperformed the baselines set by Weston et al. (2015) when evaluating the weakly supervised cases.

The final selected bAbI dataset (#2) which represents more

[5]All reported improvements are *absolute*. The testing accuracies and losses are presented in Table 1. The baseline provided by Weston et al. (2015) yielded 48% when using the weakly supervised LSTM.

Task Type	Dataset	Vanilla		Dropout		Adversarial Examples		Advers. examples. + Dropout	
		Perf	**Loss**	**Perf**	**Loss**	**Perf**	**Loss**	**Perf**	**Loss**
Question Reasoning	**bAbI #1**	73.8%	0.592	73.8%	0.603	68.3%	0.925	77.5%	**0.528**
	bAbI #2	30.8%	1.408	30.6%	1.435	30.8%	**1.399**	29.9%	1.428
	bAbI #6	47.9%	0.694	56.5%	0.662	67.2%	0.636	69.8%	**0.585**
Subjectivity Detection	**Movie Review**	64.4%	**0.600**	61.6%	0.617	63.5%	0.619	64.7%	0.606
Cz/Sk Sentiment Analysis	**Social Media**	42.5% (F1)	1.012	42.4%	1.024	43.0% (F1)	**1.010**	42.8% (F1)	**1.010**
	Movie Review	41.9% (F1)	**1.071**	41.4%	1.076	42.0% (F1)	**1.071**	40.6% (F1)	1.076
	Product Review	74.3%	**0.606**	73.4%	0.619	72.1%	0.725	57.7%	1.098
Language Detection	**DSL 10k**	55.3%	0.840	55.9%	0.891	57.0%	**0.812**	53.0%	0.863
	DSL 70k	71.8%	0.594	63.8%	0.694	70.8%	**0.579**	70.4%	0.611
	DSL 130k	73.4%	**0.539**	71.6%	0.567	66.7%	0.620	66.6%	0.600
	DSL 190k	77.8%	0.453	75.1	0.509	78.6%	**0.444**	67.4%	0.579

Table 1: The first and second columns describe the task type and the dataset name, respectively. The final four columns demonstrate the testing accuracies or F1-scores and the mean testing losses. The bold formatting indicates the loss which is the best among the dataset experiment.

challenging questions has not significantly benefited from the employment of the adversarial examples.

The next evaluated dataset is the Movie Review dataset for subjectivity detection. We use a model which employs embeddings of dimension 200; the GRU dimension is set to 200 and MLP hidden layer dimension is set to 256. We observe that employing both adversarial examples and dropout during training stabilizes it. Approximately 0.3% increase in accuracy is achieved, however, the testing loss is significantly smaller (p-value 0.039) than the testing loss yielded by the model with only dropout. We suggest that the model is more confident of its responses.

Then, we evaluated the three Czech/Slovak sentiment analysis datasets. We used all four classes (positive, negative, neutral, bi-polar) and trained models using the their phonetic transcriptions. We have experimented also with stemming instead of phonetic transcriptions, however, the performance was poor and all models tended to overfit in early epochs.

The Social Media dataset benefited from the adversarial training by 0.05% increase in F1-score and the testing loss significantly decreases either with or without using dropout (p-values 0.009 and 0.026, respectively). We observe similar gain in F1-score (+0.15%) in the case of omitting the bi-polar class. For these experiments we employ embeddings of dimension 200; the GRU dimension is set to 400 and MLP hidden layer dimension is set to 512.

The sentiment analysis dataset (Movie Review) has not significantly benefited from the adversarial examples.

The performance of the final sentiment analysis dataset (Product Review) surprisingly decreased in all our experiments when using the adversarial examples even though it is similar to the previous two datasets. The main difference is the size of the dataset as the Product Reviews contains much more training examples. We experiment also with character-level models (presented in Table 1) for which we employ embeddings of dimension 15; the GRU dimension is set to 400 and MLP hidden layer dimension is set to 512.

The last dataset that we evaluate is the Discriminating between Similar Language (DSL) dataset which we attempt to model also on a character level. We evaluated this dataset multiple-times with various training set size limits (10, 70, 130 and 170 thousand of examples).

In all DSL experiments we use a model which employs embeddings of dimension 20. The GRU dimension is set to 200 and MLP hidden layer dimension is set to 256.

The results of the experiments which were trained using the smallest training set and regularized by adversarial examples indicate that the use of adversarial examples significantly improves the training loss (relative drop of 4%). At the same time, the model accuracy is improved as well. Contrary to this observation, the experiments with the larger training sets have not benefited in accuracy when employing adversarial examples either with or without dropout. Nevertheless, the average testing losses have (except the 130k training set) significantly decreased. We conclude that the greater the training set is, the smaller impact the adversarial training makes.

6. Discussion

The conducted experiments lead us to the following claims. Firstly, we observe that the employment of adversarial examples improves the model loss on majority of datasets of our collection.[6]

We conclude that the models whose capacity is sufficient so that they overfit the training data (the number of their parameters is large) can benefit from using adversarial examples. In these cases, we claim that employing adversarial examples during training acts as a regularization technique. Furthermore, our preliminary hypotheses on the trends between the dataset characteristics and the effect of using the adversarial examples are as follows. Note that a wider collection of datasets would be required to make our propositions more credible.

English datasets exhibited better performance from the adversarial examples than the Czech or Slovak ones. We hypothesize that this effect might be caused by the fact that the Slavic languages feature a more complex morphology

[6] We support the claim of Caswell et al. (2016) and Miyato et al. (2016b) who suggest that the use of adversarial examples during the training might be beneficial even for recurrent networks.

and a greater number of distinct words (tokens). Nevertheless, even though strict stemming was employed, we did not achieve such performance improvement compared to the one we have observed in other English datasets.

Furthermore, we claim that the networks trained on artificial data (such as bAbI tasks (Weston et al., 2015)) with a rather small number of distinct tokens are more likely to be *vulnerable* to adversarial examples and their use during training prevents the model from overfitting particularly efficiently.

Based on our experiments, we observe that the impact of the studied technique decreases as the size of the training dataset increases. We conclude that this phenomenon supports the hypothesis that adversarial training serves as a regularization technique, which is usually less effective in case more training data is available.

We provide novel evidence that the adversarial perturbation of character embeddings can also lead to performance improvement when used in the process of training character-level models.

7. Conclusion

This paper focuses on employing adversarial examples during training of deep recurrent neural networks. The contribution of this paper is threefold.

Firstly, we compiled a collection of eight datasets for which a unified stream interface was created. The datasets were chosen in order to represent a distinct set of characteristics which were chosen in advance.

Secondly, the employment of adversarial examples during the network training was evaluated in various settings. We focused on embeddings which are randomly initialized at the beginning of the training.

Finally, we have proposed several hypotheses about relations between the dataset characteristics and the effect of model training when using adversarial examples. In some cases, we outperformed the baselines provided by the authors of the dataset publications.

We conclude that the adversarial training can function as a regularization for RNNs processing natural text. This is a direct extension of the work of Miyato et al. (2016a) who have experimented with pretrained embeddings.

7.1. Future Work

We consider the following possible research directions in our future work regarding the adversarial examples.

Firstly, additional datasets might be evaluated in order to provide more detailed study. In particular, tasks such as *machine translation*, which do not aim to classify the input, could be studied. For this purpose, supplementary architectures such as *sequence-to-sequence models* shall be evaluated.

Secondly, our further research will focus on the embedding structure analysis, primarily on the changes caused by employing the adversarial examples. Even though we have not been able to prove a significant change in their structure yet, we hope the interpretation of the embedding differences might be found. In addition, we will study the stability of training when using adversarial examples.

Furthermore, supplementary techniques for adversarial example creation will be analyzed, notably Generative Adversarial Networks (Goodfellow et al., 2014a). Another approach could employ genetic algorithms for embedding perturbations.

Finally, additional types of machine learning, such as (deep) reinforcement learning (RL) could be examined. Focusing on NLP, the end-to-end neural dialogue systems are suitable for further analysis.

8. Acknowledgements

The research described herein has been supported by the Grant No. DG16P02B048 of the Ministry of Culture of the Czech Republic. Data has been used from and stored into the LINDAT/CLARIN repository, a large research infrastructure supported by the Ministry of Education, Youth and Sports of the Czech Republic under projects LM2015071 and CZ.02.1.01/0.0/0.0/16_013/0001781.

Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

9. Bibliographical References

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Cheng, Q., Chen, G., et al. (2016). Deep speech 2: End-to-end Speech Recognition in English and Mandarin. In *International Conference on Machine Learning*, pages 173–182.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer Normalization. *arXiv preprint arXiv:1607.06450*.

Bishop, C. M. (1995). Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116.

Caswell, I., Nie, A., and Sen, O. (2016). Exploring Adversarial Learning on Neural Network Models for Text Classification. *Standford NLP Project*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation. *Conference on Empirical Methods in Natural Language Processing*.

Collobert, R. and Weston, J. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Fawzi, A., Moosavi-Dezfooli, S.-M., and Frossard, P. (2016). Robustness of Classifiers: From Adversarial to Random Noise. In *Advances in Neural Information Processing Systems*, pages 1624–1632.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014a). Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014b). Explaining and Harnessing Adversarial Examples. *International Conference on Learning Representations*.

Grandvalet, Y., Canu, S., and Boucheron, S. (1997). Noise injection: Theoretical prospects. *Neural Computation*, 9(5):1093–1108.

Habernal, I., Ptáček, T., and Steinberger, J. (2013). Sentiment Analysis in Czech Social Media Using Supervised Machine Learning. In *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 65–74.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. *Journal of Machine Learning Research*.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural computation*, 9(8):1735–1780.

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ICML*.

Ioffe, S. (2017). Batch renormalization: Towards reducing minibatch dependence in batch-normalized models. *arXiv preprint arXiv:1702.03275*.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning*, volume 6. Springer.

Jia, R. and Liang, P. (2017). Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2011–2021.

Kingma, D. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S. (2017). Self-normalizing neural networks. *arXiv preprint arXiv:1706.02515*.

Lu, J., Issaranon, T., and Forsyth, D. (2017). Safetynet: Detecting and Rejecting Adversarial Examples Robustly. *arXiv preprint arXiv:1704.00103*.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 142–150. Association for Computational Linguistics.

Matsuoka, K. (1992). Noise injection into inputs in back-propagation learning. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):436–440.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *ICLR Workshop*.

Miyato, T., Dai, A. M., and Goodfellow, I. (2016a). Adversarial Training Methods for Semi-Supervised Text Classification. *arXiv preprint arXiv:1605.07725*.

Miyato, T., Dai, A. M., and Goodfellow, I. (2016b). Virtual Adversarial Training for Semi-Supervised Text Classification. *arXiv preprint arXiv:1605.07725*.

Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., and Frossard, P. (2016a). Universal Adversarial Perturbations. *arXiv preprint arXiv:1610.08401*.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. (2016b). Deepfool: A Simple and Accurate Method to Fool Deep Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2574–2582.

Nguyen, A., Yosinski, J., and Clune, J. (2015). Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436.

Pang, B. and Lee, L. (2004). A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2016a). Practical Black-box Attacks Against Deep Learning Systems Using Adversarial Examples. *arXiv preprint arXiv:1602.02697*.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. (2016b). The Limitations of Deep Learning in Adversarial Settings. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 372–387. IEEE.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Salimans, T. and Kingma, D. P. (2016). Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 901–909.

Simard, P. Y., Steinkraus, D., Platt, J. C., et al. (2003). Best practices for convolutional neural networks applied to visual document analysis. In *ICDAR*, volume 3, pages 958–962.

Sukhbaatar, S., Weston, J., Fergus, R., et al. (2015). End-to-end Memory Networks. In *Advances in neural information processing systems*, pages 2440–2448.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing Properties of Neural Networks. *International Conference on Learning Representations*.

Vidnerová, P. and Neruda, R. (2016). Evolutionary Generation of Adversarial Examples for Deep and Shallow Machine Learning Models. In *Proceedings of the The 3rd Multidisciplinary International Social Networks Conference on SocialInformatics 2016, Data Science 2016*, page 43. ACM.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., et al. (2017). TACOTRON: Towardss End-to-end Speech Synthesis. *arXiv preprint arXiv:1703.10135*.

Weston, J., Bordes, A., Chopra, S., Rush, A. M., van Merriënboer, B., Joulin, A., and Mikolov, T. (2015). Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks. *arXiv preprint arXiv:1502.05698*.

Yin, J., Jiang, X., Lu, Z., Shang, L., Li, H., and Li, X. (2015). Neural Generative Question Answering. *arXiv preprint arXiv:1512.01337*.

Zampieri, M., Tan, L., Ljubešic, N., Tiedemann, J., and Nakov, P. (2015). Overview of the dsl shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 1–9.

Determining Trolling in Textual Comments

Luis Gerardo Mojica de la Vega and Vincent Ng
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688, USA
{mojica,vince}@hlt.utdallas.edu

Abstract

Social media websites, electronic newspapers and Internet forums allow visitors to leave comments for others to read and interact. This exchange is not free from participants with malicious intentions, who *troll* others by positing messages that are intended to be provocative, offensive, or menacing. With the goal of facilitating the computational modeling of trolling, we propose a trolling categorization that is novel in the sense that it allows comment-based analysis from both the trolls' and the responders' perspectives, characterizing these two perspectives using four aspects, namely, the troll's intention and his intention disclosure, as well as the responder's interpretation of the troll's intention and her response strategy. Using this categorization, we annotate and release a dataset containing excerpts of Reddit conversations involving suspected trolls and their interactions with other users. For the final version of the paper, we plan to identify the difficult-to-classify cases in our corpus and suggest potential solutions for them.

Keywords: social media, text categorization, trolling, cyberbullying

1. Introduction

In contrast to traditional content distribution channels like television, radio and newspapers, Internet opened the door for direct interaction between the content creator and its audience. Young people are now gaining more frequent access to online, networked media. Although most of the time, their Internet use is harmless, there are some risks associated with these online activities, such as the use of social networking sites (e.g., Twitter, Facebook, Reddit). The anonymity and freedom provided by social networks makes them vulnerable to threatening situations on the Web, such as *trolling*.

Trolling is "the activity of posting messages via a communications network that are intended to be provocative, offensive or menacing" (Bishop, 2013). People who post such comments are known as *trolls*. According to Hardaker (2010), a troll's "real intention(s) is/are to cause disruption and/or trigger or exacerbate conflict for the purpose of their own amusement". Worse still, the troll's comments may have a negative psychological impact on his target/victim and possibly others who participated in the same conversation. It is therefore imperative to identify such comments and perhaps even terminate the conversation before it evolves into something psychologically disruptive for the participants. Monitoring conversations is a labor-intensive task: it can potentially place a severe burden on the moderators, and it may not be an effective solution when traffic is heavy. This calls for the need to develop automatic methods for identifying malicious comments, which we will refer to as *trolling attempts*.

In fact, there have recently been some attempts to automatically identify comments containing *cyberbullying* (e.g., Van Hee et al. (2015)), which corresponds to the most severe cases of trolling (Bishop, 2013). However, we believe that it is important not only to identify trolling attempts, but also comments that could have a negative psychological impact on their recipients. As an example, consider the situation where a commenter posts a comment with the goal of amusing others. However, it is conceivable that not everybody would be aware of these playful intentions, and these people may disagree or dislike the mocking comments and take them as inappropriate, prompting a negative reaction or psychological impact on themselves.

In light of this discussion, we believe that there is a need to identify not only the trolling attempts, but also comments that could have a negative psychological impact on its receipts. To this end, we seek to achieve the following goals in this paper. First, we propose a comprehensive categorization of trolling that allows us to model not only the troll's intention given his trolling attempt, but also the recipients' *perception* of the troll's intention and subsequently their reaction to the trolling attempt. This categorization gives rise to very interesting problems in pragmatics that involve the computational modeling of intentions, perceived intentions, and reactions to perceived intentions. Second, we create a new annotated resource for computational modeling of trolling. Each instance in this resource corresponds to a *suspected* trolling attempt taken from a Reddit conversation, it's surrounding context, and its immediate responses and will be manually coded with information such as the troll's intention and the recipients' reactions using our proposed categorization of trolling. For the final version of the paper, we plan to identify the instances that are difficult to classify with the help of a classifier trained with features taken from the state of the art and present an analysis of these instances.

To our knowledge, our annotated resource is the first one of its sort that allows computational modeling on both the troll's side and the recipients' side. By making it publicly available, we hope to stimulate further research on this task. We believe that it will be valuable to any researcher who is interested in the computational modeling of trolling.

2. Related Work

In this section, we discuss related work in the areas of trolling, bullying, abusive language detection and polite-

ness, as they partially address the problem presented in this work.

In the realm of psychology, Bishop (2013; 2014) elaborate a deep description of a troll's personality, motivations, effects on the community that trolls interfere in and the criminal and psychological aspects of trolls. Their main focus are flaming (trolls), and hostile and aggressive interactions between users (O'sullivan and Flanagin, 2003).

On the computational side, Mihaylov et al. (2015b; 2015a; 2016) address the problem of identifying opinion manipulation trolls, including paid trolls in news community forums. Not only do they focus solely on troll identification, but the major difference with this work is that all their predictions are based on non-linguistic information such as number of votes, dates, number of comments and so on. In a networks related framework, Kumar et al. (2014) and Guha et al. (2004) present a methodology to identify malicious individuals in a network based solely on the network's properties rather than on the textual content of comments. Cambria et al. (2010) propose a method that involves NLP components, but fail to provide an evaluation of their system.

There is extensive work on detecting offensive and abusive language in social media (Nobata et al., 2016; Xiang et al., 2012). There are two clear differences between their work and ours. One is that trolling is concerned about not only abusive language but also a much larger range of language styles and addresses the intentions and interpretations of the commenters, which goes beyond the linguistic dimension. The other is that we are additionally interested in the reactions to trolling attempts, real or perceived, because we argued that this is a phenomenon that occurs in pairs through the interaction of at least two individuals, which is different from abusive language detection. Also, Xu et al. (2012a; 2012b; 2013) address bullying traces. Bullying traces are self-reported events of individuals describing being part of bullying events, but we believe that the real impact of computational trolling research is not on analyzing retrospective incidents, but on analyzing real-time conversations. Chen et al. (2012) use lexical and semantic features to determine sentence offensiveness levels to identify cyberbullying, offensive or abusive comments on Youtube. On Youtube as well, Dinakar et al. (2012) identified sensitive topics for cyberbullying. Dadvar et al. (2014) used expert systems to classify between bullying and no bullying in posts. Van Hee et al. (2015) predict fine-grained categories for cyberbullying, distinguishing between insults and threats and identified user roles in the exchanges. Finally, Hardaker (2010) argues that trolling cannot be studied using established politeness research categories.

3. Trolling Categorization

In this section, we describe our proposal of a comprehensive trolling categorization. While there have been attempts in the realm of psychology to provide a working definition of trolling (e.g., Hardaker (2010), Bishop (2014)), their focus is mostly on modeling the troll's behavior. For instance, Bishop (2014) constructed a "trolling magnitude" scale focused on the severity of abuse and misuse of internet mediated communications. Bishop (2013) also categorized trolls based on psychological characteristics focused on pathologies and possible criminal behaviors. In contrast, our trolling categorization seeks to model not only the troll's behavior but also the impact on the recipients, as described below.

Since one of our goals is to *identify* trolling events, our datasets will be composed of *suspected* trolling attempts (i.e., comments that are suspected to be trolling attempts). In other words, some of these suspected trolling attempts will be real trolling attempts, and some of them won't. So, if a suspected trolling attempt is in fact not a trolling attempt, then its author will *not* be a troll.

To cover both the troll and the recipients, we define a (suspected trolling attempt, responses) pair as the basic unit that we consider for the study of trolling, where "responses" are all the direct responses to the suspected trolling attempt. We characterize a (suspected trolling attempt, responses) pair using four aspects. Two aspects describe the trolling attempt: (1) **Intention `(I)`** (what is its author's purpose?), and (2) **Intention Disclosure `(D)`** (is its author trying to deceive its readers by hiding his real (i.e., malicious) intentions?). The remaining two aspects are defined on each of the (direct) responses to the trolling attempt: (1) **Intention Interpretation `(R)`** (what is the responder's perception of the troll's intention?), and (2) the **Response strategy `(B)`** (what is the responder's reaction?). Two points deserve mention. First, R can be different from I due to misunderstanding and the fact that the troll may be trying to hide his intention. Second, B is influenced by R, and the responder's comment can itself be a trolling attempt. We believe that these four aspects constitute interesting, under-studied tasks. The possible values of each aspect are described in Table 1.

For a given (suspected trolling attempt, responses) pair, not all of the 189 ($= 3 \times 3 \times 3 \times 7$) combinations of values of the four aspects are possible. There are logical constraints that limit plausible combinations: a) *Trolling* or *Playing Intentions* `(I)` must have *Hidden* or *Exposed* Intention Disclosure `(D)`, b) *Normal* intentions `(I)` can only have *None* Intention disclosure `(D)` and c) *Trolling* or Playing interpretation `(R)` cannot have *Normal* response strategy `(B)`.

3.1. Conversation Excerpts

To enable the reader to better understand this categorization, we present an example excerpt taken from the original (Reddit) conversations. The first comment on the excerpt, generated by author `C0`, is given as a minimal piece of context. The second comment, written by the author `C1` in italics, is the suspected trolling attempt. The rest of the comments comprise all direct responses to the suspected trolling comment.

`C0:` Please post a video of your dog doing this. The way I'm imagining this is adorable.

`C1:` I hope the dog gets run over by a truck on the way out of the childrens playground.

`C2:` If you're going to troll, can you at least try to be a bit more convincing?

`C0:` Haha I hope the cancer kills you.

In this example, we observe that `C0`'s first comment is making a polite request. In return, `C1` made a mean spirited comment whose *intention* is to disrupt and possibly hurt

Class	Description	Size %
Intention (I)		
Trolling	The comment is malicious in nature, aims to disrupt, annoy, offend, harm or spread purposely false information	53.6% (537)
Mock Trolling or Playing	The comment is playful, joking, teasing or mocking others without the malicious intentions as in the Trolling class	8.9% (89)
No Trolling	A simple comment without malicious or playful intentions.	37.7% (375)
Intention Disclosure (D)		
Exposed	A troll, clearly exposing its malicious or playful intentions	34.7% (347)
Hidden	A troll hiding its real malicious or playful intentions	11.5% (115)
None	The comment's author is not a troll, therefore there are no hidden nor exposed malicious or playful intentions	53.8% (539)
Intentions Interpretation (R)		
Trolling	The responder believes that the suspected troll is being malicious, annoying, offensive, harmful or attempts to spread false information	59.7% (785)
Mock Trolling or Playing	The responder believes that the suspected troll is being playful, joking, teasing or mocking without the malicious intentions	5.3% (70)
No Trolling	The responder believes that the suspected comment has no malicious intentions nor is playful, it is a simple comment.	35.0% (461)
Response Strategy (B)		
Engage	Fall in the perceived provocation, showing an emotional response, upset or annoyed	26.9% (354)
Praise	Acknowledge the perceived malicious or playful intentions and positively recognize the troll's ingenuity or ability	3.0% (39)
Troll	Acknowledge the perceived malicious and counter attack with a trolling attempt	24.0% (316)
Follow	Acknowledge the perceived malicious or playful intention and play along with the troll, further trolling	3.0% (39)
Frustrate	Acknowledges the perceived malicious or playful intentions and attempt to criticize or minimize them	13.0% (171)
Neutralize	Acknowledges the perceived malicious or playful intentions and give no importance to them	9.5% (125)
Normal	There is no perception or interpretation of a trolling attempt and the response is a standard comment	20.7% (272)

Table 1: Classes for trolling aspects: Intention, Intention Disclosure, Intention Interpretation and Response Strategy. *Size* refers to the percentage per class, in parenthesis is the total number of instances in the dataset.

C0. Also, C1's comment is not subtle at all, so his intention is clearly *disclosed*. As for C2, she is clearly acknowledging C1's trolling intention and her *response strategy* is *frustrate*. Now, in C0's second comment, we observe that his *interpretation* is clear: he believes that C1 is *trolling* and the negative effect is so tangible that his response strategy is to *troll* back or *counter-troll* by replying with a comparable mean comment.

4. Corpus and Annotation

Reddit[1] is popular website that allows registered users (without identity verification) to participate in fora grouped by topic or interest. Participation consists of posting stories that can be seen by other users, voting stories and comments, and comments in the story's comment section, in the form of a forum. The forums are arranged in the form of a tree, allowing nested conversations, where the replies to a comment are its direct responses. We collected all comments in the stories' conversation in *Reddit* that were posted in August 2015. Since it is infeasible to manually annotate all of the comments, we process this dataset with the goal of extracting threads that involve suspected trolling attempts and the direct responses to them. To do so, we used Lucene[2] to create an inverted index from the comments. Given that trolling comments have malicious intentions, we queried the inverted index for comments containing one or more of the following six categories of words: (1) the word "**troll**" as well as words having an edit distance of 1 from it; (2) the list of highly offensive words identified by Nitta et al. (2013); (3) the list of impoliteness cues identified by Danescu-Niculescu-Mizil et al. (2013); (4) the list of words having a negative prior polarity according to Wilson et al.'s (2005) prior polarity lexicon; (5) 1061 swear words and short phrases collected from the internet, blogs, and forums; and (6) words that appear in the same WordNet synset as "anger". While it is certainly not the case that all trolling comments contain words belonging to one of these categories, we believe such comments would be reasonable candidates of real trolling attempts as our wordlists cover a broad range of strong and weak indicators of malice. This search produced a dataset in which 44.3% of the comments are real trolling attempts. Moreover, it is possible for commenters to believe that they are witnessing a trolling attempt and respond accordingly even where there is none due to misunderstanding. Therefore, the inclusion of comments that do not involve trolling would allow us to learn what triggers a user's interpretation of trolling when it is not present and what kind of response strategies are used.

For each retrieved comment, we reconstructed the original conversation tree it appears in, from the original post (i.e., the root) to the leaves, so that its parent and children can

[1] https://www.reddit.com/

[2] https://lucene.apache.org/

be recovered[3]. We consider a comment in our dataset a suspected trolling attempt if at least one of its immediate children contains the word *troll*. For annotation purposes, we created snippets of conversations exactly like the ones shown in our example, which consists of the parent of the suspected trolling attempt, the suspected trolling attempt, and all of the direct responses to the suspected trolling attempt.

We had two human annotators who were trained on snippets (i.e., (suspected trolling attempt, responses) pairs) taken from 200 conversations and were allowed to discuss their findings. After this training stage, we asked them to independently label the four aspects for each snippet. We recognize that this limited amount of information is not always sufficient to recover the four aspects, so we give the annotators the option to discard instances for which they couldn't determine the labels confidently. The final annotated dataset consists of 1000 conversations composed of 6833 sentences and 88047 tokens. The distribution over the classes per trolling aspect is shown in Table 2 in the column "Size".

On the 100 doubly-annotated snippets, we obtained substantial inter-annotator agreement according to Cohen's kappa statistic (Cohen, 1968) for each of the four aspects: Intention: 0.788, Intention Disclosure: 0.780, Interpretation: 0.797 and Response 0.776. In the end, the annotators discussed their discrepancies and managed to resolve all of them.

5. Trolling Attempt Prediction

In this section, we present preliminary results on predicting the four aspects of our task.

5.1. Feature Sets

In our preliminary experiments, we experiment with two feature sets. We plan to conduct experiments with additional features in the final version of the paper.

N-gram features. We encode each lemmatized and unlemmatized unigram and bigram collected from the training comments as a binary feature. In a similar manner, we include the unigram and bigram along with their POS tag as in Xu et al. (2012a). To extract these features we used Stanford CoreNLP (Manning et al., 2014).

GloVe Embeddings (glv). Word embeddings were created to overcome certain problems with the bag of words (BOW) representation, like sparsity, and weight in correlations of semantically similar words. For this reason, and following Nobota et al. (Nobata et al., 2016), we create a distributed representation of the comments by averaging the word vector of each lowercase token in the comment found in the Twitter corpus pre-trained GloVe vectors (Pennington et al., 2014). The resulting comment vector representation is a 200 dimensional array that is concatenated with the existing BOW.

[3]We removed the comments whose text had been deleted

5.2. Preliminary Results

Using the features described in the previous subsection, we train four independent classifiers using logistic regression[4], one per each of the four prediction tasks. All the results are obtained using 5-fold cross-validation experiments. In each fold experiment, we use three folds for training, one fold for development, and one fold for testing. All learning parameters are set to their default values except for the regularization parameter, which we tuned on the development set.

Results of the four tasks are shown in Table 2. In each task, we show the F-score on each of its classes in each row as well as the accuracy (the percentage of instances correctly predicted). The last row of the table shows the total accuracy obtained by averaging the accuracies over the four tasks.

The column mfc shows the results of the frequent class baseline, where the classifier always predicts an instance as belonging to the most frequent class. The percentage of instances belonging to each class can be seen in the last column (Size). As we can see, the mfc baseline achieves an overall accuracy of 52.5.

The next two columns show the results of classifiers trained on exactly one of the two feature groups described in the previous subsection. To get an idea of how strong the mfc baseline is, we can compare it with the classifier trained using only n-gram features (ngr). As we can see, the majority baseline is as strong as ngr w.r.t. overall accuracy (52.5). Nevertheless, ngr is a lot more interesting: it makes predictions on a variety of classes. glv outperforms ngr statistically significantly w.r.t. accuracy on all but Interpretation (paired t-tests, $p < 0.05$). On all but the Response Strategy tasks, glv performs better on the most frequent class than ngr but worse on the second most frequent class.

6. Error Analysis

In order to provide directions for future work, we plan to analyze the errors made by the classifiers on the four prediction tasks in the final version of the paper.

7. Conclusion

We presented a new view on the computational modeling of trolling in Internet fora where we proposed a comprehensive categorization of trolling attempts that for the first time considers trolling from not only the troll's perspective but also the responders' perspectives. Perhaps most importantly, we create an annotated dataset that we believe is the first of its sort. We intend to make publicly available with the hope of stimulating research on trolling.

8. Bibliographical References

Bishop, J. (2013). The effect of de-individuation of the internet troller on criminal procedure implementation: An interview with a hater. *International Journal of Cyber Criminology*, 7(1):28.

[4]We use the *scikit-learn* (Pedregosa et al., 2011) implementation.

Aspect/Class	mfc F	ngr F	glv F	Size %
I: Intention				
No trolling	69.7	64.2	71.9	53
Trolling	0.0	41.2	27.4	38
Playing	0.0	0.0	0.0	9
Accuracy	53.5	52.8	57.9	-
D: hidden				
Hidden	0.0	0.0	0.0	12
Exposed	0.0	51.2	31.8	35
None	69.9	66.5	72.0	53
Accuracy	53.9	56.8	58.9	-
R: Interpretation				
No trolling	0.0	50.8	38.7	34
Playing	0.0	0.0	0.0	6
Trolling	74.2	74.3	76.3	60
Accuracy	59.2	64.4	64.9	-
B: Response Strategy				
Frustrate	0.0	30.0	19.4	12
Troll	0.0	33.6	27.3	24
Follow	0.0	18.8	6.1	3
Praise	0.0	5.8	0.0	3
Neutralize	0.0	37.0	26.8	9
Normal	0.0	41.2	35.4	20
Engage	43.8	40.5	49.9	29
Accuracy	36.0	36.0	38.0	-
All Tasks Combined				
Total Accuracy	52.5	52.5	54.9	-

Table 2: Results on predicting Intention, Intentions Disclosure, Interpretation, and Response strategy.

Bishop, J. (2014). Representations of 'trolls' in mass media communication: a review of media-texts and moral panics relating to 'internet trolling'. *International Journal of Web Based Communities*, 10(1):7–24.

Cambria, E., Chandra, P., Sharma, A., and Hussain, A. (2010). Do not feel the trolls. *ISWC, Shanghai*.

Chen, Y., Zhou, Y., Zhu, S., and Xu, H. (2012). Detecting offensive language in social media to protect adolescent online safety. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 71–80. IEEE.

Cohen, J. (1968). Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

Dadvar, M., Trieschnigg, D., and de Jong, F. (2014). Experts and machines against bullies: a hybrid approach to detect cyberbullies. In *Advances in Artificial Intelligence*, pages 275–281. Springer.

Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J., and Potts, C. (2013). A computational approach to politeness with application to social factors. *arXiv preprint arXiv:1306.6078*.

Dinakar, K., Jones, B., Havasi, C., Lieberman, H., and Picard, R. (2012). Common sense reasoning for detection, prevention, and mitigation of cyberbullying. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(3):18.

Guha, R., Kumar, R., Raghavan, P., and Tomkins, A. (2004). Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM.

Hardaker, C. (2010). Trolling in asynchronous computer-mediated communication: From user discussions to academic definitions. *Journal of Politeness Research*, 6(2):215–242.

Kumar, S., Spezzano, F., and Subrahmanian, V. (2014). Accurately detecting trolls in slashdot zoo via decluttering. In *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*, pages 188–195. IEEE.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mihaylov, T. and Nakov, P. (2016). Hunting for troll comments in news community forums. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 399–405, Berlin, Germany, August. Association for Computational Linguistics.

Mihaylov, T., Georgiev, G. D., Ontotext, A., and Nakov, P. (2015a). Finding opinion manipulation trolls in news community forums. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning, CoNLL*, volume 15, pages 310–314.

Mihaylov, T., Koychev, I., Georgiev, G., and Nakov, P. (2015b). Exposing paid opinion manipulation trolls. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 443–450, Hissar, Bulgaria, September. INCOMA Ltd. Shoumen,

BULGARIA.

Nitta, T., Masui, F., Ptaszynski, M., Kimura, Y., Rzepka, R., and Araki, K. (2013). Detecting cyberbullying entries on informal school websites based on category relevance maximization. In *IJCNLP*, pages 579–586.

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., and Chang, Y. (2016). Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, pages 145–153. International World Wide Web Conferences Steering Committee.

O'sullivan, P. B. and Flanagin, A. J. (2003). Reconceptualizing 'flaming' and other problematic messages. *New Media & Society*, 5(1):69–94.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Van Hee, C., Lefever, E., Verhoeven, B., Mennes, J., Desmet, B., De Pauw, G., Daelemans, W., and Hoste, V. (2015). Detection and fine-grained classification of cyberbullying events. In *International Conference Recent Advances in Natural Language Processing (RANLP)*, pages 672–680.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354, October.

Xiang, G., Fan, B., Wang, L., Hong, J., and Rose, C. (2012). Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1980–1984. ACM.

Xu, J.-M., Jun, K.-S., Zhu, X., and Bellmore, A. (2012a). Learning from bullying traces in social media. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 656–666. Association for Computational Linguistics.

Xu, J.-M., Zhu, X., and Bellmore, A. (2012b). Fast learning for sentiment analysis on bullying. In *Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining*, page 10. ACM.

Xu, J.-M., Burchfiel, B., Zhu, X., and Bellmore, A. (2013). An examination of regret in bullying tweets. In *HLT-NAACL*, pages 697–702.

Automatic Annotation of Semantic Term Types in the Complete ACL Anthology Reference Corpus

Anne-Kathrin Schumann, Héctor Martínez Alonso
ProTechnology GmbH, Dresden, Germany, Thomson Reuters Labs, Toronto, Canada
annek_schumann@gmx.de, hector.martinezalonso@thomsonreuters.com

Abstract

In the present paper, we present an automated tagging approach aimed at enhancing a well-known resource, the ACL Anthology Reference Corpus, with semantic class labels for more than 20,000 technical terms that are relevant to the domain of computational linguistics. We use state-of-the-art classification techniques to assign semantic class labels to technical terms extracted from several reference term lists. We also sketch a set of research questions and approaches directed towards the integrated analysis of scientific corpora. To this end, we query the data set resulting from our annotation effort on both the term and the semantic class level level.

Keywords: semantic labeling, terminology, history of science

1. Introduction

Science changes continually: While certain research topics may be in a state of stagnation or decline, other research fronts move forward rapidly. However, even "dormant" (Menard, 1971) science can regain importance if new data is produced or methods are developed to tackle unresolved research problems. Scientific thought exhibits intricate evolutionary patterns (Fleck, 1980) and paradigm change (Kuhn, 1962) can affect the structure and outline of a whole discipline.

The availability of large collections of digitized scientific text enables systematic studies of the processes that drive scientific development. Recent years have seen a notable increase in quantitative studies of scientific text collections, e. g. Hall et al. (2008), Gupta and Manning (2011), Michaelis et al. (2013), Mariani et al. (2014), Babko-Malaya et al. (2015), Schumann and QasemiZadeh (2015b), Asooja et al. (2016), Francopoulo et al. (2016), Schumann (2016), Heyer et al. (2016).

The present study is a contribution to this research strand. Our work centers on the use of semantic labeling techniques for the automatic enhancement of a small corpus of manual term and semantic class annotations. The ultimate goal of our work, however, is to use this information as one feature in the profiling of scientific papers, communities, and disciplines. In using semantic class labels as one source of information, we take a macro- rather than a micro-perspective: While individual words and terms are certainly indicators of scientific trends and developments, it is necessary to relate them to more coarse-grained categories for an overall view of a scientific discipline. Semantic class annotations allow us to answer detailed questions about the evolution of computational linguistics over time. The data set described in this paper is made available to the research community[1].

2. Motivation and Related Work

The present investigation is conceptually related to earlier studies dedicated to the lexical analysis of diachronic corpora. Since the well-known work by Hall et al. (2008), topic modeling has been widely employed in the analysis of diachronic data. Topic modeling, however, has the disadvantage that it is ignorant to the concept of domain relevance. Topics, therefore, have to be painstakingly inferred post-hoc from word sets, and it is not straightforward which conclusions can be made on the basis of a topic model. Later work has shown that interesting insights can be obtained even with relatively simple methods of analysis, if the domain terminology is used as a clean, high-quality lexical representation of the data (Schumann and Qasemi-Zadeh, 2015b; Schumann, 2016; Heyer et al., 2016). The present study continues this line of research by relating individual terminological units to coarse-grained semantic classes. In particular, by adding semantic class information to existing knowledge about terminological units, we enable multi-dimensional queries of the data. On the basis of terminological and semantic class information, we can, for example, ask not only which terms have been trending in computational linguistics at a given time, but we can study the evolution of various sub-fields of computational linguistics and check which associations individual terms form within these sub-fields. This does, however, not prevent us from "drilling down" to the level of individual terms, but, if necessary, we can also take a more coarse-grained perspective by "zooming out" from there, as in traditional OLAP-style[2] analyses. In sum, we believe that our approach provides two types of added value if compared to earlier research:

- The lexical basis of our investigation has a sound terminological foundation. The lexical items analyzed are not random words, but they have been identified as relevant by subject-matter experts.
- We add a second level of analysis and thus provide the means for more expressive queries of the data.

Technically, our study is related to well-established lines of work in taxonomy enrichment and domain knowledge-base population (Montoyo et al., 2001; Bergamaschi et al., 2007;

[1] https://github.com/anetschka/complingterm.

[2] Online analytical processing (Codd et al., 1993).

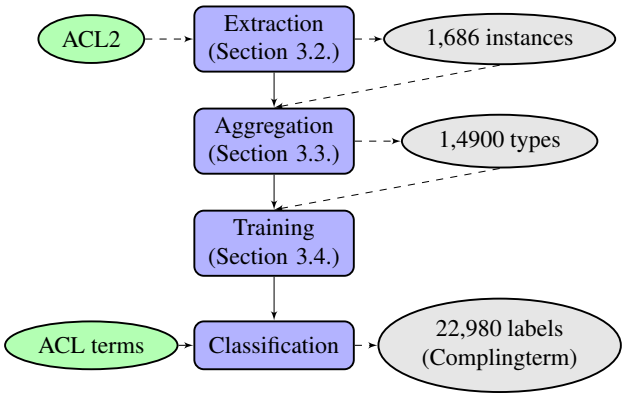


Figure 1: Flowchart representation of the semantic annotation process.

Pekar and Staab, 2003; Ruiz-Casado et al., 2007; Popescu et al., 2008; Ji and Grishman, 2011) in that we aim at the automatic type-based estimation of the semantic class of words or word sequences. Inspired by this family of approaches, we label each term, which is a specialized nominal expression of length one or more, with a semantic class.

3. Data Preparation and Semantic Labeling

We work on the ACL Anthology Reference Corpus (ACL ARC) in its first version (Bird et al., 2008). This corpus contains more than 10,000 scholarly articles from the computational linguistics domain that were published between 1965 and 2006.

We also use two additional data sets that have been created on the basis of the ACL ARC. In particular, we use a list of technical terms (ACL RD-TEC 1.0, termed *ACL1*) that was created by means of automatic term extraction (Q. Zadeh and Handschuh, 2014). More specifically, the term list was created with the help of several term extractors, and each term candidate was then manually validated by the main curator of the resource. This process resulted in more than 20,000 specialized terms that were deemed valid.

In our experiments, the ACL1 term list is used to identify all known terms. Moreover, we use a set of in-line, double-blind term and semantic class annotations (ACL RD-TEC 2.0, termed *ACL2*) provided on a subset of abstracts from the ACL ARC (QasemiZadeh and Schumann, 2016). These annotations were created by two human annotators in a multi-step process that resulted in both term span and semantic class annotations, following annotations guidelines that differentiate between seven semantic classes, as shown in Table 1. We use these high-quality annotations to train our classification models. Figure 1 provides a graphical representation of our work-flow for data preparation and annotation. Data flows from green input data ellipses to gray output data ellipses are represented with dashed lines. Blue boxes represent major work steps in the process and are explained in the sections to follow.

3.1. Data Preparation Work-flow

Annotation instances from ACL2 can be divided into 3 categories:

- Perfect matches: identical term spans marked by both annotators
- Partial matches: overlapping, but not identical term spans
- Annotation conflicts: term spans marked by only one of the two annotators

Type	Example
Technologies	parsing
Tools	parser
Language resources	corpus
Lang. resource products	Brown corpus
Models	language model
Measures	Bleu score
Other	*residual class*

Table 1: Semantic classes in ACL2.

Each instance has at least one semantic class assigned to it, but in all categories multiple (conflicting) class assignments can occur. Table 1 shows which semantic classes have been annotated in ACL2 (see Schumann and Qasemi-Zadeh (2015a) for details). We have prepared our training and test data as follows:

1. We extracted reliable *annotation instances* from ACL2. This is explained in more detail in Section 3.2.
2. We created consistent *annotation types* by aggregating annotation instances by their term lemmas. We relabeled a part of the annotations and merged the original semantic classes into larger containers. This procedure is explained in Section 3.3.
3. We merged both term lists (ACL1+2) to create a maximal term list and identified term occurrences in the whole ACL ARC.
4. The classifier is described in Section 3.4.

3.2. Extraction of Reliable Annotation Instances

From ACL2, we extracted all consensual annotations, that is, term occurrences that were annotated by both annotators with the same span and semantic class: Among the 4,849 manual annotation instances, 2,583 share exactly the same span, being complete matches of each other. Among those, 1,686 also share the same semantic class. Appendix A shows how these terms are distributed over the different publication years in the data set and how many abstracts were annotated for each year.

As can be seen from the table, the distribution is highly skewed in favor of an, overall, too large “other” class. Why is this data so unbalanced? In many cases, the “other” class contains linguistic units that are neither language resources nor language resource products. Examples of such “other” instances are terms like “verbal interaction” or ”Japanese kanji-kana characters”. However, linguistic left-overs do not make up the complete “other” class: Specialized language is embedded in discourse and, therefore, terminological units in real-world abstracts are sometimes not those that one might find in a specialized taxonomy or ontology. For example, they might be ambiguous if taken out of context, or they might be discoursive variants of known terms that still bear terminological weight. Examples of such items are terms like “syntactic descriptions” (which might be anything between a strongly formalized and a free-form

description) or "error characters". Such units are, indeed, characteristic of academic language and should not be ignored. However, with the heavy skew observed, the ACL2 data set seems hardly usable for automatic prediction.

3.3. Conversion of Annotation Instances to Annotation Types

Since our work was aimed at creating a reliable set of semantic term type annotations, *instance* annotations from ACL2 had to be converted to annotation *types*. We did this by grouping the 1,686 perfect matches from ACL2 by their term lemmas, arriving at 1,490 annotation types. All "other" attributions were then reconsidered and, if necessary, relabeled manually. The main goal of this step was to arrive at a more even class distribution in the training data and, most notably, a smaller residual class. Since in the ACL2 data identical lemmas can have diverging semantic labels in different annotation instances, re-annotation with the aim of producing consistency with respect to other annotation *instances* seemed justified. All relabellings were discussed by both authors of this paper. If necessary, ACL papers from the corpus were analyzed in detail. Then, to deal with very tiny classes, we merged the 7 semantic classes originally annotated in ACL2 into 4 larger classes, namely:

- **Mathematics**: This class contains the Models and Measures classes from ACL2.
- **Technologies**: This is the superclass of the ACL2 classes Technologies and Tools.
- **Linguistics**: This class contains terms with a linguistic background, that is the Language resources and Language resource products classes from ACL2 along with a relevant share of relabeled "other" instances.
- **Interdisciplinary**: After the re-annotation, what is left of the "other" class now contains general, higher-level interdisciplinary terms that nevertheless bear terminological weight.

The 4 classes were formed by merging conceptually similar tiny classes to form larger and slightly more general classes. Figure 2 provides a graphical overview of the semantic classes before and after merging. Green boxes represent the 7 semantic classes that were manually annotated in ACL2. These were merged into 4 coarse-grained classes for more reliable automatic prediction (blue boxes). The tree in Figure 2 is also labeled with example terms from ACL2[3]. As a result of our restructuring of the data, the bulk of the purely linguistic terms has been moved to the Linguistics class. What remains in Interdisciplinary can now be related to technicalities of the scientific process and to scientific discourse. Examples of the Interdisciplinary class are terms such as "speech-act indirectness", "np-hard problem", or "telephone communication". Table 2 gives an overview of the training data resulting from our preparatory work. The table shows that our efforts have produced a more even class distribution. Classes with very few instances have been merged with larger classes.

[3]Note that due to our work-flow, we have access to both coarse-grained and fine-grained (ACL2) semantic information for the 1,490 training terms. For all remaining terms we have coarse-grained semantic labels.

Type	Number
Mathematics	226
Technologies	677
Linguistics	283
Interdisciplinary	304
Overall	**1,490**

Table 2: Term type distribution in training data

3.4. Automatic prediction

In order to assign a semantic class to the unlabeled terms extracted from ACL1, we have implemented a logistic-regression classifier trained on the annotated data (ACL2). We have performed feature selection using ten-fold cross-validation. The resulting classifier uses the following features:

1. **BoW**: Identity of the term headword. If the term is longer than one word, we treat all words from the second as a bag of words (BoW). In this way, we give the headword of the term a special treatment, which makes it easier to identify as a trigger term for a certain class.
2. **Length**: The length of the term in number of words, and the proportion of capitalized characters. These features help identify multi-word expressions and determine whether they are terminological units, or whether they are acronyms.
3. **Brown**: The Brown clusters (Brown et al., 1992) from the full ACL corpus for the words in the term. Brown clusters group words in a corpus according to their immediate surrounding bi-grams and provide good features to estimate semantic classes.
4. **Embeddings**: The average word embedding for all words in the term. We use embeddings from the ACL corpus with 100 dimensions and a word window of 5. Using embeddings allows us to incorporate distributional information of words involved in a term that is larger in scope than the information captured by Brown clusters.
5. **WordNet**: The number of senses in WordNet (Miller, 1995) for the term headword, as well as the list of semantic types (e.g. *noun.cognition*) for these senses. The intuition behind these features is that more polysemous words are more likely to be the head of terms, and by including their possible coarse-grained senses it becomes easier to characterize their semantic class.

We have used ten-fold cross-validation during development to determine which classifier setup was more robust. However, the classifier we used to actually tag the corpus is trained on the full dataset. In this way, the scores we provide are a conservative estimate of the actual performance of the classifier. Table 3 shows the performance of the classifier in terms of F1 score for the chosen classifier, which uses all the features listed, and a comparison baseline that only uses **BoW**.

As can be seen from the table, our classifier performs reasonably well, reaching an average F1 score of 0.73. Classification performance is obviously influenced by the number of training examples with the largest class achieving the

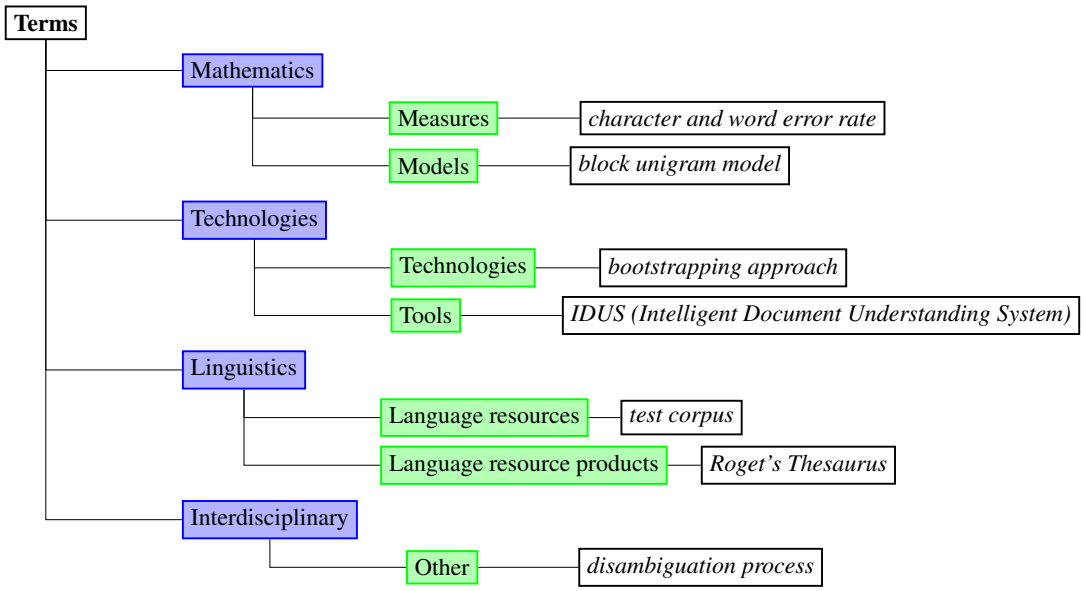


Figure 2: Taxonomic representation of the semantic categories used for training data preparation. Leaves in the tree are example terms extracted from ACL2 data.

	Baseline	**Full**
Technologies	0.75	0.83
Linguistics	0.61	0.70
Math. concepts	0.59	0.64
Interdisciplinary	0.53	0.60
Micro-Average	**0.65**	**0.73**

Table 3: F1 classification scores.

best individual score. The rather fuzzy in terms of distribution and lexically very varied Interdisciplinary class exhibits the lowest score. Conversely, the Technologies class is easier to identify because it is often made up of longer expressions and tends to show frequent, informative words like *system*.

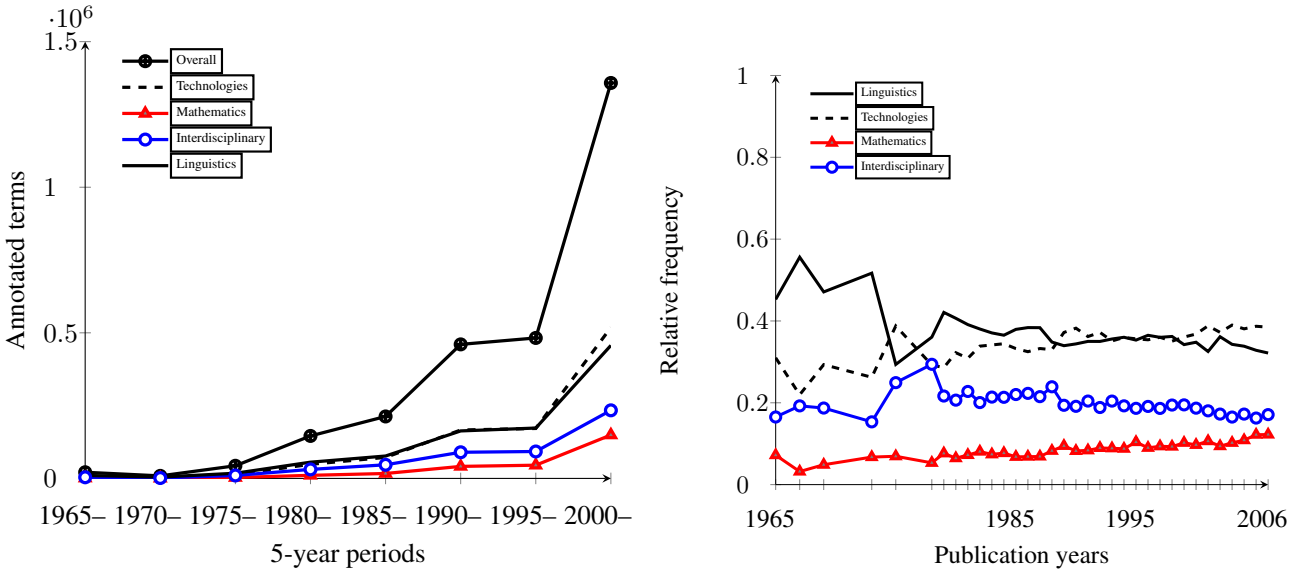
It is not straightforward to compare our classification result to the outcome of other annotation efforts, not only because there are only few such efforts, but also because annotation evaluation scenarios vary considerably. However, QasemiZadeh and Schumann (2016) provide a detailed analysis of their manual annotation of the same data set. They report class-wise IAA scores that range between 0.44 (for the Model class, compare Figure 2) and 0.83 (for the Tool class, which is easily identifiable, but too small for classification, compare Appendix A), respectively. This shows that in manual annotation, too, class parameters influence annotation quality. The overall quality of our automatic classification, therefore, seems reasonably close to that of the current manual annotation benchmark. It must be noted, however, that the score reported by QasemiZadeh and Schumann (2016) is a combined score for both term identification and semantic class assignment.

We also performed a manual analysis of a small subset of the predictions, namely a 30-term subset for each of the four classes. Out of thirty examples of the Technologies category, we only would relabel one ("bottom-up approach") as part of Interdisciplinary. In the predictions for Linguistics, we find the false positive "electronic mail", which should be a Technology. This mis-prediction is a result of the WordNet feature providing the super-sense *noun.communication* to the word "mail". The Mathematics category ends up being the label of choice for all expressions with the word "model" as a headword ("IBM Models 1-2"). "Model" is a fairly polysemous word that could yield terms of any category, but the frequency and the annotation preference for labeling them as terms in Mathematics enforces this bias. The Interdisciplinary class does indeed contain more noise, and has many terms that should belong to the Linguistics category ("simultaneous speech" or "subject-object relation"). We attribute the permeation between these two classes to some of the features that give account for polysemy, as both Linguistics and Interdisciplinary contain more words with numerous possible senses, in addition to being the classes with highest word variety in terms of type–token ratio. Our manual analysis corroborates that Technologies is by far the most reliable label of our predictions. Appendix B shows some of the examples that underwent the manual analysis just described. For each term, the table shows the classification probabilities given by our classifier for the four possible classes. The value for the resulting class is highlighted in bold.

4. Analysis of Resulting Data Set

4.1. Structure

The data set resulting from our annotation is essentially an annotated term list. This list contains 22,980 computational linguistics terms. Each term is annotated with a probability vector with four components, where each component represents the probability that a given term is an instance of the respective semantic class, as illustrated in Appendix B. Finally, a class label indicates the semantic class with the highest probability.



(a) Number of term occurrences with semantic label over time.

(b) Relative importance of semantic classes over time.

Figure 3: Classification results: annotated terms and semantic classes over time.

4.2. Querying the Data

As pointed out earlier, we maintain that semantic class annotations facilitate the analysis of scientific text corpora. In the last section of this paper, we, therefore, sketch some of the research questions that can be tackled through multi-dimensional analyses involving both the term and the semantic class level.

One of the most prominent properties of the ACL data as a whole is its exponential growth over time. This finding is supported by Figure 3a which plots the number of known and semantically labeled terms over time periods of 5 years. The tendency of scientific disciplines to grow exponentially has been described already by Menard (1971) and it is, indeed, a property that informs all studies on the ACL data. Figure 3b plots the classification results over time, when class-wise term counts are normalized by the number of all terms per year. Thus, the plot shows the relative importance of each semantic class over the course of the publication period. The probably most prominent trend in this plot is that Linguistics and Technologies, over the years, switch ranks, with Linguistics slightly losing importance and Technology mentions becoming more frequent. We also observe a slight, but steady increase in the relative frequency of terms referring to mathematical concepts.

To check whether our dataset allows us to ask more specific questions about the development of computational linguistics, we set up a simple database holding information about both terms, their semantic classes and term occurrences across all publication years. We have used this database to extract frequency data that was further analyzed with the following techniques:

- We used word rank comparisons to identify "trending" terms, that is, lexical units that gain importance in a given period, as proposed by Schumann and Qasemi-Zadeh (2015b). The method is based on the comparison of two ranked word lists for consecutive time periods and identifies lexical units that undergo a strong positive rank shift, that is, words that exhibit an "upwards" movement in the transition from one time period to the next.

- We used frequency and productivity scores to analyze individual terms' life cycles, as proposed by Schumann (2016). In this approach, productivity is formalized in terms of entropy, that is, a base term with many and frequently used related multi-word units is considered particularly productive. Unlike simple frequency analysis, this approach helps, for instance, to differentiate between short-term tendencies and longer-term trends.

1986–1989	**2000–2006**
tree	measure
formalism	language model
user model	f-measure
representations	n-gram
domain model	distribution
discourse model	f-score
measure	n-grams
perplexity	nist
language model	translation model
projection	parse tree

Table 4: Mathematics terms with high positive frequency rank shifts in two different time intervals. In both cases, frequency counts were compared to frequency counts from the preceding time interval.

In a first step, we used our SQL database to obtain an overview of the terms pertaining to the four semantic classes and their frequencies at various points in time. For

distribution		**measure**	
1986–1989	1990–1996	1986–1989	1990–1996
distribution	distribution	measure	measure
probability distribution	probability distribution	distance measure	f-measure
gaussian distribution	uniform distribution	similarity measure	similarity measure
binomial distribution	joint distribution	evaluation measure	evaluation measure
class distribution	normal distribution	association measure	distance measure
probability distribution matrix	frequency distribution	confidence measure	statistical measure
	binomial distribution	theoretic measure	f measure
	distributional similarity		cosine measure
	prior distribution		precision measure
	gaussian distribution		association measure
	...		...

Table 5: Top terms matching "distribution" and "measure" in two time periods. Terms are sorted by their frequencies.

instance, we counted the occurrences of terms pertaining to Mathematics in 4 time intervals: 1980-1985, 1986-1989, 1996-1999, and 2000-2006. Using the rank comparison technique, we could then contrast different methodologies and models that were popular in computational linguistics research at different times. Table 4 exemplifies this by giving an overview of the terms with the highest positive frequency rank shifts from Mathematics for two different time intervals.

On the basis of the comparison exemplified in Table 4, it seems relatively easy to contrast explicit, knowledge-based or abstract modeling in the 1980s with scoring and statistical analysis, approaches that were prominently used in the later time period. However, the table also seems to indicate that there is nothing disruptive about this change in the Mathematics class. Rather, concepts like "measure" or "language model" seem to have kept gaining importance, whereas other concepts seem to have stagnated. This claim can easily be substantiated by querying our database for frequencies and collocations containing terms such as "feature", "distribution", "measure", or "model". These terms exhibit strong productivity increases, that is, a growing number of related multi-word units, in the 1980s and a continuous increase in frequency over the complete period of time under study. This means that these terms, at the time of their rise, did not necessarily denote completely novel concepts, but they have found new fields of application or new ways were found to build and apply such models, features, etc. In this sense, terms like "distribution" or "measure" can be hypothesized to constitute motors of scientific innovation. Table 5 details the results of example queries for the terms "distribution" and "measure" for only two time periods, namely 1986–1989 and 1990–1995. The rise in productivity for the two terms over this short time period seems rather strong, however, as claimed before, the examples illustrate a scientific evolution rather than a disruptive change. It is an interesting task to identify the area of computational linguistics that has initiated this development, and with the rich data set presented here, it seems actually feasible to solve this task.

By repeating the analysis for the Technologies class we were able to highlight some interesting technological trends that have been prominent in computational linguistics over the last decades. For the 1990–1995 interval, for instance, the ranks shift analysis highlights the following trends: the use of WordNet as a lexical resource, the use of finite-state transducers for complex analysis tasks, work on word sense disambiguation, but also on part-of-speech tagging and other kinds of linguistic annotation, to mention only the most prominent items in our result list. Techniques used are not purely statistical, as highlighted by terms such as "lexical rule" or "heuristic".

For the 2000–2006 period, many of the lexical units highlighted by the rank shifts analysis in Technologies are related to recent work on the use of machine learning in computational linguistics, e. g. terms such as "classifier" or "feature". However, ontologies have also gained importance and the 2000–2006 slice of Technologies includes novel lexical units such as "ontology learning", "ontology acquisition", or "ontology induction". Annotation, both manual and automatic, remains an important topic and FrameNet is introduced as a new resource. The growing use of the term "NLP" and the rise of statistical machine translation seem to constitute other important tendencies.

5. Conclusion

In this paper, we have described our methodology for creating a large data set of semantically annotated terms. We have merged linguistic information from several existing data sets and performed manual re-annotation to arrive at a more even distribution of semantic class instances. We have used state-of-the-art classification techniques to provide semantic class labels for known term instances. Both manual and automatic evaluation of our classification results indicate reasonably good classification quality.

We have also carried out an initial analysis of the resulting data set to exemplify which kinds of questions can be answered using the rich annotations provided by our data set. Although we have by no means described a complete methodology for the analysis of scientific corpora, we believe that our analysis shows what is possible with the wealth of data available. State-of-the-art methods can be used not only to identify large-scale trends such as the growing importance of statistical and machine learning methods in computational linguistics. The semantic class labels resulting from our classification effort allow to per-

form this analysis on a more fine-grained level and search for interesting phenomena in actual subfields of computational linguistics. We have also shown that the analysis of mere frequency and productivity information allows us to identify some of the concepts that have in the past served as motors of innovation, such as the concepts "measure" and "distribution". In the future we hope to develop an integrated methodology that will be able to reliably relate terms and groups of terms to specific time periods and subfields and to trace scientific innovations back to the nuclei from which they originated.

6. Bibliographical References

Asooja, K., Bordea, G., Vulcu, G., and Buitelaar, P. (2016). Forecasting Emerging Trends from Scientific Literature. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association.

Babko-Malaya, O., Seidel, A., Hunter, D., HandUber, J. C., Torrelli, M., and Barlos, F. (2015). Forecasting Technology Emergence from Metadata and Language of Scientific Publications and Patents. In *Proceedings of the 15th International Conference on Scientometrics and Informetrics (ISSI 2015)*. Boğaziçi Universitesi.

Bergamaschi, S., Bouquet, P., Giazomuzzi, D., Guerra, F., Po, L., and Vincini, M. (2007). An Incremental Method for the Lexical Annotation of Domain Ontologies. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 3(3):57–80.

Bird, S., Dale, R., Dorr, B., Gibson, B., Joseph, M., Kan, M.-Y., Lee, D., Powley, B., Radev, D., and Tan, Y. F. (2008). The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*. European Language Resources Association.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-Based n-gram Models of Natural Language. *Computational Linguistics*, 18(4):467–479.

Codd, E., Codd, S. B., and Salley, C. T. (1993). Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandat. Technical report, E.F. Codd & Associates.

Fleck, L. (1980). *Entstehung und Entwicklung einer wissenschaftlichen Tatsache. Einführung in die Lehre vom Denkstil und Denkkollektiv*. Suhrkamp, Frankfurt am Main. First edition in 1935, Benno Schwabe & Co., Basel.

Francopoulo, G., Mariani, J., and Paroubek, P. (2016). Predictive Modeling: Guessing the NLP Terms of Tomorrow. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association.

Gupta, S. and Manning, C. D. (2011). Analyzing the Dynamics of Research by Extracting Key Aspects of Scientific Papers. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*. Asian Federation of Natural Language Processing.

Hall, D., Jurafsky, D., and Manning, C. D. (2008). Studying the History of Ideas Using Topic Models. In *Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Heyer, G., Kantner, C., Niekler, A., Overbeck, M., and Wiedemann, G. (2016). Modeling the dynamics of domain specific terminology in diachronic corpora. In *Proceedings of 12th International conference on Terminology and Knowledge Engineering (TKE 2016)*.

Ji, H. and Grishman, R. (2011). Knowledge Base Population: Successful Approaches and Challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*. Association for Computational Linguistics.

Kuhn, T. S. (1962). *The Structure of Scientific Revolutions*. University of Chicago Press.

Mariani, J., Paroubek, P., Francopoulo, G., and Hamon, O. (2014). Rediscovering 15 Years of Discoveries in Language Resources and Evaluation: The LREC Anthology Analysis. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*. European Language Resources Association.

Menard, H. W. (1971). *Science: Growth and Change*. Harvard University Press, Cambridge, Mass.

Michaelis, J. R., Mcguiness, D. L., Chang, C., and Babko-Malaya, O. (2013). Towards explanation of scientific and technological emergence. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2013*.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Montoyo, A., Palomar, M., and Rigau, G. (2001). WordNet Enrichment with Classification Systems. In *Proceedings of WordNet and Other Lexical Resources: Applications, Extensions and Customisations Workshop. The Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL 2001)*. Association for Computational Linguistics.

Pekar, V. and Staab, S. (2003). Word classification based on combined measures of distributional and semantic similarity. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003)*. Association for Computational Linguistics.

Popescu, A., Grefenstette, G., and Moëllic, P. A. (2008). Gazetiki: Automatic Creation of a Geographical Gazetteer. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital libraries*. ACM.

Q. Zadeh, B. and Handschuh, S. (2014). The ACL RD-TEC: A dataset for Benchmarking Terminology Extraction and Classification in Computational Linguistics. In *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*. Association for Computational Linguistics and Dublin City University.

QasemiZadeh, B. and Schumann, A.-K. (2016). The ACL RD-TEC 2.0: A Language Resource for Evaluating Term Extraction and Entity Recognition Methods. In

Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA).

Ruiz-Casado, M., Alfonseca, E., and Castells, P. (2007). Automatising the learning of lexical patterns: An application to the enrichment of WordNet by extracting semantic relationships from Wikipedia. *Data & Knowledge Engineering*, 61(3):484–499.

Schumann, A.-K. and QasemiZadeh, B. (2015a). The ACL RD-TEC Annotation Guideline: A Reference Dataset for the Evaluation of Automatic Term Recognition and Classification. Technical report.

Schumann, A.-K. and QasemiZadeh, B. (2015b). Tracing Research Paradigm Change Using Terminological Methods: A Pilot Study on "Machine Translation" in the ACL Anthology Reference Corpus. In *Proceedings of 11th International Conference on Terminology and Artificial Intelligence*. CEUR Workshop Proceedings.

Schumann, A.-K. (2016). Brave New World: Uncovering Topical Dynamics in the ACL Anthology Reference Corpus using Term Life Cycle Information. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*.

Year	Lang. resources	Lang. resource products	Measures	Models	Other	Technologies	Tools	Sum	Abstracts
1978	0	0	0	0	36	4	0	40	3
1980	0	0	0	1	12	9	2	24	5
1982	0	0	0	0	16	11	4	31	3
1984	0	0	0	2	37	17	1	57	5
1986	0	0	0	1	91	10	2	104	7
1988	6	0	0	4	40	23	3	76	12
1990	14	1	2	0	52	25	0	94	11
1992	8	9	4	2	75	60	6	164	19
1994	5	1	2	6	88	57	4	163	16
2001	4	0	18	14	89	88	4	217	18
2003	26	0	18	29	158	111	4	346	29
2005	17	0	20	12	89	64	2	204	23
2006	7	1	11	2	95	44	6	166	20
Overall	**87**	**12**	**75**	**73**	**878**	**523**	**38**	**1,686**	**171**

Appendix A: Time distribution of semantic classes in ACL2 (perfectly matching instances).

Term	Technologies	Linguistics	Interdisciplinary	Mathematics
general-to-specific learning	**0.97**	0.00	0.03	0.00
spoken document categorization	**0.74**	0.25	0.01	0.00
arabic stemmer	**0.89**	0.05	0.06	0.00
constraint propagation algorithm	**0.97**	0.00	0.00	0.03
token processing	**0.98**	0.00	0.01	0.01
standard grammar textbook	0.40	**0.54**	0.03	0.02
pre-discourse meaning	0.02	**0.88**	0.09	0.01
korean language	0.01	**0.96**	0.02	0.01
case particle	0.13	**0.78**	0.03	0.05
prosody	0.08	**0.52**	0.36	0.04
knowledge editing	0.30	0.00	**0.69**	0.00
multilingual information exchange	0.31	0.00	**0.68**	0.01
simultaneous speech	0.18	0.12	**0.69**	0.01
subject-object relation	0.01	0.28	**0.67**	0.05
categorisation research	0.46	0.03	**0.50**	0.00
homomorphism	0.08	0.17	0.31	**0.44**
IBM Models 1-2	0.09	0.00	0.02	**0.88**
time complexity	0.00	0.00	0.39	**0.61**
bigram distribution modelling	0.27	0.00	0.00	**0.73**
likelihood reestimation	0.43	0.00	0.00	**0.57**

Appendix B: Example output of the classifier used for manual analysis, resulting class is highlighted in bold.

Annotated Corpus of Scientific Conference's Homepages for Information Extraction

Piotr Andruszkiewicz, Rafał Hazan
Instititute of Computer Science, Warsaw University of Technology
Warsaw, Poland
P.Andruszkiewicz@ii.pw.edu.pl, R.Hazan@stud.elka.pw.edu.pl

Abstract

In this paper, we present a new corpus that contains 943 homepages of scientific conferences, 14794 including subpages, with annotations of interesting information: name of a conference, its abbreviation, place, and several important dates; that is, submission, notification, and camera ready dates. The topics of conferences included in the corpus are equally distributed over five areas: artificial intelligence, natural language processing, computer science, telecommunication, and image processing. The corpus is publicly available. Beside the characteristics of the corpus, we present the results of information extraction from the corpus using SVM and CRF models as we would like this corpus to be considered a reference data set for this type of task.

Keywords: annotated corpus, scientific conference's homepages, information extraction

1. Introduction

Up-to-date information about conferences is important for scientists who track conferences of their interest and check, e.g., dates of a conference, deadlines, that could change especially during submission period. Thus, a system that gathers such information could ease scientists' lives. The system should collect data about conferences and keep it up-to-date. Moreover, it should provide data in a structured way to facilitate searching conferences and obtaining information about any changes. The crucial part of this system are methods for collecting data about conferences automatically, e.g., homepages of a conference for the current and previous years, when and where a conference will be held, submission, notification, camera ready dates, etc.

In this paper, we present a new corpus that contains homepages of conferences with annotations of interesting information, e.g., name of a conference, its abbreviation, several important dates for the conference. The motivation behind this task was that according to our knowledge there is no any publicly available corpus in such a domain. The corpus can be used to train a tool for information extraction from unstructured sources containing data describing conferences. We chose conference home pages as a source as they contain up-to-date information. Structured services, such as WikiCFP, do not always update information (e.g., when deadline changes) and cannot be used in a real system for gathering up-to-date information about conferences.

Beside the characteristics of the corpus, we present results of information extraction as a baseline and a proof that this corpus can be used as a reference data set for information extraction from homepages of conferences. The corpus is publicly available and can be downloaded from the following website http://ii.pw.edu.pl/~pandrusz/data/conferences/.

The remainder of this paper is organised as follows: In Section 2. we describe the corpus we created. In Sections 3. the preprocessing and features are described. The experimental results are discussed in Section 4.. Section 5. presents related work. Finally, Section 6. summarizes the conclusions of the study and outlines avenues to explore in the future.

2. The Corpus

On the internet one can find corpora for information extraction, e.g., the corpus for information extraction from researcher's homepages (Tang et al., 2008), seminar announcements (Califf and Mooney, 1999; Freitag and McCallum, 1999). However, we could not find any publicly available corpus for scientific conferences. Therefore, we created an annotated corpus for the task of information extraction from homepages of scientific conferences. This corpus is publicly available and can be found on the website http://ii.pw.edu.pl/~pandrusz/data/conferences/.

Our decision to collect homepages of conferences, not Call For Papers (CFPs), is based on the following findings. We verified 100 passed conferences from our corpus for which we were able to find a running homepage and determine the important dates for a conference. Then we compared the data from a homepage and from WikiCFP service. It appeared that in WikiCFP about 70% of conferences have not up-to-date information about important dates, mostly submission date, as this date changes most often as the deadline approaches/passes. The dates are stable until the submission date comes, then dates are changed on a homepage, however, they are not updated in WikiCFP. Furthermore, data provided in CFPs is limited, e.g., it usually lacks information about sponsors. In (Xin et al., 2008) the authors stated that only less than 10% of CFPs analysed by them presented information about sponsors. Moreover, a service might not have information about conference we are looking for because it is field specific or covers only small part of all conferences in the field. According to authors of (Xin et al., 2008), it was possible to find only 40% of the textual CFPs of the top 293 computer science conferences listed at *Citeseer* (http://citeseer.ist.psu.edu/impact.html), while searching such conference services.

In our work CFPs proved useful in gathering not detailed information on conferences; namely, the list of conference webpage addresses. During the process of gathering

the corpus we wanted to make it as automatic as possible. To that end, as a first step we gathered a list of conferences from a conference hub. We chose the WikiCFP (http://wikicfp.com/cfp/) for that purpose, as it is a well known service and contains CFPs for areas we are interested in. Then we downloaded the homepage link and other data about each conference from WikiCFP. After that we downloaded the homepages and subpages (the depth level was restricted to one) within the same domain as the main page.

In the next step of corpus creation, for each conference (sub)page and each entity, e.g., submission date equal to 15 January 2015, we automatically found all instances of this entity and annotated them in the html source code of the web page.

The method of searching for an instance of the entity could not be a simple comparison of characters for several reasons. For instance, there are different ways of writing dates, the names of the conference provided in WikiCFP could differ slightly from the name on the page. Thus, we employed the following method for name of a conference comparison. We removed all conference name stop-words, e.g., word Workshop from Workshop on Collaborative Online Organizations. This led to Collaborative Online Organization that was searched. The conference name stop-words list has been manually created and consists of: The, International, Conference, Workshop. Moreover, we allow other single words to appear between words that were being searched. We applied case sensitive search. When conference name stop-words were neighbours of found instances, we added these conference name stop-words to annotation as their constitute a name. Though we do not annotate the year number and the consecutive number of the conference if they appear at the beginning of the name. To deal with different formats of date, we employed GATE tool and its default JAPE (Java Annotation Patterns Engine) rules (Kenter and Maynard, 2005). After the automatic process, the annotations were verified by three persons and manually corrected/added where necessary. This step is necessary as WikiCFP may not have up-to-date information as already explained. In case of disagreement majority voting was used.

The corpus we created contains 943 annotated homepages, 14794 pages including subpages, of scientific conferences. Hence, there are more than 15 pages per conference on average. The topics of conferences are equally distributed over five topics; namely, artificial intelligence, natural language processing, computer science, telecommunication, and image processing. The following entities were annotated: name and abbreviation of the conference, place, dates of the conference, submission, notification, final version due dates, the tags used in corpus are *cname*, *abbre*, *where*, *when*, *subm*, *notf*, *finv*, respectively. The annotated entity types are the most important considering a system that gathers information about conferences and is used by scientists to track conferences of their interest. However, we plan to annotate additional entity types, e.g., general and local chairs, invited speakers, sponsors.

The statistics of the corpus are presented in Table 1. Column *Avg. length* presents an average token length of an entity type. Thus, *abbreviation* contains one token. The name of a *place*, where a conference is held, sometimes consists of two tokens, which is consistent with names of cities or countries, e.g., New Zealand. The length of the name of a conference is almost 7. The lenght of each *important date* is about 3.5 and is consistent within the dates. The *date* of a conference is longer (about 4.8) because it contains a range of days. Two next columns *Inst.* and *Inst. per conference* present the number of instances of an entity type in the corpus and the average number per conference. *Important dates* are less frequent on homepages. The most frequent is *place* and surprisingly it is mentioned over 80 times per conference.

Entity type	Tag	Avg. length	Inst.	Inst. per conference
Name	cname	6.93	9954	10.6
Abbreviation	abbre	1.00	52222	55.4
Place	where	1.07	79091	83.9
Date	when	4.78	11261	11.9
Submission	subm	3.54	3196	3.4
Notification	notf	3.56	2081	2.2
Final ver. due	finv	3.54	3851	4.1

Table 1: The characteristics of the corpus.

3. Preprocessing and Features

To reduce the number of features, we use Snowball stemmer (Porter, 2001) in the preprocessing phase. We also remove words from a custom stoplist. Words that often occur in conference names, e.g., 'the', 'and', 'on', are not included in the stoplist.

We extract a main article or paragraphs from a web page using Boilerpipe (Kohlschütter et al., 2010) library. Text is not removed from the web page in order to avoid situation in which important elements are removed by mistake.

In our approach, we distinguish four group of features; namely, local, offset, layout, and dictionary features.

3.1. Local Features

Local features are created based on a current word that is being analysed. The commonly used feature is a *word*. This feature is not created for words from the stoplist and those tokens that contain nonalphabetic characters. The second feature contains part of speech (POS) tags for a current word calculated by *Penn Pos Tagger* from *factorie* package (McCallum et al., 2009). *Short word* feature is assigned a value *true* for words containing 2 to 5 characters. *Shape of a word* represents numbers with *1*, capital letters with *A* and small letters with *a*. If there are more than two the same characters in the value of this feature, the sequence is reduced to two characters. For *type of a word* feature we created eight types of words. *Short phrase* is set for words being a sequence of length of one or two words, for instance, named entities with two words, e.g., Carl Brunto. *Long phrase* indicates words of sequences with at least three words. We distinct between short and long phrases because conference names are usually long phrases

Table 2: The importance of features groups for entities (F1 measure, the best results marked in bold).

Features	Name	Abbrev.	Place	Date	Submission	Notification	Final ver. due
All	**0.36**	**0.76**	**0.67**	**0.80**	**0.60**	0.46	**0.65**
Without local	0.09	0.55	0.66	0.33	0.50	0.35	0.52
Without offset	0.33	0.68	0.62	0.67	0.00	0.00	0.00
Without layout	0.26	0.52	0.54	0.71	0.58	0.48	0.60
Without dict.	0.33	0.74	0.55	0.69	0.56	**0.49**	0.58

and locations of conferences are usually short. *Date* indicates dates that are present on a web page. Other types are: *Number* - assigned for numbers, e.g., 12, 1st; *acronym* indicates words of the following shapes: AAaa, AaAA, AAa, AA, AaaAaa AaaAA, AA1AA, AAaAA, *punctuation marks*, *special char* represents nonalphanumeric chars that are not punctuation marks. Other words are marked with *standard word* type and represent words that probably are not interesting in the case of information we want to extract.

3.2. Offset Features

Predecessor represents features calculated for the word that precedes a current word. We assume that we take into account only *type of a word* feature for one predecessor. *Successor* feature is constructed in the analogous way. Important dates of conferences represented as lists or tables are easy to understand for humans and hard to process by machine learning algorithms. We can find dates on the left, right, below or above a description of a date. We created date surrounding words feature to help machine learning algorithms in important dates extraction. It describes a date by up to six words before a date. If a date is followed by a colon then it contains up to six words after a date. The words from *date surrounding words* feature are used to calculate features for a current word. We create these features only for dates, because we do not want to increase the number of features too much.

3.3. Layout Features

Block feature informs about the blocks a word belongs to. We assign a separated value for each of the following blocks: head title, title, subtitle, paragraph, list, and table as the distribution over blocks differ for entities of interest. The number of a paragraph for a word is represented by *Paragraph number* feature. We consider only first six paragraphs because more than half of interesting entities is present in these paragraphs based on the corpus. This feature is important for conference names and abbreviations, dates and locations of conferences detection as these entities often occur at the beginning of a web page, according to our corpus. The important dates usually lie in further parts of a web page.

One of the subpages of a main conference homepage may contain entities of interest. Therefore, subpages are added to the training data. We restrict subpages to only those accessible by links with the following names: index, home, call for papers, registration, important dates. Moreover, each word from subpage is indicated by *subpage* feature containg anchor text, e.g., SUB=index.

Words modified by the following HTML tags: STRONG, B (bold), U (underlined), and FONT (use different fonts) are marked with *Emphasised* feature. This feature is meant for dates of a conference as they are more often underlined. Abbreviations and names of conferences do not show this regularity.

Links (A HTML tag) are represented by *Hyperlink* feature. This feature surprisingly indicates rather links to other conferences than information important in our task. Statistics calculated on our corpus confirm that.

3.4. Dictionary Features

Within *location* features, for a location in a web page a LOC=true, for a country COUNTRY=true and for a city CITY=true features are created. To calculate these features, gazetteer from ANNIE module of GATE (Kenter and Maynard, 2005) is used and location names from the corpus are added. The aforementioned features are helpful in conference location extraction.

Words that have not been found in the dictionary are marked with *Out of dictionary* feature. Our dictionary of English words contains 112505 words. This feature is designed for abbreviation extraction because this type of entity has the highest fraction of words not found in the dictionary.The feature suggests also location entity as it has the second highest value of not being found in the dictionary.

We created word dictionaries for place and date, name and abbreviation of a conference. They contain words that occur the most in sentences containing an important entity of a given type. Feature *promising surrounding words* marks words from sentences that contain at least one word from the dictionary. As the dictionaries are not mutually exclusive, *promising surrounding words* feature indicates that a word is important rather for entity extraction than for a specific entity type.

3.5. Multi-token Sequences

While describing features for our model, we assume that a single token; that is, a word, a number, or a nonalphanumeric character, is considered as a base object used by a model and assigned with one of interesting entity types, including *other* that means an object is not of one of the interesting entity types. This leads to a case when a sequence of tokens may have different entity types assigned even they are one entity of, e.g., conference name type. For instance, a sequence International Conference on Artificial Intelligence & Applications may have the following entity types assigned: International - conference name, Conference - conference name, on - other, Artificial - conference

name, and so on. Therefore, we expand a base object of a model to be a sequence of tokens that groups words forming one instance of entity. While detection of dates is an easy task, finding sequences that represent other named entities is not trivial. Hence, we prepared a heuristic algorithm customised for finding token sequences on conference web pages that is based on the following rules: each sequence consists of words that begin with a capital letter; these words may be separated by one word that starts with small letter; sequences are found within a sentence; a sequence cannot be separated by comma, dash nor colon. For example, words 'International Conference on Advancements in Information Technology' is treated by this algorithm as one sequence.

For sequences with at least two words we need to calculate features in one of the following ways: 1) calculate features for the first word only; 2) calculate features for each word separately and use all the features; 3) combine features for all words into one feature. For example, feature *word* is calculated according to the second approach and International Conference on Mechanics has the following features W=International, W=Conference, W=on, W=Mechanics. Third approach is used for POS features, e.g., 'Workshop on Applications of Software Agents' has a feature POS=INNNNNS.

4. Experiments

In our experiments we divide the corpus into training and test sets according to the proportion of 70/30. For Support Vector Machine (SVM) model (Cortes and Vapnik, 1995) cross validation is performed on the training set in order to find the best parameters, then the model is trained on the whole training set. We use LibSVM implementation (Chang and Lin, 2011). For multiclass classification we employ one versus the rest approach (Fan et al., 2008). For a web page we choose the only one entity of a given type that has the highest score among those indicated by an algorithm. Only *location* entity may have two instances because usually a country and a city is provided on a web page as a *location* of a conference.

4.1. Importance of Features

In our first group of experiments we verify using SVM how important the groups of features customised for information extraction from scientific conferences web pages are. We want to show how domain specific features influence the final results. As features in groups are sparse, a model with only one group of features would obtain very low accuracy and the comparison of models built with only one group of features would not be reliable. Therefore in each iteration we analyze all groups of features but one, in order to estimate how relevant is the group which was left out.

The results of the experiments are shown in Table 2. For all entity types but *Notification* we obtain the best results for all types of features included. For *Notification* we achieve the best results for the case without dictionary features, however, the results for all types of features case are not far behind (0.49 vs 0.46 in terms of F1). The results show that each group of features carries some information that is important for (at least one) interesting entity type. Thus, we could say that it is crucial to prepare features that are specific for a given domain. As the obtained results have shown, lack of some features may reduce the accuracy for some entity types to zero, for instance, lack of *offset* features for *important dates*.

For scientific conference web pages *local* features identify more general objects, such as dates and named entities that contain desired information. *Offset* features describe surroundings of a word, its context, which is necessary for *important dates* extraction. *Layout* features generate important features functions that inform on the localisation of a given word within a web page. They help in the case when an entity is not placed in the main text of a web page. *Dictionary* features improve the results mostly by its *location* feature that indicates potential places where a conference is held.

4.2. Models Comparison

Having the influence of features verified, we investigate the applicability of different models with regard to variations of their basic objects used; namely, single tokens, and sequences. In this set of experiments we use preprocessing and all the groups of features mentioned in Section 3..

For SVM model we start with comparison of single tokens and sequences used as basic objects that the model is working with. The results for linear SVM classifier run on single tokens as basic objects[1] are shown in the first row of Table 3. The accuracy of the model, also linear SVM, that uses sequences as basic objects is presented in the second row in the same table. The single token SVM performs significantly poorer than sequence SVM for *name* of a conference and *important dates*. The reason behind is that first model assigns a label to each single token independently and mentioned entities consists of several tokens. We try to ease SVM with this task by incorporating *offset* features, however, it seems that it is not enough to help single token SVM with extraction of entities that consist of several consecutive words. By providing the SVM already extracted potential sequences we overcome this problem. For sequence SVM we observe also 6 percentage points (p.p.) decrease in F1 for *abbreviation* detection, where linear SVM performs the best.

We present only the results of linear SVM because the non-linear SVM with RBF kernel function has not obtained significantly better results. Therefore, we stay with linear one due to less complexity and shorter training time. Our model has a high number of features, hence there is no need to increase the dimensionality by applying a kernel function (Hsu et al., 2003).

In our experiments we also use Linear Conditional Random Fields, CRF (Lafferty et al., 2001) with three different templates of factors. The first template connects factors with an input variable and an output variable. The second represents the relation between consecutive output variables. The third has only one argument that is an output variable. Single tokens CRF (Lin. CRF in Table 3) significantly outperforms both SVM models in *name* extraction (0.57 versus 0.36 and 0.15 in F1) due to the fact that it models sequences

[1] It means that the model assigns a label; that is, a type of entity, to a single token.

Features	Measure	Name	Abbrev.	Place	Date	Submission	Notification	Final ver. due
Lin. SVM	Precision	0.14	0.79	0.74	0.72	0.41	-	0.32
	Recall	0.16	0.86	0.59	0.79	0.06	0.00	0.08
	F1	0.15	**0.82**	0.66	0.76	0.11	0.00	0.13
Lin. SVM seq.	Precision	0.38	0.76	0.75	0.80	0.66	0.54	0.71
	Recall	0.34	0.75	0.60	0.80	0.54	0.40	0.59
	F1	0.36	0.76	**0.67**	0.80	0.60	0.46	**0.65**
Lin. CRF	Precision	0.74	0.75	0.66	0.82	0.73	0.25	0.56
	Recall	0.47	0.82	0.53	0.69	0.09	0.01	0.14
	F1	**0.57**	0.78	0.59	0.75	0.17	0.02	0.22
Lin. CRF seq.	Precision	0.61	0.77	0.66	0.82	0.67	0.63	0.70
	Recall	0.40	0.84	0.56	0.82	0.57	0.40	0.50
	F1	0.48	0.80	0.61	**0.82**	**0.61**	**0.49**	0.58

Table 3: The results of extraction for entities (the best F1 results marked in bold).

of label (SVM lacks this property). However, for entities that do not consist of several consecutive words we have not observed improvements; even contrary, we notice small decrease for *place* and *date*. Surprisingly, single token CRF cannot handle *important dates* extraction like in the case of single token SVM. However, sequence CRF (Lin. CRF seq. in Table 3) discovers them on the comparable level to sequence SVM. Both models based on sequences handle *important dates* significantly better because the sequence discovery algorithm extracts potential entities, that may have different formats, very well. Moreover, sequences also help CRF in *date* extraction (the best obtained results), like for SVM. Sequences discovery for *name* is not as good as discovering sequences for *important* dates. That is why we observe 9 p.p. decrease in extraction of that entity for CRF based on sequences compared to the one based on single tokens. However, sequences slightly increase CRF results for *abbreviation* and *place*.

Summarising, linear CRF based on single tokens outperforms other models for *name*. Linear SVM, also based on single tokens, obtains the best results for *abbreviation*. Dates are extracted better with models based on sequences than single tokens. For *place* the winner is SVM on both single tokens and sequences (SVM on sequences outperforms SVM on single tokens by only 1 p.p.), however, all other models are not worse than 8 p.p. in terms of F1. Thus, different models may be used for specific entity types in order to achieve the best cumulative results.

5. Related work

Previous works in that field focused mostly on information extraction from CFPs using different approaches. Extracting information from CFPs has drawbacks mentioned in Section 2.. In (Lazarinis, 1998) rule based method was employed to extract date and country from a CFP. Linear CRF was used in (Schneider, 2006) in order to extract seven attributes about conferences from CFPs with the use of layout features. However, in this approach only plain text of CFPs was used and layout features were based on lines of text, indicating, e.g., first token in line or first line in the text. We use HTML sourcecode of web pages, including formatting. Thus, our data has much more richer layout. In (Ireson et al., 2005) a general platform for performing and assessing information extraction from workshop CFPs was described. The platform was used in Pascal Challenge on Evaluating Machine Learning for Information Extraction. The organizers of the challenge provided a standardised corpus of CFPs, a set of tasks, and methodology for evaluation. The results of the challenge can be found in the aforementioned paper. Issertial and Tsuji (2011) focused also on information extraction from CFPs, including that which come via e-mails. They used rule-based methods to extract information about conferences from conference services, like WikiCFP, and combined them in one system in order to facilitate the process of finding conferences that are of interest for a user. In contrast to aforementioned works (Xin et al., 2008) extracted information about conferences from web pages with Constrained Hierarchical Conditional Random Fields. However, the set of homepages used in experiments has not been published. We created the annotated corpus, performed extraction and made both the corpus and the results public in order to encourage researchers to improve the baseline for this corpus.

In information extraction many approaches have been proposed. One of them is a rule-based method employed in (Ciravegna, 2001; Hazan and Andruszkiewicz, 2013). Support Vector Machines (SVM) classifier was applied to information extraction from web pages also (Andruszkiewicz and Nachyla, 2013). A variety of Conditional Random Fields (CRF) methods were widely used (Tang et al., 2008; Wu and Weld, 2010; Li et al., 2011; Rocktäschel et al., 2013; Wang and Feng, 2013; Andruszkiewicz and Nachyla, 2013; Cuong et al., 2015). Constrained CRF applied in (Xin et al., 2008) allows a miner for specifying constrains for extracted entities. Furthermore, Markov Logic Networks (MLNs) were used in information extraction from web pages (Andruszkiewicz and Nachyla, 2013).

6. Conclusions and Future Work

To sum up, we created the corpus of 943 annotated homepages of scientific conferences and make it publicly available. Moreover, we performed the experiments with single- and multi-token SVM and CRF for this set in order to set a baseline for this corpus.

In future work, we plan to apply other models, e.g., hierarchical CRF, MLNs, to obtain better results. Especially, we want to focus on *important dates* extraction by experimenting with different models and gathering more instances of these entity types. We also would like to extend our corpus by adding new conferences and annotations, e.g., chairs, committee members, in order to encourage researchers to make experiments on our corpus.

7. Bibliographical References

Andruszkiewicz, P. and Nachyla, B. (2013). Automatic extraction of profiles from web pages. In *Intelligent Tools for Building a Scientific Information Platform - Advanced Architectures and Solutions*, pages 415–431.

Califf, M. E. and Mooney, R. J. (1999). Relational learning of pattern-match rules for information extraction. In Jim Hendler et al., editors, *Proceedings of the Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Applications of Artificial Intelligence, July 18-22, 1999, Orlando, Florida, USA.*, pages 328–334. AAAI Press / The MIT Press.

Chang, C. and Lin, C. (2011). LIBSVM: A library for support vector machines. *ACM TIST*, 2(3):27.

Ciravegna, F. (2001). $(LP)^2$, an adaptive algorithm for information extraction from web-related texts. In *Proceedings of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Cuong, N. V., Chandrasekaran, M. K., Kan, M., and Lee, W. S. (2015). Scholarly document information extraction using extensible features for efficient higher order semi-CRFs. In Paul Logasa Bogen II, et al., editors, *Proceedings of the 15th ACM/IEEE-CE Joint Conference on Digital Libraries, Knoxville, TN, USA, June 21-25, 2015*, pages 61–64. ACM.

Fan, R., Chang, K., Hsieh, C., Wang, X., and Lin, C. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Freitag, D. and McCallum, A. K. (1999). Information extraction with HMMs and shrinkage. In *In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction*, pages 31–36.

Hazan, R. and Andruszkiewicz, P. (2013). Home pages identification and information extraction in researcher profiling. In *Intelligent Tools for Building a Scientific Information Platform - Advanced Architectures and Solutions*, pages 41–51.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2003). A practical guide to support vector classification.

Ireson, N., Ciravegna, F., Califf, M. E., Freitag, D., Kushmerick, N., and Lavelli, A. (2005). Evaluating machine learning for information extraction. In Luc De Raedt et al., editors, *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pages 345–352. ACM.

Issertial, L. and Tsuji, H. (2011). Information extraction and ontology model for a 'call for paper' manager. In David Taniar, et al., editors, *iiWAS'2011 - The 13th International Conference on Information Integration and Web-based Applications and Services, 5-7 December 2011, Ho Chi Minh City, Vietnam*, pages 539–542. ACM.

Kenter, T. and Maynard, D. (2005). Using GATE as an Annotation Tool, January.

Kohlschütter, C., Fankhauser, P., and Nejdl, W. (2010). Boilerplate detection using shallow text features. In Brian D. Davison, et al., editors, *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010*, pages 441–450. ACM.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Carla E. Brodley et al., editors, *ICML*, pages 282–289. Morgan Kaufmann.

Lazarinis, F. (1998). Combining information retrieval with information extraction for efficient retrieval of calls for papers. In *20th Annual BCS-IRSG Colloquium on IR, Autrans, France. 25th-27th March 1998*, Workshops in Computing. BCS.

Li, Y., Jiang, J., Chieu, H. L., and Chai, K. M. A. (2011). Extracting relation descriptors with conditional random fields. In *Fifth International Joint Conference on Natural Language Processing, IJCNLP 2011, Chiang Mai, Thailand, November 8-13, 2011*, pages 392–400. The Association for Computer Linguistics.

McCallum, A., Schultz, K., and Singh, S. (2009). FACTORIE: probabilistic programming via imperatively defined factor graphs. In Yoshua Bengio, et al., editors, *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 1249–1257. Curran Associates, Inc.

Porter, M. F. (2001). Snowball: A language for stemming algorithms.

Rocktäschel, T., Huber, T., Weidlich, M., and Leser, U. (2013). Wbi-ner: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs. In *Proceedings of the 7th International Workshop on Semantic Evaluation*, pages 356–363.

Schneider, K. (2006). Information extraction from calls for papers with conditional random fields and layout features. *Artif. Intell. Rev.*, 25(1-2):67–77.

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. (2008). Arnetminer: extraction and mining of academic social networks. In Ying Li, et al., editors, *KDD*, pages 990–998. ACM.

Wang, G. and Feng, X. (2013). Tool wear state recognition based on linear chain conditional random field model. *Eng. Appl. of AI*, 26(4):1421–1427.

Wu, F. and Weld, D. S. (2010). Open information extraction using wikipedia. In Jan Hajic, et al., editors, *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16,*

2010, Uppsala, Sweden, pages 118–127. The Association for Computer Linguistics.

Xin, X., Li, J., Tang, J., and Luo, Q. (2008). Academic conference homepage understanding using constrained hierarchical conditional random fields. In James G. Shanahan, et al., editors, *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, California, USA, October 26-30, 2008*, pages 1301–1310. ACM.

WikiRank:Improving Keyphrase Extraction Based on Background Knowledge

Yang Yu, Vincent Ng
Human Language Technology Research Institute
University of Texas at Dallas
{yangyu,vince}@hlt.utdallas.edu

Abstract

Keyphrase is an efficient representation of the main idea of documents. While background knowledge can provide valuable information about documents, they are rarely incorporated in keyphrase extraction methods. In this paper, we propose WikiRank, an unsupervised method for keyphrase extraction based on the background knowledge from Wikipedia. Firstly, we construct a semantic graph for the document. Then we transform the keyphrase extraction problem into an optimization problem on the graph. Finally, we get the optimal keyphrase set to be the output. Our method obtains improvements over other state-of-art models by more than 2% in F1-score.

Keywords: Keyphrase Extraction, Knowledge Graph, Semantic Graph

1. Introduction

As the amount of published material rapidly increases, the problem of managing information becomes more difficult. Keyphrase, as a concise representation of the main idea of the text, facilitates the management, categorization, and retrieval of information. Automatic keyphrase extraction concerns "the automatic selection of important and topical phrases from the body of a document". Its goal is to extract a set of phrases that are related to the main topics discussed in a given document(Hasan and Ng, 2014).

Existing methods of keyphrase extraction could be divided into two categories: supervised and unsupervised. While supervised approaches require human labeling, at the same time needs various kinds of training data to get better generalization performance, more and more researchers focus on unsupervised methods.

Traditional methods of unsupervised keyphrase extraction mostly focus on getting information of document from word frequency and document structure(Hasan and Ng, 2014), however, after years of attempting, the performance seems very hard to be improved any more. Based on this observation, it is reasonable to suspect that the document itself possibly cannot provide enough information for keyphrase extraction task.

To get good coverage of the main topics of the document, Topical PageRank (Liu et al., 2010) started to adopt topical information in automatic keyphrase extraction. The main idea of Topical PageRank is to extract the top topics of the document using LDA, then sum over the scores of a candidate phrase under each topic to be the final score. The main problems with Topical PageRank are: First, The topics are too general. Second, since they are using LDA, they only classify the words to several topics, but don't know what the topics exactly are. However, the topical information we need for keyphrase extraction should be precise. As shown in Figure 1, the difference between a correct keyphrase *sheep disease* and an incorrect keyphrase *incurable disease* could be small, which is hard to be captured by rough topical categorization approach.

To overcome the limitations of aforementioned approaches, we propose WikiRank, an unsupervised automatic keyphrase extraction approach that links semantic meaning to text

The key contribution of this paper could be summarized as follows:

1. We leverage the topical information in knowledge bases to improve the performance of keyphrase extraction.

2. We model the keyphrase extraction as an optimization problem, and provide the corresponding solution as well as a pruning approach to reduce the complexity.

2. Existing Error Illustration with Example

Figure 1 shows part of an example document[1]. In this figure, the gold keyphrases are marked with bold, and the keyphrases extracted by the TextRank system are marked with parentheses. We are going to illustrate the errors exist in most of present keyphrase extraction systems using this example.

Overgeneration errors occur when a system correctly predicts a candidate as a keyphrase because it contains a word that frequently appears in the associated document, but at the same time erroneously outputs other candidates as keyphrases because they contain the same word(Hasan and Ng, 2014). It is not easy to reject a non-keyphrase containing a word with a high term frequency: many unsupervised systems score a candidate by summing the score of each of its component words, and many supervised systems use unigrams as features to represent a candidate. To be more concrete, consider the news article in Figure 1. The word Cattle has a significant presence in the document. Consequently, the system not only correctly predict *British cattle* as a keyphrase, but also erroneously predict *cattle industry*, *cattle feed*, and *cattle brain* as keyphrases, yielding overgeneration errors.

[1]Document from DUC-2001 Dataset *AP900322-0200 Government Boosts Spending to Combat Cattle Plague*

[2]Prefix "wiki" represents the namespace "https://en.wikipedia.org/wiki/"

(**Mad cow disease**) has killed 10,000 cattle, restricted the **export** market for Britain's (cattle industry)
wiki:Bovine_spongiform_encephalopathy wiki:Death wiki:Cattle wiki:Export wiki:Market_(economics) wiki:British_Empire wiki:Agribusiness

and raised fears about the safety of eating beef. The **government** insists the disease poses only a remote risk to human
wiki:Fear wiki:Safety wiki:Eating wiki:Beef wiki:Government wiki:Disease wiki:Risk

health, but scientists still aren't certain what causes the disease or how it is transmitted ... (**Mad cow disease**), or
wiki:Health wiki:Scientist wiki:Causality wiki:Transmission_(medicine) wiki:Bovine_spongiform_encephalopathy

Bovine Spongiform Encephalopathy, or **BSE**, was diagnosed only in 1986. The symptoms are very much like **scrapie**, a
wiki:Bovine_spongiform_encephalopathy wiki:Bovine_spongiform_encephalopathy wiki:Symptom wiki:Scrapie

(**sheep disease**) which has been in Britain since the 1700s. The (incurable disease) eats holes in the brains of its victims;
wiki:Sheep wiki:Great Britain wiki:Cure wiki:Cannibalism wiki:Human_Brain

in late stages a sick animal may act skittish or stagger drunkenly ... The **government** banned the use of sheep offal in
wiki:Disease wiki:Animal wiki:Government wiki:Ban_(law) wiki:Offal

(cattle feed) in June 1988, and later banned the use of (cattle brain), spleen ... has proposed a ban on **exports** of (**British**
wiki:Brain wiki:Spleen wiki:Ban_(law) wiki:Export wiki:United_Kingdom

cattle) older than 6 months ... has complained of "**BSE** hysteria" in the media and has insisted that the risk of the (disease
wiki:Cattle wiki:Bovine_spongiform_encephalopathy wiki:Mass_hysteria wiki:Mass_media wiki:Risk wiki:Disease

passing) to humans is "remote." ... known as (Creutzfeldt Jakob disease). About two dozen cases were reported in (Britain
wiki:Human wiki:Creutzfeldt-Jakob_disease wiki:Great_Britain

last year).

Figure 1: Part of the Sample Document [2]

Bold: Gold Keyphrase In parentheses: Keyphrase generated by TextRank algorithm Underlined: Keyphrase annotated to Wikipedia Entity by TagMe

Redundancy errors occur when a system correctly identifies a candidate as a keyphrase, but at the same time outputs a semantically equivalent candidate (e.g., its alias) as a keyphrase. This type of error can be attributed to the failure of a system to determine that two candidates are semantically equivalent. Nevertheless, some researchers may argue that a system should not be penalized for redundancy errors because the extracted candidates are in fact keyphrases. In our example, *bovine spongiform encephalopathy* and *bse* refer to the same concept. If a system predicts both of them as keyphrases, it commits a redundancy error.

Infrequency errors occur when a system fails to identify a keyphrase owing to its infrequent presence in the associated document. Handling infrequency errors is a challenge because state-of-the-art keyphrase extractors rarely predict candidates that appear only once or twice in a document. In the *Mad cow disease* example, the keyphrase extractor fails to identify *export* and *scrapie* as keyphrases, resulting in infrequency errors.

3. Proposed Model

The WikiRank algorithm includes three steps: (1) Construct the semantic graph including concepts and candidate keyphrases; (2)(optional) Prune the graph with heuristic to filter out candidates which are likely to be erroneously produced; (3) Generate the best set of keyphrases as output.

3.1. Graph Construction

3.1.1. Automatic Concept Annotation

This is one of the crucial steps in our paper that connects the plain text with human knowledge, facilitating the understanding of semantics. In this step, we adopt *TAGME* (Ferragina and Scaiella, 2010) to obtain the underlying concepts in documents.

TAGME is a powerful topic annotator. It identifies meaningful sequences of words in a short text and link them to a pertinent Wikipedia page, as shown in Figure 1. These links add a new topical dimension to the text that enable us to relate, classify or cluster short texts.

3.1.2. Lexical Unit Selection

This step is to filter out unnecessary word tokens from the input document and generate a list of potential keywords using heuristics. As reported in (Hulth, 2003), most manually assigned keyphrases turn out to be noun groups. We follow (Wan and Xiao, 2008a) and select candidates lexical unit with the following Penn Treebank tags: NN, NNS, NNP, NNPS, and JJ, which are obtained using the Stanford POS tagger (Toutanova et al., 2003), and then extract the noun groups whose pattern is zero or more adjectives followed by one or more nouns. The pattern can be represented using regular expressions as follows

$$(JJ) * (NN|NNS|NNP|NNPS)+$$

where JJ indicates adjectives and various forms of nouns are represented using NN, NNS and NNP .

3.1.3. Graph building

We build a semantic graph $G = [V; E]$ in which the set of vertices V is the union of the concept set C and the candidate keyphrase set P—i.e., $V = P \cup C$. In the graph, each unique concept $c \in C$ or candidate keyphrase $p \in P$ for document d corresponds to a node. The node corresponds to a concept c and the node corresponds to a candidate keyphrase p are connected by an edge $(c, p) \in E$, if the candidate keyphrase p contains concept c according to the annotation of *TAGME*. Part of the semantic graph of the sample document is shown in Figure 2. Concepts corresponding to 2 are shown in Table 1.

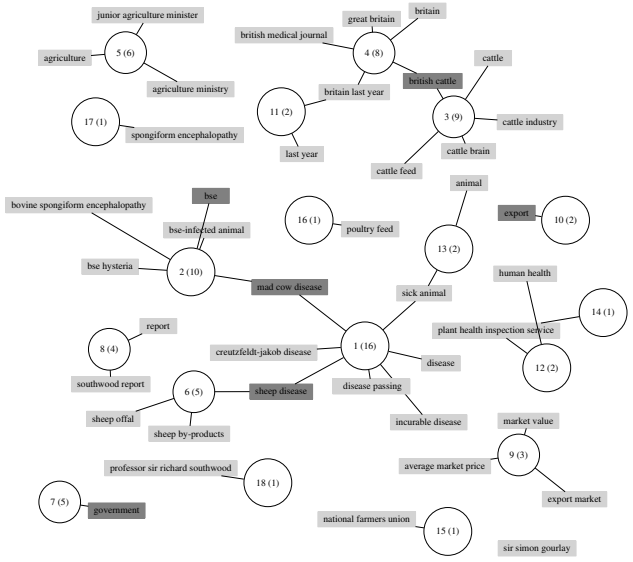


Figure 2: Part of the Semantic Graph of the Sample Document

Circle: Concept Rectangle: Candidate Keyphrase

Dark Rectangle: Gold Keyphrase

#	Concept	Frequency
1	Disease	16
2	Bovine spongiform encephalopathy	10
3	Cattle	9
4	Great Britain	8
5	United States Department of Agriculture	6
6	Sheep	5
7	Government	5
8	Report	4
9	Market (economics)	3
10	Export	2
11	Last year	2
12	Health	2
13	Animal	2
14	Animal and Plant Health Inspection Service	1
15	National Farmers Union of England and Wales	1
16	Poultry feed	1
17	Transmissible spongiform encephalopathy	1
18	Professor	1

Table 1: Part of the Concepts Annotated from the Sample Document

3.2. WikiRank

3.2.1. Optimization Problem

According to (Liu et al., 2010), good keyphrases should be relevant to the major topics of the given document, at the same time should also have good coverage of the major topics of the document. Since we represent the topical information with concepts annotated with *TAGME*, the goal of our approach is to find the set Ω consisting of k keyphrases, to cover concepts (1) as important as possible (2) as much as possible.

Let w_c denote the weight of concept $c \in C$. We compute w_c as the frequency c exists in the whole document d. To quantify how good the coverage of a keyphrase set Ω is, we compute the overall score of the concepts that Ω contains.

Consider a subgraph of G, G_{sub}, which captures all the concepts connected to Ω. In G_{sub}, the set of vertices V_{sub} is the union of the candidate keyphrase set Ω, and the set Adj_Ω of concepts that nodes in Ω connect to. The set of edges E_{sub} of G_{sub} is constructed with the edges connect nodes in Ω with nodes in Adj_Ω.

We set up the score of a concept c in the subgraph G_{sub} as following:

$$S(c) = \sum_{i=0}^{deg(c)} \frac{w_c}{2^i} \tag{1}$$

where w_c is the weight of c as we defined before, and $deg(c)$ is the degree of c in the subgraph G_{sub}. Essentially, $deg(c)$ is equal to the frequency that concept c is annotated in the keyphrase set Ω.

The optimization problem is defined as:

$$\begin{aligned} \max_{\Omega} \quad & \sum_{c \in Adj_\Omega} S(c) \\ \text{s.t.} \quad G_{sub} &= [V_{sub}; E_{sub}] \\ V_{sub} &= \Omega \cup Adj_\Omega \\ E_{sub} &= \{(c,p) | p \in \Omega, c \in Adj_\Omega\} \\ Adj_\Omega &= \{c | c \in \textstyle\sum_{p \in \Omega} Adj(p)\} \\ |\Omega| &\leq k \end{aligned} \tag{2}$$

The goal of the optimization problem is to find the candidate keyphrase set Ω, such that the sum of the scores of the concepts annotated from the phrases in Ω is maximized.

3.2.2. Algorithm

We propose an algorithm to solve the optimization problem, as shown in Algorithm 1. In each iteration, we compute the score s_p for all candidate keyphrases $p \in |P|$ and include the p with highest score into Ω, in which s_p evaluates the score of concepts added to the new set Ω by adding p into Ω.

3.3. Approximation Approach with Pre-pruning

In practice, computing score for all the candidate keyphrases is not always necessary, because some of the candidates are very unlikely to be gold keyphrase that we can remove them from our graph before applying the algorithm to reduce the complexity.

In this section, we introduce three heuristic pruning steps that significantly reduces the complexity of the optimization problem without reducing much of the accuracy.

Step 1. Remove the candidate keyphrase p from original graph G, if it is not connected to any concept.

The intuition behind this heuristic is straightforward. Since our objective function is constructed over concepts, if a candidate keyphrase p doesn't contain any concept, adding it to Ω doesn't bring any improvement to the objective function, so p is irrelevant to our optimization process. Pruning p would be a wise decision.

	DUC			Inspec			ICSI			Nus		
	P	R	F score	P	R	F score	P	R	F score	P	R	F score
SingleRank	26.21	24.45	25.30	25.21	24.10	24.64	3.42	2.49	2.88	0.23	0.98	0.37
Topical PageRank	27.33	23.92	25.51	25.58	24.31	24.93	3.98	2.68	3.20	0.64	1.38	0.87
Our System	28.72	26.44	27.53	28.14	25.97	27.01	4.71	3.96	4.30	7.27	12.16	9.10

Table 2: The Result of our System as well as the Reimplementation of SingleRank and Topical PageRank on four Corpora

Algorithm 1 Keyphrase Generalization

Input:
$|C|, P, W = \{w_1, \ldots, w_{|C|}\}$
k: ▷ Size of output keyphrase set
$M_{|P|\times|C|}$: ▷ Adjacency matrix
Output:
Ω ▷ The set of selected keyphrases
Initialization:
$\Omega \leftarrow \emptyset$
$S = \{s_1 \leftarrow 0, \ldots, s_{|P|} \leftarrow 0\}$

while $|\Omega| < k$ **do**
  for $p = 1$ to $|P|$ **do**
    $s_p \leftarrow 0$
    for $c = 1$ to $|C|$ **do**
      if $M_{p,c} \neq 0$ **then**
        $s_p = s_p + w_c$
      end if
    end for
  end for
  $q \leftarrow \arg\max_{q=1\ldots|P|}\ s_q$
  $\Omega \leftarrow \Omega \cup \{P_q\}$
  for $c = 1$ to $|C|$ **do**
    if $M_{q,c} \neq 0$ **then**
      $w_c \leftarrow w_c/2$
    end if
  end for
end while
return Ω

Step 2. Remove the candidate keyphrase p from original graph G, if it is only connected to one concept that only exists once in the document

If a candidate keyphrase contains fewer concepts, or the concepts connects to it barely exist in the document, we think this candidate keyphrase contributes less valuable information to the document. In practice, there are numerous (c, p) pairs in graph G that is isolated from the center of the graph. We believe they are irrelevant to the major topic of the document.

Step 3. For a concept c connecting to more than m candidate keyphrases, remove any candidate keyphrase $p \in Adj(c)$ which (1)Does not connect to any other concept. AND (2)The ranking is lower than mth among all candidate keyphrases connect to c.(In practice, m is usually 3 or 4.)

According to equation 1, if there are already m instances of concept c in the G_{sub}, adding the $m+1$th instance of c will only contribute $\frac{w_c}{2^m}$ to $S(c)$. At the same time, among all the candidate keyphrases connected to concept c, our optimization process always chooses the ones that connect to other concepts as well over the ones that do not connect to any other concept. Combining these two logic, a candidate satisfying the constrains of Step 3 is not likely to be picked in the best keyphrase set Ω, so we can prune it before the optimalization process.

4. Experiments and Results

4.1. Corpora

The **DUC-2001** dataset (Over, 2001), which is a collection of 308 news articles, is annotated by (Wan and Xiao, 2008b).

The **Inspec** dataset is a collection of 2,000 abstracts from journal papers including the paper title. This is a relatively popular dataset for automatic keyphrase extraction, as it was first used by (Hulth, 2003) and later by Mihalcea and (Mihalcea and Tarau, 2004) and (Liu et al., 2009).

The **NUS Keyphrase Corpus** (Nguyen and Kan, 2007) includes 211 scientific conference papers with lengths between 4 to 12 pages. Each paper has one or more sets of keyphrases assigned by its authors and other annotators. The number of candidate keyphrases that can be extracted is potentially large, making this corpus the most challenging of the four.

Finally, the **ICSI Meeting Corpus** (Janin et al., 2003), which is annotated by Liu et al. (2009a), includes 161 meeting transcriptions. Unlike the other three datasets, the gold standard keys for the ICSI corpus are mostly unigrams.

4.2. Result

For comparing with our system, we reimplemented SingleRank and Topical PageRank. Table 2 shows the result of our reimplementation of SingleRank and Topical PageRank, as well as the result of our system. Note that we predict the same number of phrase ($k = 10$) for each document while testing all three methods.

The result shows our result has guaranteed improvement over SingleRank and Topical PageRank on all four corpora.

5. Conclusion and Future Work

We proposed an unsupervised graph-based keyphrase extraction method WikiRank. This method connects the text with concepts in Wikipedia, thus incorporate the background information into the semantic graph and finally construct a set of keyphrase that has optimal coverage of the concepts of the document. Experiment results show the method outperforms two related keyphrase extraction methods.

We suggest that future work could incorporate more other semantic approaches to investigate keyphrase extraction

task. Introducing the results of dependency parsing or semantic parsing (e.g., OntoUSP) in intermediate steps could be helpful.

6. Bibliographical References

Ferragina, P. and Scaiella, U. (2010). Tagme: On-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1625–1628, New York, NY, USA. ACM.

Hasan, K. S. and Ng, V. (2014). Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland, June. Association for Computational Linguistics.

Hulth, A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, pages 216–223, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liu, Z., Li, P., Zheng, Y., and Sun, M. (2009). Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 257–266, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liu, Z., Huang, W., Zheng, Y., and Sun, M. (2010). Automatic keyphrase extraction via topic decomposition. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 366–376, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. In *Proceedings of Empirical Methods for Natural Language Processing*, pages 404–411.

Nguyen, T. D. and Kan, M.-Y. (2007). Keyphrase extraction in scientific publications. In *Proceedings of the 10th International Conference on Asian Digital Libraries: Looking Back 10 Years and Forging New Frontiers*, ICADL'07, pages 317–326, Berlin, Heidelberg. Springer-Verlag.

Over, P. (2001). Introduction to DUC-2001: An intrinsic evaluation of generic news text summarization systems. In *Proceedings of the 2001 Document Understanding Conference*.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wan, X. and Xiao, J. (2008a). Collabrank: Towards a collaborative approach to single-document keyphrase extraction. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 969–976, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wan, X. and Xiao, J. (2008b). Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, pages 855–860. AAAI Press.

WikiDragon: A Java Framework For Diachronic Content And Network Analysis Of MediaWikis

Rüdiger Gleim[1], **Alexander Mehler**[1], **Sung Y. Song**[2]
Text-Technology Lab, Goethe University[1]
Robert-Mayer-Straße 10, 60325 Frankfurt
{gleim, mehler}@em.uni-frankfurt.de, gnosygnu@gmail.com[2]

Abstract

We introduce WikiDragon, a Java Framework designed to give developers in computational linguistics an intuitive API to build, parse and analyze instances of MediaWikis such as Wikipedia, Wiktionary or WikiSource on their computers. It covers current versions of pages as well as the complete revision history, gives diachronic access to both page source code as well as accurately parsed HTML and supports the diachronic exploration of the page network. WikiDragon is self-enclosed and only requires an XML dump of the official Wikimedia Foundation website for import into an embedded database. No additional setup is required. We describe WikiDragon's architecture and evaluate the framework based on the simple English Wikipedia with respect to the accuracy of link extraction, diachronic network analysis and the impact of using different Wikipedia frameworks to text analysis.

Keywords: Wikipedia, Java Framework, Diachronic Text Analysis, Diachronic Network Analysis, Link Extraction

1. Introduction

Wikis such as Wikipedia, Wiktionary, WikiSource or WikiNews constitute a popular form of collaborative writing and thus a valuable resource for computational linguistics. This relates to research exploiting wiki data as a resource, e.g. for word sense disambiguation (Uslu et al., 2018; Dandala et al., 2013), explicit semantic analysis (Gabrilovich and Markovitch, 2007) or training data for named entity recognition (Nothman et al., 2008). Furthermore, wikis are also considered as a research object on their own, for example with respect to measuring the impact of authors and edits (Priedhorsky et al., 2007), studies on discourse structure (Mehler et al., 2018) as well as on edit categories (Daxenberger and Gurevych, 2012) or dynamics of conflicts in Wikipedias (Yasseri et al., 2012). The Wiki principle is also used to publish research results like digital editions of texts as Wikiditions (Mehler et al., 2016).

Generally speaking, researchers need efficient as well as manageable means to access wiki data. However, one has to face the challenge that this data is not limited to articles but also includes discussions, pages in other namespaces such as portals, rich metadata, revision histories, information about (anonymous or registered) writers as well as about the link-based networking of pages. For small queries, using the Web API of wikis to collect data may be feasable. But in order to analyse large amounts of data (e.g. all articles of the main namespace) the approach to work online is inadequate because of the intense use of bandwith and server load. The alternative is to work offline based on local reconstructions of wikis. Notice that this scenario induces a big data problem: take the example of the German Wikipedia: as of 2016-02-03 its pages sum up to 23.84 GB of uncompressed data (current versions only). Its history pages sum-up to 2.92 TB of uncompressed data. Building an offline instance of a wiki based on the MediaWiki software requires extensive work to setup the environment, ensuring that all plugins are installed and importing the official XML dumps into a local MySQL database. Alternative approaches to work with wikis offline lack proper interfaces to explore the data or cover only certain aspects: for example by modeling only a subset of all namespaces (e.g. the main namespace and categories) or by developing interfaces to special instances such as Wiktionary.

Wikimedia's XML dumps, which are the only and most complete data sets available contain revision texts in a proprietary markup which needs to be processed to get HTML. This process is a challenging task, since MediaWiki allows for recursively including templates, uses parser functions and even a scripting language (Lua). Ignoring these particularities entails a high risk of information loss. This also holds, for example, for text network analysis, since page links are only provided as part of SQL dumps which, in contrast to the latter XML dumps, only map the latest state of text linkage. Thus, reconstructing past states of a wiki's article network cannot be done by referring to SQL dumps.

WikiDragon aims to solve the challenge of providing (nearly) complete representations of wikis offline *and* deliver a Java API which meets all requirements to import, represent, extract and analyze MediaWikis in a diachronic manner with respect to text, authorship, network structure and metadata. It is self-contained, runs on Windows as well as Linux and does not require any additional setup. Java developers can use WikiDragon to focus on their research question and code extensions fitting their needs.

Section 1.1. gives an overview of related work in this field. Section 2. points out special features of WikiDragon's architecture and sketches the essential qualities of the Core API. Section 3. provides an evaluation by measuring computation time and required space to build a database, by measuring the accuracy of link extraction against a gold standard, by performing an exemplary diachronic network analysis and by examining the impact of variants of markup parsing to subsequent text analysis. Section 4. gives a summary and a prospect of future work.

1.1. Related Work

There is a vast number of alternative parsers to MediaWiki markup [1] and related tools to extract specific information. Simple approaches attempt to convert MediaWiki Markup into plain text or HTML by applying replacement rules such as regular expressions. More sophisticated approaches also try to implement functions and templates. Nonetheless such advanced frameworks still lack supporting the MediaWiki extension Scribuntu which allows the use of the scripting language Lua. This becomes an issue especially in Wiktionary where whole conjugation and declination tables are written in Lua. The *Sweble Wikitext Parser*[2] (Dohrn and Riehle, 2011) is a Java framework to transform MediaWiki markup into a full fledged object model which can then be transformed into plaintext or HTML. Sweble supports template expansion and a selection of parser functions. The *Java Wikipedia API (Bliki engine)*[3] takes it a step further by providing support for Lua. XOWA[4] is an offline reader for Wikimedia wikis such as Wikipedia and Wiktionary, dedicated to accurate rendering of MediaWiki markup. It provides template expansion, parser functions and processing of Lua code. The project consists of the *XOWA App* and the underlying *XOWA Embeddable Parser*. The offline reader is implemented in Java and based on SWT and Mozilla XULRunner for visualization and HTML rendering. SQLite is used as database backend and Apache Lucene for indexing. The Embeddable Parser is based on Java, Lua and JTidy. It provides an API which allows it to be used for MediaWiki source parsing by other applications bringing their own data model and store. WikiDragon uses XOWA Embeddable Parser to provide (nearly) accurate HTML rendering of Wiki pages and to extract network structures induced by page links.

The *Java Wikipedia Library (JWPL)* (Zesch et al., 2008; Ferschke et al., 2011) is a framework which, based on a MySQL database, provides local access to Wikipedia instances. It includes Sweble to provide plain text versions of articles as well as access to the object model. The standard integration of Sweble into JWPL does not include template expansion. Current versions of a Wikipedia can be build based on dumps using the *JWPL DataMachine* and imported into MySQL. The *JWPL Core* is supplemented by the *JWPL RevisionMachine* to get access to the edit history whereas *JWPL TimeMachine* provides means to create automatized snapshots of a Wikipedia in a specific time interval. Both data model and API of JWPL Core focus by design on the Wikipedia and on the Article- and Category Namespace (including discussion). JWPL supports the import of SQL dumps representing article as well as category links and provide access to this network structure by means of the Java API. Past states of contents can be generated via RevisionMachine or TimeMachine. However it is not possible to reconstruct past states of the network structure as SQL dumps only represent the current state of pages.

Java Wiktionary Library (JWKTL) (Zesch et al., 2008) is a related but stand-alone project which uses an embedded BerkeleyDB to create a parsed lexicon based on a Wiktionary XML dump for English, German and Russian.

WikiWho (Flöck and Acosta, 2014) is an algorithm to compute token-level provenance and changes for Wiki revisioned content. The approach thus focuses on the aspect of the authorship of text components at token level. The system can be used via a web service[5]. WikiWho does not expand templates nor does it track links of pages.

DBpedia (Lehmann et al., 2015) is a community project to extract structured information from Wikipedia in different languages on large scale. Parsed information is represented in RDF which can either be downloaded or queried online via SPARQL. Thus the focus is on explicating information which is hidden in uni- or semi-structured text. Even if extraction- and data projects use their own pipelines for the acquisition and processing of raw data, there could be added value in quickly making any past state available offline using WikiDragon. That way, the information growth of a Wikipedia instance could be investigated over time. In this context, the integration of XOWA for the translation of wiki markup into HTML could is also interesting to related projects.

WikiDragon goes beyond the objectives of offline readers for an accurate representation of the current state by including all content and meta data available to current and past states. Furthermore it is unique in the way it combines the design goal of complete coverage with access through a consistently object-oriented Java API, which can be adapted to any database system via interface implementation.

2. WikiDragon Architecture

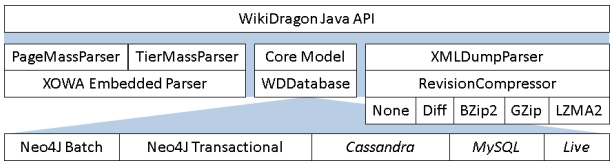


Figure 1: Architecture of WikiDragon depicting the layer model of the system. Italic components are subject of future work.

WikiDragon is designed on the principle to focus on accessing a collection of MediaWiki instances by means of an intuitive Java API while minimizing efforts for setting up the system environment. Figure 1 depicts the general architecture of the framework. The WikiDragon Java API marks the top layer of the architecture. This is the primary interface for users to work on Wikis. Importing, compressing and representing Wiki instances is an integral part of the system: A new Wiki is imported by providing a path to a compressed XML dump file (either current revisions only, or the full history version) to the XMLDumpParser. The import applies multi-threading to speed up the compression of revisions. The most effective compression is achieved

[1] MediaWiki maintains a list of alternative parser on https://www.mediawiki.org/wiki/Alternative_parsers

[2] https://osr.cs.fau.de/software/sweble-wikitext/

[3] https://bitbucket.org/axelclk/info.bliki.wiki/

[4] http://xowa.org/

[5] https://api.wikiwho.net/en/api/

by only storing differences between subsequent page revisions. However, other compression techniques like BZip2, LZMA2 or none at all are also available and may be used when disc space is less an issue. Another important component of WikiDragon is the integration of XOWA for ad hoc or multi-threaded mass parsing of page-revisions or diachronic snapshots called *PageTiers*.

WikiDragon supports multiple database systems by design to be able to provide the best fitting paradigm for a given user environment or task: An embedded database requires no setup, is ideal for a user working on a workstation and yet suitable to host a collection of small or medium-size Wikis or one large instance. Server-based DBMS require additional setup but have the opportunity to detach the experimentation environment from the storage layer. Large scale distributed DBMS like Cassandra are perfect to host multiple large Wikis and still access them by using the same Java API as for the embedded databases. Currently there exist two implementations which use the embeddable graph database management system Neo4J as backend: One variant is based on the batch-inserter interface of Neo4J which is fast, but at the cost of reduced features such as lack of transactions and deletion of elements. The second variant is based on the transactional interface of Neo4J and provides concurrent access which makes it suitable to build a webservice on top of WikiDragon. Note that the integration of Cassandra, MySQL and the Live-mode (based on Web APIs) are subject of future work.

2.1. Process Chain

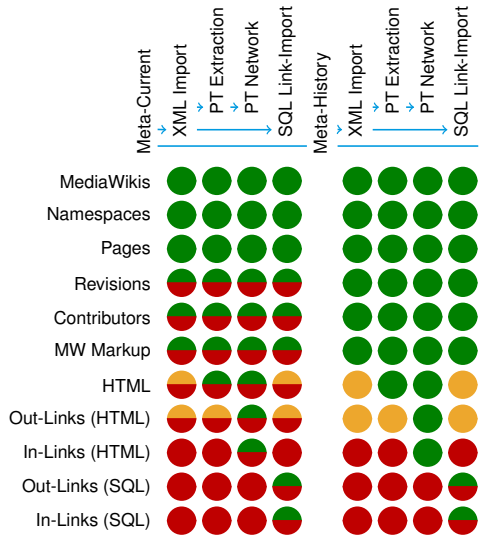


Figure 2: Resource-Feature matrix of WikiDragon for two XML dump variants *meta-current* and *meta-history*. Green denotes direct availability, red unavailability and orange indicates that the information can be parsed ad hoc (not necessarily efficient). Split circles distinguish availability with regard to current (top) or history (bottom) version.

Figure 2 depicts the availability of API features with respect to the XML dump used and the processing steps executed. Using the Meta-Current XML dump is the fastest way to build up a database as it contains only the most recent version of each page. In contrast the Meta-History dump takes considerably more time for import but allows diachronic access. Most features are available directly after the XML import. HTML can be parsed ad hoc (marked yellow in the matrix), which is suitable for small amounts of access. For mass text processing, performing a PageTier (PT) extraction is required which represents a snapshot of a specific point in time (which could also be the most recent state). Outgoing page links are available by ad hoc parsing. For incoming links a PageTier Network needs to be extracted. Finally it is possible to import SQL link dumps from the official Wikimedia sources which represent the current state of linking.[6]

2.2. Core API

The WikiDragon Core Model API (see figure 3) provides classes and methods to browse through collections of MediaWikis, pages, revisions, namespaces and contributors. For example a developer can fetch a page by a specific namespace and title, ask for the latest revision, get the contributor and then ask for all revisions he or she has written. Also accessing HTML and outgoing and incoming links (depending of the kind of processing, c.f section 2.1.) is possible.

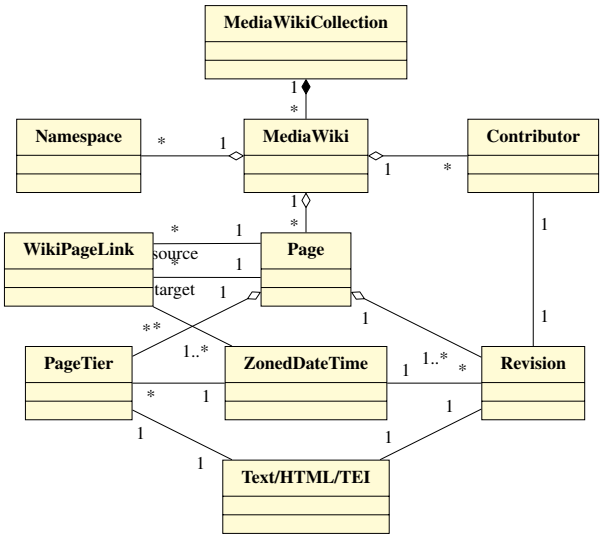


Figure 3: UML diagram depicting the (simplified) Core Model API.

WikiDragon provides two means to access past states of a MediaWiki instance. The first and most straight-forward solution is to fetch the revision of a page which has been active at a specific point in time. This gives access to HTML as well as outgoing links because they are parsed ad hoc. It is possible to perform a complete parsing of all revisions in HTML. Because of the extensive amount of time and space this takes on large MediaWiki instances it is also possible to restrict the parsing to specific pages or revisions by implementing a filter. The second method to access past states

[6]Import links from SQL is performed into whatever database system used- it does not depend on a relational DBMS.

is to create a *PageTier* representing a snapshot at a specific point in time. A snapshot is solely identified by its timestamp and contains all contents parsed in HTML. Optionally this HTML can be used to explicitly compute and store the page network at that time. The advantage is that with this performed it is possible to ask for all all incoming links of a given page at a specific time. In practice this second approach is the best choice when a sequence of snapshots should be examined since only those revisions are parsed and processed which are relevant.

3. Evaluation

In this section we evaluate how WikiDragon performs on typical use cases when working on Wikis. We start by importing the history dump of the simple English Wikipedia into a Neo4J database as backend using non-transactional mode[7]. For our experiments we use uncompressed versions of XML und SQL dumps which are available online[8]. The import is done using up to 8 threads for a combined Diff-BZip2 compression of revisions. This is the basis for all subsequent experiments. As a reference, we also perform an import without any compression. Then we create annual snapshots within the system to get parsed HTML of all pages (including all namespaces) and extract the page networks. Finally we create and evaluate the page network of the latest state of each page against an import of the official SQL dumps representing the current state of page- and category-links as well as redirects.

Task	Time	DB Size	$\|V\|$	$\|E\|$	$\frac{\|GCC\|}{\|V\|}$
Import ref	*00:29:01*	*32.35* GB	-	-	-
Import Diff	01:29:05	5.18 GB	-	-	-
2004-01-01	00:02:12	5.39 GB	1,025	3,250	0.870,244
2005-01-01	00:01:10	5.40 GB	3,938	25,248	0.886,237
2006-01-01	00:02:08	5.44 GB	12,158	109,030	0.898,339
2007-01-01	00:04:54	5.53 GB	32,821	301,002	0.925,932
2008-01-01	00:12:17	5.70 GB	61,773	619,343	0.940,039
2009-01-01	00:28:00	6.00 GB	106,729	1,099,710	0.947,371
2010-01-01	00:35:39	6.63 GB	152,914	2,784,841	0.953,771
2011-01-01	00:42:35	7.17 GB	193,434	3,463,370	0.950,913
2012-01-01	00:53:36	7.81 GB	225,884	4,169,790	0.951,347
2013-01-01	01:12:14	8.75 GB	268,830	5,157,335	0.961,254
2014-01-01	03:09:10	9.40 GB	309,732	3,727,148	0.961,431
2015-01-01	01:11:51	10.23 GB	342,115	6,669,247	0.961,530
2016-01-01	01:07:12	11.10 GB	382,936	7,119,001	0.930,064
2017-01-01	01:11:26	12.03 GB	410,131	7,545,065	0.931,100
2017-10-01	01:08:56	13.00 GB	433,189	7,895,909	0.932,076
Link-Dump	06:20:37	13.31 GB	433,214	8,022,464	0.938,425

Table 1: Statistics of the creation of PageTiers of the simple English Wikipedia representing annual snapshots of HTML content and page network.

Table 1 summarizes our evaluation on import and PageTier creation. The latter consists of two steps: parsing content in HTML and extracting a network based on the HTML code. Extracted information is stored in the database. If the network structure is not needed, this step could be skipped to save disc space and computing time. The combined Diff-Zip2 compression for revisions results in a database of 5.18GB, saving about 84% of disc space compared the uncompressed version while taking about 3 times longer for import. While the number of pages $|V|$ constantly grows, the number of $|E|$ edges drops intermediately when extracted on 2014-01-01. Nonetheless the ratio of the size of the greatest connected component $|GCC|$ to the number of vertices $|V|$ is generally stable, even around this time.
We use the network structure of the PageTier on midnight 2017-10-01 to evaluate the extraction against an import of the official SQL link dumps[9]. Table 2 shows that the results for page-links and redirects are above 98%. Recall and FScore for categorization is about 4% less, indicating that the extraction still misses an overproportional amount of category links. This will be adressed in future work.

Link-Type	Precision	Recall	FScore
Page-Link	0.990,380	0.987,971	0.985,423
Categorization	0.988,782	0.944,177	0.944,971
Redirect	0.999,885	0.999,901	0.999,785

Table 2: Evaluation of link extraction based on HTML against SQL link dumps of the simple English Wikipedia.

These experiments were conducted using Neo4J as backend. However WikiDragon is by design not limited to a specific backend but can be adapted to other database systems by implementing specific interfaces. Since Neo4J does not provide horizontal scaling for write operations, incorporating other database systems like Cassandra[10] or BlazeGraph[11] are likely to significantly improve performance for import- especially when processing large Wikipedias like German or English. Neo4J has been chosen because it provides sufficient performance in many use cases and does not require any additional standalone installation.

3.1. Comparing WikiDragon and JWPL

Whenever conclusions are drawn on any given resource we need to ensure that the representation of the resource is as accurate as possible. In this subsection we examine the impact of the different approaches of WikiDragon (XOWA) and JWPL (Sweble) to HTML rendering on exemplary use cases of language studies. We assume that different technologies lead to different results which should be reflected in terms of distribution characteristics such as the type-token ratio. We use the actual Web API of the simple English Wikipedia to build a reference as gold standard. We chose this instance because it allows us to perform a complete analysis over all articles. A complete download based on the English Wikipedia based on the Web API would not have been feasable.
For WikiDragon and JWPL we use the meta-current dumps of the simple English Wikipedia. The import for JWPL was created using DataMachine with standard parameters. Since JWPL only considers pages in the namespaces of *articles*, *articles talk* and *categories*, we focus our analysis on articles. Furthermore JWPL appears to automatically resolve redirects by default so that only non-redirect articles are considered. For our experiment we use the intersection

[7] Transactional mode can be activated in Neo4J lateron without the need to rebuild the database. All experiments published in this work are conducted without transactions.

[8] https://dumps.wikimedia.org/simplewiki/20171001/

[9] Note that there is a delta of 25 pages which have been created on 2017-10-01 after midnight.

[10] http://cassandra.apache.org/

[11] https://www.blazegraph.com/

extracted by WikiDragon, JWPL and Web API of the simple English Wikipedia (extracted on 2017-09-29, using the latest revision IDs of the dump used for WikiDragon and JWPL) resulting in 127,634.00 articles. We extract plain text from each article by the default methods provided by the frameworks. As normalization towards JWPL we strip the title header as well as the categories section from the WikiDragon output. From the reference Web API output we strip the section index which is automatically created for larger pages as well as the "edit"-labels which are generated for section titles. For tokenization we use *PTBTokenizer* of *Stanford CoreNLP*(Manning et al., 2014).

Figure 4 shows a comparison of the type-token ratio distributions based on WikiDragon/XOWA (blue) and JWPL/Sweble (red) against the Web API as reference. The distributions are sorted in ascending order based on the Web API variant. As the plots indicate, both distributions deviate from the gold standard to some degree. In case of Sweble the distribution is scattered much more, indicating a significant difference to the original data. Accordingly, the correlation coefficients[12] between XOWA and Web API are considerably higher (0.955, T=15,679, $P < 2.2e^{-16}$) than between Sweble and Web API (0.663, T=3,462, $P < 2.2e^{-16}$).

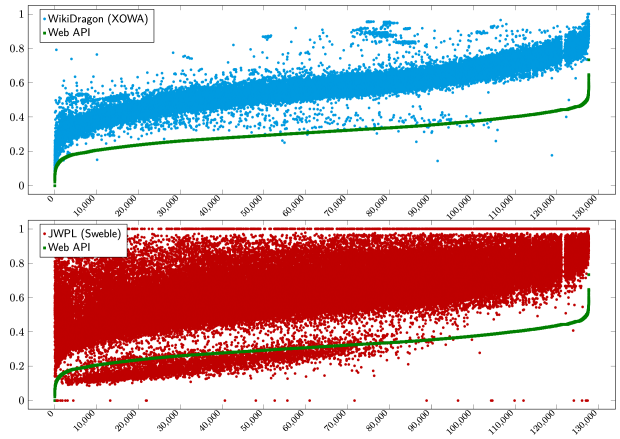


Figure 4: Comparison of the type-token distributions based on WikiDragon/XOWA (blue) and JWPL/Sweble (red) against the Web API as reference.

4. Conclusion

We introduced WikiDragon as a Java framework to build, process and analyse Wikis offline. It is unique in terms of combining the aim of complete coverage of wikis with an efficient API access for developers. We showed that the system enables researchers to conduct diachronic studies on content and network structure of wikis. An evaluation against the official Wikimedia dumps showed high accuracies for link extraction based on HTML. To our knowledge the diachronic network analysis on the simple English Wikipedia is the first ever performed. We examined the impact of using different HTML rendering engines for MediaWiki markup. Results show that XOWA performs more accurate than Sweble. Future work will address the integration of additional DBMS, the development of API extensions for Wikimedia projects such as Wiktionary and WikiNews and increasing accuracy of link detection. WikiDragon is available on GitHub[13].

[12]The distance correlations haven been computed and averaged over random samples based on 10 cycles using 10,000 items each.

5. Bibliographical References

Dandala, B., Mihalcea, R., and Bunescu, R. (2013). Word sense disambiguation using wikipedia. In Gurevych I. et al., editors, *The People's Web Meets NLP. Theory and Applications of Natural Language Processing.*, pages 241–262. Springer, Berlin, Heidelberg.

Daxenberger, J. and Gurevych, I. (2012). A corpus-based study of edit categories in featured and non-featured wikipedia articles. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 711–726, December.

Dohrn, H. and Riehle, D. (2011). Design and implementation of the sweble wikitext parser: Unlocking the structured data of wikipedia. In *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, WikiSym '11, pages 72–81, New York, NY, USA. ACM.

Ferschke, O., Zesch, T., and Gurevych, I. (2011). Wikipedia revision toolkit: Efficiently accessing wikipedia's edit history. In *Proceedings of the ACL-HLT 2011 System Demonstrations*, pages 97–102, Portland, Oregon, June. Association for Computational Linguistics.

Flöck, F. and Acosta, M. (2014). Wikiwho: Precise and efficient attribution of authorship of revisioned content. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 843–854, New York, NY, USA. ACM.

Gabrilovich, E. and Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *In Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1606–1611.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., and Bizer, C. (2015). DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6(2):167–195.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Mehler, A., Gleim, R., vor der Brück, T., Hemati, W., Uslu, T., and Eger, S. (2016). Wikidition: Automatic lexiconization and linkification of text corpora. *Information Technology*, pages 70–79.

Mehler, A., Gleim, R., Lücking, A., Uslu, T., and Stegbauer, C. (2018). On the self-similarity of Wikipedia talks: a combined discourse-analytical and quantitative approach. *Glottometrics*, 40:1–44.

Nothman, J., Curran, J. R., and Murphy, T. (2008). Transforming wikipedia into named entity training data. In *In*

[13]https://github.com/texttechnologylab/WikiDragon

Proceedings of the Australasian Language Technology Association Workshop 2008, pages 124–132.

Priedhorsky, R., Chen, J., Lam, S. T. K., Panciera, K., Terveen, L., and Riedl, J. (2007). Creating, destroying, and restoring value in wikipedia. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work*, GROUP '07, pages 259–268, New York, NY, USA. ACM.

Uslu, T., Mehler, A., Baumartz, D., Henlein, A., and Hemati, W. (2018). fastsense: An efficient word sense disambiguation classifier. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Yasseri, T., Sumi, R., Rung, A., Kornai, A., and Kertész, J. (2012). Dynamics of conflicts in wikipedia. 7:e38869, 06.

Zesch, T., Müller, C., and Gurevych, I. (2008). Extracting lexical semantic knowledge from wikipedia and wiktionary. In *in Proceedings of LREC*.

Studying Muslim Stereotyping through Microportrait Extraction

Antske Fokkens♣, Nel Ruigrok♡, Camiel Beukeboom♢,
Sarah Gagestein♠ and Wouter van Atteveldt♢
♣ Computational Lexicology and Terminology Lab. Vrije Universiteit Amsterdam, the Netherlands.
♠ LJS Nieuwsmonitor. Amsterdam, the Netherlands
♢ Department of Communication Science. Vrije Universiteit Amsterdam, the Netherlands
♡ Taalstrategie. Amsterdam, the Netherlands.
{antske.fokkens,c.j.beukeboom,w.van.atteveldt}@vu.nl,nelruigrok@nieuwsmonitor.org,sarah@taalstrategie.nl

Abstract

Research from communication science has shown that stereotypical ideas are often reflected in language use. Media coverage of different groups in society influences the perception people have about these groups and even increases distrust and polarization among different groups. Investigating the forms of (especially subtle) stereotyping can raise awareness to journalists and help prevent reinforcing oppositions between groups in society. Conducting large-scale, deep investigations to determine whether we are faced with stereotyping is time-consuming and costly. We propose to tackle this challenges through the means of *microportraits*: an impression of a target group or individual conveyed in a single text. We introduce the first system implementation for Dutch and show that microportraits allow social scientists to explore various dimensions of stereotyping. We explore the possibilities provided by microportraits by investigating stereotyping of Muslims in the Dutch media. Our (preliminary) results show that microportraits provide more detailed insights into stereotyping compared to more basic models such as word clouds.

Keywords: stereotyping, digital social science, text analysis

1. Introduction

Research in social sciences and computational linguistics shows that language use plays a crucial role in creating stereotypes about social categories (like Muslims). Several fields within Social Sciences have studied how stereotypic beliefs are reflected in language and consequently become shared knowledge (Ruscher et al., 2005; Klein et al., 2008; Beukeboom, 2014, among others). These studies show that people (unconsciously) express stereotypic beliefs they hold about described individuals or groups in both linguistic content (what information is provided) and form. Computational linguistic studies show that language models learned from corpora reflect such human bias, including stereotypes (Caliskan et al., 2017, e.g.).

Insights from computational linguistics and communication science are complementary. Distributional semantic models applying purely mathematical models to large corpora of text reveal that biases are present in the texts. Yet, these abstract language models do not provide means to reveal how bias is expressed in language exactly. Social science studies fill this gap, for example by studying the *frames* that media use in their coverage (Entman, 1993). These studies, however, only cover limited sets of data as they often rely on experimentally manipulated sentences or manual annotation to study biased language use. Moreover, the need for double-blind annotations on each new set under investigation makes it challenging to study the expression of stereotypes over time and across a high variety of sources. Automatic approaches that can help tackle this problem can therefore greatly enhance the possibility of investigating stereotyping in natural language. Previous work has shown that NLP can be used to identify various forms of framing (Card et al., 2016). These general classifiers perform deeper insight than pure distributional models, but still do not provide the means to look at the more subtle aspects of stereotyping.

This paper introduces a method that allows for stereotype detection in natural language by means of **microportraits**. A microportrait is the collection of descriptions that a single text provides on a given entity or event. We can investigate stereotyping by targeting information about individuals or groups. A microportrait of a person, for instance, combines the labels used to refer to them, the characteristics assigned to them and the roles they played in various events. We provide the first implementation of microportrait extraction for Dutch and use this to investigate stereotyping of Muslims in the Dutch media.

The contributions of this paper are the following:

1. We introduce the notion of microportraits and outline how they can be used to investigate expressions of stereotypes in texts.

2. We provide the first implementation for 'vanilla' microportrait extraction for Dutch. This implementation is open source.

3. We analyze a corpus of microportraits around the Dutch election time and share the first observations from this study

Even though results on the quality of the microportraits themselves are preliminary, the outcome of our study indicates that microportraits can lead to deeper insight into stereotypical descriptions compared to more basic methods such as LDA. We therefore conclude that microportrait extraction is a valuable new task in NLP with large potential in interdisciplinary research.

This paper is structured as follows. Section 2 provides background information on stereotyping in general, recent advances in stereotyping of Muslims and related work in NLP. We introduce microportraits and outline the proposed

method for studying stereotyping in Section 3. In Section 4, we describe the pilot study. The initial evaluation and first steps towards a proper evaluation set are described in Section 5. Sections 6 and 7 provide our discussion and conclusion.

2. Background and Related Work

In this section, we outline related work. We first provide an overview of the ways in which stereotypes can be expressed in language in Subsection 21. We then address related work in NLP in Subsection 22. This section ends with an overview of previous findings concerning stereotyping of Muslims in the media.

2.1. Linguistic cues for stereotyping

There is a wide range of literature on stereotyping and biased language use, hence a full overview is beyond the scope of this paper. Instead, we provide an overview of the principle observations and dimensions as outlined by Beukeboom and Burgers (2017).

Following Beukeboom and Burgers (2017)'s Social Categories and Stereotypes Communication (SCSC) framework, we distinguish aspects of stereotyping along two dimensions: the first dimension looks at **labels** that are used to refer to groups or individuals as well as **characteristics** and **behaviors** of these groups and individuals. Second, for both of these aspects it holds that stereotyping can be expressed through the **content** and through the **form** of an expression or, in other words, through the **frames** used.

2.1.1. Bias in labels

The bias in label content is reflected in terminological choices which can be neutral terms (e.g., 'refugees') or terms with a negative connotation (e.g., 'fortune-seekers', 'aliens'). Bias in label form is observed in at least two ways. First, noun labels, compared to adjectival descriptions, are more strongly associated with stereotypic inferences. For instance, when a person is described as 'a Jew', this is more defining and recipients tend to more strongly expect stereotypical Jewish habits. Adjectival references in contrast (i.e., 'Jewish'), are seen as just one characteristic that is less profound and immutable (Carnaghi et al., 2008). Second, labels can be expressed in more generic or more specific form. Here we distinguish generic labels (*Muslims are..*) from subsets (*some Muslims are...*), subtypes (*fundamental Muslims are..*) and individuals (*my Muslim neighbor is..*). Statements that make use of generic labels play an important role in transferring category knowledge (Cimpian and Markman, 2008; Gelman, 1988) and can therefore be particularly influential in spreading stereotypes.

2.1.2. Bias in describing characteristics and behavior

In describing characteristics and behavior, bias is mainly observed in the selection of information. Characteristics and behaviors are more likely mentioned when they fit the existing stereotype (Klein et al., 2008, among others). This tendency has a reinforcing effect. When people are more frequently exposed to stereotype-congruent information, this leads to a continuous confirmation of existing (negative) stereotypic associations, as has for instance been shown for negative news concerning immigrants (Schemer, 2012). At the same time, there can be opposing movements where counter-stereotypical information is provided, for instance because it is socially unacceptable to express prejudiced stereotypic beliefs (Ruscher et al., 2005). However, this does not necessarily disconfirm the stereotype. It depends on the form on which this information is delivered.

Research on linguistic form shows that stereotype incongruent information is often formulated differently than information that is congruent with an existing stereotype (Beukeboom, 2014). Research on the Linguistic Intergroup Bias (Maass et al., 1995; Wigboldus et al., 2000) shows that stereotype-congruent behavior is more likely to be described in terms of enduring personality traits (e.g., *the woman is emotional*), but more behavioral and concrete when it is stereotype-incongruent (e.g., *the man is crying*). Similarly, the Stereotypic Explanatory Bias shows that stereotype-incongruent behaviors are more often explained (Hammer and Ruscher, 1997). And the Negation Bias shows that stereotype-incongruent information is more likely described with a negation that simultaneously introduces a stereotype congruent term (e.g., using 'not oppressed' to describe female Muslims rather than 'independent' or 'free') (Beukeboom et al., 2010). With these biases stereotype incongruent behaviors are framed as unexpected one-time events.

We are not aware of any NLP approach that aimed to specifically study the above described biases, yet there are previous approaches that addressed identifying stereotyping. The following subsection will outline this related work in NLP and explain how microportrait extraction provides a next step.

2.2. Stereotyping in NLP

Using NLP to identify issues of stereotyping is still a relatively unexplored field. Nevertheless, recent studies have addressed various aspects of this topic. We distinguish three related themes: 1) studies that investigate how biases and stereotypes are reflected in language models, among others searching for approaches to avoid that such connotations are reflected by algorithms applying these language models, 2) studies that aim to identify offensive or stereotypical language use and 3) studies that develop software to study how stereotyping is reflected in language.

Most studies fall in the first category. Howard and Borenstein (2017) provide an overview of ways in which our biases are reflected in machine learning algorithms. Notably, Caliskan et al. (2017) show that human biases are reflected in distributional models for both 'morally neutral' concepts such as flowers and insects as well as concepts where this is problematic such as gender and race (Caliskan et al., 2017). The same insights with regards to gender bias had previously been made by Bolukbasi et al. (2016), who also propose a method for 'fixing' this bias.

Binns et al. (2017)'s title refers to the observation of bots taking over bad language. They build a classifier aiming to detect when language is offensive. A similar approach is found in the research of Tulkens et al. (2016) which aims to detect racist discourse. Both studies reveal the challenge

involved in creating a gold standard for such studies due to the subjective nature of determining when a statement is crossing the line.

Related work that studies bias in linguistic expressions directly, and is therefore closest related to this research, includes van Miltenburg (2016) and van Miltenburg et al. (2016). Their studies show that image descriptions include cultural biases (e.g., people will describe a child as "black" or "Asian", but white children are simply "a child") and a tendency to use negation to indicate that something is different from what is expected (e.g., a man without shirt). Joseph et al. (2017) investigate to what extent we can observe stereotyping in Twitter, distinguishing between *affective stereotyping* and *semantic stereotyping*, where the former refers to the feeling we associate with a label and the later refers to other associations (such as the activities associated with a specific profession).

The main difference between the studies mentioned above and the approach proposed in this paper is that microportraits aim to reflect the story that is being told of a single entity or group with a single article. By extracting these stories on a large scale, we can then look for patterns and see e.g., how a specific group is portrayed in the media. From this point of view, the study that is probably closest related to this work is Card et al. (2016) who study frame detection through persona description, showing that extracting small stories around persona provides useful information in detecting frames annotated in the Media Frames Corpus (Card, Dallas and Boydstun, Amber E and Gross, Justin H and Resnik, Philip and Smith, Noah A, 2015), (Card et al., 2015). Their approach combines syntactic relations and labels applying a Dirichlet process and using the outcome in a Bayes model for identifying frames. Our study mainly differs from this approach in that it offers the patterns of co-occurring descriptions directly to social scientists rather than offering potentially identified frames. As such, our approach stimulates bottom-up investigation of expressed stereotypes and can be seen as complementary to the work by Card et al. (2016). The outcome provides the possibility of going beyond simple choices between labels and properties, but also allows researchers to investigate how labels, properties and roles relate to each other. We will explain the idea behind the model in more detail in Section 3.1.

2.3. Muslim Stereotyping in the media

The process of 'media logic' influences the way in which the public debate as found in the media is held. With more and more people turning to commercial news outlets and social media for their information (Sonck and de Haan, 2015), the debate is increasingly dominated by entertainment and simplification, focusing on conflict and persons (Welbers et al., 2015). Such news coverage leads to a higher level of (political) cynicism among the general public (Cappella and Jamieson, 1996; Wolfsfeld, 2011). Populist parties use these feelings of unease among the public with one-liners that fit well into the media logic of the media. As a consequence, negative stereotypes and prejudice can take hold of the public which may very well harm relations between communities and provoke societal conflict (Rehman, 2007).

In the United States, discrimination toward Arab Muslims increased after September 11th, 2001 (Sheridan, 2006). Stereotypes, prejudice, and discrimination toward Arab Muslims increased even further in the wake of international terrorism by extremist groups who claimed to have ties to Islam (e.g., ISIS). This is also true for the Netherlands (Adriaansen et al., 2010; Ruigrok et al., 2017b).

Two common components of the Arab Muslim male stereotype are (i) that Arab Muslims are part of the out-group (Saeed, 2007), and (ii) that they are angry, violent, and often terrorists, personified by images of Osama bin Laden (Jackson, 2010). Muslims are presented as terrorists 81% of the time on television (Dixon and Williams, 2015). Many individuals report being afraid of Muslim and/or Arab men, often because they are perceived as violent and a threat to America (Gottschalk and Greenberg, 2008; Sides and Gross, 2013). Muslim women, on the other hand, are stereotyped as oppressed (Stadlbauer, 2012). Specifically, many inhabitants of non-Muslim countries believe that the veil of Muslim women (e.g., hijab, niqab, and burqa) is a representation of oppression (Wagner et al., 2012).

3. Introducing Microportrait Extraction

The goal of this section is to introduce microportrait extraction as an NLP task. After defining what microportaits are, we explain how they can be used to study stereotyping in Section 31). We then describe the first implementation of microprotrait extraction (Section 32).

3.1. Microportraits and Stereotypes

Microportraits are designed to study framing and stereotyping. The idea is that by exploring how specific people are described on a large scale, researchers can identify common patterns in descriptions of people who share certain characteristics. For the use case in this paper, for instance, we explored how people that are explicitly labeled as "Dutch" or "Muslim" are described in Dutch media. The basic units of a microportraits are **descriptions**. A description can be a **label** assigned to an entity, a **property** assigned to them or a **role** they play in a specific event. For instance, the expression *the pious Muslim smiled* contains the following three descriptions: the label *Muslim*, the property *pious* and the agent or arg0 role in *smiling*. The **microportrait** of a person is the collection of all descriptions of this person within a single article. Table 1 illustrates the microportraits for the referents of *Muslim* and *John* in the snippet *the pious Muslim smiled when John waved at him.* All descriptions related to the same referent share the same identifier (*docIdt2* for the Muslim and *docId3* for John in Table 1).

When applied to large volumes of text, researchers can use microportraits to identify which labels, properties and activities tend to co-occur and what choices writers make when describing a person. For instance, do they choose to indicate that someone belongs to a religious group by using a property (*a Muslim man*) or a label (*a Muslim*)? What other properties and labels are used when talking to individuals from this group? Do certain sources talk in terms of "us" and "them"? When do writers feel inclined to make a specific origin, religious background, hobby or some other feature (such as looks or achievements) explicit?

identifier	relation	label
docIdt2	label	*Muslim*
docIdt2	property	*pious*
docIdt2	arg0	*smile*
docIdt2	arg2	*wave*
docIdt2	label	*him*
docIdt3	label	*John*
docIdt3	arg0	*wave*

Table 1: Illustration of microportrait

More subtle forms of stereotyping can be identified by investigating how descriptions of groups differ when talking about a specific theme. Do people from certain backgrounds easily receive labels such as *thief*, *criminal* and *perpetrator* (asserting involvement and making it part of their being) whereas others are *suspect* of, e.g. *stealing* (leaving the option of innocence open and highlighting (possibly incidental) behavior)? This can be investigated by searching for descriptions related to a specific topic and investigating how various groups are described in relation to them. A mixed approach would start by a bottom-up approach to identify typical labels, properties and roles and then using the outcome in a top-down study for examining details in choices of representation.

Microportraits provide profiles of people, groups and other entities within a document. Without prior knowledge of the stereotypical traits associated with a group, a microportrait in isolation cannot provide insight in stereotyping. It can also not be used to identify the emergence of new biases that may influence the stereotypical views in society. By investigating patterns in microportraits extracted from a large corpus, we can however identify biases in the kind of information provided when specific groups are described (stereotypical content) and whether there are differences in how this group is presented (stereotype reflection in form). Microportraits can be used to investigate several dimensions of stereotypes depending of whether one takes a bottom-up approach, a top-down approach or a mixture of the two.

The bottom-up approach is the most straight-forward way of studying stereotyping through microportraits. It can be applied by simply calculating the Pointwise-Mutual Information score of descriptions that co-occur in the same microportrait. We can use the outcome of these calculations to see, for instance, which descriptions are typically used for Muslims and how they compare to descriptions typically used for other groups.

3.2. Microportrait extraction

We have implemented a first version for extracting microportraits from Dutch news and applied this to study identify how Dutch media talk about Muslims and immigrants from Muslim countries compared to how they talk about Dutch people. This subsection describes the current implementation of our system for Dutch microportrait extraction, which combines syntactic patterns with entity coreference resolution. Because it does not provide the deeper semantic interpretation provided by semantic roles yet, we refer to the microportraits coming out of the current implementation as *vanilla microportraits*.

We use a small pipeline that forms a subpart of the pipeline for event extraction developed as part of the larger NewsReader (Vossen et al., 2016) and BiographyNet (Fokkens et al., 2017) pipelines for event extraction. The pipline includes the Alpino parser for dependency parsing (Bouma et al., 2001) and the ixa-pipe named entity recognizer (Agerri and Rigau, 2016). Both modules are run using a wrapper that provides output in the NLP Annotation Format (Fokkens et al., 2014a, NAF).[1] We implemented a new system for Dutch coreference resolution, based on the Stanford multisieve-entity coreference resolution approach (Lee et al., 2013) that applies coreference resolution on top of this small pipeline.[2]

In the first step, we extract descriptions at a sentence level. We start by taking nouns as labels and then identify their properties by extracting their modifiers and attributes through copula constructions using the dependency structure. We also use syntactic dependencies to identify the 'roles' an entity plays. Alpino outputs 'deep' dependencies that indicate that the subject of a passive sentence has an object relation with the main verb. We take the simplified assumption that (deep) subjects are agents, (deep) objects are patients and (deep) indirect objects recipients. Prepositional complements receive the label *prep-role*, where *prep* corresponds to the surface form of the preposition. In the ideal form, the roles in microportraits would consist of semantic roles, but the only open source semantic role labeler for Dutch we are aware of yielded worse results than our current patterns on a development set.

The outcome of this step yields so-called 'nano-portraits', which are descriptions related to an entity within a clause. In the second step, we combine the nano-portraits related to the same entity using the output of the coreference resolution module. This results in collections of descriptions related to the same referent within a document: the so-called 'vanilla microportrait'. The next section will explain how we used automatically extracted vanilla microportraits in order to investigate whether Muslims are stereotyped in a discriminatory manner in Dutch media.

4. Muslims in Dutch News

In this section, we describe the outcome of a pilot study using microportrait extraction for identifying stereotyping of Muslims. The outcome of this study was used to shed light on how Dutch media talked about Muslims in political news during election time. From a computational linguistics point of view, this study serves the purpose of exploring what microportraits have to offer for this type of investigation. In particular, we explore whether they allow communication scientists to go beyond methods that are commonly used, such as word clouds based on Tf-idf scores.

[1] This Alpino wrapper is available at: `https://github.com/cltl/morphosyntactic_parser_nl` and the IXA-named-entity recognizer at `https://github.com/ixa-ehu/ixa-pipe-nerc`.

[2] The implementation is available under the Apache license under: `https://github.com/antske/coref_draft`

	Dutch	Muslims
labels and properties	famous, average, Dutch origin, fast, beautiful, free	radical, moderate, conservative, Sunni extremist, pious
roles (agent)	take, miss, win, break, drive, make, score	insult, convert, adhere, rape, murder, extinct

Table 2: English translation of most typical descriptions

	Dutch	Muslim
label	0.0000	-0.020
property	0.050	-0.073
roles	0.008	-0.140

Table 3: Positive/Negative reporting

4.1. Method

We collected news articles from national daily newspapers, online newsites and news blogs from the period from September 5th up until March 15th 2017. This resulted in a total of 622,480 articles. The use case focuses on articles that address politics in election time. We therefore extracted articles about politics that appeared in the period from January first to March 15th 2017, covering the election day for the Dutch parliament and 2.5 months leading up to it, from this set. We consider an article to be about politics when it explicitly mentions one of the political parties or one of its prominent members or a member of the parliament or government. This resulted in a selection of 15,573 articles.

In this set of political articles, we compare which words, labels, properties or roles occur typically with people labeled as *Dutch* and which are typical for people labeled as *Muslim*. We investigate both the full set of news articles as well as the subset mentioning politicians or political parties.[3]

We use two approaches: first, we use Latent Dirichlet Analysis (LDA) in order to identify which words typically occur in articles referring to Dutch people, Muslims or both. In the next step, we extracted microportraits from these texts in order to identify what was said about Muslims and Dutch people, respectively, on a more fine-grained level. We investigated the most typical labels and properties as well as the most typical roles.

4.2. Analysis

The outcome of the exploratory study using LDA reveals that articles that only refer to Dutch people are mainly about (winning) sports, the most typical words in news articles about Muslims are *aanslag* "assault", *president* (idem) and *amerikaanse* "American", due to the discussion about Trump's proposal to refuse Muslims from certain countries entrance to the United States. Articles that mentioned both "Muslims" and "Dutch" seem mainly to be directly related to the elections, society and integration (with typical words such as *stemmen* "vote", *integratie* "integration" and *democratie* "democracy").

The results obtained through microportrait extration are presented in Table 2. The table provides English translations of the labels, properties and roles most typically mentioned when Dutch articles explicitly talk about Dutch people or Muslims.

The terms in Table 2 indicate a shocking difference in the way media portray Muslims. However, this list is still anecdotal. In order to gain further insight into whether Dutch people are indeed generally portrayed positively whereas Muslims are portrayed negatively, we let four student assistants annotate descriptions indicating whether they provided a positive or negative picture of the person being described. The most frequent descriptions from our corpus were presented out of context (so that annotators did not know whether they applied to Muslims, Dutch people or neither) and annotated independently by at least two annotators each. We then assigned negative scores (-1) to descriptions annotated as negative and positive scores (+1) to descriptions labeled as positive in our microportraits. The results are presented in Table 3. For all three categories, results indicate that people labeled as 'Muslim' are described more negatively than people called 'Dutch'. Where the former are on average portrayed (slightly) negatively, descriptions associated with Dutch people are neutral to positive. Results are significant for all three categories.

5. Initial Validation and Annotations

In this section we present our validation steps and the first steps towards creating a gold standard for evaluation.

5.1. Validation

We performed an initial validation of the method by checking the precision 1,058 descriptions from randomly selected articles. Manual inspection revealed that 98.1% of the descriptions were correctly extracted from the text. Furthermore, 87.2% of the descriptions was placed into the correct microportrait. These results seems suspiciously high, but this is mainly due to the the far majority of descriptions being expressed by basic substructures in a sentence that the Alpino parser analyzes correctly despite possibly making errors in more complex parts of the sentence. The high result of the validation of the placement in the correct microportrait is high, because assignments are partially due to local structures (such as the adjective modifying a noun or a predicative structure).

Most mistakes we found in detecting descriptions occurred in long sentences. There is no reason to assume that journalists use longer sentences when talking about Dutch people or Muslims. We can therefore expect errors to be equally distributed over the two classes and there are currently no indications of our approach leading to a bias in the overall outcome.

The validation looks promising, but a more detailed and solid evaluation of the method is necessary. We are currently in the process of creating gold standard data. The current status of this gold standard is described in the coming subsection.

5.2. Annotation Instructions

In order to provide a better evaluation, a gold standard evaluation set is currently under development. As is common

[3]This research is also reported (in Dutch) in Ruigrok et al. (2017a). More details can be found at `https://www.microportretten.nl`.

for new complex annotation tasks, annotations are carried out in multiple rounds leading to updates in the annotation guidelines. Annotators used the following procedure:

- mark all labels used to refer to an entity
- mark all modifiers of each label as 'property'
- connect each property to the appropriate label
- mark all activities and connect them to the appropriate label, indicating whether the label plays the role of 'agent','patient' or some other role
- connect all labels referring to the same entity to the same external identifier

During the first annotation cycle, annotators were instructed to annotate all microportraits occurring in the text. This resulted in low inter agreement scores. The main reason for these low scores were that annotators found it difficult to annotate the full text. Even after an additional revision round, annotations were incomplete and which annotations were missing differed from one annotator to another. We therefore launched a second cycle in which annotators only annotated the full microportrait if it contained a label or property from a predefined set (including the Dutch words for Muslim, Christian, Jewish, Belgian, German, Moroccan, Turkish, Dutch and derived terms). The annotations following these new guidelines are currently ongoing.

6. Discussion

The outcome of our use case reveals that microportraits can provide a different perspective than more basic approaches such as LDA. In particular, our method detected severe forms of stereotyping that remained unnoticed when looking a word co-occurrence alone. It must be taken into consideration, however, that the final evaluation of the technology is ongoing. Solid evaluation in interdisciplinary studies involves intrinsic evaluation (performance of the coreference resolution tool and microportrait extraction) and extrinsic evaluation (do the mistakes made by the tools introduce a bias that influences the research questions) (Fokkens et al., 2014b). At this point, more elaborate evaluation of the accuracy of our tools is necessary.

The high precision scores in our validation are encouraging. As mentioned in Section 51, we did not find any indication of a bias or errors that would systematically miss (positive) descriptions of Muslims or (negative) descriptions of Dutchmen. It is therefore unlikely that the outcome of our pilot study is the result of a bias in the tool. An additional indication that the results are likely to be accurate is that the topics covered in the news during the investigated period provide a plausible explanation. The observation that the label 'Dutch' is mainly used when stressing sport achievements is in line with the outcome of the LDA investigation. The corpus contained several articles addressing the events of New Year's Eve in Cologne as well as crimes committed by ISIS which are both topics likely to contain the negative stereotypical descriptions we identified for Muslims.

Nevertheless, the indications outlined above cannot replace a proper evaluation: when results are checked rather than evaluated on an independently created datasets, borderline cases will often be considered 'correct' leading to higher results. Moreover, these verifications do not give insight into the recall of the system. The correspondence between microportraits and covered topics supports our outcome, but does not provide a guarantee. The completion of a gold standard dataset is therefore essential for continuing this line of research. This is ongoing work and we plan to report the outcome of this evaluation in future work.

7. Conclusion

This paper introduced microportraits: the collection of all descriptions of a single entity (a person, group, object or event) found in a single document. We summarized insights from communication science on how stereotypes can be reflected in language use and showed how microportraits can be used to study stereotyping in text.

We implemented a basic pipeline for microportrait extraction for Dutch.[4] The pipeline extracts so-called 'vanilla microportraits' which consist of descriptions based on syntactic patterns. All tools implemented for this research are freely available under the Apache license.

We applied this to investigate stereotyping of Muslims in the Dutch media comparing how people or groups labeled explicitly as 'Dutch' or 'Muslim' are described. Whereas a basic bottom-up study using LDA mainly indicates themes that are discussed (sports for Dutch, terrorism and political crisis for Muslims), the microportraits provide a more detailed and specific insight into how groups are portrayed. Evaluation of the tool and methodology are currently ongoing. We carried out validation checks that indicate solid performance and the patterns that emerge from the microportraits can be explained by observations from the data. Though this is promising, validations checks are merely indicative and solid evaluation is needed. The creation of a gold-standard evaluation set is ongoing. Nevertheless, we believe that the preliminary outcome of our use case clearly illustrates the potential use of microportraits. As such, the main contribution of this paper is the introduction of this new NLP task that is of high interest for researchers in the social sciences wanting to investigate stereotyping.

8. Acknowledgements

The work presented in this paper was funded by the Netherlands Organization for Scientific Research (NWO) via VENI grant 275-89-029 awarded to Antske Fokkens. The pilot study was funded by the Stichting Democratie and Media (Democracy & Media Foundation).

9. Bibliographical References

Adriaansen, M., van Praag, P., et al. (2010). Nieuwe scheidslijnen en de turbulente relatie tussen politiek, media en burgers.

Agerri, R. and Rigau, G. (2016). Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

[4] https://github.com/cltl/micro-portraits

Beukeboom, C. and Burgers, C. (2017). How stereotypes become shared knowledge: Biased language use in communication about categorized individuals.

Beukeboom, C. J., Finkenauer, C., and Wigboldus, D. H. (2010). The negation bias: when negations signal stereotypic expectancies. *Journal of personality and social psychology*, 99(6):978.

Beukeboom, C. J. (2014). Mechanisms of linguistic bias: How words reflect and maintain stereotypic expectancies. *Social cognition and communication*, 31:313–330.

Binns, R., Veale, M., Van Kleek, M., and Shadbolt, N. (2017). Like trainer, like bot? inheritance of bias in algorithmic content moderation. In *International Conference on Social Informatics*, pages 405–415. Springer.

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., and Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Bouma, G., Van Noord, G., and Malouf, R. (2001). Alpino: Wide-coverage computational analysis of dutch. *Language and Computers*, 37(1):45–59.

Caliskan, A., Bryson, J. J., and Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Cappella, J. N. and Jamieson, K. H. (1996). News frames, political cynicism, and media cynicism. *The Annals of the American Academy of Political and Social Science*, 546(1):71–84.

Card, D., Boydstun, A. E., Gross, J. H., Resnik, P., and Smith, N. A. (2015). The media frames corpus: Annotations of frames across issues. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 438–444.

Card, D., Gross, J., Boydstun, A., and Smith, N. A. (2016). Analyzing framing through the casts of characters in the news. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1410–1420, Austin, Texas. Association for Computational Linguistics.

Carnaghi, A., Maass, A., Gresta, S., Bianchi, M., Cadinu, M., and Arcuri, L. (2008). Nomina sunt omina: on the inductive potential of nouns and adjectives in person perception. *Journal of personality and social psychology*, 94(5):839.

Cimpian, A. and Markman, E. M. (2008). Preschool children's use of cues to generic meaning. *Cognition*, 107(1):19–53.

Dixon, T. L. and Williams, C. L. (2015). The changing misrepresentation of race and crime on network and cable news. *Journal of Communication*, 65(1):24–39.

Entman, R. M. (1993). Framing: Toward clarification of a fractured paradigm. *Journal of communication*, 43(4):51–58.

Fokkens, A., Soroa, A., Beloki, Z., Ockeloen, N., Rigau, G., van Hage, W. R., and Vossen, P. (2014a). Naf and gaf: Linking linguistic annotations. In *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 9–16.

Fokkens, A., Ter Braake, S., Ockeloen, N., Vossen, P., Legêne, S., and Schreiber, G. (2014b). Biographynet: Methodological issues when nlp supports historical research. In *LREC*, pages 3728–3735.

Fokkens, A., ter Braake, S., Ockeloen, N., Vossen, P., Legêne, S., Schreiber, G., and de Boer, V. (2017). Biographynet: Extracting relations between people and events. In *Europa baut auf Biographien*.

Gelman, S. A. (1988). The development of induction within natural kind and artifact categories. *Cognitive psychology*, 20(1):65–95.

Gottschalk, P. and Greenberg, G. (2008). *Islamophobia: making Muslims the enemy*. Rowman & Littlefield.

Hammer, E. D. and Ruscher, J. B. (1997). Conversing dyads explain the unexpected: Narrative and situational explanations for unexpected outcomes. *British journal of social psychology*, 36(3):347–359.

Howard, A. and Borenstein, J. (2017). The ugly truth about ourselves and our robot creations: The problem of bias and social inequity. *Science and Engineering Ethics*, pages 1–16.

Jackson, L. (2010). Images of islam in us media and their educational implications. *Educational Studies: Journal of the American Educational Studies Association*, 46(1):3–24.

Joseph, K., Wei, W., and Carley, K. M. (2017). Girls rule, boys drool: Extracting semantic and affective stereotypes from twitter. In *CSCW*, pages 1362–1374.

Klein, O., Tindale, S., and Brauer, M. (2008). The consensualization of stereotypes in small groups. *Stereotype dynamics: Language-based approaches to the formation, maintenance, and transformation of stereotypes*, pages 263–292.

Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M., and Jurafsky, D. (2013). Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.

Maass, A., Milesi, A., Zabbini, S., and Stahlberg, D. (1995). Linguistic intergroup bias: differential expectancies or in-group protection? *Journal of Personality and Social Psychology*, 1(68):116–126.

Rehman, J. (2007). 9/11 and the war on terrorism: The clash of 'words', 'cultures' and 'civilisations': Myth or reality. In M.N. Craith, editor, *Language, Power, and Identity Politics*, pages 198–215, New York. Palgrave Macmillan.

Ruigrok, N., Fokkens, A., Gagenstein, S., and van Atteveldt, W. (2017a). Stereotyperende microportretten van moslims in het (politieke) nieuws. Technical report.

Ruigrok, N., van Atteveldt, W., Gagestein, S., and Jacobi, C. (2017b). Media and juvenile delinquency: A study into the relationship between journalists, politics, and public. *Journalism*, page 1464884916636143.

Ruscher, J. B., Cralley, E. L., and O'Farrell, K. J. (2005). How newly acquainted dyads develop shared stereotypic

impressions through conversation. *Group Processes & Intergroup Relations*, 8(3):259–270.

Saeed, A. (2007). Media, racism and islamophobia: The representation of islam and muslims in the media. *Sociology Compass*, 1(2):443–462.

Schemer, C. (2012). The influence of news media on stereotypic attitudes toward immigrants in a political campaign. *Journal of Communication*, 62(5):739–757.

Sheridan, L. P. (2006). Islamophobia pre–and post–september 11th, 2001. *Journal of interpersonal violence*, 21(3):317–336.

Sides, J. and Gross, K. (2013). Stereotypes of muslims and support for the war on terror. *The Journal of Politics*, 75(3):583–598.

Sonck, N. and de Haan, J. (2015). Media: Tijd in beeld.

Stadlbauer, S. (2012). A journey to a "pure islam": Time, space, and the resignification of ritual in post 9/11 faith testimonies of muslim women. *Narrative Inquiry*, 22(2):348–365.

Tulkens, S., Hilte, L., Lodewyckx, E., Verhoeven, B., and Daelemans, W. (2016). The automated detection of racist discourse in dutch social media. *Computational Linguistics in the Netherlands Journal*, 6(1):3–20.

van Miltenburg, E., Morante, R., and Elliott, D. (2016). Pragmatic factors in image description: The case of negations. In *Proceedings of the 5th Workshop on Vision and Language*, pages 54–59. ACL.

van Miltenburg, E. (2016). Stereotyping and bias in the flickr30k dataset. In Jens Edlund, et al., editors, *Proceedings of Multimodal Corpora: Computer vision and language processing (MMC 2016)*, pages 1–4.

Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A.-L., Aprosio, A. P., Rigau, G., et al. (2016). Newsreader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. *Knowledge-Based Systems*, 110:60–85.

Wagner, W., Sen, R., Permanadeli, R., and Howarth, C. S. (2012). The veil and muslim women's identity: Cultural pressures and resistance to stereotyping. *Culture & Psychology*, 18(4):521–541.

Welbers, K., van Atteveldt, W., Kleinnijenhuis, J., Ruigrok, N., and Schaper, J. (2015). News selection criteria in the digital age: Professional norms versus online audience metrics. *Journalism*, 17(8):1037–1053.

Wigboldus, D. H., Semin, G. R., and Spears, R. (2000). How do we communicate stereotypes? linguistic bases and inferential consequences. *Journal of Personality and Social Psychology*, 1(78):5–18.

Wolfsfeld, G. (2011). *Making sense of media and politics: Five principles in political communication*. Taylor and Francis.

10. Language Resource References

Card, Dallas and Boydstun, Amber E and Gross, Justin H and Resnik, Philip and Smith, Noah A. (2015). *The media frames corpus*. School of Computer Science, Carnegie Mellon University, 2.0.

Analyzing the Quality of Counseling Conversations: the Tell-Tale Signs of High-quality Counseling

Verónica Pérez-Rosas[1], Xuetong Sun[1], Christy Li[1], Yuchen Wang[1]
Kenneth Resnicow[2] and Rada Mihalcea[1]
[1]Computer Science and Engineering, [2] School of Public Health
University of Michigan
{vrncapr, xuetong, christyq, lucasyc,kresnic,mihalcea}@umich.edu

Abstract

Behavioral and mental health are pressing issues worldwide. Counseling is emerging as a core treatment for a variety of mental and behavioral health disorders. Seeking to improve the understanding of counseling practice, researchers have started to explore Natural Language Processing approaches to analyze the nature of counseling interactions by studying aspects such as mirroring, empathy, and reflective listening. A challenging aspect of this task is the lack of psychotherapy corpora. In this paper, we introduce a new dataset of high-quality and low-quality counseling conversations collected from public web sources. We present a detailed description of the dataset collection process, including preprocessing, transcription, and the annotation of two counseling micro-skills: reflective listening and questions. We show that the obtained dataset can be used to build text-based classifiers able to predict the overall quality of a counseling conversation and provide insights into the linguistic differences between low-quality and high-quality counseling.

Keywords: conversation analysis, counseling, mental health

1. Introduction

Mental and behavioral disorders, such as substance abuse, are top on the list of the most costly and prevalent conditions worldwide.[1] Particularly in the US, a recent survey on public health reported that in 2014 3.3% of all adults had co-occurring mental illness and substance abuse disorders.[2]

As behavioral counseling has been been shown to be an effective treatment method for these conditions, the number of people seeking counseling services is increasing (Chava, 2014). Despite its potential benefits, such as combating addiction and providing broader disease prevention and management, the mechanisms behind successful behavioral counseling have not been fully elucidated (Moyers et al., 2009).

Specific counseling skills have shown to increase the likelihood of positive health outcomes (Gaume et al., 2009; Vader et al., 2010). Regardless of the counseling method, counselors follow general principles, such as supporting autonomy, expressing empathy, centering on the patient and engaging patients using specific skills such as reflective listening (Charles et al., 1997; Harting et al., 2004). In contrast, using a more directing style – characterized by counselors providing instruction and advice, and patients obeying, adhering and complying (Miller and Rollnick, 2013) – is usually avoided.

The guidelines described above can be used to differentiate between low and high quality counseling. Thus, in a broad classification, psychotherapy conversations where counselors follow preferred practices can be considered as high-quality (or guideline adhering) counseling, whereas those conversations where they do not can be regarded as low-quality counseling (or guideline non-adhering).

Following this idea, our paper analyzes counseling conversations with the final goal of distinguishing between low and high quality counseling. In particular, we focus our analysis on counseling conducted using Motivational Interviewing (MI), a well-established evidence-based counseling style for treating addiction and other behaviors (Moyers et al., 2009; Catley et al., 2012; Apodaca et al., 2014).

Our work makes two main contributions. First, we introduce a new dataset of counseling conversations collected from public web sources. With this dataset, we seek to address the problem of lack of psychotherapy corpora for NLP applications, as most of current psychotherapy corpora have important constrains regarding their public accessibility due to ethical and privacy concerns. Second, we show that the collected dataset can be used to build text-based classifiers able to predict the overall quality of a counseling conversation and provide insights into the tell-tale signs of high-quality counseling.

2. Related Work

While clinical mental health counseling has been found useful in the treatment of public health issues, evaluating its quality remains a problem. This is mainly because most studies on clinical psychology have been limited by the need for human-based evaluation and by small sample sizes.

Computational approaches for the analysis of counseling interactions have focused on two main lines of work.

First, seeking to develop tools for the automatic evaluation of counseling practice, several linguistic based approaches have been proposed to aid the automatic identification of counselor and client behaviors that are correlated to successful interventions (Klonek et al., 2015). (Can et al., 2012) used n-grams, similarity features between coun-

[1]Word Health Report 2001, http://www.who.int/whr/2001/media_centre/en/

[2]The State of Mental Health in America https://www.samhsa.gov/disorders/substance-use

selor and client speech, and dialog meta-features to automatically detect and code counselors' reflective listening. A method based on labeled topic models is presented in (Atkins et al., 2012; Atkins et al., 2014), where authors focus on automatically identifying conversation topics that relate to counselor behaviors such as reflective listening, questions, support, and empathy. Methods that combine acoustic and linguistic datastreams have also been proposed to evaluate the quality of counseling interactions. (Xiao et al., 2014) presented a study on the automatic evaluation of counselor empathy based on analyzing correlations between prosody patterns and empathy showed by the therapist during counseling interactions.

Second, aiming to improve the understanding of counseling interactions, researchers have started to explore NLP approaches to study aspects such as language mirroring, empathy, and reflective listening. (Tanana et al., 2015) addressed the identification of counselor's statements that discuss client's change talk using recursive neural networks to model sequences of counselor and client verbal exchanges. (Lord et al., 2015) analyzed the language style synchrony between therapists and clients during MI encounters. Their approach relies on the psycholinguistic categories from the Linguistic Inquiry and Word Count lexicon to measure the degree to which the counselor language matches the client language. More recently, (Althoff et al., 2016) explored language style and symmetry in counseling interactions by analyzing a large sample of text-message-based counseling. Their main findings suggest that counselors who are more successful act with more control in the conversations and show lower levels of verbal coordination (mirroring) than their less successful counterparts.

Furthermore, there are ongoing efforts on creating annotated resources that facilitate NLP advances in the analysis of clinical text in applications such as automatic annotation of pathology reports and oncology reports as well as data from biomedical journals (Roberts et al., 2007; Albright et al., 2013; Verspoor et al., 2012). Despite this efforts, to our knowledge, there are only few psychotherapy corpora available. One of them is the "Alexander Street Press", [3] which is a large collection of transcripts and video recordings of therapy sessions on different subjects such as anxiety, depression, family conflicts, and others. There are also other psychology datasets available under limited access from the National Institute of Mental Health (NIMH).[4]

In this paper, we present the development of a counseling conversations dataset that can be used to implement data-driven methods for the automatic evaluation of counseling quality. We specifically focus on the overall conversation quality, with the final goal of providing linguistic cues associated with high-quality counseling.

[3] http://alexanderstreet.com/products/counseling-and-psychotherapy-transcripts-series

[4] http://psychiatry.yale.edu/pdc/resources/datasets.aspx

3. A Dataset of Low and High Quality Counseling

3.1. Collecting Counseling Conversations from the Web

We started by identifying video clips containing brief counseling interactions conducted using Motivational Interviewing (MI) from publicly available video-sharing sources such as YouTube and Vimeo. Keywords used to search for these videos include "motivational interviewing", "MI counseling", "effective MI", "good MI", "MI counseling demonstration", "role play MI" for the high-quality category, and "ineffective MI", "bad MI", "bad counseling", "how not to do MI", "the bad counselor" for the low-quality category. To select the videos, we used the following guidelines: the video should include only two participants, i.e., counselor and client; the video should include minimal interruptions, such as background narrative, music, or animation; the session should address a behavior change e.g. smoking cessation, drinking; and finally, the counselor-client interaction should last at least 3 minutes.

The obtained recordings consist mainly of MI counseling demonstrations from MI training services and students' MI role-play practice from undergraduate-level psychology courses. The sessions address various health topics including smoking cessation, alcohol consumption, substance abuse, weight management, mental disorders, and medication adherence and portray several practice settings such as private practice, school counseling, and pharmacy counseling among others.

After collecting our initial pool of videos, we conduct a second filtering step to verify that the counseling was conducted using MI and that the video caption matched the video content, i.e., portray either a high-quality or a low-quality counseling interaction. To evaluate MI use (or the lack of it) we followed the guidelines in MI literature (Miller and Rollnick, 2013). The criteria to label a counseling interaction as either low or high quality is as follows: during high-quality counseling, the conversation should present, to some extent, reflective listening, questions, as well as collaboration and support. In contrast, the low-quality counseling should show a predominant directive style, which includes confrontation, advising without permission, and lack of listening.

The final video set includes 151 counseling conversations. From this, 72 video clips were labeled as high-quality counseling and the remaining 79 as low-quality counseling. The length of the conversations varies from 5-20 minutes. Table 1 shows transcript excerpts corresponding to high-quality and low-quality counseling conversations in the dataset.

Preprocessing. All the videos are first converted into standard mp4 format and then preprocessed to address issues frequently present in shared video content such as introductory titles, animations, and narratives. In most cases these interruptions appeared only at the beginning of the video so we manually trimmed that portion of the video until the counselor-patient interaction started. This process can also be optimized using automatic methods such as optical character and facial recognition, however, we opted for a manual approach in order to obtain accurate examples

HIGH-QUALITY COUNSELING	LOW-QUALITY COUNSELING
T: Hi miss NAME my name is NAME. I'm a social worker at the Family Health Center *C*: Dr. NAME asked me if I would spend some time with you today *T*: I'm really glad that you here. I'm just curious as to why he would send you to me *C*: well I came to see Dr. Steele last week because of increasing stress and anxiety. That's kind of getting the best of me and in the course of my appointment with him he was asking how I was dealing with that stress and I mentioned that my one or two glasses of wine a few nights a week is turning into more frequent. *T*: yeah so he he actually see me because he went to him for increased stress and he's concerned that your alcohol consumption may be a part of that increase and prior to prescribing you anything you want to make sure that you had someone to talk to about that	*T*: Okay, so I wrote a prescription for an antibiotic for NAME that should help with the ear infection but in looking through this chart, I mean, it seems like he's had six or seven of these just in the past year or so that's really a big problem *C*: Yeah it's pretty stressful for both of us. It gets really upset *T*: Well, one of the primary risk factors for multiple ear infections and kids is actually smoke exposure. Are you smoking? *C*: Yeah. I, yeah, I do smoke but I don't smoke around him. I try really hard not to smoke around him *T*: Well, the fact that he's having these ear infections is indicating to me that he is being exposed to smoke and so what can you tell me about that?

Table 1: Transcript excerpts corresponding to high-quality and low-quality counseling conversations

Code	Count	Verbal examples
Question	1122	What do you think it would take to change your mind about participating in physical activity?
Reflection	813	It sounds like you're concerned by your weight and you want to start to make positive changes.

Table 2: Frequency counts and verbal examples of Questions and Reflections in the dataset

of counselor-patient interactions.

Transcription. In order to transcribe the video clips, we adopt a semi-automatic approach. First, we use the YouTube automatic captioning to obtain the conversation transcript and then we manually labeled the conversation turns as either counselor or patient speech.

3.2. Annotation of Counseling Skills

Seeking to evaluate the counseling interaction between counselors and patients, we decided to annotate two core interview micro-skills in counseling practice: reflective listening and questions (Tollison et al., 2008). These two micro-skills are assessed using the Motivational Interviewing Treatment Integrity (MITI) coding scheme version 4 (Moyers et al., 2016), which is the current gold standard for MI fidelity evaluation. Hence, all the video clips in our dataset are manually annotated to identify questioning (Questions) and reflective listening statements (Reflections).

In order to conduct the annotation, two undergraduate students were trained in the use of the MITI 4.0. During this training, the annotators learned how to parse the counselor-patient interaction (i.e., deciding which portion of the conversation shows the given behavior), practiced the correct assignment of behavior codes, and conducted team coding on sample sessions.

The 151 sessions were randomly distributed among the two annotators. During the coding process, the annotators used both the audio recording and the transcript. The annotation was conducted at conversation turn-level using Nvivo,[5] an annotation and quantitative analysis suite that allows selecting text in the transcript and labeling it with a given code, e.g., reflection or question.

In order to verify the reliability of the annotations, we calculated the inter-annotator agreement in a sample of 20 counseling conversations, with even distribution for the Low-quality and High-quality categories. The intra-class correlation scores for both Questions and Reflections codes are 0.96 and 0.94, respectively, thus showing good levels of agreement between the two annotators.

4. Discriminating Between High-quality and Low-quality Counseling

4.1. Analysis of Counseling Conversations

We start by exploring linguistic differences between the counseling interactions to get insights into the mechanisms of high-quality counseling. Our analyses are based on the semantic word classes from the LIWC lexicon and the semantic word-class scoring by (Mihalcea and Pulman, 2009). Table 3 shows the top classes for both, low and high quality counseling.

The results show interesting differences between the two types of conversations. While high-quality counseling focus on aspects related to encouragement and reflective listening such as family, positive feeling, feelings, and hearing, low-quality counseling shows a more directive lan-

[5] http://www.qsrinternational.com/what-is-nvivo

High quality Counseling			Low quality Counseling		
Class	Score	Sample words	Class	Score	Sample words
Family	2.04	Mom, wife, parents, husband	Self	1.33	I, we, me, my, our
Feel	1.85	Feel, pain, feeling, sense	Negate	1.31	Not, don't, no, can't, without
Posfeel	1.52	Like, care, enjoy, glad	Inhibition	1.81	quit, stop, control, avoid
Anxiety	1.38	Afraid, worried, overwhelmed	Time	1.06	Now, start, today, before
Optimism	1.31	Ready, hope, confidence, determined	Present	1.06	Know, do, need, want
Hear	1.25	Sounds, heard, talking, said	Pronoun	1.04	You, I, it, your, we

Table 3: Results from LIWC word class analysis. Top ranked semantic classes associated to low and high quality counseling are shown.

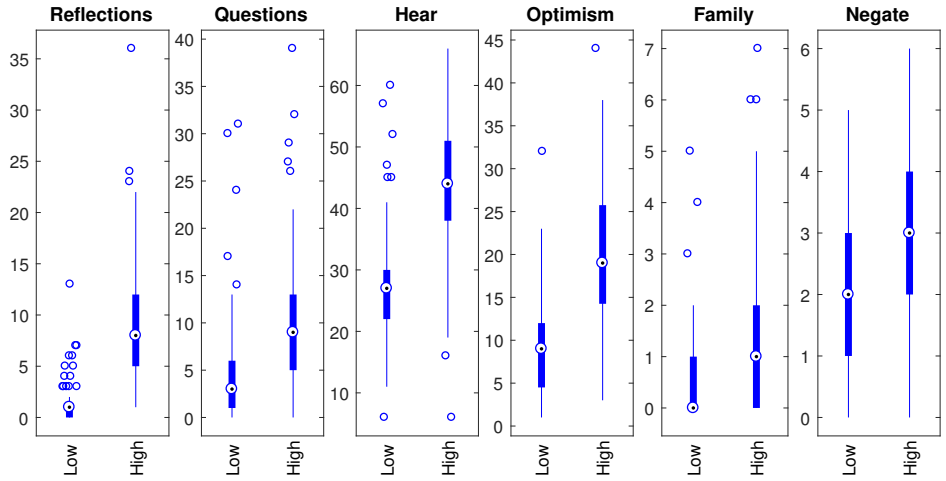


Figure 1: Distribution of reflections, questions, listening (hear), optimism, family and negation (negate) word classes in high and low quality counseling

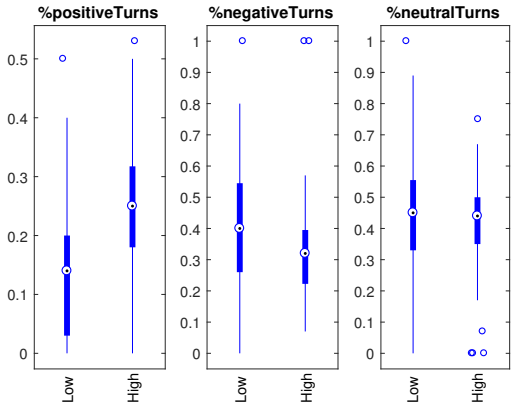


Figure 2: Distribution of positive, negative, and neutral counselor turns in high and low quality counseling

guage by using more self-references, inhibition and negation words.

To further analyze these trends, we plot the distribution of reflections, questions, and lexicon word-classes over the two types of counseling conversations. Results shown in Figure 1 show important differences between the low and high quality groups, thus suggesting that they are potentially good predictors of counseling quality.

In addition, we analyzed the sentiment expressed by the counselor during the encounters as a potential predictor of counseling quality. This could provide information on whether counselors focus on positive or negative aspects of the client communication, and how this relates to the conversation quality, i.e., low or high. Thus, we analyze the sentiment expressed by counselors during each turn in the conversation. Given the effort required to manually annotated the sentiment in each conversation, we opted for using an automatic off-the-shelf sentiment classifier from the Stanford Core NLP package (Manning et al., 2014). We obtain a sentiment score for each counselor turn, scored from very negative to very positive, and calculate the percentage of positive, negative, and neutral turns during the conversation. Figure 2 shows the sentiment distribution over the high-quality and low-quality conversations. The box plots in the figure suggest differences between the low and high quality groups, particularly for positive sentiment. In order to look more closely into the positive sentiment trend during the counseling encounters, we plot the distribution of positive turns by the counselor across the low and high quality counseling conversation. The plot in Figure 3 shows that counselors increasingly focus on positive aspects of the client expressions, particularly during high-quality conversations.

4.2. Distinguishing Between High and Low Quality Counseling

In this section, we explore the use of linguistic cues to build a computational model that predicts the overall quality of

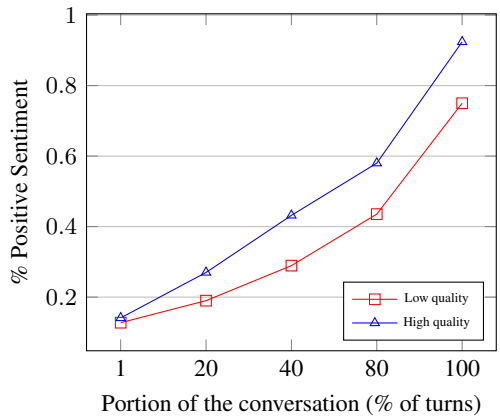


Figure 3: Positive sentiment across five equal segments of the conversation duration.

the counseling conversation

The feature set consists of the cues identified during our exploratory analyses as potential indicators of counseling quality, as well as additional text features used during standard NLP feature extraction such as ngrams. The features are extracted from the transcripts of the counseling conversations. During our experiments, we first explore the predictive power of each cue separately, followed by an integrated model that attempts to combine all the linguistic cues to improve the prediction of counseling quality. The different features are as follows:

***N-grams*:** These features represent the language used by the counseling-conversation participants and include all the unique words and word-pairs present in the transcript. We extract a vector containing the frequencies of each word and word pair present in the transcript.

***Semantic information*:** We use categories from the LIWC (Tausczik and Pennebaker, 2010), Opinion Finder (Wilson et al., 2005) and the Wordnet Affect (Strapparava and Valitutti, 2004) lexicons to derive features that identify identifying words as belonging to certain semantic categories that are potential markers of the conversation quality.

Metafeatures: We also extract a set of metafeatures that describe the conversation interaction, including the number of counselor turns, client turns, average words during client and counselor turns, and the ratio of counselor and client words in each turn.

Sentiment: These features are designed to capture the sentiment trend in the counselor responses during the conversation. To derive this features, we first obtain the sentiment expressed by counselors during each turn, scored from very negative to very positive $(--, -, 0, +, ++)$ using the sentiment analysis classifier from Stanford Core NLP, and then obtained a set of descriptors that capture the sentiment trend. The set includes the percentage of positive, negative, and neutral turns during the conversation, the number of times the sentiment changes during the conversation, as well as counts of sequences increasing and decreasing sentiment intensity i.e., $-+$, $-++$, $-+$, $+--$.

Feature set	Acc.	F-score	
		High	Low
Baseline	52.31%		
Ngrams	82.78%	0.82	0.83
Lexicons	72.84%	0.71	0.73
Metafeatures	76.15%	0.77	0.74
MITI Behav	83.44%	0.83	0.83
Sentiment	70.86%	0.69	0.81
All features	**87.41%**	0.87	0.87

Table 4: Overall prediction results and F-scores for high-quality and low-quality counseling conversations using several linguistic feature sets.

Feature set	Acc.
All features	**87.41%**
– Ngrams	83.44%
– Lexicons	85.43%
– Sentiment	86.10%
– Metafeatures	87.41%
– MITI Behav	87.41%

Table 5: Feature ablation study.

MITI behaviors This set includes the number of reflections and questions by the counselor during the conversation as well as the ratio of reflections to questions. The counts are derived from the turn-level annotations described in section 3.2.

We conduct several experiments to discriminate between low-quality and high-quality encounters. During our experiments, the evaluations are done at conversation level. The classifiers are built using the Support Vector Machine algorithm[6] and the different sets of linguistic features. We perform leave-one-out cross-validation in all our experiments and we use the majority class baseline as a reference value.

Results shown in Table 4 show that all the feature sets perform above the baseline, with the MITI behaviors being the best performing features, followed by the n-grams features. We also observe that the combination of all feature sets provides the best performance.

Seeking to explore the role played by the different feature sets, we conduct an ablation study, where we remove one feature set at the time from the best performing model i.e., "all features". As observed in Table 5, the ngrams features contribute the most to the final model, followed by the lexicon features. Interestingly, the results show that the combination of n-grams and lexicons offer similar performance as the MITI behaviors features. These results are encouraging as they suggest that standard linguistic features can achieve similar performance as manually coded features (MITI behaviors) while evaluating the overall quality of counseling conversations.

[6] As implemented in the Weka library.

5. Conclusion and Future Work

In this paper, we introduced a new dataset of low-quality and high-quality counseling conversations that were collected from public sources. Through several classification experiments, we showed that such a dataset can be used to build accurate classification models able to discriminate between low-quality and high-quality counseling, with accuracy figures up to 87%.

Furthermore, we showed that standard NLP features can provide performance similar to manually coded features for this task.

We also provided insights into the linguistics markers of high-quality counseling and showed that it is characterized by positive and encouraging language.

6. References

Albright, D., Lanfranchi, A., Fredriksen, A., Styler, W. F., Warner, C., Hwang, J. D., Choi, J. D., Dligach, D., Nielsen, R. D., Martin, J., et al. (2013). Towards comprehensive syntactic and semantic annotations of the clinical narrative.

Althoff, T., Clark, K., and Leskovec, J. (2016). Large-scale Analysis of Counseling Conversations: An Application of Natural Language Processing to Mental Health. *Transactions of the Association for Computational Linguistics*.

Apodaca, T. R., Borsari, B., Jackson, K. M., Magill, M., Longabaugh, R., Mastroleo, N. R., and Barnett, N. P. (2014). Sustain talk predicts poorer outcomes among mandated college student drinkers receiving a brief motivational intervention. *Psychology of Addictive Behaviors*, 28(3):631.

Atkins, D. C., Rubin, T. N., Steyvers, M., Doeden, M. A., Baucom, B. R., and Christensen, A. (2012). Topic models: A novel method for modeling couple and family text data. *Journal of family psychology*, 26(5):816.

Atkins, D. C., Steyvers, M., Imel, Z. E., and Smyth, P. (2014). Scaling up the evaluation of psychotherapy: evaluating motivational interviewing fidelity via statistical text classification. *Implementation Science*, 9(1):49.

Can, D., Georgiou, P. G., Atkins, D. C., and Narayanan, S. S. (2012). A case study: Detecting counselor reflections in psychotherapy for addictions using linguistic features. In *INTERSPEECH*, pages 2254–2257. ISCA.

Catley, D., Harris, K. J., Goggin, K., Richter, K., Williams, K., Patten, C., Resnicow, K., Ellerbeck, E., Bradley-Ewing, A., Malomo, D., et al. (2012). Motivational interviewing for encouraging quit attempts among unmotivated smokers: study protocol of a randomized, controlled, efficacy trial. *BMC public health*, 12(1):456.

Charles, C., Gafni, A., and Whelan, T. (1997). Shared decision-making in the medical encounter: What does it mean? (or it takes at least two to tango). *Social Science & Medicine*, 44(5):681 – 692.

Chava, Z. (2014). Expenditures for mental health among adults, ages 18-64, 2009-2011: Estimates for the u.s. civilian noninstitutionalized population. *Agency for Healthcare Research and Quality, Rockville, MD*, Statistical Brief #454.

Gaume, J., Gmel, G., Faouzi, M., and Daeppen, J.-B. (2009). Counselor skill influences outcomes of brief motivational interventions. *Journal of Substance Abuse Treatment*, 37(2):151 – 159.

Harting, J., van Assema, P., van der Molen, H. T., Ambergen, T., and de Vries, N. K. (2004). Quality assessment of health counseling: performance of health advisors in cardiovascular prevention. *Patient Education and Counseling*, 54(1):107 – 118.

Klonek, F. E., Quera, V., and Kauffeld, S. (2015). Coding interactions in motivational interviewing with computer-software: What are the advantages for process researchers? *Computers in Human Behavior*, 44:284–292.

Lord, S. P., Sheng, E., Imel, Z. E., Baer, J., and Atkins, D. C. (2015). More than reflections: empathy in motivational interviewing includes language style synchrony between therapist and client. *Behavior therapy*, 46(3):296–303.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Mihalcea, R. and Pulman, S. (2009). Linguistic ethnography: Identifying dominant word classes in text. In *Computational Linguistics and Intelligent Text Processing*, pages 594–602. Springer.

Miller, W. R. and Rollnick, S. (2013). *Motivational interviewing: Helping people change, Third edition*. The Guilford Press.

Moyers, T. B., Martin, T., Houck, J. M., Christopher, P. J., and Tonigan, J. S. (2009). From in-session behaviors to drinking outcomes: a causal chain for motivational interviewing. *Journal of consulting and clinical psychology*, 77(6):1113.

Moyers, T. B., Rowell, L. N., Manuel, J. K., Ernst, D., and Houck, J. M. (2016). The motivational interviewing treatment integrity code (miti 4): Rationale, preliminary reliability and validity. *Journal of Substance Abuse Treatment*, 65(Supplement C):36 – 42. Motivational Interviewing in Substance Use Treatment.

Roberts, A., Gaizauskas, R., Hepple, M., Davis, N., Demetriou, G., Guo, Y., Kola, J., Roberts, I., Setzer, A., Tapuria, A., and Wheeldin, B. (2007). The CLEF corpus: semantic annotation of clinical text. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, pages 625–629.

Strapparava, C. and Valitutti, A. (2004). WordNet-Affect: an affective extension of WordNet. In *Proceedings of LREC*, volume 4, pages 1083–1086.

Tanana, M., Hallgren, K., Imel, Z., Atkins, D., Smyth, P., and Srikumar, V. (2015). Recursive Neural Networks for Coding Therapist and Patient Behavior in Motivational Interviewing. *2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 71–79.

Tausczik, Y. R. and Pennebaker, J. W. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psy-*

chology, 29(1):24–54.

Tollison, S. J., Lee, C. M., Neighbors, C., Neil, T. A., Olson, N. D., and Larimer, M. E. (2008). Questions and reflections: The use of motivational interviewing microskills in a peer-led brief alcohol intervention for college students. *Behavior Therapy*, 39(2):183 – 194.

Vader, A. M., Walters, S. T., Prabhu, G. C., Houck, J. M., and Field, C. A. (2010). The language of motivational interviewing and feedback: counselor language, client language, and client drinking outcomes. *Psychology of Addictive Behaviors*, 24(2):190.

Verspoor, K., Cohen, K. B., Lanfranchi, A., Warner, C., Johnson, H. L., Roeder, C., Choi, J. D., Funk, C., Malenkiy, Y., Eckert, M., et al. (2012). A corpus of full-text journal articles is a robust evaluation tool for revealing differences in performance of biomedical natural language processing tools. *BMC bioinformatics*, 13(1):1.

Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. (2005). Opinionfinder: A system for subjectivity analysis. In *Proceedings of HLT/EMNLP on Interactive Demonstrations*, HLT-Demo '05, pages 34–35. Association for Computational Linguistics.

Xiao, B., Bone, D., Van Segbroeck, M., Imel, Z. E., Atkins, D. C., Georgiou, P. G., and Narayanan, S. S. (2014). Modeling therapist empathy through prosody in drug addiction counseling. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Interpersonal Relationship Labels for the CALLHOME Corpus

Denys Katerenchuk[1], **David Guy Brizan**[1,2], **Andrew Rosenberg**[3]
[1]CUNY Graduate Center
365 Fifth Avenue, New York, NY 10016
[2]University of San Francisco
101 Howard Street, San Francisco, CA 94105
[3]IBM Research AI
1099 Kitchawan Rd., Yorktown Heights, NY, 10598
[1]dkaterenchuk@gradcenter.cuny.edu, [2]dgbrizan@usfca.edu, [3]amrosenb@us.ibm.com

Abstract

The way we speak to our friends, colleagues, or partners is different in both the explicit context, what we say, and the implicit, how we say it. Understanding these differences is important because it provides additional information that can be used in natural language processing tasks. For example, knowing the relationship between interlocutors can help to narrow the range of topics and improve automatic speech recognition system results. Unfortunately, the lack of corpora makes exploration of this problem intractable. In this work, we release a set of interpersonal relationship labels between conversation participants for the CALLHOME English corpus. We make the labels freely available for download on our website and hope that this effort can further boost research in this area.

Keywords: corpus annotation, interlocutor relationship labels, CALLHOME corpus

1. Introduction

Communication is one of the most invaluable tools humans have. It enables us to understand each other, share ideas, coordinate work, and build relationships. Through speech, we carry explicit information in terms of words, as well as implicit information that is usually expressed by an acoustic signal of the human voice. For example, when you are on the phone with a friend, it is often clear when the person is distressed, even when they are trying to conceal or deny this fact. Word choice and speech intonation are formed on the conscious and unconscious levels. Unlike the conscious level, we have little control over the unconscious. Understanding the relation between explicit and implicit information during conversations can provide additional information about the speakers.

We have all been in a park or a restaurant and overheard or observed strangers interacting with each other. Have you ever tried to guess the relationships between them? Were they two old friends, cousins, or maybe a couple? On what distinct characteristics did you base your assumption? Was it the body language, voice, or word choice? It may seem like an easy task for humans to identify the relationships. However, this is a very difficult task for computers.

People are interacting with cellphones, smart TVs, and computers on a daily basis using voice-based interfaces. However, these interactions can be harmed by misunderstandings. One reason for the occurrence of the misunderstandings is that these systems rely on automated speech recognition (ASR) systems, which, despite showing strong improvements in performance, are far from perfect. One way to improve an ASR system's performance is to take into account not only spoken words, but also speaker and domain information. For example, information about speaker's gender, age, or voice characteristics can be used to reduce recognition word error rate (Saon et al., 2013; Serizel and Giuliani, 2017). The knowledge of a discussion topic, in a similar fashion, can aid the accuracy (Chen et al., 2015). If conversations involve multiple speakers, knowing the relationships among the participants may be beneficial because many times the topics and voice differ when we converse with colleagues or family members. Unfortunately, there are very few resources available in this area. As noted by (Kendall, 2011), many existing corpora lack the labels researchers need to investigate the effects that interlocutor relationships play in language change.

The main focus of this paper is to provide a set of labels to boost research of language and its differences between family members and friends. The results of such research can be used in improving multiple NLP areas. We release a set of annotated labels for the well-known CALLHOME English corpus of phone dialogues (Canavan et al., 1997). The labels are available for download at `https://github.com/dkaterenchuk/callhome_labels`.

2. Related Work

In recent years, a great deal of notable research has been done on studying implicit information from speech conversations and written dialogues. The early work in this domain by Stirman and Pennebaker (2001) has shown that there is a correlation between word choices and the mental states of the authors. Their work analyzed poetry documents to identify suicidal writers. They found that these authors tend to use more words that are related to themselves rather than to others. Another paper, from authors in the same research group, showed that it is possible to identify the level of romantic interest during a speed dating session and also to predict the likelihood of a long term relationship (Ireland et al., 2011). This work is based on the analysis of word choice and language style, known as linguistic style matching.

Speech contains a rich source of implicit information and a lot of work has been done to study its communication. For example, Rao et al. (2012) and Han et al. (2014) (among many others) show that voice can carry information about emotions. Polzehl et al. (2010) proposed a method for predicting a speaker's personality traits. This information about speakers can be used to create personalized responses of conversational agents as described in the work by Siddique et al. (2017). In addition to speaker information, our voices carry data about intent and deception, as was shown in the work by Sanaullah and Gopalan (2013), Levitan et al. (2015b) and Mendels et al. (2017). The ways we converse with coworkers or partners is also unique. The study of Spanish phone conversations by Yella et al. (2014) shows that with the accuracy of 75%, it is possible to recognize if a conversation is between partners or family members. Previously, we studied a similar problem of identifying relationships between friends and relatives (Katerenchuk et al., 2014). The results confirm that the way we speak to our friends is different from conversations amongst family members.

These research efforts and their outcomes were made possible by the data availability. For example, the release of the SpeedDate corpus (Ranganath et al., 2009) made working on investigation of romantic interactions possible to researchers. Similarly, Maekawa et al. (2000) and Campbell (2002) collected spontaneous speech of Japanese speakers that lead to improvements including phoneme recognition (Fourtassi et al., 2014), domain adaptation (Asami et al., 2017), etc. The most similar dataset to ours is the CallNotes corpus (Carrascal et al., 2012). This is a set of phone conversations designed for speech analysis. The main difference is that the data is collected from native Spanish speakers. The dataset of our choice is based on phone conversations too, but the conversations are collected from native English speakers. Through this work, we hope to encourage research in understating vocal and textual differences between conversation participants.

3. Corpus Description

This section provides an overview of our decision to work with the CALLHOME English corpus, the description of the dataset, and the process for interpersonal relation label creation.

3.1. Data Requirements

Data collection is often an expensive and time-consuming process. For this reason, we decide to look at available English dialogue corpora. The CALLHOME English corpus (Canavan et al., 1997) is a well known resource of English phone conversations. The main advantage of this dataset is that it complies with our requirements:

- The dialogs are in English.
- The conversations are dyadic[1].
- The speech is spontaneous.
- The participants are free to discuss any topic.

These requirements are important for the following reasons: 1) the annotators speak English and can perform labeling task reliably, 2) the real-world conversations that are not enforced by a specific topic provide natural research environment for the future analysis. For these reasons, we find the CALLHOME English corpus is a great choice of data for our annotation task.

3.2. Data Description

The CALLHOME English corpus was developed by the Linguistic Data Consortium (LDC) and contains 120 unscripted phone conversations between native English speakers. The speakers are representatives of various demographic groups. The conversation participants were aware of the recordings; however, the conversations were on any topic of their choice and did not have any additional constraints. All phone calls were placed from North America to friends or family members who largely live outside of the USA and Canada. Each phone conversation is around 30 minutes in length for a total of 56.7 hours of audio. The conversations are divided into train (80 conversations), development (20) and test (20) sets.

The CALLHOME English corpus also provides transcripts. The transcripts cover a continuous 5 or 10 minute segment taken from a recorded conversation. The total time of transcribed audio is 18.3 hours. The transcribers were given a set of instructions that limit the transcribed segment to the middle of the conversation, preserve disfluencies, sounds, simultaneous speech and mispronunciations. Additional instructions and corpus descriptions appear in Canavan et al. (1997).

The corpus also provides anonymized speaker data. The information, presented in the corpus, describes speaker's call ID, gender, age, years of education completed, state where the speaker grew up, and country or area code with first three digits of the dialed number. While the corpus supplies speaker information, it omits any data about interpersonal relationships between the speakers.

3.3. Annotation

The annotations we provide were performed by a group from the Speech Lab @ Brooklyn College, CUNY (formerly of Queens College, CUNY). The annotators were asked to listen to the full conversations and refer to the transcripts, where available, to identify relationships between the call participants. The decision for each label is based on evidence from the conversation. The evidence could be a spoken or transcribed phrase such as *"our parents"* that signifies the speakers are siblings or a direct speech, such as *"hello mom,"* that shows that the conversation is between a parent and a child. Annotators described the relationship using any term they like. However, all annotations were entered into a shared document, which led to a relatively rapid convergence to a small set of labels.

[1]There is one case in which multiple speakers appear on the call. This is noted in our labels.

Despite this, there are still some individual differences in the labels that are resolved after annotation is completed.

FRIEND	RELATIVE	
80	28	
FRIEND	SIBLING	PARENT-CHILD
80	15	13

Table 1: Label distribution

We find that most conversations are between friends – some of whom could be identified as work colleagues. We ultimately settled on two binary interpersonal relationships, FRIENDS and FAMILY, for the main label set. The line between these groups can be very thin since very close friends may feel like relatives and cousins or siblings may also be friends.

We were unable to find a finer-grained distinction of *types* of friends reliably across the whole corpus. As a result, the friendship sub-categories are not available and the conversation is labeled as "friend" in both cases. One conversation, numbered 5046, stands of an exemplar of the reasons why: the participant friends showed familiarity with each others' families, the details of their homes and obligations to send presents in celebration of birthdays–a friendship of pleasure. Despite this, our annotators also determined that the relationship likely started with the pair having worked together–a friendship of utility. If we employ Aristotelean friendship categories, this relationship likely falls into at least two bins. We find many examples of this complex, multi-class friendship type in the CALLHOME corpus.

In the case of family members, in contrast to friends, we provide additional labels that further define the relationships. These additional labels consist of relationships such as mother, father, sister, brother, and cousin for each participant of the call, where they could be determined.

The annotation task is non-trivial in many cases. We are unable to provide labels for 12 conversations (10% of the corpus) because 1) the relationship cannot not be determined with confidence or, 2) in two instances, more than two speakers joined the conversation. These situations cause the interpersonal relationship between the speaking parties to change over the course of the conversation. An interesting quality of the CALLHOME data is that a small number of the conversations is between representatives of a religious group who refer to each other as "sisters," when they are actually friends or colleagues. In these cases, the annotators have to find additional evidence of the relations and disregard these direct addresses.

In total, there are 108 annotated phone conversations. A summary of the data annotation can be found in the Table 1. The majority of instances, 80 out of 108, are labeled as FRIEND. The remaining 28 conversations are between family members and labeled as RELATIVE. The finer grained distinction between relative types is defined by 15 instances of conversions between siblings and 13 between parents and children. This creates a highly unbalanced corpus. For this reason we provide the labels as a single set without a division for training, developing and testing subsets. We leave the normalization method or an appropriate use case of the data up to the user. The annotations of the CALLHOME English corpus are available at `https://github.com/dkaterenchuk/callhome_labels`.

4. Data Analysis

We report our initial results on classifying interpersonal relationships that appeared in our previous work (Katerenchuk et al., 2014). During this initial exploration, we use a subset of the annotated data. The data consists of 56 phone conversations where 28 conversations are between friends and 28 are between relatives. Furthermore, we use 10-fold cross validation during the classification. In our experiments we use acoustic and textual data representations.

Our acoustic data representation pipeline is based on openSMILE, an open-source tool (Eyben et al., 2010). OpenSMILE provides a set of configuration files for acoustic feature extraction. We use the emotion.conf configuration from IS09 (Schuller et al., 2009). This configuration extracts 384 features that includes five low-level descriptors (LLDs) of acoustic features: 1) Zero crossing rate, 2) RMS Energy, 3) F0, 4) Harmonic-to-Noise Ratio, and 5-16) 12 MFCC coefficients. The change (Δ) of each of these LLDs is also calculated. This leads to a total of $16 \cdot 2 = 32$ LLDs. Twelve functionals are then applied to these: 1) mean, 2) standard deviation, 3) skewness, 4) kurtosis, 5-8) value and relative position of minima and maxima, 9) range between minima and maxima, 10-12) linear regression coefficient, offset and MSE.

Textual representation is extracted from the transcripts. Since we wanted to investigate the relationships, we use a set of words proposed by Chung and Pennebaker (2007). In their work they show that function words, such as pronouns, articles and prepositions, are highly correlated with the speakers' attributes. The counts for each of these is used as a representation. In addition, we use turn-taking information, interruptions, cuts off, delays in response, and other conversation related data.

The problem of identifying interpersonal relationships is cast as a classification task. The models are trained using both acoustic and textual data representations. We would like to point out that during our exploration the feature set is larger than the number of data points. In the real word system this set up is not ideal and might lead to overfitting. However, it should provide an intuition for further exploration. The goal is to investigate if any features contain predictive information and can identify the relationships between the speakers. The choice of our learning algorithms was limited to: 1) SMO, an SVM optimization algorithm, 2) J48, a decision tree algorithm, 3) Naive Bayes, and 4) BayesNet, a Bayesian Network

Classifier	Acoustic Feature	Text Features	Acoustic + Text	Acoustic Side A	Acoustic Side B	Acoustic Segment
SMO	42.9%	57.1%	39.3%	37.5%	48.2%	48.2%
J48	55.6%	57.1%	60.7%	60.7%	62.5%	57.1%
Naive Bayes	44.6%	60.7%	57.1%	50.0%	37.5%	50.0%
BayesNet	55.4%	55.4%	51.9%	55.6%	73.2%	51.8%

Table 2: Results

learning algorithm. In addition, we create experiments to analyze different settings of conversations and answer the following questions:

1. Can we identify relationships from a conversation?
2. Do we need to hear both sides of the conversation?
3. Is the whole conversation required to make a prediction?

The results of the experiments are shown in Table 2. From the table we can see that providing a full conversation, both acoustic and textual representations are indicative of the speaker relationships. However, the text based representation seems to have more information for this task, yielding 60.7% accuracy when using the Naive Bayes algorithm. Combining both representations achieved the same accuracy but with a different learning algorithm. From the analysis of the features, the MFCC-based acoustic signal is the most informative of the relationships. An interesting fact was discovered from transcript extracted text features. We found that conversations between friends are more egocentric and are reflected in higher frequencies of personal pronouns such as "my" and "I" (Table 3). In contrast, relatives appear to be more likely to discuss other people, which corresponds to a higher usage of third person pronouns. For more details on the feature importance we refer the reader to our previous work (Katerenchuk et al., 2014).

From the analysis of only one side of a conversation, we find that predictive results improve and produce the accuracy of 73.2%. This stronger result, however, comes with a caveat – only one speakers of the pair shows a strong predictive signal. In the case of current dataset, speakers receiving the call show higher predictive results. It is possible that this is a phenomenon of the distribution of data or that it can be attributed to the specifics of the callers; in this case that the speaker on side A places a call to a speaker on side B, who is likely located outside North America and who may share experiences which are more likely to be classified correctly.

Lastly, we explore the case where only a part of a conversation is available. From each audio, we extract a segment of 10 minutes from the middle of a conversation. We find that the accuracy increases in the majority of cases. This can be attributed to a number of possible causes including the fact that the speakers can be uncomfortable with being recorded and thus tend to be cautious at the start of the conversation,

Text Feature	Gain Value	Dir.
my_PER_(TOKEN,UTT,SEC)	0.19, 0.19, 0.19	↑
i_PER_UTT	0.19	↑
so_PER_(TOKEN,UTT,SEC)	0.18, 0.18, 0.16	↑
of_PER_SECOND	0.15	↑
a_PER_SECOND	0.15	↑
had_PER_UTT	0.13	↑
max Δ mfcc[6]	0.18	↑
Linear Regression slope F0	0.18	↓
max Δ mfcc[10]	0.16	↓

Table 3: Info gain valuesh

using more stilted language. Also, accommodation theory or entrainment may provide an explanation. Niederhoffer and Pennebaker (2002) discovered that conversation participants tend to mimic each others' styles. Levitan et al. (2012) and Levitan et al. (2015a) showed that this behavior remains persistent through speech as well. For an extensive analysis of the representations and models, refer to our previous work (Katerenchuk et al., 2014).

5. Conclusion

We release a set of labels for the CALLHOME English telephone conversation corpus. The labels describe the relationships between the participants as friends or family members. This dataset should enable researchers to work on analyzing textual and acoustic information in conversations among friends or family. Understanding the patterns may enable researchers to use this knowledge and improve various NLP tasks. The labels are freely available for download at `https://github.com/dkaterenchuk/callhome_labels`.

6. Acknowledgements

We would like to thank Min Ma, Michelle Morales, Rachel Rakov, Syed Reza, and Clayton Violand for their annotation work in this project.

This material is based on research sponsored by DARPA under agreement number FA8750-13- 2-0041. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

7. Bibliographical References

Asami, T., Masumura, R., Yamaguchi, Y., Masataki, H., and Aono, Y. (2017). Domain adaptation of dnn acoustic models using knowledge distillation. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 5185–5189. IEEE.

Campbell, N. (2002). Recording techniques for capturing natural every-day speech. In *LREC*.

Canavan, A., Graff, D., and Zipperlen, G. (1997). Callhome american english speech. Linguistic Data Consortium, Philadelphia.

Carrascal, J., de Oliveira, R., and Cherubini, M. (2012). A Note Paper on Note-Taking: Understanding Annotations of Mobile Phone Calls.

Chen, X., Tan, T., Liu, X., Lanchantin, P., Wan, M., Gales, M. J., and Woodland, P. C. (2015). Recurrent neural network language model adaptation for multi-genre broadcast speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Chung, C. and Pennebaker, J. W. (2007). The psychological functions of function words. *Social communication*, pages 343–359.

Eyben, F., Wöllmer, M., and Schuller, B. (2010). Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM.

Fourtassi, A., Schatz, T., Varadarajan, B., and Dupoux, E. (2014). Exploring the relative role of bottom-up and top-down information in phoneme learning. In *ACL (2)*, pages 1–6.

Han, K., Yu, D., and Tashev, I. (2014). Speech emotion recognition using deep neural network and extreme learning machine. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Ireland, M. E., Slatcher, R. B., Eastwick, P. W., Scissors, L. E., Finkel, E. J., and Pennebaker, J. W. (2011). Language style matching predicts relationship initiation and stability. *Psychological science*, 22(1):39–44.

Katerenchuk, D., Brizan, D. G., and Rosenberg, A. (2014). "Was that your mother on the phone?": Classifying interpersonal relationships between dialog participants with lexical and acoustic properties. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Kendall, T. (2011). Corpora from a sociolinguistic perspective. *Revista Brasileira de Linguística Aplicada*, 11(2):361–389.

Levitan, R., Gravano, A., Willson, L., Benus, S., Hirschberg, J., and Nenkova, A. (2012). Acoustic-prosodic entrainment and social behavior. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies*, pages 11–19. Association for Computational Linguistics.

Levitan, R., Benus, S., Gravano, A., and Hirschberg, J. (2015a). Entrainment and turn-taking in human-human dialogue. In *AAAI Spring Symposium on Turn-Taking and Coordination in Human-Machine Interaction*.

Levitan, S. I., An, G., Wang, M., Mendels, G., Hirschberg, J., Levine, M., and Rosenberg, A. (2015b). Cross-cultural production and detection of deception from speech. In *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*, pages 1–8. ACM.

Maekawa, K., Koiso, H., Furui, S., and Isahara, H. (2000). Spontaneous speech corpus of japanese. In *LREC*.

Mendels, G., Levitan, S. I., Lee, K.-Z., and Hirschberg, J. (2017). Hybrid acoustic-lexical deep learning approach for deception detection. *Proc. Interspeech 2017*, pages 1472–1476.

Niederhoffer, K. G. and Pennebaker, J. W. (2002). Linguistic style matching in social interaction. *Journal of Language and Social Psychology*, 21(4):337–360.

Polzehl, T., Moller, S., and Metze, F. (2010). Automatically assessing personality from speech. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pages 134–140. IEEE.

Ranganath, R., Jurafsky, D., and McFarland, D. (2009). It's not you, it's me: detecting flirting and its misperception in speed-dates. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 334–342. Association for Computational Linguistics.

Rao, K. S., Kumar, T. P., Anusha, K., Leela, B., Bhavana, I., and Gowtham, S. (2012). Emotion recognition from speech. *International Journal of Computer Science and Information Technologies*, 3(2):3603–3607.

Sanaullah, M. and Gopalan, K. (2013). Deception detection in speech using bark band and perceptually significant energy features. In *Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on*, pages 1212–1215. IEEE.

Saon, G., Soltau, H., Nahamoo, D., and Picheny, M. (2013). Speaker adaptation of neural network acoustic models using i-vectors. In *ASRU*, pages 55–59.

Schuller, B., Steidl, S., and Batliner, A. (2009). The interspeech 2009 emotion challenge. In *Tenth Annual Conference of the International Speech Communication Association*.

Serizel, R. and Giuliani, D. (2017). Deep-neural network approaches for speech recognition with heterogeneous groups of speakers including children. *Natural Language Engineering*, 23(3):325–350.

Siddique, F. B., Kampman, O., Yang, Y., Dey, A., and Fung, P. (2017). Zara returns: Improved personality induction and adaptation by an empathetic virtual agent. *Proceedings of ACL 2017, System Demonstrations*, pages 121–126.

Stirman, S. W. and Pennebaker, J. W. (2001). Word use in the poetry of suicidal and nonsuicidal poets. *Psychosomatic medicine*, 63(4):517–522.

Yella, S. H., Anguera, X., and Luque, J. (2014). Inferring social relationships in a phone call from a single party's speech. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4843–4847. IEEE.

Text Mining for History: first steps on building a large dataset

Suemi Higuchi, Cláudia Freitas, Bruno Cuconato, Alexandre Rademaker
FGV/CPDOC and PUC-Rio, PUC-Rio, FGV/EMAp, IBM Research and FGV/EMAp
suemi.higuchi@fgv.br, claudiafreitas@puc-rio.br, bcclaro@gmail.com, alexrad@br.ibm.com

Abstract

This paper presents the initial efforts towards the creation of a new corpus on the history domain. Motivated by the historians' need to interrogate a vast material in a non-linear way, our approach privileges deep linguistic analysis on an encyclopedic-style data. In this context, the work presented here focuses on the preparation of the corpus, which is prior to the mining activity: the morphosyntactic annotation and the definition of semantic types for entities and relations relevant to the History domain. Taking advantage of the semantic nature of appositive constructions, we manually analyzed a sample of eleven hundred sentences in order to verify its potential as additional semantic clues to be considered. The results show that we are on the right track.

Keywords: digital humanities, text mining, corpus annotation, appositives

1. Introduction

Language is a rich repository of information about our practices, constituting raw material for research in Human and Social Sciences. In close connection with Computational Linguistics, Humanities and Social Sciences, the growing field of the Digital Humanities has at its disposal tools and resources that offer new ways to explore many available corpora.

In this paper we present our initial efforts towards the creation of a resource dedicated to text mining in the history domain. The mining strategy is linguistically motivated: inspired by (Hearst, 1992) we assume that certain semantic relations have a linguistic realization, and therefore the inclusion of linguistic metadata such as part-of-speech, lemma, and syntactic information in the corpus is essential. The target of the mining - the corpus - is the *Dicionário Histórico-Biográfico Brasileiro* (Brazilian Historical-Biographical Dictionary), DHBB for short, that contains almost 12 millions tokens in about three hundred thousand sentences.

The DHBB is a reference work, written by historians and social scientists and published by the Contemporary Brazilian History Research and Documentation Center (CPDOC) of Getulio Vargas Foundation (FGV). It contains almost eight thousand entries with information ranging from the life and career trajectories of individuals to the relationships between the characters and events that the country has hosted. The primary motivation to mine the DHBB came from the need to query the material looking for information that requires almost total reading of the whole body of texts. Examples of such inquiry could be the kinship (or personal) relationship between politicians and their connection to entities such as institutions, movements, events or places throughout their public life. That is, we aim to construct a resource able to answer questions such as "Which politicians were born before the 1960s, had military training and held a position in the Executive Branch?".

We are aware of the vast amount of knowledge spread around the entries in a non-linear way. After all, dictionaries and encyclopedias are made to be consulted and not to be read linearly. In this context, the focus of this paper is to report the first efforts related to the preparation of the material – in particular, the morphosyntactic linguistic annotation and the definition of semantic types for entities and relations relevant to the History domain, taking the appositives as important syntactic relation to observe when annotating semantic relations.

Our main purpose is not only to mine the DHBB, but to create a public corpus to foster Portuguese NLP in general, and NLP in the history domain, in particular. Most large Portuguese annotated corpora are composed of newspaper texts; the DHBB entries, on the other hand, are written in encyclopedic style, and this "novelty" can be a challenge for automatic parsers.

The paper is organized as follows: Section 2. presents the preparation of the corpus relating to the morphosyntactic annotations and the motivation for the comparison exercise performed between two parsers: UDPipe and PALAVRAS. In Section 3. we present the entities types and relations relevant to the History domain. In Section 4. we detail the manual analysis that we conducted of the appositive relations between entities, the evaluation of the outputs generated by the parsers and the revision of entities identification/segmentation of proper names in a sample of the corpus. Finally, concluding remarks are summarized in Section 5..

2. Corpus Preparation

The first edition of the DHBB dates from 1984 in printed version only, and it was in 2010 that its content was fully made available on the Internet[1]. Since its beginning, CPDOC has developed an internal information system to maintain the data through forms and reports that interact with a relational database. The database structure can be summarized as one main table that contained a text field with the entries encoded in HTML and some metadata: basically, the name of the entry and its nature (whether biographical or thematic). This structure showed to be quite limiting when it concerns maintenance and improvements on the dictionary. These issues are described in details in (Paiva et al., 2014) and were the reason for our proposal of maintaining the entries as text files using a lightweight human-readable markup syntax, like YAML (Ben-Kiki and Evans, 2005) and Markdown (Gruber, 2004). A consider-

[1] Available at `http://cpdoc.fgv.br/acervo/dhbb`

able effort was then made to bring up this structure. The decision to adopt plain text files was justified by clear reasons: easiness of maintenance using any text editor (tool independence); conformity to long-term standards by being software and platform independent; easiness to exploit the possibilities of DHBB's files as a resource for NLP; enrichment of the entries with metadata of any kind at any time, even those extracted from natural language processing. The data is freely available at `https://github.com/cpdoc`.

Among the many linguistic metadata that we are adding to DHBB corpus, one important annotation layer is the syntactic analysis.[2] The syntactic analysis is being done according to the Universal Dependencies standards (Nivre et al., 2016). The Universal Dependencies (UD) project,[3] in its ambitious and encompassing mission of affording a single set of tags and parallel analyses common to several different languages, provides a multilingual natural language processing (NLP) framework. The general philosophy is to provide a universal inventory of categories and guidelines to facilitate consistent annotation of similar constructions across languages, while allowing language-specific extensions when necessary. An example of such annotation is given in Figure 1, showing the main grammatical relations involving a verb, an oblique agent and an appositive.

In order to parse DHBB, in this first stage we run both PALAVRAS (Bick, 2000), a rule-based multi-level constraint grammar parser developed specifically for the Portuguese language, and UDPipe (Straka and Straková, 2016), a machine learning pipeline for tokenization, tagging, lemmatization and dependency parsing. UDPipe follows the UD's guidelines, being language-independent, and can be trained given annotated data in CoNLL-U format.[4]

The motivation for the double processing is twofold: first of all, we believe that comparing the outputs of different systems is a way to optimize the linguistic human revision, as suggested in (Truggo, 2016). Additionally, we aim to compare linguistic analysis of both systems in a genre (encyclopedic) unusual to these parsers.

On the whole, the DHHB corpus comprises: automatic morphosyntactic annotations given by the parsers for the whole corpus, manual entity relations annotations for the golden sample, and an entity lexicon built semi-automatically from lexical-syntactical patterns, taking advantage of the highly predictable written style of the DHBB.

3. Entities and Relations

Entity recognition is a crucial task for text mining since its main focus is on instances of general semantic types like person, location, time and organization. Our definition of entity closely follows the ACE (Automatic Content Extraction) proposals (Doddington et al., 2004), capturing all kinds of information that can identify something or someone relevant, whether it's a proper name or not. In an entry about *Revolução de 1930* (Revolution of 1930), for instance, we intend to recover data about this specific event even when it is referred as *revolução* (revolution) as in "Essa carta pode ajudar no esclarecimento de um ponto importante das articulações da *revolução*, pois a bibliografia sobre o período refere-se a dois encontros entre Vargas e Prestes" (This letter can help clarify an important point of the articulations of the *revolution*, since the bibliography on the period refers to two meetings between Vargas and Prestes).[5]

To elicit the semantic types relevant for the history domain, we combined knowledge from domain experts and a corpus driven approach based on a wide reading of entries, aimed at validating and increasing the initial proposed classes. As a result, we conceived seven classes, presented in Table 1.

Classes	Examples
`PER` (person)	*Getúlio Vargas, Lula, presidente*
`ORG` (organization)	*Petrobras, Partido Democrático Social, PDS*
`POL` (political formulation)	*Plano Collor, Programa de Estabilização Monetária, AI-5*
`EVN` (event)	*Revolução de 1930, Atentado do Riocentro*
`LOC` (local)	*São Paulo, palácio Guanabara*
`DOC` (document)	*Diário pessoal de Getúlio Vargas*
`TME` (time)	*Janeiro de 2001, 1927 a 1929*

Table 1: Entity classes for DHBB

Inspired by the set of relations proposed by the Second HAREM task (Freitas et al., 2008) we devised our own set of relations to connect the entities. During the process of text analysis, in particular looking at appositives occurrences, a few other relations were identified as relevant to our goals. Table 2 presents the final list of relations and examples.

4. DHBB: Hands-On

For the work presented here, whose primary purpose is to offer some insights to the text mining in the history domain, we selected a sample of 35 dictionary entries containing 38,554 tokens in 1,115 sentences. To convert our sample in a golden set, we conducted a manual revision of the appositive relations between entities: (i) revising the segmentation of the entities names; and (ii) manually identifying the induced semantic relationship between the entities.

In the following, we detail each of these steps. Along the process, we also analyzed (i) the quality of the automatic parsing as to appositive structures; and (ii) the impact of named entities domain lexicon in proper names segmentation. Table 3 presents some features of the sample.

[2]The DHBB files with linguistic metadata are available in `https://github.com/cpdoc/dhbb-nlp`.

[3]http://universaldependencies.org

[4]http://universaldependencies.org/format.html

[5]We are yet to address the co-reference resolutions.

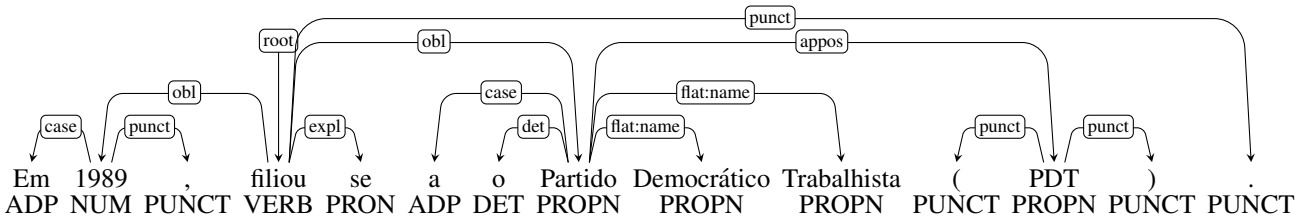


Figure 1: *In 1989, [he] affiliated with the Democratic Labour Party (PDT).*

Relations	E1	E2
`ident` (coreference)	*Partido dos Trabalhadores*	*PT*
`role`	*Alberto Coelho*	*president*
`loc` (local)	*port of Alcantara*	*in Lisbon*
`part`	*Porto Seguro*	*BA*
`date`	*promulgation of Nova Carta*	*18/9/1946*
`link-inst` (institutional relation)	*Vandilson Costa*	from Partido Comunista do Brasil
`link-fam` (family relation)	*Nilo Augusto*	*son of Gercino Coelho and Eunice Coelho.*
`link-pers` (personal relation)	*Orígenes Lessa*	*friend of his brother Fúlvio*
`attrib` (attribute)	*João Abdalla and Amélia Abdalla*	*of Arab origin*
`participant`	*Getulio Vargas*	*in the Revolution of 1930*
`context` (happens)	*XXXVIII ministerial meeting*	*of General Agreement on Tariff and Trade*

Table 2: Relations between entities

Freq	information
38,554	tokens
1,115	sentences
472	sentences with at least one appositive
796	appositives
10	types of semantic relations

Table 3: Details of the revised sample of 35 DHBB entries

Finally, we should note that we have not revised all syntactic annotations in the sample. We have focused our attention only on the names segmentation and the appositives relations. It is an ongoing work the release of a completely revised syntactic analysis of the corpus.

4.1. Appositives

Appositives are syntactic relations especially productive for text mining, with contributions to the building of semantic lexicons, noun phrase co-reference resolution and information extraction from texts (Freitas et al., 2006). They provide descriptive information about the head noun, thus enriching its characterization: when a given noun is tagged as an appositive, a relationship with another term is derived. Appositive relations induce many different semantic relations between entities. In Table 2, the examples of relations `ident`, `role` and `link-fam` all appear on the text as appositives relations between the entities.

Using the output of UDPipe, the revision process was steered in two steps. First, we revised the entities segmentation/identification. Then, we used the PALAVRAS output to check for any missed or incorrect appositive annotation from UDPipe. Along these steps, we annotated the semantic relations between entities expressed in appositive constructions within our golden sample.

The explicit semantic nature of appositives led us to a semantic strategy for revision the parser analysis. That is, we extracted from the parsed sentences the triples formed by the `appos` relation – the linearization of the noun phrases that have their heads connected by an `appos` relation. The extracted triples can be trivially analyzed by humans and if abnormal noun phrases appear it indicates a possible parser mistake.

From table 3, we know that 796 appositive relations were found in 472 sentences. Considering that the UD schema can provide up to 35 possible syntactic relations, the frequency of appositives compared to the other relations can be a clue to depict other linguistic analyses within the corpus. For instance, we observe that core arguments such as (`obl`, `obj`) are far more frequent than subjects (`nsubj`), the reason lies on the style of the narrative that privileges the use of *implicit subjects*, a construction that does not exist in the English language but that is very common in Portuguese written texts (e.g, Figure 1). Table 4 shows the distribution of the first fifteen syntactic relations in the golden sample.

Ten different types of semantic relations from our tagset were identified among the 796 appositive occurrences. Table 5 presents the distribution of semantic relations assigned by the human reviewer for each appositive relation on the golden sample.

Not surprisingly, the most common types of semantic relationship that the appositive constructions reveal are those of *role* and *ident*. On the other hand, personal relations such as friendship or fellowship are almost never made explicit

Freq	information
7,710	case
5,599	det
4,755	punct
3,967	nmod
2,693	obl
1,905	flat:name
1,418	amod
1,115	root
1,107	obj
1,013	conj
899	nsubj
804	cc
796	**appos**
752	advmod
544	compound

Table 4: Distribution of the 15 most frequent syntactic relations occurring in the sample

Num	semantic relation	%
300	role	37.7
200	ident	25.1
73	attrib	9.2
73	date	9.2
65	link-fam	8.2
62	part	7.8
11	link-inst	1.4
6	loc	0.8
5	other	0.6
1	link-pers	0.1

Table 5: Frequency distribution of types of relations in the revised sample

in DHBB, at least not through appositive constructions.

4.2. Evaluation of systems performance

The PALAVRAS system recognized 797 cases of appositives and UDPipe 954 cases.[6] After manual revision, our sample contains 796 occurrences. Although these numbers may suggest that PALAVRAS achieved a better score than UDPipe, these numbers taken globally do not reveal the effective quantity of mistakes that were corrected. Below we elaborate on the comparison of the golden set (the revised sample) with the UDPipe output. Unfortunately, since PALAVRAS analysis follows an entirely different tagset and directives, we are not able to make a detailed comparison of both systems. However, during the revision underlying the construction of the golden sample, we observed that PALAVRAS also produced many incorrect analyses.

When comparing UDPipe's output with the revised sample, we distinguished the following cases:

AllCorrect correct identification of the arguments of the relation and correct identification of appositive. See Figure 1.

ErrDepRel correct identification of the arguments of the relation but incorrect identification of appositive. Figure 2a.[7]

ErrHead incorrect identification of the arguments of the relation but correct identification of appositive, Figure 2b.

FullErr incorrect identification of argument and relation, Figure 2c.

MissingAppos an appositive relation was not detected, Figure 2d.

Table 6 presents the results of the qualitative analysis of UDPipe performance on appositive structures.

Num	Errors/success	%
492	AllCorrect	53.1
9	ErrDepRel	1
175	ErrHead	18.9
203	ErrNotAppos	21.9
47	ErrMissingAppos	5.1

Table 6: Frequency distribution of UDPipe's errors concerning appositive relations

Appositives are tricky linguistic structures to be parsed automatically, since its main formal clue, punctuation, can be easily confused with coordination. For example, in the sentence

> Entre 1959 e 1960, coordenou o setor financeiro da campanha eleitoral do marechal Henrique Teixeira Lott, candidato à presidência da República apoiado pelo PSD e o PTB. (*Between 1959 and 1960, he coordinated the financial sector of the election campaign of Marshal Henrique Teixeira Lott, candidate for the presidency of the Republic supported by the PSD and PTB*)

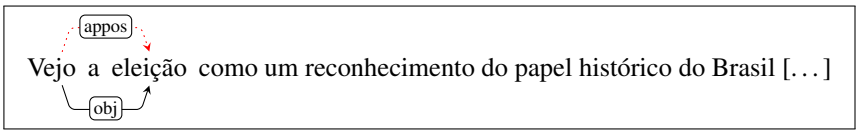
UDPipe (erroneously) analyzed "candidato" in coordination with "setor". Also, in the sentence

> ...votou a favor da emenda constitucional que previa a reeleição de presidente da República, governadores e prefeitos, ...(*voted in favor of the constitutional amendment that foresees the reelection of the president of the Republic, governors and mayors,*)

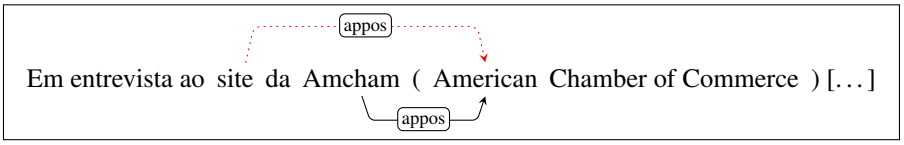
both PALAVRAS and UDPipe were mistaken in identifying an appositive structure between "governadores" and "reeleição" when it is a clear case of coordination.

[6] PALAVRAS uses two tags to indicate the general idea of appositives, we have considered both tags.

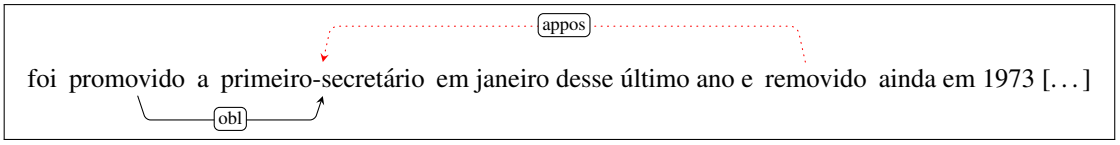
[7] For the rest of this paper, edges above a sentence in red dotted lines represent incorrect analyses, while edges below represent the correct analysis.



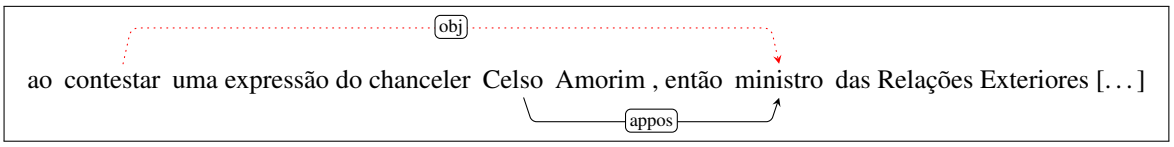
(a) *[I] see the election as a recognition of Brazil's historic role*



(b) *In an interview to Amcham's (American Chamber of Commerce) website*



(c) *[he] was promoted to first secretary in January of the previous year, and removed in 1973*



(d) *when challenging chancellor Celso Amorim's expression, then Foreign Affairs minister*

4.3. Proper nouns identification and segmentation evaluation

Prior to semantic classification of the named entities (NE) we need to correctly identify them. By the highly idiosyncratic nature of proper nouns, errors resulting from wrong segmentation are usual. As an example, PALAVRAS and UDPipe considered *Ministério das Minas e Energia* (Ministry of Mines and Energy) as two separated names: *Ministério das Minas* and *Energia*. Regarding person names, UDPipe split the last name of the person *José Afonso de Melo*, as a noun modifier of the first, and not as part of the whole name. In this case, PALAVRAS did it right, joining the tokens in a single unit.

Each parser has its own way of performing the proper noun segmentation and we tried to reduce the errors by creating domain lexicons from external resources. For the lexicon of *person names*, for instance, we have used both DHBB metadata and a list provided by the personal archives system from CPDOC (Rademaker et al., 2015), being possible to gather 18,171 names. As to the *organizations*, we have almost entirely used pattern recognition in the corpus to extract the names: with AntConc (Anthony, 2016) we searched for simple patterns like *presidir o [A-Z]* (to chair the [capital letter]) or *estudar em [A-Z]* (to study at [capital letter]). This process lead to a lexicon of 3,637 entities.

In the entire corpus we found 83,898 person names (7,514 of them unique) that exist in the lexicon being mentioned on the text, which represents 42% of the whole list. Concerning the organizations we found 69,775 names (3,029 of them unique) occurring in the corpus, which represents 83% of the lexicon. The reason why we have a higher number of organization names matches is due the approach used to construct the list, as explained above, which make use of lexical patterns and concordance lines to extract the names from the corpus.

Concerning our golden subset we found 430 persons (219 of them distinct) persons mentioned on the text. As to the organization lexicon, we found 360 organizations (116 of them distinct) organizations occurring in the corpus.

We know that the use of lexicons has limitations such as limited coverage and variation in the writing of names, i.e. the same person can be mentioned in different ways ranging from the complete full name to the nickname. On the other hand, we believe that the incorporation of lexical entries, associated with semantic classes, are a simple and effective method to bootstrap the creation of lexical-syntactic patterns, crucial for semantic annotation between entities.

Some studies demonstrate positive results when adopting similar approach of using lexicons. In (Florian et al., 2003), the authors investigated the combination of a set of diverse statistical named entity classifiers applied to an English corpus: when no lexicons (gazetteers) or other additional training resources are used, the combined system attains a performance of 91.6 F1 on the English development data; but after integrating gazetteers containing some 50 thousand names of cities, 80 thousand proper names and 3,5 thousand organizations, the F-measure error was reduced by a factor of 15 to 21%.

We then evaluated the impact of using the lexicon for automatic post-processing the UDPipe before comparing it with the golden sample.

Entity name recognition was done in a greedy way. If both "José Machado Coelho de Castro" and "José Machado" were in the lexicon, and the former were in a sentence, only the former would be recognized. In fact, even if "Machado Coelho" or "Castro Abreu" were in the lexicon and the phrase were "José Machado Coelho de Castro Abreu nasceu em 1931", only "José Machado de Castro" would be recognized, provided "José Machado Coelho de Castro Abreu" is not in the lexicon.

Thus, we have the following: 790 mentions of proper

names from the lexicons were found in the golden set sentences, with the most frequent name occurring 53 times. Although we have made corrections in 460 tokens, only thirty of the affected tokens had been marked with an `appos` relation.

In 18 of these cases, the wrong segmentation of the name had caused an error in the syntactic dependence of the appositive token, and this has been fixed with the incorporation of the lexicon, see the example of Figure 3.

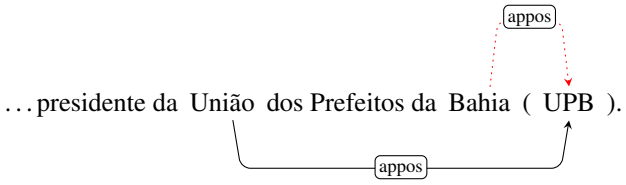


Figure 3: president of the Mayor's Union of Bahia State (UPB)

And in the remaining 12 cases, one of the name's token had been erroneously marked as having an appositive relation with the first token. This was also been fixed with the lexicon, suggesting a relevant role of the lexicon for the syntactic analysis as shown in the example of Figure 4.

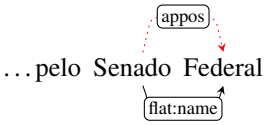


Figure 4: by the Federal Senate

5. Concluding Remarks

In this paper, we present the first efforts towards the creation of an annotated corpus for the history domain. Motivated by the historians' need to interrogate a vast text material, our approach privileges linguistic analysis, as opposed to techniques such as topic modeling, which we believe to be complementary.

In this context, a crucial step is to prepare the material that will be mined. In our case, the preparation includes the annotation of morphosyntax, entities and semantic relations. Although the morphosyntactic annotation is already being performed automatically, the results are not reliable, at least in relation to appositives, as we understand from our analyses in Table 6.

Although highly informative for text mining, appositives seem to be quite difficult structures for systems. On the other hand, it is worth remembering that UDPipe was trained on Bosque-UD (Rademaker et al., 2017), a corpus of a different genre (newspaper), not too big (227,842 tokens), and, ironically, built upon manual revision of PALAVRAS analyses. From a linguistic point of view, the apposition is a syntactic relation only apparently simple (and this point is signalled in the UD guidelines dedicated to *appos*), and, to a Brazilian grammarian, it is "an obscure object" (Perini, 1996).

As to proper names and the lexicons, to compile a comprehensive list of names, we faced difficulties that are particular to the corpus and to the Brazilian Portuguese. The first challenge is related to the DHBB guidelines and has to do with normalization of person names. Since the first version of DHBB, the editors have tried to standardize the different types of information included in the dictionary. For this, they developed general writing guidelines that state how the information should be written, the preferred order of stating facts, and so on. For instance, there are rules for writing names of people, institutions, political parties, social movements, treaties, historical episodes and places. Some of these rules aimed at facilitating information retrieval in the earlier printed versions of the DHBB or at making the dictionary accessible to the general public. For example, the spelling of proper names follows some general orthography principles of that time: the letters 'Y' and 'W' are replaced by 'I' and 'V' ('Darcy' becomes 'Darci', 'Oswaldo' becomes 'Osvaldo'), in some cases 'Z' becomes 'S' (then 'Souza' becomes 'Sousa' and 'Menezes' becomes 'Meneses'). Such rules may appear unusual and dispensable in modern times when data is digitized and expected to be retrieved by search engines capable of answering more advanced requests with wildcard, range, and fuzzy queries. In later versions these normalization rules were dropped and therefore entity names across entries might be inconsistent. Similar to this issue is the Brazilian orthographic reform that took place in 2009. Some of the changes made include extinguishing the use of some hyphens and accents, like "infra-Estrutura" (*infrastructure*) that has become "infraestrutura" and "assembléia" (*assembly*), now "assembleia". All these variations must be in the lexicons in order to improve the parser processing and semantic classification.

In addition, there are some domain's particularities, like entity types such as "Policy Formulation", that would hardly be included in general-purpose NE systems. Another important issue that we have glimpsed but did not focus on in this study has to do with co-reference resolution. Cases like Figure 1 illustrate the non-explicitness of the subjects that is very common in the sentences of the DHBB. The syntactic structure analyses in conjunction with clues like the biographee's name given in the first sentence, can appear in the strategies to be adopted. There is a long way to go.

6. Bibliographical References

Anthony, L. (2016). Antconc version 4.4.1, computer software.

Ben-Kiki, O. and Evans, C. (2005). Yaml ain't markup language (yaml™) version 1.1.

Bick, E. (2000). *The Parsing System PALAVRAS: Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Aarhus University Press.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., and Weischedel, R. M. (2004). The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, pages 837–840.

Florian, R., Ittycheriah, A., Jing, H., and Zhang, T. (2003). Named entity recognition through classifier combination. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume*

4, CONLL '03, pages 168–171, Stroudsburg, PA, USA. Association for Computational Linguistics.

Freitas, C., Duarte, J. C., Santos, C. N., Milidiú, R. L., Rentería, R. P., and Quental, V. (2006). A machine learning approach to the identification of appositives. In *Advances in Artificial Intelligence-IBERAMIA-SBIA 2006*, pages 309–318. Springer.

Freitas, C., Santos, D., Gonçalo Oliveira, H., Carvalho, P., and Mota, C. (2008). Relações semânticas do rerelem: além das entidades no segundo harem. *Linguateca*, pages 77–96, Jan.

Gruber, J. (2004). Markdown language.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Paiva, V. D., Oliveira, D., Higuchi, S., Rademaker, A., and Melo, G. D. (2014). Exploratory information extraction from a historical dictionary. In *IEEE 10th International Conference on e-Science (e-Science)*, volume 2, pages 11–18. IEEE.

Perini, M. A. (1996). *Gramática descritiva do português*. Editora Ática.

Rademaker, A., Oliveira, D. A. B., de Paiva, V., Higuchi, S., e Sá, A. M., and Alvim, M. (2015). A linked open data architecture for the historical archives of the getulio vargas foundation. *International Journal on Digital Libraries*, 15(2-4):153–167.

Rademaker, A., Chalub, F., Real, L., Freitas, C., Bick, E., and de Paiva, V. (2017). Universal dependencies for portuguese. In *Proceedings of the International Conference on Dependency Linguistics*, Pisa, Italy.

Straka, M. and Straková, J. (2016). UDPipe. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Truggo, L. F. (2016). Classes de palavras - da grécia antiga ao google: um estudo motivado pela conversão de tagsets. Master's thesis, Programa de Pós-Graduação em Estudos da Linguagem, Departamento de Letras PUC-Rio.

Building Evaluation Datasets for Cultural Microblog Retrieval

Lorraine Goeuriot, Josiane Mothe, Philippe Mulhem, Eric SanJuan
LIG, IRIT, LIA
Univ. Grenoble Alpes, CNRS, Grenoble INP, France
Université de Toulouse, France
Université d'Avignon, France
{lorraine.goeuriot,philippe.mulhem}@univ-grenoble-alpes.fr
josiane.mothe@irit.fr, eric.sanjuan@univ-avignon.fr

Abstract

ECLEF Microblog Cultural Contextualization is an evaluation challenge aiming at providing the research community with datasets to gather, organize and deliver relevant social data related to events generating large number of microblogs and web documents. The evaluation challenges runs every year since 2016. We describe in this paper the resources built for the challenge, that can be used outside of the context of the challenge.

Keywords: microblog search, social network moining, cultural data, evaluation, informativeness

1. Introduction

Many statistic studies have shown the importance of social media; They seem to be now the main Internet activity for Americans, even when compared to email[1], and most of the social media. Chinese users spend an average of almost 90 minutes per day on social networks[2]. Social media is thus a key media for any company or organization, specifically in Business Intelligence related activities. Companies use social data to gather insights on customer satisfaction, but can also relate this data to forecast product or services revenues (Rui and Whinston, 2011) or measure and optimize their marketing. On the other hand, there are several levers that make social media such popular. In the context of Twitter, Liu *et al.* mention content gratification ("content of the information carried through Twitter") and technology gratification ("easy to use") as the main gratifications influencing users'intention to continue to use Twitter; other gratifications being process ("searching for something or to pass time") and social ("interactivity with other parties through media") gratifications (Liu et al., 2010).
With regards to events such as festivals, social media is now widely used, and gathers various communities at cultural events: organizers, media, attendees, general public not attending the event. These communities are generally interested in different aspects of the generated information:

- the organizers: social media is a nice way to promote an event because it is community-driven. Social media is also useful during the event to get feedback from attendees and because it allows short and timely updates. After the event, data analytics on the discussion is also a useful feedback ;
- the media: other media make use of the content put by organizers and attendees to report the event , as well as to inform the public;
- the public attending a festival: social media is a mean to get information on the event, and communicate with other attendees on the vent it-self or related topics;
- the public not attending a festival: to get attendees and media feedback about the event using social media .

Social media is becoming a core component of communication for any event, either professional or cultural.
Mining and organizing the information surrounding a cultural event can help broadening the perception and visualization of its social impact. In particular, microblogs are increasingly used in cultural events like festivals. For instance, more than 10 million tweets containing the keyword *festival* were sent and shared over the summer 2015. On one hand this massive social activity can transform a local cultural event into an international buzz feed. On the other hand, major festivals that do not follow the social mainstream could fail in attracting and renewing the public. Several national public scientific programs at the crossfield of computer science and humanities aim at studying this phenomena, and its impact on the tourism industry as well as its impact on major national public institutions and society.
We present in this paper the corpus compiled for the CLEF Cultural Microblog Contextualization challenge. This corpus has been built to study the social media sphere surrounding a cultural event, and contains microblogs, a knowledge source, as well as all the web pages linked from the microblogs. We introduce use cases in Section 3. and describe the datasets in Section 4.

2. State of the Art

There are two major trends when building data sets. In the one hand, reference collections are build in order to make it possible for researchers from all over the world to confront their methods and algorithms on common data. This view is the one the main conferences and programs for evaluation

[1] http://www.socialmediatoday.com/content/17-statistics-show-social-media-future-customer-service, http://www.businessinsider.com/social-media-engagement-statistics-2013-12?IR=T

[2] http://www.setupablogtoday.com/chinese-social-media-statistics/

follow. On the other hand, specific and owner-based collections are developed to answer or evaluate a very specific task or are based upon data with specific ownership.

In this paper, we do not intend to provide an exhaustive overview of the various datasets evaluation programs or researchers released. Rather, we focus on collections related either to microblog or events. Moreover, since usually collections are built in order to fit specific tasks, we focus on the ones related to IR tasks.

TREC (Text REtrieval Conference) is one of the major conference for IR evaluation. It runs every year since 1992 a set of tasks that aim at studying specific IR tasks.

TREC Contextual Suggestion track [3] is probably the collection most related to ours. This track aims at providing users with recommendations of "interesting places and activities based on the user's location, personal preferences, past history, and environmental factors such as weather and time" (Dean-Hall et al., 2015). In this collection, there is a set of attractions which consist in information on the attraction it-self and on its context (city, URL and title). This track started in 2012. The 2016 collection can be download at `http://145.100.59.205:8095/TREC2016_CS_Collection.zip`.

With regard to Twitter, while many papers refer to datasets that are built using the social network API, a few datasets have been developed and released for different tasks.

With regard to location extraction there are two public collections: the Ritter collection (Strauss et al., 2016) and the MSM2013 collection (Rowe et al., 2015), both of which are reference collections in the domain. The first collection was initially used by Ritter *et al.* (Strauss et al., 2016) while the second was the training dataset from Making Sense of Microposts 2013 (MSM2013). These two datasets are provided along with manual annotations on locations. Table 1 shows the number of tweets and their distribution (according to whether they mention a location or not) in both datasets. These data sets have been used for example in (Rowe et al., 2015; Ngoc and Mothe, 2018).

In TREC, the Microblog ran from 2011 until 2015 and targets an IR-oriented task. In 2011, this task addresses real-time adhoc search over 16M tweets (Soboroff et al., 2012). In 2015, a real time filtering task was introduced.

In NTCIR[4], the task is as follows: given a microblog post, retrieve or generate a coherent and useful response. Two Asian languages are considered: Chinese with post-comment pairs from Weibo, and Japanese on Twitter (Shang et al., 2016).

IN CLEF, the Tweet Contextualization task was first introduced as part of the INEX QA task (SanJuan et al., 2010) and became a full task the year after. The Tweet Contextualization task mainly focuses on helping a reader to understand a tweet by providing him a short summary of what the tweet is about. The organizers provided sets of tweets each year and the associated Wikipedia dump which was used for building the summaries. Bellot *et al.* provides an overview of the task and lessons learn (Bellot et al., 2016).

3. Use Case Scenario

The goal of the proposed corpus to help developing processing methods to mine the social media sphere surrounding cultural events such as festivals according to several points of views. Tweets linked to an event make a dense, rich but very noisy corpus. As described in (Gimpel et al., 2011), informal language, out of the language phrases and symbols, hashtags, hyperlinks, multi-words abbreviations, are all elements that lead to the fact that the information conveyed by a tweet is often imprecise. Additionally, many tweets are strict or near duplicates, leading to the fact that a special effort has to be put on their management during microblogs retrieval. Tweets also support interaction between users, leading to some of then to an interaction role without any topical content. The interest of mining such data is to extract relevant, and informative content, as well as to potentially discover new information.

The corpus provided is centered on festival participants, and therefore the use cases that we focus on are related to the tweets flow related to such cultural events. Tweets may focus on the whole festival, where others may concentrate on one specific show, or even a detail of such show. Typical use cases related to our corpus are:

1. A participant may get microblogs about the cultural event in which he is taking part, but one or several microblogs taken apart often contain implicit information: the ability to provide "contextual" information that is needed to understand the microblogs is then one interesting scenario. Such background information is important when the user is on the festival site and has a low bandwidth or because he does not want to switch between applications on his handphone. In this case, contextualization systems to be experimented have to provide with a short but highly informative summary extracted from Wikipedia that explains the background of one microblog text.

2. A participant in a specific location wants to know what is going on in surrounding events relatively to artists, music or shows that he would like to see. Starting from a list of bookmarks in the Wikipedia app, the participant seeks for a short list of microblogs summarizing the current trends about related cultural events. The idea of this scenario goes then from wikipedia to microblogs, and we are to what can be achieved by a personalized information retrieval system, in which the user profile is the user's Wikipedia app bookmark list. On important point though, is that we are more focusing on microblogs from insiders than outsiders, i.e., from real participants to the cultural event than from comments from people that are commenting the event from outside.

3. According to a given program of a festival (accessible through the official web site of the festival for instance), the organizers or any user may look for all the tweets related to the festival highlight. In this case,

[3] `https://sites.google.com/site/treccontext/trec-2016/trec-2016-contextual-suggestion-guidelines`

[4] NII Testbeds and Community for Information access Research

Table 1: Summary of the Ritter and MSM2013 datasets used to evaluate location extraction models on tweets.

	Ritter's dataset	MSM2013 dataset
# of tweets	2,394	2,815
# of tweets containing a location (TCL)	213 (8.8%)	496 (17.6%)
# of tweets without location (TNL)	2,181	2,319

the official program is used as a source for generating queries. Namely, the program is a list of triplets <title of highlight, date/time, location>, and each triplet, each triplet being a query. For the organizers of a festival, the interest of this scenario is to obtain an overview of the festival according to microblogs. For a attendee of a festival, the interest of this scenario is to get a recall about what he saw (for instance when a guest who joins a band on stage, we do not always know his name: this use case allows to gather such information *a posteriori*).

While our goal is to build datasets that will help research centered on the use cases above, we can foresee new research challenges that could be investigated with our dataset: cultural events are often facing a big data challenge: direct stakeholders (organizers, artists, attendees), as well as indirect ones (media, public) can express themselves about the event, in different ways, media, and even languages. This data can be seen as a virtual sphere surrounding the event itself. Mining and organizing such data could bring very useful information on the events and their content. Besides the use cases given above, we believe such a corpus could lead to explore many other challenges in the domain, like the integration of time and localization in microblogs contextualization and retrieval.

4. Datasets

The dataset created for this evaluation lab contains several parts, described in the sections below.

4.1. Microblogs collection

We collected all public microblog from twitter containing the keyword festival from May 2015 to November 2016 using a private archive service with twitter agreement based on streaming API[5]. The average of unique microblogs (i.e. without re-tweets) between June and September is $2,616,008$ per month.

These microblogs are provided in UTF8 csv format with the 13 fields, among them 12 are listed in table 2. The "Comments" row in table 2 gives some figures about the existing corpus.

Because of privacy issues, they cannot be publicly released but can be analyzed inside the organization that purchase these archives and among collaborators under privacy agreement. CLEF Labs[6] provide this opportunity to share this data among academic participants. These archives can be indexed, analyzed and general results acquired from them can be published without restriction.

[5] `https://dev.twitter.com/streaming/public`

[6] https://mc2.talne.eu

4.2. Linked web pages

66% of the collected microblogs contain Twitter *t.co* compressed urls. Sometimes these urls refer to other online services like *adf.ly, cur.lv, dlvr.it*, *ow.ly*, *thenews.uni.me* and *twrr.co.vu* that hide the real url.

4.3. Wikipedia Crawl

Unlike tweets and web pages, wikipedia is under Creative Common license, and its contents can be used to contextualize tweets or to build complex queries referring to wikipedia entities. Using the tools from INEX tweet conceptualization track[7] we extracted from wikipedia an average of 10 million XML documents per year since 2012 in the four main twitter languages: en, es, fr and pt. These documents reproduce in an easy to use XML structure the contents of the main wikipedia pages: title, abstract, section and subsections as well as wikipedia internal links. Other contents as images, footnotes and external links are stripped out in other to obtain a corpus easy to process by standard NLP tools. By comparing contents over the years, it is possible to detect long term trends.

4.4. Textual Assessments for Evaluation

Along with this three data sources (Microblogs, related Web and wikipedia Encyclopedia), two types of search queries with related textual references are provided to evaluate systems for Microblog:

- Contextualization based on Wikipedia where given a tweet as query the system has to provide a short summary extracted from the wikipedia that provides all necessary background knowledge to fully understand the tweet.
- Summarization based on tweets where given a topic represented by a set of wikipedia entities, extract a reduce number of tweets that summarizes main trends about that topic in festivals.

System outputs were evaluated based on informativeness as in(SanJuan et al., 2010). Manual runs and Questionnaire data were provided by the French ANR GAFES project.

5. Results obtained with the dataset - Challenge participation

Participation to these challenges have been reported in (Goeuriot et al., 2016) and (Ermakova et al., 2017). In our case, we shall focus on general results that show what is achievable over this data.

[7] `http://tc.talne.eu`

Table 2: Fields of the Microblogs collection.

Name	Description	Comments
text	text of the tweet	99% of the tweets contain a non empty text 66% contain an external compressed URL
id	unique id of tweet	total $80,641,580$ tweets.
from_user	author of tweet (string)	$16,128,316$ organizations among $3,577,724$ users.
iso_language_code	encoding of the tweet	133 tags: en (57%), es (15%), fr (6%) and pt (5%).
source	interface used for posting the tweet	frequent tags: twitter Web Client (16%) iPhone and Twitterfeed clients (11% each).
<geo_type, geo_coordinates_0, geo_coordinates_1>	geolocalization	triplet valued in 2.3% of the tweets.
<created_at, time>	date/time of tweet	15.1% of the tweets are created on Sundays, and 13.3% on Thursdays.

The first outcome was that a microblog corpus over a such a long time period allows to represent tokens by temporal series: every token is represented by the vector of its occurrences in the microblogs grouped by week (78 weeks in total). This is like a temporal word embedding and allows to span along similar trends at concomitant periods which in the case of cultural events like festivals is essential. Following (Murtagh, 2016) presented at CLEF 2016 CMC workshop, an interface was set up in order to gain a better understanding of these correlations[8]. Figure 1 shows a hierarchical clustering among cities, music styles and other festival themes. It naturally clusters together Cannes, Hollywood, films, movies and film makers. However, it also reveals some proximity between Deezer, the main French music streaming service and free/trance music festivals. In a similar fashion, there appears to be a correlation between Apple and London due to the past Apple music festival in London (September 2015 and 2016[9]). Although it can also be noted that Spotify doesn't appear to be correlated to any specific festival event over this corpus.

The second main outcome was the use of WikiPedia as an exhaustive multilingual terminological resource over microblogs related to cultural events. Contextualizing microblogs appeared to be more effective than focus retrieval approaches to link microblogs with WikiPedia pages. That is, instead of considering the content as a query, WikiPedia text anchors were matched against it. Furthermore, by using systems like FELTS[10] based on state of the art Hash functions, it was even possible to upgrade this approach and apply it in real time on the stream of microblogs.

Another outcome was the difficulty to identify microblog languages in this corpus without using specialized lexical resources(Hamon et al., 2017). Not only does the term festival appears in microblogs in any known language, but microblogs can refer to several languages at a time, for example they can use a term that related languages (ex: Spanish covering 15% of the corpus and Catalan in only 0.21% of the microblogs) or language variants (ex: Mexican Spanish in 0.01% of the corpus). They can also mix two different languages like Parisian French (6%) and one Arabic dialect (0.09%).

On a further note, when looking at 16,769,807 collected unique urls in microblogs, it appears that less than 1/10 th refer to some accessible public content. Meanwhile, 2/3rds refer to content in private social networks and the rest refer to content that is hidden behind some pay wall. Therefore, it is not possible to automatically crawl the web sphere around these microblogs.

6. Conclusion

We presented in this paper the Cultural Microblog Challenge (MC2) corpus, a temporal comprehensive representation of the virtual sphere surrounding cultural events. This corpus is composed of tweets, web pages linked to by these tweets, and of one knowledge source.

The built corpus has the big interest to provide a snapshot of: a) tweets, and, b) web pages pointed to by the tweets, these pages being downloaded **as soon as the tweet is received**. From a scientific point of view, it will be possible to rerun experiments on the exact same sets of web documents, even years after the event took place. Reproducibility of results is then ensured, unlike with the Bibsonomy test collection (Benz et al., 2010), in which only URLs of web pages are provided, lowering the capacity to really compare systems, as web pages evolve in time. The topics covered by the corpus have several benefits:

- The amount of data in the corpus is manageable by academic research teams (around 20 millions of tweets, several millions of web documents, possibly split into smaller subsets depending on the task expected). This point is important as we expect numerous participants to experiment their ideas on the MC2 corpus;
- Forcing the corpus perimeter to festival cultural events still covers a variety of festivals (cinema, music, theater, ...) that may have different features regarding their related social spheres;
- The cultural domain is usually well documented in resources like Wikipedia, so the MC2 corpus will not

[8] https://mc2.talne.eu/shiny/gafes/ts_c/

[9] https://en.wikipedia.org/wiki/Apple_Music_Festival

[10] https://github.com/jourlin/FELTS

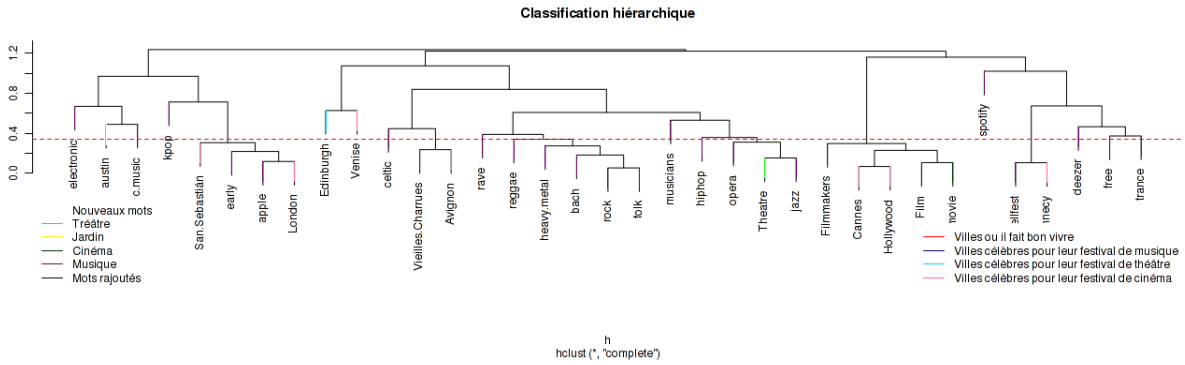


Figure 1: Correlations between temporal series associated with words

suffer from the lack of knowledge that may be used during retrieval.

Without limiting the possible uses of this corpus, we foresee that the concurrent gathering of web pages and tweets may also pave the way to other studies, like co-evolutions of tweets and referred web pages over several occurrences of the same festival, or co-dynamics of topics in web pages and tweets.

7. Bibliographical References

Bellot, P., Moriceau, V., Mothe, J., SanJuan, E., and Tannier, X. (2016). INEX tweet contextualization task: Evaluation, results and lesson learned. *Inf. Process. Manage.*, 52(5):801–819.

Benz, D., Hotho, A., Jäschke, R., Krause, B., Mitzlaff, F., Schmitz, C., and Stumme, G. (2010). The social bookmark and publication management system bibsonomy - A platform for evaluating and demonstrating web 2.0 research. *VLDB J.*, 19(6):849–875.

Dean-Hall, A., Clarke, C. L. A., Kamps, J., Kiseleva, J., and Voorhees, E. M. (2015). Overview of the TREC 2015 contextual suggestion track. In *TREC*, volume Special Publication 500-319. National Institute of Standards and Technology (NIST).

Ermakova, L., Goeuriot, L., Mothe, J., Mulhem, P., Nie, J., and SanJuan, E. (2017). CLEF 2017 microblog cultural contextualization lab overview. In *CLEF*, volume 10456 of *Lecture Notes in Computer Science*, pages 304–314. Springer.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *ACL (Short Papers)*, pages 42–47. The Association for Computer Linguistics.

Goeuriot, L., Mothe, J., Mulhem, P., Murtagh, F., and SanJuan, E. (2016). Overview of the CLEF 2016 cultural micro-blog contextualization workshop. In *CLEF*, volume 9822 of *Lecture Notes in Computer Science*, pages 371–378. Springer.

Hamon, O., Monnin, C., and de Loupy, C. (2017). Syllabs team at CLEF MC2 task 1: Content analysis. In *CLEF (Working Notes)*, volume 1866 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Liu, I. L. B., Cheung, C. M. K., and Lee, M. K. O. (2010). Understanding twitter usage: What drive people continue to tweet. In *PACIS*, page 92. AISeL.

Murtagh, F. (2016). Semantic mapping: Towards contextual and trend analysis of behaviours and practices. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.*, pages 1207–1225.

Ngoc, H. T. B. and Mothe, J. (2018). Location extraction from tweets. *Inf. Process. Manage.*, 54(2):129–144.

Rowe, M., Stankovic, M., and Dadzie, A.-S. (2015). #microposts2015 - 5th workshop on 'making sense of microposts':big things come in small packages. In *WWW (Companion Volume)*, pages 1551–1552. ACM.

Rui, H. and Whinston, A. B. (2011). Designing a social-broadcasting-based business intelligence system. *ACM Trans. Management Inf. Syst.*, 2(4):22:1–22:19.

SanJuan, E., Bellot, P., Moriceau, V., and Tannier, X. (2010). Overview of the INEX 2010 question answering track (qa@inex). In *INEX*, volume 6932 of *Lecture Notes in Computer Science*, pages 269–281. Springer.

Shang, L., Sakai, T., Lu, Z., Li, H., Higashinaka, R., and Miyao, Y. (2016). Overview of the NTCIR-12 short text conversation task. In *NTCIR*. National Institute of Informatics (NII).

Soboroff, I., Ounis, I., Macdonald, C., and Lin, J. J. (2012). Overview of the TREC-2012 microblog track. In *TREC*, volume Special Publication 500-298. National Institute of Standards and Technology (NIST).

Strauss, B., Toma, B., Ritter, A., de Marneffe, M., and Xu, W. (2016). Results of the WNUT16 named entity recognition shared task. In *NUT@COLING*, pages 138–144. The COLING 2016 Organizing Committee.

Training and Adapting Multilingual NMT for Less-resourced and Morphologically Rich Languages

Matīss Rikters, Mārcis Pinnis, Rihards Krišlauks
Tilde
Vienības gatve 75A, Riga, Latvia
{matiss.rikters, marcis.pinnis, rihards.krislauks}@tilde.lv

Abstract

In this paper, we present results of employing multilingual and multi-way neural machine translation approaches for morphologically rich languages, such as Estonian and Russian. We experiment with different NMT architectures that allow achieving state-of-the-art translation quality and compare the multi-way model performance to one-way model performance. We report improvements of up to +3.27 BLEU points over our baseline results, when using a multi-way model trained using the transformer network architecture. We also provide open-source scripts used for shuffling and combining multiple parallel datasets for training of the multilingual systems.

Keywords: neural machine translation, multilingual machine translation, morphologically rich languages

1. Introduction

One of the major advantages of neural machine translation (NMT) is that unlike statistical machine translation (SMT), which was the previous industry standard (and is still actively used in commercial applications), NMT is trained and used jointly as a single end-to-end system without the need to optimize multiple independent models and relations between the models. However, training NMT systems for individual language pairs has shown to take significantly more time (e.g., two to three weeks or up to a week with newer platforms, such as Marian (Junczys-Dowmunt et al., 2016) or Google's Tensor2Tensor toolkit [1]) than training of SMT systems (e.g., less than a day or up to several days for large systems). But even with this advantage, using the traditional approaches, one would still need to train a separate model for each translation direction. Since running a high amount of GPU-intensive NMT models in a production environment can quickly sum up to an enormous resource-usage cost, it has been natural (as shown by related work in Section 2) to look for solutions that allow compressing the models into an even more dense end-to-end solution that is able to handle multiple languages and language pairs simultaneously.

Another benefit of a single model for multiple translation directions could be the ability to learn not just from the training data of the language pair in question, but also from language pairs that include one of the languages. The advantages of learning from multiple translation directions at the same time can be (1) the ability for a model to learn how to translate language specific attributes that are common to multiple languages at the same time, and (2) to learn and generalize translations that may not occur in the parallel corpus of, e.g., A↔B, but do occur in parallel corpora of, e.g., A↔C and C↔B and therefore are deducible.

This work has been driven by the need to identify the best neural network architectures for the development of one-way and multi-way NMT systems for low-resource language pairs that can be applied for low-resource NMT system development (and/or system adaptation) within the project "Forest Industry Communication Technologies".

The structure of this paper is as follows: Section 2 summarizes related work in multilingual and multi-way NMT; Section 3 introduces the setup of our experimental environment and data used; Section 4 outlines the main results in translation quality as well as speed and resource usage, and in Section 5 we look at several examples how translations produced by one-way systems differ from multi-way system translations. Finally, we conclude the paper in Section 6 and introduce plans for future work.

2. Related Work

Multilingual NMT has recently been investigated by several research groups. For instance, Firat et al. (2017) modify the current state-of-the-art attentional NMT approach by supplementing it with the ability to learn from multiple language pairs and multiple translation directions at the same time. They are able achieve this by creating a shared attention mechanism across the involved resources. The authors report improvements in translation quality over most individual baselines, using a single multilingual model trained on five language pairs in both directions. The authors especially highlight that by combining data from language pairs with many resources with data from a low-resource language pair, the quality gains for the low-resource language pair are higher.

Johnson et al. (2016) introduce a simple method for training a single-model multilingual NMT system, which does not require any modifications to the architecture of the system. They achieve this by adding a target language identifying token in the beginning of each source sentence of the training data. While they only report comparable and not outperforming results for models trained on high-resource language pairs, the biggest improvements are achieved in low-resource and even zero-shot translation. An interesting aspect of this approach is that, when trained on many translation directions at once, the same input sentence can be translated into any supported target language by changing only the target language identifying token.

[1] T2T: Tensor2Tensor Transformers - `https://github.com/tensorflow/tensor2tensor`

Ha et al. (2016) use a similar approach to Johnson et al. (2016) by only modifying training data and using the same NMT system architecture. The main difference is that they add a language identifying token to each subword unit and apply this pre-processing to both - source and target sentences of the training data. Another difference is that they don't use particularly deep network architectures in their experiments. The authors describe two experiment scenarios where they train systems to translate from multiple source languages into one target language by (1) adding an additional parallel corpus and (2) adding a monolingual corpus as the additional source and target data. The achieved improvements reach up to 2.6 BLEU points for the first approach and up to 3.15 BLEU points for the second approach.

3. Experiment Setup

In our experiments, we mainly followed the path of Johnson et al. (2016) by not making any modifications to the network architecture and modifying only the data during training and inference. We did, however, experiment with different encoder and decoder cell types and add slight modifications to the data iterator module for it to automatically read the multilingual multi-way training data in equal batches for each translation direction and prepend the target language symbol at the beginning of each source sentence. Our recurrent neural network NMT systems were trained with Nematus (Sennrich et al., 2017) using four main configurations. For training of the NMT systems with convolutional neural networks and transformer networks, we used Sockeye (Hieber et al., 2017). All SMT systems were trained using using the Moses (Koehn et al., 2007) toolkit in the Tilde MT platform (Vasiļjevs et al., 2012). The details of the models are as follows:

- Recurrent neural network models
  - Maximum sentence length of 50;
  - Multiplicative long short-term memory (Krause et al., 2017) (MLSTM) shallow one-way (MLSTM-SU - the baseline model)
    - Encoder and decoder cell type – MLSTM (same as used by Pinnis et al. (2017));
    - A shared subword unit vocabulary (Sennrich et al., 2016) of 25,000 tokens;
  - Gated recurrent units (GRU)
    - Encoder and decoder cell type – GRU;
    - Shallow multilingual multi-way (GRU-SM)
      - 1-layer encoder and 1-layer decoder;
    - Deep - one-way (GRU-DU) and multilingual multi-way (GRU-DM)
      - 4-layer encoder and 4-layer decoder;
      - 2 GRU transition operations applied in the encoder layer; 4 GRU transition operations applied in the decoder layer; 2 GRU transition operations applied in decoder layers after the first layer;
      - Additional incremental training (Freitag and Al-Onaizan, 2016) after convergence of the GRU-DM model, using only parallel training and development data of a single translation direction;
- Fully convolutional neural network models - one-way (FConv-U) and multilingual multi-way (FConv-M)
  - Encoder and decoder cell type - convolutional neural network (CNN);
  - 15-layer encoder and 15-layer decoder;
  - Maximum sentence length of 128;
- Transformer neural network models - one-way (Transformer-U) and multilingual multi-way (Transformer-M)
  - Encoder and decoder cell type - transformer;
  - Maximum sentence length of 128;
  - 6-layer encoder with convolutional embeddings;
  - 6-layer transformer decoder;
  - Each block (self-attention or feed-forward network) is
    - Pre-processed with layer normalization;
    - Post-processed with dropout and a residual connection;
- SMT one-way models (SMT)
  - Word alignment performed using fast-align (Dyer et al., 2013);
  - 7-gram translation models and the 'wbe-msd-bidirectional-fe-allff' reordering models;
  - Language model trained with KenLM (Heafield, 2011);
  - Tuned using the improved MERT (Bertoldi et al., 2009).

Common parameters for all multilingual multi-way experiments:

- Multilingual training data was shuffled in equal batches per translation direction and with the target language identifier added before each sentence as described by Johnson et al. (2016).
- A shared subword unit vocabulary of 50 000 tokens was used.

For all one-way experiments we used a smaller shared subword unit vocabulary of 24 500 tokens.
All other parameters for the models were identical – we clip the gradient norm to 1.0 (Pascanu et al., 2013), use a dropout of 0.2 and trained the models with Adadelta (Zeiler, 2012). We used a word embedding of size of 500, and hidden layers of size 1024. All models were trained until they reached convergence on validation data.

Language pair	Before filtering (Total/Unique)	After filtering (Unique)
En ↔ Et	62.5M / 24.3M	18.9M
En ↔ Ru	60.7M / 39.2M	29.4M
Ru ↔ Et	6.5M / 4.4M	3.5M

Table 1: Training data sentence counts before and after filtering

3.1. Data

For training, we used English↔Russian, English↔Estonian, and Russian↔Estonian data. The one-way models were trained on English↔Estonian and Russian↔Estonian data while the multilingual multi-way models were trained on data from all three language pairs in both directions. The training corpora consist of multiple publicly available and proprietary datasets. Among the public datasets, the largest were the MultiUN (Chen and Eisele, 2012), DGT-TM (Steinberger et al., 2012), Open Subtitles (Tiedemann, 2009), Tilde MODEL (Rozis and Skadiņš, 2017), and Microsoft Translation Memories and UI Strings Glossaries (Microsoft, 2015). The corpora were cleaned and filtered in order to reduce noise in the parallel training data. During filtering, we removed non-parallel sentence pairs, sentences with sentence splitting errors, and duplicate entries. Data processing was performed in two steps – first, a low content overlap filter, which is based on the cross-lingual alignment tool *MPAligner* (Pinnis, 2013), was applied, followed by the standard data processing pipeline of the Tilde MT platform. For some corpora, the filtering resulted in an overall reduction of more than 50% of the original size. Corpora with content overlap below a certain threshold were manually examined and left out from the final dataset. The data filtering procedure is described in greater detail in the paper by Pinnis et al. (2017). An overview of the training data statistics before and after filtering for each language pair is given in Table 1.

For Estonian↔Russian, we selected 2000 random sentences from the training data to be used as validation data. The validation datasets for all other translation directions were obtained from the ACCURAT development datasets (Skadiņa et al., 2012). In the multilingual multi-way model training scenarios, we concatenated $\frac{1}{6}^{\text{th}}$ of each 2000 sentence validation dataset, resulting in batches of 333 sentences from each translation direction, which we used as development data. As for evaluation data – we used the ACCURAT balanced evaluation corpus (Skadiņš et al., 2010) consisting of 512 sentences in each translation direction, for which the Russian version was prepared by in-house translators.

4. Results

In this section, we describe the results of our experiments. We evaluate MT system translation quality using BLEU (Papineni et al., 2002). we also analyse translation speed and GPU memory usage during translation, as well as training duration. While training models for multiple translation directions, we were mainly focused on improving the translation quality when translating between Russian and Estonian, because this specific language pair had the poorest performance among the baseline systems.

4.1. Translation Quality

Table 2 shows how each of the models that we described in the previous section compares to the baseline in terms of development and evaluation data translation quality. When we compare the baseline one-way model (MLSTM-SU) to the other one-way models, the results show that the GRU-DU and FConv-U models reach lower translation quality on all development sets and all but one (for FConv-U) or two (for GRU-DU) evaluation sets. The GRU-DU model insignificantly out-performs the baseline model on the Estonian→Russian evaluation set (by 0.04 BLEU points) and the Estonian→English evaluation set (by 0.08 BLEU points). The FConv-U model shows slightly higher results (by 0.18 BLEU points) on the Estonian→English evaluation set. However, the results of the Transformer-U model are interesting. Although it got lower results on the Estonian↔Russian evaluation sets (by -1.15 and -2.01 BLEU points), it outperformed the baseline model on the Estonian↔Russian evaluation sets (by 2.29 and 3.3 BLEU points). A potential explanation of these results is that the Transformer-U model becomes more advantageous than the MLSTM-SU model when using larger data sets, however, for smaller datasets the MLSTM-SU model is still able to achieve state-of-the-art results.

Next, we look at whether the multi-way models allow increasing translation quality over one-way models. The results show that the GRU multi-way model outperforms the one-way models for all language pairs on all datasets. However, the convolutional and transformer models increase quality only for the low-resource language pairs. The quality improvement for the Estonian↔Russian language pairs ranges from 2.16 BLEU points (for the FConv-M model on the Estonian→Russian evaluation set) up to 5.28 BLEU points (for the Transformer-M model on the Russian→Estonian evaluation set). For the high-resource language pairs, on the other hand, both FConv-M and Transformer-M models show significantly lower translation quality than their respective one-way models. The quality decrease ranges from -2.11 BLEU points (for the Transformer-M model on the Estonian→English evaluation set) down to -5.17 BLEU points (for the FConv-M model on the Estonian→English evaluation set). This shows that the newer NMT architectures in multi-way scenarios are beneficial only to low-resource language pairs.

Finally, if we look at which models achieved the highest overall results on evaluation sets, it is evident that the transformer models performed the best. For the low-resource language pairs, the best results were achieved by the multi-way model. However, for the high-resource language pairs, the best results were achieved by the respective one-way models.

The reason why the results of the SMT system on the development set for Estonian↔Russian (underlined) are so much higher than for all other models may be due to the characteristic of SMT systems being good at memorizing similar sentences to what they have already seen during training.

	Development				Test			
	Ru → Et	Et → Ru	En → Et	Et → En	Ru → Et	Et → Ru	En → Et	Et → En
SMT	27.74	25.48	17.99	25.89	9.88	7.27	21.44	29.69
MLSTM-SU	17.51	18.46	23.79	34.45	11.11	12.32	26.14	36.78
GRU-SM	13.70	13.71	17.95	27.84	10.66	11.17	19.22	27.85
GRU-DU	17.03	17.42	23.53	33.63	10.33	12.36	25.25	36.86
GRU-DM	17.07	17.93	23.37	33.52	13.75	14.57	25.76	36.93
FConv-U	15.24	16.17	21.63	33.84	7.56	8.83	24.87	36.96
FConv-M	14.92	15.80	18.99	30.25	10.65	10.99	21.65	31.79
Transformer-U	17.44	18.90	**25.27**	**37.12**	9.10	11.17	**28.43**	**40.08**
Transformer-M	**18.03**	**19.18**	23.99	35.15	**14.38**	**15.48**	25.56	37.97

Table 2: Translation quality results for all model architectures on development and evaluation data. The best results are in bold.

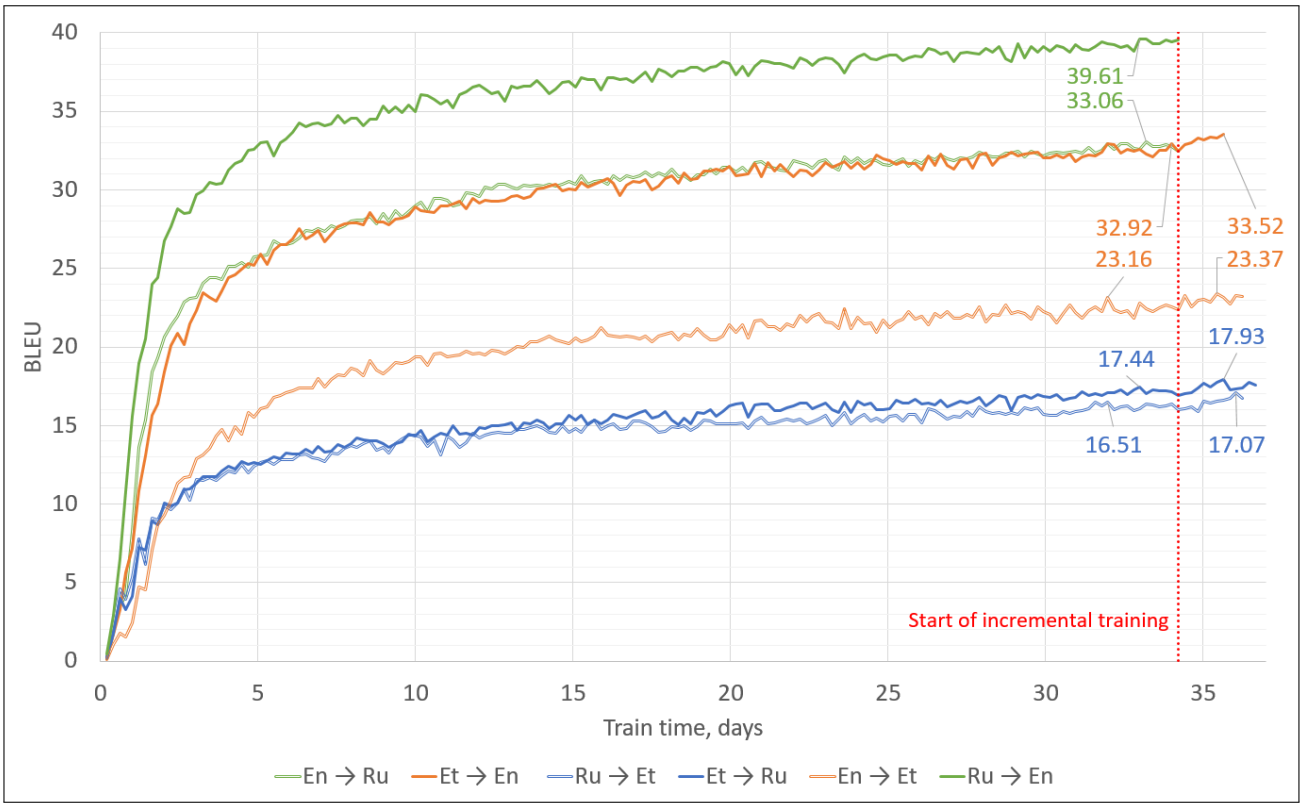


Figure 1: Training progress for the deep multilingual multi-way model (GRU-DM).

As stated in the previous section, this was the only language pair for which the development dataset was derived from the training dataset. For all other language pairs, we used a separate dataset.

When the GRU-DM model had converged, we performed additional incremental training for two language pairs in both ways (English↔Estonian and Russian↔Estonian). Figure 1 illustrates the training progress of this model and the four individual incrementally trained models. The idea of the incremental training was to adapt the system to a specific domain, which in this case would be translation into a single language. Incremental training improved the translation quality of the multi-way GRU-DM model for the individual language pairs by up to 0.60 BLEU points.

Figure 2 shows the training progress for multiple variations of Russian↔Estonian models. The deep one-way models (Estonian↔Russian GRU-DU) reached the early stopping criterion very quickly, but did not get as high as the other models over more time. The other RNN-based models converged after observing approximately 142 million sentences during training. The transformer models stand out the most by being the very first to stop training, as well as reaching the highest BLEU scores the quickest.

4.2. Resource Usage During Translation

Training models with deeper architectures increases resource usage in both – training time and required computational power. The higher resource usage is present during translation as well. Table 3 shows a comparison of time and GPU RAM consumption when translating the evaluation dataset using the NMT systems with several architectures from our experiments. In the table, we isolate models trained with Nematus from models trained with Sockeye, as they are based on different deep learning frameworks, re-

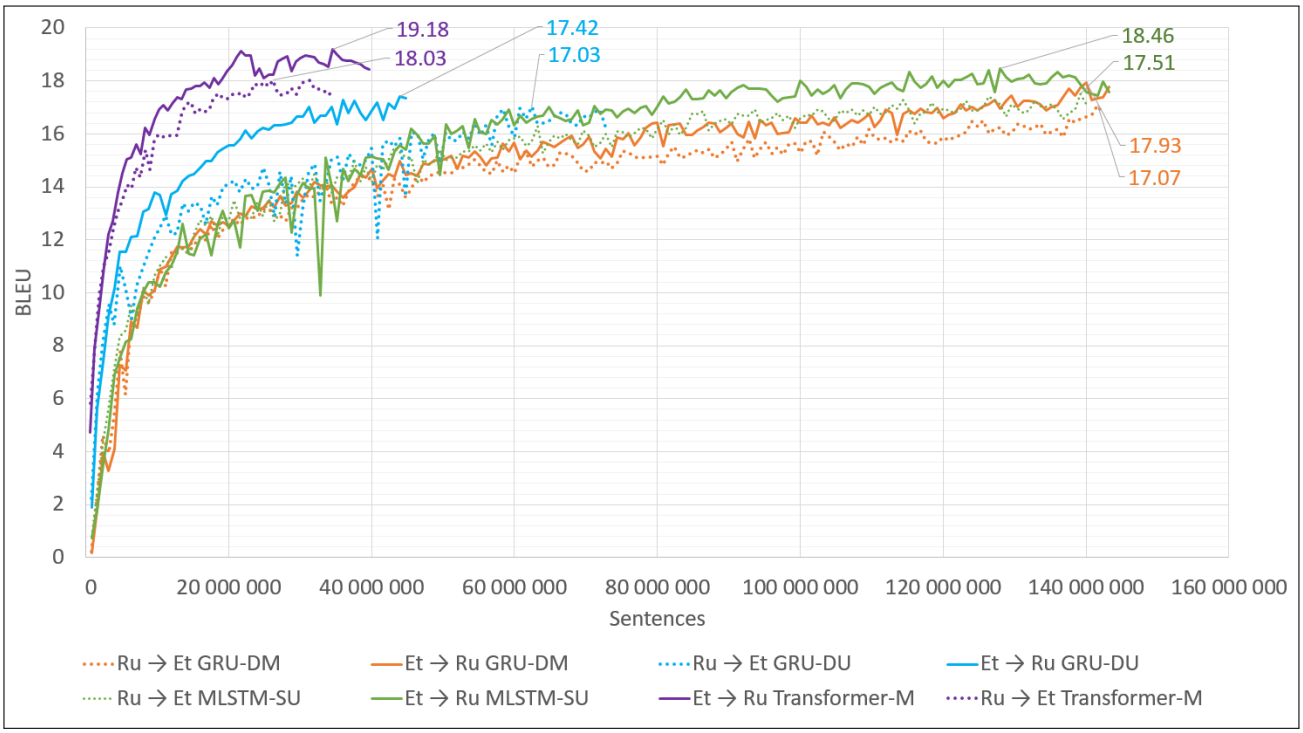


Figure 2: Training progress for Russian↔Estonian systems

	Seconds		Sentences	GPU RAM,	Train time,
	Translation	Per sentence	per second	MB	days
Theano-based Nematus					
MLSTM-SM	274.57	0.54	1.86	651	16.4
GRU-SM	211.51	0.41	2.42	**611**	8.5
GRU-DM	460.07	0.90	1.11	979	36.6
MXNet-based Sockeye					
FConv-M	177.19	**0.35**	**2.89**	971	4.5
Transformer-M	191.05	0.37	2.68	1391	**3.8**

Table 3: Resource usage for all NMT model architectures during translation. The most efficient values are in bold. The final column shows the training time until the system converges.

spectively, Theano (Theano Development Team, 2016) and MXNet (Chen et al., 2015).

The highest-scoring Transformer models are the quickest to train and also nearly the fastest during translation, but they consume more than twice the amount of GPU memory during translation. The GRU-DM model, which was the runner-up model for translating Estonian↔Russian uses 30% less GPU memory during translation, but takes 2.4 times longer to complete the job, and training also took 50% longer. All tests were performed on a machine with an NVIDIA Titan X (Pascal) GPU, Intel Core i7-6850K CPU @ 3.60GHz, 64GB of RAM, and 1TB SSD. We only used a single GPU for training and translating, even though the frameworks have support for multi-GPU training and translation.

It is worth mentioning that while training all shallow RNN models – multi-way or one-way – the training time for a single model to converge did not change noticeably. The same can be said about CNN and Transformer models. In the case of deep RNN models, training time increased by about 2-3 times, reaching 3-4 weeks on a single GPU.

5. Translation Examples

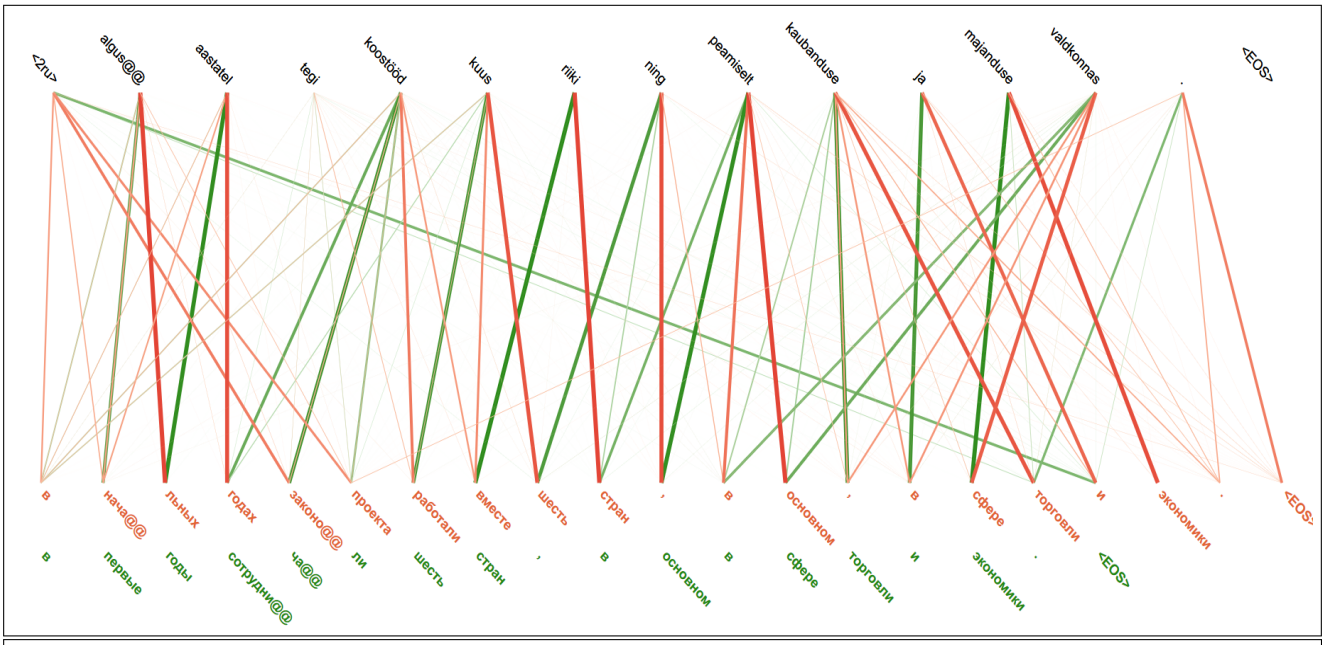
In this section, we show three examples where we compare sentences from one-way and multi-way architectures (e.g. the deep GRU models or transformer models).

In Figure 3, we compare one of the poorest-scoring translations generated with both the overall highest-scoring multi-way system (Transformer-M) and its one-way counterpart. The BLEU score of both translations is identical, but while the translation of Transformer-M is almost perfect (with fluency issues in the last two words), the translation of Transformer-U features a more significant lexical choice mistake. I.e., the words "*kasutab*" (uses) and "*regulaarselt*", which are correctly translated by the multi-way model as "использует" (uses) and "регулярно" (regularly), are mistranslated by the one-way model as "практикуют" (practice) and "работу" (work).

Figure 4 shows a comparison of a sentence that had one of the highest BLEU scores out of all GRU-DU translations compared with the same sentence translated using GRU-DM. There is a redundant word ("законопроекта" - bill project or draft law) in the translation of the one-way

Source:	Üle poole rahvastikust kasutab Internetti regulaarselt.
Transformer-U:	более половины населения практикуют работу с Интернетом.
(transl. into English):	*More than half of the population practice working with the Internet.*
Transformer-M:	более половины населения регулярно использует Интернет.
(transl. into English):	*More than half of the population regularly uses the Internet.*
Reference:	более половины жителей регулярно пользуются интернетом.
English Reference:	More than half the population are regular internet users.

Figure 3: Translation examples comparing the highest-scoring system (multi-way transformer) with its one-way counterpart. BLEU score of both - **15.62**.



Source:	Algusaastatel tegi koostööd kuus riiki ning peamiselt kaubanduse ja majanduse valdkonnas.
GRU-DU:	в начальных годах законопроекта работали вместе шесть стран , в основном , в сфере торговли и экономики.
(transl. into English):	*In the initial years of the bill project, six countries worked together, mainly in the sphere of trade and economy.*
GRU-DM:	в первые годы сотрудничали шесть стран , в основном в сфере торговли и экономики.
(transl. into English):	*In the first years, six countries cooperated, mainly in the sphere of trade and economy.*
Reference:	в первый год сотрудничество вели шесть стран , в основном в сфере торговли и экономики.
English Reference:	In the early years , the cooperation was between six countries and mainly about trade and the economy.

Figure 4: Translation examples comparing the second highest-scoring system (deep multi-way GRU) with its one-way counterpart. BLEU scores - **47.63** (GRU-DU - orange alignments) and **67.04** (GRU-DM - green alignments).

Source:	Charles tõusis ja vaatas aknast välja.
Transformer-U:	Шарль встал и посмотрел в окно.
(transl. into English):	*Charles stood up and looked out the window.*
Transformer-M:	Шарль встал и оглянулся в окно.
(transl. into English):	*Charles stood up and looked out the window.*
Reference:	Чарльз поднялся и посмотрел в окно.
English Reference:	Charles rose and looked out of the window.

Figure 5: Translation examples comparing the highest-scoring system (multi-way transformer) with its one-way counterpart. BLEU scores - **61.48** (Transformer-U) and **26.27** (Transformer-M).

model, which is not present in the source. It is also evident in the attention alignments (visualised using the toolkit by Rikters et al. (2017)) that the sub-word units of this word are strongly aligned only to the target language tag at the beginning of the source sentence. This may mean that these are not translations of any specific sub-word units of the source sentence. The translation of the multi-way model does not exhibit such a problem in this example.

In Figure 5, we show the third example. Here the translation from the one-way transformer model scores higher according to BLEU than the multi-way model. The only difference between these two translations is how the Estonian word "*vaatas*" (looked) is translated. The Transformer-U model produced the translation "посмотрел" (looked), which matches the reference translation, but the Tranformer-M model produced the translation "оглянулся" (looked back), which is the wrong lexical choice in the given context.

6. Conclusion

In this paper, we described a wide range of experiments on training and evaluating multilingual and multi-way neural machine translation systems. Our results show that for low-resource language pairs, such as Estonian↔Russian, we can achieve a significant improvement in translation quality by adding data from other languages over using only one-way parallel data. Multi-way NMT systems in both directions improved translation quality (by 3.09 - 5.28 BLEU points for Russian→Estonian and 2.16 - 4.31 BLEU points for Estonian→Russian) for all three model architectures (deep GRU, convolutional, and transformer), for which we performed multi-way experiments. Our experiments also show that the largest improvements in BLEU scores, as well as the highest overall BLEU scores in the low-resource multi-way scenario were achieved by training systems with the Transformer model.

While the multilingual approach helped gaining improvements for the low-resource language pair, it did degrade the performance for the high-resource language pairs by several BLEU points. In almost all of our experiments the multilingual models showed a drop in translation quality by 2.87 - 3.22 BLEU points for English→Estonian and 2.11 - 5.17 BLEU points for Estonian→English. However, the results showed that the most stable architecture for multi-way model training was the deep GRU model architecture. It showed improvements for both low-resource and high-resource language pairs on both development and evaluation data sets.

The results also showed that when training one-way systems for the low-resource language pairs, the newer convolutional and self-attention (i.e., transformer) models underperformed. The best results in these experiments were achieved by the MLSTM-based models (outperforming the convolutional models by up to 3.55 BLEU points and the transformer model by 2.01 BLEU points).

While manually analysing the evaluation sets, we noticed that there were several sentences translated perfectly by Transformer-M, but much worse by GRU-DM and vice versa. This suggests that further investigation may be required to find out whether a combination of the systems can lead to translations of even higher quality. There are many successful methods for MT system combination that could be utilized, for example, using confusion networks (Peter et al., 2017) to align hypotheses and pick the best parts of each as the final translation. A more neural network specific option for MT system combination by combining outputs according to the attention alignments produced by the neural networks (Rikters and Fishel, 2017) could also be used for this purpose.

Finally, we provide an update to Nematus[2] that allows training of multi-way models by providing multiple parallel corpora as input data. We also release a set of scripts[3] that can be used to prepare a multi-way corpus from multiple parallel corpora for training of multi-way NMT systems with other frameworks.

7. Acknowledgements

In accordance with the contract No. 1.2.1.1/16/A/009 between the "Forest Sector Competence Centre" Ltd. and the Central Finance and Contracting Agency, concluded on 13th of October, 2016, the study is conducted by Tilde Ltd. with support from the European Regional Development Fund (ERDF) within the framework of the project "Forest Sector Competence Centre".

8. Bibliographical References

Bertoldi, N., Haddow, B., and Fouet, J.-B. (2009). Improved Minimum Error Rate Training in Moses. *The Prague Bulletin of Mathematical Linguistics*, 91(1):7—-16.

Chen, Y. and Eisele, A. (2012). MultiUN v2: UN Documents with Multilingual Alignments. In *LREC*, pages 2500–2504.

Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., and Zhang, Z. (2015). MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems. *Neural Information Processing Systems, Workshop on Machine Learning Systems*, pages 1–6.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics.

Firat, O., Cho, K., Sankaran, B., Yarman Vural, F. T., and Bengio, Y. (2017). Multi-way, Multilingual Neural Machine Translation. *Computer Speech and Language*, 45:236–252.

Freitag, M. and Al-Onaizan, Y. (2016). Fast Domain Adaptation for Neural Machine Translation. *arXiv [cs.CL]*.

Ha, T.-L., Niehues, J., and Waibel, A. (2016). Toward Multilingual Neural Machine Translation with Universal Encoder and Decoder. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, page 16.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on*

[2]Multilingual NMT iterator - git.io/vAgfv

[3]Multilingual NMT Corpora Tools - git.io/vAOoJ

Statistical Machine Translation, pages 187–197. Association for Computational Linguistics.

Hieber, F., Domhan, T., Denkowski, M., Vilar, D., Sokolov, A., Clifton, A., and Post, M. (2017). Sockeye: A Toolkit for Neural Machine Translation. *ArXiv e-prints*, dec.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al. (2016). Google's Multilingual Neural Machine Translation System: Enabling Zero-shot Translation. *arXiv preprint arXiv:1611.04558*.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is Neural Machine Translation Ready for Deployment? A Case Study on 30 Translation Directions. In *Proceedings of the 9th International Workshop on Spoken Language Translation (IWSLT)*, Seattle, WA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Krause, B., Lu, L., Murray, I., and Renals, S. (2017). Multiplicative LSTM for sequence modelling. In *5th International Conference on Learning Representations*, page 9, Toulon, France, feb.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. *... of the 40Th Annual Meeting on ...*, pages 311–318.

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 1310–1318.

Peter, J.-T., Ney, H., Bojar, O., Pham, N.-Q., Niehues, J., Waibel, A., Burlot, F., Yvon, F., Pinnis, M., Sics, V., et al. (2017). The QT21 Combined Machine Translation System for English to Latvian. In *Proceedings of the Second Conference on Machine Translation*, pages 348–357.

Pinnis, M., Krišlauks, R., Miks, T., Deksne, D., and Šics, V. (2017). Tilde's Machine Translation Systems for WMT 2017. In *Proceedings of the Second Conference on Machine Translation (WMT 2017), Volume 2: Shared Task Papers*, pages 374–381, Copenhagen, Denmark. Association for Computational Linguistics.

Pinnis, M. (2013). Context Independent Term Mapper for European Languages. In *Proceedings of the 9th International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, Hissar, Bulgaria.

Rikters, M. and Fishel, M. (2017). Confidence Through Attention. In *Proceedings of the 16th Machine Translation Summit (MT Summit 2017)*, Nagoya, Japan.

Rikters, M., Fishel, M., and Bojar, O. (2017). Visualizing Neural Machine Translation Attention and Confidence. volume 109, pages 1–12, Lisbon, Portugal.

Rozis, R. and Skadiņš, R. (2017). Tilde MODEL-Multilingual Open Data for EU Languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017). Nematus: a Toolkit for Neural Machine Translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain, April. Association for Computational Linguistics.

Skadiņa, I., Aker, A., Mastropavlos, N., Su, F., Tufiș, D., Verlic, M., Vasiļjevs, A., Babych, B., Clough, P., Gaizauskas, R., Glaros, N., Paramita, M. L., and Pinnis, M. (2012). Collecting and Using Comparable Corpora for Statistical Machine Translation. In Nicoletta Conference Chair Calzolari, et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 438–445, Istanbul, Turkey. European Language Resources Association (ELRA).

Skadiņš, R., Goba, K., and Šics, V. (2010). Improving SMT for Baltic Languages with Factored Models. In *Frontiers in Artificial Intelligence and Applications*, volume 219, pages 125–132.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlter, P. (2012). DGT-TM: a Freely Available Translation Memory in 22 Languages. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 454–459.

Theano Development Team. (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May.

Tiedemann, J. (2009). News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Recent advances in natural language processing*, volume 5, pages 237–248.

Vasiļjevs, A., Skadiņš, R., and Tiedemann, J. (2012). LetsMT!: A Cloud-Based Platform for Do-It-Yourself Machine Translation. In Min Zhang, editor, *Proceedings of the ACL 2012 System Demonstrations*, number July, pages 43–48, Jeju Island, Korea. Association for Computational Linguistics.

Zeiler, M. D. (2012). ADADELTA: an Adaptive Learning Rate Method. *arXiv preprint arXiv:1212.5701*.

9. Language Resource References

Microsoft. (2015). *Translation and UI Strings Glossaries*.

Cross-lingual Terminology Extraction for Translation Quality Estimation*

Yu Yuan[†] [*], Yuze Gao[‡], Yue Zhang[‡], Serge Sharoff[*]
[†] School of Languages & Cultures, Nanjing University of Information Science & Technology, 210044, China
[‡] Information Systems Technology and Design, Singapore University of Technology and Design, 487372
[*] Centre for Translation Studies, University of Leeds, LS2 9JT, United Kingdom
hittle.yuan@gmail.com, {yuze_gao, yue_zhang}@sutd.edu.sg s.sharoff@leeds.ac.uk

Abstract

We explore ways of identifying terms from monolingual texts and integrate them into investigating the contribution of terminology to translation quality.The researchers proposed a supervised learning method using common statistical measures for termhood and unithood as features to train classifiers for identifying terms in cross-domain and cross-language settings. On its basis, sequences of words from source texts (STs) and target texts (TTs) are aligned naively through a fuzzy matching mechanism for identifying the correctly translated term equivalents in student translations. Correlation analyses further show that normalized term occurrences in translations have weak linear relationship with translation quality in term of usefulness/transfer, terminology/style, idiomatic writing and target mechanics and near- and above-strong relationship with the overall translation quality. This method has demonstrated some reliability in automatically identifying terms in human translations. However, drawbacks in handling low frequency terms and term variations shall be dealt in the future.

Keywords: Bilingual terminology, translation quality, supervised learning, correlation analysis

1. Introduction

Terminology helps translators organize their domain knowledge, and provides them means (usually terms in various lexical units) to express subject knowledge adequately. Translation scholars and practitioners maintain that terminology correctness is associated with the quality of translation (and interpretation) (Hartley et al., 2004; Xu and Sharoff, 2014; Kim et al., 2015; Brunette, 2000; Karoubi, 2016).

The acknowledgement of the contribution of terminology to translation quality is also echoed by the translation industry and users (Secară, 2005; Lommel et al., 2014; Warburton, 2013). Accurately reproducing the content of the original and using appropriate terminology has become the official assessment criteria of some famous in-use translation-error-based evaluation schemes. For instance, the MeLLANGE project (Secară, 2005) defines more than six terminology errors[1], and the Multidimensional Quality Metrics lists terminology as one of the eight major dimensions, which is subdivided into three children issue types (term inconsistency, termbase[2], and terminology domain[3]) (Lommel et al., 2014). From a user's expectation perspective, appropriate terminological use is also viewed as one of the important quality parameters. For the purpose of marketing, companies will localize the manuals that accompany their products. Localization cannot be done at the expense of quality to endanger the customer satisfaction. Their dissatisfaction will lead to more potential damaging losses in revenue. Therefore, speed and quality is what localization services users are looking for (Warburton, 2013).They would expect that all the terms are translated correctly and consistently, and translators will not invent terms randomly wherever source language (SL) terms cannot find an equivalent in target language (TL) without scientific analysis and sufficient documentation. For both sides, adherence to specified terminology is considered a central concern in translation for the delivery of quality-assured translations.

It is clear that finding an equivalent for terms in a translation impacts the overall quality of translation. When assessing a translation, evaluators should consider how well a translator achieves in successfully rendering those terms in the target language. However, this element of translation has not drawn enough attention from researchers in machine translation quality estimation, and in human translation quality assessment, the whole evaluation of the translation of terminology is carried out by human evaluators manually and subjectively, with or without references. Manual compilation of bilingual term lists for each translation evaluation task is an expensive and laborious effort, hence the rarity of an up-to-date, specialized and relatively comprehensive term database for translation quality estimation purpose.

The main contributions of our work include: language and model adaptation by training term classifiers using a corpus in the bio-medical domain and applying the optimal classifiers to cross-domain and cross-language texts; investigating the contribution of terminology to translation quality with empirical evidence; a working pipeline for terminology-focused quality evaluation to extract and exploit terminology information from raw source texts (STs) and target texts (TTs).

2. Related Work

Different from monolingual term extraction, bilingual term extraction (BTE) faces the additional problem of finding translation equivalents in parallel or comparable texts.

This work is done when the first author works as a research fellow at SUTD.

[1]The main terminological errors are incorrect terminology, false cognate, term translated by non-term, inconsistent with glossary, inconsistent within target text (TT), inappropriate collocation, and user-defined errors.

[2]a term is translated is translated with a term nonconforming to the specification.

[3]a term is translated with a term from a different domain.

There are roughly three approaches to bilingual term extraction, depending on what resources are used:

- **Parallel-corpus Based** Various strategies (Gómez Guinovart and Simoes, 2009; Macken et al., 2013) have been advanced for extracting lexical equivalence from parallel corpora. The main fallacy of methods in this approach is that they rely on the morphosynactic analyser of the term extractor that does not recognize all candidate terms and those chunk-based methods, having extended the alignment model with automatically extracted language pair specific rules. As a consequence, this method blurs the distinction between terms and non-terms.

- **Comparable-corpus Based** Bilingual corpora in specialized domains are actually scarce and it is expensive to build high quality parallel texts of specialized domains. A practical solution to this limitation is to make use of comparable corpora (Rocheteau and Daille, 2011; Xu et al., 2015; Hakami and Bollegala, 2017) that are available in large quantities. However, term extraction along this line is often limited to noun phrases (< 5 words) from monolingual comparable corpora. Thus, the recall of such an approach is not satisfactory under some circumstances. For other studies in this approach, ambiguity of term translations and identification of synonymous terms need to be further addressed.

- **Web-data Based** Web data mining is another means to collect terminology pairs (Erdmann et al., 2009; Gaizauskas et al., 2015). Despite the favourable findings from the evaluation process, one of the biggest limitations of the current approach is that the precision still warrants improvement in comparison to other methods that are parallel-corpus based.

To sum up, these systems and pipelines are designed for terminology management or dictionary compilation purpose rather than translation quality evaluation. They cannot readily serve our purpose of finding term pairs from the translated texts to be evaluated. On the one hand, term extraction methods are often tuned towards specific genres or domains (e.g. automobile, agricultural), and on the other hand they often focus on specific types of terms (e.g. MWT or NPs). We aim to evaluate how well terms are translated in students' translations on different topics from various domains. Therefore, a method of automatically identifying terms from both STs and TTs and linking them is needed.
For this purpose, in line with the prediscussed methods, we come up with a solution that uses language-independent features to train a classifier to classify ngrams into terms and non-terms in both STs and TTs, and we present a terminology-focused translation quality evaluation pipeline (See Figure 1). Our approach differs from other Machine Learning (ML) approaches based on linguistic features and context information (Li et al., 2012; Hakami and Bollegala, 2017). Instead, only minimal linguistic processing is used in our approach for data and feature set extraction, such as tokenisation and lemmatisation. The following is a brief description of the features we use to train the term classifiers.

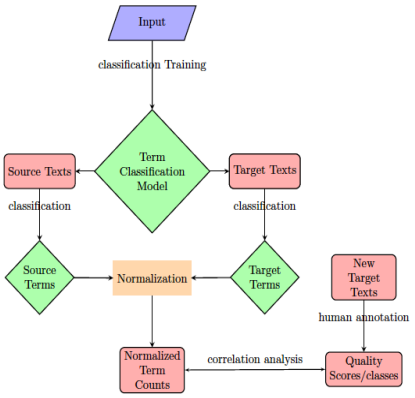


Figure 1: Terminology-focused Translation Quality Evaluation Pipeline

3. Quality Oriented Cross-lingual Term Extraction

To address the issue of cross-lingual term extraction from translational data, we present a supervised learning approach for monolingual term extraction. First, a range of representative and language-independent algorithms are exploited to compute term representations to train different classifiers. Then, monolingual terms identified by the selected, optimal classification model will be used for the normalization process, which normalizes the term counts (i.e. the number of terms 'identified by the classifier') in TTs to be the relative term frequencies in association with the number of 'terms' (as identified by the classifier as well) in STs and the length (i.e. number of tokens) of TT. This normalized term count can serve as a quality indicator in quality estimation tasks (i.e. supervised classification or regression to predict quality scores or class labels) as illustrated in the correlation analysis afterwards.

3.1. Term Classification

N-gram technique is commonly used as a language-independent approach, particularly for under-resourced language. Therefore, the term candidate classification is framed as a N-gram classification task rather than the conventional sequence labelling methods that are commonly seen in previous work (Zhou and Su, 2004; Finkel et al., 2004).
From a pragmatic point of view, our features are computed by JATE 2.0 (Zhang et al., 2016). Most representative and language-independent statistic ATR techniques are available in the package. These features (See Table 1) .
We briefly describe the features below:
TTF, namely Term Total Frequency, is the total frequency of a candidate in the target corpus. This algorithm takes into account frequency information for retrieving words or phrases that are both highly indicative of document content and highly distinctive within a text collection.
ATTF takes the average of TTF by dividing it by the number of documents in which the candidate term occurs.

Feature	Algorithm
TTF	Total Term Freqeuncy
ATTF	Average Total Term frequency
TTF-IDF	TTF with Inverse doccument Freq.
RIDF	Residual IDF
C-Value	C-Value
RAKE	Rapid Keyword Extraction
χ^2	Chi-square
Weirdness	Weirdness
GlossEx	Glossary Extraction
TermEx	Term Extraction

Table 1: Features Used for Term Extraction

TTF-IDF is adapted from the classical Term Frequency - Inverse Document Frequency (TF-IDF), which replaces the local distribution measure with global distribution across whole the corpus. It assigns higher value to words that appear more frequently in fewer number of documents across the whole corpus.

RIDF, known as residual IDF, captures the deviation of the actual IDF score of a candidate from its expected IDF score on a Poisson distribution, of which a real term (or keywords) is assumed to be higher than non-term (or ordinary words).

C-value considers the impact of frequency and length of a candidate term and thus is capable of enhancing the conventional statistics of frequency and becoming sensitive to nested terms, such as the candidate term 'T cell' nested in longer terms 'peripheral blood T cell', 'naive T cell' and 'T cell activation'.

Despite that C-value is initially proposed to extract multi-word terms (MWTs), it demonstrates flexibility to handle shorter and even single-word terms (SWTs).

RAKE, short for Rapid Automatic Keyword Extraction, can evaluate the exclusivity, essentiality and generality of extracted candidates. The measurement is based on three metrics, including word frequency, degree of word (the occurrence of a word in longer candidate MWTs and ratio of degree to frequency.

$\boldsymbol{\chi^2}$ measure is commonly used for testing whether bigram tokens co-occur by chance. JATE 2.0 adapted the measure to work with both SWTs and MWTs. If a term has no co-occurrence information, a zero score is assigned.

Weirdness, or **specificity**, is a type of contrastive ranking technique, which is particularly interesting with regard to identifying low frequent terms.

GlossEx is another hybrid approach which measures the goodness of a term by combining term specificity (i.e. termhood) and term association (i.e. unithood). The former quantifies how much an item is related to a specific domain and the latter describes the degree of association of words the term contains.

TermEx is very similar to GlossEx with extra extension of entropy-related Domain Consensus (DC) metric. DC gives more weights to a term that has even probability distribution across the documents of the domain corpus. Another two components are the Domain Pertinence (DP) and Lexical Cohension (LC), which are essentially the same as Weirdness and TC in GlossEx respectively. The final algorithm is a linear combination of the three metrics with adjustable weights (default to be 1/3 in JATE 2.0).

As mentioned earlier, all these 10 algorithms have been implemented in JATE 2.0, we just need to adapt them for working on Chinese texts.

3.2. Term Count Normalization

The normalization process aims to relate the term counts in the TTs to the terms in the STs so that the consistency of term alignments TTs can be measured and compared across different translations. Our assumption is that a higher relative number of terms counts indicates a more successful translation in terms of term adequacy which in turn contributes to the overall translation quality text-wide. The purpose of this normalization process thus is to obtain a form of term count that is comparable within translations of different lengths from STs containing different number of source terms. In the following experiment, we compute the normalized term count for each translation at the document level. Here is how the normalized term count in TTs is calculated:

$$\boldsymbol{T}_{norm} = \frac{Count_{trg} * Len_{trg}}{C_{[100]*Count_{src}}} \quad (1)$$

where T_{norm} is the normalized term count in proportion to the length of target text (Len_{trg}) in terms of the number of tokens and the number of terms in source text ($Count_{src}$), and $Count_{trg}$ is the count of terms identified in the target text, with $C_{[100]}$ a constant number 100 serving as the text length normalization base, $Count_{src}$ the number of terms in the source text.

4. Experiment

As previously stated, our experiment consists of three parts: training a monolingual term classifier, computing normalized term counts in TTs and applying the normalized term counts to quality estimation. For the last step, we do not report the results of a full quality estimation task but instead analyse the correlation of the normalized term occurrences in translations with their quality scores.

4.1. Training Monolingual Term Classifiers

4.1.1. Corpora

Five corpora, covering 3 different domains and 2 different languages (of varying sizes), are selected in the experiment to train and test our term classifiers. GENIA corpus (Kim et al., 2003) is a collection of biomedical documents and it is the most popular dataset used in ATR.

TTC, short for Terminology Extraction, Translation Tools and Comparable Corpora, a recent European project covering 8 languages, aims on the contribution of various linguistic resource for bilingual term acquisition and translation (Blancafort et al., 2010). Two English-Chinese comparable corpora (i.e. totalling 4 datasets) for two specialized domains in Wind Energy (TTC-W) and Mobile technology (TTC-M) are used in our experiment as test sets. Detailed information of all 6 corpora we used is presented in Table 2.

Corpus	# of documents	Size(tokens)	Reference Term List
GENIA	1,999	420,000	35,800
TTC-W (EN)	172	750,855	188
TTC-M (EN)	37	308,263	143
TTC-W (ZH)	178	4,263,336	204
TTC-M (ZH)	92	2,435,232	150

Table 2: Corpora Used for Training Term Classifiers

N-gram Datasets	# of terms	# of non-terms	# recall
GENIA	4,240	45,350	41%
TTC-W (EN)	120	30,925	76.5%
TTC-M (EN)	149	20,505	98%
TTC-W (ZH)	125	132,407	41.8%
TTC-M (ZH)	168	105,599	57.1%

Table 3: Terms and Non-terms in N-gram Datasets

4.1.2. Dataset Pre-processing

Firstly, all training and testing corpora are tokenised and we restrict our attention to the N-gram candidate terms with a maximum allowable length of 5 ($1 <= n <= 5$) in our current experiment. Next, stop words are removed from the list of n-gram candidates.

In the final step, training datasets and testing datasets are processed by N-gram string matching with ten features output separately by the ten algorithms. The N-gram datasets are further matched with specific Reference Term List (RTL) from each dataset. Any matched N-gram will be labelled as true positive and those having no matches will be viewed as non-terms. By this way, we eventually have 4,240 true terms from GENIA. See Table 3 for the details of our N-gram training and testing sets generated in our experiment.

4.1.3. Term Classification Models

We eventually trained 6 different models on GENIA traing corpus and then have them tested on the four TTC comparable corpora data. For both Chinese and English, we highlighted 3 optimal models each in the coloured, bold font. The classifers with best F1 score are considered as best models in our experiment. As shown in Table 4, on the Chinese test data, the optimal model achieved a precision up to 64% (true term as positive), and on the English test data, we obtained a precision up to 75%. These trained classifiers generally perform better than the Top N precisions of statistic based models (Yuan et al., 2017). Details of performances of all classifiers during the training are provided in Table 4.

4.2. Correlation with Translation Quality Scores

4.2.1. Translation Data

In the following we describe our data. The first dataset consists of 50 trainee translators' translation to a short passage about xenotransplantation (280 words). The second dataset is a course summative work from Shanghai University of International Business and Economics (SUIBE). There are 42 translations for a rotatory closure design patent in the dataset. We choose these two datasets because they are all trainee translations and they contain very domain specific words that are potentially terminology and challenging for trainees. Hereinafter, we refer to them as the XENO data and SUIBE data. The basis statistics of both datasets are shown in Table 5.

As the XENO dataset is part of our quality estimation dataset and it has been annotated by two individual annotators according to the scheme of ATA Certification Programme Rubric for Grading (Version 2011)[4].The performance of a translator is measured against four dimensions ranging from *Usefulness/Transfer* (content transfer), *Terminology/Style* (terminology and lexical equivalence), *Idiomatic Writing* (idiomaticness) to *Target Mechanics* (target language conventions), using a predefined range finder. Four subscores then make up the final score on a percentile scale.

Classifier	Testing Dataset	Precision	Recall	F1
Random Forest	GENIA(held-out)	0.80	0.84	0.8
	TTC-W(EN)	**0.79**	**0.71**	**0.75**
	TTC-M(EN)	**0.77**	**0.74**	**0.75**
	TTC-W(ZH)	**0.58**	**0.69**	**0.63**
	TTC-M(ZH)	0.57	0.60	0.58
LinearSVC	GENIA(held-out)	0.70	0.69	0.70
	TTC-W(EN)	0.66	0.79	0.72
	TTC-M(EN)	0.67	0.76	0.71
	TTC-W(ZH)	0.56	0.51	0.53
	TTC-M(ZH)	0.54	0.56	0.55
SVC RBF	GENIA(held-out)	0.73	0.73	0.73
	TTC-W(EN)	0.69	0.82	0.75
	TTC-M(EN)	**0.70**	**0.82**	**0.75**
	TTC-W(ZH)	0.51	0.53	0.52
	TTC-M(ZH)	**0.59**	**0.65**	**0.62**
MultinomialNB	GENIA(held-out)	0.64	0.59	0.61
	TTC-W(EN)	0.51	0.89	0.65
	TTC-M(EN)	0.53	0.97	0.69
	TTC-W(ZH)	0.74	0.49	0.59
	TTC-M(ZH)	**0.66**	**0.62**	**0.64**
SGD	GENIA(held-out)	0.70	0.69	0.7
	TTC-W(EN)	0.69	0.79	0.74
	TTC-M(EN)	0.67	0.82	0.73
	TTC-W(ZH)	0.60	0.49	0.54
	TTC-M(ZH)	0.58	0.59	0.58
SLR	GENIA(held-out)	0.70	0.70	0.70
	TTC-W(EN)	0.68	0.81	0.74
	TTC-M(EN)	0.70	0.81	0.75
	TTC-W(ZH)	0.58	0.51	0.54
	TTC-M(ZH)	0.59	0.59	0.59

Table 4: Model Performance on Development and Testing Datasets

Dataset	Domain	Passages	# of sentence	Length
XENO	Xenotransplantation	50	14	234 ~ 473
SUIBE	Patent	42	11	297 ~ 376

Table 5: Basic Statistics for Two Trainee Translation Datasets

4.2.2. Query Terms in Translations

As the list of terms is first generated in the term classification process at the corpus level, we need to query the identified terms in translation case by case. Meanwhile, in order to mitigate the influence of their negative effects, we adopted a tri-gram (letter for English and character for Chinese) similarity matching policy on any candidate term pairs. If any ST or TT ngram has a similarity larger than 0.7 with the candidate terms identified by the classifier from the whole lot of ST or TTs, we deem them a success term

[4] http://www.atanet.org/certification/aboutexams_rubic.pdf

		ST			TT		
		precision	recall	F1	precision	recall	F1
XENO	SLR	0.19	0.5	0.28	0.01	1.00	0.01
	MNB	0.02	0.88	0.04	0.00	0.88	0.01
	RF	0.04	0.5	0.07	0.01	1.00	0.01
	SGD	0.18	0.5	0.26	0.01	1.00	0.01
	SVCRBF	0.14	0.38	0.20	0.01	1.00	0.02
	LinearSVC	0.19	0.5	0.28	0.02	0.75	0.04
SUIBE	SLR	0.47	0.47	0.47	0.00	0.94	0.01
	MNB	0.26	0.41	0.32	0.01	0.88	0.01
	RF	0.18	0.18	0.18	0.00	0.94	0.01
	SGD	0.47	0.47	0.47	0.00	0.94	0.01
	SVCRBF	0.41	0.41	0.41	0.02	0.94	0.03
	LinearSVC	0.47	0.47	0.47	0.01	0.88	0.02

Table 6: Monolingual Terminology Identification on Two Datasets

translation. Therefore, terms, such as 'slightly conical pipe segments (' and 'conical pipe segment' and 锥形 管 (6 and 圆锥形 管 段 are likely to be matched when we are going to find out how many terms are correctly translated.

4.2.3. Evaluation

To investigate whether the automatically identified terminology are related to the quality of the trainees' translations, we compute the Pearson correlation coefficient (r), Spearman rank correlation coefficient (R_s) and Kendall's Rank Correlation Coefficient (τ) (Bolboaca and Jäntschi, 2006). The correlation coefficients are calculated as:

$$r = \frac{n\sum x_i y_i - (\sum x_i \sum y_i)}{\sqrt{n\sum {x_i}^2 - (\sum x_i)^2}\sqrt{n\sum {y_i}^2 - (\sum y_i)^2}},$$

where n is the number samples and x_i,y_i are the paired instances of the observed and estimated variables,

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)},$$

where n is the number of smaples and d is the pairwise distances of the ranks of the variables x_i and y_i, and

$$\tau = \frac{S}{\sqrt{n(n-1)/2 - T}\sqrt{n(n-1)/2 - U}}$$

$$T = \sum_t t(t-1)/2$$

$$U = \sum_u u(u-1)/2,$$

where S is the difference between the number of concordant pairs[5] and the number of discordant pairs[6], t is the number of observation of variable x that are tied[7] and u is the number of observation of variable y that are tied[8].

4.3. Results and Findings

We report the confusion matrix of terms identified monolingually by the six classifiers trained above from the English STs and their Chinese TTs in Table 6.

On the XENO data, as is shown in Table 6, six classifiers except for the Multinomial Bayes (MNB) perform rather consistently on the English source text, but display rather apparent variation on the Chinese translations. We manually analysed their predictions on the termhood of ngrams. Almost all classifiers have successfully identified 'xenotransplant', 'xenotransplantation','transplant surgeon' as terms, but they failed to identify two terms 'recipient' (受体) and the institute 'America's Food and Drug Administration'. A possible explanation is that they are ignored because they are singletons which pose difficulty for our statistics-based features to capture the subtlety. For the Chinese translations, all the representative terms, such as '器官 移植 医生' (transplant surgeon), '异种 器官 移植 手术' or '异种 器官 移植' (xenotransplantation), have been successfully identified.

Meanwhile, on the SUIBE patent data, classifiers manifested a significant deterioration of performance, with many terms mistagged. The majority of terms in the ST, such as 'rotary closure', 'neck inner wall', 'radial rib', 'conical pipe segment' and 'pivoting range', were misclassified. On Average, only less than one-third of the true terms (22) from the ST could be recognized by our classifiers.This is in contrast to the good recall of the classifiers on the translations, as is shown in the confusion matrix in Table 6. Term equivalents in the translations for those ST terms that were misclassified are able to be identified by our classifiers (together with a large proportion of false positives). This huge drop of performance on SUIBE ST may be due to the cross-domain effect. As we directly apply the classification model trained from the biomedical data (GENIA 1.0), though our features are supposed to be domain independent, we suspect the domain-shift issue may still impact the classification model. Note that in the table low precision of term classification may be due to the ngram generation process that produces a large amount of sequences of words that are classified as terms, recall is more important in our study though.

As for the alignment process, we report the types, e.g. one-to-one[9], one-to-many[10], many-to-one [11], many-to-many[12] and the length of alignment information for the ST and TTs. Judging from the list of the aligned pairs, the process of alignment has introduced some noises. Either ST terms or equivalent term translations often contain extra words, punctuations. For instance, The pair 'xenotransplantation.' and (异种 器官 移植) has an extra full stop mark. Other types of errors that could be problematic for the later term query include one-to-many and many-to-one alignment, which will cause confusion to the query for matching terms in both ST and TTs. In order to mitigate the influence of their negative effects, we adopted a tri-gram (letter for En-

Data	Length		Type	
XENO	1-word	143	one-to-one	42
	2-word	282	one-to-many	101
	3-word	341	many-to-one	678
	4-word	621	many-to-many	4833
PATENT	1-word	71	one-to-one	57
	2-word	127	one-to-many	14
	3-word	113	many-to-one	50
	4-word	80	many-to-many	334

Table 7: Alignment Types and Distribution

[5]For any pair of observations (x_i, y_i) and (x_j,y_j), where $i \neq j$,they are concordant if the ranks for both elements agree.

[6]if $x_i > x_j$ and $y_i < y_j$, or $x_i < x_j$ and $y_i > y_j$.

[7]if $x_i = x_j$

[8]if $y_i = y_j$

[9]one ST word is aligned to one TT word.

[10]one ST word is aligned to more than one TT word.

[11]More than one ST word is aligned to one TT word.

[12]More than one ST word is aligned to more than one TT word.

Dataset	Terms	Human Annotation	Correlation		
			Pearson	Spearman	Kendall Tau
XENO	$\mu = 3.68, SD = 3.45$	Usefulness/Transfer ($\mu = 24.31, SD = 4.73$)	$r = 0.43, p < 0.01$	$\rho = 0.48, p < 0.0001$	$\tau = 0.37, p < 0.0001$
		Terminology/Style ($\mu = 16.67, SD = 3.06$)	$r = 0.46, p < 0.01$	$\rho = 0.52, p < 0.0001$	$\tau = 0.39, p < 0.0001$
		Idiomatic Writing ($\mu = 17.12, SD = 2.63$)	$r = 0.32, p = 0.02$	$\rho = 0.35, p = 0.01$	$\tau = 0.26, p = 0.01$
		Target Mechanics ($\mu = 9.79, SD = 1.35$)	$r = 0.36, p = 0.01$	$\rho = 0.39, p < 0.01$	$\tau = 0.31, p < 0.01$
		Final Score ($\mu = 71.57, SD = 12.41$)	$r = 0.66, p < 0.0001$	$\rho = 0.72, p < 0.0001$	$\tau = 0.55, p < 0.0001$
SUIBE	$\mu = 10.52, SD = 10.19$	Final Score ($\mu = 87.07, SD = 5.86$)	$r = 0.53, p < 0.001$	$\rho = 0.60, p < 0.001$	$\tau = 0.44, p < 0.0001$

Table 8: Correlation between Term occurrences and Translation Quality

glish and character for Chinese) similarity matching policy on any candidate term pairs. If any ngram has from a ST and a TT both have a similarity larger than 0.7 with the candidate term pairs in the aligned list, we deem them a success term translation. Therefore, terms, such as 'slightly conical pipe segments (' and 'conical pipe segment' and 锥形 管 (6 and 圆锥形 管 段 are likely to be matched when we are going to find out how many terms are correctly translated.

According to the values of three correlation metrics in Table 8, for the XENO dataset, the number of terms identified in both datasets show a positive linear relationship with the four subscores (See Table 8) inbetween weak and moderate ($p < 0.01$). In contrast, the occurrence of terms with the final score (weighted summation of all subscores) goes up beyond moderate ($p < 0.0001$). For the PATENT data, as it has only one final score for all translations, we could also find a moderate linear relationship between the rightly translated terms in the translations ($p < 0.001$). Despite two datasets are evaluated by different annotators under various criteria, correlation scores, either Pearson r, Spearman ρ or Kendall's τ all suggest that the number of correctly translated terms does contribute to translation quality on the whole.

However, it is surprising that there exists only a weak correlation between the second subscore (Terminology/Style) and the term occurrence in the translations. We checked those translations with zero hit of terms but over strong quality scores. We found translation of terminology, semantic adequacy and language fluency are present in the translation indeed, see Table 9. Typical terms in the specific domain, such as '异种 器官 移植 '(xenotransplantation), '器官 移植 外科 医生'(transplant surgeons),'美国 食物 药物 管理局'(America 's food and drug administration) are adequately translated. One thing in common with these translations is that through the translation terms are rendered with slight variation. For example, in one sample, both '器官 移植 外科 医生'(transplant surgeons) and '器官 移植 手术 师'(transplant surgery technician) are used for the same source term 'transplant surgeon'. Both translations are acceptable expressions in Chinese in terms of adequacy and fluency. This term inconsistency or variation may have to do with why such translations are evaluated reasonably high even with few or no term counts by our trained term classifiers. It implies that our approach of term classification may have fallacy in handling term variation.

#	ST	TT
1	Transplant surgeons work miracles.	器官移植外科医生带来了奇迹。
	They take organs from one body and integrate them into another,	他们将器官从一个身体中取出并将它们植入他者体内,
	granting the lucky recipient a longer, better life.	让那些幸幸得到 它们的人活得更长，更好。
2	America's food and drug administration has already published draft guidelines for xenotransplantation.	美国的食品药物管理局已经出版了异种器官移植 草案 细则。
	The ethics of xenotransplantation are relatively unworrying.	这种手术在伦理道德 领域 相对而言，不那么令人担忧了。
3	So far attempts to make artificial organs have been disappointing:	到目前为止，试图人工制造器官的可能性已经被否定了，
	Nature is hard to mimic. hence the renewed interest in trying to use organs from animals.	毕竟自然是很难去模仿的，因此，人们正将更多的目光集中在动物器官上。

Table 9: Adequately Translated Terms:Term Variation

5. Conclusion

In this study, we explored ways of identifying terms from monolingual texts and integrate them into investigating the contribution of terminology to translation quality. It is found that the number of term frequencies identified automatically has weak linear correlation with the four subscores for the xenotransplantation data. When it comes to the overall final score for both datasets, such correlations increase to be above moderate and strong. This study indicates that the term occurrence in translation could be an valuable quality indicator for estimating translation quality. In the future, we deem it necessary to try other weakly supervised method to improve term identification accuracy, particularly those low frequency terms and their variations. Ultimately, term occurrence will be incorporated into the existing feature set (Yuan et al., 2016) in quality estimation tasks.

6. Acknowledgements

We are thankful to the generosity of Dr. Xiaoying Hu's sharing the SUIBE data and Jie Gao's kind assistance in preparing the monolingual training and testing data. The first author is supported by the China Scholarship Council-University of Leeds Scholarship (Support No. [2013]3009).

7. References

Blancafort, H., Daille, B., Gornostay, T., Heid, U., Méchoulam, C., and Sharoff, S. (2010). TTC: Terminology Extraction, Translation Tools and Comparable Corpora. In European Association for Lexicography, editor, *Proceedings of the XIV Euralex International Congress*, pages 263–268, Leeuwarden/Ljouwert, Netherlands, July.

Bolboaca, S.-D. and Jäntschi, L. (2006). Pearson versus spearman, kendall's tau correlation analysis on structure-activity relationships of biologic active compounds. *Leonardo Journal of Sciences*, 5(9):179–200.

Brunette, L. (2000). Towards a terminology for translation quality assessment: A comparison of tqa practices. *Translator: Studies in Intercultural Communication*, 6(2):169–182.

Erdmann, M., Nakayama, K., Hara, T., and Nishio, S. (2009). Improving the extraction of bilingual terminology from wikipedia. *ACM Trans. Multimedia Comput. Commun. Appl.*, 5(4):31:1–31:17, November.

Finkel, J., Dingare, S., Nguyen, H., Nissim, M., Manning, C., and Sinclair, G. (2004). Exploiting context for biomedical entity recognition: from syntax to the web. In Nigel Collier, et al., editors, *COLING 2004 International Joint Workshop on Natural Language Processing*

in Biomedicine and Its Applications (NLPBA/BioNLP) 2004, pages 91–94. COLING, August.

Gaizauskas, R., Paramita, M. L., Barker, E., Pinnis, M., Aker, A., and Solé, M. P. (2015). Extracting bilingual terms from the web. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 21(2):205–236.

Gómez Guinovart, X. and Simoes, A. (2009). Parallel corpus-based bilingual terminology extraction. In Anne Condamines et al., editors, *8th International Conference on Terminology and Artificial Intelligence*, Toulouse,France, November.

Hakami, H. and Bollegala, D. (2017). A classification approach for detecting cross-lingual biomedical term translations. *Natural Language Engineering*, 23(1):31–51.

Hartley, A., Mason, I., Peng, G., and Perez, I. (2004). Peer and self-assessment in conference interpreter training. research project. pedagogical research fund in languages, linguistics and area studies. *The Higher Education Academy, Heslington, United Kingdom. Retrieved May*, 7:2005.

Karoubi, B. (2016). Translation quality assessment demystified. *Babel*, 62(2):253–277.

Kim, J.-D., Ohta, T., Tateisi, Y., and Tsujii, J. (2003). Genia corpus–a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl 1):i180–i182.

Kim, T., Hwang, M., Hwang, M., Song, S., Jeong, D., and Jung, H. (2015). Translation of technical terminologies between english and korean based on textual big data. *Software: Practice and Experience*, 45(8):1115–1126.

Li, L., Dang, Y., Zhang, J., and Li, D. (2012). Domain term extraction based on conditional random fields combined with active learning strategy. *Journal of Information &Computational Science*, 9(7):1931–1940.

Lommel, A., Uszkoreit, H., and Burchardt, A. (2014). Multidimensional quality metrics (mqm). *Tradumàtica*, No.(12):0455–463.

Macken, L., Lefever, E., and Hoste, V. (2013). Texsis: Bilingual terminology extraction from parallel corpora using chunk-based alignment. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 19(1):1–30.

Rocheteau, J. and Daille, B. (2011). TTC TermSuite: A UIMA Application for Multilingual Terminology Extraction from Comparable Corpora. In *5th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 9–12, Chiang Mai, Thailand, November. System Demonstrations.

Secară, A. (2005). Translation evaluation–a state of the art survey. In *Proceedings of the eCoLoRe/MeLLANGE Workshop*, pages 39–44, Leeds, UK.

Warburton, K. (2013). Processing terminology for the translation pipeline. *Terminology: International Journal of Theoretical and Applied Issues in Specialized Communication*, 19(1):93.

Xu, R. and Sharoff, S. (2014). Evaluating term extraction methods for interpreters. In *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*, pages 86–93, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Xu, Y., Chen, L., Wei, J., Ananiadou, S., Fan, Y., Qian, Y., Eric, I., Chang, C., and Tsujii, J. (2015). Bilingual term alignment from comparable corpora in english discharge summary and chinese discharge summary. *BMC bioinformatics*, 16(1):149.

Yuan, Y., Sharoff, S., and Babych, B. (2016). Mobil: A hybrid feature set for automatic human translation quality assessment. In *Proc Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portorož, Slovenia, May.

Yuan, Y., Gao, J., and Zhang, Y. (2017). Supervised learning for robust term extraction. In *Proceedings of 2017 International Conference on Asian Language Processing (IALP)*, Singapore, December. IEEE.

Zhang, Z., Gao, J., and Ciravegna, F. (2016). Jate 2.0: Java automatic term extraction with apache solr. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Zhou, G. and Su, J. (2004). Exploring deep knowledge resources in biomedical name recognition. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, JNLPBA '04, pages 96–99, Geneva, Switzerland. Association for Computational Linguistics.

Machine Translation of Low-Resource Spoken Dialects: Strategies for Normalizing Swiss German

Pierre-Edouard Honnet[*,1], **Andrei Popescu-Belis**[*,2], **Claudiu Musat**[3], **Michael Baeriswyl**[3]

[1] Telepathy Labs
Schützengasse 25
CH-8001 Zürich
Switzerland
pierre-edouard.honnet@telepathy.ai

[2] HEIG-VD / HES-SO
Route de Cheseaux 1, CP 521
CH-1401 Yverdon-les-Bains
Switzerland
andrei.popescu-belis@heig-vd.ch

[3] Swisscom (Schweiz) AG
Genfergasse 14
CH-3011 Bern
Switzerland
claudiu.musat@swisscom.com
michael.baeriswyl@swisscom.com

Abstract

The goal of this work is to design a machine translation (MT) system for a low-resource family of dialects, collectively known as Swiss German, which are widely spoken in Switzerland but seldom written. We collected a significant number of parallel written resources to start with, up to a total of about 60k words. Moreover, we identified several other promising data sources for Swiss German. Then, we designed and compared three strategies for normalizing Swiss German input in order to address the regional diversity. We found that character-based neural MT was the best solution for text normalization. In combination with phrase-based statistical MT, our solution reached 36% BLEU score when translating from the Bernese dialect. This value, however, decreases as the testing data becomes more remote from the training one, geographically and topically. These resources and normalization techniques are a first step towards full MT of Swiss German dialects.

Keywords: machine translation, low-resource languages, spoken dialects, Swiss German, character-based neural MT

1. Introduction

In the era of social media, more and more people make online contributions in their own language. The diversity of these languages is however a barrier to information access or aggregation across languages. Machine translation (MT) can now overcome this limitation with considerable success for well-resourced languages, i.e. language pairs which are endowed with large enough parallel corpora to enable the training of neural or statistical MT systems. This is not the case, though, for many low-resourced languages which have been traditionally considered as oral rather than written means of communication, and which often lack standardized spelling and/or exhibit significant variations across dialects. Such languages have an increasing presence in written communication, especially through social media, while remaining inaccessible to non-speakers.

This paper presents a written MT system for a mostly spoken family of dialects: Swiss German. Although spoken in a technologically developed country by around five million native speakers, Swiss German has never been significantly used in writing – with the exception of folklore or children books – before the advent of social media. Rather, from primary school, speakers of Swiss German are taught to use High German in writing, more precisely a variety known to linguists as Swiss Standard German, which is one of the three official federal languages along with French and Italian. Swiss German is widely used in social media, but foreigners or even Swiss speakers of the other official languages cannot understand it.

In this paper, we describe the first end-to-end MT system from Swiss German to High German. In Section 2, we present the Swiss German dialects and review the scarce monolingual and even scarcer parallel language resources that can be used for training MT. In Section 3, we review previous work on Swiss German and on MT of low-resource languages. In Section 4, we address the major issue of dialectal variation and lack of standard spelling – which affects many other regional and/or spoken languages as well – through three solutions: explicit conversion rules, phonetic representations, and character-based neural MT. These solutions are combined with phrase-based statistical MT to provide a standalone translation system, as explained in Section 5. In Section 6 we present evaluation results. We first find that the similarity between the regions of training vs. test data has a stronger effect on performance than the similarity of text genre. Moreover, the results show that character-based NMT is beneficial for dealing with spelling variation. Our system is thus an initial general purpose MT system making Swiss German accessible to non-speakers, and can serve as a benchmark for future, better-resourced attempts.

2. Collecting Swiss German Resources

2.1. A Heterogeneous Family of Dialects

Definition. Swiss German (Russ, 1990; Christen et al., 2013) is a family of dialects used mainly for spoken communication by about two thirds of the population of Switzerland (i.e. over five million speakers). Swiss German is typically learned at home as a first language, but is substituted starting from primary school by High German for all written forms, as well as for official spoken discourse, for instance in politics or the media. Linguistically, the variety of High German written and spoken in Switzerland is referred to as *Swiss Standard German* (see Russ (1994),

[*] Work conducted while the first and second authors were at the Idiap Research Institute, Martigny, Switzerland.

Chapter 4, p. 76–99) and is almost entirely intelligible to German or Austrian speakers. On the contrary, Swiss German is generally not intelligible outside Switzerland.

In fact, Swiss German constitutes a group of heterogeneous dialects, which exhibit strong local variations. Due to their spoken nature, they have no standardized written form: for instance, the word *kleine* (meaning small' in Standard German) could be written as *chlyni*, *chliini*, *chline*, *chli* or *chlii* in Swiss German. Linguistic studies of the Swiss German dialects (see Russ (1990) or Christen et al. (2013)) generally focus on the phonetic, lexical or syntactic variations and their geographical distribution, often concluding that such variations are continuous and non-correlated with each other. Finally, little teaching material in Swiss German is available to foreigners.

Divisions. The areas where each dialect is spoken are influenced both by the administrative divisions (cantons and communes) and by natural borders (topography). Within the large group of Germanic languages, the dialects of Switzerland belong to the *Alemannic* group. However, while a majority of dialects are *High Alemannic* (yellow area on map in Figure 1), those spoken in the city of Basel and in the Canton of Valais belong respectively to the *Low Alemannic* and the *Highest Alemannic* groups. Within the High Alemannic group, a multitude of divisions have been proposed. One of the most consistent ones is the Brunig-Napf-Reuss line between the eastern and western groups (red line in Fig. 1). A fine-grained approach could easily identify one or more dialects for each canton.

For the purpose of this study, we distinguish only two additional sub-groups on each side of the Brunig-Napf-Reuss line, and refer to them using the largest canton in which they are spoken. Westwards, we distinguish the Bernese group from the group spoken around Basel (cantons of Basel-Country, Solothurn and parts of Aargau). Eastwards, we distinguish the Zürich group from the easternmost group around St. Gallen. Therefore, for training and testing MT on various dialects, we consider in what follows six main variants of Swiss German, represented on the map in Figure 1.

Notations. We refer to Swiss German as 'GSW' (abbreviation from ISO 639-2) followed by the indication of the variant: GSW-BS (city of Basel), GSW-BL (regions of Basel, Solothurn, parts of Aargau), GSW-BE (mainly canton of Bern), GSW-ZH (canton of Zurich and neighbors), GSW-SG (St. Gallen and easternmost part of Switzerland), GSW-VS (the German-speaking part of the canton of Valais/Wallis). These groups correspond to the dialect labels used in the Alemannic Wikipedia (see Section 2.2 below): *Basel*, *Baselbieter*, *Bern*, *Zurich*, *Ùndertòggeborg*, and *Wallis* (Valais). In contrast, Swiss Standard German is referred to as 'DE-CH', a qualified abbreviation from IETF. Moreover, below, we will append the genre of the training data to the dialect abbreviation.

Usage and Need for MT. Swiss German is primarily used for spoken communication, but the widespread adoption of social media in Switzerland has significantly increased its written use for informal exchanges on social platforms or in text messages. No standardized spelling has emerged yet, a fact related to the lack of GSW teaching as a second language. GSW is still written partly with reference to High German and partly using a phonetic transcription, also inspired from German pronunciation. Access to such content in social media is nearly impossible to foreigners, and even to speakers of different dialects, e.g. Valaisan content to Bernese speakers. Our goal is to design an MT system translating all varieties of GSW (with their currently observed spelling) towards High German, taking advantage of the relative similarity of these languages. By pivoting through High German, other target languages can then be supported. Moreover, if a speech-to-text system existed for Swiss German (Garner et al., 2014), our system would also enable spoken translation.

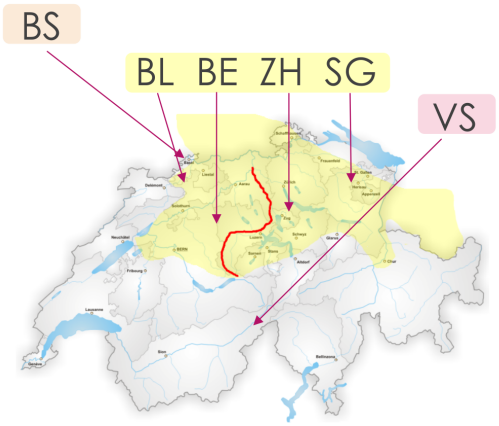


Figure 1: Map of Switzerland, with six main dialects that we identify for our study. The area in yellow indicates the High Alemannic dialects. Image source: https://commons.wikimedia.org/wiki/File:Brunig-Napf-Reuss-Linie.png.

2.2. Parallel Resources

Despite attempts to use comparable corpora or even monolingual data only (reviewed in Section 3), parallel corpora aligned at the sentence level are essential resources for training statistical MT systems. In our case, while written resources in Swiss German are to some extent available (as reviewed in Section 2.3), it is rare to find their translations into High German or vice-versa. When these are available, the two documents are often not available in electronic version, which requires a time-consuming digitization effort to make them usable for MT.[1]

One of our goals is to collect the largest possible set of parallel GSW/DE texts, in a first stage regardless of their licensing status. We include among such resources parallel lexicons ("dictionaries"), and show that they are helpful for training MT. We summarize in Table 1 the results of our data collection effort, providing brief descriptions of each resource with especially their variant of GSW and their domain. We describe in detail each resource hereafter.

[1]Many of them are children books, such as *Pitschi* by Hans Fischer, *The Gruffalo* by Julia Donaldson, or *The Little Prince* by Antoine de Saint-Exupéry. Other examples include transcripts of Mani Matter's songs, or several Asterix comics in Bernese.

Dataset	Train	Dev.	Test	Total
GSW-BE-Novel	2,667	218	183	3,251
GSW-BE-Wikipedia	–	180	67	247
GSW-VS-Radio	463	100	50	613
GSW-ZH-Wikipedia	–	45	50	95
GSW-BE-Bible	–	–	126	126
GSW-Archimob	40,159	2,710	2,710	45,579
GSW-ZH-Lexicon1	1,527	–	–	1,527
GSW-BE-Lexicon2	1,224	–	–	1,224

Table 1: GSW/DE parallel datasets partitioned for MT training, tuning and testing, with sizes in numbers of parallel sentences. The lexicons (last two lines) were not used for testing, and 183 additional lines from GSW_BE_Novel are kept apart for future testing.

GSW-BE-Novel. Translations of books from DE into GSW are non-existent. We thus searched for books written originally in GSW and then translated into DE. Among the growing body of literature published in Swiss German, we found only one volume translated into High German and available in electronic form: *Der Goalie bin ig* (in English: *I am the Keeper*), written in Bernese by Pedro Lenz in 2010. The DE translation stays close to the original GSW-BE text, therefore sentence-level alignment was straightforward, resulting in 3,251 pairs of sentences with 37,240 words in GSW-BE and 37,725 words in DE.

GSW-BE-Wikipedia and **GSW-ZH-Wikipedia**. The Alemannic version of Wikipedia[2] appeared initially as a promising source of data. However, its articles are written not only in Swiss German, but also in other Alemannic dialects such as Alsatian, Badisch and Swabian. As its contributors are encouraged to write in their own dialects, only a few articles are homogeneous and have an explicit indication of their dialect, using an Infobox with one of the six labels indicated above. Among them, even fewer have an explicit statement indicating that they have been translated from High German (which would make the useful as parallel texts). We identified two such pages and sentence-aligned them to serve as test data: "*Hans Martin Sutermeister*" translated from DE into GSW-BE and "*Wädenswil*" from DE into GSW-ZH.[3]

GSW-VS-Radio. A small corpus of Valaisan Swiss German (also called *Wallisertiitsch*) has been collected at the Idiap Research Institute (Garner et al., 2014).[4] The corpus consists of transcriptions of a local radio broadcast[5] translated into High German.

GSW-BE-Bible. The Bible has been translated in several GSW dialects, but the only electronic version available to us were online excerpts in Bernese.[6] However, this is not translated from High German but from a Greek text, hence the alignment with any of the German Bibles is problematic.[7] We selected the contemporary *Gute Nachricht Bibel* (1997) for its modern vocabulary, and generated parallel data from four excerpts of the Old and New Testament, while acknowledging their particular style and vocabulary. The following excerpts were aligned: *Üse Vatter*, *D Wienachtsgschicht*, *Der barmhärzig Samaritaner* and *D Wält wird erschaffe*.

GSW-Archimob. Archimob is a corpus of standardized Swiss German (Samardžić et al., 2016), consisting of transcriptions of interviewees speaking Swiss German, with a word-align normalized version in High German.[8] The interviews record memories of WW II, and all areas of Switzerland are represented. In most cases, the normalization provides the corresponding High German word or group of words, but in other cases it is Swiss German with a standardized orthography devised by the annotators. Using a vocabulary of High German, we filtered out all sentences whose normalizations included words outside this vocabulary. In other words, we kept only truly High German sentences, along with their original Swiss German counterparts, resulting in about 45,000 GSW/DE word-aligned sentence pairs.

GSW-ZH-Lexicon and **GSW-BE-Lexicon**. The last two parallel resources are vocabularies, i.e. lists of GSW words with their DE translation. As such, they are useful for training our research systems, but not for testing them. The first one is based on *Hoi Zäme*, a manual of Zürich Swiss German intended for High German speakers. The data was obtained by scanning the printed version, performing OCR[9] and manually aligning the result. Although the book contains also parallel sentences, only the bilingual dictionary was used in our study, resulting in 1,527 words with their translations. A similar dictionary for Bernese (GSW-BE vs. DE) was found online[10] with 1,224 words for which we checked and corrected the alignments.

2.3. Monolingual Resources

The Phonolex dictionary, a phonetic dictionary of High German,[11] was used for training our grapheme-to-phoneme converter (see Section 4.2). It contains High German words with their phonetic transcriptions.

[2] http://als.wikipedia.org

[3] These pages are respectively available at https://de.wikipedia.org/wiki/Hans_Martin_Sutermeister (High German), https://als.wikipedia.org/wiki/Hans_Martin_Sutermeister (Bernese), https://de.wikipedia.org/wiki/W%E4denswil (High German), and https://als.wikipedia.org/wiki/W%E4denswil (Zurich Swiss German).

[4] www.idiap.ch/dataset/walliserdeutsch

[5] Radio Rottu, http://www.rro.ch.

[6] www.edimuster.ch/baernduetsch/bibel.htm

[7] www.die-bibel.de/bibeln/online-bibeln/

[8] http://www.spur.uzh.ch/en/departments/korpuslab/ArchiMob.html

[9] Tesseract: https://github.com/tesseract-ocr/

[10] www.edimuster.ch/baernduetsch/woerterbuechli.htm

[11] www.bas.uni-muenchen.de/forschung/Bas/BasPHONOLEXeng.html. We also use it to find OOV words.

About 75 pages from the Alemannic Wikipedia mentioned above have been collected and used to derive orthographic normalization rules in Section 4.1. To build language models (see Section 5) we used the News Crawls 2007–2015 from the Workshop on MT.[12]

3. Previous Work on Swiss German and the MT of Low-Resource Languages

The variability of Swiss German dialects has been investigated in a number of studies, such as those by Russ (1990), Scherer (2012a), and Christen et al. (2013). This variability was illustrated in a system for generating Swiss German text, with fine-grained parameters for each region on a map (Scherrer, 2012b).

Language resources for Swiss German are extremely rare. The ArchiMob corpus (Samardžić et al., 2016) is quite unique, as it provides transcripts of spoken GSW narratives, along with their normalization, as presented above (Samardžić et al., 2015). First performed manually – thus generating ground-truth data – the normalization was then performed automatically using character-based statistical MT (Scherrer and Ljubešić, 2016).

Initial attempts for MT of GSW include the above-mentioned system for generating GSW texts from DE (Scherrer, 2012a), and a system combining ASR and MT of Swiss German from Valais (Garner et al., 2014). A normalization attempt for MT, on a different Germanic dialect, has been proposed for Viennese (Hildenbrandt et al., 2013).

The MT of low-resource languages or dialects has been studied on many other important cases, in particular for Arabic dialects which are also predominantly used for spoken communication (Zbib et al., 2012). The lack of a normalized spelling of dialects has for instance an impact on training and evaluation of automatic speech recognition: a solution is to address spelling variation by mining text from social networks (Ali et al., 2017). Other strategies are the crowdsourcing of additional parallel data, or the use of large monolingual and comparable corpora to perform bilingual lexicon induction before training an MT system (Klementiev et al., 2012; Irvine and Callison-Burch, 2013; Irvine and Callison-Burch, 2016). The METIS-II EU project replaced the need for parallel corpora by using linguistic pre-processing and statistics from target-language corpora only (Carl et al., 2008). In a recent study applied to Afrikaans-to-Dutch translation, the authors use a character-based "cipher model" and a word-based language model to design a decoder for the low-resourced input language (Pourdamghani and Knight, 2017).

The Workshops on Statistical MT have proposed translation tasks for low-resourced languages to/from English, such as Hindi in 2014 (Bojar et al., 2014), Finnish in 2015, or Latvian in 2017. However, these languages are clearly not as low-resourced as Swiss German, and possess at least a normalized version with a unified spelling. In 2011, the featured translation task aimed at translating text messages from Haitian Creole into English, with a parallel corpus of similar size as ours (ca. 35k words on each side, plus a Bible translation). The original system built in the wake of the 2010 Haiti earthquake leveraged a phonetic mapping from French to Haitian Creole to obtain a large bilingual lexicon (Lewis, 2010; Lewis et al., 2011).

4. Normalizing Swiss German for MT

Three issues must be addressed when translating Swiss German into High German, which all contribute to a large number of out-of-vocabulary (OOV) words (i.e. previously unseen during training) in the source language:

1. The scarcity of parallel GSW/DE data for training (see Section 2.2), which cannot be easily addressed by the strategies seen in Section 3.
2. The variability of dialects across training and testing data, which increases dialect-specific scarcity.
3. The lack of a standard spelling, which introduces intra-dialect and intra-speaker variability.

There are several ways to address these issues. The most principled one is the normalization of all GSW input using unified spelling conventions, coupled with the design of a GSW/DE MT system for normalized input. However, such a goal is far too ambitious for our scope. Instead, we propose here to normalize Swiss German input for the concrete perspective of MT by converting unknown GSW words either to known GSW ones or to High German ones, which are preserved by the GSW/DE MT system and increase the number of correctly translated words.[13]

This procedure, summarized below, rests on the assumption that many OOV GSW words are close to DE words, but with a slightly different pronunciation and spelling (see examples in the third column of Table 2). Each of the three strategies follow the same procedure:

1. For each OOV word w, apply the normalization strategy. If it changes w into w' then go to (2), if not to (4).
2. If w' is a known GSW word then replace w with w' and proceed to (4), if not, go to (3).
3. If w' is a known DE word then replace w with w'. If not, leave w unchanged and go to (4).
4. Translate the resulting GSW text.

This normalization method has two possible chances to help MT, by converting OOV words either into a known GSW word, or into a correct DE word which is no longer processed by MT. We describe below three strategies to normalize GSW text input before GSW/DE MT.

4.1. Explicit Spelling Conversion Rules

The first strategy is based on explicit conversion rules for every OOV word w, which is changed into w' by applying in sequence several spelling conversion rules, keeping the result if it is a GSW or a DE word, as explained above. The orthographic rules implemented in our system are shown in Table 2, with possible conversion examples.

4.2. Using Phonetic Representations

The second approach is based on the assumption that despite spelling differences, variants of the same word will

[12] http://www.statmt.org/wmt17/translation-task.html

[13] The MT system is specifically built so that OOV words are copied in the target sentence, rather than deleted.

Spelling	Convert to	Example
.*scht.*	.*st.*	Angscht → Angst
.*schp.*	.*sp.*	Schprache → Sprache
ˆgäge.*	ˆgegen.*	Gägesatz → Gegensatz
CäC	CeC	Präsident → President
ˆgm.*	ˆgem	Gmeinde → Gemeinde
ˆgf.*	ˆgef	gfunde → gefunde(n)
ˆgw.*	ˆgew	gwählt → gewählt
ˆaa.*	ˆan.*	Aafang → Anfang
.*ig$	.*ung$	Regierig → Regierung
ˆii.*	ˆein.*	Iiwohner → Einwohner

Table 2: Orthographic conversion rules using meta-characters ˆ and $ for the beginning and end of a word, .* for any sequence of characters, and C for any consonant.

have the same pronunciation. Thus, converting an out-of-vocabulary (OOV) word to its phonetic transcription may allow finding the equivalent word which is present in the vocabulary. In this case, substituting the OOV word with a known word with the same pronunciation should help MT, assuming the same meaning.

For this, a grapheme-to-phoneme (G2P) converter is needed. It consists of an algorithm which is able to convert sequences of characters into phonetic sequences, or go from the written form of a word to its pronunciation. The idea is to build it on High German, as we expect Swiss German to be written in a phonetic way, which means that the G2P conversion should be close to High German pronunciation rules. In our experiments, a G2P converter was trained on the Phonolex dictionary, which contains High German words with their phonetic transcriptions. A GSW phonetic dictionary was created by using this system. To translate a new OOV word, we convert the word to its pronunciation, and check whether the resulting pronunciation exists either in the phonetic GSW dictionary or in the phonetic DE dictionary, following the procedure explained at the beginning of this section.

4.3. Character-based Neural MT

Mainstream neural MT systems are typically trained using recurrent neural networks (RNNs) to encode a source sentence, and then decode its representation into the target language (Cho et al., 2014). The RNNs are often augmented with an attention mechanism to the source sentence (Bahdanau et al., 2014). However, training an NMT is not feasible for GSW/DE, as the size of our resources is several orders of magnitude below NMT requirements. However, several recent approaches have explored a new strategy: the translation system is trained at the character level (Ling et al., 2015; Costa-jussà and Fonollosa, 2016; Chung et al., 2016; Bradbury et al., 2017; Lee et al., 2017), or at least character-level techniques such as byte-pair encoding are used to translate OOV words (Sennrich et al., 2016).

As the available data is limited, one possible approach is to combine a PBSMT and a CBNMT system: the former translates known words, while the latter translates OOV ones. The CBNMT system has two main advantages: it can translate unseen words based on the spelling regularities observed in the training data, and it can be trained with smaller amounts of data compared to the requirements of standard NMT methods.

Among the CBNMT approaches, we use here Quasi Recurrent Neural Networks (QRNNs) (Bradbury et al., 2017), which take advantage of both convolutional and recurrent layers. The increased parallelism introduced by the use of convolutional layers allows to speed up both training and testing of translation models. There are two advantages to use CBNMT for OOV translation only. First, the training data may be sufficient to capture spelling conversion better than hand-crafted rules such as those in Table 2. Second, we can use smaller recurrent layers, as the character sequences to translate for OOV words are much shorter than sentences.

We built a CBNMT system for OOV words based on open source scripts for TensorFlow available online, using the implementation of the QRNN architecture proposed by Kyubyong Park,[14] with the following modifications:

1. We added a "start of word" symbol to avoid mistakes on the first letter of the word. This was done outside the translation scripts, by adding the ':' symbol to each word before the first letter, and removing it after translation.
2. We modified the QRNN translation script to allow the translation of input texts without scoring the translation (for the production mode, when no reference is available).
3. We added the possibility to translate an incomplete minibatch, by padding the last incomplete batch with empty symbols (0).[15]
4. We set the following hyper-parameters: the maximum number of characters is 40, as no longer words were found in our GSW vocabulary. The minibatch size was kept to 16, and the number of hidden units was kept to 320, as in the default implementation.

We trained the CBNMT model using unique word pairs from the Archimob corpus (see 2.2 above), i.e. a Swiss German word and its normalized version, with a training set of 40,789 word pairs and a development set of 2,780 word pairs.

5. Integration with Machine Translation

We use phrase-based statistical MT for the core of our system, as data was not sufficient to train neural MT. We experimented indeed with two NMT systems[16], which are typically trained on at least one million sentences, and tuned on 100k. In our case, the available data did not allow NMT to outperform PBSMT, which is used below.

Using the Moses toolkit[17] to build a PBSMT system (Koehn et al., 2003), we used various subsets of the parallel GSW/DE data presented in Section 2.2 above to learn

[14] https://github.com/Kyubyong/quasi-rnn

[15] Originally, if the size of the minibatch is n, and the number of sentences modulo n is y (i.e. there are $n * x + y$ sentences), then only the $n * x$ first sentences were translated by the system, which ignored the y last ones.

[16] The DL4MT toolkit https://github.com/nyu-dl/dl4mt-tutorial and the OpenNMT-py one https://github.com/OpenNMT/OpenNMT-py.

[17] http://www.statmt.org/moses/

translation models. As for the target language model, we trained a tri-gram model using IRSTLM[18] over ca. 1.3 billion words in High German, and tuned the system using the development data indicated above in Section 2.3. As explained above, the normalization strategies are used to attempt to change GSW OOV words into either GSW or even DE words that are in the vocabulary. As we will see, we have combined two strategies in several experiments. We will use the most common metric for automatic MT evaluation, i.e. the BLEU score (Papineni et al., 2002).

6. Results and Discussion

6.1. Effects of Genre and Dialect

The first system to translate from Swiss German into High German was built using Moses trained on the Bernese novel corpus (GSW-BE-Novel in Table 1 above), with character-based NMT for OOV words. Table 3 shows the BLEU scores obtained when testing this system on test sets from different regions or topics. Moreover, we also vary the tuning sets, including ones closer to the target test domains to assess their impact. The best BLEU score is around 35% which can be compared, for instance, with Google's NMT scores of 41% for EN/FR and 26% for EN/DE, trained on tens of millions of sentences and nearly one hundred processors (Wu et al., 2016). In our case, our modest resources enable us to reach quite a high score thanks to the normalization strategy and the relative similarity of GSW to DE.

Test set	Tuning (dev) set	BLEU
GSW-BE-Novel	GSW-BE-Novel	35.3
GSW-BE-Wikipedia	GSW-BE-Novel	21.9
same	GSW-BE-Wikipedia	21.7
GSW-ZH-Wikipedia	GSW-BE-Novel	16.2
same	GSW-ZH-Wikipedia	15.3
GSW-VS-Radio	GSW-BE-Novel	9.7

Table 3: BLEU scores for various tuning and test sets for the baseline system trained on GSW-BE-Novel. Performance decreases significantly as the dialect and domain are more remote from the training/tuning data.

A typical output of this system is:

GSW-BE source: *fasch wiwenernolänger hätt wöueküsse . oder hanisächtnumegmeint?*

DE MT: *fast wie wenn er noch länger hätte wollten küsse . oder hab ich es wohl nur gemeint ?*

Human DE reference: *als hätte er noch länger küssen wollen . oder etwa nicht ?*

The scores in Table 3 show the following trends:

1. When testing on similar data, i.e. the same dialect and same domain, the scores are the highest, and in the same range as state of the art EN-DE or EN-FR systems.
2. When changing domain (testing on Wikipedia data in the same dialect), the scores are decreasing.
3. When testing on different dialects, the scores decrease more. This is true both for GSW-ZH and GSW-VS. As the dialect and domain are further from the data used to train the system, the score gets lower. GSW-VS is known to be very different from any other GSW dialect, and radio broadcast data is expected to be very different from the novel used at training time.

[18] http://hlt-mt.fbk.eu/technologies/irstlm

6.2. Effect of the Size of Training Data and Language Model

To evaluate first the effect of using more training data, with larger vocabularies, a new system was trained using the same data as in the previous experiments, complemented with the two bilingual lexicons presented in Section 2.2. Table 4, second column, presents the resulting BLEU scores, which increase in all cases by about 1 BLEU point. As expected, using more training data in the form of bilingual lexicons yields more reliable translation models.
To build more robust systems, we also used a larger target language model built on the NewsCrawl 2007-2015 data from WMT (see Section 2.3) instead of only the DE side of our parallel data, which is still used for training the translation models, as above. Table 4, third column, gives the BLEU scores on the same test sets, using the larger target language model. We observe that the scores decrease slightly for the Bernese test sets, and hypothesize that this is due to the different domains of the language model and the test set. However, as the larger language model is trained on more diverse data, we will keep using it below for its robustness.

	BLEU	
Test set	Small LM	Large LM
GSW-BE-Novel	36.2	34.1
GSW-BE-Wikipedia	23.6	22.7
GSW-ZH-Wikipedia	17.3	17.7
GSW-VS-Radio	10.0	10.7

Table 4: BLEU scores for various test sets (Bern, Zurich, Valais dialects) for a Moses-based system trained over data *including the two GSW/DE dictionaries* with two language models (LM).

6.3. Out-of-Vocabulary GSW Words

The three approaches proposed for normalization were evaluated on the same datasets as the previous systems. Additionally, two other approaches combining, on one side orthographic and phonetic based conversions, and on the other side CBNMT and phonetic conversion, were evaluated. Table 5 summarizes the results for the baseline system and the proposed approaches.
Baseline1 corresponds to the system with a language model trained only on the parallel GSW-DE data, while Baseline2 is using a larger language model, described in Sec. 6.2. We can make the following observations:

- In all the cases except GSW-BE-Novel, the orthographic approach improves the BLEU score of the baseline system, and the improvement is bigger for more remote dialects and domains.
- The phonetic approach improves the score in 4 out of 6 cases. In the remaining cases, we suppose that some words did not require pre-processing, and that the pre-processing may have converted the word to a false pos-

Test set	Baseline1	Baseline2	Phon.	Orth.	Orth. & Phon.	CBNMT & Phon.	CBNMT
GSW-Archimob	10.9	10.8	10.8	11.2	13.9	27.9	**32.9**
GSW-BE-Novel	**36.2**	34.1	34.3	34.4	34.4	35.6	35.4
GSW-BE-Wikipedia	23.6	22.7	23.2	23.6	23.7	20.5	**24.0**
GSW-ZH-Wikipedia	17.3	17.7	17.1	18.9	18.2	22.0	**22.1**
GSW-VS-Radio	10.0	10.7	11.0	12.2	12.0	8.7	**22.9**
GSW-BE-Bible	5.7	5.8	6.2	6.1	**6.4**	6.3	6.3

Table 5: BLEU scores for several test sets and normalization strategies (orthographic, phonetic, character-based NMT).

itive (i.e. the algorithm found a matching word, but it was not the correct one for translation).

- Combining both approaches always results in better scores than the baseline, but in the case for which the phonetic approach score deteriorated, orthographic conversion only performs better.
- In all the cases, combining CBNMT with the baseline PBSMT works the best. The highest improvement is brought when dialect or domain are different (except for the Bible), because more data was used to train the CBNMT models. This is especially true for the GSW-Archimob test set, which has similar data as the one used to train the CBNMT models.
- Baseline1 performs better than all the systems for GSW-BE-Novel test set. This is expected as the training data is both from the same dialect and the same domain. Additionally, the language model is trained on this same data.

7. Conclusion

In this paper, we proposed solutions for the machine translation of a family of dialects, Swiss German, for which parallel corpora are scarce. Our efforts on resource collection and MT design have yielded:

- a small Swiss German / High German parallel corpus of about 60k words;
- a larger list of resources which await digitization and alignment;
- three solutions for input normalization, to address variability of region and spelling;
- a baseline GSW-to-DE MT system reaching 36 BLEU points.

Among the three normalization strategies, we found that character-based neural MT was the most promising one. Moreover, we found that MT quality depended more strongly on the regional rather than topical similarity of test vs. training data.

These findings will be helpful to design MT systems for spoken dialects without standardized spellings, such as numerous regional languages across Africa or Asia, which are natural means of communication in social media.

8. Acknowledgments

We are grateful to Swisscom for the grant supporting the first author from January to June 2017.

9. Bibliographical References

Ali, A. M., Nakov, P., Bell, P., and Renals, S. (2017). WERd: Using social text spelling variants for evaluating dialectal speech recognition. *arXiv preprint arXiv:1709.07484*.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., Monz, C., Pecina, P., Post, M., Saint-Amand, H., Soricut, R., Specia, L., and Tamchyna, A. (2014). Findings of the 2014 Workshop on Statistical Machine Translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation (WMT)*, pages 12–58, Baltimore, MD, USA.

Bradbury, J., Merity, S., Xiong, C., and Socher, R. (2017). Quasi-recurrent neural networks. In *Proceedings of the Int. Conf. on Learning Representations (ICLR)*, Toulon, France.

Carl, M., Melero, M., Badia, T., Vandeghinste, V., Dirix, P., Schuurman, I., Markantonatou, S., Sofianopoulos, S., Vassiliou, M., and Yannoutsou, O. (2008). METIS-II: low resource machine translation. *Machine Translation*, 22(1):67–99.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Christen, H., Glaser, E., Friedli, M., and Renn, M. (2013). *Kleiner Sprachatlas der deutschen Schweiz*. Verlag Huber, Frauenfeld, Switzerland.

Chung, J., Cho, K., and Bengio, Y. (2016). A character-level decoder without explicit segmentation for neural machine translation. *arXiv preprint arXiv:1603.06147*.

Costa-jussà, M. R. and Fonollosa, J. A. R. (2016). Character-based neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 357–361, Berlin, Germany.

Garner, P. N., Imseng, D., and Meyer, T. (2014). Automatic speech recognition and translation of a Swiss German dialect: Walliserdeutsch. In *Proceedings of Interspeech*, Singapore.

Hildenbrandt, T., Moosmüller, S., and Neubarth, F. (2013). Orthographic encoding of the Viennese dialect for machine translation. In *Proceedings of the 6th Language & Technology Conference (LTC 2013)*, pages 7–9, Poznan, Poland.

Irvine, A. and Callison-Burch, C. (2013). Combining bilingual and comparable corpora for low resource ma-

chine translation. In *Proceedings of the 8th Workshop on Statistical MT*, pages 262–270, Sofia, Bulgaria.

Irvine, A. and Callison-Burch, C. (2016). End-to-end statistical machine translation with zero or small parallel texts. *Natural Language Engineering*, 22(4):517–548.

Klementiev, A., Irvine, A., Callison-Burch, C., and Yarowsky, D. (2012). Toward statistical machine translation without parallel corpora. In *Proceedings of the 13th Conference of the European Chapter of the ACL*, pages 130–140, Avignon, France.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the ACL*, pages 48–54, Edmonton, Canada.

Lee, J., Cho, K., and Hofmann, T. (2017). Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics (TACL)*, 5:365–378.

Lewis, W. D., Munro, R., and Vogel, S. (2011). Crisis MT: Developing a cookbook for MT in crisis situations. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 501–511, Edinburgh, UK.

Lewis, W. D. (2010). Haitian Creole: How to build and ship an MT engine from scratch in 4 days, 17 hours, and 30 minutes. In *Proceedings of the 14th Annual Conference of the European Association for Machine Translation (EAMT)*, Saint-Raphaël, France.

Ling, W., Trancoso, I., Dyer, C., and Black, A. W. (2015). Character-based neural machine translation. *arXiv preprint arXiv:1511.04586*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, PA, USA.

Pourdamghani, N. and Knight, K. (2017). Deciphering related languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2513–2518, Copenhagen, Denmark.

Russ, C. V. J. (1990). High Alemannic. In Charles V. J. Russ, editor, *The Dialects of Modern German. A linguistic survey*, pages 364–393. Routledge, London, UK.

Russ, C. V. J. (1994). *The German language today. A linguistic introduction*. Routledge, London, UK.

Samardžić, T., Scherrer, Y., and Glaser, E. (2015). Normalising orthographic and dialectal variants for the automatic processing of Swiss German. In *Proceedings of the 7th Language and Technology Conference (LTC)*, Poznan, Poland.

Samardžić, T., Scherrer, Y., and Glaser, E. (2016). ArchiMob – a corpus of spoken Swiss German. In *Proceedings of the 10th Int. Conf. on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia.

Scherrer, Y. and Ljubešić, N. (2016). Automatic normalisation of the Swiss German ArchiMob corpus using character-level machine translation. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS)*, Bochum, Germany.

Scherrer, Y. (2012a). *Generating Swiss German sentences from Standard German: a multi-dialectal approach*. Ph.D. thesis, University of Geneva, Switzerland.

Scherrer, Y. (2012b). Machine translation into multiple dialects: The example of Swiss German. In *7th SIDG Congress - Dialect 2.0*, Vienna, Austria.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, volume 1, pages 1715–1725, Berlin, Germany.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O. F., and Callison-Burch, C. (2012). Machine translation of Arabic dialects. In *Proceedings of the 2012 Conference of the North American Chapter of the ACL (NAACL-HLT)*, pages 49–59, Montreal, Canada.

Improving Domain-specific SMT for Low-resourced Languages using Data from Different Domains

Fathima Farhath, Pranavan Theivendiram, Surangika Ranathunga, Sanath Jayasena, Gihan Dias

Department of Computer Science and Engineering
University of Moratuwa
Katubedda 100400, Sri Lanka
{fathimafarhath, pranavan.11, surangika, sanath, gihan}@cse.mrt.ac.lk

Abstract

This paper evaluates the impact of different types of data sources in developing a domain-specific statistical machine translation (SMT) system for the domain of official government letters, between the low-resourced language pair Sinhala and Tamil. The baseline was built with a small in-domain parallel dataset containing official government letters. The translation system was evaluated with two different test data sets. Test data from the same sources as training and tuning gave a higher score due to over-fitting, while the test data from a different source resulted in a considerably lower score. With the motive to improve translation, more data was collected from, (a) different government sources other than official letters (pseudo in-domain), and (b) online sources such as blogs, news and wiki dumps (out-domain). Use of pseudo in-domain data showed an improvement for both the test sets as the language is formal and context was similar to that of the in-domain though the writing style varies. Out-domain data, however, did not give a positive impact, either in filtered or unfiltered forms, as the writing style was different and the context was much more general than that of the official government documents.

Keywords: domain-specific statistical machine translation, low-resourced languages, Sinhala, Tamil, Domain Adaptation

1. Background

Sri Lanka is a multi-ethnic country where Sinhala and Tamil are the official languages. However, only a small number of the population can communicate in both the languages. Therefore, to smoothly carry on the official government communication with the public, dissemination of information has to happen in both the languages. This requires the official government letters to be produced in both the languages. However, this process has become an extra burden to government institutions as they lack professional translators with bilingual knowledge. In order to reduce the burden on the translators to a certain extent, the alternative option is to use the assistance of *machine translation* (MT) in their translation workflow.

For constructing an MT system, prevailing methodologies are *statistical machine translation* (SMT) (Kohen, 2009) and *neural machine translation* (NMT) (Bahdanau, 2014). Though NMT is gradually establishing its root as a viable alternative to SMT ((Bojar et al., 2016), the applicability of it in a low-resource setup is questionable as it requires a large amount of parallel data to boost its gain (Koehn & Knowles, 2017). However, Sinhala and Tamil is a low-resource language pair with minimum linguistic resources and a very little prior research on machine translation between them. Therefore building an NMT system is not a fruitful option (Tennage et al. 2017) whereas an SMT system seems more feasible. However, since SMT systems are also data driven, this requires careful analysis in data selection and utilization in order to get the optimum out of available resources.

The aim of this research is to develop an effective SMT model that can be used to build a translation system for translating official government letters from Sinhala-to-Tamil and vice-versa. However, the size of the parallel dataset that contains official letters (collection contains letters from the Department of Education, Department of Official Languages, and from few regional administrative offices of the government) is relatively small (7,757 parallel sentences). Even collecting this small set of official letters in a ready-to-use form in both the languages required much effort and time due to many institutional problems and resource constraints.

When the system was evaluated using a test set (Test-1) that was randomly picked from the collection of letters from where the training and tuning data are also derived, the *bilingual evaluation understudy* (BLEU) scores (Papineni et al., 2002) of the translation were considerably high. However, when this same system was evaluated using a test set (Test-2) from a different set of letters, from which no data were included into training or tuning, the scores were significantly inferior. The exceptionally high scores in the former case are due to the over-fitting of the *language model* (LM) and *translation model* (TM) to the training data. Test-2 scored lower since a high number of *out of vocabulary* (OOV) words were present in the test data, and less context-related language flow was present in the model.

Since the in-domain parallel data described above was considerably small for this low-resourced domain-specific system, we explored the use of data from other sources that have some relevance and easier to access. Under this, a significant amount of new parallel data was gathered from documents produced by various government institutions. The writing style of these documents slightly differs from that of the letters from which the original in-domain data were derived. Yet, the new data added value to the vocabulary as they have a similar context and terminology to those in official government documents. In literature, this type of data is referred to as 'pseudo in-domain' data (Axelrod et al., 2011). We were able to collect about 15,000 pseudo in-domain parallel sentences, which would allow us to explore domain adaptation (Koehn and Schroeder, 2007).

In addition, a large amount of monolingual data in both Sinhala and Tamil was gathered from online sources such

as blogs, news, and wiki dumps. Since the writing style of these differs from that of official letters and the topics of interest deviate from that of official government letters, we consider this as out-domain data. In literature, when developing domain-adapted systems, using out–domain data is recommended in a filtered form, where sentences are removed from the corpus based on perplexity-based measure differences (Moore and Lewis, 2010). We experimented with out-domain data in our system, by using them in unfiltered (raw form) as well as in filtered form.

Our experimental results indicate that adding pseudo in-domain data to parallel as well to monolingual data contributes to an increase in translation accuracy over both the test sets, while the contribution of the out-domain monolingual data was not significant neither in the filtered form nor unfiltered form.

The rest of the paper is organized as follows. The second section relates on literature on MT, Sinhala – Tamil MT, and commonly used approaches on domain adaptation. Section three describes the data. The fourth section elaborates on the experiments carried out. In section five, we present the evaluation and analysis based on the experiments and results. And finally, the sixth chapter presents the conclusion of the research along with future work.

2. Related Work

2.1 Machine Translation

Today, machine translation is a mature natural language processing task. Successful translation systems have been produced for many language pairs in the world, European languages in particular. These have enabled the use of machine translation in professional translation workflows to enhance the productivity in domains such as medical, automobile, manufacturing and legal (Kohen, 2009).

For more than a decade, SMT has been the fundamental technique for machine translation. However, recently NMT has invaded this position by producing better outputs than SMT (Bahdanau et al. 2014). Both the methodologies are data-driven. Yet, in NMT, the higher performance gain is achieved with greater amounts of parallel data and more computing power (Koehn & Knowles, 2017). This makes NMT less practicable for a low-resource setup.

2.2 Sinhala –Tamil Machine Translation

Sinhala language falls under the family of Indo-Aryan languages, while Tamil belongs to the Dravidian family. When comparing with English, the differences between Sinhala and Tamil are less, where the default sentence structure in both the languages is comprised of subject-object-verb. Yet these two languages are highly inflectional and many differences exist in their syntactic structures.

Both Sinhala and Tamil lack quality linguistic resources (Weerasinghe, 2003). Minimal research is done in building MT systems for this language pair. An initial *phrase-based statistical machine translation* (PBSMT) feasibility study for this language pair was done by Weerasinghe (2003) on 5000+ parallel sentences. To date, the highest BLEU score for the open domain PBSMT is 10.1 for Sinhala to Tamil (Pushpananda et al., 2014), and 13.11 for Tamil to Sinhala (Pushpananda et al., 2015).

The initial NMT approach for this language pair was by Tennage et al. (2017) with a total size of 23,000+ of parallel data (includes training, validation and testing), for the domain of official government documents. The results revealed that the quality of the output of NMT is much worse compared to that of SMT.

.

2.3 Domain Adaptation for Domain-specific SMT

Writing style of a language differs along the genre (e.g. blogs, scientific writing, and legal documents). Moreover, the literal meaning of words and the flow of the language highly depend on it. SMT systems developed for open-domain are not capable of addressing these domain specific variations, as they are trained using general data. The best way to build a domain specific SMT system is to develop a SMT system solely with a large amount of in-domain data. Yet, finding such an amount of in-domain data is practically infeasible in the context of many languges. Domain adaptation (Kohen, 2009) could be used in such situations.

Two major approaches are used in domain adaptation:

1. **Using an open-domain system to fine tune into a specific domain:**

Koehn and Schroeder (2007) suggest the use of cross-domain adaptation. Here, a considerably small amount of in-domain data is being exploited over a considerably large amount of out-domain data using a linear interpolation technique.

Foster and Kuhn (2007) used the concept of mixture modeling (McLachlan and Peel, 2004) to develop dynamic domain adaptation. Here, for different domains, adaptation was done using a cross-domain technique. By analyzing the input text, a mixture model is generated based on an unsupervised clustering method and mixture weights are estimated dynamically. This is an extended version of Koehn and Schroeder's (2007) system, as they cater domain adaptation for multiple domains in one system in a dynamic manner.

Civera and Juan (2007) use mixture modeling in domain adaptation to enhance the word alignments by intervening the alignment process to generate topic-dependent word alignment over general alignment. Yet they doubt on the applicability of this technique as the performance of SMT depends on many factors.

2. **Data Filtration techniques to extract data from open-domain corpus that are similar to the in-domain data**

In order to guarantee that data is from a same or similar domain, different filtration techniques are used in collecting and filtering open-domain monolingual data (Eck et al., 2004), as well as parallel data (Hildebrand et al., 2005).

Data filtration is the process where a given set of data is being processed to remove the less similar sentences from an out-domain corpus with reference to the given in-domain corpus.

One of the measures used for filtration is perplexity (Kohen, 2009). Different techniques are being developed based on this concept to filter parallel as well as monolingual data. This concept is used in SMT domain adaptation with the motive to reduce the influence of highly deviating or less similar sentences.

Gao et al. (2002) suggest the use of a simple perplexity metric of sentences based on the in-domain LM to filter off the sentences that have a perplexity higher than a threshold. Moore and Lewis (2010) convey the idea of using the cross-entropy difference between the in-domain and out-domain LMs as the measure for filtration. They point out that this methodology works better in reducing the perplexity than Gao et al.'s (2002) method. Both these techniques can be applied for parallel as well as monolingual data. Axelrod et al. (2011) suggest the use of bilingual cross-entropy differences, which can only be used for filtering parallel data.

Though these techniques are based on minimizing the perplexity, improvement in the SMT translation quality is not assured (Moore and Lewis 2010, Axelrod et al., 2011), as the behavior of SMT systems depends on multiple factors.

3. Data Set

Gathered data was categorized into three, namely, in-domain, pseudo in-domain and out-domain, based on the context and writing styles.

Data gathered from official letters (e.g., from the Department of Education etc.,) was considered as in-domain.

Since the size of in-domain data was small, additional data was gathered from other government sources such as annual reports, parliament order papers, circulars, and establishment codes. Though these were from government institutions, the writing style was different from letters described above (e.g. the parliament order papers were more like question and answer form), thus these were categorized under pseudo in-domain. A reasonable amount of pseudo in-domain parallel data with respect to the in-domain data was collected. Some source documents of in-domain and pseudo in-domain were hard copies in a single language (i.e., either the Tamil or Sinhala version of the document), while some were soft copies in PDF format. The single-language source documents were manually translated and typed. Data from PDF documents were extracted using a custom developed tool. Parallel data was created by using the sentence alignment tool created by Hameed et al. (2017). To make sure that there are no duplicates in the training, tuning and testing sets, duplicate sentences were removed using a custom script. In addition, we collected some monolingual Tamil sentences of this category, where the sources were annual reports.

Other easily accessible data sources were from the web, (such as articles from blogs, news and wiki dumps), and other free sources. This out-domain data was collected from some freely available sources (Ramasamy et al., 2012, Goldhahn et al., 2012) as well as by web crawling. Yet, the context with respect to official government letters, was quite different. Therefore, these were categorized as out-domain data. However, it was possible only to gather monolingual data under this category. Since we had a comparatively larger amount of out-domain data w.r.t. in-domain data, with the motive to use the data that is more relevant to the context, filtration is done based on perplexity measure (Moore and Lewis, 2010). Here, the extraction of relevance sentences (w.r.t the in-domain corpus) was done based on a threshold value. The difference between the perplexities of the sentences on the in-domain based LM and out-domain based LM is considered. Sentences with this difference greater than the threshold value are considered. The threshold value is dynamically set based on the tuning set perplexities. And this filteration process is iterated multiple times.

Two test sets were prepared for evaluations. One set was a set of sentences randomly picked from the collection from where the training and tuning data were derived (Test-1). The other test set sample (Test-2) was from official letters of an office of a university; no data from these letters were used in training or tuning sets. The average sentence lengths of Test -1 were 10.95, 9.90 and Test-2 were 13.94, 11.21 for Sinhala and Tamil, respectively.

Statistics on the parallel data, Tamil monolingual data and Sinhala monolingual data are shown in Table-1, Table-2, and Table-3, respectively. Moreover, in Table-2 and Table-3, in column 4, the perplexity (lower the perplexity higher the similarity between the reference and sample data (Jelinek et al., 1977)) values of each monolingual data source calculated with respect to the tuning set are listed.

Source	S	W (Sinhala)	W (Tamil)
In-domain	6,428	80,849	73,066
Pseudo in-domain	15,645	237,498	197,271
Tuning	1,000	12,740	11,544
Test-1	340	3,724	3,368
Test-2	340	4,740	3,810

S: # sentences, W: # words

Table 1: Sources of parallel data

Source	S	W (Tamil)	Perplexity
In-domain	6,428	80,849	214.6239
Pseudo in-domain	76,692	788,544	415.5571
Out-domain	1,525,966	21,348,157	2210.4860
Filtered out-domain	14,682	178,840	1814.9250

Table 2: Tamil monolingual data

Source	S	W (Sinhala)	Perplexity
In-domain	6,428	73,066	87.4207
Pseudo in-domain	15,646	237,498	604.4787
Out-domain	4,735,658	72,531,342	918.3833
Filtered out-domain	159,597	2,865,591	518.1778

Table 3: Sinhala monolingual data

4. Experiments

The experimental setup was built using the *Moses* (Kohen et al., 2007) PBSMT system. As the publicly available tokenizers did not work for this pair, we used our tailor-made tokenizer for Sinhala as well as Tamil. Parallel data was filtered using standard *Moses* filtrations to remove misaligned sentences and sentence pairs with high length ratio differences.

To generate the word alignment, Giza++ (Och and Ney, 2003), was used with 'grow-diag-final-and' as the symmetrization heuristic and 'msd-bidirectional-fe' as the reordering technique.

'Good Turing' was used as the smoothing technique for the phrase table score smoothing. In addition to phrase translation score; lexical translation scores, word and phrase penalties, and linear distortion were used as features in the TM, which are commonly used features (Kohen, 2009).

An LM of order 5 (5-gram) was created using *SRILM* (Stolcke, 2002). For the setup with multiple data sources, by experimenting on different configurations (single model with all data, log-linear interpolation of multiple LM and linear interpolation), log-linear interpolation of multiple LM was chosen as it gave the best scores. Here individual LMs were created for each type of source, and were used as individual sub modules under LM by giving individual weights. At the time of tuning, these weights were adjusted (based on the relevance to the tuning set translation).

XenC (Rousseau, 2013) was used to do the filtration over the out-domain data. Two separate LMs were created for in-domain and out-domain using SRILM. These LMs were used in calculating the perplexity difference of each sentence between both the models. This difference between the perplexities is used in determining the eligibility of a sentence to be filtered out (a sentence with this value higher than the threshold are filtered out).

Cube pruning techniques available in Moses were used with a stack size of 5,000, and a maximum phrase length of 5.

The feature weights of each model were tuned using *Minimum Error Rate Training* (MERT) (Och, 2003) on 100 best translations of 1000 sentences / phrases.

The baseline was set up only with the in-domain data (letters) as the TM and LM sources. The pseudo in-domain data was added to TM and LM in a step-wise approach and evaluated. On top of this configuration, the out-domain monolingual data was added in unfiltered and filtered forms one by one separately, and the impact was evaluated. Same set of experiments were carried on in either direction of the language pair.

5. Evaluation & Analysis

Performance of each setup was evaluated using two separate test sets, each containing 340 unique sentences/phrases based on BLEU.

Results for the five different experimental setups for both the test sets are tabulated in Table 4 and Table 5 for Sinhala-to-Tamil and Tamil-to-Sinhala, respectively. The BLEU columns show the BLEU scores and the OOV columns show number of OOV in the translated output which is the unique word count of untranslated words.

Setup A is the baseline where the TM and LM are built using only the in-domain data. In setup B, the pseudo in-domain parallel data is added to the TM of the setup A (baseline). Setup C is extended by adding an extra LM to setup B where this new LM is built using the target side of the pseudo in-domain. In setup D, another LM built using the unfiltered out-domain data is added to the setup C. The LM built with the filtered out-domain data added to setup C is represented by setup E.

In addition, another two sets of results are included into Table 4 and Table 5 as Setup F and setup G.

- Setup F: This is the experimental results reported by Tennage et al. (2018) for NMT for the same data set as setup C. Yet they experiment only using one test set (Test-1) and they have not reported on OOVs (so cells in the last 3 columns are left blank in Table 4 and Table 5 for setup F).
- Setup G: This shows the scores calculated on the output of the Google Translate for both of the test sets that are used in our experiments. Since Google Translate drops the OOVs in its output (in some cases, the entire sentences are dropped) OOV cells are left blank for this setup.

Based on BLEU sores (refer columns 3 and 4 in Tables 4 and 5), the baseline SMT system scores are higher than NMT (setup F) and Google Translate (setup G).

When comparing the result of each test set, there is a drastic difference between them as well as in number of OOV. The first set (Test-1) had an abnormally high range of BLEU scores, while the second set (Test-2) had a much lower range of BLEU scores. This can be explained based on the nature of the test data. Test-1 was a subset from where the training and tuning data were derived, while Test-2 does not intersect with the type of sentences that training and tuning data contains. These contrasting results are because of over-fitting of Test-1. Moreover, there is a score difference noticed for Google Translate as well. This may be due to the average length difference as well as source-target length ratio differences of test sets (Average sentence length of Test-1 is lesser than Test-2's. Source-target sentence length different ratio is higher in Test-2 than Test- 1. This makes translating Test-2 more complex than Test-1).

Based on the results for the TM enhancement, by adding the pseudo in-domain parallel data to the system, all four test results (two in either direction) showed a noticeable increase in the BLEU score. As more parallel data was added, the vocabulary of the system increases and improves the word alignment. This improves the TM, which helps to reduce the OOV in both directions (refer to the last two columns in Table 4 and Table 5). In all 4 test cases, the number of unknown words has reduced in Setup B, with respect to Setup A.

Adding pseudo in-domain data to LM results in a slight improvement in the BLEU scores. Here**,** the score was affected by the improvement in the LM, as the system

learns more on language flow (e.g., out of available options, which translation will suit more in a certain context), and more on inflection options. For example, the word "අවසන් වසර" (/awasan wasara/) in Sinhala means 'final year'. However, the word "අවසන්" (/awasan/) alone can mean 'last', 'final', 'ending', or 'finish'. With a small LM (in-domain only), the system translated this Sinhala word into the Tamil word phrase "நிறைவு ஆண்டு" (/niraiwu aandu/), where the word "நிறைவு" (/niraiwu/) bears the meanings 'ending' or 'finish', which is not appropriate. When pseudo in-domain LM is added, this word got translated to the correct term "இறுதி ஆண்டு" (/iruthi aandu/), where the meaning is 'final year'.

However, the integration of out-domain data reduced the BLEU scores in both filtered and unfiltered cases in either direction. Moreover, the result of filtered data was inferior to that of unfiltered data for all the test cases though the perplexity of the filtered data was lower than that of unfiltered data (refer column 4 in Table 2 and Table 3).

Source	Setup	**BLEU**		**OOV**	
		Test-1	Test-2	Test-1	Test-2
Baseline	A	36.32	7.55	375	645
A + PIS :TM	B	**37.01**	9.13	285	505
B + PIS : LM	C	**37.01**	9.17	285	505
C + UOD:LM	D	36.26	**9.18**	285	505
C + FOD: LM	E	36.06	9.17	285	505
NMT	F	6.78	-	-	-
Google	G	14.49	7.38	-	-

PIS: pseudo in-domain, UOD: unfiltered out-domain, FOD: filtered out-domain

Table 4: Experimental results for Sinhala-to-Tamil

Source	Setup	**BLEU**		**OOV**	
		Test-1	Test-2	Test-1	Test-2
Baseline	A	45.19	12.46	555	937
A + PIS :TM	B	**46.64**	13.15	438	760
B + PIS : LM	C	46.30	**13.17**	438	760
C + UOD:LM	D	46.01	12.46	438	760
C + FOD: LM	E	45.98	11.70	438	760
NMT	F	6.84	-	-	-
Google	G	14.96	7.69	-	-

Table 5: Experimental results for Tamil-to-Sinhala

This kind of behavior has been mentioned by Moore and Lewis (2010) as well. One reason they mentioned is that as the perplexity reduces, more weight is given to the out-domain LM, though still the writing style is drastically different. This can be the reason even in our experiments as well. For example, the phrase "சிக்கலான நிலையை உருவாக்கியுள்ளது." (/sikkalaana nilaiai uruwahiyulathu/) means 'has created a problematic situation', where the word "சிக்கலான" (/sikkalaana/) has meant to be 'problematic' though it can also take the meaning as 'complex', 'issue', or 'conflict'. Without out-domain data, the phrase is translated as "ගැටළු සහගත වී ඇත" (/gatalu sahagatha wee atha/), which is the correct translation. However, when the out-domain data is added, system translates it as "සංකීර්ණ වී ඇත" (/sankeerna wee atha/), which means 'has become complex' which is not the proper translation according to the context.

Perplexity values of the Tamil out-domain corpus (2210.4860) are comparatively higher than that of Sinhala out-domain corpus (918.3833). Tamil out-domain corpus had more blog articles where the writing style was more informal with more colloquial style data. Sinhala out-domain corpus had more news articles where the writing style is less colloquial. This could be the reason for this high variation in perplexity values.

6. Conclusion

This paper presented a domain-specific SMT system for the low-resourced language pair, Sinhala and Tamil. Evaluations were carried on analyzing the impact of different data types in improving the system for domain-specific SMT, as the available in-domain data was minimal and coming from few government institutions. Results show that the use of pseudo in-domain data gave positive results in TM and a less significant improvement for LM. However, the use of out-domain monolingual data did not improve the performance in unfiltered or filtered form, while the filtered data resulted in inferior results to unfiltered data. Further, results obtained with an NMT system as well as from Google Translate highlight the effectiveness of our SMT work.

This conveys the message that for our domain of consideration, data from sources that have informal writing such as news and blogs will not add value. Furthermore, the results reveal that the system requires quality data in higher quantities from diverse subject matters and sources (e.g., numerous government institutions), to perform better.

In future, with more data, we plan to experiment on dynamic model adaptation based on the context of the letters, as suggested by Foster and Kuhn (2007). We hope that this will help to get a fine-tuned system that can be used for better translation of letters from different government institutions based on the context.

7. Acknowledgement

The authors of this paper would like to extend their gratitude to the Senate Research Council (SRC) Grant of University of Moratuwa and the Department of Official Languages (DOL) for the funding provided for the research, and the individuals who contributed in data gathering and corpus preparation.

8. Bibliographical References

Axelrod, A., He, X. and Gao, J., (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the conference on empirical methods in natural language processing*, pages 355-362. Association for Computational Linguistics.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A.J., Koehn, P., Logacheva, V., Monz, C. and Negri, M., (2016). Findings of the 2016 Conference on Machine Translation. In *ACL 2016 first conference on machine translation (WMT16)* pp. 131-198. The Association for Computational Linguistics.

Civera, J., & Juan, A. (2007). Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 177-180. Association for Computational Linguistics.

Eck, M., Vogel, S., & Waibel, A. (2004). Language Model Adaptation for Statistical Machine Translation Based on Information Retrieval. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, pages 327–330.

Foster, G., & Kuhn, R. (2007). Mixture-model adaptation for SMT. In *Proceedings of the second workshop on statistical machine translation*, pages 128-135. Association for Computational Linguistics.

Gao, J., Goodman, J., Li, M., & Lee, K. F. (2002). Toward a unified approach to statistical language modeling for Chinese. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(1), 3-33.

Goldhahn, D., Eckart, T., & Quasthoff, U. (2012). Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)*, pages 759-765.

Hameed, R. A., Pathirennehelage, N., Ihalapathirana, A., Mohamed, M. Z., Ranathunga, S., Jayasena, S., Dias, G., & Fernando, S. (2016). Automatic Creation of a Sentence Aligned Sinhala-Tamil Parallel Corpus. In *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP)*, pages 124-132

Hildebrand, A. S., Eck, M., Vogel, S., & Waibel, A. (2005). Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of European Association for Machine Translation (EAMT)*, Vol. 2005, pages 133-142.

Jelinek, F., Mercer, R. L., Bahl, L. R., & Baker, J. K. (1977). Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1), S63-S63.

Koehn, P., & Schroeder, J. (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the second workshop on statistical machine translation*, pages 224-227. Association for Computational Linguistics.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 177-180. Association for Computational Linguistics.

Koehn, P. (2009). Statistical machine translation. Cambridge University Press.

Koehn, P., & Knowles, R., (2017). Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*.

McLachlan, G., & Peel, D. (2004). Finite mixture models. John Wiley & Sons.

Moore, R. C., & Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference short papers*, pages 220-224. Association for Computational Linguistics.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Volume 1 pages 160-167. Association for Computational Linguistics.

Och, F. J., & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1), 19-51.

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311-318. Association for Computational Linguistics.

Pushpananda, R., Weerasinghe, R., & Niranjan, M. (2014). Sinhala-Tamil Machine Translation: Towards better Translation Quality. In *Australasian Language Technology Association Workshop 2015*, pages 123-133.

Pushpananda, R., Weerasinghe, R., & Niranjan, M. (2015, April). Statistical Machine Translation from and into Morphologically Rich and Low Resourced Languages. In *Conference on Computational Linguistics and Natural Language Processing (CICLing)*, pages 545-556. Springer.

Ramasamy, L., Bojar, O., & Žabokrtský, Z. (2012). Morphological processing for English-Tamil statistical machine translation. In *Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages*, pages 113-122.

Rousseau, A. (2013). Xenc: An open-source tool for data selection in natural language processing. *The Prague Bulletin of Mathematical Linguistics*, 100(1):73–82.

Stolcke, A. (2002). SRILM-an extensible language modeling toolkit. In *International Conference on Spoken Language Processing*, pages 901– 904.

Tennage, P., Sandaruwan, P., Thilakarathne, M., Herath, A., Ranathunga, S., Jayasena, S., and Dias, G. (2017). Neural Machine Translation for Sinhala and Tamil Languages. In *Proceeding of 21st International Conference on Asian Language Processing (IALP)*.

Tennage, P., Sandaruwan, P., Thilakarathne, M., Herath, A., and Ranathunga, S. (2018). Handling Rare Word Problem using Synthetic Training Data for Sinhala and Tamil Neural Machine Translation. *11th International Conference on Language Resources and Evaluation (LREC)*.

Weerasinghe, R. (2003). A statistical machine translation approach to Sinhala-Tamil language translation. *Towards an ICT enabled Society*, pages 136-141.

Discovering Parallel Language Resources for Training MT Engines

Vassilis Papavassiliou, Prokopis Prokopidis, Stelios Piperidis
Institute for Language and Speech Processing
Athena Research and Innovation Center, Athens, Greece
{vpapa, prokopis, spip}@ilsp.gr

Abstract

Web crawling is an efficient way for compiling the monolingual, parallel and/or domain-specific corpora needed for machine translation and other HLT applications. These corpora can be automatically processed to generate second order or synthesized derivative resources, including bilingual (general or domain-specific) lexica and terminology lists. In this submission, we discuss the architecture and use of the ILSP Focused Crawler (ILSP-FC), a system developed by researchers of the ILSP/Athena RIC for the acquisition of such resources, and currently being used through the European Language Resource Coordination effort. ELRC aims to identify and gather language and translation data relevant to public services and governmental institutions across 30 European countries participating in the Connecting Europe Facility (CEF).

Keywords: parallel language resources, document pair detection, web crawling, machine translation

1. ILSP-FC Architecture

ILSP-FC[1] is a comprehensive solution for acquiring domain-specific, monolingual and bilingual datasets from the web. One of the main components of the system is an efficient crawler that initializes its frontier (i.e. the list of pages to be visited) from a seed URL list provided by the user, classifies fetched pages as appropriate for the user's aims (i.e. in the targeted language and/or relevant to the targeted domain), extracts links from fetched web pages, adds them to the list of pages to be visited and repeats this process until an expiration criterion is reached. In the case of focused crawls for domain-specific content, the input expected from the user also includes a domain profile, i.e. a list of terms that describe the domain.

In order to ensure scalability, the system is based on open-source libraries that allow configuration of workflows that can be executed on top of the Hadoop framework for distributed data processing. Due to its modular architecture, each of the system's components can be easily substituted by alternatives with the same functionalities. The main components integrated in ILSP-FC (Papavassiliou et al., 2013) are:

Page Fetcher adopts a multi-threaded crawling implementation in order to ensure concurrent visiting of multiple web pages/hosts and fetching of user-targeted specific document types (e.g. html, docx, pdf).

Normalizer detects the text encoding of the downloaded web pages and if needed, converts it to UTF-8. It also parses the structure of each web page and extracts its metadata (e.g. title, description, keywords, publisher, author, license etc.). In order to extract textual content and metadata from a set of formats (txt, docx and pdf), it uses open libraries[2].

Cleaner segments the main text in paragraphs, identifies boilerplate (e.g. advertisements, disclaimers) and extracts structural information like titles, headings and list items. For this task, a modified version of Boilerpipe (Kohlschütter et al., 2010) is used.

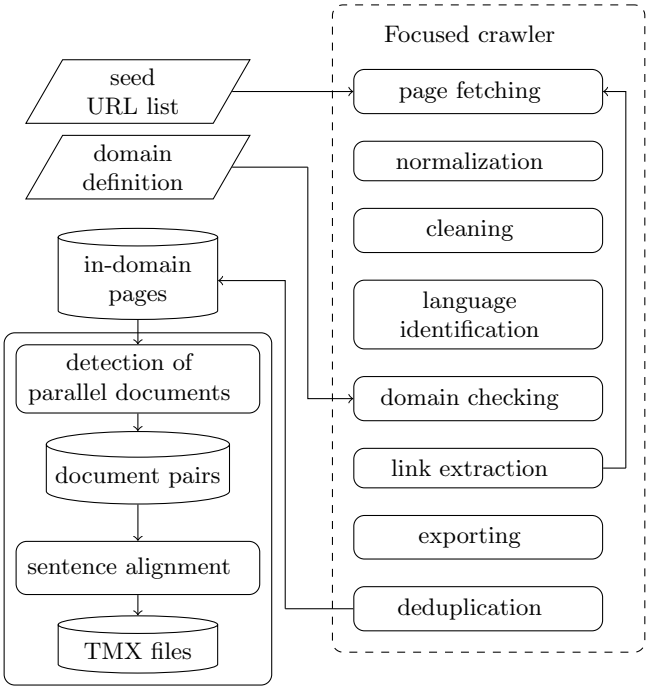


Figure 1: ILSP-FC workflow for acquisition of parallel LRs

Language Identifier detects the language of a document, as well as paragraphs in a language different from the main one. It uses open source language identification libraries like `lang-detect`[3], that perform at over 99% precision at document level for more than 50 languages.

Domain checker compares the content of the page to a user-provided domain profile. Based on the number of terms' occurrences, their location in the web page and the weights of found terms, a page relevance score is calculated. This score is compared with a predefined threshold and the web page is categorized as relevant to the domain.

Link Extractor examines the "neighbourhood" of links for terms, special patterns and/or semantic annotations (e.g.

[1] `http://nlp.ilsp.gr/ilsp-fc/`

[2] `https://pdfbox.apache.org/`, `https://poi.apache.org/text-extraction.html`

[3] `https://github.com/shuyo/language-detection`

link[hreflang], a[id=linkTranslateVersion], etc.), and ranks links by the probability that they point to candidate translations and/or in domain pages. Thus it organizes the list of URLs to be fetched as a priority queue and guides the crawler to visit "interesting" pages first.

De-duplicator checks each document against all others and identifies (near) duplicates based on lists of quantised word frequencies extracted from each document; and on the percentage of common paragraphs (e.g. 80%). In case a pair of (near) duplicates has been detected, the shortest is discarded.

Pair Detector (a.k.a. bitext identification module) identifies documents that could be considered parallel. This module exploits the results of online discovery and prioritization of translation links during crawling, and uses alternative criteria based on special patterns included in the URLs, cooccurrences of images with the same filename, similar sequences of digits, and structure similarity. It does not use any knowledge of the targeted languages (e.g. bilingual lexica or MT output) but applies language independent methods for pair detection.

Sentence Aligner uses open-source aligners (e.g. Hunalign (Varga et al., 2005), Maligna (Jassem and Lipski, 2008)) to extract sentence alignments from document pairs, and exports results in TMX (i.e. a TMX file for each identified document pair).

As an alternative to the pipeline use of the tool for acquisition of parallel LRs from the web (cf. Figure 1), specific modules (e.g. domain checking, document pairing, and sentence alignment) can be called as standalone modules for all relative tasks. They can therefore be used for the processing of resources residing in proprietary data repositories. ILSP-FC is available under a GPL license. Licensing and support alternatives for commercial uses and applications are also available.

2. Adapting to ELRC requirements for parallel LRs

The European Language Resource Coordination[4] (Lösch et al., 2018) effort aims to identify and gather language and translation data relevant to public services and governmental institutions across 30 European countries participating in the Connecting Europe Facility[5]. Through a series of workshops, ELRC has been showcasing the benefits of the CEF automated translation platform (CEF eTranslation) and has been trying to mobilize public sector representatives to share public language resources. These LRs will contribute in enhancing CEF eTranslation and, in the end, providing EU citizens with better multilingual services. To complement these data gathering efforts, ELRC has also employed ILSP-FC and other automatic methods for acquiring multilingual LRs. In the course of the project and based on feedback by all consortium partners, the tool was continuously tested and enhanced at ILSP in order to provide more accurate results. It was eventually deployed at all four ELRC partner sites for acquiring language resources for specific (EN-X) language pairs, where X stands for official EU languages in CEF-affiliated countries[6].

In order to meet ELRC needs to cover all CEF languages, missing resources were constructed and integrated in the tool (for example language profiles for both Norwegian written standards, Bokmål and Nynorsk). The accuracy of the language identifier, when examining Norwegian texts is 98%/90% for text chunks of at least 500/100 characters, respectively.

Turning to the identification of bitexts, the tool employs a combination of methods that are language-pair agnostic, i.e. they do not use bilingual lexica or MT results that are often difficult to generate. For evaluation purposes, the bitext identification module was submitted (Papavassiliou et al., 2016) in the WMT 2016 Bilingual Document Alignment Shared Task (Buck and Koehn, 2016) and scored a high recall of 91%. It ranked 7th among 21 participations, and, to the best of our knowledge, first among those not using language-pair specific resources or MT output as a feature for document alignment.

For segment alignment, the system uses open source aligners to construct collections of candidate parallel segments. A battery of criteria are applied on these candidates with the purpose of filtering out or annotating automatically specific types of translation units (TUs) (i.e. detecting potential alignment or translation issues, or sentence pairs of limited or no use) and of generating precision-high multilingual LRs for training MT systems. This filtering was carried out by:

(i) identifying duplicate TUs, or TUs with identical text in both languages,

(ii) estimating the alignment quality by calculating the so-called alignment score,

(iii) excluding TUs based on the segment length ratio (note: segments with a length ratio close to 1 have similar length, whereas segments having a ratio far from 1 would indicate potential alignment problems)

(iv) identifying TUs in which numbers in the segment for one language are different compared to the segment for the other language,

(v) excluding TUs that are included in document pairs that contain many TUs (e.g. over 40%) of type 0:1, an indication that such document pairs consist of comparable rather than parallel documents,

(vi) excluding TUs consisting solely of URLs, emails, and dates.

[4] http://lr-coordination.eu

[5] https://ec.europa.eu/inea/en/connecting-europe-facility

[6] Responsibility for data collection was shared among all ELRC consortium members in the following way: DFKI (6 countries): Germany, Austria, Luxembourg, Netherlands, Hungary, Czech Republic; ELDA (8 countries): Ireland, Spain, Portugal, Belgium, Italy, Malta, France, (U.K.); Tilde (8 countries): Latvia, Estonia, Lithuania, Finland, Sweden, Denmark, Iceland, Norway; ILSP (8 countries): Greece, Cyprus, Slovakia, Slovenia, Bulgaria, Poland, Romania, Croatia

lr-id	lr-name	"good"	"bad"
115	Parallel corpus (Greek - English) in the public administration domain	12332 (98.59%)	177 (1.41%)
379	Parallel corpus (Bulgarian - English) in the public administration domain	11094 (98.51%)	168 (1.49%)

Table 1: Automatic evaluation of parallel sentences contained in two LRs generated with ILSP-FC

The optional domain checking procedure during crawling can be complemented by the use of post-crawling topic classification with tools like the JRC Eurovoc Indexer (JEX (Steinberger et al., 2012)). We have used JEX to tag multilingual documents with identifiers that correspond to domains, micro-thesauri and thesauri concepts from Eurovoc[7], the EU's multilingual thesaurus.

3. Construction of parallel LRs out of public sector data

One of the methods employed for the construction of parallel LRs in the ELRC project, was to identify public administration websites (e.g. websites of ministries, local authorities, embassies, courts, etc.) as candidate sources for extracting content relevant to the CEF Digital Service Infrastructures (DSIs)[8]. Then ELRC consortium partners used ILSP-FC to crawl these websites and process their content in order to construct a mono/bilingual collection for each website. Finally, the outcomes of the websites were merged based on the language and the relevance of their content. For instance, the acquired content from websites of Polish cultural organizations was merged to generate a monolingual LR of 10.2M tokens and a parallel EN-PL LR of 36.3K TUs.

Although crawling with the tool for the purposes of ELRC is ongoing work, parallel LRs for several language pairs have already been generated and a number of them (EN-EL, EN-BG, EN-MT, EN-SV, EN-IS, EN-ET, EN-ES) have become available through the ELRC repository[9] (Piperidis et al., 2018) by consortium members.

4. Evaluation experiments

Assessing the usefulness of a system that discovers, acquires and transforms bitexts from the web involves many different evaluation questions: Does the system identify most of the document pairs published on a web site? Are these document pairs as noise-free as possible? Are the aligned sentences extracted from the document pairs clean enough for training MT systems? Apart from our participation in the WMT 2016 shared task, we also conducted experiments covering a variety of language pairs, in order to evaluate both the acquisition procedure and the acquired resources, focusing on parallelness (at document and segment level) and domainess.

In an effort to evaluate recall for the pair detector module, we used parallel datasets crawled from the Global Voices group of websites (Prokopidis et al., 2016). In this dataset, documents are connected with specific links when one is the translation of the other. Since this is not the case for many multilingual web sites, during the evaluation of the pair detector we omitted the tool's methods that exploit special patterns in URLs and links that point to translations. Thus, we only used methods that are based on a) cooccurrences of images with the same filename in HTML source, b) edit distance of sequences of digits in the main content of webpages and c) structural similarity. We evaluated these methods in the task of reconstructing the English-Greek parallel collection, i.e. of identifying the 3581 document pairs for this language pair. The recall and precision rates were 68.56% and 92.50% respectively. The main reason for the relatively low document-level recall is that many document pairs consisting of very short documents were not identified. However, the recall at token level (i.e. the percentage of tokens retrieved from all translated sentences) was 91.18%, a fact that implies that "lost" bitexts contained less that 9% of the actual parallel content.

In another evaluation experiment, we automatically estimated the quality of the sentence-level parallelness of the LRs created with ILSP-FC. To this end, we trained the C-Eval (Zariņa et al., 2015) parallel corpora cleaning and evaluation tool on the DGT-TM 2015 release[10] to build an automatic classifier identifying non-parallel sentences in a parallel corpus. We then classified the parallel sentences contained in two datasets delivered as ELRC LRs #115 (EN-EL) and #379 (EN-BG). Results in Table 1 indicate that LRs created with the tool include a high percentage ($\approx$ 98.5% for these specific LRs) of useful translation segments.

In order to test domainess, we used JEX to assign Eurovoc identifiers on the English text of LRs #115 and #379. In the results shown in Table 2, the assigned identifiers seem to correctly depict the "nature" of #155, for which about 47% and 29% of its content was acquired from the websites of the Greek Ministry of National Defence and the Ministry of Foreign Affairs respectively. Similarly, about 85% of the sentence pairs in #379 were compiled from the websites of the Ministry of Foreign Affairs, and the President of the Republic of Bulgaria. We also examined two parallel collections (EN-HR and EN-PL) in the *culture* domain. In the results in Table 3 only one identifier (*approximation of laws*) seems irrelevant to the targeted domain.

[7] `http://eurovoc.europa.eu/`

[8] CEF DSIs include Online Dispute Resolution; Europeana; Open Data Portal; eJustice; and Electronic Exchange of Social Security Information

[9] `https://elrc-share.eu`

[10] `https://ec.europa.eu/jrc/en/language-technologies/dgt-translation-memory`

id	label	weight
1182	Greece	0,324
5335	military intervention	0,189
3489	cooperation policy	0,186
2628	armed forces	0,184
5786	military personnel	0,183
2499	European defence policy	0,176

id	label	weight
5063	Bulgaria	0,461
3763	Romania	0,295
3489	cooperation policy	0,211
914	Eastern Europe	0,207
12	accession to the European Union	0,169
1474	EC agreement	0,164

Table 2: Eurovoc identifiers assigned automatically by JEX for LRs #115 (EN-EL, top) and #379 (EN-BG, bottom)

id	label	weight
2023	music	0,217
2543	Poland	0,19
2459	cultural policy	0,157
208	cultural cooperation	0,143
529	copyright	0,138
2897	approximation of laws	0,109

id	label	weight
5563	Croatia	0,199
208	cultural cooperation	0,151
2840	heritage protection	0,143
2459	cultural policy	0,142
3200	cultural relations	0,135
4877	cultural object	0,125

Table 3: Eurovoc identifiers assigned automatically by JEX for an EN-PL (top) and an EN-HR (bottom) LR in the culture domain

5. Acknowledgements

This work has been supported by the European Language Resource Coordination (ELRC), a service contract that operated under the EC's Connecting Europe Facility SMART 2014/1074 programme from April 2015 to April 2017. ELRC continues until December 2019 under SMART 2015/1091 LOT 2 "Language Resource coordination and collection with related legal and technical work" and SMART 2015/1091 LOT 3 "Acquisition of additional Language Resources and related refinement/processing services and their provision of the Language Resource Repository of CEF Automated Translation Platform". We would like to thank ELRC consortium members from DFKI, TILDE, and ELDA, who provided feedback, reported issues and used the system described in this paper to acquire parallel resources for the purposes of the contract. Access to the Okeanos cloud service[11] provided by the Greek Research & Technology Network, is greatly appreciated.

[11] https://okeanos.grnet.gr/

6. Bibliographical References

Buck, C. and Koehn, P. (2016). Findings of the WMT 2016 Bilingual Document Alignment Shared Task. In *Proceedings of the First Conference on Machine Translation*, pages 554–563, Berlin, Germany.

Jassem, K. and Lipski, J. (2008). A new tool for the bilingual text aligning at the sentence level. In *Proceedings of Intelligent Information Systems Conference*, Zakopane, Poland.

Kohlschütter, C., Fankhauser, P., and Nejdl, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pages 441–450, New York, USA.

Lösch, A., Mapelli, V., Piperidis, S., Vasiļjevs, A., Smal, L., Declerck, T., Schnur, E., Choukri, K., and van Genabith, J. (2018). European Language Resource Coordination: Collecting Language Resources for Public Sector Multilingual Information Management. In *Proceedings of the 11th Language Resources and Evaluation Conference*, Miyazaki, Japan.

Papavassiliou, V., Prokopidis, P., and Thurmair, G. (2013). A modular open-source focused crawler for mining monolingual and bilingual corpora from the web. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 43–51.

Papavassiliou, V., Prokopidis, P., and Piperidis, S. (2016). The ILSP/ARC submission to the WMT 2016 Bilingual Document Alignment Shared Task. In *Proceedings of the First Conference on Machine Translation*, pages 733–739, Berlin, Germany.

Piperidis, S., Labropoulou, P., Deligiannis, M., and Giagkou, M. (2018). Managing public sector data for multilingual applications development. In *Proceedings of the 11th Language Resources and Evaluation Conference*, Miyazaki, Japan.

Prokopidis, P., Papavassiliou, V., and Piperidis, S. (2016). Parallel Global Voices: a Collection of Multilingual Corpora with Citizen Media Stories. In *Proceedings of the 10th Language Resources and Evaluation Conference*, Portorož, Slovenia.

Steinberger, R., Ebrahim, M., and Turchi, M. (2012). JRC Eurovoc Indexer JEX - A freely available multi-label categorisation tool. In *Proceedings of the 8th Language Resources and Evaluation Conference*, pages 798–805, Istanbul, Turkey.

Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V., and Nagy, V. (2005). Parallel corpora for medium density languages. In *Recent Advances in Natural Language Processing*, pages 590–596, Borovets, Bulgaria.

Zariņa, I., Ņikiforovs, P., and Skadiņš, R. (2015). Word alignment based parallel corpora evaluation and cleaning using machine learning techniques. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 185–192, Antalya, Turkey.

A Fine-grained Error Analysis of NMT, PBMT and RBMT Output for English-to-Dutch

Laura Van Brussel, Arda Tezcan, Lieve Macken
Ghent University, Department of Translation, Interpreting and Communication
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
laura.vanbrussel@ugent.be, arda.tezcan@ugent.be, lieve.macken@ugent.be

Abstract

This paper presents a fine-grained error comparison of the English-to-Dutch translations of a commercial neural, phrase-based and rule-based machine translation (MT) system. For phrase-based and rule-based machine translation, we make use of the annotated SCATE corpus of MT errors, enriching it with the annotation of neural MT errors and updating the SCATE error taxonomy to fit the neural MT output as well. Neural, in general, outperforms phrase-based and rule-based systems especially for fluency, except for lexical issues. On the accuracy level, the improvements are less obvious. The target sentence does not always contain traces or clues of content being missing (omissions). This has repercussions for quality estimation or gisting operating only on the monolingual level. Mistranslations are part of another well represented error category, comprising a high number of word-sense disambiguation errors and a variety of other mistranslation errors, making it more complex to annotate or post-edit.

Keywords: machine translation, error classification, bilingual corpus

1. Introduction

Since 2016, the landscape of automated translation has substantially changed with the arrival of neural machine translation (NMT). The output quality of this newest system is a hot topic for research at the moment. It has already been compared with the previous state-of-the-art phrase-based machine translation (PBMT) engines and even with rule-based machine translation (RBMT) engines, focusing on the overall performance by applying various automatic metrics, by manual ranking and scoring (Shterionov, Casanellas, Superbo, & O'Dowd, 2017), post-editing or manual error classification (Bentivogli, Bisazza, Cettolo, & Federico, 2016). The scope of the studies range from one to multiple language directions (Toral and Sánchez-Cartagena 2017; Klubička, Toral, and Sánchez-Cartagena 2017; Bojar et al. 2016). Unlike previous work, where engines are developed in research institutes or test suites are built for evaluation, in this paper, we take a different angle by using commercial MT systems and real-life texts from different genres, and thus bring more ecological validity into the field.

In this article, we compare the output of commercial NMT, PBMT and RBMT systems for English to Dutch. Since it provides a detailed overview of the types of errors, we want to discover if the findings for other language pairs apply to English-to-Dutch as well, identify the actual improvements that NMT systems bring to automated translation and get a grip on their potential shortcomings.

2. Related Work

This analysis is carried out in the framework of the SCATE project (Tezcan, Hoste, & Macken, 2017b) and draws on its corpus of PBMT and RBMT errors. We used SCATE's error taxonomy to annotate the same sentences, this time translated by Google's Neural Machine Translation (GNMT)[1].

A substantial part of the research in the field focuses on the language pair English-German. For English to German, Bentivogli et al. (2016) found that NMT output contains less lexical, morphological and word-order errors, which leads to a lower overall post-editing effort. However, according to the authors, the performance of NMT degraded more quickly for longer sentences.

Popović (2017) looked into both the overall performance and the specific language-related issues for German-English, using the output of the best NMT and a PBMT engine which participates in the WMT 2016 shared news translation task. The BLEU score and ChrF-score[2] for NMT were higher than for the PBMT output in both language directions. She manually annotated a subset of 264 sentences for English-to-German and 204 for German-to-English extracted from the total corpus of 3000 sentences. In her study, the number of correct sentences was remarkably higher for the NMT system than for the PBMT system. As for the language-specific issues, NMT outperformed the PBMT system in terms of verb aspects (form, order and omission), articles, English noun collocations and German compounds, as well as phrase structure. This led to improved fluency. Burchardt et al. (2017) use a test suite drawn from grammatical resources, and online lists, consisting of typical translation errors, to compare the output of different NMT, PBMT and RBMT systems. This very controlled, difficulty-isolating method, showed a higher intra-system output variation among NMT systems. They also found that NMT scores best on composition, function words, long-distance dependency, multiword expressions, subordination and verb valence. Ambiguity, tense and mood of verbs, on the other hand, are handled best by RBMT systems. Terminology and named entities, finally, form the mainstay of PBMT systems based on their results. By using a similar challenge-set approach, Isabelle, Cherry, and Foster (2017) focus on short sentences that contain one particular language phenomenon at a time, which reveals the strengths and weaknesses of NMT compared to PBMT for English to French. The controlled input in both studies is both a strength and a trade-off for ecological validity. Language-specific errors hardly ever occur in isolation. The performance of systems can differ if multiple difficulties need to be handled in the same sentence.

[1] MT output generated in June 2017.
[2] Character n-gram F-score (Popovic, 2015)

Toral and Sánchez-Cartagena (2017), Bojar et al. (2016) and Castilho et al. (2017) take more language directions into account to evaluate and compare NMT and PBMT.

The MT systems involved in the news-shared task at WMT 2016 (Bojar et al., 2016) covered the language pairs English to German, Czech, Russian, Finnish, Romanian and Turkish. Thanks to a combination of BLEU scores and human ranking, the output of the best systems could be determined, listing NMT (more specifically the engine submitted by the University of Edinburgh) on top in most language directions or in second place except for English-Finnish. Toral and Sánchez-Cartagena (2017) took the output of the best NMT and PBMT systems from the news translation task of WMT16 as a starting point. They have proven that for all language pairs, there is a higher intersystem variability for NMT output and that the NMT output was more fluent. Furthermore, NMT generates more (correct) word reorderings for almost all language pairs (not the case for EN-DE and EN-FI). A negative correlation between sentence length and performance was confirmed for the majority of language directions. For sentences longer than 40 words, the PBMT systems even outperformed NMT, which they ascribe to the sub-word unit operating level of NMT. Finally, NMT performs better on inflection and reordering for all language directions.

A word of caution has been added by Castilho et al. (2017). For English-German, English-Portuguese, English-Greek and English-Russian professional translators were asked to perform three tasks: post-editing, annotating and ranking the PBMT and NMT output. Although for all language pairs, NMT was ranked as most fluent, NMT produced more correct sentences, contained fewer inflectional and word order errors, and needed less effective post-edits, "the progress is not always evident" they warn. The participants indicated that NMT errors were more difficult to identify, compared to the obvious word-order errors and disfluencies occurring in PBMT output. This is attributable to the higher omission, addition and mistranslation rates for NMT (as opposed to PBMT) in some language directions. They concluded that the throughput and temporal effort only marginally improved thanks to NMT.

3. Error classification

For the annotation and classification of the MT errors, we made use of the SCATE error taxonomy (Tezcan et al., 2017b), which differentiates between fluency (assessing the well-formedness of the target language) and accuracy errors (concerning the transfer of source content).

For more information on the annotation guidelines and process, we would refer to Tezcan, Hoste, and Macken (2017). The advantage of the SCATE annotation method is that both fluency and accuracy errors are annotated separately and that the erroneous MT section is linked to the source section in the case of accuracy errors (except for omissions and additions, which are only labelled in source and target, respectively). The categories in the classification are based on MT-specific errors. A text span can receive multiple labels if different types of errors occur in this span. The fine-grained MT error annotations of SCATE serve as training material to develop Quality Estimation systems for MT (Tezcan, Hoste, & Macken, submitted, 2017a).

For the annotation of NMT, we added two extra categories: i) 'fluency-grammar-extra-repetition' (see section 5.2.2), and ii) 'accuracy-mistranslation-semantically unrelated' (see section 5.1.1). The existing annotations for 'fluency-grammar-extra' in the RBMT and PBMT subsets were revised in case the new category suited the output better. Accordingly, all subsets now bare the same updated annotation labels, allowing for a fair comparison.

4. Research setup

4.1 Data Sets

The SCATE corpus of MT errors was built with sentences extracted from the Dutch Parallel Corpus (Macken, De Clercq, & Paulussen, 2011). From this balanced, commercial and copy-right-cleared corpus, an equal number of 665 sentences was selected from three different text types (non-fiction, external communication and journalistic texts[3]).

4.2 MT systems

For the SCATE corpus of MT errors, created in 2014, Systran[4] was used as RBMT system and Google Translate as PBMT system. Around the beginning of October 2016, Google switched to neural, launching Google's Neural Machine Translation system. The architecture of this model consists of deep Long Short-Term Memory recurrent neural networks (LSTM RNNs) with eight encoder and eight decoder layers that use residual connections and attention connections (Wu et al., 2016).

4.3 Annotations

A total of six annotators (all with a linguistic background) worked on this project. For the PBMT and RBMT subsets, two pairs annotated in parallel in June 2014; in June 2016, one pair was assigned this task for NMT. For the actual annotation process, the brat rapid annotation[5] tool was used. To ensure consistency and a higher inter-annotator agreement, annotation guidelines and a reference translation were provided, together with periodic revision moments for questions and answers.

5. Error analysis

A quick glimpse at the overall error statistics in Table 1 reveals that also for English-Dutch, NMT makes fewer mistakes and generates more sentences that are completely correct.

	RBMT	PBMT	NMT
Accuracy	1309	741	472
Fluency	1831	1531	719
Total	**3140**	**2272**	**1191**

Table 1: Total number of errors

[3] The Dutch Parallel Corpus comprises two additional text types that were not used by SCATE: administrative texts and instructive texts.

[4] Systran Enterprise Edition, version 7.5

[5] http://brat.nlplab.org/

	RBMT	PBMT	NMT
Correct sentences	81	130	217
In %	12%	20%	33%

Table 2 : Correct sentences in MT output

Table 2 illustrates that the NMT output surpasses the other systems. One third of the sentences has been translated correctly by NMT, while this rate is much lower for RBMT and PBMT.

5.1 Accuracy errors

Accuracy concerns the transfer of information and meaning from source to target language. The following main categories can be distinguished: mistranslation, do-not-translate (DNT), untranslated, addition, omission, and mechanical.[6] 'Mistranslations' comprise all errors for which the source content has been translated incorrectly (the subcategories will be mentioned below). The label 'DNT' is used for instances in which one or more source words have been translated unnecessarily, e.g. for proper names. 'Addition' refers to errors in which the target content is not present in the source, while for 'omission' some source content is absent in the target sentence. All mistakes concerning non-meaning (mostly punctuation errors only visible on bilingual level) fall under the category 'mechanical'.

Accuracy errors	RBMT	PBMT	NMT
Mistranslation	972	483	330
DNT	116	14	22
Untranslated	65	69	44
Addition	61	39	2
Omission	43	115	62
Mechanical	52	21	12
Total	**1309**	**741**	**472**

Table 3: Overview of the number of accuracy errors

Table 3 shows that overall, NMT scores better on accuracy than previous systems. However, upon closer inspection, it becomes evident that PBMT handles DNT issues better than NMT. This comes as no surprise, since most of the DNT errors are instances of proper names, a reported strength of PBMT (Burchardt et al., 2017). We also observe that RMBT output contains the fewest omissions. The main category 'mistranslation' is obviously a tough nut to crack for automated translation, as it is the category with the highest number of accuracy errors in all three systems, urging us to dig a little deeper.

5.1.1 Mistranslation errors

'Mistranslation' refers to incorrectly translated source content and is subdivided in the following subcategories: multiword expressions (MWE), part of speech (POS), sense, partial and other. The label 'partial' is used for partial translations of verbs (especially for Dutch, separable verbs). The container 'other' comprises mistranslations of the verb tense and voice, or the number (noun/ verb). To cover the instances for which the target word(s) could never be a plausible translation of the given source word, we introduce the label "semantically unrelated". An example:

> EN: … to build the first ever dynamic billboard to **grace** the streets of Glasgow.
>
> NL: … om het eerste dynamische billboard te bouwen om de straten van Glasgow te **grazen**.

In the sentence above 'grace' is translated by 'grazen', the Dutch equivalent of 'to graze'. This new category reveals a high number of semantically unrelated mistranslations in the NMT output, an error that does not occur in RBMT and only rarely in PBMT output.

	RBMT	PBMT	NMT
MWE	288	139	87
POS	52	44	19
Sense	580	208	117
Partial	4	41	6
Semantically Unrelated	0	9	44
Other	48	42	57
Total	**972**	**483**	**330**

Table 4 : Differentiation of mistranslation errors

Table 4 further illustrates the improvement that NMT has made on almost all mistranslation categories, except for 'other'.

5.1.2 Omissions

Castilho et al. (2017) reported the problem of omissions in NMT output. When scanning the error statistics in Table 5, we can see that also in our data set, NMT makes fewer omission errors than PBMT. However, the ratio of omitted words per omission error is much higher in NMT than in PBMT and RBMT.

	# Omissions	# Annotated words	Average # words per omission
RBMT	43	46	1,07
PBMT	115	125	1,09
NMT	62	93	1,50

Table 5: The number of omission errors compared to the number of words per omission error

Looking back at the corpus, we see that the nature of the omission errors has changed. Often, the NMT output does not provide any clues that source content has been omitted. Wu et al. (2016) already commented: "MT systems sometimes produce output sentences that do not translate all parts of the input sentence – in other words, they fail to completely 'cover' the input, which can result in surprising translations".

[6]The actual SCATE taxonomy also includes the categories 'terminology', 'source' and 'other', but these are left out here as there were no occurrences for any of the three systems.

	RBMT	PBMT	NMT
Total omissions	**43**	**115**	**62**
Content words	6	80	53
Function words	37	35	9
% Content words	0,14 %	69,96 %	85,48 %
Visibility	40	89	19
Invisibility	3	26	43
% Invisibility	7 %	23 %	69 %

Table 6: Subdivision of omission errors based on their type and visibility

For each omission, the number of words, type and visibility was annotated. The type of omission differentiates between content words and function words, the latter having a less severe impact on the accuracy of the translation. NMT drops, on average, more content words than RBMT and PBMT.

Another interesting aspect is the visibility of the omission error when there are traces in the target sentence that source content is missing. The visibility of an omission error can be defined as the expectation of content being missing when only reading the translation (without comparing it to the source). In other words, is the omission visible/ expected at monolingual level? To check for visibility of omission errors, all sentences with omission errors were extracted. The annotator indicated, with yes or no, if it was evident from reading only the target sentences if source content was missing.

A few examples will illustrate the invisibility of the omission error in the MT output when only the Dutch target translation is read. These are all examples from NMT.

EN: In Kinshasa, a Belgian colleague is busy with a similar fistula project with which we **would like to** collaborate more effectively.

NL: In Kinshasa is een Belgische collega bezig met een soortgelijk fistelproject waarmee we effectiever samenwerken.

EN: "There is almost no contact now between Israeli and Palestinian writers," Grossman told **me**.

NL: "Er is nu bijna geen contact tussen Israëlische en Palestijnse schrijvers, " vertelde Grossman.

For RBMT and PBMT, the omissions are being anticipated by fluency errors. The reverse is true for NMT, where the fluency is no indicator that all the source content is being transferred into the target. Even text spans of 4 words are being fluently omitted by NMT. We can conclude that the nature of the omission errors has changed. This forms a major issue not only for annotators and post-editors, but also for everybody using these commercial systems online for free, disposing only of the target text. In research, this issue challenges quality estimation of NMT output and gisting.

5.2 Fluency errors

As mentioned before, fluency deals with the well-formedness of the target language, regardless of the transmission of content and meaning from the source into the target sentence. The following categories are identified: grammar, lexicon, orthography, multiple errors and other. The labels of most error categories are self-explanatory, except for the label 'multiple errors', which is used when an accumulation of fluency errors on a text span makes it hard to identify the error separately. Table 7 gives an overview of the fluency performance of the different systems.

Fluency	RBMT	PBMT	NMT
Grammar	864	932	260
Orthography	290	253	95
Lexicon	533	235	358
Multiple errors	144	110	6
Other	0	1	0
Total	**1831**	**1531**	**719**

Table 7: Overview of the number of fluency errors

As previous research confirms, fluency is handled best by NMT for English-Dutch as well. The improvements are enormous for all categories, except for lexicon, which, therefore, draws our attention.

5.2.1 Lexicon

Lexical errors are split into two subcategories: 'non-existent' and 'lexical choice'. The latter distinguishes 'content words' from 'function words'. Table 8 gives an overview of all the lexical errors in the MT output. From the results in Table 8, it is clear that NMT makes much more lexical choice errors than PBMT, but it is actually only the content words that cause difficulties. For function words, NMT scores best.

Lexical choice	RBMT	SMT	NMT
Total	468	181	304
Content word	290	91	226
Function Word	178	90	78

Table 8: Subdivision of lexical choice errors

In many instances, the category 'Fluency-Lexical Choice' occurs together with an accuracy error for 'Mistranslation-Sense-Content word'. This type of fluency error is the clue that source content has not been rendered correctly.

5.2.2 Grammar

Another well represented fluency error category is grammar. Comparing the three paradigms, we see the progress that has been made. A further subdivision of this category is presented in Table 9.

Grammar	RBMT	PBMT	NMT
Word form	143	245	73
Word order	372	311	42
Extra word(s)	162	99	46
Missing word(s)	162	247	83
Multi word syntax	24	27	11
other	1	3	5
Total	**864**	**932**	**260**

Table 9: Subdivision of grammar errors

It is worthwhile to take a look into 'extra words'. Table 10 shows the newly added 'repetition' subcategory to label words or word groups that are unnecessarily repeated. The rest category 'other' contains all other extra words in the

target sentences, that should not be there. An example of repetition is illustrated below:

> EN: Located above Glasgow Central Station, on the corner of Union Street and Gordon Street, the **55 square metre** LED screen faces directly onto Renfield street -- the second largest retail location in the United Kingdom, and is visible across a range of over 600 metres.
>
> NL: **Het 55 vierkante meter** LED- scherm ligt boven het central station van Glasgow, op de hoek van Union Street en Gordon Street. Het scherm **heeft een oppervlakte van 55 meter** en is direct zichtbaar op Renfield Street, de tweede grootste winkelplaats in het Verenigd Koninkrijk.

The words in bold in the source sentence have been translated in 2 places (also in bold) in the target sentences. A subdivision of all extra word errors is presented in Table 10.

	RBMT	PBMT	NMT
Extra words	162	99	46
Repetition	9	6	15
Other	153	93	31

Table 10: Subdivision of 'grammar extra words' errors

Although NMT has less superfluous words in its output, it has a higher number of repetitions of one or more words than the other systems.

5.3 Long sentences

In literature, long sentences have been reported as a weakness of NMT systems. Bentivogli et al. (2016) found that the performance of NMT degraded faster with increased segment length. Toral and Sánchez-Cartagena (2017) confirmed this negative correlation and even reported that PBMT outperforms NMT for sentences consisting of 40 or more words.

	RBMT	PBMT	NMT
Long sentences			
# Errors	446	335	157
# Target words	1791	1749	1677
# Unique annotated words	821	685	195
% Erroneous words	46%	39%	12%
Short sentences			
# Errors	230	175	104
# Target words	969	958	950
# Unique annotated words	225	228	113
% Erroneous words	23%	24%	12%

Table 11: Performance on long sentences (min. 40 words) compared to short sentences (max. 10 words)

Table 11 shows us that NMT still outstrips PBMT for long sentences. In fact, two of the 38 long sentences in our corpus were translated without errors by NMT. PBMT and RBMT produced no correct long sentences. For the sake of completeness, we include the performance of all engines on all 145 short sentences found in our corpus as well. In addition to the number of errors, Table 11 also presents the number of target words and the number of unique annotated words for each system. In the number of unique annotated words, every erroneous word is only counted once, even though it might be annotated multiple times in the same sentence.

The percentage of wrong words in long and short sentences in our subset for NMT is the same. The expected degradation of NMT performance in long sentences, doesn't hold (anymore). To overcome the accumulation of errors in longer sentences in NMT, different architectures have been examined and tested (Barone, Helcl, Sennrich, Haddow, & Birch, 2017) and all kinds of attentional mechanisms have been investigated (Luong, Pham, & Manning, 2015) and implemented. The GNMT's architecture has also been enhanced by a bi-directional encoder for the bottom layer only, allowing for a maximum possible parallelisation during computation (Wu et al., 2016).

6. Conclusions and outlook

In this paper we compared the NMT output with RBMT and PBMT translations, providing an overview of the strengths and weaknesses of NMT. We explained why we expect that NMT output is more difficult to post-edit, by elaborating on the special and less transparent character of some types of NMT errors. Omissions and mistranslations that are semantically unrelated to the source, will be a future challenge, especially for all activities that only take the translation product into account (e.g. gisting and quality estimation of MT output).

7. Bibliographical References

Barone, A. V. M., Helcl, J., Sennrich, R., Haddow, B., and Birch. (2017). Deep architectures for NMT. In Proceedings of the conference on Machine Translation (WMT), Vol. 1, pp. 99–107, Copenhagen, Denmark, September. Association for Computational Linguistics.

Bentivogli, L., Bisazza, A., Cettolo, M., and Federico, M. (2016). Neural versus phrase-based machine translation quality: a case study. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 257–267, Austin, Texas, November.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M. et al. (2016). Findings of the 2016 conference on Machine Translation, In Proceedings of the First Conference on Machine Translation: Shared Task Papers (WMT), Vol. 2, pp. 131–198, Berlin, Germany, August. Association for Computational Linguistics.

Burchardt, A., Macketanz, V., Dehdari, J., Heigold, G., Peter, J.-T., and Williams, P. (2017). A Linguistic Evaluation of Rule-Based, Phrase-Based, and Neural MT Engines. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), pp. 159–170.

Castilho, S., Moorkens, J., Gaspari, F., Calixto, I., Tinsley, J., and Way, A. (2017). Is Neural Machine Translation the New State of the Art? *The Prague Bulletin of Mathematical Linguistics*, *108*(1), pp. 109–120.

Isabelle, P., Cherry, C., and Foster, G. (2017). A challenge set approach to evaluating machine translation. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2486–2496, Copenhagen,

Denmark, September. Association for Computational Linguistics.

Klubička, F., Toral, A., and Sánchez-Cartagena, V. M. (2017). Fine-Grained Human Evaluation of Neural Versus Phrase-Based Machine Translation. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), pp. 121–132.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1412–1421). Lisbon, Portugal: Association for Computational Linguistics.

Macken, L., De Clercq, O., and Paulussen, H. (2011). Dutch parallel corpus: a balanced copyright-cleared parallel corpus. *Meta: Journal Des Traducteurs/Meta: Translators' Journal*, *56*(2), pp. 374–390.

Popovic, M. (2015). chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation* (WMT), pp. 392–395, Lisboa, Portugal, September. Association for Computational Linguistics.

Popović, M. (2017). Comparing Language Related Issues for NMT and PBMT between German and English. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), 209–220.

Shterionov, D., Nagel, P., Casanellas, L., Superbo, R., and O'Dowd, T. (2017). Empirical evaluation of NMT and PBSMT quality for large-scale translation production. In Proceedings of the 20th Annual Conference of the European Association for Machine Translation (EAMT), p. 74, Prague, Czech Republic, May.

Tezcan, A., Hoste, V., and Macken, L. (submitted). Estimating Word-Level Quality of Statistical Machine Translation Output Using Monolingual Information Alone. *Computer, Speech & Language*.

Tezcan, A., Hoste, V., and Macken, L. (2017). A Neural Network Architecture for Detecting Grammatical Errors in Statistical Machine Translation. *The Prague Bulletin of Mathematical Linguistics*, *108*(1), pp. 133-145.

Tezcan, A., Hoste, V., and Macken, L. (2017). SCATE Taxonomy and Corpus of Machine Translation Errors. In G. Corpas Pastor & I. Durán Muñoz (Eds.), *Trends in e-tools and resources for translators and interpreters*, Brill: pp. 219–248.

Toral, A., and Sánchez-Cartagena, V. M. (2017). A Multifaceted Evaluation of Neural versus Phrase-Based Machine Translation for 9 Language Directions. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Vol. 1, pp. 1063-1073. Valencia, Spain, April. Association for Computational Linguistics.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv Preprint arXiv:1609.08144*.

Collection and Analysis of Code-switch Egyptian Arabic-English Speech Corpus

Injy Hamed, Mohamed Elmahdy, Slim Abdennadher
The German University in Cairo
El Tagamoa El Khames, New Cairo, Cairo, Egypt
{injy.hamed, mohamed.elmahdy, slim.abdennadher}@guc.edu.eg

Abstract

Speech corpora are key components needed by both: linguists (in language analyses, research and teaching languages) and Natural Language Processing (NLP) researchers (in training and evaluating several NLP tasks such as speech recognition, text-to-speech and speech-to-text synthesis). Despite of the great demand, there is still a huge shortage in available corpora, especially in the case of dialectal languages, and code-switched speech. In this paper, we present our efforts in collecting and analyzing a speech corpus for conversational Egyptian Arabic. As in other multilingual societies, it is common among Egyptians to use a mix of Arabic and English in daily conversations. The act of switching languages, at sentence boundaries or within the same sentence, is referred to as code-switching. The aim of this work is a three-fold: (1) gather conversational Egyptian Arabic spontaneous speech, (2) obtain manual transcriptions and (3) analyze the speech from the code-switching perspective. A subset of the transcriptions were manually annotated for part-of-speech (POS) tags. The POS distribution of the embedded words was analyzed as well as the POS distribution for the trigger words (Arabic words preceding a code-switching point). The speech corpus can be obtained by contacting the authors.

Keywords: Speech corpus, Dialectal Egyptian Arabic, Conversational Egyptian Arabic, Egyptian Arabic-English, code-switching, code-mixing

1. Introduction

The Arabic language is one of the most widely used languages in the world. It is the 6^{th} most used language based on number of of native speakers. Nearly 250 million persons use Arabic as their first language and it is the second language for around four times that number (Elmahdy et al., 2009). There are three types of the Arabic language: Classical Arabic, Modern Standard Arabic (MSA), and Colloquial (dialectal) Arabic. The classical Arabic is the most formal type of Arabic. It is used in the Quran and early Islamic literature. MSA is the official modern language used in the Arab world. It is derived from the Classical Arabic, with some simplifications such as removal of diacritic marks. MSA is the form of Arabic taught in schools and is used in writings, formal speeches, interviews, news broadcasts, movies' subtitling and education. However, MSA is not the language used in everyday life and is considered as a second language for all Arabic-speakers. Each country or region within a country has its own dialect. Colloquial (dialectal) Arabic is the language used in daily conversations and informal writings such as chats, blogs and comments on on-line social media such as Facebook and Twitter. However, they do not necessarily have a standard written form.

In addition to the three forms of Arabic, as in other multilingual environments, many Arabic-speakers use code-switching and code-mixing in their conversations. Such mixed speech is usually defined as a mixture of two distinct languages: primary language (also known as the matrix language); which is spoken in majority and secondary language (also known as the embedded language); by which words (in the case of code-mixing) or phrases (in the case of code-switching) are embedded into the conversation. There are two types of code-switching:

- Inter-sentential Code-switching: defined as switching languages from one sentence to another. For example: "كان عاجبني. It was very interesting."
  (I liked it. It was very interesting.)

- Intra-sentential Code-switching (also known as code-mixing): defined as using multiple languages within the same sentence. For example:
  " .implement a semantic search engine بن احنا كنا"
  (We were implementing a semantic search engine.)

Another type of language alternation is referred to as "borrowing" or "loanword". This is defined as having the whole sentence in one language except for words which are borrowed from the secondary language. For example: "انا بحب الresearch عمتا." (I generally love research.)

Researchers use different definitions for code-switching. Some researchers define code-switching as incorporating sentences from different languages, each having its own grammatical rules. Therefore, following this definition, borrowing is not considered as a type of code-switching. On the other hand, some researchers consider borrowing as a type of intra-sentential Code-switching. In the scope of this paper, for the sake of simplicity, we will use the term "code-switching" to refer to all types of language alternation; inter-sentential code-switching, intra-sentential code-switching as well as borrowing.

This phenomenon of multilingualism comes as a result of several factors such as globalization, immigration, colonization, the rise of education levels, as well as international business and communication. Code-switching is seen in several Arab countries, where for example, English is commonly used in Egypt and French in Morocco,

Tunisia, Lebanon and Jordon.

Arabic varieties are characterized by Diglossia where dialectal forms usually differ considerably from their formal language, and are thus considered by researchers to be a separate language (Ferguson, 1959). For instance, the huge variation between MSA and the Egyptian Colloquial Arabic was shown in the study conducted by Kirchhoff and Vergyri (Kirchhoff and Vergyri, 2005). The authors studied the data sharing between both languages, where the author calculated the percentage of shared unigrams, bigrams and trigrams in the LDC CallHome corpus (telephone conversations in Egyptian Colloquial Arabic) and the FBIS corpus (broadcast news in MSA). It was found that the corpora only overlap by 10.3% in unigrams, 1% in bigrams and $< 1\%$ in trigrams. The same overlap computation was done for the CHRISTINE corpus (conversational British English) and the American English Broadcast News data from the NIST 2004 Rich Transcription evaluations. The overlap was found to be 44.5% in unigrams, 19.2% in bigrams and 5.3% in trigrams. Consequently, existing speech corpora in Modern Standard Arabic will be unsuitable for recognizing dialectal Arabic. Moreover, it is also unsuitable to use monolingual speech corpora to recognize code-switch speech.

In literature, there are two main approaches used to recognize code-switched speech: language-dependent and language-independent. In the language-dependent approach, monolingual ASR systems are used. This is done by first detecting the boundaries at which language switching occurs using language boundary detection (LBD) algorithms. For each language-homogeneous segment, the language is identified using language identity detection (LID) algorithms. Finally, each segment is recognized by its respective monolingual ASR system. This approach is suitable for building multilingual ASR systems that handle multiple languages, where the input speech is monolingual, as well as those that handle inter-sentential code-switching. However, in the case of intra-sentential code-switching, the language switching occurs within the same sentence and the language-homogeneous segments can be short. The accuracy of the LBD and LID algorithms can then become a limitation to the overall ASR system performance. In this case, the language-independent approach is more suitable for the task. It involves building a truly multilingual language model, acoustic model and pronunciation dictionary that encompass both- or all- languages involved. Recognition is then done in a one-pass approach. Building multilingual language and acoustic models is considered to be a holistic task. One of the main challenges is the need for code-switching corpora for training and testing purposes. Despite the great efforts done in collecting speech and text corpora, there is still shortage in the available corpora for code-switching languages. The lack of existing speech corpora for code-switched Egyptian Arabic-English is a bottleneck in the creation of ASR systems for conversational Egyptian Arabic. Few speech corpora are available for Dialectal Egyptian Arabic, such as CALLHOME (Canavan et al., 1997). Dialectal Egyptian Arabic speech corpora were also gathered by and (Elmahdy et al., 2009) and (El-Sakhawy et al., 2014). These corpora may contain a small percentage of code-switching, however, they are mainly designed for Dialectal Egyptian Arabic. Up to our knowledge, there are no speech corpora dedicated to code-switching occurring in dialectal Egyptian Arabic. In this paper, we present our first efforts in filling this gap. The collected corpus can be used in training and testing data for building a multilingual Egyptian Arabic-English ASR system. It can also serve as a data set for linguistic analysis on the Egyptian Arabic-English code-switching behaviour.

The remainder of the paper is structured as follows: Related work will be discussed in Section 2 . Section 3 describes the process of corpus creation; speech collection and transcription. In Section 4 , the corpus is analyzed to examine/provide insights on its code-switching features. Finally, Section 5 concludes and provides future work.

2. Related Work

There are two types of speech corpora: read speech and spontaneous speech. Spontaneous speech corpora are usually preferred in building ASR systems as they are closer to natural speech. However, they require manual transcription, which is a labour-intensive and time-consuming task. Both approaches have been recorded for collecting code-switched speech corpora. (Chan et al., 2005) collected a Cantonese-English speech corpus through read newspaper content. (Li et al., 2012) gathered a Mandarin-English speech from four different sources: (1) conversational meetings ; (2) group project meetings; (3) student interviews speech. The speech was then transcribed by an annotator and manually verified by Mandarin-English bilingual speakers. (Lyu et al., 2015) collected the SEAME corpus, where Mandarin-English audio recordings were collected from interviews and conversational speech and were manually transcribed. (Chong et al., 2012) developed a Malay conversational speech corpus, containing Malay-English code-switching through the recording and transcription of conversational speech.

3. Approach

The purpose of this work is to develop a corpus for spontaneous Egyptian Arabic-English code-switching speech. This was done by recording informal interviews rather than collecting read speech to gather casual speech of spontaneous nature. This decision was made under the assumption that code-switching occurs most frequently in spontaneous speech.

For the interview setups, each interview involved two interviewers and one or two interviewee(s). Only the interviewees' speech were transcribed. The participants were asked to discuss technical topics such as the courses they teach, work experiences, as well as their B.Sc., M.Sc. and Ph.D. projects. The participants were Egyptian teaching assistants in the German University in Cairo. The participants were of both genders, with their ages ranging between 23 and 28. The mother tongue of all participants

is Arabic, and they are all fluent in English.

The interviews were recorded in a quiet closed room at the German University in Cairo. The recordings were collected using two table-mounted microphones. All audio recordings were stored in mono channel pulse-code modulation (PCM) at 16kHz sampling rate. In order to minimize the interference with power lines, the laptop charger was disconnected. The transcriptions were done manually by two Egyptian annotators who master both languages. The transcriptions were done using TranscriberAG. All transcriptions were recorded in UTF-8. Arabic transcriptions were written without diactritic marks. All non-speech sounds, pauses, and unintelligible speech segments were transcribed with some predefined filler tags.

4. Corpus Evaluation

The interviews were held with 12 participants; 6 males and 6 females. A total of 5.3 hours were recorded, with an average duration of a participant's recording of 26.5 minutes. A total of 4.5 hours were transcribed. The transcriptions made up a total of 1,234 sentences and 17,769 words, with an average of 14 words per sentence. The transcribed text is analyzed from two perspectives: (1) code-switching and code-mixing distribution, (2) part-of-Speech (POS) distribution of embedded words and POS trigger tags. The most frequently used Arabic trigger words (preceding a code-switching point) are identified. The most frequently used unigrams, bigrams and trigrams are also presented as well as examples from the transcriptions.

4.1. Code-switching and code-mixing analysis

The transcriptions contain a total of 17,769 words; 11,045 (62.1%) Arabic and 6,641 (37.4%) English words. This shows a high usage of the embedded English language in the conversations. The transcriptions show a high rate of code-switching and code-mixing. The 1,234 sentences are divided by language as follows: 124 monolingual Arabic, 125 monolingual English and 985 mixed. This shows a percentage of 79.8% of code-mixing in speech. It also shows a percentage of 10.1% of code-switching, where the whole sentence was uttered in the embedded language. Only 10% of the sentences are uttered purely in the matrix (Arabic) language.

The transcriptions were further analyzed from the code-mixing perspective. In the scope of the code-mixed sentences, 34.4% of the words are in the embedded language. The average number of code-switch points (Arabic-English and English-Arabic) in a sentence is 4. On average, there are 2 embedded English parts. The average number of English words in an embedded part is 2.

4.2. Trigger Words

Trigger words are defined as the Arabic words preceding a code-switching point. There are in total 535 unique Arabic words preceding a code-switching point. Table 4.2. shows the most frequent trigger words.

Trigger word	Percentage
ال	31.0%
في	4.8%
و	3.4%
يعني	1.5%
هو	1.3%

Table 1: The most frequent trigger words.

4.3. Top unigrams, bigrams and trigrams

The number of unigrams, bigrams and trigrams gathered from the transcriptions are shown in Table 4.3.. Figure 4.3. shows the most frequently used mixed n-grams in the conversations.

N-grams order	Count
unigrams	4,330
bigrams	14,205
trigrams	15,855

Table 2: The number of unigrams, bigrams and trigrams gathered from the transcriptions.

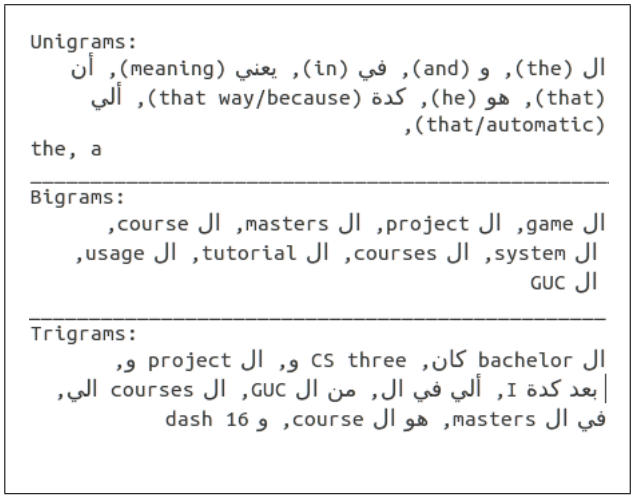


Figure 1: The top most frequently used unigrams, bigrams and trigrams.

4.4. Part-of-Speech analysis

The transcriptions were analyzed to determine the Part-of-speech (POS) of the embedded English words. A total of 388 sentences were manually annotated for POS tags. Figure 2 shows the POS distribution of the embedded English words. It can be seen from the data that the participants used the English language mostly in nouns.
The POS of the Arabic words preceding a code-switching point was also examined. These are considered to be trigger POS tags. Figure 3 shows the POS distribution of the trigger POS tags. It can be seen that code-switching mostly occurs after articles, followed by verbs and prepositions.

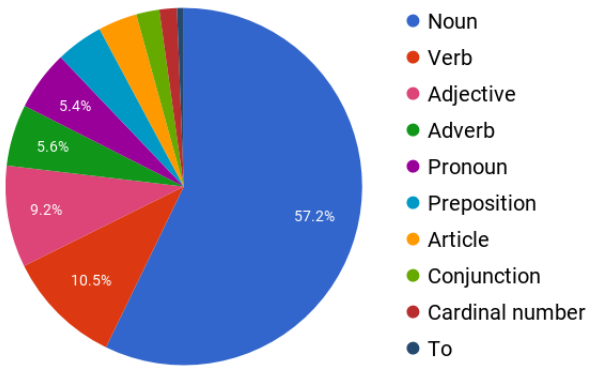


Figure 2: The POS distribution of the embedded English words in the set of manually annotated sentences.

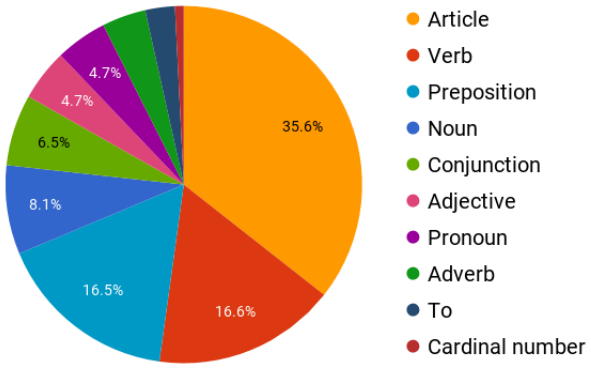


Figure 3: The POS distribution of the trigger POS tags in the set of manually annotated sentences.

4.5. Transcriptions examples

Figure 4 presents a sample of the transcribed sentences.

5. Conclusion

Code-switching has become a prevalent phenomenon in everyday conversations in multilingual societies. In Egypt, it has become common, especially among youth, to mix Arabic and English in conversations. This created a need for ASR systems to be able to recognize mixed Egyptian Arabic-English speech. In order to build such a multilingual ASR system, an Egyptian Arabic-English speech corpus is crucial. In this work, we collected spontaneous speech through informal interviews. The topics of the interviews were chosen to be technical, thus more probable to contain code-switching. The collected speech was manually annotated. The transcriptions were analyzed in terms of the code-switching and code-mixing behaviour. It was seen that both phenomena were used extensively. A subset of the transcribed data was manually annotated for POS tags. Analyses were performed on the POS tags of the English embedded words as well as the Arabic words preceding a code-switching point. It was found that English embedded words were mostly used for nouns, and that code-switching (from Arabic to English) occurs most frequently after articles. It was found that code-switching occurs at a rate of 30% after the artice ال. It is to be noted that in technical contexts, many technical terms are embedded as loanwords, thus contributing to this high code-switching rate. Accordingly, further research needs to be done to investigate the code-switching rate in other domains. We also intend to continue working on the corpus and collect more recordings and transcriptions.

6. Bibliographical References

Canavan, A., Zipperlen, G., and Graff, D. (1997). Callhome egyptian arabic speech. *Linguistic Data Consortium*.

Chan, J. Y., Ching, P., and Lee, T. (2005). Development of a cantonese-english code-mixing speech corpus. In *Ninth European Conference on Speech Communication and Technology*.

Chong, T. Y., Xiao, X., Tan, T.-P., Chng, E. S., and Li, H. (2012). Collection and annotation of malay conversational speech corpus. In *Speech Database and Assessments (Oriental COCOSDA), 2012 International Conference on*, pages 30–35. IEEE.

El-Sakhawy, D., Abdennadher, S., and Hamed, I. (2014). Collecting data for automatic speech recognition systems in dialectal arabic using games with a purpose. In *Multimodal Analyses enabling Artificial Agents in Human-Machine Interaction*. Springer LNAI.

Elmahdy, M., Gruhn, R., Minker, W., and Abdennadher, S. (2009). Modern standard arabic based multilingual approach for dialectal arabic speech recognition. In *Eighth International Symposium on Natural Language Processing (SNLP'09)*, pages 169–174. IEEE.

Ferguson, C. A. (1959). Diglossia. *word*, 15(2):325–340.

Kirchhoff, K. and Vergyri, D. (2005). Cross-dialectal data sharing for acoustic modeling in arabic speech recognition. *Speech Communication*, 46(1):37–51.

Li, Y., Yu, Y., and Fung, P. (2012). A mandarin-english code-switching corpus. In *LREC*, pages 2515–2519.

Lyu, D.-C., Tan, T.-P., Chng, E.-S., and Li, H. (2015). Mandarin–english code-switching speech corpus in south-east asia: Seame. *Language Resources and Evaluation*, 49(3):581–600.

- {79-82} هي ليها علاقة بنفس ال field الي هو AI
- {116-120} بس يعني ال courses الي ليها علاقة بال AI عندنا يعني that's so far
- {376-284} ال game دي بتحكي زي أن أنت بتاخد ألاول training بيديلك معلومات أزاي ت conserve energy أزي تحسن ال usage و ال consumption بتاعك
- {325-330} بس مشكلة ال fish eye camera أن بتلاقي قرب ال edges الصورة بتبقا أوي distorted
- {128-138} ال algorithm دة بيشيل ال effect without knowing ال, room size male or female مين الي بيتكلم without knowing
- {22-25} عملت ال masters و خلصتها فبقالي كدة في الجمعة اربع سنين
- {110-114} أن احنا كنا بنحاول ن convert visual information into sounds for visually impaired people
- {21-23} كنا بن implement a semantic search engine
- {1271-1281} أنا اما ببعت مثلا video أو صوت مثلا أزاي ا detect أن الحتة دي هي random أو دي مثلا noise فأنا مش لازم ابعتها و تاخد مكان
- {2725-2727} عشان كل ما أنت بت adapt yourself أكتر للمكان

Figure 4: Some of the transcribed sentences. The time interval in seconds is shown for each sentence between braces.

Multimodal Lexical Translation

Chiraag Lala and Lucia Specia
Department of Computer Science, University of Sheffield
Regent Court, 211 Portobello, Sheffield, S1 4DP, UK
{clala1, l.specia}@sheffield.ac.uk

Abstract

Inspired by the tasks of Multimodal Machine Translation and Visual Sense Disambiguation we introduce a task called Multimodal Lexical Translation (MLT). The aim of this new task is to correctly translate an ambiguous word given its context - an image and a sentence in the source language. To facilitate the task, we introduce the MLT dataset, where each data point is a 4-tuple consisting of an ambiguous source word, its visual context (an image), its textual context (a source sentence), and its translation that conforms with the visual and textual contexts. The dataset has been created from the Multi30K corpus using word-alignment followed by human inspection for translations from English to German and English to French. We also introduce a simple heuristic to quantify the extent of the ambiguity of a word from the distribution of its translations and use it to select subsets of the MLT Dataset which are difficult to translate. These form a valuable multimodal and multilingual language resource with several potential uses including evaluation of lexical disambiguation within (Multimodal) Machine Translation systems.

Keywords: Multimodal Machine Translation, Visual Sense Disambiguation, Multimodal Multilingual Language Resources

1. Introduction

Multimodal Machine Translation is the task of translating text using information in other modalities (such as images) as auxiliary cues. It has been recently framed as a shared task as part of the last two editions of the Conference on Machine Translation (WMT16, WMT17) (Specia et al., 2016; Elliott et al., 2017). Within the Conference on Machine Translation, the task is defined as: Given an image and its description in the source language, the objective is to translate the description into a target language, where this process can be supported by information from the image, as depicted in Figure 1.

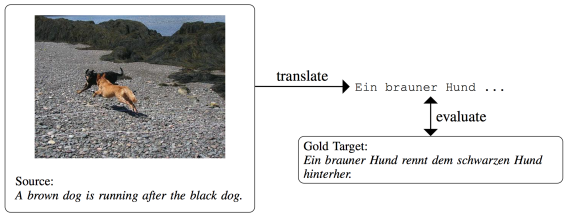


Figure 1: Multimodal Machine Translation Shared Task

One of the main motivations to introduce multimodality in Machine Translation is the intuition that information from other modalities could help find the correct sense of ambiguous words in the source sentence, which could potentially lead to more accurate translations. For example, the English sentence "A man is holding a seal" could have at least two different translations in German depending on the sense of the word *seal* - (1) "Ein Mann hält ein Siegel", and (2) "Ein Mann hält einen Seehund". The images (Figure 2) could help a Multimodal Machine Translation system disambiguate the correct sense of the word *seal* and translate accordingly.

Disambiguation of word senses, popularly known as Word Sense Disambiguation or Lexical Disambiguation, is a widely studied natural language processing task. Given an ambiguous word and its context, the objective is to assign the correct sense of the word based on a pre-defined sense inventory. A review of approaches to Word Sense Disambiguation can be found in Navigli (2009) and Raganato et al. (2017).



(a) Ein Mann hält ein Siegel



(b) Ein Mann hält einen Seehund

Figure 2: Two different translations of "A man is holding a seal" depending on the visual context

In standard Word Sense Disambiguation, words are disambiguated based on their textual context. However, in a multimodal setting we could also disambiguate words using visual context. This modified version of Word Sense Disambiguation that uses visual context instead of textual context is called Visual Sense Disambiguation.

In monolingual work, Visual Sense Disambiguation has previously been attempted for ambiguous nouns like the word 'bank' which could refer to a financial institution or a river bank (Barnard et al., 2003; Loeff et al., 2006; Saenko and Darrell, 2009; Chen et al., 2015). Recently, Visual Sense Disambiguation has also been attempted for ambiguous verbs like the word 'play' which could refer to playing a musical instrument or playing a sport (Gella et al., 2016).

In Machine Translation, including Multimodal Machine

Translation, disambiguation of word sense happens implicitly. For instance, in the same example "A man is holding a seal", we would come to know whether the system disambiguated the correct sense of the word *seal* only indirectly from the translation produced by the system. The corresponding translation of the word *seal* in the target language (*Siegel* or *Seehund* in German) acts as a "sense label". Further, in Multimodal Machine Translation, we would like know which modality (visual or textual) contributed to the disambiguation and to what extent.

The main contribution of this paper is to facilitate the study of Word Sense Disambiguation within Multimodal Machine Translation framework by:

1. Generating a language resource of ambiguous words and its translations together with visual and textual contexts. We call this the Multimodal Lexical Translation Dataset (MLT Dataset).
2. Introducing a new task - Multimodal Lexical Translation (MLT) - where the objective is to translate a single word into the target language given an image (visual context) and a sentence in the source language (textual context).
3. Demonstrating a simple way to evaluate lexical disambiguation within Multimodal Machine Translation using the MLT Dataset.

We build this resource for English to German and English to French translations.

2. Language Resource - MLT Dataset

The MLT Dataset is a collection of 4-tuples of the form:

$$\{(x_i, y_i, \mathbf{x}_i, \mathbf{v}_i)\}_{i=1}^{n} \quad (1)$$

where x_i is an ambiguous[1] word, $\mathbf{x}_i$ is its textual context (a source sentence), $\mathbf{v}_i$ is its visual context (an image), and y_i is its translation that conforms with both the textual and visual contexts.

2.1. Generating the MLT Dataset

We make use of the Multi30K dataset (Elliott et al., 2016; Elliott et al., 2017), an extension of the Flickr30K dataset (Young et al., 2014), which consists of 31,014 triples of the form $(\mathbf{v}_i, \mathbf{x}_i, \mathbf{y}_i)$ where $\mathbf{v}_i$ is an image, $\mathbf{x}_i$ is a description of the image in the source language (English) and $\mathbf{y}_i$ is a translation of the description in the target language (German and French) by human translators (i is an integer index ranging from 1 to 31,014). From this sentence-level dataset, we extract the ambiguous words and their lexical translations using the following steps:

Pre-processing → Word Alignment → Automatic Filtering → Human Filtering.

2.1.1. Pre-processing

Sentences in all languages are lowercased and tokenized using scripts from the Moses toolkit[2] (Koehn et al., 2007). German sentences, which can contain compound words like 'sonnenblumenkerne' (sunflower seeds), are split/decompounded using pre-computed model of SEmantic COmpound Splitter (SECOS)[3] (Riedl and Biemann, 2016). Since we are not interested in distinguishing morphological variants of the words, we also lemmatized[4] all sentences in the respective languages, which reduced vocabulary size and led to better word alignment in the later step.

2.1.2. Word Alignment

After the pre-processing step, the word tokens in the Multi30K parallel corpus are aligned using Fast Align[5] (Dyer et al., 2013). Fast Align generates asymmetric word alignments depending on which language in the parallel corpus is treated as the source. We generate both alignments - 'forward' (where English is treated as the source language) and 'reverse' (where German or French is treated as the source language). To learn better word alignments, we train Fast Align on a larger corpus comprising of the Europarl parallel corpus[6] (Koehn, 2005) in addition to the Multi30K parallel corpus for the English-German and English-French language pairs separately. The Europarl corpus also undergoes the same pre-processing steps in Section 2.1.1. before word alignment.

2.1.3. Automatic Filtering

In this step we remove all the word alignments having stop words and select only those alignments which are to be found in both 'forward' and 'reverse' directions. In addition, we filter out the alignments between words with different Part-Of-Speech (POS) tags (using the NLP tool in footnote 4). Next, we remove all English words that get aligned to a single word in the target language across the entire Multi30K corpus, retaining only the potentially ambiguous English words, i.e. those aligned to multiple words in the target language. These retained alignments are converted into a dictionary format where 'Keys' are the potentially ambiguous English words and 'Values' are the words in the target language that get aligned to it. For instance, in English-French language direction we have cases like:

four → *quart, quartequatre*
woods → *forêt, bois*
western → *occidental, western*
hat → *casque, casquette, chapeau, haut, bonnet, couvre, képi, béret*

One dictionary from the word alignments of each language pair is built independently, i.e. one for English-German and one for English-French.

2.1.4. Human Filtering

Finally, each dictionary (English-German and English-French) from the Automatic Filtering step is given to human annotators for a final inspection and filtering. Human

[1] We use the term 'ambiguous' for those words in the source language that have multiple translations in the target language in a given parallel corpus, loosely representing different 'senses' of the word in that corpus.

[2] https://github.com/moses-smt/mosesdecoder

[3] https://github.com/riedlma/SECOS

[4] http://staffwww.dcs.shef.ac.uk/people/A.Aker/activityNLPProjects.html

[5] https://github.com/clab/fast_align

[6] http://www.statmt.org/europarl/

annotators are native speakers of French/German who are also fluent in English. They were asked to

1. filter out instances which they believe did not have multiple senses, e.g. 'western → *occidental, western*'

2. filter out target words which they believe are not translations of the source word in any context, such as *haut* in the example 'hat → *casque, casquette, chapeau, haut, bonnet, couvre, képi, béret*'

The annotators were given the freedom to use any other resource, such as bilingual dictionaries, existing translation tools, etcetera that may help them filter the dictionaries. After the final filtering and inspection, for each (Key, Value) pair in the dictionaries we retrieve the visual and textual contexts from the Multi30K dataset to complete the MLT Dataset language resource.

2.2. Dataset Statistics and Examples

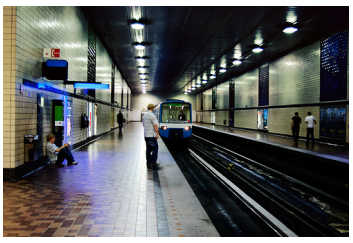
We extracted 1108 words in English which are ambiguous in either German or French or both (i.e. they have multiple translations in the target language). Each of these words can appear in multiple sentences, thus resulting in a total of 98,647 MLT datapoints.
Both English-German and English-French MLT language resources are made freely available[7] under the Creative Commons Attribution Non Commercial ShareAlike 4.0 International license.

English - French
We extracted 661 words in English which are ambiguous in French with 2.98 translations per word (on average) and 22.73 instances per translation (on average) totaling to 44,779 MLT datapoints.

English - German
We extracted 745 words in English which are ambiguous in German with 4.09 translations per word (on average) and 17.69 instances per translation (on average) totaling to 53,868 MLT datapoints. A couple of examples are shown below.

Examples

1. Ambiguous Word x_1: *subway*
   Lexical Translation y_1: *bahnstation*
   Textual Context $\mathbf{x}_1$: *"a few people are waiting in a subway, with an arriving car in the distance."*
   Visual Context $\mathbf{v}_1$:



2. Ambiguous Word x_2: *subway*
   Lexical Translation y_2: *subway*
   Textual Context $\mathbf{x}_2$: *"pedestrians bombard a city street covered in consumerism, including signs for burger king, mcdonalds, subway, and heineken."*
   Visual Context $\mathbf{v}_2$:



3. Multimodal Lexical Translation Task

Once we have the MLT Dataset, which is of the form in Equation 1, then at least three versions of MLT task can be defined. Given an ambiguous word x, translate it (or disambiguate its sense) using its

1. Textual context only, i.e. source sentence $\mathbf{x}$ only

2. Visual context only, i.e. image $\mathbf{v}$ only

3. Both Textual and Visual contexts $(\mathbf{x}, \mathbf{v})$

These three versions can help identify the relative importance of the textual context and visual context when translating (or disambiguating) an ambiguous word.

3.1. Evaluating Machine Translation Systems

MLT can be used to evaluate (Multimodal) Machine Translation systems in their ability to correctly translate ambiguous words. Consider a MLT datapoint $(x, y, \mathbf{x}, \mathbf{v})$ of the form in Equation 1. A Multimodal Machine Translation system S can take two arguments as inputs - the source sentence $\mathbf{x}$ and the image $\mathbf{v}$ - and generate an output $S(\mathbf{x}, \mathbf{v})$, which is a translation of the source sentence in the target language. A straightforward evaluation strategy is to simply check if the correct lexical translation y of the ambiguous word x, as given in the reference translation, is also found in the system's output $S(\mathbf{x}, \mathbf{v})$[8]. When run across all the examples in the MLT dataset, we can then count the number of times a system translated ambiguous words correctly to compute its accuracy in the task. We call this the *MLT accuracy*. More elaborate metrics and metrics that also consider variants of the words in the reference, will be developed and tested in future. For now, we use this simple accuracy measure and demonstrate a potential application of the MLT Dataset.

[7] https://github.com/sheffieldnlp/mlt

[8] For consistency, the system's outputs undergo the same preprocessing steps in Section 2.1.1.

3.1.1. Evaluating Machine Translation Systems

In Elliott et al. (2017), the Multimodal and Text-only Machine Translation systems submitted to the shared task were evaluated and ranked using the Meteor metric and Human scoring.

The Meteor metric (Denkowski and Lavie, 2014) calculates a sentence-level similarity score between 0 and 100 between the system output and the reference (human) translation, where 0 means no similarity and 100 means 'perfect' similarity. Similarity is computed as a function of the proportion of words that can be aligned between the system and human translations, allowing for different types of alignments (e.g. lemma, synonym). The overall Meteor score of a system is the mean of the sentence-level scores over the test set.

Human scoring was carried out in Elliott et al. (2017) using bilingual Direct Assessment (Graham et al., 2017), where the assessors were asked to evaluate the semantic relatedness between the system outputs and the source sentence (not the reference translation) given the image. The assessors gave a sentence-level score between 0 and 100, where 0 indicates that the meaning of the source sentence is not preserved in the system output, and 100 means that the meaning is 'perfectly' preserved. The sentence-level scores were standardized according to each individual assessor's overall mean and standard deviation score. The overall Human score of a system was computed as the mean of the standardized sentence-level scores over the test set.

We evaluate the participating systems of the Multimodal Machine Translation Task using the MLT Accuracy. The MLT Accuracy of a system measures the proportion of ambiguous words in the test set that are correctly translated by the system. The ambiguous words in the test set and its lexical translations are obtained from the MLT Dataset. We extract ambiguous words from the official test sets – Multi30K 2017 test set and Ambiguous COCO test set (Elliott et al., 2017) – for our evaluation. The MLT Accuracy is measured in percentages on a scale of 0 to 100.

The performance of all submissions to the Multi30K 2017 test set is shown in Table 1 for English to German, and Table 2 for English to French. The performance of all submissions to the Ambiguous COCO test set is shown in Table 3 for English to German, and Table 4 for English to French. For the Ambiguous COCO test set, no human evaluation was performed.

3.1.2. System ranking correlation

We observe that our evaluation of MLT Accuracy is mostly consistent with Meteor and human scores. To measure the extent of this consistency, we computed the Spearman's rank correlation coefficient ρ_s and Pearson's Correlation Coefficient ρ_p between MLT and Meteor or MLT and Human in Tables 1, 2, 3, and 4. The results are as follows.

For EN-DE on Multi30K 2017 test set (Table 1)
$\rho_s(\text{MLT}, \text{Meteor}) = 0.94 \quad \rho_s(\text{MLT}, \text{Human}) = 0.90$
$\rho_p(\text{MLT}, \text{Meteor}) = 0.99 \quad \rho_p(\text{MLT}, \text{Human}) = 0.78$

For EN-FR on Multi30K 2017 test set (Table 2)
$\rho_s(\text{MLT}, \text{Meteor}) = 0.93 \quad \rho_s(\text{MLT}, \text{Human}) = 0.54$

System	**MLT** ↑	Meteor ↑	Human ↑
NICT_1_NMTrerank_C	**75.49**	53.9	70.3
LIUMCVC_NMT_C	**74.70**	53.8	65.1
LIUMCVC_MNMT_C	**73.78**	54.0	77.8
DCU-ADAPT_MultiMT_C	**71.54**	50.5	68.1
UvA-TiCC_IMAGINATION_U	**70.75**	53.5	74.1
UvA-TiCC_IMAGINATION_C	**70.75**	51.2	59.7
CUNI_NeuralMonkeyTextualMT_U	**69.96**	51.0	68.1
CUNI_NeuralMonkeyMultimodalMT_U	**69.30**	50.2	60.6
OREGONSTATE_2NeuralTranslation_C	**68.64**	50.6	54.4
CUNI_NeuralMonkeyTextualMT_C	**67.72**	49.2	54.2
CUNI_NeuralMonkeyMultimodalMT_C	**64.95**	47.1	55.9
OREGONSTATE_1NeuralTranslation_C	**64.82**	48.9	53.3
SHEF_ShefClassProj_C	**60.74**	43.4	49.4
SHEF_ShefClassInitDec_C	**60.47**	44.5	46.6
AFRL-OHIOSTATE-MULTIMODAL_U	**23.06**	20.2	36.6

Table 1: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Multi30K 2017 test set for English to German.

System	**MLT** ↑	Meteor ↑	Human ↑
NICT_1_NMTrerank_C	**82.50**	72.0	79.4
LIUMCVC_NMT_C	**81.34**	70.1	60.5
LIUMCVC_MNMT_C	**81.23**	72.1	71.2
DCU-ADAPT_MultiMT_C	**81.00**	70.1	74.1
OREGONSTATE_2NeuralTranslation_C	**78.68**	68.3	65.4
OREGONSTATE_1NeuralTranslation_C	**75.78**	67.2	60.8
CUNI_NeuralMonkeyTextualMT_C	**75.55**	67.0	61.9
CUNI_NeuralMonkeyMultimodalMT_C	**74.97**	67.2	74.2
SHEF_ShefClassInitDec_C	**73.70**	62.8	54.7
SHEF_ShefClassProj_C	**72.42**	61.5	54.0

Table 2: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Multi30K 2017 test set for English to French.

System	**MLT** ↑	Meteor ↑
DCU-ADAPT_MultiMT_C	**68.50**	46.8
LIUMCVC_NMT_C	**68.24**	48.9
NICT_1_NMTrerank_C	**67.19**	48.5
UvA-TiCC_IMAGINATION_C	**67.19**	45.8
LIUMCVC_MNMT_C	**66.40**	48.8
CUNI_NeuralMonkeyMultimodalMT_U	**65.35**	45.6
UvA-TiCC_IMAGINATION_U	**64.30**	48.1
CUNI_NeuralMonkeyTextualMT_U	**63.78**	46.0
OREGONSTATE_1NeuralTranslation_C	**62.99**	46.5
OREGONSTATE_2NeuralTranslation_C	**62.99**	45.7
CUNI_NeuralMonkeyTextualMT_C	**62.99**	43.8
CUNI_NeuralMonkeyMultimodalMT_C	**57.74**	42.7
SHEF_ShefClassProj_C	**55.64**	40.0
SHEF_ShefClassInitDec_C	**54.33**	40.7

Table 3: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Ambiguous COCO test set for English to German.

System	**MLT** ↑	Meteor ↑
LIUMCVC_MNMT_C	**77.55**	65.9
NICT_1_NMTrerank_C	**77.55**	65.6
DCU-ADAPT_MultiMT_C	**76.42**	64.1
LIUMCVC_NMT_C	**75.28**	63.4
OREGONSTATE_2NeuralTranslation_C	**74.83**	63.8
CUNI_NeuralMonkeyTextualMT_C	**74.83**	62.5
CUNI_NeuralMonkeyMultimodalMT_C	**74.83**	62.5
OREGONSTATE_1NeuralTranslation_C	**70.75**	61.6
SHEF_ShefClassProj_C	**68.93**	57.0
SHEF_ShefClassInitDec_C	**68.48**	57.3

Table 4: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Ambiguous COCO test set for English to French.

$\rho_p(\text{MLT}, \text{Meteor}) = 0.94 \quad \rho_p(\text{MLT}, \text{Human}) = 0.68$

For EN-DE on Ambiguous COCO test set (Table 3)
$\rho_s(\text{MLT}, \text{Meteor}) = 0.80 \quad \rho_p(\text{MLT}, \text{Meteor}) = 0.90$

For EN-FR on Ambiguous COCO test set (Table 4)
$\rho_s(\text{MLT}, \text{Meteor}) = 0.95 \quad \rho_p(\text{MLT}, \text{Meteor}) = 0.96$

Ranking of the systems using MLT Accuracy differs only slightly from the ranking using Meteor or human scores, and the top performing systems are often the same. On further inspection, we notice that the high correlation of MLT Accuracy and Meteor is mainly due to words with skewed distributions for their translations.

For instance, the ambiguous word *lean* has the following translations in German in our training set - *lehnen* (to be leaning), *schlank* (slim), *stützen* (support), and *beugen* (bend). However, in the training set *lehnen* occurs 137 times while the rest of the lexical translations combined occur only 16 times. Such a skewed distribution makes the word *lean* virtually unambiguous (or less ambiguous) compared to the cases when the distribution is more uniform over the translations. We propose a way to deal with such skewed distributions using a simple heuristic we call the 'ambiguity score'.

3.1.3. Ambiguity Score

On inspecting the generated MLT Dataset, we noticed that ambiguity of words is not a simple concept to define and measure. Some words appear to be more ambiguous than others based on the distribution of their translations in the training set, while other words, like *lean* (see Section 3.1.2.), appear less ambiguous in the training set due to the skewed distribution of its translations. We propose to quantify the extent of ambiguity using a simple heuristic that looks at the distribution of the translations.

Consider a word en in English with *n* different translations in German $de_1, de_2, ..., de_n$. Let $freq(de_i|en)$ denote the number of times the word de_i occurs as a translation of en in the training set. Also, without loss of generality, arrange the translations in decreasing order of frequency, i.e. $freq(de_1|en) > freq(de_2|en) > ... > freq(de_n|en)$. Then we define ambiguity score of en as:

$$Ambiguity(en) = \frac{\sum_{i=2}^{n} freq(de_i|en)}{freq(de_1|en)} \qquad (2)$$

Using the above formulation, an ambiguity score of zero signifies unambiguous words and closer to zero signifies low ambiguity. The higher the ambiguity score, the more difficult it is to translate/disambiguate the source word correctly. Thus, to increase the difficulty of the MLT Dataset we can filter out words with ambiguity scores below a certain threshold. To demonstrate this, we set an ambiguity threshold of 0.2 and filter out all those words in the MLT Dataset with ambiguity score below this threshold. We then evaluate the WMT 2017 participating systems using MLT accuracy on this difficult version of the MLT Dataset (denoted as $\text{MLT}_{0.2}$). The results are shown in Tables 5, 6, 7 and 8:

System	**$\text{MLT}_{0.2}$** ↑	Meteor ↑	Human ↑
NICT_1_NMTrerank_C	**69.08**	53.9	70.3
LIUMCVC_NMT_C	**69.08**	53.8	65.1
LIUMCVC_MNMT_C	**68.10**	54.0	77.8
DCU-ADAPT_MultiMT_C	**65.75**	50.5	68.1
UvA-TiCC_IMAGINATION_U	**65.36**	53.5	74.1
CUNI_NeuralMonkeyTextualMT_U	**63.99**	51.0	68.1
UvA-TiCC_IMAGINATION_C	**63.01**	51.2	59.7
OREGONSTATE_2NeuralTranslation_C	**62.23**	50.6	54.4
CUNI_NeuralMonkeyMultimodalMT_U	**61.64**	50.2	60.6
CUNI_NeuralMonkeyTextualMT_C	**61.25**	49.2	54.2
OREGONSTATE_1NeuralTranslation_C	**59.69**	48.9	53.3
CUNI_NeuralMonkeyMultimodalMT_C	**58.71**	47.1	55.9
SHEF_ShefClassProj_C	**56.16**	43.4	49.4
SHEF_ShefClassInitDec_C	**53.82**	44.5	46.6
AFRL-OHIOSTATE-MULTIMODAL_U	**18.40**	20.2	36.6

Table 5: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Multi30K 2017 $\text{MLT}_{0.2}$ subset for English to German.

System	**$\text{MLT}_{0.2}$** ↑	Meteor ↑	Human ↑
NICT_1_NMTrerank_C	**71.43**	72.0	79.4
LIUMCVC_MNMT_C	**71.17**	72.1	71.2
DCU-ADAPT_MultiMT_C	**69.61**	70.1	74.1
LIUMCVC_NMT_C	**69.09**	70.1	60.5
OREGONSTATE_2NeuralTranslation_C	**65.71**	68.3	65.4
OREGONSTATE_1NeuralTranslation_C	**63.38**	67.2	60.8
CUNI_NeuralMonkeyTextualMT_C	**60.78**	67.0	61.9
SHEF_ShefClassInitDec_C	**60.00**	62.8	54.7
CUNI_NeuralMonkeyMultimodalMT_C	**59.48**	67.2	74.2
SHEF_ShefClassProj_C	**58.44**	61.5	54.0

Table 6: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Multi30K 2017 $\text{MLT}_{0.2}$ subset for English to French.

System	**$\text{MLT}_{0.2}$** ↑	Meteor ↑
DCU-ADAPT_MultiMT_C	**65.59**	46.8
LIUMCVC_NMT_C	**64.16**	48.9
UvA-TiCC_IMAGINATION_C	**63.44**	45.8
NICT_1_NMTrerank_C	**60.93**	48.5
UvA-TiCC_IMAGINATION_U	**60.93**	48.1
LIUMCVC_MNMT_C	**59.50**	48.8
OREGONSTATE_2NeuralTranslation_C	**59.14**	45.7
CUNI_NeuralMonkeyMultimodalMT_U	**59.14**	45.6
CUNI_NeuralMonkeyTextualMT_U	**57.71**	46.0
OREGONSTATE_1NeuralTranslation_C	**56.99**	46.5
CUNI_NeuralMonkeyTextualMT_C	**56.63**	43.8
CUNI_NeuralMonkeyMultimodalMT_C	**51.61**	42.7
SHEF_ShefClassInitDec_C	**50.54**	40.7
SHEF_ShefClassProj_C	**49.10**	40.0

Table 7: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Ambiguous COCO $\text{MLT}_{0.2}$ subset for English to German.

System	**$\text{MLT}_{0.2}$** ↑	Meteor ↑
LIUMCVC_MNMT_C	**66.83**	65.9
NICT_1_NMTrerank_C	**65.83**	65.6
OREGONSTATE_2NeuralTranslation_C	**64.82**	63.8
DCU-ADAPT_MultiMT_C	**64.32**	64.1
LIUMCVC_NMT_C	**63.32**	63.4
CUNI_NeuralMonkeyTextualMT_C	**61.81**	62.5
CUNI_NeuralMonkeyMultimodalMT_C	**61.81**	62.5
OREGONSTATE_1NeuralTranslation_C	**58.29**	61.6
SHEF_ShefClassProj_C	**55.78**	57.0
SHEF_ShefClassInitDec_C	**53.77**	57.3

Table 8: Performance of systems submitted to the Multimodal Shared Task at WMT 2017 on Ambiguous COCO $\text{MLT}_{0.2}$ subset for English to French.

The first thing to notice is that for every system the MLT accuracy drops when evaluated on $MLT_{0.2}$ (Tables 5, 6, 7 and 8) as compared to the full MLT Dataset (Tables 1, 2, 3 and 4). This shows that by setting an ambiguity threshold we are extracting ambiguous words which are more difficult to translate/disambiguate.
In general, for any threshold τ, we can extract a subset MLT_τ of the MLT Dataset consisting only of words with ambiguity score $\geq \tau$. In other words, the threshold τ can be used to regulate the difficulty of the MLT Dataset. Also, system rankings change as threshold τ changes. This can help in error analysis and identify the strengths and weaknesses of the systems.

3.1.4. Analysis

According to MLT accuracy, for the teams that submitted both constrained and unconstrained models (those using additional external data for training), unconstrained models show improvement over their constrained counterparts in most cases (see Tables 1 and 5). For teams that submitted both multimodal and text-only systems, the role of multimodality is not evident as far as MLT Accuracy is concerned: sometimes multimodal systems perform better and sometimes text-only systems perform better. However, Human scores show that overall multimodal systems tend to be better than the text-only counterparts. MLT Accuracy fails to show this maybe because in its current form the matching performed between reference and system words is still too simplistic and does not take synonyms into account.



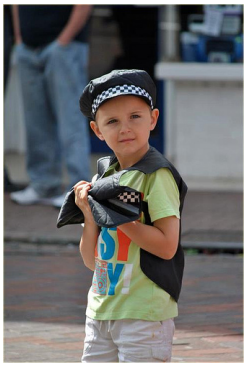
(a) hut



(b) kappe



(c) mütze



(d) kopfbedeckung

Figure 3: Different kinds of 'hats' translated into German differently based on the visual context in Multi30K corpus

Qualitative example In our English-German MLT Dataset, the word *hat* has been identified ambiguous because professional translators have translated it differently depending on the textual and visual contexts. Sometimes it has been translated as *hut* which refers to hats with edges/extensions coming off from all sides and usually worn in summer (see Figure 3a). Sometimes it has been translated as *kappe* which refers to the modern caps with shade extending out from front side only, usually worn in sports (see Figure 3b). Sometimes it has been translated as *mütze* which refers to differently designed hats usually worn in winters (see Figure 3c) and sometimes as *kopfbedeckung* which means a headgear which could refer to any kind of object worn on the head (see Figure 3d).
Now consider the following MLT data point from the 2016 test set whose textual context $\mathbf{x}$ was translated by the systems submitted to the Multimodal Shared Task:

Ambiguous Word x: *hat*
Lexical Translation y: *hut*
Textual Context $\mathbf{x}$: *"a man in an orange hat starring at something."*
Visual Context $\mathbf{v}$:



While most systems translated *hat* to *hut* in this example, a few translated differently. 'CUNI_NeuralMonkeyMultimodalMT_C' translated it as *kappe*, 'OREGONSTATE_1NeuralTranslation_C' translated it as *mütze*, and 'SHEF_ShefClassInitDec_C' translated it as *kopfbedeckung*. All these words are referring to the same object but have slightly different senses as seen earlier in Figure 3. From the image - visual context $\mathbf{v}$ - it can be seen that this hat looks a bit unusual because of its colour, texture and brand logo on it. Perhaps, that could be a reason why some systems chose another translation instead of *hut*. This can be considered as an instance where ambiguity is being introduced by the image.

4. Future Work

Currently, the MLT evaluation using counts and accuracy is too simplistic and has its limitations. First of all, it is based on exact matching of the surface-form of the gold standard lexical translation with the corresponding word in the system generated translation. Thus, any other form of the correct translation that does not appear in our gold standard lexical translation will be considered an error. This could be partly addressed by performing morphological analysis during the matching process. Secondly, no partial credit is given to synonymous words. This is a more difficult issue to address as synonyms can also have different senses. For instance, in the *hat* example discussed in the previous Section 3.1.4. all systems translated *hat* into some form of hat, which could be considered synonymous to some extent. Maybe not the correct kind of hat. Our future work will be focused on developing more elaborate scoring for MLT evaluation.

Additionally, we are interested in understanding whether the disambiguation happening within the system is due to the textual context, the visual context, both or none. For this, we propose to use a (Multimodal) Machine Translation system to translate the MLT data in four different ways. Recall from equation 1, MLT data is of the form $(x, y, \mathbf{x}, \mathbf{v})$. Given a Machine Translation System S, we should be able to compare four kinds of output with the reference.

$$S(\mathbf{x}, \mathbf{v}) \sim S(\mathbf{x}, \mathbf{0}) \sim S(x, \mathbf{v}) \sim S(x, \mathbf{0}) \sim y \quad (3)$$

where $\mathbf{0}$ refers to absence of visual context (no image) and $\sim$ refers to comparison of the different outputs. Such a comparison should help measure a system's ability of making use of different modalities. Thus, in addition to the *inter-system* comparisons that was demonstrated in section 3.1.1., in future we will work on these *intra-system* comparisons depicted in equation 3.

5. Conclusion

We introduced the Multimodal Lexical Translation language resource and the process of generating it from Multi30K using word alignments followed by human filtering. 53,868 MLT data points for English to German and 44,779 MLT data points for English to French have been generated.

Different versions of MLT tasks were also introduced. We demonstrated the use of MLT task to evaluate Multimodal and Text-only Machine Translation systems' ability to translate ambiguous words correctly. For this, submissions to the WMT 2017 Multimodal Shared task were evaluated using a simple MLT accuracy metric. This metric, in spite of its limitations, was found to be consistent with Meteor and human scoring used in Elliott et al. (2017). Further, we introduced a simple heuristic to quantify ambiguity based on the distribution of the translations in the training set and demonstrated its use to extract more ambiguous subset of the MLT Dataset which was found to be more difficult to translate/disambiguate.

We observed that in most cases unconstrained (Multimodal) Machine Translation models perform better than their constrained counterparts in terms of MLT accuracy. The contribution of multimodality in machine translation has not yet proved evident in terms of MLT. The qualitative example of *hat* in Section 3.1.4. showed us that the MLT Dataset can be useful to compare different Machine Translation Systems.

We believe the multimodal multilingual MLT Dataset is a useful language resource that can facilitate, among many things, the study of lexical disambiguation within Multimodal and Text-only Machine Translation systems.

6. Acknowledgements

This work is supported by the MultiMT project (H2020 ERC Starting Grant No. 678017). The authors also thank Mareike Hartmann, Charles Escudier, Julia Ive, Frédéric Blain, Pranava Madhyastha, and Josiah Wang.

7. Bibliographical References

Barnard, K., Johnson, M., and Forsyth, D. (2003). Word sense disambiguation with pictures. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2003) Workshop on Learning Word Meaning from non-Linguistic Data*, pages 1–5.

Chen, X., Ritter, A., Gupta, A., and Mitchell, T. (2015). Sense discovery via co-clustering on images and text. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5298–5306.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2013)*, pages 644–649.

Elliott, D., Frank, S., Sima'an, K., and Specia, L. (2016). Multi30k: Multilingual english-german image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74.

Elliott, D., Frank, S., Barrault, L., Bougares, F., and Specia, L. (2017). Findings of the second shared task on multimodal machine translation and multilingual image description. In *Proceedings of the Second Conference on Machine Translation*, pages 215–233.

Gella, S., Lapata, M., and Keller, F. (2016). Unsupervised visual sense disambiguation for verbs using multimodal embeddings. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*, pages 182–192.

Graham, Y., Baldwin, T., Moffat, A., and Zobel, J. (2017). Can machine translation systems be evaluated by the crowd alone. *Natural Language Engineering*, 23(1):3–30.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the Association for Computational Linguistics*, pages 177–180.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. *Proceedings of 10th Machine Translation Summit (MT Summit)*, pages 79–86.

Loeff, N., Alm, C. O., and Forsyth, D. A. (2006). Discriminating image senses by clustering with multimodal features. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 547–554.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):10.

Raganato, A., Bovi, C. D., and Navigli, R. (2017). Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1156–1167.

Riedl, M. and Biemann, C. (2016). Unsupervised compound splitting with distributional semantics rivals supervised methods. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 617–622.

Saenko, K. and Darrell, T. (2009). Unsupervised learning of visual sense models for polysemous words. In *Advances in Neural Information Processing Systems*, pages 1393–1400.

Specia, L., Frank, S., Sima'an, K., and Elliott, D. (2016). A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation*, pages 543–553.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Literality and Cognitive Effort: Japanese and Spanish

Isabel Lacruz, Michael Carl, Masaru Yamada
Kent State University, Kent, Ohio, United States
CRITT, Copenhagen Business School, Copenhagen, Denmark
Kansai University, Osaka, Japan
ilacruz@kent.edu, mc.ibc@cbs.dk, yamada@apple-eye.com

Abstract

We introduce a notion of pause-word ratio computed using ranges of pause lengths rather than lower cutoffs for pause lengths. Standard pause-word ratios are indicators of cognitive effort during different translation modalities.The pause range version allows for the study of how different types of pauses relate to the extent of cognitive effort and where it occurs in the translation process. In this article we focus on short monitoring pauses and how they relate to the cognitive effort involved in translation and post-editing for language pairs that are different in terms of semantic and syntactic remoteness. We use data from the CRITT TPR database, comparing translation and post-editing from English to Japanese and from English to Spanish, and study the interaction of pause-word ratio for short pauses ranging between 300 and 500ms with syntactic remoteness, measured by the CrossS feature, semantic remoteness, measured by HTra, and syntactic and semantic remoteness, measured by Literality.

Keywords: pause-word ratio, Literality, cognitive effort

1. Introduction

The Multiling subset of the CRITT TPR-DB database (Carl et al., 2016a) provides a large corpus of translation process data that facilitates comparisons across different languages and different translation modalities. It assembles user activity data obtained from translation tasks into several languages using a common set of six short English source texts. In particular, keystroke and eye tracking data were recorded during from-scratch translation sessions and during post-editing of machine translations.

In this paper, we focus on translation and post-editing data from the BML12 study for English-to-Spanish (Mesa-Lao, 2014) and the ENJA15 study for English to Japanese (Carl et al., 2016b). By introducing refinements of the pause-word ratio measure of cognitive effort (Lacruz and Shreve, 2014) given by different ranges of pause lengths, we identify different patterns of cognitive effort for the two language pairs. These point to possible differences in the translation process when languages are more or less remote from each other that merit systematic investigation. These differences are potentially of interest to researchers in Natural Language Processing and Text-to-Speech synthesis.

In terms of language structure, Spanish is much closer to English than Japanese is to English. It is therefore to be expected that translation related tasks will be more effortful for English>Japanese than for English>Spanish. In addition, typed production of Japanese using an input method editor (IME) is more complex, and so more effortful, than typed production of Spanish. The expected extra effort involved in English>Japanese translation tasks as compared to English>Spanish translation tasks has been confirmed, for example in Carl et al. (2016b) and Schaeffer et al. (2016).

Linguistic complexity, as opposed to typing complexity, is a factor that contributes to increased cognitive effort expended on translation tasks (Dragsted, 2011). One type of complexity arises from translation entropy, which is computed by HTra in the CRITT TPR database. Translation entropy of a source text word is derived from the number of different translation choices made by different translators. For example, English *seat* could be rendered in several ways in both Spanish and Japanese. Examples include *asiento* or *silla* in Spanish, and 倚子 or 安楽椅 in Japanese. The actual value of HTra for a source text word is determined by the relative frequencies of different target language translation choices in the data set. HTra is 0 for a word for which only a single translation is provided, but increases with the number of translation choices provided. Details can be found in Carl et al. (2016a). Carl et al. (2016b) provide evidence that HTra is higher when the target language is more remote from the English source language. In particular, HTra is about double for English>Japanese than for English> Spanish when the same source texts are used.

Another type of linguistic complexity arises when word alignments differ in the source and target languages. For example, adjective alignments are different in English, where the default is adjective – noun, and in Spanish, where the default is noun – adjective. As another example, verb alignments with the subject and object are different in English (subject – verb – object) and Japanese (subject – object – verb). The CrossS feature in the CRITT TPR database gives information about alignment differences between the source and target texts. Working left to right, as each new word is encountered in the source text CrossS counts how many skips must be made in the target text (positive for skips right, negative for skips left) to reach the position of the corresponding target text word. When words align perfectly in source and target, the CrossS value will be 1. The absolute value |CrossS| will tend to be higher when the syntactic structures of the source and target languages require more re-ordering during translation. A more complete discussion can be found in Carl et al (2106a). Carl et al. (2016b) provide evidence that |CrossS| values are higher when the target language is more syntactically remote from the source language. In particular, |CrossS| values are almost double for English>Japanese than for English>Spanish when the same source texts are used.

HTra is a measure of semantic remoteness and |CrossS| is a measure of syntactic remoteness between source and target language segments. High values of either one are likely to be associated with less literal translations, since

the source text will need to be extensively reframed to be easily comprehensible. Carl et al. (2016a) introduced the Literality[1] feature into the CRITT TPR database as a way to simultaneously measure semantic and syntactic remoteness. At the word level,

$$\text{Literality} = \text{HTra}^* \, |\text{CrossS}|;$$

at the segment level, Literality is computed as the average Literality of all words in the segment. Segment Literality will be lower when there is low variability from one translator to another and when translations of a source text segment differ little from straight word-by-word renderings. It might therefore be expected that translations of segments with higher levels of Literality will be achieved with higher expenditure of cognitive effort. We investigate this line of reasoning using the BML12 and ENJA15 subsets of the CRITT TPR database. When cognitive effort is measured by pause-word ratio (Lacruz and Shreve, 2014), the arguments outlined above suggest that high segment Literality will be associated with high cognitive effort in from-scratch translation and in post-editing for both English>Spanish and English>Japanese.

2. Pause-Word Ratio for Pause Ranges

Pauses in language production have long been associated with cognitive effort. (e.g., Schilperoord, 1996; Dragsted, 2004; Kumpulainen, 2015). Lacruz et al. (2012) and Lacruz and Shreve (2014) introduced pause metrics that have been associated with cognitive effort in a variety of translation tasks. The simplest is the Pause-Word Ratio (PWR), the ratio of the number of pauses in a segment to the number of words in the segment.

Correlations with other measures of cognitive effort mostly have not been sensitive to the precise pause threshold time (but see Schaeffer et al. (2016)), and commonly used thresholds have ranged from 300ms to 5000ms. However, Lacruz and Shreve (2014) observed a variety of pause patterns during post-editing and noted that clusters of short pauses sometimes accompanied particularly effortful edits. They suggested that these short pauses might be associated with monitoring during the post-editing process. By its nature, monitoring occurs relatively late in the cognitive process, so studying short pauses could give insight into later stages of the post-editing or translation processes.

Previous studies have focused on pauses whose lengths were above a certain threshold, specifically above 300ms, above 500ms, above 1000ms, above 2000ms, and above 5000ms. However, in order to isolate the effects of short monitoring pauses in the translation process and to understand better their influence on cognitive effort, we examine here segment level pause-word ratios in the BML12 and ENJA15 studies for pauses whose lengths fall into different time ranges, specifically 300-500ms, 500-1000ms, 1000-2000ms, 2000-5000ms, and at least 5000ms. Loosely speaking, these time ranges correspond to short, medium, and long pauses. Monitoring pauses are likely to be short, in the 300-500ms or 500-1000ms ranges.

[1] Literality is a counter-intuitive term, since highly literal translations can have a low Literality score.

Other studies have focused on the time course of the translation process (cf. Lacruz, 2017). Several have investigated the early stages (cf. Schaeffer et al., 2016). To date, however, little research has been carried out to focus specifically on later stages of the translation process.

There were noticeable differences in the distribution of these range PWRs between the BML12 and ENJA15 data, both for post-editing and for from-scratch translation.

2.1 Translation from Scratch

For the BML12 from-scratch translation data, there is a consistent pattern where the PWR values for a segment show a steady downward trend as the pauses become longer. However, the ENJA15 data consistently shows a different pattern, where the PWR values for a segment rise to a maximum for the 500-1000ms range and then follow a steady downward trend as the pauses become longer. In almost all ranges, the PWR values for English>Japanese translation are higher than those for English>Spanish. This is consistent with previous findings that translation is more effortful when the languages are more remote (Carl et al., 2016b; Schaeffer et al., 2016). For a sample segment, these patterns are illustrated in Figure 1.

It is notable that the shorter pauses, which we have argued are likely to be monitoring pauses, contribute significantly to the pause-word ratio for both language pairs. In other words, it appears that a considerable amount of the cognitive effort expended in translating from scratch is devoted to monitoring what is being produced.

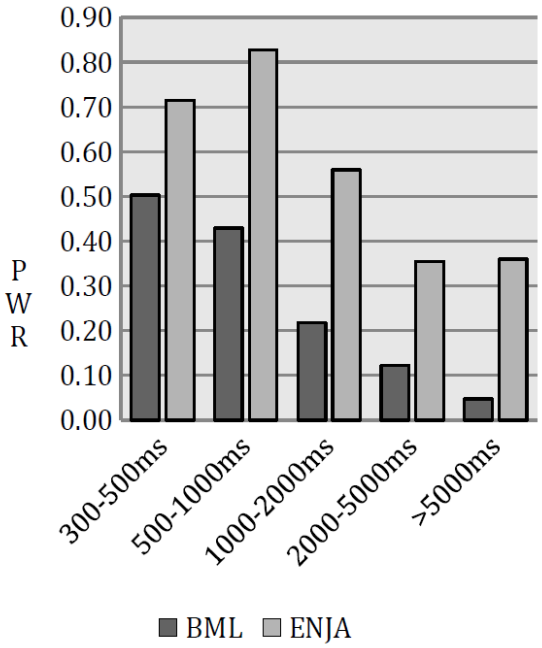


Figure 1: Translation PWR comparisons

2.2 Post-editing

The PWR patterns for post-editing share characteristics of those for from-scratch translation, except that in both language pairs PWR values for the 1000-2000ms pause range tend to be lower than for the longest pauses. Across almost all pause ranges in both language pairs, PWR values for post-editing are lower than for translation from scratch. This is consistent with previous findings that

post-editing is less cognitively effortful than translation from scratch (e.g., Green et al., 2013). The proportional discrepancy between English>Japanese and English> Spanish is more apparent for post-editing than for from-scratch translation, which suggests that the post-editing effort advantage might be less for English>Japanese. See Figure 2 for a representative visualization of the post-editing PWR patterns.

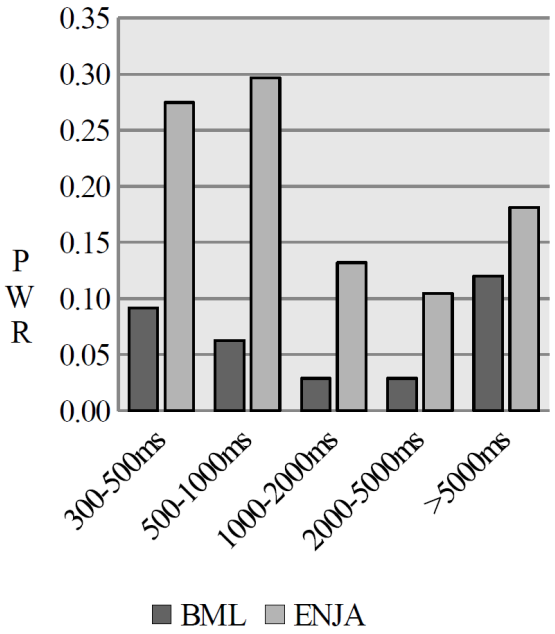


Figure 2: Post-editing PWR comparisions

As with translation from scratch, it is notable that the shorter pauses, presumed to be monitoring pauses, contribute significantly to the pause-word ratio for both language pairs. In other words, it appears that a considerable amount of the cognitive effort expended in post-editing is devoted to monitoring what is being produced. Also notable is the clear up-tick in very long pauses in post-editing as compared with translation from scratch. This is possibly due to the need for post-editors to completely rework areas that are unintelligible in the MT text.

3. Literality, HTra, and CrossS

We investigate more closely how the remoteness of one language from another could influence the cognitive effort involved in monitoring during from-scratch translation and post-editing. For this we focus on how Literality, HTra, and CrossS levels of source text segments correlate with PWR computed from the shortest pause range, 300-500ms., which we assume to measure monitoring effort.

3.1 Translation from Scratch

For translation from scratch, significant correlations ($p < .05$) are marked with a star in Table 1. As expected, Literality, which is a measure of semantic and syntactic remoteness is strongly and significantly correlated with monitoring effort for both English>Spanish and English > Japanese. This effect does not carry over to HTra, which we take as a measure of conceptual remoteness. For each language pair, there is no signifcant correlation between HTra and monitoring effort. The lack of correlation might be attributable to the fact that semantic difficulties will have been largely resolved before translation production begins. The lack of distinction between language pairs breaks down for CrossS, which is a measure of structural remotenesss. For English>Japanese, there is strong and significant correlation between |CrossS| and monitoring effort. However, there is only a moderate, non-significant correlation for English>Spanish. This distinction between the language pairs is a natural one. Since Japanese is structurally remote from English, it seems likely that there would need to considerable on-line monitoring of language production during translation in order to be sure of maintaining structural integrity. On the other hand, monitoring would likely be less intense for English> Spanish, since greater structural closeness will promote more fluency in the translation process.

	Literality	HTra	|CrossS|
BML	.93*	.03	.51
ENJA	.86*	.21	.82*

Table 1: Pearson correlations with 300-500ms PWR. Translation from scratch

3.2 Post-editing

For post-editing, significant correlations ($p < .05$) are marked with a star in Table 2.

	Literality	HTra	|CrossS|
BML	-.93*	-.86*	-.91*
ENJA	-.05	.91*	.16

Table 2: Pearson correlations with 300-500ms PWRs Post-editing

The post-editing correlations are markedly different from the from-scratch translation correlations. For English> Spanish, all the remoteness indicators correlate significantly, but strongly negatively, with monitoring effort. At first sight this might seem strange. However, MT quality from English to Spanish is good, so the machine will successfully resolve most of the difficult semantic and structural problems. These resolutions, which will likely be highly salient to the post-editor and which can usually be accepted quickly, will likely reduce the need to expend much monitoring effort on segments that would have required much more effort to translate from scratch. On the other hand, machine translation solutions for segments where there are few semantic or structural problems to resolve may be much less salient to the post-editor. Out of conscientiousness, the post-editor may search for hidden difficulties, leading to unnecessarily high monitoring effort. This type of paradox has previously been observed in a translation decision task carried out by Lacruz (2017).

The very different correlations for English>Japanese, where the only strong and significant positive correlation is between semantic remoteness and monitoring effort might be attributable to lack of confidence in the less reliable machine translation programs for English>

Japanese leading to multiple monitoring comparisons between the source text and the machine translation. Structural proposals are more transparent than semantic proposals by the machine, so will require less effort to accept or reject, and errors will generally be easier to correct than semantic errors.

4. Conclusion

We introduced the idea of grouping production pauses of similar lengths in order to make a preliminary investigation of potential differences between the translation and post-editing processes when the source and target languages are closely related (English and Spanish) and when they are semantically and structurally remote (English and Japanese). The first main preliminary finding was a difference in the distribution of pauses of different lengths for Spanish and Japanese translations or post-editings of the same English source texts. We argued that this may have been mostly caused by differences at the level of short, monitoring pauses. The second main preliminary finding was that structural and semantic differences between languages have a differential effect on monitoring effort during from-scratch translation and post-editing. This was particularly striking for post-editing.

This preliminary work suggests that it may be interesting to carry out a more in-depth study, possibly using different language pairs, and also raises the possibility that it may be informative to examine different types of behavioral metrics obtained in various translation modalities using a similar time-range approach. For example, it may be instructive to investigate different ranges of gaze times in eye tracking experiments in order to identify the locus of cognitive effort associated with different parts of the translation process.

5. Bibliographical References

Carl, M., Schaeffer, M., and Bangalore, S. (2016a). The CRITT translation process research database. In M. Carl, S. Bangalore, & M. Schaeffer (Eds.), *New Directions in Empirical Translation Process Research*. New York, NY : Springer, pp. 13-54.

Carl, M., Aizawa, A., and Yamada, M. (2016b). English-to Japanese translation vs. Dictation vs. Post-editing : comparing translation modes in a multilingual setting. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 4024-4031, Portorož, Slovenia, May 23-28. European Language Resource Association (ELRA).

Carl, M., Lacruz, I., Yamada, M., & Aizawa, A. (2016c). Measuring the translation process. Proceedings from *22nd Annual Meeting of the Society for Natural Language Processing (NLP 2016), Tohoku University, Sendai, Japan, March 7–10,* (pp. 957–960). Japan: Association for Natural Language Processing.

Dragsted, B. (2004). *Segmentation in translation and translation memory systems: An empirical investigation of cognitive segmentation and effects of integrating a TM system into the translation process*. Copenhagen: Samfundslitteratur.

Dragsted, B. (2012). Indicators of difficulty in translation - Correlating product and process data. *Across Languages and Cultures* 13(1): pp. 81–98.

Green, S., Heer, J., and Manning, C.D. (2013). The efficacy of human post-editing for language translation. In Wendy E. Mackey (Conference Chair) et al., editors, Proceedings of the SIGCHI conference on human factors in computing systems (SIGCHI'13), pages 439-448, Paris, France, April 27 – May 2. Association for Computing Machinery (ACM).

Kumpulainen, M. (2015). On the operationalisation of 'pauses' in translation process research. *Translation & Interpreting*, *7*(1): 47-58.

Lacruz, I., Shreve, G. M., and Angelone, E. (2012). Average pause ratio as an indicator of cognitive effort in post-editing: A case study. In S. O'Brien, M. Simard, & L. Specia (Eds.), Proceedings of AMTA 2012, 10th Conference of the Association for Machine Translation of the Americas, Workshop on Post-Editing Technology and Practice (WPTP 2012), pages 1-10, San Diego, CA, October 28–November 1.

Lacruz, I., and Shreve, G. (2014) Pauses and cognitive effort in post-editing. In S. O'Brien, L. Balling, M. Carl, M. Simard & L. Specia (Eds.), *Post-editing: Processes, technology and applications*. Newcastle upon Tyne (UK): Cambridge Scholars Publishing, pp. 246 – 272.

Lacruz, I. (2017). Cognitive effort in translation, editing and post-editing. In J. Schwieter & A. Ferreira (Eds.), *Handbook of translation and cognition*. Blackwell Handbooks in Linguistics. Malden, MA: John Wiley & Sons, pp. 386-401.

Lacruz, I. (2018). An experimental investigation of stages of processing in post-editing. In I. Lacruz & R. Jääskeläinen (Eds.), *Innovation and expansion in translation process research*. Volume XVIII of the American Translators Association Scholarly Monograph series. Amsterdam/Philadelphia: John Benjamins, pp. 217-240.

Mesa-Lao, B. (2014). Gaze Behaviour on Source Texts: An Exploratory Study Comparing Translation and Post-editing. In S. O'Brien, L. Winther Balling, M. Carl, M. Simard, & L. Specia (Eds.) *Post-editing: Processes, technology and applications*. Newcastle upon Tyne (UK): Cambridge Scholars Publishing, pp. 219-245.

Schaeffer, M., Dragsted, B., Hvelplund, K. T., Balling, L. W., and Carl, M. (2016). Word translation entropy : Evidence of early target language activation during reading for translation. In M. Carl, S. Bangalore, & M. Schaeffer (Eds.), *New Directions in Empirical Translation Process Research*. New York, NY : Springer, pp. 183-210.

Schaeffer, M., Carl, M., Lacruz, I., & Aizawa, A. (2016). Measuring cognitive translation effort with activity units. *Baltic Journal of Modern Computing,* 4(2) : pp. 331–345.

Schilperoord, J. (1996). *It's about time: Temporal aspects of cognitive processes in text production* (Vol. 6). Amsterdam: Rodopi.

Evaluation of Machine Translation Performance Across Multiple Genres and Languages

Marlies van der Wees[1], Arianna Bisazza[2], Christof Monz[1]
[1]Informatics Institute, University of Amsterdam
[2]Leiden Institute of Advanced Computer Science, Leiden University

Abstract

In this paper, we present evaluation corpora covering four genres for four language pairs that we harvested from the web in an automated fashion. We use these multi-genre benchmarks to evaluate the impact of genre differences on machine translation (MT). We observe that BLEU score differences between genres can be large and that, for all genres and all language pairs, translation quality improves when using four genre-optimized systems rather than a single genre-agnostic system. Finally, we train and use genre classifiers to route test documents to the most appropriate genre systems. The results of these experiments show that our multi-genre benchmarks can serve to advance research on text genre adaptation for MT.

Keywords: Machine translation, parallel benchmarks, text genres, genre adaptation

1. Introduction

Text genre differences have shown to affect the output quality of statistical machine translation (SMT) systems: SMT systems trained on one genre often achieve poor performance when used for translating another genre (Foster and Kuhn, 2007; Matsoukas et al., 2009; Wang et al., 2012, among others). In addition, even if different genres in a test set are both present in equal amounts in the bilingual training data, performance differences between the test genres can be large, mostly due to poor model coverage for certain genres (van der Wees et al., 2015a; van der Wees et al., 2015b).

In this paper, we evaluate the impact of genre differences on phrase-based SMT for a diverse set of language pairs, covering both commonly and rarely studied language pairs. For common language pairs, parallel training data is abundant but limited to a few genres such as parliamentary and legal proceedings. For low-resource languages the situation is—by definition—much worse, with very few to no bilingual corpora available. To alleviate this problem, we present in this paper novel parallel training and evaluation corpora covering four genres for four language pairs that we automatically harvested from the web.

Next, we evaluate the usefulness of the newly collected bilingual resources by exploiting them for genre adaptation of SMT systems. Most existing adaptation approaches depend on the availability of provenance information and make the strong assumption that a translation task has known domain, genre or topic that is exploited to adapt the system (Matsoukas et al., 2009; Foster et al., 2010; Bisazza and Federico, 2012; Chen et al., 2013; Chen et al., 2014; Kobus et al., 2016; Sennrich et al., 2016; Freitag and Al-Onaizan, 2016; Chu et al., 2017, among others). While this is a fair assumption in a controlled research setting, it is less realistic in real world applications, such as general-purpose online MT services. In this paper, we provide the SMT system with a test document of unknown origin, and we show that we can use automatic genre classification to guide each test document to the most appropriate pre-trained system. While similar setups have been used in previous work (Xu et al., 2007; Banerjee et al., 2010; Pecina et al., 2011; Wang et al., 2012; Pecina et al., 2015), we are the first to extend this setup to four genres and four language pairs. Finally, we show that an adaptation method based on automatic classifiers also improves translation quality for genres with no parallel training data available.

2. Multi-genre benchmarks

In this section, we describe the construction of multi-genre corpora for four language pairs and four genres, which we obtained using an automated web-harvesting process.

2.1. Language pairs and genres

While most research in MT is evaluated on a small number of well resourced language pairs and domains or genres, we opt for a more balanced distribution of source languages that allows us to measure to what extent our findings for common language pairs generalize to languages with limited resources. We therefore evaluate our experiments in this paper on the following language pairs: *Arabic→English*, *Chinese→English*, *Bulgarian→English*, and *Persian→English*. For each of these language pairs we consider four different genres: *news*, as it can be found in (online) newspapers and in transcripts of broadcast news; *editorial*, covering Op-Ed pieces in (online) newspapers, that represent a subjective, and unlike news less matter-of-fact point of view; *colloquial*, covering informal conversation such as blog comments and Internet forum discussions; and *speech*, covering speeches for which transcripts are available such as TED talks and other public speeches.

2.2. Benchmark construction

For the language pairs and genres of interest, we collect parallel corpora from the web from twenty different websites, each covering at least one of our genres of interest.[1] All websites contain manual translations at the sentence level

[1]Sixteen websites contain news documents, six websites contain editorial documents, eight websites contain colloquial documents, and three websites contain speech transcripts.

	Train set		Dev set		Test set	
Genre	Lines	Tokens	Lines	Tokens	Lines	Tokens
Colloquial	273K	8.9M	1.5K	77.3K	1.5K	73.0K
Editorial	156K	4.7M	1.5K	45.6K	1.5K	47.3K
News	600K	18.0M	1.5K	50.4K	1.5K	48.1K
Speech	140K	3.4M	1.5K	35.7K	1.5K	38.7K
Total	1.2M	35.0M	6.0K	209K	6.0K	207K

(a) Arabic→English data.

	Train set		Dev set		Test set	
Genre	Lines	Tokens	Lines	Tokens	Lines	Tokens
Colloquial	55K	1.7M	1.5K	42.5K	1.4K	35.8K
Editorial	370K	10.2M	1.5K	43.1K	1.5K	42.6K
News	584K	16.4M	1.5K	39.2K	1.5K	35.8K
Speech	146K	3.3M	1.5K	42.6K	1.5K	37.5K
Total	1.2M	31.6M	6.0K	169K	5.9K	152K

(b) Chinese→English data.

	Train set		Dev set		Test set	
Genre	Lines	Tokens	Lines	Tokens	Lines	Tokens
Colloquial	–	–	–	–	1.4K	33.9K
Editorial	–	–	–	–	178	5.1K
News	215K	5.3M	1.2K	30.2K	2.0K	49.5K
Speech	206K	3.9M	1.2K	22.5K	2.0K	44.6K
Total	422K	9.2M	2.4K	52.7K	5.6K	133K

(c) Bulgarian→English data.

	Train set		Dev set		Test set	
Genre	Lines	Tokens	Lines	Tokens	Lines	Tokens
Colloquial	629K	16.4M	1.5K	40.3K	1.5K	37.7K
Editorial	–	–	–	–	600	19.4K
News	618K	16.8M	1.5K	44.5K	1.5K	47.4K
Speech	119K	2.5M	1.5K	31.2K	1.5K	35.6K
Total	1.4M	35.7M	4.5K	116K	5.1K	140K

(d) Persian→English data.

Table 1: Specifications of the harvested multi-genre training, development and test sets for four language pairs. Tokens are counted on the English side. We make the evaluation corpora available at `http://ilps.science.uva.nl/resources/genre-benchmarks`.

between English and one or more other languages. Unfortunately it is not always known which language is the original language and which language is a translation, especially for user comments, which may be written in any of a website's supported language.

Collection of the parallel data is done in an automated fashion. For each website, we provide URLs of one or more overview pages (e.g., a sitemap or an archive page) and then extract from it a list of URLs containing actual text documents. Since all websites support at least English, we start with collecting English documents, regardless of the original language of the websites' documents. Next, we identify translations of these English pages by (i) following direct links, such as 'read this article in Arabic,' or (ii) replacing the language abbreviation in the url, e.g., replacing '`en-US`' with '`ar`'.

To determine the genre of each text we use categories indicated on the respective website. For example, websites that support news articles and user comments (i.e., the genres news and colloquial), have clear website sections for these different genres.

While not all language-genre combinations are equally common, we can construct at least a translation test set for each of the four genres in each of the four language pairs. To do so, we first create sentence-parallel corpora using a combination of Moore's sentence alignment (Moore, 2002) and Champollion sentence alignment (Ma, 2006). We then organize the collected bilingual data into training, development, and test sets, such that each portion contains documents from non-overlapping time periods.[2] We tokenize all Arabic data using MADA (Habash and Rambow, 2005), segment the Chinese data following (Tseng et al., 2005), and use a simple in-house tokenizer for the other languages. The total numbers of foreign→English sentence pairs for the four genres and four language pairs are listed in Tables 1a–1d. In addition, Table 2 shows English example sentences for each of the four genres.

Genre	Example sentence(s)
Colloquial	Ministers should be sitting and attending the oath, like in Italy.
Editorial	This may sound like pie in the sky, but we have already tasted it in Africa, where Sierra Leone's agenda for prosperity 2013–2017 and the Liberia Vision 2030 exemplify the potential of such programs.
News	She is not only the first Saudi woman to ever attempt the climb but also the youngest Arab to make it to the top of the world's highest peak.
Speech	These are just a few of the milestones of recent progress. I have another reason to be optimistic. I know global health is guided by the right values.

Table 2: English example sentences for four genres in the web-harvested evaluation corpora.

3. Evaluating genre differences in SMT

In this section, we use our newly assembled resources to evaluate SMT performance across different genres and language pairs.

[2]The benchmarks are available for download at `http://ilps.science.uva.nl/resources/genre-benchmarks`.

Test genre	Baseline	SMT system optimized for				Combined best BLEU
		Coll.	Edit.	News	Speech	
Coll.	11.7	**13.8**	10.8	11.7	11.2	**17.9** (+1.1)
Edit.	22.6	19.6	**23.5**	21.6	21.0	
News	22.6	20.2	21.7	**23.2**	21.2	
Speech	11.5	11.5	11.1	11.0	**11.7**	
All	16.8	16.6	16.4	16.6	16.0	

(a) Arabic→English results.

Test genre	Baseline	SMT system optimized for				Combined best BLEU
		Coll.	Edit.	News	Speech	
Coll.	11.4	**11.6**	11.3	10.7	11.3	**13.9** (+0.5)
Edit.	15.5	14.9	**16.3**	14.6	14.3	
News	13.3	12.8	13.3	**13.5**	12.4	
Speech	12.8	12.5	12.5	12.1	**13.9**	
All	13.4	13.1	13.4	12.7	13.2	

(b) Chinese→English results.

Test genre	Baseline	SMT system optimized for				Combined best BLEU
		Coll.	Edit.	News	Speech	
Coll.	**29.1**	–	–	28.0	28.1	**33.4** (+0.6)
Edit.	24.7	–	–	**25.4**	21.3	
News	39.8	–	–	**40.4**	34.7	
Speech	27.4	–	–	25.8	**28.4**	
All	32.8	–	–	31.9	30.5	

(c) Bulgarian→English results.

Test genre	Baseline	SMT system optimized for				Combined best BLEU
		Coll.	Edit.	News	Speech	
Coll.	22.4	**22.5**	–	20.9	21.5	**22.3** (+0.4)
Edit.	**15.7**	15.2	–	15.6	15.1	
News	24.2	22.3	–	**24.3**	23.0	
Speech	21.3	19.5	–	20.7	**22.6**	
All	21.9	20.8	–	21.3	21.5	

(d) Persian→English results.

Table 3: Translation quality in BLEU of four test genres using genre-optimized systems and a genre-agnostic baseline. Best results for each test set genre are boldfaced. 'Combined best BLEU' indicates the overall BLEU score when combining the bold-faced results of all test genres in a single test set, followed by the difference with the genre-agnostic system.

3.1. Experimental setup

All SMT systems in this paper are trained using an in-house phrase-based SMT system similar to Moses (Koehn et al., 2007). To train our systems, we use our web-crawled corpora, supplemented with commonly used training data, if available: LDC corpora for Arabic→English and Chinese→English, and Europarl data (Koehn, 2005) for Bulgarian→English. In addition, we use a 5-gram language model that linearly interpolates various Gigaword subcorpora with the English sides of the bilingual training corpora. To evaluate the effect of our new bilingual resources, we do not vary the language model between experiments.

In order to create genre-specific SMT systems, we have to adequately use the available data. Simply concatenating the different corpora yields a general SMT system that performs reasonably well across a variety of genres, i.e., those covered in the training data, but is not optimal for each individual genre. Since we aim to create genre-specific systems, we use the fill-up technique proposed by Bisazza et al. (2011), in which we combine models trained on a particular genre with models trained on the remaining training corpora. Using this model combination technique, an additional feature is learned that favors genre-specific models, and 'backs off' to additional (out-of-genre) models for phrases that are unseen in the genre of interest. For instance, to train our news translation system, we train two phrase tables: one using all news data and one using all non-news data. We use the latter to complement the first with phrase pairs that are not covered in the first.

Following the above strategy, we can train genre-specific systems for all genres for which we have training data. Genres not covered in the training data have to be translated using a system trained on a mixture of genres or on one of the other genre-specific systems. For example, editorial Persian→English data is scarce, so for Persian editorial documents we have to resort to our colloquial, news, speech or mixed system. In addition to using the fill-up approach, we tune each genre-specific system on a development set covering only the genre of interest.

3.2. Results

Tables 3a–3d show the translation quality results for all language pairs. For each language pair, we measure case-insensitive BLEU (Papineni et al., 2002) for our four test genres with the available genre-specific systems as well as the genre-agnostic system. Note that some Arabic→English and Chinese→English BLEU scores might be lower than those reported in literature since our test data contains only a single reference translation.

The results confirm our expectation that the various test set genres benefit from being translated using a genre-optimized system rather than using a general system: generally, the highest BLEU scores are located on the diagonal of each table. In cases where no genre-specific system is available, we see that the best results are mostly obtained using the general system rather than a system optimized for a different genre.

4. Genre adaptation using automatic classifiers

We observed that translation quality is usually best when translating each genre using its respective genre-specific baseline system. This motivates the hypothesis that translation of a mixture-of-genre test set can be improved by using a genre classifier, which routes test sentences or documents to the most appropriate MT system. Adapting an MT system using this strategy involves two steps: training accurate

Genre	Arabic→English system		
	Genre-agnostic	Manual oracle	Genre-classified
Colloquial	11.7	13.8▲	13.8▲
Editorial	22.6	23.5▲	23.5▲
News	22.6	23.2▲	23.2▲
Speech	11.5	11.7	11.6
Overall	16.8	17.9▲	17.8▲

(a) Arabic→English results.

Genre	Chinese→English system		
	Genre-agnostic	Manual oracle	Genre-classified
Colloquial	11.4	11.6	11.5
Editorial	15.5	16.3▲	16.3▲
News	13.3	13.5	13.6
Speech	12.8	13.9▲	14.0▲
Overall	13.4	13.9▲	13.9▲

(b) Chinese→English results.

Genre	Bulgarian→English system		
	Genre-agnostic	Manual oracle	Genre-classified
Colloquial	29.1	29.1	28.6▼
Editorial	24.7	25.4△	25.4△
News	39.8	40.4▲	40.4▲
Speech	27.4	28.4▲	28.4▲
Overall	32.8	33.4▲	33.1▲

(c) Bulgarian→English results.

Genre	Persian→English system		
	Genre-agnostic	Manual oracle	Genre-classified
Colloquial	22.4	22.5	22.5
Editorial	15.7	15.7	15.6
News	24.2	24.3	24.2
Speech	21.3	22.6▲	22.6▲
Overall	21.9	22.3▲	22.1▲

(d) Persian→English results.

Table 4: Translation results in BLEU of baseline and genre-adapted systems. Manual oracle results are combined from several genre-optimized systems using manual genre labels of the test documents, see Tables 3a–3d. Statistically significant differences are indicated with △ or ▽ at the $p \leq 0.05$ level and with ▲ or ▼ at the $p \leq 0.01$ level.

genre classifiers (§4.1.) and incorporating these classifiers into an end-to-end MT pipeline (§4.2.).

Adaptation using automatic classifiers has been applied in a few previous efforts (Xu et al., 2007; Banerjee et al., 2010; Wang et al., 2012). However, all three methods limit their application to one language pair covering two provenance-based domains, which are typically very distinct, e.g., patents versus 'generic'. To the best of our knowledge, we are the first to extend this setup up to four genres and four language pairs, where documents within a genre originate from a variety of sources.

4.1. Training genre classifiers

Since we apply our genre classifiers to different languages, we aim at developing a single classification procedure that can be used on any source document regardless of the language it is written in. For this purpose, we apply our experiments to three languages: Arabic, Chinese, and English. To train the classifiers we randomly select documents from the training data listed in Tables 1a and 1b. The complete selection comprises 1,000 documents per genre, thus enforcing equal prior classification probabilities for all genres.

We train genre classifiers with Support Vector Machines (SVM) with linear kernels, using the WEKA data mining software (Hall et al., 2009). As our features, we use the union of the 500 most common words per genre. We do not remove stopwords since they have a high potential to distinguish between various genres, which is long known in text genre classification literature (Karlgren and Cutting, 1994; Kessler et al., 1997; Stamatatos et al., 2000; Dewdney et al., 2001). Using this classifier-feature combination, the classification accuracy on the documents in the test portion of our web-crawled corpora is 97.0%, 83.9%, and 88.1% for Arabic, Chinese, and English, respectively.

4.2. Genre adaptation experiments

Armed with accurate genre classifiers, we next classify for each document in the test set its genre, and guide it to the most appropriate SMT system. Note that while we do have access to the true genre labels in this controlled research scenario, we intentionally mimic a more realistic situation in which an incoming test document has unknown origin.

Figures 4a–4d show the translation quality in BLEU for all language pairs using (i) a genre-agnostic baseline system trained and tuned on a mixture of genres, (ii) several genre-specific systems which we combine manually and refer to as our 'oracle' system, and (iii) several genre-specific systems which we combine using automatic genre classifiers. We measure statistical significance with respect to the genre-agnostic baseline using approximate randomization (Riezler and Maxwell, 2005), reporting significant differences at the $p \leq 0.05$ (△/▽) or $p \leq 0.01$ (▲/▼) level.

For Arabic→English and Chinese→English (Tables 4a and 4b, respectively), we train our classifiers on four genres with a balanced prior distribution. Our Arabic genre classifier achieves near-perfect classification accuracy (97%), which is reflected by BLEU scores that are very similar to the oracle system. Our best Chinese genre classifier yields lower accuracy (84%), however BLEU scores of the genre-classified system do not suffer from this sub-optimal classification performance. On closer inspection we see that some documents actually benefit from being translated by a different genre-optimized system, for example the Chinese news documents classified as editorial improve with 0.4 BLEU if translated using the editorial system.

Next, we look at the languages for which not all genres are covered in the training data. Consequently, we can only train classifiers for two (Bulgarian) or three (Persian) of the

genres in the test set. For the remaining test genres, the predicted genre will be one of the genres in the training data, and translation is performed using the corresponding genre-specific system. Note that the genre-agnostic baseline system is never recommended based on classifier predictions, despite sometimes being the best option.

Table 4c shows the end-to-end results for Bulgarian→English translation. The Bulgarian genre classifier achieves 100% accuracy on news and speech. The editorial test documents are all classified as news, which is advantageous for the SMT output quality. Genre predictions for the colloquial test documents are distributed evenly over news and speech, achieving a BLEU score of 28.6. While the genre-agnostic system performs better (29.1), the result using an automatic classifier is superior to translating all colloquial documents with either the news (28.1) or the speech (28.0) system. This finding indicates that automatic genre classification can even be profitable if no training data for a given genre is available.

Finally, Table 4d shows the end-to-end results for Persian→English translation. The Persian classifier achieves 90% accuracy on the genres covered in the training data; colloquial, news, and speech. However, BLEU scores using the genre-agnostic and the genre-optimized systems are very similar for all genres except speech. Improvements using the genre-classified system are therefore small.

5. Conclusions and future work

In this paper, we have presented parallel evaluation corpora covering four genres for four language pairs. We used these multi-genre benchmarks to show that BLEU differences between genres can be large and that, for all genres and all language pairs, translation quality improves when using four genre-optimized systems rather than a single genre-agnostic system. Finally, we trained and used genre classifiers to route test documents to the most appropriate genre systems, and showed that this setup can be used to successfully adapt SMT systems to four different genres, even for genres with no available parallel training data.

While experiments in this paper are limited to phrase-based SMT, they can also be applied to neural MT, for which current research is still limited to a few language pairs and domains.

Acknowledgements

This research was funded in part by NWO under project numbers 639.022.213, 612.001.218, and 639.021.646.

6. Bibliographical References

Banerjee, P., Du, J., Li, B., Kumar Naskar, S., Way, A., and van Genabith, J. (2010). Combining multi-domain statistical machine translation models using automatic classifiers. In *Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*.

Bisazza, A. and Federico, M. (2012). Cutting the long tail: Hybrid language models for translation style adaptation. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 439–448.

Bisazza, A., Ruiz, N., and Federico, M. (2011). Fill-up versus interpolation methods for phrase-based SMT adaptation. In *Proceedings of the 8th International Workshop on Spoken Language Translation*, pages 136–143.

Chen, B., Kuhn, R., and Foster, G. (2013). Vector space model for adaptation in statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1285–1293.

Chen, B., Kuhn, R., and Foster, G. (2014). A comparison of mixture and vector space techniques for translation model adaptation. In *Proceedings of the 11th Conference of the Association for Machine Translation in the Americas*, pages 124–138.

Chu, C., Dabre, R., and Kurohashi, S. (2017). An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 385–391.

Dewdney, N., VanEss-Dykema, C., and MacMillan, R. (2001). The form is the substance: classification of genres in text. In *Proceedings of the Workshop on Human Language Technology and Knowledge Management*.

Foster, G. and Kuhn, R. (2007). Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 128–135.

Foster, G., Goutte, C., and Kuhn, R. (2010). Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 451–459.

Freitag, M. and Al-Onaizan, Y. (2016). Fast domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06897*.

Habash, N. and Rambow, O. (2005). Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 573–580.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Karlgren, J. and Cutting, D. (1994). Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th International conference on Computational Linguistics*, pages 1071–1075.

Kessler, B., Numberg, G., and Schütze, H. (1997). Automatic detection of text genre. In *Proceedings of the eighth conference of the European chapter of the Association for Computational Linguistics (EACL)*, pages 32–38.

Kobus, C., Crego, J., and Senellart, J. (2016). Domain control for neural machine translation. *arXiv preprint arXiv:1612.06140*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical

machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the X Machine Translation Summit*, pages 79–86.

Ma, X. (2006). Champollion: A robust parallel text sentence aligner. In *LREC 2006: Fifth International Conference on Language Resources and Evaluation*, pages 489–492.

Matsoukas, S., Rosti, A.-V. I., and Zhang, B. (2009). Discriminative corpus weight estimation for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 708–717.

Moore, R. C. (2002). Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas*, pages 135–144.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Pecina, P., Toral, A., Way, A., Papavassiliou, V., Prokopidis, P., and Giagkou, M. (2011). Towards using web-crawled data for domain adaptation in statistical machine translation. In *Proceedings of the 15th Conference of the European Association for Machine Translation*, pages 297–304.

Pecina, P., Toral, A., Papavassiliou, V., Prokopidis, P., Tamchyna, A., Way, A., and van Genabith, J. (2015). Domain adaptation of statistical machine translation with domain-focused web crawling. *Language resources and evaluation*, 49(1):147–193.

Riezler, S. and Maxwell, J. T. (2005). On some pitfalls in automatic evaluation and significance testing for MT. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64.

Sennrich, R., Haddow, B., and Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 86–96.

Stamatatos, E., Fakotakis, N., and Kokkinakis, G. (2000). Text genre detection using common word frequencies. In *Proceedings of the 18th conference on Computational linguistics*, pages 808–814.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D., and Manning, C. (2005). A conditional random field word segmenter. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*, pages 168–171.

van der Wees, M., Bisazza, A., and Monz, C. (2015a). Five shades of noise: Analyzing machine translation errors in user-generated text. In *Proceedings of the First Workshop on Noisy User-generated Text (WNUT)*, pages 28–37.

van der Wees, M., Bisazza, A., Weerkamp, W., and Monz, C. (2015b). What's in a domain? Analyzing genre and topic differences in statistical machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Learning (ACL-CoNLL)*, pages 560–566.

Wang, W., Macherey, K., Macherey, W., Och, F., and Xu, P. (2012). Improved domain adaptation for statistical machine translation. In *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas*.

Xu, J., Deng, Y., Gao, Y., and Ney, H. (2007). Domain dependent statistical machine translation. In *Proceedings of the XI Machine Translation Summit*, pages 515–520.

A Multilingual Dataset for Evaluating Parallel Sentence Extraction from Comparable Corpora

Pierre Zweigenbaum
LIMSI, CNRS,
Université Paris-Saclay,
F-91405 Orsay, France
pz@limsi.fr

Serge Sharoff
University of Leeds
Leeds, United Kingdom
s.sharoff@leeds.ac.uk

Reinhard Rapp
Magdeburg-Stendal University
of Applied Sciences and
University of Mainz, Germany
reinhardrapp@gmx.de

Abstract

Comparable corpora can be seen as a reservoir for parallel sentences and phrases to overcome limitations in variety and quantity encountered in existing parallel corpora. This has motivated the design of methods to extract parallel sentences from comparable corporad. Despite this interest and work, no shared dataset has been made available for this task until the 2017 BUCC Shared Task. We present the challenges faced to build such a dataset and the solutions adopted to design and create the 2017 BUCC Shared Task dataset, emphasizing issues we had to cope with to include Chinese as one of the languages. The resulting corpus contains a total of about 3.5 million distinct sentences in English, French, German, Russian, and Chinese, mostly from Wikipedia. We illustrate the use of this dataset in the shared task and summarize the main results obtained by its participants. We finally outline remaining issues.

Keywords: Comparable corpora, parallel sentences, parallel sentence extraction, cross-language similarity, annotated corpus

1. Parallel Sentence Extraction from Comparable Corpora

Parallel sentences are the fuel needed to train machine translation systems. Large parallel corpora have been obtained from international bodies or collected from the Web. However, they only cover a small subset of the variety of language pairs, domains and genres that are found in language. Besides, because by construction at least half of the sentences in these corpora are the result of (human) translation, they are likely to display translation biases such as calques and other such phenomena.

Comparable corpora are (typically multilingual) corpora selected with similar criteria such as domain, genre, time period. In contrast to parallel corpora, they display much more variety and are normally original texts rather than translations. They hold much promise therefore as a complement to parallel texts for machine translation and other applications.

One way in which comparable corpora have been used to help machine translation is by spotting parallel sentences that occur naturally in these corpora, and using these sentence pairs to extend parallel corpora (Munteanu et al., 2004). This has motivated research into methods that aim to perform this task, such as (Utiyama and Isahara, 2003; Munteanu et al., 2004; Abdul-Rauf and Schwenk, 2009; Smith et al., 2010). This task is usually called *Parallel Sentence Extraction from Comparable Corpora*.

It is however difficult to compare earlier work and assess progress because of the absence of a shared dataset with gold standard annotations. Some past shared tasks addressed related objectives. Cross-language plagiarism detection in PAN (Potthast et al., 2012) aims to spot text that has been translated into a target language and reused in (inserted into) text in that target language. It is therefore quite close to our task. However, plagiarism detection can take advantage of differences in style between the original target text and the translated text, and of intrinsic properties of 'translationese'. This is not the case in our task, where all sentences are expected to be original. Cross-language text similarity as in SemEval 2016 (Agirre et al., 2016) assesses the level of semantic similarity of pairs of sentences on a given scale. It is also close to our task. Nevertheless, it has been proposed with already paired sentences instead of large monolingual corpora, thus removing the sentence spotting stage. Bilingual document alignment in a large Web collection has been proposed in WMT 2016 (Buck and Koehn, 2016). However, on the one hand it addressed documents instead of sentences; and on the other hand, it included meta-information in the form of document URLs, a property that we want to avoid.

This highlights the need for a publicly available dataset that would make it possible to compare methods that extract parallel sentences from comparable corpora. This paper describes the principles according to which we designed such a corpus, their implementation, the resulting corpus and a first use of that corpus in a shared task. This corpus was built in the context of the BUCC 2017 Shared Task described in (Zweigenbaum et al., 2017). The present paper provides more detail about our motivation and design criteria, about the rationale we followed to implement these design criteria, and about the processing of the Chinese part of the corpus.

2. A Dataset for Parallel Sentence Extraction from Comparable Corpora

2.1. Desiderata for a Dataset for the Task

We aimed to build a bilingual corpus to measure progress on the identification of parallel sentences in monolingual corpora. This led us to the following desiderata and design choices.

No metadata. We wish to focus on the cross-language comparison of sentence contents. Instead, most work so far has relied to a more or less large extent on meta-information: belonging to paired documents such as

linked Wikipedia pages or Web pages, sharing images or links to external documents, news published in a close time frame, etc. Our target corpus should not give such clues, and should therefore include no metadata on the documents in which the sentences are found.

Realistic size. Spotting parallel sentences is useful when performed at scale. We therefore aimed at a corpus with at least millions of sentences: although not especially large by today's standards, this already requires to use scalable algorithms.

Natural text. We wished to provide natural text rather than a simple list of sentences. First, this makes the task more realistic, since extraction of sentences from comparable corpora happens in the context of complete documents. Second, we can expect that the document context is likely to influence assessment of comparability between the sentences. This desideratum is probably less necessary, but we considered it would make the corpus closer to what the task should address.

Known true positives. Since we want to be able to evaluate system results, we must have a gold standard. This is the most challenging part of building such a corpus: recall that ultimately we want systems to spot pairs of sentences that occur naturally in a pair of monolingual corpora and happen to be translations of each other. We know of no such situation in which such sentence pairs would be marked in some way.

2.2. Pragmatic Choices

2.2.1. Creating an Artificial Corpus

Spotting naturally occurring sentence pairs in comparable corpora, if performed by humans, can be extremely time consuming: exhaustively spotting such pairs in, say, two corpora of 400,000 sentences each may require the examination of 160 billion sentence pairs. When preparing a gold standard is not feasible a priori, some shared tasks, e.g. ad hoc information retrieval, have resorted to pooling of system results then a posteriori human evaluation. A posteriori, we know that BUCC 2017 Shared Task participants produced a few hundred thousand sentence pairs: this is many orders of magnitude below that of the above-mentioned a priori evaluation, but is still sizeable. We did not have human resources to allocate to such a human evaluation either. We therefore decided to design a synthetic corpus containing controlled parallel sentences. We performed this by inserting known parallel sentence pairs into existing monolingual corpora.

We chose Wikipedia articles (20161201 dumps [1]) as our monolingual corpora and News Commentary (v11[2]) as our source for parallel sentence pairs.

In terms of domains, although in principle Wikipedia covers all domains, it over-represents named entities, specifically contemporary people and locations. The domain of News Commentary is that of commentaries on international news, hence it mentions a large number of contemporary people and locations. This results in a reasonable match of the domains of the two corpora.

Wiki genres	**%**	**NC genres**	**%**
Encyclopedic	66.4%	Argumentative	84.3%
Hard news	16.9%	Academic	2.6%
Argumentative	4.2%	Hard news	1.9%
Reviews	2.7%	Personal	1.6%
Academic	2.5%	Encyclopedic	1.4%

Table 1: Most common genres in our sources

We also performed a quantitative analysis of the two datasets in terms of topics and genres. Extraction of keywords and comparison of the cosine similarity between the resulting vectors (Sharoff, 2013) gives an estimate of how similar the documents are across the corpora. The inter-decile range of the cosine similarity scores between the News Commentary texts and their nearest Wikipedia counterparts is $[0.971, 0.980]$, i.e., for any News Commentary text it is nearly guaranteed that there is a sufficiently similar Wikipedia text.

Using the genre classifier from (Sharoff, 2018) we also assessed the genre composition of the two corpora, see Table 1. Even though precision of automatic genre classification for different genres varies from 65% to 85%, the results indicate a general trend confirming that the News Commentary corpus corresponds of a considerable portion of Wikipedia in terms of genres, albeit with a different distribution. Wikipedia contains many news-like or argumentative texts, which are similar to the News Commentary corpus, while the latter also contains some encyclopedic introductions and research-like texts similar to those found in Wikipedia.

The following two examples illustrate the similarities in both topics and genres between the two sources: Wikipedia id=13811803 *"Saltwater Keynesian economists" argue that business cycles represent market failures, and should be counteracted through discretionary changes in aggregate public spending and the short-term nominal interest rate. "Freshwater economists" often reject the effectiveness of discretionary changes in aggregate public spending as a means to efficiently stabilize business cycles.*

News Commentary: *The Chicago School claims that real-world market economies produce roughly efficient (so-called "Pareto optimal") outcomes on which public policy cannot improve. Thus, any state intervention in the economy must make someone worse off. The MIT School, by contrast, argues that real-world economies are afflicted by pervasive market failures, including imperfect competition and monopoly, externalities associated with problems like pollution, and an inability to supply public goods such as street lighting or national defense.*

In the remainder of this section we use French and English as a running example of a language pair. For convenience, we often call 'monolingual sentence' a sentence found in the monolingual corpora and 'parallel sentence' a sentence from the parallel corpora.

[1] http://ftp.acc.umu.se/mirror/wikimedia.org/dumps/

[2] http://www.casmacat.eu/corpus/news-commentary.html

2.2.2. Inserting Parallel Sentences in Monolingual Corpora

The inserted parallel sentence pairs should not be trivially detectable in the monolingual corpora. In other words, these sentences should be coherent with the context in which they are inserted. We aimed at topical coherence by looking in the monolingual corpora for sentences with similar contents to the parallel sentences and using the spotted sentences as insertion points. To perform this efficiently, we used a search engine to index each English sentence of the monolingual corpus (English Wikipedia dump, converted to text and split into sentences) and each French sentence of the monolingual corpus (French Wikipedia dump, converted to text and split into sentences). We used the Solr search engine and queried it for each sentence pair in the parallel corpus (French-English News Commentary) to find the most similar French sentence and English sentence for this pair.[3] If two similar enough sentences were found, we inserted the English parallel sentence after the matching English sentence in the monolingual corpus and the French parallel sentence after its matching French sentence in the monolingual corpus. Similarity constraints were enforced on the one hand by the Solr query parameters and on the other hand by post-filters including: a length in words (before stopword removal) in the range $[10, 20]$; a length ratio in the range $[0.8, 1.2]$.

In early experiments, we observed that the cohesion of the resulting sequence of two sentences was better if the start of the inserted sentence contained the common words with the pre-existing monolingual sentence (those words that make the sentences similar). To favor this, we decided to truncate the queries built from parallel sentences to the first T words. T was set to 5 words based on observations in these experiments.

2.2.3. Making Inserted Sentences Less Conspicuous

In early experiments, we realized that very short parallel sentences might happen to be inserted among much larger monolingual sentences, or the reverse. This would increase the risk that such sentences might break the cohesion of the original text. To reduce this risk, we acted on the distribution of sentence lengths in both the monolingual and parallel sentences: we excluded sentences shorter than 10 words and longer than 20 words. This range of lengths covered a large percentile of the original sentences, and is typical of what Machine Translation systems address.

We also realized that due to construction idiosyncrasies, Wikipedia texts had specific distributions of typographical features such as the presence of some typographical quotation marks; they were also subject to conversion issues that created systematic clues of their origin. This was notably caused by the use of Wikipedia templates, that were particularly numerous in the French Wikipedia. We endeavored to remove such idiosyncrasies by revisiting the Wikipedia conversion workflow. We added our own extensions to an existing Wikipedia conversion tool, WikiExtractor.py,[4]. We included sentence splitting based on NLTK, and removed the sentences that contained a Wikipedia template.

Removing sentences brought the additional advantage of making the original text slightly less cohesive: in that context of slightly reduced cohesion, the potential cohesion issues incurred by the addition of (parallel) sentences were likely to be less noticeable.

2.2.4. Controlling Unknown True Positives

The insertion of known parallel sentences aimed to control the true positives present in the datasets we were building. Parallel sentences might however already exist in the pair of monolingual corpora we started from. Indeed, the true nature of the task would be to find these pre-existing, naturally occurring parallel sentences. But we were instead aiming to populate our monolingual corpora with known parallel sentence pairs. We therefore needed to prevent as much as possible naturally occurring parallel sentence pairs from remaining in our monolingual corpora. The strategy we adopted in this purpose was to desynchronize our comparable corpora. Since we started from Wikipedia articles in two languages, we knew that interlinked articles would be highly likely to contain such parallel sentences: this is indeed a property that is often desired by past work on parallel sentence extraction. This is also how our previous shared task on detection of comparable texts has been setup (Sharoff et al., 2015): the gold standard was based on the iwiki links.

In contrast to such work, we built pairs of monolingual corpora which never contained two interlinked Wikipedia articles. This was also in line with our desideratum not to include meta-information on the sentences, such as being found in two interlinked articles.

The main drawback in doing so is that the most comparable pairs of documents for a given language pair are removed from the pairs of corpora we built: only one out of two interlinked pages can be kept in one of our corpora. This reduces the comparability of our datasets. However, the two sides of each dataset still share several dimensions along which they are comparable:

- They belong to the same genre distribution, mainly 'encyclopedic article' (see Table 1).
- They were written in the same time period: contemporary prose.
- Because Wikipedia has a dense coverage of many topics, removing one page does not suppress a topic entirely.

As we discuss later, some of the participant systems did detect original sentence pairs that were translations of each other, i.e., that we had not artificially inserted into the monolingual corpora. This is another clue of the comparability of these corpora.

2.2.5. Preparing Training and Test Splits

A shared task dataset needs to have separate training and test splits. Because of the way we selected insertion points for parallel sentences in our initial monolingual corpora, the two sentences of a parallel pair may occur in quite different

[3] We used the following Solr parameters: efType=edismax, qs=5, ps=5, ps2=5, mm=70%, stopwords=true.

[4] https://github.com/attardi/wikiextractor

Pair	Sample (2%)			Training (49%)			Test (49%)		
	fr	en	gold	*fr*	en	gold	*fr*	en	gold
de-en	32593	40354	1038	413869	399337	9580	413884	396534	9550
fr-en	21497	38069	929	271874	369810	9086	276833	373459	9043
ru-en	45459	72766	2374	460853	558401	14435	457327	566356	14330
zh-en	8624	13589	257	94637	88860	1899	91824	90037	1896

Table 2: Corpus statistics: number of monolingual sentences (*fr*, en) and of parallel pairs (gold) for each split and each language pair. The *fr* column stands for the non-English language in each pair. Reprinted from (Zweigenbaum et al., 2017).

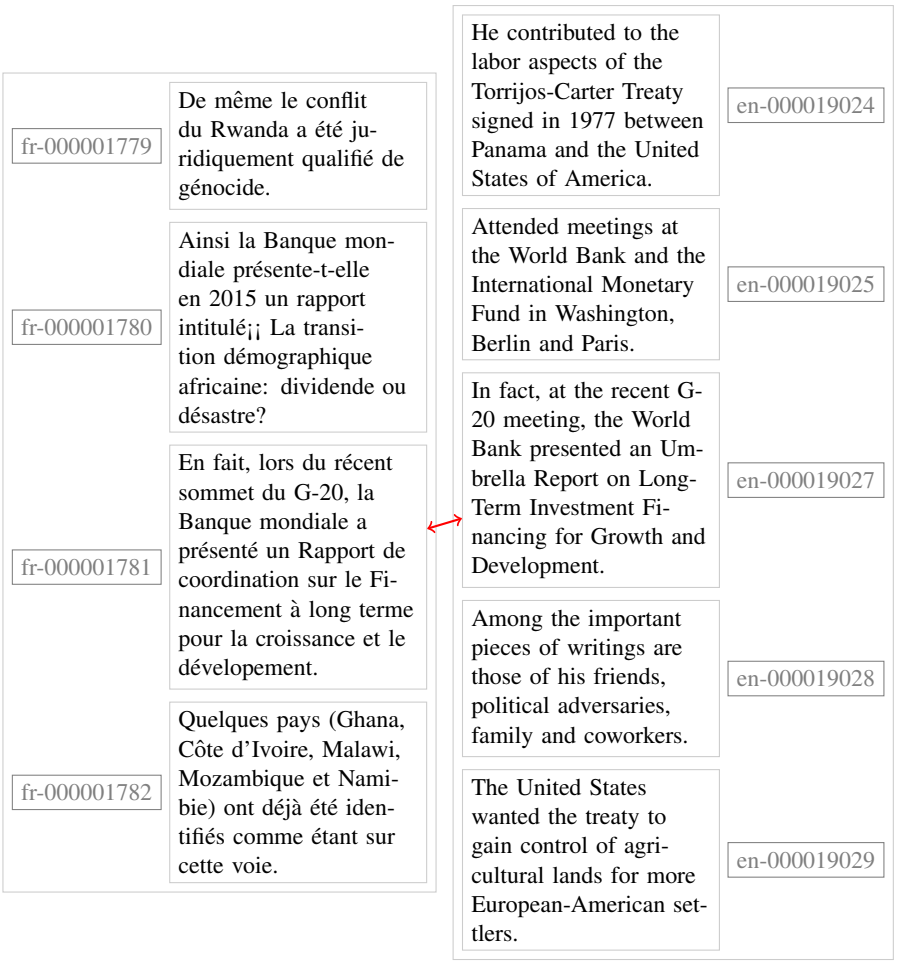


Figure 1: Excerpt from the English-French corpus: fr-000001781 and en-000019027 are inserted parallel sentences.

parts of these monolingual corpora. Splitting the resulting corpora after insertion was therefore liable to separate a large proportion of sentence pairs. Therefore we randomly split the documents of the corpora before parallel sentence pair insertion. An additional constraint was the need to separate interlinked articles. This constraint was taken into account at splitting time, correcting the random assignment of a document to a given split (and redrawing another assignment) if an interlinked document was already present in this split. We actually split the corpora into three parts: a small sample split shown on the shared task Web site, and training and test splits that required registration prior to download. We arbitrary chose 2% of the total for the size of the sample split, and gave training and test half of the remaining data (i.e., 49% each). The corpus preparation process was then performed on each split.[5]

We applied this process to five languages: Chinese (zh), English (en), French (fr), German (de), Russian (ru); this produced four bilingual datasets (see Table 2). Figure 1 shows an example drawn from the English-French dataset.

2.3. The Case of Chinese

Adding a language to our corpora, i.e., providing an additional language paired with English, requires the following data and components:

- A monolingual corpus: Wikipedia is a possible candidate for a large number of languages.
- A parallel corpus with English on one side: in the

[5] An anonymous reviewer rightly pointed out that a development split could have been provided too.

present work we used News Commentary 2016, which pairs English with eleven languages: Arabic, Dutch, Chinese, Czech, German, Spanish, French, Italian, Japanese, Portuguese, and Russian.

- A configuration for indexing and search in the Solr search engine, typically based on a tokenizer, stop words, and possibly more language components.
- Constraints on the range of sentence lengths.

We report here how we included Chinese data in the present corpus.

The Chinese writing system does not separate words with spaces[6]. This raises issues for tokenization that have consequences on our dataset construction pipeline. Various methods have been proposed to tokenize Chinese, including Conditional Random Fields classifiers in the Stanford Chinese Word Segmenter (Tseng et al., 2005) and in the Chinese Mecab[7]. Independently of these methods, several guidelines have been proposed for human annotation of Chinese tokens, including the Chinese Penn Treebank and the Peking University standard (Duan et al., 2003). This results in tokens with shorter or larger spans depending on the guideline, for instance 有线 (*cable*) 电视 (*television*) according to Peking University vs. 有线电视 (*cable television*) according to Chinese Penn Treebank. Chinese tokenizers display the same variety in their choices of token span length; some, such as Stanford or jieba,[8] leave it to the user to choose which strategy to apply (full=short, default=large, search=multiple solutions). We attempted to avoid these considerations by working directly with characters. This was initially motivated by the technical choice of Solr (v6.4.0), whose only option for a Chinese tokenizer was bigrams. We kept sentences between 15 and 40 characters, which we estimated to yield sizes comparable to the English sentences (between 10 and 20 words). Query sentences were truncated to 15 characters (instead of 5 words). However, working with characters raised the following issues. First, character-based Solr search resulted in a lower sentence similarity than in other languages. It often occurred that given a (parallel) Chinese sentence as a query, the matching unigrams or bigrams of characters would be stop words or other common (bigrams of) characters. We compiled a set of 533 stop expressions including punctuation, short common words (一 one, 一切 every), and locutions such as 不仅 (not only), 一方面 (on the one hand), 另一方面 (on the other hand), 反过来说 (on the other hand), etc. Stop expressions were removed from parallel sentences before using them as queries.

Besides, sentences would sometimes start with a common locution followed by a comma: 从历史上看， (from a historical point of view), 由此可见， (from this, it can be seen that), etc. Seeing the large variety of such locutions, we decided to remove any leading sequence of up to six characters followed by a comma from the start of Chinese parallel sentences before using them as queries.

These heuristics were designed and tuned by human review of samples of resulting sequences of two sentences.

3. Use in Two Shared Tasks

These datasets were used in the BUCC 2017 and 2018 Shared Tasks (Zweigenbaum et al., 2017; Zweigenbaum et al., 2018). Participants were taskeed with detecting in a bilingual pair of corpora the inserted parallel sentences. Three of the four language pairs were addressed by the participants in 2017: French, German, and Chinese, with a maximum F-score of 0.84 on German-English (Azpeitia et al., 2017) (see Table 3). All four language pairs were addressed in 2018, with improved F-scores topping at 0.86 for German-English again.

Year	de-en	fr-en	ru-en	zh-en
2017	84	79	–	43
2018	86	81	81	75

Table 3: Best F-scores (%) at the BUCC Shared Tasks in 2017 and 2018

4. Discussion and Perspectives

The resulting corpora can be obtained from the BUCC Web site.[9] They total about 3.5 million sentences in five languages. Participants found methods to cope with this large number of sentences without metadata. To our knowledge, no participant tried to take advantage of a possible lack of cohesion or other features of the inserted sentences that would come from their artificial insertion into the monolingual corpora. The remaining question is that of the accuracy of the provided gold standard, which only accounts for artificially inserted parallel sentences: this accuracy may be reduced by the possible existence of naturally occurring parallel sentence pairs. We estimated their rate of occurrence to be at most 5%, based on a human assessment of the proportion of sentence pairs among the false positives of the most precise systems that are actually true parallel sentence pairs (Zweigenbaum et al., 2017).

5. Acknowledgements

PZ acknowledges funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 676207. We thank Zheng Zhang for his help in evaluating the cohesion of Chinese sentence pairs in earlier experiments. Issues in the cohesion of the final corpora are our own responsibility.

6. Bibliographical References

Abdul-Rauf, S. and Schwenk, H. (2009). Exploiting comparable corpora with TER and TERp. In *Proceedings of the 2nd Workshop on Building and Using Comparable Corpora: From Parallel to Non-parallel Corpora*, BUCC '09, pages 46–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

[6] Throughout this paper we use the term *Chinese* to refer to Modern Standard Chinese, often called Mandarin Chinese.

[7] https://github.com/panyang/MeCab-Chinese

[8] https://github.com/fxsjy/jieba

[9] https://comparable.limsi.fr/bucc2017/bucc2017-task.html

Agirre, E., Banea, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Mihalcea, R., Rigau, G., and Wiebe, J. (2016). SemEval-2016 Task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, San Diego, California, June. Association for Computational Linguistics.

Azpeitia, A., Etchegoyhen, T., and Martínez Garcia, E. (2017). Weighted set-theoretic alignment of comparable sentences. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 41–45, Vancouver, Canada, August. Association for Computational Linguistics.

Buck, C. and Koehn, P. (2016). Findings of the WMT 2016 bilingual document alignment shared task. In *Proceedings of the First Conference on Machine Translation*, pages 554–563, Berlin, Germany, August. Association for Computational Linguistics.

Duan, H., Bai, X., Chang, B., and Yu, S. (2003). Chinese word segmentation at Peking University. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 152–155, Sapporo, Japan, July. Association for Computational Linguistics.

Munteanu, D. S., Fraser, A., and Marcu, D. (2004). Improved machine translation performance via parallel sentence extraction from comparable corpora. In Daniel Marcu Susan Dumais et al., editors, *HLT-NAACL 2004: Main Proceedings*, pages 265–272, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Potthast, M., Gollub, T., Hagen, M., Kiesel, J., Michel, M., Oberländer, A., Tippmann, M., Barrón-Cedeño, A., Gupta, P., Rosso, P., and Stein, B. (2012). Overview of the 4th international competition on plagiarism detection. In Pamela Forner, et al., editors, *CLEF 2012 Evaluation Labs and Workshop, Online Working Notes, Rome, Italy, September 17-20, 2012*, volume 1178 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Sharoff, S., Zweigenbaum, P., and Rapp, R. (2015). BUCC shared task: Cross-language document similarity. In *Proc Eighth Workshop on Building and Using Comparable Corpora*, pages 74–78, Beijing, China, July. Association for Computational Linguistics.

Sharoff, S. (2013). Measuring the distance between comparable corpora between languages. In Serge Sharoff, et al., editors, *BUCC: Building and Using Comparable Corpora*, pages 113–129. Springer.

Sharoff, S. (2018). Functional text dimensions for annotation of web corpora. *Corpora*, 13(1).

Smith, J. R., Quirk, C., and Toutanova, K. (2010). Extracting parallel sentences from comparable corpora using document level alignment. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 403–411. Association for Computational Linguistics.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D., and Manning, C. (2005). A Conditional Random Field word segmenter for SIGHAN Bakeoff 2005. In *SIGHAN Workshop on Chinese Language Processing*. Association for Computational Linguistics, January.

Utiyama, M. and Isahara, H. (2003). Reliable measures for aligning Japanese-English news articles and sentences. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 72–79, Sapporo, Japan, July. Association for Computational Linguistics.

Zweigenbaum, P., Sharoff, S., and Rapp, R. (2017). Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada, August. Association for Computational Linguistics.

Zweigenbaum, P., Sharoff, S., and Rapp, R. (2018). Overview of the 2018 BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 11th Workshop on Building and Using Comparable Corpora*, Miyazaki, Japan, May. ELRA.

Manual vs Automatic Bitext Extraction

Aibek Makazhanov, Bagdat Myrzakhmetov, Zhenisbek Assylbekov
Nazarbayev University, National Laboratory Astana
Nazarbayev University, School of Science and Technology
53 Kabanbay Batyr ave., Astana, Kazakhstan
aibek.makazhanov@nu.edu.kz, bagdat.myrzakhmetov@nu.edu.kz, zhassylbekov@nu.edu.kz

Abstract

We compare manual and automatic approaches to the problem of extracting bitexts from the Web in the framework of a case study on building a Russian-Kazakh parallel corpus. Our findings suggest that targeted, site-specific crawling results in cleaner bitexts with a higher ratio of parallel sentences. We also find that general crawlers combined with boilerplate removal tools tend to retrieve shorter texts, as some content gets cleaned out with the markup. When it comes to sentence splitting and alignment we show that investing some effort in data pre- and post-processing as well as fiddling with off-the-shelf solutions pays a noticeable dividend. Overall we observe that, depending on the source, automatic bitext extraction methods may lack severely in coverage (retrieve fewer sentence pairs) and on average are fewer precise (retrieve less *parallel* sentence pairs). We conclude that if one aims at extracting high-quality bitexts for a small number of language pairs, automatic methods best be avoided, or at least used with caution.

Keywords: bitext extraction, crawling, sentence alignment

1. Introduction

The Web has long been used as one of, if not, the most important source for obtaining language resources of various purpose, size and quality. Given the large amount and accessibility of text available in a variety of languages, it is particularly appealing to mine the Web for parallel and comparable corpora (bitexts), resources essential for building and evaluating data-driven machine translation systems.

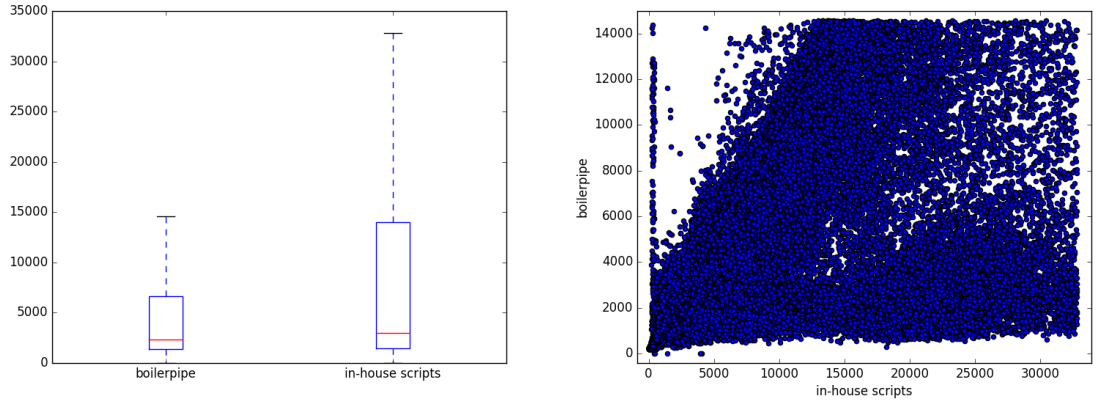
The task of extracting bitexts from the Web typically involves the following four major consecutive steps: (i) data collection (crawling), (ii) document alignment, (iii) sentence segmentation, and (iv) sentence alignment. Depending on the degree of user involvement in the first two steps, approaches to the problem can be classified roughly as: (i) manual methods (Koehn, 2005; Tiedemann, 2007; Myrzakhmetov et al., 2016) that crawl specific websites and usually rely on their idiosyncratic properties at the document alignment step; (ii) semi-automatic methods (Esplà-Gomis and Forcada, 2010; Papavassiliou et al., 2013) that make no site-specific assumptions, but still require a target list of URLs to extract bitexts from; (iii) completely automatic methods (Smith et al., 2013; Ljubešić et al., 2016; Resnik and Smith, 2003) that crawl the entire Web and produce whatever bitexts they can.

Choosing appropriate extraction strategy depends greatly on the task at hand. Thus, if the goal is to extract bitexts for dozens of language pairs automatic approaches may come in handy. However, if one needs a high quality parallel corpus for just one or two pairs of languages, what is the difference between bitexts obtained by manual and automatic methods? Is the final quality of manual extraction worth time and effort? To our disappointment, we could not find detailed answers to these questions in the literature. Various implementations of the extraction strategies are typically evaluated in terms of the alignment quality of the obtained bitexts and/or performance of SMT systems trained on those bitexts. Comparative evaluation mostly concerns different implementations of the semi-automatic approaches (Toral et al., 2014; Esplà-Gomis et al., 2014; Laranjeira et al., 2014). We believe that even a rough estimate of the quality/quantity trade off between bitexts obtained manually and automatically can provide important insights and guidelines for building parallel corpora. In this work we attempt to provide such an estimate.

We divide the process of bitext extraction (BE) in two stages as a pre- and post-document alignment (DA) stage. The pre-DA stage involves crawling and DA itself. The post-DA stage is about sentence splitting and alignment, with possible cleaning steps that may improve the quality of an extracted bitext. We compare (semi-) automatic and manual approaches to both BE stages using a range of metrics and manual and automatic estimates.

1.1. General Setting

Before proceeding let us describe a particular setting considered herein. First, the present work relies heavily on particular sources, namely certain websites that adhere to the same language policy of making multilingual releases in Kazakh and Russian languages. Such websites almost always provide page-level alignment, e.g. a Russian version of a page contains a direct link to a Kazakh version of the same page and vice versa. Thus, we handle document alignment at the stage of crawling. Second, for BE we use only freely accessible tools; hence our conclusions may not generalize to commercial analogs, if any. Lastly, to conduct our experiments we use the following four websites as the development set: `adilet.zan.kz`, `akorda.kz`, `astana.gov.kz`, and `strategy2050.kz`, which are shortened to `adilet`, `akorda`, `astana`, and `strategy`, and hereinafter collectively referred to as 'four (web)sites'.



(a) Side-by-side boxplots

(b) Scatter plot

Figure 1: Lengths (in characters) of the texts extracted by the boilerpipe and in-house scripts

2. General vs Targeted Crawling

The crawling step can be divided into three parts: (1) obtaining the links to parallel web pages (spidering), (2) downloading raw HTML files, (3) extracting plain texts from the downloaded web pages. The downloading step is trivial; it is steps (1) and (3) where general and targeted crawling diverge.

General crawling. For a given page-aligned website we obtain the links to all pages in one language (*source* language), using the `GNU wget` tool in spider mode and specifying a regular expression pattern to accept only the links to pages in *source* language (e.g. `'/ru/'` for the links in Russian). Then we use `GNU wget` again to download all the pages in the source language. Next from raw HTML source pages we obtain the corresponding links to pages in *target* language from the raw source language HTML files and download all the web pages in the target language. Finally, we remove boiler-plate text, i.e. navigational elements, templates, and advertisements which are not related to the main content, and extract plain text from the downloaded HTML-files using the `boilerpipe` tool. We also remove duplicates using `fdupes` tool[1].

Targeted crawling. For this task we use site-specific crawlers, which are essentially in-house python scripts (one per website). Here instead of obtaining the links and later using them to download the content. We can traverse certain branches of a website (say, news), download pages in both source and target languages, and extract meaningful content relying on the HTML-structure of the site: all in one go.

2.1. Comparison of the Spider Methods

For a given website, let B be the set of unique source-language links obtained by the baseline method, and I be the set of unique source-language links obtained by the in-house scripts. Cardinalities of B, I, $B \cap I$, $B \setminus I$, $I \setminus B$ for the four websites are given in Table 1. One should notice that the numbers in Table 1 are not directly comparable between the two methods: in the baseline approach we use the `wget` tool to obtain *all* the URLs from a given website, whereas the hand-made scripts are designed to crawl only particular branches (legal documents, news, announcements, etc.) on each website. Theoretically, `wget` in `--spider` mode should obtain *all* links from a given website, and one should expect that $B \supset I$ (or almost so depending on a time lag between two crawls), but this is not always the case as it can be seen from Table 1. Moreover, in some cases the in house scripts obtained more links, and as our closer analysis reveals in the case of `akorda` there were 3 unique links per page in B.

2.2. Comparison of the Text Extraction Methods

We consider the links from $B \cap I$, i.e. those links which were obtained by both the baseline and the in-house methods. According to Table 1, there are 108,960 of such links per each language. We downloaded the web pages from those links, and then we applied the `boilerpipe` tool and the in-house Python scripts to extract texts from them. After that character lengths were calculated for all texts, and outliers were removed using the Tukey's IQR rule (Tukey, 1977).

Side-by-side boxplots (Fig. 1a) show that the automatic extractors seem to produce longer texts than the automatic and site-agnostic `boilerpipe`. The scatter plot (Fig. 1b) provides more detail clearly showing that the majority of longer texts were extracted by the in-house scripts. Average length of the extracted texts is 3,226.56 for `boilerpipe` and 5,290.62 for the in-house scripts. According to the Shapiro-Wilk test (Shapiro and Wilk, 1965) normality is violated for text lengths in both cases, $p < 0.001$. Wilcoxon signed-rank test (Wilcoxon, 1945), confirms the hypothesis that `boilerpipe` produces shorter texts on average, $p < 0.001$. Thus, boilerpipe at best provides comparable quality of text extraction, and in worst case it throws away useful chunks of text (main content).

[1] https://github.com/adrianlopezroche/fdupes

Website	$\|B\|$	$\|I\|$	$\|B \cap I\|$	$\|B \setminus I\|$	$\|I \setminus B\|$
adilet	86,234	87,363	79,088	7,146	8,275
akorda	33,572	6,357	6,347	27,225	10
astana	28,067	6,512	6,496	21,571	16
strategy	28,078	28,253	17,029	9,246	11,224
TOTAL	175,951	128,485	108,960	65188	19525

Table 1: Number of links obtained by the baseline (B) and by the in-house (I) methods

Website	Annotator-1		Annotator-2		Annotator-3		Final	
	Wget + boilerp.	In-house scripts	Wget + boilerp.	In-house scripts	Wget + boilerp.	In-house scripts	Wget + boilerp.	In-house scripts
adilet	0.7950	0.9375	0.8050	0.9425	0.7125	0.8825	0.7500	0.9075
akorda	0.8200	0.9525	0.8125	0.9450	0.7625	0.8675	0.7750	0.9050
astana	0.6400	0.7925	0.6350	0.7950	0.5050	0.7325	0.5975	0.7400
strategy	0.7550	0.7700	0.7200	0.7525	0.6250	0.6575	0.6425	0.6900

Table 2: Crawling vs Alignment: manual evaluation, proportions of parallel pairs are provided

Metrics	Wget+boilerpipe				In-house crawling scripts			
	adilet	akorda	astana	strategy	adilet	akorda	astana	strategy
# parallel	6,522,758	94,855	208,515	250,536	22,728,878	70,949	63,729	201,671
parallel/total	0.7680	0.8309	0.5698	0.6680	0.8803	0.9116	0.7215	0.6650
short/parallel	0.3079	0.1211	0.0676	0.2276	0.6312	0.0190	0.0098	0.0202
junk/total	0.1260	0.0998	0.3897	0.1191	0.0743	0.0294	0.0610	0.0982

Table 3: Crawling vs Alignment: automatic evaluation

2.3. Impact of Crawling

To see how the two different ways of crawling affect the final quality of the bitext we applied both methods to the four websites and obtained two sets of parallel documents, which we proceed to split into sentences using NLTK Punkt tokenizer(Kiss and Strunk, 2006), and align with Hunalign(Varga et al., 2007). Thus we end up with two sets of bitexts which differ only in the way they were crawled. To compare these set we randomly sampled 400 pairs from each population[2] and asked three annotators if each pair of sentences intended to provide the same content in the two different languages[3]. The fourth annotator made the final decision. One can see from Table 2 that despite the differences in annotations, crawling with the in-house Python scripts consistently outperforms the `Wget+boilerpipe`-based crawling across the annotations. The difference in sample proportions for the 'Final' column is significant at 0.001 level for `adilet`, `akorda`, `astana`, and at 0.1 level for `strategy`, two-sided z-test for proportions. This supports the hypothesis that the in-house scripts result in *cleaner* bitexts than the `wget+boilerpipe`-based crawling.

We can also compare the levels of noise in the bitexts by considering the following metrics: (i) proportion of short pairs among parallel pairs; (ii) proportion of obviously nonparallel pairs (junk) among all pairs. Here a pair considered short whenever the lengths of each sentence in a pair does not exceed *three* words. Junk pairs are defined as follows: (i) at least one of the sides (source or target) is empty; (ii) at least one of the sides does not contain any letters (Latin and Cyrillic); (iii) both sides are identical after tokenization and lowercasing. The results of this comparison across the four websites are given in Table 3. We can see again that the targeted crawling produces cleaner bitexts than that of general.

2.4. Automatic Evaluation

Although manual evaluation is the best way to estimate alignment quality, it requires significant time and effort. Therefore we follow Assylbekov et al. (2016) and use the set of length- and context-based features to automatically estimate alignment quality for a given pair of sentences. We train a number of classifiers on the data obtained from the previous experiment, i.e. manually annotated 3200 sentence pairs (8 samples 400 sentence pairs each). From a number of tested classifiers we chose the gradient boosting classifier since it achieved the highest accuracy and low variance.

The results of the automatic evaluation on the entire data set are given in Table 3. One can see that ratio of parallel pairs are within 5% margin of error from

[2]According to Cohran's sample size formula (Cochran, 2007), this is enough to ensure 5% margin of error at 95% confidence level.

[3]The rationale behind asking this question instead of the question "Are these sentences translations of each other?" is provided by (Resnik and Smith, 2003).

Metrics	Manual BE pipeline				`Bitextor`			
	adilet	akorda	astana	strategy	adilet	akorda	astana	strategy
# of parallel	22,658,881	70,025	70,449	202,633	296,650	42,217	20,129	67,643
parallel/total	0.9075	0.9484	0.7992	0.7303	0.8725	0.7900	0.3000	0.4350
short/parallel	0.3412	0.0095	0.0075	0.0195	0.0000	0.0285	0.0250	0.0057
junk/total	0.0000	0.0000	0.0000	0.0000	0.0000	0.0025	0.0000	0.0000

Table 4: End-to-end bitext extraction: manual vs automatic

Method	P	P/T	S/P	J/T
Automatic SSA	5,766,307	0.7946	0.1701	0.0657
Domain-adapted SS	5,738,912	0.8068	0,1765	0,0650
SA + AD	5,748,352	0.8052	0,1771	0,0649
SA + L + MD	5,750,528	0.8436	0,1742	0,0264
SA + L + MD + JR	5,750,497	0.8446	0,0944	0,0000

Table 5: Evaluation of consecutive modification of SSA tools: P - # of sentence pairs deemed parallel; T - total pairs; S - short pairs; J - junk pairs; SS - sentence splitting; SA - sentence alignment; AD - automatic dictionary; L - lemmatization; MD - manual dictionary; JR - junk removal.

the estimates in the 'Final' column of Table 2. Therefore we have a quick and reliable method to get the approximate proportion (and number) of parallel sentence pairs in a given corpus, which can be used to evaluate any further modifications to the BE pipeline.

3. Bitext Extraction Beyond Crawling

Once crawling and document alignment is complete, we can proceed to the next stage of BE, which includes sentence splitting and alignment (SSA). Here we also distinguish between automatic and manual approaches. An automatic approach entails direct application of the SSA tools, whereas a manual approach assumes pre- and post-processing of data, as well as adapting SSA tools for a particular language pair.

Starting with the parallel documents that we have extracted at the crawling stage, we perform SSA: first automatically and then with successive application of the following modifications: (i) domain adaptation of a sentence splitter (retraining Punkt tokenizer); (ii) using automatically extracted dictionary for sentence alignment (`-realign` option of Hunalign); (iii) using hand-crafted dictionary in conjunction with preliminary lemmatization of texts[4]; (iv) removal of junk pairs (cf. subsection 2.3.).

After consecutive application of all the modifications we have compared the resulting bitext with the one that was extracted by a popular off-the-shelf tool `Bitextor` (Esplà-Gomis and Forcada, 2010). Bitextor was run on the four sites on default settings. As it can be seen from Table 4 the manual pipeline extracts more parallel sentences (especially for `adilet` data) and is more accurate than an unsupervised automatic tool. However, on average, Bitextor produces cleaner bitexts with lower short/parallel ratios and surprisingly low amount of junk.

Lastly we would like to assess the dynamics of modifying SSA tools. To this end, we perform automatic evaluation of the bitexts extracted after successive application of each modification. The results of this assessment (averaged over the four sites) are given in Table5. As it can be seen consecutive application of the modifications "cleanses" resulting bitexts, as the ratio of junk pairs decreases. At the same time there is a steady increase in the ratio of parallel sentences. However, the number of parallel and short/parallel pairs do not change monotonically. When we apply domain adapted SS the amount of parallel pairs drops almost 30K, but after that enjoys a steady growth. It turns out that as we provided Punkt sentence splitter with the list of abbreviations a total number of sentences dropped as the number of incorrect segmentations (due to dotted abbreviations) has been reduced.

4. Related Work

Koehn (2005) employs manual approach by mining the Euro Parliament Proceedings. The documents are aligned by comparing time stamps and the HTML structure. Similarly Tiedemann (2007) focuses on the single website to obtain movie subtitles and align those using time stamps. Smith et al. (2013) use the CommonCrawl Web snapshot. Candidate documents are retrieved by matching language names and codes to URLs. The candidates are later aligned using an extension of the STRAND algorithm (Resnik and Smith, 2003). Ljubešić et al. (2016) employ a combination of the SpiderLing crawler and the Bitextor. The former crawls the top level domains and saves the links to the websites whose pages are identified as written in predefined target languages. This list of links is later provided to the Bitextor, which aligns documents using their various properties, such as size, file name, difference in length etc.

[4]For lemmatization we use: Mystem (Segalovich, 2003) for Russian, and a data-driven morphological disambiguator (Makhambetov et al., 2015) for Kazakh.

5. Conclusion

We have compared manual and automatic approaches to the problem of extracting bitexts from the Web in the framework of a case study on building a Russian-Kazakh parallel corpus. We conclude that if one aims at extracting high quality bitexts for a small number of language pairs, automatic methods best be avoided, or at least used with caution. In the future we plan to expand the work on manual bitext extraction, and experiment with more Kazakhstani websites, including *not* page-aligned ones. Our ultimate goal is to build a high quality, decent-sized parallel corpus for the Russian-Kazakh pair.

6. Acknowledgements

This work has been funded by the Nazarbayev University under the research grant №129-2017/022-2017 and by the Committee of Science of the Ministry of Education and Science of the Republic of Kazakhstan under the research grant AP05134272.

7. Bibliographical References

Assylbekov, Z., Myrzakhmetov, B., and Makazhanov, A. (2016). Experiments with Russian to Kazakh Sentence Alignment. *Izvestija KGTU im.I.Razzakova*, 38(2):18–23.

Cochran, W. G. (2007). *Sampling techniques*. John Wiley & Sons.

Esplà-Gomis, M. and Forcada, M. (2010). Combining content-based and url-based heuristics to harvest aligned bitexts from multilingual sites with bitextor. *The Prague Bulletin of Mathematical Linguistics*, 93:77–86.

Esplà-Gomis, M., Klubička, F., Ljubešić, N., Ortiz-Rojas, S., Papavassiliou, V., and Prokopidis, P. (2014). Comparing two acquisition systems for automatically building an english-croatian parallel corpus from multilingual websites. In *LREC*, pages 1252–1258.

Kiss, T. and Strunk, J. (2006). Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Laranjeira, B., Moreira, V. P., Villavicencio, A., Ramisch, C., and Finatto, M. J. B. (2014). Comparing the quality of focused crawlers and of the translation resources obtained from them. In *LREC*, pages 3572–3578. Citeseer.

Ljubešić, N., Esplà-Gomis, M., Toral, A., Rojas, S. O., and Klubička, F. (2016). Producing monolingual and parallel web corpora at the same time - spiderling and bitextor's love affair. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Makhambetov, O., Makazhanov, A., Sabyrgaliyev, I., and Yessenbayev, Z. (2015). Data-driven morphological analysis and disambiguation for kazakh. In *Proceedings of the 2015 Computational Linguistics and Intelligent Text Processing, Part I*, pages 151–163, Cairo, Egypt. Springer International Publishing.

Myrzakhmetov, B., Sultangazina, A., and Makazhanov, A. (2016). Identification of the parallel documents from multilingual news websites. In *Application of Information and Communication Technologies (AICT), 2016 IEEE 10th International Conference on*, pages 197–202. IEEE.

Papavassiliou, V., Prokopidis, P., and Thurmair, G. (2013). A modular open-source focused crawler for mining monolingual and bilingual corpora from the web. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 43–51.

Resnik, P. and Smith, N. A. (2003). The web as a parallel corpus. *Computational Linguistics*, 29(3):349–380.

Segalovich, I. (2003). A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In *MLMTA*.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611.

Smith, J. R., Saint-Amand, H., Plamada, M., Koehn, P., Callison-Burch, C., and Lopez, A. (2013). Dirt cheap web-scale parallel text from the common crawl. In *ACL (1)*, pages 1374–1383.

Tiedemann, J. (2007). Building a multilingual parallel subtitle corpus. *Proc. CLIN*, page 14.

Toral, A., Rubino, R., Esplà-Gomis, M., Pirinen, T., Way, A., and Ramirez-Sanchez, G. (2014). Extrinsic evaluation of web-crawlers in machine translation: a case study on croatian–english for the tourism domain. In *Proceedings of the 17th Conference of the European Association for Machine Translation (EAMT)*, pages 221–224.

Tukey, J. W. (1977). Exploratory data analysis.

Varga, D., Halacsy, P., Kornai, A., Nagy, V., Nemeth, L., and Tron, V. (2007). Parallel corpora for medium density languages. *Recent Advances in Natural Language Processing IV: Selected Papers from RANLP 2005*, 292:247.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83.

A Morphologically Annotated Corpus of Emirati Arabic

Salam Khalifa, Nizar Habash, Fadhl Eryani,
Ossama Obeid, Dana Abdulrahim,[†] Meera Al Kaabi[‡]
Computational Approaches to Modeling Language Lab, New York University Abu Dhabi, UAE
University of Bahrain, Bahrain[†]
The United Arab Emirates University, UAE[‡]
{salamkhalifa@nyu.edu, nizar.habash@nyu.edu, darahim@uob.edu.bh}

Abstract

We present an ongoing effort on the first large-scale morphologically manually annotated corpus of Emirati Arabic. This corpus includes about 200,000 words selected from eight Gumar corpus novels in the Emirati Arabic variety. The selected texts are being annotated for tokenization, part-of-speech, lemmatization, English glosses and dialect identification. The orthography of the text is also adjusted for errors and inconsistencies. We discuss the guidelines for each part of the annotation components, and the annotation interface we use. We report on the quality of the annotation through an inter-annotator agreement measure.

Keywords: Gulf Arabic, Part-of-Speech Tagging, Morphology, Annotation

1. Introduction

There has been an increasing number of natural language processing (NLP) efforts focusing on dialectal Arabic, especially with the increasing amounts of written material on the web. However, resources for dialectal Arabic NLP tasks such as part-of-speech (POS) tagging, morphological analysis and disambiguation are still lacking compared to those for Modern Standard Arabic (MSA). MSA is the official language in more than 20 countries, where it is used in official communications, news, and education. Yet, it is not the commonly spoken variety of Arabic; the dialectal varieties of Arabic are what is used in the day-to-day communication. Dialectal Arabic is also commonly used in written form on social media platforms, forums and blogs.

Using available resources developed for MSA such as POS taggers and tokenizers gives limited performance when used on dialectal Arabic (Habash and Rambow, 2006; Jarrar et al., 2014; Khalifa et al., 2016a). Many researchers moved into the direction of creating tools and resources targeting the dialects specifically. Egyptian Arabic is one of the dialects that received earlier efforts for developing tools and resources. More resources are being developed for other dialects such as Levantine, Tunisian, Moroccan and Yemeni Arabic. Gulf Arabic, as we define it to be the native spoken variety in the Gulf Cooperation Council, is still lagging behind other Arabic dialects with respect to resource and tool creation, given the considerable amount of dialectal content online.

In this paper, we present an ongoing project for creating a manually annotated corpus of about 200,000 words of the Gulf Arabic of the United Arab Emirates – Emirati Arabic. The corpus is annotated for tokenization, POS, lemmas and English glosses in addition to spelling conventionalization and dialect identification. This resource will support the development of Arabic dialect enabling technologies, such as automatic POS tagging and morphological disambiguation, which in turn will facilitate efforts on different NLP tasks such as machine translation.

The rest of this paper is organized as follows. We discuss related work on dialectal corpora in Section 2. In Section 3. we describe the corpus used in this effort. We then present the annotation guidelines that are used to annotate the corpus in Section 4. We discuss the annotation process and the annotation quality results in Section 5.

2. Related Work

In this section we review a number of efforts on Arabic corpus creation, that significantly supported research and tool development for Arabic NLP.

2.1. Modern Standard Arabic Resources

The Penn Arabic Treebank (PATB) (Maamouri et al., 2004) has been a central resource for developing MSA resources. It was developed at the Linguistic Data Consortium (LDC), and it mainly consists of newswire text from different news sources. The PATB corpus is annotated for tokenization, segmentation, POS tagging, lemmatization, diacritization, English gloss and syntactic structure. The PATB has 12 parts of more than 1.3 million words. The annotated data has been a backbone of many state-of-the-art tools such as analyzers and disambiguators including MADAMIRA (Pasha et al., 2014) and its predecessor MADA (Habash et al., 2009), in addition to YAMAMA (Khalifa et al., 2016b), and most recently a neural morphological disambiguatior (Zalmout and Habash, 2017) and a fine grained POS tagger (Inoue et al., 2017). In addition, the PATB guidelines (Maamouri et al., 2009) have inspired the creation of similar guidelines for the dialects including our own.

2.2. Dialectal Arabic Resources

In the scope of dialectal Arabic, there have been many recent contributions to the development and creation of resources. Below, we discuss the highlights of those contributions.

Egyptian Arabic Resources Egyptian Arabic (EGY) was one of the first dialects that received the attention of the NLP community. The earliest effort, to the best of

our knowledge, is the Egyptian Colloquial Arabic Lexicon (ECAL) (Kilany et al., 2002) which was developed as part of the CALLHOME Egypt corpus (Gadalla et al., 1997). The ECAL served as the seed to the EGY morphological analyzer (CALIMA) (Habash et al., 2012a). Later on, the Egyptian Arabic Treebank (ARZATB) (Maamouri et al., 2012a; Maamouri et al., 2014) was created by the LDC using CALIMA to provide analysis options for the annotation process. The ARZATB has currently 400,000 words in eight parts annotated in a similar fashion to the PATB. The annotation guidelines for the ARZATB (Maamouri et al., 2012b) followed that of the PATB with decisions specific to the dialect. Since the release, the ARZATB has been used extensively for developing EGY resources such as the EGY part of MADAMIRA, MADA and YAMAMA, in addition to a noise-robust morphological disambiguator for EGY (Zalmout et al., 2018). Other developed corpora and POS taggers for EGY include the work of Al-Sabbagh and Girju (2012) where they created their own POS tagset and corpus with the intention to facilitate certain NLP applications like subjectivity and sentiment analysis.

Levantine Arabic and and Other Dialectal Arabic Resources Levantine Arabic (LEV) received some notable efforts including the Levantine Arabic Treebank (LATB) of Jordanian Arabic (Maamouri et al., 2006) which contains around 27,000 annotated words in a similar fashion to ARZATB. A more recent resource is the annotated corpus of Palestinian Arabic (Curras) (Jarrar et al., 2014; Jarrar et al., 2016). ARZATB and Curras were used to create morphological analyzers and disambiguators (Eskander et al., 2016). Other dialects such as Yemeni and Moroccan Arabic followed the same approach (Al-Shargi et al., 2016). In addition to the dialects mentioned above, there were recent efforts on creating corpora for other dialects, namely Tunisian and Algerian (McNeil and Faiza, 2011; Masmoudi et al., 2014; Zribi et al., 2015; Smaïli et al., 2014). Other works targeted multi-dialect corpora (Diab et al., 2010; Zaidan and Callison-Burch, 2011; Diab et al., Forthcoming 2013; Bouamor et al., 2014; Cotterell and Callison-Burch, 2014), and, most recently, the ongoing Multi Arabic Dialect and Application Resources project (MADAR) (Bouamor et al., 2018) which includes corpora for 25 different city dialects.

Gulf Arabic Resources As far as Gulf Arabic (GLF) is concerned, the only existing annotated corpora include the Emirati Arabic Corpus (EAC) (Halefom et al., 2013) and the Emirati Arabic Language Acquisition Corpus (EMALAC) (Ntelitheos and Idrissi, 2017) that were created by linguists with emphasis on the phonological and morphosyntactic phenomena of Emirati Arabic. We recently collected a large-scale corpus of Gulf Arabic (Khalifa et al., 2016a) containing more than 100 million words covering six Gulf Arabic varieties. In regards to other tools and resources, we recently developed a morphological analyzer for Gulf Arabic verbs ($\text{CALIMA}_{\text{GLF}}$) (Khalifa et al., 2017). We are also aware of the previously developed rule-based stemmer for Arabic Gulf dialect (Abuata and Al-Omari, 2015).

In this work, we use about 200,000 words from the Emirati Arabic portion of the Gumar corpus to manually annotate for tokenization, POS tagging, lemma, English gloss and dialect identification. Additionally we conventionalize the spelling in accordance with the Conventional Orthography for Dialectal Arabic (CODA) rules (Habash et al., 2012b; Habash et al., 2018).

For recent surveys on Arabic resources for NLP, see Zaghouani (2014), Shoufan and Al-Ameri (2015) and Zeroual and Lakhouaja (2018).

3. Annotating the Gumar Corpus

We discuss next the Gumar Corpus and the portion of it we use to annotate in this effort.

3.1. Gumar Corpus

The Gumar corpus is a large-scale corpus of Gulf Arabic containing more than 100 million words. The corpus consists mainly of documents of long conversational novels also known as روايات النت 'Internet Novels'. This type of literature is very popular among female teenagers in the Gulf area. These novels are written mostly in dialectal Arabic, where the lengthy conversations between the characters of the story are in the dialect and the narration in between the conversations can sometimes be in MSA.

The writers of the novels remain anonymous and use *noms de plume*. The novels are publicly available online, where most of the writers ask for their pen name to be mentioned if the novel is to be published in a different platform than the original. The genre of the novels is mainly romantic, but also features tragedy and drama. The corpus can be browsed online,[1] it is currently annotated using MADAMIRA in EGY mode.

On the document level, Gulf Arabic text makes up more than 90% of the corpus, the rest of the corpus consists of other Arabic dialects in addition to MSA. Emirati Arabic text covers around 11% of the Gumar corpus.

3.2. The Annotated Gumar Corpus

We chose a set of 200,110 word tokens for the annotation task. The text consists of the first 25,000 words (rounded up to the nearest full sentence) from eight different novels by eight different authors. This allows us to cover different writing styles. The text is comprised of 15,277 sentences with an average of 13 words per sentence. Table 1 shows the list of the novels from which the text is selected. We name this subset of the corpus the *Annotated Gumar Corpus*. In the future we plan to continue adding annotations to it from other Gumar novels including different dialects.

Additionally, a total of about 12,000 words – 1,500 words from each of the eight parts rounded up to the nearest full sentence – are chosen to evaluate Inter-Annotator Agreement (IAA) throughout the annotation process. Thus, the total number of words to be annotated is about 212,000 words.

[1] Please visit `https://camel.abudhabi.nyu.edu/gumar/`

Parts	Tokens	Sentences	العنوان	Title Transltation	المؤلف	Author
Part 1	25,022	1,387	صوت دقات الخوافق لا طريته مثل حبات المطر فوق الخيام	'The sound of the beating hearts when I remember it, is like the drops of rain on the tents'	b3thra had2a	'Calm scatter'
Part 2	25,009	3,176	والله أحبك لو على رقبتي سيف	'I swear to God I love you even if there is a sword on my neck'	أم نشوان	'Umm Nashwan'
Part 3	25,004	1,732	بنيت لك في داخل فؤادي من اللّهفة قصر	'I built you a castle in my heart out of longing'	أسماء	'Asmaa'
Part 4	25,002	1,919	مجنون ساره	'Crazy about Sarah'	فاتنة دبي	'The enchantress of Dubai'
Part 5	25,004	1,412	وش عذرك يوم تخون وش ذنبي يوم صدقتك	'What is your excuse when you betray what was my fault when I believed you'	غايتي رضا ربي	'My Lord's satisfaction is my purpose'
Part 6	25,039	1,439	ملامح وجهي القديم	'Features of my old face'	حديقة الظلام	'The garden of darkness'
Part 7	25,002	2,211	له منزلن بالحيل مرموق ، وسط الحشا بأقصى الضمايا	'Your love has become part of me'	ضحية حبيبي	'The victim of my lover'
Part 8	25,028	2,001	طحت طيحة في هواكم	'I fell hard in your love'	السولعي	'The Gazelle'
Total	**200,110**	**15,277**				

Table 1: The list of novels (parts) used for annotation and their raw word counts. The English titles and author names are approximate translations of the original Arabic ones. The author names are *noms de plume*.

4. Annotation Guidelines

In this section, we present the guidelines with examples for each of the different annotation tasks. The annotation contains six different tasks: spelling conventionalization according to CODA, tokenization, POS tagging, lemmatization, English glossing and dialect Identification on the word level.

4.1. CODA Spelling Guidelines

Emirati Arabic is similar to other dialects where there is no standard orthography. For example the word for 'hunger' may be spelled phonetically اليوع *Alywς*[2] or using the MSA cognate الجوع *Aljwς*. Hence, there will always be inconsistencies between different writers or even within the same writer (Habash et al., 2012b). In this annotation effort we follow the newly revised set of CODA* guidelines which include consonant mapping, vowel spelling and affixation and cliticization rules (Habash et al., 2018).

4.2. Tokenization Guidelines

Previous efforts used different tokenization and segmentation schemes depending on the goal of the task. In this annotation task we use the D3 tokenization (Habash, 2010), where we keep the baseword and separate all the clitics including the ال *Al* 'the' definite article. The clitics include all attachable prepositions, particles and pronouns. For example, the word بالجوع *bAljwς* 'with the hunger' is tokenized as [ب+ال+جوع] *b+Al+*[*jwς*], where the baseword in this case is [جوع] [*jwς*].

[2]Arabic transliteration is presented in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007): (in alphabetical order)
ي و ه ن م ل ك ق ف غ ع ظ ط ض ص ش س ز ر ذ د خ ح ج ث ت ب أ
Â b t θ j H x dð r z s š S D T Ď ς γ f q k l m n h w y
and the additional symbols: ' ء, Â أ, Ă إ, Ā آ, ŵ ؤ, ŷ ئ, ى ,ħ ة, ý ى.

4.3. POS Guidelines

In this work, we opted to use a new POS tagset – CAMEL POS. CAMEL POS is inspired by the ARZATB tagset and guidelines (Maamouri et al., 2012b) which is based on the PATB guidelines (Maamouri et al., 2009). The CAMEL POS is designed as single tagset for both MSA and the dialects with the following goals in mind: (a) facilitating research on adaptation between MSA and the dialects, and among the dialects; (b) supporting backward compatibility with previously annotated resources; and (c) enforcing a functional morphology analysis that is deeper and more compatible with Arabic morphosyntactic rules than form-based analysis (Alkuhlani and Habash, 2011). The CAMEL POS tags and features are the union of those in MSA and the dialects. Features are available to use when needed. For example case and state features are used more often in MSA; but on the other hand, dialects tend to have many more clitics than MSA, including non-MSA ones.

One of the main differences between CAMEL POS and ARZATB is that the morphological features of both gender and number of nominals are annotated functionally (Alkuhlani and Habash, 2011; Smrž, 2007). This decision allows us to assign the features to the baseword without the need to specify the surface form affixes that mark form gender and number. This is not the case in ARZATB, where broken plural nouns are tagged singular because they do not use the sound plural affixes.

The other main difference is that we omit case and state features for nominals, and voice and mood for verbs as the dialects have almost lost them completely, except for some high frequency fossilized MSA forms, such as طَبعَاً *TabςAã* 'of course' which retains an indefinite ending.

The main part of the word, that is the baseword, is tagged in the following format: 'POS.features', where 'POS' is the core POS tag and 'features' is the possible feature combination that goes with the POS tag, a '.' separates the POS from the feature combination. Proclitics, however, get only a 'POS' tag since they have no features. However, pronom-

inal enclitics get a similar tag format as the baseword (i.e. 'PRON.features').

CAMEL POS provides full array of features: (i) **A**spect with the values **P**erfective, **I**mperfective and **C**ommand; (ii) **P**erson with the values **1**st, **2**nd, **3**rd; (iii) **G**ender with values **M**asculine and **F**eminine; (iv) **N**umber with values **S**imgular, **D**ual and **P**lural and (v) **S**tate with values **D**efinite, **I**ndefinite and **C**onstruct; (vi) **C**ase with values **N**ominative, **G**enitive and **A**ccusative; (vii) **V**oice with values **A**ctive and **P**assive and (viii) **M**ood with values **S**ubjunctive, **I**ndicative and **J**ussive. Not all the features mentioned are necessarily relevant to the dialects. In the full POS tag, the specified values of the different features will appear in the following order:

<POS>.<A><P><G><N>.<S><C><V><M>
The second period is not necessary if none of the last four features is specified.

Table 2 shows the list of POS tagset used in this annotation effort compared with the ones used ARZATB. The tagset is divided into three categories according to the tokenization scheme we follow: *proclitics* (14 tags), *enclitics* (2 tags) and *baseword* (39 tags). Together with the features, CAMEL POS tagset maps to ARZATB and retains backward compatibility. It also offers an intuitive Arabic scheme that is suitable to use for annotation.

For a subset of POS tags in the baseword category, each POS tag has a limited number of possible feature combinations that is paired with it. Below is the list of the POS tags that take features and their possible ordered combination.

- **NOUN, NOUN_*, ADJ, ADJ_*** All nominals take the combination of **G**ender, **N**umber. For example جالس *jAls* 'sitting' is tagged ADJ.MS ; In the occasional uses of **S**tate, such as طبعَاً *TabςAã* 'of course' the tag would be NOUN.MS.I
- **VERB** All verbs take the combination of **A**spect, **P**erson, **G**ender and **N**umber. For example يقطع *yqTς* 'cut' is tagged as VERB.I3MS
- **PRON** All pronouns take the combination of **P**erson, **G**ender and **N**umber. For example انتي *Anty* 'you [fs]' is tagged as PRON.2FS
- **PRON_DEM** All demonstrative pronouns take the combination of **G**ender and **N**umber. For example هَاذَا *hAðA* 'this' is tagged as PRON_DEM.3MS

In cases where a feature is not present, such as gender in verbs of first person inflections, the gender feature is simply dropped and does not require a placeholder since the possible feature values are ordered and unique. For example the imperfective 1st person verb اقول *Aqwl* 'I say' will be tagged as VERB.I1S

4.4. Lemma Guidelines

The lemma is the citation form of the the word. We follow the same guidelines of the lemma specification from Graff et al. (2009), where nominals are cited using the masculine singular form of the word or the feminine singular form if no masculine form exists. For example, the

CAMEL POS Arabic	CAMEL POS	ARZATB POS
***PROCLITIC* tags**		
أداة_تعريف	PART_DET	DET
حرف_عطف	CONJ	CONJ
حرف_جر	PREP	PREP
أداة_نفي	PART_NEG	NEG_PART
أداة_استقبال	PART_FUT	FUT_PART
أداة_مضارعة	PART_PROG	PROG_PART
أداة_ربط	CONJ_SUB	SUB_CONJ
ضمير_إشَارة	PRON_DEM	DEM_PRON
ضمير_استفهام	PRON_INTERROG	INTERROG_PRON
أداة	PART	PART
حرف_ربط	PART_CONNECT	CONNEC_PART
أداة_توكيد	PART_EMPHATIC	EMPHATIC_PART
جواب_شرط	PART_RC	RC_PART
أداة_نداء	PART_VOC	VOC_PART
***ENCLITIC* tags**		
أداة_نفي	PART_NEG	NEG_PART
ضمير	PRON	*SUFF_DO:[PGN]
ضمير	PRON	POSS_PRON_[PGN]
ضمير	PRON	PRON_[PGN]
***BASEWORD* tags**		
اسم	NOUN	NOUN
اسم_عدد	NOUN_NUM	NOUN_NUM
اسم_علم	NOUN_PROP	NOUN_PROP
اسم_كم	NOUN_QUANT	NOUN_QUANT
صفة	ADJ	ADJ
صفة_عدد	ADJ_NUM	ADJ_NUM
صفة_مقارنة	ADJ_COMP	ADJ_COMP
ظرف	ADV	ADV
ظرف_استفهام	ADV_INTERROG	INTERROG_ADV
ظرف_موصول	ADV_REL	REL_ADV
فعل	VERB	IV/PV/CV
شبه_فعل	VERB_PSEUDO	PSEUDO_VERB
اسم_فعل	VERB_NOM	VERB
ضمير	PRON	PRON_[PGN]
ضمير_إشَارة	PRON_DEM	DEM_PRON_[GN]
ضمير_استفهام	PRON_INTERROG	INTERROG_PRON
ضمير_تعجب	PRON_EXCLAM	EXCLAM_PRON
ضمير_موصول	PRON_REL	REL_PRON
أداة	PART	PART
أداة_تعريف	PART_DET	DET
أداة_نفي	PART_NEG	NEG_PART
أداة_استقبال	PART_FUT	FUT_PART
أداة_مضارعة	PART_PROG	PROG_PART
أداة_فعل	PART_VERB	VERB_PART
أداة_نداء	PART_VOC	VOC_PART
أداة_استفهام	PART_INTERROG	INTERROG_PART
أداة_استثناء	PART_RESTRICT	RESTRIC_PART
أداة_تفصيل	PART_FOCUS	FOCUS_PART
أداة_توكيد	PART_EMPHATIC	EMPHATIC_PART
جواب_شرط	PART_RC	RC_PART
أداة_ربط	CONJ_SUB	SUB_CONJ
حرف_جر	PREP	PREP
حرف_عطف	CONJ	CONJ
حرف_ربط	PART_CONNECT	CONNEC_PART
رقم	DIGIT	NOUN_NUM
اختصار	ABBREV	ABBREV
تعجب	INTERJ	INTERJ
أجنبي	FORIEGN	FOREIGN
علامة_ترقيم	PUNC	PUNC

Table 2: Table shows the CAMEL POS tagset used in the annotation of Annotated Gumar Corpus compared to the POS tagset in ARZATB. CAMEL POS Arabic shows the Arabic name of the tag.

lemma for the noun سيايير *syAyyr* 'cars' (NOUN.FP) is سَيَّارَة *say∼Arah* which is feminine singular since there is no masculine singular form of the word. The verbs are cited using the perfective 3rd person masculine singular form. For example, the lemma for the verb يشوفن *yšwfn* 'they see [f.p]' (VERB.I3FP) is شَاف *šAf* . For all other tags (i.e. particles, adverbs, ... etc) the lemma is the same form of the baseword. In this annotation effort, the lemma is the only form we require to be manually diacritized.

4.5. English Gloss Guidelines

The English gloss in this context refers to the semantic translation of the Arabic lemma. For nominals we use the singular form, and for verbs we use the infinitive form. An Arabic lemma could have multiple synonymous English glosses. For example كبير *kbyr* would have the following English glosses 'large; great; important; major; senior'.

4.6. Word Level Dialect Identification

Dialect identification is the task of tagging a certain context with a given dialect tag. Deciding the dialect tag depends on the context of the sentence and/or the document. This can be challenging since many words in their written form may be shared by many dialects and MSA. Additionally, it is not uncommon to find dialect code switching between MSA and a dialect, and even a dialect with another dialect (less commonly) (Elfardy and Diab, 2012). Hence we tag per word, but rely on the context of the sentence and even the document to identify the dialect.

In Table 3 we show an example of a fully annotated sentence and the POS tag in ARZATB for comparison. For full description of each of the annotation tasks and examples, the full guidelines can be accessed online.

5. Annotation Process

In this section, we discuss the annotation process details, the tool we used, and some annotation quality evaluation results.

5.1. MADARi Interface

We used a newly developed interface for morphological annotation and spelling correction called MADARi (Obeid et al., 2018). MADARi is a web-based interface that supports joint morphological annotation (tokenization, POS tagging, lemmatization) and spelling correction at any point of the annotation process, which minimizes error propagation. English glossing and dialect identification are also supported in the interface. MADARi assigns initial answers to the new text using MADAMIRA in EGY mode, whose databases we extended with $\text{CALIMA}_{\text{GLF}}$ for more coverage. MADARi has many utilities to facilitate the annotation process that we utilize for more efficiency, of which examples are discussed in the next subsection. Figure 1 shows a screenshot of the annotation view in MADARi.

5.2. Manual Annotation

The annotator starts on an automatically pre-annotated document. They carefully examine the spelling of each word and all its analysis choices in context with reference to the raw text at all times. For each word the annotator faces one of the following scenarios:

- All annotation tasks are correct: the annotator has to only validate the answer.
- Correct analysis but wrong spelling: the annotator has to adjust the spelling and then validate the answer.
- Wrong analysis (wholly or partially) but correct spelling: the annotator can manually adjust the analysis or can use the 'analysis search' utility provided by MADARi to get an analysis for a word with similar structure and then they would only have to change the lemma and the gloss entries. Finally they validate the answer.
- Wrong analysis and spelling: the annotator has to adjust the spelling and follow the previous step.

At any point of the annotation process, the annotator is able to apply mass changes to spelling and/or analysis across the document they are working on. However, the annotator must insure that all the words affected by the change are in similar contexts. The annotator can also modify their answers any time during the annotation through feedback they get if they have any inquiries. This allows the annotator to skip over words they are not confident about and leave the answer unvalidated.

Once the annotation task is fully completed, the annotator may 'submit' the finished document to be later exported. This will allow all the analyses made by the annotator to be accessible to all the other annotators when they look up the analysis for similar words.

5.3. Inter Annotator Agreement

We evaluated the quality of the annotation using the Inter Annotator Agreement (IAA) measure between two annotators on a selected text of 1,500 words. We measured the agreement on: (i) word boundary, that is the agreement on whether word boundaries are the same (no splits/merges); (ii) CODA spelling; (iii) baseword form; (iv) baseword POS; (v) baseword features; (vi) clitic form (averaged across all clitic positions) and (vii) clitic POS (averaged across all clitic positions). To align the pair of annotations, we perform a word level alignment within the sentences. We use a weighted Levenshtein distance to maximize alignment, where insertions and deletions are weighted as 1 and substitutions are weighted as follows:

$$W_{edit}(t_1, t_2) = \frac{2Lev(t_1, t_2)}{max(|t_1|, |t_2|)} \quad (1)$$

Above, t_1, t_2 are the two word tokens, and Lev is the Levenshtein distance at the *character* level. We employ this character-based weighing scheme to encourage the alignment of words with spelling changes. Using the same IAA measure, we measured the similarity between each annotator and the initial answers from the $\text{CALIMA}_{\text{GLF}}$-extended MADAMIRA.

The results are presented in Table 4 in terms of percent agreement. MADAMIRA provided a very helpful starting point. In at least 75% of the case, annotators agreed with

Raw sentence	خليفه يحس باليوع ويالس يقطع الدياي : الحمد لله ماباجي شي وبنفتك					
CODA sentence	خليفة يحس بالجوع وجالس يقطع الدجاج : الحمد لله ما باقي شي وبنفتك					
Transliteration	xlyfħ yHs bAljwς wjAls yqtς AldjAj : AlHmd llh mA bAqy šy wbnftk					
English Translation	Khalifa feeling hungry and cutting chicken: Thank God it is almost over					
Word	**CODA**	**TOK/POS**	**Lemma**	**Gloss**	**Dialect**	**ARZATB analysis**
خليفه	خليفة	خليفة/NOUN_PROP.MS	خَلِيفَة	Khalifa	GLF	خليف/NOUN_PROP+ة/NSUFF_FEM_SG
يحس	يحس	يحس/VERB.I3MS	حَسّ	feel	GLF	ي/IV3MS+حس/IV
باليوع	بالجوع	ب/PREP+ال/PART_DET+جوع/NOUN.MS	جُوع	hunger	GLF	ب/PREP+ال/DET+جوع/NOUN
ويالس	وجالس	و/PREP + جالس/ADJ.MS	جَالِس	sitting	GLF	و/PREP+جالس/ADJ
يقطع	يقطع	يقطع/VERB.I3MS	قَطَّع	cut	GLF	ي/IV3MS+قطع/IV
الدياي	الدجاج	ال/PART_DET+دجاج/NOUN.MS	دِجَاج	chicken	GLF	ال/DET+دجاج/NOUN
:	:	:/PUNC	:	:	GLF	:/PUNC
الحمد	الحمد	ال/PART_DET+حمد/NOUN.MS	حَمد	gratitude	GLF	ال/DET+حمد/NOUN
لله	لله	ل/PREP+الله/NOUN_PROP.MS	أَللّٰه	God	GLF	ل/PREP+الله/NOUN_PROP
ماباجي	ما	ما/PART_NEG	مَا	not	GLF	ما/NEG_PART
	باقي	باقي/ADJ.MS	بَاقِي	remaining	GLF	باقي/ADJ
شي	شي	شي/NOUN.MS	شَيّ	something	GLF	شي/NOUN
وبنفتك	وبنفتك	و/CONJ+ب/PART_FUT+نفتك/VERB.I1P	اِفتَكّ	get rid of	GLF	و/CONJ+ب/FUT_PART+ن/IV1P+فتك/IV

Table 3: An annotation example in the CAMEL POS scheme showing the different entries per word, in addition to the annotations in the ARZATB tagset for comparison. While Arabic is written from right to left, the tags above are displayed from left to right.

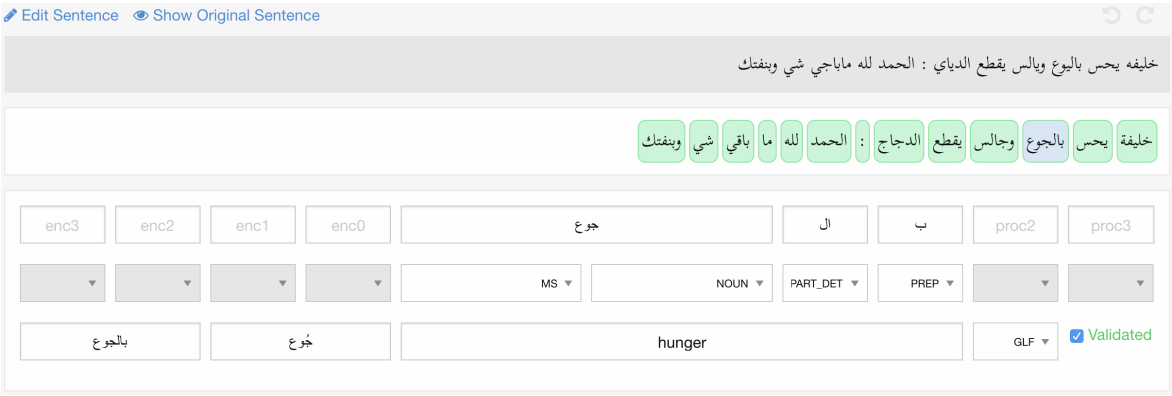


Figure 1: Example of the annotation step using the MADARi interface. The top gray box shows the raw sentence; next are the word tokens reflecting any spelling changes made. The section below shows all the fields required to annotate; they are initially populated using MADAMIRA. This example is of a manually annotated entry following the discussed guidelines.

Category	**A1 vs M**	**A2 vs M**	**A1 vs A2**
Word Boundary	89.7	89.1	98.9
CODA	78.8	78.1	94.7
Baseword Form	79.2	79.1	95.1
Baseword POS	80.2	80.4	96.1
Baseword Features	77.3	75.8	95.2
Average Clitic Form	96.0	95.9	99.4
Average Clitic POS	95.5	95.5	99.0

Table 4: Percentages of agreement between two annotators (i.e. A1 and A2) and between each annotator and the extended MADAMIRA (i.e. M) initial answers.

MADAMIRA's analysis choice. For each aligned pair of annotations, we compute the number of agreements for the considered categories (i–vii). The IAA score across the various categories ranges from 94.7% on CODA to over 99% on clitic annotations. Moreover, the measures between the annotators and MADAMIRA's answers show that both annotators changed many of the initial answers and their change was consistent to a large extent.[3]

6. Conclusion and Future Work

We presented an ongoing project for creating a manually annotated corpus of about 200,000 words of Emirati Arabic – the Annotated Gumar Corpus. We discussed the full guidelines for the different annotation components that include spelling adjustments, tokenization, POS tagging, lemmatization, English glossing and dialect identification. We used a newly developed interface for morphological annotation and spelling correction. We described the manual annotation process and finally measured the quality of the annotation through an IAA measure that found agreements

[3] At the time of writing this paper, the annotation of Parts 1, 2 and 3 had reached 75%, 65% and 66% of progress, respectively. The latest status of the annotation process can be viewed online along with all the guidelines mentioned in this paper. Please visit `http://resources.camel-lab.com`

ranging between 94.7% to more than 99% for different annotation tasks. In the future, we plan to expand the annotated text to include other genres and dialects. We are also interested in using the annotations to improve the quality of Arabic dialect POS tagging and morphological disambiguation.

7. Acknowledgements

This project is funded by a New York University Abu Dhabi Research Enhancement Fund. We would like to thank Ramy Eskander and the team of annotators at Ramitechs. We also thank the creators of MADARi from the MADAR project. Finally we thank Sondos Krouna for insightful discussions on POS decisions.

Bibliographical References

Abuata, B. and Al-Omari, A. (2015). A Rule-based Stemmer for Arabic Gulf Dialect. *Journal of King Saud University - Computer and Information Sciences*, 27(2):104 – 112.

Al-Sabbagh, R. and Girju, R. (2012). A Supervised POS Tagger for Written Arabic Social Networking Corpora. In Jeremy Jancsary, editor, *Proceedings of KONVENS 2012*, pages 39–52. ÖGAI, September. Main track: oral presentations.

Al-Shargi, F., Kaplan, A., Eskander, R., Habash, N., and Rambow, O. (2016). A Morphologically Annotated Corpus and a Morphological Analyzer for Moroccan and Sanaani Yemeni Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Alkuhlani, S. and Habash, N. (2011). A Corpus for Modeling Morpho-Syntactic Agreement in Arabic: Gender, Number and Rationality. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL'11)*, Portland, Oregon, USA.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A Multidialectal Parallel Corpus of Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Bouamor, H., Habash, N., Salameh, M., Zaghouani, W., Rambow, O., Abdulrahim, D., Obeid, O., Khalifa, S., Eryani, F., Erdmann, A., and Oflazer, K. (2018). The MADAR Arabic Dialect Corpus and Lexicon. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, May.

Cotterell, R. and Callison-Burch, C. (2014). A multi-dialect, multi-genre corpus of informal written Arabic. In *LREC*, pages 241–245.

Diab, M., Habash, N., Rambow, O., AlTantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *Proceedings of the LREC Workshop for Language Resources (LRs) and Human Language Technologies (HLT) for Semitic Languages: Status, Updates, and Prospects*.

Diab, M., Hawwari, A., Elfardy, H., Dasigi, P., Al-Badrashiny, M., Eskander, R., and Habash, N. (Forthcoming – 2013). Tharwa: A Multi-Dialectal Multi-Lingual Machine Readable Dictionary.

Elfardy, H. and Diab, M. (2012). Token Level Identification of Linguistic Code Switching. *Proceedings of COLING 2012: Posters*, pages 287–296.

Eskander, R., Habash, N., Rambow, O., and Pasha, A. (2016). Creating resources for Dialectal Arabic from a single annotation: A case study on Egyptian and Levantine. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3455–3465, Osaka, Japan.

Gadalla, H., Kilany, H., Arram, H., Yacoub, A., El-Habashi, A., Shalaby, A., Karins, K., Rowson, E., MacIntyre, R., Kingsbury, P., Graff, D., and McLemore, C. (1997). CALLHOME Egyptian Arabic Transcripts. In *Linguistic Data Consortium, Philadelphia*.

Graff, D., Maamouri, M., Bouziri, B., Krouna, S., Kulick, S., and Buckwalter, T. (2009). Standard Arabic Morphological Analyzer (SAMA) Version 3.1. Linguistic Data Consortium LDC2009E73.

Habash, N. and Rambow, O. (2006). MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. In *Proceedings of ACL*, pages 681–688, Sydney, Australia.

Habash, N., Soudi, A., and Buckwalter, T. (2007). On Arabic Transliteration. In A. van den Bosch et al., editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, N., Rambow, O., and Roth, R. (2009). MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In Khalid Choukri et al., editors, *Proceedings of the Second International Conference on Arabic Language Resources and Tools*. The MEDAR Consortium, April.

Habash, N., Eskander, R., and Hawwari, A. (2012a). A Morphological Analyzer for Egyptian Arabic. In *NAACL-HLT 2012 Workshop on Computational Morphology and Phonology (SIGMORPHON2012)*, pages 1–9.

Habash, N., Diab, M. T., and Rambow, O. (2012b). Conventional Orthography for Dialectal Arabic. In *LREC*, pages 711–718.

Habash, N., Khalifa, S., Eryani, F., Rambow, O., Abdulrahim, D., Erdmann, A., Faraj, R., Zaghouani, W., Bouamor, H., Zalmout, N., Hassan, S., shargi, F. A., Alkhereyf, S., Abdulkareem, B., Eskander, R., Salameh, M., and Saddiki, H. (2018). Unified Guidelines and Resources for Arabic Dialect Orthography. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, May.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Halefom, G., Leung, T., and Ntelitheos, D. (2013). A corpus of Emirati Arabic. Technical Report NRF Grant (31 H001), United Arab Emirates University.

Inoue, G., Shindo, H., and Matsumoto, Y. (2017). Joint Prediction of Morphosyntactic Categories for Fine-Grained Arabic Part-of-Speech Tagging Exploiting Tag Dictionary Information. In *Proceedings of the 21st Conference on Computational Natural Language Learning*

(CoNLL 2017), pages 421–431, Vancouver, Canada, August. Association for Computational Linguistics.

Jarrar, M., Habash, N., Akra, D., and Zalmout, N. (2014). Building a Corpus for Palestinian Arabic: a Preliminary Study. *ANLP 2014*, page 18.

Jarrar, M., Habash, N., Alrimawi, F., Akra, D., and Zalmout, N. (2016). Curras: An Annotated Corpus for the Palestinian Arabic dialect. *Language Resources and Evaluation*, pages 1–31.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016a). A Large Scale Corpus of Gulf Arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Portorož, Slovenia.

Khalifa, S., Zalmout, N., and Habash, N. (2016b). YAMAMA: Yet Another Multi-Dialect Arabic Morphological Analyzer. In *Proceedings of the International Conference on Computational Linguistics (COLING): System Demonstrations*, pages 223–227.

Khalifa, S., Hassan, S., and Habash, N. (2017). A Morphological Analyzer for Gulf Arabic Verbs. *WANLP 2017 (co-located with EACL 2017)*, page 35.

Kilany, H., Gadalla, H., Arram, H., Yacoub, A., El-Habashi, A., and McLemore, C. (2002). Egyptian Colloquial Arabic Lexicon. LDC catalog number LDC99L22.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.

Maamouri, M., Bies, A., Buckwalter, T., Diab, M., Habash, N., Rambow, O., and Tabessi, D. (2006). Developing and Using a Pilot Dialectal Arabic Treebank. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006*.

Maamouri, M., Bies, A., Krouna, S., Gaddeche, F., and Bouziri, B., (2009). *Penn Arabic Treebank Guidelines*. Linguistic Data Consortium.

Maamouri, M., Bies, A., Kulick, S., Tabessi, D., and Krouna, S. (2012a). Egyptian Arabic Treebank DF Parts 1-8 V2.0 - LDC catalog numbers LDC2012E93, LDC2012E98, LDC2012E89, LDC2012E99, LDC2012E107, LDC2012E125, LDC2013E12, LDC2013E21.

Maamouri, M., Krouna, S., Tabessi, D., Hamrouni, N., and Habash, N. (2012b). Egyptian Arabic Morphological Annotation Guidelines.

Maamouri, M., Bies, A., Kulick, S., Ciul, M., Habash, N., and Eskander, R. (2014). Developing an egyptian arabic treebank: Impact of dialectal morphology on annotation and tool development. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Masmoudi, A., Ellouze Khmekhem, M., Esteve, Y., Hadrich Belguith, L., and Habash, N. (2014). A corpus and phonetic dictionary for tunisian arabic speech recognition. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

McNeil, K. and Faiza, M. (2011). Tunisian Arabic Corpus : Creating a Written Corpus of an "Unwritten" Language.

Ntelitheos, D. and Idrissi, A. (2017). Language Growth in Child Emirati Arabic. *Hamid Ouali (ed.) Perspectives on Arabic Linguistics 29*, pages 229–248.

Obeid, O., Khalifa, S., Habash, N., Bouamor, H., Zaghouani, W., and Oflazer, K. (2018). MADARi: A Web Interface for Joint Arabic Morphological Annotation and Spelling Correction. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, May.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Shoufan, A. and Al-Ameri, S. (2015). Natural Language Processing for Dialectical Arabic: A Survey. In *ANLP Workshop 2015*, page 36.

Smaïli, K., Abbas, M., Meftouh, K., and Harrat, S. (2014). Building resources for Algerian Arabic dialects. In *15th Annual Conference of the International Communication Association Interspeech*.

Smrž, O. (2007). ElixirFM — Implementation of Functional Arabic Morphology. In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, pages 1–8, Prague, Czech Republic, June. ACL.

Zaghouani, W. (2014). Critical Survey of the Freely Available Arabic Corpora. In *Proceedings of the Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools, LREC*, pages 1–8.

Zaidan, O. F. and Callison-Burch, C. (2011). The Arabic Online Commentary Dataset: an Annotated Dataset of Informal Arabic With High Dialectal Content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41.

Zalmout, N. and Habash, N. (2017). Don't Throw Those Morphological Analyzers Away Just Yet: Neural Morphological Disambiguation for Arabic. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 704–713.

Zalmout, N., Erdmann, A., and Habash, N. (2018). Noise-Robust Morphological Disambiguation for Dialectal Arabic. In *Proceedings of the 26th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies Conference (HLT-NAACL2018)*.

Zeroual, I. and Lakhouaja, A., (2018). *Arabic Corpus Linguistics: Major Progress, But Still A Long Way to Go*, pages 613–636. Springer International Publishing, Cham.

Zribi, I., Ellouze, M., Belguith, L. H., and Blache, P. (2015). Spoken Tunisian Arabic Corpus" STAC": Transcription and Annotation. *Research in computing science*, 90:123–135.

CoNLL-UL: Universal Morphological Lattices for Universal Dependency Parsing

Amir More,[1,*] Özlem Çetinoğlu,[2] Çağrı Çöltekin,[3] Nizar Habash,[4] Benoît Sagot,[5] Djamé Seddah,[5,6] Dima Taji,[4] Reut Tsarfaty[1]

[1]Open University of Israel, [2]Institut fur Maschinelle Sprachverarbeitung, [3]University of Tübingen, [4]New York University Abu Dhabi, [5]INRIA, [6]Université Paris Sorbonne

habeanf@gmail.com, ozlem@ims.uni-stuttgart.de, ccoltekin@sfs.uni-tuebingen.de, nizar.habash@nyu.edu benoit.sagot@inria.fr, djame.seddah@inria.fr, dima.taji@nyu.edu, reutts@openu.ac.il

Abstract

Following the development of the *universal dependencies* (UD) framework and the CoNLL 2017 Shared Task on end-to-end UD parsing, we address the need for a universal representation of morphological analysis which on the one hand can capture a range of different alternative morphological analyses of surface tokens, and on the other hand is compatible with the segmentation and morphological annotation guidelines prescribed for UD treebanks. We propose the *CoNLL universal lattices* (CoNLL-UL) format, a new annotation format for word lattices that represent morphological analyses, and provide resources that obey this format for a range of typologically different languages. The resources we provide are harmonized with the two-level representation and morphological annotation in their respective UD v2 treebanks, thus enabling research on universal models for morphological and syntactic parsing, in both pipeline and joint settings, and presenting new opportunities in the development of UD resources for low-resource languages.

Keywords: Morphology, Universal Dependencies, Morphological Analysis, Morphological Ambiguity

1. Introduction

The development of the *universal dependencies* (UD) framework and its treebank collection (Nivre et al., 2016; Nivre et al., 2017) follows many shared tasks and multilingual evaluation campaigns in which the linguistic representation schemes across different languages vary (Buchholz and Marsi, 2006; Nivre et al., 2007; Seddah et al., 2013; Butt et al., 2002; Zeman et al., 2012). The UD treebanks collection, in contrast, obeys a single set of annotation guidelines, and respects the discrepancies between surface input tokens and the output nodes in the syntax trees (a.k.a., the *two-level representation* principle.)[1]

The UD initiative has paved the way to the development of cross-lingual models for word segmentation, part-of-speech tagging and dependency parsing (Straka and Straková, 2017), as well as cross-linguistic typological investigations (Futrell et al., 2015). Recently, the CoNLL 2017 Shared Task on Multilingual UD Parsing (Zeman et al., 2017), which used a variant of the UD datasets from the UD v2.0 release, introduced a truly end-to-end parsing setting: participants had to parse raw texts into dependency trees, implying the initial phases of sentence tokenisation, word segmentation, part-of-speech tagging and morphological annotation (if their parsing models required morphological information in their input).

The UD annotation scheme provides guidelines for unambiguously annotating the morphological and syntactic levels of representation of natural language sentences, but it does not provide means to formally capture a range of potential analyses, and in particular the *ambiguous morphological analyses*, of raw surface tokens. Lexical resources that capture this ambiguity, in the form of morphological analyzers or existing lexicons, are available for many of the participating treebanks,[2] but they are far from useful for UD parsing, for two main reasons: (a) existing lexical resources rely on variety of formats and underlying theories that are incompatible with the UD morphological scheme, and (b) the morpho-syntactic interface assumed by these tools' representation is often incompatible with the respective trees in the UD treebanks.

To fill this gap, we propose an annotation format which allows for capturing the full range of potentially-ambiguous morphological analyses of raw surface tokens, and at the same time is compatible with UD treebanks in form and function and respects its two-level representation principle. We name this format "CoNLL-UL", in which UL stands for a *universal lattice* structure meant for formally capturing morphological ambiguity. In addition, we provide adaptations of existing morphological analyzers and lexica to our proposed format, making a wide range of CoNLL-UL resources freely available for the community.

Our contribution is hence many-fold. We first introduce a UD-compatible annotation format that is suitable for representing competing morphological analyses — each of which consisting of word segmentation, POS tagging and morphological features — for tokens or token sequences. Secondly, we provide a set of lexical resources for broad-coverage morphological analysis obeying this CoNLL-UL format, based on different sources: (i) For Arabic, Hebrew, and Turkish, morphologically rich-and-heavily-ambiguous languages, we developed or adapted morphological analyzers such that their output is in the CoNLL-UL format and subscribe to the word segmentation and morphological annotation theories of their respective UD v2 treebanks (Section 3.1). (ii) For several languages with less-complex morphology, we converted the output of existing freely-

* Corresponding author.

[1] http://universaldependencies.org/

[2] universaldependencies.org/conll17/data.html

FROM	TO	FORM	LEMMA	UPOS	CPOS	FEATURES	MISC	ANCHORS
0	1	*her*	her	DET	_	Definite=Def	_	goldid=1
1	2	*şey*	şey	NOUN	_	Case=Nom|Number=...	_	goldid=2
2-4	güzeldi	_						
2	3	*güzel*	güzel	ADJ	_	Case=Nom|Number=...	_	goldid=3
3	4	*di*	i-	VERB	_	Aspect=Perf...	_	goldid=4

(a) Morphological analysis of the Turkish phrase `her şey güzeldi` (everything was beautiful) in the CoNLL-UL format.

ID	FORM	LEMMA	UPOS	CPOS	FEATS	HEAD	DEPREL	DEPS	MISC
1	*her*	her	DET	_	Definite=Def	2	det	_	_
2	*şey*	şey	NOUN	_	Case=Nom|Numbe...	3	nsubj	_	_
3-4	güzeldi								
3	*güzel*	güzel	ADJ	_	Case=Nom|Numbe...	0	root	_	_
4	*di*	i-	VERB	_	Aspect=Perf...	3	cop	_	_

(b) The CoNLL-U representation of the Turkish phrase `her şey güzeldi .`

Table 1: The relationship between CoNLL-UL and CoNLL-U for a linear (unambiguous) lattice.

available morphological lexicons to the CoNLL-UL format. In that way, these lexica can be used for generating all possible analyses of a given token in the required lattice structure, provided it is known to the lexicon (see Section 3.2). (iii) For languages that do not have freely-available (if any) lexical resources, we provide a rudimentary tool to induce a CoNLL-UL lexicon from a UD treebank, which can be used as a baseline broad-coverage morphological analyzer.

We propose that CoNLL-UL will serve as a complement to the CoNLL-U format, and likewise, that conforming lexical resources will complement the respective treebanks in the UD treebank collection. CoNLL-UL will help researchers exchange language resources and tools at the morphological level, therefore improving their systems and allowing for proper cross-lingual comparison. Moreover, universal access to broad-coverage lexical resources harmonized with the UD treebanks scheme will pave the way for (otherwise infeasible) research on joint models for universal morpho-syntactic parsing.

We detail our proposal, motivated by the CoNLL-2017 shared task, in Section 2, and describe the resources we make available in Section 3. We cover related work and contrast particularities of broad-coverage morphological analyses with the morphology specifically annotated in the UD treebanks, and discuss the limitations of UD morpho-syntax — to be potentially addressed by the UD community in the future — in Section 4, and we conclude with a summary of our contributions in Section 5.

2. The CoNLL-UL Proposal

In this section we detail our proposed universal annotation format for morphological ambiguity, which represents the competing analyses of a given source token.

Throughout this paper, we use the following terminology. A *source token* is a sequence of characters in the raw input text, segmented from surrounding characters by conventional, typographic (non linguistic) criteria. It corresponds to "tokens" in the UD model. A *tree token* is a lexical unit output by a morphological analyzer that is meant to serve as a leaf node in syntactic structures. It corresponds to "words" in the UD model. The CoNLL-UL annotation format scheme is similar to the SPMRL lattice format, where every lattice represents a surface token, and each edge in the lattice represents a tree token with the following properties:

FROM: Index of the outgoing vertex of the edge;

TO: Index of the incoming vertex of the edge;

FORM: Tree token (word form or punctuation mark);

LEMMA: Lemma or stem of the word form; underscore if not available;

UPOSTAG: Universal POS tag;

XPOSTAG: Language-specific POS tag; underscore if not available;

FEATS: List of morphological features from the universal feature inventory or from a pre-defined language-specific extension; underscore if not available;

MISC: Any other annotation related to morphology; underscore if not available;

ANCHORS: Identifiers linking to specific disambiguation if needed; underscore if not available

We also borrow from the UD format properties for specifying a surface token spanning multiple tree tokens, respecting the two-level representation principle:

RANGE: Start and end vertex ids in the tree token lattice;

TOKEN: Source token;

MISC: Any other token-level annotation (e.g. spelling issue or canonical representation); underscore if not available

We generalise this notation for dealing with source tokens that are different from the corresponding tree token in the case of 1-to-1 mappings, using ranges of length one.

Note that as opposed to CONLL-U, integer indices in the FROM, TO and RANGE fields index vertices in a lattice, and not tree tokens. Nevertheless, in the case of a linear lattice (a lattice with only one path) as shown in Tables (1a) and (1b), the CoNLL-U representation can be directly obtained as follows:

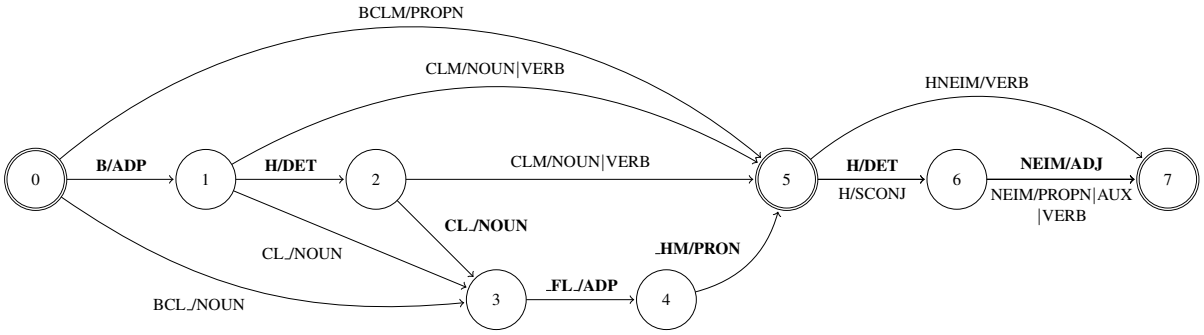


Figure 1: Lattice of possible analyses in terms of transliterated tree token sequences for the first two Hebrew source tokens of the phrase בצלם הנעים של העצים (BCLM HNEIM FL HECIM, meaning "in the pleasant shadow/shade of the trees", using the transliteration scheme of Sima'an et al. (2001)). The correct disambiguation of each source token, in the context of the phrase, is highlighted in bold. Lemma and morphological property ambiguity are not shown for brevity; '|' indicates part-of-speech ambiguity

FROM	TO	FORM	LEMMA	UPOS	CPOS	FEATURES	MISC	ANCHORS
0-5	בצלם	_						
0	5	*BCLM*	BCLM	PROPN	_	_	_	_
0	1	*B*	B	ADP	_	_	_	goldid=1
0	3	*BCL_*	BCL	NOUN	_	Gender=Masc|Number=Sing	_	_
1	2	*H*	H	DET	_	PronType=Art	_	goldid=2
1	3	*CL_*	CL	NOUN	_	Gender=Masc|Number=Sing	_	_
2	3	*CL_*	CL	NOUN	_	Gender=Masc|Number=Sing	_	goldid=3
2	5	*CLM*	CILM	VERB	_	Gender=Masc|Number=Si…	_	_
2	5	*CLM*	CILM	VERB	_	Gender=Masc|Mood=Imp…	_	_
2	5	*CLM*	CLM	NOUN	_	Definite=Cons|Gender=Mas…	_	_
2	5	*CLM*	CLM	NOUN	_	Gender=Masc|Number=Sing	_	_
3	4	*_FL_*	FL	ADP	_	_	_	goldid=4
4	5	*_HM*	ANI	PRON	_	Gender=Masc|Number=Plur|Person=3	_	goldid=5
5-7	הנעים	_						
5	7	*HNEIM*	HNEIM	VERB	_	Gender=Masc…	_	_
5	6	*H*	H	DET	_	PronType=Art	_	goldid=6
5	6	*H*	H	SCONJ	_	_	_	_
6	7	*NEIM*	NEIM	ADJ	_	Gender=Masc|Number=Sing	_	goldid=7
6	7	*NEIM*	NEIM	ADJ	_	Definite=Cons|Gender=…	_	_
6	7	*NEIM*	NEIM	ADV	_	Polarity=Neg	_	_
6	7	*NEIM*	NEIM	AUX	_	Gender=Masc|Number=Sing…	_	_
6	7	*NEIM*	NEIM	PROPN	_	_	_	_
6	7	*NEIM*	NEIM	VERB	_	Gender…|Tense=Part|VerbForm=Part	_	_
6	7	*NEIM*	NEIM	VERB	_	Gender…|VerbForm=Part	_	_

Table 2: The CoNLL-UL representation of the lattice shown in Figure 1. Tree tokens and lemmas are transliterated here for the convenience of the reader. Some morphological features strings are shortened for brevity. Note the reuse of columns for source token span lines as in CoNLL-U, where the FROM column is a range, the TO column is a source token, and the FORM column is a misc field set to underscore when empty

- ignore the "FROM" column and the "MISC" column for source tokens,
- ignore source token specifications with ranges of length 1,
- ignore the "ANCHORS" column, and
- increment by one the starting vertex id for ranges with length greater than 1.[3]

We now illustrate this format on an example. Let us consider the Hebrew source token sequence בצלם הנעים (transliterated as BCLM HNEIM).[4] Figure 1 displays the lattice of possible analyses in terms of tree tokens, as could be output by a non-deterministic morphological analyzer. This lattice illustrates two ambiguity types: (i) *morphological segmentation ambiguity*, which is directly visible in the different path lengths in the lattice, (ii) *morphological tagging ambiguity*, visible in the two analyses provided for the tree token **צלם** (*CLM*). The corresponding CoNLL-UL representation is provided in Table 2.

[3]Only the starting vertex id is incremented since ranges in CoNLL-UL correspond to vertices in lattices, of which there are two (start and end) for each possible tree token, whereas ranges in CoNLL-U correspond to tree tokens themselves. Lattice vertices are 0-indexed, therefore the starting vertex id is incremented, rather than the ending vertex id decremented.

[4]Henceforth, source tokens will be typeset in `typewriter` and tree tokens in *italics* throughout the paper.

FROM	TO	FORM	LEMMA	UPOS	CPOS	FEATURES	MISC
0	1	*encodent*	encoder	VERB	_	Mood=Ind|Number=Plur|Person=3|Tense=Pres…	_
0-2	auxquels	_					
0	1	*à*	à	ADP	_	_	_
1	2	*lesquels*	lequel	PRON	_	Gender=Masc|Number=Plur	_

Table 3: Two entries resulting from the conversion of the L*efff* in the CoNLL-UL format.

Let us now consider the correct segmentation ב ה צל של הם (*B H CL FL HM*, meaning in-the-shadow-of-them) of the source token בצלם (BCLM) in context. These segments would form the syntactic words of a CoNLL-U file describing UD trees. We relate the CoNLL-UL morphologically ambiguous files to CoNLL-U unambiguous files by *anchoring* lattice arcs with their syntactic identifiers in the ANCHORS field. This simplifies the process of merging one or more morphological disambiguations, a necessary step for evaluation of the prediction and for joint morpho-syntactic processing.

3. Morphological Analysis with CoNLL-UL

We have developed a set of resources and tools that can perform morphological analysis and output it in the CoNLL-UL format. Morphological analysis can be performed either online, using for instance finite-state or statistical/neural models, and/or based on lexical resources. In this section, we first briefly describe several CoNLL-UL-compatible morphological analyzers we developed for languages such as Arabic, Hebrew and Turkish; we then sketch how we converted existing lexicons into the CoNLL-UL format, which can be used straightforwardly as the basis for simple morphological analyzers.

The Hebrew and Turkish morphological analyzers, and lexicons mentioned in this paper are freely available. The Arabic morphological analyzer requires a license, which can be acquired by following the instructions at the provided link. In addition, we apply our analyzers to existing UD treebanks, and provide the resulting CoNLL-UL analyses to the community.[5]

3.1. Morphological Analyzers for MRLs

As our first contribution to the bootstrapping of universal morphological resources, we provide here adaptations of morphological analyzers for three Morphologically Rich Languages (MRLs): Arabic, Hebrew, and Turkish. For these morphological analyzers, their pre-existing morphological analyses adhere to schemes that differ from those employed in the respective UD treebanks. These discrepancies are due to differences between the morphological theories adopted by the UD treebanks developers and those employed by the developers of the morphological analyzers. Therefore, the adapted resources we provide are non-trivial to obtain, and required careful alignment of the morpho-syntactic analyses with their UD treebank counterparts.

For Arabic, we adapted the morphological analyzer used in MADAMIRA (Pasha et al., 2014), which is built on top of the databases of SAMA (Maamouri et al., 2010) to output morphology that adheres to the UD Arabic treebank (Taji et al., 2017).[6] The Arabic UD treebank, as with other Arabic treebanks, uses the Penn Arabic treebank tokenization scheme (Maamouri et al., 2004) which segments all proclitics and enclitics except for the definite article. It is worth noting that the format we propose here is independent of the specifics of this tokenization scheme and it can be used with a number of other schemes (Habash, 2010).

For Hebrew, we used the HEBLEX morphological analyzer of More and Tsarfaty (2016), based on the BGU Lexicon (Itai and Wintner, 2008), adapted to the UD Hebrew treebank.[7] We only modified the HEBLEX SPMRL lattices format to follow the proposed CoNLL-UL format, as the HEBLEX annotations have already been adapted to the treebank counterpart (More and Tsarfaty, 2017).

For Turkish, we developed a new morphological analyzer based on TRmorph (Çöltekin, 2010).[8] The analyzer follows the segmentation and morphological analysis scheme of the UD Turkish treebank v2.0 (Sulubacak et al., 2016) and Turkish-PUD treebank (Zeman et al., 2017). These treebanks have employed a different segmentation approach compared to the METU-Sabancı Turkish Treebank (Oflazer et al., 2003). In addition, form and lemma representations, POS tags and morphological tag sets have changed. The existing morphological analyzers are not compatible with this new representation. Thus we introduce a finite-state implementation that complies with the UD v2.0 scheme.

On top of that, for languages in the UD treebanks collection that may not have existing lexical resources and/or morphological analyzers publicly available, we adapted a data-driven rudimentary morphological analyzer (More and Tsarfaty, 2017) that induces a morphological lexicon from existing UD treebanks which provide broad-coverage morphological analyses and adhere to the proposed CoNLL-UL format. The analyzer can use the induced lexicon, as well as the converted morphological lexicons below, to provide analyses of input text in the CoNLL-UL format.

3.2. Converted Morphological Lexicons

As a complement to the CoNLL-UL-compatible analyzers described above, we have created a set of 53 CoNLL-UL-compatible morphological lexicons covering 38 languages, based on existing freely available resources.[9] The source lexicons, the conversion processes and the resulting inventory of freely available CoNLL-UL lexicons are described

[5] https://conllul.github.io

[6] https://camel.abudhabi.nyu.edu/calima-star/

[7] https://github.com/habeanf/yap

[8] https://github.com/coltekin/TRmorph/tree/trmorph2

[9] http://pauillac.inria.fr/~sagot/udlexicons.html

in (Sagot, 2018).[10] Here we only provide in Table 3 two examples converted from the Lefff, the Alexina lexicon for French. The first one illustrates the 1-to-1 case, with an entry converted from the following original entry:

encodent encoder v P3p,

which includes the wordform (i.e. the [source and tree] token) *encodent* 'encode$_{\text{3pl.pres.ind}}$', its lemma, its Lefff POS and its Lefff morphosyntactic tag. The other example illustrates the 1-to-m case with the source token `auxquels`, which is analyzable as reflecting the sequence of two tree tokens *à lesquels* 'to which'.

4. Related Work and Perspective

Our work overlaps somewhat with previous proposals, in particular the ISO norm "Morphosyntactic Annotation Framework" (hereafter MAF, (Clément and Villemonte de La Clergerie, 2005)).[11] In principle, MAF allows for the representation of any analysis represented in CoNLL-UL, whereas not every analysis in MAF can be represented in CoNLL-UL. This is because CoNLL-UL is intentionally coupled to CoNLL-U in both form and function: first, we use the CoNLL-U flat, tab-delimited file format for consistency and ease of use; second, we intentionally impose the same restrictions on CoNLL-UL morphology that UD itself is restricted to, such that these two resources maintain harmony. As a result of the latter, we can maintain a two-way compatibility promise: every morphological disambiguation in a UD v2 treebank can be represented as a CoNLL-UL lattice, and every possible path in a CoNLL-UL lattice can serve as the syntactic words of a UD-annotated tree. Thus, we ease the burden on morphological and syntax parser research and development, such that they are relieved of adapting lexical resources (or their analyses) to UD-compliant morphology.

The representation scheme for lattices used by the SPMRL shared task datasets (Seddah et al., 2013) and which were introduced by (Tsarfaty et al., 2012; Tsarfaty, 2013),[12] allowed for annotating morphological ambiguity of these same languages. Seeker and Çetinoğlu (2015) extended the SPMRL representation to accommodate marking the gold and optionally a predicted morphological analysis. Our proposal extends the latter with two additions: (i) we use the UD convention of specifying a surface token spanning multiple tree tokens; and (ii) we allow the specification of multiple anchors relating lattice arcs to tree tokens, for possibly grounding more than one syntactic tree (i.e., a forest) in the morphological lattice.

Since we wanted to maintain compatibility with the current version of CoNLL-U, our CoNLL-UL proposal has some limitations. First, although it fully covers 1-to-1 and 1-to-m mappings between source and tree tokens, it only covers n-to-1 mappings via "words with spaces" or special 'GoesWith' dependencies, and does not cover n-to-m mappings. Yet such cases do occur in many languages.[13] It is especially true when taking into account noisy user-generated content and speech productions. Moreover, our proposal does not cover all types of lattices, including cases that cannot be covered with only one set of indices, as used in CoNLL-U and CoNLL-UL. There are lattice shapes which require independent mechanisms for indexing source and tree tokens (or, rather, states in a tree token lattice).[14] However, addressing all these cases would require a format that could not be directly compatible with the (current version of the) UD format/model. We therefore leave open the complete investigation of these issues for future work.

5. Conclusion

Although lexical resources for morphological analysis exist for many languages, they respect varied approaches to morphology. In the context of the CoNLL 2017 Shared Task, the morphological annotation in UD treebanks has not been harmonized with these existing lexical resources, making it hard to develop joint morpho-syntactic parsers.

We propose the CoNLL-UL annotation for morphological ambiguity and provide adapted resources harmonized with existing UD resources. CoNLL-UL addresses the need for a UD/CoNLL-U interface to existing lexical resources, and our adapted resources provide a good starting point, with at least three important MRLs: Arabic, Hebrew, and Turkish. We also adapt morphological lexicons in the Apertium, Giellatekno, and Alexina frameworks, providing lexical resource coverage for numerous languages. For languages without lexical resources, we provide a baseline solution using the UD treebanks to induce data-driven lexica.

We suggest our proposal, together with the adapted resources and tools we provide, form to complement the set of UD treebanks, and facilitate research and development of cross-lingual morphological and syntactic parsing.

6. Acknowledgements

The authors would like to thank Joakim Nivre for his insightful comments. The work of the first and last authors is supported via research grants by the European Research Council (ERC-StG 677352) and the Israeli Science Foundation (ISF 1739/26), for which we are grateful. Özlem Çetinoğlu is supported by the Deutsche Forschungsgemeinschaft (DFG) via the SFB 732, project D2. Benoît Sagot and Djamé Seddah were partly funded by the French ANR projects ParSiTi (ANR-16-CE33-0021 and SoSweet (ANR-15-CE38-0011-01), as well as by the Program "Investissements d'avenir" ANR-10-LABX-0083 (Labex EFL).

[10] The lexical information is represented in the CoNLL-UL format, with a minor adaptation; as a lexicon is not a collection of sentences, but a collection of entries, each entry is annotated as a separate sentence, but "sentence" boundaries are not included.

[11] Official page on the ISO website: `https://www.iso.org/fr/standard/51934.html?browse=tc`. Freely available earlier working draft: `http://atoll.inria.fr/~clerger/MAF/html/index.html`.

[12] `http://www.spmrl.org`.

[13] See for instance in French the sequence of source tokens `près du Mans` 'near Le Mans' and its analysis in terms of tree tokens *près_de Le_Mans*, where tree token *près_de* "corresponds" to source token `près` and the "first half" of the source token `du`, and tree token *Le_Mans* "corresponds" to the "second half" of source token `du` and source token `Mans`.

[14] See for instance the non canonical French source token sequence `c t`, which can be analysed (at least) in two ways, namely as the tree token sequence *c' était* 'it was' and the tree token sequence *c' est tes* 'it is your$_{\text{plur.}}$'.

7. Bibliographical References

Buchholz, S. and Marsi, E. (2006). CoNLL-X shared task on multilingual dependency parsing. In *Proc. of the Tenth Conference on Computational Natural Language Learning*, pages 149–164, New York City, USA.

Butt, M., Dyvik, H., King, T. H., Masuichi, H., and Rohrer, C. (2002). The parallel grammar project. In *Proceedings of the 2002 Workshop on Grammar Engineering and Evaluation - Volume 15*, COLING-GEE '02, pages 1–7, Stroudsburg, PA, USA. Association for Computational Linguistics.

Çöltekin, Ç. (2010). A freely available morphological analyzer for Turkish. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, pages 820–827.

Clément, L. and Villemonte de La Clergerie, E. (2005). MAF: a morphosyntactic annotation framework. In *proc. of the 2nd Language & Technology Conference (LT'05)*, pages 90–94, Poznan, Poland.

Futrell, R., Mahowald, K., and Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Itai, A. and Wintner, S. (2008). Language resources for Hebrew. *Language Resources and Evaluation*, 42(1):75–98, March.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The penn arabic treebank: Building a large-scale annotated arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467.

Maamouri, M., Graff, D., Bouziri, B., Krouna, S., Bies, A., and Kulick, S. (2010). Standard arabic morphological analyzer (sama) version 3.1. *Linguistic Data Consortium, Catalog No.: LDC2010L01*.

More, A. and Tsarfaty, R. (2016). Data-driven morphological analysis and disambiguation for morphologically rich languages and universal dependencies. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 337–348, Osaka, Japan, December. The COLING 2016 Organizing Committee.

More, A. and Tsarfaty, R. (2017). Universal joint morph-syntactic processing: The open university of israel's submission to the conll 2017 shared task. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 253–264.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 shared task on dependency parsing. In *Proc. of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal dependencies v1: A multilingual treebank collection. In *LREC*.

Nivre, J., Agić, Ž., Ahrenberg, L., Aranzabe, M. J., Asahara, M., Atutxa, A., Ballesteros, M., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bosco, C., Bouma, G., Bowman, S., Candito, M., Cebiroğlu Eryiğit, G., Celano, G. G. A., Chalub, F., Choi, J., Çöltekin, Ç., Connor, M., Davidson, E., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dobrovoljc, K., Dozat, T., Droganova, K., Dwivedi, P., Eli, M., Erjavec, T., Farkas, R., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D., Garcia, M., Ginter, F., Goenaga, I., Gojenola, K., Gökırmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Grūzītis, N., Guillaume, B., Habash, N., Hajič, J., Hà Mỹ, L., Haug, D., Hladká, B., Hohle, P., Ion, R., Irimia, E., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kotsyba, N., Krek, S., Laippala, V., Lê Hồng, P., Lenci, A., Ljubešić, N., Lyashevskaya, O., Lynn, T., Makazhanov, A., Manning, C., Mărănduc, C., Mareček, D., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Missilä, A., Mititelu, V., Miyao, Y., Montemagni, S., More, A., Mori, S., Moskalevskyi, B., Muischnek, K., Mustafina, N., Müürisep, K., Nguyễn Thị, L., Nguyễn Thị Minh, H., Nikolaev, V., Nurmi, H., Ojala, S., Osenova, P., Øvrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Ramasamy, L., Real, L., Rituma, L., Rosa, R., Saleh, S., Sanguinetti, M., Saulīte, B., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shakurova, L., Shen, M., Sichinava, D., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Tsarfaty, R., Tyers, F., Uematsu, S., Uria, L., van Noord, G., Varga, V., Vincze, V., Washington, J. N., Žabokrtský, Z., Zeldes, A., Zeman, D., and Zhu, H. (2017). Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Oflazer, K., Say, B., Hakkani-Tür, D. Z., and Tür, G. (2003). Building a Turkish Treebank. In Anne Abeille, editor, *Building and Exploiting Syntactically-annotated Corpora.* Kluwer Academic Publishers, Dordrecht.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Sagot, B. (2018). A Multilingual Collection of CoNLL-U-compatible Morphological Lexicons. In *In proceedings of the 11th Language Resources and Evaluation Conference (LREC 2018)*, Miyazaki, Japan.

Seddah, D., Tsarfaty, R., Kübler, S., Candito, M., Choi, J. D., Farkas, R., Foster, J., Goenaga, I., Gojenola Galletebeitia, K., Goldberg, Y., Green, S., Habash, N., Kuhlmann, M., Maier, W., Nivre, J., Przepiórkowski, A., Roth, R., Seeker, W., Versley, Y., Vincze, V., Woliński, M., Wróblewska, A., and Villemonte de La Clergerie, E. (2013). Overview of the SPMRL 2013 Shared Task: A Cross-Framework Evaluation of Parsing Morphologically Rich Languages. In *Proceedings of the Fourth*

Workshop on Statistical Parsing of Morphologically-Rich Languages, pages 146–182, Seattle, Washington, USA.

Seeker, W. and Çetinoğlu, O. (2015). A graph-based lattice dependency parser for joint morphological segmentation and syntactic analysis. *TACL*, 3:359–373.

Sima'an, K., Itai, A., Winter, Y., Altman, A., and Nativ, N. (2001). Building a tree-bank of Modern Hebrew text. *Traitment Automatique des Langues*, 42(2).

Straka, M. and Straková, J. (2017). Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August. Association for Computational Linguistics.

Sulubacak, U., Gokirmak, M., Tyers, F., Çöltekin, Ç., Nivre, J., and Eryiğit, G. (2016). Universal dependencies for turkish. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3444–3454, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Taji, D., Habash, N., and Zeman, D. (2017). Universal dependencies for arabic. *WANLP 2017 (co-located with EACL 2017)*, page 166.

Tsarfaty, R., Nivre, J., and Andersson, E. (2012). Joint evaluation of morphological segmentation and syntactic parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 6–10, Jeju Island, Korea, July. Association for Computational Linguistics.

Tsarfaty, R. (2013). A unified morpho-syntactic scheme of stanford dependencies. In *ACL (2)*, pages 578–584.

Zeman, D., Mareček, D., Popel, M., Ramasamy, L., Štěpánek, J., Žabokrtský, Z., and Hajič, J. (2012). Hamledt: To parse or not to parse? In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gökırmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, Ç., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada, August. Association for Computational Linguistics.

Manually Annotated Corpus of Polish Texts Published between 1830 and 1918

Witold Kieraś, Marcin Woliński
Institute of Computer Science, Polish Academy of Sciences
Jana Kazimierza 5, 01-248 Warszawa, Poland
wkieras@ipipan.waw.pl, wolinski@ipipan.waw.pl

Abstract

The paper presents a manually annotated 625,000 tokens large historical corpus of Polish. The corpus consists of samples from texts published between 1830 and 1918 — fiction, drama, popular science, essays and newspapers of the period. The corpus provides three layers: transliteration, transcription and morphosyntactic annotation. The annotation process as well as the corpus itself are described in detail in the paper.

Keywords: historical corpora, manually annotated corpora, corpus linguistics, Polish

1. About the project

The paper presents a historical manually annotated corpus of Polish. The corpus consists of samples excerpted from texts published between 1830 and 1918 and is morphosyntactically annotated for the purpose of a larger project aimed at creating a diachronic model of Polish inflection. Together with two other manually annotated corpora (one historical and one contemporary) it will serve as a point of reference for a corpus-driven research in diachronic computational morphology of Polish.

2. Related work

The annotation of the presented corpus took place in parallel with the Baroque corpus of Polish project in which a manual annotation of a gold-standard dataset was also conducted (Bronikowska et al., 2016). The two tasks shared the same web application developed specially for both projects and kept close cooperation in many details. The Baroque corpus covers a 1601-1772 time span leaving over half a century gap between the two projects, which will be hopefully filled in future projects. We are not aware of existence of any other manually annotated historical corpora of Polish.

Among resources for other Slavonic languages a relatively similar project was accomplished for Slovene (Erjavec, 2012; Erjavec, 2015) where a 300,000 tokens large corpus of historical texts was manually annotated and used as gold-standard dataset for automatic annotation of a larger collection of texts.

3. Source Data

We are using an existing collection of samples excerpted from Polish texts published between 1830 and 1918 for the purpose of researching historical inflection and spelling (Bilińska et al., 2016). In literary studies and historical linguistics the period represents the second half of the so called New Polish development stage of the language. The time span of the corpus marks important dates in Polish history having significant impact on social, cultural and political changes which subsequently influenced literary and linguistic developments. Especially year 1918 is considered a turning point in the history of the whole Central-European region and assumed to be the actual end of the 19th century.

The collection consists of 1000 samples of ca. 1000 words each, thus the whole collection is ca. 1 million words large. We will refer to it as F19-1M for short. The samples of the corpus are divided into five separate subcorpora of equal size representing the following functional styles: fiction, essays, science and popular science, short newspaper articles, drama. The samples were excerpted mostly from scans of original first editions of texts stored in digital libraries. They were carefully transliterated with regard to historical spelling rules, including misspelled words in the original editions.

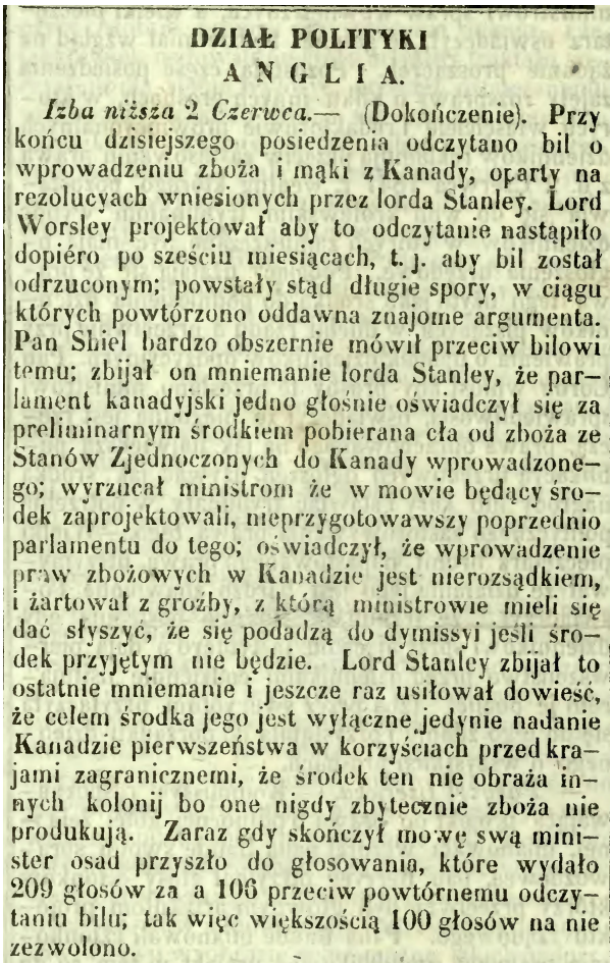
Samples in F19-1M are also evenly distributed between years: for every year and every stylistic subcorpus there is at least one and up to four samples, with an average of 11.24 (standard deviation 1.4) samples per year in the whole corpus. Every sample is accompanied by metadata and source files (scans). An example of source scan is shown in Figure 1.

Although the corpus represents all major Polish literary centres in all five stylistic subcorpora, a bias towards the capital city is significant as nearly 40% of sampled texts were published in Warsaw. Other major publishing centres are Lviv (16%), Cracow (12%), Poznań (7%) and Vilnius (5%), all inhabited by a dominant Polish speaking community at that time. However, the corpus also represents an important part of Polish literary activity performed in exile as about 6% of sampled texts were first published in Paris. In total, publications from over 70 different towns where included in the corpus.

4. Preprocessing

F19-1M is available as a collection of plain text files. Since our goal is to manually annotate only about half of it, we have decided to excerpt 3125 shorter samples of ca. 160 words each. This means that from each F19-1M sample we have extracted three smaller samples for our manually annotated corpus.

Before annotators can start their work, the samples need to be transcribed to modernised spelling and morphologically analysed to provide possible inflectional interpretations of each token. The processing described in this section takes place in Anotatornia web application (Woliński et al., 2017) which then serves the processed samples to annotators.



DZIAŁ POLITYKI
ANGLIA.

Izba niższa 2 Czerwca.— (Dokończenie). Przy końcu dzisiejszego posiedzenia odczytano bil o wprowadzeniu zboża i mąki z Kanady, oparty na rezolucyach wniesionych przez lorda Stanley. Lord Worsley projektował aby to odczytanie nastąpiło dopiéro po sześciu miesiącach, t. j. aby bil został odrzuconym; powstały stąd długie spory, w ciągu których powtórzono oddawna znajome argumenta. Pan Shiel bardzo obszernie mówił przeciw bilowi temu; zbijał on mniemanie lorda Stanley, że parlament kanadyjski jedno głośnie oświadczył się za preliminarnym środkiem pobierana cła od zboża ze Stanów Zjednoczonych do Kanady wprowadzonego; wyrzucał ministrom że w mowie będący środek zaprojektowali, nieprzygotowawszy poprzednio parlamentu do tego; oświadczył, że wprowadzenie praw zbożowych w Kanadzie jest nierozsądkiem, i żartował z groźby, z którą ministrowie mieli się dać słyszyć, że się podadzą do dymissyi jeśli środek przyjętym nie będzie. Lord Stanley zbijał to ostatnie mniemanie i jeszcze raz usiłował dowieść, że celem środka jego jest wyłączne jedynie nadanie Kanadzie pierwszeństwa w korzyściach przed krajami zagranicznemi, że środek ten nie obraża innych kolonij bo one nigdy zbytecznie zboża nie produkują. Zaraz gdy skończył mowę swą minister osad przyszło do głosowania, które wydało 209 głosów za a 106 przeciw powtórnemu odczytaniu bilu; tak więc większością 100 głosów na nie zezwolono.

Figure 1: A fragment of an original 1843 daily newspaper. The exact same fragment is annotated in Figure 2.

4.1. Transcription

Historical texts exhibit significant orthographic variation. For example, the word *komisja* ('commission') appears in F19-1M in the following spellings: *komisja, kommisja, komissja, kommissja, komisya, kommisya, komissya, kommissya, komisyja*. The variation can be coped with by means of transcription, which needs to be done automatically to avoid laborious and time-consuming manual editing. For the purpose of transcription, we use the converter created in the IMPACT project – a rule-based tool[1] (Kresa and Szafran, 2013) for substituting letters or sequences of letters based on the context in which they appear. The procedure itself is simple, but it requires building a relatively large set of rules which are created manually. The number of rules exceeds 3000 and carries not only matching patterns but also a list of exceptions for every rule. Fortunately, a large part of the rules created for the Baroque Corpus of Polish (Bronikowska et al., 2016) could be reused for transcription of 19th c. texts.

4.2. Morphological Analysis

Subsequently, the transcribed samples need to be processed by morphological analyser which provides possible morphological interpretations for text tokens. The annotators may choose one of them but are also allowed to provide their own interpretations in case none of the analyser's answers is correct or the token is unknown to the analyser. The obvious choice for morphological analysis is Morfeusz 2 (Woliński, 2014), the most widely used in Polish NLP and highly configurable analyser. It allows to customise all linguistically sensitive parts of the analysis: inflectional dictionary, tokenisation and tagset.

The linguistic basis of Morfeusz is *Grammatical Dictionary of Polish* (Saloni et al., 2015; Woliński and Kieraś, 2016), consisting of over 330,000 lexemes and nearly seven million word forms, which makes it the largest and most widely used inflectional data source for Polish. SGJP covers the whole list of entries taken from the largest general purpose dictionary of Polish (Doroszewski, 1958–1969) printed on paper in 11 volumes. Doroszewski's dictionary consists of 125,000 lexical entries including a substantial range of archaic, obsolete and dialectal words. Its extensive lexical basis goes back to even last decades of 18th century vocabulary which makes it a perfect lexical source for morphological analysis of 19th century texts. In fact, the extensive coverage of archaic vocabulary in SGJP is usually a curse when processing contemporary data, but in the case of 19th c. it is actually a blessing. Thus the morphological dictionary needed very few lexical additions to cover 19th century vocabulary of the corpus.

SGJP's linguistic data needed to be additionally modified ("aged") in order to cover regular inflectional phenomena non-existing in contemporary Polish but widespread in 19th c. texts. For example, plural instrumental forms of adjectives ending in *-emi* rather then *-ymi* (e.g. *wielkiemi* vs. the only possible contemporary form *wielkimi* 'large'); singular instrumental and locative forms of neuter gender ending in *-em* as opposed to masculine ending *-ym*; plural nominal and accusative forms of some nouns could take an *-a* ending (of Latin origin) instead of contemporary *-y* (*traktata* vs. *traktaty* 'treaties') etc.

Moreover, the analyser's segmentation rules needed to be adjusted to historical joint and disjoint spellings which were significantly different than contemporary ones.

Only 1.72% of tokens in our data did not receive any interpretation from the analyser. This rate is only marginally higher then in the case of contemporary analyser applied to contemporary texts, which proves that the modified analyser performs well.

4.3. Tagging

As described in Section 5, an automatic tagger is being used in the process of manual annotation to simulate one of the annotators simultaneously annotating each sample. Since no training data for tagging 19th c. or any other historical data set for Polish is available, we use a standard contemporary manually annotated corpus of Polish (Przepiórkowski et al., 2012) to train a stochastic tagger (Waszczuk, 2012). The data was only converted to comply to the tagset designed for 19th c. project (to the possible extent as not all grammatical phenomena annotated in the 19th c. project were also annotated in the contemporary corpus). The stochastic model therefore represents only a rough approx-

[1] https://bitbucket.org/jsbien/pol

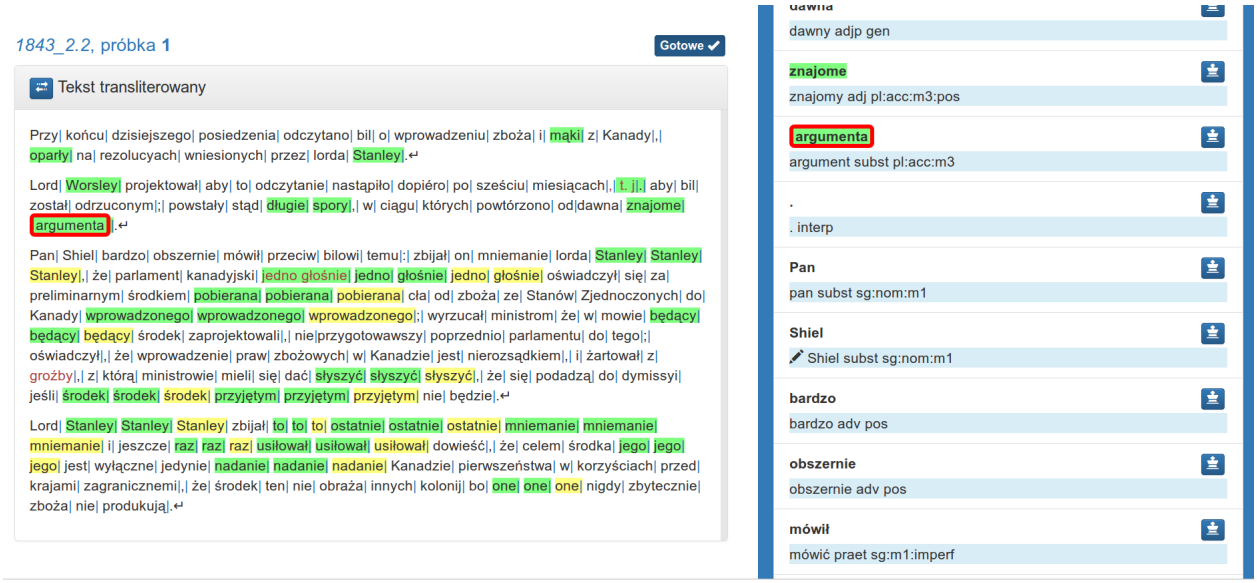


Figure 2: Anotatornia as seen by adjudicator reviewing conflicts between human annotator and tagger. Left hand part of the window shows a running text with annotation discrepancies highlighted. The right hand side shows a list of interpretations provided by an annotator and the tagger. Conflicts between the two are marked in green. The adjudicator can choose one of the two provided interpretations, choose her own from interpretations provided by the morphological analyser, or introduce her interpretation manually.

imation of 19th c. morphosyntax and is not expected to perform flawlessly, but it is expected to be able to handle standard grammatical phenomena such as case, gender and number agreements within phrases. This should be sufficient to pick up simple errors made by human annotators.

During the annotation process the tagger's model is periodically, incrementally retrained together with newly annotated data to improve its performance in the further course of annotation.

5. Annotation

The process of manual annotation of the corpus was conducted in a multi-access web application called Anotatornia (Woliński et al., 2017), which was developed to suite simultaneously two projects devoted to manual annotation of historical Polish text. The other project is the so called Baroque corpus (Kieraś et al., 2017). Thus Anotatornia is focused on satisfying the needs of historical data annotation. It operates on text represented in two layers: transliterated and transcribed.

The annotation is conducted in a hybrid mode conjoining manual work of a qualified linguist and automatic tagging followed by additional verification by human adjudicator ("super-annotator"). Each sample is automatically tagged, but the results of tagging are not disclosed to the annotator on any stage of the process. The annotator can only see possible interpretations provided by the morphological analyser and needs to choose one of them or create her own in case of misinterpreted or unknown tokens. The annotator's choices are then confronted with those made by the tagger and conflicting tokens are highlighted to the annotator, but only her own decisions are shown. This way the annotator is encouraged to check her work for simple mistakes but not tempted to switch to the tagger's version.

After this additional verification, an adjudicator steps in to revise and resolve any remaining conflicting decisions between human annotator and tagger (cf. fig. 2). Adjudicator's work consists mainly of choosing between two possibilities, but she is also allowed to introduce her own interpretation different from those selected by annotator and tagger. Although it is possible that no conflicting decisions between annotator and tagger occur and adjudicator's intervention would not be necessary, in practice every sample in the corpus was additionally reviewed by adjudicators, since the accuracy of the automatic tagger trained on the contemporary data does not exceed 90%.

The annotation process was conducted by a team of nine people, each of them specializing in Polish linguistics with either master's or doctoral degree in the field. Most of them have an extensive experience in various annotation tasks in previous projects. The four most experienced and most active annotators worked also as adjudicators. The possibility of adjudicating one's own conflicts was excluded. The annotators followed a detailed annotation manual. Specific problems were resolved using a mailing list.

Based on the final version of the annotated corpus, the annotation process as described above generates a 14.27% conflict rate between human annotator and tagger. As expected, a large majority of the conflicts are resolved in favour of human annotators (87.22%) but a significant number of human errors are also found and corrected as the remaining 12.78% have been either resolved in favour of the tagger (6.69%) or changed to an alternative interpretation provided by the adjudicator (6.09%).

As a result of the annotation process, 2944 samples were annotated by one human annotator and tagger. Each sample required additional verification by the adjudicator as the situation of full agreement between human annotator and tagger hasn't occurred even once. The number of conflicts ranged between 8 and 152 (large number of conflicts usually involved serious segmentation problems) with an average of 32.34 per sample (median 30). Thus, in the hybrid annotation mode presented above adjudicator's workload is significantly higher comparing to the standard annotating situation with two human annotators followed by adjudicator. On the other hand, time and financial cost of the whole annotation process performed in the hybrid mode drops radically nearly to the level of single annotator mode without the necessity of complete abandoning of any additional quality control. We believe that the hybrid annotation proved to be useful and would apply the same strategy in future projects.
The total number of 625,000 tokens were annotated in the project.

6. Tagset

The tagset of the presented corpus generally reflects ideas behind the tagset of the National Corpus of Polish (NCP) as well as the one used by Morfeusz morphological analyser. The two are similar, yet not identical. The crucial difference between them concerns the grammatical gender. The 19th century tagset basically follows Morfeusz's tagset, however some minor differences were introduced motivated mainly by the Baroque tagset and the desire for basic coherence of the two historical tagsets.
In Polish historical linguistics it is assumed that the main morphosyntactic processes are over by 1830 and Polish morphosyntax of the presented period is basically the same as the contemporary one. The linguistic differences are reflected in phonetics, vocabulary, surface morphology and word order but they do not affect the morphosyntactic tagset. However, some useful extensions of the contemporary tagset were introduced to ease some corpus linguistic research. The 19th century tagset as well as the Baroque one marks the auxiliary verbs of pluperfect tense and future imperfective tense assigning different tags to past and future forms of BYĆ ('to be') verb in constructions of those tenses than in the case of other syntactic constructions (such as passive voice). This will allow to track the decline of pluperfect tense in Polish from early 17th to early 20th century as well as the variability of future imperfective tense construction according to word order (*będę robić* vs. *robić będę* 'I will do') and the use of either infinitive or past verb forms (*będę robić* vs. *będę robił* 'I will do').
Another minor difference between both historical tagsets and the contemporary one is the introduction of numeral adjectives and adverbs as separate parts of speech. Grammatically, numeral adjectives and adverbs share the same features as regular adjectives and adverbs and thus they are not distinguished in the contemporary data, but were introduced to comply to the traditional diachronic description of Polish.
Regardless of those similarities between historical tagsets and their differences comparing to the contemporary one, the 19th century tagset far more resembles the latter. The Baroque tagset needs to cover a much longer time span and the range of morphosyntactic phenomena involved is much more extensive, therefore it is significantly more complex.

	19c 625k	**NKJP 1.2M**
CRF	90.48%	91.44%
bi-LSTM	93.38%	95.28%

Table 1: 10-fold cross-validation accuracy results for two taggers based on 19th and contemporary training data.

7. Usage

Manually annotated resources typically serve as training data for machine learning applications. In the case of the presented corpus, so far two stochastic taggers representing different methodologies, namely conditional random fields (Waszczuk, 2012) and bi-LSTM neural networks (Krasnowska-Kieraś, 2017), were trained and evaluated based on the historical data. As shown in Table 1, the taggers obtained slightly lower results comparing to their evaluation on the manually annotated 1.2 mln subcorpus of the National Corpus of Polish which is commonly used as a gold-standard dataset for contemporary Polish. The results however are not fully comparable since both corpora were annotated with slightly different tagsets. The size of the training data is also relevant to the results as the 19th c. corpus is twice smaller than the manual subcorpus of NCP.
Another NLP tool scheduled to be built in near future using a manually annotated corpus is an automatic transcriber not relying on manually crafted rules. Our rule-based transcriber has yielded good results, but the set of rules became relatively large, which makes them hard to manage and causes some efficiency difficulties. Since corpus annotators were also required to correct transcription errors, the resulting corpus can serve as training data for a machine learning application which can possibly obtain better or at least as good results as the rule-based transcriber, while being more computationally efficient.

8. Conclusions and Future Work

The current project does not cover a task of building a large, automatically annotated corpus of 1830-1918 period or any other large corpora, however the corpus described in this article as well as tools used for its annotation provide a sufficient technical environment for building an extensive corpus of 19th c. Polish. Although strictly philological problems such as text acquisition, balancedness etc. are still open to the future creators of the corpus, the technical leg of the potential project is ready to use. A proof of concept of such corpus will be built based on the tools mentioned above and publicly available collections of historical texts such as Wikisource.[2]
The final version of the manually annotated corpus is publicly available both as a collection of TEI XML files and as a searchable web-based resource.[3] The later is based on MTAS search engine (Brouwer et al., 2017), which allows

[2] https://pl.wikisource.org/
[3] http://korpus19.nlp.ipipan.waw.pl

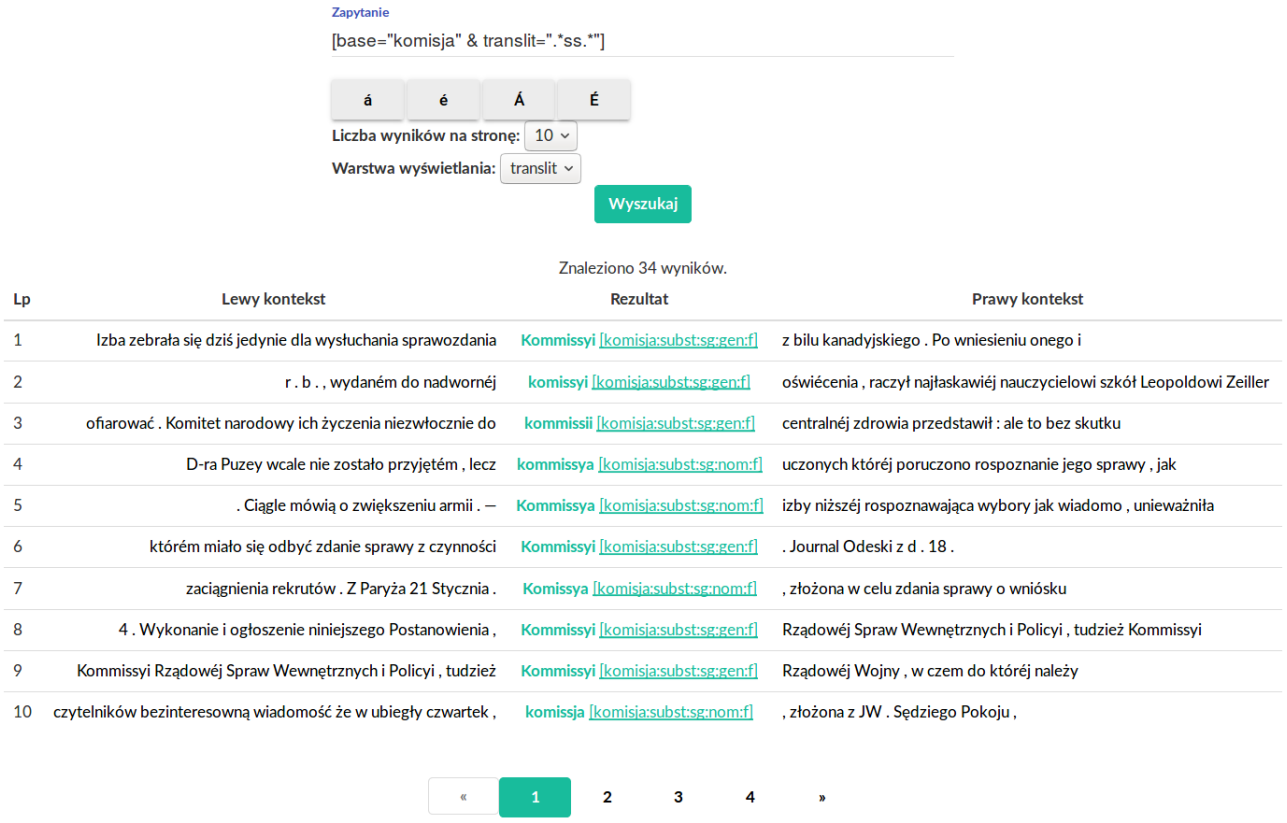


Figure 3: A corpus search query in which all occurrences of a noun *komisja* 'commission' are found with restriction only to those spelled with double *s* in original transliterated document.

to refer to both transcription and transliteration text layers of the corpus, as well as to morphosyntactic annotation layer. Conditions referring to various layers may be combined in one corpus query, so for example all occurrences of a word lemmatized as *komisja* 'commision' restricted to only those spelled originally with double *s* can be found. The query together with several first hits can be seen in Figure 3.

9. Acknowledgements

The work being reported was financed by a National Science Centre, Poland grant DEC-2014/15/B/HS2/03119.

10. Bibliographical References

Bilińska, J., Derwojedowa, M., Kieraś, W., and Kwiecień, M. (2016). Mikrokorpus polszczyzny 1830-1918. *Komunikacja specjalistyczna*, 11:149–161.

Bronikowska, R., Gruszczyński, W., Ogrodniczuk, M., and Woliński, M. (2016). The Use of Electronic Historical Dictionary Data in Corpus Design. *Studies in Polish Linguistics*, 11(2):47–56.

Brouwer, M., Brugman, H., and Kemps-Snijders, M. (2017). MTAS: A Solr/Lucene based Multi Tier Annotation Search solution. In *Selected papers from the CLARIN Annual Conference 2016, Aix-en-Provence, 26–28 October 2016, CLARIN Common Language Resources and Technology Infrastructure*, number 136, pages 19–37. Linköping University Electronic Press, Linköpings universitet.

Doroszewski, W. editor. (1958–1969). *Słownik języka polskiego PAN*. Wiedza Powszechna – PWN.

Erjavec, T. (2012). The goo300k corpus of historical slovene. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Erjavec, T. (2015). The imp historical slovene language resources. *Lang. Resour. Eval.*, 49(3):753–775, September.

Kieraś, W., Komosińska, D., Modrzejewski, E., and Woliński, M. (2017). Morphosyntactic Annotation of Historical Texts. The Making of the Baroque Corpus of Polish. In *International Conference on Text, Speech, and Dialogue*, pages 308–316. Springer, Cham.

Krasnowska-Kieraś, K. (2017). Morphosyntactic disambiguation for Polish with bi-LSTM neural networks. In *Proceedings of 8th Language & Technology Conference*, pages 367–371. Poznań, Poland.

Kresa, M. and Szafran, K. (2013). Przykład nowego zastosowania słownika polszczyzny historycznej. *Prace Filologiczne*, LXIV:159–171.

Przepiórkowski, A., Bańko, M., Górski, R.L., and Lewandowska-Tomaszczyk, B., editors. (2012). *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Saloni, Z., Gruszczyński, W., Woliński, M., Wołosz, R.,

and Skowrońska, D. (2015). *Słownik gramatyczny języka polskiego*. 3. edition.

Waszczuk, J. (2012). Harnessing the CRF complexity with domain-specific constraints. The case of morphosyntactic tagging of a highly inflected language. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, Mumbai, India.

Woliński, M. and Kieraś, W. (2016). The on-line version of Grammatical Dictionary of Polish. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation, LREC 2016*, pages 2589–2594, Portorož, Slovenia. ELRA, European Language Resources Association (ELRA).

Woliński, M., Kieraś, W., and Komosińska, D. (2017). Anotatornia 2 – an annotation tool geared towards historical corpora. In *Proceedings of 8th Language & Technology Conference*. Poznań, Poland. under review.

Woliński, M. (2014). Morfeusz reloaded. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 1106–1111, Reykjavík, Iceland. ELRA.

11. Language Resource References

Magdalena Derwojedowa. (2017). *Corpus of Polish Texts of 1830-1918 period*. University of Warsaw, `http://www.f19.uw.edu.pl`.

ZIL IPI PAN. (2014). *Morfeusz 2*. `http://sgjp.pl/morfeusz`, version 2.0.

Zygmunt Saloni and Marcin Woliński and Robert Wołosz and Włodzimierz Gruszczyński and Danuta Skowrońska. (2015). *Grammatical Dictionary of Polish*. `http://sgjp.pl`, version 3.

Evaluating Inflectional Complexity Crosslinguistically: a Processing Perspective

Claudia Marzi[1], **Marcello Ferro**[1], **Ouafae Nahli**[1], **Patrizia Belik**[2], **Stavros Bompolas**[3], **Vito Pirrelli**[1]
[1]Institute for Computational Linguistics-CNR, [2]Universitat Politècnica de Valencia, [3]University of Patras
patrizia_belik@yahoo.com, stavros.bompolas@gmail.com
{claudia.marzi, marcello.ferro, ouafae.nahli, vito.pirrelli}@ilc.cnr.it

Abstract

The paper provides a cognitively motivated method for evaluating the inflectional complexity of a language, based on a sample of "raw" inflected word forms processed and learned by a recurrent self-organising neural network with fixed parameter setting. Training items contain no information about either morphological content or structure. This makes the proposed method independent of both meta-linguistic issues (e.g. format and expressive power of descriptive rules, manual or automated segmentation of input forms, number of inflectional classes etc.) and language-specific typological aspects (e.g. word-based, stem-based or template-based morphology). Results are illustrated by contrasting Arabic, English, German, Greek, Italian and Spanish.

Keywords: paradigm-based morphology, inflectional complexity, prediction-based processing, recurrent self-organising networks.

1. Introduction

There is little doubt that some languages are inflectionally more complex than others. Everybody would agree with the intuitive statement that the English conjugation system is simpler than the German system, and that the latter is, in turn, simpler than the verb system of Modern Standard Arabic. However, the naïve view is faced with two apparent paradoxes. When linguists try to pinpoint the source of this complexity, the task is far more elusive than expected, and goes well beyond a purely descriptive notion of diversity in the battery of realisational means (e.g. number of different affixes, number of cells in the corresponding paradigms, amount of stem allormophy etc.) provided by each system. Besides, there seems to be a poor correlation between our intuitive notion of morphological complexity and actual evidence of the pace of acquisition of more or less complex inflectional systems in child language. In some cases, apparently simpler inflectional markers may take more time to be acquired than formally more complex and articulated ones. What looks like a prohibitively difficult learning task in the light of the complexity and uncertainty of the inference steps required for mastering it, may turn out to be relatively unproblematic for human speakers. In the present paper we entertain a usage-oriented, cognitively motivated approach to issues of morphological complexity, based on a neurobiologically inspired model of word processing and learning, and explore its theoretical and computational implications.

2. Background

Assessing and understanding the comparative complexity of the inflectional system of a language relative to a functionally-equivalent system of another language remains an open question, which has animated much of the contemporary debate on the nature of word knowledge and its connection with issues of word usage (Ackerman and Malouf, 2013; Bane, 2008; Bearman et al., 2015; Juola, 1998; Moscoso del Prado Martín et al., 2004). In a crosslinguistic perspective, the way morphosyntactic features are contextually realised through processes of word inflection probably represents the widest dimension of grammatical crosslinguistic variation, somewhat belittling universal invariances along other dimensions (Evans and Levinson, 2009).

Descriptive linguists have often approached the issue of comparative inflectional complexity by providing comprehensive catalogues of the morphological markers and patterns in a given language or languages (Bickel and Nichols, 2005; McWorther, 2001; Shosted, 2006). Accordingly, the complexity of an inflectional system is measured by simply enumerating the number of category values instantiated in the system (e.g. person, number or tense features) and the range of available markers for their realisation: the bigger the number, the more difficult the resulting system. The notion of Enumerative Complexity (or E-complexity) is however dubious (Ackerman and Malouf, 2013). Suppose we have two hypothetical inflectional systems, each with two categories only (say, singular and plural) and three different endings for each category: A, B, C for singular, and D, E and F for plural. In the first system, paradigms are found to present three possible pairs of endings only: <A, D>, <B, E>, <C, F> (corresponding to three different inflection classes). In the second system, any combination is attested. Clearly, the latter system would be more difficult to learn than the former, as it makes it harder to infer the plural form of a word from its singular form. Nonetheless, both systems present the same degree of E-complexity.

Of late, less combinatorial approaches to morphological description have played down the role of E-complexity in inflection. These approaches, generally referred to as "paradigm-based", or "word-based", or "abstractive" grammatical frameworks, examine the systemic organisation of underlying patterns of surface variation, to conceive of an inflectional system as a network of implicative relations holding between fully-inflected forms (Blevins, 2003; Blevins, 2016; Burzio, 1998; Bybee, 1995; Bybee and McClelland, 2005; Matthews, 1991; Pirrelli and Battista, 2000). Implicative relations allow novel forms to be pre-

dicted and inferred on the basis of known forms, thereby making it easier for a human speaker to process, retain and access them. Not only do implicative relations shed light on the way children come to master the inflectional system of their mother tongue, but they also constrain systems of word shapes, providing a limit on the range of E-complexity that languages can afford.

A number of information theoretic approaches have been proposed to model this view in terms of Kolmogorov complexity (Kolmogorov, 1965) and Shannon entropy (Shannon, 1948). The idea behind Kolmogorov complexity is to measure a dataset of inflected forms as the shortest possible grammar needed to describe them. This however leads to a definition of morphological complexity heavily dependent on the grammar formalism adopted (Bane, 2008; Walther and Sagot, 2011). Ackerman and Malouf (2013) use Shannon's information entropy to quantify prediction of an inflected form as a paradigm-based change in the speaker's uncertainty. They conjecture that inflectional systems tend to minimise the average conditional entropy of predicting each form in a paradigm on the basis of any other form of the same paradigm (Low Conditional Entropy Conjecture or LCEC). This is measured by looking at the distribution of inflectional markers across inflection classes in the morphological system of a language. Although LCEC proves to be able to capture a substantial part of the inferential complexity within paradigms, it presupposes a segmentation of inflected forms into stems and affixes, while ignoring implicative relations holding between stem allomorphs. Use of principal parts can remedy this in a principled way (Finkel and Stump, 2007, among others). However, while entropy measures can provide extremely valuable insights into the organisation of static, synchronic paradigms, there are crucial complementary questions about how such patterns are processed and learned which remain unaddressed. In what follows, we will focus on these important issues from a neuro-computational perspective. In particular, we are interested in evaluating the net effect of the complexity of an inflectional system on the processing behaviour of a recurrent neural network, excluding the role of word token frequency effects on prediction-driven processing (Pickering and Garrod, 2013). To factor out frequency effects, we ran simulations on uniformly distributed inflectional data. Our work can hence be understood as a purely morphological evaluation of complexity, based on lexical rather than corpus data. Since uniform distributions increase the entropy of a system, our results define some sort of upper bounds for inflectional complexity: if all factors (including frequency) are taken into account, the effects we observe here will likely be more prone to potentially confounding factors. This is in the spirit of information-theoretic work on paradigm-based morphology, as well as 'discriminative learning' research in animal behaviour and language learning (Rescorla and Wagner, 1972; Ramscar and Yarlett, 2007; Ramscar and Dye, 2011), and justifies our choice of a specific type of recurrent neural network, namely a Temporal Self-Organising Map (Ferro et al., 2011; Marzi and Pirrelli, 2015; Pirrelli et al., 2015; Marzi et al., 2016) as a workbench for simulating pardigm-based effects. Ultimately, it is intended to bridge the gap between an algorithmic/mathematical understanding of processing-based morphological complexity (Balling and Baayen, 2008; Balling and Baayen, 2012), and the neurobiological (or implementational) level of Marr's hierarchy (Marr, 1982).

3. Method and data

According to Dressler and colleagues (Bittner et al., 2003), European languages can be arranged along an inflectional complexity continuum, ranging from a more inflecting-fusional type (left) to a more isolating type (right):

Lithuanian→Greek→Russian→Croatian→Italian→ Spanish→German→Dutch→French→English.

Somewhat paradoxically, developmental evidence provides an indication that inflectional contrasts in prototypically inflecting verb systems are reported to be acquired at an earlier stage than inflectional contrasts in more isolating verb systems.[1]

Here, we would like to investigate the related question about how degrees of inflectional complexity/regularity affect word processing strategies. For this purpose, we analyse the performance of recurrent self-organising neural networks learning a few languages in the typological continuum above: namely, English, German, Greek, Italian and Spanish. To broaden our typological data, Standard Modern Arabic was added to the range of tested languages. For each language we sampled the 50 top-frequency verb paradigms found in a few reference resources: CELEX (Baayen et al., 1995) for German and English; the Paisà Corpus (Lyding et al., 2014) for Italian; the European Spanish Subcorpus of the Spanish TenTen Corpus (www.sketchengine.co.uk); the SUBTLEX-GR corpus (Dimitropoulou et al., 2010) for Modern Greek; the Penn Arabic Treebank (Maamouri et al., 2004). To control paradigm implicative relations, we selected a comparable set of 15 paradigm cells (14 cells for Arabic).[2] The sample contains a shared set of 6 present and 6 past tense forms for English, German, Greek, Italian and Spanish. Infinitive, gerund/present participle and past participle forms were added for English, German, Italian and Spanish, whereas 3 singular forms of the simple future were included for Modern Greek. The Arabic set contains 7 imperfective and 7 perfective forms, including 1S, 2MS, 3MS, 3FS, 1P, 2MP, 3MP cells. Only inflected "raw" forms from the selected cells were included for training a recurrent neural network, with no additional morphological information. Each language-specific dataset is administered to a Temporal Self-Organising Map (hereafter *TSOM,* see section 3.2. for more details) for 100 epochs. In one epoch, all word forms are randomly input to the map five times, and each training session was repetated five times with results averaged over repetitions to control random variabil-

[1] For example, Noccetti (2003) reports that the transition from pre- to proto-morphology in Italian verb acquisition has an early onset at Brown's stage II, with mean length of utterance 2 (Brown, 1973), in contrast with the comparative late emergence of the third-person singular marker *-s* in the acquisition of the English present tense.

[2] The full set of data, for each language, is available at http://www.comphyslab.it/redirect/?id=lrec2018_data

ity. TSOM parameters are identically initialised across the 6 languages, with the only exception of available memory nodes (Table 1).[3]

3.1. The data

The selected paradigm cells for the target languages offer evidence of graded levels of morphological (ir-)regularities. Greek, Italian, Spanish and German present highly inflecting conjugation systems, with extensive stem allomorphy, exhibiting varying degrees of (ir)regularity. Inflecting processes include prefixation, suffixation, vowel alternation, infixation and suppletion. Arabic stem formation is based on the interspersion of discontinuous consonantal roots and variable vowel patterns. English offers the by far simplest inflectional system, with extensive syncretism and a rather dichotomous subdivision of paradigms between regular and irregular ones.

For all test languages except Modern Greek, word forms are orthographically transcribed, and administered to the network one symbol at a time as raw letter strings (starting with the start-of-word symbol *'#'* and ending with the end-of-word symbol *'$'*), with no information about their morphological structure. To account for the complex interaction between morphologically-conditioned and phonologically-conditioned stem allomorphy in Greek conjugation (Ralli, 2005; Ralli, 2006), Greek word forms are transcribed phonologically, and input one segment at a time. Once more, no information about morphological structure is input. To assess the network sensitivity to morphological structure and the processing behaviour of the map across morpheme boundaries (see section 4.), after training, word forms in all test languages were segmented morphologically according to a prefix-stem-suffix schema: e.g. Greek *e-krin-a* 'I judged', German *ge-dach-t* 'thought' (past participle), Arabic *ya-ktub-u* 'he writes'. Stem allomorphs within a single paradigm (whether morphologically/phonologically predictable or not) are segmented as whole units, with no explicit indication of either the root or the alternating pattern: e.g. Arabic *katab-a* 'he wrote' vs. *ya-ktub-u* 'he writes'. Only purely suffixal stem formation is segmented: e.g. Greek *aɣapi-s-a* 'I loved', Italian *perd-ut-o* 'lost' (past participle).

3.2. Recurrent self-organising neural networks

TSOMs are recurrent self-organising networks consisting of two-dimensional grids of artificial memory/processing nodes that learn to dynamically memorise input strings as chains of maximally-responding processing nodes (Best Matching Units, or BMUs), whose level of sensitivity to input symbols in context is a continuous function of their distributional regularities in training (Ferro et al., 2011; Marzi and Pirrelli, 2015; Pirrelli et al., 2015; Marzi et al., 2016). In a TSOM, each processing node has two layers of synaptic connectivity: an input layer, connecting each node to the current input stimulus (i.e. orthographic or phonological symbols), and a (re-entrant) temporal layer, connecting each node to all other nodes. Every time a symbol is presented to the input layer, activation propagates to all map nodes through input and temporal connections, and the most highly activated node (BMU) is calculated (see Figure 1). Given the BMU at time *t*, the temporal layer encodes the expectation of the current BMU for the node to be activated at time *t+1*. The strength of the connection between consecutively activated BMUs is trained through the following principles of discriminative learning: given the input bigram *ab*, the connection strength between the BMU that get mostly activated for *a* at time *t* and the BMU for *b* at time *t+1* will:

(i) increase if *a* often precedes *b* in training (entrenchment),
(ii) decrease if *b* is often preceded by a symbol other than *a* (competition).

The complex interaction between entrenchment and competition in a TSOM accounts for important dynamic effects of self-organisation of stored words (Marzi et al., 2014; Marzi et al., 2016). In particular, at a sublexical level, systematically recurrent patterns tend to recruit context-sensitive specialised (and stronger) chains of BMUs. If the bigram *'ab'* is repeatedly input to the TSOM, the map tends to develop a specialised BMU('*b*') for '*b*' in '*ab*' and a highly-weighted outward connection from BMU('*a*') to BMU('*b*'), reflecting a strong expectation of BMU('*a*') for a prospective BMU('*b*'). In detail, during training, weights on both connectivity layers are adjusted in an experience-dependent fashion: after an initial period of random variability, where nodes activate chaotically, a map gradually develops more and more specialised sequence of BMUs for word forms - or sub-lexical chains - that are functionally dependent on the frequency distribution and the amount of formal redundancy in the training data. On the one hand, specialised inter-node connectivity makes BMUs less confusable and more salient, as they receive stronger support through temporal connections than any other node. On the other hand, less specialised and more blended BMUs are densely and less strongly connected with many others, to meet the input of more words. When a TSOM is trained

language	form length min/max	paradigms reg./irreg.	word types/ training size	TSOM nodes
Arabic	4/11	18/28	560/601	40x40
English	2/11	20/30	208/750	35x35
German	3/11	16/34	504/750	40x40
Greek	2/13	37/13	744/750	42x42
Italian	2/12	23/27	748/750	42x42
Spanish	2/15	23/27	715/750	40x40

Table 1: Language training sets. Form length is measured by the number of orthographic/phonetic symbols. In the Italian sample, we encoded the orthographic accent as a separate character (e.g. *è* = e').

[3]For the sake of data comparability, the number of memory nodes for each language was decided empirically to control for cross-linguistic differences in cardinality and length of word types (see Table 1). For all trained languages, the percentage of used nodes among all available nodes ranges between 31% and 35%.

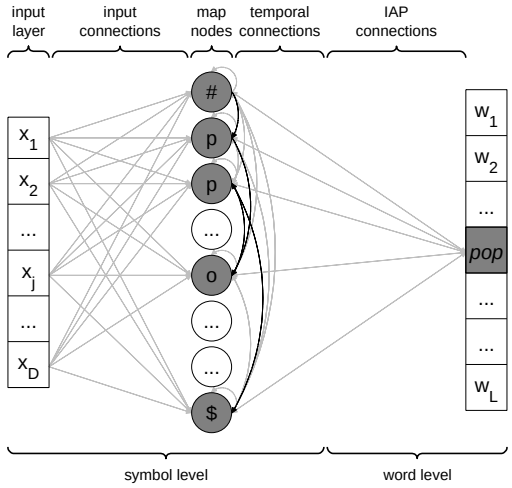


Figure 1: Functional architecture of a Temporal Self-Organising Map (TSOM). Each input word form is presented by a unique time-series of symbols, which are administered one at a time.

on higly redundant input data such as verb paradigms, specialisation and blending may interact. By inputting all verb forms with a uniform token distribution, we factor out the effect of frequency and focus our analysis on the effect of formal redundancy only. Thus, due to the prediction-driven bias of the temporal layer of re-entrant connections, strong expectations over upcoming input symbols account for successful serial word processing, with processing accuracy being a function of how confident the TSOM is about the position of the current symbol in the input string.

These dynamics make it possible to test the behaviour of a TSOM on specific lexical tasks: *word recall* and *serial word processing*. For each time series of input symbols (i.e. each word form), the processing response of the map is represented by the synchronic activation pattern of all the BMUs that most highly get activated for that input sequence. Thus, the task of *word recall* tests how accurately a map can retrieve the input word from its synchronic activation pattern, namely how accurately the activation nodes of the map can encode information about the timing of the input symbols that make up the word. Accuracy in recall verifies that, for each input form, activation propagation (i.e. sequential activation) of nodes within each synchronic pattern correctly activates the BMUs associated with the symbols of each word. Scores are given in Table 2, showing very high accuracy and remarkably cross-linguistic similarity.

Conversely, *serial word processing* can be monitored by evaluating the ability of a map to predict an incrementally presented input word. Proceduraly, by presenting one symbol at a time on the input layer, a TSOM is prompted to complete the current input string by anticipating the upcoming BMU to be activated. Anticipation/prediction scores across input words are calculated by incrementally assigning each correctly anticipated symbol in the input form a 1-point score, i.e. the anticipation score of the preceding symbol incremented by 1. Otherwise, for unpredicted symbols the score is 0. The more input symbols are anticipated, the easier the prediction of that word. Results will be given and discuss in the ensuing section (4.).

language	recall %	sd %
Arabic	99.93	0.16
English	99.62	0.86
German	99.76	0.18
Greek	99.84	0.06
Italian	99.79	0.15
Spanish	99.94	0.13

Table 2: For each language, percentage values of correctly recalled word types and standard deviations are given, averaged over 5 map instances.

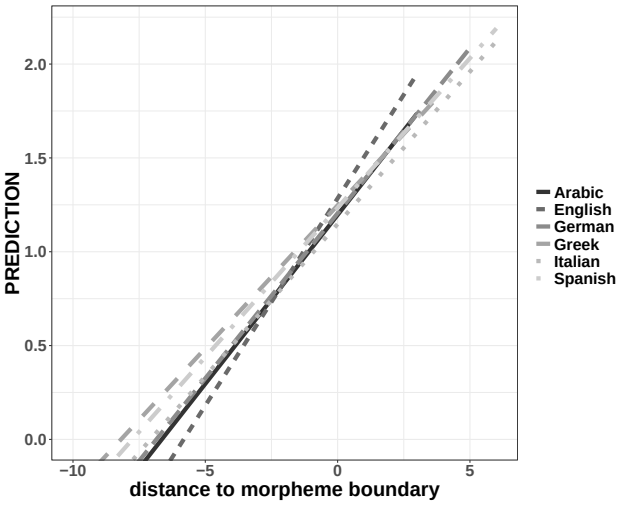


Figure 2: Marginal plot of interaction effects between language and distance to morpheme boundary, in an LMER model fitting the number of symbols predicted by TOMs. Fixed effects: languages, distance to morpheme boundary. Random effects: TSOM instances, paradigms, word forms.

4. Results and discussion

All results in this section are analysed with linear models for mixed effects (LMERs). For all models/languages, we treated TSOM instances, verb paradigms and word forms as random effects. In particular, we show how inflectional systems of different complexity (independent variables) affect TSOM processing, by focusing on symbol prediction rate as a dependent variable.

Figure 2 plots, for each language, the rate of symbol prediction in serial word processing. It should be appreciated that Arabic, German, Italian and Spanish exhibit remarkably similar trends, with not significantly different slopes (p-values >.05). Only Greek and English present significantly different slopes (p-values <.001), with Greek forms being the hardest to process (lower slope), and English forms the easiest ones (higher slope).

To evaluate the impact of formal transparency on processing, the effect of regularity is fitted in a second LMER model where languages are considered as random effects. Across our selected languages, verb forms in regular paradigms are systematically more predictable (p-value

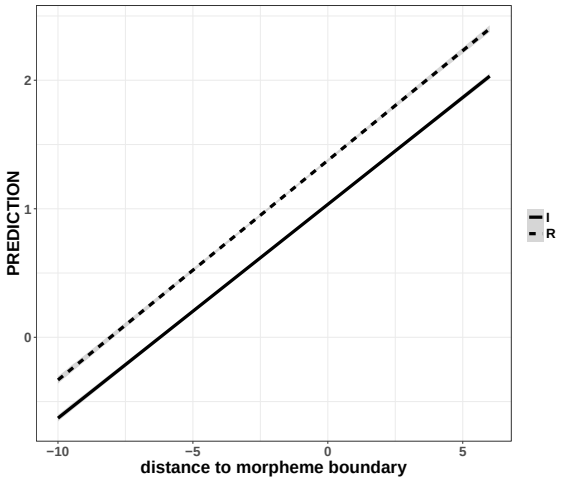


Figure 3: Marginal plot of interaction effects between categorical (ir-)regularity and distance to morpheme boundary, in an LMER model fitting the number of symbols predicted by TOMs. Fixed effects: irregulars (I) vs. regulars (R), distance to morpheme boundary. Random effects: languages, TSOM instances, paradigms, word forms.

<.001) than forms in irregular ones, as shown by the marginal plot in Figure 3.

To investigate in more detail the impact of inflectional complexity on processing, we fitted an LMER of symbol prediction for each language, with classes of morphological regularity (regulars vs. irregulars) and morphological structure (stem vs. suffix) as fixed effects (Figure 4).

The marginal plots in Figure 4 better show a clear serial processing effect of the distance of an input symbol to the stem-ending boundary, over and above the length of the input string. Unsurprisingly, Italian and Spanish show a very similar behaviour, with irregular forms exhibiting fusional effects that blur the boundary between stem and inflectional endings, and comparable (but not identical) number of stem allomorphs (Boyé and Cabredo Hofherr, 2006; Pirrelli, 2000). Remarkably, both German and Greek exhibit systematic (albeit not always predictable) processes of stem formation, followed by a fairly homogenous pool of inflectional endings. As a result, in both languages, the base stem (or present stem) is often followed by a highly embedded and unpredictable sequence of symbols which account for the negative slopes in the corresponding segments. In Arabic imperfective forms, prefixation is used to convey person features. This makes selection of inflectional endings fairly predictable, given the stem. Finally, in our pool of languages, English offers the by far simplest inflectional system, with extensive syncretism and a rather dichotomous subdivision of paradigms between regular and irregular ones.

Slopes are also modulated by degrees of regularity/transparency of the stem. Discontinuous patterns of morphological structure are often found in irregular paradigms of concatenative languages (e.g. English *drink/drunk*, German *finden/fanden*), and are systematically attested in non-concatenative morphologies (e.g. Arabic *kataba/yaktubu*). It is well known in the literature on se-

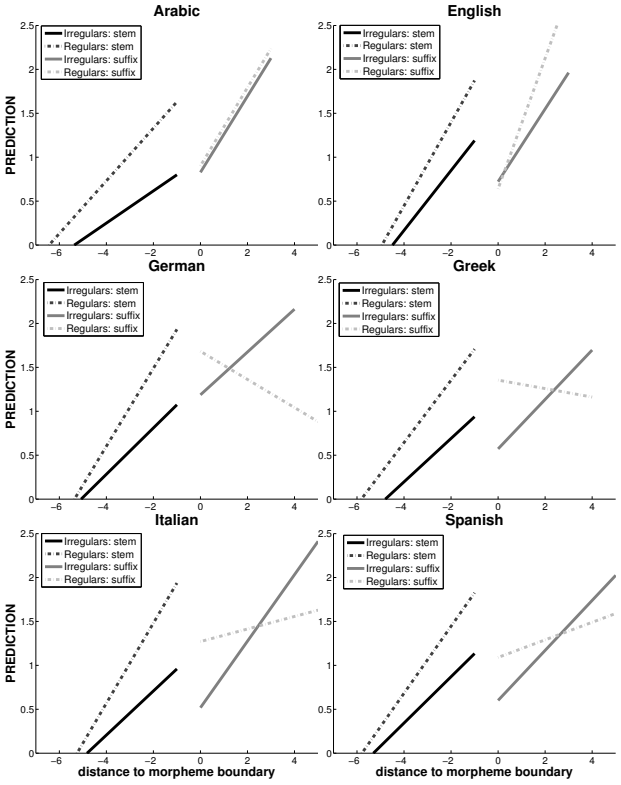


Figure 4: For each language, marginal plots of interaction effects between morphological (ir-)regularity and distance to morpheme boundary, in LMER models fitting the number of symbols predicted by TOMs for stem and suffix. Fixed effects: regularity (dashed lines) vs. irregularity (solid lines), distance to morpheme boundary, stem and suffix as separate patterns, suffix length. Random effects: TSOM instances, paradigms, word forms.

rial alignment that discontinuous patterns are more difficult to be processed and tracked down (Hahn and Bailey, 2005). In the present context, stem allomorphs are less predictable since their "uniqueness point", i.e. the point at which they can be distinguished from all neighbouring allomorphs, are normally delayed, slowing down processing (Balling and Baayen, 2012). Other things being equal,[4] the order of magnitude of this competing effect is a function of the number of stem allomorphs: the more they are, the more confusable the input stem is. Conversely, in regular paradigms the same stem shows up systematically in all cells. Hence the stem suffers from no intra-paradigmatic competition. These factors provide, on average, a net processing advantage of stems in regular paradigms, as confirmed by the significant difference in prediction rate between stems of regular vs. irregular paradigms in all languages (Figure 4). However, the clear advantage in stem processing is somewhat compensated by the difference in the prediction rate on suffixes. In German, Greek, Italian and Spanish suffixes in irregulars are predicted significantly more easily than suffixes in regular forms, as shown by the steeper segments in the positive

[4]The effect is modulated by other factors we are not controlling here: i.e. the formal similarity between the input stem and its intra-paradigmatic competitors, the entropy of the paradigm, the lexical neighbourhood of the word form.

x range of Figure 4. Besides, for all languages, there is a deeper drop in prediction rate at the stem-suffix boundary (for $x = 0$ as the first symbol of the suffix) in regular forms. In fact, stem allomorphs typically select only a subset of paradigm cells. Hence they can be followed by fewer inflectional endings than regular stems are. This reduces processing uncertainty, by constraining the range of possible continuations at the stem-suffix boundary of irregularly inflected forms. As a result, irregulars tend to blur the TSOM sensitivity to the verb morphological structure, favouring a somewhat more holistic processing strategy.

Results and statistical significance are confirmed when we consider a more fine-grained meausure for inflectional complexity based on a gradient of morphological regularity, which takes into account the number of stem alternants of a given paradigm.[5] It represents a graded - and continuous - meausure of paradigmatic (ir-)regularity that considers, for each inflected form, the number of stem-sharing forms (or stem family size), instead of a dichotomous and formal classification of paradigms (regulars vs. irregulars). Thus, given the number of inflected form-types for each paradigms, the average stem family size correlates better with the non-categorical idea of inflectional complexity.

5. Concluding remarks

Our evidence is in line with Low Conditional Entropy Conjecture (Ackerman and Malouf, 2013). The processing cost of considerably different inflectional systems appears to oscillate within a fairly limited range of variation, whose upper bound and lower bound are marked, in our language sample, by Modern Greek and English respectively. All other conjugations present no statistically significant differences in the processing overhead they require, in spite of their typological diversity, which is nonetheless reflected by the different processing profiles exhibited by sublexical constituents in the different languages.

In a functional perspective, this evidence can be interpreted as the result of a balancing act between two potentially competing communicative requirements: (i) a recognition-driven tendency for a maximally contrastive system; and (ii) a production-driven bias for a maximally generalisable inflection system, where, for each paradigm, all forms in the paradigm can possibly be deduced from any one of its forms.

This interpretation is also compatible with another clear pattern shown by our data. In each of our sample languages, the difference between the processing cost of forms in irregular paradigms compared with the processing cost of forms in irregular paradigms shows an interesting structure-sensitive profile. The higher processing cost of irregular stems is compensated by a lower cost in processing the inflectional endings selected by irregular stems. Once more, these structural effects tend to reduce processing costs at the level of the whole word, making the inflectional system as functional as possible from an information theoretic perspective. In recognising that scale effects play an important role in the processing behaviour of our model at the word level, and that constrains on word processing are likely to obtain universally, we also highlight the fundamental communicative role of words as optimal-sized units for describing general functional tendencies in language, and for studying language as a complex information system.

Inflectional complexity is multifactorial and dynamic. Its variability can be observed and measured on many counts: number and types of stem allomorphs, number and types of inflectional affixes, transparency/compositionality effects, stem-stem predictability, stem-affix predictability, affix-affix predictability, intra-paradigmatic and inter-paradigmatic frequency distributions etc. In this paper, we investigated inflectional complexity by controlling a number of interacting factors through language-specific training regimes, on which we ran a psycho-linguistically plausible computer model of inflection learning. In this way, we could understand more of factor interaction through a quantitative analysis of the way the performance of our system is affected across different training regimes. Methodologically, it allows for much more flexible and controlled test/analysis protocols than those commonly used with human subjects in experimental psycholinguistics.

In addition, understanding more of the real cognitive hurdles a human learner has to face in the process of effectively acquiring an inflectional system of average complexity may also shed some light on optimal practices for language teaching.

[5] This graded notion takes into accout exceptional alternating stems in otherwise regular paradigms (e.g. Italian *aprire/aperto* and Spanish *abrir/abierto*, "open" infinitive/"opened" past participle). At the same time, it captures the difference between partially irregular paradigms and radically idiosynchratic ones.

6. Bibliographical References

Ackerman, F. and Malouf, R. (2013). Morphological organization: The low conditional entropy conjecture. *Language*, 89(3):429–464.

Baayen, H. R., Piepenbrock, P., and Gulikers, L., (1995). *The CELEX Lexical Database (CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

Balling, L. W. and Baayen, R. H. (2008). Morphological effects in auditory word recognition: Evidence from danish. *Language and Cognitive processes*, 23(7-8):1159–1190.

Balling, L. W. and Baayen, R. H. (2012). Probability and surprisal in auditory comprehension of morphologically complex words. *Cognition*, 125(1):80–106.

Bane, M. (2008). Quantifying and measuring morphological complexity. In *Proceedings of the 26th West Coast Conference on Formal Linguistics*, pages 69–76.

Matthew Bearman, et al., editors. (2015). *Understanding and Measuring Morphological Complexity*. Oxford University Press.

Bickel, B. and Nichols, J. (2005). Inflectional synthesis of the verb. In David Gil Martin Haspelmath, Matthew S. Dryer et al., editors, *The World Atlas of Language Structures*, pages 94–97. Oxford University Press.

Dagmar Bittner, et al., editors. (2003). *Development of Verb Inflection in First Language Acquisition: a cross-linguistic perspective*. Mouton de Gruyter, Berlin.

Blevins, J. P. (2003). Stems and paradigms. *Language*, 79(2):737–767.

Blevins, J. P. (2016). *Word and Paradigm Morphology*. Oxford University Press, Oxford.

Boyé, G. and Cabredo Hofherr, P. (2006). The structure of allomorphy in spanish verb inflection. *Cuadernos de Lingüística*, 13:9–24.

Brown, R. (1973). *A first language: the early stages*. George Allen I& Unwin.

Burzio, L. (1998). Multiple correspondence. *Lingua*, 104:79–109.

Bybee, J. and McClelland, J. L. (2005). Alternatives to the combinatorial paradigm of linguistic theory based on domain general principles of human cognition. *The Linguistic Review*, 22(2-4):381–410.

Bybee, J. (1995). Regular morphology and the lexicon. *Language and Cognitive Processes*, 10(5):425–455.

Dimitropoulou, M., Duñabeitia, J. A., Avilés, A., Corral, J., and Carreiras, M. (2010). Subtitle-based word frequencies as the best estimate of reading behavior: the case of greek. *Frontiers in Psychology*, 1(218):1–12.

Evans, N. and Levinson, S. C. (2009). The myth of language universals: Language diversity and its importance for cognitive science. *Behavioral and Brain Sciences*, (32):429–92.

Ferro, M., Marzi, C., and Pirrelli, V. (2011). A self-organizing model of word storage and processing: implications for morphology learning. *Lingue e Linguaggio*, 10(2):209–226.

Finkel, R. and Stump, G. (2007). Principal parts and morphological typology. *Morphology*, 17:39–75.

Hahn, U. and Bailey, T. M. (2005). What makes words sound similar? *Cognition*, 97:227–267.

Juola, P. (1998). Measuring linguistic complexity: The morphological tier. *Journal of Quantitative Linguistics*, 3(5):206–213.

Kolmogorov, A. N. (1965). Three approaches to the quantitative definition of information. *Problems of information transmission*, 1(1):1–7.

Lyding, V., Stemle, E., Borghetti, C., Brunello, M., Castagnoli, S., Dell' Orletta, F., Dittmann, H., Lenci, A., and Pirrelli, V. (2014). The paisá corpus of italian web texts. Proceedings of the 9th Web as Corpus Workshop (WaC-9)@ EACL 2014, pages 36–43. Association for Computational Linguistics.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The penn arabic treebank: Building a large-scale annotated arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467.

Marr, D. (1982). *Vision. A Computational Investigation into the Human Representation and Processing of Visual Information*. Freeman and Company.

Marzi, C. and Pirrelli, V. (2015). A neuro-computational approach to understanding the mental lexicon. *Journal of Cognitive Science*, 16(4):493–535.

Marzi, C., Ferro, M., and Pirrelli, V. (2014). Morphological structure through lexical parsability. *Lingue e Linguaggio*, 13(2):263–290.

Marzi, C., Ferro, M., Cardillo, F. A., and Pirrelli, V. (2016). Effects of frequency and regularity in an integrative model of word storage and processing. *Italian Journal of Linguistics*, 28(1):79–114.

Matthews, P. H. (1991). *Morphology*. Cambridge University Press, Cambridge.

McWorther, J. (2001). The world's simplest grammars are creole grammars. *Linguistic Typology*, (5):125–166.

Moscoso del Prado Martín, F., Kostić, A., and Baayen., R. H. (2004). Putting the bits together: An information theoretical perspective on morphological processing. *Cognition*, 94(1):1–18.

Noccetti, S. (2003). Acquisition of verb morphology in italian: A case study. In Dagmar Bittner, et al., editors, *Development of Verb Inflection in First Language Acquisition: a cross-linguistic perspective*, pages 351–378. Mouton de Gruyter, Berlin.

Pickering, M. J. and Garrod, S. (2013). An integrated theory of language production and comprehension. *Behavioral and Brain Sciences*, 36:329–392.

Pirrelli, V. and Battista, M. (2000). The paradigmatic dimension of stem allomorphy in italian verb inflection. *Italian Journal of Linguistics*, 12(2):307–380.

Pirrelli, V., Ferro, M., and Marzi, C. (2015). Computational complexity of abstractive morphology. In Matthew Baerman, et al., editors, *Understanding and Measuring Morphological Complexity*, pages 141–166. Oxford University Press, Oxford.

Pirrelli, V. (2000). *Paradigmi in morfologia. Un approccio interdisciplinare alla flessione verbale dell'italiano*. Istituti Editoriali e Poligrafici Internazionali, Pisa.

Ralli, A. (2005). *Morfologia [Morphology]*. Patakis, Athens.

Ralli, A., (2006). *On the role of allomorphy in inflectional morphology: evidence from dialectal variation*, pages 123–152. Polimetrica, Monza.

Ramscar, M. and Dye, M. (2011). Learning language from the input: Why innate constraints can't explain noun compounding. *Cognitive Psychology*, 62(1):1–40.

Ramscar, M. and Yarlett, D. (2007). Linguistic self-correction in the absence of feedback: A new approach to the logical problem of language acquisition. *Cognitive Science*, 31(6):927–960.

Rescorla, R. A. and Wagner, A. R. (1972). A theory of pavlovian conditioning: variations in the effectiveness of reinforcement and non-reinforcement. In Abraham H Black et al., editors, *Classical conditioning II: Current research and theory*, pages 64–99. Appleton-Century-Crofts, New York.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423.

Shosted, R. (2006). Correlating complexity: a typological approach. *Linguistic Typology*, (10):1–40.

Walther, G. and Sagot, B. (2011). Modélisation et implémentation de phénomènes flexionnels non-canoniques. *Traitement Automatique des Langues*, 2(52):91–122.

Parser combinators for Tigrinya and Oromo morphology

Patrick Littell[1], **Tom McCoy**[2], **Na-Rae Han**[3], **Shruti Rijhwani**[4]
Zaid Sheikh[4], **David Mortensen**[4], **Teruko Mitamura**[4], **Lori Levin**[4]
[1]National Research Council of Canada, [2]Johns Hopkins University, [3]University of Pittsburgh, [4]Carnegie Mellon University
Digital Technologies, Dept. of Cognitive Science, Dept. of Linguistics, Language Technologies Institute
1200 Montreal Road, 3400 N. Charles Street, 2816 Cathedral of Learning, 5000 Forbes Avenue,
Ottawa, ON K1A 0R6, Baltimore, MD 21218, Pittsburgh, PA 15260, Pittsburgh, PA 15213
patrick.littell@nrc.gc.ca, tom.mccoy@jhu.edu, naraehan@pitt.edu, {srijhwan, zsheikh, dmortens, teruko, lsl}@cs.cmu.edu

Abstract

We present rule-based morphological parsers in the Tigrinya and Oromo languages, based on a *parser-combinator* rather than *finite-state* paradigm. This paradigm allows rapid development and ease of integration with other systems, although at the cost of non-optimal theoretical efficiency. These parsers produce multiple output representations simultaneously, including lemmatization, morphological segmentation, and an English word-for-word gloss, and we evaluate these representations as input for entity detection and linking and humanitarian need detection.

Keywords: morphology, parsing, Tigrinya, Oromo

1. Introduction

In this paper, we experiment with using parser combinators (Hutton and Meijer, 1988; Frost and Launchbury, 1989) for the rapid development of practical morphological parsers, as an alternative or supplement to the typical finite-state transducers (Karttunen and Beesley, 1992; Karttunen, 1993). This paradigm offered some practical advantages over a finite-state system, allowing parsers to be written very rapidly in a familiar programming language, although at a cost of runtime efficiency.

We present morphological parsers for two Afroasiatic languages, the Tigrinya language of Eritrea and Ethiopia (§4.1.), and the Oromo language of Ethiopia and Kenya (§4.2.).[1] These parsers were designed during the LoReHLT17 "surprise-language" evaluation (Strassel and Tracey, 2016) (§3.) to support machine translation, entity detection and linking, and humanitarian need detection (Strassel et al., 2017). These parsers were operable within about 36 hours of learning the identity of the languages, although they underwent further development during the next two weeks of evaluation.

2. Parser combinators

2.1. Introduction

The "parser combinator" paradigm (Burge, 1975; Wadler, 1985; Hutton and Meijer, 1988; Frost and Launchbury, 1989) is a kind of declarative programming that simultaneously defines the grammar being parsed and the executable code that parses it. This paradigm involves defining parser functions with a particular semantics, using a general-purpose programming language (in our case, Python), as well as defining higher-order "combinators" that take one or more component parsing functions as arguments and return a complex parsing function (for example, the composition of the two functions).[2] The resulting grammar is itself an executable function, that parses text as a recursive-descent parser, but one in which every component has access to all the capabilities and libraries of the general-purpose programming language (e.g. regular expression libraries, file I/O, etc.).

The basic building block of a parser-combinator grammar is the *atomic parser*. An atomic parser can do something as simple as recognize a single letter—in our examples, a parser defined as `Tex("d")` would recognize a single character `"d"` at the edge of the input's `Text` representation. By default, our parsers consume input from the right edge of a word because of most languages' tendency towards suffixation, but this is configurable. Parsers return a set of ⟨*output, remnant*⟩ ordered pairs, where *remnant* is what is left over from the input (e.g., the string without `"d"`), and *output* is any arbitrary Python object, typically some manner of structured or augmented representation of what was parsed. The return value of a parser is defined as a set, rather than a single ordered pair, because there may be, at any particular stage of the parse, multiple competing hypotheses regarding what the output representation should be and what remains to be parsed.

In our implementations, we typically define one atomic function to parse one morph, rather than define atomic parsers for individual characters. We also make significant use of generators, trivial atomic parsers that consume no input, but put a textual representation (such as a gloss) into one of several output channels. The concatenation of parsers with generators achieves the transduction between multiple different types of representation.

The other kind of parsers in this paradigm are *parser combinators*, functions that take one or more parsers as input and return a parser as output. The two prototypical parser combinators are concatenation and disjunction (implemented in our system by overloading Python's `+` and `|` operators, respectively). A concatenation `A+B` is defined as applying the parser A to each remnant produced by B and concatenating the output representations[3]; a disjunction `A|B` is simply

[1] `github.com/littell/ethi_morph`

[2] The usage of "combinator" here comes not from combinatoric logic but from the "combinator pattern" (`wiki.haskell.org/Combinator_pattern`) of software design.

[3] As noted above, our parsers default to right-to-left parsing,

```
> ROOT = Tex("jump") | Tex("talk") | Tex("think")
> SUFFIX = Tex("ed") | Tex("ing") | Tex("s") | NULL
> WORD = ROOT + SUFFIX
> WORD.parse("jumping") != [] # Does the parser return any results?
True
> WORD.parse("talker") != []
False
```

Figure 1: A simple English parser for twelve inflected verb forms, illustrating how concatenation and disjunction combinators allow the executable definition of a parser to resemble a familiar BNF-like grammar specification. Note that in a real grammar, we would not list every root or stem (e.g., "jump") as a literal expression in the code; rather, we would typically define a "lookup" parser that loads in one or more dictionary files from disk.

```
> ROOT = Tex/Mor/Lem("ugaandaa") + Glo/Nat("Uganda")
> SUFFIX = Tex/Mor("tti") + Glo("LOC") + Nat("in (.*)")
> WORD = ROOT + SUFFIX
> WORD.parse("ugaandaatti")
[{"breakdown":"ugaandaa-tti", "lemma":"ugaandaa", "gloss":"Uganda-LOC", "natural":"in Uganda"}]
```

Figure 2: A simple Oromo example parser with multiple output representations: a morphological breakdown, a lemma, a glossed breakdown, and a more naturalistic English-like gloss.

defined as the union of the results.

By overloading `+` and `|` in this way, we can express the *code* that executes as if it were the *grammar* that it is parsing. That is to say, a formal representation of the grammar is also the actual code that executes to parse this grammar; Figure 1 illustrates this for a few English verbs.

2.2. Example

The parsers described in this paper use a combination of parser function objects (which consume input strings) and generator function objects (parsers that trivially succeed without consuming input, while outputting additional representations) to convert one representation into others. Types of representation are conceptualized in this system as "channels", from which the parsers consume or output text representations. For example, in Figure 2, there are five channels, an input channel `Tex` (text) and four output channels: `Mor` (morphological breakdowns), `Lem` (lemmas), `Glo` (glossed breakdowns), and `Nat` (naturalistic English-like glosses).

A textual representation (like the suffix `"tti"`) is turned into a parser or generator by associating it with a specific channel: `Tex(X)` defines a parser that consumes `X` from the `Tex` input channel, while `Mor(X)` defines a generator that consumes no input but outputs `X` into the `Mor` output channel. Where the representations happen to be identical, as they are in this example for the suffix `"tti"`, this can be abbreviated as `Tex/Mor(X)`; this is just syntactic sugar for `Tex(X) + Mor(X)`.

In this Oromo example, the surface form `ugaandaatti` ('in Uganda') is broken down and transformed by a concatenation of parsers (the `Tex` components) and generators (the `Mor`, `Lem`, `Glo`, and `Nat` components). The `Tex` parsers consume the suffix `"tti"` and the root `"ugaandaa"` in turn; while the `Mor` generators consume no input but put `"tti"` and `"ugaandaa"` into the appropriate output channel (with the formatting appropriate for that channel, in this case a hyphen). The nature of concatenation is such that if any of these components fails to produce an ⟨*output,remnant*⟩ pair, the whole will fail to produce any output, so the `Mor`-channel generation only survives if the `Tex`-channel parsing succeeds.

Meanwhile, the `Glo` components generate `"LOC"` and `"Uganda"`, the `Lem` component only generates `"ugaandaa"`, and the `Nat` components produce a naturalistic gloss by generating `"Uganda"` and also inserting it into a particular template (`"in (.*)"`).

Since these parsers can return multiple outputs, the outputs were ranked heuristically by adding penalties for generating lemmas that were not found in any of the available dictionaries, for parses found in the dictionary but with unlikely (according to an English language model) definitions, and for parses that contain certain dispreferred morphemes.

2.3. Advantages of parser combinators

Parser combinators have several practical advantages, particularly in time-constrained situations:

Familiar grammar format The morphological grammars have a familiar, Backus-Naur-like format that mirrors the way linguists already think about grammars. Unlike grammars defined in terms of continuation classes, these grammars are easy to refactor as the linguists/programmers discover more about the morphotactics of the language in question.

For example, when one discovers that tense suffixes do not immediately follow verb roots, but there is a mood suffix that can intervene, one does not have to update every verb root so that its continuation class is `MoodSuffix` rather

so B is evaluated first unless the programmer specifies otherwise. Also, what "concatenation" means depends on the type of output representation; often, it is just a string concatenation of the output strings with a delimiter.

than `TenseSuffix`; one simply has to change `VerbRoot + TenseSuffix` to `VerbRoot + MoodSuffix + TenseSuffix` on one line of the grammar.

Familiar programming syntax and environment The programming syntax and execution environment is familiar Python. Boilerplate and repetitive code (e.g., a class of morphemes all of which have a complex environmental restriction or cause a particular morphophonological change) can be automated within the code itself; it is unnecessary to have a separate transpilation or pre-processing step, as was done in (Littell et al., 2014), to enable a new command or syntactic sugar. Even complex functions entirely outside of the parsing paradigm (e.g., orthographic conversion and normalization, dictionary lookup, etc.) can be wrapped up as a parser object and integrated into the morphological grammar.

Multi-output parsers It is straightforward to associate a parser with multiple different kinds of outputs (Hutton, 1992) letting us simultaneously write parsers that target different representations for different NLP tasks.

Intuitive representation of morphological phenomena Some morphological phenomena that are awkward to express as finite-state transducers are more straightforwardly expressed in a recursive grammar, such as the kinds of templatic or circumfixal morphology that requires finite-state transducers to be extended with flag "memory" (Pretorius and Bosch, 2003; Bower et al., 2017).

It is worth noting that there is no conceptual requirement that an atomic parser define a string truncation like "remove `'d'` from the end of a string", although because of the concatenative nature of most morphology this is the most common kind of atomic parser. The relationship between the input string and the remnant string can be any string-to-string transduction. In the Tigrinya system (§4.1.), we defined some parsers using regular expression substitutions, to handle some particularly difficult plurals that involve both reduplication and root-and-pattern morphology.[4]

Ease of extension We should emphasize here that the above advantages are not just put forward as benefits of a particular parsing *library*, but of a programming technique; one of the benefits of mastering this technique is that, because parser combinators are themselves simple to write from scratch, the programmer is not constrained by the capabilities of an existing parsing library. Extending a library—or rewriting it entirely—is often only a matter of an hour or two, and is not a separate process from programming the grammar itself, since both the library and the grammar are written in the same programming language.

Nevertheless, the small parser-combinator library that we release with these Tigrinya and Oromo grammars should serve as a good starting point for the development of morphological parsers in other languages, as it includes a number of convenience features for the particular problem domain, like predefined output channels for typical word representations (e.g. morphological breakdowns and glosses), specialized parsers for root-and-pattern morphology and reduplication, and combinators that allow parsing either from the left (for prefixes) and the right (for suffixes).

2.4. Disadvantages of parser combinators

On the other hand, there are some drawbacks compared to finite-state systems:

Efficiency After compilation, a finite-state transducer executes in linear time, while parser combinators result in a recursive descent parser with potentially exponential time complexity. For the most part, *morphological* grammars do not have the kind of complexity (in particular left-recursion) that leads to worst-case performance, but nonetheless it is important to note that the responsibility for parser performance here falls back onto the programmer, rather than being handled in the compiler, which is a clear benefit to the finite-state paradigm.[5]

Multi-representation ambiguity While the ability of our parser combinators to define relationships between multiple levels of representation (e.g. text, breakdown, gloss, etc.) was practically useful in a multi-task setting like LoReHLT17, the ambiguity in *each* representation is multiplicative with others. For example, if a particular word has five possible parses in one representation, three in another, and two in another, the parser could return as many as thirty representations. This can pose an efficiency problem, since parser combinators do not have the inherent efficiency of finite-state systems when faced with parse ambiguity.

This is a downside of the particular multi-representation system that we engineered here, but more broadly the arbitrary complexity of *outputs* in a parser-combinator system (which we leveraged here to allow multiple output representations) present another possible source of inefficiency that is not a concern in two-level finite-state systems.

3. LoReHLT17

Our parsers were constructed in the context of the 2017 Low Resource Human Language Technologies (LoReHLT) evaluation[6]. LoReHLT takes the form of a "surprise language" exercise (Oard, 2003), in which competitors are asked to produce machine translation, entity detection and linking (EDL), humanitarian need detection, and sentiment detection in one or more low-resource languages within a series of timed checkpoints, without knowing ahead of time what the languages will be. LoReHLT17 had its first checkpoint after 3 days and two additional checkpoints after 10 and 17 days; future LoReHLT evaluations will have their first checkpoint after 24 hours. In this timeframe, any hand-written rule-based systems must prioritize *programmer time* along with runtime efficiency: they must be the kind of systems that can be written in a day or two.

When a team writing morphological analyzers is asked to support more than one task such as MT or NER/EDL, it is not unusual for the consumers (which might include both

[4] Note, however, that allowing parsers to execute arbitrary transductions removes some guarantees—it is possible to define parsers that never halt—and some possibilities for optimization.

[5] In the time-constrained environment of LoReHLT17, however, the training and test corpora were sufficiently small that runtime efficiency was not the primary bottleneck; programmer time was a more pressing concern, especially during the earliest stages.

[6] `www.nist.gov/itl/iad/mig/lorehlt17-evaluations`

```
> NTexMor = lambda x : Tex/Mor(x) | Truncate("t", Tex) + Tex("n") + Tex/Mor(x)
> ROOT = Tex/Mor/Lem("nyaat") + Glo/Nat("eat")
> SUFFIX = NTexMor("na") + Glo("1PL.PRS") + Nat("we (.*)")
> WORD = ROOT + SUFFIX
> WORD.parse("nyaanna")
[{"breakdown":"nyaat-na", "lemma":"nyaat", "gloss":"eat-1PL.PRS", "natural":"we eat"}]
```

Figure 3: An Oromo example parser illustrating the phonological process that `t` → `n` | `_n`. The `NTexMor` line encodes this phonological rule, and writing a morpheme as an argument to `NTexMor`, as with `NTexMor("na")`, indicates that the rule applies at boundaries involving that morpheme.

NLP systems and human annotators) to request different representations of the morphology. Does the team have to write three or four different parsers or *reparsers* to parse outputs into different formats? Our approach makes this unnecessary, providing a good balance between programmer time and output flexibility.

4. System description

This year's LoReHLT task involved Tigrinya and Oromo, two languages spoken in the Horn of Africa. In this section we describe the parser-combinator systems that we created for these languages. These two case studies illustrate some of the benefits of the parser-combinator framework in the context of a time-sensitive task.

4.1. Tigrinya

The Tigrinya language is a member of the Semitic branch of the Afroasiatic family. It is spoken by over 7 million people, mainly in Ethiopia and Eritrea. [7] Like many other Semitic languages, Tigrinya has a templatic morphology system, meaning that the surface form of a word's root morpheme differs in different morphological contexts. For example, the lexical entry for 'bee' is just the consonants */nhb/*, and the vowels appearing between these consonants differ for different inflections of this root, such as the singular form *[nihibi]* and the plural form *[ʔanahib]*. Such root-template patterns are nonconcatenative, meaning that they cannot be expressed simply as the concatenation of morphemes. Because finite-state transducers are inherently concatenative mechanisms, nonconcatenative processes pose a special challenge to finite-state methods. Below we will describe how our system overcomes this challenge as an example of how our parser combinator framework solves some issues inherent in using finite-state methods.

Tigrinya is written in the Ge'ez script which is a form of alphasyllabary (abugida). We begin by converting the script to IPA using Epitran[8], a Python library for transliterating orthographic text as IPA (International Phonetic Alphabet). We used two Epitran mappings for Tigrinya. The first is Epitran's `tir-Ethi` mapping, which is a faithful one-to-one transliteration wherein each Ethiopic symbol is realized as a consonant-vowel sequence (e.g., ትግርኛ → *tigriɲa*). There is, however, an ambiguity in one set of letters (the "sixth series"), which represents both consonants followed by /ɨ/ (e.g. /gɨ/) and consonants that are not followed by any vowel (e.g. /g/). In this first Epitran mapping, all such letters are transliterated as *Cɨ* unless they are at the end of the word, in which case they are transliterated as just *C*.

The second mapping is `tir-Ethi-pp`, a 'precision-phonemic' IPA representation (e.g., ትግርኛ → *tigriɲa*). In this system, sixth series letters are realized as consonants and /ɨ/ is inserted where demanded by the syllable structure, yielding a phonemic representation that is closer to Tigrinya speech and more suitable for some tasks downstream from our morphological analyzer, including speech recognition. Keeping to the `tir-Ethi` representation system-internally enabled us to simplify the grammar development process where the success of morphological analysis is defined on the level of orthographic, rather than phonetic, tokens, with the exception of the word-final position where the final vowel, present in the script, is omitted.

Grammar development centered around several different linguistic aspects. Concatenative morphology was the initial focus leading up to checkpoint 1. Plural suffixes, negative prefixes and pronominal clitics on verbs were handled, while minimal attention was paid to morpho-phonemic processes. At the same time, a frequency list of Tigrinya word types was compiled from a corpus, which served as (1) a type list to prioritize, and (2) in the absence of gold data, a basis for recall-oriented performance evaluation and monitoring.

Tightening up morpho-phonemic rules for better handling of allomorphs and treatment of templatic verbal morphology became the main goals for the second checkpoint. However, documentation on numerous templates was incomplete at best. Given this lack of information and time pressure, we decided to look to existing solutions. Gasser's HornMorpho (Gasser, 2011) is a finite-state transducer (FST) capable of analyzing Tigrinya verbs (but not nouns) with a reported 96% accuracy. We compiled a list of the 5,700 most frequent verb types, processed them through HornMorpho, re-parsed the output to conform to our system's output, and made the cached analyses available to the lookup routine.

Before the final checkpoint, we concentrated on the so-called "internal plurals" – plural nouns built from consonantal roots and templates. They were handled via several regular-expression-based root patterns, 8 sets in total, extending the system's coverage to such plural nouns as ኣናሂብ *ʔanahib* 'bees', whose singular form is ንህቢ *nihibi* 'bee'. To handle root-and-template patterns, we expressed these templates as Python regular expressions, and wrapped

[7] According to Ethnologue: `https://www.ethnologue.com/language/tir`.

[8] `github.com/dmort27/epitran`

each into a parser function; this way non-concatenative morphology and concatenative morphology could be combined.

The class of nouns for the 'bee/bees' example above used the following template, which operates on a triplet of the pre-defined consonant class `C`, which are integrated into the specified positions into a regular expression

```
pat2 = 'ʔa(%s)a(%s)ɨ(%s)' % (C, C, C)
```

Additionally, the parser's lexicon base was dramatically expanded to include a gazetteer, hand-edited entries, and multiple dictionaries sourced from the web and known multilingual resources.

As stated earlier, coverage of the morphological system as measured against a Tigrinya corpus was the only metric of performance available to us, and each update to the system was made sure to result in an increase in recall. At the same time, we closely monitored the system's performance on the annotation front, with annotators reporting in anomalies or undesirable parser behaviors.

Of particular importance was achieving a balance between coverage and overanalysis. An example would be place names such as ኡጋንዳ *ʔuganɨda* 'Uganda': unless they are present in the lexicon, such proper nouns will be overanalyzed, with the final 'a' separated out as the 3rd person singular feminine possessive marker. With an abundance of country and place names ending in 'a', such overanalysis would have a negative impact on downstream applications such as entity detection and linking. We addressed this issue by employing the `Cost` channel of the parser combinator library, which allows the programmer to add an arbitrary penalty anywhere in the grammar. The total penalty associated with a particular parse output is then taken into account by the heuristic disambiguator.

In the case of 'a', the suffixation rule was penalized with a sufficient amount of cost to make sure that the analyzed root+'a' form gets outranked by the whole, unanalyzed word, unless the remnant root is found in the lexicon. Many similar affixation rules were penalized with varying amounts of cost, which were manually determined and assigned with the aim to engineer desired ranking behaviors among multiple analysis candidates.

4.2. Oromo

Oromo is a language in the Cushitic branch of the Afroasiatic language family. It is spoken by nearly 25 million people in Ethiopia. [9] Unlike Tigrinya, Oromo does not possess templatic morphology. Instead, its morphological processes are generally concatenative, which means that they involve only the concatenation of stems and affixes (though it also has several phonological changes that occur at morpheme boundaries, meaning that its morphological processes do not consist of plain concatenation). These affixes specify many different features including tense and aspect (for verbs) or gender and number (for nouns).

Because Tigrinya morphology poses certain challenges to traditional finite-state methods, we used the Tigrinya section to showcase a theoretical advantage of our system in terms of its expressive capabilities. Oromo morphology, on the other hand, does not pose such theoretical challenges, so we will instead use this section to illustrate the practical benefits of our parser combinator framework. That is, even for a case (such as Oromo) where the increased expressive capabilities of parser combinators are not a benefit, we show how the ease of implementation of parser combinators can make them a good choice for implementing morphological parsing under time constraints.

Oromo uses the Roman script, which is sufficiently close to a phonological representation of the language that we did not have to transliterate Oromo into IPA as was necessary for Tigrinya. For checkpoint 1, we focused mainly on nominal morphology because nouns are the most crucial part of speech for some of the downstream tasks such as NER-EDL. For example, Figure 2 identifies the lemma *ugaandaa* within the word *ugaandaatti*, informing the downstream NER system that the two words refer to the same named entity. For the next checkpoint, we added handling of other parts of speech to the parser, with a focus on verbs (since most multi-morpheme words in Oromo are either nouns or verbs). For the final version of the parser included in the submission to checkpoint 3, we polished the existing functionalities of the parser, increased its lexical coverage, and incorporated orthographic normalization.

As stated above, one of the major advantages of using parser-combinators is in saving programmer time. The construction of the Oromo grammar provides case studies in how parser combinators can save programmer time through three of their key properties: integration with Python, multiple channels, and streamlined handling of phonological transformations.

Integration with Python: It is easy to incorporate Python functions and objects into the parsing pipeline. We used a custom Oromo orthography normalizer, written in Python, to handle considerable variability in Oromo orthography such as multiple different spellings for "Ethiopia" (⟨Itoophiya⟩, ⟨Itoophiyaa⟩, ⟨Itophiyaa⟩, etc.). Also, as our set of lexical resources for Oromo grew throughout the project, we added dictionaries including one that was dynamically updated as human annotators annotated text in Oromo. Here, the Python environment made it easy to read in each dictionary and compile it into a Python dictionary for lexical lookup, whereas without the Python IO tools some more tedious method for adding each new word/definition pair to the parsing program would have been necessary.

Multiple channels: Figure 2 shows a simple chunk of Oromo grammar written using parser-combinators. This grammar has four different channels covering morphological breakdown, lemma, gloss, and natural gloss. In a standard finite-state parser, each one of these four channels would have had to be implemented by its own separate finite-state transducer, but with parser-combinators all four can coexist in the same file. Such consolidation is helpful in two ways: First, it saves the programmer time during the initial creation of the grammar. Second, when making future updates to the grammar, the consolidation means that the programmer only has to update one program rather than

[9] According to the 2007 Ethiopian census, available at `http://microdata.worldbank.org/index.php/catalog/2747`.

Language	Measurement	Original word	Lemma
Tigrinya	typed_mention_ceaf_plus F1	0.478	**0.521**
Oromo	typed_mention_ceaf_plus F1	0.355	**0.376**

Table 1: Results on entity detection and linking, before and after adding lemmatization.

Language	Measurement	Original word	Lemma
Tigrinya	SFType F1	0.325	**0.333**
Tigrinya	SFType occurrence-weighted F1	0.422	**0.471**
Oromo	SFType F1	0.047	**0.086**
Oromo	SFType occurrence-weighted F1	0.051	**0.124**

Table 2: Results on Situation Frame detection, before and after adding lemmatization.

four, possibly saving the effort of version control across multiple files. Since our grammars were continuously being updated, streamlining the updating process was very important.

Streamlined handling of phonology: In traditional FSTs, phonological transformations are typically handled with two separate transducers, one that builds up what might be termed an underlying representation of a word (such as *inlogical*) and another that transforms this underlying representation into a surface form (such as *illogical*). This is computationally elegant, but can be difficult to engineer, since it separates the phonological effects of morphemes in code from the specification of the morphemes that trigger those effects, necessitating workarounds (e.g. special, non-pronounced characters) to pass information between the transducers. For example, many phonological transformations only occur at morpheme boundaries, so a finite-state grammar writer has to keep track of morpheme boundaries to make sure that *inlogical* turns to *illogical* but *only* does not turn into *olly*. Phonological transformations can also be sensitive to exactly which morphemes are being combined, so it might be necessary to keep track not just of where the morpheme boundaries are but also what sort of morpheme boundaries they are so that, for example, *inlogical* turns to *illogical* but *unlikely* does not turn into *ullikely*.

For parser-combinators, however, a phonological rewrite rule can be represented simply as a parser and treated like any other parser in the BNF-style rules of the grammar. This removes the need for a separate finite-state transducer to handle the phonology and also removes the need to keep track of different types of morpheme boundaries.[10] Since this approach require no single lexical representation intermediate between orthographic form and morphological form, as is typically the case with FST morphological analyzers, every juncture can be handled on a morpheme-by-morpheme basis. This necessarily results in some loss of generality but is actually a boon for maintainability. Figure 3 shows one Oromo phonological rewrite rule implemented with parser-combinators.

5. Experiments and results

We report here the results of adding the Tigrinya and Oromo parsers (specifically, their lemmatization function) to our LoReHLT17 entity detection and linking (§5.1.) and humanitarian need assessment (§5.2.) systems.
It should be noted that in all tasks the scores for Oromo are substantially lower than the scores for Tigrinya; all participants in LoReHLT17 encountered this effect, due largely to the relatively small parallel corpora and lexicons available in Oromo and the large amount of spelling variation in Oromo text.

5.1. Entity Detection and Linking

Entity detection and linking (EDL) for LoReHLT17 was concerned with the recognition of named entities (a subset of proper nouns) in text, their categorization as one of four entity types (person, organization, location, geopolitical entity), and their linking to an external knowledge-base of entities (compiled from several existing databases). The primary metric for EDL in LoReHLT17 was *typed_mention_ceaf_plus*, an F1 measure of detecting the entity and getting both the category and the link correct. We used word-to-word translation with bilingual lexicons for linking entities to the knowledge-base (Pan et al., 2017). Adding lemmatization improved translation of the entities and resulted in F1 point gain for both Tigrinya and Oromo, as seen in Table 1.

5.2. Situation Frame detection

Situation frames (SFs) are a structured representation of events intended to "enable information from many different data streams to be aggregated into a comprehensive, actionable understanding of the basic facts needed to mount a response to an emerging situation" (Strassel et al., 2017). Situation frame detection involves detecting eight humanitarian requirements (e.g. water, food, medicine, evacuation) and three background issues (e.g. terrorism or civil unrest), linking these needs and issues to places, and determining whether or not this is a current, urgent, unrelieved need.
The basic evaluation metric for SF detection in LoReHLT17 is SFType[11]—whether the frame identify the correct

[10] On the other hand, since these rules are themselves complex parsers, they increase the complexity of the grammar and including many of them can affect the runtime performance of the parser.

[11] Additional metrics, such as SFType+Location, evaluate whether both the type and other fields of the frame are correct. These are bounded from above by SFType, and our improvement

needs and issues—measured by mean F1 and occurrence-weighted F1, which differ according to whether or not a situation frame is given greater weight when multiple annotators have annotated it. (That is to say, if only one annotator detects an evacuation need in a document, it counts less towards occurrence-weighted F1 than if all annotators detected it.)

Table 2 shows the results of adding lemmatization to our keyword-based situation frame detection system, compared to a system that attempts to identify keywords without lemmatization. Lemmatization adds a ~.01 F1 point improvement (~.05 when weighted for occurrence) to the Tigrinya system, and roughly doubles Oromo performance with a ~.04 F1 point improvement (~.07 when weighted for occurrence).

6. Future research

While this paper has presented parser combinators as if they were in opposition to finite-state methods, the two paradigms are compatible; the ability of parser combinators to incorporate arbitrary functions into their parsing paradigm means that there are no conceptual reasons why some parts of the grammar could not be parsed in a finite-state manner and others in a recursive-descent manner. We are therefore looking into the possibility of integrating Foma FSTs (Hulden, 2009) as parser functions, and/or compiling "safe" subgraphs of the grammar into finite-state systems, to take advantage of the linear time execution where it is possible.

The other benefit of finite-state parsers is that they can be run "backwards" (that is, generating rather than parsing). Incorporating this ability into a parser-combinator framework would be valuable both for pure parser-combinator systems and for the hybrid systems proposed above. The small parser combinator library released with these parsers already supports this to a limited degree: as seen in the examples in §2., the syntax for expressing a parser (like `Tex("tti")`) and a generator (like `Mor("tti")`) is identical, and a subset of parsers/generator functions have implementations such that they can either parse or generate depending on what channel is considered to be the input. A parser that consists solely of these functions can parse and generate in any direction (that is, between *any* two representations that the parser supports); however, both implementations described here use parser functions that do not yet have corresponding generators. We intend, in further development, to augment the library such that all parsers have a corresponding generator and thus any grammar written with this library can parse/generate between all of its representations.

7. Conclusion

By utilizing a declarative programming paradigm that allowed our linguist-programmers to use a familiar grammar formalism within a familiar general-purpose programming language, we created Tigrinya and Oromo parsers within a limited time-frame, that nonetheless led to consistent improvements in entity detection and linking and humanitarian need detection.

These parsers allowed us to rapidly capture some types of morphology (in particular root-and-pattern morphology) that, although possible within finite-state systems, can be difficult to engineer. In future work, we plan to generalize these specialty parsers into a general framework for parsing morphology-specific phenomena. We also intend to make wrappers that allow interoperation with finite-state systems, so that the efficiency of finite-state techniques can be combined with the ease of engineering of parser-combinator techniques.

on these metrics by adding lemmatization are roughly proportional to the SFType improvements shown in Table 2.

8. Acknowledgements

This project was sponsored by the Defense Advanced Research Projects Agency (DARPA) Information Innovation Office (I2O), program: Low Resource Languages for Emergent Incidents (LORELEI), issued by DARPA/I2O under Contract No. HR0011-15-C-0114.

9. Bibliographical References

Bower, D., Arppe, A., Lachler, J., Moshagen, S., and Trosterud, T. (2017). A morphological parser for Odawa. In *Proceedings of 2nd Workshop on Computational Methods for Endangered Languages (ComputEL-2)*.

Burge, W. H. (1975). *Recursive Programming Techniques*. Addison Wesley.

Frost, R. and Launchbury, J. (1989). Constructing natural language interpreters in a lazy functional language. *The Computer Journal*, 32:108–121.

Gasser, M. (2011). HornMorpho: a system for morphological processing of Amharic, Oromo, and Tigrinya. In *Proceedings of the Conference on Human Language Technology for Development*, Alexandria, Egypt.

Hulden, M. (2009). Foma: A finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 29–32. Association for Computational Linguistics.

Hutton, G. and Meijer, E. (1988). Monadic parser combinators. *Journal of Functional Programming*, 8:437–444.

Hutton, G. (1992). Higher-order functions for parsing. *Journal of Functional Programming*, 2:323–343, July.

Karttunen, L. and Beesley, K. R. (1992). *Two-Level Rule Compiler. Technical Report ISTL-1992-2*. Xerox Palo Alto Research Center, Palo Alto, CA.

Karttunen, L. (1993). *Finite-state lexicon compiler. Technical Report ISTL-NLTT-1993-04-02*. Xerox Palo Alto Research Center, Palo Alto, CA.

Littell, P., Price, K., and Levin, L. (2014). Morphological parsing of Swahili using crowdsourced lexical resources. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3333–3339, Reykjavik, Iceland, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1686.

Oard, D. W. (2003). The surprise language exercises. *ACM Transactions on Asian Language Information Processing (TALIP)*, 2(2):79–84, June.

Pan, X., Zhang, B., May, J., Nothman, J., Knight, K., and Ji, H. (2017). Cross-lingual name tagging and linking for 282 languages. In *ACL*.

Pretorius, L. and Bosch, S. E. (2003). Finite-state computational morphology: An analyzer prototype for Zulu. *Machine Translation*, pages 191–212.

Strassel, S. and Tracey, J. (2016). LORELEI language packs: Data, tools, and resources for technology development in low resource languages. In *LREC 2016: 10th Edition of the Language Resources and Evaluation Conference, Portoroz*, pages 3273–3280.

Strassel, S., Bies, A., and Tracey, J. (2017). Situational awareness for low resource languages: the LORELEI situation frame annotation task. In *SMERP2017: First International Workshop on Exploitation of Social Media for Emergency Relief and Preparedness*, Aberdeen.

Wadler, P. (1985). How to replace failure by a list of successes: A method for exception handling, backtracking, and pattern matching in lazy functional languages. In *Conference on Functional Programming Languages and Computer Architecture*, pages 113–128. Springer.

Massively Translingual Compound Analysis and Translation Discovery

Winston Wu, David Yarowsky
Department of Computer Science, Center for Language and Speech Processing
Johns Hopkins University
{wswu,yarowsky}@jhu.edu

Abstract

Word formation via compounding is a very widely observed but quite diverse phenomenon across the world's languages, but the compositional semantics of a compound are often productively correlated between even distant languages. Using only freely available bilingual dictionaries and no annotated training data, we derive novel models for analyzing compound words and effectively generate novel foreign-language translations of English concepts using these models. In addition, we release a massively multilingual dataset of compound words along with their decompositions, covering over 21,000 instances in 329 languages, a previously unprecedented scale which should both productively support machine translation (especially in low resource languages) and also facilitate researchers in their further analysis and modeling of compounds and compound processes across the world's languages.

Keywords: compounds, multilingual, translation

1. Introduction

Morphological compounding (e.g. *lighthouse* or *airport*) is one of the most common and productive methods of word formation across the world's languages (Denning et al., 2007), and yet its derivational processes and semantics can be quite complex.

Consider the semantic concept *hospital*, which can be realized via compound morphology in a remarkable diversity of semantic compositions, including:

Lang.	Compound	Literal Semantics
nl	ziekenhuis	sick + house
no	sykehus	sick + house
hu	kórház	disease + house
eo	malsanuelejo	sick + place
ms	rumah sakit	house + sick
zh	病院	disease + institution

There are clearly a wide variety of semantic associations constituting this concept (e.g. sick/disease + house/place/institution), a variety of constituent orders (e.g. sick+house vs. house+sick) and potentially a variety of compounding processes beyond simple concatenation (e.g. *sykehus* in Norse (no) is a compound of *syk* and *hus* with the insertion of an *e*).

The following paper presents a massively cross-linguistic computational model of both compound morphology compositional processes and compound semantics on a scale of over 300 languages.

Furthermore, the paper not only presents a derived analysis of the compounding process and semantics of compounds within a *single* language (e.g. German), as with much prior related work (e.g. Koehn and Knight (2003)), but does so via a joint model across essentially all the world's languages with adequate dictionary resources, an unprecedentedly large scale for this class of research, and with significant additional synergistic multilingual power.

In addition, the paper successfully applies these models and results to the valuable application of predicting novel translations of compound words, both to English (e.g. *kórház* → *disease* + *house* → *hospital*) and from English (e.g. *hospital* → *disease* + *house, sick* + *place, etc.* → *kórház etc.*), with valuable applications for translation dictionary expansion and out-of-vocabulary handling in machine translation, again on this uniquely large multilingual scale.

Finally, in conjunction with this paper, we release a novel and uniquely large-scale 329-language, 21,000+ example dataset[1] of these compound morphological analyses and their associated compositional and compound translations, a valuable resource for training models for derivational morphology processes and compound semantics, with direct application to machine translation, on this massively multilingual scale.

2. Compound Discovery

We begin only with freely available multilingual translation dictionaries extracted from open-source Wiktionary[2], with the hope that they contain both substantial examples of compounding in each language (e.g. *sykehus* (Norwegian) = *hospital* (English)) as well as translations of the constituents of these compounds (e.g. *syk* = *sick* and *hus* = *house*). Using these dictionaries, we develop a multi-iteration method for discovering both compound translation models or "recipes" motivated across multiple languages that can be used to construct or analyze new compound words that may not be in the dictionary. While there are many existing methods for compound splitting (e.g. Koehn and Knight (2003; Macherey et al. (2011)), we concern ourselves with compounds with two components, which can be combined by simple concatenation, optionally using a glue or filler string at the point of concatenation (Garera and Yarowsky, 2008), or by dropping the last character of the left component (henceforth drop-left). These three compounding processes productively cover a wide spectrum of our multilingual data and serve as an efficient foundation for training the semantic models of compounding in the absence of simple concatenation.

[1] github.com/wswu/worcomal
[2] www.wiktionary.org

Concept		Left +	Right
ninety	zh	九 (nine)	十 (ten)
Monday	nl	maan (moon)	dag (day)
December	fi	joulu (Jule)	kuu (month)
midnight	hu	éj (night)	fél (half)
Frenchman	nl	frans (French)	man (man)
businessman	th	นัก (person)	ธุรกิ (business)
pianist	de	klavier (piano)	spieler (player)
granddaughter	no	datter (daughter)	datter (daughter)
queen	hu	király (king)	nő (woman)

Table 1: Examples of compounds formed by simple concatenation of two components.

In the first iteration of our method, we begin by considering only simple concatenation of two components that both exist in the dictionary (e.g. kór+ház = sick+house). By collecting words in all languages that can be decomposed into such components, we construct compound recipes as in Fig. 1. This process involves accounting for the varied order of components using a reordering and clustering method, augmenting the initial list of compounds in a second iteration of compound discovery by allowing glue characters and the drop-left mechanism, and scoring each word decomposition to indicate its validity as a compound words.
Throughout this paper, we will use the concept "hospital" as a running example. This is an interesting illustrative example since it is not a compound word in English, but occurs as a compound in many other languages.

2.1. Simple Concatenation

Many compound words can be discovered by simply splitting a string into all possible two parts and performing a dictionary lookup on each part. In fact, the large majority of compound words in our dataset are simple concatenations of legal stand-alone words. Table 1 presents a sample of such simple compound words. Note that concatenation can result in false positives (e.g. Dutch hospitaal = hospita 'landlady' + al 'even'), which will be identified by the compound score described later.

2.2. Component Clustering

Within a compound word, the semantic ordering of the components often varies between languages. For example, compound words for the concept "hospital" have different component ordering in different languages:

Dutch:	zieken 'sick'	+	huis 'house'
Malay:	rumah 'house'	+	sakit 'sick'

To account for this variation, we cluster the components[3] using a notion of syntagmatic and paradigmatic analysis. For each concept, we first filter out components that only occur once to remove noise. To illustrate, the top 5 left and right components for the concept "hospital", before correcting for ordering, have the same components on both the left and right sides (Table 2). The numbers indicate the number of languages where we see that component on the left or right side, respectively.

[3]We use components to mean their English translations.

Left		Right	
sick	8	house	7
disease	7	home	6
house	6	institution	4
home	5	place	4
ill	4	court	3

Table 2: Component language counts for "hospital" before correcting for ordering.

hospital =

		+		
sick	11		house	10
disease	7		home	8
ill	7		building	5
illness	5		box	3
sickness	4		family	3
pain	3		place	3
patient	3		case	2
ache	2		dwelling	2
bottle	2		household	2
cottage	2		institution	2

Figure 1: Compounding recipe for the concept 'hospital' using simple concatenation. Numbers indicate the number of languages whose compound words' components had that translation.

Since words in some languages (e.g. Malay) have a "flipped" order relative to the dominant sequence, so "house" and "home" appear both as left and right components. For clustering, we compute a distance matrix between all components, where components on opposite sides of a compound word have a distance of 1, and components on the same side have a distance of 0. Clustering into two clusters with these distances results in cleaner compounding recipes as in Fig. 1. For presentation purposes, we match the order of the recipe components to the most common order across the compound words for a certain concept. Note also that the English component counts are per language, rather than per word. This was done to avoid overcounting (e.g. Hungarian kórház and kórházi both decompose into sick + house, so counting by word would artificially inflate the counts for each of those components).

2.3. Compound Validity Score

Not all words that can be decomposed into valid components are valid compound words. For example, the Dutch hospitaal (decomposed as hospita 'landlady' + al 'even') is clearly not a semantically meaningful compound word. To filter out these poor decompositions, we devise a measure of how well a compound's components follow the compound's recipe. A straightforward but effective score is the geometric mean of the highest counts of the left and right components, respectively. In situations where the a component is not in the recipe, it receives a language count of 0.1 (we do not use 0 because it will zero out the other term in the geometric mean). For example, the Hungarian kórházi (kór + házi; disease + house) receives a score of $\sqrt{8 \cdot 12} = 9.8$, indicating a good decomposition, while the Dutch 'hospitaal' (hospita + al; landlady + even) receives a score of $\sqrt{0.1 \cdot 0.1} = 0.1$, indicating a bad decomposition. Scores for the concept "hospital" are presented in Table 3. Roughly

Score	Lng	Decomposition
12.96	nn	sjukehus = sjuk + hus ; sick + house
12.96	nl	ziekenhuis = zieken + huis ; sick + house
12.96	nl	ziekenhuis = zieke + huis ; sick + house
12.96	tpi	haus sik = haus + sik ; house + sick
12.96	no	sykehus = syk + hus ; sick + house
12.96	nb	sykehus = syk + hus ; sick + house
12.96	ms	rumah sakit = rumah + sakit ; house + sick
12.96	sv	sjukhus = sjuk + hus ; sick + house
12.96	da	sygehus = syge + hus ; sick + house
12.96	da	sygehus = syg + hus ; sick + house
12.96	id	rumah = sakit rumah + sakit ; house + sick
12.96	sv	sjukhus = sjuka + hus ; sick + house
12.96	sv	sjukhus = sjuke + hus ; sick + house
12.96	no	sykehus = syke + hus ; sick + house
10.58	vi	bệnh viện = bệnh + viện ; sick + home
9.79	hu	kórház = kór + ház ; disease + house
9.79	hu	kórházi = kór + házi ; disease + house
6.93	gd	taigh-eiridinn = taigh + eiridinn ; house + patient
6.48	eo	malsanulejo = malsanulo + ejo ; sick + place
4.89	gd	taigh-leighis = taigh + leighis ; house + heal
4.00	zh	病院 = 病癥 + 院; disease + institution
4.00	zh	病院 = 病患 + 院; disease + institution
4.00	zh	病院 = 病 + 院; disease + institution
2.83	zh	病院 = 病者 + 院; patient + institution
2.83	zh	病院 = 病號 + 院; patient + institution
2.83	zh	病院 = 病人 + 院; patient + institution
2.83	zh	病院 = 病體 + 院; sickness + institution
2.00	zh	醫院 = 醫 + 院; heal + institution
1.18	tr	hastane = hasta + ne ; sick + en
1.18	tg	бемористон = бемор + -истон ; sick + -land

Table 3: High-scoring decompositions for compounds of the concept "hospital".

half of the decompositions across all concepts are not valid, and we generally found a score over 2 is sufficient to filter out false positives.

2.4. Compound Augmentation

The first iteration of compound word discovery only took into account simple direct concatenation. However, as noted previously, this is clearly not the only method for forming compounds. To augment the supported processes for generating compound words, we utilize the compounding recipes to construct new words by performing a Cartesian product on the left and right (English) component sets. To construct new compound words, each pair of (foreign) components is concatenated using two new mechanisms: a glue letter, (e.g. Swedish 'construction workers' byggnadsarbetare = byggnad + *s* + arbetare), and dropping the last letter of the left component (e.g. Finnish 'homework' kotitehtävä = koti*o* + tehtävä). We apply component clustering and score the new compound words as previously described. Note that recipes for one concept may result in a compound word for a new concept (e.g. in the second iteration, the Chinese 难处 'difficulty' was constructed using the *hospital* recipe 'ill' (难受) + 'place' (处) using the drop-left mechanism).

For our running example of "hospital", this second iteration resulted in the recipe in Fig. 2. Concatenation resulted in 37 total compounds. The single glue and drop-left increased

hospital =

sick	14		house	12
ill	11		home	8
disease	8		building	5
illness	7		place	4
diseased	5	+	box	3
patient	4		family	3
sickness	4		area	2
pain	3		bottle	2
ache	2		case	2
cure	2		cottage	2

Figure 2: Compounding recipe for the concept 'hospital' including glue character and drop left mechanisms.

this count to 51 compounds, and a 2 character glue added two more words, for a total of 53 compounds. Out of these possible compounds, 17 words had a compounding score less than 1.0, indicating they do not follow the "hospital" recipe.

3. Experiments

We would like to see how well our methods work on compound words it has not seen before. Specifically, we evaluate our compounding methods on two tasks:

1. f2e: Given a foreign compound word, can we decompose and translate it into English?
2. e2f: Given an English concept, can we predict what the compound word would be in a target language?

For these experiments, we randomly chose a test set of 100 languages, with one word from each language that is likely to be a compound word according to our model. Due to the large multilingual breadth of this test set, evaluating on this test set gives a good idea of how the model performs on any given language of the world, rather than focusing on a single language with much more limited cross-linguistic generality. The randomly chosen test set is shown in Table 4. We remove each test word from the training dictionary to simulate it as being out-of-vocabulary.

4. Results and Analysis

For the e2f task, we were able to successfully recover 87 of the 100 test words. In other words, after removing the test word from the dictionary, the model was able to reconstruct the translation of the compound word only from its components, because other languages used the same components in the compounding recipe, either in direct association or via previously unobserved derived associations via the Cartesian product and/or reordering models. This result underscores that even if one does not observe a certain combination of components in any of the training data (e.g. sickness+building), the model's inference that this semantic compounding is viable via model components of both reordering and semantic clustering of observed decompositions of other attested forms translating as *hospital*, facilitates our prediction that this novel association is more likely to occur and mean *hospital* if observed in monolingual target language data.

Lang	Test Pair		Foreign	Literal Foreign	f → e			e → f
Code	English (e)	Foreign (f)	Segmentation	Comp Translations	TopHyp	2ndHyp	Rank	Found?
ace	english	bahsa inggréh	bahsa inggréh	language english	**english**	old english	1	y
ada	thank you	mo tsumi	mo tsumi	you thank	**thank you**	-	1	y
ady	old man	лIыжъы	лIы жъы	man old	**old man**	husband	1	y
af	silkworm	sywurm	sy wurm	silk worm	**silkworm**	-	1	y
akk	head	??	? ?	head go	**head**	begin	1	y
akz	murderer	aatiibi	aati ibi	person kill	homicide	murder	5	y
am	garlic	□□ □□□□□	□□ □□□□□	white onion	**garlic**	-	1	y
ang	dolphin	mereswīn	mere swīn	sea pig	**dolphin**	guinea pig	1	y
arz	there	□□□□	□□□ □	here you	these	here you are	0	y
ast	because	porque	por que	for that	**because**	why	1	y
av	rat	кIудияб гIункIкI	кIудияб гIункIкI	big mouse	**rat**	-	1	n
bg	enviable	завиден	за виден	in eminent	**enviable**	-	1	y
bi	us	yumipela	yu mipela	you us	**us**	virus	1	y
bn	thirteen	□□	□ □	1 three	-	-	0	y
chn	newcomer	cheechako	chee chako	new come	**newcomer**	-	1	y
ckb	newspaper	□□□□□□□□	□□□ □□□□	day letter	agenda	-	0	y
cmn	jade	猪龍	猪龍	hog dragon	**jade**	-	1	y
co	thirteen	trèdeci	trè deci	three ten	**thirteen**	thirty	1	n
crh	swan	aqquş	aq quş	white bird	**swan**	-	1	y
cs	plane angle	rovinný úhel	rovinný úhel	plane angle	**plane angle**	-	1	y
csb	triangle	trzënórt	trzë nórt	three angle	**triangle**	-	1	y
cv	eighteen	вунсаккăр	вун саккăр	ten eight	eighty	eighteenth	0	y
da	cosmodrome	rumhavn	rum havn	space harbour	**cosmodrome**	spaceport	1	y
ee	grandson	tɔgbuiyɔviŋutsu	tɔgbuiyɔvi ŋutsu	grandchild man	**grandson**	-	1	n
enm	housewife	huswif	hus wif	house woman	**housewife**	maid	1	y
eo	yolk	ovoflavo	ovo flavo	egg yellow	**yolk**	egg yolk	1	y
eu	work of art	artelan	arte lan	art work	art	artist	3	y
fi	gene therapy	geenihoito	geeni hoito	gene therapy	**gene therapy**	-	1	y
frm	devil	le diable	le diable	the devil	**devil**	-	1	y
fro	increase	encroistre	en croistre	on increase	**increase**	augment	1	y
fur	cough	tossi	tos si	cough herself	**cough**	-	1	y
ga	consider	déan trácht	déan trácht	do consider	mean	**consider**	2	y
he	pancreas	□□□□	□□ □□	breast heart	heart	**pancreas**	2	y
hi	ten thousand	□□□□□	□□□□ □	thousand zero	**ten thousand**	thousand	1	y
hsb	good evening	dobry wječor	dobry wječor	good evening	**good evening**	good afternoon	1	y
ht	seventy	swasantdis	swasant dis	sixty ten	**seventy**	sixtieth	1	y
hy	anatolia	անատոլիա	անատոլի ա	anatoli oh	**anatolia**	-	1	y
ia	boulder	petra grosse	petra grosse	stone big	**boulder**	capitate bone	1	y
ik	parents-in-law	nulliq-nulliq	nulliq nulliq	mother-in-law father-in-law	**parents-in-law**	-	1	y
io	context	kuntexto	kun texto	with text	**context**	-	1	y
it	saint george	san giorgio	san giorgio	saint george	**saint george**	-	1	y
jbo	eleven	papa	pa pa	one one	-	-	0	y
jv	indonesian	basa indonesia	basa indonesia	language indonesia	**indonesian**	-	1	y
ka	foreign affairs	საგარეო პოლიტიკა	საგარეო პოლიტიკა	foreign politics	**foreign affairs**	foreign policy	1	n
kl	south africa	afrika kujalleq	afrika kujalleq	africa south	**south africa**	-	1	y
km	thirty	□□	□ □	three zero	**thirty**	-	1	y
kpy	eight	ӈыёӷмыллыӈэн	ӈыёӷ мыллыӈэн	three five	**eight**	-	1	y
krl	wristwatch	ranneh̆čuasut	ranneh čuasut	wrist clock	**wristwatch**	-	1	y
li	adverb	biewaord	bie waord	at word	**adverb**	say	1	y
liv	seventeen	seistuoistõn	seis tuoistõn	seven teen	-	-	0	y
lld	twenty-eight	vintot	vint ot	twenty eight	**twenty-eight**	-	1	y
lus	find	hmuchhuak	hmu chhuak	find go	-	-	0	n
mel	leopard	rimau biteang	rimau biteang	tiger star	**leopard**	-	1	y
min	twenty-five	duo puluah limo duo	puluah limo	twenty five	-	-	0	n
mns	fifty	атлов	ат лов	five ten	fifteen	**fifty**	2	y
mt	light year	sena dawl	sena dawl	year light	**light year**	-	1	n
mww	question	lo lus noog lo	lus noog	word ask	-	-	0	y
nb	continent	fastland	fast land	firm country	mainland	**continent**	2	y
nl	abyss	afgrond	af grond	off ground	**abyss**	floor	1	n
nmn	wife	tâa qáe	tâa qáe	person female	female	woman	3	y
nn	arise	oppstå	opp stå	up stand	**arise**	get up	1	y
no	billionaire	milliardær	milliard ær	billion eider	**billionaire**	-	1	y
non	rome	rómaborg	róma borg	rome city	**rome**	-	1	y
nrf	red wine	rouoge vîn	rouoge vîn	red wine	**red wine**	wine	1	y
oc	coal	carbon	car bon	dear good	cheap	**coal**	2	y
ofs	take	nima	ni ma	after one	no	behind	11	y
os	monday	къуырисæр	къуыри сæр	week head	weekend	**monday**	2	y
osx	iron	īsarn	īs arn	ice eagle	**iron**	-	1	y
pih	phonecard	foenkaad	foen kaad	telephone card	**phonecard**	calling card	1	y
pjt	nipple	ipi mulya	ipi mulya	breast face	-	-	0	y
pro	long	lonc tems	lonc tems	long time	duration	long-term	0	y
ps	nine	□□□	□□ □	nine pashto alphabet	**nine**	-	1	n
rm	saddle	sela	se la	up the	**saddle**	-	1	n
ru	dry	высохнуть	вы сохнуть	you dry	wither	section	0	y
sa	garden	□□□□	□□ □□	with forest	submarine	underwater	23	y
sgs	samogitian	žemaitiu kalba	žemaitiu kalba	samogitian language	**samogitian**	-	1	n
sh	give birth	narádati	na rádati	on give birth	bear	bring	11	y
sk	independent	nezávislý	ne závislý	don't addicted	**independent**	-	1	y
sms	june	ǩie´ssmään	ǩie´ss mään	summer month	**june**	-	1	y
sne	whale	kien paos	kien paos	fish whale	**whale**	cetus	1	y
su	silkworm	hileud sutra	hileud sutra	caterpillar silk	**silkworm**	-	1	y
tet	mango	haas-fuan	haas fuan	mango fruit	**mango**	-	1	y
th	lao	□□□□□	□□ □□□	person laos	**lao**	fool	1	y
ti	husband	□□□ □□	□□□ □□	master house	landlord	master	4	y
tpi	bull	bulmakau man	bulmakau man	cow male	**bull**	male	1	y
tzm	not	□□□	□□ □	not and	**not**	grandfather	1	y
udm	russian	ӟуч кыл	ӟуч кыл	russian language	**russian**	old east slavic	1	y
uk	rape	зґвалтувати	з ґвалтувати	from rape	**rape**	-	1	y
uz	dandelion	gulqoqi	gul qoqi	flower dandelion	**dandelion**	-	1	y
vec	author	scritore	scrito re	written king	-	-	0	y
vi	war	chiến tranh	chiến tranh	war fight	**war**	warrior	1	y
vo	seventy	veldeg	vel deg	seven ten	seventeen	**seventy**	2	n
vro	estonian	eesti kiil	eesti kiil	estonia language	**estonian**	-	1	y
wa	somebody	ene sakî	ene sakî	some person	**somebody**	-	1	y
wlm	care	ar ardelw	ar ardelw	on care	defend	listen	0	y
wyi	skull	galk gawang	galk gawang	bone head	**skull**	cranium	1	y
wym	night watchman	nāhtwāhter	nāht wāhter	night guard	**night watchman**	-	1	n
za	zhuang	vahcuengh	vah cuengh	language zhuang	**zhuang**	-	1	y
zh	noise	響聲	響聲	noise sound	sound	**noise**	2	y
zza	step by step	gam gam	gam gam	step stair	**step by step**	-	1	y

Table 4: The 100-language test set (with correct English generation shown in bold).

For the f2e task, we measured both accuracy and mean rank of the model's translation hypotheses. Our method generated the correct English translation in 86/100 cases, a quite respectable performance given the great multilingual diversity of the test set, and the presence of frequently low resource languages where out-of-vocabulary compound words missing from the dictionary are quite common. Out of these cases, the correct English translation had a mean rank of 1.7 in the model's ordered list of hypotheses, indicating that most of the time the correct English trans-

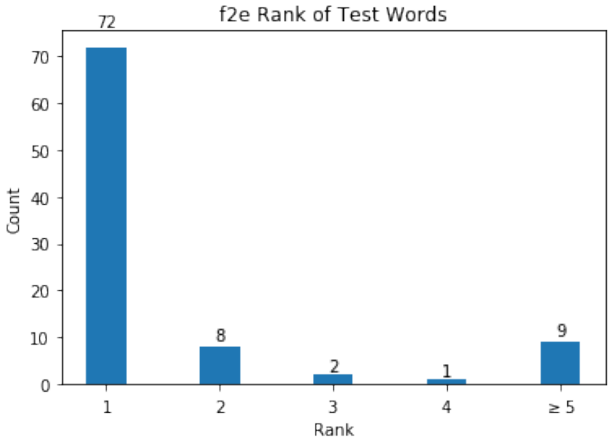


Figure 3: For the f2e task, the top hypothesis for 86/100 test words was correct.

lation was first on the list of hypotheses (Fig. 3), with the "correct" answer (directly matching the test reference) highlighted in bold. We examine a few error cases below.

In cases where the English translation could not be found, the reason was most likely that one or both of the component word translations had not been seen associated with the target English translation in at least one other language, without which it would be impossible to associate and generate the target word. Several test cases were not the top hypothesis (e.g. akz: murderer, he: pancreas, sh: give birth, and several others). However, these errors are quite reasonable (e.g. the predicted translations of akz *aatiibi* = literally *person* + *kill* were *homicide, murder* and *killer (3rd choice)*, which was a reasonable synonym of the reference translation of *murderer*. Likewise, the top hypotheses of the other "errors" are semantically quite similar to their reference translations, even synonyms of the true answer, and indeed could be scored as correct by a manual human evaluation in several cases. A third category of errors (ofs: take, and sa: garden) seem to have occurred because these test words are not actually compound words, and hence not expected to be generated via a compound-morphology-based translation model.

In the case of true unknown words, one would ideally employ our method to generate a large list of possible compound words, then filter them using a monolingual corpus or a language model. However, many of the world's languages are low-resource and do not have broad-coverage monolingual corpora available (excepting a small number of widely-translated works such as the Bible, which have a limited vocabulary). An alternative would be to ask a native speaker to verify the existence of and/or correct choice among the hypothesized these words, a relatively efficient use of native-informant time, especially when prioritized for missing dictionary concepts of high frequency in English and/or high importance in the target domain.

4.1. Dataset Analysis

Utilizing only simple direct concatenation, we were able to discover over 21,000 instances of English concepts that had were direct compounds of simpler constituents of known translation and explainable by one of the model's recipies. By extending the modeled compounding mechanism to a single-character glue or dropping the last character of the left component, our method discovered an additional 2,700 concepts successfully analyzed as compounds.

4.2. Language-Specific Compounding Mechanisms

By examining the different processes used in constructing compound words, we obtain a greater understanding of how specific languages perform compounding. Table 5 shows stereotypical language-specific patterns. For example, most languages construct compounds simply by concatenating two words directly without insertions or deletions (although often in variable order). English often uses 'i' and 's' as glue characters, while German uses 'n' and 's'. This information is not only useful for predicting whether a word is a compound, but can also be useful when generating previously unknown compound word translations into the language. A complete table and statistical analysis of these observed insertions and deletions in each language is included with our dataset, along with language-specific probabilities for the use of each type of compounding mechanism, a useful foundation for any compound-generating language model.

Certain languages like Chinese and Japanese are slightly problematic when discovering compounds using a character-dropping mechanism. For these languages, such a mechanism is not necessarily productive given that the dropped character is not merely a sound-insertion or basic-semantic-linking character but an important semantic component (that would not normally be associated with a single character in an alphabetic or even syllabic writing system). For instance, the Chinese 杀人 murder = 杀害 murder + 人 person is reasonable, but 音乐 music = 音律 tuning + 乐 music is not. Despite this caveat, single character dropping is still a widely observed and productive compounding process in these languages.

5. Related Work

Researchers have explored word compounding, though largely in the monolingual setting or on the order of a couple of languages. One multilingual effort similar to ours is MorBoComp (Guevara et al., 2006), a database of word compounds in 20 languages. The project seems to have stalled, and we were unable to access the data mentioned in their work. Our work encompasses a much larger set of languages (by a factor of 15x) and a much larger set of derived instances (even if their described database was actually available), and posits compound generation and analysis models absent from their work.

While we used straightforward but effective compound splitting algorithms, many more complicated splitting methods have been proposed, e.g. using n-gram counts (Sornlertlamvanich and Tanaka, 1996), supervised methods (Clouet and Daille, 2014), and monolingual and bilingual corpora (Koehn and Knight, 2003; Macherey et al., 2011) and could be productively employed in extensions of our work.

In contrast to several of these other works, the approach and analysis in our paper is simple yet effective in that it only

Lang	Concat	DL	Glue	Common Glues
af	0.75	0.17	0.08	g, s
br	0.85	0.12	0.03	o, r
ca	0.79	0.16	0.05	a, l
co	1.0	0	0	
com	0.5	0.5	0	
cop	0.27	0.73	0	
crh	0.81	0.17	0.01	t, b
cs	0.75	0.21	0.04	o, d
dbl	1.0	0	0	
de	0.8	0.14	0.06	n, s
dv	1.0	0	0	
ee	0.87	0.11	0.02	a
el	0.65	0.35	0	
en	0.93	0.06	0.01	i, s
eo	0.63	0.33	0.05	n, r
es	0.79	0.17	0.04	r, l
esu	1.0	0	0	
et	0.74	0.14	0.12	i, a
eu	0.69	0.24	0.06	k, l
fa	0.55	0.45	0	
fax	1.0	0	0	
ff	1.0	0	0	
fi	0.86	0.1	0.04	n, s
fy	0.5	0.42	0.08	l, t
ha	0.5	0	0.5	n, t
haw	0.57	0.37	0.06	k, h
ht	0.55	0.27	0.18	n, s
hu	0.82	0.14	0.04	i, t
ia	0.83	0.1	0.08	i, l
inh	0.12	0.88	0	
io	0.66	0.24	0.1	a, n
is	0.7	0.22	0.08	a, s
ist	0	1.0	0	
it	0.77	0.18	0.05	s, r
ja	0.32	0.68	0	
jbo	0.21	0.05	0.74	n, r
jv	0.82	0	0.18	n, p
ku	0.7	0.24	0.05	e, d
kum	0	1.0	0	
kw	0.9	0.08	0.02	l
ky	0.93	0.07	0	
la	0.72	0.22	0.06	d, c
lad	0.53	0.29	0.18	g, i
lb	0.76	0.18	0.06	e, s
lv	0.77	0.21	0.02	s, i
pl	0.72	0.22	0.05	o, d
prg	0	1.0	0	
pro	0.55	0.45	0	
ps	0.57	0.43	0	
pt	0.77	0.19	0.04	s, g
qu	0.85	0.14	0.01	y, m
raj	0	1.0	0	
rap	0.88	0.12	0	
rm	0.45	0.43	0.12	r, g
ro	0.77	0.19	0.04	r, i
rup	0.5	0.45	0.05	c, t
scn	0.55	0.33	0.12	n, g
sco	0.48	0.43	0.1	n, g
shn	1.0	0	0	
tpi	0.83	0.14	0.03	k, b
tr	0.85	0.11	0.04	l, s
uz	0.88	0.09	0.03	l, f
vai	1.0	0	0	
vec	0.57	0.32	0.11	n, r
vep	0.71	0.29	0	
vi	0.74	0.25	0.0	t, n
zh	0.34	0.66	0	

Table 5: Percentage of compound words in our dataset that were formed using the each compounding mechanism, along with common glue characters, if applicable.

requires the usually very readily available on-line dictionaries in multiple languages (e.g. via Wiktionary or Panlex) without any analyzed seed training data. Because of this, our approach does not require potentially expensive linguistic annotation, and easily extends to multiple languages, as demonstrated compellingly by our successful scaling to 329 extremely diverse languages incorporating many morphological processes and character sets.

Translation of compound words using dictionaries have been explored by Garera and Yarowsky (2008). Our approach is similar in that we use multiple bilingual dictionaries, but we study and model the compounding phenomenon in more depth as well as on a much, much larger scale, with the significant benefits of much greater novel semantic pair discovery (both via direct observation and via our transitive cluster and reordering models). In addition, we release a very large 329-language 21,000+ instance large public resource of analyzed compound words and components and statistical analyses of their processes across all languages.

In terms of applications, handling compound words well has been shown to improve machine translation, e.g. into English (Koehn and Knight, 2003) and German (Stymne et al., 2013) and has helped simplify medical text (Abrahamsson et al., 2014). We expect that our very large scale publicly distributed compound-based translation dictionaries and associated generative and analytic models will be useful for out-of-vocabulary handling in downstream machine translation systems, especially for low-resource languages.

6. Conclusion

While most languages exhibit broad-scale word formation via compounding, they often differ substantially in terms of the diverse processes by which words compound and novel concepts are realized via these compound processes. Using only freely available bilingual dictionaries and no annotated training data, we derived novel models for analyzing compound words and effectively generated novel foreign-language translations of English concepts using these models. In addition, we release a massively multilingual dataset of compound words along with their decompositions, covering over 21,000 instances in 329 languages, a previously unprecedented scale which we believe will both productively support machine translation (especially in low resource languages) and also facilitate researchers in their further analysis and modeling of compounds and compound processes across the world's languages.

7. Acknowledgments

This work was supported in part by the DARPA LORELEI program. The findings, conclusions, and opinions found in this work are those of the authors and do not necessarily represent the views of the funding agency.

8. Bibliographical References

Abrahamsson, E., Forni, T., Skeppstedt, M., and Kvist, M. (2014). Medical text simplification using synonym replacement: Adapting assessment of word difficulty to a compounding language.

Clouet, E. and Daille, B. (2014). Splitting of compound terms in non-prototypical compounding languages.

Denning, K., Kessler, B., and Leben, W. R. (2007). *English vocabulary elements*. Oxford University Press.

Garera, N. and Yarowsky, D. (2008). Translating compounds by learning component gloss translation models via multiple languages. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.

Guevara, E., Scalise, S., Bisetto, A., and Melloni, C. (2006). Morbo/comp: a multilingual database of compound words. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA).

Koehn, P. and Knight, K. (2003). Empirical methods for compound splitting. In *10th Conference of the European Chapter of the Association for Computational Linguistics*.

Macherey, K., Dai, A. M., Talbot, D., Popat, A. C., and Och, F. (2011). Language-independent compound splitting with morphological operations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1395–1404. Association for Computational Linguistics.

Sornlertlamvanich, V. and Tanaka, H. (1996). The automatic extraction of open compounds from text corpora. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Stymne, S., Cancedda, N., and Ahrenberg, L. (2013). Generation of compound words in statistical machine translation into compounding languages. *Computational Linguistics*, 39(4).

Building a Morphological Treebank for German from a Linguistic Database

Petra Steiner, Josef Ruppenhofer
Institut für Deutsche Sprache
{ruppenhofer, steiner}@ids-mannheim.de

Abstract

German is a language with complex morphological processes. Its long and often ambiguous word forms present a bottleneck problem in natural language processing. As a step towards morphological analyses of high quality, this paper introduces a morphological treebank for German. It is derived from the linguistic database CELEX which is a standard resource for German morphology. We build on its refurbished, modernized and partially revised version. The derivation of the morphological trees is not trivial, especially for such cases of conversions which are morpho-semantically opaque and merely of diachronic interest. We develop solutions and present exemplary analyses. The resulting database comprises about 40,000 morphological trees of a German base vocabulary whose format and grade of detail can be chosen according to the requirements of the applications. The Perl scripts for the generation of the treebank are publicly available on github. In our discussion, we show some future directions for morphological treebanks. In particular, we aim at the combination with other reliable lexical resources such as GermaNet.

Keywords: treebank, morphology, word structure, deep-level morphological analyses, CELEX, German

1. Introduction

German is a language with complex processes of word formation, of which the most common are compounding, derivation and conversion. The resulting lexical units usually have long orthographical forms. Moreover, many word forms have more than one combinatorially possible analysis, as in Figure 1: *Hauptbahnhof* 'central station' consists of three morphs which can be combined in two ways on the level of immediate constituents but only the first combination is the correct structure.

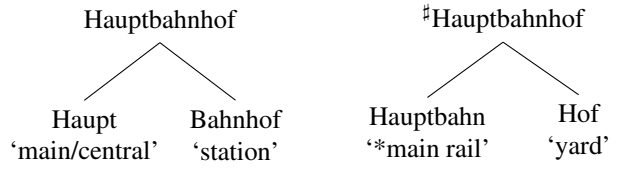


Figure 1: Ambiguous analysis of *Hauptbahnhof* 'central station'

Other word forms have ambiguous boundaries of morphs as in Figure 2 where the word form *Zugriff* 'grasp/access' is the product of a conversion process from *zugreifen* 'to grab/to grasp' and not a compound of other forms which could be erroneously recognized by a morphological analysis program.

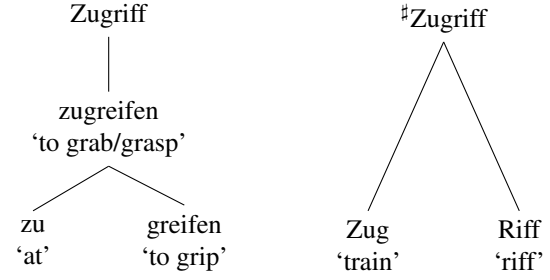


Figure 2: Ambiguous analysis of *Zugriff* 'grasp/access'

Morphological splitters for German such as Gertwol (Haapalainen and Majorin, 1995), MORPH (Hanrieder, 1996), SMOR (Schmid et al., 2004), or TAGH (Geyken and Hanneforth, 2006) generate many ambiguous analyses. Usually, this problem is approached by filtering procedures on the output analyses. For the ranking of the different morphological analyses, the geometric mean is a common score (Cap, 2014; Koehn and Knight, 2003; Steiner and Ruppenhofer, 2015). Another method is pattern matching with tokens (Henrich and Hinrichs, 2011) or comparisons of lemmas (Weller-Di Marco, 2017) or strings (Daiber et al., 2015) with corpus data. Ziering and van der Plas (2016) use normalization combined with ranking by the geometric mean. Würzner and Hanneforth (2013) apply a probabilistic context free grammar for parsing adjectives. Ma et al. (2016) apply Conditional Random Fields modeling for letter sequences. More recent approaches exploit semantic information for the ranking (Riedl and Biemann, 2016; Ziering et al., 2016). Besides Würzner and Hanneforth (2013), none of the above-mentioned authors tackled the challenge of generating deep-level analyses and with the exception of Henrich and Hinrichs (2011) who are using compounds from GermaNet (Hamp and Feldweg, 1997), all of them rely merely on corpus data as input for the heuristics and scores. However, carefully produced lexical data with morphological information would be a valuable asset for many applications in natural language processing. High quality morphologically deep-level analyses could especially be used as

1. input for statistical approaches for full morphologically parsing of German words
2. base of frequency counts for the testing of statistical hypotheses about morphological tendencies and laws
3. gold standards and test kits for morphological analyzers
4. morphological resources for morphological analyzers
5. input for textual analyses

Work on such kind of data is still in its beginning. This is shown in the following Section 2. where the related work is summarized in a concise way.
The work we present here is the generation of a morphological treebank for German. It is based on the German part of the refurbished CELEX database (Baayen et al., 1995), a manually constructed and human-supervised lexical resource. Section 3. describes this data with an emphasis on those parts which are relevant for the tree extraction process as well as the problems and flaws of the data. It also gives a sketch of the preprocessing. Section 4. presents the procedures we use. It starts with the extraction of all relevant information from the database, followed by the recursive construction of the morphological analyses. A heuristic for excluding unwanted diachronic information is presented, followed by details of the output format. The results of the script are presented in Section 5. The conclusion in Section 6. provides some further perspectives.

2. Related Work

Most German morphological data resources are restricted to lists of flat analyses. For instance, the test set of the 2009 workshop on statistical machine translation[1] was used by Cap (2014). It comprises 6,187 word tokens with splits on the upper level and interfixes removed. For example, in (1) the interfix *-s* and the hyphen have been deleted in the analysis.

(1) lexeme: *Abschreckungs-Ära* 'era of deterrence'
analysis: Abschreckung|Ära 'deterrence|era'

This is connected with some typical features of the much used morphological tool SMOR (Schmid et al., 2004). However, these interfixes are frequently marking boundaries between morphological constituents of higher levels. This is a reason why Steiner and Ruppenhofer (2015) modified the output of this tool to splits as (2).

(2) Abschreckung|s-|Ära

Henrich and Hinrichs (2011) augmented the GermaNet database with information on compound splits. This is restricted to nouns and does not provide interfixes or deep-level structures. However, in connection with this project, Steiner (2017)[2] derives deep-level structures with information on interfixes and grammatical properties from the GermaNet compounds which can be combined with analyses from CELEX.
DErivBase[3] (Zeller et al., 2013) comprises derivational families (word nests), however, the unsupervised generation of this derivational lexicon is based on heuristics of rules and string transformations. These rules do not always produce word families whose members are actually morphologically connected and the process of generation does not comply with linguistic evidence. However, the sets are produced as data for semantic (similarity) tasks and therefore do not claim grammatical correctness. Still, they contain some inconsistencies, e.g. the abridged word nest in (3) with its connection of formally similar words such as *Pause* 'pause, break' and *pausen* 'to calk, copy'. Zeller et al. (2014) assign evaluation measures to the lemma pairs of the nests for coping with this problem at least for the semantic level.

(3) $pausen_V$ 'to calk' – $abpausen_V$ 'to copy' – $pausieren_V$ 'to pause' – [...] –
$pausenlos_A$ 'without pause' – $Pause_{Nf}$ 'break' – $Zwischenpause_{Nf}$ 'short break'

The German part of the CELEX database (Baayen et al., 1995) comprises word tree information for a lexicon containing words of all parts of speech and is therefore an important source for deep-level morphological analyses of German, which are not available elsewhere. The linguistic information is combined with frequency information based on corpora (Burnage, 1995) which makes it useful for automated morphological analysis of unknown words. The original drawbacks of the German part of the database were an outdated format and use of obsolete orthographical conventions. However, these problems were tackled by Steiner (2016), so that the refurbished database yields a foundation for further exploitation. The lexicon with its 51,728 entries is relatively small but it covers a core vocabulary, similar to the small dictionary *Der kleine Wahrig* (Wahrig-Burfeind and Bertelsmann, 2007).
Shafaei et al. (2017) use the German data of CELEX for inferring derivational families which are more precise than DErivBase. The produced database DErivCELEX is drawn from the original CELEX version with its old orthographical standard[4] and therefore contains some inconsistencies and mistakes from string transformations such as (4). As some derivations of CELEX include diachronic information which became intransparent, the word nests might contain some word forms whose relatedness is rather historical than semantic, e.g. in the abridged set in (5) where constituents of *Flüssigkeit* 'fluid, liquid', *Floß* 'raft', *überflüssig* 'superfluous' and *beeinflussen* 'to influence' are all diachronically linked to *Fluss* 'river' (which is missing in DErivCELEX) and *fließen* 'to flow'.

(4) $^*blaün_V$ for *bläuen* 'to blue'

(5) $durchfließen_V$ 'to flow through' – $Floß_N$ 'raft' – $überflüssig_A$ 'superfluous' – $Zufluß_N$ 'feeder' – $unbeeinflußbar_A$ 'uninfluenceable' – [...] – $flößbar_A$ 'floatable' – $zusammenfließen_V$ 'to flow together' – $Beeinflussung_N$ 'influence' – $Zusammenfluß_N$ 'confluence' – $Flüssigkeit_N$ 'fluid, liquid' – [...] – $fließen_V$ 'to flow' – [...] – $beeinflussen_V$ 'to influence' – $beeinflußbar_A$ 'influenceable'

Just like DErivBase, DErivCELEX does not contain morphological analyses, but word family sets. DErivCELEX inherits the quality of CELEX with its manually corrected analyses; therefore it does not exhibit errors such as in (3). Shafaei et al. (2017) assert that CELEX does not treat prefixation as a form of derivation. In general, this assertion is

[1] http://www.statmt.org/wmt09/translation-task.html
[2] see https://github.com/petrasteiner/morphology
[3] http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/DErivBase/DErivBase-v2.0.zip
[4] http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/DErivBase/DErivCelex-v1.txt

unjustified, though some first constituents of verbs are classified as free morphs which Shafaei et al. (2017) consider as prefixes. The CELEX classification is justifiable from a linguistic viewpoint concerning the differences between prefixes and particles. However, as this restricts the sets of the derivational families, Shafaei et al. (2017) produce a second database based on a wider definition.
Dutch morphological analysis is covered by CELEX too. However, we are not aware of any exploitation for morphological deep-level analyses. For English, Cotterell et al. (2016) reanalyse a part of CELEX deep-level morphological analyses and thus generate 7,454 morphological parses. For other languages, there are some resources of derivational families such as in CroDeriV for Croatian (Filko and Šojat, 2017), Démonette for French (Hathout and Namer, 2016), DeriNet for Czech (Žabokrtský et al., 2016) or DerIvaTario for Italian (Talamo et al., 2016). These could be exploited for the derivation of morphological trees. However, automatic analyses are not trivial if generation rules are incomplete or multiple derivational rule paths are possible. Besides this, compounds are not considered by these lists.

3. The Refurbished CELEX-German Database

Developed in the early Nineties, the original CELEX database coding comprised a workaround for special characters. In German, these are mainly umlauts and signs such as *ß*. Furthermore, it uses an out-dated spelling convention which makes the lexicon partially incompatible with text written after 1996. For instance, the modern spelling of the original CELEX entry *Einfluß* 'influence' is *Einfluss*.
In Steiner (2016) entries such as for the lemma *Einflussbereich* 'range of influence' (6) for the orthographical part of the database and (7) for the morphologically database were aligned as in (8). Please note that these examples only present the essential and abridged information of the structure information and the morphological trees. A database with modern encoding but old spelling with *ß* was also derived as in (9). Trees as in Figure (3) could be derived directly from the database.

(6) 10236\Einflu$bereich\8\Ein-flu$-be-reich\N

(7) 10236\Einflussbereich\Einfluss+Bereich\NN\
((((ein)[V|.V],(fliess)[V])[V])[N],
((be)[N|.N],(Reich)[N])[N])[N]

(8) Einflussbereich\Einfluss+Bereich\NN\
((((ein)[V|.V],(fließ)[V])[V])[N],
((be)[N|.N],(Reich)[N])[N])[N]

(9) Einflußbereich\Einfluß+Bereich\NN\
((((ein)[V|.V],(fließ)[V])[V])[N],
((be)[N|.N],(Reich)[N])[N])[N]

However, trees of this kind have some gaps: they do not contain categorial information for affixes nor for the derivation process, e.g. the noun *Einfluss* 'influence'. Therefore, simple transformations of the data would yield only incomplete derivations.

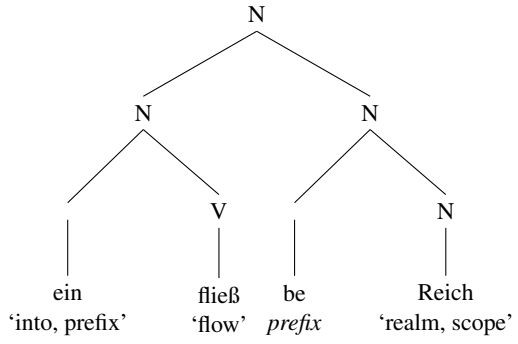


Figure 3: Rudimentary morphological analysis of *Einflussbereich* 'range of influence'

Another drawback is the missing information on the infinitive stem. While in (8) this would be *-en* for *fließen*, in the analysis of *Abenddämmerung* 'evening dawn = nightfall' in (10) there is an elision of the schwa of the infinitive stem *dämmern* 'to dawn'.
Some derivations in the German CELEX database provide diachronic information which is correct but often unwanted for many applications, for example in (11) (*Schnellzug* 'fast(-speed) train') where *Zug* 'train' is diachronically derived from *ziehen* 'to draw', see Figure 4. This analysis is completely opaque from a synchronic point of view. On the other hand, some derivations such as the ablaut changes between *fließen* and *Einfluss* in Figure 3 or *gehen* 'to go' and *Gang* 'gait,path,aisle' in *Abgangszeugnis* 'leaving certificate' (12) could be of interest.

(10) 111\Abenddämmerung\Abend+Dämmerung\NN
((Abend)[N],((dämmer)[V],(ung)[N|V.])[N])[N]

(11) 34419\Schnellzug \schnell+Zug\AN\
((schnell)[A],((zieh)[V])[N])[N]

(12) 207\Abgangszeugnis\Abgang+s+Zeugnis\NxN\
((((ab)[V|.V],(geh)[V])[V])[N],(s)[N|N.N],
((zeug)[V], (nis)[N|V.])[N])[N]

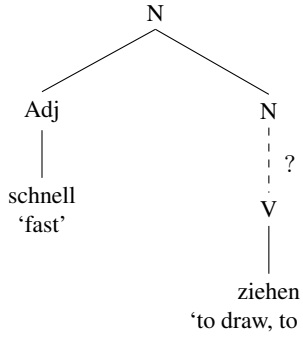


Figure 4: Rudimentary and questionable morphological analysis of *Schnellzug* 'fast train'

Interfixes can be inferred from the database entry. In Figure 5, the interfix is represented as an affix (*x*) within the categories of the immediate constituent structure. In the

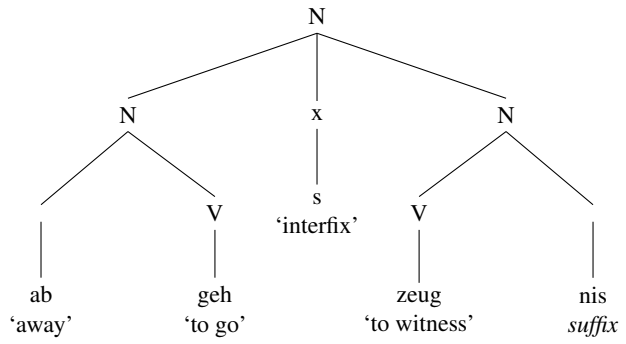


Figure 5: Rudimentary morphological analysis of *Abgangszeugnis* ‘leaving certificate’

CELEX entries they are part of the categorial description of the immediate structures, such as *NxN* within the example (12). As every complex entry has such information on the immediate constituents and their categories, it is possible to collect this information recursively and top-down from the CELEX entries.

Though most of its data are flawless, the original CELEX database contains some mistakes which were not treated by the refurbishment of Steiner (2016) which covered only changes of coding and spelling. We found

- missing constituents and missing part-of-speech information within the morphological trees
- missing constituents within the field of immediate constituency information
- inconsistent morphological analyses, such as *Kenntnisnahme* ‘notice, attention’ in Figure 6 which should have been analysed as a conversion (Zusammenrückung), similar to *Maßnahme* ‘measure, step’ which in CELEX is analyzed as resulting from a conversion of *maßnehmen* ‘to take measures’.

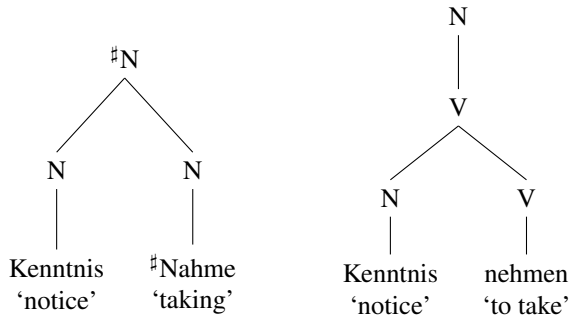


Figure 6: Erroneous and correct morphological analysis of *Kenntnisnahme* ‘notice’

We augmented the script for the transformation to a modern standard by 18 additional rules, which covered 65 instances before we could use the data for extracting the morphological trees. We are aware of the fact that we could not find all mistakes. The Perl script *OrthCELEX.pl* for the refurbishment and correction of the German CELEX data is available on github.[5]

4. Procedures

The extraction of the CELEX-German treebank is based on the refurbished and corrected database which we have described in the last section. Figure 7 shows the dataflow and the main procedures.

We do not produce one single treebank, but leave it to the users which format and information they choose for the trees they intend to build. For example, semantic word nests might require less diachronic information than finding anaphora in texts. Conversions can be of interest or not. The generating script provides some parameters for refinements and output formats. We first extract all the information which could be required and then build the trees recursively and top-down according to the options.

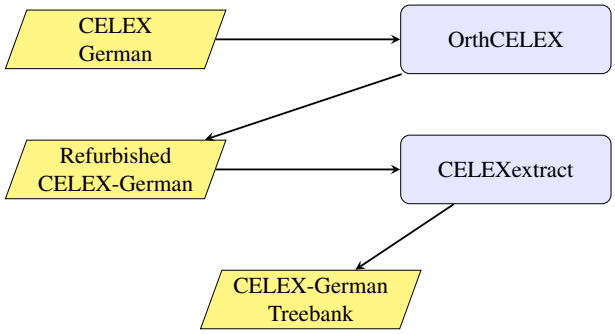


Figure 7: The dataflow from CELEX-German to the CELEX Treebank

4.1. Data Extraction

We start with extracting all relevant information. Some forms can be assigned more than one part of speech as in (13), or more than one gender as in 14, or they are morphologically ambiguous as in (15) and Figure 8. Therefore we build an inverted index of all lemmas.

(13) a. aber ‘but, conj’
b. aber ‘really, intensifier’

(14) a. Band ‘volume/book, noun’
b. Band ‘band (music), noun’
c. Band ‘ribbon/strap, noun’

(15) a. erzen ‘made out of ore, bronze, adj’
b. erzen ‘to address by *er*, verb’

We extract all immediate constituents and also their categories, then we internally add the infinitive forms of the verbs which are included within these entries. This is necessary for finding these forms within the inverted index of the entries. Also we refurbish the German syntactic database of CELEX to the modern standard and extract the parts of speech of the entries.

As the users can choose if they would like to generate not just compounds and derivatives but also conversions, we extract the relevant information for this word-formation class too.

[5] see https://github.com/petrasteiner/morphology

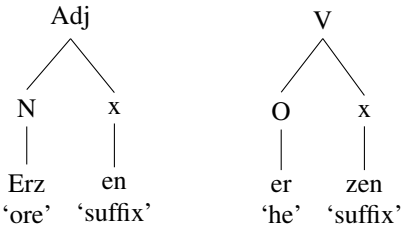


Figure 8: The ambiguity of the form *erzen*

4.2. Building the Trees

For each entry of the morphological database, the procedure starts from the list of its immediate constituents and recursively collects all information from the entries of the constituents. Algorithm 1 presents the recursive process. Table 1 shows the parameters.

4.3. Prevention of Diachronic Information

Diachronic information, as in example (11) and Figure 4 with *Zug* ‘train’ being diachronically derived from *ziehen* ‘to draw’, can be of interest, however, for many applications it is considered as unnecessary or even disturbing. Therefore, the script permits users to choose a threshold of similarity within the range of [0:1] which is compared to a measure using the Levenshtein distance.

For accepting or rejecting two parts of words as morphologically related, the procedure will cut two forms f_1, f_2 with length l_1 and l_2 to the strings s_1, s_2 of the smaller length $(min(l_1, l_2))$ and calculate the Levensthein distance (LD) of these. Special characters such as *ä* or *ß* are transformed to *a* and *ss*, uppercase characters to lowercase. Then the quotient of both values is compared to a threshold *t* as in (16):

$$\frac{LD(s_1, s_2)}{min(l_1, l_2)} < t \tag{16}$$

For example, in (17) both the derived form (e.g. $f_1 = Zug$) and its component (e.g. $f_2 = zieh$) are cut to the smaller size of these forms in lowercase letters. In this case, that yields the strings $s_1 = zug$ and $s_2 = zie$. After this, the quotient of $LD(s_1, s_2)$ and the smaller length is compared to the threshold. (17) shows that the analysis for this case would be interrupted for a threshold below $0.\overline{6}$. A value of 1 would show total dissimilarity, one of 0 absolute similarity.

$$\frac{LD(zug, zie)}{min(3, 4)} = \frac{2}{3} \tag{17}$$

In case that singular variations were needed, we also added a small list of exceptions.

4.4. Output Formats

Our tool supports various output formats. Table 1 lists the optional parameters which are available. The depth of the morphological trees can be determined, same as including the analysis of conversions or which linguistic information should be provided, e.g. the parts of speech and classes of the bound morphs. If the threshold for the Levenshtein-based measure is defined, the top-down generation of the morphological trees will be stopped for elements which are more dissimilar to each other than permitted. For the output style, the user can choose parentheses or a notation with pipe bars (”|”) for the splits on the same level.

Algorithm 1: Building the morphological treebank

```
Input: CELEX-German revised
Output: A Morphological Treebank
initialization of parameters: depths of analysis,
  levenshtein threshold, linguistic information,
  parts of speech, style of output;
forall entries of CELEX do
    if entry is complex or a conversion then
        foreach constituent of entry do
            if constituent is simplex
              or depth of analysis reached then
                retrieve linguistic information/PoS
                as required;
                return linguistic information and
                  constituent
            end
            else
                foreach part of constituent do
                    depth of analysis++;
                    analysedeeper part with
                      parameters and depth;
                    return result of analysedeeper
                end
            end
        end
    end
end

 sub analysedeeper part (parameters and level)
  if part is simplex
  or depth of analysis reached
  then
    retrieve linguistic information/PoS as required;
    return linguistic information and part
end
else
    foreach subpart of part do
        analysedeeper subpart
          if levenshtein threshold and
          analysedeeper subpart is dissimilar then
            skip deeper analysis;
            retrieve linguistic information/PoS as
              required;
            return subpart
        end
        else
            return result of analysedeeper subpart
        end
    end
end
```

	Parameters
•	Depth of analysis for compounds and derivatives
•	Analysis of conversions
•	Depth of analysis for conversions
•	Lingustic information of the morphs
•	Threshold for the Levenshtein measure
•	Style of format

Table 1: Parameters for the tree generation

5. Results

The list of all word-formation products of the German database (compounds, derivatives, results of conversions) comprises 40,097 entries.

5.1. Coverage

We tested the coverage of this treebank on the *Korpus Magazin Lufthansa Bordbuch (MLD)* which is part of the DeReKo-2016-I (Institut für Deutsche Sprache, 2016) corpus[6]. It is an in-flight magazine with articles on traveling, consumption and aviation. For the tokenization, we enlarged and costumized the tokenizer by Dipper (2016) for our purposes. Multi-word units were automatically identified based on the multi-word dataset which we had augmented before. The xml-annotated data comprises 276 texts with 5,202 paragraphs, 16,046 sentences and 260,115 tokens. The number of word-form types is 38,337. Of these types, 5,435 are included in the CELEX-derived treebank. If we add all entries, including also the simplex forms, the overlap of the types is 8,622. We are comparing a list of lemmas with a list of word forms, this means that not every full form can be covered. Therefore, the overlap is a good start, especially as (longer) word-formation products could be analyzed in combination with a word splitter for flat structures.

5.2. Output

The following shows the entries of *Einflussbereich*, *Schnellzug*, and *Abgangszeugnis*. For the parameter setting of all linguistic information, the notation with |, and a Levenshtein threshold of 0.6, the results are presented in (18), for parenthesis notation and no restrictions on diachronic conversions in (19) and for a flat representation of the immediate constituents see (20).

(18)
```
Einflussbereich
(*Einfluss_N*
  (*einfließen_V*
    ein_x|
    fließen_V))|
(*Bereich_N*
  be_x|
  Reich_N)

Schnellzug
 schnell_A|
 (Zug_N)

Abgangszeugnis
(*Abgang_N*
   (*abgehen_V*
     ab_x|
     gehen_V))|
 s_x|
 (*Zeugnis_N*
   zeugen_V|
   nis_x)
```

(19)
```
Einflussbereich
 (*Einfluss_N*
  (*einfließen_V*
    (ein_x)
    (fließen_V)))
 (*Bereich_N*
   (be_x)
   (Reich_N))

Schnellzug
 (schnell_A)
 (*Zug_N*
    ziehen_V)

Abgangszeugnis
 (*Abgang_N*
   (*abgehen_V*
    (ab_x)
    (gehen_V)))
 (s_x)
 (*Zeugnis_N*
   (zeugen_V)
   (nis_x))
```

(20)
```
Einflussbereich
 Einfluss_N|
 Bereich_N

Schnellzug
 schnell_A|
 Zug_N

Abgangszeugnis
 Abgang_N|
 s_x|
 Zeugnis_N
```

Figures 9 and 10 show the complete analyses for *Einflussbereich* and *Abgangszeugnis* with all intermediate constituents.

6. Conclusion and Further Perspectives

This article introduces to the first German morphological treebank. Its form and output can be determined by the user of the Perl script *CELEXextract.pl* which is available on our repository.[7]

The possible analyses comprise compounds, derivatives and conversions of different depths and linguistic information as is required. The current database is relatively small,

[6] see Kupietz et al. (2010) for further information

[7] see https://github.com/petrasteiner/morphology

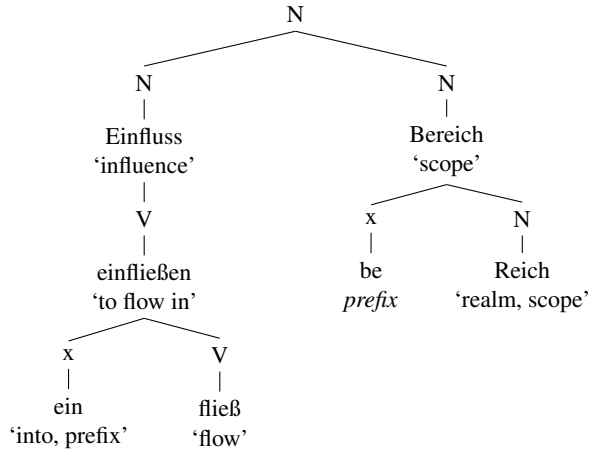


Figure 9: Complete morphological analysis of *Einflussbereich* ‘range of influence’

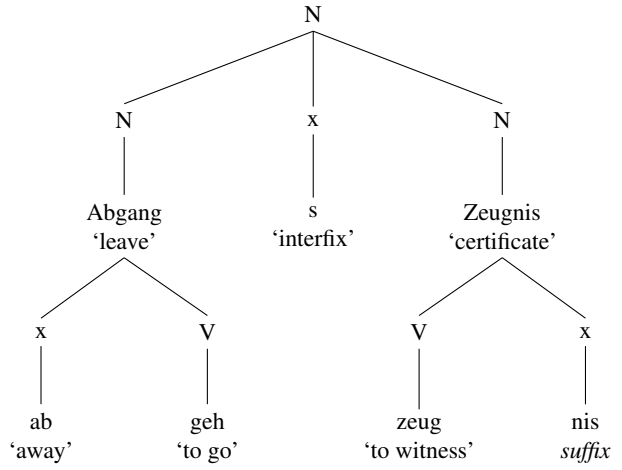


Figure 10: Complete morphological analysis of *Abgangszeugnis* ‘leaving certificate’

however, it will be augmented by other sources. This work has already started. Recently, Steiner (2017) combined the splits of the nominal compounds of GermaNet (Henrich and Hinrichs, 2011) with the more fine-grained analyses of CELEX’s basic vocabulary. While the GermaNet compounds on their own yield about 68,000 trees, here the recursive production of the morphological trees stops as soon as derivatives are reached. But merging both resources results in a German Treebank of ca. 100,000 analyses comprising the processes of compounding and derivation for each entry.

Compared to automatically inducing morphological resources, which then have to be cleaned and/or evaluated, the effort of using manually produced data for the induction of deep morphological analyses is relatively small and the effect is rewarding.

On the foundation of the existing database, more complex words can be analyzed in combination with a morphological splitter for flat structures. This method enlarges the coverage by the combinatorial potential of language and will avoid the abundance of ambiguous word analyses.

7. Acknowledgements

The author was supported by the German Research Foundation (DFG) under grant RU 1873/2-1. We would like to thank the reviewers for their valuable feedback.

8. Bibliographical References

Burnage, G. (1995). CELEX: A Guide for Users. In Harald Baayen, et al., editors, *The CELEX Lexical Database (CD-ROM)*. Linguistic Data Consortium, Philadelphia, PA.

Cap, F. (2014). *Morphological processing of compounds for statistical machine translation*. Ph.D. thesis, Universität Stuttgart.

Cotterell, R., Kumar, A., and Schütze, H. (2016). Morphological segmentation inside-out language processing. In Jian Su, et al., editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2325–2330. The Association for Computational Linguistics.

Daiber, J., Quiroz, L., Wechsler, R., and Frank, S. (2015). Splitting compounds by semantic analogy. In *Proceedings of the 1st Deep Machine Translation Workshop*, pages 20–28. ÚFAL MFF UK.

Dipper, S. (2016). Tokenizer for German.

Filko, M. and Šojat, K. (2017). Expansion of the derivational database for Croatian. In *First Workshop on Resources and Tools for Derivational Morphology (DeriMo)*.

Geyken, A. and Hanneforth, T. (2006). TAGH: A Complete Morphology for German based on Weighted Finite State Automata. In *Finite State Methods and Natural Language Processing. 5th International Workshop, FSMNLP 2005, Helsinki, Finland, September 1-2, 2005. Revised Papers*, volume 4002, pages 55–66. Springer.

Haapalainen, M. and Majorin, A. (1995). Gertwol und morphologische Disambiguierung für das Deutsche. In *Proceedings of the 10th Nordic Conference on Computational Linguistics, Helsinki, Finland*.

Hamp, B. and Feldweg, H. (1997). GermaNet - a lexical-semantic net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Hanrieder, G. (1996). MORPH - ein modulares und robustes Morphologieprogramm für das Deutsche in Common Lisp. In Roland Hauser, editor, *Linguistische Verifikation - Dokumentation zur Ersten Morpholymics 1994*, pages 53–66. Niemeyer, Tübingen.

Hathout, N. and Namer, F. (2016). Giving lexical resources a second life: Démonette, a multi-sourced morpho-semantic network for French. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1084–1091, Paris, France. European Language Resources Association (ELRA).

Henrich, V. and Hinrichs, E. (2011). Determining immediate constituents of compounds in GermaNet. In *Proceedings of the International Conference Recent Advances in*

Natural Language Processing 2011, pages 420–426. Association for Computational Linguistics.

Koehn, P. and Knight, K. (2003). Empirical methods for compound splitting. In *Proceedings of the tenth conference of the European Chapter of the Association for Computational Linguistics-Volume 1*, pages 187–193. Association for Computational Linguistics.

Kupietz, M., Belica, C., Keibel, H., and Witt, A. (2010). The German reference corpus DeReKo: A primordial sample for linguistic research. In Nicoletta Calzolari, et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 1848–1854, Valletta, Malta. European Language Resources Association (ELRA).

Ma, J., Henrich, V., and Hinrichs, E. (2016). Letter sequence labeling for compound splitting. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 76–81, Berlin, Germany. Association for Computational Linguistics.

Riedl, M. and Biemann, C. (2016). Unsupervised compound splitting with distributional semantics rivals supervised methods. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pages 617–622. Association for Computational Linguistics.

Schmid, H., Fitschen, A., and Heid, U. (2004). SMOR: A German computational morphology covering derivation, composition and inflection. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. European Language Resources Association (ELRA).

Shafaei, E., Frassinelli, D., Lapesa, G., and Padó, S. (2017). DErivCELEX: Development and evaluation of a German derivational morphology lexicon based on CELEX. In *Proceedings of the DeriMo workshop*, Milan, Italy.

Steiner, P. and Ruppenhofer, J. (2015). Growing trees from morphs: Towards data-driven morphological parsing. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology, GSCL 2015, University of Duisburg-Essen, Germany, 30th September - 2nd October2015*, pages 49–57.

Steiner, P. (2016). Refurbishing a morphological database for German. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*, pages 1103–1108. European Language Resources Association (ELRA).

Steiner, P. (2017). Merging the Trees. Building a Morphological Treebank for German from Two Resources. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories, January 23–24, 2018*, TLT'16, pages 146–160, Prague, Czech Republic.

Talamo, L., Celata, C., and Bertinetto, P. M. (2016). Derivatario: An annotated lexicon of italian derivatives. *Word Structure*, 9(1):72–102.

Wahrig-Burfeind, R. and Bertelsmann, G. L. (2007). *Der kleine Wahrig: Wörterbuch der deutschen Sprache [der deutsche Grundwortschatz in mehr als 25000 Stichwörtern und 120000 Anwendungsbeispielen; mit umfassenden Informationen zur Wortbedeutung und detaillierten Angaben zu grammatischen und orthografischen Aspekten der deutschen Gegenwartssprache]*. Wissen Media Verlag.

Weller-Di Marco, M. (2017). Simple compound splitting for german. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 161–166, Valencia, Spain. Association for Computational Linguistics.

Würzner, K.-M. and Hanneforth, T. (2013). Parsing morphologically complex words. In *Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing, FSMNLP 2013, St. Andrews, Scotland, UK, July 15-17, 2013*, pages 39–43.

Žabokrtský, Z., Sevcikova, M., Straka, M., Vidra, J., and Limburská, A. (2016). Merging data resources for inflectional and derivational morphology in Czech. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA).

Zeller, B., Šnajder, J., and Padó, S. (2013). DerivBase: Inducing and evaluating a derivational morphology resource for German. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1201–1211. Association for Computational Linguistics.

Zeller, B., Padó, S., and Šnajder, J. (2014). Towards semantic validation of a derivational lexicon. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1728–1739, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Ziering, P. and van der Plas, L. (2016). Towards unsupervised and language-independent compound splitting using inflectional morphological transformations. In *Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 644–653. Association for Computational Linguistics.

Ziering, P., Müller, S., and van der Plas, L. (2016). Top a splitter: Using distributional semantics for improving compound splitting. In *Proceedings of the 12th Workshop on Multiword Expressions*, pages 50–55, Berlin, Germany, August. Association for Computational Linguistics.

9. Language Resource References

Baayen, Harald and Piepenbrock, Richard and Gulikers, Léon. (1995). *The CELEX lexical database (CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania, 1.0, ISLRN 204-698-863-053-1.

Institut für Deutsche Sprache. (2016). *Deutsches Referenzkorpus / Archiv der Korpora geschriebener Gegenwartssprache 2016-I (Release from 31.03.2016)*. Institut für Deutsche Sprache, 1.0.

Baselines and Test Data for Cross-Lingual Inference

Željko Agić **Natalie Schluter**
Department of Computer Science
IT University of Copenhagen
Rued Langgaards Vej 7, 2300 Copenhagen S, Denmark
zeag@itu.dk nael@itu.dk

Abstract

The recent years have seen a revival of interest in textual entailment, sparked by i) the emergence of powerful deep neural network learners for natural language processing and ii) the timely development of large-scale evaluation datasets such as SNLI. Recast as natural language inference, the problem now amounts to detecting the relation between pairs of statements: they either contradict or entail one another, or they are mutually neutral. Current research in natural language inference is effectively exclusive to English. In this paper, we propose to advance the research in SNLI-style natural language inference toward multilingual evaluation. To that end, we provide test data for four major languages: Arabic, French, Spanish, and Russian. We experiment with a set of baselines. Our systems are based on cross-lingual word embeddings and machine translation. While our best system scores an average accuracy of just over 75%, we focus largely on enabling further research in multilingual inference.

Keywords: natural language inference, cross-lingual methods, test data

1. Introduction

Natural language processing is marking a very recent resurgence of interest in textual entailment. Now revamped as **natural language inference** (NLI) by Bowman et al. (2015) with their SNLI dataset, the task of differentiating contradictory, entailing, and unrelated pairs of sentences (Fig. 1) has entertained a large number of proposals.[1] The timely challenge lends itself to various deep learning approaches such as by Rocktäschel et al. (2015), Parikh et al. (2016), or Wang et al. (2017), which mark a string of very notable results.

Yet, the SNLI corpus is in English only. As of recently, it includes more test data from multiple genres,[2] but it remains exclusive to English. Following Bender (2009) in seeking true language independence, we propose to extend the current NLI research beyond English, and further into the majority realm of low-resource languages.

Since training data is generally unavailable for most languages, work on transfer learning is abundant for the basic NLP tasks such as tagging and syntactic parsing (Das and Petrov, 2011; Ammar et al., 2016). By contrast, the research in cross-lingual entailment is not as plentiful (Negri et al., 2013). To the best of our knowledge, at this point there are no contributions to SNLI-style cross-lingual inference, or for that matter, work on languages other than English at all.

Contributions. In the absence of training data for languages other than English, we propose a set of baselines for **cross-lingual neural inference**. We adapt to the target languages either by i) employing multilingual word embeddings or alternatively by ii) translating the input sentebces into English.

We create **multilingual test data** to facilitate evaluation by manually translating $4 \times 1{,}332$ premise-hypothesis sentence pairs from the English SNLI test data into four other major languages: Arabic, French, Russian, and Spanish.

premise	Female gymnasts warm up before a competition.
entailment	Gymnasts get ready for a competition.
contradiction	Football players practice.
neutral	Gymnasts get ready for the biggest competition of their life.

Figure 1: Example sentence 4-tuple from the SNLI test set, lines 758–760.

We also experiment with automatic translations of the SNLI test data to serve as a proxy for large-scale evaluations in the absence of manually produced data.

2. Cross-Lingual Inference

Following the success of neural networks in SNLI-style inference, we take the neural attention-based model of Parikh et al. (2016) as our starting point. To date, their system remains competitive with the current state of the art. As their attention model is based solely on word embeddings, and is independent of word order, it is particularly suitable for the baseline we present here: a purely multi-lingual embeddings based cross-lingual NLI system. Moreover, their approach is computationally much leaner than most competitors, making it a fast and scalable choice.[3]

In short, the Parikh et al. (2016) model sends sentence pairs, i.e., premises and hypotheses, through a neural pipeline that consists of three separate components:

i) **ATTENTION:** Scores combinations of pairs of words across input sentence pairs. Scores of these word pairs are given by a feed-forward network with ReLU activations that is assumed to model a homomorphic function for linear-time computation. Attention weights for phrases softly aligned with a word are obtained by summing their component vectors each factored by their normalized score.

[1] https://nlp.stanford.edu/projects/snli/

[2] https://repeval2017.github.io/shared/

[3] For more details, see the original paper, and an illustrative overview of the model: https://explosion.ai/blog/deep-learning-formula-nlp.

ii) **COMPARISON:** Word vectors and their aligned phrase counterparts are compared and combined into a single vector using a feed-forward neural network.

iii) **CONCATENATION:** A network that sums over the above output vectors for each input sentence, concatenates this representation and feeds it through a final feed-forward network followed by a linear layer.

To be trained, the model expects SNLI annotations, and an ideally very large vocabulary of distributed word representations.

In this paper, we have at our disposal only a large training corpus of English NLI examples, but a distinct language in which we want to predict for NLI: the target language. We train the system described above on the English training set. We exploit the fact that the system is purely embeddings-based and train with multilingual embeddings for a set of languages including English and the prediction language. Multilingual embeddings are sets of word embeddings generated for multiple languages where the embeddings from the union of these sets are meant to correspond to one another semantically independent of the language the words the embeddings correspond to actually belong. At prediction time, we can safely use the embeddings of the target language.

Mapping. One method for obtaining multilingual word embeddings is to apply the translation matrix technique to a set of monolingual embeddings (Mikolov et al., 2013a) with the aid of a bilingual dictionary containing the source-target word pairs. The method works by finding a transformation matrix from the target language monolingual embeddings to the English monolingual embeddings that minimizes the total least-squared error. This transformation matrix can then be used on words not seen in the bilingual dictionary.

Multilingual embeddings. If parallel sentences or even just parallel documents are available for two or more languages, we can use this data to embed their vocabularies in a shared representation. For example, through an English-Russian parallel corpus we would represent the words of the two languages in a shared space.

There are several competing approaches to training word embeddings over parallel sentences. In this paper, we experiment with four.

BICVM: The seminal approach by Hermann and Blunsom (2014) for inducing bilingual compositional representations from sentence-aligned parallel corpora only.[4]

INVERT: Inverted indexing over parallel corpus sentence IDs as indexing features, with SVD dimensionality reduction on top, following Søgaard et al. (2015) in the recent implementation by Levy et al. (2017).[5] Instead of embedding just language pairs, this method embeds multiple languages into the same space. It is thus distinctly multilingual, rather than just bilingual.

RANDOM: Our implementation of the approach by Vulić and Moens (2016) whereby bilingual SGNS embeddings of Mikolov et al. (2013b) are trained on top of merged pairs of parallel sentences with randomly shuffled tokens.

RATIO: Similar to RANDOM, except the tokens in bilingual sentences are not shuffled, but inserted successively by following the token ratio between the two sentences.

[4] https://github.com/karlmoritz/bicvm

[5] https://bitbucket.org/omerlevy/xling_embeddings/

	ara	**fra**	**spa**	**rus**
eng to ...	25.58	55.80	39.65	30.31
... to **eng**	37.48	46.90	44.04	31.17

Table 1: Machine translation quality (BLEU) for translating the test data from and into English.

Machine translation. One alternative to adapting via shared distributed representations is to use machine translation.

If high-quality translation systems are readily available, or if we can build them from abundant parallel corpora, we can simply translate any input to English and run a pre-trained English NLI model over it. Moreover, we can translate the training data and train target language models similar to Tiedemann et al. (2014) in cross-lingual dependency parsing.

The MT approach only lends itself to medium- to high-density languages. The mapping requires only the monolingual data and bilingual dictionaries, while the bilingual embeddings need parallel texts or documents, both of which are feasible for true low-resource languages.

3. Test Data

The SNLI data are essentially pairs of sentences—premises and hypotheses—each paired with a relation label: contradiction, entailment, or neutral. We had human experts manually translate the first 1,332 test pairs from English into Arabic, French, Russian, and Spanish. We copied over the original labeling of relations, and the annotators manually verified that they hold. That way we can directly evaluate the NLI performance for these five languages.

Further, we translated our test sets into English by Google Translate for our MT-based system as it adapts through translation and thus expects input in English. We also automatically translated the 1,332 original English sentences into our new test languages to check how well we can approximate the "true" accuracies by using translated test data. This way we can facilitate cross-lingual NLI evaluations on a larger scale.

The BLEU scores for the two translation directions are given in Table 1, where we see a clear split by similarity as the translations tend to be better between English, French, and Spanish, and worse outside that group.

4. Experiment

Our experiment involves adapting a neural NLI classifier through multilingual word embeddings and machine translation. We run the Kim et al. (2017) implementation of the attention-based system of Parikh et al. (2016).[6] All models are trained for 15 epochs and otherwise with default settings. While this system typically peaks at over 100 epochs,

[6] https://github.com/harvardnlp/struct-attn

	ara	eng	fra	spa	rus
map to eng					
FASTTEXT	55.75	79.74	51.64	51.94	48.59
bilingual					
BICVM	56.82	76.26	59.03	59.48	54.30
RANDOM	57.35	77.42	63.21	61.01	56.97
RATIO	54.46	78.10	58.64	60.09	51.18
multilingual					
INVERT	54.76	75.10	62.60	60.55	54.76
translation					
FASTTEXT	72.28	–	77.23	75.93	76.54
GLOVE	75.86	–	80.05	78.75	79.59

Table 2: Overall accuracy of the cross-lingual approaches for the target languages and English.

we sacrifice some accuracy to provide more data points in the comparison given the time constraints.

We set the dimensionality to 300 for all our embeddings. Other than that, they are trained with their default settings. In mapping we use the pretrained FASTTEXT vectors[7] for all five languages (Bojanowski et al., 2016). We map the target language embeddings to English as Mikolov et al. (2013a), using the Dinu et al. (2014) implementation[8] and Wiktionary data.[9]

We train our bilingual embeddings on the UN corpus (Ziemski et al., 2016). The corpus covers English and the four target languages with 11M sentences each. The sentences are aligned across all five languages. The Moses tokenizer[10] (Koehn et al., 2007) was used to preprocess the corpus and the test data for training and evaluation.

In the MT approach, we only experiment with translating the input, and not with translating the training data due to time constraints. There, we use two English SNLI models: one with FASTTEXT and the other with GLOVE 840B embeddings (Pennington et al., 2014).[11]

Results. We report the overall accuracy and F_1 scores for the three labels. Table 2 gives the overall scores of our cross-lingual NLI approaches. In general, the more resources we have, the better the scores: Training bilingual embeddings surpasses the mapping to English, while translating to English using a top-level MT system tops the adaptation via embeddings.

The mapping to English works slightly better for Arabic than for the other languages, and scores an average of 52%. The RANDOM bilingual embeddings top their group with an average accuracy of 59.6% followed by INVERT at 58.1%, while RATIO and BICVM are below at 56.1 and 57.4%. The MT approach expectedly tops the table at 75.5% accuracy.

In Table 3 we see that our best bilingual embeddings system RANDOM has a preference for entailment, with ca 9% in F_1 over the other two labels, which makes sense for a model aimed at capturing semantic similarity. This also holds true for the original Parikh et al. (2016) evaluation on English.

	con	ent	neu
ara	55.82	64.17	50.91
fra	57.63	68.73	61.72
spa	55.78	66.98	57.80
rus	56.83	60.61	53.29

Table 3: F_1 scores for **con**tradiction, **ent**ailment, and **neu**tral for our best system, RANDOM.

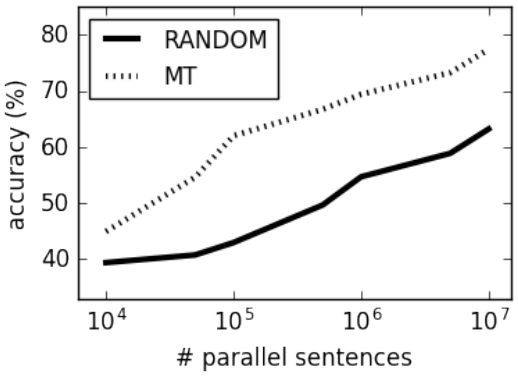


Figure 2: French NLI accuracy in relation to parallel corpus size for RANDOM embeddings.

We report all our English scores as a sanity check. In 100 training epochs, Parikh et al. (2016) score 86.8% with GLOVE 840B as their top score, while we mark 83.4% in 15 epochs. With the significantly smaller FASTTEXT embeddings we reach an accuracy of 79.7%. The multilingual embeddings average at 76.7% for English, where RATIO peaks at 78.1%, likely as its sequential shuffling of parallel texts most closely captures the English sentence structure.

Discussion. Figure 2 plots a learning curve for the French RANDOM approach. We see that its accuracy steadily increases by adding more parallel data into building the bilingual embeddings. As a side note, the MT-based system benefits if the English side of the embeddings grows in size and quality. The figure points out that i) adding more data benefits the task, and that ii) the accuracy of our RANDOM approach stabilizes at around 1M parallel sentences. As per Søgaard et al. (2015) most language pairs can offer no more than 100k sentence pairs, this puts forth a challenge for future cross-lingual NLI learning research.

Replacing the manually prepared test sets with the ones automatically translated from English underestimates the true accuracy by absolute -2.57% on average. The higher the translation quality, the better the estimates we observe: While the difference is around -1% for French and Spanish, it is -7% for Arabic. Still, in proxy evaluation, as with our MT-based adaptation approach in general, we exercise caution: SNLI sentences are image captions, mostly $\leq$15 words long and thus relatively easy to translate (cf. Bowman et al. (2015), Fig. 2) in comparison to, e.g., newspaper text.

[7] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[8] http://clic.cimec.unitn.it/~georgiana.dinu/down/

[9] https://dumps.wikimedia.org/

[10] https://github.com/moses-smt/mosesdecoder/

[11] https://nlp.stanford.edu/projects/glove/

5. Related Work

Prior to SNLI, there has been work in cross-lingual textual entailment using parallel corpora (Mehdad et al., 2011) and lexical resources (Castillo, 2011), or crowdsourcing for multilingual training data by Negri et al. (2011). We also note two shared tasks, on cross-lingual entailment with five languages (Negri et al., 2013) and English relatedness and inference (Marelli et al., 2014).

Cer et al. (2017) provide multilingual evaluation data within a shared task in semantic textual similarity. There, paired snippets of text are evaluated for their degree of equivalence, and could thus be treated as a fine-grained proxy for SNLI-style evaluations.

SNLI is the first large-scale dataset for NLI in English (Bowman et al., 2015), two orders of magnitude larger than any predecessor. It was recently expanded with test data for multiple genres of English to allow for cross-domain evaluation.[12] Prior to our work, there have been no SNLI-style cross-lingual methods or evaluations.

6. Conclusions

We have proposed the first set of cross-lingual approaches to natural language inference, together with novel test data for four major languages. In experiments with three types of transfer systems, we record viable scores, while at the same time exploring the scalability of cross-lingual inference for low-resource languages.

We are actively enlarging the test data and introducing new languages. Our multilingual test sets and word embeddings are freely available.[13]

7. References

Ammar, W., Mulcaire, G., Ballesteros, M., Dyer, C., and Smith, N. (2016). Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.

Bender, E. M. (2009). Linguistically naïve != language independent: Why NLP needs linguistic typology. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 26–32.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *EMNLP*, pages 632–642.

Castillo, J. J. (2011). A wordnet-based semantic approach to textual entailment and cross-lingual textual entailment. *International Journal of Machine Learning and Cybernetics*, 2(3):177–189.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *SemEval*, pages 1–14.

[12] https://www.nyu.edu/projects/bowman/multinli/

[13] https://bitbucket.org/nlpitu/xnli

Das, D. and Petrov, S. (2011). Unsupervised part-of-speech tagging with bilingual graph-based projections. In *ACL*, pages 600–609.

Dinu, G., Lazaridou, A., and Baroni, M. (2014). Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*.

Hermann, K. M. and Blunsom, P. (2014). Multilingual models for compositional distributed semantics. In *ACL*, pages 58–68.

Kim, Y., Denton, C., Hoang, L., and Rush, A. M. (2017). Structured attention networks. *arXiv preprint arXiv:1702.00887*.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *ACL*, pages 177–180.

Levy, O., Søgaard, A., and Goldberg, Y. (2017). A strong baseline for learning cross-lingual word embeddings from sentence alignments. In *EACL*, pages 765–774.

Marelli, M., Bentivogli, L., Baroni, M., Bernardi, R., Menini, S., and Zamparelli, R. (2014). Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *SemEval*, pages 1–8.

Mehdad, Y., Negri, M., and Federico, M. (2011). Using bilingual parallel corpora for cross-lingual textual entailment. In *ACL*, pages 1336–1345.

Mikolov, T., Le, Q., and Sutskever, I. (2013a). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Negri, M., Bentivogli, L., Mehdad, Y., Giampiccolo, D., and Marchetti, A. (2011). Divide and conquer: Crowdsourcing the creation of cross-lingual textual entailment corpora. In *EMNLP*, pages 670–679.

Negri, M., Marchetti, A., Mehdad, Y., Bentivogli, L., and Giampiccolo, D. (2013). Semeval-2013 task 8: Cross-lingual textual entailment for content synchronization. In *SemEval*, pages 25–33.

Parikh, A., Täckström, O., Das, D., and Uszkoreit, J. (2016). A decomposable attention model for natural language inference. In *EMNLP*, pages 2249–2255.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Rocktäschel, T., Grefenstette, E., Hermann, K. M., Kočiskỳ, T., and Blunsom, P. (2015). Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Søgaard, A., Agić, v., Martínez Alonso, H., Plank, B., Bohnet, B., and Johannsen, A. (2015). Inverted indexing for cross-lingual nlp. In *ACL*, pages 1713–1722.

Tiedemann, J., Agić, Ž., and Nivre, J. (2014). Treebank

translation for cross-lingual parser induction. In *CoNLL*, pages 130–140.

Vulić, I. and Moens, M.-F. (2016). Bilingual distributed word representations from document-aligned comparable data. *Journal of Artificial Intelligence Research*, 55:953–994.

Wang, Z., Hamza, W., and Florian, R. (2017). Bilateral multi-perspective matching for natural language sentences. *arXiv preprint arXiv:1702.03814*.

Ziemski, M., Junczys-Dowmunt, M., and Pouliquen, B. (2016). The united nations parallel corpus v1.0. In *LREC*, pages 3530–3534.

CATS: A Tool for Customized Alignment of Text Simplification Corpora

Sanja Štajner[1], **Marc Franco-Salvador**[2], **Paolo Rosso**[3], **Simone Paolo Ponzetto**[1]
[1] DWS Research Group, University of Mannheim, Germany
[2] Symanto Research, Nuremberg, Germany
[3] PRHLT Research Center, Universitat Politècnica de València, Spain
{sanja, simone} @informatik.uni-mannheim.de, marc.franco@symanto.net, prosso@prhlt.upv.es

Abstract

In text simplification (TS), parallel corpora consisting of original sentences and their manually simplified counterparts are very scarce and small in size, which impedes building supervised automated TS systems with sufficient coverage. Furthermore, the existing corpora usually do not distinguish sentence pairs which present full matches (both sentences contain the same information), and those that present only partial matches (the two sentences share the meaning only partially), thus not allowing for building customized automated TS systems which would separately model different simplification transformations. In this paper, we present our freely available, language-independent tool for sentence alignment from parallel/comparable TS resources (document-aligned resources), which additionally offers the possibility for filtering sentences depending on the level of their semantic overlap. We perform in-depth human evaluation of the tool's performance on English and Spanish corpora, and explore its capacities for classification of sentence pairs according to the simplification operation they model.

Keywords: text simplification, tools and resources, sentence similarity

1. Introduction

Automated text simplification (ATS) has the goal of automatically transforming sentence structure and lexical choices in a way that it provides better understanding and wider accessibility to large audiences. The main obstacle for successful supervised ATS is the scarcity and limited size of parallel TS corpora which would contain original sentences and their manual simplifications. The parallel TS corpus for Brazilian Portuguese, compiled for the purposes of the *PorSimples* project (Aluísio et al., 2008) contains around 4,500 aligned sentences, and the parallel TS corpus for Spanish, compiled for the purposes of the *Simplext* project (Saggion et al., 2015) contains only around 1,000 aligned sentences. The largest existing TS comparable corpora is the English Wikipedia – Simple English Wikipedia (EW–SEW), consisting of 170,000 sentence pairs (Kauchak, 2013), or 150,000 full matches and 130,000 partial matches in the newer version (Hwang et al., 2015). In both cases, the sentences were automatically aligned from comparable English Wikipedia and Simple English Wikipedia articles. However, the use of EW–SEW dataset for modeling TS has been disputed (Amancio and Specia, 2014; Štajner et al., 2015; Xu et al., 2015) for several reasons: (1) the simplified articles are not necessarily direct simplifications of the original articles; (2) the quality of simplifications is not checked; (3) the dataset does not cover sentence splitting which is one of the most common operations in text simplification.

The Newsela corpora[1] of document-aligned news texts, manually simplified at four different simplification levels have been freely available for a few years for research purposes. These corpora have several advantages over the EW–SEW dataset (Xu et al., 2015; Štajner et al., 2017): (1) simplified texts present direct simplifications of the original articles; (2) simplification was performed by trained human editors, following strict guidelines; (3) by sentence-aligning those corpora one can get training material for simplifications at various levels, i.e. train different simplification models depending on the intended reader group; and (4) they provide comparable training material in two languages, English and Spanish. At the beginning of 2016, the Newsela corpora contained around 2,000 original news articles in English and around 250 original news articles in Spanish (both with their corresponding manually simplified versions at four different simplification levels).

The current state-of-the-art systems for automatic sentence-alignment of original and manually simplified text are the Greedy Structural WikNet (GSWN) method (Hwang et al., 2015) used for sentence-alignment of original and simple English Wikipedia, and the HMM-based (using Hidden Markov Model and Viterbi algorithm) method (Bott and Saggion, 2011) used for sentence-alignment of the Spanish Simplext corpus (Saggion et al., 2015). The HMM-based method can be applied to any language as it does not require any language-specific resources. It is based on two hypothesis: (H1) that the original order of information is preserved, and (H2) that every 'simple' sentence has a corresponding 'original' sentence. The GSWN method does not assume H1 or H2, but it only allows for '1-1' sentence alignments (which is very restricting for TS) and it is language-dependent as it requires the English Wiktionary[2].

In this paper, we present a freely available tool for sentence- and paragraph-alignment from document-aligned TS corpora, the CATS (Customized Alignment for Text Simplification) tool.[3] The tool offers two main functionalities:

1. **CATS-Align**: sentence- or paragraph- alignment of parallel texts; and

[1] https://newsela.com/

[2] https://www.wiktionary.org/

[3] The CATS tool and documentation can be downloaded from: https://github.com/neosyon/SimpTextAlign

2. **CATS-Measure**: three different sentence (or paragraph) similarity measures which can be further used to filter retrieved sentence/paragraph pairs for customised modeling of text simplification operations.

The CATS-Align has two main advantages over the state-of-the-art GSWN method:

- CATS is language-independent and resource light. Two out of three similarity metrics that CATS require only pre-trained word embeddings in the language for which is to be used, while the third similarity metric does not even require word embeddings (it is based on character n-gram matching).
- The GSWN method only allows for '1-1' sentence alignments, while CATS-aligns additionally offers a possibility for building a dataset which covers sentence splitting ('1-n' alignments).

Similarly to the HMM-based method, our alignment methods assume the hypothesis H2. We provide them in both variants, using the hypothesis H1 and without it (Section 2.2). The detailed human evaluation of our methods and the HMM-based method (Section 3) for both English and Spanish showed that our methods are significantly better, especially when aligning sentences from distant complexity levels.

The CATS tool was released together with our previous paper (Štajner et al., 2017). In that work, we were purely interested in augmenting the parallel datasets for training ATS systems and we performed an intrinsic evaluation of the tool only on the English part of Newsela corpora, and an extrinsic evaluation by using newly aligned dataset in a PBSMT approach to English ATS.

Here we build on our previous work by describing the full potential of the CATS tool: (1) for automatic sentence-alignment of both English and Spanish corpora; and (2) for automatic filtering of aligned sentence pairs according to the simplification operation: content deletion, information addition, and paraphrasing without significant semantic change.

In Section 2 we present different modes of the CATS tool. In Section 3, we present a detailed human evaluation and error analysis of the CATS-Align on both English and Spanish Newsela corpora. We evaluate CATS-Measure automatically on the 'gold standard' English Wikipedia dataset for classification of sentence pairs in three classes (*full matches*, *partial matches*, and *no match*) in Section 4. Finally, in Section 5, using the labels assigned by human annotators on the Newsela dataset during the human evaluation of CATS-Align, we test whether the CATS-Measure can be used to automatically classify the automatically-aligned sentence pairs (aligned by CATS-Align) into four classes depending on the type of simplification operation they model.

2. The CATS

Our CATS software can work in two different regimes:

- **CATS-Measure** for providing three different similarity measures that can be applied either on paragraph or sentence level;
- **CATS-Align** for choosing best paragraph or sentence alignments in a given document-aligned corpus.

2.1. CATS-Measure

CATS-Measure provides similarity measures for three different sentence/paragraph similarity methods:

1. **C3G:** The Character N-Gram (CNG) (Mcnamee and Mayfield, 2004) similarity model (with *n* set to 3) with log TF-IDF weighting (Salton and McGill, 1986).
2. **WAVG:** The continuous skip-gram model (Mikolov et al., 2013b) of the TensorFlow toolkit[4] on the English Wikipedia. For each text snippet (i.e. sentence or paragraph, depending on the task) we average its word vectors to obtain a single representation of its content. This setting has shown good results in other NLP tasks (e.g. for selecting out-of-the-list words (Mikolov et al., 2013a), or for language variety identification (Franco-Salvador et al., 2015)).
3. **CWASA:** The Continuous Word Alignment-based Similarity Analysis (CWASA) model (Franco-Salvador et al., 2016) was initially proposed for plagiarism detection with excellent results. Unlike the WAVG method, CWASA does not average word vectors and was thus proposed as more adequate for long texts.

In all three methods, similarity between the vectors is calculated using the cosine similarity. In WAVG and CWASA methods we use 300-dimensional vectors calculated with the continuous skip-gram model. We use the September 2016 Wikipedia dumps as input to train the English vectors. The model uses negative sampling, context windows of size 10, and 20 negative words for each sample. For Spanish, we use the freely available pretrained 300-dimensional vectors obtained using the skip-gram model with negative sampling on a large collection of various Spanish corpora with a total of approximately 1.5 billion words (Cardellino, 2016).

2.2. CATS-Align

For aligning sentences or paragraphs from a document-aligned TS corpora, CATS-Align offers two different alignment strategies (MST and MST-LIS) depending on whether we assume the hypothesis H1 (see Section 1) that the simplified text presents the information in the same order as the original text:

- **Most Similar Text (MST):** Having a set of 'simple' text snippets S, a set of 'complex' text snippets C, and one of the similarity methods (Section 2.1), MST compares similarity scores of all possible pairs (s_i, c_j), and aligns each $s_i \in S$ with the closest one in C.
- **MST with Longest Increasing Sequence (MST-LIS):** MST-LIS uses the hypothesis H1. It first uses the MST strategy, and then postprocess the output by extracting – from all obtained alignments – only those alignments $l_i \in L$, which contain the longest increasing sequence of offsets j_k in C.

[4] https://www.tensorflow.org/

Ex.	Original	Simplified
1a	Hand parts take up to 10 hours to print and another couple of hours to assemble with elastic cords to keep the hands open.	Hand pieces take up to 10 hours to print.
1b	Hand parts take up to 10 hours to print and another couple of hours to assemble with elastic cords to keep the hands open.	Putting them together takes another couple of hours.
2a	With one his wife bought him for Father's Day, sheets of colored plastic, and free designs and advice found online, he made a hand for about $20	Chi made his first 3-D hand with a printer his wife bought him for Father ś Day.
2b	With one his wife bought him for Father's Day, sheets of colored plastic, and free designs and advice found online, he made a hand for about $20	He found free designs online.
2c	With one his wife bought him for Father's Day, sheets of colored plastic, and free designs and advice found online, he made a hand for about $20	With them, he printed a hand for about $20.

Table 1: Examples of '1-n' alignments obtained by CATS-Align on the Newsela corpora (Newsela, 2016), which can be used either for modeling sentence compression (each example separately) or sentence splitting (by merging examples 1a–1b, and merging examples 2a–2c).

In order to allow for '1–n' alignments (i.e. sentence splitting), we allow for repeated offsets of C in L.
The 'simple' text snippets not contained in L are included in the set U of unaligned snippets.
Finally, we align each $u_m \in U$ by restricting the search space in C to those offsets of 'complex' text snippets that correspond to the previous and the next aligned 'simple' snippets. For instance, if $L = \{(s_1, c_4), (s_3, c_7)\}$ and $U = \{s_2\}$, then the search space for the alignments of s_2 is reduced to $\{c_4 ... c_7\}$.
We denote the MST-LIS alignment strategy by adding '*' to the name of the similarity method (e.g. C3G*).

2.2.1. Modeling Sentence Splitting and Compression

In both alignment strategies (MST and MST-LIS), we allow the same original sentence to be aligned with multiple simple sentences, in order to allow for modeling both sentence splitting and sentence compression later on. This is one of the important differences between our alignment models and the state-of-the-art GSWN method, which only allows '1-1' alignment and thus does not offer a possibility for building a dataset which covers sentence splitting ('1-n' alignments). An example of our customisable alignment tool is presented in Table 1. While each of the separate examples (1a–2c) can be later used for modeling sentence compression, by merging the examples 1a–1b and 2a–2c, we also build good training materials for modeling sentence splitting operations.

2.2.2. Two-Step Alignment

Additionally, the CATS-Align offers two-step alignment option, by first performing paragraph-alignment, and then sentence-alignment within each pair of aligned paragraphs. In this option, paragraphs and sentences can be aligned by any of the six previously mentioned strategies (three similarity methods times two alignment strategies), and not necessarily the same one. Two-step C3G alignment (C3G-2s) has shown best results in the extrinsic evaluation when used for building ATS systems (Štajner et al., 2017).

3. Human Evaluation on Newsela Datasets

We randomly selected 10 original English articles and 10 original Spanish articles, together the four corresponding simpler versions (at different levels of simplification) for each of them, and sentence-aligned them with seven different alignment strategies offered by the CATS tool: C3G, C3G*, CWASA, CWASA*, WAVG, WAVG*, C3G-2step, and the HMM-based alignment tool (Bott et al., 2012). Then we asked two native speakers of English (first trained on additional 3 original articles and their corresponding simplified versions) and two native speakers of Spanish (first trained in the same manner) to classify the obtained sentence pairs (a total of approx. 3,500 sentence-pairs for each language) in one of the four classes:

- 3: full match (full semantic overlap),
- 2: partial match (partial semantic overlap where the original sentence contains less information than the simplified sentence)
- 1: partial match (partial semantic overlap where the original sentence contains more information than the simplified sentence),
- 0: no match (no semantic overlap).

While sentence pairs with full matches can be used to model paraphrasing, sentence pairs with partial matches can be used to model deletions (class '1' where the original sentence contains more information than the simplified sentence) or additions (class '2' where the original sentence contains less information than the simplified sentence). Several examples from different classes are presented in Table 2. As can be seen, the CATS tool can successfully align sentences with full semantic overlap which differ only by one lexical/phrasal substitution (the second and the third example in Table 2), as well as those which are much stronger paraphrases of each other (the first example in Table 2). It can also align the sentences which have only partial semantic overlap (examples 4a, 4b, and 5 in Table 2).

3.1. Results of Human Evaluation

The results of this human evaluation are presented in Tables 3 and 4, as a percentages of different classes/matches. Given the human effort needed for such evaluation, we focused only on three level pairs: aligning the sentences from

Ex.	Original	Simplified	Class	Sim.
1	**After focusing on** the latest **artificial** limb technology**, he began to hunt** for **more basic** options.	**He looked into** the latest **prosthetic** limb technology **and began hunting** for **cheaper, less complicated** options.	3	0.52
2	Like many researchers, **entrepreneurs** and even artists in recent years, he turned to the 3-D printer.	Like many researchers, **businesspeople** and even artists in recent years, he turned to the 3-D printer.	3	0.80
3	**With** one his wife bought him for Father's Day, sheets of colored plastic, and free designs and advice found online, he made a hand for about $20.	**By using** one his wife bought him for Father's Day, sheets of colored plastic, and free designs and advice found online, he made a hand for about $20.	3	0.95
4a	**A non-profit** group called Women On 20s**, formed to convince President Barack Obama to put a woman's image on the $20 note, already has done some polling.**	**There is a** group called Women On 20s.	1	0.45
4b	**A non-profit group called Women On 20s, formed to convince President Barack Obama to put** a woman's **image** on the $20 **note, already has done some polling.**	**It wanted** a woman's **picture** on the $20 **bill**.	1	0.26
5	The plastic comes out in layers.	**They melt and** the plastic comes out in layers.	2	0.82
6	**Lew has said that Hamilton's image will remain part of** the new $10 bill.	The new $10 bill **will have the picture of a woman, he said**.	0	0.47
7	**These are some of the candidates to be** the first woman on **U.S. currency notes** in more than a **century**.	**Who will be** the first woman on **American money** in more than a **100 years**?	0	0.30

Table 2: Examples of different classes of alignments obtained by CATS-Align on the Newsela corpora (Newsela, 2016), together with their similarity scores obtained by C3G-2s alignment strategy. Differences between original and simplified sentences are presented in bold.

the original articles (Level 0) and the first simpler level (Level 1), aligning the sentences for the original articles (Level 0) and the simplest articles (Level 4), and aligning the sentences from the two simplest levels (Level 3 and Level 4). Due to the nature of simplification operations needed to be applied between levels 0 and 1, and those needed between levels 0 and 4, we expect a greater lexical and n-gram overlap between the sentences needed to align between levels 0 and 1, than those sentences needed to align between levels 0 and 4. Furthermore, we are interested in exploring whether the success rate of the alignment tool stays stable whenever we align two neighbouring levels, thus taking into account both 0–1 and 3–4 alignments. Finally, we investigate whether the success rate stays stable across the two languages.

3.1.1. Comparison with the State of the Art

In both languages and on all level pairs, the CATS alignments were able to find higher number of full and partial matches than the state-of-the-art HMM alignment method (Tables 3 and 4). The differences in the percentage of full matches found by the CATS alignments and the HMM method are particularly pronounced when we align 0-1 levels (up to 9.4% difference on the English dataset, and up to 10.8% difference on the Spanish dataset). The differences in the percentages of partial matches modeling deletion (*Part-Del*) between the CATS alignments and the HMM method while aligning 0-1 levels are noticeable on the Spanish dataset (up to 13.4% difference), while there is no much difference on the English dataset (only up to 3.3% difference). In aligning other level pairs (3-4 and 0-4) the differences in the percentages of partial matches modeling deletion were significant regardless the language.

3.1.2. The Influence of Hypothesis H1

We noticed that the use of hypothesis H1 reduces the percentage of full matches regardless of language, level pairs, and similarity measure. However, it sometimes increases the number of partial matches which model deletion (see Tables 3 and 4).

3.1.3. CATS Alignments across Languages

When comparing the performances of CATS alignments across the two languages, we find that alignment of 0-1 levels yields in slightly higher percentage of full matches on the English dataset than on the Spanish dataset, but at the cost of having lower percentage of partial matches modeling deletion (*Part-Del*). When aligning Level 3 with Level 4, we find similar percentage of full matches in both languages and a higher number of partial matches (deletions) on the English dataset. In the case of aligning Level 0 with Level 4, we also have a higher percentage of full matches on the English dataset than on the Spanish dataset, in addition to a higher percentage of partial matches. However, the differences in the percentage of full matches and partial matches between the two languages might not reflect the performances of the system on those languages but rather the nature of simplifications performed on the Newsela arti-

Align	Method	Class			
		Full	Part-Add	Part-Del	No
0-1	C3G	**69.6%**	2.8%	23.8%	**3.9%**
0-1	C3G*	67.4%	2.8%	**24.3%**	5.5%
0-1	CWASA	69.1%	2.8%	**24.3%**	**3.9%**
0-1	CWASA*	66.9%	2.8%	**24.3%**	6.1%
0-1	WAVG	**69.6%**	2.2%	**24.3%**	**3.9%**
0-1	WAVG*	67.4%	2.2%	**24.3%**	6.1%
0-1	HMM	60.2%	2.8%	21.0%	16.0%
3-4	C3G	44.5%	0.9%	**32.7%**	**21.8%**
3-4	C3G*	42.7%	0.9%	33.2%	23.2%
3-4	CWASA	**45.5%**	1.4%	31.3%	**21.8%**
3-4	CWASA*	42.7%	0.9%	31.3%	25.1%
3-4	WAVG	44.1%	1.4%	32.2%	22.3%
3-4	WAVG*	41.2%	0.9%	**32.7%**	25.1%
3-4	HMM	38.9%	0.9%	23.7%	36.5%
0-4	C3G	**10.0%**	0.5%	43.6%	46.0%
0-4	C3G*	6.2%	0.0%	**48.8%**	**45.0%**
0-4	CWASA	9.5%	0.5%	33.6%	56.4%
0-4	CWASA*	6.6%	0.0%	35.1%	58.3%
0-4	WAVG	9.5%	0.5%	45.0%	**45.0%**
0-4	WAVG*	6.6%	0.0%	43.6%	49.8%
0-4	HMM	4.7%	0.0%	20.1%	74.4%

Table 3: Distribution of different sentence-alignments according to the human evaluation on the English Newsela corpora. For each pair of levels, the highest percentage of *full* and *part-del* matches, and the lowest percentage of *no* matches are shown in bold.

Align	Method	Class			
		3 (Full)	2 (Add)	1 (Del)	0 (No)
0-1	C3G	61.4%	3.4%	**31.8%**	**3.4%**
0-1	C3G*	55.7%	3.4%	30.7%	10.2%
0-1	CWASA	61.4%	3.4%	29.5%	5.7%
0-1	CWASA*	55.7%	3.4%	30.7%	10.2%
0-1	WAVG	**62.5%**	2.3%	28.4%	6.8%
0-1	WAVG*	55.7%	2.3%	**31.8%**	10.2%
0-1	HMM	50.6%	1.1%	18.4%	29.9%
3-4	C3G	**44.8%**	1.0%	**26.7%**	**27.6%**
3-4	C3G*	41.9%	1.0%	25.7%	31.4%
3-4	CWASA	43.8%	1.9%	21.0%	33.3%
3-4	CWASA*	41.9%	1.9%	24.8%	31.4%
3-4	WAVG	41.9%	1.9%	21.9%	34.3%
3-4	WAVG*	41.0%	1.9%	23.8%	33.3%
3-4	HMM	38.1%	0.0%	18.1%	43.8%
0-4	C3G	**3.8%**	0.0%	37.1%	59.0%
0-4	C3G*	1.9%	0.0%	**43.8%**	**54.3%**
0-4	CWASA	**3.8%**	1.0%	35.2%	60.0%
0-4	CWASA*	1.0%	0.0%	35.2%	63.8%
0-4	WAVG	**3.8%**	1.0%	35.2%	60.0%
0-4	WAVG*	2.9%	0.0%	26.7%	70.5%
0-4	HMM	1.0%	0.0%	10.5%	88.6%

Table 4: Distribution of different sentence-alignments according to the human evaluation on the Spanish Newsela corpora.

cles in those two languages, i.e. it is possible that the simplification in Spanish leads to more sentence splittings thus reflected in higher number of partial matches, while the simplification in English leads to more paraphrasing without sentence splitting and thus more full matches. The performances of the CATS-Align are rather reflected in number of *no matches*, which according to the results in Tables 3 and 4 indicate similar performances of CATS-Align for both languages in the case of aligning 0-1 levels, and a slightly better performances of the tool on the English than on the Spanish dataset in the case of 3-4 and 0-4 alignments.

3.2. Error Analysis

We found that over 50% of original sentences from Level 0 get split into two or more (up to even five sentences) simple sentences when simplifying into Level 4. This sometimes results in low scores (*no match*) by human evaluation of isolated sentence pairs, although all simple sentences that correspond to the same original sentence, when seen together, perfectly match the original sentence (Table 5). In other words, when treated separately, some of the sentences aligned between levels 0 and 4 present false negatives, as they will later be merged together for training ATS systems, where they will then represent good training material for sentence splitting. These type of errors could be avoided by performing human evaluation on already merged sentences (those that together model sentence splitting), but in that case we would not have the count of correct deletions and additions, which might be useful for tasks other than simplification, or for modeling these specific text simplification operations.

4. Automatic Evaluation on Wikipedia

The 'gold standard' Wikipedia dataset for sentence-alignment (Hwang et al., 2015) contains pairs of sentences and their 'gold label' (*full match*, *partial match*, or *no match*) with the same meaning as in our manual evaluation task. As the C3G, CWASA and WAVG methods output similarity score for each sentence pair, we use them to predict the labels for the sentence pairs in the Wikipedia dataset. Table 6 shows the F_1-measures obtained by our systems (using different combinations of sentence similarity measures and classification algorithms) and the state-of-the-art GSWN system, as well as several baselines used by Hwang *et al.* (2015) on two classification tasks: classifying between *Good&GoodPartial* matches vs. *Others* (Task 1), and between *Good* matches vs. *Others* (Task 2). The HMM-method requires full texts to use the H1 hypothesis and thus cannot be successfully applied to these tasks.

Detailed results of our various classification methods on Tasks 1 and 2, presenting precision (P), recall (R) and F_1-measure (F) on the *Good & GoodPartial* class (Task1) or the *Good* class (Task 2), are presented in Table 7. We observe similar behaviour of our similarity metrics in both tasks; the C3G method obtains significantly better recall than the word-embedding-based methods (CWASA and WAVG), while CWASA and WAVG obtain better precision than the C3G. All three methods combined together significantly boost both recall and F_1-measure, significantly outperforming all previously proposed methods on Task 1, and

Ex.	Simplified (Level 4)	Original (Level 0)
1	Todos los estudiantes tienen que hacer el examen SAT para ser admitidos en la universidad.	En otras palabras, si los estudiantes afroamericanos con las puntuaciones del SAT más bajas que las de sus compañeros blancos también reciben peores calificaciones en la universidad, o se cambian con más frecuencia a carreras "más fáciles", entonces, para empezar, seguramente estaban menos preparados para la universidad.
2	Algunas personas opinan que los estudiantes de color están menos preparados para entrar a la universidad.	En otras palabras, si los estudiantes afroamericanos con las puntuaciones del SAT más bajas que las de sus compañeros blancos también reciben peores calificaciones en la universidad, o se cambian con más frecuencia a carreras "más fáciles", entonces, para empezar, seguramente estaban menos preparados para la universidad.
3	Dicen esto cuando los estudiantes afroamericanos tienen calificaciones más bajas que las de sus compañeros blancos en ese examen.	En otras palabras, si los estudiantes afroamericanos con las puntuaciones del SAT más bajas que las de sus compañeros blancos también reciben peores calificaciones en la universidad, o se cambian con más frecuencia a carreras "más fáciles", entonces, para empezar, seguramente estaban menos preparados para la universidad.
4	También, cuando reciben calificaciones más bajas en la universidad o se cambian a carreras "menos difíciles".	En otras palabras, si los estudiantes afroamericanos con las puntuaciones del SAT más bajas que las de sus compañeros blancos también reciben peores calificaciones en la universidad, o se cambian con más frecuencia a carreras "más fáciles", entonces, para empezar, seguramente estaban menos preparados para la universidad.

Table 5: An example of sentence pairs which obtained score 0 (no match) each, but in fact present a good example of an '1–n' alignment obtained by CATS-Align C3G method on the Newsela corpora (Newsela, 2016).

Approach	*Task1*	*Task2*
C3G+CWASA+WAVG	**.643**	.705
C3G+CWASA	.621	.680
C3G+WAVG	.602	.691
CWASA+WAVG	.506	.664
C3G	.612	.695
CWASA	.490	.671
WAVG	.481	.650
GSWN (Hwang et al., 2015)	.607	**.712**
Unconst.WordNet (Hwang et al., 2015)	.515	.636
Ordered Vec.Space (Hwang et al., 2015)	.415	.564
Unconstr. Vec.Space (Hwang et al., 2015)	.431	.550

Table 6: F_1-measures on Task1 (*Good & Good Partial vs. Others*) and Task2 (*Good vs. Others*). The best results for each task are shown in bold.

CATS measures	Task 1			Task 2		
	P	R	F	P	R	F
C3G+CWASA+WAVG	.808	**.534**	**.643**	.829	**.614**	**.705**
C3G+CWASA	.780	.516	.621	.780	.603	.680
C3G+WAVG	.760	.498	.602	.791	**.614**	.691
CWASA+WAVG	**.829**	.364	.506	.808	.563	.664
C3G	.777	.505	.612	.803	.574	.669
CWASA	.792	.355	.490	.827	.502	.625
WAVG	.791	.346	.481	**.830**	.527	.645

Table 7: Detailed results on Task 1 (*Good & Good Partial vs. Others*) and Task2 (*Good vs. Others*). The best results for each measure are shown in bold.

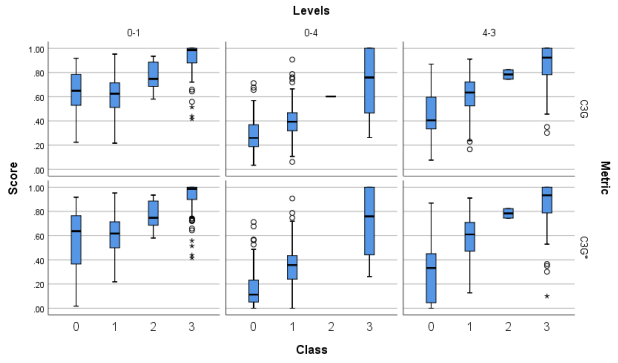
obtaining comparable results to the state of the art on Task 2 (see Table 6). Here is important to mention that, unlike the current state-of-the-art method on Task 2 (Hwang et al., 2015), our methods are **language-independent**, **resource-light**, and allow for retrieving material for sentence splitting (**allow for '1–n' matches**).

5. Automatic Classification of Alignments

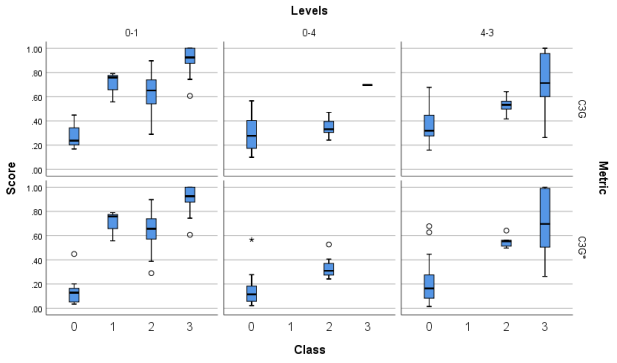
Finally, we explore if the similarity measures provided by CATS-Measure can be used to classify the aligned sentence pairs according to the type of simplification operation they model.

5.1. Distribution of Similarity Scores

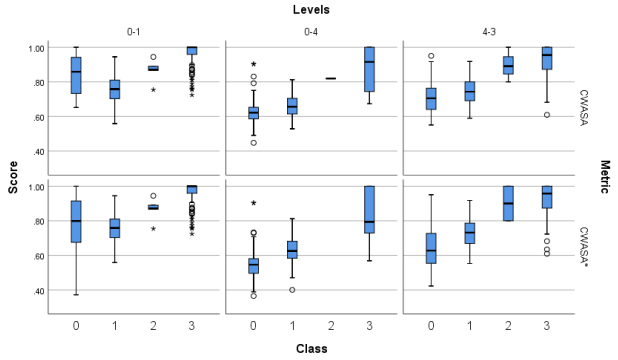
We first explore the distribution of similarity scores across different transformation types and different text levels on both English and Spanish human annotated datasets (Figure 1). As can be seen, all six sentence similarity metrics (C3G, C3G*, CWASA, CWASA*, WAVG, WAVG*) seem to have better discriminatory power between the four classes (0–3) on the Spanish dataset than on the English dataset (discriminatory power seen as the overlap of boxplots for different classes), with the C3G and C3G* being the best among the six metrics. Here is important to mention that the metrics do not have to be able to distinguish between the classes 1 (insertion) and 2 (deletion). To discriminate between those two classes (in the case of similar metrics scores) we can use the difference in the sentence length between the two sentences, i.e. deletions and insertions should lead to the opposite sign when we subtract the length of the original sentence (in words) from the length of the simplified sentence (in words). The distributions of sentence similarity scores (Figure 1) indicate that we can expect better classification results for Spanish than for English experiments.



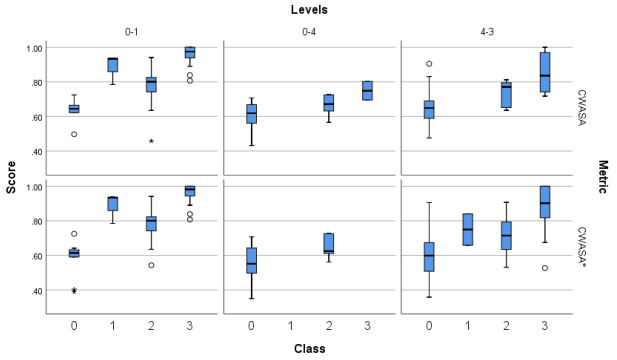
(a) English (C3G vs. C3G*)



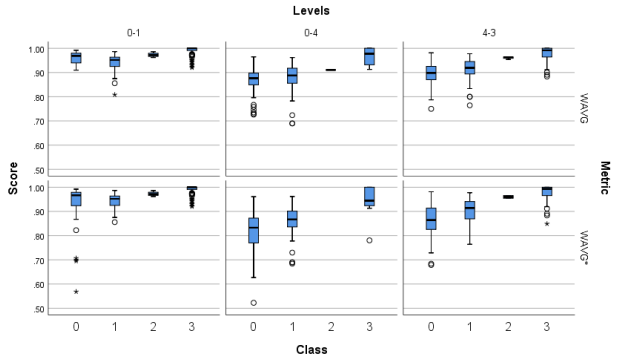
(b) Spanish (C3G vs. C3G*)



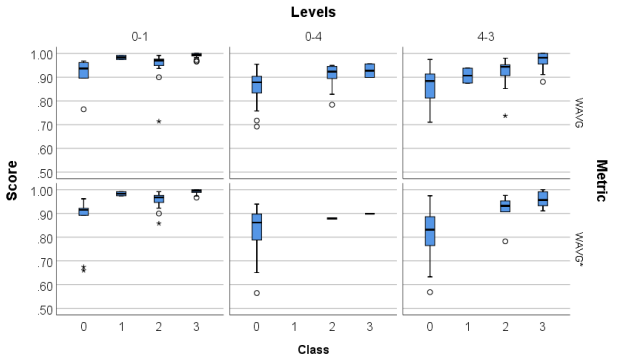
(c) English (CWASA vs. CWASA*)



(d) Spanish (CWASA vs. CWASA*)



(e) English (WAVG vs. WAVG*)



(f) Spanish (WAVG vs. WAVG*)

Figure 1: Distribution of the six similarity scores across different classes, text levels, and languages.

5.2. Classification Experiments

For classification experiments, we use the labels assigned by human evaluators as the 'gold standard' labels. As features, we use the similarity metrics obtained by CATS-Measure (one metric at the time) and the differences in word count between the original and simplified sentence (in order to distinguish between *additions* and *deletions*) which achieve similar scores for semantic similarity by all our similarity metrics (see Figure 1). Given that we observed certain differences in distribution of similarity metrics across different text levels, we also experiment with adding the level pair as an additional (third) feature for the classification.

We used five different classifiers: Logistic (le Cessie and van Houwelingen, 1992), SMOs – Weka implementation of SVM (Platt, 1998) with feature standardisation, JRip rule learner (Cohen, 1995), J48 – Weka implementation of C4.5 decision tree (Quinlan, 1993), and Random Forest (Breiman, 2001), in a 10-fold cross-validation setup with 10 repetitions in Weka Experimenter (Hall et al., 2009).

As can be seen from the classification results presented in Table 8 (only for the best classifier, logistic), although CATS-Align achieved lower performances on the Spanish dataset, the *no matches* can easier be automatically filtered for Spanish than for English (lower number of false positives for Spanish than for English). The percentage of false positives for the *no match* class (fPos) indicate that if we are interested only in filtering out *no matches*, we can successfully achieve this by training the classifiers on a small number of human annotated sentence pairs, especially for Spanish (where, in the best scenario, using the WAVG*, difference in sentence length, and the level pair, the percent-

Features	English		Spanish	
	w-F	fPos	w-F	fPos
C3G+len+levels	71.5	32.5%	82.6	16.7%
C3G+len	74.9	32.0%	82.3	21.4%
C3G*+len+levels	**75.3**	32.2%	84.1	16.3%
C3G*+len	**75.3**	32.2%	**84.8**	16.3%
CWASA+len+levels	70.1	27.5%	80.9	19.1%
CWASA+len	70.2	30.2%	83.8	14.9%
CWASA*+len+levels	72.6	**26.6%**	80.5	15.3%
CWASA*+len	72.3	27.0%	81.4	13.9%
WAVG+len+levels	65.0	45.5%	77.0	23.1%
WAVG+len	65.1	52.5%	78.6	30.3%
WAVG*+len+levels	69.7	42.0%	82.9	**13.6%**
WAVG*+len	69.6	42.0%	83.1	15.3%

Table 8: Results of automatic classification of sentence pairs into four categories (*no match*, *deletions*, *additions*, and *full matches*) presented as the weighted average F_1 measure (*w-F*) and the percentage of false positives for the *no match* class (*fPos*), i.e. cases in which *no match* was classified as any other category. The best scores achieved for each classification evaluation metric (*w-F* and *fPos*), and for each language, are presented in bold.

age of false positives for the *no match* class is only 13.6). If we are interested in classifying sentence pairs by different transformation operations, this can again be successfully achieved with classifiers trained on the small number of instances, with better results for Spanish than for English (weighted F-measure of 84.8 for Spanish, and 75.3 for English).

The results in Table 8 also indicate that specifying the level pair from which sentences were aligned improves the F-measure on the Spanish classification task (though not necessarily decreases the number of false positives for the *no match* class), but has no effect on the English classification task.

6. Conclusions

One of the main problems of the state-of-the-art automatic text simplification systems is the absence and the small size of parallel datasets (pairs of original sentences and their manually simplified versions) which leads to insufficient coverage of supervised systems. The CATS tool presented in this paper offers several different ways of sentence- and paragraph-aligning of document-aligned texts on different text complexity levels. It additionally offers three sentence similarity metrics which can be applied on sentence pairs and used for automatically classifying simplification operations as *full matches*, *additions*, *deletions*, and *no matches*. Our detailed human evaluation of the alignment module (CATS-Align) showed that it can successfully align sentence pairs from document-aligned corpora in English and Spanish. The results of classification experiments confirmed that the sentence similarity measures offered by our CATS-Measure can be used as features for classification of sentence pairs as *full matches*, *additions*, *deletions*, and *no matches* on both English and Spanish Newsela corpora. More importantly, they showed that wrongly aligned sentence pairs can be automatically filtered out by classifiers built on small size human annotated datasets (approximately 1,000 instances).

Finally, the resource-light and language-independent sentence similarity metrics offered by CATS-Measure performed similar to the state-of-the-art systems for classifying sentence pairs from English Wikipedia and Simple English Wikipedia as *full matches*, *partial matches*, and *no matches*, proving thus that they are not effective only on the news domain but also on the encyclopedic domain.

The CATS tool with both CATS-Align and CATS-Measure options is freely available on: `https://github.com/neosyon/SimpTextAlign`.

7. Acknowledgements

We would like to thank Newsela for making their data available, and to Luke Orland for his valuable comments and suggestions. Part of the work presented in this article was performed during second author's research visit to the University of Mannheim, supported by Contact Fellowship awarded by the DAAD scholarship program "STIBET Doktoranden". The work of Paolo Rosso has been partially funded by the SomEMBED TIN2015-71147-C2-1-P MINECO research project.

8. Bibliographical References

Aluísio, S. M., Specia, L., Pardo, T. A., Maziero, E. G., and Fortes, R. P. (2008). Towards brazilian portuguese automatic text simplification systems. In *Proceedings of the eighth ACM symposium on Document engineering*, DocEng '08, pages 240–248, New York, NY, USA. ACM.

Amancio, M. A. and Specia, L. (2014). An Analysis of Crowdsourced Text Simplifications . In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 123–130.

Bott, S. and Saggion, H. (2011). An Unsupervised Alignment Algorithm for Text Simplification Corpus Construction. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 20–26. ACL.

Bott, S., Saggion, H., and Figueroa, D. (2012). A Hybrid System for Spanish Text Simplification. In *Proceedings of the Workshop on Speech and Language Processing for Assistive Technologies (SLPAT), held in conjunction with NAACL-HLT 2012*, pages 75–84.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Cardellino, C. (2016). Spanish Billion Words Corpus and Embeddings. http://crscardellino.me/SBWCE/.

Cohen, W. W. (1995). Fast Effective Rule Induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123.

Franco-Salvador, M., Rangel, F., Rosso, P., Taulé, M., and Martí, M. A. (2015). Language variety identification using distributed representations of words and documents. In *Proceeding of the 6th International Conference of CLEF on Experimental IR meets Multilinguality, Multimodality, and Interaction (CLEF 2015)*, volume LNCS(9283), pages 28–40. Springer-Verlag.

Franco-Salvador, M., Gupta, P., Rosso, P., and Banchs, R. E. (2016). Cross-language plagiarism detection over continuous-space- and knowledge graph-based representations of language. *Knowledge-Based Systems*, 111:87 – 99.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11:10–18.

Hwang, W., Hajishirzi, H., Ostendorf, M., and Wu, W. (2015). Aligning Sentences from Standard Wikipedia to Simple Wikipedia. In *Proceedings of NAACL&HLT*, pages 211–217.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1537–1546. ACL.

le Cessie, S. and van Houwelingen, J. C. (1992). Ridge Estimators in Logistic Regression. *Applied Statistics*, 41(1):191–201.

Mcnamee, P. and Mayfield, J. (2004). Character n-gram tokenization for European language text retrieval. *Information Retrieval*, 7(1):73–97.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *Proceedings of Workshop at International Conference on Learning Representations*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Newsela. (2016). Newsela article corpus. `https://newsela.com/data`. Version: 2016-01-29.

Platt, J. C. (1998). Fast Training of Support Vector Machines using Sequential Minimal Optimization. In B. Schoelkopf, et al., editors, *Advances in Kernel Methods 6 Support Vector Learning*, chapter 12, pages 41–65. MIT Press Cambridge.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA.

Saggion, H., Štajner, S., Bott, S., Mille, S., Rello, L., and Drndarevic, B. (2015). Making It Simplext: Implementation and Evaluation of a Text Simplification System for Spanish. *ACM Transactions on Accessible Computing*, 6(4):14:1–14:36.

Salton, G. and McGill, M. J. (1986). *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc.

Štajner, S., Bechara, H., and Saggion, H. (2015). A Deeper Exploration of the Standard PB-SMT Approach to Text Simplification and its Evaluation. In *Proceedings of ACL&IJCNLP (Volume 2: Short Papers)*, pages 823–828.

Štajner, S., Franco-Salvador, M., Ponzetto, S. P., Rosso, P., and Stuckenschmidt, H. (2017). Sentence alignment methods for improving text simplification systems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 97–102.

Xu, W., Callison-Burch, C., and Napoles, C. (2015). Problems in Current Text Simplification Research: New Data Can Help. *Transactions of the Association for Computational Linguistics (TACL)*, 3:283–297.

KIT-Multi: A Translation-Oriented Multilingual Embedding Corpus

Thanh-Le Ha, Jan Niehues, Matthias Sperber, Ngoc Quan Pham and Alexander Waibel
Karlsruhe Institute of Technology, Karlsruhe, Germany
firstname.lastname@kit.edu

Abstract

Cross-lingual word embeddings are the representations of words across languages in a shared continuous vector space. Cross-lingual word embeddings have been shown to be helpful in the development of cross-lingual natural language processing tools. In case of more than two languages involved, we call them multilingual word embeddings. In this work, we introduce a multilingual word embedding corpus which is acquired by using neural machine translation. Unlike other cross-lingual embedding corpora, the embeddings can be learned from significantly smaller portions of data and for multiple languages at once. An intrinsic evaluation on monolingual tasks shows that our method is fairly competitive to the prevalent methods but on the cross-lingual document classification task, it obtains the best figures. We are in the process to produce the embeddings for more languages, especially the languages which belong to the same family or sematically close to each others, such as Japanese-Korean, Chinese-Vietnamese, German-Dutch, or Latin-based languagues. Furthermore, the corpus is being analyzedd regarding its usage and usefulness in other cross-lingual tasks.

Keywords: multilingual embeddings, cross-lingual embeddings, neural machine translation, multi-source translation

1. Introduction

Inducing cross-lingual word embeddings is essentially acquiring word embeddings in different languages. The cross-lingual word embeddings can then be used as pre-trained models in cross-lingual applications such as cross-lingual document classification, information retrieval, textual entailment and question answering. Cross-lingual word embeddings can also help to perform transfer learning from a well-resource language to another low-resource language on various tasks, e.g. in building WordNet or annotating semantic relations.

There have been various methods of cross-lingual embedding induction being proposed, but most of them are essentially bilingual in the perspective that they learn to induce *bilingual embeddings* from *bilingual* data [1]. Basically these methods optimize some cross-lingual constraints so that the semantic similarity between words corresponds to the closeness of these representations in a common vector space. Consequently, if they need cross-lingual embeddings for a new language pair, they must apply their inducing method on that new bilingual data. Furthermore, there would be some domain mismatch between the new acquired embeddings and the others if the new bilingual data are from different domain. The aforementioned limitations of those cross-lingual corpora motivates us to design a *multilingual embedding* inducing method from a *single* corpus which is available in as many languages as possible.

In this paper, we propose such an approach utilizing a multilingual neural machine translation (NMT) system to constrain the embeddings from n source languages while translating into the same target language (as we call it multi-source NMT). The source embeddings employed in this model are implicitly forced to learn the common semantic regularities in order to maximize the translation quality of every language pair in the system. Once the multi-source NMT model is trained to a good state, the source word embeddings can be simply extracted from the model and used as a multilingual word embeddings.

The contribution of this work is the introduction of a method and its product corpus, KIT-Multi[2], consisting multilingual word embeddings of English-German-French. Other languages such as Chinese, Japanese, Korean, Vietnamese, Dutch, Italian, Romanian, Spanish or Portuguese are being added. We conducted some preliminary evaluations on KIT-Multi and compares to other cross-lingual embedding corpora. It has been shown that our multilingual corpus achieves competitive performances in standard evaluations as well as it has better coverage while using much less data for the training process. The evaluations on other languages would be pulished in the final version of the paper.

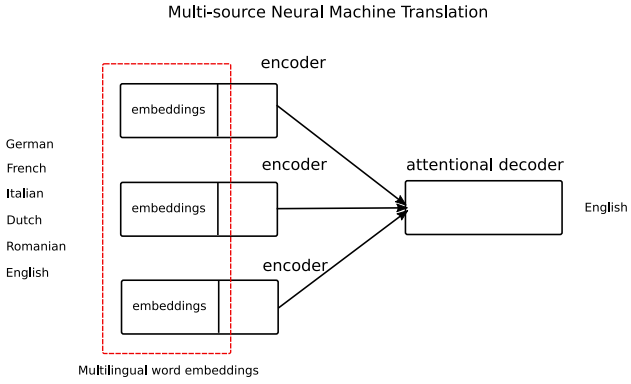


Figure 2: Multi-source Neural Machine Translation system and how to get multilingual word embeddings from it.

[1] For a thoroughly review of the most popular and advantageous techniques of cross-lingual word embedding induction, please refer to Upadhyay et al. (2016). For even more detailed and broader survey, please refer to Ruder (2017).

[2] The corpus is published and constantly updated at http://i13pc106.ira.uka.de/~tha/KIT-Multi/

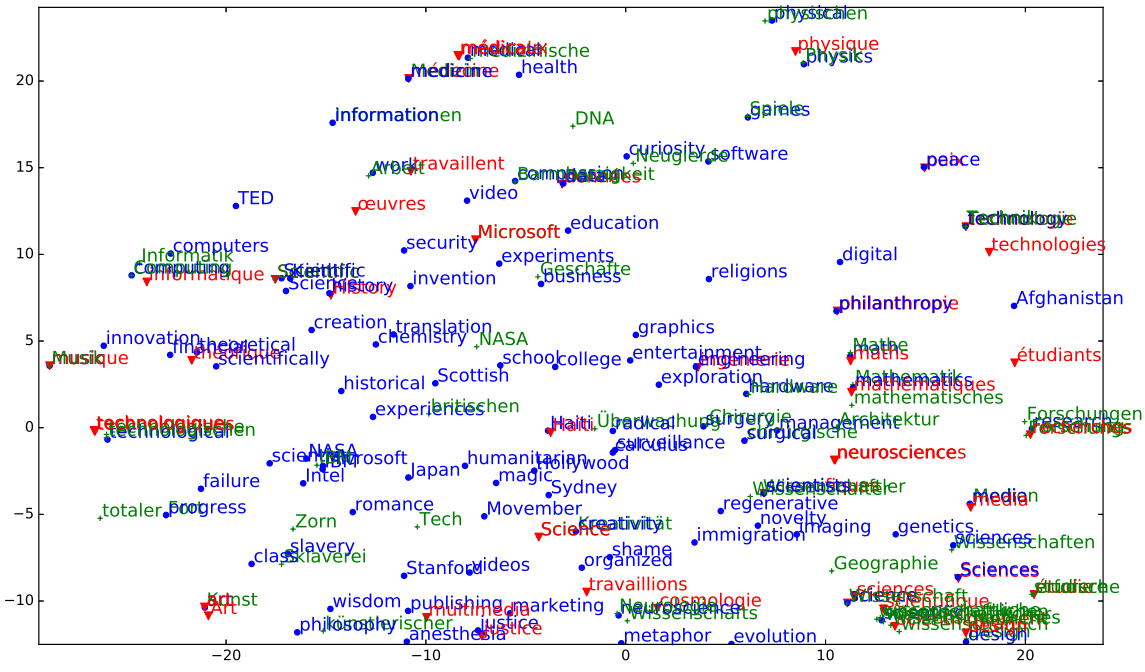


Figure 1: The multilingual word embeddings from the shared representation space of the source. To illustrate more clearly, only the word vectors of the words related to "*science*" are projected and visualized. The blue words are the English words, green for German and the red ones are the French words. Please zoom in to see more detailed.

2. Multilingual Word Embedding Corpus

2.1. Embedding Induction Method

A neural machine translation system (Bahdanau et al., 2014) consists of an encoder representing a source sentence and an attention-based decoder that produces the translated sentence. One of the most notable differences of NMT compared to the conventional statistical approach is that the source words can be represented in a continuous space (i.e. word embeddings) in which the semantic regularities are induced automatically. Being applied to multilingual settings, NMT systems have been proved to be benefited from additional information embedded in a common semantic space across languages (Johnson et al., 2016; Ha et al., 2016; Currey et al., 2017). An interesting and positive side effect of such a system is the simultaneous induction of multilingual embeddings from the source side.

In a multi-source NMT systems where the sentences from *several sources languages* are translated to *one target language*, the source embeddings are tied to a common semantic space across languages. So the source embeddings has its inherent cross-lingual characteristics, which could be extremely helpful for the cross-lingual applications employing the embeddings. More specifically, in our previous work on multi-source NMT (Ha et al., 2016), the words in each source sentence are coded with the language of that sentence before feeding to the training process of a standard neural machine translation system. For example, the source sentence in English: `they have since abandoned that project` would become ***en***`_they` ***en***`_have` ***en***`_since` ***en***`_abandoned` ***en***`_that` ***en***`_project`. anguage coding is conducted in the preprocessing phrase. Our multilingual embeddings are the derived product of this multi-source system. The figure 2 describes the process.

2.2. `KIT-Multi` Corpus

Our corpus is induced from WIT3's TED subtitle corpus (Cettolo et al., 2012) including bilingual corpora from French, German, Dutch, Italian and Romanian to English. TED is a much smaller multilingual data compared to Europarl and contains other languages than European languages. The multi-source NMT is trained using the NMT framework `OpenNMT`[3] (Klein et al., 2016) to translate from aforementioned languages (including English) to the only target language English. The statistics of TED bilingual corpora and our multilingual embedding corpus are shown in Table 1 and Table 2, respectively.

Language pairs	Number of sentences
German-English	196794
French-English	195025
Dutch-English	230866
Italian-English	220812
Romanian-English	210402

Table 1: Statistics of pair-wise TED bilingual corpora

[3] `http://opennmt.net`

Languages	Number of entries
English	21001
French	25685
German	24182
Dutch	24167
Italian	23422
Romanian	25505

Table 2: The size of the KIT-Multi embedding corpus

Figure 1 illustrates the visualization of multilingual word embeddings extracted from the multi-source NMT system and projected to the 2D space using *t-SNE* (Maaten and Hinton, 2008). It shows how different words in different languages, i.e. English-German-French, can be close in the shared semantic space after being trained to translate into a common language (English).

Table 3 shows the closest words in the semantic space based on Cosine similarity with respect to some examples. We also include the language codes to clarify the origin of each word. From the table, we can see that the most close words are actually the words having the same meaning but in other languages.

@en@research	
Word	Cosine Similarity
@de@Forschung	0.727675
@fr@recherches	0.697122
@de@Forschungs	0.671166
@fr@recherche	0.643990
@de@geforscht	0.637604
@en@humanity	
Word	Cosine Similarity
@de@Menschlichkeit	0.691524
@fr@humanité	0.684639
@de@Menschheit	0.645123
@de@Menscheit	0.634902
@en@mankind	0.621472

Table 3: Top 5 closest words by Cosine similarity.

3. Preliminary evaluation of KIT-Multi

In this section, we describe some initial evaluation of our multilingual embedding corpus over some standard intrinsic and extrinsic evaluations, in comparisons with some other popular approaches for cross-lingual word embedding induction.

We mostly follow the experimental layout and settings of Upadhyay et al. (2016), conducting intrinsic and extrinsic evaluations on three European languages: English, French and German. The intrinsic evaluation is the *monolingual word similarity* task. The extrinsic evaluation focuses on the *cross-lingual document classification*. In this task, a document classifier is trained on a training set composed by a language L_1 and then predict the test set which is in the different language L_2. The process is then reversed for the language pair, and the classification accuracy is used to judge the quality of the cross-lingual embeddings. The corpora chosen to be compared are the corpora induced by Skip - Bilingual Skip-gram (Luong et al., 2015), CVM - Bilingual Compositional Model (Hermann and Blunsom, 2014) and VCD - Bilingual Vectors from Comparable Data (Vulic and Moens, 2015), which are all trained on much bigger Europarl v7 parallel corpora[4] (Koehn, 2005). To show the impact of the corpus size, we also train the Bilingual Skip-gram embeddings with the same corpora used to train our model, and name it Skip-TED. For the details of those methods, please refer to Upadhyay et al. (2016).

In the intrinsic monolingual evaluation, we consider the word embeddings in one language at a time, i.e. the monolingual word embeddings, in order to conduct the *word similarity*. The Spearman's rank correlation coefficient (Myers et al., 1995) between system similarity and human is the measure to judge the quality of the induced word embeddings. The English evaluation datasets are SimLex999 (*En-999*) and WordSim353 (*En-353*), in which the former (Hill et al., 2016) is claimed to better capture the similarity rather than both similarity and relatedness like in the latter (Finkelstein et al., 2002). The German (*De*) and French (*Fr*) datasets are the WordSim353 counterparts (Camacho-Collados et al., 2015; Leviant and Reichart, 2015).

The scores in Table 4 show that our word embeddings are competent in term of monolingual aspect even though they are not trained to be adapted to monolingual quality. Moreover, our word embeddings perform better than the Skip embeddings trained on the same data by a large margin.

As shown in Table 5, the classifiers trained on our embeddings achieve highest accuracy on both directions of English⇔German, considerably better than other approaches. It is notable that, our model is trained on a substantially smaller corpus.

4. Related Work and Discussion

In (Upadhyay et al., 2016), the most popular and advantageous techniques for multilingual word embedding induction have been thoroughly evaluated. Corpora induced by Skip and VCD are the methods having the capability of monolingual adaptation by adjusting a hyper-parameter (in Skip models) or the portion of texts in each language (in VCD models). Furthermore, since they are designed based on the *skip-gram* models (Mikolov et al., 2013), it is unsurprising that they perform well on monolingual tasks. Corpora induced by CVM and our KIT-Multi, in contrast, are designed with cross-lingual orientation so that they focus more on similarity instead of relatedness. Aforementioned, our KIT-Multi corpus has shown its potential by achieving high accuracies on the task despite being induced from a significally smaller corpus. Compared to the corpora acquired by their method, our embedding inherently induced in multilingual settings, with an arbitrary number of source and target languages, instead of being limited to bilingual. Those advantages allow us to extend our corpus seamlessly to many languages using small multilingual corpus, ideally from TED talks.

[4] http://www.statmt.org/europarl/

Language	Skip	Skip-TED	CVM	VCD	*KIT-Multi*
En-999	0.34	0.22	**0.37**	0.32	***0.37***
En-353	0.53	0.39	0.43	**0.59**	*0.45*
De	0.52	0.40	0.40	**0.54**	*0.51*
Fr	**0.50**	0.09	0.38	0.43	*0.48*

Table 4: Monolingual evaluation tasks.

L_1	L_2	Skip	Skip-TED	CVM	VCD	*KIT-Multi*
En	De	85.2	84.3	85.0	79.9	***86.6***
De	En	74.9	73.5	71.1	74.1	***79.7***

Table 5: The accuracy of cross-lingual document classification task using the word embeddings.

5. Conclusion and Future Work

In this proposal, we introduce a method to extract multilingual embedding corpus and its production, `KIT-Multi`. We would like to extend it for more languages as well as more cross-lingual natural language processing applications. The corpus will be available in Japanese, Korean, Chinese, Vietnamese, English, German, Dutch, Italian, French, Spanish and Portuguese at the time of the conference. We welcome other groups download and use it in other tasks and discuss about its usefulness.

6. Bibliographical References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR*, abs/1409.0473.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2015). A framework for the construction of monolingual and cross-lingual word similarity datasets. In *ACL (2)*, pages 1–7.

Cettolo, M., Girardi, C., and Federico, M. (2012). Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy, May.

Currey, A., Barone, A. V. M., and Heafield, K. (2017). Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: the concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Ha, T.-L., Niehues, J., and Waibel, A. (2016). Toward multilingual neural machine translation with universal encoder and decoder. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, Seattle, WA, USA.

Hermann, K. M. and Blunsom, P. (2014). Multilingual Models for Compositional Distributed Semantics. *Acl*, pages 58–68.

Hill, F., Reichart, R., and Korhonen, A. (2016). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F. B., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2016). Google's multilingual neural machine translation system: Enabling zero-shot translation. *CoRR*, abs/1611.04558.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. M. (2016). OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Leviant, I. and Reichart, R. (2015). Separated by an un-common language: Towards judgment language informed vector space modeling. *arXiv preprint arXiv:1508.00106*.

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Bilingual Word Representations with Monolingual Quality in Mind. *Workshop on Vector Modeling for NLP*, pages 151–159.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Myers, J. L., Well, A., and Lorch, R. F. (1995). *Research design and statistical analysis*. Routledge.

Ruder, S. (2017). A survey of cross-lingual embedding models. *CoRR*, abs/1706.04902.

Upadhyay, S., Faruqui, M., Dyer, C., and Roth, D. (2016). Cross-lingual Models of Word Embeddings: An Empirical Comparison. *Acl 2016*, pages 1661–1670.

Vulic, I. and Moens, M.-F. (2015). Bilingual Word Embeddings from Non-Parallel Document-Aligned Data Applied to Bilingual Lexicon Induction. *Acl-2015*, pages 719–725.

Multi-lingual Argumentative Corpora in English, Turkish, Greek, Albanian, Croatian, Serbian, Macedonian, Bulgarian, Romanian and Arabic

Alfred Sliwa, Yuan Ma, Ruishen Liu, Niravkumar Borad
Seyedeh Fatemeh Ziyaei, Mina Ghobadi, Firas Sabbah, Ahmet Aker
Department of Computer Science and Applied Cognitive Science
University of Duisburg-Essen
Lotharstraße 65, 47057 Duisburg
{alfred.sliwa.92, yuan.ma, ruishen.liu, niravkumar.borad
seyedeh.ziyaei, mina.ghobadi, firas.sabbah}@stud.uni-due.de
a.aker@is.inf.uni-due.de

Abstract

Argumentative corpora are costly to create and are available in only few languages with English dominating the area. In this paper we release the first publicly available corpora in all Balkan languages and Arabic. The corpora are obtained by using parallel corpora where the source language is English and target language is either a Balkan language or Arabic. We use 8 different argument mining classifiers trained for English, apply them all on the source language and project the decision made by the classifiers to the target language. We assess the performance of the classifiers on a manually annotated news corpus. Our results show when at least 3 to 6 classifiers are used to judge a piece of text as argumentative an F1-score above 90% is obtained.

Keywords: Multilingual Argument Annotations, Argument Mining, Parallel Corpora

1. Introduction

Argument mining refers to the automatic extraction of arguments from natural texts. An argument consists of a claim (also referred to as the conclusion of the argument) and several pieces of evidence called premises that support or reject the claim (Lippi and Torroni, 2016b). Identifying arguments in large volumes of textual data has the potential to revolutionarise our access to information. Argument based search for information would for example facilitate individual and organisational decision-making, make learning more efficient, enable quicker reporting on present and past events, to name just a few broad applications. Even more important is argument mining in the multi-lingual context, by which argument based retrieval would be available to people in the language of their preference.

Current studies report methods for argument mining in legal documents (Reed et al., 2008), persuasive essays (Nguyen and Litman, 2015), Wikipedia articles (Levy et al., 2014; Rinott et al., 2015), discussion fora (Swanson et al., 2015), political debates (Lippi and Torroni, 2016a) and news (Sardianos et al., 2015; Al-Khatib et al., 2016). In terms of methodology, supervised machine learning is a central technique used in all these studies. This assumes the availability of data sets – argumentative texts – to train and test the argument mining models. Such data sets are readily available in English and – although in comparably smaller quantities – in very few European languages such as German or Italian. Languages other than these are currently neglected. Due to this lack of data the research and development of argumentation mining outside English and few European languages is very limited, rendering multilingual argument mining and language independent argument based retrieval impossible.

In this research we aim to fill this gap. We aim to create multi-lingual corpora annotated with argumentative structures automatically. For this purpose we make use of parallel corpora containing multiple bilingual documents aligned at sentence level, i.e. every sentence in a document written in a source language such as English is translated into a target language such as Greek. As the sentences in the documents are parallel it infers that if one of the sentence pairs is argumentative so it is also the other sentence. This also means annotating for instance English sentences for arguments leads also argumentative annotations in the target languages. One way of annotating the English sentences would be through human annotators. However, this is very intensive and costly task, especially when the task is to annotate several thousands documents which is the case in our research. Another way is to rely on existing robust argumentation tools and perform the annotation automatically through argument projection – a task recently proposed by (Aker and Zhang, 2017). This is exactly what we do in this research. We use eight different robust argument annotation tools created for English Aker et al. (2017), apply these on the English documents and annotate every sentence whether it is argumentative or not. We then project the annotations to the target languages consisting of Balkan languages such as Turkish, Greek, Albanian, Croatian, Serbian, Macedonian, Bulgarian and Romanian. The corpora we annotate is the SETimes corpora aligned at sentence level by Tyers and Alperen (2010). Since these corpora are publicly available we also release our annotations for the public. In addition, we also gathered around 3543 parallel English and Arabic documents from Huffington Post and processed them similar to the Balkan languages. By request we will make both annotated corpora available. With these pieces of information it is feasible to download the articles and also track-back the annotations.

The availability of such rich argumentative corpora is the first step to close the gap between English and under-resourced languages in terms of argument mining and kick-off efforts in creating argument mining solutions for languages other than English. Furthermore, we also think that it will start opening research direction towards multi-

Language Pair	# Documents	# Sentences
BG-EN	22,531	161,436
EL-EN	23,210	166,430
HR-EN	21,062	158,963
MK-EN	22,865	154,570
RO-EN	22,992	172,573
SQ-EN	22,947	171,885
SR-EN	22,779	164,377
TR-EN	22,800	166,510
AR-EN	3,543	85,831

Table 1: Characteristics of Parallel Corpora. All numbers refer to English data.

lingual argument mining and retrieval.

2. Data: SETimes and Huffington Post

The SETimes is an open source parallel corpus of news articles in the Balkan languages such as Turkish, Greek, Albanian, Croatian, Serbian, Macedonian, Bulgarian and Romanian and English, originating from a multi-lingual news website (Tiedemann, 2009). All documents extracted from the news website[1] are translated to XML files for each language pair and aligned at sentence level (Tyers and Alperen, 2010).

In addition to the SETimes corpora we also collected 3543 Arabic-English parallel news articles from Huffington Post. Although the articles are parallel, i.e. translations of each other, the alignment information between the sentences is not given so that we implemented simple heuristics such as sentence position, sentence length and dictionary-based translation overlap to provide this information. Table 1 shows both corpora in numbers.

3. Method

3.1. Pipeline

Our pipeline of annotating the data described in Section 2. is shown in Figure 1 – the figure shows the pipeline on SETimes as example however, processing the English-Arabic corpus happens analog. The first step is about reading English sentences from a parallel corpus such as English-Greek. For each sentence we extract rich feature sets detailed in the next section and apply 8 different argument mining models to annotate the sentence as argumentative or non-argumentative. Finally, we write the answer back to the corpus. More precisely, we record the type (argumentative or not) determined based on the majority vote among 8 annotators (at least 5 annotators are required to make clear decision) and as well as the decisions of each annotator. If we have 4:4 decisions for each type the overall result depends on the confidence ranking of each voter.

3.2. Argument Mining Tool

We used 8 different argument mining models to annotate the English sentences Aker et al. (2017). Seven of the models make use of traditional machine learning methods such as SVM, decision trees, etc. The 8^{th} model applies Convolutional Neural Networks (CNN) to predict the type labels. The CNN model uses various word embeddings. The other 7 models rely on rich set of features grouped into structural, lexical, syntactical and contextual categories.

[1] www.setimes.com, however this website is not maintained anymore.

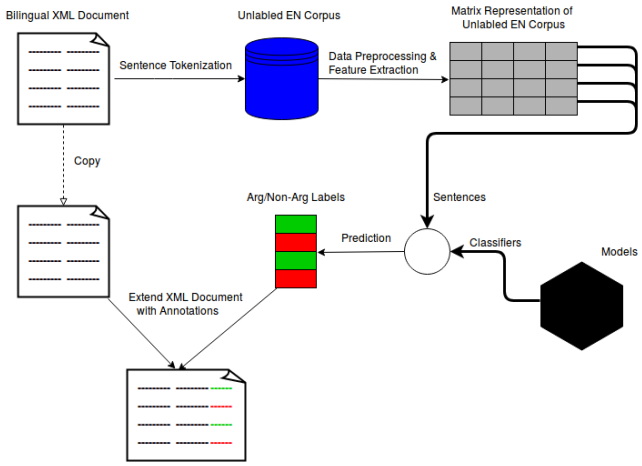


Figure 1: Data Processing Pipeline System

4. Annotation results

According to annotation results we list statistics about classification distribution for each analyzed parallel corpus. Table 2 shows the distribution of predicted argumentative and non-argumentative sentences for each bilingual data source. Note a sentence is regarded as argumentative when majority of the argument mining tools predicted that particular class. Otherwise the sentence is marked as non-argumentative. From the table we see that around 2/3 of the sentences are non-argumentative and around 1/3 are argumentative. This picture is repeated for each language pair.

5. Evaluation

In terms of evaluation we measure the aggregated performance of our eight argument mining models on a manually annotated news corpus (see Section 5.1.). In order to achieve this we use three distinct corpora to train and test the models. The corpora include persuasive essays (Nguyen and Litman, 2015), Wikipedia articles (Rinott et al., 2015) and news articles. At first we divide our data set into training set, validation set and held-out test set. All instances of essay, Wikipedia corpus and 80 news articles are used for training and hyper parameter tuning our models. We perform 4-fold cross-validation with 20 news articles held-out for validation purposes. The held-out test set contains 20 news articles and is used to determine the evaluation performance of the trained models. At this point we compute an aggregated prediction vector based on the individual model votes. By introducing a threshold k for the minimum number of argumentative votes, we can decide whether a particular test instance is assigned as argumentative or not. The threshold k takes eight different values from the set $\{1, 2, 3, 4, 5, 6, 7, 8\}$. In this way we report eight different evaluation results. The procedure of the evaluation experiment can be tracked in Figure 2.

Language Pair	# Argumentative Sentences	# Non-Argumentative Sentences
BG-EN	54,171	107,265
EL-EN	55,486	110,944
HR-EN	54,417	104,546
MK-EN	51,552	103,018
RO-EN	59,079	113,494
SQ-EN	58,750	113,135
SR-EN	56,232	108,145
TR-EN	56,833	109,677
AR-EN	29,602	56,229

Table 2: Classification Distribution of Argumentative/Non-Argumentative Sentences for Parallel Corpora

5.1. Annotated news corpus

In addition to the essay and Wikipedia corpora we also manually annotated 100 news articles from The Guardian newspaper, related to the general topic of "Brexit". On average the articles have a length of $\overline{x} = 75$ sentences (range of [24,186]). Each article is annotated for claims and premises by an expert. For the purpose of training and testing for the above classifiers we do not distinguish between claims and premises but treat both annotation types as argument. Any sentence in the news article not marked as claim or premise is regarded as non argumentative.

5.2. Evaluation results

In Table 3 we report the evaluation results for each threshold k. Each evaluation report contains statistics about precision, recall and F1-score values for argumentative class, non-argumentative class and average among both classes. It is observable that in case of low threshold values the precision score for Argument class is low but the recall score for the same class is high. Because of the fact that many test instances are classified as argumentative there are more non-argumentative instances incorrectly classified as argumentative. This leads to a high recall score but to a low precision score. In case of high threshold values the recall score for Argument class is low but the precision score for the same class is high. As only high probable test instances are classified as argumentative there are more argumentative instances incorrectly classified as non-argumentative. This yields in high precision score but low recall score.
We can see that by increasing k the precision also increases but the recall simultaneously decreases. According to average F1-score the optimal threshold values are 4 and 5 with a score of 0.96.

6. Conclusion

In this paper we described the issue with argument mining in languages other than English, namely the non-availability of argumentative training data. We motivated the idea of overcoming this disadvantage using parallel data and automatic argument annotation. We processed the SETimes corpora as well as an English-Arabic corpus that we collected from HuffingtonPost. Our processing includes the annotation of English sentences as argumentative or non-argumentative. We used 8 different argument mining models and make use of majority voting to mark the class labels for the sentences. Our annotations are freely available by request.

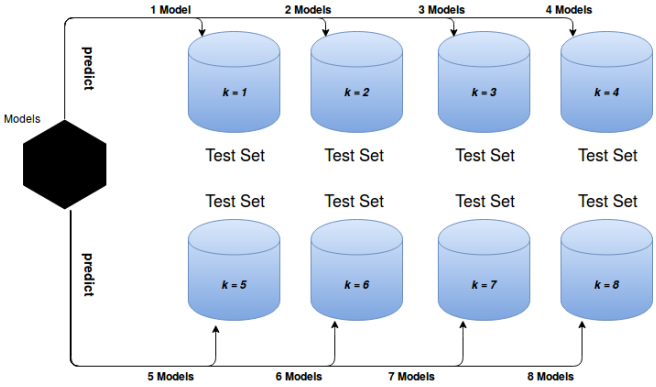


Figure 2: The Evaluation procedure containing eight replicated test sets where a various number of argument mining models k contribute to argumentative sentence prediction.

We also evaluated the performance of our classifiers on a manually annotated news corpus. Our results show that best F1-score is achieved when 3 to 6 classifiers are used to judge whether a piece of text is argumentative or not. In future we plan to use the map argumentative corpora to train argument mining systems in the respective languages.

7. Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media".

8. Bibliographical References

Aker, A. and Zhang, H. (2017). Projection of argumentative corpora from source to target languages. In *Proceedings of the 4th Workshop on Argument Mining*, pages 67–72.

Aker, A., Sliwa, A., Ma, Y., Lui, R., Borad, N., Ziyaei, S., and Ghobadi, M. (2017). What works and what does not: Classifier and feature analysis for argument mining. In *Proceedings of the 4th Workshop on Argument Mining*, pages 91–96.

Al-Khatib, K., Wachsmuth, H., Kiesel, J., Hagen, M., and Stein, B. (2016). A news editorial corpus for mining argumentation strategies. In *Proceedings of the 26th International Conference on Computational Linguistics*.

Levy, R., Bilu, Y., Hershcovich, D., Aharoni, E., and Slonim, N. (2014). Context dependent claim detection.

Threshold k	Class	Precision	Recall	F1-Score
1	Argument	**0.61**	**1.0**	0.76
	Non-Argument	1.0	0.08	0.14
	Average/Total	0.77	.0.62	0.51
2	Argument	**0.74**	**0.99**	0.85
	Non-Argument	0.98	0.51	0.67
	Average/Total	0.84	.0.79	0.78
3	Argument	**0.89**	**0.98**	0.94
	Non-Argument	0.97	0.83	0.9
	Average/Total	0.93	.0.92	0.92
4	Argument	**0.96**	**0.96**	0.96
	Non-Argument	0.95	0.95	0.95
	Average/Total	0.96	.0.96	0.96
5	Argument	**0.98**	**0.95**	0.96
	Non-Argument	0.93	0.97	0.95
	Average/Total	0.96	.0.96	0.96
6	Argument	**0.99**	**0.91**	0.95
	Non-Argument	0.89	0.99	0.94
	Average/Total	0.95	.0.95	0.95
7	Argument	**1.0**	**0.82**	0.9
	Non-Argument	0.79	1.0	0.88
	Average/Total	0.91	.0.89	0.89
8	Argument	**1.0**	**0.44**	0.61
	Non-Argument	0.55	1.0	0.71
	Average/Total	0.82	.0.67	0.65

Table 3: Evaluation Results of Argumentative Sentence Detection on the News Articles held-out test set for varying threshold values k.

Lippi, M. and Torroni, P. (2016a). Argument mining from speech: Detecting claims in political debates. In *AAAI*, pages 2979–2985.

Lippi, M. and Torroni, P. (2016b). Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology (TOIT)*, 16(2):10.

Nguyen, H. V. and Litman, D. J. (2015). Extracting argument and domain words for identifying argument components in texts. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 22–28.

Reed, C., Mochales Palau, R., Rowe, G., and Moens, M.-F. (2008). Language resources for studying argument. In *Proceedings of the 6th conference on language resources and evaluation-LREC 2008*, pages 91–100. ELRA.

Rinott, R., Dankin, L., Perez, C. A., Khapra, M. M., Aharoni, E., and Slonim, N. (2015). Show me your evidence-an automatic method for context dependent evidence detection. In *EMNLP*, pages 440–450.

Sardianos, C., Katakis, I. M., Petasis, G., and Karkaletsis, V. (2015). Argument extraction from news. *NAACL HLT 2015*, page 56.

Swanson, R., Ecker, B., and Walker, M. (2015). Argument mining: Extracting arguments from online dialogue. In *Proceedings of 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2015)*, pages 217–227.

Tiedemann, J. (2009). News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, et al., editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria.

Tyers, F. M. and Alperen, M. S. (2010). South-east european times: A parallel corpus of balkan languages. In *Proceedings of the LREC Workshop on Exploitation of Multilingual Resources and Tools for Central and (South-) Eastern European Languages*, pages 49–53.

SemR-11: A Multi-Lingual Gold-Standard for Semantic Similarity and Relatedness for Eleven Languages

Siamak Barzegar[1], Brian Davis[2], Manel Zarrouk[1] Siegfried Handschuh[3], Andre Freitas[4]
[1]Insight Centre for Data Analytics, National University of Ireland, Galway
siamak.barzegar,manel.zarrouk@insight-centre.org
[2]Department of Computer Science, Maynooth University
brian.davis@mu.ie
[3]Department of Mathematics and Computer Science, University of Passau
siegfried.handschuh@uni-passau.de
[4]School of Computer Science, The University of Manchester
andre.freitas@manchester.ac.uk

Abstract

This work describes SemR-11, a multi-lingual dataset for evaluating semantic similarity and relatedness for 11 languages (German, French, Russian, Italian, Dutch, Chinese, Portuguese, Swedish, Spanish, Arabic and Persian). Semantic similarity and relatedness gold standards have been initially used to support the evaluation of semantic distance measures in the context of linguistic and knowledge resources and distributional semantic models. SemR-11 builds upon the English gold-standards of Miller & Charles (MC), Rubenstein & Goodenough (RG), WordSimilarity 353 (WS-353), and Simlex-999, providing a canonical translation for them. The final dataset consists of 15,917 word pairs and can be used to support the construction and evaluation of semantic similarity/relatedness and distributional semantic models. As a case study, the SemR-11 test collections was used to investigate how different distributional semantic models built from corpora in different languages and with different sizes perform in computing semantic relatedness similarity and relatedness tasks.

Keywords: Gold standard, Semantic Similarity, Semantic Relatedness, Multi-linguality, Word-embeddings

1. Motivation

The ability to automatically determine and quantify the degree of semantic similarity and semantic relatedness between pairs of words or expressions is one of the archetypal tasks for assessing the ability of a system to perform semantic interpretation. The ability to quantify semantic relatedness can provide a lightweight semantic interpretation operation which can be applied in different areas of Artificial Intelligence, Natural Language Processing and Information Retrieval. Examples of applications include coping with lexical and semantic gaps in Question Answering Systems (Freitas, 2015; Freitas and Curry, 2014), using the semantic relatedness score as a ranking function in Information Retrieval systems (Freitas et al., 2012) and serving as a semantic scoping mechanism in deductive/abductive methods (Freitas et al., 2014).

Due to its simplicity in comparison to other tasks such as Question Answering, Text Entailment and Machine Translation, semantic similarity and relatedness gold standards have been initially used to support the evaluation of the interaction between semantic distance measures and of linguistic and knowledge resources (Resnik, 1995; Lin, 1991; Wu and Palmer, 1994; Agirre et al., 2009). As the conditions to process large-scale corpora emerged, distributional semantic models automatically built from textual corpora were created (Turney and Pantel, 2010a) using, in most cases, a vector space representation of meaning. As distributional semantic models can induce modes with a more comprehensive underlying vocabulary and also capture a broader set of semantic relations, new gold-standards emerged (Finkelstein et al., 2001), evolving from capturing semantic similarity to semantic relatedness behavior. More recently, the creation of neural/predictive word embedding models (Mikolov et al., 2013; Pennington et al., 2014) pushed semantic similarity and relatedness gold-standards to evolve in the direction of quantifying more fine-grained semantic relations (Hill et al., 2015).

Currently, most of the existing gold-standards for evaluating semantic similarity and relatedness have focused on the English language, with some initiatives providing initial gold-standards for few other languages (Faruqui and Dyer, 2014). This paper describes SemR-11, a multi-lingual gold-standard which aims at generalizing existing semantic similarity and relatedness gold-standards to 11 languages (German, French, Russian, Italian, Dutch, Chinese, Portuguese, Swedish, Spanish, Arabic and Persian). The resource is built using a principled translation method over four reference gold-standards: Miller & Charles (Miller and Charles, 1991), Rubenstein & Goodenough (Rubenstein and Goodenough, 1965), WS-353 (Finkelstein et al., 2001) and Simlex-999 (Leviant and Reichart, 2015). The final resource contains in total 15,917 word pairs.

The resource aims to contribute to research in the following directions:

- Supporting the development of linguistic resources and distributional semantic models for non-English languages.
- Providing a comparative framework for analyzing the impact of language structural features and types (e.g. analytic, isolating and synthetic languages) in the development of semantic relatedness models.

- Evaluating the use of machine translation to support semantic similarity and relatedness (Freitas et al., 2016).
- Creating semantic similarity and relatedness models which work on languages not having a high-volume supporting corpora.

This paper is organized as follows: Section 2. describes the state-of-the-art in existing gold-standards for semantic similarity and relatedness computations as well as their language variants; Section 3. describes the English gold-standards which were used as a reference for the machine translation process; Section 4. describes the SemR-11 gold-standard and its creation process.

2. Related Work

Camacho-Collados et al. (2017) developed a multi-lingual gold-standard which includes 518 word pairs for five languages (English, German, Italian, Spanish and Persian). It is composed of nominal pairs of multi-word expressions, domain-specific terms and named entities that are manually scored between 0 to 4 where 0 indicates that they are completely dissimilar and 4 denotes that the two words are synonymous. This dataset (Camacho-Collados et al., 2017) focuses on semantic similarity.
Bruni et al. (2014) introduced a test collection containing 3000 word pairs. The MEN dataset obtained by crowdsourcing using Amazon Mechanical Turk [1] via the CrowdFlower[2] interface. The dataset focuses on semantic relatedness pairs on the English language (similarly to the WS-353 dataset (Finkelstein et al., 2001)). They developed it, specifically to test multimodal models. Compared to WS-353, MEN is sufficiently large, and the human judgments are relative rather than absolute. At (Bruni et al., 2014), each rater chose the word pair that was more similar out of two random pairs of words. They used this technique to have a comparative judgment rather than absolute scores for single pairs, which was used in the WS-353.
Agirre et al. (2009) split the WS-353 (Finkelstein et al., 2001) into two test collections (WS-Sim and WS-Rel) containing 203 and 252 word pairs on the English language, respectively. *WS-Sim* focused on only measuring similarity, and the other one on only relatedness.

3. Reference Gold-standards

SemR-11 consists of the translation of four semantic similarity and relatedness gold-standards: Miller & Charles (MC) (Miller and Charles, 1991), Rubenstein & Goodenough (RG) (Rubenstein and Goodenough, 1965), WordSimilarity 353 (WS-353) (Finkelstein et al., 2001) and Simlex-999 (Leviant and Reichart, 2015). These four datasets were selected for being consensual gold-standards for the evaluation of semantic similarity and relatedness models.
The problem of measuring the semantic similarity and relatedness of two concepts can be stated as follows: given two concepts A and B, determine a numerical measure f(A, B) which expresses the semantic similarity or relatedness between concepts A and B. The notion of semantic similarity is associated with taxonomic (is-a) relations, while semantic relatedness represents more general relations. *Car* and *train* are examples of similar concepts (both share a common taxonomic ancestor, *vehicle*) while *car* and *wheel* are related concepts (a wheel is part of a car). As a consequence, semantic similarity is considered a particular case of semantic relatedness. Alternatively semantic similarity can also be defined as two concepts sharing a high number of salient features (attributes): synonymy (car/automobile), hyperonymy (car/vehicle), co-hyponymy (car/van/truck), while semantic relatedness can be defined as two words semantically associated without being necessarily similar: function (car/drive), meronymy (car/tyre), location (car/road), attribute (car/fast) (Freitas, 2015).
The gold standards are described below:

Pairs	Simlex-999	WS-353
closet - clothes	1.15	8.0

Table 1: Semantic Similarity vs Semantic Relatedness

- **Wordsimilarity 353:** WS-353 (Finkelstein et al., 2001) is certainly the most popular evaluation gold standard for distributional semantic models. The dataset is focused on semantic relatedness. The dataset contains two subsets: *set 1* (153 word pairs, evaluated by 13 subjects), and *set 2* (200 word pairs evaluated by 16 subjects) each one containing pairs from different parts-of-speech, a proper noun and pairs involving subjective bias.
- **Rubenstein & Goodenough:** RG (Rubenstein and Goodenough, 1965) contains 65 pairs which are often used to evaluate Distributional Semantic Models. RG reflects similarity of words rather than their relatedness. It is build by rating of 15 annotators to score the semantic similarity of each pair.
- **Miller & Charles:** MC (Miller and Charles, 1991) is a subset of 30 noun pairs from the RG gold standard which are re-annotated following new similarity guidelines. Ten pairs were selected from the highest level (between 3 and 4 on a scale from 0 to 4), ten pairs from the intermediate level (between 1 and 3), and ten pairs from the lowest level (0 to 1) of semantic similarity.
- **SIMLEX-999:** Simlex-999 (Hill et al., 2016; Leviant and Reichart, 2015) is aimed to measure how well Distributional Semantic Models capture similarity, rather than relatedness. Simlex-999 contains a range of 111 adjective, 666 noun and 222 verb pairs with an independent rating for each pair. It was built by using 500 annotators via Amazon Mechanical Turk.

4. SemR-11

The process of creating SemR-11 (Table 3) consisted in the translation of the three gold-standards WS-353, MC,

[1] https://www.mturk.com/mturk/welcome
[2] http://crowdflower.com/

Language	Parametes	MC	RG	WS-353	SIMLEX-999
German	# of Tokens	40	52	431	1094
	Vocabulary Size	40	52	431	1094
French	# of Tokens	37	45	430	1106
	Vocabulary Size	37	43	424	1097
Russian	# of Tokens	38	48	435	-
	Vocabulary Size	36	46	426	-
Italian	# of Tokens	34	43	426	1051
	Vocabulary Size	34	43	424	1051
Dutch	# of Tokens	37	45	426	1025
	Vocabulary Size	37	45	426	1018
Chinese	# of Tokens	37	51	471	-
	Vocabulary Size	37	51	471	-
Portuguese	# of Tokens	37	46	434	1149
	Vocabulary Size	37	46	434	1141
Swedish	# of Tokens	35	44	430	1002
	Vocabulary Size	35	44	430	995
Spanish	# of Tokens	35	44	437	993
	Vocabulary Size	35	44	437	991
Arabic	# of Tokens	38	54	448	-
	Vocabulary Size	36	49	448	-
Persian	# of Tokens	34	43	456	-
	Vocabulary Size	34	43	436	-

Table 2: The vocabulary and token distribution for each language of four gold-standards

Language	SemR-11				SE17 T2
	MC	RG	WS 353	Simlex 999	
German	✓	✓	✓	✓	✓
French	✓	✓	✓	✓	
Russian	✓	✓	✓		
Italian	✓	✓	✓	✓	✓
Dutch	✓	✓	✓	✓	
Chinese	✓	✓	✓		
Portuguese	✓	✓	✓	✓	
Swedish	✓	✓	✓	✓	
Spanish	✓	✓	✓	✓	✓
Arabic	✓	✓	✓		
Persian	✓	✓	✓		✓

Table 3: SemR-11 and its relation to existing multi-lingual gold standards.

English	Portuguese
hard;difficult;9.69	difícil;difícil;10
apparent;obvious; 9.08	visível;óbvio;9.15
disease;infection;7.08	doença;infecção;4.46

Table 4: Comparison between English and Portuguese gold standards.

RG for eleven languages and of the Simlex-999 for seven European languages (German, French, Italian, Dutch, Portuguese, Swedish and Spanish). Also SemR-11 has been compared with an existing multi-lingual gold standards [3]. The word pairs were translated by paid professional translators[4], skilled in data localisation tasks.
All translated pairs followed the protocol below:

1. Given a pair of words, translators should assume the most similar senses associated with the pair.
2. Translators should preserve the lexical category of the sense identified for that word.

In the end, 15,917 word pairs were translated to 11 languages. Table 2 quantifies the vocabulary and token distribution for each language.
The datasets are available on the Web[5].
The SemR-11 gold-standard assumes that the translations are preserving the similarity and relatedness scores of their original English human annotation. The target task was described to the human translators, who had access to the word pairs and scores.

English	French
woman;wife;283	Femme;femme;10
girl;child;4.77	Fille;enfant;5
understand;know;5.69	Comprendre;connaître;6.92

Table 5: Comparison between English and French gold standards

[3]SemEval-2017 Task 2
[4]Lionbridge Natural Language Solutions
[5]https://github.com/Lambda-3/Gold-Standards/tree/master/SemR-11

word-Pairs	MC	RG	WS-353	Simlex-999
English	food;rooster	monk;oracle	closet;clothes	clothes;closet
German	nahrung;hahn	mönch;orakel	Wandschrank;Kleidung	Kleider;Schrank
French	nourriture;coq	moine;oracle	cabinet;vêtements	vêtements;placard
Russian	еда;петух	монах;оракул	стенной шкаф;одежда	-
Italian	cibo;gallo	monaco;oracolo	ripostiglio;vestiti	vestiti;armadio
Dutch	voedsel;haan	monnik;orakel	kast;kleren	kleding;kast
Chinese	食物;公鸡	僧侣;甲骨文	壁橱;衣服	-
Portuguese	comida;galo	monge;oráculo	armário;roupas	roupas;roupeiro
Swedish	mat;tupp	munk;orakel	garderob;kläder	kläder;förråd
Spanish	comida;gallo	monje;oráculo	armario;ropa	ropa;armario
Arabic	طعام;ديك	راهب;وحي	خزانة;ملابس	-
Persian	غذا;خروس	راهب;وحی	گنجه;لباس	-

Table 6: Examples with all the languages for each of four datasets

Tables 4 and 5 show examples of translated pairs of Simlex-999 test collection (with the associated average similarity score) into Portuguese and French languages, respectively, while Table 6 provides example of word-pairs for each language and dataset.

5. Use Case

Distributional Semantic Models (DSM) are consolidating themselves as fundamental components for supporting automatic semantic interpretation in different application scenarios in natural language processing. From *question answering systems*, to *semantic search* and *text entailment*, distributional semantic models support a scalable approach for representing the meaning of words, which can automatically capture comprehensive associative commonsense information by analysing word-context patterns in large-scale corpora in an unsupervised or semi-supervised fashion (Freitas, 2015; Turney and Pantel, 2010b; Sales et al., 2016).

The SemR-11 test collection was used by Freitas et al.(2016), Sales et al.(2018) and Barzegar et al.(2018) to evaluate how different distributional semantic models built from corpora in different languages and with different sizes, perform in computing semantic relatedness similarity and relatedness tasks. Additionally, SemR-11 was used to analyze the role of machine translation approaches to support the construction of high-quality distributional vectors and computing semantic similarity & relatedness measures for other languages.

6. Acknowledgments

This publication has emanated from research funded in part from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645425 SSIX and Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289.

We would like in particular to thank Alexandros Poulis and Juha Vilhunen from the Global Services for Machine Intelligence Group, Lionbridge Finland [6] ensuring the production word of high quality translations for our similarity datasets.

7. Reference

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. Association for Computational Linguistics.

Barzegar, S., Davis, B., Handschuh, S., and Freitas, A. (2018). Multilingual semantic relatedness using lightweight machine translation. In *IEEE International Conference on Semantic Computing*. IEEE.

Bruni, E., Tran, N.-K., and Baroni, M. (2014). Multimodal distributional semantics. *J. Artif. Intell. Res.(JAIR)*, 49(2014):1–47.

Camacho-Collados, J., Pilehvar, M. T., Collier, N., and Navigli, R. (2017). Semeval-2017 task 2: Multilingual and cross-lingual semantic word similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017). Vancouver, Canada*.

Faruqui, M. and Dyer, C. (2014). Community evaluation and exchange of word vectors at wordvectors.org. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Baltimore, USA, June. Association for Computational Linguistics.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Freitas, A. and Curry, E. (2014). Natural language queries over heterogeneous linked data graphs: A distributional-compositional semantics approach. In *Proceedings of*

[6] https://www.lionbridge.com/en-us/global-services-for-machine-intelligence

the 19th international conference on Intelligent User Interfaces, pages 279–288. ACM.

Freitas, A., Curry, E., and O'Riain, S. (2012). A distributional approach for terminological semantic search on the linked data web. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 384–391. ACM.

Freitas, A., da Silva, J. C. P., Curry, E., and Buitelaar, P. (2014). A distributional semantics approach for selective reasoning on commonsense graph knowledge bases. In *International Conference on Applications of Natural Language to Data Bases/Information Systems*, pages 21–32. Springer.

Freitas, A., Barzegar, S., Sales, J. E., Handschuh, S., and Davis, B. (2016). Semantic relatedness for all (languages): A comparative analysis of multilingual semantic relatedness using machine translation. In *Knowledge Engineering and Knowledge Management: 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings 20*, pages 212–222. Springer.

Freitas, A. (2015). *Schema-agnositc queries over large-schema databases: a distributional semantics approach.* Ph.D. thesis, Digital Enterprise Research Institute (DERI), National University of Ireland, Galway.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hill, F., Reichart, R., and Korhonen, A. (2016). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Leviant, I. and Reichart, R. (2015). Separated by an un-common language: Towards judgment language informed vector space modeling. *arXiv preprint arXiv:1508.00106*.

Lin, J. (1991). Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *ICLR Workshop Papers*.

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, 12:1532–1543.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Sales, J. E., Freitas, A., Davis, B., and Handschuh, S. (2016). A compositional-distributional semantic model for searching complex entity categories. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 199–208.

Sales, J. E., Souza, L., Barzegar, S., Davis, B., Freitas, A., and Handschuh, S. (2018). Indra: A word embedding and semantic relatedness server. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Turney, P. D. and Pantel, P. (2010a). From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188.

Turney, P. D. and Pantel, P. (2010b). From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188, January.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics.

Corpora with Part-of-Speech Annotations for Three Regional Languages of France: Alsatian, Occitan and Picard

Delphine Bernhard[1], Anne-Laure Ligozat[2], Fanny Martin[3], Myriam Bras[4], Pierre Magistry[5], Marianne Vergez-Couret[6], Lucie Steiblé[1], Pascale Erhart[1], Nabil Hathout[4], Dominique Huck[1], Christophe Rey[7], Philippe Reynés[3], Sophie Rosset[5], Jean Sibille[4], Thomas Lavergne[8]

[1]LiLPa, Université de Strasbourg, France
[2]LIMSI, CNRS, ENSIIE, Université Paris-Saclay, F-91405 Orsay, France
[3]Laboratoire Habiter le Monde - HM - EA 4278, Université de Picardie Jules Verne, Amiens, France
[4]CLLE, Université de Toulouse, CNRS, UT2J, France
[5]LIMSI, CNRS, Université Paris-Saclay, F-91405 Orsay, France
[6]Queen's University, Belfast
[7]LT2D (Lexiques, Textes, Discours, Dictionnaires), Université de Cergy-Pontoise, IUF
[8]LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, F-91405 Orsay, France

[1]{dbernhard,lucie.steible,pascale.erhart,huck}@unistra.fr
[2,5,8]{annlor,magistry,sophie.rosset,thomas.lavergne}@limsi.fr
[3]{fanny.martin,philippe.reynes}@u-picardie.fr
[4]{myriam.bras,nabil.hathout,jean.sibille}@univ-tlse2.fr
[6]m.vergez-couret@qub.ac.uk
[7]christophe.rey@u-cergy.fr

Abstract

This article describes the creation of corpora with part-of-speech annotations for three regional languages of France: Alsatian, Occitan and Picard. These manual annotations were performed in the context of the RESTAURE project, whose goal is to develop resources and tools for these under-resourced French regional languages. The article presents the tagsets used in the annotation process as well as the resulting annotated corpora.

Keywords: corpus, annotation, part-of-speech, Alsatian, Occitan, Picard

1. Introduction

Since the constitutional amendment (Article 75-1) published in 2008, regional languages are officially part of the heritage of France, although the only official language in France is French (Article 2). As such, it is essential to implement modern means for their preservation and transmission, relying particularly on digital technologies. Regional languages of France can be considered as low-resourced, that is to say there are no or only few electronic resources (corpora, lexicons, dictionaries) and tools. All languages with little resources have in common that their computerisation has a low financial profitability which does not compensate for considerable development costs. However, endowing these languages with electronic resources and tools is a major concern for their dissemination, protection and teaching (including for new speakers). Automatic tools help ensure data collection (scanning), storage in standardized formats, categorization and retrieval. Moreover, the availability of digital data and automatic processing tools can transform the attitude of speakers towards regional languages, in particular increase the use of written material that often remains marginal. In a broader perspective, it is the diversity of world languages which would be better preserved and the amount of data available to researchers in human and social sciences (linguistics, sociology, anthropology, literature, history, ...) would increase (Soria et al., 2013).

The overall objective of the RESTAURE[1] project is to provide computational resources and processing tools for three regional languages of France: Alsatian, Occitan and Picard. The three of them belong to the languages of France listed in Cerquiglini's 1999 report (Cerquiglini, 1999), which inventories the regional languages of France within the meaning of the European Charter for Regional or Minority Languages. They have no official status in France and as such, have suffered from a lack of institutional support until recently. The goal of this project is to bring these languages to the front and foster NLP research on these languages. Processing natural language is complex and the development of NLP tools requires significant resources, both human and financial. This explains the lack of such tools for regional languages of France.

In this work, we present the methodology used to create corpora annotated with part-of-speech information for the three regional languages considered in RESTAURE. The main challenge for writing the annotation guidelines was the lack of comprehensive grammatical descriptions, encompassing all the dialectal variants found in our corpora. In addition, the annotation triggered further discussion on tokenisation issues and the use of POS tagsets and taggers developed for closely related languages. The whole process was made possible thanks to the close cooperation between

[1]http://restaure.unistra.fr/

linguists and NLP specialists, as well as the parallel and collaborative work on three different languages facing similar challenges.

2. Description of Alsatian, Picard and Occitan

In this section, we briefly describe the three French regional languages considered in the project, in particular with respect to their morpho-syntactic properties.

2.1. Alsatian

Alsatian is spoken in North-Eastern France and is part of the High German dialects, which are subdivided into Central German and Upper German. The majority of the Alsatian dialects belongs to (Low) Alemannic, an Upper German dialect. A small part of the dialectal space (North-West) belongs to Central German Rhine Franconian. Like all dialectal, phonological and, partially, lexical spaces, the Alsatian dialects are characterized by *spatial variation*, which is the main characteristic of a dialect. Since the second half of the 20th century, the writers and speakers of Alsatian have had, in their vast majority, a plurilingual repertoire, with French taking more and more importance and driving them to use, in Alsatian, linguistic strategies and *calques* (loan translations) from French. The fundamental morphosyntactic characteristics are little affected by this phenomenon. They are common to all Alsatian dialects and are very similar to those of standard German. Finite verbs are marked with morphemes of tense, mode and person-number and noun phrases are marked with number, case and gender. The tense and mode system is much simpler to standard German and there is only one person number marker for the plural persons (Huck, to appear). However, the surface form of the morphemes can present intradialectal variations. The Alsatian dialects are used mainly orally and written production is limited to some literary works (poetry, theater plays), linguistic descriptions (dictionaries, lexicons), small contributions to otherwise French publications (chronicles in newspapers) and online texts (Wikipedia, social networks). What is more, several spelling conventions have been proposed, but none of them can be considered as a widely accepted and used standard.

2.2. Occitan

Occitan, or Oc language, is spoken in a large area in the south of France, in several valleys of Italy and the Aran valley in Spain. Occitan is not a unitary language, it has several varieties, organized in 6 large dialects (Auvernhàs, Gascon, Lengadocian, Lemosin, Provençau, and Vivaro-aupenc). It is a Romance language: as such, it shares many morpho-syntactic properties with other Romance languages (e.g., number and genre inflection marks on all the items of the noun phrase ; tense, person, number inflection marks on finite verbs). It is much closer to Catalan than to French: it is for example a null subject language as all the other Romance languages except French and oïl languages as Picard, Francoprovençal, Rheto-Romance languages and North-Italian dialects. Unlike French and Picard, Occitan has different verb inflection marks for each person. The morpho-syntactic level is also affected by variation across dialects (e.g., verbal inflection varies from one dialect to the other). As far as spelling is concerned, Occitan is not standardized as a whole but has two major spelling standards: the classical system, inspired from the troubadour's medieval spelling, and another system, closer to French conventions (Sibille, 2002).

2.3. Picard

The linguistic area of Picard includes the Hauts-de-France administrative region, and the Hainaut province in Belgium. Picard is an oïl language, which also belongs to the larger Romance language group. It differs from French with respect to several aspects. Word order can be different: for example, *il o foait keud assé* in Picard translates to *il a fait assez chaud* (*it has been quite hot*) –in French, the adverb is placed before the adjective, while in Picard the adjective is placed before the adverb. Even if Picard and Occitan are both Romance languages, Picard is closer to French concerning inflection. As in French and the other langues d'oïl, gender and number are mainly marked in Picard by means of determiners at the level of the noun phrase. Another device is shared with the other langues d'oïl: in the verbal phrase, a subject personal pronoun must be used in order to express personal rank in a deflective way. Picard does not have a unitary standardized spelling system and Picard texts can contain a dot which must not be considered as a word boundary, e.g., *lon.mint* (*for a long time)*, *erwet.tent* (*look*), *fin.mes* (*women*). More often than in French, words can also contain apostrophes – *c'min* (*path*) – and hyphens – *gardin-neux* (*gardeners*).

3. Elaboration of the Tagsets

As we wanted to exploit the proximity to better-resourced languages, an important issue is that of the tagset, i.e. the list of part-of-speech categories used for the manual and, afterwards, automatic annotation. One solution would have been to use the tagsets from annotated corpora and part-of-speech annotation tools for closely-related languages, such as the German TreeTagger or Stanford Tagger for Alsatian. However, these tagsets are usually very detailed: the German TreeTagger and StanfordTagger use a set of 54 tags,[2] the French TreeTagger identifies 33 tags[3] etc. Such level of detail is not necessarily needed and entails several drawbacks: higher cost for training annotators and reduced performance for the part-of-speech (POS) taggers, which have to discriminate between very similar categories. Furthermore, we wanted to create corpora annotated with the same standard tagset if possible, in order to facilitate the diffusion of the corpora, to be able to compare our experiments with state-of-the-art work and to enable comparison between the languages of the project. We thus chose to base the tagsets for Alsatian, Occitan and Picard on the universal POS tags defined in the context of the Universal Dependencies project (Nivre et al., 2016).[4] A first issue was to

[2] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/stts_guide.pdf

[3] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/french-tagset.html

[4] http://universaldependencies.org/u/pos/index.html

Tag	Full name
ADJ	adjective
ADP	adposition
ADP+DET	**preposition-determiner contraction**
ADV	adverb
AUX	auxiliary
CCONJ	coordinating conjunction
DET	determiner
EPE	**epenthesis**
INTJ	interjection
MOD	**modal verb**
NOUN	common noun
NUM	numeral
PART	particle
PRON	pronoun
PROPN	proper noun
PUNCT	punctuation
SCONJ	subordinating conjunction
SYM	symbol
VERB	verb
X	other

Table 1: Common tagset. The tags which are not part of the Universal POS tags are in bold format.

evaluate this tagset with respect to our languages and check that it suits our needs.

3.1. Unified Tagset

The Universal POS tags version 2 contains 17 core part-of-speech categories. Yet, we introduced three additional tags: ADP+DET for contractions of a preposition and a determiner; MOD for modal verbs; and EPE for epenthesis.[5] Epenthesis is found in our corpora because oral phenomena are often preserved. These tags were added for ease of annotation –we could have used features with existing tags instead, but it was easier to have only one level of annotation– but can be projected to Universal POS tags: ADP+DET can be split into ADP and DET, MOD becomes AUX, and EPE becomes X. The resulting tagset is presented in Table 1.

3.2. Annotation Guidelines

For guiding the manual annotation, we tried to follow the Universal POS tag documentation as much as possible, but also considered the choices made for closely-related languages. Annotation guidelines were created for all three languages.

Alsatian The documentation for the Universal POS tags was the foundation for the guidelines (Bernhard et al., 2018). The descriptions of the STTS tags for German (Schiller et al., 1999) were consulted for some decisions. Alsatian grammars were also used (Jenny and Richert, 1984; Jung, 1983). Universal POS tags documentation was mostly followed, except for the three added tags mentioned before, and the FM tag for foreign words (see Table 2). Yet, some choices were distinct from Universal

[5]Addition of one or several letters to comply with the phonotactics of the language, for example in Alsatian: *fànga␣-n-/EPE ␣à drucka* (en: *begin to print*).

Tag	Full name
ADJ	adjective
ADP	adposition
ADV	adverb
APPART	preposition-determiner contraction
AUX	auxiliary
CONJ	coordinating conjunction
DET	determiner
EPE	epenthesis
FM	*foreign words*
INTJ	interjection
MOD	modal verb
NOUN	common noun
NUM	numeral
PART	particle
PRON	pronoun
PROPN	proper noun
PUNCT	punctuation
SCONJ	subordinating conjunction
SYM	symbol
VERB	verb
X	other

Table 2: Alsatian tagset. The FM tag is not part of the common tagset.

Tag level 1	Tags level 2	Full name
A	Af, Ao, Ak, Ai, As	adjective
C	Cc, Cs	conjunction
D	Da, Dd, Di, Ds, Dt, Dr, Dk, Dp	determiner
F		punctuation
I		interjection
N	Nc, Np, Nk	noun
P	Pp, Pd, Pi, Ps, Pt, Pr, Px, Pk	pronoun
R	Rg, Rx, Rp, Rq	adverb
S	Sp, Spda, Sd	preposition
V	Vm, Va	verb
X		residual

Table 3: Occitan tagset

POS tags: for example, verb particles separated from their verb were tagged as PART, like in the STTS guidelines (*Er nùtzt/VERB mich üss/PART*, en: *He exploits me*).

Occitan The description of morpho-syntactic tags in the lexicon of inflected forms LOFLOC (Vergez-Couret, 2016) was adapted and extended to create annotation guidelines (Bras, 2018). The standard GRACE tagset (Rajman et al., 1997), which comes from the MULTEXT (Ide and Véronis, 1994) and EAGLES (von Rekowski, 1996) tagsets, was chosen, as it has been used for several similar annotated corpora for French and Catalan. We created a conversion script to project the GRACE tags onto Universal POS tags. Table 3 shows the two levels of this tagset (38 tags); the complete description of the tags is given in the guidelines (Bras, 2018).

Tag	Full name
ADJ	adjective
ADP	preposition
ADPDET	preposition-determiner contraction or partitive
ADPLOC	prepositional locution
ADV	adverbs
CCONJ	coordinating conjunction
SCONJ	subordinating conjunction
DET	determiner
EPE	epenthesis
INTJ	interjection
NOUN	common noun
NOUNCCOMP	composed common noun
PROPN	proper noun
NUM	cardinal numbers
PRONDEM	demonstrative pronoun
PRONIND	indefinite pronoun
PRONPERS	personal pronoun
PRONPOSS	possessive pronoun
PRONREL	relative pronoun
PRONINT	interrogative pronoun
PART	particle or other function
PUNCT	punctuation
SYM	symbol
VERBINF	infinitive verb
VERBCONJ	conjugated verb
VERBPP	past participle verb
VERBPPR	present participle verb
X	other: loan word, typo, abbreviation...

Table 4: Picard tagset. The tags not found in our common tagset are in italics. The SYM tag was not encountered in the corpus

Picard The annotation guidelines are based on the Universal POS tags and the French Treebank annotation guidelines (Abeillé and Clément, 2003). A specific documentation was nevertheless created, to take into account the specifics of the Picard language, as well as to collect the possible issues and research topics (Martin et al., 2018). Some adjustments were made with respect to the Universal POS tags, namely subcategories, to obtain a better description of the Picard language (see Table 4). A conversion script was created to generate Universal POS tags from these more specific tags.

4. Constitution and Annotation of the Corpora

4.1. Corpus Selection

The first step was to collect texts with rights to distribute without restriction, so that the corpora could be made available. For some texts, we scanned printed texts and performed OCR.[6] One of the main challenges was to obtain resources which represent a large variety of textual genres and geolinguistic variants.

[6]A specific work was performed on OCR for the three languages, but this work is out of the scope of this paper.

Alsatian The annotated corpus is composed of two main sources: WKP – Wikipedia articles from the Alemannic Wikipedia[7] and HRM – chronicles written in an information magazine published by the Haut-Rhin department (southern Alsace) General Council. In addition, two more specific genres were used for the annotator training phase: one excerpt from a theater play and some recipes. Given that the Alemannic Wikipedia contains articles written in several dialects from the Alemannic linguistic area, we only used articles which were specifically categorized as being written in Alsatian.

Occitan The RESTAURE project led to the finalization of the BaTelÒc text base (Bras and Vergez-Couret, 2016).[8] BaTelÒc is a wide coverage text collection, with written texts of literature (prose, drama and poetry) and other genres such as technical texts and newspapers, and embraces dialectal and spelling variations. 3.7 million words have already been gathered. All the texts in the base are encoded according to XML TEI P5 format. As the texts contained in BaTelÒc pose copyright issues, we selected 55 extracts of 60 words maximum from 17 texts from different authors to create the annotated corpus. We also added 8 texts from the online Occitan newspaper Lo Jornalet [9] with their kind permission. Lo Jornalet contains texts mostly in Lengadocian, and some in Gascon (all in Alibert's classical norm); we selected several texts from each dialect to complete the Occitan corpus. Finally, we also included one text from the Ciel d'òc online virtual library.[10]

Picard We benefited from the textual resources already collected within the PICARTEXT project,[11] a large literary resource. This text database is panchronic and has in its current version, which is still evolving, about 8 million tokens, taken from literature and ranging from the 17th century to the 21st century. The PICARTEXT base was tagged in XML according to the guidelines of the TEI P5. One of our objectives was to enrich this first textual database with literary texts of various genres (poetry, theater, tales, short stories, novels, etc.), of various time periods and taking into account the different varieties of Picard. We selected a subset of 32 texts according to the project criteria: diachronic diversity, variety of dialects and genres.

Table 5 provides the sizes of each corpus and sub-corpus.

4.2. Corpus Preparation

The selected texts were specifically prepared for the manual annotation process.

Alsatian Given the proximity of Alsatian to Standard German, and in order to facilitate the annotation work, the texts were pre-tagged using the TreeTagger (Schmid, 1994) for German and available lexicons for the Alsatian dialects. In details, the following pre-processing steps were performed:

[7]http://als.wikipedia.org
[8]http://redac.univ-tlse2.fr/bateloc/
[9]https://www.jornalet.com/
[10]http://www.cieldoc.com/
[11]http://www.u-picardie.fr/LESCLaP/PICARTEXT/Public/

Lang.	Source	Tokens	Types
Alsatian	WKP (13 doc.)	8,432	3,129
	HRM (6 doc.)	3,542	1,345
	recipes (1 doc.)	364	203
	theater (1 doc.)	232	140
	total (annotated)	12,570	4,497
Occitan	Jornalet (Lengadocian)	927	261
	Jornalet (Gascon)	2403	339
	BaTelÒc (Lengadocian)	5802	1455
	Ciel d'òc (Lengadocian)	538	219
	BaTelÒc (Gascon)	1630	514
	BaTelÒc (Provençau)	1359	390
	BaTelÒc (Lemosin)	462	170
	BaTelÒc (Vivaro-Aupin)	501	145
	BaTelÒc (Auvernhàs)	1386	296
	total (annotated)	15,008	3788
Picard	narrative	8,372	2,066
	poetry	1,924	585
	theater	862	304
	total (annotated)	11,158	2,564

Table 5: Reference corpus

1. Tokenisation using a custom tokenizer for Alsatian (Bernhard et al., 2017).
2. Annotation with the TreeTagger for German in order to identify "unknown words", i.e. words which do not belong to the German TreeTagger lexicon and hence are typically Alsatian.
3. Automatic creation of a custom TreeTagger lexicon by looking up these unknown words in available Alsatian lexicons. Since there is a great amount of spelling variation in written Alsatian, we perform approximate lookup using a variant of the Double Metaphone phonetic algorithm adapted to Alsatian (Bernhard, 2014). This allows us to retrieve POS category information for the words even if they do not appear with exactly the same spelling in the lexicon (e.g. *'Sùnneblüem'* and *'Sunneblüem'*). In order to increase lexicon coverage, we also perform approximate lookup in German lexicons.
4. Transformation of Alsatian spellings for closed class words into their German equivalent in the texts using a custom correspondence dictionary (e.g., Alsatian *nìt* corresponds to Standard German *nicht*). We have shown in previous work that this improves the performance of the German TreeTagger when used for tagging Alsatian (Bernhard and Ligozat, 2013).
5. Second annotation with the German TreeTagger performed on these transformed texts. We provide TreeTagger with the custom lexicon containing suggested categories for unknown words.
6. Transformation of the result of this second annotation into the input format requested by the manual annotation tool, using a correspondence table between our POS tags and the German TreeTagger POS tags.

Occitan Pre-processing was also used:

1. We first used the POS tagger of the APERTIUM translation platform used in the Occitan/Spanish and Occitan/Catalan translators (Armentano I Oller, 2008) to tag an initial corpus in one dialect (Lengadocian). A specific tokeniser for Occitan and a specific inflectional lexicon (LOFLOC) were created to adapt the tagger to our needs, and APERTIUM tags were converted to GRACE tags with a specific script.
2. This first tagger's outputs were manually corrected on a subcorpus.
3. Then, a supervised machine learning tagger, Talismane (Urieli, 2013), was trained on the corrected corpus.
4. The rest of the corpus was annotated with Talismane.
5. Talismane outputs were manually corrected and used for further annotation (Bras and Vergez-Couret, 2014).

Picard We randomly extracted 30 lines excerpts from each of the selected texts. The first issue was tokenisation, which prompted the development of a tokenisation script for Picard. Bernhard et al. (2017) detail the specific issues of tokenisation for Picard, as well as the choices made. In contrast to Alsatian and Occitan, the Picard corpus was not pre-annotated.

4.3. Annotation Methodology

Alsatian The pre-tagged texts were manually corrected with the Analog tool (Lay and Pincemin, 2010). In addition to the POS tags, the annotators were also requested to provide a gloss (translation into French), the lemma, grammatical properties for verbs, nouns and adjectives, as well as location named entities but these further pieces of information will not be discussed in this paper. Overall, 6 persons took part in the annotation. One annotator (A1) annotated all the 21 documents. In order to measure inter-annotator agreement, two annotators (A2 and A3) annotated respectively 6 and 5 of the 21 documents. Finally, annotator A4 made the final decisions and corrections (adjudication) for all the 21 documents. A4 was helped by two experts in the Alsatian dialects (A3 and A6) to solve difficult issues and was also provided with correction proposals by annotator A5 for 7 of the documents annotated by A1. In addition, online resources were consulted for the adjudication: dictionaries (*Wörterbuch der elsässischen Mundarten* (Martin and Lienhart, 1899 1907)[12] ; *DWDS – Digitales Worterbuch der deutschen Sprache*[13]) and the Universal Dependencies version 2.0 German corpus accessed through the search interface by the University of Turku.[14]

We measured the inter-annotator agreement (IAA) between annotators A1 to A3 and the final adjudication by A4. Agreement was measured in terms of percentage agreement and the Kappa coefficient κ (Cohen, 1960), computed with the `irr` R package.[15] Table 6 details these IAA values.

[12] `http://woerterbuchnetz.de/ElsWB/`

[13] `https://www.dwds.de/`

[14] `http://bionlp-www.utu.fi/dep_search`

[15] `https://cran.r-project.org/package=irr`.

	A1		A4		# Tokens
	%	κ	%	κ	
A1			92.8	0.920	12,570
A2	84.1	0.824	84.6	0.830	2,135
A3	93.0	0.922	93.7	0.930	2,638

Table 6: Inter-annotator agreement for the Alsatian corpus.

	Beginning		Middle	
	%	κ	%	κ
O1 - O2	88.9	0.873	95.3	0.947
O1 - O3	93.8	0.929	94.7	0.940
O2 - O3	88.1	0.864	94.7	0.940
O1 - O2 - O3	87.0	0.889	92.3	0.942
#Tokens	370		169	

Table 7: Inter-annotator agreement for the Occitan corpus .

Overall, the IAA observed for A1 and A3 was better than for A2. Lower agreement is usually observed for rarer POS tags (e.g. INTJ, SYM, SCONJ). It should be noted that the documents annotated by A2 belong mostly to the training corpus (4 out of 6 texts) and these annotations were performed early in the annotation process. The annotation guide was substantially modified after this first annotation phase by introducing the APPRART, MOD and FM categories. Also, the EPE POS tag was introduced very late in the annotation process and was annotated as X before. In the future, some of the verifications performed by A4 could be automated (e.g. detect missing tags), and integrated to corpus pre-processing and annotation guidelines.

Occitan A first Lengadocian corpus was tagged with the first tagger described in 4.2. and then manually corrected by four annotators. We then trained the Talismane tagger to annotate a bigger corpus including texts in two dialects, Lengadocian and Gascon (Vergez-Couret and Urieli, 2014). This corpus was manually corrected by two annotators. Then we trained Talismane again in order to be able to pre-annotate the corpora including the 6 dialects of the Occitan language. The annotations were corrected by three annotators (O1, O2 and O3) using the Analog tool in order to get the final corpus. Inter-annotator agreement was measured at two different time points: at the beginning and at the middle of the manual annotation phase (see Table 7). The agreement has improved over the time, in particular for annotator pairs O1 - O2 and O2 - O3.

Picard The Picard corpus was annotated in a csv format. The texts were manually tokenised, and each token was annotated with its part-of-speech, its lemma and its French translation in context. Three Picard speakers worked on the manual annotation. A first manual annotation of 20 texts was performed by annotator P1, and then discussed and modified if necessary with annotator P2. The issues detected in this first step were then discussed with the research team in order to adapt the tagset and the guidelines. The remaining annotations were made by P2. All annotations were reviewed by annotators P2 and P3. The annotation phase took place during a period of about 11 months, and each step required additional research: tokenisation requires morphological and grammatical studies taking into account the Picard variety since tokenisation rules differ according to the language variety. The issues of lemmas and translations are interdependent. Moreover, Picard dictionaries are not comprehensive, and a translation for a text in a particular Picard variety could be found in a dictionary of another Picard variety. Finding the right translation thus often required searching in all available dictionaries. Since P2 performed most of the annotation, we measured intra-annotator agreement for several versions of the corpus.[16] For the June 2016 and January 2017 versions, $\kappa = 0.922$ with a percent agreement of 92.9%, the differences being mostly caused by typos and changes in the tagset. Between the January 2017 and the July 2017 version, $\kappa = 0.784$ with a percent agreement of 80.2%. These lower figures are explained mostly by the more important changes in the tagset, the figures for stable tags remaining higher than 0.9 (0.967 for the ADP tag, 1.000 for the PROPN tag...). Finally, a verification script was applied to the corpus to check its coherence: if a word is labeled with different tags, whose distribution is very uneven, the contexts (i.e. preceding and following tag) are compared and if a same context leads to different tags, the annotation is checked manually.

4.4. Resulting Resources and Dissemination

Token	French	Lemma	Tag	*English*
Spàrichle	asperge	Spàrichel	NOUN	asparagus
ìn	dans	ìn	ADP	in
e	une	e	DET	a
Sìbb	passoire	Sìbb	NOUN	sieve
üss	de	üss	ADP	of
Metàll	métal	Metàll	NOUN	metal
màche	mettre	màche	VERB	put

Table 8: Annotation example for Alsatian (some additional annotations are not presented) for the sentence.

Token	Lemma	Tag level 1	Tag level 2	*English*
Los	lo	D	Da	the
cavals	caval	N	Nc	horses
èran	èsser	V	Vm	were
luènh	luènh	R	Rg	far away
.	.	F		

Table 9: Annotation example for Occitan.

Table 8 shows an annotation example for Alsatian, Table 9 for Occitan, and Table 10 for Picard. Each line represents a token, and the colums contain the different annotations (POS tag, lemma, French translation). The columns with the English translations are not available in the corpora and are provided for the sake of readability.

Authors: Matthias Gamer, Jim Lemon, Ian Fellows Puspendra Singh.

[16] For one text of the corpus only, because changes were made in the text excerpts and tokenization, which makes it difficult to perform a completely automatic evaluation.

Token	Tag	Lemma	French	*English*
I	PRONPERS	i	il	he
avot	VERBCONJ	avoir	avait	had
fauqu'	VERBPP	fauquer	coupé	cut
chés	DET	euch	les	the
projecteurs	NOUN	projecteur	projecteurs	spotlights
qu'	PRONREL	qui	qui	that
is	PRONPERS	i	ils	they
illuminottent	VERBCONJ	illuminoter	éclairaient	lit
eul	DET	euch	la	the
fosse	NOUN	fosse	fosse	pit
.	PUNCT	.	.	.

Table 10: Annotation example for Picard

The annotation guidelines and the corpora are available for all three languages on the Zenodo platform, in the RESTAURE project community (see Section 9. for the corpus list).[17]

5. Related Work

Creating annotated corpora for under-resourced languages presents several difficulties. First, it requires assembling a large textual corpus, which can be a challenge for these languages which have few electronic resources. Work on part-of-speech tagging for under-resourced languages is often based on parallel corpora, following (Yarowsky et al., 2001), but there are no such existing electronic corpora for the three languages considered.

Then, a tagset has to be created or adapted to the language, which requires linguistic expertise. Finally, the annotation also requires annotators with linguistic expertise. Crowdsourcing can be used for part-of-speech annotation (Hovy et al., 2014), and was even used for Alsatian (Millour et al., 2017). Yet, crowdsourcing necessitates an adapted platform, and communication to possible speakers, who for example in the case of Picard, are rare. A possible direction for POS tagging could be to create a minimum tag dictionary for the most frequent word types, such as used by (Garrette and Baldridge, 2013). This kind of approach still requires a test corpus to evaluate the tagger; and the performance remains low compared to more resourced languages.

6. Conclusion

We have presented our methodology for producing corpora with POS annotations for three regional languages of France, namely Alsatian, Occitan and Picard. The tagsets are based on an extended version of the Universal POS tags, with some language-specific additions to account for particular linguistic phenomena. The annotation guidelines as well as the manually annotated corpora are freely available. We plan to use these corpora to develop part-of-speech taggers accommodating the spatial variation encountered in the three languages.

7. Acknowledgements

This work was supported by the French "Agence Nationale de la Recherche" (ANR) through the RESTAURE project (no.: ANR-14-CE24-0003). We are grateful to the annotators: Clément Dorffer, Gwendoline Hollner, Aurélie Abadie, Louise Esher, Sébastien Gonzales. We would also like to thank those who kindly permitted us to include their texts in our corpora: Yves Bisch, Conseil départemental du Haut-Rhin, OLCA, Laurenç Gosset and l'Institut d'Estudis Occitans, Tricio Dupuy and Ciel d'Òc, Sèrgi Viaule, Eric Chaplain and Editions des Régionalismes, Maurici Romieu and Reclams, Benaset Dazeas and Lo Congrès Permanent de la lenga occitana.

[17] https://zenodo.org/communities/restaure/

8. Bibliographical References

Abeillé, A. and Clément, L. (2003). Annotation morphosyntaxique. Les mots simples - Les mots composés. Technical report, LLF, Université Paris 7.

Armentano I Oller, C. (2008). Traduction automatique occitan-catalan et occitan-espagnol: difficultés affrontées et résultats atteints. In *IXème Congrès International de l'Association Internationale d'Etudes Occitanes*, Aachen.

Bernhard, D. and Ligozat, A.-L. (2013). Hassle-free POS-Tagging for the Alsatian Dialects. In Marcos Zampieri et al., editors, *Non-Standard Data Sources in Corpus Based-Research*, ZSM Studien, pages 85–92. Shaker. Volume 5.

Bernhard, D., Todirascu, A., Martin, F., Erhart, P., Steiblé, L., Huck, D., and Rey, C. (2017). Problèmes de tokénisation pour deux langues régionales de France, l'alsacien et le picard. In *DiLiTAL 2017*, Actes de l'atelier " Diversité Linguistique et TAL ", pages 14–23, Orléans, France, Jun.

Bernhard, D., Erhart, P., Huck, D., and Steiblé, L., (2018). *Part-of-Speech Annotation Guidelines for the Alsatian Dialects*. DOI: 10.5281/zenodo.1171925.

Bernhard, D. (2014). Adding Dialectal Lexicalisations to Linked Open Data Resources: the Example of Alsatian. In *Proceedings of the Workshop on Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era (CCURL 2014)*, pages 23–29, Reykjavík, Iceland, May.

Bras, M. and Vergez-Couret, M. (2014). Annotation morphosyntaxique d'un corpus de textes occitans: l'expérience de BaTelÒc. In *XIème Congrès de l'Association Internationale d'Etudes Occitanes*, Lerida, Spain, June.

Bras, M. and Vergez-Couret, M. (2016). BaTelÒc: A text base for the Occitan language. In Vera Ferreira et al., editors, *Language Documentation and Conservation in Europe*, pages 133–149. Honolulu: University of Hawaï Press.

Bras, M., (2018). *Part-of-Speech Annotation Guidelines for the Occitan Language*, February. DOI: 10.5281/zenodo.1173113.

Cerquiglini, B. (1999). Les langues de la France. Technical report, Rapport au Ministre de l'Éducation Nationale, de la Recherche et de la Technologie et à la Ministre de la Culture et de la Communication.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Garrette, D. and Baldridge, J. (2013). Learning a Part-of-Speech Tagger from Two Hours of Annotation. In *HLT-NAACL*, pages 138–147.

Hovy, D., Plank, B., and Søgaard, A. (2014). Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 377–382.

Huck, D. (to appear). Dialectal speech in Alsace / Alsatian. In Hans Boas, et al., editors, *Varieties of German Worldwide*, volume 1. Cambridge University Press, Cambridge.

Ide, N. and Véronis, J. (1994). Multext (Multilingual Tools and Corpora). In *14th Conference on Computational Linguistics (COLING'94), Kyoto, Japan*.

Jenny, A. and Richert, D. (1984). *Précis pratique de grammaire alsacienne: en référence principalement au parler de Strasbourg*. Librairie Istra.

Jung, E. (1983). *Grammaire de l'alsacien: dialecte de Strasbourg avec indications historiques*. Oberlin.

Lay, M.-H. and Pincemin, B. (2010). Pour une exploration humaniste des textes: AnaLog. In Sergio Bolasco, et al., editors, *Proceedings of 10th International Conference Journées d'Analyse statistique des Données Textuelles*, pages 1045–1056, Sapienza University of Rome.

Martin, E. and Lienhart, H. (1899-1907). *Wörterbuch der elsässischen Mundarten*, volume 1-2. KJ Trubner.

Martin, F., Rey, C., and Reynés, P., (2018). *Part-of-Speech Annotation Guidelines for Picard*. DOI: 10.5281/zenodo.1173428.

Millour, A., Fort, K., Bernhard, D., and Steiblé, L. (2017). Vers une solution légère de production de données pour le TAL: création d'un tagger de l'alsacien par crowdsourcing bénévole. In *Actes de Traitement Automatique des Langues Naturelles (TALN)*.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, may.

Rajman, M., Lecomte, J., and Paroubek, P. (1997). Format de description lexicale pour le français. Partie 2 : Description morpho-syntaxique. Technical report, EPFL & INaLF. GRACE GTR-3-2.1.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS. Technical report, Universität Stuttgart & Universität Tübingen.

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49.

Sibille, J. (2002). Ecrire l'occitan : essai de présentation et de synthèse. In Dominique Caubet, et al., editors, *Codification des langues de France*, pages 17–37. L'Harmattan, Paris, France.

Soria, C., Mariani, J., and Zoli, C. (2013). Dwarfs sitting on the giants' shoulders—how LTs for regional and minority languages can benefit from piggybacking major languages. In *Proceedings of XVII FEL Conference*, pages 73–79.

Urieli, A. (2013). *Robust French syntax analysis: reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. Ph.D. thesis, Université de Toulouse II le Mirail.

Vergez-Couret, M. and Urieli, A. (2014). Pos-tagging different varieties of Occitan with single-dialect resources. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages Varieties and Dialects*. Association for Computational Linguistics and Dublin City University.

Vergez-Couret, M. (2016). Description du lexique Loflòc. Research report, CLLE-ERSS, Apr.

von Rekowski, U. (1996). ELM-FR: A typed French incarnation of the EAGLES-TS – Definition of Lexical Specification and Classification Guidelines. Technical report, GSI-Erli.

Yarowsky, D., Ngai, G., and Wicentowski, R. (2001). Inducing Multilingual Text Analysis Tools via Robust Projection across Aligned Corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8.

9. Language Resource References

Bernhard, D., Erhart, P., Huck, D., and Steiblé, L. (2018). Annotated Corpus for the Alsatian Dialects. DOI: 10.5281/zenodo.1170129.

Bras, M., Esher, L., Sibille, J., and Vergez-Couret, M. (2018). Annotated Corpus for Occitan. DOI: 10.5281/zenodo.1182949.

Martin, F., Rey, C., and Reynés, P. (2018). Annotated Corpus for Picard. DOI: 10.5281/zenodo.1172576.

Part-of-Speech Tagging for Arabic Gulf Dialect Using Bi-LSTM

Randah Alharbi[1,2], Walid Magdy[1], Kareem Darwish[3], Ahmed AbdelAli[3], Hamdy Mubarak[3]
[1]School of Informatics, University of Edinburgh
[2]College of Computer Science & Information Systems, Umm Al-Qura University
[3]QCRI, Hamad bin Khalifa University
raharbi@uqu.edu.sa, wmagdy@inf.ed.ac.uk
{kdarwish,hmubarak,aabdelali}@hbku.edu.qa

Abstract

Part-of-speech (POS) tagging is one of the most important addressed areas in the natural language processing (NLP). There are effective POS taggers for many languages including Arabic. However, POS research for Arabic focused mainly on Modern Standard Arabic (MSA), while less attention was directed towards Dialect Arabic (DA). MSA is the formal variant which is mainly found in news and formal text books, while DA is the informal spoken Arabic that varies among different regions in the Arab world. DA is heavily used online due to the large spread of social media, which increased research directions towards building NLP tools for DA. Most research on DA focuses on Egyptian and Levantine, while much less attention is given to the Gulf dialect. In this paper, we present a more effective POS tagger for the Arabic Gulf dialect than currently available Arabic POS taggers. Our work includes preparing a POS tagging dataset, engineering multiple sets of features, and applying two machine learning methods, namely Support Vector Machine (SVM) classifier and bi-directional Long Short Term Memory (Bi-LSTM) for sequence modeling. We have improved POS tagging for Gulf dialect from 75% accuracy using a state-of-the-art MSA POS tagger to over 91% accuracy using a Bi-LSTM labeler.
Keywords: Part-of-Speech (POS), Gulf Arabic (GA), Dialectal Arabic (DA), Bidirectional Long Short Term Memory (Bi-LSTM)

1. Introduction

Part-of-speech (POS) tagging is one of the main building blocks in many Natural language processing (NLP) applications (Jurafsky and Martin, 2009). POS tagging of Arabic is challenging due to its highly inflectional nature. Arabic language has two variants, namely: Modern Standard Arabic (MSA), the formal language that used in news and official business, and Dialectal Arabic (DA), the more informal version used in everyday life. Although they share some common characteristics, they differ at many linguistic levels (Katz and Diab, 2011). Most state of the art POS taggers for Arabic are designed and trained for MSA. Though the accuracy of MSA taggers is high (around 96%), these taggers fail to achieve high scores for DA (Pasha et al., 2014). For example, as we show in this work, a state-of-the-art Farasa (Abdelali et al., 2016) MSA tagger achieves only 75% accuracy on Gulf Arabic(GA) dialect. With the wide spread of social media websites and chatting applications, DA became widely used (Diab et al., 2010). There is a need to develop NLP tools and applications for them. Hence the need for designing a specific tool for POS tagging for DA is of utmost importance. This paper aims at developing a POS tagger for one of the most widely used dialects, namely Gulf Arabic (GA).

This work answers the question "how much gain in accuracy can be achieved by designing a dedicated DA POS tagger rather than utilizing MSA specific tools to adapt to dialects?". Our results show that we can achieve higher accuracy when DA POS tagger is used. Thus our Gulf POS tagger has achieved 91.2% accuracy for POS tagging GA using Bi-LSTM, which is 16% higher than the state-of-the-art MSA POS tagger.

The contributions of this work are as follows:

- We offer an annotated data set for GA POS tagging task along with annotation guidelines used, and we make it freely accessible for the research community[1].
- We asses the state of the art MSA POS tagger on Gulf dialect and provide analyses of failures and successes.
- We offer the first POS tagger for Gulf dialect that achieves an accuracy of 91.2%.

2. Background

In this section we will cover Arabic language characteristic, POS tagging for DA and similar work in literature.

2.1. Dialectal Arabic

Arabic language has two variants: MSA and DA. MSA is the primary language of news, media and education in the Arab world (Khoja, 2001). It is mostly written than spoken (Habash, 2010; Abuata and Al-Omari, 2015). DA is the language used in daily informal communication. It was mostly spoken than written, but it gradually became the mean of communication in social media (Darwish et al., 2012).

The Arabic language has many characteristics that make it challenging especially for NLP tasks. Mainly, there are three main categories for Arabic words: nouns, verbs and particles. Each one of them can be divided into subcategories which can be represented using up to 330,000 when choosing a detailed tag set(Habash, 2010). One of the main challenges of Arabic language is having multiple meanings and POS tags for the same Arabic form, especially when diacritics are absent. Diacritics are the symbols that represent Arabic short vowels and they are optional which introduce some ambiguity since there are words which have the same consonant letters but different part of speech and different pronunciations. Table1 presents differ-

[1] http://alt.qcri.org/resources/da_resources/

POS	Diacriticization	Gloss	Example sentence
Noun	دَوْرِي [**daw**.rɪy]	league	دوري الأبطال Champions League
Noun	دَورِي [**dow**.rɪy]	My turn	إنه دوري أنا It is my turn
Noun	دَورِي [**do**.rɪy]	My role	أديت دوري I fulfilled my role
Noun	دَورِي [**do**.rɪy]	My floor	إنه دوري It is my floor
Verb	دَوِّرِي [**daw.wɪ**.rɪy]	Search around	دوري هناك search around there
Verb	دُورِي [**du:**.rɪy]	Turn around	دوري يا دنيا Oh world turn around
Adjective	دَوْرِي [**daw**.rɪy]	Periodic	فحص دوري Periodic inspection
Adverbial phrase	دَوْرِي [**daw**.rɪy]	periodically	بشكل دوري periodically

Table 1: Example of one Arabic word that can have different pronunciations, meanings and different Parts-of-speech

ent forms of the word دوري [dwry] which can be diacriticised in many ways to form different meanings with different POS tags. Words in Arabic are formed by applying different patterns to a root in order to generate a stem (Zeroual et al., 2017). Patterns can indicate the words part of speech because it carries morphological information (Darwish et al., 2014). For example the pattern فعيل [faʕi:l][2] always indicates adjectives. Affixes (prefixes and suffixes) are attached to the stem. Prefixes can indicate information such as determination of a noun or tense of a verb (Boudlal et al., 2011). Suffixes can indicate gender and number. An example of Arabic complex segmentation is the word وسنجعلكم [wasanajaʕlkum] which means 'and we will make you'. It is segmented as و+س+نجعل+كم[wa+sa+najaʕl+kum] where نجعل [najaʕl] is the stem and each one of these segments is called clitic. For more explanation see (Darwish and Magdy, 2014; Habash, 2010).

Researchers usually consider five main dialects for DA, namely: Egyptian, Iraqi, Levantine, Maghribi, and Gulf (Samih et al., 2017). Although Gulf Arabic is the largest existing dialect in social media, there is very limited attention towards building NLP tools for it.

DA is derived from MSA; nevertheless, they differ at many linguistic levels. Some notable differences are in terms of:

- Vocabulary: Arabic dialects have richer vocabulary than MSA some of which are borrowed from other languages (Habash et al., 2012a).
- Word order: in dialects it is usually Subject-Verb-Object (SVO) while it is Verb-Subject-Object (VSO) in MSA(Diab and Habash, 2007).
- DA words are written as they are pronounced since there is no orthographic standards for dialects. This fact causes inconsistency in writing some words for example the word صدق [sˤɪdq], which means 'truth' is written as صج [sˤɪdʒ] in some Gulf dialects variants. Another result of writing words as they are pronounced is that some letters are dropped when pronounced. For example the word قاعد[qa:ʕɪd], which means 'he is sitting' is written as قاع [qa:ʕ], and the word طالع [tˤa:lɪʕ], which means 'look' is written as طاع [tˤa:ʕ] in Kuwaiti Gulf dialect.
- MSA has richer morphology than dialects for example most dialects do not have dual forms and do not differentiate among plural forms in terms of gender. Dialects have some affixes that do not exist in MSA such as, the prefix ح [ħa], which indicates the meaning of سوف [sawfa], which means 'I will' and the suffixes ج [ʒ] and س [ɪs], which indicate the meaning of the second person pronouns ك [ka] in Kuwaiti Gulf and Saudi Gulf, respectively.
- MSA has strict case ending rules in their grammars while dialects have no strict rules.

In this paper, we focus our study on GA, which is one group of dialects that share many characteristics. It is the dialect of countries surrounding the Arab Gulf, such as Saudi Arabia, Kuwait, Qatar, Bahrain, Oman, United Arab Emirates and Iraq. GA has additional characteristics that distinguish it from other dialects, for example:

- Phonologically: GA maintains the pronunciation of : ذ [ð], ث [θ] and ظ [ðˤ] unlike other dialects. Moreover, the sound ق [q] has different pronunciations e.g. قال [qa:l] , جال [dʒa:l] and كال [ka:l]] which means 'he said'(Khalifa et al., 2016).
- Morphologically: In most cases, there is no case inflection on GA words. Also, the prefix ب [ba] and the verb راح [raaħ] are used to indicate future tense. In addition, the words مب [mub], موب [mob], ما [ma:], ماهو [ma:hu] and ماهوب [ma:hu:b] are used for negation (Khalifa et al., 2016).

These differences emphasize the need for specially designed NLP tools for dialects to prevent the performance drop when using MSA tools.

[2]IPA is used to present Arabic words phonetically

2.2. POS Tagging for Arabic

The literature on DA POS tagging shows different approaches for producing dialects morphological analyzer and POS taggers. One is adapting MSA tools to dialects; the second is creating dialect specific tools. There are many strategies adopted by researchers to adapt MSA morphological analyzers for dialects. The work by (Duh and Kirchhoff, 2005) uses a list of possible POS tags produced by MSA morphological analyzer which is LDC-distributed Buckwalter stemmer to decide the tag and incorporates different methods to improve results. Their objective is to have minimally supervised POS tagging. The supervised system achieved 74.88% and the minimally supervised system achieved 68.48% after improvement. Another strategy is to pre-process the data by changing its representation (Habash and Rambow, 2006).MAGEAD(Habash and Rambow, 2006) is a morphological analyzer for MSA and Levantine family. MAGEAD used finite state machine on top of AT&T transducer. They changed the representation of the dialectal word. The new representation includes some features such as a root, a meaning index and morphological behavior class (MBC) which is considered variant independent. They only reported context recall 95.5% with 60% coverage of Levantine Arabic verb forms (Pasha et al., 2014). The second approach is to target dialect directly. CALIMA (Habash et al., 2012b) is a morphological analyzer that targets Egyptian dialect. It is rule-based and it has 4632 rules to predict the correct tags. Its accuracy on POS tagging task is 84%. MADAMIRA (Pasha et al., 2014) is also a morphological analyzer with two versions. One for MSA and another for Egyptian dialect. It is an amalgamation of the two analyzers AMIRA (Diab, 2009) and MADA (Rambow et al., 2009). They used SVM to predict the correct POS tag among all possible analyses produced by the analyzer. They achieved 92.4% on Egyptian data. The Egyptian version is slower than the MSA version because of the morphological complexity of the dialect. (Alsabbagh and Girju, 2012) proposed transformation-based Egyptian dialect POS tagger trained on Twitter Egyptian corpus. Functional based annotation scheme was used for POS tagging. They achieved 87.6% F-Measure scores. They did a pre-processing step to normalize the text in order to reduce spelling variations of dialectal words and speech effects while we used the input text as it is.

To the best of our knowledge there is no POS tagger for Gulf dialect and our work represents the first attempt to train a Gulf dialect POS tagger.

2.3. Bi-LSTM POS Tagging Relevant Work

Although not applied for dialect, but there are some works that used neural network approach, strictly speaking Bi-LSTM, for POS tagging (Ling et al., 2015; Plank et al., 2016; Darwish et al., 2017). This approach proved to have high accuracy scores even when used with rich morphology languages such as Turkish and Arabic (Plank et al., 2016). (Ling et al., 2015) worked on language modeling and POS tagging tasks for English. Their Bi-LSTM taggers achieved 97.36% without any features and 97.57% using some features which they did not specify. (Plank et al., 2016) proposed Bi-LSTM POS tagger and tested it on twenty-two languages including Arabic. They experimented with different word representations and the best representation for most languages was when combining word and character representation except for Arabic in which word representation was the best representation. They achieved 98.91% accuracy on MSA. Using word embedding combined with word and character representation achieved 98.87% which is less than when embedding is not used. (Darwish et al., 2017) used the same technique of (Ling et al., 2015) and proposed two word level features which were meta-type of the word and stem template for the word. Their Bi-LSTM tagger works at clitic level in which each word is segmented into its clitics using gold segmentation. Their best performing system achieved 96.1% accuracy when using both features and no word embedding.

3. POS Tagging Methodology

Part of speech tagging can be done in a supervised manner or unsupervised. There are many approaches to POS tagging: Rule-Based Approach, Markov Model Approach, Maximum Entropy Approach, Support Vector Machine(SVM) Approach and Neural Network Approach (Wilks, 1996). In this section we present our POS tagging approach; first we describe the set of features we extracted, then we discuss the two machine learning approaches we used, which are SVM and Bi-LSTM. It is worth mentioning that our taggers operate at clitic level instead of word level where a clitic is a word segment that has single POS tag.

3.1. SVM Based POS Tagger

SVM is used in many NLP classification tasks including POS tagging and proves to achieve high accuracy results with MSA (Darwish et al., 2017; Giménez and Màrquez, 2003). For this work, we used an SVM multi-class, specifically the SVM^{multiclass} tool developed by Thorsten Joachims (Joachims, 2008). SVM^{multiclass} uses regularization parameter C to prevent overfitting (Manning et al., 2009). Each tag of POS tags was considered as a class, and a set of features mentioned at the end of the section were extracted for each clitic and used to train the SVM classifier. In this work we use a combination of features that includes probabilistic, binary, and Arabic-specific features. For probabilistic features we used a combination of bigrams, trigrams, and 4-grams of tags and clitics. For binary features we used some features including meta-types of clitics, which indicate if a clitic is a number, a foreign word, a user mention or a URL. For Arabic specific features, we used stem template feature introduced by Abdelali et al. (2016). Where stem template represents the word pattern applied to the root mentioned in section 2.1 . The template for each clitic has been extracted and concatenated to word representation.

The set of used features for SVM are:

1. **Clitic features**: each unique clitic in our training set acted as a feature, and an additional feature is added to represent out-of-vocabulary (OOV) clitics. We experimented with three different values for clitic features. The first value is binary (whether it exists or not). The second is the log of clitic counts in training data. The

third is the Term Frequency-Inverse Document Frequency (TF-IDF) score for each clitic in which tweets are considered as documents.

2. **Probabilistic features**:
   - The probability of the co-occurrence of the tag and clitic $P(tag_i/clitic_i)$ and $P(clitic_i/tag_i)$
   - The probability of the previous tags bigram, trigram and four-gram $P(tag_i|tag_{i-1}), P(tag_i|tag_{i-1}, tag_{i-2})$ and $P(tag_i|tag_{i-1}, tag_{i-2}, tag_{i-3})$
   - The probability of the next tag bigram, trigram and 4-gram: $P(tag_i|tag_{i+1})$, $P(tag_i|tag_{i+1}, tag_{i+2})$ and $P(tag_i|tag_{i+1}, tag_{i+2}, tag_{i+3})$
   - The probability of the tag given the previous four and the next four clitics: $P(tag_i|clitic_{i-1}, clitic_{i-2}, clitic_{i-3}, clitic_{i-4})$
3. **Binary features**:
   - Meta-type of the clitic, whether it is a number, a foreign word, a mention, a hash tag, or URL.
   - Clitic position (initial, middle, end) of the word.
   - If a clitic is a prefix, a suffix or a stem.
   - Leading letters آأاإبت [different forms of Alif, ta], which can indicate that a clitic is a verb.
   - If the previous tag is a progressive particle or a determiner. So this will indicate if a clitic is a verb or a noun, respectively.

The values for probabilistic features are calculated using the Maximum Likelihood Estimation (MLE), while a non-zero value of 10^{-10} is assigned for unseen n-grams.

3.2. Bi-LSTM Based POS Tagger

Bi-LSTM is a special type of Recurrent Neural Network (RNN). It has proved to be a good choice for sequence modeling tasks (Ling et al., 2015) such as speech processing, POS tagging, phrased based chunking ... etc. It is also less sensitive to training data size (Plank et al., 2016). Moreover, Bi-LSTM can capture the context around source words up to very long sequences in both directions (previous and upfront) (Wang et al., 2015). It also does not need hand crafted features to work well. These characteristics make it a suitable fit for POS tagging of DA. Since there is not much training data available for DA – GA in this case – and since DA lacks standards to design powerful features, a model is needed that auto-fits its features and characteristics. Bi-LSTM structure differs from the classic RNN in that it adds a memory cell to the neural network architecture that learns to memorize information about a sequence for long periods of time. It also takes two passes of the input sequence, both forward and backward. For the sequence of clitics c_1, c_2.. c_m, where m is the number of clitics in a sequence, it manages clitic c_i by encoding information about c_1 ... c_{i-1} and c_m ... c_{i+1} sub-sequences. Bi-LSTM takes a sequence of features i.e. word representations, word embedding and any hand-crafted features. The feed-forward states of the network outputs the tag sequence for the previous clitics. The back-forward state holds tags information for the next clitics. The two states are combined using the following function:

$$l_i = tanh(L^f S_i^f + L^b S_i^b + b_l)$$

where L^f, L^b and b_l are parameters for combining the forward and backward states, S_i^f and S_i^b are the forward states and backward states respectively (Darwish et al., 2017). For implementation, we used Java Neural Network (JNN) toolkit for language modeling and part-of-speech tagging proposed by (Ling et al., 2015). In order to reach good generalization for any language processing task we need to have good word representations (Ling et al., 2015). In JNN, there are two representations: word representation and compositional character representation i.e. character-to-word (C2W). Word representation combined with C2W representation is called (CC2W+W). The input of our network is a sequence of features: clitic representations (word , C2W and CC2W+W), meta type, and/or stem template. The output will be a sequence of tag predictions in which each tag is formed by combining the forward and backward state of the network. JNN uses a tanh activation function. We include the stem template feature introduced by (Abdelali et al., 2016). The template for each clitic has been extracted and concatenated to the representation. Some clitics have no valid patterns e.g. ال [al] determiner. We also include meta-type feature which is an additional information added about the type of clitic i.e. to specify whether it is a number, an adjective number, a prefix, a suffix, a foreign, a punctuation, an Arabic letter and twitter specific types: hashtags, URLs and mentions.

4. Experimental setup

4.1. Data

We used gold annotated dataset which is built using gold segmented GA tweets taken from Samih et al. (2017). It consists of 343 Tweets with 6,844 tokens and 10,255 clitics. we used simplified Arabic Tree Bank (ATB) 18 tag sets proposed by (Darwish et al., 2017), but we neglected the abbreviation tag, since it is unlikely to be used in DA. Moreover, we added four new tags for twitter specific data which are MENTION, URL, HASH, and EMOT for twitter mentions, hyperlinks, hash tags and emotion punctuations respectively. The total number of POS tags is 21 tags.
We did manual annotation for the data according to the following guidelines:

- Each words clitic should be labeled with one tag.
- The number of tags of a word is equal to the number of segments for that word.
- If a stem can be classified as an adverb or an adjective, we consider it an adjective.
- We only label a stem as an adverb if it always appears as an adverb.
- Any loan or foreign word written in Arabic letters (transliterated) was labeled with its original tag in the foreign language. For example, واټ از زیس "what is this" was labeled as PART, V, PRON, respectively.

The data used for experiments are 233 tweets for the training set, 33 tweets for the development set and 77 tweets for the testing set. We have another dataset which we used to test the effect of having more data that enable the systems

to learn features accurately. Since the size of the GA training data was limited and due to the high overlap between GA and MSA, we opted to augment our data with MSA data. Specifically, we used a set of 20,000 tweets, which were gold-segmented into about 890,000 segments and we tagged them using Farasa. In order to combine them with Gulf data without losing dialectal characteristics, the Gulf data was replicated to be the same size as the MSA data. We refer to this dataset as "Gulf++".

4.2. Baselines

We apply two baselines to compare the performance of our POS tagger. The first is the simple majority class baseline, where all words are labeled with the most common POS tag, namely "NOUN". The second baseline is obtained by applying the Farasa POS tagger (Darwish et al., 2017) to our GA data. Since Farasa does not cover Twitter specific tags, we assumed the tagging results for these tags to be predicted correctly.

4.3. Evaluation Metric

In order to evaluate the performance of any POS tagger, there are several evaluation measures available. In this paper, we used accuracy to measure effectiveness. Accuracy is the most widely used metrics for POS tagging (Craig Hagerman, 2012). The accuracy of the POS tagger is the ratio of correctly tagged words of a test set of all words where a correct tag is the tag that matches the true tag annotated by humans (Jurafsky and Martin, 2009). Errors analysis will be conducted using confusion matrix which indicates how many times a tag is confused with other tags (Manning et al., 2009)

5. Evaluation and Discussion

Our majority-class baseline, which assigned the "NOUN" tag to all words achieved an accuracy of 21.16%, which indicates that the POS tagging for GA is not a trivial task. The second baseline that uses the state-of-the-art Farasa POS tagger achieved 75.13% accuracy. This suggests that GA is somewhat close to MSA; a similar conclusion reached in (Samih et al., 2017). However, it is still far behind the performance on MSA, which is over 96%. This motivates the design of a dialect-specific POS tagger.

5.1. SVM Approach

Table 2 summarizes the results of experiments with SVM. Three clitic features values were tested. The best performance on the Gulf test set was achieved using the binary feature values. The accuracy was 85.8%. The best performance of the Gulf++ test set was 86.0% using TF-IDF scores. Adding meta-type information clearly increased the performance of the SVM. This was expected because these types have strong indications of the word's tag. Extending the dataset to include the MSA training example enhanced the performance of all settings. The fact that the performance benefited from having more training data to certain limits (Joachims, 2002) corroborates this observation .

The best performing system among all experiments was the system with the following setting: using the Gulf++ training set, with TF-IDF for the clitics features values and using meta-type features. The system achieved 86.0% accuracy. Further analysis of the types of errors produced by the system was carried out using a confusion matrix. The most common error was confusing verbs with nouns by 28.03%. This might have been due to the absence of short vowels and diacritics since some clitics have the same consonant letters but different pronunciations. It might also have occurred as a result of preferring the noun tag for out-of-vocabulary words because it is more common. The next most common error was confusing pronouns with noun suffixes and vice versa, which formed 24.7% of system errors. This error is common because the list of suffixes and the list of attached pronouns are similar. The third person pronoun, ـه [ha] and ـة [ta] the marker of feminine nouns is an example of this. Speakers tend to write them interchangeably in writing dialect since no strict orthographic rules are available. The third common error was confusing adjectives with nouns by 12.5%. Table 3 lists system errors rate.

Dataset	Features	Accuracy
Gulf	binary	81.85
Gulf	Binary + meta-type	85.8
Gulf++	Binary + meta-type	85.8
Gulf	log	81.85
Gulf	log + meta-type	85.2
Gulf++	log + meta-type	85.7
Gulf	TF-IDF	78.1
Gulf	TF-IDF + meta-type	80.4
Gulf++	TF-IDF + meta-type	86.0

Table 2: SVM experiments results

Error Type	Percentage
V –> NOUN	28.03%
ADJ –>NOUN	12.5%
PRON –> NSUFF	13.3%
NSUFF –> PRON	11.4%
PART –> ADJ	8.9%
PART –>NOUN	3.03%
HASH –> NOUN	3.03%
NOUN –>V	2.7%
PROG_PART –>PREP	2.3%

Table 3: Most common errors for the best SVM system

5.2. Bi-LSTM Approach

Different representations were put into an experiment, namely: compositional character representation (C2W), word representation and a combination of both CC2W+W. In general, CC2W+W proved to have the highest accuracy among all representations. Word representation is next in accuracy, followed by C2W. In essence, C2W representation achieved its best performance when no feature was used and using word level features caused its accuracy to drop. It seems that characters had a stronger relation than word level relations. Conversely, word representation achieved higher accuracy when both word level features, meta-type and template, were used. Finally, CC2W+W representation benefited from all word level features. It seems that CC2W+W was best when the training set was large.

Features	Gulf			Gulf++		
	C2W	Word	CC2W+W	C2W	Word	CC2W+W
None	86.7	85.9	88.5	88.5	86.8	89.6
Meta-Type	86.3	87.5	88.9	88.6	88.3	90.5
Template	85.6	88.7	88.6	88.6	89.0	90.7
Meta-Type+Template	85.4	89.7	89.1	88.7	90.6	91.2

Table 4: Bi-LSTM experiments results

System	Meta-Type	Template	Accuracy
SVM (Gulf+Binary)	Yes	No	85.8
SVM (Gulf++, TF-IDF)	Yes	No	86.0
Bi-LSTM (word representation, Gulf)	Yes	Yes	89.7
Bi-LSTM (word representation, Gulf++)	Yes	Yes	90.6
Bi-LSTM (CC2W+W (Gulf++))	Yes	Yes	**91.2**

Table 5: Best performing systems in SVM and Bi-LSTM experiments

Error Type	Percentage
V –>NOUN	16.6%
ADJ –>NOUN	16.0%
PRON –>NSUFF	12.4%
NSUFF –>PRON	9.4%
NOUN –>V	7.7%
NOUN –>ADJ	5.9%
PART –>NOUN	4.7%
NOUN –>PART	2.3%
PREP –>PART	2.3%

Table 6: Most common errors for the best Bi-LSTM system

When the training set was small the word representation fared better (see Table 4).

Experimenting with features shows that the highest accuracy values were achieved when the meta-type features and the template feature were combined. When meta-type features were added, CC2W+W representation achieved the highest accuracy scores, followed by word representation (see Table 4). This was true because the meta-type was a set of features that was meaningful to the clitic itself, not to its characters sequence. The template feature was also more meaningful for the clitic level than for the character level. The template feature helped to overcome one of the most common errors made by Arabic POS taggers which is confusing adjectives with nouns. Hence there is an improvement of adjective tag accuracy from 67.4% to 71.6% for CC2W+W representation on Gulf data and for noun tag accuracy from 82.2% to 82.9%. The improvement was due to the fact that adjectives and nouns have consistent templatic forms.

Adding more data to the training set improved the results in all settings with and without features, different representations. For example the system benefited from enriching the training data on meta-types features because it had the chance to observe more meta-types features and learn their effects on tag prediction. This observation was supported in part by the fact that per-tag accuracies for our best Bi-LSTM system were 100% for MENTION, HASH, NUM and URL tags, in which all had corresponding meta-type features.

The highest system was achieved with the following settings: CC2W+W representation, meta-type and template features and on Gulf++ dataset. The system accuracy was 91.2%, out-of-vocabulary accuracy was 73.5%. And the scores for precision, recall and F-score are 83.9%, 91.2% and 87.4% respectively. The errors trends were the same as the SVM errors. The most common error was confusing verbs with nouns with 16.6%. The second was confusing adjectives with nouns with 16.0%, followed by confusing pronouns with nouns suffixes, 12.4%. Table 6 gives a summary of the systems confusion matrix.

5.3. Discussion

To summarize the previous analyses, SVM is fast and achieves good result with a basic set of features, while Bi-LSTM is slower but can achieve higher accuracy levels without using any features. Still the effect of a good combination of word representation and features combination can enhance the results to a great extent. This fact is corroborated by the findings of Darwish et al. (2017), Plank et al. (2016), and Ling et al. (2015) in which all Bi-LSTM taggers benefited from features. Unlike the results of (Darwish et al., 2017) in which the SVM tagger outperformed the Bi-LSTM tagger by 0.1%. Our results show that Bi-LSTM outperforms SVM. This may be due to the fact that the Gulf dialect has no well-known grammatical standards or orthographic rules. Knowing the standards of MSA helps with feature engineering for those standards and characteristics, which helps in improving the accuracy of SVM. In the same vein, Bi-LSTM can capture non-lexical relations and dialectal trends and model them well without the need for highly dialectal features. Generally, both systems outperform Farasa, which supports our hypothesis that there should be specially designed tools to manage DA.

Table 5 summarises the results of the best systems among all experiments. Although the clitic level features were shared by MSA and Gulf, they improved the performance. Adding more detailed features can enhance the results.

6. Conclusion

In this work, we test the performance of a state-of-the-art MSA POS tagger on Gulf Arabic. The tagger achieved an

accuracy of 75% only. This motivated the design of a GA POS taggers using two approaches of POS tagging.

First, we designed an SVM tagger. A set of features was put into the test; their effect on accuracy was reported. Second, we examined a Bi-LSTM POS tagger. In both approaches we tested for the effect on accuracy of adding more data to the training set. A dataset for GA POS tagging task was prepared for use and made accessible to the research community. The best performance of Bi-LSTM is 91.2% using CC2W+W representation and meta-types and template features. On the other hand, the best performance of SVM is 85.96% by setting the clitic feature value to TFIDF and using meta-types features. Both systems achieved their highest accuracy when trained on the Gulf++ dataset.

However, Bi-LSTM outperforms SVM in most of its settings. The accuracy of Farasa on our Gulf dataset (75%) indicates that Gulf Arabic is close to MSA to some extent. The accuracy boost we could achieve supported our assumption that we need specifically designed and trained tools for DA.

Our future work includes investigating and adding more dialect data to the training set rather than MSA data; consulting dialect linguistic resources to engineer more informative features for SVM and Bi-LSTM.

7. Acknowledgements

We would like to thank Prof. Awwad Alahmadi for his insightful discussion and examples regarding Arabic linguistics and for taking care of the IPA transcriptions.

8. References

Abdelali, A., Darwish, K., Durrani, N., and Mubarak, H. (2016). Farasa: A Fast and Furious Segmenter for Arabic. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 2016:11–16.

Abuata, B. and Al-Omari, A. (2015). A rule-based stemmer for Arabic Gulf dialect. *Journal of King Saud University - Computer and Information Sciences*, 27(2):104–112.

Al-sabbagh, R. and Girju, R. (2012). A Supervised POS Tagger for Written Arabic Social Networking Corpora. *Proceedings of KONVENS 2012 (Main track: oral presentations), Vienna, September 19*, 2012:39–52.

Boudlal, A., Belahbib, R., Lakhouaja, A., Mazroui, A., Meziane, A., and Bebah, M. (2011). A markovian approach for Arabic root extraction. *International Arab Journal of Information Technology*, 8(1):91–98.

Craig Hagerman. (2012). Evaluating the Performance of Automated Part-of-Speech Taggers on an L2 Corpus. pages 29–42.

Darwish, K. and Magdy, W. (2014). Arabic Information Retrieval. *Foundations and Trends® in Information Retrieval*, 7(4):239–342.

Darwish, K., Magdy, W., and Mourad, A. (2012). Language processing for arabic microblog retrieval. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2427–2430. ACM.

Darwish, K., Abdelali, A., and Mubarak, H. (2014). Using Stem-Templates to improve Arabic POS and Gender/Number Tagging. *International Conference on Language Resources and Evaluation (LREC-2014)*, pages 2926–2931.

Darwish, K., Mubarak, H., and Abdelali, A. (2017). Arabic POS Tagging : Don ' t Abandon Feature Engineering Just Yet. *WANLP 2017 (co-located with EACL 2017)*, pages 130–137.

Diab, M. and Habash, N. (2007). Arabic dialect processing tutorial. *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Tutorial Abstracts on XX - NAACL '07*, (April):5–6.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). COLABA: Arabic dialect annotation and processing. *LREC Workshop on Semitic Language Processing*, (January 2016):66–74.

Diab, M. (2009). Second Generation AMIRA Tools for Arabic Processing : Fast and Robust Tokenization , POS tagging , and Base Phrase Chunking. *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, pages 285–288.

Duh, K. and Kirchhoff, K. (2005). POS Tagging of Dialectal Arabic : A Minimally Supervised Approach. *Computational Linguistics*, (June):55–62.

Giménez, J. and Màrquez, L. (2003). Fast and accurate part-of-speech tagging: The SVM approach revisited. *Ranlp*, pages 153–163.

Habash, N. and Rambow, O. (2006). MAGEAD: a morphological analyzer and generator for the Arabic dialects. *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (Coling-ACL)*, M:681–688.

Habash, N., Diab, M., and Rambow, O. (2012a). Conventional Orthography for Dialectal Arabic. *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, (January 2012):711–718.

Habash, N., Eskander, R., and Hawwari, A. (2012b). A Morphological Analyzer for Egyptian Arabic. *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology SIGMORPHON2012*, pages 1–9.

Habash, N. Y. (2010). *Introduction to Arabic Natural Language Processing*, volume 3. Morgan & Claypool Puplisher.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*, volume 29. SPRINGER SCIENCE+BUSINESS MEDIA. LLC.

Joachims, T. (2008). SVM-Multiclass: Multi-Class Support Vector Machine.

Jurafsky, D. and Martin, J. H. (2009). Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. *Speech and Language Processing An Introduction to Natural Language Processing Computational Linguistics and Speech Recognition*, 21:0–934.

Katz, G. and Diab, M. (2011). Introduction to the Special Issue on Arabic Computational Linguistics. *ACM Transactions on Asian Language Information Processing*, 10(1):1–4.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016). A Large Scale Corpus of Gulf Arabic. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4282–4289.

Khoja, S. (2001). APT : Arabic Part-Of-speech Tagger. *Proceedings of the Student Workshop at NAACL*, pages 20—-25.

Ling, W., Luis, T., Marujo, L. L., Astudillo, R. F., Amir, S., Dyer, C., Black, A. W., Trancoso, I., Fermandez, R., Amir, S., Marujo, L. L., and Luís, T. (2015). Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, (September):1520–1530.

Manning, C. D., Ragahvan, P., and Schutze, H. (2009). An Introduction to Information Retrieval. *Information Retrieval*, (c):1–18.

Pasha, A., Al-badrashiny, M., Diab, M., Kholy, A. E., Eskander, R., Habash, N., Pooleery, M., Rambow, O., and Roth, R. M. (2014). MADAMIRA : A Fast , Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. *Proceedings of the 9th Language Resources and Evaluation Conference (LREC'14)*, pages 1094–1101.

Plank, B., Søgaard, A., and Goldberg, Y. (2016). Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL-16)*, pages 412–418.

Rambow, O., Habash, N., Rambow, O., and Roth, R. (2009). MADA + TOKAN : A toolkit for Arabic tokenization , diacritization , morphological disambiguation , POS Tagging, Stemming and Lemmatization. (January 2017).

Samih, Y., Eldesouki, M., Attia, M., Darwish, K., Abdelali, A., Mubarak, H., and Kallmeyer, L. (2017). Learning from Relatives : Unified Dialectal Arabic Segmentation. (CoNLL):1–10.

Wang, P., Qian, Y., Soong, F. K., He, L., and Zhao, H. (2015). A Unified Tagging Solution: Bidirectional LSTM Recurrent Neural Network with Word Embedding. *arxiv: EMNLP-reject*.

Wilks, Y. (1996). Natural language processing. *Commun. ACM*, 39(1):60–62.

Zeroual, I., Lakhouaja, A., and Belahbib, R. (2017). Towards a standard Part of Speech tagset for the Arabic language. *Journal of King Saud University - Computer and Information Sciences*, 29(2):171–178.

Web-based Annotation Tool for Inflectional Language Resources

Abdulrahman Alosaimy, Eric Atwell
School of Computing, University of Leeds
Leeds LS2 9JT, UK
{scama, e.s.atwell}@leeds.ac.uk

Abstract

We present Wasim, a web-based tool for semi-automatic morphosyntactic annotation of inflectional languages resources. The tool features high flexibility in segmenting tokens, editing, diacritizing, and labelling tokens and segments. Text annotation of highly inflectional languages (including Arabic) requires key functionality which we could not see in a survey of existing tools. Wasim integrates with morphological analysers to speed up the annotation process by selecting one from their proposed analyses. It integrates as well with external POS taggers for kick-start annotation and adaptive predicting based on annotations made so far. It aims to speed up the annotation by completely relying on a keyboard interface, with no mouse interaction required. Wasim has been tested on four case studies and these features proved to be useful. The source-code is released under the MIT license.

Keywords: annotation, inflectional language, morphosyntactic, Arabic

1. Introduction

POS tagging text in inflectional languages is usually hard. A typical problem is substantial lexical data sparseness due to the high number of possible inflexions of a single word. To reduce sparseness and number of Out-of-Vocabulary (OOV) words, inflected words are often segmented prior to or in parallel with POS tagging. However, the segmentation process is prone to errors. Inflection boundaries are often not marked which increases the number of homographs (two or more words spelt in the same form but with different POS tag or pronunciation (e.g. due to differences in diacritization). Some orthographical changes are caused by inflexions, making it hard to recover the original word form. As a result, a segmentation process sometimes fails to detect morphemes.

Wasim is a web-based tool for semi-automatic annotation of text for the purpose of gold standard corpus production[1]. It was developed for the annotation of our Sunnah Arabic Corpus (SAC) (Alosaimy & Atwell, 2017), a collection of classical Arabic sayings ascribed to the prophet Mohammad. It has also been tested in four case studies.

For the project, we investigated the required set of features needed for annotating SAC and used these as criteria in a survey of existing tools. In our search for currently available tools, we limited our survey to tools that **1.** are web-based: to integrate it with other systems, and to allow easier access through browsers. **2.** Annotate text tokens with morpho-syntactic tags in CoNLL-U v.2 format (Nivre & Agic, 2017)[2]. **3.** Support right-to-left languages. **4.** are available to download for research purposes.

Morphosyntactic annotation of SAC (and other highly inflectional language corpora) requires additional specialized functionality:

1. Segmentation of one word into a set of segments
2. Addition of orthographical accents or diacritics
3. Listing a set of solutions from a lexicon dictionary (internally or externally using a morphological analyser)
4. Consistency validation and integrating annotation guidelines (e.g. homographs).
5. Adaptive prediction based on historical tagging
6. Efficient keyboard-based navigation and labelling

In the next section, we provide an overview of major related tools for annotating corpora, with a tabular comparison of support for these features with Wasim.

2. Related Work

We limit our literature review to tools that meet our four conditions, which results in five tools. Brat (Stenetorp et al., 2012) is a widely-used visualization and annotation tool that is mainly for syntactic annotation in addition to morpho-syntactic annotation. WebAnno (Yimam, Gurevych, de Castilho, & Biemann, 2013) is a Java-based set of well-documented tools for multiple annotation tasks. Arborator (Gerdes, 2013) is a dependency annotation tool, that supports RTL languages natively. Sequence Annotation Web Tool (Samih, Maier, & Kallmeyer, 2016) is a basic web-based tool for the annotation of token sequences with an arbitrary set of labels (e.g. POS tags). The authors claimed to publish the code on GitHub, but we could not find a link to it, so we exclude it from the table comparison. CorA (Marcel Bollmann, Florian Petran, Stefanie Dipper, 2014) is a web-based tool for morpho-syntactic annotation of non-standard texts.

In Table 1, we compare the support of the six features. Although these tools did not meet all of our requirements, we must say that some of them support other features (e.g. syntactic annotation) that are not needed in our project, and therefore are not listed in the table. We run these tools for testing, and the support to these features is to the best of the author knowledge. Some tools support multi-token span annotation, but this assumes tokens are segmented, so we consider segmentation feature as not supported.

[1] Source code and demo is available at: http://wasim.al-osaimy.com

[2] CoNLL-U format has been used in Universal Dependencies project and well described in http://universaldependencies.org/

Features	Brat	WebA	Arb	CorA	Wasim
Segment one word into segments.	✓				✓
Support Diacritics		✓	✓	✓	✓
Suggest a set of solutions from a lexicon / dictionary					✓
Consistency validation		✓			✓
Adaptive predicting based on historical tagging		✓		✓	✓
Efficient Keyboard-based navigation and annotation		✓			✓

Table 1: Tools and their support for a range of features.

3. Major features

The annotation of text in a highly inflectional language is usually harder because:

1. Words are highly ambiguous, which results in many homographs (i.e. more need of a lexicon),
2. Words need to be segmented into a set of morphemes, and
3. As a result, tagger performance is usually poorer and relies on a lexicon or a morphological analyser to improve the accuracy.

Semi-automatic annotation should help to remove the ambiguity of words as it should be able to correct tagger errors. Often, these errors are in the ranking of the solution set provided by the morphological analyser. Therefore, the most essential feature is the integration of a morphological analyser, which allows the annotator to re-select the proper analysis in case of incorrect automatic tagging.

In addition, an efficient way to segment words into a set of morphemes is a necessity. For example in Arabic, many words are inflected and an inflected word (multi-word token) consists on average of 2.06 syntactic words (or morphemes)[3].

3.1 Morphological Analyser Integration

Wasim integrates with morphological analysers to speed up the process of annotation. Morphological analysers take a word as input and produce a list of possible analyses (which include word's segmentation and lemma and segment's POS tag and features). By providing a set of possible analyses, Wasim allows annotators to select one analysis. Once a solution is chosen, all its values of POS tag, lemma, segmentation, and morphological features will be reflected in the word analysis. The chosen solution can be edited though.

In our SAC project, a word may be tagged with up to ten features, in addition to segmenting the word into a set of morphemes and marking its POS tag. We hypothesise that it will more efficient to select a solution instead of doing them all from scratch. However, this hypothesis depends on the quality of the morphological analyser. Annotators have to mark all features if the analyser returns no results.

[3]http://universaldependencies.org/treebanks/ar-comparison.html

Once a newly-created analysis is detected, it will be saved in the server for possible later re-use.

Wasim provides two ways of morphological analyser integration. First, using an embedded supplementary tool that acts as a simple lexicon memory: It reads the annotated part of the corpus and index words with their annotations. Then, it allows HTTP requests to be made from Wasim, and it will return all possible solutions of the token in hand.

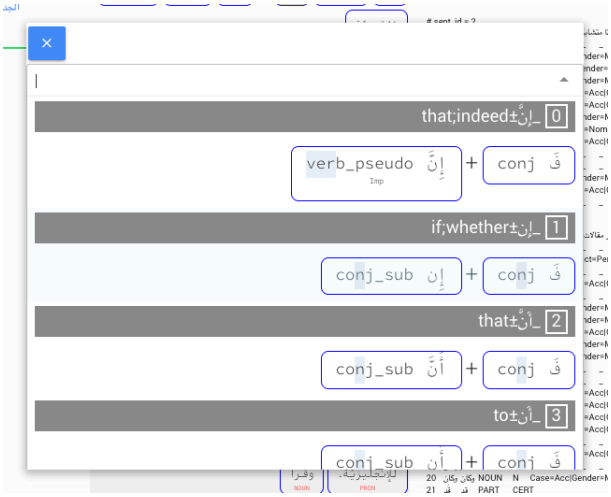


Fig 1 The list of possible solutions from a morphological analyser. A solution is usually a bundle of POS tag, segmentation, lemma and morphological features. Selecting one solution will replace all its content to each proper annotation field.

Second is using an external morphological analyser (MA). MA outputs must be in CoNLL-U format with word id in the MISC column that maps to the original word index of the submitted sentence (e.g. WID=2). The reason is to allow Wasim to group MA's analyses of one word together.

A mapping between MA's tagset and the project tagset may be required, and this can be defined in the configuration. If the mapping results in an ambiguous tag in the project's tagset, Wasim will duplicate the analysis for each possible tag. For example, if NOUN is mapped to PN and N, two analyses will be presented to the annotator.

3.2 Consistency Reinforcement

Consistency (a.k.a. "stability" when measuring the consistency of one annotator alone over time) of the corpus annotation process is important to ensure that all annotators in all texts follow the same procedure of annotation over time. High consistency means little disagreement in the annotation, and this helps training machine learning algorithms successfully.

To increase the consistency of the segmentation and tagging of a corpus, Wasim followed three approaches. First, it allows the use of an automatic POS tagger. Second, it integrates with morphological analysers. Third, it generates periodically a list of common homographs. Homographs are associated with their possible POS tags and segmentation. Possible segmentations are only shown when the token in hand is a homograph.



Figure 2: A screenshot that shows Wasim in a browser. The middle part represents one sentence where each box is a token (with its XPOS tag). Tokens of inflected word are linked by + symbol. The left side shows feature annotation. The top bar represents actions such "save file" and "undo last action". On the right side, CoNLL-U synchronized representation of the sentences.

Usually, in annotation guidelines, there is some guidance for specific words, usually homographs. However, in highly inflectional languages, those homographs are overwhelming, and such offline guidelines may miss some homographs, and/or the guideline document may be lengthy. This feature serves as an online guideline for annotators, which is automatically built up.

In the segmentation layer, Wasim warns the annotator when a segmentation of a word differs from previous segmentation of the same word. If the annotator insists, the new segmentation will be added. A similar process is applied for morphological tagging.

The list is regenerated periodically from the annotated part of the corpus, and the possible segmentations/POS tags of homographs are kept. Each homograph will have a set of examples in context for each sense. Moderators can edit the list, and/or add guideline notes for tagging of special cases. The list will appear in Wasim with relevant notes when selecting a word in the list.

3.3 POS tagging Integration

Instead of starting the annotation process of a corpus from scratch, Wasim integrates with UDPipe (Straka & Straková, 2017) to kick start the annotation process. UDPipe provides trained models for more than 60 languages that tokenize, tag, lemmatize and dependency parse raw text and save results in CoNLL-U formatted files. Wasim uses UDPipe as well to improve its prediction model by periodically adding instances of the corpus that has been annotated so far.

Other tools can be used as long as they generate CoNLL-U formatted files. For example, SAWAREF toolkit can be used for Arabic and the translation from popular POS tagger into CoNLL-U format can be done using one of its tools.

4. Data Representation

Wasim follows the Universal Dependencies v 2.0 (UD) (Nivre & Agic, 2017) in the way it represents sentence segmentation, POS tagging, morphological features, segmentation, and lemmatization. All annotation is stored as CoNLL-U files, which can be downloaded anytime. Since Wasim does not annotate syntactic relationships, related columns are marked as missing.

Unlike some other representations, CoNLL-U is morpheme-based tagging with the ability to recover the original word form prior to segmentation. In addition, each morpheme has two POS tags; one from coarse universal tagset (UPOS) and one from the author's defined fine-grained tagset (XPOS). This enables sharing and comparing of cross-linguistically consistent grammatical annotation of more than 100 treebanks available in UD project. CoNLL-U format serves two purposes: a well-formed structure for saving annotations (like XML) and as a high-level guideline for morphological tagging. Annotators are encouraged to use UPOS tags, and a simple mapping from XPOS to UPOS can be provided in the configuration.

The UD project does not have a standard format for diacritization as it is language-specific. We followed our project's representation of diacritization of Arabic[4]. Wasim allows users to enforce such representation by performing a series of transformations using "regex" expressions. Moderators can enforce a similar approach for other languages. Diacritization changes a word from its original form; Wasim, however, keeps the original sentence form before diacritization in the comments part of the sentence.

5. Tool Description

The Wasem tool has two main components: a *front-end* interface which allows interaction with annotator and provides warnings and feedbacks, and a *back-end* server that manages sessions and storage of CoNLL-U files.

The front-end web-based tool is built using Ionic framework using Typescript/Javascript programming language. The main screen contains four sections: a toolbar at the top is used for warnings, helpful shortcuts, and for a glance of shortcuts. The rest is separated into three columns. The middle column shows the words in small boxes (each with its XPOS tag beneath it) with the current word in process highlighted in a different colour. Multi-word tokens show morphemes linked by a "+"

[4] http://sac.al-osaimy.com/guidelines

symbol. Instead of displaying words in a tabular format (like in CorA, SAWT), we display words in natural paragraph flow; allowing annotator to easily examine a word's context. The left column shows key-value pairs of the lemma and morphological features, and the right column shows the synchronized CoNLL-U format of the current document. Closed features are a dropdown list with an auto-complete feature. Figure 2 shows a screenshot that shows the main components of Wasim.

CoNLL-U representation on the right side is editable at any time, as Wasim synchronizes changes. Changes will be validated and errors are reported in an error log box below it. In case of valid changes, such changes are reflected on the Wasim widgets. This should give an option to the annotator to make changes in bulk like copying previous annotations, though this should be used carefully in Gold Standard manual annotation.

In addition, three useful subviews are displayed on demand: **A.** a list of other alternative solutions retrieved from a morphological analyser. **B.** a tabular format of morphological features and possible values. **C.** a segmentation view that allows segmenting words easily. The front-end of Wasim can be seen as a CoNLL-U file editor: it parses the file, validates the syntax and visualizes the sentences with a synchronized side-by-side view of the CoNLL-U file.

The back-end is a server operated using Node.js Express server, and is responsible for authentication and management of annotated and raw files. A connection with the server using WebSocket is established for several reasons: such as morphological analyser requests, logging sessions, diacritization requests, and temporary session backup.

Each project is a folder in the system that contains document files, configuration files, a database of homographs and a file of the corpus lexicon. It manages the versioning of files using the standard Git version control system. The Git system tracks all the changes that are made to files, and allows multiple operations, e.g. *diff* to show changes to a file in the colour-coded interface. Annotated documents are moved to a subfolder.

All annotations are stored in CoNLL-U format as plain text files. Accessing one file from an annotator will grab a copy of that file; however, this might allow other annotators to work on the same file. To prevent that, Wasim implements a simple lock system where a file is locked while a connection is maintained with the server (using WebSocket). We only release the lock if the annotator accesses another file or the connection is closed.

Wasim is designed to be configurable to support preferences and project related setup. Project setup includes its name, language, remote Git repository, UDPipe model, morphological analyser path and several other preferences. Projects must define their own fine-grained tagset (unless UD tagset is used), with their morphological features. Wasim allows custom key-binding for actions. The configuration files are saved in the project level as JSON files.

The annotation process can be fully manual or semi-manual. In case of semi-manual, the corpus is first tagged using UDPipe models. Automatically generated tags can then be checked and manually edited using Wasim. In the next section, we will describe the supported morphosyntactic layer in more detail.

6. Morphosyntactic tasks

Wasim provides an easy interface for the annotation of up to six tasks. While these tasks can be processed sequentially, we allow annotators to work on any of the tasks at the same time. Tasks sometimes are interrelated, e.g. if the automatic tagger produced the wrong POS tag, it might as well have produced the wrong morphological segmentation/lemma ..etc. Since Wasim uses morphological analysers, if the annotator chose one solution, it will affect multiple tasks at the same time. Therefore, we allow the annotator to edit previous tasks without leaving the screen. However, we expect the annotator to use the morphological analyser (MA) feature at the beginning of a word's segmentation, diacritize then segment the word, mark POS tag, and finally mark morphological features.

Since Wasim allows the user to annotate text on many levels at the same time, an annotator might skip a task accidentally. Wasim provides a guide to go through tasks in keyboard mode. It highlights tasks sequentially to keep the annotator's focus on the current task.

However, depending on the corpus annotation goals and preferences, an annotator can customize the view; e.g. deactivate one/multiple tasks, or disable CoNLL-U view. The annotator can write post-process rules to check the validity and consistency of different tasks as well as constraints on different tasks.

Wasim is designed to increase productivity for these particular annotation tasks while sacrificing some degree of simplicity, eg there are many shortcuts/buttons on the screen. While the learning curve (the rate of a person's progress in gaining experience) is steep, we hypothesized that once the annotator is trained, Wasim features will reduce the time required for annotating each word.

6.1 Morphological segmentation

Inflectional languages tend to inflect morphemes to express different grammatical features. Unlike many other annotation tools, we do not assume the text to be tokenized/segmented. Annotators can easily tokenize words by editing their forms. Word can be segmented as well by placing a pointer in the proper position and inserting a special character ("+" sign by default). The two generated morphemes will clone the data from the original word except for its form which will be divided. The multi-token form will remain the same though. The original word in the main screen will be replaced by two morphemes linked by "+" symbol. The annotator can remove segmentation by simply hitting the "backspace" button in one morpheme, and it will merge to the previous morpheme.

With the integration of morphological analysers, annotators should mostly select the proper segmentation/tagging from its provided list. Manually segmenting one word should be resorted to as a last choice, the case when there is no proper segmentation.

Since we follow CoNLL-U representation, UD representation keeps the form of both the word and the token in its two-level indexing scheme. The form of one token can be rewritten as if it was not inflected. Free morpheme form can be altered because of the inflexion, and an annotator can recover its original form, e.g. "John's" can be recovered to either "john+has" or "john+is". The original form (John+'s) will be written in the MISC (last) column. The result CoNLL-U will be like the following:

```
1-2  John's  _   _     _  _  _  _  _  _
1    John    _   NOUN  N  _  0  _  _  _
2    has     has AUX   BE _  0  _  _  ORG='s
```

6.2 Diacritization

A diacritic (sometimes called accent or short vowel) is an optional small glyph added to letters to change the sound of the letter. Diacritization is the process of adding those glyphs. In our Sunnah project, we asked for this addition as diacritics reduce the ambiguity of words.

This process is tedious as it requires the annotator to move the cursor letter by letter to add diacritics. Since the number of the possible diacritization patterns is low, we enable the use of morphological analysers to generate the possible diacritization of a word. The annotation process is then eased by only selecting the correctly-diacritized word. The annotator has the ability, though, to edit the form if no appropriate solution is provided.

Additionally, Wasim uses a diacritization tool (Abdulrahman Alosaimy & Atwell, 2018) that borrows more thorough diacritization forms in similar contexts. This method is different from most other diacritizer as it does not "compute" diacritization, but rather "borrows" it if the word is found in a similar context. Context can be defined in different ways: e.g. n-word-gram.

Wasim allows moderators to enforce standards on the diacritization. For example, in Arabic, it can be configured to ignore diacritization of letters preceeded by a long vowel. These transformation rules can be enforced using a set of regular expressions (regex)[5]. These rules will only be applied to a subset of morpheme/words that conform to certain conditions. For example, in the guidelines of SAC, we require no diacritization on the Lam letter of the definite article "Al-". We had a rule that removes such diacritization of the subset morphemes that has a POS tag: DET.

6.3 POS tagging

POS tagging in Wasim is morpheme-based. We assume that the tag set is assignable to any morpheme regardless of its location (e.g. prefix or base). Tags can be easily chosen from a list of POS tags ordered by their frequency or alphabetically. The most common POS tags are shown at the top, and pressing its associated number will assign it to the current in hand morpheme.

[5] A regular expression, or regex is famous way to define a search and replace pattern.

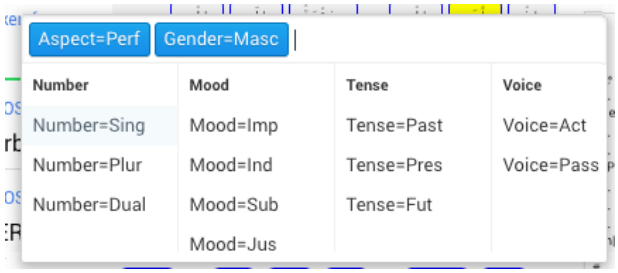


Figure 3: Features annotation popup one-line input with auto-complete feature of a VERB token.

6.4 Morphological features

Morphological features can be easily marked through a popup that offers a single input line for all morphological features together. This popup, shown in Figure 3, offers keyboard navigation to select the features. It also acts as a search input, so that only features that match the input text are visible.

Only the subset of morphological features that is compatible with segment's POS tag is shown. For example, "Mood" is only shown with VERBs. The compatibility table is configurable, but by default, we used the computability of UPOS tag and UD morphological features.

Once the input gets the focus of the user, it shows a drop-down list of all possible values. Once a value is selected (e.g. "MASC" for gender), other incompatible values hide accordingly. The goal is to speed up the annotation by selecting values in one place and asking for relevant morphological features only.

6.5 Lemmatization

Wasim offers a simple interface for lemmatization. If it is integrated with a morphological analyser, the lemma of the chosen solution will be assigned. The lemma, however, can be edited manually.

6.6 Sentence Segmentation layer

Wasim provides the ability to alter the text and separate one sentence into two. By convention, ConLL-U format leaves an empty line as an indicator of sentence start/end.

7. Case Studies

We provide four case studies to show the use of four languages. In each case, we evaluate one major feature and the effect of that feature on the speed and accuracy.

In each case, we annotate a paragraph (an average of 70 words) depending on the target language of the case. While the text size is small and might not clearly show the improvement, these experiments are for illustration purposes rather than to actually measure the difference. In addition, the annotator who has done these four experiments is the author of the tool, therefore, most of the effect of the learning curve is excluded.

For each case, the text is divided into two halves, H1 and H2, and both halves are tagged twice (two rounds). In all cases and for both rounds, the annotator is the same person. Both halves are tagged with the feature enabled (F=True) and then disabled (F=False) but in a different order for each half. The steps are {H1$_{F=True}$,H2 $_{F=False}$,H1

$_{F=False}$,H2 $_{F=True}$ }, and first two steps are first round. In the last two steps, the annotator already knows the texts and should annotate it faster. However, results between step 3 and 4 are comparable as the word counts are similar.

In Arabic cases, we used Quranic Arabic Corpus and asked the annotation to follow its annotation guidelines, and the annotator understands well its tagset. UDPipe is trained as well on Quranic Arabic Corpus (Dukes & Habash, 2010) (converted to CoNLL-U by the author and available on Github[6]). The morphological analyser used here is MADAMIRA and its results are parsed and converted to CoNLL-U format using Sawaref toolkit. A manual mapping from MADAMIRA tagset to QAC is defined and used.

Time is used as a metric for efficiency. The Intra-rater reliability is high in all cases which shows that using features does not affect the accuracy. Mismatches between the two rounds are reviewed and corrected in a third round. The accuracy in terms of the fraction of correctly annotated words is then evaluated for the two rounds compared with the gold standard (third round). More metrics are reported per case requirement. In all cases, we only evaluate the accuracy of segmentation and POS tagging, although all tasks are done. Diacritization, lemmatization, and other features accuracy are not included. At the end, we show summary statistics on our Sunnah Arabic Corpus Annotation.

7.1 Modern Standard Arabic and Morphological Analyser

In this case, the annotator used the morphological analyser to select one candidate analysis from a list of proposed analyses. "Using MA" reports the case of annotators selecting an analysis even though such analysis was corrected later. We report the number of times the annotator used MA and the number he edited the proposed analysis. Clearly, the results show that using MA is helpful in speed and accuracy, but in most cases, it is prone to errors. Using MA improved the annotation accuracy and speed significantly.

	Using MA		Without	
	round 1	round 2	round 1	round 2
Word count	50	51	51	50
Morphs count	72	70	70	72
Accuracy	96%	100%	84%	84%
Time (secs)	1358	635	1819	1729
Time (s/m)	18.86	9.07	25.99	24.01
Uses of MA	39	43	-	-
Number of edits	30	31	-	-

Table 2: Using MA feature comparison.

7.2 Quranic Arabic and Consistency Reinforcement (CR)

In this case, we show how the warning and helper guidelines help to improve the accuracy. Consistency Reinforcement feature used the whole QAC corpus to build the list of homographs and their segmentation and tagging. We report the number of homographs that were displayed on the screen. 5-8 out of 25-24 morphemes shows the high number of homographs in the Quranic Arabic Corpus (a case of highly inflectional language).

	Using Consistency Helper		Without	
	Step 1	Step 4	Step 2	Step 3
Word count	15	16	16	15
Morphs count	25	24	24	25
Accuracy	100%	100%	100%	93%
Time (secs)	269	278	331	284
Time (s/m)	10.76	11.58	13.79	11.36
homographs	5	8	-	-

Table 3: The accuracy and speed when using CR feature.

7.3 Sunnah Arabic and Keyboard Navigation

In this case, we ask the annotator not to use the keyboard for navigation except for typing the correct form or diacritization. We also report the number of mouse clicks vs. the number of uses of keyboard key presses.

	Using Keyboard		Using Mouse	
	Step 1	Step 4	Step 2	Step 3
Word count	31	30	30	31
Morphs count	38	37	37	38
Accuracy	100%	100%	100%	100%
Time (secs)	355	307	677	262
Time (s/m)	9.34	8.3	18.3	6.89
Presses/clicks	131	166	147	87

Table 4: The accuracy, speed, keyboard presses and mouse clicks comparison with two modes.

7.4 English and UDPipe

In this case, we show that Wasim is language agnostic and can work for non-highly inflectional and/or left-to-right languages as well. We used a trained model of English Treebank (provided by UD project) to kick start the annotation process of assigning universal POS tags. We do not show the effect of adaptive training UDPipe model since the text excerpt is too small. Obviously, tagging English text is more efficient since it is not an inflectional language, and is not morphologically rich compared to Arabic.

	Using Tagger		Without	
	round 1	round 2	round 1	round 2
Word count	31	30	30	31
Accuracy	96%	100%	96%	90%
Time (secs)	67	47	170	203
Time (s/w)	2.16	1.57	5.67	6.55
No. of Edits	0	0	1	3

Table 5: Comparison between using with and without MA

7.5 General Case: Sunnah Arabic Corpus

We have used Wasim for the ongoing project of morphological annotation of the SAC. So far, words have an average of 1.3 morphemes, and we spend 10.9 secs/morpheme on average to annotate a morpheme with all features enabled, i.e. 9.17 morphemes per minute. Features include POS tagging, segmentation, lemmatization, and six morphological features.

[6] https://github.com/aosaimy/qac.conllu

In the SAC, the speed of the annotation is rising over time due to two reasons: the automatic tagger becomes more accurate over time, the annotators are gaining experience. Obviously, the speed of annotation depends on several factors like text, language, course vs fine-grained tagging, and annotator experience. Therefore, reported speed measures should be viewed with caution.

8. Conclusion

We presented Wasim, an open-source web-based tool efficiency-oriented for semi-automatic annotation of inflectional languages resources. Wasim supports multiple tasks including segmenting tokens, diacritizing and labelling tokens and segments. It integrates the UDPipe toolkit to kick-start the annotation process and can be integrated with a morphological analyser to speed up the annotation process. We illustrated the improvement in accuracy and time in four cases with different genres and languages.

For future work, we plan to add support for additional layers for syntax, co-referencing, and named entities.

9. Bibliographical References

Alosaimy, A., & Atwell, E. (2017, December). Sunnah Arabic Corpus: Design and Methodology. *Proceedings of the 5th International Conference on Islamic Applications in Computer Science and Technologies (IMAN 2017)*.

Alosaimy, A., & Atwell, E. (2018). Diacritization of a Highly Cited Text: A Classical Arabic Book as a Case. In *2nd IEEE International Workshop on Arabic and derived Script Analysis and Recognition (ASAR 2018)*. London, UK.

Dukes, K., & Habash, N. (2010). Morphological Annotation of Quranic Arabic. *LREC*.

Gerdes, K. (2013). Collaborative Dependency Annotation. In *DepLing* (pp. 88–97).

Marcel Bollmann, Florian Petran, Stefanie Dipper, J. K. (2014). CorA: A web-based annotation tool for historical and other non-standard language data. *Proceedings of the 8th Workshop on Language Technology for Cultural Heritage , Social Sciences , and Humanities (LaTeCH)*, 86–90.

Nivre, J., & Agic, L. ˇZeljko. (2017). Universal dependencies 2.0 CoNLL 2017 shared task development and test data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Samih, Y., Maier, W., & Kallmeyer, L. (2016). SAWT: Sequence Annotation Web Tool. *EMNLP 2016*, 65.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., & Tsujii, J. (2012). BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 102–107).

Straka, M., & Straková, J. (2017). Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies* (pp. 88–99). Vancouver, Canada: Association for Computational Linguistics.

Yimam, S. M., Gurevych, I., de Castilho, R. E., & Biemann, C. (2013). WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *ACL (Conference System Demonstrations)* (pp. 1–6).

HiNTS: A Tagset for Middle Low German

Fabian Barteld[*,†,‡], **Sarah Ihden**[*,†], **Katharina Dreessen**[†], **Ingrid Schröder**[†]
[*] these authors contributed equally
[†]Department of Language, Literature and Media, Institute for German Studies, Universität Hamburg, Germany
[‡]Language Technology Group, Department of Informatics, Universität Hamburg, Germany
firstname.lastname@uni-hamburg.de

Abstract

In this paper, we describe the "Historisches Niederdeutsch Tagset" (HiNTS). This tagset has been developed for annotating parts-of-speech and morphology in Middle Low German texts, a group of historical (1200–1650) dialects of German. A non-standardized language such as Middle Low German has special conditions and requirements which have to be considered when designing a tagset for part of speech and morphology. We explain these requirements, i.e. the need to encode ambiguities while allowing the annotator to be as specific as possible, and our approach for dealing with them in the tagset. We then describe two special features of the tagset. In order to prove the benefit of these tags and corresponding annotation rules, we present example searches and the possible analyses arising from the results of such searches. Besides the usefulness of our tagset, we also considered its reliability in annotation using inter-annotator agreement experiments. The results of these experiments are presented and explained.

Keywords: non-standard language, part-of-speech tagset, morphological tagset

1. Introduction

In this paper, we describe a tagset called "Historisches Niederdeutsch Tagset" (HiNTS). The tagset was developed at the University of Hamburg in the context of the project 'Reference Corpus Middle Low German / Low Rhenish (1200–1650)' (ReN). The aim of the project is the creation of an annotated corpus of Middle Low German (GML) texts, a group of historical dialects of German that were used between 1200 and 1650 in the northern part of the German language area and in the Baltic regions. The corpus will be part of the 'Corpus of Historical German Texts', together with the corpora 'Altdeutsch' (Old German), 'Mittelhochdeutsch' (Middle High German), and 'Frühneuhochdeutsch' (Early New High German).[1] ReN provides diplomatically transcribed Middle Low German and Low Rhenish texts from 1200 to 1650 that are lemmatised and annotated with part of speech (POS) and morphology using HiNTS, the tagset described in this paper. Since GML is a historical language, which is not well described and hardly any resources for annotating GML texts exist, there are special requirements for the tagset which we will focus on in this paper.

When annotating a historical language, the annotators lack the intuition of a native speaker. Therefore, decisions about a category can only be based on the comparison and statistical analysis of texts (Dipper et al., 2013, 2). This however, should be avoided while annotating a text and left for later analysis using the annotated data. We present design decisions for our POS tagset that result from this requirement.

HiNTS is also used to annotate morphology. Here, many tokens are ambiguous, for example, as regards their inflectional ending. To allow the annotators to encode such an ambiguity, HiNTS allows the combination of multiple features of one inflectional category.

[1]The project is described in Peters and Nagel (2014) and on the website `www.referenzkorpus-mnd-nrh.de`. For information on the annotation used in ReN and possible grammatical analyses, see Schröder (2014).

In the following section, we present related work and detail the special needs for a tagset designed to annotate a historical language. We subsequently describe special features concerning the POS (Section 3) and the morphological tags (Section 4) of HiNTS dealing with the relevant requirements. To evaluate our design choices, we show example queries that illustrate the usefulness of the tagset. Furthermore, we present inter-annotator agreement experiments which show that GML can be annotated reliably using HiNTS.

2. Related Work and Requirements for a Tagset

The Stuttgart-Tübingen Tagset (STTS, Schiller et al. (1999)) is the de-facto standard for the annotation of German texts with parts of speech and morphology. In recent years, it has been adapted to non-standard, for example historical, texts (Zinsmeister et al., 2014). One of these adaptations is the Historical Tagset (HiTS, (Dipper et al., 2013)) that has been developed for the annotation of Old German (750–1050), Middle High German (1050–1350) and Early New High German (1350–1650) texts in the context of a group of projects that aim to create a 'Corpus of Historical German Texts'.

One of the additions that HiTS introduced is the distinction between a type-based and a token-based tag. This allows the tagset to encode differences in the function of a word as a result of a grammatical change (Dipper et al., 2013, 15). However, during the annotation process other ambiguities may appear (Barteld et al., 2014; Seemann et al., 2017). These might have different reasons and can only be resolved using an aforementioned statistical analysis in certain cases. However, even in these cases, such an analysis might best be left till after the corpus is annotated. Furthermore, the annotators might not be able to distinguish these cases from other cases, where the ambiguity cannot be resolved.

Consequently, HiNTS – which is derived from HiTS for the annotation of GML texts – is designed following two

principles: on the one hand, it aims to allow the annotator to assign tags that are not fully specified while on the other hand it allows the annotator to be as specific as possible. We will illustrate this on the level of POS and of morphology in this paper.

3. Special POS Tags in HiNTS

According to our requirements for a tagset, we developed some specific tags and annotation rules for HiNTS. For POS, one of the main differences between HiNTS – our tagset for GML – and HiTS – the tagset used for Old German, Middle High German and Early New High German – lies in the category of determiners and pronouns. According to HiTS, determiners are lexemes that can appear either determining a noun – such as *this* in *this speech* – or substituting for a noun and thus, constituting a noun phrase on its own such as *this* in *this is the beginning*. In contrast, pronouns are lexemes that never determine but always substitute for a noun – hence, a pronoun is always the head of a noun phrase, e.g. the indefinite pronoun *somebody*. In such a lexeme-based approach, it is not possible to decide between pronoun and determiner based on a specific instance where the word appears as the head of a noun phrase. Only if another instance of the same word exists where it is used in the determiner slot, can it be said that the word in question is a determiner. In the case of pronouns on the other hand, it cannot be decided definitely that a word is a pronoun based on corpus data. Still, lexemes can be distinguished that are pronouns with a high probability, if they appear frequently in a corpus and only as heads of noun phrases. The problem of a lexeme-based approach is that you have to know while annotating which words can only appear as pronouns. Especially for less widely researched historical languages – where no native speaker intuition is available –, this is not possible without any doubt or any exception. Moreover, given the current state of knowledge about GML, even scholars of this language will not be able to resolve the ambiguity between pronoun and determiner. This holds for many other aspects of GML as well.

Therefore, our tagset HiNTS avoids lexeme-based assumptions: the categories should be assignable given a single instance. In the case of determiners and pronouns, HiNTS simply distinguishes between annotation units determining a noun, such as *dyt* 'this' in *dyt ghut kanstu allene nyth ghewynnen* 'you cannot get this property by your own' and tokens providing the head of a noun phrase such as *nemant* 'nobody' in *dat my nemant kunne lyken* 'that nobody could resemble me'.

After the annotation of a corpus using HiNTS, this allows one to identify all lexemes which only appear as heads of noun phrases – substituting for a noun – and thus are likely to be pronouns in the basic word meaning.

A further difference between HiNTS and HiTS concerns the distinction between coordinating and subordinating conjunctions. In Modern German main clauses and subordinate clauses can be easily distinguished: in main clauses the finite verb usually stands in the second position whereas in subordinate clauses the verb stands in the final position. In GML word order is more open to variation. Due to this, it is not always possible to distinguish between main and subordinate clauses and consequently between subordinating and coordinating conjunctions. This is especially true if the conjunction is ambiguous. According to Schiller and Lübben (1875–1880), the word *wente* for example can mean 'denn' ('because'; coordinating) or 'weil' ('because'; subordinating), 'dass' ('that'; subordinating) or 'aber' ('but'; coordinating). This ambiguity is exemplified in (1). Härd (2008, 1461) states that *wente* can be a coordinating as well as a subordinating conjunction.

(1) vnde ik sach et · vnde betugede et · **wente** dit is godes sone
'and I saw it and attested it **because/that/but** this is god's son'
(Buxteh. Ev.)[2]

Often, in sentences with ambiguous conjunctions such as *wente*, it is very difficult and sometimes even impossible to decide whether it is a coordinating or a subordinating conjunction. In order to avoid a potentially wrong interpretation by the annotator and to provide a consistent annotation, structure-based rules are necessary. In HiNTS the tag 'KON' is used for conjunctions in verb-second sentences, e.g. *wente dit is godes sone*, while 'KOUS' is the tag for the conjunction in a sentence with the finite verb in a later position than verb second (including verb final, the position where verbs with subordinating conjunctions appear in Modern German), e.g. *wente dit godes sone is*. In ambiguous sentences such as *wente he kam* ('because/but/that he came') where the finite verb could stand in the second as well as in the final position the tag 'KO*'[3] is used. The advantage of such an annotation principle is that the user of the corpus can search for specific structural contexts of a conjunction. In that way, one can investigate which conjunction is used how often in combination with the respective position of the finite verb. This helps historical linguists to research change processes concerning the position of the finite verb in coordinated and subordinate clauses in historical German.

In the following, we will illustrate this by the results of a search for the prototypical subordinating conjunction *dat* in the third pre-release of the "Reference Corpus Middle Low German / Low Rhenish (1200–1650)" (ReN) (ReN-Team, 2017) containing 32 texts. In Modern German, a subordinator usually appears only in sentences with verb final position. In GML there is much more variation. As the results in Table 1 show, in $1,775$ sentences with the subordinator *dat* the finite verb appears in a later position than verb second. This includes not only sentences like (2) with a verb final position like in Modern German but also sentences like (3) where the finite verb stands in a later position than verb second but not in the last position. As Dreessen and Ihden (2015) have shown the reason for this divergent word order lies in the structure 'finite verb before infinite verb' within the verbal complex as in (3) as well as in the

[2]The examples are taken from the Reference Corpus Middle Low German / Low Rhenish (1200–1650). More information on the particular texts can be found in the metadata of the corpus.

[3]The asterisk is not part of a regular expression here but simply marks the ambiguity.

POS tag	Frequency
KOUS	1,775
KON	260
KO*	68
total	2,103

Table 1: POS tags of the conjunction 'dat' in ReN (ReN-Team, 2017)

	Frequency (token)	Frequency (lemma)
Masc-Neut	1219	178
Masc-Fem	765	142
Neut-Fem	326	70
total	2310	390

Table 2: Ambiguity between two genders in ReN

post-field. Moreover, Table 1 shows that there are even 260 *dat*-sentences with a verb-second structure (the word order of coordinating sentences in Modern German) such as in (4) (annotated as 'KON'). Here, the reason for the divergent word order again is the post-field (*eyn droghenere*). Both the structure 'finite before infinite verb' in the verbal complex and the post-field are frequently used in GML texts. In a lexeme-based approach one would have annotated the same tag for the conjunction *dat* in all of the sentences. Thus, no information on divergent word order in the sentences beginning with a conjunction – a special feature of historical German – would become visible through the tags.

(2) weret szake · **dath** ghy dar eynn jar effte sossze blyuen *worden*
'[if] it was the case that you would stay there one year or six'
(Agneta Willeken)

(3) O wy vnde we · **dat** ik gy *wart* gheboren
'oh alack and alack that I ever was born'
(Bord. Marien-Kl.)

(4) Se seden · **dat** vnse here *were* eyn droghenere
'they said that our lord was a cheat'
(Buxteh. Ev.)

As we pointed out, the structure-based annotation rule in HiNTS concerning conjunctions provides a consistent annotation without a potentially wrong interpretation by the annotator. Moreover, the annotation is absolutely transparent to users of ReN which is a crucial advantage for search queries and the further work with the results.

4. Special Features of the Morphological Tagset: Annotating Ambiguity

A great challenge in annotating non-standard languages such as historical languages is the annotation of ambiguity (Barteld et al., 2014). In GML, words that are morphologically ambiguous in the view of a researcher are rather frequent, for example concerning grammatical gender. Some of the nouns having one specific grammatical gender in Modern German could occur with different genders in historical German. In (5), according to the GML dictionary by Lasch et al. (1956ff), the noun *spegel* 'mirror' can be either a masculine or a neuter noun. The annotators should try to use contextual clues, e.g. the form of a determiner, to decide between possible genders for a specific instance. However, this is often not possible. In (5), for instance, the linguistic context does not allow a definite disambiguation for the gender of *speyghel*[4], since there is no determiner of this noun giving a clue concerning its gender: The word *der* is part of a genitive construction determining the noun *Sassen*. Disambiguation could for example be enabled by the definite article for a masculine noun *de* (*dit is de spegel*) or the one for a neuter noun *dat* (*dit is dat spegel*). Hence, it is not possible to assign an unambiguous gender to *speyghel*. In such cases, annotators should have the option of encoding this ambiguity. STTS and HiTS use the asterisk (*) for this. According to this, the token *speyghel* in (5) would be annotated with 'NA.*.Nom.Sg'. The problem with this mark-up is that it only encodes that there is a gender ambiguity but not that the female gender is not possible. According to our requirements for a tagset explained above, in HiNTS as much information as possible must be provided. Therefore, we use portmanteau tags (Leech et al., 1994), which allow the tagset to exclude the female gender in (5) and to represent the ambiguity between the masculine and neuter noun. Hence, in (5) the complete tag for the token *speyghel* in HiNTS is 'NA.Masc-Neut.Nom.Sg'.

(5) Dit is der sassen **speyghel**
'this is the Saxon's mirror'
(Oldb. Ssp., headline)

In search queries one could use the gender portmanteau tags to find out which ambiguity between two genders occurs most frequently. Table 2 shows that in ReN (ReN-Team, 2017) the most frequent ambiguity is the one between the masculine and the neuter gender; the ambiguity between the neuter and the feminine gender occurs least frequently. By having a closer look at one of the lemmas with gender ambiguity then, one could also examine which gender is used more often in the unambiguous cases. Table 3 shows the results for the two examples *lîf*[1] 'life' and *strît*[1] 'fight'.[5] Whereas concerning the lemma *lîf*[1], there are more unambiguous proofs for the neuter (34) than for the masculine gender (11), the lemma *strît*[1] provides no unambiguous proofs for the neuter but 24 for the masculine gender. This could lead to the assumption that in contrast to the lemma *lîf*[1], the lemma *strît*[1] is generally used as a masculine noun and that the neuter noun is on the fringes. When the corpus is completed, this hypothesis could be examined on a wider range of texts.
This shows how encoding all possibilities in the case of gender ambiguity as portmanteau tags can help us to understand the gender system of GML.

[4]This is a spelling variant of *spegel*.

[5]The numbers denote the different word senses according to the GML dictionary by Lasch et al. (1956ff).

	Frequency	
	lîf[1]	strît[1]
Masc-Neut	85	60
Masc	11	24
Neut	34	0
total	130	84

Table 3: Gender of lîf[1] and strît[1]

However, one could suppose that such a detailed annotation as above could be more difficult for the annotator than only using the asterisk and thus could lead to more disagreement between annotators. We will address this issue – among others – in an inter-annotator agreement experiment described in the following section.

5. Inter-Annotator Agreement

In the previous sections, two principles guiding the creation of the tagset HiNTS have been described: firstly, leaving decisions between categories open (as with pronoun and determiner) while annotating formal features that are correlated with these categories (concerning pronoun/determiner: determining or substituting a noun; concerning conjunctions: the position of the finite verb). Secondly, the tagset contains portmanteau tags for specifying morphological ambiguity.

For the assessment of inter-annotator agreement, a segment of a GML text has been annotated independently by two annotators. While annotating POS and morphology, the annotators are allowed to change the segmentation (and also to correct the transcription of the text). This has to be taken into account when calculating the inter-annotator agreement. The annotations are aligned using the method described in Barteld et al. (2016). For this study, only tokens that were aligned by the described method are considered. These are 988 tokens.

Looking only at the coarse-grained POS tags (i.e. without morphology), the overall percentage agreement is 94.33%. This is lower than what has been reported for Modern German with a comparable tagset, e.g. 98.57% for a newspaper corpus (Brants, 2000). But it is quite high, when taking the non-standard nature of the historical texts into account. Scheible et al. (2011) report 91.6% for Early Modern German (1650–1800).

Table 4 shows the agreement for different subsets of the part-of-speech tagset. The numbers for *conjunctions*, i.e. the tags *KOUS*, *KON* and *KO** is higher than the overall average and also higher than most of the other subsets of the non-inflected parts of speech. This is despite the fact that the distinction between subordinating and coordinating is ambiguous in GML and shows that the categorization scheme presented in Section 3 – along with its usefulness for users – can be applied with a rather high agreement. Table 5 shows the confusion matrix of the respective tags. This matrix shows that while the annotators mostly agree, there are deviations between the annotators in the seemingly simple case of distinguishing between subordinating and coordinating conjunction, despite the structural rule that should be applied by the annotators. This supports the assertion that the distinction between coordination and subordination is not straightforward for GML.

Tags	Agreement (%)
Inflected	91.29
Non-inflected	96.06
Prepostions	100.00
Conjunctions	95.31
Particles	94.44
Adverbs	90.00
All	94.33

Table 4: Agreement

	KO*	KON	KOUS
KO*	0	0	0
KON	0	49	0
KOUS	1	2	12

Table 5: Confusion matrix for KON, KOUS and KO*

Allowing the annotators to include ambiguity in the annotation as described in Section 4 leads to a huge tagset with $42,752$ individual tags. However, many of the possible ambiguous combinations in morphology will only be theoretically valid and never appear in the corpus. In the annotated segment, one annotator used 194 tags, the other 196. When simplifying the annotations based on the scheme used by STTS and HiTS, i.e. encoding ambiguity with *, these numbers are reduced only marginally to 191 for both annotators. Using this tagset, an agreement (Cohen's κ, Cohen (1960)) of 0.818 is reached. The agreement is similar (0.819) when only using "*" to encode ambiguity. This is partly due to the fact, that the annotators often disagreed about ambiguity itself: one annotator marked 54 tokens as ambiguous, the other annotator 95. This leads to another argument for the more specific annotation of ambiguity. Using only "*", there is no information on the possible tags – when adding all possible genders, cases and so on, this information is kept.

6. Conclusion

We showed that a historical language such as GML brings special conditions and requirements having to be considered in a tagset for annotating POS and morphology of GML texts: GML is not well described and there are nearly no resources which could be used for the annotation. Moreover, in contrast to the current German language, GML cannot be annotated based on the intuition of a native speaker. Since the annotation of the texts should be used for a new grammar of GML, one of our most important requirements is that some decisions should not be made before annotating but be left for later studies in order not to anticipate the results coming from the analysis of the annotated texts. Therefore, in the POS part of the tagset, we avoid lexeme-based assumptions (as for the group of determiners and pronouns) and prefer structure-based rules. Concerning the distinction of coordinators and subordinators, such rules reduce the influence of potentially wrong interpretations of the annotators and thus ensure a consistent annotation. As we showed in an example search, these rules could

for example be used to analyse the position of the verb in sentences beginning with a conjunction.

The greatest challenge in annotating the morphology of GML are ambiguous forms, such as those concerning grammatical gender. In order to provide as much information as possible, we use portmanteau tags. An example query on nouns with gender ambiguity illustrated how our portmanteau-tag annotations could be used for studies on the frequency of specific gender ambiguities and the dominance of one gender in unambiguous forms.

According to the results of our inter-annotator agreement experiments concerning POS, the overall percentage agreement (94.33%) is quite high for non-standard texts. Moreover, within the group of conjunctions, agreement is even higher than the overall rate and higher than most of the other non-inflected parts of speech. This shows that our structure-based annotation rule concerning conjunctions is not only of benefit for the user but it can also be applied by the annotators reliably. Concerning morphology, the results of the inter-annotator agreement experiments have shown that the use of portmanteau tags in our corpus does not lead to a significantly higher rate of disagreement and therefore is not only useful but can also be annotated as reliably.

7. Acknowledgements

The authors are funded by the German Research Foundation (DFG) as part of the project "Referenzkorpus Mittelniederdeutsch/Niederrheinisch (1200–1650)".

We would like to thank the anonymous reviewers for their helpful remarks and Piklu Gupta for improving our English.

8. Bibliographical References

Barteld, F., Ihden, S., Schröder, I., and Zinsmeister, H. (2014). Annotating descriptively incomplete language phenomena. In *Proceedings of LAW VIII - The 8th Linguistic Annotation Workshop*, pages 99–104, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Barteld, F., Schröder, I., and Zinsmeister, H. (2016). ${}_t\gamma$–Inter-annotator agreement for categorization with simultaneous segmentation and transcription-error correction. In Stefanie Dipper, et al., editors, *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, Bochumer Linguistische Arbeitsberichte, pages 27–37, Bochum, Germany, September.

Brants, T. (2000). Inter-annotator agreement for a German newspaper corpus. In *Proceedings of the Second International Conference on Language Resources and Evaluation, LREC 2000*, Athens, Greece, May/June. European Language Resources Association.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Dipper, S., Donhauser, K., Klein, T., Linde, S., Müller, S., and Wegera, K.-P. (2013). HiTS: ein Tagset für historische Sprachstufen des Deutschen [HiTS: A tagset for historical variants of German]. *Journal for Language Technology and Computational Linguistics*, 28(1):1–53.

Dreessen, K. and Ihden, S. (2015). Korpuslinguistische Studien zur mittelniederdeutschen Syntax [Corpus studies on GML syntax]. In Markus Hundt et al., editors, *Deutsch im Norden*, volume 6 of *Jahrbuch für Germanistische Sprachgeschichte*, pages 249–275. Walter De Gruyter, Berlin, Boston.

Härd, J. E. (2008). Syntax des Mittelniederdeutschen [GML syntax]. In Werner Besch, et al., editors, *Sprachgeschichte. Ein Handbuch zur Geschichte der deutschen Sprache und ihrer Erforschung, 2. Teilband*, volume 2 of *Handbücher zur Sprach- und Kommunikationswissenschaft*, pages 1456–1463. De Gruyter Mouton, Berlin, Boston, 2nd edition.

Lasch, A., Borchling, C., Cordes, G., Möhn, D., and Schröder, I. (1956ff). *Mittelniederdeutsches Handwörterbuch [Concise dictionary of GML]*. Kiel/Hamburg.

Leech, G., Garside, R., and Bryant, M. (1994). CLAWS4: The tagging of the British National Corpus. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING 1994, Volume 1*, pages 622–628, Kyoto, Japan, August.

Peters, R. and Nagel, N. (2014). Das digitale ‚Referenzkorpus Mittelniederdeutsch / Niederrheinisch (ReN)' [The digital 'Reference Corpus Middle Low German/ Low Rhenish (ReN)']. In Vilmos Ágel et al., editors, *Paradigmen der Sprachgeschichtsschreibung*, volume 5 of *Jahrbuch für Germanistische Sprachgeschichte*, pages 165–175. Walter De Gruyter, Berlin, Boston.

Scheible, S., Whitt, R. J., Durrell, M., and Bennett, P. (2011). A gold standard corpus of Early Modern German. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 124–128, Portland, Oregon, USA, June. Association for Computational Linguistics.

Schiller, K. and Lübben, A. (1875–1880). *Mittelniederdeutsches Wörterbuch [GML dictionary]*. Aschendorff, Münster.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS [Guidelines for tagging German corpora with the STTS]. Technical report, University of Stuttgart and University of Tübingen.

Schröder, I. (2014). Das Referenzkorpus: Neue Perspektiven für die mittelniederdeutsche Grammatikographie. [The Reference Corpus: New perspectives for GML grammaticography]. In Vilmos Ágel et al., editors, *Paradigmen der Sprachgeschichtsschreibung*, volume 5 of *Jahrbuch für Germanistische Sprachgeschichte*, pages 150–164. Walter De Gruyter, Berlin, Boston.

Seemann, N., Merten, M.-L., Geierhos, M., Tophinke, D., and Hüllermeier, E. (2017). Annotation challenges for reconstructing the structural elaboration of Middle Low German. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 40–45, Vancouver, Canada, August. Association for Computational Linguistics.

Zinsmeister, H., Heid, U., and Beck, K. (2014). Adapting a part-of-speech tagset to non-standard text: the case of

STTS. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 4097–4104, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

9. Language Resource References

ReN-Team. (2017). *Referenzkorpus Mittelniederdeutsch/Niederrheinisch (1200-1650)*. Archived in Hamburger Zentrum für Sprachkorpora, Version 0.3. Publication date 2017-06-15. http://hdl.handle.net/11022/0000-0006-473B-9.

Leveraging Lexical Resources and Constraint Grammar for Rule-Based Part-of-Speech Tagging in Welsh

Steven Neale[1], Kevin Donnelly[2], Gareth Watkins[1], Dawn Knight[1]
[1]School of English, Communication and Philosophy
Cardiff University, Wales, United Kingdom
{NealeS2, WatkinsG13, KnightD5}@cardiff.ac.uk
[2]Independent Researcher
kevin@dotmon.com

Abstract

As the quantity of annotated language data and the quality of machine learning algorithms have increased over time, statistical part-of-speech (POS) taggers trained over large datasets have become as robust or better than their rule-based counterparts. However, for lesser-resourced languages such as Welsh there is simply not enough accurately annotated data to train a statistical POS tagger. Furthermore, many of the more popular rule-based taggers still require that their rules be inferred from annotated data, which while not as extensive as that required for training a statistical tagger must still be sizeable. In this paper we describe *CyTag*, a rule-based POS tagger for Welsh based on the VISL Constraint Grammar parser. Leveraging lexical information from *Eurfa* (an extensive open-source dictionary for Welsh), we extract lists of possible POS tags for each word token in a running text and then apply various constraints – based on various features of surrounding word tokens – to prune the number of possible tags until the most appropriate tag for a given token can be selected. We explain how this approach is particularly useful in dealing with some of the specific intricacies of Welsh - such as morphological changes and word mutations - and present an evaluation of the performance of the tagger using a manually checked test corpus of 611 Welsh sentences.

Keywords: Part-of-speech, lexical resources, open-source dictionaries, Constraint Grammar

1. Introduction

POS tagging is a well-explored problem in NLP, and highly-accurate taggers have been built using statistical and probabilistic methods for decades. These taggers are typically trained using already-annotated text, from which the probabilities of POS tags being appropriate for certain word tokens are calculated based on features such as the lexical properties of tokens (capitalisation, common prefixes and suffixes etc.) or the POS tags of their *n*-neighbouring tokens. However, the amount of pre-annotated data that these taggers require to be properly trained is considerable, and usually in the region of many hundreds of thousands to millions of word tokens.

For languages such as Welsh – for whom resources are typically more scarce – pre-annotated data in context in these kinds of quantities is very difficult either to create or to obtain. The traditional alternative to probabilistic POS tagging is the rule-based approach, whereby tags are assigned based on pre-defined rules concerning which syntactic categories can be co-located together. Crafting and refining the rules by hand is, however, almost as costly and labour-intensive as producing manually-annotated data from which to train a probabilistic POS tagger.

This paper introduces *CyTag*[1], a rule-based tagger that leverages an open source dictionary and uses the VISL Constraint Grammar parser to assign POS tags to Welsh words in context, using a minimal set of easily-adaptable rules and without the need for millions of tokens of pre-annotated data. Evaluating our tagger using a gold standard dataset consisting of 611 manually checked sentences, we obtain high precision and excellent recall at a level comparable with POS tagging accuracies reported in Welsh and expected in other languages. Our contribution is a robust, open-source and high-performing POS tagger for Welsh – crafted using minimal hard-coded rules – that demonstrates how existing lexical resources can be leveraged to construct accurate taggers for lesser-resourced languages.

2. Background

After decades of development and refinement, probabilistic POS taggers are these days highly accurate – *TreeTagger*[2](Schmid, 1994) tags with a reported accuracy of 94-96%, while the *Stanford Log-linear Part-of-Speech Tagger*[3] (Toutanova et al., 2003) is capable of tagging with an accuracy of over 97%. However, these scores are only attainable after training the tools with considerable quantities of pre-annotated data. Schmid (1994), for example, reports that the accuracy of *TreeTagger* is around 82-84% using a training corpus of 10,000 words, 91-93% with a training corpus of 100,000 words and requires a training corpus of 1 million words to achieve its reported accuracy of 94-96%.

Rule-based POS tagging – whereby pre-defined rules concerning which syntactic categories can be co-located together are used to determine the correct POS tags to assign to word tokens in context – is the traditional alternative to the probabilistic approach. However, this introduces an entirely different bottleneck – considerable time and extensive knowledge is required in order to craft and refine the rules in the first place. The widely-used *Brill Tagger* attempts to

[1] cytag.corcencc.org

[2] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[3] https://nlp.stanford.edu/software/tagger.shtml

address this bottleneck by automatically acquiring and inferring rules for POS tagging from a running text, and its accuracy has been comparable to that of probabilistic taggers(Brill, 1992). However, more pre-annotated data than is typically available for lesser-resourced languages is still required for this approach, and so again it boils down to a decision (costly either way) between crafting enough rules or annotating sufficient training data by hand.

Examples of POS tagging in Welsh are scarce, but there are some tools available that have attempted to tackle the problem. The *Welsh Natural Language Toolkit (WNLT)*[4] is a bespoke plugin for the *GATE* text processing platform[5] that handles POS tagging for Welsh by looking up terms in a pre-defined lexicon and using hard-coded rules to narrow down ambiguity (words ending in 'iant' or 'cyn' are selected as masculine nouns while those ending in 'es' or 'ell' are selected as feminine nouns, for example) and reports precision, recall and F1 scores of 81%, 82% and 81% respectively. There is also the *Welsh Parts-of-Speech Tagger API*[6], a free-to-use web-based service for tagging Welsh sentences with POS tags and mutations (see section 3.2.1.) developed by the Language Technologies Unit at Bangor University with funding from the Welsh Government.

A far more accurate approach is that adopted by the *Bangor Autoglosser*[7], a multilingual tagger developed to assign POS tags to conversational texts in Welsh, English and Spanish (Donnelly and Deuchar, 2011). The *Autoglosser* is based on Constraint Grammar (CG) (Karlsson, 1990; Karlsson et al., 1995), a language-independent parser whereby easily-adaptable rules based on surface level features and morphology are used to 'discard' ambiguous tags from a list of possible 'readings' for a word token in a running text. By producing a list of possible readings for word tokens from English, Spanish and Welsh dictionaries and pruning those readings with CG-formatted rules, Donnelly and Deuchar (2011) were able to tag the *Siarad* (Welsh-English) and *Patagonia* (Welsh-Spanish) corpora with reported accuracies of 98% and 99% respectively.

3. *CyTag* – A Constraint Grammar-based POS tagger for Welsh

3.1. Motivation – The *CorCenCC* project

Our motivations for developing a bespoke solution for Welsh POS tagging are based on the requirements, aims and scope of the *CorCenCC – National Corpus of Contemporary Welsh (Corpws Cenedlaethol Cymraeg Cyfoes)*[8] project, through which the work is funded. The aim of the project is to construct a 10 million-word corpus of the Welsh language sampled from spoken, written and e-language sources in contemporary contexts, and incorporating crowdsourced contributions to give Welsh speakers the opportunity to involve themselves directly in the creation of the corpus.

[4] http://hypermedia.research.southwales.ac.uk/kos/wnlt/

[5] https://gate.ac.uk/

[6] http://techiaith.cymru/api/parts-of-speech-tagger-api/?lang=en

[7] http://bangortalk.org.uk/autoglosser.php

[8] http://www.corcencc.org

3.1.1. The *CorCenCC* POS Tagset

An already completed task for the project has been the development of the *CorCenCC POS Tagset*[9], the current version of which contains 145 fine-grained POS tags collapsing into 13 EAGLES[10]-conformant categories. The necessity for a tagger that is compatible with this tag-set – as well as with the various transcription conventions that have been put in place for marking-up spoken contributions to the corpus – was an influencing factor in our decision to create our own POS tagger rather than to rely on an existing solution. Thus, it is from this bespoke tagset that the POS categories *CyTag* assigns are selected.

Basic	Enriched	Description
E	Ebu	**E**nw **b**enywaidd **u**nigol (noun, feminine, singular)
	Egll	**E**nw **g**wrywaidd **ll**uosol (noun, masculine, plural)
	Epg	**E**nw **p**riod **g**wrywaidd (noun, proper, masculine)
	Epb	**E**nw **p**riod **b**enywaidd (noun, proper, feminine)
	...	
B	Be	**B**erf **e**nw (verb noun, eq. infinitive verb)
	Bpres3u	**B**erf **pres**ennol, **3**ydd pers. **u**nigol (verb, present, 3rd pers. singular)
	Bdyf1ll	**B**erf **dyf**odol, pers. **1**af **ll**uosol (verb, future, 1st pers. plural)
	Bdyf2ll	**B**erf **dyf**odol, **2**il pers. **ll**uosol (verb, future, 2nd pers. plural)
	...	
Rha	Rhaperth	**Rha**genw **perth**ynol (pronoun, relative)
	Rhadib1ll	**Rha**genw **dib**ynnol, pers. **1**af **ll**uosol (pronoun, dependent, 1st pers. plural)
	...	

Table 1: Examples (selected, non-exhaustive) of the relationship between 'basic' and 'enriched' *CorCenCC* POS tags, and their granularity.

We define the EAGLES-conformant categories and the fine-grained POS tags that collapse into them as the 'basic' and 'enriched' tagsets, respectively. Thus, the 'basic' tagset is made up of 13 tags representing major syntactic categories ('noun', 'article', 'preposition', 'conjunction', 'numeral', 'adjective', 'adverb', 'verb', 'pronoun', 'interjection', and 'punctuation') plus two categories representing 'unique' particles to Welsh, and 'other' forms (such as abbreviations, acronyms, symbols, digits etc.). The 'enriched' tagset is the full set of 145 fine-grained tags that collapse down into the 13 categories from the 'basic' tagset, and largely cover different morphological features of the tags in each 'basic' tag category such as gender (masculine or feminine), number (singular or plural), person (1st person,

[9] http://cytag.corcencc.org/tagset

[10] *Expert Advisory Group on Language Engineering Standards*: http://www.ilc.cnr.it/EAGLES/browse.html

3rd person etc.), or tense (past, present, future etc.). Table 1 shows a selection of nouns, verbs and pronouns from the tagset, demonstrating how the tags themselves are formed consistently from Welsh descriptions of their morphological features (always 'll' for plural forms, always 'b' for feminine and 'g' for masculine, etc.).

3.2. The *CyTag* process

Following a similar methodology to the one behind the *Bangor Autoglosser* (Donnelly and Deuchar, 2011), *CyTag* is based on Constraint Grammar (CG) (Karlsson, 1990; Karlsson et al., 1995), and in particular is built around the latest version of the software – VISL CG-3[11]. As described in Section 2., CG works by applying rules that 'discard' ambiguity by 'pruning' a 'cohort' (list) of the available 'readings' (possible tags) for a given word token, based on the surface and morphological features of either itself or its neighbouring word tokens. Although this means that we must be able to produce a list of possible readings (POS tags, morphological information etc.) for each token in a running text, these readings do not have to be in context – it is the job of the CG rules to choose the correct reading based on context later. Thus, unlike most POS tagging solutions, the initial cohort of readings can be extracted from word-level lexica such as dictionaries, which even in lesser-resourced languages are far more readily-available than fully annotated sentences in context.

Thus, *CyTag* assigns POS tags to tokens using three steps:

- A list of possible POS tags is produced for each token,
- Using CG-formatted rules, the list of possible tags for each token is pruned to as few as possible (ideally one),
- The optimal tag for each token is selected, using additional processing to help select a tag in any cases that were still ambiguous after running CG.

3.2.1. Producing possible POS tags for each token

An initial list of possible POS tags for each token is produced over two steps. Firstly, any tokens whose definite POS can be identified without needing to look it up elsewhere are assigned the appropriate tag outright – for example, regular expressions are used to determine whether or not the token is a punctuation mark ('.', '!', '?'), a digit ('1980'), or a symbol ('£', '%'). Next, gazetteers are checked to determine whether or not the token is already known to be an acronym – such as 'GIG' ('NHS' in English) – or an abbreviation – such as 'Cyf.' ('Ltd.' in English) or 'e.e.' ('e.g.' in English).

Once all of the tokens with definite POS tags have been identified as such, the second step is to look up the remaining tokens in a pre-defined lexicon, currently containing lemmas and POS tags for 210,438 Welsh word forms. The lexicon has been extracted from *Eurfa (v3.0)*[12], the largest Welsh dictionary available under an open license and containing approximately 211,000 word forms derived from 10393 Welsh lemmas. As well as lexical categories such as nouns, verbs, and pronouns, *Eurfa* contains full morphological information for each entry – including gender, person (first, second, or third), number (singular or plural) and tense (present, past, imperfect etc.). This allows entries to be easily mapped to tags from the CorCenCC POS tagset for inclusion in the pre-defined lexicon.

After all of the possible entries for a given token have been found in the lexicon, a CG-formatted cohort of readings (list of possible POS tags for the token) can be produced. For example, given the token 'a' we get the following readings:

"<a>"
    "a" {9,18} [cy] Cys cid :and:
    "a" {9,18} [cy] Rha perth :who:
    "a" {9,18} [cy] U gof ::

Looking up the token in the lexicon has told us that there are three possible readings for 'a', which is the 18th token in the 9th sentence and is in Welsh ('[cy]') – firstly, it could be a coordinating conjunction ('Cyscid') equivalent to 'and' in English; secondly, it could be a relative pronoun ('Rhaperth') equivalent to 'who' in English; and thirdly, it could be an interrogative particle ('Ugof'), which is often used at the start of a clause in which a question is asked and has no real equivalent in English.

One particular nuance of Welsh that we begin to deal with at this stage of the *CyTag* process is mutation, a common phenomenon in Welsh in which the first letter of a word can change (or 'mutate') depending on, for example, the preceding word or on word order – some examples of how this phenomenon works can be seen in Table 2. We deal with mutation by first checking if the token begins with a known mutated form – along with a code denoting mutation type ('am' for aspirate, 'nm' for nasal, 'sm' for soft, or 'hm' for added 'h') – and then adding what would be it's de-mutated form to a list of possible tokens to look up in the lexicon. For example, *CyTag* would catch the 'ch' (aspirate mutation of the letter 'c') at the start of 'char' or the 'ngh' (nasal mutation of 'c') at the start of 'nghar' and would add 'car' (also 'car' in English) to the list of possible de-mutated forms for either 'car' or 'nghar'. To handle one particular case of a soft mutation in which the letter 'g' is dropped from the beginning of a word, we remove the letter 'g' from the start of every token we encounter that starts with it and add the now 'g'-less form to the list of possible de-mutated tokens.

If after these checks the list of possible de-mutated forms for a given token is populated, we look up each of these possible forms in the lexicon and if they are found, we add them to the cohort of readings for the token. Thus, when the readings are pruned in the second step of the *CyTag* process, any possible de-mutated forms can be taken into account by the CG-formatted rules, depending on which mutation type would have caused the change in form and on the (selected or potential) readings for preceding tokens. Given the token 'mae', for example, the original search of the lexicon will return a reading for the present tense, third-person singular verb ('Bpres3u') 'bod', equivalent

[11] http://visl.sdu.dk/cg3.html

[12] For more information, see: http://www.eurfa.org.uk/

Mutation	Effect	Example
aspirate (am)	c → ch	car → ei char (her car)
	p → ph	pensil → ei phensil (her pencil)
	...	
nasal (nm)	c → ngh	car → fy nghar (my car)
	b → m	bag → fy mag (my bag)
	...	
soft (sm)	ll → l	llyfr → ei lyfr (his book)
	rh → r	rhosyn → ei rosyn (his rose)
	...	
added 'h' (hm)	a → ha	afal → ein hafal (our apple)
	w → hw	ysgol → ein hysgol (our school)
	...	

Table 2: Examples (selected, non-exhaustive) of the four major mutation types in Welsh.

to 'is' from 'to be' in English. However, thanks to our mutation lookup rules – which know that any word word beginning with the letter 'm' could be the nasally-mutated form of a word beginning with the letter 'b' – we can also account for the possibility that 'mae' is a nasally-mutated ('nm') form of the masculine singular noun ('Egu') 'bae', equivalent to 'bay' in English:

"<mae>"
"bod" {4,19} [cy] B pres 3 u :be:
"bae" {4,19} [cy] E g u :bay: + nm

A second important nuance of Welsh that it is important to us account for when producing cohorts of readings from the lexicon is the common occurrence of elision in Welsh, which is complicated by the fact that a) shortened (or 'elided') forms of words can become attached either to the start *or* the end of surrounding words, and that b) the same elision could have potentially come from more than one full word. For example, an 'f' elided onto the beginning or an 'm' onto the end of a word could both have originally been the word 'fy'; an 'n' elided onto the end of a word could have originally been either of the words 'yn' and 'ein'; an 'r' elided onto the end of a word could have originally been either of the words 'y' and 'yr'. We deal with elisions by running a multiple lookup of the all the possible words that a particular elision could have originally been, and returning the full list of readings for those possibilities – for example, given an 'n' elided onto the end of a word as our token, we return readings for three possible forms of 'yn' – an uninflected preposition ('Arsym') corresponding to the English 'in', a predicative particle ('Utra'), and a verbal particle ('Uberf') – as well as a reading for 'ein' – a first-person plural dependent pronoun ('Rhadib1ll') corresponding to the English 'our':

"<'n>"
"yn" {8,2} [cy] Ar sym :in:
"yn" {8,2} [cy] U tra ::
"yn" {8,2} [cy] U berf ::
"ein" {8,2} [cy] Rha dib 1 ll :our:

3.2.2. Pruning the list of possible POS tags for each token

Once the cohort of possible readings for each token has been constructed, the next step in the *CyTag* process is to pass the list to VISL CG-3, which has been provided with the path to a bespoke 'grammar' file containing rules to help 'constrain' or 'prune' the readings for those tokens that are still ambiguous. The grammar currently contains 243 rules, each designed to select or remove certain options from the cohort of readings for a given token depending on the POS, morphological features and/or mutation type of its neighbouring tokens. Rules are formatted as follows:

action (reading) if (neighbour (features))

whereby *action* is what the rule should do (select the reading, remove the reading etc.), *reading* is the particular reading for a given token that the action should be performed on, *neighbour* is the neighbouring token of interest on whose features the action depends (1 for the following token, -1 for the preceding token etc.) and *features* are the POS, morphological features and/or mutation type of the neighbouring token that we expect to find in order to satisfy the rule and perform the original action on the reading.

An example in practice is the disambiguation of the token 'yn', which is often used in conjunction with the word 'mae' (from the verb 'bod', 'to be' in English) either side of a noun as a connecting particle – it can either be a predicative particle linking the noun to other nouns and adjectives, or a verbal particle linking it to another verb. For example, if we consider the short phrase:

"Mae Cymru (hefyd Saesneg: Wales) yn wlad Geltaidd"

translating to the English:

"Wales (English: Wales) is a Celtic country"

or as a very literal token-to-token translation:

"(be) Wales ... (*'yn' particle*) Celtic country"

For this segment, an initial reading could look as follows:

"<Mae>"
"bod" {377,1} [cy] B pres 3 u :be:
"<Cymru>"
"Cymru" {377,2} [cy] E p b :Wales:
...
"<yn>"
"yn" {377,9} [cy] U tra ::
"yn" {377,9} [cy] U berf ::
"yn" {377,9} [cy] Ar sym :in:

"<wlad>"
"gwlad" {377,10} [cy] E b u :country: +sm
"<Geltaidd>"
"Celtaidd" {377,11} [cy] Ans cad u :Celtic: +sm

Here, we want our CG grammar to be able to select for us that in this context the token 'yn' is the predicative particle ('Utra') linking the nouns 'Cymru' ('Wales') and 'wlad' ('country'), and not the verbal particle ('Uberf') or the preposition ('Arsym') corresponding to the English word 'in'. Assuming no previous rules have decided that the token *should* be tagged as a verbal particle or as a preposition, the grammar should eventually arrive at the following rule:

SELECT ("yn" U tra) if (1 (E));

Here, we instruct the grammar to *select* the reading where the token "yn" correspondes to the predicative particle ('Utra') *if* the following token ('1') is a noun ('E'). Because we already know that the 'yn' in the example phrase is followed by a noun ('wlad', a soft mutation of 'gwlad' – note the '+sm' in the readings), the verbal particle and the preposition are discarded and the predicative particle can be selected.
In the previous example, we can see how words that could have been mutated are represented in the cohort of readings. Another example in practice shows how the presence of possible mutations can be used by the grammar to select the appropriate reading for a given token. For example, if we consider the short phrase:

"cwmnïau cydweithredol yng Nghymru..."

translating to the English:

"cooperative companies in Wales..."

For this segment, the initial reading looks as follows:

"<cwmnïau>"
"cwmni" {58,16} [cy] E g ll :companies:
"<cwdweithredol>"
"cydweithredol" {58,17} [cy] Ans cad u ...
"<yng>"
"yn" {58,18} [cy] Ar sym :in:
"fy" {58,18} [cy] Rha dib 1 u :my:
"<Nghymru>"
"Cymru" {58,19} [cy] E b u :Wales: + nm

Here, we can see that the word 'Nghymru' is a nasal mutation ('+ nm') of the word 'Cymru' ('Wales' in English), and knowing that such mutations occur after prepositions we can implement the following rule in the CG grammar:

SELECT (Ar) IF (1 (nm));

Here, the grammar will *select* the reading corresponding to a preposition ('Ar') *if* the following token ('1') has been affected by a nasal mutation ('nm'), ensuring that the preposition corresponding to the English word 'in' is correctly selected for the token 'yn' in this context, as opposed to the pronoun ('Rhadib1u') corresponding to the English word 'my'.

3.2.3. Post-CG disambiguation steps

Once the CG grammar has pruned the cohorts to as close to one reading per token as possible, some final steps are employed to try and eliminate any remaining ambiguity. The simplest of these is that when a token has two readings that have the same POS tag – but different meanings in English, hence two readings – then the POS tag that the readings share is selected. For example, the Welsh word 'ceisio' and its soft mutated form 'geisio' both produce two readings, corresponding to their two English meanings in the lexicon extracted from Eurfa ('try' and 'seek'):

"<ceisio>"
"ceisio" {178,17} [cy] B e :try:
"ceisio" {178,17} [cy] B e :seek:

However, as both of these readings have the same POS tag – 'Be', an infinitive verb (also known in Welsh as a 'verb noun') – we can safely assign this tag to the word token in the running text.
A similar process is used when a word token is encountered that has been pruned to two readings of proper noun, but it has not been possible to discern the gender of the token. For example, when producing the initial cohorts of readings, any word that is not in our lexicon but begins with a capital letter – such as 'Eleanor' – might be assumed to be a proper noun. Because we cannot deduce the gender of the proper noun at this stage, a cohort of two readings ('Epg' for a masculine and 'Epb' for a feminine proper noun) is produced:

"<Eleanor>"
"Eleanor" {16,3} [cy] E p g :Eleanor:
"Eleanor" {16,3} [cy] E p b :Eleanor:

Word tokens such as these are searched for in a small collection of *gazetteers* – lists of stand-alone terms which have been leveraged from linked open data by running simple SPARQL queries against DBpedia[13]. These gazetteers contain lists of:

- Masculine given names,
- Feminine given names,
- Surnames,
- Place names.

If the word token can be found in either of the masculine or feminine given name gazetteers, then the appropriate tag ('Epg' for masculine or 'Epb' for feminine) can be selected. If the word token is found in both given name gazetteers, in the surnames gazetteer, or in the place names gazetter, then the morphological information about gender is discarded and the token is tagged as a 'neutral' proper noun ('Ep').

[13] See: `http://wiki.dbpedia.org/` or `http://dbpedia.org/sparql` for the SPARQL endpoint

Or, if the word token is not found in any of the gazetteers, then the token is tagged as a 'neutral' proper noun.
The most complex post-CG disambiguation steps involve the querying of two bespoke dictionaries, which are produced based on the tags found in a 611 sentence gold standard evaluation corpus (see Section 4.). These dictionaries comprise a *tag-token coverage dictionary*, and a *tag-sequence dictionary*.

Tag-Token Coverage Dictionary: This dictionary is created by taking each individual word token in the input (611 sentence gold-standard) corpus, and counting the number of times that token is assigned each CorCenCC POS tag. The final dictionary contains the most commonly-assigned tag for each unique word token. For example, the word 'yn' – which as we know from the first example in section 3.2.2. can either be a predicative particle ('Utra'), a verbal particle ('Uberf'), or a preposition ('Arsym') corresponding to the English word 'in' – is most commonly tagged as a preposition in the input corpus, and thus represented in the tag-token coverage dictionary as:

{"yn": "Arsym"}

If we encounter a word token that is still ambiguous at this point, we can check whether it is present in the tag-token coverage dictionary, and if it is we can then assign it the POS tag that would most commonly be given to that token.

Tag-Sequence Dictionary: This dictionary is created by cycling through every 3 token n-gram in a given sentence in the input (611 sentence gold-standard) corpus, and recording the POS tags of the tokens either side of the middle token, which we replace with the word 'find'. The 3-gram of POS tags either side of the word 'find' is then stored in the dictionary with the the word that 'find' replaced. For example, upon finding a 3-gram in the corpus with POS tags of 'Bpres3u' (present tense verb, 3rd person singular), 'Egu' (masculine singular noun), and 'Utra' (predicative particle) – a common 3-gram combination which would be produced by a simple phrase such as 'mae [noun] yn...' or '[noun] is' in English – the following entry would be added to the tag-sequence dictionary:

{"['Bpres3u', 'find', 'Utra']": "Egu"}

If we find that a word token is still ambiguous after all of the preceding disambiguation steps, we can now query the dictionary to see if it contains a 3-gram of 'find' surrounded by the POS tags of the n^{-1} and n^{1} word tokens. Continuing with the example 3-gram above, if the ambiguous word token was preceded by 'Bpres3u' and followed by 'Utra', then we could consider 'Egu' as the POS tag to assign to the ambiguous token.

4. Evaluation

We have evaluated the performance of *CyTag* using a 611 sentence (14,876 token) gold standard evaluation corpus that has being constructed as part of the ongoing work on the *CorCenCC* project. The corpus is comprised of eight example input files (included with the *CyTag* software) containing excerpts from a variety of existing Welsh corpora – *Kynulliad3*[14] (Welsh Assembly proceedings), *Meddalwedd*[15] (translations of software instructions), *Kwici*[16] (Welsh Wikipedia articles), and *LER-BIML*[17] (multi-domain spoken corpora) – and from the short abstracts of three additional Welsh Wikipedia articles. The 611 sentences were first tagged using *CyTag*, and then each token in the resulting output was manually checked by a Welsh language speaker. If an incorrectly tagged token was found, the correct POS (in line with the *CorCenCC POS Tagset* outlined in Section 3.1.1.) was noted down instead.

	No. of tokens
Total	14,876
Pre-CG:	
– with only one reading	8,917
– with multiple readings	5,198
– with no readings	761
— assumed to be proper nouns	504
Post-CG:	
– disambiguated	14,403
— pruned to one reading by CG	13,461
— two readings with same POS	65
— ambiguous gender proper nouns	504
— found in gazetteer	1
— tag from coverage dictionary given	372
– still ambiguous	216
– unknown	257

Table 3: Token counts at various stages of the *CyTag* process.

Table 3 shows how many tokens have been disambiguated at different stages of the *CyTag* process. From a total of 14,876 tokens in the 611 sentence input corpus, a total of 14,115 tokens have been assigned readings from Eurfa prior to CG being run. Of these, 8,916 have been assigned a single reading, with 5,198 having been assigned multiple readings that will need to be disambiguated. Of the 761 tokens that were not assigned a reading, 504 tokens have been assumed to be proper nouns (due to their capitalisation) and will have been automatically assigned two readings each – masculine proper noun ('Epg') and feminine proper noun ('Epb') – leaving 257 tokens unknown. After CG has been run, 14,403 tokens have been disambiguated (pruned down to one token), leaving the 257 tokens that were unknown prior to CG being run, and 216 tokens that are still ambiguous – CG and our various post-CG disambiguation steps were unable to prune these tokens down to a single read-

[14] http://cymraeg.org.uk/kynulliad3
[15] http://techiaith.cymru/corpws/Moses/Meddalwedd
[16] http://cymraeg.org.uk/kwici
[17] http://www.lancaster.ac.uk/fass/projects/biml

ing. Of the 14,403 tokens that were disambiguated, 13,361 of these were pruned by the CG rules, 65 of them were tokens that had been assigned two readings pre-CG but with the same POS tag (on account of two different meanings in English), 504 of them were proper nouns with ambiguous gender, 372 of them were assigned the most likely tag based on their presence in the coverage dictionary (see Section 3.2.3.), and 1 token was ambiguous, but found in the *CorCenCC* gazetteers described in Section 3.2.3..

	POS Type	
	Basic tags	**Enriched tags**
Tokens	14,876	
Tagged	14,403	
Still ambiguous	216	
Unknown	257	
Tagged correctly	13,866	13,488
Precision	96.27	93.64
Recall	96.61	96.52
F1	96.44	95.06

Table 4: Results of running *CyTag* over the 611 sentence corpus, taking into account performance over both the basic and enriched sections of the *CorCenCC POS Tagset*.

Table 4 shows the results of running the 611 sentences through *CyTag*, taking into account the difference in performance if we consider the full, enriched section of the *CorCenCC POS Tagset* or only the basic categories section into which the enriched tags collapse. The table demonstrates that from a total of 14,876 tokens, *CyTag* was able to assign a POS tag to 14,403 of them, with 216 tokens still being ambiguous post-CG and 257 tokens left unknown. Comparing the output of *CyTag* to the same 611 manually checked sentences from the gold standard evaluation corpus, we can see that 13,866 tokens have been assigned the correct tag from the basic POS categories, while 13,488 tokens have been assigned the correct tag from the enriched POS categories – this results in precision, recall and F1 values of 96.27, 96.61 and 96.44 over the basic POS categories and 93.64, 96.52 and 95.06 over the enriched POS categories, respectively.

	POS Type	
	Basic tags	**Enriched tags**
Tokens (multiple readings)	5,198	
Tagged correctly	4,885	4,800
Precision	93.70	91.99
Recall	95.31	95.23
F1	94.50	93.58

Table 5: Results of running *CyTag* over the 611 sentence corpus, considering its performance over only those tokens that had multiple readings pre-CG.

Table 5 shows the results of running the 611 sentences through *CyTag* when only those tags that were assigned multiple readings prior to CG. Comparing the output of *CyTag* to the same 611 manually checked sentences from the gold standard evaluation corpus, we can see from the table that from a total of 5,198 tokens *CyTag* was able to assign the correct basic POS tag to 4,885 tokens, and the correct enriched POS tag to 4,800 tokens. This results in precision, recall and F1 values of 93.70, 95.31 and 94.50 over the basic POS categories and 91.99, 95.23 and 93.58 over the enriched POS categories, respectively.

Finally, Table 6 shows how successfully each component of *CyTag* was able to assign POS tags, considering how accurately readings were pruned to one by CG and how accurate each of the post-CG disambiguation steps described in Section 3.2.3. proved to be. As the table demonstrates, CG is able to prune correctly prune readings for a given token to just one with a high degree of accuracy (97.33% for basic POS tags and 94.88% for enriched POS tags), and our disambiguation step of stripping one of the readings out when a token has two readings with the same tag (on account of the token having different meanings in English represented in Eurfa) is also highly accurate. Only one ambiguous token was found in the *CorCenCC* gazetteers, and so although it was correctly tagged we cannot discern whether other words might be tagged erroneously after being found in the gazetteers. We have had reasonable success in assuming that capitalised words that were unknown prior to running CG were proper nouns – 80.56% of these assumptions turned out to be correct, although the accuracy for these tokens using the enriched tagset is rather lower (71.23%) due to the fact that in many of these cases it was simply not possible to discern the gender of the proper noun. However, the accuracy of our tagging of ambiguous tokens based on their presence in the coverage dictionary (see Section 3.2.3.) is lower at 41.13%.

	POS Type	
	Basic tags	**Enriched tags**
Pruned to one by CG	13,101 (97.33%)	12,772 (94.88%)
Two readings, same tag	65 (100%)	64 (98.46%)
Proper noun assigned	406 (80.56%)	359 (71.23%)
Found in gazetteer	1 (100%)	1 (100%)
Found in coverage dict.	153 (41.13%)	153 (41.13%)

Table 6: Success rate of pruning readings to one using CG, and the various post-CG disambiguation methods also employed.

5. Discussion

The results of running the 611 sentences through *CyTag* are very positive – F1 scores above 95% represent a marked improvement over *WNLT* and are approaching the reported ac-

curacy of the *Bangor Autoglosser*. The strong performance of *CyTag* is also notable in the context of our gold standard corpus and the diverse range of sources from which the 611 sentences are extracted – Giesbrecht and Evert (2009) observe that reported POS tagging accuracies in the high 90%s usually come from focused evaluations on refined or edited texts with few errors or non-standard forms, while true accuracies over unseen texts from diverse sources are more likely to fall to the low 90%s or even the high 80%s. We can therefore be confident that *CyTag*'s performance would be consistently high across many different domains and sources – vital in the context of tagging the balanced, representative National Corpus of Contemporary Welsh for the *CorCenCC* project.

There are a number of particularly encouraging observations that can be made about the results presented in Section 4.. As Table 5 demonstrates, *CyTag* also performs well – with F1 scores approaching 95% – on only those tokens which had more than one reading prior to CG being run. Thus, while it is true that a large number of tokens are disambiguated by default as a result of only having one possible reading to begin with, the overall results are not necessarily skewed by this, and our CG-formatted rules are able to prune the more ambiguous tokens with good effect. This is highlighted by the breakdown of how accurate various disambiguation steps are, as demonstrated in Table 6: the accuracy with which we can prune the readings for a token down to one using CG is clearly highlighted, and this is supported by the accuracy with which certain assumptions – that unknown, capitalised words are probably proper nouns, and that two readings with the same tag can be cut down to one – can be applied.

Also noteworthy is the fact that in all of our evaluations, the results for enriched POS tags are very close to those obtained when considering only the basic POS tag categories, despite there being 145 enriched tags compared to only 13 basic tags, and thus much more room for error with the enriched set. Were it the case that our results were far better over the basic POS tagset than over the enriched tagset, we could conclude that the tagger was able to determine the major categories of words, but was having more trouble identifying morphological features (such as noun gender or number) and exceptions. Thankfully, this is not the case, and it's much more likely that only a small-number of cross-category discrepancies need to resolved in order to yield improved results. For example, we notice that *CyTag* can have trouble determining which POS tag to assign to the versatile token 'yn', with a number of instances where it has been tagged as a predicative particle ('Utra') where it should have been an uninflected preposition ('Arsym'), or vice-versa.

Moving forwards, implementing new rules to address these kinds of cross-category discrepancies should be more straightforward than if we were required to try and address inter-category discrepancies such as ambiguous noun genders or verb tenses, and enable us to prune readings for a given token with increased accuracy. This would mean less tokens remaining – or less readings for a remaining token – at the post-CG disambiguation stage, where again there is room for further development. In particular, our coverage dictionary could be redesigned or improved, any additional techniques for discerning the gender of proper nouns explored, and new disambiguation steps explored and developed, in order to handle more of those tokens that cannot be pruned by CG-formatted rules alone and to further increase the standard to which we can POS tag Welsh sentences in context.

6. Conclusions

We have described *CyTag*, a rule-based tagger for Welsh that leverages lexical information from an open source dictionary and uses Constraint Grammar to select the most appropriate POS tags for words in context. The high precision and excellent recall of the tagger – as demonstrated by our evaluation over a gold standard dataset of 611 manually-checked Welsh sentences – are very promising for Welsh and are in line with the accuracy expected of POS taggers over unseen text in other languages. As well as a high-performing open-source POS tagger for Welsh, our work demonstrates that by leveraging existing knowledge and resources and with minimal, easily-adaptable rules, accurate taggers can be developed even for languages for whom pre-annotated training data is scarce.

In future work, we intend to build additional layers of auxiliary tagging into *CyTag* to account for cases of function being different to pure form – for example, nouns being used as adjectives or plural forms being used as honorific singular forms (such as 'chi', a third-person plural pronoun which is often used to address people respectfully in the singular). We will also focus on the recognition and tagging of multi-word expressions (MWEs), with Welsh having a number of word and token combinations that make little sense outside of the multi-word context (such as 'ar agor', which can be treated as a single adjective equivalent to 'open' in English). Finally, in the context of the *CorCenCC* project, we will be using *CyTag* as the foundation of an extended pipeline incorporating a Welsh adaptation of the *UCREL Semantic Analysis System (USAS)*[18](Piao et al., 2018), in order to assign both syntactic and semantic tags to the National Corpus of Contemporary Welsh.

Acknowledgements

This work has been funded by the UK Economic and Social Research Council (ESRC) and Arts and Humanities Research Council (AHRC) as part of the *Corpws Cenedlaethol Cymraeg Cyfoes (The National Corpus of Contemporary Welsh): A community driven approach to linguistic corpus construction* project (Grant Number ES/M011348/1).

References

Brill, E. (1992). A Simple Rule-based Part of Speech Tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing (ANLC '92)*, pages 152–155. Association for Computational Linguistics.

Donnelly, K. and Deuchar, M. (2011). Using constraint grammar in the Bangor Autoglosser to disambiguate multilingual spoken text. In *Constraint Grammar Applications: Proceedings of the NODALIDA 2011 Workshop*.

[18]http://ucrel.lancs.ac.uk/usas/

Giesbrecht, E. and Evert, S. (2009). Part-of-speech tagging – a solved task? An evaluation of POS taggers for the Web as corpus. In *Proceedings of the 5th Web as Corpus Workshop (WAC5)*.

Karlsson, F., Voutilainen, A., Heikkilä, J., and Anttila, A. (1995). *Constraint Grammar: A Language-Independent Framework for Parsing Unrestricted Text*. Mouton de Gruyter, Berlin / New York.

Karlsson, F. (1990). Constraint Grammar as a Framework for Parsing Running Text. In *Proceedings of the 13th Conference on Computational Linguistics (COLING '90)*. Association for Computational Linguistics.

Piao, S., Rayson, P., Knight, D., and Watkins, G. (2018). Towards a Welsh Semantic Annotation System. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference (LREC 2018)*.

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49.

Toutanova, K., Klein, D., Manning, C., and Singer, Y. (2003). Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, pages 252–259. Association for Computational Linguistics.

Graph Based Semi-Supervised Learning for Tamil POS Tagging

Mokanarangan Thayaparan, Surangika Ranathunga, Uthayasanker Thayasivam
Dept. of Computer Science and Engineering
University of Moratuwa, Katubedda 10400, Sri Lanka
{mokanarangan.11, surangika, rtuthaya }@cse.mrt.ac.lk

Abstract

Parts of Speech (POS) tagging is an important pre-requisite for various Natural Language Processing tasks. POS tagging is rather challenging for morphologically rich languages such as Tamil. Being low-resourced, Tamil does not have a large POS annotated corpus to build good quality POS taggers using supervised machine learning techniques. In order to gain the maximum out of the existing Tamil POS tagged corpora, we have developed a graph-based semi-supervised learning approach to classify unlabelled data by exploiting a small sized POS labelled data set. In this approach, both labelled and unlabelled data are converted to vectors using word embeddings and a weighted graph is constructed using Mahalanobis distance. Then semi-supervised learning (SSL) algorithms are used to classify the unlabelled data. We were able to gain an accuracy of 0.8743 over an accuracy of 0.7333 produced by a CRF tagger for the same limited size corpus.

Keywords: Semi-Supervised Learning, Low-resourced languages, Graph-based SSL, Word Embedding, POS tagging

1. Introduction

In the recent past, supervised learning methods have produced high accuracies for Parts-of-Speech (POS) tagging (Gimenez and Marquez, 2004). In particular, sequence models such as hidden Markov models (HMM) and conditional random fields (CRF) have given good results (Huang et al., 2015). However, these techniques rely on the availability of relatively large amounts of annotated data. Hence, building an accurate domain insensitive POS tagger is challenging for low resourced languages.

Tamil is one such low resourced language, which is widely used in South India and Sri Lanka. There have been several POS taggers developed for Tamil language using supervised learning techniques (Dhanalakshmi et al., 2009)(Pandian and Geetha, 2009). Since the annotated corpora used in this research have been of small size and from a single domain, these supervised techniques greatly suffer from accuracy and domain adaptability (Rani et al., 2016). For example, FIRE corpus (Forum for Information Retrieval Evaluation, 2014), a widely used freely available Tamil POS annotated corpus contains only 80k words. In contrast, the Wall Street corpus, which is an English POS-annotated corpus has a word count of 1,173K words (Gimenez and Marquez, 2004), meaning that the size of the FIRE corpus is approximately 15 times smaller than the Wall Street corpus. Thus, when using a small corpus such as FIRE, we cannot expect similar accuracy to that of English when supervised techniques are used. Moreover, these approaches depend on language dependent features such as morphological tags (Dhanalakshmi et al., 2009) thus limiting the scalability for adapting to other low resourced languages.

In contrast to supervised approaches, semi-supervised approaches such as graph based semi-supervised learning and manifold regularization (Niyogi, 2013) use both labeled and unlabelled data for their classification, and have proven to work with a small data sets (Zhu et al., 2003). Despite having smaller sized POS-tagged data for Tamil, there has been only two research leveraging the opportunity presented by semi-supervised learning. Ganesh et al. (2014) have used segmentation patterns to implement a bootstrapping approach for POS tagging. This approach relies on language dependent data such as suffix context patterns. Rani et al. (2016) use small annotated training data to build a classifier model using context-based association rule mining. This approach neither includes any language-specific linguistic information nor requires a large corpus. However, they collect all possible words occurring in the same context from the untagged data into a list called context-based list, thus limiting it from scaling to large monolingual corpus.

Graph based semi-supervised learning (SSL) has gained traction in Natural Language Processing tasks such as question answering (Celikyilmaz et al., 2009), structural tagging (Subramanya et al., 2010), and speech language recognition (Liu et al., 2016). Graph based SSL builds a meaningful graph using labelled and unlabelled instances. It then employs an SSL algorithm such as harmonic functions (Zhu et al., 2005) or label propagation (Zhu et al., 2003) to label the unlabelled instances. Graph based SSL is easily parallelizable and scalable to large data (Zhu et al., 2005).

In this paper, we present a novel graph-based semi-supervised approach to produce an accurate POS tagger for Tamil using a limited size corpus. Our idea is inspired by Talukdar and Pereira (2010)'s case study on modified absorption, which is a label propagation algorithm. They have implemented a Named Entity recognizer by building a connected word graph. Similarity between words is measured using WordNet. Then they employ label propagation to assign labels to all the unlabelled nodes.

Since Tamil is a low resourced language with no proper WordNet, we built a connected word graph using word vectors and employed label propagation. Our method is based on the clustering hypothesis that relative distance of word vectors of similar categories is lower than those between different categories. We use neural word embedding (Word2Vec (Mikolov et al., 2013), FastText (Joulin et al., 2016)) to create word vectors. Mahalanobis distance is used for measuring the distance (metric learning)

between these vectors in order to construct the graph. Mahalanobis distance generalizes the standard Euclidean distance, and has proven to be more effective (Davis et al., 2007). We empirically tested with four different metric learning algorithms (Information Theoretic Metric Learning (ITML) (Davis et al., 2007), Sparse Determinant Metric Learning (SDML) (Qi et al., 2009), Least Squares Metric Learning (LSML) (Liu et al., 2012), and Local Fisher Discriminant Analysis (LFDA) (Sugiyama, 2006)) to calculate Mahalanobis distance. Once the graph is constructed with labeled and unlabeled nodes, to assign labels to unlabeled nodes, we experimented with three different SSL algorithms (LP-ZGL) (Zhu et al., 2003), Absorption (Talukdar et al., 2008) and Modified Absorption (MAD) (Talukdar and Pereira, 2010)). Local Fisher Discriminant Analysis (LFDA) metric learning coupled with Label Propagation(LP-ZGL) yielded a maximum accuracy of 0.8743 for the FIRE corpus against a baseline accuracy of 0.7338 achieved by using a traditional CRF model. Unlike supervised learning approaches, our approach does not require a large high quality annotated data set, or language dependent features.

Thus the contributions of this paper are: (1) converting words to vectors using neural word embedding and building meaningful word graphs, (2) using Mahalanobis distance to measure relationships between word vectors, hence measuring the correlation between variables, and (3) using a language independent graph based semi-supervised approach for POS tagging in Tamil.

The rest of the paper is organized as follows. Section 2 discusses graph based semi supervised learning techniques and previous attempts on Tamil POS tagging. Section 3 details the data set used in our experiment. Section 4 discusses the methodology and how we implemented the system. Section 5 details the experiments carried out and the relevant results. Section 6 and Section 7 document the conclusion and future work, respectively.

2. Related Work

2.1. Graph based Semi-supervised Learning

Graph theory and Natural Language Processing are well studied disciplines, but are commonly perceived as distinct with different algorithms and with different applications. But recent research has shown that these disciplines are connected and graph-theoretical approaches can be employed to find efficient solutions for NLP problems. Entities are connected by a range of relations in many NLP problems and graph is a natural way to capture the relationship between the entities. Graph based approaches have been used in word sense disambiguation, entity disambiguation, thesaurus construction, textual entailment and semantic classification (Mihalcea and Radev, 2011).

Graph based semi-supervised learning builds graphs connecting labeled and unlabeled data points, and perform classification by propagating the labels. The graph is constructed to reflect our prior knowledge about the domain. The intuition is that similar data points have similar labels. We let the hidden/observed labels be random variables on the nodes of this graph. Labels are injected to unlabeled nodes from labeled nodes. Graphs provide a uniform representation for heterogeneous data and are easily parallelizable (Zhu et al., 2005).

One of the challenges of graph based approach is building the graph that reflects the relationship between entities. Depending on the task, the nodes and edges may represent a variety of language related units and links. Different NLP tasks have approached this challenge in different ways. For the task of opinion summarization, Zhu et al. (2013) constructed a graph of sentences linked by edges whose weight combines the term similarity and objective orientation similarity. And to perform discourse analysis in chat, Elsner and Charniak (2010) predicted the probabilities for pair of utterance as belonging the same conversation thread or not based on lexical, timing and discourse-based features. Then constructed a graph with each nodes representing the utterances and the edges representing the probability score between the nodes. Although these approaches are evidences for the versatility of graph based approaches, these cannot be adopted to a word level problem like sequential tagging. Using graph methods for sequential tagging relies on the belief that similar words will have the same tag. Unlike the aforementioned approaches, here the nodes in these graph represents words or phrases and the the edges will indicate the similarity between nodes. Talukdar and Pereira (2010) tag words with NER information through a label propagation algorithm on a word similarity graph built using WordNet information. Words are represented are the graph vertices and the edge denotes the WordNet relationship. This approach cannot be adopted for a low resource language which doesn't have a proper WordNet.Subramanya et al. (2010) POS tags on a similarity graph where local sequence contexts (n-grams) are vertices. The similarity function between graphs is the cosine distance between the point-wise mutual information vectors (PMI) representing each node. The point-wise mutual information is calculated between n-gram and set of context features. These context features includes suffixes, left word and right word contexts. The challenge of this approach is the scalability for a morphologically complex language like Tamil.

2.2. Tamil POS tagging

Tamil is a low resourced, morphologically rich language with many inflections and a complex grammatical structure. Thus, automatic POS tagging for Tamil is a challenging task. Supervised learning approaches have been heavily undertaken in Tamil for POS tagging. These include CRF models using morphological information (Pandian and Geetha, 2009) and Support Vector Machines (SVM) using semantic information (Dhanalakshmi et al., 2009). These models had been trained using different corpora containing approximately 200k annotated words. These annotated corpora or taggers are not publicly available.

There have been very few attempts in using semi-supervised approaches for Tamil language to develop POS taggers. Ganesh et al. (2014) have used language features with a bootstrapping approach to obtain a precision of 86.74%. They have presented a pattern based bootstrapping approach using only a small set of POS labelled suffix context patterns. The patterns consist of a stem and a sequence

of suffixes, obtained by segmentation using a manually created suffix list. This bootstrapping technique generates new patterns by iteratively masking suffixes with low probability of occurrences in the suffix context, and replacing them with other co-occurring suffixes. This approach relies on language specific information.

Rani et al. (2016) have employed a semi-supervised rule mining approach using morphological features for Hindi, Tamil, and Telugu languages. They have used a combination of a small annotated and untagged training data to build a classifier model using a concept of context-based association rule mining. These association rules work as context-based tagging rules.

3. Data set

For our experiment, we used the FIRE Tamil Corpus. The FIRE Tamil corpus contains 80k POS tagged words with 21 different tags as shown in Table 1.

NN	Noun
NNC	Compound Noun
RB	Adverb
VM	Verb Main
SYM	Symbol
PRP	Personal Pronoun
JJ	Adjective
NNP	Pronoun
PSP	Prepositions
QC	Quantity Count
VAUX	Verb Auxiliary
DEM	Determiners
QF	Quantifiers
NEG	Negatives
QO	Quantity Order
WQ	Word Question
INTF	Intensifier
NNPC	Compound Pro Noun
CC	Coordinating Conjunction
RBP	Adverb Phrase

Table 1: POS tagsets for FIRE Tamil Corpus

4. Methodology

Our work is inspired by Talukdar and Pereira (2010)'s case study on the performance of different algorithms for classification in graphs. In this work, words are represented as nodes and the similarity between nodes are measured using WordNet distance. Since Tamil is a low resourced language, this approach was not viable for us. Another approach was to represent words by converting them to vectors and computing the similarity. Subramanya et al. (2010) had employed a point wise mutual information (PMI) based approach to convert the word to vectors and compute the similarity by measuring the cosine distance. His approach used hand-crafted features that will not work with same efficiency across different languages.

Hence, an efficient way of representing a word in the vector space has to be determined. In addition, it is required to identify mechanisms for (1) constructing a meaningful graph based on the word vector, and (2) classifying unlabelled words based on the constructed graph by measuring the similarity.

4.1. Representing a word in the vector space

We adopted the Word2Vec model proposed by Mikolov et al. (2013) and convert the word into the vector space to construct the graph. To the best of our knowledge, Word2Vec has never been used to construct weighted word graphs to be used in SSL. Similarly we also experimented with Fast Text skipgram (Bojanowski et al., 2016) and bag of words models (Joulin et al., 2016). The key difference between Word2Vec and FastText is that Word2Vec treats each word in corpus as an atomic entity and generates a vector for each word. In contrast, FastText treats each word as composed of ngrams and the vector word is made of the sum of these vectors.

4.2. Constructing a meaningful graph based on the word vector

Each word is converted to a d dimensional vector space. Out of the n words in the list, n_l are labelled($n >>> n_l$). We employ 32 different tags to denote each POS entity (Dhanalakshmi et al., 2009). $G = (V, E, W)$ is the graph we are interested in constructing; where V is the set of vertices with $|V| = n$, E is the set of edges. W is the symmetric $n \times n$ matrix of edge weights we want to learn. Usually we could choose a standard distance metric (Euclidean, City-Block, Cosine, etc.). Instead, Mahalanobis distance has proven to be effective with clustering problems over the standard metrics (De Maesschalck et al., 2000).

We use a supervised method for learning the Mahalanobis distance. For this purpose, we need to calculate the positive definite matrix A of size $d \times n$ that parametrizes the Mahalanobis distance, $d_A(x_i, x_j)$ (Dhillon et al., 2010; Davis et al., 2007; Sugiyama, 2006) between words x_i and x_j as shown in Equation (1).

$$d_A(x_i, x_j) = (x_i - xj)^T A(x_i - x_j) \tag{1}$$

Since A is positive definite, it can be decomposed into $P^T P$, where P is another matrix of size $d \times d$

$$\begin{aligned} d_A(x_i, x_j) &= (x_i - x_j)^T P^T P(x_i - x_j) \\ &= (Px_i - Px_j)^T (Px_i - Px_j) \\ &= d_I(Px_i, Px_j) \end{aligned} \tag{2}$$

There are many proposed methods for calculating the transformation matrix P. We empirically experimented with different metric learning algorithms, including Information Theoretic Metric Learning (ITML) (Davis et al., 2007), Sparse Determinant Metric Learning (SDML) (Qi et al., 2009), Least Squares Metric Learning (LSML) (Liu et al., 2012), and Local Fisher Discriminant Analysis (LFDA) (Sugiyama, 2006).Researches in link prediction in networks (Shaw et al., 2011), music recommendation (McFee et al., 2011) and bio metrics verification (Ben et al., 2012) has shown that metric learning plays a vital role increasing accuracy of the system.

ITML minimizes the differential entropy between multivariate Gaussian under constraints on the distance function. Davis et al. (2007) have expressed the problem as that of minimizing the LogDet divergence subject to linear constraints. SDML uses l_1-penalized log-determinant regularization to calculate the metric. This algorithm exploits the sparsity nature underlying the intrinsic high dimensional feature space. LSML uses an algorithm that minimizes a convex objective function corresponding to the sum of squared residuals of constraints. Finally LFDA, is a linear supervised dimensionality reduction method which is particularly useful when dealing with cases where one or more core classes consist of separate clusters in input space.

We calculate P using each of these metric learning algorithms and project the words into a new space to calculate Px_i. Based on Equation 2, we compute the Euclidean distance in the linearly transformed matrix. Gaussian kernel $[2, 16]$ was used to compute the similarity between words as shown in Equation 3 (Dhillon et al., 2010). We then sparsify the graph by selecting k neighbors for each node and set the weights to zero for all others (Zhu et al., 2003).

$$W_{ij} = exp(\frac{-d_A(x_i, x_j)}{2\sigma^2}) \quad (3)$$

The culmination of all these steps results in a meaningful graph where relative distances of word vectors of similar categories will be lower than those between different categories.

4.3. Classifying Unlabelled Nodes based on the Constructed Graph

Once the graph is constructed, unlabelled words in the graph should be classified. For this, we experimented with Label Propagation(LP-ZGL), and Absorption and Modified Absorption (MAD) techniques. LP-ZGL (Zhu et al., 2003) was one of the first graph based SSL methods. LP-ZGL propagates the labels over the graph by penalizing any label assignment where two nodes connected by a highly weighted edge are assigned different labels. LP-ZGL prefers smooth labeling over the graph. This property is also shared by the other two algorithms. Absorption (Talukdar et al., 2008) has been used for open domain class-instance acquisition. Absorption is an iterative algorithm where label estimates depend on the previous iteration. Modified Absorption (MAD) (Talukdar and Pereira, 2010) shares the same properties of the Absorption algorithm but can be expressed as an unconstrained optimization problem. We experimented with all these algorithms to estimate the labels of the untagged words.

5. Experiments and Results

5.1. Experiments

We split the data into 60k words for training and 20k words for testing. To the best of our knowledge, there has been only Named Entity Recognition research (Abinaya et al., 2014) done in Tamil using FIRE corpus and no POS tagging research done.

We trained both Word2Vec and FastText models with a word window of three (the commonly used window size) using the Tamil Wikipedia corpus (Wikipedia, 2016) (about 1M words) after removing only the punctuation marks. We used these models to convert word to vector form. Each vector is of 300 dimensions. For graph construction, a subset of 3000 sentences with approximately 50k unlabelled words from the Tamil Wikipedia corpus were added to the set. We constructed the word graphs using the aforementioned four metric learning approaches and employed three labeled propagation approaches to identify the best combination.

Since most of the successful approaches related to Tamil POS tagging have been carried out using Conditional Random Fields (CRF) (Pandian and Geetha, 2009), we used the same approach with word trigram feature as our baseline method. Here, trigrams were selected because for Word2Vec and FastText models also, a word window of three was used.

5.2. Results

The following Tables 2-5 document the results obtained for each graph construction algorithm in combination with the classification methods.

Word To Vector Algorithm	MAD	Abs	LP-ZGL
Word2Vec (SkipGram)	0.7534	0.7531	0.7201
Word2Vec (Bag of words)	0.6945	0.6967	0.6754
Fasttext (SkipGram)	0.8146	0.814	0.822
Fasttext (Bag of Words)	0.795	0.7952	0.801

Table 2: Accuracy of Information Theoretic Metric Learning

Word To Vector Algorithm	MAD	Abs	LP-ZGL
Word2Vec (SkipGram)	0.7012	0.701	0.721
Word2Vec (Bag of words)	0.6641	0.6542	0.665
Fasttext (SkipGram)	0.7886	0.7935	0.7988
Fasttext (Bag of Words)	0.7712	0.775	0.7767

Table 3: Accuracy of Sparse Determinant Metric Learning

Word To Vector Algorithm	MAD	Abs	LP-ZGL
Word2Vec (SkipGram)	0.734	0.733	0.732
Word2Vec (Bag of words)	0.701	0.71	0.711
Fasttext (SkipGram)	0.8547	0.861	0.8634
Fasttext (Bag of Words)	0.823	0.834	0.845

Table 4: Accuracy of Least Squares Metric Learning

Word To Vector Algorithm	MAD	Abs	LP-ZGL
Word2Vec (SkipGram)	0.7678	0.7775	0.7757
Word2Vec (Bag of words)	0.7664	0.7567	0.7456
Fasttext (SkipGram)	0.8673	0.8573	**0.8743**
Fasttext (Bag of Words)	0.85	0.853	0.86

Table 5: Accuracy of Local Fisher Discriminant Analysis

As illustrated above, Local Fisher Discriminant Analysis(LFDA) combined with Label propagation yields the best accuracy of 0.8743. LFDA is a linear supervised dimensionality reduction method. It proved effective in our case since each of our words had a size of 300 dimensions. FastText(skipgram) in combination with label propagation consistently performed better than other algorithms in all graph construction methodologies.

To test the robustness of the approach, we trained the best performing combination (LFDA and LP-ZGL) with 20k words and tested with 60k words. It yielded an accuracy of 0.753. Meanwhile, the baseline CRF model only gave an accuracy score of 0.633. This proves that our approach is more robust even when the labelled data set is comparatively small.

6. Conclusion

Our research establishes the fact that graph based semi-supervised approaches are more robust than supervised classification algorithms for POS tagging when the data set is relatively small. Thus graph based semi supervised data can be employed in the early stages of creating POS tagged data sets. Human annotators can correct the automatically annotated corpus with less effort, and the corrected annotated data set can be used in an iterative manner to re-train the tagger. Thus, graph based semi-supervised approaches are particularly useful for POS tagging of low-resourced languages such as Tamil. We used neural word embedding to create a vector representation of words, and Mahanalobis distance to measure distance between word vectors in order to build the graph. This shows that word embedding provides an excellent alternative for WordNet in measuring similarity between words, especially for languages that do not have a WordNet. This is useful not only for graph building, but for any task that requires measuring the similarity of words.

7. Future work

Our language independent work has shown promise with low resources. We have only done the research for one language, and this research should be extended to other languages to verify the general applicability of the presented methodology. We hope to extend this idea for other low resourced sequential tagging problems such as Named Entity Recognition. This research can also be extended to improve and incorporate other word embedding techniques such as VarEmbed that uses morphological priors for probabilistic neural word embedding (Bhatia et al., 2016). We can also experiment with other graph construction algorithms such as b-matching (Jebara et al., 2009). The main limitation of this technique is the amount of time taken to build the graph. Thus we intend to look into different code optimization methods. While we have compared our approach with the pure CRF implementation, Lample et al. (2016) has shown that CRF in combination with LSTM can provide a higher accuracy for Named entity recognition but that approach has not been tried for POS tagging in morphologically complex languages such as Tamil. We are eager to see how our approach stacks up with them.

8. Acknowledgement

This research is partially funded by the National Languages Processing Centre of University of Moratuwa, Sri Lanka. The authors would also like to thank the LKDomain registry for partially funding this publication.

9. Bibliographical References

Abinaya, N., John, N., Ganesh, B. H., Kumar, A. M., and Soman, K. (2014). Amrita_cen@ fire-2014: Named entity recognition for indian languages using rich features. In *Proceedings of the Forum for Information Retrieval Evaluation*, pages 103–111. ACM.

Ben, X., Meng, W., Yan, R., and Wang, K. (2012). An improved biometrics technique based on metric learning approach. *Neurocomputing*, 97:44 – 51.

Bhatia, P., Guthrie, R., and Eisenstein, J. (2016). Morphological priors for probabilistic neural word embeddings. 3 August.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Celikyilmaz, A., Thint, M., and Huang, Z. (2009). A graph-based semi-supervised learning for question-answering. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 719–727, Stroudsburg, PA, USA. Association for Computational Linguistics.

Davis, J. V., Kulis, B., Jain, P., Sra, S., and Dhillon, I. S. (2007). Information-theoretic metric learning. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 209–216, New York, NY, USA. ACM.

De Maesschalck, R., Jouan-Rimbaud, D., and Massart, D. L. (2000). The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18.

Dhanalakshmi, V., Rajendran, S., Soman, K. P., and Edu, K. (2009). POS tagger and chunker for tamil language.

Dhillon, P. S., Talukdar, P. P., and Crammer, K. (2010). Inference driven metric learning (idml) for graph construction.

Elsner, M. and Charniak, E. (2010). Disentangling chat. *Comput. Linguist.*, 36(3):389–409, September.

Ganesh, J., Parthasarathi, R., Geetha, T. V., and Balaji, J. (2014). Pattern based bootstrapping technique for tamil POS tagging. In *Mining Intelligence and Knowledge Exploration*, Lecture Notes in Computer Science, pages 256–267. Springer, Cham.

Gimenez, J. and Marquez, L. (2004). Svmtool: A general pos tagger generator based on support vector machines. In *In Proceedings of the 4th International Conference on Language Resources and Evaluation*.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Jebara, T., Wang, J., and Chang, S.-F. (2009). Graph construction and b-matching for semi-supervised learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 441–448, New York, NY, USA. ACM.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June. Association for Computational Linguistics.

Liu, E. Y., Guo, Z., Zhang, X., Jojic, V., and Wang, W. (2012). Metric learning from relative comparisons by minimizing squared residual. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, ICDM '12, pages 978–983, Washington, DC, USA. IEEE Computer Society.

Liu, Y., Kirchhoff, K., Liu, Y., and Kirchhoff, K. (2016). Graph-based semisupervised learning for acoustic modeling in automatic speech recognition. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 24(11):1946–1956, November.

McFee, B., Barrington, L., and Lanckriet, G. R. G. (2011). Learning content similarity for music recommendation. *CoRR*, abs/1105.2344.

Mihalcea, R. F. and Radev, D. R. (2011). *Graph-based Natural Language Processing and Information Retrieval*. Cambridge University Press, New York, NY, USA, 1st edition.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. 16 January.

Niyogi, P. (2013). Manifold regularization and semi-supervised learning: Some theoretical analyses. *Journal of Machine Learning Research*, 14:1229–1250.

Pandian, S. L. and Geetha, T. V. (2009). Crf models for tamil part of speech tagging and chunking. In *Proceedings of the 22Nd International Conference on Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy*, ICCPOL '09, pages 11–22, Berlin, Heidelberg. Springer-Verlag.

Qi, G.-J., Tang, J., Zha, Z.-J., Chua, T.-S., and Zhang, H.-J. (2009). An efficient sparse metric learning in high-dimensional space via l1-penalized log-determinant regularization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 841–848, New York, NY, USA. ACM.

Rani, P., Pudi, V., and Sharma, D. M. (2016). A semi-supervised associative classification method for POS tagging. *Int J Data Sci Anal*, 1(2):123–136, 1 July.

Shaw, B., Huang, B., and Jebara, T. (2011). Learning a distance metric from a network. In J. Shawe-Taylor, et al., editors, *Advances in Neural Information Processing Systems 24*, pages 1899–1907. Curran Associates, Inc.

Subramanya, A., Petrov, S., and Pereira, F. (2010). Efficient graph-based semi-supervised learning of structured tagging models. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 167–176, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sugiyama, M. (2006). Local fisher discriminant analysis for supervised dimensionality reduction. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 905–912, New York, NY, USA. ACM.

Talukdar, P. P. and Pereira, F. (2010). Experiments in graph-based semi-supervised learning methods for class-instance acquisition. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1473–1481, Stroudsburg, PA, USA. Association for Computational Linguistics.

Talukdar, P. P., Reisinger, J., Paşca, M., Ravichandran, D., Bhagat, R., and Pereira, F. (2008). Weakly-supervised acquisition of labeled class instances using graph random walks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 582–590, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zhu, X., Ghahramani, Z., and Lafferty, J. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pages 912–919. AAAI Press.

Zhu, X., Lafferty, J., and Rosenfeld, R. (2005). *Semi-supervised learning with graphs*. Ph.D. thesis, Carnegie Mellon University, language technologies institute, school of computer science.

Zhu, L., Gao, S., Pan, S. J., Li, H., Deng, D., and Shahabi, C. (2013). Graph-based informative-sentence selection for opinion summarization. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, pages 408–412, New York, NY, USA. ACM.

10. Language Resource References

Forum for Information Retrieval Evaluation. (2014). *FIRE Corpus*. Indian Institute of Science, Bangalore.

Wikipedia. (2016). *Tamil Wikipedia Corpus*. Wikipedia.

What Causes the Differences in Communication Styles? A Multicultural Study on Directness and Elaborateness

Juliana Miehle[1], **Wolfgang Minker**[1], **Stefan Ultes**[2]
[1]Institute of Communications Engineering, Ulm University, Albert-Einstein-Allee 43, 89081 Ulm, Germany
[2]Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK
juliana.miehle@uni-ulm.de, wolfgang.minker@uni-ulm.de, su259@cam.ac.uk

Abstract

With the aim of designing a Spoken Dialogue System which adapts to the user's communication idiosyncrasies, we present a multicultural study to investigate the causes of differences in the communication styles *elaborateness* and *directness* in Human-Computer Interaction. By adapting the system's behaviour to the user, the conversation agent may appear more familiar and trustworthy. 339 persons from Germany, Russia, Poland, Spain and the United Kingdom participated in this web-based study. The participants had to imagine that they are talking to a digital agent. For every dialogue turn, they had to read four different variants of the system output and indicate their preference. With the results of this study, we could demonstrate the influence of the user's culture and gender, the frequency of use of speech based assistants as well as the system's role on the user's preference concerning the system's communication style in terms of its *elaborateness* and its *directness*.

Keywords: Spoken Dialogue System, User Adaptation, Communication Idiosyncrasies

1. Introduction

For humans, speech is the most intuitive and most natural way to communicate. Therefore, scientists and engineers aim to realise methods and systems that enable not only interpersonal communication but also interaction with machines through natural spoken language. Today, we are able to communicate with various computer applications via speech. However, the usability and acceptance of Spoken Dialogue Systems is rather low and in public opinion, such systems often fall into disrepute (Hempel, 2008).

For Human-Human Interaction, it has been shown that people adapt their interaction styles to one another across many levels of utterance production when they communicate, e.g. by matching each other's behaviour or synchronising the timing of behaviour (Burgoon et al., 2007; Niederhoffer and Pennebaker, 2002; Brennan, 1996; Pickering and Garrod, 2004; Nenkova et al., 2008). Moreover, various studies suggest to adapt Spoken Dialogue Systems to the user in a similar way (Cassell and Bickmore, 2003; Forbes-Riley et al., 2008; Stenchikova and Stent, 2007; Reitter et al., 2006; Mairesse and Walker, 2010). By adapting the system's behaviour to the user, the conversation agent may appear more familiar and trustworthy and the dialogue may be more effective. Therefore, current research focuses on user-adaptive Spoken Dialogue Systems, e.g. (Honold et al., 2014; Ultes et al., 2015; Casanueva et al., 2015). Pragst et al. (2015) specifically focus on the adaptiveness of Dialogue Management to the cultural background and the emotional state of the user.

Our aim is to design a Spoken Dialogue System which adapts to the user's communication idiosyncrasies. According to various cultural models for Human-Human Interaction (Hofstede, 2009; Elliott et al., 2016; Kaplan, 1966; Lewis, 2010), different cultures prefer different communication styles. Moreover, Burleson (2003) presents a study of culture and gender differences in close relationships, emotion and interpersonal communication. Empirical research assessing gender, ethnic and cultural differences is reviewed. It is shown that social constructionist theories, like the different cultures view of gender, anticipate differences among social groups. These differences influence forms and functions of social relationships, the character of emotional experiences and the uses to which communication is put.

However, it is unclear which cultural idiosyncrasies found in Human-Human Interaction may be transferred to Human-Computer Interaction as it has been shown that there exist clear differences in Human-Human Interaction and Human-Computer Interaction (Doran et al., 2003). Miehle et al. (2016) showed that communication idiosyncrasies found in Human-Human Interaction may also be observed during Human-Computer Interaction in a Spoken Dialogue System context. Moreover, cultural differences between Germany and Japan have been identified. However, not all results are consistent with the existing cultural models for Human-Human Interaction and the authors infer that the communication patterns are not only influenced by the culture, but also by the dialogue domain and other user states and traits.

In order to obtain a more detailed view, the study described in the work at hand is composed broader and investigates more different cultures. In addition, five European cultures are examined whose communication styles are much more alike than the German and Japanese communication idiosyncrasies investigated by Miehle et al. (2016). Moreover, the aim of this work is to identify what causes the differences in communication styles except for the user's cultural background. Therefore, we explore not only the influence of the user's culture but also of the gender, the frequency of use of speech based assistants as well as the system's role. To investigate this, we designed and conducted a user study with 339 participants from Germany, Russia, Poland, Spain and the United Kingdom. For three different dialogues, the study participants had to indicate their preference concerning the system output in every dialogue turn. For the system's output we varied the two communication

styles *elaborateness* and *directness* as Pragst et al. (2017) shows that these dimensions influence the user's perception of a dialogue and are therefore valuable possibilities for adaptive Dialogue Management. With the findings of our study, we demonstrate which dimensions cause the differences in the communication styles *elaborateness* and *directness* in Human-Computer Interaction.
The structure of the paper is as follows: In Section 2., the experimental setting for the multicultural user study will be described. Afterwards, the evaluation results will be presented in Section 3., before concluding in Section 4.

2. Experimental Setting

We have designed a multicultural user study to investigate what causes the differences in the communication styles *elaborateness* and *directness* in Human-Computer Interaction. To do this, We created three dialogues where the digital agent assumed three different roles. For every dialogue step, we formulated four options of how the agent talks to the user:

- direct, elaborate (D, E)
- direct, concise (D, C)
- indirect, elaborate (I, E)
- indirect, concise (I, C)

As described by Pragst et al. (2017), *elaborateness* refers to the amount of additional information provided to the user and *directness* describes how concretely the information that is to be conveyed is addressed by the speaker. If the user, for example, asks the digital agent called Kristina whether she can tell him about today's weather, the four variations of the system output look as follows:

- It will be cloudy mostly and it might rain during the afternoon. (D, E)
- It will rain. (D, C)
- I would propose to take an umbrella. A scarf and gloves would be good as well. (I, E)
- You should take an umbrella. (I, C)

This example shows that in the concise version of a system utterance only the requested information is given to the user, while the elaborate version of the same utterance results in giving a more detailed weather forecast. Moreover, the direct option gives an accurate description of the weather, whereas the indirect approach to answering that question is the advise to take an umbrella. In this case, the weather is not stated directly but can be inferred from the given information.
The study has been conducted on-line. The participants had to imagine that they are talking to a digital assistant. They were shown the three dialogues, including the user input and different options for the system output. An exemplary dialogue turn (as it has been presented to the study participant) can be seen in the following:

YOU: Kristina, can you tell me about the weather today?
KRISTINA: I would propose to take an umbrella. A scarf and gloves would be good as well.
KRISTINA: It will be cloudy mostly and it might rain during the afternoon.
KRISTINA: You should take an umbrella.
KRISTINA: It will rain.

For each dialogue turn, the participants had to read the four different variants of the system output carefully and decide afterwards which one they prefer. All descriptions and all dialogues have been provided in the participants' mother tongues (German, English, Polish, Russian and Spanish). The translations have been made by native speakers who were instructed to be aware of the linguistic features and details of the differences to assure equivalence in the translations. Moreover, the quality of the translations has been assured by the use of backward translations.
In the following, we describe the three dialogues as well as the group of participants.

2.1. Description of the dialogues

The application of our digital agent Kristina is to help people in European countries get health-related information. For the user study, we have created three dialogues where Kristina assumed three different roles.
In the first dialogue, the agent took the role of a social companion. The dialogue is chat-oriented and Kristina and the user make small talk about the weather and the user's mood. The study participants had to imagine that they are talking to Kristina about the weather as they want to go swimming later on.
For the second dialogue, the participants had to put themselves in the shoes of a parent who asks Kristina for help to bathe their baby. Kristina assumed the role of an expert providing the user with facts and descriptions regarding baby care. In contrast to the first dialogue, this one is task-oriented with the purpose of information retrieval.
During the third dialogue, Kristina acted as a personal assistant. The users had to imagine that they are looking after Eugene who is in need of care and ask Kristina about his sleeping habits. Kristina retrieved useful information about the sleeping routine of the care recipient Eugene from personal profile data and provided it to the user.

2.2. The study participants

Altogether, 339 persons from Germany, Russia, Poland, Spain and the United Kingdom participated in the user study. They have been recruited and paid using the Clickworker Survey Service[1] where the target group can be defined according to demographic data. The participants were aged between 18 and 55 years, 166 of them have been male (48.97%), 173 have been female (51.03%). The participants' detailed gender and age distribution is shown in Table 1. It can be seen that the number of participants is evenly distributed among the five different cultures. Moreover, the average age of the ten different groups ranges between 25.44 and 30.75 years.

[1] www.clickworker.com

Culture	Gender	#Participants	Average age
German	male	32	30.75
	female	34	30.62
English	male	35	29.26
	female	35	30.09
Polish	male	34	27.09
	female	34	29.24
Russian	male	33	28.42
	female	35	28.83
Spanish	male	32	25.44
	female	35	28.94

Table 1: The participants' gender and age distribution.

The participants were asked how often they use a speech based assistant like Apple Siri, Google Assistant or Microsoft Cortana. The results are shown in Table 2. It can be seen that 35.69% stated that they use speech based assistants a couple of times a month, 27.73% indicated that they never use such systems, 24.78% use them several times a week and only 11.80% use them every day.

Usage	#Participants	%
every day	40	11.80
several times a week	84	24.78
a couple of times a month	121	35.69
never	94	27.73

Table 2: The participants' frequency of use of speech based assistants.

3. Evaluation results

In this section, the results of our user study are described. For the evaluation of the results, we do not take into account the first and the last dialogue turn of every dialogue as these system outputs have been used for greeting and leave-taking in order to have complete dialogues for the survey. This results in three dialogues, the first one contains four system utterances, the second one three utterances and the third one five utterances.

The overall evaluation results can be seen in Table 3. The first part ("All") as well as Figure 1 show the average across all twelve utterances and all ten user groups shown in Table 1. It can be seen that 50.64% selected the direct and elaborate (D, E) version of the system utterances, while 16.25% selected the direct and concise (D, C) version, 22.64% selected the indirect and elaborate (I, E) version and only 10.47% selected the indirect and concise (I, C) version. Isolating the two dimensions *directness* and *elaborateness*, we get the result that 66.89% selected the direct (D) version of the system utterances, while only 33.11% chose the indirect (I) version. Moreover, 73.28% selected the elaborate (E) and 26.72% the concise (C) version of the system utterances. This indicates that the participants tend to prefer the direct and elaborate variations. Furthermore, the *elaborateness* of the system has a greater influence on the user's preference than the *directness*.

Following the same approach, we investigated the influence of the system's role (second part in Table 3 containing the rows "Dialogue 1", "Dialogue 2" and "Dialogue 3") as well as the user's culture (third part in Table 3 containing the rows "German", "English", "Polish", "Russian" and "Spanish"), gender (fourth part in Table 3 containing the rows "male" and "female") and frequency of use of speech based assistants (last part in Table 3 containing the rows "every day", "several times a week", "a couple of times a month" and "never"). In the following, the detailed discussion of the results will be presented.

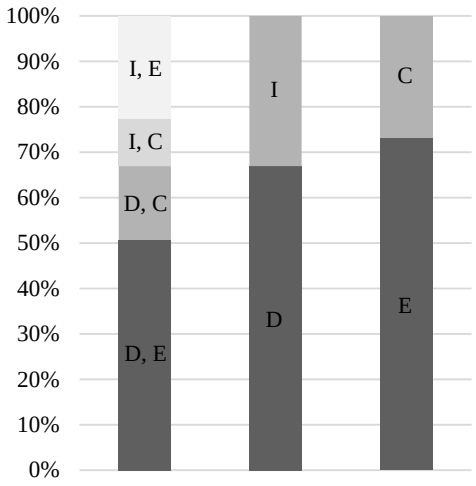


Figure 1: The overall evaluation results, averaging across all twelve utterances and all ten user groups.

3.1. Influence of the system's role

First of all, the influence of the system's role is examined. As described in Section 2.1., the digital agent called Kristina assumed three different roles during the dialogues. The results concerning the *elaborateness* are shown in Figure 2, those concerning the *directness* are depicted in Figure 3. In Dialogue 1, where the agent's role was the social companion, 72.86% chose the elaborate and 74.41% chose the direct versions of the system utterance. This leads to significantly more direct choices than the average over all three dialogues (66.89%). In Dialogue 2, where the agent acted as an expert, 86.14% chose the elaborate and 51.33% chose the direct versions of the system utterance. This means that the elaborate options have been chosen significantly more often than the average over all three dialogues (73.28%). Moreover, the direct options have been chosen significantly less often than the average over all three dialogues (66.89%). In Dialogue 3, where the agent assumed the role of an assistant, 65.90% chose the elaborate and 70.21% chose the direct versions of the system utterance. This means that the elaborate options have been preferred significantly less often than the average over all three dialogues (73.28%). Moreover, the direct options have been chosen significantly more often than the average over all three dialogues (66.89%).

We can conclude that the system's role significantly influences the user's preference in the system's communication style. The largest differences to the average over all three dialogues occur when the system acts as an expert (Dialogue 2). In this case, the elaborate and indirect options have been selected most often.

Group	Style	#Part.	%
All	direct, elaborate	2060	50.64
	direct, concise	661	16.25
	indirect, elaborate	921	22.64
	indirect, concise	426	10.47
Dialogue 1	direct, elaborate	797	58.78
	direct, concise	212	15.63
	indirect, elaborate	191	14.09
	indirect, concise	156	11.50
Dialogue 2	direct, elaborate	438	43.07
	direct, concise	84	8.26
	indirect, elaborate	438	43.07
	indirect, concise	57	5.60
Dialogue 3	direct, elaborate	825	48.67
	direct, concise	365	21.53
	indirect, elaborate	292	17.23
	indirect, concise	213	12.57
German	direct, elaborate	370	46.72
	direct, concise	148	18.69
	indirect, elaborate	165	20.83
	indirect, concise	109	13.76
English	direct, elaborate	421	50.12
	direct, concise	143	17.02
	indirect, elaborate	193	22.98
	indirect, concise	83	9.88
Polish	direct, elaborate	399	48.90
	direct, concise	158	19.36
	indirect, elaborate	179	21.94
	indirect, concise	80	9.80
Russian	direct, elaborate	420	51.47
	direct, concise	124	15.20
	indirect, elaborate	172	21.08
	indirect, concise	100	12.25
Spanish	direct, elaborate	450	55.97
	direct, concise	88	10.95
	indirect, elaborate	212	26.37
	indirect, concise	54	6.72
male	direct, elaborate	978	49.10
	direct, concise	342	17.17
	indirect, elaborate	450	22.59
	indirect, concise	222	11.14
female	direct, elaborate	1082	52.12
	direct, concise	319	15.37
	indirect, elaborate	471	22.69
	indirect, concise	204	9.83
every day	direct, elaborate	237	49.38
	direct, concise	83	17.29
	indirect, elaborate	111	23.13
	indirect, concise	49	10.21
several times a week	direct, elaborate	500	49.60
	direct, concise	168	16.67
	indirect, elaborate	231	22.92
	indirect, concise	109	10.81
a couple of times a month	direct, elaborate	715	49.24
	direct, concise	232	15.98
	indirect, elaborate	346	23.83
	indirect, concise	159	10.95
never	direct, elaborate	608	53.90
	direct, concise	178	15.78
	indirect, elaborate	233	20.66
	indirect, concise	109	9.66

Table 3: The overall evaluation results.

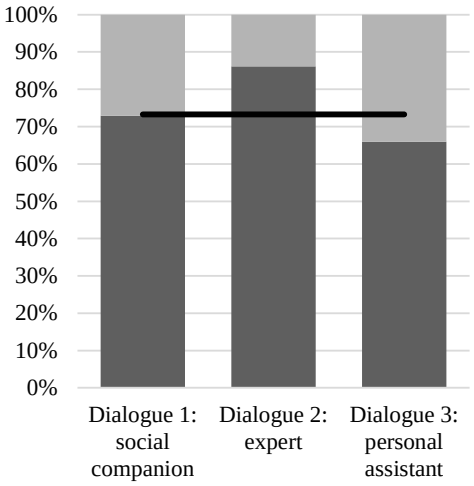


Figure 2: In Dialogue 2, the elaborate options (dark) have been chosen significantly ($p < 0.001$) more often than the average over all three dialogues (black line). In Dialogue 3, the concise options (light) have been chosen significantly ($p < 0.001$) more often than the average over all three dialogues. In Dialogue 1, there is no significant difference to the average over all three dialogues.

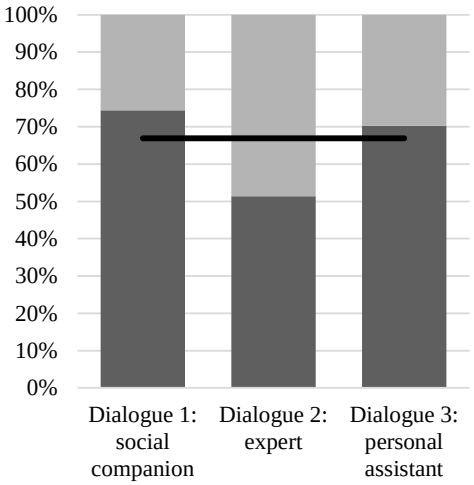


Figure 3: In Dialogue 1, the direct options (dark) have been chosen significantly ($p < 0.001$) more often than the average over all three dialogues (black line). In Dialogue 2, the indirect options (light) have been chosen significantly ($p < 0.001$) more often than the average over all three dialogues. In Dialogue 3, the direct options (dark) have been chosen significantly ($p < 0.05$) more often than the average over all three dialogues.

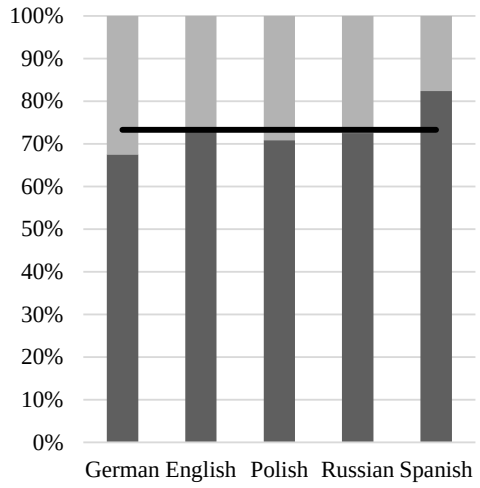


Figure 4: Among all cultures, the elaborate versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the concise versions (light). Moreover, there are significant differences to the average (black line) for German ($p < 0.05$) and Spanish ($p < 0.001$). There are no significant differences to the average for English, Polish and Russian.

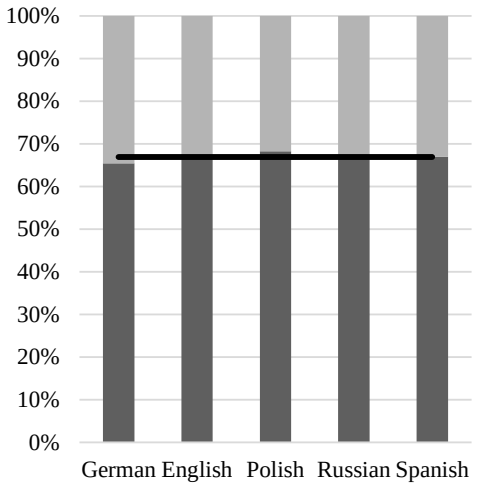


Figure 5: Among all cultures, the direct versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the indirect versions (light). There are no significant differences to the average (black line).

3.2. Influence of the user's culture

In this section, the influence of the user's culture is discussed. As described in Section 2.2., a total of 339 persons from five different cultures participated in the study.

The results concerning the *elaborateness* are shown in Figure 4. It can be seen that the participants of all cultures selected the elaborate versions (dark) of the system utterances significantly more often than the concise versions (light). This shows that all five cultures prefer an elaborate communication style. However, there are small differences among the investigated cultures. While the German participants selected the elaborate options significantly less often than the average (73.28%), the Spanish participants selected the elaborate options significantly more often than the average.

The results concerning the *directness*, which are depicted in Figure 5, show that the participants of all cultures selected the direct versions (dark) of the system utterances significantly more often than the indirect versions (light). This indicates that all five cultures prefer a direct communication style. Moreover, there are no significant differences to the average (66.89%).

This leads us to the conclusion that there is no difference between the investigated European cultures concerning the *directness* of the system's output. In contrast, there are indeed significant differences on the user's preference of the system's *elaborateness*.

3.3. Influence of the user's gender

In the following, the influence of the user's gender is investigated. As described in Section 2.2., altogether 166 male and 173 female persons participated in our study.

The results are depicted in Figures 6 and 7. It can be seen that both the male and the female participants selected the elaborate and direct versions (dark) of the system utterances significantly more often than the concise and indirect versions (light) and that there are no significant differences to the averages. Moreover, no significant difference between the two groups could be found concerning the system's *directness* whereas the women selected the elaborate options significantly more often than the men. This leads us to the conclusion that the gender does not influence the user's preference concerning the *directness* of a system utterance. In contrast, the gender seems to influence the preference concerning the *elaborateness*. Even if both genders prefer the elaborate options over the concise options, the female participants selected the elaborate options significantly more often than the male participants did.

3.4. Influence of the user's culture and gender

In Sections 3.2. and 3.3., the participants' culture and gender have been considered separately. In the following, we will examine whether there are gender differences within the cultures.

The results concerning the *elaborateness* are shown in Figure 8. The German female participants selected the elaborate options significantly more often than the German male participants did and the Polish female participants selected the elaborate options significantly more often than the Polish male participants did. These results support the conclusion drawn from the results depicted in Figure 6 that the gender may influence the user's preference concerning the *elaborateness* of the system utterances. In contrast, there are no significant differences between men and women for English, Russian and Spanish, what leads us to the conclusion that it depends on the culture whether there are gender differences concerning the *elaborateness*.

The results concerning the *directness*, which are depicted in Figure 9, show that there is a significant difference between men and women for Spanish: the Spanish female participants selected the direct options significantly more often than the Spanish male participants did. There are no sig-

nificant differences between men and women for German, English, Polish and Russian. This shows that in some cultures the gender may indeed influence the user's preference concerning the *directness* of the system utterances and that the conclusion drawn from the results depicted in Figure 7 are not valid for all cultures.

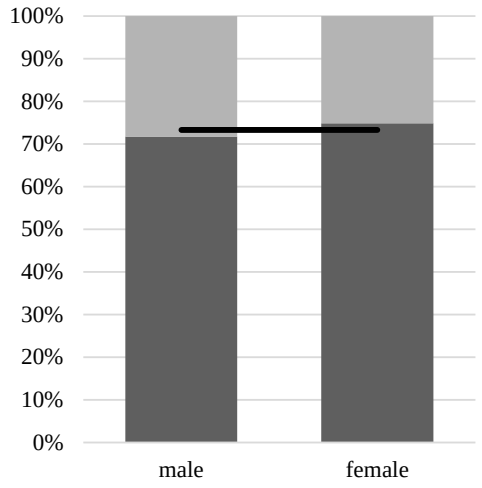


Figure 6: Both the male and the female participants selected the elaborate versions (dark) of the system utterances significantly ($p < 0.001$) more often than the concise versions (light). Moreover, there is a significant ($p < 0.05$) difference between the two groups. There are no significant differences to the average (black line).

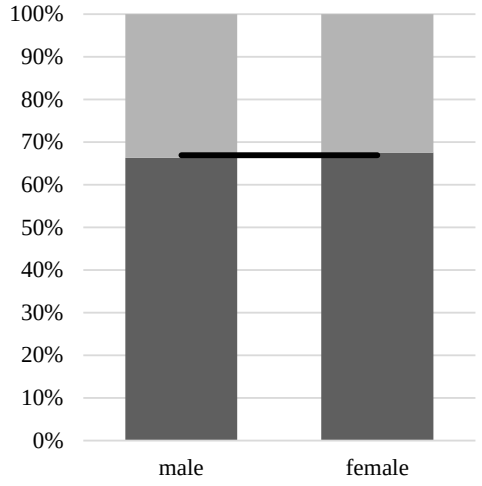


Figure 7: Both the male and the female participants selected the direct versions (dark) of the system utterances significantly ($p < 0.001$) more often than the indirect versions (light). There is no significant difference between the two groups and there are no significant differences to the average (black line).

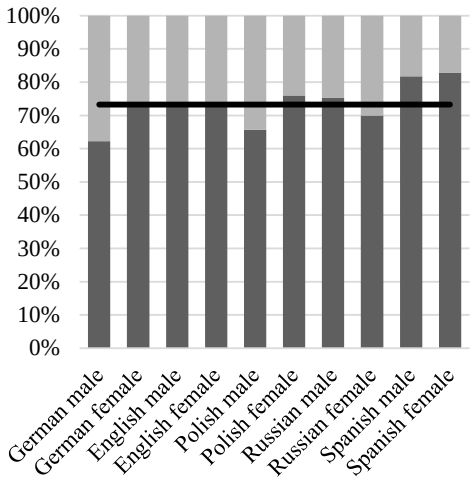


Figure 8: Among all cultures and genders, the elaborate versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the concise versions (light). Moreover, there are significant differences between men and women for German ($p < 0.005$) and Polish ($p < 0.005$). There are no significant differences between men and women for English, Russian and Spanish.

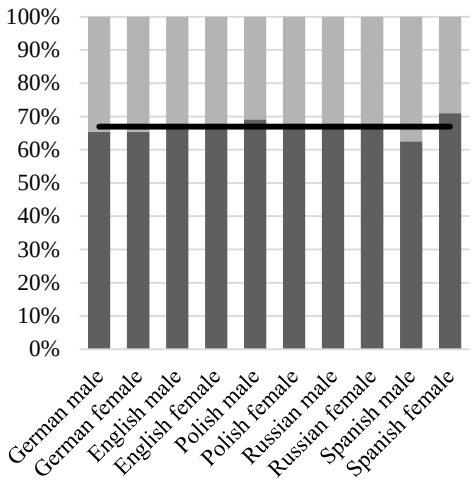


Figure 9: Among all cultures and genders, the direct versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the indirect versions (light). Moreover, there is a significant difference between men and women for Spanish ($p < 0.05$). There are no significant differences between men and women for German, English, Polish and Russian.

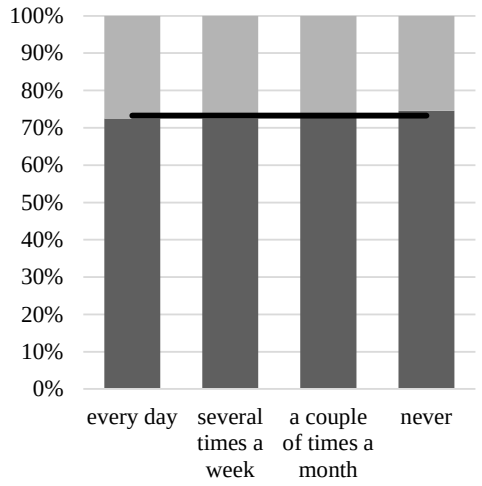


Figure 10: Among all four groups, the elaborate versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the concise versions (light). There are no significant differences to the average (black line).

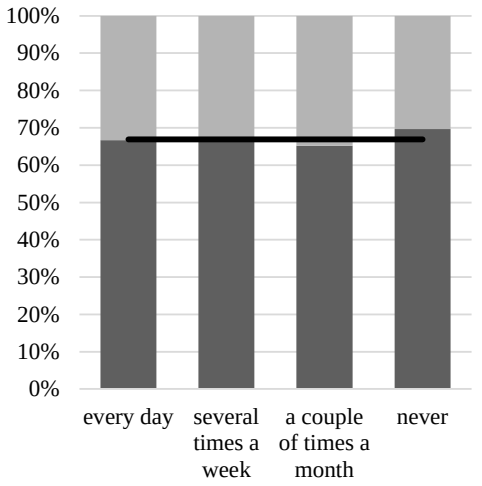


Figure 11: Among all four groups, the direct versions (dark) of the system utterances have been chosen significantly ($p < 0.001$) more often than the indirect versions (light). There are no significant differences to the average (black line).

3.5. Influence of the user's frequency of use of speech based assistants

In this section, the influence of the user's frequency of use of speech based assistants like Apple Siri, Google Assistant or Microsoft Cortana is discussed. As described in Section 2.2., the study participants had to rate how often they use a speech based assistant on the following scale:

- every day
- several times a week
- a couple of times a month
- never

The results concerning the *elaborateness* are shown in Figure 10, those concerning the *directness* are depicted in Figure 11. It can be seen that among all four groups, the elaborate and the direct versions (dark) of the system utterances have been chosen significantly more often than the concise and the indirect versions (light). Moreover, for both the *elaborateness* and the *directness*, there are no significant differences to the averages. Therefore, we conclude that the user's frequency of use of speech based assistants does not influence their preference in the system's communication style.

3.6. Summary of the findings

Summing up the results of the user study which have been described in Sections 3.1.-3.5., we conclude:

- The system's role significantly influences the user's preference in the system's communication style.
- There is no difference between the investigated European cultures concerning the *directness* of the system's output.
- In contrast, there are cultural differences on the user's preference of the system's *elaborateness*.
- It depends on the culture whether there are gender differences concerning the *elaborateness* and *directness* of the system utterances.
- The user's frequency of use of speech based assistants does not influence their preference in the system's communication style.

4. Conclusion and Future Directions

In this work, we presented a multicultural study investigating what causes the differences in the communication styles *directness* and *elaborateness* in Human-Computer Interaction. Our aim was to explore the influence of the user's culture and gender, the frequency of use of speech based assistants as well as the system's role. Therefore, we created three dialogues with different options for the system output and conducted a user study with 339 participants from Germany, Russia, Poland, Spain and the United Kingdom. The study participants had to indicate their preference concerning the system output in every dialogue turn. With the results of this study, we have shown that the system's role significantly influences the user's preference in the system's communication style. Moreover, we recognised differences among the cultures even though five European cultures were examined whose communication styles are very alike. We also showed that it depends on the culture whether there are gender differences concerning the user's preference in the system's communication style and that the user's frequency of use of speech based assistants has no influence on the user's preference in the *directness* and *elaborateness* of the system.
In future work, we have to identify how the different dimensions that cause the differences in the communication styles *elaborateness* and *directness* may be implemented in the

Dialogue Management to design a Spoken Dialogue System which adapts its behaviour to the user's communication idiosyncrasies.

5. Acknowledgements

This work is part of a project that has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 645012.

6. Bibliographical References

Brennan, S. E. (1996). Lexical entrainment in spontaneous dialog. *Proceedings of ISSD*, 96:41–44.

Burgoon, J. K., Stern, L. A., and Dillman, L. (2007). *Interpersonal adaptation: Dyadic interaction patterns*. Cambridge University Press.

Burleson, B. R. (2003). The experience and effects of emotional support: What the study of cultural and gender differences can tell us about close relationships, emotion, and interpersonal communication. *Personal Relationships*, 10(1):1–23.

Casanueva, I., Hain, T., Christensen, H., Marxer, R., and Green, P. (2015). Knowledge transfer between speakers for personalised dialogue management. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 12.

Cassell, J. and Bickmore, T. (2003). Negotiated collusion: Modeling social language and its relationship effects in intelligent agents. *User Modeling and User-Adapted Interaction*, 13(1):89–132.

Doran, C., Aberdeen, J., Damianos, L., and Hirschman, L. (2003). Comparing several aspects of human-computer and human-human dialogues. In *Current and new directions in discourse and dialogue*, pages 133–159. Springer.

Elliott, C., Adams, R. J., and Sockalingam, S. (2016). Multicultural toolkit: Toolkit for cross-cultural collaboration. Awesome Library. `http://www.awesomelibrary.org/multiculturaltoolkit.html`. Accessed: 2016-05-01.

Forbes-Riley, K., Litman, D., and Rotaru, M. (2008). Responding to student uncertainty during computer tutoring: An experimental evaluation. *Lecture Notes in Computer Science*, 5091:60–69.

Hempel, T. (2008). *Usability of speech dialog systems: listening to the target audience*. Springer Science & Business Media.

Hofstede, G. (2009). *Culture's Consequences: Comparing Values, Behaviors, Institutions and Organizations Across Nations*. Sage.

Honold, F., Bercher, P., Richter, F., Nothdurft, F., Geier, T., Barth, R., Hoernle, T., Schüssel, F., Reuter, S., Rau, M., Bertrand, G., Seegebarth, B., Kurzok, P., Schattenberg, B., Minker, W., Weber, M., and Biundo, S. (2014). Companion-technology: Towards user- and situation-adaptive functionality of technical systems. In *10th International Conference on Intelligent Environments (IE 2014)*, pages 378–381. IEEE.

Kaplan, R. B. (1966). Cultural thought patterns in intercultural education. *Language learning*, 16(1-2):1–20.

Lewis, R. D. (2010). *When Cultures Collide: Leading Across Cultures*. Brealey.

Mairesse, F. and Walker, M. A. (2010). Towards personality-based user adaptation: psychologically informed stylistic language generation. *User Modeling and User-Adapted Interaction*, 20(3):227–278.

Miehle, J., Yoshino, K., Pragst, L., Ultes, S., Nakamura, S., and Minker, W. (2016). Cultural communication idiosyncrasies in human-computer interaction. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 74–79, Los Angeles, USA, September. Association for Computational Linguistics.

Nenkova, A., Gravano, A., and Hirschberg, J. (2008). High frequency word entrainment in spoken dialogue. In *Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: Short papers*, pages 169–172. Association for Computational Linguistics.

Niederhoffer, K. G. and Pennebaker, J. W. (2002). Linguistic style matching in social interaction. *Journal of Language and Social Psychology*, 21(4):337–360.

Pickering, M. J. and Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(2):169–190.

Pragst, L., Ultes, S., Kraus, M., and Minker, W. (2015). Adaptive dialogue management in the kristina project for multicultural health care applications. In *Proceedings of the 19thWorkshop on the Semantics and Pragmatics of Dialogue (SEMDIAL)*, pages 202–203.

Pragst, L., Minker, W., and Ultes, S. (2017). Exploring the applicability of elaborateness and indirectness in dialogue management. In *Proceedings of the 8th International Workshop On Spoken Dialogue Systems (IWSDS)*.

Reitter, D., Keller, F., and Moore, J. D. (2006). Computational modelling of structural priming in dialogue. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 121–124. Association for Computational Linguistics.

Stenchikova, S. and Stent, A. (2007). Measuring adaptation between dialogs. In *Proc. of the 8th SIGdial Workshop on Discourse and Dialogue*.

Ultes, S., Kraus, M., Schmitt, A., and Minker, W. (2015). Quality-adaptive spoken dialogue initiative selection and implications on reward modelling. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 374–383.

FARMI: A FrAmework for Recording Multi-Modal Interactions

Patrik Jonell[1], Mattias Bystedt[1], Per Fallgren[1]
Dimosthenis Kontogiorgos[1], José Lopes[1], Zofia Malisz[1]
Samuel Mascarenhas [2], Catharine Oertel[1], Eran Raveh[3], Todd Shore[1]

[1] KTH Speech, Music and Hearing, Stockholm, Sweden
[2] GAIPS INESC-ID, Lisbon, Portugal
[3] Multimodal Computing and Interaction, Saarland University, Germany
pjjonell@kth.se

Abstract

In this paper we present (1) a processing architecture used to collect multi-modal sensor data, both for corpora collection and real-time processing, (2) an open-source implementation thereof and (3) a use-case where we deploy the architecture in a multi-party deception game, featuring six human players and one robot. The architecture is agnostic to the choice of hardware (e.g. microphones, cameras, etc.) and programming languages, although our implementation is mostly written in Python. In our use-case, different methods of capturing verbal and non-verbal cues from the participants were used. These were processed in real-time and used to inform the robot about the participants' deceptive behaviour. The framework is of particular interest for researchers who are interested in the collection of multi-party, richly recorded corpora and the design of conversational systems. Moreover for researchers who are interested in human-robot interaction the available modules offer the possibility to easily create both autonomous and wizard-of-Oz interactions.

Keywords: multisensory processing, human-robot interaction, multimodal interaction

1. Introduction

Recording group interactions is an essential step towards a better understanding of the mechanisms of multi-party interaction both between humans and between humans and robots. In such interactions, humans communicate via a number of different channels. Expressions in any modality, as well as the interactions between modalities and participants, need to be reliably captured to enable a proper analysis of how humans communicate in group settings. Capturing multi-modal and multi-party interaction is still a tremendous challenge in terms of hardware and software development. Recordings of this kind generate very large amounts of data. Ensuring synchronisation between sensor, audio and video streams is crucial for reliable analysis.

When it comes to recording multi-modal datasets or implementing multi-modal dialogue systems, it is difficult to find a framework that has been widely adopted by the community. Given that there are a lot of constraints imposed by the hardware, often the only option left is to develop a framework around that same hardware. In addition, such custom frameworks frequently lack detailed documentation. There were previous attempts to develop frameworks that support multi-modal dialogue systems, for instance (Bohus and Horvitz, 2009) and IrisTK (Skantze and Al Moubayed, 2012). IrisTK enables the recording of multi-modal data collections, however, data stream synchronisation is not fully solved. Mint.tools (Kousidis et al., 2013) tackle synchronisation problems and perform recordings in an annotation friendly manner, however this framework does not close the loop into a dialogue system and is not, to our knowledge, publicly available.

An artificial agent needs to have additional methods of perceiving human behaviour in order to be context aware in human-robot interaction (Turk, 2014). Apart from natural language understanding, several shallow input modalities such as eye gaze or body posture can be used to detect different patterns of human behaviour and reason towards generating appropriate agent behaviour and feedback. Open source frameworks for real-time recognition of social signals from many sensors exist (Wagner et al., 2013) also featuring machine learning and pattern recognition tools. Such modules can be included in dialogue system pipelines. (Thompson and Bohus, 2013; Bohus et al., 2017) provide a platform for analysis and development of multimodal interactive systems that is extensible and deployable for many situated intelligence tasks such as speech recognition leveraging multimodal signals, human-robot interaction supported with face recognition and so on. The platform is restricted however to sensors provided by one software company.

2. Goals

In this paper, we first describe the proposed architecture. By architecture we mean an abstract conceptualisation of how the components of the system interplay. Next, we present a framework that implements the aforementioned architecture.

Our goal is to tackle the problems encountered in previous attempts and present an open-source framework specifically developed to handle multi-party, multi-modal data recordings. The framework is modular, lending flexibility to different future applications. It has been designed to be modality agnostic, meaning that several modalities can be added or removed depending on the sensor data available and the perception models to be used. Both code and documentation are available for download [1].

[1] https://github.com/kth-social-robotics/multisensoryprocessing

As a test case for the framework, we present an application scenario involving the Furhat robot-head (Al Moubayed et al., 2012) as a participant in a role-playing game "Werewolf". In the scenario, we include several input modalities of verbal and non-verbal human cues that can lead to better understanding of the context and situational awareness of the conversation: speech, gaze and body posture and movement. In the case study, as we describe in Section 5., we start by collecting low level sensor data that we further aggregate in near-real time to higher level decision making models. We use such models in order to reason effectively how to generate appropriate agent behaviour. Note that in our implementation, we do not support e.g. incremental processing, such as the approach presented in (Kennington et al., 2017), but rather, after aggregating gathered data, we make a decision on what the current state or intent of the participant is.

3. Architecture

The framework implements a modular message-oriented architecture for sending, receiving and handling of sensory data from multiple sources in a distributed manner. Beyond the sensory hardware and software itself, the system is composed of fours layers: Sensor API modules, sensor data preprocessors, message processors and end consumers. Each layer is composed of individual modules which send and/or receive information from other modules asynchronously via a message broker. Despite being published and consumed in an asynchronous manner, each message includes information from a central time server that enables messages published by different physical systems to be synced to within a negligible margin of error.

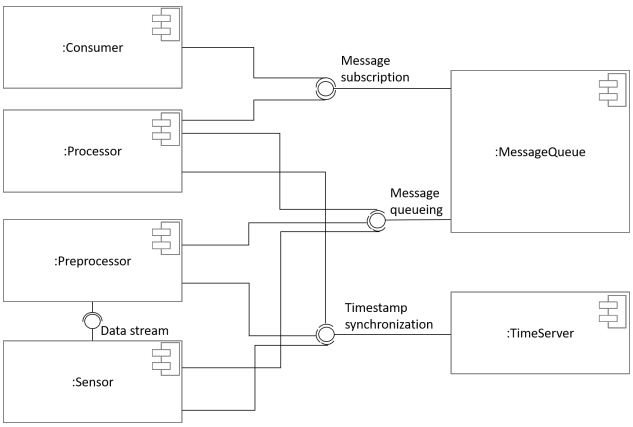


Figure 1: Interfaces between application layers.

Sensor modules interface with the sensory control software (e.g. for sound cards or gaze-tracking equipment), sending sensory data to any module which is listening on its relevant socket. Upon initialisation, each sensor module broadcasts information using the message broker which is required to connect to it and read socket data.

Pre-processors convert data streams from sensor modules into individual messages, which are submitted to the message queue for publishing to other modules. For instance, the speech recognition result in the case of the microphone sensors.

Processors process messages from other modules which they have been registered to receive via the message queue, e.g. performing dialogue act detection recognition on the data received from the speech recogniser or predicting visual attention from gaze data.

Consumers use data handled by the framework for domain-specific purposes, e.g. dialogue planning, language generation or action planning for robots or virtual agents.

The individual modules in each layer are only grouped conceptually; A given module defines the module(s) it should be listening to through the message queue, and thus is dependent only on the message queue itself and the time server to synchronise timestamps for messages as well as raw data streams (see Figure 1).

3.1. Data Synchronisation

Data streams and messages are synchronised by offsetting the timestamps using a delta function from a central time server, ensuring that timestamps for data from each module are comparable. When each module initialises, it requests a time from the time server; The difference of this time from the local time is then used to offset timestamps sent with data from that module. Downstream modules, which process data produced by other modules (i.e. all but sensory modules), include the offset timestamp sent by the previous module in the stream, thus allowing the original time of the original sensory input to be easily reconstructed.

When using a local area network for sensory data transfer, issues related to latency were negligible (see Section 5.). However, the system architecture described here is agnostic to the exact timestamp synchronisation algorithm used and thus a more sophisticated one can be implemented in the case that greater precision is required.

4. Framework

We implemented the aforementioned architecture into a framework as a distributed system. The framework is written in Python and gives the developer an easy way to define sensors, preprocessors, processors and other consumers and manages the communication and synchronisation of the data streams for the user. The framework's main responsibilities are: (1) managing the messaging and the data streams between the different components, (2) manage synchronisation between data streams, and logging and recording of the input signals. For the message queue Rabbit MQ was chosen, and for the data sockets between the components ZeroMQ was used. We present an example in Figure 2 for the case where only speech data is being recorded from a single microphone (sensor), processed with automatic speech recognition (ASR) and interpreted by the Natural Language Understanding (NLU) module.

4.1. Communication Between Components

There are two primary ways of communication between the components (see Figure 2): (1) messages through the message broker, and (2) direct data streams between processing units. The message broker is a centralised server which other processes can connect to, subscribe to topics and receive messages sent out on those topics by other components. The direct data streams are on the other hand a peer-to-peer connection between components. This allows for streaming data without straining the message broker with a flood of messages for each packet. The main limitation is that the receiver needs to know the IP address of the sender. This is handled by each sensor sending out a message through the broker with its own IP address and information on what kind of data it is broadcasting. The communication via the message broker is performed by publishing and listening for messages on given topics. A topic can, for example, look like *microphone.data.participant_A* and a listening process can subscribe to either exact topics or use wildcards in order to receive a broader range of messages, e.g. *microphone.data.** to receive microphone data for all participants.

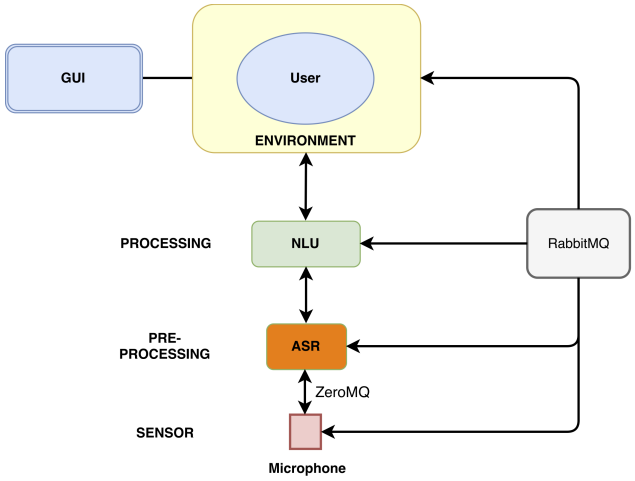


Figure 2: We use a layered structure of data processing. Using the framework we aggregate low-level sensor data to high level information on the human's state and intent from verbal and non-verbal cues. In this example we present a single-sensor usage of the framework that can update the environment in real-time with the incoming utterances.

4.2. Synchronisation of Data Streams

The framework handles synchronisation of data streams in a last in - first out fashion. We process sensor data and aggregate to have a common frame rate amongst the sensors that need to combine data streams. To ensure the data streams have occurred in the same time, each node that the data passes adds its own timestamp as meta-data to the pack. As such the time the data is captured on sensor level is kept on track to be able to handle the latency in the system for differently processed data. Pre-processors listen to messages from a sensor in which there is interest to receive data from, but it can listen to many sensors at the same time which creates the need for latency handling.

Using the same frame rate for combined sensors we can track what frames are expected to be processed an if there are any frames lost in the process. The timestamp of the original data packet sent from a sensor is kept throughout and is therefore are easily aligned. Our synchronisation strategy does not assume very timely latencies and will therefore aggregate data streams from combined sensors only when data from all sensors is available. For example if a pre-processor expects data streams from two types of sensors to aggregate and send to a processor, it will wait until all sensors have completed their data streams before it will pass the information to the next layer.

4.3. Data Transportation

The data streams are transported in local network for smaller latencies, however in our distributed architecture, several components can be part of the framework that are either in the cloud or in remote repositories. As described above the data packs are transported in two ways: through a message broker and through peer-to-peer connections. Each data pack transported across each layer is logged as a message exchanged between components with the timestamps attached.

4.4. Logging and Recording of Data

The framework contains a module that can record and log all of the data being sent between the components. This is done by subscribing to all topics and writing the data to file. It is possible to start multiple recording instances to decrease the load on an individual data recorder, as the quantity of data can become large with high-quality sensor equipment.

As the framework adds a timestamp to each data packet, it is also possible to visualise the flow of data and where potential bottlenecks exists in the system. However, this kind of visualisation is currently not implemented into the framework.

Logging the data as messages adds another advantage to post-processing the recordings. The messages can be replayed as in real-real time in order to reproduce data captured from specific sensors. This helps debugging the recording system, but also replicate human behaviour given specified sensor data.

In the next section we describe how we made use of the proposed architecture by implementing a framework and components that can capture sensor data in a use case scenario performing multi-party multi-modal spoken interaction between humans and a robot.

5. Use Case: The Werewolf Game

A good example for utilising the presented architecture is a multi-modal, multi-party group interaction. This scenario requires multitude of sensors to capture the entire dynamics of the group interaction. In this study, we used the this architecture for recording a multi-modal corpus of the "Werewolf" game. Werewolf is a multi-party role-playing social game. Each player's objective is to eliminate the opposing players by deceiving them regarding the player's identity – a citizen or a Werewolf. Other studies have used this

game before to study deceptive behaviour (Chittaranjan and Hung, 2010) and gaze patterns (Oertel and Salvi, 2013).
The game is suitable for studying deceptive behaviour. However, due to its engaging, multi-party nature, it is also suitable for studying multi-party turn-taking phenomena, and therefore provides the possibility to investigate other research directions as well. For instance, the role of eye-gaze coordination during turn-holds and turn-grabs in a multi-party setting, or relation between participant engagement aggregated eye-gaze patterns. The use of a robot in this use-case poses further possibilities for analysing the collected data. For example investigating how participants react to the robot as one of the players. Are they ignoring the robot or are they treating him as an equal entity? Would they eliminate it more often or would prefer to let it stay in the game? What do participants like about the robot's behaviour and where would they expect further improvements?

5.1. Experimental Setup



Figure 3: Experimental setup of multi-party, multi-modal interactions with a social robot in the "Werewolf" game.

The recording environment presented here is aimed for a Wizard of Oz experimental setup. Although it can be adjusted to function without a wizard, these efforts are beyond the scope of this work.

5.1.1. Multi-Sensory Data and Hardware Setup

During the game, the players sit around a table as shown in Figure 3, so that each player can create a direct eye contact with all the other players. In this setup, the moderator was not seen by the players, but heard through a loudspeaker. The setup was designed for a game with six human players and one robot player. The number of participant is only limited by the amount of sensors available, as the architecture can collect data for each sensor independently.
Each participant had a microphone for recording and recognising their speech, a Kinect v2[2] camera or Tobii glasses[3] for capturing eye gaze, gloves and a hat with motion capture reflective markers for capturing hand gestures and head pose, and a - camera for capturing facial expressions. Each of these hardware sensors had a equivalent software sensor (see Table 1) to process and send the raw data to the rest of the system. In addition, a video camera was positioned above the table to provide the wizard (who sat behind a curtain) a live feed of the session. This camera's data was not sent to the rest of the system, i.e. was not recorded or taken into account while playing the game. All of the raw data from the sensors were recorded and saved to an external storage in real time.
The eye-gaze information from the Kinect was translated into a fixation on a specific player, if applicable, in real time using GazeSense[4]. The equivalent information coming from data recorded by the Tobii glasses was recorded and after combined with motion capture, mapped to 3D space in a similar manner. Furthermore, any information that might be relevant for the robot's decision making was fed into the FAtiMA system (Dias et al., 2014), as described below. The system includes some hand-crafted theory of mind rules (Gardner, 2011), which are designed to detect deception.

5.2. Architecture Implementation

5.2.1. Pre-Processors

All of the sensors mentioned in 5.1.1. had a corresponding pre-processor, as specified by the architecture. The eye-tracking, motion capture and Kinect gaze pre-processors parsed the incoming data and sent them out as messages through the message broker. Players' spoken utterances were sent as raw data to IBM Watson's cloud-based automatic speech recognition (ASR)[5]. OpenFace (Baltrušaitis et al., 2016) was used for facial feature extraction for extracting action units (Ekman and Friesen, 1978).

5.2.2. Processors

In this use case, we mapped the ASR output onto the most likely dialogue act among *accuse*, *defend* or *support*, using a template file with common n-grams characterising these dialogue acts, as found in a Werewolf corpus (Oertel and Salvi, 2013). Since the ASR sends incremental data, we constantly updated this as soon as new ASR results are available. For the facial expression recogniser, we used action units obtained by OpenFace. The processor could collect several high-level facial cues that are valuable for the decision-making component. The cues (e.g. opened or closed eyes, frowning, smiling, etc.) could then, with a basis in literature on deception, be used as behavioural features for deciding whether a player is behaving deceptively or not.

5.2.3. Decision-Making

Given the high amount of sensory input and processor units that are present in the architecture, there is a need for a decision-making component that receives processed high-level information from the sensors and points to concrete actions. The cognitive framework Fearnot AffecTIve Mind Architecture (FAtiMA) (Dias and Paiva, 2005) was used to achieve that. In particular, the modular version presented in (Dias et al., 2014). FAtiMA is a flexible system designed to work in an array of different settings where a user wants to

[2] https://developer.microsoft.com/windows/kinect
[3] https://www.tobiipro.com/
[4] https://eyeware.tech/
[5] https://www.ibm.com/watson/services/speech-to-text/

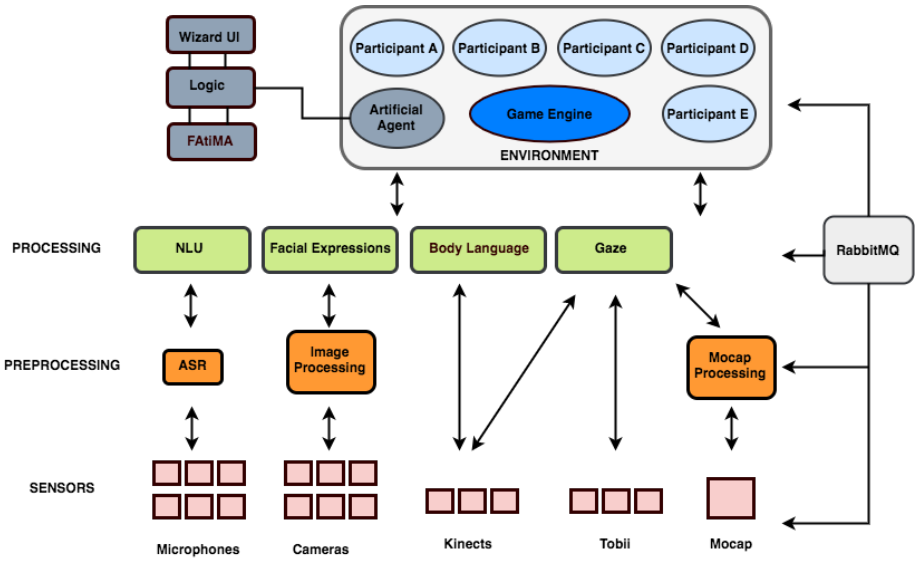


Figure 4: The multi-sensory processing architecture used in the use case study. We have separated the processing of information into three data processing layers: sensor, preprocessor and processor. Each participant has a current state initiated in the processing environment given the aggregated data captured by the previous layers.

create an agent with some degree of higher-level information processing without the need of implementing a system from scratch.

In our system, FAtiMA is based on two components: A dynamic belief system which is updated consecutively, and a fixed set of rules.

In the setting of the Werewolf game, a simple rule could, for instance, be that Furhat should accuse a player of lying if they accused another participant that according to the belief system has behaved truthfully in the previous round. Unless stated otherwise, the belief system will not initially have any information regarding the truthfulness of participant. This information has to be explicitly given to the framework from collected data via the pipeline of processed sensory input. By implementing a series of rules and updating the belief system accordingly through the processes of the framework, a more sophisticated decision-making system emerges. This framework can be used to facilitate the decisions of a wizard, which was the case in the Werewolf scenario. However, the framework does also provide the grounds necessary to represent a fully autonomous agent instead of the wizard.

5.2.4. Wizard Interface

In the current setting, the wizard has access to a few actions via a computer keyboard. These actions correspond to the three dialogue act categories identified in the data collected in (Oertel and Salvi, 2013): *accuse*, *defend* and *support*. For each of these acts, the template file randomly picks an utterance from a pre-defined set. Besides these acts, a few other common dialogue acts and "small talk" utterances are available. Once the game reached the voting round, the wizard uses the an interface to vote in the name of the robot player. While voting, the likelihood of each participant being the werewolf as calculated by the decision-making system is displayed in the wizard interface.

Sensors	Microphones Kinect USB cameras Motion capture (Vicon) Eye-trackers (Tobii)
Pre-processors	Automatic speech recognition Eye-tracking data parser Kinect gaze data parser Motion capture data parser Facial feature extraction
Processors	Dialogue act recogniser Facial expression recogniser

Table 1: The processing layers and their corresponding processing units that were implemented for the Werewolf game.

6. Discussion

The current paper described an architecture for recording multi-modal interactions. Its purpose was threefold. First, it proposed a solution to handle data-stream synchronisation. Second, it was designed to be easily extendable and third and final it was created to be freely available to everyone dealing with multi-modal systems.

The framework was applied to the ?werewolf" use-case scenario. This scenario proved to be quite challenging. Six players were interacting in real-time with a robot. We were tracking voice activity and gaze (respectively head direction) simultaneously for all participants. Moreover, we converted the sensor information into visualisations that were displayed in the wizard interface. We encountered several challenges in this use-case. For instance, we found that some sensors required the complete use of a computer, blocking the use of other sensors, at a given time. This resulted in a very complex system using multiple computers.

The choice for a distributed system architecture seemed appropriate in this case, and its behaviour proved to be robust. However, there is certainly room for further improvements to the framework. For example, currently the framework still requires to start and maintain processes on each individual machine. In addition, system debugging is currently not trivial as messages are broadcasted to the whole system. It is often difficult to track in the code possible bugs. Improved logging features and extended documentation about all of the components is needed and is currently being developed. Despite the fact that there surely are some more challenges to be solved, depending on use case scenario, we believe that the framework we are releasing can be very useful to the community.

7. Conclusion and Future Work

In the current paper, we presented a generalisable architecture for multi-modal, multi-party recordings. We illustrated its usability by relating it to research questions of the "Werewolf" use-case scenario. Future work consists of creating new processing units and automating the process of installing and running multiple sensors on multiple computers. The authors started implementing the architecture in other projects. For example, a sensor unit exists in place for data input using an OptiTrack[6] motion capture system.

8. References

Al Moubayed, S., Beskow, J., Skantze, G., and Granström, B. (2012). Furhat: A back-projected human-like robot head for multiparty human-machine interaction. In Anna Esposito, et al., editors, *Cognitive Behavioural Systems*, pages 114–130, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg.

Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2016). Openface: an open source facial behavior analysis toolkit. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE.

Bohus, D. and Horvitz, E. (2009). Dialog in the open world: platform and applications. In *Proceedings of the 2009 international conference on Multimodal interfaces*, pages 31–38. ACM.

Bohus, D., Andrist, S., and Jalobeanu, M. (2017). Rapid development of multimodal interactive systems: a demonstration of platform for situated intelligence. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 493–494. ACM.

Chittaranjan, G. and Hung, H. (2010). Are you a werewolf? detecting deceptive roles and outcomes in a conversational role-playing game. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 5334–5337. IEEE.

Dias, J. and Paiva, A. (2005). Feeling and reasoning: A computational model for emotional characters. In *EPIA*, volume 3808, pages 127–140. Springer.

Dias, J., Mascarenhas, S., and Paiva, A. (2014). Fatima modular: Towards an agent architecture with a generic appraisal framework. In *Emotion Modeling*, pages 44–56. Springer.

Ekman, P. and Friesen, W. V. (1978). *Facial action coding system: A technique for the measurement of facial movement*. Consulting Psychologists Press, Palo Alto, CA, USA.

Gardner, H. (2011). *Frames of mind: The theory of multiple intelligences*. Basic books.

Kennington, C., Han, T., and Schlangen, D. (2017). Temporal alignment using the incremental unit framework. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 297–301. ACM.

Kousidis, S., Pfeiffer, T., and Schlangen, D. (2013). Mint. tools: Tools and adaptors supporting acquisition, annotation and analysis of multimodal corpora. *Proceedings of Interspeech 2013*.

Oertel, C. and Salvi, G. (2013). A gaze-based method for relating group involvement to individual engagement in multimodal multiparty dialogue. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 99–106. ACM.

Skantze, G. and Al Moubayed, S. (2012). IrisTK: a statechart-based toolkit for multi-party face-to-face interaction. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 69–76.

Thompson, A. L. and Bohus, D. (2013). A framework for multimodal data collection, visualization, annotation and learning. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 67–68. ACM.

Turk, M. (2014). Multimodal interaction: A review. *Pattern Recognition Letters*, 36:189–195.

Wagner, J., Lingenfelser, F., Baur, T., Damian, I., Kistler, F., and André, E. (2013). The social signal interpretation (ssi) framework: multimodal signal processing and recognition in real-time. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 831–834. ACM.

[6] `http://optitrack.com`

Creating Large-Scale Argumentation Structures for Dialogue Systems

Kazuki Sakai[1], **Akari Inago**[2], **Ryuichiro Higashinaka**[3],
Yuichiro Yoshikawa[1], **Hiroshi Ishiguro**[1], **Junji Tomita**[3]
[1]Osaka University, [2]Ochanomizu University, and [3]NTT Media Intelligence Laboratories

Abstract

We are planning to develop argumentative dialogue systems that can discuss various topics with people by using large-scale argumentation structures. In this paper, we describe the creation process of these argumentation structures. We created ten structures each having more than 2000 nodes of five topics in English and five topics in Japanese. We analyzed the created structures for their characteristics and investigated the differences between the two languages. We conducted an evaluation experiment to ascertain that the structures can be applied to dialogue systems. We conducted another experiment to use the created argumentation structures as training data for augmenting the current argumentation structures.

Keywords: Argumentation, Large-scale argumentation structures, Dialogue systems

1. Introduction

Argumentation is a process of reaching consensus through premises and rebuttals and is important for making decisions and exchanging views. Argumentation has long been studied in the fields of rhetoric, informal logic, and more recently artificial intelligence. For example, there have been studies on automatically extracting conclusions and premises from documents (Rosenthal and McKeown, 2012; Yanai et al., 2016; Lippi and Torroni, 2016). Other studies have devised argumentation models (Toulmin, 1958; Reed and Rowe, 2004; Walton, 2013) and visualizations of the models (Gordon et al., 2007; Reed and Rowe, 2004; Snaith et al., 2010). To develop dialogue systems that can support humans in argumentation, we are planning to develop argumentative dialogue systems that can discuss various topics with people by using large-scale argumentation structures; we argue that large-scale argumentation structures are necessary for systems to respond appropriately to various arguments raised by users.

Recently, corpora containing argumentation structures of discussions/meetings have been made available (Janin et al., 2003; Renals et al., 2007). Other studies have extracted argumentation structures from corpora (Fernández et al., 2008; Bui et al., 2009). However, these structures are small; thus, insufficient as knowledge for dialogue systems. One of the largest argumentation databases currently available is AIFdb (Lawrence et al., 2012), which is an open database containing argumentation structures in argument interchange format (AIF). Although AIFdb contains many argumentation structures, each structure is small; for example, the largest contains about 250 nodes, and the average number of nodes per structure is 8.14 (See Table 1).

In this paper, we describe the creation process to construct large-scale argumentation structures for dialogue systems. We manually created several large-scale argumentation structures based on a conventional argumentation model (Walton, 2013), and each structure has more than 2000 nodes. We created the structures in two languages; English (major language) and Japanese (author language). To verify the effectiveness of the created argumentation structures, we conducted an evaluation experiment to ascertain that the structures can be applied to dialogue systems. We conducted another experiment to use the created argumentation structures as training data for augmenting the current argumentation structures.

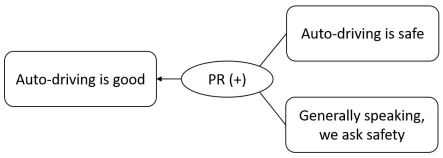


Figure 1: Argumentation model

2. Argumentation Structures

We describe the argumentation model on which our argumentation structures are based then discuss our process of creating them.

2.1. Argumentation Model

As shown in Figure 1, we use a simplified version of the model described in (Walton, 2013). The model has a graph structure, and nodes represent premises and edges represent relationship between nodes. Each node has a natural language statement representing the content of its premise. A node is connected to other nodes by directed arcs that represent a supportive (+) or non-supportive (−) relationship. If the logical connection is based on an argumentation scheme (Walton, 1996) (e.g. practical reasoning (PR), the logic is "if G is something good and action X leads to G, X should be done"), the scheme name is represented on the arcs.

Figure 2 shows the design of our argumentation structure with a specific purpose for dialogue systems. The structure has two parts represented by main issue nodes that enable the system to have opposing stances (we call the stances A and B). Below the main issue nodes, there are what we call viewpoints nodes that represent conversational topics. Under each viewpoint node, there are premise nodes that represent statements regarding each topic.

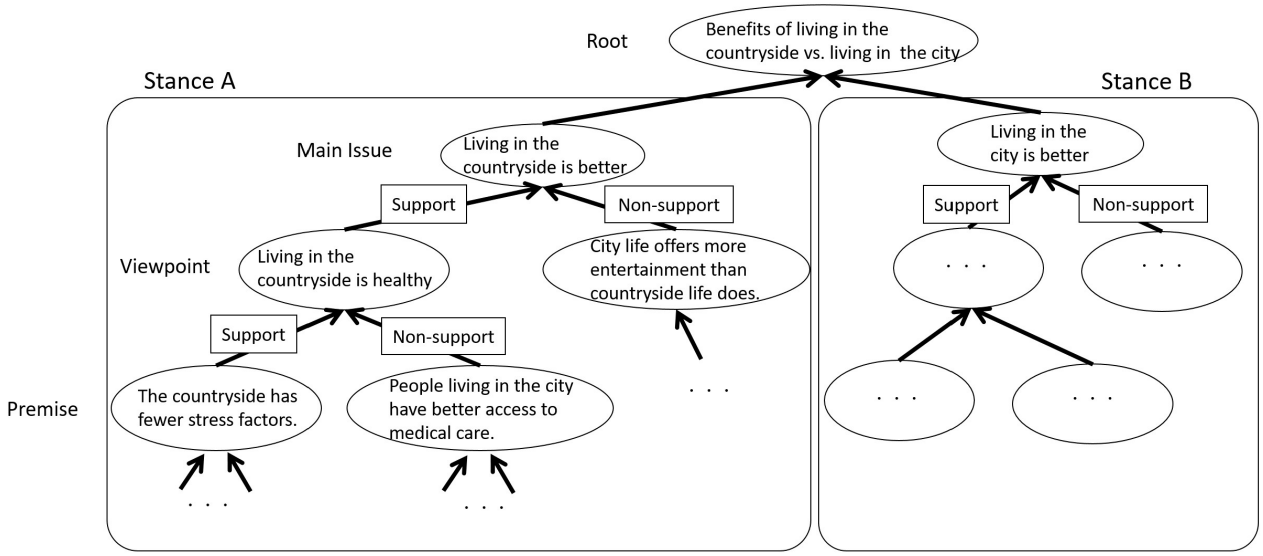


Figure 2: Design of our argumentation structures for dialogue systems

Let us suppose we have a structure about "Benefits of living in the countryside vs. living in the city", which is represented as the root node. The main issue nodes are "Living in the countryside is better" and "Living in the city is better". The viewpoint nodes would include "Living in the countryside is healthy". The premise nodes would include statements such as "The countryside has fewer stress factors" and "People living in the city have better access to medical care". With this design, we believe we can create dialogue systems that can discuss on a certain issue with a stance and from multiple viewpoints.

2.2. Creation Process

Now let us describe the process of creating large-scale argumentation structures. First, we determine the main proposition and two main issue nodes representing two stances. Next, we create viewpoint nodes that represent topics. Then, under the viewpoint nodes, we create premise nodes.

We recruited more than 30 annotators for creating the argumentation structures. First, we determined the propositions and their stances and created the viewpoint nodes to a certain extent. The annotators iteratively created supporting or non-supporting premises for existing premises under viewpoint nodes. When creating premises, they used argumentation schemes whenever possible. To maintain objectivity of the data, the logical relationships between two nodes were checked by other annotators. If the relationship was inappropriate, the annotators corrected or removed the corresponding nodes. This process was repeated until the relationship was appropriate. By repeating this process, we can create large-scale argumentation structures.

2.3. Constructed Structures

We manually created five English and five Japanese structures each with five different topics. Figure 3 shows an example of visualizing the argumentation structures. Each constructed structure has more than 2000 nodes that are supporting or not supporting other nodes. The structures are also hierarchical, as described in Section 2.1.

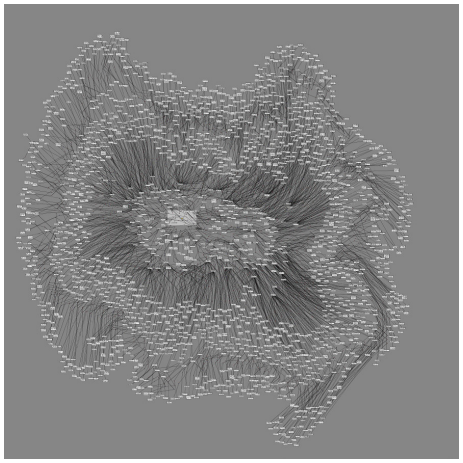


Figure 3: Visualization of constructed structure. Topic is countryside vs. city (English).

The five topics of the graphs written in English are as follows: the pros and cons of driving automobiles (Auto driving), benefits of living in the countryside vs. living in the city (Countryside), who is the greater pop icon, Lady Gaga or Taylor Swift? (Lady Gaga), which is the better Japanese meal, sushi or ramen? (Sushi), and which is the better living environment, east or west coast? (East coast).

The five topics of the structures written in Japanese are as follows: the pros and cons of driving automobiles (Auto driving), benefits of living in the countryside vs. living in the city (Countryside), which is the better place to travel to in Japan, Hokkaido or Okinawa? (Hokkaido), which is the better breakfast, bread or rice? (Bread), and which is the better theme park, Tokyo Disney Resort (TDR) or Universal Studio Japan? (TDR).

	No. of files	No. of nodes per structure		Depth per node		No. of words per node		No. of sentences per node		Branches per node	
		ave.	var.	ave.	var.	ave.	var.	ave.	var.	ave.	var.
AIFdb	7066	8.14	165.52	2.18	1.13	32.44	5389.81	1.95	11.39	1.50	0.78
Our argumentation structures (English)	5	2281.60	5214.64	4.46	0.39	16.90	50.59	1.01	0.01	3.74	19.20
Our argumentation structures (Japanese)	5	2253.80	15885.36	4.42	0.25	19.20	56.33	1.00	0.00	3.63	8.70

Table 1: Statistics of constructed Structures. AIFdb is representative of currently available data. Note that we used nltk (English) for tokenizing words and sentences, and MeCab (Japanese) for tokenizing words.

	English		Japanese	
	scheme	frequency	scheme	frequency
1	CE (+)	141	CE (+)	374
2	EH (+)	99	CO (+)	244
3	EO (+)	59	EX (−)	242
4	PO (+)	47	CO (−)	142
5	VC (+)	43	PO (+)	118
6	AO (−)	40	EH (+)	88
7	AD (+)	33	EO (+)	79
8	CO (+)	30	EC (+)	75
9	PR (+)	28	CE (−)	72
10	EC (+)	26	PO (−)	47

Table 2: Top 10 schemes used in each English and Japanese structures. Sign (+) or (−) with schemes means node supports or refutes another node by using logic represented by scheme, respectively.

A Life in the countryside is better.
B But there is a lot more entertainment facilities in the city than in the countryside.
A But you cannot enjoy the splendor of nature or bountiful natural resources in cities like you can in the country. The countryside offers more areas for exploring.
B But not as many places are available for adventure in the countryside because most of the land is owned by someone as part of their ranch or farm.
A But adventures first started by exploring the natural wonders of the world.
A For many cities, the concept of green space in urban development is relatively new, not to mention being a luxury.
B But cities make full use of the natural resources around them.

Figure 4: First part of dialogue text on topic of countryside vs. city. Stance A is countryside; stance B is city.

2.4. Analysis of Constructed Structures

2.4.1. Comparison with Existing Data

Table 1 shows the statistics of the created structures. We chose AIFdb as the existing data; we crawled some larger structures from the AIFdb website (`http://www.aifdb.org/search`). As can be seen from the table, our structures have more nodes per structure than AIFdb. The number of sentences in the node is comparatively smaller than that of AIFdb, suggesting the possibility of appropriateness for dialogue system utterances. The average number of branches is larger than that of AIFdb and the variance is higher, which indicates the possibility of generating more various utterances for user utterances.

2.4.2. Comparison between Languages

To investigate the difference between languages, we compared the usage of schemes because schemes are likely to reflect the way of thinking in a language. Table 2 shows the top ten schemes used in English and Japanese. There were some differences depending on the language; for example, the scheme called argument from composition (CO), whose logic is "A is part of X, A has property Y, therefore, X has property Y", used in Japanese is more frequent than that used in English. An example argument using CO is that a premise "emergence of automobiles leads to law amendment" concludes "technical innovation leads to law amendment." On the other hand, the scheme called argument from verbal classification (VC), whose logic is "P has property F, and for all x, x has property G if x has F, then P has G", used in English is more frequent than that used in Japanese. An example argument using VC is that two premises "automobiles require the latest electronics" and "the latest electronics are expensive" conclude "auto driving makes vehicles more expensive." Although further examination is needed, if we can create a structure with a certain distribution of schemes, we may be able to develop dialogue systems appropriate to a certain language.

3. Towards Development of Argumentative Dialogue Systems

To apply the created argumentation structures to dialogue systems, we first verified whether the structures can be applied to dialogue generation then developed a dialogue system prototype that used the created structures.

3.1. Dialogue Generation

We generated dialogue text based on argumentation structures. We verified that appropriate argumentative dialogue can be created by just traversing along with the structures if the structures are constructed based on logical relationships. Note that we used only English argumentation structures for this study.

3.1.1. Generation Methods

We created two dialogue texts from each of the five English structures. This was done by traversing structures un-

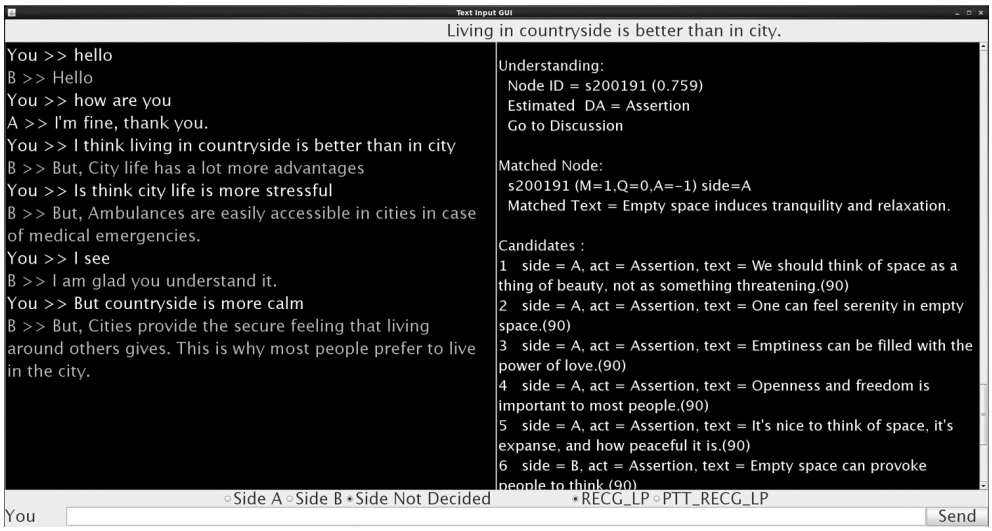


Figure 5: Text chat interface of our dialogue system prototype. The prototype works either in English or Japanese. Here, the topic is "the benefits of living in the countryside vs. living in the city".

questionnaire	item	ave.
Q1	Understandability of content	3.81
Q2	Naturalness of dialogue	3.01
Q3	Understandability of stances	3.52

Table 3: Averaged questionnaire scores

der viewpoint nodes[1] in a depth-first search fashion. When visiting each node, the statement associated with it was appended to the dialogue text. The dialogue-generation process finished when 20 utterances were generated. Figure 4 shows a sample dialogue (on the benefits of living in the countryside versus the city). Note that when transiting to a node from stance A to stance B, or vise-versa, "But" is inserted to make the dialogue look natural.

We recruited 19 participants (12 males and 7 females; average age: 28.9). They read ten dialogue texts (two samples $\times$ five structures) that were randomly ordered and answered a questionnaire. The questionnaire included three questions: (Q1) "Was the meaning of individual utterances easy to understand?", (Q2) "Did the dialogue look natural?", and (Q3) "Was it easy to understand how the two systems agreed or disagreed?" They answered these three questions on a five-point Likert scale, where 5 meant the highest degree of agreement.

3.1.2. Results

Table 3 shows the questionnaire results. The score for Q1 was much higher than 3, so the dialogue texts were understandable to all participants. The score for Q2 and Q3 was higher than 3, so the texts successfully exhibited the characteristics of argumentative dialogue. The evaluation results indicate that the dialogue texts generated by argumentation structures were reasonable, suggesting the effectiveness for use in dialogue systems.

[1]For each structure, we selected the top two viewpoint nodes in the number of descendants.

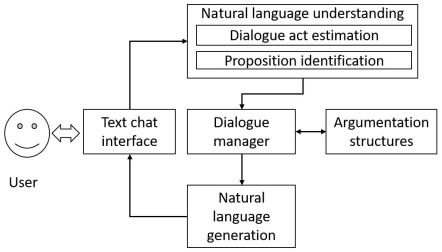


Figure 6: Architecture of our dialogue system prototype. Prototype works either in English or Japanese.

3.2. Dialogue System Prototype

We constructed a dialogue system prototype that argues on the basis of the constructed argumentation structures used in English and Japanese (Higashinaka et al., 2017). Figure 5 shows a text chat interface of our dialogue system prototype, and Figure 6 shows its architecture. When the user types, the sentence is input to two main modules: dialogue act estimation and proposition identification. The dialogue-act-estimation module estimates four types of dialogue acts; assertion, question, concession, and retraction. The proposition-identification module finds, on the basis of word2vec (Mikolov et al., 2013), the corresponding node that has the statement whose meaning is the closest to the input sentence. Finally, the system uses the found node to generate the response text by using the statement of the node. We confirmed that the prototype system can return reasonable responses when the user utterance is within the topic in question.

4. Towards Automatic Creation of Structures

Although the constructed structures are useful for dialogue systems, it is expensive to expand the structures and create new structures of other topics since our current cre-

Table 4: Classification results for English argumentation structures. Bold font indicates the highest score in each column.

	Auto driving	Countryside	Gaga	Sushi	East coast	Average
baseline	0.502	0.524	0.537	0.513	0.511	0.517
SVM (uni + bi + tri)	0.539	0.564	0.575	0.554	0.566	0.560
SVM (uni + uni pair + bi pair)	0.548	0.578	0.597	0.575	**0.580**	0.576
ERT (uni + bi + tri)	0.543	**0.596**	**0.601**	0.583	0.574	**0.579**
MNB (uni + bi + tri)	0.522	0.539	0.533	0.551	0.531	0.535
MNB (uni + uni pair + bi pair)	0.509	0.559	0.468	0.545	0.526	0.522
CBOW (no w2v)	0.532	0.580	0.575	0.587	0.563	0.567
CBOW (use w2v)	0.546	0.576	0.553	0.579	0.567	0.564
LSTM (no w2v)	0.522	0.561	0.527	0.531	0.511	0.530
LSTM (use w2v)	**0.554**	0.568	0.585	0.587	0.558	0.571
BLSTM (no w2v)	0.541	0.585	0.583	0.561	0.536	0.561
BLSTM (use w2v)	0.551	0.586	0.586	**0.592**	0.562	0.576

Table 5: Classification results for Japanese argumentation structures. Bold font indicates the highest score in each column.

	Auto driving	Countryside	Hokkaido	Bread	TDR	Average
baseline	0.515	0.508	0.504	0.498	0.498	0.505
SVM (uni + bi + tri)	0.586	0.592	0.606	0.581	0.595	0.592
SVM (uni + uni pair + bi pair)	**0.622**	0.611	0.642	0.626	0.630	0.626
ERT (uni + bi + tri)	0.617	**0.627**	0.637	**0.626**	**0.638**	**0.629**
MNB (uni + bi + tri)	0.589	0.591	0.596	0.606	0.592	0.595
MNB (uni + uni pair + bi pair)	0.592	0.602	0.598	0.601	0.603	0.599
CBOW (no w2v)	0.569	0.584	0.613	0.575	0.582	0.585
CBOW (use w2v)	0.579	0.594	0.595	0.569	0.592	0.586
LSTM (no w2v)	0.575	0.589	0.604	0.578	0.582	0.586
LSTM (use w2v)	0.603	0.616	0.624	0.586	0.609	0.608
BLSTM (no w2v)	0.576	0.593	0.614	0.586	0.607	0.595
BLSTM (use w2v)	0.608	0.625	**0.646**	0.596	0.617	0.618

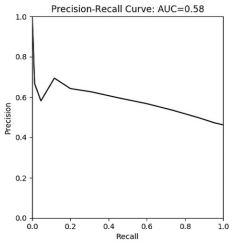


Figure 7: Precision-Recall curve of method that had highest average score in English

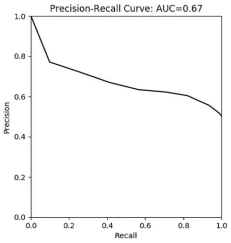


Figure 8: Precision-Recall curve of method that had highest average score in Japanese

ation process is carried out manually. As preliminary work, we are conducting research on estimating the support/non-support relationship between two statements by using machine learning methods and our large-scale argumentation structures as training data.

4.1. Classification Methods

We classified the relationship between two statements as support or non-support. The data were pairs of statements: a node that has a statement directly connected to another node that has the other statement. We examined the following machine learning methods:

Support Vector Machine (SVM) Two statements were represented as a feature vector. Then, the vector was input to a linear SVM. See below for how we created the feature vectors.

Extremely Randomized Tree (ERT) Two statements were represented as a feature vector. Then, the vector was input to an ERT.

Multinomial Naive Bayes (MNB) Two statements were represented as a feature vector. Then, the vector was input to an MNB.

Continuous Bag-of-Words (CBOW) Each statement was embedded, and each vector was averaged. Then two vectors were concatenated, and their relationship was estimated using the softmax function.

Long Short-Term Memory (LSTM) Each statement was embedded, and each vector was input to LSTM. Then two vectors were concatenated, and their relationship was estimated.

Bidirectional LSTM (BLSTM) Each statement was embedded, and each vector was input to BLSTM. Then two vectors were concatenated, and their relationship was estimated.

Two sets of features were used for SVM and MNB; (1) the word uni-, bi-, and tri-grams of statements (uni + bi + tri) and (2) the word uni-grams of statements and word uni-, and bi-grams pairs between two statements (uni + uni

pair + bi pair). The word uni-, bi-, and tri-grams of statements were used for ERT. On the other hand, a sequence of one-hot word vectors were used for CBOW, LSTM, and BLSTM. These methods used two ways to embed each statement; the pre-trained word2vec model (use w2v) and not using this model (no w2v). The Japanese word2vec was learned from the data of Wikipedia, while the English word2vec was learned from the data of Google News. The classifiers were trained using pairs of statements in four argumentation structures written in the same language then were tested using pairs of statements in the other argumentation structure. Specifically, for training in CBOW, LSTM, and BLSTM, training data were divided into two parts; for training (90%) and development to tune the parameters (10%).

4.2. Results

Table 4 lists the accuracy of the support/non-support classification for English structures, and Table 5 lists the accuracy of the support/non-support classification for Japanese structures. In Japanese and English, an extremely randomized tree method with uni-, bi-, tri-grams had the highest score. The score for Japanese was higher than that for English. This is because in Japanese, there seems to be more linguistic constructs that denote/infer discourse or logical relationships.

In Japanese and English, we confirmed that both highest scoring methods could classify the relationships more accurately than each baseline (McNemar's test, Japanese: $p < .001$, English: $p < .001$).

However, as shown in Figures. 7 and 8, the value of recall is very small in the high precision region (more than 90%). Therefore, we consider it is currently difficult to automatically augment argumentation structures. The progress in the field of argumentation mining would help in the future (Lippi and Torroni, 2016).

5. Conclusion

We created large-scale argumentation structures for dialogue systems. We compared the structures for their characteristics to conventional argumentation structures and usage of schemes between English and Japanese. We conducted a subjective evaluation of dialogue generation by using the argumentation structures and described the development of a dialogue system prototype. For automatic augmentation of such structures, we conducted an experiment of automatic support/non-support classification. For future work, we will develop a method for automatically constructing such structures. We will also evaluate a dialogue system prototype and improve the dialogue system; for example, we will develop dialogue strategies for natural argumentation.

6. Bibliographical References

Bui, T. H., Frampton, M., Dowding, J., and Peters, S. (2009). Extracting decisions from multi-party dialogue using directed graphical models and semantic similarity. In *Proc. SIGDIAL*, pages 235–243.

Fernández, R., Frampton, M., Ehlen, P., Purver, M., and Peters, S. (2008). Modelling and detecting decisions in multi-party dialogue. In *Proc. SIGDIAL*, pages 156–163.

Gordon, T. F., Prakken, H., and Walton, D. (2007). The carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10):875–896.

Higashinaka, R., Sakai, K., Sugiyama, H., Narimatsu, H., Arimoto, T., Fukutomi, T., Matsui, K., Ijima, Y., Ito, H., Araki, S., Yoshikawa, Y., Ishiguro, H., and Matsuo1, Y. (2017). Argumentative dialogue system based on argumentation structures. In *Proc. SemDial*, pages 154–155.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., and Stolcke, A. (2003). The ICSI meeting corpus. In *Proc. ICASSP*, volume 1, pages 364–367.

Lawrence, J., Bex, F., Reed, C., and Snaith, M. (2012). AIFdb: Infrastructure for the argument web. In *Computational Models of Argument (COMMA)*, pages 515–516.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology*, 16(2):10:1–10:25.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proc. NIPS*, pages 3111–3119.

Reed, C. and Rowe, G. (2004). Araucaria: software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13:961–980.

Renals, S., Hain, T., and Bourlard, H. (2007). Recognition and understanding of meetings the AMI and AMIDA projects. In *Proc. ASRU*, pages 238–247.

Rosenthal, S. and McKeown, K. (2012). Detecting opinionated claims in online discussions. In *Proc. 2012 IEEE Sixth International Conference on Semantic Computing*, pages 30–37.

Snaith, M., Lawrence, J., and Reed, C. (2010). Mixed initiative argument in public deliberation. In *Proc. 4th International Conference on Online Deliberation*, pages 2–13.

Toulmin, S. E. (1958). *The uses of argument*. Cambridge university press.

Walton, D. (1996). *Argumentation schemes for presumptive reasoning*. Routledge.

Walton, D. (2013). *Methods of argumentation*. Cambridge University Press.

Yanai, K., Kobayashi, Y., Sato, M., Yanase, T., Miyashi, T., Niwa, Y., and Ikeda, H. (2016). Debating artificial intelligence. *Hitachi Review*, 65(6):151.

Exploring Conversational Language Generation for Rich Content about Hotels

Marilyn A. Walker[1], **Albry Smither**[2], **Shereen Oraby**[1], **Vrindavan Harrison**[1], **Hadar Shemtov**[3]
University of California Santa Cruz[1], Google Content Studio[2], and Google Research[3]

Abstract

Dialogue systems for hotel and tourist information have typically simplified the richness of the domain, focusing system utterances on only a few selected attributes such as price, location and type of rooms. However, much more content is typically available for hotels, often as many as 50 distinct instantiated attributes for an individual entity. New methods are needed to use this content to generate natural dialogues for hotel information, and in general for any domain with such rich complex content. We describe three experiments aimed at collecting data that can inform an NLG for hotels dialogues, and show, not surprisingly, that the the sentences in the original written hotel descriptions provided on webpages for each hotel are stylistically not a very good match for conversational interaction. We quantify the stylistic features that characterize the differences between the original textual data and the collected dialogic data. We plan to use these in stylistic models for generation, and for scoring retrieved utterances for use in hotel dialogues.
KEYWORDS: dialogue, conversation, natural language generation, hotels domain.

1. Introduction

Research and advanced development labs in both industry and academia are actively building a new generation of conversational assistants, to be deployed on mobile devices or on in-home smart speakers, such as Google Home. None of these conversational assistants can currently carry on a coherent multi-turn conversation in support of a complex decision task such as choosing a hotel, where there are many possible options and the user's choice may involve making trade-offs among complex personal preferences and the pros and cons of different options.

For example, consider the hotel description in the InfoBox in Figure 1, the search result for the typed query *"Tell me about Bass Lake Taverne"*. These descriptions are written by human writers within Google Content Studio and cover more than 200 thousand hotels worldwide. The descriptions are designed to provide travelers with quick, reliable and accurate information that they may need when making booking decisions, namely a hotel's amenities, property, and location. The writers implement many of the decisions that a dialogue system would have to make: they make decisions about content selection, content structuring, attribute groupings and the final realization of the content (Rambow and Korelsky, 1992). They access multiple sources of information, such as user reviews and the hotels' own web pages. The descriptions cannot be longer than 650 characters and are optimized for visual scanning. There is currently no method for delivering this content to users via a conversation other than reading the whole InfoBox aloud, or reading individual sections of it.

Structured data is also available for each hotel, which includes information about the setting of a hotel and its grounds, the feel of the hotel and its rooms, points of interest nearby, room features, and amenities such as restaurants and swimming pools. Sample structured data for the Bass Lake Taverne is in Figure 2.[1] The type of information available in the structured data varies a great deal according to the type of hotel: for specialized hotels it includes highly distinctive low-frequency attributes for look-and-feel such as "feels swanky" "historical rooms" or amenities such as "direct access to beach", "has hot tubs", or "ski-in, ski-out".

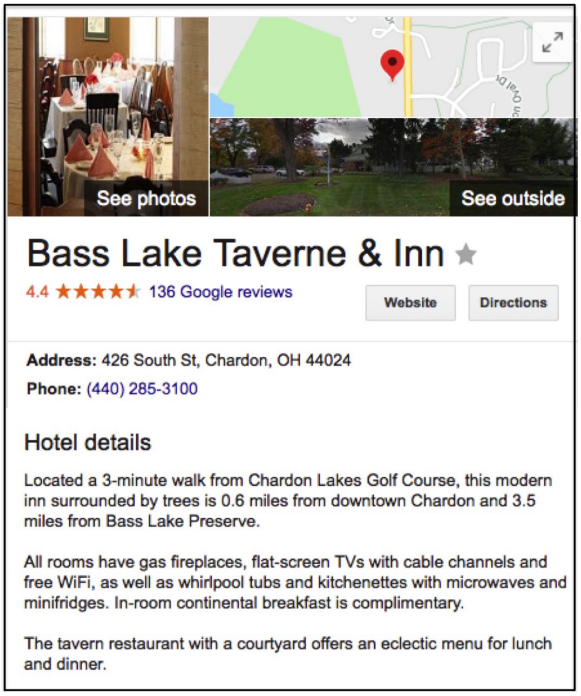


Figure 1: InfoBox Hotel Description for Bass Lake Taverne

Research on dialogue systems for hotel information has existed for many years, in some cases producing shared dialogue corpora that include hotel bookings (Devillers et al., 2004; Walker et al., 2002; Rudnicky et al., 1999; Villaneau and Antoine, 2009; Bonneau-Maynard et al., 2006; Hastie et al., 2002; Lemon et al., 2006). Historically, these systems have greatly simplified the richness of the domain, and supported highly restricted versions of the hotel booking task, by limiting the information that the system can talk about to a small number of attributes, such as location, number of stars, room type, and price. Data collection involved users being given specific tasks where they simply had to find a hotel in a particular location, rather than satisfy the complex preferences that users may have booking hotels. This reduction in content simplifies the decisions

[1]The publicly available Yelp dataset[2] has around 8,000 entries for US hotels, providing around 80 unique attributes.



Figure 2: Sample of Hotel Structured Data for "Bass Lake Taverne"

that a dialogue manager has to make, and it also reduces the complexity of the natural language generator, since a few pre-constructed templates may suffice to present the small number of attributes that the system knows about. It is also important to note that the challenges for the hotel domain are not unique. Dialogue systems for movies, weather reports, real estate and restaurant information also have access to rich content, and yet previous work and current conversational assistants reduce this content down to just a few attributes.

This paper takes several steps toward solving the challenging problem of building a conversational agent that can flexibly deliver richer content in the hotels domain. Section 2. first reviews possible methods that could be applied, and describes several types of data collection experiments that can inform an initial design. After motivating these data collection experiments, the rest of the paper describes them and their results (Section 3., Section 4., and Section 5.). Our results show, not surprisingly, that both the hotel utterances and the complete dialogues that we crowdsource are very different in style than the original written InfoBox hotel descriptions. We compare different data collection methods and quantify the stylistic features that characterize their differences. The resulting corpora are available at `nlds.soe.ucsc.edu/hotels`.

2. Background and Experimental Overview

Current methods for supporting dialogue about hotels revolve either around search or around using a structured dialogue flow. Neither of these methods on their own support fully natural dialogue, and there is not yet an architecture for conversational agents that flexibly combines unstructured information, such as that found in the InfoBox or in reviews or other textual forms, and structured information such as that in Figure 2.

Search methods could focus on the content in the current InfoBox, and carry out short (1-2 turn) conversations by applying compression techniques on sentences to make them more conversational (Andor et al., 2016; Krause et al., 2017b). For example, when asked "Tell me about Bass Lake Taverne", Google Home currently produces an utterance providing its location and how far it is from the user's location. When asked about hotels in a location, Google Home reads out parts of the information in the Infobox, but it does not engage in further dialogue that explores individual content items. Moreover, the well-known differences between written and oral language (Biber, 1991) means that selected spans from written descriptions may not sound natural when spoken in conversation, and techniques may be needed to adapt the utterance to the dialogic context. Our first experiment, described in Section 3. asks crowdworkers to (1) indicate which sentences in the InfoBox are most important, and (2) write dialogic paraphrases for the selected sentences in order to explore some of these issues.

Another approach is to train an end-to-end dialogue system for the hotels domain using a combination of simulation, reinforcement learning and neural generation methods (Nayak et al., 2017; Shah et al., 2018; Liu et al., 2017; Gašić et al., 2017). This requires first developing a user-system simulation to produce simulated dialogues, crowdsourcing utterances for each system and user turn, and then using the resulting data to (1) optimize the dialogue manager using reinforcement learning, (2) train the natural language understanding from the user utterances, and (3) train the natural language generation from the crowd-sourced system utterances. Currently however it is not clear how to build a user-system simulation for the hotels domain that would allow more of the relevant content to be exchanged, and there are no corpora available with example dialogue flows and generated utterances.

To build a simulation for such complex, rich content, we first need a model for how the dialogue manager (DM) should (1) order the content across turns, and (2) select and group the content in each individual turn. Our assumption is that the most important information should be presented earlier in the dialogue, so one way to do this is to apply methods for inducing a *ranking* on the content attributes. Previous work has developed a model of user preferences to solve this problem (Carenini and Moore, 2000), and shown that users prefer systems whose dialogue behaviors are based on such customized content selection and presentation (Stent et al., 2002; Polifroni et al., 2003; Walker et al., 2007). These preferences (ranking on attributes) can be acquired directly from the user, or can be inferred from their past behavior. Here we try two other methods. First, in Section 3., we ask Turkers to select the most important sentence from the InfoBox descriptions. We then tabulate which attributes are in the selected sentences, and use this to induce a ranking. After using this tabulation to collect additional conversational utterances generated from meaning representations (Section 4.), we carry out an additional experiment (Section 5.) where we collect whole dialogues simulating the exchange of information between a user and a conversational agent, given particular attributes to be communicated. We report how information is ordered and grouped across these dialogues.

An end-to-end training method also needs a corpus for training for the Natural Language Generator (NLG). Thus we also explore which crowdsourcing design yields the best conversational utterance data for training the NLG. Our first experiment yields conversationalized paraphrases that match the information in individual sentences in the original Infobox. Our second experiment (Section 4.) uses content selection preferences inferred from the paraphrase experiment and collects utterances generated to match meaning representations. Our third experiment (Section 5.), crowdsources whole dialogues for selected hotel attributes: the utterances collected using this method are sensitive to the context while the other two methods yield utterances that can be used out of context.

To measure how conversational our collected utterances are, we build on previous research that counts linguistic features that vary across different situations of language use (Biber, 1991), and tabulates the effect of variables like the mode of language as well as its setting. We use the linguistic features tabulated by the Linguistic Inquiry and Word Count (LIWC) tool (Pennebaker et al., 2015). See Table 1. We select features to pay attention to using the counts provided with the LIWC manual that distinguish natural speech (Column 4) from articles in the New York Times (Column 5). Our hotel descriptions are not an exact genre match to the New York Times, but they are editorial in nature. For example, Table 1 shows that spoken conversation has shorter, more common words (Sixltr), more function words, fewer articles and prepositions, and more affective and social language.

Category	Abbrev	Examples	Speech	NYT
Summary Language Variables				
Words/sentence	WPS	-	-	21.9
Words >6 letters	Sixltr	-	10.4	23.4
Linguistic Dimensions				
Total function words	funct	it, to, no, very	56.9	42.4
Total pronouns	pronoun	I, them, itself	20.9	7.4
Personal pronouns	ppron	I, them, her	13.4	3.6
1st pers singular	i	I, me, mine	7.0	.6
2nd person	you	you, your, thou	4.0	.3
Impersonal pronouns	ipron	it, it's, those	7.5	3.8
Articles	article	a, an, the	4.3	9.1
Prepositions	prep	to, with, above	10.3	14.3
Auxiliary verbs	auxverb	am, will, have	12.0	5.1
Common Adverbs	adverb	very, really	7.7	2.8
Conjunctions	conj	and, but, whereas	6.2	4.9
Negations	negate	no, not, never	2.4	.6
Common verbs	verb	eat, come, carry	21.0	10.2
Psychological Processes				
Affective processes	affect	happy, cried	6.5	3.8
Social processes	social	mate, talk, they	10.4	7.6
Cognitive processes	cogproc	cause, know, ought	12.3	7.5
Other				
Affiliation	affiliation	friend, social	2.0	1.7
Present focus	focuspresent	today, is, now	15.3	5.1
Informal language	informal	-	7.1	0.3
Assent	assent	agree, OK, yes	3.3	0.1
Nonfluencies	nonflu	er, hm, umm	2.0	0.1
Fillers	filler	Imean, youknow	0.5	0.0

Table 1: LIWC Categories with Examples and Differences between Natural Speech and the New York Times

The experiments use Turkers with a high level of qualification and we ensure that Turkers make at least minimum wage on our tasks. For the paraphrase and single-turn HITs for properties and rooms, we ask for at least 90% approval rate and at least 100 (sometimes 500) HITs approved and we always do location restriction (English speaking locations). For the dialog HITs we paid 0.9 per HIT, and restricted Turkers to those with a 95% acceptance rate and at least 1000 HITs approved. We also elicited the dialogs over multiple rounds and excluded Turkers who had failed to include all 10 attributes on previous HITs.

We present a summary of all of our experiments in Table 3 and then discuss the relevant columns in each section. A scan of the whole table is highly informative however, because Biber (1991) makes the point that differences across language use situations are not dichotomous, i.e. there is not one kind of oral language and one kind of written language. Rather language variation occurs continuously and on a scale, so that language can be "more or less" conversational. The overall results in Table 3 demonstrates this scalar variation, with different methods resulting in more or less conversationalization of the content in each utterance.

3. Paraphrase Experiment

The overall goal of the Paraphrase experiment is to evaluate the differences between monologic and dialogic content that contain the same or similar information. These experiments are valuable because the original content is given in unordered lists that facilitate visual scanning, as opposed to a conversation in which the dialogue system needs to decide the order in which to present information and whether to leave some information out.

We ask Turkers to both select "the most important" content out of the hotel descriptions, and then to paraphrase that content in a conversational style. We use this data to induce an importance ranking on content and we also measure how the conversational paraphrases of that content differ from the original phrasing. We used a randomly selected set of 1,000 hotel descriptions from our corpus of 200K, with instructions to Turkers to:

- Select the sentence out of the description that has the most important information to provide in response to a user query to "tell me about HOTEL-NAME".
- Cut and paste that sentence into the "Selected Sentence" box.
- Rewrite your selected sentence so that it sounds conversational, as a turn in dialogue. You may need to reorder the content or convert your selected sentence to multiple sentences in order to make it sound natural.

For each of the descriptions, three Turkers performed this HIT, yielding a total of 3,000 triples consisting of the original description, the selected sentence, and the human-generated dialogic paraphrases. For example, for the hotel description in Figure 3, two Turkers selected S1 and the other selected S4. These sentences have different content, so for each attribute realized we increase its count as part of our goal to induce a ranking indicating the importance of different attributes. The dialogic paraphrases the same Turkers produced are shown in Figure 4. The paraphrases contain fewer words per sentence, more use of anaphora, and more use of subjective phrases taking the listener's perspective such as *you will enjoy*.

S1	The elegant rooms, decorated in warm tones, feature high ceilings and lots of natural light, plus Turkish marble bathrooms, Bose sound systems, HDTVs and designer toiletries; some have views of the park.
S2	Suites include living rooms and soaking tubs; some have city views.
S3	Grand suites offer personal butler service.
S4	Open since 1930, this opulent landmark sits across the street from Central Park on New York's famed 5th Avenue.

Figure 3: An InfoBox description for the hotel *The Pierre, A Taj Hotel, New York*, split into sentences and labeled.

T1	This hotel's elegant rooms are decorated in warm tones. They feature high ceilings with lots of natural light. The rooms feature Turkish marble bathrooms, designer toiletries, high-definition televisions and Bose sound systems. Some rooms even offer views of the park.
T2	Located on 5th Avenue, this landmark hotel is located across the street from Central Park and dates back to 1930.
T3	Each room is elegantly decorated in warm tones. You will enjoy high ceilings and natural light. The bathrooms are done in Turkish marble and have designer toiletries. For entertainment, you will find HDTVs and Bose sound systems. There are views of Central Park from some rooms.

Figure 4: Turker generated paraphrases of the hotel description shown in Table 3. The Turkers T1 and T3 selected S1 as containing the most important information and Turker T2 selected S4.

attribute	F
locale_mountain	1.0
has_bed_wall_in_rooms	.67
has_wet_room	.67
feels_quaint	.61
has_crib	.50
feels_artsy	.44
is_whitewashed	.44
has_private_bathroom_outside_room	.44
feels_nautical	.42
has_luxury_bedding	.40
welcomes_children	.39
is_dating_from	.38
feels_retro	.38
all_inclusive	.34
has_casino	.33
has_heated_floor	.33
has_city_views	.33
has_boardwalk	.33
has_hammocks	.33
has_onsite_barbecue_area	.33

Table 2: Turker's Top 20 Attributes, shown with their frequency F of selection when given in the content.

Results. We build a ranked ordering of hotel attribute importance using the selected sentences from each hotel description. We count the number of times each attribute is realized within a sentence selected as being the most informative or relevant. We count the number of hotels for which each attribute applies. The attribute frequency F is given as the number of times an attribute is selected divided by the product of the number of hotels to which the attribute applies and the number of Turkers that were shown those hotel descriptions. Finally, the attributes are sorted and ranked by largest F.

Table 2 illustrates how the tabulation of the Turker's selected sentences provides information on the ranking of attributes that we can use in further experimentation. However, the frequencies reported are conditioned on the relevant attribute being available to select in the Infobox description, and many of the attributes are both low frequency and highly distinctive, e.g. the attribute `local_mountain`. A reliable importance ranking using this method would need a larger sample than 1000 hotels. It is also possible that attribute importance should be directly linked to how distinctive the attribute is, with less frequent attributes always mentioned earlier in the dialogue.

The first three columns of Table 3 summarize the stylistic differences between the original Infobox sentences and the collected paraphrases. Column 3 provides the p-values showing that many differences are statistically significant. Differences that indicate that the paraphrases are more similar to oral language (as in Speech, column 4 of Table 1), include the use of adverbs, words of affiliation, common verbs, and a reduced number of words per sentence. Expected differences that are not realized are in increases in affective and social language, reduced use of Articles and long words (SixLtr), greater use of conjunctions. So while this method improves the conversational style of the content realization, we will see that our other methods produce *more* conversational utterances. While this method is inexpensive and may not require such expert Turkers, the utterances collected may only be useful for systems that do not use structured data and so need paraphrases of the original Infobox data that is more conversational.

4. Generation from Meaning Representations

The second experiment aims to determine whether we get higher quality utterances if we ask crowdworkers to generate utterances directly from a meaning representation, in the context of a conversation, rather than by selecting from the original Infobox hotel descriptions. Utterances generated in this way should not be influenced by the original phrasing and sentence planning in the hotel descriptions.

Instructions for our second experiment are shown in Figure 5. Here we give Turkers specific content tables and ask them to generate utterances that realize that content. Note that the original hotel descriptions, as illustrated in Figure 1 consists of three blocks of content, property, rooms and amenities. For each hotel in a random selection of 200 hotels from the paraphrase experiment, we selected content for both rooms (4 attributes) and properties (6 attributes) by picking the attributes with the highest scores (as illustrated for a small set of attributes in Table 2). Thus hotel has two unique content tables assigned to it, one pertaining to the hotel's rooms, and the other for the hotel grounds. Each hotel content table is given to three Turkers which results

Category	InfoBox	Paraphrase	p-val	Props+Rooms	Dialogues	p-val Props+Rooms vs. Para	p-val Props+Rooms vs. Dial
Impersonal Pronouns	0.97	3.80	0.00	3.36	5.19	0.00	0.00
Adverbs	0.97	3.41	0.00	3.57	6.25	0.12	0.00
Affective Processes	4.98	4.81	0.14	8.09	8.55	0.00	0.26
Articles	8.08	9.06	0.00	11.54	7.62	0.00	0.00
Assent	0.02	0.04	0.00	0.07	1.13	0.11	0.00
Auxiliary Verbs	1.69	6.12	0.00	8.02	11.81	0.00	0.00
Common Verbs	3.64	7.94	0.00	10.97	15.07	0.00	0.00
Conjunctions	8.07	8.13	0.54	7.33	6.52	0.00	0.00
First Person Singular	0.01	0.02	0.00	0.41	3.41	0.00	0.00
Negations	0.03	0.07	0.00	0.27	0.44	0.00	0.00
Personal Pronouns	0.06	1.15	0.00	3.87	10.17	0.00	0.00
Second Person	2.31	0.45	0.00	2.43	5.63	0.00	0.00
Six Letter Words	4.81	19.15	0.00	20.74	15.50	0.00	0.00
Social Processes	16.24	5.63	0.00	8.53	14.66	0.00	0.00
Total Pronouns	1.03	4.94	0.00	7.23	15.36	0.00	0.00
Words Per Sentence	22.86	14.69	0.00	14.52	10.90	0.00	0.00
Affiliation	1.18	0.95	0.00	1.13	5.97	0.00	0.00
Cognitive Processes	2.58	3.11	0.00	9.18	10.47	0.00	0.00
Focus present	3.64	7.91	0.00	9.35	14.40	0.00	0.00
Function	26.79	37.62	0.00	44.58	53.08	0.00	0.00
Informal	0.42	0.35	0.03	0.51	1.76	0.00	0.00
nonflu	0.37	0.28	0.00	0.42	0.63	0.00	0.00
prep	9.65	8.50	0.00	9.83	9.88	0.00	0.72

Table 3: Conversational LIWC features across all Utterance Types/Data Collection Methods

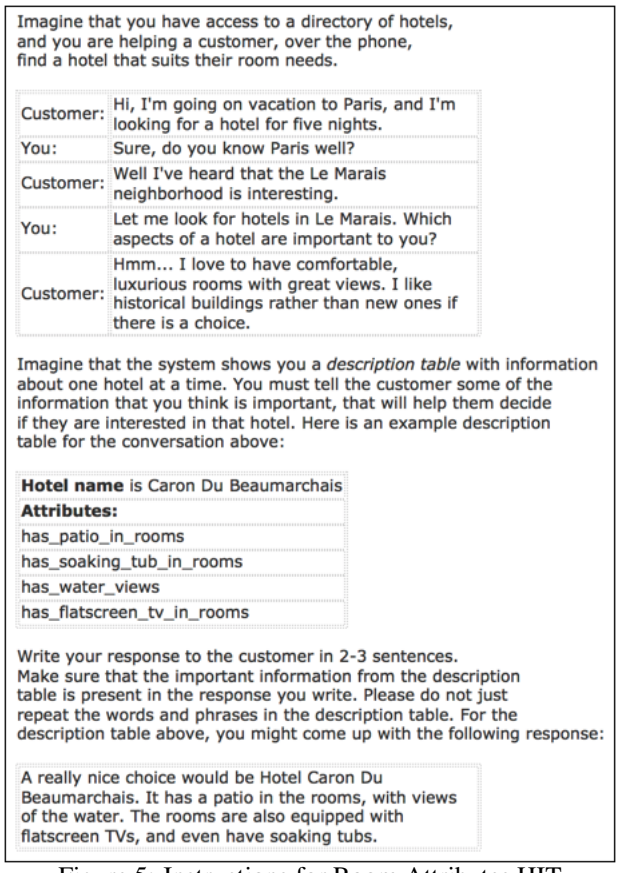


Figure 5: Instructions for Room Attributes HIT

Prop	A good choice is 1 Hotel South Beach in Miami Beach. It's luxurious, lively, upscale, and chic, with beach access and a bar onsite.
Prop	I think that 1 Hotel South Beach will meet your needs. It's a chic luxury hotel with beach access and a bar. Very lively.
Room	One of the excellent hotels Miami Beach has to offer is the 1 Hotel South Beach. The upgraded rooms are full featured, including a kitchen and a desk for work. Each room also has a balcony.
Room	The 1 Hotel South Beach doesn't mess around. When you come to stay here you won't want to leave. Each upgraded room features a sunny balcony and personal kitchen. You also can expect to find a lovely writing desk for your all correspondence needs.

Figure 6: Example utterances generated by Turkers in the second experiment. Turkers were given specific content tables from which to generate dialogue utterances that realize that content.

in a total of 1,200 utterances collected. Turkers were instructed to create utterances as conversational as possible.

Results. Sample utterances for both properties and rooms are shown in Figure 6.

Column 5 in Table 3 shows the frequencies of LIWC's conversationalization features for the utterances collected in this experiment, and Column 7 reports statistical significance (p-values) for comparing these collected utterances to the paraphrases collected in Experiment 1, using an unpaired t-test on the two datasets. We can see that some of the attributes that indicate conversationalization indicate that this method yields more conversational utterances: there are significantly more more auxiliary verbs and common verbs. There is a greater use of first person and second person pronouns, as well as words indicating affective, social and cognitive processes. Counts of function words and focusing on the present are also higher as would be expected of more conversational language.

5. Dialogue Collection Experiment

The final data collection experiment focuses on utterance generation in an explicitly dialogic setting. In order to collect dialogues about our hotel attributes, we employ a technique called "self-dialogue" collection, which to our knowledge was pioneered by Krause et al. (2017a), who claim that the results are surprisingly natural. We ask individual Turkers to write a full dialogue between an agent

and a customer, where the Turker writes both sides of the dialogue. The customer is looking for a hotel for a trip, and the agent has access to a description table with a list of 10 attributes for a single hotel. The agent is tasked with describing the hotel to the customer. Figure 7 shows our HIT instructions that provided a sample dialogue as part of the instructions to the Turkers.

This experiment utilizes 74 unique hotels, a subset of those used in the property and room experiments above (Section 4.), where we have both 6 property attributes, and 4 room attributes. We aimed to collect 3 dialogues per hotel (from unique Turkers), but due to some Turkers failing to follow instructions, the final corpus consists of 205 dialogues (comprised of 58 hotels with 3 dialogues each, 15 hotels with 2 dialogs each, and 1 hotel with only 1 dialogue).

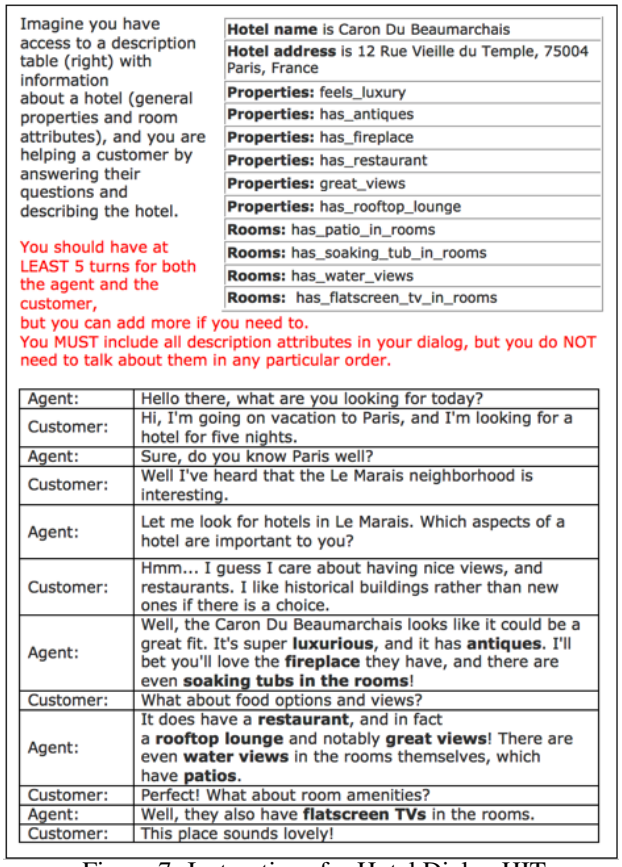


Figure 7: Instructions for Hotel Dialog HIT

Figure 4 and Figure 5 provide sample dialogues from the corpus, with the 10 required attributes shown in bold, and the agent (A) and customer (C) turns shown with their respective turn numbers. In Dialogue 1, we see an example of a creative dialogue where the Turker designs a situational context for the dialogue where the customer is looking for a hotel for a bachelorette party weekend, and has specific requirements about flooring and amenities. We note that in this dialogue, the agent only begins to discuss the hotel in their third turn. In Dialogue 2, we see a much more basic dialogue, where the agent begins to list properties and room attributes earlier on the in dialogue (at Agent Turn 2), and the full list of attributes is exhausted halfway through the conversation, at Agent Turn 3.

Results. We begin by analyzing information that both the dialogue manager and the natural language generator would need to know, namely how frequently attributes are grouped in a single turn in our collected dialogues, by counting the number of times certain keywords are mentioned related to the attributes in the dialogues. Table 6 shows attributes groups that occur at least 4 times in the dataset, showing the group of attributes and the frequency count. We note that

A1	Hi! How can I help you today?
C1	I am planning a trip to New York for a bachelorette party weekend and need help finding a hotel.
A2	OK, what will you girls be planning to do?
C2	We're going to a Broadway show but other than that just going to dinner and hitting some bars.
A3	OK, I think the Hotel Indigo in the Chelsea section would be great! It's **upscale** and has a great **hip** and **contemporary** vibe with that **buzzy** New York City energy feel.
C3	That sounds like what we're looking for. I know this is a weird question but one of the girls sometimes has an allergy to carpet, is there any type of option for a non carpeted room?
A4	Actually, this hotel has **hardwood floors in the rooms**.
C4	Great! I think we may be bringing some snacks and maybe some of our own alcohol. Can we arrange for a fridge or do they just have ice buckets?
A5	The rooms have **mini bars** as well as **coffee** if you girls need some help waking up for your time out. There is also a **bar** on site so you can start the party before you even head out.
C5	Great! One more question, one of the girls does need to keep in touch with work. Do you offer WiFi?
A6	The hotel has **desks in every room** and offers a **business center** if she needs anything like a printer or a desktop computer.
C6	I think we'll go ahead and book this. It sounds perfect!

Table 4: Situational Context for Content Hotel Dialogue

A1	Good evening, how can i help you?
C1	I am looking for a good hotel to have a business conference in the Brooklyn area.
A2	Sure, let me see what i can find. Hotel Indigo Brooklyn may be just what you are looking for. It has a **hip** feel with an **onsite bar**, **Business center**, **restaurant**, **free wifi**. Its got it all.
C2	That sounds excellent. What room amenities are offered?
A3	There is **coffee** in the rooms and a **mini fridge**. All the rooms have been recently **upgraded** and did i mention it has a **fitness room**? I has full **room service** as well.
C3	Wow, that sounds great.Whats the address? I need to make sure its in the right area for me.
A4	Sure, its 229 Duffield Street, Brooklyn, NY 11201, USA.
C4	Thanks, thats just the right spot. Go ahead and make me a reservation for next Tuesday.
A5	excellent! Its done!
C5	Thanks you have been extremely helpful!

Table 5: Straightforward Attribute Listing Hotel Dialogue

the attributes within the groups are generally either: 1) semantically similar, e.g. "modern" and "contemporary"; 2) describe the same aspect, e.g. "feels elegant" and "feels upscale"; or 3) describe the same general attribute, e.g. "has breakfast buffet", "has free breakfast", and "has free breakfast buffet". It is interesting to note that the semantic similarity is not always completely obvious (for example, "has balcony in rooms" and "has fireplace" may be used to emphasize more luxurious amenities that are a rare find).

Attribute Group	Count
(has_business_center, has_meeting_rooms)	13
(has_bar_onsite, has_restaurant)	9
(feels_contemporary, feels_modern)	9
(has_bar_onsite, has_business_center)	8
(has_business_center, has_desk_in_rooms)	7
(feels_casual, feels_contemporary, feels_modern)	6
(feels_modern, has_business_center)	6
(has_business_center, has_convention_center)	6
(feels_elegant, feels_upscale)	4
(has_bar_onsite, has_bar_poolside)	4
(has_microwave_in_rooms, has_minifridge_in_rooms)	4
(feels_contemporary, feels_elegant, feels_modern)	4
(feels_contemporary, feels_upscale)	4
(has_balcony_in_rooms, has_fireplace)	4
(feels_chic, feels_upscale)	4
(has_business_center, has_desk_in_rooms, has_wi_fi_free)	4
(has_breakfast_buffet, has_free_breakfast, has_free_breakfast_buffet)	4
(has_coffee_in_rooms, has_desk_in_rooms, has_microwave_in_rooms, has_minifridge_in_rooms)	4

Table 6: Attributes Frequently Grouped in a Single Turn

Our assumption is that more important attributes should be presented earlier in the dialogue, and that a user-system dialogue simulation system design (Shah et al., 2018; Liu et al., 2017; Gašić et al., 2017) would require such information to be available. Thus, in order to provide more information on the importance of particular attributes, we analyze where in the conversation (i.e. first or second half) certain types of attributes are mentioned. For example, we observe that attributes describing the "feel", such as "feels chic" or "feels upscale", are mentioned around 700 times, and that for 80% of those times they appear in the first half of the conversation as opposed to the second half, showing that they are often used as general hotel descriptors before diving into detailed attributes. Attributes describing room amenities on the other hand, such as "has kitchen in rooms" or "has minifridge", were mentioned around 530 times, with a more even distribution of 53% in the first half of the conversation, and 47% in the second half.

We also observe that most attributes are first introduced into the conversation by the agent, but that a small number of attributes are more frequently first introduced by the customer, specifically: *has_swimming_pool_indoor, popular_with_business_travelers, has_onsite_laundry, welcomes_families, has_convention_center, has_ocean_view, has_free_breakfast_buffet, has_swimming_pool_saltwater*.

Next, we compare our collected dialogues to the single-turn dialogue descriptions described in Section 4.). Specifically, we focus on the "agent" turns of our dialogues, as they are more directly comparable to the property and room turns. Table 7 describes the average number of turns, number of sentences per turn, words per turn, and attributes per turn across the property, room, and agent dialogue turns. We note that the average number of sentences, words, and attributes per turn for our property and room descriptions are higher in general than the agent turns in our dialogues, because the dialogues allow the agent to distribute the required content across multiple turns.

	Properties	Rooms	Dialogues
Number of turns	600	600	1227
Sentences per turn			
Average	2.80	2.55	1.80
Words per turn			
Average	41.37	39.81	21.45
Attributes per turn			
Average	6	4	1.62

Table 7: Comparing Property, Room, and Agent Dialogue Turns

Column 6 (Dialogues) of Table 3 reports the frequencies for conversational features in the collected data, with p-values in Column 8 comparing the dialogic utterances to the property+room utterances collected in Experiment 2 (Section 4.. The dialogic data collection results in utterances that are more conversational according to these counts, with higher use of impersonal pronouns and adverbs, auxiliary verbs and common verbs, and first person and second person pronouns. We also see increases in words indicative of affective, social and cognitive processes, more informal language, and reduced use of Six Letter words, fewer words per sentence and greater use of language focused on the present. Thus these utterances are clearly much more conversational, and provide information on attribute ordering across turns as well as possible ways of grouping attributes. The utterances collected in this way might also be useful for template induction, especially if the induced templates could be indexed for appropriate use in context.

6. Conclusion and Future Work

This paper presents a new corpus that contributes to defining the requirements and provide training data for a conversational agent that can talk about all the rich content available in the hotel domain. All of the data we collect in all of the experiments is available at `nlds.soe.ucsc.edu/hotels`. After completing three different types of data collection, we posit that the self-dialogue collection might produce the best utterances but at the highest cost, with the most challenges for direct re-use. The generation from meaning representations produces fairly high quality utterances, but they are not sen-

sitive to the context, and our results from the dialogic collection suggest that it might be useful to collect additional utterances using this method that sample different combinations of attributes, and select fewer attributes for each turn.

In future work, we plan to use these results in three different ways. First, we can train a "conversational style ranker" based on the data we collected, so that it can retrieve pre-existing utterances that have good conversational properties. The features that this ranker will use are the linguistic features we have identified so far, as well as new features we plan to develop related to context. Second, we will experiment directly using the collected utterances in a dialogue system, first by templatizing them by removing specific instantiations of attributes, and then indexing them for their uses in particular contexts.

7. References

Andor, D., Alberti, C., Weiss, D., Severyn, A., Presta, A., Ganchev, K., Petrov, S., and Collins, M. (2016). Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 2442–2452.

Biber, D. (1991). *Variation across speech and writing*. Cambridge Univ Pr.

Bonneau-Maynard, H., Ayache, C., Bechet, F., Denis, A., Kuhn, A., Lefèvre, F., Mostefa, D., Quignard, M., Rosset, S., Servan, C., et al. (2006). Results of the french evalda-media evaluation campaign for literal understanding. In *The fifth international conference on Language Resources and Evaluation (LREC 2006)*.

Carenini, G. and Moore, J. (2000). A strategy for generating evaluative arguments. In *Proc. of the 1st International Conference on Natural Language Generation (INLG-00)*, Mitzpe Ramon, Israel.

Devillers, L., Maynard, H., Rosset, S., Paroubek, P., McTait, K., Mostefa, D., Choukri, K., Charnay, L., Bousquet, C., Vigouroux, N., et al. (2004). The french media/evalda project: the evaluation of the understanding capability of spoken language dialogue systems. In *LREC*.

Gašić, M., Hakkani-Tür, D., and Celikyilmaz, A. (2017). Spoken language understanding and interaction: machine learning for human-like conversational systems. *Computer Speech and Language 46*, pages 249 – 251.

Hastie, H. W., Prasad, R., and Walker, M. A. (2002). Automatic evaluation: Using a date dialogue act tagger for user satisfaction and task completion prediction. In *LREC*.

Kittredge, R., Korelsky, T., and Rambow, O. (1991). On the need for domain communication knowledge. *Computational Intelligence*, 7(4):305–314.

Krause, B., Damonte, M., Dobre, M., Duma, D., Fainberg, J., Fancellu, F., Kahembwe, E., Cheng, J., and Webber, B. (2017a). Edina: Building an open domain socialbot with self-dialogues. *arXiv preprint arXiv:1709.09816*.

Krause, S., Kozhevnikov, M., Malmi, E., and Pighin, D. (2017b). Redundancy localization for the conversationalization of unstructured responses. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 115–126.

Lavoie, B. and Rambow, O. (1997). A fast and portable realizer for text generation systems. In *Proc. of the Third Conference on Applied Natural Language Processing, ANLP97*, pages 265–268.

Lemon, O., Georgila, K., and Henderson, J. (2006). Evaluating effectiveness and portability of reinforcement learned dialogue strategies with real users: the talk towninfo evaluation. In *Spoken Language Technology Workshop, 2006. IEEE*, pages 178–181. IEEE.

Liu, B., Tur, G., Hakkani-Tur, D., Shah, P., and Heck, L. (2017). End-to-end optimization of task-oriented dialogue model with deep reinforcement learning. *arXiv preprint arXiv:1711.10712*.

Nayak, N., Hakkani-Tur, D., Walker, M., and Heck, L. (2017). To plan or not to plan? discourse planning in slot-value informed sequence to sequence models for language generation. In *Proc. of Interspeech 2017*.

Pennebaker, J. W., Boyd, R. L., Jordan, K., and Blackburn, K. (2015). The development and psychometric properties of liwc2015. Technical report.

Polifroni, J., Chung, G., and Seneff, S. (2003). Towards the automatic generation of mixed-initiative dialogue systems from web content. In *EUROSPEECH, European Conference on Speech Processing*.

Rambow, O. and Korelsky, T. (1992). Applied text generation. In *Proc. of the Third Conference on Applied Natural Language Processing, ANLP92*, pages 40–47.

Rudnicky, A., Thayer, E., Constantinides, P., Tchou, C., Shern, R., Lenzo, K., Xu, W., and Oh, A. (1999). Creating natural dialogs in the carnegie mellon communicator system. In *Eurospeech*, pages 1531–1534.

Shah, P., Hakkani-Tür, D., and Heck, L. (2016). Interactive reinforcement learning for task-oriented dialogue management. In *NIPS 2016 Deep Learning for Action and Interaction Workshop*.

Shah, P., Hakkani-Tür, D., Tür, G., Rastogi, A., Bapna, A., Nayak, N., and Heck, L. (2018). Building a conversational agent overnight with dialogue self-play. *arXiv preprint arXiv:1801.04871*.

Stent, A., Walker, M., Whittaker, S., and Maloor, P. (2002). User-tailored generation for spoken dialogue: An experiment. In *ICSLP*.

Villaneau, J. and Antoine, J.-Y. (2009). Deeper spoken language understanding for man-machine dialogue on broader application domains: a logical alternative to concept spotting. In *Proceedings of the 2nd Workshop on Semantic Representation of Spoken Language*, pages 50–57. Association for Computational Linguistics.

Walker, M., Rudnicky, A., Aberdeen, J., Bratt, E., Garofolo, J., Hastie, H., Le, A., Pellom, B., Potamianos, A., Passonneau, R., Prasad, R., Roukos, S., Sanders, G., Seneff, S., and Stallard, D. (2002). DARPA communicator evaluation: Progress from 2000 to 2001. In *ICSLP*.

Walker, M. A., Stent, A., Mairesse, F., and Prasad, R. (2007). Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research (JAIR)*, 30:413–456.

Identification of Personal Information Shared in Chat-Oriented Dialogue

Sarah Fillwock*, **David Traum**†
*Michigan State University
428 S. Shaw Lane, East Lansing, MI 48824
fillwoc2@msu.edu

†Institute for Creative Technologies, University of Southern California
12015 Waterfront Dr, Playa Vista, CA 90094
traum@ict.usc.edu

Abstract

We present an analysis of how personal information is shared in chat-oriented dialogue. We develop an annotation scheme, including entity-types, attributes, and values, that can be used to annotate the presence and type of personal information in these dialogues. A collection of attribute types is identified from the annotation of three chat-oriented dialogue corpora and a taxonomy of personal information pertinent to chat-oriented dialogue is presented. We examine similarities and differences in the frequency of specific attributes in the three corpora and observe that there is much overlap between the attribute types which are shared between dialogue participants in these different settings. The work presented here suggests that there is a common set of attribute types that frequently occur within chat-oriented dialogue in general. This resource can be used in the development of chat-oriented dialogue systems by providing common topics that a dialogue system should be able to talk about.

Keywords: dialogue, personal information, attributes, knowledge base, chat-oriented, user model

1. Introduction

Spoken dialogue has been studied from the perspectives of many different disciplines, with a primary focus on analyzing the turn-taking procedure of dialogue, the effects of social context, and the underlying structure of sequences of dialogue (Eggins and Slade, 1997). Within this body of work, spoken dialogue is often divided into two major categories: task-oriented dialogues and chat-oriented dialogues. Although task-oriented dialogues have received the most attention historically, interest in chat-oriented dialogue has been on the rise in recent decades. One major factor in this increase of interest has been the popularity of artificial conversational dialogue agents.

As defined by Eggins and Slade (1997), chat-oriented dialogue - or casual conversation as they name it - is characterized by its topic flexibility, the informal nature of the exchange, and the fact that the participants are not trying to accomplish any particular functional task through their dialogue. Aiming to overcome the seeming open-domain generality of chat-oriented dialogues, much of the effort has been placed into identifying the sub-types of chat-oriented dialogue and characterizing the general structure of these sub-types (Eggins and Slade, 1997; Slade and Gardner, 1993; de Silva Joyce and Slade, 2000).

Although there has been much work on defining and analyzing the characteristic structure of chat-oriented dialogues, there has been less work on exploring the content of these dialogues in terms of their characteristic topics. An understanding of the topics that are commonly focused on during chat-oriented dialogues would provide both greater insight into the function of chat-oriented dialogues on an inter-personal level and also allow for interesting analyses based on the popular topics of conversation. One promising area of inquiry is the sharing of personal information between the conversational participants.It has been observed that people frequently focus on sharing personal information about themselves in chat-oriented dialogues (Mitsuda et al., 2017). Consequently, this provides evidence that focusing on topics related to personal information will cover a significant portion of topics relevant to chat-oriented dialogue.

Our goal in this work was to identify the different types of personal information that people share in chat-oriented dialogue, and to investigate how communication of personal information occurs in different dialogue activities. In order to do this, we annotated the utterances in human dialogues with the information that the speaker was sharing about themselves or about entities close to them, such as family members, friends, and organizations. We did this for three different dialogue datasets. These annotations were used to determine categories of information that a person can share, and our annotation results were compared across the three different dialogue datasets to find general patterns for all chat-oriented dialogues, as well as information-sharing variations that occurred in the different corpora.

2. Related Work

Much of the previous work in chat-oriented dialogue has focused on identifying its structure - on both the micro level in terms of grammatical patterns and speech functions and the macro level in terms of conversational stages and genres (Eggins and Slade, 1997). As Slade and Gardner (1993) note, one proposal from Suzanne Eggins is that different instances of chat-oriented dialogue differ based on qualities of the power relationships, amount of contact, and emotional attachment between the conversational participants. de Silva Joyce and Slade (2000) go on to further describe Eggins' definition of two subcategories of casual conversation: polite - where there is limited contact between the participants outside of their current conversation - and confirming - where the participants are in close contact and

have developed an emotional attachment.

Within casual conversations, Eggins and Slade (1997) have argued that the interaction takes the form of sequences of chunks and chats. They define chunks as the more structured, monologue-like interactions in conversations where one party tends to dominate, and chats as the highly interactive segments of conversation where multiple speakers tend to be involved in the conversation and compete for turns. They further conceptualize the idea of 'chunks' by defining different categories of interaction that occur in these segments - such as gossip, anecdote, and joke-telling - and breaking down each category into a structured sequence of conversational stages.

Although these previous works examine the content of the chat-oriented dialogues in their work on understanding the dialogue structure, the focus is not on identifying and understanding what people tend to talk about in their casual conversations. Other work has taken a different approach to studying chat-oriented dialogue by focusing on the progression of topics and its effect on the conversational participants' experience in the interaction.

Previous work from Hirano et al. (2016) has investigated different personalization strategies in chat-oriented dialogue and found that the strategies related to topic elaboration and topic changing significantly increased the satisfaction of the conversational participants. A major component of these strategies is the ability to use the information known about one dialogue participant to guide the other dialogue participant in selecting an appropriate new topic. However, this work did not cover what types of partner information are used in these personalization strategies to drive the decisions made by a dialogue participant.

Mitsuda et al. (2017) studied a similar phenomenon when they developed a taxonomy of categories for the types of perceived information that a human can glean from an utterance. Their work provides evidence that chat-oriented dialogues are dominated by personal information, since 78.5% of the categories of perceived information are directly related to personal information about the speaker or about people, events, and organizations that the speaker has a relationship with. However, this work did not focus on personal information about the speaker, since it also included categories for general world knowledge and did not sufficiently discriminate types of personal information.

Work by Allwood et al. (2011) that studied the differences in topics that are discussed in monocultural and intercultural first-time encounters also uncovered common topics that are directly related to the sharing of personal information, such as age, family, and religion. It is notable that Allwood et al. (2011) observed that there was significant overlap in the topics that were discussed between all three dialogue situations that they studied: dialogues between two Chinese participants, two Swedish participants, and one Chinese and one Swedish participant. This provides strong support that personal information is a vital component of all chat-oriented dialogue, regardless of culture and participant similarity. However, similar to the work by Mitsuda et al. (2017), the topics that were identified in this work are also too broad to be useful for gaining an understanding of personal information shared in dialogue.

One source of detailed types of personal information is Schema.org, an open source resource that contains schemas for structured data, primarily for use by web developers in order to create web pages that are easily indexable by search engines. The schemas on Schema.org are collections of properties that can be used to described specific concepts. For our work, we were particularly interested in the schema definition of the concept 'Person', part of which can be found in Table 1. We used the properties of a Person as defined by Schema.org as the starting point for defining the personal attribute types that are shared in chat-oriented dialogues.

Property	**Definition**
Nationality	Nationality of the person.
Net Worth	The total financial value of the person as calculated by subtracting assets from liabilities.
Owns	Products owned by the person.
Parent	A parent of this person.
Performer In	Event that this person is a performer or participant in.
Related To	The most generic familial relation.

Table 1: Examples of the properties of a 'Person' along with their definitions, as defined by Schema.org

3. Attributes and Entities

For this work, we were interested in identifying the types of personal information that are shared by a speaker in chat-oriented dialogue. Within this context, the term *attribute instance* is used to refer to a single piece of personal information shared by a speaker, such as that they were born in Phoenix or that they are 45 years old. An *attribute instance* can be completely described in terms of its *attribute type* and its *attribute value*, such as (birthplace Phoenix) or (age 45).

In this work, an *attribute instance* is applied to a particular individual of an *entity type* (not necessarily a person). For example, it is possible to say that Phoenix is a large city, where this *attribute instance* would have an *attribute type* of 'size' and an *attribute value* of 'large'. In this case, the *entity type* of the *attribute instance* is 'Place', because the individual being described - Phoenix - is a city.

4. Corpora

In order to look at a range of chat data rather than be limited to a specific setting, we focused on three different datasets of dyadic chat. The three datasets were the Story-swapping Corpus, the Switchboard Corpus (Godfrey and Holliman, 1993), and the SpeedDate Corpus.

4.1. Story-swapping Corpus

In 2015, Gilani et al. (2016) conducted a study on story-swapping that investigated the impact that different virtual storytellers had on a human participant who engaged in a story-swapping dialogue with the virtual storyteller. Participants interacted with different versions of a virtual storyteller in a "get to know you" scenario where they answered predetermined ice-breaker questions. The participants were given the ice-breaker questions beforehand and initiated each exchange with the virtual storyteller by asking one of the questions. In response, the virtual storyteller produced a relevant story that answered the question, and asked the same question back to the participant, who then gave an answer in response.

Because our work aimed to identify attributes shared in natural human dialogues, only the utterances given by the human participant were included from this dataset and the utterances produced from the virtual storytellers were ignored. Although this dataset could be viewed as an inorganic source of human conversation due to the scripted nature of the topics and the artificial storyteller participant, the ice-breaker questions function as guiding topics to the conversation and the information that a human participant chose to share in response to the topics was produced naturally from the human.

4.2. Switchboard Corpus

The Switchboard: Telephone Speech Corpus for Research and Development was collected between 1990 and 1991 by Texas Instruments, with sponsorship from DARPA. It consists of two-sided telephone conversations between human participants, where the participants were connected via a robotic switchboard operator and were unfamiliar with the other person they were speaking to. Upon initiation of a telephone conversation, the operator gave the participants a specific topic to discuss with each other based on their previously indicated topics of interest. Some example topics are capital punishment, pets, cars, and recycling. In general, the course of the ensuing conversation focused on the assigned topic as speakers exchanged their opinions and relevant experiences to the topic at hand.

The Switchboard Corpus was originally released by the Linguistics Data Consortium in 1992-1993, but was released again in 1997 with some errors fixed. In total, it contains 2400 conversations between 543 speakers (Godfrey et al., 1992). Transcripts of the recorded telephone conversations were also produced, which has resulted in much work on annotating the dialogues with different linguistics features, from phonetics to syntax (Calhoun et al., 2010). Of particular importance to this work are the word-level, turn, and utterance boundary transcriptions, where the speakers were labeled as 'A' and 'B'. An extension of the 1997 "Switchboard 1 Release 2" Corpus - called the "Switchboard Dialog Act Corpus" - was used in this work. It contains word-level transcriptions of the dialogues segmented into turn-taking utterances, where each utterance is tagged with a dialog act but the dialog act was stripped from the utterance for our purposes (Stolcke et al., 2000).

4.3. SpeedDate Corpus

In 2005, Jurafsky et al. (2009) used three speed-dating sessions run at an elite private American university to collect casual dialogues. These speed-dating sessions were composed of graduate students from the university who participated in 4-minute "get to know you" sessions with each other on a one-to-one basis. Each session occurred in an open setting, and each participant wore an audio recorder during the session so that the audio could be captured. Transcribers at a professional transcription service used the recordings to create a transcript for each date between two graduate students. In total, there were 991 dates that had usable transcript data.

Although the majority of each dialogue followed the typical open-ended structure of sharing personal information in a get-to-know-you setting - as expected of a first date - the content of many of the dialogues was also directly affected by the speed-dating environment, since participants typically began their conversations with a discussion of the nature of the speed-dating activity. One common occurrence was that participants would often comment on the tasks that were required of them - such as their difficulty in filling out the surveys about each person with whom they have a date. In addition, participants would often bring up noteworthy elements of the speed-dating environment - for example, that there is a large proportion of law students taking part in the speed-dating activity.

5. Procedure for the Identification of Attribute Instances

75 dialogues from each corpus were randomly selected for annotation. Each utterance was appraised for whether there was a derivable attribute instance. If there was an attribute instance, then Schema.org was investigated to find an attribute type that would capture the information in the attribute instance. In the case that no corresponding attribute type could be found, a new type was created. This new type was then considered in each subsequent utterance as a possible attribute type - along with the properties from Schema.org - if an attribute instance was found. Once the attribute type of an utterance was identified, then the value of the attribute instance was selected as a substring from the utterance and the entity type was identified by classifying what the attribute was being assigned to. If there was no attribute instance derivable from the utterance, then it was not annotated.

Frequently, there were multiple attribute instances within a single utterance. All possible attribute instances that could be derived from a single utterance were identified and annotated for that utterance.

Table 2 shows an example of the annotations for a dialogue from the Switchboard Corpus.

6. Attributes in Chat-Oriented Dialogues

6.1. Entity Types

We found a total of 12 different entities that people share information about in these chat-oriented dialogues. The overwhelming majority of attributes were given about a Person (96.4%). Table 3 depicts the 12 entity types that were

DIALOGUE	ATTRIBUTE VALUE	ATTRIBUTE TYPE	ENTITY TYPE
A: I used to jog somewhat.	jog	previous-activities	person
B: I had an exercise bike.	exercise bike	previously-own	person
B: I used to have one.			
B: And I finally got rid of it cause I never used it.			
B: But I do use my treadmill.	treadmill	activities	person
A: Uh-huh.			
A: Well, that's good.			
A: Yeah.			
A: My parents have a treadmill.	parents, treadmill	parent.t, owns	person, person

Table 2: Example of attribute annotations for a portion of a dialogue from the Switchboard Corpus

identified as well as their respective distributions in the dialogues.

Entity Types	Frequency
Person	4503
Place	61
Organization	32
Pet	30
Car	21
Program	7
Job	6
Course	3
Restaurant	2
Activity	2
Sports Team	1
Event	1

Table 3: Entity types found in dialogue

6.2. Attribute Types

We found a total of 166 types of attributes that people share in these chat-oriented dialogues. Most attribute types were slot-based, where a specific filler could be found for the attribute based on the information shared in the utterance - such as a person's name or the university they are currently studying at. However, there were many instances of a binary-valued attribute type, where an utterance would indicate that a particular property was true or false for the given entity under discussion. An example of this occurs in the sentence 'My daughter agrees with me' since this utterance clearly depicts that the speaker has a daughter. This can be captured through a binary attribute type, which was called 'children.daughter.t'. All attribute types that end in .t indicate a binary attribute with the value of true, whereas attribute types that end in .f indicate a binary attribute with the value of false. Both slot-based and binary attribute types were mapped to a single word in the utterance as the attribute value.

These 166 attribute types are grouped into 12 broad categories. The first 9 categories pertain to the attribute types that were used most frequently in relation to the entity type of 'Person' and can be seen in Table 4. The remaining 3 categories include attribute types that were never used in relation to the entity type of 'Person' and can be seen in Table 5. The 12 categories are described in more detail below.

6.2.1. Demographics

The category *Demographics* includes attributes for basic identifying information - such as age and name - as well as traits that distinguish different populations from one another - such as heritage and membership in different organizations. This category also contains attributes that indicate well-being.

6.2.2. Personality

Personality attributes capture the distinctive elements of a person that grant them individuality. It includes a person's likes, dislikes, fears, goals, plans, and physical traits.

6.2.3. Relationships

Relationships contains attributes that provide information about the different relationships a person has with other people - mother, father, children, neighbor, sibling, and so on. It includes whether the person has these relationships and how long they have known each other. A person's history of romantic relationships is also captured by the attributes in this category.

6.2.4. Work

The attributes in the category *Work* focus on the details of a person's current - and past - employment. It includes characteristics like job title, length of employment, location of work, and company.

6.2.5. Education

Similar to *Work*, *Education* encapsulates attributes that define a person's educational history. It captures relevant details such as where a person has attended school, what field they have studied, and the degree they pursued there.

6.2.6. Residence

Residence contains attributes for current and past living arrangements, such as the type of dwelling, the location of residency, and the length of a time a person lived there. It also includes attributes for more detailed aspects of the living arrangements, such as specific qualities of the home.

6.2.7. Possessions

Attributes in the *Possessions* category indicate the material wealth of a person. These attributes capture what objects a person owns and doesn't own, as well as any information on their financial state.

6.2.8. Behavior

Behavior attributes encapsulate the different recurring activities that a person does, as well as the different experiences they have had in their lifetime. There are also at-

Demographics	Personality	Relationships	Work	Education	Residence	Behavior
age	dislikes	affiliatedwith	company.type	currenteducation.field	home.details	activities
alive.f	familiarwith	children.daughter.t	previouswork.company.name	currenteducation.graduation	home.length	activities-no
birthdate	favorites	children.f	previouswork.company.type	currenteducation.length	home.location	activities.length
birthmonth	fears	children.number	previouswork.description	currenteducation.location	home.situation	experiences-no
birthplace	goals	children.son.t	previouswork.length	currenteducation.t	home.type	experiences-yes
birthyear	goals-maybe	children.t	previouswork.location	currenteducation.type	previoushome.details	previousactivities
childhood	goals-no	cousin.t	previouswork.status	currenteducation.year	previoushome.length	travelled.location
deathdate	interestedin	frequencyofvisits	previouswork.t	degree	previoushome.location	travelled.not
deathyear	isa	friend.t	previouswork.title	instructor	previoushome.type	usesservice
gender	isnota	grandchild.f	previouswork.type	previouseducation.completed.f	usesservice-no	
heritage	languages	grandchild.granddaughter.t	vacationtime	previouseducation.completed.t		
memberof	likes	grandparent.grandfather.t	work.company.name	previouseducation.field		
mentalstate	misses-no	grandparent.grandmother.t	work.company.type	previouseducation.graduation		
middlename	misses-yes	grandparent.t	work.f	previouseducation.length		
name	notfamiliarwith	inlaw.brother.t	work.length	previouseducation.location		
nickname	notinterestedin	inlaw.mother.t	work.status	previouseducation.t	**Possessions**	**Distinguishments**
physicalstate	plans	inlaw.t	work.t	previouseducation.type	financialstate	accomplishments
previousmemberof	plans-maybe	knows	work.title	previouseducation.type.f	owns	artistof
	plans-no	neighbor.t	work.type		owns-no	producerof
	previousfavorites	nibling.t			owns.length	
	previousgoals	parent.father.t			previousfinancialstate	
	previousinterests	parent.mother.t			previouslyown	
	previouslikes	parent.t				
	religion	previousromantic.length				
	trait	previousromantic.t				
	traits-no	previousromantic.type				
		romantic.f				
		romantic.length				
		romantic.t				
		romantic.type				
		sibling.brother.number				
		sibling.brother.t				
		sibling.sister.t				
		sibling.t				
		stepparent.father.t				
		uncle.t				

Table 4: Attribute types found in dialogue, attributable to a Person

Pet	Vehicle	Location
breed	model	commonactivity
	numberofdoors	containedinplace
	vehicleengine	crime
	vehiclemodeldate	demographics
	price	largerthan
		proximity
		previousnames
		servescuisine
		similarto
		size
		type
		rules
		clients.type

Table 5: Attribute types found in dialogue, not attributable to a Person

tributes that indicate certain activities or experiences a person does not participate in, or specific locations that a person has never travelled to.

6.2.9. Distinguishments

The attributes in the *Distinguishments* category list a person's accomplishments.

6.2.10. Pet

Pet attributes include only those attributes which were used to describe pets (namely, their breed). Other attributes from the *Personality* category were also associated with pets in the dialogues.

6.2.11. Vehicle

Vehicle attributes focus on the model and price of a Vehicle entity, as well as other features like the number of doors.

6.2.12. Location

Location attributes are used in the descriptions of places, such as cities and businesses. These cover a variety of characteristics, including demographic distributions, proximity to other places, and size.

7. Attribute and Entity Type Evaluation

We performed a small inter-annotator agreement study on 15 dialogues, which was composed of 5 randomly selected dialogues from each corpus. We calculated inter-annotator agreement on both attribute type and entity type, but not for attribute values. Because the possible attribute values for any attribute type are open-ended paraphrases of a part of the utterance, it is impossible to give annotators a finite list of possible attribute values. For this reason, it is difficult to operationalize the similarity between the attribute values chosen by different annotators. In addition, we believe that the inter-annotator agreement on the attribute types provides a good approximation for the agreement that would be found on the attribute values, since these two concepts are directly related.

We calculated the AC1 measure of inter-annotator agreement for both attribute and entity types. As defined by Gwet (2002), AC1 aims to overcome the issue in other inter-annotator agreement calculations - namely, Cohen's Kappa and Scott's π-statistic - that causes them to produce unexpectedly low agreement measures when given data with large differences in the trait frequencies, which

is an accurate description of our data. It accomplishes this through a revised chance-agreement measure, which is explained in detail in Gwet (2002). We calculated AC1 for each attribute type by treating them as a binary labels for each utterance, such that an annotator either indicated the particular attribute type was 'present' or 'not present' in a particular utterance. AC1 is calculated as:

$$AC1 = \frac{p - e(\gamma)}{1 - e(\gamma)} \quad (1)$$

where p is the base agreement measure between the annotators and $e(\gamma)$ is the revised chance-agreement probability. We only calculated AC1 for those attribute and entity types that were annotated by at least one annotator, which was 94 attribute types (out of 166 total) and 2 entity types (out of 12 total). The final AC1 value is calculated as the average of the AC1 values calculated for each attribute type and entity type and is shown in Table 6.

Annotators were a mix of experts and non-experts. Half of the annotation material came from the authors' work on identifying the attribute and entity types. The other half came from novice annotations, since they were conducted by an outsider to the work who was given an annotation manual written by the authors. Each annotator was given a finite list of possible attribute types and entity types that an utterance could be classified with.

Annotation Element	AC1
Attribute Type	99.76%
Entity Type	96.28%

Table 6: Inter-Annotator Agreement Statistics

The inter-annotator agreement is so high because each utterance contains at most a few attributes or entities, and annotators agree on the many attribute and entity types that are not present in each utterance.

Since we are mainly interested in annotator agreement on the presence of the different labels, we corrected for this artifact of the inter-annotator agreement measure by also calculating precision, recall, and F1-score for the attribute and entity type annotations. We calculated these measures for each individual attribute type and entity type. The final measures shown in Table 7 reflect the average of these individual measures.

Annotation Element	Precision	Recall	F1-Score
Attribute Type	36.4%	34.6%	35.5%
Entity Type	59.4%	60.7%	60.0%

Table 7: Precision, Recall, and F1-Score for Annotations

The resulting precision, recall, and F1-score presents a more pessimistic perspective on the inter-annotator agreement for attribute and entity types in dialogue utterances, especially for the attribute types. The fine-grained distinction between the different attribute types seems to be a limiting factor in the reliability of the attribute type labels. Frequently, the annotators agreed that an utterance should be labeled with a specific type category (such as 'currenteducation' or 'previouswork'), but they then chose different sub-categories (such as 'previouswork.type' versus 'previouswork.description'). In addition, it can be argued that there is a degree of semantic overlap between many of the attribute types, such as 'goals' and 'interests', which is also evidenced through the annotations made by each of the annotators. As such, these attribute types would be better captured through a higher-order type that encapsulates both.

With the limitations of the current annotation scheme identified, it is important to note that this work does not purport to have identified the single true taxonomy of personal information present in chat-oriented dialogues. Instead, we aimed to get a sense of the types of information that tend to occur in chat-oriented dialogues and offer reasonably plausible categories of topics that can be found.

	Attribute	Instances		Utterances		
Corpora	Total	Range	Average	Total	Range	Average
Story-swapping	1921	6-61	25.6	4509	21-173	60.1
Switchboard	1495	2-70	19.9	13982	64-444	186.4
SpeedDate	1251	2-46	16.7	8673	66-189	115.6

Table 8: Distribution of the number of attribute instances and utterances between the three corpora

8. Observations and Analysis of the Attribute Distributions

The total number of utterances and attribute instances that occurred over the 75 dialogues of each corpus - as well as their ranges and averages within a single dialogue in each corpus - can be found in Table 8. Although the total number of utterances is much lower in the Story-swapping Corpus than the other two, this can be attributed to the fact that only the half of each dialogue that was spoken by the human participant was considered. In total - when taking into account the other half of the dialogues - the total number of utterances over the 75 dialogues for the Story-swapping Corpus is approximately 9000, which is similar to the total number of utterances of the other two corpora.

Tables 9, 10, and 11 show the twenty most frequently occurring attributes for the Story-swapping Corpus, the Switchboard Corpus, and the SpeedDate Corpus, respectively. Table 12 shows the twenty most frequently occurring attribute types over all of the corpora and their distribution across the corpora. Out of the twenty most-frequent attribute types for each corpus, eight attribute types appeared for all corpora: likes, activities, dislikes, birthplace, home.location, travelled.location, goals, and previouseducation.type.

Furthermore, out of the 166 total different attribute types that were found, 62 of them occurred in all three corpora. Since only 129 different attribute types appeared in the Switchboard Corpus, 110 in the Storyswapping Corpus, and 91 in the SpeedDate Corpus, the fact that 62 attribute types occurred in all three corpora shows that there is a great degree of overlap between the different corpora, since, at minimum, half of all of the attribute types found in each corpus were also found in both of the others. A major source of overlap between the three corpora is their abundance of personality, demographics, and behavior attribute

Attribute Types	Frequency
likes	423
trait	124
experiences-yes	114
experiences-no	82
activities	70
dislikes	69
previouseducation.location	63
favorites	63
birthplace	62
home.location	52
travelled.location	52
previousactivities	46
age	44
goals	40
traits-no	38
activities-no	37
work.type	32
previouseducation.type	31
work.title	30
work.status	24

Table 9: Top twenty attributes in the Story-swapping Corpus

Attribute Types	Frequency
likes	141
activities	96
home.location	88
romantic.type	72
owns	58
trait	56
work.t	52
children.t	49
age	47
dislikes	35
work.status	30
goals	30
travelled.location	26
birthplace	24
home.type	24
parent.mother.t	24
previoushome.location	23
previouseducation.type	21
friend.t	21
childhood	20

Table 10: Top twenty attributes in the Switchboard Corpus

Attribute Types	Frequency
currenteducation.field	157
name	122
likes	103
currenteducation.year	82
birthplace	80
currenteducation.type	76
goals	46
activities	41
previouseducation.location	39
travelled.location	38
previouseducation.type	35
currenteducation.t	29
friend.t	26
previoushome.location	26
plans	24
dislikes	17
home.location	16
currenteducation.length	13
size	12
sibling.sister.t	12

Table 11: Top twenty attributes in the SpeedDate Corpus

Attribute Types	Frequency	Top-Twenty Appearances
likes	667	3
activities	207	3
trait	191	2
currenteducation.field	170	1
birthplace	166	3
home.location	156	3
name	137	1
experiences-yes	135	1
dislikes	121	3
travelled.location	116	3
goals	116	3
previouseducation.location	106	2
age	100	2
experiences-no	92	1
romantic.type	89	1
previouseducation.type	87	3
currenteducation.year	86	1
currenteducation.type	84	1
owns	72	1
favorites	69	1

Table 12: Twenty most frequent attribute types found overall in the three corpora, and the number of corpora in which each appeared in the individual top-twenty lists

types, which can most clearly be seen when comparing their respective most-frequent attribute types, although the specific attribute types in these categories varied between the corpora.

Upon closer examination of the distributions for each corpus, there were telling differences between the three corpora that relate to their variation in domain and dialogue participants. The most apparent discrepancy occurs in the distribution of work-related and education-related attribute types. It is seen in the Story-swapping Corpus and Switchboard Corpus that both had more frequent work-related attribute types than the SpeedDate Corpus, which had more frequent education-related attributes. This makes sense in light of the fact that all of the conversation participants in the SpeedDate Corpus were graduate students, and thus their education - and not their employment - was most salient.

It was also observed that relationship-related attribute types occurred most frequently in the Switchboard Corpus, as compared to the other corpora. The focus of the dialogues in the Switchboard Corpus was less on the person speaking - unlike in the ice-breaker paradigms of the other two corpora - and instead, the conversational participants were discussing any relevant experience in relation to specified topics - such as childcare and the justice system. Often, these speakers drew on the experiences of those closely related to them - either as romantic partners, family members, or friends - which can explain why much information about a person's relationships was observed in these dialogues.

Overall, there was much overlap in the distribution of attribute types between the three different chat-oriented corpora. In general, it was observed that many of the most frequently occurring attribute types between all three corpora often belong to the same personal information categories of personality, behavior, education, and relationships. These distributions of common attribute types observed in this work begin to indicate what personal information is focused on in chat-oriented dialogue, regardless of the paradigm in which the dialogue was conducted, and provides further evidence that the act of sharing personal information is similar across different dialogue paradigms. In addition, the differences that are observed between the attribute types of the three corpora aid in delineating different types of corpora, by indicating dominating topics for the participants involved.

Although the three corpora used in this work differed in the paradigms by which they were collected, it could be argued that all three belong to the same overarching category of polite casual conversation as defined by Eggins (de Silva Joyce and Slade, 2000). It may be the case that the common attribute types observed in this work would not extend to the confirming subcategory of casual conversation, where the participants are well-acquainted and in frequent contact with one another, or to other types of casual conversation. Further exploration of the similarities and differences between the common attribute types of different categories of chat-oriented dialogue remains as an area of future work.

9. Conclusions and Future Work

We have created two useful resources for chat-oriented dialogue - first, an enhanced collection of personal attributes that can be used as topics for dialogue interaction and, second, a corpus of dialogues annotated for attribute types, attribute values, and entity types. These resources were created through examination of three different kinds of chat dialogue corpora. Comparative analysis shows that while there are significant differences in the types and frequencies of attributes in the different corpora (which could be used to distinguish different genres of chat), there are also significant similarities that occur in two or all three.

We plan to use these resources for several purposes in our future work. First, these annotations can be used as training and testing data for classifiers that recognize when people mention these attributes in their utterances. This capability will enable dialogue agents to develop a user model of the other conversational participants and to personalize the dialogues to the topics most interesting to them. In addition, the commonly discussed personal attributes indicated by this work can be incorporated into the backstory development of dialogue agents so that they can engage in more human-like chat.

10. Acknowledgements

The first author was supported by the National Science Foundation under grant CNS-1560426, "REU Site: Research in Interactive Virtual Experiences" (PI: Evan Suma). The effort described here has been partly supported by the U.S. Army. Any opinions, content or information presented does not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

11. Bibliographical References

Allwood, J., Lindström, N. B., and Lu, J. (2011). Intercultural dynamics of fist acquaintance: comparative study of swedish, chinese and swedish-chinese first time encounters. In *International Conference on Universal Access in Human-Computer Interaction*, pages 12–21. Springer.

Calhoun, S., Carletta, J., Brenier, J. M., Mayo, N., Jurafsky, D., Steedman, M., and Beaver, D. (2010). The nxt-format switchboard corpus: a rich resource for investigating the syntax, semantics, pragmatics and prosody of dialogue. *Language Resources and Evaluation*, 44:387–419.

de Silva Joyce, H. and Slade, D. (2000). The nature of casual conversation: Implications for teaching. *Teachers' Voices*, 6:viii–xv.

Eggins, S. and Slade, D. (1997). *Analysing Casual Conversation*. Continuum.

Gilani, S. N., Sheetz, K., Lucas, G., and Traum, D. (2016). What kind of stories should a virtual human swap? In *The Sixteenth International Conference on Intelligent Virtual Agents*, pages 128–140, Los Angeles, CA, USA, September. Springer.

Godfrey, J., Holliman, E., and McDaniel, J. (1992). Switchboard: telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520. IEEE.

Gwet, K. (2002). Kappa statistic is not satisfactory for assessing the extent of agreement between raters. *Statistical methods for inter-rater reliability assessment*, 1(6):1–6.

Hirano, T., Higashinaka, R., and Matsuo, Y. (2016). Analyzing post-dialogue comments by speakers – how do humans personalize their utterances in dialogue? –. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 157–165, Los Angeles, September. Association for Computational Linguistics.

Jurafsky, D., Ranganath, R., and McFarland, D. (2009). Extracting social meaning: Identifying interactional style in spoken conversation. *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 638–646.

Mitsuda, K., Higashinaka, R., and Matsuo, Y. (2017). What information should a dialogue system understand?: Collection and analysis of perceived information in chat-oriented dialogue. In *International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

Slade, D. and Gardner, R., (1993). *Teaching Casual Conversation: The Issue of Simplification*. ERIC.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V., and Meteer, M. (2000). Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3).

12. Language Resource References

John J. Godfrey and Edward Holliman. (1993). *Switchboard-1 Release 2*. Linguistic Data Consortium, 1.0, ISLRN 988-076-156-109-5.

A Vietnamese Dialog Act Corpus Based on ISO 24617-2 Standard

Thi-Lan Ngo[1,2], **Khac Linh Pham**[2], **Hiadeki Takeda**[3]
[1]University of Information and Communication Technology (ICTU), Thainguyen Vietnam
[2]University of Engineering and Technology (UET-VNU), Hanoi Vietnam
[3]National Institute of Informatics (NII), Tokyo Japan
ntlan@ictu.edu.vn, 15021790@vnu.edu.vn, takeda@nii.ac.jp

Abstract

The voice-based human-machine interaction systems such as personal virtual assistants, chat-bots, and contact centers are becoming increasingly popular. In this trend, conversation mining research also is getting the attention of many researchers. Standardized data play an important role in conversation mining. In this paper, we present a new Vietnamese corpus annotated for dialog acts using the ISO 24617-2 standard (2012), for emotions using Ekman's six primitives (1972), and for sentiment using the tags "positive", "negative" and "neutral". Emotion and sentiment are tagged at functional segment level. We show how the corpus is constructed and provide a brief statistical description of the data. This is the first Vietnamese dialog act corpus.

Keywords: Vietnamese corpus, dialog act corpus, ISO 24617–2 standard, conversational mining.

1. Introduction

In recent years, an extremely rapid progress in speech processing and recognition technology has led the emergence of voice-based human-machine interaction systems such as mobile virtual assistants, automatic contact centres, and chat-bots. These applications accommodate different purposes but they all need the ability to understand conversations while interacting with the user. Therefore, with the development trend of intelligent systems using the conversational interface, studies on conversation mining such as conversation structure analysis, conversation topic modelling, user intent understanding, and user emotion or satisfaction identification have attracted the attention of many researchers. In these research, standardized dialog act corpora are the foundation. It is widely accepted that dialog act annotation is very valuable in further understanding of interaction structure, and in the design of artificial spoken or text dialogue (Wrede and Shriberg, 2003; Stolcke et al., 2006). There were several dialog act corpora available to the research community like as TRAINS (Traum, 1996), VERBMOBIL (Alexandersson et al., 1998), SWBD-DAMSL (Jurafsky et al., 1997), MRDA (Shriberg et al., 2004), AMI (McCowan et al., 2005) and so on. However, different corpora often apply a different scheme or modifying the existing scheme for dialog act annotation to serve task-specific needs. This creates a hardship in comparison of results and conclusions obtained when using different approaches due to a wide scatter of data in terms of the used annotation. Currently, ISO 24617-2 standard (ISO, 2012) is seen as "lingua franca" for dialog act annotation (Chowdhury et al., 2016; Bunt et al., 2012). Experimental studies in DBOX (Amanova et al., 2016) and DialogBank (Wijnhoven, 2016) corpora and the DIT theory (Bunt, 2009), of the DIT++ annotation scheme that underlies the ISO 24617-2 standard have shown good effects of the ISO standard on dialog act annotation. Thus, in our work, we build a Vietnamese spoken corpus, called ViDA corpus, annotated dialog act using the ISO 24617–2 standard (ISO, 2012). Addition, we emotion tagging for functional segments (FSs) using the Ekman's list of basic emotions (Ekman, 1972) and sentiment annotation for FSs using three common labels: "positive", "negative" and "neutral". The differences between our work and the previous studies is that:

- First, this is the first dialog act corpus for Vietnamese. Our corpus is not only annotated all labels, dimensions, relation defined in the ISO 24617–2 standard but also annotated meta information such as gender, the dialect of participants.
- Second, this is the first corpus ever that annotate emotion, sentiment at functional segment level. The earlier corpora on emotion analysis and sentiment analysis only annotate at a sentence, turn or document level.
- Third, we create a Vietnamese dialect dictionary for Vietnamese automatic dialect/accent detection in spoken conversation systems.

To build the corpus, we use IARPA Babel Vietnamese Language Pack IARPA-babel107b-v0.7 (IARPA-babel107b) (Andrus, Tony, et al, 2017). A brief description of IARPA–babel107b is presented in Subsection 2.1. The process of our corpus annotation is shown in Figure 1 and detailed in Section 2. Subsection 2.2 presents the segmentation of turns in our corpus into a Functional segment. In Subsection 2.3, we introduce Vietnamese dialog act annotation based on the ISO 24617-2 standard. Subsection 2.4 describes emotion and sentiment tagging. Finally is the conclusion part of the plan for future developments of our corpus.

2. Corpus and Annotation

2.1. Dataset pre–processing

We select transcripts of Vietnamese conversations obtained by an automatic speech recognition (ASR) in the IARPA (Intelligence Advanced Research Projects Activity) Babel program for data annotation. IARPA Data is published in

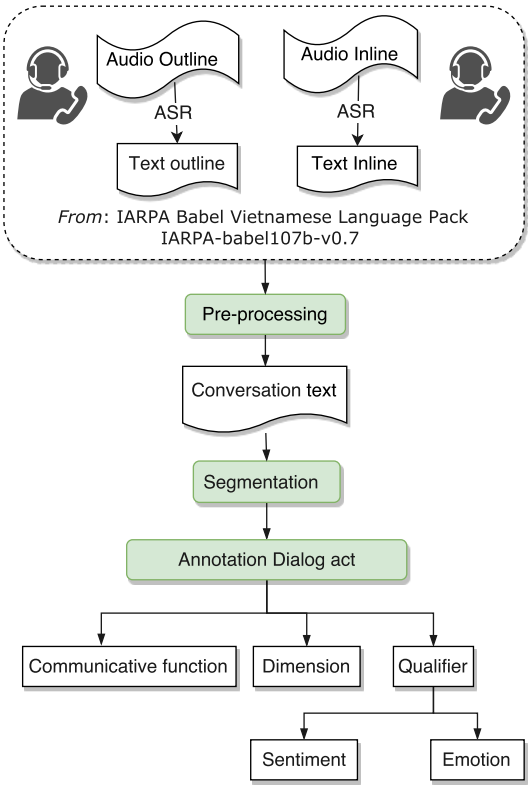


Figure 1: The process of data annotation.

IARPA-babel107b on LDC [1]. IARPA–babel107b has about 201 hours of Vietnamese conversational and scripted telephone speech with corresponding transcripts. The data is spoken in the North, Central and Southern dialect regions in Vietnam. We randomly select 28 dialogues in this dataset with any topic where the number of dialogue with each dialect regions is balanced to build our dialog act corpus. Data in IARPA–babel107b is made in style in which a conversation between two persons includes an inline audio file with an inline text file (corresponding transcript of the inline audio file) and an outline audio file with an outline text file (corresponding transcript of the outline audio file). We make conversational texts from the inline texts and outline texts. After that, we review the conversation texts to rearrange it to the correct order of turns in the conversation using audio files. Error words from the results of ASR are retained. There are 1823 from error words in total 23803 words (the word error rate is 7.66% WER). We also note the meta information of our data including dialect regions and gender of participants, call time, duration of the phone calls, the number of turns in a conversation (Table 1).

In the pre–processing stage, we build a Vietnamese dialect dictionary includes 167 distinct Southern Vietnam dialect words, 55 distinct Central Vietnam dialect words and their translation to the "standard" North Vietnam dialect. It is useful for automatic dialect/accent detection in spoken document retrieval systems. In human-machine interaction, it can help the system understand and communicate with users better. Instead of using the standard North Vietnamese dialect words for every machine, a friendly conversation interface application can detect the user's dialect then use that dialect to communicate with the user.

The segmentation and annotated process have done by two persons. The inter-annotator agreement score is computed by Fleiss kappa measure (Fleiss and Cohen, 1973).

2.2. Segmentation

Turns of conversation texts are segmented into functional segment (FS) unit, i.e., "minimal stretch of communicative behaviour that has one or more communicative functions" according to ISO 24617-2 standard. Our corpus has 28 dialogues, 2273 turns and 5065 functional segments. On average, each dialogue has 81.2 turns, 178.9 functional segments and each turn contain an average of 2.2 functional segments. The agreement score of the segmentation process is 0.62 Fleiss kappa measure.

2.3. Dialog Act Annotation based on ISO 24617-2

The ISO standard is amalgamated contributions from pre-existing schemes, and is multi–functional and multi–dimensional - several communication acts can apply to stretches within the same contribution to the conversation of a participant. The ISO scheme see a dialogue act under 8 components, includes: (1) a sender; (2) one or more addressees; (3) a communicative function; (4) a semantic content; (5) a dimension; (6) functional dependence relations; (7) feedback dependence relations; and (8) rhetorical relations. In the dialog act annotation step, we annotated dimensions and dialog act for FSs. It contains 57 dialog acts in 9 dimensions: task, auto-feedback, allo-feedback, time management functions, turn management, discourse structuring, own communication own communication management, partner communication management, and social obligation management. An example of dimensions, function and relationship between dialog act are shown in Table 5. The agreement scores of our Dialog Act annotation process is 0.76 Fleiss kappa measure. The proportion of dialogue acts in different dimensions in Table 2 shows that there is not much difference between in Vietnamese dialogues and English dialogues.

2.4. Qualifier

Our objective is to create an annotated corpus that will be base resources for future researches in Vietnamese dialogue/conversation mining, namely such as suggestion mining, emotion mining, sentiment mining, request mining, argument mining. In this corpus, we label sentiment at functional segments level into 3 categories: positive, negative, neutral. The agreement scores of our sentiment annotation process is 0.85 Fleiss kappa measure.

Also in this corpus, we annotate emotions at functional segments level according to the Ekman's (1972) list of basic emotions includes *joy, sadness, surprise, anger, fear* and *disgust*. We use *none* label for a FS does not express emotion. There are many different taxonomy for labeling the emotion, but we use Ekman's because it is used widely, is popular among researchers and simple enough to add into

[1] https://catalog.ldc.upenn.edu/LDC2017S01

Table 1: Information of Dialogues in ViDA corpus

STT	Dialogue	Participant 1		Participant 2		Number of Turns
		Dialect	*Gener*	*Dialect*	*Gener*	
1	D01_121544	North	female	Central	male	112
2	D02_013915	North	male	North	male	88
3	D03_222039	Central	female	Central	female	77
4	D04_002213	North	female	North	female	109
5	D05_165823	North	female	North	female	79
6	D06_200633	South	male	South	female	62
7	D07_225133	North	male	North	male	83
8	D08_162435	Central	female	Central	female	10
9	D09_203451	North	female	North	male	73
10	D10_202308	North	female	North	male	42
11	D13_183537	North	male	North	male	94
12	D14_014233	Central	male	Central	female	46
13	D15_182837	North	female	North	female	111
14	D16_202407	South	male	North	female	90
15	D17_023815	South	female	North	female	150
16	D18_160344	Central	male	Central	female	8
17	D19_162645	South	male	North	female	13
18	D20_151856	North	male	North	male	191
19	D22_005928	Central	female	Central	female	19
20	D24_130313	North	male	North	female	136
21	D26_120203	North	male	North	male	33
22	D27_223011	South	female	South	male	129
23	D28_003504	South	male	South	female	106
24	D29_134735	South	male	South	female	85
25	D39_173220	South	male	South	male	154
26	D47_192712	North	female	North	female	54
27	D51_002706	Central	male	Central	female	72
28	D53_010928	North	male	North	male	48

Table 2: Distribution of dialog acts in the ViDA corpus

Dimension	Number	Percent
task	3137	60.72
autoFeedback	801	15.51
alloFeedback	19	0.37
turnManagement	533	10.32
timeManagement	353	6.83
discourseStructuring	186	3.60
ownCommunicationManagement	100	1.94
partnerCommunicationManagement	24	0.46
socialObligationsManagement	13	0.25

Table 3: Distribution of sentiment in the ViDA corpus

positive	489	9.76%
negative	655	13.07%
neutral	3866	77.17%

Table 4: The distribution of emotion in ViDA corpus

anger	205	4.09%
disgust	126	2.51%
fear	129	2.57%
joy	383	7.65%
sadness	313	6.25%
surprise	358	7.15%
none	3496	69.78%

the ISO dialog act schema. The agreement scores of our emotion annotation process is 0.82 Fleiss kappa measure.

Sentiment and emotion annotation at FS level has many advantages in sentiment and emotion analysis field compare to other levels. Previous studies in this field usually performed at the sentence/turn level or document level. A sentence can carry more than one emotions or sentiments. A complex sentence can include many different parts that convey context information like emotion/sentiment cause, emotion/sentiment condition. Emotion/Sentiment annotation of FS, the smallest part of sentence/turn that has the meaningful communicative function, will help us to understand emotions/sentiment more concretely. Also, because FS tend to be much shorter than turn/sentence/document, the sentiment and emotion classification at FS level can be much easier and be able to achieve higher accuracy. For example, a sentence "I was very surprised at first then started feeling sad about the results" expresses both surprise and

sadness. We can emotion analysis of the first FS "I was very surprised at first" separate with the second FS "then started feeling sad about the results" easier than all the sentence. Similarity, it can help to extract the context part of emotion with emotion parts in a sentence "I was happy yesterday because it was my 5-year wedding anniversary. I would be excited if he gave me a small gift but he forgot it made me very sad".

3. Conclusion

Vietnamese is a very low-resource language. With the number of almost 100 millions speakers around the world (one of the most most spoken language) and the fast growing economy, the demand for Vietnamese standardized resource is greater than ever. Our corpus is aim to provide a first base resource for a variety of potential researches in Vietnamese natural language processing: mining suggestion, request, emotion, sentiment, argument in conversation, dialogue; Vietnamese dialog act identification; detection of user's gender, dialect. In the future, we intend to increase the size of our corpus and study deeper into the specific approaches of the these potential researches. We also intend to integrate them into real application such as personality virtual assistants, chat bots, contact center.

4. References

Alexandersson, J., Buschbeck-Wolf, B., Fujinami, T., Kipp, M., Koch, S., Maier, E., Reithinger, N., Schmitz, B., and Siegel, M. (1998). *Dialogue acts in Verbmobil 2*. DFKI Saarbrücken.

Amanova, D., Petukhova, V., and Klakow, D. (2016). Creating annotated dialogue resources: Cross-domain dialogue act classification. *Inform*, 26(11.5):36–0.

Andrus, Tony, et al. (2017). *IARPA Babel Vietnamese Language Pack IARPA-babel107b-v0.7 LDC2017S01*. IARPA (Intelligence Advanced Research Projects Activity) Babel program, distributed via LDC, ISLRN 401-277-958-467-7.

Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A. C., Hasida, K., Petukhova, V., Popescu-Belis, A., and Traum, D. R. (2012). Iso 24617-2: A semantically-based standard for dialogue annotation. In *LREC*, pages 430–437.

Bunt, H. (2009). The dit++ taxonomy for functional dialogue markup. In *AAMAS 2009 Workshop*, pages 13–24.

Chowdhury, S. A., Stepanov, E. A., and Riccardi, G. (2016). Transfer of corpus-specific dialogue act annotation to iso standard: Is it worth it? In *LREC*.

Ekman, P. (1972). Universal and cultural differences in facial expression of emotion. In *LREC'12*, page 207283. Nebraska Symposium on Motivation.

Fleiss, J. L. and Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

ISO. (2012). *ISO 24617–2: 2012 – Language resource management – Semantic annotation framework (SemAF) - Part 2: Dialog acts*. Geneva, Swizerland: International Organization for Standardization, ISLRN 401-277-958-467-7.

Jurafsky, D., Shriberg, E., and Biasca, D. (1997). Switchboard swbd-damsl labeling project coders manual, draft 13. *Technical Report 97–02*.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., et al. (2005). The ami meeting corpus. In *International Conference on Methods and Techniques in Behavioral Research*, volume 88.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The icsi meeting recorder dialog act (mrda) corpus. Technical report, INTERNATIONAL COMPUTER SCIENCE INST BERKELEY CA.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-Dykema, C., and Meteer, M. (2006). Dialogue act modeling for automatic tagging and recognition of conversational speech. *Dialogue*, 26(3).

Traum, D. (1996). Conversational agency: The trains-93 dialogue manager. In *In proceedings of Twente Workshop on Language Technology*. Citeseer.

Wijnhoven, K. (2016). Annotation representations and the construction of the dialogbank.

Wrede, B. and Shriberg, E. (2003). Relationship between dialogue acts and hot spots in meetings. In *Automatic Speech Recognition and Understanding, 2003.*, pages 180–185. IEEE.

Table 5: Example of dialog act, emotion and sentiment annotation on ViDA

Speaker	**Turn transcription**	**ID**	**FS**	**Dimension:function (relationship)«emotion »[sentiment]**
S	Alo chào anh ạ (Alo Hi you)	da1	alo	discourseStructuring: opening
		da2	chào anh ạ (Hi you)	socialObligationsManagement: initialGreeting
A	à chào em em dạo này khỏe không (ah hi you how are you)	da3, da4	à (ah)	autoFeedback: autoPositive (fe:da2) discourseStructuring: opening
		da5	chào em (hi you)	socialObligationsManagement:returnGreeting (fu:da2)
		da6	em dạo này khỏe không (are you fine)	task:propositionalQuestion
S	<laugh >em khỏe lắm tháng sau em ra sài gòn <breath > anh - em hứa sẽ ra thăm anh (<laugh >I'm good I'm going to Saigon next month <breath >you - I promise I will come to visit you)	da7, da8	<laugh >	turnManagement:turntake timeManagement:stalling «joy »
		da9	em khỏe lắm (I'm good)	task:answer (fu:da6)
		da10	tháng sau em ra sài gòn (I'm going to Saigon next month)	task:inform
		da11, da12	<breath >	turnManagement:turnKeep timeManagement:stalling
		da13	anh - (you -)	ownCommunicationManagement:retraction
		da14, da15	em hứa se ra thăm anh (I promise I will come to visit you)	ownCommunicationManagement:selfCorrection (fu:da13) task: promise

Annotating Reflections for Health Behavior Change Therapy

Nishitha Guntakandla *and* Rodney Nielsen
Human Intelligence and Language Technologies
University of North Texas
Denton, Texas 76201
nishithaguntakandla@my.unt.edu *and* Rodney.Nielsen@unt.edu

Abstract

We present our work on annotating *reflections*, an essential counselor behavioral code in motivational interviewing for psychotherapy on conversations that are a combination of casual and therapeutic dialogue. We annotated all the therapists' utterances from ten transcripts spanning more than five hours of conversation with Complex Reflection, Simple Reflection, or No Reflection. We also provide insights into corpus quality and code distributions. The corpus that we constructed and annotated in this effort is a vital resource for automated health behavior change therapy via a dialogue system. As the on-going work, additional conversations are being annotated at least by one annotator.

Keywords: Reflections, Motivational Interview, Behavioral codes, Companion robot conversations, Virtual health agents, Social-therapeutic agents

1. Introduction

Recent research has focused on developing companionable robots (companion bots) (Sarma et al., 2014) and relational agents (Vardoulakis et al., 2012) who build trust, rapport, and therapeutic alliance over time by engaging in both casual, and therapeutic dialogue. The goal of such social-therapeutic agents[1] is to play a key role in health care at home. They do so by incorporating therapeutic dialogue into friendly conversations with elderly individuals suffering from depression (Nielsen et al., 2010; Ring et al., 2016; Bickmore et al., 2005). Older adults those who lack social support can have negative effects on the health and well-being (Vardoulakis et al., 2012). Furthermore, it is not possible to have a therapist available to every individual at each instance of their life to provide assistance. Situations like these enhance the need for social-therapeutic agents.

As psychology studies (Keeley et al., 2014; Arkowitz et al., 2015) show, depression can be treated by Motivational Interviewing (MI) (Miller and Rollnick, 2012), a goal-oriented and client-centered style of psychotherapy. In MI, the therapist is directive in aiding the client to elicit the change talk, resolve ambivalence, and evoke motivation to make positive behavioral changes. The proficiency of the counselor in MI delivery is measured using behavioral coding systems like the MI Skills Code (MISC)(Miller et al., 2003) or the Motivational Interviewing Treatment Integrity (MITI)(Moyers et al., 2014). These systems are designed to manually annotate the counselor and the client utterances in a session with behavioral codes, which is an extremely labor intensive and costly task.

Existing work (Can et al., 2012; Pérez-Rosas et al., 2016; Tanana et al., 2015) annotated behavioral codes exclusively on MI sessions, which predominantly focus on therapeutic dialogue. MI sessions are conducted to help clients change behaviors that they identify or agree to being very important to improve their health. Unlike MI sessions, social-therapeutic agent conversations are a combination of casual (non-therapeutic) and therapeutic dialogue. They commence the conversation with rapport building and identify a behavior that should be discussed. Hence, there is a need for building an appropriate corpus that can be used to develop such agents. We address this need by constructing and annotating companionbot[2] conversations with behavioral codes. In this work, as the first step, we only focus on annotating *reflections* among all the counselor behavioral codes .

Reflective listening is a core MI micro-skill that therapists possess to illustrate their interest in a client's declaration, which intensifies the client's trust in the therapist. Reflections are statements that define therapeutic presumptions about what the client is trying to express about their inner experience (Miller et al., 2002). Specifically, counselor reflections are the thoughts, feelings, and meaning of what a client said. Reflections tend to be collaborative and non-judgmental and aim to guide the client towards resolving ambivalence. The outcome of using reflective statements is the client feels listened to, understood, less resistant, and more motivated to change. Counselor reflections can be categorized into Simple or Complex. Simple Reflections (SR) typically add little or no meaning to what the client has said. They do not go far beyond the client's original statement. Complex Reflections (CR) add substantial meaning by using inference and synthesis to state something the client has implied. Following is an instance presenting these two forms of reflection:

Client: *My mother is driving me crazy. She says she wants to remain independent, but she calls me four times a day with trivial questions. Then she gets mad when I give*

[1] social-therapeutic agents: This term is used to indicate any computer-based agent or bot that provide both social support and therapy.

[2] In our work we used companionbot, a specific type of social-therapeutic agent to collect the data

Therapist: So, they -- they pick you up and take you to their home, I guess?
Participant: Oh, yeah, mhm, yeah, I have an apartment at their place.
Therapist: Okay.
Participant: But {Breath} -- it's downstairs.
And, uh, it's -- I'm too much trouble for the, you know -- not on their end,
but I -- I feel I'd be just too much trouble staying there for them to take care of all the time.
Therapist: Okay, I see.
So where do you live right now?
Participant: Uh, in {Breath} -- in an assisted, uh, -- well an independent care apartment house.

Figure 1: Sample transcript from companionbot conversations.

her advice.
Interviewer: *Things are very stressful with your mother.* ***(SR)***
Interviewer: *You're having a hard time figuring out what your mother really wants.* ***(CR)***

In this work, we:

- Describe how we collected behavioral code annotations for more than 1536 counselor utterances taken from wizard-of-Oz-style conversations between human and companionbots.
- Conduct a corpus analysis, and present both inter-annotator agreement and code distribution among annotated utterances.



Figure 2: Prototype of our companionbot that is placed in the participants' room for them to interact with.

2. Corpus Construction

In this section, we describe the collection of companionbot conversations over which we annotate reflections.

2.1. Data Collection

We conducted Wizard of Oz experiments(Dahlbäck et al., 1993) to collect data for our corpus. Three psychologists (wizards) were employed to engage in one-to-one conversations with participants remotely. A total of 16 elderly individuals from the Denver, Colorado and Dallas-Fort Worth, Texas metro regions participated in these experiments. A fixed physical companionbot (shown in Figure 2) was placed in the participant's room for them to interact with at the scheduled times. Arrangements were made for a wizard to interact with a subject for 30 minutes in each session. Additionally, the psychologists were given guidelines on how to play the role of the companionbot and conduct appropriate dialogue. As a whole, 324 sessions were recorded. Audio files from all and video files from most of the sessions were gathered. The final corpus comprises a total of 162 hours of audio.

2.2. Transcription

The collected audio files were transcribed manually by linguists following the guidelines from (Strassel, 2003). The identifiers, *Therapist* and *Participant*, are assigned to the psychologist and the participant turns, respectively. On average, each transcript contains 115 turns of therapist and participant dialogue. Within each turn, a speaker produces multiple sentences/utterances. Speech is segmented at the end of a sentence. This sentence segmentation follows the directives:

1. Sentence is an independent clause.
2. If a complex sentence contains a coordinating conjunction (*and*, *but*, *for*, *nor*, *or*, *so*, and *yet*) and the following clause is not dependent on the first clause, then these two clauses are segmented into two separate sentences.

Also, additional rules are followed to deal with noise, pauses, and false starts.

Utterances (Behavioral Code)
Participant: So I know that in the past, what I did was I either got down on the floor and did, let's say, twenty sit ups, {Breath} {Pause} or I walked around two blocks. And by the time you do anything physical like that, {Pause} you – you've forgotten it. You d- – you no longer wanna go grab that food. {Pause} [At least I], **Therapist:** [Mhm] **(NR)** {Pause} **(NR)** **Participant:** At least I don't. It's over! The moment has passed. **Therapist:** Yeah. **(NR)** {Pause} **(NR)** Yeah. **(NR)** Exercise is a l- +lot – it's a lot better **(CR)** **Participant:** Yeah. **Therapist:** than – than eating. **(CR)**
Therapist: And how is your health right now? **(NR)** **Participant:** It's not the best. (( )) but they say I'm getting along. (( )) so that's what counts. **Therapist:** Yeah, so how old are you? **(NR)** **Participant:** Uh, eighty. Just turned eighty. **Therapist:** Okay, {Pause} so, your health is not the best **(SR)**

Table 1: Example annotations illustrating Simple, Complex and No reflections.

3. Annotation

Our annotation process is presented in this section. To compute inter-annotator reliability, as the first stage our annotators annotated ten of the acquired transcripts (Section 2.2.). These transcripts cover more than five hours of conversation, and contain a total of 1536 counselor utterances. They were acquired from multiple participants and form a representative portion of the collected data set (Section 2.1.). Each transcript was annotated by at least two annotators, and a total of three annotators participated in this task. The annotators label each counselor utterance[3] as: Complex Reflection (CR), Simple Reflection (SR), or No Reflection (NR). The remaining transcripts are being annotated by at least one of the three annotators (those annotations are still in progress). Table 1 depicts a sample annotation from the corpus. We describe the guidelines and tool employed for annotation below.

3.1. Guidelines

MISC and MITI are coding systems have been used extensively in prior work to manually annotate counselor and client utterances with behavioral codes. MISC is an exhaustive system that defines rules for assigning codes to both counselor and client utterances. MITI is built on MISC which solely targets counselor behavior. Since reflection is essentially counselor behavior, we use the MITI 4.2.1 standard (Moyers et al., 2014) as part of our guidelines while annotating reflections. Additionally, we define the following rules that supersede MITI to annotate our transcripts:

1. Utterances that are structured as reflections but, fall under a different behavioral code according to the MITI guidelines are also annotated as reflections. For instance, declarative questions and reflections are only distinctive with the tone in which the speech is delivered.

2. When a counselor provides a series of reflections in a single turn, all the utterances in the series are annotated as individual reflection.

3. If a client speech or noise interrupts a counselor's reflection, the utterance splits. In this case, both the utterances are coded with reflection.

3.2. Annotation Tool

We use GATE (General Architecture for Text Engineering) (Cunningham et al., 2013) to ease the annotation process and promote consistency. Annotation codes are enumerated in an *annotation schema*, an xml file. The GATE user interface is used by the annotators to map counselor utterances in the transcripts to the codes defined in the annotation schema. Finally, an output file enriched with annotations is generated.

4. Corpus Analysis and Quality

We analyze the code distribution, assess the quality of the annotated corpus, and present insights in this section. Table 3 presents the raw counts and percentage of code distribution annotated in the corpus. The total amount of reflection observed in the annotated transcripts is 8% (125) of total counselor utterances. Unlike companionbot

[3]In this work, we use the terms *sentence* and *utterance* interchangeably.

Work	Corpus	# Sessions	Duration	# Utterances	Agreement
(Can et al., 2012)	Drug abuse problem (HMCBI), alcohol use disorders (ESP21, ESPSB)	12	-	-	Overall: 0.661 - 0.764
(Pérez-Rosas et al., 2016)	Clinical Trails, Standardized patients, Brief MI encounters, Coaching phone calls	10	4.5 hrs	1160	CR : 0.49 SR : 0.34
Our Work	Conversational bot conversations	10	5 hrs	1536	Overall: 0.64 CR: 0.64 SR: 0.49

Table 2: Comparison of existing corpora annotated with behavioral codes with our work

conversations, MI corpora are entirely therapeutic and usually portray a higher percent of reflection.

Code	Count	Percentage
Complex Reflection (CR)	98	6.4%
Simple Reflection (SR)	27	1.7%
No Reflection (NR)	1411	91.9%

Table 3: Distribution of each code in the annotated corpus.

Code combination	Kappa
CR vs SR vs NR	0.64
(CR or SR) vs NR	0.69
CR vs (SR & NR)	0.64
SR vs (CR & NR)	0.49

Table 4: Inter-annotator reliability for different combination of codes expressed by Cohen's Kappa values.

We use Cohen's Kappa (Viera et al., 2005) to assess inter-annotator reliability in our corpus. Table 4 presents the Kappa values between several combinations of codes. The Kappa computed over whole annotated counselor utterances with three codes is 0.64 which is considered to be substantial agreement for categorical values (Landis and Koch, 1977). For a detailed analysis, we also report annotators' agreement on each code separately. Scores indicate that annotators identified any reflection (i.e., SR or CR), and only CR in transcripts, fairly well. We observed that three of the annotated transcripts contain zero SR and its overall frequency is low, which justifies the lower Kappa score obtained for SR.

To better perceive the agreements, we also present the normalized contingency matrix between the annotators for all codes in a heat map (Figure 3). High diagonal values in the heat map indicate that our annotations are reliable. The heat map also indicates minor ambiguity between CR and NR. This is because of the complexity involved in distinguishing the therapeutic dialogue from the non-therapeutic dialogue in some cases. For instance, consider the following sample turn from one of the annotated transcripts:

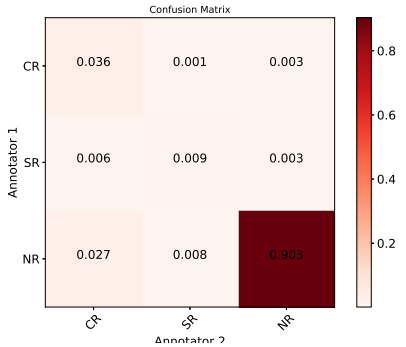


Figure 3: Heat map for the normalized contingency matrix between the labels of the two annotators.

Therapist: [Yes, you – you] said that he had children and the children live with him or the mom?
Participant: Uh, they stay split fifty-fifty.
Therapist: Okay, so they also get exposed to some of this trauma.

The sentence in the example of Therapist's last turn is a complex reflection in casual conversation.

5. Previous Work

Companionbots that can be used in health care at home need to engage in conversations that are both casual and therapeutic. Research on MI interventions that are solely therapeutic(Can et al., 2016; Pérez-Rosas et al., 2017; Tanana et al., 2015) is being actively pursued. Efforts focused on collecting MI session transcripts and annotating them with behavioral codes have become popular. In this section, we detail on how we expand on the existing resources by incorporating therapeutic dialogue within a non-therapeutic setting.

Mainly, we compare and contrast our novel annotated corpus of companionbot conversations with related corpora (Table 3.1.). (Can et al., 2012) is one of the prominent works that annotated both counselor and client utterances following MISC guidelines. This work reports inter-annotator agreement on 12 MI sessions with an overall agreement of 0.764 and 0.661 when considered at utterance-level and word-level, respectively. Authors mention the actual code agreement lies in between the

above specified scores. Moreover, (Pérez-Rosas et al., 2016) is another significant work that focused explicitly on annotating counselor behavioral codes following MITI guidelines. Inter-annotator agreement of 0.49 and 0.34 is reported for CR and SR codes, respectively. The characteristics of our annotated data set correspond with other existing corpora. Although it is comparatively difficult to detect the therapeutic reflections among casual and therapeutic utterances, it is evident from this comparison that our corpus is also a high quality indispensable resource.

The sensitive nature of Motivational Interviewing sessions do not permit the public release of the existing corpora in this area. However, we are constructing a corpus that can be distributed for on-going research.

6. Conclusion

In this work, we detail the collection of companionbot conversations and the process of annotating reflections on counselor utterances. The inter-annotator agreement indicates that the annotations have substantial quality. We exemplify any ambiguity present in the annotations and specify the reasons behind it. Our corpus addresses the need for a suitable corpus that would enable the development of companion bots and other dialogue systems focused on identifying and processing complex reflections in conversations with therapists. As the work in progress, at least one of the annotators are annotating the additional conversations. This corpus will be available for research upon request [4].

7. Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant 1262860. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

8. References

Arkowitz, H., Miller, W. R., and Rollnick, S. (2015). *Motivational interviewing in the treatment of psychological problems*. Guilford Publications.

Bickmore, T. W., Caruso, L., Clough-Gorr, K., and Heeren, T. (2005). 'it's just like you talk to a friend' relational agents for older adults. *Interacting with Computers*, 17(6):711–735.

Can, D., Georgiou, P. G., Atkins, D. C., and Narayanan, S. S. (2012). A case study: Detecting counselor reflections in psychotherapy for addictions using linguistic features. In *INTERSPEECH*, pages 2254–2257. ISCA.

Can, D., Marín, R. A., Georgiou, P. G., Imel, Z. E., Atkins, D. C., and Narayanan, S. S. (2016). âit sounds like...â: A natural language processing approach to detecting counselor reflections in motivational interviewing. *Journal of counseling psychology*, 63(3):343.

Cunningham, H., Tablan, V., Roberts, A., and Bontcheva, K. (2013). Getting more out of biomedical documents with gate's full lifecycle open source text analytics. *PLoS computational biology*, 9(2):e1002854.

Dahlbäck, N., Jönsson, A., and Ahrenberg, L. (1993). Wizard of oz studiesâwhy and how. *Knowledge-based systems*, 6(4):258–266.

Keeley, R. D., Burke, B. L., Brody, D., Dimidjian, S., Engel, M., Emsermann, C., Thomas, M., Moralez, E., Koester, S., Kaplan, J., et al. (2014). Training to use motivational interviewing techniques for depression: a cluster randomized trial. *The Journal of the American Board of Family Medicine*, 27(5):621–636.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Miller, W. R. and Rollnick, S. (2012). *Motivational interviewing: Helping people change*. Guilford press.

Miller, W. R., Moyers, T. B., Ernst, D., and Amrhein, P. (2003). Manual for the motivational interviewing skill code (misc). *Unpublished manuscript. Albuquerque: Center on Alcoholism, Substance Abuse and Addictions, University of New Mexico*.

Moyers, T., Manuel, J., and Ernst, D. (2014). Motivational interviewing treatment integrity coding manual 4.1. *Unpublished manual*.

Nielsen, R. D., Voyles, R. M., Bolanos, D., Mahoor, M. H., Pace, W. D., Siek, K. A., and Ward, W. H. (2010). A platform for human-robot dialog systems research. In *AAAI Fall Symposium: Dialog with Robots*.

Pérez-Rosas, V., Mihalcea, R., Resnicow, K., Singh, S., and An, L. (2016). Building a motivational interviewing dataset. In *CLPsych@HLT-NAACL*, pages 42–51. The Association for Computational Linguistics.

Pérez-Rosas, V., Mihalcea, R., Resnicow, K., Singh, S., Ann, L., Goggin, K. J., and Catley, D. (2017). Predicting counselor behaviors in motivational interviewing encounters. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 1128–1137.

Ring, L., Bickmore, T., and Pedrelli, P. (2016). Real-time tailoring of depression counseling by conversational agent. *Iproceedings*, 2(1):e27.

Sarma, B., Das, A., and Nielsen, R. (2014). A framework for health behavior change using companionable robots. In *INLG*, pages 103–107. The Association for Computer Linguistics.

Strassel, S. (2003). Slx corpus of classical sociolinguistics interviews.

Tanana, M., Hallgren, K., Imel, Z. E., Atkins, D. C., Smyth, P., and Srikumar, V. (2015). Recursive neural networks for coding therapist and patient behavior in motivational interviewing. In *CLPsych@ HLT-NAACL*, pages 71–79.

Vardoulakis, L. P., Ring, L., Barry, B., Sidner, C. L., and Bickmore, T. (2012). Designing relational agents as long term social companions for older adults. In *International Conference on Intelligent Virtual Agents*, pages 289–302. Springer.

Viera, A. J., Garrett, J. M., et al. (2005). Understanding

[4] Email authors for release details and to obtain the corpus.

interobserver agreement: the kappa statistic. *Fam Med*, 37(5):360–363.

Annotating Attribution Relations in Arabic

Amal Alsaif, Tasniem Alyahya, Madawi Alotaibi, Huda almuzaini, Abeer Algahtani

Al-Imam Mohammad Ibn Saud Islamic University
College of Computer Sciences and Information
{asmalsaif, tnalyahya, msalotaibi, hamozeani}@imamu.edu.sa;aakalqahtani@sm.imamu.edu.sa}

Abstract

We present a first empirical effort in annotating attribution in Modern Standard Arabic (MSA). Identifying attributed arguments to the source is applied successfully in diverse systems such as authorship identification, information retrieval, and opinion mining. Current studies focus on using lexical terms in long texts to verify, for example, the author identity. While attribution identification in short texts is still unexplored completely due to the lack of resources such as annotated corpora and tools especially in Arabic on one hand, and the limited coverage of different attribution usages in Arabic literature, on other hand. The paper presents our guidelines for annotating attribution elements: cue, source, and the content with required syntactical and semantic features in Arabic news (Arabic TreeBank - ATB) insight of earlier studies for other languages with all required adaptation. We also develop a new annotation tool for attribution in Arabic to ensure that all instances of attribution are reliably annotated. The results of a pilot annotation are discussed in addition to the inter-annotators agreement studies towards creating the first gold standard attribution corpus for Arabic.

Keywords: attribution, annotation tool, NLP, Arabic discourse, annotation guidelines, ATB, inter-annotator agreement.

1. Introduction

Textual information is one of the huge significant data available in the World Wide Web (WWW), which rapidly spread globally. This information is versatile and reflects different opinions as well as behaviours. For example, most of news is reporting people's speech and opinions about particular events, with additional explanation and analysis by the writer to tend people into desired understanding and background. Referring attributed arguments to the source and distinguishing different opinions than the author/writer opinion are not straightforward processes (Pareti 2012). Attribution could be direct speech (quotations) with no influence by the author on the content, or indirect speech when the abstract object is presented in a different way than the exact speech or it is not clear who state the argument. Usually quotation tools are used in direct speech such as punctuations (: and "), speech acts such *say/qAl/قال* or *comment/Akbar/أخبر*/and some particles such as that/An/أن, as in Example 1. On other hand, indirect speech may miss one or more of these tools which leads to ambiguity on determining the exact source and content boundaries, Examples 2 and 3. This makes identifying quotations and opinions of other people require advanced analysis and tools in addition to the basic lexical-syntactic analysis in the literature (Pareti 2012).

(1)
قال مدرب مانشستريونايتد مورينيو [بخصوص المباراة]:"*كانت حصة تدريبية جيدة جدا* ".
Manchester United coach Mourinho **said** [about the match]: "*it was a very good training share"*

(2)
علق مصدر مسؤول في الحكومة {رفض ذكر اسمه} [على تصريح الرئيس المتضمن رفع اسعار البترول] *أن هذا القرار من شأنه زعزعت الدخل العام للاسر*.
An official source in the government{refused to mention his name} **commented** [on the president's statement, which included raising oil prices], *that this decision would affect public income for families*

(3)
أكد السويسري كريستيان جروس {الذي يعتبر مدرب الفريق الأول لكرة القدم}، [عند اجتماعه باللاعبين]**على** *أهمية المباراةاليوم*.
Swiss coach Christian Gross, {who is considered the coach of the first football team}, he **confirmed** [when meeting with the players**] on** *the importance of the game today*.

Attribution annotation has recently received significant attention in Natural Language Processing (NLP) due to its relevance in particular to information extraction; question answering, story generation, summarization, and opinion mining (Guzmán-Cabrera et al. 2009; Juola and Baayen 2005; Neumann and Schnurrenberger 2009; Wiebe 2002). Most of these studies dealt with attribution in English. However, to the best of our knowledge, there are no empirical studies on annotating attribution in Arabic to generate a gold standard corpus. We propose in this paper our approach of identifying attribution in News corpus of contemporary Arabic, the Arabic Treebank (Maamouri et al. 2004) of both direct and indirect quotations. The work is inspired by a large discourse annotation project, the PDTB (Prasad et al. 2006; Prasad et al. 2007) that annotate attribution that serve discourse relations in English. The attribution annotation was extended by Silvia and her colleagues (Pareti et al. 2013) in annotation attribution regardless the existing of discourse relations.

The rest of the paper is structured as follows: Section 2 overviews the popular similar attribution annotation efforts and related applications. Section 3 presents a brief review about attribution in Arabic literature. Our schema of annotating the main elements and semantic features of attribution is discussed in section 4. ESNAD annotation tool and the human annotation process are discussed in Section 5. Then, Section 6 presents the pilot annotation of 20 news articles from different resources with a discussion about the agreement studies between annotators. We conclude with observations and actions required for creating the first gold standard corpus of attribution in Arabic.

2. Related work

Over the literature, there have been several studies that addressed the attribution in computational linguistics, mostly for English. These studies varied on considering one or more aspects of a lexical, syntactical, and semantic structure of automatic quotations, with very limited effort on indirect quotations. For example, work performed by (Mamede and Chaleira 2004; Elson and McKeown 2009) on narrative text and (Pouliquen, Steinberger, and Best 2007; Sarmento, Nunes, and Oliveira 2009; Schneider et al. 2010) on news text are based on lexical terms and some syntactic rules to infer the author of the quoted text. Experimental evidence in these studies clearly indicates the unreliability on identifying all attribution instances. Later, the work by (Elson and McKeown 2010; Fernandes, Motta, and Milidiú 2011; O'Keefe et al. 2012) had better success due to the fact they were based on NLP approaches such as rule-based and statistical machine learning of syntactical structure feature. Other studies consider specific types of attribution such as opinions at sentence level in the Multi-Perspective Question Answering MPQA (Wiebe 2002). The set of features used in annotating the MPQA includes: on, inside (content), and outside. The source is not annotated independently; it is labelled as 'outside' together with everything in the sentence other than the cue and the content.

The limitation of used data limits feature extraction too. (Elson and McKeown 2010) used a corpus consisting of about 3,000 quotes, and manually identified candidate features of the speaker mainly their gender and attributes each quote to the most likely speaker. This proposed approach achieved an average accuracy of 83%. In (Fernandes, Motta, and Milidiú 2011), they used a corpus contains annotation of entities, co-references, quotes, associations between quotations and authors, and part-of-speech tagging features. The performance achieved for author attribution was 79,02%. While the dataset in (O'Keefe et al. 2012) used the same corpus of (Elson & McKeown, 2010) and adds about 5,000 attributions from the PDTB which is news from the Wall Street Journal (WSJ) and Sydney Morning Herald (SMHC) that has been annotated with over 3,500 direct quotations and their speakers, but the cue element is not annotated since the cue can be inferred implicitly. The accuracy of this system was 84,1% and 91,2% for WSJ and SMH respectively. As concluded by (Pareti 2016), there is a lack of a comprehensive theory of attribution and a large gold standard annotated corpus with all attribution elements and features, which clearly influence on a performance of machine learning systems to identify attribution elements: the quote, its source, the purpose of reporting, its cue and its truthful level.

A part from PARC3 corpus (Pareti 2016), attribution was not the core of most annotated corpora. It was integrated in a limited extend with other discourse phenomena such as factuality in FactBank and discourse relations in RST and the PDTB. In the FactBank (Saurí and Pustejovsky 2009), the attributed span itself is not marked, but its events are linked to their source by introducing predicates in order to derive their factuality. Consequently, the annotation schema resulted in a relatively high inter-annotation agreement %81. The work in (Carlson and Marcu 2001) on the RST discourse treebank and (Wolf and Gibson 2005) the GraphBank projects consider attribution as a discourse relation. However, the first annotates only intra–sentential attributions with an explicit source and a verb cue, the latter annotates attribution if no other discourse relation is present. Inter-annotation agreement was evaluated and reported in all three copra. FactBank have resulted in a relatively high inter-annotation agreement (kappa=0.8). The results of the inter-annotator agreement within the RST framework was tested by multiple judges during multiple phases of the development of the RST corpus. Kappa values of the RST and the GraphBank reflects considerably high levels of agreement, greater than 0.8.

The large existing resource for annotating attribution as inter-sentential discourse relation is in the PDTB project (Prasad et al. 2006; Prasad et al. 2007) with 10K annotation of attribution. Attribution is a relation between abstract object and the source entity that must relate to one of discourse relations. However, this approach leaves out several instances of attribution and therefore some related features, e.g. nested attribution with no annotation. They annotate features for each accepted attribution in their scheme such as: type, source, determinacy, and polarity; Where the type may indicate one of the four distinct sub-types: assertion proposition, belief proposition, facts and, eventualities. An analysis of inter-annotator agreement was conducted on the PDTB corpus. The high inter-annotator agreement achieved indicates that discourse connectives and their arguments expose a well-defined level of discourse structure that can be reliably annotated.

The PDTB paradigm was applied similarly to other languages such as Chinese (Zhao and Zobel 2005; Huang and Chen 2011; Zhou and Xue 2012; Zhou et al. 2014; Zhou and Xue 2015), Arabic (Al-Saif and Markert 2010), Hindi (Kolachina et al. 2012), Czech (Mírovský, Jínová, and Poláková 2014), and Turkish (Zeyrek and Kurfalı 2017). Attribution in all these studies is not embraced currently in the annotation scheme and left to be extended in the future. In (Li et al. 2014) a Connective-driven Dependency Tree (CDT) structure as a representation scheme for Chinese discourse structure is proposed. CDT takes advantage of both RST and PDTB, and well adapts to the special characteristics of Chinese discourse relation including attribution. Later in (Kong and Zhou 2017), a CDT-styled end-to-end Chinese discourse parser was developed.

Further extension of the PDTB to annotate attribution was proposed by Pareti and her colleagues to annotate direct and indirect attribution in a comprehensive coverage in the Italian Attribution Corpus (ItAC) (Pareti and Prodanof 2010) and the Penn Attribution Relation Corpus (PARC3) (Pareti 2016). The ItAC is a small-scale Italian pilot corpus (461 instances in 50 articles) annotating key features of attribution: the source, cue, content, supplement, and additional features. Still, the schema needs to be tested for inter-annotator agreement. PARC 3.0 is the first large English corpus fully annotated with 19,712 attribution relations. It is initially grounded on the annotation in the PDTB, in addition to annotating new instances not related to discourse relations and annotate nested attributions. The attribution components include: source, cue, content, and supplement with a set of features included in the PDTB: attribution type, source type, factuality, and scopal polarity. While the annotation schema proposed in ItAC has not yet

been validated by inter-annotator agreement, the inter-annotator agreement results for the annotation of the spans in PARC 3.0 corresponding to source, cue, content and supplement are reported to be 100%, 91%, 94%, and 46% respectively. The ItAC and PARC3 are the only corpora have allowed the identification of several attribution structures not addressed by former studies.

Among a few studies of authorship identification in Arabic (Ouamour and Sayoud 2013; Rabab'ah et al. 2016) a recent study that addressed Arabic authorship identification on short text was conducted by (Rabab'ah et al. 2016) on 38,386 tweets for 12 users. Using SVM classifier with lexical and syntactic features, their system achieved accuracy of 68.67% on assigning each tweet with corresponding author. This study focused on the quote (tweet) and the source (the user) only with no use of other attribution elements and features, or nested attribution.

3. Attribution in Arabic

Arabic is categorised into either classic Arabic (CA) or contemporary Modern standard Arabic language (MSA). Both are sharing main characteristics of Arabic morphology, syntax and semantic. The differences lay on the usage and the level of construction with new vocabularies throw generations. MSA is used nowadays in books, education, news, but not always used in spoken language due to the effects of dialects of different regions (Habash 2010). Arabic NLP studies use mostly MSA, especially in information analysis and retrieval. Arabic processing requires deep multi-layer text processing in tokenization, stemming, morphological, syntactical analysis, and discourse (Farghaly and Shaalan 2009). In Example 4, the verbal sentence 'then they will read it' is represented in Arabic as one white spaced word (فسيقرأونها/ fsyqrAwnha/then they will read it) with many clitics: one proclitic, one prefix, one postfix and one enclitic are all attached to the stem يقرأ/yqrA/read. Moreover, most written text in MSA lacks using punctuations and diacritics (long vowels) in standardized manner, leading to high ambiguity in any automatic text processing.

(4) Gloss and syntactic analysis of a token
فسيقرأونها / then they will read it:

ف /then (connective)
+س / will (tense)
+ يقرأ /read (present verb)
+ ون/they (subject)
+ ها /it (object)

Traditional literature in semantic science/علم المعاني discusses speech reporting (أسناد /نقل الكلام/رواية) as one of the writing styles and study its basic structure using lexical, syntactic, or pragmatics features not connected to any discourse theories (Ali 2004). Similar to other languages, reporting others speech may use direct (exact words) or indirect quoting (not exact words but keeping the semantic). The content is usually enclosed in quotation or (:) marks to be direct, or precede by adverb (أن/An/that) to be indirect quoting. In fact, indirect reporting has many forms: changing lexical words and syntax of the original text, may use (أن/An/that) or not, may have two explicit verbs (speech act and a verb in the verbal phase in the content), or one verb serving the two verbs purpose with implicit declaration speech act. The writer shows his style when reporting other speech using indirect quotation, he may emphasise on some points or to tend the reported argument to be evidence to specific claims, which sometimes differ that the source intention and so making rumours. For example, He said "I'm going early" is a direct reporting, whereas He said (that) he was going early is indirect speech reporting. Modern Arabic studies barrow some theories from other languages that focus on the usage and semantics of reporting speech, and how relate to the source and its writing style. For example, the famous debatable theory "the Speech Acts theory/نظرية أفعال الكلام" by Searle (Searle 1976) that was adapted from Austin theory (Austin 1975). They claimed that speech acts are used to express the purpose of the quoting by the author. For example, the semantic of say/قال is different than assure/شدد even though the quote is identical. According to Searle, there are two types of Speech Acts: *constatives (declaratives) and performatives*. Constatives are used for propositions which can be stated to the truth value. For example, he said: *sky is blue/السماء زرقاء* :قال the proposition presents a status which can be assessed to the truth at that time with no action by the source.

Performative verbs express specific action belong to a purpose such as: Assertives (i.e. suggest/AqtrH /اقترح), Directives (i.e. ask/اسأل), Commissives (i.e. promise/ wEd /وعد) in Example 5, Expressives (i.e. forgive/AEtdr/ /اعتذر), and Declaratives (i.e. confirm/Akd/أكد). A paper by (Bouayad and Belkheer 2012) presents a collection of most frequent speech acts in Arabic. However, there is no classification of direct/indirect speech acts in use, and no clear distinguishing between implicit and explicit attribution, some books mix them with direct/indirect speech.

(5)

أعدك *أن أحضر باكرا .*

*I **promise** you that I will come early*

Although, these studies criticise that Speech Acts theory is incomplete and speech acts may lie under more than one class, there is no special Arabic theory would cover this lack to be used in a complete empirical study of attribution. Further, there is no guideline for annotating the quotations (direct and indirect) in textual corpus that could be used on machine learning modelling. This highlights the importance of analysis of a corpus-based study of attribution in Arabic, which might be used to prove and extract new classification of attribution types and analyse new indirect reporting styles in use.

4. Attribution schema of Arabic

This section describes our first attempt in annotating attribution relation in MSA. The ground base of the proposed annotation is the work of the PDTB (Prasad et al. 2006) and Silvia in (Pareti and Prodanof 2010; Pareti 2011; Pareti 2012) but tailored and extended to suit the MSA. We share with other studies the interest in identifying the basic elements of attribution: cue, the source, and the content. Authors while paraphrase the others speech may use temporal cues and locations to increase the truth of their reporting or describing the entity in more details, so called supplement information. Each basic elements may have

supplement. The author, also, may use implicit indirect speech and increase the ambiguity in determining the source of the claim, the author or others. The schema is broken down into four main classes with clear definition and examples of each: the cue, attribution general features, source, and content. Each class has further features to cover all semantic aspects.

Examples in this paper are presented according to the following convention: the cue is (bold-faced), the source (underlined), the content (italic), the cue supplement (enclosed in brackets), the source supplement (enclosed in braces), and the content supplement (enclosed in parentheses).

4.1 Cue

The cue can be defined as the lexical anchor that links the source with the content. The cue may occur in different syntactic forms: a reporting verb/speech act either *constatives* تصريحية or *performatives*انجازية (such as emphasis/Akd/أكد in Example 3), adverb (such as adding/ mDyfA/مضيفا in Example 6), an adjective (such as describing/واصفا), and a prepositional phrase (such according to/بحسب). The cue also can be omitted such as (Ahmad: I will go/أحمد: سوف أذهب) which is understood from the context as *said "…"*.

(6)
مضيفا *" هذا ما فكرنا به عندما كنا نخطط لاستعدادات الموسم الجديد "*.
He **added** *"That's what we thought of when we were planning for the new season."*

Cue status is a feature indicates whether the cue is explicitly occurred in the text (explicit) or omitted (implicit). *Explicit* cue is usually one of the declarative reporting acts (such as say/قال, mention/ذكر and declare/صرح), with direct speech when the content is introduced by punctuation marks (: or ""), or with indirect speech when it is not clear whether the content has exact words of actual speech. The particle (that /ان) could be used for indirect speech. On other hand, *implicit* cue is a feature when the cue is either omitted or being a *performative* speech act with indirect speech only such as (*he thanks his teacher/شكر معلمه, he suggests not to answer/ اقترح عدم الاجابة, he promised to his dad coming early/ وعد اباه العودة باكرا*). The reason behind counting these speech acts as implicit cue is that ability of converting them into explicit cues by adding one of reporting acts which are declaration acts (such as say/قال, mention/ذكر, declare/صرح) and the exact speech in the content but the writer preferred to make them implicit. For example, *he said thankfully to his teacher "…."/…. :قال شاكرا لمعلمه, he said I suggest to not answer/قال اقترح عدم الاجابة and he said to his dad I promise to come early/قال واعدا أباه انه سيعود باكرا*). Examples 1 has explicit cue with direct speech, Example 2 has explicit cue with indirect speech and while Example 3 has implicit cues with indirect speech.

Cue supplement is the text span that describe the status of the cue (when?where?how?) which is relevant to the interpretation of the cue such as adverb *laughing/ضاحكا*, temporal phrase, or place of the attribution cue. Some modifiers such as prepositions as *on/على , text on/ نص على* are tagged as part of the cue, not as supplement.

Cue negation determines if the cue is modified by a negation tools (e.g., did not/lm/ لم) or the cue itself indicates negation semantically (i.e *denied/rfD/رفض*).

Cue digression when the cue digresses a former quotation such as in Example 6 where the attribution could not be stand alone at the first place.

4.2 Source

The Source is the entity holding the content. As in the PDTB (Prasad et al. 2006; Prasad et al. 2007) , the source is annotated by marking a text span expressing the source and also its type. Syntactically the source will be the subject of the declaration verb either in explicit or implicit cue. The source type express diverse types of agents: (i) the writer of the text (Mustafa 2011) - the writer is reporting someone lese speech directly, (ii) any specific agent other than the writer either explicitly occurred in the text (EXP-AG) such as in Example 1, (iii) the source in not explicitly appear in the same sentence of the attribution but could be inferred from previous context, this tag as an implicit agent (IMP-AG) such as in Example 6, (iv) or the source is anonymous and the writer did not refer this speech to a specific agent (Miss) as in Example 7 . Our new feature here is **the source supplement** to tag any expression or relative clauses related to the source, sometimes the writer prefers to describe the agent in more details as in Examples 2 and 3.

(7)
يذكر *أن السعودية تواصل لليوم الثالث تسيير رحلاتها المتتابعة داخلياً ودولياً بكل انسيابية.*
It is **said** that *Saudi Arabia continues for the third day running its successive flights internally and internationally smoothly.*

4.3 Content

The attributed material is annotated as **a content feature,** by determining the text span boundaries that might range from only one word into multiple sentences. The content may cover the cue too if the cue plays as basic verb in the content such as in implicit indirect attribution (Khaled congratulated his brother on success/هنأ خالد أخيه على النجاح) **Content supplement:** any clauses the writer added to the content to present some background information about the entities or events in the content but is mostly not part of the reported speech. For example, the relative clause in Example 8 *those who use their mobiles while driving/ الذين يستخدمون الجوال أثناء القيادة* is part of the content and not supplement. **Content negation**: the content or attributed speech is negated when using either negation function words such as *did not/لم and will not/لن*as in (Ahmad said I will not go to the school/قال أحمد لن أذهب للمدرسة) and using a noun *none/عدم* in Example 9, or using negation verbs or nouns such as refusal/الرفض and denial/انكار in the content itself.

(8)
طالب السيالي[إدارة المرور] *بملاحقة مشاهير السناب شات الذين يستخدمون الجوال أثناء القيادة.*
Al-Sayali **requested** [the Traffic Department] *to track down the snap celebrities* who use mobile phones while driving.

(9)
ذكر الحمادي: *عدم تأهل نادي الهلال لبطولة أسيا.*

Al-Hammadi **said**: *Al-Hilal is not qualified for the Asian Championship.*

4.4 General features

Apart from the text span features (cue, source, content and their supplements), the attribution relations have further key semantic features that might be used in discourse processing and information/opinion extraction systems. The general features include: *attribution style, determinacy, and attribution purpose.* **Attribution style** distinguishes whether the content is quoted with exact words of the spoken (direct) as Example 1, or reports someone else speech without using that person's exact words (indirect) as in Example 2. Implicit cues often use indirect speech, while explicit cues could use either direct or indirect. **Determinacy feature**, was borrowed from the PDTB, it identifies the factuality of the attribution relation itself; not the content. The feature will be indeterminacy (Non-Factual) when the relation in the scope of hypothesis, negation, or future tense such as using (will/سوف or may-perhaps/ربما) as modifier to the cue, see Example 10. In hypothesis, the conditional function words such as *if* /لو is used to express the factuality of attribution, as in Example 11.

(10)

إن فتشنا في الملفات ربما **يقول** الجيل الجديد *أين الرقابة، وأين المسؤول، وأين الإعلام؟*

If we look at the files, the new generation might **say**, "*Where is censorship, where is the administrator, and where is the media?"*

(11)

لو **أخبرنا** *أنه سيعبر أجواءنا ،كنا سنحتفي به.*

If he **told** us that *he will pass our atmosphere*, we would celebrate him.

Attribution purpose, while the purpose of the speech reporting is transfer the news, facts, or stories, this feature signifies the nature of the relation between an agent and the cue, it describes the reason of using this particular cue in reporting the speech. Our annotation guidelines tried to base on well-established linguistic theories as possible. As mentioned earlier we use the classification of speech acts (cues) in the Speech Acts Theory (Searle 1976) and it is application on Arabic to determine the purpose of the reporting. Our taxonomy has flat distribution of five distinct purposes of attribution: (i) **Assertion** when commit to the truth of the proposition (e.g. said/قال , assert/أكد/جزم, mention/dkr/ذكر), (ii) **Directive** for requesting (ask/سأل, order/أمر or request/طلب) or questioning (question /استفهم), (iii) **Expression** purpose is to express a feeling or regretting (e.g. apologies /اعتذر, congratulate/هنأ), (iv) **Declarative** to declare changing on affairs (e.g. announced/ أعلن, informed/أبلغ, admitted/اعترف and stated/أفاد). (v) **Commissive** acts express any commitments **(**e.g. bet/راهن, promise/وعد, and oath/ أقسم). Assertion and Declarative speech acts often used to express explicit attribution direct/indirect. While other purposes (Directive, Expression, and Commissives) are commonly used for implicit indirect attribution.

5. Building the first attribution corpus for Arabic

5.1 Corpus

We planned to enhance the discourse layer in the Leeds Arabic Discourse TreeBank (LADTB) (Al-Saif and Markert 2010), the valuable discourse resource for Arabic, by annotating explicit, implicit, direct and indirect attribution in 530 news articles from Arabic treebank ATB-Part1 (Maamouri et al. 2004). The ATB has morphological and syntactical gold standard annotation, used in many studies in Arabic NLP community. Adding our annotation to this corpus will encourage further studies on computational linguistics.

5.2 Annotation tool

Annotation tools share the graphic-based visualization that inspire users to gather complex annotations in an easy and reliable technique. While there are some general purpose annotation tools such as the GATE tool (Ide and Suderman 2009), BRAT tool (Stenetorp et al. 2012), MMAX2 tool (Müller and Strube 2006), and WebAnn (Yimam et al. 2013), few of them support relational annotation and Arabic calligraphy. We therefore, decided to develop a new Java-based annotation tool for Arabic (ESNAD: Extracting Sentence Attribution in Arabic Discourse) with a user-friendly interface to ensure highly reliable annotation which could be used for similar languages such as Urdu, see Figure 1.

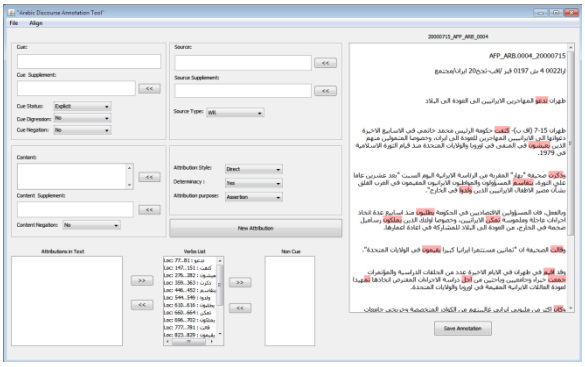


Figure 1: The main interface of ESNAD with initial highlighting of all verbs.

5.3 Human annotation process

The ESNAD tool will highlight all verbs (extracted from the ATB) and marked them as potential attribution cues. The annotator distinguishes between verbs presenting speech acts or not which are therefore not attribution. The annotator still has an ability to mark any cue such as adverbs or clauses from text on the right. The annotator should follow the annotation schema presented in Section 4 and annotate basic attribution elements and their features for direct/indirect quotation and explicit/implicit cue. We designed the tool to prevent any data entries by the user: so either marking desired text span on the text on the right side, or selecting a label from predefined labels as in our scheme. The tool saves all annotation into a text file to conduct the inter-annotator agreement and produce a gold standard annotated corpus by expert verification of disagreed cases. We conduct a pilot annotation study on 20

news articles from ATB-Part1 and other news websites to validate our guidelines. Annotators are native Arabs with a good linguistic background. The distribution of our annotation and observations are discussed in the next section.

6. Pilot annotation study

The annotation was conducted by 2 annotators (4 different Arabic speakers, 10 files each pair) after 2 rounds training on different 5 files using the tool and the schema. The inter-annotators agreement is measured using observed agreement, f-score and kappa to take into account agreement by chance (Siegel and Castellan 1988), as in Table 1. Among 734 potential instances, annotators agreed on 98% (161 as attribution cases and 534 as not attributed verbs). A high reliable annotation is recorded for the cue, its status (implicit/explicit) and attribution style (direct/indirect). A good agreement is recorded for attribution purpose and source type (~85%). The low kappa of the source type turns our attention on a high agreement by chance. This result is justified by being the default label of the source is the writer in the tool for each instance and the annotator should change it when appropriate, but he missed doing so for this feature. As a result, we will make this feature without default value in the tool.

	Accuracy	**F-score**	**Kappa**
Attribution Cue (734)	98%	0.96	0.95
Attribution Style (161)	94%	0.96	0.88
Cue Status (161)	94%	0.96	0.80
Attribution Purpose (161)	85%	0.79	0.76
Source Type (161)	83%	0.76	0.61

Table 1: inter-annotator agreement on label features

For text features we use the agreement measurement agr; it is introduced in (Wiebe, Wilson, and Cardie 2005)and used in many annotation studies. As in Equation 1, *agr* is an average of agr-annotror1 and agr-annotator2 when each one is calculated by dividing number of matching words of the two text annotations by the total number of words of that annotation.

*agr=1/i *∑((# of matching words)/(# of total words in ann(i)*

Table 2 shows highly reliable agr agreement of all text features with higher that 96%. Not surprisingly that supplement features of (cue, source and content) have more disagreed instances because we did not limit them to temporal or relative clauses only. Thus, annotator sometimes is confused whether include the clause into cue, source or content themselves or leave them to the supplements. For example, prepositions such as (*assert to the press/صرح للصحفيين*, *pointed to/أشار إلى*) in cue supplement, or in source supplement. All disagreed cases are discussed intensively with all annotators to clarify the annotation guidelines and to increase the usability of the annotation tool. Cases that are still debatable are verified by third expert who is not involved initially in this manual annotation. The fine-grained features in our pilot study increases the challenge of automatic identification of full

Text Span Agreement (agreed attribution=161)	
Cue supplement	90%
Source	97%
Source supplement	97%
Content	97%
Content supplement	96%

Table 2: inter-annotator agreement of text features using agr metric

attribution elements in short text. The pilot study results a mini-annotated corpus, Table 3 presents the annotation distribution. From the few articles (20) we found 161 instances of attribution only 60 of them are direct quotations, the rest is indirect (paraphrasing). As expected in news text 80% of the instances are declaration/assertion attribution. Supplement features are used frequently in news to present the background information related to entities and events in the augmented element. The cue used in implicit attribution are mostly indirect and influenced by the writer intention of reporting specific news. The purpose and the source of attribution may be used on validation the news and a level of reality. While there is no defined list of speech acts in Arabic for each class of purposes, we expect the disagreement will continue on this feature. We plan to study the list of cues we have in current mini corpus in terms of ambiguity and provide it to the annotators in the next phase.

Attribution instances	161		
distinct cues	85		
Explicit cue	130	**Direct At**	60
Implicit cue	31	**Indirect At**	101
Common purposes	Declaration(72), Assertion(62), Expression(14),		
Common used cues	say/qAl/قال, declare/A'gn/أعلن, add/ATaf/أضاف		
Supplements	Cue(49), source(37), content(10)		

Table 3: Pilot corpus of attribuation in Arabic

7. Conclusion

Attribution annotation in Arabic requires a comprehensive analysis of contemporary corpus due to the lack of resources in Arabic linguistics evaluating different kinds of quotation and reporting others speech and opinions. We propose annotation guidelines and annotation tool to build the first attribution corpus for MSA news articles in particular. Insight of our pilot annotation experience, the guidelines and the annotation tool are slightly adapted before conducting a full annotation of attribution in the ATB. The corpus will be a valuable resource for authorship, rumour identification, Named Entity recognition with source feature, speech acts, and sentiment and polarity systems.

8. Acknowledgment

This work is funded by the deanship of scientific research at Al-Imam Mohammad Ibn Saud Islamic University,

Saudi Arabia, No. 370904. The team is grateful to Dr. Gihan Issa, Prof. Bonnie Webber, and Dr. Fatma Alshihri for the valuable linguistic discussion.

9. Bibliographical References

Al-Saif, Amal, and Katja Markert. 2010. "The Leeds Arabic Discourse Treebank: Annotating Discourse Connectives for Arabic." In LREC.

Ali, Mohammed Mohammed Younis. 2004. Introduction to the science of significance and communication (New United Book House).

Austin, John L. 1975. 'How to do things with words (JO Urmson & M. Sbisa, Eds.)', Harvard U. Press, Cambridge, MA.

Bouayad, Nawara, and Omar Belkheer. 2012. Categorize speech actions in journalistic discourse Algerian written in Arabic, Journal of Al-Athar

Carlson, Lynn, and Daniel Marcu. 2001. Discourse tagging reference manual, ISI Technical Report ISI-TR-545, 54: 56.

Castor, A. and Pollux, L. E. (1992). The use of user modelling to guide inference and learning. Applied Intelligence, 2(1):37–53.

Elson, David K, and Kathleen McKeown. 2010. "Automatic Attribution of Quoted Speech in Literary Narrative." In AAAI. Citeseer.

Elson, David K, and Kathleen R McKeown. 2009. "A tool for deep semantic encoding of narrative texts." In Proceedings of the ACL-IJCNLP 2009 Software Demonstrations, 9-12. Association for Computational Linguistics.

Fernandes, William Paulo Ducca, Eduardo Motta, and Ruy Luiz Milidiú. 2011. "Quotation extraction for portuguese." In Proceedings of the 8th Brazilian Symposium in Information and Human Language Technology (STIL 2011), Cuiabá, 204-08.

Grandchercheur, L.B. (1983) Vers une modélisation cognitive de l'être et du néant. In S.G Paris, G.M. Olson, & H.W. Stevenson (Eds.), Fondement des Sciences Cognitives. Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 6-38.

Guzmán-Cabrera, Rafael, Manuel Montes-y-Gómez, Paolo Rosso, and Luis Villasenor-Pineda. 2009. Using the Web as corpus for self-training text categorization, Information Retrieval, 12: 400-15.

Huang, Hen-Hsen, and Hsin-Hsi Chen. 2011. "Chinese Discourse Relation Recognition." In IJCNLP, 1442-46.

Ide, Nancy, and Keith Suderman. 2009. "Bridging the gaps: interoperability for GrAF, GATE, and UIMA." In Proceedings of the Third Linguistic Annotation Workshop, 27-34. Association for Computational Linguistics.

Juola, Patrick, and R Harald Baayen. 2005. A controlled-corpus experiment in authorship identification by cross-entropy, Literary and Linguistic Computing, 20: 59-67.

Kolachina, Sudheer, Rashmi Prasad, Dipti Misra Sharma, and Aravind K Joshi. 2012. "Evaluation of Discourse Relation Annotation in the Hindi Discourse Relation Bank." In LREC, 823-28.

Kong, Fang, and Guodong Zhou. 2017. A CDT-styled end-to-end Chinese discourse parser, ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), 16: 26.

Li, Yancui, Wenhe Feng, Jing Sun, Fang Kong, and Guodong Zhou. 2014. "Building Chinese Discourse Corpus with Connective-driven Dependency Tree Structure." EMNLP, 2105-14.

Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. "The penn arabic treebank: Building a large-scale annotated arabic corpus." In NEMLAR conference on Arabic language resources and tools, 466-67.

Mamede, Nuno, and Pedro Chaleira. 2004. Character identification in children stories. in, Advances in natural language processing (Springer).

Mírovský, Jiří, Pavlína Jínová, and Lucie Poláková. 2014. "Discourse Relations in the Prague Dependency Treebank 3.0." In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations, 34-38.

Müller, Christoph, and Michael Strube. 2006. Multi-level annotation of linguistic data with MMAX2, Corpus technology and language pedagogy: New resources, new tools, new methods, 3: 197-214.

Mustafa, Mansour. 2011. The theory of verbs in the imaginary discourse between Searle and genes, Journal of Al-Athar, 12.

Neumann, Hendrik, and Martin Schnurrenberger. 2009. E-Mail authorship attribution applied to the Extended Enron Authorship Corpus (XEAC) .

O'Keefe, Tim, Silvia Pareti, James R Curran, Irena Koprinska, and Matthew Honnibal. 2012. A sequence labelling approach to quote attribution. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 790-99. Association for Computational Linguistics.

Ouamour, S, and Halim Sayoud. 2013. "Authorship attribution of short historical arabic texts based on lexical features." In Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on, 144-47. IEEE.

Pareti, Silvia. 2011. Annotating attribution relations and their features. In Proceedings of the fourth workshop on Exploiting Semantic Annotations in Information Retrieval, 19-20. ACM.

Pareti,Silvia. 2016. PARC 3.0: A Corpus of Attribution Relations, LREC.

Pareti, Silvia 2012. "A Database of Attribution Relations." In LREC, 3213-17.

Pareti, Silvia, Timothy O'Keefe, Ioannis Konstas, James R Curran, and Irena Koprinska. 2013. Automatically Detecting and Attributing Indirect Quotations. In EMNLP, 989-99.

Pareti, Silvia, and Irina Prodanof. 2010. Annotating Attribution Relations: Towards an Italian Discourse Treebank. In LREC.

Pouliquen, Bruno, Ralf Steinberger, and Clive Best. 2007. Automatic detection of quotations in multilingual news.

In Proceedings of Recent Advances in Natural Language Processing, 487-92.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Aravind K Joshi, and Bonnie L Webber. 2006. Attribution and its annotation in the Penn Discourse TreeBank, TAL, 47: 43-64.

Prasad, Rashmi, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. 2007. The penn discourse treebank 2.0 annotation manual .

Rabab'ah, Abdullateef, Mahmoud Al-Ayyoub, Yaser Jararweh, and Monther Aldwairi. 2016. "Authorship Attribution of Arabic Tweets." In AICCSA

Sarmento, Luis, Sergio Nunes, and E Oliveira. 2009. "Automatic extraction of quotes and topics from news feeds." In 4th Doctoral Symposium on Informatics Engineering.

Saurí, Roser, and James Pustejovsky. 2009. FactBank: a corpus annotated with event factuality, Language Resources and Evaluation, 43: 227.

Schneider, Nathan, Rebecca Hwa, Philip Gianfortoni, Dipanjan Das, Michael Heilman, Alan Black, Frederick L Crabbe, and Noah A Smith. 2010. Visualizing topical quotations over time to understand news discourse .

Searle, John R. 1976. A classification of illocutionary acts, Language in society, 5: 1-23.

Siegel, Sidney, and NJ Castellan. 1988. Nonparametric systems for the behavioural sciences, McGraw Hill International Editions.

Stenetorp, Pontus, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. "BRAT: a web-based tool for NLP-assisted text annotation." In Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, 102-07.

Strötgen, J. and Gertz, M. (2012). Temporal tagging on different domains: Challenges, strategies, and gold standards. Proceedings of the twelve International Conference on Language Resources and Evaluation (LREC12), pages 3746–3753, Istanbul, Turkey, may.

Superman, S., Batman, B., Catwoman, C., and Spiderman, S. (2000). Superheroes experiences with books. The Phantom Editors Associates, Gotham City, 20th edition.

Wiebe, Janyce. 2002. Instructions for annotating opinions in newspaper articles .

Wiebe, Janyce, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language, Language Resources and Evaluation, 39: 165-210.

Wolf, Florian, and Edward Gibson. 2005. Representing discourse coherence: A corpus-based study, Computational Linguistics, 31: 249-87.

Yimam, Seid Muhie, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. 2013. "WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations." In ACL (Conference System Demonstrations), 1-6.

Zeyrek, Deniz, and Murathan Kurfalı. 2017. TDB 1.1: Extensions on Turkish Discourse Bank, LAW XI 2017: 76.

Zhao, Ying, and Justin Zobel. 2005. Effective and scalable authorship attribution using function words. In Asia Information Retrieval Symposium, 174-89. Springer.

Zhou, Lanjun, Binyang Li, Zhongyu Wei, and Kam-Fai Wong. 2014. "The CUHK Discourse TreeBank for Chinese: Annotating Explicit Discourse Connectives for the Chinese TreeBank." In LREC, 942-49.

Zhou, Yuping, and Nianwen Xue. 2012. PDTB-style discourse annotation of Chinese text. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, 69-77. Association for Computational Linguistics.

Yuping Zhou Nianwen Xue. 2015. The chinese discourse treebank: a chinese corpus annotated with discourse relations, *Language Resources and Evaluation*, 49: 397-431.

10. Language Resource References

Mohamed Maamouri, Ann Bies, Seth Kulick, Fatma Gaddeche, Wigdan Mekki, Sondos Krouna, Basma Bouziri, Wajdi Zaghouani (2010). The Penn ArabicTreeban: Part 1v 4.1. Distributed via LDC. LDC2010T13. ISLRN 512-715-458-848-0.

The ADELE Corpus of Dyadic Social Text Conversations: Dialog Act Annotation with ISO 24617-2

Emer Gilmartin[1], Brendan Spillane[1], Maria O'Reilly[2], Christian Saam[1], Ketong Su[1], Leigh Clark[3], Benjamin R. Cowan[3], Killian Levacher[4], Arturo Calvo[5], Loredana Cerrato[6], Nick Campbell[1], Vincent Wade[1]

ADAPT Centre Trinity College Dublin[1], Phonetics Laboratory, Trinity College Dublin[2], University College Dublin[3], IBM Research Ireland[4], Accenture Norge[5], EIT Digital Sweden[6]

gilmare@tcd.ie

Abstract

Social or interactional dialog is less well described than task-based or instrumental dialog, although there is increasing interest in the genre, particularly in light of new spoken and text dialog applications which aim to relate to the user as well as perform tasks. Dialog act annotation aids understanding of interaction structure; essential to the design of sucessful artificial dialog. Much social text interaction may be closer to social talk than to traditional written language. In this paper we briefly describe social or casual talk, and review how current dialog annotation schemes and particularly the ISO standard 24617-2 (Semantic annotation framework, Part 2: Dialogue Acts) treat non-task elements of dialog. To aid in training a casual talk system, we collected a corpus of 193 dyadic text dialogs, based on a novel 'getting to know you' social dialog elicitation paradigm. We describe the annotation of the dialogs, and propose additional acts to better cover greeting and leavetaking. We report on preliminary analyses of the corpus, and provide an overview of the corpus content and its relationship to spoken language. The corpus, coding manual, and annotations are being packaged and will be made available to interested researchers.

Keywords: Casual Conversation, Dialog Corpus, Dialog Acts

1. Introduction

Many dialogs and indeed parts of dialogs can be described as task-based or instrumental, with clear goals, as in the case of a service encounter or business meeting. Others, such as friendly chats or longer casual conversations, are more interactional in nature. Acknowledging that modelling casual conversation would prove difficult, early dialog technology research focused largely on practical goal-oriented dialog (Allen et al., 2001). Recently, more attention is being paid to social aspects of spoken and text interaction, with the desire for systems which can engage, entertain, and provide the illusion of companionship to users. Successful design of such systems requires relevant data. Dialog act annotation aids understanding of interaction structure; such understanding is essential for designing artificial spoken or text dialog. Below, we briefly review accounts of casual social talk and discuss how current dialog annotation schemes and particularly the 'ISO standard 24617-2 Semantic annotation framework, Part 2: Dialogue acts' (ISO, 2012) (henceforth ISO standard) treat non-task elements of dialog. We then describe the collection and annotation, using the ISO standard, of 193 text dialogs elicited using a novel 'getting to know you' paradigm. We describe the distribution of dialog acts in the corpus as a whole, report on annotation of the dialog acts used in greeting and leave-taking sequences, and describe proposed new social dialog acts for these sequences. We also outline our ongoing work in this area.

2. Instrumental and Interactional Dialog

With live text exchange a part of everyday life we have seen an explosion of casual writing. The structure and content of much of this writing is dialogic and informal, in contrast to traditional unilateral written texts. With the popularity of messaging applications, both SMS and more recent developments such as Facebook Messenger, text conversations are commonplace for social and practical goals. Chatbot implementation on the web and through social media platforms has made conversational commerce and customer care through artificial dialog ubiquitous additions to company websites and social media. Spoken dialog applications such as Alexa or Siri now seek to give the impression of a companionable interlocutor as well as performing practical tasks. We are interested in building dialog systems to create the illusion of a more social or friendly interaction, whether for casual or interactional conversation or to 'lubricate' more transactional exchanges. As a first step we have built a corpus of dyadic text conversations where participants play a 'getting to know you game'. We believe the text of such exchanges is closer to the language found in casual conversational speech than to traditional formal written language, as observed in Fairclough's ideas of 'conversationalization' of text (Fairclough, 1992). Below we briefly review relevant literature on casual conversation.

Casual conversation has been viewed as the most basic use of language (Malinowski, 1936). Such conversation has been theorised to aid in the building of social bonds and in the avoidance of unfriendly or threatening silence, rather than simply serving to providing the medium for the exchange of information or expression of thought (Jakobson, 1960; Brown and Yule, 1983). Schneider (Schneider, 1988) noted casual talk did not seem to conform to Gricean maxims governing the efficient transfer of linguistic information, and proposed a set of maxims peculiar to this genre, based on the importance of avoiding silence and maintaining politeness, suggesting that Grice's Co-operative Prin-

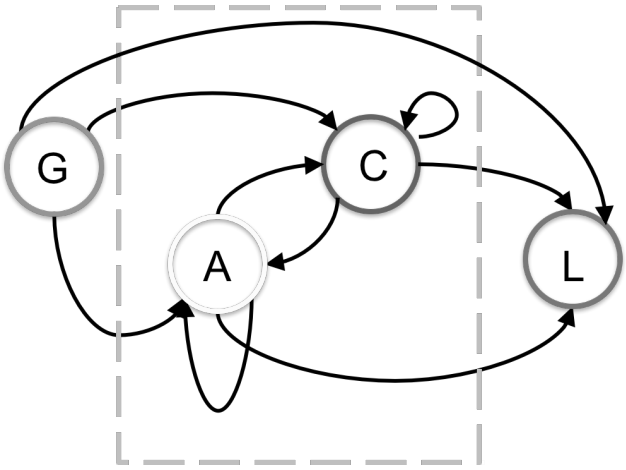


Figure 1: A simplified version of Ventola's conversational phases. The Greeting and Leavetaking phases occur at either end of the interaction, while Approach and Centring stages can recur and alternate throughout the 'body' of the interaction.

ciple itself (Grice, 1975) remained relevant. Casual conversation is postulated to be the medium through which people form and refine their social reality (Eggins and Slade, 2004). Casual conversation can include stretches of small talk, discussion, narrative, and gossip. Laver viewed small talk as performing a transitional function from silence and greetings to the 'meat' of the interaction, and back to closing sequences and leave taking (Laver, 1975). Casual conversation is not monolithic, its structure has been described in terms of distinct phases; ritualised opening greetings, approach segments of light uncontroversial small talk, and, in longer conversations, more informative centre phases (consisting of sequential but overlapping topics), and then ritualised leave-takings (Ventola, 1979). A simplified schematic of the phases of casual conversation can be seen in Figure 1.

Instrumental and interactional exchanges differ in duration; task-based conversations are bounded by task completion and tend to be short, while casual conversation can go on indefinitely. The task-based vs. casual or social distinction is somewhat misleading as casual or social talk has a very important goal - bonding and maintenance of relationships. This goal is implicit and longterm, and thus dialog success is not as easily measured in casual talk as it is in a simple task based interaction such as buying a pizza, where the goal of the transaction is explicit and success can be measured by the accomplishment of this goal. The design of successful companion applications will require the system to engage in casual or social conversation, and knowledge of the structure and dynamics of such interaction will permit the design of better conversational interfaces. Such knowledge will entail clear and efficient coding of the dialog acts present in such talk.

3. Dialog Act Annotation of Interactional Talk

Existing dialog act annotation schemes are largely task-based, perhaps due to the focus on task-based dialog for much of the history of modern dialog systems (Allen et al., 2001). While there have been some schemes based on text conversations (Kim et al., 2010), the vast bulk of schemes have been based on spoken interaction. Several annotation schemes have been developed, often in conjunction with particular corpora or experiments, such as the schemes developed to annotate Trips and Trains, Switchboard, ICSI, and the AMI corpus (Traum, 1999; Core and Allen, 1997; Jurafsky et al., 1997; Shriberg et al., 2004; McCowan et al., 2005). More extensive domain independent schemes such as DIT++ (Bunt, 2006) have also been developed, culminating in the ISO standard.

In existing dialog act annotation schemes, social talk annotation is often restricted to acts such as greeting or apologies. Surveying 14 schemes, Petukova found that 10 schemes included greeting functions, 4 included introductions, 6 had goodbyes, 5 included apology type functions, and 5 contained thanking (Petukhova, 2011). Three systems (AMI, MALTUS, and Primula) provided broader tags to reflect ideas of positivity and negativity, politeness, and positive and negative face work. The ISO standard has tags in the Social Obligations Management (SOM) dimension covering nine social communicative functions, essentially social 'niceties': initialGreeting, initialSelfIntroduction, returnSelfIntroduction, apology, acceptApology, thanking, acceptThanking, initialGoodbye, and returnGoodbye. We are interested in identifying gaps in coverage of social talk in the ISO standard, and in broadening coverage to include social and casual as well as task-based dialog. Below we describe the collection and annotation of the ADELE Corpus of social text dialogs.

4. ADELE Corpus - Collection

A corpus of 193 two-person text dialogs was collected and annotated with the ISO standard to provide initial training data for the ADELE project, a personalized intelligent companion capable of engaged, yet natural and informed, conversational social dialog. Our first objective is to model the early stages or 'onboarding' phase when the machine and user introduce themselves and the system collects useful information about the user through a friendly conversation, corresponding to Ventola's Greeting and Approach stages. Below we briefly describe the scenario and participants, and the interaction platform.

4.1. Scenario

The scenario used was designed to elicit dyadic social dialog. The dialogs were text-based, between English speaking adults connecting remotely via a web-based interface. Each participant was given a persona with information on home, relationships, nationality, job, hobbies and interests. The objective was to discover this information about the interlocutor and also to discover any facts or interests in common. Participants were instructed to be friendly and chatty. In order to promote friendly chat rather than 'interviewing' behaviour, one point was given for each utterance, one for

each trait discovered about the opposing interlocutor discovered, and five points were given when commonalities were discovered.

4.2. Interaction Platform Design

The data were collected using a Dialog Interface, a Google Chrome extension for the team collaboration tool Slack[1], developed in HTML5, CSS3 and JQuery. Interactions were scheduled using a Matching Engine, a RESTful API developed in Java JAX-RS and Jersey that creates new conversations between pairs of available participants and assigns each one a randomly-generated persona. All data were collected in the dialog database (PostgreSQL). Through the Dialog Interface, participants could converse with their partner in the experiment, record the persona traits of their partner, view their own traits and mark which were common.

4.3. Data Collection

The conversations were collected over two months in late 2016. There were 37 participants (26M/11F, age 18-43), all either native English speakers or at least IELTS International English Language Testing Service level 6.5. All participants were postgraduate students or employees of Trinity College Dublin, recruited by advertising within the College. During the experiment, the participants interacted over the Dialog Interface to discover attributes of their partner's persona. When all the attributes of a persona were discovered, participants marked the conversation as finished. If they wished, they could continue their participation and be re-assigned a new fictitious persona and another anonymous participant to start a new conversation with. Thus, a participant could take part in more than one conversation over the course of the data collection, but not with a previous partner.

5. ADELE Corpus - Annotation

A total of 193 completed dialogs were collected. The conversations were annotated using a modified version of the ISO standard, based on 'gold standard' examples of dialogs from the Switchboard corpus annotated using the ISO standard, and presented on the Dialog Bank website (Bunt et al., 2016). A pilot annotation of a subset of the corpus was carried out by two annotators to determine the suitability of the ISO standard to these dialogs and whether extensions were necessary.

One purpose for the annotated corpus was to train a spoken dialog system which would be able to play the roleplaying game described above with a human partner. Therefore, lexical tags were added to the information transfer dialog acts whenever relevant information was included in a participant's contribution. These tags reflected the topic being discussed. In order to distinguish between utterances which moved the dialog forward (by informing the interlocutor of one of the pieces of information needed to accomplish the task) and follow up (friendly comments on this information), any inform act which was not a 'first mention' of relevant information was tagged as a comment, with lexical tags used to mark which topic the act referred to. The lexical tags took the form **[topic]** where the value for topic could be any of the persona attributes in the task, and the tags were appended to the dialog act tags for the relevant functional segments, so the annotation **inform[food]** describes **'I love Chinese food'**.

During the course of the pilot annotation, annotators noted that there were recurring dialog components in extended greeting/introductions and leave-taking (henceforth GIL) sequences which could not easily be satisfactorily annotated using the set of dialog act tags in the SOM dimension of the ISO standard. The fragment below illustrates some of these difficulties in an extended greeting/introduction sequence.

1. A: Hi
2. B: Hello, I'm Ann. I'm from Mexico City. Yourself?
3. A: Hi Ann, nice to meet you. I'm John.
4. B: Hey John, nice to meet you too. How are you today?
5. A: Good, good. You? I'm from Paris, living in London now.
6. B: I'm in good form!.

In the fragment there are four instances of **hello**, **hi**, or **hey**. The first two can be accounted for by the ISO but not the latter as there is no 'generic' greet tag, but only initialGreet and returnGreet. The expression **nice to meet you** and response **nice to meet you too** in lines 3 and 4 are clearly formulaic greetings but it is unclear how to annotate them. A similar situation obtains with the **How are you today? – Good, good** and **You?** (ellipsis of **How are you?**) **– I'm in good form!** in lines 4-6. If the first part of these adjacency pairs are annotated as setQuestions and the second parts as informs or answers, these tags could be placed in the SOM dimension. However, these composite treatments are clumsy to implement during annotation and do not reflect the illocutionary force of the expressions as clearly as existing SOM tags (initalGreet, returnGreet) do for the **Hi** and **Hello** in lines 1 and 2.

To make annotation more efficient, additional acts were created in the SOM category to more easily mark such sequences and similarly problematic sequences in leave-taking sequences. For greeting sequences, the new tags were **ntmy** and **repNtmy** to tag utterances such as 'It's nice to meet you', and responses such as 'Likewise' or 'Nice to meet you too' , **hay** and **repHay** sequences like 'How are you?', and responses such as 'Fine.', and **greet** for extra 'Hello' and 'Hi' utterances. For leave-taking, the new tags were **wntmy** and **repWntmy** for 'It was nice to meet you' and 'It was nice to meet you too'. Table 1 shows the new GIL acts common examples of how they occur in the corpus, and counts. The annotation process, development and analysis of these acts are more fully described in (Gilmartin et al., 2017).

The entire corpus of 193 dialogs was then annotated using this expanded scheme. Conversations were annotated using a Microsoft Excel spreadsheet adapted from those on the

[1] https://slack.com/

Act	Common Examples	Functional Area
ntmy	Nice to meet you	greeting
	Good to talk to you	greeting
repNtmy	Nice to meet you too	greeting
	Good to talk to you too	greeting
hay	How are you?	greeting
	How's it going?	greeting
repHay	Fine	greeting
greet	Hello	greeting
	Hi	greeting
wntmy	It was lovely to meet you	leave-taking
	Nice talking to you	leave-taking
repWntmy	It was nice to meet you too	leave-taking
	Likewise	leave-taking

Table 1: Acts introduced for the ADELE annotation and common surface forms

DialogBank website[2], which were designed for annotation using the ISO standard. The DiAML-TabSW version of the Excel template was used. For each utterance, each dialog act and the relevant functional segment was given a unique identifier and noted on a separate line with details of sender, addressee, and relevant rhetorical relations.

Greeting sections were marked as beginning with the first utterance of the conversation, and ending with the last production of a formulaic greeting/introduction or greeting/introduction response. Leave-taking sequences were marked from the first attempt to close the conversation to the final utterance of the conversation. The data contained 9231 turns or 'utterances' where a turn was defined as the text entered before a user pressed the return key to send their contribution. The vast bulk of utterances were tagged with a single label (7811, 84.7%), 1209 (13%) had two tags, 181 (2%) had three tags, while 26 (0.3%) and 3 utterances had four and five tags.

Of 10889 dialog act tags, 2336 or 21.5% were included in GIL sequences. 1329 tags related to greeting and 1007 to leave-taking. GIL sequences sometimes contained other acts unrelated to greeting, introduction, or leave-taking, as in the above example where the dialog acts contained in **I'm from Mexico City. Yourself?** in line 2 are an inform and setQuestion related to the task. The question is answered on line 5 near the end of the greeting/introduction sequence. The number of dialog acts directly involved in GIL sequences was calculated by disregarding such 'interloping' acts.

Greeting/introduction alone accounted for 1034 labels, while leave-taking alone accounted for 786 labels, making a total of 1820 acts of greeting/introduction and leave-taking, or 16.7% of all dialog acts tagged in the corpus. The leave-taking totals include 194 Leave-taking Introductions – utterances which introduce the closure of the dialog.

[2]https://dialogbank.uvt.nl/

These utterances could be included in the Discourse Structuring dimension, in which case the total for GIL drops to 1626 or 15% of all dialog act labels, which is the most conservative estimate of the proportion of GIL tags in the corpus. The total SOM acts in the corpus including SOM categories outside GIL from the ISO standard amounts to 1824 or 17%. In terms of the prevalence of the new greeting tags, in 187 conversations the hay (How are you?) tag appeared 68 times, the ntmy (Nice to meet you) tag appeared 101 times, and the extra greet tag appeared 66 times (each conversation contained two initialGreets). The response tags repHay and repNtmy appeared less frequently, with 49 instances of repHay and 25 of repNtmy. For the leavetaking tags, there were 139 wntmy (It was nice to meet you) tags and 47 repWntmy tags. These figures are summarized in Table 2.

Act	**Common Examples**	**Functional Area**	**Count**
ntmy	Nice to meet you	Greeting	101
repNtmy	Nice to meet you too	Greeting	25
hay	How are you?	Greeting	68
repHay	Fine	Greeting	49
greet	Hello	Greeting	66
wntmy	It was lovely to meet you	Leave-taking	139
repWntmy	It was nice to meet you too	Leave-taking	47

Table 2: Distribution of new GIL acts

Description	**Count**	**%**
Words	50,439	-
Utterances	9,954	-
- 1 dialog act	7,998	80%
- 2 dialog acts	1,524	15%
- 3 dialog acts	336	3%
- 4+ dialog acts	96	2%
Dialog Acts	12461	-
- Informs	6265	50%
- Questions	2136	17%

Table 3: Words, Utterances, and Dialog Acts in the ADELE Corpus

6. ADELE Corpus Description

The 193 dialogs contained a total of 50,438 words over 9954 utterances (as shown in Table 3), where an utterance was defined as the text entered before the user pressed return. The number of words and utterances per conversation both have log-normal frequency distributions, as shown in Figure 2. The mean number of words per conversation was 250.41, and mean utterances per conversation was 47.76 The number of words per utterance, also shown in Figure 2 ranges from 1 to 49.

6.1. Dialog Structure in the ADELE Corpus

In the ADELE Corpus the ratio of statements to questions is almost 3:1, which, in conjunction with the prevalence of single act utterances, demonstrates that the conversations did not follow a simple 'ping-pong' question-answer structure (a more rigid question/answer dialog would have a ratio of statements to questions closer to 1:1).

6.2. Dialog Acts in ADELE Corpus

The majority of utterances (7998) contained a single dialog act, 1524 contained two dialog acts, with 336 containing 3

dialog acts, 74 with 4 dialog acts, 19 containing 5 and 3 containing 6 dialog acts. It should be noted that the ISO allows multiple tags to attach to the same functional segment (part of utterance which can be described by a dialog act) or utterance.

Of the total 12461 dialog acts annotated, the most common were informs (statements) with 6265 tokens. For the purposes of this corpus, first mentions of relevant information were tagged as informs (2704) while other informs (subsequent mentions or comments) were tagged as comments (3561). There were a total of 2136 questions. Set questions (wh-questions) accounted for 1303 acts, while 'return' set questions (e.g.'and yourself?') totalled 445, and propositional questions (yes/no questions) totalled 331. There were 58 check questions and 10 choice questions. It should be noted that these question totals do not include 'how are you?' questions which were separately tagged as **hay** acts, and that greeting and leave-taking 'nice to meet you' expressions were not tagged as informs, but as **ntmy** or **wntmy**.

6.3. Lexical Density of ADELE Conversations

In order to informally check our assumption, that text chat would be close to spoken conversation, we calculated the lexical density of each of the ADELE conversations and compared their mean with the mean lexical density of conversations in the Cardiff Conversation Database of informal dyadic spoken conversations (Aubrey et al., 2013). Lexical density is a measure of the density of information in a sample of language, calculated as the ratio of lexical ('content') words to the total number of words. This measure is used in genre differentiation and spoken language has been found to have significantly lower lexical density than written text (Ure, 1971; Halliday, 1989; Biber et al., 1999). Lexical density can thus be used as an indicator of how casual the language in a sample is, and compared with other samples. Figure 3 shows the distribution of lexical density values per conversation in the ADELE corpus. The mean value is 0.48. For comparison purposes we calculated the mean lexical density for the conversations in the Cardiff Conversational Database (CCDb), a collection of short dyadic informal spoken conversations. The CCDb mean lexical density was 0.46. All lexical density statistics were obtained using the Web-based Lexical Complexity Analyser (Ai and Lu, 2010).

7. Conclusion

We have described the collection, annotation, and preliminary analyses of the ADELE corpus, a new collection of casual or social dyadic text interactions. The dialog act annotation has resulted in the creation of new dialog acts, expanding the coverage of greeting, introduction, and leavetaking sequences. We found the structure of the conversations to differ from the series of 'question-answer' sequences prevalent in task-based dialog, with the ADELE conversations having a 3:1 ratio of statements to questions. There is a high proportion of Social Obligation Management (SOM) acts, and particularly Greeting, Introduction, and Leavetaking (GIL) acts, in the ADELE corpus. To provide context, Petukova reports percentages of SOM acts in three task-based corpora (AMI, OVIS, and DIAMOND) as ranging from 0.5 to 7.8% of total dialog acts (Petukhova, 2011). The prevalence in the ADELE corpus (15%) is much higher. The bulk of SOM acts in ADELE are greetings/introductions and leave-taking, which is likely due to the more sociable nature of the interactions in ADELE, and to the nature of the 'getting to know you' scenario. It would be very interesting to see how the GIL acts added to the tags for ADELE were accounted for in other corpora. Using lexical density measures as an indicator of 'conversationableness' of the data, we found that lexical density of the ADELE conversations is close to that of the casual spoken language in the Cardiff Conversational Database, which provides some support for claims that the language of casual text-based interaction is closer to spoken conversation than traditional written text.

The factors mentioned above provide some preliminary evidence that the content of the ADELE corpus is social and casual, and similar to conversational speech. However, the tight central tendency for utterances per conversation is not a feature of casual talk, which tends to be open-ended and thus variable in length. The clear central tendency in the distribution of interaction duration in the ADELE corpus is probably an artefact of the nature of the task – there are a fixed number of topics spoken about and participants would tend to move on after discussing each topic, thus limiting conversation length.

Our dialog act annotation of the ADELE corpus demonstrates the need for more investigation of the dynamics and structure of conversations which are not strictly task-based, and for consideration of how such conversations are described in terms of dialog acts. We began by tackling the 'edges' of the interactions in Greeting and Leavetaking. The 'meat' of such conversations tends to weave in and out of different types of interaction – discussion, narrative, gossip, and highly interactive chat or smalltalk. Interactional success depends on 'keeping the conversation going' for extended periods, in contrast to task-based interaction where efficient accomplishment of short term practical goals drives success. In a service encounter such a buying a pizza, the topic can change as soon as the information requested is attained. However, in casual conversation, such changes may be less abrupt, with topic shading allowing friendly conversational flow to be maintained. Maintaining casual conversation involves local activity by participants to choose and develop topics of interest. We are currently using the ADELE data to explore the dynamics of topic changes in conversations which are not strictly task-based to inform the design of companionable text and spoken dialog systems.

We are also validating the expanded annotation scheme used for ADELE with naïve annotators, further investigating the characteristics of the language and dialog acts in the corpus in comparison with other corpora of spoken and written language, continuing our investigation of dialog structure and dialog acts in the Approach and Centring stages of casual conversation, and using the corpus in the ongoing development of the ADELE system. We plan to release the corpus, annotation manual, and annotations to the research community.

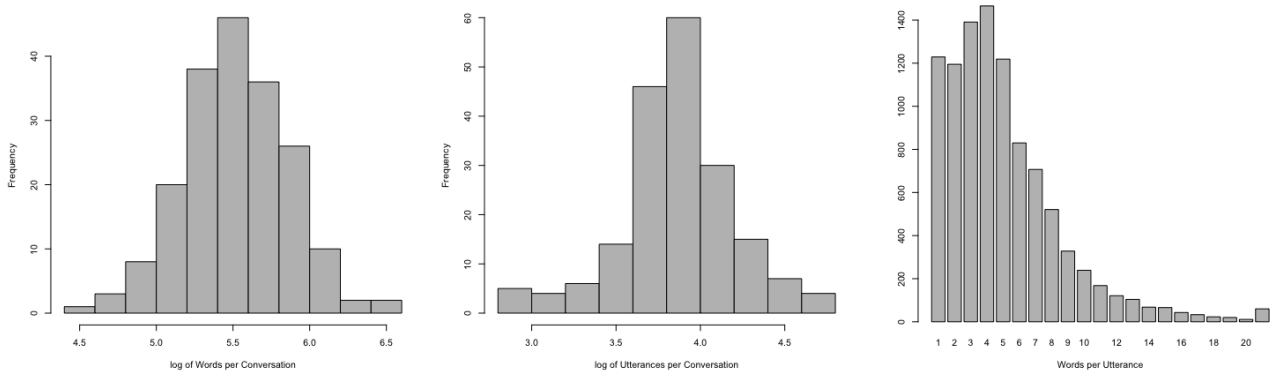


Figure 2: Frequency distributions for log(Words per Conversation), log(Utterances per Conversation, and Words per Utterance, for the ADELE Corpus.

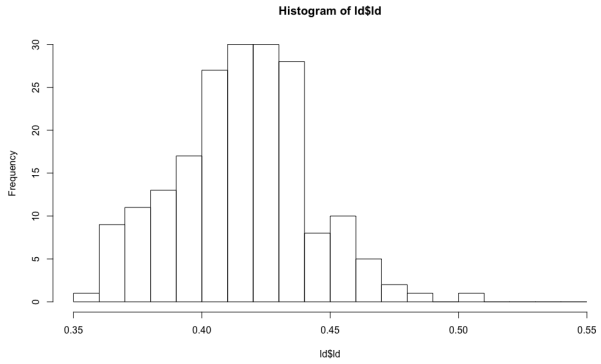


Figure 3: Histogram of Lexical Density per Conversation

8. Acknowledgements

This work was supported by the ADAPT Centre for Digital Content Technology, which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund, and by the European Coordinated Research on Long-term Challenges in Information and Communication Sciences and Technologies ERA-NET (CHISTERA) JOKER project, JOKe and Empathy of a Robot/ECA: Towards social and affective relations with a robot.

9. Bibliographical References

Ai, H. and Lu, X. (2010). A web-based system for automatic measurement of lexical complexity. In *27th Annual Symposium of the Computer-Assisted Language Instruction Consortium. Amherst, MA*.

Allen, J., Ferguson, G., and Stent, A. (2001). An Architecture for More Realistic Conversational Systems. In *International Conference on Intelligent User Interfaces: Proceedings of the 6th International Conference on Intelligent User Interfaces*, volume 14, pages 1–8.

Aubrey, A. J., Marshall, D., Rosin, P. L., Vandeventer, J., Cunningham, D. W., and Wallraven, C. (2013). Cardiff conversation database (ccdb): A database of natural dyadic conversations. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 277–282. IEEE.

Biber, D., Johansson, S., Leech, G., Conrad, S., Finegan, E., and Quirk, R. (1999). *Longman Grammar of Spoken and Written English*, volume 2. Longman London.

Brown, G. and Yule, G. (1983). *Teaching the Spoken Language*, volume 2. Cambridge University Press.

Bunt, H., Petukhova, V., Malchanau, A., Wijnhoven, K., and Fang, A. C. (2016). The dialogbank. In *LREC*.

Bunt, H. (2006). Dimensions in Dialogue Act Annotation. In *Proc. of LREC*, volume 6, pages 919–924.

Core, M. G. and Allen, J. (1997). Coding Dialogs with the DAMSL Annotation Scheme. In *AAAI Fall Symposium on Communicative Action in Humans and Machines*, pages 28–35. Boston, MA.

Eggins, S. and Slade, D. (2004). *Analysing Casual Conversation*. Equinox Publishing Ltd.

Fairclough, N. (1992). *Discourse and social change*. Polity, Cambridge.

Gilmartin, E., Spillane, B., Saam, C., O'Reilly, M., Su, K., Cowan, B. R., Levacher, K., Calvo Dehesa, A., Cerrato, L., Campbell, N., and Wade, V. (2017). Annotation of Greeting, Introduction, and Leavetaking in Text Dialogues Using ISO 24617-2. In *Proceedings of the Thirteenth Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, Montpellier, France.

Grice, H. P. (1975). Logic and conversation. In John P. Kimball, et al., editors, *Syntax and semantics. Vol.3, Speech acts*. Academic Press, New York [etc.] ; London.

Halliday, M. A. (1989). Spoken and written language.

ISO. (2012). *ISO 24617-2:2012 - Language resource management – Semantic annotation framework (SemAF) – Part 2: Dialogue acts*. International Organization for Standardization, Geneva, Switzerland.

Jakobson, R. (1960). Linguistics and poetics. In Th. A. Sebeok, editor, *Style in language*, pages 350–377. MA: MIT Press, Cambridge.

Jurafsky, D., Van Ess-dykema, C., et al. (1997). Switchboard Discourse Language Modeling Project (Final Report).

Kim, S. N., Cavedon, L., and Baldwin, T. (2010). Classifying Dialogue Acts in One-on-One Live Chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 862–871. Association for Computational Linguistics.

Laver, J. (1975). Communicative Functions of Phatic Communion. In Adam Kendon, et al., editors, *Organization of behavior in face-to-face interaction*, pages 215–238. Mouton, Oxford, England.

Malinowski, B. (1936). The Problem of Meaning in Primitive Languages. In *The meaning of meaning: a study of the influence of language upon thought and of the science of symbolism*, pages 296–336. Kegan Paul, Trench, Trübner, London, 4th ed. rev edition.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., and Karaiskos, V. (2005). The AMI Meeting Corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88.

Petukhova, V. (2011). *Multidimensional Dialogue Modelling*. Ph.D. thesis, Tilburg University, Tilburg, Netherlands.

Schneider, K. P. (1988). *Small Talk: Analysing Phatic Discourse*, volume 1. Hitzeroth Marburg.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The ICSI Meeting Recorder Dialog Act (MRDA) Corpus. Technical report, International Computer Science Institute, Berkeley.

Traum, D. (1999). Speech Acts for Dialogue Agents. *Foundations of Rational Agency*, 14:169–202.

Ure, J. (1971). Lexical density and register differentiation. *Applications of linguistics*, pages 443–452.

Ventola, E. (1979). The Structure of Casual Conversation in English. *Journal of Pragmatics*, 3(3):267–298.

An Assessment of Explicit Inter- and Intra-sentential Discourse Connectives in Turkish Discourse Bank

Deniz Zeyrek, Murathan Kurfalı
Informatics Institute, Middle East Technical University, Ankara,
dezeyrek@metu.edu.tr, kurfali@metu.edu.tr

Abstract

The paper offers a quantitative and qualitative analysis of explicit inter- and intra-sentential discourse connectives in Turkish Discourse Bank, or TDB version 1.1, a multi-genre resource of written Turkish manually annotated at the discourse level following the goals and principles of Penn Discourse TreeBank. TDB 1.1 is a 40K-word corpus involving all major discourse relation types (explicit discourse relations at intra- and inter-sentential positions, implicit discourse relations, alternative lexicalizations and entity relations) along with their senses and the text spans they relate. The paper focuses on the addition of a new set of explicit intra-sentential connectives to TDB 1.1, namely converbs (a subset of subordinators), which are suffixal connectives mostly corresponding to subordinating conjunctions in European languages. An evaluation of the converb sense annotations is provided. Then, with corpus statistics, explicit intra- and inter-sentential connectives are compared in terms of their frequency of occurrence and with respect to the senses they convey. The results suggest that the subordinators tend to select certain senses not selected by explicit inter-sentential discourse connectives in the data. Overall, our findings offer a promising direction for future NLP tasks in Turkish.

Keywords: Turkish, discourse relations, intra-/inter-sentential connective

1. Introduction

Discourse parsing is a challenging task for NLP. It involves various subtasks, such as discourse connective detection, argument detection and sense prediction. Since the release of discourse treebanks, particularly PDTB (Prasad et al., 2014), work on discourse parsing has gained an impetus. But most of this work is on English or European languages; work on other languages is scarce. Turkish Discourse Bank, or TDB, a multi-genre corpus of written Turkish, has been created with this motivation. It has the goal of providing researchers with a clearly defined level of discourse structure and semantics and support NLP and LT research in Turkish (Zeyrek and Webber, 2008).[1] Following the rules and principles of PDTB, TDB (version 1.0) has annotated 8483 relations made salient by explicit discourse connectives, involving the discourse connective itself as well as the two text spans it relates (Demirşahin and Zeyrek, 2017).

Discourse relations not only hold between adjacent sentences (inter-sententially) but can also exist between clauses within a single sentence (intra-sententially). While TDB 1.0 is a richly annotated resource of explicit discourse connectives, it does not annotate all types of intra-sentential discourse connectives and leaves out senses as well as other major discourse relation types that PDTB annotates. Given the need to create a more complete version of TDB and considering budgetary constraints, we have decided to enhance a modest portion of TDB 1.0 with new annotations. Thus, TDB 1.1, a 40K-word subcorpus has been created as described in Zeyrek and Kurfalı (2017). The subcorpus includes a collection of 20 text files (each with 2.000 words) distributed according to the genres covered by TDB 1.0 with the following frequencies: fiction (novel and short story) (35%), news (30%), research monograph (2%), magazine article (2%), memoir (2%), interview (1%).[2]

Previous research has shown that parsing decisions conditioned on whether the relation is intra- or inter-sentential yields more effective parsing than decisions based on a single model approach (Stepanov and Riccardi, 2013). In this paper, our focus is what we can gain by considering the distribution of intra- and inter-sentential discourse relations across the corpus and by considering their senses. To this end, we have added converbs, a group of explicit intra-sentential discourse connectives, which were missing in TDB 1.1. Briefly, converbs are suffixal connectives and are a subset of subordinators, which also include postpositions. We first describe this enhancement on the corpus. Then, we assess the distribution of explicit inter- and intra-sentential discourse connectives and evaluate the converb annotations with respect to their senses. The corpus statistics suggest that (a) explicit intra-sentential connectives quantitatively overweigh explicit inter-sentential connectives, (b) explicit intra-sentential discourse connectives belonging to the subordinator type tend to select certain senses not selected by explicit inter-sentential connectives. We suggest that these findings are promising for future discourse parsing efforts of Turkish.

The rest of the paper is organized as follows: §2. starts with a brief overview of the TDB project and the underlying approach to discourse is provided. It continues with §2.1., a section on the major annotation categories and a description of intra- and inter-sentential connections. In §2.2., the addition of converbs is explained and an evaluation with respect to their sense tags is presented. §3. introduces the results of a quantitative analysis of explicit inter-sentential and explicit intra-sentential discourse connectives in TDB 1.1, providing frequency lists and examples from the cor-

[1] In creating TDB, we are concerned with the local level of discourse, a term we use for low-level relations such as discourse relations (Hobbs, 1985). Thus, as in PDTB, we do not commit ourselves to a description of discourse at the global level.

[2] Information regarding TDB can be accessed at: http://www.textlink.ii.metu.edu.tr/turkish-discourse-bank.

pus. §4. includes an analysis of intra- vs. inter-sentential connections based on senses and discusses the possible implications of the current analysis on Turkish discourse parsing research. Finally, §5. summarizes and concludes the study.

2. Goals and Principles of TDB

In this section, the major annotation categories of TDB are explained as the basis of our work in the current paper. The annotation scheme reflects our approach to discourse, which we adopt from years of research in discourse as well as the PDTB principles.

Our annotation scheme aims to capture discourse relations such as contrast, expansion, contingency, etc. that hold between two text spans. As in the PDTB framework, we take discourse relations as a lexically-grounded phenomenon, where each discourse relation is anchored to a discourse connective. The annotations are created by a tool specifically designed for the TDB project. The TDB tool is a Java-based infrastructure with output representations in XML. It uses standoff annotation methodology, where the beginning and end offsets of the annotated content are kept (Aktaş et al., 2010).

We refer to the lexical anchors of discourse relations as discourse connectives (DCs), which are often (but not limited to) syntactic classes, e.g. conjunctions (*ve* 'and ', *ya da* 'or', *çünkü* 'because') and adverbs (*ne var ki* 'nevertheless', *ayrıca* 'in addition'). DCs always relate text spans with an abstract object interpretation, i.e. eventualities, facts, propositions (Asher, 2012), which are referred to as the arguments of a discourse connective and tagged as Arg1 and Arg2. These tags do not indicate any kind of ordering in texts; rather, Arg2 is the text span that is syntactically related to the DC; Arg1 is the other argument. In Turkish, conjunctions and adverbials typically have the Arg1-Arg2 argument ordering.

An important problem to solve in the PDTB-style discourse annotation is to tease apart the discourse and non-discourse role of connectives. For example, in *apples and pears*, *and* is not a discourse connective since the text spans it relates do not have abstract object interpretations. In TDB, only the discourse use of connectives are annotated, eliminating the non-discourse uses by eye and leaving them unannotated.

In Turkish, discourse connectives are not only represented lexically but also morphologically. Morphology is important particularly for intra-sentential connections. Thus, we distinguish a set of intra-sentential DCs, referred to as subordinators subsuming postpositions (*için*, 'in order to, for the purpose of', *gibi* 'as, like') and converbs (e.g. *-sa* 'if', *-ken* 'meanwhile'). Converbs are a unique aspect of Turkish and other Turkic languages reflecting the role of morphology in clause combining. They usually correspond to subordinate conjunctions typical of European languages (Johanson, 1995). The clause combining role of converbs and postpositions is subordination and they have the same ordering of arguments, i.e. Arg2-Arg1.[3]

[3]Thus, postpositions are named as complex subordinators, converbs are referred to as simplex subordinators (Zeyrek and Webber, 2008).

Examples (1) and (2) illustrate two converbs -sA 'if' and -CA 'as'.[4] Here and in the rest of the paper, we show discourse connectives by underlining. Arg1 is rendered in italic fonts, Arg2 in bold fonts. Each connective is annotated with senses from the PDTB 2.0 sense hiearchy, presented in Fig.1. Where relevant, the sense tag of the relation is presented in square parentheses.

(1) **Izini bulursanız**, *bu numaraya haber verin.*
If **you find his trail**, *call this number.* [Contingency:Condition]

(2) **İndik**çe, *mahzende beliren rafları... gördü.*
As **he descended**, *he saw the shelves ... in the cellar.* [Temporal:Synchronous]

2.1. Relation Types Annotated in TDB 1.1

TDB 1.1 annotates four types of discourse relations: (i) relations with an overt lexical or morphological signal (conjunctions, adverbials, subordinators), (ii) implicit relations, where the relations lack an overt signal, (iii) alternative lexicalizations and (iv) entity relations.[5] The annotations are created by determining and selecting the discourse connective span together with its binary arguments and senses.

Implicit relations: Where a discourse relation is not made explicit by a connective, readers can still infer a relation; these have been referred to as implicit discourse relations. In PDTB, when an implicit relation is inferred between text spans, the annotator is asked to insert an explicit connective that makes the relation salient. A sample annotation following the PDTB guidelines is provided in (3).

(3) *Nihayet o da satıldı.* (Implicit=sonuçta) **Mülk olarak elde Süleymaniye'deki konak ile Küçük Çamlıca'daki köşkten başka bir şey kalmadı**.
Eventually, it was sold too. (Implicit=as a result) **There was nothing in their hand as property except the villa in Süleymaniye and the manor house in Çamlıca.** [Contingency:Cause:Result]

Alternative lexicalizations: Discourse relations can also be expressed by other means of lexicalizing a relation (AltLex), such as *because of this* (Prasad et al., 2010). In TDB, we annotate such cases as "phrasal expressions" (Zeyrek et

[4]Throughout the paper, all examples are rendered in Turkish orthography. We use capital letters to capture vowel or consonant harmony, which are operative in stems as well as suffixes. For example, the converb 'if' in example (1) has two surface forms due to vowel harmony: -se and -sa. We use H to indicate any high vowel, A to indicate e or a (see Table 5). Similarly, the converb 'as' in example (2) has 4 surface forms due to vowel as well as consonant harmony: -ca, -ce, ça, çe. To capture consonant harmony, i.e. the harmony of a voiced consonant with its voiceless counterpart as in c and ç (the letters for voiced and voiceless affricates, respectively), we use C. To indicate the harmony of d and t, we use D. (see Table 7).

[5]PDTB also annotates no relations. We leave annotation of no relations for further work.

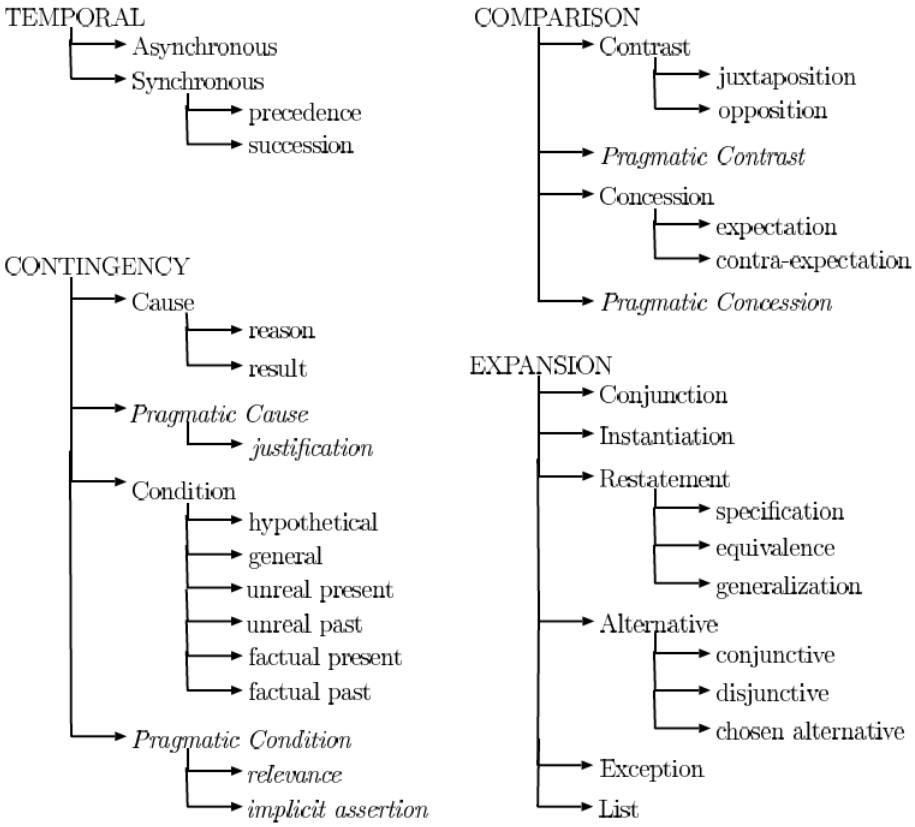


Figure 1: PDTB 2.0 sense hierarchy
(Prasad et al., 2014)

al., 2013). These are devices that are often compositionally formed on the basis of a core element, such as a postposition *rağmen* 'despite' resulting in expressions as *buna rağmen* 'despite this'. A sample annotation is provided in (11) below.[6] Phrasal expressions form a subset of alternative lexicalizations in the PDTB framework (Prasad et al., 2010).

Entity relations: It is possible for a discourse relation to hold between two entities rather than eventualities. Such cases are referred to as Entity Relations (EntRel) and are not assigned a sense tag in PDTB-style annotation, as shown in example (4).

(4) *Aşıklı Höyük bu yerleşimlerden biri.* **Aksaray ilinin Kızılkaya Köyü'nün yakınında, Melendiz Nehri'nin kıyısında yer alıyor.**
The Aşıklı Tumulus is one of these settlements. **It is located by the Melendiz River near the Kızılkaya village of Aksaray province.** [EntRel]

Implicit relations, alternative lexicalizations and entity relations are out of scope of this study but have been overviewed here for the sake of completeness. In the rest of the paper, the term *discourse connective* refers to *explicit* DCs unless otherwise stated.

[6]In TDB 1.0, around 5% of the annotated relations consists of phrasal expressions (Demirşahin and Zeyrek, 2017).

As mentioned above, discourse relations may hold both inter-sententially and intra-sententially. Thus, a DC can be categorized as inter-sentential (inter-S) or intra-sentential (intra-S) according to where its arguments are situated in the text. Intra-S connectives relate two text spans situated in the same sentence. We use orthographic conventions in postprocessing the data in terms of inter- and intra-S connections. Typically, in Turkish, arguments of an intra-S connective are not separated by terminal punctuation marks, e.g. '.', '!', '?' (example 5). Arguments of an inter-S connective are in different sentences (example 6).

(5) *Müthiş soğuk bir gündü* ve **tipi şeklinde kar yağıyordu.**
It was an extremely cold day and **there was a blizzard.** [Expansion:Conjunction]

(6) *Kızınca bir çocuk kadar bile olamazdım.* **Bir tenekeye tekme atamazdım,** mesela.
Even when I was angry, I would not be able to act like a child. **I could not even kick the bin,** for instance. [Expansion:Instantiation]

2.2. Annotation Procedure of Converbs and Evaluation

As with other connectives, converbs have a DC and a non-DC use and to distinguish these cases is important. The abstract object criterion is helpful, showing us when to take a converb as a DC. For example, sentence (7)

can only be interpreted as having a single abstract object interpretation, blocking the interpretation of an *entering event* separate from the *running event*; thus in this case, the converb -ArAk 'by (means of)' fulfills a non-DC role. As opposed to this, in example (8), the converb has a DC role since it relates two abstract objects (a *questioning event* and a *referring event*) expressed in the arguments.

(7) Ayşe eve koşarak girdi.
Ayşe walked into the house (by) running.

(8) **.. bir bakıma kendine de gönderme yap**arak, *yazılış mantığını sorgular.*
.. he questions the wording (by) **referring to himself in a way.**

In Turkish, when a converb has a DC role, it is always an explicit intra-S connective. Thus, with the addition of converbs, we extend the coverage of explicit intra-S DCs in TDB 1.1. This allows us to provide an assessment of explicit intra-S and inter-S connectives, which is the focus of our study.

Converb annotations are realized in two phases. First, connective-argument annotations are created by two independent annotators who go over the whole corpus determining the DC use, eliminating the NDC use of converbs. The resulting annotations are discussed in meetings, where agreed versions are produced by a unanimous decision and recorded. In the second phase, the converb-argument annotations are checked for their correctness by two independent annotators and sense tags are assigned; then the cycle is repeated. The inter-annotator agreement for the senses at all levels of the sense-hierarchy is strong, i.e. $\geqslant 0.8$ (see Table 1).[7]

Sense	IAA
Level-1	89.5%
Level-2	81.9%
Level-3	80.0%

Table 1: IAA results of converb Level-1 senses in TDB 1.1

Overall, 12 unique tokens of converbs realized by 38 different surface forms amounting to 105 converb tokens are added to the data. Hence, the number of explicit connectives increased in the data. The current coverage of TDB 1.1 is shown in Table 2 (Zeyrek and Kurfalı, 2017).

3. Distribution of Intra- and Inter-sentential Connectives in the Corpus

A quantitative analysis of explicit intra-S DCs (including converbs) and inter-S DCs shows that explicit intra-S DCs are more frequent than explicit inter-S DCs, as shown in Table 3.

[7]The converb annotations were created by Işın Demirşahin, Ahmet Faruk Acar, Arzu Burcu Güven and Nihan Soycan, postgraduate students at Cognitive Science Department, Middle East Technical University. The IAA is measured by the exact match method (Miltsakaki et al., 2004).

TYPE	Frequency
Explicit	**868**
Implicit	407
EntRel	541
AltLex	108

Table 2: Distribution of discourse relation types in TDB 1.1 including the converbs added to Explicit relations

TYPE	Frequency	Ratio
Inter-S Explicit DR	688	79.2%
Intra-S Explicit DR	180	20.8%

Table 3: Distribution of inter-S and intra-S explicit discourse relations in TDB 1.1

Analysis of inter-sentential connectives: We find that among explicit inter-S DCs, *ama* ('but, yet') is the most frequent DC as well as the most ambiguous one with 7 different sense tags. Table 4 presents the 10 most frequent explicit inter-S connectives together with their canonical syntactic types, which constitute 82.7% of all the explicit inter-S tokens in the corpus.

DC	Gloss	Syntactic type	Frequency
ama	*but*	conjunction	48
çünkü	*because*	conjunction	21
ayrıca	*in addition*	adverb	17
sonra	*then*	adverb	13
ancak	*however/yet*	adverb	12
oysa	*however*	adverb	12
fakat	*but*	conjunction	7
aslında	*in fact*	adverb	7
ve	*and*	conjunction	7
önce	*before*	adverb	5

Table 4: 10 most frequent inter-S DCs and their canonical syntactic types

Example (9) presents one of the inter-S uses of *ama*.

(9) *Birden geldiğini duydum.* Ama, **göremedim onu.**
Suddenly, I heard her come. Yet, **I could not see her.** [*Comparison: Concession: contra-expectation*]

Analysis of intra-sentential connectives: Of all the explicit intra-DCs in the corpus, *ve* 'and' is the most frequent connective comprising 28.3% of the explicit intra-S DCs and 23.2% of the explicit DCs in TDB 1.1 (Table 5). Two converbs, namely -HnCA 'when' and -ken 'meanwhile' are among the most frequent explicit intra-S DCs in the corpus.

Example (10) below shows the purpose sense of the postposition *için*, the second most frequent explicit intra-S DC in the corpus.

(10) **Onu görmek** için *tüm zamanınızı parkta geçirmeye başlarsınız.*

DC	Gloss	Syntactic type	Frequency
ve	*and*	conjunction	195
için	*since/in order to*	postposition	91
ama	*but*	conjunction	71
sonra	*after*	postposition	58
-HncA	*when*	converb	22
gibi	*as/like*	postposition	18
çünkü	*because*	conjunction	17
-ken	*meanwhile*	converb	16
ancak	*however/yet*	adverb	15
kadar	*to the degree that*	postposition	13

Table 5: 10 most frequent explicit intra-S DCs and their canonical syntactic types

You would start to spend all your time in the park in order to **see her.** [*Contingency:Purpose*]

4. An Analysis Based on Senses

Table 6 shows the distribution of explicit intra- and inter-S DCs across Level-1 senses.[8]

	Expl. Inter-S DCs		Expl. Intra-S DCs	
Sense	Count	Ratio	Count	Ratio
Contingency	35	0.19	167	0.24
Temporal	25	0.14	149	0.22
Comparison	**82**	**0.45**	126	0.18
Expansion	36	0.22	**246**	**0.36**

Table 6: Distribution of Level-1 senses among explicit inter- and explicit intra-S discourse relations

The numbers in bold fonts indicate the most frequently occurring instances. Table 6 also shows that while Comparison is the most frequent Level-1 sense selected for explicit inter-S connectives, Expansion is the most common Level-1 sense chosen for explicit intra-S connectives.

4.1. Observations on the Sense of Intra-sentential Discourse connectives

In this section we offer two observations. First, we find that there are several second- or third-level senses which seem to be preferred by subordinators (both postpositions and converbs) as shown in Table 7.[9]

We note that most of the senses in Table 7 can well be conveyed by means of an AltLex containing the core lexeme of the intra-S connective, as in the use of the postposition *rağmen* 'despite' in the phrasal expression *buna rağmen* 'despite this' (see example (11)). We conjecture that this implies the ubiquity of the senses selected by postpositions or phrasal expressions derived from them.

Sense	Freq.	DC/Gloss	Syntactic type of DC
Contingency: Purpose	79	için (*since/in order to*)	postposition
Temporal: Synchronous	57	-DHğHndA (*when*)	converb
Expansion: Manner	28	kadar (*as*)	postposition
Comparison: Concession: expectation	22	rağmen (*despite*)	postposition
Comparison: Degree	11	kadar (*to the degree that*)	postposition

Table 7: The most frequent second- or third-level senses selected only by converbs or postpositions in their discourse connective roles

(11) *..pistin görülmediği anlaşılmış,* buna rağmen **radar yardmı istenmemiştir**.
..it was understood that the airfield was not seen; despite this, **radar help was not requested.** [*Comparison:Concession:contra-expectation*]

Secondly, Table 8 shows that three explicit temporal DCs exhibit a tendency towards Asynchronous:precedence or Asynchronous:succession depending on whether they establish inter- or intra-level connection. By definition, in precedence relations, Arg1 precedes Arg2; in succession relations, Arg2 precedes Arg1. The inter-S *sonra* 'later' (syntactically an adverb) exhibits Arg1-Arg2 ordering and hence prefers the Temporal:Asynchronous:precedence sense. On the other hand, the intra-S *sonra* 'after' (syntactically a postposition) displays Arg2-Arg1 ordering and tends to convey the Temporal:Asynchronous:succession sense. Examples (12) and (13) exemplify argument ordering and the sense of an adverb and a postposition, respectively.

(12) *Durun, oturun biraz. Anlatın.* **Gidersiniz** sonra. *Wait, have a sit for a while. Tell us.* **You can leave** later. *[Temporal:Asynchronous:precedence*]

[adverb]

(13) **İyice kendine geldikten** sonra *getirilen kuru elbiseleri giydi.*
After **regaining his consciousness,** *he wore the dry clothes they brought.* [*Temporal:Asynchronous:succession*]

[postposition]

4.2. Implications

The differences between inter- and intra-sentential relations in discourse have long been recognized and employed

[8] TDB allows assigning multiple senses to a discourse relation. Therefore, in Table 6, overall, there are more senses than the number of discourse relations.

[9] *Manner* and *Degree* have been introduced on the basis of Turkish data (Zeyrek and Kurfalı, 2017). *Purpose* has been borrowed from PDTB 3.0 sense hierarchy (Webber et al., 2016).

DC/Syntactic type	Sense (Temporal:Asynchronous)
ardından (inter-S)/Adv	precedence (2/2)
ardından (intra-S)/P	succession(2/2)
sonra (inter-S)/Adv	precedence (13/13)
sonra (intra-S)/P	succession (58/58)
önce (inter-S)/Adv	succession (5/5)
önce (intra-S)/P	precedence (8/8)

Table 8: Precedence or succession senses selected by 3 explicit Temporal:Asynchronous connectives. The numbers in parentheses refer to the number of times the subsense is selected and the number of all temporal relations selected for the connective, respectively. Adv stands for adverb, P for postposition.

with success in computational analysis of discourse involving automatic argument extraction and sense labeling (Sporleder and Lascarides, 2008; Joty et al., 2013; Liu and Lapata, 2017; Braud and Denis, 2014). In the current study, for the first time, we offered corpus statistics from Turkish regarding the distribution of inter- and intra-sentential relations with respect to their senses to set the basis for future computational analysis and discourse parsing studies. Our findings suggest that whether or not a given explicitly marked relation holds inter- or intra-sententially may carry valuable information for Turkish as well. Therefore, the reported results are promising for future efforts on automatic sense disambiguation of explicit discourse connectives in Turkish.

5. Summary and Conclusion

This paper started with new enrichments on TDB 1.1 involving a new type of explicit intra-S DCs - converbs, added to the corpus as a further type of subordinator discourse connectives. With the addition of converbs, two goals have been accomplished: (a) a unique aspect of Turkish morphology that has a bearing on clause combining and hence intra-sentential discourse relations has been captured in the corpus, and (b) the overall frequency of intra-sentential explicit discourse connectives increased. This enhancement allowed us to make a comparison between explicit intra- and inter-sentential discourse connectives in TDB 1.1. With various corpus statistics, the paper showed that explicit intra-S DCs occur more frequently than explicit inter-S DCs and it presented evidence for the senses preferred by explicit intra-sentential connectives, namely subordinators but not by explicit inter-S discourse connectives. The paper thus argued that the analysis of explicit intra- and inter-S DCs with respect to the senses they choose is a promising direction for further discourse parsing studies on Turkish. Our aim for the future is to implement the ideas that arise from the current work on various NLP tasks.

6. Acknowledgements

This research was partially supported by METU Project Funds (project ID: BAP-07-04-2017-001). We thank Dilek Deniz Bilgiç for recording the converb annotations and Tunç Güven Kaya for his help in postprocessing the data. We are grateful to our three anonymous reviewers. All remaining errors are ours.

7. References

Aktaş, B., Bozsahin, C., and Zeyrek, D. (2010). Discourse relation configurations in Turkish and an annotation environment. In *Proceedings of the fourth linguistic annotation workshop*, pages 202–206. Association for Computational Linguistics.

Asher, N. (2012). *Reference to abstract objects in discourse*, volume 50. Springer Science & Business Media.

Braud, C. and Denis, P. (2014). Combining natural and artificial examples to improve implicit discourse relation identification. In *COLING*.

Demirşahin, I. and Zeyrek, D. (2017). Pair annotation as a novel annotation procedure: The case of Turkish Discourse Bank. In *Handbook of Linguistic Annotation*, pages 1219–1240. Springer.

Hobbs, J. R. (1985). On the coherence and structure of discourse. *Technical report CSLI-85-37, Center for the Study of Language and Information, Stanford University.*

Johanson, L. (1995). On Turkic converb clauses. *Haspelmath & König (eds.)*, 1995:313–48.

Joty, S. R., Carenini, G., Ng, R. T., and Mehdad, Y. (2013). Combining intra-and multi-sentential rhetorical parsing for document-level discourse analysis. In *ACL (1)*, pages 486–496.

Liu, Y. and Lapata, M. (2017). Learning contextually informed representations for linear-time discourse parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1300–1309.

Miltsakaki, E., Prasad, R., Joshi, A. K., and Webber, B. L. (2004). The Penn Discourse Treebank. In *LREC*.

Prasad, R., Joshi, A., and Webber, B. (2010). Realization of discourse relations by other means: alternative lexicalizations. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1023–1031. Association for Computational Linguistics.

Prasad, R., Webber, B., and Joshi, A. (2014). Reflections on the Penn Discourse Treebank, comparable corpora, and complementary annotation. *Computational Linguistics*, 40(4):921–950.

Sporleder, C. and Lascarides, A. (2008). Using automatically labelled examples to classify rhetorical relations: An assessment. *Natural Language Engineering*, 14(3):369–416.

Stepanov, E. A. and Riccardi, G. (2013). Comparative evaluation of argument extraction algorithms in discourse relation parsing. In *13th International Conference on Parsing Technologies (IWPT 2013)*, volume 36, page 44.

Webber, B., Prasad, R., Lee, A., and Joshi, A. (2016). A discourse-annotated corpus of conjoined VPs. *LAW X*, page 22.

Zeyrek, D. and Kurfalı, M. (2017). TDB 1.1: Extensions on Turkish Discourse Bank. *LAW XI 2017*, pages 76–81.

Zeyrek, D. and Webber, B. L. (2008). A discourse resource for Turkish: Annotating discourse connectives in the METU Corpus. In *IJCNLP*, pages 65–72.

Zeyrek, D., Demirsahin, I., Sevdik-Callı, A., and Çakıcı, R. (2013). Turkish Discourse Bank: Porting a discourse annotation style to a morphologically rich language. *Dialogue & Discourse*, 4(2):174–184.

Compilation of Corpora for the Study of the Information Structure–Prosody Interface

Mónica Domínguez[1], Alicia Burga[1], Mireia Farrús[1], Leo Wanner[2,1]
University Pompeu Fabra[1], Catalan Institute for Research and Advanced Studies (ICREA)[2]
C. Roc Boronat, 138
Barcelona, Spain
{monica.dominguez, alicia.burga, mireia.farrus, leo.wanner}@upf.edu

Abstract

Theoretical studies on the Information Structure–prosody interface argue that the content packaged in terms of theme and rheme correlates with the intonation of the corresponding sentence. However, there are few empirical studies that support this argument and even fewer resources that promote reproducibility and scalability of experiments. In this paper, we introduce a methodology for the compilation of annotated corpora to study the correspondence between Information Structure and prosody. The application of this methodology is exemplified on a corpus of read speech in English annotated with hierarchical thematicity and automatically extracted prosodic parameters.

Keywords: prosody, information structure, theme, rheme, thematicity, language resource, annotated corpora

1. Introduction

The interest in the Information Structure–prosody correspondence applied to natural language speech generation lies in the derivation of prosody that is communicatively oriented and, therefore, more expressive. Knowing the linguistic mechanisms involved in human communication is pertinent to the achievement of multifaceted speech technologies that can carry out more complex tasks linked to conversational settings. The so-called Information Structure–prosody interface stands out as a solid ground for starting to build up such a communicative model in the computational field. However, empirical approaches to the Information Structure–prosody interface are scarce, studies on a corpus of more than two speakers are uncommon and the availability of corpora is, so to say, exceptional.

It is usually one aspect of Information Structure that is studied: thematicity. Thematicity defines how content is packaged in terms of "what is being talked about", i.e., the 'theme' and "what is being said", i.e., the 'rheme'. Most of the approaches draw upon a binary flat thematic division and established a one-to-one correspondence between theme–rheme and rising–falling intonation patterns respectively; see, e.g., (Steedman, 2000; Haji-Abdolhosseini and Müller, 2003; Büring, 2003).

A different view on thematicity is that advocated by I. Mel'čuk in the context of the MTT (Mel'čuk, 2001). Compared to the traditional theme–rheme dichotomy, thematicity in the MTT introduces two key features that enhance the scope of the theme–rheme span division, namely: (i) the notion of *specifier*, which sets up the context of the sentence, and (ii) the fact that thematicity is defined over propositions, rather than over sentences. This second feature implies that thematicity is *per se* hierarchical: if a proposition is embedded, its thematicity will be embedded as well. Previous studies proved that hierarchical thematicity corresponds to a wider range of intonation patterns, and is, therefore, a more adequate representation than binary approaches, especially for long syntactically complex sentences; see, e.g., (Domínguez et al., 2016a).

In this paper we present a methodology for the compilation of a corpus for research on the Information Structure–prosody interface from an empirical perspective. This methodology is based on the formal description of information (or communicative) structure by Mel'čuk (2001), which has been already used for the annotation of hierarchical thematicity of written text in (Bohnet et al., 2013); and an automatic annotation of prosody based on a modular pipeline for extraction of acoustic parameters (Domínguez et al., 2016c). An example application is introduced and demonstrated in the online platform *Praat on the Web* (Domínguez et al., 2016b). Classification experiments on a corpus of read speech in English are carried out to validate our approach.

The rest of the paper is structured as follows. The next section presents the motivation and background of this work. The methodology proposed for the compilation of a corpus to study the Information Structure correspondence is described in Section 3. A sample application of a small corpus of read speech in American English is introduced in Section 4. Then, the validation of our approach is presented in Section 5. Finally, conclusions are drawn in Section 6.

2. Motivation and Background

The role of Information Structure (IS) in comprehension of read and spoken speech has been reported for a long time in linguistic and cognitive sciences (Clark and Haviland, 1977; Bock et al., 1983; Fowler and Housum, 1987; van Donselaar and Lentz, 1994). Recent studies in German (Meurers et al., 2011) and Catalan (Vanrell et al., 2013),

for example, show that characteristic intonation patterns that make a distinction between theme and rheme spans contribute to a better understanding of the message.

The relationship between Information Structure and intonation had been discussed even before the **To**nes and **B**reaks **I**ndex (ToBI) (Silverman et al., 1992) was agreed upon as a convention to represent intonation cues. (Beckman and Pierrehumbert, 1986) suggest that the characteristic bitonals for theme and rheme are rising (L*+H) and falling (H+L*), respectively. (Steedman, 2000) proposes a question–answer setting for the identification of theme and rheme and builds upon Beckman's assumption to hypothesize on complete intonation patterns for theme (L*+H LH%) and rheme (H* LL%).

Some attempts have been made on exploring additional aspects of prosody, apart from ToBI contours, in connection with Information Structure. These studies are, as a rule, restricted to one prosodic element in isolation; see, e.g. (Calhoun, 2010) on rhythm (or, rather, 'metrical structure', as the author defines it); (Xu, 1999) on F0 alignment and (Féry, 2013) on prominence and phrasing.

With respect to empirical studies, the intonation of thematicity[1] is studied in German using one speaker (Baumann, 2012). (Féry and Kügler, 2008) study the process of tonal scaling on a corpus of German consisting of eighteen speakers and 2,277 sentences of the same syntactic structure with a varying number of constituents, word order and theme–rheme structure.

All of these studies coincide in that: (i) they analyze only one aspect of prosody, mostly intonation (the variation of F0); and (ii) their representation of binary thematicity is not formalized as required in computational linguistics.

3. Methodology

This paper envisages the compilation of corpora to study the Information Structure–prosody interface from a methodological perspective based on the formal representations of hierarchical thematicity as described by Mel'čuk (2001) and the annotation guidelines established in (Bohnet et al., 2013). The proposed methodology aims to facilitate the following goals:

- to compile large amounts of data from different registers and languages;
- to analyze the hierarchical thematicity–prosody correspondence in human speech using corpus-driven approaches;
- to explore a parametric representation of prosodic elements in its relationship to Information Structure.

Such a methodology addresses two main research issues in this field: (i) the lack of empirical analysis of the IS–prosody correspondence; and (ii) testing of theories on the IS–prosody interface in corpus-based computational models. In order to address these two issues we propose a processing pipeline implemented in the online platform based on Praat (Boersma and Weenink, 2017): *Praat on the Web* (Domínguez et al., 2016b).[2] This platform takes as the basic annotation file, a *TextGrid*,[3] as in standard Praat, and allows scripting of subroutines on both audio and text input using a modular approach, which is not possible in standard Praat.

Figure 1 sketches the pipeline for compilation of corpora to study the IS–prosody interface. The following input is required: (i) a speech (*wav*) file containing the prosodic information ('Pros') and; (ii) the corresponding text (*txt*) file annotated with thematicity ('IS') following the guidelines established in (Bohnet et al., 2013). In module 1, the text is converted to TextGrid format adapting the original annotation to a specific organization into tiers based on the description of thematicity proposed by (Mel'čuk, 2001), as will be detailed in Section 3.1.. Module 2 generates using Praat in-built functions two objects that are needed to extract prosodic information from speech: the pitch and intensity objects. Then, the annotation of prosodic and linguistic features is executed from modules 3 to 6 resulting in an annotated TextGrid with prosodic and thematicity features. Finally, the pipeline outputs a comma separated values (*csv*) file that can be fed to the validation stage to be analyzed by a statistics package or used as input for classification algorithms.

3.1. Annotation of Hierarchical Thematicity

The fact that thematicity, in Mel'čuk's view, is defined over propositions rather than sentences implies that thematicity is *per se* hierarchical, allows embeddedness and, thus, involves different levels of thematicity. For instance, a theme (T1) can be embedded in another theme or rheme (R1) span. Figure 2 shows the levels of embeddedness in example (1), where T1(P2), for instance, is a level 2 theme that belongs to a level 2 proposition (P2) that is embedded in the main T1 span. As more than one thematicity span may exist within the same proposition, abbreviations include a number (e.g., 'SP1') that indicates the number of occurrences at each level (e.g., 'SP2' would be the second specifier in a specific thematicity level).

Guidelines for annotation of hierarchical thematicity were defined and tested in (Bohnet et al., 2013) for the study and annotation of communicative structure in written text. In order to deal with spoken material, an adaptation of these guidelines must be carried out mostly in terms of format to fit in the requirements of the TextGrid format.

[1] A number of other studies refer to thematicity with the term 'givenness'; see, e.g., (Schwarzschild, 1999), and thus talk about 'given' and 'new' information (Chafe and Li, 1976; Clark and Haviland, 1977; Brown, 1983).

[2] `http://kristina.taln.upf.edu/praatweb/`

[3] A dedicated format of Praat for annotation of speech that maps a minimum of one tier to the whole time-stamp of the associated sound file.

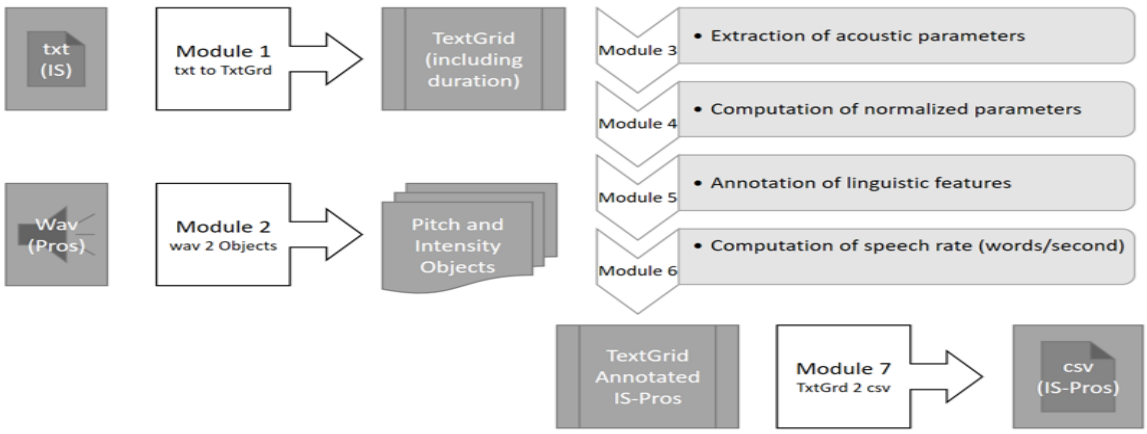


Figure 1: Processing pipeline for IS-Prosody corpus compilation.

(1) [Men {[who]T1(P2) [have played hard all their lives]R1(P2)}P2]T1 [aren't about to change their habits]R1 , [[he]T1(SP1) [says]R1(SP1)]SP1.

Example (1) shows a sentence annotated with thematicity following the guidelines established in (Bohnet et al., 2013). In this annotation, the main proposition (P1) is often not required for annotation, as it is assumed to be marked by the full stop signalling the end of the sentence. However, P1 is required in the TextGrid format, as it is the segment used for further computation of relative prosodic parameters, as will be introduced in the next section. Therefore, the proposed annotation includes two tiers for each level of representation: one for propositions (e.g., L1P) and one for thematicity labels (e.g., L1T).[4] Figure 2 presents the annotation of example (1) using the visualization tool available in Praat on the Web.

Within the processing pipeline, module 5 annotates linguistic features related to the communicative structure of the text, namely, the number of words in each span, the number of embedded spans it contains and the string of labels associated to that interval at the L1P level.

3.1.1. Annotation of Prosody

Automatic extraction and computation of acoustic parameters is carried out using the extension of Praat for feature annotation and the automatic prosody tagger presented in (Domínguez et al., 2016b; Domínguez et al., 2016c).

Table 1 shows the complete list of absolute and relative acoustic parameters (grouped by the three acoustic elements: F0, intensity, and rhythm), and abbreviations (within brackets) used in this paper.

Absolute values are extracted using different predetermined functions available in Praat. Normalized values relative to the whole sample are computed for each segment of analysis, usually a thematicity span (it may be another segment, e.g., a word). Normalized values for mean absolute values of F0, intensity and speech rate are computed using the 'z-score' normalization. Parameters referring to a time point are computed extracting the point of maximum F0 and minimum intensity respectively and calculating the relative time position in the span with a minmax score. Minmax normalization is computed following the equation 1:

$$minmax.t = \frac{x.t - min.t}{max.t - min.t} \quad (1)$$

where:

x.t = point in time where a peak or valley is located within an interval (e.g., word),

min.t = starting point in time of the corresponding interval, and

max.t = ending point in time of the corresponding interval.

Table 1: Prosodic parameters.

Absolute Parameter	Relative Parameter
mean F0 (F0) standard deviation F0 (std.F0) minimum F0 (min.F0)	z-score F0 (z_F0)
maximum F0 (max.F0)	time point of max.F0 (maxF0.t)
mean intensity (int) standard deviation intensity (std.int)	z-score int (z_int)
minimum intensity (min.int)	time point of min.Int (minInt.t)
maximum intensity (max.int)	
duration (dur)	z-score dur (z_dur)
speech rate in words/sec (sr.w)	z-score sr (z_sr)
speech rate in syllables/sec (sr.s)	

[4] The numbers of the levels are correlative indicating the order in the hierarchy: first (1), second (2), third (3), etc.



Figure 2: Example of Hierarchical Thematicity visualization.

In the minmax normalization, the minimum value is the starting time of the interval, which is mapped to 0, and the maximum value is the ending time of the interval, which is mapped to 1. So, the entire range of time points is mapped to the range 0 to 1. This gives us an idea of the relative time location of the peak within a time segment (in this case a word). In other words, the computed minmax score provides information on the location of the F0 peak ('maxF0.t') and intensity valley ('minint.t'). Thus, if an F0 peak is located within the first half of the time span, it will have a score between 0 and 0.5, and if an intensity valley is located within the second half of the span, the score will be between 0.5 and 1.

4. Example of Application

A selection of 109 isolated sentences from the Penn Treebank (Charniak and al., 2000) (section of the Wall Street Journal) was made from the annotated material used in (Bohnet et al., 2013). The corpus contains not only simple sentences, but also coordination, subordination and the combination of both. This varied syntactic composition is related to the representativeness of communicative structure in terms of: number of thematicity levels (up to three in the corpus); position of spans within the sentence and with respect to each other; and continuity or lack of continuity of spans (in particular, rheme spans can be discontinuous). The corpus has an average of fifteen words per sentence with a minimum of three words and a maximum of thirty. This selection of sentences was recorded in a professional studio by twelve native speakers of American English.

There is a balanced six-to-six distribution of male and female speakers. Participants are assigned an anonymous identifier with the format: *speaker (abbreviated as 'spk') – number (a correlative natural number) – gender ('f' for female or 'm' for male)*, resulting in, e.g., 'spk1f'.

Two datasets are created extracting acoustic parameters from different segments (see Table 2). Acoustic data from all twelve speakers is included in the sentence and thematicity span dataset (abbreviated as SSD and TSD, respectively). The main difference between these two datasets is that in SSD the segments are sentences and the classes to be predicted account for the L1 thematicity of each sentence, whereas in TSD the segments are thematicity spans with their corresponding labels assigned to them.

Table 2: Datasets derived from the corpus of read speech.

Acronym	Dataset Name	Speakers	Attributes	Instances	Classes
SSD	Sentence Span Dataset	12	11	1,308	17
TSD	Thematicity Span Dataset	12	14	6,036	31

5. Validation Experiments

Classification experiments are carried out using the Weka 3.8 Workbench (Hall et al., 2009). A bagging classifier with RepTree is using as classifier with a 10-fold cross-validation configuration. We set out to demonstrate the hypothesis that prosodic parameters are related to thematicity labels at the level of two different partitions: the sentence as a whole and each thematicity span. The objective of these experiments is to observe in isolation prosodic parameters and hierarchical thematicity in order to get a closer insight on their relationship. In these experiments, the correspondence of prosody and thematicity is put to test assuming a bidirectional relation between them, but acknowledging that both of them are dependent upon other linguistic phenomena.

5.1. Prediction of Labels within Thematicity Span

The TSD with all thematicity labels is used to perform the prediction of thematicity labels (a total of thirty-one distinct labels) using as attributes acoustic features and number of words in each span. The purpose of the experiment is to observe the correspondence between hierarchical thematicity and acoustic parameters using all speech samples in our corpus. A ZeroR classifier is used as baseline to evaluate and compare the level of improvement

to the bagging classifier. Table 3 shows average precision (P), recall (R) and F-measure (F) results for the bagging classifier (Bag) and baseline (BL).

Table 3: Absolute improvement classification results.

	Precision			Recall			F-Measure		
	BL	Bag	**AbsImp**	BL	Bag	**AbsImp**	BL	Bag	**AbsImp**
TSD	0.05	0.71	**0.66**	0.22	0.71	**0.49**	0.08	0.70	**0.62**
SSD	0.29	0.73	**0.44**	0.54	0.75	**0.21**	0.38	0.74	**0.36**

5.2. Prediction of labels in each sentence

A second experiment is carried out at the sentence level. For each sentence span, acoustic parameters are extracted to predict the thematicity label sequence at L1 using the SSD (a total of seventeen distinct labels are to be predicted). A simple rule classifier (ZeroR), based on a majority vote, is used as baseline. Classification with ZeroR shows a low precision (P=0.29) and F-measure (F=0.38) while recall is 0.54. Then, a bagging classifier is used and results show a considerable increase in all measures with an average absolute improvement over the baseline of P=+0.44 R=+0.19 and F=+0.36. Table 3 reports precision, recall and F-measure results from this classification.

6. Conclusions and Future Work

The contributions of this approach for the compilation of corpora to study the Information Structure–prosody interface are the following: (i) it is automatized to adapt to format requirements; (ii) it assumes a formal representation of thematicity to annotate text instead of the *ad hoc* perspective taken by traditional approaches; and (iii) it proposes the automatic extraction of prosodic cues related to three prosodic elements.

Resources for the automatic conversion of *txt* hierarchical thematicity to TextGrid format and from TextGrid format to *csv* format as well as *arff* files and pitch and intensity objects derived from the audio files used in the example application are made available in the authors' repository.[5] The material from the example application described in this paper (despite its modest size) is the first annotated resource to study the correspondence in English between prosody and hierarchical thematicity as described by Mel'čuk (2001). Moreover, preliminary experiments (Domínguez et al., 2014; Domínguez et al., 2016a) already proved the adequateness of tripartite hierarchical thematicity over traditional binary approaches and its applicability to prosody enrichment in speech synthesis applications (Domínguez et al., 2017).

An automatic approach for the annotation of hierarchical thematicity based on syntactic dependencies is currently being looked into. This advance combined with the present methodology will foster empirically-grounded models to study the Information Structure–prosody interface and to allow the integration of communicatively-oriented approaches to speech technologies.

7. Acknowledgements

This work is part of the KRISTINA project, which has received funding from the *European Union's Horizon 2020 Research and Innovation Programme* under the Grant Agreement number H2020-RIA-645012. It has been also partly supported by the Spanish Ministry of Economy and Competitiveness under the María de Maeztu Unit of Excellence Programme (MDM-2015-0502), and the third author is partially funded by the *Ramón y Cajal* program.

[5]This material as well as the code of each module is available under a Creative Commons GNU v.3 License in the following repository: https://github.com/TalnUPF/compilationISpros/

8. Bibliographical References

Baumann, S. (2012). *The Intonation of Givenness. Evidence from German*. Max Niemeyer Verlag, Berlin, Boston.

Beckman, M. E. and Pierrehumbert, J. (1986). Intonational Structure in Japanese and English. *Phonology Yearbook*, 3:255–310.

Bock, J. K., Mazzella, J. R., and Bock, K. (1983). Intonational marking of given and new information: Some consequences for comprehension. *Memory & Cognition*, 11(1):64–76.

Boersma, P. and Weenink, D. (2017). Praat: doing phonetics by computer [Computer program], http://www.praat.org/, version 6.0.14.

Bohnet, B., Burga, A., and Wanner, L. (2013). Towards the annotation of penn treebank with information structure. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1250–1256, Nagoya, Japan.

Brown, G. (1983). Prosodic structure and the given/new distinction. In A. Cutler et al., editors, *Prosody: Models and Measurements*, pages 67–77. Springer, Berlin, Heidelberg.

Büring, D. (2003). On d-trees, beans, and b-accents. *Linguistics & Philosophy*, 26(5):511–545.

Calhoun, S. (2010). The centrality of metrical structure in signalling information structure: A probabilistic perspective. *Language*, 1(86):1–42.

Chafe, W. L. and Li, C. N. (1976). Givenness, contrastiveness, definiteness, subjects, topics, and point of view in subject and topic. *Subject and Topic*, pages 25–55.

Charniak, E. and al., E. (2000). *BLLIP 1987-89 WSJ Corpus Release 1 LDC2000T43*. Linguistic Data Consortium, Philadelphia.

Clark, H. H. and Haviland, S. E. (1977). Comprehension and the given-new contract. *Discourse production and comprehension. Discourse processes: Advances in research and theory*, 1:1–40.

Domínguez, M., Farrús, M., Burga, A., and Wanner, L. (2014). Towards Automatic Extraction of Prosodic Patterns for Speech Synthesis. In *Proceedings of the 7th International Conference on Speech Prosody*, pages 1105–1109, Dublin, Ireland.

Domínguez, M., Farrús, M., Burga, A., and Wanner, L. (2016a). Using hierarchical information structure for prosody prediction in content-to-speech applications. In *Proceedings of the 8th International Conference on Speech Prosody*, pages 1019–1023, Boston, USA.

Domínguez, M., Farrús, M., and Wanner, L. (2016b). An Automatic Prosody Tagger for Spontaneous Speech. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 377–387, Osaka, Japan.

Domínguez, M., Latorre, I., Farrús, M., Codina, J., and Wanner, L. (2016c). Praat on the Web: An Upgrade of Praat for Semi-Automatic Speech Annotation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 218–222, Osaka, Japan.

Domínguez, M., Farrús, M., and Wanner, L. (2017). A Thematicity-based Prosody Enrichment Tool for CTS. In *Accepted to show and tell demonstrations at INTERSPEECH17*, Stockholm, Sweden.

Féry, C. and Kügler, F. (2008). Pitch accent scaling on given, new and focused constituents in German. *Journal of Phonetics*, 36(4):680–703.

Féry, C. (2013). Focus as prosodic alignment. *Natural Language & Linguistic Theory*, 31(3):683–734.

Fowler, C. A. and Housum, J. (1987). Talkers' signaling of "new" and "old" words in speech and listeners' perception and use of the distinction. *Journal of Memory and Language*, 26(5):489–504.

Haji-Abdolhosseini, M. and Müller, S. (2003). Constraint-Based Approach to Information Structure and Prosody Correspondence. In *Proceedings of the 10th International Conference on Head-Driven Phrase Structure Grammar*, pages 143–162. CSLI Publications.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1).

Mel'čuk, I. A. (2001). *Communicative Organization in Natural Language: The semantic-communicative structure of sentences*. Benjamins, Amsterdam, Philadephia.

Meurers, D., Ziai, R., Ott, N., and Kopp, J. (2011). Evaluating Answers to Reading Comprehension Questions in Context: Results for German and the Role of Information Structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, TIWTE '11, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Schwarzschild, R. (1999). Givenness, avoidf and other constraints on the placement of accent. *Natural Language Semantics*, 7(1):141–177.

Silverman, K., Beckman, M., Pitrelli, J., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J., and Hirschberg, J. (1992). TOBI: A Standard for Labeling English Prosody. In *2nd International Conference on Spoken Language Processing (ICSLP 92)*, pages 867–870, Banff, Canada, October.

Steedman, M. (2000). Information structure and the syntax-phonology interface. *Linguistic inquiry*, 31(4):649–689, Fall.

van Donselaar, W. and Lentz, J. (1994). The function of sentence accents and given/new information in speech processing: different strategies for normal-hearing and hearing-impaired listeners? *Language and Speech*, 37(4):375–391.

Vanrell, M., Mascaró, I., Torres-Tamarit, F., and Prieto, P. (2013). Intonation as an Encoder of Speaker Certainty: Information and Confirmation Yes-No Questions in Catalan. *Language and Speech*, 56(2):163–190.

Xu, Y. (1999). Effects of tone and focus on the formation and alignment of f0 contours. *Journal of Phonetics*, 27(1):55–105.

Preliminary Analysis of Embodied Interactions between Science Communicators and Visitors Based on a Multimodal Corpus of Japanese Conversations in a Science Museum

Rui Sakaida[1], Ryosaku Makino[2], Mayumi Bono[1,3]
[1] National Institute of Informatics, [2] Faculty of Human Sciences, Waseda University,
[3] Graduate University for Advanced Studies (SOKENDAI)
[1] 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan
[2] 2-579-15 Mikajima, Tokorozawa city, Saitama, 359-1192, Japan
lui@nii.ac.jp, rmakino@aoni.waseda.jp, bono@nii.ac.jp

Abstract

This paper introduces preliminary analyses of embodied interactions, drawing on a multimodal corpus of Japanese conversations, which we video-recorded during scientific communications at a museum in Tokyo, the Miraikan. A comparison of similar cases extracted from our multimodal corpus shows both similarities and differences, not only in language use but also in bodily conduct in certain interactional sequences. We focus on a number of sequences, such as those where science communicators invite visitors to walk to the next exhibit, and our detailed analyses show that the practices of science communicators are context-free and context-sensitive interactional procedures, adapted and adjusted to the different situations communicators may encounter. After presenting our analyses, based on a corpus from a naturally occurring but partly controlled setting, we suggest that we can investigate both the generality and the situatedness of interactional practices. In the future, using such multimodal corpora, we will be able to both qualitatively and quantitatively analyze language use and non-verbal behaviors in situated activities.

Keywords: Multimodal corpus, Embodied interaction, Scientific communication

1. Introduction

This paper introduces a preliminary analysis of embodied interactions based on a multimodal corpus of Japanese conversations that we video-recorded in a specific social setting, i.e., scientific communication in a museum. We filmed face-to-face conversations between science communicators (SCs) and visitors at the Miraikan science museum in Tokyo (Figure 1). The data of this corpus, which we call the Miraikan Science Communication Corpus, were recorded in a very specific situation, unlike many classical corpora, which contain strictly controlled data. In this paper, we demonstrate that we cannot analyze both the generality and situatedness of our language use and non-verbal behaviors in conversations without drawing on such a unique corpus.

In the following sections, we first discuss why we should construct multimodal corpora recorded in complicated situations. Then, considering the need for such multimodal corpora, we explain the characteristics of our corpus. Next, we describe how we recorded the data and made annotations. Finally, we present our preliminary analysis of the corpus—the method was qualitative and comparative—suggesting the usefulness of the corpus.



Figure 1: Scientific communication in the Miraikan.

2. Background

As language resources, many corpora of Japanese conversations have been constructed and published in the field of corpus linguistics. For example, Den and Enomoto (2007) constructed a corpus of video-recorded three-party conversations. In each session, three participants were asked to sit around a table and talk about a topic selected at random, and started/ended by talking with the researcher. In this way, we can design a highly controlled dataset of both spoken language and non-verbal behaviors, such as eye-gaze and gestures.

Social scientists have long enthusiastically recorded and analyzed naturally occurring conversations, which would have occurred with or without being recorded by researchers (e.g., Schegloff, 1968; Sacks et al., 1974). It is not until we observe naturally occurring conversations that we can investigate the procedures by which conversations properly start or end (Schegloff and Sacks, 1973). Often, during conversations, the participants stand facing each other (Kendon, 1990) or even walk to another place (Mondada, 2012).

There is a methodological division between corpus linguistics and the social sciences. For researchers in corpus linguistics, highly controlled language resources are useful for analyzing the general characteristics of language use or non-verbal communication quantitatively or statistically. For social scientists, experimentally recorded corpora are not satisfactory for analysis of the variety of situational organizations of our language use or non-verbal behaviors. Therefore, many social scientists have recorded data suitable for their own research questions, instead of drawing on highly controlled corpora. However, conversations recorded by social scientists are often too complicated and improvisational to analyze the language use or non-verbal behaviors statistically or comparatively. In general, social scientists prefer to depend on and stick to the data that they have recorded themselves, rather than share their data with other researchers or research communities. Thus, each social scientist must record,

annotate or transcribe, and analyze his or her data individually, which takes a great deal of time and effort.
Therefore, to connect the motivation of social scientists to analyze naturally occurring conversation with corpus linguistics, corpora of naturally occurring, but somewhat controlled, conversations must be constructed and published.

3. Characteristics of the Corpus

Considering the need for corpora of natural, but controlled, conversations, we made a corpus with the following characteristics.
First, the corpus was recorded in a semi-institutional setting, i.e., scientific communication in a museum. The social sciences have focused on and analyzed social interactions in highly institutional settings for many years (Drew and Heritage, 1992); for example, in medical settings (Heath, 1986), classrooms (Mehan, 1979), or courtrooms (Atkinson and Drew, 1979). Turn taking, sequence organization, and bodily orientation occur in different orders that are typical of each setting. By contrast, scientific communication in the Miraikan is not as institutionalized as the communication in other social settings, especially for the visitors (Bono et al., 2014). In general, people visiting museums or galleries not only interact with their companions but also react to strangers, negotiating which exhibits to look at or when to move to the next exhibit (vom Lehn et al., 2001; vom Lehn, 2013). In contrast to the usual modes of visiting museums, the characteristics of the interactions found in our corpus are unique. The visitors, not knowing what kind of communication they are supposed to conduct in the Miraikan, try to understand the sort of activity they are involved in, the membership category (Sacks, 1972) that is relevant for themselves, and how they should respond to the SCs.
In addition, conversations in the corpus are embedded in a complicated environment, i.e., an exhibition room in the Miraikan. Therefore, they involve a complicated embodied activity, i.e., appreciation and explanation of the exhibits. Our embodied actions in conversations are usually coupled with environmental elements (Streek et al., 2011). In particular, many of our gestures are coupled environmentally (Goodwin, 2007). For example, the meaning of a pointing gesture is made clear because it is tied to the objects being pointed out (Goodwin, 2003). Furthermore, conversations are not simply conducted while sitting around a table, which is the case in most conversation corpora. Rather, we frequently talk while standing and facing each other, operating or manipulating objects, conducting joint activities, or even walking together. Recently, spatial configuration (Kendon, 1990), objects (Nevile et al., 2014), multiactivity (Haddington et al., 2014), and mobility (Haddington et al., 2013) in conversations have been hot topics in interaction studies. In our corpus, by virtue of the very complicated environment of the exhibition room, the participants have to pay attention to various exhibits or move between exhibits. The SCs often point to the exhibit relevant to the current topic of conversation and encourage the visitors to pay attention to it. Or, after explaining an exhibit, the SCs may invite the visitors to walk to the next exhibit together.

4. Recording and Annotation

4.1 Recording

To construct the corpus, we asked the National Museum of Emerging Science and Innovation (the Miraikan) to allow us to video-record routine conversations between the science communicators (SCs) and visitors. However, to control the setting to a certain extent, the space used for the recordings was separated from the rest of the exhibition room. The themes of the exhibition area where we made the recordings were the "Spread of Space" and "Challenge the Universe with a Giant Telescope".
The staff of the Miraikan selected 15 expert SCs to participate in our recordings. The aim of our project and how we would manage the data was explained to all of the SCs and visitors who participated in the recordings, and they granted us permission not only to use the data for our own work but also to publish the videos, transcripts, and annotation data as a multimodal corpus.
The video-recordings were made over 10 days in February and March 2013, for about 1 hour per day. On each day, we asked two SCs to talk separately with three groups of visitors as naturally as possible. Although we did not tell the SCs which exhibit to explain or the route to the next exhibit, they usually followed the same route and explained the same series of exhibits. Most of the SCs started the talk with an explanation of the model of the solar system, and invited the visitors to the next exhibit, a model of the Subaru Telescope. Each group consisted of from 1 to 5 people. The average group had 2.26 visitors.

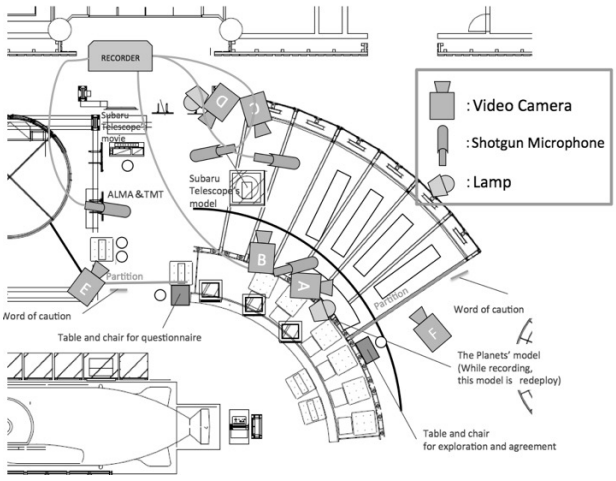


Figure 2: Layout of the cameras and microphones.



Figure 3: Screenshot of the merged data.

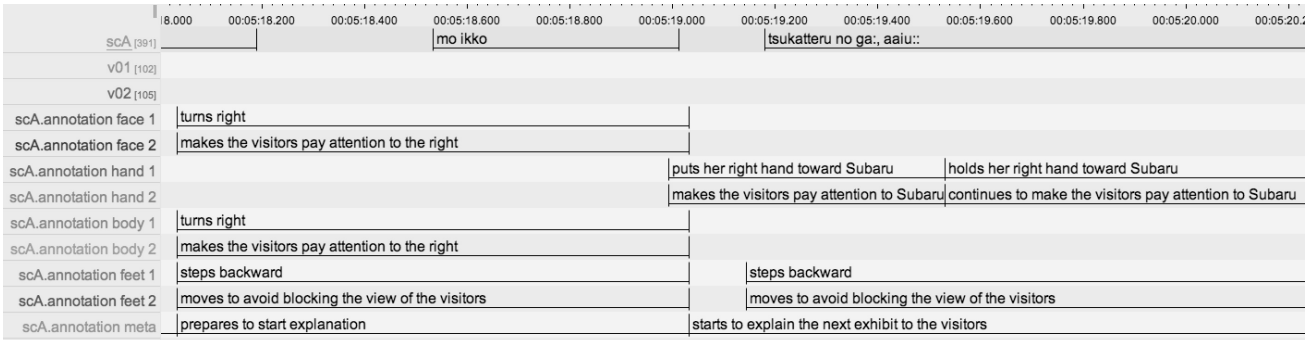


Figure 4: Example of relevant annotation (Bono and Sunaga, 2016)

After the recording, participants were asked to give basic personal information, such as their age, writing it on their face sheets. The participants were from 5 to 66 years old, and the average age was 31.26 (SD = 15.78)[1].

The recording instruments included six video cameras[2] and seven microphones (Figure 2). Five video cameras were fixed around the separated area, while a professional camera operator recorded the front view of the participants with a mobile video camera. Similarly, four shotgun microphones[3] were fixed in position and several pin microphones[4] were attached to the participants' chests. After the recording, all the recorded sound was mixed, with the noise removed.

As a result, 15 expert SCs and 79 visitors participated in the recordings, and 35 sessions were recorded in total. The average length of the sessions was about 14 minutes. All of the video and audio data were merged into one file for each session (Figure 3).

4.2 Annotation

To facilitate use of the corpus, we annotated the data in several ways, using ELAN[5]. First, all of the utterances and vocal behaviors such as laughing or coughing were segmented into inter-pausal units (Koiso et al., 1998) and transcribed in Japanese[6]. In addition, the body movements of the SCs that were relevant to the current interactional practice were annotated. Inspired by Ethnomethodological Conversation Analysis, we described the action that each physical movement achieved. We call the annotation method "Relevant Annotation" (Bono and Sunaga, 2016). In this way, the face, hand, body, and feet movements of each SC were annotated and the action that all of the movements achieved was described in meta-level tiers (Figure 4). To date, Relevant Annotation has been used to target selected segments from 32 data files, and the total length of the annotated data is about 36 minutes.

5. Analysis: Synchronous Walking during Conversation

Thanks to the characteristics of the corpus, we can compare similar sequences of naturally occurring, but somewhat controlled, embodied interactions between SCs and visitors. At present, our analyses are not statistical or quantitative, but qualitative. Nevertheless, our analyses suggest that a

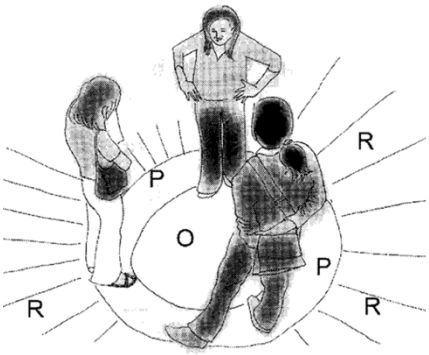


Figure 5: F-formation[7]

multimodal corpus of a natural, but semi-controlled, situation can be useful for comparative studies of both verbal and nonverbal behaviors. For example, it is common to focus on turn design (Drew, 2012), and by analyzing it, we can demonstrate the way in which interlocutors use a specific format of utterance. Joh et al. (2015) demonstrated that science communicators draw on a certain format of utterance, such as, "Do you know X?" to introduce new topics of conversation, simultaneously drawing the visitors' attention to the next exhibit to be shown. Also, we focus on participants' body movements and illustrate the way in which a particular body movement functions in conversations. Makino et al. (2015) found that a unique standing position, which we call the "H-formation", is employed to display readiness to begin explaining an exhibit. In this paper, we analyze how SCs and visitors together move from exhibit to exhibit.

In many studies of multimodal interactions, walking has been investigated as a typical means of establishing an encounter before beginning a conversation (Kendon, 1990; Mondada, 2009) and then breaking it up again after the conversation is completed (Heath, 1986; Broth and Mondada, 2013). Of course, we may walk *during* a conversation or interaction. As a matter of fact, in interaction studies, mobility has been the subject of the most pioneering research (McIlvenny et al., 2009, 2014; Haddington et al., 2013). Walking together is such an ordinary activity in our social life that we rarely regard it as complex, or as a particular accomplishment. However, many empirical studies have suggested that the organization of walking together is a complexly situated

[1] Nine visitors did not write their ages on the face sheets.
[2] SONY HDR XR550V/PMW EX1R
[3] SENNHEISER MKH 416-P48U3/AKG C414-XLII
[4] SONY UWP-V1
[5] https://tla.mpi.nl/tools/tla-tools/elan/
[6] The original annotations are written in Japanese, and for this paper the authors translated them into English.
[7] The picture is originally from Kendon (1990), and was altered into illustration by Bono (2008).

issue for participants in naturally occurring social settings, such as in guided tours (De Stefani and Mondada, 2014), a supermarket (De Stefani, 2013), dance classes (Broth and Keevallik, 2014), or situations mediated by specific objects, such as revolving doors (Weilenmann et al., 2014).

Following Kendon (1990), people establish an F-formation, facing one another and engaging in overlapping transactional segments, when they stand in public places and interact (Figure 5). By contrast, when people move to a new place while maintaining the interaction, the F-formation dissolves, because it is difficult to walk forward while facing another person (De Stefani and Mondada, 2014). The dissolution of the original F-formation can lead to an absence of conversation or end the current conversation. For instance, it can be difficult for a hearer to display his/her availability or recepiency (Heath, 1986) to the speaker while walking; this can appear as a weak orientation toward the conversation. To elucidate how people can continue to talk even when walking, we observed the SCs and visitors twice as they moved between the two exhibits in the exhibition room, while maintaining their communication.

In the following analysis, we used the annotations explained above and presented additional annotations where necessary. For instance, excerpt 1 (Figure 6), which is analyzed below, was extracted from the example of the relevant annotation shown above (Figure 4). The annotations were edited and revised by the authors. In the transcripts, using the transcription system introduced by Mondada (2009, 2012), vocal utterances are given in black letters and body movements are in gray letters.

5.1 Excerpt 1

In excerpt 1 (Figure 6), an SC successfully invites the two visitors to the next exhibit, which is a model of the Subaru Telescope, in a multimodally organized and sequentially relevant way. In the time immediately preceding the excerpt, the SC had begun to explain the three ways in which researchers investigate the universe. In line 01, the SC says *konpyuutaa wo tsukatte shumireeshon wo suru no to:* (doing the simulation using a computer, and), which refers to the second way of studying the universe (the first way had already been referred to before the start of the excerpt.) After noting the second way, the SC utters *to:* (and), (line 01) and suspends her utterance. At the same time, the SC turns right and steps backward. Immediately thereafter, she looks at the Subaru, saying *mo ikko* (one more thing) (line 02). As the SC is saying this, the visitors respond to the multimodal invitation by the SC, turning to the right. At the end of her utterance, the SC points to the Subaru with her right hand, saying *tsukatteru no ga:, aaiu::* (they are using, that sort of) (line 02). *aaiu* is a kind of distal demonstrative adjective in Japanese, which can spatial-deictically refer to something at a distance from both the speaker and the hearer (cf. Hayashi, 2004) and project the referent itself will be uttered immediately.

```
01  SC  konpyuutaa wo tsukatte shumireeshon wo suru no to+:, (0.2)+(0.15)
          computer    P    use       simulation    P   do  thing and
        "Doing the simulation using a computer, and,"
    sc                                                  +turns right and steps backward
    sc                                                           +looks at Subaru
02  SC  mo† i*kko +(0.15) tsukatteru no ga:, aa*i†u:: (0.05)+(0.45) booenkyo+o.
        more  one             using    thing P      that                  telescope
        "one more thing,"      "they are using, that sort of,"                "telescope."
    sc            +points to Subaru                          +stops pointing and looks at Vs
    sc                                                                      +starts to walk->
    v1      *turns right and steps backward   *steps forward a little
    v2    †turns right                          †starts to walk->
03      (0.25)
04  V1  (u*:+::n).
          huh
        "Huh."
    sc      +looks forward while nodding
    v1    *starts to walk->
05  SC  booenkyo+o tte mita koto ari masu ka:?
          telescope    P   saw  have_ever JD_PL  P
        "Have you ever seen a telescope?"
    sc          +turns to Vs while walking->
06  V1  (0.35) <a[ri masu:>
                 have JD_PL
                "Yes, I have."
07  V2           [booenkyoo (0.3) [un, a, un un (0.7) un.
                   telescope       yeah  oh yeah yeah      yeah
                 "A telescope...yeah, oh, yeah…yeah."
08  SC                           [oho
                                  "Oh."
09  V1  ufufu[fufufu.]
10  SC       [sa + su]ga: (0.05)+(0.3)+(1.2)†(0.25) mi*raikan ni irassyaru dake ari masu ne:.
               great                                  Miraikan   to   come_PL   may_well JD_PL P
             "Great!"                                "You may well come to Miraikan."
    sc          +turns to Subaru briefly
    sc                      +turns to Vs
    sc                            +stops walking
    v1                                  *stops walking
    v2                               †stops walking
```

Figure 6: Transcript of excerpt 1.

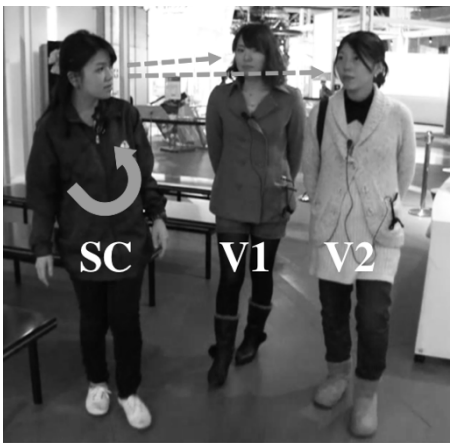


Figure 7: Spatial formation while walking in excerpt 1.

When the SC is saying *aaiu::*, both of the visitors react to the SC in different ways. While V1 steps forward a little, V2 begins to walk. After saying *aaiu::*, during a pause, the SC stops pointing at the Subaru and looked at the visitors. Orienting herself to the visitors, the SC says *booenkyoo* (telescope) (line 02) and, as soon as she completes her turn, she begins to walk. After a silence in line 03, noticing that both of the interactants had begun to walk, V1 also begins to walk. In this way, the SC succeeds in inviting the visitors to pay attention and start walking to the next exhibit, with the topic of their conversation naturally transitioning from the general explanation of astronomy to a specific exhibit. Even after beginning to move to the next exhibit, the SC continus to orient herself to maintaining conversation with the visitors. Just after starting to walk, V1 responds to the SC, saying *u:::n* (huh) (line 04), to which the SC also responds with a nod, while looking forward. According to Goodwin (1981), if they look away from the speaker, hearers use vocal tokens or nods to avoid the loss of their display of recepiency. Here the SC orients to maintaining the conversation, even without gazing at her interactants. In line 05, the SC asks a question of the visitors, *booenkyoo tte mita koto ari masu ka:?* (Have you ever seen a telescope?). During this question, the SC turns to the visitors while continuing to walk forward, adopting a particular posture called body torque (Schegloff, 1998) (Figure 7). This posture embodies the SC's orientation to two actions: walking and talking. Maintaining the specific posture, the SC, gazing at the visitors, continues to talk with them until they reach the next exhibit (lines 06–10). Distributing her body orientation to several actions (Nishizaka, 2013), the SC skillfully ensures the continuance of their interactional space (Mondada, 2009) even while moving to the next exhibit.

5.2 Excerpt 2

In excerpt 2 (Figure 8), the SC's practice in the first half is, to some extent, similar to that of the SC in excerpt 1, although the SCs in the two excerpts are different. As in excerpt 1, the SC in this excerpt first points to the next exhibit, Subaru (line 04) and then starts to walk, inviting the visitors to move to it (line 05). However, we can observe a rather different practice in the second half.

```
01  SC  gunma tenmondai ga tabun okkii desu ne.
        Gunma observatory P probably big JD_PL P
        "Gunma observatory is probably big."
02      (0.3)
03  V2  a, (0.3) a, sooiu [koto,
        oh       oh   so    thing
        "Oh, oh, it means that."
04  SC                     [itten  go meetaa (.) no: (0.45) kagami, +chou(do) dakara: (0.2)
                           one point five meter      P          mirror     exactly     so
                            "One point five meters',"             "mirror, so exactly,"
    sc                                                                  +looks at and points to Subaru
05  SC  are desu yo. (0.1)+(0.9)       +(0.15) kono†(0.25)*(0.15)
        that JD_PL P                           this
        "that is it."                          "This,"
    sc                  +walks aside +looks at the Vs
    v1                                              *starts to walk->
    v2                                           †starts to walk->
06  V1  (unn?)
        "Huh?"
07  SC  subaru to on+naji youna: katachi wo (0.75) shite masu.
        Subaru P  similar  like   shape  P          be  JD_PL
        "Subaru's shape is similar ((to that of Gunma observatory's telescope))."
    sc              +turns to Subaru and walks toward it->
08      (1.15)
09  SC  tsutsu jana+ku te,  moo kanzenni, (.) oopun desu kedo. (0.15)+(0.15)†(0.05)
        cylinder    not   P  exactly perfectly      open JD_PL though
        "It is no longer a kind of cylinders, and perfectly open though."
    sc              +turns to Vs and points to Subaru              +stops walking
    v2                                                                 †stops walking
10  SC  kore, subaru no (0.15)*(0.6) mokei nan'desu kedo.
        this  Subaru  P              model  JD_PL  though
        "This is Subaru's model."
    v1                        *stops walking
```

Figure 8: Transcript of excerpt 2.



Figure 9: Spatial formation before walking in excerpt 2.


Figure 10: Spatial formation while walking in excerpt 2.

In the first part of the excerpt, the SC and the visitors are speaking of the way in which they can watch stars in the sky. In line 01, the SC, in saying *Gunma tenmondai ga tabun okkii desu ne* (Gunma observatory is probably big), answers the question, which had been asked by one of the visitors just before this excerpt, of where to go to watch the stars. V2, who had asked the question, responds to her response, saying *a, a, sooiu koto* (Oh, oh, it means that).

In line 04, the SC continues to explain in detail about the Gunma observatory. The SC refers to *itten go meetaa no kagami*, (a 1.5-meter mirror,) (line 04), with which the telescope of Gunma observatory is equipped. Immediately after saying this, the SC begins to look at and point to the model of the Subaru telescope, stating *chou(do) dakara are desu yo* (so exactly, that is it) (lines 04–05). *are* (that) is a Japanese distal demonstrative pronoun, and by using it speaker can implicitly request the interlocutor(s) to specify what it refers to, typically by connecting the speaker's body movement with the surrounding environment or any object around them. In excerpt 1, the SC used a similar distal demonstrative *aaiu* (that sort of), which is generally used as an adjective to strongly project the immediate occurrence of the noun it modifies, as mentioned in 5.1. On the other hand, *are* is a demonstrative pronoun, so it can be produced without mentioning the object it refers to when spatial-deictically used. Here, the SC points to the Subaru model with his right index finger, inviting the recipients to attend to the direction of the Subaru and recognize what the SC is pointing to (Goodwin, 2003). Just after speaking, the SC steps aside, toward the direction of Subaru, looking at the visitors, and then says *kono* (this) (line 05). Responding to these multimodal behaviors of the SC, the visitors begin to walk (line 05). *kono* (this) is a proximal demonstrative adjective, which implies that they should look at the referent, i.e., the model of Subaru, more closely than from the current position. The SC's word selection and modification, that is, at first saying *are* (that) and then *kono* (this), successfully invite the visitors to walk toward the Subaru to look more closely.

In the latter half of this excerpt, the SC takes the visitors to the next exhibit in a rather different way to that in excerpt 2. Unlike in excerpt 1, the SC in excerpt 2 walks forward, without gazing at the visitors. The F-formation, which had been in effect before walking (Figure 9), is completely dissolved during the walk (Figure 10). Instead, the SC employs another way to continue their conversation. In line 07, the SC turns to the Subaru and starts to walk to it, producing a continuation of the utterance he had already begun before walking, *subaru to onnaji youna: katachi wo shite masu* (the Subaru's shape is similar). In line 05, the utterance of the SC, *kono*, projected that his turn-constructional unit (Sacks et al., 1974) was not yet completed and would lead to an explanation of "this". In fact, after a pause (line 05), the SC begins to continue the explanation of "this", with the relationship between the current topic of conversation and the next exhibit, i.e., Gunma observatory and Subaru observatory, becoming clear. In this sequence, the SC succeeds in taking the visitors to the next exhibit, in accordance with the visitors' previously expressed interest, that is, in watching stars in the sky. This practice appears to be an alternative to the practice in excerpt 1, i.e., looking at the visitors while continuing to talk.

6. Discussion

Our analysis in section 5 identified the following interactional practices. Specifically, the SCs and visitors were able to proceed in concert to the next exhibit while maintaining their conversation by engaging in the following behaviors in a step-wise fashion. First, the SC referred and pointed to the next exhibit, a model of the Subaru Telescope, thereby inviting the visitors to look at it. Then, the visitors looked at the exhibit in question[8]. Third, without saying anything about walking, the SC started to walk toward the next exhibit, inviting the visitors to follow. Fourth, the visitors followed the SC, demonstrating their implicit agreement with this plan of action. Fifth, the SC continued to talk to the visitors until they all reached the next exhibit.

Two aspects of these behavioral patterns seem to enable the SCs to maintain their conversation. First, before starting to walk, the SCs pointed to the next exhibit; this established the template for the new F-formation to be created at the next exhibit after the current F-formation had dissolved. Second, by not explicitly referencing the move to the next exhibit and by continuing to talk to the visitors while moving, the SCs implied that walking *per se* was not central to their activity; rather, walking was an interactional forum in which their main activity occurred. Consequently, even if the F-formation was dissolved, the SCs could maintain their scientific communication.

Nevertheless, there were several observable differences in the interactional practices between the two excerpts. First, the ways each of the SCs invited the visitors to attend to Subaru were different. In excerpt 1, the SC stepped

[8] In excerpt 2, from the beginning, the visitors were facing the Subaru, so it was not clear exactly when they were closely attending to it, but they certainly did so, responding to the SC's invitation.

backward before pointing to Subaru so that the visitors could clearly look at it. On the other, in excerpt 2, the SC pointed to Subaru without stepping toward it, as he was not obstructing the visitors' view in that position. As a result, the SC's multimodal conduct in excerpt 1 more strongly projected the following sequence, i.e., starting to walk and moving to the next exhibit. One of the visitors (V2) in excerpt 1, actually began walking earlier than the others in both excerpts, even before the SC herself had begun walking. Secondly, the ways the SCs drew the visitors' attention to the conversation while walking were unique to each SC. In excerpt 1, the SC gazed at the visitors in a body-torqued position (Schegloff, 1998) while walking, maintaining an orientation toward their conversation. In excerpt 2, through the use of two different Japanese demonstratives, *are* and *kono*, and suspending his turn, the SC projected the continuation of his utterance concerning the exhibit they were walking to, even before walking. In such similar but different ways, the two SCs successfully invited their visitors to walk together and move to the next exhibit. In short, the practices of science communicators consist of context-free and context-sensitive interactional procedures (Sacks et al., 1974), which are adapted and adjusted to various situations they come across. In this way, comparing similar cases extracted from the multimodal corpus, we can see both similarities and differences not only in language use but also in bodily conduct in a certain interactional sequence.

7. Concluding Remarks

By demonstrating our analyses based on the multimodal corpus of a naturally occurring, but partly controlled setting, we investigated both the generality and the situatedness of interactional practices, exhibiting the possible uses of such corpora. In our daily lives, continuing a conversation may be difficult because of various situational factors. Nevertheless, we often manage to continue conversations in some way, in many instances, when extraordinary but quite reasonable multimodal practices are observed. Such behaviors are embedded in complicated environmental and/or social relationships, as in scientific communication in the Miraikan. In the future, by using similar multimodal corpora, we will be able to both qualitatively and quantitatively analyze language use and non-verbal behaviors in situated activities.

8. References

Atkinson, J. M. and Drew, P. (1979). Order in Court: The Organisation of Verbal Interaction in Judicial Settings. NJ: Humanities Press.

Bono, M. (2008). Units of analysis for understanding conversational structure: F-formation. *Journal of Japanese Society for Artificial Intelligence*, 23 (4): 545–551.

Bono, M., Ogata, H., Takanashi, K. and Joh, A. (2014). The practice of showing 'who I am': A multimodal analysis of encounters between science communicator and visitors at science museum. In Stephanidis, C. & Antona, M. (Eds.), *Universal Access in Human-Computer Interaction: Universal Access to Information and Knowledge*, *Lecture Notes in Computer Science*, 8514, 650–661, Berlin and Heidelberg: Springer.

Bono, M. and Sunaga, M. (2016). A proposal of annotation scheme for body movement based on participants' understandings. *JSAI SIGs Conference Papers*, SIG-SLUD-B503: 7–12.

Broth, M. and Mondada, L. (2013). Walking away: The embodied achievement of activity closings in mobile interaction. *Journal of Pragmatics*, 47: 41-58

Den, Y. and Enomoto, M. (2007). A scientific approach to conversational informatics: Description, analysis, and modeling of human conversation. In Nishida, T. (Ed.), *Conversational Informatics: An Engineering Approach*, 307–330. Hoboken, NJ: John Wiley & Sons.

De Stefani, E. (2013). The collaborative organisation of next actions in a semiotically rich environment: Shopping as a couple. In Haddington, P., Mondada, L. and Nevile, M. (Eds.), *Interaction and Mobility: Language and the Body in Motion*, 123–151, Berlin: De Gruyter.

De Stefani, E. and Mondada, L. (2014). Reorganizing mobile formations: When 'guided' participants initiate reorientations in guided tours. *Space and Culture*, 17 (2): 157–175.

Drew, P. and Heritage, J. (Eds.) (1992). Talk at Work: Interaction in Institutional Settings. Cambridge: Cambridge University Press.

Drew, P. (2012). Turn design. In Sidnell, J. & Stivers, T. (Eds.), *The Handbook of Conversation Analysis*, 131–149. NJ: Wiley-Blackwell.

Goodwin, C. (1981). Conversational Organization: Interaction between Speakers and Hearers. New York: Academic Press.

Goodwin, C. (2003). Pointing as situated practice. In Kita, S. (Ed.), *Pointing: Where Language, Culture, and Cognition Meet*, 217–242. NJ: Lawrence Erlbaum.

Goodwin, C. (2007). Environmentally coupled gestures. S. D. Duncan, J. Cassell, & E. T. Levy (Eds.), *Gesture and the Dynamic Dimension of Language: Essays in honor of David McNeill*, 195–212. Amsterdam: John Benjamins.

Haddington, P., Mondada, L. and Nevile, M. (Eds.) (2013). Interaction and Mobility: Language and the Body in Motion. Berlin: De Gruyter.

Haddington, P., Keisanen, T., Mondada, L. and Nevile, M. (Eds.) (2014). Multiactivity in Social Interaction: Beyond Multitasking. Amsterdam: John Benjamins.

Hayashi, M. (2004). Projection and grammar: Notes on the 'action-projecting' use of the distal demonstrative are in Japanese. *Journal of Pragmatics*, 36: 1337–1374.

Heath, C. (1986). Body Movement and Speech in Medical Interaction. Cambridge: Cambridge University Press.

Joh, A., Bono, M. and Takanashi, K. (2015). Constructing a dialogue at science museum: A practice of 'Do you know X?' from the perspective of interaction analysis. *Cognitive Studies*, 22 (1): 69–83.

Kendon, A. (1990). Conducting Interaction: Patterns of Behavior in Focused Encounters. Cambridge: Cambridge University Press.

Koiso, H., Horiuchi, Y., Tutiya, S., Ichikawa, A. and Den, Y. (1998). An analysis of turn-taking and backchannels based on prosodic and syntactic features in Japanese Map Task dialogues. *Language and Speech*, 41: 295–321.

Makino, R., Furuyama, N. and Bono, M. (2015). Spatial-orientational behavior for the narrative in a field: A case study on science communicators' standing position employed to display the readiness to start giving an explanation of the exhibit. *Cognitive Studies*, 22 (1): 53–68.

McIlvenny, P., Broth, M. and Haddington, P. (2009). Communicating place, space and mobility. *Journal of Pragmatics*, 41: 1879–1886.
McIlvenny, P., Broth, M. and Haddington, P. (2014). Moving together: Mobile formations in interaction. *Space and Culture*, 17 (2): 104–106.
Mehan, H. (1979). Learning Lessons: Social Organization in the Classroom. Cambridge: Harvard University Press.
Mondada, L. (2009). Emergent focused interactions in public places: A systematic analysis of the multimodal achievement of a common interactional space. *Journal of Pragmatics*, 41: 1977–1997.
Mondada, L. (2012). Garden lessons: Embodied action and joint attention in extended sequences. In Nasu, H. & Waksler, F. C. (Eds.), *Interaction and Everyday Life: Phenomenological and Ethnomethodological Essays in Honor of George Psathas*, 293–311. MA: Lexington Books.
Nevile, M., Haddington, P., Heinemann, T. and Rauniomaa, M. (Eds.) (2014). Interacting with Objects: Language, Materiality, and Social Activity. Amsterdam: John Benjamins.
Nishizaka, A. (2013). Distribution of visual orientations in prenatal ultrasound examinations: When the healthcare provider looks at the pregnant woman's face. *Journal of Pragmatics*, 51: 68–86.
Sacks, H. (1972). An initial investigation of the usability of conversational data for doing sociology. In Sudnow, D. (Ed.), *Studies in Social Interaction*, 31–73, New York: Free Press.
Sacks, H., Schegloff, E. A. and Jefferson, G. (1974). A simplest systematics for the organization of turn-taking for conversation. *Language*, 50 (4): 696–735.
Schegloff, E. A. (1968). Sequencing in conversational openings. *American Anthropologist*, 70 (6): 1075–1095.
Schegloff, E. A. and Sacks, H. (1973). Opening up closings. *Semiotica*, 8: 289–327.
Schegloff, E. A. (1998). Body torque. *Social Research*. 65 (5): 536–596.
Streeck, J., Goodwin, C. and LeBaron, C. (Eds.) (2011). Embodied Interaction: Language and Body in the Material World. Cambridge: Cambridge University Press.
vom Lehn, D., Heath, C. and Hindmarsh, J. (2001). Exhibiting interaction: Conduct and collaboration in museums and galleries. *Symbolic Interaction*, 24 (2): 189–216.
vom Lehn, D. (2013). Withdrawing from exhibits: The interactional organisation of museum visits. In Haddington, P., Mondada, L. and Nevile, M. (Eds.), *Interaction and Mobility: Language and the Body in Motion*, 65–90, Berlin: De Gruyter.
Weilenmann, A., Normark, D. and Laurier, E. (2014). Managing walking together: The challenge of revolving doors. *Space and Culture*, 17(2): 122–136.

Improving Crowdsourcing-Based Annotation of Japanese Discourse Relations

Yudai Kishimoto, Shinnosuke Sawada, Yugo Murawaki, Daisuke Kawahara, Sadao Kurohashi
Graduate School of Informatics, Kyoto University
Kyoto, Japan
{kishimoto, sawada, murawaki, dk, kuro}@nlp.ist.i.kyoto-u.ac.jp

Abstract

Although discourse parsing is an important and fundamental task in natural language processing, few languages have corpora annotated with discourse relations and if any, they are small in size. Creating a new corpus of discourse relations by hand is costly and time-consuming. To cope with this problem, Kawahara et al. (2014) constructed a Japanese corpus with discourse annotations through crowdsourcing. However, they did not evaluate the quality of the annotation. In this paper, we evaluate the quality of the annotation using expert annotations. We find out that crowdsourcing-based annotation still leaves much room for improvement. Based on the error analysis, we propose improvement techniques based on language tests. We re-annotated the corpus with discourse annotations using the improvement techniques, and achieved approximately 3% improvement in F-measure. We will make re-annotated data publicly available.

Keywords: discourse annotation, crowdsourcing

1. Introduction

Humans understand text not by interpreting clauses or sentences individually, but by linking such a text fragment with another in a particular context. To allow computers to understand text, it is essential to capture the precise relations between these text fragments. The task of analyzing these relations is called discourse parsing. Discourse relations are conventionally divided into two types: explicit and implicit. Explicit relations are overtly marked with discourse connectives such as "and" and "however." By contrast, implicit relations lack discourse connectives.

Discourse parsing is an important and fundamental task in natural language processing. Systems for discourse parsing are, however, available only for major languages, such as English and Chinese. This is because few languages have corpora annotated with discourse relations and if any, they are small in size. Moreover, creating a new corpus of discourse relations by hand is costly and time-consuming.

Kawahara et al. (2014) addressed this problem by using crowdsourcing. Crowdsourcing allows for cheap and speedy annotation. However, crowdsourcing-based annotation tends to be of poorer quality than expert annotation.

In this paper, we evaluate the quality of the annotations produced by Kawahara et al. (2014). By asking experts to annotate a part of the corpus, we evaluate the quality of the annotation. Next, we analyze the corpus and find two problems. Then we propose solutions to each of these problems. Finally, we re-annotate the corpus with discourse relations through crowdsourcing. Experimental results show that the accuracy is improved by an F-measure of 3%.

2. Related Work

There are several corpora with discourse annotations for English, such as the Penn Discourse Treebank (PDTB) (Prasad et al., 2008) and the RST Discourse Treebank (RST-DT) (Carlson et al., 2001). PDTB's and RST-DT's annotations are done as another layer on the Wall Street Journal section of the Penn Treebank. The PDTB consists 2,159 articles and each discourse relation consists of two text spans (arguments) and a relation label. The RST-DT consists 385 articles and discourse relations can be represented as a tree structure. Discourse corpora for Chinese (Zhou and Xue, 2012) and Turkish (Zeyrek et al., 2013) have also been developed based on the PDTB. In Japanese, Kaneko and Bekki (2014) built a Japanese corpus with temporal and causal relations using the Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003). They annotated only 66 sentences.

In recent years, various language resources were created through crowdsourcing. Snow et al. (2008) ran five crowdsourcing tasks including word similarity and RTE. Guillaume et al. (2016) produced a French corpus with dependency syntax annotation by using gamification. Kawahara et al. (2014) produced a Japanese corpus with discourse annotations, which we will review in section 3.

3. Annotating Discourse Relations Using Crowdsourcing

3.1. Corpus Specifications

We overview the corpus with discourse annotations produced by Kawahara et al. (2014). The target documents are web pages extracted from the Kyoto University Web Leads Corpus (Hangyo et al., 2012). Each document consists of the first three sentences of a Japanese web page. The web pages cover a variety of domains and the first three sentences are long enough to annotate with discourse relations through crowdsourcing.

They adopted a clause as the discourse unit. The clause is a span delimited by relatively strong boundaries in a sentence. They are automatically identified with hand-written rules by the KNP parser (Kurohashi and Nagao, 1994). Kawahara et al. (2014) annotated all possible combinations of clauses with discourse annotations.

Table 1 shows the discourse relation tagset. This tagset consists of two layers, where the upper layer contains three classes and the lower layer contains seven classes.

Upper type	Lower type	Example
CONTINGENCY	Cause/Reason	【ボタンを押したので】【お湯が出た】 [Since (I) pushed the button] [hot water came out]
	Purpose	【試験に受かるために】【必死に勉強した】 [To pass the exam] [(I) desperately studied]
	Condition	【ボタンを押せば】【お湯が出る。】 [If (you) push the button] [hot water will be turned on]
	Ground	【ここにカバンがあるから】【まだ社内にいるだろう。】 [Here is (his) bag] [(he) would be still in the company]
COMPARISON	Contrast	【京都は雨だが、】【宮崎は晴れだ。】 [It is raining in Kyoto] [however it is sunny in Miyazaki]
	Concession	【あのレストランは確かにおいしいが】【値段は高い。】 [That restaurant is surely good] [but the price is high]
Other or None (hereafter referred to as "OTHER")		—

Table 1: Discourse relation tagset with examples.

3.2. Crowdsourcing-Based Annotation

Annotating a corpus with discourse relations consumes a great deal of time and cost. Kawahara et al. (2014) addressed this problem by using crowdsourcing.

Crowdsourcing is a mechanism for ordering tasks to internet users (hereafter referred to as "workers"). Using crowdsourcing, we can produce language resources cheaply and speedily. However, the quality of the resources is often questionable. To mitigate this problem, Kawahara et al. (2014) asked 10 workers to answer each question and aggregated the answers.

Another technique for quality control is to simplify the annotation task by dividing it into subtasks. Kawahara et al. (2014) proposed a two-step annotation. The first subtask was to determine whether a clause pair has a discourse relation other than "OTHER." In this subtask, workers were given a document and asked to choose between "OTHER" or non-"OTHER" for every clause pair. Once 10 answers were corrected, Kawahara et al. (2014) calculated the probability that the clause pair has a non-"OTHER" discourse relation using GLAD (Whitehill et al., 2009), which proved to be more reliable than majority voting. If the probability was larger than 0.01, the clause pair was passed to the next subtask. Otherwise, the clause pair was labeled as "OTHER." This task can be seen as a filtering step because the vast majority of clause pairs are to be labeled as "OTHER."

The second subtask was to classify the discourse relation of a clause pair that passed the first subtask. In this subtask, workers were given a clause pair and its context, and asked to select one of the 7 relations. Once 10 answers were collected, Kawahara et al. (2014) calculated the probability of each discourse relation type using GLAD and assigned the discourse relation type with the highest probability to each clause pair.

Kawahara et al. (2014) conducted the two-stage crowdsourcing experiment using Yahoo! Crowdsourcing[1] and created the annotation comprising 10,000 Japanese web pages in less than eight hours (the first subtask ran for three hours, and the second one ran for five hours). In the first subtask, 9,068 clause pairs (15.3% of all the clause pairs) were passed to the next subtask, and 4,927 clauses pairs (54.3% of the 9,068 clause pairs) were annotated with discourse relations other than "OTHER" in the second subtask.

[1] http://crowdsourcing.yahoo.co.jp/

4. Evaluation and Improvement of Annotation Quality

One major question left unanswered by Kawahara et al. (2014) is how good the quality of the crowdsourcing-based annotation is. In this section, we compare a part of the crowdsourcing-based annotation with the annotation given by experts. The crowdsourcing-based annotation is hereafter referred to as the "Old Annotation." We also report two problems found in the Old Annotation and propose a solution to each of these problems.

We annotated 500 documents as gold data. Three professional annotators with background in linguistics annotated these documents. We assigned a discourse relation type with majority vote. If all annotators disagreed with each other, one consensus label was chosen through discussion. In evaluation, we randomly sampled 313 documents from gold data. This annotation is hereafter referred to as the "Expert Annotation."

Table 2 shows the accuracy of the Old Annotation if the Expert Annotation is used as gold data. We calculated a micro average of discourse relations excluding "OTHER" ($[MicroAve.]$ in Table 2), and the F-measure was 49.5%. This result indicates that the Old Annotation had much room for improvement.

We analyzed the Old Annotation and found two problems. The first problem is that some explicit clause pairs were labeled as "OTHER" in the first subtask.

(1) (i) 特に自分は料金を知っていた**ので**
[Especially because I knew the fee]
(ii) 驚くほどでした。
[it is amazing]

Example (1) has a discourse connective "ので (because)", and consequently example (1) was given Cause/Reason in the Expert Annotation. In the Old Annotation, however,

	All relations (Explicit and Implicit)			Explicit relations			Implicit relations		
	Precision	Recall	F-measure	Precision	Recall	F-measure	Precision	Recall	F-measure
Cause/Reason	0.574 (35/61)	0.565 (35/62)	0.569	0.500 (2/4)	1.000 (2/2)	0.667	0.579 (33/57)	0.550 (33/60)	0.564
Purpose	0.417 (5/12)	0.385 (5/13)	0.400	-	-	-	0.417 (5/12)	0.385 (5/13)	0.400
Condition	0.654 (17/26)	0.515 (17/33)	0.576	1.000 (7/7)	0.438 (7/16)	0.609	0.526 (10/19)	0.588 (10/17)	0.556
Ground	0.333 (3/9)	0.273 (3/11)	0.300	0.000 (0/1)	-	-	0.375 (3/8)	0.273 (3/11)	0.316
Contrast	0.167 (2/12)	0.500 (2/4)	0.250	0.000 (0/2)	0.000 (0/0)	0.000	0.200 (2/10)	0.500 (2/4)	0.286
Concession	0.636 (7/11)	0.280 (7/25)	0.389	0.500 (1/2)	0.200 (1/5)	0.286	0.667 (6/9)	0.300 (6/20)	0.414
$[MicroAve.]$	0.527 (69/131)	0.466 (69/148)	0.495	0.625 (10/16)	0.435 (10/23)	0.513	0.513 (59/115)	0.472 (59/125)	0.492

Table 2: Accuracy of the Old Annotation.

	All relations (Explicit and Implicit)			Explicit relations			Implicit relations		
	Precision	Recall	F-measure	Precision	Recall	F-measure	Precision	Recall	F-measure
Cause/Reason	0.588 (40/68)	0.645 (40/62)	0.615	0.500 (2/4)	1.000 (2/2)	0.667	0.594 (38/64)	0.633 (38/60)	0.613
Purpose	0.444 (8/18)	0.615 (8/13)	0.516	-	-	-	0.444 (8/18)	0.615 (8/13)	0.516
Condition	0.821 (23/28)	0.697 (23/33)	0.754	1.000 (10/10)	0.625 (10/16)	0.769	0.722 (13/18)	0.765 (13/17)	0.743
Ground	0.364 (4/11)	0.364 (4/11)	0.364	-	-	-	0.364 (4/11)	0.364 (4/11)	0.364
Contrast	0.083 (2/24)	0.500 (2/4)	0.143	0.000 (0/5)	0.000 (0/0)	0.000	0.105 (2/19)	0.500 (2/4)	0.174
Concession	0.500 (3/6)	0.120 (3/25)	0.194	1.000 (1/1)	0.200 (1/5)	0.333	0.400 (2/5)	0.100 (2/20)	0.160
$[MicroAve.]$	0.516 (80/155)	0.541 (80/148)	0.528	0.650 (13/20)	0.565 (13/23)	0.605	0.496 (67/135)	0.536 (67/125)	0.515

Table 3: Accuracy of the New Annotation.

example (1) was classified as "OTHER" at the first subtask. It turned out that 40.2% (384/955) of the explicit clause pairs in the Old Annotation were annotated with "OTHER." This is probably because workers had not paid attention to discourse connectives.

Moreover, about 47% (182/384) of the explicit clause pairs annotated with "OTHER" at the first subtask have discourse connectives related to Condition. Let us consider the following examples:

(2) (i) 送信が行われる**と**、
[If you send (data),]
(ii) 送信終了のメッセージが表示されます。
[the display shows the message "send completely".]

(3) (i) ミルクで粉をこねて、
[(She) kneaded a powder with milk,]
(ii) おナベで焼く**と**、
[baked it in a pot,]
(iii) パンケーキがフワッとふくらみました。
[then, a pancake was swollen.]

In example (2), the clause pair (i) and (ii) has a Condition relation, because the discourse connective "と" means "if" in English. However, in example (3), the clause pair (ii) and (iii) does not have a discourse relation, because the discourse connective "と" means "then" in English. These examples illustrate the fact that some discourse connectives have ambiguity, and it is difficult to classify such discourse connectives, especially related to Condition.

Our solution to this problem is to skip the first subtask if adjacent clause pairs have discourse connectives without ambiguity. To detect such discourse connectives, we use the KNP parser, which identifies them by hand-crafted rules. These pairs are directly passed to the second subtask. Note that adjacent clause pairs with discourse connectives are sometimes to be labeled as "OTHER" because discourse connectives can connect non-adjacent pairs. Moreover, clause pairs which have discourse connectives with ambiguity are treated as implicit relations.

The second problem is that the instruction was not instructive enough for workers to understand the criteria to determine the discourse relations. Because the discourse annotation task is a bit complicated for crowd workers, we have to simplify the annotation task and to make workers understand the criteria with a simple instruction. Kawahara et al. (2014) showed workers the following instruction:

Condition
e.g.)【明日、晴れれば】←→【ゴルフに行く】
[If it is fine tomorrow,] [I will play golf.]

Kawahara et al. (2014) expected workers to understand the criteria from the instruction. However, the result shown in Table 2 suggests that workers could not understand the criteria. To alleviate this problem, we add the description of discourse connective phrases to the instruction. This additional explanation aims to force workers to do positive language tests using discourse connectives.

Condition
(「１すれば２」と言える関係。１・２が逆でも可。)
[we can insert "if" between 1 and 2.]
e.g.)【1 明日晴れれば、】午前は買い物に行こう。【2 午後は映画に行こう。】(「１すれば２」と言える)
[【1 If it is fine tomorrow, 】let's go to the shopping in the morning.【2 Let's go to the movies in the afternoon. 】(we can insert "if" between 1 and 2)]

In the example above, if "すれば (if)" can be inserted between a clause pair, workers are expected to choose Condition. In the instructions of Cause/Reason and Contract, positive language tests are accompanied by negative language tests:

Cause/Reason
(「１したがって２」と言えるが、「１さらに２」と言えない関係。１・２が逆でも可。)
[we can insert "therefore" between 1 and 2, but cannot insert "moreover" between 1 and 2.]
e.g.)【1 雨が降った。】よく見ると【2 道が濡れている。】(「１したがって２」と言える)
[【1 It rained. 】Look closely,【2 the road is wet. 】(we can insert "therefore" between 1 and 2)]

In the example above, if "したがって (therefore)" can be inserted between the clause pair, workers are expected to choose Cause/Reason. Meanwhile, if "さらに (moreover)" can be inserted between the clause pair, workers can rule out Cause/Reason. We need the negative tests because positive tests sometimes applied to non-target clause pairs in our preliminary experiment.

5. Experiment

With the two improvement techniques described in section 4. we re-annotated the corpus with discourse annotations (hereafter referred to as "New Annotation") and evaluated the quality of the annotation.

Table 3 shows the accuracy of the New Annotation. The New Annotation achieved about 3% higher accuracy than the Old Annotation. The micro-averaged accuracy of explicit relations in the New Annotation was about 9% higher than the Old Annotation. The improvement can mainly be attributed to the first subtask where clause pairs wrongly classified as "OTHER" in the Old Annotation were now given correct labels. Table 3 also shows that the recall was 7.5% higher than in the Old Annotation. This indicates the effectiveness of the new instruction with language tests.

Let us consider the following examples:

(4) (i) どなたにも飲みやすいおいしいワインです。
[This wine is delicious and easy to drink for everyone.]
(ii) おどや販売部長の土谷が山形のワイナリーにお願いして
[Tuchiya, who is a sales manager of Odoya, requested (a brand new product) for winery in Yamagata,]
(iii) おいしいワインを作ってもらいました。
[(and) They made delicious wine.]

The discourse relation between (ii) and (iii)：
- Expert Annotation: OTHER
- Old Annotation: Purpose
- New Annotation: OTHER

(5) (i) ２泊で香川県に旅行に行ってきた。
[(I) went on a trip to Kagawa Prefecture for two nights (and three days).]
(ii) 一応おそめの夏休みである。
[It was a late summer vacation.]
(iii) 奥さんと２歳のきょうこを連れての家族旅行である。
[It was a family trip with my wife and a 2 year old (doughter), Kyoko.]

The discourse relation between (ii) and (iii)：
- Expert Annotation: OTHER
- Old Annotation: OTHER
- New Annotation: Cause/Reason

In example (4), the Old Annotation disagrees with the Expert Annotation, but the New Annotation agrees with the Expert Annotation presumably because this pair failed the language tests (we can insert "in order to"[2] between (ii) and (iii)). However, the New Annotation disagrees with the Expert Annotation in example (5). We conjecture that workers misjudged that this pair passed the language tests. To cope with this problem, we plan to add more expressions for language tests in the instruction. However, new instructions are not guaranteed to force workers to run the language tests because the instruction will become complicated for crowdsourcing. We need to craft concise instructions.

[2]This discourse connective is used for the positive language test of Purpose.

6. Conclusion

In this paper, we evaluated the quality of a Japanese corpus with discourse annotations and proposed improvement techniques based on language tests. The experiment showed that the quality of re-annotation data using our methods was improved by an F-measure of 3%. We will make the re-annotated data publicly available.

The experimental results indicate that the annotation of implicit discourse pairs remains an issue. In the future, we would like to follow in a more direct manner the workflow of the PDTB's annotation procedure for implicit pairs: (1) identifying a discourse connective that could be inserted between arguments without changing the discourse relation between them, and then (2) specifying the discourse relation (Prasad et al., 2014). To adapt this procedure to crowdsourcing, we plan to implement discourse connective identification in the form of a cloze test.

7. Bibliographical References

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Cambridge University Press.

Carlson, L., Marcu, D., and Okurowski, M. E. (2001). Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue - Volume 16*, SIGDIAL '01, pages 1–10. Association for Computational Linguistics.

Guillaume, B., Fort, K., and Lefebvre, N. (2016). Crowdsourcing complex language resources: Playing to annotate dependency syntax. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3041–3052. The COLING 2016 Organizing Committee.

Hangyo, M., Kawahara, D., and Kurohashi, S. (2012). Building a diverse document leads corpus annotated with semantic relations. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*, pages 535–544, Bali,Indonesia, November. Faculty of Computer Science, Universitas Indonesia.

Kaneko, K. and Bekki, D. (2014). Building a japanese corpus of temporal-causal-discourse structures based on sdrt for extracting causal relations. In *Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL)*, pages 33–39.

Kawahara, D., Machida, Y., Shibata, T., Kurohashi, S., Kobayashi, H., and Sassano, M. (2014). Rapid development of a corpus with discourse annotations using two-stage crowdsourcing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 269–278.

Dublin City University and Association for Computational Linguistics.

Kurohashi, S. and Nagao, M. (1994). A syntactic analysis method of long Japanese sentences based on the detection of conjunctive structures. *Computational Linguistics*, 20(4).

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. (2008). The Penn discourse treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC2008)*, pages 2961–2968.

Prasad, R., Webber, B., and Joshi, A. (2014). Reflections on the penn discourse treebank, comparable corpora, and complementary annotation. *Computational Linguistics*, 40(4):921–950.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. (2008). Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics.

Whitehill, J., Wu, T., Bergsma, J., Movellan, J. R., and Ruvolo, P. L. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, et al., editors, *Advances in Neural Information Processing Systems 22*, pages 2035–2043. Curran Associates, Inc.

Zeyrek, D., Demirşahin, I., Çallı, A. B. S., and Çakıcı, R. (2013). Turkish discourse bank: Porting a discourse annotation style to a morphologically rich language. *Dialogue & Discourse*, 4(2):174–184.

Zhou, Y. and Xue, N. (2012). PDTB-style discourse annotation of Chinese text. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 69–77. Association for Computational Linguistics.

Persian Discourse Treebank and coreference corpus

Azadeh Mirzaei, Pegah Safari
Department of Linguistics, Allameh Tabataba'i University, Tehran, Iran; Department of Technology, Alzahra University, Tehran, Iran
azadeh.mirzaei@atu.ac.ir, phsafari@yahoo.com

Abstract
This research addresses the investigation of intra-document relations based on two major approaches: discourse analysis and coreference resolution which results in building the first Persian discourse Treebank and a comprehensive Persian coreference corpus. In discourse analysis, we have explored sentence-level relations defined between clauses in complex sentences. So we specified 34682 discourse relations, the sense of the relations, their arguments and their attributes mainly consisted of the source of the message and its type. Our discourse analysis is based on a corpus consisted of 30000 individual sentences with morphologic, syntactic and semantic labels and nearly half a million tokens. Also 18336 of these sentences are double-annotated. For coreference annotation, since a document-based corpus was needed, we prepared a new corpus consisted of 547 documents and 212646 tokens which is still under development. We enriched it with morphological and syntactical labels and added coreference information at the top. Currently, we have annotated 6511 coreference chains and 21303 mentions with a comprehensive annotation scheme to compensate some specification of Persian such as being pro-drop or lacking gender agreement information.

Keywords: Persian discourse Treebank, Persian Coreference Corpus, logical relation, mention, referent.

1. Introduction

In this research we addressed the investigation of intra-document relations with two major approaches: discourse analysis and co-reference resolution. In discourse analysis, we can inspect the logical relations between sentences inside a document and also we can explore sentence-level relations defined between clauses in complex sentences. Since there was already a rich annotated Persian corpus consisted of 30000 individual sentences, we augmented sentence-level discourse information on the top of its morphologic, syntactic and semantic layers. But for coreference annotation, a document-based corpus was needed; because in this annotation, the relation between a referent and its mention is defined inside a document and it is not anymore restricted to an individual sentence. So we prepared a new corpus and enriched it with morphological and syntactical labels to be used for learning processes in the future. Then we added coreference information at the top. Currently, we are adding a new document-level discourse annotation to our new corpus as well. So in this paper, first of all, we describe the process of preparing the first Persian discourse Treebank and then elaborate the preparation process of our first comprehensive Persian coreference corpus.

2. Persian Discourse Treebank

Persian Discourse Treebank (PerDTB) as the first discourse corpus in Persian, has been developed based on the schema of Penn Discourse Treebank (weber et al., 2003; Prasad et al., 2008) which has been used in discourse projects of the other languages such as Arabic (Al-Saif and Markert, 2010), Chinese (Zhou and Xue, 2012), Czech (Mladov´a et al., 2008), Hindi (Oza et al., 2009), Italian (Tonelli et al., 2010) and Turkish (Zeyrek and Webber, 2008).
The corpus is based on nearly 30,000 sentences which has received morphologic and syntactic labels (Rasooli et al. 2013) and also went through the semantic role labeling process (Mirzaei & Moloodi 2016). Although the corpus is based on individual sentences, it's richly annotated in different levels and it can provide us valuable features for building learners in future. Also with this corpus, we can focus on intra-sentential relations (the relations inside a sentence) which are one of the main types of discourse relations. The corpus consists of 18336 complex sentences and 11646 simple sentences. In annotation process of individual sentences, if the sentence is a complex clause/sentence, according to the systemic functional grammar (Halliday & Matthiessen 2013), we have to specify the logical relations between its clauses, the type of the relations and their attributes while for simple clauses just the type of the relation is specified. In complex sentences where there is no logical relation, if there is a clause showing the source of the message, the attribute of the sentence is annotated as well. In The following examples, the first one shows these kinds of sentences with its attribute specified while in the second one with intra-sentential relation, the connective is marked.

- او با خودش گفت که هر روز دو برابر روز قبل گل‌کاری خواهم کرد.

He said to himself that he would plant each day twice the previous day. (complex clause without any relation)

- چون لکهٔ تمشک مقاوم است باید محل لکه را در محلول پودر رختشویی بخیسانید و اگر شستید و نرفت باید مقداری دوغ یا ماست رویش بریزید.

As the stain of blueberry is resisted, you have to steep it inside the detergent and if you have washed it and it still remained, you have to pour yoghurt on it. (complex clause with logical relation)

Attribute of the relation is defined according to the PDTB standards and it mainly consists of source and type. By source we mean the source of the message which can be the writer of the sentence (Wr), other person mentioned in the text (Ot) or one arbitrary person not mentioned (Arb). In our corpus there are 30795 writer (Wr), 5545 other (Ot) and 552 arbitrary (Arb) sources. Type refers to the objectivity or subjectivity of the message/ sentence stated by source. It can show us whether it is an assertion, declarative sentence or just a subjective sentence to show the source's attitudes. So the type is categorized into four groups: assertion, facts, beliefs and eventualities. Assertion and belief are both similar as they force the agent/source to be committed to the truth of the sentence while they are different in the commitment degree. Eventualities are completely different and they show the intention or attitude of the source. In our corpus, type is annotated based on the category of its verb and it is classified into four main groups which is the same as PDTB: Communicative verbs (Comm) for assertions,

Propositional Attitude verbs (PAtt) for beliefs, Factive or semi-factive verbs (Ftv) for facts and Control verbs (Ctrl) for eventualities. In our Treebank there are 2885 Comm, 561 PAtt, 1019 Ftv and 3304 Ctrl.

3. Annotation Procedure of PerDTB

From linguistic point of view, Persian Discourse Treebank is based on the systemic functional grammar and logical metafunction concept (Halliday & Matthiessen 2013) and its annotation scheme is based on the standards used in the Penn Discourse Treebank (weber et al., 2003; Prasad et al., 2008). In this scheme, first of all, the logical relations between clauses are specified. If there is any connective, the relation is categorized into explicit and the sense of the connective is determined. Since the sense classification is one of the main procedures of annotation, it has been elaborately described in the next section. After that, the arguments of the relation should be specified. Persian is a free-word-order language, so the position of connective can be sometimes permutated in the sentence and also the arguments can take different positions (one can precede the other or they can be nested). When there is no connective, the annotator inserts a meaningful connective which is added to the discourse information layer, specifies its sense and classifies the relation as implicit. AltLex is another discourse relation type which is used when the relation is alternatively lexicalized by some non-connective expression in the sentence and the last relation type, EntRel (entity-based relation), is specified when one clause contains an entity and the other one describes it. Also PDTB contains another relation type, NoRel, which is not defined in our Treebank due to the annotation of individual sentences. Table1 shows the frequency of relation types in our corpus. The last row shows simple clauses or complex sentences without logical relation in which just the attribution is specified and in the first row, Explicit and AltLex relations are counted as joint. Its reason has been elaborated in Annotators Agreement section.

Relation Types	#frequency
Explicit+AltLex	13129
Implicit	1108
EntRel	54
Clause	20371

Table 1: Distribution of relation types in PerDTB

At the end, for all of the relation types except EntRel, we can specify the attributes of the relation or its individual arguments which can take different forms. For example, to express the source of a relation we can use a clause, a prepositional phrase, an adjective phrase, etc. (e.g., scientists say, according to scientists, quoted by scientists). Also in order to facilitate the annotation process, we have developed an annotating application with administrative panels to supervise the process and guide the annotators if it was needed. Figure 1 shows the general view of the program.

4. Sense Annotation

In Explicit, Implicit and AltLex relations, sense annotation of the connective or the lexical structure is one of the main steps in the process. According to PTDB, sense has four main classes in the first level (temporal, expansion, contingency and comparison) and each class is again divided into sub-categories as the second level and even more elaborately sub-divided into sub-types in the third level. Table 2 shows the hierarchical structure of the senses.

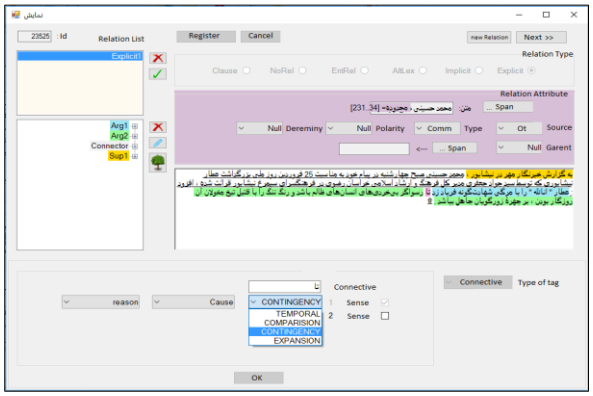


Figure 1: The general view of the annotation tool

Classes (first level)	Types (second level)	Sub-types (third level)
Comparison	Contrast, Pragmatic contrast, Concession, Pragmatic Concession	Juxtaposition, opposition (contrast) Expectation, contra-expectation (concession)
Contingency	Cause, Pragmatic cause, Condition, Pragmatic Condition	Reason, result (cause) Justification (pragmatic cause) Hypothetical, general, unreal present, unreal past, factual present, factual past (condition) Relevance, implicit assertion (pragmatic condition)
Expansion	Conjunction, Instantiation, Restatement, Alternative, Exception, List	Specification, equivalence, generalization (restatement) Conjunctive, disjunctive, chosen alternative (alternative)
Temporal	Asynchronous, Synchronous	Precedence, succession (asynchronous)

Table 2: The hierarchical structure of the senses

It is noteworthy that most of connectives are homonym and as a result they have different senses in different contexts such as “and”/و/ or “that”/که/ which sense can be categorized into all of the four main classes based on their context. Also there would be multiple sub-categories for an individual connective like “as”/چون/ (condition and cause). This homonymy can lead to different perception of the text and it can be the main reason of disagreement between annotators. The following examples show this homonymy and different sense classes for a single connective:

- پارسا هم چون حوصله‌اش سررفت شروع به لجبازی کرد. (contingency, cause)
As Parsa was bored, he started to become obstinate.

- چون تو با منی قلمرو را با یزدان دو بخش خواهم کرد. (contingency, condition)

As you are with me, I would divide the realm with God.

- چون او وارد شد، همه بر پای خاستند. (temporal)

As he entered, everyone stood up.

In the next section, we have presented some of the useful statistics of sense distribution in our Treebank which can lead to a better perception of the structure.

5. Statistics of PerDTB

PerDTB consists 29982 individual sentences. Nearly 61% of them are complex sentences with more than one verb and just about 38% are simple clauses. These complex clauses are not guaranteed to have logical intra-relations which can be proved by 40% of complex sentences classified as Clause. In most of them, there is a clause playing the role of attribute and showing the source of the message. Table 3 gives these statistics about the corpus.

#Sentences	29982
Average Length of Sentences	16.61
#Verbs	62889
# Sentences with One Verb	11617
# Sentences with Two Verbs	9917
# Sentences with More than Two Verb	8419

Table 3: Statistics of PerDTB

Table 4 shows the distribution of sense classes in three discourse relation types. Expansion is the most dominant sense with about 41% which shows that an individual sentence is more used for Expansion while Comparison with the least frequency (nearly 9%) shows that it is usually expressed through multiple sentences. This restriction to one or multiple sentences can be considered as a feature for sense classification.

Class	Explicit+AltLex	Implicit	Total
Temporal	1813	47	1860
Contingency	5383	385	5768
Comparison	1330	75	1405
Expansion	5644	673	6317
Total	14170	1180	15350

Table 4: Distribution of sense classes in PerDTB

Table 5 and 6 show the top five tags of the second and the third level of sense in the corpus.

Second level of Sense (types)	#frequency
Conjunction	4269
Cause	3548
Condition	2195
Restatement	1385
Synchronous	1014

Table 5: Top five types of senses (in the second level)

third level of Sense (sub-types)	#frequency
specification	1149
reason	1498
equivalence	196
juxtaposition	410
generalization	40

Table 6: Top five sub-types of senses (in the third level)

As we have mentioned, due to the homonymy, the sense classification of some connectives can be so confusing. Table 7 lists the six top homonym connectives.

Connective	Senses
در حالی که (during)	Temporal (30), Comparison (22), Contingency (2)
بلکه (but)	Comparison (69), Contingency (2), Expansion (29)
وقتی (when)	Temporal (295), Comparison (1), Contingency (133), Expansion (1)
تا (until)	Temporal (69), Contingency (762), Expansion (2)
که (that)	Temporal (208), Comparison (48), Contingency (800), Expansion (327)
و (and)	Temporal (360), Comparison (287), Contingency (925), Expansion (4591)

Table 7: Top six homonym connectives

Table 8 shows the top ten connectives which length is just one-token while table 9 shows the top ten connectives with more than one token in length. These connectives are compound or they are prepositional/ noun phrases that always appear continuously and their function is the same as one-token connectives.

Connective	#frequency	Connective	#frequency
و (and)	5993	وقتی (when)	377
اگر (if)	1612	ولی (however)	253
که (that)	1338	یا (or)	242
تا (until)	846	چون (since)	220
اما (but)	395	زیرا (because)	151

Table 8: Top ten one-token connectives

Connective	count	Connective	count
هنگامی که (when)	93	وقتی که (when)	53
در حالی که (during)	91	برای این که (because)	40
زمانی که (when)	67	در نتیجه (As a result)	38
هر گاه (when)	64	چرا که (because)	33
در صورتی که (if)	60	تا زمانی که (until)	25

Table 9: Top ten multi-token connectives

Table 10 shows the top ten discontinuous connectives which are consisted of two or multiple parts. Each part shows one piece of the logical relation and all of the parts, together, show one relation.

Connective	count	Connective	count
نه تنها | بلکه (Not only| But also)	30	گرچه | اما (Although| but)	9
نه | نه (Neither| Nor)	29	اگر | پس (If| Then)	9
هم | هم (Also| Also)	26	چه | چه (Either| Either)	7
یا | یا (Either| Or)	21	هرچند | اما (Although| But)	7
اگرچه | اما (Although| But)	12	اگر | دراین صورت (If| Then)	6

Table 10: Top ten discontinuous connectives

6. Inter-annotation Agreement of PerDTB

Our PerDTB annotation group consisted of four PhD candidates in linguistics and one MA graduated of Persian

language and literature. They were native Persian speakers and were presented a comprehensive guideline, describing all the logical relations with abundant examples. In order to measure their inter-annotation agreement, in the next phase, we double annotated 18336 sentences. We have used the kappa statistics (Cohen, 1960) for our measurement purpose which is defined with respect to the probability of inter-annotator agreement, P(A), and the agreement expected by chance, P(E):

$$k = \frac{\mathrm{P(A)} - \mathrm{P(E)}}{1 - \mathrm{P(E)}}$$

Our results show that the inter-annotator agreement of sense classification is 'good' with kappa value of 0.769. Also the agreement of Implicit, EntRel and Clause relation is 'very good' with k=0.856 but the agreement of Explicit and AltLex decreases to 'moderate' level with k=0.446. The reason of this disagreement can be interpreted as the state of the language. Persian, typologically is in a transitional state and it is moving from an agglutinative language toward becoming more analytic. The disagreement shows this transition. It shows that although the annotators agree on the logical function of a structure but they don't have much agreement on its essence (connective in Explicit vs. non connective expression in AltLex). This situation generally doesn't affect the annotation of logical relations between clauses.

7. Persian Coreference Corpus (PerCoref)

After developing our first Persian discourse corpus and inspecting logical relations inside complex sentences, we put a step forward to investigate intra-document relations through coreference resolution. Coreference resolution is one of the building blocks for high level NLP tasks such as question-answering systems, summarization, machine translation, etc. these reasons provoked preparation of coreference corpus in different languages like English (Hirshman, 1998), Dutch (Hendrickx, et al., 2008), Japanese (Iida, et al., 2007), Polish (Ogrodniczuk, et al., 2013), Spanish and Catalan (Recasens, & Martí, 2010). Also in Persian different coreference resources were prepared but as far as we know there is not yet any comprehensive coreference corpus in Persian to compensate some features of the language. For example, Persian is a pro-drop language, there is no gender agreement and sometimes the number agreement is ignored. So developing a comprehensive coreference corpus to cover all kinds of referential expressions and including complementary information to cover the eliminated info was needed. We tried to include any valuable information to compensate these restrictions in our corpus with elaborated coreference relation tags and covering different referential candidates. The tagset of the project is based on a coreference guideline by Komen (2009) and the theoretical issues about cohesive strategies in Halliday and Hasan (2014).

For our purpose, the previously used corpus for PerDTB was based on individual sentences and it was not suitable enough to cover coreference relations inside a cohesive text. So we developed a document based corpus. At the moment it has overall 212646 tokens and 547 documents which are majorly crawled from news articles and have been processed manually to receive morphological and syntactical labels.

8. Annotation Procedure of PerCoref

As the annotation scheme of our corpus, first of all, we specify any referential candidate (mention) which can be pronoun, noun phrase or verb for null-subject sentences. After that, their references are specified and the type of each reference is marked. Generally, the reference type can be categorized into three groups: direct reference, indirect reference or no-reference. Direct reference is subdivided into anaphoric (appear before mention) or cataphoric (appear after mention) while indirect reference as it indirectly refers to the speaker or listener of the text, is classified into speaker reference (refereed to speaker) or addressee reference (refereed to listener). However, some pronouns such as "nobody"/هیچکس/ or "other"/دیگران/ actually doesn't refer to any specific entity and we have marked them as no-reference pronouns. The following examples elaborate each type of references.

- علی به خانه آمد. همه از آمدن او خوشحال بودند. (Anaphoric)

Ali came home. Everyone was pleased with his return.

- من دیروز او، یعنی علی، را دیدم. (Cataphoric)

I see him, I mean Ali, yesterday.

- ما باید نسبت به جامعه خود بیشتر مسئول باشیم. (Indirect reference)

We have to be more responsible for our society.

- هنوز هیچکس علت را نمی‌داند. (No-reference)

No one knows the reason yet.

Table 11 shows the distribution of reference types in our corpus.

Reference Types		#frequency
Direct Reference	Anaphoric	18713
	Cataphoric	1485
Indirect Reference		281
No-Reference		824
Total		21303

Table 11: Distribution of reference types in PerCoref

In the next step, the type of the relation between the mention and its reference is investigated. This relation type is defined for Direct and Indirect references while no-reference expressions just receive some complimentary information including the number (single, plural) and a label to show the semantic class of the mention consisted of: personal, time, place or other. In the next section the coreference relation types are completely described. For the span selection of mentions, only the head of the phrasal group is included and also for each mention its closest referent is annotated. However, sometimes the referent is discontinuous with multi parts. In these cases, we have specified all separated parts in the referent span. At present, our corpus contains 6511 coreference chains and 21303 mentions. By chains, we mean the mentions refereed to the same entity.

For annotation procedure, we have developed an annotator program suited to our specific scheme. Figure 2 shows the overall view of the application. Also our annotation group consisted of one linguistic PhD, five linguistic PhD candidates, one MA graduated in Persian language and literature and one BA of English translation who were native Persian speakers supplied with a comprehensive guideline and adequate examples.

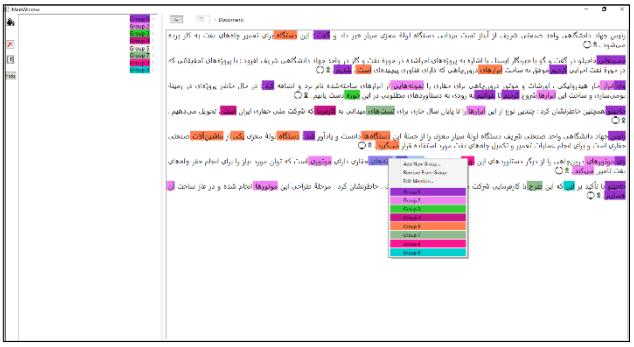


Figure 2: The overall view of the annotation tool

9. Coreferece Relation Types

Coreference relation type is one of the most valuable information added to our corpus during the annotation process. It shows the relationship between the mention and its referent. Direct and Indirect referents have their own coreference relation type categories. If the type of the referent is indirect, it is speech reference and the mention is first-person plural pronoun then there are three relation types: Exclusive-we which is defined when the mention refers to the speaker and his/her clique. Co-present-we in which the mention refers to the speaker and all of the persons who are listening to his/her notes and as the last type, all-inclusive-we which doesn't refer to any specific person and just indicates 'we' as human being. For the first-person single pronoun, all-inclusive-I is the same as all-inclusive-we but deictic-I simply indicates the writer or speaker. When the type of the referent is addressee reference, there are two major relation types: deictic and generic; the former refers to a specific person and the latter doesn't indicate any specific person but the human being as a whole. As in Persian there are two pronouns for second-person: single and plural, the type is classified into four groups. On the other hand, for direct referent, there are six major groups: identity, inferred, quantifier, cross-speech, event and person/number suffix on verb. Identity is defined when the mention and its referent both indicate the same entity in the real world. Since there is number agreement in this case, single or plural is specified. Inferred relation type unlike identity happens when the mention and referent are not identical but the first one infers the second in some way. According to this inferring style, the relation type is subdivided into five categories: hyponymy, meronymy, member-collection, antonymy and portion-mass. In hyponymy, the mention is a type of its referent and the mentions inside a chain are their co-hyponyms. In meronymy, the mention is not a type of referent but it is one of its parts. Again the mentions are co-meronyms with each other in a chain. Member-collection refers to the situation when all of the mentions in a chain can be linked together as members of a larger collection like trees in a jungle. Antonymy as its name suggests is used when the mentions are their antonyms. In portion-mass category, the mention is a measurable portion of referent as the mass. Quantifier as the third main relation type is defined to support immeasurable portions of referent used as mention like "some"/تعدادی/ or "few"/برخی/. Cross-speech, the fourth type, is used when we have direct speech in the text. In this situation the relation between mention in direct speech and the referent in indirect one is marked as cross-speech to register the disagreement in the pair. In event type, mention refers to an event introduced in the text and we annotate the verb of the clause as its referent. The last relation type is person/number suffix on verb which shows the subject of the null-subject sentences. Since Persian is a pro-drop language, including this relation type would provide valuable information. Also the number and the person of dropped pronoun are specified. The examples below show the main coreference relation types annotated in our corpus:

- من دیروز علی را دیدم و با او حرف زدم. (Identity)

I saw Ali yesterday and talked to him.

- علی به زمین افتاد. دستش صدمه زیادی دید. انگشتش شکست. مچش دررفت و زانویش کبود شد. (Inferred-hyponymy)

Ali fell down. His hand was hurt badly. His finger was broken, his wrist was displaced and his elbow was bruised.

- دیروز به مغازه رفتم. چشمم به شکر افتاد و دو کیلو خریدم. از آنجا که میوه خوبی هم داشت، مقداری میوه هم خریداری کردم.(Inferred-Portion-mass, Quantifier)

Yesterday, I went to the grocery. As I saw sugar, I bought two kilos. Since the fruits were fresh, I bought some fruits as well.

- آنها مشقشان را انجام ندادند. این خیلی بد است. (Event)

They didn't do their homework. It is so unpleasant.

- امروز از آنها تشکر کردم. دیروز هدیه‌ای به من داده‌بودند. (person/number suffix on verb)

Today I thanked them. Yesterday they had given me a gift.

- دوستم به من زنگ زد و گفت: "تو را دیروز ندیدم. حالت خوب است؟" (Cross-speech)

My Friend called me and asked: "Yesterday I didn't see you. Are you OK?"

10. Statistics of PerCoref

PerCoref consists of 547 documents which have been mainly selected from news articles with 212646 tokens. The corpus is under development and it would become larger. Table 12 shows some useful statistics of our corpus.

Number of documents	547
Number of tokens	212646
Number of sentences	6688
Number of paragraphs	4449
Avg. paragraphs per documents	8.13
Avg. sentences per paragraph	1.5
Avg. tokens per sentence	31.79

Table 12: Statistics of PerCoref

Although the Average number of sentences in paragraphs doesn't sound reasonable but as the last row shows, the sentences are long which means that the number of complex sentences tends to be much more than simple clauses and it would compensate the short average length of paragraphs.

Size of chain	frequency	% of all mentions
1	3692	17.33
2	929	4.36
3	460	2.15
4	287	1.34
5	211	0.99
6-10	521	2.44
11-50	401	1.88
More than 50	11	0.05

Table 13: Length of chains

In PerCoref, we have specified 6511 chains with average size of 3.27 mentions while the longest chain contains 87 mentions. Table 13 shows the frequency of different sizes for the coreference chains.
Table 14 and 15 show the distribution of coreference relation types for direct and indirect references respectively. As we can see, in table 14, Identity with nearly 58% is the dominant relation type for direct reference while cross-speech with less than 5% is the least frequent relation in the corpus. Person/number suffix on verb with significant frequency of about 14% as the second dominant relation shows that Persian speakers are considerably inclined to drop pronouns in their written texts.

Coreference relation type	count	% of count
Identity	11812	58.4
person/number suffix on verb	3021	14.9
Inferred	2005	9.9
Event	1509	7.4
Quantifier	968	4.7
Cross-speech	901	4.4

Table 14: frequency of relations types for direct references

Relation types for indirect referent	count	% of count
Exclusive-we	136	48.3
Co-present-we	75	26.6
Generic-you (plural)	25	8.8
Generic-we	15	5.3
Deictic-you (plural)	12	4.2
Generic-you (single)	9	3.2
Deictic-you (single)	4	1.4
Deictic-I	3	1
Generic-I	2	0.7

Table 15: frequency of relations types for indirect references

In table 15, we can see that excusive-we with 48% and co-present-we with 26% are the most frequent relation types but the usage of the other types is trivial in comparison with these two categories except Generic-you in its plural form with nearly 9% frequency.
Table 16 shows the frequency of sub types for inferred relation type.

Inferred sub type	Count	% of count
hyponymy	581	29.2
Co-hyponymy	343	17.2
Meronymy	387	19.4
Co-meronymy	118	5.9
Member-collection	470	23.6
Portion-mass	60	3
Antonym	14	0.7
Other	14	0.7

Table 16: frequency of inferred sub types

Hyponymy with 29% is the dominant sub type while its pair, co-hyponymy is in the fourth place with 17%. It shows that the length of the hyponym chains is less than their frequency. This situation is so considerable for meronymy and co-meronymy. Meronymy with 19% frequency, as the third frequent sub type, shows that this category occurs significantly while the length of these chains is too short which is shown by co-meronymy with just 5% frequency. Other in the last row, happens when the annotators cannot find suitable sub type for inferred relations.

11. Conclusion

In this paper, first of all with a focus on intra-document relations, we presented the structure and preparation process of the first Persian Discourse Treebank (PerDTB). It is based on the previously morphological and syntactical annotated corpus with nearly 30000 individual sentences concentrated on intra sentential relations. The next version of this Treebank is under development which is based on the documents in coreference corpus with half a million words to cover the other kind of relations such as the relations between two adjacent sentences. After that, in order to explore more intra-document relations, we developed the first comprehensive Persian Coreference Corpus (PerCoref) and described its preparation process. It consists of 547 documents with 212646 tokens which received POS, syntactic and coreference tags and it is still under developed. At the moment, we have annotated 6511 coreference chains and 21303 mentions and we are planning to double annotate our corpus to measure inter-annotator agreement. For more information about releasing notes of this version of PerDTB or the current version of PerCoref, you can visit Peykaregan website (http://www.peykaregan.com) and also for complementary information on PerDTB the following website is available: http://opensourceiran.ito.gov.ir/web/guest/-2.

12. Acknowledgements

Our first corpus, PerDTB, was funded by Iran Information Technology Organization (ITO) and Computer Research Center of Islamic Sciences (CRCIS) and our second corpus, PerCoref, was funded by CRCIS. During this project we had different annotation phases: PerDTB annotation, POS and syntactical annotation as preparation of coreference corpus and annotating PerCoref. We really appreciate the linguists who helped us with these annotation phases: Farzaneh Bakhtiyari (coreference preparation), Parinaz Dadras (all phases), Saeedeh Ghadrdoost-Nakhchi (all phases), Manoucher Kouhestani (PerDTB annotation), Neda Poormorteza-Khameneh (all phases), Mostafa Mahdavi (coreference preparation), Samira Mirzaei (coreference preparation) and Salimeh Zamani (all phases); and other colleagues especially Mahdi Behniafar.

13. References

Al-Saif, A. and Markert, K. (2010). The Leeds Arabic Discourse Treebank: Annotating Discourse Connectives for Arabic. *In Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC-2010)*, Valletta, Malta.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychosocial measurement*, 20(1), pp. 37-46.

Halliday, M., Matthiessen, C. M., & Matthiessen, C. (2014). *An introduction to functional grammar*. Routledge.

Halliday, M. A. K., & Hasan, R. (2014). *Cohesion in english*. Routledge.

Hendrickx, I., Bouma, G., Daelemans, W., Hoste, V., Kloosterman, G., marie Mineur, A., Van, J., Vloet, D., and Verschelde, J.-L. (2008). A Coreference Corpus and Resolution System for Dutch. *In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008)*, pp 144–149, Marrakech, Morocco. European Language Resources Association (ELRA).

Hirshman, L. and Chinchor, N. (1998). MUC-7 coreference task definition. Version 3.0. *In Proceedings of the Seventh Message Understanding Conference (MUC-7).*

Iida, R., Komachi, M., Inui, K., and Matsumoto, Y. (2007). Annotating a Japanese Text Corpus with Predicate-Argument and Coreference Relations. *In Proceedings of the Linguistic Annotation Workshop (LAW 2007)*, pp 132–139, Stroudsburg, PA, USA. Association for Computational Linguistics.

Komen, E. R., (2009). Coreference Annotation Guidelines, *http://repository.ubn.ru.nl/bitstream/handle/2066/78810/78810.pdf*.

Mirzaei, A. & Moloodi, A. (2016). Persian proposition Bank. *In Proceedings of the 10th International Language Resources and Evaluation*, Portorož (Slovenia), May, pp. 3828-3835.

Mladová, L., Zikánová, Š., and Hajičová, E., (2008). From Sentence to Discourse: Building an Annotation Scheme for Discourse Based on Prague Dependency Treebank. *In Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*.

Oza, U., Prasad, R., Kolachina, S., Meena, S., Sharma, D.M., and Joshi, A., (2009a). Experiments with Annotating Discourse Relations in the Hindi Discourse Relation Bank. *In Proceedings of the 7th International Conference on Natural Language Processing (ICON-2009)*, Hyderabad, India.

Ogrodniczuk, M., Głowi´nska, K., Kope´c, M., Savary, A., and Zawisławska, M. (2013). Polish Coreference Corpus. *In Vetulani, Z., editor, Proceedings of the 6th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pp 494–498, Pozna´n, Poland.

Prasad, R., Dinesh, N., Lee A., Miltsakaki E., Robaldo L., Joshi A. and Webber, B. (2008). The Penn Discourse TreeBank 2.0. In Proceedings of the Sixth International Language Resources and Evaluation (LREC'08), Marrakech, Morocco, May.

Rasooli, M. S., Kouhestani, M. and Moloodi, A. (2013). Development of a Persian syntactic dependency treebank. *In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* Atlanta, USA, pp. 306-314.

Recasens, M. and Martí, M. A. (2010). AnCora-CO: Coreferentially annotated corpora for Spanish and Catalan. *Language Resources and Evaluation*, 44(4):315–345.

Tonelli, S., Riccardi, G., Prasad, R., and Joshi, A. (2010). Annotation of Discourse Relations for Conversational Spoken Dialogs. *In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta.

Webber, B., Joshi A., Stone M., and Knott, A. (2003). Anaphora and discourse structure. Computational Linguistics, 29(4):545–587.

Zeyrek, D. and Webber, B., (2008). A Discourse Resource for Turkish: Annotating Discourse Connectives in the METU Corpus. *In Proceedings of IJCNLP-2008.* Hyderabad, India.

Zhou, Y. and Xue, N. (2012). PDTB-style Discourse Annotation of Chinese Text. *In Proc. 50th Annual Meeting of the ACL*, pp: 60- 77, Jeju Island, Korea.

Automatic Labeling of Problem-Solving Dialogues for Computational Microgenetic Learning Analytics

Yuanliang Meng*, **Anna Rumshisky***, **Florence Sullivan**†, **Kevin Keith**†
*Text Machine Lab, Department of Computer Science, University of Massachusetts Lowell
†College of Education, University of Massachusetts Amherst
{ymeng, arum}@cs.uml.edu
fsullivan@educ.umass.edu, kevinkeith@landmark.edu

Abstract

This paper presents a recurrent neural network model to automate the analysis of students' computational thinking in problem-solving dialogue. We have collected and annotated dialogue transcripts from middle school students solving a robotics challenge, and each dialogue turn is assigned a code. We use sentence embeddings and speaker identities as features, and experiment with linear chain CRFs and RNNs with a CRF layer (LSTM-CRF). Both the linear chain CRF model and the LSTM-CRF model outperform the naïve baselines by a large margin, and LSTM-CRF has an edge between the two. To our knowledge, this is the first study on dialogue segment annotation using neural network models. This study is also a stepping-stone to automating the microgenetic analysis of cognitive interactions between students.

Keywords: dialogue act annotation, recurrent neural networks, microgenetic learning analytics

1. Introduction

Microgenetic analysis is an observational research technique in which the researcher attends closely to the discourse interactions and the use of tools within the learning environment in order to understand the genesis (or the origins) of cognitive change (Wertsch, 1991). It has a broad impact on the design and enactment of curriculum, the design of learning environments, as well as pedagogical practices. Microgenetic Learning Analytics (MLA) (Sullivan et al., 2015) requires the collection and analysis of all interactions among students over a given period of time. It has been noted that the robustness of microgenetic analysis derives from high-density observation (Siegler, 2006). However, it is the time-consuming nature of collecting and analyzing high-density observations that has restricted the application of microgenetic analysis to small case studies.

This paper is an initial attempt to identify computational methods that will allow for automation of MLA of conversational data. Such automation will transform educational researchers' ability to perform microgenetic analysis.

This paper presents a pilot study to automate the analysis of cognitive interactions in dialogue. We have transcripts of the dialogues from middle school students solving a robotics challenge (Sullivan et al., 2015). Dialogue turns (utterances) are annotated with carefully designed labels, known as the "computational thinking codes (CT codes)". We need to build computational models to assign the codes to dialogue turns automatically. In a broader sense, this is a dialogue action (DA) annotation task, with multiple agents in the dialogue. It is also related to the sequential sentence labeling task, in which a series of sentences need to be labeled.

2. Related Work

Many different classification algorithms have been used for dialogue act annotation. The most popular ones use support vector machines (SVM) (Margolis et al., 2010; Tavafi et al., 2013), Hidden Markov models (HMM) (Kim et al., 2010; Tavafi et al., 2013), conditional random fields (CRF) (Kim et al., 2010; Tavafi et al., 2013), and decision trees (Moldovan et al., ; Samei et al., 2014). These methods require handcrafted features, and their performance depends on the application domain. Few if any works to date have used the recently popular neural network-based approaches for dialogue segment labeling.

We can also consider DA annotation a special case of sequential sentence labeling, in which every sentence is an utterance. Existing systems for sequential sentence labeling are mostly based on traditional statistical machine learning methods too, such as Naïve Bayes (Huang et al., 2013), SVM (McKnight and Srinivasan, 2003; Hirohata et al., 2008), HMM (Lin et al., 2006), and CRF (Hirohata et al., 2008; Hassanzadeh et al., 2014). Recently, popular approaches use neural networks with word embeddings (Socher et al., 2013; Kim, 2014) and/or character embeddings (Zhang et al., 2015; Conneau et al., 2016) to train sentence encoders, and then perform classification. Those neural networks use either convolutional layers or recurrent layers to learn deep representations, and often produce better results than older systems. However, one drawback of most of the models is that they do not make use of context, but focus on representing each sentence independently (Dernoncourt et al., 2016). Moreover, the sequence of labels are not directly modeled like in CRF, although RNNs can capture that indirectly.

Dernoncourt et al. (2016) tries to combine the properties of RNN and CRF. They use character and token embeddings to train a sentence encoder, and use an output sequence optimization layer to incorporate transition probabilities of labels. However, their data is from medical paper abstracts, which have a strong tendency to follow certain styles of writing, and are much shorter than dialogues. They only have 5 classes, all with very distinctive semantic properties. Our 10 labels demand deeper understanding to be distinguished.

In fact, the same technique has been used in name entity

recognition and other token-based sequence tagging tasks (Huang et al., 2015; Dernoncourt et al., 2017). The sequence optimization layer is also called a CRF layer. Sentences are much longer than name entities and contain more complex meanings, and thus sequential sentence labeling is a more difficult task.

3. Dataset

The dataset consists of dialogues of middle school students solving a robotics challenge together. There are two collaborative teams. Team A has 1 boy and 2 girls, and Team B has 2 boys. All students wore wireless microphones, and their interactions were videotaped. In this study, we obtained the transcribed dialogues and use text data for analysis.

In order to use RNNs to process sequences of sentences, we need to train a sentence encoder. Given the small dataset, we also tried external resources. Conneau et al. (2017) proposed a model to produce "universal sentence representations". They train sentence encoders on Natural Language Inference (NLI) task and claim such a process involves high-level understanding of text, and thus generates better sentence embeddings than other supervised or unsupervised learning methods. In our experiments, we use their pre-trained sentence encoder to represent sentences. The encoder is trained on SNLI (Bowman et al., 2015) and MultiNLI (Williams et al., 2017).

3.1. Computational Thinking Coding Scheme

The dialogue of each team is segmented into 20-minute conversation chunks. Eventually we have 10 chunks for Team A and 9 for Team B. Then the dialogue turns of the 19 files are fully annotated. We developed a computational thinking coding scheme, as shown in Table 1.

Code	Category	Description
A	Analysis	Planning, developing ideas
ATO	Algorithmic thinking -operation	Discussing programming functions
ATV	Algorithmic thinking -variable	Discussing quantities to use
D	Design	Building additions to the robot
DO	Debugging observations	Describing movement of robot
NSTO	Non-specific test outcome	
QB	Query building	Questions about using the Legos
QS	Query software	Ques. about software capabilities
QR	Query robot	Ques. about operations of robot
O	Other	Anything else

Table 1: Computational thinking coding scheme.

Every dialogue turn is labeled with one of the codes in Table 1. A dialogue turn is an utterance generated by a speaker. Occasionally a turn is split and assigned with two different labels, when the speaker expresses two ideas. In this case we describe it as two dialogue turns from the same speaker. In total we annotated 5723 dialogue turns from the 19 files. Utterances from individuals who are not the team members are excluded from analysis.

The code "QB" does not exist in any data we use, so it is ignored. The remaining classes are heavily imbalanced. For example, 58% of the dialogue turns have "O" label but only 0.6% have the "QS" label. Some techniques are commonly used to deal with imbalanced data, such downsampling, upsampling, setting weights etc., driven by specific purposes. In our experiment, we do not employ any of the techniques, but train our models on raw data.

3.2. 5-folds cross validation

Since the dataset is relatively small and the classes are imbalanced, we use 5-fold cross validation to evaluate the final results. In each iteration of training/testing, 15 out of 19 files are chosen as training set, and the remaining 4 files as test set. In the five iterations, the test sets are all different without overlapping, except that one file is used twice (because $4 \times 5 = 20$).

One may argue the files reflect different stages of the robotics challenge, and are not independent. However, we investigate the data in details and find local context much more relevant than long-distance context, so treating the files independently should not create significant errors.

Hyper-parameters are tuned on one split of data only. The 15 training files are divided as training and validation set by 13:2.

4. Experiments

We built two models. One is a conventional CRF model, and the other is an RNN with a CRF layer.

4.1. Linear Chain CRF model

Until very recently, linear chain CRFs have been the preferred choice for sequence labelling tasks in NLP. We ran this model as a baseline to compare to. This model estimates the conditional probability of a label sequence, assuming that a label at a certain time step depends on its preceding label as well as input data. The objective is to find the feature weights that maximize the conditional probability as shown in equation 1.

$$p(\mathbf{y}|\mathbf{x}) \propto \exp\left(\sum_{t=1}^{T}\left(\sum_{i=1}^{D} w_i f_i(y_t, y_{t-1}, X)\right)\right) \quad (1)$$

In our case, every time step t is corresponding to a dialogue turn with a label. The input X can be the whole series of input data although in practice we only use a portion of the whole input at each time step. Naturally, we can use the text of each dialogue turn, and possibly of nearby turns. Therefore we rewrite the feature function as $f_i(y_t, y_{t-1}, x_t)$.

Here f_i uses the previous label y_{t-1}, current label y_t and current input text representation x_t as input. D is the dimensionality of the feature space. w_i is the set of parameters to be trained. In order to make this model a baseline for direct comparison, we used a pre-trained model to generate sentence embeddings, folllowing the work of Conneau et al. (2017). Each sentence is represented as a 4096-dimensional vector. In addition to that, we also add another feature to indicate if the speaker has changed from the previous time step. As a result, the feature space has 4097 components. In our dataset, two consecutive turns are from different speakers in the vast majority of cases, but occasionally an utterance from the same speaker are split, usually due to a change of idea. Our implementation used the Python wrapper for the CRFSuite library (Okazaki, 2007).

The features used in the linear chain CRF model do not create an adequate context representation. Adding features from neighboring sentences to the model can potentially improve it, however, it increases feature dimensionality rapidly, making it difficult to capture long-distance dependencies. Recurrent neural networks models provide a more elegant solution to this issue.

4.2. LSTM Model with CRF Layer

LSTM uses gated cells to preserve memory through multiple time steps. Therefore it is possible to incorporate contextual information. Neural networks also tend to process high-dimensional representations better than some of the more conventional statistical machine learning methods.

We add a CRF output layer to the model, which allows us to jointly model the output labels. The CRF layer essentially maximizes the label sequence score s, computed as follows:

$$s(y_{1:T}) = \sum_{t=2}^{T} \mathbf{A}[y_{t-1}, y_t] + \sum_{t=1}^{T} \mathbf{a}[y_t] \qquad (2)$$

where $\mathbf{A}$ is the matrix of transition probabilities, and $\mathbf{a}$ represents the individual probability of a label. Dialogue turn representations are fed into a bidirectional LSTM (Bi-LSTM) layer. The output of the Bi-LSTM layer is concatenated with the speaker identity input, and then followed by two fully connected hidden layers.

The full model architecture is shown in Figure 1. The bottom row shows the sequence of input utterances, with sentence embeddings and speaker identities. 1 means the speaker is the same as the previous one, and 0 otherwise. Backward and forward LSTMs are shown in blue, followed by the concatenation layer in green. Yellow blocks are hidden layers. The top row is the CRF layer. In this example three output labels are shown (A = "Analysis", DO = "Debugging operation", ATV = "Algorithmic thinking / Variable"). Rightward arrows indicate that each label depends on its preceding label.

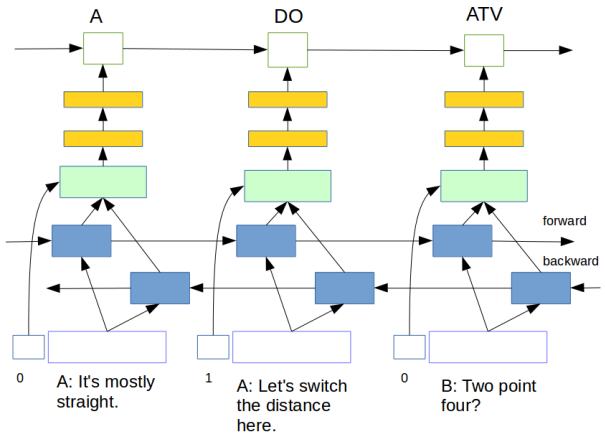


Figure 1: Bi-LSTM-CRF model.

We experimented different ways to obtain sentence encoders. (1) We use the same sentence encoder as in 4.1., so the sentence vectors fed into the Bi-LSTM layer are the same as before. (2) We train our own sentence encoder. For each sentence, pre-trained word embeddings are fed into a Bi-LSTM layer, followed by a Maxpooling layer. The encoder is co-trained with the classifier. (3) We combined the encoders from (1) and (2), concatenating the outputs of the two. In each case, we also supplement it with speaker identity.

Ideally the model should be time-variant and uses time step as variable too, because the time/stage of a dialogue may have an impact on the labels. However, currently we only have the full dialogues of two teams and there is no sufficient data to train a time-variant model.

4.3. Hyper-parameters

For the Linear chain CRF model, we set $c1 = 1.0$, $c2 = 10^{-3}$. These are L1 and L2 regularizations. We tried different number of $max_iterations$ and found 200 to be desirable. All other hyper-parameters are just default values of the library.

For LSTM-CRF, our final model has 1024 LSTM units to process sequences of sentence embeddings, and the hidden layers have 1024 and 512 neurons, respectively. Input dropout for sentence embeddings is 0.4. The two hidden layers both have a dropout rate 0.5. When a sentence encoder is trained, 512 LSTM units are used. After Maxpooling, the results are concatenated with pre-trained sentence embeddings. The configuration of other parts of the neural network are intact.

There is no dropout for speaker identity input. All models are trained with 100 epochs. We found the results do not have noticeable changes when the epochs are set in the 80$\sim$200 range.

The glove.840B.300d word vectors[1] are used for word embeddings. It was trained on 840 billion tokens and represents tokens with 300d vectors.

4.4. Results

From 5-fold cross validation, the precision, recall and F1 scores of all 5 runs are calculated and averaged. Within each run, the scores are the prevalence-weighted macro-averages across classes. In other words, the scores are averaged over all classes and weighted by their frequencies. Using this method, recall will be the same as accuracy. It should also be noted that F1 score is not necessarily between precision and recall in this method. We calculated the standard deviation of F1 scores to check the stability of the results.

We list two naïve baselines to compare to. The majority baseline assumes a model assigns the majority class "O" to all occurrences. From the 5-fold CV, there are 5556 predicted labels and 3303 of them have "O" as the true label. Therefore the majority baseline is 3303/5556=0.594. The random baseline assumes a model assigns labels based on their statistical distributions in training data. Then the expected accuracy will be $\sum_i p_i^2$ where pi is the probability of class i. In our case the value is 0.384.

All the results are shown in Table 2. Our models outperform the two naïve baselines by a large margin. Judging from F1 score, the LSTM-CRF model with pre-trained sentence encoder has the best performance, although the recall

[1] https://nlp.stanford.edu/projects/glove/

Model	Prec	Rec/Acc	F1	F1 std
Majority	-	.594	-	
Random	-	.384	-	
CRF	.653	**.678**	.644	.0319
CRF+neighbor	.640	.675	.636	.0393
LSTM-CRF, pretrained	.660	.677	**.661**	.0241
LSTM-CRF, co-trained	.636	.654	.626	.0436
LSTM-CRF, combined	**.666**	.666	.654	**.0235**

Table 2: Evaluation results. "Majority" means assigning the majority label ("O" in our case) to all dialogue turns. "Random" means randomly assigning labels based on their statistical distributions. Precision, recall, and F1 scores are the averages of the 5-fold cross validation evaluation, respectively. "F1 std" is the standard deviation of the 5 F1 scores.

score is no better than that of the CRF model. LSTM-CRF with co-trained sentence encoder does not perform very well, probably because the training set is too small. Combining the two encoders does not boost results. It may suggest that the pre-trained sentence encoder is overall much better.

"CRF+neighbor" refers to the CRF model with neighboring sentence embeddings as input, in addition to current sentence. Such a technique incorporates more context in the model, but actually the performance is lower. Probably the data dimensionality has become too high for a CRF model to work properly.

Taking a close look, we find the CRF models tend to ignore low-frequency classes. For example, the smallest class QR ("query robot") occurs 37 times in all the 5556 test instances. The CRF model does not capture any of them (0 recall), but the LSTM-CRF model with pre-trained sentence embeddings manages to identify 1 of them correctly. The second smallest class QS ("query software") occurs 116 times. The CRF model only identifies 1, but the LSTM-CRF model finds 18 of them. Depending on the purpose, there can be more specific ways to evaluate the results, which will not be discussed in this paper.

5. Conclusion

In order to automatically label dialogue turns with computational thinking codes, we experimented with two computational approaches: (1) a linear chain Conditional Random Field (CRF) model, and (2) a recurrent neural network model with a CRF layer. Both of them beat naïve baselines by a large margin. Although our dataset is relatively small, the RNN-based model seems to outperform the conventional CRF model.

Our research is also a stepping-stone for microgenetic analysis of observational data. Eventually the analysis will help us understand collaborative learning and working in a team setup.

6. Bibliographical References

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Conneau, A., Schwenk, H., Barrault, L., and LeCun, Y. (2016). Very deep convolutional networks for natural language processing. *CoRR*, abs/1606.01781.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.

Dernoncourt, F., Lee, J. Y., and Szolovits, P. (2016). Neural networks for joint sentence classification in medical paper abstracts. *CoRR*, abs/1612.05251.

Dernoncourt, F., Lee, J. Y., Uzuner, Ã., and Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. *JAMIA*, 24(3):596–606.

Hassanzadeh, H., Groza, T., and Hunter, J. (2014). Identifying scientific artefacts in biomedical literature: The evidence based medicine use case. *Journal of Biomedical Informatics*, 49:159–170.

Hirohata, K., Okazaki, N., Ananiadou, S., and Ishizuka, M. (2008). Identifying sections in scientific abstracts using conditional random fields. In *In Proc. of the IJCNLP 2008*.

Huang, K.-C., Chiang, I.-J., Xiao, F., Liao, C.-C., Liu, C. C.-H., and Wong, J.-M. (2013). Pico element detection in medical text without metadata: Are first sentences enough? *Journal of Biomedical Informatics*, 46(5):940–946.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Kim, S. N., Cavedon, L., and Baldwin, T. (2010). Classifying dialogue acts in one-on-one live chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 862–871, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Lin, J., Karakos, D., Demner-Fushman, D., and Khudanpur, S. (2006). Generative content models for structural analysis of medical abstracts. In *Proceedings of the Workshop on Linking Natural Language Processing and Biology: Towards Deeper Biological Literature Analysis*, BioNLP '06, pages 65–72, Stroudsburg, PA, USA. Association for Computational Linguistics.

Margolis, A., Livescu, K., and Ostendorf, M. (2010). Domain adaptation with unlabeled data for dialog act tagging. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, DANLP 2010, pages 45–52, Stroudsburg, PA, USA. Association for Computational Linguistics.

McKnight, L. and Srinivasan, P. (2003). Categorization of sentence types in medical abstracts. In *AMIA*. AMIA.

Moldovan, C., Rus, V., and Graesser, A. C. (). Automated speech act classification for online chat.

Okazaki, N. (2007). Crfsuite: a fast implementation of conditional random fields (crfs).

Samei, B., Li, H., Keshtkar, F., Rus, V., and Graesser, A. C. (2014). Context-based speech act classification in intelligent tutoring systems. In Stefan Trausan-Matu, et al., editors, *Intelligent Tutoring Systems*, volume 8474 of *Lecture Notes in Computer Science*, pages 236–241. Springer.

Siegler, R., (2006). *Microgenetic analysis of learning*, pages 464–510. John Wiley & Sons, Hoboken, NJ.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer.

Sullivan, F. R., Adrion, W. R., and Keith, P. K. (2015). Microgenetic learning analytics: A computational approach to research on student learning. In *Paper presentation at the annual meeting of the American Educational Research Association*, Chicago, IL, April.

Tavafi, M., Mehdad, Y., Joty, S., Carenini, G., and Ng, R. (2013). Dialogue act recognition in synchronous and asynchronous conversations. In *Proceedings of the SIGDIAL 2013 Conference*, page 117–121, Metz, France, August. Association for Computational Linguistics, Association for Computational Linguistics.

Wertsch, J. (1991). *Voices of the mind: A sociocultural approach to mediated action*. Harvard University Press, Cambridge, MA.

Williams, A., Nangia, N., and Bowman, S. R. (2017). A broad-coverage challenge corpus for sentence understanding through inference. *CoRR*, abs/1704.05426.

Zhang, X., Zhao, J. J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In Corinna Cortes, et al., editors, *NIPS*, pages 649–657.

Increasing Argument Annotation Reproducibility by Using Inter-annotator Agreement to Improve Guidelines

Milagro Teruel[1], **Cristian Cardellino**[1], **Fernando Cardellino**[1], **Laura Alonso Alemany**[1], **Serena Villata**[2]
[1]Universidad Nacional de Córdoba, Argentina
[2]Université Côte d'Azur, CNRS, Inria, I3S, France
mteruel@unc.edu.ar, ccardellino@unc.edu.ar, fndcardellino@gmail.com, lauraalonsoalemany@unc.edu.ar, serena.villata@inria.fr

Abstract

In this abstract we present a methodology to improve Argument annotation guidelines by exploiting inter-annotator agreement measures. After a first stage of the annotation effort, we have detected problematic issues via an analysis of inter-annotator agreement. We have detected ill-defined concepts, which we have addressed by redefining high-level annotation goals. For other concepts, that are well-delimited but complex, the annotation protocol has been extended and detailed. Moreover, as can be expected, we show that distinctions where human annotators have less agreement are also those where automatic analyzers perform worse. Thus, the reproducibility of results of Argument Mining systems can be addressed by improving inter-annotator agreement in the training material. Following this methodology, we are enhancing a corpus annotated with argumentation, available at `https://github.com/PLN-FaMAF/ArgumentMiningECHR` together with guidelines and analyses of agreement. These analyses can be used to filter performance figures of automated systems, with lower penalties for cases where human annotators agree less.

Keywords: Argument Mining, Annotation Guidelines, Reproducibility

1. Introduction and Motivation

Argument Mining tackles a very complex phenomenon, involving several levels of human communication and cognition. Due to this complexity, data-driven approaches require a huge amount of data to properly characterize the phenomena and find patterns that can be exploited by an automatic analyzer. However, only small annotated corpora are available, and moreover they cannot be used in combination because they are based on different theoretical frameworks or cover different genres.

In this abstract we present work in progress in building a corpus annotated with arguments. As inherent part of this work, we are applying a methodology for early detection of ill-defined annotation concepts. We detect those by inspecting annotated texts for sources of disagreement between annotators, and redefining the annotation scheme so that these disagreements are minimized. This has been a successful approach to improve guidelines, as in (Hovy et al., 2006).

In preliminary explorations, we have found that agreement-driven modifications in the annotation scheme improve an automatic analyzer. Our final objective is to find an annotation scheme that is a tradeoff between theoretically based concepts, application needs, stability of human annotation and performance of automatic analyzers.

2. Annotated Corpus

2.1. ECHR Judgments

Four human annotators have annotated 7 judgments from the European Court of Human Rights (ECHR) in English, obtained from the Court website[1], totaling 28,000 words. Approximately half of the words were annotated as belonging to an argument component, as can be seen in Figure 1.

One of the judgments was annotated by all 4 annotators and discussed collectively as training. In this annotation, agreement between judges was never lower than $\kappa = .54$. Then, two pairs of judges annotated two judgments independently, we analyze agreement measures on those two pairs. More annotation pairs are currently being annotated and will be updated in the repository of the project (`https://github.com/PLN-FaMAF/ArgumentMiningECHR`).

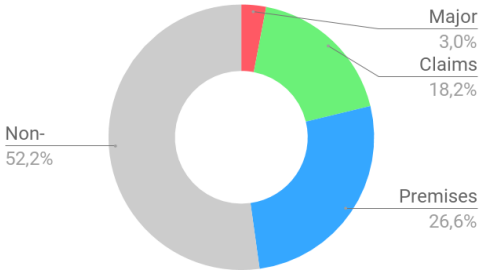


Figure 1: Proportion of component labels in the corpus.

2.2. Annotation Objectives

The objective of our annotation is to identify arguments composed by claims and premises (justifications) that are related to each other. Our initial annotation scheme was loosely based on (Toulmin, 2003), following the main adaptations that (Stab and Gurevych, 2015) propose to take the concepts from a theoretical model to practical annotation guidelines. Argument components were classified as:

Major Claim : a general statement expressing the author's stance with respect to the topic under discussion.

Claim : a controversial statement whose acceptance depends on premises that support or attack it. Claims are the central components of an argument and they either support or attack the major claim. We associate each claim with the actor that has issued it.

Premise : reasons given by the author for supporting or attacking the claims. They are not controversial but

[1]`hudoc.echr.coe.int`

factual. Specifically for this corpus, we distinguish the subclasses: Facts, Principles of Law and Case-law.

Argument components are connected to each other by relations, mainly **support** or **attack** relations (Simari and Rahwan, 2009). Claims support or attack other claims or a major claim, premises may support or attack claims or other premises. Additionally, we have established two more minor relations, specific for this corpus: **duplicate** (holding between claims or premises) and **citation** (holding between premises, when one cites a reference Case-law).
We have used brat (Stenetorp et al., 2012) as a tool for annotation.

3. Inter-annotator disagreement as an opportunity for improvement

3.1. Measuring disagreement

Disagreement between annotators is typically part of the annotation process in a qualitative way. Usually, annotation guidelines are iteratively refined in a long process where annotators discuss conflictive examples to specify vague concepts and establish annotation protocols.
Argument analysis is a highly subjective task, with typically low levels of inter-annotator agreement. Low inter-annotator agreement results in low reproducibility and also in poorer performance of automatic analyzers that are trained with these resources. We address reproducibility (and consequently automatic performance) by applying standard inter-annotator agreement measures to the annotated corpus from a very early stage, to make high-level decisions on the annotation scheme, instead of minute protocols (e.g., (Habernal, 2014)) to try to delimit concepts that have very high subject variation to begin with. We take low inter-annotator agreement as a symptom of ill-defined or far-fetched concepts, beyond the scope of our current annotation effort.
In order to assess the reproducibility of human annotations, we used Cohen's kappa (Cohen, 1960). This coefficient is a standard to measure inter-annotator agreement. It reports agreement between pairs of annotators, factoring out the probability that annotators would have agreed by chance. Other measures of inter-annotator agreement, like Krippendorf's alpha (Krippendorff, 1980) or Fleiss kappa (Fleiss and Cohen, 1973) will be included when we have a more extensive annotated corpus.

3.2. Annotators agree on what is argumentative

First of all, we found **high** agreement between annotators to determine whether a sentence contained an argument component, with Cohen's kappa ranging between $\kappa = .77$ and $\kappa = .84$. When this agreement is considered at token level, it varies between $\kappa = .59$ and $\kappa = .84$. We note that most disagreements occur between annotators that annotate less or more proportion of words as argumentative. Indeed, some annotators tend to consider more spans of text as argument components than others. However, there is a high agreement on spans identified as argumentative by annotators that consider less spans of text as argumentative. This has been addressed in the second version of the guidelines with a more detailed definition of argumentative text.

3.3. Major claims are major disagreements

For the classification of argument components as premises, claims or major claims, we found lower agreement, ranging from $\kappa = .48$ to $\kappa = .56$. Looking at the confusion matrices of annotations of pairs of annotators, in Figure 2, we find that there are important disagreements between all of the categories. However, the category of *major claim* seems to be the most conflictive: in one of the pairs, annotators did not have any overlap, in the other, they had more proportion of disagreement than of agreement. Therefore, this category seems to be ill-defined. Spans that are classified as major claims by one annotator tend to be classified as claims by the other, so we decided to collapse those two categories. When we do that, we obtain better agreement, as can be seen in Figure 3. We could think that this improvement is due to a smaller number of categories. However, the kappa coefficient, which factors out the number of categories, also improves: when those two categories are collapsed, then the agreement increases from $\kappa = .48$ to $\kappa = .51$ and from $\kappa = .56$ to $\kappa = .64$.

3.4. What the Court says are premises or claims

To analyze disagreements between premises and claims, we carried out a detailed analysis by subclasses, displayed in Figure 4. We found that claims issued by the ECHR are a major source of disagreement, because the concept is mixed with that of fact or principle of law. This can be expected, as claims by a Court in a judgment are performative speech acts that have the status of principles of law after the judgment is issued, and principles of law have the same status as facts in a reasoning by a court. However, epistemologically these three concepts are difficult to reconcile. To a minor extent, claims issued by the government tend to be mixed with premises labeled as facts, probably also because the legal status of the government, which can be easily assimilated to an actor capable of doing performative speech acts. Moreover, the category of premise as fact also accumulates a high number of disagreements with the category of non-argumentative text.
However, not considering fact premises as part of the annotation comes unnatural and is inadequate from a point of view of argumentation theory and of application utility. Therefore, in the second version of the guidelines, we have addressed this problem by refining the protocol, describing this phenomenon at length, and determining that the claims issued by the Court are to be taken as claims in the judgment where they are issued and as Principles of Law or Case Law if they are cited from another case.
In general, there is some confusion between premises interpreted as facts or as case-law, and also between premises considered case-law or law principles. However, these confusions can be easily addressed by a formal delimitation of case-law using shallow textual cues, also refining annotation guidelines.

3.5. Annotators seem to agree on relations

To assess the level of agreement for relations, we looked into relations that held between argument components where two annotators agreed. That meant between 46% and 74% of the components. For those, annotators agreed

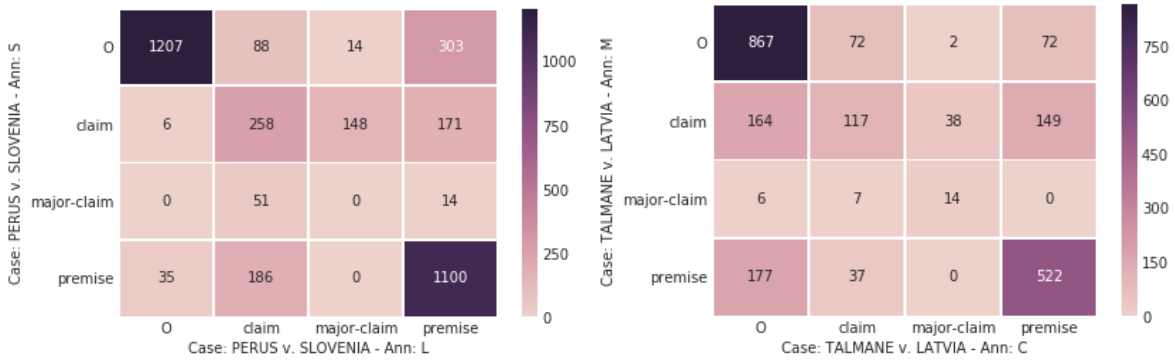


Figure 2: Confusion matrices for annotations of components between pairs of annotators, distinguishing major claims, claims and premises. Agreement for the matrix on the left is $\kappa = .56$ and on the right $\kappa = .48$.

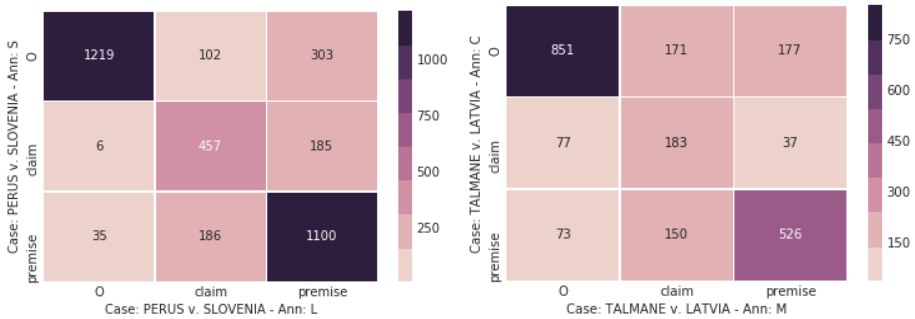


Figure 3: Confusion matrices for annotations of components between pairs of annotators, distinguishing only claims and premises. Agreement for the matrix on the left is $\kappa = .64$ and on the right $\kappa = .51$.

on the existence of a relation between components only in between 10% and 19% of the cases. When they agreed that a relation held between a given pair of components, annotators tended to agree on whether the relation was of attack, support or citation, with agreement ranging from 85% to 100% in most cases. However, the number of cases where such analysis could be carried out is so small that we require a bigger corpus to obtain more significant figures and draw conclusions upon them.

4. Automatic classification fails where humans disagree

In this section we show the relation between inter-annotator agreement and the performance of an automated classifier. To do that, we rely on the Argument classifier developed by (Eger et al., 2017), a neural end-to-end argumentation mining system with a multi-task learning setup. This system has been trained with part of the corpus, then annotated a different part of the corpus and its predictions compared with human annotations.

The comparison of human and automatic annotations is shown in Figure 5, with results showing the predictions of the classifier trained with major claims and not trained with major claims. We find that indeed major claims cannot be recognized by the classifier. This can be explained by the low proportion of major claims in the annotated corpus (see Figure 1), but neural classifiers tend to overfit the data and it could be expected that some major claims would have been identified. We also see that the confusion between premises and non-argumentative text is higher than the confusion between claims and non-argumentative text, and the confusion between premises and non-argumentative text is also higher than the confusion between claims and non-argumentative text. In consequence, there seems to be a strong relation between disagreements between humans and misperformance of automatic analyzers. Addressing the first will probably have a positive impact on the second.

5. Conclusions and Future Directions

We have presented an annotation effort for argumentation exploiting inter-annotator agreement as an indicator of ill-delimited concepts. Our aim is to enhance the reproducibility of argument annotation. Finding sources of disagreement across categories has allowed to make high-level decisions concerning the objectives of annotation. We have dropped the category of *major claim*, a major cause of disagreement that was not central to descriptive adequacy or application needs. We have described with more insight the case when the Court issues a claim that can be later considered as law, providing more insight and an unambiguous protocol of when these statements must be considered claims (when they are in the same judgments) or when they must be considered premises (when they were issued in previous judgments).

We have carried out preliminary experiments with a classifier, showing that automatic analyzers tend to fail where human annotators disagree.

In future work we will analyze the performance of the

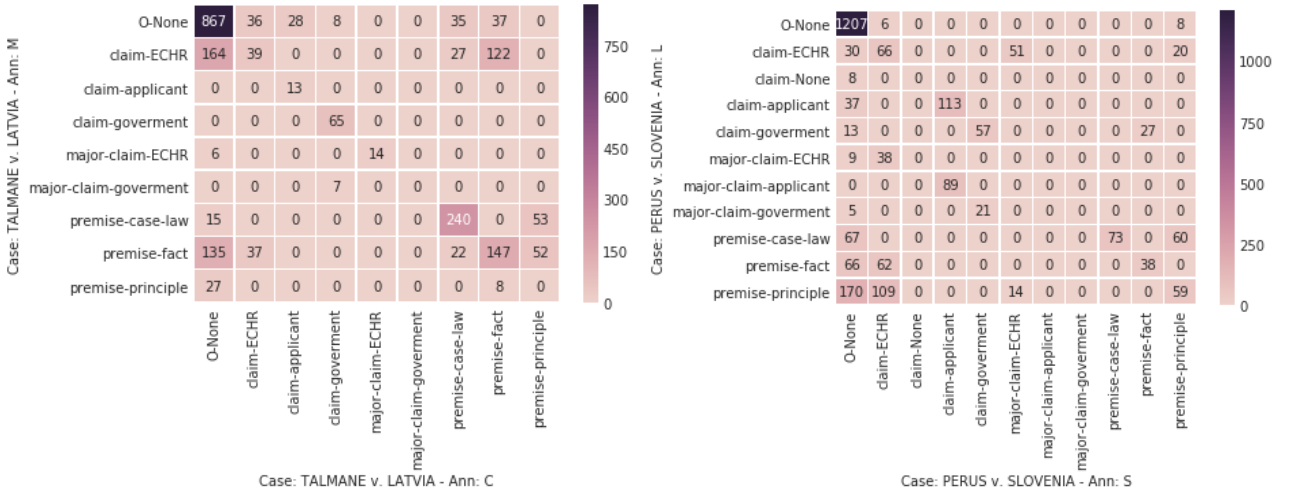


Figure 4: Confusion matrices for annotations of components between pairs of annotators, distinguishing their attributes. Agreement for the matrix on the left is $\kappa = .45$ and on the right $\kappa = .33$.

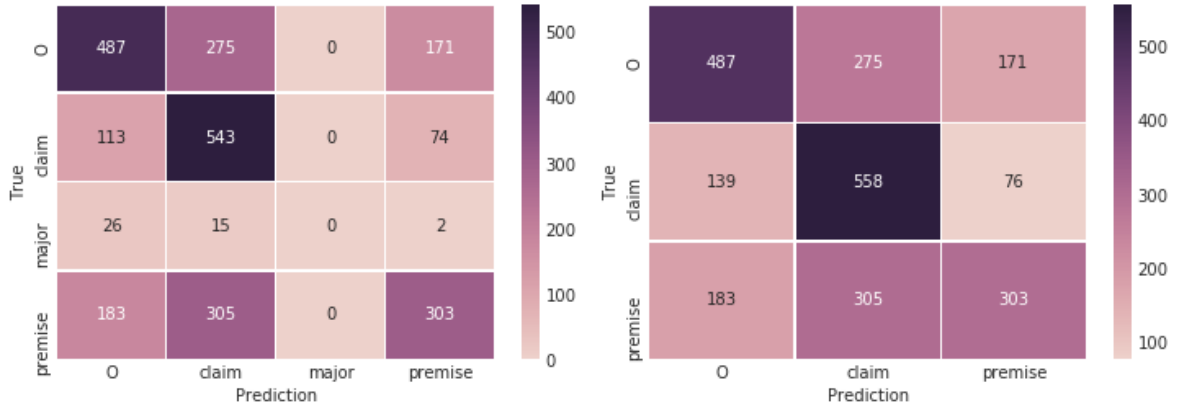


Figure 5: Confusion matrices for annotations of components between an automatic classifier and the human gold standard, distinguishing major claims (left) and not distinguishing them (right).

classifier with different configurations of the annotated resource: removing cases with low inter-annotator agreement and collapsing categories with high confusion between annotators. We will also be exploring inter-annotator agreement in relations.

We are currently annotating more documents with the second version of the guidelines. The newly annotated documents and the updated guidelines will be made available at the resource repository `https://github.com/PLN-FaMAF/ArgumentMiningECHR`.

6. References

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational & Psycological Measure*, 20:37–46.

Eger, S., Daxenberger, J., and Gurevych, I. (2017). Neural end-to-end learning for computational argumentation mining. *CoRR*, abs/1704.06104.

Fleiss, J. L. and Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619.

Habernal, I., (2014). *Argumentation in User-Generated Content: Annotation Guidelines*. UKP Lab, Computer Science Department, Technische Universität Darmstadt, April.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). Ontonotes: The 90 In *NAACL 2006*, pages 57–60, Stroudsburg, PA, USA. Association for Computational Linguistics.

Krippendorff, K. (1980). *Content analysis: an introduction*. Sage, Beverly Hills, California.

Guillermo Ricardo Simari et al., editors. (2009). *Argumentation in Artificial Intelligence*. Springer.

Stab, C. and Gurevych, I., (2015). *Guidelines for Annotating Argumentation Structures in Persuasive Essays*. UKP Lab, Computer Science Department, Technische Universität Darmstadt.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: A web-based tool for nlp-assisted text annotation. In *EACL 2012*, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

Toulmin, S. E. (2003). *The Uses of Argument*. Cambridge University Press, July.

Semi-Supervised Clustering for Short Answer Scoring

Andrea Horbach and Manfred Pinkal
University of Duisburg-Essen & Saarland University, Germany
andrea.horbach@uni-due.de, pinkal@coli.uni-saarland.de

Abstract

This paper investigates the use of semi-supervised clustering for Short Answer Scoring (SAS). In SAS, clustering techniques are an attractive alternative to classification because they provide structured groups of answers in addition to a score. Previous approaches use unsupervised clustering and have teachers label some items after clustering. We propose to re-allocate some of the human annotation effort to before and during the clustering process for (i) feature selection, (ii) for creating pairwise constraints and (iii) for metric learning. Our methods improve clustering performance substantially from 0.504 kappa for unsupervised clustering to 0.566.

Keywords: Short-Answer Scoring, Clustering, Computer-Assisted Language Learning

Prompt (ASAP dataset): *During the story, the reader gets background information about Mr. Leonard. Explain the effect that background information has on Paul. Support your response with details from the story.*
Answer (2 points): *Paul sees himself in Mr. Leonard. They both can't read but both are good at track.*
Answer (0 points): *He (the narrator) finds out that Mr. Leonard was once a star athelete at his college but dropped out because of grades.*

Prompt (PG dataset): *What did the Declaration of Independence do?*
Answer (correct): *declare independance from britain*
Answer (incorrect): *protect the constitution and people*

Figure 1: Example prompts and answers

1. Introduction

In an educational context, short answer questions are a type of exercise asking for short free-form answers of a few words or sentences in response to a given prompt. They occur, for example, in the form of science questions or reading comprehension exercises and can address both native speakers, such as high school students, and foreign language learners. Automatic Short Answer Scoring (SAS) is the task of automatically assigning a score to such answers, either in numeric form (points) or by a binary true/false label. Figure 1. shows two example prompts with some scored answers.

Automatic scoring of short answer exercises is a challenge: in contrast to multiple choice or simple gap-filling exercises, SAS has to assess the semantic correctness of answers, and it is thus related to fields such as natural language understanding, paraphrase detection or textual entailment. In addition, it has to deal with noisy user input containing spelling and grammar errors.

Most approaches to SAS consider automatic scoring as a classification task, relying on supervised machine learning (ML) techniques which require manually labeledtraining data In contrast Basu et al. (2013),Brooks et al. (2014) and Horbach et al. (2014) have focused on the use of clustering techniques for SAS. The rationale behind this procedure is that answers that are similar to each other – and therefore end up in the same cluster – are also likely to receive the same label for scoring and can thus be scored in one grading step by a teacher. Ideally, a teacher has to label only a single answer representative for the whole cluster which is then *propagated* to all members of the cluster.

An added value of clustering is that it provides valuable structural information, while ML classifiers just assign a score (Brooks et al., 2014). As an example, automatic clustering of the answers for the question from the PG dataset, "*What is one right or freedom from the First Amendment of the U.S. Constitution?*" yields different groups of correct and incorrect answers, such as, {*freedom of speech, the right of free speech, to have freedom of speech, ...*} or {*freedom of religion, freedom to practise religion, the freedom of religion, ...*}, {*to bear arms, the right to bare arms, right to arms, ...*}. The first two clusters contain correct answers referring to different facts, the last one contains answers making the same error. Teachers may use the output clusters to identify common misconceptions among students and assign feedback to whole groups of answers.

Some amount of human scoring is required for both supervised ML and clustering: annotation of training data in the one case and annotation of representative cluster members as a basis for propagation in the other case. Zesch et al. (2015) compared the performance of clustering with that of ML methods, keeping the number of manually labeled items constant. They carried out their study on the PG dataset and in addition on the ASAP dataset (see below, Section 3.). They reported that clustering proved beneficial only on the short and simple answers (a few words) of the PG dataset. On the ASAP dataset with more complex, longer answers, clustering falls far behind ML methods in their experiments.

In this paper, we show that *semi-supervised clustering* can substantially improve clustering results. While existing clustering approaches use manually labeled data only for post clustering label propagation, we distribute human effort and use human-labeled data in multiple ways before, during and after clustering:

- **Feature selection:** We use labeled items for feature selection *before* the actual clustering, as lustering algorithms are known to suffer more from noisy features than supervised learning algorithms that can select the features relevant for a task (Alelyani et al., 2013).

- **Clustering with constraints:** We employ two methods of using labeled instances as *seeds during* the clustering: (i) for guiding the clustering process through relational constraints, that indicate whether two instances cannot or must belong to the same cluster, and (ii) for metric learning, i.e. adapting the distance metric according to those constraints. We reuse the items labelled for feature selection as seeds, so the second step does not require additional annotation effort.

- **Label propagation**: We use label propagation *after* clustering to assign a label to each cluster based on the teacher-assigned label of just one item of the cluster. Following Horbach et al. (2014), we annotate the the item closest to the centroid and propagate its label to all cluster members, as this procedure selects prototypical instances and is superior to propagating a random label.

To the best of our knowledge, we are the first to use semi-supervised clustering for SAS. We show that the multi-purpose usage of labeled instances can overcome half of the the gap between clustering and supervised ML methods, also for the complex ASAP dataset. We conclude that clustering with label propagation is an alternative to supervised ML methods, since it has the advantage of providing teachers with structured sets of answers.

After this introduction, Section 2. report about related work. We then present in Section 3. the used data and feature set. We describe our experiments in Section 5. and conclude with Section 6..

2. Related Work

A variety of approaches for automatic SAS have been proposed (see Burrows et al. (2014) for an overview), ranging from rule-based systems, (Sukkarieh and Pulman, 2005), to ML systems using various kinds of automatically extracted features. ML-based approaches can be generally classified in two ways: those that assess an answer based on the similarity with some sort of teacher-specified target answer and those where a target answer is not used or not even available. Approaches of the first type create feature vectors for each answer that express their similarity to the target answer based on semantic similarity or lexical overlap. The resulting classifier is not restricted to answers for just one question, but generalizes to different prompts (Mohler et al., 2011; Meurers et al., 2011). Approaches that do not compare to a target answer build one classifier per prompt using features based on the content of individual answers, such as lemma or character n-grams or dependency triples occurring in an answer. With our work, we follow the feature extraction of the second approach by creating feature vectors for clustering representing the content of each answer and clustering answers per prompt.

Our work is related to studies that address the number of answers needed as training data for SAS. Heilman and Madnani (2015) show that – as in many ML scenarios – the SAS task profits from larger training set sizes. Active learning (AL) is an ML technique that aims at selecting training instances in such a way that the classifier can learn most from them (Settles, 2010); AL methods have been used successfully in a variety of NLP tasks. In the SAS context, Horbach and Palmer (2016) explore active learning methods to select ML training instances and find that uncertainty sampling methods have an advantage over random sampling of training instances.

Within the field of clustering for SAS, the most prominent contribution is the Powergrading (PG) study by Basu et al. (2013) who use k-medoids and LDA clustering for answers to US citizenship exam questions. They learn a similarity metric between answers on a part of the data which contains gold standard information about semantic equivalence between answers. As features for this decision, they use various kinds of similarities between answers including both surface similarity on string and lemma level and wikipedia-based LSA similarity. They use a two-layered clustering technique that separates the answer space into clusters and subclusters.

Basu et al. (2013) propose a novel measure to evaluate their clustering approach: they count the number of "actions" a teacher has to take in order to label a complete (sub)-cluster with one label for all cluster members or to label individual answers until all answers in a clustering are correctly graded. This approach is not directly comparable to label propagation. Also, it does not provide a fully realistic measure of the teacher's effort because teachers do not know in advance which individual answers within a cluster are incorrectly labeled and therefore need to inspect all answers in a cluster in order to identify those which they need to re-label. Their evaluation is more comparable to an oracle variant of label propagation where we propagate the *majority label* of a cluster instead of the one closest to the centroid: when assigning a label to a complete cluster in the style of Basu et al. (2013), a teacher will select the label that fits the majority of the data. We also calculate the results of majority-based propagation as an oracle condition, but retain centroid-based propagation as the relevant evaluation method.

3. Datasets

We run experiments on two datasets:

ASAP: We use the 10 individual prompts from the ASAP 2 dataset[1] from the Kaggle automatic scoring competition. Each prompt contains around 1800 items with humanly annotated scores from 0 to 2 or 3 (depending on the dataset) in steps of 1.0. Answers are between one and a few sentences in length. The average number of tokens per prompt varies between 26 and 66.

PG: The PG dataset consists of 10 annotated answer sets (out of a total of 20 prompts) from immigration tests that tend to be very short (average number of tokens per prompt between 1.3 and 5) and therefore also much more repetitive. While the ASAP dataset does not contain duplicates, they occur very frequently in PG. Answers are annotated as either correct or incorrect.

[1] https://www.kaggle.com/c/asap-sas

4. Method

4.1. Features and Feature Selection

In congruence with previous work, we concentrate mainly on lexical features as they are highly predictive for this task. In the Kaggle competition for the ASAP dataset, the top 5 best-performing systems used mainly lexical features for scoring (Higgins et al., 2014) (the best-performing system (Tandalla, 2012) was indeed one that additionally used hand-crafted regular expressions for each prompt). For the more complex answers of ASAP, which usually consist of complete sentences instead of short phrases, we use lemma and character n-grams, and dependency subtrees as features. For the PG dataset with very short phrasal answers, dependency parsing provided unsatisfying results, so we restricted ourselves to character and word n-gram features.

We use the labeled seeds to perform supervised feature selection, as clustering is particularly sensitive to noisy features (Alelyani et al., 2013).

We use Weka's information gain-based attribute selection and test different numbers of features including the full feature set. For most prompts, we reach optimal clustering results with either 200 or 100 features. We use the optimal size of feature sets per prompt in all experiments for both clustering and supervised ML. We tried other linear feature selection algorithms but found no significant differences in performance. We also explored subset evaluation as an alternative, using Weka's Cfs Subset evaluation and found it to be less suitable than Information Gain.

4.2. Semi-supervised Clustering

Clustering algorithms aim at grouping similar objects together, where similarity is measured by a distance metric. Standard clustering algorithms work completely unsupervised, only based on the distance metric. Semi-supervised clustering makes use of seed data gained through human annotation. Seed data can either be given in the form of labeled items expressing cluster membership, or as relational information stating that two items should or should not belong to the same cluster.

In our SAS scenario, we assume that there is a one-to-many rather than a one-to-one relation between scores and clusters. I.e. one score (out of the maximum of 4 different scores for the ASAP dataset) can contain answers that fall into different groups of semantically similar answers (see the example given in Section 1.). Especially for low-scoring answers there is certainly more than one way to "get it wrong", and thus we cluster into more clusters than there are labels. Answers with different scores should definitely go into different clusters, answers with the same score may or may not belong to the same cluster, dependent on their semantic relatedness. Therefore, we cannot use categorical seed information to estimate the number of clusters and to initialize seed clusters as it is done for instance by Basu et al. (2002), but have to use relational information. Since scoring of individual answers is a much more natural task for teachers than assessing the similarity between different answers, we derive the relational pairwise constraints required for semi-supervised clustering from individually labeled items, the seeds. More specifically, we create a *cannot link (CL) constraint* stating that two answers should not go into the same cluster for each pair of seeds with different scores. In general, semi-supervised clustering can also use *must link constraints* stating that two items belong two the same cluster. We cannot derive reliable *must link* information from answer scores, so we employ *cannot links* only.

Implementation Setup We use the Weka implementation (Hall et al., 2009) of the unsupervised k-means algorithm (KM) (Lloyd, 1982) as our baseline algorithm, as do Zesch et al. (2015): k-means minimizes an objective function that sums over the squared distances of each item to its cluster centroid. We use as distance metric Euclidian distance between feature vectors.

For semi-supervised clustering, we use extensions of k-means introduced in the *metric pairwise constrained k-means* (MPCKM) algorithm by Bilenko et al. (2004), who integrate the usage of pairwise constraints and metric learning into the k-means algorithm and provide an extension of the Weka API for that.[2] Constraints are integrated into the clustering in the form of penalties for constraint violations that are added to the objective function. Each constraint is associated with an importance weight.

Metric learning is done in the MPCKM algorithm after each k-means iteration by adjusting the weights of individual features in two ways: first, by moving existing clusters from the previous iteration further away from each other and second, by increasing the distance between items with violated CL constraints.

4.3. Label Propagation

For our experiments, we assume the following scenario: a teacher is given one item per cluster for scoring, and the score is propagated to all members of the cluster. Accordingly, we evaluate our experiments using *label propagation* following both Horbach et al. (2014) and Zesch et al. (2015).

We use **centroid propagation** as a realistic method, where the label for all answers in a cluster is based on just one labeled instance. We select an item for labelling which is prototypical for its cluster by selecting the one closest to the cluster centroid.

We consider **majority propagation** to provide an upper bound of performance that we could reach when labeling a cluster based on the label of one instance: the scoring of a cluster is optimal if the one labeled element belonged to the majority class for that cluster. This evaluation is an oracle condition that indicates the quality and potential of a given clustering, as there is no reliable way to automatically select such an element.

4.4. Treatment of Duplicate Items in Clustering

The PG dataset contains high numbers of duplicate answers; there are 2434 unique answer for a total of 6980 individual answers. Multiple annotation of duplicates does not add any information. Hence, we make sure that we never select duplicates when sampling answers for human annotation. However, the negative impact of performance is higher if we get a very frequent answer wrong compared

[2] http://www.cs.utexas.edu/users/ml/risc/

to erring on an answer that is only given by one student. Therefore we do not remove duplicates for cluster on all answers, such that multiple occurrences of an answer have more influence in the clustering process, as they have, e.g., a higher probability to be selected for centroid-based label propagation.

4.5. Baselines

We compare our clustering results to two baselines: *unsupervised k-means clustering* on the full feature set and *supervised ML*. To enable meaningful comparison between the methods, we keep the number of annotated instances n constant across all experimental conditions. Thus we create n clusters in unsupervised k-means clustering as all human annotation effort can be used to label cluster centroids, we create fewer clusters in the semi-supervised case, where some annotations are used for labeling seeds. Accordingly, the ML baseline, implemented by Weka's SMO algorithm (Hall et al., 2009), is trained on n labeled items, as done in Zesch et al. (2015).

In addition to their approach, we do not only calculate the baseline with the complete feature set. Supervised ML algorithms also profit from feature selection and it would be an unfair comparison to optimize features just for clustering. We also performed feature selection for ML and report results for the best configuration per prompt. We randomly sample the data for the classifier 100 times and report average results. For the optimized feature set we also report the best individual run as an upper bound.

5. Experiments

Our experiments address the question how a set of answers can be optimally graded with only a limited amount of available human annotation effort:

Experiment 1 compares variants of the k-means algorithm that correspond to different degrees of supervision to confirm the contribution of the individual components of the MPCKM algorithm. **Experiment 2** investigates the optimal tradeoff for distributing a given amount of human annotations between labeling seeds before clustering and labeling cluster centroids after clustering. In **Experiment 3** we cross-check that our semi-supervised results cannot be reached with approaches that use unsupervised feature selection. In **Experiment 4** we investigate how human annotation effort can be further minimized by reusing seeds for label propagation.

5.1. Experimental Setup

Dataset Sizes In order to evaluate always on the same number of answers per dataset, we use the first 1000 answers to each ASAP prompt and all 698 answers to each PG prompt.

Evaluation Metric We report Cohen's quadratically weighted kappa (Cohen, 1968) after label propagation. In our grading scenario, where we want to measure the quality of the resulting grading of a set of answers and compare to supervised classification methods. This type of evaluation is more meaningful than evaluation measures applied in other clustering tasks, such as the widely used bCubed metric (Amigó et al., 2009).

5.2. Experiment 1: Different Degrees of Supervision in Clustering

In our first experiment, we measure the influence of different levels of supervision. We go from unsupervised k-means clustering (KM_{all}), over k-means clustering that uses seeds only for feature selection (KM_{sel}) and semi-supervised clustering that additionally derives CL constraints from the seeds (*KMCL*), to the full *MPCKM* clustering algorithm with feature selection, CL constraints, and metric learning. We aim at investigating the effect of a fixed "small" number n of labelled data on clustering performance, which at the same time should be large enough to induce clusters of reasonable quality. We decided for n = 150 (out of a total of 1,000 answers per question) for ASAP, and n = 50 for PG (the comparably low number is due to the high amount of duplicates in the answers). This overall number of annotation steps is split into those answers that are used for both feature selection and constraints (the seeds), and those that are used to label the centroid of each cluster; i.e., the number of annotations for labeling clusters centroids determines the number of clusters created.

In this experiment, the split of the n labelled items between seeds and labelled cluster centroids is 110:40 for ASAP and 40:10 for PG. These proportions of seeds and cluster centroids are selected as the optimal ones, based on the results of Experiment 2 (see below).

Tables 1 and 2 show the results for the different k-means variants. In addition to centroid-based label propagation, we report majority propagation MPCKM$_{best}$ for the full MPCKM algorithm as an upper bound for clustering performance.

We can learn the following from the experiment: first, investing labeled items into feature selection pays off (KM$_{all}$ vs KM$_{sel}$) for the ASAP dataset. Second, we see that adding constraints alone gives us an additional small improvement (KMCL vs KM$_{sel}$), while adding metric learning (MPCKM) adds substantially to the performance. The improvement is consistent for centroid based label propagation and for the majority propagation upper bound MPCKM$_{best}$. Third, we see that the best clustering method comes closer to the ML baseline trained on the full dataset (ML_{all}. By making optimal use of the manually labelled data, we could thus close more then half of the gap between the performance of clustering and machine learning stated in Zesch et al. (2015).

For the PG dataset, basic clustering already outperforms ML methods, arguably because the very short answers of the PG dataset yield an already small feature set that contains little noise. For the following experiments, we therefore report results on the more challenging ASAP dataset only. Note that our scores for ASAP are not directly comparable to the scores of the top performing systems from the Kaggle competition, as the evaluation setup, especially the number of training and test instances used, is different.

5.3. Experiment 2: Finding a Tradeoff between the Numbers of Seeds and the Number of Clusters

In this experiment, we determine the optimal tradeoff between the number of seeds, which are used for feature se-

p.	KM_{all}	KM_{sel}	KMCL	MPCKM	*$MPCKM_{best}$*	ML_{all}	ML_{sel}	ML_{best}
	Clustering 40 cluster, 110 seeds					supervised ML 150 items		
1	0.462	0.541	0.547	0.593	*0.668*	0.651	0.673	*0.711*
2	0.432	0.47	0.469	0.496	*0.574*	0.571	0.571	*0.62*
3	0.343	0.378	0.377	0.379	*0.451*	0.384	0.384	*0.437*
4	0.543	0.547	0.549	0.581	*0.651*	0.639	0.655	*0.693*
5	0.617	0.622	0.631	0.69	*0.756*	0.68	0.72	*0.782*
6	0.682	0.646	0.64	0.74	*0.765*	0.692	0.745	*0.787*
7	0.352	0.398	0.402	0.447	*0.533*	0.565	0.565	*0.622*
8	0.448	0.44	0.437	0.471	*0.556*	0.553	0.566	*0.61*
9	0.546	0.564	0.567	0.61	*0.686*	0.647	0.66	*0.698*
10	0.614	0.614	0.614	0.651	*0.715*	0.629	0.684	*0.738*
avg	0.5039	0.522	0.5233	0.5658	*0.6355*	0.6011	0.6223	*0.6698*

Table 1: Result on the ASAP dataset

p.	KM_{all}	KM_{sel}	KMCL	MPCKM	*$MPCKM_{best}$*	ML_{all}	ML_{sel}	ML_{best}
	Clustering 10 cluster. 40 seeds					supervised ML 50 items		
avg	0.7928	0.7244	0.7493	0.7695	0.7864	0.7001	0.7213	0.8848

Table 2: Average results on the PG dataset

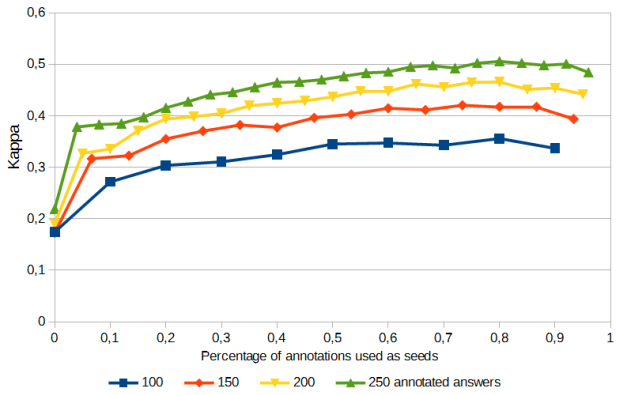


Table 3: Tradeoff between the number of seeds and the number of clusters for different overall amounts of human annotation steps.

lection and constraints before clustering, and the number of clusters, where the centroid of each cluster is labeled after clustering. To do so, we evaluate the effect of different splits between labelled seeds and cluster centroids for MPCKM clustering with centroid-based label propagation for different sizes of n. The results obtained for n = 100, 150, 200, 250 are shown in the curves of Table 3; plotted on the x-axis is the percentage of annotation steps used as seeds. The curves cover distributions from 0 seeds (n clusters) to n-10 seeds (10 clusters).

Unsurprisingly, we see that a higher overall number of annotated data yields a better clustering performance. As an interesting result of the experiment, we observe that the curves peak always between 75 and 80% of annotated data used as seeds, i.e., we profit more from adding more seeds then from adding more clusters.

5.4. Experiment 3: Comparison with Unsupervised Dimensionality Reduction

In Experiment 1, we used labeled seeds for supervised attribute selection. The clustering literature, however, also proposes unsupervised dimensionality reduction methods (Alelyani et al., 2013). Since this might have a similar effect without using any seeds, we compare our results on supervised feature selection from Experiment 1 to two methods of unsupervised feature selection. Principal Component Analysis (PCA, (Pearson, 1901)) is a dimensionality reduction technique that converts high-dimensional data into a smaller number of independent variables. We performed PCA using the t-SNE toolkit (van der Maaten and Hinton, 2008) for the ASAP dataset, reducing it to 500 features. As a second option, we consider feature selection by frequency, following the rationale that features occurring in only a few items are less helpful: in the *frequency*-filtered feature set condition, we only use features that occur in at least 20 answers.

First, we compare whether these two feature selection methods are beneficial for unsupervised k-means clustering, i.e., we compare to the unsupervised baseline KM_{all} with 150 cluster from Experiment 1 that uses all features. To account for the fact that metric learning as used in the MPCKM algorithm might be beneficial even in the absence of constraints, we also evaluate using metric learning without pairwise constraints (MKM). Results are presented in the second and third column of Table 4.

We see that neither of the unsupervised feature selection methods helps for KM_{all}. We also see that metric learning, which was beneficial in combination with constraints

features	KM_{all}	MKM	MPCKM
all features	0.504		0.566
PCA	0.323	0.317	0.374
frequency-filtered	0.500	0.489	0.563

Table 4: Unsupervised feature selection for two versions of completely unsupervised clustering: k-means (KM) and k-means with metric learning (MKM) and unsupervised feature selection as a preprocessing step for semi-supervised clustering (MPCKM).

in Experiment 1, does not help here: MKM performance decreases compared to KM_{all}. PCA shows in general a much worse performance on both the KM_{all} and MKM conditions.

Next we explore whether a combination of unsupervised feature reduction with *semi-supervised clustering* helps. To do that we run the MPCKM algorithm from Experiment 1 including supervised feature selection, but with each of the two unsupervised feature reduction methods as a preprocessing step. We can see in the last column of Table 4 that we do not beat our previous results using PCA, but reach very similar results with frequency-reduced features. Such unsupervised feature selection methods thus provide the additional benefit of reducing runtime substantially, and we will investigate them in future work.

5.5. Experiment 4: Reusing Seeds for Label Propagation

In this experiment, we examine how seeds selected before clustering can be reused for labelling cluster centroids after clustering. In experiments 1 to 3, we have selected the seeds for feature selection and constraints randomly; and by chance some seeds will overlap with cluster centroids, which have to be labeled for label propagation. Our goal in this experiment is to select seeds in such a way that they will have a higher overlap with the cluster centroids. For random seed selection and our setting with 40 clusters and 110 seeds out of 1000 answers, we can expect to find on average 11% of the cluster centroids among the seeds, i.e., on average 4.4 out of 40 centroids.

We increase this random overlap through an informed iterative selection of seeds. We start with a small set of 20 initial seed items for clustering. We then select one new seed at a time based on the previous clustering, recluster, and repeat this procedure until 110 seeds (as in experiment 1) are reached. We use a sampling strategy inspired by *diversity sampling* in AL, cf. (Brinker, 2003) where the goal is to cover the complete feature space. In order to cover as many clusters as possible by our seeds, we select the cluster with the lowest frequency of labeled items (and the bigger one in case of ties) to choose the new seed for the next iteration. In order to get a good representative for that cluster that will be reusable in label propagation, we choose the item closest to the centroid as the next one to be labelled. To avoid artifacts of randomization, we average all results over 5 random seed sets per prompt.

We find that selecting seeds through *diversity sampling* increases the overlap between seeds and cluster centroids to on average 11; the actual clustering performance does not differ substantially from random sampling. Those saved 12 human annotation steps can of course be used as additional seeds in our assumed setup of 150 available human annotation steps. We thus use an additional annotation setup where we keep adding seeds using diversity sampling until the total number of labeling steps reaches or surpasses a fixed number of labeling steps for the first time. (As the number of actually labeled data does not always increase completely linearly in each sampling step, we adapted this value to 148 instead of 150 in order to make sure that we do not label on average more than 150 items.) In that setup, we get some further performance improvement up to on average **0.577**, our overall best result for 150 human annotation steps. [3]

6. Conclusion

In this paper, we have examined semi-supervised clustering methods for short answer scoring in a scenario where a set of items has to be graded with a fixed limited amount of human annotations. We have shown how this limited effort can best be used in the form of seeds for feature selection and constraints and post clustering for centroid-based label propagation. We have found that using MPCKM clustering with pairwise CL constraints and metric learning combined with supervised feature selection brings a large performance boost that (i) cannot be reached using unsupervised methods alone and (ii) comes closer to the performance of supervised machine learning methods. Selecting seeds based on diversity additionally reduces human effort as such seeds can be efficiently used for label propagation without having to label new examples.

As direction for future work, we will also explore the usage of different similarity metrics such as sentence similarity of answers pairs, which are potentially highly useful for clustering but not applicable in an ML-based approach.

Acknowledgments

This work has been funded by the Cluster of Excellence "Multimodal Computing and Interaction" of the German Excellence Initiative.

7. Bibliographical References

Alelyani, S., Tang, J., and Liu, H. (2013). Feature selection for clustering: A review. In *Data Clustering: Algorithms and Applications*, pages 29–60.

Amigó, E., Gonzalo, J., Artiles, J., and Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Inf. Retr.*, 12(4):461–486, August.

Basu, S., Banerjee, A., and Mooney, R. J. (2002). Semi-supervised clustering by seeding. In *Proceedings of the Nineteenth International Conference on Machine Learning*, ICML '02, pages 27–34, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[3] We also investigated a more extreme reusing strategy that reuses any seed contained in a cluster for label propagation (not just the centroid), but found it to be less effective, potentially, because less prototypical items are used for label propagation.

Basu, S., Jacobs, C., and Vanderwende, L. (2013). Powergrading: A clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 1:391–402.

Bilenko, M., Basu, S., and Mooney, R. J. (2004). Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 11–, New York, NY, USA. ACM.

Brinker, K. (2003). Incorporating diversity in active learning with support vector machines. In *In Proceedings of the 20th International Conference on Machine Learning*, pages 59–66. AAAI Press.

Brooks, M., Basu, S., Jacobs, C., and Vanderwende, L. (2014). Divide and correct: Using clusters to grade short answers at scale. In *Proceedings of the First ACM Conference on Learning @ Scale Conference*, L@S '14, pages 89–98, New York, NY, USA. ACM.

Burrows, S., Gurevych, I., and Stein, B. (2014). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117, October.

Cohen, J. (1968). Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. *Psychol Bull.*, (70):213–220.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November.

Heilman, M. and Madnani, N. (2015). The impact of training data on automated short answer scoring performance. *Silver Sponsor*, pages 81–85.

Higgins, D., Brew, C., Heilman, M., Ziai, R., Chen, L., Cahill, A., Flor, M., Madnani, N., Tetreault, J. R., Blanchard, D., Napolitano, D., Lee, C. M., and Blackmore, J. (2014). Is getting the right answer just about choosing the right words? the role of syntactically-informed features in short answer scoring. *Computation and Language*.

Horbach, A. and Palmer, A. (2016). Investigating active learning for short-answer scoring. In *Proceedings of BEA 2016*, pages 629–637.

Horbach, A., Palmer, A., and Wolska, M. (2014). Finding a tradeoff between accuracy and rater's workload in grading clustered short answers. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, pages 588–595, Reykjavik, Iceland.

Lloyd, S. P. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137.

Meurers, D., Ziai, R., Ott, N., and Kopp, J. (2011). Evaluating answers to reading comprehension questions in context: Results for german and the role of information structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9, Edinburgh, Scottland, UK. Association for Computational Linguistics.

Mohler, M., Bunescu, R., and Mihalcea, R. (2011). Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 752–762, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572.

Settles, B. (2010). Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11.

Sukkarieh, J. Z. and Pulman, S. G. (2005). Information extraction and machine learning: Auto-marking short free text responses to science questions. In *Artificial Intelligence in Education - Supporting Learning through Intelligent and Socially Informed Technology, Proceedings of the 12th International Conference on Artificial Intelligence in Education, AIED 2005, July 18-22, 2005, Amsterdam, The Netherlands*, pages 629–637.

Tandalla, L. (2012). Scoring short answer essays.

van der Maaten, L. and Hinton, G. (2008). Visualizing high-dimensional data using t-sne.

Zesch, T., Heilman, M., and Cahill, A. (2015). Reducing annotation efforts in supervised short answer scoring. In *Proceedings of the Building Educational Applications Workshop at NAACL*.

Analyzing Vocabulary Commonality Index Using Large-scaled Database of Child Language Development

Yan Cao[1,2], Yasuhiro Minami[1], Yuko Okumura[2], and Tessei Kobayashi[2]
[1]The University of Electro-Communications, [2]NTT Communication Science Laboratories
[1]1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan, [2]2-4 Hikaridai, Seika, Kyoto 619-0237, Japan
caoyan@sd.is.uec.ac.jp, minami.yasuhiro@is.uec.ac.jp, {okumura.yuko, kobayashi.tessei}@lab.ntt.co.jp

Abstract
The present study proposed a vocabulary commonality index for child language development to investigate to what extent each child acquires common words during the early stages of lexical development. We used large-scaled, vocabulary-checklist data from Japanese-speaking children (N=1,451) aged 8-48 months to estimate their age of acquisition (AoA) of 2688 words by logistic regression. Then we calculated the vocabulary commonality index for each child with two datasets. The results showed that as their vocabulary size increases, children who have the same vocabulary size tend to produce common words with the same ratio.

Keywords: Large-scaled Vocabulary Database, Child Language Development, Age of Acquisition, Vocabulary Commonality Index

1. Introduction

How children acquire vocabulary is among the most central issues in the fields of cognitive science and developmental psychology. Many previous studies have scrutinized what types of words young children acquire during their early stages of lexical development using a vocabulary checklist methodology, such as the MacArthur-Bates Communicative Development Inventories (e.g., Bates et al., 1995; Caselli et al., 1995; Fenson et al., 1994; Frank et al., 2017). For example, Caselli, Casadio and Bates (1999) found that 18-30 month-old English- and Italian-speaking children tend to produce more social words (e.g., people's names, games, routines, etc.) than other types of words in their first 50 words and more common nouns (e.g., animals, foods, toys, etc.) in their first 100-500 words. Although such category analyses of vocabulary are useful for grasping children's overall tendencies in developmental stages, they are less satisfactory for understanding more detailed changes and the individual differences of vocabulary development because the category ranges are wide-ranging and sparse (i.e., only four categories).

One possible solution for grasping detailed changes and individual differences in vocabulary development is to directly compare the word items (rather than categories) acquired by each child with the word items acquired on average by many children and roughly estimate the commonality between both word sets. If we can successfully estimate the commonality in a statistically reliable way, we might be able to understand to what extent each child acquires common words and further clarify the detailed changes and individual differences of vocabulary development.

In the present study, we propose a vocabulary commonality index for child language development by estimating to what extent each child can say common words. We also create a new vocabulary-checklist that includes much more word items (i.e., 2688 words) than the Japanese version of MacArthur-Bates CDI (i.e., 448 words). This is because we thought that the word range of MacArthur-Bates CDI was insufficient to grasp detailed changes and individual differences in vocabulary development. Therefore, using such a new vocabulary-checklist, we create a large-scaled database of vocabulary database in Japanese-speaking children.

2. Data Collection

The data for the study came from two datasets.

2.1 Dataset 1: Tablet Survey

2.1.1 Participants

We collected the data of 1,451 Japanese-speaking toddlers/children (776 boys and 675 girls) whose ages ranged from 8 to 48 months from their parents living in Kyoto, Osaka, and Nara. The participants were recruited from a local newspaper.

2.1.2 Vocabulary Checklist Application

We collected vocabulary data from parents using a tablet PC application that included 2,688 words (Kobayashi, Okumura & Minami, 2016). First, we selected 2,052 words as basic words (common nouns, verbs, adjectives, etc.) from the early vocabulary list that we longitudinally collected from about 800 web users throughout Japan (Kobayashi & Nagata, 2010). Next we added 636 such special words as anime characters and railroad/train names. These 2,688 word items are the most likely ones acquired by children who grew up in a Japanese environment and included almost all of the words from the Japanese version of MacArthur-Bates CDI (Ogura, & Watamaki, 2004; Watamaki & Ogura, 2004).

At the child playroom in our laboratory, the participants checked whether their child could comprehend or say each word (Fig. 1). The vocabulary checklist application consisted of three parts. In Part 1, 2,052 basic words were classified into 29 categories (Fig. 2). The participants had to complete all of the categories before they could finish Part 1 and move to Part 2, which consisted of 636 special words. Part 3 consisted of a free description field that allowed participants to add words that were not on the checklist.



Fig. 1 Parent using tablet PC in a playroom



Fig. 2 Vocabulary checklist application: category list in Part 1 (left) and a panel to check comprehension or production to target words (right)

2.2 Dataset 2: Web Survey

2.2.1 Participants

We sent out an in-depth survey on children's vocabulary acquisition to the members of a large internet research company called Macromill. We received responses from the parents of 1,446 Japanese-speaking toddlers/children (684 boys, 759 girls and 3 with no reported sex) whose ages ranged from 8 to 48 months.

2.2.2 Web-based Survey

Parents completed a web-based survey that asked them to check whether their child could just comprehend a word (without saying it), or whether they could say it or whether neither option fit for identical word items (N = 2,688) in Dataset 1. This survey also asked respondents to input such information as gender, age range, prefecture, region, education level, and job.

3. Methods

We used data from the Tablet Survey to estimate the age of acquisition (AoA) due to its reliability and analyzed the vocabulary commonality index with two datasets.

3.1 Age of Acquisition

As a first step, we calculated the age of acquisition (AoA) in the following way. First, we calculated the acquisition rate of comprehending and speaking each word at every month. According to Minami and Kobayashi (2013), acquisition rate *f(x)* of comprehending and speaking at every month *x* is modeled by the logistic function of Eq. (1). We also introduced parameter *a* that set an upper limit, which was different from the standard logistic function:

$$f(x) = \frac{ae^{cx+b}}{1 + e^{cx+b}}. \qquad (1)$$

The acquisition curves of all the words were modeled by a logistic function using a nonlinear least squares method based on the Gauss-Newton algorithm. We also set constraints so that the upper limit of the acquisition curve of comprehension exceeded the upper limit of the acquisition curve of speaking because we believe that a children's tendency to comprehend precedes speaking. Fig. 3 shows an example of the acquisition curve. The squares indicate the comprehension data, and the circles indicate the speaking data. The logistic function clearly fits in both cases.

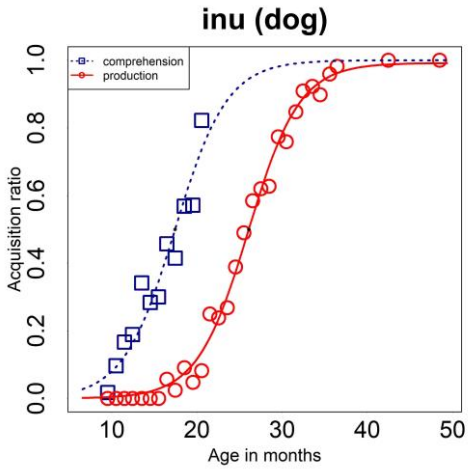


Fig. 3 Acquisition curve of inu (dog)

We also estimated the AoA for each word: the age at which 50% of children can say a word (calculated by x satisfying *f(x) = 0.5* in Eq. (1) with Brent's method). Then we listed those words by their AoA estimation orders. Table 1 shows the first 30 words from the AoA estimation.

3.2 Vocabulary Commonality Index

To clarify the developmental pattern of child vocabulary growth, we estimated to what extent each child can say common words, i.e., how child vocabulary commonality changes based on developmental levels. The Vocabulary Commonality Index (*VocIndex*) is calculated as follows in Eq. (2):

$$VocIndex(i) = \frac{|\, set\ Voc(i) \cap set\ AoA(i) \,|}{|\, set\ AoA(i) \,|}, \qquad (2)$$

where $set\ Voc(i)$ is a set of words that child *i* can say and $set\ AoA(i)$ is a set of words ranked by AoA estimation order with the same number of words in $set\ Voc(i)$. $|\, set\ AoA(i) \,|$ counts how many words are in $AoA(i)$. Moreover, from Eq. (2), note that the higher the vocabulary commonality index is, the more common words a child can say. If all the words that a child can say are common words, her vocabulary commonality index will be 1.

4. Results and Discussion

4.1 Analysis 1: Data from Dataset 1

We calculated the vocabulary commonality index for each child and plotted two types of graphs for it. Fig. 4 shows the relationship between the vocabulary commonality index and age in months. Each point represents an individual child, indicating their age in months and vocabulary commonality index. Because the distribution of the scatter plots was irregular, no clear relationship seems to exist between the vocabulary commonality index and a child's age in months.

Table 1 First 30 words in production for Japanese children based on the AoA estimation. The list includes many infant-directed speech (IDS) words.

Rank	Word	Translation	AoA (days)
1	inaiinaiba	peek-a-boo	433.9
2	manma	(IDS word of meal)	454.8
3	wanwan	(dog sound)	474.4
4	mama	mommy	481.0
5	hai	yes	506.2
6	aq	oh (expression of surprise)	509.2
7	papa	daddy	511.4
8	baibai	bye-bye	513.6
9	a-a	aah (expression of failure)	530.2
10	anpanman	(character name)	546.3
11	nenne	(IDS word of sleep)	561.1
12	bubbu	(vehicle sound)	561.3
13	nyannyan	(cat sound)	561.8
14	iya	no	565.3
15	a-n	(request for opening mouth)	597.6
16	wanwan	(character name)	601.1
17	nainai	(IDS word of cleaning)	606.5
18	ba-ba	grandma	617.6
19	kukku	(IDS word of shoes)	618.0
20	douzo	Here you are	621.3
21	pan	bread	623.2
22	dakko	(IDS word of holding)	625.0
23	shi	(pee sound)	627.5
24	arigatou	thank you	629.7
25	chu	(kiss sound)	630.7
26	un	yes	641.2
27	ji-ji	grandpa	641.2
28	ocha	tea	645.6
29	atchi	there	647.8
30	fu-fu	(blow sound)	654.9

In contrast, Fig. 5 shows a strong relationship between vocabulary commonality index and child vocabulary size. When the vocabulary size was set to the x-axis, a vocabulary commonality index pattern is greatly suggested. When the vocabulary size of each child is small, the vocabulary commonality index varies widely. However, as the vocabulary size increases, the deviation of the vocabulary commonality index gradually becomes smaller. In other words, children who have the same vocabulary size tend to produce common words with the same ratio. One possibility why the vocabulary commonality index decreases from around 2000 words is that as children's vocabulary size increases, they tend to produce more words that are not included on the vocabulary checklist.

Third, using the vocabulary commonality index makes it possible to identify any parents who randomly checked the answers in the vocabulary checklist application. We calculated the probability that common words existed in the AoA word list when the participants randomly selected the answers in the vocabulary checklist application and added the result to the red line in Fig. 5. No participants improperly completed their questionnaires.

Finally, to show the data's trend more clearly, we calculated the moving average values for the vocabulary commonality index and the vocabulary size with intervals of 60 data (Fig. 6). We also calculated the moving standard deviations with a window size of 60 on the vocabulary commonality index. As the total vocabulary size continues to increase, the vocabulary size index also increases. However, during the beginning of children's vocabulary production, the vocabulary commonality index once falls and then goes up again. Table 2 shows the moving average (MA) values and the moving standard deviations (MSDs) of some vocabulary sizes.

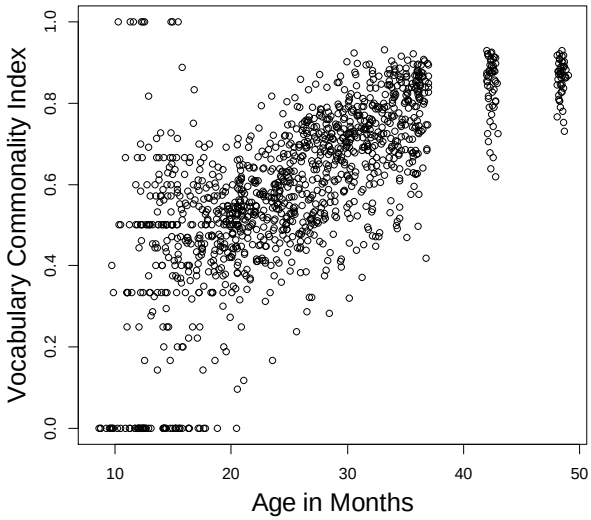


Fig. 4 Vocabulary commonality index plotted by age in months (dataset 1)

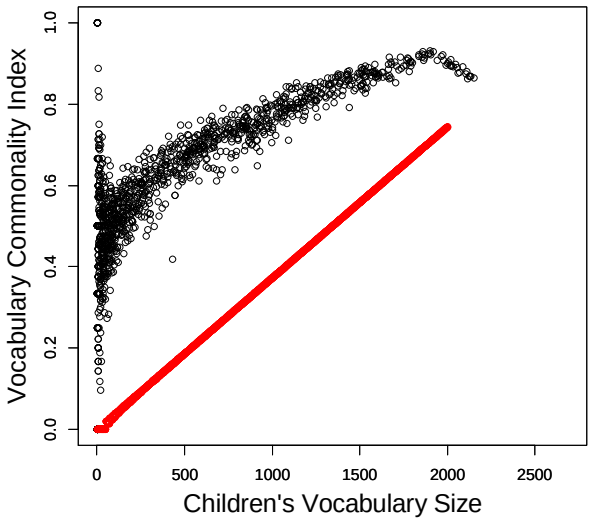


Fig. 5 Vocabulary commonality index plotted by children's vocabulary size (dataset 1)

4.2 Analysis 2: Data from Dataset 2

Using data from Dataset 2 we calculated the vocabulary commonality index for each child and plotted a scatter plot (Fig. 7). Each point shows one child's

vocabulary size versus her vocabulary commonality index. Fig. 7 shows a strong relationship between the vocabulary commonality index and vocabulary size (Fig. 5). Across the Tablet and Web Surveys, as the size of their vocabulary increases, children with the same vocabulary size tend to say common words at the same ratio.

However, several outliers are markedly distant from other points in Fig. 7. To identify these outliers, we consider points under or over 2 times the standard deviation from the moving average outliers. The results are shown in Fig. 8.

We proposed a vocabulary commonality index for child language development by a mathematical method. The present results across two datasets identified the following child vocabulary development pattern: as vocabulary size increases, children at earlier stages of lexical development tend to produce common words at a certain, stable proportion. These findings may play important roles in further studies of child language development. However, the results of this study are limited to 2,688 words. Future studies need to look at the properties of the vocabulary commonality index with a smaller vocabulary size and test for a significant difference between genders.

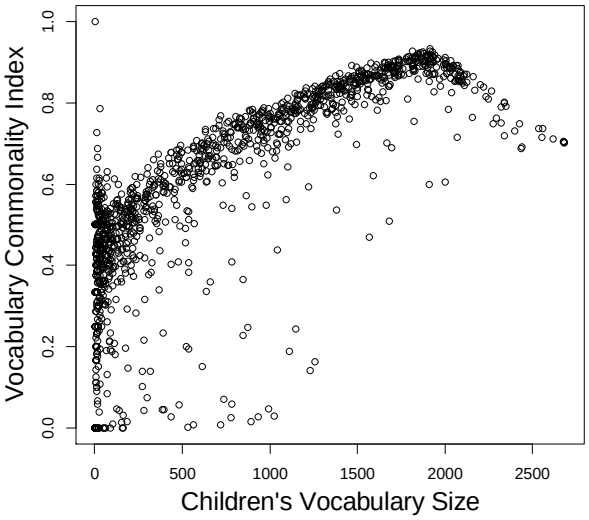


Fig. 7 Vocabulary commonality index plotted by children's vocabulary size (dataset 2)

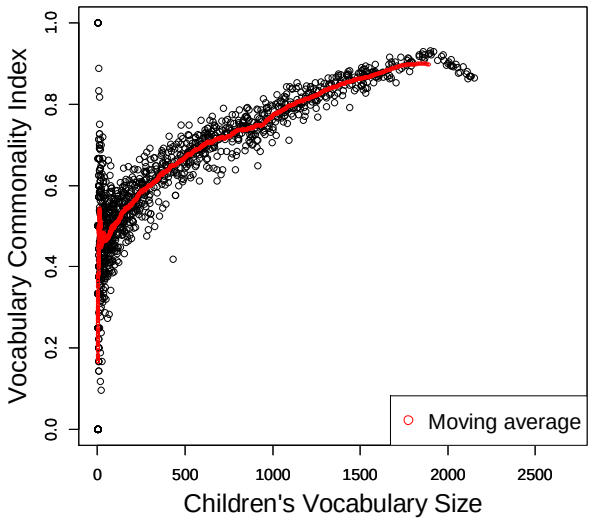


Fig. 6 Moving averages (dataset 1)

Table 2 Moving average values

Vocabulary Size	MA of VocIndex	MSD	MA + 2MSD	MA – 2MSD
30	0.47	0.10	0.67	0.27
50	0.47	0.08	0.63	0.31
70	0.48	0.08	0.64	0.32
90	0.50	0.05	0.60	0.40
150	0.54	0.05	0.64	0.44
300	0.60	0.05	0.70	0.50
450	0.65	0.05	0.75	0.55
850	0.74	0.03	0.80	0.68
1200	0.81	0.02	0.85	0.77
1850	0.90	0.02	0.94	0.86

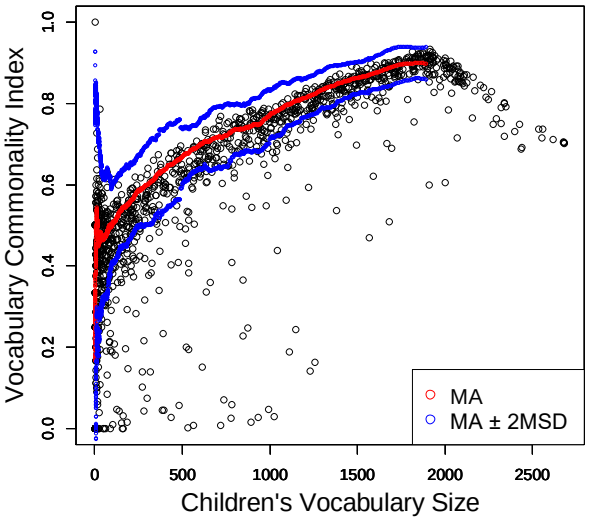


Fig. 8 Moving average (MA) and 2 times moving standard deviation (MSD) from the moving average (dataset 2)

5. Acknowledgements

A part of this study was supported by a Grant-in-Aid for Scientific Research (B) 17H02190. The authors thank the parents for participating in this study.

6. References

Bates, E., Dale, P. S., & Thal, D. (1995). Individual differences and their implications for theories of language development. The Handbook of Child Language, 96-151.

Caselli, M. C., Bates, E., Casadio, P., Fenson, J., Fenson, L., Sanderl, L., & Weir, J. (1995). A cross-linguistic study of early lexical development. Cognitive Development, 10(2), 159-199.

Caselli, C., Casadio, P., & Bates, E. (1999). A comparison of the transition from first words to grammar in English and Italian. Journal of Child Language, 26(1), 69-111.

Fenson, L., Dale, P. S., Reznick, J. S., Bates, E., Thal, D. J., Pethick, S. J., & Stiles, J. (1994). Variability in early communicative development. Monographs of the Society for Research in Child Development, i-185.

Frank, M. C., Braginsky, M., Yurovsky, D., & Marchman, V. A. (2017). Wordbank: An open repository for developmental vocabulary data. Journal of Child Language, 44(3), 677-694.

Kobayashi, T. & Nagata, M. (2010). Early lexical development in Japanese Children: Age of acquisition of first 50 words specified using web diary method. The 20th Child Language Seminar (CLS2010), London, UK.

Kobayashi, T., Okumura, Y., and Minami, Y. (2016). Collecting data on child vocabulary development database by vocabulary-checklist application. IEICE Technical Report HCS2015-59(2016-01), pp. 1-6.

Minami, Y. and Kobayashi, T. (2013). Developmentally-Appropriate Vocabulary Search System. The IEICE transactions information and systems D, Vol., J96-D, No.10, pp. 2612-2624.

Ogura, T., & Watamaki, T. (2004). The Japanese MacArthur communicative development inventory: Words and gestures. Kyoto: Kyoto International Social Welfare Exchange Center.

Watamaki, T., & Ogura, T. (2004). The Japanese McArthur communication development inventory: Words and sentences. Kyoto: Kyoto International Social Welfare Exchange Center.

The ICoN Corpus of Academic Written Italian (L1 and L2)

Federica Cominetti, Mirko Tavosanis
Università di Firenze, Università di Pisa
Piazza di San Marco, 4, Firenze, Italy, via Santa Maria 36, Pisa, Italy
fedcominetti@gmail.com, mirko.tavosanis@gmail.com

Abstract

This paper describes the ICoN corpus, a corpus of academic written Italian, some of the directions of research it could open, and some of the first outcomes of research conducted on it. The ICoN corpus includes 2,115,000 tokens written by students having Italian as L2 students (level B2 or higher) and 1,769,000 tokens written by students having Italian as L1; this makes it the largest corpus of its kind. The texts included in the corpus come from the online examinations taken by 787 different students for the ICoN Degree Program in Italian Language and Culture for foreign students and Italian citizens residing abroad. The texts were produced by students having 41 different L1s, and 18 different L1s are represented in the corpus by more than 20,000 tokens. The corpus is encoded in XML files; it can be freely queried online and it is available upon request for research purposes. The paper includes the discussion of preliminary research in the field of collocations, showing that, in the texts included in the corpus, while learners and natives do use multiword expressions in a similar way, learners can overuse relatively infrequent forms of multiword adverbials, or use some adverbials in a non-standard way.

Keywords: Corpus linguistics, Learners, Italian, multiword expressions, collocations

1. Introduction

Learner corpora are useful and widely used in the study of language and language learning. In this paper we describe the building of the largest (to date) learner corpus of academic written Italian as L2 and of a large comparable corpus of academic written Italian as L1, collectively called "The ICoN corpus", already partially described in Tavosanis (2014) and (2016). The corpus can be queried online while its complete contents are available on request for research purposes.[1]

2. The Context of Learner Corpora Research and the Status of Learner Corpora for Written Italian

Being a learner corpus, the ICoN corpus belongs to a class of corpora well understood in theory and much appreciated for its practical uses (for an overview, see Granger, Gilquin and Meunier 2015). Learner corpora today number in the hundreds: the list of learner corpora maintained by Magali Paquot at the Université Catholique de Louvain includes to date 166 entries, while the Learner Corpus Bibliography maintained by the Learner Corpus Association (at the address www.learnercorpusassociation.org/resources/lcb/) includes more than 1,500 entries.

In the research tradition of the field, "in terms of medium and text type, the dominant focus was – and to a large extent still is – on writing, in particular essay writing" (Granger - Gilquin -Meunier 2015, 2). Well known examples of this include for example the ICLE (International Corpus of Learner English) corpus. This corpus (described in detail in Granger, Dagneaux, Meunier, Paquot, 2009) includes 2.5 millions of words coming from 3,640 unabridged essays, with an average length of 700 words, written by students with 11 different L1s. The essays do cover a variety of topics but for the most part are examples of argumentative writing and general use of English (rather than of English for specific purposes).

The ICoN Corpus, as will be described in detail in the following paragraphs, follows closely this tradition and, at least in size and composition, it can be roughly compared to the ICLE corpus. However, similar corpora are still relatively rare in the study of Italian as FL / L2. The most used resources of this kind are arguably the twin corpora VALICO (Varietà di Apprendimento della Lingua Italiana Corpus Online) and VINCA (Varietà di Italiano di Nativi Corpus Appaiato), realized at the University of Torino (Barbera - Marello 2004). Those corpora do include productions elicitated with similar techniques from FL / L2 learners (VALICO) and mother tongue speakers of Italian (VINCA). In both contexts, informants were mainly university students, with levels of L2 located for the most part in the A2-B1 range. VALICO includes 3,804 texts while VINCA has 680 texts (Allora – Colombo - Marello 2011), for a minimum size of 100 words; the web site of the project currently declares an overall size of 568,000 tokens for VALICO. This twin-corpora approach has a self-evident usefulness for comparison between L1 and L2 productions and it was followed in the building of the ICoN corpus.

Among other learner corpora of written Italian, in addition to a wealth of spoken corpora, we would also like to cite: ISA (Italiano Scritto di Americani), cf. Rastelli 2006; ADIL2 (Archivio Digitale di Italiano L2), realized at the Università per Stranieri of Siena, 432,606 tokens of which two-third are spoken data, cf. Palermo 2009; CAIL2 (Corpus di Apprendenti di Italiano L2), realized at Università per Stranieri di Perugia, cf. Bratankova 2015 (see also Andorno – Rastelli 2009 for an overview of Italian research on the topic). Among those, however, the ICoN corpus is the only one specifically devoted to upper-intermediate and advanced varieties.

3. The ICoN Degree Program and the ICoN Examinations

The ICoN corpus has been assembled at the University of Pisa in close cooperation with ICoN (Italian Culture of the Net), a consortium composed of nineteen Italian

[1] The paper was written jointly. However, for attribution of value, we declare that paragraphs 1-5 and 7 were the work of Mirko Tavosanis and paragraphs 6 and 8-10 were the work of Federica Cominetti.

universities whose aim is the promotion of the Italian language and culture all over the world through e-learning technologies (Tavosanis 2003). Among other educational initiatives, ICoN offers a three-year degree program in Italian Language and Culture, reserved to foreign students and Italian citizens resident abroad. The corpus is composed by short essays composed by the students during the examinations.
Each exam of the ICoN degree program includes the writing of a short text on a specific topic relevant to some aspect of the Italian humanistic culture (e.g. "The Baroque restoration practices of Maratta and the nineteenth century restoration practices of Cavalcaselle: compare two different attitudes towards conservation of the work of art"; "Analyze the relationship between Petrarch and Humanism"; "Illustrate the concept of equivalence and its role in the method of interlinguistic confrontation"). The corpus ICoN collected all such texts written by the ICoN students from 2001, year of the start of the degree program, until 2014.
Examination procedures in all cases required students to choose one topic for the essay among three different given topics. The task had to be completed in less than 90 minutes, but during this time students also had to answer to 30 fixed-choice questions (inside the limit of 90 minutes, students could freely choose how much of their time give to the questions or to the composition of the text). The use of supporting materials, such as textbooks and notes, was forbidden and enforcement of this rule was trusted to proctors in the examination rooms, set in the residence countries of students.
As for the text, the reference length given to students was 300 words (corresponding to an average of around 2200 characters), with penalties for texts both too long and too short. Moreover, during the examination, the number of words was shown to the student by a counter inserted in the writing interface: not surprisingly, the average length of the corpus texts is of 302 tokens.

4. Composition of the ICoN Corpus

The corpus in its published form is composed of 12,556 essays written by 787 different students. A single student then produced 16 texts on average. The inclusion of texts written by students that dropped out of the program and the fact that many students were still completing the program at the time of corpus generation are the reasons why the average is below the 21 texts required by the degree course. On the other hand, some students took the same examination more than once, and therefore some students are credited in the corpus with more than 21 texts.
The total corpus tokens are 3,794,000, of which about 2,115,000 written by Italian L2 students and approximately 1,769,000 written by Italian L1 students. The sum of the two sub-corpora is higher than the total, because some students were registered as "bilinguals" and their work has been included in both corpora. However, only 16 students are eligible for this, corresponding to a total of only 90,000 tokens (2.37% of the total).
The texts included in the two subcorpora were composed by the two groups of students in identical circumstances. This makes it possible to compare the sub-groups according to the consolidated VALICO / VINCA model. In this perspective, two strengths of the ICoN corpus are undoubtedly:

• its size, far superior to the size of the other comparable corpora
• the fact that its contents have not been realized for linguistic purposes but represent actual and evaluated exams; since it was created in a real-life context outside of dedicated language instruction, this kind of output can been considered as somewhat more than "semi-natural" (in the sense of "a pedagogical task that is natural in the context of the language learning classroom": Granger 2015: 10).

As for the encoding, texts were exported and collected as XML files with UTF-8 encoding. Personal data were thoroughly anonymized by the ICoN consortium: birth date was given as year of birth; country of residence was the only information given regarding residence. The encoding was carried out including each text as plain text inside a <doc> element including as attribute values the information regarding the student, as in this sample:

```
<doc id = "pre2008_esamistudente_9489.xml"
idstudente="9489" risposta="RISPOSTA 4"
Area_Titolo_Studio="Linguistica"
Condizione_Lavorativa="In Altra Condizione"
Data_Nascita="1976" Lingue_Conosciute="Rus
Eng Pol"  Madrelingua="Cze" Naz="Rep. Ceca"
Posizione_Lavorativa="" Provincia=""
Sesso="F" Titolo_Studio="Laurea 1°
Livello">
```

The XML files generated in this way were then used both for direct search and as sources for the search engine with Web interface described below.

5. Level of Knowledge of Written Italian

Students enrolled in the ICoN degree program are supposed to have reached a level at least equal to B2. In many cases, their level is definitely higher, while in some rare cases we feel it would be more appropriate to classify their production ability at level B1. Texts included in the corpus are therefore relatively homogeneous productions. At the moment no attempt has been made to evaluate more accurately the level of real competence of each individual student.
An example of a full text is this (Greek L1):

Dialoghi con Leucò è una raccolta di poesie scritte da Pavese e pubblicata nel 1947.L'opera consiste nei 27 componimenti in forma di dialogo fra personaggi mitici.Probabilmente l'autore si ispirò dalla Tesalique Mitologie dell' autrice Paula Phillipson.Nell'opera traspare una teatralità intesa non come visione ma come forma dialettica, basata appunto nei dialoghi che si svolgono fra i vari personaggi a volte in maniera tragica a volte in maniera ironica. Nel centro d'ispirazione del poeta e la sacralità del mito inteso come quel sostratto culturale insostituibile che accomuna tutta l'umanità. Sono gli antichi miti della cultura greca che appartengono a un' era remotta ma che tornano nel nostro tempo con una ciclicità come il continuo alternarsi della vita e della morte, come simboli privilegiati della transcendenza

umana. In tutte le culture,e specialmente in quella classica, sono i miti che hanno dato conforto alle angoscie umane della morte e dei fenomeni inspiegabili.
Pavese e reduce di letture antropologiche, psicologiche ed etnologiche che nel doppoguerra lo avevano appassionato. Autori come Forbenius, Fraser, Levy-Bruhl, opere dello Jung e Levy Strauss. In questo aspetto molto lo influenzò anche la sua conoscenza con Bianca Garufi.Questi temi gia apparsi nella sua opera Feria d'Agosto diventano più marcati, mentre l'idea della donna, dea e belva insieme, riempie la concezione della femmina.La donna appare come forzza della terra e della fertilità,la stessa che da vita e morte,in tutta la sua matterialità e mai come una transcendenza.
L' autore considerò questi componimenti come la sua opera più completa, non a caso una coppia fu trovata accanto allo scrittore quanto morì suicida nel 1950.

Another factor to take into account is the frequency of imitation, sometimes almost literal, from the teaching materials used by the students. This phenomenon is obvious and natural in all circumstances; it is even more so in a situation in which both didactic materials and final papers take the form of written texts displayed on computer screens.

6. L1s of Students

Regarding geographic distribution, students graduated with ICoN come from more than 60 countries around the world. Their L1s reflect this variety: in the registration phase students have indicated in fact 41 different L1s, including Italian. The L1s whose speakers have produced more than 20,000 tokens in total are 18 (as shown in Table 1).

Language	Tokens
Spanish	704,643
Portuguese	233,275
Serbian and Croatian	173,279
Russian	147,605
Greek	128,307
Polish	122,878
German	68,430
Albanian	66,470
French	45,694
English	44,721
Rumanian	43,518
Bulgarian	35,165
Japanese	33,552
Latvian	25,652
Lithuanian	25,158
Hungarian	24,618
Turkish	23,797
Czech	21,568
Sum	1,968,330

Table 1: Tokens according to the L1s of students

Languages with fewer tokens are, in descending order, Arabic, Maltese, Luo, Macedonian, Finnish, Leonese, Slovenian, Ukrainian, Dutch, Bosnian, Estonian, Maithili, Faroese, Azerbaijani, Tagalog, Slovenian, Vietnamese, Urdu, Kohati Hindko (the latter, along with the Leonese, has no ISO 639-2 code).
Serbian and Croatian were merged, not only because most of the linguists consider Serbian and Croatian a single language, but also because in many cases the students themselves (both Serbians and Croats) stated that their mother tongue was “Serbian and Croatian”.
In addition to the above mentioned case of 16 students declaring to be bilinguals in Italian and another language, one student declared to be bilingual in Russian and Belarusian, one in Spanish and Catalan and one in Italian and Venetian dialect, while three native speakers of Maltese are bilingual with English. In all cases, speakers were counted as L1 for all the languages in which they had indicated this competence. In the currently on-line interface it is also possible to search in relation to these speakers also “Belarusian”, “Catalan” and “Venetian”, which (unlike Maltese) are not represented by speakers who have these languages as exclusive L1.

7. Publication and Web Interface

The corpus has been put online at the Web address http://corpusicon.fileli.unipi.it/ and it is hosted by the University of Pisa. Its current search interface, created by Net7 s.r.l., is shown in Figure 1.



Figure 1: The interface of the ICoN Corpus

In its current form, the interface allows to search for single words. The search can be filtered by L1, country of birth, country of residence, current nationality, sex, age (by age brackets), academic qualifications and area of academic qualifications. Results are displayed as shown in Figure 2.
It is planned to replace the current interface with a more flexible interface by the first half of 2018. In a further

step, the contents of the corpus could be POS-tagged to improve the quality of research.



Figure 2: Sample of search results

8. Preliminary Research on Collocations

The ICoN corpus has already been extensively used in our didactic practice at the University of Pisa. It has been also used in researching the use of Italian definite articles by learners (Cominetti - Tavosanis forthcoming) and the use of focalizing adverbs by learners (Roy - Tavosanis forthcoming).

A further research topic we are currently applying the ICoN corpus to regards the field of collocations (see among others Firth 1957, Sinclair 1991, Evert 2005, 2009, Masini 2009).

Collocations can be defined as frequent word combinations whose syntactic and semantic properties cannot be entirely derived from those of the components. Collocations can be characterized by different levels of cohesion, ranging from fully fixed idioms (i.e. *burn the midnight oil*) to lexical preferences in contexts where other paradigmatic choices are available (i.e. *brush teeth* is preferred, but *wash teeth* is acceptable).

In recent years, the study of collocations and combinations of words has proved to be of great interest not only to lexicology and lexicography, but also to their applications to other fields, including first and second language acquisition. In L2/FL research, the study of collocations allows addressing questions of great theoretical value, such as the Sinclairian dynamics between open choice principle and idiom principle. Learners' behavior has an important heuristic value in suggesting how the mental lexicon is stored, and how important should the formulaic language be considered in comparison to paradigmatic choices (cf. among others Granger 1998, Nesselhauf 2005, Bratankova 2015).The ICoN corpus thus provides an ideal resource, allowing the comparison between native (ITA sub-corpus) and non-native (STRA sub-corpus) productions.

Up to now, a qualitative analysis of the corpus has allowed detecting some aspects of language formulaicity particularly problematic to learners, such as the preposition selection (1), the support verb selection (2) and the paradigmatic fixity (3, 4):

(1) Il romanticismo, realismo e naturalismo dell' Ottocento vengono sostituiti *per* movimenti che si trovano piu' in sintonia colle nuove preocupazioni filosofiche di questo periodo (L1 Spanish).

'Romanticism, realism and naturalism of the nineteenth century have been substituted *from* movements that are more attuned to the new philosophical concerns of this age'.

(2) Monteverdi esordi' molto giovane con il Primo libro dei madrigali, il genere molto in voga nel Cinquecento, che lo *fece* famoso. (L1 Russian).

'Monteverdi made his debut very young with his first madrigal book, that *rendered* him famous'.

(3) Carducci *al suo turno* fece un ampio studio del autore e dopo molti anni di ricerca pubblicò la sua opera che conteneva dettagliate annotazioni del Canzoniere. (L1 Luo).

'Carducci *on his time* made a large study of the author and after many years of research published his work that included detailed notes of the Canzoniere'.

(4) La situazione porta a volte il carattere drastico per la gioventù che, spinta dal malessere, entra nei *cerchi criminali*. (L1 Russian).

'The situation carries sometimes the drastic character to the young that, driven by unease, enter the *crime circles*'.

In (1) the preposition *per* 'by' is selected instead of *da* 'from, by'. In (2) the support verb *rendere* 'make' is wrongly substituted by *fare* 'make, do'. In (3) and (4) respectively, the paradigmatic fixity of idioms *a sua volta* 'on his turn' and *giri criminali* 'crime rings' is violated, since the correct words *volta* and *giri* are substituted by synonyms *turno* and *cerchi*.

A quantitative analysis reveals that, if the three mentioned kinds of errors are relatively frequent, the specific examples prove to be rare. For example, the wrong expression *al suo turno*, found in (3), is attested only 4 times in the corpus, while the standard form *a sua volta* counts 93 occurrences in the L2 sub-corpus (a very similar figure is found in the L1 sub-corpus, where 114 occurrences are attested).

A more accurate statistical analysis on the topic has been conducted using EXTra (Passaro - Lenci 2016), a term recognition system that evaluates the association measure of structured POS-sequences. Both effect-size measures (i.e. Mutual information) and significance measures (i.e. t-score, log-likelihood) are used, so that the dimension of the corpus and the particularly high frequency of some tokens (like articles, prepositions) do not affect the results. In particular, the following kinds of multiword lexicon and word collocations have been investigated up to now: multiword nouns with structure Noun Prep Noun (i.e. *punto di vista* 'point of view') and Noun Adj (i.e. *essere umano* 'human being'), multiword verbs (Verb Noun, Verb Prep Noun, Verb Noun Adj) and adverbials (Prep Noun, Prep Adj Noun, Prep Noun Adj) and prepositional locations (Prep Noun Prep).

Such preliminary analysis has shown that the most frequent multiword expressions included in the L1 and L2 sub-corpora tend to be the same. See, for example, in Table 2 (cf. Annex), the list of the 20 multiword nouns (structure Noun Prep Noun and Noun Adj) characterized

by the highest LMI in L1 and L2 sub-corpora. Table 3 includes the adverbials characterized by the highest LMI.
In general, both in the case of nouns and adverbials, the two sub-corpora share the same forms. As for nouns, the datum that learners' behavior is not far from natives' is interesting and could not be easily assumed. As for adverbials, however, being the adverbials a grammatical class, the similarity of the L1 and L2 lists is less surprising. Actually, any significant difference can in this case highlight untypical or distorted uses. For example, in STRA, alongside the very cohesive *allo stesso tempo* 'at the same time', the synonym *nello stesso tempo* is found, while the latter does not appear in the ITA list. In fact, 117 occurrences of *nello stesso tempo* are found in STRA and only 32 in ITA, while *allo stesso tempo* is found 172 times in STRA and 151 in ITA. The ITA data are compatible with standard Italian as testified by a Google search: *nello stesso tempo* has 1,5 million occurrences while *allo stesso tempo* appears 13,3 million times. However, the same search on the CORIS corpus (Rossini Favretti 2000) finds 1771 occurrences for *nello stesso tempo* and just 1908 for *allo stesso tempo*, so it is necessary to be careful in drawing conclusions from rough data.
Another interesting example is provided by the adverbial *dall'altra parte* 'on the other side', quite frequent in STRA (87 occurrences, LMI 3088) but rare in ITA (8 occurrences, LMI < 1000). In this case, the surprisingly high frequency in STRA detects a mistake: the learners use *dall'altra parte* in place of the correct form *d'altra parte* 'after all, moreover' as it is clear in the following example:

(5) Allora potevano lavorare di più d'estate quando il tramonto del sole veniva molto più tardi dell'inverno e *dall'altra parte* potevano cominciare a lavorare molto più presto di mattina come il sole d'estate veniva molto più prima. (L1 Czech).

'Then they could work more in the summer when the sunset arrived much later than in winter and *on the other side* they could start working much earlier in the morning since the summer sun rises earlier'.

Moreover, in the STRA LMI list the adverbial *da un'altra parte* 'somewhere else' appears as well (25 occurrences), while it does not appear in ITA's list (only 1 occurrence). Also in this case, the surprisingly high frequency reveals a non-standard use by the learners: in fact, an analysis of the occurrences in STRA reveals that *da un'altra parte* is often used in contrast to *da una parte* 'on one side', in a context where natives normally use *dall'altra parte* 'on the other side'. In other cases, *da un'altra parte* is wrongly used instead of *da un altro lato* 'on the other side'.
In the mentioned examples, the data extracted from the two sub-corpora testify that ITA more accurately reflects the use of the adverbials typical of standard Italian, while in STRA two tendencies emerge: relatively infrequent forms can be overused, while some adverbials can be used in a non-standard way.

9. Future Work

To date, the corpus has been used for didactic purposes and for traditional research, especially for the use of Italian definite articles. It is expected that use will continue.
However, approaches with a stronger computational side are also envisaged. Preliminary results as those seen in § 8 are promisingly showing that collocations are one of the features – although subtle – that allow detecting non-native productions even in presence of an advanced L2 competence and native-like grammatical behaviors. For this reason, we are willing in particular to explore the possible application of this feature to the setting of tools for the task of Automatic Native Language Identification, which has already been tested to a limited extent on the corpus ICoN.

10. Acknowledgements

The corpus work was done within the Italian national research program PRIN "SCRIBE / Scritture Brevi" (2013-2016, coordinated by Pietro Trifone - University of Rome "Tor Vergata"). The ICoN Consortium has generously contributed to the realization of the corpus through the provision of materials and dedicated funding.

11. Bibliographical References

Allora, A. - Colombo, S. - Marello, C. (2011). I corpora VALICO e VINCA: stranieri e italiani alle prese con le stesse attività scritte. In Maraschio, M. - De Martino, D. (eds.) *La Piazza delle lingue. L'italiano degli altri*. Firenze, Accademia della Crusca, pp. 49-61

Andorno, C. - Rastelli, S. (2009). Corpora di italiano L2. Tecnologie, metodi, spunti teorici. Perugia, Guerra.

Barbera, M. - Marello, C. (2004). VALICO (Varietà di apprendimento della lingua italiana Corpus Online): una presentazione. *Itals* II, 4, pp. 7-18.

Blanchard, D. - Tetreault, J. - Higgins, D. – Cahill, A. – Chodorow, M. (2013). TOEFL11: A Corpus of Non-Native English, Educational Testing Service, Research Report ETS RR–13-24, http://www.ets.org/Media/Research/pdf/RR-13-24.pdf

Bratankova, L. (2015). Le collocazioni Verbo + Nome in apprendenti di italiano L2. Tesi di dottorato, Università per stranieri di Perugia.

Bruni, F. - Alfieri, G. - Fornasiero, S. - Tamiozzo Goldmann, S. (2006). Manuale di scrittura e comunicazione. Bologna, Zanichelli.

Cominetti, F. - Tavosanis, M. (forthcoming). Interferenza della L1 nell'apprendimento degli articoli in italiano L2: una ricerca sul corpus ICoN. In *Atti del convegno SILFI 2016*, Madrid.

Dell'Orletta, F. - Montemagni, S. - Venturi, G. (2011). Read-it: Assessing Readability of Italian texts with a View to Text Simplification. In *Proceedings of the 2nd Workshop on Speech and Language processing for Assistive Technologies*, Edinburgh, pp. 73-83.

Dell'Orletta, F. - Montemagni, S. - Venturi, G. (2016). Esplorazioni computazionali nello spazio dell'interlingua: verso una nuova metodologia di indagine. In *Atti del XLVIII Congresso Internazionale della Società di Linguistica Italiana*. Roma, Bulzoni, pp. 143-161.

Evert, S. (2005). The Statistics of Word Cooccurences. Word pairs and Collocations. Ph.D. Thesis, University of Struttgart.

Evert, S. (2009). Corpora and collocations. In Ludeling, A. - Kyto, M. (eds.) *Corpus Linguistics: An International Handbook*, Volume 2. Berlin, New York, de Gruyter, pp. 1212-1248.

Ferris, D. R. - Hedgcock, J. (2013) Teaching L2 Composition: Purpose, Process, and Practice. 3rd edition. Routledge.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-55. In *Studies in Linguistic Analysis, Philological Society*. Oxford, pp. 1-32, reprint in F. R. Palmer (ed.) (1968). Selected papers of J.R. Firth 1952-1959. Harlow, Longman, pp. 168-205.

Gilquin, G. (2015). From design to collection of learner corpora. In Granger, S. - Gilquin, G. - Meunier, F. (eds.) *The Cambridge Handbook of Learner Corpus Research*. Cambridge, Cambridge University Press, pp. 9-34.

Granger, S. (1998). Prefabricated patterns in advanced EFL writing: collocations and formulae. In Cowie A. P. (ed.). *Phraseology: Theory, Analysis and Applications*. Oxford, Oxford University Press, pp. 145-160.

Granger, S. - Dagneaux, E. - Meunier, F- - Paquot, M. (2009). *International corpus of learner English*. Louvain, Presses universitaires de Louvain.

Granger, S. - Gilquin, G. - Meunier, F. (2015). Introduction: learner corpus research – past, present and future. In Granger, S. - Gilquin, G. - Meunier, F. (eds.) *The Cambridge Handbook of Learner Corpus Research*. Cambridge, Cambridge University Press, pp. 1-5.

Köpke, B., - Schmid, M. S., - Keijzer, M. - Dostert, S. (eds.) (2007). Language Attrition: theoretical perspectives. Amsterdam, John Benjamins.

Koppel, M., - Schler, J. – Zigdon, K. (2005). Determining an author's native language by mining a text for errors. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD '05)*, pp. 624-62.

Masini, F. (2009). Combinazioni di parole e parole sintagmatiche. In Lombardi Vallauri, E. - Mereu, L. (eds.). *Spazi linguistici. Studi in onore di Raffaele Simone*. Roma, Bulzoni, pp. 191-209.

Nesselhauf, N. (2003). The use of collocations by advanced learners of English and some implications for teaching. In *Applied Linguistics*, 24, pp. 223-242.

Nesselhauf, N. (2005). Collocations in a Learner Corpus, Amsterdam-Philadelphia, John Benjamins Publishing Company.

Palermo, M. (ed.) (2009). Percorsi e strategie di apprendimento dell'italiano lingua seconda: sondaggi su ADIL2. Perugia, Guerra Edizioni.

Passaro L. C. - Lenci A. (2015). Extracting terms with EXTra. In Proceedings of EUROPHRAS 2015. (pp. 188–196). Málaga (Spain), July 2015.

Rastelli, S. (2006). ISA 0.9. Written Italian of Americans: syntactic and semantic tagging of verbs in a learner corpus. *Studi Italiani di Linguistica teorica e Applicata* 1, pp. 73-99.

Rossini Favretti, R. (2000). Progettazione e costruzione di un corpus di italiano scritto: CORIS/CODIS. In Rossini Favretti R. (ed.), *Linguistica e informatica. Multimedialità, corpora e percorsi di apprendimento*. Bulzoni, Roma, pp. 39-56.

Roy, T. - Tavosanis, M. (forthcoming). Il focalizzatore *anche* nei testi scritti di studenti con lingue indoarie come L1. In *Atti del convegno SLI 2017*, Napoli.

Sinclair, J. M. (1991). Corpus, Concordance, Collocation. Oxford, Oxford University Press.

Spirito, R. (1999). La scrittura accademica. In Palloti G. (ed). *Scrivere per comunicare*. Milano, Bompiani, pp. 205-219.

Tavosanis, M. (2003). Insegnamento di lingua e cultura italiana a stranieri: l'esperienza di ICoN. In *Italiano e italiani nel mondo. Italiani all'estero e stranieri in Italia: identità linguistiche*. Roma, Bulzoni, vol. 2, pp. 1-13.

Tavosanis, M. (2014). Corpus ICoN: una raccolta di elaborati di italiano L2 prodotti in ambito universitario. In *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of the Fourth International Workshop EVALITA 2014*. Pisa, Pisa University Press, pp. 370-373.

Tavosanis, M. (2016). L'italiano L2 negli elaborati universitari del corpus ICoN: esempi di analisi. In Manco, A., - Mancini, A. (eds.), *Scritture brevi: segni, testi e contesti. Dalle iscrizioni antiche ai tweet*. Napoli, Università degli studi di Napoli «L'Orientale», pp. 177-187.

	STRA	LMI	ITA	LMI
1	punto di vista (1)	1928	punto di vista (1)	1213
2	punto di riferimento (6)	579	essere umano (3)	352
3	essere umano (2)	544	mezzo di comunicazione (5)	324
4	stato d'animo (4)	445	stato d'animo (4)	316
5	mezzo di comunicazione (3)	440	messa in scena (15)	311
6	movimento culturale (14)	365	punto di riferimento (2)	308
7	centro urbano (7)	356	centro urbano (7)	306
8	opera d'arte (8)	350	opera d'arte (8)	258
9	punto di partenza (12)	346	presa di coscienza (-)	241
10	classe sociale (-)	329	classe dirigente (11)	239
11	classe dirigente (10)	266	pena di morte (-)	216
12	tasso di natalità (17)	248	punto di partenza (9)	199
13	rito di passaggio (19)	226	diritto di voto (16)	174
14	posto di lavoro (-)	208	movimento culturale (6)	164
15	messa in scena (5)	202	sala cinematografica (-)	146
16	diritto di voto (13)	195	servizio di leva (-)	139
17	storia d'amore (18)	175	tasso di natalità (12)	136
18	capo del governo (-)	159	storia d'amore (17)	107
19	mezzo di trasporto (-)	154	rito di passaggio (13)	106
20	gioco di parole (-)	141	via d'uscita (-)	102

Table 2: Multiword nouns (Noun Prep Noun, Noun Adj) in STRA and ITA ordered by decreasing LMI.

	STRA	LMI	ITA	LMI
1	per la prima volta (1)	11383	per la prima volta (1)	8541
2	allo stesso tempo (2)	11217	allo stesso tempo (2)	6341
3	per esempio (3)	7516	ad esempio (4)	5933
4	nello stesso tempo (-)	5235	per esempio (3)	2946
5	ad esempio (3)	5124	al tempo stesso (-)	2704
6	dall'altra parte (-)	3088	in un certo senso (8)	1952
7	in gran parte (10)	2112	in primo luogo (-)	1940
8	in un certo senso (6)	2068	allo stesso modo (13)	1826
9	a volte (13)	1538	a livello internazionale (-)	1359
10	in primo piano (14)	1448	in gran parte (7)	1338
11	nel primo caso (-)	1336	in particolare (-)	1201
12	nello stesso modo (-)	1235	in modo particolare (-)	1176
13	allo stesso modo (8)	1109	a volte (9)	1108
14	al primo posto (-)	1105	in primo piano (10)	1091
15	da un'altra parte (-)	1048	in seguito (-)	1070

Table 3: Preposition-headed adverbials in STRA and ITA ordered by decreasing LMI.

Revita: a Language-learning Platform at the Intersection of ITS and CALL

Anisia Katinskaia, Javad Nouri, Roman Yangarber
University of Helsinki, Finland
first.last@cs.helsinki.fi

Abstract

This paper presents Revita, a Web-based platform for language learning—*beyond the beginner level*. We anchor the presentation in a survey, where we review the literature about recent advances in the fields of computer-aided language learning (CALL) and intelligent tutoring systems (ITS). We outline the established *desiderata* of CALL and ITS and discuss how Revita addresses (the majority of) the theoretical requirements of CALL and ITS. Finally, we claim that, to the best of our knowledge, Revita is currently the only platform for learning/tutoring beyond the beginner level, that is functional, freely-available and supports multiple languages.

Keywords: Computer-assisted language learning, CALL, intelligent tutoring systems, ITS, second language acquisition, SLA, foreign language acquisition, FLA, learner corpus.

1. Introduction

Over the recent years we observe a clear emerging trend toward flexible online language-learning tools that are accessible anywhere on demand. Despite the growing need for and popularity of such tools, most of the existing systems do not address the fundamental requirements of language learners and teachers.

In this paper, we demonstrate Revita, a freely available online platform, which has been designed to support language learning/tutoring *beyond the beginner level*. A variety of resources exist on the Web—various free and commercial applications—which support beginners, some with millions of users. However, once the learner has passed the beginner's level, and reached low-intermediate to advanced (LI-A) level (i.e., above A1 on the CEFR scale), resources available to her become drastically limited. As far as our research has shown, no systems today provide substantial support for LI-A learners in multiple languages.

Revita currently works with several languages—in various stages of development, ranging from initial, "beta" versions to fairly well developed ones. The languages include "big" languages—Finnish, Russian, Swedish$^{(\beta)}$, German$^{(\beta)}$, and Kazakh$^{(\beta)}$—and several endangered minority languages, currently Komi-Zyrian, Udmurt, Meadow Mari$^{(\beta)}$, Erzya$^{(\beta)}$, North Saami$^{(\beta)}$, Komi-Permiak$^{(\beta)}$, and Sakha (Yakut)$^{(\beta)}$.[1]

The system automatically generates a wide variety of randomized exercises targeted for the learner, based on arbitrary, *real* texts, which can be chosen and uploaded to the platform by the learner herself (or by her teacher). The system aims to adapt the level of exercises to every user depending on her level of competence, which it tries to estimate based on her answers to previously completed exercises.

Revita lies at the intersection of two established areas of research: intelligent tutoring systems (ITS) and computer-assisted language learning (CALL)—the project seeks intelligent solutions for language learning. On the other hand, Revita has the potential for enriching the language teaching process as well, because the platform can be used for collecting, mining and analyzing educational data.

The paper is organized as follows: Section 2. presents an overview of previous work in computational approaches to language learning. Section 3. describes the Revita system, a language-learning tool developed at the intersection of ICALL and ITS; Section 3.2. describes the main features of the system. Section 4. positions Revita in the field of Educational Data Mining (EDM), and Section 5. concludes with pointers for future work.

2. Prior work

The idea of using computers for language learning was introduced over 50 years ago. Computer-assisted language learning (CALL) is an active research area, which originated as a sub-area of computer-assisted instruction (CAI). CALL is broadly defined as "the search for and study of applications of the computer in language teaching and learning," (Levy, 1997). CALL includes a broad variety of technologies applied to language learning and teaching, in which the computer is accepted by researchers and by teachers as a *support* tool; it is not considered to be a *replacement* for the teacher. One of the early CALL systems, PLATO, (Hart, 1981) was created in the early 1970s.

Most of the major, popular CALL systems today are commercial, and are intended for learning/tutoring at the elementary/beginner level, viz., Duolingo, LinguaLeo, Babbel, and others. Those systems that claim to cater to advanced learners, typically offer a collection of learning materials and exercises, e.g., `https://www.deutsch-lernen.com/`. Crucially, most of these systems offer materials in the form of a closed, *fixed* set of exercises.

Exercises may be of several types. A type of exercise, known as a "*cloze*" (deletion) test,[2] is where a portion of the text has some of the words removed, and the learner is asked to recover the missing words. Clozes

[1] The system is online at `revita.cs.helsinki.fi`

[2] In the literature, first described in (Taylor, 1953).

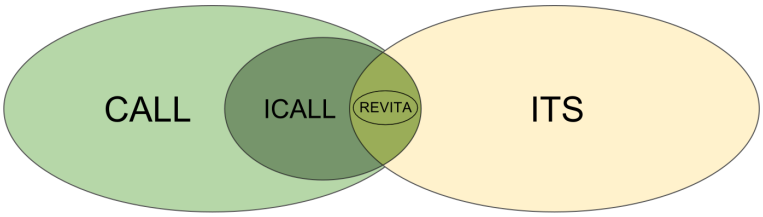


Figure 1: Landscape of tools for language learning.

require an understanding of the context, semantics and syntax in order to fill in the blanks with the correct missing words. Alternatively, exercises may offer selection from a multiple-choice menu.

Thus, most of systems in the literature that we have reviewed, are based on the so-called *"canned"* approach: i.e., exercises that the learner receives are drawn from a pre-made, *fixed*, and therefore limited set—even if the set may be large and varied. The exercises are created in advance of the study session by human experts, and are based on fixed textual material, pre-determined by the designers of the exercises , rarely expand on a context bigger than one sentence.

Overall, however, the existing systems are mainly targeted toward beginners. Since hand-crafting a substantial volume of exercises for more advanced users requires much greater resources, it is more difficult to provide sufficient coverage to support the intermediate/advanced learner, due to the growing cost-to-benefit ratio: the demands on the systems grow, while the market shrinks—many more learners/consumers of language learning services are at the beginner, elementary level, than at the LI-A levels. This ratio is a key factor that has suppressed wider emergence of CALL/ITS beyond the beginner levels.

On the other hand, offering a fixed, limited set of exercises is in conflict with the principles of *adaptability* of the learning process to the profile of the particular user. Applying a "one-size-fits-all" paradigm to students with different levels of competence goes against established pedagogical practices.

Some systems do track personal progress and the user's state of knowledge; however, they do not *adapt* future exercises to the user depending on her prior errors; the set of exercises is the same for everyone.[3]

Further, most systems provide poor feedback about about the answers given by the learners. In the extreme, some systems provide to the learner only a summary of how many of her answers were correct, without any more specific information to help correct errors and improve in the future.[4]

Many of the CALL systems that do attempt to address the needs of intermediate-to-advanced learners employ complex linguistic or grammatical terminology[5]. This may be undesirable in certain settings, e.g., in cases where the learners may have quite substantial competency in the language, yet lack proficiency in linguistic concepts and terminology. This is especially true, e.g., of *"heritage"* learners; these are learners who acquired passive competency in a language from hearing parents/relatives speaking in diasporic settings, or in the settings of minority languages having no official state status—but who have no formal training in it.

As CAI developed, the field of intelligent tutoring systems (ITS) emerged with somewhat different goals. Within ITS another role for the computer was accepted—specifically, the role of computer as *tutor*.

ITSs have been applied in various knowledge domains—mathematics, the sciences, business management, etc.—and focus on the ability to generalize and apply knowledge to specific tasks, and on dynamic adaptation depending on the user and the performed tasks. Initially, ITSs were envisioned as aiming to replicate the function of human teachers (Self, 1990; Shute, 1991)—not only to support learning, but also to attend to the learner's progress, (Self, 1998).

Decades of research have shown the effectiveness of ITS technology for learning. Students who used ITSs showed improvements in performance, compared to students who were exposed only to traditional classroom settings, (Kulik and Fletcher, 2016). Several systems, which were used by tens or hundreds of thousands of students annually, significantly improved learning performance: e.g., Cognitive Tutor (Oxman et al., 2014; Ostrow and Heffernan, 2014), ALEKS (Craig et al., 2013), ASSISTments (Koedinger et al., 2010).

Most ITS approaches found in practice are simple; e.g., they employ simple heuristics for assessment of user progress based on the number of correct answers given in succession (Heffernan and Heffernan, 2014), and they do not attempt to model the underlying conceptual domain. Several seminal approaches have been developed to provide highly refined assessment, based on theoretical psychometric principles—e.g., the Knowledge Space Theory (KST), (Doignon and Falmagne, 2012; Falmagne and Doignon, 2010; Falmagne et al., 1990)—and incorporated into large-scale, commercial science-tutoring systems, such as ALEKS (Craig et al., 2013).

In KST the learner's competency is modeled not as a scalar—e.g., as the A1-C2 "levels" on the CEFR scale—but rather as a position in a complex *knowledge space*, which is modeled as a graph containing many

[3]For example, `https://learning.lengalia.com`

[4]For example, `http://icelandiconline.is/`

[5]For example: `learning.lengalia.com` , `icelandiconline.is` , `www.easypersian.com`

(possibly millions of) nodes. Each node in the graph represents a possible knowledge state of a learner—encoding the set of concepts mastered in that state—and the possible paths in the graph indicate the possible paths toward acquiring full competency in the domain. More precisely, the learner's competency is modeled as a probability distribution over the nodes in the knowledge graph: the learner is most likely to be positioned at those nodes where probability mass is concentrated.

ITS, when applied specifically for language learning—and supported by other intelligent and/or adaptive methodologies, such as adaptive systems (AS), expert systems (ES), natural language processing (NLP), automatic speech recognition (ASR), etc.—defines the domain of intelligent CALL, or ICALL. The goal of ICALL is broadly defined as building advanced applications for language learning using NLP and language resources—corpora, lexicons, etc., (Volodina et al., 2014).

Although ITSs were seen as generally useful in ICALL, they were not in the focus of ITS research over the last two decades, because: they were not considered to be intelligent enough, were felt to be too "behavioral" in nature, and were too bound to "drill-and-kill" types of exercises, (Hubbard and Siskin, 2004); another important reason was the lack of mature technology (Bush, 2008; Mozgovoy, 2012).

Currently, ITSs are acknowledged as useful for language learning, and "able to address more than just simple grammar and vocabulary teaching/learning, but they need to be designed keeping in mind learners' and instructors' needs" (Bush, 2008). We found that most of the existing ITSs for language learning remain in the *prototype* stage, having been developed purely for research purposes. Our literature and online search has revealed only three systems which claim to be functional, and to have been in use in classroom settings:

- E-Tutor, for learning university-level German (Heift, 2008; Heift, 2010)
- TAGARELA, for learning university-level Portuguese (Amaral and Meurers, 2007; Amaral and Meurers, 2011)
- Robo-Sensei, for learning Japanese, mostly focusing on translation (Nagata, 2009)

The first two systems build adaptive learner models, and tailor feedback messages depending on individual performance. The third system has a single sequence of activities distributed over 24 lessons, which is fixed and identical for all users. All of these systems are monolingual—designed for a single language. Most importantly, at the time of this writing, a search on the Web shows that *none* of these systems is currently available for use.

At the three most recent International Conferences on ITS—over the last 6 years—only four papers about language learning have been presented, and none of them relate to developing any new ITSs. Thus, overall we observe a lack of papers about ITSs for language learning, a lack of assessment, and a lack of functioning systems.

Golonka et al. (2014) published a review of computational methodologies in language learning; the review mentions three evaluations of ITS for language learning (Nagata, 1993; Nagata, 1997; Dodigovic, 2007). Although all three showed that ITS are more effective than traditional tutoring, the author was critical of the *evaluation* of the new methodologies.

3. Revita system description

The Revita system, situated at the intersection of ICALL and ITS—see Figure 1—attempts to address the requirements of both, and aims to move beyond existing solutions, (Katinskaia et al., 2017).

3.1. Addressing current problems in CALL and ITS

We briefly review the desiderata of language-learning systems and the key problems in developing ITSs—the main observed pitfalls. Many of these have been brought to light in prior surveys, e.g., in (Slavuj et al., 2015). For each desideratum or problem we briefly mention how Revita satisfies the requirement, or how we intend to address it in the future.[6]

1. *Over-restricting the learning domain horizontally*: Horizontal restriction refers to the tendency to focus on a single linguistic skill. Revita offers a variety of practice modes for exercising reading, grammar and vocabulary skills, to cover a broad range of linguistic concepts and phenomena.[7]
2. *Over-restricting the learning domain vertically*: lack of support for learners at different levels of proficiency (especially CEFR levels A2–C2), typically focusing on the beginner levels. Revita targets the *low-intermediate to advanced* levels (LI-A).
3. *Active vs. passive learning*: i.e., focus on the user's skills in active production of language vs. passive absorption of material. In Revita, the focus is on active skills, achieved by eliciting unrestricted user input; the learner produces word forms in the context of the story.
4. *Learning materials should reflect real-world communicative situations*: We allow the learners/teachers to use arbitrary, *real-world* texts, rather than allowing only artificial texts pre-fabricated for instructional purposes.
5. *Providing learning materials that keep learners motivated* during the learning process. In large part related to item 4, above: if the exercises are detached from real-world, useful contexts, they become uninteresting, and lead to loss of motivation. Revita provides various types of exercises,

[6] See the operational platform at `revita.cs.helsinki.fi`

[7] In the future, we will add exercises in the aural modality, by using text-to-speech tools, which are quite well-developed for many of the target languages.

and entertaining, engaging modes of practice, including competition against an opponent.

6. *Reference to assessment scales*: Most ITSs do not link to a generalized framework of reference, such as the CEFR or ILR scale. Detailed internal assessment is a crucial part of Revita. At present, the system keeps track of the learner's performance on a large set of grammatical and lexical "concepts,"—which serves as an indicator of competency in the corresponding linguistic skills (e.g., "mastered the inflection of nouns of class X"). In future work, we plan to incorporate these concepts fully into the KST framework, and model the knowledge space in accordance with the theory (Doignon and Falmagne, 2012). The model will be learned automatically from observed learner responses (Schrepp, 1999), and the inferred (statistical) precedence relations among the linguistic concepts.
7. *Feedback messages*: should be personalized, and should guide the learner toward finding the correct answer. Rather than responding to the user simply with "correct vs. incorrect", the system should point to the source of the problem and offer a chance to attempt the exercise again using new clues returned by the system. Revita returns immediate feedback to the learner—about grammatical errors, incorrect choices in multiple-choice exercises, or wrong translations in flash-card vocabulary exercises. We plan to extend the feedback mechanisms to be more personalized by analyzing the history of the learner's responses.
8. *Mimicking a language teacher*: ITS is envisioned as being capable to mimicking (and complement) successfully the functions of a language teacher. In order to do so, the ITS must be designed specifically for the purpose of language learning—i.e., based on sound pedagogy and language-teaching methodology, supported by foreign/second language acquisition theory (FLA/SLA). This theory must form the basis for designing intelligent CALL systems. In Revita we work with SLA/FLA experts, to assure adherence to this requirement.

Lastly (also related to problems 5, and 7) when the feedback that the system provides is limited to "right/wrong," the system gives the impression of not caring about the learner's development, and failing to fulfil the requirements of a good tutor.

In summary, CALL systems are seen in the literature as widely available, but not sufficiently "intelligent," nor adaptive. ITSs in the language-learning domain are intelligent, but not available to end-users, or not free of charge. Further, all ITSs that we found are monolingual; ICALL systems are multilingual, but the more languages, the simpler the system, and the more basic the level of the exercises.

The principal characteristics that distinguish an intelligent tutor are: the ability to diagnose the knowledge structure and the skills of the learner; personal adaptation of instruction to the learner; provision of personalized feedback. To meet these requirements, the ITS needs to implement a *(domain) knowledge model*, a *student model* and an *instruction model*.

1. *Domain knowledge model:* the (linguistic) domain knowledge model in Revita is embodied in language-specific rules, which drive the creation of exercises, described below in Section 3.2.. The rule component is specialized for each language,[8] and requires some input from language experts.
2. *Student model:* the system records all of the student's answers (correct and incorrect), which affect the choice of future exercises. These responses will be used to implement KST functionality, to determine learner's current state of knowledge. Both the Knowledge model and the Student model are based on data collected from the students' practice sessions.
3. *Instruction Model:* embodies the pedagogical principles: in which order concepts and exercises should be presented, what feedback to provide, etc. Currently, Revita uses rules as Instruction model to determine which new exercises should be presented to the user based on her Student model. The system then attempts to emphasize practicing those concepts: A. in which the user is not proficient, and B. which the user is best prepared to absorb next. In this process, the exercises turn less frequently—i.e., with lower probability—to concepts: 1. on which the learner previously made very few mistakes (to avoid boredom), and 2. on which she (almost) always makes mistakes (to avoid frustration).

Future work includes incorporating state-of-the-art theories for driving instruction by means of KST and Bayesian knowledge tracing (BKT), (Pardos and Heffernan, 2010).

3.2. Exercise modes

In a broad sense, active language learning is reflected in the functionality of Revita on several levels, including: support active learning by providing the ability to seek out materials to match one's own interests (and sharing interesting materials with friends), as well as selecting categories on which to concentrate in an exercise session, using progress feedback to direct the focus and emphasis of exercises.[9]

The bulk of Revita's exercises are based on the stories the user has selected. One of the bedrocks of Revita's philosophy is that language learning is stimulated and becomes more productive when the learner

[8] For example, in German and Swedish, Revita creates flash-card exercises to test the knowledge of noun gender via their articles—the article is determined by the noun's gender. This is not relevant for gender-free languages.

[9] This feature is helpful for preparation for standardized tests—by specifying a set of topics to focus on.

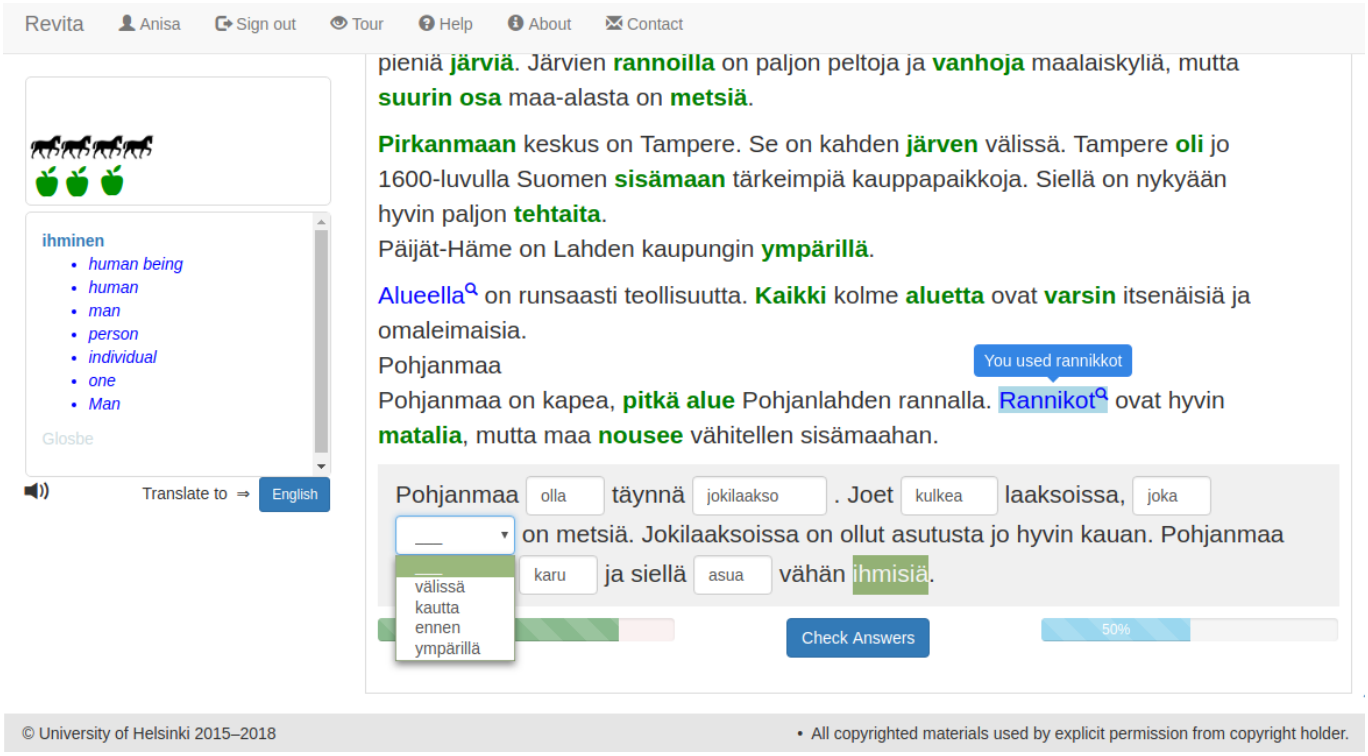


Figure 2: Practice mode. Current snippet of the story (in Finnish) at the bottom, over grey background, contains multiple choice (menu) and cloze (textbox) exercises. Previous snippets above; correctly answered questions in dark green, incorrectly answered questions in blue. User clicked the last word in the current snippet to see its meaning (shown in box on the left).

is working with materials which hold *personal interest for the learner*. The system has a small public library of texts for every language; however every learner is encouraged to upload materials into her own, private library.[10] Learning materials can be shared with other users of the platform ("friends").

Several modes are available for working with a story. The *reading mode* allows the learner to skim through a story and view translations of unfamiliar words—which also adds them to the personal vocabulary (stack of flashcards) for later practice.

In *practice mode* (see Figure 2), the system automatically generates exercises based on a story chosen by the user. The story is presented piece by piece; each piece ("snippet") is about 30-40 words in length, with several words chosen for exercises—these words are omitted and replaced by a multiple-choice quiz, or a "close" quiz. In the figure, the current snippet is at the bottom of the screen (over a grey background).

Previously completed snippets remain above the current snippet (over a white background). In the prior snippets, correctly answered exercises are highlighted in green, and incorrectly answered ones are in blue; hovering over an incorrectly answered exercise provides more insights about the mistake.

For example, "Pohjanmaa *on* täynnä *jokilaaksoja*." is the first sentence (in Finnish) in the snippet in Figure 2, (meaning "Pohjanmaa *is* full *of river valleys*"). The system created cloze exercises by hiding the words in the boxes and providing their base forms (*lemmas*) to the learner as hints:

"Pohjanmaa *olla* täynnä *jokilaakso*."

From the verb lemma "*olla*" ("to be") and noun lemma "*jokilaakso*" ("river valley") the learner should derive the original, correctly inflected forms. The task of the user is to guess which form of the skipped word was used in the context by the author.

Non-inflected words (prepositions, conjunctions, etc.) are used for multiple-choice quizzes. Inflected words (nouns, verbs, etc.) are used for "close" quizzes: the base form (*lemma*) of the word is shown as a hint, and the learner needs to guess the appropriate grammatical form in context.

This approach has its advantages as well as problems. On one hand, it is easy to validate the user's answer, because the system knows the original form used in the story. On the other hand, the learner may possibly insert a different form which is valid in the given context, yet different from the form found in the story. It is a challenging NLP problem to determine whether the user's answer is *also* acceptable—grammatically and semantically—for the given lemma in the given context. This requires a high-precision language model, which is one of the objects of our current research.

[10] This is done by loading a local file (.txt or .doc), by copy-pasting text, or by providing a remote URL of the website containing text to upload into the library.

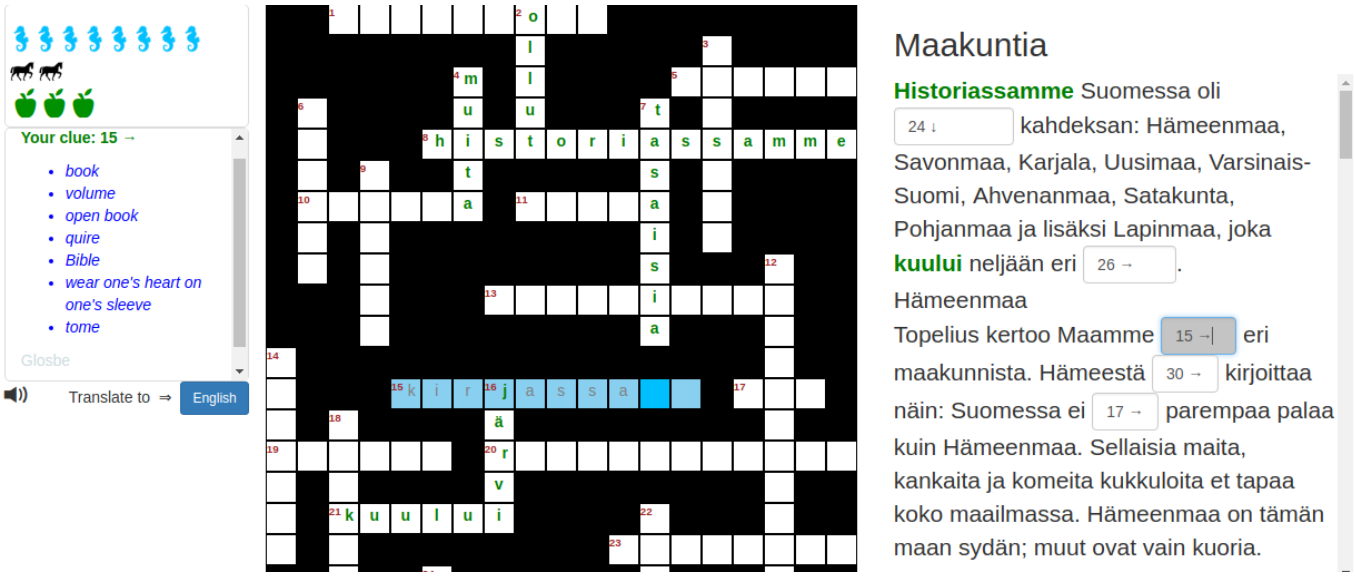


Figure 3: Crossword mode. The current story (in Finnish) on the right contains cloze exercises, in textboxes (exercises that have been resolved are in dark green, e.g., the first word in the story). Each cloze corresponds to entries in the crossword (middle); currently selected entry (15→) is highlighted. The clues to the entries are translations of the missing words (on the left).

After the learner responds to all quizzes in the current fragment of the story, the system provides her with immediate feedback, and then moves on to the following fragment with new exercises. All possible candidates for exercises are determined and saved at the time when the story is uploaded into the system and analyzed by several lower-level natural language processing (NLP) modules including morphological analysis, disambiguation, etc.

Candidates are created using language-specific rules. For example, for Russian, the system may include rule A: [word='в', POS=prep, case={loc, acc}], Another rule, B, in 3 parts, may be:

1. [POS=prep, case=**X**]
2. [POS=adj, case=**X**, number=**Y**, gender=**Z**]
3. [POS=noun, case=**X**, number=**Y**, gender=**Z**].

These (declarative) rules state general facts about the language, and refer to the part of speech tags and morphological tags of words that may be encountered in a story. Rule A states the preposition "в" (meaning "in") governs the locative or accusative case. Rule B stipulates the agreement[11] within a noun phrase that contains the parts of speech (POS) [prep adj noun]: case agreement between preposition, adjective and noun, as well as number and gender agreement between the adjective and noun. Using such rules, the system finds constructions, such as:

"... в тёмном лесу ..."
in dark-Masc.Sg.Loc forest-Masc.Sg.Loc
"... in a dark forest ..."

where the case, number and gender categories agree. Then this sequence may be offered as an exercise, e.g., with a multiple choice for the preposition (with the remaining options filled by *distractor* prepositions), and "cloze" boxes for the noun and adjective. Various approaches to the problem of generation of reliable distractors are described in (Lee and Luo, 2016; Correia et al., 2010; Rakangor and Ghodasara, 2014; Sakaguchi et al., 2013; Hill and Simha, 2016; Liang et al., 2017).

The actual choice of candidates for exercises depends on the user model—the history of answers given previously by the user. The system computes weights (probabilities) for all potential candidates in the story snippet; words or constructions receive a lower weight if they had been answered mostly incorrectly (or mostly correctly) in previous sessions—since that implies that they were too difficult (or too easy) for the learner at present. Candidates with smaller weights appear with a lower probability in the next exercises. Weights are assigned not only to particular words, but also to their various grammatical categories; thus, if a user, e.g., sometimes makes mistakes in a certain nominal case, the system will provide more exercises for this case.

The *crossword mode* encourages practicing grammar and vocabulary simultaneously. A crossword is built automatically and randomly based on the text of a story (see Figure 3). Candidates for the crossword are selected according to the same principles as in practice mode (above); hidden words need to be inserted back into the story in the correct inflected form. As clues, the learner receives the translations of the missing words (rather than their lemmas, as in practice mode). All answers are saved and used for updating the weights of candidates on subsequent exercises, as is done for the practice mode.

In all modes—reading, practice, crossword—the

[11] By unification over the variables, which are in bold.

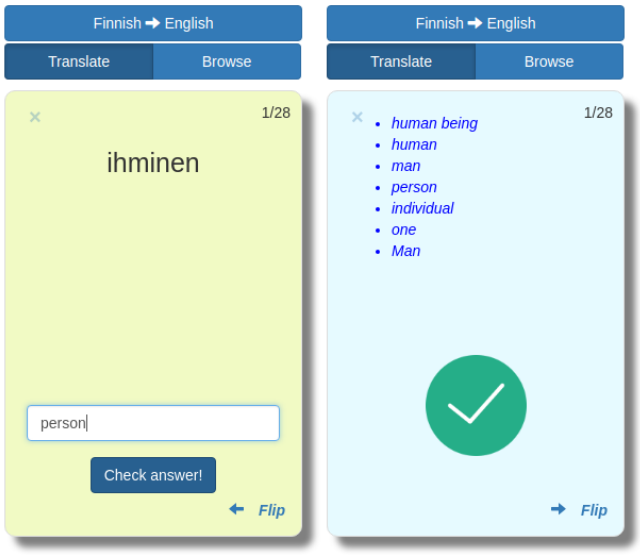


Figure 4: Flashcards for practicing vocabulary.

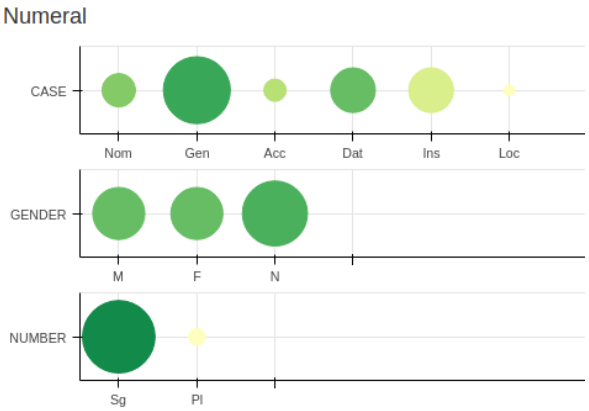


Figure 5: Progress visualization for numerals in Russian

learner can request a translation of any unfamiliar word. Translations are looked up in third-party on-line multi-lingual dictionaries. Words for which translations were requested are added into the user's stack of flashcards—used in the *flashcard* mode, see Figure 4, to practice vocabulary. In the figure, the user first receives the word to be translated (left), types a translation in the box below, flips the card and receives feedback (right) whether the translation is among those that are recognized by the system.

For all of the above exercise modes—reading, practice, crossword, flashcards—Revita provides a *Competition mode extension*.[12] For example, for the practice mode, the idea behind the competition is that the user must guess the correct grammatical word given the context, as before, but now she is simultaneously racing to answer the questions faster than an opponent, who is solving the same exercises at the same time. The learner needs to get more correct answers in a shorter period of time to win the competition. At present, the opponent is a *bot*, which aims to imitate as nearly as possible the learner's own rate of answering questions and state of knowledge (probability of answering the given question correctly). In this way, the learner is *competing with herself*—i.e., trying to improve on her own previous performance.[13]

These exercise modes—all building on the original story text selected by the user—provide a variety of interactions, which help the user practice on one's own, outside the classroom, and which are designed to stimulate and engage the user—more so than a traditional textbook might.[14]

[12] At present, competition is implemented only the practice mode, but the idea is straightforward and the extension will be implemented for all exercise modes.

[13] We plan to implement competition with a human friend as well—another user on the platform.

[14] In the future, the goal is to incorporate exercise modes for reading and aural comprehension, and for speaking.

3.3. Assessment

Currently the system implements a simple system for progress assessment: it checks all of the learner's answers, tracking how exercises involving various grammatical concepts—which are categories, such as tense and mood for verbs, number and case for nouns, etc.—were answered by the learner, and how many exercises with different concepts were practiced.

The learner may keep track of her progress via a visualization (see Figure 5). The size of the balls indicates the relative frequency with which this grammatical concept was encountered by the user in exercises so far; the color ranges from green to red, depending on the percentage of correct answers given so far; if the majority is correct, the color tends toward green, otherwise toward red. This visualization is one way to summarize the learner's knowledge state at the present time; we plan to extend this visualization with information about progress across time.

3.4. Evaluation by learners and teachers

To date, we have introduced Revita to professional teachers of 12 different languages, who provided extensive and useful feedback about their particular language. All of them have stated an interest in using the system in their pedagogical practice.

We are currently in the process of testing the system with students. These initial tests focus on LI-A learners of Finnish and Russian, with wider testing planned to follow.

4. Revita as a source of educational data

Educational data mining (EDM) is defined as the "research field concerned with the application of data mining, machine learning and statistics to information generated from educational settings (e.g., universities and intelligent tutoring systems)," (Baker and Yacef, 2009). The goal of EDM is to help predict the learner's behavior—which means deriving accurate

Student models from user's feedback; the same feedback can be used to improve the Domain models inside the system. More generally, EDM also allows us to study the effects of the educational support provided by the system, and to advance scientific knowledge about the process of language learning and acquisition.

Thus, the final aspect of Revita to which we draw attention in this paper is that as the platform is used by the learners, over time it builds a rich and valuable resource: the detailed educational data in the form of learner responses—correct and incorrect,—the contexts of the responses within the stories, all words which were unfamiliar in their contexts (i.e., for which translations were requested)—as well as the timings of all of these interactions.

This is valuable data, no less valuable than learner corpora, which language educators have worked very hard to collect for decades. The larger a learner corpus grows, the better it allows us to build accurate models of the learners and the learning process, to identify the common mistakes, and the patterns of mistakes. This information allows the system to derive more accurate Domain, Student and Instruction models.

The Student model for each user is based directly on the totality of the data collected for that user. Important aspects of the Domain model are implicit in—and can be inferred from—generalizations over data collected from many learners. For example, if we consider two concepts, A and B, and over a large number of learners we can observe that those who answer B correctly also (almost always) answer A correctly, but not vice versa, then we can infer that, with high probability, concept A is a pre-requisite for B.

Further, looking for patterns of the *order*, in which concepts are mastered by students—over a large population of students—we can obtain statistical information about how the learner's native language affects the order in which concepts in the target language are likely to be acquired. It seems intuitively clear that a group of learners with a common native language will likely exhibit common traits—observable through their responses.[15]

It is also reasonable to assume that having the ability to model these commonalities explicitly, and exploit them in the learning process (in the Instruction model) will provide for a more intelligent tutoring system.

The Instruction model is currently computed by rules, which operate on the data stored in the learners' history. The same data will form the foundation for more complex Instruction models, which we plan to explore, viz., Knowledge Space Theory, and Bayesian knowledge tracing.

It is clear that the educational data collected through Revita over time will find a variety of purposes and important applications. It will impact various aspects of our overall enterprise—namely, gaining insight into the process of language learning, and improving the methodology of automated language instruction and tutoring.

High-quality, detailed and large-scale data is a primary requirement for the application of modern data-driven machine learning and statistical methods. The key point, however, is that *without* a powerful, automated platform such as Revita, data on a comparable scale would be impossible to collect.

[15]For example: native speakers of Turkic languages may find some aspects of learning Finnish (the target) easier than native speakers of Slavic languages—vowel harmony is one such aspect, because it is present in similar forms in both Finnish and Turkic. On the other hand, other aspects of Finnish may be much more intuitive to Slavic speakers, because historically Finnish has been in more immediate contact with Slavic languages, and the development of many of its linguistic features was affected through this contact.

5. Conclusion

An in-depth literature review and investigation of the currently available tools indicates that ICALL systems do not exhibit sufficiently intelligent characteristics, and ITS systems described in the literature appear to be laboratory experiments, not available to learners in practice.

We have presented Revita, an open platform developed at the intersection of ITS and CALL, aiming to address all of the main drawbacks of existing language-learning systems.

Revita aims to bring intelligent language technology into the freely available online language-learning space, which is becoming more popular, not only among the learners, but also among the teachers. It also serves as an instrument for collection of valuable educational data.

The functionality of the system is under continual development. Modifications to Revita are motivated by feedback from the users and SLA/FLA experts.

6. Acknowledgments

This research was supported in part by the FinUgRevita Project, Academy of Finland, Grant No. 267097.

We are grateful to Kim Salmi, Max Koppatz and Jose María Hoya Quecedo for their contributions to the development.

7. Bibliographical References

Amaral, L. and Meurers, D. (2007). Conceptualizing student models for ICALL. In *User modeling*, volume 2007, page 11th. Springer.

Amaral, L. A. and Meurers, D. (2011). On using intelligent computer-assisted language learning in real-life foreign language teaching and learning. *ReCALL*, 23(1):4–24.

Baker, R. S. and Yacef, K. (2009). The state of educational data mining in 2009: A review and future visions. *JEDM-Journal of Educational Data Mining*, 1(1):3–17.

Bush, M. D. (2008). Computer-assisted language learning: From vision to reality? *CaLICO Journal*, 25(3):443–470.

Correia, R. P. d. S., Baptista, J., Mamede, N., Trancoso, I., and Eskenazi, M. (2010). Automatic generation of cloze question distractors. In *Second Language Studies: Acquisition, Learning, Education and Technology*.

Craig, S. D., Hu, X., Graesser, A. C., Bargagliotti, A. E., Sterbinsky, A., Cheney, K. R., and Okwumabua, T. (2013). The impact of a technology-based mathematics after-school program using ALEKS on student's knowledge and behaviors. *Computers & Education*, 68:495–504.

Dodigovic, M. (2007). Artificial intelligence and second language learning: An efficient approach to error remediation. *Language Awareness*, 16(2):99–113.

Doignon, J.-P. and Falmagne, J.-C. (2012). *Knowledge spaces*. Springer Science & Business Media.

Falmagne, J.-C. and Doignon, J.-P. (2010). *Learning spaces: Interdisciplinary applied mathematics*. Springer Science & Business Media.

Falmagne, J.-C., Koppen, M., Villano, M., Doignon, J.-P., and Johannesen, L. (1990). Introduction to knowledge spaces: How to build, test, and search them. *Psychological Review*, 97(2):201.

Golonka, E. M., Bowles, A. R., Frank, V. M., Richardson, D. L., and Freynik, S. (2014). Technologies for foreign language learning: a review of technology types and their effectiveness. *Computer Assisted Language Learning*, 27(1):70–105.

Hart, R. (1981). Language study and the PLATO system. *Studies in Language Learning*, 3(1):1–24.

Heffernan, N. T. and Heffernan, C. L. (2014). The assistments ecosystem: building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497.

Heift, T. (2008). Modeling learner variability in CALL. *Computer Assisted Language Learning*, 21(4):305–321.

Heift, T. (2010). Developing an intelligent language tutor. *CALICO journal*, 27(3):443–459.

Hill, J. and Simha, R. (2016). Automatic generation of context-based fill-in-the-blank exercises using co-occurrence likelihoods and Google N-grams. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 23–30.

Hubbard, P. and Siskin, C. B. (2004). Another look at tutorial CALL. *ReCALL*, 16(2):448–461.

Katinskaia, A., Nouri, J., and Yangarber, R. (2017). Revita: a system for language learning and supporting endangered languages. In *Proceedings of the Joint 6th Workshop on NLP for Computer Assisted Language Learning and 2nd Workshop on NLP for Research on Language Acquisition at NoDaLiDa*. Linköping University Electronic Press.

Koedinger, K. R., McLaughlin, E. A., and Heffernan, N. T. (2010). A quasi-experimental evaluation of an on-line formative assessment and tutoring system. *Journal of Educational Computing Research*, 43(4):489–510.

Kulik, J. A. and Fletcher, J. (2016). Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of Educational Research*, 86(1):42–78.

Lee, J. and Luo, M. (2016). Personalized exercises for preposition learning. *Proceedings of ACL-2016 System Demonstrations*, pages 115–120.

Levy, M. (1997). *Computer-assisted language learning: Context and conceptualization*. Oxford University Press.

Liang, C., Yang, X., Wham, D., Pursel, B., Passonneaur, R., and Giles, C. L. (2017). Distractor generation with generative adversarial nets for automatically creating fill-in-the-blank questions. In *Proceedings of the Knowledge Capture Conference*, page 33. ACM.

Mozgovoy, M. (2012). Towards WordBricks—a virtual language lab for computer-assisted language learning. In *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on*, pages 229–232. IEEE.

Nagata, N. (1993). Intelligent computer feedback for second language instruction. *The Modern Language Journal*, 77(3):330–339.

Nagata, N. (1997). An experimental comparison of deductive and inductive feedback generated by a simple parser. *System*, 25(4):515–534.

Nagata, N. (2009). Robo-Sensei's NLP-based error detection and feedback generation. *Calico Journal*, 26(3):562–579.

Ostrow, K. and Heffernan, N. (2014). Testing the multimedia principle in the real world: a comparison of video vs. text feedback in authentic middle school math assignments. In *Educational Data Mining 2014*.

Oxman, S., Wong, W., and Innovations, D. (2014). White paper: Adaptive learning systems. *Integrated Education Solutions*.

Pardos, Z. and Heffernan, N. (2010). Navigating the parameter space of Bayesian Knowledge Tracing models: Visualizations of the convergence of the expectation-maximization algorithm. In *Educational Data Mining 2010*.

Rakangor, S. and Ghodasara, Y. (2014). Automatic fill in the blanks with distractor generation from given corpus. *International Journal of Computer Applications*, 105(9).

Sakaguchi, K., Arase, Y., and Komachi, M. (2013). Discriminative approach to fill-in-the-blank quiz generation for language learners. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 238–242.

Schrepp, M. (1999). Extracting knowledge structures from observed data. *British Journal of Mathematical and Statistical Psychology*, 52(2):213–224.

Self, J. (1990). Theoretical foundations for intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, 1(4):3–14.

Self, J. (1998). The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. *International Journal of Artificial Intelligence in Education (IJAIED)*, 10:350–364.

Shute, V. J. (1991). Rose garden promises of intelligent tutoring systems: Blossom or thorn. *Nasa technical report*.

Slavuj, V., Kovačić, B., and Jugo, I. (2015). Intelligent tutoring systems for language learning. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on*, pages 814–819. IEEE.

Taylor, W. L. (1953). Cloze procedure: A new tool for measuring readability. *Journalism Quarterly*, 30.

Volodina, E., Pilán, I., Borin, L., and Tiedemann, T. L. (2014). A flexible language learning platform based on language resources and Web services. In *LREC*, pages 3973–3978.

The Distribution and Prosodic Realization of Verb Forms in German Infant-Directed Speech

Bettina Braun, Katharina Zahner
Department of Linguistics, University of Konstanz, Germany
PO Box 186, 78467 Konstanz, Germany
E-mail: bettina.braun@uni-konstanz.de, katharina.zahner@uni-konstanz.de

Abstract

Infant-directed speech is often seen as a predictor for infants' speech processing abilities, for instance speech segmentation or word learning. In this paper, we examine the syntactic distribution (position), accentuation and prosodic phrasing of German verb forms and discuss that many verb forms are prime candidates for early segmentation: they frequently appear at the start or end of prosodic phrases; if they are not phrase-initial, they are often preceded by closed-class word forms and they are frequently accented (imperative verb forms: 72% of the cases, infinitive verb forms: 82% of the cases). It thus appears that German infants ought to be able to extract verbs as early as nouns, given appropriate stimulus materials.

Keywords: corpus, infant-directed speech, prosody, verbs, German

1. Introduction

Infant-directed speech (henceforth, IDS) is often seen as a predictor for infants' speech processing abilities, for instance speech segmentation or early word learning (e.g., Johnson et al., 2014; Laing, 2017; Laing et al., 2016). Regarding segmentation, infants have been shown to segment nouns from fluent speech earlier than verbs. The ability to extract nouns emerges between 7 and 9 months of age (e.g., Altvater-Mackensen & Mani, 2013; Bartels et al., 2009; Jusczyk & Aslin, 1995; Jusczyk et al., 1999; Kuijpers et al., 1998; Männel & Friederici, 2013; Schreiner & Mani, 2017; Zahner, Schönhuber, & Braun, 2016). When nouns occur after or before a prosodic boundary, segmentation succeeds even at 6 month of age (Johnson et al., 2014; Seidl & Johnson, 2006). Verbs, on the other hand, are segmented only several months later, by 11 months in French (Marquis & Shi, 2008) or 13.5 months in American English (Nazzi et al., 2005) – this asymmetry is observed even when verbs in the experiments have the same number of syllables and stress patterns as the noun test words. There are two frequent arguments in the literature for why nouns are segmented earlier than verbs (syntactic framing and prosodic marking). In the Introduction, we will evaluate these two arguments for the case of German drawing on German infant studies, where available, and with regard to the grammatical structure of German. Then we derive predictions for the prosodic realization of German verb forms in IDS. In the main part of the paper, we analyze these forms in the KIDS Corpus (Zahner, Schönhuber, Grijzenhout, et al., 2016).

Argument 1. Nouns are easier to recognize than verbs, since nouns occur in more restricted syntactic frames than verbs (see Höhle et al., 2004, for German; Willits et al., 2014, for English). For instance, Höhle et al. (2004) analyzed a corpus that comprises recordings of 16 mothers talking to their children aged 18 to 27 months, in total 15.000 words. They reported that indefinite articles were often followed by nouns (70% of the cases), while the personal pronoun *sie* and a verb co-occurred in only around 30% of the cases. Children have been shown to rely on such distributional information for syntactic categorization from around 12 to 16 months (Höhle et al., 2004; Mintz, 2006). Specifically, German 16 month-olds use determiners to correctly categorize words as nouns even when the kind of determiner was different during familiarization and test (Höhle et al., 2004), making it necessary to generalize. Thus, syntactic frames might assist syntactic categorization (Mintz, 2003) and segmentation (Willits et al., 2014, for verbs in frequent ing-constructions in English).

Syntactically, German is a V2 language, with the finite verb in second position. In German main clauses and *wh*-questions, the finite verb appears in second position (V2, 1a), in polar questions and imperatives in initial position (V1, 1b) and in subordinate clauses in sentence-final position (1c).

(1a) **Declarative main clause, with/without modal**
Peter **isst** eine Banane.
Peter eats a banana.
'Peter is eating a banana.'
Peter **will** seine Banane essen.
Peter wants his banana eat.
'Peter wants to eat his banana.'
***wh*-question**
What **isst** Peter?
What eats Peter?
'What does Peter eat?'

(1b) **Polar question**
Isst Peter eine Banane?
Eats Peter a banana?
'Does Peter eat a banana.'
Imperative
Iss deine Banane!
Eat your banana.
'Eat your banana!'

(1c) **Main clause, subordinate clause**
Ich bin froh, dass Peter seine Banane **isst.**
I am happy that Peter his banana eats.
'I am happy that Peter is eating his banana.'

Note that German verbs are conjugated depending on person and number (e.g., infinitival form: *essen* 'eat', personal verb forms: 1sg *esse*, 2sg *isst*, 3sg *isst*, 1pl *essen*, 2pl *esst*, 3pl *essen*).

Argument 2. Nouns are easier to recognize than verbs because nouns are prosodically more salient than verbs (accentuation and prosodic phrasing). When comparing the experimental materials used in their verb segmentation study to the materials in an analogous noun segmentation study (Jusczyk et al., 1999), Nazzi et al. (2005) found that the nouns were more often followed by a phrasal boundary than the verbs and were more often preceded by a pitch-accented syllable. The authors argue that this leads to a clearer "perceptual demarcation" for nouns compared to verbs (Nazzi et al., 2005, p. 293). These arguments may well be true for the stimuli with which infants are tested in speech segmentation tasks in the laboratory and also for read speech in English and German (but see also Conwell, 2017, for a recent corpus study on noun/verb homophones in English child-directed speech that showed that nouns are more salient than verbs in phrase-medial position). However, we know very little about the prosodic realization of nouns and verbs in IDS in German. In the adult grammar, verbs are only accented under certain conditions (Féry, 2011; Féry & Herbst, 2004; Gussenhoven, 1983, 1984; Selkirk, 2011), such as in corrective or contrastive contexts where the lexical content of the verb is contrasted (e.g., 'Peter SOLD the book; he didn't buy it'; capitals indicate accentuation), for intransitive verbs when the subject is topical ('Truman DIED'), for transitive verbs when the objects are contextually given and/or pronominalized (e.g., 'Peter BOUGHT it'). In colloquial speech of every-day conversations, verbs are expected to be more often accented, since during the course of the conversation, referents and locations become known to the interlocutors and hence tend to be pronominalized or unaccented (Arnold et al., 2013). While we do not have any analyses of adult conversational speech to back up this claim at the moment, we can test the accentuation in natural infant-directed speech.

Hypothesis. Our hypothesis is that certain verb forms are accented very frequently because conversations with young children are predominantly situated in the *here* and *now* and involve daily routines, objects and events (e.g., Dominey & Dodane, 2004; Papoušek et al., 1991; Werker & McLeod, 1989), which are in turn visually given and consequently unaccented. The verb is then the only word left that can receive the pitch accent.

For the analysis, we use the KIDS Corpus (Zahner, Schönhuber, Grijzenhout, et al., 2016), a multi-layered corpus that consists of 524 intonation phrases (IPs) directed to infants younger than one year, (see Zahner, Schönhuber, Grijzenhout, et al., 2016, for a more detailed description of the corpus; note that the TextGrids of the corpus are publically available on http://ling.uni-konstanz.de/pages/home/braun/KIDS.html - sounds are available on request).

2.2 Annotation

2.2.1 Lexical information and word class

In the KIDS Corpus, all data were labeled using Praat (Boersma & Weenink, 2014) in ten tiers (see Zahner, Schönhuber, Grijzenhout, et al., 2016, for a detailed description of annotation layers). Figure 1 shows the annotation layers that are relevant for the current distribution analysis of verb forms and the prosodic investigation of phrasing an accentuation. Tier 1 shows the orthographic transcription of the utterances. The word class of all words in the corpus was tagged, see tiers 2 and 3. Tier 2 presents broad labels for the syntactic categories, e.g., "adj" (adjective), "adv" (adverb) "conj" (conjunction), "det" (determiner), "int" (interjection) "kop" (copula), "name" (name), "nonw" (nonce word), "noun" and "part" (particle). On tier 3, we used the labels from the Stuttgart-Tübingen-TagSet (STTS, Schiller et al., 1999). For verbs, these are:

- VAFIN: finite auxiliary
- VAPP: past participle of an auxiliary
- VMFIN: finite modal auxiliary
- VMINF: infinite modal auxiliary
- VVFIN: finite lexical verb
- VVIMP: imperative of a lexical verb
- VVINF: infinitive of a lexical verb
- VVPP: past participle of a lexical verb

As described in the Introduction, finite German verbs can occur in different sentence positions. Therefore, we additionally coded the syntactic position of finite verbs as follows: sentence-initial "V1", "V2" and "sentence-final" (or "other" in case the syntactic structure was unclear due to elisions, noise, etc.). Infinitival forms typically occur in clause-final position but may be followed by adverbials, interjections, subordinate clauses, etc. The constituents were also coded, if present.

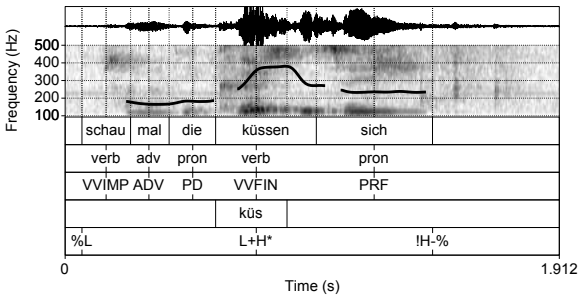


Figure 1: Example utterance from KIDS with smoothed f0 contour and the relevant annotation layers

2.2.2 Prosody

The intonational annotation in KIDS (accented syllable and tonal events, see Figure 1, tiers 4 and 5) was provided by two trained annotators (authors of this paper). Pitch accents and boundary tones were annotated according to GToBI (German Tone and Break Indices, Grice et al., 2005), a widely used annotation system within the framework of autosegmental-metrical

phonology (see, Ladd, 2008, for an overview). The pitch accent and boundary tone inventory as well as specific labeling conventions are described in more detail in Zahner et al. (2015).

For the distributional prosodic analyses of verb forms, we furthermore extracted the position of the verb forms in the respective prosodic phrase. For infinitival forms, which typically occurred in sentence-final position, we coded whether they were followed by a prosodic phrase break (intermediate phrase, ip, or full intonational phrase, IP). For imperatives, we coded their position in the intonational phrase (initial, medial, final).

3. Results

3.1 Syntactic distribution of verb forms

Verbs are the most frequent word class in the KIDS Corpus (verbs: 23%, pronouns: 19%, adverbs: 18%, nouns: 12%). There are 395 verbs, which distribute over the verb classes as follows: 153 finite verbs, 56 finite modal verbs, 77 infinitives, 76 imperatives, 18 participle forms and 13 finite copula verbs.

Of the 153 **finite verbs forms**, there are 69 different verb forms (from 37 lemmas). The most frequent lemmas are *sein* 'to be' (18 times), *haben* 'to have' (17 times) and *tun* 'to do' (12 times). Position-wise, finite verbs mostly occur in V2 (102 times), followed in frequency by V1 (38 times), the final position is rare (11 times). When the finite verb is in second position (V2), the first position is often filled by closed-class elements: most often by a demonstrative pronoun (28 cases, in 7 cases preceded by an interjection, conjunction, or particle). The next frequent categories in the preverbal field were question words (19 times) and adverbs (19 times). Noun phrases were rare (6 times). The syntactic positions of **finite modal verbs** (17 times *können* 'can', 16 times *wollen* 'want', 10 times *müssen* 'must', 9 times *sollen* 'should' and 4 times *dürfen* 'may') are similar to full finite verbs.

Infinitival verb forms occur mostly in utterance-final position (63 times), in 7 cases, the infinite verb is followed by a one-word particle interjection or a name. In the remaining 7 cases, they are followed by phrases such as *ich weiß* 'I know', by syntactic right-locations *das vielleicht haben, die Rassel* (lit: 'it maybe have, the rattle') or *das schoen zusammendruecken, das Buch* (lit: 'it well press, the book'), by a subordinate clause *muss mal aufpassen, dass ....* ('have to pay attention that... ') or by an adverbial *musst du mal krabbeln lernen bis dahin* ('you have to learn to crawl till then').

The **imperative verb forms** are mostly *guck* or *schau* 'look' (28 and 22 times, respectively) and *komm* 'come' (10 times). Syntactically, they typically occur in utterance-initial position (52 times, 12 times preceded by an interjection, a name, or an object noun), but there are also 24 epenthetic occurrences and 10 occurrences in sentence-final position.

The **participle verb forms** typically occur in phrase-final position (11 times), but they are also followed by particles, adverbial phrases, dislocations, imperatives (*schau* 'look') and other parts of a verbal complex (*bekommen haben* 'have gotten', *rausgesucht hast* 'have found out').

3.2 Prosodic Analysis

Figure 2 shows the absolute number of accented and unaccented verb forms in KIDS (for categories with N > 5); for comparison, nouns in KIDS are accented in 90% of the cases (215 accented, 15 unaccented).

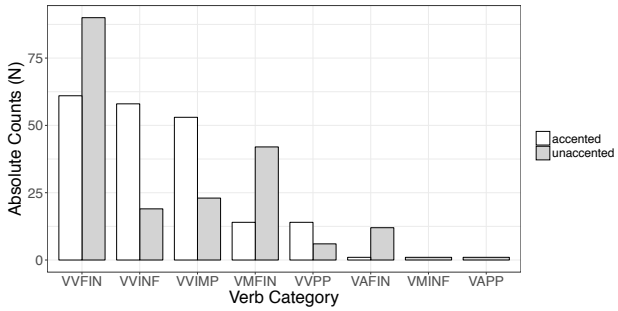


Figure 2: Distribution of accented and unaccented verb types in the KIDS Corpus

Figure 2 shows that **finite verb forms** (full lexical verbs, modal verbs and auxiliaries, VVFIN, VMFIN, VAFIN, respectively) are more often unaccented than accented, while all other verb forms are more often accented than unaccented. A x^2-square test (excluding the infrequent verb classes VAFIN, VMINF, and VAPP) shows that the accented and unaccented verb forms are not distributed equally across verb types ($x^2 = 53.7$, df = 4, $p < 0.0001$). The distribution of GToBI accents for these verb forms is shown in Table 1. Note that the group of finite lexical verbs (VVFIN) also includes the verbs *haben* 'to have' and *sein* 'to be', which, in general, are only accented in forms with polarity contrast (context: He does not have a car. target: He HAS a car, cf. Turco et al. (2014)). When we exclude these two lemmas, the ratio of accented to unaccented finite verbs is more balanced (47% vs. 35%; cf. last row in Table 1).

Accent type	VVFIN	VVINF	VVIMP
unaccented	**59.5% (91)**	**24.7% (19)**	**30.2% (23)**
H*	**10.5% (16)**	**22.1% (17)**	**36.9% (28)**
!H*	5.9% (9)	14.3% (1)	2.6% (2)
L+H*	7.2% (11)	14.3% (11)	13.2% (10)
L*	7.2% (11)	10.4% (8)	5.3% (4)
L* (before H%)	2.0% (3)	5.2% (4)	5.3% (4)
ratio acc:unacc	35% (w/o *have/be*: 47%)	68%	68%

Table 1: Frequent pitch accents across verb forms (bold face indicates most frequent realizations).

Since the finite forms are most variable in their syntactic distribution, we plotted the accentuation of these forms across syntactic position (V1, V2, utterance-final, and other) in Figure 3. This allows us to test whether accented and unaccented finite verbs are distributed

equally across syntactic position. This is clearly not the case ($x^2 = 12.4$, df = 2, $p < 0.006$, Yates' corrected and excluding the category 'other').

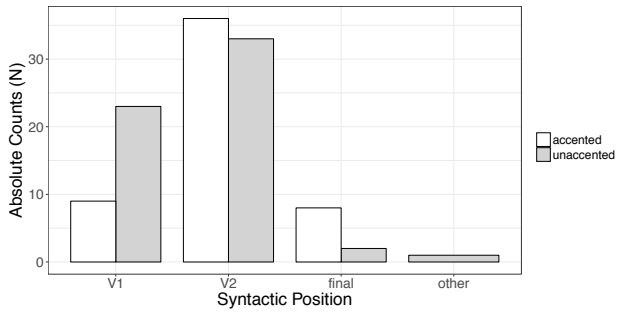


Figure 3: Distribution of accented and unaccented finite forms across syntactic position (without *haben*, *sein*)

Figure 3 shows that finite verbs are often accented in final position (a position that is rare in IDS, as it is restricted to subordinate clauses). In V1 position, the proportion of unaccented finite forms by far outnumbers accented forms, while the distribution is more balanced (or even slightly reversed) for V2 position. Figure 1 and Figure 4 show example realizations of accented finite verbs in V2 position (and an additional accented participle in Figure 4).

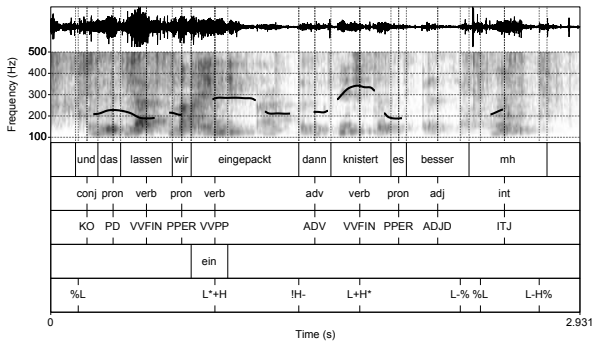


Figure 4: Example realization of an accented finite verb in V2 position

The **infinite verb forms** are twice as often accented than unaccented (Tab. 1) and typically occur at utterance ends, hence being followed by a prosodic IP phrase break (N=63, Tab. 2). In rare cases, infinitival verbs are followed by adverbials, *bitte* 'please', before a dislocation or a name, or without a prosodic phrase break (N=10).

Position of infinitival form	ip-break	IP-break	no phrase break
before adverbial	0	0	3
before bitte 'please'	0	0	1
before dislocation	0	0	1
before interjection	1	0	0
before (pet)name	0	0	4
before negation	0	1	0
before clause	1	0	1
at utterance end	0	**63**	0

Table 2: Position of infinitival verb form and prosodic phrasing.

Imperative verb forms are also more than twice as often accented than unaccented. They are almost exclusively separated from surrounding material by an intermediate phrase (75 out of 76 times), in 5 instances they constitute a separate intonation phrase (IP).

4. Discussion and Conclusion

In infant-directed speech, certain grammatical structures of a language are overrepresented, others underrepresented, mainly due to shorter utterances and the communication about events in the *here* and *now* (Dominey & Dodane, 2004; Papoušek et al., 1991; Werker & McLeod, 1989). Furthermore, the prosodic realization of word categories may be shifted because of frequent pronominalization, increased prosodic variability, and an increased number of prosodic phrases. Previous research has shown that IDS is beneficial for word segmentation (Floccia et al., 2016; Schreiner & Mani, 2017), but the driving factors have to be determined. Here, we analyzed two factors that may affect processing, the syntactic distribution and prosodic properties of German verb forms. Our distributional analyses show a high proportion of sentences in which the verb appears in second position, which is not unexpected for a V2 language. Interestingly, about half of these verb forms are accented (when we exclude the lemmas *have* and *be,* which are typically unaccented in finite forms). Verbs in V2 position are frequently preceded by closed-class forms, making their occurrence more predictable. Moreover, there are also a considerable number of imperatives and infinitival forms in IDS, which are more often accented than unaccented and, furthermore, occur mostly in phrase-initial and phrase-final position, respectively. In both positions, they are demarcated by a prosodic phrase boundary. Recent research in infant speech segmentation has shown that words are segmented better when accented (Männel & Friederici, 2013) and when occurring at phrase-initial and phrase-final positions (Johnson et al., 2014).

More fine-grained prosodic analyses show that, if accented, imperatives and infinitives are frequently realized with high-toned pitch accents (mostly H* or L+H*), which are perceived as prominent by German adult listeners (Baumann & Röhr, 2015) and are more easily segmented by German infants (Männel & Friederici, 2013; Zahner, Schönhuber, & Braun, 2016). Although these accented verb forms are not the most frequent forms that German infants encounter (finite verbs appear more often), they may qualify as candidates for early speech segmentation and thus provide German infants with a head start for the acquisition of verbs in general.

5. Acknowledgements

The project was partly funded by the Excellence Initiative of the University of Konstanz (LingVisAnn 663/13).

6. References

Altvater‐Mackensen, N. & Mani, N. (2013). Word‐form familiarity bootstraps infant speech segmentation. *Developmental Science,* 16(6), 980-990.

Arnold, J.E., Kaiser, E., Kahn, J.M. & Kyoungsook, K.L. (2013). Information structure: Linguistic, cognitive and processing approaches. *Wiley Interdisciplinary Reviews. Cognitive Science,* 4(4), 403-413.

Bartels, S., Darcy, I. & Höhle, B. (2009). Schwa syllables facilitate word segmentation for 9-month-old German-learning infants. *Proceedings of the 33rd Annual Boston University Conference on Language Development*, Somerville, M.A.: Cascadilla Press, 73-84.

Baumann, S. & Röhr, C. (2015). The perceptual prominence of pitch accent types in German. *Proceedings of the 18th International Congress of Phonetic Sciences*, Glasgow, UK.

Boersma, P. & Weenink, D. (2014). Praat: Doing phonetics by computer. Version 5.3.84 [Computer program].

Conwell, E. (2017). Prosodic disambiguation of noun/verb homophones in child-directed speech. *Journal of Child Language,* 44(3), 734.

Dominey, P.F. & Dodane, C. (2004). Indeterminacy in language acquisition: The role of child directed speech and joint attention. *Journal of Neurolinguistics,* 17(2), 121-145.

Féry, C. (2011). German sentence accents and embedded prosodic phrases. *Lingua,* 121, 1906-1922.

Féry, C. & Herbst, L. (2004). German sentence accent revisited. In S. Ishihara, M. Schmitz & A. Schwarz (Eds.), *Interdisciplinary studies on information structure 1* (pp. 43–75).

Floccia, C., Keren-Portnoy, T., DePaolis, R., Duffy, H., Delle Luche, C., Durrant, S., et al. (2016). British English infants segment words only with exaggerated infant-directed speech stimuli. *Cognition,* 148, 1-9.

Grice, M., Baumann, S. & Benzmüller, R. (2005). German intonation in autosegmental-metrical phonology. In J. Sun-Ah (Ed.), *Prosodic Typology. The Phonology of Intonation and Phrasing* (pp. 55-83). Oxford: Oxford University Press.

Gussenhoven, C. (1983). Focus, Mode and the Nucleus. *Journal of Linguistics,* 19(2), 377-417.

Gussenhoven, C. (1984). *On the Grammar and Semantics of Sentence Accents*. Dordrecht, Holland; Cinnaminson, N.J., U.S.A.: Foris Publications.

Höhle, B., Weissenborn, J., Kiefer, D., Schulz, A. & Schmitz, M. (2004). Functional elements in infants' speech processing: The role of determiners in the syntactic categorization of lexical elements. *Infancy,* 5(3), 341-353.

Johnson, E.K., Seidl, A. & Tyler, M.D. (2014). The edge factor in early word segmentation: Utterance-level prosody enables word form extraction by 6-month-olds. *PLoS ONE 9(1):* 9(1), 1-14.

Jusczyk, P.W. & Aslin, R.N. (1995). Infants' detection of the sound patterns of words in fluent speech. *Cognitive Psychology,* 29(1), 1-23.

Jusczyk, P.W., Houston, D.M. & Newsome, M. (1999). The beginnings of word segmentation in English-learning infants. *Cognitive Psychology,* 39(3), 159-207.

Kuijpers, C.T., Coolen, R., Houston, D.M. & Cutler, A. (1998). Using the head-turning technique to explore cross-linguistic performance differences. In C. Rovee-Collier, L. Lipsitt & H. Hayne (Eds.), *Advances in infancy research* (Vol. 12). Stamford: Ablex.

Ladd, D.R. (2008). *Intonational phonology* (Vol. 119). Cambridge [u.a.]: Cambridge University Press.

Laing, C.E. (2017). A perceptual advantage for onomatopoeia in early word learning: Evidence from eye-tracking. *Journal of Experimental Child Psychology,* 161, 32.

Laing, C.E., Vihman, M. & Keren-Portnoy, T. (2016). How salient are onomatopoeia in the early input? A prosodic analysis of infant-directed speech. *Journal of Child Language,* 44(5), 1-23.

Männel, C. & Friederici, A.D. (2013). Accentuate or repeat? Brain signatures of developmental periods in infant word recognition. *Cortex,* 49(10), 2788-2798.

Marquis, A. & Shi, R. (2008). Segmentation of verb forms in preverbal infants. *Journal of the Acoustical Society of America,* 123(4), EL105-EL110.

Mintz, T.H. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition,* 90, 91–117.

Mintz, T.H. (2006). Finding the verbs: Distributional cues to categories available to young learner. In K. Hirsh-Pasek & R. M. Golinkoff (Eds.), *Action meets word: How children learn verbs*. Oxford: Oxford University Press.

Nazzi, T., Dilley, L.C., Jusczyk, A.M., Shattuck-Hufnagel, S. & Jusczyk, P.W. (2005). English-learning infants' segmentation of verbs from fluent speech. *Language and Speech,* 48(Pt 3), 279-298.

Papoušek, M., Papoušek, H. & Symmes, D. (1991). The meanings of melodies in motherese in tone and stress languages. *Infant Behavior and Development,* 14(4), 415-440.

Schiller, A., Teufel, S., Stöckert, C. & Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset) [Technical Report]. Retrieved from. Universitäten Stuttgart und Tübingen [Universities of Stuttart und Tübingen].

Schreiner, M.S. & Mani, N. (2017). Listen up! Developmental differences in the impact of IDS on speech segmentation. *Cognition,* 160, 98-102.

Seidl, A. & Johnson, E.K. (2006). Infant word segmentation revisited: Edge alignment facilitates target extraction. *Developmental Science,* 9(6), 565-573.

Selkirk, E.O. (2011). The syntax-phonology interface. In J. A. Goldsmith, J. Riggle & A. Yu (Eds.), *The Handbook of Phonological Theory* (2nd edition). Oxford: Blackwell.

Turco, G., Braun, B. & Dimroth, C. (2014). When contrasting polarity, the Dutch use particles, Germans intonation. *Journal of Pragmatics,* 62(94-106).

Werker, J.F. & McLeod, P.J. (1989). Infant preference for both male and female infant-directed talk: A developmental study of attentional and affective responsiveness. *Canadian Journal of Psychology,* 43(2), 230.

Willits, J.A., Seidenberg, M.S. & Saffran, J.R. (2014). Distributional structure in language: contributions to noun-verb difficulty differences in infant word recognition. *Cognition,* 132(3), 429.

Zahner, K., Pohl, M. & Braun, B. (2015). Pitch accent distribution in German infant-directed speech. *Proceedings of Interspeech*, Dresden, 46-50.

Zahner, K., Schönhuber, M. & Braun, B. (2016). The limits of metrical segmentation: Intonation modulates infants' extraction of embedded trochees. *Journal of Child Language,* 43(6), 1338-1364.

Zahner, K., Schönhuber, M., Grijzenhout, J. & Braun, B. (2016). Konstanz prosodically annotated infant-directed speech corpus (KIDS corpus). *Proceedings of the 8th International Conference on Speech Prosody*, Boston, USA, 562-566.

Cross-linguistically Small World Networks are Ubiquitous in Child-directed Speech

Steven Moran, Danica Pajović, Sabine Stoll
Department of Comparative Linguistics, University of Zurich
Plattenstrasse 54, CH-8032 Zurich, Switzerland
{steven.moran, danica.pajovic, sabine.stoll}@uzh.ch

Abstract

In this paper we use network theory to model graphs of child-directed speech from caregivers of children from nine typologically and morphologically diverse languages. With the resulting lexical adjacency graphs, we calculate the network statistics {N, E, <k>, L, C} and compare them against the standard baseline of the same parameters from randomly generated networks of the same size. We show that typologically and morphologically diverse languages all share small world properties in their child-directed speech. Our results add to the repertoire of universal distributional patterns found in the input to children cross-linguistically. We discuss briefly some implications for language acquisition research.

Keywords: network theory, linguistics, corpus linguistics, child language acquisition

1. Overview

Despite the remarkable diversity of linguistic structures in the world's 7000 or so languages, children can acquire any language. This fact presents many questions, including importantly: what are the underlying cognitive mechanisms that enable children to acquire language? And are there universal patterns in the linguistic input to children that potentially bootstrap these mechanisms?

Consider one salient difference among the world's languages (especially the under-studied ones): how words are constructed and how they relate to syntax. When analyzed in detail, it is rather difficult to define what a word is cross-linguistically (Hall et al., 2008). In some languages words represent what English speakers consider full phrases; in other languages the word and morpheme (smallest function bearing linguistic unit) are synonymous. Contrast two utterances from Indonesian (Gil and Tadmor, 2007) and Cree (Brittain, 2015):

(1) O, Ei lagi minum susu.
oh Ei more drink milk
'Oh, Ei is drinking more milk.' (Indonesian)

(2) Chi-wâp-iht-â-n â kâ-pushch-ishk-iw-â-t.
2-light-by.head-TR.INAN.NON3-2SG>0 Q PVB.CONJ-put.on-by.foot-STEM-TR.ANIM-3SG>4SG
'You see? She was putting it on.' (Cree)

Indonesian is an example of a language with a fairly low degree of synthesis, whereas Cree belongs to one of the most genuinely polysynthetic language families of the world (and features both noun incorporation and polypartite stems).[1] Clearly the frequency in which children hear a particular form is a function of synthesis combinatorics (Stoll et al., 2017). That is, in languages where morphology is in a closer one-to-one relationship between word and grammatical function, these forms will occur more frequently in the input. There will be greater transition probabilities in languages with more tokens than in morphologically-rich languages which have more types. Nevertheless, regardless of morphology, children from all languages learn to identify words and to produce them.

For a long time, Universal Grammar (UG) was the answer to such problems in language acquisition. In UG, language is the product of innate functions (Chomsky, 1957), where rules and parameters are hard-wired and the acquisition process involves language-specific tuning of linguistic structures (Chomsky, 2000). Because the language acquisition device is posited as innate, models of UG are not necessarily data-driven, but instead theoretical and mainly focused on 'Language' as an abstract system – centered historically on the syntactic structure of English and a few other major languages.

No matter what theoretical approach researchers adopt, they must explain how children identify patterns in their linguistic input and make use of these in productive generalizations as observed in their linguistic output. Usage-based or constructivist approaches are functionalist in that they take into account the way that language is used and contexts in which linguistic elements appear. Increased access to richly annotated linguistic data and computing power, coupled with approaches particularly in corpus linguistics, have shown that there are discernible distributional and predictable patterns in the input to children. For example, grammatical knowledge can be learned from patterns in CDS (Gegov et al., 2011; Freudenthal et al., 2007; Redington et al., 1998; Cartwright and Brent, 1997). Distributional patterns are also predictors of different grammatical categories to varying degrees, depending on the grammatical properties of the language (Mintz, 2003; Stoll et al., 2009; Stumper et al., 2011; Moran et al., In press).

Gegov et al. (2011) call these *invisible patterns*, which they aim to discover using network theory to model language acquisition data. This line of inquiry is summarized by Vitevitch (2008), Beckage et al. (2011) and Gegov et al. (2011). Networks have many properties that allow us to model,

[1] Another example is verbal inflection: English typically has four forms, e.g. kick, kicks, kicked, kicking. But compare Chintang, a language spoken in rural Nepal. It has more than 4000 inflectional forms per verb (Stoll et al., 2017).

compare and visualize data from vastly different input sources (Mihalcea and Radev, 2011). Network analysis has been applied to various issues in language acquisition (Ke and Yao, 2008; Vitevitch, 2008; Solé et al., 2010; Beckage et al., 2011; Gegov et al., 2011). One area of research into the input that children receive has been to model language-specific child-directed speech as lexical adjacency network graphs (Adamo and Boylan, 2008; Ke and Yao, 2008).

Small-world patterns (low average path length (L) and high clustering coefficient (C), see definitions below) and scale-free structures (power-law degree distribution) purportedly may be the product of language evolution towards an optimal cost-path navigation in the mental lexicon for speech production (Barabási and Albert, 1999; Ke, 2007).[2] This is perhaps not surprising given universal mechanisms of network formation, which as Ke (2007) notes, are "common pattern of life systems at all levels, ranging from food webs studied by ecologists, to the neural systems in the brain studied by neuroscientists which have been applied in computer sciences as artificial neural networks".

Ke (2007) advocates for networks as a means to model and investigate global structures in CDS and notes that convergent features in global networks appear whether or not those networks are encoded with semantic or grammatical relationships. These global structural characteristics reflect principles of self-organization of the lexicon and purportedly facilitate cognitive processing (see Discussion). Until now, only English and Chinese CDS have been investigated in detail with lexical adjacency networks. A cross-linguistic analysis has been presumably absent due to a lack of accessible and interoperable typologically-rich cross-linguistic data from longitudinal child language acquisition corpora.

2. Data and language sample

The ACQDIV corpus consists of ten longitudinal language acquisition corpora from nine languages, which are listed in Table 1.[3] The ACQDIV corpus focuses on the acquisition period from ages two-to-three.

ISO	Language	Speakers	Classification
ctn	Chintang	6K	Sino-Tibetan
cre	Cree	87.2K	Algic
ind	Indonesian	23.2M	Austronesian
jpn	Japanese	128.1M	Japanese
ike	Inuktitut	34.5K	Eskimo-Aleut
rus	Russian	166.2M	Indo-European
sot	Sesotho	5.6M	Niger-Congo
tur	Turkish	70.9M	Altaic
yua	Yucatec	766K	Mayan

Table 1: Language sample

These languages were selected from five clusters calculated via maximum diversity sampling (Stoll and Bickel, 2013) from the AUTOTYP database (Bickel et al., 2017) and from the World Atlas of Language Structures (Dryer and Haspelmath, 2013). The clustering algorithm identifies maximal diversity with respect to several widely-studied typological parameters, including presence and nature of agreement and case marking; word order; degree of synthesis; polyexponence and inflectional compactness of categories; syncretism; and inflectional classes.

In most corpora in ACQDIV, the recording sessions for each target child took place every other week (two corpora are much denser). Session lengths vary both within and across corpora and range from half an hour to four hours. All recording sessions were transcribed and morphologically glossed.[4] The size of the corpora also vary considerably, as shown in Table 2.

Corpus	Utterances	Sessions
Chintang	393030	477
Cree	20648	25
Indonesian	915759	997
Inuktitut	46683	77
Japanese	437348	362
Russian	827589	450
Sesotho	69575	115
Turkish	401262	373
Yucatec	93185	125

Table 2: Corpus size

Each corpus was developed and coded independently. Six corpora are encoded in CHILDES/CHAT or TalkBank XML (MacWhinney, 2000). Three are encoded in SIL's Toolbox in project-specific schemas. In recent work we describe how we transformed these different data formats into a single uniform and normalized database (Moran et al., 2016), which we query, analyze and visualize with various tools including SQLite,[5] R (R Core Team, 2013) and Python (using Networkx (Hagberg et al., 2008), NLTK (Bird et al., 2009), Numpy (Walt et al., 2011), Pandas (McKinney and others, 2010), SciPy (Jones et al., 2001))[6] and Gephi (Bastian et al., 2009).

3. Method

A network is a graph data structure that consists of vertices and edges (nodes and links), or formally: G = (V, E) where V is a collection of vertices, $V = \{V_i, i = 1, n\}$, and E is a collection of edges over V: $V, E_{ij} = \{(V_i, V_j), V_i \in V, V_j \in V\}$. For a thorough description of graphs and graph types in regard to network analysis with child language acquisition data, see Pajovic (2016).

[2] Recent work by Broido and Clauset (2018) shows that scale-free networks are actually rare across scientific domains. Whether the scale-free property exists in lexical adjacency networks of child-directed speech should be investigated.

[3] Three letter language name identifiers are ISO 639-3 codes. Population figures are from the Ethnologue (Lewis et al., 2009).

[4] Additional annotation layers of interest such as utterance-level translations, time stamps for the beginning and end of utterances, coding for addressees, morpheme segmentation, and part-of-speech tags are available for all corpora to various degrees.

[5] https://www.sqlite.org/

[6] https://www.python.org/

We create each lexical co-occurrence graph by splitting utterances on white space characters to delimit unique word forms as they are transcribed by experts for each language in our sample. Unique word forms represent the nodes in a network. A link is placed between two nodes if they directly co-occur within an utterance. An example is given in Figure 1. The size of the nodes in this example is determined by the degree of the node, i.e. the more links a node has, the bigger it is drawn. This network was produced from these two example utterances from Russian, below:[7]

(3) a. *xxx* *idi* *mjachik*
M.SG.NOM.AN go.IPFV.IMP.2SG ball.M.SG.ACC.INAN
narisuju.
draw.PFV.NPST.1SG
xxx come I'll draw you a little ball.

b. *Idi* *mjachik* *narisuju.*
go.IPFV.IMP.2SG ball.M.SG.ACC.INAN draw.PFV.NPST.1SG
Come I'll draw you a little ball.

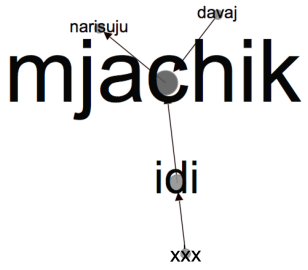


Figure 1: Lexical adjacency network for two Russian utterances

This example illustrates that nodes appearing multiple times in the data set will only appear as a single node in the graph network. If an utterance only consists of a single word (and if this word never appears in any multi-word utterance), this word will be placed as a 'lonely' node in the network. This procedure actually creates a so-called *multi-digraph*, because it allows multiple links from a source to a target node. We use the graph library NETWORKX (Hagberg et al., 2005) to convert multidigraphs into *weighted digraphs* in which the weight of edges correspond to the frequency of the edge from target to source node.

For each language, we create a single network graph that models pooled child-directed speech from all caregivers to children between the ages of two and two-and-a-half. For example, our Russian CDS network contains all the pooled utterances from 24 adults. We think this concatenation is necessary given that the actual amount of speech children are exposed to is much greater than the small fraction of the actual input each child receives during the timespan under observation. Further we are not concerned here with the output of the children, but instead what input a hypothetical learner will encounter during the language acquisition process. Therefore we also include child-surrounding speech when it is available.

Research applying network theory to questions in child language acquisition has typically focused on three types of networks: co-occurrence, syntactic and semantic, and network parameters with high coverage (for a summary, see Gegov et al. (2011) and Pajovic (2016)).

For each language in our sample, we create a lexical adjacency network and load it into R. We use the IGRAPH libraries (Csardi and Nepusz, 2006) to calculate:

- N: the total number of nodes
- E: the total number of edges
- <k>: the key parameter, i.e. the average number of links adjacent to a node
- L: the average length of the shortest path between all pairs of nodes (average geodesic length)
- C: the clustering coefficient, i.e. the likelihood of neighbors of a node being connected, averaged across all nodes

The first three statistics are straightforward. The fourth consists of the short average path length (L) from one node to every other node. It is the most characteristic feature of small-world networks. Here 'small' refers to the fact that any two nodes in such a network can be reached through a few intermediate nodes (Watts and Strogatz, 1998; Ke, 2007; Milgram, 1967).

The fifth network statistic that we measure is the clustering coefficient of a node V_i. It is defined as the number of all edges between all of the nodes in a connected neighborhood divided by the total number of possible edges in the entire neighborhood of V_i. A clustering coefficient of 0 suggests that no neighbor of a node is connected to the other neighbors of that node; 1 means that all neighbors of a node are connected to each other. Values between 0 and 1 imply that there is a number of neighbors of a node which are also neighbors of each other (Watts and Strogatz, 1998; Mihalcea and Radev, 2011). Compared to random networks of equal size, small-world networks have a much higher clustering coefficient (Ke, 2007).

To evaluate each statistic, we use the standard approach of constructing random graphs, then we calculate their statistics for the five parameters above, and then we compare the two sets. To generate random networks, we use the Erdős-Rényi *G(n,p)* model, where *n* is the number of nodes in the network we want to compare (e.g. the Russian CDS lexical adjacency graph), and *p* is the probability of edge creation (also calculated from each lexical adjacency graph). Our random networks are directed graphs and we calculate *p* as $2m/(n(n-1))$ where *m* is the number of edges in our input.

4. Results

Our results are given in Table 3. The networks constructed from CDS all show small-world characteristics: their number of edges are greater than in the randomly-generated graphs; degree is higher in the random graphs; the principally short average path lengths are similar as in the random

[7]The child's name has been anonymized in this example.

graphs; and the clustering coefficient is much higher in the CDS networks than in the random graphs.[8]

When we compare the networks across languages, Inuktitut has the smallest number of edges and also the smallest node degree, but the highest average path length. This is in-line with linguistic expectations given Inuktitut's regular agglutinative morphology; there are few combinations of bigrams delimited by white space. On the other hand compare Indonesian, which has a higher key parameter (the number of connections a node has). This finding is also in-line with linguistic expectations. As illustrated above, Indonesian's morphology is isolating and words are combined much more frequently than in the morphologically more complex languages in our sample. Overall, we see interesting differences between the network parameters in Table 3 that reflect differences in the typological structures of languages in our sample, which we plan to explore in detail in future work.

5. Discussion

Although the *ability* to learn language is held to be innate, non-nativist and input-based approaches to language acquisition theorize that children are not born with grammatical categories or rules, but acquire them by generalizing from the CDS that they hear. Hence grammatical categories may be so-called emergent, that is, they emerge during the language acquisition process (e.g. Tomasello (2009), Cohn (2011), and Theakston and Lieven (2017)) and are not hardwired into our genetics.

Therefore one area of important research is to examine CDS from typologically maximally diverse languages and to identify distributional patterns in the input that appear cross-linguistically. Network theory is one tool for modeling CDS and for mining patterns in it.[9]

In this paper we show that typologically diverse and morphologically very different languages all exhibit small-world network properties when we model CDS as lexical adjacency graphs. Our finding is in line with child language acquisition models that have defined network links in terms of semantic or grammatical relationships, both of which exhibit convergent features in their global structures (Ke, 2007), but of course more work is needed, cf. Telesford et al. (2011).

What we have not shown and cannot answer at this point is whether distributional patterns facilitate cognitive processing. This is of course a key question in cognitive science and beyond the scope of this paper. Regarding small world characteristics, it is not difficult to imagine how their characteristic properties, including efficient information transfer and properties of regional specialization, could account for universal properties like fast retrieval from the mental lexicon. However, more substantive work is needed to show for example that small world properties constrain memory models to facilitate retrieval, e.g. Reitter and Lebiere (2012). Nevertheless, to answer whether general-purpose mechanisms are involved in language learning, we need to also know what distributional regularities exist in languages cross-linguistically, so we can determine on which mechanisms they might operate.

[8] Interestingly, these properties are found in the networks created from the utterances produced by each child. When compared to the adults' graphs, the children's graphs show much lower average degrees, but this is to be expected as the vocabulary size of the children is much smaller. We will follow this line of research elsewhere.

[9] Critics note that graphs and matrices (or data tables) are interchangeable data structures and that networks provide no additional benefits than matrices. They are wrong, however, because networks can also be visualized and therefore provide researchers with additional tools and techniques for exploratory analysis, e.g. network growth models in time-series analysis.

6. Acknowledgements

The research leading to these results has received funding from the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 615988 (PI Sabine Stoll). We gratefully acknowledge Shanley Allen, Aylin Küntay, and Barbara Pfeiler, who provided the Inuktitut, Turkish, and Yucatec data for our analysis, respectively. We also thank three anonymous reviewers for their feedback.

7. Author contributions

SM, DP, SST designed the research. DP, SM performed the research and analyzed the results. SST provided data. SM wrote the paper.

8. Bibliographical References

Adamo, M. and Boylan, S. (2008). A network approach to lexical growth and syntactic evolution in child language acquistion. *Unpublished manuscript*.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.

Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: An open source software for exploring and manipulating networks. In *International AAAI Conference on Weblogs and Social Media*.

Beckage, N., Smith, L., and Hills, T. (2011). Small worlds and semantic network growth in typical and late talkers. *PloS one*, 6(5):e19348.

Bickel, B., Nichols, J., Zakharko, T., Witzlack-Makarevich, A., Hildebrandt, K., Rießler, M., Bierkandt, L., Zúñiga, F., and Lowe, J. B. (2017). The AUTOTYP typological databases. version 0.1.0. Online: `https://github.com/autotyp/`.

Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Brittain, J. (2015). Corpus of the Chisasibi Child Language Acquisition Study (CCLAS). Source: 19-A1-2006-08-16ms.u289. Online: `http://phonbank.talkbank.org/access/Other/Cree/CCLAS.html`.

Broido, A. D. and Clauset, A. (2018). Scale-free networks are rare. *arXiv preprint 1801.03400v1*.

Cartwright, T. A. and Brent, M. R. (1997). Syntactic categorization in early language acquisition: Formalizing the role of distributional analysis. *Cognition*, 63(2):121–170.

CDS	N	E	E_random	<k>	<k>_random	L	L_random	C	C_random
Chintang	60535	173350	347894	5.727265	11.49398	3.861909	6.497079	0.01949274	0.0001666573
Cree	4577	9945	19943	4.345641	8.714442	3.925917	5.89046	0.0354484	0.002096497
Indonesian	26264	257285	516430	19.59222	39.32607	3.174789	3.73446	0.07556786	0.001496866
Inuktitut	11705	7851	15703	1.341478	2.683127	6.714106	23.04485	0.008594746	0.000283802
Japanese	24051	151235	303140	12.57619	25.2081	3.156601	4.261239	0.03455292	0.001044477
Russian	47895	290832	581179	12.14457	24.26888	3.382483	4.599158	0.03299296	0.0004945935
Sesotho	5519	22409	44735	8.120674	16.21127	3.265634	4.363033	0.05016898	0.002802469
Turkish	61537	324164	648276	10.53558	21.06947	3.621385	4.933678	0.03675761	0.0003510219
Yucatec	15967	38638	77134	4.839732	9.661677	4.020815	6.317499	0.02289779	0.0006281727

Table 3: Network parameters comparison

Chomsky, N. (1957). *Syntactic Structures*. Mouton and Co.

Chomsky, N. (2000). *New horizons in the study of language and mind*. Cambridge University Press.

Cohn, A. (2011). Features, segments, and the sources of phonological primitives. In G Nick Clements et al., editors, *Where do phonological features come from? Cognitive, physical and developmental bases of distinctive speech categories*, pages 15–42. John Benjamins Publishing Company.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5):1–9.

Matthew S. Dryer et al., editors. (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Freudenthal, D., Pine, J. M., Aguado-Orea, J., and Gobet, F. (2007). Modeling the developmental patterning of finiteness marking in English, Dutch, German, and Spanish using MOSAIC. *Cognitive Science*, 31(2):311–341.

Gegov, E., Gobet, F., Atherton, M., Freudenthal, D., and Pine, J. (2011). Modelling language acquisition in children using network theory. In B. Kokinov, et al., editors, *European Perspectives on Cognitive Science*. New Bulgarian University Press.

Gil, D. and Tadmor, U. (2007). The MPI-EVA Jakarta Child Language Database. A joint project of the Department of Linguistics, Max Planck Institute for Evolutionary Anthropology and the Center for Language and Culture Studies, Atma Jaya Catholic University. Source: HIZ-1999-05-20.0556. Online: `https://jakarta.shh.mpg.de/acquisition.php`.

Hagberg, A., Schult, D., and Swart, P. (2005). NetworkX: Python software for the analysis of networks. Technical report, Technical report, Mathematical Modeling and Analysis, Los Alamos National Laboratory, 2005. http://networkx. lanl. gov.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA, August.

Hall, T. A., Hildebrandt, K. A., and Bickel, B. (2008). Introduction theory and typology of the word. *Linguistics*, 46(2):183–192.

Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open source scientific tools for Python. Online: `http://www.scipy.org/`.

Ke, J. and Yao, Y. (2008). Analysing language development from a network approach. *Journal of Quantitative Linguistics*, 15(1):70–99.

Ke, J. (2007). Complex networks and human language. *arXiv preprint cs/0701135*.

Lewis, M. P., Simons, G. F., and Fennig, C. D. (2009). *Ethnologue: Languages of the world*, volume 16. SIL international Dallas, TX.

MacWhinney, B. (2000). *The CHILDES project: The database*, volume 2. Psychology Press.

McKinney, W. et al. (2010). Data structures for statistical computing in Python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. SciPy Austin, TX.

Mihalcea, R. and Radev, D. (2011). *Graph-based natural language processing and information retrieval*. Cambridge University Press.

Milgram, S. (1967). The small world problem. *Psychology Today*, 2(1):60–67.

Mintz, T. H. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition*, 90(1):91–117.

Moran, S., Schikowski, R., Pajović, D., Hysi, C., and Stoll, S. (2016). The ACQDIV database: Min(d)ing the ambient language. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Moran, S., Blasi, D. E., Schikowski, R., Küntay, A. C., Pfeiler, B., Allen, S., and Stoll, S. (In press). A universal cue for grammatical categories in the input to children: frequent frames. *Cognition*.

Pajovic, D. (2016). Acqdiviz: Visualising development in longitudinal first language acquisition data. Master's thesis, University of Zurich.

R Core Team, (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Redington, M., Chater, N., and Finch, S. (1998). Distributional information: A powerful cue for acquiring syntactic categories. *Cognitive Science*, 22(4):425–469.

Reitter, D. and Lebiere, C. (2012). Social cognition: Memory decay and adaptive information filtering for robust information maintenance. In *Proceedings of the Twenty-*

Sixth Conference on Artificial Intelligence (AAAI-12), Toronto, Canada.

Solé, R. V., Corominas-Murtra, B., Valverde, S., and Steels, L. (2010). Language networks: their structure, function, and evolution. *Complexity*, 15(6):20–26.

Stoll, S. and Bickel, B. (2013). Capturing diversity in language acquisition research. *Language Typology and Historical Contingency: In Honor of Johanna Nichols. Amsterdam: John Benjamins*, pages 195–216.

Stoll, S., Abbot-Smith, K., and Lieven, E. (2009). Lexically restricted utterances in russian, german and english child-directed speech. *Cognitive Science*, 33:75–103.

Stoll, S., Mazara, J., and Bickel, B. (2017). The acquisition of polysythetic verb forms in Chintang. In Michael Fortescue, et al., editors, *Handbook of Polysynthesis*. Oxford University Press.

Stumper, B., Bannard, C., Lieven, E., and Tomasello, M. (2011). Frequent frames in German child-directed speech: A limited cue to grammatical categories. *Cognitive Science*, 35(6):1190–1205.

Telesford, Q. K., Joyce, K. E., Hayasaka, S., Burdette, J. H., and Laurienti, P. J. (2011). The ubiquity of small-world networks. *Brain Connectivity*, 1(5):367–375.

Theakston, A. and Lieven, E. (2017). Multiunit sequences in first language acquisition. *Topics in Cognitive Science*.

Tomasello, M. (2009). *Constructing a language*. Harvard University Press.

Vitevitch, M. S. (2008). What can graph theory tell us about word learning and lexical retrieval? *Journal of Speech, Language, and Hearing Research*, 51(2):408–422.

Walt, S. v. d., Colbert, S. C., and Varoquaux, G. (2011). The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of small-world networks. *Nature*, 393(6684):440–442.

L1-L2 Parallel Treebank of Learner Chinese: Overused and Underused Syntactic Structures

Keying Li, John Lee
Department of Linguistics and Translation
City University of Hong Kong
keyingli3-c@my.cityu.edu.hk, jsylee@cityu.edu.hk

Abstract

We present a preliminary analysis on a corpus of texts written by learners of Chinese as a foreign language (CFL), annotated in the form of an L1-L2 parallel dependency treebank. The treebank consists of parse trees of sentences written by CFL learners ("L2 sentences"), parse trees of their target hypotheses ("L1 sentences"), and word alignment between the L1 sentences and L2 sentences. Currently, the treebank consists of 600 L2 sentences and 697 L1 sentences. We report the most overused and underused syntactic relations by the CFL learners, and discuss the underlying learner errors.

Keywords: learner corpus, parallel treebank, Chinese as a foreign language

1. Introduction

Learner corpora, which consist of texts written by non-native speakers, are increasingly used in quantitative studies in second language acquisition. Some of these corpora have been annotated to answer various research questions. To support analysis of grammatical mistakes made by learners, a number of them have been error-tagged (e.g., Yannakoudakis et al., 2011; Dahlmeier et al., 2013; Lee et al., 2016). To better characterize learner syntax, others have been part-of-speech (POS) tagged (e.g., Díaz-Negrillo et al., 2010; Reznicek et al., 2013), and syntactically analysed with constituent trees (e.g., Nagata and Sakaguchi, 2016) and dependency trees (e.g., Ragheb and Dickinson, 2014; Berzak et al., 2016).

Building on learner treebanks, Lee et al. (2017b) proposed to use "L1-L2 parallel treebanks" — parse trees of non-native sentences ("L2 sentences") aligned to their target hypotheses ("L1 sentences") — to facilitate analyses of learner language. Figure 1 shows an example tree pair. It includes the parse tree of the learner sentence and of its target target hypothesis, both annotated in the Universal Dependencies (UD) scheme for Chinese (Leung et al., 2016; Lee et al., 2017), as well as word alignments between the two sentences. Such a treebank has the potential to enhance Contrastive Interlanguage Analysis (CIA) (Granger, 2015) and Error Analysis (EA) by supporting a greater range of automatic, quantitative studies. For CIA, they would enable comparisons between native and interlanguages not only on the lexical level but also on the syntactic level. For EA, parallel parse trees would give more fine-grained characterization of the syntactic environment in which learner errors occur.

This paper reports on the construction of an L1-L2 parallel treebank for Chinese and presents a preliminary analysis. After summarizing previous work (Section 2), we give details on the texts in the treebank (Section 3) and on the linguistic annotations (Section 4). We then discuss the most overused and underused syntactic structures in the learner texts, as well as the underlying errors (Section 5). Finally, we conclude in Section 6.

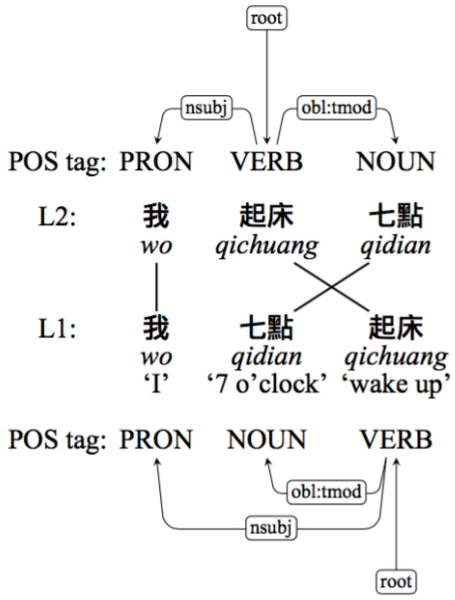


Figure 1: An example L1-L2 tree pair, including word alignments between the learner sentence ("L2") and its target hypothesis ("L1"), and the parse trees of the two sentences, annotated in the Universal Dependencies scheme for Chinese (Leung et al., 2016; Lee et al., 2017a).

2. Previous work

Most annotation efforts on learner treebanks have focused on English. Currently, the two major dependency treebanks for learner language are the Treebank of Learner English (TLE) (Berzak et al., 2016) and the project on Syntactically Annotating Learner Language of English (SALLE) (Ragheb and Dickinson, 2014). They both contain English texts written by non-native speakers. TLE annotates a subset of sentences from the Cambridge FCE corpus (Yannakoudakis et al., 2011), while SALLE has been applied on essays written by university students. A phrase-structure treebank for learner English (Nagata and Sakaguchi, 2016) has also been constructed for the texts in the Konan-JIEM Learner Corpus (Nagata et al., 2011). None of these treebanks, however, are L1-L2 parallel treebanks: they either do not provide explicit target

hypotheses (Ragheb and Dickinson, 2014; Nagata and Sakaguchi, 2016), or have not yet provided parse trees for the target hypotheses (Berzak et al., 2016).

As interest grows in learning Chinese as a foreign language (CFL), a number of large CFL corpora have been compiled and annotated (e.g., Zhang, 2009; Wang et al., 2015; Lee et al., 2016). Lee et al. (2017a) reported the first attempts to perform dependency annotation on CFL texts.

Lee et al. (2017b) described a case study on a small-scale L1-L2 parallel treebank, to gauge the potential of using search queries on the treebank as dynamically defined error categories. Errors are typically marked in a learner corpus with error tags, each of labels a problematic text span with an error category, and sometimes provides a corrected version of the text span (Yannakoudakis et al., 2011; Dahlmeier et al., 2013). From an L1-L2 parallel treebank, a researcher can retrieve sentences that exhibit a particular kind of error with a search query that consists of the desired tree patterns with word or node alignments. The case study used such queries to identify various kinds of word-order errors in CFL texts, and argued that they can supplement error tagsets by giving researchers more flexibility and precision in error definition (Lee et al., 2017b).

3. Textual material

The treebank contains sentences written by students of Chinese as a foreign language (CFL), as well as the target hypotheses of these sentences.

3.1 Learner sentences

The texts in our corpus have been collected from CFL learners of Chinese at Xi'an Normal University, China. There are a total of 22 learners, at both intermediate and advanced levels, with 5 different native language backgrounds: Korean, Russian, Turkish, Arabic and English. To protect their privacy, we anonymized their data but retained information on their gender, number of years of CFL studies, and native language.

The students received Chinese-language classes two to three times per month at the university. These classes offered detailed instructions on writing skills, e.g., thematic exercises and trainings in aspects of diction, punctuation, sentence, together with understandings of different logic connections under various native language backgrounds. After class, students were required to submit narratives, with diverse topics such as "The most memorable trip", "Experience of cultural differences", "My family members", etc. Based on scanned copies of their essays, we manually transcribed them in digital format. So far, we have collected a total of 27 essays, which consist of 600 sentences.

3.2 Target hypotheses

It is well known that there can be multiple target hypotheses for each learner sentence (Reznicek et al., 2013). One can focus on "minimal edits", which aim to make the sentence grammatically correct with the shortest edit distance; one can also perform "fluency edits", as advocated by Sakaguchi et al. (2016), which aim to make the sentence not only grammatical correct but also native-like, often involve rewriting the sentence as a whole. Even when performing the same kind of edits, individual annotators may come up with multiple valid target hypotheses.

For our treebank, a native speaker of Mandarin Chinese performed corrections on the 600 L2 sentences to produce a "minimal edit" target hypothesis for each sentence. An L2 sentence may be split up, or several L2 sentences may be combined. There are a total of 697 L1 sentences (target hypotheses) in the treebank. In future work, we plan to include "minimal edit" target hypotheses from other annotators, as well as "fluency edit" hypotheses, and study how multiple hypotheses affects interlanguage analysis.

4. Dependency annotation

Our learner corpus is annotated in the form of an L1-L2 parallel dependency treebank. The treebank consists of sentences written by CFL learners ("L2 sentences"); their target hypothesis ("L1 sentences"); the parse trees of the L1 and L2 sentences; and word alignment between the L1 sentences and L2 sentences. Figure 1 shows an example of a tree pair in our treebank.

We performed manual word segmentation, POS tagging, and dependency annotations on all sentences. Both the L1 and L2 sentences were annotated in the Universal Dependencies (UD) framework (Nivre et al., 2016). We have chosen to adopt the UD framework because of the large variety of languages for which UD treebanks exist, which can potentially facilitate contrastive analysis. Consider an investigation on the transfer hypothesis in texts written by CFL learners whose native language is X. One can examine differences in the aligned L1-L2 sentences within the portion of the treebank produced by native speakers of X. One can then further evaluate the extent to which UD treebanks of language X and of Chinese, ideally with comparable text types and topics, exhibit similar differences in linguistic properties.

For the annotation of L1 sentences, we followed the UD guidelines for Chinese (Leung et al., 2016). For the annotation of L2 sentences, we adapted the UD guidelines to take interlanguage characteristics into account (Lee et al., 2017a). Similar to current treebanks for learner English (Section 2), our guidelines adhere to the principle of "literal annotation", which asks annotators to perform syntactic analysis "as if the sentence were as syntactically well-formed as it can be, possibly ignoring meaning" (Ragheb and Dickinson, 2014). Dependency analysis on learner sentences can be challenging. According to a study on inter-annotator agreement (Lee et al., 2017a), the overall agreement can reach 94.0% for POS tags and 82.8% for labelled attachment. The agreement rate was lower within text spans that contain learner errors, dropping to 91.0% for POS tags and 75.1% for labelled attachment.

5. Overuse and underuse analysis

We would like to identify the grammatical structures with which the CFL learners in our corpus experienced the most difficulty. We applied the log-likelihood statistic (Rayson,

2008) to find the dependency relations that are most overused or underused among the L2 sentences (learner sentences), with respect to the L1 sentences (target hypotheses).

Table 1 shows the four syntactic relations, in the same Universal Dependencies (UD) scheme for Chinese (Leung et al., 2016; Lee et al., 2017a), that exhibit the most significant deviations in the learner sentences. The most underused dependency relation is `mark:adv`, which corresponds to the adverbializer. The most overused ones ones are `parataxis`, `discourse:sp`, and `compound:dir`, which correspond to parataxis, sentence-final particles and directional verb compounds. We now discuss the learner errors underlying these overuse and underuse phenomena.

Linguistic Structure	**UD Relation**	**LL Score**	**L2 Freq**	**L1 Freq**
Adverbalizer	`mark:adv`	4.38	31	50
Parataxis	`parataxis`	2.76	259	224
Sentence-final particles	`discourse:sp`	1.7	151	130
Directional verb compound	`compound:dir`	1.02	37	29

Table 1: Linguistic structures that are most overused and underused by learners, shown with their corresponding relations in Universal Dependencies (UD) and log-likelihood (LL) scores

5.1 Parataxis

Many learner sentences are written in a colloquial style, with multiple clauses placed side-by-side without any linking words between them. Although each clause is syntactically well-formed, the overall result is a run-on sentence. Consider the sentence in Table 2. Its first two clauses, headed by *kan* 'see' and *chuxian* 'appear', are linked with the relation `parataxis`(*kan*, *chuxian*).

A target hypothesis for a run-on sentence may sometimes be constructed by inserting appropriate conjunctions, but when the clauses are related only in a discursive way, the sentence may need to be split into several smaller, independent sentences. As shown in Table 2, the `parataxis` relations are thus replaced with `root`. Due to the abundant number of run-on sentences, the `parataxis` relation turned out to be the most overused one in our treebank. This phenomenon is also reflected in the higher number of L1 sentences (697 sentences) compared to L2 sentences (600 sentences).

5.2 Sentence-final particles

In Chinese, sentence-final particles, such as *le* 了, *de* 的, and *ba* 吧, can be placed at the end of clauses or sentences. They have a wide range of functions, such as modifying the modality, and expressing discourse and pragmatic information. In our UD scheme, it is annotated as a child (modifier) of the main predicate in a `discourse:sp` relation. As shown in Table 1, `discourse:sp` is among the most overused relations in the treebank, which suggests that errors related to the use of sentence-final particles is a frequent error type in learner Chinese text.

Among the various sentence-final particles, *le* is most overused. Table 3 shows an example sentence with an unnecessary *le*, and hence a superfluous relation `discourse:sp`(*gandong* 感动 'moved', *le* 了). In general, *le* should not be used in a simple assertion of a past event that did not involve a change of state (Li and Thompson, 1989).

The overuse of sentence-final particles appears to correlate with the native language. Native speakers of Korean, for example, overuse *le* more often than native speakers of English. This may be explained by the fact that Korean, similar to Chinese, uses sentence-final particles to mark clause types (Pak, 2006), while English does not. This finding also corresponds to a study of two CFL learners' acquisition of *le* (Sun, 1993).

Overuse of the `discourse:sp` relation is also in part due to the confusion between *le* as a sentence-final particle and *le* as an aspect marker. Though they share the same form, they function differently and appear at different positions. A verb typically occurs with the aspect marker *le* if the direct object is definite (Li and Thompson, 1989). Learners sometimes place it by mistake at the end of the sentence, such as in the sentence **yisheng gei wo kai yiping shenjinganyao le* 医生给我开一瓶神经安药了 'The doctor gave me some drugs for the nervous system'. It is annotated as a sentence-final particle, producing a superfluous `discourse:sp` relation.

Despite its overall trend of overuse, learner usage of *le* exhibits much variation. There are also many cases of omission, for example when the predicate is an Accomplishment or an Achievement verb. In such cases, *le* is required to present the resultative state after the attainment of the goal, such as in the sentence *women likai de shijian dao le* (我们离开的时间到了) 'Our time to leave has come'. Failure to use *le* in this context is also a rather common error in our data.

5.3 Directional verb compound

A directional verb compound is frequently used to express motion events, and is a typological feature of Chinese as a serial-verb language (Peyraube, 2006). The compound consists of at least two verbs, where the second verb is a the directional or deictic motion verb. Consider a sentence such as *ta pa shang le dingfeng* 他爬上了顶峰 'he climbed up to the peak'. In the two-verb series *pa* 'climb' and *shang* 'up', the second verb *shang* serves as a deictic motion verb. The UD scheme marks this type of compound with the relation `compound:dir`(*pa*, *shang*). As shown in Table 1, `compound:dir` is one of the more overused relations, indicating that unnecessary use of directional verbs is a significant learner error.

L2: … *偶然看到微信有一个功能是附近的人，我按了以后附近的人中出现了他的微信号，我申请加了他 …   … **ouran kan dao weixin you yi ge gongneng shi fujin de ren, wo an le yihou fujin de ren zhong chuxian le ta de weixin hao, wo shenqing jia le ta, …*   'By chance I saw a feature in WeChat called "People Nearby", after I clicked the button his number appeared among the people nearby, I sent him a friend invite, …'   `parataxis`(*kan* 看 'see', *chuxian* 出现 'appear')
L1: … 偶然看到微信有一个功能是附近的人。我按了以后，附近的人中出现了他的微信号。我申请加了他 …   *… ouran kan dao weixin you yi ge gongneng shi fujin de ren. wo an le yihou fujin de ren zhong chuxian le ta de weixin hao. wo shenqing jia le ta, …*   'By chance I saw a feature in WeChat called "People Nearby". After I clicked the button, his number appeared among the people nearby. I sent him a friend invite, …'   `root`(*kan* 看 'see')   `root`(*chuxian* 出现 'appear')

Table 2. Example of L1 and L2 sentences illustrating overuse of the `parataxis` relation (see Section 5.1).

L2: *但我去"麦加"朝觐时我很感动了   **dan wo qu "maijia" chaojin shi wo hen gandong le*.   'But when I went on pilgrimage to Mecca, I was very moved.'   `discourse:sp`(*gandong* 感动 'moved', *le* 了)
L1: 但我去"麦加"朝觐时我很感动   'But when I went on pilgrimage to Mecca, I was very moved.'

Table 3. Example of L1 and L2 sentences illustrating overuse of the `discourse:sp` relation (see Section 5.2).

A qualitative analysis suggested that many of these errors involve Chinese directional verbs that can play multiple roles: used independently (e.g., *xia* 下 'descend', *jin* 进 'enter') or modified by a directional complement (e.g., *xialai* 下来 'come down', *jinlai* 进来 'come in'); or serving as a directional complement itself (*fangxia* 放下 'put down', *paojin* 跑进 'run into'). Table 4 shows an example where *jin* 'enter' unnecessarily takes the directional complement *lai* 'come', yielding the superfluous relation `compound:dir`(*jin*, *lai*).

L2: *她进来了教室，坐在我左边。   **ta jin lai le jiaoshi, zuo zai wo zuobian.*   'She came into the classroom and sat on my left.'   `compound:dir`(*jin* 进 'came', *lai* 来 'into')
L1:她进了教室，坐在我左边。   *ta jin le jiaoshi, zuo zai wo zuobian.*   'She entered the classroom and sat on my left.'

Table 4. Example of L1 and L2 sentences illustrating overuse of the `compound:dir` relation (see Section 5.3).

5.4 Adverbializer

As shown in Table 1, the most underused structure in the learner texts is the adverbializer. In Chinese, the manner adverbializer *de* 地 turns an adjective that follows it into an adverb. Consider the sentence *nabian de ren qiguai de kan zhe wo* 那边的人奇怪地看着我 'The person over there looked at me strangely'. In this sentence, the adverbializer *de* turns the preceding adjective *qiguai* 'strange' into the adverb 'strangely'. In our UD scheme, this grammatical function is annotated as the relation `mark:adv`(*qiguai*, *de*).

The adverbializer *de* is easily confusable with a homonym, the particle *de* 的, which may follow an adjective that modifies a noun. For example, in the noun phrase *qiguai de wenti* 奇怪的问题 'strange problem', the particle *de* is inserted between the adjective *qiguai* 'strange' and the noun *wenti* 'problem'. The UD scheme uses the relation `mark:rel`(*qiguai*, *de*) to mark this structure.

The adverbializer *de* and the particle *de* share a similar linguistic environment in that they both modify adjectives. In our data, learners tend to overuse the particle and underuse the adverbializer. As shown in the example in Table 5, this phenomenon results in the relation `mark:rel`(*tebie* 特别 'special', *de* 的) in place of the expected relation `mark:adv`(*tebie* 特别 'special', *de* 地). This kind of error was a major contributor to the underuse of `mark:adv`.

6. Conclusions

We have described an on-going effort to build a large-scale L1-L2 parallel dependency treebank — i.e., parse trees of non-native sentences ("L2 sentences"), aligned to the parse trees of their target hypotheses ("L1 sentence") — for Chinese. The treebank is annotated in the Universal Dependencies (UD) framework.

We presented a preliminary analysis on the treebank, identifying the most overused and underused syntactic

relations in the learner text, with respect to the log-likelihood score. The adverbializer is the most underused, while parataxis, sentence-final particles, directional verb compounds are the most underused.

We are currently expanding the size of the treebank, including the number and kinds of target hypotheses. In future work, we plan to perform a most exhaustive analysis of overused and underused grammatical structures, and apply the treebank data to evaluate the transfer hypothesis in conjunction with other UD treebanks.

L2: *当时我特别的同感他们 **dangshi wo tebie de tonggan tamen* 'At that time, I felt special sympathetic to them.' `mark:rel`(*tebie* 特别 'special', *de* 的)
L1: 当时我特别地同感他们 *dangshi wo tebie de tonggan tamen* 'At that time, I felt especially sympathetic to them.' `mark:adv`(*tebie* 特别 'special', *de* 地)

Table 5. Example of L1 and L2 sentences illustrating underuse of the `mark:adv` relation (see Section 5.4).

7. Acknowledgements

The work reported in this paper was partially supported by a Strategic Research Grant (Project no. 7004494) from City University of Hong Kong.

8. Bibliographical References

Berzak, Y., Kenney, J., Spadine, C., Wang, J. X., Lam, L., Mori, K. S., Garza, S., and Katz, B. (2016). Universal Dependencies for Learner English. In *Proc. ACL*.

Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). Building a Large Annotated Corpus of Learner English: The NUS Corpus of Learner English. In *Proc. 8th Workshop on Innovative Use of NLP for Building Educational Applications*.

Díaz-Negrillo, A., Meurers, D., Valera, S., and Wunsch, H. (2010). Towards Interlanguage POS Annotation for Effective Learner Corpora in SLA and FLT. *Language Forum*, 36(1-2):139–154.

Granger, S. (2015). Contrastive Interlanguage Analysis: A Reappraisal. *International Journal of Learner Corpus Research* 1(1):7–24.

Lee, L. H., Gaoqi, R. A. O., Yu, L. C., Endong, X. U. N., Zhang, B., & Chang, L. P. (2016). Overview of NLP-TEA 2016 Shared Task for Chinese Grammatical Error Diagnosis. In *Proc. 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA2016)*.

Lee, J., Leung, H., & Li, K. (2017a). Towards Universal Dependencies for Learner Chinese. In *Proc. NoDaLiDa Workshop on Universal Dependencies*.

Lee, J., Li, K., & Leung, H. (2017b). L1-L2 Parallel Dependency Treebank as Learner Corpus. In *Proc. 15th International Conference on Parsing Technologies*.

Leung, H., Poiret, R., Wong, T. S., Chen, X., Gerdes, K. and Lee, J. (2016). Developing Universal Dependencies for Mandarin Chinese. In *Proc. 12th Workshop on Asian Language Resources*.

Li, C. N. and Thompson, S. A. (1989). *Mandarin Chinese: A Functional Reference Grammar*. University of California Press.

Nagata, R., Whittaker, E., and Sheinman, V. (2011). Creating a Manually Error-tagged and Shallow-parsed Learner Corpus. In *Proc ACL*.

Nagata, R. and Sakaguchi, K. (2016). Phrase Structure Annotation and Parsing for Learner English. In *Proc. ACL*.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proc. LREC*.

Pak, M. (2006). Jussive clauses and agreement of sentence final particles in Korean. *Japanese/Korean Linguistics*, *14*, 295-306.

Peyraube, A. (2006). Motion events in Chinese. *Space in languages: Linguistic systems and cognitive categories*, 66, 121-135.

Ragheb, M. and Dickinson, M. (2014). Developing a Corpus of Syntactically-Annotated Learner Language for English. In *Proc. 13th International Workshop on Treebanks and Linguistic Theories (TLT)*.

Rayson, P. (2008). From Key Words to Key Semantic Domains. *International Journal of Corpus Linguistics*, *13*(4), 519-549.

Reznicek, M., Lüdeling, A., and Hirschmann, H. (2013). Competing Target Hypotheses in the Falko Corpus: A Flexible Multi-Layer Corpus Architecture. In Ana Díaz-Negrillo, editor, *Automatic Treatment and Analysis of Learner Corpus Data*, pages 101–123, Amsterdam. John Benjamins.

Sakaguchi, K., Napoles, C., Post, M., & Tetreault, J. (2016). Reassessing the goals of grammatical error correction: Fluency instead of grammaticality. *Transactions of the Association for Computational Linguistics*, *4*, 169-182.

Sun, D. (1993). Preliminary analysis of the acquisition of le 了 by foreign learners. *Language Teaching and Linguistic Studies*, 2, pages 65-75.

Wang, M., Malmasi, S., and Huang, M. (2015). The Jinan Chinese Learner Corpus. In *Proc. 10th Workshop on Innovative Use of NLP for Building Educational Applications*.

Yannakoudakis, H., Briscoe, T., and Medlock, B. (2011). A New Dataset and Method for Automatically Grading ESOL Texts. In *Proc. ACL*.

Zhang, B. (2009). The Characteristics and Functions of the HSK Dynamic Composition Corpus. *International Chinese Language Education*, 4(11).

The Use of Text Alignment in Semi-Automatic Error Analysis: Use Case in the Development of the Corpus of the Latvian Language Learners

Roberts Darģis[1], Ilze Auziņa[2], Kristīne Levāne-Petrova[3]
Faculty of Computing, University of Latvia[1], Institute of Mathematics and Computer Science, University of Latvia[2,3]
Raina bulvaris 19, Riga, LV-1459, Latvia[1], Raina bulvaris 29, Riga, LV-1459, Latvia[2,3]
{ roberts.dargis, ilze.auzina, kristine.levane-petrova }@lumii.lv

Abstract

This article presents a different method for creation of error annotated corpora. The approach suggested in this paper consists of multiple parts - text correction, automated morphological analysis, automated text alignment and error annotation. Error annotation can easily be semi-automated with a rule-based system, similar to the one used in this paper. The text correction can also be semi-automated using a rule-based system or even machine learning. The use of the text correction, word, and letter alignment enables more in-depth analysis of errors types, providing opportunities for quantitative research. The proposed method has been approbated in the development of the corpus of the Latvian language learners. Spelling, punctuation, grammatical, syntactic and lexical errors are annotated in the corpus. Text that is not understandable is marked as unclear for additional analysis. The method can easily be adapted for the development of error corpora in any other languages with relatively free word order. The highest gain from this method will be for highly inflected languages with rich morphology.

Keywords: learner corpus, error annotation, word alignment

1. Introduction

The purpose of this article is to describe the error-annotating methodology and the tool that is used to annotate *The Corpus of the Latvian Language Learners* (Latvian as L2 and foreign). As the Sylviane Granger admits, the learner corpora constitute a new resource for second language acquisition and foreign language teaching specialists, especially if they are error-tagged. (Granger, 2003).

Appropriately designed learner corpus and consistently annotated errors can provide answers to global questions such as: what is the most frequent type of error, how the native language influence the error type. As the developed corpus includes the texts of different levels of language acquisition the corpus can provide an answer to very specific questions, for example, are mistakes related to noun endings more frequent for B2 or C1 level? Based on the data from the corpus, also different workbooks might be developed for people learning a second language.

Latvian is a language with rich morphology and a relatively free word order. Latvian in general can be considered a phonetic language – a language with a relatively simple relationship between orthography and phonology. From the language acquisition perspective, Latvian has several specific properties: short and long vowels and diphthongs, highly inflected language, rather free word order. These properties have to be taken into account in error-annotation.

There are many learner corpora for English and last decades learner corpus have been created for other languages as well, for example, French, Swedish, Norwegian, Dutch, Spanish and German (Granger, 2008), and their range is expanding.

Currently, *The Corpus of the Latvian Language Learners* is being created. The corpus contains the successfully passed tests of the State Language Proficiency Testing (Certification) that is used to evaluate a person's (henceforth – Applicant's) state language proficiency level. For every language proficiency level (A1, A2, B1, B2, C1, C2) 150 tests have been used that makes in total 900 tests. If the State Language Proficiency Examination is passed successfully, the Applicant receives the state language proficiency certificate, that is required for employment requirements and acquisition of a permanent residence permit. The methodology and tool described in this paper are used to create this corpus.

At this moment, there is no other Latvian learner corpus. One more learner corpus of Latvian is being developed (www.esamkorpuss.lv) by PhD student Inga Znotiņa. The corpus "Esam" is a learner corpus that consists of the texts that have been written by university students, learners of the second Baltic language; namely, Latvian for students of Lithuanian background, and Lithuanian for students of Latvian background. (Znotiņa, 2015; Znotiņa, 2017).

The paper is further structured as follows: section 2 describes the creation stages of the corpus, section 3 gives an introduction to the error annotation guidelines, section 4 describes the automated processing of the data, section 5 explains the computing of the statistics of the annotated errors. The paper is concluded in section 6.

2. The Creation Stages of the Corpus

There are several stages of creating the corpus:

1. Data digitalization;
2. Text correction;
3. Automated morphological annotation, including tokenization, part-of-speech tagging, lemmatization;
4. Original and corrected text alignment;
5. Automated error annotation and manual revision.

First, texts are digitalized by manually transcribing hand-written test. The transcriptions fully correspond to the original document (test). Sometimes handwriting deciphering causes difficulties.

After data digitization, the texts are manually corrected. The texts are overwritten according to the norms of the Latvian language. All spelling, grammatical, lexical and punctuation errors are corrected. If there is a redundant word in the sentence, it is deleted, while the released word is written in the sentence (syntactical error). To be able to align words, inadequate word order is not changed, but it will be annotated. If some portion of the text is unclear, it is left unchanged, and it will be annotated.

Further, the data is automatically processed. Original and corrected text is tokenized, morphologically annotated and aligned. From the alignments, initial error annotations are generated and prepared for manual revision.

3. Criteria of Error Annotation

Learner corpora are usually error annotated, that is, spelling (orthographic), lexical, and grammatical errors in the corpus have been annotated with the help of a standardized system of error tags (Granger, 2003).

The texts are error annotated using an error taxonomy created for the Latvian language (Table 1). Similar error taxonomy is used in the learner corpus of the second Baltic language "Esam" (Znotiņa, 2015). This error taxonomy can be used for other inflected languages with free word order.

Type	Subtype
Spelling errors	Upper / lower case letter
	Diacritics
	Separately / together spelled words
	Missing letters
	Redundant letters
	Other spelling errors
Punctuation errors	Missing punctuation
	Redundant punctuation
	Incorrect punctuation
Grammatical errors	Incorrect word form (such as inflection, gender, number, definite/indefinite ending, tense, person)
	Derivation
	Morphophonemic consonant alternation
Syntactic errors	Word order
	Redundant word
	Missing word
Lexical errors	Meaning
	Compliance
	Readability
	Collocation
Unclear text	

Table 1: Error types

Most of the spelling and grammatical errors are tied to a single token, but there are some constructions, that consists of multiple words, for example, analytical forms. In these cases, it is necessary to be able to annotate multiword expression as a single token.

If in some segment the word order is incorrect, it is not changed, because it will make automatic alignment a lot more difficult and sometimes even impossible. Other errors are still annotated in these segments, and the text segment is marked as one with wrong word order.

On the contrary to the English language, in Latvian, punctuation is very important. The punctuation is based on the grammatical principles, and the different use of punctuation marks often completely change the meaning of the sentence.

Occasionally the spelling errors may overlap with grammatical errors. Error annotation system, therefore, should allow annotating several types of errors (usually grammatical and spelling errors) for one wordform simultaneously.

There are ambiguous errors, for example, one missing diacritic can change the grammatical meaning, but the misuse of diacritics is a common error in Latvian learners' texts as well. In these cases, both error types (grammatical and spelling) are annotated.

4. The Automated Processing of the Data

The automated processing consists of three steps:

1. Tokenization and morphological analysis;
2. Text alignment (including token and letter level analysis);
3. Automatic error annotation.

Each of this step is described in more details in the following subsections.

4.1 Morphological Analysis

First, the original text and corrected text are tokenized and automated morphological annotations are generated. Morphological annotation consists of a morphological tag (including part of speech), lemma and stem. In most cases, only the morphological information from the corrected text is used. Although the morphological annotation is done for the original text as well, this information is often inaccurate because of the many grammatical errors. Morphological information from the original text is used only when there is no corresponding word in the corrected text, i.e., the word was redundant in the original text, and it was deleted in the corrected text.

For Latvian the morphological annotator developed by Paikens was used (Paikens, 2013).

4.2 Text Alignment

The tokens are aligned, using word level alignment into one-to-one relationships, where each token in the corrected text has one or none aligned tokens in the original text and vice versa. The alignment is found by using a similar approach to the one used in word error rate calculations in speech recognition. The token relationships are found by computing the alignment with the lowest edit distance. The edit distance is calculated as follows:

- The cost of deleting a token is 1.
- The cost of inserting a token is 1.
- The cost of substituting a token is the relative edit distance between tokens.

The relative edit distance is obtained by computing the edit distance between tokens and dividing it by the length of the longest token, so the value is between zero and one. If the cost of the substitution were 1, the same as in speech recognition tasks, in segments with insertions/deletions and many spelling errors, there would be multiple alignments for the same cost, because there would be no way how to tell which token is the inserted/deleted one.

After token level alignment, the next step is letter level alignment for the substituted tokens. The letter level alignments are used to generate automatical error annotations and to improve user experience in manual error labeling by emphasizing the differences in two tokens. A significant portion of spelling errors is an incorrect use of diacritical marks or letter case, ignoring them when computing letter alignment helps to get the correct alignment especially when if there are some missing or redundant letters.

4.3 Automatic Error Annotation

Automatic error annotations, which later will be manually edited using annotation revision interface (Figure 1), are generated by a rule-based system from the alignments and morphological annotations.

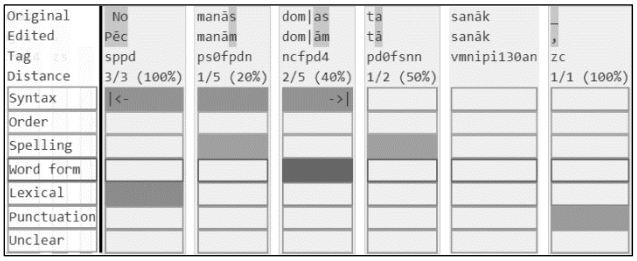


Figure 1: Error annotation revision interface

The order of the rules is important because after the first applicable rule is found, the evaluation of the rules is stopped.

The rules go as follows:

- If both tokens (the original and the corrected one) matches, there is no error.
- If the token consists only of punctuation marks, it is punctuation error.
- If one of the tokens is missing (it was a redundant or missing word), it is a syntax error.
- If the relative edit distance between tokens is greater than 0.8, it is considered that the word is most likely replaced with a different word and it is a lexical error.
- If none of the rules above applied, it can be one or both of two error types – spelling or grammatical error.

Letter level alignments and morphological information are used to determine if it is spelling or grammatical error. It is annotated as a grammatical error if the differences between two tokens are at the ending of a word. Otherwise, it is a grammatical error. The token contains grammatical and spelling errors if the differences are at the beginning of the word and the ending of the word. For the words in the corrected text, the boundary between the beginning and end of the word is obtained from automatical morphological annotations. For the words in the original text, the boundary is projected from corrected text using letter level alignments.

5. The Analysis of Annotated Errors

The analysis of any data could be divided into two types – quantitative and qualitative analysis. In quantitative analysis, the data is grouped by some feature, for example, by the misspelled letter. For this analysis, it is necessary to know what kind of feature meaningful statistics could be obtained and how to get this feature from the data automatically. If meaningful features are not known, or it is not possible to extract them automatically, qualitative analysis is an option where one tries to identify the features or extract them manually.

The categories used in error annotation tool include all of the error types. The possibilities for automatic error subtype determination and other meaningful feature extraction differs for each error type. The available options for quantitative analysis of the error corpus from the annotations suggested in this paper will be discussed in this section.

5.1 The Analysis of Spelling Errors

For spelling errors, it is possible to do a quantitative analysis of subtypes from words with spelling errors using letter level alignments. The subtype analysis is done only for words that contain only spelling errors because if the words contain grammatical errors as well, it is hard to automatically differentiate which inconsistencies in letter level alignment are due to grammatical errors and which due to spelling errors. In many cases, it is also hard to manually differentiate between grammatical and spelling errors.

5.2 The Analysis of Punctuation Errors

Punctuation errors are the simplest error type. With quantitative analysis, it would be possible to show which punctuation errors are the most frequent. More complex quantitative error analysis could be added as well, for example, investigating in what context commas are missing or redundant most frequently. Commas are important in languages with free word order.

5.3 The Analysis of Grammatical Errors

The simplest quantitative analysis of grammatical errors could be done from the morphological annotations of corrected text to determine in which part of speech (such as nouns, pronouns, verbs, adjectives), inflection, tense, person, etc., learners make most mistakes.

5.4 The Analysis of Syntactic Errors

For syntactic errors, the relative percentage of text segments with syntactic errors in different language levels (A1 to C2) can be quantitatively analyzed.

In the error annotations, syntactic errors are annotated in two subtypes – word order errors and other syntactic errors. Word order errors are annotated separately because these errors are not corrected. The main reason is that the alignment approach used currently assumes the order in both texts are the same.

For word order errors, no other quantitative analysis is possible because the corrected text is not available. In other syntactic errors, the correct text is available, so more detailed quantitative analysis is possible for this subtype.

5.5 The Analysis of Lexical Errors

For lexical errors, a meaningful quantitative analysis is not straightforward. Because of the spelling and grammatical errors, the original words cannot be grouped directly. To work around this problem original words could be grouped based on similarity. If the differences between some group of words look like spelling errors (for example, different use of diacritics), these words could be considered to be the same and grouping them would provide more meaningful quantitative analysis. Further research is required to make better conclusions about the best approach for the analysis of this error type.

5.6 The Analysis of Unclear Text

Segments that can't be understood are annotated as unclear text. Similar to the word order errors, the relative percentage of text segments with unclear text in different language levels (A1 to C2) can be quantitatively analyzed.

6. Inter-Annotator Agreement

To evaluate inter-annotator agreement 20 documents containing 1942 tokens were annotated by two users. The annotation was done in two steps. First, the text was rewritten by each user individually. Then, each user annotates errors on their rewritten version of text.

Comparing the texts rewritten by each user, 92.7% of tokens matched (1800 out of 1942 tokens). Error level inter-annotator agreement was calculated only on matched tokens. The number of tokens annotated with different error classes only by User A, User B or equally by both of the users are shown in Table 2. The inter-annotator agreement was measured with Cohen's kappa coefficient (κ) (Cohen, 1960). The value is within the interval [−1, 1], where κ = 1 means perfect agreement, κ = 0 agreement equal to chance, and κ = −1 "perfect" disagreement.

Error Type	User A	User B	Both	κ
Spelling	9	23	219	0.85
Grammatical	13	9	70	0.83
Lexical	19	3	8	0.40
Punctuation	1	1	74	0.98
Unclear text	17	23	2	0.04
Word order	13	0	0	0.00
Syntactical	0	6	3	0.49

Table 2: Inter-annotator agreement

7. Corpus Statistics

The corpus contains 142684 tokens from 1496 documents. On average 22.2% tokens contained errors. The distribution of different error types in the corpus is given in the Table 3. Percentages relative to tokes with errors sums up to more than 100% because one token can contain multiple errors.

Error Type	Count	Percentage from tokens with errors	Percentage from total tokens
Spelling	14956	47.13%	10.48%
Grammatical	8075	25.45%	5.66%
Punctuation	5857	18.46%	4.10%
Lexical	1756	5.53%	1.23%
Word order	1703	5.37%	1.19%
Unclear text	1546	4.87%	1.08%
Syntactical	1321	4.16%	0.93%

Table 3: The distribution of different error types

To evaluate how good the naive error prediction system works, the number of tokens marked with different error types only by user, only by system or both was calculated (Table 4). Correctness was chosen as a measurement of the system's performance. Correctness is the percentage of the unchanged tags from the total number of tokens that contained any type of error. The error prediction system was developed to speed up the annotation process, it wasn't meant to be 100% correct. Examining the statistics it can be can concluded that error prediction system predicts a spelling error when it is actually a grammatical error. This is something that could be improved. The system's current version will never predict a lexical error. The time spent on building a system that predicts lexical errors might not be worth it because It is hard to predict this kind of errors and the inter-annotator agreement for lexical errors were also significantly lower than for other error types. Lexical errors are also less common than Spelling, grammatical or punctuation errors.

Error Type	User	System	Both	Correctness
Punctuation	161	22	5696	99.42%
Lexical	1756	0	0	94.47%
Grammatical	1075	745	7000	94.26%
Spelling	299	2090	14657	92.47%

Table 4: The correctness of error prediction system

8. Conclusion and Further Work

The error annotation method suggested in this article proved to be easily understandable and usable for the annotators. The time the annotation process took was similar to the time necessary for classical annotation process. The use of text correction and alignments enables opportunities for a lot more detailed quantitative statistical analysis.

As mentioned earlier, the biggest drawback of this error annotation approach is limitation on word order errors, but there are many flective languages (for example, most of the Slavic languages) for which the word order is not grammatically significant. In the Latvian learners' corpora inter-annotator about word order error was 0 (close to chance).

The development of automated text correction process would give the highest impact to annotator's experience and would reduce the time necessary for the development of the corpus.

Revision of the current automatic error annotation rules and refinement from the lessons learned during the development of the corpus could improve user experience.

9. Acknowledgements

This work has received financial support from the Faculty of Computing, University of Latvia.

The results reported in this paper are part of the Latvian Language agency's research project "Quality of the Latvian language: results of the state language proficiency test".

The tools developed and used in this project has received financial support from the European Regional Development Fund under the grant agreement No. 1.1.1.1/16/A/219.

10. Bibliographical References

Cohen J (1960) A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20(1):37–46

Granger, S. (2008) Learner corpora. Corpus Linguistics : An International Handbook. Anke Lüdeling, Merja Kytö (eds.). Berlin, New York : Walter de Gruyter, 2008, pp. 259–275.

Granger, S. (2003) Error-tagged learner corpora and CALL: A promising synergy. CALICO Journal, Vol. 20, No. 3, 2003, pp. 465–480.

Paikens, P., Rituma, L., Pretkalniņa, L. (2013) Morphological analysis with limited resources: Latvian example, Proceedings of NODALIDA 2013, pp. 267–278.

Znotiņa, I. (2017) Computer-Aided Error Analysis for Researching Baltic Interlanguage. Rural Environment, Education, Personality. Proceedings of the 10th International Scientific Conference, 2017, pp. 238–244. http://llufb.llu.lv/conference/REEP/2017/Latvia-Univ-Agricult-REEP-2017_proceedings-238-244.pdf

Znotiņa, I. (2015) Learner corpus annotation in Latvia and Lithuania. Sustainable Multilingualism, No. 7. 2015, pp. 145–159.

Error annotation in a Learner Corpus of Portuguese

Iria del Río & Amália Mendes
University of Lisbon, Center of Linguistics - CLUL
igayo@letras.ulisboa.pt, amaliamendes@letras.ulisboa.pt

Abstract

We present the error tagging system of the COPLE2 corpus and the first results of its implementation.. The system takes advantage of the corpus architecture and the possibilities of the TEITOK environment to reduce manual effort and produce a final standoff, multi-level annotation with position-based tags that account for the main error types observed in the corpus. The first step of the tagging process involves the manual annotation of errors at the token level. We have already annotated 47% of the corpus using this approach. In a further step, the token-based annotations will be automatically transformed (fully or partially) in position-based error tags. COPLE2 is the first Portuguese learner corpus with error annotation. We expect that this work will support new research in different fields connected with Portuguese as second/foreign language, like Second Language Acquisition/Teaching or Computer Assisted Learning.

Keywords: learner corpus, error annotation, second language acquisition.

1. Introduction

Error tagging has been proved to be an important aspect in learner corpora research, since it helps to identify problematic areas in the learning process (Granger, 2004) and provides useful data for many areas of study (Díaz-Negrillo and Thompson, 2013). Nevertheless, error tagging is not always present in learner corpora. We can identify at least two important causes for this fact: error-tagging is a high time-consuming task that has to be performed manually; there are no standards, and taxonomies are a result of particular projects with specific interests (Díaz-Negrillo and Fernández-Domíguez, 2006). Error tagging techniques have evolved over the past few years from inline annotations with a unique interpretation, to standoff, multi-layer annotations with multiple error hypotheses. On the contrary, the conceptual design of taxonomies shows less development, with fewer changes in the categories and dimensions observed. Finally, the automatization of the process is still a challenge.

We present the error annotation system for the COPLE2 corpus and the first results of its implementation. We show that our system takes advantage of the COPLE2 architecture as well as the TEITOK platform possibilities to reduce manual effort and produce a final annotation that follows the actual trends for error tagging. Since COPLE2 is the first corpus with error annotation for Portuguese, we hope that our work will open new possibilities in the study of Portuguese as second/foreign language.

The paper is structured as follows: section 2 shows related work in error annotation; section 3 presents the COPLE2 corpus and its error annotation system; in section 4 we show our first annotation results; finally, section 5 presents the conclusions and future challenges.

2. Related work

The analysis of error tagging development leads to three relevant conclusions (among others). First, conceptual aspects related to the design of taxonomies show little variation through the years. Secondly, innovations have affected mainly the technical aspects of the annotation process. Finally, manual annotation is still the most common procedure and implies a high human effort.

Concerning the design of taxonomies, we can verify that most of them are: designed for written text, while schemes for oral data are scarce; grounded on three linguistic areas: spelling, grammar and lexis, leaving out others like phonetics or discourse; POS-centered, so certain linguistic units are undefined and certain levels of analysis are unexamined (Díaz-Negrillo and Fernández-Ramírez, 2006).

Moving to technical aspects, there has been an evolution from in-line and flat architectures to multi-layer standoff systems in all areas of corpus annotation. In first learner corpora with error annotation[1], like the Cambridge Learner Corpus (CLC) (Nicholls, 2003), or the International Corpus of Learning English (ICLE) at Louvain (Granger et al., 2009), the tags were inserted in the learner text and a unique interpretation was proposed. We can see below an example of this type of annotation from the Louvain corpus:

(1) […] barons that (GVT) lived $had lived$ in those (FS) castels $castles$. (ICLE-Louvain; Dagneaux et al. 1998: 16).

Lüdeling et al. (2005) points out two problems of this approach: (i) the number and category of annotation layers must be decided in the corpus design phase; (ii) it is difficult to annotate beyond the token level, that is, sequences of words. The first problem goes against one of the design principles for error annotation stated by Granger (2003), flexibility. The second problem can be solved if an XML format is used, as in FreeText (Granger 2003: 470) or CLC. However, as noted again by Lüdeling et al. (2005), 'it is not possible to annotate overlapping ranges on different annotation layers since these cannot be mapped on a single ordered tree'. We can add a third problem of this methodology: annotations are mixed with the original learner text, which makes it difficult to manage the different levels of information in the corpus.

The FALKO corpus (Lüdeling et al., 2005) introduced a paradigm shift in the area. This system proposed for the first time a multi-layer and standoff design for error (and other types of) annotation in learner corpora. This architecture solved the problems that we mentioned above. On the one hand, the multi-layer design allows for the annotation of different types of information at the same time. For error annotation this means that different hypothesis for a given error can be proposed, and that in

[1] For a detailed review see Díaz-Negrillo and Fernández-Domíguez (2006).

general each layer corresponds to one level of interpretation. Besides this, the multi-layer architecture makes possible to add/remove layers when needed, which makes the system more flexible. On the other hand, standoff annotations make possible to store the different annotations apart from the original text. Finally, they allow for the annotation of sequences of words and also for managing overlapping ranges of text. Most recent learner corpora with error annotation show this type of design. We can find it in FALKO, MERLIN (Boyd et al., 2014) (which uses the same target hypothesis than FALKO) or CzeSL (Rosen et al., 2013).

Finally, one of the main problems of error tagging is that annotation is performed manually, being automatization one of the pending tasks. Different strategies have been tested to solve this drawback. Kutuzov and Kuzmenko (2015) explore the option of pre-processing learner texts with a spell-checker to identify potential errors. Rosen et al. (2013) apply different tools designed for native language to the learner texts and compare their output with manual error annotation. They conclude that this strategy helps to identify potential errors and may even replace manual annotation in large-scale projects. Andersen (2011) explores the possibility of developing automatic rules for error detection and correction derived from manually error-annotated text.

3. Error annotation in the COPLE2 corpus

3.1 The COPLE2 corpus

COPLE2 (Mendes et al., 2016) is a learner corpus of Portuguese as a second/foreign language developed at the University of Lisbon. It contains written and oral productions of Portuguese learners with different L1s and proficiencies (15 languages, A1 to C1 levels), and provides rich TEI annotation through the TEITOK environment (Janssen, 2016). The corpus contains complete metadata related to the learner (age, native language/s, years studying Portuguese, etc.), the topic of the text or the circumstances where the text was produced. The original hand-written texts and oral productions (audios) are accessible in the platform. All the changes made by the students (additions, deletions, transpositions of segments, etc.) are annotated, as well as the corrections suggested by the Portuguese teachers. The texts are tokenized, lemmatized and POS tagged using the Neotag tagger (Janssen, 2012). All the information is stored together with the original texts in XML files that can be searched through the CQP query language (Christ et al., 1999).

3.2 Error annotation in the COPLE2 corpus

For error annotation in COPLE2 (del Río et al., 2016) we take advantage of the corpus architecture and the information already annotated, as well as of the TEITOK possibilities to build an annotation system that: (i) deals with the challenges of error annotation; (ii) follows the current trends in the field; (iii) reduces and simplifies the manual annotation as much as possible and tries to automatize it.

Error annotation in COPLE2 is performed through two complementary systems: a flat, token-based system with three error categories that is applied inside the XML files, and a fine-grained, standoff, multi-level system. The token-based system makes possible a quick and simple annotation, supports the visualization of the corrected text and complex queries using CQP. But, what is more important: it allows for the automatic generation of the fine-grained annotation system using all the information annotated in the corpus and the possibilities of the TEITOK platform. Next, we will describe both systems in detail and the relation between them.

In the token-based annotation, errors may be classified into three linguistic areas: orthographic, grammatical and lexical. Each area contains three fields of annotation: word form, lemma and POS. Depending on the problem/s affecting the original student form, the annotator has to select the affected linguistic area/s and introduce the required correct form/s (word form, POS, lemma). For example, given the input: *um cidade* ('a_{MASC} $city_{FEM}$') instead of *uma cidade* ('a_{FEM} $city_{FEM}$), where what we have is an agreement problem between a determiner and a noun, the annotator introduces the correct word form for the determiner (*uma* instead of *um*) and the correct POS, but the lemma remains the same (*um*). Multiple linguistic areas can be filled for a given token at the same time (for example, when a student form shows an orthographical problem, a grammatical problem and a lexical problem). All the error annotations are integrated in the XML files with the students' texts and the other annotations mentioned in section 3.1. For errors that go beyond the token and do not fit into this schema, the first token of the wrong sequence is annotated with a special code that stands for "multi-token". This way, we ensure that all the errors are identified and classified.

Because of its simplicity and its integration in the TEITOK architecture, this system shows several advantages. First, from the taxonomical point of view, it is simple and general. The annotator decides between a limited number of possibilities (three types of errors with three possible corrections: word form, POS and lemma). There are no fined-grained error types with linguistic details to judge. Moreover, it is intuitive because the annotator decides on the error type by recovering the expected form in that particular context, i.e., the corrected form determines the error type. Furthermore, it allows for three different target hypotheses for a given error. Besides this, the system is perfectly integrated in the TEITOK environment: it allows for complex queries at the token level using all the information stored in the corpus through CQP; it makes possible a visual representation of the learner text corrected at three different levels (orthographic, grammatical and lexical). However, taking into account what we discussed in section 1, it is clear that this system presents some problems for error annotation: it only works at the token level[2]; it offers a limited categorization and description of errors types; and it is limited to three linguistic areas, while some errors go beyond those areas.

Due to these limitations, the token-level annotation is complemented with a fine-grained, standoff, multi-level system that uses error tags plus corrected forms. The annotations are stored standoff in XML files, can be applied to sequences of words and to overlapping fragments of text. The tagset designed for this system is similar to the taxonomies described in Tono (2003),

[2]Although, as we will see in next section, we have found that only a small percentage of the total errors identified so far in COPLE2 did not fit at all in a token based interpretation.

Nicholls (2003) or Dagneaux et al. (2005). It contains 38 tags and it is structured in two levels of information: (i) general linguistic area affected; (ii) error category (and subcategories in some cases). Level 1 includes (for the moment) the same three linguistic areas that the token-based system: Orthographic (includes spelling and punctuation errors), Grammatical (includes agreement errors; errors affecting verb tense, mode, etc.) and Lexical (lexical choice errors). Level 2 accounts for common error categories like agreement or wrong POS. The tags are position-based: the first letter corresponds to level 1 and the subsequent letters to level 2. For example, for agreement errors affecting gender, the tag is "GAG" which stands for "Grammar + Agreement + Gender".

Most of the tags and their corresponding corrections can be automatically generated (at least partially) comparing the original form of the student with the corrections (plus lemma and POS) introduced at the token level. The first letter of the tag can be always generated just checking the linguistic level where the corrections were added. The subsequent letters of the tag can be inferred using the linguistic information annotated in the corpus. For example, we have an error tag for accentuation marks (SS). For this error type, we can compare the student form and the *orthographically corrected form* to check if the difference affects only accentuation marks and, in that case, assign the corresponding letters to the error tag (SS). With this strategy, we take advantage of the TEITOK and COPLE2 possibilities to automatically produce a detailed error annotation with low manual effort.

4. Results of error annotation at the token level

We have started the error annotation at the token level.[3] So far, we have annotated 442 texts (47% of the total files), corresponding to 72,858 tokens (42.5% of the total tokens in the corpus). We have added 14,984 annotations. Of these, 13,581 are token-based (91%) and 1,403 go beyond the token (9%). The token-based annotations have the following distribution: 6,432 orthographical errors; 5,881 grammatical errors; 1,268 lexical errors.

For the moment, our results indicate that the token-based representation may account for most of the errors found. However, these results may be biased by the fact that the annotator has tried to adjust the annotation to the token-based representation and we think that a deeper analysis is necessary to draw precise conclusions. For example: we have annotated predicative adjectives with disagreement problems at the token level, as in:

(1) *As praias são muito **lindos**,* […] > *lindas*[4] ('The beaches$_{FEM}$ are very **beautiful**$_{MASC}$ > beautiful$_{FEM}$').

In this case, the error is visible on the adjective although the error goes beyond the token level, affecting a grammatical structure (the sentence, in this example). Technically it is possible to annotate at the token level, but conceptually maybe this is not the ideal representation of the error. One simple example of an error that cannot be annotated at the token level is the following, where two tokens have to be corrected into one:

(2) *Foi uma expêriencia que eu nunca **tenho esquecido*** > *esqueci* ('It was an experience that I **haven't forgotten** > forgot').

Our next step will be to automatically generate the tags of the fine-grained tagset from the token-based annotations. We will do it through conversion scripts that take as input all the XML annotations and generate as output a new XML with the corresponding standoff annotations (tag + correction suggested). We have done the calculations and it is possible to generate (fully or partially) 29 of the 38 tags. From the remaining 9 tags, 6 go beyond the token, affect mainly the verbal phrase and correspond to rare errors. One example is the tag GVH, for errors affecting verbal periphrasis, like in:

(3) *Espero que não **va acontecer** > va a acontecer* ('I hope it is not **going happen** > going to happen').

The other 3 tags are token-based but require human interpretation. One example is the tag LN (Lexical+Nonexistent_Word), for the cases where the student created a new word (that does not exist in Portuguese) using recognizable morphological processes, as in:

(4) *e **estabilitamos** a melhor relação > estabelecemos* ('and we **establish** the best relation').

In this example the student created a new verb *estabilitar*, probably from the adjective *estável* (stable), instead of using *estabelecer* (to establish).

5. Conclusions and future work

We have implemented a system for error annotation in COPLE2 that attempts to reduce manual effort by taking advantage of the corpus information and the possibilities of the TEITOK environment. We have started to apply the system, and we have already annotated 47% of the corpus at the token level, being COPLE2 the first Portuguese learner corpus with error annotation. From in-line, token-based and flat annotations we will generate automatically standoff, multi-level annotations, which will contain position-based tags covering 38 error types. Most of the tags will be fully generated using this automatic approach, although some of them will require manual work.

Currently, we continue annotating at the token level and developing the scripts for the automatic generation of tags. Besides this, we have identified some future lines of work. First of all, we need to explore how to transform the multi-token in-line annotations into tags, reducing as much as possible the manual effort. One way could be to identify error patterns (using information concerning the word form, POS, word order, etc.) in multi-token

[3] For the moment, only one annotator is performing the task. In the future, we would like to count with at least two different annotators.

[4] All the examples are from COPLE2 and have the following format: the error is marked in bold, the correction is shown after the ">" symbol and a translation in English (with the corresponding correction) is provided.

structures that correspond to a certain tag, automatizing the generation. A second line of work is related to the addition of new linguistic levels for error annotation, like semantics or discourse. In fact, some annotation cases at the token level suggest the need of higher linguistic levels of abstraction in the scheme.

We believe that error annotations (token-based plus error tags) together with all the information already stored in the corpus (metadata, student's modifications, teacher's corrections) will allow for complex and rich linguistic queries in COPLE2. We expect that this information can be useful for researchers of different fields like Second Language Acquisition, Foreign Language Teaching and Learning or Computer Assisted Language Learning.

6. Acknowledgements

This work was partially supported by Fundação Calouste Gulbenkian (Proc. nr. 134655), Fundação para a Ciência e a Tecnologia (project PEst-OE/LIN/UI0214/2013; postdoctoral research grant SFRH/BPD/109914/2015) and Associação para o Desenvolvimento da Faculdade de Letras da Universidade de Lisboa (ADFLUL).

7. Bibliographical References

Andersen, Ø. (2011). Semi-automatic ESOL error annotation. *English Profile Journal*, 2.Christ, O., Schulze, B., Hofmann, A. and Koenig, E. (1999). *The IMS Corpus Workbench: Corpus Query Processor (CQP): User's Manual*. Institute for Natural Language Processing. University of Stuttgart. (CQP V2.2).

Dagneaux, E., Denness, S., Granger, S., Meunier, F., Neff, J. and Thewissen, J. (Eds.) (2005). *Error Tagging Manual. Version 1.2*. Centre for English Corpus Linguistics. Université Catholique de Louvain.

Dagneaux, E., Denness, S. and Granger, S. (1998). Computer-aided Error Analysis. *System*, 26:163–174.

del Río, I., Antunes, S., Mendes, A. and Janssen, M. (2016). Towards error annotation in a learner corpus of Portuguese. In *Proceedings of the 5th NLP4CALL and 1st NLP4LA workshop in Sixth Swedish Language Technology Conference (SLTC)*. Umeå University, Sweden, 17-18 November.

Díaz-Negrillo, A. and Thompson, P. (2013). Learner corpora: Looking towards the future. In N. Ballier, A. Diaz-Negrillo, & P. Thompson (Eds.), *Automatic treatment and analysis of learner corpus data* (pp. 249–264). Amsterdam & Philadelphia: John Benjamins.

Díaz-Negrillo, A. and Fernández-Domíguez, J. (2006). Error Tagging Systems for Learner Corpora. *RESLA*, 19:83--102.Granger, S. (2004). Computer learner corpus research: current status and future prospects. In U. Connor & T. Upton (Eds.), *Applied Corpus Linguistics: A Multidimensional Perspective* (pp. 123-145). Amsterdam & Atlanta: Rodopi.

Granger, S. (2003). Error-tagged Learner Corpora and CALL: A Promising Synergy. *CALICO Journal* 20 (3). Special issue on error analysis and error correction in computer-assisted language learning, pp. 465--480.

Kutuzov, A. and Kuzmenko, E. (2015). Semi-automated typical error annotation for learner English essays: Integrating frameworks. In *Proceedings of the 4th workshop on NLP for Computer Assisted Language Learning* at NODALIDA 2015, Vilnius, 11th May, 2015, Volume , Issue 114, 2015-05-06, pp. 35-41.

Lüdeling, A., Walter, M., Kroymann, E. and Adolphs, P. (2005). Multi-level annotation error annotation in a learner corpora. In *Proceedings of Corpus Linguistics* 2005 1, Birmingham (England), July 2005, 14-17. Rosen, A., Hana, J., Štindlová, B. and Feldman, A. (2013). Evaluating and automating the annotation of a learner corpus. *Language Resources and Evaluation*, pp. 1--28.

Tono, Y. (2003). Learner corpora: Design, development and applications. In D. Archer, P. Rayson, A. Wilson and T. McEnery (Eds.), *Proceedings of the Corpus Linguistics 2003 Conference*. Lancaster University, pp. 800--809.

8. Language Resource References

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Štindlová, B. and Vettori, C. (2014). The MERLIN corpus: Learner Language and the CEFR. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pp.1281--1288.

Granger, S., Dagneaux, E., Meunier, F., Paquot, M. (Eds.) (2009). International Corpus of Learner English. Version 2. UCL: Presses Universitaires de Louvain.

Janssen, M. (2016). TEITOK: Text-Faithful Annotated Corpora. In *Proceedings of LREC 2016*, Portorož, Slovenia.

Janssen, M. (2012). NeoTag: a POS Tagger for Grammatical Neologism Detection. In *Proceedings of LREC 2012*, Istanbul, Turkey.

Mendes, A., Antunes, S., Janssen, M. and Gonçalves, A. (2016). The COPLE2 Corpus: a Learner Corpus for Portuguese. In *Proceedings of LREC 2016*, Portorož, Slovenia.

Nicholls, D. (2003). The Cambridge Learner Corpus – error coding and analysis for lexicography and ELT. In D. Archer, P. Rayson, A. Wilson and T. McEnery (Eds.), *Proceedings of the Corpus Linguistics 2003 Conference*. Lancaster University, pp. 572--581.

An SLA Corpus Annotated with Pedagogically Relevant Grammatical Structures

Leonardo Zilio, Rodrigo Wilkens and Cédrick Fairon
Centre de Traitement Automatique du Langage - CENTAL
Université catholique de Louvain - UCL
{leonardo.zilio, rodrigo.wilkens, cedrick.fairon}@uclouvain.be

Abstract

The evaluation of a language learner's proficiency in second language is a task that normally involves comparing the learner's production with a learning framework of the target language. One of the most well known frameworks is the Common European Framework for Languages (CEFR), which addresses language learning in general and is broadly used in the European Union, while serving as reference in countries outside the EU as well. In this study, we automatically annotated a corpus of texts produced by language learners with pedagogically relevant grammatical structures and observed how these structures are being employed by learners from different proficiency levels. We analyzed the use of structures both in terms of evolution along the levels and in terms of level in which the structures are used the most. The annotated resource, SGATe, presents a rich source of information for teachers that wish to compare the production of their students with those of already certified language learners.

Keywords: SLA, syntactic annotation, CEFR, EFCAMDAT, learner profile

1. Introduction

The evaluation of a language learner's capacity when producing texts in a foreign language is not an easy task. The factors that impact the overall categorization of the produced text are many, roughly ranging from vocabulary to discourse strategies, passing by syntax and semantics. One way of facilitating this task is to have a profile of the language learner skills, so that there are some hints on what to expect from a learner in each language level.

For this reason, there are different frameworks that organize the order in which the different language skills should be targeted at each step, while also indicating the required skills for the evaluation of a learner's proficiency. Examples of this type of frameworks are the Common European Framework for Languages (CEFR) and the Cambridge ESOL, which are based on levels, and IELTS and TOEFL, which are based on scores. These frameworks pinpoint, in differently organized fashion, how it is expected that the second language learning will take place for the learner, by listing skills and associating them with an expected level (or score).

In this study, we use a different approach. Instead of pointing out the skills that the learner should be able to master in order to be evaluated as having achieved a certain level of proficiency, our objective is to look directly at the production of learners that have already been evaluated as having achieved a certain degree of proficiency. By investigating texts produced by learners and quantitatively observing how different types of grammatical structures are used by learners from different language levels and by analyzing the distribution of grammatical structures in their textual production, we aim at finding out which structures are more or less active and how they evolve in frequency along the different language mastery levels.

For describing the distribution of grammatical structures in texts produced by language learners, we annotated an SLA corpus with pedagogically relevant grammatical structures, which are referred to in, for instance, learner's grammars and the English Grammar Profile (EGP). The resource, which we named *SLA in Grammatically Annotated Texts* (SGATe), contains more fine-grained information than it would be possible to retrieve from common parsing methods, and it provides teachers with an interesting tool for comparing the written production of their own students with the annotations that are present in the corpus, which show the use of grammatical structures by certified learners.[1]

This paper is organized as follows: Section 2. describes language learning frameworks and systems that provide annotation of grammatical information; Section 3. describes the EF-Cambridge Open Language Database (EFCAMDAT) (Geertzen et al., 2013) and the annotation process, while also describing a precision evaluation of annotated structures; Section 4. presents the evaluation of a sample of the annotated data; in Section 5. we discuss the results of the annotation, by presenting more detailed information on the distribution of grammatical structures in the corpus; and Section 6. is where we present our final remarks on this study.

2. Related Work

There are different frameworks that describe how a second language should be learned and that focus on evaluating when a given learner has achieved a certain level of proficiency, such as the Common European Framework of Reference (CEFR), the Cambridge ESOL, the TOEFL, and the IELTS. For this study, the CEFR is of special relevance.

[1]The annotations were added on top of EF-Cambridge Open Language Database (EFCAMDAT) (Geertzen et al., 2013) and can be found at the EFCAMDAT Website: `https://corpus.mml.cam.ac.uk/efcamdat2/`. Alternatively, the same annotations, together with other material related to this paper, can be found at the following Website: `http://cental.uclouvain.be/resources/smalla_smille/sgate/`.

The CEFR (Verhelst et al., 2009) presents a guide in terms of language levels and content that is meant to serve as a parameter for the teaching of foreign languages in the European Union. It provides a description of communication goals that a language learner should achieve in each of six main levels: A1, A2, B1, B2, C1, and C2. As a general guide that was not designed to cover specific languages, but to present broad communicative guidelines, it leaves various gray areas in terms of the learning process, so that the information for each level does not cover the different needs of a language learner regarding specific grammar and vocabulary content, or even a specific language. As such, the curricula of different language courses do not need to be necessarily the same even if they follow the specified CEFR levels (Alderson, 2007; Little, 2007).

Since we intend to observe the distribution of grammatical structures in a corpus of written production, it is also important to consider systems that were developed for annotating pedagogically relevant information. In this regard, we have the FLAIR and the SMILLE systems, both of which annotate grammatical structures based on the CEFR and use similar methods for annotating them.

The *FLAIR* system (Chinkina et al., 2016; Chinkina and Meurers, 2016) is described as an online information retrieval system that uses efficient algorithms to retrieve, annotate and rerank Web documents based on the grammatical constructions they contain. FLAIR searches online documents based on keywords selected by the user, parses the first twenty documents retrieved by the search engine and ranks them according to the settings the user selected as most important. It can recognize 87 different types of grammatical structures described in the official English language curriculum of schools in Baden-Württemberg (Chinkina et al., 2016).

The SMILLE system (Zilio and Fairon, 2017; Zilio et al., 2017a; Zilio et al., 2017b) has as its main focus the recognition of grammatical structures in online texts chosen by language learners, so that these structures can be highlighted in the text, thus aiding the learner to notice them while reading the text. SMILLE's grammatical annotation tool can recognize up to 107 different grammatical structures that were derived from Altissia's[2] pedagogical curriculum for the English language, which is based on the CEFR.

Both FLAIR and SMILLE use the Stanford Parser (Manning et al., 2014) for lemmatization, part-of-speech tagging, and dependency parsing, and then retrieve more complex, pedagogically relevant grammatical information by means of a set of rules specific to the structures that are to be recognized in the text. The main difference between the systems in terms of annotation of the grammatical information is the selection of structures that are annotated. In this regard, SMILLE presents the possibility of annotating more fine-grained structures, such as different types of gerunds and of infinitives with "to".

3. Methodology

For being able to describe how language learners actually use the grammatical structures they learn, we selected the EF-Cambridge Open Language Database (EFCAMDAT) (Geertzen et al., 2013), which presents a collection of texts produced by learners of English from different levels of proficiency. The corpus is divided according to the Common European Framework of Reference for languages (CEFR) (Verhelst et al., 2009) and contains a total of 532 thousand documents (33 million tokens) written by 83,385 learners from 137 countries. Each document has a score indicating how well the learner performed in the task and is linked to a specific topic (e.g. "introducing yourself by email"). The data is distributed into three main levels, each one with two sublevels (all referenced by a letter and a number): basic (breakthrough or A1, and waystage or A2), independent (threshold or B1, and vantage or B2), and proficient (effective operational proficiency or C1, and mastery or C2).

We used the SMILLE system (Zilio and Fairon, 2017; Zilio et al., 2017a; Zilio et al., 2017b) to annotate the corpus with 107 grammatical structures that are pedagogically relevant and analyzed their distribution on the different language levels. Here is an abbreviated list of the structures that the system can recognize: articles, adjectives, adverbs, personal pronouns, demonstrative pronouns, emphatic pronouns, possessive pronouns, reflexive pronouns, quantifiers, nouns, numerals, plural with special endings, plural nouns, irregular verb forms, modals, semi-auxiliaries, prepositional verbs, different types of infinitives with "to" and infinitives without "to", ellipsed infinitive, gerunds, subjects of gerund, participles, verb tenses (including perfect and continuous aspects), imperatives, passive voice, conditionals and forms of expressing hypothesis, connectives, relative clauses, sentences with "have got", question tags, short answers, wh-questions, short forms, genitives, and verb "wish" followed by past or past perfect[3].

These grammatically rich annotations on top of the EFCAMDAT corpus gave birth to SGATe, namely *SLA in Grammatically Annotated Texts*, a resource in which it is possible to observe the second language acquisition of different learners in terms of pedagogically relevant grammatical structures. Since many of these structures are complex and require rules on top of parser information for being annotated, and since the automatic annotation was not conducted on texts produced by native speakers of English, we performed an evaluation in terms of precision, which is described in the following subsection.

3.1. Evaluation of the Automatic Annotation

The annotations in the SGATe (SLA in Grammatically Annotated Texts) resource were manually evaluated in terms of precision by one linguist. This evaluation was designed to verify how well SMILLE's handcrafted rules can annotate a corpus of texts produced by learners, including even the most basic levels. Since the structures are automatically annotated, the evaluation also contributes to show in which of the annotated structures we can rely on for analyzing the annotated data. The evaluation was carried out on a random sample of the corpus: we extracted a sample

[2] www.altissia.com.

[3] In section 3.1., we present example sentences for some specific structures that we evaluated in this study.

of 40 documents from each of the 6 CEFR levels, totaling 240 documents.

Since it would be impossible to evaluate every grammatical structure that were annotated by SMILLE's system in SGATe, and since some of them simply rely on parser morphosyntactic annotation, we selected 33 structures to be evaluated that do not rely on morphosyntactic annotation[4] and that present a general overview of the features that SMILLE can annotate. Here is a list of these structures with a simple example sentence for each of them (the main words associated with the structure are marked in *italic*):

1. **Gerunds after preposition:**
   They were accused *of breaking* into a shop.
2. **Gerunds as complement of a verb:**
   We *enjoyed meeting* your friends.
3. **Gerunds instead of infinitive (no change of meaning):**
   They continued *working* hard.
4. **Gerunds instead of infinitive (change of meaning):**
   I remember *visiting* this place before.
5. **Gerunds as subject of a verb:**
   Traveling broadens the mind.
6. **Adjective + infinitive with "to":**
   I'm very *pleased to meet* you.
7. **Noun + infinitive with "to":**
   I've some *work to finish*.
8. **Verb + infinitive with "to":**
   He *refused to come*.
9. **Verbs "let" or "make" + infinitive without "to":**
   The film *made* me *cry*.
10. **Expression "let's" + infinitive without "to":**
    Let's play tennis this afternoon.
11. **Infinitive without "to" after "rather" or "better":**
    I'd *rather have* told him myself.
12. **Present perfect continuous:**
    She *has been waiting* for one hour.
13. **Past perfect continuous:**
    I *had been waiting* for one hour when the bus arrived.
14. **Future perfect:**
    I*'ll have finished* work by 5 o'clock tonight.
15. **Imperative:**
    Shut that door!
16. **Passive voice:**
    He *was seen* in London.
17. **Adverbs with passive voice:**
    The incident was *quickly* forgotten.
18. **Connectives:**
    I will call you *as soon as* I need help.[5]
19. **Relative clauses:**
    The man *who is sitting there* is my boss.
20. **First conditional:**
    If it *rains*, I'll stay at home.
21. **Second conditional:**
    If I *stopped* smoking, I could run faster.
22. **Third conditional:**
    If you *had taken* the exam, you might have passed it.
23. **Hypothesis with "would":**
    If I had more money, I *would buy* some new clothes.
24. **Hypothesis with "would have":**
    If I had studied hard, I *would have passed* the exam.
25. **Prepositional verbs:**
    I *agree with* you.
26. **Phrasal verbs:**
    Please *come in*, the doctor is expecting you.
27. **Verb "wish" followed by past:**
    I *wish* I *had* a car.
28. **Genitive marker:**
    It is *John's* book.
29. **Quantifiers:**
    There are *some* books left.
30. **Special forms of plural:**
    There are two *knives* on the kitchen table.
31. **Semi-auxiliaries:**
    We *haven't got to* read that book.
32. **Question tags:**
    It's cold today, *isn't it*?
33. **Wh-questions:**
    Why have you come so late?

4. Results

We excluded from the precision results those annotation errors that were caused by bad spelling or structural organization of sentences, but these were a minor issue, representing only 1.24% of the annotated sample data. The overall precision of the system for the evaluated structures was 90.10% (weighed precision: 92.46%), with median at 97.50%. When we looked at the differences from level to level, we see a very bad overall precision at level A1, and then no palpable difference between the other levels, but a

[4]There is only one structure that is based on morphosyntactic annotation, and that is the genitive marker. We included this annotation, because it is an important grammatical structure of the English language, and sometimes it poses a problem for learners.

[5]This is an example of connective of time, but several types of connectives were evaluated. Here is the full list of types of connectives: time, comparison, alternative, reason, purpose, condition, opinion, addition, explanation, and summary. Although we grouped them as one type of structure, we evaluated them also separately, as it will be further discussed in Section 4.

Table 1: Precision scores for the evaluated structures

CEFR levels	Overall precision (%)	Weighed precision (%)
A1	58.54	67.21
A2	89.84	91.05
B1	91.75	91.40
B2	90.89	91.70
C1	91.02	90.48
C2	90.81	90.65

much higher overall precision score, as can be seem in Table 1. There are many possible reasons for this discrepancy from the A1 level to the others, one of them is the possibility that A1 documents present a writing style that lacks naturalness, and this makes it harder for the parser and for the system's rules to recognize the correct text patterns required for annotating the grammatical structures.

Table 2 shows the precision scores for the different evaluated structures in our sample of SGATe. Although most of the structures had a very good precision overall, some structures had bad performance, like gerunds as subject of verb, that had an overall precision of 55.56%, and did not perform well in any level. Other structures, like imperatives, had a not so high performance overall (82.03%), but performed very well if we exclude the A1 and A2 Levels (90.48%). The same is true for the genitive marker, which performed very badly in Levels A1 through B1, which pulled its overall performance down to 67.53%, but actually got a nice score in the higher levels (90.20%). We verified a similar result regarding connectives, but this time in terms of granularity. The evaluation of connectives showed a precision of 87.06%, but we also observed that two classes of connectives, namely connectives of example and connectives of purpose, had a much lower precision score (58.02% and 63.36%, respectively), which was compensated by the good performance of the other classes. The high level of precision for most of the annotated structures ensures that the annotated resource can be used for a deeper analysis of its content.

5. Profile of Grammatical Structures per Level

The SGATe (SLA in Grammatically Annotated Texts) resource comprises the entire EF-Cambridge Open Language Database (EFCAMDAT) (Geertzen et al., 2013) annotated with 107 pedagogically relevant grammatical structures. However, for diving deeper into the grammatical structures, we did not use the whole corpus, as we explain in this section. For a first exploratory analysis we used a linear regression algorithm for detecting the tendency of progression in the structures distribution along the different levels. In terms of selection of grammatical structures for this observation, we analyzed verb tenses and other structures that depend only on the parser's morphosyntactic information, and, from the structures that we evaluated in this study, we selected only those that had precision scores above 80%.

As we observed in Section 3.1., the automatic annotation doesn't perform well in Level A1, so, for the profiling presented here, we excluded data from that level. We also filtered out documents from the corpus for which the score was lower than 80%[6], because texts with lower scores may present some errors that can badly interfere with the automatic annotation. We also balanced as best as we could the number of texts from each level, so that Levels A2, B1 and B2 had 9 thousand documents each, and C1 had more than 4 thousand documents[7]. As a final step, we normalized the frequency of the grammatical structures in each level by using a frequency-per-sentence score, which was further converted to logarithm, to compensate for the fact that language data tend to appear in a Zipf distribution.

After this balancing and normalization process that was performed on SGATe data to give us a more reliable information on the tendency of use of grammatical structures along the levels, we divided the structures in three categories, regarding their tendency to evolve along the levels:

Table 2: Precision scores for the evaluated structures

	Structure	Total	Precision
1	Gerunds after preposition	109	95.41%
2	Gerunds as complement	3	100.00%
3	Gerunds (no change of meaning)	18	100.00%
4	Gerunds (change of meaning)	2	100.00%
5	Gerunds as subject of a verb	9	55.56%
6	Adjective + infinitive with "to"	93	94.62%
7	Noun + infinitive with "to"	75	74.67%
8	Verb + infinitive with "to"	370	91,35%
9	"let"/"make" + infinitive	25	88.00%
10	"let's" + infinitive	2	100.00%
11	"rather"/"better" + infinitive	1	100,00%
12	Present perfect continuous	12	100.00%
13	Past perfect continuous	2	100.00%
14	Future perfect	3	100.00%
15	Imperatives	128	82.03%
16	Passive voice	233	87.12%
17	Passive adverbs	29	72.41%
18	Connectives	765	87,06%
19	Relative clauses	103	94,17%
20	First conditional	38	100.00%
21	Second conditional	12	100.00%
22	Third conditional	1	100.00%
23	Hypothesis: "would"	48	97.92%
24	Hypothesis: "would have"	1	100.00%
25	Prepositional verbs	199	97.49%
26	Phrasal verbs	104	96.15%
27	"wish" followed by past	1	100,00%
28	Genitive marker	77	67.53%
29	Quantifiers	320	97,50%
30	Special forms of plural	109	99.08%
31	Semi-auxiliaries	79	75,95%
32	Question tags	3	100,00%
33	Wh-questions	35	97,14%

[6]This is based on the actual score that was given to the texts by L2 evaluators of the Cambridge University while assessing the learner's performance on an exam.

[7]All documents with scores above 80% were included in the C1 data. C2 documents were too few to include.

increasing tendency (angle of the line above 30 degrees), decreasing tendency (angle of the line below -30 degrees) and neutral tendency (angle of the line between -30 and 30 degrees). As a means of ensuring the reliability of our results, we considered the linear tendency reliable only if the error of the slope in the linear model scored below 0.15.

We present here the structures divided by category with information about the angle in brackets. These are the structures with an increasing tendency: adjectives followed by infinitive with "to" (53°), relative clauses (53°), gerunds after preposition (52°), past perfect tense (51°), passive voice (45°), and gerunds as complement of a verb (34°). Two examples of these structures are plotted in Figure 1. These are the structures that tend to be roughly equally used along the levels A2 to C1: imperatives (-3°), verbs "let" or "make" + infinitive without "to" (13°), present participles (3°), and present simple (-1°). Two examples of these structures can be seen in Figure 2. Finally, these are the structures that presented a decreasing tendency: short forms (-43°), present continuous (-43°), and gerunds instead of infinitive (no change of meaning) (-35°). We plotted two examples of these structures in Figure 3.

The tendency lines presented some interesting information, like the decrease in the use of short forms and the present continuous, while the past perfect and relative clauses get more used. Passive voice also has an increasing tendency, which is expected, because it is considered to be a more complicated structure to master.

Since many structures do not present a clear ascending, descending or neutral tendency (i.e., the error of the slope was 0.15 or higher), probably presenting more prominent uses in different levels, we also looked at the peaks of use of each grammatical structure. For doing this, we used the same data that was normalized by sentence, and we looked in which levels the structures occurred the most (considering a confidence interval of 95% for determining if the difference was significant). Structures that occurred the most at Level A2 were the following: short forms, past simple, past simple of the verb "to be", past simple of the verb "to have", past continuous, present simple of the verb "to do", present continuous, and gerunds instead of infinitive (no change of meaning). These are the structures that occurred the most at Level B1: use of "going to", future perfect, future, and expression "let's" followed by infinitive. Structures that occurred the most at Level B2 were the following: genitive markers, present participles, and present simple of the verb "to be". These are the structures that occurred the most at Level C1: first conditional, second conditional, hypothesis with "would", future continuous, gerunds after preposition, imperatives, passive voice, past perfect, past perfect continuous, present perfect of the verbs "to be" and "to have", present perfect continuous, present perfect, present simple of the verb "to be", relative clauses, verbs "let" or "make" + infinitive without "to", adjective + infinitive with "to", verb + infinitive with "to", and connectives.

With this second analysis, we could observe that the verb tenses are well distributed along the corpus, with present simple and present continuous, and past simple of auxiliary verbs at level A2, followed by future at Level B1, and then the perfect tenses at Level C1. Connectives are also more concentrated on Level C1, which was a bit of a surprise, since, for instance, the English Grammar Profile tends to present them as lower level structures. The same is true for first and second conditionals, which are normally regarded as A2 or B1-level structures, but have a greater concentration at level C1. This is maybe a sign of the difference between the time of learning and the actual mastery of the grammatical structure.

6. Conclusion

In this paper, we described the automatic annotation of the EF-Cambridge Open Language Database (EFCAMDAT), a corpus of texts produced by language learners, with pedagogically relevant grammatical information. This layer of annotation that was added to the EFCAMDAT originated a resource that we called SGATe (SLA in Grammatically Annotated Texts) and that allowed us to analyze the distribution of grammatical structures in the production of language learners. As such, we could describe the actual active use of structures by the learners.

On top of the data from SGATe, we used a linear regression and later an analysis of peaks of occurrence to determine the behavior of grammatical structures in the corpus. This presented us with some expected results, such as passive voice being more used in higher levels, but also showed some interesting results, like the predominance of use of connectives in C1 Level, as opposed to lower levels, as is described in the English Grammar Profile.

The new layer of annotation presented in SGATe allows for teachers to observe how learners tend to employ the grammatical content that is learned, but also allows researchers to observe how the different structures are distributed in the corpus. Considering that EFCAMDAT comprises 137 different nationalities, one further point of interest would be to observe which type of influence the different mother tongues may have on the written production of learners of English.

7. Acknowledgements

The authors would like to thank the Walloon Region (Projects BEWARE n. 1510637 and 1610378) for support, and Altissia International for research collaboration.

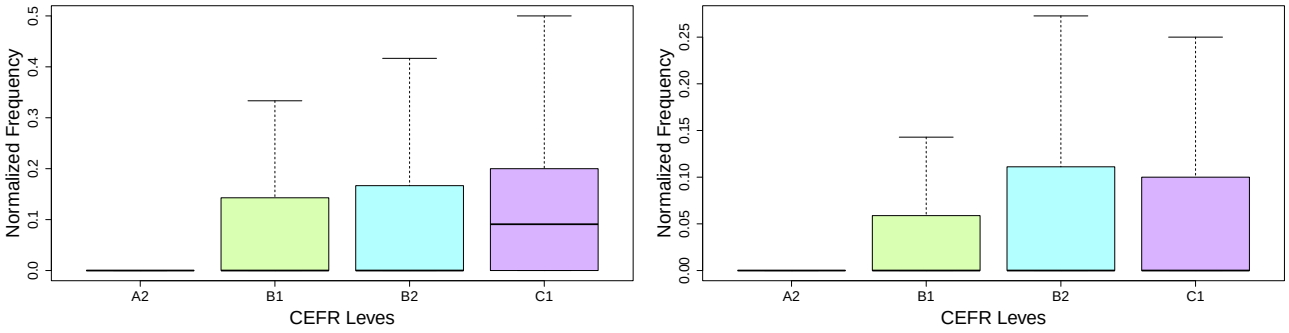
8. Bibliographical References

Alderson, J. C. (2007). The cefr and the need for more research. *The Modern Language Journal*, 91(4):659–663.

Chinkina, M. and Meurers, D. (2016). Linguistically aware information retrieval: providing input enrichment for second language learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications, San Diego, CA*.

Chinkina, M., Kannan, M., and Meurers, D. (2016). Online information retrieval for language learning. *ACL 2016*, page 7.

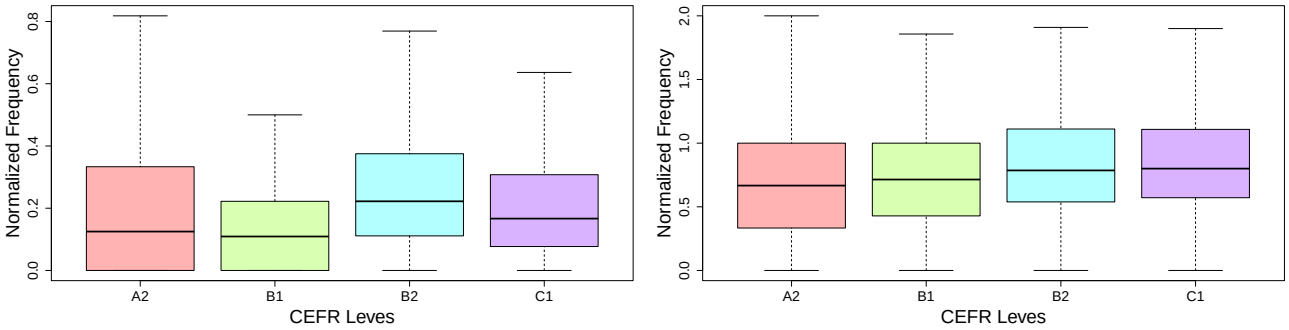
Geertzen, J., Alexopoulou, T., and Korhonen, A. (2013). Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*.



(a) Passive

(b) Relative Clauses

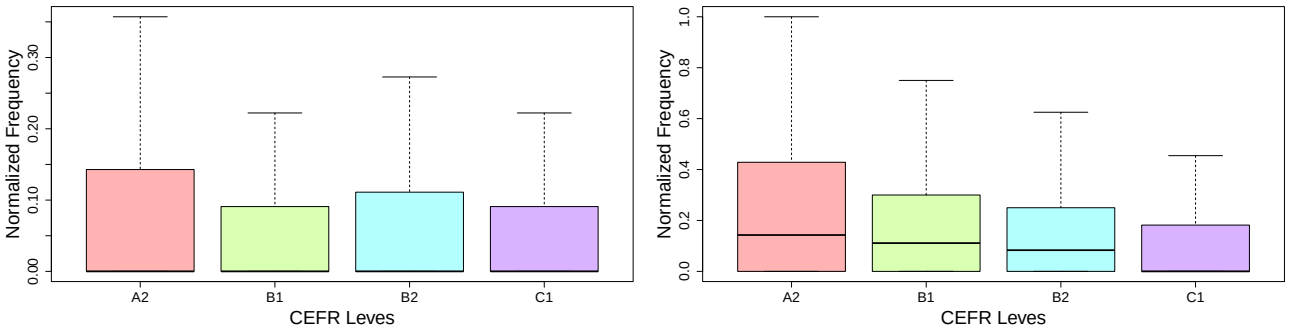
Figure 1: Examples of structures that have a tendency to be progressively more prominent along the language levels.



(a) Present Participle

(b) Present Simple

Figure 2: Examples of structures that have a tendency to be equally used along the language levels.



(a) Present Continuous

(b) Short Form

Figure 3: Examples of structures that have a tendency to be progressively less prominent along the language levels.

Little, D. (2007). The common european framework of reference for languages: Perspectives on the making of supranational language education policy. *The Modern Language Journal*, 91(4):645–655.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Verhelst, N., Van Avermaet, P., Takala, S., Figueras, N., and North, B. (2009). *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press.

Zilio, L. and Fairon, C. (2017). Adaptive system for language learning. In *Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on*, pages 47–49. IEEE.

Zilio, L., Wilkens, R., and Fairon, C. (2017a). Enhancing

grammatical structures in web-based texts. In *Proceedings of the 25th EUROCALL*, pages 839–846. Accepted.

Zilio, L., Wilkens, R., and Fairon, C. (2017b). Using nlp for enhancing second language acquisition. In *Proceedings of Recent Advances in Natural Language Processing*, pages 839–846.

9. Language Resource References

Geertzen, J., Alexopoulou, T., and Korhonen, A. (2013). Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*.

Portable Spelling Corrector for a Less-Resourced Language: Amharic

Andargachew Mekonnen Gezmu, Andreas Nürnberger, Binyam Ephrem Seyoum
Otto-von-Guericke-Universität Magdeburg, Addis Ababa University
Otto-von-Guericke-Universität Magdeburg,
Fakultät für Informatik,
Data and Knowledge Engineering Group,
Universitätsplatz 2,
39106 Magdeburg.
{andargachew.gezmu, andreas.nuernberger}@ovgu.de, binyam.ephrem@aau.edu.et

Abstract
This paper describes an automatic spelling corrector for Amharic, the working language of the Federal Government of Ethiopia. We used a corpus-driven approach with the noisy channel for spelling correction. It infers linguistic knowledge from a text corpus. The approach can be ported to other written languages with little effort as long as they are typed using a QWERTY keyboard with direct mappings between keystrokes and characters. Since Amharic letters are syllabic, we used a modified version of the System for Ethiopic Representation in ASCII for transliteration in the like manner as most Amharic keyboard input methods do. The proposed approach is evaluated with Amharic and English test data and has scored better performance result than the baseline systems: GNU Aspell and Hunspell. We get better result due to the smoothed language model, the generalized error model and the ability to take into account the context of misspellings. Besides, instead of using a handcrafted lexicon for spelling error detection, we used a term list derived from frequently occurring terms in a text corpus. Such a term list, in addition to ease of compilation, has also an advantage in handling rare terms, proper nouns, and neologisms.

Keywords: spelling corrector, corpora, noisy channel, less-resourced language

1. Introduction

Documents in many languages have been digitized and are available in different media especially on the web. Giant software vendors (e.g., Google and Microsoft) are also localizing their products to the native languages of their target customers. There is a need to develop computational solutions to the classic problems of computational linguistics for the respective languages. Spelling error detection and correction are among the oldest computational linguistics problems (Blair, 1960). Spelling correction is considered from two perspectives: non-word and real-word correction. When typographical or cognitive errors accidentally produce valid words we get real-word errors, otherwise, we get non-word errors. These problems are mostly treated separately. In this paper, we dealt with non-word errors.

Many spelling correctors are developed using rule-based approaches. However, it is difficult to develop and maintain all language-dependent rules (Norvig, 2009). In addition, such systems limit themselves to isolated-word correction without considering the context. Therefore, we proposed and evaluated an approach that takes into account the context of misspellings and infers linguistics knowledge from a text corpus.

2. Related Work

Earliest research on spelling correction is based on phonetic and string similarities such as Metaphone and Damerau-Levenshtein edit distance algorithms (Damerau, 1964). Candidate corrections are ranked from manually compiled lexicons with the help of these algorithms. GNU Aspell and Hunspell are good examples that follow this approach. Mekonnen (2012) has followed the same approach for Amharic. In a related approach, Ahmed et al. (2009) used similarity scores of letter n-grams to rank candidate corrections. In these approaches, the lexicons along with some linguistics rules are used for spelling error detection. Yet there was also an attempt to detect errors without using lexicons (Morris and Cherry, 1975). This approach depends on n-gram letter-sequences from a target text. It generates an "index of peculiarity"; and based on the index, determines which words are spelling errors in the target text. For example, the typo 'exmination' contains 'exm' and 'xmi', trigrams which are peculiar and will be included in the list. Even though this approach has the advantage of being language independent and appears to work for less-resourced languages, many misspellings do not contain the unusual n-grams and so would not appear in the list (Mitton, 2010).

Recent research on spelling error correction focuses on using some web corpora to infer knowledge about spellings. Most of these systems are based on the noisy channel model (Kernighan et al., 1990; Kukich, 1992; Brill and Moore, 2000; Whitelaw et al., 2009; Gao et al., 2010). Also, additional features of spellings such as phonetic similarities and modified edit distance (e.g., Winkler (2006)) are used to generate plausible candidates for spelling correction (Toutanova and Moore, 2002).

3. Approach

Like other Semitic languages, word formation in Amharic depends mainly on root-and-pattern morphology and exhibits prefixes, suffixes, and infixes. Amharic is morphologically-rich in the way that grammatical relations and syntactic information are indicated at the word level. These features are some of the main hurdles for rule-based computational processing (Fabri et al., 2014). It is difficult to develop and maintain all language-dependent rules for spelling correction especially when the languages have complex morphology like Amharic (Norvig, 2009). Thus, we have applied a data-driven (corpus-driven) approach with the noisy channel for spelling correction. According to the noisy channel approach, for a misspelled word x, the most likely candidate correction w_n out of all possible candidate corrections C with $w_1w_2...w_{n-1}$ preceding words context is suggested by the maximum probability of $P(w_n|w_1w_2...w_{n-1}x)$, which is computed by Equation 1

below. $P(w_1w_2 \ldots w_{n-1}w_n)$ is the prior probability and $P(x|w_n)$ the likelihood where both are represented in the language and error models; see sections 3.1 and 3.2 for details. Obviously, x is conditionally dependent only on w_n and assumes the preceding words are correct.

$$\text{argmax}_{w_n \in C} P(w_1w_2 \ldots w_{n-1}w_n)P(x|w_n) \qquad (1)$$

Based on the proposed approach, the spelling error detection and correction processes are as follows. An input word that is not in the term list, which is compiled from the most frequent words in a text corpus, is flagged as a spelling error. Candidate corrections that are closer (nearer) to the misspelling are generated from the term list. For language independence, we measure nearness using Damerau-Levenshtein edit distance (Damerau, 1964). Since most of the misspellings fall within two edit distance from their corrections (Damerau, 1964; Gezmu et al., 2017), we selected all words in the term list that are one up to two edit distance from the misspelled word. Then the candidates are scored and ranked according to their prior and likelihood probabilities. In case there is no candidate correction, the misspelled term will be split. This step is needed to correct misspellings result from missed out spaces between words, like የሃንስነገይመጣል. The correction is to segment the expression as የሃንስ /johannɨs/[1], ነገ /nəgə/ and ይመጣል /jɨməṭ'al/.

3.1 Language Model and Corpora

In a text corpus, linguistic knowledge resides in the n-grams of the corpus and it is possible to acquire new knowledge using a large number of documents. It also contains rare terms, proper nouns, technical terms, brand names, and newly coined terms (neologisms). Manually compiled or handcrafted lexicons normally do not include most of these types of terms. But by using the most frequent words in the corpus, we can easily create a term list that incorporates the most widely used terms.

Tachbelie and Menzel (2007) evaluated n-gram word-based and morpheme-based Amharic language models. They have used a news corpus to build the models. The word-based model smoothed with the Kneser-Ney (Kneser and Ney, 1995) method has a better performance on a test data than the morpheme-based model. The result complies with the empirical study of Chen and Goodman (1998). The researchers have found that among the best performing n-gram smoothing methods is the Kneser-Ney with its modified version. To this end, we build a trigram word-based language model smoothed with the modified Kneser-Ney method.

For Amharic language model, being a less-resourced language, the only available sizable text corpora are HaBiT (HaBiT, 2016; Rychlý and Suchomel, 2016) and Crúbadán (Scannell, 2007). Both are created from automatically crawled web pages. HaBiT contains about 17.6 million tokens (words) whereas Crúbadán contains about six million tokens. Except for their size difference, both corpora are essentially the same. Since they obtain text from all types of web documents, we expected spelling errors in these corpora. We have found out that they contain a number of spelling errors through a manual check. Therefore, we build our own Contemporary Amharic Corpus (CACO) of about 19 million tokens from sources which we assumed are proofread. We have also used HaBiT for comparison.

The CACO is compiled from various sources that are published since the mid of twentieth century. It was collected from publicly available archives of three Amharic newspapers (አዲስ አድማስ, አዲስ ዘመን, and ሪፖርተር), two magazines (ንቁ and መጠበቂያ ግንብ), eight fictions (ኦሮማይ, የልምዣት, አልወለድም, ግርዶሽ, ልጅነት ተመልሶ አይመጣም, የአመጽ ኑዛዜ, የቅናት ዛር, and አግዐዚ), four historic novels (አሉላ አባነጋ, ማዕበል የአብዮቱ ማግሥት, የማይጨው ቁስለኛ, and የታንጉት ሚስጢር), two short novels (የዓለም መስታወት and የቡና ቤት ስዕሎችና ሌሎችም ወጎች), five history books (አጭር የኢትዮጲያ ታሪክ, ዳግማዊ አጤ ምኒልክ, ዳግማዊ ምኒልክ, የእቴጌ ጣይቱ ብጡል (፲፰፻፴፪ - ፲፱፻፲) አጭር የሕይወት ታሪክ, and ከወልወል እስከ ማይጨው), two politics books (ማርክሲዝምና የቋንቋ ችግሮች and መሬት የማን ነው), and two children books (ፒኖኪዮ and ውድድር). In addition, Amharic news articles and legal documents (ነጋሪት ጋዜጣ) from ELRA-W0074 (2014), news articles from Ethiopian News Agency, and the Amharic Bible[2] are used.

Paragraphs from the body of the documents are extracted. Then the paragraphs are transliterated to Latin-based characters using a modified version of the System for Ethiopic Representation in ASCII (SERA) (Yitna and Yacob, 1997). The modification is in transliterating labiovelars, which represent consonants followed by a back low diphthong ʷa, and vowels that are written independently. For example, the labiovelar ቧ /bʷa/ and the vowel ኡ /ʔu/ using the original SERA is transliterated as *bWa* and *'u* but with the modified version as *bua* and *u*, respectively. The same modification is adapted for ease of typing by the popular Amharic keyboard input methods such as Google's and Keyman's. Besides, four of Amharic phonemes have one or more homophonic character representations and there are other peculiar labiovelars (e.g., ቐ /kʰʷ/, ጒ /gʷi/, and ጔ /gʷe/). In the contemporary Amharic writings, the homophonic characters are commonly observed to be used interchangeably and there is no uniform use of the peculiar labiovelars. For consistent spelling, the Ethiopian Languages Academy (ELA) proposed a spelling reform (ELA, 1970; Aklilu, 2004). Following their reform, homophonic characters are merged into their common forms; ሐ and ኀ are replaced with ሀ, ሠ with ሰ, ዐ with አ, and ፀ with ጸ. The replacement includes their variant forms. This process can be considered as case folding in English (Yacob, 2003). We have normalized the peculiar labiovelars by substituting them with their closer counterparts (e.g., ቐ /kʰʷ/ withቈ /kʰu/). However, unlike, the spelling reform we preferred ጸ to ፀ; and kept ኧ and ዀ because in many input methods they are easily accessible and are commonly found in Amharic writings.

After the transliteration of the paragraphs, numbers are replaced by a placeholder (e.g., "በ1990 ዓ. ም." is preprocessed as "be # a m"); hyphenated words are split (e.g., "ስነ-ስርዓት" as "sne sr'at"); unique sentences are identified and extracted by their boundaries either double colon-like symbols (።) or question marks (? or ፧); and are tokenized based on orthographic-word boundaries, a white space or a colon-like symbol (፡).

[1] The International Phonetic Alphabets (Hayward and Hayward, 1992; IPA, 2015) are written only for the sake of readability.

[2] We used the New World Translation of the Bible which is translated into the contemporary (not archaic) Amharic.

To train the English language model, all sentences from British National Corpus (BNC) are extracted (BNC XML Edition, 2007). In order to equate the preprocessing steps of both languages, the sentences are preprocessed in a manner similar to Amharic corpora as follows: as case folding letters are lowercased, numbers are replaced by a placeholder # symbol, hyphenated words are split, contracted forms (clitics) are conflated (e.g., is n't into isn't), part-of-speech tags are discarded, and they are tokenized based on white space. Table 1 shows the number of sentences and tokens in each corpus.

Furthermore, through a manual check, we have analyzed that terms which appear only once in the respective corpora are mostly misspelled. Before we build the language model for each corpus, as a further preprocessing step, we have deleted sentences that contain words that occur only once in the entire corpus.

	CACO	HaBiT	BNC
No. of sentences	1,335,446	1,197,880	5,847,803
No. of tokens	18,933,305	17,605,866	97,111,951

Table 1: The number of sentences and tokens in the CACO, HaBiT, and BNC corpora.

The corpora statistics after the final preprocessing step is shown in Table 2.

	CACO	HaBiT	BNC
No. of sentences	1,010,590	873,426	5,690,343
No. of unigrams	366,654	350,789	228,999
No. of bigrams	5,811,598	4,986,029	11,008,294
No. of trigrams	9,996,057	8,302,152	40,168,232

Table 2: The corpora statistics after preprocessing.

The language models are trained using the KenLM language modeling toolkit (Heafield et al., 2013). The models are saved in the binary ARPA format for efficiency. The prior probability $P(w_1w_2\ldots\ w_{n-1}w_n)$ for trigram language model is estimated by Equation 2, based on chain rule of probability and Markov's assumption. The log probabilities that are used to compute this conditional probability along with backoff weights have been precomputed and are stored in the ARPA file language models.

$$\prod_{i=1}^{n} P(w_i|w_{i-2}w_{i-1}) \quad (2)$$

3.2 Error Model

Most Amharic characters are syllabary (Bloor, 1995; Unicode Consortium, 2017). For instance, በ /bə/, ቡ /bu/, ቢ /bi/, ባ /ba/, ቤ /be/, and ቦ /bo/ are all syllabic scripts with CV pattern. They conflate consonants and vowels even if they are typed with QWERTY keyboard input methods with direct mappings between keystrokes and characters. Hence there is a need to separate the two components to properly model spelling errors. This is done by transliteration of the letters into Latin alphabets by using the modified version of the SERA.

To train the error model, there is no sizable Amharic spelling error corpus. But we have made an assumption: as Amharic scripts are typed with English QWERTY keyboard, the key slips that cause spelling errors in English and Amharic are related. So, a substring based English spelling error model that represents the likelihood probability, $P(x|w_n)$, is useful for languages that can be transliterated into Latin alphabets. Such an error model is created by Norvig (2009) based on forty thousand spelling errors. Since this approach suits our need, we have adapted the error model.

3.3 Term Splitting

For spelling errors resulting from missed out spaces, term splitting is necessary. To generate candidate corrections for a spelling error, the expression was segmented to all possible valid words using a term list. Then using a language model a prior probability to each candidate was assigned. The candidate which has the highest probability is the plausible spelling correction. For example, Table 3 demonstrates how to split the above-mentioned example (i.e., ዮሃንስነገይመጣል transliterated with the modified SERA as *yohansnegeymeTal*), using the CACO language model and the corresponding term list. The probability of *yohans nege ymeTal* is the highest of all. Thus, the expression is split as such and transliterated back into Amharic as ዮሃንስ ነገ ይመጣል.

Candidates	Log 10 probability	Probability
yo hans nege ymeTal	-19.15934944	$6.92868 * 10^{-20}$
yoha ns nege ymeTal	-20.61217499	$2.44245 * 10^{-21}$
yohan s nege ymeTal	-19.17063332	$6.75098 * 10^{-20}$
yohans nege ymeTal	-11.64624405	$2.25817 * 10^{-12}$

Table 3: Example of a term splitting.

4. Evaluation

To evaluate the performance of our approach and to demonstrate its easy portability to other languages, first we made an evaluation based on Amharic test data and compared the results with the baseline systems: GNU Aspell and Hunspell; and then we performed an evaluation on English. For evaluation of spelling error detection capability, precision, recall, and F1 measure were used as metrics; and the relative positions of the correct spellings in the plausible suggestions list were used to evaluate spelling error correction. To interface with Aspell and Hunspell we used the PyEnchant[3] with their latest dictionaries available for both languages.

4.1 Test Data

We used manually annotated spelling error test corpora for evaluation. For Amharic we used a test corpus compiled by Gezmu et al. (2017)[4]; and for English the one that was compiled by Mitton (1985) from the book "English for the Rejected" (Holbrook, 1964) which is available in the Oxford Text Archive. Even though this one was originally handwritten by poor spellers, its contextual information makes it still useful for evaluation purposes.

For Amharic test data, 367 sentences were tagged with 287 non-word spelling errors, but 35 of the non-word misspellings appear twice in the documents with different

[3] Available at https://pypi.python.org/pypi/pyenchant/

[4] The annotated test corpus is available at the appendix of Gezmu et al. (2017).

contexts. For the sake of making a comparative evaluation with the baseline systems, 252 of the unique non-word misspellings were used in the evaluation. Removal of the duplicates is needed because the baseline systems do not use the context of the misspellings and two similar misspellings are identical test cases for them. In the English test data, 1,043 unique non-word errors were used, including one misspelling "o clock" which was not tagged by mistake in the original test corpus.

4.2 Evaluation Metrics

The evaluation metrics are from the perspectives of spelling error detection capability and the quality of plausible suggestions offered to each spelling error.

Spelling error detection capabilities are evaluated by precision, recall, and F1 measure, in the manner of the binary classification of terms as the misspelling and correct term classes. These evaluation metrics are calculated based on Equations 3–5 (Huyssteen et al., 2004); where True Negatives (TN) are correctly flagged misspellings, False Positives (FP) are unidentified misspellings, True Positives (TP) are correctly identified well-spelled words, and False Negatives (FN) are wrongly flagged well-spelled words. The desirable property for any spelling error detector would be to score 100% precision as it should flag all misspellings, and only misspellings; and also to score 100% recall as it should recognize all valid words as correct, and all invalid words as misspellings. Hence, recall is mostly an indication of the language coverage. F1 Measure gives an overall view of the capability of a spelling error detector. To compute the actual scores, we used the manually compiled test data as the *gold standard* for the evaluation.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

$$F1\ Measure = \frac{2 * (Precision * Recall)}{Precision + Recall} \quad (5)$$

The quality of suggestions offered by a spelling corrector is measured by the relative positions of the correct spellings in the suggestions list (Mitton, 2008). In the best scenario, the right correction always appears on the topmost of the list.

5. Results and Discussions

This section gives a detailed description of the results of our evaluation. The results are presented broadly in terms of spelling error detection and spelling error correction for Amharic and English.

5.1 Amharic Results

Figure 1 indicates the precision and recall graph for Amharic spelling error detection. The precision and recall scores are computed based on the different word lists compiled from the most frequent words in CACO and HaBiT corpora. The optimum results were obtained when a term list is composed of seven or more frequent words from the HaBiT and eight or more frequent words from the CACO corpus were used. The corresponding precision, recall, and F1 measures are given in Table 4. It also shows the scores for the baseline systems.

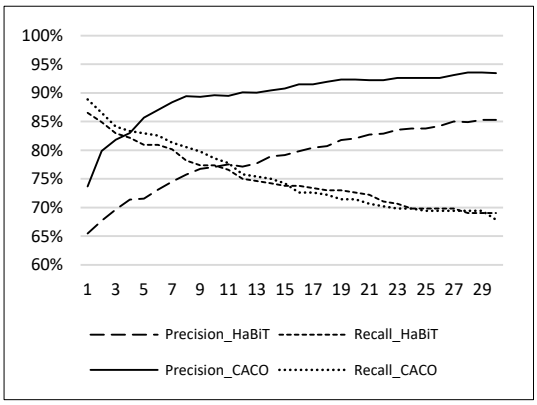


Figure 1: Precision and recall graph for spelling error detection in Amharic.

The evaluation results indicate that the F1 measure improves for the proposed system using the CACO corpus from 78% to 85% for spelling error detection compared to the baseline systems (see Table 4). However, we did not get any improvement when we used the term list compiled from the HaBiT corpus.

Metric	Proposed using CACO	Proposed using HaBiT	Aspell	Hunspell
Precision	89.4%	74.5%	79.4%	79.1%
Recall	80.6%	80.2%	76.6%	76.6%
F1	84.8%	77.2%	78.0%	77.8%

Table 4: Amharic spelling error detection results.

The measures of qualities of suggestions offered by the baseline and proposed systems for Amharic spelling error are shown in Table 5. According to the results, 77% of correct spellings appear in the top five suggestions list for the proposed system using CACO compared to 34% for Hunspell and 62% for Aspell. When we used the HaBiT corpus, 75% of correct spellings appear in the top five suggestions list, which is lower than that of the CACO corpus by 2%. Furthermore, when we consider the correct spellings that appear in the top first suggestions list, the proposed approach that uses the CACO corpus scored 9% higher than that uses the HaBiT corpus. This performance gain indicates that our approach is dependent on the underlying corpus used.

	Proposed using CACO	Proposed using HaBiT	Aspell	Hunspell
Top first	52.0%	42.9%	33.7%	16.7%
Top two	67.5%	61.9%	45.2%	26.6%
Top three	73.8%	69.4%	53.2%	29.0%
Top four	75.8%	73.8%	59.9%	33.7%
Top five	77.0%	75.4%	61.9%	34.1%

Table 5: Percentage of the topmost correct suggestions provided for Amharic spelling error correction.

5.2 English Results

The optimum F1 measure for English spelling error detection is obtained when we have used a term list that is compiled from fifty-seven or more frequent words from the BNC corpus. Its corresponding precision, recall, and F1

measures are given in Table 6 along with those of the baseline systems. The F1 measure for the proposed system is 96% and 97% for both baseline systems. The proposed system is lower than the baseline systems by 1%.

Metric	Proposed using BNC	Aspell	Hunspell
Precision	95.4%	99.2%	97.8%
Recall	96.1%	95.4%	95.3%
F1	95.7%	97.3%	96.6%

Table 6: English spelling error detection results.

The measures of qualities of suggestions offered by the baseline and proposed systems for English spelling error are shown in Table 7. With the proposed system, 74% of correct spellings appear in the top five suggestions list compared to 56% for Hunspell and 61% for Aspell.

	Proposed using BNC	Aspell	Hunspell
Top first	56.6%	27.4%	26.7%
Top two	66.0%	36.0%	38.8%
Top three	70.3%	50.1%	46.8%
Top four	72.3%	55.7%	52.7%
Top five	73.5%	60.5%	56.4%

Table 7: Percentage of the topmost correct suggestions offered for English spelling error correction.

6. Conclusion

We have proposed a method of an Amharic spelling corrector. Special characteristics of our approach are that it infers linguistic knowledge from text corpus and can be ported to other written languages with little effort as long as they are typed using a QWERTY keyboard with direct mappings between keystrokes and characters. The effort it requires is tokenization and transliteration to Latin characters. The proposed method was evaluated with the baseline systems. The evaluation results for Amharic and English test data confirm that our approach has a better performance than the baseline systems. This is mainly because of the application of a smoothed language model, generalized error model and the ability to take into account the context of misspellings. Since our approach is in a way to infer linguistics knowledge from a text corpus, the quality of the corpus that we have used has a direct effect on its performance. This is clearly shown with the performance differences between the two different Amharic corpora used.

A corpus-driven approach is related to lexicons used for spelling error detection. We can hardly find a manually compiled lexicon with reasonable coverage for a less-resourced language which has rich morphology like Amharic. Instead of using a handcrafted lexicon, using a term list derived from frequently occurring terms from a text corpus has advantages. Such a term list, in addition to ease of compilation, has also benefits in handling rare terms, proper nouns, technical terms, brand names, and newly coined terms (neologisms).

For future work, we will try to evaluate our approach for real-word spelling errors.

7. Acknowledgements

We would like to thank the anonymous reviewers for their invaluable comments. Our thanks also goes to Daniel Yacob at Ge'ez Frontier Foundation for making available some of the Amharic e-books that we have used for the CACO corpus.

8. Bibliographical References

Ahmed, F., Luca, E. W., and Nürnberger, A. (2009). Revised n-gram based automatic spelling correction tool to improve retrieval effectiveness. *Research Journal on Computer Science and Computer Engineering with Applications* (Polibits), 40: 39–48.

Aklilu, A. (2004). Sabean and Geez symbols as a guideline for Amharic spelling reform. In Proceedings of the First International Symposium on Ethiopian Philology, pages 18–26, Addis Ababa, Ethiopia, October. Addis Ababa University Press.

Blair, C. R. (1960). A program for correcting spelling errors. *Information and Control*, 3: 60–70.

Bloor, T. (1995). The Ethiopic writing system: A profile. *Journal of the Simplified Spelling Society*, 19(2): 30–36.

Brill, E., and Moore, R. C. (2000). An improved error model for noisy channel spelling correction. In Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, pages 286–293, October. Hong Kong, ACL.

Chen, S. F., and Goodman, J. (1998). An empirical study of smoothing techniques for language modeling. Tech. rep. TR-10-98, Computer Science Group, Harvard University.

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3): 171–176.

ELA. (1970). የአማርኛ ፡ ፊደል ፡ ሕግን ፡ አንዲጠብቅ ፡ ለማድረግ ፡ የተዘጋጀ ፡ ራዎር ፡ ማስታወሻ, *Journal of Ethiopian Studies*, 8(1): 119–134.

Fabri, R., Gasser, M., Habash, N., Kiraz, G., and Wintner, S. (2014). Linguistic Introduction: The orthography, morphology, and syntax of Semitic languages. In I. Zitouni (Ed.), *Natural Language Processing of Semitic Languages*, Berlin, Germany: Springer-Verlag, pp 3–41.

Gao, J., Li, X., Micol, D., Quirk, C., and Sun, X. (2010). A large scale ranker-based system for search query spelling correction. Proceedings of the 23rd International Conference on Computational Linguistics, pages 358–366, Beijing, August. ACL.

Gezmu, A. M., Seyoum, B. E., Lema, T. T., and Nürnberger, A. (2017). Manually annotated spelling error corpus for Amharic. Tech. rep. FIN-001-2017, Data and Knowledge Engineering Group, Otto-von-Guericke-Universität Magdeburg.

Hayward, K., & Hayward, R. J. (1992, June). Amharic. *Journal of the International Phonetic Association*, 22(1-2), 48-52.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 690–696, Sofia, Bulgaria, August. ACL.

Holbrook, D. (1964). English for the rejected: Training literacy in the lower streams of the secondary school. Cambridge, Cambridge University Press. Retrieved

January 8, 2016, from http://www.dcs.bbk.ac.uk/~roger/holbrook-tagged.dat

Huyssteen, V. G., Eiselen, E., and Puttkammer, M. (2004). Re-evaluating evaluation metrics for spelling checker evaluations. In Proceedings of First Workshop on International Proofing Tools and Language Technologies, pages 91–99, Patras, Greece. University of Patras.

IPA. (2015, June 1). *Full IPA Chart*. Retrieved August 12, 2017, from International Phonetic Association: https://www.internationalphoneticassociation.org/sites/default/files/IPA_Kiel_2015.pdf

Kernighan, M. D., Church, K. W., and Gale, W. A. (1990). A spelling correction program based on a noisy channel model. In Proceedings of the 13th conference on Computational linguistics - Volume 2, pages 205–210, Helsinki, Finland, August. ACL.

Kneser, R., and Ney, H. (1995). Improved backing-off for m-gram language modeling. In International Conference on Acoustics, Speech, and Signal Processing, pages 181–184, Detroit, Michigan, May. IEEE.

Kukich, K. (1992). Techniques for Automatically Correcting Words in Text. *ACM Computing Surveys*, 24(4): 377-439.

Mekonnen, A. (2012). Development of an Amharic spelling corrector for tolerant-retrieval. In Proceedings of the International Conference on Management of Emergent Digital EcoSystems, pages 22–26, Addis Ababa, Ethiopia, October. ACM.

Mitton, R. (1985). A collection of computer-readable corpora of English spelling errors, *Cognitive Neuropsychology*, 2(3): 275–279.

Mitton, R. (2008). Ordering the suggestions of a spellchecker without using context. *Natural Language Engineering*, 15(2): 173–192.

Mitton, R. (2010). Fifty years of spellchecking. *Writing Systems Research*, 2(1): 1–7.

Morris, R., and Cherry, L. L. (1975). Computer detection of typographical errors. *IEEE Transactions on Professional Communication*, PC-18(1): 54–64.

Norvig, P. (2009). Natural Language Corpus Data. In T. Segaran, and J. Hammerbacher (Eds.), *Beautiful Data: The Stories Behind Elegant Data Solutions*. Sebastopol, Canada: O'Reilly Media, Inc., pp 219–242.

Rychlý, P., and Suchomel, V. (2016). Annotated Amharic corpora. In International Conference on Text, Speech, and Dialogue, pages 295–302, Brno, Czech Republic, September. Springer International Publishing.

Scannell, K. P. (2007). The Crúbadán project: Corpus building for under-resourced languages. In C. Fairon, H. Naets, and A. Kilgarriff (Eds.), *Building and Exploring Web Corpora*. Louvain-la-Neuve, Belgium: UCL Presses, pp 5–15.

Tachbelie, M. Y., and Menzel, W. (2007). Sub-word based language modeling for Amharic. In N. Nicolov, G. Angelova and R. Mitkov (Eds.) *Recent Advances in Natural Language Processing V: Selected papers from RANLP 2007*. Amsterdam: John Benjamins Publishing Company, pp 301–310.

Toutanova, K., and Moore, R. C. (2002). Pronunciation modeling for improved spelling correction. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pages 144–151, Philadelphia, July. ACL.

Unicode Consortium. (2017). The Unicode® Standard: Version 10.0 – Core Specification. Mountain View, CA, USA: Unicode, Inc.

Whitelaw, C., Hutchinson, B., Chung, G. Y., and Ellis, G. (2009). Using the web for language independent spellchecking and autocorrection. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing - Volume 2, pages 890–899, Singapore, August. ACL and AFNLP.

Winkler, W. E. (2006). Overview of record linkage and current research directions. Tech. rep., Statistical Research Division, U.S. Census Bureau.

Yacob, D. (2003). Application of the Double Metaphone Algorithm to Amharic Orthography. In The XVth International Conference of Ethiopian Studies, pages 921–934, Hamburg, July. Otto Harrassowitz Verlag.

Yitna, F., and Yaqob, D. (1997). The System for Ethiopic Representation in ASCII. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.3191

9. Language Resource References

BNC XML Edition. (2007). The British National Corpus, version 3, distributed via Bodleian Libraries, University of Oxford, on behalf of the BNC Consortium. URL: http://www.natcorp.ox.ac.uk/

ELRA-W0074. (2014). Amharic-English bilingual corpus, distributed via ELRA, 1.0, ISLRN 590-255-335-719-0.

HaBiT. (2016). Harvesting big text data for under-resourced languages, distributed via Natural Language Processing Centre, Faculty of Informatics, Masaryk University. URL: http://habit-project.eu

A Speaking Atlas of the Regional Languages of France

Philippe Boula de Mareüil[1], Frédéric Vernier[1], Albert Rilliard[1,2]

[1]LIMSI, CNRS & Univ. Paris-Saclay, Orsay, France; [2] Univ. Federal de Rio de Janeiro, Brazil
{philippe.boula.de.mareuil, frederic.vernier, albert.rilliard}@limsi.fr

Abstract

The aim is to show and promote the linguistic diversity of France, through field recordings, a computer program (which allows us to visualise dialectal areas) and an orthographic transcription (which represents an object of research in itself). A website is presented (https://atlas.limsi.fr), displaying an interactive map of France from which Aesop's fable "The North Wind and the Sun" can be listened to and read in French and in 140 varieties of regional languages. There is thus both a scientific dimension and a heritage dimension in this work, insofar as a number of regional or minority languages are in a critical situation.

Keywords: geolinguistics, dialectology, speaking atlas

1. Introduction

Even if the modern Western world appears to be dominated by just a few widespread languages dialectologists in the field quickly observe a great deal of diversity. The idea of reporting this diversity on maps is not novel (Le Dû *et al*., 2005): from 1897 to 1901, E. Edmont traveled through France and its Gallo-Romance outskirts to carry out surveys in over 600 communes, mapped in the *Atlas Linguistique de la France* [ALF] (Gilliéron & Edmont, 1902–1910). Since the Second World War, the *Atlas linguistiques de la France par régions* (Séguy, 1973a; Tuaillon, 1976) have continued to document increasingly threatened languages and dialects — the limit between languages and dialects being ill-defined (Sibille, 2010).

Recently, audio recordings have been digitised, at least for the Occitan (or *Oc*, Southern Gallo-Romance) domain, as part of the *Thesaurus Occitan* [Thesoc] (Sauzet & Brun-Trigaud, 2013) and the Francoprovençal domain, with the *Atlas Multimédia de la région Rhône-Alpes* [ALMuRA] (Médélice, 2008), but they are not entirely available — and the *Oïl* (i.e., Northern Gallo-Romance) domain is even more under-resourced (Léonard & Djordjević, 2009). Following the principle of paper dialectological atlases and benefiting from computer technology, the *Corpus de la parole* now gives online access to an audio corpus of great wealth (Jacobson & Baude, 2011). However, comparable data are sorely lacking, and despite national projects such as SyMiLa (http://symila.univ-tlse2.fr/), concerning syntactic variation, research on dialectal variation in France is somewhat lagging behind what is done in the Netherlands and Norway (Heeringa, 2004), Germany and Italy (Iannàccaro & Dell'Aquila, 2001), for example. The dominant position of the French language masks a linguistic diversity among the most exemplary of Europe, which the European Charter for Regional or Minority Languages (signed by France in 1999 but not ratified) proposed to promote. Similarly, Unesco insists on the need for multilingualism in cyberspace (Vannin *et al*., 2012); and general public-oriented sites are multiplying, offering people to record themselves all over the globe and asking social networks to vote for this or that recording, but without linguistic control. It is therefore important, even urgent, to collect recordings using a coordinated approach, applying a common protocol to give a better picture of the plurality of uses.

A few decades ago, it would have been easier to find speakers of dialects and regional languages in France. However, the Internet, which now makes it possible to contact a number of associations for the protection and promotion of minority language, was less developed at that time, as were the means of collecting and storing large quantities of recordings. Since most of these dialects and languages are now endangered, we describe here a speaking linguistic atlas that aims to preserve them. This atlas takes the form of a website presenting an interactive map of France (for the moment limited to the metropolis, although recordings were also collected in the overseas), with over a hundred survey points on which one can click to hear as many speech samples and read a transcript of what is said. The orthographic transcription issue for the regional languages of France continues to be debated: particular attention was thus paid to this problem.

In the absence of well-established written traditions and a single authority, the regional languages of France do not have unanimously recognised and accepted standards (Caubet *et al*., 2002; Vannin *et al*., 2012). Consequently, the transcription solutions proposed vary from one language to another, even among the territorial languages of France, the spellings of which are based on the Latin alphabet and are meant to be close to the pronunciation (at least to a minimum). The orthographies adopted are more or less phonetic (reflecting a particular local pronunciation) or diasystemic (emphasising the unity of a set of dialects). Sometimes, the system is hybrid, for the sake of efficiency, noting what in pronunciation differs from French, while following the orthographic conventions of French. These systems have their advantages and disadvantages.

These different types of writing were tested through a case study: the transcription of the same text of a hundred words, which we will begin by presenting. We will then describe the surveys we conducted, the speakers whose recordings were collected and transcribed, the protocol we applied, and the mapping we established. We will present quantitative data at the conference.

2. Survey Points and Speakers

The material we have focused on consists of Aesop's fable "The North Wind and the Sun" (120 words in French, about 1 mn of speech), translated and recorded in more than 140 investigation points in as many varieties of regional languages of France. This text (used by the International Phonetic Association to describe a large number of languages and dialects of the world) was a way of reviving a 19th century tradition consisting of

translating the parable of the prodigal son. The corpus collected resulted in a digital database covering Romance languages (such as Occitan, Francoprovençal, Catalan and Corsican), Germanic languages (Alsatian, Franconian and West Flemish), Breton and Basque.

In order to meet these speakers, we contacted universities and local radio stations (if any), associations and public offices promoting regional languages and clubs of *patois* speakers, city halls, tourist offices and immersive schools (*diwan, calandretas*). We sent hundreds of e-mails and made hundreds of phone calls. A considerable amount of time was spent finding speakers of regional languages who are now a small minority, organising and optimising travel. This research required extensive documentation not only on dialects but also on the history and geography of the regions explored. In 2014–2016, we conducted about thirty field surveys, with a few days per mission, seeking to maximise the number of speakers per locality.

We then selected one speaker per locality, retaining only one commune within a radius of less than 20 km. We favoured native speakers, but have not disregarded neo-speakers, in cities like Bordeaux or Nice. In some cases, there was little choice; in others, one of the speakers stood out for his/her oratory talent, the aesthetic nature of his/her voice or his/her local anchorage. The distinction between "neo-speakers" and "natural speakers" is not always obvious: there are native speakers of neo-Breton, for example; so-called "natural" speakers may have a highly Frenchified regional language; on the contrary, some speakers who have not inherited the dialect directly from their parents can commit to successfully replicating local particularities. When a speaker was born and has always lived in the same place, associating his/her recording with a point on the map is easy, but modern life encourages mobility. When a speaker explicitly said (s)he spoke the dialect of a place that was not his/her place of residence, we kept this place to associate it to him/her — usually the commune where the speaker had grown up.

The speakers, rather engaged in the cultural and linguistic field, were from varied socio-professional backgrounds. A typical profile was the following: a retired man from the farmer world (or workers' world, in the North and the East), having experienced social climbing, having worked for instance as a teacher. It should be noted that in the ALF, too, E. Edmont investigated mainly educated men, on average older in the North than in the South. All those who responded positively gave us an extremely warm welcome, expressing the moving feeling that their *patois* was going to disappear with them, in the *Oïl* domain and the Croissant (intermediate zone between *Oïl* and *Oc* domains, so named because of its crescent shape). Without necessarily being "local scholars", they generally showed a very thorough knowledge of the culture, history and geography of their region, making the exchanges very enriching. Most of them were recorded at home or in a meeting room with a Zoom H4 recorder in Wave format (in stereo at a sampling frequency of 44.1 kHz and 16-bit quantisation, before intensity equalisation and conversion to MP3 for uploading). Each recording was associated with a signed consent for free distribution.

3. Protocol and Transcription

A common protocol was applied, in which the speakers were first asked to read the fable in French. They were then asked to translate this text into their regional language, either directly with the French text in front of them, or from notes they had preferred to take. Some speakers (especially in the Basque Country, Brittany, West Flanders and the *Oïl* domain) wanted to write their translations entirely. This situation occurred either because the regional language and French were very different in their structures (with a different word order, as in Breton), or because, on the contrary, the regional language was close to French, as in the *Oïl* domain, and the speakers needed "to get back into the swing of things". French, in France, dominates so much in everyday interactions and public space that the translation task is not easy. In Corsican, Catalan, Occitan, Francoprovençal, Alsatian and sometimes in the *Oïl* domain, the speakers we recorded succeeded very well in translating the Aesop fable on the fly. A word like *voyageur* 'traveler' could pose difficulties, because at the time *patois* was spoken, some informants said, people did not use to travel). Still, translating the fables of La Fontaine or others was an exercise to which many of our informants were accustomed. Sometimes the speakers moved away from a literal translation to get closer to their oral traditions. These different translation strategies are also a testimony of the wealth and diversity of linguistic ingenuity.

For Occitan (*langue d'oc*), we transcribed the recordings into classical spelling (Alibert, 1935), sometimes drawing on what the speakers had provided us. We had the transcriptions checked and corrected by specialists who had access to the audio, as was the case for the Catalan and Corsican transcriptions we had produced.

The unified spellings of Basque (*batua*) and Breton (*peurunvan*) were well known by our Basque and Breton speakers, respectively. For Alsatian, a flexible system close to ORTHAL (Zeidler & Crevenat-Werner, 2008) was used, with written German (*Schriftsprache*) as a reference, respecting dialectal peculiarities. For Luxembourgish Franconian (spoken in our Apach survey point), the Luxembourg standard was followed, whereas for Moselle and Rhine dialects other systems were adopted, also inspired by modern German, but using the 'æ' digraph and grave accents to indicate/overcharacterise the open quality of some vowels, as well as consonants such as 'd' to note dull non-aspirated stop consonants, like [t] (Hudlett, 2004).

In the *Oïl* domain, we sometimes face individual spellings, such as in Berrichon (among central dialects), Lorrain or Walloon, for which in France there is no real standard. This also applies to three survey points in the linguistic Croissant (Pleuville, Bussière-Poitevine and Éguzon-Chantôme), for which a French-like spelling was adopted. The writer thus encodes words as he wants them to be heard, respecting his own phonetics. More often than not, as proposed by Contejean (1876), it is a Frenchified spelling that is applied by our *Oïl* informants, with the following rules:

- preservation of the plural mark for nouns, pronouns and adjectives (in general *s*);
- preservation of the person mark for conjugation;

- a concern for clarity, avoiding accumulating elisions through apostrophes (markers of an orality considered inherent in *patois*).

The Feller-Carton system, for Picard (Carton, 2009), also proposes two principles, with subregion-dependent rules to solve cases where these two principles are in contradiction: (1) priority to the French spelling provided that it does not create ambiguity; (2) priority to the phonetics of Picard.

We will see how or if speakers resolved possible contradictions. In all cases, we made sure of the adequacy between what was said and what was transcribed, because it is frequent not to write exactly the words that were spoken and not to read exactly the words that were handwritten or printed. When necessary, we deleted disfluencies in the audio signal, as well as comments like *es pas aisit* ('it is not easy' in Oc-citan) which were not part of the fable.

4. Cartography

The survey points, defined by their latitude and longitude, were arranged on a map of France where we included boundaries between linguistic domains, in addition to the borders between departments which delimit administrative regions. The former are eminently more debatable and generally less abrupt (Paris, 1888; Gilliéron & Mongin, 1905). If one looks at isoglosses drawn according to different criteria, they may not coincide, and it is then difficult to determine which ones are preferable to use. It is also known that the intercomprehension criterion, sometimes put forward in folk linguistics, quickly finds its limits: it can be asymmetrical and may depend on the communicative situation. Since any classification is questionable (Goebl, 2002; Gaillard-Corvaglia *et al*., 2007, Sumien, 2012), the one we propose has no pretension to be definitive. It retains 25 regional languages or primary dialects, grouped as shown in Fig. 1.

We kept a classical partition in Romance languages (*Oïl*, *Oc*, Catalan, Francoprovençal and Corsican), Germanic languages (Alsatian, West Flemish, Franconian) and "Other languages" (Basque and Breton). The latter languages, on the French territory, are traditionally subdivided into dialects: Luxembourgish, Moselle and Rhine for Franconian, Labourdin, Lower-Navarrese and Souletin for Basque; Trégorrois, Léonard, Cornouaillais and Vannetais for Breton. Although each of these ten dialects is represented by a survey point, we did not include these labels on the map for scale reasons.

It turns out that the majority of associations for the defence and promotion of *Oïl* and *Oc* languages insist on the plural in the first case and on the singular in the second case, even if Gascon is unquestionably quite distinct from the other *Oc* varieties (Chambon, & Greub, 2002). These two points of view being difficult to reconcile (Abalain, 2007), out of respect for our informants, we indicated "langue(s) d'oïl" and "langue d'oc (occitan)" in the legend. Warm colours (in the reds) were chosen for *Oc* varieties, cool colours (in the blues) for *Oïl* varieties, while green was retained for Francoprovençal and shades of yellow were kept for Germanic varieties. Thicker lines emphasise the frontiers between these different domains, while the limits of the Croissant (attached to Occitan), appear blurred to suggest the transitory nature of this linguistic area.

In order to draw these linguistic limits, we classified the 600+ ALF suvey points in our 25 categories, based on various regional atlases and publications: for the *Oïl* domain (Bourcelot, 1966–1978; Brasseur, 1980–2011; Carton & Lebègue, 1989–1997; Dondaine, 1972–1978; Dubuisson, 1971–1982; Guillaume & Chauveau, 1975–1983; Lanher *et al*., 1979–1988; Massignon & Horiot, 1971–1983; Simoni-Aurembou, 1973–1978; Taverdet, 1975–1980; Martin, 2015); for the Francoprovençal domain (Gardette, 1952; Tuaillon & Contini, 1996); for the *Oc* domain (Séguy, 1973a, 1973b ; Desrozier & Ros, 1974; Sumien, 2008). The Occitan domain was subdivided, following Tuaillon and Contini (1996) into: Gascon, Languedocien, Provençal, North-Occitan and Croissant. We could have subdivided the North-Occitan subdomain (and the Croissant) into Limousin, Auvergnat and Vivaro-Alpine. However, the more we multiply borders, the more we create problems: speakers from Velay, for example, are difficult to classify. Featuring the Croissant, on the other hand, shows the shape of this set of transitional dialects between *Oïl* and *Oc*. As for the domain of central dialects of *Oil*, we have not sought to subdivide it into Tourraine, Orléanais, etc., for lack of speakers of these dialects.

An algorithm was then designed to draw lines passing in the middle of two points of different categories, to smooth out the contours and to colour the zones so defined. The smoothing of the boundaries between dialectal areas, more or less sawtooth-like, can be set by the user, using a slider. To refine the layouts, a few points were added near the linguistic borders of the Basque, Breton, Catalan, Alsatian, Flemish and Franconian domains, as well as at the confluences of the Gascon, Languedocian and North-Occitan areas. In addition, a "Search" field is offered to the user, and options are provided for displaying or not boundaries between departments, ALF points and our survey points, Jersey, French-speaking Switzerland and Belgium on the one hand, Overseas France on the other.

The resulting interactive map, with over 140 points that can be clicked on to hear as many versions of the same text, Aesop's fable "The North Wind and the Sun", is available at https://atlas.limsi.fr (in French and English). Its French translation can be listened to (and read) by clicking on Paris, at that address.

The site makes it possible to immediately appreciate how certain geographically close dialects differ linguistically. This is the case for survey points in Bresse, between Burgundy and Savoy (Francoprovençal), points in Romance (Welche) and German-speaking Alsace, and a fortiori points in the Basque Country and Béarn (Gascon), gathered in the department of Pyrénées-Atlantiques. It is striking (although not surprising) that no dialect speaker was found around Paris: for central dialects, we only found speakers of Berrichon-Bourbonnais. In the East,

especially, it is as if dialects had been relegated to Belgian, German and Swiss borders. We found speakers of Champenois and Lorrain almost only in the Ardennes and the Vosges, respectively. The annexation of Moselle by Germany during World War II may have given some respite to dialects (including Romance dialects, which were not prohibited as French was), but the speaker we recorded in that department is today an exception.

5. Conclusion and Future Work

This multimedia atlas provides a kind of colour photograph of the dialectal rainbow of (hexagonal) France: an instantaneous photograph of current uses, a tool for teaching variation, showing that the territorial languages of France, though threatened, are still today a precious reality (D'Hervé, 2005). It does not tell us much about the vitality of these languages, the number of their speakers, the social dynamics that are at work, despite the amount of data collected. Nor does it tell us how rewarding the personal meetings with the informants were. The practice of regional languages has become scarce and is now a minority: the chain of intergenerational transmission within family cells is often interrupted. It was therefore a challenge to give greater visibility to the diversity of the French linguistic landscape.

The problem of speaker representativeness remains a real issue, with respect to a traditional dialectological framework that assumes an irremovable link between the land and the speaker. This framework is more and more outdated in a country dominated by mobility and urbanisation (Mufwene & Vigouroux, 2012). Large cities such as Marseilles, Toulouse or Montpellier necessarily melt many differences, and the influence of French on regional languages is now inevitable. The applied protocol (a translation) can also have an influence, prompting us to take a critical look at the productions so elicited. However, depending on the speakers, the translation can favour both calques and a search for maximum deviation from French.

We wanted to reach linguists and teachers, but also the general public. This objective was achieved in view of the success of our site (over 300,000 visits by the summer of 2017), printed and audiovisual media as well as social networks. The aim was to catch up on the delay taken by France in the valorisation of its minority languages. Associating them to modernity (with innovative cartographic tools to present boundaries between dialectal areas), will hopefully help revitalise our linguistic heritage, or at least give it recognition as a vehicle for creativity, for lack of being able to counteract the decline in minority language practices.

We will continue to give media coverage to this speaking atlas, which we are going to extend to the French Overseas Territories. To go further, it is possible not only to use the web to display research results, but also to collect new information, using a crowdsourcing methodology now very much in fashion. Yet, proposing to complete surveys through the Internet will require some precautions for minority languages.

6. Acknowledgements

This work was largely financed within the framework of the DGLFLF "Langues et Numérique 2016" call for proposals. We thank F. Manni and M. Russo for their cooperation in this project, as well as G. Brun-Trigaud, who provided the latitudes and longitudes of more than 600 ALF points, and K. Biers for comments on an earlier version of this paper. We would like to express our sincere gratitude to all those who have agreed to give us their time and lend their voice to this achievement.

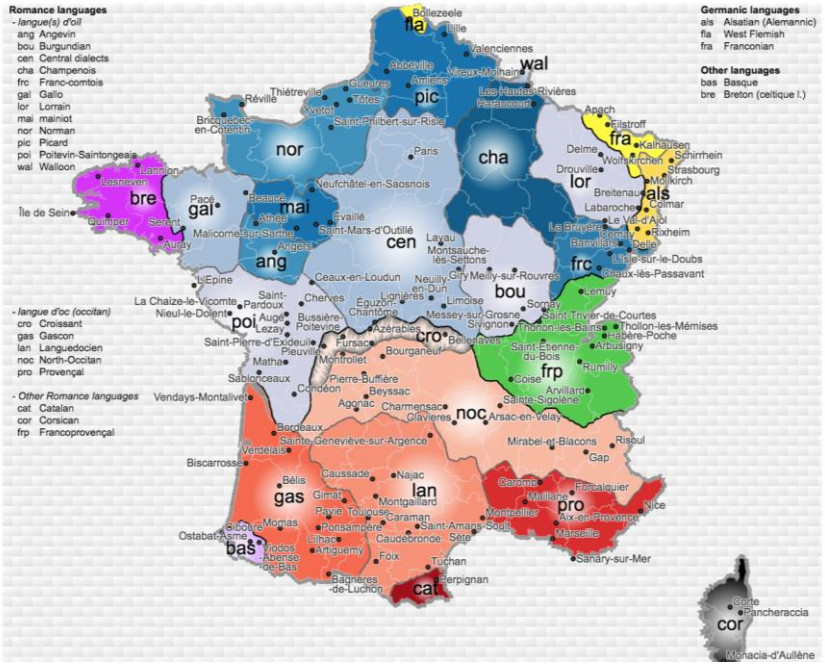


Figure 1: screenshot of the site < https://atlas.limsi.fr/index-en.html>.

7. Bibliographical References

Abalain, H. (2007). *Le français et les langues historiques de la France*. Éditions Jean-Paul Gisserot, Paris.Alibert, L. (1935)., *Gramatica occitana segon los parlars lengadocians*. Societat d'Estudis Occitans, Toulouse.

Bourcelot, H. (1966–1978). *Atlas linguistique et ethnographique de la Champagne et de la Brie*. Éditions du CNRS, Paris.

Brasseur, P. (1980–2011). *Atlas linguistique et ethnographique normand*, Éditions du CNRS, Paris.

Carton, F. (2009). Pourquoi et pour qui on transcrit ? Les graphies du picard moderne. *La Linguistique*, 45(1):113–123.

Carton, F. & Lebègue, M. (1989–1997). *Atlas linguistique et ethnographique de la Picardie*. Éditions du CNRS, Paris.

Caubet, D., Chaker, S. & Sibille, J. (2001). *Codification des langues de France*, L'Harmattan, Paris.

Chambon, J.-P. & Greub, Y. (2002). Note sur l'âge du (proto)gascon. *Revue de linguistique romane*, 66:263–264.

Contejean, C. (1876). *Glossaire du patois de Montbéliard*. Société d'émulation de Montbéliard, Montbéliard.

D'Hervé, G. (2005). Le gallo dans l'enseignement, l'enseignement du gallo. *Marges Linguistiques*, 10:262–281.

Desrozier, P. & Ros, J. (1974)). *L'ortografia occitana. Lo lemosin*, Centre d'Estudis Occitans, Montpellier.

Dondaine, C. (1972–1978). *Atlas linguistique et ethnographique de la Franche-Comté*. Éditions du CNRS, Paris.

Dubuisson, P. (1971–1982). *Atlas linguistique et ethnographique du Centre*. Éditions du CNRS, Paris.

Gaillard-Corvaglia, A., Léonard, J.-L., Darlu, P. (2007). Testing Cladistics on Dialect Networks and Phyla (Gallo-Romance Vowels, Southern Italo-Romance Diasystems and Mayan Languages. *Proc. 9th Meeting of the ACL SIG in Computational Morphology and Phonology*, Prague (pp. 23–30).

Gardette, P. (1952). Atlas linguistique et ethnographique du Lyonnais. *Revue de géographie de Lyon*, 27(1): 90–92.

Gilliéron, J. & Edmont, E. (1902–1910). *Atlas linguistique de la France*. Champion, Paris.

Gilliéron, J. & Mongin, J. (1905). *Étude de géographie linguistique :* « s*cier » dans la Gaule romane du Sud et de l'Est*. H. Champion, Paris.

Goebl, H. (2002). Analyse dialectométrique des structures de profondeur de l'ALF. *Revue de linguistique romane* 66(261–262): 5–63.

Guillaume, G. & Chauveau, J.-P. (1975–1983). *Atlas linguistique et ethnographique de la Bretagne Romane, de l'Anjou et du Maine*. Éditions du CNRS, Paris.

Heeringa, W. (2004). *Measuring dialect pronunciation differences using Levenstein distance*, PhD thesis, Rijksuniversiteit, Groningen.

Hudlett, A. (2004). *Charte de la graphie harmonisée des parlers franciques – platt – de la Moselle germanophone*. Publications du Centre de Recherche sur l'Europe Littéraire, Mulhouse.

Iannàccaro, G. & Dell'Aquila, V. (2001). Mapping languages from inside: notes on perceptual dialectlogy. *Social & Cultural Geography*, 2(3): 265–280.

Jacobson, M. & Baude, O. (2011). *Corpus de la parole* : collecte, catalogage, conservation et diffusion des ressources orales sur le français et les langues de France. *Traitement automatique des langues*, 52(3):47–69.

Lanher, J., Litaize, A., Richard, J. (1979–1988,). *Atlas linguistique et ethnographique de la Lorraine romane*. Editions du CNRS, Paris.

Le Dû, J., Le Berre, Y., Brun-Trigaud, G. (2005). *Lectures de l'atlas linguistique de la France de Gilliéron et Edmont. Du temps dans l'espace*. CTHS, Paris.

Martin, F. (2015). *Espaces & lieux de la langue au XXIe siècle en Picardie : approche complexe de la structuration des pratiques linguistiques en situations ordinaires – enquêtes en Picardie*. PhD thesis, Université de Picardie Jules Verne, Amiens.

Massignon, G. & Horiot, B. (1971–1983). *Atlas linguistique et ethnographique de l'Ouest*. Editions du CNRS, Paris.

Médélice, J. É. (2008). Présentation du projet de l'Atlas Linguistique Multimédia de la Région Rhône-Alpes et des zones limitrophes (ALMURA) et commentaires du poster. In G. Raimondi & L. Revelli, *Dove va la dialetto-logia?*, Edizioni dell'Orso, Alessandria (pp. 199–205).

Mufwene, S.S. & Vigouroux, C.B. (2012). Individuals, Populations, and Timespace: Perspectives on the Ecology of Language. *Cahiers de sociolinguistique*, 38(2): 111–137.

Paris, G. (1888). Les parlers de France. *Revue des patois gallo-romans*, 17:161–175.

Séguy, J. (1973a). Les Atlas linguistiques de la France par régions. *Langue française*, 18: 65–90.

Séguy, J. (1973b). La dialectométrie dans l'Atlas linguistique de la Gascogne. *Revue de linguistique romane*, 37:1–24.

Sibille, J. (2010). Les langues autochtones de France métropolitaine. Pratiques et savoirs. In C. Gruaz, C. Jacquet-Pfau (Eds.), *Autour du mot : pratiques et compétences*, Lambert-Lucas (pp. 69–85).

Simoni-Aurembou, M.-R. (1973–1978). *Atlas linguistique et ethnographique de l'Île-de-France et de l'Orléanais*, Éditions du CNRS, Paris.

Sumien, D. (2008). Classificacion dei dialèctes occitans. *Linguistica Occitana*, 7 : 1–55.

Sumien, D. (2012). Le catalogue des langues romanes : clarifier les critères et les idéologies. *Revue des langues romanes*, 116(1):5–33.

Taverdet, G. (1975–1980). *Atlas linguistique et ethnographique de la Bourgogne*. Éditions du CNRS, Paris.

Tuaillon, G. (1976). *Comportement de recherche en dialectologie française*. Éditions du CNRS, Paris.

Tuaillon, G. & Contini, M. (1996). *Atlas Linguistique Roman (ALiR)*. Istituto poligrafico e zecca dello stato, Rome.

Vannin, L. & Le Crosnier, H. (2012). *Net.lang. Réussir le cyberespace numérique*. C1F Éditions, Caen.

Zeidler, E. & Crévenat-Werner, D. (2008). *Orthographe alsacienne : bien écrire l'alsacien de Wissembourg à ferrette*. Jjérôme Do Bentzinger Colmar.

Towards Language Technology for Mi'kmaq

Anant Maheshwari,[1] **Léo Bouscarrat,**[2] **Paul Cook**[3]
1. Computer and Information Science, University of Pennsylvania
2. École des Mines de Saint-Étienne
3. Faculty of Computer Science, University of New Brunswick
anantm95@seas.upenn.edu, leo.bouscarrat@etu.emse.fr, paul.cook@unb.ca

Abstract

Mi'kmaq is a polysynthetic Indigenous language spoken primarily in Eastern Canada, on which no prior computational work has focused. In this paper we first construct and analyze a web corpus of Mi'kmaq. We then evaluate several approaches to language modelling for Mi'kmaq, including character-level models that are particularly well-suited to morphologically-rich languages. Preservation of Indigenous languages is particularly important in the current Canadian context; we argue that natural language processing could aid such efforts.

Keywords: Indigenous languages, web corpora, language modelling

1. The Need for Mi'kmaq Language Technology

For over one hundred years the Indian residential school system in Canada forcibly removed Indigenous children from their families, subjected them to physical and sexual abuse, and attempted to erase their Indigenous identities, in part by prohibiting the use of Indigenous languages (Truth and Reconciliation Canada, 2015). Thousands of children died while in this system. The last of these schools did not close until 1996. The Indian residential school system has been referred to as a form of cultural genocide.

On 11 June 2008 then-Prime Minister of Canada Stephen Harper issued an official apology to the survivors of Indian residential schools.[1] The Truth and Reconciliation Commission of Canada was formed to document events surrounding these schools, and to identify ways to improve conditions for Indigenous peoples. In December 2015 the Truth and Reconciliation Commission of Canada released ninety-four calls to action to facilitate the process of reconciliation (Truth and Reconciliation Canada, 2015). Amongst these is the principle that "Aboriginal languages are a fundamental and valued element of Canadian culture and society, and there is an urgency to preserve them."

Natural language processing (NLP) could help to play a role in the preservation of Indigenous languages by building language technology tools — such as spelling checkers, next word prediction systems, and machine translation systems — to aid in using Indigenous languages in computer-mediated communication.

Mi'kmaq is an Eastern Algonquian language, spoken primarily in Eastern Canada. It is a polysynthetic, free word-order language (Johnson, 1996). Rand (1888) demonstrates the morphological richness of Mi'kmaq with the word *yăle-oole-mâktāwe-pɔkŏse*, meaning "I am walking about, carrying a beautiful black umbrella over my head.". Although it has a rich oral tradition, various Roman scripts have been introduced for Mi'kmaq (Battiste, 1985), differing primarily in their representations of vowel length. Roughly 8,000 people reported Mi'kmaq as their mother tongue in the Canada 2011 Census.[2] Mi'kmaq is the most widely spoken Indigenous language in the province of New Brunswick.

Although there exist Mi'kmaq dictionaries (Rand, 1888; DeBlois, 1996) and translated texts (DeBlois, 1990), Mi'kmaq remains a low-resource language. In particular, no (large) machine-readable corpora are available for Mi'kmaq.

There has been very little work in computational linguistics or NLP on Mi'kmaq. The Crúbadán project (Scannell, 2007)[3] built web corpora for over 400 writing systems (as proxies for languages) of which Mi'kmaq was one.[4] These corpora are not publicly available. Brown (2014) studied language identification for over 1300 languages, with Mi'kmaq being included amongst these.

To the best of our knowledge, this is the first computational work to focus specifically on Mi'kmaq.[5] The long-term goals of this research are to 1.) create a large Mi'kmaq corpus to support corpus linguistic studies of, lexicographical analysis of, and training NLP systems for, Mi'kmaq; and 2.) to build a suite of NLP tools for Mi'kmaq, which could potentially contribute to language preservation. In this preliminary work, in Section 2 we first build a web corpus of Mi'kmaq, using methods similar to Scannell (2007), and analyze it. Language models are an important component of systems for many NLP tasks, including spelling correction and machine translation. In Section 3 we evaluate several approaches to language modelling for Mi'kmaq, including character-level approaches that could capture the morphological-complexity of Mi'kmaq. We conclude in Section 4 by discussing directions for future work.

[1] https://www.aadnc-aandc.gc.ca/eng/1100100015644/1100100015649

[2] http://www12.statcan.gc.ca/census-recensement/2011/as-sa/98-314-x/98-314-x2011003_3-eng.cfm

[3] http://crubadan.org/

[4] The number of writing systems included in this project has since climbed to over 2000.

[5] The Truth and Reconciliation Commission of Canada's calls to action also state that "The preservation, revitalization, and strengthening of Aboriginal languages and cultures are best managed by Aboriginal people and communities." None of the authors of this work are Aboriginal people. We have consulted with the Mi'kmaq-Wolastoqey Centre at the University of New Brunswick in carrying out this research.

2. A Mi'kmaq Web Corpus

Baroni and Bernardini (2004) describe an approach to automatically creating topically-focused corpora from the web, whereby tuples are randomly formed from a user-created keyword list for the topic of interest, and these tuples are then sent as queries to a search engine. The top-n results for these queries are downloaded, and then post-processed to remove mark-up and boilerplate content, documents in unwanted languages, and duplicate documents, with the final result being a topically-focused corpus. This approach to corpus construction remains widely-used, and is incorporated into commercial lexicographical tools (Baroni et al., 2006).[6]

The method of Scannell (2007) for creating corpora for specific languages begins with an approach quite similar to that of Baroni and Bernardini (2004), except that the search engine queries consist of a high frequency word in the language of interest combined with other words in that language. The words used in the queries are further controlled to avoid words that happen to also occur in another language (e.g., *die* is both an English and German word). The search engine results include many documents in the intended language.[7]

2.1. Corpus Construction

In this section we describe our approach to creating a Mi'kmaq web corpus, which uses an approach similar to that of Scannell (2007).

Seed Word Selection The Universal Declaration of Human Rights (UDHR) is available in over five-hundred languages, including Mi'kmaq.[8] We selected as seed words those words that occur in the Mi'kmaq translation of the UDHR, available through NLTK (Bird et al., 2009).[9]

Although the full text corpora created by Scannell (2007) are not publicly-available, various summaries of the corpora are provided, including a list of the word types, and their corresponding frequencies, in each corpus. In preliminary experiments we considered basing our seed words on the word frequency list provided for Mi'kmaq; however, we found the resulting corpora to be much smaller than when using seed words derived directly from the UDHR (which Scannell (2007) also considered as a source for seed words). We therefore only consider the seed words from the UDHR in the remainder of this paper.

Query Generation We considered two approaches to forming queries from the list of seed words, which were then sent to a commercial search engine.

Random We used the BootCaT tools (Baroni and Bernardini, 2004)[10] to randomly select 3-tuples from the seed word list, and then used these as queries.

Crúbadán Our second approach to query generation is based on that of Scannell (2007). We divide the seed words into high and low frequency terms, based on a frequency cut-off of five in the UDHR. We then randomly select two low-frequency words and one high-frequency one, and form a query of the form "(low1 OR low2) AND high", where low1 and low2 are low-frequency words and high is a high-frequency word.

In Section 2 2 we compare these two approaches to query generation, with varying numbers of queries.

Search Engine Queries We used the BootCaT tools to issue the queries generated above to the Bing Web Search API, and retrieve the top-10 result URLs for each query. We further used the BootCaT tools to remove duplicate URLs from the results.

Downloading Content Ferraresi et al. (2008) note that very small documents tend to contain little material appropriate for inclusion in corpora (due to the overhead of HTML markup), whereas very large documents tend to correspond to lists or catalogs, and not more-standard text. Following Ferraresi et al. (2008), we only download documents with size 5–200KB, and MIME-type text/html.

Markup and Boilerplate Removal We remove HTML markup and boilerplate text — e.g., navigation bars, headers, and footers — from the downloaded documents using jusText (Pomikálek, 2011), which preserves paragraph structure present in the HTML in the extracted text. JusText is able to incorporate information from a language-dependent stopword list in determining which document portions are boilerplate and which are more-conventional text. However, in preliminary experiments we observed that using a stopword list derived from the UDHR resulted in many portions of Mi'kmaq text not being recognized as such, and therefore chose to disable this feature.[11]

Language identification Although many language identification tools are available (Cavnar and Trenkle, 1994; Lui and Baldwin, 2012, for example), very little language identification research has considered Mi'kmaq, with Scannell (2007) and Brown (2014) being notable exceptions, and even these have not focused specifically on this language. Here we implement a simple approach to language identification. We represent each language for which the UDHR is available in NLTK as a vector of the character trigram frequencies in the corresponding version of the UDHR. For each document in our corpus, we similarly represent it as a vector of character trigram frequencies, and then compute its cosine similarity with the vector representing each language. We discard any document for which the most-similar language is not Mi'kmaq.

Deduplication The web contains many duplicate and near-duplicate documents. We perform (near and exact)

[6] https://www.sketchengine.co.uk/

[7] Scannell (2007) also considers restricted web crawls, using the result URLs as seeds for the crawler, to find additional documents; in this preliminary work we do not consider crawling.

[8] http://www.un.org/en/universal-declaration-human-rights/

[9] http://www.nltk.org/

[10] http://bootcat.dipintra.it/

[11] BTE (Finn et al., 2001) is an alternative boilerplate extraction tool that does not make use of a stopword list, and has been used in many corpus construction efforts (Baroni and Bernardini, 2004; Sharoff, 2006; Ferraresi et al., 2008, for example); however, unlike jusText, BTE does not preserve paragraph structure in the extracted text, and we rely on this structure in subsequent processing.

Queries	Random		Crúbadán	
	# Docs	# Tokens	# Docs	# Tokens
100	36	33k	62	35k
500	88	62k	138	79k
1000	120	92k	167	90k

Table 1: The number of documents and tokens in corpora constructed using the random and Crúbadán approaches to query generation, for increasing numbers of queries.

Corpus	# Docs	# Tokens	# Types
This paper	69	76k	24k
Crúbadán (Mi'kmaq)	31	100k	12k

Table 2: The number of documents, tokens, and types in the corpus created in this paper, and the Mi'kmaq corpus of Crúbadán (Scannell, 2007).

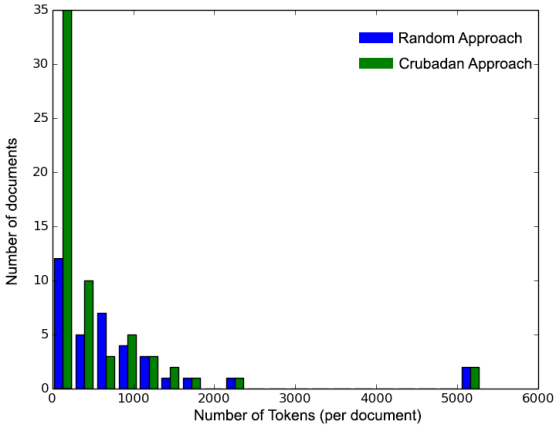


Figure 1: The number of documents, binned by number of tokens per document, for the random and Crúbadán approaches to query generation, using 100 queries.

deduplication at the sub-document level based on paragraphs (as provided by jusText), using onion (Pomikálek, 2011), under its default configuration. In this setup, onion iterates through the corpus once, and removes any paragraph for which more than 50% of its 7-grams have been seen in the corpus up to that point.

2.2. Corpus Analysis

We now analyze corpora constructed using the methodology described in the previous subsection, and compare the random and Crúbadán query generation strategies. For this analysis we applied simple whitespace-based tokenization. We further used a preliminary approach to language identification — based on the same methodology as described in Section 2 1 — but that only compared the representation of a given target document to known English, French, and Mi'kmaq text. English and French are official languages of Canada, and preliminary observations indicated that English and French were common in the data prior to language identification.

Table 1 shows the number of documents and tokens in corpora constructed using the random and Crúbadán approaches to query generation, for increasing numbers of queries. The growth in corpus size observed as the number of queries is increased suggests that, in future work, larger corpora could potentially be constructed by issuing more queries. Turning to the differences between the random and Crúbadán approaches to query generation, for each number of queries, the Crúbadán approach gives many more documents than the random approach, while the number of tokens is not drastically different between the two approaches.

Figure 1 shows the number of documents of varying lengths (measured in number of tokens) in the corpora created using the random and Crúbadán query generation strategies with 100 queries. Although the Crúbadán approach gives many more documents, many of these documents are very short. For the remainder of this paper we therefore consider the corpus built using the random approach — which is not dominated by very short documents — using 1000 queries.

In Table 2 we compare the number of documents, tokens, and types in the corpus created using the random approach to query generation with 1000 queries ("This paper"), with that of the Crúbadán Mi'kmaq corpus (Scannell, 2007). Here, and for the remainder of the paper, we use the language identification strategy described in Section 2 1 and tokenize our corpus using a simple regular expression-based tokenizer (whereas for the previous analysis a preliminary approach to language identification was used, and tokenization was based on whitespace). At 76k tokens, our corpus is substantially smaller than that of Scannell (2007). These differences in corpus size could be due in part to the relative aggressiveness with which the corpora have been cleaned, in steps such as boilerplate removal and deduplication. Nevertheless, the number of documents and types in our corpus are greater than that of the Crúbadán Mi'kmaq corpus, suggesting that there could be more diversity of authors, text types, or topics in our corpus. In future analysis, quantitative (Kilgarriff, 2001) or qualitative (Kilgarriff, 2012) corpus comparison methods could be applied in an effort to better understand their differences in composition.[12]

A Mi'kmaq speaker analyzed a sample of 25 randomly-selected paragraphs from our corpus to determine the precision of the language identification. 19 paragraphs (76%) were exclusively Mi'kmaq, while 4 paragraphs (16%) contained a mixture of Mi'kmaq and English, for a total of 23 paragraphs (92%) having Mi'kmaq content. Most of the content in the corpus is Mi'kmaq, although there is scope to improve the language identification. In the next section, we use this corpus in language modelling experiments.

3. Language Modelling

Language models are a key component of systems for many NLP tasks. As a first step towards our goal of building NLP systems for Mi'kmaq, we carry out preliminary language

[12]Crucially, this analysis would be possible because the methods of Kilgarriff (2001) and Kilgarriff (2012) require only word frequency lists, which are available for the Crúbadán corpora.

Language model	Bits-per-character
KenLM n=2	2.51
KenLM n=3	2.48
char-rnn	4.36
CNN	2.60

Table 3: Bits-per-character for each approach to language modelling considered.

modelling experiments. Because Mi'kmaq is a polysynthetic language, we consider character-level and subword-level language models, in addition to more-conventional (word) n-gram models. Specifically we consider the following approaches to language modelling:

KenLM We use KenLM (Heafield et al., 2013) to build conventional (word) n-gram language models with modified Kneser-Ney smoothing, for differing orders of n.

char-rnn We consider a character-level LSTM language model. Specifically we use a TensorFlow implementation[13] of char-rnn,[14] with its default parameter settings.

CNN Kim et al. (2016) propose a language model that forms word-level representations, based on a character-level CNN, which then feed into a multi-layer (word-level) LSTM. Kim et al. (2016) found this approach to perform particularly well on morphologically-rich languages, and so we are especially interested in evaluating it on Mi'kmaq. We use a TensorFlow implementation of this model,[15] with the "small" model settings from Kim et al. (2016) because of the relatively small size of our corpus.

We randomly split our corpus into 80% training, 10% development, and 10% testing data, based on sentences. We use a simple regular expression-based approach to detect sentence boundaries. We train our language models on the training data, use the development data for preliminary experiments and parameter tuning, and evaluate our models on the testing data.

KenLM and CNN predict words, and can therefore be evaluated in terms of perplexity. char-rnn, on the other hand, predicts characters, and is evaluated in terms of bits-per-character. Following Hwang and Sung (2017) we convert between perplexity (PPL) and bits-per-character (BPC) as follows:

$$\mathrm{PPL} = 2^{\mathrm{BPC} * \frac{N_C}{N_W}} \quad (1)$$

where N_C and N_W are the number of characters and words, respectively, in the test data.

[13] https://github.com/crazydonkey200/tensorflow-char-rnn

[14] https://github.com/karpathy/char-rnn

[15] https://github.com/mkroutikov/tf-lstm-char-cnn

Table 3 shows bits-per-character for each approach to language modelling considered. In the case of conventional n-gram models, a trigram model (KenLM n=3) outperforms a bigram model (KenLM n=2). Higher order n-gram models (not shown in Table 3) performed comparably to the trigram model. Both neural network-based models that incorporate character-level information, char-rnn and CNN, perform worse than conventional n-gram models. In both cases this could be because sufficient training data is not available to learn the parameters of the neural networks. However, the encouraging performance of CNN suggests that there is also substantial scope for future work to explore this model's various parameter settings — e.g., the dimensionality of the embeddings, the width and number of filters — to better tune it to very low-resource settings, and for the specific case of Mi'kmaq.

4. Discussion

NLP could potentially contribute to the preservation of Indigenous languages — which is particularly important in the current Canadian context of Truth and Reconciliation — by building tools to help use Indigenous languages in computer-mediated communication. To the best of our knowledge this is the first computational work to specifically consider Mi'kmaq. In this preliminary work, we built a web corpus of Mi'kmaq, and evaluated several approaches to language modelling for Mi'kmaq.

Our first direction for future work is to build a larger Mi'kmaq corpus. Our analysis in Section 2 2 indicates that this could be achieved by issuing more search engine queries. On the other hand, Scannell (2007) argues that web crawling using seed URLs returned by the queries is important for building corpora for low-resource languages. Although the corpus created in this paper includes more documents than the Mi'kmaq corpus of Scannell (2007), crawling could still be useful to find more Mi'kmaq documents, and we intend to explore this in future work.

Another important area for future work is language identification. Our analysis in Section 2 2 suggests that other languages are often present along with Mi'kmaq within a single paragraph. Some approaches to language identification are able to recognize which portions of text correspond to a particular language (Jurgens et al., 2017). Extending such methods to recognize Indigenous languages, and Mi'kmaq in particular, is an important area of future work. Future work on Mi'kmaq language identification should also take into account the variation in Mi'kmaq writing systems.

In Section 3 we have identified some potential future directions with respect to language modelling. Recent approaches to low-resource language modelling have incorporated cross-lingual word embeddings learned from bilingual dictionaries (Adams et al., 2017), which are available for Mi'kmaq. We also intend to evaluate such approaches to language modelling in future work.

Finally, because of the morphological complexity of Mi'kmaq, we are particularly interested in morphological analyzers for Mi'kmaq. As a first step, we intend to consider evaluating unsupervised approaches to morphological analysis (Smit et al., 2014) on Mi'kmaq.

5. Acknowledgements

We thank Starlit Simon for her annotation work which enabled the evaluation of the language identification. We thank David Perley for his input on this research. This research is financially supported by Mitacs and the University of New Brunswick.

6. Bibliographical References

Adams, O., Makarucha, A., Neubig, G., Bird, S., and Cohn, T. (2017). Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 937–947, Valencia, Spain.

Baroni, M. and Bernardini, S. (2004). BootCaT: Bootstrapping corpora and terms from the Web. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 1313–1316, Lisbon, Portugal.

Baroni, M., Kilgarriff, A., Pomikálek, J., and Rychlý, P. (2006). WebBootCaT: a web tool for instant corpora. In *Proceedings of XII EURALEX International Congress (EURALEX 2006)*, pages 123–131, Torino, Italy.

Battiste, M. (1985). Micmac literacy and cognitive assimilation. In Barbara Burnaby, editor, *Promoting Native Writing Systems in Canada*, pages 7–16. OISE Press/Ontario Institute for Studies in Education, Toronto, Canada.

Bird, S., Loper, E., and Klein, E. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc., Sebastopol, CA.

Brown, R. (2014). Non-linear mapping for improved identification of 1300+ languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 627–632, Doha, Qatar.

Cavnar, W. B. and Trenkle, J. M. (1994). N-gram based text categorization. In *Proceedings of the Third Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, USA.

DeBlois, A. D. (1990). *Micmac Texts*. Canadian Museum of Civilazation, Hull, Canada.

DeBlois, A. D. (1996). *Micmac Dictionary*. Canadian Museum of Civilazation, Hull, Canada.

Ferraresi, A., Zanchetta, E., Baroni, M., and Bernardini, S. (2008). Introducing and evaluating ukWaC, a very large web-derived corpus of English. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4): Can we beat Google?*, pages 47–54, Marrakech, Morocco.

Finn, A., Kushmerick, N., and Smyth, B. (2001). Fact or fiction: Content classification for digital libraries. In *Proceedings of the Second DELOS Network of Excellence Workshop on Personalisation and Recommender Systems in Digital Libraries*, Dublin, Ireland.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria.

Hwang, K. and Sung, W. (2017). Character-level language modeling with hierarchical recurrent neural networks. In *Proceedings of 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5720–5724, New Orleans, USA.

Johnson, P. (1996). Mi'kmaq. In Frederick E. Hoxie, editor, *Encyclopedia of North American Indians*, pages 376–378. Houghton Mifflin Company, Boston, USA.

Jurgens, D., Tsvetkov, Y., and Jurafsky, D. (2017). Incorporating dialectal variability for socially equitable language identification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 51–57, Vancouver, Canada.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics*, 6(1):97–133.

Kilgarriff, A. (2012). Getting to know your corpus. In *Proceedings of Text, Speech, Dialogue (TSD 2012)*, Brno, Czech Republic.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2016). Character-aware neural language models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, pages 2741–2749, Phoenix, USA.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012) Demo Session*, pages 25–30, Jeju, Republic of Korea.

Pomikálek, J. (2011). *Removing Boilerplate and Duplicate Content from Web Corpora*. Ph.D. thesis, Masaryk University.

Rand, S. T. (1888). *Dictionary of the language of the Micmac Indians : who reside in Nova Scotia, New Brunswick, Prince Edward Island, Cape Breton and Newfoundland*. Nova Scotia Printing Company, Halifax, Canada.

Scannell, K. (2007). The Crúbadán Project: Corpus building for under-resourced languages. In *Building and Exploring Web Corpora: Proceedings of the 3rd Web as Corpus Workshop, incorporating Cleaneval*, pages 5–15, Louvain-la-Neuve, Belgium.

Sharoff, S. (2006). Creating general-purpose corpora using automated search engine queries. In Marco Baroni et al., editors, *Wacky! Working papers on the Web as Corpus*, pages 63–98. GEDIT, Bologna, Italy.

Smit, P., Virpioja, S., Grönroos, S.-A., and Kurimo, M. (2014). Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden.

Truth and Reconciliation Canada. (2015). *Honouring the truth, reconciling for the future: Summary of the final report of the Truth and Reconciliation Commission of Canada*. Truth and Reconciliation Commission of Canada, Winnipeg, Canada.

Pronunciation Dictionaries for the Alsatian Dialects to Analyze Spelling and Phonetic Variation

Lucie Steiblé, Delphine Bernhard
LiLPa, Université de Strasbourg, France
lucie.steible@unistra.fr, dbernhard@unistra.fr

Abstract

This article presents new pronunciation dictionaries for the under-resourced Alsatian dialects, spoken in north-eastern France. These dictionaries are compared with existing phonetic transcriptions of Alsatian, German and French in order to analyze the relationship between speech and writing. The Alsatian dialects do not have a standardized spelling system, despite a literary history that goes back to the beginning of the 19th century. As a consequence, writers often use their own spelling systems, more or less based on German and often with some specifically French characters. But none of these systems can be seen as fully canonical. In this paper, we present the findings of an analysis of the spelling systems used in four different Alsatian datasets, including three newly transcribed lexicons, and describe how they differ by taking the phonetic transcriptions into account. We also detail experiments with a grapheme-to-phoneme (G2P) system trained on manually transcribed data and show that the combination of both spelling and phonetic variation presents specific challenges.

Keywords: Alsatian, transcription, spelling variation

1. Introduction

The Alsatian dialects, which belong to the High German dialects, are still spoken by approximately 500,000 speakers in Alsace, a region in north-eastern France (INSEE et al., 1999). Being non-dominant varieties, in a French-speaking country, they are mostly considered as *oral languages*, and have no standardized spelling system. There are however some occasions on which these dialects are written, dating back to 1816 when the first theater play in Alsatian, *Der Pfingstmontag* by Jean-Georges-Daniel Arnold, was published. Despite this written production, now stretching back to two centuries, almost no computational tools exist for the Alsatian dialects, which can therefore be considered as low-resourced.

Given the absence of a standardized spelling, this article aims at investigating the relations between oral and written forms. This work was performed using lexicographic and lexical resources: for one resource, we used the transcriptions which were already provided and for the other resources we manually transcribed recordings of native speakers illustrating the lexical entries. Letter sequences and phonemes were then automatically aligned. Finally, we evaluated the challenges for a grapheme-to-phoneme (G2P) tool facing both spelling and phonetic variation. Our ultimate objective is to provide resources for the Alsatian dialects, gain a better understanding of the phonological and spelling systems and develop tools able to deal with spelling variation in text corpora.

The main contributions of this article are as follows:

- We present the first pronunciation dictionaries for the Alsatian dialects, obtained by the manual transcription of several datasets ;
- We investigate phonetic and spelling variation in these datasets, with a focus on the specific status of consonants ;
- We compare various spelling systems for the Alsatian dialects, with a focus on their orthographic depth ;
- We train and evaluate a G2P system based on the manual transcriptions.

2. Related Work

Spelling variation is an issue for many different applications which have to process texts lacking spelling consistency (informal texts on the Web 2.0, text messaging, historical texts, dialects, etc.). While speech is rarely taken into account in traditional (written) text processing pipelines, it can be useful when dealing with spelling variation. Phonetic indexing algorithms, like Double Metaphone (Philips, 2000), have been developed in the context of information retrieval to account for spelling differences in words or names. The goal is to encode the input string using simplified phonetic rules. More sophisticated methods like grapheme-to-phoneme can also be used for searching words in corpora and dictionaries.[1] The goal is either to be able to retrieve words without knowing their exact spelling, or to abstract from the spelling variations in non-standardized languages, by using a phonetic index (Divay and Vitale, 1997).

In addition to improving the recall of queries, phonetic indexing and grapheme-to-phoneme techniques have been used to normalize texts in non-standard spellings. For instance, Cook and Stevenson (2009) take the phonemes of the standard word into account when normalizing graphemes in SMS text messages. Porta et al. (2013) integrate grapheme-to-phoneme transcription rules as well as rules expressing phonological change in a rule-based transducer from Old Spanish to Modern Spanish. These studies demonstrate that using knowledge about phonemes is relevant when dealing with non-standard spellings.

[1] See e.g. the *Trésor de la Langue Française* French dictionary (`http://atilf.atilf.fr/tlf.htm`) which uses pseudo-phonetic input or the Picartext textual database for the Picard language (`https://www.u-picardie.fr/LESCLaP/PICARTEXT/Public/`) which matches word forms based on phonetic correspondences.

In order to apply a grapheme-to-phoneme system to non-standard text a first obvious condition is to have such a system at disposal. In particular for low-resource languages, which also present high levels of spelling variation, the development of grapheme-to-phoneme systems is hindered by the lack of complete studies about the phonological system as well as the absence or the small volume of existing pronunciation dictionaries (word-pronunciation pairs). Moreover, when spelling is not normalized, designing grapheme-to-phoneme systems is known to be a complex task (Adda-Decker et al., 2011). The lack of resources can possibly be addressed by re-using data from other languages: Deri and Knight (2016) present an approach to train G2P models for low-resourced languages based on word-pronunciation pairs acquired from Wiktionary and on an adaptation of high-resource languages G2P models to closely related low-resource languages. This method nevertheless requires phoneme inventories to compute a phonetic distance metric between languages.

3. Datasets and Preprocessing

There are five main dialectal areas in Alsace, characterized by differences in their sound inventories: Rhine Franconian, South Franconian, High Alemannic, Low Alemannic from the north of the region, and Low Alemannic from the south of the region (Huck et al., 1999). The last two Low Alemannic variants are dominant on the Alsatian territory, and are the varieties studied here. They differ in several phonetic aspects, including the use of `[C]` in the south and `[X]` in the north after a front vowel, upholding of the `[g]` between vowels only in the south (in the north, the `[g]` became `[v]`) and more open vowels in final position in the south. All of them are German dialects, which creates a specific situation in this French region. According to a recent study, 43% of Alsatians can speak the dialect (OLCA / EDinstitut, 2012), and all of them also speak and write the national language, French.

The four datasets under study for the Alsatian dialects are the following:

- The DICTMULTI dataset is extracted from a printed multilingual dictionary (Adolf, 2006). It uses its own spelling system and provides the phonetic transcription for each word, made by the author of the dictionary. The whole dictionary is transcribed.
- The ELSASSICH dataset is taken from an online dictionary (*Elsässich Web diktionnair*) (Bitsch and Matzen, 2007). It uses its own spelling system, and the phonetic transcription was made manually by one of the authors of the present article, using the voice recordings provided for each lexical item. The whole online dictionary contains 3,333 entries, of which we transcribed 702.
- The OLCA datasets were produced by the local office for the preservation of Alsatian (Office pour la Langue et les Cultures d'Alsace et de Moselle, nd). It follows the ORTHAL spelling system (Zeidler and Crévenat-Werner, 2008), designed to be close to German spelling rules while at the same time preserving the variation (here between the northern part of the Alsace region –hereafter OLCA67– and the southern part –hereafter OLCA68). We used the same manual transcription process as for the ELSASSICH datasets, for a total of 2,859 entries, out of the 10,719 available.

The first dataset already had phonetic transcriptions, made by the dictionary author. Without audio files for the purpose of double checking, these transcriptions were used in their original form. The two last datasets, coming with audio, were phonetically transcribed by one of the authors, using the X-SAMPA transcription system (Wells, 1995). The resulting pronunciation dictionaries are available on the Zenodo platform (Steiblé, 2018) along with a documentation (Steiblé and Bernhard, 2018).

The written form was intentionally hidden during the transcription phase to avoid, as much as possible, the influence of the orthographic norms in use in the datasets. When there were doubts about some phonemes, specifically stops and their voice or voiceless quality, the waveforms were analyzed using PRAAT (Boersma and Weenink, nd).

Due to the high variability of the Alsatian dialects, no complete inventory of phonemes is available. We used, though, some area-specific inventories from sociolinguistic publications to help us (Zeidler and Crévenat-Werner, 2008; Huck et al., 1999; Bothorel-Witz et al., 1984).

Since the Alsatian dialects are in close contact with both French and Standard German, we also used datasets in these languages to draw a comparison between them and Alsatian. Our data for German come from two sources: MaryTTS (MARY) (Schröder and Trouvain, 2003), and Voxforge (VOX) (VoxForge, 2007). The French dataset is Lexique3 (LEX3) (New, 2006).

A quantitative summary of the datasets used in this study is provided in Table 1.

Language	Source	Entries	Phonemes
Alsatian	DICTMULTI	1,594	7,277
	ELSASSICH	702	4,372
	OLCA67	1,458	10,825
	OLCA68	1,401	10,472
German	MARY	26,233	218,669
	VOX	8,463	55,688
French	LEX3	125,733	834,011

Table 1: Summary of data resources

All word / phonetic transcription pairs were automatically aligned using the Phonetisaurus tool, a WFST-driven grapheme-to-phoneme framework (Novak et al., 2016). The alignment pairs link one or several letters (graphemes) with one phoneme. Figure 1 shows the alignment obtained automatically for the word *Spritzkànn* (watering can).

This alignment process allows us to observe the consistency between sounds and graphemes, in other words the possible differences between acoustical reality and the use of a matching letter or sequences of letters. All the analyses hereafter pertain to the 100 most frequent phoneme/grapheme pairs in each dataset (these pairs amount to 82 to 97% of all the pairs).



Figure 1: Example of a grapheme-phoneme alignment by Phonetisaurus. The phonetic transcription uses X-SAMPA.

4. Spelling and Phonetic Variations

4.1. Characters and Phonemes in Alsatian

Overall, there are 28 unique characters common to the spelling systems used in the four Alsatian datasets.[2] They correspond to 98.7% of the total character occurrences in all the datasets. Among the remaining 6 characters found,[3] *é*, *ï* and *x* are very rare and can be seen only in loanwords. *ë* and *ì* on the contrary are quite frequent. *ë*, pronounced `[E]`, is only used in DICTMULTI and *ì*, pronounced `[I]`, is used in DICTMULTI, OLCA67 and OLCA68.

There are 42 unique phonemes common to all the 4 Alsatian datasets.[4] These phonemes can be considered as the essential set required to describe the phonology of Alsatian dialects. They correspond to 95.6% of the total phoneme occurrences in all the datasets. The other phonemes are present in three, two or only one dataset.[5] These rarer phonemes are linked to the high variability seen in the Alsatian dialects, since some of them are allophones of the phonemes from the main list above: `B` (only in Low Alemannic from the south) and `b`, `C` (in Low Alemannic from the south) `X`, etc. Some of the variation is not related to the north/south geographical gap, but rather depends on individual variation: some speakers use `R`, and others `r`, for example. There is also a tendency found in some varieties from the south of the region to diphthong some vocalic sounds, which are very rare in other forms of Alsatian. Some phonemes are characteristic for French loanwords, such as the nasal vowels `[a_~]` and `[o_~]`.

4.2. Variation at the Sublexical Level

Based on the automatic alignments between graphemes and phonemes (see Section 3.), we first investigate the orthographic depth and consistency of each spelling system and then focus on the special case of consonants.

4.2.1. Orthographic Depth and Consistency

There are many ways to spell sounds. In alphabetical systems, a sort of ideal could be to have a straightforward relation between phonemes and characters. This optimal situation would create a direct relation between one phoneme and one character: always the same pronunciation for one character and always the same way to write one given phoneme. Such a system could be described as having a consistent, *shallow* orthography. On the opposite, when it is difficult to assess the pronunciation from the spelling, or when one phoneme can be written in various ways, the orthography is called *deep*.

Orthographic depth can be evaluated by counting the average number of pronunciations for one grapheme, and, in contrast, the average number of graphemes that can be used to spell one phoneme. These figures provide an accurate picture of the consistency between sounds and graphemes. We obtained those ratios by counting the average number of pronunciations for one given grapheme, and the number of ways to spell one given phoneme, for the 100 most frequent alignment pairs from our datasets. Of course, even when the phonetical transcriptions were made by one and the same person for the OLCA and ELSASSICH datasets, the writing system is not the same, leading to differences in the average numbers. Figure 2 displays the orthographic consistency for all our datasets, including German and French for the sake of comparison. The Alsatian dialects and German have quite similar ratios. Both can be described as having an almost "shallow orthography", or "transparent orthography". Compared to French, a language known for its complex spelling system, Alsatian is quite straightforward, despite the lack of a unified orthographic norm.

The most transparent spelling system in Alsatian is ELSASSICH. One possible explanation could be that the authors of this specific spelling system are linguists, which could account for a more stable relation between phonemes and graphemes. While these ratios are certainly a method to evaluate orthographic depth, other experiments should be performed to assess the degree of readability and usability of each system.

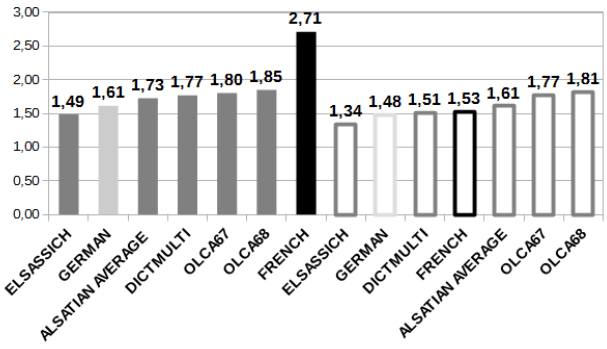


Figure 2: Average number of graphemes by phoneme, per language/spelling system (plain bars), and average number of pronunciations by grapheme (hollow bars).

4.2.2. Consonants and "Voicing"

The status of the consonants in Alsatian is quite unclear. In dictionaries, pronunciation tables are sometimes provided, but they are contradictory, and the consonants set differs from one to another. Fricatives are usually considered as always voiceless (Adolf, 2006) which leads to the lack of distinction between `[f]` and `[v]`, for example. The categorization of plosives is less clear.

In Alsatian, all plosives are voiceless (Steiblé, 2014). Despite this, there are two categories of plosives, opposed by the feature [*fortis*] (Jessen, 1998; Kohler, 1979). In writing, minimal pairs can be represented, using the same characters

[2] a à ä b c d e f g h i j k l m n o ö p q r s t u ü v w z

[3] ë é ï ì x y

[4] 9: 2 @ a a: ai b d e E e: E: f g h i I i: I: j k l m n N o O o: O: p r s S t u U u: v w X y y:.

[5] 2: 9 a_~ B Ei EI o_~ aI ao aU C R ts y9 Y z 3: ? @: 2I 2y 9i ei i@ ia Ia ie iE IE Oi U: Y: y9: ya

as in French or German, such as: *Gass* (street) versus *Kass* (crate). The difference between the two sets of plosives is linked to speech events, involving two distinct temporal patterns for the production of the consonants, as shown in Figure 3. The distinction between the series is based on various cues (Steiblé, 2014). The main cues are linked to the temporal orchestration of stops production. In both absolute and relative durations, the opposition is statistically significant. The Voice Termination Time (Agnello, 1975) is shorter in the *fortis* stops (13% of the total stop, versus 30% for the *lenis* stops), and the burst and Voice Onset Time (Klatt, 1975) phase is longer for the *fortis* stops (25% of the total stop, versus 17% for the *lenis*). The silent stop gap duration is longer in the *fortis* stops (63% of the total stop, versus 53% for the *lenis*).

During the transcription by one of the authors of the ELSASSICH, OLCA67 and OLCA68 datasets, if the consonant status was unclear, the recordings of the entries were observed on PRAAT (Boersma and Weenink, nd). If this visual check (searching for the patterns showed above) was not certain, the intra-segmental relative durations were compared to the values found in (Steiblé, 2014) for *fortis* and *lenis*, respectively. Despite the acoustical and written evidence, it has been said that the Alsatian dialects do not show any distinction between the consonants (Hug, nd), probably because they differ from both German and French sounds. In fact, our dataset tends to prove that there is enough distinction between the two series to allow for a strong phonological awareness, leading to the use of the two series of letters.

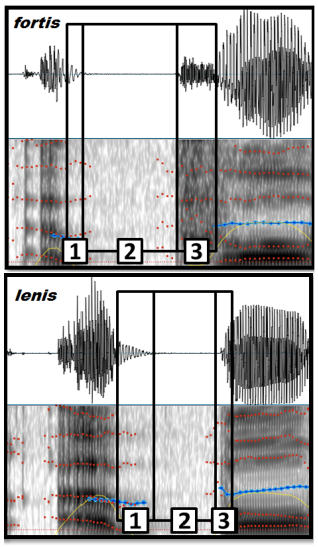


Figure 3: Typical temporal patterns of the Alsatian plosives: top = fortis plosive, bottom = lenis plosive. The rectangles frame three intra-segmental phases : 1, the Voice Termination Time, 2, the silent stop gap, and 3, the burst and Voice Onset Time.

In German, spelling rules would warrant that a voiceless consonant be written using the matching graphemes from the voiceless list (p,t,k,c,f,s,ss,ß,ch,sch,etc). In our German datasets, however, exceptions can be found and quantified, providing us with a consistency ratio. In Alsatian, admittedly, all consonants are voiceless, but the *fortis* ones are written using the same letters as the German voiceless set, and the *lenis* ones using the complementary voiced set (b,d,g,v,z,j,etc). The consistency ratio between a consonant and the characters used to spell it amounts to 95% in French (which accounts for the clear voiced/voiceless opposition in this language). In German, the ratio reaches 90%, and Alsatian reaches 89% (over all the four datasets), despite the lack of a unified orthographic norm. It is likely that the consistency ratio found in Alsatian is not related to the variation in spelling but to it being a German dialect.

The consistency ratio of Alsatian, when measured this way, is equivalent to that of standard German, despite the lack of a unified orthographic norm in Alsatian, as shown in Figure 4.

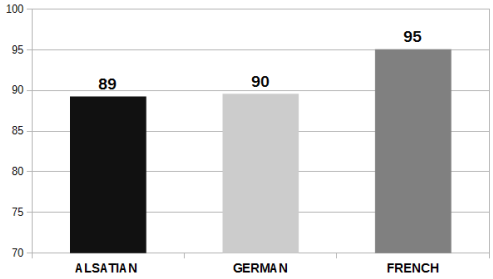


Figure 4: Percentage of consonant graphemes matching the phonemes, in terms of phonological oppositions

5. G2P for the Alsatian Dialects

In this section, we use the pronunciation dictionaries to train a G2P system. This presents a double challenge: (i) the small size of the pronunciation dictionaries and (ii) the presence of both phonetic and spelling variation. Given these challenges, our goal here is not to produce a full-fledged G2P system, but to assess the performance levels which can be reached for now, both on the four Alsatian datasets and on the task of retrieving spelling variants extracted from a corpus of texts in Alsatian.

For G2P, we chose to adopt a data-driven approach and trained the Phonetisaurus (Novak et al., 2016) system. The results are presented in Table 2 and are evaluated in terms of Word Error Rate (WER) – the percentage of lexicon entries with at least one error in their 1-best transcription– and Phoneme Error Rate (PER) – the Levenshtein distance between the predicted and the reference transcription, divided by the number of phonemes in the reference transcription (Hixon et al., 2011). We trained the G2P model on each dataset, then applied it to the other datasets. We also performed a closed test for each dataset by using the training data as test data. As could be expected, the results are low, given the small size of the training data. Training on the largest datasets (DICTMULTI and OLCA67) yielded the best results. The results obtained by training on OLCA67 and then applying the model to OLCA68 (and the other way round) could indicate that the use of the same spelling system plays a positive role. However, it should be mentioned that OLCA67 and OLCA68 also roughly correspond to the same vocabulary set, only for two different dialectal areas,

	DICTMULTI		ELSASSICH		OLCA67		OLCA68	
Training dataset	**WER**	**PER**	**WER**	**PER**	**WER**	**PER**	**WER**	**PER**
DICTMULTI	*11.70%*	*4.49%*	82.75%	29.83%	88.98%	35.49%	90.70%	38.02%
ELSASSICH	88.76 %	37.59%	*47.37%*	*19.04%*	96.12%	45.32%	97.14%	48.59%
OLCA67	71.59%	27.45%	85.67%	31.90%	*23.87%*	*7.18%*	66.52%	20.22%
OLCA68	72.33%	29.14%	85.67%	33.19%	67.02%	20.84%	*29.30%*	*9.52%*

Table 2: G2P results. The best results for each dataset have a grey background. The results of the closed tests are in italics.

and this could account for the results obtained. We also tried to train models using combinations of the datasets but this only leads to small improvements or no improvements at all, despite the increase in the amount of training data. This might indicate that the system is not able to handle what might often correspond to contradicting cues coming from heterogeneous datasets.

Finally, we applied the four Alsatian G2P models to a list of 110 words extracted from a corpus of texts in Alsatian written by several authors and corresponding to several dialectal areas and spelling systems. These words are grouped in 28 variant clusters, e.g., [*Frejndschaft ; Freundschaft ; Friindschàft ; Frindschàft ; Frìndschàft*] (friendship). Our goal here is to assess the ability of the G2P models to provide identical transcriptions for spelling variants. The model which performs best at this task is OLCA67, with an F-measure of 0.11. As could be expected, the precision is high (1.0), but the recall is very low (0.06), especially when compared with a rule-based Double Metaphone approach (Bernhard, 2014), which is less precise (0.47) but has a much better recall (0.90). Here however the results can be improved by pooling all the datasets for training the G2P model, leading to a better F-measure (0.15) due to an increase in recall (0.08), with only a very small decrease in precision (0.97). When the task does not require phonetic realism, but rather the ability to generalize over spelling variants, a non-homogeneous training dataset seems to be a better option. One research direction could be to add known spelling variants in the pronunciation dictionaries before training.

6. Conclusion

In this article, we have described four word-pronunciation datasets for the Alsatian dialects, covering different spelling systems. Our analysis of the relationship between sound and spelling has shown that we observe spelling and phonetic variation between the datasets, but that the spelling systems are rather self-consistent, having an almost shallow orthography. We have also identified a set of essential phonemes required to describe Alsatian phonology. In future work, we plan to increase the size of the pronunciation dictionary using a semi-automated approach relying on the G2P models described in Section 5.. The pronunciation dictionaries could be used in the future to create writing aids (spellchecking), or to assist dictionary lookup for spelling variants.

7. Acknowledgements

This work was supported by the French “Agence Nationale de la Recherche” (ANR) (project no.: ANR-14-CE24-0003). We would like to thank Paul Adolf for kindly providing an electronic copy of his dictionary.

8. Bibliographical References

Adda-Decker, M., Lamel, L., Adda, G., and Lavergne, T. (2011). A First LVCSR System for Luxembourgish, a Low-Resourced European Language. In *Language and Technology Conference*, pages 479–490. Springer.

Adolf, P. (2006). *Dictionnaire comparatif multilingue: français-allemand-alsacien-anglais*. Midgard, Strasbourg, France.

Agnello, J. (1975). Voice onset and voice termination features of stutterers. In L.M Webster et al., editors, *Vocal tract dynamics and dysfluency: the proceedings of the first annual Hayes Martin Conference on Vocal Tract Dynamics.*, New-York. Speech and Hearing Institute.

Bernhard, D. (2014). Adding Dialectal Lexicalisations to Linked Open Data Resources: the Example of Alsatian. In *Proceedings of the Workshop on "Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era" at LREC 2014*, pages 23–29, Reykjavík, Islande.

Bitsch, R. and Matzen, R. (2007). de Elsässich Web diktionnair. Online: `http://www.ami-hebdo.com/elsadico/index.php`.

Boersma, P. and Weenink, D. (nd). Praat, a system for doing phonetics by computer [Computer program]. Downloaded from: `http://www.praat.org`.

Bothorel-Witz, A., Philipp, M., and Spindler, S. (1984). *Atlas linguistique et ethnographique de l'Alsace*. CNRS.

Cook, P. and Stevenson, S. (2009). An Unsupervised Model for Text Message Normalization. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*, CALC '09, pages 71–78.

Deri, A. and Knight, K. (2016). Grapheme-to-phoneme models for (almost) any language. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 399–408.

Divay, M. and Vitale, A. J. (1997). Algorithms for grapheme-phoneme translation for English and French: Applications for database searches and speech synthesis. *Computational linguistics*, 23(4):495–523.

Hixon, B., Schneider, E., and Epstein, S. L. (2011). Phonemic Similarity Metrics to Compare Pronunciation Methods. In *INTERSPEECH*, pages 825–828.

Huck, D., Laugel, A., and Laugner, M. (1999). *L'élève dialectophone en Alsace et ses langues*. Oberlin, Strasbourg, France.

Hug, M. (nd). Orthographe alsacienne. Online: `http://elsasser.free.fr`.

INSEE, Barre, C., and Vanderschelden, M. (1999). Insee - Population - L'enquête "Etude de l'histoire familiale" de 1999.

Jessen, M. (1998). *Phonetics and Phonology of Tense and Lax Obstruents in German*. Studies in Functional and Structural Linguistics. John Benjamins Publishing Company, Amsterdam, jan.

Klatt, D. (1975). Voice onset time, frication, and aspiration in word-initial consonant clusters. *Journal of speech and hearing research*, 18(4):686–706.

Kohler, K. (1979). Phonetic Explanation in Phonology: The Feature Fortis/Lenis. *Phonetica*, 36:332–343.

New, B. (2006). Lexique 3: Une nouvelle base de données lexicales. In *Actes de la Conférence Traitement Automatique des Langues Naturelles (TALN 2006)*.

Novak, J. R., Minematsu, N., and Hirose, K. (2016). Phonetisaurus: Exploring grapheme-to-phoneme conversion with joint n-gram models in the WFST framework. *Natural Language Engineering*, 22(06):907–938, nov.

Office pour la Langue et les Cultures d'Alsace et de Moselle. (nd). Olca, lexiques français-alsacien. Online: `https://www.olcalsace.org/fr/lexiques`.

OLCA / EDinstitut. (2012). Etude sur le dialecte alsacien. Online: `https://www.olcalsace.org/sites/default/files/documents/etude_linguistique_olca_edinstitut.pdf`.

Philips, L. (2000). The Double Metaphone Search Algorithm. *C/C++ Users Journal*, 18(6):38–43, June.

Porta, J., Sancho, J.-L., and Gómez, J. (2013). Edit Transducers for Spelling Variation in Old Spanish. In *Proceedings of the Workshop on Computational Historical Linguistics at NoDaLiDa 2013*, volume 87 of *Linköping Electronic Conference Proceedings*, pages 70–79.

Schröder, M. and Trouvain, J. (2003). The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching. *International Journal of Speech Technology*, 6:365–377.

Steiblé, L. (2014). *Le contrôle temporel des consonnes occlusives de l'alsacien et du français parlé en Alsace*. Phd thesis, Université de Strasbourg.

Steiblé, L. and Bernhard, D., (2018). *Phonetic Transcription for the Alsatian Dialects*. DOI: 10.5281/zenodo.1174219.

VoxForge. (2007). Free Speech... Recognition (Linux, Windows and Mac) - voxforge.org.

Wells, J. (1995). Computer-coding the IPA: a proposed extension of SAMPA. Online: `https://www.phon.ucl.ac.uk/home/sampa/ipasam-x.pdf`.

Zeidler, E. and Crévenat-Werner, D. (2008). *Orthographe alsacienne: bien écrire l'alsacien de Wissembourg à Ferrette*. J. Do Bentzinger, Colmar, France.

9. Language Resource References

Steiblé, L. (2018). Pronunciation Dictionaries for the Alsatian Dialects. DOI: 10.5281/zenodo.1174214.

ChAnot: An Intelligent Annotation Tool for Indigenous and Highly Agglutinative Languages in Peru

Rodolfo Mercado-Gonzales, José Pereira-Noriega, Marco Sobrevilla, Arturo Oncevay
Research Group on Artificial Intelligence (IA-PUCP)
Departamento de Ingeniería, Pontificia Universidad Católica del Perú, Lima, Peru
{rmercado, jpereira, msobrevilla, arturo.oncevay}@pucp.edu.pe

Abstract

Linguistic corpus annotation is one of the most important phases for addressing Natural Language Processing (NLP) tasks, as these methods are deeply involved with corpus-based techniques. However, meta-data annotation is a highly laborious manual task. A supportive alternative requires the use of computational tools. They are likely to simplify some of these operations, while can be adjusted appropriately to the needs of particular language features at the same time. Therefore, this paper presents ChAnot, a web-based annotation tool developed for Peruvian indigenous and highly agglutinative languages, where Shipibo-Konibo was the case study. This new tool is able to support a diverse set of linguistic annotation tasks, such as morphological segmentation markup, POS-tag markup, among others. Also, it includes a suggestion engine based on historic and machine learning models, and a set of statistics about previous annotations.

Keywords: Annotation Tool, Corpus Annotation, Peruvian Indigenous Languages, Shipibo-Konibo

1. Introduction

The research field of Natural Language Processing (NLP) aims the analysis automation, representation and generation of human language through computational techniques (Cambria and White, 2014). These methods usually are based on supervised machine learning approaches, which employ the analysis of corpus, composed of annotated samples, and other strategies for reaching a particular NLP goal. The samples within a corpus need to be enriched with additional information or meta-data, plus the correct solution regarding the NLP task, generating an annotated linguistic corpus (Müller and Strube, 2006).

Nevertheless, the development of an annotated linguistic corpus might be a highly time-consuming task. The process requires mainly human effort, which is performed by linguists or experts in a particular language. While the corpus is more specialized or the language is less widespread, the expertise requirements will increase. In that way, an easier specialized annotation tool is what is needed. (Müller and Strube, 2006).

This paper introduces ChAnot, a new web-based annotation tool, which is focused in Peruvian indigenous and highly agglutinative languages. ChAnot enables the corpus annotation for various NLP tasks, such as Morphological Analysis (lemma and morphological segmentation), Part-of-Speech tagging, Name Entity Recognition and Syntactic Analysis (using a BRAT[1] interface (Stenetorp et al., 2012)). The interactive interface and architecture are adjusted appropriately to the needs of the Peruvian indigenous languages and humans annotators. Moreover, ChAnot includes a suggestion engine based in machine learning models and a set of important statistics about historic annotation.

The text below is organized as follow. Section 2 presents some existing annotation tools. Then, the main and distinctive functionalities in ChAnot are detailed in Section 3. Later, Section 4 presents briefly features regarding the Peruvian indigenous languages, focusing in Shipibo-Konibo, which is the case study language. Finally, conclusions and future works are discussed.

2. Related Work

Due to the increasing of NLP applications during the last years, an important number of annotation tools have been developed. Each tool has features according to specific objectives. Therefore, there are annotation tools supporting a set of NLP tasks independently of the language, while other are focused in a specific kind of languages.

One of the mainly important annotation software is MMAX2[2], which is a GUI-based tool, that like most of the other tools lets the users to select a portion of text and annotate some properties over it (text span annotation). This process enables the markup of POS-tag, word senses, coreference, dependency relations, among others. Besides, the XML format is used to store the meta-data (Müller and Strube, 2006).

Most of the current annotation tools are web-based, language independent, use machine learning for managing suggestions and support a variety set of text annotation tasks. Within this range, BRAT is likely to be the most popular one, which additionally includes high-quality annotation visualization and is fully configurable (Stenetorp et al., 2012). Another popular tool is WebAnno[3], which additionally offers annotation project management (including management of users and roles) (Yimam et al., 2013).

Likewise, as it was mentioned before, there are some annotation tools developed according to features of a specific language, such as Fassieh (Attia et al., 2009). This GUI-based tool lets the user to perform morphological, POS-tag, phonetic and semantic annotations for Arabic texts.

[1] Available in: http://brat.nlplab.org/

[2] Available in: http://mmax2.net/

[3] Available in: https://webanno.github.io

As it may be observed, the described tools have many features in common. Generally speaking, they are very helpful for different annotation tasks. Nevertheless, a customized tool for the features and reality of Peruvian indigenous languages is what is pursued in this study. The main motivations and specific functionalities are described in next section.

3. ChAnot Annotation Tool

The main reason for developing a new tool was the need to exploit morphological-rich Peruvian languages. Performing morphological segmentation among with morpheme meta-data annotation has not been entirely possible in most of the other tools, as it was noticed in the previous part. Another motivation is the lack of experience of the human annotators (linguists and native speakers of Peruvian languages), who have never done this kind of work before, and even some of them are beginners in the use of software.
In this context, ChAnot[4] is an intelligent web-based annotation tool focused in Peruvian indigenous and agglutinative languages, which allows text processing through morphological (lemma and morphemes), POS-tag, named entity and syntactic annotation (with an interface integrated with BRAT).
The name of the tool is composed from the terms *Chana*, the native name of a bird that represents knowledge in the Shipibo-Konibo culture, and the first part of *anotador*, which means annotator in Spanish. The latter is the main official language in Peru, and the reason why the interface has included Spanish terminology.
Likewise, ChAnot was implemented using a client-server architecture that can be accessible from any modern web browser, using a back-end implemented in Java. The annotated data is stored on a MySQL database for most tasks, except for the syntactic annotation which is saved in BRAT format.

3.1. ChAnot Workflow

ChAnot was designed to perform two main annotation phases. The first phase is a morphological and POS-tag annotation. Then, using the previous information, and with the support of a BRAT interface, enables a syntactic annotation phase. NER task annotation is also available, although not integrated in the main work flow.

- **Input Phase**
  ChAnot receives a plain text file encoded in UTF-8 with a sentence per line. Each sentence has two parts (separated by a vertical bar): the own sentence in indigenous language followed by the translation of this sentence in Spanish (translation is not necessary), as it is shown in Figure 1.

- **Annotation Phase**
  After uploading the input file to ChAnot, the user must select an annotation task to perform. The only restriction for performing syntactic annotation is the requirement of a previous morphological and POS-tag annotation.

[4]Source code and video-demo available in: https://github.com/iapucp/chanot-lrec2018 and chana.inf.pucp.edu.pe/resources/chanot

```
Rámara pápa Iquitoain iki | Ahora papá está en Iquitos
Bakísha ea kái | Voy a ir mañana
```

Figure 1: Input format sample

- **Output Phase**
  Finally, the annotation is saved in a database. It can also be exported in a XML file structured by sentences, words and affixes. Figure 2 presents an output XML sample.

```
<corpus>
  <sentence id="0" text="Escuela tapo non matsóti iki ." translation="Barreremos el suelo.">
    <word id="0" lemma="escuela" posTag="Nombre" subPosTag="Contable" token="Escuela"/>
    <word id="1" lemma="tapo" lemmaPred="tapo" posTag="Nombre" subPosTag="Contable" token="tapo"/>
    <word id="2" lemma="noa" posTag="Pronombre" subPosTag="Personal" token="non">
      <affix order="1" text="n" type="Clít. nominal"/>
    </word>
    <word id="3" lemma="matsóti" posTag="Verbo" subPosTag="Transitivo" token="matsóti">
      <affix order="1" text="ti" type="Sufijo verbal"/>
    </word>
    <word id="4" lemma="iki" posTag="Verbo Auxiliar" subPosTag="" token="iki"/>
    <word id="5" lemma="." posTag="Puntuación" subPosTag="" token="."/>
  </sentence>
</corpus>
```

Figure 2: Output format sample

3.2. ChAnot Functional Features

Unlike other tools, ChAnot enables a complete morphological annotation (lemma plus morphological segmentation with meta-data) and possess a communication interface with BRAT for dependency syntax annotation using the previous meta-data. The main features of ChAnot are detailed below.

- **Accessibility**
  Since the intended users of ChAnot are people who are not very familiar with computers or technology, it is a must to have a tool that can be accessible from anywhere and without the need to install any complicated software. In order to accomplish that, ChAnot is accessible from any modern web browser.

- **User Management**
  Each human annotator has a private account to work in ChAnot. This account let them manage their annotation files through a menu (see Figure 3). There is no crossover of information and no possibility of getting corrupted data by external users either.

- **Statistics**
  ChAnot generates a set of important statistics per annotation file and per user, such as the current number of annotated sentences or words, and even the average time that was spent for the annotation task of each sentence. These statistics could be very useful as a complexity metric in the evaluation of the historically annotated sentences. Besides, it will be useful to evaluate the progress of human annotators for further analysis.

- **Interactive Interface**
  In order to ease annotation tasks for users, the interfaces were designed to be intuitive and they transmit information trough different colors. In most of the ChAnot interfaces, green means processed or completely annotated, yellow refers to work in progress, whereas red represents

a not processed task or data. Figures 3 and 5 illustrates the color usage.

Likewise, all text and messages in this tool are in Spanish, because it is the most spoken language in Peru.

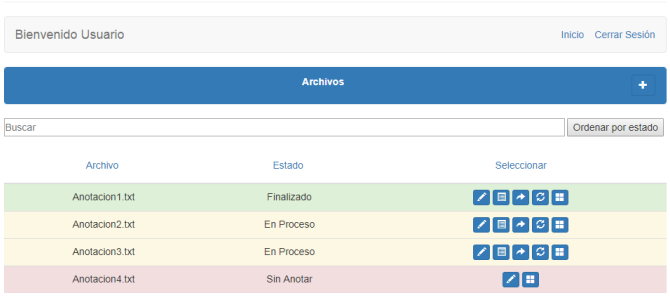


Figure 3: A user's main menu interface with a list of documents available for annotation tasks. The different colors represents the progress in each file.

- **Automatic Tokenization**
  ChAnot automatically split the input sentences in words, numbers, symbols and punctuation marks. This is possible because Peruvian indigenous languages and our tool use white spaces as separators, so annotation is performed at the token level. This feature differs from most of other tools, which are mainly based on text span annotation.

- **BRAT Interface**
  Additionally, it was decided to take advantage of the BRAT capabilities in dependencies annotation for syntax, so a direct connection was configured from ChAnot.

  In the interface, each annotated sentence of the file is transformed into the BRAT files format. For that purpose, each word is split in its morphemes according the morphological annotation in ChAnot, which is entirely preserved (see the details in subsection 3.3). Figure 4 presents an annotation task in BRAT, where the POS-tags and morphological segmentation was obtained from the ChAnot interface.

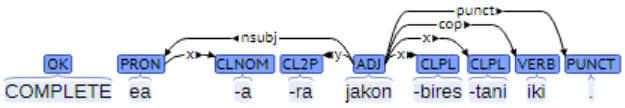


Figure 4: Dependencies annotation with a BRAT interface.

It is important to highlight, that some dependency rules for annotation were introduced in the interface, in order to reduce the workload for the annotators in BRAT.

3.3. NLP Annotation Tasks in ChAnot

- **Morphological Annotation**
  In the morphological scope, annotation through ChAnot allows a complete morphological segmentation. For each word in a sentence, the lemma is introduced, and if applicable all the affixes could be added. The lemmas and morphemes must be annotated with their respective category, which are customizable regarding the language through a configuration file. Besides, the relative position of the sub-word units in the term is a requirement for finishing each word annotation. Figure 5 shows an example of a word with two affixes.

  As it is observed, the annotation includes the most relevant morphological information of each word plus the full meaning of each affix. The corpus can be exploited for other tasks such as morphological segmentation or disambiguation.

- **POS-Tag Annotation**
  ChAnot allows Part-of-Speech tagging using a two-level predefined POS-tag set. The first level provides general information about grammatical properties of a word, while the second describes more granulated information regarding the different kinds of the upper level.

  The POS-tag set can be modified according to the preferences or some specific language features. For the case study (Section 4), the definition of a POS-tag set aligned to the UD (Universal Dependencies) standard (Nivre et al., 2016) was an important factor.

- **Named Entity Annotation**
  As a complementary function, it is possible to identify if there is a named entity (NE) in the text. In ChAnot, the annotation is performed by introducing the specific NE category and a relative position tag. It is possible to discriminate single and multi-word entities in the sentence.

- **Syntactic Dependencies Annotation**
  A Universal Dependency (UD) Treebank (Nivre et al., 2016) is the main resource-like goal for the Peruvian indigenous languages (work in-progress). For that purpose, ChAnot includes a BRAT interface, as it was previously described.

3.4. Automatic Suggestions in ChAnot

ChAnot has integrated three different machine learning-based models that assist the workload of the annotators. One for the automatic identification of POS-tags, the second one for the lemma prediction and a third for named entities recognition. The first two models are part of the Ship-LemmaTagger toolkit (Pereira-Noriega et al., 2017), while the latter one is a newly hybrid model that uses a combination of rules and predictions based on previous annotations.

All of these predictive models are implemented as Python web services in the server side. Furthermore, they are automatically updated in a periodic scheme with data coming from new annotations that are made by the users. As these models are frequently adjusted, the quality improvement of their suggestions is what is expected.

Furthermore, there are two kind of suggestions embedded in ChAnot: historic and machine learning-based. The former recalls the previous annotations (lemma, POS-tags and affixes) that any user performed in the past, and presents the suggestion marking the word with a yellow fill. The latter one works as it was described before, and the suggestions are presented for entirely new words, highlighting them with a red fill. Figure 5 presents a sample.

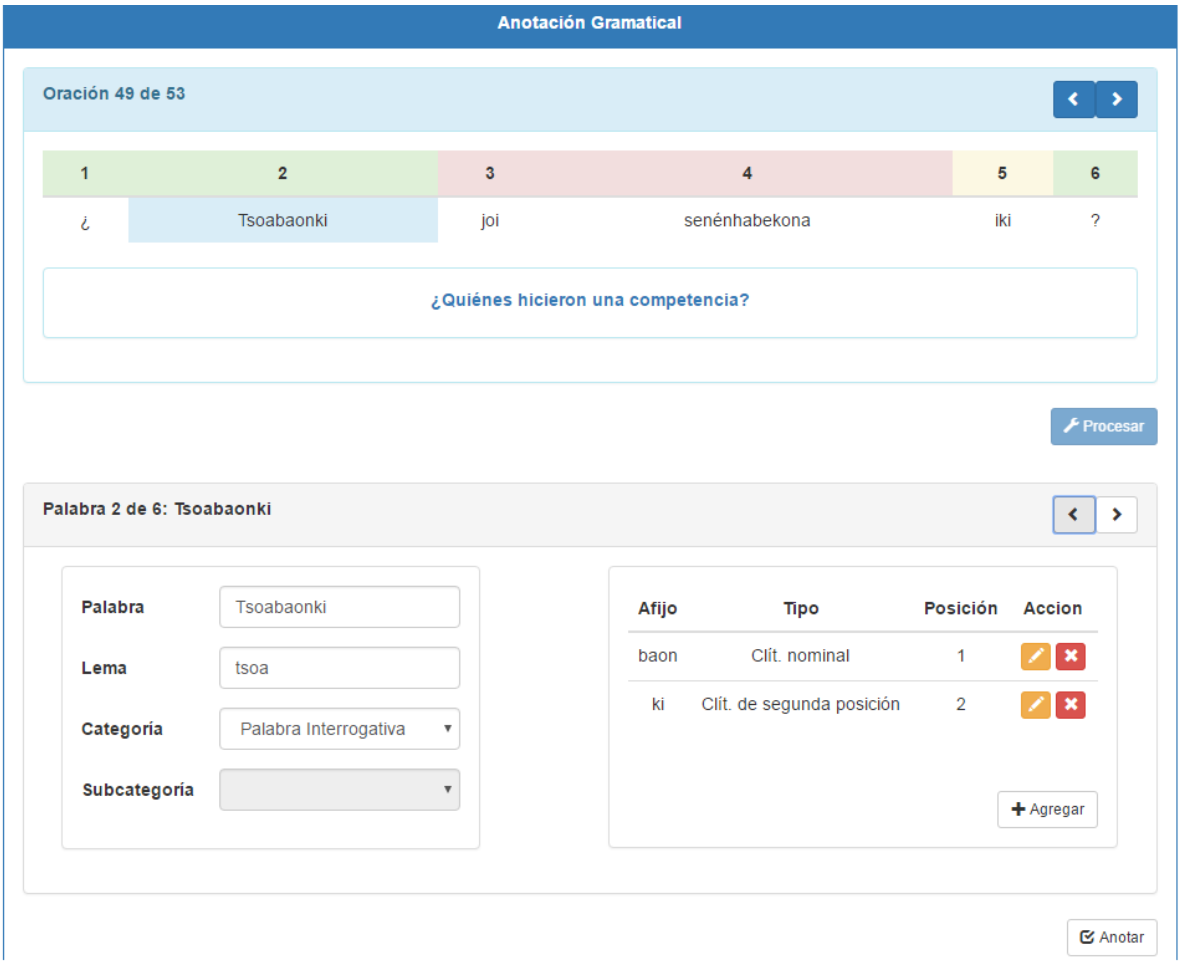


Figure 5: Morphological and grammatical annotation sample (*Who made a competition?*): the word *Tsoabaonki* was split in *tsoa +baon +ki*, and each sub-word unit received additional information. Regarding the colors, the 3rd and 4th tokens has not been annotated yet (red), but they contain machine learning-based suggestions, whereas the 5th term (yellow) includes a historic-based suggestion.

4. Case Study: Shipibo-Konibo (shp)

4.1. Background

Peru presents a diverse culture map including many indigenous communities and cultures, who are minorities in the country. In order to support the preservation of their traditions and languages, the Ministry of Culture of Perú has identified 19 linguistic families including 47 indigenous languages, and 24 of them have been made official for government-service purposes. Among them, Shipibo-Konibo is the sixth language with the highest number of native speakers, with about 22 517 speakers, and is taught in 299 public schools through bilingual educational programs (Ministerio de Cultura del Perú, 2016).

Shipibo-Konibo is a highly agglutinative language, with more than 100 suffixes and about 13 prefixes for word inflection (Valenzuela, 2003). Besides, there are not too many academic experts with experience in computational annotation tasks. In that context, ChAnot reflects as much information of this language for solving specific NLP tasks, while at the same time tries to be as easy as possible for unexperienced annotators. In the same way, ChAnot could be extended for more Peruvian indigenous languages, who share functional linguistics properties in the morpho-syntax domain, such as Kakataibo, Matsés, among others.

4.2. Corpus Annotation and Evaluation

Using ChAnot, the experts have developed a corpus of 1630 annotated sentences (Shipibo-konibo sentences were collected from educational texts[5]), where each word within them contains: annotation of lemma, POS-tag, sub-POS-tag, and a list of all the affixes that compose the word. These affixes include category and relative positions. Besides, 204 and 78 sentences got named entity and dependencies (with the BRAT interface) annotations, respectively. However, the latter two tasks are not part of the analysis.

An evaluation of the machine learning-based suggestion engine was performed. It was simulated an increasing annotat ion scheme with automatically updated models in a 300-sentence block. In this sense, the accuracy achieved a peak of 74% for lemmatization, and 89% for POS-tagging (see Figure 6).

[5]Available in: https://github.com/iapucp/chanot-lrec2018/tree/master/resources/data

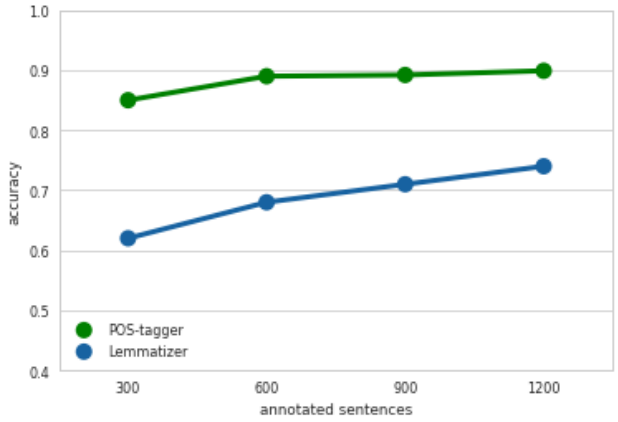


Figure 6: Evaluation of the lemmatizer and POS-tagger integrated in ChAnot

5. Conclusions and Future Work

The need for an annotation tool customized around the features of Peruvian indigenous languages allowed the design and implementation of ChAnot. The tool is likely to speed up the construction of linguistic corpora, by presenting suggestions based on past annotations samples, as well as predicting suggestions for new words with machine learning models that are frequently adjusted with newly annotated data. Furthermore, the wide range of functional features and annotations tasks are core points of the tool, which has the potential to scale easily for more complex jobs.

Future work is focused in the functional aspect. First, crowdsourcing, as it is included in WebAnno (Yimam et al., 2013), may be a relevant feature for enable the participation of more experts. However, the simultaneously integration of crowdsourcing and active learning schemes is what is really desired. The initial steps has been already made, as there are now some predictive models embedded in the tool. Another functional feature would be the inclusion of a spell-checker for the indigenous language, as it might work as a previous validation step for the required text to annotate (Alva and Oncevay, 2017).

Furthermore, there are new tasks that are expected to be annotated in the short term. Alignment is one of them, due to the presence of the translated text in Spanish (from parallel corpora). That could help enormously in the corpus-based machine translation experiments that has been performed recently (Galarreta et al., 2017). Finally, other NLP level tasks may be supported in the future, such as the semantic layer with word sense disambiguation annotations.

Acknowledgments

We highly appreciate the linguistic team effort that made possible the corpus annotation as well as provided feedback for the interaction functionalities: Dr. Roberto Zariquiey, Alonso Vásquez, Gabriela Tello, Renzo Ego-Aguirre, Lea Reinhardt and Marcela Castro. We are also thankful to our native speakers (Shipibo-Konibo) collaborators: Juan Agustín, Carlos Guimaraes, Ronald Suárez and Miguel Gomez. Finally, we gratefully acknowledge the support of the "Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica" (CONCYTEC, Peru) under the contract 225-2015-FONDECYT.

6. Bibliographical References

Alva, C. and Oncevay, A. (2017). Spell-checking based on syllabification and character-level graphs for a Peruvian agglutinative language. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 109–116.

Attia, M., Rashwan, M. A., and Al-Badrashiny, M. A. (2009). Fassieh, a semi-automatic visual interactive tool for morphological, PoS-Tags, phonetic, and semantic annotation of Arabic text corpora. *IEEE transactions on audio, speech, and language processing*, 17(5):916–925.

Cambria, E. and White, B. (2014). Jumping NLP curves: A review of natural language processing research. *IEEE Computational intelligence magazine*, 9(2):48–57.

Galarreta, A. P., Melgar, A., and Oncevay, A. (2017). Corpus creation and initial SMT experiments between Spanish and Shipibo-Konibo. In *RANLP (In-press)*.

Ministerio de Cultura del Perú. (2016). Base de datos de pueblos indígenas u originarios - Pueblos indígenas del Perú. http://bdpi.cultura.gob.pe/lista-de-pueblos-indigenas. Accessed: 2016-31-03.

Müller, C. and Strube, M. (2006). Multi-level annotation of linguistic data with MMAX2. *Corpus technology and language pedagogy: New resources, new tools, new methods*, 3:197–214.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal Dependencies v1: A multilingual treebank collection. In *LREC*.

Pereira-Noriega, J., Mercado-Gonzales, R., Melgar, A., Sobrevilla-Cabezudo, M., and Oncevay-Marcos, A. (2017). Ship-LemmaTagger: Building an NLP toolkit for a Peruvian native language. In *International Conference on Text, Speech, and Dialogue*, pages 473–481. Springer.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). BRAT: A web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

Valenzuela, P. (2003). *Transitivity in Shipibo-Konibo grammar*. Ph.D. thesis, University of Oregon.

Yimam, S. M., Gurevych, I., de Castilho, R. E., and Biemann, C. (2013). WebAnno: A flexible, web-based and visually supported system for distributed annotations. In *ACL (Conference System Demonstrations)*, pages 1–6.

The DLDP Survey on Digital Use and Usability of EU Regional and Minority Languages

Claudia Soria, Valeria Quochi, Irene Russo
CNR-ILC
Pisa, Italy
{firstname.lastname}@ilc.cnr.it

Abstract

The digital development of regional and minority languages requires careful planning to be effective and should be preceded by the identification of the current and actual extent to which those languages are used digitally, the type and frequency of their digital use, the opportunity for their use, and the main obstacles currently preventing it. This paper reports about the design, the results and the key findings of an exploratory survey launched by the Digital Language Diversity Project about the digital use and usability of regional and minority languages on digital media and devices. The aim of the survey - the first of this kind - was to investigate the real usage, needs and expectations of European minority language speakers regarding digital opportunities, with a strong focus on electronic communication. The survey is restricted to four languages (Basque, Breton, Karelian and Sardinian) at different stages of digital development, which offers a starting point to develop strategies for assessing digital vitality of these languages and overcoming specific difficulties such as, for instance, the lack of official data.

Keywords: minority languages, digital survival, electronic communication

1. Background and Motivation

In this paper we present the results of the first survey about the actual usage of four European minority languages and the related needs of their speakers in terms of digital opportunities. The survey is part of the work carried out by the Digital Language Diversity Project (DLDP) (Soria et al., 2016)[1], a three-year Erasmus+ project started in September 2015.

The goal of the DLDP is to help minority language speakers' communities in the acquisition of intellectual and practical skills to create, share, and reuse online digital content in their languages. At the same time we want to define general guidelines and best practices for the promotion of minority languages with poor digital representation, a fact that further prevents their usability on digital media and devices.

One of the underlying assumptions of the Digital Language Diversity Project is that the sustainability and preservation of regional and minority languages is closely tied to their being perceived by their speakers as being fully-fledged languages that can be used in any context, the digital one included. Unfortunately, this is far from being a reality not only for regional and minority languages, but for the majority of the world languages. In most cases, the technical or infrastructural impediments for the digital use of European regional and minority languages are modest and fairly easily solvable. Marginalisation and minoritisation of those languages mostly derives from the concurrency of the national and global languages for which digital content and services are more easily available, which further discourages regional and minority language speakers from using those languages digitally. In order to break this vicious circle and make those languages digitally appealing and usable to an extent that can compete with other major languages, it is necessary to approach the problem in terms of "digital language planning".

In order to be able to plan for digital development, we first need to identify the current and actual extent to which RML are used digitally, the type and frequency of their digital use, the opportunity for their use, and the main obstacles currently preventing it so as to get a clear understanding of the different factors that may affect the digital use of RMLs. Some reports carried out for individual languages and specific media are available, like the Language White Papers (Uszkoreit and Rehm, 2012) published by the META-NET Network that has clearly shown how 30 European languages are at risk of digital extinction because of lack of sufficient support in terms of language technologies. The META-NET work, initially for each of the EU official languages, was then extended to cover as well some regional languages such as Basque (HernÃ¡ez et al., 2012), Catalan (Moreno et al., 2012), Galician (GarcÃa-Mateo and Arza, 2012), and Welsh (Evas, 2013). The reports mostly assessed the status of those languages in terms of language technology support. A general survey covering all regional and minority languages of the EU, the different types of digital media and services available, as well as inquiring about the *attitudes* and *desires* to make a digital use of the language is still lacking. The DLDP effort can therefore be seen as a first step towards the design of a survey about digital use of minority languages in both professional and informal contexts, specifically tailored on RMLs in the digital world and structured around a crucial question: is it possible for regional or minority language speakers to have a digital life in those languages? The paper is organised as follows: a description of the methodology underlying the design of the survey; an analysis of the results collected, with a separate section for each language; a summary of the key findings and an indication of the work planned for the future.

[1] http://www.dldp.eu

2. Survey Design and Methodology

The focus of the DLDP survey is on four European minority languages at different stages of digital development (Basque, Breton, Karelian and Sardinian). This allows us to explore and compare: (a) the behavior, perception and actual desires of different speakers' communities relative to the digital online use of their native languages; (b) the extent of the availability of digital content and language technology in 'digitally-different' languages; and (c) speakers' awareness of the latter.
All this has the final objective of devising and suggesting ad-hoc strategies for promoting the active digital usage of these languages and the development of language-based digital applications, which will substantially help in the preservation or revitalization of RMLs.

The questionnaire was developed around a set of topics, among which:
RML knowledge: perceived degree of fluency in the language inquired and values attributed to the language, whether mainly affective, identitarian, or instrumental;
Activism: whether the respondent qualifies as a language activist[2];
Extent and frequency of use of the language: is the language mainly used in oral or written form? In informal contexts only or in institutional ones as well? How often is the language used in those contexts?
E-communication: is the language ever used for writing e-mails, texting, chatting or other instant messaging? If yes, how often? If not, why?
Digital use: is the language ever used for surfing the Net, reading, writing/ commenting blogs, forums or websites? If yes, how? Only passively or actively as well? If not, why?
Technological support: if a specific keyboard is needed to type in the language, is it available?
Digital media: are digital media such as websites, blogs, Internet TV, audio and video streaming, e-books, etc. available in the language?
Wikipedia: is a Wikipedia available in the language and if yes, is it read and/or edited?
Social media: how often is the language used on social media (Facebook, Twitter, Instagram etc.)?
Localised software and interfaces: are operating systems, software and social media interfaces localised in the language? If yes, is the localised version used?
Language resources: are language resources and tools (such as online dictionaries, spell checkers, machine translation interfaces) available for the language?

This list of topics was further developed into a set of questions that were evaluated and discussed with the DLDP Advisors, distinguished scholars in the field of digital language revitalization, digital language activists, NLP professionals and policy makers[3]. DLDP Advisory Board brings together the most notable and/or active personalities in the field, with the twofold goal of getting advice and suggestions on the activities and goals of the project and also of enlarging the dissemination possibilities of the project outcomes and message.

2.1. Methodology

A basic questionnaire was developed in English and was then translated into Basque, Breton, Karelian and Sardinian, and made available as a Google form.[4]

The survey was carried out between July and September 2016, exclusively online. Informants were mostly recruited via the language associations involved in the projects and institutional contacts. The survey was also advertised on social media and through personal contacts. The profile of the typical respondent thus belongs to a person who makes at least a minimal digital use of the language in his/her everyday life, a fact which has to be taken into account when reading the results.
The addressees of the survey were individuals who reported to be speakers of one of the four case-study languages of the project and who are digitally literate.
As our main purpose was to understand the extent to which a language was used on the Internet and over a number of digital media and services, we were satisfied with digitally literate speakers only, as the reasons for not using the language digitally for those with no/low digital skills are evident.
We received a total of 1.301 replies, 749 from men, and 538 from women (breakdowns by age groups and sex are shown in Table 1). We are aware of the fact that the population inquired does not represent a balanced sample, which would have required the availability of data about the composition of a minority language population in terms of age, gender and other demographic variables.
Finally, the responses were normalised and analysed with the help of native speakers of each target language involved in the project.
In the next section, we describe the data collected for the four case studies of the DLDP project.

3. Survey Results

3.1. Basque: a digitally fit language asking for more opportunities

Basque is an isolate, non-Indo-European language spoken by about 900.000 speakers[5] living mainly in the Basque country and surrounding geographical areas situated in the northern part of Spain and the neighbouring south-western part of France. It has official status in the Spanish Basque Autonomous Community, but not in France.

[2]Language activists tend to be intentionally more assertive in their use of the language and, as a consequence, they can't represent average speakers.

[3]http://www.dldp.eu/en/content/advisors

[4]For Karelian, coherently with the co-existence of three recognised varieties and considering the possibility that speakers of one might not be able or willing to participate if the text was written in another variety, the questionnaire was translated into Olonets (or Livvi Karelian), South Karelian and North Karelian. However, for the analysis and the presentation of the results, the responses have been normalised to North Karelian, the variety most supported by the DLDP partner, KKS.

[5]source: NPLD.eu

Age Group	Gender	Basque	Breton	Karelian	Sardinian
<20	Female	9	9	0	10
	Male	11	5	1	6
	N/A	1	0	0	0
	All	*21*	*14*	*1*	*16*
20-29	Female	37	17	5	24
	Male	35	18	6	49
	N/A	2	1	1	1
	All	*74*	*36*	*12*	*74*
30-39	Female	59	21	6	38
	Male	67	23	3	67
	N/A	1	1	1	1
	All	*127*	*45*	*10*	*106*
40-49	Female	54	13	17	46
	Male	55	26	12	72
	N/A	0	1	0	1
	All	*109*	*40*	*29*	*119*
50-59	Female	36	10	16	44
	Male	49	29	15	65
	N/A	1	0	0	0
	All	*86*	*39*	*31*	*109*
60-69	Female	2	5	34	15
	Male	7	19	24	54
	N/A	0	0	1	0
	All	9	*24*	*59*	*69*
>70	Female	0	2	3	6
	Male	1	2	11	17
	N/A	0	0	0	0
	All	*1*	*4*	*14*	*23*
All		*427*	*202*	*156*	*516*

Table 1: Number of respondents with breakdown by age groups and sex

From the responses to the questionnaire, Basque emerges as a language that is regularly and actively used for e-communication, on the Internet and on social media by 97.2% of respondents, in particular for chatting and other forms of instant messaging, but also for e-mail. This is in line with the political status of the language, officially used in public administration and schools.

Even if no particular technological barrier is reported to impede the use of the language and localised digital services and interfaces are available for the Basque language, more effort is needed to sustain the language at the professional and institutional level, and more entertainment services and products are required in Basque targeted to young people.

For example, despite of knowing about the existence of localised operating systems and interfaces, some of the respondents are not using Basque in their devices, applications or softwares. A third part of the respondents feel that using Spanish tools is easier also because the way of finding and installing software in Basque is not as easy as it is in other languages, and, as a result of that, the user has to do an extra effort.

Some of the respondents are claiming for a site, where all available resources in Basque are listed and ready to be downloaded by users. In addition to that, it has been mentioned a need of spreading the information about the existing services/interfaces in Basque, especially among the young people. A more detailed description of the results is given in (Hernaiz and Berger, 2017).

3.2. Breton: strong awareness of the importance of digital presence

Breton is a Celtic language of France spoken mainly in western Brittany by around 200.000 speakers[6], and the number is said to be decreasing. Hopefully this trend will change thanks to the growing inclusion of the language in educational contexts. However, at the time of writing, Breton still has neither official status nor specific protection in France.

Overall, the survey reveals a strong desire of Breton speakers to use their language digitally, and their awareness of the importance of Internet presence for its revitalization. Nevertheless, although almost all respondents say they use Breton on the Internet, the seem to do so mostly passively (e.g. for reading rather than for writing).

For e-communication instead active use is widespread, for e-mail in particular, where the language appears to be used regularly by 88.5% of respondents. The same holds for active use of Breton on social media, with Facebook being the most used network, perhaps thanks to the localised interface available.

[6]source: NPLD.eu

Most of the respondents are aware of the existence of a Wikipedia in Breton, with a 19% of them even contributing to it by editing existing articles or writing new ones (8%). While the digital basics are firmly in place, the relative lack (or lack of awareness) of advanced services, apps and localised software stands out. At the same time, the respondents manifest a strong desire in this direction. For instance, automatic translation is almost completely lacking except for the online translation of Breton to French offered by *Ofis ar Brezhoneg*[7]; Google Translate is not available for Breton, yet. Indeed, if popular apps and key software interfaces are not provided in Breton soon, unable to compete with French apps, the language will inevitably appear less appealing to the younger generations. A more detailed description of the survey results can be found in (Hicks, 2017).

3.3. Karelian: motivated speakers want to be visible online

Karelian, a Uralic language of the Finnic branch, is a non-territorial language spoken in Russia (the Republic of Karelia and Tver oblast) and Finland. Two main dialects are distinguished: Olonets Karelian (also called Livvi) and Karelian proper, the latter of which is divided further into North Karelian (also called Viena Karelian) and South Karelian. The latest estimate suggest that in Finland around 11,000 people can speak Karelian well to fluently, with another 20,000 have some knowledge of the language. The language has official status in both countries. The digital portrait of the Karelian language reveals that Karelian speakers have a high linguistic self-esteem and are willing to use their language online, a crucial issue for the revitalization of a non-territorial language. However, there are a lot of necessary online and digital resources missing, and speakers often lack information about those that are available. For example, the existence of online dictionaries and of a Karelian Wikipedia [8] could benefit many Karelian speakers and the community if only they were aware of the existence of such resources.

In terms of social networks the use of Karelian is very heavily restricted to Facebook that anyway has users from all demographic groups and it is very well suited for extensive written communication, making it easy to connect with Karelians from different parts of the country and even across the border. The situation is not likely to change considerably anytime soon, although the amount of people using Karelian on Facebook might increase and there might be an increased Karelian presence on Twitter and Instagram, if more young people get interested in the language and start using it and if the Karelian revitalization efforts can get a better visibility in Finnish political discussion.

Among the reasons people are not using Karelian online we find that they don't know how to change their keyboard settings for this purpose. This is an important point to communicate to the community, since if writing Karelian is perceived as something that is difficult, it can easily make people less eager of using it. The interested reader can find additional information in (Salonen, 2017).

3.4. Sardinian: a digital language without self-awareness

Sardinian is a Romance language spoken by more than one million speakers in the island of Sardinia, Italy. The language is officially recognised but despite this, its presence in education and media is very limited. The responses to the questionnaire about Sardinian show high consideration of the language by its native speakers and a strong desire to use it in their everyday digital life. However, we observe a strong imbalance between the use of the language on social media and for e-communication on the one hand, and on the other the availability or awareness of other types of media and services in Sardinian. The language appears to be frequently used online both actively - in particular for texting, chatting, interacting on social media and blogs - and passively - for surfing and reading the Internet -. As for social media, it appears to be particularly vital on Facebook, by large the most used network, which even has a localised interface available. Between 73% and 87% of the respondents claim to use Sardinian digitally everyday but mostly in informal, private-life contexts. This fact which shows that, despite the official recognition granted in 1998, the language is still relatively little used on public sites and in official, formal contexts.

On the side of Internet media and services, we observe a different situation. Speakers tend to be unaware or not to use media and services in Sardinian even when available. For instance, almost half of the respondents is not aware that a Wikipedia is available and active for Sardinian, while almost the other half is only a passive user of it and many prefer the Italian version because more informative. Also, while online newspapers and news are relatively widely available, as is entertainment and - thanks to a previous investment by the Region - some Public Administration services, more advanced media such as smartphone apps, Internet TV, audio and video streaming are still missing. An interesting observation is that, despite there are no technical obstacles for using Sardinian online (i.e. typing is possible using the standard keyboard), many respondents nevertheless lament a lack of competence in written Sardinian that prevents them to write it with the necessary confidence. A more detailed description of the survey results can be found in (Russo and Soria, 2017).

Overall, the survey results clearly show the importance of encouraging the speaker community about using their language online as much as possible. The existence of a considerable number of language resources such as dictionaries, spell checkers, and even an automatic translation system is a good sign of the potential for this language to become a fully digital language, provided that the speakers are supported and encouraged to overcome the psychological barriers that are yet holding them back from considering Sardinian as a language in its full rights.

4. Key findings

From the exploratory study carried out, it appears that the four languages investigated place themselves at very dif-

[7] http://www.fr.brezhoneg.bzh/42-traducteur-automatique.htm

[8] The Karelian wikipedia, now online, was not publicly accessible at the time of the survey.

ferent levels of digital use and usability. However, a few themes emerge consistently:

- Regional and minority language speakers have a strong desire to use their languages digitally, in all the sociolinguistic domains and for all the purposes where major languages are used
- Social media has a huge potential as a domain that drives language revitalisation, but this sociolinguistic space is still restricted
- There is a clear demand for increased regional language usage in the public domain
- Minoritised language speakers need to be supported and encouraged regarding their ability to use their languages digitally and for their importance as digital content creators
- The lack of structural support for these languages is a serious issue that needs to be addressed. The digital development of these languages is not sustainable when it has to rely on the work of a handful of activists and volunteers

Table 5. presents a synoptic overview of the main findings regarding use of the languages over the main media and services addressed by the survey. Besides giving an approximate idea of the range, amount and frequency of digital uses, the survey also opens a window about the reasons why a language is not used. These are briefly summarised in Table **??**. These obstacles and limitations fall into three main categories:

1. technological barriers, such as the unavailability of a specific keyboard or spell checkers that would ease the writing;
2. linguistic barriers: lack of competence in the written language is often seen as a main problem that restrains people from using their language in written form. For a language such as Sardinian, where a standard is available but not yet widespread, this is a strongly felt issue;
3. psychological barriers: fear of being misunderstood, teased or of looking offensive. This category also comprises

5. Further work

To the best of our knowledge, this survey is the first of this kind and in theory it is applicable to any language with minor adaptations. Since one of the aims of the DLDP is to keep collecting data about digital language diversity, we encourage the broad community to adopt, adapt and localise the survey to gather data about other languages, and we are willing to offer support in this direction. To this end we make the model for the survey available upon request under a CC-BY-4.0 license.

For those who are interested in the original, raw data of the survey, they are deposited in the ILC4CLARIN Repository and made available under a CC-BY 4.0 license (Soria et al., 2017)[9].

Finally, the DLDP team is currently finalising a Digital Language Vitality Scale with measurable indicators for practically assessing the level of digital development of any language, with the questionnaire being a possible source of information, especially when official data are inexistent or not available. In the near future these two instruments will be used to better assess the four languages and derive four complete case studies that will hopefully inspire other similar exercises for many other RMLs.

Acknowledgements

The DLDP is a project funded under the Erasmus+ programme (Grant Agreement no. 2015-1-IT02-KA204-015090). The opinions expressed reflect only the authors' view and the Erasmus+ National Agency and the Commission are not responsible for any use that may be made of the information contained.

We thank our project partners, Klara Ceberio Berger, Antton Gurrutxaga Hernaiz, Davyth Hicks and Tuomo Salonen for their precious contribution to the set up and dissemination of the survey campaign first and then for the help in the analysis of the data. We are also grateful to all our advisors who offered their comments and suggestions, which helped us greatly improve our study. Any liability for the content of this paper, is of course the authors' alone.

Bibliographical References

Evas, J. (2013). *Y Gymraeg yn yr Oes Ddigidol – The Welsh Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). META-NET. Available online at `http://www.meta-net.eu/whitepapers`.

GarcÃa-Mateo, C. and Arza, M. (2012). *O idioma galego na era dixital – The Galician Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

Hernaiz, A. G. and Berger, K. C. (2017). Basque – a digital language? Reports on Digital Language Diversity in Europe. Work carried out within the Digital Language Diversity Project (www.dldp.eu), EC Erasmus+ Programme (2015-1-IT02-KA204-015090).

HernÃ¡ez, I., Navas, E., Odriozola, I., Sarasola, K., Diaz de Ilarraza, A., Leturia, I., Diaz de Lezana, A., Oihartzabal, B., and Salaberria, J. (2012). *Euskara Aro Digitalean – The Basque Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

Hicks, D. (2017). Breton – a digital language? Reports on Digital Language Diversity in Europe. Digital Language Diversity Project (DLDP). Work carried out within the Digital Language Diversity Project (www.dldp.eu), EC Erasmus+ Programme (2015-1-IT02-KA204-015090).

[9] `http://hdl.handle.net/20.500.11752/ILC-77`

Dimension	Basque	Breton	Karelian	Sardinian
e-communication	95.8%	88.5%	56.4%	71.8%
Available media (As reported by at least 50% of respondents)	All but Internet TV	Websites, streaming audio, Internet TV, streaming video, blogs and forums	Websites	Websites, blogs and forums
Wikipedia use	81.2%	68.3%	-	43.7%
Social Media use (Active use by more that 50% of respondents)	Facebook, Twitter	- (Facebook use not reaching treshold)	- (Facebook use not reaching treshold)	Facebook
Available digital services (As reported by at least 50% of respondents)	All but e-commerce, advertising	Online newspapers, Entertainment, online news and search engines	Online newspapers, online news	Online news

Table 2: Overview of key findings about digital use

Moreno, A., Bel, N., Revilla, E., Garcia, E., and VallverdÃº, S. (2012). *La llengua catalana a l'era digital – The Catalan Language in the Digital Age*. META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer. Available online at `http://www.meta-net.eu/whitepapers`.

Russo, I. and Soria, C. (2017). Sardinian – a digital language? Reports on Digital Language Diversity in Europe. Digital Language Diversity Project (DLDP). Work carried out within the Digital Language Diversity Project (www.dldp.eu), EC Erasmus+ Programme (2015-1-IT02-KA204-015090).

Salonen, T. (2017). Karelian – a digital language? Reports on Digital Language Diversity in Europe. Digital Language Diversity Project (DLDP). Work carried out within the Digital Language Diversity Project (www.dldp.eu), EC Erasmus+ Programme (2015-1-IT02-KA204-015090).

Soria, C., Russo, I., Quochi, V., Hicks, D., Gurrutxaga, A., Sarhimaa, A., and Tuomisto, M. (2016). Fostering digital representation of eu regional and minority languages: the digital language diversity project. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'16)*, pages 3256–3260, Portoroz, Slovenia, May. European Language Resource Association (ELRA).

Hans Uszkoreit et al., editors. (2012). *METANET White Paper Series*. Springer.

Language Resource References

Claudia Soria and Valeria Quochi and Irene Russo and Anneli Sarhimaa and Eleonore Kruse and Davith Hicks and Tuomo Salonen and Antton Gurrutxaga Hernaiz and Klara Ceberio Berger. (2017). *Digital Language Diversity Project Survey Data*. Istituto di Linguistica Computazionale "A. Zampolli" - Consiglio Nazionale delle Ricerche (ILC-CNR), distributed via CLARIN, 1.0, ISLRN http://hdl.handle.net/20.500.11752/ILC-77.

ASR for Documenting Acutely Under-Resourced Indigenous Languages

Robbie Jimerson, Emily Prud'hommeaux
Rochester Institute of Technology
Rochester, New York, USA
rcj2772,emilypx@rit.edu

Abstract

Despite its potential utility for facilitating the transcription of speech recordings, automatic speech recognition (ASR) has not been widely explored as a tool for documenting endangered languages. One obstacle to adopting ASR for this purpose is that the amount of data needed to build a reliable ASR system far exceeds what would typically be available in an endangered language. Languages with highly complex morphology present further data sparsity challenges. In this paper, we present a working ASR system for Seneca, an endangered indigenous language of North America, as a case study for the development of ASR for acutely low-resource languages in need of linguistic documentation. We explore methods of leveraging linguistic knowledge to improve the ASR language models for a polysynthetic language with few high-quality audio and text resources, and we propose a tool for using ASR output to bootstrap new data to iteratively improve the acoustic model. This work serves as a proof-of-concept for speech researchers interested helping field linguists and indigenous language community members engaged in the documentation and revitalization of endangered languages.

Keywords: low-resource languages, automatic speech recognition, indigenous languages

1. Introduction

By the end of this century, it is estimated that at least half and as many as 90% of the world's nearly seven thousand languages will be extinct (Krauss, 1992; Crystal, 2000). With each language that is lost, we lose insight not only into the culture of the the people who spoke that language, but also into the characteristics of that language that can shed light on the underlying structure of human language. Some communities on the verge of losing their language are engaged in preservation efforts, and many linguists carry out field work with native speakers to document endangered languages. Automatic speech recognition (ASR) has the potential to serve as a useful tool in these preservation and documentation efforts, but building models for these languages presents numerous challenges.

One particular challenge is a dearth of data, specifically, transcribed and labeled audio data to train the acoustic model and large amounts of text to train the language model. Languages that lack extensive data are known as *under-resourced* or *low-resource languages*, and all but a handful of the world's languages fall into this category. In fact, the set of languages considered to be low-resource for the purposes of ASR research includes many very widely spoken languages, including Bengali (spoken natively by 200 million people) and Vietnamese (with more native speakers than French) (Harper, 2014). Although researchers do not currently have access to large amounts of labeled data in these languages, it would be relatively easy to acquire more data with small investments of time and money to train speakers of the language to collect and label data.

Languages like Vietnamese and Bengali, which for political and economic reasons happen to have few ASR resources, stand in contrast to what we will call *acutely under-resourced languages*. Acutely under-resourced languages are typically spoken by very few people, are rarely written down, and may even lack a standardized writing system. Speakers of these languages, who sometimes live in remote parts of the world, might be reluctant to share their knowledge with outsiders or even to acknowledge that they speak the language. Linguists and community members routinely work to preserve and document these languages, often with the financial support of the government or non-governmental organizations. To date, however, there has been limited research performed in developing ASR systems for these languages, despite the potential benefit it would provide for language documentation and preservation.

In this paper, we present a case study in developing an ASR system for an acutely under-resourced language by focusing on Seneca, an endangered indigenous language of North America. We first provide an overview of the language and the obstacles to developing a robust ASR system for the language, given not only the poverty of the existing resources but also the unusually complex and productive morphology of the language. We propose approaches for leveraging linguistic knowledge and existing resources to increase the accuracy of our recognizer, as well as a tool for iteratively improving ASR performance and for optimizing the utility of ASR output for stakeholders engaged in transcription of audio data for language documentation purposes. A subset of the data we explore will be made available to other groups interested in developing ASR systems for low-resource polysynthetic languages. Our results demonstrate the potential for applying ASR to streamline and enhance the important task of documenting and preserving acutely under-resourced and endangered languages.

2. Background

2.1. Language documentation

Language documentation is the subfield of linguistics focused on producing a permanent and complete record of a language, which should include not only information about the grammar and lexicon but also labeled and annotated audio and textual data illustrating the information

contained in the grammar and lexicon and exemplifying how the language is used in everyday life (Himmelmann, 2006; Austin, 2014). Although social anthropologists have long collected language data from the communities they study, it is only in the last half-century that theoretical linguists have focused their efforts on systematically documenting languages in this way.

The importance of documentation for the communities of speakers of endangered languages is clear. Documenting an endangered language is one way to preserve important parts of a culture. Efforts by speech communities to maintain and revitalize extinct or endangered languages have also benefited from comprehensive language documentation. Theoretical linguists, while contributing to support for speech communities, are also motivated by the desire to find cross-linguistic evidence of language phenomena that can support or refute theoretical frameworks, with the goal of providing insight into the cognitive underpinnings of language.

The challenge in language documentation is that generating detailed transcriptions of recorded speech data and annotations of those transcriptions requires time, linguistic expertise, and technical knowledge. With this in mind, the primary goal of our work on developing ASR systems for acutely under-resourced languages is to provide an efficient and useful mechanism for helping linguists and community members working on endangered language documentation to produce complete and accurate annotated transcriptions of naturalistic, spontaneous speech data.

2.2. ASR for low-resource languages

The last several years have seen a surge in interest in developing robust ASR systems for low-resource languages (Besacier et al., 2014), fueled in part by the U.S. Intelligence Advanced Research Projects Activity (IARPA) Babel initiative (Harper, 2014). The IARPA Babel datasets consist of roughly 10 hours of transcribed speech for a number of relatively widely-spoken but low-resource languages, including Cantonese, Bengali, Turkish, Zulu, and Haitian Creole. The majority of the recent research on ASR for these languages has focused on optimizing the acoustic model in order to overcome the constraints imposed by having a limited amount of labeled audio training data. Researchers have explored modifications in approaches used to train the acoustic models (Grézl et al., 2014; Miao et al., 2013; Thomas et al., 2013); improvements in the features included in the models (Cui et al., 2014; Gales et al., 2014; Ghahremani et al., 2014; Tüske et al., 2014; Prabhavalkar et al., 2013); and supplementing the acoustic training data with data from other languages (Thomas et al., 2013; Gales et al., 2014; Grézl et al., 2014; Imseng et al., 2014; Tüske et al., 2014).

ASR has the potential to serve as a useful tool in language preservation and documentation efforts. To date, however, there has been little interest in building full ASR systems specifically for endangered language documentation. Much of the recent work specifically on developing ASR for low-resource languages has focused on tasks such as forced alignment of phonemes given manually generated transcriptions (DiCanio et al., 2012; DiCanio et al., 2013; Vetter et al., 2016), keyword spotting or spoken term detection (Prabhavalkar et al., 2013; Rosenberg et al., 2017; Metze et al., 2015), or pure phonetic transcription without word-level or utterance level information (Kong et al., 2016; Liu et al., 2016; Das et al., 2016; Hasegawa-Johnson et al., 2017).

The SPICE project's Rapid Language Acquisition Tool (Schultz et al., 2007), while offering promise as a means to collect data and exploit multilingual resources for building language technology systems for under-served languages, is geared toward languages with numerous speakers and large amounts of digitally available text data. In addition, most of the work stemming from this project has focused on TTS systems, rather than the development of ASR systems (Schultz et al., 2013; Schlippe et al., 2014). A more recent large-scale effort to develop language technologies for low-resource languages is the BULB project (Adda et al., 2016b; Adda et al., 2016a), which shares our goal of documenting endangered languages. The focus of the BULB project, however, is the development of a tablet-based interface for recording and transcribing languages lacking an established writing system. Smaller, language-specific efforts include the work of Mitra et al. (2016), who investigated using ASR for the documentation of Yoloxóchitl Mixtec, an endangered language with relatively abundant labeled audio data (125 hours). Although this work also included the use of hand-corrected ASR output to improve existing acoustic models, the authors did not incorporate synthetic data to improve either the acoustic or language models. We refer the reader to the excellent survey by Besacier et al. (2014) for a more complete discussion of the history of ASR for under-resourced languages.

Our work stands in contrast to previous work on ASR for low-resource languages in several ways. First, unlike Turkish or Cantonese which have millions of native speakers, our language of interest, Seneca, in spoken natively by a handful of people and as a second language by only a few hundred more, many of whom are reluctant to allow their language to be recorded. Second, there is very little written data in Seneca available; we cannot simply crawl the web to collect additional training data for our language model in the way that researchers working on any of the IARPA Babel languages can. Third, very little previous effort has been directed at enhancing the language models, particularly on leveraging existing data and linguistic knowledge to produce synthetic text data to augment the language model training data. Finally, and perhaps most crucially, the objective of our work is not to develop a framework for quickly developing an ASR system for any arbitrary language with unknown linguistic properties; instead, our goal is to provide linguists and endangered language community members with data and tools for documenting a language whose linguistic properties are known by the stakeholders.

Although we will in future work investigate many of the methods for improving the acoustic model described in the literature on low-resource languages, the focus of the work presented here is on leveraging in-domain data and linguistic knowledge to improve the language model and to reduce the OOV rate, which is unusually high due to the extreme morphological complexity of the language. Using the output of our ASR system, we then generate data files that can be used by stakeholders to transcribe and annotate new audio data. This newly transcribed and annotated data can be used for documentation purposes and can be reincorporated into the ASR system as additional training data to improve the existing models.

3. Data

3.1. The Seneca language

Historically spoken primarily in the areas of North America now known as New York, Ontario, and Quebec, the Iroquoian language family includes Seneca, Cayuga, Onondaga, Oneida, Tuscarora, Mohawk, and Cherokee. All except Mohawk and Cherokee are considered severely endangered, and all are acutely under-resourced. Seneca, the language we will discuss, is spoken across three reservations in Western New York: the Cattaraugus, Allegany and Tonawanda Reservations. There are currently fewer than 50 native speakers of Seneca, most of them elderly, and a few hundred second-language speakers.

Iroquoian languages have polysynthetic morphological systems, in which words are composed of many morphemes. Unlike agglutinative languages such as Turkish or Hungarian, which are also highly inflected, polysynthetic languages often permit noun incorporation, a process by which fully inflected nouns can optionally be inserted between a verb and the morphemes that accompany that verb. As shown in Figure 3.1., the basic Iroquoian verb is made up of four morphemes: the prepronominal prefix indicating tense, the pronominal prefix indicating the subject, the verb root, and the aspect suffix. Every Iroquoian verb must have at least a pronominal prefix, verb root, and aspect suffix. Seneca has fifty-two possible pronominal prefixes (varying by person, number, gender, and other features), thirty prepronominal prefixes (including verb tense, case, and other grammatical features), and four aspect suffixes (including ongoing action, completed action, habitual action). Thus, for a given verb there can be as many as 4680 different forms – not including any potentially incorporated nouns – which stands in stark contrast to a morphologically poor language such as English, in which a regular verb can have up to only five possible forms. This very high degree of morphological complexity yields severe vocabulary sparsity problems.

3.2. ASR Training and Testing Data

The audio recordings used to the train the Seneca acoustic model consist of roughly 80 minutes of spontaneous, naturalistic speech produced by five adult speakers, three male and two female. All five are first-language Seneca speakers whose second language is English, and all five are middle-aged or elderly. Additional information about the acoustic training data is provided in Table 3.2.. Recordings were made over many years under a variety of conditions using various pieces of recording equipment, yielding a diverse set of audio data.

prepron. prefix	**pronom. prefix**	**verb root**	**aspect suffix**
ʌ	g	ɑdʌnod	ʌʔ
future	1sg	sing	punctual

'I will sing'

Figure 1: Morphological structure of the Seneca verb. [ʌgʌdeɪrɑnoʊdʌʔ], meaning *I will sing*.

Speaker A is from the Cattaraugus Seneca reservation located 30 miles south of Buffalo, NY. In his brief recording, he tells the story of his great-grandfather, who used to hunt bears without a gun. Speaker B is from the Allegany Seneca reservation located by Salamanca, NY. His brief recording is a description of his garden and the plants he usually includes in his garden each year. This data was recorded and transcribed by Wallace Chafe, an emeritus professor of linguistics at UC Santa Barbara.

Speaker C is also from the Cattaraugus Seneca reservation. In his brief recording, he discusses the habits of deer. Speaker D is from the Cold Spring portion of the Allegany Seneca reservation. Her audio data consists of 30 minutes of conversations in Seneca with the first author, a member of the Seneca nation who is a second-language speaker of Seneca. The topics in this recording are wide ranging and include the speaker's family and upbringing, various stories from her childhood, and current events. Speaker E is from the Allegany Seneca reservation. This data totals 41 minutes of conversations in which the speaker discusses with other Seneca speakers a wide range of topics, including personal narratives and Seneca culture and folklore. This data was recorded and transcribed by the first author.

In addition to the transcriptions of the audio data described above, we have access to two other sources of textual data for training the language model. The first is a collection of transcribed stories and narratives produced by a Seneca speaker from the Allegany Seneca reservation. The second source is the Seneca Topic Reference Guide, a pedagogically oriented resource created by various Seneca speakers from across both the Cattaraugus and Allegany reservations. The utterances in this document were designed to enable a learner to have a simple conversation with another speaker in a question-and-answer format.

The held-out audio data used to test the ASR system was produced by Speaker E and was 12 minutes in length, with 40 utterances and 672 words.

4. Methods

We use the Kaldi (Povey et al., 2011) toolkit to build and test our ASR models. The acoustic model was created

	Minutes	Words	Sentences
Speaker A	3	139	20
Speaker B	2	126	21
Speaker C	4	265	20
Speaker D	75	4375	474
Speaker E	60	6059	400
Total	**144**	**10964**	**1235**

Table 1: Breakdown of acoustic training data by speaker

	Sentences	Words	Types
Stories	572	3925	817
Topic Ref.	221	573	219
Total	**793**	**4498**	**1036**

Table 2: Additional language model data.

following the "Kaldi for Dummies" tutorial recipe, which uses the standard 13 dimensional cepstral mean-variance normalized MFCCs, plus their first and second derivatives, within a GMM framework. The recipe was extended to apply LDA transformation and Maximum Likelihood Linear Transform to the features. Other training techniques included boosted Maximum Mutual Information (bMMI) and Minimum Phone Error (MPE). Both bMMI and MPE were trained over 4 iterations and bMMI used a boost weight of 0.5.

As discussed above, our focus is on leveraging existing resources to improve the language model and to reduce what we expect to be a very high OOV rate, given the morphological complexity of the language. We will compare three ASR systems, each with a different lexicon and language model.

The baseline model was created using the transcriptions of the audio data used to train the acoustic model. In addition, a list of 1,992 words extracted from a Seneca-English dictionary (Chafe, 1967) and combined with other words from the transcriptions of the acoustic training data, resulting in a lexicon of 2156 words. The second language model was built using the data described above plus data described in Table 3.2., adding an additional 739 sentences and 4498 words to the training data for the language model and 329 new words to the lexicon.

The third model was built using all of the above data plus additional synthetic data created using a deterministic algorithm for generating a morphologically rich set of Seneca verb forms from verb roots given the phonological processes that apply across morpheme boundaries (Chafe, 2015). The most frequently occuring verb roots in the data used to train models 1 and 2, above, were identified. Each verb root was then processed by the algorithm to generate multiple other common but unseen forms of that verb. In all, about 5000 verb forms, synthetically generated in this way, were added to the lexicon. An overview of all three models is shown in Table 4.

	Sentences in Corpus	Words in Lexicon
LM 1	778	2156
LM 2	1571	2485
LM 3	1571	7549

Table 3: Number of utterances and words in each of the three language models.

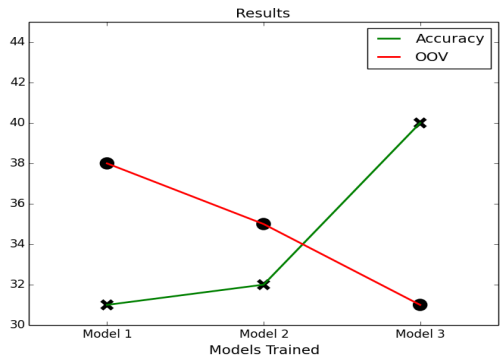


Figure 2: OOV rates and accuracy of three ASR models.

5. Results

Figure 2 plots the ASR accuracy against the OOV rate for the three models. As expected, given Seneca's complex morphology and the small amount of available language model training data, the OOV rate for all three models is very high but decreases with each addition of data to the language model, from 38% to 35% to 31%. The largest reduction in the OOV rate came from the introduction of the synthetic data, which included only the most frequent verbs in the original text data. We anticipate further reductions in the number of OOVs with more extensive use of the algorithm to generate more possible verb forms.

Despite the large number of OOVs, the recognizer performs adequately given the small amount of training data, with WER decreasing from 69% to 68% to 65% with each addition of data to the language model. The accuracy of our systems compare favorably with that reported by research groups working on low-resource languages with much simpler morphology (e.g., 60-70% WER on four of the IARPA Babel languages in Cui et al. (2014)) or in artificially low-resource scenarios (e.g., 60% WER in Thomas et al. (2013)).

Recall that one goal of our work is to provide a tool that linguists and endangered language speakers can use to more efficiently transcribe and annotate recorded language data. To that end, we have created a tool that works in conjunction with Kaldi to speed the process of labeling new training data. Using Kaldi's online wav decoder (online-wav-gmm-decode-faster), an unlabeled .wav file can be decoded using one of the trained Seneca models. The decoder produces a text file with the beginning and end timestamps of the spoken Seneca utterances. We convert this file using custom tools to a TextGrid file containing the aligned Seneca utterances. Using Praat (Boersma, 2001), a linguist or speaker

of the language can, with relatively little training, quickly review words and utterances, listen to the associated audio, easily correct the transcription produced by the ASR system, and adjust the boundaries between words and utterances. The corrected transcripts and annotations can then be saved out to simple text files for use by other linguists and community members. In addition, as more audio data is collected, the audio along with corrected ASR transcripts and timestamps can be incorporated into the acoustic model, resulting in improved ASR performance.

6. Conclusions and future work

In this paper, we used the Iroquoian language, Seneca, as a case study for exploring how to develop an ASR system for an acutely under-resourced and endangered language, with the goal of creating a tool for facilitating language documentation and preservation. Our methods, which included generating synthetic linguistically-informed data in order to lower the OOV rate and improve the language model, demonstrate the feasibility of this project. A subset of the data will be made available to other researchers interested in developing robust ASR systems for under-resourced highly inflected languages.

Our future work will concentrate on exploring two avenues to further reduce the word error rate of our recognizer. We will first apply methods similar to those described in the literature to build more robust acoustic models using DNNs. We are particularly interested in adapting our acoustic model training to include data from Mohawk and Oneida, two Iroquoian languages with very similar phonetic inventories but much more substantial audio resources. In addition, we plan to continue our research using automated morphological parsing tools, such as Morfessor (Smit et al., 2014), to reduce the OOV rate in our data. Our preliminary work using these tools has been disappointing, with very low morphological parsing accuracy, but we anticipate that training the supervised version of the parser with sufficient synthetic verb forms will result in meaningful accuracy improvements.

With the continued rise of globalization and the corresponding decreasing isolation of many indigenous communities, the need to document endangered languages grows more urgent. Automatic speech recognition and other computational linguistic technologies have the potential to transform the way linguists and community members preserve and revitalize their languages, and in turn, the culture they encompass and the insight into human cognition that they provide.

7. References

Adda, G., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H. E., Gauthier, E., Godard, P., Hamlaoui, F., Idiatov, D., Kouaratab, G.-N., Lamela, L., Makassoe, E.-M., Riallandb, A., Stukerf, S., Van de Veldec, M., Yvona, F., and Zerbiang, S. (2016a). Innovative technologies for under-resourced language documentation: The bulb project. In *Workshop CCURL 2016-Collaboration and Computing for Under-Resourced Languages-LREC*.

Adda, G., StÃ$\frac{1}{4}$ker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P., Hamlaoui, F., Idiatov, D., Kouarata, G.-N., Lamel, L., Makasso, E.-M., Rialland, A., de Velde, M. V., Yvon, F., and Zerbian, S. (2016b). Breaking the unwritten language barrier: The BULB project. In *Proceedings of the SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages*, pages 8–14.

Austin, P. K. (2014). Language documentation in the 21st century. *JournaLIPP*, 3:57–71.

Besacier, L., Barnard, E., Karpov, A., and Schultz, T. (2014). Automatic speech recognition for under-resourced languages: A survey. *Speech Communication*, 56:85–100.

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot International*, 5(9-10):341–45.

Chafe, W. L. (1967). *Seneca morphology and dictionary*, volume 4. Smithsonian Press.

Chafe, W. (2015). *A Grammar of the Seneca Language*. Univ of California Press.

Crystal, D. (2000). *Language death*. Ernst Klett Sprachen.

Cui, X., Kingsbury, B., Cui, J., Ramabhadran, B., Rosenberg, A., Rasooli, M. S., Rambow, O., Habash, N., and Goel, V. (2014). Improving deep neural network acoustic modeling for audio corpus indexing under the IARPA Babel program. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Das, A., Jyothi, P., and Hasegawa-Johnson, M. (2016). Automatic speech recognition using probabilistic transcriptions in swahili, amharic, and dinka. In *INTERSPEECH*, pages 3524–3528.

DiCanio, C., Nam, H., Whalen, D. H., Bunnell, H. T., Amith, J. D., and Castillo Garcia, R. (2012). Assessing agreement level between forced alignment models with data from endangered language documentation corpora. In *INTERSPEECH*.

DiCanio, C., Nam, H., Whalen, D. H., Timothy Bunnell, H., Amith, J. D., and García, R. C. (2013). Using automatic alignment to analyze endangered language data: Testing the viability of untrained alignment. *The Journal of the Acoustical Society of America*, 134(3):2235–2246.

Gales, M. J., Knill, K. M., Ragni, A., and Rath, S. P. (2014). Speech recognition and keyword spotting for low-resource languages: Babel project research at CUED. In *SLTU*, pages 16–23.

Ghahremani, P., BabaAli, B., Povey, D., Riedhammer, K., Trmal, J., and Khudanpur, S. (2014). A pitch extraction algorithm tuned for automatic speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 2494–2498. IEEE.

Grézl, F., Karafiát, M., and Vesely, K. (2014). Adaptation of multilingual stacked bottle-neck neural network structure for new language. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 7654–7658. IEEE.

Harper, M. (2014). IARPA Babel program. *https://www.iarpa.gov/index.php/research-programs/babel*.

Hasegawa-Johnson, M. A., Jyothi, P., McCloy, D., Mirbagheri, M., di Liberto, G., Das, A., Ekin, B., Liu, C., Manohar, V., Tang, H., et al. (2017). Asr for under-resourced languages from probabilistic transcription. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 25(1):50–63.

Himmelmann, N. P. (2006). Language documentation: What is it and what is it good for. *Essentials of language documentation*, 178(1).

Imseng, D., Motlicek, P., Bourlard, H., and Garner, P. N. (2014). Using out-of-language data to improve an under-resourced speech recognizer. *Speech communication*, 56:142–151.

Kong, X., Jyothi, P., and Hasegawa-Johnson, M. (2016). Performance improvement of probabilistic transcriptions with language-specific constraints. *Procedia Computer Science*, 81:30–36.

Krauss, M. (1992). The world's languages in crisis. *Language*, 68(1):4–10.

Liu, C., Jyothi, P., Tang, H., Manohar, V., Sloan, R., Kekona, T., Hasegawa-Johnson, M., and Khudanpur, S. (2016). Adapting asr for under-resourced languages using mismatched transcriptions. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5840–5844. IEEE.

Metze, F., Gandhe, A., Miao, Y., Sheikh, Z., Wang, Y., Xu, D., Zhang, H., Kim, J., Lane, I., Lee, W. K., et al. (2015). Semi-supervised training in low-resource asr and kws. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4699–4703. IEEE.

Miao, Y., Metze, F., and Rawat, S. (2013). Deep maxout networks for low-resource speech recognition. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 398–403. IEEE.

Mitra, V., Kathol, A., Amith, J. D., and García, R. C. (2016). Automatic speech transcription for low-resource languages-the case of yoloxóchitl mixtec (mexico). In *INTERSPEECH*, pages 3076–3080.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society.

Prabhavalkar, R., Livescu, K., Fosler-Lussier, E., and Keshet, J. (2013). Discriminative articulatory models for spoken term detection in low-resource conversational settings. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8287–8291. IEEE.

Rosenberg, A., Audhkhasi, K., Sethy, A., Ramabhadran, B., and Picheny, M. (2017). End-to-end speech recognition and keyword search on low-resource languages. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 5280–5284. IEEE.

Schlippe, T., Ochs, S., and Schultz, T. (2014). Web-based tools and methods for rapid pronunciation dictionary creation. *Speech Communication*, 56:101–118.

Schultz, T., Black, A. W., Badaskar, S., Hornyak, M., and Kominek, J. (2007). Spice: Web-based tools for rapid language adaptation in speech processing systems. In *Eighth Annual Conference of the International Speech Communication Association*.

Schultz, T., Vu, N. T., and Schlippe, T. (2013). Globalphone: A multilingual text & speech database in 20 languages. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8126–8130. IEEE.

Smit, P., Virpioja, S., Grönroos, S.-A., Kurimo, M., et al. (2014). Morfessor 2.0: Toolkit for statistical morphological segmentation. In *The 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Gothenburg, Sweden, April 26-30, 2014*. Aalto University.

Thomas, S., Seltzer, M. L., Church, K., and Hermansky, H. (2013). Deep neural network features and semi-supervised training for low resource speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6704–6708. IEEE.

Tüske, Z., Golik, P., Nolden, D., Schlüter, R., and Ney, H. (2014). Data augmentation, feature combination, and multilingual neural networks to improve asr and kws performance for low-resource languages. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Vetter, M., Müller, M., Hamlaoui, F., Neubig, G., Nakamura, S., Stüker, S., and Waibel, A. (2016). Unsupervised phoneme segmentation of previously unseen languages. In *INTERSPEECH*, pages 3544–3548.

Building a Sentiment Corpus of Tweets in Brazilian Portuguese

Henrico Bertini Brum, Maria das Graças Volpe Nunes
Interinstitutional Center for Computational Linguistics (NILC)
Institute of Mathematical and Computer Sciences, University of São Paulo
henrico.brum@usp.br, gracan@icmc.usp.br

Abstract

The large amount of data available in social media, forums and websites motivates researches in several areas of Natural Language Processing, such as sentiment analysis. The popularity of the area due to its subjective and semantic characteristics motivates research on novel methods and approaches for classification. Hence, there is a high demand for datasets on different domains and different languages. This paper introduces TweetSentBR, a sentiment corpus for Brazilian Portuguese manually annotated with 15.000 sentences on TV show domain. The sentences were labeled in three classes (positive, neutral and negative) by seven annotators, following literature guidelines for ensuring reliability on the annotation. We also ran baseline experiments on polarity classification using six machine learning classifiers, reaching 80.38% on F-Measure in binary classification and 64.87% when including the neutral class. We also performed experiments in similar datasets for polarity classification task in comparison to this corpus.

Keywords: Sentiment Analysis, Corpus Annotation, Social Media

1. Introduction

Sentiment Analysis (SA) became a popular area of Natural Language Processing in the last decade. The classification of semantic orientation of documents is a challenge for artificial intelligence methods since it is based not only on the regular meaning of words, but also on their semantic role in the context and on the author's intention. Furthermore, the amount of data available in blogs, social media posts and forums has created a great opportunity for researchers to build datasets for evaluating methods and studying new linguistic phenomena.

Websites on e-commerce, movie reviews and hotel reservations usually allow the user to provide an objective evaluation besides the written commentaries. This objective evaluation (binary recommendation, star score, 10-point scale) can be a good feature for automatic labeling large datasets on semantic orientation, thus improving the resources for researches over the past decades (Pang et al., 2002; Pang and Lee, 2005; Blitzer et al., 2007).

The limitation of this technique is the data available in this conditions. Social media, for example, is a large source of user opinions and evaluation (Pak and Paroubek, 2010), but the lack of an objective score attached to the posts demands a manual annotation in order to data become useful for SA, even though the data is enriched by linguistic phenomena such as expressions, slangs and irony.

Manual annotation ends up being more expensive and time consuming, since it demands several guarantees of accuracy, such as developing guidelines, training annotators and revising the annotation (Hovy and Lavid, 2010).

In this paper we introduce TweetSentBR (TTsBR), a corpus manually annotated with data extracted from Twitter. The section 2. presents some related work on SA and corpus annotation. Section 3. presents the corpus and its properties, such as the size, the annotation tags, the information on annotators and the process of data extraction. Section 4. presents data analysis and polarity classification experiments on the corpus. Section 5. is a brief discussion on the importance of the corpus and how it can be used in Brazilian Portuguese research on SA.

2. Related Work

Several works present new methods and approaches for tasks such as polarity classification (Turney, 2002; Pang and Lee, 2005), detection of irony (Carvalho et al., 2009; Reyes et al., 2012) and aspect extraction in text (Hu and Liu, 2004).

One of the major issues of this area is the building of datasets for evaluating methods and for training machine learning models. Turney (2002), one of the first works on polarity classification, used product reviews labeled as "recommended" and "not recommended". The source of the data was a website called Epinions, where users could evaluate products and leave a five star score for each review. The authors considered any review with less than 3 stars as "not recommended". Pang et al. (2002) uses a similar score (star rating) in order to compile a corpus of movie reviews on three classes (positive, negative and neutral).

The automatic approach worked very well for building large datasets, but the method limited research on domains where users input an objective score. Despite of the challenges of the manual annotation, researches began building new datasets by training annotators to label the data. Socher et al. (2013) introduces Stanford Sentiment Treebank, a relabeling of the previous IMDB corpus presented in (Pang and Lee, 2005). SemEval, an important semantic evaluation event, also produces several datasets for English designed for SA tasks (Nakov et al., 2016). Some authors even used distant supervision techniques for automatic labeling large datasets quickly using features such as *emoticons* (Go et al., 2009).

In Brazilian Portuguese, several works presented corpora for SA. Freitas et al. (2012) introduce ReLi, a sentiment corpus of book reviews manually annotated in three classes (positive, neutral and negative). The authors have chosen books from different genres in order to vary the linguistic phenomena in the corpus (from teenage books to literature classics). ReLi contains annotation of semantic ori-

entation, part-of-speech tagging and aspect of opinion, and it was later used as resource for researches in SA (Balage et al., 2013; Brum et al., 2016). One of the issues on this corpus observed on the literature is the unbalanced classes - the majority of sentences is neutral (72%), while the negative class represents only 4% of the data.

On the product review domain, Hartmann et al. (2014) presented Buscape corpus, a large corpora in Brazilian Portuguese. The corpus contains 13.685 reviews labeled as positive and negative, using scores given by users on Buscape, a popular e-commerce website. A similar dataset is Mercado Livre corpus, introduced in Avanço (2015), containing 43.818 product reviews also labeled automatically and balanced between the two classes.

Silva et al. (2011) collected a corpus from Twitter in Portuguese. The dataset was collected by searching two entities in the social network (Dilma and Serra, two running candidates at the time) and manually annotated as positive or negative. The corpus contains 76.358 documents balanced between positive and negative. The corpus was originally constructed for sentiment stream analysis meaning it contains several retweets and links, phenomena that may interfere on sentiment classification but is vital to maintain the stream for the former task.

Also on binary polarity classification, Moraes et al. (2015) introduce the Corpus 7x1, a brazilian portuguese corpus on Twitter comments during the 2014 World Cup semi-finals. The corpus presents some interesting user behavior such as irony, sarcasm, cheering and angry due to the final match score. Corpus 7x1 contains 2.728 tweets labeled manually in three classes - the neutral class represents tweets that do not align with either positive or negative sentiments.

Moraes et al. (2016) also uses Twitter as the source of data, but compile a corpus of computer products containing 2.317 tweets. The data is manually labeled in three classes and the authors also performed experiments on SA using lexical-based classifiers and SVM.

A large Twitter corpus was compiled by (Correa Junior et al., 2017) using distant supervision. The authors labeled tweets in Brazilian Portuguese using emojis representing positive and negative sentiments following the work of Go et al. (2009) in English. The corpus contains 554.623 positive tweets and 425.444 negative. The approach is a fast way to label data, but the method can not guarantee the absence of noise data such as irony, sarcasm or incorrect labels.

3. TweetSentBR

TweetSentBr is composed of 15.000 tweets (17.166 tokens) extracted using Python-Twitter [1], a wrapper for Twitter API. Due to the limitations of Twitter API, we developed a continuous crawler in order to obtain documents during the first semester of 2017. The final dataset is split in two documents - a training set with 12.999 documents labeled in positive (44%), neutral (26%) and negative (29%); and a test set composed of 2001 documents with similar distribution to the training set, 45%, 25% and 29% respectively. See Table 1 for the number of documents in each class.

[1] https://github.com/bear/python-twitter

Class	Training set	Test set	Total
Positive	5.745 *(44%)*	903 (45%)	6.648
Neutral	3.414 *(26%)*	512 (25%)	3.926
Negative	3.840 *(29%)*	586 (29%)	4.426
Total	12.999	2.001	15.000

Table 1: Amount of documents in the corpus in each class.

3.1. Data source

Data was extracted from Twitter between January and July in 2017. We chose to focus on the TV show domain because of the large amount of user generated content on Twitter during the exhibition of the shows. Hashtags (#) are used on social media to group messages on topics and the TV shows usually ask for its audience to use a specific hashtag in order to get visibility in these social networks. Some of the program hashtags group hundreds of thousands of messages during the exhibition of a show and that content can represent suggestions, complaints, evaluations and questions to the entities related to the programs.

We empirically defined nine programs from three major TV channels in Brazil based on their popularity and activeness in social media. Talk-shows, reality shows (gastronomy and music) and variety shows were chosen in order to diversify the phenomena in the corpus. The periodicity of the exhibitions are also different, some shows go live daily when others go live once or twice a week.

Since we were looking for user generated content, we ignored documents generated by public entities, such as celebrities, companies, TV channels or any official user on Twitter. We also discarded retweets, which are the reposts of popular posts in the social network.

3.2. Classes definition

Following Hovy and Lavid (2010) recommendations, a codebook or manual was written to ensure the agreement between annotators. The codebook contains examples, definitions and tips for the annotation process. The rules and guidelines were formed by empirically observing the dataset crawled before the annotation update based on the feedback from the annotators during the early stages of annotation.

The definitions were created based on the domain and the input received by the annotators after the first contact with the data. These are the guidelines for the annotation in TTsBR:

Positive class: Positive sentences describe feelings of pleasure, satisfaction, compliment or recommendation. The target of the sentiment must be the TV show or any entity related to it (host, guests, audience, sketches, invited bands...). Positive comparisons, such as "*This show is better than the other*" are considered positive and emojis can be strong indicatives of positivity.

Negative class: Negative sentences describe feelings of disagreement, disapprove, complaint or hate. The target of the sentiment must be the TV show or any entity related to it (hosts, guests, audience, sketches, invited bands...). Neg-

ative sentences can be direct, as in "*Today's show is terrible...*" or implied in a suggestion, as in "*the host could improve its jokes, right?*". Factual information such as delays, abrupt cuts or technical failures are also considered negative as long as it refers for the show or any entity related to it. Emojis are also good indicatives of negativity.

Neutral class: The neutral label must be used for any sentence the annotator could not identify as an opinion (positive or negative) direct or implied. Factual sentences that do not represent a hit or a flaw, such as "*Show X just started*", inaccurate semantic orientation ("*Don't know what to think about this*") and sentences the annotator can not completely comprehend were instructed to be annotated as neutral. Some tweets in the corpus were generated by social media robots (most of them on audience measurements) and the annotators labeled these as neutral as well.

We also wanted to keep track of the sentences that most caused doubt in the annotators. The annotators had a check box to mark in case of doubt in the annotation, even though this option did not prevent the annotator from labeling the sentiment of the sentence. The annotators were instructed to mark the *doubt* option every time they felt divided between two or more classes or when they took more than the average time (2 minutes) in a sentence. The addition of a *doubt* option gives us new information on the data and also reduces the stress on annotators. In the first stages of annotation, only an average of 10% of sentences were marked.

3.3. Annotation process

For the annotation process we recruited seven native speakers of Brazilian Portuguese in three different areas - linguistics, journalism and computer science. The annotation process was based on Hovy and Lavid (2010), following the eight steps of annotation in order to improve the reliability of the resource.

We developed a user friendly interface for the annotators to label the tweets (Figure 1). The interface contains the codebook, the phases of annotation, a progress bar and a panel with tweets for labeling. The annotation panel shows to the annotator the three classes and a box to be checked when in doubt (even though every tweet must have a label chosen in order to proceed to the next phase) and a side box with quick tips, contact information and a link to the codebook.

Each annotator received a set with one hundred tweets to be labeled. After this step we measured the average time, agreement (all received the same set) and we took notes of the questions about the codebook guidelines. Then we proceeded to rewrite the codebook, adding more examples and detailing the definitions based on the questions presented on the first annotation.

The next step was a meeting with all annotators to receive the feedback of the annotators, when the tweets were revised by everyone and we presented the new version of the codebook.

We then proceeded to the regular annotation. First the participants labeled 300 tweets in order to measure agreement. We used Krippendorf's Alpha (Kripendorff, 2004) to measure the agreement of the annotators. In this phase we obtained 52.9% on the nominal measure and 70% on the interval measure. The annotators began the individual phases when each one labeled around 2.000 tweets in six weeks, completing the training portion of the corpus.

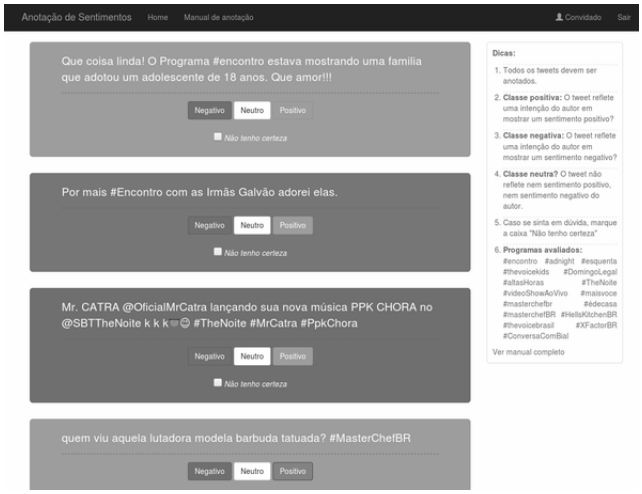


Figure 1: Snapshot of the annotation interface.

For the release, we define the general sentiment for each document based on a major voting of the labels provided by each annotator. Some documents were only labeled by one annotator, while others were annotated by 3 or 7 annotators. 45 documents tied and have no sentiment label, even though they were kept in the dataset with a "*none*" label.

3.4. Release and distribution

The dataset is available in `http://bitbucket.org/HBrum/tweetsentbr/`. Twitter has a Privacy Policy forbidding the redistribution of the data, so we managed to provide only the *ids* of the tweets in the corpus. Any user with a working identification can search for the tweets freely.

We provide the dataset with the *ids*, the hashtags used in the search, full annotators labels, the count of how many annotators checked the *doubt* option and the general sentiment for each document, as well as a tool for downloading the dataset as long as having a valid credential (provided by Twitter itself). An example of the dataset is presented below:

```
id        hashtag     labels  h  s split
-----------------------------------------
86304477 #encontro    [1,1,1] 0  1 train
86558371 #theNoite    [1,0,0] 2  0 test
86506323 #encontro    [1]     1  1 train
86466839 #masterChef [-1]    0 -1 test
```

3.5. Comparison with Brazilian datasets

Comparison between datasets is a general challenge in NLP since each task carries its own issues, needs and goals. Even in a specific field it can be hard to directly compare

Corpus	Domain	# pos	# neu	# neg	Balanced	# Documents	Annotation
Buscape	Product reviews	6.873	-	6.812	yes	13.685	Automatic
Mercado Livre	Product reviews	21.820	-	21.499	yes	43.818	Automatic
Pelesent	General (tweets)	554.623	-	425.444	no	980.067	Automatic
Elections - Dilma	Politics (tweets)	46.808	-	19.835	no	66.643	Manual
Elections - Serra	Politics (tweets)	1.371	-	8.347	no	9.718	Manual
Corpus 7x1	Sports (tweets)	449	1.101	1.178	no	2.728	Manual
Computer-BR	Products (tweets)	197	1.677	443	no	2.317	Manual
ReLi	Book reviews	2.883	8.991	596	no	12.470	Manual
TTsBR	**TV shows (tweets)**	**6.648**	**3.926**	**4.426**	**no**	**15.000**	**Manual**

Table 2: Amount of documents in the corpus in each class.

datasets and its attributes. Table 2 presents as overview of sentiment analysis corpora compared to TTsBR.

TTsBR stands as a novel domain corpora to be used in the polarity classification task. In tweet domain, it figures out as the second biggest corpora manually annotated for binary classification in Brazilian Portuguese and the biggest for 3-polarity classification.

The data distribution between classes can be a issue for machine learning methods. A common approach for handling unbalanced corpora is using under-sample (removing samples from the majority classes until the corpus is equally distributed). In this scenario, TTsBR only loses 21% of its size (11.768 documents) when ReLi, $7x1$ and Computer-BR end up with 1.788, 1.347 and 591 respectively.

When comparing automatic labeled methods, by score-based annotation (Buscape and Mercado Livre) or distant supervision (Pelesent), manual approaches suffer from time consumption, but also gain in reliability. The distant supervision approach, for example, performed in Pelesent demands the removal of emojis and emoticons from the final corpus. This information may be important for a linguistic approach of even semantic study of user behavior online.

We believe TTsBR can be helpful for evaluating new polarity classification approaches or linguistic studies, since it relies on a popular topic, is publicly available, was mostly revised by more than one annotator and the annotation methodology is documented and easily open for replication.

4. Experiments

In order to investigate the properties of the corpora, we defined a series of experiments to determine word frequency, the class balance, and we also performed polarity classification using baseline methods for Portuguese.

For the experiments, we performed a preprocessing of the data - we replaced numbers (dates, currency values) by a *NUMBER* token, we also replaced user names and links by the tokens *USERNAME* and *URL*, respectively. We trimmed repetition of characters (eg. "looooove" turns into "love") to a minimum of 3 repeat characters

4.1. Corpus statistics

In order to extract some information from the corpus, we measured the relevance of words on each class. We calculated the tf-idf value of each term ignoring hashtags, stopwords, emojis and punctuation. We chose to report only the polarity classes (positive and negative) since the neutral class groups several characteristics (facts, out-of-topic sentences, confusing content) that led the analysis to terms not expressive, such as the name of the shows, users nicknames and neutral verbs (present, watch,...).

	Positive class		Negative class	
#	PT-BR	EN	PT-BR	EN
1	amo	to love	ridículo	ridiculous
2	fofura	cute	péssimo	awful
3	adorando	loving	lixo	trash
4	emocionada	emotional	tirem	to remove
5	linda	beautiful	mala	boring

Table 3: Five most relevant terms according to tf-idf for each polarity class.

The terms indicated in Table 3 show a notable semantic orientation represented in the classes - the positive class shows the verb "to love" and positive adjectives, while the negative class shows adjectives (the word "trash" is popular used as adjective on Twitter) and the verb "to remove" that may indicate a request for removing a guest from a show or even a participant from a reality show.

4.2. Polarity classification task

In order to evaluate the corpus on the polarity classification task we used six machine learning classifiers - a linear SVM (C: 1), a Bernoulli Naive Bayes (alpha:0.1), Logistic Regression, a Multilayer Perceptron (2 layers, 200 neurons, learning-rate:00.1), a Decision Tree classifier and Random Forest approach with 200 estimators.

For data representation we used a bag-of-words with occurrence of terms, presence of negation words ("not", "never",...), positive and negative emoticons, emojis, presence of positive and negative words and PoS tags. We used the lexicons presented in Avanço et al. (2016) for negation words, positive/negative emoticons and words. For PoS tagging we used NLPnet tagger (Fonseca et al., 2015) and Emoji Sentiment Ranking (Novak et al., 2015) for emoji polarity probability.

Table 5 presents the results obtained by each classifier using the train/test evaluation scheme detailed in section 3.

Method	F-Pos	F-Neu	F-Neg	F-Measure
SVM	73.3	47.2	62.2	60.89
NB	70.4	48.2	58.8	59.12
LR	**76.6**	51.7	66.3	**64.87**
MLP	76.3	**51.9**	65.6	64.60
Dec. Tree	67.8	40.3	55.4	54.50
Rand. F	74.9	43.2	**68.4**	62.18

Table 4: Classification of TweetSentBR.

We can observe in Table 5 that neutral class achieves constantly the least F-1, it may be caused by the unbalanced distribution of classes in TTsBR. Other indication is the highest values in F-1 obtained in positive class, since it contains the majority of documents.

For further experiments we used only the classifier that achieved the best F-Measure in this experiment (Logistic Regression).

We compared the results obtained in polarity classification with other corpora in literature. For 3-class polarity classification we were unable to obtain $7x1$ and Computer-BR corpora, therefore we only compared TTsBR with ReLi (Freitas et al., 2012). In order to compare different datasets we used 10-fold cross instead of the train/test scheme.

Method	F-Pos	F-Neu	F-Neg	F-Meas.
ReLi	60.0	**87.7**	21.5	56.45
TTsBR	**74.2**	46.9	**63.9**	**61.68**

Table 5: TweetSentBR compared to ReLi using Logistic Regression.

The F-Measure of classes indicates how an unbalanced corpus can skew the classification. ReLi has 7% of documents labeled as negative and it affects directly the F1 of this class. The same occurs with TTsBR, but not so drastically, since the data is not as skewed as the former. Either way, TTsBR achieved better results when compared to ReLi.

We performed the same classification with the other Brazilian Portuguese datasets presented in subsection 3.5., but since the other corpora have only two labels (positive and negative) we limited TTsBR and ReLi to it's only two polarities. The results are shown in Table 6.

Method	F-Pos	F-Neg	F-Measure
Buscape	86.83	86.11	86.47
Mercado Livre	94.57	94.46	94.52
Elections-Dilma	94.13	85.45	89.78
Elections-Serra	88.71	98.22	93.46
Pelesent	92.04	44.51	68.27
ReLi	90.14	42.02	66.08
TTsBR	84.78	75.99	80.38

Table 6: TweetSentBR compared to Brazilian sentiment corpora in binary classification using LR classifier.

We can observe the impact of the neutral class in the classification. Compared to the results obtained in Table 5, the binary classification improved 10% in F-Measure for ReLi and almost 20% in TTsBR.

Since we are not looking for a better F-Measure more than for an overview of the behavior of each corpora on classification, the results obtained in each corpus have no need to be compared.

The highest values of F-Measure were obtained in Mercado Livre and in both Elections datasets. it may indicate the features used in classification form a good subspace for analysis in both domains (politics and product reviews). TTsBR still presents a gap in the positive and negative F-Measures obtained, but this behavior is present in every unbalanced corpora.

5. Discussion and future work

TweetSentBR is a manually annotated corpus designed for polarity classification. The corpus was formed using a novel domain for the Brazilian Portuguese language that can be exploited by new machine learning approaches such as deep learning architectures and ensembles.

It also offers new resources for linguistic approaches on natural language by observing the expressions, social media behavior or hate speech detection. The *doubt* label, for example, can be used for a better evaluation of classifiers by comparing machine learning flaws with human uncertainty on labeling data.

This corpus also differs from other approaches by including the neutral class. The addition of the neutral class approximates the corpora to popular applications, since the polarity classifiers available in the industry must find solutions for separating the opinions of users from noisy data. ReLi (Freitas et al., 2012) and Computer-BR (Moraes et al., 2016) are the only corpora we found in the literature that describes the use of a neutral class on the annotation.

It also motivates new research on features for better describing the neutral polarity space in classification, since the results achieved when removing the class are improved in almost 20% using Multilayer Perceptron with features used in the literature.

We believe this corpus can still be improved by labeling more data manually. This could improve classification by reducing the unbalance in class distribution, which could help classifiers to achieve better results.

We are currently working on a semi-supervised approach for automatically expanding the corpus based on self-training and co-training (da Silva et al., 2016). We beleieve this can improve the size of the corpus with few human effort and could also be applied for different domains and tasks in the future.

6. Acknowledgement

We acknowledge financial support from CNPq and CAPES as well as the help of Amanda Carneiro, Fernando Nóbrega, Juliana Batista, Rafael Anchiêta and Thales Bertaglia for the volunteer annotation task of TTsBR, as well as Marcos Treviso for the development of the annotation interface.

7. Bibliographical References

Avanço, L. V., Brum, H. B., and Nunes, M. (2016). Improving opinion classifiers by combining different

methods and resources. *XIII Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*, pages 25–36.

Avanço, L. V. (2015). Sobre normalização e classificação de polaridade de textos opinativos na web.

Balage, P. P., Pardo, T. A. S., and Aluısio, S. M. (2013). An evaluation of the brazilian portuguese liwc dictionary for sentiment analysis. *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology (STIL)*, pages 215–219.

Blitzer, J., Dredze, M., Pereira, F., et al. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *ACL*, 7:440–447.

Brum, H., Araujo, F., and Kepler, F. (2016). Sentiment analysis for brazilian portuguese over a skewed class corpora. *International Conference on Computational Processing of the Portuguese Language*, pages 134–138.

Carvalho, P., Sarmento, L., Silva, M. J., and De Oliveira, E. (2009). Clues for detecting irony in user-generated contents: oh...!! it's so easy;-. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 53–56. ACM.

Correa Junior, E. A., Marinho, V. Q., Santos, L. B. d., Bertaglia, T. F., Treviso, M. V., and Brum, H. B. (2017). Pelesent: Cross-domain polarity classification using distant supervision. *arXiv preprint arXiv:1707.02657*.

da Silva, N. F. F., Coletta, L. F., Hruschka, E. R., and Hruschka Jr, E. R. (2016). Using unsupervised information to improve semi-supervised tweet sentiment classification. *Information Sciences*, 355:348–365.

Fonseca, E. R., Rosa, J. L. G., and Aluísio, S. M. (2015). Evaluating word embeddings and a revised corpus for part-of-speech tagging in portuguese. *Journal of the Brazilian Computer Society*, 21(1):2.

Freitas, C., Motta, E., Milidiú, R., and César, J. (2012). Vampiro que brilha... rá! desafios na anotaçao de opiniao em um corpus de resenhas de livros. *Encontro de Lingística de corpus*, 11:3.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(2009):12.

Hartmann, N. S., Avanço, L. V., Balage Filho, P. P., Duran, M. S., Nunes, M. d. G. V., Pardo, T. A. S., and Aluisio, S. M. (2014). A large corpus of product reviews in portuguese: tackling out-of-vocabulary words. *9th International Conference on Language Resources and Evaluation*.

Hovy, E. and Lavid, J. (2010). Towards a 'science' of corpus annotation: a new methodological challenge for corpus linguistics. *International journal of translation*, 22(1):13–36.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Kripendorff, K. (2004). Reliability in content analysis: Some common misconceptions. *Human Communications Research*, 30:411–433.

Moraes, S. M. W., Manssour, I. H., and Silveira, M. S. (2015). 7x1pt: um corpus extraído do twitter para análise de sentimentos em língua portuguesa. *Proceedings of Symposium in Information and Human Language Technology*.

Moraes, S. M., Santos, A. L., Redecker, M. S., Machado, R. M., and Meneguzzi, F. R. (2016). Classificação de sentimentos em nível de sentença: uma abordagem de múltiplas camadas para em lingua portuguesa. *XIII Encontro Nacional de Inteligência Artificial e Computacional*.

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., and Stoyanov, V. (2016). Semeval-2016 task 4: Sentiment analysis in twitter. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*.

Novak, P. K., Smailović, J., Sluban, B., and Mozetič, I. (2015). Sentiment of emojis. *PloS one*, 10(12):e0144296.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Association for Computational Linguistics.

Reyes, A., Rosso, P., and Buscaldi, D. (2012). From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering*, 74:1–12.

Silva, I. S., Gomide, J., Veloso, A., Meira Jr, W., and Ferreira, R. (2011). Effective sentiment stream analysis with self-augmenting training and demand-driven projection. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 475–484. ACM.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Turney, P. D. (2002). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA. Association for Computational Linguistics.

‘Aye’ or ‘No’? Speech-level Sentiment Analysis of Hansard UK Parliamentary Debate Transcripts

Gavin Abercrombie and **Riza Batista-Navarro**
School of Computer Science
University of Manchester
Kilburn Building, Manchester M13 9PL
gavin.abercrombie@postgrad.manchester.ac.uk
riza.batista@manchester.ac.uk

Abstract

Transcripts of UK parliamentary debates provide access to the opinions of politicians towards many important topics, but due to the large quantity of textual data and the specialised language used, they are not straightforward for human readers to process. We apply opinion mining methods to these transcripts to classify the sentiment polarity of speakers as being either *positive* or *negative* towards the motions proposed in the debates. We compare classification performance on a novel corpus using both manually annotated sentiment labels and labels derived from the speakers’ votes (‘aye’ or ‘no’). We introduce a two-step classification model, and evaluate the performance of both one- and two-step models, as well as the use of a range of textual and contextual features. Results suggest that textual features are more indicative of manually annotated class labels. Conversely, in addition to boosting performance, contextual metadata features are particularly indicative of vote labels. Use of the two-step debate model results in performance gains and appears to capture some of the complexity of the debate format. Optimum performance on this data is achieved using all features to train a multi-layer neural network, indicating that such models may be most able to exploit the relationships between textual and contextual cues in parliamentary debate speeches.

Keywords: Hansard transcripts, parliamentary debates, sentiment analysis

1. Introduction

In the United Kingdom, transcripts of parliamentary debates (known as *Hansard*) are publicly and freely available. This provides access to a wealth of information concerning the opinions and attitudes of Members of Parliament (MPs) and their parties, towards arguably the most important topics facing society, as well as potential insights into the parliamentary democratic process. However, the large quantity of recorded material in Hansard, combined with the esoteric speaking style and opaque procedural language of Parliament, makes manual retrieval of information from these data a daunting task for the non-expert citizen.

Despite the fact that opinion mining has been one of the most active areas of research in natural language processing (NLP), and a widespread need for political information has been cited as a motivation for the development of opinion mining technologies (Pang and Lee, 2008), automatic analysis of the positions taken by speakers in parliamentary debates has received relatively little attention from researchers.

Sentiment anlysis is the task of automatically identifying the polarity (*positive* or *negative*) of the position taken by the holder of an opinion towards a *target*, such as an organization, a policy, a movement, or a product. We apply sentiment analysis methods to speeches made in the House of Commons of the UK Parliament to classify their sentiment polarity as being either *positive* (in support) or *negative* (in opposition) towards the target of each speech; that is, the *motion* proposed in the debate in question.

Prior work on this task has relied on the use of MPs’ *division votes* as sentiment polarity labels, under the assumption that these votes represent the speakers’ opinions towards the subjects under discussion: votes for ‘Aye’ (that the motion be approved) or ‘No’ (that it be negated) are presumed to indicate *positive* and *negative* sentiment, respectively.

However, as MP voting is to a large extent constrained by party affiliations, with members often under pressure to follow the party whip regardless of their personal opinion (Searing, 1994; Norton, 1997), we perform sentiment analysis experiments on the Hansard Debates with Sentiment Tags (HanDeSeT) corpus, which features manually annotated sentiment labels in addition to those extracted from division votes (Abercrombie and Batista-Navarro, 2018).

In Parliament, the tabled motions under debate, by their nature, either approve of or oppose some piece of legislation or state of affairs, and hence also display sentiment polarity towards those targets. We therefore present a two-stage sentiment analysis model in which first, the sentiment of the motion towards the subject of the debate is determined, before sentiment analysis is carried out on the corresponding speeches.

Our contributions In this paper, we compare the use of speakers’ division votes with manually annotated polarity labels for the evaluation of sentiment analysis systems, and introduce a two-step sentiment analysis model for parliamentary debates in which the sentiment of both *speeches* and *motions* are classified.

For the two-step model, we also propose an alternative method for determining motion sentiment that infers polarity labels from the relationship to the Government of the speakers who introduce the motions

Additionally, we evaluate the use of *n*-gram textual features and a range of contextual features extracted from metadata related to the speakers.

2. Background: UK parliamentary debates

The UK Parliament consists of two chambers: the House of Commons and the House of Lords. The former is the superior legislative chamber, the target of most public and media attention, and the focus of this study.

Each debate in the House of Commons begins with a *motion* proposed by an MP. Following this, MPs may speak, when invited, any number of times during a debate. Each speaking turn may be comprised of a short statement or question, or a longer passage, divided into paragraphs in the transcript.

At any time during a debate, but most typically at the end, a *division* may be called. At this point MPs physically file through one of two *division lobbies* to register their vote—'aye' to support, and 'no' to oppose the motion in question. Labels extracted from the records of these divisions are referred to in this paper as division vote sentiment labels.

3. Related Work

Sentiment analysis has attracted substantial interest in NLP research, where the majority of work focusses on determining people's opinions in product reviews (e.g., Pang et al. (2002), Mukherjee and Bhattacharyya (2012)) and social media posts (e.g., Pak and Paroubek (2010), Rosenthal et al. (2017)).

In the political speech domain, several papers address the application of opinion classification to debates from the United States Congress. For example, Thomas et al. (2006) use a supervised classification model (support vector machine) to determine whether or not individual speech segments support a piece of legislation, using contextual discourse information to obtain enhanced performance, while Burfoot et al. (2011) apply a collective classification approach to Congressional speeches, using the speakers' voting records to obtain sentiment labels. In Europe, Grijzenhout et al. (2010) perform sentiment analysis at the paragraph level on manually labelled Dutch parliamentary transcripts.

For a related but somewhat different task on UK Hansard transcripts, Duthie et al. (2016) present a manually annotated corpus for the detection of speakers' positions, not towards the subject of debate, but rather other members' 'ethos'—which they define as the 'character' of the target, who is another participant in the debate.

For sentiment analysis on this domain, Onyimadu et al. (2013) use a sentiment lexicon to identify opinionated text in House of Commons debates for ternary (*positive*, *negative*, *neutral*) classification at the sentence level, reporting an average accuracy of 43% agreement between a classifier's predictions and the manually applied gold standard labels.

The most similar approach to ours is that of Salah (2014), which compares text classification using machine learning techniques and the use of sentiment lexicons to predict 'speaker attitude' on the concatanated speeches of MPs in the House of Commons, again relying on members' division votes as labels. We challenge the assumption that these votes reflect speaker sentiment by comparing these labels with those of human annotators. We also extend their use of party affiliation information, including other meta information about the debate participants, and examine whether these features are indeed predictive of sentiment as expressed in the speeches, or simply of likely voting outcome.

4. Data: the HanDeSeT corpus

We use the Hansard Debates with Sentiment Tags (HanDeSeT) corpus (Abercrombie and Batista-Navarro, 2018).[1] The corpus consists of 1251 units, each of which is composed of a parliamentary speech of up to five utterances and an associated motion. Content inserted by the Hansard reporters, certain set procedural phrases, and quotations have all been removed from the text.

Each speech has two binary (`1` for *positive* or `0` for *negative*) sentiment polarity labels, produced with different labelling methods:

1. A speaker-vote label extracted from the division associated with the corresponding debate: 'aye' = `1`, 'no' = `0`.
2. A manually annotated gold standard label.

All motions also have been assigned two sentiment labels:

1. A label derived from the party affiliation of the MP who proposes the motion—`1` if they are a member of the governing party or coalition at the time of the debate, `0` otherwise.
2. A manually annotated gold standard label.

In addition, the following metadata is included with each unit: *debate ID*, *speaker party affiliation*, and *motion party affiliation*.

A detailed description of the corpus and annotation process can be found in Abercrombie and Batista-Navarro (2018).

5. Debate speech sentiment models

The motions tabled in these parliamentary debates express either positive or negative sentiment towards a piece of legislation, policy, or state of affairs, and members of the chamber speak either in support of, or in opposition to the motion. For example, a motion may call on members to approve or reject a Bill, Act or Paper, or express approval or condemnation of a policy or situation.

The sentiment polarity of the motion under debate may therefore have a significant effect on the language used by a speaker when either supporting or opposing the motion. For example, for motions that commend the Government, speeches which support the motion are likely to incorporate positive language, while those that oppose the motion will tend to include typically negative language. On the other hand, for motions that oppose Government policy, speeches favourable to the motion are themselves also likely to use typically negative language towards the Government, and unfavourable speeches will conversely use positive language, as in Example 1.[2]

[1] HanDeSeT is available at `https://data.mendeley.com/datasets/xsvp45cbt4`.

[2] For further examples, see Abercrombie and Batista-Navarro (2018).

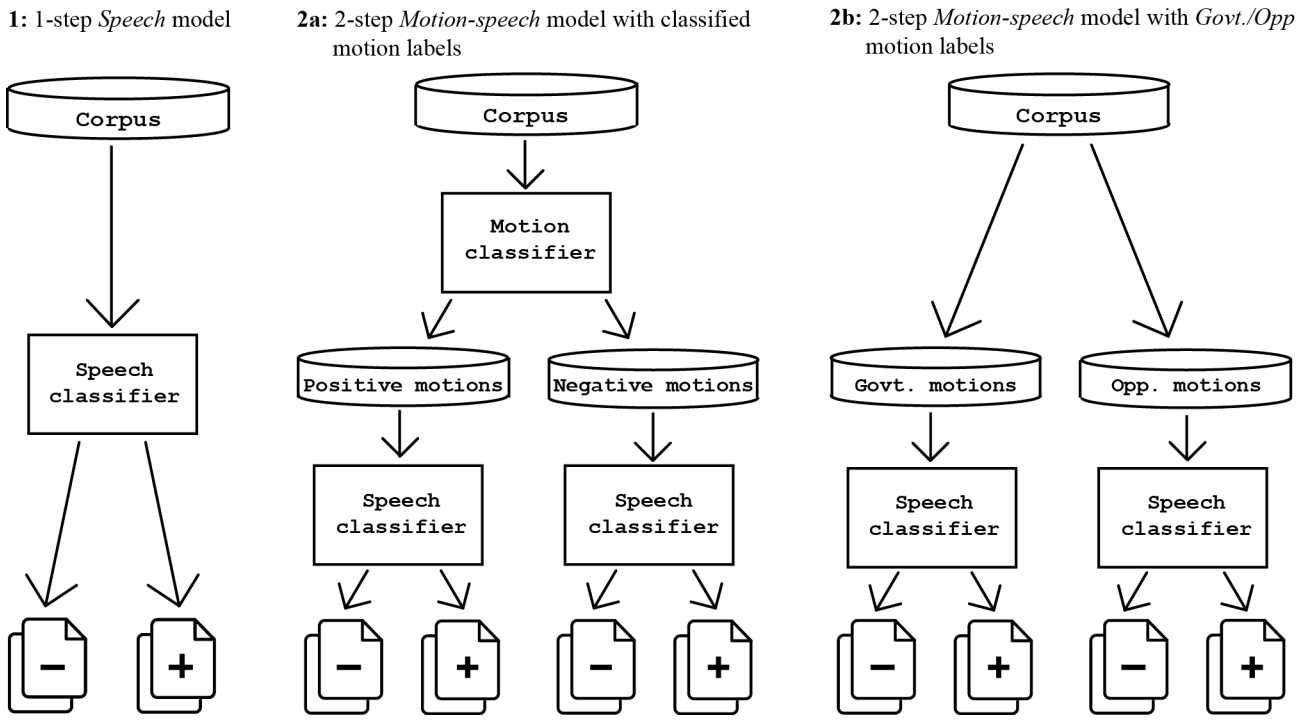


Figure 1: Three classification models for sentiment analysis of parliamentary debates. In model 1, all speeches are classified together, while in models 2a and 2b, speeches given in response to *positive* and *negative* motions are classified separately.

(1) **Motion:** That the Corporation Tax (Instalment Payments) Regulations 1998 (S.I., 1998, No.3175), dated 17th December 1998, a copy of whichwas laid before this House on 17th December, be revoked.

Speech: I do not support the regulations. The Government's rhetoric and practice do not add up. If I may paraphrase a well-respected authority, that which we call a tax riseby any other name would sting as hard, and that would be the effect of the regulations.

In this case, the motion expresses negative sentiment towards a piece of legislation, and the speech (extract) uses negative language to communicate positive sentiment towards the motion.

This 'double negative' effect presents complications for the learning of textual classification features, where lexical features that may be indicative of sentiment can differ in their polarity depending on the sentiment of the motion to which they respond. We therefore propose two models for comparison–as well as two different ways of classifying debate motions (see Figure 1):

1. Model 1: A one-step *Speech sentiment* analysis model, in which all units in the corpus are passed to the classifier simultaneously.

2. Model 2: A two-step *Motion-speech sentiment* analysis model, in which the corpus is first divided into those units with motions expressing positive, and those expressing negative sentiment polarity, before these two groups are classified separately. For this model, we also compare two methods of applying sentiment labels to the motions:

   (a) 2a: Sentiment classification using *n*-gram text features and learned from manually annotated labels.

   (b) 2b: Under the assumption that motions proposed by the Government are positive, and those proposed by other parties are negative, motions are divided by the party affiliation of the MP that proposes them–*positive* if they are a member of the governing party or coalition, *negative* if not.

6. Experiments

We perform experiments to compare sentiment classification performance using combinations of the following:

- Two machine learning models:
  - Support Vector Machines (SVM)–linear support vector classification.
  - Multi-layered Perceptron (MLP)–a neural network with 100 hidden layers, using rectified linear unit (ReLu) activation, L-BFGS optimization and maximum 200 epochs.
- Supervised learning of sentiment polarity classes using both manually annotated labels and division vote labels.
- The two debate models: the one-step *Speech sentiment* model, and the two-step *Motion-speech sentiment* model. For the *Motion-speech* model, we also

compare classification of the motions using *n*-gram textual features with labelling them simply according to the party affiliation of the MP who proposes the motion–*positive* if they are a member of the governing party or coalition, *negative* otherwise.

- The following learning features:
  - Textual features extracted from lowercased, tokenized motions and speeches:
    - *N-grams*: all *uni-*, *bi-*, and *trigrams*, and combinations of these.
  - Contextual metadata features for speech classification:
    - *Speaker party affiliation*. Intuition suggests that a speaker's party membership should be a strong indicator of sentiment towards many topics, and Salah (2014) showed this to be the case, at least as far as correlation with speakers' division votes goes.
    - *Debate ID* number. As there are usually multiple speeches in each debate, and MPs will often express similar sentiments to members of their own party in a particular debate, we also follow Salah (2014) in including this feature to capture possible correlations between MPs' speech and voting behaviour.
    - *Motion party affiliation*. Because MPs are likely to be more or less supportive of a motion depending on who proposes it, we add that Member's party as a further contextual feature.

7. Results & Discussion

We present the results of classification using 10-fold cross-validation. Due to slight imbalances in class labels, F1 scores are reported in addition to accuracy.

For motion classification, the SVM classifier achieves accuracy of 92.1% and an F1 score of 0.921, while the MLP classifier obtains accuracy of 93.0% and an F1 score of 0.931. Considering human agreement rates on this task (Cohen's κ = 0.91[3]), this is probably close to the optimal performance that could be expected.

Many of the features most indicative of positive motion sentiment are related to the practicalities of legislation, reflecting the fact that many of these motions are brought by the Government in an effort to pass law. Many negative motions include structures such as '(this House) believes that/notes that/disagrees with/calls on the Government to...', and this is also reflected in the most discriminating *n*-gram features (see Table 1).

Speech classification performance scores are presented in Table 2. The higest accuracy and F1 scores overall, using both labelling methods, are achieved using all features to train the MLP classifier.

These results provide a number of insights into the relationships between the labelling methods used, the textual and metadata features in the corpus, and the debate models applied.

[3]For more on inter-annotator agreement for this corpus, see Abercrombie and Batista-Navarro (2018).

	Positive	Negative
1	security	notes
2	connection	amend
3	given	believes
4	purposes	calls
5	general	government
6	new	calls government
7	schedule	dated
8	proceedings	eu
9	session	disagrees
10	programme	number

Table 1: Top 10 most discriminating *positive* and *negative* *n*-gram features ranked by SVM training coefficients using manually annotated labels.

7.1. Labelling Methods

Results indicate a correlation between the labelling method used and performance resulting from the use of different feature types for classification. Use of manually annotated labels leads to slightly better performance when only textual features are considered, while with division vote labels, the inclusion (or exclusive use) of meta data leads to considerable gains in performance (see Figure 2).

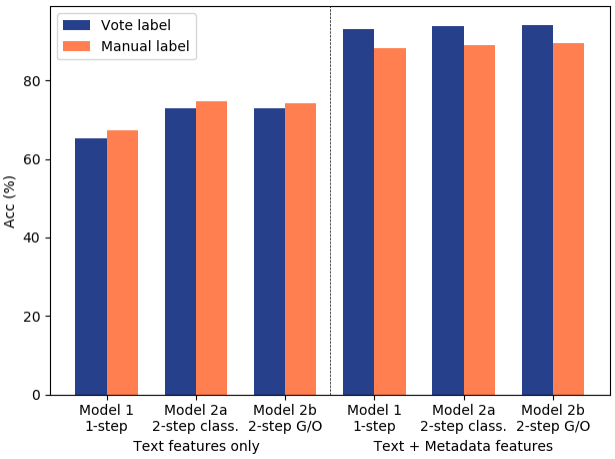


Figure 2: Comparison of manual and division vote labelling methods (using MLP classifier) with contextual features excluded or included.

It therefore appears that information in the text correlates more closely to human understanding of the sentiment expressed in the speech, while contextual information regarding the speakers involved is more indicative of voting intention, with *speaker party affiliation* a particularly strong indicator of this label.

However, while these results support the hypothesis that manual labels are more indicative of speech sentiment, considering the associated costs and the relatively small differences in performance, use of division votes may be the

Debate model	Motion label	Speech label	Features	SVM Acc. (%)	SVM F1 score	MLP Acc. (%)	MLP F1 score
1: *Speech*	n/a	Vote	Text only	64.3	0.699	65.3	0.699
			Text+Party	78.8	0.815	79.2	0.817
			Text+Party+ID	82.7	0.848	87.1	0.888
			Text+Party+ID+Motion	82.6	0.847	93.0	0.938
			Party+ID	83.3	0.853	86.0	0.878
			Party+ID+Motion	83.5	0.854	92.9	0.938
		Manual	Text only	66.7	0.718	67.3	0.713
			Text+Party	76.2	0.791	76.6	0.793
			Text+Party+ID	79.7	0.821	82.4	0.845
			Text+Party+ID+Motion	79.8	0.821	88.2	0.896
			Party+ID	79.9	0.821	82.1	0.842
			Party+ID+Motion	80.0	0.822	88.4	0.897
2a: *Motion-Speech*	Classifier	Vote	Text only	72.9	0.743	72.8	0.739
			Text+Party	83.9	0.835	83.4	0.830
			Text+ID+Party	86.1	0.853	90.7	0.905
			Text+ID+Party+Motion	86.5	0.859	93.9	0.940
			Party+ID	83.3	0.821	91.4	0.915
			ID+Party+Motion	83.2	0.818	93.5	0.935
		Manual	Text only	<u>74.7</u>	0.710	<u>74.6</u>	0.713
			Text+Party	81.0	0.760	81.1	0.772
			Text+Party+ID	83.1	0.794	86.2	0.837
			Text+Party+ID+Motion	82.9	0.790	89.1	0.883
			Party+ID	80.7	0.747	87.1	0.859
			Party+ID+Motion	79.6	0.734	89.0	0.878
2b: *Motion-Speech*	Govt./opp	Vote	Text only	73.1	<u>0.756</u>	72.9	<u>0.748</u>
			Text+Party	85.1	0.853	84.8	0.850
			Text+Party+ID	87.5	0.874	91.7	0.919
			Text+Party+ID+Motion	**87.8**	**0.877**	**94.1**	**0.943**
			Party+ID	82.9	0.820	92.9	0.930
			Party+ID+Motion	84.9	0.848	93.5	0.937
		Manual	Text only	74.3	0.736	74.2	0.736
			Text+Party	72.6	0.799	82.8	0.809
			Text+Party+ID	84.8	0.828	87.4	0.860
			Text+Party+ID+Motion	84.4	0.824	89.6	0.892
			Party+ID	80.8	0.768	88.3	0.876
			Party+ID+Motion	80.6	0.770	89.1	0.885

Table 2: Accuracy and F1 scores for one- and two-step models—the latter using automatically classified motion sentiment labels or *Government/opposition* motion sentiment labels. Results include division vote and manually annotated sentiment labels, and speech sentiment classification is performed using the support vector machine (SVM) and the multi-layer perceptron (MLP) classifiers. The best overall scores for each metric are in bold and best scores using textual *n*-gram features only are underlined.

more pragmatic choice for this task for practical purposes.

7.2. Debate Models

Compared to the one-step *Speech* model, use of the *Motion-speech* models produces improved results for both classifiers under most model-feature configurations. It therefore seems that use of such a two-step model may go some way towards capturing the complex nature of these debates in which positive language can indicate negative sentiment polarity and vice-versa.

Exceptions to this occur when the classifier is trained using contextual metadata features only. Here, as textual features are ignored, the two-step model becomes effectively redundant.

Interestingly, the use in model 2b of labels derived from the relationship of the MP who proposes the motion to the Government (*Government* or *opposition*) is generally as effective as training a classifier on manually annotated labels (model 2a). This suggests that a two-step *Motion-speech* model can be used without the need for costly manual annotations, at least as far as motion sentiment labels are concerned.

7.3. Features

For textual features, the inclusion of bi- and trigrams does not appear to significiantly improve speech classification performance over the use of only unigrams for this task, particularly for the two-step models (see Figure 3).

	1: One-step *Speech* model				2a: Two-step *Motion-speech* model							
	All motions				Positive motions				Negative motions			
	Vote label		Manual label		Vote label		Manual label		Vote label		Manual label	
	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
1	re-search	labour	commu-nity	labour	rural	proper-ly	accept	treaty	tory	labour	week	labour
2	ridicu-lous	money	article	oppor-tunity	hunting	position	impor-tant	pro-gramme mo-tion	place	shadow	tell	shadow
3	street	shadow	ridicu-lous	unfair	press	night	rules	night	commu-nities	suggest-ing	home	snp
4	young people	centres	decis-ion	treaty	open	like	fox	post	impact	snp	young	chilcot
5	work-ing	canna-bis	condi-tions	kent	right	central	increa-ses	prin-ciple	particu-larly	look	yester-day	consult-ation
6	issue	raise	crisis	large	higher	getting	settle-ment	concern	lost	general	public	app-roach
7	higher	leader	people	order	equip-ment	im-posed	pro-gress	in-crease	women	iraq	today	motion
8	cent in-crease	re-quired	early	lowest	dogs	state	poss-ible	floor	conser-vative	centres	welsh	future
9	left	time-table	higher	proper-ly	sub-stan-tial	brought	congra-tulate	head	explain	use	needs	suggest-ing
10	investi-gation	central	young people	coun-cils	area	wales	higher	review	yester-day	benefit	legal	contract
*	0.2	0.2	0.4	0.9	0.2	0.5	0.4	0.2	-0.3	0.5	0.4	0.2

Table 3: Top 10 most discriminating textual n-gram features ranked by coefficients learned by training the SVM classifier. The bottom row of this table (*) shows the total mean sentiment score of the items in each column, as extracted from SentiWordNet 3.0.

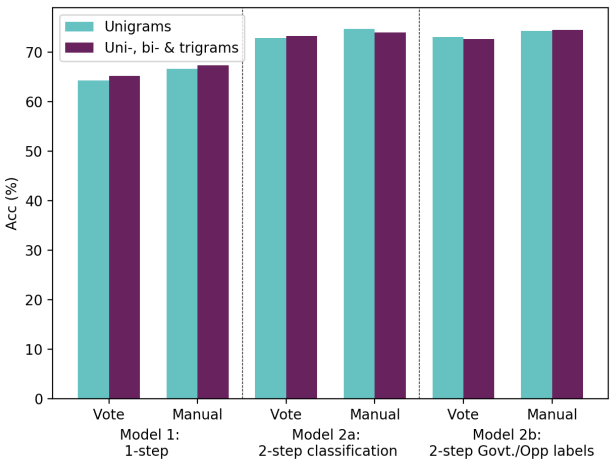


Figure 3: Comparison of MLP classification accuracy using unigram only and uni-, bi-, and trigram textual features. In most configurations, the addition of bi- and trigrams does not notably improve performance over use of unigrams alone.

Ranking of *n*-grams by their SVM training coefficients also reveals that few bigrams and no trigrams feature in the top 10 most discriminating features (see Table 3). Examination of these predictive items underlines the fact that discriminating textual features for this task are not generally those that would be thought of as expressing positive or negative sentiment, even when using the two-step model. Calculating the average polarity of these lexical items (mean score of all entries for each item) according to a sentiment lexicon,[4] we find that 36.7% are neutral, 42.5% positive, and only 16.7% negative. This suggests that MPs tend to follow parliamentary guidelines to practise 'good temper and moderation',[5] avoiding negative language in these debates, whatever point they may be making.

The acquisition of sentiment polarity we see here by objectively neutral language may also be due to the corpus containing a combination of debates on a wide variety of subjects and a relative sparsity of speeches addressing each of these topics. In debates which are skewed towards hav-

[4]SentiWordNet 3.0 (Baccianella, S. and Esuli, A. and Sebastiani, F., 2010), available at `http://sentiwordnet.isti.cnr.it/`.

[5]May (1844) in `https://www.parliament.uk/documents/rules-of-behaviour.pdf`.

ing more speakers either supporting or opposing the motion, topic words can become indicative of one or the other polarity. Hence, in this corpus, generally neutral lexemes such as 'fox' or 'Wales' become indicative of positive and negative sentiment polarity respectively.

While use of contextual metadata features, improves overall performance, in some cases their inclusion leads to incorrect classification. This is prevalent in cases where an MP's sentiment is contrary to that of the majority of other members of their party, or in debates where MPs do not vote along party lines. In such cases, *party affiliation* can be a confounding feature and lead to incorrect classification.

7.4. Classifiers

Using textual features only, there is no significant difference between the performance of the two classifers. However, when contextual metadata features are included, the MLP tends to obtain higher accuracy and F1 scores, suggesting that such neural networks may be better able to exploit the complex relationships between textual and contextual cues in these parliamentary debates.

7.5. Error Analysis

Even using the best performing model-classifier-label-features configurations, some speeches are not classified correctly.

We manually examined the examples for which, using all learning features, and no matter which labels or model were used, the MLP classifier's predicted labels did not match the supervision labels. In the majority of these cases, we observed the following:

1. Speeches were longer than average (μ 218.8 vs. 167.8 words for the whole corpus).

2. Either: speech sentiment labels did not agree with the majority of that speaker's party (19.4% of errors), the speaker's party was split in the debate concerned (11.9%), the speaker was the only member of their party in this debate (22.4%), or the debate featured only that one speech (4.5%).

In the remaining cases, speeches by Conservative MPs were erroneously classified as *negative*, and those of Labour or SNP speakers as *positive*. It therefore appears that the *party affiliation* feature may carry too much weight. While this feature is clearly strongly indicative of speaker sentiment, it can lead the classifier to over-generalise.

For the use of textual features only, we also examined examples in which the best performing (highest accuracy) configuration–the *Motion-speech* model with SVM and manual labels–classified speeches incorrectly. While it is difficult to identify a common thread between all these cases, it appears that on many occasions, these speeches feature speakers addressing off-topic or tangentially related subject matter (see Example 2, in which the speaker talks about a different event than the target of the motion).

(2) **Motion:** That the draft European Union Referendum (Date of Referendum etc.) Regulations 2016, which were laid before this House on 22 February, be approved.

Speech: On suspicious intentions, may I remind the right hon. Gentleman that he campaigned with the Conservative party and the Labour party in Scotland, telling the people of Scotland that if they voted no in the Scottish referendum, they would be guaranteed to remain in the EU? What is his position on that point today?

Even when speeches do contain subjective language directed at the motion, as in Example 3, multiple opinion targets, such as other MPs, parties, and topics, can also be present, complicating the task of sentiment analysis at this level of granularity.

(3) **Speech:** We have always been opposed, and we continue to be opposed, to guillotines. They are wrong in principle and in this case. However, we are realistic and we know that the Government have a majority. We welcome very much the comments and support of the hon. Member for Thurrock...
First, the Bill is unnecessary and should not have been introduced...
As the Government failed to think the matter through and to act, it is unfair that hon. Members should be penalised by lack of time...
Secondly, until a few minutes ago, I was under the impression that the Opposition line was to make their point on the guillotine, but not to divide the House. That will only penalise us, as we will lose another 15 to 20 minutes. I ask the hon. Member for Grantham and Stamford to think.

8. Conclusions

We have evaluated the use of manually annotated labels and division vote labels for sentiment analysis of speeches taken from Hansard UK House of Commons debate transcripts in the HanDeSeT corpus. We have also introduced a new two-step model for debate speech sentiment analysis, and evaluated its performance against the one-step model. We also compared the performance on this task of both SVM and MLP classifiers, and the use of both textual *n*-gram features and contextual metadata features.

Results suggest that while contextual metadata can be highly predictive of their division vote, manually annotated labels more closely reflect speakers' sentiment as expressed in their speeches. However, considering the large overlap between the two sets of labels, for future work or to create larger datasets, manual annotation of these may not be cost-effective.

Our two-step *Motion-speech* model outperforms a simple one-step model in nearly all label-feature-classifier configurations, and therefore seems better able to take account of the complexities inherent in the structure of House of Commons debates, such as double negation. Additionally, we

have found that labelling motions according to the relationship to the Government of the speakers who propose them can approximate the effects of sentiment classification in debate motions, thus avoiding the need for costly manual annotations for this step.

Overall, it seems that sentiment analysis of Hansard transcripts at the speech level does not yield major insights beyond those that could be obtained by merely examining MPs voting records. A more fine-grained analysis may be required to access the opinions expressed in these debates. In future work, we will focus on applying sentiment analysis to the different targets of the speakers' sentiment such as the various topics and subtopics that arise in parliamentary debates.

9. Acknowledgements

The authors would like to thank Loren Hosein, Kieran Lilwall and Anthony Chambers for their contributions to this project, and the anonymous reviewers for their invaluable comments.

10. Bibliographical References

Abercrombie, G. and Batista-Navarro, R. (2018). A sentiment-labelled corpus of Hansard parliamentary debate speeches. In *Proceedings of ParlaCLARIN*. Common Language Resources and Technology Infrastructure (CLARIN).

Burfoot, C., Bird, S., and Baldwin, T. (2011). Collective classification of congressional floor-debate transcripts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1506–1515. Association for Computational Linguistics.

Duthie, R., Budzynska, K., and Reed, C. (2016). Mining ethos in political debate. In *COMMA*, pages 299–310.

Grijzenhout, S., Jijkoun, V., Marx, M., et al. (2010). Opinion mining in Dutch Hansards. In *Proceedings of the Workshop From Text to Political Positions, Free University of Amsterdam*.

May, T. E. (1844). *A treatise upon the law, privileges, proceedings and usage of Parliament*. C. Knight & Company.

Mukherjee, S. and Bhattacharyya, P. (2012). Feature specific sentiment analysis for product reviews. *Computational Linguistics and Intelligent Text Processing*, pages 475–487.

Norton, P. (1997). Roles and behaviour of British MPs. *The Journal of Legislative Studies*, 3(1):17–31.

Onyimadu, O., Nakata, K., Wilson, T., Macken, D., and Liu, K. (2013). Towards sentiment analysis on parliamentary debates in Hansard. In *Joint International Semantic Technology Conference*, pages 48–50. Springer.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 19–21. European Language Resources Association (ELRA).

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135, July.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Rosenthal, S., Farra, N., and Nakov, P. (2017). Semeval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518.

Salah, Z. (2014). *Machine learning and sentiment analysis approaches for the analysis of Parliamentary debates.* Ph.D. thesis, University of Liverpool, UK.

Searing, D. (1994). *Westminster's world: understanding political roles*. Harvard University Press.

Thomas, M., Pang, B., and Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 327–335. Association for Computational Linguistics.

11. Language Resource References

Abercrombie, G and Batista-Navarro, R. (2018). *HanDeSeT: Hansard Debates with Sentiment Tags*. Mendeley Data.

Baccianella, S. and Esuli, A. and Sebastiani, F. (2010). *SentiWordNet 3.0*. SentiWordNet.

Scalable Visualisation of Sentiment and Stance

Jon Chamberlain[1], Udo Kruschwitz[1], Orland Hoeber[2]
[1]University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK
[2]University of Regina, 3737 Wascana Pkwy, Regina, SK S4S 0A2, Canada
{jchamb,udo}@essex.ac.uk, orland.hoeber@uregina.ca

Abstract

Natural language processing systems have the ability to analyse not only the sentiment of human language, but also the stance of the speaker. Representing this information visually from unevenly distributed and potentially sparse datasets is challenging, in particular when trying to facilitate exploration and knowledge discovery. We present work on a novel visualisation approach for scalable visualisation of sentiment and stance and provide a language resource of e-government public engagement of 9,278 user comments with stance explicitly declared by the author.

Keywords: sentiment, stance, information visualisation, opinion mining, e-government, language resource

1. Introduction

The ever growing amount of user generated content online presents a variety of challenges, one of them being how to summarise large collections of user comments in online discussion forums and social networks, capturing the gist of what has been said and reflecting the different view points on a range of topics. Natural language processing systems have the ability to analyse not only the sentiment of this communication, but also the stance of the speaker.

Online participation is increasing at an exponential rate and the desire for consumption of such data has driven innovations in communication platforms, in particular on social media platforms such as Facebook[1] and Twitter[2]; however, similar scenarios can exist at much smaller, local scales such as comments on online newspaper reports and e-government consultation schemes.

Representing information visually from these unevenly distributed and potentially sparse datasets is problematic, in particular when trying to facilitate exploration and knowledge discovery. Traditional linear scaling will result in large values dominating smaller ones; logarithmic scaling addresses this problem, but is not intuitive to casual users.

This paper addresses these challenges by discussing related work in the area of sentiment and stance analysis, as well as information visualisation relevant to this field (Section 2.). We then propose a novel approach to scalable visualisation of sentiment and stance (Section 3.). In Section 4. we outline a case study of how this approach can be used for visualising user participation on an e-government portal, followed by further details of how the language resource was acquired and how it will be shared with the community (Section 5.). In Section 6. we compare our approach to other visualisation techniques and, in conclusion, we discuss the implications of this approach and its relevance to the NLP researchers (Section 7.).

2. Related Work

Summarising and visualising online discussions has become ever more important for users to understand the complexities of multi-faceted issues in modern society. There are many approaches for visualising sentiment or stance as time series data, including timelines (Kucher et al., 2014), dense pixel displays (Hao et al., 2011), summaries of online conversations (Riccardi et al., 2016) and summaries of Twitter data (Mohammad et al., 2016a). Websites such as CreateDebate[3] and Kialo[4] attempt to organise discussion into supporting and refuting statements; the latter providing users with interactive overviews and maps of the discussion. These attempts to "empower reason" go some way towards helping users; however, it remains a challenge to present numerous discussions points in a relevant and unbiased way.

Some approaches have developed methods for dealing with topically organised text (Hoque and Carenini, 2016); however, dealing with potentially extreme differences in the amount of data, whilst still supporting cross-topic comparisons, remains problematic. If the data are unevenly distributed or sparse, a large number of data points for a particular topic can make it difficult to visually convey the information in the smaller topics. Furthermore, if there is a desire to evaluate the sentiment of the human language against the formal stance taken by the author, care must be taken to ensure that the visual encodings are comparable.

Adding interactivity enables the user to engage in information seeking among the textual data, using the sentiment and stance visualisation as a source of "information scent" (Pirolli and Card, 1999). While much progress has been made in searching such collections, e.g. via paradigms like faceted search (Russell-Rose and Tate, 2013), exploration and navigation have traditionally attracted less attention. Integrating natural language processing, information visualisation, and interactive information retrieval holds great value for enabling people to discover meaningful information from within complex information repositories (Hoeber, 2014). This research presents an approach that enables the visual analysis and comparison of sentiment and stance over a potentially large number of topics (either represented explicitly or derived from the data).

[1] https://www.facebook.com
[2] https://twitter.com
[3] http://www.createdebate.com
[4] https://www.kialo.com

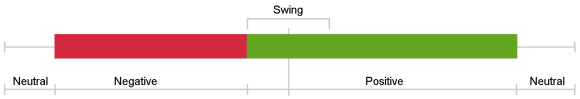


Figure 1: A basic representation of how a single bar is calculated using the percentages of total responses (positive, negative, and neutral).

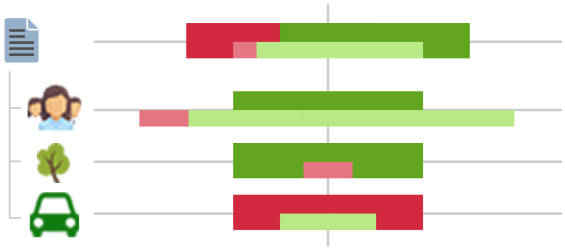


Figure 2: Representation of an expanded (by topic) double bar showing stance and sentiment simultaneously.

3. Approach

The goal of our approach is to visualise both sentiment and stance simultaneously, scaling the representation to allow for sets of different sizes to be shown and compared, and highlighting the polarisation of opinion.

Figure 1 shows how a single bar (stance or sentiment) is calculated. The full width of the bar represents 100% of the comments received, at either end of the bar are the proportion of neutral comments. Positive and negative comments are in the centre of the bar, with swing represented by deviation from the centre line. Figure 2 shows how these bar representations can be combined (stance embedded within sentiment), and used within a topic hierarchy.

This representation gives both overview and detail, displaying hierarchically organised sentiment and stance, and supporting sensemaking for analysis and synthesis of information. Such topical organisation can be also be considered an instance of faceted browsing (Tunkelang, 2009; Russell-Rose and Tate, 2013).

This approach is scalable to unevenly distributed datasets where most of the nodes have only a few data points but some have comparatively very large amounts. Linear representations do not scale for this type of data and logarithmic transformation is not intuitive to the user.

With the data organised based on a topical hierarchy, the visualisation enables topics to be compared to their parents and to one another and, within a given topic, the sentiment and stance can also be compared.

While doing so hides the absolute value of the number of comments made for each of the sentiment and stance classes, it enables a visual comparison of the relative swing across each topic. Following Shneiderman's *Information Seeking Mantra* ("overview first, zoom and filter, details on demand") (Shneiderman, 1996), this approach enables the user to interactively explore the information in a flexible manner. The default view of the sentiment and stance



Figure 3: User exploration is supported through interactivity of the graphs on mouse-over. A fly-out box shows more detail of how the chart is constructed by displaying an abridged comment, sentiment and stance.



Figure 4: Visualisations of explicit stance for three planning applications: #131452 has a total of 167 comments (94 object, 73 support, 0 neutral); #150239 has 336 comments (75 support, 252 object, 9 neutral); and #146223 has 4 comments (2 object, 1 support, 1 neutral).

provides an overview of general opinion expressed within the data. The hierarchical topics can be expanded and collapsed (e.g., zoomed), topics can be removed from the view (e.g., filtered), and specific sentiment and stance selections can be made in order to retrieve the associated comments (e.g., details on demand), see Figure 3.

More importantly, the visualisation provides a clear depiction of situations where there is significant swing in the sentiment or stance, and when the sentiment and stance are inconsistent with one another. These are important conditions since they represent potential issues that may need further study and analysis.

4. Case Study

It is common for Community Question Answering (cQA) systems to include methods for users to make comments, as well as to make upvotes or likes on comments (for example, StackOverflow[5] or Yahoo!Answers[6]) but it is less usual for the stance of the comment to be explicitly entered unless the platform is specifically for debate (e.g. CreateDebate and Kialo). Another example of where this does occur is in the commenting of planning applications.

In the past, opinions on a planning applications would have been submitted in writing to the planning office to support or object; however, this process has now moved online. Users can now access the details of an application through an online portal, leave text comments, upload files and are also explicitly asked whether they are writing to support or object to an application, or whether they are making a

[5] https://stackoverflow.com
[6] https://uk.answers.yahoo.com

general observation (i.e. a neutral stance). Our case study analyses data from such a portal[7], operated by Colchester Borough Council (CBC), based in Essex, UK.

The portal is used for conducting public consultation and to gain local insight regarding planning applications. The comments are then used alongside existing planning policy to shape decisions. Planning officers and local government officials have a gauge of public opinion regarding proposed applications, as well as a way of capturing knowledge from the community that planning officers may have missed.

One of the most important roles of the planning officer is to summarise the planning documents and comments submitted from the public in a planning document that goes before a committee to decide whether the planning application is approved, requires more information or is rejected. Online comments are not searched by planning officers but each is given due consideration, making a summary visualisation important in their workflow. In this particular case the user (planning officer) is not so interested in the quantity of opinions, rather whether so-called Material Planning Considerations[8] have been raised: matters that will influence the decision of the application such as parking, traffic, conservation, safety, etc. In this case the scalable visualisation ignores comment quantity and instead represents the balance between support, object and neutral opinions (see Figure 4). Further hierarchical decomposition and representations of the data would allow such issues to be located quickly (e.g. the last bar in Figure 2).

The comments vary in size from a single word (e.g. "object!") to hundreds of paragraphs covering numerous topics that provide evidence for the stance. Topic clustering may prove useful to break the long comments down; however, sentence-based analysis has been successful in analysing stance in online article comments (Riccardi et al., 2016). Examples (from the corpus) of straightforward stance include support (Example 1) and objection (Example 2); however, neutral stance (or general observations) can be more ambiguous (Example 3 and 4).

(1) Fantastic to bring new jobs to the town and boost the local economy.

(2) The local infrastructure cannot cope, especially in light with the proposed development of land currently occupied by the MoD.

(3) West Mersea Town Council are unable to comment, more detailed information required.

(4) Councillors have discussed this application and have no objections.

Within the corpus many additional ambiguous statements exist, inline with previous findings in Twitter data (Mohammad et al., 2016a). Despite being a formal system for members of the public to submit comments to a planning consultation, authors are often ambiguous in their meaning, with the use of sarcasm (Example 5), emotional responses (Example 6) and rhetorical questions (Example 7).

(5) People say it will kill the town centre, but our Council has already done that.

(6) Essential community services are being cut - is this fair?

(7) Why do we need these built when a large portion of green space in the local area is being bulldozed for mass housing?

The planning officer will be most interested in finding material considerations, in particular information that was not previously known (Example 8).

(8) I have seen great crested newts at the lower end of this project nearer the train station!

Table 1: Distribution of explicit, stance-labeled online text comments per planning application.

n(2,840)	**Min**	**Max**	**Mean**	**SD**
Total	1	435	3.5	15.4
Negative	0	420	1.8	12.0
Positive	0	252	1.0	7.3
Neutral	0	9	0.4	0.8
Swing	-417	203	-0.8	13.0

5. Language Resource

Planning application data, comments and associated metadata from 2008-16 was accessed from the CBC portal on 8 Feb 2017 (a total of 15,703 applications and 9,278 online text comments).

CBC receive an average 1,745 sd(271) planning applications per year of which the majority (67.5%) get conditional approval but 10.0% are refused (the remaining applications can be assigned 18 other statuses such as withdrawn, observations only, etc).

2,840 (18.1%) of the planning applications had at least one online comment (see Table 1) and some applications attract considerable interest. Nine applications had over 100 comments (the maximum being 435), 144 applications had over 10 comments and 349 applications had over five comments. This highlights the problem of scalability the planning team face when attempting to summarise this data.

Some applications will be simple and take very little time to process; others will take considerably longer and be more complex in the summary. Each comment also contains the explicitly entered stance of the author regarding the application (object, support or neutral) and by summing these stance values we can calculate the swing.

The highest object swing was 417 from 435 total comments and the highest support swing was 203 from 246 total comments. The average swing for the entire dataset was 0.8 sd(13.0) object, an indication that users are more likely to use the portal to post comments objecting to proposals.

[7] `http://www.colchester.gov.uk/article/13483/Search-Applications-Online`, accessed 1 Feb 2018

[8] `http://www.colchester.gov.uk/article/13488/Commenting-on-a-Planning-application`, accessed 1 Feb 2018

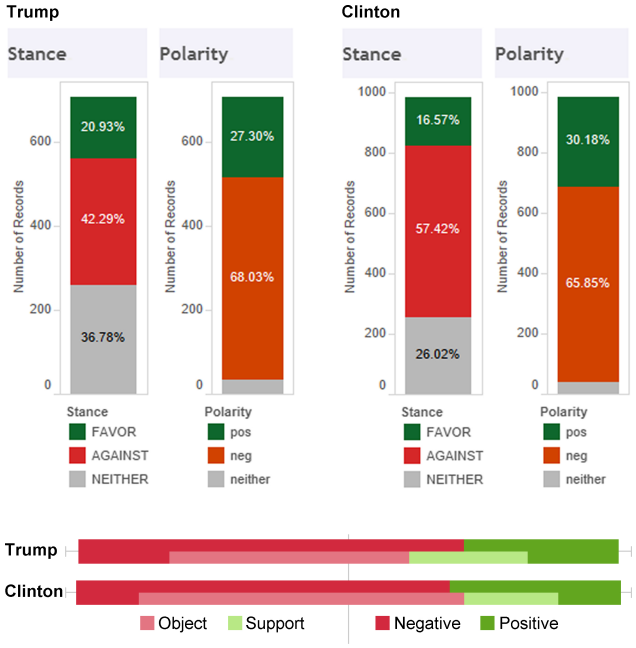


Figure 5: Visualisation of the SemEval dataset, comparing the detailed view (top) of SemEval for stance and sentiment (polarity) with the compact view of our approach (bottom) on two targets: Donald Trump and Hillary Clinton.

The comments are publicly available online via the portal and the distribution of the anonymised language resource for research purposes has been agreed with CBC.

6. Comparison to SemEval

Stance and opinion datasets for popular social media such as Twitter have started to emerge (Mohammad et al., 2016a), and have found their way into academic competitions such as SemEval (Mohammad et al., 2016b). Here we compare our proposed visualisation approach with that used for the visualisation of the SemEval dataset. Figure 6 compares the SemEval visualisation with our compact visualisation for two targets: Donald Trump and Hillary Clinton. Whilst lacking the details of the SemEval approach, our approach still visualises the similarity between sentiment (also called polarity) distribution and the differences between stance of the two targets. Comparing three or more targets would become increasing more difficult for the SemEval approach, whereas our approach would scale to the amount required.

7. Discussion

The impact of scalable visualisation of sentiment and stance can be seen within the workflow of planning applications. Planning officers have a legal obligation to consider all material issues raised during the consultation period of planning. Should considerations be missed or ignored and the planning decision challenged, the council would be liable for any compensation due to the decision being changed. Therefore, the laborious and meticulous process of analysing each comment is required; however, the benefit of support through summarisation and visualisation techniques would increase the efficiency of dataflow through the organisation and identify potential risk areas.

Analysis of the data from the case study supports the notion that negative motivations are more powerful than positive ones for those participating in online forums i.e., people are more likely to complain or object than they are to agree or support but it is not surprising to see this bias in official channels of communication.

In a wider application, this type of visualisation would be helpful for the general public to view complex data sources, in particular those that have been subject to manipulation through fake news and reporter bias. An example is within the voting records of politicians (their recorded stance) compared to the words they use during speeches and interviews (their observed sentiment and stance). This avenue of research is being actively explored and may yield interesting insights into voting participation and behaviour relevant for NLP involvement in future e-government engagement initiatives.

Previous research (Mohammad et al., 2016a) has highlighted that, whilst reasonably accurate to classify using machine learning algorithms, sentiment analysis is not sufficient to understand human responses in the form of comments or discussion, and additionally that stance is a complex concept to analyse and visualise. Determining targets of opinion vs targets of interest (i.e. targets being mentioned vs targets that are not) make the task even more difficult, hence our desire to make available a dataset with stance explicitly mentioned.

In order to assist planning officers to identify and make sense of the application comments, this visualisation will be implemented and evaluated on their data to inform the development of a new platform (see Figure 6). Work to extend the approach to support information seeking strategies is ongoing.

8. Conclusion

In this paper we present an approach to scalable visualisation of sentiment and stance to enable analysis and exploration of complex large-scale data. This approach has been implemented on a case study of e-government planning consultation data where participants explicitly declare their stance to an application as well as write a comment. This work has the potential to provide impact in terms of cost savings and efficiency in the long term for e-government engagement and wider applications. The language resource used in this research will be made available to practitioners in this field.

9. Acknowledgements

We acknowledge support from grant number ES/L011859/1, from The Business and Local Government Data Research Centre, funded by the Economic and Social Research Council to provide researchers and analysts with secure data services. Thanks to Sally Harrington and Andrew Tyrrell from CBC for advising on the case study.

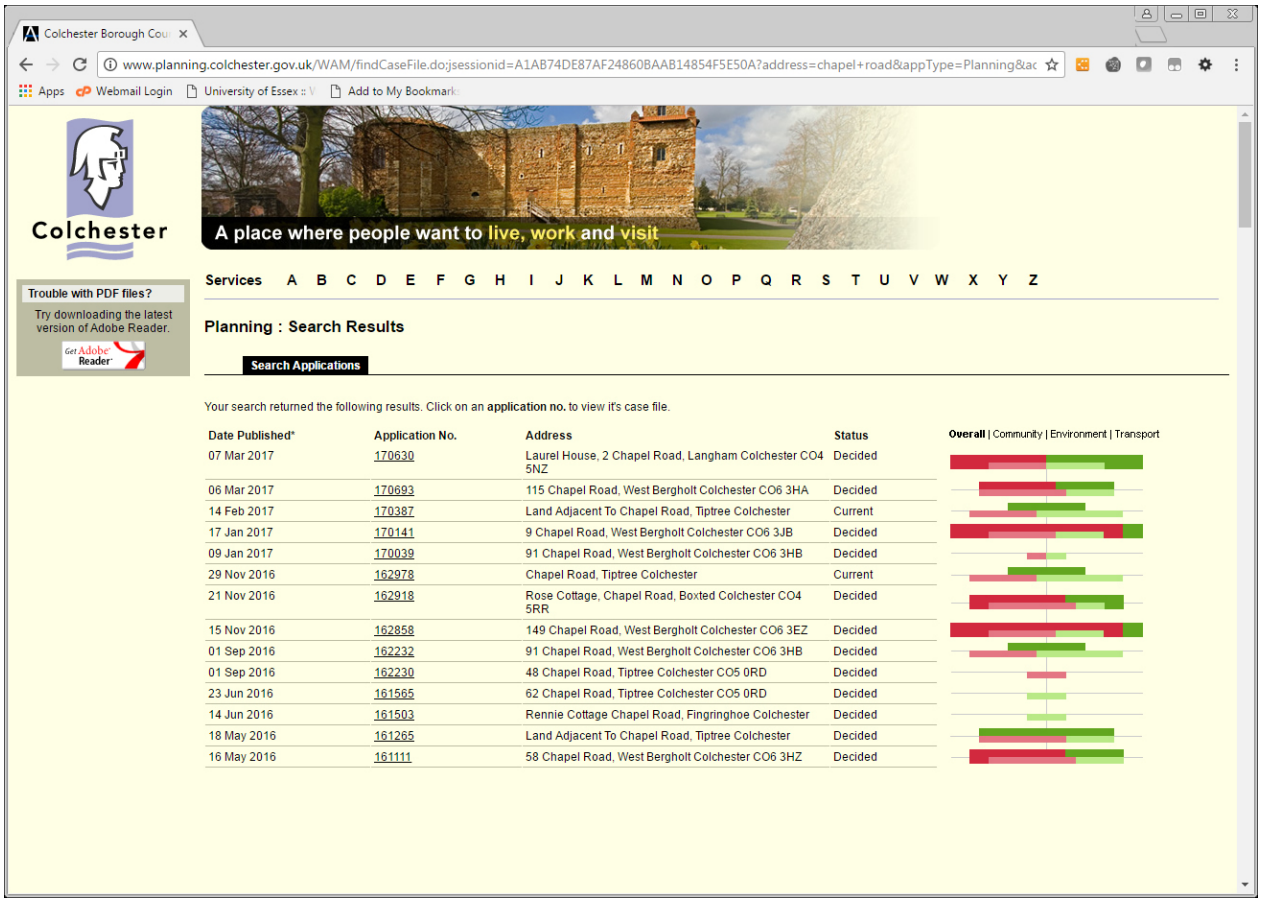


Figure 6: A screenshot showing how the visualistion can be applied to the existing portal to enhance understanding and prioritisation of planning applications.

10. Bibliographical References

Hao, M., Rohrdantz, C., Janetzko, H., Dayal, U., Keim, D. A., Haug, L. E., and Hsu, M. C. (2011). Visual sentiment analysis on Twitter data streams. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 277–278.

Hoeber, O. (2014). Visual search analytics: Combining machine learning and interactive visualization to support human-centred search. In *Proceedings of Beyond Single-Shot Text Queries: Bridging the Gap(s) Between Research Communities Workshop*, pages 37–43.

Hoque, E. and Carenini, G. (2016). MultiConVis: A visual text analytics system for exploring a collection of online conversations. In *In Proceedings of the International Conference on Intelligent User Interfaces*, pages 96–107.

Kucher, K., Kerren, A., Paradis, C., and Sahlgren, M. (2014). Visual analysis of stance markers in online social media. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 259–260.

Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016a). A dataset for detecting stance in tweets. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016b). Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June.

Pirolli, P. and Card, S. (1999). Information Foraging. *Psychological Review*, 106(4):643–675.

Riccardi, G., Béchet, F., Danieli, M., Favre, B., Gaizauskas, R. J., Kruschwitz, U., and Poesio, M. (2016). The SENSEI project: Making sense of human conversations. In *Proceedings of the First International Workshop on Future and Emergent Trends in Language Technology*, pages 10–33.

Russell-Rose, T. and Tate, T. (2013). *Designing the Search Experience: The Information Architecture of Discovery*. Elsevier (Morgan Kaufmann).

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343.

Tunkelang, D. J. (2009). *Faceted search*. Morgan & Claypool Publishers.

NoReC: The Norwegian Review Corpus

Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, Fredrik Jørgensen[†]
Language Technology Group, Department of Informatics, University of Oslo
{erikve, liljao, eivinabe, cathsta, samiat}@ifi.uio.no
[†]Schibsted Media Group
fredrik.jorgensen@schibsted.com

Abstract

This paper presents the Norwegian Review Corpus (NoReC), created for training and evaluating models for document-level sentiment analysis. The full-text reviews have been collected from major Norwegian news sources and cover a range of different domains, including literature, movies, video games, restaurants, music and theater, in addition to product reviews across a range of categories. Each review is labeled with a manually assigned score of 1–6, as provided by the rating of the original author. This first release of the corpus comprises more than 35,000 reviews. It is distributed using the CoNLL-U format, pre-processed using UDPipe, along with a rich set of metadata. The work reported in this paper forms part of the SANT initiative (Sentiment Analysis for Norwegian Text), a project seeking to provide open resources and tools for sentiment analysis and opinion mining for Norwegian.

Keywords: Sentiment Analysis, Opinion Mining, Corpus, Norwegian, Reviews

1. Introduction

For Norwegian, training and evaluation data are still lacking for many core NLP tasks. The current work aims to fill the gap for the particular task of sentiment analysis. The SANT project – Sentiment Analysis for Norwegian Text – seeks to create, and make publicly available, resources and tools for sentiment analysis for Norwegian. The SANT effort described in the current paper marks the release of the Norwegian Review Corpus[1] (NoReC). The dataset comprises more than 35,000 full-text reviews from a range of different domains, collected from several of the major Norwegian news sources. Each review is rated with a numerical score on a scale of 1–6, and can be used for training and evaluating models for document-level sentiment analysis, i.e., the task of predicting overall positive or negative polarity for a given text.

1.1. Rating by dice

A particularity of review journalism in Norway, is the wholesale adoption of dice rolls ('terningkast') as a standard rating scale: The item under review is rated on a scale of 1–6, commonly visualized by the face of a die with a corresponding number of 'dots' or pips. The practice is thought to have been introduced already in 1952 when reviewing movies in the newspaper *Verdens Gang (VG)*. Given that the result of a die roll is otherwise associated with randomness, it is somewhat surprising that it would catch on as a symbol for summarizing reviews – something one would typically hope to construe as a well-informed and deliberate judgment of merit and quite the opposite of chance or luck. Nontheless, by now it has found widespread use in all sorts of arts and consumer journalism and is used when reviewing everything from books, theater and music, to home electronics, restaurants, and children's clothing.
The rating practice described above has several benefits for the goal of document-level SA: (i) It eliminates the need for costly manual annotation since the numerical rating (i.e., the die roll) directly provides us with labels that can be used for training models for detecting the overall document polarity. (ii) There is no need for manually defined mappings to align different rating schemes as the reviews all use a uniform scale. (iii) The wide range of available news sources using the same rating practice, including all the major national newspapers, facilitates the creation of a large-scale dataset. (iv) Models trained on the dataset can be expected to generalize well across domains given the balance of different topics covered in the corpus.

1.2. Sources and partners

The SANT project represents a newly initiated collaboration between the Language Technology Group (LTG) at the Department of Informatics at the University of Oslo, and three of Norway's largest media groups; the Norwegian Broadcasting Corporation (NRK – the state-owned public broadcaster) and the privately held Schibsted Media Group and Aller Media. This first release (Ver. 1.0.1) of NoReC comprises 35,189 reviews extracted from eight different news sources contributed by the three media partners. In terms of publishing date the reviews mainly cover the time span 2003–2017, although it also includes a handful of reviews dating back as far as 1998. We briefly present the sources provided by the different partners below.

Schibsted Media Group The Schibsted group has contributed content from their full portfolio of Norwegian news sources: VG, Aftenposten, Fædrelandsvennen, Bergens Tidende, and Stavanger Aftenblad. While the latter three rank among Norway's largest regional newspapers, Aftenposten is the largest national newspaper in terms of circulation and VG is the largest online news source with more than 2.4 million readers across all platforms.

Aller Media The Aller publishing company has contributed content from two sources. The first is the online version of the newspaper Dagbladet – the second most vis-

[1] https://github.com/ltgoslo/norec

Source	Abbr.	# Reviews
Verdens Gang	VG	11,888
Dagbladet	DB	5300
Stavanger Aftenblad	SA	5146
P3.no	P3	5017
DinSide.no	DinSide	2944
Fædrelandsvennen	FVN	2296
Bergens Tidene	BT	1675
Aftenposten	AP	923

Table 1: Number of reviews across sources (also showing abbreviated names).

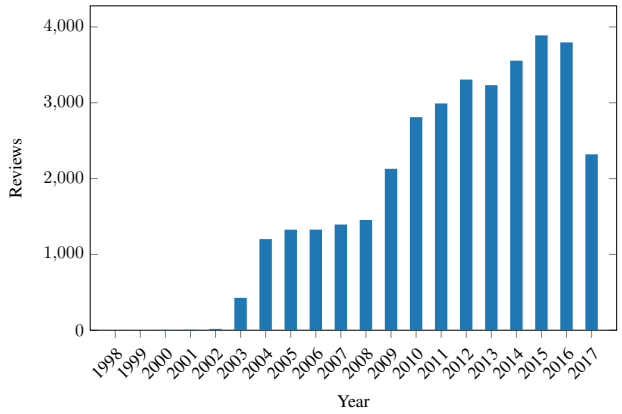


Figure 1: Number of reviews over time.

ited online news source in Norway – publishing reviews for music recordings and live performances, theater and related stage performances, movies, literature, restaurants and more. The second source, DinSide.no, is a website specializing in product reviews, covering a wide range of product types, from home electronics to cars and clothing.

NRK The Norwegian Broadcasting Corporation is a state-owned media house, with a special mandate to be a non-commercial, politically independent public broadcaster. For the review corpus, NRK has contributed content from the website P3.no which has an extensive back catalog of 'die-rated' reviews of movies, TV series, computer games, and music (both recordings and live performances).

2. Corpus creation

The original document collections were provided from the media sources in various JSON, HTML and XML formats, and a substantial effort has gone into identifying relevant documents and extracting text and associated metadata. The extraction process can be summarized by the following four steps: (i) Identify reviews, (ii) convert review content to an intermediate and canonical HTML format, (iii) extract text and pass it through linguistic pre-processing, producing representations in CoNLL-U format, and finally (iv) extract relevant metadata to a JSON representation with normalized attribute–value names. We briefly comment on each of these steps in turn below.

2.1. Identifying reviews

Some of the initial data dumps also included other articles beyond reviews, and in these cases reviews had to be identified. While in some cases this can be done simply by checking for an appropriate metadata field indicating the rating score, other cases require checking for links pointing to an image of a die (indicating the rating), or similar heuristics. Moreover, for some of the sources, a single document may contain multiple reviews, for example for product comparisons. In these cases we had to identify and separate out the different sub-reviews. Different publishing conventions require targeting different types of cues in the document structure, like headers, bold-faced content or die-face images. This also involves extraction of titles and rating scores for the different sub-reviews. The identified sub-reviews become separate documents in the NoReC data set. In total, 35,189 distinct reviews were extracted from the data provided by the media partners. Table 1 shows the number of reviews included from the various sources, while Figure 1 shows the number of reviews per year. Note that because the dataset was assembled in 2017, we only have partial data for that year. As is also evident, the corpus contains only very few reviews prior to 2003.

2.2. Converting content to canonical HTML

The raw data dumps from the sources are mostly in HTML format, but may also be e.g. JSON objects, and have different conventions for document structuring and use of markup. In order to streamline the downstream text extraction, all documents are converted to a 'canonical' HTML format where all textual content is located either inside a header or a paragraph tag. In addition to containing the review text, the raw documents also contain images, ads and other content not considered part of the running text. In order to identify and mark the non-relevant text we use a combination of heuristics based on simple string matching and properties like paragraph length and ratio of content to markup. For example, care was taken to identify 'you-might-also-be-interested-in' type links that are injected throughout the texts in an attempt to keep the reader on the website and generate more clicks. Importantly, however, we chose not to remove any content when converting to our intermediate HTML format, instead introducing a new tag – `remove` – in which we enclose content considered non-relevant. This non-destructive approach preserves the original content, as to not close the door on changes to the subsequent task of text extraction later.

2.3. Linguistic enrichments and CoNLL-U

Given the canonical HTML format described above, it is straightforward to extract the relevant text. In order to enable various types of downstream uses of the dataset, we further pre-process the raw text using the UDPipe toolkit (Straka et al., 2016), representing each review as a CoNLL-U file, following the format defined in Universal Dependencies version 2.[2] In this step we perform sentence segmentation, tokenization, lemmatization, morphological analysis, part-of-speech tagging and dependency parsing, following

[2] `http://universaldependencies.org/format.html`

the Universal Dependencies scheme (Nivre et al., 2016). However, the pre-processing set-up is slightly complicated by the fact that the Norwegian language has two official written standards – Bokmål (the main variety) and Nynorsk – both of which are represented in the review corpus. Below we first describe how language identification is performed, and then go on to give more details about UDPipe and the resulting CoNLL-U data.

Identifying language varieties The two official varieties of Norwegian are closely related and they are mostly distinguished by minor lexical differences. Still, the differences are strong enough that different pre-processing pipelines must be used for the different standards (Velldal et al., 2017), hence it is important to identify the standard used within a particular document. We use the `langid.py` (Lui and Baldwin, 2012) language identification tool to identify the standard for each review, using its pre-trained models.[3] We performed an evaluation of `langid.py` on 1599 reviews of which 1487 were written in Bokmål and 112 in Nynorsk (based on selecting reviews from authors known to write in a given variety). On this sample `langid.py` achieved 100% accuracy. While the main variety, i.e. Bokmål, dominates the distribution in the corpus with 34,656 documents, we also identified 533 documents in Nynorsk (mainly from the sources Fædrelandsvennen, Bergens Tidende and P3.no).

UDPipe configuration We apply UDPipe (Straka et al., 2016) v.1.2 with its pre-trained models for Norwegian Bokmål and Nynorsk. This version of the UDPipe software and the pre-trained models were developed for the CoNLL 2017 shared task (Zeman et al., 2017), which was devoted to parsing from raw text to Universal Dependencies for more than 40 different languages. We use the models trained for participation in the shared task (Straka and Straková, 2017), not the models provided as baseline models for the participants. The Norwegian models were trained on the UD 2.0 versions of the Norwegian UD treebanks (Øvrelid and Hohle, 2016; Velldal et al., 2017) in conjunction with the aforementioned shared task, and the subsequent choice of model (Bokmål vs Nynorsk) was determined by the language identified for each particular review.

UDPipe obtained competitive results for Norwegian in the shared task, with rankings ranging between first place (lemmatization; both variants) and ninth place (Bokmål dependency parsing LAS) out of 33 participating teams. For reference, in terms of performance for the different subtasks, UDPipe reported F1 scores – for Bokmål / Nynorsk respectively – on sentence segmentation of 96.38 / 92.08, tokenization of 99.79 / 99.93, lemmatization of 96.66 / 96.48, morphological analysis of 95.56 / 95.25, part-of-speech tagging of 96.83 / 96.54, and Labeled Accuracy Scores for dependency parsing of 83.89 / 82.74.

[3] `langid.py` can actually identify three different variants: `no`, `nn` and `nb`, for Norwegian (mixed), Nynorsk and Bokmål, respectively. While the precise details of how the classifier was trained are not clear, it appears to us after some experimentation that the classfication of Bokmål is more accurate when specyfing `no` rather than `nb` and hence is what we use here (together with `nn`). We still use the language codes *nb* and *nn* when adding information about the detected standards to the metadata in NoReC.

	#
Documents	35,189
Sentences	918,681
Tokens	14,998,667
Types – full-form	511,150
Types – lemmas	438,306
Average document length	426

Table 2: Basic corpus counts.

CoNLL-U files When extracting the text from the canonical HTML to pass it to UDPipe, we strip away all mark-up and discard all content marked for removal as described in Section 2.2. Double newlines were inserted between paragraphs and excess whitespace trimmed away. Importantly, however, the text structure is retained in CoNLL-U by taking advantage of the support for comments to mark paragraphs and sentences. In addition to the global document ID number, each paragraph and sentence is also assigned a running ID within the document, using the following form:

- Paragraphs: `<review-id>-<paragraph-id>`, e.g. `000001-03` for paragraph 3 in document 1.
- Sentences: `<review-id>-<paragraph-id>-<sentence-id>`, e.g. `000001-03-02` for sentence 2 in paragraph 3 in document 1.

After completing the UDPipe pre-processing, the corpus comprises a total of 918,681 sentences and 14,998,667 tokens; see Table 2 for an overview of some core corpus counts. A script for executing the entire pipeline from text extraction through UDPipe parsing is made available from the NoReC git repository.

2.4. Metadata and thematic categories

For all the identified reviews, we also provide various kinds of relevant metadata, made available in a JSON representation with normalized attribute–value names across reviews. Metadata extracted from the various sources include information like the URL of the originally published document, numerical rating, publishing date, author list, domain or thematic category, original ID in the source, and more. Beyond this we also add information about the identified language variety (Bokmål/Nynorsk), assigned data split (test/dev/train, as further described in Section 2.5.), assigned document ID, and finally a normalized thematic category.

Thematic categories The 'category' attribute warrants some elaboration. The use of thematic categories and/or tags varies a lot between the different sources, ranging from highly granular categories to umbrella categories encompassing many different domains. Based on the original inventory of categories, each review in NoReC is mapped to one out of nine normalized thematic categories, using English names. The distribution over categories is shown in Table 3, sorted by frequency.

For some sources, this normalization is a matter of simple one-to-one mapping, while for others it is more complex,

Category	#
screen	13,085
music	12,410
literature	3526
products	3120
games	1765
restaurants	534
stage	530
sports	117
misc	102
total	35,189

Table 3: Number of reviews across categories.

involving heuristics based on the presence of certain tags and keywords in the title. The granularity in the final set of categories is limited by the granularity in the sources. However, the original (Norwegian) source categories are preserved in a seperate attribute in the metadata ('source-category').

As seen from Table 3, the two categories that are by far the largest are 'screen' and 'music'. While the former covers reviews about movies and TV-series, the latter covers both musical recordings and performances. The related category 'stage' covers theater, opera, ballet, musical and other stage performances besides music. The perhaps most diverse category is 'products', which comprises product reviews across a number of sub-categories, ranging from cars and boats to mobile phones and home electronics, in addition to travel and more. The remaining categories of 'literature', 'games', 'restaurants', and 'sports' are self-explanatory, while the 'misc' category was included to cover topics that were infrequent or that could not easily be mapped to any of the other categories by simple heuristics.

2.5. Formats and availability

Distributed under a CC BY-NC 4.0 license,[4] NoReC is available for download from the following git repository:

`https://github.com/ltgoslo/norec`

Formats NoReC is distributed in two formats. The first is the CoNLL-U format as described in Section 2.3., containing tokenized and lemmatized text annotated with PoS tags and dependency graphs. This is considered the primary format. Secondly, we also distribute the canonical HTML representation of the 'raw' review documents as described in Section 2.2. For each format, each review is stored as a separate file, with the filename given by the review ID. To facilitate a low barrier of use for different types of end-users, we also include scripts for converting from CoNLL-U to running tokenized text (using either full-forms or lemmas) and from HTML to raw text without pre-processing. The metadata for each review is provided as a JSON object, all listed in a single file and indexed on the document IDs. The NoReC git repository also includes a Python module with basic functionality for reading the CoNLL-U and JSON representations, as to make experimentation with the corpus as accessible and convenient as possible.

[4] `https://creativecommons.org/licenses/by-nc/4.0/`

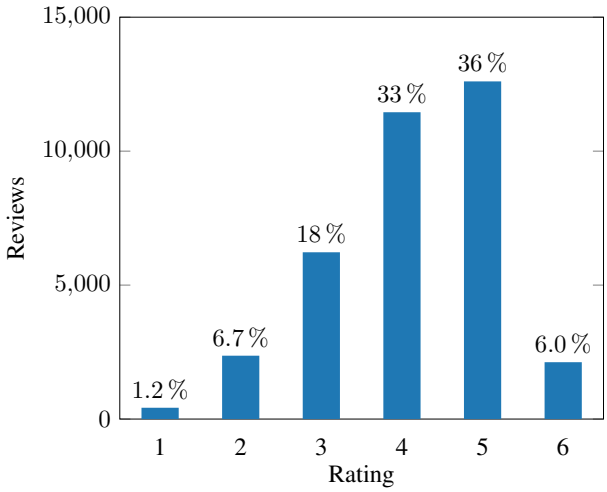


Figure 2: Number of reviews across ratings.

Train/dev/test splits To facilitate replicability of experiments the corpus comes with pre-defined standard splits for training, development and testing, with a 80–10–10 ratio. We created the splits per category – sorting the reviews for each category by date and then reserving the first 80% for training, the next 10% for development and the final 10% for testing. This strategy ensures an identical category distribution across splits, while at the same time preserving a stable distribution of ratings as well. The distribution of news source across splits will vary more, but this is less critical as we are primarily interested in ensuring a balanced distribution for ratings and categories. Defining the splits relative to the timeline also reduces the risk of having different reviews of the same item in different splits (e.g., the same product reviewed by multiple news sources). It also generally presents a more realistic test scenario: a trained model will typically be applied to fresh data, with all the time effects that this entails, like shifts in vocabulary, etc.

3. Distribution of ratings

From the perspective of SA, the most immediately relevant piece of metadata in the corpus is obviously the rating. As discussed previously, all the reviews were originally published with an integer-valued rating between 1 and 6, visually indicated using the face of a die. Figure 2 shows the distribution of reviews relative to rating scores. We see that the distribution is highly skewed, with rating values of 4 and 5 being the most common, while very few reviews are given the lowest possible rating of 1. We observe a sharp drop in frequency when moving from rating 5 to 6, perhaps indicating that the distance between these two ratings is perceived as greater than between say 4 and 5.

A similar tendency to lean towards the higher ratings is typically also reported for user-generated reviews, though with a stronger preference for the highest score (Baccianella et al., 2009). In Figures 3 and 4 we see a more detailed view of the rating distribution for each category and source.

In Figure 3 we see that the 'stage' and 'products' categories are most strongly skewed towards the rating of 5. As most of the product reviews were gathered from 'DinSide.no', we see a similar distribution for this source in Figure 4.

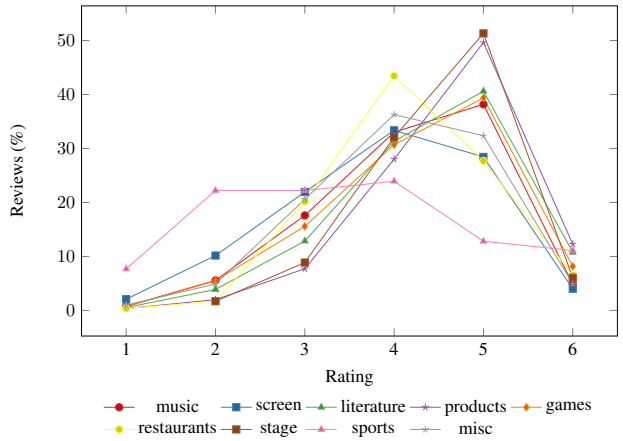


Figure 3: Distribution of ratings for each category.

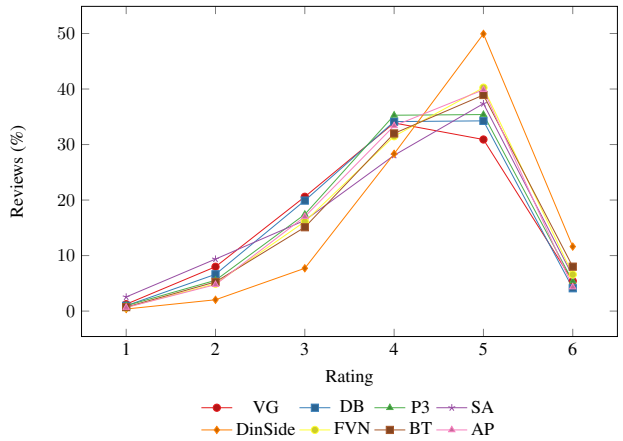


Figure 4: Distribution of ratings for each source.

Overall, however, most source and categories exhibit a similar pattern; a majority of ratings at 4 and 5, and with relatively fewer reviews in either extreme of the scale. One notable exception is the category 'sports', and to some degree also 'restaurants'.

4. Related work

The dataset described in the current paper is the first of its kind for Norwegian. For other languages, however, the field has seen a substantial amount of SA research based on rated reviews, either user-generated or by professional reviewers. This has often been based on single-domain datasets, and examples include (for English unless otherwise noted) movie reviews collected from aggregator sites like IMDb.com (Pang and Lee, 2004; Maas et al., 2011) and RottenTomatoes.com (Pang and Lee, 2005; Socher et al., 2013), hotel reviews from TripAdvisor (Wang et al., 2010), book reviews (in Arabic) (Aly and Atiya, 2013), app reviews compiled from Apple App store and Google Play (Guzman and Maalej, 2014), and reviews of restaurants and other businesses in the Yelp open dataset.[5] However, the unbalanced nature of these datasets (single domains) can impose inherent limitations on the ability of models to generalize. Some datasets combine reviews from multiple domains for better balance, like the French SA corpus of Vincent and Winterstein (2013), combining reviews of movies, books and hotels (from Allocine.fr, Amazon.fr, and TripAdvisor.fr, respectively), or the Arabic SA corpus of ElSahar and El-Beltagy (2015), combining reviews of hotels, restaurants, movies, restaurants and product reviews (from TripAdvisor, elCinema.com, Qaym.com and Souq.com). There also exists several datasets based on product reviews from Amazon, which can potentially also have the advantage of covering a more diverse selection of domains. An example includes the Amazon dataset of Blitzer et al. (2007), comprising reviews of books, DVDs, electronics, and kitchen appliances.

The NoReC dataset described in the current paper covers a wide range of domains, combining product reviews across a diverse range of categories, such as literature, restaurants, sports, music, various stage arts and more.

5. Future work

While we plan to further add more reviews to NoReC for future releases, covering additional domains or categories, the SANT project will also seek to build on NoReC to (i) experiment with both polarity classification and rating inference on the document-level using neural architectures, (ii) extract SA lexicons encoding the polarity of individual words, and finally (iii) also move beyond the document-level and manually add more fine-grained and aspect-based SA annotations for a sub-set of the corpus. These annotations will also be used to (iv) train a classifier separating subjective and objective sentences. Across all these activities, the various thematic categories will be useful for assessing cross-domain effects (e.g., how well does an SA classifier trained on movie reviews perform for home electronics?) and potentially even for training domain-specific models. It will of course also be important to assess how well SA resources developed on the basis of the reviews generalize to non-review texts, and we plan to annotate aspect-based sentiment for a selection of general-domain news texts as well.

6. Summary and outlook

The current paper has described the creation of the Norwegian Review Corpus; NoReC (Ver. 1.0.1). The final dataset comprises more than 35,000 full-text reviews ($\approx$ 15 million tokens) from a wide range of different domains, collected from several major Norwegian news sources. Each review is rated with a numerical score on a scale of 1–6, and can be used for training and evaluating models for document-level sentiment analysis. Resources for sentiment analysis have so far been unavailable for Norwegian. While the primary distribution format of the corpus is CoNLL-U – based on only the extracted text and applying UDPipe for a full preprocessing pipeline from sentence segmentation to dependency parsing – the release also includes HTML representations of the full reviews with all content preserved. Each review is in addition associated with a rich set of metadata, including thematic category. We also provide pre-defined splits for training, development and testing.

[5] https://www.yelp.com/dataset

7. Acknowledgments

This work was carried out as part of the SANT research project (Sentiment Analysis for Norwegian Text), funded by an IKTPLUSS grant from the Research Council of Norway (project no. 270908). SANT represents a newly initiated collaboration between the Department of Informatics at the University of Oslo, and three of Norway's largest media groups; the Norwegian Broadcasting Corporation, Schibsted Media Group and Aller Media.

8. Bibliographical references

Aly, M. and Atiya, A. (2013). LABR: A large scale arabic book reviews dataset. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, volume 2 (short papers), pages 494–498, Sofia, Bulgaria.

Baccianella, S., Esuli, A., and Sebastiani, F. (2009). Evaluation measures for ordinal regression. In *Proceedings of the 9th International Conference on Intelligent Systems Design and Applications*, pages 283–287, Pisa, Italy.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Meeting of the Association for Computational Linguistics*, pages 187–205, Prague, Czech Republic.

ElSahar, H. and El-Beltagy, S. R. (2015). Building large arabic multi-domain resources for sentiment analysis. In *Proceedings of the 16th International Conference on Intelligent Text Processing and Computational Linguistics*, pages 23–34, Cairo, Egypt.

Guzman, E. and Maalej, W. (2014). How do users like this feature? A fine grained sentiment analysis of app reviews. In *Proceedings of the 22nd IEEE International Requirements Engineering Conference*, pages 153–162, Karlskrona, Sweden.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Meeting of the Association for Computational Linguistics System Demonstrations*, pages 25–30, Jeju, Republic of Korea.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Meeting of the Association for Computational Linguistics*, pages 142–150, Portland, OR, USA.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1659–1666, Portorož, Slovenia.

Øvrelid, L. and Hohle, P. (2016). Norwegian Universal Dependencies. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1579–1585, Portorož, Slovenia.

Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pages 271–278, Barcelona, Spain.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Meeting of the Association for Computational Linguistics*, pages 115–124, Ann Arbor, MI, USA.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, WA, USA.

Straka, M. and Straková, J. (2017). Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada.

Straka, M., Hajič, J., and Straková, J. (2016). UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 4290–4297, Portorož, Slovenia.

Velldal, E., Øvrelid, L., and Hohle, P. (2017). Joint UD parsing of Norwegian Bokmål and Nynorsk. In *Proceedings of the 11th Nordic Conference of Computational Linguistics*, pages 1–10, Gothenburg, Sweden.

Vincent, M. and Winterstein, G. (2013). Construction et exploitation d'un corpus français pour l'analyse de sentiment (building and exploiting a French corpus for sentiment analysis). In *Proceedings of the 20th Conference Traitement Automatique des Langues Naturelles*, volume 2 (short papers), pages 764–771, Les Sables-d'Olonne.

Wang, H., Lu, Y., and Zhai, C. (2010). Latent aspect rating analysis on review text data: A rating regression approach. In *Proceedings of the 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 783–792, Washington DC, DC, USA.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, c., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada.

SenSALDO: Creating a Sentiment Lexicon for Swedish

Jacobo Rouces, Nina Tahmasebi, Lars Borin, Stian Rødven Eide
Språkbanken, University of Gothenburg
{jacobo.rouces, nina.tahmasebi, lars.borin, stian.rodven.eide}@gu.se

Abstract

The natural language processing subfield known as *sentiment analysis* or *opinion mining* has seen an explosive expansion over the last decade or so, and sentiment analysis has become a standard item in the NLP toolbox. Still, many theoretical and methodological questions remain unanswered and resource gaps unfilled. Most work on automated sentiment analysis has been done on English and a few other languages; for most written languages of the world, this tool is not available. This paper describes the development of an extensive sentiment lexicon for written (standard) Swedish. We investigate different methods for developing a sentiment lexicon for Swedish. We use an existing gold standard dataset for training and testing. For each word sense from the SALDO Swedish lexicon, we assign a real value sentiment score in the range [-1,1] and produce a sentiment label. We implement and evaluate three methods: a graph-based method that iterates over the SALDO structure, a method based on random paths over the SALDO structure and a corpus-driven method based on word embeddings. The resulting sense-disambiguated sentiment lexicon (SenSALDO) is an open source resource and freely available from Språkbanken, The Swedish Language Bank at the University of Gothenburg.

Keywords: sentiment analysis, Swedish, lexicon, lexical resource

1. Introduction

The natural language processing (NLP) subfield known as *sentiment analysis* or *opinion mining* has seen an explosive expansion over the last decade or so. Since the publication of the comprehensive overview of the field by Pang and Lee (2008), we have seen hundreds of papers as well as dedicated workshops on this topic in NLP conferences.

Even though sentiment analysis has become a standard implement in the NLP toolbox, many theoretical and methodological questions remain unanswered and resource gaps need to be filled. Most work on automated sentiment analysis has been done on English and a few other languages; for most of the written languages of the world,[1] this tool is not available. This paper describes the development of an extensive sentiment lexicon for written (standard) Swedish, an essential component in sentiment analysis.

The theoretical and methodological issues that arise in connection with sentiment analysis of texts lie partly in the intersection of the linguistic subfields of pragmatics and lexical semantics. Depending on your view of the scope of these subdisciplines, you may end up with very different thoughts about the prior-polarity – i.e., lexical-semantic – and contextual – i.e., pragmatic – elements of sentiment information, and how these are combined in concrete text analysis. An added complication is that sentiments and emotions are central objects of study also in other fields, notably psychology.

In practice this means that we find many different proposals in the literature, for how prior sentiment polarity should be represented in the lexicon, which kinds of lexical entities should be attached (lemmas, lexemes or word senses), and how contextual information is to be encoded and used when calculating the sentiment of a text passage from its constituent parts.

The methodological position taken in this paper is, in brief, that prior sentiment polarity forms part of a word's sense, and that a word sense only has one prior polarity. In our case the polarity is expressed as a real number in the range $[-1, 1]$, with higher positive values associated to more positive sentiments. Connotations are considered to form part of the word sense (as opposed to, e.g., the practice in WordNet). From this it follows that, if a word appears in text with two different sentiment values, it must either represent two senses of this lexeme or, alternatively, reflect a contextual effect.

The focus on word senses as bearers of prior polarity is in line with our general view on lexical-semantic resources for NLP, where the word sense takes center stage.[2] Thus, our point of departure in this paper is the Swedish SALDO lexical resource (Språkbanken, 2015b). SALDO is an onomasiological lexicon, i.e., organized by content (lexical entries are word senses), rather than by form (lemmas or lexemes). For a detailed description of the organization of SALDO and a discussion of the underlying theoretical and methodological principles, we refer the reader to Borin et al. (2013).

However, one aspect of SALDO's organization will be important in the context of what follows below, namely the basic lexical-semantic relations defining the network structure of SALDO, which provide important information for creating SenSALDO. It is superficially similar to WordNet, but quite different from it in the principles by which it is structured. The basic organizational principle of SALDO is hierarchical. Every entry in SALDO – representing a word sense[3] – is supplied with one or more semantic descriptors,

[1] According to a standard reference, *Ethnologue* (Simons and Fennig, 2017), there are about 7,000 spoken languages in the world. A fair estimate would be that at the most 1,000 of these have a tradition of writing (Borin, 2009). Sentiment analysis tools are available for far fewer languages than this.

[2] Notably, our use of *word sense* is to be construed as 'lexical word sense', which also is intended to cover lexicalized multiword expressions.

[3] Each word sense in SALDO is additionally connected to one or more form units (lemmas plus part of speech and full inflec-

which are themselves also entries in the dictionary. All entries in SALDO are actually occurring words or conventionalized or lexicalized multi-word expressions (MWEs) of the language. The primary – obligatory – descriptor is the entry which better than any other entry fulfills two requirements: (1) it is a close semantic neighbor of the entry to be described; and (2) it is more central than it.

In defining SALDO, criterion (1), semantic neighborhood, is understood as involving a direct semantic relationship between lexical items,[4] for instance synonymy, hyponymy, argument–predicate relationship, etc. Since there can be only one primary descriptor for any given entry in SALDO, one of these relationships must be chosen in every case, but this will not necessarily be the same. The predecessor of SALDO was characterized as an "associative thesaurus" (Lönngren, 1998), and its basic structure can still be said to be 'micro-thesaural', i.e., more similar to what we find in Roget's *Thesaurus* (Roget, 1852) or its Swedish counterpart by Bring (Bring, 1930; Borin et al., 2014; Borin et al., 2015) (Språkbanken, 2015a; Språkbanken, 2017a) rather than the straightforward taxonomical structure exemplified by WordNet.

As for criterion (2), centrality is determined by means of several criteria, e.g., stylistic value, word-formation complexity, the type of semantic relation holding between an entry and its primary descriptor, acquisition order in first-language acquisition, etc. In practice, however, frequency is among the best predictors of centrality even when the latter has been determined by these other means. It turns out that about 90% of the SALDO entries have primary descriptors which are at least as frequent as the entries themselves in a corpus of more than one billion words of Swedish.

Since the primary descriptor must be another actual lexical entry, in reality SALDO forms 40-some different hierarchies, where no more suitable primary descriptor can be found.[5] Here, an artificial lexical item (called *PRIM*) is used in order to make a single rooted tree for the primary-descriptor relation.

In addition to the obligatory primary descriptor, any number of secondary decsriptors can be added, whose main purpose is to disambiguate or further group entries with the same primary descriptor. Their usage is much more pragmatic and less consistent than in the case of the primary descriptors.

The lexical-semantic organization of SALDO is predicated on the idea of the vocabulary of a language having a core part and a more peripheral part. Consequently, the higher levels in the lexical-semantic hierarchy of SALDO contain simpler and more basic entries. Contrast this with WordNet, where the higher nodes in the hierarchy contain very abstract vocabulary (e.g. 'entity').

2. State of the Art

Many methods have been developed and tested for building sentiment lexicons, English being the most popular language for these. The methods may rely on corpus analysis (making use of word co-occurrence, syntactic patterns, or distant-supervision signals) or on existing lexicons (usually using some sort of label propagation exploiting the structure of the lexicon), although both approaches can be combined (Devitt and Ahmad, 2013; Hamilton et al., 2016). The different methods can also involve a varying degree of manual annotation.

Among the English lexicons built with mostly-automatic lexicon-driven methods, SentiWordNet (Baccianella et al., 2010) has become a popular resource. It is created by combining a semi-supervised learning step that uses existing relations between WordNet 3.0 entries (Fellbaum, 1998), (such as *synonymy*, *antonymy*, and *related with*), and a random-walk step over a graph built using the implicit *definiens-definiendum* relation between words in the entries and words in the glosses of the entries (Esuli and Sebastiani, 2007). However, these relations require WordNet, or an equivalent lexicon, which in turn requires a big amount of manual work by trained lexicographers. Therefore, this kind of approach has severe limitations for languages with fewer resources than English.

Among the English lexicons using corpus-driven approaches, SENTPROP (Hamilton et al., 2016) is a recent state-of-the-art approach that builds a directed weighted graph of terms using the nearest neighbors in the space of word embeddings obtained from applying singular value decomposition to the positive pointwise mutual information matrix obtained from the corpus. Then, it uses random walks in a similar fashion to SentiWordNet.

Given a set of labeled training words annotated as positive and negative, Rothe et al. (2016) find an orthogonal transformation of the embedding space that maximizes the distance among those with different labels and minimizes the distance among those with the same label.

Amir et al. (2015) train different linear regression models (least squares and regularized variants) over different word embeddings (GloVe, CBOW, skip-gram, struct skip-gram) **add something like: comparable to our word2vec method**. Bar-Haim et al. (2017) expand an already existing sentiment lexicon by training a linear SVM. They obtain an accuracy of 90.5%. Both of these methods are applied exclusively on English and produce sentiment labels for non-disambiguated lemmas.

For Swedish, two openly available sentiment lexicons exist (Nusko et al., 2016; Rosell and Kann, 2010). In addition, there are some Swedish sentiment lexicons or word lists produced by automatic translation of corresponding English resources, e.g., by Mohammad and Turney (2010)[6] and Chen and Skiena (2014).

Rosell and Kann (2010) developed a Swedish sentiment lexicon using random walks over a graph of synonyms and

tional and compounding information). These are formally organized as an independent lexical resource – *SALDO's Morphology* (Språkbanken, 2015c) – which consequently can be used in NLP applications independently of SALDO, e.g., for lemmatization and morphological analysis of Swedish text.

[4] SALDO contains all parts of speech, not only the open lexical classes. Thus the noun *yta* 'surface' has as its primary descriptor the preposition *på* 'on'.

[5] For instance, the preposition *på* 'on' has no primary descriptor.

[6] http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

4 positive and 4 negative seed words. The graph was built using the *Synlex/People's Dictionary of Synonyms* (Kann and Rosell, 2005), which used Swedish-English lemma pairs concatenated with their inverse relation to generate candidate synonym pairs. The pairs were filtered by grading and then averaging the grades. The result of Synlex was 16006 words with 18920 weighted pairs, which were used as edges of the graph in the random walks. The random walk algorithm is described in Algorithm 1 (with some notational changes from Kann and Rosell (2005) to describe our extensions later in Section 3.).

Algorithm 1 Random walk Algorithm

Require: Similarity weighted directed graph G with set W words as vertices and weighted edges defined by similarity function $0 \leq \text{sim}(w_1, w_2) \leq 1$.

function $\text{prob}(w_1, w_2) = \frac{\text{sim}(w_1,w_2)}{\sum_{w_3 \in W} \text{sim}(w_1,w_3)}$
for all word w **do**
  for all $i \in [0, 1, ..., I]$ **do** ▷ $I = 100$
    $v_+ \leftarrow 0$ and $v_- \leftarrow 0$ and $w_{\text{now}} \leftarrow w$ and $l \leftarrow 0$
    for all $j \in [0, 1, ..., M]$ **do** ▷ $M = 250$
      sample w_{next} with $\text{prob}(w_{\text{now}}, w_{\text{next}})$
      $w_{\text{now}} \leftarrow w_{\text{next}}$
      $l \leftarrow 1/\text{sim}(w_{\text{now}}, w_{\text{next}})$
      if $w_{\text{now}} \in S_+$ **then**
        $v_+ \leftarrow v_+ + m/l$
      if $w_{\text{now}} \in S_-$ **then**
        $v_- \leftarrow v_- + m/l$
  sentiment$(w) \leftarrow \frac{v_+ - v_-}{I}$

Nusko et al. (2016) propose a tree traversal method on the tree defined by the primary descriptor relation between SALDO entries. This method starts with 6 seeds with a manually assigned polarity and recursively calculates the sentiment of children based on the sentiment of the parent.[7] The algorithm calculates a confidence score for each sentiment, which decreases at a constant rate from the distance to the original seed (steps of -0.25 from a confidence of 1 for the descendants of the core words), and sets a threshold of 0.5 as the lowest acceptable confidence. It also uses secondary descriptors, but only when the secondary descriptor is *inte* (Swedish negation 'not'), which indicates that the child and parent have opposite semantic values and therefore the sign on the sentiment value should also be inverted, or a strength modifier like *lite* 'a little', or *enastående* 'outstanding'. It obtains a sentiment for 2133 entries. Three annotators labeled 150 entries as positive, negative or neutral, and for 117 of the entries the annotators were in full agreement. From these a 71% precision was obtained. The original 150 entries were sampled using equally sized stratification over the three confidence levels.

3. Methods

We model the sentiment associated to a word sense using a real value in the interval $[-1, 1]$. After first considering using a three-dimensional model like that of SentiWordNet (Baccianella et al., 2010), we collected some experimental evidence indicating that this was largely unnecessary since the additional degree of freedom was all but unused in practice (Rouces et al., 2018).

We implement different methods, which we describe below, extending the methods in Rosell and Kann (2010) and Nusko et al. (2016) and also try a corpus-oriented approach similar to the one in Hamilton et al. (2016). For all methods, we produce continuous scores and discrete labels (positive, neutral, negative). What is relevant about the scores is not their magnitudes but the relative order that they produce. The values and their distributions depend on idiosyncrasies of the methods employed and do not necessarily resemble what would be produced by direct human annotations, but instead can be fit to any desired distribution. The discrete labels are less fine-grained, but may be more appropriate for certain applications.

3.1. Inheritance over Graph

Our first method is a modified and extended version of the tree traversal method presented by Nusko et al. (2016), where sentiment of a word sense is inherited from the primary descriptor (which defines a tree structure). We extended it such that the traversal occurs over the directed acyclic graph defined by using both primary and secondary descriptors. The secondary descriptors of an entry are used not only for polarity inversion or intensification, but their sentiment value is also used, although with a lower weight.

The algorithm cannot use a simple breadth-first exploration, because for a given node, in general, some incoming neighbors will be at a different distance from the seed set than others, and the node will be reached before all the incoming neighbors have been calculated. This prevents all elements in the frontier to be expanded in a single iteration.

In addition, even when attempting different partially-successful passes over the frontier, the algorithm would stagnate easily because some secondary descriptors are not reachable from the given seed words. For this reason, the algorithm incorporates a best-effort mechanism for stagnation cases, whereby a sentiment is calculated for a node with the lowest possible number of unreached secondary-descriptor incoming nodes, ignoring these, and a new pass is performed over the frontier. Primary descriptors are never ignored. A priority queue is used for the nodes with unreached secondary-descriptor incoming nodes. If it is still not possible to calculate new nodes from all their parents, the process is repeated until it is possible, or the queue and the frontier are empty.

We used the same seeds as Nusko et al. (2016).

The method outputs sentiment scores for each sense, so in order to obtain discrete labels we apply thresholds. The thresholds are obtained from the percentiles of each class in a training set obtained from sampling two thirds of the gold standard described in section 4.. For example, the positive class in the gold standard corresponds to 6.34%, so we classify the 6.34% most positive senses as *positive*. The other third is used for testing. This is the only learning needed by this algorithm.

[7] In Nusko et al. (2016) the seeds and their children are referred as "core words" and "seeds" respectively.

3.2. Random Paths over Graphs

Our second method is an adaptation of the method developed by Rosell and Kann (2010) and presented in the algorithm in Algorithm 1. Our adaptation allows it to be used on SALDO sense-disambiguated entries instead of Swedish lemmas, and extends the edges of graph G (by creating new non-zero sim() relations) in order to make it more dense and prevent the existence of isolated nodes, which would be too common using only Synlex (which has $131,020$ nodes connected by only $25,516$ relations).
For this, we use the union of several graphs.

- A manual conversion of Synlex to SALDO. Rather than the original Synlex, which is a crowd-sourced resource with many questionable entries, and which is not word-sense disambiguated, we use a manually curated version which forms part of an emerging Swedish enriched wordnet, *Swesaurus* (Språkbanken, 2017b),[8] where an experienced lexicographer has (i) removed incorrect entries and (ii) replaced each remaining entry with its corresponding SALDO word sense. In some cases the degree of synonymy (the weight) has also been modified. We use the original weights in the $[0, 1]$ interval.
- The edges defined by primary descriptors in SALDO.
- The edges defined by secondary descriptors in SALDO.
- The edges that connect SALDO entries that have the same primary descriptor.

Since the new graphs are unweighted, their edges were assigned a weight of 0.5. This is a simple heuristic that reflects that they represent a certain level of similarity, but not the level of synonymy represented by a value of 1 in Synlex.

3.3. Classification over word2vec

As opposed to the previous methods, which are purely lexicon-driven, this approach is partly corpus-based. We used already existing vector representations of SALDO entries (which are sense-disambiguated) that were derived from *word2vec* lemma embeddings (Johansson and Nieto Piña, 2015). This was done by solving a constrained optimization problem where each lemma embedding is a linear combination of the embeddings of the senses associated to that lemma, and the distance between neighboring senses (i.e. neighbors in SALDO's descriptor graph) is minimized. The corpus size was 1 billion words, and the vector space dimensionality was 512. We trained a logistic regression (logit) classifier and a support vector classifier with a radial basis function (RBF) kernel. All the classifiers used a one-vs-rest approach of the three-class classification. For the classifiers we used 5-fold cross-validation stratified by the (pos,neu,neg) classes. For each fold, the SVM/RBF meta-parameters (C, γ) were estimated using 5-fold cross-validation over the training set. Although not equivalent, the linear nature of the logit classifier makes it comparable to the method in Rothe et al. (2016).
These methods output labels, but scores are obtained computing $p((pos) - p((neg))$, where p is the probability for a given entry to belong to the positive or negative classes. For the logit classifier, p is straightforward. For the support vector classifier, an extension of Platt scaling for multiple classes is used (Wu et al., 2004).

[8] Swesaurus contains information from several sources, but the curated Synlex data can be extracted from the LMF XML file by finding `SenseRelation` elements (inside `Sense` elements) containing `<feat att="source" val="fsl" />`.

4. Results

For training and testing the different methods, we used the direct annotation gold standard developed in Rouces et al. (2018), composed of 1998 entries from SALDO entries labeled as negative (value -1), neutral (value 0), or positive (value $+1$). The values were averaged over three annotators (so if an entry is labeled as positive by two annotators and as neutral by one, the final value would be $2/3$).
Table 1 shows the results for each method. We employed two different sets of measures for measuring the quality of the gold standard: one is based on ranks and other is based on discrete labels.

- The rank-based measures are the Spearman rank-order correlation coefficient (ρ) (Kokoska and Zwillinger, 2000), in the interval $[-1, 1]$, the p-normalized Kendall tau distance (τ_p) (Fagin et al., 2004) in the interval $[0, 1]$ (the one used in Baccianella et al. (2010)), and Kendall's tau-b (τ_b) (Kendall, 1945) (the one used in Rothe et al. (2016)). Both τ_p and τ_b are suited to handle ties —which in our case means word senses with equal sentiment values— but they do so in different ways. In addition to the direct annotation values in the test set, we also used more fine-grained sentiment values of 278 entries that are available as part of the same gold standard (Rouces et al., 2018), which were obtained using Best-Worst Scaling (BWS) and also comprised in the $[-1, 1]$ range. The reason for this is that these values are more fine-grained than the Direct Annotation (DA) values (which due to the use of 3 annotators range over only 7 possible values), and therefore ties are less common in the gold standard, making some ranking comparison algorithms more suitable. Since the BWS values were created only for the entries annotated as non-neutral by the DA scoring ($|\text{value}| \geq 0.5$), they cannot all be used for testing (or else the training set would be too biased towards neutral elements). Therefore, the intersection of the DA test set and the entries with BWS value is used for applying the rank-based measures.
- The measures based on discrete labels are the precision and recall values for each label, derived from the confusion matrix, which is also included.

'Graph inheritance' corresponds to the method in Nusko et al. (2016), although the results are not completely equivalent because of the stratification used in the evaluation in Nusko et al. (2016). 'Graph inheritance ext' corresponds to the extended version described in Section 3.1.. 'Graph

	DA									BWS
	ρ	τ_p	τ_b	precision	recall	acc.	confusion matrix			τ_b
							GS	SL		
								pos	neu	neg	
graph inheritance	0.39	0.39	0.38	pos: 0.28 neu: 0.91 neg: 0.33	pos: 0.26 neu: 0.90 neg: 0.42	0.82	pos neu neg	10 23 3	28 391 12	1 21 11	0.49
graph inheritance ext	0.33	0.42	0.32	pos: 0.22 neu: 0.90 neg: 0.27	pos: 0.21 neu: 0.89 neg: 0.35	0.81	pos neu neg	8 26 2	30 386 15	1 23 9	0.46
graph random paths	0.30	0.31	0.24	pos: 0.25 neu: 0.90 neg: 0.39	pos: 0.23 neu: 0.90 neg: 0.50	0.82	pos neu neg	9 26 1	29 390 12	1 19 13	0.46
word2vec +logit	0.47	0.21	0.38	pos: 0.37 neu: 0.93 neg: 0.46	pos: 0.54 neu: 0.88 neg: 0.52	0.84	pos neu neg	15 25 1	13 301 11	0 15 13	0.61
word2vec +svc /rbf	**0.55**	**0.15**	**0.45**	pos: 0.65 neu: 0.92 neg: 0.65	pos: 0.46 neu: 0.96 neg: 0.44	**0.89**	pos neu neg	13 7 0	15 328 14	0 6 11	**0.62**

Table 1: Results for evaluating the different methods for constructing the sentiment lexicon in Swedish. Note that the Kendall tau τ_p is a distance, and therefore it is inversely related to the Spearman correlation ρ. GS and SL stand for gold standard and sentiment lexicon respectively.

random paths' corresponds to the method described in Section 3.2.. The last three rows correspond to the results of the different classifiers used in Section 3.3..

word sense ID	gloss	value	label
ond..4	'bad'	-0.9959	neg
farlig..1	'dangerous'	-0.9919	neg
kriminalitet..1	'criminality'	-0.9838	neg
skrämma..1	'frighten'	-0.9797	neg
problem..1	'problem'	-0.9716	neg
angrepp..1	'attack'	-0.9594	neg
förhållande..1	'relationship'	-0.0345	neu
radio..1	'radio'	-0.0264	neu
sälja..1	'sell'	-0.0223	neu
surdeg..1	'sourdough'	0.0426	neu
god..2	'tasty'	0.9675	pos
riktig..2	'genuine'	0.9716	pos
hjälpa..1	'help (v)'	0.9797	pos

Table 2: Examples of sentiment values and labels obtained with the word2vec-svc-rbf method. The values have been fitted to the uniform distribution in $[-1, +1]$

The extension of the original graph inheritance method by using all the secondary descriptors in SALDO slightly reduces the quality of the results, which seems to indicate that the semantic connection behind the secondary descriptors in general is too weak and not useful for this task.

The method word2vec-svc-rbf performs consistently better than the rest. Table 2 shows some examples obtained from this method. SentiWordNet is reported to have τ_p values of 0.281 and 0.231 for positive and negative dimensions (their sentiment model has 2 degrees of freedom). All our embeddings-based methods outperform both measures (τ_p is a distance, and therefore lower values are desired). Rothe et al. (2016) reports $\tau_b = 0.654$. We obtain $\tau_b = 0.45$ when testing against the DA values, which is significantly lower. However, this probably owes to τ_b penalizing the big amount of ties in the DA values (61.95% of the possible pairs), as the method obtains $\tau_b = 0.63$ (a very close value) when testing against the BWS values, where ties are much less common (0.63%).

The resulting sentiment lexicon – SenSALDO (Språkbanken, 2018) is an open-source resource and freely available from Språkbanken, The Swedish Language Bank at the University of Gothenburg. In a companion publication (Rouces et al., forthcoming), we discuss applications of this resource in text mining for digital humanities research.

Acknowledgements

This work has been supported by a framework grant (*Towards a knowledge-based culturomics*[9] – contract 2012-5738) as well as funding to Swedish CLARIN (*SweClarin*[10] – contract 2013-2003), both awarded by the Swedish Research Council, and by infrastructure funding granted to Språkbanken by the University of Gothenburg.

5. Bibliographical References

Amir, S., Astudillo, R., Ling, W., Martins, B., Silva, M. J., and Trancoso, I. (2015). Inesc-id: A regression model for large scale twitter sentiment lexicon induction. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 613–618.

[9] https://spraakbanken.gu.se/eng/culturomics

[10] https://sweclarin.se/eng

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of LREC*, pages 2200–2204.

Bar-Haim, R., Edelstein, L., Jochim, C., and Slonim, N. (2017). Improving claim stance classification with lexical knowledge expansion and context utilization. In *Proceedings of the 4th Workshop on Argument Mining*, pages 32–38.

Borin, L., Forsberg, M., and Lönngren, L. (2013). SALDO: A touch of yin to WordNet's yang. *Language Resources and Evaluation*, 47(4):1191–1211.

Borin, L., Allwood, J., and de Melo, G. (2014). Bring vs. MTRoget: Evaluating automatic thesaurus translation. In *Proceedings of LREC 2014*, pages 2115–2121, Reykjavík. ELRA.

Borin, L., Nieto Piña, L., and de Johansson, R. (2015). Here be dragons? The perils and promises of inter-resource lexical-semantic mapping. In *Semantic resources and semantic annotation for Natural Language Processing and the Digital Humanities. Workshop at NODALIDA 2015*, pages 1–11, Linköping. Linköping Electronic Conference Proceedings.

Borin, L. (2009). Linguistic diversity in the information society. In *Proceedings of the SALTMIL 2009 workshop on Information Retrieval and Information Extraction for Less Resourced Languages*, pages 1–7, Donostia. University of the Basque Country.

Bring, S. C. (1930). *Svenskt ordförråd ordnat i begreppsklasser*. Hugo Gebers förlag, Stockholm.

Chen, Y. and Skiena, S. (2014). Building sentiment lexicons for all major languages. In *Proceedings of ACL 2014*, pages 383–389, Baltimore. ACL.

Devitt, A. and Ahmad, K. (2013). Is there a language of sentiment? An analysis of lexical resources for sentiment analysis. *Language Resources and Evaluation*, 47(4):475–511.

Esuli, A. and Sebastiani, F. (2007). Random-walk models of term semantics: An application to opinion-related properties. *Proceedings of LTC 2007*, pages 221–225.

Fagin, R., Kumar, R., Mahdian, M., Sivakumar, D., and Vee, E. (2004). Comparing and aggregating rankings with ties. In *Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '04, pages 47–58, New York. ACM.

Fellbaum, C. (ed). (1998). *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, Mass.

Hamilton, W. L., Clark, K., Leskovec, J., and Jurafsky, D. (2016). Inducing domain-specific sentiment lexicons from unlabeled corpora. *arXiv preprint arXiv:1606.02820*.

Johansson, R. and Nieto Piña, L. (2015). Embedding a semantic network in a word space. In *Proceedings of NAACL-HLT 2015*, pages 1428–1433, Denver. ACL.

Kann, V. and Rosell, M. (2005). Free construction of a free Swedish dictionary of synonyms. In *Proceedings of NODALIDA 2010*, Joensuu. University of Eastern Finland.

Kendall, M. G. (1945). The treatment of ties in ranking problems. *Biometrika*, pages 239–251.

Kokoska, S. and Zwillinger, D. (2000). *Standard Probability and Statistics Tables and Formulae*. Chapman & Hall / CRC.

Lönngren, L. (1998). A Swedish associative thesaurus. In *Euralex '98 proceedings, Vol. 2*, pages 467–474, Liège. EURALEX.

Mohammad, S. and Turney, P. (2010). Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles. ACL.

Nusko, B., Tahmasebi, N., and Mogren, O. (2016). Building a sentiment lexicon for Swedish. In *Proceedings of the* From Digitization to Knowledge *workshop at DH 2016, Kraków*, pages 32–37, Linköping. LiUEP.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

Roget, M. P. (1852). *Thesaurus of English Words and Phrases*. Longman, London.

Rosell, M. and Kann, V. (2010). Constructing a Swedish general purpose polarity lexicon: Random walks in the People's dictionary of synonyms. In *Proceedings of SLTC 2010*, pages 19–20, Stockholm. KTH.

Rothe, S., Ebert, S., and Schütze, H. (2016). Ultradense word embeddings by orthogonal transformation. *arXiv preprint arXiv:1602.07572*.

Rouces, J., Tahmasebi, N., Borin, L., and Eide, S. R. (2018). Generating a gold standard for a Swedish sentiment lexicon. In *Proceedings of LREC 2018*, Miyazaki. ELRA.

Rouces, J., Borin, L., Tahmasebi, N., and Eide, S. R. (forthcoming). Defining a gold standard for a Swedish sentiment lexicon: Towards higher-yield text mining in the digital humanities. In *Proceedings of DHN 2018, Helsinki*, Aachen. CEUR WS.

Simons, G. F. and Fennig, C. D. (eds). (2017). *Ethnologue: Languages of the world*. SIL International, Dallas, 20th edition. Online version: `http://www.ethnologue.com`.

Wu, T.-F., Lin, C.-J., and Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5(Aug):975–1005.

6. Language Resource References

Språkbanken. (2015a). *Bring*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: Bring, v. 1.1. A Swedish version of Roget's Thesaurus from 1930 (∼149k entries), URL `https://spraakbanken.gu.se/eng/resource/bring`.

Språkbanken. (2015b). *SALDO*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: SALDO, v. 2.3. A lexical-semantic resource for modern Swedish (∼130k entries), URL `https://spraakbanken.gu.se/eng/resource/saldo`.

Språkbanken. (2015c). *SALDO's Morphology*. Språkbanken, University of Gothenburg, Sweden, Språk-

banken lexical resources: SALDO's Morphology, v. 2.3. A morphological lexicon (full-form and inflectional paradigms) for modern Swedish (∼128k entries), URL https://spraakbanken.gu.se/eng/resource/saldom.

Språkbanken. (2017a). *Bring*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: Blingbring, v. 0.3. A Swedish version of Roget's Thesaurus from 1930, where outdated entries have been removed and the (lemma) entries have been partially linked to SALDO word senses (∼149k entries), URL https://spraakbanken.gu.se/eng/resource/blingbring.

Språkbanken. (2017b). *Swesaurus*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: Swesaurus, v. 0.2. A Swedish wordnet combining curated information from a number of free lexical resources (∼72k entries), URL https://spraakbanken.gu.se/eng/resource/swesaurus.

Språkbanken. (2018). *SenSALDO*. Språkbanken, University of Gothenburg, Sweden, Språkbanken lexical resources: SenSALDO, v. 0.1. A word-sense prior-polarity sentiment lexicon for modern Swedish (∼7.6k entries), URL https://spraakbanken.gu.se/eng/resource/sensaldo http://hdl.handle.net/10794/sensaldo.

Corpus Building and Evaluation of Aspect-based Opinion Summaries from Tweets in Spanish

Daniel Peñaloza, Rodrigo López, Juanjosé Tenorio, Héctor Gómez, Arturo Oncevay-Marcos, Marco A. Sobrevilla Cabezudo
Research Group on Artificial Intelligence (IA-PUCP)
Departamento de Ingeniería, Pontificia Universidad Católica del Perú, Lima, Peru
{a20110645,a20112387,juanjose.tenorio,hector.gomez,arturo.oncevay,msobrevilla}@pucp.edu.pe

Abstract

This project involves the presentation and analysis of a corpus of Spanish extractive and abstractive summaries of opinions. The purpose of this work is to display a corpus of diverse summaries that could be used as a reference for academic research as we have not found one for the Spanish language as far as we know. We have analyzed the summaries based on the agreement between them as this shows how different they are written between each other and on aspect coverage and sentiment orientation as this proves the difference between the content that each summary tries to express. After the experimentation, we have found that even if each annotator uses a different expression to summarize a text, all of them contain similar messages. Furthermore, when writing, all of them prioritize on common aspects that are more representative of the corpus.

Keywords: aspect-based opinion summarization, corpora building, Spanish

1. Introduction

Automatic summarization is one of the most challenging problems in Natural Language Processing (NLP). The task is even more difficult if the summarized content is generated entirely by the program, and it does not present any extract of the original text. Theres are two kinds of summaries: abstractive and extractive (Labbé and Portet, 2012). In extractive summarization, we try to select the principal ideas or sentences and grouping them in the summary. In abstractive summarization, we try to write a text which contains all ideas.

The relevance of the summarization task increases if the context of the high variety and volume of information is considered. Social networks, such as micro-blogging sites, nowadays are offering the possibility to exploit any kind of publications, where user opinions are greatly valuable. Opinion mining (OM) is the NLP subfield advocated to this problem. The main goal is to extract a subjective value from the text, that might include a certain polarity degree, and it could be focused on a particular entity or aspect (Liu, 2012).

Therefore, it is likely to be a special difficulty in summarizing user opinions from social media, as there is not an specific order among all the opinion posts grouped by the same topic. Besides, each post may address a different aspect or entity related to the main topic, so there is a need to identify them first. After that, the subjectivity of each aspect must be measured, such as a positive or negative polarity (Wu et al., 2016). There is also a quantitative scope by considering the distribution of different positions regarding an opinion, as no one should be treated as more relevant than others (Liu, 2012).

Aspect-based opinion summarization is an attractive task in this context. However, the primary research is centralized in some specific languages or domains (mainly English), due to the lack of summaries corpus to evaluate the process in different ones. For that reason, this study proposes the development of a new opinion summaries corpus for aspect-based opinion mining in Spanish. The documents/posts are extracted from Twitter, and they are related to the political context in Spain. In addition, there is an evaluation performed over the new corpus. It is important to highlight this corpus will be available publicly for anyone to do further researches.

The paper is organized as follows: Section 2 describes the related work for building of opinion summaries corpora. Then, the corpus named STOMPOL is presented in Section 3. After that, the annotation process for extractive and abstractive summaries is detailed in Section 4, while the corpus analysis is reported and analyzed in Section 5. Finally, conclusions and future work are discussed.

2. Related Work

There are several works about Opinion Summarization for English. One problem is that every work used its own corpora to do their experiments.

The dataset proposed in (Hu and Liu, 2004) is the principal resource in aspect-based opinion summarization. Nevertheless, that corpus did not contain manual summaries, but aspects annotated and their associated sentiment.

Another works where opinion summaries were built are presented in (Tadano et al., 2010) (related to videogames), (Xu et al., 2011) (related to restaurants) and (Carenini et al., 2006) (related to digital camera and DVD player). They generated 25, 30 and 28 summaries respectively.

To do a quantitative analysis of the summaries, Tadano (Tadano et al., 2010) utilized ROUGE-1 to compare them as it indicates an n-gram recall between summaries. We have to highlight these works did not present any kind of qualitative analysis of opinion summaries based on aspects.

For romance languages, we have found one work about corpora building. This work was proposed in (López et al., 2015). In this paper, authors built a corpora of abstractive and extractive opinion summaries for Brazilian Portuguese

and did a qualitative analysis by comparing the aspect coverage between summaries because his generated summaries covered many aspects of the same entity. Equally important, They made a sentiment orientation analysis as them must maintain the same sentiment expressed of the entity that the reference text had.
For Spanish, as far as we know, there is only one work about opinion summarization. In this work, authors built opinion summaries for Tourism sector (Esteban and Lloret, 2017) and then they implemented an abstractive opinion summarization method.

3. STOMPOL Corpus

The STOMPOL Corpus is one the corpora proposed since the TASS 2015 (Cámara et al., 2016). This corpus have been used in the task of polarity classification at aspect level. The STOMPOL corpus is a corpus composed by Spanish tweets related to a political aspects that appeared in the Spanish political campaign of regional and local elections that were held on 2015.
Each tweet presents one or more of the following aspects in its content: Economics, Health System, Education, Political party and other aspects. Also, each aspect is related to one or more of these sentiments: positive, negative and neutral (according the presented aspect).
In general, these corpus is composed by 784 tweets about 6 political parties. In Table 1 are presented the distribution of tweets by political party in STOMPOL corpus. We have to note that one tweet may contain comments about one or more political parties.

Political Party	Number of tweets
Ciudadanos	135
Izquierda Unida	118
Partido Popular	240
Partido Socialista Obrero Español	159
Podemos	114
Unión, Progreso y Democracia	103

Table 1: STOMPOL Corpus

4. Annotation Process

The main goal of this work was to create reference extractive and abstractive summaries in order to support future works about opinion summarization.
Due to the large number of tweets in STOMPOL corpus and the difficulty to read all of them to generate a summary, we chose to extract a few number of tweets for each political party. Table 2 shows the number of tweets considered for each one.
One point to highlight is the different number of tweets for each political party. It happened because we tried to select a sample which represented the real distribution of aspects in the STOMPOL corpus due to the imbalanced of corpus at aspect-level and, also, tried to cover all aspects.
In general, we created several reference summaries (abstractive and extractive) in order to reduce any possible bias and to understand the opinion summary generation in a better way. For each political party, we generated 3 extractive and 3 abstractive summaries, building 36 opinion

Political Party	Number of tweets
Ciudadanos	36
Izquierda Unida	26
Partido Popular	27
Partido Socialista Obrero Español	26
Podemos	30
Unión, Progreso y Democracia	33

Table 2: Sub corpus of STOMPOL Corpus

summaries in total. Although the number of summaries is small, it is enough for an analysis as seen on the related works. Table 3 shows the statistics of the corpus generated, in terms of number of summaries, sentences, tokens and its average.

	Abstractive	Extractive
Number of summaries	18	18
Number of sentences	77	130
Number of tokens	1864	1859
Average sentences by summary	4.27	7.22
Average tokens by summary	103.56	103.28

Table 3: Statistics of Corpus

Both kinds of summaries were handcrafted by 4 annotators. Each annotator had to generate one abstractive summary and one extractive summary for each political party. This task had a duration of 1 week and a half.
In order to improve to annotation process, we established some guidelines in relation to how to build the summary, the aspects coverage and the summary length.
In relation to how to build the summary, we suggested to the annotator to read all tweets and try to understand the principal ideas. Then, they had to build the summaries oriented to aspects.
In relation to aspects coverage, we provided a list of aspects (listed in Section 3) and the possible words related to them in order to facilitate the annotation process.
Finally, in relation to the summary length, we opted for generating summaries which contained between 90 and 110 words. We did not choose a compression rate because it is not used in the literature for this kind of task (López et al., 2015).

4.1. Extractive Summaries

In order to facilitate the building of extractive summaries, we had to do one previous step. This consisted in distinguish all tweets and the sentences in them. To do this, we assigned an identifier for each tweet of a political party. Then, we manually separated the sentences in the tweet and assigned an identifier to them too. We may see an example of extractive summary about the political party Podemos in Figure 1. In this Figure we may see the identifier <D5_S1>(first line), where D5 represents the tweet number 5 and S1 represents the sentence number 1 of that tweet. After this previous step, the annotators had to select some sentences which represented relevant sentimental information about each political party.
One point to highlight in Figure 1 is that extractive summary is composed by seven sentences from different tweets (D5, D32, D8, D10, D17, D16 and D9). This indicates that

@lematelot49 @ahorapodemos llevo toda mi vida acompañado de esa palabra "crisis" para que me vengan con milongas la OCDE <D5_S1>
@Pablo_Iglesias_ @CorruptionTimes @flameycitron y dicen que no hay dinero para los enfermos de Hepatitis C <D32_S1>
@la_tuerka @Pablo_Iglesias_ dice conocer la realidad social de los más necesitados pero calla ante asesinato de miles de cristianos por ISIS <D8_S1>
¿Perdiendo votos antes de empezar? <D10_S2>
@ahorapodemos pues aunque parezca increíble, han vivido a costa de las minorías. <D17_S1>
@ahorapodemos @ierrejon que poca humanidad y poca verguenza <D16_S1>
@andresysara10 en #España ???? , por desgracia , para tener políticos honrados , debemos esperar a q nazcan . <D9_S1>

Figure 1: Example of extractive summary

important sentences for annotators were written by different users. Finally, we may see the lack of coherence in this kind of summary.

4.2. Abstractive Summaries

In the case of abstractive summaries, the annotators had to understand the overall opinion and write the summary in their own words. The summary should contain the main aspects and to detail (if it is possible) the reasons which motivate each aspect-level sentiment.
We may see an example of abstractive summary about the political party Podemos in Figure 2.

En lo que se refiere a economía, se hace mención a una crisis posiblemente asociada con el partido. Mientras que lo que respecta a sanidad hay pacientes que no pueden ser atendidos en los hospitales. Además de esto, cuando se habla sobre el partido la gente expresa abiertamente su descontento con los políticos y con el partido en sí, ya sea por sus posiciones u opiniones en diversos temas, o también debido a que tan involucrados están en temas de corrupción. Este partido empieza a perder apoyo puesto que ocultan información y su pasado es bastante turbio.

Figure 2: Example of abstractive summary

5. Corpus Analysis

The analysis of the corpus consisted of three experiments. First, we checked the inter-annotator agreement between summaries to see how similar they are based on the words that each annotator used. Second, we checked the aspect coverage of each summary to see how related they are based on what elements are each summary talking about. Finally, we checked the sentiment orientation of each summary to see how comparable they are based on the feeling that each annotator wanted to express.

Summary	Abstractive	Extractive
Ciudadanos	0.310	0.692
Izquierda Unida	0.351	0.649
Partido Popular	0.282	0.568
Partido Socialista Obrero Español	0.311	0.609
Podemos	0.293	0.606
Unión Progreso y Democracia	0.270	0.587

Table 4: ROUGE metric of the summaries

Summary	No Agreement	Agreement	Total Agreement
Ciudadanos	0.818	0.000	0.090
Izquierda Unida	0.615	0.307	0.076
Partido Popular	0.789	0.105	0.105
Partido Socialista Obrero Español	0.705	0.235	0.058
Podemos	0.733	0.133	0.133
Union Progreso y Democracia	0.700	0.300	0.000

Table 5: Sentence Agreement on Extractive Summaries

The experiments were done to prove three affirmations. (i) There is not a perfect summary because each person has a different way of expressing themselves, (ii) there are aspects that are more relevant to the annotator when writing a summary and (iii) each annotator adds a sentiment to the summary they want to write.

5.1. Inter-Annotator Agreement

In order to obtain the inter-annotator agreement we used the ROUGE metric. The measure counts the number of overlapping words between the generated summary and a reference summary (Lin, 2004) For our experiment we used Rouge-1, this compares single words between summaries and is expressed as a recall. We used this metric to compare all the summaries of an specific entity with each other for both the extractive summaries and the abstractive summaries. The results of these experiments are in Table 4.
On every case, extractive summaries had better results than abstractive summaries. This is because on abstractive summaries, the annotators have the freedom to use any word they want and the metric compares that the words used between them are exactly the same.
Additionally, we have also used the tags of the extractive summaries to obtain if the annotators are using the same sentences. This can be seen in Table 5. The results indicate that even when having the option to use the same words for the summary, the annotators do not do that except for some important sentences that are common between them.

5.2. Aspect Coverage

As we have seen on the results of the inter-Annotator Agreement, the annotators usually use different sentences in order to express themselves. For that reason, we have also reviewed the aspects of the entity that have been referenced by the annotators as they express more the ideas that the annotators wanted to include on their summaries.
For the extractive summaries we used the tags of the STOMPOL corpus that indicate which aspect has been referenced for a sentence. Furthermore, a sentence can reference multiple times the same aspect so we had only taken into consideration which one of the aspects are being referenced not the frequency of repetition.
For the abstractive summaries we did a manual review of each summary. Each annotator used their own words to

Summary	Extractive	Abstractive
Ciudadanos	**0.40**	**0.33**
Ciudadanos 1	0.40	0.40
Ciudadanos 2	0.40	0.40
Ciudadanos 3	0.40	0.20
Izquierda Unida	**0.93**	**0.46**
Izquierda Unida 1	1.00	0.60
Izquierda Unida 2	1.00	0.40
Izquierda Unida 3	0.80	0.40
Partido Popular	**0.66**	**0.60**
Partido Popular 1	0.60	0.60
Partido Popular 2	0.80	0.60
Partido Popular 3	0.60	0.60
Partido Socialista Obrero Español	**0.53**	**0.46**
Partido Socialista Obrero Español 1	0.40	0.40
Partido Socialista Obrero Español 2	0.60	0.40
Partido Socialista Obrero Español 3	0.60	0.60
Podemos	**0.66**	**0.60**
Podemos 1	0.60	0.60
Podemos 2	0.80	0.60
Podemos 3	0.60	0.60
Union Progreso y Democracia	**0.66**	**0.66**
Union Progreso y Democracia 1	0.80	0.80
Union Progreso y Democracia 2	0.80	0.60
Union Progreso y Democracia 3	0.40	0.60

Table 6: Aspect coverage of each summary

express themselves so in these case it is not as simple as looking for the mentions of the aspect. We had to read each sentence to find when the annotators used synonyms to talk about the aspects.

In Table 6, it may be seen the percentages of how much of every summary covers the aspects of a political party. For example, if a summary talks about the political party itself and the economy plan then it covers two aspects. As we are interested in five specific aspects then the summary will cover 40 percent. Here we may see that even if the average aspect coverage of an entity is low, it is because the annotators had decided to cover on a specific set of aspects. This is revealed by the aspect coverage of each annotator summary as usually they only differ by a small margin, meaning that there is an agreement on which aspects are more relevant to include on the summary.

On the other hand, we may see that usually extractive and abstractive summaries have similar results on the aspect coverage. This happens because the annotators try to express the same ideas for both summaries even if they can not use their own words. On the other hand, there was a case where the extractive summary covers more than the abstractive summary. This happens because when writing with his own words, the annotator has the possibility to focus on what is more relevant to him instead of using the sentence that seems better to him even if that sentence cover aspects than aren't important to the annotator.

5.3. Sentiment Orientation

Even though we have covered the elements that annotators where writing about, that is not complete enough as people express themselves with a sentiment in mind that is reflected on the way each annotator communicates and the words they use. Consequently, we had to analyze the sentiment orientation of each summary in order to get the complete meaning of each summary. For these analysis we are considering that the sentences expressed on each summary are positive or negative as these are the most general sentiments that can be associated to a sentence.

Regarding the aspect coverage, we used the tags on the STOMPOL corpus that associates some words on a sentence to a polarity to analyze the extractive summaries. As we had only referenced if a sentence is positive or negative without taking into account the intensity of how it is expressed, we have not taken into consideration if a sentence has multiple occurrences of the same polarity on different words and only considered as it is positive or negative.

On the other hand, for the abstractive summaries we used Spanish Sentistrength (López et al., 2012) for the analysis. This is a dictionary of common words used on the Spanish language where each word is associated to a polarity.

The results of this experiment are presented on the Table 7. In particular, extractive summaries show disproportionate results compared to the actual polarity of the corpus. Conversely, abstractive summaries show better results. This happens because when doing extractive summaries, the annotator is limited by the sentences that they can use and gives more preference to covering the more relevant aspects instead of trying to express their opinion.

6. Conclusions and Future Work

In this paper, we presented a corpus of summaries of Spanish tweets related to political aspects. Additionally, we presented an analysis of this corpus to detail its contents. From the inter-annotator agreement we have demonstrated that the summaries are different from each other on the sentences they use to express their message. However, from the aspect coverage we can say that even though the summaries are different, they are about similar elements of the entity. Notably, the words used on the summary also change the message that tries to express as demonstrated on the sentiment orientation where the freedom of a annotator to use their own words give the abstractive summaries better results.

For future works, we will give more emphasis on the comparison between the summaries and the corpus as it can reveal more details on the distribution of the summaries as displayed on the sentiment orientation where the results of the abstractive summaries are more similar to the corpus than the extractive. Also, we might take into consideration the order in which the summaries are generated as writing first the extractive summaries could influence in the annotators to use the same words in the abstractive summaries.

Another essential point is to improve the tools used to measure the summaries. For example, we may find a sentiment dictionary that is more suited to the theme of the corpus that will obtain better results. In the same way, we can take into consideration the intensity of the polarity expressed on the sentences to get a more real grasp of what they are trying to express.

Finally, the corpus will be available in the following link: https://github.com/iapucp/spop-summ-lrec2018

Summary	Actual Polarity		Extractive		Abstractive	
	Positive	Negative	Positive	Negative	Positive	Negative
Ciudadanos	0.363	0.636	0.666	0.333	0.366	0.633
Izquierda Unida	0.347	0.652	0.300	0.700	0.253	0.746
Partido Popular	0.162	0.837	0.500	0.500	0.138	0.861
Partido Socialista Obrero Español	0.279	0.720	0.435	0.564	0.172	0.827
Podemos	0.207	0.792	0.500	0.500	0.228	0.771
Unión Progreso y Democracia	0.365	0.634	0.611	0.388	0.441	0.558

Table 7: Sentiment Orientation

7. Bibliographical References

Carenini, G., Ng, R., and Pauls, A. (2006). Multi-document summarization of evaluative text. In *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 305–312.

Cámara, E. M., García-Cumbreras, M. A., Villena-Román, J., and García-Morera, J. (2016). TASS 2015 - The evolution of the Spanish opinion mining systems. *Procesamiento del Lenguaje Natural*, 56:33–40.

Esteban, A. and Lloret, E. (2017). Travelsum: A spanish summarization application focused on the tourism sector. *Procesamiento del Lenguaje Natural*, 59:159–162.

Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd international conference on computational linguistics*, pages 340–348. Association for Computational Linguistics.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Labbé, C. and Portet, F. (2012). Towards an abstractive opinion summarisation of multiple reviews in the tourism domain. In *The First International Workshop on Sentiment Discovery from Affective Data (SDAD 2012)*, pages 87–94.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

López, R., Tejada, J., and Thelwall, M. (2012). Spanish sentistrength as a tool for opinion mining peruvian facebook and twitter. *Artificial Intelligence Driven Solutions to Business and Engineering Problems*, pages 82–85.

López, R. E., Avanço, L. V., Filho, P. P., Bokan, A. Y., Cardoso, M. S., Nóbrega, F. A., Cabezudo, M. A., Jackson, W., Souza, A. C., Seno, E. M., et al. (2015). A qualitative analysis of a corpus of opinion summaries based on aspects. In *The 9th Linguistic Annotation Workshop held in conjuncion with NAACL 2015*, pages 62–71. Association for Computational Linguistics.

Tadano, R., Shimada, K., and Endo, T. (2010). Multi-aspects review summarization based on identification of important opinions and their similarity. In *PACLIC*, pages 685–692.

Wu, H., Gu, Y., Sun, S., and Gu, X. (2016). Aspect-based opinion summarization with convolutional neural networks. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 3157–3163. IEEE.

Xu, X., Meng, T., and Cheng, X. (2011). Aspect-based extractive summarization of online reviews. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, pages 968–975. ACM.

Application and Analysis of a Multi-layered Scheme for Irony on the Italian Twitter Corpus TWITTIRÒ

Alessandra Teresa Cignarella[1,2], **Cristina Bosco**[1], **Viviana Patti**[1], **Mirko Lai**[1,2]
[1]Dipartimento di Informatica, Università degli Studi di Torino, Italy
[2]PRHLT Research Center, Universitat Politècnica de València, Spain
{cigna,bosco,patti}@di.unito.it
mirko.lai@unito.it

Abstract

In this paper we describe the main issues emerged within the application of a multi-layered scheme for the fine-grained annotation of irony (Karoui et al., 2017) on an Italian Twitter corpus, i.e. TWITTIRÒ, which is composed of about 1,500 tweets with various provenance. A discussion is proposed about the limits and advantages of the application of the scheme to Italian messages, supported by an analysis of the outcome of the annotation carried on by native Italian speakers in the development of the corpus. We present a quantitative and qualitative study both of the distribution of the labels for the different layers involved in the scheme which can shed some light on the process of human annotation for a validation of the annotation scheme on Italian irony-laden social media contents collected in the last years. This results in a novel gold standard for irony detection in Italian, enriched with fine-grained annotations, and in a language resource available to the community and exploitable in the cross- and multi-lingual perspective which characterizes the work that inspired this research.

Keywords: irony, figurative language processing, corpora, social media, Italian

1. Introduction

The automatic recognition of irony is, still nowadays, a challenging task to be performed both by human annotators and automatic NLP systems (Mihalcea and Pulman, 2007; Reyes et al., 2010; Kouloumpis et al., 2011; Maynard and Funk, 2011; Reyes et al., 2012; Hernández Farías et al., 2016; Sulis et al., 2016). The growing interest on this task is attested by the proposal of shared tasks focusing on irony detection and its impact on sentiment analysis in social media, in the context of periodical evaluation campaigns for NLP tools for many languages, see for instance the pilot task on irony detection proposed for Italian in *Sentipolc@Evalita*, in the 2014 and 2016 editions (Basile et al., 2014; Barbieri et al., 2016) and the battery of related tasks proposed for French at *DEFT@TALN2017* (Benamara et al., 2017). For what concerns English, after a first interesting task at *SemEval-2015* (i.e. *Task 11*) focusing on *Sentiment Analysis of Figurative Language in Twitter* (Ghosh et al., 2015), in 2018 a shared task on irony detection in tweets has been proposed for the first time (*SemEval-2018 Task 3: Irony detection in English tweets*)[1]. In the latter, the organizers propose not only the classical binary classification task, where the systems must determine whether a tweet is ironic or not, but also a fine grained multiclass classification task on different types of irony, where the systems must predict one out of four labels describing: i) verbal irony realized through a polarity contrast, ii) verbal irony without such a polarity contrast, iii) descriptions of situational irony, and iv) non-irony (Van Hee, 2017; Van Hee et al., in press 2018). The setting proposed for the Semeval-2018 is an indication of the growing interest for a deeper analysis of the linguistic phenomena underlying ironic expressions. Such kind of deeper analysis naturally calls for the definition and the exploitation of schemes allowing the annotation of finer-grained features and resources in order to hopefully improve the performance of automatic systems in this especially challenging task.

This work aims at the creation of a novel resource for irony detection in the Italian language called TWITTIRÒ. We considered as starting point for this work the scheme provided in (Karoui et al., 2017), which was initially applied to a set of 400 Italian tweets. In particular, we would like to highlight how the complexity of a pragmatic device such as irony, also attested in literature (Grice, 1975; Grice, 1978; Sperber and Wilson, 1981; Wilson and Sperber, 2007; Reyes et al., 2010; Fink et al., 2011; Reyes et al., 2012), makes the annotation task particularly challenging, as will be discussed in a deep analysis of the disagreement between the native Italian speakers involved in the development of the resource. Human annotators, even skilled or domain experts, are indeed always connected to their individual experience, their individual sense of humor and a certain situational context. Nevertheless, even if they are biased, humans can easily detect the presence of irony when it occurs. Our investigation concerns the linguistic devices known in pragmatics as signals of irony and their relevance for modeling irony in a computational perspective.

As we will explore in detail in the following section, the TWITTIRÒ corpus consists of three sub-corpora characterized by linguistic differences and peculiarities. With the description and analysis of the current release of the TWITTIRÒ corpus we aim at providing a deeper investigation of the issues that arose with the application of the scheme to Italian irony-laden texts, which has been preliminary investigated in (Cignarella et al., 2017; Karoui et al., 2017). Since the resulting annotated corpus will be exploited as reference dataset within the context of the next Evalita evaluation campaign[2], it will be made available to the community and exploitable in the cross- and multi-lingual perspec-

[1]https://competitions.codalab.org/competitions/17468

[2]http://www.evalita.it/2018

tive depicted in (Karoui et al., 2017) from the end of 2018[3]. The paper is organized as follows: in the next sections we describe the dataset on which we worked, focusing on the collection, the annotation process and the annotation scheme. Section 4. is centered on the analysis of the disagreement detected during annotation and presents some hints about the quantitative results. Finally we show, in Section 5., a selection of cases especially difficult to deal with our annotation scheme.

2. Building the Corpus

This project aims at developing an Italian Twitter corpus, to be used as language resource in the training of NLP tools and to become a benchmark in evaluation campaigns for this language, for what concerns irony detection. Since a preliminary resource composed of 400 Italian tweets was available, where a very interesting scheme for describing irony at a fine-grain level (Karoui et al., 2017) has been applied, we considered it as a starting point for our work. In order to extend the corpus, we collected new data (i.e. 1,200 tweets), whose balancing is coherent with that applied in this small existing corpus, and we applied the same scheme in order to build TWITTIRÒ, which thus now includes 1,600 tweets as shown in Table 1[4].

In this section we describe the methodology applied in the collection of these new tweets, and the internal structure of the novel dataset TWITTIRÒ. Some Italian corpora containing Twitter data, where the presence of irony is marked, have been made available to the community in the last few years, thus we extracted from them the new 1,200 tweets to be included in TWITTIRÒ. In particular, the tweets were collected from the following three different pre-existent datasets: TW-SPINO, TW-SENTIPOLC14 and TW-BS.

TW-SPINO is a portion of SENTITUT (Bosco et al., 2013) which contains tweets collected from the satirical blog *Spinoza.it*. The language used is grammatically correct featured by a high register and style, while the topics are variegate with a preference for jokes concerning politics and news.

TW-SENTIPOLC14 (Basile et al., 2014) contains tweets generated by common Twitter users and therefore it is less homogeneous than TW-SPINO. The use of grammar is sometimes very poor, colloquial expressions and vulgarities typical of Computer-Mediated Communication (CMC) appear, such as the frequent use of creative hashtags, mentions, repetitions of laughters. We selected here the political tweets with reference to the government of Mario Monti between 2011 and 2012.

TW-BS (Stranisci et al., 2016) contains tweets on the debate of the reform of Italian School "Buona Scuola". Similarly to TW-SENTIPOLC14, also here devices typically exploited in CMC are shown. For instance, being the reform of the education system a highly criticized one, the use of sentences written in ALL CAPS (to decode shouting) is wide. Part of this corpus has been included in the test set within the Sentipolc 2016 evaluation campaign (Barbieri et al., 2016).

Table 1 shows the composition of TWITTIRÒ and the distribution of tweets over the three sub-corpora.

TW-SPINO	TW-SENTIPOLC14	TW-BS
400	**600**	**600**

Table 1: Tweet distribution in the TWITTIRÒ corpus

The original Italian resource described in (Karoui et al., 2017) was part of a project for studying irony in a multilingual perspective and including also similar English and French datasets annotated with the same scheme. As for what concerns the French and English datasets, tweets were retrieved by using Twitter APIs and filtered through specific *hashtags* exploited by users to self-mark their ironic intention (*#irony*, *#sarcasm*, *#sarcastic*). Providing that Italian users exploit a series of humorous hashtags, but no long-term single hashtag is established and shared among them, the same procedure could not be applied.

In the following section we focus on the novel 1,200 tweets only, provided that the collection, annotation and disagreement analysis of the previously developed 400 tweets corpus has been discussed in (Karoui et al., 2017) within the context of multilingual experiments. Also for what concerns the details and guidelines of the annotation schema applied in TWITTIRÒ, we refer to the same paper and to (Cignarella et al., 2017), where we discussed a preliminary stage of development of the novel resource.

3. Annotation Process

The annotation process of the 1,200 tweets corpus, coherently of what was done in (Karoui et al., 2017), involved three people previously trained in similar tasks: all tweets were tagged by two independent annotators (A1 and A2) and by a third (A3) only for the tweets where a disagreement was detected.

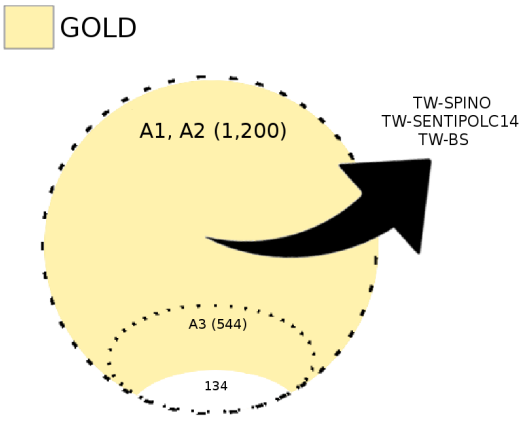


Figure 1: Portions of different agreement level and annotator's contribution on TWITTIRÒ data

[3] https://github.com/IronyAndTweets/Scheme

[4] Considering that the complexity of tasks related to the detection of pragmatic phenomena does not only depend on the inner structure of irony, but also on unbalanced data distribution, we built the novel resource by maintaining the same proportions considered in the first collection of 400 Italian ironic tweets described in (Karoui et al., 2017).

Figure 1 shows the portions of TWITTIRÒ that were annotated, and how many annotators give their contribute in the annotation for achieving the agreement (or not achieve it). As it can be seen, both annotators A1 and A2 annotated all the corpus, but, after their work, the disagreement analysis shows that they gave different annotations on 544 tweets. Therefore, annotator A3 provided a further annotation for these tweets, allowing the achievement of the agreement for 410 additional tweets. The last 134 tweets remain in disagreement and they were, then, discarded.

3.1. Annotation Scheme and Examples

The multi-layer scheme described in (Karoui et al., 2017) includes 4 different levels. Provided that Level 1 concerns the classification of tweets into **ironic** or **not ironic** which has been previously applied on the data we collected for our corpus (that are all ironic), we don't discuss its application in this paper. The three human annotators A1, A2, and A3, indeed, worked on the application of tags concerning Level 2 and Level 3.

LEVEL 2: IRONY ACTIVATION. As far as Level 2 is concerned, for each ironic tweet the annotators decided whether the type of contradiction that activates irony was EXPLICIT or IMPLICIT, that is determined by a contradiction between two items directly cited within the message, or by a contradiction between a directly cited item and some other things in the external context.

LEVEL 3: IRONY CATEGORIES. Both explicit and implicit activation types can be expressed in different ways. At this level, annotators were asked to classify the linguistic device triggering irony by applying one category tag from the following list: ANALOGY, EUPHEMISM, HYPERBOLE/EXAGGERATION, CONTEXT SHIFT (explicit only), OXYMORON/PARADOX (explicit only), FALSE ASSERTION (implicit only), RHETORICAL QUESTION or OTHER (humor or situational irony). In Table 2 we provide a brief description for each category, while here below we also discuss some examples, in order to better clarify the application of the scheme to our Italian social media texts.

1. ANALOGY
   Leo Messi: "Firmo quello che mi dice papà". Pure la Boschi.
   → *Leo Messi: "I sign what daddy tells me". Also Minister Boschi.*

The analogy here links two figures: the footballer Lionel Messi and the Italian Minister Boschi. The figure of speech is referred to the fact that the world-known footballer once affirmed that, if his father tells him to sign something, he would do it without hesitation. In particular, the athlete has signed (apparently without knowing) some contracts regarding the rights on his public image, and money has been transferred in fiscal paradises such as Uruguay and Belize. The second element of the tweet is Maria Elena Boschi, whose dad's shady affairs and implications with the bankruptcy of *Banca Etruria* are still nowadays elements of tension and discussion.

2. HYPERBOLE
   #M5S #Renzi, se tra un anno non ci saranno 170 mila insegnanti di ruolo in più, te li porto tutti a Palazzo_Chigi #labuonascuola.
   → *#M5S #Renzi, if in one year from now there will not be 170,000 teachers more, I will bring them all to Palazzo_Chigi #labuonascuola.*

While reforming Italian School, Prime Minister Renzi promised the opening of 170,000 new job places for teachers. The exaggeration in the tweet is referred to the fact that, the user states that if this should not happen, he will drag all those unoccupied workers to Palazzo Chigi in Rome, where Italian Prime Ministers normally live.

3. EUPHEMISM
   Nel 2006 Charlie Hebdo aveva pubblicato delle vignette satiriche su Maometto. Ci hanno messo un po' a capirle. [nicodio]
   → *In 2006 Charlie Hebdo had published some satirical images on Mohamad. It took them a while to understand them.*

CATEGORY	DESCRIPTION
ANALOGY	In this category are summoned *analogy*, and also other figures of speech that comprehend mechanisms of comparison, such as *simile* and *metaphor*.
EUPHEMISM	It is a figure of speech which is used to reduce the facts of an expression or an idea considered unpleasant in order to soften the reality.
CONTEXT SHIFT	It occurs by the sudden change of the topic/frame in the tweet.
FALSE ASSERTION	Indicates that a proposition, fact or an assertion fails to make sense against the reality. The speaker expresses the opposite of what he thinks or something wrong with respect to a context. External knowledge is fundamental to understand the irony (it is, in fact, implicit only).
HYPERBOLE	It is a figure of speech which consists in expressing an idea or a feeling with an exaggerated way.
OXYMORON / PARADOX	This category is equivalent to the category FALSE ASSERTION except that the contradiction is explicit.
RHETORICAL QUESTION	It is a figure of speech in the form of a question asked in order to make a point rather than to elicit an answer.
OTHER	This last category represents ironic tweets, which can not be classified under one of the other seven previous categories. It can occur in case of humor or situational irony.

Table 2: Description of categories

The tweet is a reference to the terrorist attack that took place at the offices of the French satirical weekly newspaper *Charlie Hebdo* on January 7, 2015. The second part of the tweet states: "It took them a while to understand them", it is an euphemism implying that the attack of 2015 is a consequence of a satirical comic strip published in 2006 about Mohamed, from the French newspaper.

4. CONTEXT SHIFT
   L'auto di Salvini assalita al campo rom. Rovinato il safari. [@paniruro]
   → *The car of Salvini assaulted at the Roma camp. The Safari is ruined. [@paniruro]*

The tweet points at news from November 8, 2014 in which Matteo Salvini, Secretary of the Italian right party *Lega Nord*, visited a Roma camp in Bologna, and his car has been assaulted, punched and kicked by youths of left-wing associations. The implicit connection of the user is that the young rebels behaved as aggressively as fierce animals normally do during a safari trip, the context is therefore, shifted.

5. FALSE ASSERTION
   Brunetta sostiene di tornare a fare l'economista, Mario Monti terrorizzato progetta di mollare tutto ed aprire un negozio di pescheria.
   → *Brunetta affirms that he will go back to be an economist, Mario Monti plans of leaving everything and opening a fish monger's.*

The false assertion is referred to many affirmations of the politician Renato Brunetta within the past years "*I am more rigorous then Tremonti [...] I know well all this topics, because I AM an economist, Tremonti isn't.*". Several times Brunetta publicly discredited other economists and colleagues such as Tremonti or Monti, who are instead his peers.

6. OXYMORON / PARADOX
   "Potrei non opporre veti a un presidente del Pd", ha detto Berlusconi iscrivendosi al Pd.
   → *"I could not deny rights to a President of the PD", said Berlusconi while subscribing to the party.*

A tweet in which Silvio Berlusconi (center-right politician) declares that he will subscribe to the PD (center-left party) is clearly a paradox, but the user is subtly making a reference to the blurry ideology of the leftist party, which, since a couple of years seems more a right-centered party. Hence, not so different from the berlusconian party: FORZA ITALIA.

7. RHETORICAL QUESTION
   Mario Monti? non era il nome di un antipasto? #FullMonti #laresadeiconti #elezioni #308.
   → *Mario Monti? Wasn't it the the name of an entree? #FullMonti #finalcountdown #elections #308*

The tweet contains a rhetorical question based on a pun that associates the name of the premier Mario Monti and a common pizza flavor named *Mari e Monti* (seas and mountains), in which normally you can find seafood combined with mushrooms. Other typical elements of social media texts are present, such as the humorous hashtag *#FullMonty*[5], and the hashtag *#308* which is referred to the number of deputies who voted "yes" to the new harsh financial law proposed by Monti.

8. OTHER
   Sicilia, arriva barcone di migranti e a bordo c'è anche un gatto. Vengono a rubarci i nostri like.
   → *Sicily, a big boat full of migrants arrives and there's also a kitty on board. They come here and steal our likes.*

The tweet regards a news fact[6] and it has been labeled as OTHER because of the presence of an overlapping of more than one category. Firstly, the commonplace on immigrants "They come here and steal our jobs" has been mutated in the era of social networks, in "They come here and steal our likes". The joke is based on the implicit knowledge of the Internet-world that videos and pictures containing kitties receive tons of likes from users. All this adds up to a paradox, because thinking that a tragic situation as the arrival of migrants on boats is compared at the pursuit of likes on the net, is just dramatic.

Finally, let us notice that the annotation scheme provided in (Karoui et al., 2017) also includes **LEVEL 4**, which is referred to an even finer-grained annotation of irony and takes into account the presence of several **clues** such as punctuation negation words, emoticons, punctuation marks, interjections, named entities (and mentions). Nevertheless, since the extraction of the information about this level can be done, to a great extent, by automatic tools, this specific task is not addressed by manual annotation and it is not discussed in this paper.

4. Annotation Analysis

In this section we will analyze the distribution of the annotated labels within the corpus and the inter-annotator agreement.

4.1. Label Distribution

The annotation process described in the last section allowed the achievement of an agreement for 1,066 tweets out of the 1,200 tweets collected and annotated. Together with the 400 tweets analyzed in (Karoui et al., 2015) and described at the beginning of section 2. they can be considered as a novel gold standard for Italian, that is TWITTIRÒ, consisting of almost 1,500 tweets. In the rest of this section we describe the distribution of the annotated labels on the TWITTIRÒ corpus.

Figure 2 shows the distribution among the three subcorpora from where the tweets were extracted (TW-SPINO,

[5] http://it.wikipedia.org/wiki/Full_Monty_-_Squattrinati_organizzati

[6] http://www.ansa.it/web/notizie/rubriche/associata/2014/01/04/Immigrazione-soccorso-anche-gatto-migrante-barcone_9851581.html

TW-SENTIPOLC-14, and TW-BS), labeled with either EXPLICIT or IMPLICIT tag, concerning the type of activation of irony.

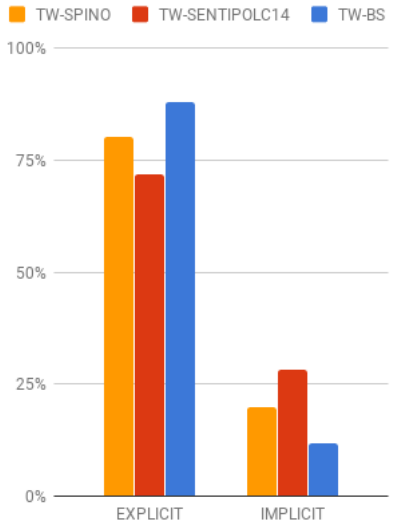


Figure 2: Distribution of types

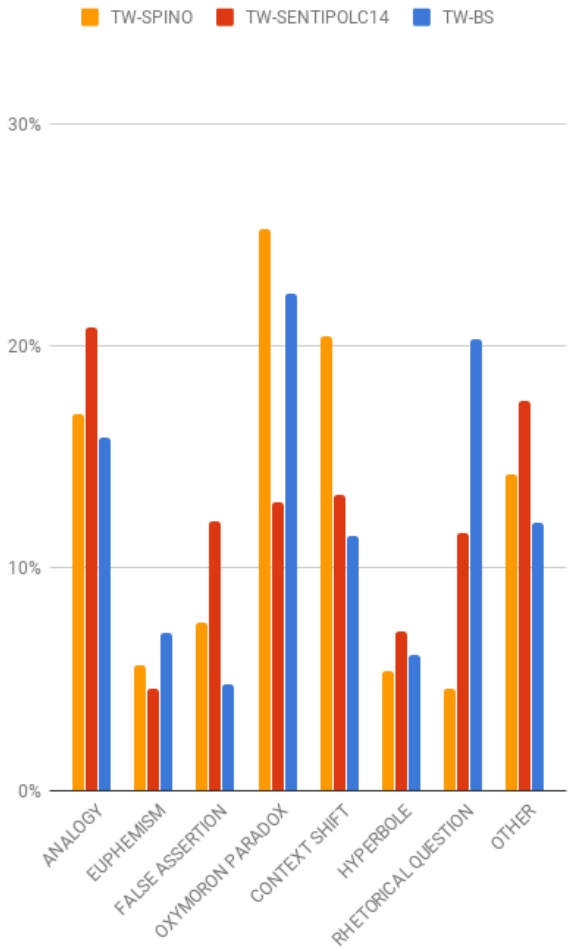


Figure 3: Distribution of categories

As it can be seen in Figure 2, in the majority of tweets, irony is triggered by an explicit contradiction in all the three datasets (80% in TW-SPINO, 72% in TW-SENTIPOLC14, and 88% in TW-BS). This confirms the findings in (Karoui et al., 2017) on the first section of the corpus, which highlight that Italian displays a different behavior concerning the preference for explicit activation type. In fact, languages such as French and English seem to favor implicit activation type, as shown in (Karoui et al., 2017).
Figure 3 shows the distribution of labels for Level 3 of the annotation scheme, describing how devices that trigger irony are distributed in the three different sub-corpora. The sub-corpus TW-SPINO is characterized by a strong use of the devices of OXYMORON/PARADOX and CONTEXT SHIFT. This suggests that the contributors of the satirical blog *Spinoza.it* often recur to this explicit kind of devices in order to create a sense of surprise, as in the following example:

- **TW-SPINO** - Marino si è dimesso. Ora la metro sembra nuova. [pirata_21]
  Marino resigned. Now the underground looks new. [pirata_21]
  LEVEL 2: EXPLICIT
  LEVEL 3: OXYMORON/PARADOX

Another observation that can be discussed about Figure 3, is the fact that TW-BS, a corpus composed by tweets about the reformation of the Italian School: *"La Buona Scuola"*, contains a high number of RHETORICAL QUESTION tags, which we can link to the dissatisfaction of the people, or at least, their perplexity on the matter.

- **TW-BS** - Ma i punti de #labuonascuola sono riferiti anche a quella pubblica? #perdire
  Are the bullet points of #labuonascuola also referred to public school? #justsaying
  LEVEL 2: EXPLICIT
  LEVEL 3: RHETORICAL QUESTION

4.2. Inter-annotator Agreement

As cited in the previous section, we compared the annotation of A1 and A2 on all the 1,200 tweets, calculating that the two annotators achieved the agreement for 656 tweets, but they disagree on the other 544. To solve the disagreement we applied a further independent annotation of a third human expert (A3), and we achieved the agreement on further 410 tweets.
We compared the annotation of A1 and A2 on all the 1,200 tweets. Their agreement, calculated through Cohen's kappa coefficient shows some interesting results. In fact, the inter-annotator agreement (IAA) between A1 and A2 concerning the choice between IMPLICIT and EXPLICIT (Level 2), calculated with Cohen's coefficient, is $\kappa = 0.41$, that is a low agreement (Landis and Koch, 1977). On the other hand, the IAA regarding category tags (Level 3) is $\kappa = 0.46$, i.e. a moderate agreement.
Figure 4 shows how the labels of category tags chosen by A1 is distributed for each category chosen by A2. This type of illustration is helpful to understand how the choices of one annotator agree (or disagree) with the choices of the

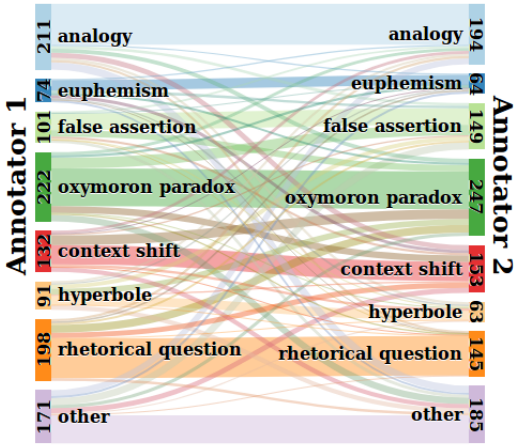


Figure 4: Detail over the agreement between A1 and A2

other, and at the same time what happens in case of disagreement on the tagging of a certain tweet. On the left the choices of A1 are displayed, whereas on the right we see the choices of A2. All the lines, or "fluxes", connecting left and right, describe the sparsity of the annotation work, and are coherent with a certain rate of disagreement. In fact, the more the annotators agree, the more the lines are straight and the less we would see divergent fluxes.

We noticed that 55 times (about 10% of disagreement cases), OXYMORON/PARADOX was chosen by an annotator and FALSE ASSERTION was chosen by the other. This statistics corroborate the intuitions of previous work (Cignarella et al., 2017), stating that there might be a problem on the comprehension of the true value of these two specific category tags. A relationship exists between the category FALSE ASSERTION (only implicit) and the category OXYMORON\PARADOX (only explicit). In fact, according to our multi-layered scheme for irony, those two categories, cover similar types of irony. Often the decision of Level 3 (Category Type) is triggered from the previous decision of Level 2 (Contradiction Type). For example, the category FALSE ASSERTION can be chosen only when we label the tweet as IMPLICIT. On the other hand, the category tags CONTEXT SHIFT and OXYMORON/PARADOX can occur only if Level 2 presents an EXPLICIT type of irony activation.

Observing the tag distribution between A1 and A2 in Figure 4, the tag OXYMORON/PARADOX is the more frequently exploited, followed by ANALOGY.

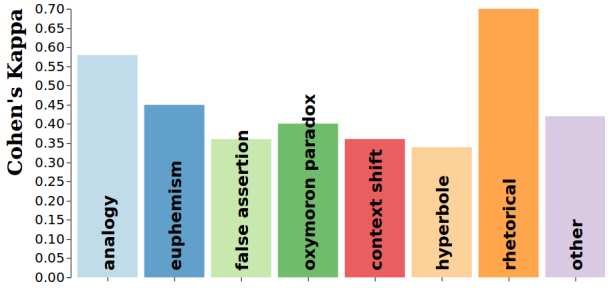


Figure 5: IAA between A1 and A2 on each category tag

To further validate our intuitions, we calculated the agreement of A1 and A2 on each category tag, and as it can be seen in Figure 5 the category tags OXYMORON/PARADOX (κ = 0.40) and FALSE ASSERTION (κ = 0.36) are among the three worst categories in agreement, preceded only by HYPERBOLE (κ = 0.34). This means that even though annotators exploit the OXYMORON/PARADOX tag several times, they rarely agree on its correct application. The categories with the highest inter-annotator agreement are instead RHETORICAL QUESTION (κ = 0.70) and ANALOGY (κ = 0.56). In general, as it is summarized in Table 3, annotators A1 and A2 reach a moderate agreement, κ = 0.46, on category tags. Moreover, it is interesting to notice that the value of the average kappa, on all 1,200 is increased by the pretty good value of the annotation on the sub-corpus of SPINOZA (κ = 0.57) and lowered by the poor results on TW-SENTIPOLC14 (κ = 0.34).

n# of tweets	sub-corpus	IAA
1,200	ALL	κ = 0.46
300	TW-SPINO	κ = 0.57
312	SENTIPOLC	κ = 0.34
588	TW-BS	κ = 0.47

Table 3: Cohen's kappa (A1 and A2) on each sub-corpus

Our intuition is that the use of correct grammar, good writing style and punctuation, revised by the authors of the satirical blog, improves the precision of the annotation. This fact does not apply to the sub-corpora of TW-SENTIPOLC14 and TW-BS which present a more heterogeneous shape and style. As we already mentioned, also after the application of a third human independent annotation, we didn't reach an agreement for classifying 134 tweets according to our scheme. Thinking that a deeper observation of tweets in disagreement can give interesting information about the different nuances through which irony is produced and about the complexity of the task, we further discuss such hard cases in the next section.

5. Discussion on Difficult Cases

Of the last remaining 134 tweets, which still are in a condition of disagreement (referred as *HardCases* henceforth), 20% pertains to TW-SPINO, 23% pertains to TW-SENTIPOLC14 and 57% to TW-BS.

In the following, we provide linguistic examples and description of one tweet of *HardCases* extracted from each sub-corpus.

- **TW-SPINO** - Alla stazione di Budapest nasce una bimba chiamata Speranza. In Italia l'avrebbero chiamata "Ci scusiamo per il disagio". [guli1979]
  At the Budapest train station a child named Hope was born. If had been born in Italy she'd be called "We're sorry for the inconvenience". [guli1979]
  A1: ANALOGY,
  A2: OXYMORON,
  A3: CONTEXT SHIFT.

The tweet makes an implicit reference to the primary train operator in Italy, *Trenitalia*, which is well known to offer

unsatisfying service and to often accumulate hours of delay on its convoys. The sentence "We're sorry for the inconvenience" is what travelers hear from a registered voice aired through the loud speakers in the train stations when such delay happen.

- **TW-SENTIPOLC14** - Tutti i ministri del governo Monti sono più vecchi di me. Sono soddisfazioni.
  All the ministers in the Monti government are older than me. This is satisfying!
  A1: CONTEXT SHIFT,
  A2: OTHER,
  A3: EUPHEMISM.

The ironic tweet refers to the fact that Mario Monti, the Italian Prime Minister between 2011 and 2013 in the wake of the Italian debt crisis, is an old man and, as he was appointed, he composed his commission with politicians even older than him.

- **TW-BS** - @Corriereit "non si consegna mai il compito senza rileggere": regola n.1 della "buona scuola". Glielo diciamo?
  @Corriereit "You never hand in the paper without correcting first": first rule of "la buonascuola". Should we tell them?
  A1: ANALOGY,
  A2: CONTEXT SHIFT,
  A3: RHETORICAL QUESTION.

The tweet makes an implicit reference to the event in which the first draft of the document presenting the school reform *"La Buona Scuola"* contained six orthographic errors and, has therefore been mocked ever since as a non-serious school reform. Furthermore the sentence "You never hand in the paper without correcting first" is an echoic mention based on common knowledge among Italian students, because it is what each teacher would repeat to pupils before they hand in a test.

Below each tweet we reported the annotation tags, assigned by each annotator. It can be observed that, when irony is activated, there can be the co-occurrence of one or more categories. None of the choices made from the annotators could be depicted a priori as *right* or *wrong*. Therefore, a good improvement in the application of the scheme would be that of accepting the labeling of two or more categories from each annotator on a single tweet.

6. Conclusions

The present research aimed at seeking more answers on the applicability of the multi-layered annotation scheme for irony (Karoui et al., 2017) on Italian texts extracted from Twitter. In doing so, we expanded the TWITTIRÒ corpus in order to create a gold standard which can be exploited in the cross- and multi-lingual perspective embraced in the research that inspired it.

In the present work, we conducted both a quantitative and qualitative study on the annotated data, with a focus on the annotation phase, its outcomes and an analysis of annotation disagreement. We deeply discussed the inter-annotator agreement (IAA), commenting on the difficult cases, providing several tweets of example. A second point of interest has been the study of style and composition of the three sub-corpora TW-SPINO, TW-SENTIPOLC14, and TW-BS, revealing interesting paths to be followed in future work.

While performing a linguistic analysis of the TWITTIRÒ corpus, we observed that syntax plays a significant role in the activation of irony in Italian, especially in social media short messages. Therefore, we are planning to enrich our actual dataset with additional syntactic information such as Part-of-Speech tags and syntactic relations in *Universal Dependencies* (UD) format. The choice of this type of format comes firstly, because of its popularity within Computational Linguistics, as it has been already exploited in other evaluation campaigns (*EVALITA* and *CoNLL*). Secondly, its application on social media text has already been proven useful (Sanguinetti et al., 2017; Sanguinetti et al., in press 2018).

Together with the application of new morphological and syntactic labels, we are also planning to explore the activation of different types of irony (Sulis et al., 2016; Van Hee et al., 2016), such as the presence of situational irony, sarcasm and puns based on stereotypes.

Acknowledgments

C. Bosco, A. T. Cignarella and V. Patti were partially funded by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media*, S1618_L2_BOSC_01) and by Fondazione CRT (*Hate Speech and Social Media*, 2016.0688).

Bibliographical References

Barbieri, F., Basile, V., Croce, D., Nissim, M., Novielli, N., and Patti, V. (2016). Overview of the Evalita 2016 SENTIment POLarity Classification Task. In *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016.*, volume 1749 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Basile, V., Bolioli, A., Nissim, M., Patti, V., and Rosso, P. (2014). Overview of the Evalita 2014 SENTIment POLarity Classification Task. In *Proceedings of the 4th Evaluation Campaign of Natural Language Processing and Speech tools for Italian (EVALITA'14)*, pages 50–57, Pisa, Italy. Pisa University Press.

Farah Benamara, et al., editors. (2017). *Analyse d'opinion et langage figuratif dans des tweets : présentation et résultats du Défi Fouille de Textes DEFT2017*. TALN 2017, Orléans.

Bosco, C., Patti, V., and Bolioli, A. (2013). Developing Corpora for Sentiment Analysis: The Case of Irony and Senti-TUT. *IEEE Intelligent Systems*, 28(2):55–63.

Cignarella, A. T., Bosco, C., and Patti, V. (2017). TWITTIRÒ: a Social Media Corpus with a Multi-layered Annotation for Irony. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017*, volume 2006 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Fink, C. R., Chou, D. S., Kopecky, J. J., and Llorens, A. J. (2011). Coarse- and Fine-Grained Sentiment Analysis of Social Media Text. *Johns Hopkins APL Technical Digest*, 30(1):22–30.

Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Barnden, J., and Reyes, A. (2015). Semeval-2015 task 11: Sentiment Analysis of Figurative Language in Twitter. In *Proceedings of SemEval 2015, Co-located with NAACL*, page 470–478. ACL.

Grice, P. H. (1975). Logic and Conversation. *Syntax and Semantics 3: Speech Arts*, pages 41–58.

Grice, P. H. (1978). Further Notes on Logic and Conversation. *Pragmatics*, 1:13–128.

Hernández Farías, D. I., Patti, V., and Rosso, P. (2016). Irony Detection in Twitter: The Role of Affective Content. *ACM Transactions on Internet Technologies*, 16(3):19:1–19:24.

Karoui, J., Benamara, F., Moriceau, V., Aussenac-Gilles, N., and Belguith, L. H. (2015). Towards a contextual pragmatic model to detect irony in tweets. In *53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*.

Karoui, J., Benamara, F., Moriceau, V., Patti, V., and Bosco, C. (2017). Exploring the Impact of Pragmatic Phenomena on Irony Detection in Tweets: A Multilingual Corpus Study. In *Proceedings of the European Chapter of the Association for Computational Linguistics*.

Kouloumpis, E., Wilson, T., and Moore, J. D. (2011). Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the ICWSM: International AAAI Conference on Web and Social Media*.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Maynard, D. and Funk, A. (2011). Automatic Detection of Political Opinions in Tweets. In *Proceedings of the ESWC: Extended Semantic Web Conference*.

Mihalcea, R. and Pulman, S. (2007). Characterizing Humour: An Exploration of Features in Humorous Texts. In *International Conference on Intelligent Text Processing and Computational Linguistics*.

Reyes, A., Rosso, P., and Buscaldi, D. (2010). Finding Humour in the Blogosphere: the Role of Wordnet Resources. In *Proceedings of the 5th Global WordNet Conference*.

Reyes, A., Rosso, P., and Buscaldi, D. (2012). From Humor Recognition to Irony Detection: The Figurative Language of Social Media. *Data & Knowledge Engineering*, 74:1–12.

Sanguinetti, M., Bosco, C., Mazzei, A., Lavelli, A., and Tamburini, F. (2017). Annotating Italian social media texts in Universal Dependencies. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 229–239.

Sanguinetti, M., Bosco, C., Lavelli, A., Mazzei, A., Antonelli, O., and Tamburini, F. (in press, 2018). PoSTWITA-UD: an Italian Twitter treebank in Universal Dependencies. In *Proceedings of LREC18*.

Sperber, D. and Wilson, D. (1981). Irony and the Use-Mention Distinction. *Philosophy*, 3:143–184.

Stranisci, M., Bosco, C., Farías, D. I. H., and Patti, V. (2016). Annotating sentiment and irony in the online italian political debate on #labuonascuola. In *Proceedings of the Tenth International Conference on Language Resources (LREC 2016)*.

Sulis, E., Hernández Farías, D. I., Rosso, P., Patti, V., and Ruffo, G. (2016). Figurative messages and affect in Twitter: Differences between #irony, #sarcasm and #not. *Knowledge-Based Systems*, 108:132 – 143. New Avenues in Knowledge Bases for Natural Language Processing.

Van Hee, C., Lefever, E., and Hoste, V. (2016). Exploring the realization of irony in Twitter data. In Nicoletta Calzolari, et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA).

Van Hee, C., Lefever, E., and Hoste, V. (in press, 2018). SemEval-2018 Task 3: Irony detection in English Tweets. In *In Proceedings of the International Workshop on Semantic Evaluation (SemEval-2018)*. New Orleans, LA, USA, June 2018.

Van Hee, C. (2017). *Can machines sense irony?: exploring automatic irony detection on social media*. Ph.D. thesis, Ghent University.

Wilson, D. and Sperber, D. (2007). On verbal irony. *Irony in language and thought*, pages 35–56.

Classifier-based Polarity Propagation in a Wordnet

Jan Kocoń, Arkadiusz Janz, Maciej Piasecki
Faculty of Computer Science and Management
Wrocław University of Science and Technology, Wrocław, Poland
{jan.kocon, arkadiusz.janz, maciej.piasecki}@pwr.edu.pl

Abstract

In this paper we present a novel approach to the construction of an extensive, sense-level sentiment lexicon built on the basis of a wordnet. The main aim of this work is to create a high-quality sentiment lexicon in a partially automated way. We propose a method called Classifier-based Polarity Propagation, which utilises a very rich set of wordnet-based features, to recognize and assign specific sentiment polarity values to wordnet senses. We have demonstrated that in comparison to the existing rule-base solutions using specific, narrow set of semantic relations, our method allows for the construction of a more reliable sentiment lexicon, starting with the same seed of annotated synsets.

Keywords: sentiment analysis, polarity propagation, wordnet

1. Introduction

Sentiment analysis of natural language utterances is continuously increasing its importance as one of the most expected techniques. The best results have been obtained with supervised approaches trained on the basis of annotated texts from a selected domain, e.g. movie or product reviews. However, cross-domain applications show a significant drop in the performance of classifiers trained on a corpus built from texts from the other domains. This can be attributed to a high correlation of a classifier with words and phrases that are specific for the positive and negative utterances of the given domain. However, language expresses some lexical means of conveying sentiment polarity in a way that is shared across different domains. A compromise between performance and domain adaptability can be achieved using hybrid methods. A lexicon of sentiment polarity could be a very useful basis for constructing such a domain independent, hybrid system, if such a lexicon is large, comprehensive and reliable enough.

plWordNet 3.1 emo[1] (Maziarz et al., 2016; Zaśko-Zielińska and Piasecki, 2018) is a very large lexical semantic network for Polish, in which more than 190,000 different lemmas and 285,000 Polish lexical meanings are described by the lexico-semantic relations. plWordNet has become one of the largest Polish dictionaries ever built, and the largest wordnet in the world. A substantial part of plWordNet was manually described by emotive annotation (Zaśko-Zielińska and Piasecki, 2018). In this work we would like to expand this emotive annotation to a very large scale and make plWordNet a basis for a large hybrid emotive lexicon of Polish, as well as for the development of a hybrid system for sentiment and emotion analysis in Polish texts. Existing solutions for sentiment propagation over a wordnet are mostly based on a simple label propagation algorithm, starting with a relatively small initial seed. However, generally they do not take into account more complex wordnet structures, thus we may lose some information which can be a good indicator of sentiment polarity. Exploiting the full wordnet structure gives us an opportunity to capture the polarity of senses in a more accurate way. We may want to consider not only a wider network context of a word sense, but also a richer set of lexical relations to propagate the sentiment polarity. Yet another problem with the existing approaches is that many solutions depend on handcrafted propagation rules that cannot be easily transferred to a wordnet built for another language. Here we propose a method which allows for automated discovery of propagation rules by using the wordnet structure in a more extensive way to recognise the sentiment polarity of senses.

2. Related Works

SentiWordNet, one of the most commonly used sentiment resources for English, was introduced in (Esuli and Sebastiani, 2006). The main goal of the authors was to construct a large lexical resource with sentiment polarity assigned to meanings, rather than words[2]. There were many attempts to construct sense-level sentiment lexicons, but most of them were evaluated only for English. The easiest way to create a sense-level sentiment lexicon for another languages is simply to map SentiWordNet annotations to a non-English wordnet via existing mappings between the two wordnet or even translating first Princeton WordNet (Fellbaum, 1998) into another language. However, wordnets for different languages may differ significantly, e.g., in the number of relation instances and a different semantic structure. One of the most commonly used techniques for this task is relational label propagation using a random walk algorithm.

(Vossen et al., 2008) compared a lexicon constructed by applying a simple polarity transfer from SentiWordNet and a lexicon built by label propagation on Dutch WordNet (Vossen et al., 2013). The simple transfer of annotations resulted in a general decrease of performance in comparison to SentiWordNet. The second approach was based on random walk with propagation rules exploiting a narrow set of lexical relations from Dutch WordNet. The authors noted, that the factors such as the seed size, seed composition and the number of iterations had a great impact on propagation performance. Thus, they evaluated their approach on

[1] http://plwordnet.pwr.edu.pl

[2] However, the annotation in SentiWordNet is still done on the level of synsets, not individual word senses.

three datasets of different quality: high-quality, low-quality and mixed. The best results were achieved using the largest dataset of mixed quality, derived from the General Inquirer (Stone, 1966) – a sentiment lexicon. The conclusion was, that the size is the most important factor. The authors also proposed a third approach combining this transfer method with label propagation, with almost the same result. The results may also suggest, that simple transfer methods are not perfect, but combining multiple approaches with transfer methods may bring us promising results.

The authors of (Maks et al., 2014) expanded research on the sentiment propagation to non-English wordnets. They applied the same propagation method to five wordnets of different languages. Words and their polarity were acquired from the well-known sentiment lexicon – the General Inquirer Lexicon, and then translated with a machine translation service to five languages. The words were manually mapped to their corresponding synsets in particular wordnets, and used as a seed for propagation. The resulting lexicons varied significantly with respect to their size and precision. The authors concluded, that the way a given wordnet has been built seems to affect the propagation performance. Thus, we should not apply the same propagation scheme for every wordnet.

(Mahyoub et al., 2014) was the first attempt to build an Arabic sentiment lexicon on the basis of Arabic WordNet (Black et al., 2006). They introduced two steps in their procedure: the *expansion step* – a sentiment lexicon is expanded by iteratively reaching concepts of the wordnet; the *scoring step* – the sentiment score of the reached concepts is computed according to their distance from the seeds. A task-based evaluation was applied to evaluate this solution. The acquired polarity scores were incorporated into features for the sentiment classification task, next evaluated on the Arabic corpora.

There were several attempts to construct a large sentiment lexicon for Polish in an automated way e.g. (Haniewicz et al., 2013), (Haniewicz et al., 2014). (Haniewicz et al., 2013) attempted to build a polarity lexicon from web documents. They utilized an older version of plWordNet, so still without sentiment annotation, as a general lexical resource in order to develop domain-aware polarity lexicons. plWordNet was utilized to identify semantic relations between the acquired terms. To determine their polarity, a supervised learning with Naive Bayes and SVM was applied. This approach was extended in (Haniewicz et al., 2014), where the semantic lexicon was expanded up to 140,000 terms, using simple rule-based propagation method based on an adaptation of the random walk algorithm.

SentiWordNet construction in its recent stages was generally based on glosses from Princeton WordNet. (Misiaszek et al., 2013) proposed a lexicon construction method for wordnets, in which a simple transfer method could not be easily applied, or external sources of knowledge, such as tagged and disambiguated glosses, are not available. They used relational propagation scheme with local, collective classification method to determine polarity of a synset. The training features for the classifier were obtained using only a close neighbourhood of annotated synsets, consisting of nodes with known polarity. They manually annotated specific synsets in a wordnet and used them as seeds for the propagation process. However, the details of the extraction of features were not specified, and there was no evaluation for their approach.

In (Kulisiewicz et al., 2015) the propagation was performed by using an adaptation of Loopy Belief Propagation (*LPA*) on Princeton WordNet 3.0. Three different variants of the *LPA* have been tested and evaluated. First, the authors compared their results with polarity scores from SentiWordNet (*Mean Square Error*), but without the *Objective* class. Second evaluation was a comparison with polarity of words existing in the General Inquirer Lexicon. The resultant performance was ambiguous and the main conclusion was, that semantic relations within wordnet may not be well correlated with the sentiment relations.

3. Emotive Annotation in plWordNet

3.1. plWordNet model

plWordNet in brief, follows generally the main ideas of Princeton WordNet(Fellbaum, 1998), consists of *lexical units* linked by *lexico-semantic relations* and grouped into *synsets*. A lexical unit (LU) represents a lexical meaning and is a triple: lemma, Part of Speech and sense identifier. Contrary to WordNet, LUs, not synsets, are the basic building blocks of plWordNet. Use examples are in a natural way assigned to LUs, as well as glosses. Lexico-semantic relations are defined by detailed guidelines including substitution tests and referring to the use examples that can be observed in text corpora. Moreover, plWordNet is developed by a corpus-based wordnet development method in which corpus exploration and the work with examples of the use of different lemmas and their potential senses (not synsets or concepts shared in lexical meanings) are crucial for wordnet editing. Finally, it is also worth to notice that the construction of plWordNet follows in this aspect the long term tradition of the lexicography.

3.2. Emotive annotation scheme

Thus, following this fundamental construction decisions, *emotive annotation* in plWordNet has been also defined on the level of LUs. LUs are the natural targets of the emotive annotation which is strongly associated with the use of LUs. Initially, as a result of a pilot project (Zaśko-Zielińska et al., 2015), emotive annotations have been manually added for a selected subset of more than 31,000 LUs in plWordNet 2.3 emo. LUs were described, see also (Zaśko-Zielińska and Piasecki, 2018), by:

- *markedness*,
- *intensity* of sentiment polarity,
- *basic emotions*,
- *fundamental human values*,
- *usage examples*.

The annotation goes beyond a typical sentiment polarity annotation and that is why it is called *emotive*.

First of all, LUs are dived into *neutral* vs *marked* with respect to sentiment *polarity*.

In the case of the *intensity*, we assumed a rather modest scale for sentiment polarity of five grades, namely: *strong* or *weak* vs *negative* and *positive*, plus *neutral* LUs in the middle of the scale. We keep the number of grades limited, as the annotation is performed by two annotators per one LU. Each LU is annotated by a linguist and a psychologist. The work of annotators is controlled and verified by a super-annotator and is based on a strict lexicographic procedure and detailed guidelines, see (Zaśko-Zielińska et al., 2015; Zaśko-Zielińska and Piasecki, 2018). In general, the procedures combines work on the corpus data, several linguistic tests and analysis of glosses, relation structure, as well as definitions in traditional dictionaries. We were afraid that with a larger number of intensity grades the inter-annotator agreement could be low. This assumption was not experimentally verified, but the achieved IAA for the applied scale, see Sec. 3.3. is high.

Sentiment analysis often uses sets of basic emotions proposed by Ekman (Ekman, 1992) or Plutchik (Plutchik, 1980). In order to make plWordNet emo compatible with a number of other resources, we used the set of eight basic emotions recognised by Plutchik. It contains Ekman's six basic emotions (Ekman, 1992): *joy* , *fear*, *surprise*, *sadness*, *disgust*, *anger*, complemented by Plutchik's *trust* and *anticipation*. Annotators are allowed to assigned more than one emotion per LU. In this way complex emotions can be also expressed.

From the very beginning of the pilot project, we use the set of *fundamental human values* postulated by Puzynina (Puzynina, 1992) later followed in many works on lexicography and derivation, as a tool for the analysis of the evaluative attitude of a hearer or speaker, see (Zaśko-Zielińska and Piasecki, 2018). The set of the fundamental human values includes: *użyteczność* 'utility', *dobro drugiego człowieka* 'another's good', *prawda* 'truth', *wiedza* 'knowledge', *piękno* 'beauty', *szczęście* 'happiness' (all of them positive), *nieużyteczność* 'futility', *krzywda* 'harm', *niewiedza* 'ignorance', *błąd* 'error', *brzydota* 'ugliness', *nieszczęście* 'misfortune' (all negative) (Puzynina, 1992).

We do not expect perfect agreement on, both, basic emotions and fundamental human values assigned by the annotators. Moreover, assignment of basic emotions and fundamental human values is a tool supporting annotators in making the final decision about the grade of the sentiment polarity, which is done after the emotions and values are assigned. Nevertheless, the overlap of both types of sets is very high in the case of almost annotated LUs.

Use examples are sentences provided for the analysed LUs by annotators in order to justify their decisions and to illustrate the assigned annotation. The annotators select use examples from a corpus, if the source texts are available on an open licence, otherwise an use example is created in a way similar to the sentences observed in the corpus.

Below we present examples of annotation: ***dziad*** 1 gloss:"stary mężczyzna" 'an old man'
⟨ Annot.:A1, BE: {*złość* 'anger', *wstręt* 'disgust'}, FHV:{*nieużyteczność* 'futility', *niewiedza* 'ignorance'}, SP:$-s$
Exam: "Stary dziad nie powinien podrywać młodych dziewczyn."
'An old geezer should not pick up young girls.' ⟩
⟨ Annot.:A2, BE: {*wstręt* 'disgust'}, FHV:{*nieużyteczność* 'futility', *brzydota* 'ugliness'}, SP:$-w$
Exam: "Jakiś dziad się dosiadł do naszego przedziału i wyciągnął śmierdzące kanapki z jajkiem." 'An old geezer joined our compartment and took out stinky egg sandwiches.' ⟩
⟨ Annot.:A3, BE: {*wstręt* 'disgust'}, FHV:{*nieużyteczność* 'futility', *brzydota* 'ugliness'}, SP:$-s$
Exam:"Kilkanaście lat minęło i zrobił się z niego stary dziad."
'Several years have passed and he has become an old geezer' ⟩
gość 3 '≈a fellow, ≈a man' gloss:"z podziwem o kimś godnym szacunku, kto się czymś wykazał" 'with admiration about someone who is worth respect, who showed something exceptional'
⟨ Annot.:A1, BE: {*zaufanie* 'trust', *radość* 'joy'}, FHV:{*wiedza* 'knowledge', *dobro* 'another's good', *użyteczność* 'utility'}, SP:$+w$
Exam: "Mój pracodawca jest świetnym gościem."
'≈My employer is a very good man.' ⟩
⟨ Annot.:A2, BE: {*zaufanie* 'trust', *radość* 'joy'}, FHV:{*szczęście* 'happiness', *wiedza* 'knowledge', *dobro* 'another's good'}, SP:$+s$
Exam: "Paweł to jest dopiero gość!" 'Paweł, he is a really good man!' ⟩
⟨ Annot.:A3, BE: {*zaufanie* 'trust', *radość* 'joy'}, FHV:{*szczęście* 'happiness', *wiedza* 'knowledge'}, SP:$+s$
Exam:"Boże, ale z niego gość, potrafił taką sprawę załatwić w pięć minut."
'My God! What a man is he, he has been able to solve such a problem in five minutes' ⟩
szalbierski 2 'deceitful'
⟨ Annot.:A1, BE: {*smutek* 'sadness', *złość* 'anger'}, FHV: {*krzywda* 'harm', *błąd* 'error' }, SP:$-s$,
Exam: "Nie chciałam brać udziału w tym szalbierskim planie, którego pomyślność zależała od stopnia naiwności nieświadomych klientów."
'I did not want to take part in this deceitful plan, whose success depended on the level of naiveness of the unaware clients.'⟩
⟨A2, BE: {*smutek* 'sadness', *złość* 'anger'}, FHV: {*krzywda* 'harm', *błąd* 'error'}, SP:$-s$,
Exam: "Mam szalbierski pomysł, który pomoże nam naciągnąć paru idiotów."
'I have a deceitful idea which might help us to con a couple of idiots. ' ⟩

Following the approach of the pilot project, and keeping the annotation scheme unchanged, in June 2017 we have started work on large scale expansion of this pilot project. Annotation procedure, guidelines and tools have been improved and expanded and the target size is adding emotive annotations to ≈100,000 more LUs, so the expected target number of annotated LUs by June 2018 is ≈130,000 LUs.

3.3. Statistics

At the time when the experiments were carried out, there was more than $83k$ annotations, covering more than $54k$

LUs and $41k$ synsets (Janz et al., 2017). This data has been successfully applied to PolEval 2017 Sentiment Analysis Task (Ptaszyński et al., 2017). In this previous plWordNet version, about $22k$ of the polarity annotations are different than neutral and these annotations cover $13k$ LUs and $9k$ synsets (22% of all synsets that include annotated LUs). We found, that $1.5k$ of these synsets were annotated with differences among the polarity values assigned to their synset members. If we exclude neutral LUs, only 345 of them have diversified polarity intensity (e.g. synset that contains two LUs annotated as *strong positive* and one annotated as *weak positive*). If we exclude both neutral and ambiguous annotations, there are only 41 synsets expressing potential conflicting, opposite polarity values of their LUs, i.e. synsets that include both positive and negative LUs. However they comprise only 3.8% of all marked synsets, i.e. synsets that do not contain any neutral LUs, namely 9164 in total. The contemporary intermediate state of the process of the emotive annotation of plWordNet is illustrated in Table 1. The overall numbers of annotations, as well as distribution of the polarity intensity is shown.

As our annotators work in a completely independent way, we were able to measure the inter-annotator agreement (IAA) with respect only to the sentiment polarity by using the Cohen's Kappa measure (Cohen, 1960), see Tab. 2. Due to the large number of annotators, we simplified the problem of IAA a little bit, and we have calculated the agreement between the first and the second decision registered in the system for a LUs. All LUs with at least one annotation from the pilot project were excluded from this analysis. The observed IAA values, both, 0.78 for all decisions and around 0.75 for different sentiment polarity values, are very high. The value for the neutral polarity is in fact a value for the decision: marked vs unmarked (or polarised vs non-polarised) LUs. It can show that the annotators are quite confident about the neutrality of LUs. However, also it can be biased by the fact that describing a LU as a neutral can be easier than by other values. The evaluation of the neutrality of a given LU is made in the first step of the annotation procedure and LUs decided as neutral are not further analyses. This issue needs further investigation.

As the neutral annotations dominate (more than 70% of all decisions, in the case of nouns even more than 80%), we have calculated an estimated IAA value for the marked LUs only, all LUs with neutral tags were excluded. The obtained values are much higher than for all decisions, so we can conclude that neutral values do not increase artificially the general IAA.

Negative polarity values dominate in annotation: 17.1% vs 8.51% in Tab. 2. This correlates with the observed dominance of the negative basic emotions, i.e. 76.48% emotions of noun LUs and 70.13% of adjective LUs are negative. A similar dominance of words marked negatively could be also observed in the dictionary of the colloquial Polish language (Anusiewicz and Skawiński, 1996). For instance, if we compare two thematic fields of this dictionary, namely: acting towards somebody's harm – enforcing some particular behaviours (id:2.3.2) and acting towards somebody's profit (id.: 2.3.3), we can notice that the former includes 324 entries while the latter only 20 (Zaśko-Zielińska and Piasecki, 2018).

4. Sentiment Polarity Propagation

We propose a method called *Classifier-based Polarity Propagation* (henceforth CPP), which utilises a rich set of features. This richness arises from their construction, as they take into account even broad neighbourhood context of synsets (up to 2 levels around the synset, see Figure 1), and refer to an extended set of semantic relations.

In Section 5. we compare the results obtained by CPP with the rule-based and relation-based method called *Seed Propagation*, using the best configuration presented by Maks and Vossen (2011).

4.1. Polarity Transfer from Units to Synsets

We have analysed the contemporary annotation of plWordNet from the perspective of the diversification of polarity intensity of LUs belonging to one synset, see Sec. 3.3.. As we could notice, unless, cases of significant differences of the polarity values in one synset are rare, still we can find a number of cases in which the values express smaller differences, e.g. between strong vs weak or weak vs neutral. In contrast to SentiWordNet the manual annotation in plWordNet is done only on the level of LUs (Zaśko-Zielińska et al., 2015) and synsets are not manually assigned sentiment polarity values.

The acquired statistics show, that synsets are relatively homogeneous in terms of the polarity values of their member LUs. Thus, we decided that moving the polarity intensity annotations from the LU-level to the synset-level can be meaningful and profitable[3]. In order to simplify the polarity transfer problem, we decided to project these values onto only three coarse-grained values: *positive*, *negative*, and *neutral*. In order to do this, in the first step, each original polarity value is assigned an heuristic weight: 2 for *strong* variants and 1 for weak variants, neutral and ambiguous. Next, the weights are summed up for each synset. For example, for a synset with a set of LUs annotated with the following values: {*strong negative, negative, strong positive, neutral*}, the total weight for the *positive* coarse-grained value equals 2, for the *negative* one equals 3 (2 + 1) and for the *neutral* is 1. Finally, a generalised sentiment polarity value for a synset is determined on the basis of a simple heuristic: coarse-grained synset polarity value is set to the one which have obtained the highest total weight inside the given synset. In the case of the above example the synset is assigned the *negative* polarity value. If two polarity values have the same total weight, we apply the following rules to solve this discrepancy:

- $\{positive, neutral\} \rightarrow positive$
- $\{negative, neutral\} \rightarrow negative$
- $\{positive, negative\} \rightarrow neutral$

[3]This can simplify some applications or facilitate comparison of even mapping of the sentiment polarity annotation between plWordNet and other wordnets with the help of the manually built interlingual mapping of good coverage.

PoS	# Comp	# Sing	-s	-w	n	+w	+s	amb
N	43,883	1,251	6.45	6.09	80.34	2.82	1.63	2.67
Adj	23,035	49	8.42	14.41	58.26	9.33	4.35	5.24
Verb	5,084	2191	5.11	21.32	48.08	14.97	1.09	9.43
Adv	28	731	7.64	16.73	52.31	12.65	5.14	5.53
All	72,030	4,222	6.93	10.17	70.30	6.05	2.44	4.12

Table 1: Contemporary (Feb. 2018) sentiment polarity annotation of plWordNet 4.0 in progress (Comp – completed, Sing – one annotator only so far); -s, -w, n, +w, +s, amb (negative strong/weak, neutral, positive weak/strong, ambiguous) are shown in percentage points.

PoS	All	-s	-w	n	+w	+s	amb
All	0.78	0.77	0.78	0.82	0.74	0.73	0.65
Mrk.	0.84	0.80	0.84	–	0.89	0.80	0.86

Table 2: Inter-annotator agreement (IAA), measured in Cohen's' κ, for different sentiment polarities: -s, -w, n, +w, +s, amb (negative/positive vs strong/weak, neutral, ambiguous). *All* describes agreement for all decisions, *Mrk* – estimated IAA for marked LUs only.

4.2. Features

From the structure of plWordNet we selected only the most frequent relations as a basis for features. As a result, the selected relations cover more than 95% of all relation instances (occurrences) in plWordNet: *hyponymy*, *hypernymy*, *fuzzynymy*, *similar_to*, *feature_value*, *meronymy*, *holonymy*, *collection_meronym*, *collection_holonym*, *type*, *member*, *taxonomic_meronym*, *taxonomic_holonym*.
Each synset is described by a set of feature values that all together form a kind of *bag-of-words* representation. This representation refers both to synsets and their polarities. Elements of the bag are constructed on the basis of the following four components each.

- *Relation* – one of the 13 selected wordnet relations.
- *Direction* – the direction of the relation, expressing whether the synset described by the feature is the *source* or *target* of the relation instance (i.e. outgoing vs ingoing relation instance).
- *Target* – it can be one of the two: a *synset_ID* (identifier) or *synset_polarity*, coarse-grained of the target synset encoded as: $-1, 0, 1$ for, respectively: *negative, neutral, positive*).
- *Level* – represents the distance in which the given relation instance is on the path connected to the synset being described, it can a *directly* linked to this synset (level=0), but also in some further distance, always only one link is expressed without the information about the rest of the path, so the level informs how broad is the context, see the example presented in Figure 1.

In total, we use 13 wordnet relations (types), 2 directions, 2 types of targets (exclusively) and up to $m = 2$ levels, so elements of the bag-of-word representation can be constructed in $13 \cdot 2^3 = 104$ ways. For example a feature of the type *hyponym_source_id_level_2* introduces into the

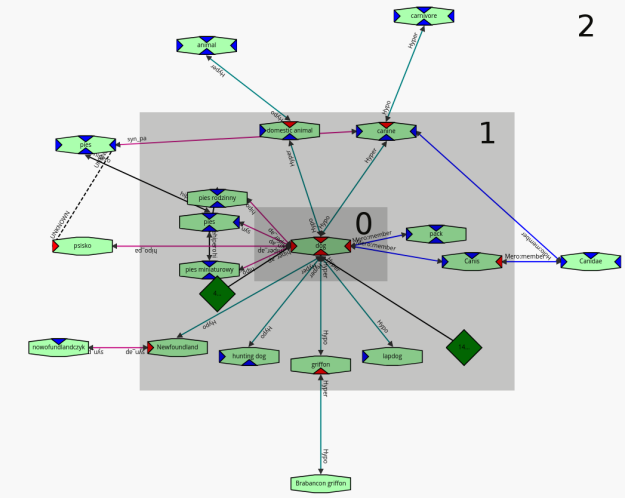


Figure 1: Example of synsets at the specific level (1 and 2), with respect to the synset at level 0.

representation all IDs of synsets which are sources of all hyponym relation instances, for which the target is a synset indirectly accessible (by the third link), see Figure 1.
In order to produce a single bag for the classifier, we concatenate the values of this bag into a single document and convert it with `TfidfVectorizer` into a vector representation. Acquired vectors for a bag of synset IDs and a bag of polarities are different in terms of their vector spaces, due to the different type of collected elements. However, if we consider only a single type of a bag, e.g. a bag of synset IDs collected for the hyponymy relation on level k, the constructed vectors for this type of a bag should be represented always in the same vector space.

4.3. Classifier and Propagation Method

To train a model for a classifier we need a set of manually annotated synsets with sentiment polarity annotations (i.e. *seeds*). Next, each synset is automatically described with 104 different bags of words as a complex representation, which takes into account its broader relational context of linked synsets and their annotations. The constructed bags-of-words are input to the `TfidfVectorizer` module from `scikit-learn`[4] Python machine learning package for building vectors. This feature extraction method allows to convert a collection of elements into a matrix of tf.idf features. Each synset belongs to one of the three fol-

[4] `http://scikit-learn.org`

lowing coarse-grained classes: *positive, negative, neutral*. Finally all vectors representing all bag-of-words are concatenated. The final model contains $38,108$ vectors – one vector represents one synset. The size of the final vector space is $38,108 \times 49,0170$. Transformed vectors are used to train a predictive model. We used Logistic Regression from `scikit-learn` package as a classifier.

A classifier with a trained model is applied in propagating annotations to the unlabelled part of plWordNet. At the beginning we treat our seeds as a set of synsets at level-0 (see Figure 1). Each next iteration is a classification of unlabelled synsets at the 1st level, using information from annotated synsets from the surrounding context. We tested two approaches to performing propagations, namely in each iteration:

- *naive* – we preserve the graph depth order of the remaining synsets to be classified,
- *sorted* – before each iteration we sort synsets at the 1st level by the number of relations with synsets that already have been assigned a polarity value (descending order). Then we start propagation with synsets at the top of this list.

5. Evaluation

The proposed method assumes that the propagation is performed only for synsets. However, the existing polarity annotations in the plWordNet refer only to LUs, so preprocessing was required. First, we used a simple generalization function, to assign the polarity to the synsets, depending on the polarity of their LUs (see Section 4.1.). As a result the original a 5-degree scale of sentiment polarity intensity was projected onto the coarse-grained 3-degree scale. Next, we prepared a large graph of plWordNet, consisting of synsets with sentiment polarity annotation transferred from their members, as a basis for wordnet-based evaluation of the method.

5.1. Wordnet-based Evaluation

During evaluation we included complete plWordNet annotation with $43k$ synsets annotated with sentiment polarity (positive, negative, neutral) in this particular version (from Oct. 2017). For each method and configuration we performed 10-fold cross-validation. Annotated synsets were divided into 10 parts, where 9 parts (about 40,400 synsets in total) were treated as a seed for the baseline (or a training set for CPP) and the 10th part (about 3,600 synsets) as a test set.

We implemented a simple rule-based seed-driven propagation method described in (Maks and Vossen, 2011) to obtain a *baseline* (henceforth BASE). Then we compared results of its application with the CPP method in two variants, described in Section 4.: naive (CPP-N) and sorted (CPP-S).

5.2. Task-based Evaluation

In (Qian et al., 2017) the authors proposed a simple, yet effective solution to recognize sentiment polarity of sentences using Bidirectional LSTM network. The proposed solution is based on additional regularization terms incorporating linguistic knowledge into the network. The regularization terms $L_{t,k}$ have been combined with the original cross entropy loss:

$$C(\theta) = -\sum_i \hat{y}_i \log y_i + \alpha \sum_i \sum_t L_{t,i} + \beta \|\theta\|^2 \quad (1)$$

The authors called their solution Linguistically Regularized LSTM (henceforth LR-LSTM) which is a model of LSTM network but expanded with a set of regularizers to better reflect the linguistic role of sentiment, negation and intensity of words (Qian et al., 2017). Here we have applied all four proposed regularization terms namely non-sentiment regularizer (NSR), sentiment regularizer (SR), negation regularizer (NR) and intensity regularizer (IR). In case of SR, the idea is that the model should restrict the polarity distribution of adjacent words in text to drift in the same way, especially in case of sentiment words included in provided lexicon:

$$p_{t-1}^{(SR)} = p_{t-1} + s_{c(x_t)} \quad (2)$$

$$L_t^{(SR)} = \max\left(0, D_{KL}(p_t \| p_{t-1}^{(SR)}) - M\right) \quad (3)$$

LR-LSTM was prepared and evaluated on English, thus we needed to adapt their work to the Polish language. To show the impact of the lexicon on the accuracy in sentiment analysis tasks three different variants of a lexicon (of the same size) were prepared:

1. a lexicon with randomly assigned sentiment scores,
2. a sentiment lexicon after rule-based propagation,
3. a sentiment lexicon constructed with CPP.

Dataset

We have collected a corpus of $4,039$ user reviews from Trip Advisor[5]. Table 3 presents the distribution of Users' Ratings assigned to the collected reviews. During the evaluation of our lexicon we needed to convert values of Users' Ratings into sentiment polarity values. Thus we automatically replaced rating values of reviews with the coarse grained polarity values according to the following schema: $\{1, 2\} \rightarrow negative$, $\{3\} \rightarrow neutral$, $\{4, 5\} \rightarrow positive$. The reviews were manually revised once again, just to correct assigned sentiment if necessary. The final classification was limited only to the recognition of $positive$ and $negative$ reviews.

Experimental Setting

In this section, we present our experimental setting for task-based evaluation, especially the characteristics of evaluated lexicons and the training procedure for LR-LSTM. On a basis of a collected corpus, we have prepared three different sentiment lexicons with the same distribution of words. The first one contains words with randomly assigned polarities (RAND). The second (BASE) and third (CPP-N) were

[5] `https://www.tripadvisor.com`

Rating	# Reviews	# Words	# Sentences	Words per Rev.	Words per Sent.
rate-1	298	25809	2249	86.61	11.47
rate-2	274	25044	2079	91.40	12.04
rate-3	659	50030	4712	75.92	10.62
rate-4	1419	94203	9123	66.39	10.32
rate-5	1389	80178	8419	57.72	9.52

Table 3: The distribution of Users' Ratings and polarity classes assigned to the reviews in the corpus.

derived directly from plWordNet using a simple averaging procedure, i.e. for every word appearing in the corpus we collect its synsets and average their polarities to derive the final polarity assignment.

For the training procedure, we plugged constructed lexicons to the LR-LSTM network. The parameters proposed in (Qian et al., 2017) were modified in order to adapt the network to our task. The number of training mini-batches was increased to $4,000$, each with 15 samples. To train the model we used *adaGrad* with the learning rate $lr = 0.05$, and the coefficients for all regularizers were the same as in (Qian et al., 2017), $\alpha = 0.5$ and $\beta = 0.0001$ respectively. To ensure that the network will be able to achieve the highest performance, we performed this training procedure multiple times for each lexicon. Vector representation for LR-LSTM was computed with *FastText* (Joulin et al., 2016) using *SkipGram* model with the size of a vector $dim = 300$ and the minimal frequency $minCount = 50$.

5.3. Results and Discussion

Table 4 presents the results obtained during experiments in the wordnet-based evaluation. We calculated precision (P), recall (R) and F-measure (F) for different coarse-grained polarity value separately: negative (NEG), positive (POS) and neutral (NEU). We compared differences between the two pairs: {BASE, CPP-N} and {CPP-N, CPP-S}. In Tab. 4 results for which differences were statistically significant are highlighted. We analysed the statistical significance of differences using paired-differences Student's t-test with a significance level $\alpha = 0.05$ (Dietterich, 1998).

Measure	BASE	CPP-N	CPP-S
P-NEG	84.01	84.58	84.73
P-NEU	92.18	**93.75**	93.66
P-POS	69.20	**83.11**	82.95
R-NEG	68.63	**75.82**	75.90
R-NEU	95.80	**97.02**	96.97
R-POS	64.64	**68.41**	67.80
F-NEG	75.52	**79.91**	79.81
F-NEU	93.95	**95.34**	95.35
F-POS	66.77	**74.99**	74.61

Table 4: Precision (P), recall (R) and F-score (F) for separate coarse-grained polarity values. BASE results are compared to CPP-N and CPP-S. Statistically significant differences are emphasised.

The naive solution (CPP-N) is significantly better than BASE in all test cases except the precision for the *negative* value. The order of neighbours classified in each iteration is not important in this case, because there was no significant difference between CPP-N and CPP-S variants. These three approaches were evaluated once again, but this time we decided to incorporate also the instances of the $ambiguous$ class, which seems to be more realistic scenario. Table 5 presents precision, recall and F-score obtained in this experiment. CPP approaches outperformed baseline solution even for the most difficult $ambiguous$ class, but in this scenario the resulting performance was slightly higher for CPP-S solution, which suggests that in some cases sorting has a positive effect on the final propagation.

Measure	BASE	CPP-N	CPP-S
P-NEG	**79.21**	73.83	74.60
P-NEU	90.45	94.37	**94.53**
P-POS	**60.52**	59.32	59.01
P-AMB	40.89	**54.41**	53.73
R-NEG	60.83	**75.35**	74.83
R-NEU	**95.24**	94.51	94.45
R-POS	57.19	**64.98**	66.78
R-AMB	33.76	41.88	**42.18**
F-NEG	68.81	74.58	**74.71**
F-NEU	92.78	94.44	**94.49**
F-POS	58.80	62.02	**62.65**
F-AMB	36.98	**47.33**	47.25

Table 5: Precision (P), recall (R) and F-score (F) for separate classes of polarity extended with propagation for *ambiguous* units.

To investigate the impact of a lexicon in sentiment recognition task we compared the precision (P), recall (R), and F-score (F) of LR-LSTM for different polarity classes – the results for this experiment are presented in table 6. Unfortunately, the convergence of adapted network was quite unstable. We decided to select the model with the highest performance on the validation dataset, in the same way as it was conducted in original work (Qian et al., 2017). The final scores for the best model were determined by averaging the values obtained from multiple executions of this network on the validation dataset.

The resulting accuracy for the models was in many cases similar (due to the class imbalance in our corpus), that is why we also decided to use more specific measures for evaluation. An observed precision and recall for $positive$ and $negative$ reviews is slightly different, especially when we compare a model using randomly generated lexicon (RAND) with the models using lexicons constructed in a controlled way (BASE, CPP-N). However, the difference between rule-based propagation and CPP is small which may suggest that hybrid methods combining neural approaches with language resources are still imperfect for this

Measure	RAND	BASE	CPP-N
P-NEG	0.761	0.821	0.880
R-NEG	0.910	0.842	0.837
F-NEG	0.828	0.831	0.858
P-POS	0.957	0.921	0.951
R-POS	0.875	0.931	0.930
F-POS	0.914	0.926	0.940

Table 6: Precision (P), recall (R) and F-score (F) for specific polarity classes, in the task of sentence-level sentiment recognition with LR-LSTM.

task and are not able to fully utilize the potential of such resources.

6. Further Works

By June 2018 we plan to complete and publish the emotive annotation of plWordNet 4.0 emo on an open licence (the intermediate results discussed here). The target size is more than 130k manually annotated LUs from all Parts of Speech. We presented an intermediate version of more than 76k manually annotated LUs in a way expressing high Inter-annotator Agreement (i.e. good consistency between annotators was achieved). This version have been already published as a part of plWordNet 3.1 emo. Next, the annotation will be automatically spread to the rest of plWordNet LUs. In parallel, the experiment-based emotive lexicon in Sentimenti[6] will be built. The method of automated selection of LUs proposed by us will be used to prepare the subsequent batches of LUs for the experiments. plWordNet descriptions of all selected LUs will be supplemented with possibly missing glosses and use examples, but not with emotive annotations, because we expect still to achieve some complementarity. We need also to solve the problem of appropriate prompting of LUs to the experiment participants, i.e. to find a way in which a certain meaning of a lemma is clearly targeted.

7. Acknowledgements

Work co-financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education and in part by the National Centre for Research and Development, Poland, under grant no POIR.01.01.01-00-0472/16, and as a result of works carried out within the project funded by the National Science Centre, Poland under the grant agreement No UMO-2015/18/M/HS2/00100.

8. Bibliographical References

Anusiewicz, J. and Skawiński, J. (1996). *Słownik polszczyzny potocznej*. Wrocław.

Black, W., Elkateb, S., Rodriguez, H., Alkhalifa, M., Vossen, P., Pease, A., and Fellbaum, C. (2006). Introducing the Arabic wordnet project. In *Proceedings of the Third International WordNet Conference*, pages 295–300. Global WordNet Association.

[6] Project supported by the National Centre for Research and Development, Poland, under grant no POIR.01.01.01-00-0472/16.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:1895–1923.

Ekman, P. (1992). An argument for basic emotions. *Cognition & Emotion*, 6(3):169–200.

Esuli, A. and Sebastiani, F. (2006). SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In *Proceedings of 5th Conference on Language Resources and Evaluation LREC 2006*, pages 417–422.

Christiane Fellbaum, editor. (1998). *WordNet – An Electronic Lexical Database*. The MIT Press.

Haniewicz, K., Rutkowski, W., Adamczyk, M., and Kaczmarek, M., (2013). *Towards the Lexicon-Based Sentiment Analysis of Polish Texts: Polarity Lexicon*, pages 286–295. Springer, Berlin, Heidelberg.

Haniewicz, K., Kaczmarek, M., Adamczyk, M., and Rutkowski, W., (2014). *Polarity Lexicon for the Polish Language: Design and Extension with Random Walk Algorithm*, pages 173–182. Springer.

Janz, A., Kocoń, J., Piasecki, M., and Monika, Z.-Z. (2017). plWordNet as a Basis for Large Emotive Lexicons of Polish. In *LTC'17 8th Language and Technology Conference*, Poznań, Poland, November. Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Kulisiewicz, M., Kajdanowicz, T., Kazienko, P., and Piasecki, M., (2015). *On Sentiment Polarity Assignment in the Wordnet Using Loopy Belief Propagation*, pages 451–462. Springer International Publishing, Cham.

Mahyoub, F. H., Siddiqui, M. A., and Dahab, M. Y. (2014). Building an arabic sentiment lexicon using semi-supervised learning. *Journal of King Saud University - Computer and Information Sciences*, 26(4):417 – 424. Special Issue on Arabic NLP.

Maks, I. and Vossen, P. (2011). Different approaches to automatic polarity annotation at synset level. In *Proceedings of the First International Workshop on Lexical Resources*, pages 62–69.

Maks, I., Izquierdo, R., Frontini, F., Agerri, R., Vossen, P., and Azpeitia, A. (2014). Generating polarity lexicons with wordnet propagation in 5 languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Maziarz, M., Piasecki, M., Rudnicka, E., Szpakowicz, S., and Kędzia, P. (2016). plwordnet 3.0 – a comprehensive lexical-semantic resource. In N. Calzolari, et al., editors, *Proc. of COLING 2016, 26th Inter. Conf. on Computational Linguistics*, pages 2259–2268. ACL.

Misiaszek, A., Kajdanowicz, T., Kazienko, P., and Piasecki, M., (2013). *Relational Propagation of Word Sentiment in WordNet*, pages 137–140. Springer Berlin Heidelberg, Berlin, Heidelberg.

Plutchik, R. (1980). *EMOTION: A Psychoevolutionary Synthesis*. Harper & Row.

Ptaszyński, M., Masui, F., Janz, A., Kocoń, J., Piasecki, M., Monika, Z.-Z., and Dybała, P. (2017). Three attempts in PolEval 2017 Sentiment Analysis Task. In *LTC'17 8th Language and Technology Conference*, Poznań, Poland, November. Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu.

Puzynina, J. (1992). *Język wartości [The language of values]*. Scientific Publishers PWN.

Qian, Q., Huang, M., Lei, J., and Zhu, X. (2017). Linguistically regularized lstm for sentiment classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1679–1689. Association for Computational Linguistics.

Stone, P. J. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. The MIT Press.

Vossen, P., Maks, I., Segers, R., and van der Vliet, H. (2008). Integrating lexical units, synsets and ontology in the cornetto database. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Vossen, P., Maks, I., Segers, R., van der Vliet, H., Moens, M., Hofmann, K., Tjong Kim Sang, E., and de Rijke, M. (2013). Cornetto: a lexical semantic database for Dutch. In P. Spyns et al., editors, *Speech and Language Technology for Dutch, Results by the STEVIN-programme*, Theory and Applications of Natural Language Processing. Springer Berlin Heidelberg.

Zaśko-Zielińska, M. and Piasecki, M. (2018). Towards emotive annotation in plWordNet 4.0. In *Proceedings of the The 9th Global WordNet Conference GWC'18*.

Zaśko-Zielińska, M., Piasecki, M., and Szpakowicz, S. (2015). A large wordnet-based sentiment lexicon for Polish. In R. Mitkov, et al., editors, *Proc. of the Conf. Recent Advances in Natural Language Processing – RANLP'2015*, pages 721—730.

SMILE Swiss German Sign Language Dataset

Sarah Ebling*, Necati Cihan Camgöz†, Penny Boyes Braem*, Katja Tissi*, Sandra Sidler-Miserez*, Stephanie Stoll†, Simon Hadfield†, Tobias Haug*, Richard Bowden†, Sandrine Tornay‡, Marzieh Razavi‡, Mathew Magimai-Doss‡,

*University of Applied Sciences of Special Needs Education (HfH), Schaffhauserstrasse 239, 8050 Zurich, Switzerland
{sarah.ebling, katja.tissi, tobias.haug}@hfh.ch, boyesbraem@fzgresearch.org, sandysidler@gmail.com

†CVSSP, University of Surrey, GU2 7XR, Guildford, UK
{n.camgoz, s.m.stoll, s.hadfield, r.bowden}@surrey.ac.uk

‡Idiap Research Institute, 1920 Martigny, Switzerland
{sandrine.tornay, marzieh.razavi, mathew}@idiap.ch

Abstract

Sign language recognition (SLR) involves identifying the form and meaning of isolated signs or sequences of signs. To our knowledge, the combination of SLR and sign language assessment is novel. The goal of an ongoing three-year project in Switzerland is to pioneer an assessment system for lexical signs of Swiss German Sign Language (*Deutschschweizerische Gebärdensprache*, DSGS) that relies on SLR. The assessment system aims to give adult L2 learners of DSGS feedback on the correctness of the manual parameters (handshape, hand position, location, and movement) of isolated signs they produce. In its initial version, the system will include automatic feedback for a subset of a DSGS vocabulary production test consisting of 100 lexical items. To provide the SLR component of the assessment system with sufficient training samples, a large-scale dataset containing videotaped repeated productions of the 100 items of the vocabulary test with associated transcriptions and annotations was created, consisting of data from 11 adult L1 signers and 19 adult L2 learners of DSGS. This paper introduces the dataset, which will be made available to the research community.

Keywords: Swiss German Sign Language (DSGS), automatic sign language assessment, sign language testing, sign language recognition and analysis, Microsoft Kinect v2, GoPro, L2 acquisition

1 Introduction

Swiss German Sign Language (*Deutschschweizerische Gebärdensprache*, DSGS) has approximately 5,500 Deaf[1] L1 users. In addition, an estimated 13,000 hearing persons use DSGS. Among them are *children of deaf adults* (CODAs), sign language interpreters, teachers, social workers, and persons otherwise interested in the language (Boyes Braem, 2012). With the exception of CODAs, they are often adult L2 learners of DSGS.

DSGS is composed of five dialects that originated in former schools for the Deaf. The differences between the dialects are primarily lexical and pertain, e.g., to semantic fields such as food (distinct signs for regional food items, such as specific breads) and date specifications (distinct signs for weekdays and months) (Boyes Braem, 1983).

The goal of the ongoing three-year SMILE *(Scalable Multimodal Sign Language Technology for Sign Language Learning and Assessment)* project in Switzerland is to pioneer an assessment system for lexical signs of DSGS that relies on sign language recognition (SLR) technology. SLR involves identifying the form and meaning of isolated signs or sequences of signs. While SLR has been applied to sign language learning (Spaai et al., 2005; Huenerfauth et al., 2017), to our knowledge, the combination of SLR and sign language assessment is novel.

The assessment system that is being developed as part of the SMILE project aims to give adult L2 learners of DSGS feedback on the correctness of the manual parameters (i.e., handshape, hand position, location, and movement) of isolated signs they produce. In its initial version, the system will include automatic feedback for a subset of a DSGS vocabulary production test consisting of 100 lexical items. The testing scenario in the project is as follows: Learners are prompted with a DSGS gloss[2] of the sign on a monitor in front of them. They then produce the sign while their production is recorded by a video camera in front of them. Following this, they receive feedback from the automatic assessment system.

State-of-the-art SLR approaches (Camgöz et al., 2017) are based on deep learning (Goodfellow et al., 2016) methods that require vast amounts of data. Therefore, to provide the SLR component of the assessment system with sufficient training samples, a large-scale dataset containing videotaped repeated productions of the 100 items of the vocabulary test with associated transcriptions and annotations was created, the SMILE Swiss German Sign Language Dataset, which consists of data from 11 adult L1 signers and 19 adult L2 signers of DSGS. This is the first DSGS dataset of its kind. The paper at hand introduces the dataset, which will be made available to the research community.

The remainder of this paper is organized as follows: Section 2 introduces existing sign language datasets and corpora. Section 3 describes the process of creating the DSGS dataset: selecting items for the vocabulary production test (Section 3.1), developing the recording software (Section 3.2), carrying out the recordings (Section 3.3),

[1] It is a widely recognized convention to use the upper-cased word *Deaf* for describing members of the linguistic community of sign language users and, in contrast, to use lower-cased *deaf* when describing the audiological state of a hearing loss (Morgan and Woll, 2002).

[2] Sign language glosses are spoken language words used as labels for semantic aspects of signs. Glosses are typically written in upper-case letters.

post-processing, transcribing, and annotating the data (Section 3.4), and distributing the resulting dataset (Section 3.5). Section 4 offers a conclusion and outlook.

2 Related Work

In the context of language, a corpus denotes a "finite-sized body of machine-readable text, sampled in order to be maximally representative of the language variety under consideration" (McEnery and Wilson, 2001, p. 32), where *text* may refer to original written text, transcriptions of speech, and transcriptions of sign language. The units of interest in the assessment system in our project (cf. Section 1) are not continuous utterances but isolated signs. Transcribed recordings of repeated productions of these signs form a dataset.

Several sign language corpora and datasets exist, some created for the purpose of conducting linguistic analyses, and some to serve as training data for sign language technology systems, e.g., SLR systems. Table 1 provides an overview of different sign language corpora and datasets. Depending on the field of study, researchers prioritized different aspects of data collection. Linguists mainly focused on having large vocabularies to be able to understand and extract underlying rules of sign languages. On the other hand, SLR researchers concentrated on having multiple repetitions of sign samples from different signers to learn distinctive signer-independent representations using statistical machine learning algorithms.

Most SLR methods begin with extracting the upper body pose information, which is a challenging task due to the color ambiguity between the signers and the background (Cooper et al., 2011). With the availability of consumer depth cameras, such as Microsoft Kinect (Zhang, 2012), and real-time pose estimation algorithms (Shotton et al., 2013; Cao et al., 2017), SLR researchers created datasets containing human pose information, which accelerated the field.

Due to the articulated nature of sign languages, datasets which are collected using generic video cameras suffer from motion blur. This limits both the linguistic analysis and SLR algorithms, which try to investigate and learn the manual attributes of signs, respectively. In addition, the estimated pose becomes noisy where the performed signs contain rapid upper body motion. To address this limitation, we used a diverse set of visual sensors including high speed and high resolution GoPro video cameras, and a Microsoft Kinect v2 depth sensor to collect the SMILE dataset.

3 Compilation of the SMILE Swiss German Sign Language Dataset

3.1 Selection of Test Items

As described in Section 1, the assessment system that includes an SLR component in our project is based on a DSGS vocabulary production test consisting of 100 individual signs. In addition, the test features five practice items that are excluded from subsequent processing. The test is aimed at beginning adult L2 learners of DSGS, targeting level A1 of the Common European Framework of Reference for Languages (CEFR) (Council of Europe, 2009). Learning materials for some parts of level A1 have been developed for DSGS (Boyes Braem, 2004a; Boyes Braem, 2004b; Boyes Braem, 2005a; Boyes Braem, 2005b). The basis of the development of the DSGS vocabulary production test was a list of glosses of 3,800 DSGS signs taken from these materials.

The work of arriving at a set of 105 test items (100 main items plus five practice items) was carried out by a team of Deaf and hearing sign language researchers and involved both excluding certain categories of signs, similar to what had previously been done for a lexical comparison study involving DSGS (Ebling et al., 2015), and prioritizing specific signs. In particular, signs denoting persons (e.g., CHARLIE-CHAPLIN), organizations (e.g., GALLAUDET), places (e.g., AUSTRALIEN 'AUSTRALIA'), and languages (e.g., DEUTSCH 'GERMAN') were excluded. This was because many of these signs are borrowed from other sign languages, and some are initialized signs, i.e., signs in which the handshape corresponding to the first letter of the spoken language word in the DSGS manual alphabet is used. For example, the sign ASIEN ('ASIA') is produced by combining the letter A from the DSGS manual alphabet with a circular movement.

Body-part signs (e.g., NASE 'NOSE') as well as pronouns (e.g., DU 'YOU [sg.]') were also discarded, as they mostly correspond to indexical (pointing) signs in DSGS. Number signs were removed since they tend to have many variants, particularly numbers greater than ten. For example, there are three variants for the number sign ELF ('ELEVEN') in DSGS. Primarily fingerspelled components were also removed from the list, e.g., signs for the months of the year (such as JANUAR 'JANUARY' consisting of the sign J from the DSGS manual alphabet), as assessing fingerspelling production was not among the core aims of the final test. Signs composed of multiple successive segments were also eliminated; this was because the segments they consisted of were often also contained in the list of 3,800 signs as individual lexemes. For example, the list contained the sign ABENDESSEN ('DINNER') as well as the signs ABEND ('EVENING') and ESSEN ('MEAL'). Signs marked as being old variants were also ignored (e.g., an earlier form of the sign BAUERNHAUS 'FARMHOUSE'), as current-day DSGS learners could not be expected to know them. Like Vinson et al. (2008) and Mayberry et al. (2013), who compiled lists of signs to be used in acceptance/familiarity studies, we excluded productive forms from our list. However, unlike in these studies, our reason for exclusion was that we anticipated it to be hard to elicit the correct forms for productive signs using glosses as prompts. For example, a test taker might not know which form to sign from a gloss like GEHEN-FUSS ('GO-FOOT').

We further removed signs that appeared in less than four of the five DSGS dialects from the list of item candidates to ensure high familiarity of the learners with the signs. Since the items to be selected formed part of a sign production test, our goal was to test production of as many different sign forms as possible. We therefore reduced homonymy in the following way: We identified groups of form-identical signs and for each group gave preference to the sign whose

Study	Language	Research Field	# Items	# Samples	# Signers	Acquisition Tool
The NGT Corpus (Crasborn and Zwitserlood, 2008)	SL of the Netherlands	Linguistic	N/A	15 Hours	92	Video Camera
ATIS (Bungeroth et al., 2008)	Multilingual	Linguistic	292	595 Sentences	N/A	Video Camera
DGS Corpus (Prillwitz et al., 2008)	German SL	Linguistic	N/A	2.25 million Tokens	328	Video Camera
BSL Corpus (Schembri et al., 2013)	British SL	Linguistic	N/A	40000 Lexical Items	249	Video Camera
LSE-SIGN (Gutierrez-Sigut et al., 2015)	Spanish SL	Linguistic	2400	2400 Samples	2	Video Camera
AUSLAN (Johnston, 2010)	Australian SL	Linguistic	N/A	1100 Videos	100	Video Camera
RWTH-BOSTON (Dreuw et al., 2008)	American SL	Linguistic, SLR	483	843 Sentences	4	Video Camera
ASSLVD (Athitsos et al., 2008)	American SL	Linguistic, SLR	3000	12000 Samples	4	Video Camera
Dicta-Sign (Matthes et al., 2012)	Multilingual	Linguistic, SLR	N/A	6-8 Hours (/Participant)	16-18 (/Language)	Video Camera
SIGNUM (von Agris and Kraiss, 2010)	German SL	SLR	450	33210 Sequences	25	Video Camera
CopyCat (Zafrulla et al., 2010)	American SL	SLR	22	420 Phrases	5	Accelerometer & VC
RWTH-PHOENIX-Weather (Forster et al., 2014)	German SL	SLR	1231	6931 Sentences	9	Video Camera
DEVISIGN (Chai et al., 2015)	Chinese SL	SLR	2000	24000 Samples	8	Kinect v1 Sensor
BosphorusSign (Camgöz et al., 2016)	Turkish SL	SLR	636	24161 Samples	6	Kinect v2 Sensor

Table 1: Existing sign language corpora and datasets

Removed:
Name signs: persons, organizations, places, languages
Body-part signs
Pronouns
Number signs
Primarily fingerspelled components
Signs composed of multiple successive segments
Old signs
Productive signs
Signs appearing in less than four of the five DSGS dialects
Homonyms
Signs overlapping with co-speech gestures
Signs with ambiguous German glosses
Signs with occurrence <3 in DSGS corpora
Prioritized:
Signs with concepts in Efthimiou et al. (2009)
Signs for concepts included in all of the following studies: Vinson et al. (2008), Mayberry et al. (2013), and Efthimiou et al. (2009)

Table 2: Item selection for the DSGS vocabulary production test

meaning was contained in a list of 1,000 common sign language concepts (Efthimiou et al., 2009). In cases where several homonyms were contained in this list, we gave preference to the one with the highest overall token count in the small DSGS corpora currently available. We also eliminated signs that overlapped with co-speech gestures, such as SUPER corresponding to a thumbs-up gesture. Chen Pichler (2009) was among the first to point out that gestures represent a "source for phonological transfer" in L2 sign acquisition (p. 39). In this sense, excluding signs that resembled co-speech gestures represented another step towards ensuring that what was being tested was sign language as opposed to spoken language production. Glosses whose underlying German words were semantically ambiguous (e.g., AUFNEHMEN can have the meaning of both *recording* and *including*, LEICHT can denote the concepts *lightweight* and *easy*) were also discarded. We thus tried to ensure that glosses alone would be sufficient as prompts in the test setting. Lastly, we removed signs that occurred fewer than three times in the DSGS corpora available.

From the resulting set, we gave direct preference to signs whose meanings appeared in the list of 1,000 common sign language concepts (Efthimiou et al., 2009) and well as preference to signs that appeared in all three sign/concept lists mentioned previously (Vinson et al., 2008; Mayberry et al., 2013; Efthimiou et al., 2009). Table 2 summarizes the item selection process.

3.2 Recording Software and Setup

To obtain high quality sign samples, we used a diverse set of visual sensors: a Microsoft Kinect v2 depth sensor, two GoPro Hero 4 Black video cameras (one in high speed mode and the other in high resolution mode), and three webcams. The GoPro cameras and the Microsoft Kinect sensor were fitted on a rigid mount. The mount was placed in front of the signer facing the signing space, and three webcams were placed to the left, the right, and the top of the signer to capture the signs performed from different angles. Sample recording output from all of the sensors and our recording setting can be seen in Figure 1 and Figure 2, respectively.

We modified the publicly available BosphorusSign Recording Software due to its user-friendly interface and color-coded multi-view signer-operator interaction scheme, which are described in detail in Camgöz et al. (2016). To synchronize the capture from multiple sensors, we first developed a recording driver to control webcams using EmguCV (Emgu, 2013). In addition, we developed an API for GoPro cameras in C#, which allows the recording software to have access to all of the functionality of the cameras. The interface was modified to give the operator control over the GoPros. The modified BosphorusSign Recording Software interface can be seen in Figure 3.

The recording software allows for capturing video streams from all the sensors simultaneously. Given a recording script, which contains a set of items and their corresponding prompts, the operator starts a recording session by us-



Figure 1: Sample recording output from the video cameras.



Figure 2: Recording setting.

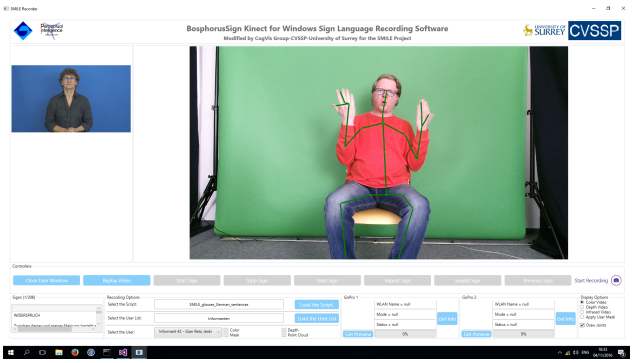


Figure 3: Recording interface.

ing the *Start Recording* button (cf. Figure 3). By clicking the *Start Sign* button, the operator indicates the signer to start performing a sign while annotating the beginning of the sample over all streams. When the sign is performed, the operator clicks the *Stop Sign* button to annotate the end of the sample. The operator can then choose to proceed to the next item on the list by using the *Next Sign* button or can request the repetition of the sign by clicking the *Repeat Sign* button. The operator can also use the *Invalid Sign* button to annotate a sign sample as invalid. Once the recording session is finished, the operator presses the *Stop Recording* button and stops the capture on all of the sensors.

3.3 Recording Procedure

The focus of the data collection described in this paper was on obtaining training data for the SLR system. Therefore, in an attempt to reduce the number of instances in which no sign was produced at all, the participating signers were provided with the test items prior to the recordings in the form of a list of glosses with accompanying German example sentences.[3] Table 3 shows a selection of glosses along with context examples. The sentences had been gathered from a DSGS online lexicon[4] and, where necessary, shortened and modified. The rationale behind providing German example sentences in addition to DSGS glosses was to further reduce any semantic ambiguity remaining even after clearly ambiguous glosses had been eliminated in the item selection process (cf. Section 3.1).

Upon recording, participants were asked to perform each sign in three separate passes. The glosses with German example sentences served as prompts for the first two passes, while the prompt for the third pass was a video of a signer performing the sign. The video corresponded to the base form of the sign in a DSGS lexicon (Boyes Braem, 2001).

While the DSGS vocabulary production test is ultimately aimed for use by L2 learners, the goal of the recordings described here was to obtain both L1 and L2 data for training the recognition system. In total 40, 20 L1 and L2 signers each participated in the recordings (due to technical problems, not all recordings were used for the dataset; cf. Section 3.4). The L1 participants were recruited by the Deaf members of our project team; they were native DSGS signers and/or trained DSGS instructors.[5] To recruit L2 participants, a call for participation was released via various channels, such as e-mail, social media, and personal contacts. L2 participants had to have completed four courses in the course framework of the Swiss Deaf Association corresponding to parts of CEFR level A1. Both L1 and L2 participants were asked to complete a background questionnaire prior to the recordings. The background questionnaire was a modified version of a questionnaire developed in the DGS Corpus Project (Hanke, 2017). Participants gave their informed consent for the video recordings and collection of background information as well as usage thereof in the SMILE project.[6] In addition, they were offered the option of giving informed consent for passing the data on to other researchers and to the public via a secure web interface. All but two participants gave their consent for these latter options as well.

L1 participants were paid by the hour, L2 participants were given the choice between getting paid and receiving detailed video feedback on their sign productions from the Deaf members of our project team, who are also trained

[3] In a recent test of the assessment scenario of the project, no sign was produced for 20.56% of all prompts using a nearly identical item set (Haug, 2017).

[4] `https://signsuisse.sgb-fss.ch/` (last accessed September 7, 2017)

[5] Limiting the L1 subject pool to native signers was not an option for DSGS due to the small population of signers upon which one could draw.

[6] A DSGS version of the informed consent had been made available beforehand.

Gloss	Example sentence
ANGESTELLT_1A ('EMPLOYED_1A')	Sie ist in einer grossen Firma angestellt. ('She is employed by a large corporation.')
THEATER_1A ('THEATRE_1A')	Das Theater findet in Basel statt. ('The theatre play takes place in Basel.')
WARTEN_1A ('WAIT_1A')	Ich warte, bis der Arzt kommt. ('I am waiting for the doctor to come.')

Table 3: Glosses and example sentences

DSGS instructors. Since participants were expected to perform 300 signs, it was decided that they should sit on a chair rather than remain standing while signing. In the introductory message signed by a Deaf member of our team and supplemented with German subtitles, participants were told that the goal of the study was to obtain information about natural variation in the production of isolated signs and that following five practice items, they were asked to sign 100 signs three times, the first and second time with glosses as prompts, the third time with a model video of a signer performing the sign. For the third pass, participants were asked to mirror the sign they saw in the video, not repeat a potential dialect variant that they might have produced in the previous two passes. They were told that the order of the signs in the three passes was different and were asked to return to a neutral position after each sign. They were not required to look into a particular camera but rather direct their eye gaze towards the general area of the cameras. Participants were specifically instructed to sign the base forms of the lexical items, not modified versions based on the context evoked in the example sentences. Recordings lasted between 30 and 45 minutes.

3.4 Transcription and Annotation

In the context of sign languages, *transcription* usually refers to the process of providing a written version of signing recorded on video, while *annotation* describes the enhancement of the primary data with additional information, e.g., of linguistic nature. Both steps, transcription and annotation, provide valuable information for an SLR system. To perform transcription and annotation on the videos obtained through the procedure outlined in Sections 3.2 and 3.3, the videos were postprocessed and imported into iLex, a software tool for creating and analyzing sign language lexicons and corpora (Hanke and Storz, 2008). In iLex, all occurrences of a sign in a transcript (sign tokens) are linked back to their sign type in the lexicon, and changes of the sign type affect all sign tokens in all transcripts.

For each recording, three videos corresponding to three of the six camera perspectives (cf. Figure 1 for an example of all perspectives) were imported and synchronized based on information on the starting and stopping times of the cameras (cf. Section 3.2). Participant and movie metadata were also automatically imported into iLex. One transcript was created for each recording. Based on information on the starting and stopping times of the individual signs (cf. Section 3.2), a tier holding the targeted signs as tags and another tier recording for each tag the pass it belonged to were introduced. The team of Deaf and hearing sign language researchers then manually postcorrected the sign tag boundaries where necessary.

Table 4 shows the transcription/annotation scheme. The scheme consists of twelve tiers. As detailed above, information for the first two tiers, "Pass" and "Target sign", was automatically imported and manually postcorrected. The team manually annotated information for the remaining tiers for the second pass. If a sign was produced multiple times in this pass (recall from Section 3.2 that self-correction was permitted during the recordings), only the last attempt was considered. A four-eyes principle was observed, i.e., each annotation produced by one annotator was checked by another. In addition, cases for which the annotators were not certain were discussed in weekly group meetings.

The "Sign produced" tier (cf. Table 4) records the glosses of the signs actually performed. "Category of sign produced" is a classification of the productions in this tier into one of six categories:

1. **Same lexeme as target sign:** same meaning, same form
2. **Same lexeme as target sign:** same meaning, slightly different form
3. **Same lexeme as target sign:** same meaning, different form
4. **Same lexeme as target sign:** slightly different meaning, slightly different form
5. **Different lexeme than target sign:** same meaning, different form
6. **Different lexeme than target sign:** different meaning, different form

Instances of Category 1 are sign productions that are identical to the target sign, i.e., to the base form as produced in the model video (cf. Section 3.3). Sign productions assigned to Category 2 have the same meaning as the target sign and a slightly different but acceptable form.[7] For example, the sign SPRACHE_1A ('LANGUAGE_1A') might be produced in a slightly different location, resulting in a sign denoted by the *qualified gloss*[8] SPRACHE_1A'loc in the "Sign produced" tier. Members of Category 3 were judged by the annotators to differ clearly and significantly from acceptable variant forms (cf. below for the link between categories and test decisions, i.e., decisions

[7] These instances are sometimes called *allophonic variants*.

[8] Cf. Konrad et al. (2012) for an introduction to qualifiers and qualified glosses.

No.	Tier name	Description
1	Pass	“first”, “second”, or “third”
2	Target sign	Which sign was to be produced?
3	Sign produced	Which sign was actually produced?
4	Category of sign produced	One of six categories
5	Confidence	Confidence of assignment in Tier 4
6	Parameter(s) different	Deviating manual parameter(s)
7	Handedness different	Deviating handedness
8	Hand configuration different	Deviating hand configuration
9	Comment parameter	(free text)
10	Comment style	(free text)
11	HamNoSys (Prillwitz et al., 1989) of target sign	automatically inserted from iLex lexicon
12	HamNoSys of sign produced	HamNoSys notation of sign produced in Tier 3

Table 4: SMILE transcription/annotation scheme

regarding the correctness of the productions). For example, if SPRACHE_1A, which has an open handshape, were to be produced with a closed handshape, this occurrence would be labeled with Category 3 and notated as SPRACHE_1A’hdf in the “Sign produced” tier. Instances of Category 4 are morphophonemic/semantic variants, e.g., modifying SPRACHE_1A from singular to plural, resulting in a slightly different form and slightly different meaning. Sign productions that represent dialect variants are assigned to Category 5, indicating identical meanings but different forms.[9] Sign productions with both an entirely different meaning and form, e.g., productions of the sign BAUM_1A (‘TREE_1A’) for the prompt SPRACHE_1A, are assigned to Category 6.

Table 5 shows the mapping of category assignments to test decisions: Members of Categories 1, 2, 4, and 5 are rated as correct, while members of Categories 3 and 6 are considered incorrect.

A “Confidence” tier (cf. Table 4) records the annotators’ joint confidence of the assignment of Categories 2 and 3 in the “Category of sign produced” tier, with “certain” and “uncertain” as possible values. Our analysis showed that the distinction between permissible variants (Category 2) and incorrect productions (Category 3) of a sign was in some cases especially challenging. Therefore, cases for which the team was uncertain were extracted for presentation to a group of seven outside sign language experts.

For cases in which the sign form produced does not coincide with the target form, a “Parameter different” tier (cf. Table 4) records the deviating parameters, with all cross-combinations of parameters as possible values (“handshape”; “handshape and hand position”; “handshape, hand position, and location”; etc.).

If the number of hands of the target and produced sign differ, this is notated by indicating the handedness of the produced sign as either “one-handed”, “two-handed symmetrical”, or “two-handed dominance”. Similarly, differing hand configuration is recorded along the following values: “one hand next to the other”, “dominant hand on top of non-dominant”, “non-dominant hand on top of dominant”, “dominant hand closer to body”, “dominant hand further away from body”, “one hand crossing the other”, “hands interlocked”, “hands without contact with each other”, and “hands without contact with the body”.

Two tiers allow for comments pertaining to the articulation of the parameters (“Comment parameter”) and to signing style (“Comment style”).

Finally, the last two tiers contain Hamburg Notation System for Sign Languages (HamNoSys) (Prillwitz et al., 1989) notations of the target sign (“HamNoSys target sign”, inserted directly from the lexicon) and the sign produced (“HamNoSys sign produced”). HamNoSys consists of approximately 200 symbols describing the manual parameters hand shape, hand position (with finger direction and palm orientation as sub-parameters), location, and movement. The symbols together constitute a Unicode font.

The second pass of the recordings was completely annotated for 30 transcripts, of which 11 are L1 transcripts and 19 are L2 transcripts. Technical issues were the reason why not all 40 recordings were transcribed/annotated. Figure 4 shows a sample iLex transcript.

3.5 Distribution

The SMILE Swiss German Sign Language Dataset will be publicly available for academic purposes upon signing an end user license agreement. We will share all of the modalities that were collected using the Microsoft Kinect v2 sensor, namely color videos, depth maps, user masks, and 3D pose information. Other color video streams such as High Definition (4K Resolution) and High Speed (240 frames per second) GoPro and Webcam streams will also be made available. Furthermore, to encourage and to expedite sign language recognition research on our dataset, we will distribute body pose, facial landmarks, and hand pose information extracted using the state-of-the-art deep-learning-based key point detection library OpenPose (Cao et al., 2017). For linguistic research purposes, we will release all of our iLex annotations including sign form and category annotations, which were mentioned in Section 3.4. The contents of the dataset that will be released can be seen in Table 6. The dataset will be available at `https://www.idiap.ch/project/smile`.

[9] Recall from Section 1 that DSGS is composed of five dialects and that the items of the DSGS vocabulary production test at hand are known to appear in at least four of these five dialects.

Category	Same lexeme as target sign?	Same meaning?	Same form?	Test decision
1	yes	yes	yes	correct
2	yes	yes	slightly different	correct
3	yes	yes	no	incorrect
4	yes	slightly different	slightly different	correct
5	no	yes	no	correct
6	no	no	no	incorrect

Table 5: Link between category assignments and test decisions

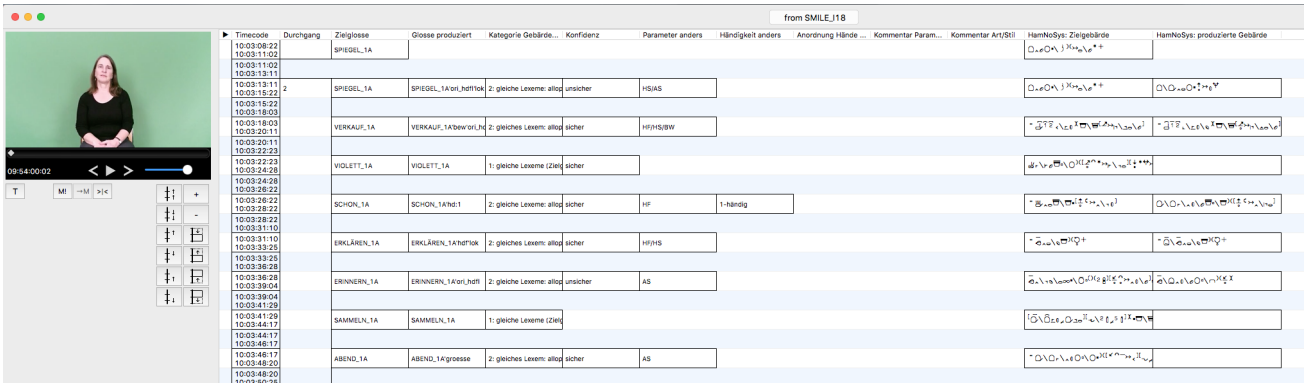


Figure 4: Sample transcript in iLex

Modality	File Type	Resolution	Content
Kinect Color Video	.MP4 Video File	1920x1080 Pixels @ 30 FPS	24bpp Image Sequence
GoPro Color Video [HD]	.MP4 Video File	3840x2160 Pixels @ 30 FPS	24bpp Image Sequence
GoPro Color Video [HS]	.MP4 Video File	1280x720 Pixels @ 240 FPS	24bpp Image Sequence
Webcam Color Videos	.MP4 Video File	1280x720 Pixels @ 30 FPS	24bpp Image Sequence
Depth Map	.RAR Binary File	512x424 Pixels @ 30 FPS	16bpp Image Sequence
User Mask	.RAR Binary File	512x424 Pixels @ 30 FPS	8bpp Binary Image Sequence
Kinect Pose Information	.CSV File	25 Joints	3D Joint Coordinates and Angles
Body Pose Information	.JSON File	18 Joints	2D Joint Coordinates and Confidences
Facial Landmarks	.JSON File	70 Joints	2D Joint Coordinates and Confidences
Hand Pose Information	.JSON File	2x21 Joints	2D Joint Coordinates and Confidences
iLex Annotations	.XML File	(not applicable)	Linguistic Annotations

Table 6: Contents of the SMILE Swiss German Sign Language Dataset [HS: High Speed, HD: High Definition]

4 Summary and Future Directions

This paper has introduced the SMILE Swiss German Sign Language Dataset, a large-scale dataset containing videotaped repeated productions of the 100 items of a DSGS vocabulary production test with associated transcriptions and annotations. The dataset was created for use in a project whose goal is to pioneer an assessment system for lexical signs of DSGS that relies on sign language recognition. In its initial version, the system includes automatic feedback for a subset of the items of the vocabulary test. A prototype of the system is currently under development. Following this, the system will be extended to provide feedback for the complete set of items of the vocabulary test.

5 Acknowledgements

This work was funded by the SNSF Sinergia project "Scalable Multi-modal Sign Language Technology for Sign Language Learning and Assessment (SMILE)", grant agreement number CRSII2_160811. We would like to thank the three students of the DSGS interpreting training program who helped collect the data as well as the external DSGS experts who took part in our online and focus group studies. We are greatly indebted to Thomas Hanke for continuing to share his vast expertise in sign language technology. We would also like to thank NVIDIA Corporation for their GPU grant.

6 Bibliographical References

Athitsos, V., Neidle, C., Sclaroff, S., Nash, J., Stefan, A., Yuan, Q., and Thangali, A. (2008). The American Sign Language Lexicon Video Dataset. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–8.

Boyes Braem, P. (1983). Studying Swiss German Sign Language dialects. In *Proceedings of the 3rd International Symposium on Sign Language Research (SLR)*, pages 247–253, Rome, Italy.

Boyes Braem, P. (2001). A multimedia bilingual database for the lexicon of Swiss German Sign Language. *Sign Language & Linguistics*, 4(1/2):133–143.

Boyes Braem, P. (2004a). *Gebärdensprachkurs Deutschschweiz, Stufe 1: Linguistischer Kommentar*. SGB-FSS, Zurich, Switzerland. CD-ROM.

Boyes Braem, P. (2004b). *Gebärdensprachkurs Deutschschweiz, Stufe 2: Linguistischer Kommentar*. SGB-FSS, Zurich, Switzerland. CD-ROM.

Boyes Braem, P. (2005a). *Gebärdensprachkurs Deutschschweiz, Stufe 3: Linguistischer Kommentar*. SGB-FSS, Zurich, Switzerland. CD-ROM.

Boyes Braem, P. (2005b). *Gebärdensprachkurs Deutschschweiz, Stufe 4: Linguistischer Kommentar*. SGB-FSS, Zurich, Switzerland. CD-ROM.

Boyes Braem, P. (2012). Overview of research on signed languages of the Deaf. Lecture held at the University of Basel. Retrieved from `http://www.signlangcourse.org` (last accessed November 13, 2015).

Bungeroth, J., Stein, D., Dreuw, P., Ney, H., Morrissey, S., Way, A., and van Zijl, L. (2008). The ATIS Sign Language Corpus. In *International Conference on Language Resources and Evaluation (LREC)*.

Camgöz, N. C., Kindiroglu, A. A., Karabuklu, S., Kelepir, M., Ozsoy, A. S., and Akarun, L. (2016). BosphorusSign: A Turkish Sign Language Recognition Corpus in Health and Finance Domains. In *International Conference on Language Resources and Evaluation (LREC)*.

Camgöz, N. C., Hadfield, S., Koller, O., and Bowden, R. (2017). Subunets: End-to-end hand shape and continuous sign language recognition. *IEEE International Conference on Computer Vision (ICCV)*.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*.

Chai, X., Wanga, H., Zhoub, M., Wub, G., Lic, H., and Chena, X. (2015). DEVISIGN: Dataset and Evaluation for 3D Sign Language Recognition. Technical report, Beijing, Technical Report.

Chen Pichler, D. (2009). Sign production in first-time hearing signers: A closer look at handshape accuracy. *Cadernos de Saúde, Número especial, Línguas gestuais*, 2:37–50.

Cooper, H., Holt, B., and Bowden, R. (2011). Sign language recognition. In *Visual Analysis of Humans*, pages 539–562. Springer.

Council of Europe. (2009). *Common European framework of reference for languages: learning, teaching, assessment*. Cambridge University Press, Cambridge.

Crasborn, O. A. and Zwitserlood, I. (2008). The Corpus NGT: An Online Corpus for Professionals and Laymen. In *3rd Workshop on the Representation and Processing of Sign Languages (LREC)*, pages 44–49.

Dreuw, P., Neidle, C., Athitsos, V., Sclaroff, S., and Ney, H. (2008). Benchmark Databases for Video-Based Automatic Sign Language Recognition. In *International Conference on Language Resources and Evaluation (LREC)*, pages 1–6.

Ebling, S., Konrad, R., Boyes Braem, P., and Langer, G. (2015). Factors to consider when making lexical comparisons of sign languages: Notes from an ongoing study comparing German Sign Language and Swiss German Sign Language. *Sign Language Studies*, 16(1):30–56.

Efthimiou, E., Fotinea, S., Vogler, C., Hanke, T., Glauert, J., Bowden, R., Braffort, A., Collect, C., Maragos, P., and Segouat, J. (2009). Sign language recognition, generation, and modelling: A research effort with applications in Deaf communication. In C. Stephanidis, editor, *Universal Access in Human-Computer Interaction*, pages 21–30. Springer, Berlin, Germany.

Emgu, C. (2013). Emgu cv: Opencv in .net (c#, vb, c++ and more). *Online: http://www. emgu. com*.

Forster, J., Schmidt, C., Koller, O., Bellgardt, M., and Ney, H. (2014). Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In *International Conference on Language Resources and Evaluation (LREC)*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT press.

Gutierrez-Sigut, E., Costello, B., Baus, C., and Carreiras, M. (2015). LSE-Sign: A Lexical Database for Spanish Sign Language. *Behavior Research Methods*, pages 1–15.

Hanke, T. and Storz, J. (2008). iLex: A database tool for integrating sign language corpus linguistics and sign language lexicography. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC)*, pages 64–67, Marrakech, Morocco.

Hanke, T. (2017). Wörterbuch ohne Wörter? Zum Entstehen eines Wörterbuches der Deutschen Gebördensprache. In Heidelberger Akademie der Wissenschaften, editor, *Jahrbuch der Heidelberger Akademie der Wissenschaften für 2016*, pages 84–88. Universitätsverlag Winter, Heidelberg.

Haug, T. (2017). Development and Evaluation of Two Vocabulary Tests for Swiss German Sign Language. Master's thesis, University of Lancaster. Submitted.

Huenerfauth, M., Gale, E., Penly, B., Pillutla, S., Willard, M., and Hariharan, D. (2017). Evaluation of Language Feedback Methods for Student Videos of American Sign Language. *ACM Transactions on Accessible Computing*, 10(1).

Johnston, T. (2010). From Archive to Corpus: Transcription and Annotation in the Creation of Signed Language Corpora. *International Journal of Corpus Linguistics*, 15(1):106–131.

Konrad, R., Hanke, T., König, S., Langer, G., Matthes, S., Nishio, R., and Regen, A. (2012). From form to function: A database approach to handle lexicon building and spotting token forms in sign languages. In *Proceedings of the 5th LREC Workshop on the Representation and Processing of Sign Languages*, pages 87–94, Istanbul, Turkey.

Matthes, S., Hanke, T., Regen, A., Storz, J., Worseck, S., Efthimiou, E., Dimou, A.-L., Braffort, A., Glauert, J., and Safar, E. (2012). Dicta-Sign: Building a Multilingual Sign Language Corpus. In *5th Workshop on the*

Representation and Processing of Sign Languages: Interactions Between Corpus and Lexicon.

Mayberry, R., Hall, M., and Zvaigzne, M. (2013). Subjective frequency ratings for 432 ASL signs. *Behavior Research Methods*, 46(2):526–39.

McEnery, T. and Wilson, A. (2001). *Corpus Linguistics*. Edinburgh University Press, Edinburgh, Scotland, 2nd edition.

Morgan, G. and Woll, B. (2002). The development of complex sentences in British Sign Language. In Gary Morgan et al., editors, *Directions in Sign Language Acquisition: Trends in Language Acquisition Research*, pages 255–276. John Benjamins, Amsterdam, Netherlands.

Prillwitz, S., Leven, R., Zienert, H., Hanke, T., and Henning, J. (1989). *HamNoSys: Version 2.0: An Introductory Guide*. Signum, Hamburg, Germany.

Prillwitz, S., Hanke, T., König, S., Konrad, R., Langer, G., and Schwarz, A. (2008). DGS Corpus Project–Development of a Corpus Based Electronic Dictionary German Sign Language/German. In *3rd Workshop on the Representation and Processing of Sign Languages: Construction and Exploitation of Sign Language Corpora*, page 159.

Schembri, A., Fenlon, J., Rentelis, R., Reynolds, S., and Cormier, K. (2013). Building the British Sign Language Corpus.

Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., and Moore, R. (2013). Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124.

Spaai, G., Lichtenauer, J., Hendriks, E., de Ridder, H., Arendsen, J., de Ridder, H., Fortgens, C., Bruins, M., and Elzenaar, M. (2005). Elo: An electronic learning environment for practising sign vocabulary by young deaf children. In *Proceedings of the International Congress for Education of the Deaf (ICED)*.

Vinson, D., Cormier, K., Denmark, T., Schembri, A., and Vigliocco, G. (2008). The British Sign Language (BSL) norms for age of acquisition, familiarity, and iconicity. *Behavior Research Methods*, 40(2):1079–1087.

von Agris, U. and Kraiss, K.-F. (2010). SIGNUM Database: Video Corpus for Signer-Independent Continuous Sign Language Recognition. In *4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, pages 243–246.

Zafrulla, Z., Brashear, H., Hamilton, H., and Starner, T. (2010). A Novel Approach to American Sign Language (ASL) Phrase Verification using Reversed Signing. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 48–55.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10.

IPSL: A Database of Iconicity Patterns in Sign Languages. Creation and Use

Vadim Kimmelman, Anna Klezovich, George Moroz
University of Amsterdam, Higher School of Economics, Higher School of Economics
Amsterdam (the Netherlands), Moscow (Russia), Moscow (Russia)
{vadim.kimmelman, belkannkl, agricolamz}@gmail.com

Abstract

We created the first large-scale database of signs annotated according to various parameters of iconicity. The signs represent concrete concepts in seven semantic fields in nineteen sign languages; 1542 signs in total. Each sign was annotated with respect to the type of form-image association, the presence of iconic location and movement, personification, and with respect to whether the sign depicts a salient part of the concept. We also created a website: https://sl-iconicity.shinyapps.io/iconicity_patterns/ with several visualization tools to represent the data from the database. It is possible to visualize iconic properties of separate concepts or iconic properties of semantic fields on the map of the world, and to build graphs representing iconic patterns for selected semantic fields. A preliminary analysis of the data shows that iconicity patterns vary across semantic fields and across languages. The database and the website can be used to further study a variety of theoretical questions related to iconicity in sign languages.

Keywords: iconicity, sign language, linguistic typology

1. Iconicity Patterns in Sign Languages

Iconicity is a fundamental property of human languages, both in the spoken and signed modalities (Perniss et al., 2010). Following Taub (2012), we define iconicity as a presence of a mapping (resemblance) between a mental image and the phonetic form of a sign (that is, sounds in spoken languages, or handshape, movement, and location in sign languages). For instance, the sound of the English word *ding* resemble the sound of a bell, and the handshape of the sign TREE in Russian Sign Language (RSL) (the palm with all fingers outstretched) resembles a crown of a tree—therefore, the word and the sign are iconic.

For various reasons (see Taub (2012) for a discussion), until recently, iconicity has not been actively investigated even by researchers working on sign languages. However, in recent years, a theoretical model of iconicity has been suggested (Taub, 2001), and several empirical studies have compared iconicity in different sign languages and in gestures of hearing people (Brentari et al., 2015; Padden et al., 2013; Padden et al., 2015).

Taub (2001) develops the Analogue-Building Model of Linguistic Iconicity.[1] According to this model, iconic encoding of a concept comprises of three steps: image selection (a particular image representing a concept is selected), schematization (the image is simplified), and encoding (the image is encoded using the linguistic means). For instance, to create an iconic sign for the concept 'tree', first an image of a prototypical tree is selected (a tree with a trunk and crown), then it is schematized (for instance, the schematized representation does not refer to leaves), and then it is encoded (e.g, the hand is used to represent the crown, and the arm to represent the trunk).

Based on this model, Taub (2012) discusses that iconic signs can be further classified according to the type of the concept-image association (how the image is selected to represent the concept) and the image-form association (how the phonetic shape is selected to represent the image). For instance, concerning concept-image association, sometimes only a part of the object is represented by the sign (e.g. representing the whiskers of a cat for the concept 'cat'). Concerning the image-form association, in some cases we observe *object* handshapes where the hand represents the object itself (e.g. a flat hand representing a knife, as in the Greek sign KNIFE), while in others we observe *handling* handshapes where the hand represents the hand of an agent handling an object (e.g. a fist representing a hand of someone holding a knife, as in the British sign KNIFE)).

Several recent studies focused on this latter typology, specifically, on the choice between the object and handling handshapes in iconic signs. For instance, Padden et al. (2013) compare iconic signs for instruments in American Sign Language (ASL), Al-Sayyid Bedouin Sign Language (ABSL), and New Zealand Sign Language (NZSL), as well as gestures of hearing Americans and Bedouins. They find that both groups of hearing gesturers have very strong tendency to produce handling handshapes for instruments, while ASL and ABSL signers have a strong tendency to use object handshapes in the corresponding signs; NZSL turned out to be in between in this respect. In a different study, Padden et al. (2015) find that signers of ASL but not gesturers use the object/handling difference to encode the difference between nouns and verbs. Brentari et al. (2015) compare signers and gesturers from two countries (the US and Italy) and find similarities and differences between the groups. The research so far thus indicates that (1) there is typological variation between sign languages and languages vs. gesture variation in the choice of the form-meaning mappings and (2) there are linguistic factors that influence the choice of mappings (e.g. encoding the noun-verb distinction).

However, all these studies have been based on a very small number of sign languages, they only focused on one aspect of iconicity (the choice between the object vs. handling handshapes), and only in one semantic field (namely, instruments). Another approach is represetned by the ASL-

[1] The model applies to both spoken and signed languages, but we focus on the latter in this paper.

Lex database (Caselli et al., 2016) which contains approximately 1000 ASL signs annotated (among other features) for iconicity ratings. However, the ratings only reflect a degree of iconicity (on a 7 point scale) for the whole signs, and does not discuss iconic features. Furthermore, it only contains data from one sign language. The same is true for the study in Cates et al. (2013) who annotated more than 700 ASL signs for iconicity of the three major parameters: hanshape, location, and movement. However, these parameters were only annotated as being iconic or non-iconic, without further analysis of iconicity.[2]

In order to study iconicity patterns in a more systematic manner, we created the database "Iconicity Patterns in Sign Languages" (IPSL).[3] It contains 1542 signs from seven semantic fields in nineteen sign languages and annotated them according to five iconic parameters. In addition, we created a website with several tools to visualize the iconicity patterns. In the rest of the paper we describe the creation of the database, the features of the website, and illustrate how they can be used in research on iconicity in sign languages.

2. Creating a Database of Iconicity Patterns

2.1. The Data

In order to create the IPSL database, we used the on-line dictionary of sign languages Spreadthesign (www.spreadthesign.com). This dictionary contains video recordings of isolated signs and signed sentences (up to 15 000 entries per language) in 31 sign languages. The dictionary has been created as a tool to facilitate learning of sign languages across the world; it has not been specifically designed for linguistic research. For instance, typically only one sign per concept is provided, while a sign language might have several signs. In addition, the procedure of data collection lead to the fact that lexical signs and multi-sign descriptions of non-lexicalized concepts are not systematically distinguished. However, since we focused only on the most basic concrete concepts, we consider the data to be good enough for our purposes.

We selected nineteen sign languages from all languages present at the website, namely Russian, French, American, British, Spanish, Italian, German, Polish, Brazilian, Turkish, Portuguese, Czech, Lithuanian, Swedish, Greek, Romanian, Latvian, Estonian, and Icelandic Sign Languages. The choice was governed by two main considerations: (1) we only selected the languages for which a majority of the 15 000 signs were present on the website at the moment of selection (the fall of 2016); (2) we excluded Ukranian and Belorussian Sign Language due to their close relatedness to Russian Sign Language.

We included 87 concrete concepts from seven semantic fields: transport, nature, instruments (tools), house, clothes, food, and animals. We reasoned that this selection represents a reasonable sample of the basic concrete concepts.[4]

It turned out that some sign languages are missing signs for some of the concepts. Furthermore, we decided to exclude any entries where a concept was described by a sequence of more than two signs, as they are unlikely to be lexicalized. This resulted in the database of 1542 annotated signs (out of the 1653 theoretically possible items).

2.2. Basic annotations

The annotation is based on Taub (2012) with some important modificaitons. First, we annotate each sign with respect to the form-image association (Taub, 2012). This specifically refers to the role that the handshape plays in depicting the object. We distinguish four types of form-image association:

1. Handling: the hand of the signer represents the hand of an agent that holds or handles the object, as in the RSL sign HAMMER;

2. Object: the hand depicts the shape of the object itself, as in the RSL sign CHAIR;

3. Contour: the hand represents the outline or surface of an object, as in the RSL sign HOUSE;

4. Tracing: the hands move to trace an outline or surface of an object, as in the RSL sign MOUNTAIN;

Following an insight from Taub (2012), we introduced a binary iconic feature of personification. It receives a positive value if the whole body (the hands, arms, upper body, and head) are a part of the iconic representation. A sign with a contour or tracing associations cannot involve personification by definition, because the hands do not represent the hands of a person. Signs with handling handshapes necessarily involve personification, but a sign with an object handshape can also involve personification, for instance, in the case of the hands representing paws or wings of an animal, as in the Polish Sign Language sign BIRD.

Notice that the form-meaning association feature concerns the role of the handshape. However, locations and movements in the signs can also be iconic. We thus also introduced a binary iconic feature of location. It receives a positive value if the location of the sign is iconic, that is, it represents the marked location of the object that is being represented. This happens with signs like SUN, MOON, SKY (see the RSL sign SUN) which are often located above the neutral space, or with pieces of clothing which are often located on the body (see the RSL sign T-SHIRT).

We also introduced a binary iconic feature of associated action. It receives a positive value if the movement in the sign depicts an action (independent movement or handling) associated with the object, as in the RSL sign KEY.

Finally, we decided to annotate the signs according to one of the possible concept-image associations (Taub, 2012), namely we introduced the part-whole feature, which receives a positive value if the sign represents not the whole

[2]The list of references on iconicity here is far from exhaustive, however, to our knowledge, no research comparable to the current project has ever been conducted.

[3]Note that this abbreviation is also sometimes used for Indo-Pakistani Sign Language.

[4]We did not include abstract concepts because iconic signs expressing abstract concepts contain an additional level of complexity, namely metaphorical mapping (Taub, 2001). This can be a topic of future research and extension of the database.

object, but only a noticeable part of the object, as in the RSL sign CAT.[5]

The iconicity features interact with each other in intricate ways. For instance, as discussed above, the positive value of personification is only possible for object and handling handshapes. We discuss the interactions between the features and the rules that we postulated to make the annotations consistent in detail on the About tab of our website.

The iconicity features that we selected are probably not exhaustive. However, they represent the iconic properties of all the major parameters of a sign (handshape, movement, and location). The database can be further extended with any novel features deemed necessary by future researchers.

Not all signs in the IPSL database are iconic. We distinguish two types of non-iconic signs: those involving fingerspelling, and those which do not involve fingerspelling and for which we cannot find any iconic motivation. Note that this means that the signs are in fact not transparent, but not necessarily non-iconic. We also tried to be careful in assigning iconic motivation to dubious cases, which means that in cases of doubt we would use the non-iconic label instead of trying to come up with a far-fetched interpretation.

2.3. Compounds and two-handed signs

Some signs have a complex morphological (or at least phonological) structure. We distinguish two cases that are annotated following additional rules, namely compounds and asymmetric two-handed signs.

Compounds are signs that have two clearly distinguishable sequential parts. For compounds, we annotate each part separately, using the "&" sign to separate the values of features: e.g. *handling&object* for a compound with the handling association in the first part, and the object association in the second part, as in the Italian sign SPOON.

Another case of complex signs are two-handed signs. In symmetric two-handed signs, the two hands have the same handshape and movement, so the hands are also necessarily the same with respect to the iconicity features, and we do not annotate them separately. However, in asymmetric two-handed signs the two hands might (and often do) depict separate parts or aspects of the depicted object. For instance, in the RSL sign HELICOPTER, one hand represents the cabin and the other hand the turning blades. Since these two-parts can be different with respect to iconicity features, we annotate them separately, using the "+" sign between the parts. The active hand is always annotated first: e.g. *handling+object*, see for instance the Turkish sign SPOON.

Sometimes one or both parts of a compound are asymmetric two-handed signs. This represents the most complex type of cases in our dataset, and it is annotated following the rules described above. For instance, we annotate the form-image mapping of the sign Icelandic sign LAMP as *tracing+contour&object+contour*.

[5]Taub (2012) also considers associated action to be a type of concept-image association. We are agnostic with respect to the exact nature of this feature, but still consider it a useful feature to systematically describe iconicity.

2.4. Reliability of Annotations

In order to ascertain the reliability of the annotation procedure, the whole data set (1542 sign tokens) has been fully annotated by Authors 1 and 2 independently. We then compared the annotations to calculate agreement per each of the features that we annotated.

In order to adjust the estimation for chance agreement, we calculated Cohen's kappa statistics using the *fmsb* package (Nakazawa, 2017) in R (R Development Core Team, 2016). We considered compounds and asymmetric two-handed signs as containing multiple observations, so we separated them into separate cells (e.g. *handling+object* was turned into two observations: *handling* and *object*, located in separate cells in the dataset).

The resulting estimated values of Cohen's kappa, with confidence intervals and qualitative characterizations (as provided by the *fmsb* package) for iconic features, are summarized in Table 1.

	Cohen's kappa	CI
Form-image	0.87	0.85-0.89
Personification	0.78	0.75-0.81
Associated action	0.8	0.78-0.83
Location	0.76	0.73-0.78
Part-whole	0.65	0.61-0.69

Table 1: Annotators' agreement per iconic feature

From the table it should be clear that we have high agreement for all features, especially for the form-image mapping. The lowest agreement is observed for the part-whole feature. This is explained by the fact that the authors originally interpreted this feature in a different way for three classes of signs. Firstly, the signs for animals where the hands represent paws/wings of an animal should be analyzed as [-part-whole], as the whole body represents the body of the animal. One of the authors however consistently glossed these signs as [+part-whole]. The other two classes are the asymmetrical signs discussed in the previous section, in which the role of the second hand is unclear with respect to this feature.

After the quantitative analysis of agreement, the two authors discussed the cases of disagreement and agreed upon a correct annotation for each of those cases. Therefore, the final dataset can be considered reliably annotated following the guidelines described in this paper and on the website.

3. The IPSL Website

To make the database easily accessible to other researchers interested in iconicity, we created the IPSL website: https://sl-iconicity.shinyapps.io/iconicity_patterns/, built with *shiny* (Chang et al., 2017). The website contains a detailed description of the creation of the IPSL database, the full database (which can be searched on-line with *dt* (Xie, 2016) or downloaded), and visualistion tools.

The first tool visualizes the concepts on the map of the world. The user can select one of the 84 concepts, which are repseresented as dots on the map with color-coding of the form-image mapping. Each dot on the map is clickable and opens the video of the relevant sign (see Figure 1). The user can also filter the data points by specifying the values

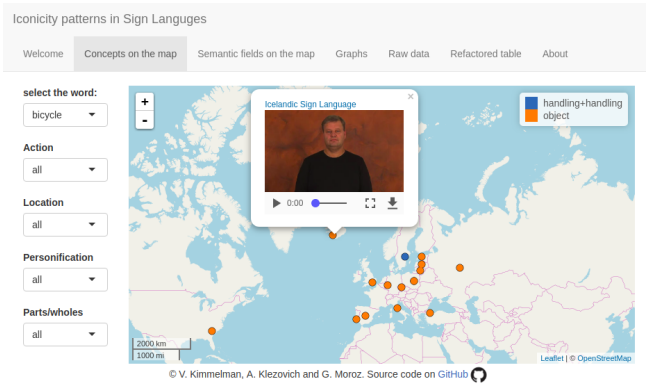


Figure 1: Concepts on the map: the sign BICYCLE.

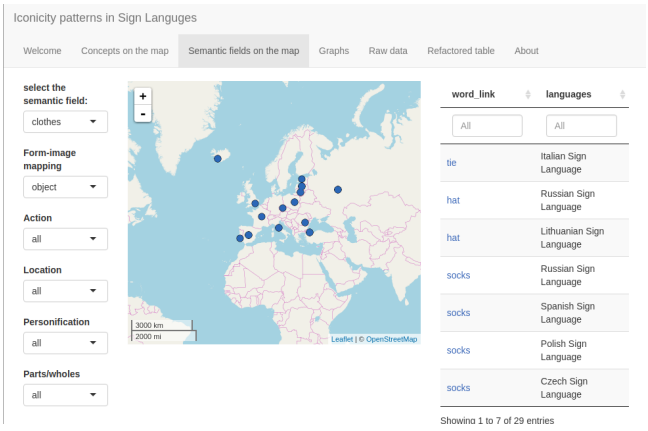


Figure 2: Semantic fields on the map: clothes.

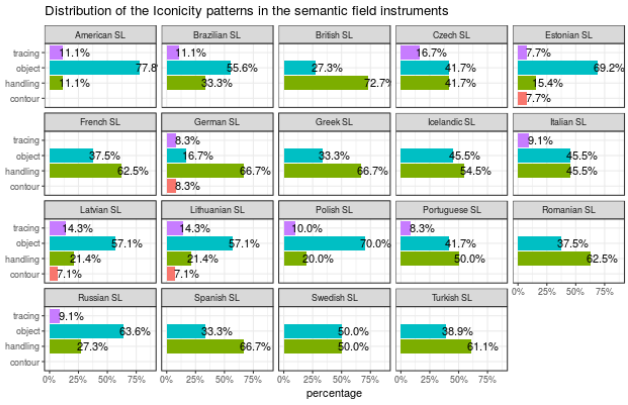


Figure 3: Form-image mappings for instruments.

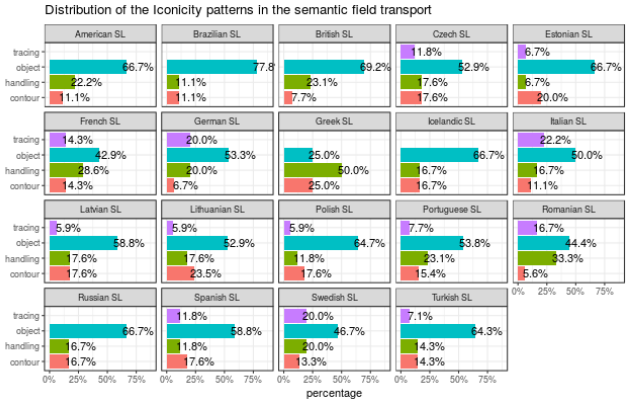


Figure 4: Form-image mappings for transport.

of the associated action, localization, personification, and part-whole features. The maps are created with the *lingtypology* package (Moroz, 2017).

The second tool visualizes the semantic fields on the map. The user can select one of the seven semantic fields, and then specify the form-image mapping and the values of the associated action, localization, personification, and part-whole features; the signs which conform the selection appear on the map (see Figure 2). In addition, a table is generated containing the concepts and languages with the relevant feature combinations.

Finally, the third tool creates graphs (bar charts) built with *ggplot2* (Wickham and Chang, 2016). The user can select a semantic field (or all semantic fields), and the charts show the distribution of concepts by the form-image mapping feature for each language, either in absolute values or in percentages (see Figure 3).

4. Linguistic Analysis

The database and the website can be used for various types of linguistic analyses of iconicity. The first conclusion that can be drawn from data is that iconicity patterns are influenced by both linguistic and semantic differences.

For instance, we can replicate the findings from Padden et al. (2013) and Brentari et al. (2015) for the form-image mappings in the domain of instruments. As Figure 3 shows, sign languages indeed vary with respect to this feature: some (e.g. ASL and RSL) prefer the object mapping, others (e.g. British and Romanian Sign Languages) prefer the handling mapping, while some (e.g. Italian Sign Language) do not show a strong preference.

Another finding not previously studied in detail is that different semantic fields have different preferences. For instance, while in the field of instruments the object and handling mappings are prevalent, for transport all but one sign language use predominantly the object mapping (Figure 4). Other iconic features also show some cross-linguistic variation and semantic preferences. For instance, Romanian Sign Language seems to use iconic localization relatively less frequently than other sign languages, while ASL has the highest proportion of signs that uses parts of the object to represent the object. Semantically, iconic locations are used predominantly in nature-related and clothing-related concepts, while personification, in addition to being used in all signs with the handling mapping, is commonly used in signs for animals.

5. Future Use of the IPSL Database

There is a variety of ways in which the IPSL database can be used for further research.

We are currently investigating a number of questions based on the database (Kimmelman et al., in preparation). Firstly, we are ivestigating whether the language and the semantic fields are indeed significant predictors of all iconicity features, and whether these predictors interact with each other. Secondly, we are studying the interactions of the iconicity features in order to find patterns and frequent constellations of features which might reflect some common iconic strate-

gies. Finally, we are comparing the sign languages to each other with repsect to iconicity to see whether such a measurement would reflect historical and geographical connections between them.

The database might be further extended. It is clear that the list of iconic features is not exhaustive: for instance, we have not considered non-manual expressions. The analysis of iconicity in terms of handshape, location, and movement might be too rough: maybe these parameters should be further subdivided and analysed for iconicity in more detail.

In addition, the database can be extended by adding more signs: extra signs for the existing semantic fields, concrete signs from other semantic fields, and abstract signs. It is clear that 87 concepts are not enough to generalize our findings to the whole lexicon.

Another possible extension would be an additon of signs for abstract concepts. As demonstrated by (Taub, 2001), abstract concepts are expressed in sign languages primarily through metaphoric-iconic signs. This means that an abstract concept is associated via metaphoric mapping with a concrete image, and this image is then enocded iconically. For instance, a sign for FEAR might be based on a metaphorical mapping between fear and the heart beating fast, and the concrete event of the heart beating fast is then encoded iconically (as e.g. in Austrian Sign Language: FEAR). If such signs are included in the database, we can annotate the iconic features for the concrete concept the usual way, but novel features describing the nature of metaphoric mapping should be added.

It would also be interesting to investigate the relationship between iconic features identified here and lexical features of these signs. This can be done by annotating further lexical information for the signs in the databse. For instance, these lexemes can be analyzed in terms of Pustejovsky's qualia structure (Pustejovsky, 1995), to find out whether certain iconic features of lexemes correspond to certain sub-features of their qualia structures.

To sum up, in this paper, we reported the process of creation of the IPSL database of iconic features in 1542 signs from nineteen sign languages and of the database website containing several visualization tools. We also demonstrated how the database can be use to replicate and extend previous typological approaches to iconicity in sign languages, and formulated a number of research questions that can be further investigated with the database.

6. Acknowledgements

The publication was prepared within the framework of the Academic Fund Program at the National Research University Higher School of Economics (HSE) in 2018 (grant Nr. 18-05-0053) and by the Russian Academic Excellence Project "5-100". Vadim Kimmelman's research has been supported by the the Dutch Science Foundation (NWO) grant Nr. 360-70-520.

7. Bibliographical References

Brentari, D., Renzo, A. D., Keane, J., and Volterra, V. (2015). Cognitive, Cultural, and Linguistic Sources of a Handshape Distinction Expressing Agentivity. *Topics in Cognitive Science*, 7(1):95–123, January.

Caselli, N. K., Sevcikova Sehyr, Z., Cohen-Goldberg, A. M., and Emmorey, K. (2016). ASL-Lex: A lexical database for ASL. *Behavior Research Methods*, 49(2):784–801.

Cates, D., Gutiérrez, E., Hafer, S., Barett, R., and Corina, D. (2013). Location, location, location. *Sign Language Studies*, 7(1):433–461.

Chang, W., Cheng, J., Allaire, J., Xie, Y., and McPherson, J., (2017). *shiny: Web Application Framework for R*.

Kimmelman, V., Klezovich, A., and Moroz, G. (in preparation). Iconicity Patterns in Sign Languages. Manuscript, Moscow-Amsterdam.

Moroz, G., (2017). *lingtypology: easy mapping for Linguistic Typology*.

Nakazawa, M., (2017). *fmsb: Functions for Medical Statistics Book with some Demographic Data*.

Padden, C., Meir, I., Hwang, S.-O., Lepic, R., Seegers, S., and Sampson, T. (2013). Patterned iconicity in sign language lexicons. *Gesture*, 13(3):287–308.

Padden, C., Hwang, S.-O., Lepic, R., and Seegers, S. (2015). Tools for Language: Patterned Iconicity in Sign Language Nouns and Verbs. *Topics in Cognitive Science*, 7(1):81–94.

Perniss, P., Thompson, R. L., and Vigliocco, G. (2010). Iconicity as a General Property of Language: Evidence from Spoken and Signed Languages. *Frontiers in Psychology*, 1.

Pustejovsky, J. (1995). *The Generative Lexicon*. MIT press, Cambridge.

R Development Core Team, (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Taub, S. F. (2001). *Language from the Body: Iconicity and Metaphor in American Sign Language*. Cambridge University Press, Cambridge.

Taub, S. F. (2012). Iconicity and metaphor. In Roland Pfau, et al., editors, *Sign Language: An International Handbook*, pages 388–412. De Gruyter Mouton, Berlin.

Wickham, H. and Chang, W., (2016). *ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*.

Xie, Y., (2016). *DT: A Wrapper of the JavaScript Library 'DataTables'*.

Sign Languages and the Online World
Online Dictionaries & Lexicostatistics

Shi Yu[1,2,3], **Carlo Geraci**[2,3], **Natasha Abner**[4]
[1]École des Hautes Études en Sciences Sociales, [2]Institut Jean Nicod CNRS, [3]École Normal Supérieure,
29, Rue d'Ulm, 75005 Paris, France
[4]Department of Linguistics, University of Michigan
450 Lorch Hall 611 Tappan Street Ann Arbor, MI 48109-1220, USA
shi@shiyu.fr, carlo.geraci76@gmail.com, nabner@gmail.com

Abstract

Several online dictionaries documenting the lexicon of a variety of sign languages (SLs) are now available. These are rich resources for comparative studies, but there are methodological issues that must be addressed regarding how these resources are used for research purposes. We created a web-based tool for annotating the articulatory features of signs (handshape, location, movement and orientation). Videos from online dictionaries may be embedded in the tool, providing a mechanism for large-scale theoretically-informed sign language annotation. Annotations are saved in a spreadsheet format ready for quantitative and qualitative analyses. Here, we provide proof of concept for the utility of this tool in linguistic analysis. We used the SL adaptation of the Swadesh list (Woodward, 2000) and applied lexicostatistic and phylogenetic methods to a sample of 23 SLs coded using the web-based tool; supplementary historic information was gathered from the Ethnologue of World Languages and other online sources. We report results from the comparison of all articulatory features for four Asian SLs (Chinese, Hong Kong, Taiwanese and Japanese SLs) and from the comparison of handshapes on the entire 23 language sample. Handshape analysis of the entire sample clusters all Asian SLs together, separated from the European, American, and Brazilian SLs in the sample, as historically expected. Within the Asian SL cluster, analyses also show, for example, marginal relatedness between Chinese and Hong Kong SLs.

Keywords: Web-based annotation tool, Sign Language, Lexicostatistics, Phylogenesis, Online dictionaries

1. Introduction

Lexicostatistics provides a means of determining the degree of similarity across languages by simply looking at portions of their vocabulary (Swadesh, 1971). Though such studies have been largely limited to spoken languages, promising results have been documented once similar methodologies are applied to sign languages (Woodward, 2000; McKee and Kennedy, 2000). In particular, Woodward (2000) proposes a sign language (SL) adaptation of the Swadesh list. Like the original Swadesh list, Woodward's list contains 100 items that are meant to identify basic/universal concepts which are supposed to reveal the degree to which pairs of SLs are related. However, this method has not been systematically tested or applied to SLs. A key reason for this is a lack of reliable data and the absence of software applications that allow for easy annotation of video data. In recent years, however, many SL dictionaries have appeared on the internet and can be freely consulted, solving the empirical problem of gathering the relevant data. The existing applications (ELAN, Ilex, SignStream, etc.) for annotating video data are stand-alone applications designed primarily to work with files stored on local machines. Moreover, these applications are designed with research flexibility in mind and do not come "pre-equipped" with theoretically-informed annotating codes or coding categories.

We have created a web-based tool that addresses these outstanding issues. The web-based application imports videos of signs from online dictionaries and provides a theoretically-informed annotation schema for the main articulatory properties of these signs. We show here how this tool facilitates theoretically-informed typological and historical analysis of sign languages, using the interface to systematically investigate the degree of similarities across 23 SLs. Thus, our methodology implements Woordward's original idea of comparing pairs of sign languages in such a way as to conduct an effective cross-linguistic comparison of a large sample of SL. The video data used for the present analyses come mainly from the online dictionary of the Spread The Sign Project (Domfors and Fredäng, 2008) and also from LSD Visual Sign Language Dictionary (Hong Kong Sign Language: `http://www.sign-aip.net/sign-aip/en/home/index.php`), Taiwan Sign Language Online Dictionary (Tsay et al. (2008): `http://lngproc.ccu.edu.tw/TSL/indexEN.html`) and NHK Sign Language CG (Japanese Sign Language: `https://www2.nhk.or.jp/signlanguage/index.cgi`). We present a case study of four Asian SLs (Japanese SL: *JSL*, Chinese SL: *CSL*, Taiwanese SL: *TSL*, and Hong Kong SL: *HKSL*) and explore relations within this historically and areally related group. We also apply this analytic approach to the handshapes of the 23 SLs in our database and use a cluster analysis to identify relationships across the sample.

The rest of the paper is organized as follows: Section 2 introduces some basic principles of lexicostatistics applied to SL, Section 3 describes the web application we created for phonological annotation of signs, Section 4 describes the comparative approach applied to the data, Section 5 reports the results of a case study on four Asian SLs, and Section 6 provides the analysis of handshapes of the sample of 23 SLs. Finally, Section 7 concludes the paper.

2. Lexicostatistics & Sign Language

Lexicostatistics is a method used in historical and comparative linguistics to determine the relationship between pairs of languages based on the degree of shared lexicon (Dobson, 1969; Rea, 1990). List(s) of concepts/meanings which are assumed to be universally instantiated in the world's languages (e.g., blood, many, leaf, etc.) are used to compare the lexica (Swadesh, 1971). Large scale comparison may then be made by means of distance matrices and cluster analysis. Although the lists and the methodology have been criticized (Hoijer, 1956; Gudschinsky, 1956), lexicostatistics has proven to be a good method to work with underdescribed and unwritten languages (Crowley and Bowern, 2010; Lehmann, 2013). The percentage of overlapping properties across items from the list determines the linguistic distance between two sign languages. For spoken languages, languages that share more than 81% of signs are treated as dialects of the same language; if the percentage is between 36% and 81%, they are treated as different languages from the same family; while if the percentage is below 36%, the two languages then belong to distinct families (Crowley and Bowern, 2010).

Woodward (1993) adapted the original Swadesh list for the purpose of sign language comparison (Figure 1). In particular, he removed body parts and pronouns because they are often represented in SLs by pointing to the referent; thus, they may lead to an overestimation of the relationship between SLs. In his works Woodward compared pairs of languages like American and French SL (Woodward, 1978) and several South Asian and East Asian sign languages (Woodward, 1993). McKee and Kennedy (2000) used Woodward's list to compare British (BSL), Australian (Auslan) and New Zealand SLs (all closely historically related) to the historically unrelated American SL. For each item in the list, pairs of languages may be evaluated on the similarity of the articulatory properties used in the languages' signs for that item. The articulatory properties themselves may be drawn from the four major phonemic classes of SLs: handshape, location, movement, and palm orientation. Such a comparison produces results like those shown in Table 1 for Auslan and BSL (adapted from McKee and Kennedy (2000)).

Auslan &BSL	Hs	Loc	Mov	Ori	Notes
egg			x		
grass			x		Different weak hand
look for	x				Two handed in BSL

Table 1: Example of lexical comparison

These approaches are based on pairwise comparisons of SLs and they show that the lexicostatistics method can be successfully applied to languages in the visual modality (but see Section 4 for commentary on some of the problems of this method of comparison). However, previous research has not attempted a systematic comparison of a large sample of SLs.

3. An Annotation Tool For Online Dictionaries

In this section we describe the front-end of the web-based tool that we created for annotation videos from online dictionaries. The annotation tool has been created using JavaScript, and a JavaScript plugin (Video.js) was used to display the video files fetched from online SL dictionaries, the video is displayed continuously with repetition. The workspace is accessible by standard web browsers and is divided in three major areas: 1) on the top-left side, the video-streaming for annotation, 2) in the central part, the main annotation area, and 3) one the right, the list of words to be annotated (Figure 2). Languages are chosen by using a dropdown menu on top of the list. The results of annotations for a specific sign are summarized below the video.

The data set is first imported by using the English version of Woodword's list, a script is used to fetch the corresponding words and videos in other languages on the Spread The Sign online dictionary, where the same word is grouped together across languages. Our data set thus include the word in English and the corresponding word in the original language. All words are checked during the annotation whenever ambiguity arises (e.g., two entries for the word "dust", as noun and as verb)

We included 55 handshapes in our annotation tool. These 55 handshapes are supposed to be representative of handshapes used in sign languages and have been proven to be able to capture most handshape configuration in our data set. Several categories of handshape include multiple handshape images that are allophonic variations in SLs. For annotation, this step requires only a click on the correspondant handshape. Also in this section, the hand part feature (i.e., Orientation (Brentari, 1998)) can be selected using the dropdown list, the two-handed option is used to annotate signs with identical articulation of both hands.

The second section contains features of place of articulation, based on Brentari (1998) model, we included neutral space and four major regions. For signs produced in neutral space, the choice is between horizontal, vertical, or lateral. For signs produced on a major body region (*head*, *torso*, *arm*, *hand*), the annotator may use the dropdown list to specify one among eight micro regions each.

- Head: *top*, *forehead*, *eye*, *cheek*, *nose*, *lip*, *mouth*, *chin*, and *below-chin*
- Arm: *upper*, *elbow-front*, *elbow-back*, *forearm-front*, *forearm-back*, *forearm-ulnar*, *wrist-front*, and *wrist-back*
- Hand: *palm*, *finger-fronts*, *back of palm*, *back of fingers*, *radial-side*, *ulnar-side*, *tip*, and *heel*
- Torso: *neck*, *shoulder*, *clavicle*, *torso-top*, *torso-mid*, *torso-bottom*, *waist*, and *hips*

1. all	26. grass	51. other	76. warm
2. animal	27. green	52. person	77. water
3. bad	28. heavy	53. play	78. wet
4. because	29. how	54. rain	79. what
5. bird	30. hunt	55. red	80. when
6. black	31. husband	56. right	81. where
7. blood	32. ice	57. river	82. white
8. child	33. if	58. rope	83. who
9. count	34. kill	59. salt	84. wide
10. day	35. laugh	60. sea	85. wife
11. die	36. leaf	61. sharp	86. wind
12. dirty	37. lie	62. short	87. with
13. dog	38. live	63. sing	88. woman
14. dry	39. long	64. sit	89. wood
15. dull	40. louse	65. smooth	90. worm
16. dust	41. man	66. snake	91. year
17. earth	42. meat	67. snow	92. yellow
18. egg	43. mother	69. star	93. full
19. fat	44. mountain	70. stone	94. moon
20. father	45. name	68. stand	95. brother
21. feather	46. narrow	71. sun	96. cat
22. fire	47. new	72. tail	97. dance
23. fish	48. night	73. thin	98. pig
24. flower	49. not	74. tree	99. sister
25. good	50. old	75. vomit	100. work

Figure 1: Woodward's vocabulary list for sign language comparison.

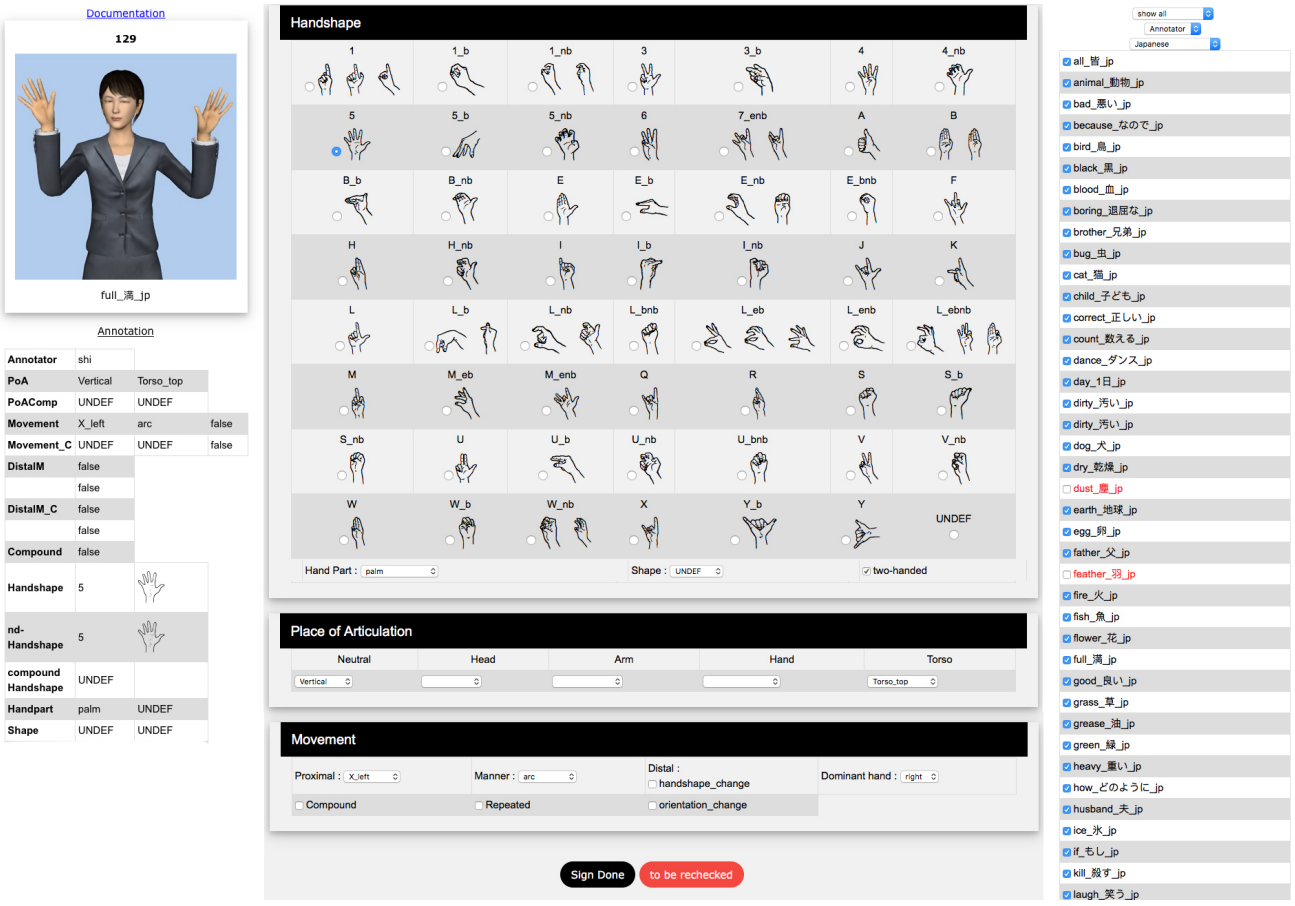


Figure 2: Workspace of the web-based annotation tool.

For movement in the third section, both dropdown list and check button are used for annotation. For proximal movements, we annotate the axis on which the movement is performed and its direction (forward, backward, down, up, left, and right). For distal movements the non-exclusive options are handshape change and orientation change. We also annotated the manner of movement (straight, circular, arch, etc.) and presence/absence of repetitions.

Each of the previous sections are duplicated in the case of compound sign, when the option "Compound" is selected, additional sections will display on the screen for the annotation of the second part of the sign. For all the options of annotation, we reserved an "undefined" option for empty value and also for the review of ambiguous signs for annotation. Annotation results are sent to the server and saved in the JSON format. This file is then transformed to a *.csv* file for the purpose of linguistic analysis.

4. Methodology

Previous studies compared signs by looking at the global similarities of the four main classes of phonemes (Handshape, Location, Movement and Orientation). However, none of them has been explicit on how similarity is measured. In particular, for each class of phonemes it was never specified the set of contrastive features that would determine a significant difference between any two phonemes. For instance, consider the following handshapes:



They all have four selected fingers, some of them also have a selected thumb. Some of them have spread or stacked fingers, some have flexed non-base joints, others have flexed base joints. Under a holistic analysis all these handshapes could be considered similar. However, a feature-based analysis would distinguish the handshapes not just on the number of selected fingers, but also based on thumb selection, whether the selected fingers are spread or not, the base and/or non-base joints are flexed, etc.

Similar considerations extend to the other classes of phonemes. For instance, it is unclear whether the neutral space was treated as a single entity or whether different planes have been distinguished (horizontal, vertical, lateral). Even more problematic is the case of orientation where the definition itself may lead to different interpretations of what counts as similar/identical. Indeed, orientation can be defined either in absolute terms with respect to signer's body or relative to the plane of articulation (Quer et al., 2017).

In our study, we decided to use a theoretically-informed annotation procedure and implement a feature based analysis directly in the annotation tool (see Section 3). Rather then establishing identity/similarity based on the global assessment of pairs of (video) signs, we used Brentari (1998) model to generate the set of features upon which difference is then measured. The signs of each language are independently annotated by selecting the relevant feature values. Pairwise comparison is made post-hoc by counting the number of identically specified features. Pairs of signs sharing all features are considered identical. Pairs of signs where only one feature value is different feature are treated as similar. Pairs of signs that are different for more than one feature are treated as different. This procedure of assessing the articulatory properties of signs is in many respects stricter than those used in previous studies and it has the risk of biasing the data by maximizing differences. It also treats as equally relevant features that generates macroscopic differences (like selected fingers) and features that creates less perceivable differences (like flexed non-base joints). However, these biases can be mitigated by neutralizing some differences (e.g., collapsing [$\pm$ spread] handshapes in one single group, grouping locations by major regions, etc.) or by weighting features. In this study, we decided to consider all features and not to apply any weight correction. However, we show the effect of collapsing some feature values for handshape and place of articulation.

In previous studies, Annotators' subjective perception could affect data evaluation in two steps of the procedure. First when s/he tries to identify the individual phonemes for each sign, and then when s/he has to establish whether pairs of phonemes are identical, similar or different. Our procedure is based on the annotation of the articulatory properties of individual signs. It does not mitigate the subjective evaluation occurring when identifying the correct phoneme, but it removes any subjectivity from the evaluation of similarities between two signs.

5. Comparing Asian Sign Languages

We applied our annotation procedure to investigate potential relations between pairs of languages. Our data set is annotated by one sign language expert to keep the homogeneity and the correctness of the annotation. The kinds of comparisons and analyses reported here are similar to those reported in previous studies (a.o., Woodward (2000) and McKee and Kennedy (2000)). However, the fact that we adopted an extremely rich set of features allows us to perform a more effective comparison of the articulatory properties of the signs.

As a case study, we conducted an analysis on four Asian SLs: JSL, CSL, TSL and HKSL. Unfortunately, very little is known about historical relations among these SLs. We cross-checked information available to us such as the Ethnologue of World's Languages and Wikipedia and we found that JSL is related to TSL (and Korean SL), while CSL (variety of Shanghai) is related to although not mutually intelligible with HKSL.

In the following tables we provide the results. On the first column we indicate the pairs of languages; on the second column we report the percentage of signs that are identical in the two languages; on the third column we report signs that are similar (i.e. only one feature/phoneme is different); while the last column reports the percentage of signs that are different (i.e. two or more features/phonemes are different).

Table 2 reports the results of the comparison made with the full set of phonological features; Table 3 reports the results after handshapes with the same selected fingers but different joint flection have been collapsed into one

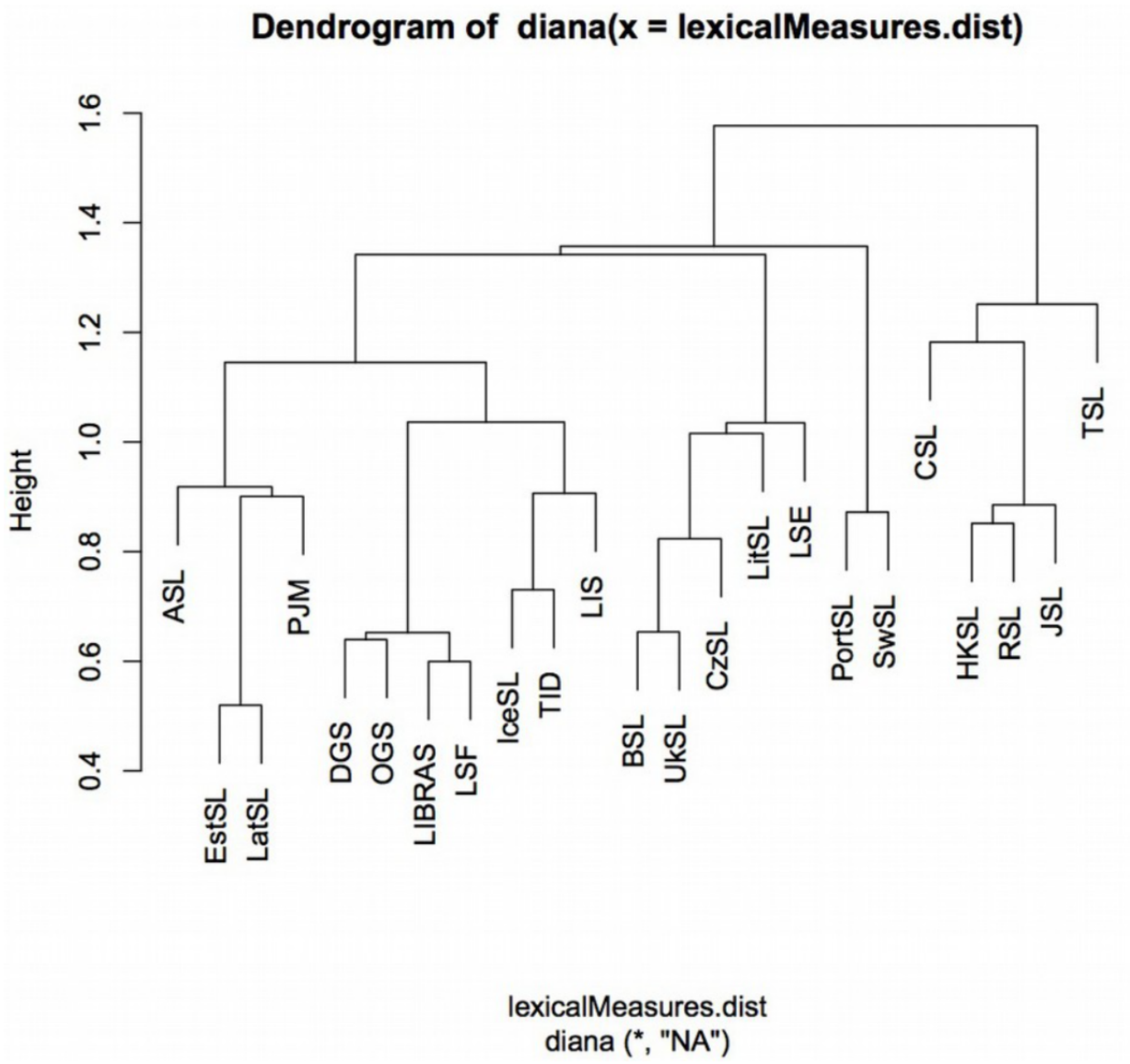


Figure 3: Divisive hierarchical cluster analysis of 23 SLs based on handshapes. ASL: *American SL*, BSL: *British SL*, CSL: *Chinese SL*, CzSL: *Czech SL*, EstSL: *Estonian SL*, LSF: *French SL*, DGS: *German SL*, OGS: *Austrian SL*, HKSL: *HongKong SL*, IceSL: *Icelandic SL*, LIS: *Italian SL*, JSL: *Japanese SL*, LatSL: *Latvian SL*, LitSL: *Lithuanian SL*, PJM: *Polish SL*, PortSL: *Portuguese SL*, LIBRAS: *Brazilian SL*, RSL: *Russian SL*, LSE: *Spanish SL*, SwSL: *Swedish SL*, ASL: *Taiwan SL*, TID: *Turkish SL*, UkSL: *Ukrainian SL*.

level; Table 4 reports results after place of articulation has been collapsed into five major regions (neutral space, head, torso, arm, hand), while table 5 reports results after handshape and place of articulation have been collapsed.

Languages	Identical	Similar	Different
JSL&CLS	0.00%	9.28%	90.72%
JSL&TSL	4.30%	13.98%	81.72%
JSL&HKSL	3.09%	9.28%	87.62%
CSL&TSL	4.26%	13.83%	81.91%
CSL&HKSL	9%	16%	74%
TSL&HKSL	3.19%	12.77%	84.04%

Table 2: Comparison made with the full set of features

Languages	Identical	Similar	Different
JSL&CLS	1.03%	11.34%	87.63%
JSL&TSL	5.37%	15.05%	79.57%
JSL&HKSL	4.12%	11.34%	84.84%
CSL&TSL	5.32%	17.02%	77.66%
CSL&HKSL	10.10%	22.22%	67.68%
TSL&HKSL	4.26%	14.89%	80.85%

Table 3: After collapsing handshapes with the same selected fingers

What emerges by looking at the percentages of the different tables is that a comparison based on pure articulatory features does not let emerge any cross-linguistic similarity. However, when the effect of some features is neutralized,

Languages	Identical	Similar	Different
JSL&CLS	2.06%	10.31%	87.63%
JSL&TSL	5.38%	19.35%	75.27%
JSL&HKSL	4.12%	12.37%	83.51%
CSL&TSL	4.26%	18.09%	77.66%
CSL&HKSL	14.14%	23.23%	62.63%
TSL&HKSL	6.38%	12.77%	80.85%

Table 4: After collapsing locations in major areas

Languages	Identical	Similar	Different
JSL&CLS	3.09%	13.40%	83.51%
JSL&TSL	6.45%	23.66%	69.89%
JSL&HKSL	5.15%	15.46%	79.38%
CSL&TSL	5.32%	21.28%	73.40%
CSL&HKSL	15.15%	24.24%	60.60%
TSL&HKSL	7.45%	14.89%	77.66%

Table 5: After collapsing locations and handshape

some similarities emerge. In particular, Table 5 shows that CSL and HKSL share around 40% of the signs in the Woodward list and should be treated as two different languages of the same family. JSL and TSL share almost 30%. While this is not enough to consider them as languages of the same family, it somehow makes justice of the fact that the two are not mutually intelligible.
Traditional lexicostatistics methodologies leave open the question whether at a higher level detailed analysis these languages belong to the same linguistic group or not. We address this question in the next section.

6. On Handshape Features

In this section we report the analysis of handshape similarities conducted on the 23 SLs available in our annotated data set (see Figure 3)
We conducted a divisive cluster analysis (Baayen, 2008). What emerges somewhat clearly is that the Asian languages (plus RSL) are clustered together, while all European languages plus American and Brazilian SL are split in secondary clusters. Based only on handshape, we can readily distinguish two large sign language families.

7. Conclusions

Documentation of individual SL history is quite fragmented and often unreliable, especially when it comes to describe contact with other SLs. In this paper we used lexicostatistics and phylogenetic methods to investigate the degree of similarity across 23 SLs. This has been made possible thanks to the use of online resources and a new web-base annotation tool that we created specifically for this purpose. Results showed that lexicostatistics methods are reliable as long as the degree of analysis remains at a superficial level. The variability and degree of freedom introduced by more fine-grained annotations of the articulatory properties of signs make methods based on holistic assessment of similarity less reliable. However, once more sophisticated analysis are used, cross-linguistic similarities emerge even once looking at a relatively large sample of languages.

8. Acknowledgements

This research has been funded by Fyssen Fundation Research Grant (PI: Carlo Geraci).

9. Bibliographical References

Baayen, R. H. (2008). *Analyzing linguistic data: A practical introduction to statistics using R*. Cambridge University Press.

Brentari, D. (1998). *A prosodic model of sign language phonology*. MIT Press.

Crowley, T. and Bowern, C. (2010). *An introduction to historical linguistics*. Oxford University Press.

Dobson, A. J. (1969). Lexicostatistical grouping. *Anthropological Linguistics*, pages 216–221.

Domfors, L.-Å. and Fredäng, P. (2008). Spread the sign. `http://www.spreadthesign.com/`.

Gudschinsky, S. C. (1956). The ABC's of lexicostatistics (glottochronology). *Word*, 12(2):175–210.

Hoijer, H. (1956). Lexicostatistics: A critique. *Language*, 32(1):49–60.

Lehmann, W. P. (2013). *Historical linguistics: An introduction*. Routledge.

McKee, D. and Kennedy, G. (2000). Lexical comparison of signs from American, Australian, British and New Zealand sign languages. *The signs of language revisited: An anthology to honor Ursula Bellugi and Edward Klima*, pages 49–76.

Quer, J., Cecchetto, C., Donati, C., Kelepir, M., Pfau, R., and Steinbach, M. (2017). *The SignGram Blueprint: A Guide to the Preparation of Comprehensive Reference Grammars for Sign Languages*. De Gruyter Mouton, Berlin/Boston.

Rea, J. A. (1990). Lexicostatistics. *Research Guide on Language Change, Trends in Linguistics Studies and Monographs*, 48:217–222.

Swadesh, M. (1971). *The origin and diversification of language*. Transaction Publishers.

Tsay, J., Tai, J. H.-Y., Lee, H.-H., Chen, Y., and Liu, C.-H. (2008). Taiwan sign language online dictionary. 3rd edition. Institute of Linguistics, National Chung Cheng University, Taiwan. `http://lngproc.ccu.edu.tw/TSL/indexEN.html`.

Woodward, J. (1978). Historical bases of American Sign Language. In Patricia Siple, editor, *Understanding Language through Sign Language Research*. Academic Press.

Woodward, J. (1993). Lexical evidencefor the existence of South Asian and East Asian sign language families. *Journal of Asian Pacific Communication*, 4:91–106.

Woodward, J. (2000). Sign languages and sign language families in Thailand and Viet Nam. In Karen Emmorey et al., editors, *The signs of language revisited: An anthology to honor Ursula Bellugi and Edward Klima*. Lawrence Erlbaum Associates.

Elicitation protocol and material for a corpus of long prepared monologues in Sign Language

Michael Filhol, Mohamed Nassime Hadjadj
LIMSI, CNRS, Université Paris Saclay
Rue John Von Neumann, 91400 Orsay, France
{michael.filhol, hadjadj}@limsi.fr

Abstract

In this paper, we address collection of prepared Sign Language discourse, as opposed to spontaneous signing. Specifically, we aim at collecting long discourse, which creates problems explained in the paper. Being oral and visual languages, they cannot easily be produced while reading notes without distorting the data, and eliciting long discourse without influencing the production order is not trivial. For the moment, corpora contain either short productions, data distortion or disfluencies. We propose a protocol and two tasks with their elicitation material to allow cleaner long-discourse data, and evaluate the result of a recent test with LSF informants.

Keywords: Sign Language, elicitation, discourse

1. Purpose and motivation

Sign Language (SL) corpora have long been scarce resource, which impeded linguistic studies as access to observable language productions was limited until this millenium. At the same time the growth of SL recognition and spread of Sign linguistics increased the need for properly recorded data. Also then, light digital cameras together with video viewing, editing and annotation software became commonplace, and cheap. Consequently, a lot more data became available to SL research and several projects were even almost dedicated to produce reference corpora, in a variety of genres. We will be referring to them in this paper.

However, the nature of recorded Sign still makes it more difficult to collect, use, browse, share. It often weighs over 200 times more storage space than its equivalent in text, is not yet searchable in its primary form, and is almost impossible to anonymise, which makes any publicity subject to legal requirements. So while corpus studies have undoubtably been facilitated by a growing number of data collections in the past two decades, finding an appropriate corpus and gaining access to it if any exists is often still a problem when starting a SL study.

In our recent work, we have been studying camera-facing short news reports in French Sign Language (LSF), to model a neutral reference language genre without limiting possible topics. Now we would like to study longer discourse of the same genre to look at higher-level discourse/rhetoric operations.

Arguably, this is an idealised version of the language since it is not the way SL is used most of the time. Nevertheless it captures the well existing idea of canonical fluent discourse, i.e. constructed discourse with none of the following:

- interference from external channels, e.g. dialogue interruptions, reaction to feedback signals;
- disfluency, e.g. hesitations, backtracks, etc.

Avoiding interference justifies the use of camera-facing monologue, as opposed to conversation setups. Avoiding disfluency requires prepared discourse, as opposed to spontaneous. In other words the final production is intended to match a planned result, not built on the fly. Ensuring this allows to afford confidence that observed articulations are a product of intended—hence assumed correct—language usage, and not that of a reparation strategy or filler for a pause for thought.

2. Existing material and properties

Various corpus projects were conducted, of different genres, using different setups and produced with more or less specific purposes. They are now numerous enough to prevent exhaustivity (though not always easily accessible), but this section presents a few contrastive major examples, relevant for comparison with our objectives.

Many corpus projects have a dialog task setup. Dialogue is generally more ecological in the sense that it captures language in its most used and living form, but does not allow to model canonical discourse-level structures.

Often though, one party is (or plays the part of) a listener only, merely asking a few questions while avoiding to interfere with the discourse. The "BSL corpus"[1] (Schembri, 2008), the "corpus NGT"[2] (Crasborn and Zwitserlood, 2008) and some tasks of the DictaSign corpus (Matthes et al., 2010) fall in this category. This does open a window on longer monologues, but nevertheless keeps the dialogue feedback channel open, which inflicts on the dynamics of the production. And a bigger problem for us yet is the lack of discourse preparation, as those tasks are generally elicited narrations, organised and produced spontaneously.

A few resources contain prepared elicited discourse, like the story-telling tasks in LS-COLIN (Cuxac et al., 2002), the joke task in the DGS Korpus (Nishio et al., 2010). In LS-COLIN, informants were given the assignment and elicitation material in a separate room prior to standing in the studio, and could take the time to prepare their production. As far as the documents describe the collection process, no material was visible during the recording though. By

[1] BSL = British Sign Language
[2] NGT = Dutch Sign Language

contrast, the entire DictaSign corpus was collected with a helper screen on which the task elicitation material was available. It did not display informants' preparation notes (the corpus does not contain prepared discourse), but it is an example of corpus collection making use of inline visual support.

For our own work on (short) news reports, we created a corpus "40 brèves" that matched our requirements of prepared camera-facing monologue, inspired by the WebSourd® internet site in its time[3]. The resource (Filhol and Tannier, 2014) is a set of 120 videos (30-second average duration, face and side views), prepared by professional translators from short news items written in French. We note that they chose not to use visual support while signing, except for occasionally long proper names written on a whiteboard to support inline fingerspelling[4].

3. Problems

Our recent studies have therefore mostly been based on videos of prepared captures which are short in length. To enable studies of higher-level discourse operations such as rhetoric argument building, semantic sectioning and long-term contextualising, we developed a need for both prepared and long Sign Language monologues. This section presents a few challenging problems that lie in the way of such corpus collection.

3.1. Protocol

Contrarily to spoken languages, SLs have no written form which can be read out and captured without a trace of the preparation. Therefore, prepared discourse usually has either to be delivered by heart, or to work around the lack of written support. This means that all SL corpora containing prepared discourse fall in either of the categories below:

1. delivered by heart, and short enough to allow hesitation-free discourse in one take after enough rehearsal;
2. longer rehearsed productions, at the expense of discourse flow, i.e. containing some disfluency in the contents, e.g. backtracks for omissions, stalled and repeated items, eyes disengaging and recollecting thoughts;
3. well organised discourse sequences, at the expense of allowing visible prep notes during the shooting process, e.g. a knee pad, a whiteboard facing the informant, a screen under the camera ("teleprompter" technique).

Case (1) was fit for the average 30-second item of the "40 brèves" corpus, as it allowed fully controlled output, still ensuring a reasonable prep time. One could learn long discourse like actors literally learn hour-long plays, but it would take the rehearsal process to a level beyond what can reasonably be expected from a corpus informant. Short-term memory here does not allow fully controlled productions that exceed a minute in duration.

[3]It shut down with the company in 2015.

[4]Fingerspelling is a way of spelling a written name or word using an alphabet of handshapes.

The remaining techniques do allow for longer discourse, but have symmetric advantages and drawbacks. On the one hand, memorised discourse (2) inevitably results in disruptions in the discourse flow. On the other hand, allowing visual support at the time of capture (3) generates false articulations and dynamics because of the physical constraints added to head orientation, eye gaze direction, etc.

A map task in the DGS Korpus project was planned and actually even aborted after its pilot test for that reason (Hanke et al., 2010). A similar task was nonetheless conducted in DictaSign, using a screen under the camera so that the informant could see it at any time. The result is a strong effect throughout the task, frequently as obvious as postures held with arms stopped in a physically constrained position, back hunched, eyes apparently locked on the helper screen, squinting and scanning its contents, until the signing resumes (fig. 1). In such case, like in the DGS Korpus map task, data distortion is too strong and the data is no more useable to inform studies of regular SL gestures.



Figure 1: Data distortion in the DictaSign map task.

3.2. Elicitation

Elicitation of long discourse in Sign Language also becomes tricky as the material given to elicit the productions can create bias.

The "40 brèves" corpus was signed by professional translators, and we observe that all productions (100%) have changed—if not completely reversed—the order in which named entities were introduced in the discourse. Translation being precisely the task of not changing the meaning, it shows that a lot is imposed by the language on discourse construction, which is what we want to study, and not bias. Now source text given for translation is known to generally have an impact in the target language equivalents, especially if translated by people not professionally aware of the problem and trained to overcome it. So to allow for a wider range of informants in future tasks, including native signers uncomfortable with text all together, it is necessary to avoid text as elicitation material.

A common technique to elicit narration is to ask the informant to read a picture story (approx. 1 page long), and to retell it. Several corpora cited above incorporate such a

task. Interestingly, very little reordering takes place with picture stories in the resulting discourse. Unlike translation of texts for which sentence-by-sentence progression is very likely wrong[5], a picture-by-picture progression this time is often the result everybody expects. However, it is probably due to the chronological nature of the story itself, imposing its own sequence of events, hence may be limited to the story-telling genre.

4. Proposal

We explained that when collecting long prepared discourse, one generally chooses between discourse flow and undistorted articulations.
But our objective is to enable discourse-level observations, and SL cues meaning "that is why", "besides", etc. are very often conveyed by subtle head tilts or slight rhythmic breaks not necessarily accompanied by annotatable dictionary signs. We therefore can afford to sacrifice neither flow nor articulatory correctness. This section suggests a protocol to collect corpus data with the most of both.

4.1. Protocol

Memory works only up to a limit in discourse length. So allowing for prep notes seems unavoidable to secure long discourse flow. Now when notes are allowed, experimenters generally try to maximise the ratio between visibility of the notes to the informant and invisibility on the resulting film. So they strive to embed the blackboard or monitor in the camera setup as much as possible, like the ideal teleprompter is invisible to the television viewer.
However, we argue that note support undermines the production *whenever* they are visible to the informant while they are performing, hence must be avoided then. And sparing the later viewer (e.g. linguist) of the sight of notes in the result is no serious concern to them. Arguably even, in view of analysing the data, making the observer oblivious to the moments when informants might be looking at their notes can turn out a serious problem. Being able to tell which articulations are not the result of an external cue increases confidence in taking them for linguistic features, as opposed to possible corpus artefacts.
Moreover, we suggest that note support undermines the production *only when* they are visible to the informant while they are performing. Whether or not they look at notes outside of what is later analysed is not relevant.
In summary:

- only note-supported footage can reliably yield long yet fully fluent discourse;
- only non-supported signing can be analysed as fully acceptable language productions;
- memory allows for non-supported but short stretches of clean discourse;
- any break in the flow of discourse should be identifyable in the resulting data.

Therefore, our proposition to collect long and fluent elicited signed discourse is:

- submit the assignment in advance to the informants, together with a description of the following rules, for them to prepare their intended productions;
- allow any preparation notes, drawings, personal recordings, etc. in the studio at the time of collection;
- prevent any visual access to them during signing;
- allow any number of breaks at any moment of the production, provided they are made obvious (e.g. signing stops, pause is called, informant must turn around to further read notes).

The periods between points where the signing breaks and resumes are then marked and edited out. The resulting corpus data is left with fully unsupported discourse, of arbitrary length, whose rhetoric and logical sequence is reliably planned, only containing a number of breaking points.

4.2. Elicitation

As we said in §3.2., it is hard not to bias the discourse construction order when expecting longer eventful stories. We acknowledge that pictures are a good alternative to avoid signed input, only we wish also to avoid pre-constructed discourse which tends to be induced by picture stories.
We also acknowledge that translation is in principle a good way of eliciting any exact meaning, but aside from the very limited world of native professionals, it is difficult to find reliable informants for such task.
However, there is a middle ground which translators are often trained to move to as a first step from text, namely *deverbalising* (Seleskovitch and Lederer, 1985). To deverbalise a text is to draw an explanatory diagram, in which all entities are present, together with their relationships and all other relevant information. Discourse is then built to from this drawing, in other words without any influence of the sequence and lexical choices of the source.
In elaborating new elicitation material, we chose to try out two different strategies, respectively inspired by the two above. One involves a picture sequence corresponding to the chronological sequence of the events depicted, about RMS Titanic. However, to avoid productions too strictly focused on signing each picture in turn, we have included informational pictures (size of ship, number of life boats, etc.) which do not contain an event and whose contents informants could choose to include anywhere they saw fit. It is too long to be included here but it is available online for download[6].
The other strategy consists in a two-page deverbalising diagram about the Omar Raddad affair, a famous unsolved criminal case in France. It bears no inherent order or sequence, but contains a lot of information which had to be ordered. It is given in figure 2, and is available online[7].

[5] Working with news texts a lot, we observed that the headline information came first in a paragraph while pieces of context and satellite information were always appended. When translated to SL, this was systematically reversed to contextualise first and bring the major (focused) clause last, inside the context previously set up.

[6] perso.limsi.fr/filhol/research/files/elicitation-Titanic.pdf

[7] perso.limsi.fr/filhol/research/files/elicitation-OmarRaddad.pdf

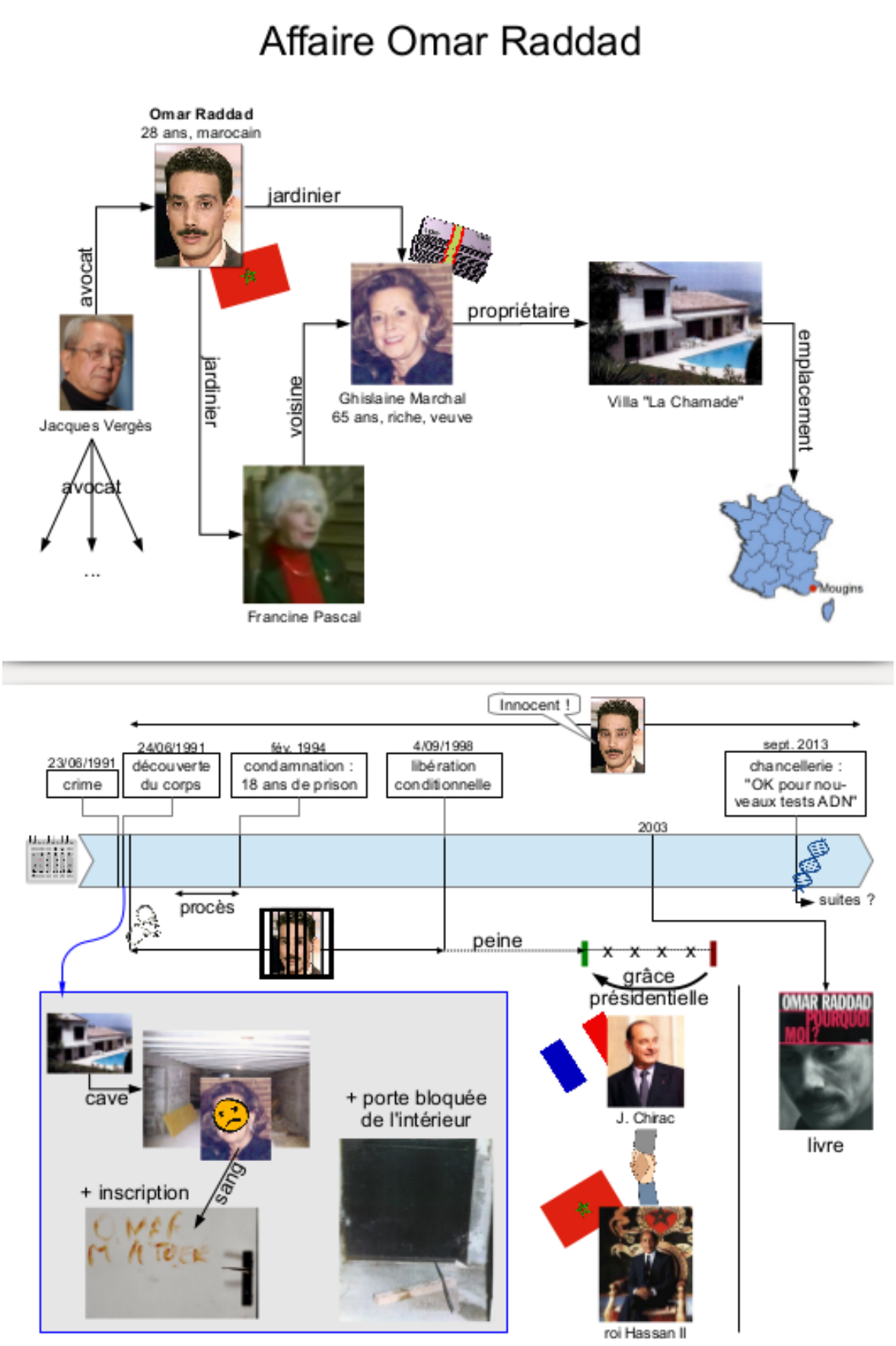


Figure 2: Two-page elicitation material for the “Omar Raddad” task

5. Capture and evaluation

We have conducted five sessions using the protocol and material above, each with a different informant. We included them as two tasks of the more general mocap[8] corpus collected last year (El-Fatah Benchiheub et al., 2016). In those sessions, a single face view was collected, including mocap markers on the face for the benefit of the other target studies (fig. 3). The data is still readable for traditional video observation.



Figure 3: Snapshot of the collected data.

This effort came as a first test before deploying it for more experiments, and possibly more lengthy elicitation yet, if evaluated positively. This section gives a few figures on the collected data, and evaluates the advantage of the protocol in terms of disfluencies per minute of resulting discourse.

Table 1 summarises the number of disfluencies found in a subset of the data collected using our protocol, per type and per session task. The "breaks" counted are the intentional (signalled) pauses taken by the informant. The disfluencies are counted and categorised in the following types: hesitations (H), filler/thinking gestures (F), reparation backtracks (B), interrputed/corrected signs (I) and superfluous repetitions (R).

Table 2 normalises the counts per minute of signing, giving both values including (*t.d.*: total disfluencies) and excluding (*n.b.d.*: non-break disfluencies) the called out breaks. The former gives a probable minimal rate of expected disfluencies in prepared discourse and will be compared to disfluency counts in non-paused discourse. The latter gives a lower rate, which quantifies the confidence in the data *between* the intentional breaks, subsequent studies now being able to call those out.

For comparison, we used some of the LS-COLIN data (non-chronological tasks), which we have full access to. Table 3 reports on the same counts in three sessions where informants explain the switch to the Euro currency in France, and their experience of it. The last column is the normalised number of disfluencies per minute of signed discourse.

[8]Motion capture.

Session	Duration	Breaks	H	F	B	I	R
Omar–S2	3 min 43 s	3	0	0	1	1	3
Omar–S3	3 min 15 s	4	4	0	1	4	2
Omar–S4	1 min 58 s	1	1	1	0	0	1
Titanic–S2	4 min 17 s	2	1	0	1	0	3

Average signing time without an intentional break: 56.6 s

Table 1: Disfluency and intentional break count per type in a collected data sample

Session	t.d./min	n.b.d./min
Omar–S2	2.15	1.35
Omar–S3	4.62	3.38
Omar–S4	2.03	1.52
Titanic–S2	1.63	1.17

Table 2: Normalised counts per minute of signing of the data analysed in table 1

Session	Duration	H	F	B	I	R	d./min
Euro–La	1 min 35 s	2	1	4	2	8	10.74
Euro–Kh	1 min	1	1	1	0	5	8
Euro–Ch	55 s	10	0	0	2	3	16.36

Table 3: Disfluency count per type and normalised counts per minute of signing in an LS-COLIN data sample

The disfluency rate here is over 11.7 per minute, as opposed to 2.6 when allowing intentional breaks, which is over four times less in density. When calling out the breaks, the rate is 1.855, i.e. 6.3 times less.

Admittedly, the elicitation material was different, thus the comparison bears some approximation. But apart from the material itself, the only major difference in the protocol used was the absence of an explicit permission to take breaks. We therefore believe the comparison to have relevance, especially as none of the observed ranges overlap.

This suggests that the proposed protocol significantly reduces disfluency in the resulting data, while still preserving it from distortion by external visual cues at the time of capture. Besides, the data pieces that are known not to deserve full trust in relevance are called out, which increases the observer's confidence in the remaining surface forms.

6. Conclusion

We have explained why it is difficult to collect long discourse that is both clean (undistorted) and prepared (constructed and flow intact) in Sign Language. With common studio elicitation techniques, captured discourse usually does away with one of those properties. We have proposed and tested an elicitation protocol which aims at ensuring most of both, and demonstrated some improvement in the produced data.

We have also produced elicitation material for two different tasks, based on two different strategies. One is derived from the translator's technique of deverbalising texts into diagram linking all elements of the source meaning; the

other remains closer to the more common technique of picture story elicitation.

Future work should include an evaluation of the respective impact of those strategies on the resulting data. We hypothesise that a trade-off exists here as well, this time between bias on discourse ordering (by picture stories, as noted in §4.2.) and bias on signing space usage (by deverbalised diagrams). Diagrams indeed tend to be mapped into signing space with little rearrangement by informants when they immediately make sense to them.

7. Bibliographical References

Crasborn, O. and Zwitserlood, I. (2008). The corpus NGT: an online corpus for professionals and laymen. In *3rd Workshop on the Representation and Processing of Sign Languages*, Marrakesh, Marocco.

Cuxac, C., Fusellier, I., Monteillard, N., Sallandre, M.-A., Jirou, G., Risler, A., Lejeune, F., Braffort, A., Choisier, A., Collet, C., Gherbi, R., Dalle, P., Jausions, G., and Lenseigne, B. (2002). LS-COLIN: Rapport de fin de recherche (final project report).

El-Fatah Benchiheub, M., Berret, B., and Braffort, A. (2016). Collecting and analysing a motion-capture corpus of french sign language. In *7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining*.

Filhol, M. and Tannier, X. (2014). Construction of a french–LSF corpus. In *Language resource and evaluation conference (LREC), Building and Using Comparable Corpora*.

Hanke, T., Hong, S.-E., König, S., Langer, G., Nishio, R., and Rathmann, C. (2010). Designing elicitation stimuli and tasks for the DGS corpus project. In *Theoretical Issues in Sign Language Research (TISLR conference)*, Purdue University, Indiana, USA.

Johnston, T. and Schembri, A., (2006). *Sustainable data from digital fieldwork*, chapter Issues in the creation of a digital archive of a signed language, pages 7–16. University of Sydney Press.

Johnston, T. (2008). *The Auslan Archive and Corpus. The Endangered Languages Archive*. University of London.

Matthes, S., Hanke, T., Storz, J., Efthimiou, E., Dimou, A., Karioris, P., Braffort, A., Choisier, A., Pelhate, J., and Sáfár, E. (2010). Elicitation tasks and materials designed for dicta-sign's multi-lingual corpus. In *4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, Malta.

Nishio, R., Hong, S.-E., König, S., Konrad, R., Langer, G., Hanke, T., and Rathmann, C. (2010). Elicitation methods in the DGS (german sign language) corpus project. In *4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, Malta.

Schembri, A. (2008). British sign language corpus project: Open access archives and the observer's paradox. In *3rd Workshop on the Representation and Processing of Sign Languages: Construction and Exploitation of Sign Language Corpora*, Marrakesh, Marocco.

Seleskovitch, D. and Lederer, M. (1985). Interpréter pour traduire. *L'Information Grammaticale*, 25:44–47.

Deep JSLC: A Multimodal Corpus Collection for Data-driven Generation of Japanese Sign Language Expressions

Heike Brock, Kazuhiro Nakadai
Honda Research Institute Japan Ltd.
Honcho 8-1, Wako-shi, Saitama, Japan
h.brock@jp.honda-ri.com, nakadai@jp.honda-ri.com

Abstract

The three-dimensional visualization of spoken or written information in Sign Language (SL) is considered a potential tool for better inclusion of deaf or hard of hearing individuals with low literacy skills. However, conventional technologies for such CG-supported data display are not able to depict all relevant features of a natural signing sequence such as facial expression, spatial references or inter-sign movement, leading to poor acceptance amongst speakers of sign language. The deployment of fully data-driven, deep sequence generation models that proved themselves powerful in speech and text applications might overcome this lack of naturalness. Therefore, we collected a corpus of continuous sentence utterances in Japanese Sign Language (JSL) applicable to the learning of deep neural network models. The presented corpus contains multimodal content information of high resolution motion capture data, video data and both visual and gloss-like mark up annotations obtained with the support of fluent JSL signers. Furthermore, all annotations were encoded under three different encoding schemes with respect to directions, intonation and non-manual information. Currently, the corpus is employed to learn first sequence-to-sequence networks where it shows the ability to train relevant language features.

Keywords: sign language, deep learning corpus, assistive language technologies

1. Introduction

Research has shown that information conveyed using Sign Language (SL) is much more comprehensible and accessible than written information display for the majority of native SL speakers (Traxler, 2000). However, human translation services are neither always available on demand, nor applicable in certain settings such as internal company meetings. The automatic display of spoken or written content by signing CG avatars or robot agents could therefore offer a cheap possibility to make information more accessible to Deaf or Hard of Hearing (DHH) individuals with lower written language literacy (Huenerfauth, 2008). Unfortunately, automated SL generation technologies have not yet reached a level of quality that would foster their full acceptance by DHH users (Kipp et al., 2011b). This is mainly due to the language's multi-dimensional character: information is not only conveyed using single gestures and finger movements, but also Non-Manual Signs (NMSs) such as body language and facial expressions. Furthermore, lexical items within a signed conversation can be dynamically changed to express spatial and temporal information (Huenerfauth and Hanson, 2009). This creates non-linear, multichannel dependencies that (a) complicate the translation of SL as compared to spoken languages and (b) require specific descriptions for motion synthesis. As a result, temporal and spatial relations of generated signing motion sequences might easily appear artificial and unnatural when applied to three-dimensional avatar animations (Kacorri et al., 2015).

To increase the acceptance of three-dimensional sign animations within affected users, effort is made to develop new methodologies that can better represent the temporal, spatial and multimodal aspects of signed content information (Ebling and Huenerfauth, 2015; Kacorri and Huenerfauth, 2016). To date, all of these methods rely on the concatenation and interpolation of pre-recorded sample sequences and separate lexical items. Systems that can reproduce natural signed utterances by intrinsic learning of their specific lingustic features are not known to be reported yet, commonly restricting existing data sources to non-repetitive corpus collections of signs and short phrases of specific content domains. Considering the recent success of deep neural networks in sequence generation tasks such as text-to-speech translations (Oord et al., 2016), one can expect similar architectures to also be meaningful for the synthesis of SL. To learn and evaluate respective network models, it is essential to have access to a suitable training corpus. In this work, we therefore present a fully annotated Japanese Sign Language (JSL) corpus that was specifically designed for the learning of deep sequence generation models for sign animation synthesis. Here, it is particularly important to ensure the accuracy, density and completeness of the recorded signing data up to subtle finger and facial movements and the repetitive occurrence of NMSs with wide intra-feature variability. Therefore, the presented deep JSL Corpus (JSLC) constitutes a collection of natural JSL sentences recorded with a multi camera optical motion capture system and includes various types of NMSs as well as variations in sentence intonations and signing speed. The corpus is designed to offer possibilities for easy data augmentation and ready to be applied under various deep network models. To bring the quality of automatically generated sign animation display to a next level of realism within the next years, it shall now be used for the development of an appropriate network model.

2. Sign Language Synthesis and JSL

Starting with the beginning of the last decade, various research efforts were made to enhance communication between hearing and non-hearing individuals, and to make information more accessible to DHH users. This led to a constant increase of assistive language tools, such as sign

translation software (Sun et al., 2013), dictionaries (Pro Deaf, 2017) or even commercial devices like the MotionSavvy tablet app (Motion Savvy, 2017). However, most of these technologies are not capable of processing continuous and natural sentence utterances, and even less are concerned with the synthesis of signed expressions from a given set of text or speech data. Current SL generation methods are either fully synthesized from form notation such as VRML (Kipp et al., 2011a) or semi-automatically synthesized using a corpus of pre-captured gestures (Gibet et al., 2016; Ebling and Huenerfauth, 2015). Most recent neural network models that achieved superior results in speech generation are not known to be reported for applications with SL. One of the main reasons for this leeway is the multi-dimensional character of continuous signing with its inherent use of NMSs and spatial and temporal references that impede the use of machine translation algorithms. Another reason is the necessity of an extensive, highly detailed and accurate motion data set that is very costly to obtain: respective data is best obtained by using a multi-camera optical motion capture system for body, finger and facial capturing, and requires specialized post-processing procedures to exclude marker occlusions and mislabeling. Besides, existing SL motion capture data sets are commonly not publicly available. This imposes the need to acquire new data for system development.

Most current sign animation technologies were developed for the processing of comparatively well-researched SLs such as American Sign Language (ASL) (Lu and Huenerfauth, 2011), German Sign Language (DGS) (Ebling and Huenerfauth, 2015) or French Sign Language (LSF) (Gibet et al., 2016). JSL in contrast remains a relatively unexplored language under both syntactic and semantic aspects. Grammatical structures and lexical items can be undefined or subject to regional and demographic variations. For this reason, the generation of JSL sequences also constitutes a rare research domain. To date, sign animations were created for the following two fields of application: medical content translation for enhanced doctor-patient communication (Nagashima et al., 2016) and the broadcasting of weather forecasts in the national TV program (Umeda et al., 2016). Both animations are based on semi-automatized synthesis methods that utilize data from a problem-specific corpus of optical motion capture data. These corpora contain a variant number of single lexical items and few short complete sentences that were each signed once by one single speaker. Such sparsity in the vocabulary domain and the absence of NMSs is likely unsuitable for the deep learning of signed sequence connections. Therefore, we built the present deep JSLC in a different way that is not known to be reported similarly by any other research yet.

3. Corpus Definition

A large amount of data is necessary to train deep sequence generation networks that could cover the full vocabulary spectrum of JSL. However as compared to speech or text data, the acquisition of detailed, accurate and complete SL expressions is a very expensive and time-consuming task. To evaluate the potential benefit of deep sequence generation techniques for natural sign display, it was considered sufficient to first train networks on a smaller subset of JSL vocabulary that could be extended for better generalization later. Therefore, we defined the proposed deep JSLC as a collection of sentence expressions and multi-word phrases within a dense vocabulary domain. Apart from the choice of repetitive vocabulary and content, this was achieved by the utilization of compound words built from lexical items and space and size classifiers (e.g. hospital signed as MEDICAL/ BUILDING and doctor signed as MEDICAL/ MAN, or to swim signed as SWIM and pool signed as SWIM/ PLACE).

Furthermore, the corpus should incorporate a combination of the most important and variant linguistic features of JSL that can hardly be reproduced by conventional generation methods in a satisfying way (Figure 1):

- **Direction:** Directional modification of lexical items conveys spatial and grammatical content information, such as the passive voice of words (e.g. BEING SPOKEN AT versus TO SPEAK) or the center of actions (e.g. TO GO and TO COME, TO SEE, TO BEG)
- **Syntax:** Intonation conveyed by syntactic NMSs (e.g. a raised eyebrow or head shaking) provides information on the sentence structure such as negation, past tense or question
- **Adjective inflection:** Positive, comparative and superlative forms of adjectives are expressed using a varying amount of facial NMSs (e.g. pressing eyes together, open mouth)
- **Content separation:** Lexical items are separated by a subtle nod of the head (e.g. my mother signed as ME/ MOTHER versus me and my mother signed as ME (+NOD)/ MOTHER)

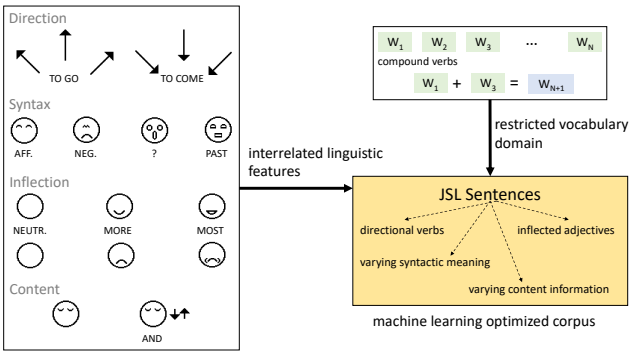


Figure 1: Signed expressions are difficult to convey in a natural way using virtual avatars. The fundamental structure of the corpus was designed specifically to enable a more reliable and accurate machine learning of contextual interrelation within JSL sentence expressions.

For use in deep neural network models, the previous constraints were used to define JSL sentences of large variations within every single linguistic feature and in between all features. In concrete, the main corpus was built as follows: to account for the dense domain criteria, 69 daily

conversation samples of 5-10 lexical items each were chosen as fundamental corpus sentence patterns from a set of intermediate-level SL practice sentences. These 69 sentence patterns were then modified to define 4 to 6 pattern variations by a random combination of dense-domain compound verbs and the previous linguistic features. The following sentences A, B, C and D for example constitute variations of one sentence pattern, whereas PT1 denotes a reference to oneself, PT2 denotes a pointing to the opponent person (respectively conversation partner) and PT3 denotes a pointing to a third person, object or place:

A PT3/ MOVIE/ BUILDING/ PT3/ INTERESTING/ MOVIE/ WATCH/ PAST (translating as 'I watched an interesting movie in this cinema.')

B MOVIE/ BUILDING/ PT3/ VERY INTERESTING/ MOVIE/ WATCH/ PAST/ PT2? (translating as 'You watched a very interesting movie in the cinema?')

C PT3/ MOVIE/ BUILDING/ PT3/ PT1/ SATO/ MAN/ MOVIE/ WATCH/ NO (translating as 'I did not watch an interesting movie in this cinema with Mr. Sato.')

D MOVIE/ BUILDING/ PT3/ PT1/ SATO/ WOMAN/ PRETTY INTERESTING/ MOVIE/ WATCH/ PAST (translating as 'I watched a pretty interesting movie in the cinema with Ms. Sato.')

Utilizing this sentence pattern variation strategy, a collection of 430 JSL sentences were defined to build the basic framework for corpus collection. These sentences constitute natural JSL expressions as expressed by native signers and therefore contain a very unbalanced word distribution: especially lexical items that support the semantic understanding within a signed sentence such as PT3 (providing contextual references to objects and persons) occur frequently within a JSL sentence. To reduce their impact on the network learning, we defined an additional set of 260 short training phrases. These phrases were built from the basic sentences and constituted a semantically meaningful succession of 3 to 6 less frequent words. Phrase variations taken from the sample sentences A, B, C and D were for example INTERESTING/ MOVIE/ WATCH/ PAST, MOVIE/ WATCH/ NO and SATO/ WOMAN/ PRETTY INTERESTING/ MOVIE/ WATCH.

4. Corpus Acquisition

All basic sentences and phrases were signed between one to three times in varying speed and sentence intonation by one Child Of Deaf Adults (CODA), leading to 931 corpus sentences and 502 corpus phrases with varying spatial and semantic content information. The data acquisition was furthermore assisted by a deaf native JSL speaker who supervised the grammatical correctness and naturalness of the signed corpus content.

During signing, the displacement of 123 markers attached to the signer's body was captured using an optical Vicon camera system of 48 cameras with a sampling rate of 120Hz. Additional data for extensive corpus annotation or the learning of sentence recognition networks was acquired using a Microsoft Kinect (sampling rate 30Hz) and a consumer video camera (sampling rate 60Hz). All capture modalities were synchronized via an external trigger and recorded a total of 10.384 signed utterances within a vocabulary of 197 lexical items.

To ensure sufficiently dense data for subsequent natural signing avatar generation, 92 out of the 123 optical motion capture markers were utilized for the acquisition of detailed finger and facial movements such as the blinking of eyes or the raise of eyebrows (Figure 2). These markers were of 3mm size, with 24 markers placed on each hand up to the wrist and 44 makers attached to the face of the signer. All Vicon recordings were post-processed to ensure correct marker labeling and to eliminate missing frames caused by marker occlusions during data recording.

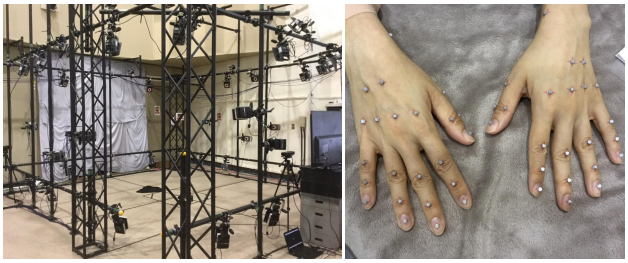


Figure 2: A set of 48 optical cameras was used to record the signed motion sentences. Marker were densely placed on body, finger and face of the sign speaker to obtain a highly-dimensional collection of sign motion data.

The cleaned data was made available in C3D and BVH format, which are two common data formats for the storage and processing of optical motion capture data. Whereas C3D contains the raw three-dimensional point clouds of all marker positions as obtained during the motion capture process, BVH contains kinematic information of a virtual character's body joints. To obtain such higher-level data, the raw C3D data was rigged onto the skeleton of a virtual avatar (Figure 3) and the three-dimensional joint position and rotation of 107 relevant skeleton joints (including finger joints and controllers for facial expression) saved as the file's main motion data streams. For the given problem, the BVH format should be considered particularly useful since newly generated motion streams can be directly transferred onto the corresponding virtual avatar for visual display and evaluation.

5. Corpus Annotation

The fundamental content of the deep JSLC was annotated in gloss notation with the support of the signer and the supervising native speaker during the process of corpus definition. Throughout data acquisition, the gloss notation was actively refined in real-time to ensure that every sentence was expressed in the most natural way of signing. Furthermore, every corpus sentence was annotated as a visual markup with the help of the JS Pad dictionary (Lab, 2016) for the creation of Japanese Sign Writing (Matsumoto et al., 2009) and the additionally captured video data. A sample annotation of the previous sentence variation A is shown in both gloss notation and visual markup in Figure 4.

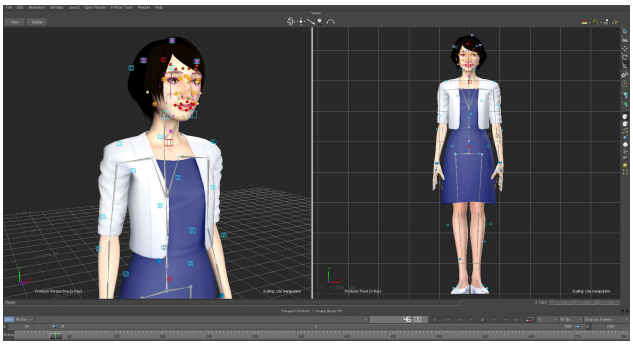


Figure 3: 3D model used to rig the raw motion capture data onto an avatar skeleton to provide joint position and rotation data in BVH format.

A major advantage of deep sequence generation models such as Recurrent Neural Networks (RNNs) is that they do not require the training data to be pre-segmented (Graves, 2013). However for eventual use in baseline networks or automatic sentence segmentation models, additional time annotations for separation of all lexical items in the corpus were determined. These were based on changes in hand and finger shapes as well as motion directions using the synchronized video data.

5.1. Encoding

Three different types of encoding were chosen for subsequent corpus deployment in potential neural network models. Here, the idea was to provide different types of encoding to evaluate whether the presence of specific linguistic feature information could reproduce non-manual signing aspects in a better way.

The first encoding constitutes a simple one-hot encoded representation of all occurring lexical items and does not contain any additional information on the linguistic features incorporated within the deep JSLC. The second encoding constitutes the simple one-hot encoded representation plus additional information on all linguistic features as additional elements of the one-hot encoded vectors for each respective sign. In concrete, these additional vector elements represent information on:

1. Use of left or right hand to convey primary sentence meaning
2. Start location of the primary hand within a position segment defined in relation to the upper body
3. End location of the primary hand within a position segment defined in relation to the upper body
4. Signing in active or passive form (if applicable, else none)
5. Stage of adjective inflection (if applicable, else non)
6. Inclusion of interrogation (if applicable, else non)
7. Inclusion of negation (if applicable, else non)

The third encoding was based on the visual markup annotation, following the representation of Sign Writing Markup Language (SWML) (Costa et al., 2001). Here, every lexical item was encoded as the combination of its Sign Writing components in SWML. For example the lexical item MOVIE is built by one head icon with the SWML index 04-01-001-01-01-01, two handshape icons (left and right hand) with the SWML indices 01-05-001-01-01-03 and 01-05-001-01-01-11 and two directional icons (for left and right hand) with the SWML indices 02-03-006-01-01-13 and 02-03-006-01-02-01.

6. Corpus Deployment

We evaluated the corpus usability for the learning of JSL sentence structure with a straightforward modification of the sequence to sequence model (Seq2Seq) for English-French translation (Sutskever et al., 2014). However, the determination of generated sequence quality is a difficult task that is commonly performed by rigging the generated sequences on a virtual character, and by subsequently assessing their naturalness and understandability in user studies. For this reason, we used the corpus in a reversed recognition scenario here: the acquired motion data streams were first encoded by a RNN cell and then passed to a decoder RNN cell providing an output expectation of the expressed sentence. Since the Seq2Seq model is bidirectional, evaluation of corpus efficiency can be expected to also hold valid for generation scenarios.

Using a smaller subset of 810 of the full corpus sentences only, we first learned several variations of a basic Seq2Seq network with 1 to 3 hidden layers and a varying number of cells per layer for all three encodings. Here, it should be noted that deep networks are commonly trained on much larger data sets. However since SL data collections cannot be acquired as easily as text or image data, the number of available training data can already be considered numerous for the given data content. The optimizer used during network training was an Adagrad optimizer, and both Gated Recurrent Units (GRUs) and Long Short-Term Memory Units (LSTMs) were used as cell types. Results indicate a constant decrease of training loss over time within all network models, whereas the Sign Writing based encoding performed slightly better than the two simpler, gloss based encodings. Best results with maximal test accuracy of $\approx 20\%$ after 2000 training epochs were achieved with 1 hidden unit of 256 LSTM cells (Figure 5). However, all network architectures showed significant overfitting and did not generalize well on unknown test data: especially rare words were easily misclassified and labeled as frequent words of little discriminative character (e.g. PT3, MAN or WOMAN). Given the unbalanced word distribution within JSL sentences, this is not surprising, and a better balanced corpus should improve network quality considerably.

To test eventual effects of an enlarged data collection, we included the 502 additional phrases in the training data and learned a new recognition network using the same encoding and network parameter in the next step. Test accuracy of the network raised to $\approx 40\%$ after 2000 training epochs, while overfitting was significantly decreased. The subset of underrepresented word phrases that can be freely added to

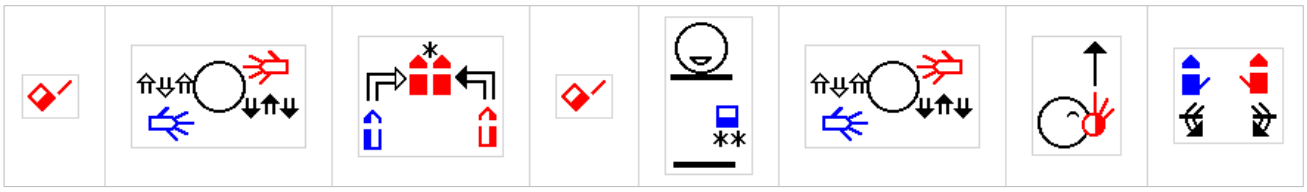


Figure 4: Visualization of previously introduced sentence variation A in Japanese Sign Writing and its corresponding gloss annotation.

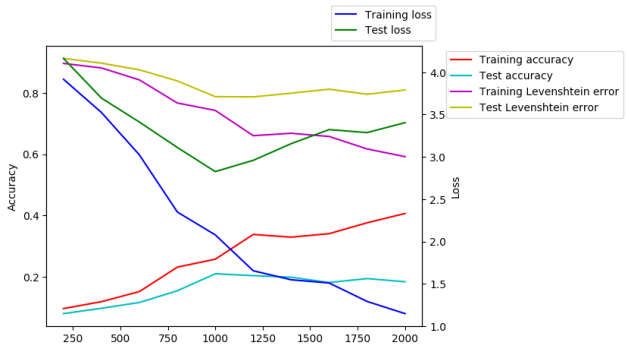


Figure 5: Evolution of training and testing accuracy and loss for sentence recognition as obtained with a standard sequence to sequence model and 1 hidden layer of 256 LSTM cells for a smaller subset of the full corpus sentences only. Over 2000 epochs, a recognition network of $\approx$ 20% test accuracy was learned. The network is strongly overfitting.

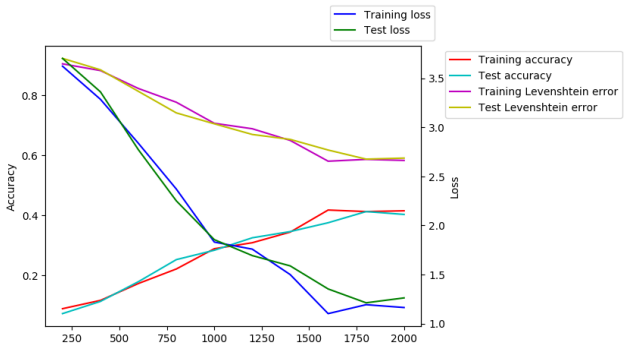


Figure 6: Evolution of training and testing accuracy and loss for sentence recognition as obtained with a standard sequence to sequence model and 1 hidden layer of 256 LSTM cells using a larger number of corpus data for network training. Over 2000 epochs, a recognition network of $\approx$ 40% test accuracy was learned that generalizes better to unknown data.

the main training data should therefore be considered as an useful extension of the full sentence collection.

7. Discussion

Previous results suggests that the presented corpus is generally capable to train a Seq2Seq network that understands common multi-modal interrelation within JSL utterances. Test accuracies of $\approx$ 40% do not appear sufficient for application in real-life scenarios yet, but reach the best obtained accuracies of similarly continuous and weakly supervised sentence recognition scenarios (Koller et al., 2016). To date, no specific modifications of the model parameters were performed, and we expect to achieve better results of improved accuracy and smaller loss by adding a suitable data embedding and an attention model. Moreover, it was shown that a higher number of training data is beneficial for network learning. Better recognition and generation networks should therefore be achieved by further augmenting the corpus size and balancing out the general word distribution. Thanks to the corpus design with its repetitive occurrence of identical sentences and phrases, respective data can be synthesized relatively easily from the existing data in the following, using sequence alignment methods such as squeezing, stretching or undersampling.

All in all, we believe that the specific characteristics of its corpus design and content make the present deep JSLC a very valuable collection of JSL motion data. It shall now be used to define suitable network parameters and variations such as attention models, and to subsequently learn a wide variety of sequence generation networks. In a last step, the usability and eventual benefit of the trained generation networks shall be evaluated with respect to realism and naturalism of the resulting animations.

8. Conclusion

We presented a new corpus of JSL sentence expressions for application in advanced data-driven deep neural networks. This corpus was defined so that it can easily be applied to advanced sequence generation models for the synthesis of Sign Language animations. As opposed to previous SL corpora of similar application purpose, the corpus was built from randomized variations of pre-defined sentence patterns only. It incorporates many spatial and temporal references as well as non manual signs to intrinsically learn interrelations of relevant linguistic features within signed expressions or conversations. The corpus is extensively annotated in gloss and visual mark up, and its signed data content made available using three different motion sensing modalities (motion capture, depth images and video images) that can be utilized in various additional corpus works. First experiments showed the general applicability of the presented corpus in sequence to sequence networks for sentence recognition. In the following, these networks

shall be enhanced and modified to provide intelligent networks that can help to generate naturally signing avatars in the future.

9. References

Costa, A. R., Dimuro, G. P., et al. (2001). Supporting deaf sign languages in written form on the web. In *10th World Wide Web Conference, Hong Kong*. Home Page of Web and Society Track, `http://www10.org/cdrom/posters/frame.html`.

Ebling, S. and Huenerfauth, M. (2015). Bridging the gap between sign language machine translation and sign language animation using sequence classification. In *Proceedings of the 6th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT)*.

Gibet, S., Lefebvre-Albaret, F., Hamon, L., Brun, R., and Turki, A. (2016). Interactive editing in french sign language dedicated to virtual signers: requirements and challenges. *Universal Access in the Information Society*, 15(4):525–539.

Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Huenerfauth, M. and Hanson, V. (2009). Sign language in the interface: access for deaf signers. *Universal Access Handbook. NJ: Erlbaum*, 38.

Huenerfauth, M. (2008). Generating american sign language animation: overcoming misconceptions and technical challenges. *Universal Access in the Information Society*, 6(4):419–434.

Kacorri, H. and Huenerfauth, M. (2016). Selecting exemplar recordings of american sign language non-manual expressions for animation synthesis based on manual sign timing. In *Proc. SLPAT 2016 Workshop on Speech and Language Processing for Assistive Technologies*, pages 14–19.

Kacorri, H., Huenerfauth, M., Ebling, S., Patel, K., and Willard, M. (2015). Demographic and experiential factors influencing acceptance of sign language animation by deaf users. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '15, pages 147–154, New York, NY, USA. ACM.

Kipp, M., Heloir, A., and Nguyen, Q. (2011a). Sign language avatars: Animation and comprehensibility. In *Intelligent Virtual Agents*, pages 113–126. Springer.

Kipp, M., Nguyen, Q., Heloir, A., and Matthes, S. (2011b). Assessing the deaf user perspective on sign language avatars. In *The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '11, pages 107–114, New York, NY, USA. ACM.

Koller, O., Ney, H., and Bowden, R. (2016). Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3793–3802.

Lu, P. and Huenerfauth, M. (2011). Data-driven synthesis of spatially inflected verbs for american sign language animation. *ACM Transactions on Accessible Computing (TACCESS)*, 4(1):4.

Matsumoto, T., Kato, M., and Ikeda, T. (2009). Jspad: A sign language writing tool using signwriting. In *Proceedings of the 3rd International Universal Communication Symposium*, IUCS '09, pages 363–367, New York, NY, USA. ACM.

Motion Savvy. (2017). UNI sign language translation tablet. `http://www.motionsavvy.com/index.html`. Accessed: 2017-09-21.

Nagashima, Y., Watanabe, K., Terauchi, M., Kato, N., Uchida, T., Umeda, S., Miyazaki, T., Azuma, M., and Hiruma, N. (2016). A support tool for analyzing the 3d motions of sign language and the construction of a morpheme dictionary. In *International Conference on Human-Computer Interaction*, pages 124–129. Springer.

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Pro Deaf. (2017). Libras sign language translator. `http://prodeaf.net/en-us`. Accessed: 2017-09-21.

Sun, C., Zhang, T., Bao, B.-K., Xu, C., and Mei, T. (2013). Discriminative exemplar coding for sign language recognition with kinect. *IEEE Transactions on Cybernetics*, 43(5):1418–1428.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Traxler, C. B. (2000). The stanford achievement test: National norming and performance standards for deaf and hard-of-hearing students. *Journal of deaf studies and deaf education*, 5(4):337–348.

Umeda, S., Uchida, T., Azuma, M., Miyazaki, T., Kato, N., and Hiruma, N. (2016). Automatic production system of sign language cg animation for meteorological information.

10. Language Resource References

Matsumoto Lab. (2016). *Japanese Sign Writing Dictionary*. Gifu University, JSPad executable, 0.3.1.

Modeling French Sign Language: a proposal for a semantically compositional system

Mohamed Nassime Hadjadj, Michael Filhol, Annelies Braffort
LIMSI–CNRS B.P. 133, 91403 Orsay cedex, France
{mohamed.hadjadj, michael.filhol, annelies.braffort}@limsi.fr

Abstract

The recognition of French Sign Language (LSF) as a natural language in 2005 created an important need for the development of tools to make information accessible to the deaf public. With this prospect, the goal of this article is to propose a linguistic approach aimed at modeling the French sign language. We first present the models proposed in computer science to formalize the sign language (SL). We also show the difficulty of applying the grammars originally designed for spoken languages to model SL. In a second step, we propose an approach allowing to take into account the linguistic properties of the SL while respecting the constraints of a modelisation process. By studying the links between semantic functions and their observed forms in Corpus, we have identified several production rules that govern the functioning of the LSF. We finally present the rule functioning as a system capable of modeling an entire utterance in French sign language.

Keywords: French sign language, computer modelling, formal grammar

1. Introduction

Since the '60s, numerous efforts have proposed linguistic models to describe sign languages (Stokoe, 1960). However, none of the proposed studies has yet succeeded to fully describe the specificities of SL, such as the use of the signing space and the simultaneous articulation of multiple channels. Thus, researchers in Sign Language Processing (SLP) must model languages without fine linguistic descriptions, without a written form and with a limited amount of corpus (Cuxac and Dalle, 2007). In this context, most SL modeling approaches assume that the difference of modality has a minor impact on the organization of the utterance. The modeling of SLs is then carried out primarily using models originally designed for spoken languages. This operation implies linearity constraints that characterize other languages. The aim of this article is to propose a modeling approach to the French Sign Language (LSF), allowing to take into account the specificities relating to their visual-gestural modality.

In the section below (section 2), we present the most important approaches for the modeling of SL. Section 3 present our corpus study methodology to identify production rules that govern the functioning of the LSF. Finally, Section 4 proposes the combination of rules as a system capable of modeling an entire utterance.

2. Research in Sign Language Processing

Research in SLs modeling can be classified into two categories. In the first category, models were based on approaches initially proposed to model spoken languages. In the second category, models were designed specifically for SLs. In the following, we present some examples of approaches that love been developed with aim to translate from text to SL.

2.1 Models initially dedicated to spoken languages

The two projects **Team** (Zhao et al., 2000) and **ASL Workbench** (Speers, 2001) present a translation system from English to ASL using a syntactic transfer. To generate ASL, **Team Project** represents statements with the tree-adjoining grammar (STAGs) (Shieber, 1994). **Workbench**, for its part, use a LFG grammar (Kaplan and Bresnan, 1982). Under the translation project **Visicast** Marshall and Sàfàr (2003), propose a translation system from English to British Sign Language (BSL), this time using a semantic transfer. The semantic presentations in the form of a DRS (Discourse Representation Structure) structure are converted into an HPSG representation (Pollard and Sag., 1994), the utterances are generated using the HamNoSys phonetic model (Prillwitz et al., 1989).

The grammars initially developed to model spoken languages predict a systematic linear sequence of units. In SL models based on these grammars, a signed structure is considered the equivalent of a sentence and a gestural unit as the equivalent of a word. However, a considerable part of LSF contains structures that do not present a linear order (Garcia et al., 2010). The models based on spoken languages are not very efficient to fully describe SLs. To support this statement, we take the following structure as an example: "The city located next to Red Sea", its interpretation in LSF involves gestures that are specific to visual-gestural languages. Signed utterances do not only form a sequence of signs. We have rather a structure as described in figure 1.

1. The sign unit: Red Sea.
2. A gaze direction activates a part of the signing space. The placement of the flat weak hand refers to the Red Sea or what is defined in SL linguistics as "Classifier (proform)".
3. The pointing sign (sign that points to a referent), the gaze direction is always maintained.
4. The sign unit: City



Figure 1: form of the structure “The city located next to Red Sea” in LSF

2.2 Models dedicated to sign languages

The model of Huenerfauth (2006), then taken up by López-Colino (2012) proposes an approach to take into account the multilinearity observed in SL. His model named "P/C" defines two parts to describe a signed structure: The "C-node" corresponds to the linear order of the constituents. The "P-node" allows to segment a node in a structure to create new parallel axes. This configuration allows to simultaneously specify several articulations. However, the different axes are constrained by the boundaries of a partition. Consequently, if this approach offers a way to represent multilinearity, it does not allow to deal with complex synchronization patterns.

3. Presentation of our approach

To describe the linguistic organization of SL, we adopt an approach that does not assume the formal requirements of linear models. In other words, no hypothesis or prediction of a linguistic functioning is advanced beforehand. We rely on a more general descriptive approach to integrate all of the linguistic phenomena observed in SL corpus.

3.1 The key concepts

Our approach is to define, based on corpus studies, a systematic link between **"observable forms"** and what we call **"semantic functions"**. By observable form, we mean any gestural articulation, including the different synchronizations that take place between them. For example, the following articulations:

A: "Eyebrow raising"

B: "Move the index finger down"

Each of these two articulations can be considered as a form. Also, an observable form may include an overlap of A and B. This type of synchronization is also a form.

The notion of semantic function designates an interpretation of such observable form. Contrary to the notion of "signified" in general linguistics, reserved only for concepts of lexical signs, the notion of semantic function in our approach can be linked to different levels of interpretation. In this sense, the examples below are considered as possible functions:

C: the concept of "pen"

D: expression of doubt on a variable element

E: location of an object (*obj1*) in relation to another object (*obj2*)

SLs are, like all languages, considered as a system governed by a set of rules shared by a linguistic community. Our goal is to identify these rules with an experimental study. Any systematic association between an observable form and a semantic function gives rise to a rule, which participates in the linguistic organization of the language. As part of a project to generate the LSF, we identify **production rules** (links from function to form). Once a production rule is defined, it is formalized with the AZee language (Filhol, 2014). Each rule consists of three elements:

An identifier: usually an abbreviated name for its semantic function, e.g. “pen”.

The arguments of the rule: this is the set of parameters on which the rule may depend, e.g. “obj1”&”obj2”.

Associated form: these are the invariant forms of the rule and their possible dependencies to the parameters. This includes all the joints and the necessary synchronization constraints between them.

Box diagrams (figure 2), more explanatory than AZee code, can illustrate a production rule. In those diagrams:

- The horizontal axis represents the production time.
- The boxes represent time intervals in which an articulation must take place. The articulators are set in bold; their positions are given in italics
- The blue boxes are invariant specifications. The boxes in red represent the time intervals during which an argument is to be produced.

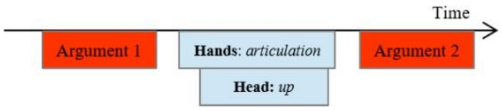


Figure 2: Example of a production rule

3.2 Methodology

The methodology we propose to define production rules consists in performing function-to-form iterations to identify a group of identical parameterized forms associated with the same semantic function. We present the different steps of this approach in the following sequence:

1. The first step is to begin with a starting criterion X to be specified. It can be either a form or a function.
2. Locate and list all occurrences of X in a selected corpus.

   Nocc refers to the number of occurrences found.

3. For each occurrence of criterion X, associate description elements that are:

- Elements of interpretation if X is a criterion of form.
- Observable forms if X is a semantic function criterion

4. Identify groups of occurrences sharing identical elements.
   - *Ngp* is the number of groups identified, groups are numbered from 1 to *Ngp*
   - For each $k \in [1..Ngp]$: D_k is the set of description elements that have been associated with all occurrences of the group *k*
   - *Nout* is the number of occurrences that are not included in any group.
5. **If** X is a function, which may depend on arguments {*A1*, *A2*, …, *An*}, and *Ngp* = 1, and *Nout* < 15% of all occurrences in the only group

   Then: we define a production rule for the function X and its associated form. It is specified by the triplet:

   - Identifier = X
   - Arguments = $\{A_i\}$
   - Form = *D1*

   Else: **for each** group k, k ∈ [1 ... *Ngp*]: proceed to step 1 with the new criterion X.k, specified by D_k

3.3 AZee formalism

To describe forms of the last D_k, we use the AZee language, which is a functional language first described in (Filhol et al., 2014), where a more detailed specification can be found. It defines a specific type "score", whose values are full specifications of a signed articulation (the name of this type is an analogy to a music score on which several voices are specified in parallel). It allows to define "AZops", which are equivalent to functions in a programming language. To define a production rule, one defines an AZop, with named arguments for parameters and with a score as a return value. Applying a rule is like applying the function, with a value for each parameter.

Figure 3 gives an AZop example for the rule placing a weak hand classifier in the signing space, which depends on a classifier "C" (a set of additional constraints to apply to the hand), and on a target point "loc". It produces the arrangement given in the box diagram of figure 4, where boxes (1) and (2) are manual specifications of a small downward movement establishing the weak hand at the argument location. Box (3) specifies the eye gaze, which must also be directed to the same target point. In the code, indentation denotes argument nesting under their header lines. This makes code more readable, but a bracketed notation is also possible.

Lines 3 and 5 declare the AZop's arguments, each followed with "nodefault", i.e. they are mandatory when applying the AZop. The "sync" operator on line 7 takes a list of named boxes to arrange on a timeline and returns a score, which is the return value of the AZop. The three boxes are: "classmvt" (box 1, from line 8) specifying the downward movement," classcfg" (box 2, line 35) configuring the hand with the classifier constraints, and "eyegaze" (box 3, line 45), specifying the eye gaze direction. Lines 42-44 synchronise the start and end boundaries of box 2, and lines 50-52 those of box 3. References to the arguments are prefixed with an '@'. Dependencies on the target location of the classifier are visible on lines 17, 34 and 49. The dependency on the classifier itself appears on line 39.

```
1  "place-classifier-in-space"
2  azop
3      'classifier  # type: AZOP (h:SIDE -> CSTR)
4      'nodefault
5      'loc         # type: POINT
6      'nodefault
7      sync
8          'classmvt  # BOX 1
9          sequence
10             key
11                 0
12                 place
13                     translate
14                         site
15                             'PA
16                             w
17                         @loc
18                     scalevect
19                         up
20                         small
21             transition
22                 1
23                 path
24                     'straight/accel
25                     site
26                         'PA
27                         w
28             key
29                 1
30                 place
31                     site
32                         'PA
33                         w
34                     @loc
35         'classcfg  # BOX 2
36         key
37             1
38             apply
39                 @classifier
40                 'h
41                 w
42         'start/end
43         'classmvt:0:0
44         'classmvt:-1:0
45         'eyegaze  # BOX 3
46         key
47             1
48             lookat
49                 @loc
50         'start/end
51         'classmvt:0:-.5
52         'classmvt:-1:-1
```

Figure 3: AZop example for the rule placing a weak hand classifier in the signing space.

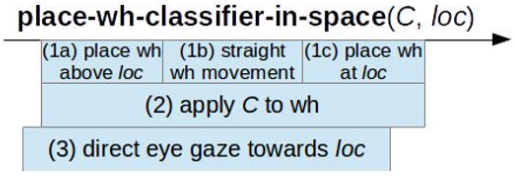


Figure 4: Box diagram example for the rule placing a weak hand classifier in the signing space

4. Juxtaposition as a form

As presented in Section 2, most work in automatic processing of sign languages follows the same process of analysis applied to spoken languages. Thus, the application of derivational grammars on SL requires a systematic

sequentiality between the signed units. In our approach, no production rule is defined to satisfy constraints assumed by a formal grammar. Any production rule allowing juxtaposition of signed units must be motivated by a link from function to form.

In order to explain linear structure nonetheless, we begin this study with a criterion of form J: the juxtaposition of two successive and interpretable items. This juxtaposition can relate to the succession of two interpretable units such as "country" and "Brazil" but also between “country Brazil" and its geographical location. We present in the following sub-sections the corpus on which we conducted this study as well as the results obtained.

4.1 Corpus

The corpus we relied on for this study consisted of 40-news item, each is signed by 3 professional signers, totaling 120 videos, or one hour of journalistic signing (Filhol and Tannier, 2014).

4.2 Iteration J

Iteration J: juxtaposition of two units, item1 and item 2

- Nocc = 321
- Ngp = 7
- Nout =21

Group J.1: item 2 gives the status of item 1. This may concern its name, status, property...

Group J.2: item 1 is located in relation to item 2

Group J.3: Item 1 presents the context of item 2

Group J.4: Finger spelling

Group J.5: Chronological sequence of two events

Group J.6: Negation of the first item and assertion of the second item

Group J.7: Enumeration

4.3 New iterations from group J.1

In the framework of this article, we present mainly the study of occurrences from group J.1. Functions from other groups require a finer analysis to be considered as production rules.

Function Criterion J.1: « item2 describes the State of item1

- Nocc = 153
- Ngp = 3
- Nout = 11

Group J.1.1: movement of the chin upwards on the beginning of item2; transition time of 2~3 frames between item1 and item2

Group J.1.2: movement of the chin upwards on the beginning of item1; transition time of 2~3 frames

Group J.1.3: longer transition time (8~9 frames) between the two items Iteration

Form Criterion J.1.1: Juxtaposition of item1 and item2; chin moves up on item2; 2~3 frames between the two items

- Nocc = 70
- Ngp = 1
- Nout = 5

Single group 1: item2 gives additional information on the item1.

Example: "Pierre, aged 25, is taken hostage"; Item1 = Pierre, Item2 = 25 years old

Function Criterion J.1.1.1: side information added

- Nocc = 65
- Ngp = 1
- Nout = 0

Methodological condition (5a) is verified. This therefore raises a **production rule**, specified as:

- identifier: item1 is given the additional side information item2 (**add-info**)
- arguments: base_item (item1), add_info (item2)
- form: see figure 5.

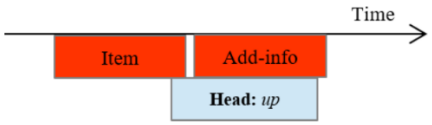


Figure 5: Form of add info rule

Form Criterion J.1.2: Juxtaposition of item1 and item2; chin moves up on item1; 2~3 frames between the two items

- Nocc = 33
- Ngp = 1
- Nout = 6

Single group 1: item2 is to be understood as the category of item1

Example: item1 = “country”; item2 = “Brasil”; combined interpretation = “Brasil”

Function Criterion J.12.1: item2 is to be understood as the category of item1

- Nocc = 27
- Ngp = 1
- Nout = 0

Production rule J.1.2.1

- identifier: item2 is to be understood as the category of item1**(cat)**
- arguments: base item (item1), category (item2)
- form: see figure 6.

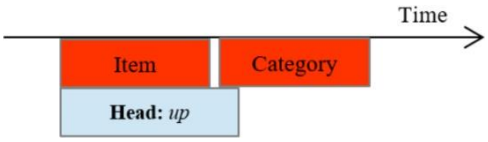


Figure 6: Form of cat rule

Form Criterion J.1.3: Juxtaposition of item1 and item2; longer transition time (8~9 frames) between the two items

- Nocc = 39
- Ngp = 1
- Nout = 6

Single group 1: item2 is the point being made about item1

Example: item1 = "power"; item2 = "tourism"; combined interpretation = "the strength/power is tourism"

Function Criterion J.1.3.1 : item2 is the point being made about item1

- Nocc = 33
- Ngp = 1
- Nout = 0

Production rule J.1.3.1

- identifier: item2 is the point being made about item1 **(info-focus)**
- Arguments : base_item (item1), focus (item2)
- Form: see figure 7.

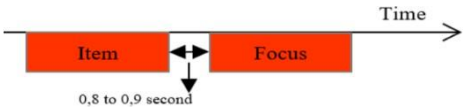


Figure 7: Form of info-focus rule

We present in the table below (table 1) a summary of all the iterations carried out as well as the production rules that were identified from this study.

5. Production rules as a system

The recursive aspect of rules, the embedding of one structure within another, is a feature that generates an infinite set of utterances from a finite set of rules. Each of the rules identified in this study depends on a nesting of arguments to generate a language. In other words, their production requires at least a level of nesting. We hypothesize that this nesting can reach several levels in order to model structures in LSF from a finite set of rules. In order to evaluate this hypothesis, we have described 14 news items of the corpus of the 40. We used production rules observed in this study but also other rules identified in previous studies.

To formalize the complete structure of a news item, we have first described combinations of forms by rules carrying an appropriate semantic function. For example, the description of the succession of the COUNTRY and BRAZIL units, if the observed form allows, is done from the production rule **cat** (J.1.2.1). Once first-level production rules are defined, we try to find production rules that can combine them. For example, the following form in LSF "tourist town Dahab" (figure 8) is described from the rules of production by the nesting presented in figure 9.

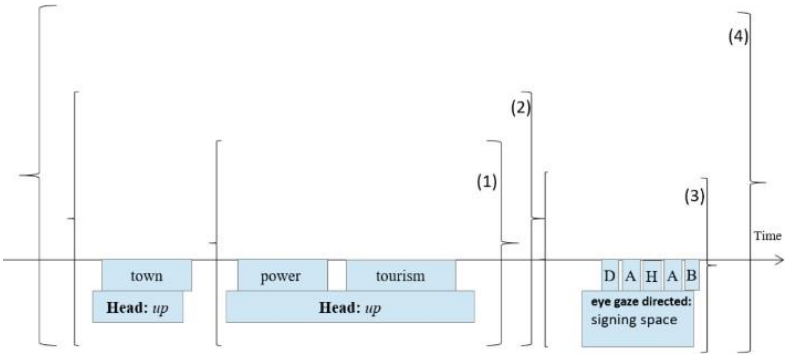


Figure 8: Form of the structure "tourist town Dahab" in LSF

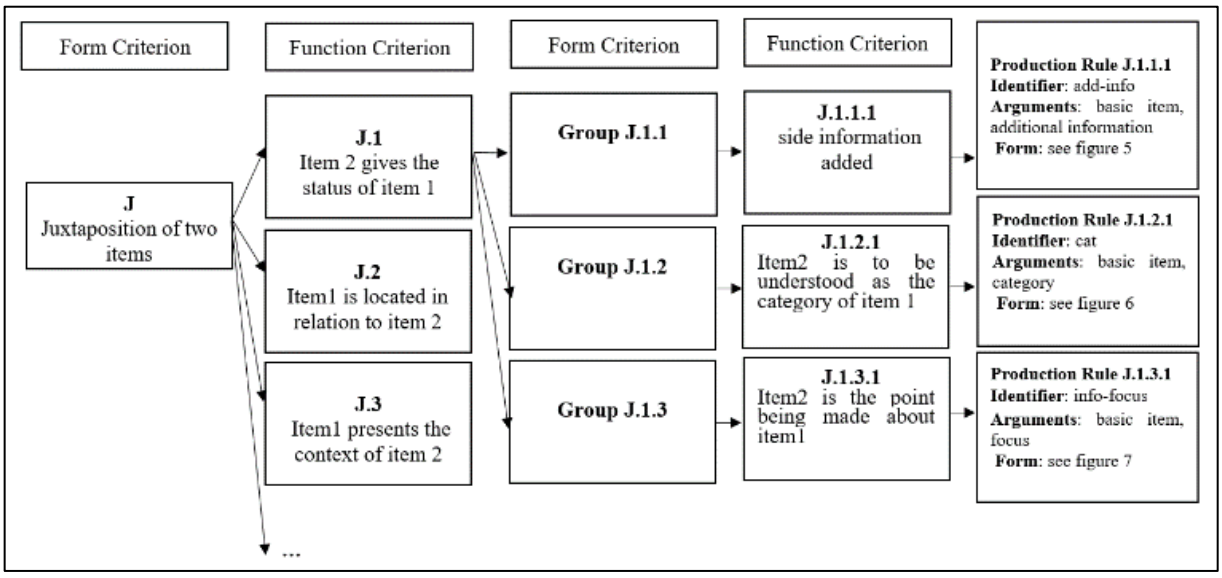


Table 1: synthesis of the study

1. The partition inside brackets 1 corresponds to the form generated by the **info-focus** rule (J.1.3.1). The second argument "tourism" is a focus on the first argument "power"

2. The partition inside brackets 2 corresponds to the form generated by the **add-info** rule. It takes as the first argument "city", its second argument is additional information "power tourism" (the accolade 1)

3. The partition inside brackets 3 corresponds to the form generated by the spelling rule (J.4). It takes a succession of letters as argument.

4. The partition inside brackets 4 corresponds to the form generated by the rule **cat**. It takes as its first argument the rule **add-info** (the accolade 2); its second argument gives the category of the first argument, its name "Dahab" (the brackets 3).

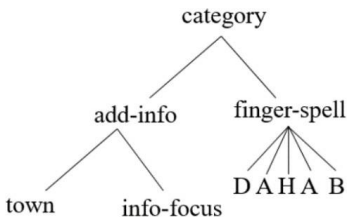


Figure 9: Functional tree "tourist town Dahab" in LSF

From this example, we observe that production rules create a system that allows to produce complete structures from multilevel nesting while respecting the form of the rules and their appropriate functions. Following the same process, we managed to describe most of the structures of the 14 news item. On 321 juxtaposition link, only 21 remain unexplained, often groups of forms with non-identifiable semantic functions. With this experiment, we have moved from a local function-form link that characterizes a single rule to the application of the same principle to describe this time a semantic composition of a structure that requires the nesting of several rules.

6. Conclusion et perspective

This article proposed a linguistic approach to formally describe LSF from corpus data. Using an LSF corpus, we have identified several production rules (semantic function to form links) that allow to juxtapose signed units. Then, we described how these rules could be used as a system to model an entire utterance in LSF. In future work, our goal is to study the possible combinations between the identified production rules. The definition of these patterns should allow to develop a grammar able to evaluate the acceptability of a rule structure in LSF. It should yield a linguistic model more flexible than a syntagmatic grammar.

7. Bibliographical References

Cuxac, C and Dalle, P (2007) Problématique des chercheurs en traitement automatique des langues des signes, volume 48 of Traitement Automatique des Langues. Lavoisier.

Filhol, M ., Hadjadj, MN and Annick. Choisier (2014) Non-manual features: the right to indifference. In Language resource and evaluation conference (LREC), 6th workshop on the representation and processing of Sign Language: beyond the manual channel, Reykjavik, Islande.

Filhol, M and Tannier, X (2014) Construction of a french- lsf corpus, building and using comparable corpora. In Proceedings of LREC 2014

Garcia, B. L'Huillier ., MT, Fremeaux, A ., and Makouke,D. (2010). Discursive and morphological processes in the formation and the stabilisation of lexematic units in french sign language. In Theoretical Issues in Sign Language Research (TISLR), UCL, London

Huenerfauth, M (2006) Generating American Sign Language Classifier Predicates for English-to-ASL Machine Translation. PhD thesis, University of Pennsylvania, Philadelphia, PA

Kaplan, R M and Bresnan J (1982) LexicalFunctional Grammar: A Formal System for Grammatical Representation. In Bresnan 1982b, 173–281. Reprinted in Dalrymple et al. (1995: 29– 135).

López-Colino, F & Colas, J. (2012) Spanish Sign Language synthesis system. Journal of visual languages and computing 23 121-136

Marshall, E and. Sáfár, I (2003) A prototype text to British Sign Language (BSL) translation system. In 41st Annual Meeting of the Association of Computational Linguistics, Sapporo, Japan, pp. 113–116.

Pollard, C and Sag, I (1994) Head-Driven Phrase Structure Grammar, Chicago: University of Chicago Press

Prilllwitz S ., Leven, R ., Zienert, H ., Hanke, T and Henning, J (1989) Hamburg Notation System for Sign Languages: an introductory guide - HamNoSys version 2.0., Hamburg (D), Signum Press

Shieber, S (1994) Restricting the Weak-Generative Capability of Synchronous Tree Adjoining Grammars. Computational Intelligence, 10(4)

Zhao, L., Wiliam, K ., Schuler, C ., Vogler, N ., and. Palmer, M (2000). ìA Machine Translation System from English to American Sign Language. î Association for Machine Translation in the Americas

Construction of the *Corpus of Everyday Japanese Conversation*: An Interim Report

Hanae Koiso[†], Yasuharu Den[‡,†], Yuriko Iseki[†], Wakako Kashino[†], Yoshiko Kawabata[†], Ken'ya Nishikawa[†], Yayoi Tanaka[†], Yasuyuki Usuda[†]

[†] National Institute for Japanese Language and Linguistics
10-2 Midoricho, Tachikawa, Tokyo 190-8561, Japan
{koiso, iseki, waka, kawabata, nishikawa, yayoi, usuda}@ninjal.ac.jp

[‡] Graduate School of Humanities, Chiba University
1-33 Yayoicho, Inage-ku, Chiba 263-8522, Japan
den@chiba-u.jp

Abstract

In 2016, we launched a new corpus project in which we are building a large-scale corpus of everyday Japanese conversation in a balanced manner, aiming at exploring characteristics of conversations in contemporary Japanese through multiple approaches. The corpus targets various kinds of naturally occurring conversations in daily situations, such as conversations during dinner with the family at home, meetings with colleagues at work, and conversations while driving. In this paper, we first introduce an overview of the corpus, including corpus size, conversation variations, recording methods, structure of the corpus, and annotations to be included in the corpus. Next, we report on the current stage of the development of the corpus and legal and ethical issues discussed so far. Then we present some results of the preliminary evaluation of the data being collected. We focus on whether or not the 94 hours of conversations collected so far vary in a balanced manner by reference to the survey results of everyday conversational behavior that we conducted previously to build an empirical foundation for the corpus design. We will publish the whole corpus in 2022, consisting of more than 200 hours of recordings.

Keywords: Corpus of everyday Japanese conversation, corpus design, legal and ethical issues, corpus evaluation

1. Introduction

Everyday conversation is the most basic form of human communication. In order to understand our diverse and situated interactional behavior, it is needed to collect and analyze various kinds of conversations in our daily life. Although several corpora of Japanese conversations have been developed, most of them are biased in terms of speaker attributes and situations, mainly targeting conversations in experimental settings, such as map task dialogs, and artificial situations, such as chats among university students recruited for recording purposes. There are few corpora of Japanese conversations that covers real situations in daily life.[1]

In 2016, we launched a new corpus project, in which we are building a large-scale corpus of everyday Japanese conversation, the *Corpus of Everyday Japanese Conversation*, CEJC. The main features of the CEJC are i) that we target conversations embedded in naturally occurring activities in daily life, without the exogenous intervention of researchers imposing topics or displacing the context of action (Mondada, 2012); ii) that we collect various kinds of everyday conversations in a balanced manner so as to capture the diversity of everyday conversations and to observe natural conversational behavior in our daily life; and iii) that we collect and publish not only audio but also video data in order to precisely understand the mechanism of our real-life social behavior.

In this paper, we first introduce an overview of the corpus, including corpus size, conversation variations, recording methods, structure of the corpus, and annotations to be included in the corpus. Next, we report on the current stage of the development of the corpus development and legal and ethical issues discussed so far. Then we present some results on the preliminary evaluation of the data being collected.

[1] For corpora of other languages that cover everyday situations, see e.g., Burnard and Aston (1998) and Nelleke (2000).

2. Corpus Design

2.1. Corpus size

We plan to publish more than 200 hours of conversations. Based on the data we have recorded and transcribed so far, the total number of words, conversations, and conversants are estimated at 2.1 million words (short-unit words, see below), 400 conversations, and a total of 1200 conversants, including 600 different participants.

2.2. Conversation variation

The CEJC will contain various kinds of everyday conversations in a balanced manner. To estimate distributions of various conversational attributes in our daily life, we conducted a survey of everyday conversational behavior with about 250 Japanese adult informants (Koiso et al., 2016b). The questionnaire included when, where, how long, with whom, and during what kind of activity informants were engaged in conversations in their daily life. Based on the results, we derived rough distributions of conversation forms, conversation places, and accompanying activities as a measure of the design of a balanced corpus. The survey result will be compared with the conversation data collected so far in Section 3.



Figure 1: Video images of a conversation between husband and wife while cooking at home. The left image was recorded by a Kodak PIXPRO SP360 4K camera located on the table, while the top- and bottom-right images were recorded by two GoPro cameras placed facing each other on the bookshelf and the sideboard. As for speech, the two conversants wear IC recorders (SONY ICD–SX734), and their voices were recorded with their own recorders. All conversants' voices were also recorded by another IC recorder located on the center of the table. Due to the restriction stated in the consent form, the faces of the participants are airbrushed for the protection of personal information in a printed material, although they are left intact in the video data to be published.

2.3. Recording method

In order to record various kinds of naturally occurring conversations in daily situations, we employ two methods, *individual-based* and *situation-specific* methods (Koiso et al., 2016a).

Individual-based method In this method, we recruit 40 informants balanced in terms of sex and age (man/woman $\times$ 20s/30s/40s/50s/over 60 $\times$ 4 informants), provide them with portable recording devices (compact action cameras and IC recorders) for approximately two to three months, and have them record about 15 hours of conversations in their daily activities. The informant him/herself carries portable recording devices and records his/her everyday activities in a variety of situations such as at home, at a restaurant, and outdoors. In principle, the project members do not mediate their field recordings. We developed the individual-based method by referring to the approach adopted for the demographically sampled part of the British National Corpus (Crowdy, 1995; Burnard and Aston, 1998). Figure 1 shows video images of a conversation between husband and wife while cooking at home.

About four to five hours of conversations, among 15 hours, per informant, i.e., a total of about 180 hours, are selected for the CEJC by taking into account the balance of conversation variations, quality of recorded data, and legal and ethical issues.

The informant also has to i) judge, for instance, whether recording is permitted where they are conversing, and get permission if necessary, ii) explain the purpose of the recording to other conversants, iii) obtain their consents to publish the recorded conversation, including video data, iv) have them fill in informant sheets including their date of birth, residence, birthplace, sex, occupation, and relationship to the informant, and v) note the recording date and time, an overview of the conversation, and the layout of the conversants and the recording devices.

Situation-specific method In addition to the individual-based method, we also use the situation-specific method to compensate for a lack, or shortage, of recordings in institutional settings, e.g., meetings at workplaces and exchanges with store employees, for which recording based on the individual-based method is technically and/or ethically difficult. In this method, we select specific situations and the recording staff set up a recording environment. Although the project members coordinate recording settings, only conversations in these naturally occurring activities are recorded. The size and types of conversations collected based on this method will be decided by referring to data collection status based on the individual-based method.

2.4. Structure of the CEJC

Figure 2 shows the layered structure of the CEJC. About 600 to 800 hours of conversations will be recorded, and

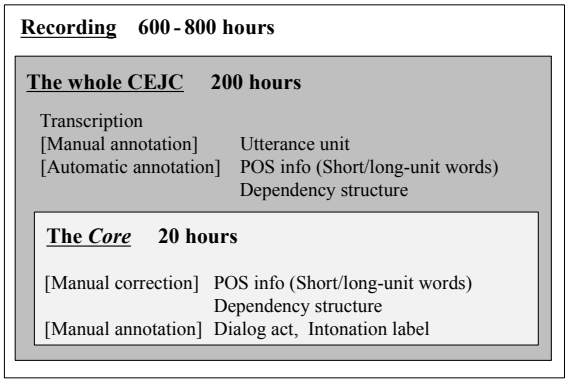


Figure 2: Layered structure of the CEJC

speakerID	startTime	endTime	text	note
IC01	2502.617	2503.920	(U Kono mae) nomikai doko de non da no.	(U xx) : transcription of questionable or inaudible talk
			Last time, where did you drink?	.: boundary of an utterance unit
IC03	2504.661	2505.651	Etto Akasaka.	
			Um, in the Akasaka area.	
IC04	2507.718	2508.495	Akasaka no	
			In Akasaka,	
IC03	2508.791	2509.744	(L)	(L) : laughter
IC04	2509.287	2510.202	ryootei.	
			at a fine dining restaurant?	
IC03	2510.912	2511.480	(L Iya iya).	(L xx) : speech while laughing
			No, no.	
IC01	2511.432	2512.185	Chigau chigau.	
			Different, different.	
IC01	2512.749	2513.451	Izakaya.	
			At a casual restaurant.	
IC03	2513.641	2514.236	(W Isakaya|Izakaya).	(W xx|yy) : 'xx' reduced or incorrect pronunciation
			At a casual restaurant.	'yy' supposed-to-be correct word
IC03	2515.464	2516.201	(U Futaherumo).	
			Futaherumo.	
IC03	2516.999	2519.648	Dooki no (D hi)(D fu)	(D xx): word fragment
			dooki to futari de non da gurai de.	
			I had a drink with the same-age, hi, fu,	
			a same-age peer.	

Figure 3: Example of transcript. In the actual transcript, texts are written in Japanese characters, and the boundary of an utterance unit is marked by the 'ideographic full stop.'

among them about 200 hours will be selected for the corpus. The whole corpus contains video and audio data, transcript, and four kinds of annotations to be described in § 2.6., three of which are automatically labeled. There is a subset of the corpus, named the *Core* data set, which consists of 20 hours of conversations, corresponding to 10% of the whole corpus, which includes six kinds of manually labeled, or corrected, annotations.

2.5. Transcription

The speech is divided into transcription units at the locations of perceptible pauses and the boundaries of utterance units (see below). Each unit is orthographically transcribed by hand with reference to video and audio data using ELAN,[2] and about 20 kinds of tags, which are defined in reference to the transcription criteria and conventions previously used in the *Corpus of Spontaneous Japanese* (CSJ) (Maekawa, 2004) and in the *Chiba Three-party Conversation Corpus* (Den and Enomoto, 2007), are inserted in the text. Figure 3 shows a sample transcript.

2.6. Annotation

In addition to transcripts, the following annotations are created:

Utterance Unit Utterance units are manually identified based on long utterance-units (Den et al., 2010), which are regarded as a basic unit for interaction and determined considering syntactic, pragmatic, and interactional aspects. The periods in the sample transcript in Figure 3 indicate utterance unit boundaries.

[2] https://tla.mpi.nl/tools/tla-tools/elan/

Two types of POS information Two different POS systems, short-unit word (SUW) and long-unit word (LUW), are adopted. Most SUWs are mono-morphemic words or words made up of two morphemes, while LUWs are multi-morphemic words including compound words like compound nouns, compound verbs, and compound particles. All the data are automatically analyzed, and those in the *Core* are manually corrected.

As for SUWs, the data are analyzed using UniDic, a dictionary developed for the POS annotation of the *Balanced Corpus of Contemporary Written Japanese* (Maekawa et al., 2014). The audio data are also automatically segmented into SUWs by means of forced alignment against morphologically-segmented texts, and those in the *Core* are manually corrected.

Dependency structure Dependency structures between *bunsetsu* phrases, which are comprised of content words possibly followed by one or more function words, are automatically labeled within utterance units, and those in the *Core* are manually corrected.

Dialog act The *Core* also contains dialog acts manually annotated according to an ISO-standard-based (ISO 24617-2, 2012) scheme extended to cover various kinds of sequence organizations observed in everyday conversation.

Intonation label Part of the *Core*, which is selected based on recording conditions and degrees of dialect, is manually labeled according to a simplified version of the intonation-labeling scheme, X–JToBI (Maekawa et al., 2002), which was developed for the CSJ.

Table 1: Attributes of informants (As of Jan. 15th, 2018)

Age	Sex		Total
	male	female	
20s	student*2 student*2 self-employed	student*2 student*2 office worker*1	6
30s	civil servant*2 self-employed*2 freelance	housewife*2 office worker*2 office worker self-employed	7
40s	office worker*2 freelance*2 office worker*1 teacher*1	office worker*2 office worker*2 part-time*2 freelance	8
50s	office worker*2 manager*2 self-employed	self-employed*2 office worker*2 office worker*1	6
over 60	volunteer*2 teacher*2	volunteer*2 office worker self-employed freelance*1	6
Total	15	18	33

*1: in the process of recording
*2: finished the data selection for the CEJC

3. Current stage of the corpus development

As of January 15th, 2018, 28 informants have finished recording and five are in the process of recording. Table 1 shows the attributes of these informants. In the cases of 20 out of the 28 informants who completed recording, we have selected conversations to be compiled into the CEJC. The selected data contains about 94 hours, corresponding to 47% of the whole, 210 conversations, and a total of 783 conversants, including 424 different participants.

4. Legal and ethical issues

A notable characteristic of the CEJC is that not only audio but also video data are collected and published. There have, however, been virtually no corpora that contain video recordings of everyday conversations, and guidelines on the release of such data have not been established. Based on a variety of data collected so far, we are discussing, with a lawyer specializing in copyright and portrait-right issues, how to deal with legal and ethical problems from the aspect of portrait-right, copyright, and the protection of personal information.

The video data often contains i) the faces of third parties who have not consented to publish their faces and ii) copyrighted works, such as TV programs and books.

When the faces of third parties are inside the scope of protection of portrait rights, those parts are airbrushed by means of an image effect. The faces of people performing common activities, not sensitive activities, in public places are regarded as outside the scope of protection of portrait rights, provided that the recordings will be used for research purposes and that their faces in themselves will not be the target of the research. When a short exchange between a third partiy and conversants who have agreed to have their faces published is transcribed, the faces of the third party is concealed (See Figure 4).



Figure 4: Video image which includes a face of a waitress talking with conversants at a restaurant. Although exchanges between the waitress and the main conversants are transcribed, the face of the waitress is concealed.



Figure 5: Video image which includes a television program. TV screen is not concealed.

If the use of copyrighted works included in the video data can be interpreted as incidental, i.e., an unexpected appearance as described in the copyright provisions for 'Disclosure of Photo or Image in which Copyrighted Work Appears,' they are not concealed (See Figure 5).

Personal information including conversants' names, affiliations, and individual identification information, as well as any parts of recordings for which conversants have not given their permission for publication are replaced by anonyms or turned letters in transcripts, and the corresponding regions of the audio files are made inaudible.

5. Preliminary evaluation of CEJC

In this section, using the 94 hours of conversations that have been compiled into the CEJC, we give a preliminary evaluation of the issue of balanced by reference to the survey results of everyday conversational behavior described in Koiso et al. (2016b).

The distributions of forms, places, activities, and numbers of conversants in the current data set, as well as the survey results, are shown in Figure 6.

As for the conversation form, slight differences are seen in that the ratio of chats in the current data set is about 11% higher than that in the survey result, while the ratio of business talks/consultation is 11% lower. Overall, however,

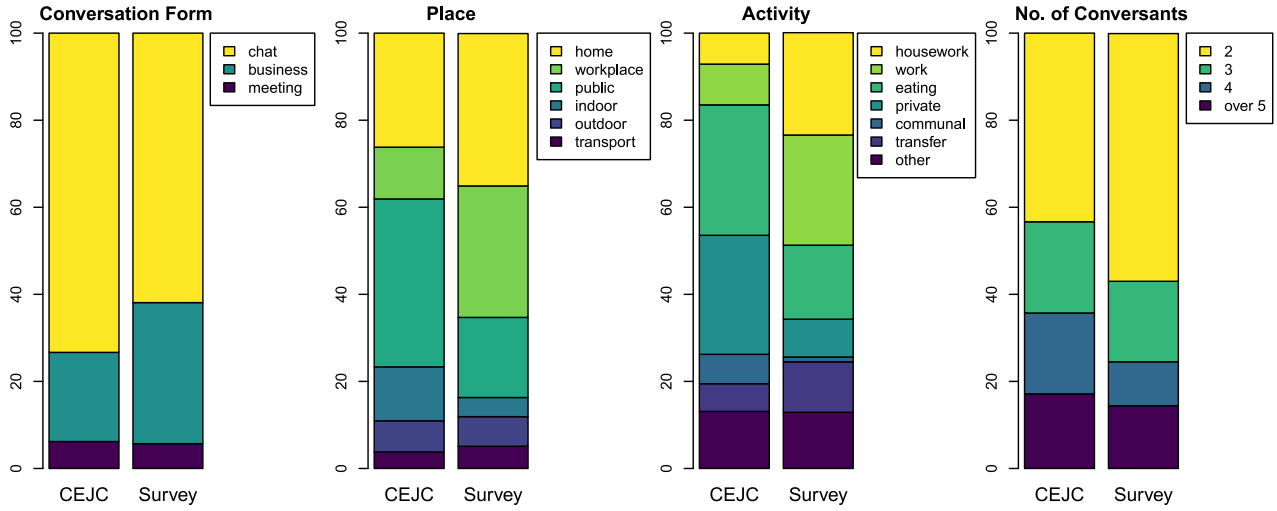


Figure 6: Distributions of conversation forms, places, activities, and numbers of conversants in the current data set and the survey results of conversational behavior

the current data set well varies in a balanced manner with reference to the survey results.

The same can be said for the number of conversants. In Figure 6, the ratio of dyadic (two-party) conversations in the current data set is about 14% lower than that in the survey result, while the ratio of conversations with more than three people is about 11% higher. Such a slight bias toward conversations with larger groups was intentionally introduced upon data selection in order to cover various kinds of conversations.

The distributions of places and activities show different tendencies between the current data set and the survey results. For example, the current data set contains more conversations at public/commercial facilities, such as restaurants and city halls, but fewer conversations at home, at school, or in the workplace than the survey results. In regard to activity, the current data set includes more conversations during leisure/communal activities and when spending time with friends but fewer conversations during housework, work, and schoolwork than the survey results.

The main reason why the current data set contains few conversations during work/schoolwork at workplaces/schools is that it is difficult to record such conversations based on the individual-based method. In the future, it will be necessary to reinforce such types of conversations based on the situation-specific method.

The current data set has considerably fewer conversations at home than the survey results, even though informants may have many opportunities for recording conversations at home. This is due to a bias in our sampling criteria. If we choose as many conversations at home as in the survey results, only similar types of conversations, such as conversations during dinner with the family at home, will be included in the corpus. We would rather select conversations with the family that were conducted outside, such as those in public/commercial facilities and at relatives' houses. This bias results in a decrease in the relative frequency of conversations at home.

Figure 7 shows the distributions of ages, sexes, and occupations of a total of 783 conversants, including 424 different participants involved in the current data set. It is found that conversants are balanced in terms of sex. By contrast, the figure shows that children under 20 years old, from elementary school students to high-school students, account only for 5 to 7% of the data and there are no high-school students at all. Since the individual-based recording method places a heavy responsibility on principal informants, such as dealing with various types of personal information, children under 20 are not recruited as informants. Children may participate in conversations when the principal informant invites them, but the possibility of recording conversations involving children depends highly on the composition of the informant's family. To solve this problem, we will select more informants who have children in their families.

6. Conclusions

In this paper, we first introduced an overview of the CEJC, including corpus size, conversation variations, recording methods, structure of the corpus, and annotations to be included in the corpus. Then we reported on the current stage of the corpus development and legal and ethical issues encountered so far. We also presented a preliminary evaluation of the data collected so far.

We focused on whether or not the 94 hours of conversations collected so far varies in a balanced manner by reference to the survey results of everyday conversational behavior. As for the conversation form and the number of conversants, the current data set varies in a balanced manner by reference to the survey results. By contrast, the current data set contains i) fewer conversations during work/schoolwork at workplaces/schools than the survey results, due to difficulty in recording such conversations using the individual-based method, ii) fewer conversations at home than the survey results, due to a bias in our sampling criteria, and iii) few conversations involving children under 20, due to the age restriction on informants. We will adopt the situation-specific method so as to compensate for these biases in the collected data.

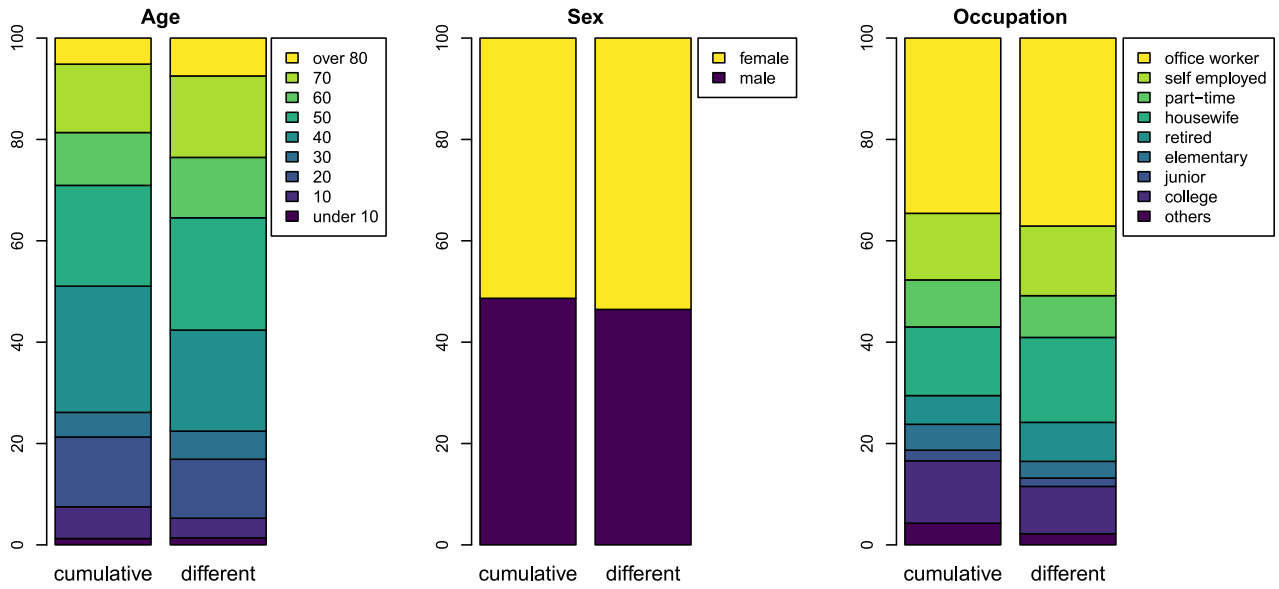


Figure 7: Distributions of ages, sexes, and occupations of a cumulative total of 783 conversants, including 424 different participants involved in the current data set

We plan to publish a part of the CEJC, about 50 hours, on a trial basis in 2018, and the entirety in 2022.

7. Acknowledgments

The work reported in this article is supported by the NINJAL collaborative research project "A Multifaceted Study of Spoken Language Using a Large-scale Corpus of Everyday Japanese Conversation".

8. Bibliographical References

Burnard, L. and Aston, G. (1998). *The BNC handbook*. Edinburgh University Press, Edinburgh, U.K.

Crowdy, S. (1995). The BNC spoken corpus. In G. Leech, G. Myers, and J. Thomas, editors, *Spoken English on computer: Transcription, mark-up and application*, pages 224–235. Longman, Harlow, U.K.

Den, Y. and Enomoto, M. (2007). A scientific approach to conversational informatics: Description, analysis, and modeling of human conversation. In T. Nishida, editor, *Conversational informatics: An engineering approach*, pages 307–330. John Wiley & Sons, Hoboken, NJ.

Den, Y., Koiso, H., Maruyama, T., Maekawa, K., Takanashi, K., Enomoto, M., and Yoshida, N. (2010). Two-level annotation of utterance-units in Japanese dialogs: An empirically emerged scheme. In *Proceedings of LREC 2010*, pages 2103–2110, Valletta, Malta.

ISO 24617-2. (2012). Language resource management — semantic annotation framework (SemAF) — Part 2: Dialogue acts.

Koiso, H., Tanaka, Y., Watanabe, R., and Den, Y. (2016a). A large-scale corpus of everyday Japanese conversation: On methodology for recording naturally occurring conversations. In *Proceedings of LREC 2016 Workshop: Just talking — Casual talk among humans and machines*, pages 9–12, Portoroz, Slovenia.

Koiso, H., Tsuchiya, T., Watanabe, R., Yokomori, D., Aizawa, M., and Den, Y. (2016b). Survey of conversational behavior: Towards the design of a balanced corpus of everyday Japanese conversation. In *Proceedings of LREC 2016*, pages 4434–4439, Portoroz, Slovenia.

Maekawa, K., Kikuchi, H., Igarashi, Y., and Venditti, J. J. (2002). X–JToBI: An extended J_ToBI for spontaneous speech. In *Proceedings of INTERSPEECH 2002*, pages 1545–1548, Denver, CO.

Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M., and Den, Y. (2014). Balanced corpus of contemporary written Japanese. *Language Resources and Evaluation*, 48(2):345–371.

Maekawa, K. (2004). Design, compilation, and some preliminary analyses of the *Corpus of Spontaneous Japanese*. In K. Yoneyama and K. Maekawa, editors, *Spontaneous speech: Data and analysis*, pages 87–108. The National Institute for Japanese Language and Linguistics, Tokyo.

Mondada, L. (2012). The conversation analytic approach to data collection. In J. Sidnell and T. Stivers, editors, *The handbook of conversation analysis*, pages 32–56. Wiley-Blackwell, Hoboken, NJ.

Nelleke, O. (2000). Building a corpus of spoken Dutch. In P. Monachesi, editor, *Computational linguistics in the Netherlands 1999: Selected papers from the 10th CLIN meeting*, pages 147–158. Utrecht Institute of Linguistics OTS.

Carcinologic Speech Severity Index Project: A Database of Speech Disorder Productions to Assess Quality of Life Related to Speech After Cancer

Corine Astésano[1], Mathieu Balaguer[2], Jérôme Farinas[1], Corinne Fredouille[3], Pascal Gaillard[1], Alain Ghio[4], Laurence Giusti[4], Imed Laaridh[3], Muriel Lalain[4], Benoît Lepage[1], Julie Mauclair[1,5], Olivier Nocaudie[1], Julien Pinquier[1], Oriol Pont[1], Gilles Pouchoulin[4], Michèle Puech[2], Danièle Robert[4], Etienne Sicard[1], Virginie Woisard[2]

[1]Toulouse University, [2]Toulouse Hospital, [3]University of Avignon, LIA,[4]Aix-Marseille Univ, CNRS, LPL, [5]Paris Descartes University

Toulouse France, Toulouse France, Avignon France, Aix-en-Provence France, Paris France

c2si@irit.fr

Abstract

Within the framework of the Carcinologic Speech Severity Index (C2SI) InCA Project, we collected a large database of French speech recordings aiming at validating Disorder Severity Indexes. Such a database will be useful for measuring the impact of oral and pharyngeal cavity cancer on speech production. That will permit to assess patients' Quality of Life after treatment. The database is composed of audio recordings from 135 speakers and associated metadata. Several intelligibility and comprehensibility levels of speech functions have been evaluated. Acoustics and Prosody have been assessed. Perceptual evaluation rates from both naive and expert juries are being produced. Automatic analyzes are being carried out. That will provide to speech therapists objective tools to take into account the intelligibility and comprehensibility of patients which received cancer treatment (surgery and/or radiotherapy and/or chemotherapy). The aim of this paper is to justify the need of this corpus and his data collection. This corpus will be available to the scientific community through the GIS Parolotheque.

Keywords: speech intelligibility and comprehensibility, quality of life assessment, speech corpus, pathological speech

1. Introduction

The decreasing mortality in cancerology highlights the importance to reduce the impact on the Quality of Life (QoL) after cancer. That particularly concerns Head and Neck Cancers (HNC), because their treatment can be mutilating and disabling.

However, the usual tools for assessing QoL are not relevant for measuring the impact of the treatment on the main functions involved by the sequelae. And, there is a clear lack of uniform methods for assessing functional outcomes.

Measuring the impact on one or several of the most altered functions after the therapeutic care of a given tumoral localization, would allow:

1. to complete the expression of the therapeutic outcomes by functional forecast index,
2. to adjust the treatment for reducing their functional consequences.

For the HNC, it is mainly about impacts of (oral) communication and feeding (swallowing) (Mlynarek AM et al 2008). QoL research has, at times, failed to provide health care professionals with clinically relevant and interpretable information that can guide treatment decisions. This has led researchers to attempt to make commonly used research tools more accessible to the clinicians. Health-related Quality of Life (HRQoL) reflecting the effect of disease and disease treatment on general well being (Cardol M et al 1999) evolved to the creation of handicap questionnaires and specific related QoL questionnaires for numerous chronic diseases with a rise in importance of specific and symptom modules. But validated tools to measure the functional outcomes of carcinologic treatment are still missing, in particular for speech disorders. Some assessments are available for voice disorders in laryngeal cancer but they are based on very poor tools for oral and pharyngeal cancers involving more the articulation of speech than voice. Because the usual tools to assess QoL are not relevant to measure the impact of the treatment on the main functions involved by the sequelae, and because it is acknowledged that an unbiased and objective assessment of the communication deficiency caused by a speech disorder calls for automatic speech processing tool, we proposed to develop a severity index of speech disorders describing the outcomes of therapeutic protocols completing the survival rates. The principle is to perform an audio recording of the patient's speech and to compute the intelligibility of the utterances produced in the aim to get a score. Middag in 2012 presented a new method that predicts running speech intelligibility in a robust way against changes in the text and against differences in the accent of the Dutch speakers applicable to patients treated for HNC.

Therefore, our hypothesis is that an automatic assessment technique can measure the impact of the speech disorders on the communication abilities giving a severity index of speech in patients treated for HNC and particularly for oral and pharyngeal cancers. We will name this index the Carcinologic Speech Severity Index (C2SI). Speech intelligibility is the usual way to quantify the severity of neurologic speech disorders. But this measure is not valid in clinical practice because of several difficulties as the familiarity effect of this kind of speech and the poor inter-judge reproducibility. Moreover, the scores do not accurately reflect listener comprehension.

In order to develop and evaluate this C2SI, a project has been contracted with French National Cancer Institute (Grant InCA SHS n°2014-135) in 2014 and CHU Toulouse, LPL Aix-En-Provence laboratory, PETRA MSH Toulouse, Octogone-Lordat Toulouse, LIA University of Avignon, Paul Sabatier Toulouse University and IRIT Toulouse laboratory, which form an interdisciplinary team. This C2SI project aims to create a speech corpus in order to validate the assumptions of the speech severity index. The corpus is presented in this paper. The structure and the list of tasks performed by each speaker are presented in section 2. Section 3

presents the available material, and some statistics on this corpus are reported in section 4.

2. Method description

The corpus associates audio recordings and QoL questionnaires. The content was chosen to assess a broad spectrum of intelligibility linked to quality of life: from acoustic to understandability.

2.1 Self Assessment questionnaires

Self assessment questionnaires are used in practice to evaluate QoL in its several dimensions.
The main generic quality of life questionnaire is the MOS-SF36 (Wade and Sherbourne 1992). It is validated in all kind of illnesses and explores physical and mental health. In the case of Head and Neck Cancer, the cancer specific HRQoL questionnaire used frequently in Europe is the European Organization for Research into the Treatment of Cancer (EORTC QLQ-C30) with its complementary module assessing HNC specific problems and symptoms respectively QLQ-H&N35. These questionnaires, as generic or specific, give independent information.
Self-questionnaire of Handicap was proposed for various functions of the upper aerodigestive tract (UADT). The Speech Handicap Index (SHI) for speech (Rinkel 2008) is validated for HNC. The Phonation Handicap Index (PHI) is a French similar tool but validated for all kind of speech production disorders (Fichaux-Bourin et al 2009).
The relations between the questionnaires of QoL and the questionnaires of handicap were often analyzed, the quality of life being of use to the validation of contents of the questionnaires of handicap. Strong correlations (0.7 to 0.9) were computed between the SHI and the speech domain of QoL questionnaire. This correlation is lower, if not absent, according to the other domains (Borggreven et al 2007, Dwivedi et al 2011, Thomas et al 2009).
Because the use of a handicap questionnaire targeting a function is well correlated to the domains of the questionnaires of QoL in relation with, we selected the generic QoL questionnaire (SF36) and the specific speech related Handicap questionnaire (SHI and PHI) to integrate the communication dimension.

2.2 Intelligibility assessment

To cover the several aspects of intelligibility, different tasks were performed by the speakers (both controls and patients):

Sustained vowel /a/ (AAA): A sustained vowel gives information about the voice level, phonation time, stability, harmonics contents, noise, unvoiced segments, etc. Despite the weak correlation between voice production and intelligibility of speech, the capacity to hold a vowel more than 5 seconds is a minimal condition for speech production. Recording this production is a global measure of the acoustic/aerodynamic balance for speech and may contribute to acoustics analysis.

Acoustico Phonetic Decoding (DAP): The limitations of intelligibility tests performed on speakers with speech production disorders lie in the ability of listeners to restore distorted sequences. This effect is emphasized when the auditors have a strong knowledge of the words used in the test and if these words are unambiguous and therefore strongly predictable (Enderby, 1983, 2008). This is generally the case for speech therapists who can make such an extensive use of these lists that they eventually know them by heart. The bias associated with this knowledge and therefore with the strong influence of the top-down perceptual mechanisms results in an overvalued intelligibility score because the phonemic restoration of the listener makes opaque the distortions of production (Warren et al., 1970; Samuel, 1981)
The solution we have adopted consists in using pseudo-words, complying with the frequent phonotactic structures in French, in large quantities so as to completely neutralize the effects of lexicality, familiarization and learning of the items by the listeners (Ghio et al, 2016).

Image Description (DES) and Spontaneous Speech (SPO): In real life, the top-down effect is present. This is why the spontaneous speech remains often used for assessing intelligibility (Woisard et al., 2013). In order to reduce the predictability of the speech produced, we recorded patients/controls describing a picture and telling their comments about a text which they read before.

Reading a Short Text (LEC): Using the same text is in complete opposition with the previous tasks but is interesting for the comparison of acoustic analysis and the automatic intelligibility scoring. This makes it possible to produce automatic phonetic alignments, even if the speech production is very altered. Speech rate, prosody, consonant and vowel precision, pauses and other speech features may be easily extracted and compared between the normal and patient groups.

2.3 Prosody assessment

Prosody helps structuring different levels of linguistic information, be it lexical, syntactic, semantic or pragmatic. The patients we focus on in this project have undergone treatment at the supraglottic level of their anatomy (glossectomy, mandibulectomy for example). Hence, the source was not affected. However, we hypothesize that compensatory mechanisms at the segmental level will impact prosodic characteristics, particularly affecting prosodic fluency. Our prosodic tasks are designed to evaluate which structural functions of prosody are most affected by these types of cancer.

Modal Prosody Function (MOD): Classically, clinical investigations of speech pathologies involve assessing the modal and emotional prosody functions, although speech intelligibility/comprehensibility is more related to structural functions, which are never tested.

Focus (FOC) and Disambiguation Syntactic (SYN) Prosody: these tasks are taken from (Aura, 2012), who adapted (Magne et al, 2005) and (Astésano et al, 2007) for clinical use. The modality task consists in producing ten identical sentences with 3 different modalities: assertion, question and injunction (*Tu manges les pâtes* ?/ . /! *eg. You eat pastas* ?/ ./!). In the focus task (Aura, 2012), speakers had to resolve a paradigmatic opposition (contrastive focus) between two words given in an auditorily presented sentence so as to prosodically highlight the relevant word ("*Tu as vu un canard ou un cochon dans le jardin?*" *eg. "You saw a duck or a pig in the garden?"* with the written answer: "*j'ai vu un CANARD dans le jardin*" *eg. "I saw a DUCK in the garden"*). The syntactic task (Aura, 2012; Astésano *et al*, 2007) consists of similar written scripts that only prosody

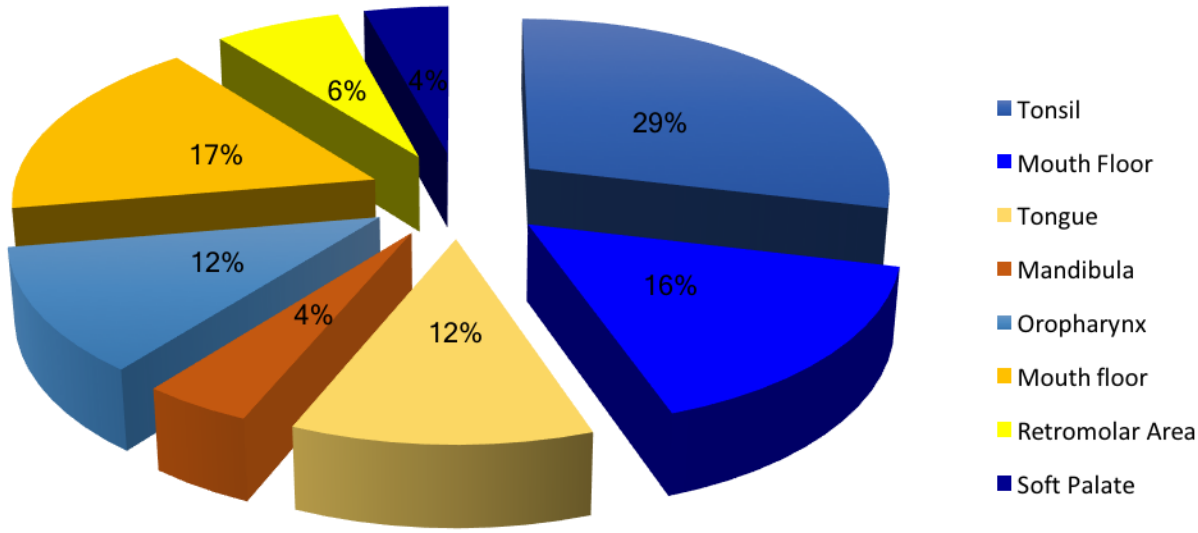


Figure 1: Tumor localisation distribution.

can disambiguate. For example, in the sentence "*les chevaux et les poneys blancs*"(eg. *"White horses and poneys"* but note that the adjective in French is at the end of the sentence), the adjective "*blancs" (eg. white)* can either apply to the second noun only (narrow scope) or to the two nouns (broad scope): prosodic cues such as final lengthening, pause and *f0* excursions can give the proper syntactic parsing (either *les chevaux// et les poneys blancs* or *les chevaux et les poneys // blancs)*.
The capacity of the speakers to properly use prosodic cues in these different tasks is then intended to be evaluated through perception tests on naive, healthy listeners (Nocaudie et al., 2017)

2.4 Comprehensibility assessment

In order to evaluate the comprehension of speech, it is important to go beyond the simple tests on isolated words.

We introduce **Sentence Verification Tasks (SVT)** in order to assess the global comprehension of running speech. In this task, speakers read a set of sentences. The semantic content of each sentence can be true (ex: "january is a winter month") or false (ex: "january is a summer month"). In the perception evaluation, participants are presented a variety of utterances across several knowledge domains and have to decide as fast as possible if these statements are true or false (Pisoni et al., 1987). The accuracy score and the response time are used as a couple of indicators of the comprehension process. Indeed, when auditors need to understand the linguistic content of a message and perform an appropriate response [True or False], the quality of the acoustic-phonetic information of the speech signal plays an important role both in the speed and accuracy of the answer provided.

A simple recording of spontaneous speech can also be interesting to assess the comprehensibility of a text. But the evaluation of an index based on these recordings is not easy as the semantic sense may be very varied. But this could be analysed in order to confirm the other indexes during perceptual analyzes.

3. Corpus description

3.1 Population

We expect a correlation between an automatic index and the perceived index given by the jury to be as high as 0.86 correlation that was achieved in University of Ghent work (Middag et al, 2008, Middag et al, 2009 and Middag, 2012). The size of the sample influences the precision of this estimation, a bigger sample bringing a bigger precision (characterized by a narrower reliable interval). To obtain a reliable interval in 95% the width of which is not superior to 0.15 around a coefficient of 0.8, it is necessary to recruit 94 patients. In september 2017, we have recorded 94 patients and 41 control speakers. That is superior to the corpus used in (Middag, 2012), which contained recordings and perceptual evaluations of 55 patients with advanced Head and Neck Cancer who were treated with concomitant chemoradiotherapy.

The 94 patients are recruited in the three main departments of Toulouse managing patients with HNC (ENT department of the Universitary Hospital, Cancerology department of the Institut Claudius Regaud (surgery and radiotherapy), Maxillofacial surgery department of the Universitary Hospital of Toulouse).

They are selected from the lists of carcinologic follow-up consultations of these 3 departments. These departments are participating within the University Institute of cancer in Toulouse (IUC-T) and will be associated with the unit of Onco-réhabilitation which is located at the IUC-T Oncopole.

3.2 Questionnaires

The SHI and PHI questionnaires presented in 2.1 are given to the patients just before the audio recordings.

3.3 Recordings

The speakers were settled in a comfortable way in an anechoic room in front of a computer. This computer was used to visually display instructions and corpus. For some tasks, the instructions were also produced with an

auditory modality (ex: pseudo-words in DAP task). The recordings were made with a Neumann TLM 102 Cardioid Condenser Microphone connected to a FOSTEX digital recorder. The sampling rate was 48 kHz, which facilitates the downsampling to 16 kHz, usually used in automatic speech processing.

The corpus is composed of subpart collections described below. The passation order is: AAA, LEC, DES, SPO. And then the prosodic tasks (MOD, FOC, SYN) or the intelligibility and comprehension ones (DAP, SVT).

3.3.1 AAA

This recording consists in the production of sustained /a/ held at 3 occasions. A lot of analyses are done by speech therapist with this kind of recordings so it was important for us to include them. Indeed, the analysis of vowel /a/ can bring important cues on stability of formants and how the person deals with the breath.

3.3.2 LEC

The reading of the 1st paragraph of "La chèvre de M. Seguin", a tale by Alphonse Daudet, is performed by the speaker. This text has been chosen because it is long enough and it includes all the French phonemes. It is also well known and widespread in clinical phonetics in France (Ghio et al., 2012).
Here is the full plain text: "*Monsieur Seguin n'avait jamais eu de bonheur avec ses chèvres. Il les perdait toutes de la même façon. Un beau matin, elles cassaient leur corde, s'en allaient dans la montagne, et là-haut le loup les mangeait. Ni les caresses de leur maître ni la peur du loup rien ne les retenait. C'était paraît-il des chèvres indépendantes voulant à tout prix le grand air et la liberté.*"

3.3.3 DES

The subject was asked to choose one among several pictures that represent the same field (sea with boats). Each subject had to describe the picture to the examiner so that the latter can redraw it just on the basis of the oral explanations.

3.3.4 SPO

The patient must give his/her opinion on the questionnaire that he/she has to fill out before the recording session. He/she must speak for at least 3 minutes. This task permits to collect spontaneous speech recordings with no constraint on the sentences.

3.3.5 MOD

Each speaker recorded 10 different scripts uttered with 3 modalities: assertion, question and injunction. Each script was presented on a computer screen, with the expected prosodic modality indicated by either of the 3 punctuation marks ('.' ; ' ?' ; ' !').

3.3.6 FOC

Each speaker recorded the same set of 20 sentences, for which they had to produce the proper focus on the scripted sentence, following the audio presentation of a question. For example, after listening to the question '*What did you see in the garden? a duck or a pig?*', they had to read the following sentence *'I saw a duck in the garden'*, with contrastive focus on DUCK.

3.3.7 SYN

Each speaker recorded 13 scripts with two syntactic conditions (narrow vs. broad scope of adjective). The sentences were written on a computer screen, with the expected syntactic grouping indicated visually by vertical bars.

3.3.8 DAP

After two training trials, each speaker had to pronounce 50 pseudo-words. The pseudo-words have the phonotactic structure of the C(C)1V1C(C)2V2 type

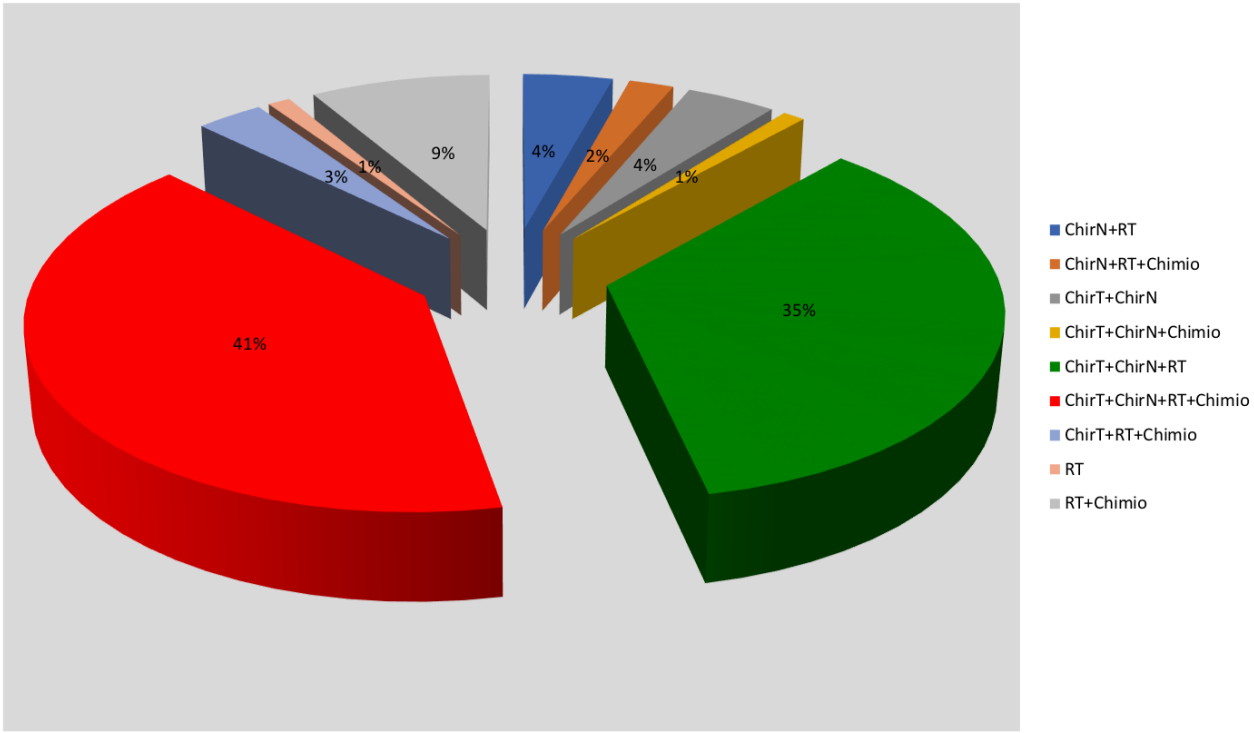


Figure 2: Patients treatment distribution.

where C(C)i is an isolated consonant or a consonant group. Such a combinatorial method makes it possible to generate about 90000 pseudo-words. Each list contains the same amount of phonemes in C1, V1, C2 and V2 position.

3.3.9 SVT

A set of 50 sentences selected from a list of 300 sentences was produced by each speaker. These sentences present a fact that can be correct or incorrect (for example: *Paris is the capital of the United Kingdom*).

4. C2SI indexes and corpus statistics

94 patients and 41 control speakers are now included in the corpus. 87 subjects and 26 controls were finally analyzed because of missing data. Among patients, 51 (59%) were men, and the mean age was 65.8 y.o. (range 36 - 87). 9 controls (35 %) were men, and the controls' mean age was different from the patient group (56.9 y.o., range 35-79, p=0.003 Mann-Whitney). The inclusion criteria were balanced regarding tumor localisation (see figure 1): 39% of oral cavity cancer (Floor of mouth, Tongue, Retromolar Area and Mandibula), and 61% of oropharyngeal cancer (Tonsil, Root of tongue, Soft Palate and when there is a larger extension "OroPharynx").
Figure 2 presents the treatment distribution of patients. The most frequent treatment related to the size of the tumors is surgery (84%). The resection of the tumor (ChirT) is associated with the node resection (ChirN) followed in 40% by a chemoradiotherapy (RT-chimio) and in 37% by only a radiotherapy (RT).

The recorded material is processed in order to produce perceptual indexes:

LEC, DES: these speech productions have been analyzed by expert therapists in order to provide an index based on the level of comprehension as follows:

- nothing is understood (not even noticed that it is sea or boats)
- only the context (sea / boats)
- identification of other elements on the picture which makes possible to differentiate it from the thematic series
- detailed descriptions are comprehensible.

Two indexes are produced from the analysis of 6 expert speech therapists leading to individual intelligibility and severity scores. These judgments are then averaged in order to produce intelligibility and a severity scores per speaker.
The distributions of perceptual intelligibility and severity scores on a scale from 0 (low intelligibility) to 10 (high intelligibility) are described in figure 3. The average for the whole population is 7 for severity and 8.3 for intelligibility.

MOD: the recordings were presented to naive listeners, who had to recognize which modality was meant, between assertion, question and injunction. Each recorded sentence was evaluated by 3 naive listeners

FOC: Each sentence previously recorded was thereafter associated with a congruous (Qu'as-tu vu dans le jardin, un cochon ou un canard ? / eg. What did you see in the garden, a pig or a duck?) or incongruous (Où as-tu vu un canard, dans le jardin ou dans la cour ? eg. Where did you see a duck in the garden or in the yard?) question. Listeners had to judge whether the perceived focus was congruous or incongruous in the manipulated dialogues. Each recorded sentence was evaluated by 3 naive listeners

SYN: each recorded sentence was presented to naive listeners who had to choose between two pictures representing either one or the other syntactic reading (narrow *vs* broad scope of adjective). Each recorded sentence was evaluated by 3 naive listeners. For tasks SYN, FOC & MOD, a perception score was calculated for each speaker, corresponding to the mean of each perceptual evaluation obtained during the test. The mean score was associated with the listeners' mean reaction time.

DAP: All of the 50 pseudo-word lists pronounced by all speakers of the database have been transcribed by 3 naive listeners. Listeners were confronted with a task that resembles acoustic-phonetic decoding followed by a written transcription. The mean distance between the transcribed and expected response is considered as a score of (un)intelligibility.

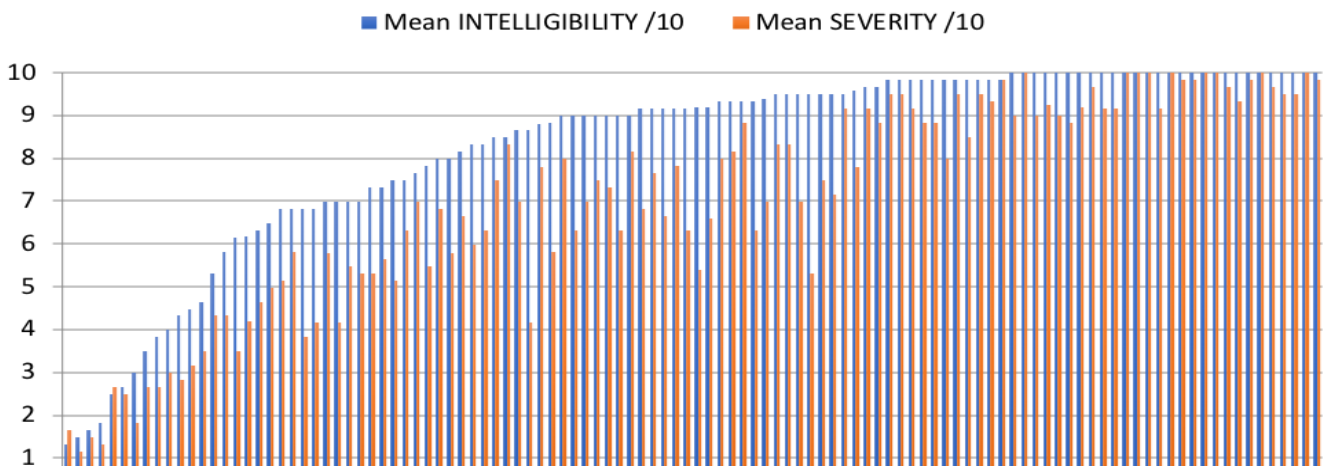


Figure 3: Distribution of intelligibility and severity scores in ordinate and subjects (by increased scores of intelligibility) in abscissa.

SVT: The sentences are evaluated by 3 naive listeners that judge if the sentence presents a correct fact or an incorrect one. This produces an indicator based of the global comprehensibility of the sentence recorded.

5. Conclusion and future work

In this paper, we have presented the design and recording of a corpus of 135 speakers, which allows us to consider the automatic production of indexes with a high level of correlation. During the constitution of the corpus, we faced several issues. Considering DAP task, patients' recordings were initially achieved, using only a visual presentation of the DAP items and the pseudo-word was simultaneously read aloud by the experimenter. But, the phonological construction of the items sometimes permitting different possible pronunciations, this configuration could have modified the speaker's repetition. To cope with this statement, we replace the aloud reading of the experimenter with a recorded synthesized voice for each item to standardize its pronunciation and to limit the potential biases. Furthermore, some tasks were considered as particularly hard to understand and to achieve by the patients (SYN, for example): the impact of these perceived difficulties will have to be checked and studied during the analysis of the results. Perceptual evaluations are in progress in order to complete the usable metadata, and to obtain reliable intelligibility/comprehensibility scores, which will be compared to self-assessed quality of life scores. We are also working now on extracting information from the different recordings in order to analyze them and to produce automatic indexes (Ghio et al. 2017, Sicard et al. 2017, Laaridh et al., 2017). This is our main goal to get objective judgments, which can help speech therapists in clinical practice. Data will be available to the scientific community by the mean of the GIS Parolotheque (https://goo.gl/4NNEZg): a scientific structure whose purpose is to facilitate access and research of pathological speech recordings (like the tumor library "thomorotheque" for access to cancer cell samples).

6. Acknowledgements

Grant n°2014-135 from Institut National pour le CAncer (INCA) in november 2014, "Sciences Humaines et Sociales, Épidémiologie et Santé Publique" call. Lead by Pr Virginie Woisard at University Hospital of Toulouse.

7. Bibliographical References

Astésano C., Bard E. G. & Turk A. (2007). Structural influences on initial accent placement in French. *Language and Speech*, *50*(3), 423-446.

Aura, K. (2012). *Protocole d'évaluation du langage fondé sur le traitement de fonctions prosodiques: étude exploratoire de deux patients atteints de gliomes de bas grade en contexte péri-opératoire* Doctoral dissertation, Université Toulouse le Mirail-Toulouse II.

Borggreven P. A., Verdonck-de Leeuw I. M., Muller M. J., Heiligers M. L., de Bree R., Aaronson N. K., Leemans C. R. (2007) *Quality of life and functional status in patients with cancer of the oral cavity and oropharynx: pretreatment values of a prospective study*. Eur Arch Otorhinolaryngol. June; 264(6):651-7

Cardol M., Brandsma J. W., de Groot I. J., van den Bos G. A., de Haan R. J., de Jong B. A. (1999) *Handicap questionnaires: what do they assess?* Disability Rehabilitation 21:97-105

Dwivedi R. C., St Rose S., Roe J. W., Chisholm E., Elmiyeh B., Nutting C. M., Clarke P. M., Kerawala C. J., Rhys-Evans P .H., Harrington K. J., Kazi R. (2011) *First report on the reliability and validity of speech handicap index in native English-speaking patients with head and neck cancer.* Head Neck. March; 33(3):341-8

Enderby P. (1983) Frenchay Dysarthria Assessment. *1st ed. San Diego: College-Hill Press*.

Enderby P., Palmer R. (2008*)* FDA-2: Frenchay Dysarthria Assessment. *2nd ed.Tex.: Pro-Ed.*

Fichaux-Bourin P., Woisard V., Grand S., Puech M., Bodin S. (2009) *Validation of a self assessment for speech disorders (Phonation Handicap Index)*. Rev Laryngol Otol Rhinol (Bord). 130(1):45-51.

Ghio A., Pouchoulin G., Teston B., Pinto S., Fredouille C., De Looze C., Robert D., Viallet F., Giovanni A. (2012), *How to manage* sound, physiological and clinical data of 2500 dysphonic and dysarthric speakers? Speech Communication, vol. 54. 2012, p. 664-679.

Ghio A., Giusti L., Blanc E., Pinto S., Lalain M., Robert D., Fredouille C. and Woisard V. (2016). *Quels tests d'intelligibilité pour évaluer les troubles de production de la parole ?* 31èmes Journées d'Etudes sur la Parole, Paris.

Ghio A., Lalain M., Giusti L., Robert D., Pouchoulin G., Rebourg M., André C., Fredouille C. and Woisard V. (2017). *Du décodage acoustico-phonétique pour mesurer l'intelligibilité de locuteurs atteints de troubles de production de la parole*, 7èmes Journées de Phonétique clinique, Paris, 29-30 june.

Laaridh I., Khader B. W., Fredouille C., Meunier C. (2017). *Automatic Prediction of Speech Evaluation Metrics for Dysarthric Speech*, Interspeech'17, Stockholm, Sweden.

Magne C., Astésano C., Lacheret-Dujour A., Morel M., Alter K., & Besson M. (2005). *On-line processing of "pop-out" words in spoken French dialogues.* Journal of cognitive neuroscience, 17(5), 740-756.

Middag C., Van Nuffelen G., Martens J. P., & De Bodt M. (2008). Objective intelligibility assessment of pathological speakers. Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech'08, 1745–1748.

Middag C., Martens J. P., Van Nuffelen G. et al. (2009). *Automated Intelligibility Assessment of Pathological Speech Using Phonological Features*. EURASIP Journal on advances in Signal Processing. 2009: 629030.

Middag, C. (2012). *Automatic analysis of pathological speech*. Doctoral dissertation, Ghent University, Department of Electronics and information systems, Ghent, Belgium.

Mlynarek A.M., Rieger J.M., Harris J.R., O'Connell D.A., Al-Qahtani K.H., Ansari K., Chau J., Seikaly H. (2008) *Methods of functional outcomes assessment following treatment of oral and oropharyngeal cancer: review of the literature.* Journal of Otolaryngol Head Neck Surg. 2008 Feb;37(1):2-10.

Pisoni D., Dedina M. (1986) "Comprehension of Digitally Encoded Natural Speech using a Sentence Verification Task: a first report" in Research on Speech Perception. Progress Report N°12, Indiana University.

Nocaudie O., Astesano C. and Woisard V. (2017). *Conservation des fonctions prosodiques post traitement des cancers de la cavité buccale et du pharynx,* 7emes Journées de Phonétique Clinique, Paris, 29-30 june.

Rinkel R. N., Leeuw I. M, van Reij E. J, Aaronson N. K, Leemans C. R. (2008) *Speech Handicap Index in patients with oral and pharyngeal cancer: better understanding of patients' complaints.* Head Neck, 30 (7) , pp. 868–874.

Samuel A. (1981) *Phonemic restoration: insights from a new methodology*. J Exp Psychol Gen.;110:474-494.

Sicard E., Mauclair J. and Woisard V. (2017). *Etude de paramètres acoustiques des voix de patients traités pour un cancer ORL dans le cadre du projet C2SI*, 7emes Journées de Phonétique Clinique, Paris, 29-30 june.

Thomas L., Jones T. M., Tandon S., Carding P., Lowe D., Rogers S. (2009) *Speech and voice outcomes in oropharyngeal cancer and evaluation of the University of Washington Quality of Life speech domain.* Clin Otolaryngol. 2009 Feb;34(1):34-42.

Wade J. E., Sherbourne C. D. *The MOS 36-item short-form health survey (SF-36)*. Medical Care (1992); 30; 473–483.

Warren R. M., Warren R. P. (1970), *Auditory illusions and confusions*. Sci. Am.; 223, 30-36.

Woisard V., Espesser R., Ghio A., Duez D. (2013). *De l'intelligibilité à la compréhensibilité de la parole, quelles mesures en pratique clinique ?* Revue de laryngologie, otologie, rhinologie, vol. 1, no. 134. 2013, p. 27-33

Parallel Corpora in Mboshi (Bantu C25, Congo-Brazzaville)

Annie Rialland[1], Martine Adda-Decker [1], Guy-Noël Kouarata[1], Gilles Adda[2], Laurent Besacier[3], Lori Lamel[2], Elodie Gauthier[3], Pierre Godard[2], Jamison Cooper-Leavitt[2]
[1] LPP, CNRS-Paris 3/Sorbonne Nouvelle, France, [2] LIMSI, CNRS, Université Paris-Saclay, France, [3] Laboratoire d'Informatique de Grenoble (LIG)/GETALP group, France

{annie.rialland, martine.adda-decker}@univ-paris3.fr, guy_kouarata@yahoo.com,
{gilles.adda, lamel, pierre.godard, cooperleavitt}@limsi.fr
{laurent.besacier, elodie.gauthier}@univ-grenoble-alpes.fr

Abstract
This article presents multimodal and parallel data collections in Mboshi, as part of the French-German BULB project. It aims at supporting documentation and providing digital resources for less resourced languages with the help of speech and language-based technology. The data collection specifications thus have to meet both field linguists' and computer scientists' requirements, which are large corpora for the latter and linguistically dense data for the former. Beyond speech, the collection comprises pictures and videos documenting social practices, agriculture, wildlife and plants. Visual supports aimed at encouraging people to comment on objects which are meaningful in their daily lives. Speech recordings are composed of the original speech in Mboshi, a respoken version and a translated version to French. These three speech streams remain time-aligned thanks to LIG-AIKUMA, which adds new features to a previous AIKUMA application. The speech corpus includes read material (5k sentences, Bible), verb conjugations and a large part of spontaneous speech (conversations, picture descriptions) resulting in over 50 hours of Mboshi speech, of which 20 hours are already respoken and orally translated to French. These parallel oral data are intended for linguistic documentation (tonology, phonology...) and automatic processing (corpus annotation, alignment between Mboshi speech and French translations).

Keywords: fieldwork, ASR, Bantu, Mboshi, French, speech, parallel corpus, pictures

1. Introduction

According to UNESCO, 43% of the 6000 estimated languages of the World are endangered. Even languages spoken by more than a million people can be threatened due to lack of transmission from one generation to the next one. Archiving languages (as well as the knowledge associated to them) is currently an emergency and an overwhelming task. The challenge is even greater when the languages are under-resourced, often lacking writing system, written texts and translated corpora. The BULB (Breaking the Unwritten Language Barrier) project aims at providing tools to language documentation and description for unwritten languages (or languages with scarce textual material) with the help of language-based technologies (in section 2). The data collection specifications thus have to meet both field linguists' and computer scientists' requirements, which are large corpora for the latter and linguistically dense data for the former. By linguistically dense, we mean calibrated material to speed up the grammar development, typically conjugations and sentence lists as proposed by Bouquiaux and Thomas (1976) among others.

The first step of the methodology of the BULB project involves the collection of parallel corpora (speech, respeaking and oral translation). They could be recorded with an application developed within the BULB project: LIG-AIKUMA, a user friendly device for linguists and speakers of the language community (in section 3). In this article, we will focus on the Mboshi language (in section 4), on the parallel corpora collected for this language (in section 5) and the first linguistic analysis done on this corpus with the help of automatic processing (section 6).

2. The BULB project

BULB (Breaking the Unwritten Language Barrier) is a French-German project supported by the French ANR and the German DFG and began in 2015. It relies on a cooperation between linguists and speech researchers in both countries. The institutions involved are KIT (Karlsruhe), University of Stuttgart, ZAS (Berlin) in Germany, LIMSI (Orsay), Laboratoire de Phonétique et Phonologie (Paris), LLACAN (Villejuif), LIG (Grenoble) in France.

Its central goal is to support language documentation and elaboration of resources for unwritten languages or less-resourced languages with the help of language-based technology, particularly automatic speech recognition and machine translation (Adda and al., 2016, Stüker and al., 2016). The methodology of the project follows Bird and al. (2014). Its steps are the following:

Collection of multimodal and parallel corpora

- collection of a large oral corpus (source language)
- respeaking of this corpus by a reference native speaker (to eliminate noises, hesitations, speaker variations)
- parallel and oral translation into a language that has access to speech technology, particularly ASR (French, the target language)
- collection of pictures and videos documenting local life and culture

Automatic treatments

- automatic time-alignment

- automatic phone annotation of source and respeaking
- automatic segmentation into words/morphemes
- automatic alignment between words/morphemes of the source and target languages.

Within the BULB project, three example languages are being addressed : Myene (Gabon), Basaa (Cameroun) and Mboshi (Congo-Brazzaville), the data collection of which are directed by LLACAN, ZAS and LPP respectively. Most of these corpora were created with LIG-AIKUMA, an application developed within the BULB project and presented section in 3.

3. LIG-AIKUMA

The initial smartphone application AIKUMA was developed by Bird and al. [2014] for the purpose of saving time in the process of language documentation and providing rather quickly, large amount of resources which could be processed and analysed later on. It enables time-aligned recording of speech with respeaking or oral translation. LIG-AIKUMA improved AIKUMA in various ways (Gauthier and al. 2016). It includes a full pipeline between recording, respeaking and translation, allowing to obtained time-aligned data between the three modes. It also implemented additional features : elicitation of speech from texts, images or videos. More detailed meta-information for speakers could be introduced and the naming of files was clarified, being based on the date and time of the recordings. Feedbacks to the user were also implemented and the application was adapted for tablets. It is downloadable from the following address : https://lig-aikuma.imag.fr and from Google Play.

4. The Mboshi language

4.1 Mboshi as an under-resourced language

Mboshi is a Bantu language (C25) of Congo Brazzaville. It is spoken in the «Cuvette region» and in Brazzaville, as well as in the diaspora. The estimated number of speakers in the «Cuvette region» is 140000 (Embanga Aborobongui, 2013). The number of speakers in Brazzaville and in the diaspora is unknown. A writing system was developed by missionaries but there is no standardised form of the orthography. There are very few texts in Mbochi, mainly translations of the Bible.

Mboshi has linguistic resources, that is, grammatical studies (Fontaney, 1988, 1989; Amboulou, 1998; Embanga Aborobongui, 2013; Kouarata, 2014) a Mboshi-French dictionary (Beapami and al., 2000) and a Mboshi-English dictionary (Ndongo Ibara, 2012). Meanwhile, it has no digital resources. Thus, Mboshi has no on-line dictionary yet.

4.2 Some linguistic characteristics of Mboshi

Mboshi is a typical Bantu language. It is classified C25, in zone C which belongs to the Western part of the Bantu domain. It has a seven vowel system (i, e, ɛ, a, ɔ, o, u) with an opposition between long and short vowels. Its consonantal system includes the following phonemes: p, t, k, b, d, β, l, r, m, n, ɲ, mb, nd, ndz, ng, mbv, f, s, ɣ , pf, bv, ts, dz, w, j. We can notice the absence of /g/, the presence of a voiced bilabial fricative / β/ and a set of prenasalized consonants (mb, nd, ndz, ng, mbv) which are common in Bantu languages (Embanga Aborobongui 2013, Kouarata 2014). The Mboshi prosodic system involves two tones and an intonational organisation without downdrift (Rialland and Embanga Aborobongui 2016). The syllable structures are simple (V, CV, CVV, Cj/wV, Cj/wVV).

While Mboshi has unremarkable vocalic and consonantal systems, it has quite complex phonological rules. A process is particularly frequent: the deletion of a vowel before another vowel, which occurs at 40% of word junctions (Rialland and al. 2015) as exemplified in (1) and (2).

(1) o-kondzi + ásɛri → okondzáseri
cl1.chief + 3sg.say.REC.
«The chief said.»

(2) o-yúlu + álámbi → o-yúlálámbi
cl1.femme + 3sg.cook.REC.
«The woman cooked.»

This process, which is common in Bantu languages, tends to obscure word segmentation and introduces an additional challenge for automatic processing, particularly automatic word segmentation and dictionary creation.

Mboshi has a noun class prefix system, which is another typical feature of Bantu languages. However, it has an unusual rule of deletion targeting the consonant of prefixes: a prefix consonant drops if the root begins with a consonant (Rialland and al. 2015). This rule triggers the formation of many words beginning by a vowel as shown by the following nouns involving the class prefix -ba.

(3) ba+kondzi → akondzi «chiefs»

(4) ba+kúsu → akúsu «tortoises»

(5) ba+ ási → bási «wives»

(6) ba+ ána → bána «children»

This type of rule is shared by a small group of languages in this area. The structure of the verb in Mboshi is also characteristic of a Bantu verb. Its structure is the following : Subject Marker – Tense/Aspect/Mood Marker/ – root-derivative extensions – Final Vowel. A verb can be very short or quite long, depending of the markers involved as shown by the examples (7) and (8):

(7) a-bva-í [bvé] «he falls»
SM3sg – fall - FV

(8) í-mi-ding-im-a [ímidingima] «he was loved»
SMcl5- PERF-love-PASS-FV

5. Composition and collection of the parallel corpora

The parallel corpora collection involves three phases : 1) collection of audio corpora, 2) Respeaking, 3) Translation. All of them were recorded with LIG-AIKUMA. A small part was also transcribed manually.

5.1 Phase 1: Composition and collection of audio corpora

Oral corpora include: a) read sentences, b) debates, c) conjugations, d) reading of Bible, e) comments on pictures

a) Read sentences.

A total of 5178 read sentences were collected. Among these, 3706 sentences were extracted from the Mboshi-French dictionary (Beapami and al., 2000) and 1472 were Mboshi translations of examples from Bouquiaux and Thomas's corpus designed for fieldwork (Bouquiaux and Thomas, 1976). These two sets of sentences were recorded using the text-based elicitation function of LIG-AIKUMA. An Mboshi written sentence associated with its translation in French was displayed on the tablet screen. The tablet was given to speakers who could manipulate it by themselves, after a short period of instruction. The recording of these sentences was divided between 3 speakers and its was made in Congo-Brazzaville. The total duration of this recording is 4h51.

This 5k read sentences corpus was used for acoustic-phonetic studies as described in section 6 (Cooper-Leavitt J. and al. 2017 a, b). It provided also a testing ground for machine learning studies e.g. unsupervised word discovery from speech and recently, it was used during the Jelinek Summer Workshop on Speech and Language Technology (JSALT) 2017 in CMU, Pittsburgh (Godard and al. 2016, Godard and al. 2018).

b) Debates

Debates were preferred to monologues or life stories as they provide an interaction which favors spontaneous speech and natural exchanges. They were moderated by one of the co-authors of this article: Guy-Noël Kouarata, who is also a native speaker and a linguist. There are 67 debates, each of them involving between 2 and 5 speakers. Altogether there were 19 participants in these debates. To avoid any overlapping, speakers were instructed not to speak at the same time. The debates deal with a variety of topics concerning traditions or reflecting current concerns, such as the death of an old person, the structure of a song, regional mushrooms, alimentation before and after the building of a good road to access the region, sorcellery, deseases, immigration… Some of these debates are valuable in terms of patrimony and some encode cultural knowledge (about plants, or mushrooms, for example). These debates were recorded with a tablet in a home in Brazzaville as shown in the following picture (1):



Figure 1. Recordings of debates with a tablet in a home in Brazzaville

Each debate lasts between 24 and 80 minutes. The whole duration of the recorded debates was 25h18mn whose 20h22 were respoken and translated (see 5.2 and 5.3.). The speakers signed a consent form for being recorded and have their recordings made public but some of these debates, being culturally sensitive, might have to be checked by specialists (botanists, for example) for an appropriate avaibility.

c) Conjugations

To facilitate the process of grammar development, verb conjugations are of special interest. Verb morphology is complex in Mboshi as well as in Bantu languages in general. This complexity introduces additional challenges for automatic processing, particularly in word segmentation and mapping between Mboshi word/morphemes and their French counterparts in translation. Conjugations were added to the corpora as linguistic facilitators for the planned processings. We knew also that it was quite easy to make systematic recording of conjugations as we could build on the experience that speakers acquire in conjugations during their schooling, which is in French from the beginning in Congo-Brazzaville. The corpora was designed to cover a large part of the verbal conjugations. 50 verb based on the main root patterns (C (V), CV or CVC) were conjugated at 15 tenses/aspects. The subjects were also varied, to capture the agreements between the subjects and the verbs. 18 different subjects (pronouns or nouns with various class prefixes) were necessary to cover the spectrum of the agreement patterns.

These conjugations were recorded by one speaker, based on conjugation tables (lists of verbs, subjects, tenses). The total duration of this recording is 5h56mn. No respeaking or translation of these conjugated forms were done as the verbs and their conjugations were known.

d) Reading of Bible extracts

Extracts of Bible in Mboshi were read by 6 speakers. The total amount of these recordings is: 4h06mn. No respeaking or translation had to be done.

e) Comments on pictures

Guy-Noël Kouarata took 1500 pictures illustrating plants, artifacts, animals and everyday activities to be included later on in an Encyclopedia or to be archived as culturally sensitive. Figures (2) and (3) provide a sample of these pictures.



Figure 2. Mask for the kyebe-kyebe danse and ceremony



Figure 3. Traditional fishtrap

These pictures were commented by 2, 3 or 4 speakers. In total, 8 speakers participated in these recordings. Each comment lasted between 20 seconds to 3 minutes. Currently, they are not respoken or translated but kept for further completion and processing. . .

5.2 Phase 2: Respeaking

The task of respeaking was performed by three different native speakers in a quiet room in Brazzaville. Speakers were instructed to repeat, eliminating hesitation pauses, speaking slowly, still naturally. The task was found more difficult than expected, with a strong tendency among the speakers to come back to their usual rate of speech, which was often quite fast. The respeaking was made for 20h22 mn of source data.

5.3 Phase 3: Translation

The translation in French was performed by Guy-Noël Kouarata in a quiet room. A rather literal translation was preferred in order to improve the possibility of matching automatically Mboshi words and their translation to French. The translation function of LIG-AIKUMA was used. The whole pipeline (oral audio corpus, respeaking, translation) was obtained for 20h22mn hours of source speech.

Type of corpus	#speakers	quantity	dur. (h)	respoken	translated	manually transcribed
Read sentences	3	5178 sentences	4h51		x (written)	x pre-existing written sentences
Debates	19	67	25h18	x (20h22)	x (20h22) (oral)	x (1h10)
Bible reading	6		4h06			
Conjugations	1	50verbs*15TAM* 18subjects	5h56			x pre-existing conjugation tables
Comments on pictures	8	1500 pictures	~15h			

Table 1. Mboshi parallel corpora : current state.

5.4 Manual transcription

A small part of the debates was transcribed, based on the annotation conventions of the Mboshi-French dictionary (Beapami and al. 2000). The duration of the manually transcribed part is 1h 10 minutes.

6. First linguistic analysis of the corpus with the help of automatic treatments

Forced text-to-speech alignment was applied to the 5178 read sentences. LIMSI's STK speech processing toolkits for the ASR were used (Gauvain and Lamel, 2003; Lamel and Gauvain 2005). A variant dictionary was built, based on the manual transcripts and rules generating variants. With forced alignment and pronunciation variants in the dictionary, various types of vowel deletion or morpheme deletion at word junction (triggered by phonetic processes or specific to some morphemes, in particular, the connective ones) could be sorted out and better quantified and understood. (Cooper-Leavitt J. and al. 2017 a, b).

7. Conclusion

This article presents multimodal and parallel corpora recorded in Mboshi (Bantu C25). Their current state is summarized in Table 1. These corpora were designed to document the language itself, to contribute to the patrimony preservation of Mboshi people and to provide materials which could be processed by current language-based technologies. Parallel corpora collection with a smartphone/tablet and the friendly-user application AIKUMA can also have other purposes than linguistic ones, for example in music, as songs and lyrics can be stored in parallel. The final corpus which will contain more annotations, respeaking and oral translation will be made available to the research community at the end of the BULB project.

8. Acknowledgements

This work was funded by the French ANR and the German DFG under grant ANR-14-CE35-0002. It also received financial support from the EFL LABEX (10-LABX-0083).

9. Bibliographical References

Adda, G., Stücker, S., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P, Hamlaoui F., Idiatov D., Kouarata G.-N., Lamel, L, Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon F., and Zerbian S. (2016). Breaking the unwritten language barrier: The Bulb project. *In Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia

Amboulou, C. (1998). *Le Mbochi : langue bantoue du Congo Brazzaville (zone C, groupe C20)*. Ph.D. thesis, INALCO, Paris.

Beapami, R. P., Chatfield R., Kouarata, G.-N. and Embengue Waldschmidt, A. (2000). *Dictionnaire Mbochi-Français*. SIL-Congo Publishers, Brazzaville.

Bird, S., Hanke, F. R., Adams, O. and Lee, H. (2014). Aikuma: A mobile app for collaborative language documentation. *Workshop on the Use of Computational Methods in the Study of Endangered Languages*, Baltimore, USA. pp 1-5

Bouquiaux, L. and Thomas J. (1976). *Enquête et description des langues à tradition orale*. SELAF, Paris, France

Cooper-Leavitt J., Lori L., Rialland A., Adda-Decker M., Adda G. (2017 a). Developing an Embosi (Bantu C25) speech variant dictionary to model vowel elision and morpheme deletion. In *Conference of Interspeech 2017 Proceedings.*

Cooper-Leavitt J., Lori L., Rialland A., Adda-Decker M., Adda G. (2017 b). Corpus base linguistic exploration via forced alignments with a "light-weight" ASR tool. In *Workshop on Language Technology*, Poznan, Poland.

Embanga Aborobongui M. (2013). *Les processus segmentaux et tonals en mbondzi (variété de la langue Embosi. C25)*. Ph.D thesis, Paris 3 University, Paris.

Fontaney, L. (1989). Mboshi: steps toward a Grammar: Part I. *Pholia* 3, pp. 87-169

Fontaney, L. (1989). Mboshi: steps toward a Grammar: Part II. *Pholia* 4, pp. 71-131

Gauthier, E., Blachon D., Besacier L., Kouarata, G.-N., Adda-Decker, M. and al. (2016). LIG-AIKUMA: a Mobile App to Collect Parallel Speech for Under-Resourced Language Studies. *Interspeech 2016* (short demo paper), Sept 2016, San-Francisco, Interspeech 2016 Proceedings.

Gauvain, J.-L. and Lamel L., 2003. Large vocabulary speech recognition based on statistical methods. In Chou and Wang J. (eds.), *Pattern Recognition in Speech and Language Processing*. CRC Press, pp. 149–189.

Godard, P., Adda, G., Adda-Decker, M., Benjumea, J., A., Besacier, Cooper-Leavitt J., L., Kouarata, G.-N., Lamel L., Maynard, H., Rialland A., Stüker, S., Yvon, F. and Zanon-Boito M. (2018). A Very Low Resource language Speech Corpus for Computational language Documentation Experiments, In *LREC 2018* (in press), Japan

Godard, P., Adda, G., Adda-Decker, M., Allauzen, A., Besacier, L., Bonneau-Maynard, H., Kouarata, G.-N., Löser, K. Rialland A., and Yvon, F. Preliminary Experiments on Unsupervised Word Discovery in Mboshi. In *Interspeech 2016*, San Francisco, California, USA

Kouarata, G.-N. (2014) *Variations de forme dans la langue Mbochi (Bantu C25)*, Ph.D. thesis, Lyon 2 University, Lyon.

Lamel, L. and Gauvain J.-L. (2005). *Speech recognition*. In R. Mitkoc (ed.). The Oxford Handbook of Computational Linguistics, Oxford University Press, Oxford, pp. 305–32

Ndongo Ibara Y. P. (2012). *Embosi-English Dictionary*. Peter Lang, Frankfurt

Rialland, A., Embanga Aborobongui M., Adda-Decker, M. and Lori Lamel, (2015). Dropping of the class-prefix consonant, vowel elision and automatic phonological mining in embosi (Bantu C 25). In Kramer, R., Zsiga E., and O. Tlale Boyer (eds.), *Selected Proceedings of the 44th Annual Conference on African Linguis*tics. Cascadilla Press, Somerville, USA, pp 221-230

Rialland A. and Embanga Aborobongui M. (2016) How intonations interact with tones in Embosi (Bantu C25), a two-tone language without downdrift. In L. Downing and A. Rialland (eds.), *Intonation in African tone languages.* Mouton de Gruyter, Berlin. pp. 195-222

Stücker, S., Adda; G., Adda-Decker, M., Ambouroue, O., Besacier, L., Blachon, D., Bonneau-Maynard, H., Godard, P, Hamlaoui F., Idiatov D., Kouarata G.-N., Lamel, L, Makasso, E.-M., Rialland, A., Van de Velde, M., Yvon F., and Zerbian S. (2016). Innovative technologies for under-resourced language documentation: The Bulb project. In *Proceedings of CCURL (Collaboration and Computing for UnderResourced Languages: toward an Alliance for Digital Language Diversity)*, Portoroz Slovenia.

.

A Multimodal Corpus of Expert Gaze and Behavior during Phonetic Segmentation Tasks

Arif Khan[1–3], Ingmar Steiner[1,2], Yusuke Sugano[4], Andreas Bulling[1,5], Ross Macdonald[6]
[1]Multimodal Computing and Interaction, Saarland University, Germany,
[2]German Research Center for Artificial Intelligence (DFKI GmbH), Saarbrücken, Germany,
[3]Saarbrücken Graduate School of Computer Science, Germany,
[4]Osaka University, Japan
[5]Max Planck Institute for Informatics, Saarbrücken, Germany
[6]University of Manchester, UK,
{arifkhan,steiner}@coli.uni-saarland.de

Abstract

Phonetic segmentation is the process of splitting speech into distinct phonetic units. Human experts routinely perform this task manually by analyzing auditory and visual cues using analysis software, which is an extremely time-consuming process. Methods exist for automatic segmentation, but these are not always accurate enough. In order to improve automatic segmentation, we need to model it as close to the manual segmentation as possible. This corpus is an effort to capture the human segmentation behavior by recording experts performing a segmentation task. We believe that this data will enable us to highlight the important aspects of manual segmentation, which can be used in automatic segmentation to improve its accuracy.

Keywords: eyetracking, gaze analysis, manual segmentation behavior

1. Introduction

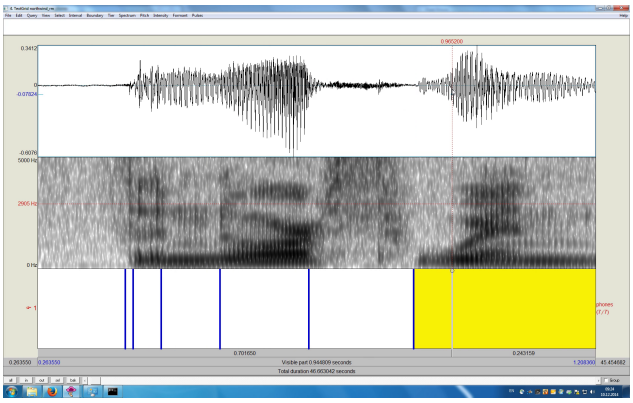
Speech segmentation is the process of splitting the acoustic speech signal into distinct units by placing timestamped boundaries. This forms a crucial data processing step for phonetic analysis, as well as speech technology applications such as text-to-speech synthesis and automatic speech recognition. The results and output quality depend on accurately segmented speech data.

Speech segmentation can be done manually, using specialized software, e.g., *Praat* (Boersma, 2001), *Wavesurfer* (Beskow and Sjölander, 2000), *ELAN* (Sloetjes and Wittenburg, 2008), and *EMU* (Winkelmann et al., 2017). In this workflow, a speech recording is displayed as a waveform and/or spectrogram, and boundaries are inserted using the mouse or keyboard (cf. Figure 1a). Short audio segments can be played back to validate the boundary placement. This process is repeated until the whole audio file is segmented. Manual segmentation by experts is considered to produce the best phonetic segmentation one can achieve for any given data (Svendsen and Soong, 1987; Wesenick and Kipp, 1996). One reason for this is that they combine experience with multiple sources of information. However, there are some critical drawbacks of manual segmentation which make it impractical for large speech corpora. The first is that it is very laborious and time consuming; on average, manual segmentation can take up to 30 s per phone (Leung and Zue, 1984; Stolcke et al., 2014) to segment. As a result, newly recorded speech data cannot be used quickly if manual segmentation is desired. Secondly, the exact placement of boundaries is subjective, and there may be disagreement between multiple experts.

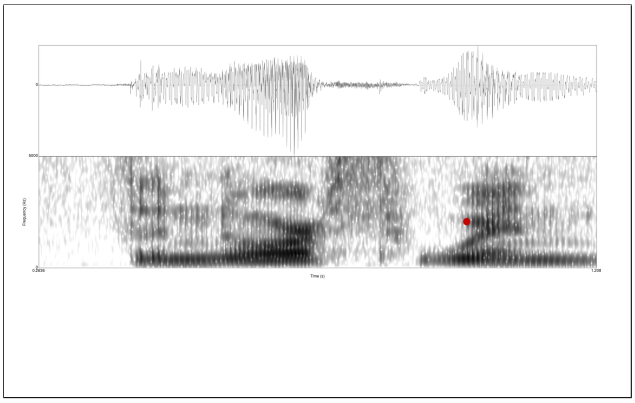
The second method of segmentation is doing it automatically, by training a model on the audio data, and then using it to segment speech. In this method, the accuracy of the segmented speech directly depends on the quality of the trained model, which itself depends on the quality of training data. Previous studies have used different approaches for automatic segmentation. For a long time, researchers have used hidden Markov models (HMMs) for automatic segmentation (Rabiner, 1989; Juang and Rabiner, 1991; Toledano et al., 2003; Brognaux and Drugman, 2016). Others have used neural networks for automatic segmentation (Karjalainen et al., 1998; Schwarz et al., 2006). One commonality of these approaches is the use of only audio as input features for training the model. The audio is processed to extract acoustic features, which are then used for training. Several techniques are available for extracting acoustic features from speech, but the most commonly used are mel-frequency cepstral coefficients (MFCCs) (Logan, 2000) and Perceptual Linear Prediction (PLP) (Hermansky, 1990). While the use of only audio as acoustic features produces acceptable results for most segmentation requirements, humans use more than audio for segmenting speech. To improve automatic segmentation, we therefore want to add more modalities to model it as closely as possible to the manual segmentation. We hope that modeling automatic segmentation in this way will produce better results.

To this end, we first need to analyze the human segmentation behavior and highlight the key information sources that humans experts use to segment speech. Our data includes gaze information, which shows where the experts look on the screen, the audio to which they listen during segmentation, video from a webcam attached to the monitor, and a screen recording of what they are viewing. To the best of our knowledge, this is the first corpus that records the human segmentation in such a setup.

The rest of this paper is organized as follows. Section 2 provides details of how the data was recorded, along with the format and structure. In Section 3, we present the results of some preliminary analysis conducted on the data. Finally,



(a) A screenshot of a sound recording and annotation in Praat. The GUI is split into three sections: waveform (top), spectrogram (middle), and annotation (bottom).



(b) Corresponding scene data reconstructed from recorded audio using Praat log and gaze information. Here, the subject is looking at a formant in the spectrogram; the fixation is rendered in red.

Figure 1: One frame from the screen capture video (left) and the corresponding reconstruction (right).

the conclusion and future use of the data is outlined in Section 4.

2. The Corpus

In order to study the behavior of human experts during speech segmentation tasks, we designed and recorded the multimodal corpus described in this section.

2.1. Preparation

We recorded a native speaker of Scottish English, reading aloud the standard passage, "The North Wind and the Sun" (International Phonetic Association, 1999). The recording was made in a sound-attenuated booth, with a close-talking microphone, sampling at 48 kHz with 24 bit quantization. The resulting file has a duration of 46 s.

2.2. Data Collection

We recorded seven subjects, with the instruction that they were to segment (but not label) the recording into phones using the Praat graphical user interface (GUI). All of the subjects who participated in the data collection are trained phoneticians with varying amounts of experience; details are given in Table 1.

The participants took different amounts of time (44 to 96 min) to complete the task. The normalized session duration for all the subjects in shown in Figure 2. We did not control the speed in which the participants completed the task, so each took time according to his or her preference, which resulted in different session durations.

We used a Tobii TX300 eyetracker,[1] to record the gaze movements and capture where the subject looked on the computer screen during the entire session, at a sampling rate of 120 Hz. For each subject, before the beginning of recording, we first calibrated the eyetracker. The calibration is done to adjust the height of head and seating position, which is different for each subject. Using the *TobiiStudio* software (v3.2.3), we also recorded the screen content itself (at a resolution of 1920×1200 pixels), as well as any audio the subjects played back from the recording during the segmentation task. In addition to the gaze information and screen recording, TobiiStudio also allowed us to log any keystrokes and mouse clicks during the recording session, as well as the video from a webcam facing the subject, at a resolution of 640×480 pixels. The screen capture and webcam were intended to validate the subjects' head movements and input device logging.

In addition to these modalities, we polled the application state of the Praat GUI, once per second, in order to log the zoom level of the audio recording shown and other application-specific data. Finally, the segmentation itself, produced by each subject over the course of the session, was saved in Praat's widely supported *TextGrid* annotation file format.

2.3. Data Processing

After each recording session, the logs from TobiiStudio and Praat were exported to ASCII text files and compressed. The screen recordings and webcam videos, as well as the audio playback recordings, were exported from TobiiStudio in ASF containers, in TechSmith Screen Capture Codec (TSCC), Microsoft Video 1, and MP3 format, respectively, the latter at 22 kHz and 16 bit quantization, at a bitrate of 128 kbit/s.

In order to manipulate the multimedia streams from each recording session more efficiently, we first converted the video to H.264 format (which allowed more robust seeking and reduced the file sizes – from 52 GB to 3 GB without noticeable loss in quality), transcoded the audio to FLAC format,[2] and multiplexed all three streams into a single Matroska video container,[3] using FFmpeg.[4]

Next, we parsed the Praat logs to identify time segments in each recording session during which the subject was viewing the same zoom level and interval of the audio recording; doing this allowed us to treat them as quasi-static *scenes* viewed by the subject. The session times as well as the audio recording times of each scene were collected into a

[1] https://www.tobiipro.com/product-listing/tobii-pro-tx300

[2] https://xiph.org/flac/

[3] https://matroska.org/

[4] https://ffmpeg.org/

Subject	Gender	Age (years)	Native Language	Experience (years)	Segmentation Time (min)
01	F	26	German	7	44
02	M	47	German	20	55
03	M	37	German	15	73
04	F	35	Polish	10	96
05	F	27	German	4	71
06	F	22	German	1.5	80
07	F	22	German	4	92

Table 1: Age, gender, native language, and segmentation experience of the subjects who participated in the data collection.

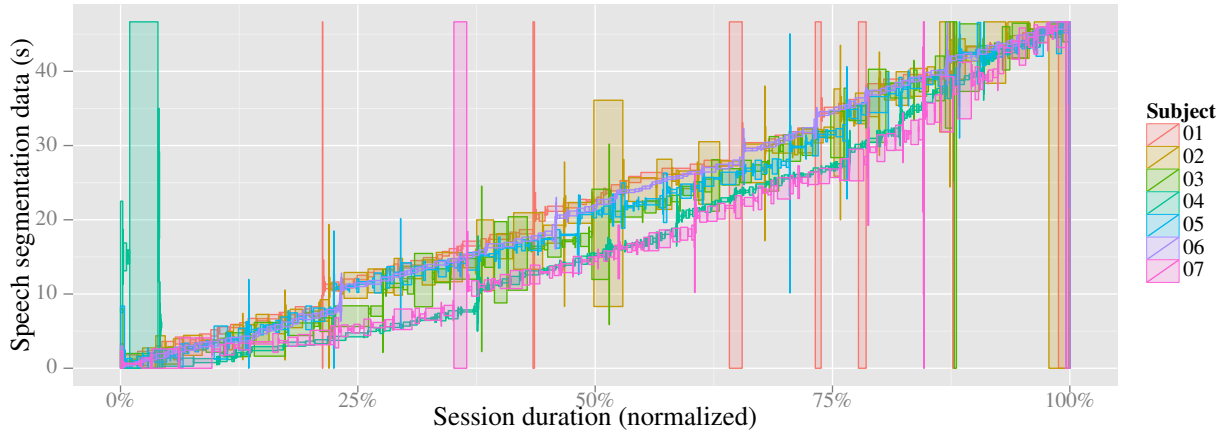


Figure 2: Speech segmentation data spans which were viewed as scenes over the (normalized) duration of the segmentation task. Each rectangle represents the portion of time (rectangle width) spent segmenting a span of recorded speech, while the rectangle height represents the duration of that span.

YAML file.[5]

After determining the constant time offset between the Praat and TobiiStudio logs, we could then select the gaze data related to each scene and store it in a structured format, validating it via the screen recording. The data is structured by scene and also includes the duration and location (absolute and classified by GUI region) of each fixation. Based on this information, we reconstructed the relevant information in each scene and synthesized it into a second video stream with the gaze location rendered as a red circle (cf. Figure 1b). We also extracted the signal time codes of each scene and added them as a subtitle track. The resulting YAML files and multimedia streams were finally packaged and provided as a data dependency for analysis.

3. Analysis

Our initial analysis concerns the eyetracking data. The main purpose of the corpus was to allow us to analyze the manual segmentation behavior and to identify modalities and features useful for modeling segmentation. For the analysis of the eyetracking data, it is important to understand the concepts of *fixation* and *saccades*. If the $\langle x, y \rangle$ location of the gaze on the screen does not change significantly within some time frame, then those gaze events are classified as a fixation. The movement of gaze between two fixations is referred to as a saccade. The actual time duration for which the $\langle x, y \rangle$ location movement should remain constant is subjective and device dependent. We used the default settings of Tobii to identify the fixations and saccades, the details of which are described by Ollson (2007). For analysis, the Praat GUI on the screen is divided horizontally into three sections, each representing a different portion of the screen. We refer to these sections as *waveform*, *spectrogram*, and *annotation*, as shown in Figure 1. The waveform represents the oscillogram of the audio recording in Praat. The spectrogram represents the time-frequency-energy representation of the signal; the x axis represents time, and the y axis, the frequency of the signal, while the grayscale value indicates the energy in each time-frequency bin. The annotation section is used by the subjects to place the boundaries. This is the only section which can be edited by the user for creating and manipulating time-aligned annotations (boundaries and labels).

3.1. Scenes

Further to the progress visualization in Figure 2, Table 2 summarizes the number of scenes the subjects viewed over the course of their session. As can be seen, subjects 01 to 03 and 05 to 07 used almost the same number of scenes for segmentation. Subject 04 viewed a larger number of scenes with the second lowest average scene length, indicating that this participant preferred to "zoom in" more than the others.

3.2. Fixations

One of the most important questions is, where the subjects look on the screen during the manual segmentation task. To

[5] http://yaml.org/

Subject	Total scenes	Total duration (s)	Average duration (s)
01	157	519.79	3.31
02	157	593.63	3.78
03	150	562.23	3.74
04	522	627.67	1.20
05	308	671.64	2.18
06	352	361.50	1.02
07	276	652.49	2.36

Table 2: The total number of scenes, sum of scenes length and average scene length the subjects used for segmenting the audio recording.

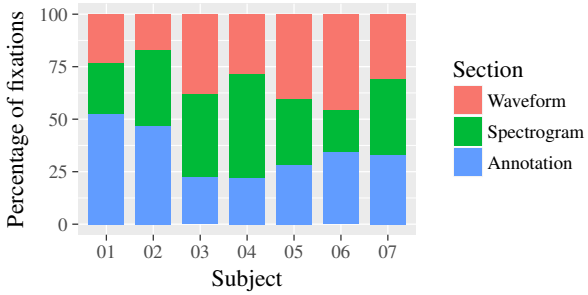


Figure 3: Average fixations for each subject in the three sections of the Praat GUI.

answer this question, we calculated the proportion of gaze events in the three sections of the screen. Figure 3 shows the percentage of fixations in each of the three screen sections for all subjects. The fixations in the *annotation* area can be disregarded, because in order to place the boundary, the subjects have to carefully "click" in the right location and during this process, a lot of gaze activity may occur in this section. The fixations in the *waveform* and *spectrogram* sections are important and have a mixed pattern. All subjects have a higher number of fixations in the *spectrogram* section than in the *waveform* section.[6]

4. Conclusion and Outlook

In this paper, we have presented a multimodal corpus of behavior data from expert phoneticians performing a manual speech segmentation task. All important information sources that are relevant to the segmentation task were recorded. This includes gaze, playback audio, video, and screen recording. The produced segmentation, as well as events logged from the keyboard, mouse, and Praat GUI are also provided. We believe that this data will prove valuable for research in observing and understanding manual segmentation.

This corpus can help identify critical information sources used by humans during manual segmentation, which can be modeled to improve the accuracy of automatic segmentation. In addition, this data can be useful in analyzing the interaction of phoneticians with speech segmentation software (Praat) and can be used to improve the usability of such a software. For example it might be possible to modify the way the boundaries are defined or to introduce a software feature to visualize the predicted complexity of speech regions while they are being segmented.

The processed data (cf. Section 2.3) has been released under a Creative Commons license (CC-BY-NC-SA) and published on GitHub,[7] along with the processing recipes. This public release *excludes* the webcam videos, in order to protect the privacy of our participants.

[6]The exception is subject 06; this may be because she had the least amount of segmentation experience (see Table 1) and relied more on the waveform section to segment.

5. Acknowledgements

We are extremely grateful to all of our participants, who gave their time for the segmentation task and provided valuable feedback.

This study was funded by the German Research Foundation (DFG) under grant number EXC 284.

6. Bibliography

Beskow, J. and Sjölander, K. (2000). WaveSurfer: An open source speech tool. In *International Conference on Spoken Language Processing (ICSLP)*, Beijing, China.

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345.

Brognaux, S. and Drugman, T. (2016). HMM-based speech segmentation: Improvements of fully automatic approaches. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 24(1):5–15.

Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752.

International Phonetic Association. (1999). *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press.

Juang, B. H. and Rabiner, L. R. (1991). Hidden Markov models for speech recognition. *Technometrics*, 33(3):251–272.

Karjalainen, M., Altosaar, T., and Huttunen, M. (1998). An efficient labeling tool for the QuickSig speech database. In *International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia.

Leung, H. and Zue, V. (1984). A procedure for automatic alignment of phonetic transcriptions with continuous speech. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 9, pages 73–76.

Logan, B. (2000). Mel frequency cepstral coefficients for music modeling. In *International Symposium on Music Information Retrieval (ISMIR)*, Plymouth, MA, USA.

Ollson, P. (2007). Real-time and offline filters for eye tracking. Master's thesis, KTH.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[7]https://github.com/m2ci-msp/eyetracking-data

Schwarz, P., Matějka, P., and Černocký, J. (2006). Hierarchical structures of neural networks for phoneme recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume I, pages 325–328, Toulouse, France.

Sloetjes, H. and Wittenburg, P. (2008). Annotation by category: ELAN and ISO DCR. In *International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

Stolcke, A., Ryant, N., Mitra, V., Yuan, J., Wang, W., and Liberman, M. (2014). Highly accurate phonetic segmentation using boundary correction models and system fusion. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5552–5556, Florence, Italy.

Svendsen, T. and Soong, F. K. (1987). On the automatic segmentation of speech signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 77–80, Dallas, TX, USA.

Toledano, D. T., Gómez, L. A. H., and Grande, L. V. (2003). Automatic phonetic segmentation. *IEEE Transactions on Speech and Audio Processing*, 11(6):617–625.

Wesenick, M.-B. and Kipp, A. (1996). Estimating the quality of phonetic transcriptions and segmentations of speech signals. In *International Conference on Spoken Language Processing (ICSLP)*, pages 129–132, Philadelphia, PA, USA.

Winkelmann, R., Harrington, J., and Jänsch, K. (2017). EMU-SDMS: Advanced speech database management and analysis in R. *Computer Speech & Language*, 45:392–410.

Statistical Analyses of Missing Translations in Simultaneous Interpretation Using a Large-scale Bilingual Speech Corpus

Zhongxi Cai[1], Koichiro Ryu[2], Shigeki Matsubara[3]
[1]Graduate School of Informatics, Nagoya University, [2]Tsu City College
[3]Information and Communications, Nagoya University
[1,3]Furo-cho, Chikusa-ku, Nagoya, 464-8601, Japan
[2]Isshindennakano 157, Tsu, Mie, 514-0112, Japan
cai.zhongxi@f.mbox.nagoya-u.ac.jp, ryu@tsu-cc.ac.jp, matubara@nagoya-u.jp

Abstract

This paper describes statistical analyses of missing translations in simultaneous interpretations. Eighty-eight lectures from English-to-Japanese interpretation data from a large-scale bilingual speech corpus were used for the analyses. Word-level alignment was provided manually, and English words without corresponding Japanese words were considered missing translations. The English lectures contained 46,568 content words, 33.1% of which were missing in the translation. We analyzed the relationship between missing translations and various factors, including the speech rate of the source language, delay of interpretation, part-of-speech, and depth in the syntactic structure of the source language. The analyses revealed that the proportion of missing translations is high when the speech rate is high and delay is large. We also found that a high proportion of adverbs were missed in the translations, and that words at deeper positions in the syntactic structure were more likely to be missed.

Keywords: Simultaneous interpretation, Speech corpus, Bilingual corpus, Speech translation

1. Introduction

In simultaneous interpretation, an interpreter must convey their translation to the target language while simultaneously listening to, comprehending, and memorizing the content of the source language speech. This is challenging and involves a number of difficulties that can reduce the quality of the interpretation. If such difficulties can be detected automatically, a support environment for simultaneous interpretation that presents the speech content or provides candidate translations could be achieved.

Gile (1995) referred to interpretation difficulties as "problem triggers". Such triggers include the density of the source speech (e.g., the delivery rate and the density of information in the content). In addition, unfamiliar names, numbers, and complex syntactic structures, etc. are challenging for interpreters. Since most of the time interpreters work at near capacity, such additional challenges can lead to failure (Gile, 1999). Interpretation failures include errors, omissions, and infelicities (Gile, 2009). In this paper, omissions are used to identify when an interpreter has encountered a difficulty.

Time constraints are severe in simultaneous interpretation such that it is impossible to translate all speech content; thus, omissions are inevitable (Dillinger, 1994). To improve interpretation quality or develop a method to train interpretation skills, several studies have investigated omissions in simultaneous interpretation. Various types of omissions have been described and several factors related to omissions have been studied (Barik, 1994). However, these analyses, which were based on observation, did not clarify the correlation between the identified factors and the occurrence of omissions.

In this paper, to detect occasions where interpreters would encounter difficulties, we statistically analyzed the correlation between source speech features and interpretation conditions and the occurrence of missing translations. In the analyses, we used 88 lectures of English-to-Japanese (E-J) interpretation data from the Simultaneous Interpretation Database (SIDB) (Matsubara et al., 2002). Note that word-level alignments were created manually.

2. Missing Translations in Simultaneous Interpretation

In simultaneous interpretation, departures from the source speech in interpreters' renditions include omissions, additions, and errors. Omissions refer to items that are present in the source speech but not included in the translation (Barik, 1994). However, if an interpreter does not translate a lexically irrelevant repetition or a mistake in the source speech, such as a false start, it is not considered an omission because these are phenomena in spontaneous language. Barik classified omissions into four categories, i.e., skipping, comprehension, delay, and compounding omissions. Barik found that more qualified interpreters omit 5% to 10% of the source speech, and less qualified interpreters omit 20% to 25%. Dillinger (1994) investigated the differences between experienced and inexperienced interpreters relative to comprehension and found that only 65% to 80% of propositions were processed accurately by experienced interpreters. However, although several factors relating to omissions, such as the rate of speech and delay, were referenced in these studies, the actual rate of speech and delay were not calculated. Thus, the correlation between the extent of such factors and the occurrence of omissions was not evaluated.

In this paper, we define the omission phenomena described above as missing translations. According to the "problem triggers" (Gile, 1995) and the omission categories (Barik, 1994), missing translations are related to speech rates, delay, the types of words, and the syntactic structures.

	Lecture	Interpretation
# of lectures	22	88
# of utterance units	5,053	18,414
# of words	28,065	141,179
Total during (sec)	12,789	51,269

Table 1: Statistics of English lectures and E-J interpretations

In this study, statistical analyses were conducted to investigate correlations between missing translations in simultaneous interpretation and these four factors.

3. Overview of Analyses

As mentioned previously, factors such as high speech rates, complex syntactic structures, names, and numbers increase interpreter workload and can result in missing translations in simultaneous interpretation. According to Barik (1994), missing translations are primarily due to interpretation delay because the content currently being delivered by the source speaker may not register with the interpreter or may be ignored by the interpreter while they are speaking their translations. In addition, as speech content accumulates during the delay, the working memory of the interpreter may become overloaded; thus, they may fail to provide accurate translations (Mizuno, 2005).

In this study, the rate of speech, delay, types of words (part-of-speech), and a word's depth in the syntactic structure were considered, and statistical analyses were conducted to investigate the relationship between the occurrence of missing translations and the extent of these factors.

3.1. Analyzed Data

In this study, we used data from the SIDB (Matsubara et al., 2002).

3.1.1. Scale and Features of Analyzed Data

The SIBD includes monologue data (lectures) and dialogue data, and their corresponding J-E and E-J interpretations. In our analyses, 22 English lectures interpreted by four interpreters (i.e., 88 E-J interpretations) were used. The data statistics are shown in Table 1.

The recorded speech data of both the source speakers and the interpreters were separated into utterance units of 200-millisecond or longer pauses. All utterance units were transcribed manually in compliance with the Corpus of Spontaneous Japanese (CSJ) (Maekawa et al., 2000), and each utterance unit was assigned a start and end time. Spontaneous language phenomena, such as fillers, repetitions, and mistakes, were tagged with discourse tags.

3.1.2. Word-level Translation Alignment

Word-level translation alignment is essential for analyses of missing translations in simultaneous interpretation. The data used in this research include translation alignment at an utterance unit level (Takagi et al., 2002). The analyzed data comprise 14,679 utterance unit level alignments. In addition, word translation correspondences were aligned manually for each aligned utterance unit. Word-level translation alignment was performed according to the following criteria.

- Content words of English speech must be aligned.
- Content words that have no corresponding words in the Japanese interpretation are aligned as "no correspondence".
- Phrases and idioms are aligned as a single correspondence.

Figure 1 shows an example of word-level translation alignment. Words highlighted with the same color in the speaker and interpreter utterances demonstrate translation correspondence. Words that are colored in the speaker utterance that do not have a corresponding color in the interpreter utterance are aligned as "no correspondence". In addition, words not colored are not aligned (not content words).

3.2. Frequency and Proportion of Missing Translations

In this study, content words aligned as "no correspondence" are defined as missing translations in the simultaneous interpretation. As described previously, omissions in interpretations can be classified into different categories. In addition to the four categories defined by Barik (1994), omissions can be classified as conscious strategic omissions, conscious intentional omissions, conscious unintentional omissions, conscious receptive omissions and unconscious omissions (Napier, 2004). For example, interpreters can omit unnecessary words and summarize content to increase interpreter simultaneity in E-J simultaneous interpretation (Tohyama and Matsubara, 2006). Note that determining the type of missing translation and whether a word is unnecessary are subjective operations (Barik, 1994). In addition, it is impossible to classify missing translations automatically. However, to analyze missing translations as defined in previous studies, aligned words that satisfy the following conditions are excluded from the analyses:

- Determiners, existential there words, and prepositions (i.e., not content words).
- Pronouns. In E-J translations, English pronouns are usually omitted to obtain a more natural Japanese translation (Anzai, 2008).
- Words tagged as repetition and corrected mistakes.

The Stanford Parser (The Stanford Natural Language Processing Group, 2002) was used to obtain part-of-speech information.

In the following, non-excluded English words that are aligned are referred to as content words. Note that aligned phrases are considered a single content word.

An example of missing translations and exclusions is shown in Figure 2, and the content words and missing translation statistics for the entire dataset are given in Table 2. Note that the proportion of missing translations in the data is 33.1%, which is considerably greater than Barik's result

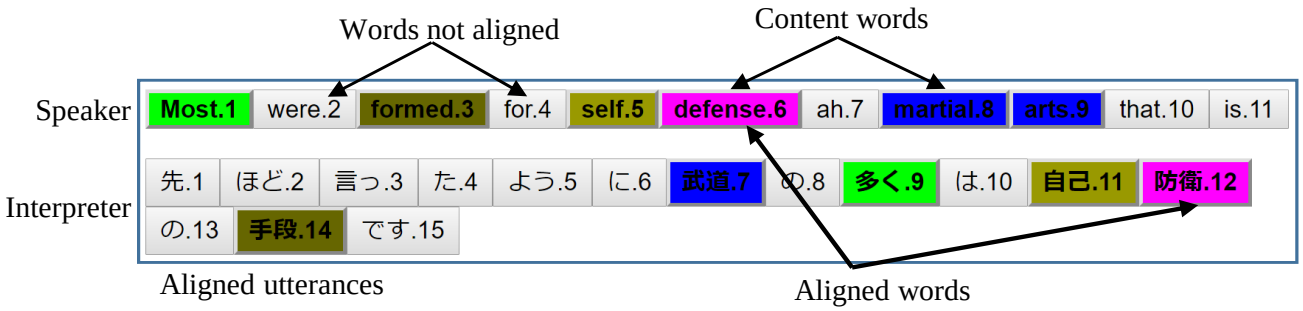


Figure 1: Word-level alignment

# of content words	46,568
# of missing translations	15,431
Proportion of missing translations	33.1%

Table 2: Content words and missing translations

Average rate of speech	3.04
Max rate of speech	16.67
Min rate of speech	0.69

Table 3: Rate of speech statistics (syllables/sec)

(1994). In Barik's study, the proportion of omissions was calculated by dividing the number of omitted words by the total number of words in the given speech data. However, in this study, we only consider content words. Consequently, the denominator is relatively small; thus, the proportion of missing translations becomes relatively large.

4. Analysis

Statistical analyses were performed to investigate the relationship between the occurrence of missing translations and the extent of speech rate, delay, part-of-speech, and a word's depth in the syntactic structure.

4.1. Speech Rate

4.1.1. Calculation of Speech Rate

The rate of an utterance unit was utilized in this study. Here, the rate of the utterance unit is calculated using the provided start and end times of the utterance. The speech rate unit is represented as "syllables/sec." Table 3 shows the rate of speech statistics.

4.1.2. Relationship between Missing Translations and Speech Rate

The speech rates were divided into intervals, such as 1-2 syllables/sec, 2-3 syllables/sec and so on. The numbers of content words and missing translations in all utterance units for each interval were aggregated. The proportions of missing translations to content words were calculated as the proportion of missing translations. Figure 3 shows the results of this analysis. Here, the horizontal axis is the speech rate and the vertical axis is the proportion of missing translations. "1-2" on horizontal axis refers to speech rates greater or equal to 1 syllables/sec and less than 2 syllables/sec, the same to "2-3", "3-4", and so on. Note that only speech rate intervals with greater than 100 content words are shown in Figure 3. As can be seen, the proportion of missing translations increases with an increasing speech rate. The proportion of missing translations is approximately 20% when the speech rate is less than 2 syllables/sec, and when the speech rate is greater than 7 syllables/sec, approximately one-half of the content words are missing in the translation.

	Missing	Non-missing	Total
Low rate (Top 25%)	3,685	7,957	11,642
High rate (Bottom 25%)	4,102	7,540	11,642
Total	7,787	15,497	23,284

Table 4: Cross-tabulation of missing translations and rates of speech

To confirm that the proportion of missing translations of content words at higher speech rates is significantly greater than that at a lower speech rate, all content words were sorted in ascending order of the rate of the utterance unit in which the content word is included, and a chi-squared test was applied to the proportion of missing translations at the top 25% (slow) and bottom 25% (fast) speech rates. The chi-squared test is a statistical hypothesis test used to determine whether data of different categories are independent. To conduct a chi-squared test, data are cross-tabulated. The cross-tabulation of the frequencies of missing translations and non-missing translations of the content words in the bottom and top 25% speech rates is shown in Table 4.

As shown in Table 4, the proportion of missing translations for the lower speech rates is 31.7%, and that for the higher speech rates is 35.2%. By applying the chi-squared to Table 4, a significant difference was found between the proportion of missing translations at higher and lower speech rates (1% significance level). This implies that the proportion of missing translations is significantly higher at high rates of speech than at low rates of speech.

4.2. Delay

4.2.1. Measurement of Delay

In this study, the ear-voice span (EVS) was utilized as delay in simultaneous interpretation. The EVS is defined as the

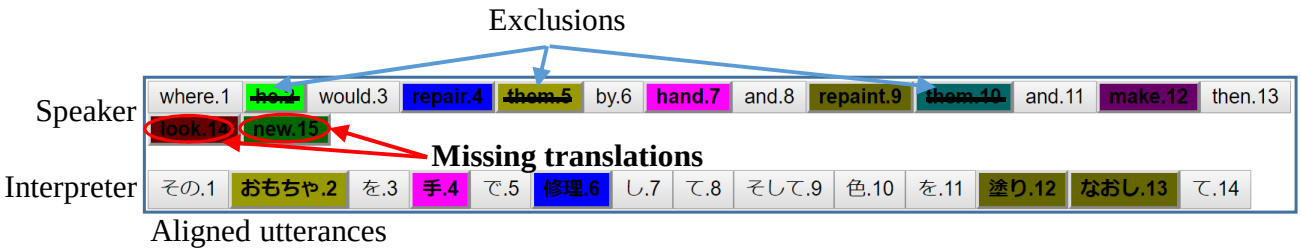


Figure 2: Sample of missing translation

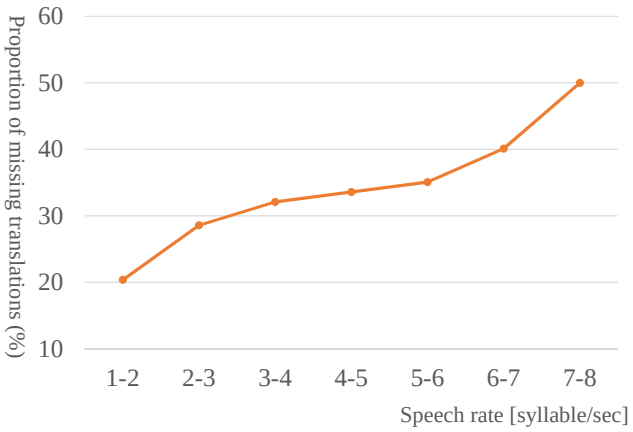


Figure 3: Relationship between speech rate and proportion of missing translations

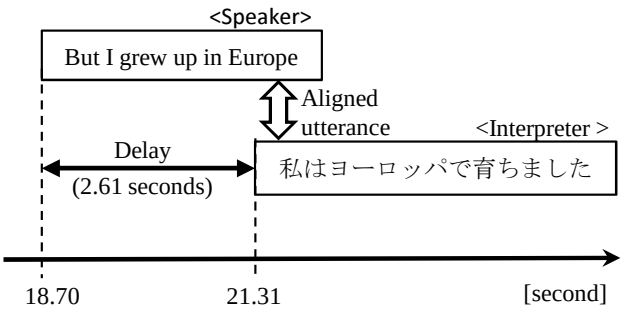


Figure 4: Delay measurement

lag between the start time of the source speaker's utterance and the start time of the interpreter's corresponding utterance (Ono et al., 2008). Figure 4 shows an example of the measurement of delay and Table 5 shows the delay statistics.

4.2.2. Relationship between Missing Translations and Delay

Delays were divided into intervals, such as 0-1 seconds, 1-2 seconds, and so on. The numbers of content words and missing translations in all utterance unit alignments whose delay falls in each delay interval were aggregated and the proportions of missing translations were calculated. Figure 5 shows the results of this analysis. "0-1" on horizontal axis refers to delays greater or equal to 0 second and less than 1 second, the same to "1-2", "2-3", and so on. Note that

Average delay	3.17
Max delay	22.90
Min delay	0.03

Table 5: Delay statistics (seconds)

	Missing	Non-missing	Total
Small delay (Top 25%)	2,070	8,637	10,707
Large delay (Bottom 25%)	4,403	6,304	10,707
Total	6,473	14,941	21,414

Table 6: Cross-tabulation of delay and missing translations

only delay intervals with greater than 100 content words are shown in Figure 5. As can be seen, when the delay is large, the proportion of missing translations becomes large. Approximately 20% of the content words are missed in the translation when the delay is less than 2 seconds. However, when the delay is greater than 10 seconds, 50% of the content words are missed in the translation.

To confirm that the proportion of missing translations with larger delay is significantly greater than that with a smaller delay, all content words not excluded in the analyses were sorted in ascending order of the delay of which utterance unit alignment, and a chi-squared test was applied to determine if there is a significant difference between the proportion of missing translations in the top 25% (small) and bottom 25% (large) delays. The cross-tabulation is shown in Table 6.

As shown, the proportion of missing translations with small delay is 19.3% and that with larger delay is 41.1%. The result of the chi-squared test indicates a significant difference between the proportions of missing translations at large and small delays (1% significance), which implies that, when the delay is large, the proportion of missing translations is significantly greater than when the delay is small.

4.3. Part-of-Speech

4.3.1. Part-of-Speech Information

In this study, part-of-speech information obtained using the Stanford Parser (The Stanford Natural Language Processing Group, 2002) was utilized. Here, if a content word was a phrase, the part-of-speech of the head of that phrase was used to represent the part-of-speech of the entire phrase.

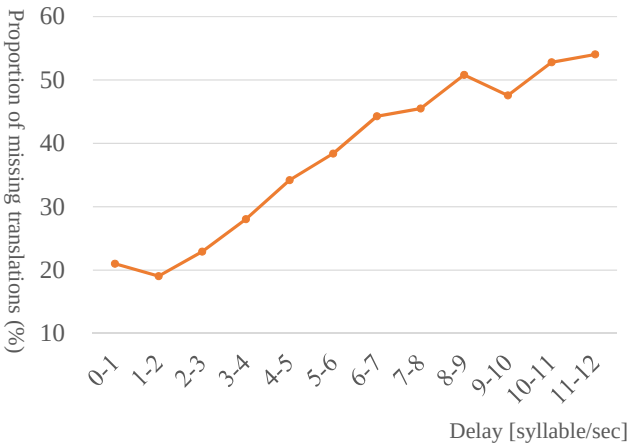


Figure 5: Relationship between delay of interpretation and proportion of missing translation

Example sentence:
You can also go white water rafting on these rivers.

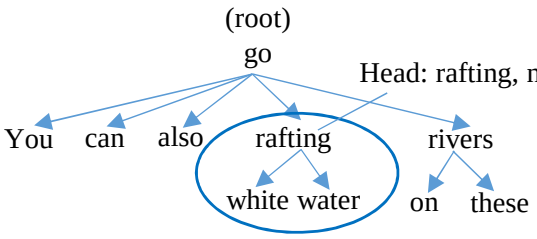


Figure 6: Part-of-speech of a phrase

Figure 6 shows the part-of-speech of a given phrase. Here, the syntactic structure of the sample sentence is shown, and "white water rafting" is aligned as a phrase. The head of this phrase is "rafting," whose part-of-speech is a noun. Thus, the part-of-speech of the whole phrase is considered to be a noun.

4.3.2. Correlation between Missing Translations and Part-of-Speech

Table 7 shows the results of the analysis of missing translations and the part-of-speech. The results indicate that the proportion of adverbs in the missing translations is 26.8% greater than the average proportion of all missing translations (33.1%). On the other hand, the proportion of nouns is 7.9% less than the average, which infers that nouns tend not to be omitted. Among nouns, the proportion of proper nouns in the missing translations is only 14%, approximately 20% less than the average, which is the lowest for all parts-of-speech. In addition, the proportion of missing translations of numbers is 29.1%, which is 4% less than the average. However, names, which are proper nouns, and numbers are regarded as one of the problem triggers (Gile, 1995) and are likely to overload interpreters and be omitted in interpretations. The result of this analysis contradicts the results of previous studies; however, as names and numbers typically play important roles in speech, interpreters may preferentially pay more attention to such information in order to translate them accurately. Another reason is that names that are unfamiliar to the interpreters and complex numbers, which likely increase the workload of interpreters, seldom appear in the speech in the analyzed data. Thus, names and numbers do not trigger problems for interpreters. However, adverbs, which play a modifying role in sentences, similar to adjectives, show a 27.9% greater proportion of missing translations than adjectives.

Part-of-speech	Words	Missing translations	Proportion of missing translation
Noun	5,230	20,742	25.2%
Verb	4,320	12,613	34.3%
Adjective	2,063	6,448	32.0%
Adverb	3,466	5,784	59.9%
Numeral	168	577	29.1%
Others	210	449	46.8%

Table 7: Relationship between part-of-speech and proportion of missing translations

To confirm that the proportions of missing translations are significantly different between each part-of-speech and the overall average, a chi-squared test was applied to the proportions of missing translations of each part-of-speech and the proportions of other content words. The results of the chi-squared test indicate that the proportions of missing translations of nouns, verbs, and adverbs differ significantly from that of the other content words (1% significance level). Note that the differences in the proportions of the missing translations between adjectives, numbers, and other content words are not significant. This implies that the proportion of missing translations of nouns, verbs, and adverbs differ significantly from that of the overall.

4.4. Depth in Syntactic Structure

When the syntactic structure is complex, it becomes difficult for interpreters to understand the information in the source speech and missing translations likely occur. Thus, it can be inferred that, as words are positioned more deeply in the syntactic structure, it is more likely that the given word will be omitted in the translation.

4.4.1. Measurement of Depth in Syntactic Structure

Word depth in the syntactic structure was calculated using a typed dependency representation derived using the Stanford Parser (The Stanford Natural Language Processing Group, 2002). Here, the root word of a sentence is at depth 0, and the number of steps from the root to a given word is considered the depth of that word. While there are several routes from the root to a given word, in this case, the shortest route is chosen. When a content word is aligned as a phrase, the depth of the head of the phrase is considered as the depth of the phrase. Figure 7 shows an example of word depth in a syntactic structure.

4.4.2. Relationship between Missing Translations and Depth in Syntactic Structure

Figure 8 shows the result of this analysis (only depths with greater than 100 content words are shown). The results

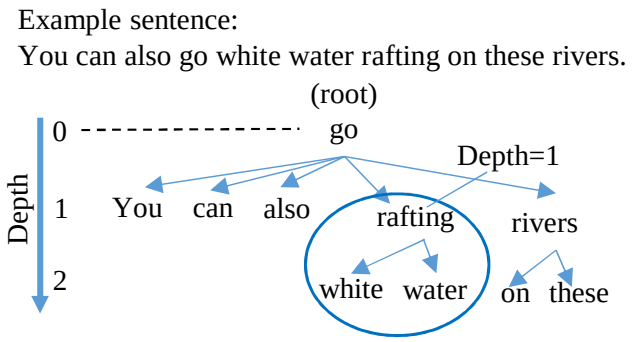


Figure 7: Word depth in syntactic structure

	Missing	Non-missing	Total
Shallow position (Top 25%)	3,775	7,867	11,642
Deep position (Bottom 25%)	4,261	7,381	11,642
Total	8,036	15,248	23,284

Table 8: Cross-tabulation of syntactic position and missing translation

infer that, as a word is positioned deeper in the syntactic structure, the more probable it is that the word will be omitted in the translation. However, words at depth 0, i.e., the roots of the sentences, have a greater proportion of missing translations than those at depths 1 and 2. This also contradicts intuitive expectations because the root word is generally the main word of the sentence. In addition, the proportion of omitted words at a depth of 6 is greater than that of adjacent depths.

A chi-squared test was applied to confirm that the proportion of missing translations of content words at deep positions in the syntactic structure is significantly greater than that at shallower positions.

All content words were sorted in ascending order according to their depth in the syntactic structure, and a chi-squared test was applied to the frequency of missing translations in the top 25% (shallow) and bottom 25% (deep) positions in the syntactic structure. However, some words at depth 1 were included in the top 25%, and some words at depth 3 were included in the bottom 25%. Note that the words used in this test were selected randomly. The cross-tabulation is shown in Table 8.

The proportion of missing translations of the shallow 25% is 32.4% and that of the deep 25% is 36.6%. The chi-squared test results indicate a significant difference between the rate of missing translations of the shallow 25% and that of the deep 25% (1% significance level). This implies that the proportion of missing translations is significantly greater when a word is deeper in the syntactic structure.

5. Conclusion

In this paper, to detect when it is difficult for an interpreter to provide an interpretation, statistical analyses of missing

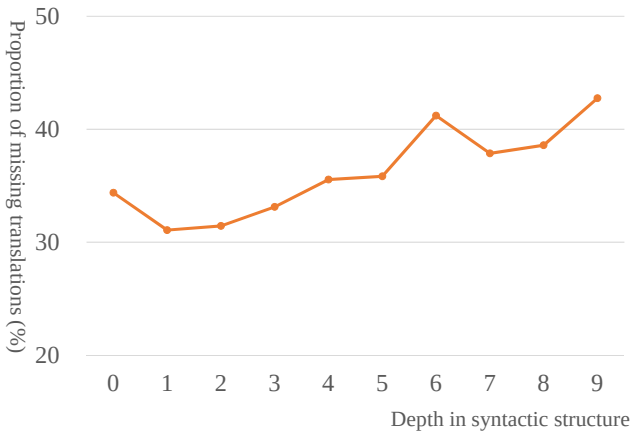


Figure 8: Relationship between word depth in syntactic structure and proportion of missing translations

translations in E-J simultaneous interpretations were described. In this study, 88 lectures from E-J interpretation data from the SIDB were utilized in our analyses, and word-level translation correspondence was manually applied to the corpus. The relationships between missing translations and various factors, i.e., speech rate, delay, part-of-speech, and depth in syntactic structure, were analyzed. The analyses revealed the following relations:

- A significant difference was confirmed between fast and slow speech rates. When the speech rate is high, the proportion of missing translations is also high.
- A significant difference was confirmed between larger and smaller delays. When the delay of an interpretation is large, the proportion of missing translations becomes high.
- Significant differences were confirmed relative to nouns, verbs, and adverbs, and no significant differences were identified relative to adjectives and numbers. The proportion of missing translations relative to adverbs was 59%, which is 26.8% greater than the average. Note that adverbs are most likely to be omitted in translations. In addition, the proportion of missing translations relative to nouns and numbers were 25.2% and 29.1%, respectively. Note that nouns and numbers represent parts-of-speech that are least likely to be omitted in translations.
- A significant difference was confirmed between shallow and deep word positions in the syntactic structure. As words are positioned deeper in the syntactic structure, it becomes more probable that the given word will be omitted in translations. However, the proportion of missing translations of root words was greater than that of words at depths 1 and 2. In addition, the proportion of missing translations of words at depth 6 was greater than that of adjacent depths.

In this paper, it has been proven that missing translations in simultaneous interpretation are related to the rate of speech, delay, part-of-speech, and depth in the syntactic

structure. However, other factors related to missing translations should be considered. In future, to identify difficulties in simultaneous interpretations, the density of the information content and the influence of combined factors will be studied.

6. Acknowledgements

This research was partially supported by the Grant-in-Aid for Scientific Research (B) (No. 26280082) of JSPS.

7. Bibliographical References

Anzai, T. (2008). *Honyaku eibunpou - Yakushikata no rule*. Babel Press (in Japanese).

Barik, H. (1994). A description of various types of omissions, additions, and errors of translation encountered in simultaneous interpretation. *META*, 16:199–210.

Dillinger, M. (1994). Comprehension during interpreting: What do interpreters know that bilinguals don't? *The Interpreters' Newsletter*, 3:41–58.

Gile, D. (1995). *Basic concepts and models for interpreter and translator training*. Amsterdam: John Benjamin.

Gile, D. (1999). Testing the effort models' tightrope hypothesis in simultaneous interpreting - a contribution. *Journal of Linguistics*, 23:153–172.

Gile, D. (2009). *Basic Concepts and Models for Interpreter and Translator Training (Revised edition)*. Amsterdam/Philadelphia: John Benjamins.

Maekawa, K., Koiso, H., Furui, S., and Isahara, H. (2000). Spontaneous speech corpus of Japanese. In *Proceedings of 2nd International Language Resources and Evaluation Conference (LREC'00)*, pages 947–952.

Mizuno, A. (2005). Process model for simultaneous interpreting and working memory. *META*, 50(2):739–794.

Napier, J. (2004). Interpretation omissions: A new perspective. *Interpreting*, 6(2):117–142.

Ono, T., Tohyama, H., and Matsubara, S. (2008). Construction and analysis of word-level time-aligned simultaneous interpretation corpus. In *Proceedings of 6th International Conference on Language Resources and Evaluation (LREC'08)*, pages 3383–3387.

Tohyama, H. and Matsubara, S. (2006). Collection of simultaneous interpreting patterns by using bilingual spoken monologue corpus. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC'06)*, pages 2564–2569.

8. Language Resource References

Matsubara, S. and Takagi, A. and Kawaguchi, N. and Inagaki, Y. (2002). *Bilingual spoken language corpus for simultaneous machine interpretation research*. Proceedings of 3rd International Language Resources and Evaluation Conference (LREC'02).

Takagi, A. and Matsubara, S. and Kawaguchi, N and Inagaki, Y. (2002). *A corpus-based analysis of simultaneous interpretation*. Proceedings of Joint International Conference of 5th Symposium on Natural Language Processing (SNLP-2002) & Oriental COCOSDA Worskshop.

The Stanford Natural Language Processing Group. (2002). *The Stanford Parser: A statistical parser*. https://nlp.stanford.edu/software/lex-parser.shtml, 3.8.0.

SynPaFlex-Corpus: An Expressive French Audiobooks Corpus Dedicated to Expressive Speech Synthesis

Aghilas Sini[1], Damien Lolive[1], Gaëlle Vidal[1], Marie Tahon[3] and Élisabeth Delais-Roussarie[2]
[1]IRISA/Université de Rennes 1, Lannion, France;
[2]UMR 6310 - Laboratoire de Linguistique de Nantes, Nantes, France;
[3] LIUM / Le Mans Université, Le Mans, France.
elisabeth.delais-roussarie@univ-nantes.fr, marie.tahon@univ-lemans.fr
{aghilas.sini, damien.lolive, gaelle.vidal}@irisa.fr

Abstract

This paper presents an expressive French audiobooks corpus containing eighty seven hours of good audio quality speech, recorded by a single amateur speaker reading audiobooks of different literary genres. This corpus departs from existing corpora collected from audiobooks since they usually provide a few hours of mono-genre and multi-speaker speech. The motivation for setting up such a corpus is to explore expressiveness from different perspectives, such as discourse styles, prosody, and pronunciation, and using different levels of analysis (syllable, prosodic and lexical words, prosodic and syntactic phrases, utterance or paragraph). This will allow developing models to better control expressiveness in speech synthesis, and to adapt pronunciation and prosody to specific discourse settings (changes in discourse perspectives, indirect vs. direct styles, etc.). To this end, the corpus has been annotated automatically and provides information as phone labels, phone boundaries, syllables, words or morpho-syntactic tagging. Moreover, a significant part of the corpus has also been annotated manually to encode direct/indirect speech information and emotional content. The corpus is already usable for studies on prosody and TTS purposes and is available to the community.

Keywords: Speech corpus, audiobooks, emotion, expressiveness,

1. Introduction

To build an expressive Text-To-Speech (TTS) system able to read books of different literary genres, using various discourse modes and speaking styles, a corpus that covers all these specificities is required. Usually, corpora built for TTS purposes are less than ten hours long and monospeaker. In addition, the content is carefully controlled to maximize the homogeneity of the synthetic speech.

Long and coherent speech data is very interesting as it gives the possibility of studying voice expressiveness under different situations. Audiobooks are a good example of such data and are valuable for prosody modeling, especially in the field of storytelling. For instance, (Montaño et al., 2013; Montaño Aparicio, 2016) show that expressive categories might exist in the storytelling speaking style. Some works have also been done to detect the speaking style in audiobooks (Székely et al., 2012b) and to evaluate the usability of audiobooks data for the generation of conversational speech (Székely et al., 2012a).

In the last decade, many corpora were built from audiobooks. For instance, (Panayotov et al., 2015) presents a multi-speaker English corpus built for speech-text alignment purposes containing thousands of hours of speech. In (Stan et al., 2013), a multilingual corpus that contains approximately sixty hours of speech data from audiobooks in 14 languages is introduced. Since this corpus contains an average of four hours per language and only one book per language, it prevents studies on speaking styles particularly for TTS synthesis. The same analysis is true for the GV-Lex corpus (Doukhan et al., 2015) which focuses on tales analysis and contains twelve tales but only one hour of speech. In other works such as (Zhao et al., 2006; Wang et al., 2006), authors use an audiobook recorded by a professional speaker, and explore the speech recording in different spaces (expressive, acoustic and perceptual). Nevertheless, none of these works proposes a large monospeaker audiobook corpus in French language including various literary genres.

This work introduces a new corpus of read French speech, suitable to build an expressive speech synthesis or to build robust storytelling prosodic models. It is built in the context of the SynPaFlex[1] research project aiming at improving text-to-speech synthesis. The main motivation to build a corpus of audiobooks is to study a large coherent speech recording, made by a single amateur voice and containing different linguistic, acoustic, phonetic and phonological phenomena. To this end, this new corpus contains an eighty seven hours collection of good quality audiobooks, extracted from Librivox, uttered by one speaker and covering various types of literary genres (novels, short stories, tales, fables, and poems). When reading the books, the speaker has an expressive speaking style and uses personification to make the characters distinguishable. Moreover, whole books or chapters are used, thus enabling to study long term discourse strategies used by a speaker.

To build such large corpora from audiobooks, many techniques have been proposed such as in (Braunschweiler et al., 2010; Boeffard et al., 2012; Mamiya et al., 2013; Charfuelan and Steiner, 2013). In this paper, we use an automatic large-scale speech-to-text aligner for the French language (Cerisara et al., 2009) to perform the segmentation into phones.

The remainder of the paper is organized as follows. Section 2 explains how data used to build the corpus were chosen. In sections 3 and 4 , the different manual and automatic an-

[1]https://synpaflex.irisa.fr/

Literary genre	Duration	Discourses annotation	Expressivity annotation
Novels	80h12m	27h21m	10h59m
Short stories	5h01m	4h08m	2h26m
Tales	1h22m	1h22m	10m
Fables	18m	18m	/
Poems	29m	29m	/
Total	87h23m	33h39m	13h25m

Table 1: Collected data durations and amount of annotated data according to speaking style.

notations included in the corpus are detailed. Finally, Section 5 gives first results about the proposed emotion annotation scheme.

2. Corpus Construction

2.1. Corpus Constraints

Designing a speech corpus requires the definition of some criteria for data selection. As this corpus is built in the framework of the SynPaFlex project, it will be used to study prosody, pronunciation, and also to build expressive speech synthesis models for French. Considering this, we have made the following requirements:

- Availability of a large quantity of data uttered by a single speaker;
- Availability of the corresponding texts;
- Good audio signal quality and homogeneous voice;
- Various discourse styles and literary genres;
- Conveying emotions in speech.

2.2. Data Selection

Investigations were conducted using two main types of sources: audio CD and on-line servers. Because of the lesser accessibility to CD audiobooks, online servers were found to be the most appropriate search areas, even if they sometimes provides data patchy in quality.

After several trials, we concentrated our efforts on the recordings made by a female speaker and available on Librivox, one of the public domain projects that provide audio recordings of book readings on a Web server. In total, eighty seven hours of audio files and the corresponding texts have been collected.

As shown in Table 1, the novel genre represents about 90% of the corpus. Among the list of selected books for this literary genre, there are "Les Misérables" (Victor Hugo), "Madame Bovary" (Gustave Flaubert) and "Les Mystères de Paris" (Eugène Sue).

As the selected audiobooks are read by a non-professional speaker, the acoustic conditions might be different between chapters and books. Some listening evaluations have been done on audio signal in order to identify those of lesser quality, allowing to potentially exclude them from further processes.

3. Automatic Annotations

3.1. Data Preparation

The whole annotation process has been conducted relying on the ROOTS toolkit (Chevelu et al., 2014), that allows storing various types of data in a coherent way using sequences and relations. This toolkit allowed us to incrementally add new information to the corpus.

Once audio data have been selected and the corresponding texts have been collected, a few manual operations have been applied to simplify further processing. Notably, as recordings were performed in different technical and environmental conditions, loudness has been harmonized using the *FreeLCS* tool[2]. Despite of that, audio data acoustic features remain more or less heterogeneous.

As texts were coming from diverse sources, their formats were unified. Then the exact orthographic transcriptions of the readings were achieved by inserting introductions and conclusions the speaker added in the recording, and by placing footnotes and end-of-book notes where they appear in the reading stream.

The next step has been to normalize the texts using rule-based techniques appropriate for the French language, and split them into paragraphs. For the rest of the process, we keep each chapter in a separate file so as to keep long term information accessible.

3.2. Speech Segmentation

The broad phonetic transcription, based on the French subset of Sampa, has been extracted and aligned with the speech signal using JTrans (Cerisara et al., 2009).

To evaluate the accuracy of the phone segmentation, an expert annotator performed a manual validation using Praat (Boersma and Weenink, 2016). Since there is only one speaker, half an hour of the SynPaFlex Corpus was taken into account to evaluate the quality of the phone labels and boundaries. The set of data used for the evaluation task has been selected respecting the proportion of the different literary genres in the corpus.

Results related to the validation are presented in Table 2. We can observe that the Phoneme Error Rate (PER) is low for every literary genres, and the average PER is 6.1%. Concerning the average alignment error, results are reported in the fourth column of Table 2. Globally, on average, the error is 11ms.

As far as errors on label assignment are concerned, they mostly occur on vocalic segments. Most of the deletion observed involve /@/ (83.31%), this phoneme being generally optional in French. The majority of substitutions concern mid vowels (37.04% for the substitution of /E/ by /e/, and 31.04% for /o/ by /O/), these realizations being the result of a specific pronunciation or simply phonetization errors.

As for boundary alignment, in 77.17% of cases, boundaries are misplaced from less than 20ms. In poems, however, errors in alignment are more important: in 35% of the vowels, boundaries have been shifted by more than 20ms. It could be explained by two distinct factors. First, the speech rate is relatively slow in poems (with an average of 5 syllables/s) in comparison to other literary genres where the speech rate

[2]http://freelcs.sourceforge.net/

	Validation subset	PER (%)	average alignment error (ms)
Novels	25m36s	5.8	11.5
Short stories	3m49s	7.1	9.4
Tales	2m47s	0.8	14.3
Fables	1m47s	6.5	12.1
Poems	1m07s	6.3	28.3
Total	35m52s	6.1	11.4

Table 2: Validation results for the segmentation step per literary genre : lengths of the validation subsets, Phoneme Error Rate (PER), and average alignment error.

Unit type	Number
Paragraphs	23 671
Sentences	54 393
Words	799 773
Orthographically distinct words	38 651
Phonemically distinct words	28 734
Non Stop Words	411 210
Syllables	1 154 714
Distinct syllables	8 910
Open	808 503
Closed	346 211
Phonemes	2 613 496
Distinct phonemes	33

Table 3: Amounts of linguistic units in the corpus

is of 6 syllables/s on average. Secondly, the acoustic models used to achieve the automatic segmentation (Cerisara et al., 2009) have been trained on the ESTER2 corpus (Galliano et al., 2009) which is a French radio broadcasts corpus. The resulting models could thus be slightly unadapted for poem reading data.

3.3. Linguistic Information

Additional linguistic information has been added to the corpus, such as syllables and Part-Of-Speech tags using the Stanford parser (Green et al., 2011). Table 3 sums up the content of the corpus in terms of linguistic units. We also plan to include syntactic information in a near future.

3.4. Acoustic and Prosodic Information

The speech signal is stored using a sampling frequency of 44.1kHz. From the signal, we have extracted (i) the energy and 12 mel-frequency cepstral coefficients (MFCC 1-12) which we have added delta and delta-delta coefficients using (Gravier, 2003), (ii) the instantaneous fundamental frequency (F_0) using the ESPS get_f0 method implementing the algorithm presented in (Talkin, 1995), and (iii) pitch-marks using our own software.

Additionally, we have added some prosody related features as the articulation rate (in syllables/s), the speech rate (in syllables/s), and F0 mean/min/max/range (in Hz) at the syllable and word levels.

4. Manual Annotations

Audio tracks corresponding to chapters of different books have also been annotated manually according to prosodic units, characters, emotions, and other events that were heard. The annotation method had first been defined on a small subset of readings, and then tested on audiobook recordings completed by other readers. It was found to be generic enough to render a global perceptive description of the speech. As Table 1 shows, 38% of the whole corpus have been processed manually to provide characters annotation, and 15% - included in those 38% - to describe emotional and prosodic patterns contents.

4.1. Prosodic Patterns

After considering the whole speech data, eight prosodic descriptors were defined, then encoded and assigned by an expert annotator to a large number of audio tracks corresponding to chapters of 18 different books, and defining a 13h25m sub-corpus. As far as possible, labels were assigned according to the perceived prosody, without taking into account the linguistic content. They characterize units which could range in length from a word to several sentences. Seven of them correspond to speech showing the following types of prosodic patterns: QUESTION (interrogative), NOTE, NUANCE, SUSPENSE, RESOLUTION (authority, or imperative), SINGING, and IDLE (no particular prosodic pattern, or declarative). The eighth label, EMOTION, was used to report - but without describing it - the presence of any perceived emotional content.

Let's notice as of now that the tag EXCLAMATION is not listed above. This is because this information can be simply deduced from another level of description: in this corpus, the *Exclamation* pattern was found strictly correlated with the emotional content of *surprise*, which is reported in the emotion labeling level (presented in Section 4.3.). Manual annotation is costly in time and redundancy is not desirable in its process. In the following analysis of the prosodic manual labeling, emotion labels *surprise* will therefore be assimilated to hidden prosodic labels for EXCLAMATION.

Another important point is that, when needed for a more precise description, labels were combined (e.g. Emotion+Question+Nuance). Hence, simply summing the labels duration for each type of prosodic pattern gives a value which exceeds the duration of the sub-corpus.

Among the prosodic parameters, the perceived pitch-curve during voice production takes an important role in assigning the labels. For instance, the NUANCE pattern, which is one of the reading strategy of the speaker, maintains listener's attention. This pattern is characterized melodically by a high pitch at the beginning, then a decrease with modulations, and finally a slight increase when it doesn't end the sentence (see Figure 1).

Table 4 shows total duration for each manual prosodic labels in the 13h25 sub-corpus.

A non-IDLE prosodic tag has been assigned to 68% of the speech. As shown in Table 4, the hidden EXCLAMATION tag is very largely represented (more than 4h42), before the IDLE one (4h21m). The first particular prosodic pattern that comes after is NUANCE (3h58m), then come all the other prosodic patterns that are relatively well represented

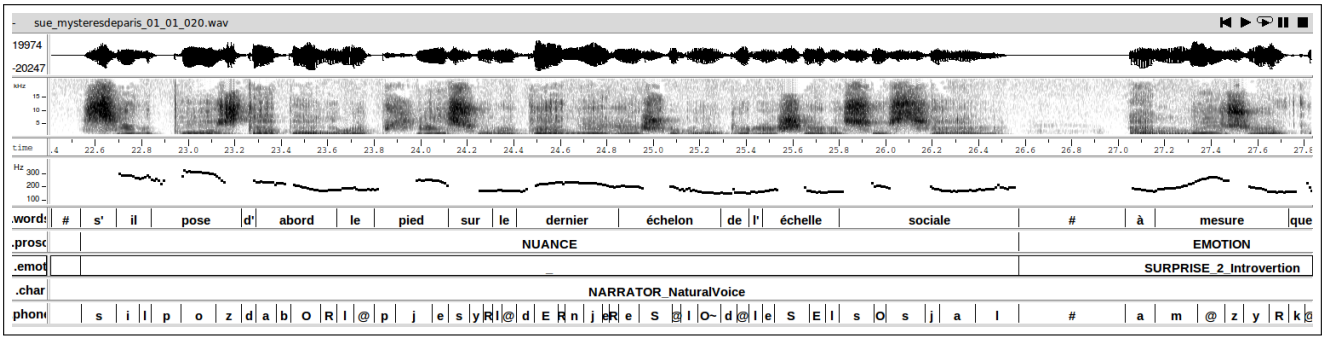


Figure 1: Manual annotations - NUANCE prosodic pattern example

Prosodic label	EXCLAMATION (hidden label)	IDLE	NUANCE	RESOLUTION	SUSPENSE	QUESTION	NOTE
Duration	4h42m	4h21m	3h58m	45m	41m	38m	39m
Sub-corpus %	34.8%	32.2%	29.5%	5.6%	5.1%	4.7%	4.8%

Table 4: Manual annotations - Total duration of prosodic patterns (including combinations) in the 13h25 sub-corpus

and evenly distributed (around 40m): RESOLUTION, SUSPENSE, QUESTION and NOTE. SINGING was found to be exceptional and is not reported here.
More than a half of the speech showing particular prosodic figures is described with combined labels, pointing out where prosody may be more complex.
Most of all, it was found that the EXCLAMATION pattern happens very frequently, especially in narration. In a way, it is an inherent part of the speaker's style.
The generic EMOTION prosodic indicator is assigned to 39% of the whole sub-corpus (5h18m), showing a large amount of emotional data. Its manual description is presented in Section 4.3.

4.2. Characters

The speaker, who is the same for the whole corpus, can personify the different characters of the book by changing her voice or her way of speaking. The character's tags were identified from the text and any turn of speech has been labeled according to the following annotation scheme:

- CHARACTER ID: indicates which character is talking according to the text, and refers to Meta-data where each character is summarily described (name, age, gender, prosody and timbre features). For instance, to personify a gloomy man, the speaker uses a low pitch, low energy and devoiced voice.

- VOCAL PERSONALITY ID: indicates which character is talking according to the vocal personality. Indeed, even if the speaker is very talented and coherent along the books, she can for example forget to change her voice when comes a new character. Therefore, for such speech intervals, the timbre remains the own speaker's timbre or corresponds to another character.

The characters labeling was annotated on more than one third of the whole corpus (33h39m) mined from 18 different books. Dialogue tags were reported as parts of the narrator's speech.
First estimates indicate that one third of the speech content is personified. The average duration for speech turns being of 7s, against 29s for the narrator. In some chapters, direct speech segments can also be very long, typically when a character becomes a narrator who tells his own story.
370 characters were identified, and the full data of their vocal personality labeling indicates a not negligible amount of prosody and vocal tone personification. Covering a wide range and typology, the speaker's voice is thus more or less radically far from her natural style (males, children and elderly people embodiments, psychological singularization, imaginary figures). These vocal personality changes often happen: around 20% of the speech is concerned and, for the half, in stark contrast with the natural speaker's voice.

4.3. Emotions

Different theoretical backgrounds are classically used to identify emotional states, principally based on either distinct emotion categories or affective dimensions (Cowen and Keltner, 2017). Usually, choosing the emotion categories and their number, or the emotion dimensions is an issue.
In the present study, the basic scheme used to manually encode emotions has three items:

- *Emotion category*: Six categories are available, those selected by the Basic Emotions theory (Ekman, 1999): SADNESS, ANGER, FEAR, HAPPINESS, SURPRISE, DISGUST. Two other categories were added to better represent the content of the different books: IRONY and THREAT.

- *Intensity level*: This item, on a scale from 1 to 3, is meant to give a measurement of the experienced emotion intensity according to the speech. For instance, one can interpret its values as follows: SLIGHTLY ANGRY (1), ANGRY (2) , and STRONGLY ANGRY (3).

- *Introversion/Extroversion*: This binary item reflects

Emotion	SURPRISE	SADNESS	JOY	ANGER	DISGUST	FEAR
Effects on the first syllable of focus word(s)	accentuation	disappearance		accentuation	accentuation	
Pitch median	high	low	according to joy type	low	low	low
Pitch curve		flat	flat (suave joy)	flat or top-down	flat or top-down	flat
Rate		slow	according to joy type	fast	fast on focus words	varying with fear intensity
Loudness		low	loud (intense joy)		low	
Timbre changes		breath during the speech	breath during the speech		yes	yes

Table 5: Examples of perceived impacts of emotion on the speech

the way the emotion is rendered through the speech (discreetly, prudently / obtrusively, ostentatiously)

The second and third items may have strong correlations with some of the widely used affective dimensions, as activation and arousal. Furthermore, an important feature of the manual emotion annotation used for the corpus is that the three items labels can be mixed together to provide a more precise description of the perceived emotion. For instance, speech can continuously convey strong and very expressive SADNESS as well as FEAR through some words, which could be tagged as [sadness-3-E + fear-1-E].

Manual emotion labeling was done on the same sub-corpus as for prosody (13h25m). A large amount of emotional content was reported (39% of the speech, including 13% with combined tags). Duration of tagged speech for each category of emotion is given in Table 6, and the number and average duration of labels are indicated in Table 7.

Significant observations have emerged during the annotation. A challenging one is that two radically different types of JOY can be conveyed by the speech, whereas none of the three items could take over their differentiation: on the one hand suave joy, and on other hand elation or gladness. Also, it is suggested that labels should be interpreted in context, notably in conjunction with the discourse mode. In particular, the expressive strategy implemented in the corpus narration is very specific, conveying almost continuously positive valence but in a subtle way, through pitch modulation and with focus words. The SURPRISE label was widely assigned to those recurrent patterns showing (i) a sudden pitch shifting upwards (ii) at least one accentuation onto the first syllable of a focus word (iii) a phonetic elongation or a short silence before this first syllable. Thus, as introduced in section (see 4 1)SURPRISE describes a recurrent emotional attitude of the reader, attracting the listener attention by regularly emphasizing the text.

Other types of variation occur when the speech conveys emotion, some examples are related Table 5 .

4.4. Other Events

Besides acoustic indications of loud noises or music, different unexpected speech events were also reported:

- Linguistic events: for example, the use of foreign languages;
- Phonetic events which are not written in the text: phoneme substitutions, elisions and insertions, high elongations, breaks and pauses, specific voice quality (e.g. whispered voice).

All these features can be of high interest for rendering a more human synthetic voice (Campbell, 2006).

The manual data-sets could provide valuable guidance for further analysis, especially by linking with linguistic information, acoustic measurements, and other descriptions. Examining how manual labels are distributed among literary genres could also be of great interest.

5. Emotion classification

This section presents binary emotion classification experiments conducted on emotional labels of the SynPaFlex corpus. The use of a state of the art methodology aims at positioning our mono-speaker read expressive speech corpus among existing multi-speaker acted or spontaneous emotional speech corpora.

5.1. Data analysis

The manual segmentation and labeling of emotion – which concerns 15% of the whole corpus – results in a total number of 8 751 segments as shown in Table 7. Among them, 5 387 convey an emotional content, while 3 364 do not. To get around the issue of a "neutral" emotion, we decided to label these segments as Idle. As mentioned previously, label combinations were used during the annotation phase to better characterize some expressive content. Consequently, these annotations are considered as new emotional labels which can not be merged with single labels easily. A deeper

Emotion	IDLE	SURPRISE	SADNESS	JOY	ANGER	DISGUST	FEAR	IRONY	THREAT
Duration	8h11m	4h42m	44m	32m	31m	15m	11m	10m	3m
Sub-corpus %	61.0%	35.0%	5.4 %	3.9 %	3.9%	1.9%	1.3%	1.2%	0.4%

Table 6: Manual annotations - Total durations of emotion categories labels (including combinations) in the 13h25 sub-corpus

Emotion	Idle	Anger	Joy	Sadness	Fear	Surprise	Disgust	Other	Comb.	Total
# Seg. manual	3 364	147	115	295	76	2 895	47	23	1 699	8751
Avg. dur (s)	8.76	2.62	2.99	2.67	2.20	3.83	2.26	2.30	3.45	5.55
# Seg. 1 s. max	30 989	447	397	929	199	12 794	125	0	0	45 880

Table 7: Number of manual annotated emotional segments and segments resulting from a 1 s. max chunking. The latest are used in the classification experiments. Other includes IRONY and THREAT labels.

investigation of these label combinations is needed in order to manage them in a speech synthesis system.
Interestingly, the SURPRISE label is highly represented among other single emotional labels. Actually, as described in Section 4 , SURPRISE better corresponds to an emotional attitude of the reader to keep the listener's attention, than an emotion conveyed from the text.
Emotional segments are defined as segments consisting of an homogeneous emotion, be it characterized by single or combined labels. Therefore, there is not constraint on segments' duration. As a consequence, some segments can be very long. For example, one IDLE segment lasts more than 43s. On average (cf Table 7), IDLE segments have the highest durations (8.76s), then comes SURPRISE segments (3.83s.) and COMBINATION labels (3.45s.).

5.2. Methodology

The following experiments aim at classifying the manual annotations with binary emotional models. We know that for multi-speaker acted emotions, classification rates usually reach high performance (for example with corpora such as EMO-DB). However, with multi-speaker spontaneous speech, the classification rates are much lower, thus reflecting the difficulty to discriminate emotions in such a context (Schuller et al., 2009b). The present corpus gives the opportunity to bring a new benchmark of performances on mono-speaker read speech.
To do so, our experimental set up follows a standard classification methodology (Schuller et al., 2009a; Schuller et al., 2013). By this way, our results are comparable with those obtained on other existing emotion corpora. In other words, emotional models are trained in cross-validation conditions (here 5 folds to keep enough data) on acoustic features. 384 acoustic features – 16 Low-Level Descriptors (LLD) $\times$ 12 functionals + Δ – are extracted on emotional segments with OpenSmile toolkit and Interspeech 2009 configuration (Schuller et al., 2009a). To avoid over fitting the data, different subsets of features are tested:

- OS192: 16 LLD $\times$ 12 functionals without Δ
- Δ OS192: 16 LLD $\times$ 12 functionals with Δ only
- OS24: 2 LLD (range + amean) $\times$ 12 functionals without Δ

In order to have homogeneous segment durations, we decided to chunk manual segments every 1 s keeping the remaining part. This operation helps in increasing the amount of data available for the experiment, as reported in Table 7. As aforementioned, COMBINATION labels are not taken into account because merging them with single labels is clearly not obvious. Also, IRONY and THREAT segments are discarded regarding to the small number of labels. To better identify the pairs of labels that can be easily discriminated from those which can not, only binary models are trained thus resulting in an emotion confusion matrix. The number of segments is equally balanced among the two classes.

5.3. Results

Models are trained with Random Forests and entropy criterion. Similar performances were obtained with optimized Support Vector Machines (polynomial kernel, C=1, $\gamma = 0.01$) and normalized features. The results are given as a confusion matrix between emotions as shown in Table 8. On average, performances obtained with the smaller set are the best: 59.9% with OS24, 59.5% with OS192 and 58.8% with ΔOS192. This first observation underlines the importance of selecting features when classifying emotions in such corpora in order to avoid over fitting the data (Tahon and Devillers, 2016).
As we were expecting, the binary emotion classification UAR results range from 43.6% to 81.8%, a typical range for induced and spontaneous speech emotion recognition. These performances also reflect the high diversity of vocal personifications during direct speech as well as different recording conditions. The most impressive classification rates are reached with ΔOS192 for IDLE/ANGER (77.7%) and ANGER/DISGUST (81.8%) emotion pairs. Is seems that the acoustic dynamics captured by this feature subset is very relevant for these two emotion pairs. With Δ features, classification rates drop compared to non-Δ features on other pairs of emotions.

UAR		Ang.	Sad.	Joy	Fea.	Dis.	Sur
OS192	Idl.	**.640**	**.618**	.572	**.638**	.592	.571
	Ang.		.550	**.677**	.563	.572	**.637**
	Sad.			**.610**	.475	.524	**.616**
	Joy				**.636**	**.620**	.573
	Fea.					.584	**.636**
	Dis.						**.600**
ΔOS192	Idl.	**.777**	**.601**	.557	**.650**	.524	.555
	Ang.		.544	.594	.523	**.818**	.584
	Sad.			.621	.525	.436	.566
	Joy				**.638**	.548	.544
	Fea.					.588	**.631**
	Dis.						.532
OS24	Idl.	**.624**	**.621**	.580	**.628**	**.612**	.563
	Ang.		.567	**.671**	.578	.580	**.623**
	Sad.			**.616**	.530	.548	**.631**
	Joy				**.638**	.596	.567
	Fea.					.580	**.633**
	Dis.						.584

Table 8: Unweighted Average Recall (UAR) results for binary emotion classification using the three feature subsets. In bold, UAR > 60%

Regarding the results obtained with the small OS24 feature subset, classification between non emotional (IDLE) and emotional segments is over 60% (bold font in Table 8) for ANGER, SADNESS, FEAR and DISGUST. Two emotion groups emerge from the results:

- IDLE/JOY (58.0%), IDLE/SURPRISE (56.3%) and JOY/SURPRISE (56.7%)
- SADNESS/FEAR (53.0%), SADNESS/DISGUST (54.8%), SADNESS/ANGER (56.7%), FEAR/DISGUST (58.0%), FEAR/ANGER (57.8%) and ANGER/DISGUST (58.0%)

The second group clearly contains negative emotions with different arousal levels.
Further experiments are needed to deeper investigate these groups such as unsupervised clustering, feature selection, etc. For example, Δ features are clearly relevant for ANGER/DISGUST classification. Moreover, emotions are likely to be strongly correlated with direct/indirect speech and also with characters. Additional analyses are required to confirm this observation. The addition of phonological and linguistic information could also help in understanding the emotional distribution of the SynPaFlex corpus.

6. Conclusion and Perspectives

This paper describes a new large speech corpus composed of eighty seven hours of audiobooks from several literary genres read by a single speaker. By being mono-speaker, this corpus can be used to study the strategy of a speaker over entire books. Annotations and speech segmentation into phones are also provided. Among them, we can mention that a manual annotation of emotional contents and characters has been done for respectively 15% and 38% of the whole corpus.
Preliminary emotion classification experiments show that the expressive read speech contained in the SynPaFlex corpus is much closer from spontaneous speech than acted speech. From binary classification results, two emotional groups emerge, one clearly containing negative content. Deeper investigations and analyses of the corpus are planned in a future work: correlation between direct and indirect phrases, emotional speech and characters. Further experiments on feature selection and clustering could also help in investigating the emotional content of this corpus. Moreover, the full corpus has already been used to build a speech synthesis voice, and informal evaluations show that the output of the unit-selection speech synthesis system is relatively good. The corpus is available on our website[3].
This corpus could also be used to study prosodic and phonological aspects of expressive read speech (character personification, speaking styles) and to develop expressive synthesized speech of good audio quality. The amount of data available with the SynPaFlex corpus is consequent enough to allow carrying a deeper analysis of the linguistic, acoustic and prosodic features associated with some of these aspects. Machine Learning techniques such as deep learning can also be used to build prosodic models.

7. Acknowledgements

This study has been realized under the ANR (French National Research Agency) project SynPaFlex ANR-15-CE23-0015 and the LABEX EFL (Empirical Foundations in Linguistics) ANR-10-LABEX-0083.

8. Bibliographical References

Boeffard, O., Charonnat, L., Le Maguer, S., Lolive, D., and Vidal, G. (2012). Towards Fully Automatic Annotation of Audio Books for TTS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 975–980, Istanbul, Turkey.

Boersma, P. and Weenink, D. (2016). Praat: Doing phonetics by computer.[computer program]. version 6.0. 19.

Braunschweiler, N., Gales, M. J., and Buchholz, S. (2010). Lightly supervised recognition for automatic alignment of large coherent speech recordings. In *11th Annual Conference of the International Speech Communication Association (Interspeech 2010)*, Makuhari, Chiba, Japan.

Campbell, N. (2006). Conversational speech synthesis and the need for some laughter. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1171–1178.

Cerisara, C., Mella, O., and Fohr, D. (2009). JTrans, an open-source software for semi-automatic text-to-speech alignment. In *10th Annual Conference of the International Speech Communication Association(Interspeech 2009)*, Brighton, U.K.

Charfuelan, M. and Steiner, I. (2013). Expressive speech synthesis in MARY TTS using audiobook data and emotionML. In *14th Annual Conference of the International Speech Communication Association (Interspeech 2013)*, pages 1564–1568, Lyon, France.

Chevelu, J., Lecorvé, G., and Lolive, D. (2014). ROOTS: a toolkit for easy, fast and consistent processing of large

[3]https://synpaflex.irisa.fr/corpus/

sequential annotated data collections. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykavik, Iceland.

Cowen, A. S. and Keltner, D. (2017). Self-report captures 27 distinct categories of emotion bridged by continuous gradients. *Proceedings of the National Academy of Sciences*, 114(38):E7900–E7909.

Doukhan, D., Rosset, S., Rilliard, A., d'Alessandro, C., and Adda-Decker, M. (2015). The GV-LEx corpus of tales in French: Text and speech corpora enriched with lexical, discourse, structural, phonemic and prosodic annotations. *Language Resources and Evaluation*, 49(3):521–547.

Ekman, P. (1999). Basic emotions. In Dalgleish T. et al., editors, *Handbook of Cognition and Emotion, 1999*.

Galliano, S., Gravier, G., and Chaubard, L. (2009). The ester 2 evaluation campaign for the rich transcription of french radio broadcasts. In *Tenth Annual Conference of the International Speech Communication Association*, pages 2583–2586, Brighton, UK.

Gravier, G. (2003). Spro: "speech signal processing toolkit". *Software available at http://gforge.inria.fr/projects/spro*.

Green, S., De Marneffe, M.-C., Bauer, J., and Manning, C. D. (2011). Multiword Expression Identification with Tree Substitution Grammars: A Parsing tour de force with French. Conference on Empirical Methods in Natural Language Processing (EMNLP'11), Edinburgh, United Kingdom.

Mamiya, Y., Yamagishi, J., Watts, O., Clark, R. A., King, S., and Stan, A. (2013). Lightly supervised GMM VAD to use audiobook for speech synthesiser. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, pages 7987–7991, New Orleans, U.S.A. IEEE.

Montaño Aparicio, R. (2016). Prosodic and Voice Quality Cross-Language Analysis of Storytelling Expressive Categories Oriented to Text-To-Speech Synthesis. *TDX (Tesis Doctorals en Xarxa)*.

Montaño, R., Alías, F., and Ferrer, J. (2013). Prosodic analysis of storytelling discourse modes and narrative situations oriented to text-to-speech synthesis. In *8th ISCA Speech Synthesis Workshop*, pages 171–176, Barcelona, Spain.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an ASR corpus based on public domain audio books. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)*, pages 5206–5210. IEEE.

Schuller, B., Steidl, S., and Batliner, A. (2009a). The interspeech 2009 emotion challenge. In *10th Annual Conference of the International Speech Communication Association (Interspeech 2009)*, Brighton, U.K.

Schuller, B., Vlasenko, B., Eyben, F., Rigoll, G., and Wendemuth, A. (2009b). Acoustic emotion recognition: A benchmark comparison of performances. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pages 552–557. IEEE.

Schuller, B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Müller, C., and Narayanan, S. (2013). Paralinguistics in speech and language—state-of-the-art and the challenge. *Computer Speech & Language*, 27(1):4–39.

Stan, A., Watts, O., Mamiya, Y., Giurgiu, M., Clark, R. A., Yamagishi, J., and King, S. (2013). TUNDRA: a multilingual corpus of found data for TTS research created with light supervision. In *14th Annual Conference of the International Speech Communication Association (Interspeech 2013)*, pages 2331–2335, Lyon, France.

Székely, E., Cabral, J. P., Abou-Zleikha, M., Cahill, P., and Carson-Berndsen, J. (2012a). Evaluating expressive speech synthesis from audiobooks in conversational phrases. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3335–3339, Istanbul, Turkey.

Székely, E., Kane, J., Scherer, S., Gobl, C., and Carson-Berndsen, J. (2012b). Detecting a targeted voice style in an audiobook using voice quality features. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP 2012)*, pages 4593–4596. IEEE.

Tahon, M. and Devillers, L. (2016). Towards a small set of robust acoustic features for emotion recognition: challenges. *IEEE/ACM transactions on audio, speech, and language processing*, 24(1):16–28.

Talkin, D. (1995). A robust algorithm for pitch tracking (rapt). *Speech coding and synthesis*, pages 495–518.

Wang, L., Zhao, Y., Chu, M., Chen, Y., Soong, F., and Cao, Z. (2006). Exploring expressive speech space in an audio-book. *3th International Conference on Speech Prosody*, page 182.

Zhao, Y., Peng, D., Wang, L., Chu, M., Chen, Y., Yu, P., and Guo, J. (2006). Constructing stylistic synthesis databases from audio books. In *7th Annual Conference of the International Speech Communication Association (Interspeech 2006)*, pages 1750–1753, Pittsburgh, Pennsylvania, U.S.A.

Increasing the Accessibility of Time-Aligned Speech Corpora with *Spokes Mix*

Piotr Pęzik
University of Lodz
Pomorska 171/173, 90-236 Lodz, Poland
piotr.pezik@uni.lodz.pl

Abstract

Spokes Mix is an online service providing access to a number of spoken corpora of Polish, including three newly released time-aligned collections of manually transcribed spoken-conversational data. The purpose of this service is two-fold. Firstly, it functions as a programmatic interface to a number of unique collections of conversational Polish and potentially also spoken corpora of other languages, exposing their full content with complete metadata and annotations. Equally important, however, is its second function of increasing the general accessibility of these resources for research on spoken and conversational language by providing a centralized, easy-to-use corpus query engine with a responsive web-based user interface.

Keywords: Speech corpora, corpus search engine, online language services

1. Introduction

High-quality corpora of spoken conversational language are expensive to acquire. Worse still, even when they are collected, annotated and made publicly available, their potential as sources of primary data may remain largely unrealized, due to the lack of search and exploration tools which could deal with the identification of speech-specific linguistic and multimodal phenomena. Apart from their research value, spoken language resources are also indispensable in the development of speech recognition systems and related language technologies. However, in order to sufficiently serve this purpose, they have to be standardized and interoperable enough to be used for both online and offline processing. Spokes Mix is an online service developed as part of the CLARIN-PL infrastructure to address both of these challenges for Polish. Firstly, the search engine of Spokes Mix makes it possible to explore not only the linguistically annotated transcriptions of spoken language, but also some characteristics of the underlying speech signal, such as duration and prosodic contours. These functionalities can be accessed through a web application, which also makes it relatively easy to customize, export and save search results by non-technical users. Secondly, the entire contents of the indexed corpora is available for programmatic search clients and it can exported on demand together with the time-aligned sound files. The present paper briefly describes both the newly and previously available corpora available through Spokes Mix. Next, some of the features of its search engine and API are described to illustrate its accessibility as a language resource service for both technical and casual users. Finally, some of the key planned improvements of Spokes Mix are also briefly outlined.

2. The Data

Spokes Mix provides access to a number of spoken corpora, including three entirely new collections of conversational Polish. To demonstrate and test the ability of Spokes Mix to simultaneously serve speech corpora in multiple languages, it also includes a collection of well-known speech corpora of English. All of these datasets are briefly specified below.

2.1. Annotation and Interoperability

The corpora currently indexed in Spokes Mix have been unified with respect to the varying degrees of bibliographic metadata and sociolinguistic annotation they originally contained. Every transcription has been automatically tokenized into word segments, part-of-speech tagged and time-aligned with the original recording. Spans of transcriptions are explicitly linked to recordings at the level of utterances, words and sound segments. The latter two levels of annotation are provided automatically using the forced-alignment tools developed by (Korżinek et al., 2017). In addition to the JSON (JavaScript Object Notation)-based proprietary data format used to serve search results and export the textual contents of the indexed corpora, popular formats of annotating speech samples are available for selected features of the search engine, such as time-aligned utterances matching corpus queries, which can be downloaded as TextGrid (Boersma, 2006) or .eaf files. Additionally, the entire corpora indexed in Spokes Mix are stored internally as ELAN-encoded files (Wittenburg et al., 2006) and EMU-databases (Cassidy and Harrington, 2001). As indicated below, we are planning to release these versions as self-contained, offline resources.

2.2. Casual-Spoken Data

The PELCRA Conversational corpus is the largest of the Polish collections available through Spokes Mix. It contains just over 218 hours (ca. 2.2 million word tokens) of unplanned, casual conversations recorded in vivo contexts. Released previously through the first version of Spokes (Pęzik, 2015), this corpus has found numerous applications in basic research on spoken Polish, e.g. (Guz, 2015), (Guz, 2017). Among the new features in this edition of the corpus is the word-token level time-alignment and annotation of intonation contours. The original CC-BY-NC license for this corpus extends to this new edition as well.

2.3. Parliamentary Proceedings

As mentioned above, Spokes Mix provides access to three new collections of spoken conversational Polish released under a CC-BY license. The first one of these is a corpus of 50 sampled recordings of parliamentary sessions and com-

mittee meetings, totalling over 12 hours (ca. 101 000 word tokens) of formal spoken Polish. The recordings were manually transcribed with a separate tier for each speaker and force-aligned word segment boundaries.

2.4. Focused Interviews

The second newly released collection of Polish spoken data contains 30 structured interviews (17 hours, ca. 200 000 word tokens) focused around the topic of selected emotions in personal experiences of the interviewed speakers. The interviews are dialogues recorded in stereo with a separate channel for each speaker. The recordings are aligned with the transcriptions both manually at the level of turns and automatically at the level of word tokens.

2.5. Semi-scripted Interviews

The last of the three new collections is still under preparation. Its currently available section contains recordings of 25 interviews with partly scripted questions (15 hours, ca. 165 000 word tokens) covering a variety of everyday topics. Similarly to the previous two collections, the corpus of semi-scripted interviews is released under a CC-BY license. The Polish corpora currently available in Spokes Mix are summarized in Table 1.

Corpus	Register	Hours	Words	License
Conversational	Casual	218	2.2 M	CC-BY--NC
Parliamentary	Formal	12	0.1 M	CC-BY
Focused Intvs.	Mixed	17	0.2 M	CC-BY
Open Intvs.	Mixed	15	0.16 M	CC-BY

Table 1: Summary of Polish speech corpora in Spokes Mix.

2.6. Spoken Corpora of English

The ability of Spokes Mix to serve multilingual speech corpora has been tested on a collection of spoken corpora of English. Currently, users can search, browse and export samples of the spoken subcorpus of the British National Corpus, which was relatively recently time-aligned and released by (Coleman et al., 2012). The service also features a large collection of English read speech in the form of the Librispeech corpus (Panayotov et al., 2015) and an experimental version of the TEDLIUM Corpus (Rousseau et al., 2014).

3. Corpus Search Engine

One of the main objectives of developing Spokes Mix was to provide a corpus search and exploration service which would make its underlying resources accessible for non-technical users. The current implementation of the search engine supports positional and simple lexico-grammatic queries which build upon the syntax described in (Pęzik, 2015). In addition to returning a specified number of contexts matching a query, the engine also calculates aggregated statistics called 'facets' computer on the entire result set and presents them as charts and other visualizations. Full results can be downloaded as Excel spreadsheets, while individual utterances matching the query can be downloaded or played in the web browser interactively using time-aligned word span highlights.

Selected tiers of automatic prosodic annotation can be integrated on demand with spans matching a corpus query. This feature is illustrated in Fig. 1, where the query for the phrase *you know* matches a total of 2720 turns in the indexed corpora. The duration of every matching span is calculated dynamically and presented in a separate column, which can be used as a sorting key. Additionally, pitch codes are presented as a separate piece of prosodic annotation for each matching span. The encoding of pitch and intensity contours is performed using the Momel technique for simplifying contours and the INTSINT scheme (Hirst et al., 2000), (Hirst, 2007) for encoding intonation patterns, with two global codes (T – top, M – middle and B – bottom) representing normalized pitch points and five relative ones (H – higher, U – up-stepped, S – same, D – down-stepped and L – lower).

In addition to presenting such dynamically mapped prosodic annotations for each concordance, it is possible to obtain a number of aggregated views of these data by simply clicking on the `Prosody` tab in the main results window. For example, a requested subset of spans matching the query *you know* can be analyzed to obtain a summary of the distribution of INTSINT codes for this phrase in the underlying corpora of English speech. An example of such a summary is shown in Table 2, which lists the ten most frequent INTSINT code combinations found in a sample of 1000 instances of *you know*. There are relatively few occurrences of this phrase which coincide with local pitch maxima (marked as T), which may be explained by the fact that the semantic bleaching of *you know* in its discourse-marking function results in prosodic and phonetic reduction. Table 3 shows some descriptive statistics calculated for a sample of 1000 duration values of *you know*, which are also displayed in the `Prosody` tab of the search results screen.

Needless to say, such hypotheses about possible correlation of prosodic features and discourse function should be verified by a closer inspection of the underlying data. The link between aggregated statistics and individual data samples is preserved in Table 2 and users can access all instances of a specific code with a single click on a given row. It is even possible to use different features or their combinations to create cluster-like visualisations of prosodically 'similar' instances of spans matching a corpus query. Fig. 2 shows a graph generated by Spokes Mix for combinations of INTSINT codes aligned with matching concordance spans. The red nodes in the graph represent INTSINT codes and they are linked with yellow nodes representing matching utterances. User may click on any of the red nodes to get an instant listing of the utterances containing the corresponding matching span.

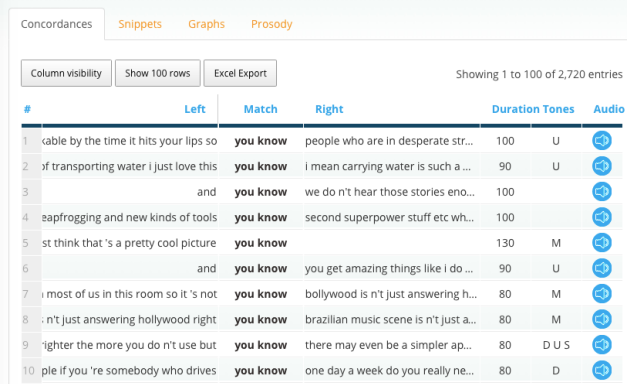


Figure 1: Dynamic mapping of prosodic annotation in corpus search results.

IntSint Code	Frequency
M	130
U	120
H	78
D	78
L	69
S	69
B	67
T	38
M_S	21
U_D	21

Table 2: Top 10 InstSint Codes in a sample of 1000 instances of the phrase *you know* in Spokes Mix.

Statistic	Value
Sample size	1000
Missing values	4
Minimum	59
1st Quartile	80
Median	100
3rd Quartile	114.5
Max	510
Mean	103.04
Variance	1840.70
St. dev	42.90

Table 3: A summary of duration values measured for 1000 occurrences of the phrase *you know* in Spokes Mix.

4. Programmatic Access

The REST (Representational state transfer)-based public API of Spokes Mix can be used not only to selectively search the underlying resources, but also to export their full transcriptions with metadata and sound files. An example response to a concordance query is illustrated in the listings shown below. The same API is used by the Spokes Mix web application, which means that, with a few exceptions, any functionality available through the web-application can

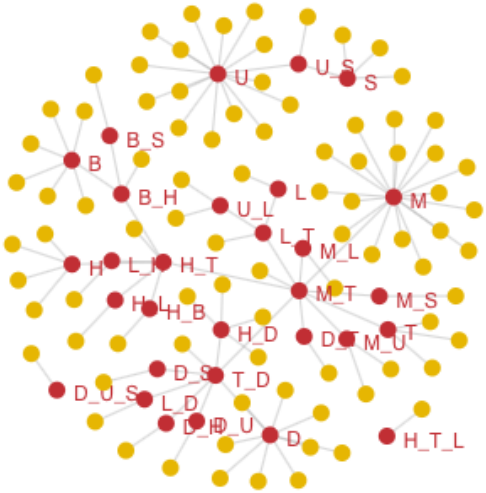


Figure 2: Interactive clusters of IntSint code combinations for 100 occurrences of *you know*.

be requested automatically by a client program using our REST service.


```
{
  "totalHits": 754,
  "docCount": 20,
  "spanCount": 21,
  "spans": [
    {
      "lp": 1,
      "sid": "1_0LDeL",
      "text_id": "sp-pl_Roee",
      "metadata": {
        "id": "1_0LDeL",
        "start": 1862390,
        "stop": 1865010,
        "seq": 647,
        "title_a": "Pogadanki rodzinne",
        "origin": "PELCRA",
        "lang": "pol",
        "source_id": "0LDeL",
        "time_aligned": true,
        "source_corpus_id": "1",
        "speaker_age": 24,
        "speaker_education": "WY",
        "speaker_role": "INTVEE",
        "speaker_sex": "FEMALE",
        "speaker_age_prec": "EXACT"}
      }]
       "matchSpan": [
      {
        "id": "154973893",
        "seq": 84,
        "tag": "adv",
        "lemma": "razem",
        "wordOrig": "razem"
      }
    ],
    "leftTxt": "kumpel z grupy z którym",
    "matchTxt": "razem",
    "rightTxt": "byliśmy w zespole ",
```

```
        "duplicate": false
    },
}
```

The media files can currently be retrieved as 16-bit WAV streams matching an arbitrary section of a recording. They may also correspond to a specific structural unit of transcription such as an utterance, a word token or a span of words matching a corpus query. For example, the timestamps specified in the `start` and `stop` fields of the concordance response shown in the listing above can be used to retrieve the corresponding audio stream for offline use. It is also possible to use an arbitrary offset and retrieve a larger context, which can then be used for manual or automatic re-alignment, further processing and annotation.

5. Planned Developments

We are planning to deposit the three newly released corpora described above in the CLARIN-PL repository as downloadable EMU databases (Cassidy and Harrington, 2001) with metadata descriptions in the CMDI format (Broeder et al., 2012), in order to increase their general visibility and reusability as offline speech databases. More collections of spoken Polish corpora are planned for inclusion as well.

6. Availability

The current version of *Spokes Mix* is available at `http://pelcra.clarin-pl.eu/spokes2-web`. It should be noted that the first stable version of this service is officially planned for June 2018, which is also when its source code will be released. Nevertheless, the current version can be used to test most of the features described in this paper.

7. Acknowledgments

Research and development described in this paper was financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

8. References

Boersma, P. (2006). Praat: doing phonetics by computer. *http://www. praat. org/*.

Broeder, D., Windhouwer, M., Van Uytvanck, D., Goosen, T., and Trippel, T. (2012). CMDI: a component metadata infrastructure. In *Describing LRs with metadata: towards flexibility and interoperability in the documentation of LR workshop programme*, volume 1.

Cassidy, S. and Harrington, J. (2001). Multi-level annotation in the Emu speech database management system. *Speech Communication*, 33(1):61–77.

Coleman, J., Baghai-Ravary, L., Pybus, J., and Grau, S. (2012). Audio BNC: the audio edition of the Spoken British National Corpus. *Phonetics Laboratory, University of Oxford*.

Guz, W. (2015). The structural non-integration of wh-clefts. *English Language & Linguistics*, 19(3):477–503.

Guz, W. (2017). Resumptive pronouns in Polish co-relative clauses. *Journal of Slavic Linguistics*, 25(1):95–130.

Hirst, D., Di Cristo, A., and Espesser, R., (2000). *Levels of Representation and Levels of Analysis for the Description of Intonation Systems*, pages 51–87. Springer Netherlands, Dordrecht.

Hirst, D. J. (2007). A Praat plugin for Momel and INTSINT with improved algorithms for modelling and coding intonation. In *Proceedings of the XVIth International Conference of Phonetic Sciences*, volume 12331236, pages 1223–1236.

Koržinek, D., Marasek, K., Brocki, Ł., and Wołk, K. (2017). Polish read speech corpus for speech tools and services. *Selected papers from the CLARIN Annual Conference 2016, Aix-en-Provence, 26–28 October 2016*, 136:54–62.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an ASR corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.

Pęzik, P. (2015). Spokes – a search and exploration service for conversational corpus data. In *Selected Papers from CLARIN 2014*, Linköping Electronic Conference Proceedings, pages 99–109. Linköping University Electronic Press, Linköpings universitet.

Rousseau, A., Deléglise, P., and Estève, Y. (2014). Enhancing the TEDLIUM corpus with selected data for language modeling and more TED talks. In *Proceedings of LREC*, pages 3935–3939.

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., and Sloetjes, H. (2006). ELAN: a professional framework for multimodality research. In *Proceedings of LREC*, volume 2006, page 5th.

The MonPaGe_HA Database for the Documentation of Spoken French Throughout Adulthood

Cécile Fougeron[1], Véronique Delvaux[2], Lucie Ménard[3], Marina Laganaro[4]
LPP-UMR7018 CNRS/Sorbonne Nouvelle[1], FNRS & UMONS[2], UQAM[3], University of Geneva[4]
Paris, France[1], Mons, Belgique[2], Montreal, Canada[3], Geneva, Switzerland[4]
cecile.fougeron@univ-paris3.fr, veronique.delvaux@umons.ac.be, menard.lucie@uqam.ca, marina.laganaro@unige.ch

Abstract

Our knowledge of life-span changes in the speech of adults is quite sparse. Existing reports are mainly based on English speakers and few studies have compared more than two extreme age groups. The present paper describes the recently constituted MonPaGe_HealthyAdults database of spoken French including 405 male and female speakers aged from 20 to 93 years old. This database aims at documenting speech throughout adulthood and at building a set of reference values for healthy speakers to be used in clinical assessment of speech. The database is built on five age groups ([20-39], [40-49], [50-59], [60-74], [75+]) and includes 4 regiolects. Speakers have been recorded on a variety of linguistic material and speech tasks in order to cover multiple speech dimensions for each speaker. These cross-sectional data form one of the largest French database available for observing typical changes in the speech of adults as a function of age, and especially in older adults.

Keywords: French, speech corpus, aging, lifespan changes

1. Introduction

Understanding potential changes in voice and speech throughout adulthood is critical for clinical research and practice, especially when dealing with neurodegenerative disorders. Data have to be age-standardized in order to tease apart speech characteristics due to age from characteristics due to the speech disorders. It is also crucial for our general understanding of the complexity of the speech production system since age-related changes can as well originate from structural changes at the peripheral level (anatomical and physiological changes in the speech apparatus affecting pulmonary function, laryngeal structure and/or vocal tract length), or from neurological changes affecting speech motor control or cognitive functions (e.g. Linville, 2001; Torre & Barlow, 2009; Seidler et al. 2010, Bilodeau-Mercure et al. 2016).

To date, our knowledge of life-span changes in the speech of adults is quite sparse. Documented in a variety of studies, age-related changes in the production of voice and speech have been reported for (a) voice quality parameters (Ramig & Ringel, 1983; Linville 2001 ; Ferrand, 2002; Shötz 2007) ; (b) pitch/speaking f0, with an increase in mean f0 for older males and a decrease for older females and a greater within-subject variability for both (Benjamin 1986, Morris & Brown 1994 ; Russell et al., 1995 Harnsberger et al., 2008) ; (c) formant frequencies, with a global formant lowering, at least on F1 (Xue & Hao, 2003, Torre & Barlow, 2009), or a trend toward a centralization of formant values (Rastatter et al. 1997) ; (d) speech rate and segment duration, showing the most robust (and documented) age-related differences, with older speakers speaking more slowly, with longer segments, and sometimes longer VOTs, and greater within-subject temporal variability, than younger adults (Ramig 1983, Morris & Brown 1994, Bilodeau-Mecure & Tremblay 2016; Verhoeven et al., 2004; Jacewicz et al., 2009; Staiger et al. 2017).

However, these results are not uncontroversial. For instance, a lack of age effect on vowel space size is found in Fletcher et al. (2015), or on speech rate in Pierce et al. (2013), or on VOT in Smith et al. (1987).

A comparison of these studies is difficult for several reasons. First, if most studies compare only two 'extreme' age groups (usually a group of young adults and a group of old adults) these groups are rarely defined on the same age intervals. Second, the speech material on which the speech dimensions are measured vary from spontaneous speech, read sentences or text, to maximum performance tasks such as diadochokinesic tests. Third, at most two speech dimensions are reported per study, which give a quite minimalist overview of the overall speech characteristics of the speakers included in each group. These methodological differences across studies limit our understanding of the age-related changes in speech, all the more that between-speakers heterogeneity is well known to increase with age.

Furthermore, most of these previous studies are based on the examination of English speakers. Only a few investigations are based on Canadian French speakers (e.g. Bilodeau-Mecure & Tremblay 2016 and other papers from this group) and to our knowledge, no cross-linguistic comparison is available. Knowing that the phonetic implementation of linguistic contrasts and prosodic features are language-dependent, it is conceivable that salient aspects of aging in speech may vary from one language to the other, or from one regional variety (regiolect) to the other. For instance, while both Northern and Southern American English older speakers read more slowly than younger adults, an age-related difference in speech rate is found only for the Northern ones when looking at spontaneous speech (Jacewicz et al. 2009).

The MonPaGe_HA database was collected with two aims. First, in order to further document life-span changes throughout adulthood, it includes speakers with a large range of ages performing various speech tasks involving multiple speech dimensions. Second, the database was built in order to constitute a set of reference values for healthy speakers that will be used for the validation of the MonPaGe speech screening protocol. This protocol was primarily designed for the assessment of French-speaking patients presenting signs of motor speech disorders. In order for the protocol to be usable in a large set of contexts (clinical practice and clinical studies), regional

variation needs to be covered. Therefore, references were seeked for speakers from four French-speaking countries, as described below.

2. The MonPaGe_HA database

The MonPaGe_HealthyAdults (MonPaGe_HA) database is made of audio recordings of 405 French-speaking adults aged from 20 to 93 years old recorded in four French-speaking countries. These data thus form one of the largest French database available for observing variation in the speech of healthy adults as a function of age, gender and regiolect, over a set of controlled speech dimensions and tasks.

2.1 Linguistic material and speech tasks

The speech material collected for each speaker is that of the MonPaGe speech screening protocol (Fougeron et al. 2016). This protocol was primarily designed for a quick (20-30 min recording) although comprehensive assessment of the speech characteristics of patients presenting signs of motor speech disorders. It is therefore conceived to cover multiple aspects of speech and voice and includes several speech tasks. Altogether, the speech protocol is organized in 8 modules, starting with an intelligibility screening (see 2.2.2), for about 6-7 minutes of recorded speech per speaker.

A detailed description of the different modules is presented in the tables at the end of the paper. In short, the speech material in the MonPaGe protocol targets different speech dimensions:

- voice, including voice quality
- articulatory precision in the articulation of both consonants and vowels
- coarticulation patterns (V-to-V, and V-to-C)
- prosodic features (expressive and linguistic prosody)
- speech (and articulation) rate and fluency

Four types of elicitation tasks are included in the material:

- Reading and/or repetition of pseudo-words containing all French consonants and vowels, tested according to their position in the pseudo-word and the level of phonetico-phonological complexity of the sequence (structural patterning, length, planning difficulty).
- Reading of sentences varying in prosody, and of a custom-made 188-word narrative text.
- Automatic production of the days of the week.
- Semi-spontaneous speech in a picture description task.

Speech-like behaviors are also tested in maximum performance tasks, such as:

- Maximum phonation time, where the speaker has to maintain voicing on a sustained vowel as long as possible in a single breath. This is a test of pneumo-phonation control.
- Diadochokinesic tasks (DDK), where the speaker is asked to repeat as fast and accurately as possible successive sequences of syllables.
- Amplitude modulation task, where the speaker is asked to modulate the amplitude of successive calls from the lowest to the highest in 4 steps.

2.2 Speakers and additional ressources

2.2.1 Sociolinguistic questionnaire

Before each recording, speakers had to respond to a questionnaire containing questions related to:

- language background and usage: 1st language(s), other language(s) used daily;
- geographical origin: region of childhood, other region(s) where the speaker had lived for 10 years or more;
- educational background: how old was the speaker when he finished his studies;
- selected medical screening: need and presence of glasses during the session ; self-assessment of hearing abilities (left and right ears on a 10-point scale) ; presence of a denture ; need of speech and language therapy in the past.

2.2.2 Intelligibity and cognitive screening

At the beginning of each recording session, the first part of the MonPaGe protocol consists of a short intelligibility test. The speaker is asked to instruct the experimenter to place some objects on a specific colored shape using a pre-set carrier sentence: “Place the [target word] on the [color] [shape]” (e.g. ‘Place the dog on the red circle’). The speaker (but not the experimenter) sees the target and associated location on the computer screen, and the experimenter needs to place the target he has heard on a sheet of paper containing empty colored shapes. For each speaker, fifteen target words are randomly extracted from a 437 words database. Each of the 437 words has 1 to 6 minimal pairs within the database, which is organized in five subsets of contrasting features (place of articulation, voice, manner, nasality/cluster and vowel). The 15 pseudo-words are extracted from these 5 subsets (3 each). Intelligibility is then scored according to the number of word correctly understood by the experimenter.

Speakers in the oldest group (75+) were also screened for cognitive deficit using MMSE in FR, BE, QC (Folstein et coll., 1975) or language deficits (with the e-GeBAS in CH (Chicherio et al. 2016)

2.3 Speakers distribution in the database

The MonPaGe_HA database includes audio recordings of 209 female and 196 male speakers, aged from 20 to 93 years old. These speakers were recorded in 4 different French-speaking locations: Paris, France (FR), Geneva, Switzerland (CH), Mons, Belgium (BE) and Montreal, Canada (QC). Table I shows the distribution of speakers per age group, sex and regiolect. Figure 1 gives the age distribution of the speakers for each group. Forty-one of these speakers have been recorded a second time with the same protocol 2 to 8 months later. This ‘Re-test’ sub-database includes 18 speakers from CH and 23 from BE, both male and female distributed over the 5 age groups.

All speakers had French as either mother tongue or predominant language. Five speakers had another language than French as first language and 89 speakers were using one or several other languages on a daily basis (English, Italian, Arabic and Spanish being the most frequent).

Speakers had no self-reported speech or voice disorders, although 18 of them reported past speech and language

therapy for a disorder that had been overcome. Seventy-two speakers (from (53 to 93 years old) reported having false teeths.

Speakers were quite representative of the standard population, i.e. the database was not constituted of academics only (or super-seniors). Speakers were recruited by students in their family circles, relatives and/or in retirement homes. An assessment of the educational level of the speaker was obtained by asking at which age the speakers finished school. Distribution was somewhat equivalent across age groups and regiolects. The mean age was 19 years old for the oldest group and 21 to 24 years old for the other groups. Quebec speakers had left school a few years later than the other speakers in average for the 40-49 group (mean: 32 y.o.).

Sex	Regiolect	20-39	40-49	50-59	60-74	75+	*total*
F	FR	12	12	10	10	10	*54*
	CH	11	11	8	9	9	*48*
	BE	11	10	10	11	11	*53*
	QC	10	8	7	18	11	*54*
	total	*44*	*41*	*35*	*48*	*41*	*209*
M	FR	10	9	10	9	9	*47*
	CH	10	10	10	10	9	*49*
	BE	11	10	9	11	10	*51*
	QC	10	6	8	16	9	*49*
	total	*41*	*35*	*37*	*46*	*37*	*196*

Table I: Distribution of the 405 speakers by age groups, sex and regiolects.

2.4 Recording conditions

Speakers were recorded at home in a quite room by trained students. A professional audio material was used for the FR and BE sets (external audiocard Foscurite Scarlet & headmounted Shure SM35-XLR microphone), while various external microphones were used for the other sets. As a consequence, recordings of different audio quality are included in the corpus. This was done on purpose in order to introduce possible noise in the reference values used for the validation of the MonPaGe protocol, which is to be used by speech pathologists in clinical practice, thus in different noise conditions and with a variety of audio equipment.

For the administration of the speech screening protocol, the computerized version of the MonPaGe was used. This application allows the prompting of the speech material/tasks in a set order, as well as the instant recordings of each production as a single audio file, indexed with the speaker's references. Speakers were seated in front of the computer and a trained experimenter administrated the protocol.

3. Acknowledgements

This database could not have been constituted without the help of (a) our colleagues of the MonPaGe projects: Pernon M., Léveque N., Borel S., Pellet P. Bagou O., Trouville R., Catalano S., Lopez U., Kocjancic-Antolik T., Fournet M. (b) and our students: Blanchard C., Baradat Garcia A., Cohen T., Davat H., de Braquillanges A, de Laubier C., Dehon S., Leclerc-Poudrier A., Mabut C., Maysounave M., Miny A., Pochat-Baron M., Rary E., Vidou C., Visentini A.

This work was partially supported by the program "Investissements d'Avenir" ANR-10-LABX-0083 (Labex EFL), the Swiss FNS Grant N. CRSII5_173711 and the Fonds National de la Recherche Scientifique (Belgium).



Figure 1: Age characteristics for each age group, per regiolect and sex

4. References

Benjamin, B. J. (1982). Phonological performance in gerontological speech. Journal of Psycholinguistic Research, 11(2), 159–167.

Bilodeau-Mecure M., Tremblay P. (2016) Age Differences in Sequential Speech Production-Articulatory and Physiological Factors. J Am Geriatr Soc.;64(11):e177-e182.

Chicherio, C., Genoud-Prachex, T., Assal, F., & , Laganaro, M. (2016). Cyber-neuro-psychologie ou l'application des nouvelles technologies à l'évaluation précoce en phase aiguë des patients cérébro-lésés. Journée d'Hiver de la Société de Neuropsychologie de Langue Française, Paris.

Ferrand, C. (2002). Harmonics-to-noise ratio: an index of vocal aging. J Voice 16, 480–487.

Fletcher, A. R., McAuliffe, M. J., Lansford, K. L., & Liss, J. M. (2015). The relationship between speech segment duration and vowel centralization in a group of older speakers. Journal of the Acoustical Society of America, 138, 2132–2139.

Folstein MF, Folstein SE, McHugh PR. « Mini - Mental State ». A practical method for grading the cognitive state of patients for the clinician. J psychiat Res 1975 ; 12: 189 - 98

Harnsberger, J.D., Shrivastav, R., Brown, W., Rothman, H., & Hollien, H. (2008). Speaking rate and

fundamental frequency as speech cues to perceived age. J. Voice 22, 58–69.
Fougeron C., Delvaux V., Pernon M., Léveque N., Borel S., Pellet P. Bagou O., Trouville R. Ménard L., Catalano S., Lopez U., Kocjancic-Antolik T., Laganaro M. (2016) Chapitre 14. MonPaGe : un protocole informatisé d'évaluation de la parole pathologique en langue française. In Actes du colloque UNADREO « Orthophonie et technologies innovantes » (Joyeux N. & Topouzkhanian S., eds).
Jacewicz, E., Fox, R. A., O'Neill, C., & Salmons, J. (2009). Articulation rate across dialect, age, and gender. Language Variation and Change, 21(2), 233–256.
Kahane, J. (1981). Anatomic and physiologic changes in the aging peripheral speech mechanism. In: Beasley, D., Davis, G. (Eds.), Aging: Communication Processes and Disorders. Grune & Stratton, New York, 21–45.
Linville, S. E. (2001). Vocal aging. San Diego, CA: Singular.
Morris, R.J., Brown, W.S. (1994) Age-related differences in speech variability among women. Journal of Communication Disorders 27, 49–64
New, B., Pallier, C., Brysbaert, M., Ferrand, L. (2004) Lexique 2 : A New French Lexical Database. *Behavior Research Methods, Instruments, & Computers*, 36 (3), 516-524. http://www.lexique.org
Pierce, J. E., Cotton, S., & Perry, A. (2013). Alternating and sequential motion rates in older adults. International Journal of Language & Communication Disorders, 48(3), 257–264.
Quené, H. (2008). Multilevel modeling of between-speaker and within-speaker variation in spontaneous speech tempo. JASA 123, 1104–1113.
Ramig, L.A., & Ringel, R.L. (1983). Effects of physiological aging on selected acoustic characteristics of voice. J Speech Hear Res. 26, 22-30.
Ramig, L. A. (1983). Effects of physiological aging on speaking and reading rates. Journal of Communication Disorders, 16(3), 217–226.
Rastatter, M., McGuire, R., Kalinowski, J., Stuart, A. (1997) Formant frequency characteristics of elderly speakers in contextual speech. Folia Phoniatrica et Logopaedica 49, 1–8.
Russell, A., Penny, L., & Pemberton, C. (1995). Speaking fundamental frequency changes over time in women: a longitudinal study. J Speech Hear Res. 38, 101–109.
Seidler, R.D., Bernard, J.A., Burutolu, T.B., Fling, B.W., Gordon, M.T., Gwin, J.T., Kwak,Y., Lipps, D.B. (2010) Motor control and aging: links to age-related brain structural, functional, and biochemical effects. Neurosci. Biobehav. Rev. 34, 721–733.
Staiger A., Schölderle T, Brendel B. Ziegler ,W. (2016) Dissociating oral motor capabilities: Evidence from patients with movement disorders. Neuropsychologia 95 (2017) 40–53
Schötz S. (2007) Acoustic Analysis of Adult Speaker Age. In: Müller C. (eds) Speaker Classification I. Lecture Notes in Computer Science, vol 4343. Springer, Berlin, Heidelberg
Smith, B. L., Wasowicz, J., & Preston, J. (1987). Temporal Characteristics of the Speech of Normal Elderly Adults. Journal of Speech, Language, and Hearing Research, 30(4), 522–529.
Torre, P., & Barlow, J.A. (2009). Age-related changes in acoustic characteristics of adult speech. J. Commun Disord. 42, 324-33.
Verhoeven, J., De Pauw, G., & Kloots, H. (2004). Speech rate in a pluricentric language: A comparison between Dutch in Belgium and the Netherlands. Lang Speech 47, 297–308.
Xue, S., & Hao, G. (2003). Changes in the human vocal tract due to aging and the acoustic correlates of speech production: a pilot study. J Speech Lang Hear Res. 46, 689–701.

Appendix:

Detailed description of the MonPaGe modules and associated speech material. Modules are included in a computerized version of the protocol, which delivers the prompts and instructions to the speaker, then records and stores the productions.

Module 'Intelligibility': see description in 2.2.2

Module 'Pneumophonatory control': this module includes the material for a standard assessment of voice quality on sustained vowels, with one production of a 2-3 seconds /a/. In order to assess maximum phonation time, the speaker is also asked to sustain a /a/ vowel for as long as possible at a comfortable pitch and intensity. Two trials are recorded and the speaker can try as much as he wants. For the two tasks above, audio examples are provided to the speakers.
Then, in order to test whether speakers are able to modulate the intensity of their productions and monitor their own intensity, the speaker is asked to produce the typical call 'ého' [eo] at 4 self-estimated amplitude levels from the lowest to the highest (no example provided since self-monitoring is tested).

Module 'Articulation': this module contains a set of 52 pseudowords used to assess the articulation of all French phonemes, with manipulation of complexity factors: syllable structure, syllable position and syllable frequency. These items are presented in table II. Items are presented to the speakers in orthographic form on a screen and in audio form via headphones in order to allow either reading or repetition of the forms. One production is recorded for each item.
Construction principles of the items are the following:

(a) The 'Peripheral vowels set' includes 3 exemplars of the peripheral oral vowels of French /i, e, ɛ, a, ɔ, o, u/ produced in a /pVpVpV/ frame in order to have the acoustic vowel space of the speaker

(b) The 'Bisyllabic pseudoword set' aims at testing the articulation of almost all the French consonants (/p,b,m,f,v,t,d,n,s,z,l,ʃ,ʒ,k,g,ʀ,j,w/) and vowels (/i, y, e, ø, ɛ, a, ɔ, o, u, ɔ̃ , ɛ̃, ɑ̃/) as well as the production of some clusters in various word positions.

All singleton consonants occur in word onset position followed by V1=/a/ and in word medial onset postion followed by various V2; singleton /t,d,s,z/ and /R/ also occurs in final coda position (see 'CVCVC' set); all French vowels occurs in V2 position.

Clusters of different complexity, CC (/sp, bl, kR/) and CCC (/spl, stR/) occur in both word-onset and word-medial onset positions, they are always associated with the same CV syllable (e.g. /spe.la/-/la.spe/). In order to test for syllable frequency effects, these clusters are associated with specific vowels to form pairs of syllables with high or low frequency according to the LEXIQUE3.01 database (New et al. 2004).

(c) The 'Coarticulation set' aims at testing anticipatory VtoV coarticulation on V1 /a/ according to V2 /a/ or /i/, and VtoC coarticulation on C and CC according to V2 /i/ or /y/.

(d) The 'Long pseudowords set' includes tri and quadrisyllabic words involving alternating articulatory configurations for voicing, place of articulation, mode of articulation and nasality.

(a) peripheral vowels	
Pa,pa,pa	/papapa/
Pou,pou,pou	/pupupu/
Pêp, pêp,pêp	/pɛpɛpɛ/
Pi,pi,pi	/pipipi/
Pop,pop,pop	/pɔpɔpɔ/
Pé,pé,pé	/pepepe/
Pau,pau,pau	/popopo/

(e) Coarticulation	
Laspu	/laspy/
Laspi	/laspi/
Laspa	/laspa/
tessi	/tesi/
tessu	/tesy/
maba	/maba/
mabi	/mabi/

(b) bisyllabic pseudowords		
CV.CV	Tabon	/tabɔ̃/
	Magou	/magu/
	Padan	/padɑ̃/
	Rafau	/Rafo/
	Sajau	/saʒo/
	Bayeu	/bajø/
	Ganain	/ganɛ̃/
	Zassain	/zasɛ̃/
	Yatu	/jaty/
	Ouaneu	/wanø/
CV.CVC	Nazor	/nazɔR/
	Yaouid	/jawid/
	Damette	/damɛt/
	Jaruz	/ʒaRyz/
	Faposse	/fapɔs/
CV.CCV	Laspé	/la.spe/
	Laspi	/la.spi/
	Kablan	/ka.blɑ̃/
	Kablon	/ka.blɔ̃/
CCV.CV	Spéla	/spe.la/
	Spila	/spi.la/
	Blanka	/blɑ̃.ka/
	Blonka	/blɔ̃.ka/
CV.CCCV	Vastra	/va.stRa/
	Vastré	/va.stre/
	Chaspli	/ʃa.spli/
	Chasplé	/ʃa.sple/
CCCV.CV	Stréva	/stre.va/
	Strava	/stRa.va/
	Splicha	/spli.ʃa/
	Splécha	/sple.ʃa/

(d) long pseudowords		
CV.CV.CVC	Dadada	/dadada/
CV.CV.CVC	Tatata	/tatata/
CV.CV.CV.CV	Kitoukitou	/kitukitu/
GV.GV.GV	Oui-oui-oui	/wiwiwi/
CV.GV.CV	Dayaza	/dajaza/
CV.CV.CV.CV	Takadacha	/takadaʃa/
CV.CV.CV	Fichoussu	/fiʃusy/
CV.CV.CV.CV	Ménabainban	/menabɛ̃bɑ̃/

(c) Syllable frequency (occurrence per million)	
/spli/ (234)	/sple/ (.4)
/stRa/ (128)	/stRe/ (35)
/spe/ (546)	/spi/ (70)
/blɑ̃/ (285)	/blɔ̃/ (36)

Table II: Pseudowords in the Module 'Articulation'.

Lundi, le chat, le loup et Papa vont à Bali. Les copains sont tout contents.	*Monday, the cat, the wolf and Daddy go to Bali. The friends are all happy.*
Mardi, Papy y va aussi. Il dit: "Je n'ai pas un sou! Qui va prendre soin de moi?" " Moi!" dit le chat, "moi!" dit le loup. "Vous?", Papy réfléchit.	*Tuesday, Grandpa joins them. He says: "I don't have money! Who will take care of me?" "I will!" says the cat; "I will!" says the wolf. "You two?" Grandpa wonders.*
Mercredi, Papy dit: "Toi, le chat! Tu es doux, tu es chou, tu n'as pas de poux! Mais pas ce loup: il a une cape rouge et je n'aime pas ce gars-là!"	*Wednesday, Grandpa says: "You, cat! You are kind, you are sweet, you don't have lice!" "but not this wolf, he has a red cape and I don't like him!".*
Jeudi, le chat et Papa se baladent à Bali. Papa glisse! Aïe! ouille! Son cou craque, son coude claque, c'est la débâcle!	*Thursday, the cat and Daddy wander in Bali. Daddy slips! Ouch! Ouch! His neck cracks, his elbow creaks, it's a fiasco.*
Vendredi, Papa a mal. Il pleure, il crie. "Toi, Papy, aide-moi, trouve le nain!" "Un nain? On n'en a jamais vu par ici?!"	*Friday, Daddy hurts. He weeps, he cries. "You, Grandpa, help me, find the dwarf!". "A dwarf? We never saw one around here?!".*
Samedi matin, le chat va voir son ami le loup et lui dit: "Aide-moi à soigner Papa!"	*Saturday morning, the cat goes to his friend the wolf and says "help me heal Daddy!".*
Samedi soir le loup lui donne sa recette magique: "Coupe un oignon, cache-le sous la souche, et lorsque le lilas fleurira, Papa sera guéri!" Abracadabra, ça y est, on a réussi!	*Saturday evening the wolf gives him a magic recipe: "cut an onion, hide it under the stump and when the lilac blooms, Daddy will be cured!" Abracadabra, we succeeded!".*
Dimanche, le chat tout doux, le loup magicien, Papa et Papy quittent Bali. Les copains sont tout contents.	*Sunday, the sweet cat, the wizard wolf, Daddy and Grandpa leave Bali. The friends are all happy.*

Table III: A 188-word story to be read by the speakers in the 'Text Reading' module.

Module 'Prosody': this module contains 6 sentences presented in orthographic form to the speakers with specific instructions to test the production of prosodic contrasts. All sentences are fully voiced for subsequent f0 measurements. The production of *assertive/interrogative contrast* is tested on a 4-syllable sentence 'Laurie l'a lu' (Laurie read it) and a 7-syllables sentence 'Mélanie vend du lilas' (Melanie sells lilac). Sentences are first presented on the screen as declarative (e.g. 'Laurie l'a lu.') and then whith a question mark (e.g. 'Laurie l'a lu ?') and the speaker is asked to say the same sentence asking a question.

The use of prosody to express different phrasing is then tested with the sentences 'Anne, Marie et moi allons à la mer' (Anne, Marie and I are going to the see shore) vs. 'Anne-Marie et moi allons à la mer' (here 'Anne-Marie' is a compound first name, so only 2 persons are involved). Expected phrasing are the following: (Anne), (Marie) (et moi) (allons à la mer) vs. (Anne-Marie) (et moi) (allons à la mer)

Module 'Diadochokinesia': DDK tests are often used in clinical practice to test the ability to make alternating movements in quick succession. Items have to be produced in a continuous manner in a single breath group and speakers are asked to speak as fast and as accurately as possible. Seven items, which vary in term of complexity, are used here. They include (a) repetitive CV syllables involving different places of articulation: /bababa/, /dedede/, /gogogo/ (the vowels are chosen to be close to the consonant place of articulation), (b) repetitive CCV syllables, of a more complex structure: /klaklakla/, /tRatRatRa/, (c) sequences of different syllables, either simple CV or complex CCV: /badego/, /klatRa/.

Module 'Days of the week': the speaker is asked to produce in a continuous manner the days of the week (starting from Monday) over a period of 30 seconds. This module aims at testing continuous speech production in an 'automated mode' since the speech material to produce is an overlearned series.

Module 'Text reading': continuous read speech is assessed in this module, in which the speaker has to read a custom-made short story. Successive (groups of) sentences of the text are presented one by one on the computer screen (Table III). The story includes 188 words. It has been especially written to allow the description of the phonetics and phonology of French speakers. Some of the words have been selected to allow comparison between isolated production (in the 'articulation' or 'days of the week' modules) and productions of similar words in a more continuous meaningful production mode (e.g. vowel production, V-to-V coarticulation (papa/papi), complex CC sequences, days of the week…). It also allows the assessment of expressive prosodic functions elicited, which are elicited in the text with many punctuation marks and direct speech. Part of the first sentence ('Les copains sont tout contents') is repeated in the last sentence, to allow comparison in the search for potential fatigue effects.

Module 'Picture description': more spontaneous production is elicited in this last module where speakers are asked to describe the picture presented in figure 2. The picture includes some of the items present in the text reading.

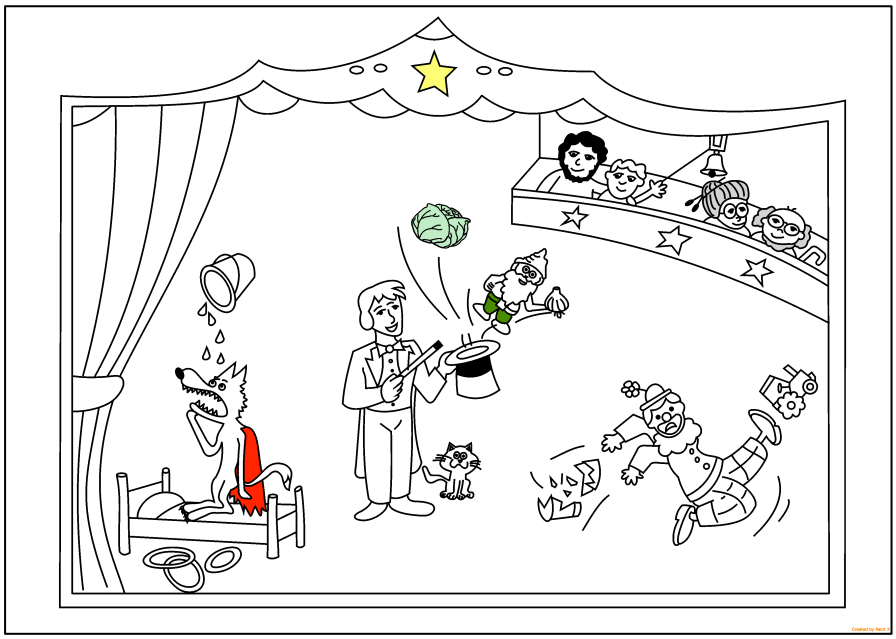


Figure 2: Picture to be described by the speakers in the 'Picture description' module.

Bringing Order to Chaos: A Non-Sequential Approach for Browsing Large Sets of Found Audio Data

Per Fallgren, Zofia Malisz, Jens Edlund
KTH Royal Institute of Technology
perfall@kth.se, malisz@kth.se, edlund@speech.kth.se

Abstract

We present a novel and general approach for fast and efficient non-sequential browsing of sound in large archives that we know little or nothing about, e.g. so called *found data* – data not recorded with the specific purpose to be analysed or used as training data. Our main motivation is to address some of the problems speech and speech technology researchers see when they try to capitalise on the huge quantities of speech data that reside in public archives. Our method is a combination of audio browsing through massively multi-object sound environments and a well-known unsupervised dimensionality reduction algorithm (SOM). We test the process chain on four data sets of different nature (speech, speech and music, farm animals, and farm animals mixed with farm sounds). The methods are shown to combine well, resulting in rapid and readily interpretable observations. Finally, our initial results are demonstrated in prototype software which is freely available.

Keywords: found data, data visualisation, speech archives

1. Introduction

1.1. Found data for speech technology

The availability of usable data becomes ever more important as data-driven methods continue to dominate virtually every field. Numerous organisations (e.g. the Wikimedia Foundation[1], the World Wide Web Consortium[2], and the Open Data Institute[3]) push hard for Open Data. Although language data, and speech data in particular, is riddled with complex legally restricting considerations (Edlund and Gustafson, 2016) and less likely to be "non-privacy-restricted" and "non-confidential" as required of Open Data, the use of data-driven methods in language technology (LT) and speech technology (ST) is nothing less than a modern success story. In the intersection of LT and other fields, such as history and politics, social sciences and health (Gregory and Ell, 2007; Sylwester and Purver, 2015; Zhao et al., 2016; Pestian et al., 2017), traditional data-driven methods play a significant role. Data is arguably yet more crucial in ST, and for decades, funding agencies have spent considerable resources on projects that record speech data. These efforts have been dwarfed by the vast amounts of user data that are being gathered by multinational corporate giants for the betterment of their proprietary technologies.

In contrast, comparable amounts of data are not available to academia and smaller companies. As a result, at least when it comes to the LT and ST tasks that are targeted by the major commercial players, systems developed by smaller entities do not have a chance to compete. This resource gap raises concerns: what happens if the giants decide to charge large sums for their solutions once we have grown accustomed to getting them cheap? How does one conduct research that requires solutions to work on tasks different from those targeted by the giants? And how do we analyse data recorded under entirely different circumstances? As it stands, the truth is that without proprietary solutions, it is difficult to achieve high-quality results.

A pressing question, then, is how can we make sufficiently large and varied speech data sets available for research and development? A stronger focus on collaboration and sharing of new data, in particular data that has been gathered using public resources, is likely to improve matters (Edlund and Gustafson, 2016), as is crowdsourcing. Another solution is *found data* – data not recorded for purposes of ST research – and in particular speech found in public archives. Data from public archives ticks many boxes for speech and ST research: there are great quantities of data to be found, in near endless supply. In Sweden alone, the two largest archives (ISOF and KB) host 13000 hours and 7 million hours of digitised audio and video recordings (with a current yearly growth well over half a million hours), respectively. Additionally, the data comes from a wide range of situations and time periods, making up a longitudinal record of speech. And though the speech archives are routinely disregarded by archive researchers for practical reasons – listening through speech is simply too time consuming and cumbersome – focus on better access to the data will generate new research far beyond speech research and technology (Berg et al., 2016).

This type of data is the rule rather than the exception in LT. People are rarely asked to generate text in order to create data. In ST, the reverse holds: creating data from scratch is commonplace. The main reason is that speech is so variable. Speech analysis has often been deemed intractable without controlling for variables such as situation, task, room acoustics, microphone, speaker (dialects, native language, even personality type). Current speech analysis methods are by-and-large created for known, relatively clean speech data. Archive data is notoriously noisy and unpredictable. In the majority of cases, the unknowns include not only the hardware used or the recording environment but also *what* was actually recorded. This is likely to cause huge problems for standard speech analysis methods. Although current commercial ASR performs impressively on

[1] https://en.wikipedia.org/wiki/Wikidata
[2] https://www.w3.org/
[3] https://theodi.org/

the kind of data it was trained on, it rapidly deteriorates if it encounters something as mundane as simultaneous speech from more than one speaker. In phonetics, vowels are often analysed by extracting their formants, but this process is notoriously sensitive to noisy data (De Wet et al., 2004). Even a simple analysis, such as the division of speech data into speech and silence is currently done using methods that are either very sensitive to noise, or rely on special hardware setups at capture-time (e.g. multiple microphones on smartphones).

1.2. Speech technology for found data

We are facing a Catch-22: we need data to improve ST, and better ST to get at the data. Without automatic analyses, the sheer size of the data becomes an obstacle rather than an asset. The 13000 hours of digitised recordings available at ISOF would take one full time listener 1625 8-hour work days just to get through the data. With a 5-day week, and no vacations, this comes to 6.35 years. We have then allotted zero time for taking notes or creating summaries. If we instead consider the 7 million hours available at KB, we are looking at 3 365 person years – no holidays included. As a first step, we need a robust method to build an impression of what the contents of any given large, unknown set of recordings might be.

There are different ways to alleviate the situation. Using some intelligent sampling technique, we could listen through a 1 percent sample of the ISOF data in just over 3 weeks of continuous listening. The sampling would have to be very smart, however, for 1 percent to give good and representative insights, and without prior knowledge of the data, smart sampling is a hard task.

We suggest that by combining suitable automatic data mining techniques with novel methods for acoustic visualisation and audio browsing, we can provide entry points to these large and tangled sets of data. The proposal includes humans in the analysis loop, but to an extent that is kept as low and efficient as possible.

We have devised a listening method *Massively Multi-component Audio Environments* and a proof-of-concept implementation *Cocktail* (Edlund et al., 2010). A large number (100+) of short sound snippets are played near-simultaneously, while new snippets are added as the old ones play out. The snippets are separated in space and listened to in stereo. The technique gives a strong impression what the snippets are in a very short time. Proof-of-concept studies showed that listeners could identify proportions of sounds (e.g. a 40/60 gender division to the left, and a 60/40 to the right) quickly and accurately. The method allows us to make quick statements about large quantities of sound data. However, it is less efficient if we know nothing of the data (the distribution in space will be random). For full effect, we need to organise snippets in some non-random order.

A number of data mining techniques organise high-dimensional data in low-dimensional spaces. Typical examples include the popular t-Distributed Stochastic Neighbor Embedding (t-SNE) (van der Maaten and Hinton, 2008) and the largely forgotten Self Organizing Map (SOM) (Kohonen, 1982). In an elegant online demonstration that inspired this work, Google AI Experiments visualise bird sounds on a 2-dimensional map[4] using t-SNE. SOMs have also been used for sound. In (Kohonen et al., 1996), the authors discuss the application of SOMs to speech. In line with ST praxis, they recommend using *cepstrum* features for speech, but they also point to single fast Fourier transforms as an efficient feature extraction method. (Kohonen et al., 1996) goes on to propose a system for speech recognition that uses SOMs to create what they refer to as *quasi-phonemes*, and uses these as input to a Hidden Markov Model decoder. More recently, (Sacha et al., 2015) used SOMs to analyse pitch contours. (Thill et al., 2008) used SOMs and a clustering algorithm to visualise a large set of dialectal pronunciation and lexical data. Their approach is related to ours. Namely, their aim was to create a 'visual data mining environment' in which the analyst is interactively involved and can explore a large number of variables relevant to a sociolinguist: geographic and social correlates of linguistic structures. One of the key characteristic of SOMs, but not of t-SNE, is that SOMs tend to preserve the topological properties of the input space. For this reason, it is a great alternative for preliminary exploration of data with many features. Our solution, then, is to conduct an experiment very similar to Google AI Experiments' bird visualisation, but with the aim to distribute audio snippets that are not necessarily known in 2-dimensional space and use this distribution as input to a multi-component environment for audio browsing purposes.

2. Method

2.1. Data

We are primarily interested in speech data. Found data, however, may contain anything, and for our first explorative investigation, we put together several data sets representing a variety of characteristics. Two of the data sets contain speech, and two contain animal sounds. Of each pair, one set is more or less clean, while the other has other material mixed in.

The first speech set is taken from the Waxholm corpus (Bertenstam et al., 1995), which consists of simple Swedish phrases captured in a human-machine context. The corpus was recorded in a studio-like setting, and the audio quality is largely good. The second dataset containing speech was recorded for this work, in a calm office environment, using a standard Samsung Galaxy S6 as the capture device. The recording is done in one take, and contains (1) of a male voice speaking in English, (2) acoustic guitar audio on its own and (3) a segment of both voice and guitar sounding simultaneously. The data sets of animal sounds [5] consists of independent recordings of birds, cows, sheep, and a lengthy recording of mixed farm sounds (with very few animals, and more wind, engines, and such). These four sessions are recorded in different environments using different capture devices.

For the first animal data set, we withhold the farm sounds, to see the results applied to three distinct animal classes. We created a second animal data set by including the mixed

[4] https://experiments.withgoogle.com/ai/bird-sounds

[5] Downloaded from https://freesound.org/

farm sounds as well, to get a handle on the effect of adding more heterogeneous data.
See Table 1 for further details regarding the audio datasets.

Audio Label	Duration	Sample Rate	Segments	Source
Waxholm Men	279sec	16K	2748	Waxholm Corpus
Waxholm Women	294sec	16K	2949	Waxholm Corpus
Spring Birds	131sec	44.1K	1315	freesound.org
Cow & Calf	62sec	48K	620	freesound.org
Sheep & Lamb	119sec	48K	1195	freesound.org
Farm Noise	337sec	48K	3379	freesound.org
Male Speech	34sec	44.1K	336	Phone Recording
Guitar	25sec	44.1K	251	Phone Recording
Speech & Guitar	31sec	44.1K	307	Phone Recording

Table 1: Specifications of audio datasets used in this study.

2.2. Process

We kept preprocessing to an absolute minimum, in order to not make any assumptions at this stage. Each data set was used to produce a spectrogram (i.e. a visual representation of a fast Fourier transform) using the Sound EXchange Library[6] (SOX), a standard library used to handle sounds. SOX seamlessly handles varying frame rates and compression formats, which allowed us to avoid making decisions that may affect the data. The command line: `sox soundfile -n spectrogram -l -r -m -y Yresolution -X pixelpersec -o specfile` generates 8-bit greyscale spectrograms with a frequency range from 0 to 22000 Hz divided into 64 pixels along the Y-axis, and a temporal resolution of 1000 pixels per second along the X-axis. This format was used for all datasets.
The spectrogram and the corresponding audio recording were then split into equal-sized frames. For the purposes of this paper, a frame width of 100ms was used throughout, giving each spectrogram a height of 64 pixels, a width of 100 pixels, and a depth of 256 shades.
The spectrogram frames were used to distribute the sound snippets into hypothetically coherent regions in 2D space, where similar things are closer to each other and dissimilar things more apart. For this training we used a SOM implementation in TensorFlow(Abadi et al., 2015), in which the greyscale pixel values of the generated spectrograms are treated as input vectors to the algorithm.
The output is a set of 2D coordinates that are mapped to the audio segments. Each SOM was trained with 200 iterations over a grid. The size of the grid changed depending on the number of data points in each studied dataset, with a minimum of 30x30. Note that we do not attempt to do clustering on the output of the algorithm.
The resulting plot is amenable to sound browsing. In our implementation, the framework generates a visible grid (see Figure 1) where each datapoint is linked to its corresponding audio snippet. The corresponding audio snippets are played when the cursor hovers over a given datapoint, so listeners can hover over different regions and listen to hypothetically similar data in quick succession or simultaneously (Edlund et al., 2010)[7].

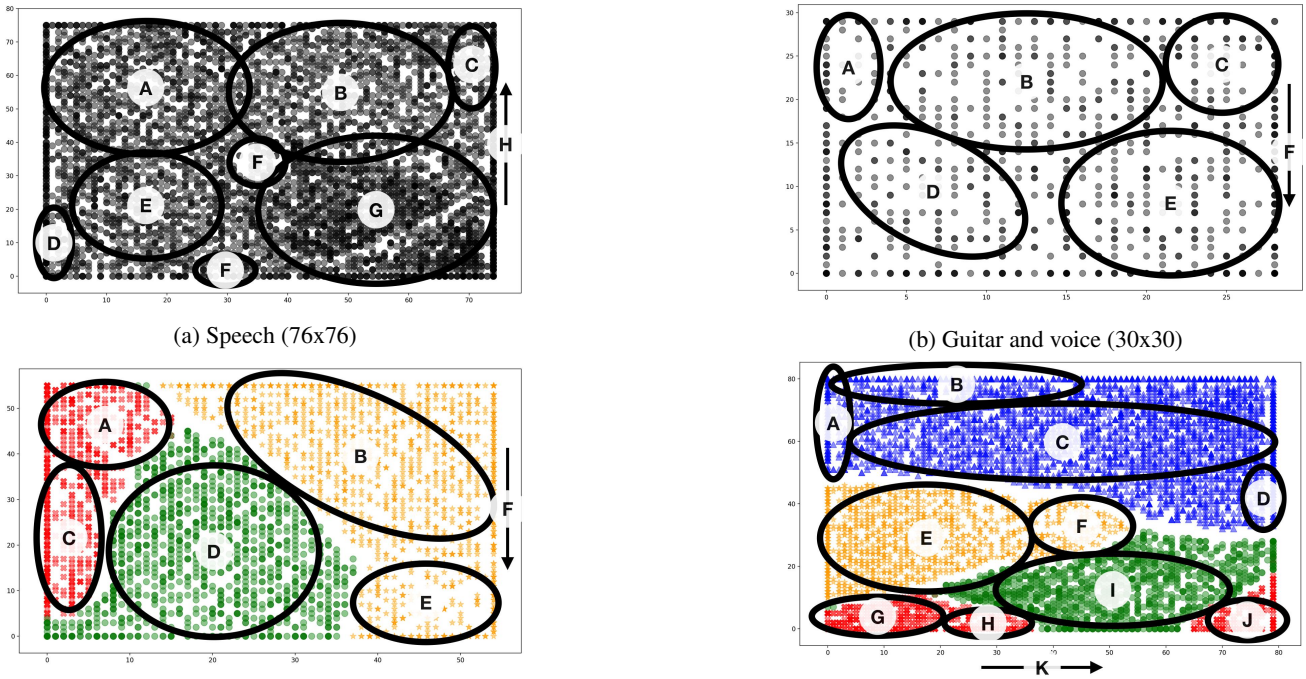
[6] http://sox.sourceforge.net/

[7] The framework code is available for download[8]

For purposes of exploration, we used the interactive plots to point out regions where it was possible to make a clear judgment of (the majority) of the snippets, quickly and with little effort. These human judgments constitute the last step in our current process chain.

3. Results

The panels in Figure 1 represent the results of our method, applied to four data sets. In each illustration, each data point (spectrogram) has been plotted as a circle with an opacity of 0.5, producing a darker effect where more than one data point is positioned in the same grid space. This visualises the SOMs. The areas marked with a black outline represent areas where audio browsing of sounds close to each other in the SOM give a clear impression of the sound's characteristics. Each such area is labeled, and the labels are used as references in the Discussion.
Figure 1a presents the visualisation of the studio recorded speech. There are no obvious clusters or regions discernible to the naked eye, except a darker area divided in two in the lower right corner (labeled G). Figure 1c show three clear regions that were coloured manually according to which recording session the point is associated with. Each sessions contains one animal type only, and is recorded on a separate occasion. The SOM training has resulted in three regions separated by empty grid cells (white regions) corresponding well to the three sessions present in the data. In Figure 1d, we see five regions separated by strings of empty cells. Apart from the blue (triangle) region, that represents the added data, the other data is identical to that of the previous figure. We see a separation of two regions of the red (cross) class, that is not present in the previous figure1c. Finally, in Figure 1b, we have a similar situation as in 1a: there are no obvious clusters. Instead, the data points seem evenly distributed over the grid.
The black outlines represent areas for which listeners reported that they could hear identifiable characteristics that separated the area from the surrounding areas with ease. The labels should read roughly as follows:
1a A: Vowels, resembles voice, can hear gender. high volume; 1a B: transitions from fricatives to vowels; 1a C: fricatives; 1a D: fricatives and quiet consonants; 1a E: short, truncated vowels; 1a F: sharp, non-human click; 1a G: silence/weak noises; 1a H: the arrow represents and overall increase in intensity.
1b A: Consonants. Sometimes alone, sometimes with guitar; 1b B: all voice; 1b C: very quiet, basically silence; 1b D: all guitar; 1b E: intensity generally increases top to bottom.
1c A: Very quiet, weak cow sounds; 1c B: calm soar of bird chirping; 1c C: loud cow sounds; 1c D: only sheep, loud at bottom and weaker sounds at top of region; 1c E: high-pitched, specific and loud bird chirping; 1c F: intensity generally increases top to bottom.
1d A: Pigs snorting; 1d B: roosters crowing; 1d C: windy farm noise, no animal sounds; 1d D: pig snort, clicking sound from farmer; 1d E: calm soar of bird chirping; 1d F: high-pitched, specific and loud bird chirping; 1d G: very quiet, weak cow sounds; 1d H: loud cow sounds, natural transition to sheep; 1d I: only sheep, weak sounds to the



(a) Speech (76x76)

(b) Guitar and voice (30x30)

(c) Animals (56x56)

(d) Animals with farm noise (81x81)

Figure 1: *SOMs based on different sound recordings. The colour-coded informationwas not visible to, nor derived from the process, but added manually for purposes of illustration: (red=birds; green=sheep; yellow=cows, blue=farm sounds). The circled and labeled areas represent manual selections of perceptually clearly similar sounds, based on audio browsing.*

left and louder to the right; 1d J: loud cow sounds; 1d K: intensity generally increases left to right.

4. Discussion & future work

Although we have only taken first steps towards combining dimensionality reduction and visualisation techniques with novel audio browsing techniques, our first results are quite promising. For the speech only data in 1a, a listener can quickly point out areas that are silent, that mainly contain vowels, and several other typical speech features. The next step here is to use the data in these relatively straightforward areas to train models. The silence, for example, will let us model silence in the recording, which will make it possible to segment the data on silence - something that is not easily done in many recordings without spending an inordinate amount of time labeling silent segments sequentially and manually. The vowels may likewise be used to train a vowel model, and separate vowels from other sound of high intensity. We may also find oddities: the tapping noise in F turns out to be the press of a space bar, upon closer inspection of the original data. It turns out that the recorded individuals were told to tap the space bar between each utterance in this particular recording. For the guitar+voice data (1b), we quickly find vicinities with nothing but voice and nothing but guitar. Again, this information can be used to create models or to inform a second clustering, effectively creating a reinforcement learning setup. For animals (1c and 1d), we see that sounds that differ in a distinct manner indeed end up further apart. At this stage, we cannot tell whether it is the recording conditions or the animal noises, or both, that have the greatest influence, yet it is clear that the method we propose would work fairly well to separate different (but unknown) datasets.

Our main goal in this work is to find a way into large sets of unknown data, and so far, we are encouraged by the results. The strength of the proposed method lies in its ability to generalise over different kinds of audio data. As such it has an advantage in the context of large collections of found data to methods that are restricted to only cover a particular sound event. With that said, it should be noted that we are aware of many of the general improvements that can be made to our process, but most if not all of them carry with them a certain amount of assumptions about the data. For our purposes, we think it will be more fruitful to focus on developing the the audio browsing techniques first. Our next step will be to create a more robust listening environment. Listening to audio spatially distributed audio snippets is surprisingly efficient, but we must find out how listeners can best navigate to and point to different regions in the soundscapes. With these methods in place, we can perform full-scale tests on the perception of non-sequentially structured audio data. In the longer perspective, our goal is to add several possible last steps to the process chain. An obvious goal is to make the process iterative. The continuous influx of rapidly acquired human judgments to the learning process is highly interesting. More specific process chains are of equal interest. The silence modeling mentioned above is one such possibility. We are further interested in taking the listener annotations, or judgments, and returning to the original sequential sounds. Simply labeling the frames with their inverted distance (in 2D-space) to the centre of some human label and displaying that curve above the diagonal sequential sound may be quite informative, showing roughly how much speech, silence, cows, or

guitars some sequence contains. From this we get crude labels for each sound segment that can be used in a number of applications, e.g. for search or as training data for supervised machine learning tasks.

5. Acknowledgements

This work is funded in full by Riksbankens Jubileumsfond (SAF16-0917: 1). Its results will be made more widely accessible through the infrastructure supported by SWE-CLARIN (Swedish Research Council 2013-02003).

6. Bibliographical References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Berg, J., Domeij, R., Edlund, J., Eriksson, G., House, D., Malisz, Z., Nylund Skog, S., and Öqvist, J. (2016). Tilltal – making cultural heritage accessible for speech research. In *CLARIN Annual Conference 2016, Aix-en-Provence, France*.

Bertenstam, J., Mats, B., Carlson, R., Elenius, K., Granström, B., Gustafson, J., Hunnicutt, S., Högberg, J., Lindell, R., Neovius, L., Nord, L., de Serpa-Leitao, A., and Ström, N. (1995). Spoken dialogue data collected in the Waxholm project. *STL-QPSR*, (49-74).

De Wet, F., Weber, K., Boves, L., Cranen, B., Bengio, S., and Bourlard, H. (2004). Evaluation of formant-like features on an automatic vowel classification task. *The Journal of the Acoustical Society of America*, 116(3):1781–1792.

Edlund, J. and Gustafson, J. (2016). Hidden resources - strategies to acquire and exploit potential spoken language resources in national archives. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Edlund, J., Gustafson, J., and Beskow, J. (2010). Cocktail–a demonstration of massively multi-component audio environments for illustration and analysis. *SLTC 2010*, page 23.

Gregory, I. N. and Ell, P. S. (2007). *Historical GIS: technologies, methodologies, and scholarship*, volume 39. Cambridge University Press.

Kohonen, T., Oja, E., Simula, O., Visa, A., and Kangas, J. (1996). Engineering applications of the self-organizing map. *Proceedings of the IEEE*, 84(10):1358–1384.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69.

Pestian, J. P., Sorter, M., Connolly, B., Bretonnel Cohen, K., McCullumsmith, C., Gee, J. T., Morency, L.-P., Scherer, S., and Rohlfs, L. (2017). A machine learning approach to identifying the thought markers of suicidal subjects: a prospective multicenter trial. *Suicide and life-threatening behavior*, 47(1):112–121.

Sacha, D., Asano, Y., Rohrdantz, C., Hamborg, F., Keim, D., Brau, B., and Butt, M. (2015). Self organizing maps for the visual analysis of pitch contours. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, number 109, pages 181–189. Linköping University Electronic Press.

Sylwester, K. and Purver, M. (2015). Twitter language use reflects psychological differences between democrats and republicans. *PloS one*, 10(9):e0137422.

Thill, J.-C., Kretzschmar, W. A., Casas, I., and Yao, X. (2008). Detecting geographic associations in english dialect features in north america within a visual data mining environment integrating self-organizing maps. *Self-organising maps: applications in geographic information science*, pages 87–106.

van der Maaten, L. and Hinton, G. E. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Zhao, R., Zhou, A., and Mao, K. (2016). Automatic detection of cyberbullying on social networks based on bullying features. In *Proceedings of the 17th International Conference on Distributed Computing and Networking*, ICDCN '16, pages 43:1–43:6, New York, NY, USA. ACM.

CoLoSS: Cognitive Load Corpus with Speech and Performance Data from a Symbol-Digit Dual-Task

Robert Herms[1], **Maria Wirzberger**[2], **Maximilian Eibl**[1], **Günter Daniel Rey**[2]
[1]Media Informatics, Chemnitz University of Technology, Germany
[2]Psychology of Learning with Digital Media, Chemnitz University of Technology, Germany
{robert.herms, maximilian.eibl}@cs.tu-chemnitz.de

Abstract

In this paper, a new corpus named CoLoSS (Cognitive Load by Speech and performance data in a Symbol-digit dual-task) is presented, which contains speech under cognitive load recorded in a learning task scenario. In order to obtain a reference for cognitive load, a dual-task approach was applied, including a visual-motor primary task that required subjects to learn abstract symbol combinations and an auditory-verbal secondary task to measure the load imposed by the primary task. We report the methodology of collecting the speech recordings, constructing the corpus and describe the properties of the data. Finally, effects of cognitive load on prosodic as well as voice quality features are investigated in conjunction with the corpus. In its current version, the corpus is available to the scientific community, e.g., for exploring the influence of cognitive load on speech or conducting experiments for speech-based cognitive load recognition.

Keywords: speech corpus, speech features, cognitive load, learning performance, dual-task

1. Introduction

The human cognitive system is characterized by capacity limitations in information processing (Plass et al., 2010). They refer to human working memory, which provides temporary storage and manipulation of information (Baddeley, 1992). Cognitive load is generally considered as the load imposed on an individual's working memory by a particular (learning) task (Paas and Van Merriënboer, 1994). According to the cognitive load theory (Sweller et al., 2011), the degree of cognitive load influences the amount and complexity of learned content (Paas et al., 2003).

Speech databases that include audio recordings of speakers under varying levels of cognitive load are rather rare and often created for own research purposes. Different task designs that were developed to investigate the limitations of human working memory in conjunction with speech parameters can be found in the literature; for instance, reading-comprehension (Yin et al., 2007), Stroop interference (Yap et al., 2010), arithmetic abilities (Gorovoy et al., 2010), and driving under cognitive load (Boril et al., 2010). Moreover, the Cognitive load with Speech and EGG (CLSE) database was created by (Yap, 2012), which includes speech recordings of subjects participating in three different tasks: Stroop test with time pressure, Stroop test with dual-task, and reading span task. According to the *INTERSPEECH 2014 Computational Paralinguistics Challenge* (Schuller et al., 2014), the partitioned form of the CLSE database is available for research purposes. Nevertheless, there is still no speech-based corpus available for either the consideration of cognitive load in a learning context or a more sensitive approach to the traditional classification problem.

In this paper, CoLoSS (**C**ognitive **L**oad by **S**peech and performance data in a **S**ymbol-digit dual-task) is introduced—a new corpus that includes speech under cognitive load recorded in a learning task scenario. Compared to existing works in the literature concerning speech-based cognitive load discrimination, the CoLoSS corpus differs in two key aspects: (1) It focuses on cognitive load induced by learning processes. (2) Numeric labels are provided as reference for cognitive load.

The fundamental goal of this work is to encourage scientists in the field of speech technologies to explore the effects of cognitive load (caused by learning) on speech and to provide the basis for regression and/or classification experiments for automatic speech-based cognitive load recognition. The corpus material will be available to the scientific community including audio files, annotations, and labels.

This paper is organized as follows: In the next section we introduce the CoLoSS corpus, including task design, recording conditions, data labelling, and data description. In Section 3, effects of cognitive load on prosodic as well as voice quality features are investigated in conjunction with the introduced corpus. Finally, we conclude this paper in Section 4 and give some future directions.

2. CoLoSS Corpus

The CoLoSS corpus represents a subset of data collected for the experimental study of (Wirzberger et al., 2017b) in which the task design and performance measures were defined. In the following, the task design, cognitive load indicators, recording conditions, and the data of the corpus will be described in detail.

2.1. Task Design

The main goal of this task design was to assess the residual cognitive resources of subjects while they were performing a learning task. For this purpose, a dual-task paradigm was applied: a visual-motor primary task involving the assignment of symbol combinations to a single symbol, while simultaneously memorizing a sequence of five digits from an auditory-verbal secondary task. Symbol assignments of the primary task reflected knowledge schemata that had to be formed across the trials. Inspired by (Wirzberger et al., 2017a), performance measures of the secondary task were considered as reference for cognitive load associated with the primary task.

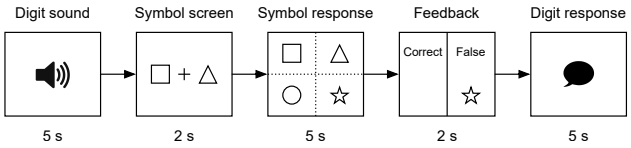


Figure 1: Schematic representation of the dual-task method applied for the CoLoSS corpus. Primary task: step two to three; secondary task: step one and five.

The task comprised a set of 64 trials and was presented to the participants using a desktop computer. Participants were guided in each trial by different screens as depicted in Figure 1. Each trial consisted of the following five steps: (1) Digit sound: a random sequence of five digits in the range of 1 to 9 (in random order) was generated by a text-to-speech system (in German); (2) Symbol screen: one out of four randomly chosen combinations of abstract geometrical symbols was displayed where the order of the symbols must be considered; (3) Symbol response: one out of four possible symbols in a randomly arranged 2×2 grid had to be selected via mouse click; (4) Feedback: feedback was obtained, accompanied by the correct symbol in case of false responses to foster correct schema acquisition; (5) Digit response: the verbal recall of the five digit sequence of step 1 in correct order was requested.

Additionally, task difficulties varied between subjects, but not within the task, by the number of symbols displayed on the screen in step 2. At this point, a distinction was made between an easy and a difficult condition by two and three symbols, respectively.

With reference to the cognitive load theory (Sweller et al., 2011), this task design is associated to various assumptions: Intrinsic cognitive load is represented in the described framework by the number of symbols used to form the combinations. Extraneous cognitive load is represented by the embedded secondary task requirements. Finally, the overall cognitive load, including germane load, is reflected in performance measures of the secondary task.

2.2. Chosen Performance Measures

In order to obtain sensitive measures of the subjects' performance concerning the primary and secondary task, an efficiency score was computed using the likelihood model approach after (Hoffman and Schraw, 2010). The calculation based upon the ratio between performance and effort, whereby performance is represented by the accuracy of problem solving and effort is represented by the time required.

For primary task efficiency, performance was obtained by symbol response correctness while reaction time needed to select a symbol constitutes the effort component. Note, reaction time was related to the visual stimulus regarding the appearance of the 2×2 grid in the *symbol response* stage.

For secondary task efficiency, the performance component was defined by the word accuracy of subjects' responses regarding the five-digit sequence in the *digit response* stage. More precisely, substituted, inserted and deleted words were considered to calculate the word error rate (WER)—the common evaluation measure for automatic speech recognition systems. The word accuracy was then computed by WA = 1 − WER. Since the number of words in the reference added up to five and negative accuracy values were set to zero, the following values could be obtained by parameter WA: 0, 0.2, 0.4, 0.6, 0.8, and 1.0. The effort component of the secondary task was determined by the verbal response duration, i.e., the time starting from the presentation of the visual stimulus (speech bubble) to the end of the last uttered digit. In addition to the actual utterance duration of the subject, the verbal response duration includes indeed the onset latency, i.e., the reaction time from the stimulus to the onset of the first uttered digit. This time span reflects complex cognitive processing for mentally representing the message, selecting words, and retrieving syntactic and phonetic properties; moreover, motor processing for articulation is required.

2.3. Recording and Postprocessing

In total, 123 German students from the Chemnitz University of Technology (Germany) participated in the task. Speech was recorded using a mono clip-on microphone at a sampling frequency of 48 kHz and a 24 bit resolution via a mobile recording device (Roland R-88). Each recording session refered to a particular subject who performed the learning task across 64 trials. A recording session lasted about 20 minutes. Afterwards, the audio segments of the uttered five-digit sequence within the *digit response* stage were extracted using time-codes (5 seconds + 0.5 seconds tolerance) from the task log-data.

The data of 28 subjects had to be excluded for different reasons (lack of sufficient working memory capacity, lack of confirmation for data sharing, lack in language proficiency, or violation of instructions). Furthermore, the speech corpus was restricted by excluding audio segments due to manifold reasons: a segment contains only silence; a segment does not include at least one digit; a segment contains disturbing noise while speaking, for example, caused by unintended gesticulation. In order to provide enough data per subject for various investigations, only subjects with at least 75% of valid audio segments were included in the corpus.

In order to determine the verbal response duration (Section 2.2), audio segments were annotated by two student assistants using time markers in the software Audacity (Team, 2012). This process involved to omit any sound including uttered content after the end of the last uttered digit. Afterwards, all duration values were double checked by another student assistant.

The secondary task efficiency, introduced in Section 2.2, constitutes a promising cognitive load indicator and, consequently, provides the basis for data labelling (Section 2.4). Since the audio data of the corpus were partly contaminated with information on the verbal response duration and thus partly with the secondary task efficiency, audio segments were further processed by trimming (see Figure 2). In more detail, energy threshold based audio activity detection was applied on the speech signal to obtain the onset of the first activity and the end of the last activity. The audio activity refers to any sound which can be caused by speech, breathing, filled pauses, lip-smacking, and so forth. Subsequently,

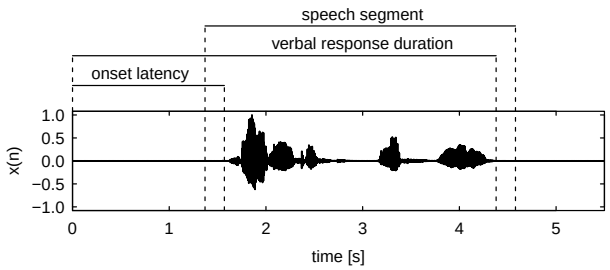


Figure 2: Onset latency, verbal response duration and the resulting segment of a recorded speech signal.

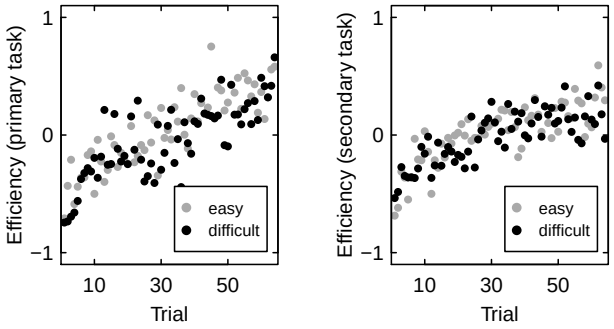


Figure 3: Standardized primary and secondary task efficiency over trials, averaged across subjects for each task difficulty.

a tolerance of 200 ms was added to the segmental boundaries of the activity detector to ensure that information on speakers' activity was not lost by detection errors. If the length of the tolerance exceeded the limits of the original audio segment, as much silence as needed was added to fill the 200 ms at the beginning and/or the end. The resulting segments were then transcoded to 16 kHz with a 16 bit resolution in mono WAV, which constitutes the audio format of the corpus.

2.4. Cognitive Load Labels

As pointed out in Section 2.1, performance measures of the secondary task can be used as a reliable and valid reference for cognitive load associated with the primary task. Furthermore, the study of Wirzberger et al. (Wirzberger et al., 2017b)—the basis for the CoLoSS corpus—backed up the hypothesis that as learning progresses with the sequence of trials, the subjects' efficiency increases concerning the primary as well as secondary task. Hence, the variable of interest for data labelling comprises the secondary task efficiency, which considers performance (word accuracy) as well as effort (time required). Again, this label assignment is linked to the following assumed relationship: Secondary task efficiency reflects the amount of the speaker's cognitive resources devoted to performing the secondary task. The higher the load imposed by cognitive learning processes in terms of the primary task, the lower the efficiency score of the secondary task.

A second variant of cognitive load labels was realized by performing a discretization of the numeric values. For instance, in this way, classification models can be trained for the automatic assessment of cognitive load as an alternative approach to the regression problem. The labels were transformed into nominal values with three distinct cognitive load levels by equal-width binning. Note, this method is an experimental approach for another representation of the original labels. The appropriate separation of numeric indicators into cognitive load classes constitutes an open issue for future work.

2.5. Data Description

Statistics of the constructed corpus are given in Table 1. It includes 70 native speakers of German, whereby 18 are male and 52 are female (9 male and 26 female per task difficulty). Due to the exclusion of some speech files (cf. Section 2.3), the number of instances varies across subjects (min = 48, max = 64, μ = 58.23, σ = 18.41). In the following, the corpus material is given at a glance:

- Audio files (WAV, mono, 16 kHz, 16 bit) containing German speech (digits) from secondary task trials
- Subject id, trial id, and information about the gender.
- Primary task condition assignment (easy and difficult)
- Primary task performance measures (symbol response correctness, reaction time, efficiency)
- Secondary task performance measures (word accuracy, verbal response duration, efficiency)
- Cognitive load labels (secondary task efficiency as numeric and nominal values)

The progression in primary task and secondary task efficiency over trials, averaged across subjects, is illustrated in Figure 3. For a deeper analysis, linear mixed-effects models with z-standardized predictors were used to consider individual effects (condition, interaction between trial and condition, and efficiency across trials per subject). Results confirm an increasing performance across the primary task, β = .273, p < .001, RMSE = 1.003, R^2 = .178, as well as the secondary task, β = .186, p < .001, RMSE = 0.983, R^2 = .478. Neither for the primary task nor the secondary task significant differences between conditions were observed. The RMSE was obtained from a leave-one-subject-out cross validation approach, whereas the R^2 resulted from a Pseudo-R^2 procedure, taking into account random effects in linear mixed-effect models.

Since the efficiency score of the secondary task was suggested for data labeling, the underlying parameters are described in more detail: Regarding the word accuracy (WA), in almost all cases, the response of the five-digit sequence was error free (WA = 1) with a frequency of 3,927 whereas the lowest frequency of 2 occurrs at WA = 0. Such confirms that the secondary task is rather simple so that it does not tend to distract subjects from working on the primary task. Considering all WA values, a mean of 0.94 and standard deviation of 0.15 is obtained. For the verbal response duration, the skewness is 0.81 and the kurtosis is 1.41 indicating that the distribution (min = 0.83, max = 5.5,

	Condition		
Description	easy	difficult	all
Number of subjects	35 (9 M, 26 F)	35 (9 M, 26 F)	70 (18 M, 52 F)
Number of instances	1,993	2,083	4,076
Average number of instances per subject	56.94	59.51	58.23
Average duration per instance [s]	2.68	2.65	2.66
Total duration [hh:mm]	01:29	01:32	03:01

Table 1: Data description of the CoLoSS corpus.

$\mu = 2.85$, $\sigma = 0.68$) is slightly skewed to the right with heavier tails and a sharper peaks than the normal distribution. Similar characteristics are given by the distribution of the efficiency scores (min = 0, max = 0.86, $\mu = 0.35$, $\sigma = 0.11$) where the skewness is 0.21 and the kurtosis is 1.15.

As described in Section 2.4, a discretized version of the numeric labels is included in the corpus. With respect to the assumptions concerning secondary task efficiency (Eff_{ST}) and by involving all conditions, the following three cognitive load (CL) classes were obtained:

$$\text{CL}(\text{Eff}_{\text{ST}}) = \begin{cases} L_1 & \text{for} \quad 0.58 < \text{Eff}_{\text{ST}} \leq 0.86 \\ L_2 & \text{for} \quad 0.29 < \text{Eff}_{\text{ST}} < 0.58 \\ L_3 & \text{for} \quad 0 \leq \text{Eff}_{\text{ST}} < 0.29 \end{cases}$$

where L_1, L_2, and L_3 represent the low, medium, and high cognitive load level, respectively. Note—for clarity reasons—the shown ranges of Eff_{ST} values are rounded to two decimals; the actual values are more accurate. The resulting distribution among classes is highly unbalanced (L_1: 109, L_2: 3.051, and L_3: 916). Therefore, it is strongly recommended to apply resampling techniques before classification models are trained.

3. Effects of Cognitive Load on Speech

Effects of cognitive load on speech were investigated by analyzing means and 95% confidence intervals of six different parameters under three different cognitive load levels (L_1, L_2, and L_3; cf. Section 2.5). Significance across these levels was tested using post-hoc pairwise t-tests with Bonferroni-Holm correction (Holm, 1979), following analyses of variance (ANOVAs) with a significance-level of $\alpha = .05$.

3.1. Feature Extraction

In this section, six common speech-related parameters, that were investigated in conjunction with the CoLoSS corpus, are introduced. Two phoneme-based as well as two acoustic prosodic features and two voice quality features were extracted:

- *Articulation rate*: This rate describes the tempo in speech using the total number of syllables divided by the total duration of the utterance excluding silent pause duration.
- *Silent pause duration*: The total duration of silent pauses within an utterance is determined; it can be an indicator for disfluency in speech.
- *Intensity*: This parameter was computed by the root mean square energy of a signal and can be understood as the acoustic equivalent to the perceptual quantity loudness.
- F_0: The fundamental frequency F_0 represents the frequency of the vocal fold vibration and can be regarded as the acoustic equivalent to the perceptual unit pitch.
- *Jitter* and *shimmer*: Both parameters are the most common descriptors that characterize the voice quality. While jitter is defined as the period-to-period variation in vocal fold frequency, shimmer refers to the period-to-period variation in the amplitude of a voice.

The intensity, F_0, jitter, and shimmer were determined using the analysis tool Praat (Boersma and others, 2002). Afterwards, means of feature contours were computed for each instance. For computing the articulation rate as well as silent pause duration for each instance, the phoneme-based feature extraction system, introduced in (Herms, 2016), was applied. In order to consider phonemes of German language in the phoneme-based feature extractor, an acoustic model including 43 phonemes was trained on the basis of the *German open source corpus for distant speech recognition* (Radeck-Arneth et al., 2015). The resulting acoustic model consists of context-dependent triphone Hidden-Markov-Models with 32 Gaussians per state.

In order to remove the inter-speaker variability, Z-score normalization was applied within each speaker context. For this purpose, the features were adjusted with a mean of 0 and a standard deviation of 1 across instances for each subject separately.

3.2. Results

Figure 4 shows the means and 95% confidence intervals of prosodic and voice quality features under different cognitive load (CL) levels.

Results concerning the mean values of the articulation rate show a significant separation between low and medium ($p < .001$) as well as low and high cognitive load ($p < .001$). Between medium and high load, a difference cannot be observed for this parameter ($p > .05$); confidence intervals overlap. For silent pause duration, there is a statistically significant increase from low to high ($p < .001$) and medium to high cognitive load ($p < .001$). Nevertheless, an overall linear trend cannot be derived across cognitive load levels due to a slight drop from low to medium load.

Regarding the distribution of the intensity, no significant differences between groups can be observed (all $p > .05$).

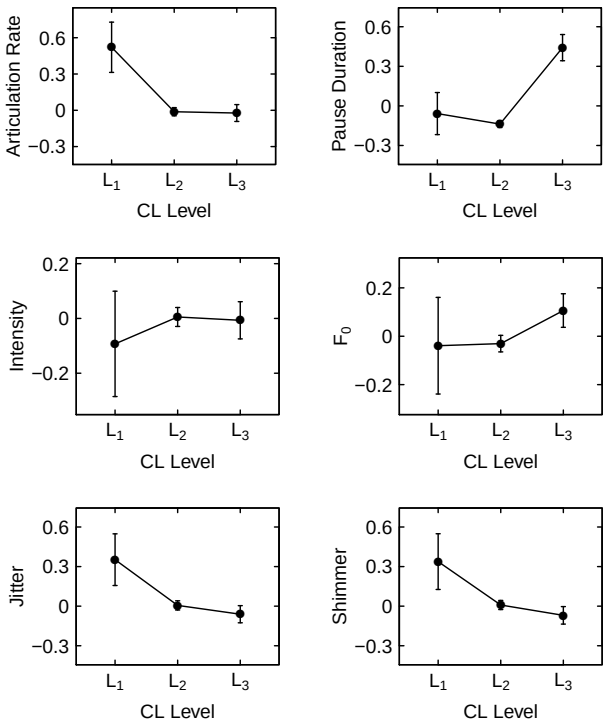


Figure 4: Means and 95% confidence intervals of speech parameters under different cognitive load (CL) levels.

From the visual impression in Figure 4, there are only marginal differences between the mean values of intensity as cognitive load increases and, moreover, confidence intervals overlap completely across cognitive load levels. The mean values of F_0 exhibit a statistically significant difference between medium and high cognitive load ($p < .001$). On the other hand, confidence intervals of F_0 overlap between low and medium as well as low and high cognitive load.

In case of the voice quality features jitter and shimmer, a monotonically decreasing trend can be observed for both parameters as the level of cognitive load increases. More precisely, the reduction of jitter and shimmer from low to medium cognitive load exhibit a statistically significant difference (both $p < .001$), whereas from medium to high cognitive load, a significant difference was obtained only for the parameter shimmer ($p < .001$). The results of both voice quality features indicate that speech includes less rough or hoarse characteristics as cognitive load increases.

4. Conclusion

We presented a new corpus named CoLoSS, which contains speech under cognitive load recorded in a learning task scenario. We used a dual-task approach to determine subjects' residual cognitive resources reflecting the degree of cognitive load. It comprises a visual-motor primary task that required subjects to learn abstract symbol combinations and an auditory-verbal secondary task to measure the load imposed by the primary task. This paper reports the methodology of collecting the speech recordings, constructing the corpus and gives a description of the data. Furthermore, effects of cognitive load on prosodic as well as voice quality features have been investigated in conjunction with the speech data of corpus.

For future work, we plan to conduct experiments aimed at the automatic speech-based recognition of cognitive load using both the numeric as well as the class labels of the corpus.

5. Acknowledgements

The presented research was conducted within the Research Training Group "CrossWorlds - Connecting virtual and real social worlds" (GRK 1780/1). The authors gratefully acknowledge funding by the German Research Foundation (DFG).

6. Bibliographical References

Baddeley, A. (1992). Working memory. *Science*, 255(5044):556.

Boersma, P. P. G. et al. (2002). Praat, a system for doing phonetics by computer. *Glot international*, 5.

Boril, H., Omid Sadjadi, S., Kleinschmidt, T., and Hansen, J. H. (2010). Analysis and detection of cognitive load and frustration in drivers' speech. *Proceedings of INTERSPEECH 2010*, pages 502–505.

Gorovoy, K., Tung, J., and Poupart, P. (2010). Automatic speech feature extraction for cognitive load classification. In *Conference of the Canadian Medical and Biological Engineering Society (CMBEC)*.

Herms, R. (2016). Prediction of deception and sincerity from speech using automatic phone recognition-based features. In *INTERSPEECH*, pages 2036–2040.

Hoffman, B. and Schraw, G. (2010). Conceptions of efficiency: Applications in learning and problem solving. *Educational Psychologist*, 45(1):1–14.

Holm, S. (1979). A simple sequentially rejective multiple test procedure.

Paas, F. G. and Van Merriënboer, J. J. (1994). Instructional control of cognitive load in the training of complex cognitive tasks. *Educational psychology review*, 6(4):351–371.

Paas, F., Tuovinen, J. E., Tabbers, H., and Van Gerven, P. W. (2003). Cognitive load measurement as a means to advance cognitive load theory. *Educational psychologist*, 38(1):63–71.

Plass, J. L., Moreno, R., and Brünken, R. (2010). *Cognitive load theory*. Cambridge University Press.

Radeck-Arneth, S., Milde, B., Lange, A., Gouvêa, E., Radomski, S., Mühlhäuser, M., and Biemann, C. (2015). Open source german distant speech recognition: Corpus and acoustic model. In *International Conference on Text, Speech, and Dialogue*, pages 480–488. Springer.

Schuller, B. W., Steidl, S., Batliner, A., Epps, J., Eyben, F., Ringeval, F., Marchi, E., and Zhang, Y. (2014). The interspeech 2014 computational paralinguistics challenge: cognitive & physical load. In *INTERSPEECH*, pages 427–431.

Sweller, J., Ayres, P., and Kalyuga, S. (2011). *Cognitive load theory*, volume 1. Springer.

Team, A. (2012). Audacity (version 2.0. 2). Retrieved from https://sourceforge.net/projects/ audacity/.

Wirzberger, M., Bijarsari, S. E., and Rey, G. D. (2017a). Embedded interruptions and task complexity influence schema-related cognitive load progression in an abstract learning task. *Acta Psychologica*, 179:30–41.

Wirzberger, M., Herms, R., Bijarsari, S. E., Rey, G. D., and Eibl, M. (2017b). Influences of cognitive load on learning performance, speech and physiological parameters in a dual-task setting. In *Abstracts of the 20th Conference of the European Society for Cognitive Psychology*, page 161.

Yap, T. F., Epps, J., Ambikairajah, E., and Choi, E. H. (2010). An investigation of formant frequencies for cognitive load classification. In *Eleventh Annual Conference of the International Speech Communication Association*.

Yap, T. F. (2012). *Speech production under cognitive load: Effects and classification*. Ph.D. thesis, University of New South Wales, Sydney, Australia.

Yin, B., Ruiz, N., Chen, F., and Khawaja, M. A. (2007). Automatic cognitive load detection from speech features. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*, pages 249–255. ACM.

VAST: A Corpus of Video Annotation for Speech Technologies

Jennifer Tracey, Stephanie Strassel
Linguistic Data Consortium, University of Pennsylvania
3600 Market Street, Suite 810, Philadelphia, PA. 19104 USA
{garjen, strassel}@ldc.upenn.edu

Abstract

The Video Annotation for Speech Technologies (VAST) corpus contains approximately 2900 hours of video data collected and labeled to support the development of speech technologies such as speech activity detection, language identification, speaker identification, and speech recognition. The bulk of the data comes from amateur video content harvested from the web. Collection was designed to ensure that the videos cover a diverse range of communication domains, data sources and video resolutions and to include three primary languages (English, Mandarin Chinese and Arabic) plus supplemental data in 7 additional languages/dialects to support language recognition research. Portions of the collected data were annotated for speech activity, speaker identity, speaker sex, language identification, diarization, and transcription. A description of the data collection and each of the annotation types is presented in this paper. The corpus represents a challenging data set for language technology development due to the informal nature of the majority of the data, as well as the variety of languages, noise conditions, topics, and speakers present in the collection.

Keywords: speech corpora, video corpora, multilingual resources

1. Introduction

The Video Annotation for Speech Technologies (VAST) corpus is the result of an effort to collect video content covering a diverse range of communication domains, data sources and video resolutions for use in training, development and testing of multiple speech technologies. The corpus comprises approximately 2900 hours of data, targeting three primary languages (English, Mandarin Chinese and Arabic) plus supplemental data in 7 additional languages/dialects to support language recognition research. Portions of the collected data were annotated for speech activity, speaker identity, speaker sex and language, diarization, and transcription. The collection of English, Mandarin, and Arabic data is referred to here as the "main corpus," and the data from supplemental languages as well as the annotations are referred to as the "sub-corpora."

2. Main corpus

Videos in the main corpus contain speech in English (including US, UK, and other varieties), Mandarin Chinese, and Arabic. Videos are assigned a primary language designation, but some videos may contain more than one language due to codeswitching in naturally occurring data. The majority of data in the VAST corpus consists of amateur videos harvested from the Internet. In addition, a small amount of audio from broadcast television news and/or informal talk shows is included in the main corpus. No annotations were performed on the broadcast data, and the discussion of the data in this paper therefore focuses primarily on the amateur video. Criteria for inclusion of videos in the main corpus are as follows:

- Videos must contain speech in one of the three primary languages
- Any variety/dialect of Arabic or English is acceptable for the main corpus, while Chinese videos must contain Mandarin
- Multi-party, informal speech is preferred over monologs, telephone-style dialogs or interviews
- There is no restriction on topic (variety of topics preferred)
- Speaker(s) are not required to appear on camera

Annotators hired and trained by LDC performed a "data scouting" task, in which they searched the web for appropriate content. During a given work session data scouts were instructed to search for videos appropriate for inclusion in the main corpus, or else they were instructed to do more focused searching for videos suitable for the SID or LID sub-corpora. Data scouting was conducted using a customized user interface developed for VAST, known as VScout. The VScout toolkit is a Firefox add-on consisting of an annotation form displayed on the left side of the browser window. Data scouts use the browser in the usual way to search, navigate video websites, and watch videos. When they find a suitable video they fill out the VScout webform, and the results are logged to a database. The data scouting process results in the following information about each video: page URL, number of speakers (1, 2, or 3+), sound conditions (background noise/speech, outdoors/indoors), speaker overlap, and language.

The average duration of amateur video clips is approximately three minutes, and Table 1 below provides a summary of some of the other features of the data. The features summarized are those that were noted during the data scouting process. Note that a video may contain both indoor and outdoor settings, so the percentages sum to greater than 100%. As can be seen in the summary, the majority of videos have some amount of background noise, and just over half have three or more speakers, making this a challenging dataset for annotation and transcription.

Feature	Percent of files
Indoor setting	59%
Outdoor setting	48%
Single speaker	25%
Two speakers	24%
Three or more speakers	51%
Background noise	67%

Table 1: Summary of Data Features

Language	Source Data	SAD	SID	LID	Diarization	Transcription
Arabic	818	197	0	91	0	91
English	768	187	99	32	43	32
Mandarin	720	280	0	20	0	29
Min Nan	11	0	0	11	0	0
Spanish	63	0	0	63	0	0
Portuguese	21	0	0	21	0	0
Russian	23	0	0	23	0	0
Polish	20	0	0	20	0	0
Total	**2444**	**664**	**99**	**281**	**43**	**152**

Table 2: Hours of amateur video in VAST corpus by subcorpus

3. Sub-corpora

Portions of the corpus were selected for inclusion in one or more sub-corpora: language ID (LID), speaker ID (SID), speech activity detection (SAD), diarization, and transcription. Except for the LID sub-corpus, all videos included in the sub-corpora were selected from the main corpus; the LID sub-corpus consists of some files selected from the main corpus as well as some additional files collected in languages not included in the main corpus. Videos that are extremely difficult for humans to annotate (due to noise conditions, number of overlapping speakers, etc.) were avoided when possible; however, limited availability of videos in some languages/dialects required inclusion of some of these more challenging videos in order to meet data volume targets. For each task, annotators were given guidelines and training, and were not permitted to perform the task in the production pipeline until task coordinators were satisfied with their level of competence. All tasks were performed by LDC annotators, with the exception of transcription, some of which was performed by LDC annotators and some of which was performed by external transcription vendors. Table 2 shows the data volume (hours of amateur video) in each subset of the corpus. As discussed in the sections that follow, some data appears in multiple sub-corpora.

3.1 SAD Sub-corpus

A portion of the data from the main corpus was annotated for speech activity detection (SAD), which included distinguishing speech from music, as well as labeling speech segments for language and speaker sex. Annotators review the output of an automatic SAD system or created segments from scratch (correction of automatic output proved less efficient than fully manual segmentation in many cases, so annotators were permitted to ignore automatic output). During this manual SAD correction and segmentation pass, annotators distinguished three categories:

- Non-speech. This category includes silence and non-vocal background noise. It may also include very short duration filled pauses or other speaker vocalizations that occur in isolation during a lengthy period of non-speech or music and were not detected by the automatic SAD pass.
- Speech. This category includes all speech including discernable background speech, backchannels, filled pauses, non-lexemes, laughter, coughing and all other speaker vocalizations.
- Music. This category includes vocal and non-vocal music: sung, instrumental, or rapped music, as well as rhythmic or chanted slogans. Music is sound which is intentionally produced to create melody (carrying a tune) and/or rhythm.

Non-speech was not explicitly segmented or labeled; any audio that is not contained within a segment is considered non-speech.

All speech segments are contained within a single "speech track"; during the SAD task annotators did not create separate tracks for each speaker's speech. Similarly, all music is contained within a single "music track". Speech segments and music segments may overlap and are maximally long and maximally inclusive. In other words, if the recording contains continuous speech for 15 seconds with no noticeable pauses, annotators created a segment on the speech track whose duration is 15 seconds – even if the number of speakers or the languages spoken changed during that segment.

While SAD annotation primarily relies on the audio recording, annotators were permitted to consult the video for help in disambiguating difficult cases. If the video and audio presented conflicting information, the annotator relied primarily on the audio. SAD annotation was performed on a total of 187 hours of English, 197 hours of Arabic, and 280 hours of Chinese.

3.1.1 Speaker Sex Labeling

Concurrent with the manual SAD correction and segmentation task, annotators also labeled each speech segment with respect to speaker sex. Segments were labeled as male (one or more speakers, all male), female (one or more speakers, all female), mixed (both male and female speakers), or unknown (speaker's sex cannot be determined).

3.1.2 Light LID Labeling

In addition to speaker sex, annotators also labeled each speech segment with respect to language. Labels are target (segment contains only target language), non-target (segment does not contain target language), mixed (segment contains target language plus one or more other languages), or unknown (language of the segment cannot be determined). For this task, "target language" is defined as the expected predominant language of the recording based on the source data auditing task.

3.2 Speaker Diarization Sub-corpus

To help address interest in speech processing in multi-speaker data, LDC performed speaker diarization on a subset of 43 hours from the English portion of the main corpus. This annotation results in distinct SAD segments for each individual speaker within this subset of data. Diarization was performed on speech segments only (i.e., voices of singers in music segments were not included), and the speaker sex and light LID labels from the SAD task were not applied to the diarized segments.

3.3 Transcription Sub-corpus

A portion of the main corpus was selected for transcription in each of the primary languages The Transcription sub-corpus includes 30 hours of English, 40 hours of Iraqi Arabic, 50 hours of Egyptian Arabic, and 29 hours of Mandarin Chinese, transcribed in a Quick Rich transcription style (Glenn et al., 2010; Bendahman et al., 2008). Selection of videos for transcription favored data with a high ratio of speech to non-speech. We also tried to maximize the number of distinct speakers in this sub-corpus. The English files for transcription were selected from the Speaker Diarization sub-corpus described above. For the Egyptian and Iraqi transcription efforts, guidelines for standardized spelling of the dialectal varieties were taken from previous transcription projects involving these dialects undertaken at LDC (Maamouri et al., 2004; Habash et al., 2012). The transcription was carried out in at least two passes, with the first pass focusing on correct verbatim transcription and speaker identification. The second pass focused especially on correcting use of transcription conventions for special categories like proper names, as well as adherence to the standardized orthography for the Egyptian and Iraqi Arabic transcription.

3.4 SID Sub-corpus

The SID sub-corpus contains multiple videos from each of 300 English speakers. For each speaker, 2-10 videos containing that speaker's voice were collected, with the aim of including a variety of interlocutors, speaking styles and acoustic/physical environments in the cluster for each speaker. The SID judgment was applied at the file level; that is, no segment-level annotation of speaker ID was applied. Each cluster of videos was audited to verify that the target speaker's voice occurs in each of the videos and that there is sufficient diversity of interlocutors and environments in the cluster. In the delivered corpus, videos in a cluster are marked "yes" or "no" for inclusion as the "core" cluster. Files included in the core are considered sufficiently diverse to form a valid cluster. Files marked "no" for inclusion in the core cluster are additional files containing the target speaker, but were found to be redundant with other videos in the cluster with respect to either interlocutors or environment. All clusters have a minimum of two videos in the "core". An effort was made to include 1-2 videos from each SID cluster in the Speaker Diarization corpus, but some clusters were collected too late in the project to be included in the Diarization annotation.

3.5 LID Sub-corpus

The LID sub-corpus contains approximately 11-23 hours of data for each of 15 languages (or language varieties), for a total of roughly 280 hours.

Languages were selected to match the language clusters used in the NIST LRE 2015 evaluation. In general, the VAST LID languages are a subset of the languages used in LRE 2015, with the exception of Gulf Arabic, which was added in VAST due to its substantial presence in the Arabic data. We included all languages from the LRE list where at least 10 hours of video were collected, plus Gulf

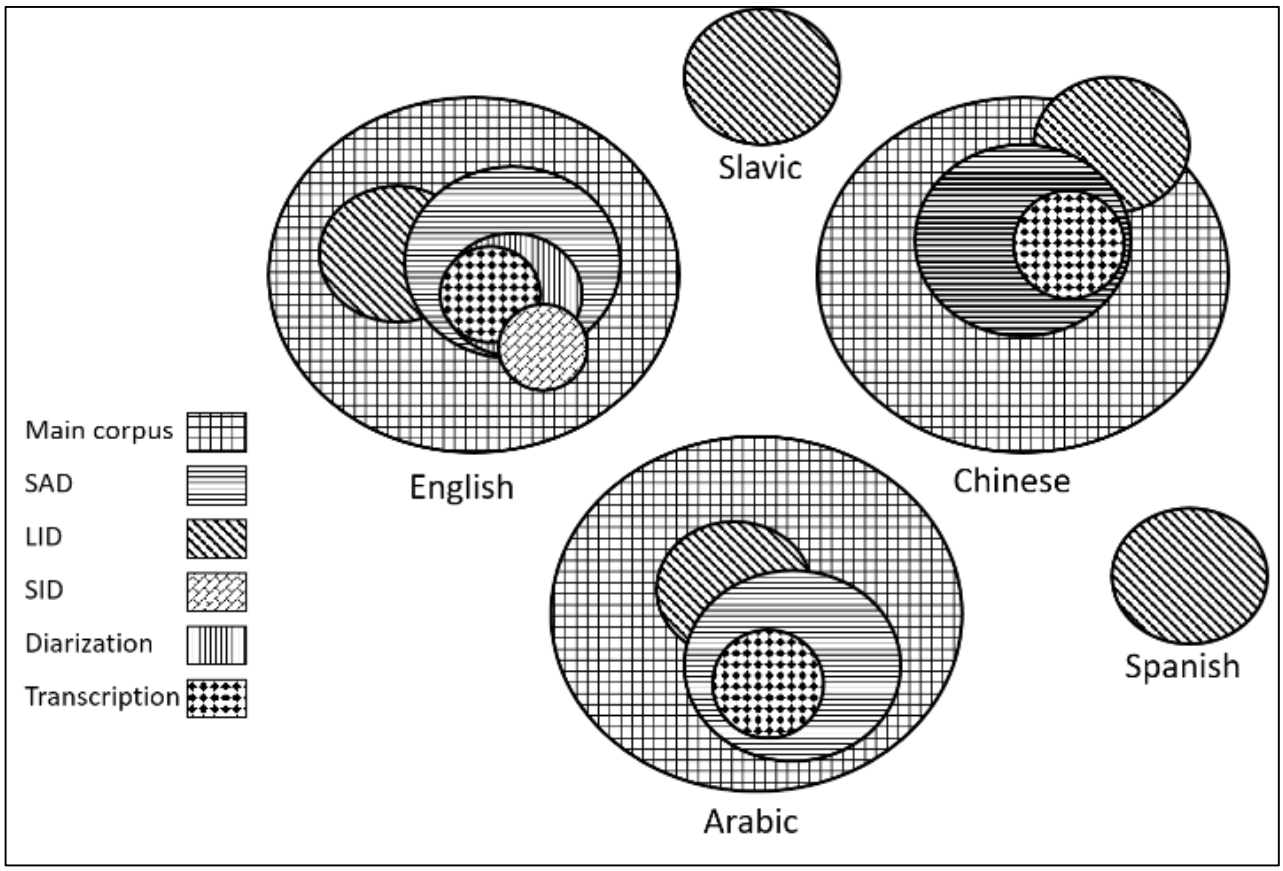


Figure 1: Relationship among VAST Sub-Corpora

Arabic. The final clusters for VAST are shown in Table 2. All included languages have at least 20 hours of data except for Maghrebi and Gulf Arabic (15 hours each), Min Nan (11 hours), and British English (12 hours). Videos designated as part of the LID corpus contain some speech in the target variety, preferably more than 50% of the speech in the video; however, some videos included may have lower proportions of target language variety speech.

Arabic Cluster	Slavic Cluster	Chinese Cluster
Egyptian Arabic Iraqi Arabic Levantine Arabic Maghrebi Arabic Gulf Arabic	Polish Russian	Mandarin Min Nan
English Cluster	**Spanish Cluster**	
British English Gen. American English	Caribbean Spanish European Spanish Latin American Spanish Brazilian Portuguese	

Table 3: VAST LID Language Clusters

3.6 Relationship among the Sub-corpora

SAD, Diarization, Transcription, and Speaker ID sub-corpora are all subsets of the main corpus, where Diarization and Speaker ID come from the English portion of the main corpus only, and SAD and transcription files are taken from all three primary languages in the main corpus. The LID sub-corpus contains some files from the main corpus (all varieties in the Arabic and English clusters, and Mandarin in the Chinese cluster), along with some files that are not part of the main corpus (Min Nan from the Chinese cluster, as well as the Slavic and Spanish clusters). With a very few exceptions, all Diarization and Transcription files are a subset of the SAD sub-corpus; English Transcription files are a subset of the files in the Diarization sub-corpus. In addition, there is partial overlap between the SID sub-corpus and the Diarization and English Transcription sub-corpora, as well as between the English, Chinese, and Arabic clusters of the LID corpus and the SAD and Transcription Corpora. Figure 1 illustrates the interactions between the various sub-corpora.

4. Availability of VAST Data and Future Work

To date, portions of the VAST data have been used in the NIST 2017 (Pilot) Speech Analytic Technologies Evaluation (NIST 2017a) and in the 2017 NIST Language Recognition Evaluation (NIST 2017b). Additional annotation of the VAST corpus is current in progress, including additional SAD annotation on data that may be used in future test sets, as well as some annotation of video features for a subset of the data. While some portions of the VAST corpus are being withheld for use in future Open SAT, SRE, and LRE evaluations, specific sub-corpora will appear in LDC's public catalog starting in 2018. The first releases will include approximately 2000 files per language (Arabic, Chinese, and English) with SAD annotation, as well as the 29 hours of Chinese transcription data.

5. 6. Bibliographical References

Bendahman, C., Glenn, M., Mostefa, D., Paulsson, N., Strassel, S. (2008). Quick Rich Transcriptions of Arabic Broadcast News Speech Data. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), pages 3605–3608, Marrakech, Morocco, May 28-30. European Language Resource Association (ELRA).

Glenn, M., Strassel, S., Lee, H., Maeda, K., Zakhary, R., Li, X. (2010). Transcription Methods for Consistency, Volume and Efficiency. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), pages 2915–2920, Valletta, Malta, May 17-23. Workshop on Language Resources and Human Language Technologies for Semitic Languages. European Language Resource Association (ELRA).

Habash, N., Diab, M., Rambow, O. Conventional Orthography for Dialectal Arabic. In Nicoletta Calzolari (Conference Chair), et al., editors, Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'12), pages 711–718, Istanbul, Turkey, May 21-27. European Language Resource Association (ELRA).

Maamouri, M., Buckwalter, T., Cieri, C. Dialectal Arabic Telephone Speech Corpus: Principles, Tool Design, and Transcription Conventions. In Proceedings of NEMLAR International Conference on Arabic Language Resources and Tools, Cairo, Egypt, September 22-23.

NIST. (2017a). 2017 (Pilot) Speech Analytic Technologies Evaluation. https://www.nist.gov/itl/iad/mig/nist-2017-pilot-speech-analytic-technologies-evaluation. Accessed February, 2018.

NIST. (2017b). 2017 NIST Language Recognition Evaluation. https://lre.nist.gov/. Accessed February, 2018.

Edit me: A Corpus and a Framework for Understanding Natural Language Image Editing

Ramesh Manuvinakurike*[2], Jacqueline Brixey*[2], Trung Bui[1], Walter Chang[1], Doo Soon Kim[1], Ron Artstein[2], Kallirroi Georgila[2]

[1]Adobe Research, San Jose, USA

[2]USC Institute for Creative Technologies, Los Angeles, USA

{manuvina,brixey}@usc.edu, {bui,wachang,dkim}@adobe.com, {artstein,kgeorgila}@ict.usc.edu

Abstract

This paper introduces the task of interacting with an image editing program through natural language. We present a corpus of image edit requests which were elicited for real world images, and an annotation framework for understanding such natural language instructions and mapping them to actionable computer commands. Finally, we evaluate crowd-sourced annotation as a means of efficiently creating a sizable corpus at a reasonable cost.

Keywords: dialogue, image editing, vision and language

1. Introduction

Photo editing is as old as photography itself. Over the years, darkroom techniques with film and light have given way to digital processing, and software suites such as Adobe Photoshop have made image editing accessible to millions of professionals and hobbyists. But even with the best tools, photo editing requires substantial knowledge, and novices often need to enlist the help of experts. Web sites such as https://www.reddit.com/r/PhotoshopRequest/ and http://zhopped.com/ contain thousands of image edit requests like the following:

- There is a spot on my wedding dress. Can someone please remove it. Please!
- Can you please fix the glare on my dog's eyes? I lost him today and he means the world to me.
- Can you please remove the people in the background? This is the only surviving photo of my mom and I would like to preserve it.
- I love this photo from our trip to Rome. Can someone please remove my ex from this photo? I am the one on the right.

As the examples above show, a natural way for novices to express their editing needs is through ordinary human language. At Adobe Research we aim to develop a software tool that will interpret such natural language image edit requests, and carry them out to the user's satisfaction.

This work presents a step in the direction of understanding human image edit requests: a corpus of such requests, and an annotation scheme for mapping these requests into actionable computer commands. A corpus could be compiled from naturally occurring examples such as the ones cited above, but this would raise concerns about privacy and ownership of the photographs, and the images themselves are unprocessed. Instead, we use a set of publicly available, richly annotated images called the Visual Genome corpus (Krishna et al., 2017). We elicited image edit requests that would pertain to these images, devised a scheme for mapping these requests into actionable commands, annotated requests according to the scheme, and evaluated the reliability of these annotations.

The contributions of this work are threefold. First, a dataset of natural language image edit requests for images with detailed image captions. These captions are particularly useful for the task of language understanding, as many of the requests make reference to objects in the images. Second, a framework for understanding these natural language instructions and mapping them to actionable computer commands. Finally, we provide a crowd-sourcing methodology to offload complex annotation between expert users and novice users and evaluate them. This is particularly useful for creating a sizable corpus.

2. Related Work

Recently, there has been a lot of work on applications that combine vision and language, e.g., understanding and generating image descriptions (Kulkarni et al., 2013), identifying visual reference in the presence of distractors (Paetzel et al., 2015; de Vries et al., 2016), visual question answering (Antol et al., 2015), visual storytelling (Huang et al., 2016), generating questions about an image (Mostafazadeh et al., 2016), and question-answer interactions grounded on information shown in an image (Mostafazadeh et al., 2017). Current image and language corpora typically consist of digital photographs paired with crowd-sourced captions (Lin et al., 2014; Krishna et al., 2017), or in some cases with questions related to those images (Mostafazadeh et al., 2016).

Much of the work above is relevant to the problem at hand. For example, understanding image descriptions is crucial for interpreting the requests quoted above, as all of them contain image descriptions (*my wedding dress; my dog's eyes; the people in the background; my ex*). However, to our knowledge, no work has yet attempted to tackle the specific task of automated image *editing* through natural language. Nor are we aware of any work that even tries to understand what users want to change when editing photos, and how

* Primary authors; work done while at Adobe Research

users talk about making those changes. This is the focus of the present work.

3. Data Collection

Edit requests were collected through crowd-sourcing using Amazon Mechanical Turk (https://mturk.com). We selected a small subset of images from the Visual Genome corpus (Krishna et al., 2017), a richly annotated set of about 108k images sampled from the MS COCO data-set (Lin et al., 2014). To provide enough visual detail for eliciting nuanced edit requests, we only used images with high resolution (1000×700 pixels and above). To elicit language that is similar to naturally occurring requests, we analyzed 200 posts from Reddit and Zhopped, and found that the images in those posts generally fell into eight high-level categories: animals, city scenes, food, nature/landscapes, indoor scenes, people, sports, and vehicles. We then chose images for elicitation that fit these categories. A total of 334 images were used for elicitation.

Each image was given to 5 crowd-source workers (called turkers), and each turker was asked to provide at least 5 unique edits that they would want to see in the image, phrased as requests in natural English (workers were free to provide more edits; Figure 1). The requests entered by the turkers were manually reviewed by the first two authors for quality and variation; incoherent submissions and unrelated requests were excluded. After filtering, we were left with 9101 edit requests with a total of 44727 word tokens (4628 unique word types). An example image with a few annotated requests is shown in Figure 2.

4. The Language of Image Edits

Review of the elicited requests provides insights into how users want to edit images.

Vocabulary. The elicited requests exhibit wide variation in vocabulary, with turkers using different terms to express essentially the same needs. For example, the requests *Make the colors pop*, *Bring out the colors*, and *Change the saturation* all express a desire for more vivid colors. Similarly, the desire for altering the dimensions of an image was expressed using the terms *crop*, *cut out*, and *delete*.

Ambiguity. Some terms are ambiguous between technical and general uses: the word *focus* may appear in a specific optical sense, but in the request *Make the bird the focus of the picture* it is probably used in the sense of general prominence. Technical terminology is also ambiguous, for instance the term *zoom*. A camera's zoom changes the focal length of the lens, and thus the angle of view and picture frame; whereas the zoom feature in a graphical user interface typically changes the view of a document. Thus, the interpretation of a request like *zoom in on the man* depends on context: as a standalone request, it probably represents the intention of changing the picture frame; but in interactive dialogue or as part of a multi-instruction sequence, it could indicate a request to change the view of the image in the editing interface.

Structure. Many of the edit requests contain a verb in the imperative, often at the beginning, such as in *Make the picture brighter*; sometimes the verb appears in a conjunct

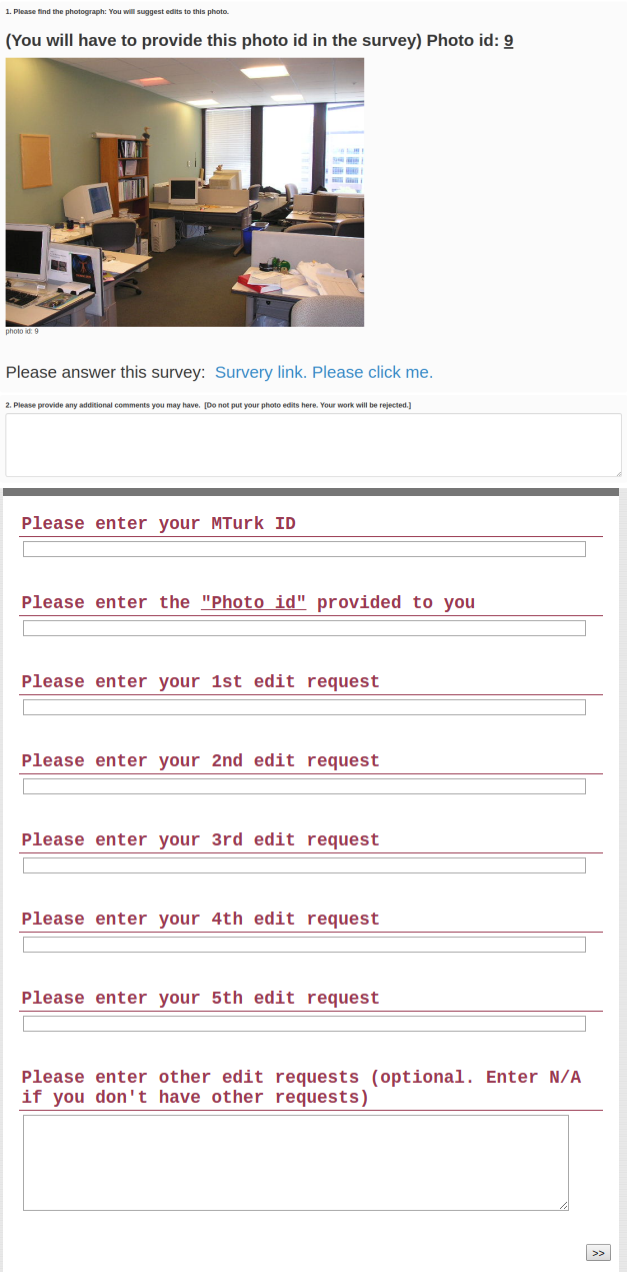


Figure 1: The interface shown to the turkers with the image for which they provide the editing commands. The turkers need to provide at least 5 unique edits.

clause, such as *The tree is distracting so remove it*. In other instances, however, the request takes the form of a comment, with the desire remaining implicit. For example, *The image is very blurry* suggests a goal of making the image less blurry, but does not directly state this intention nor how to achieve the goal.

Domain knowledge. Some variation in the language of edit requests can be attributed to the turkers' expertise. Novice turkers often use broad and high-level language, while those familiar with image editing may provide very specific instructions tailored for an editing tool. For example, a desire for better color balancing was expressed by a novice turker as *I would like to see more character and*

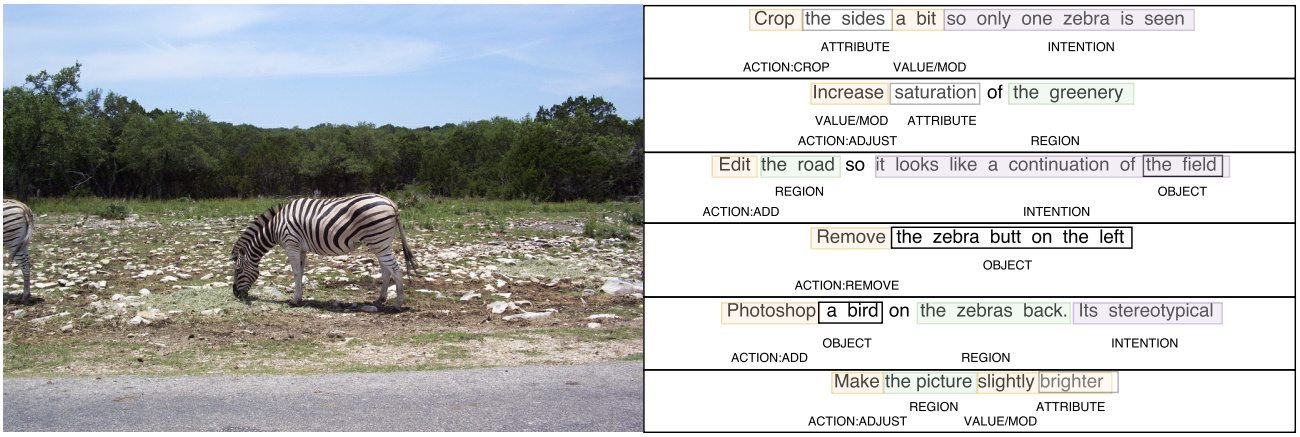


Figure 2: Example image from the corpus, with a few Image Edit Requests and their annotations.

color to the cobblestone sidewalk. It is lovely. An expert turker expressed a similar desire as *Adjust the brightness on the white tool to avoid making it look plain white.*

5. Annotation Framework

The ultimate goal of this research is to develop a tool that can carry out image edit instructions given in natural language; that is, a tool that can interpret these instructions in terms of editing functionality available in software like Photoshop, in a manner similar to how human expert photo editors interpret such requests today. To support this goal, we have devised an annotation framework that is an intermediary form between natural language and Photoshop commands. The actions in the intermediary form are fairly close to what is available in Photoshop, so carrying out these instructions is primarily a matter of interpreting the various properties and attributes – for example, which region of the photo is referred to by *my wedding dress* or *my dog's eyes*. This is by no means an easy task, but not one that concerns us here; we focus on identifying the actions, properties and attributes in the natural language requests.

The annotation framework is structured in three levels. In the first, an utterance is determined to be either an Image Edit Request (IER) or a comment. IERs receive two further levels of annotation, namely actions and entities: each IER is composed of one action and zero or more entities (an utterance which expresses multiple actions is segmented into separate IER). The terminology for the entities is borrowed from the work by Williams et al. (2015) which uses the entities to indicate the higher level intents. Comments are utterances that do not have an action.

To clarify the distinction between IERs and comments, we define an IER as an *actionable* item that can be interpreted up to some degree of certainty, albeit incompletely. A comment is an utterance that pertains to the image and may well include a request, but does not contain an action that could be completed by an image editing program given a particular image. For example, the utterance *This photo should have been taken with a Nikon camera* would be a comment as it is impossible to fulfill this request in an image editing program. Comments do not have any additional annotations.

Adjust (44.89%) Increase saturation a bit on the elephants.
Delete (13.70%) Remove the jacket hanging from the girl's side.
Crop (6.89%) Crop the photo to eliminate the space to the left and right of the elephants.
Add (6.85%) Insert a ball hitting the tennis racket.
Replace (2.47%) Please change the pamphlet she is holding into a dictionary.
Apply (1.44%) Add a Gaussian blur to the background.
Zoom (0.87%) Zoom in on the man.
Rotate (0.71%) The photo looks tilted. Rotate it clockwise so the lines are straight.
Transform (0.62%) Flip the photo horizontally.
Move (0.60%) Move the white framed picture to the blue wall.
Clone (0.33%) Use a cloning tool to blend grass to cover any patches of dirt on the ground.
Select (0.19%) Select the white dog.
Swap (0.14%) Please perform a face swap using the man in the yellow shirt and the man in the blue/black polo.
Undo (0.02%) If possible uncrop photo to allow more space to frame, rather than cut off the bike.
Merge (0.02%) Blend the grey smudges so they are the same color as the rest of the dirt.
Redo (0.01%) Redo all white traffic lines in street.
Other (0.01%) Resize photo to show large elephant and trainer.
Scroll (0.00%) No example in the corpus.

Figure 3: Action types, with frequency and examples.

IERs are annotated with at most one action, and its related entities, if any. An utterance that expresses multiple actions, such as, *Crop the left side of the photo and increase the saturation*, is marked as two IERs to accommodate the two separate actions. The action is usually an action verb which either explicitly or implicitly provides a mapping of a word or a phrase to a vocabulary that must be interpretable by most of the popular image editing programs. The frame-

Attribute Properties of the image to adjust, such as saturation, hue, or brightness.

Object An item to be inserted or deleted.

Region Location within the image where an action is being applied, such as top, entire image, or on a requested subject already in the photo.

Modifier/value Degree or direction of the change such as increase/decrease, modifiers of degree (examples: *a little*, *a lot*, *all*), directions (*to the left*), or numerical values (*25%*).

Intention User's reason or end goal for the change.

Figure 4: Types and descriptions of entities.

work supports 18 possible actions (Figure 3). The most common action represented in the data set was *adjust*, for such utterances as: *Make the image brighter*, *Increase the saturation*, and *Decrease the shadows*. Some actions are extremely rare in our corpus: this is because the framework was designed to also allow for interactive dialogue with an image editor. The framework therefore contains actions like *undo*, *redo*, *select*, *merge*, and *scroll*, which rarely come up in one-shot IERs of the type elicited here (such actions do show up when expert turkers give a complex, multi-stage request, for example: *Free select the sky, following building edge and around halo of the sun then increase contrast to reduce glare*).

The action provides first level of understanding of an IER. However, it is not sufficient to have the action alone if the user has provided additional details in an utterance. Actions support a list of five entities that complete the interpretation of an IER (Figure 4). Entities mark information about how the action is applied as an edit, such as detailing where a crop should occur or by how much the saturation level should be increased. Our framework supports the flexibility of an utterance having zero entities as well as an IER with multiple entities of the same type.

The various entity types are given in Figure 4. ATTRIBUTE holds information about what property of the image to adjust, and MODIFIER/VALUE provides information about the degree or direction of the change. For example, the IER *Increase the saturation* is annotated with the Action *adjust*, Attribute *saturation*, and Modifier *up*. We make a distinction between OBJECT, which is inserted or deleted, and REGION, which is the area where the action is to be applied. For example, in the IER *Add a dog*, the word *dog* is labeled as an Object as it is an entity to insert, but in *Brighten up the wave*, the word *wave* is regarded as a Region rather than an Object, as the person is interested in adjusting its brightness (and thus we have Action *adjust*, Attribute *brightness*, Region *wave*, and Modifier *up*). Finally, we included the entity INTENTION as users often provide information about their objective for performing the change. These intentions are by themselves not actionable but provide additional information: for example, the utterance *Paint the rocks unnatural but interesting colors like purple, green, yellow, and red to make the effect surreal* expresses a user's desire to make a *surreal* looking image, and is therefore annotated with Intention *to make the effect surreal*. A unique feature of this framework is that the same word can have multiple labels or one can be a sub-set of another. In the example *Increase the saturation*, the word *increase* is labeled both as an *adjust* action as well as a Modifier entity.

We thus propose this intermediary language scheme as a means to address the variability in vocabulary, structure, and ambiguity in IERs. To our knowledge, no other such published annotation scheme exists, and no one-to-one mapping of edit requests to executable actions in an image editing program permits for the described flexibility and range in natural language image edit utterances.

Feature	Krippendorff's alpha		
IER vs. comment	0.28	0.53	0.35
Action type	0.74	0.62	0.59
Attribute	0.47	0.41	0.38
Object	0.51	0.27	0.47
Region	0.55	0.35	0.43
Modifier/value	0.31	0.04	0.07
Intention	0.51	0.67	0.52

Table 1: Inter-rater reliability for 3 groups of annotators.

6. Analysis

To validate the annotation scheme we conducted an inter-rater reliability study on a sample of 600 utterances. Nine annotators received training, feedback on a set of 25 utterances, and support during the annotation process. The annotators were divided into groups of three, and each group annotated a different set of 200 utterances. Reliability was measured separately on actions and entities using Krippendorff's alpha (Table 1). For action type we used the nominal distance metric; IER versus comment was treated as a binary feature, and so were the five entity types, marking either presence or absence of that entity in a particular utterance. The highest agreement is reached on the action types; agreement on entities is somewhat lower but still well above chance. For some entities (value in particular) agreement borders on chance level, suggesting that annotation of entities in general and value in particular need to be better defined.

Crop and *add* actions were the most agreed upon actions between annotators. Requests to alter features of people in a photo presented the majority of discrepancies. For example, the utterance *You could have everyone smiling* was annotated as *add*, *replace*, and as a comment. Utterances with the phrases *clean up* and *edit* also presented differing annotations of *adjust*, *delete*, and *other*. Finally, utterances without an imperative verb were frequently annotated differently. The utterance, *The photo is too bright* was interpreted by annotators as a comment or as an IER with an *adjust* action. Future versions of the annotation manual will attempt to clarify these issues in order to ensure more consistent annotation.

For annotating at scale we used crowd-sourcing: 6000 elicited utterances were annotated, with only one turker annotating a given utterance. Only actions were marked by this group. Workers received an instructional video which was mandatory to watch before annotation. To determine

Annotators	Krippendorff's alpha		
3 trained	0.74	0.62	0.59
3 trained + 1 crowd	0.47	0.25	0.40

Table 2: Inter-rater reliability on identifying action type, for three groups of trained annotators and the same groups with crowd annotators.

the reliability of annotations completed by turkers, reliability between turkers and trained annotators was calculated on the 600 utterances completed by the trained annotators (Table 2). Reliability between turkers and trained annotators was much lower than between trained annotators, suggesting substantial differences between the groups; these may be due to training (turkers frequently selected the action *other*), or to the population from which the annotators were drawn.

7. Conclusion

This paper introduced a data-set for the domain of image editing using natural language. This is a currently unexplored task that combines language and vision. Our corpus comprises more than 9000 IERs collected via crowdsourcing. We built an annotation scheme for understanding such natural language image editing instructions and mapping them to actionable computer commands. Finally, we evaluated crowd-sourced annotation as a means of efficiently creating a sizable corpus at a reasonable cost. In future work, the corpus will be used for learning models that can automatically detect actions and entities, as well as the sequences of these components in an IER.

8. Acknowledgments

We would like to thank the many Mturkers who contributed image edits and annotations for this work. We would also like to thank Robert Dates, Takeshi Onishi, Arman Cohan, and Quan Tran for their annotations. The first author was also supported by a generous gift of Adobe Systems Incorporated to USC/ICT. The second author thanks the National GEM Consortium for fellowship funding and internship placement at Adobe. The last two authors were supported by the U.S. Army; statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

9. Bibliographical References

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. (2015). VQA: Visual Question Answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, Santiago, Chile.

de Vries, H., Strub, F., Chandar, S., Pietquin, O., Larochelle, H., and Courville, A. (2016). GuessWhat?! visual object discovery through multi-modal dialogue. *arXiv preprint arXiv:1611.08481*.

Huang, T.-H. K., Ferraro, F., Mostafazadeh, N., Misra, I., Agrawal, A., Devlin, J., Girshick, R., He, X., Kohli, P., Batra, D., Zitnick, C. L., Parikh, D., Vanderwende, L., Galley, M., and Mitchell, M. (2016). Visual storytelling. In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT)*, pages 1233–1239, San Diego, California, USA.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M., and Fei-Fei, L. (2017). Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., and Berg, T. L. (2013). Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2891–2903.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755, Zurich, Switzerland.

Mostafazadeh, N., Misra, I., Devlin, J., Mitchell, M., He, X., and Vanderwende, L. (2016). Generating natural questions about an image. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1802–1813, Berlin, Germany.

Mostafazadeh, N., Brockett, C., Dolan, B., Galley, M., Gao, J., Spithourakis, G., and Vanderwende, L. (2017). Image-grounded conversations: Multimodal context for natural question and response generation. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, Taipei, Taiwan.

Paetzel, M., Manuvinakurike, R., and DeVault, D. (2015). "So, which one is it?" The effect of alternative incremental architectures in a high-performance game-playing agent. In *Proceedings of the Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 77–86, Prague, Czech Republic.

Williams, J. D., Kamal, E., Ashour, M., Amr, H., Miller, J., and Zweig, G. (2015). Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS). In *Proceedings of the Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 159–161, Prague, Czech Republic.

Enriching a Lexicon of Discourse Connectives with Corpus-based Data

Anna Feltracco[1,2,3], Elisabetta Jezek[2], Bernardo Magnini[1]
[1]Fondazione Bruno Kessler, Via Sommarive 18, 38100 Trento, Italy
[2]University of Pavia, Strada Nuova 65, 27100 Pavia, Italy
[3]University of Bergamo, Via Salvecchio, 19, 24129 Bergamo, Italy
feltracco@fbk.eu, jezek@unipv.it, magnini@fbk.eu

Abstract

We present the results of the effort of enriching the pre-existing resource LICO, a Lexicon of Italian COnnectives retrieved from lexicographic sources (Feltracco et al., 2016), with real corpus data for connectives marking contrast relations in text. The motivation beyond our effort is that connectives can only be interpreted when they appear in context, that is, in a relation between the two fragments of text that constitute the two arguments of the relation. In this perspective, adding corpus examples annotated with connectives and arguments for the relation allows us to both extend the resource and validate the lexicon. In order to retrieve good corpus examples, we take advantage of the existing Contrast-Ita Bank (Feltracco et al., 2017), a corpus of news annotated with explicit and implicit discourse contrast relations for Italian according to the annotation scheme proposed in the Penn Discourse Tree Bank (PDTB) guidelines (Prasad et al., 2007). We also use an extended -non contrast annotated- version of the same corpus and documents from Wikipedia. The resulting resource represents a valuable tool for both linguistic analyses of discourse relations and the training of a classifier for NLP applications.

Keywords: discourse connectives, contrast relation, corpus examples

1. Introduction

Discourse relations and the linguistic elements marking them in text, commonly referred to as *discourse connectives*, have recently been at the core of several annotation efforts for multiple languages (including English, German, French, Italian, Portuguese, see Stede and Umbach (1998), Roze et al. (2012) among others). In this paper, we present the results of the effort of enriching the pre-existing resource LICO, a lexicon of Italian connectives retrieved from lexicographic sources (Feltracco et al., 2016), with real corpus data, thus allowing us to both extend the resource and validate the lexicon. Our goal in this contribution is limited to the class of connectives marking contrast, and the additional relations such connectives might convey, some of them being polysemous. The motivation beyond our effort is that connectives can only be interpreted and disambiguated when they appear in context, that is, in a relation between the two fragments of text that constitute the two arguments of the relation. In order to retrieve good examples, we take advantage of the existing Contrast-Ita Bank (Feltracco et al., 2017), a corpus of news annotated with explicit and implicit discourse contrast relations for Italian according to the annotation scheme proposed in the Penn Discourse Tree Bank (PDTB) guidelines (Prasad et al., 2007). Contrast-Ita Bank contains corpus annotations for 19 discourse connectives of contrasts, 14 of them are included in LICO; these provide the starting point for our work. We pick additional examples from the larger news corpus from which Contrast-Ita Bank is derived. The resulting resource represents a valuable tool for both linguistic analyses and the training of a classifier for NLP applications. The paper is structured as follows. Section 2 reports the definition of discourse connective we assume in our work. Section 3 introduces LICO and related lexica for other languages, while Section 4 reports the methodology and Section 5 presents the results of the effort of enriching the resource. The paper ends with concluding observations and hints for further work.

2. Discourse connectives

We define discourse connectives as lexical markers that are used to express relations between parts of the discourse. This definition is inspired by Ferrari (Ferrari and Zampese, 2000; Ferrari, 2010): she defines a *connective* as "each of the invariable forms [...], which introduce relations that structure "logically" the meanings of the sentence and of the text" [1].

Ferrari clarifies that relations marked by connectives hold between events or assertions, and includes as arguments for the relation also nominalisations (e.g. "after the pressing invitation ..."), i.e. cases that contain an event introduced through a nominal expression. On the other hand, she excludes those grammatical elements that introduce relative clauses or pronouns (as *who* in "I don't know who you are.") to be connectives. This is in line with the definition provided for the *arguments* of a *connective* in the Penn Discourse Tree Bank (PDTB) 2.0 project, for which connectives relate two events, states, and propositions, that can be realized mostly as clauses, nominalisations, and anaphoric expressions (Prasad et al., 2007). From this group are excluded general *cue phrases* or *discourse markers*, words/phrases that do not have the function of connectives but are used for instance to change the topic in a discourse or to initialize it, such as "but" in "But, what are you doing?". According to Ferrari (2010), connectives belong to different syntactic classes, the same defined in the PDTB schema: i) subordinating conjunctions or subordinating expressions; ii) coordinating conjunctions or coordinating ex-

[1]Original text: "Il termine *connettivo* indica in linguistica ciascuna delle forme invariabili [...], che indicano relazioni che strutturano 'logicamente' i significati della frase e del testo"(Ferrari, 2010).

pressions; iii) adverbs or adverbial expressions; iv) prepositions or prepositional expressions.
In line with this definition, Stede (2012) distinguishes connectives as never inflected, closed-class lexical items, which belong to the above mentioned syntactic categories. He also specifies that these lexical elements can only be interpreted successfully when they appear in a relation between two discourse segments.
Ferrari (2010) also proposes a non hierarchical classification for connectives depending on the "type of logical relation they convey", e.g. *temporal* and *causal*. The PDTB 3.0 project (Webber et al., 2016) proposes a hierarchical classification composed by three levels (Table 1).

I level	II level	III level
CLASS	TYPES	SUBTYPES
TEMPORAL	Synchronous	–
	Asynchronous	Precedence
		Succession
CONTINGENCY	Cause	Reason
		Result
	Condition	Arg1-as-cond
		Arg2-as-cond
	Negative Condition	Arg1-as-negcond
		Arg2-as-negcond
	Purpose	Arg1-as-goal
		Arg2-as-goal
COMPARISON	Contrast	–
	Similarity	–
	Concession	Arg1-as-denier
		Arg2-as-denier
EXPANSION	Conjunction	–
	Disjunction	–
	Equivalence	–
	Instanciation	–
	Level-of-detail	Arg1-as-detail
		Arg2-as-detail
	Substitution	Arg1-as-subst
		Arg2-as-subst
	Exception	Arg1-as-except
		Arg2-as-except
	Manner	Arg1-as-manner
		Arg2-as-manner

Table 1: The PDTB 3.0 hierarchy of relations (Webber et al., 2016).

In the first level of the hierarchy, the *class level*, sense tags are grouped in four major classes (first column of Table 1). The second level of the hierarchy (second column of Table 1) specifies further the semantics of the class level: the *type level*. For example, the tag TEMPORAL.Synchronous indicates the type *Synchronous* of the class TEMPORAL and is used for connectives that indicate that the arguments of the relation are simultaneous (e.g. "When she arrived, he was leaving"); differently, the TEMPORAL.*Asynchronous* tag is used when the connective indicates a before-after relation (e.g. "She arrived before he left"). The third level, the *subtype level* (third column of Table 1), reflects the *direction of the relations*. For example, the type CONTINGENCY.*Cause* represents an asymmetric relation between two arguments: being one the cause, and the other the result. The subtype CONTINGENCY.*Cause*.Reason is used if the argument introduced by the connective (Arg2) is the reason for the situation in the other argument (Arg1) (e.g. "I stayed at home because it was raining"), while CONTINGENCY.*Cause*.Result is used if it represents the result/effect (e.g. "It was raining, therefore I stayed at home"). Notice that not every *type* has a further *subtype*: for example, the arguments involved in a temporal relation of type *Synchronous* do not play different roles and no subtype has been proposed.

3. LICO: Lexicon for Italian Connectives

According to our knowledge, LICO (Feltracco et al., 2016), Lexicon for Italian COnnectives, is the highest coverage resource of discourse connectives available for Italian.

Connectives in LICO. In LICO connectives are listed together with orthographic, syntactic, semantic information and also possible alignments with lexica of connectives in other languages. LICO is organized in 173 entries, each one corresponding to a connective and its orthographic or lexical variants. In fact, the invariability criterion proposed by Ferrari (2010) which does not include variable forms (i.e. those forms which are subject to morphological modifications) is partially dropped in LICO. Specifically, the resource does not include forms which exhibit morphological inflection or conjugation, but includes connectives which show a certain degree of lexical variability, that is, multiword expressions which are not totally rigid from a lexical point of view (e.g. *ad esempio/per esempio* 'for example' are both registered in the resource, as two variants of unique entry).
Connectives in LICO are retrieved from three sources: i) the list of connectives mentioned by Ferrari for the entry *connettivi*[2], ii) the list of connectives tagged as *congiunzione testuale* in Sabatini Coletti 2006 (Sabatini and Coletti, 2007), except for the ones of literary use, and iii) the list of the equivalent Italian terms of the German connectives in the DimLex resource (Stede, 2002) (see Related Lexica).

LICO Structure. For each entry LICO specifies:

- possible lexical variants (e.g *dopo di ché* and *dopo di ciò*) and orthographic variants (e.g. *ciò nonostante* and *ciononostante*);
- whether the connective (or its variants) is composed by a single token or by more than one token;
- whether the connective is composed by correlating parts (e.g. *da una parte [..] dall'altra*) or not (e.g. *ciononostante*);
- the syntactic category: adverbs, prepositions, subordinating or coordinating conjunctions;
- the semantic relation(s) which the connective indicates, according to the PDTB 3.0 schema of relations (Webber et al., 2016);

[2]http://www.treccani.it/enciclopedia/connettivi_(Enciclopedia-dell'Italiano)/

- possible alignments with lexica of connectives in German;
- examples of usage of the connective for each semantic relations it indicates.

The examples in the first version of the resource are translation of the German examples already present in the DimLex resource (Scheffler and Stede, 2016; Stede, 2002; Stede and Umbach, 1998). In adopting a corpus-based approach we aim at enriching LICO with data-driven examples and validating the information in the resource.

Related lexica. LICO has been inspired by the DimLex project for German (Scheffler and Stede, 2016; Stede, 2002; Stede and Umbach, 1998)[3], an XML-encoded resource that provides information on orthographic variants, syntactic category, semantic relations in terms of PDTB3.0 (Webber et al., 2016) sense tags, and usage examples for 274 connectives. DimLex is used for automatic discourse parsing, and also for semiautomatic text annotation using the ConAno tool (Stede and Heintze, 2004). A similar repository for French is LEXCONN (Roze et al., 2012)[4], which contains more than 300 connectives with their syntactic categories and discourse relations from Segmented Discourse Representation Theory (Asher and Lascarides, 2003). The lexicon has been constructed manually, using a corpus as empirical support.
LICO is freely distributed under a CC-BY licence[5] and can be browsed with DIMLEX and LEXCONN at `http://connective-lex.info/`.

4. Enriching Connectives of Contrast in LICO

We aim at enriching the connectives of contrast in LICO with examples from corpora. Collecting these examples, we can observe in context how each connective is used: for instance, if the connective is used at the beginning of a sentence, if it requires the following verb to be in a conjunctive form, etc.
We focus on the connectives signalled in LICO as conveying a contrast relation. As we said, for each entries in LICO, the semantic relation the connectives convey is signalled adopting the PDTB schema of sense tags: we include in our research those that are tagged in LICO with the COMPARISON.*Contrast* tag (e.g. *contrariamente a*) and the COMPARISON.*Concession* tag (e.g. *nonostante*). For convenience, we will refer to these senses as CONTRAST and CONCESSION.
The entries tagged with one of these two senses (or both) are 38. For each of them, we retrieve 5 examples from a corpus; and, for those that are polysemous, we retrieve 5 examples for each of the senses they convey according to LICO. For example, the connective *mentre* has been tagged with CONTRAST, CONCESSION, TEMPORAL thus we retrieve 5 examples for the connective in these three senses. This lead us to complete the information about each connective with useful examples of its usage in all its senses.
In order to collect corpus-driven examples, we use two strategies: i) we retrieve examples from the resource Contrast-Ita Bank; ii) we pick examples from the bigger corpus from which Contrast Ita-Bank is derived and from Wikipedia documents.

Connectives and Examples from Contrast-Ita Bank. Contrast-Ita Bank (Feltracco et al., 2017) is a corpus of 169 news (65,053 tokens) annotated with explicit and implicit discourse contrast relations in Italian.
More specifically, the documents correspond to articles published in a local newspaper "L'Adige" in two different days and include reports, news about politics, news about economics, sport results. They contain narrations and quotes of oral interviews.
Contrast-Ita Bank (henceforth CIB) follows the schema proposed in the PDTB guidelines (Prasad et al., 2007) both in terms of sense tags, i.e., CONTRAST and CONCESSION are tagged in the corpus, and in terms of information annotated, i.e., for each explicit relations, the connective that conveys the relation is marked together with its arguments (named *Arg1* and *Arg2*)[6]. For instance, in Example (1) the connective is underlined, *Arg1* is in italics, and *Arg2* is in bold.

(1) *Il ministro del Lavoro e delle Pensioni britannico, Andrew Smith, ha rassegnato ieri le dimissioni* nonostante **i tentativi del premier Tony Blair di convincerlo a rimanere**.[7]
tag: CONCESSION.Arg1.as.denier

In our work we take advantage of the information associated to the connectives of contrast in CIB. In fact, the annotated connective (marked as CONTRAST or CONCESSION) together with its arguments constitute the examples we retrieved for the enrichment of LICO. For instance, Example (1) from CIB has been retrieved as an example of *nonostante* to enrich LICO. Moreover, we can get information of how the connective is used with reference to the relation it conveys; for example, we can inspect if it is found between the arguments it links, before them, if it requires a conjunctive form of the verb just in its *arg2*, etc... We keep this data in LICO by taking care of reporting the span of text of the two arguments as part of the example and by encoding the *token_id* of the connective as registered in CIB: this works as a pointer for users who can reconstruct the entire annotation of the contrast relation in CIB.

Corpus examples picking. Not all the connectives tagged as CONTRAST and CONCESSION in LICO are present in CIB, and the resource does not provide 5 examples for all the connectives of contrast it contains (some of them appear just once). Moreover, we want to retrieve examples also for the non contrastive use of the connectives, and these are not tagged in CIB. To reach our goal, we extend the search to other documents. On one hand, we con-

[3] `https://github.com/discourse-lab/dimlex/`
[4] `http://www.linguist.univ-paris-diderot.fr/~croze/D/Lexconn.xml/`
[5] `https://hlt-nlp.fbk.eu/technologies/lico`
[6] Specifically, *Arg2* is the argument that is syntactically bound to the connective, and *Arg1*, the other one (Prasad et al., 2007).
[7] Eng: *Secretary of State for Work and Pensions Andrew Smith resigned yesterday*, despite **Prime Minister Tony Blair's attempts to persuade him to stay.**

entry-id	4
orth	cont
part	phrasal
	a dire la verità
orth	cont
part	phrasal
	in verità
orth	cont
part	phrasal
	per la verità
POS	adverbial
sem relation	COMPARISON:Concession:Arg2-as-denier
examples	id="1" source = "CIB_405635_185,186" text: "La nostra idea - dice Rita - è quella di aprire un bar normale. In cui il pubblico possa consumare anche qualche cibo precotto, patatine fritte, poi i soliti panini. Magari ascoltandosi il juke-box". In verità, proprio normale, normale non sarà il bar. id="2" source = "Adige_414186" text: Sulla carta l'opposizione è di dodici consiglieri. In verità siamo rimasti in cinque id="3" source = "wiki_34540_Piario" text: Soltanto nel 1792 si verificò la definitiva divisione di quello che era il comune di "Oltressenda bassa" in Piario e Villa d' Ogna . A dire la verità i due centri vennero nuovamente uniti in un' unica realtà amministrativa durante il periodo fascista (anno 1929) , per poi separarsi in modo definitivo nel 1958 . id="4" source = "wiki_39733_Papa_Pio_XII" text: .. id="5" source = "wiki_1041014_Divisione_Nazionale" text: ..
sem relation	EXPANSION:Level-of-detail:Arg2-as-detail example id="1" source = "Adige_413952" text: ..

Figure 1: The connective *a dire la verità* and its variants in LICO.

sidered other 357 documents of the newspaper "L'Adige" (same source of CIB); we will refer to this source as *Adige*. On the other hand, we search for additional examples in documents from Wikipedia[8]. While the retrieving has been done automatically, the selection of the examples has been conducted manually. This is because we need to distinguish cases in which the connective plays such a role from cases in which it does not introduce a discourse relation. These latter cases are also known as *discourse markers* already mentioned in Section 2, and are used, for instance, to take the turn in a conversation (interactive function) or as indicators of reformulation (metatextual function) -see (Bazzanella, 1995) and (Ferrari, 2010). Once the connective is identified, we also need to disambiguate it, in order to associate it with the sense of the relation it is actually conveying. A manual annotation is thus needed for the creation of a resource of reference for both linguistic and computational uses.

5. Results

The results of our work will be presented in three sections considering that: the format used for registering the information in LICO has been updated since we introduce new elements; a new list of connectives has been created since we validated and modified the first version of LICO; a revision of the polysemy of the connectives has been carried out with the new data.

The resulting format. Figure 1 shows the connective *a dire la verità* and its variants in LICO. In the central part of the figure, we can see how examples from the three different sources (i.e. CIB, Adige, Wikipedia) have been reported in LICO. Specifically, the examples have their own id and are identified with the tag "source" which attribute is a two or three position code standing for: *i)* the source corpus (i.e. CIB or Adige or Wiki), *ii)* the source document id as identified in the source corpus, *iii)* just for example from CIB, the id of the tokens of the connective in the source document. For instance, in Figure 1 the first example is from document 405635 of CIB, and in that document the connective corresponds to tokens 185 and 186. Notice also that the source documents are distributed with LICO, as external material at users disposal.

A new list of connectives of contrast. One important benefit of considering CIB, a corpus that has been exhaustively annotated with contrast relations, concerns the enrichment of the list of 38 connectives of contrast in LICO. As can be seen in Table 2, five of the connectives tagged with CONTRAST or CONCESSION, or both relations, in CIB are not present in LICO. More precisely *al contrario di* and *seppure* were not present in LICO as connectives, while *e*, *in realtà* and *se* were in LICO but associated to a non contrastive sense.

On the other side, the corpus investigation bring us to eliminated 3 connectives from this list: *con tutto questo*, *a onor del vero*, *persino*. The three of them were found in the corpus as connectives but not conveying a contrast relation; they have been kept in LICO as connectives of the PDTB sense EXPANSION:Conjunction. [9]

[8]Documents are from Italian Wikipedia, February 2010.

[9]A deep examination highlights that the first two were retrieved from the resource Sabatini Coletti, which does not specify the relation they convey. The assignment to the contrast sense is probably derived by the fact that their synonyms (i.e. *nonostante tutto* for *con tutto questo* and *a dire la verità* for *a onor del*

connective	CIB	L	connective	CIB	L
a dire la verità (in verità)	x	x	eppure	x	x
a dire il vero		x	in realtà	x	
a dispetto di	x	x	invece	x	x
a onor del vero		x	ma	x	x
ad ogni modo		x	malgrado		x
al contrario	x	x	malgrado ciò		x
al contrario di	x		mentre	x	x
anche		x	nondimeno		x
anche se	x	x	nonostante	x	x
benché		x	per contro		x
bensì		x	per quanto		x
ciononostante		x	però	x	x
cionondimeno		x	persino		x
comunque	x	x	pur / pure	x	x
con tutto questo		x	quantunque		x
contrariamente a		x	se	x	
da un canto.. dall'altro		x	sebbene		x
da un lato.. dall'altro lato		x	sennonché		x
da una parte.. dall'altra parte	x	x	seppure	x	
e	x		solamente		x
			solo		x
			tuttavia	x	x
			viceversa		x
			Total	19	38

Table 2: Connectives of contrast in Contrast-Ita-Bank (CIB) and in LICO (L). Three were removed as they do not appear to convey the contrast relation.

We thus update the list of connectives (of contrast) in LICO including the connectives form Contrast-Ita Bank and discarding those find not conveying this relation in the corpus; the final list is 40 connectives (38 in LICO - 3 removed + 5 from CIB). This result, even limited, highlights the importance of a corpus investigation in order to enrich a lexical resource.

Checking polysemy The use of corpora not only permits to discover new connectives of contrast or contrast uses of connectives that was already in LICO, but it lead us to review the polysemy of the already listed connectives of contrast. For example, we add the sense EXPANSION:Exception:Arg2-as-except to the connectives *solamente* and *solo* (both Eng. 'only') since we find example as the following, in which it introduces an exception.

(2) L'attaccante quindi genera un certificato server falso, totalmente uguale al certificato vero, solamente che non è firmato dalla stessa CA.[10]

Table 3 shows the data of the enrichment. Notice that globally the average number of examples for each entry almost doubled, even though we added only two entries to the resource. As already specified, the corpus analysis also lead us to discover new senses for the connectives under examination (for a total of 214 senses over 175 entries with respect to the previous 205 over 173 connectives).

Data	Pre	Post
# Connectives of Contrast in LICO	38	40
Average polysemy of the connectives in LICO	1.18	1.22
# examples per connective sense (average)	1.73	3.37

Table 3: Data on connectives of contrast in Lico pre and post corpus enrichment.

6. Conclusion and Further works

We have presented our project aiming at enriching the pre-existing resource LICO (Feltracco et al., 2016), a lexicon of discourse connectives for Italian, with real corpus data for connectives marking contrast.

The adopted methodology partially takes advantage of a pre-existing resource in which discourse connectives of contrast are annotated, along with the arguments of the discourse relation they make esplicit. A complementary investigation has been conducted picking examples of the connectives in corpora and manually disambiguation their senses in the retrieved textual contexts. In particular, this latter strategy is replicable to enrich LICO with information about discourse connectives that convey relations other than contrast (e.g. temporal, causal) and it can also be adopted to enriched lexica of connectives in other language.

Corpus investigation can also be carried out in a more automatic way: for example, Bourgonje et al. (2017) use parallel corpus to discover correspondences between connectives in different language and highlight point to gaps in the examined resources. In particular, the authors report on experiments to validate the list of connectives in DimLex (Stede, 2002) and LICO in an effort to constructing a bilingual lexicon on connectives that are connected via their discourse senses.

In our case, since we want to extract clear examples and disambiguate their senses in the context, we believe the manual disambiguation of connectives was necessary. The resulting resource represents a valuable tool for both linguistic analyses and the training of a classifier for NLP applications.

7. Bibliographical References

Asher, N. and Lascarides, A. (2003). *Logics of conversation*. Cambridge University Press.

Bazzanella, C. (1995). I segnali discorsivi. *Grande grammatica italiana di consultazione*, 3:225–257.

Bourgonje, P., Grishina, Y., and Stede, M. (2017). Toward a bilingual lexical database on connectives: Exploiting a german/italian parallel corpus. In *Proceedings of the Fourth Italian Conference on Computational Linguistic (CLiC-it 2017)*.

Feltracco, A., Jezek, E., Magnini, B., and Stede, M. (2016). LICO: A Lexicon of Italian Connectives. In *Proceedings*

vero) are in fact conveying the contrast relation; in context however, they seems to convey more the EXPANSION:Conjunction relation. For what concerns *persino*, it has been retrieved as a translation of the German *auch* which however, is not associated with the contrast relation in the DimLex lexicon; we do not find example for its use as contrastive.

[10]Eng. "The attacker then generates a fake server certificate, totally equal to the true certificate, only that it is not signed by the same CA."

of the Third Italian Conference on Computational Linguistic (CLiC-it 2016), page 141.

Feltracco, A., Magnini, B., and Jezek, E. (2017). Contrast-ita bank: A corpus for italian annotated with discourse contrast relations. In *Proceedings of the Fourth Italian Conference on Computational Linguistic (CLiC-it 2017)*.

Ferrari, A. and Zampese, L. (2000). *Dalla frase al testo : una grammatica per l'italiano*. Zanichelli.

Ferrari, A. (2010). Connettivi. In *Enciclopedia dell Italiano*. diretta da Raffaele Simone, con la collaborazione di Gaetano Berruto e Paolo D Achille, Roma, Istituto della Enciclopedia Italiana.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). The Penn Discourse Treebank 2.0 Annotation Manual.

Roze, C., Danlos, L., and Muller, P. (2012). LEXCONN: a French lexicon of discourse connectives. *Discours. Revue de linguistique, psycholinguistique et informatique.*, (10).

Sabatini, F. and Coletti, V. (2007). Dizionario della lingua italiana 2008 (2007). *Milano: Rizzoli Larousse*.

Scheffler, T. and Stede, M. (2016). Adding Semantic Relations to a Large-Coverage Connective Lexicon of German. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC '16)*, Portoroz, Slovenia, May.

Stede, M. and Heintze, S. (2004). Machineassisted rhetorical structure annotation. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 425–431, Geneva.

Stede, M. and Umbach, C. (1998). Dimlex: A lexicon of discourse markers for text generation and understanding. In *Proceedings of the 17th international conference on Computational linguisticsVolume 2*, pages 1238–1242. Association for Computational Linguistics.

Stede, M. (2002). Dimlex: A lexical approach to discourse markers. In *Exploring the Lexicon - Theory and Computation*. Edizioni dell'Orso, Alessandria.

Stede, M. (2012). *Discourse processing*. Morgan & Claypool Publishers.

Webber, B., Prasad, R., Lee, A., and Joshi, A. (2016). A Discourse Annotated Corpus of Conjoined VPs. *LAW X*, page 22.

SimPA: A Sentence-Level Simplification Corpus for the Public Administration Domain

Carolina Scarton, Gustavo Henrique Paetzold, Lucia Specia
Department of Computer Science, University of Sheffield
Regent Court, 211 Portobello, Sheffield, S1 4DP, UK
{c.scarton, g.h.paetzold, l.specia}@sheffield.ac.uk

Abstract

We present a sentence-level simplification corpus with content from the Public Administration (PA) domain. The corpus contains $1,100$ original sentences with manual simplifications collected through a two-stage process. Firstly, annotators were asked to simplify only words and phrases (lexical simplification). Each sentence was simplified by three annotators. Secondly, one lexically simplified version of each original sentence was further simplified at the syntactic level. In its current version there are $3,300$ lexically simplified sentences plus $1,100$ syntactically simplified sentences. The corpus will be used for evaluation of text simplification approaches in the scope of the EU H2020 SIMPATICO project – which focuses on accessibility of e-services in the PA domain – and beyond. The main advantage of this corpus is that lexical and syntactic simplifications can be analysed and used in isolation. The lexically simplified corpus is also multi-reference (three different simplifications per original sentence). This is an ongoing effort and our final aim is to collect manual simplifications for the entire set of original sentences, with over 10K sentences.

Keywords: text simplification, simplification corpora, public administration

1. Introduction

Text simplification (TS) is the task of reducing lexical and/or structural complexity of texts (Siddharthan, 2004). It is common to divide this task in two: lexical simplification (LS) and syntactic simplification (SS). LS deals with the identification and replacement of difficult words or phrases, while SS focuses on making complex syntactic structures simpler, e.g. by changing passive into active voice or splitting a sentence with coordination in two sentences.

LS has been widely explored in recent years. Work include simple word frequency-based approaches (Carroll et al., 1998; Carroll et al., 1999; Biran et al., 2011), unsupervised approaches that use word embeddings (Glavaš and Štajner, 2015; Paetzold and Specia, 2016c), and supervised approaches (Horn et al., 2014; Paetzold and Specia, 2017). For SS, some approaches apply hand-crafted rules (Siddharthan, 2011; Candido Jr. et al., 2009; Bott et al., 2012; Brouwers et al., 2014; Barlacchi and Tonelli, 2013; Scarton et al., 2017), while others use parallel data to learn simplification operations (Woodsend and Lapata, 2011; Paetzold and Specia, 2013; Siddharthan and Angrosh, 2014; Zhu et al., 2010; Coster and Kauchak, 2011; Wubben et al., 2012; Narayan and Gardent, 2014; Xu et al., 2016; Zhang and Lapata, 2017; Nisioi et al., 2017). Corpus-based approaches tend to learn both LS and SS transformations jointly.

For English, two main parallel corpora exist: Simple Wikipedia (Zhu et al., 2010) and Newsela (Newsela, 2016), the latter with simplifications performed by professionals. However, such corpora do not distinguish among different types of simplification (i.e. lexical from syntactic transformations). Moreover, they cover general domain texts, and therefore may not be sufficient to model operations for specific domains.

In terms of specific corpora for TS evaluation, only a few exist, mostly for English LS (Horn et al., 2014; De Belder and Moens, 2012; Paetzold and Specia, 2016c; Paetzold and Specia, 2016a), all of which are composed of a sentence, a target complex word, and candidate substitutions ranked by simplicity. Although these have been used in many papers, their sentences are extracted either from Wikipedia or news articles, making them also general domain. The candidate substitutions in most of these datasets were suggested and ranked by native English speakers, which means that they do not necessarily capture the needs of specific audiences, such as non-native speakers.

For corpus-based TS – including both LS and SS – Zhu et al. (2010) released a test set of 100 original sentences and 131 simplified sentences, a subset from the Simple Wikipedia corpus. Xu et al. (2016) released a test set of 350 sentences with nine simplifications each sentence. They also used a subset of Simple Wikipedia, but containing only 1-to-1 aligned sentences, i.e. they disregarded sentence splitting, a very common operation where one sentence is broken into two or more.

These datasets are therefore either very small, have only one reference simplification, or do not cover all types of simplification. Simplification is a complex process and often more than one possible way of modifying a sentence is possible and acceptable. In addition, existing datasets are not suitable for approaches targeting a specific domain or user type. In the SIMPATICO project[1] we address the simplification of Public Administration (PA) content, such as websites that describe services, citizen rights and duties. Among the target audiences are non-native speakers of English. Our ultimate goal is to be able to provide personalised lexical and syntactic simplifications for each this target audience. In order to tackle the lack of evaluation data, we introduce SimPA: an English sentence-level TS evaluation corpus for the PA domain.

The SimPA corpus is under construction. The ultimate goal

[1] https://www.simpatico-project.eu

is to have at least one simplification for each original sentence in an entire corpus of $10,708$ original sentences that cover different types of PA-related content. he resulting corpus could be used not only for test, but also for development or fine-tuning of corpus-based approaches trained on corpora from other domains, such as the Wikipedia corpus. The current release contains $1,100$ PA domain sentences with three lexically simplified versions each, and one version further annotated with syntactic simplification. Lexical and syntactic simplification were done separately in order to build a resource with information about both tasks independently. By isolating such operations, our corpus can be directly used to evaluate specialised systems, that only perform either lexical or syntactic simplification. Nonetheless, this characteristic of the corpus should not impact its use for the evaluation of systems that perform both operations together. An user of the corpus just needs to be aware of which version of the corpus to use for each purpose. In addition to evaluation, this corpus will help with the analysis and profiling of domain-specific simplifications for the better design of simplification systems that better suit the purposes of our target audience.

Even in its current status, SimPA is the largest and most varied dataset of its kind, with multiple references for the LS version. SimPA is freely available under a Creative Commons Licence.[2]

2. Corpus Creation

In order to create SimPA, we first collected sentences from the Sheffield City Council (SCC), website[3] which is one of the partner PAs in the SIMPATICO project. Our crawler visited over 9K links, resulting in around 14K sentences. We then filtered the sentences to eliminate those without verbs, such as titles, menu items, and incomplete sentences. We also removed repeated sentences. This resulted in $10,708$ sentences, which we refer to as the 10K corpus.

In order to start with potentially more challenging sentences, maximising the opportunities that annotators would have to perform both lexical and syntactic simplifications, we sorted the 10K sentences according to their length (longer sentences first). The 5K longest sentences were then shuffled and $1,100$ sentences were selected to be annotated. Table 1 shows some statistics about the 10K, 5K and 1.1K sets.

	# tokens	# sentences	tokens per sentence
10K	249,954	10,708	23.34
5K	153,680	5,000	30.74
1.1K	33,492	1,100	30.45

Table 1: Statistics of the PA corpus.

2.1. Gathering Lexical Simplifications

The first step was to obtain lexical simplifications for our PA sentences. We define lexical simplifications as any word and phrase replacements that make the sentence more easily understood. In other words, a lexical simplification modifies the sentence locally, without altering sentence grammaticality or compromising meaning. We have collected three lexically simplified versions of each of our 1,100 PA subset sentences. This allows the creation of a more reliable evaluation dataset and the analysis of how people with different profiles attempt to minimise the lexical complexity of a sentence.

The data annotation was conducted using volunteers, all students and academic staff from universities, who considered themselves fluent speakers of English (native and non-native). Annotators were anonymised but asked to inform some demographic information (Section 3.1.).

Each annotator received 20 sentences to simplify on a online form created using the Google Forms platform.[4] They were given the following guidelines: "*Replace any words and phrases that you find complex in each sentence. Simpler words or phrases are those that you think can be more easily understood by readers, especially non-native speakers of English. Do not change the sentence in any other way.*"

In order to encourage annotators to produce reliable annotations, they could enter a £50 prize draft by providing their email. Inspecting the annotations, we found and replaced 229 spurious simplified sentences, most of which were produced by untrustworthy annotators.

The current version of SimPA has a total of $3,300$ lexically simplified PA sentences (3 versions of $1,100$ PA sentences). Table 2 shows some examples, where words in bold are the changes made by annotators (red is used to mark the original words/phrases and blue is used to mark the simplifications). They include replacements of single words (such as "experiencing" vs. "hearing"), as well as phrases (such as "a period of public consultation" vs. "consulting the public").

2.2. Gathering Syntactic Simplifications

The next step was to obtain syntactic simplifications for the lexically simplified sentences. These are any transformations that alter the syntactic structure of the sentence, such as splitting, passive-to-active voice transformation, anaphoric resolution, information reordering, etc. For this stage, thus far we collected only one simplification per sentence.

We selected one lexically simplified version of each original $1,100$ sentences from our previous annotation step. For cases with more than one distinct simplification, we randomly selected one. In this way, we always selected a lexically simplified sentence, if available. Although the sentence selection process could have involved some kind of readability assessment, we opted for the random approach, since we assumed all three simplifications are valid and correct. $1,079$ sentences from the final set featured at least one lexical simplification, only 21 did not.

Much like in the previous step, the annotators were students and academic staff from universities, and we also used the Google Forms platform. The same demographic data was collected and the number of sentences to simplify given to

[2] https://github.com/SIMPATICOProject/simpa

[3] https://www.sheffield.gov.uk

[4] https://www.google.co.uk/forms/about

Original:	If you're **experiencing** excessive noise form a **commercial property** (such as a pub, club, restaurant or cafe) you can report it to us.
Simplified:	If you're **hearing** excessive noise from a **business** (such as a pub, club, restaurant or cafe) you can report it to us.
Original:	After **a period of public consultation**, we adopted the **appraisal** and accompanying management **proposals** on 23rd October 2007.
Simplified:	After **consulting the public**, we adopted the **review** and accompanying management **plans** on 23rd October 2007.
Original:	In my personal opinion, I **don't** think the Housing Service is **given** enough **credit** for the amount of work they do for their tenants.
Simplified:	In my own opinion, I **do not** think the Housing Service is **praised** enough for the amount of work they do for their tenants.
Original:	**Where** agreement cannot be reached at the **compliance** stage and **liability** orders are moved to the enforcement stage a further £235 will be added to the **debt** owed.
Simplified:	**If** agreement cannot be reached at the **verification** stage and **responsibility** orders are moved to the enforcement stage a further £235 will be added to the **bill** owed.
Original:	The review considered **the suitability of** the technical standards being used by Local Planning Authorities and proposed a **radical** reduction in the number of **eligible** standards.
Simplified:	The review considered **how good** the technical standards being used by Local Planning Authorities **were** and proposed a **big** reduction in the number of **acceptable** standards.

Table 2: Example of manual LS

Original:	According to law, a successful challenge would result in a acquisition exercise **in which** the challenger could take part along with other interested organisations.
Simplified:	According to law, a successful challenge would result in an acquisition exercise**. In this exercise** the challenger could take part, as well as other interested organisations.
Original:	Within the 28 days application period we will talk with South Yorkshire Police for any comments and take into account any rules.
Simplified:	We will talk with the South Yorkshire Police for any comments**. This will be done** within the 28 days application period**. We will then** take into account any rules.
Original:	The number of dogs and cats that may be accommodated **will be specified** on the licence along with any other specific conditions.
Simplified:	The license **will have** the number of dogs and cats that can be accommodated along with any other specific conditions.
Original:	If required, the developer has to get together an ES describing the likely effects of the development on the environment and suggested measures to reduce problems.
Simplified:	If required, the developer has to get together an ES**. This describes the possible** effects of the development on the environment**. It also includes** suggested measures to reduce problems.
Original:	**If** you can pay the full costs of your care and support without any financial help from us, **then** you are considered as a self-funder.
Simplified:	You are a self-funder **if** you can pay for your own care and support without receiving financial help.

Table 3: Example of manual SS

each annotator was also 20. They were provided with the following guidelines: "*Apply simplification operations to each sentence. You can split it, rewrite it, or reorder its information. Please avoid adding extra information or deleting parts of the sentence (unless it is extremely necessary). The goal is for the simplified sentence to have the same meaning as the original sentence.*". Annotators could again enrol in a £50 prize draft upon completion of the task.

Examples of annotations are shown in Table 3. The first line shows a case where a relative clause was split into two sentences. The original sentence in the second line was split into three sentences where two conjoint clauses (one temporal "within 28 days" and one of addition "and") were split. In the third line, the original sentence was changed from the passive into active voice. The fourth line also contains examples of splitting, where the original sentences was split into three. Finally, the fifth sentence was reordered (the "if" clause changed places with the "then" clause).

3. Data Analysis

3.1. Annotators

Since our aim is to create models capable of generating personalised simplifications, the demographic information we collected becomes very important. We can, for instance, identify which words or syntactic structures were modified by readers with a certain native language, or who are from a given country, or have the same proficiency with English. While we acknowledge that our sample is rather small to

draw conclusions, we can use the data to already provide insights on the perception of different readers when it comes to text complexity. The demographic questions were:

- Age;
- Country of birth;
- Native language (with the option to include up to three languages);
- Educational level;
- Proficiency in English (following the CEFR[5] scale);
- Familiarity with the PA e-services; and
- Occupation.

Although all volunteers are part of the academic environment, they still have very diverse backgrounds and profiles. In addition, volunteers included undergraduate and postgraduate students, and member of staff with a variety of educational levels. Finally, the volunteers were from different faculties of universities, including humanities, sciences, engineering, medical school, among others.

176 volunteers participated in the first annotation stage (lexical simplification). Figure 1 illustrates the distributions for each demographic aspect.[6] The number of volunteers that have more than one native language is 30.

For the second stage (syntactic simplification), 85 volunteers participated in the task. Figure 2 shows the distribution of each demographic aspect for all the annotators in the second phase. 11 volunteers reported having more than one native language.

For both stages, the majority of the volunteers are British and have English as their native language (which is expected given the survey distribution channels used). Brazil is the country with the second highest proportion of volunteers in both stages. The second most common native language was Portuguese, followed by Other/Unlisted and Chinese, respectively.

In general, the majority of the volunteers are between 18 and 30 years old. In the first stage, we had more undergraduate students, whilst in the second stage the majority were postgraduate students. In both experiments, the large majority of the volunteers are either C1 or C2 in the CEFR scale. This is also expected as we requested fluent speakers of English. Finally, more than half of our volunteers had no experience with PA e-services.

3.2. Simplification Data

The lexical simplifications for the $1,100$ sentences have $33,301$ tokens in total, which is 191 tokens less than the original sentences. This indicates that the majority of the changes were at word level and only a few phrases were simplified into single words. The syntactic simplifications have $32,219$ tokens, which is $1,082$ tokens less than their lexically simplified versions. This is expected, since some syntactic transformations tend to lead to the removal of unimportant words.

[5] http://www.coe.int/en/web/common-european-framework-reference-languages

[6] We omitted the data for occupation as it was very sparse.

Table 4 shows readability scores calculated for the original, lexically simplified and syntactically simplified sentences in SimPA. The basic counts of syllables per words (tokens except punctuation), words per sentence and content words (nouns, adjectives, verbs and adverbs) per sentence show that the LS version has less words than the original and that such words are shorter.

	Original	LS	SS
Syllables per word	1.91	1.85	1.86
Words per sentence	24.04	23.91	22.76
CW per sentence	16.15	15.90	11.90
Age of acquisition ↓	316.58	306.26	299.29
Familiarity ↑	439.54	445.32	440.36
Imageability ↑	315.37	317.64	314.86
Concreteness* ↑	298.90	299.83	297.12
Flesch Reading Ease ↑	44.8	48.6	100

Table 4: Readability metrics comparing the original and two simplified corpus. * indicates no statistically significant difference according to t-test with $p < 0.05$

We also evaluated the data using four psycholinguistics metrics: age of acquisition, familiarity, imageability and concreteness, that were extracted using the approach proposed by Paetzold and Specia (2016b). Age of acquisition is significantly smaller for the lexically simplified sentences, meaning that the words used are usually learned at a younger age. Familiarity and imageability are significantly higher for them also, which suggests that the words and phrases used to replace segments are simpler than the original.

The ratio of syllables per word is almost the same for lexically and syntactically simplified sentences. The number of words and content words per sentence is smaller in syntactic than in lexical simplifications, since volunteers shortened the sentences during syntactic simplification. The age of acquisition score is significantly smaller for syntactic simplifications than for lexical ones, which would suggest that a lot of syntactic simplifications also encompass word replacements. However, the values for familiarity, imageability and concreteness are smaller for syntactic simplifications, which contradicts this hypothesis.

According to Flesch, lexical simplifications were shown not to greatly affect the readability of sentences, increasing their scores by no more than 3.8 points. Although the difference is small, it is statistically significant (t-test with $p < 0.05$). Syntactic simplifications, on the other hand, achieved a much more impressive gain of 55.2 in Flesch, more than doubling the original sentences' readability.

4. Discussion

We presented the first version of SimPA: a dataset for the analysis and evaluation of simplifications for content from the public administration domain. The dataset is currently composed of three lexically simplified versions of $1,100$ PA sentences each, as well as one syntactically simplified version of $1,100$ lexically simplified sentences.

Our analyses reveal that SimPA contains simplifications produced by people with various backgrounds, meaning that it can be used to evaluate the performance of simplifiers

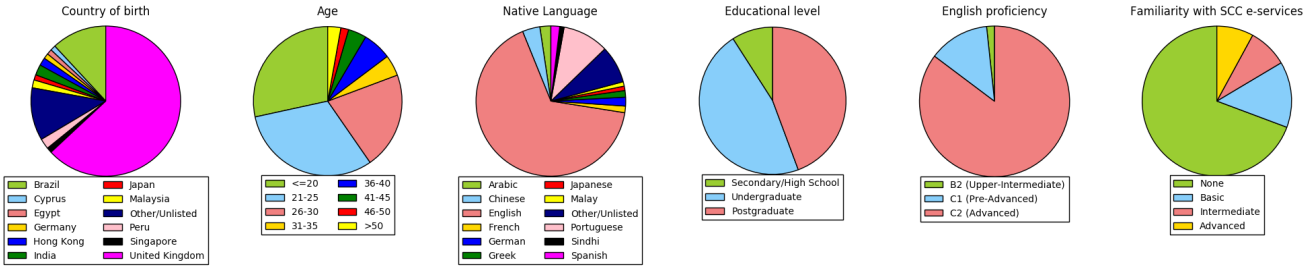


Figure 1: Demographic data of the lexical simplification experiment.

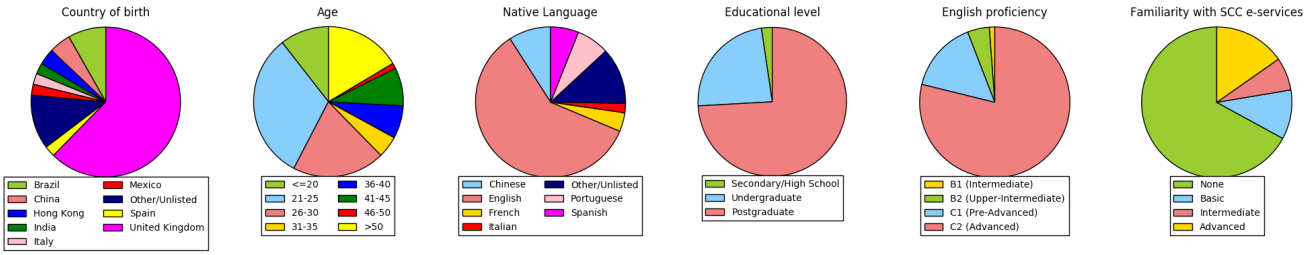


Figure 2: Demographic data of the syntactic simplification experiment.

that produce personalised output. By looking at psycholinguistic properties we found that the lexical simplifications feature many word and phrase replacements that reduce the overall age of acquisition, as well as increase the overall familiarity, imageability and concreteness of the original sentence. We also found that syntactic simplifications lead to sentences that are more than twice as readable as both the original and lexically simplified versions.

As future work, we intend to gather simplifications for the remaining of our corpus (9,608 sentences), in order to build a larger dataset for development or fine tuning purposes. For that, to make the process more efficient we will gather both lexical and syntactic simplifications in a single step (instead of splitting the process in lexical and syntactic simplification stages).

Acknowledgements

This work was supported by the EC project SIMPATICO (H2020-EURO-6-2015, grant number 692819).

5. Bibliographical References

Barlacchi, G. and Tonelli, S. (2013). Ernesta: A sentence simplification tool for children's stories in italian. In *Proceedings of the 14th CICLing*, pages 476–487.

Biran, O., Brody, S., and Elhadad, N. (2011). Putting it simply: a context-aware approach to lexical simplification. In *Proceedings of the 49th ACL*, pages 496–501.

Bott, S., Saggion, H., and Mille, S. (2012). Text Simplification Tools for Spanish. In *Proceedings of the 8th LREC*, pages 1665–1671.

Brouwers, L., Bernhard, D., Ligozat, A.-L., and François, T. (2014). Syntactic Simplification for French. In *Proceedings of the 3rd PITR*, pages 47–56.

Candido Jr., A., Maziero, E., Gasperin, C., Pardo, T. A. S., Specia, L., and Aluísio, S. M. (2009). Supporting the Adaptation of Texts for Poor Literacy Readers: a Text Simplification Editor for Brazilian Portuguese. In *Proceedings of the 4th NAACL HLT BEA*, pages 34–42.

Carroll, J., Minnen, G., Canning, Y., Devlin, S., and Tait, J. (1998). Practical simplification of english newspaper text to assist aphasic readers. In *Proceedings of 1998 AAAI Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10.

Carroll, J., Minnen, G., Pearce, D., Canning, Y., Devlin, S., and Tait, J. (1999). Simplifying text for language-impaired readers. In *Proceedings of the 9th EACL*, pages 269–270.

Coster, W. and Kauchak, D. (2011). Learning to Simplify Sentences Using Wikipedia. In *Proceedings of the MTTG 2011*, pages 1–9.

De Belder, J. and Moens, M.-F. (2012). A dataset for the evaluation of lexical simplification. In *Computational Linguistics and Intelligent Text Processing*, pages 426–437.

Glavaš, G. and Štajner, S. (2015). Simplifying lexical simplification: Do we need simplified corpora? In *Proceedings of the 53rd ACL*, pages 63–69.

Horn, C., Manduca, C., and Kauchak, D. (2014). Learning a lexical simplifier using wikipedia. In *Proceedings of the 52nd ACL*, pages 458–463.

Narayan, S. and Gardent, C. (2014). Hybrid Simplification using Deep Semantics and Machine Translation. In *Proceedings of the 52nd ACL*, pages 435–445.

Nisioi, S., Štajner, S., Ponzetto, S. P., and Dinu, L. P. (2017). Exploring neural text simplification models. In *ACL (2)*. The Association for Computational Linguistics.

Paetzold, G. H. and Specia, L. (2013). Text Simplification as Tree Transduction. In *Proceedings of the 9th STIL*, pages 116–125.

Paetzold, G. H. and Specia, L. (2016a). Benchmarking Lexical Simplification Systems. In *Proceedings of the 10th LREC*, pages 3074–3080.

Paetzold, G. H. and Specia, L. (2016b). Inferring psycholinguistic properties of words. In *Proceedings of the 2016 NAACL*, pages 435–440.

Paetzold, G. H. and Specia, L. (2016c). Unsupervised lexical simplification for non-native speakers. In *Proceedings of The 30th AAAI*, pages 3761–3767.

Paetzold, G. and Specia, L. (2017). Lexical simplification with neural ranking. In *Proceedings of the 15th EACL*, pages 34–40.

Scarton, C., Specia, L., Palmero Aprosio, A., Tonelli, S., and Martín Wanton, T. (2017). MUSST: A Multilingual Syntactic Simplification Tool. In *Proceedings of IJCNLP 2017*, pages 25–28, Taipei, Taiwan.

Siddharthan, A. and Angrosh, M. A. (2014). Hybrid text simplification using synchronous dependency grammars with hand-written and automatically harvested rules. In *Proceedings of the 14th EACL*, pages 722–731.

Siddharthan, A. (2004). *Syntactic simplification and text cohesion*. Ph.D. thesis, University of Cambridge, Cambridge, UK.

Siddharthan, A. (2011). Text simplification using typed dependencies: A comparison of the robustness of different generation strategies. In *Proceedings of the 13th ENLG*, pages 2–11.

Woodsend, K. and Lapata, M. (2011). Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of EMNLP 2011*, pages 409–420.

Wubben, S., van den Bosch, A., and Krahmer, E. (2012). Sentence simplification by monolingual machine translation. In *Proceedings of the 50th ACL*, pages 1015–1024.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., and Callison-Burch, C. (2016). Optimizing statistical machine translation for text simplification. *TACL*, 4:401–415.

Zhang, X. and Lapata, M. (2017). Sentence simplification with deep reinforcement learning. In *Proceedings of EMNLP*, pages 595–605.

Zhu, Z., Bernhard, D., and Gurevych, I. (2010). A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd COLING*, pages 1353–1361.

6. Language Resource References

Newsela. (2016). *Newsela Article Corpus*. Version: 2016-01-29, `https://newsela.com/data`.

Zhu, Z., Bernhard, D., and Gurevych, I. (2010). A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd COLING*, pages 1353–1361.

The brWaC Corpus: A New Open Resource for Brazilian Portuguese

Jorge A. Wagner Filho[1], Rodrigo Wilkens[2], Marco Idiart[3], Aline Villavicencio[1,4]
[1] Institute of Informatics, Federal University of Rio Grande do Sul, Brazil
[2] CENTAL, Université catholique de Louvain - UCL
[3] Institute of Physics, Federal University of Rio Grande do Sul, Brazil
[4] School of Computer Science and Electronic Engineering, University of Essex, UK
jawfilho@inf.ufrgs.br, rodrigo.wilkens@uclouvain.be, marco.idiart@gmail.com, avillavicencio@inf.ufrgs.br

Abstract

In this work, we present the construction process of a large Web corpus for Brazilian Portuguese, aiming to achieve a size comparable to the state of the art in other languages. We also discuss our updated sentence-level approach for the strict removal of duplicated content. Following the pipeline methodology, more than 60 million pages were crawled and filtered, with 3.5 million being selected. The obtained multi-domain corpus, named *brWaC*, is composed by 2.7 billion tokens, and has been annotated with tagging and parsing information. The incidence of non-unique long sentences, an indication of replicated content, which reaches 9% in other Web corpora, was reduced to only 0.5%. Domain diversity was also maximized, with 120,000 different websites contributing content. We are making our new resource freely available for the research community, both for querying and downloading, in the expectation of aiding in new advances for the processing of Brazilian Portuguese.

Keywords: Web as Corpus, large corpus, Brazilian Portuguese

1. Introduction

In recent years, initiatives for the construction of large corpora have attained ever-growing interest in the NLP community. They are especially relevant for applications which demand large volumes of data, such as neural methods (Pennington et al., 2014; Bahdanau et al., 2014; Johnson et al., 2016), and also tend to produce more reliable resources for statistical models. In this paper, we build on previous work (Wagner Filho et al., 2016) and aim to construct a large and freely available Web corpus for Brazilian Portuguese, compatible with the state of the art in other languages. An example of the need for new, large corpora in this language was shown by Rodrigues et al. (2016), who had to combine 19 different corpora to obtain a 1.7 billion tokens corpus and create a distributional semantics model comparable to those available for English. Besides language models, the corpus presented here can also be used, for example, in dictionary creation (Kilgarriff et al., 2008), word similarity (Levy et al., 2015) and word sense induction (Navigli and Velardi, 2010; Di Marco and Navigli, 2013).

A widely adopted approach for the construction of large corpora is the *WaCky* (*Web-As-Corpus Kool Yinitiative*) methodology (Bernardini et al., 2006; Baroni et al., 2009), which has been used to provide corpora in the scale of billions of tokens for multiple languages, extracting text content from the Web. It also enables the construction of corpora without domain biases, considering that a corpus will automatically get balanced after reaching a substantial size (Xiao, 2010).

When constructing these corpora a real concern is to gather as representative and diverse material from a domain as possible. Indeed, the *WaCky* pipeline incorporates strategies to avoid duplicated material in the resulting corpus. A widely adopted approach involves adopting a threshold for the number of overlapping n-grams randomly sampled between each pair of documents (Broder, 1997). However, the results depend on the accuracy and coverage of the sampling procedure in each of the documents, so that larger samples are more likely to produce more reliable indication of redundancy. On the other hand, comparing each two documents is a costly and time-consuming task, especially as the number of documents grows. Therefore, it is important to find a balance between the scale of redundancy detection comparisons and the degree of redundancy that is acceptable in the resulting corpus. In this paper, we analyse the amount of redundancy in existing WaC corpora and provide a quantitative analysis of corpus size as a function of redundancy. We also propose an approach for removing duplicated content taking into account a balance between corpus size (and the computational costs of tracking duplication) and content diversity.

In this work, we start with a discussion of resources created from the web (Section 2). In particular for Brazilian Portuguese, we describe how starting with a 1.6 billion token corpus (Wagner Filho et al., 2016) and following a strict methodology (Section 3), the resulting corpus is expanded to encompass 2.7 billion tokens. This result is discussed in terms of corpus size, domain diversity and content originality (Section 4). Final remarks are presented in Section 5. The resulting language resource, named *brWaC*, is freely available for research purposes, both for querying and downloading[1].

2. Related Work

With growing content availability in the Web, it became natural for researchers to look at it as a source to complement their traditional text repositories. Among notable examples to adopt this idea are the Terabyte corpus (Clarke et al., 2002) (53bi tokens) and the Web Text corpus (Liu and Curran, 2006) (10bi tokens). Nonetheless, these early re-

[1] www.inf.ufrgs.br/pln/brwac

sources often included large amounts of material that is of limited relevance (such as computer code) and duplicated materials. Moreover, content quality was not controlled. Recently, therefore, the focus in Web corpora construction shifted, from downloading large volumes of text to attaining corpus quality trough efficient post processing.

In this context, the *WaCky* (*Web-As-Corpus Kool Yinitiative*) methodology was proposed (Bernardini et al., 2006; Baroni et al., 2009). It includes four steps: (1) identification of seed URLs; (2) post-crawl cleaning; (3) removal of duplicated content; and (4) annotation. These are discussed in the next section.

Initially, four large corpora were created, targeting English, German, Italian (Baroni et al., 2009) and French (Ferraresi et al., 2010). In previous work, we have already explored this methodology to propose an initial version of the resource presented here, *brWaC*, targeting Brazilian Portuguese (Wagner Filho et al., 2016). We achieved a 1.5bi tokens version, which was also automatically classified in different readability levels. Initial efforts in this direction had already been implemented before (Boos et al., 2014), but a posterior rigorous processing of the collected content resulted in a small corpus with only 200mi tokens.

The *TenTen* Corpus Family (Jakubíček et al., 2013) is an initiative by Sketch Engine (Kilgarriff et al., 2004) to construct Web corpora for all major world languages. Currently, corpora for 31 different languages are available, including a 4 billion tokens corpus for Portuguese (*ptTenTen*). However, this corpus is not openly available, a key aspect of our research motivation. Moreover, it includes content from both the European and Brazilian variants of Portuguese, not being directly comparable to our proposal. Focusing on characteristics of the written language, while there is a large core that is shared among different Portuguese variants, there are also lexical and syntactic characteristics of Brazilian Portuguese that are marked in comparison to the others (Branco and Costa, 2007). For instance, comparing these two variants, there is a strong difference in the preference for the use of clitics. Subsequent analyses that use these corpora or models constructed from them may result in an amalgamation of different variants that does not reflect accurately the characteristics of any particular variant. This may create problems for downstream applications, like text simplification, for which the simplicity and naturalness of a text are linked to the language usage for that particular community of speakers. With this work we aim to produce a large corpus that targets Brazilian Portuguese.

3. Methodology

Here, we follow the four-step pipeline approach from the WaC methodology (Bernardini et al., 2006). Our implementation was based on the *Web as Corpus Toolkit*[2] (Ziai and Ott, 2005), an open source, modular and expansible toolkit which fitted our purposes well. New documents were collected and merged with our previous 1.5 billion corpus, following the steps in Sections 3.1 to 3.4.

[2] `http://wac-tk.drni.de`

3.1. URLs identification

Initially, a set of URLs is identified, employing queries to a search engine with random pairs of content words. The ten first results for each query are collected and expanded through a two-level recursion of the included links. Only *.br* top-level domain pages are considered, since we are targeting Brazilian Portuguese. Although this filter cannot completely ensure that documents from other variants are not occasionally included, it maximizes language homogeneity.

In this new collection, 8000 pairs of medium-frequency words from the Linguateca repository (Santos et al., 2004) were used. These resulted in 80k initial seeds obtained from the Microsoft Bing engine API. The final set, after link expansion, contained more than 38 million URLs.

3.2. Post-crawl cleaning

In a second moment, documents are filtered according to a series of criteria, such as size, and cleaned by filters of non-target content (e.g. HTML code, headers, footers and advertisement – also known as *boilerplate*). The density of stopwords and HTML tags is also controlled to ensure document quality (Pomikálek, 2013). A high minimum threshold of stopwords (25%) also ensures that selected documents are indeed in the targeted language.

3.3. Removal of duplicated content

In the third step of the pipeline, documents with intersection of content are detected and removed trough the pairwise comparison of all documents. The algorithm from Kilgarriff et al. (2006) was used. A global set of sentences is kept, and documents are processed linearly, counting large sentences (with more than 25 characters) which have already been seen previously. The percentage of these non-original sentences in a given document is then compared to a predefined maximum threshold. After empirical testing of different thresholds, we chose to tolerate a maximum of 10% of non-original large sentences, in order to maximize corpus quality.

We also modified this algorithm in order to filter sentences and include them in the global set immediately, rather than after the analysis of the whole document. This way, documents with high levels of intra-document duplication (e.g. pages from discussion lists) can also be detected and removed.

3.4. Annotation

Finally, all documents were annotated with syntactic information. To this end, the Palavras parser was used (Bick, 2000). The resultant annotated corpus was made available in both CoNLL and Moses formats.

4. Results

In this section, the obtained corpus is evaluated with relation to our three main requirements. Corpus size is comparable to those in other languages (Section 4.1), and domain biases should not be present (Section 4.2). Finally, duplicated content should be minimized (Section 4.3).

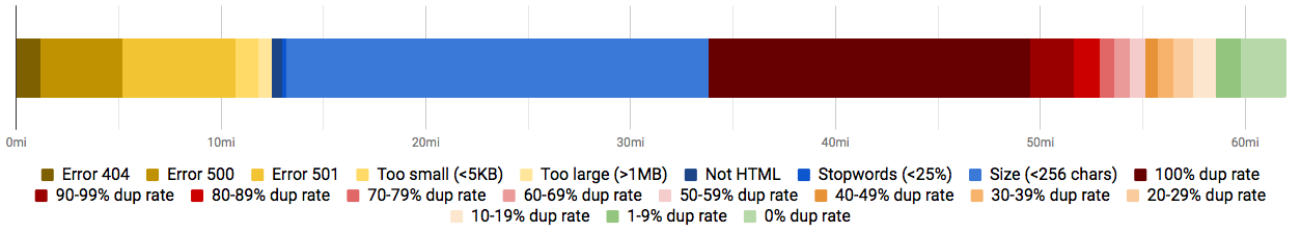


Figure 1: Documents filtered out in different steps of the pipeline. Strict rules resulted in only 5.6% of the original seeds being selected. The most common reasons for discarding were exact duplicates and small size after *boilerplate* removal.

Part of speech	% of tokens	% of types
Noun	26.38	69.652
Determiner	18.51	0.014
Preposition	17.71	0.005
Verb	15.55	13.048
Adjective	6.89	16.469
Conjunction	5.16	0.001
Adverb	5.01	0.668
Personal pronoun	2.14	0.007
Specifier	1.83	0.002
Numeral	0.63	0.019
Interjection	0.05	0.006

Table 1: Frequency of different parts of speech, in percentage of tokens and types, as annotated by the Palavras parser.

4.1. Corpus size

Around 95% of all documents initially crawled were discarded as a result of the strict set of filters, as shown in Figure 1. The main reasons were the existence of exact duplicates (100% content duplication rate) and content size smaller than 256 characters – this filter is applied after the *boilerplate* removal by the *jusText* component (Pomikálek, 2013). The large occurrence of duplicates is an expected consequence of the link recursion in the first step of the pipeline, which results in many similar seeds inside the same page domain. A detailed analysis on the presence of duplicated content in the final corpus is presented in Section 4.3.

Our collection process resulted in a corpus with 2.68 billion tokens (a 72% increase in relation to the previous version) and 5.79 million types, distributed in 145 million sentences and 3.53 million documents. These counts strictly exclude any tokens with numbers or special characters (except hyphens). Table 1 presents the frequency distribution of different parts of speech as annotated by the Palavras parser[3], also considering this restriction.

Table 2, on the other hand, presents a comparison between *brWaC* and other corpora: *frWaC* and *ukWaC*, which also follow the WaCky methodology, and *CETENFolha*[4], another corpus of Brazilian Portuguese. We can observe that *brWaC* is considerably larger than a typical corpus, such as *CETENFolha*, as expected, and is also larger amongst WaC corpora.

Corpus	#Documents	#Tokens	#Types
frWaC	2.20mi	1.02bi	3.9mi
ukWaC	2.69mi	1.91bi	3.8mi
brWaC	**3.53mi**	**2.68bi**	**5.8mi**
CETENFolha	340k	33mi	357k

Table 2: Size comparison with reference corpora. *brWaC* is considerably larger than a typical corpus, such as *CETENFolha*, and is also larger amongst other WaC corpora.

4.2. Domain diversity

The list of a hundred websites which contributed the highest number of documents to the corpus was annotated with the WebShrinker Website Categorization API[5] using the IAB taxonomy[6]. Results are shown in Figure 2. An analysis of the different content categories present indicate that the corpus was successful in fulfilling the prerequisite of being independent of domain. Another interesting observation was the relatively small number of occurrences of each of these pages. The most frequent website in the list contributed 70,348 documents, while the whole corpus includes 3.53 million. Moreover, in total, 121,075 websites are represented. This indicates that, as a result of the strict anti-duplication rules, website diversity was also maximized.

4.3. Content originality

In order to evaluate the efficiency of our approach for the removal of duplicated content (Section 3.3), we compared *brWaC* to two reference WaC corpora, *ukWaC* and *frWaC*, in terms of the occurrence of repeated sentences.

We found that, despite including considerably more sentences, our stricter approach resulted in smaller numbers of repeated sentences, especially of large size, in the *brWaC* corpus (see Table 3). While 6.3% and 9.3% of all sentences appear more than once in the British and French WaCs, respectively, in *brWaC* only 1.3% are repeated. When we consider only sentences longer than 20 tokens, the difference is even larger: 5.7% of non-unique sentences for *ukWaC*, 9% for *frWaC*, and 0.5% for *brWaC*. The log-scale sorted histograms in Figure 3 present a more detailed analysis of the repeated sentences. When considering all of them,

[3] A description of the parts of speech can be seen on `http://visl.sdu.dk/visl/pt/info/portsymbol.html`

[4] `http://www.linguateca.pt/cetenfolha`

[5] `https://www.webshrinker.com`

[6] `https://www.iab.com/guidelines/taxonomy/`

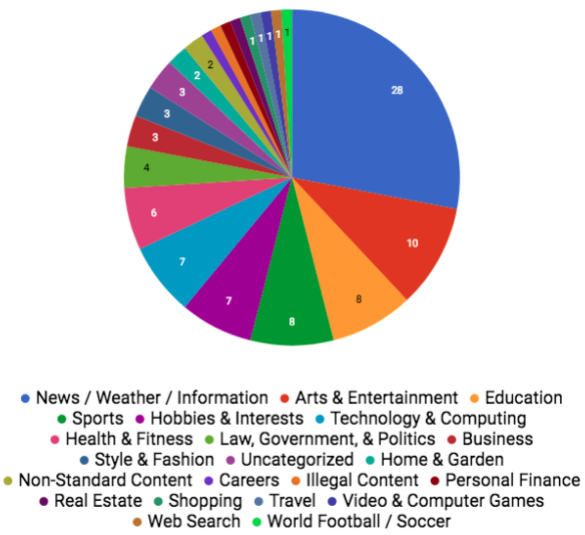


Figure 2: Annotated categories of the 100 most frequent websites in *brWaC* (legend is presented in descending order). Great diversity of content categories and relatively small number of occurrences of the most common pages indicate the fulfilment of our prerequisite of a corpus independent of domain.

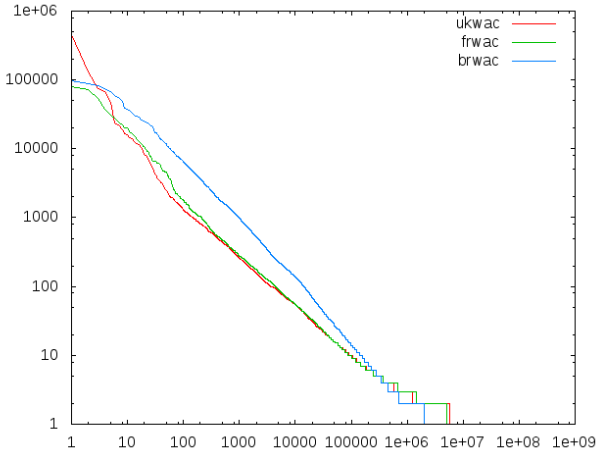
the average number of occurrences of each non-unique sentence in *brWaC* is slightly higher. Nonetheless, considering only large sentences (with more than 10 or 20 tokens each), which are an indicative of replicated content, average repetitions in *brWaC*, after the top thousand, are consistently lower.

Note that *ukWaC* and *frWaC* adopted the quadratic approach of comparison between pairs of documents, while *brWaC* adopts the linear approach of inspection of non-original long sentences in a document. We believe these results demonstrate that this alternative offers a better cost-benefit at this task.

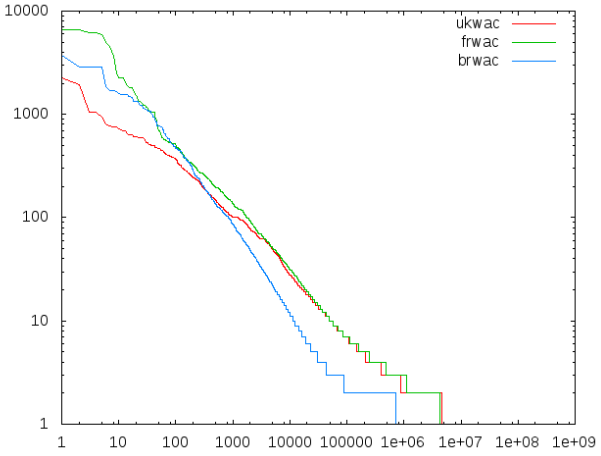
5. Conclusion

In this work, we presented the construction process of a large Web corpus, aiming to place Brazilian Portuguese on the same level as other major languages. Following the *WaCky* methodology, we constructed the *brWaC* corpus with 145 million sentences and 2.7 billion tokens. This new resource is freely available for both querying (through a NoSketch Engine interface) and downloading in CoNLL and Moses formats. Our hope is that a resource of this magnitude may form the basis and enable further studies targeting Brazilian Portuguese. We also provided an in-depth analysis of redundancy and corpus diversity, proposing the tolerance of 10% of non-original long sentences in a document as a good balance between originality of material and cost of collecting and processing documents.

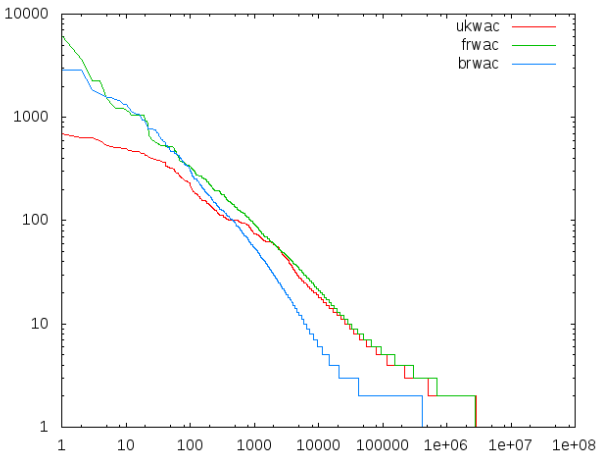
As future work, we highlight the need for more detailed analysis of the quality of this new corpus, including comparative studies with other reference corpora, obtained from different sources. We also plan on using it as the basis for creating and training vector space models.



(a) All sentences



(b) Sentences $>$ 10 tokens



(c) Sentences $>$ 20 tokens

Figure 3: Comparative log-scale sorted histograms of repeated sentences in *ukWaC*, *frWaC* and *brWaC*, when considering all sentences (top), sentences larger than 10 tokens (center) or larger than 20 tokens (bottom). The x axis represents all different sentences, and the y axis, the number of repetitions for each one, in descending order. In the first scenario, despite fewer in quantity, non-unique sentences in *brWaC* have a slightly higher average number of occurrences. For large sentences, though, average repetitions are also consistently lower after the top thousand.

		ukWaC	frWaC	brWaC
Sentences	Total	88.2	54.8	145.3
	>10 tokens	74.9	45.8	115.4
	>20 tokens	48.5	31.1	77.5
Unique sentences	Total	75.9	43.2	132.0
	>10 tokens	67.6	38.2	114.1
	>20 tokens	44.2	26.3	76.8
Repeated sentences	Total	5.6	5.1	2.0
	>10 tokens	4.6	4.2	0.7
	>20 tokens	2.8	2.8	0.4

Table 3: Comparative analysis on the incidence of unique and repeated sentences (in millions) across different Web corpora. Despite including considerably more sentences, our stricter approach to duplicated content resulted in smaller numbers of repeated sentences, especially of large size, in the *brWaC* corpus.

Acknowledgements

The authors acknowledge the partial funding of this research by CNPq-Brazil (grants n. 400715/2014-7, 312114/2015-0 and 423843/2016-8) and the Walloon Region (Project BEWARE n. 1610378).

6. References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Bernardini, S., Baroni, M., and Evert, S. (2006). A wacky introduction. *WaCky*, pages 9–40.

Bick, E. (2000). *The parsing system" Palavras": Automatic grammatical analysis of Portuguese in a constraint grammar framework*. Aarhus Universitetsforlag.

Boos, R., Prestes, K., Villavicencio, A., and Padró, M. (2014). brwac: a wacky corpus for brazilian portuguese. In *Computational Processing of the Portuguese Language*, pages 201–206. Springer.

Branco, A. and Costa, F. (2007). Self- or pre-tuning? deep linguistic processing of language variants. *Workshop on Deep Linguistic Processing, ACL2007 - International Conference of the Association of Computational Linguistics, Prague, Czech Republic, 24-29 June 2007, pp.57-64*.

Broder, A. Z. (1997). On the resemblance and containment of documents. In *Compression and Complexity of Sequences 1997. Proceedings*, pages 21–29. IEEE.

Di Marco, A. and Navigli, R. (2013). Clustering and diversifying web search results with graph-based word sense induction. *Computational Linguistics*, 39(3):709–754.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al. (2016). Google's multilingual neural machine translation system: Enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*.

Kilgarriff, A., Rychly, P., Smrz, P., and Tugwell, D. (2004). Itri-04-08 the sketch engine. *Information Technology*, 105:116.

Kilgarriff, A., Rundell, M., and Dhonnchadha, E. U. (2006). Efficient corpus development for lexicography: building the new corpus for ireland. *Language Resources and Evaluation*, 40(2):127–152.

Kilgarriff, A., Husák, M., McAdam, K., Rundell, M., and Rychlỳ, P. (2008). Gdex: Automatically finding good dictionary examples in a corpus. In *Proc. Euralex*.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Navigli, R. and Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods on Natural Language Processing (EMNLP)*, volume 14, pages 1532–1543.

Pomikálek, J. (2013). justext: Heuristic based boilerplate removal tool. *Available: Google code, online http://code. google. com/p/justext*.

Rodrigues, J., Branco, A., Neale, S., and Silva, J. (2016). Lx-dsemvectors: Distributional semantics models for portuguese. In *International Conference on Computational Processing of the Portuguese Language*, pages 259–270. Springer.

Wagner Filho, J., Wilkens, R., Zilio, L., Idiart, M., and Villavicencio, A. (2016). Crawling by readability level. In *Proceedings of 12th International Conference on the Computational Processing of Portuguese (PROPOR)*, pages 306–318.

Xiao, R. (2010). Corpus creation. In *Handbook of Natural Language Processing (2n Revised edition)*, pages 147–165. CRC Press - Taylor & Francis Group.

Ziai, R. and Ott, N. (2005). Web as corpus toolkit: User's and hacker's manual. *Lexical Computing Ltd., Brighton, UK*.

7. Language Resource References

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Clarke, C. L., Cormack, G. V., Laszlo, M., Lynam, T. R., and Terra, E. L. (2002). The impact of corpus size on question answering performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 369–370. ACM.

Ferraresi, A., Bernardini, S., Picci, G., and Baroni, M. (2010). Web corpora for bilingual lexicography: a pilot study of english/french collocation extraction and translation. *Using Corpora in Contrastive and Transla-*

tion Studies. Newcastle: Cambridge Scholars Publishing, pages 337–362.

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlỳ, P., and Suchomel, V. (2013). The tenten corpus family. In *7th International Corpus Linguistics Conference (CL)*, pages 125–127.

Liu, V. and Curran, J. R. (2006). Web text corpus for natural language processing. In *European Chapter of the Association for Computational Linguistics (EACL)*.

Santos, D., Simões, A., Frankenberg-Garcia, A., Pinto, A., Barreiro, A., Maia, B., Mota, C., Oliveira, D., Bick, E., Ranchhod, E., et al. (2004). Linguateca: um centro de recursos distribuído para o processamento computacional da língua portuguesa.

Wagner Filho, J. A., Wilkens, R., and Villavicencio, A. (2016). Automatic construction of large readability corpora. *Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 164–173.

Czech Text Document Corpus v 2.0

Pavel Král[1,2], **Ladislav Lenc**[1,2]

[1]Dept. of Computer Science & Engineering
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic

[2]New Technologies for the Information Society
Faculty of Applied Sciences
University of West Bohemia
Plzeň, Czech Republic

Abstract

This paper introduces "Czech Text Document Corpus v 2.0", a collection of text documents for automatic document classification in Czech language. It is composed of the text documents provided by the Czech News Agency and is freely available for research purposes at `http://ctdc.kiv.zcu.cz/`. This corpus was created in order to facilitate a straightforward comparison of the document classification approaches on Czech data. It is particularly dedicated to evaluation of multi-label document classification approaches, because one document is usually labelled with more than one label. Besides the information about the document classes, the corpus is also annotated at the morphological layer. This paper further shows the results of selected state-of-the-art methods on this corpus to offer the possibility of an easy comparison with these approaches.

Keywords: corpus, Czech, document classification, multi-label, text

1. Introduction

Automatic classification (or categorization) of text documents is very important for information organization and storage because of the significant increase of the amount of electronic documents and the rapid growth of the Internet. Many efficient approaches have been proposed. They are usually based on supervised machine learning. The documents are projected into the so-called vector space model, basically using the words as features for various classification algorithms. The approaches differ in the used methods, however the common point is that all of them need an annotated document corpus to train the parameters.

A sufficient number of the corpora in several languages, particularly in English, is freely available. However, to the best of our knowledge, the Czech one is missing.

The main goal of this paper consists in presenting a corpus of Czech text documents. It is composed of real newspaper articles provided by the Czech News Agency (ČTK)[1] and is available for research purposes for free. It is created for a straightforward comparison of the document classification approaches in Czech. One document is usually labelled with more than one label, therefore this corpus is usually used for evaluation of multi-label document classification. Besides the information about the document classes, the corpus is also morphologically annotated. The morphological annotation has been done fully automatically.

Another research contribution of this paper represents the reported results of selected state-of-the-art methods on this corpus to offer the possibility of an easy comparison with these approaches.

The paper structure is as follows. The following section presents other text corpora for document classification freely available for research purposes. Section 3. details our corpus. Section 4. presents the results of the selected state-of-the art methods on this dataset. The last section concludes the paper.

[1]http://www.ctk.eu/

2. Other Text Corpora

Some important existing text classification corpora in several languages are described below.

2.1. Reuters-21578

Reuters-21578[2] corpus is a collection of 21,578 documents. The training part is composed of 7769 documents, while 3019 documents are reserved for testing. The number of possible categories is 90 and the average label/document number is 1.23. This dataset is the most frequently used benchmark for English.

2.2. RCV1-V2

RCV1-V2[3] (Lewis et al., 2004) is another text classification test collection which is freely available for research purposes. It contains about 800,000 manually categorized newswire English stories from Reuters, Ltd. RCV1 contains English documents, while RCV2 is composed of text documents in French, German, Italian, Spanish and others. This dataset is also widely used as a benchmarking corpus for English and the languages mentioned above.

2.3. Other Corpora

For other corpora dedicated for text categorization, you can visit for instance `http://mulan.sourceforge.net/datasets-mlc.html`.

3. Corpus Description

3.1. General Information

The main part (for training and testing) of the Czech Text Document Corpus v 2.0 is composed of 11,955 real newspaper articles provided by the Czech News Agency. We provide also a development set which is intended to be used for tuning of the hyper-parameters of the created models. This set contains 2735 additional articles.

[2]http://www.daviddlewis.com/resources/testcollections/reuters21578/

[3]https://archive.ics.uci.edu/ml/datasets/Reuters+RCV1+RCV2+Multilingual,-

The documents belong to different categories (classes) such as weather, politics, sport, culture, etc. Each document is associated with one or more labels (classes). It is thus beneficial to use it for multi-label document classification scenarios. The multi-label classification task is considerably more important than the single-label classification because it usually corresponds better to the needs of the current applications.

The total category number is 60[4] out of which 37 most frequent ones are used for classification. The reason of this reduction is to keep only the classes with the sufficient number of occurrences to train the models. The corpus was annotated by professional journalists from the Czech News Agency. All documents are further automatically morphologically annotated using UDPipe tool.

3.1.1. Statistical Information

Table 1 shows the statistical information about the corpus[5]. It shows for instance that lemmatization decreases the vocabulary size from 150,899 to 82,986 which represents the reduction by 45%. Another interesting observation is the distribution of the POS tags in this corpus.

Figure 1 illustrates the distribution of the documents depending on the number of labels. It shows that the maximal number of categories associated with one document is eight, the majority of documents has two categories and the average label number is 2.55.

Figure 2 shows the distribution of the document lengths (in word tokens). The documents are relatively long and the longest documents are composed of more than 7000 word tokens. Another interesting information is that the most documents (about 2000) contain at most 50 words. The average document size is 277 words.

3.1.2. Download

This dataset is licensed under the Attribution-NonCommercial-ShareAlike 3.0 Unported License[6]. Therefore it is freely available for research purposes, however any commercial use is strictly excluded. This corpus is possible to download at `http://ctdc.kiv.zcu.cz/`.

Unit name	Number	Unit name	Number
Document	11,955	Word	3,505,965
Category	60	Unique word	150,899
Cat. classif.	37	Unique lemma	82,986
Noun	894,951	Punct	553,099
Adjective	369,172	Adposition	340,785
Verb	287,253	Numeral	265,430
Pronoun	258,988	Adverb	144,791
Coord. conj.	100,611	Determiner	84,681
Pronoun	74,340	Aux. verb	70,810
Subord. conj.	41,487	Particle	12,383
Symbol	2420	Interjection	142
Other	4126		

Table 1: Corpus statistical information

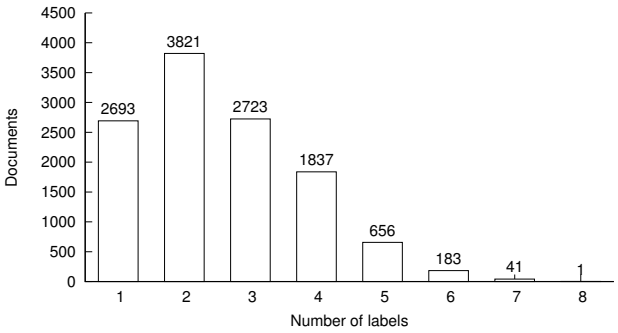


Figure 1: Distribution of documents depending on the number of labels

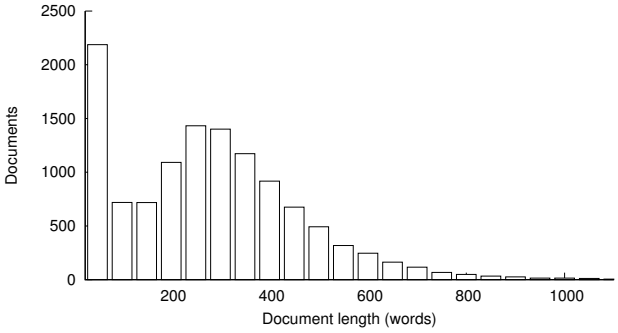


Figure 2: Distribution of the document lengths

3.2. Morphological Annotation

As already mentioned, we used UDPipe tool (Straka and Straková, 2017)[7] for automatic morphological analysis of the corpus. This tool provides en efficient pipeline for sentence segmentation, tokenization, POS tagging, lemmatization and dependency parsing. It also contains models for 50 languages of universal dependencies (UD) 2.0. This system can be further used with data in CoNLL-U format[8].

According to the authors, the accuracy of the lemmatizer and of the POS tagger are both about 98% on the UD version 2 of the Prague Dependency Treebank (PDT) 3.0[9]. The performance of the syntactic parsing is represented by labelled attachment score (LAS) which is about 83%. This system uses 17 part-of-speech categories drawn from the revised version of the Google universal POS tags.

3.3. Technical Details

Text documents are stored in the individual text files using UTF-8 encoding. Each filename is composed of the serial number and the list of the categories abbreviations separated by the underscore symbol and the *.txt* suffix. Serial numbers are composed of five digits and the numerical series starts from the value one.

For instance the file *00046_kul_nab_mag.txt* represents the document file number 46 annotated by the categories *kul* (culture), *nab* (religion) and *mag* (magazine selection). The content of the document, i.e. the word tokens, is stored in one line. The tokens are separated by the space symbols.

[4]This list is reported in Table 3 in Section 8..

[5]Development set is excluded from all analyses reported in this section.

[6]http://creativecommons.org/licenses/by-nc-sa/3.0/

[7]http://ufal.mff.cuni.cz/udpipe

[8]http://universaldependencies.org/format.html

[9]http://ufal.mff.cuni.cz/pdt3.0/

Every text document was further automatically mophologically analyzed. This analysis includes lemmatization, POS tagging and syntactic parsing. The fully annotated files are stored in *.conll* files. We also provide the lemmatized form, file with suffix *.lemma*, and appropriate POS-tags, see *.pos* files. The tokenized version of the documents is also available in *.tok* files.

3.4. Evaluation Protocol

All following experiments use the five-fold cross validation procedure, where 20% of the corpus is reserved for testing and the remaining part for training of the models. The development set was used to tune the hyper-parameters of the models.

For evaluation of the multi-label document classification, it is used the standard Precision (P), Recall (R) and F-measure (F_m) metrics (Powers, 2011). For evaluation of the single-label classification, the authors use the standard accuracy metric.

The confidence interval of the experimental results is 0.6% at a confidence level of 0.95 (Press et al., 1996).

4. Experiments

The two following sections present the results of the selected classification algorithms on this dataset. The first section deals with multi-label document classification while the second one describes the classification score of single-label classification task.

4.1. Multi-label Document Classification

The first reported approach (Hrala and Král, 2013a) uses Bag of Words (BoW) to create the features. Non-significant words are removed using Part of speech (POS) filtering and for feature selection, the mutual information method is used. In the original paper, the authors show the results of three classifiers, namely Naive Bayes, Maximum entropy (ME) and Support vector machine with three traditional multi-label classification approaches. Only the best classification accuracy obtained by ME classifier is reported in this paper.

The second method (Brychcín and Král, 2014) proposes novel unsupervised features using an unsupervised stemmer, latent Dirichlet allocation and semantic spaces (HAL and COALS). These features are integrated with word features to improve classification results. Multi-label classification scenario is realized using a set of binary classifiers. Maximum entropy model is used for classification.

Neural networks are very popular in natural language processing field today and they outperform many state-of-the-art approaches with only very simple preprocessing. The following approach (Lenc and Král, 2017) uses two different feed-forward neural networks, namely multi-layer perceptron (MLP) and convolutional neural networks (CNN) to achieve new state-of-the art results on this corpus. The authors use thresholding to realize multi-label document classification task.

The results of the above described approaches are illustrated in Table 2.

Method	P	R	F_m
ME (Hrala and Král, 2013a)	-	-	76.8
words+ME (Brychcín and Král, 2014)	88.1	72.7	79.7
unsup+ME (Brychcín and Král, 2014)	89.0	75.6	81.7
MLP (Lenc and Král, 2017)	83.7	83.6	83.9
CNN (Lenc and Král, 2017)	86.4	82.8	84.7

Table 2: Multi-label document classification results of the different approaches

4.2. Single-label Document Classification

The authors (Hrala and Král, 2013b) evaluate five feature selection methods and three classifiers on this corpus. Lemmatization and POS tagging are used for a precise representation of the Czech documents. It was demonstrated that POS-tag filtering is very important, while the lemmatization plays only a marginal role for classification. In this work, only the first document class was considered for classification and the authors consider it as the main document category. The best classification accuracy was obtained by SVM classifier and is 91.2%.

5. Conclusions

This paper introduced a novel collection of Czech text documents. This corpus is composed of real newspaper articles provided by the Czech News Agency and is available for research purposes for free. It was created to facilitate a straightforward comparison of the document classification approaches in Czech language.

This corpus is particularly intended to evaluate multi-label document classification approaches, because one document is usually associated with more than one label. Besides the information about the document classes, the corpus is automatically annotated at morphological layer. This paper further shows the results of the selected state-of-the-art algorithms on this corpus to offer the possibility of a straightforward comparison with the future research.

We plan to submit this corpus to be a part of the Language Research Infrastructure of LINDAT/CLARIN project[10].

6. Acknowledgements

This work has been supported by the project LO1506 of the Czech Ministry of Education, Youth and Sports.

7. Language Resource References

Brychcín, T. and Král, P. (2014). Novel unsupervised features for Czech multi-label document classifi cation. In *13th Mexican International Conference on Artificial Intelligence (MICAI 2014)*, pages 70–79, Tuxtla Gutierrez, Chiapas, Mexic, 16-22 November. Springer.

Hrala, M. and Král, P. (2013a). Multi-label document classification in Czech. In *16th International conference on Text, Speech and Dialogue (TSD 2013)*, pages 343–351, Pilsen, Czech Republic, 1-5 September. Springer.

[10]https://lindat.mff.cuni.cz/en

Hrala, M. and Král, P. (2013b). Evaluation of the Document Classification Approaches. In *8th International Conference on Computer Recognition Systems (CORES 2013)*, pages 877–885, Milkow, Poland, 27-29 May. Springer.

Lenc, L. and Král, P. (2017). Deep neural networks for Czech multi-label document classification. *arXiv preprint arXiv:1701.03849*.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. (2004). Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

Powers, D. (2011). Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1996). *Numerical recipes in C*, volume 2. Citeseer.

Straka, M. and Straková, J. (2017). Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August. Association for Computational Linguistics.

8. Appendix

Abbr.	Category in Czech	English translation
aut	Automobilový průmysl	Automobile industry
bos	Bohemika	Czech Rep. from abroad
bsk	Sklářský průmysl	Glass industry
bua	Burzy akciové	Stock exchanges
buk	Burzy komoditní	Commodity exchanges
bup	Burzy peněžní	Currency exchanges
bur	Burzy	Exchanges
cen	-	-
che	Chemický a farmaceutický průmysl	Chemical and pharmaceutical industry
den	Zpravodajské deníky	News schedules
dpr	Doprava	Transport
dre	Dřevozpracující průmysl	Woodworking industry
efm	Firmy	Companies
ekl	Životní prostředí	Environment
eko	Ekologie	Ecology
ene	Energie	Energy
eur	Evropská unie - zprávy	European union - news
fin	Finanční služby	Financial services
for	Parlamenty a vlády	Parliaments and governments
fot	Fotbal - zprávy	Soccer
hok	Hokej - zprávy	Ice hockey
hut	Hutnictví	Metallurgy
kat	Neštěstí a katastrofy	Accidents and disasters
kul	Kultura	Culture
mag	Magazínový výběr	Magazine selection
mak	Makroekonomika	Macroeconomics
med	Média a reklama	Media and advertising
met	Počasí	Weather
mix	Mix	Mix
mot	Motorismus	Motoring
nab	Náboženství	Religion
obo	Obchod	Trade
odb	Práce a odbory	Labour and Trade Unions
pit	Telekomunikace a IT	Telecommunications & IT
pla	Plány zpravodajství ČTK	Events news
pod	Politika ČR	Czech Republic Politics
pol	Politika	Politics
prg	Pragensie	Prague issues
prm	Lehký průmysl	Light industry
ptr	Potravinářství	Food industry
reg	Region	Region
sko	Školství	Educational system
slo	Slovenika	Slovakia from abroad
slz	Služby	Services
sop	Sociální problematika	Social problems
spc	-	-
spl	Životní styl	Life style
spo	Sportovní zpravodajství	Sports
sta	Stavebnictví a reality	Building industries and property
str	Strojírenství	Mechanical engineering
sur	Suroviny	Raw materials
tlk	Telekomunikace	Telecommunications
tok	Textil	Textile
tur	Cestovní ruch	Tourism
vat	Věda a technika	Science and technology
zah	Zahraniční	Foreign
zak	Kriminalita a právo	Criminality and law
zbr	Zbraně	Arms
zdr	Zdravotnictví	Health service
zem	Zemědělství	Agriculture

Table 3: List of the categories

Corpora of Typical Sentences

Lydia Müller, Uwe Quasthoff, Maciej Sumalvico
University of Leipzig
Augustusplatz 10, 04109 Leipzig
{lydia,quasthoff,sumalvico}@informatik.uni-leipzig.de

Abstract

Typical sentences of characteristic syntactic structures can be used for language understanding tasks like finding typical slotfiller for verbs. The paper describes the selection of such typical sentences representing usually about 5% of the original corpus. The sentences are selected by the frequency of the corresponding POS tag sequence together with an entropy theshold, and the selection method is shown to work language independently. Entropy measuring the distribution of words in a given position turns out to identify larger sets of near-duplicate sentences, not considered typical. A statistical comparison of those subcorpora with the underlying corpus shows the intended shorter sentence length, but also a decrease of word frequencies for function words associated to more complex sentences.

Keywords: typical sentences, sentence signatures, language statistics, corpus comparison

1. Introduction

Statistical analyses of language are usually based on large corpora compiled from publicly available written sources, e.g. news, Wikipedia, crawled webpages or literature (Baroni and Bernardini 2004, Biemann et al. 2013). Compared to everyday speech, such sources tend to be biased towards long sentences and complex syntactic structures. There have been attempts to compile corpora more represantative of everyday language by utilizing different sources, especially movie subtitles (Lison and Tiedemann 2016). However, the availability of such sources is limited.

In this paper, we propose a method for diminishing the bias towards complex syntactic structures usually found in large corpora. In order to accomplish this, we select "typical sentences", defined as sentences with a common syntactic structure (represented as a sequence of POS-tags). In contrast to simplified or controlled languages (like Simplified English (Ogden 1932) or Kontrolliertes Deutsch (Lehrndorfer 1996) (controlled German) there is no set of handwritten rules for syntax and vocabulary. However, the sublanguage emerging in typical sentences may be considered as an automatically genarated analogy.

As an illustration of our approach, we currently provide typical sentence corpora for English, German, French, Dutch and Italian as a part of the Leipzig Corpora Collection.[1] Random samples of up to 1 million sentences are freely available for download.

2. Selecting typical sentences

Given a POS-tagged sentence, we define the *sentence signature* as the corresponding sequence of POS tags. For a corpus of sentences, such signatures can be ordered by frequency. As one could expect, the most frequent sentence signatures belong to relatively short sentences with typical structure. Table 1 shows the top five sentence signatures from newspaper corpora of the Leipzig Corpora Collection (Goldhahn et al. 2012) with sample sentences for English. As POS tagger, the Stuttgart Tree Tagger (Schmid 1994) is used.

The following two properties of the list of sentence signatures are not so obvious: There are frequent sentence signatures belonging to large sets of near-duplicate sentences. They are usually unexpectedly long and contain numbers. The most frequent examples are given in Table 2. Such sentences are not considered typical and should be identified and removed. A simple way to identify such near-duplicates is to exploit the small variation in the vocabulary for these sets of sentences: For each sentence signature (with minimum frequency 5), the normed entropy[2] of the vocabulary distribution in each position of the sentence is calculated. Figure 1 shows the distribution of the median normed entropy for sentence signatures for German. Distributions for other languages are similar. Sentence signatures with low median normed entropy (≤ 0.5) are removed.

The choice of the median rather than arithmetic mean is motivated as follows: it is acceptable and expected, that some positions in the sentence show little or no variation in the vocabulary (e.g. articles, auxiliary verbs, even the main verb). As long as enough other positions show considerable variation (e.g. subjects and objects), the signature is a good candidate for a source of typical sentences. The median enables us to separate the signatures with variation at too few positions more clearly.

Another interesting observation is that there are unexpectedly many different sentence signatures: about 95.6% of all sentences signatures appear only once. Table 3 shows the sentence coverage for the N most frequent sentence signatures after the removal of the near-duplicates.

After removing signatures corresponding to near-duplicate sentences, we select $100,000$ most frequent signatures to form the typical sentence corpora. Although the choice of this theshold is quite arbitrary, it leads to a good tradeoff between the corpus size and the simplicity of sentences by selecting typically between 5 and 10 percent sentences (see Table 4 for details).

3. Statistical properties

In order to compare the subcorpus of typical sentences to the original corpus, we consider the following properties:

[1] `http://corpora.uni-leipzig.de`

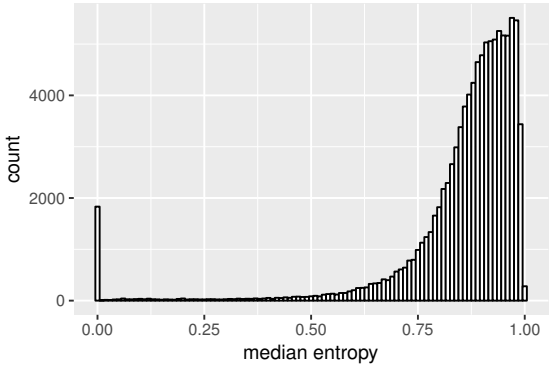
[2] The entropy is divided by the maximum entropy possible at a given position, so that the result is a number between 0 and 1.

Rank	Signature	Sample sentence
1	DT NN VBZ JJ SENT	The future is mobile.
2	PP VBD RB JJ SENT	It was just crazy!
3	DT NN VBZ RB JJ SENT	The news is all good.
4	PP VVD DT NN SENT	They licensed the technology.
5	PP VVP DT NN SENT	They love the area.

Table 1: Top 5 signatures for English with example sentences.

Signature	Sample sentence
NP NP VVD TO DT NN SENT	Aaron Blake contributed to this report.
NP NP VVD IN CD NN IN NP , NP CD , CD SENT	Alma Santos posted at 7:30 am on Sun, Oct 11, 2015.
DT NN NN IN DT NN VBZ (CD) CD SENT	The phone number at the clinic is (320) 395-2527.

Table 2: Sample signatures corresponding to near-duplicate sentences in English.



O

Figure 1: Normed entropy distribution for German sentence signatures: For each signature, the median of position-wise normed entropies is calculated.

N	% sentences
10,000	4.1 %
50,000	6.8 %
100,000	8.1 %
500,000	11.7 %
1,000,000	13.4 %
5,000,000	18.5 %
10,000,000	22.1 %

Table 3: Sentence coverage of the N most frequent signatures in the German corpus (total=211, 657, 876).

- number of sentences (see Table 4)
- distribution of sentence length (see Figure 2)
- Zipf distribution for word frequencies, frequency@200,000
- Changes in the stopword ranking
- Number of significant word co-occurrences

The described selection of sentences leads to a strong reduction of sentences in the corpora with typical sentences. For example, the German corpus with typical sentences

Table 4: Number of sentences

Language	sentences	typical sentences	percent
German	259,026,023	20,103,234	7.76
English	156,934,303	4,193,396	2.67
French	74,823,426	3,758,924	5.02
Dutch	70,332,253	5,432,161	7.72
Italian	44,636,533	2,023,640	4.53

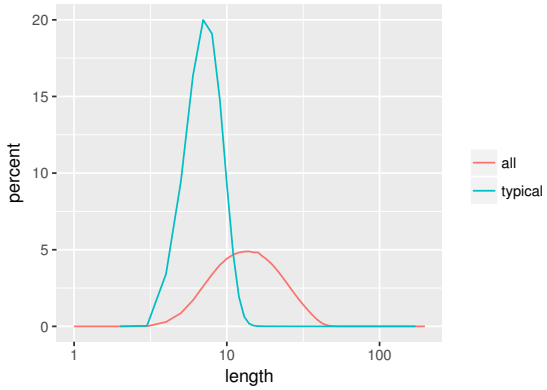


Figure 2: Sentence Length Distribution: Sentence length is measured as number of tokens. Red and turquoise line show the distribution for all sentences and only the typical sentences of German, respectively.

contains only 7.76% of the all German sentences. However, this are still more than 20 million sentences (see Table 4). Thus, the corpora are still large enough for statistical analyses.

We measure the sentence length as the number of tokens in a sentence. Typical German sentences are normally about 7 tokens in length, while the peak in the distribution for all German sentences is 14 tokens (see Figure 2).

Zipf's Law for the German corpora is shown in Figure 3. The selection of only sentences with frequent signatures does not affect the characteristics of the distribution. Due to the smaller corpus size of the typical sentences, the distribution is shifted to lower frequencies.

The simpler structure of shorter sentences results in

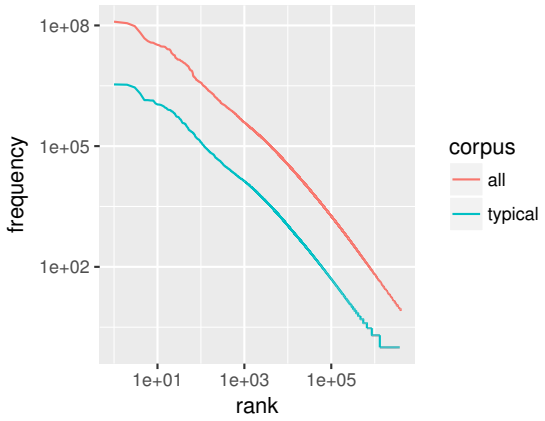


Figure 3: Zipf's Law for typical and all sentences of German.

word	type	rank	rank change
Die	article	3	–12
Das	article	6	–33
Der	article	7	–20
werden	auxiliary	13	–11
Sie	pronoun	14	–23
und	conjunction	15	+12
von	preposition	17	+11
sind	auxiliary	22	–11
wurde	auxiliary	26	–28
im	preposition	28	+15
war	auxiliary	30	–19
zu	preposition	31	+24
Es	pronoun	33	–33
Ein	article	36	–54
bei	preposition	39	+11
Er	pronoun	43	–54
In	preposition	45	–16
Eine	article	46	–83
am	preposition	47	+15
nach	preposition	49	+18
Wir	pronoun	50	–39

Table 5: Rank changes among the top-100 German words.

changes in the stopword ranking (Table 5 and 6). As could be expected, more sentences begin with simple nominal phrases and so the ranks of capitalized articles and pronouns decrease. Also the ranks of copula/auxiliary verbs decrease, as they play a key role in typical sentences. More interesting is an increase for conjunctions, especially "and", as well as prepositions. Conjunctions automatically generate longer sentences and are variable in its syntactic position. Also prepositional phrases often give rise to complex syntactic structures. These features are less frequent in typical sentences.

The number of significant co-occurrences depends on the number of occurrences of a word and is therefore not comparable across corpora of different sizes or across large sets of words of different frequencies. In Figure 4, we use the neighbors of a word as input for the calculation of sig-

word	type	rank	rank change
is	auxiliary	3	–5
The	article	4	–7
was	auxiliary	5	–8
I	pronoun	9	–9
We	pronoun	12	–48
He	pronoun	13	–46
and	conjunction	14	+11
It	pronoun	16	–46
They	pronoun	20	–98
at	preposition	23	+9
can	modal	37	–11
from	preposition	38	+16
A	article	42	–40

Table 6: Rank changes among the top-100 English words.

nificant co-occurrences. Thus, a word occurring n times co-occurs at most $2n$ times with one of its significant co-occurrences (i.e. a significant co-occurrence on the left and the right side). Therefore, we normalize the frequency of the significant co-occurrences for each word by $2n$. The distributions of the resulting ratios is shown Figure 4. The quality of word co-occurrences also changes: Due to the decrease of conjunctions, the number of pairs of similar terms decreases. Hence, there are fewer similar terms in the sentence-based word co-occurrences.

It is notable that after selecting for typical sentences, words more often appear with significant co-occurrences than before. This indicates a lower variety in the combination of words when using only typical sentences. Interestingly, there is a peak in the distribution for typical sentences at 0.5. These are words that always co-occur with a significant co-occurrence.

Figure 4 shows two distributions for each corpus of German. Hereby, the solid lines take only frequent words (frequency ≥ 100) into account, while the dashed lines show the distribution for all words. Since the corpus with typical sentences is much smaller, the word frequencies are lower and more noisy. This leads to artificial peaks in the distribution. However, the general trend is the same no matter if we use all words or only frequent words.

4. Application: Clustering of sentences and typical constituents

For high and medium frequency verbs, we find many typical sentences with the same verb in the same position. By means of clustering those sentences using word similarity, we can identify possible replacements for words in each position. For the German verb *brannte* (engl. *burned*) we find the following typical sentence of length five: *Die Scheune brannte völlig nieder. (The barn burned down completely.)* Table 7 shows the replacement words in similar sentences, while Table 8 shows an equivalent example from the English corpus. In the German example, it turns out that both buildings (or its parts) and vehicles are mentioned to burn completely. So, in position two the burning objects built two clusters, in positions four and five we find near-synonyms for formulating the sentences.

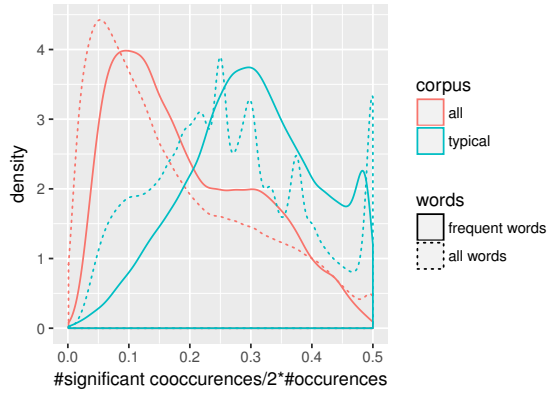


Figure 4: Co-occurrences in typical and all sentences for German. We calculate the ratio of the observed significant co-occurrences and the theoretically possible co-occurrences (i.e. 2 times the occurrence of a word). The figure shows the density distribution of these ratios for the German corpus of all sentences and the German corpus of typical sentences. Solid lines represent the distribution taking only frequent words (occurrences ≥ 100) into account, while dashed lines correspond to the distribution for all words.

Table 7: Sentence variants for German

Pos.	Sentence	Variants
1	Die	Der, Ein
2	Scheune	Gebäude, Haus, Halle, Dachgeschoss, Auto, Fahrzeug, Wagen ...
3	brannte	
4	völlig	vollständig, vollkommen, komplett, total
5	nieder	ab, aus

Table 8: Sentence variants for English

Pos.	Sentence	Variants
1	The	
2	whole	–
3	building	schoolhouse, village, house
4	was	had, –
5	completely	all, –
6	burned	
7	down	out

5. Language Resource References

In the Leipzig Corpora Collection, POS tagging is applied for about 35 languages with the following 21 languages having more than 10 million of sentences: Arabic, Danish, Dutch, English, Esperanto, French, German, Hungarian, Icelandic, Italian, Norwegian, Polish, Portugese, Romanian, Russian, Slovak, Slovene, Spanish, Swedish, Ukrainian and Vietnamese. Here, mainly the Stuttgart Tree Tagger (Schmid 1994) is used.

For these languages, the corpora of typical sentences will subsequently be produced and made available for download at *http://wortschatz.uni-leipzig.de/en/download*.

At the time of writing this paper, the corpora for English, French, German, Dutch and Italian are already available. The corpora can be recognized on the download website by the '`-typical`' addition in their names.

References

Marco Baroni and Silvia Bernardini. Bootcat: Bootstrapping corpora and terms from the web. In *Proceedings of LREC 2004*, pages 1313–1316, 2004.

Chris Biemann, Felix Bildhauer, Stefan Evert, Dirk Goldhahn, Uwe Quasthoff, Roland Schäfer, Johannes Simon, Leonard Swiezinski, and Torsten Zesch. Scalable construction of high-quality web corpora. *JLCL*, 28(2):23–59, Dec 2013.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, 2012.

A. Lehrndorfer. *Kontrolliertes Deutsch: linguistische und sprachpsychologische Leitlinien für eine (maschinell) kontrollierte Sprache in der technischen Dokumentation*. Tübinger Beiträge zur Linguistik. G. Narr, 1996.

Pierre Lison and Jörg Tiedemann. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 923–929, 2016.

C.K. Ogden. *Basic English: a general introduction with rules and grammar*. Psyche miniatures: General series. K. Paul, Trench, Trubner & Co., Ltd., 1932.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

The German Reference Corpus DeReKo: New Developments – New Opportunities

Marc Kupietz*, Harald Lüngen*, Paweł Kamocki*†, Andreas Witt*
*Institut für Deutsche Sprache,
R5, 6-13; D-68161 Mannheim
†European Language Resources Distribution Agency
9 rue des Cordelières; F-75013 Paris
{kupietz|luengen|witt}@ids-mannheim.de, kamocki@elda.org

Abstract

This paper discusses current trends in DeReKo, the German Reference Corpus, concerning legal issues around the recent German copyright reform with positive implications for corpus building and corpus linguistics in general, recent corpus extensions in the genres of popular magazines, journals, historical texts, and web-based football reports. Besides, DeReKo is finally accessible via the new corpus research platform KorAP, offering registered users several news features in comparison with its predecessor COSMAS II.

Keywords: reference corpus, very large corpus, diversity, intellectual property rights, copyright reform, corpus analysis, comparable corpora, collocation analysis, word embeddings

1. Introduction

The German Reference Corpus DeReKo is presumably the largest archive of German language texts designed for linguistic research (Kupietz et al., 2010). As of 2018 (Institut für Deutsche Sprache, 2018), it contains more than 42 billion tokens, comprising a multitude of genres such as newspaper text, fiction, or specialised text, with a current growth rate of 3.1 billion word per year. Besides the constant acquisition of new newspaper sources, one focus of corpus extension in the past four years has been on the curation of content from sources of computer-mediated communication (CMC), cf. Margaretha and Lüngen (2014); Schröck and Lüngen (2015); Lüngen et al. (2016).

In 2017, several other new genres that had previously not been available in DeReKo, have been acquired and included in the latest release (Institut für Deutsche Sprache, 2018). These are discussed in Section 3. Likewise as of 2017, the bulk of DeReKo can be accessed online via the new corpus research platform KorAP, which offers several new corpus query features, which are discussed in Section 4. Actual and envisioned improvements in DeReKo's legal status are discussed in Section 2.

2. Legal and Licensing Situation

2.1. License Types

The Institut für Deutsche Sprache (IDS), DeReKo's host institution, is not the owner of DeReKo's content. Rather, we have more than 200 license agreements with rights holders, mostly publishers, granting the use of texts for non-commercial, scientific research by registered users and strictly within the query-and-analysis-only framework (cf. Kupietz/Lüngen, 2014) offered through the corpus research interfaces of COSMAS II and KorAP (see Section 4.).

Due to the fact that the range of lexicographically edited dictionaries and lexicons on the German language offered by commercial publishers has deteriorated considerably in the last few years, IDS is interested in acquiring additional licensing that would allow for building, editing, and (semi-)commercially publishing fundamentals for comprehensive print and on-line dictionaries based on statistical and lexicographic analyses and evaluations of (parts of) DeReKo, in the future. Such a license extension towards commercial use could also open new application fields e. g. for DeReKo-based language models in commercial contexts and also in academic contexts whenever copies of DeReKo(-parts) are required which are currently prohibited by its license terms. Thus, this way the re-usability and productivity of DeReKo could be increased considerably (presented in Kupietz and Belica, 2013), the efforts required for such license renegotiations and their outcome are, however, not yet fully clear.

2.2. 2017 Copyright Reform

To address the new possibilities of distributing and using copyright-protected content that have emerged due to the development of digitisation and the web in the past 20 years, a reform of parts of the German *Urheberrechtsgesetz* (Copyright Act) has been passed by the German legislator in July 2017. The new "Act on Adapting Copyright to Current Requirements of the Knowledge Society" (UrhWissG, 2017) will enter into force on 1 March 2018.[1] It introduces new exceptions *(Schranke)* to exclusive rights, regulating how copyright-protected content may be used in the spheres of education and research, and within so-called knowledge institutions. From the perspective of DeReKo and the IDS corpus extension project, the paragraph § 60d on text and data mining in which the term "corpus" occurs several times, is most interesting. It states that available content may, without explicit permission from the copyright holders and for strictly non-commercial research purposes, be automatically reproduced, structured, and categorised for building a corpus which can then be exploited by and shared with a group of users for common research. This TDM exception

[1]The reform echoes the *Proposal for a Directive of the European Parliament and of the Council on copyright in the Digital Single Market* of 14 September 2016 (European Comission, 2016).

cannot be overridden by separate agreements. Upon completion of the research activities, however, the corpus and its copies must be deleted or handed over to an archive, library, or other educational institution. In the explanatory statement of the ministerial draft, it is pointed out that the new law does not imply a right to access copyright-protected material, but that it holds for cases where access is already given e. g. through a library or via the internet (Referentenentwurf, 2017). The said uses require equitable compensation to be paid to the collecting society VG WORT, which means that agreements will need to be negotiated.

While previously the prevailing legal opinion on the redistribution of web content that constitutes a copyright-protected work (*das Werk*) outside small projects was that an authorisation would have to be obtained form every single author (Beißwenger et al., 2017), the new legal situation will enable us to legally build linguistic corpora by scraping German language content from the web (be it original web genres, CMC, or any kind of documents that are offered openly via the web), and to process, and republish them in DeReKo for analysis by our registered, scientific user community.

3. Quantitative and Qualitative Extensions

The most recent DeReKo release (Institut für Deutsche Sprache, 2018) contains over 42 billion tokens. From most sources we continually get new material, and DeReKo's annual growth is currently at 3.1 billion tokens, see Fig. 1. After a focus on corpora of computer-mediated communication in the last few years (cf. Lüngen, 2017), recent additions to DeReKo comprise a press archive consisting of popular magazines, journals, and daily newspapers, as well as a Football Linguistics Corpus.

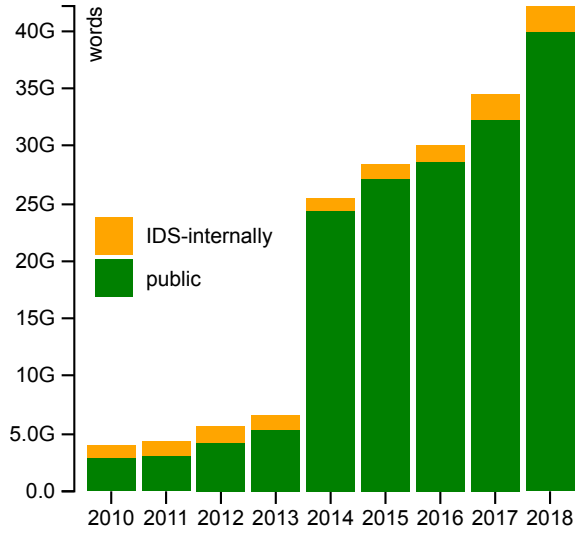


Figure 1: DeReKo-growth from 4 billion words in 2010 to 42 billion words in 2018.

As the bulk of DeReKo has always consisted of text data from daily newspapers, the new acquisitions have improved the genre dispersion in DeReKo in general. In the following, we give a brief description of the new corpora.

3.1. Press including journals and popular magazines

In an extended cooperation with a commercial news database provider, DeReKo has acquired new licenses for popular magazines. Most of them are weeklies such as *Stern, Brigitte,* or *Gala*, with the editions starting between 2007 and 2015, the earliest already in 1996. 23 of them, i.e. those that contained most data, have been included in the current release (Institut für Deutsche Sprache, 2018). They comprise altogether 90,272,352 tokens.

magazine	start	tokens
Stern	1996	50,167,964
Brigitte	2009	7,643,387
Hörzu	2007	5,466,757
GEO	2009	3,461,172
Gala	2015	2,467,552
Essen und Trinken	2015	1,814,979

Table 1: Examples of popular magazines newly included in DeReKo

Up to now, almost no popular magazines were available in DeReKo, as previously, unlike newspaper publishers, magazine publishers would frequently not hold digital exploitation rights for their content, such that agreements would have to be concluded with every author of an article in such a publication.

By the same cooperation, DeReKo has also acquired licenses for several newspapers previously not contained, mostly regional papers, but also e.g. the nation-wide *Die Welt*, as well as *Dolomiten*, the German-language daily with the highest circulation in Alto Adige, Italy. Most of the new titles start in the archive around the year 2000, the earliest one in 1992. Several papers from the north of the German-speaking area are also contained, such as *Neue Osnabrücker Zeitung* or *Ostseezeitung*, thus closing a certain gap (cf. Kupietz/Lüngen, 2014) in the regional dispersion of DeReKo. 53 new newspapers comprising altogether 6,248,900,215 tokens are newly included in the release (Institut für Deutsche Sprache, 2018). More (regional and local) newspapers from the same archive will be included in the releases to come.

daily	start	tokens
Kölner Stadt-Anzeiger	2000	551,722,542
Die Welt	1999	434,170,447
Hamburger Abendblatt	1999	346,807,799
Dolomiten (Italy)	2000	175,535,474
Neue Osnabrücker Zeitung	2012	94,276,568
Nordwest-Zeitung	2016	68,421,655
Ostsee-Zeitung	2016	37,221,069

Table 2: Examples of dailies newly included in DeReKo

Finally, the acquired archive contained a set of trade and technical journals, all starting in 2017. They include titles such as *allgemeine fleischer zeitung, Deutsches Ärzteblatt,* or *Technische Textilien*. 53 of them comprising altogether 1,183,122 tokens, are included in the latest release. Though

representing a smaller portion than the other extensions, they also improve the dispersion of genres in DeReKo.

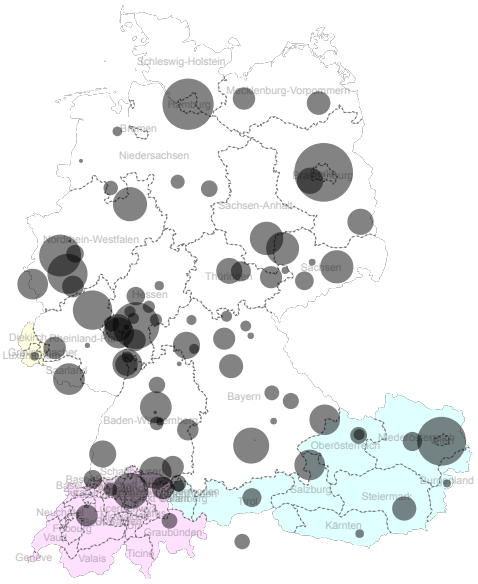


Figure 2: Geographical coverage with DeReKo press sources. The size of a circle corresponds to number of tokens.

The quantitative regional distribution of all (old and new) press sources in (Institut für Deutsche Sprache, 2018) is visualised in 2. To better judge the differences or similarities amongst the new and selected old corpora in DeReKo, we calculated the NMDS-projected distance map based on frequency lists shown in Fig. 3. In the map, DeReKo as a whole is shown in light blue font and as expected, the corpora of nationwide papers with the greatest extension in the archive (s=Der Spiegel; u=Süddeutsche; z=Die Zeit) lie close to it. Amongst the popular magazines, there is a main group shown in orange, with a cluster of yellow press/women's/family magazines (dgb, dak, gal, nwt, fis, edf, brg, ndo, elt), and a small cluster of interior design magazines (scw, cou). Ppm and art are a pop science and an art magazine, which for some reason get situated close to DeReKo as a whole, too. The blue group is clustered so closely, but its members lie next to each other and all represent food/cooking magazines (bee, eut, chk). The group shown in gold contains TV magazines (hrz, tvd, gng) and clusters nicely. Outliers different colours are loz (a corpus of fiction), wpd (wikipedia articles), and wdd (wikipedia talk pages). Ph, shown in green, is a monthly psychological magazine.[2]

3.2. Football Linguistics Corpora

Simon Meier of TU Berlin has kindly made available for DeReKo his *Football Linguistics Corpora* (Meier, 2017) (Meier, 2017), consisting of liveticker protocols and match reports from two different web publishers, from who we subsequently could also acquire the appropriate licenses.

[2]More detailed, interactive visualizations can be found on DeReKo's archive page under http://www.ids-mannheim.de/kl/projekte/korpora/archiv.html.

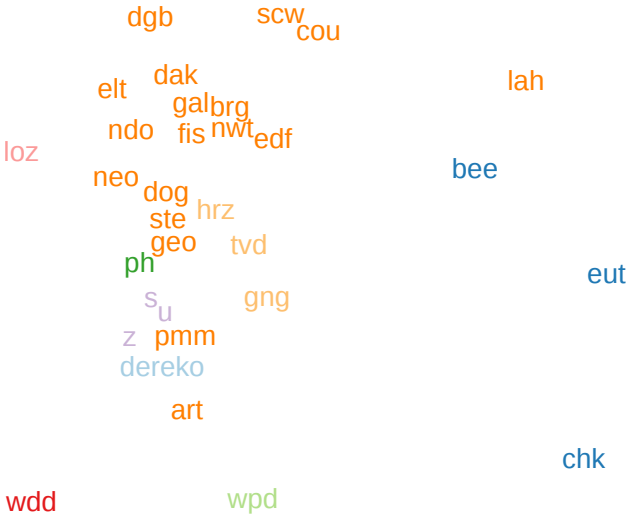


Figure 3: NMDS projection of the distances between different DeReKo sources. Based on Kilgarriff's (2001) *comparing corpora* approach (see Kupietz and Lüngen, 2014, for a detailed description of the procedure).

These reports and protocols cover all matches of the national football leagues First and Second Bundesliga, the European Champions and Europa Leagues as well as the Euro Cups, starting in 2006. With altogether 8.85 million tokens, they are part of the current release (Institut für Deutsche Sprache, 2018) and further strengthen the web genres and the sports domain within DeReKo.

3.3. Textgrid Digital Library

In 2017, we have also prepared an I5 version of the freely available TextGrid Digital Library which had previously been derived from the zeno.org Online Library.[3] It contains more than 2500 fictional and non-fictional texts (from domains such as fiction, fairy tales, (cultural) history, art, music, science, and philosophy) from the beginnings of printing until the first decades of the 20th century, amongst them nearly all German canonical literary texts for which copyright protection has expired (TextGrid, 2016). The TextGrid Digital Library corpus in its I5 incarnation contains almost 170 million tokens.

DeReKo is a corpus of contemporary German and has up to now contained only a few historical corpora, which are considered as "milestones" of the development of the German language i.e. still having influence on and relevance for the current shape of German, such the Goethe corpus or the Grimm corpus. We consider the TextGrid Digital Library relevant in that sense as well.

During the conversion to I5, it turned out that for up to 500 texts or so, only their 20th century publication date was available in a metadata field while the time of their actual creation was hidden in a bibliographic information string or not contained at all in the TEI source. Since I5 provides a dedicated metadata field for the date or period of creation and its specification seems essential for working with DeReKo (e.g. for any analysis of variation in time), we

[3]`http://www.zeno.org`

are currently running a post-processing campaign in which the creation dates are being derived from bibliographical information strings or, if not available, from external sources. For this reason, it is not contained in the release (Institut für Deutsche Sprache, 2018).

4. Querying and analyzing DeReKo

4.1. Now publicly accessible via KorAP

The corpus analysis platform KorAP (Bański et al., 2012; Diewald and Margaretha, 2016) that has been developed at the IDS as a successor to COSMAS II (Bodmer, 1996, 2005) since 2012 has been publicly available for querying DeReKo since May 2017.[4] Compared with COSMAS II, some essential features such as collocation analysis and sorting and aggregation of query hits are still under development. On the other hand, KorAP already offers some unique features for the analysis of DeReKo. One is that KorAP can efficiently query the complete DeReKo-collection with more than 40 billion words in one archive. Even though COSMAS II has no principle token number limitation related to 32 bit integers like many other query engines of its generation (COSMAS II was already designed in 1994), it is in effect limited by memory constraints to currently not much more than 7 billion words with one annotation layer per archive.

4.2. Multiple query languages

One more feature that distinguishes KorAP not only from COSMAS II, but probably from all currently existing corpus query systems is the support for multiple corpus query languages. This allows users coming from different research communities or traditions to readily use KorAP using the language they are accustomed to and enables the user to benefit from combining the advantages of different query languages. The supported query languages currently include the ANNIS Query Language (AQL; Rosenfeld 2010), Cosmas II (al Wadi, 1994), and Poliqarp (a CQP variant / extension; Przepiórkowski et al. 2010).

4.3. Multiple annotation layers

The support of an in principle unlimited number of linguistic annotation layers was one of the main reasons for the development of KorAP. The goal with regard to DeReKo is to add all annotations that are of interest to users or to tool providers and to make them queryable via KorAP. So far, the spectrum of queryable annotation layers was extended by dependencies and constituents (co-references, RST relations, etc. will follow). In addition, the set of different, competing annotations within one level was extended in order to improve the handling of erroneous and uncertain classifications. The use of linguistic annotations in search queries is often indispensable for the investigation of linguistic phenomena at an abstract level. If, however, they are needed for quantitative investigations, corpus-based hypothesis testing, and even for exploring phenomena beyond the finding of examples, their use is not trivial, and some possible pitfalls have to be considered as they do not have the status of observations, but rather the status of interpretations (Belica et al., 2011). This fact is also relevant for part-of-speech annotations for which an accuracy of up to 97-98% are reported (Giesbrecht and Evert, 2009). One problem even at such low error rates is that errors are not distributed uniformly and with an unfavourable tendency (Belica et al., 2011, p. 466). For typical linguistic questions, lower accuracy values are to be expected since such questions are typically related to less clear or common phenomena and the accuracy of the results for a handful of particular search queries can hardly be derived from the reported average accuracy of the annotation tool. Rather, it must be taken into account that although most constructions are almost always correctly annotated, others are almost always wrong, so that in the case of search requests to the latter, a very low recall must be expected. Since false-negative hits are not visible and thus cannot be easily identified, such a situation can easily lead to misinterpretations.

In order to improve the manageability of annotations, KorAP offers the possibility to use and query arbitrarily many competing annotations. Depending on the task, by using disjunctive queries, recall can be maximized (and false negatives minimized), and by using conjunctive queries, precision can be maximized (and false positives minimized). As shown in table 3 (similar example query in figure 4), the results obtained this way can differ significantly from one another, yielding almost 2 million more results for the disjunctive query than for the conjunctive one.

Furthermore, depending on the phenomenon to be investigated and the language domain to be examined, the annotations which promise the best results can be used in a targeted manner. Ideally, the annotations are based not only on different algorithms, but also on different training data.

4.4. DeReKo-based distributional analysis

Having a long history in providing and using distributional models based on collocation analysis for paradigmatic and syntagmatic analysis (Keibel and Belica, 2007)[5], we now investigate the pros and cons of these models in comparison to models based on word embeddings with respect to different linguistic applications. For this purpose we have developed a pipeline for training structured skip-gram networks (Ling et al., 2015) and building a collocation frequency database from DeReKo-releases, as well as a web interface to explore and compare them for syntagmatic (see figure 5 and paradigmatic relations. The interface will be made availably shortly. For the publication of the underlying models, however, we first need to investigate the possible implications on additional license fees and compensations to the VG WORT (see section 2.2.).

4.5. Using DeReKo in contrastive studies

In order to make DeReKo (re-)usable also in contrastive and cross-linguistic studies DeReKo takes part in two initiatives

[4] KorAP can be accessed via: `http://korap.ids-mannheim.de/kalamar/`. The source code is published openly under the BSD-2 license under `http://github.com/KorAP`. For pull requests, please consider using KorAP's Gerrit Code Review `http://korap.ids-mannheim.de/gerrit`.

[5] Freely available without registration via our open lab under http://corpora.ids-mannheim.de/ccdb

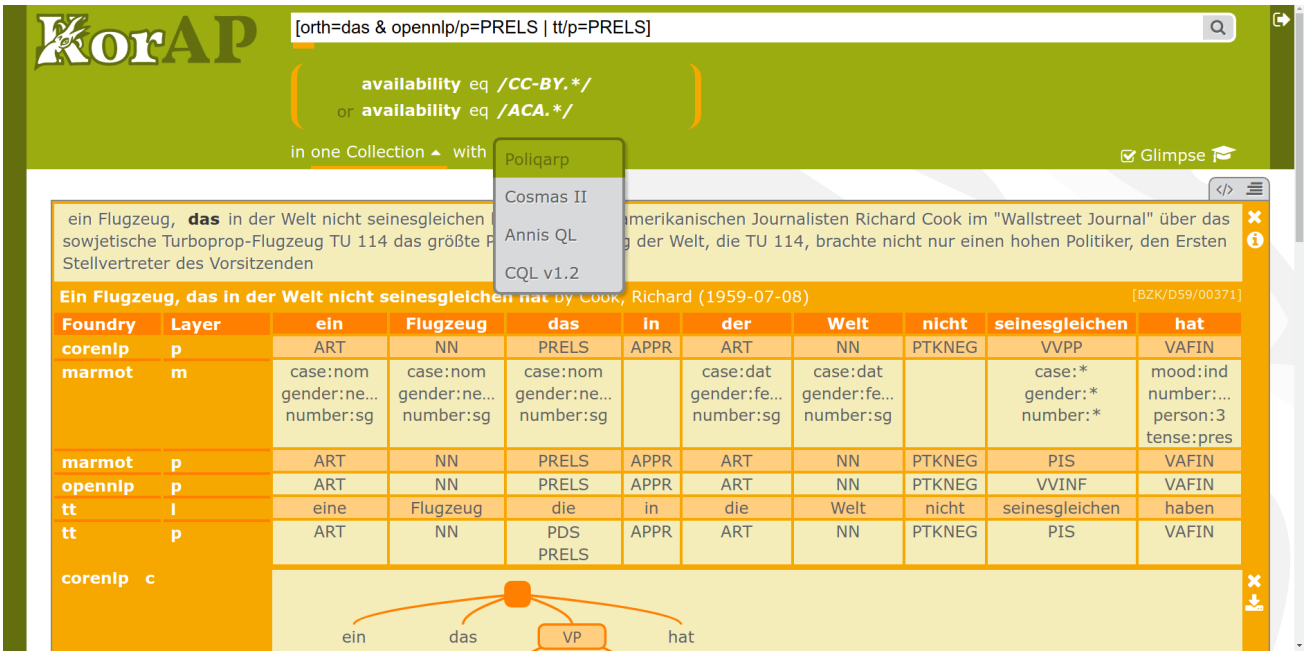


Figure 4: KorAP example query in a virtual collection using the Poliqarp QL, showing the currently available annotation layers for one hit (POS: CoreNLP, MarMoT, OpenNLP, TreeTagger; lemma: TreeTagger; morphology: MarMoT; constituency: CoreNLP; dependency: Malt (not visible)).

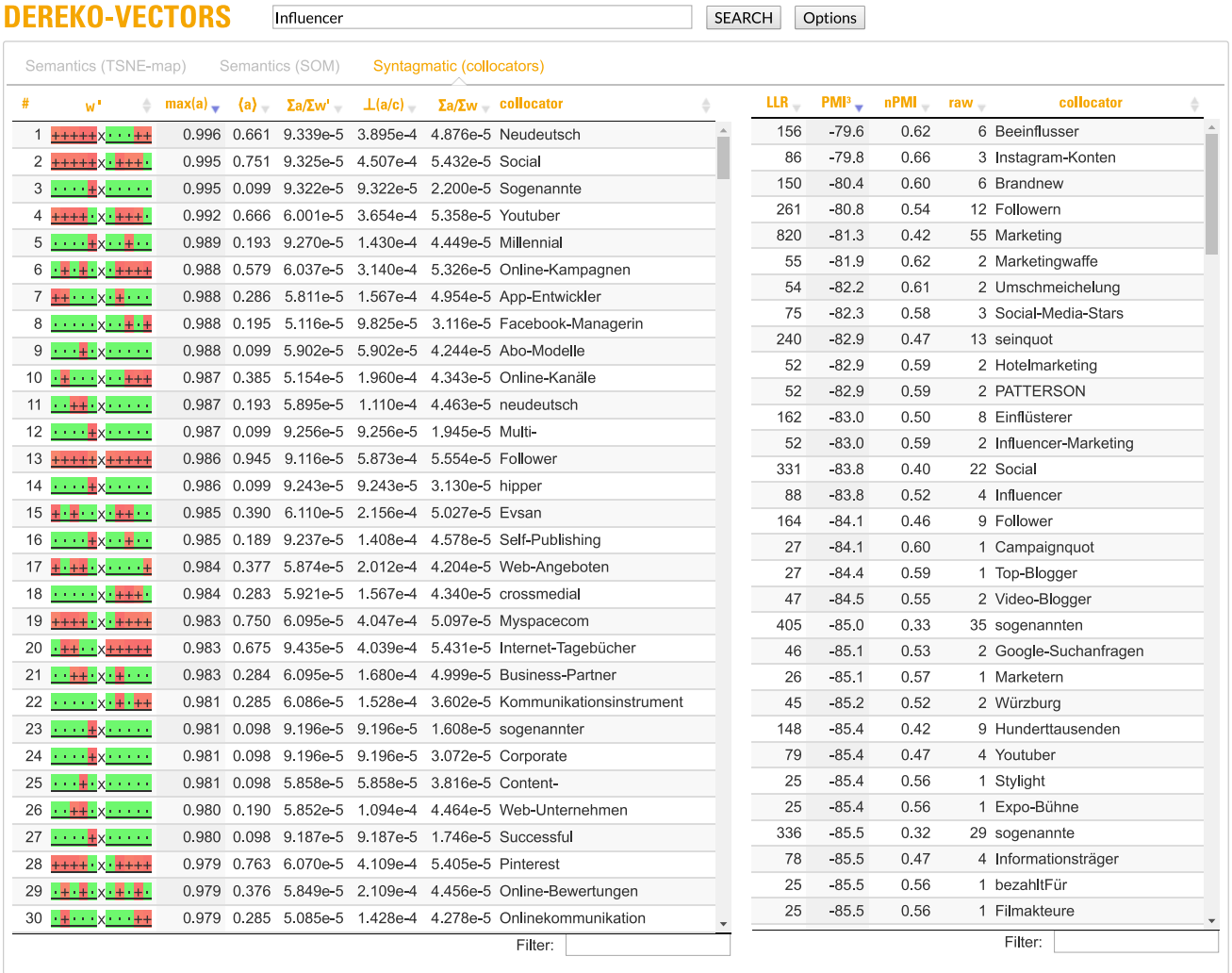


Figure 5: Preview of the upcoming interface for exploring and comparing collocation-based and word-embedding-based DeReKo-models showing the result of a classical collocation analysis for Germany's Anglicism of the Year 2017 "Influencer" on the right-hand side and word-embedding based collocators on the left-hand side.

#	Query (Poliqarp-syntax)	mio. hits
1	`[orth="das" & tt/p=PRELS]`	6,75
2	`[orth="das" & corenlp/p=PRELS]`	6,49
3	`[orth="das" & (tt/p=PRELS | corenlp/p=PRELS)]`	7,60
4	`[orth="das" & (tt/p=PRELS & corenlp/p=PRELS)]`	5,65

Table 3: Query results for 'das', annotated as relative pronoun by: (1) TreeTagger, (2) CoreNLP, (3) TreeTagger or CoreNLP, (4) TreeTagger and CoreNLP – with the respective number of hits in DeReKo-2016-I (Institut für Deutsche Sprache, 2016).

that aim at providing comparable corpora: (1) *The International Comparable Corpus (ICC)* (Kirk and Čermáková, 2017) and (2) *The European Reference Corpus (EuReCo)* (Kupietz et al., 2017). Both initiatives have in common that rather than building comparable corpora from scratch they try to re-use existing corpora and currently mostly rely upon national and reference corpora. However, while the ICC tries to mimic roughly the composition of the International Corpus of English (ICE) (Greenbaum, 1996), EuReCo takes a bottom up approach that relies on DeReKo's and KorAP's concept of *virtual corpora* or *virtual collections* (Kupietz et al., 2010). The key idea of the approach is to allow drawing virtual comparable corpora dynamically from tuples of the (monolingual) source corpora based on metadata categories like date of publication, genre, text type and topic domain, as well as mappings between the respective taxonomies of these metadata categories that are used in the source corpora. Currently, such a mapping is being built for DeReKo and the Reference Corpus of Contemporary Romanian Language (CoRoLa; Romanian Academy 2017, Mititelu et al. 2014) within the DRuKoLa-project (Cosma et al., 2016), and for the Hungarian National Corpus (HNC; Hungarian Academy of Sciences 2018, Oravecz et al. 2014), within the DeutUng project.[6]

5. Conclusion

DeReKo is a very large corpus resource for contemporary German that is continuously expanded and improved. Current developments concern new licensing opportunities, the integration of new text types, and new possibilities for querying and analyzing DeReKo via KorAP – in future also within contrastive research scenarios.

6. Bibliographical References

al Wadi, D. (1994). *COSMAS - Ein Computersystem für den Zugriff auf Textkorpora*. Institut für Deutsche Sprache.

Bański, P., Fischer, P. M., Frick, E., Ketzan, E., Kupietz, M., Schnober, C., Schonefeld, O., and Witt, A. (2012). The New IDS Corpus Analysis Platform: Challenges and Prospects. In Calzolari, N., Choukri, K., Declerck, T., Doğan, M. U., Maegaard, B., Mariani, J., Odijk, J., and Piperidis, S., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul. European Language Resources Association (ELRA).

Beißwenger, M., Lüngen, H., Schallaböck, J., Weitzmann, J. H., Herold, A., Kamocki, P., Storrer, A., and Wildgans, J. (2017). Rechtliche Bedingungen für die Bereitstellung eines Chat-Korpus in CLARIN-D: Ergebnisse eines Rechtsgutachtens. In Beißwenger, M., editor, *Emprirische Erforschung internetbasierter Kommunikation*, volume 9 of *Empirische Linguistik/ Empirical Linguistics*, pages 7–46. de Gruyter, Berlin.

Belica, C., Kupietz, M., Lüngen, H., and Witt, A. (2011). The morphosyntactic annotation of DeReKo: Interpretation, opportunities and pitfalls. In Konopka, M., Kubczak, J., Mair, C., Šticha, F., and Wassner, U., editors, *Selected contributions from the conference Grammar and Corpora 2009*, pages 451–471, Tübingen. Gunter Narr Verlag.

Bodmer, F. (1996). Aspekte der Abfragekompononente von COSMAS-II. *LDV-INFO. Informationsschrift der Arbeitsstelle Linguistische Datenverarbeitung*, 8:112–122.

Bodmer, F. (2005). COSMAS II. Recherchieren in den Korpora des IDS. *Sprachreport*, 3/2005:2–5.

Cosma, R., Cristea, D., Kupietz, M., Tufiş, D., and Witt, A. (2016). DRuKoLA - towards contrastive german-romanian research based on comparable corpora. Proceedings of the LREC-2016-Workshop Challenges in the Management of Large Corpora (CMLC4), pages 28 – 32, Paris. ELRA. `http://nbn-resolving.de/urn:nbn:de:bsz:mh39-52256`.

Diewald, N. and Margaretha, E. (2016). Krill: KorAP search and analysis engine. *JLCL*, 31(1):73–90.

European Comission (2016). *Proposal for a Directive of the European Parliament and of the Council on copyright in the Digital Single Market*. `http://ec.europa.eu/transparency/regdoc/rep/1/2016/EN/1-2016-593-EN-F1-1.PDF`.

Giesbrecht, E. and Evert, S. (2009). Part-of-speech tagging - a solved task? An evaluation of POS taggers for the Web as corpus. In Alegria, I., Leturia, I., and Sharoff, S., editors, *Proceedings of the 5th Web as Corpus Workshop (WAC5)*, San Sebastian, Spain. `http://purl.org/stefan.evert/PUB/GiesbrechtEvert2009_Tagging.pdf`.

Greenbaum, S., editor (1996). *Comparing English Worldwide: The International Corpus of English*. Clarendon Press, Oxford.

Keibel, H. and Belica, C. (2007). CCDB: A corpus-linguistic research and development workbench. In *Proceedings of the 4th Corpus Linguistics conference, Birmingham*. University of Birmingham. `http://corpus.bham.ac.uk/corplingproceedings07/paper/134_Paper.pdf` (07.01.2008).

[6]The DRuKoLA and the DeutUng project are both funded by the Humboldt-Foundation.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics*, 6(1):97–133.

Kirk, J. and Čermáková, A. (2017). From ICE to ICC: The new International Comparable Corpus. In Bański, P., Kupietz, M., Lüngen, H., Rayson, P., Biber, H., Breiteneder, E., Clematide, S., Mariani, J., Stevenson, M., and Sick, T., editors, *Proceedings of the Workshop on Challenges in the Management of Large Corpora and Big Data and Natural Language Processing (CMLC-5+BigNLP) 2017*, pages 7 – 12. IDS. https://ids-pub.bsz-bw.de/frontdoor/index/index/docId/6249.

Kupietz, M. and Belica, C. (2013). Big language data for academic and commercial use. Presentation at the Innovation Days 2013, 4 December 2013, Berlin. http://corpora.ids-mannheim.de/slides/Innovation-Days-2013_KupietzBelica_BigLanguageData.pdf.

Kupietz, M., Belica, C., Keibel, H., and Witt, A. (2010). The German Reference Corpus DeReKo: A Primordial Sample for Linguistic Research. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odjik, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, page 1848–1854, Valletta, Malta. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2010/pdf/414_Paper.pdf (24.10.2013).

Kupietz, M. and Lüngen, H. (2014). Recent developments in dereko. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the ninth conference on international language resources and evaluation (LREC'14)*, pages 2378–2385, Reykjavik, Iceland.

Kupietz, M., Witt, A., Bański, P., Tufiş, D., Cristea, D., and Váradi, T. (2017). EuReCo - Joining Forces for a European Reference Corpus as a sustainable base for cross-linguistic research. In Bański, P., Kupietz, M., Lüngen, H., Rayson, P., Biber, H., Breitenender, E., Clematide, S., Mariani, J., Stevenson, M., and Sick, T., editors, *Proceedings of the Workshop on Challenges in the Management of Large Corpora and Big Data and Natural Language Processing (CMLC-5+BigNLP) 2017*, pages 15 – 19. IDS. https://ids-pub.bsz-bw.de/frontdoor/index/index/docId/6258.

Ling, W., Dyer, C., Black, A., and Trancoso, I. (2015). Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Lüngen, H. (2017). DeReKo – Das Deutsche Referenzkorpus . *Zeitschrift für germanistische Linguistik*, 45(1):161–170. DOI: https://doi.org/10.1515/zgl-2017-0008.

Lüngen, H., Beißwenger, M., Erhardt, E., Herold, A., and Storrer, A. (2016). Integrating corpora of computer-mediated communication in clarin-d: Results from the curation project chatcorpus2clarin. In *Proceedings of KONVENS 2016*, Bochumer Linguistische Arbeitsberichte, pages 156–164.

Margaretha, E. and Lüngen, H. (2014). Building Linguistic Corpora from Wikipedia Articles and Discussions. *Journal of language Technology and Computational Linguistics*, 29(2):59–82. http://www.jlcl.org/2014_Heft2/3MargarethaLuengen.pdf.

Meier, S. (2017). Korpora zur Fußballlinguistik – eine mehrsprachige Forschungsressource zur Sprache der Fußballberichterstattung. *Zeitschrift für germanistische Linguistik*, 45(2):345–349. DOI: https://doi.org/10.1515/zgl-2017-0018.

Mititelu, V. B., Irimia, E., and Tufiş, D. (2014). CoRoLa The Reference Corpus of Contemporary Romanian Language. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. ELRA.

Oravecz, C., Váradi, T., and Sass, B. (2014). The Hungarian gigaword corpus. In Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. ELRA.

Przepiórkowski, A., Górski, R. L., Łaziński, M., and Pęzik, P. (2010). Recent Developments in the National Corpus of Polish. In Calzolari, N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Referentenentwurf (2017). *Entwurf eines Gesetzes zur Angleichung des Urheberrechts an die aktuellen Erfordernisse der Wissensgesellschaft (Urheberrechts-Wissensgesellschafts-Gesetz – UrhWissG)*. Bundesministerium der Justiz und für Verbraucherschutz, Berlin. http://www.bmjv.de/SharedDocs/Gesetzgebungsverfahren/Dokumente/RefE_UrhWissG.pdf;jsessionid=0C34528B94659CE7CF6BA15B7B3516E5.1_cid297?__blob=publicationFile&v=1.

Rosenfeld, V. (2010). An implementation of the Annis 2 query language. Technical report, Humboldt-Universität zu Berlin.

Schröck, J. and Lüngen, H. (2015). Building and Annotating a Corpus of German-Language Newsgroups. In *Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media (NLP4CMC2015)*, pages 17–22, Essen. https://sites.google.com/site/nlp4cmc2015/program.

UrhWissG (2017). Gesetz zur Angleichung des Urheberrechts an die aktuellen Erfordernisse der Wissensgesellschaft (Urheberrechts-Wissensgesellschaftsgesetz). *Bundesgesetzblatt*, Jahrgang 2017 Teil I Nr. 61:3346–3351. http://www.bgbl.de/xaver/bgbl/start.xav?startbk=Bundesanzeiger_BGBl&jumpTo=bgbl117s3346.pdf.

7. Language Resource References

Hungarian Academy of Sciences (2018). Hungarian National Corpus.

Institut für Deutsche Sprache (2016). German Reference Corpus DeReKo. Deutsches Referenzkorpus, DeReKo-2016-I.

Institut für Deutsche Sprache (2018). German Reference Corpus DeReKo. Deutsches Referenzkorpus, DeReKo-2018-I. PID: http://hdl.handle.net/10932/00-03B9-3C6A-6F80-6601-A.

Romanian Academy (2017). Reference Corpus of Contemporary Romanian Language. Romanian Academy, CoRoLa.

Risamálheild: A Very Large Icelandic Text Corpus

Steinþór Steingrímsson[1], Sigrún Helgadóttir[1], Eiríkur Rögnvaldsson[2], Starkaður Barkarson[1], Jón Guðnason[3]
[1]The Árni Magnússon Institute for Icelandic Studies, Reykjavík, Iceland
[2]The University of Iceland, Reykjavík, Iceland
[3]Reykjavik University, Reykjavík, Iceland
steinst@hi.is, sigruhel@hi.is, eirikur@hi.is, stb36@hi.is, jg@ru.is

Abstract
We present Risamálheild, the Icelandic Gigaword Corpus (*IGC*), a corpus containing more than one billion running words from mostly contemporary texts. The work was carried out with minimal amount of work and resources, focusing on material that is not protected by copyright and sources which could provide us with large chunks of text for each cleared permission. The two main sources considered were therefore official texts and texts from news media. Only digitally available texts are included in the corpus and formats that can be problematic are not processed. The corpus texts are morphosyntactically tagged and provided with metadata. Processes have been set up for continuous text collection, cleaning and annotation. The corpus is available for search and download with permissive licenses. The dataset is intended to be clearly versioned with the first version released in early 2018. Texts will be collected continually and a new version published every year.

Keywords: text corpora, Icelandic

1. Introduction

The lack of a very large Icelandic text corpus has been evident for some time. Data oriented methods have increasingly come to dominate the field of NLP and this has led to the need for more data and bigger datasets in order to achieve better performance. In the last few years, with machine learning methods such as neural networks reaching preeminence in various areas of Language Technology (LT) the importance of large text corpora and other textual resources has increased considerably. The compilation of a corpus such as the one described here has therefore been considered a top priority in order to further LT in Iceland (Nikulásdóttir et al. 2017). Large text corpora are e.g. necessary for the design of language models that are used in building a variety of LT tools such as speech recognizers, spell and grammar checkers and automatic machine translation systems.

The aim of the *IGC* project was to compile as large a corpus as possible with the minimum amount of work and resources. The corpus should be clearly versioned in order to facilitate reproducible experiments. The design should make it easy to compare NLP algorithms on contemporary Icelandic and serve as a resource for a wide range of linguistic research, research in the field of culturomics (Michel et al. 2011) and for the interested public. The corpus should be attractive for use in LT projects as well as for other research and study. Therefore we aimed for the following goals:

- The *IGC* will contain more than a billion running words, morphosyntactically tagged and lemmatized and provided with metadata.
- Only digitally available texts will be included in the *IGC*. Formats that may pose a difficulty will not be processed.
- The *IGC* will be open and constantly expanding.
- A closed version will be published every year.
- The *IGC* will be accessible through an online concordance search tool.
- Trend data from the *IGC* will be searchable in an n-gram viewer.
- The *IGC* will be made available for download with a permissive license.

In Section 2 the compilation of the *MIM* corpus (Helgadóttir et al. 2012) is described where the intention was to create a "balanced" and a "representative" text collection. In order to achieve representativity and balance, text was sampled from many genres and often only a very small chunk of text was acquired for each license. There are several problems connected with trying to achieve representativity in a corpus. For the first, what should it be representative of? And because it can be hard to determine where a variety of language ends and another begins any corpus is 'virtually by definition biased to a greater or a lesser extent' (Nelson 2010). As the goal of our current project was to create as large a corpus as possible of contemporary texts in a language spoken by less than 350 thousand people, instead of emphasizing on representativity we aimed for as much coverage as possible and providing extensive meta-data so that users of the corpus can construct their own subcorpora as needed.

A primary design goal for the *IGC* was for it to be open, and that it will be constantly expanding. To make it possible for researchers to verify findings made using the corpus, static versions will be published every year, containing all texts collected up to that point. Furthermore, in order to accomplish our goal of more than a billion words we built a collection of texts from sources where it is possible to acquire material that is not protected by copyright or where it is possible to get big chunks of text for each license secured. The two main sources considered were therefore official text and text from news media. Only digitally available texts are included in the corpus and formats that may pose problems, like pdf documents, were not processed. This results in the corpus as a whole being biased towards journalistic and official texts, but more detailed description of the corpus texts is given in section 3.2.

The texts are morphosyntactically tagged and provided with metadata. Processing pipelines are set up for

continuous text collection, text cleaning and annotation where the processing tools will be continually updated.

The paper is structured as follows. In Section 2 we describe briefly existing Icelandic corpora. In Section 3 an account is given of the creation of the *IGC*, in Section 4 availability of the corpus is discussed and in Section 5 we conclude.

2. Icelandic Corpora

Existing Icelandic corpora will be listed and described briefly in this section to explain their shortcomings and hence the need for a new corpus.

A small corpus was compiled at the Institute of Lexicography[1] for the making of the Icelandic Frequency Dictionary (*IFD*), *Íslensk orðtíðnibók*, published in 1991 (Pind et al. 1991). The *IFD* corpus[2] consists of just over half a million running words. The corpus has a heavy literary bias as about 80% of the texts are fiction. The corpus is annotated with morphosyntactic tags and lemmata. Tagging and lemmatization was manually corrected and hence the corpus has been used as a gold standard for training part-of-speech (PoS) taggers, lemmatizers and parsers. It can be stated that the *IFD* corpus has laid the ground for most work on PoS tagging, lemmatization and parsing that has been performed on Icelandic during the last 15 years.

The Tagged Icelandic Corpus (*MIM*) was released in the spring of 2013, both for search[3] and download.[4] This corpus contains 25 million running words from various genres dating from the first decade of the 21st century (Helgadóttir et al. 2012). The corpus was intended for use in LT projects and for linguistic research. About 86% of the texts are protected by copyright, the remainder being official text (parliamentary speeches, legal text, adjudications and text from government websites). The largest proportion of text, just less than 24%, comes from published books containing both fiction and non-fiction. The second largest portion, about 22%, is taken from newspapers, mostly from printed newspapers. The corpus is annotated with morphosyntactic tags and lemmata and each text segment contains metadata. To enable the use of the corpus in LT projects it was considered important to secure copyright clearance for the texts to be used. All owners of copyrighted text signed a special declaration and agreed that their material may be used free of licensing charges.

MIM-GOLD is a corpus of about 1 million running words which was sampled from the *MIM* corpus (Loftsson et al. 2010; Helgadóttir et al. 2012; Steingrímsson et al. 2015). The corpus is intended as a reliable standard for the development of LT tools. Tagging of this subcorpus has been manually corrected. *MIM-GOLD* can augment the *IFD* corpus for training stochastic taggers and developing LT tools. The *MIM-GOLD* corpus is nearly twice the size of the *IFD* corpus and the texts are more varied, less than 25% of the texts in *MIM-GOLD* are literary texts compared to about 80% of the texts in the *IFD* corpus. Training and testing using the Average Perceptron Tagger *Stagger* (Östling 2012) on *MIM-GOLD* after two correction phases has already been described (Steingrímsson et al. 2015). The result showed that there were still errors in the tagging that needed to be corrected. Work on locating and correcting these errors was completed in fall 2017.

The Icelandic Parsed Historical Corpus (*IcePaHC*)[5] is a diachronic treebank that contains about one million running words from every century between the 12th and the 21st centuries inclusive (Rögnvaldsson et al. 2011). The texts are annotated for phrase structure, PoS-tagged and lemmatized. The corpus is designed to serve both as an LT tool and a syntactic research tool. The corpus is completely free and open since most of the texts are no longer in copyright.

Íslenskur orðasjóður[6] is an Icelandic corpus of more than 550 million running words collected from all domains ending in *.is* during the autumns of 2005 and 2010 (approx. 33 million sentences). Moreover, additional newspaper texts (2 million sentences) and the Icelandic Wikipedia are included. The web texts were cleaned substantially before inclusion in the corpus.

Although the corpora mentioned in this section have been useful in LT and language research they lack in size and/or coverage to fulfill the requirements that present day LT makes. Therefore it was considered necessary to embark on the project of compiling the *IGC*.

3. Creating the Corpus

In Section 1 the aims of the corpus project were described, the primary aim being to compile as large a corpus as possible, at least a billion words, with the minimum amount of work and resources. In this section we will give an account of permissions clearance, text collection and the cleaning and annotation process.

3.1 Permission clearance and licensing

One of the design considerations for the *IGC* was to make the corpus available with a permissive license, such as a Creative Commons license.[7] However, Creative Commons licensing does not seem to be widely known in Iceland so eventually it was necessary to use the same license as was developed for the *MIM* corpus texts for some of the texts in the *IGC*. Some of the copyright protected texts in the *IGC* will be made available with a CC BY license but a great part will be tied to a modified version of the *MIM* corpus license. Work on permission clearance for the first version of the corpus concluded in early 2017. We cleared permission from 20 content providers. Together with text not protected by copyright we have access to more than 40 different text sources. The texts include general and local news from printed media and the web, transcribed television and radio news, commentary on politics and current affairs and texts on scientific matters. Furthermore we collect parliamentary

1 Now a part of the Árni Magnússon Institute for Icelandic Studies.
2 Available at http://www.malfong.is
3 http://mim.arnastofnun.is
4 http://www.malfong.is
5 http://www.linguist.is/icelandic_treebank/
6 http://wortschatz.uni-leipzig.de/ws_ice/
7 https://creativecommons.org/

Text Genre	Word Count	No. of documents	Time period
Newspaper Articles	796,526,434	3,029,985	1998-2017
Parliamentary Speeches	210,699,883	380,557	1911-2017
Adjudications	92,696,289	23,634	1999-2017
Published Books	5,729,543	120	1980-2008
Transcribed Radio/Television News	54,129,050	313,749	2004-2017
Sports News Websites	47,431,733	280,838	2002-2017
Regulations	26,038,153	12,038	1275-2017
Current Affair Blogs and articles	10,511,776	43,678	1973-2017
Informational Articles	10,796,107	55,091	2000-2017
Lifestyle	4,027,506	14,671	2010-2017
Total	1,261,026,503	4,154,528	

Table 1: Retrieved texts for the IGC 2017

speeches, adjudications from courts and a selection of recent fiction and non-fiction from The Árni Magnússon Institute's text collection.

As a consequence the downloadable corpus is divided into two parts: *IGC1* and *IGC2*. *IGC1* contains texts that can be used with a special license developed for *MIM*. The crucial point in the license agreement is that the licensee can use his results freely, but may not publish in print or electronic form or exploit commercially any extracts from the corpus, other than those permitted under the fair dealings provision of copyright law. The license granted to the licensee is non-transferable. *IGC2* contains official texts and texts that can be used with a CC BY license. All copyright holders have agreed that their material may be used free of licensing charges. Copyright owners that did not accept the CC BY license signed a special declaration developed for *MIM* with necessary amendments for the *IGC1*.

3.2 Collecting Texts

A pragmatic approach to text collection was adopted. Texts requiring a minimum of cleaning and processing and texts accompanied by relevant metadata were preferred. This applied to texts obtained from databases of text owners and text harvested from the web. Text in MS Word documents, in Excel spreadsheets or in XML format was also accepted. All documents in the corpus are provided with extensive meta-data, but they can be categorized into 10 genres as shown in Table 1.

3.2.1 Newspaper Articles

Text was acquired from the largest newspaper publishers and news websites in Iceland and a number of smaller publishers, 16 sources in total. Documents from 1998 were acquired from the largest source, *Morgunblaðið*, and from 2004 from the second largest, *Vísir.is*. Documents from these two sources contain more than 75% of the running words for all documents in this category.

3.2.2 Parliamentary Speeches

Texts not protected by copyright were collected from official sources, the biggest of which is Alþingi, the Icelandic parliament, providing parliamentary speeches dating back to 1911 in XML format, containing all relevant meta-data. The speeches have been transcribed and extensively proofread. Although the oldest speeches are from 1911 the bulk of the texts are fairly recent, as documents from the first 30 years (1911-1940) include about 6 million words but documents from the last 30 years (1988-2017) have 120 million words, more than half of this subcorpus.

3.2.3 Adjudications

Adjudications were harvested from the official websites of the Supreme Court of Iceland and the eight district courts of Iceland. Around half the documents come from the Supreme Court, dating from 1998-2017. The other half is from the district courts and date from 2006-2017.

3.2.4 Published Books

Published books from The Árni Magnússon Institute's text collection were incorporated into the corpus. Only books published since 1980 were included. They include fiction and non-fiction and vary considerably in length.

3.2.5 Transcribed Radio/Television News

We received transcribed documents of all news programs from 2006-early 2017 from the two biggest broadcasting companies in Iceland, RUV, the Icelandic National Broadcasting Service and 365 the biggest privately owned media company in Iceland. These transcripts include both scripted news, read by reporters, and transcribed interviews aired on the news programs.

3.2.6 Sports News Websites

Articles from two sport news websites, dedicated to football news, are included in the corpus. In the newspaper category sports news can also be found, but we do not separate the sports articles from those sources, but rather keep them with other articles from the same sources.

3.2.7 Law and Regulations

Icelandic law as of September 2017 is included in the corpus. The oldest documents date back to the 13th century but a majority is fairly recent, with more than 60% of the documents dated in the 21st century.

3.2.8 Current Affair Blogs and Articles

Articles on current affairs from three sources who have all been publishing articles for more than 10 years are included in the corpus. One of the three sources includes opinion pieces published in newspapers since 1973.

3.2.9 Informational Articles

The corpus collection also includes the Icelandic *Wikipedia* and the University of Iceland's *Science Web*. They include informational articles on various topics. Approximately 60% of this material comes from *Wikipedia* and 40% from the *Science Web*.

3.2.10 Lifestyle

This category includes articles from a web site concerned with famous people and lifestyle.

Table 1 lists the ten text genres and word count for texts retrieved for the first version of the *IGC*. In total more than 4 million documents were collected containing 1.26 billion words. About 57% of the texts are a part of *IGC1* or available with a special license and about 43% are a part of *IGC2*, available with CC BY license.

3.3 Text cleaning and annotation

Procedures have been devised for automatic editing and cleaning of the text, annotation and extraction of metadata. No manual post-editing is performed.

A pipeline for harvesting, cleaning and annotating the corpus texts was developed. Individual tools in the pipeline will be continually updated to produce a more precise and reliable annotation with each new version of the corpus.

The annotation phase consists of sentence segmentation, tokenization, morphosyntactic tagging and lemmatization. After morphosyntactic tagging and lemmatization, the texts, together with the relevant metadata, are transferred into TEI-conformant XML format (TEI Consortium 2017). Each document collected for the corpus is distributed in one file, which is comprised of a header, containing metadata and a body which includes the document text, lemmas and morphosyntactic tags.

N-grams (n up to 5) are also created for use with the n-gram viewer and for distribution.

```
<text>
  <body>
    <div1>
      <p n="1">
        <s n="1">
          <w lemma="hver" type="fshen">Hvað</w>
          <w lemma="vera" type="sfg3en">er</w>
          <w lemma="hægur" type="lhensf">hægt</w>
          <w lemma="að" type="cn">að</w>
          <w lemma="lesa" type="sng">lesa</w>
          <w lemma="úr" type="aþ">úr</w>
          <w lemma="þessi" type="favebþ">þessari</w>
          <w lemma="mynd" type="nveþ">mynd</w>
          <c type="punctuation">?</c>
        </s>
      </p>
      <p n="2">
        <s n="1">
          <w lemma="blár" type="lkfnvf">Bláu</w>
          <w lemma="litur" type="nkfng">litirnir</w>
          <w lemma="sýna" type="sfg3fn">sýna</w>
          <w lemma="hvar" type="aa">hvar</w>
          <w lemma="kljósandi" type="nkfn">kljósendur</w>
          <w lemma="liggja" type="sfg3fn">liggja</w>
          <w lemma="þétt" type="aae">þéttast</w>
          <c type="punctuation">.</c>
        </s>
      </p>
    </div1>
  </body>
</text>
```

Figure 1: Text body in an XML file from the corpus.

Figure 1 shows the text body in one TEI-conformant XML document. The text is divided into numbered paragraphs and within each paragraph there are numbered sentences. Each line within the sentences contain different elements for words and for punctuation. The elements for words have lemma and type elements, the type element contains the morphosyntactic tag.

3.4 Tagset

Sentence segmentation and tokenization is performed with the IceNLP toolkit (Loftsson and Rögnvaldsson 2007), the same procedures as were used for the *MIM* corpus (Helgadóttir et al. 2012). *IceStagger* (Loftsson & Östling 2013) is used for tagging the *IGC*. A corpus made by concatenating the *IFD* corpus and the *MIM-GOLD* corpus was used to train IceStagger. Dictionaries used when tagging were augmented with the dictionary of The Database of Modern Icelandic Inflection, *BÍN* (Bjarnadóttir, 2012).

The tagset is a revised version of the tagset used for the *IFD* corpus. A tag for abbreviations has been added and another for e-mail and web addresses. There are more than 670 possible morphosyntactic tags in the tagset, and 559 are found in the corpus. More than 50% of the words in the corpus are tagged with only 16 of the most frequent tags. Examples of tags are shown in Table 2, which lists the 5 most frequent tags in the *IGC*. A complete description of the tagset is available at málföng.is, where the corpus can be downloaded.[8]

Tag	Description	Count	% of Total
aa	Adverb – does not govern case	97,761,983	7.79%
c	Conjunction	93,105,617	7.41%
aþ	Preposition – governs dative	91,594,602	7.29%
ao	Preposition – governs accusative	53,114,369	4.23%
sfg3en	Verb – indicative, active, 3rd person, singular, nominative	48,700,790	3.88%

Table 2: Most frequent tags in the *IGC*

3.5 Lemmatization

A new tool is being developed for lemmatizing Icelandic text. A pre-release version of this tool was used for lemmatizing the *IGC* as results indicate a great improvement over the tool used to lemmatize the *MIM* corpus. A thorough analysis and comparison of the two systems remains to be carried out.

3.6 Metadata

All texts in the corpus are accompanied by metadata. For published texts, the metadata comprises bibliographic data like title, name of author(s), name of editor(s) (if applicable), publisher, date and place of publishing. If it was published on the web the URL is included. For other texts, metadata is recorded to identify the text. For spoken data, various information on the recorded sessions and the speakers is registered. The metadata is shown for each text example retrieved through the search interface and is

8 http://malfong.is/files/rmh_tagset_files_en.pdf

a part of the downloadable texts in TEI conformant XML format. Texts can be selected for search through the search interface classified by publishing date, author and source, which reflects approximately the classification in Table 1.

4. Availability and use

The main object of building the corpus is to make it available for use in LT projects. For other uses, such as linguistics research, teaching, lexicography or other studies the data will be available in a web-based concordance tool on the website malheildir.arnastofnun.is. The Swedish platform KORP[9] (Borin et al. 2012) which in turn uses the *IMS* Corpus Workbench[10] (Evert & Hardie 2011) as a search engine was adapted to be used with the corpus. Users of the search interface can take advantage of the annotation of the texts when specifying search criteria. Texts will be added continually to the searchable corpus as they become available.

The corpus texts are available for download in the TEI-conformant XML format (TEI Consortium 2017). As mentioned in Section 3.1 the corpus has been divided into two parts, *IGC1* and *IGC2* for download where *IGC1* is made available with a special license developed for the *MIM* corpus and *IGC2* with CC BY license. This situation is reflected in the download procedures. The corpus is available for download through the Icelandic LT resources website *Málföng*.[11]

The corpus texts are also available through an n-gram viewer based on NB N-gram viewer (Birkenes et al. 2015). The n-gram viewer is accessible on n.arnastofnun.is.

To aid developers of LT tools the corpus website allows download of the n-grams (n up to 5) used for the n-gram viewer.

5. Conclusion and further work

The new Icelandic Gigaword Corpus is a valuable resource for builders of LT tools for Icelandic. It is also useful for linguists, lexicographers, teachers, journalists and others working with or researching the Icelandic language.

The compilation of the corpus will be an ongoing process although closed versions will be published yearly. Official texts will be added continually and texts protected by copyright as long as permission for the use of the text has been secured. The tools in the corpus pipeline will also be upgraded as better tools or versions become available and the corpus texts reannotated to reflect improved precision and reliability of the tools.

6. Acknowledgements

The corpus was compiled during the years 2015 to 2017 at the Árni Magnússon Institute for Icelandic Studies and was funded mostly by the Infrastructure Fund (no. 151110-0031, project manager Eiríkur Rögnvaldsson), the Contribution Grants Fund (Mótframlagasjóður) at the University of Iceland and by the Icelandic Ministry of Education and Culture. The authors would like to thank Gunnar Thor Örnólfsson for setting up the search interface and Kristján Rúnarsson for helping with text collection. Thanks are also due to the developers of Korp, especially Lars Borin. We would also like to thank the anonymous reviewers for helpful comments. Last, but not least, thanks are due to publishers and editors who gave permission for the use of their text and made digital copies available. Without the cooperation of all these people this project could not have been completed.

9 https://spraakbanken.gu.se/korp/

10 http://www.ims.uni-stuttgart.de/forschung/projekte/CorpusWorkbench.html

11 http://www.malfong.is /

7. Bibliographical References

Birkenes, M., Johnsen, L., Lindstad, A. and Ostad, J. (2015). From digital library to n-grams: NB N-gram. In Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA-2015), NEALT Proceedings Series Vol. 23, pages 293–295, Vilnius, Lithuania.

Bjarnadóttir, K. (2012). The Database of Modern Icelandic Inflection. In Proceedings of the workshop "Language Technology for Normalization of Less-Resourced Languages"– SaLTMiL 8 – AfLaT2012 at the 8th International Conference on Language Resources and Evaluation (LREC 2012), pages 67–72, Istanbul, Turkey.

Borin, L., Forsberg, M. and Roxendal, J. (2012). Korp – the corpus infrastructure of Språkbanken. In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012), pages 474–478, Istanbul, Turkey.

Evert, S. and Hardie, A. (2011). Twenty-first century Corpus Workbench: Updating a query architecture for the new millennium. In Proceedings of the Corpus Linguistics 2011 conference, University of Birmingham, Birmingham, UK.

Helgadóttir, S., Svavarsdóttir, Á., Rögnvaldsson, E., Bjarnadóttir, K. and Loftsson, H. (2012). The Tagged Icelandic Corpus (MIM). In Proceedings of the workshop "Language Technology for Normalization of Less-Resourced Languages"– SaLTMiL 8 – AfLaT2012 at the 8th International Conference on Language Resources and Evaluation (LREC 2012), pages 67–72, Istanbul, Turkey

Loftsson, H., Yngvason, J., Helgadóttir, S. and Rögnvaldsson, E. (2010). Developing a PoS-tagged corpus using existing tools. In Proceedings of "Creation and use of basic lexical resources for less-resourced languages", workshop at the 7th International Conference on Language Resources and Evaluation (LREC 2010). Valetta, Malta.

Loftsson, H. and Rögnvaldsson, E. (2007). IceNLP: A Natural Language Processing Toolkit for Icelandic. In Proceedings of Interspeech 2007, Special Session: "Speech and language technology for less-resourced languages", Antwerp, Belgium.

Loftsson, H. and Östling, R. (2013). Tagging a Morphologically Complex Language Using an Averaged Perceptron Tagger: The Case of Icelandic. In Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA-2013), NEALT Proceedings Series 16. Oslo, Norway.

Michel, J., Shen, Y., Aiden, A., Veres, A., Gray, M., The Google Books Team, Pickett, J., et al. (2011). Quantitative analysis of culture using millions of digitized books. In *Science* 331(6014): 176-182.

Nelson, M. (2010). Building a written corpus. In A. O'Keeffe & M. McCarthy (Eds.), *The Routledge Handbook of Corpus Linguistics* (53-65). New York: Routledge.

Nikulásdóttir, A., Guðnason, J. and Steingrímsson, S. (2017). Máltækni fyrir íslensku 2018-2022: verkáætlun. [Language Technology for Icelandic 2018-2022: Strategic Plan] Mennta- og menningarmálaráðuneytið. Reykjavík.

Pind, J., Magnússon, F. and Briem, S. (1991). Íslensk orðtíðnibók [The Icelandic Frequency Dictionary]. The Institute of Lexicography, University of Iceland. Reykjavík, Iceland.

Rögnvaldsson, E., Ingason, A., Sigurðsson, E. and Wallenberg, J. (2011). Creating a Dual-Purpose Treebank. In *Journal for Language Technology and Computational Linguistics*, 26(2):141–152.

Steingrímsson, S., Helgadóttir, S. and Rögnvaldsson, E. (2015). Analysing Inconsistencies and Errors in PoS Tagging in two Icelandic Gold Standards. In: Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA-2015), NEALT Proceedings Series Vol. 23, pages 287–291, Vilnius, Lithuania.

TEI Consortium, eds. (2017). TEI P5: Guidelines for Electronic Text Encoding and Interchange. 3.2.0. Last updated on 10th July 2017. TEI Consortium. <http://www.tei-c.org/Guidelines/P5/> (September 2017).

Östling, R. (2012). Stagger: A modern POS tagger for Swedish. In: Proceedings of the Swedish Language Technology Conference, SLTC, Lund, Sweden.

TriMED: A Multilingual Terminological Database

F. Vezzani[1], **G. M. Di Nunzio**[2], **G. Henrot**[1]
[1]Dept. of Linguistic and Literary Studies, [2]Dept. of Information Engineering
University of Padua, Italy
federica.vezzani@phd.unipd.it, giorgiomaria.dinunzio@unipd.it, genevieve.henrot@unipd.it

Abstract

Three precise categories of people are confronted with the complexity of medical language: physicians, patients and scientific translators. The purpose of this work is to develop a methodology for the implementation of a terminological tool that contributes to solve problems related to the opacity that characterizes communication in the medical field among its various actors. The main goals are: i) satisfy the peer-to-peer communication, ii) facilitate the comprehension of medical information by patients, and iii) provide a regularly updated resource for scientific translators. We illustrate our methodology and its application through the description of a multilingual terminological-phraseological resource named TriMED. This terminological database will consist of records designed to create a terminological bridge between the various registers (specialist, semi-specialist, non-specialist) as well as across the languages considered. In this initial analysis, we restricted to the field of breast cancer, and the terms to be analyzed will be extracted from a corpus in English, accompanied by all relevant linguistic information and properties, and re-attached to their pragmatic equivalent in Italian and French.

Keywords: medical language, terminology, methodologies and tools for LRs construction and annotation

1. Introduction

Communication in the healthcare domain is characterized by a rigid and closed nomenclature which in many cases produces an opaque lexicon difficult to understand. Medical language often contains inconsistencies of scientific terminology such as semantic ambiguity, incorrect use of suffixes, archaism maintaining, redundancy in the formation of compounds, and etymological inconsistencies (Rouleau, 2003). As a result, patients and in general non-experts in medicine are often exposed to medical terms that can be semantically complex and hardly understandable. Moreover, despite the substantial amount of health-related information available on Internet, little is known about the quality and accessibility of that information. As a consequence, consumers using the Internet may have difficulties finding complete and accurate information on health issues. Deficiencies in information could negatively influence consumer decisions if people are relying on the Internet to make treatment decisions, including whether to seek care or not.[1] Moreover, it is important to state that due to the existence of an international language of communication, that is English, medical vocabulary is full of foreign words that can create problems during the transfer of medical knowledge across different languages.

In this work, we present a methodology for the implementation of a terminological tool that contributes to solve problems related to the opacity that characterizes communication in the medical field among its various actors. The main goals are: i) satisfy the peer-to-peer communication, ii) facilitate the comprehension of medical information by patients, and iii) provide a regularly updated resource for scientific translators. This work aims to provide a multilingual tool, a cross-evaluation study in which the languages considered are English, Italian and French.

The paper is organized as follows: after an overview of works previously developed in this regard, Section 2., we proceed in Section 3. by determining the three categories of people identified as subjects involved and affected by the complexity of medical language: physicians, patients, and technical translators. In Section 4., we go through identifying a working methodology that is the basis of the proposed linguistic resource as well as the description of the linguistic tool. We give our final remarks in Section 5.

2. Related Works

Regarding the complexity of the medical language, numerous studies demonstrate how it appears difficult to understand health information contained in drug package inserts (PATEL et al., 2002), in websites (MCCRAY, 2005; CENTER, 2008), and more generally in patients and medical doctor's communication (MCCRAY, 2005; JUCKS and BROMME, 2007; TRAN et al., 2009).

Patients are often exposed to complex medical terms and numerous research focuses on the concept of understandability related to this subject. The study of (GRABAR et al., 2014) proposes a specific lexicon in order to assess which words are potentially non-understandable and then require further explanations. The implementation of a specific lexicon in which the words are rated according to whether they are understandable or non-understandable for the medical field is considered as a first step towards the simplification of medical texts.

The understanding of words is a complex notion closely linked to Natural Language Processing (NLP) research field. Its purpose is to decide whether given documents are accessible for a given reader. The readability measures are widely used for evaluating complexity of documents (BOUAMOR et al., 2016) and it is possible to distinguish two types of readability measures: classical and computational (FRANÇOIS and FAIRON, 2013). Classical measures are essentially based on the number of characters and/or syllables in words, sentences or documents. Computational measures might involve vector space models and

[1]Health Internet ethics: ethical principles for offering Internet health services to consumers.

a wide range of descriptors and their combinations (ZENG et al., 2005; FRANÇOIS and FAIRON, 2013; LEROY et al., 2008) but, text readability formulas are mostly based on word length and sentence length. However, as McCray states in his study (MCCRAY, 2005), "Zen counting words and syllables and consulting a grade-level word list are most likely not sufficient to determine how readable a text is". This has prompted researchers (KESELMAN et al., 2007) to design more appropriate measures for medical texts which take into account term familiarity and recognition of the lexical form.

The difference between the language used by health care professionals and that used by patients is cited as a source of miscommunication (ELHADAD and SUTARIA, 2007). For example, non-expert people tend to use idiomatic expressions such as "mal de chien" (Fr) [literally, "dog pain"(En)] to refer to "douleur intense" (Fr) [severe pain, En].

The biomedical domain offers many linguistic resources for Natural Language Processing, including terminologies and corpora. However, most of these resources are prominently available for English and the access to terminological resources in languages other than English may not be so simple. Furthermore, there is a large audience of non-English speakers who can benefit from accessing health information in their native language. In this regard, (NEVEOL et al., 2014) review the extent of resource coverage for French and give pointers to access French-language resources.

In this paper, we present a methodology for the evaluation of the understandability and readability of medical language through the implementation of a terminological tool. Our main goal is then provide a linguistic resource available in order to satisfy the need for effective communication between various actors and the transmission of information in a clear and understandable way in three languages (English, French, and Italian). For this reason, our tool aims not only to provide information from a strictly linguistic point of view (like other annotation projects such as Framenet[2], Verbnet[3] or English PropBank[4]) but also to satisfy the needs of the user categories identified in Section 3.

3. Three Categories of Users

In terms of divulgation of scientific knowledge, we have proceeded by identifying three categories of people with problems related to the opacity of the medical vocabulary for different aspects and different levels of communication: physicians, patients and scientific translators.

3.1. Physicians

The international scale release of medical knowledge implies that most of the scientific texts are produced in English. For example, for several years many Italian medical journals have accept contributions only in English and have even anglicized their own denominations: 'Cardiologia' in 'Italian Heart Journal' since 2002, and 'Rivista Italiana di Pediatria' in 'Italian Journal of Pediatrics' since 2001 (SERIANNI, 2005). In terms of spreading new health care protocols and scientific discoveries, language could be a barrier to service transactions among medical specialists speaking different languages because perfect knowledge and mastery of the foreign language is not an expected outcome. At a level of peer-to-peer communication, and then specialist-to-specialist, physicians need to overcome these language barriers and access scientific research information in their mother tongue. In this way, experts could not only import new knowledge on the national territory but also export their scientific discoveries by inserting them into the international circuit. In this way, the direct benefit is raising the awareness of the importance of a proper terminology in the scientific communication between languages, the existence of possible false friends and the availability of tools specifically designed to respond with the utmost precision and reliability.

3.2. Patients

Scientific and technological development has so much influence on medicine and its diagnostic and therapeutic capabilities. This fact has shifted the focus of the physician's attention not on the patient but on the illness itself, and this has led to a crisis in the physician-patient relationship. Patients find a considerable difficulty in understanding information, both oral and written, about their own health with regard to clinical interactions despite laws and policies, emphasizing the real need to document the various health aspects in a more comprehensible way [5]. Physician-patient interaction implies a level of specialist-non-specialist communication, so patients (or more generally the public) would need to understand medical expertise by using their correspondent in the "popular" language or by using an appropriately calibrated language for the communication to be effective. It is also important to consider the increasingly frequent use of the Web as a source of medical and health information. Search engines are commonly used to access information available online but many resources are far from being effective in order to respond adequately to user requests and this may have serious consequences. Furthermore, the fact of exposing people with poor medical knowledge to a complex medical language can lead to self-diagnosis and erroneous self-treatment. In this sense, the Higher Institute of Health in Rome, Italy, has promoted the MEDUSA[6] project (MEDicina Utenti SAlute in rete), which is a citizen portal for the retrieval of qualified and reliable health information on the web. The implementation of this portal is part of the activities of a wider health education project funded by the Ministry of Health, titled *Alfabetizzazione sanitaria ed empowerment del paziente attraverso lo sviluppo di un sistema informativo elettronico nel campo della salute*[7] aiming to raise awareness in health issues and provide access, through a single platform, to information and resources of different types and nature.

[2]https://framenet.icsi.berkeley.edu/fndrupal/

[3]http://verbs.colorado.edu/ mpalmer/projects/verbnet.html

[4]https://propbank.github.io/

[5]https://goo.gl/5Avgrd

[6]https://medusa.iss.it/

[7]Health Literacy and Patient Empowerment through the Development of an Electronic Health Information System https://goo.gl/6dihwr

3.3. Technical-scientific Translators

In the sphere of communication and dissemination of information, translation, as a practical discipline, acquires a fundamental role in the correct transmission of information in different languages. In order that the translated text responds to the deontological principle of the discipline, that is, the fidelity to the source text, the technical-scientific translator must proceed step by step, decoding, deeply understanding and faithfully translating the semantic and informative content of the text. In this case, the level of communication is at a specialist semi-specialist degree. Moreover, the needs of the translation market do not allow time to conduct in-depth terminological research, forcing the professional to skip some key steps for optimal work. Translators need regularly updated terminology resources which can support them in the realization of the final product.

At present, one of the most reliable bilingual resource for terminology and translation which is available on the market is an Italian /English bilingual database of Medical Subject Headings (MeSH). This tool is used in the indexing of articles in biomedical journals of PubMed, by the Higher Institute of Health of Rome. But, as with many other medical dictionaries online, the resource is derived from terminology cards that can complete and meet the requirements outlined above.

4. TriMED

The needs outlined above for the three identified categories of users would find a valid application in a multilingual terminology database. This work aims to provide a methodology for the development of TriMED, that is a multilingual terminological database gathering a set of terminological records for each selected technique terms, at each pragmatic level identified. The resource is named TriMED because the tripartite character is intrinsic in the tool: three are the working languages (English, French, and Italian), three are the identified user categories (physicians, patients, and translators) and, consequently, three are the communication levels which are the object of the analysis of this tool. The type of textual corpus from which technical nomenclature will be extracted concerns oncology and it will collect English-language articles related to care protocols for breast cancer patients. From the corpus so drawn, the medical technical terms will be extracted and analyzed. The term record will report the product of semantic analysis and will provide translations, drawn from parallel corpus in the target languages: either Italian or French. Starting from these, the corresponding technicalities will be selected to meet the multilingual goal of the terminological database.

4.1. Structure

TriMED records will be articulated on three levels of communication:

- specialized communication, providing both scientific definitions to meet the peer communication between experts and the corresponding technical terms translated;
- semi-specialized communication, providing useful information to the technical-scientific translator for the translation of texts for this medical domain, such as semantic analysis of the term, instances of collocations, colligations, hyperonym, iponyms etc.
- non-specialized communication, providing both informative definitions to facilitate proper understanding by patients and the equivalent of the technical term commonly used in the popular language.

TriMED database will provide not only the simple translation of the term in English (in the three registers indicated), but also all the information necessary to make the medical technical term clear and semantically accessible.

4.1.1. Data Collection

At this stage, we proceeded by the selection of the source corpus and parallel corpora and by the extraction of technical terms.

First, we have selected a set of English-language articles representing our source corpus for the analysis of medical terms. In this initial phase of gathering an initial dataset and validating the application, we limit the topic of interest to breast cancer treatment protocols, but in the future, we expect to extend the domain of interest to other medical areas. Documents are selected from specialized online magazine reviews based on the highest impact factor value, such as "Breast Cancer research and treatment"[8], "Archives of Breast Cancer"[9], and "European Journal of Cancer Care"[10]. Afterwards, we have created parallel corpora for the other two languages analyzed: Italian and French. The sources we have drawn for the Italian language are: Fondazione Umberto Veronesi (in particular, protocols belonging to the initiative "Pink is Good"), AIMaC - Associazione Italiana Malati di Cancro, and A.N.D.O.S - Associazione nazionale donne operate al seno. While for the French language: Fondation du Cancer du Sein du Quebec, l'Association francophone pour les soins oncologiques de support (AFSOS) and Cairn.info. Then, we have proceeded through the manual extraction of medical terms for the three corpora. By technical terms we mean all terms that are closely related to the conceptual and practical factors of a given discipline or activity, in this case terms are related to the medical-oncological field.

4.2. Web Application

In this section, we apply the previous outlined methodology by the creation of a Web application that can respond to the principles underlying in this paper. For each technical term, we provided a set of information for the three categories of identified users. Every term has been supplied with:

- Equivalent in the informative and popular language;
- Definition;

[8] http://www.springer.com/medicine/oncology/journal/10549

[9] http://archbreastcancer.com/

[10] http://onlinelibrary.wiley.com/journal/10.1111/(ISSN)1365-2354

- Semantic analysis of the term;
- Formal features.

Upon these information, we have created TRIMed Interface that is presented as a "three-headed" tool: from the homepage you can select the category in which you identify and then access to the related information you need. In Figure 1, we show the main panels of the application: physician (Fig. 1c), translator (Fig. 1a), and patient (Fig. 1b). We have implemented the application with the Shiny R package (CHANG, 2015), the demo is available online to show how the interaction among the three levels of communication works[11].

Consulting the interface in "patient" mode, you can search the technical term (with the suggestion of completing the term) and its equivalent in the popular language and its definition are provided. With the aim of simplifying medical language, this tool allows the patient to "translate" the medical term with a more simplified or commonly used terminology. Furthermore, considering the ternary programming lines (specialists, semi-specialists, non-specialists), we intended to provide the possibility of consulting an "informative" variant for the technical term in question, which is to be considered different from the term in the popular language. This implementation is still being developed and assessed as we have done a distinction between term in popular language, by classifying it as a recurrent word mainly in oral talks between patients (for example, in clinic and hospital waiting rooms), and semi-specialist (or informative) term that has its place in informative articles and which is then extracted from written corpus. There may be perfect correspondence between popular terms and informative terms, as both are intended for the use of a non-specialized user in this field. But it is still interesting to allow for this further linguistic consideration in view of the diastratic evaluation of terminology.

As far as the "translation" mode, the interface is designed for providing to the user the purely linguistic and terminological information that underlie the translating process. After selecting the term, the user can visualize a screen presented in the form of a double-read table: vertical and horizontal. By proceeding with a vertical analysis, the user is able to consult the translation of the term in its scientific, informative and popular language version. Subsequently, the definition of the technical term and its semic analysis will be provided in the decomposition of the meaning of the lexical or morphological unit into atomic units or components of the not further segmentable meaning. Finally, the user can access to the formal features of the term necessary for its lexical framing:

- Gender;
- Pronunciation;
- Derivation and composition of the term.

This information is necessary for a translator in the choice of the translating candidate of the term taken into consideration. Keeping the table with the source term, the translator can set the target language and consult the information of the chosen translating term that will appear in a table adjacent to the source table. In this way, the user can access the same information for the selected translating term and can consult horizontally the information for the two terms.

Finally, the access mode as a "physician" user is an interface that offers in the translation point of view the opportunity for the physician to consult the technical term in his or her mother tongue. This user has the opportunity as well to select the source language and the target language in which he/she wants to examine the word and its related definition. For this user category, a direct link with related MeSH[12] terms has also been provided. By clicking on the term, the physician can access to the various information provided directly by the National Library of Medicine.

5. Conclusions

This work has been developed with a view to the evaluation of medical language in terms of understandability and readability. Through our methodology of language analysis, we are creating a linguistic resource that could answer the initial questions and needs outlined by the three categories of users. At the present time, TriMED consists of a set of 200 technical terms for French and Italian languages. We are working on English version in order to fullfill the multilingual goals.

We are planning to gather enough data to cover medical terminology in the oncology field, in particular breast cancer treatments, by trying to propose a resource that does not only include English but also other target languages requiring documented terminology in that field. Our intent is then provide a linguistic tool consisting of 2500 – 3000 technical terms for this specific domain.

Hence, the structure and idea behind TriMED allow for future implementations. For example, with regard to the enrichment of the "physician" mode, we propose to allow the user to have the direct access to articles from scientific journals related to that precise technical terms. For example, for English terms, a direct link between Mesh terms and PubMed related articles may be helpful. While for Italian terms, the link could be made with the bilingual Mesh database provided by the Roma Healthcare Institute. Similarly, French terminology could be directly linked to the InSerm site[13] which, in co-operation with Inist-CNRS (Istitut de l'Information scientifique et technique du CNRS) contributes to updating the French version of Mesh terms since 2004.

Finally, TriMED is designed to support data and knowledge discovery and integration as well as promote sharing and reuse of data by following the FAIR principles of the research data in Horizon 2020.[14]

6. Bibliographical References

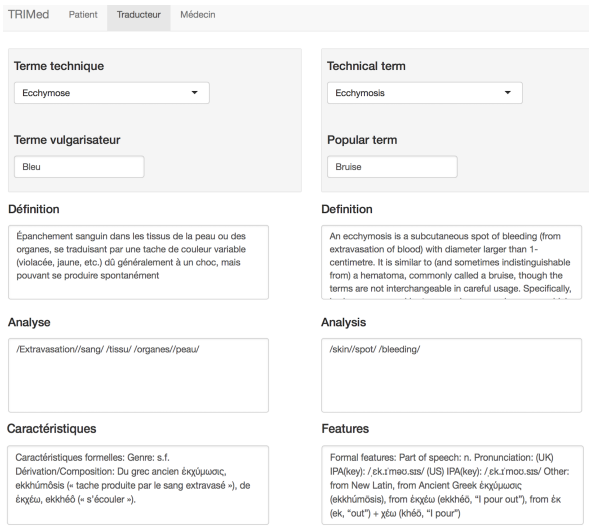
BOUAMOR, D., LLANOS, L. C., LIGOZAT, A.-L., ROSSET, S., and ZWEIGENBAUM, P. (2016). Transfer-based learning-to-rank assessment of medical term tech-

[11] https://gmdn.shinyapps.io/TriMED/

[12] https://www.nlm.nih.gov/mesh/

[13] http://mesh.inserm.fr/mesh/

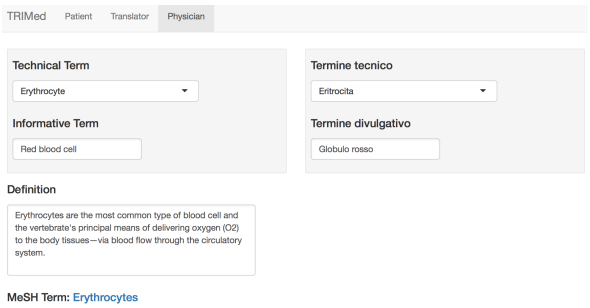
[14] https://goo.gl/WSUd9K



(a) Translator interface (French to English)



(b) Patient interface (Italian)



(c) Physician interface (English to Italian)

Figure 1: Main panels of the TRIMed Web application.

nicality. In *Proceedings of the LREC 2016*, Portorož, Slovenia, May. ELRA.

CENTER, O. P. (2008). Barriers and drivers of health information technology use for the elderly, chronically ill, and underserved. technical report, agency for healthcare research and quality. Oregon Evidence-based Practice Center.

CHANG, W., (2015). *Shiny: Web Application Framework for R*. R package version 0.11.

ELHADAD, N. and SUTARIA, K. (2007). Mining a lexicon of technical terms and lay equivalents. In Proc BioNLP Workshop, pages 49-56. ACL.

FRANÇOIS, T. and FAIRON, C. (2013). Les apports du tal à la lisibilité du français langue étrangère. TAL, 54(1): 171-202.

GRABAR, N., VAN ZYL, I., DE LA HARPE, R., and HAMON, T. (2014). The comprehension of medical words. In *Proceedings of the International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 5*, BIOSTEC 2014, pages 334–342, Portugal. SCITEPRESS - Science and Technology Publications, Lda.

JUCKS, R. and BROMME, R. (2007). Choice of words in doctor-patient communication: an analysis of health-related internet sites. Health Commun, 21(3):267-77.

KESELMAN, A., TSE, T., CROWELL, J., BROWNE, A., NGO, L., and ZENG, Q. (2007). Assessing consumer health vocabulary familiarity: an exploratory study. J Med Internet Res, 9(1): e5.

LEROY, G., HELMREICH, S., COWIE, J., MILLER, T., and ZHENG, W. (2008). Evaluating online health information: Beyond readability formulas. In AMIA Annual Symposium Proceedings.

MCCRAY, A. (2005). Promoting health literacy. J of Am Med Infor Ass, 12:152-163.

NEVEOL, A., GROSJEAN, J., DARMONI, S., and ZWEIGENBAUM, P. (2014). Language resources for french in the biomedical domain. In *Proceedings of the LREC 2014*, Reykjavik, Iceland, May. ELRA.

PATEL, V., BRANCH, T., and AROCHA, J. (2002). Errors in interpreting quantities as procedures: The case of pharmaceutical labels. International journal of medical informatics, 65(3):193-211.

SERIANNI, L. (2005). Un treno di sintomi. i medici e le parole: percorsi linguistici nel passato e nel presente. Milano, Garzanti Libri.

TRAN, T., CHEKROUD, H., THIERY, P., and JULIENNE, A. (2009). Internet et soins : un tiers invisible dans la relation médecine/patient. Health Commun, 21(3):267-77.

ZENG, Q., KIM, E., and CROWELL, J.and TSE, T. (2005). A text corpora-based estimation of the familiarity of health terminology. In ISBMDA, pages 184-192, Aveiro, Portugal.

Preparation and Usage of Xhosa Lexicographical Data for a Multilingual, Federated Environment

Sonja Bosch, Thomas Eckart, Bettina Klimek, Dirk Goldhahn, Uwe Quasthoff
Department of African Languages, University of South Africa;
Natural Language Processing Group, Agile Knowledge Engineering and Semantic Web Group, Leipzig University
boschse@unisa.ac.za, {teckart, klimek, dgoldhahn, quasthoff}@informatik.uni-leipzig.de

Abstract

The South African linguistic landscape is characterised by multilingualism and the influence between their eleven official and some local languages. Unfortunately, for most of the languages the amount and quality of available lexicographical data is suboptimal, even though its availability is essential for all educational institutions and for the development of state-of-the-art language technology. In this paper we present a new source of lexicographical data for Xhosa, a language spoken by more than eight million speakers. For its utilisation in a multilingual and federated environment it is modelled using a dedicated OWL ontology for Bantu languages and possesses all features that are currently considered integral for the promotion of resource reuse as well as long-term usage. In the future, the introduced ontology may be used for other Bantu languages as well and may ease their combination to achieve more extensive, multilingual data stocks.

Keywords: Xhosa, lexicography, research infrastructures, linked data

1. Introduction

A basic requirement for the language processing capability for any language is the availability of lexicographical data, ideally open source data which is often hard to find for less resourced languages. This includes many members of the Bantu language family. For enhancing the usability of this kind of data this paper presents a Bantu Language Model for describing lexicographical data in RDF. Furthermore, it presents a new resource of lexicographical data for the Xhosa language based on this new model. The data to be presented in this model is a representative sample of raw data for a Xhosa-English dictionary, containing approximately 6,800 lexical entries. In its final state, the data set should contain approximately 10,000 lexical entries. Whereas the available data enables us to use Xhosa as the language of instantiation, the method and model are extensible and applicable to many other Bantu languages, in particular those belonging to the same group.

Together with this paper, both the Bantu Language Model and the current state of the lexicographical data set are freely available for download and for querying via a dedicated SPARQL endpoint. It should be noted that the research reported on in this paper is work in progress. Its final version will be provided via SADiLaR, the South African Centre for Digital Language Resources. Moreover, the current version of the data is already available via CLARIN-D (see section 4.).

The remainder of this paper is structured as follows: Section 2 describes the origin of the Xhosa language material and explains essential features of the Xhosa language with a focus on its morphology. Section 3 explains the new Bantu Language Model that is based on the established MMoOn ontology[1]. Section 4 gives detailed information about the structure of the Xhosa RDF data set using a concrete example. Furthermore, information about its current extent are provided. Section 5 demonstrates the relationship of the described work in the context of federated research infrastructures and how they can simplify access to and enhance usability of modern lexicographical data. The paper closes with a short summary and an outlook to planned further work.

2. The Xhosa Source Data

The data used for this case is based on Xhosa [xho][2], one of the official languages of South Africa belonging to the so-called Bantu language family. It is spoken predominantly in the Eastern Cape and Western Cape regions. There are approximately 8.1 million Xhosa speakers[3], adding up to about 16% of the South African population. Xhosa, as member of the Nguni language group, shares many linguistic features with other Nguni languages, which include Zulu [zul], Swati [ssw], Southern Ndebele [nbl] and Northern Ndebele [nbe][4]. Xhosa, like the other Bantu languages, is structurally agglutinating and is therefore characterised by words usually consisting of more than one morpheme. Each morpheme corresponds to a single lexical meaning or grammatical function. This particular Xhosa lexicographical data set is accompanied by English translations and was compiled and made available for purposes of further developing Xhosa language resources[5]. The process involved digitisation into CSV tables and various iterations of quality control in order to make the

[1] http://mmoon.org/core/

[2] Each language is followed by its ISO 639-3 code http://www.loc.gov/standards/iso639-2/php/code_list.php in order to distinguish one language from other languages with the same or similar names and to identify the names of cross-border languages.

[3] http://www.statssa.gov.za/census/census_2011/census_products/Census_2011_Census_in_brief.pdf

[4] The names of the Nguni languages in the languages themselves are respectively: isiXhosa, isiZulu, Siswati and isiNdebele.

[5] Bilingual (Xhosa-English) word lists were compiled by JA Louw after his retirement with the intention of documenting Xhosa words and expanding existing bilingual Xhosa dictionaries by means of among others botanical, animal and bird names, grammar terms, modern forms etc., as well as lexicalisations of verbs with extensions.

data reusable and shareable. In this paper, we concentrate on nouns and verbs. The excerpt of the lexicographical data set is a representative sample of Xhosa nouns and verbs. Nouns of all possible regular and irregular combinations of noun classes, and verbs with a variety of verbal extensions (leading to lexicalisations in meaning) are represented. Nouns are listed alphabetically according to noun stems, followed by the POS, the surface form of the singular and plural class prefixes (if applicable) as well as the number(s) of the class prefixes, and finally the English translations, e.g.

Noun stem	*POS*	*Class pref sg*	*Class no.*
phathi	noun	um	1
Class pref pl	*Class no.*	*English translation*	
aba	2	superintendent	

Verbs are listed alphabetically according to verb stem, i.e. the basic verb root followed by the inflection suffix -a, or sometimes -i, e.g.

Verb stem	*POS*	*English translation*
mi	verb	be standing
tyalisa	verb	help to plant

The lexicographic data is by no means based on corpus frequencies of nouns and verb stems as for instance the Oxford School Dictionary (De Schryver, 2014) but rather on complementation of existing, established dictionaries.

2.1. Xhosa Morphology

The noun is made up of two main parts, namely a noun prefix and a noun stem. All nouns are assigned to a particular class, as reflected in the class prefix. For practical and comparative purposes, noun classes have been given numbers by scholars working in the field of Bantu linguistics. Although 23 such classes have been reconstructed in Proto-Bantu, most Bantu languages have fewer than 20 classes (Nurse and Philippson, 2003). In Xhosa for instance, the class numbers end with class 17, while classes 12 and 13 do not occur at all (Pahl, 1967). It should be added that class 15 represents the infinitive class, while class 16 is no longer used productively to form nouns in Xhosa, but rather has an adverbial significance. Each noun class is characterised by a distinct prefix, which also includes a pre-prefix, and a particular singular/plural pairing with uneven numbers signifying singular and even numbers signifying plural. These class prefixes may show agreement with other constituents in a sentence. The only class pair with specific semantic contents is class 1/2 which contains personal nouns only. This does not, however, mean that all personal nouns occur in this class pair. For the rest of the noun classes, semantic arbitrariness is observed, although certain semantic generalisations do occur, e.g. classes 9 and 10 are generally referred to as the "animal classes" since they contain many animal names, but also many other miscellaneous terms. Noun stems may also be suffixed with morphemes such those indicating diminutive, augmentative, derogatory or feminine modifications to the basic meaning of the noun. A verb consists of a series of prefixes and suffixes that are built around a basic verb root carrying the basic meaning. A final inflection suffix completes the verb stem, to which pre-stem inflection is added in the form of, for example, the following morphemes: subject agreement, object agreement, negation, tense and aspect. Verbal suffixes may include morphemes such as: negation and derivational extension. The verb therefore carries much information and is pivotal in the sentence.

2.2. Discussion of the Xhosa Data

In the Xhosa data set under discussion, noun stems are separated from their class prefixes as is the case in traditional Bantu language dictionaries. In each instance the class prefix modifies the meaning of the basic noun stem e.g.

balo (isi 7; izi 8) "arithmetic"
balo (u 11) "census"

Although noun stems can be sub-divided into a root plus suffixes, any suffixes that occur, e.g. the feminine suffix *–kazi* and the diminutive suffix *–ana*, are not identified separately in our data. This is illustrated in the following examples where the modification of the basic meaning only appears in the English translations:

caka (isi 7; izi 8) "servant"
cakakazi (isi 7; izi 8) "servant girl"
cakazana (isi 7; izi 8) "young servant girl"

Noun class pairs normally signify singular/plural that correspond to the odd and even class numbers respectively, e.g.

khwenyana (um 1; aba 2) "son-in-law"
kroti (i 5; ama 6) "hero"

There are exceptions, however, for instance the singular class 11 takes its plural in class 10 (instead of 12, which does not exist in Xhosa), e.g.

diza (u 11; iin 10) "straw"

Also, the distinction between singular and plural does not apply to nouns that denote, for example, mass or abstract concepts, as in the case of:

bisi (u 11) "milk"
ophu (um 3) "vapour"

The following examples demonstrate that phonetically and phonologically conditioned allomorphs of class prefixes 1/2 (um-/aba- versus um-/ab-); 7/8 (isi-/izi- versus is-/iz-) and 11 (u- versus ulu-) appear in the data, e.g. in the case of vowel initial noun stems or monosyllabic noun stems (Kosch, 2006):

biki (um 1; aba 2) "reporter" vs. ongi (um 1; ab 2) "nurse"
kolo (isi 7; izi 8) "school" vs. enzo (is 7; iz 8) "deed, act"
patho (u 11) "school" vs. bi (ulu 11)

"misfortune, calamity"

These examples, therefore, demonstrate that for some noun classes more than one prefix member exists, resulting in allomorphs that occur in complementary distribution.
Verb stems are listed according to their infinitive form minus the infinitive prefix, i.e. the basic verb root followed by the inflection suffix *–a*. In some few cases, the final suffix presents as *–i* or *-e*. The latter only occurs in the case of stative verbs such as *–krekrelele* "stand in line". In the data there is no morphological differentiation between basic verb stems and verb stems with suffixed extension morphemes. The modification of the basic meaning of the verb stem, however, appears in the English translation, as in:

tenda "entertain"
tendana "entertain one another"
tendeka "be able to be entertained"
tendela "entertain at or for"
tendisa "help to entertain"

3. The Bantu Language Model

For the representation of the tabular Xhosa dictionary data and their translations we chose to convert the data into the RDF (Resource Description Framework) format. The mapping of the source data to RDF, however, requires a specific vocabulary which can be some existing or newly created ontology. While the lexicon model for ontologies (Lemon) (McCrae et al., 2011) was designed to represent lexical language data, its usage has been proven to be problematic for Bantu languages (Chavula and Keet, 2014). This is mainly due to the lack of the conceptualisation for morphological language data. Even though the Lemon model evolved to become a W3C recommendation published as the OntoLex-Lemon model that is split into five specified modules[6] (McCrae et al., 2017), the necessary modelling of morphological data has not been worked into this refined model.
Therefore, we created the Bantu Language Model[7] (in short BantuLM) as illustrated in Figure 1. This ontology is fully based on the reuse of and alignment to already existing vocabularies[8]. The largest part is based on the Multilingual Morpheme Core Ontology (MMoOn Core)[9] because it provides fine-grained classes and properties for representing morphological data and, moreover, already shares a considerable amount of overlap to the ontolex module for lexical data (Klimek, 2017). By taking the Xhosa verb and noun source data as an orientation point we identified three major linguistic subdomains that were to be modelled: 1) lexicographic data which is based on the OntoLex lime module[10] and MMoOn Core, 2) morphological data which is solely based on MMoOn Core, and 3) translational data which is based on the OntoLex vartrans module[11]. Despite the best practice recommendation to make direct reuse of existing vocabularies if they appropriately fit the modelling domain in question, a different approach of vocabulary reuse has been taken. In order to represent the BantuLM under a single namespace, all classes and properties have been newly created, however, corresponding to the reused external vocabularies. I.e. all classes that are based on MMoOn Core have identical labels and are aligned by usage of the `rdfs:subClassOf` object property in accordance to the creation procedure for MMoOn Core-based data sets. Otherwise, all classes that are based on the ontolex and lime vocabulary are interconnected with their derived counterparts via the `owl:equivalentClass` object property. The equivalence of all object properties within the BantuLM vocabulary is created with the `owl:equivalentProperty` property. Consequently, all definitions of the classes and properties need to be obtained from the interconnected original vocabularies. This poses, however, no disadvantage since the naming of the classes and properties is quite self-explanatory. While this kind of duplication of vocabularies is rather unusual it is formally valid in terms of ontology creation. This modelling of the BantuLM vocabulary has been chosen in preference of user-friendliness given that the data creators are mainly linguists that have only little or no expertise in creating language resources in the RDF format or within the Linked Data framework. It is assumed that a vocabulary that is applicable to all Bantu languages is easier to use and query for non-experts if it is built on a single namespace instead of a variety of vocabularies that need to be studied before they can be actually used for language data representation.
To conclude, the BantuLM is an aggregation of those classes and properties from the mentioned vocabularies that are necessary or useful to represent not only the Xhosa source data but also other Bantu languages in general, e.g. we had no data for the class `blm:Wordform`, but other Bantu language resources might well have and can then use this class accordingly. In contrast to the reused models the BantuLM is a language-specific model and, hence, specified for its affiliation to the Bantu language family. That means in particular, that grammatical meanings such as wordclass, number or nominal classifier are newly created and consequently specific to and shared by all Bantu language resources that will be based on the BantuLM ontology.
For the creation of the Xhosa RDF inventory data set the BantuLM proved not only to be fully suitable but also contributed to an explicit semantic interrelation between the lexical and morphological elements which is rather implicit in the tabular source data[12].

[6] `https://www.w3.org/community/ontolex/wiki/Final_Model_Specification`

[7] The ontology is available under the URI: `http://mmoon.org/bnt/schema/bantulm/`.

[8] Please consult the ontology URI for more information on how to use the ontology for creating other Bantu language data.

[9] Cf. `http://mmoon.org/` and `http://mmoon.org/core/` for more information.

[10] `http://www.w3.org/ns/lemon/lime#`

Model for lexical and morphological data of Bantu languages.

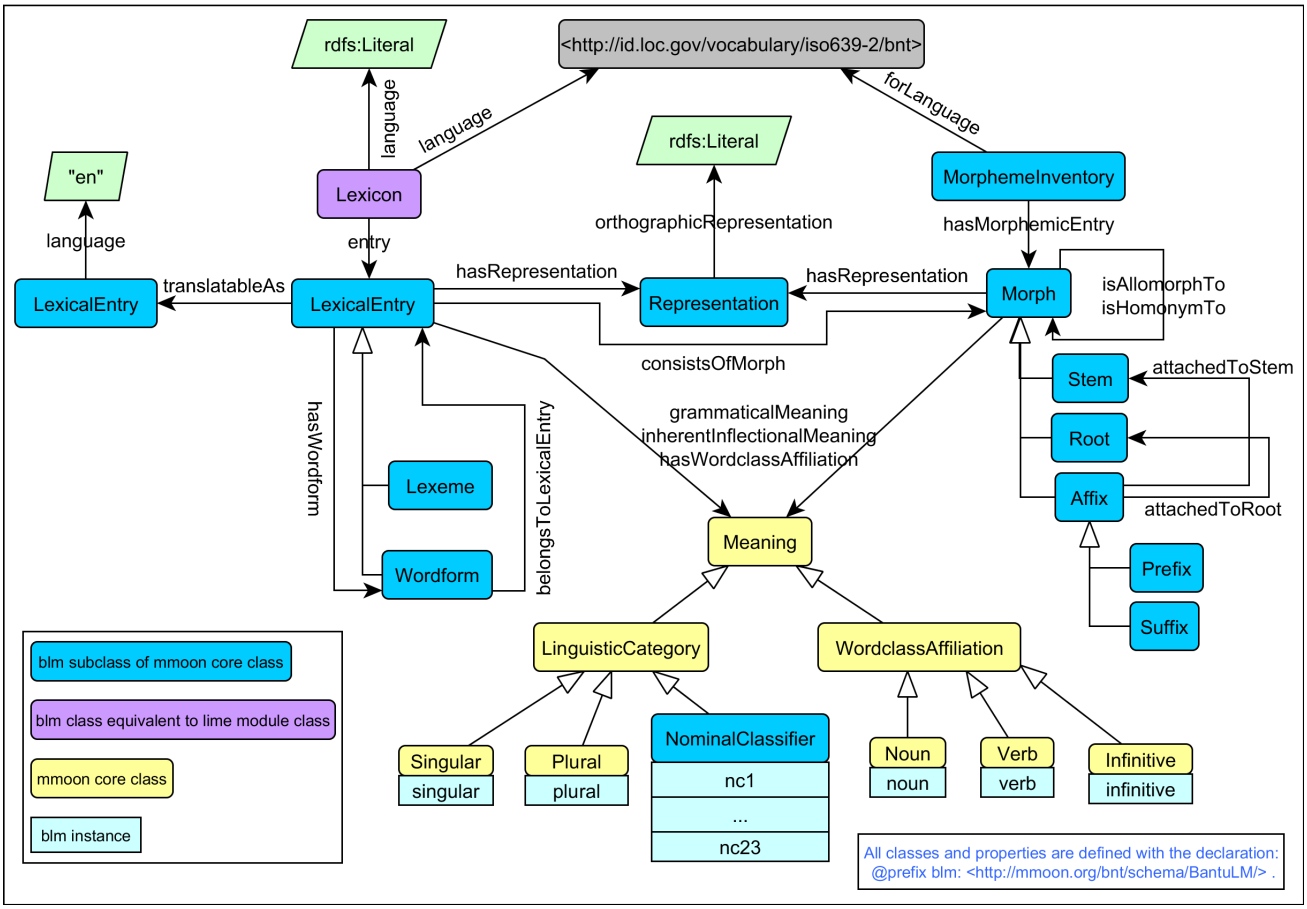


Figure 1: Ontology for the Bantu Language Model.

4. The Xhosa RDF Data set

The creation of the BantuLM ontology enabled the conversion from the Xhosa tabular source data into the Xhosa RDF data graph without any data loss. Necessary meta data is explicitly stated within the data set declaring information such as the data set creator, version and the underlying license.

In addition to the source data, the ontology-based representation of the Xhosa language data allowed for an explication of indirectly contained linguistic information. This is exemplified in Figure 2 which illustrates the graph representation of the lexical and morphological data.

With regard to the lexical data it can be seen that unique lexeme resources, like `xho_inv:lexeme_umbiki_n`[13] and `xho_inv:lexeme_ababiki_n`[14], have been created which were formerly separated as root and affix entries within the tabular data. As for the morphological data, the relationship that holds between affixes could be further specified by making use of the two object properties `blm:isAllomorphTo` and `blm:isHomonymTo`. That is, Figure 2 shows that the prefixes *aba-* and *ab-* are allomorphs to each other since they share the same meaning (noun class 2 and plural) but differ in their orthographic representation. Not illustrated, but included in the data set, are the homonymous relations that hold between affixes that share the same orthographic and/or phonological representation but differ in meaning. Such detailed linguistic information might be very useful for linguistic research investigating Bantu noun class systems.

Next to this internal enrichment of the tabular source data, the Xhosa RDF data set has been also externally enriched by linking the English translations, e.g. `xho_inv:trans_reporter_n` to lexical entries of the WordNet RDF data set[15] (McCrae et al., 2014). The object property `owl:sameAs` has been used to automatically create appropriate links. The full equivalence between the Xhosa RDF and WordNet RDF lexical resources is assured because only those lexemes have been interlinked that consisted of exactly one and the same word and also agreed in their part of speech. Figure 2 shows an example linking

[11] `http://www.w3.org/ns/lemon/vartrans#`

[12] To examine the increased expressivity, please compare an example of the source and RDF data here: `http://mmoon.org/lrec2018figures/`

[13] `http://rdf.corpora.uni-leipzig.de/resources/xho/inventory/lexeme_umbiki_n`

[14] `http://rdf.corpora.uni-leipzig.de/resources/xho/inventory/lexeme_ababiki_n`

[15] Please cf. `http://wordnet-rdf.princeton.edu/about` for more information. The data set can be found here: `http://wordnet-rdf.princeton.edu/static/wordnet.nt.gz`.

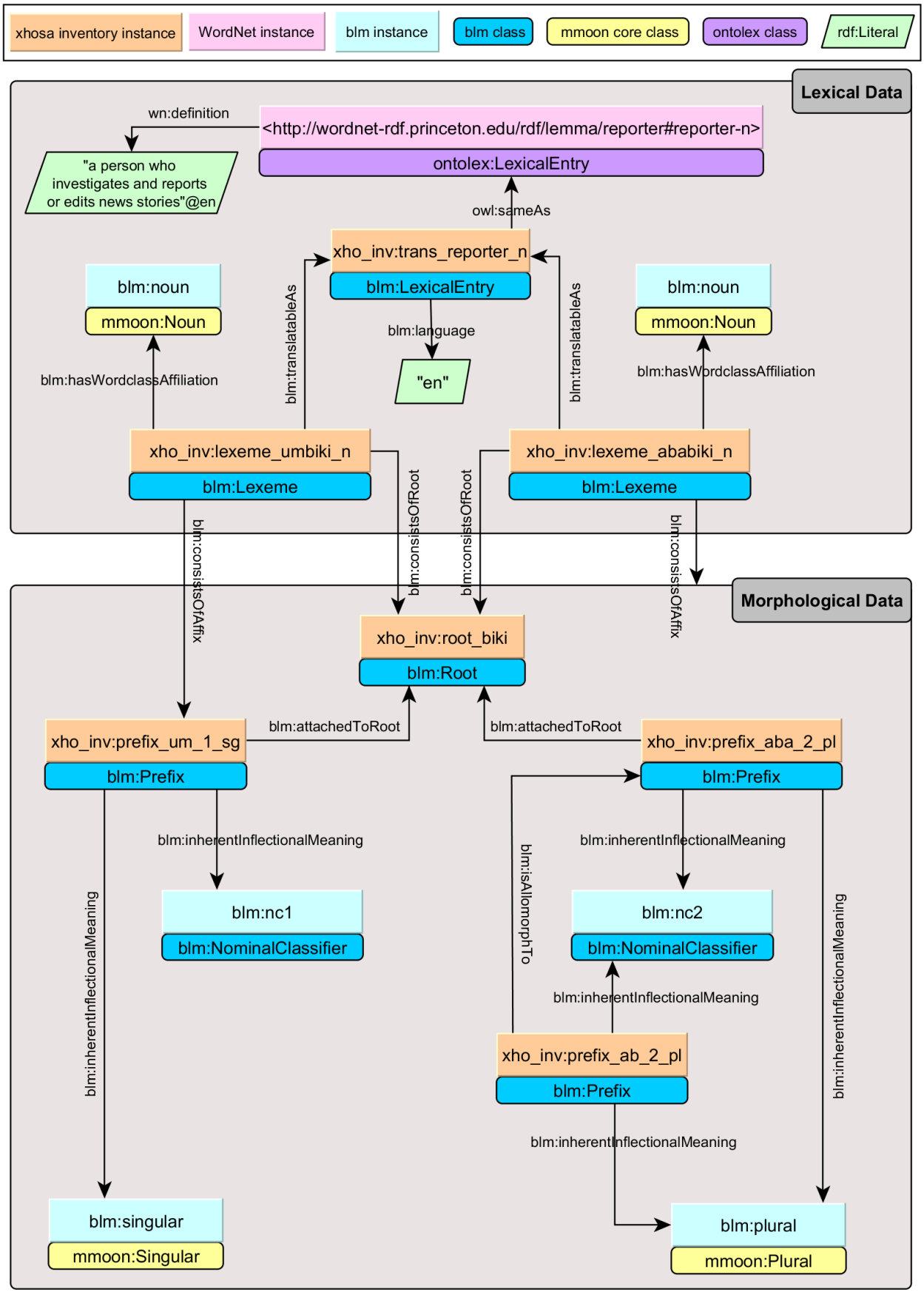


Figure 2: Excerpt from the Xhosa RDF data graph.

for the English translation *reporter* of the Xhosa nouns *umbiki* (singular) and *ababiki* (plural) to the corresponding WordNet lexical entry. Further, it can be seen that this WordNet entry ultimately[16] leads to a sense definition of the lexeme *reporter*. As a result, the interlinking of the Xhosa English translations with the WordNet RDF lexical entries, consequently, leads to an enrichment of the Xhosa noun and verb lexemes with corresponding lexical senses. Senses or sense definitions have not been part of the source data but are now accessible for all Xhosa lexemes whose translations are linked to WordNet and can be obtained by traversing through the interconnected data graph. While this enrichment with lexical senses already leads to a more coherent lexical data set for Xhosa, the linking to WordNet RDF entails an additional value in the context of the multilingual Bantu language landscape. Provided that more Bantu language data sets will be similarly converted into RDF and interlinked with WordNet, an interconnection of different Bantu language data sets could be realised by using the WordNet RDF as the pivot data basis for a multilingual Bantu language data graph.

Finally, the Xhosa RDF data set has been validated by using the RDF Unit[17] (Kontokostas et al., 2014) which conducts syntactic and semantic data quality tests of RDF data, which have been all passed by the Xhosa RDF data set.

In summary, the presented Xhosa RDF data set generates an added-value in comparison to its underlying tabular source data due to the successful internal and external data enrichment just explained. The Xhosa RDF data set in its current state contains 4,014 noun and 2,763 verb lexemes, 66 affixes as well as 2,818 links from the English translations to WordNet RDF. The Xhosa RDF data set is available within the LLOD Cloud and also accessible here: `https://github.com/MMoOn-Project/OpenBantu/blob/master/xho/inventory/ob_xho.ttl`. Moreover, the SPARQL endpoint provided at the URL `http://rdf.corpora.uni-leipzig.de/sparql` enables the querying of the data set to obtain deeper insights into the Xhosa language data.

5. Lexicographical Infrastructures in a Federated Environment

Despite strong efforts and significant progress towards open access to linguistic resources over the last years, many languages still lack those resources or their uncomplicated availability for larger user groups. Therefore, the presented work should not only be seen as another building block for a more complete landscape of linguistic resources, but in the context of federated and distributed infrastructures in a sometimes complex political and administrative environment.

Many countries with heterogeneous linguistic environments have decided to promote joint efforts for documenting their native languages for the benefit of education — primary, secondary, and academic — or the promotion of language technology, which currently is often only available for a highly resourced subset. This is especially problematic in a larger context where rights on relevant resources are held by different institutions with a varying degree of openness and each providing their own proprietary access interfaces.

As a consequence of this rather typical situation many large-scale infrastructures in the field of linguistic resources promote the usage of service-oriented architectures (SOAs) that provide data and services via standardised Web interfaces and data models. One of the benefits of this approach is that data can still be hosted by the publishing institution — being the main authority for the specific resource — and still allow access for the broader (or academic) public while promoting use and re-use in an active research environment.

In the South African context, the recently established Centre for Digital Language Resources[18] (SADiLaR) is a new research infrastructure with a focus on the creation, management and distribution of digital language resources of all official languages of the country. The ultimate aim is to provide a central repository for reusable language resources as well as applicable software tools that will be made freely available for research purposes (cf. Roux (2016)).

In the European context, CLARIN-D (cf. Hinrichs and Krauwer (2014)) is a long-term digital research infrastructure for language resources in the Humanities and Social Sciences. This includes language data bases, highly interoperable language technology tools as well as web-based language processing services. Researchers and students of Humanities and Social Sciences can use resources and technologies easily and in a standard way, without having to deal with technical complexities. The CLARIN-D infrastructure is built upon a network of centres, each of which with its own established competence and international reputation. For the time being, the described resource is hosted via CLARIN-D's infrastructure.

In our work we utilize this approach of making data available based on a standardised data model, i.e. the MMoOn Core ontology as the main basis of the BantuLM ontology, that has already proven to be adequate for describing morphological and lexical data (Klimek et al., 2016) and that is especially suitable to be used for other members of the Bantu language family as well.

The strict separation of data model, technical interface and end-user applications in a service-oriented environment opens the data for innovative applications. Among others, this is especially relevant for the field of meta-lexicography in the context of a multilingual environment. Besides the benefit of combining resources hosted and administered in different locations by different institutions, a SOA is a suitable backbone for enhancing usability with the major aim of addressing and reaching new user groups. This can be established by creating specific portals for different target audiences with varying and partially incompatible requirements. The specific demand may range from looking up simple words for language learners

[16] Please note, that there are several nodes in the WordNet RDF graph between the lexical entries and the sense definitions which are omitted in the Figure.

[17] `http://aksw.org/Projects/RDFUnit.html`

[18] http://www.nwu.ac.za/sites/www.nwu.ac.za/files/files/p-ctext/documents/Graphics.RMA.Newsletter.1.0.3.LvdB.2016-11-23.pdf

to concrete usage examples for dictionary enrichment or highly specific information of different linguistic fields for academic studies. Naturally, aspects such as necessary functions, form and content aspects and intended use are playing a vital role here (Gouws et al., 2007).

6. Summary and Outlook

The presentation of a new Xhosa lexicographical resource for a multilingual federated environment is an example for the transformation of isolated and unpublished dictionary data to the digital age. However, the data set used to develop the BantuLM ontology is only a snapshot of a resource in development. Currently, more lexemes are curated and quality assurance methods will be used to improve the already available data constantly. The publication date of the final data set is expected to be within the next 15 months.

The Bantu Language Model described in this paper can be used for many more languages. Dictionary data is available in a variety of formats, see, for instance, *http://www.cbold.ish-lyon.cnrs.fr/Dico.asp* with dictionaries for about 70 Bantu languages with 5,000 to 10,000 entries per dictionary.

A next logical step is the construction of a user interface to use this data as an actual online dictionary. For comfortable dictionary look-up an additional morphological analysis would be helpful. Again, a unified approach for many Bantu languages seems possible here. As most existing dictionaries translate to English or French, the transitive connection of several dictionaries can be used to interconnect different Bantu languages and allow their combination to a joined "virtual" resource for the whole language family in the future.

Acknowledgment The various phases of research activities related to this paper were funded by grants from the: H2020 EU projects ALIGNED (GA-644055); Smart Data Web BMWi project (GA-01MD15010B); BMBF project CLARIN-D (01UG1620C); South African Centre for Digital Language Resources (SADiLaR); Erasmus+ Programme; Scientific eLexicography for Africa project; and the South African National Research Foundation. The late JA Louw is acknowledged for making the Xhosa data available for purposes of further developing Xhosa language resources.

7. Bibliographical References

Chavula, C. and Keet, C. (2014). Is lemon Sufficient for Building Multilingual Ontologies for Bantu Languages? In *OWLED*, pages 61–72.

De Schryver, G.-M. (2014). Oxford school dictionary: Xhosa-english. Cape Town. Oxford University Press Southern Africa.

Gouws, R. H., Heid, U., Schweickard, W., and Wiegand, H. E. (2007). Dictionaries. an international encyclopedia of lexicography. supplementary volume: Recent developments with special focus on computational lexicography. an outline of the project. *Edited by Fredric FM Dolezal et al.*, page 262.

Hinrichs, E. and Krauwer, S. (2014). The CLARIN Research Infrastructure: Resources and Tools for e-Humanities Scholars. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 1525–1531, May.

Klimek, B., Arndt, N., Krause, S., and Arndt, T. (2016). Creating Linked Data Morphological Language Resources with MMoOn – The Hebrew Morpheme Inventory. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC-2016), 23-28 May 2016, Slovenia, Portoroz*.

Klimek, B. (2017). Proposing an OntoLex-MMoOn Alignment: Towards an Interconnection of two Linguistic Domain Models. In *Proceedings of the LDK 2017 Workshops: 1st Workshop on the OntoLex Model*, pages 68–73.

Kontokostas, D., Westphal, P., Auer, S., Hellmann, S., Lehmann, J., Cornelissen, R., and Zaveri, A. (2014). Test-driven evaluation of linked data quality. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 747–758, New York, NY, USA. ACM.

Kosch, I. M. (2006). Topics in morphology in the african language context. Unisa Press.

McCrae, J., Spohr, D., and Cimiano, P. (2011). Linking lexical resources and ontologies on the semantic web with lemon. In *Extended Semantic Web Conference*, pages 245–259, Berlin, Heidelberg. Springer.

McCrae, J., Fellbaum, C., and Cimiano, P. (2014). Publishing and linking wordnet using lemon and rdf. In *Proceedings of the 3rd Workshop on Linked Data in Linguistics*.

McCrae, J. P., Bosque-Gil, J., Gracia, J., Buitelaar, P., and Cimiano, P. (2017). The ontolex-lemon model: Development and applications. In *Electronic lexicography in the 21st century: Lexicography from Scratch. Proceedings of the eLex 2017 conference.*

Nurse, D. and Philippson, G. (2003). *The Bantu languages*. London: Routledge.

Pahl, H. (1967). *isiXhosa*. Johannesburg: Educum.

Roux, J. C. (2016). South African Centre for Digital Language Resources. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC-2016), 23-28 May 2016, Slovenia, Portoroz*.

A Lexicon of Discourse Markers for Portuguese – LDM-PT

Amália Mendes[1], Iria del Rio[1], Manfred Stede[2], Felix Dombek[2]
[1] University of Lisbon, Centre of Linguistics, Portugal
[2] University of Potsdam, Germany
amaliamendes@letras.ulisboa.pt, igayo@gmail.com, stede@uni-potsdam.de, felix_dombek@hotmail.com

Abstract

We present LDM-PT, a lexicon of discourse markers for European Portuguese, composed of 252 pairs of discourse marker/rhetorical sense. The lexicon covers conjunctions, prepositions, adverbs, adverbial phrases and alternative lexicalizations with a connective function, as in the PDTB (Prasad et al., 2008; Prasad et al., 2010). For each discourse marker in the lexicon, there is information regarding its type, category, mood and tense restrictions over the sentence it introduces, rhetorical sense, following the PDTB 3.0 sense hierarchy (Webber et al., 2016), as well as a link to an English near-synonym and a corpus example. The lexicon is compiled in a single excel spread sheet that is later converted to an XML scheme compatible with the DiMLex format (Stede, 2002). We give a detailed description of the contents and format of the lexicon, and discuss possible applications of this resource for discourse studies and discourse processing tools for Portuguese.

Keywords: Discourse markers, Lexicon, Discourse treebank

1. Introduction

The Lexicon of Discourse Markers (LDM-PT) provides a set of lexical items in Portuguese that have the function of structuring discourse and ensuring textual cohesion and coherence at intra-sentential and inter-sentential levels (Halliday & Hasan, 1976). Each discourse marker (DM) is associated to the set of its rhetorical senses (also named discourse relations or coherence relations), following the PDTB 3.0 sense hierarchy (Webber et al., 2016).

We consider that discourse connectives do not vary regarding inflection, they express a two-place semantic relation, have propositional arguments and are not integrated in the predicative structure. This includes conjunctions, adverbs and adverbial phrases, but also prepositions and alternative lexicalizations, as we discuss in section 4.

Our immediate goal is to provide data for the annotation of discourse relations in a Portuguese discourse treebank, although a listing of DMs will certainly prove to be useful for applications dealing with tasks such as parsing, text processing and summarization of Portuguese.

We revisit in Section 2 other lexicons of DMs, their features and structure schemata; we discuss in Section 3 the acquisition of the DMs that we integrate in our lexicon and in Section 4 the information provided for each DM. In Section 5, we present the way this information is structured and the result in XML format, while we discuss in Section 6 the use of such a lexicon in discourse studies and its applications in the automatic processing of discourse. Finally, we provide some concluding remarks in sSection 7.

2. Related work

A lexicon of DMs may be restricted to discourse connectives, i.e., devices that assure cohesion at intra and inter-sentential levels (typically, conjunctions and adverbial phrases) or it can have a larger scope by also including pragmatic markers with interactional and modal meanings (Cuenca and Marín, 2009). Even under a more restrictive perspective, there are differences in the set of categories included in lexicons. The question is additionally related to the acquisition method: while a lexicon that is compiled manually and is informed mainly by grammars and dictionaries will be more restrictive in terms of the categories and items listed, a lexicon (semi-) automatically derived from a discourse treebank will typically include a larger set of devices that the annotators have found to fulfil a cohesive function. Example of such cases are the Alternative Lexicalizations included in the Penn Discourse Treebank (PDTB) (Prasad et al., 2008; Prasad et al., 2010) and the secondary connectives (and free connective phrases) in the Prague Discourse Treebank (Rysová and Rysová, 2015), that fall outside the traditional categories associated to discourse connectives.

There are few lexicons of DMs currently available, although recent initiatives are reported for several languages. The German lexicon DiMLex (Stede, 2002) includes 275 connectives and provides information on orthographic variants, non-connective readings, focus particle and syntactic category. The association of discourse relations to each connective in DiMLex is described in Scheffler and Stede (2016). The Italian lexicon LiCO contains 173 connectives and follows closely the DiMLex structure (Feltracco et al, 2016). For French, there is LEXCONN, a large lexicon with 328 connectives, with information on their syntactic category and their discourse relation, based on SDRT (Roze et al., 2012). The DPDE is an online dictionary of Spanish DMs with 210 entries in html format. The DMs are not labelled with a rhetorical sense, but a definition is provided, together with detailed information on each connective, such as register, prosody, formulae and comparable DMs (Briz et al, 2003). Recently, the design of a Czech lexicon of DMs that exploits the Prague Dependency Treebank was presented in Mírovský et al. (2016).

Lexical resources available for Portuguese deal essentially with content words and even those focusing on multi word expressions favour content expressions. However, the DPDE online does provide a Portuguese equivalent to the set of Spanish discourse particles, and an experiment in the fully automatic identification of multilingual lexica including Portuguese has been reported (Lopes et al., 2015). In this context, the LDM-PT lexicon provides a new resource for discourse studies in Portuguese.

3. The acquisition of DMs

The identification of DMs comes from several sources. First of all, we used a list of single and phrasal elements belonging to grammatical classes, such as conjunctions and prepositions, compiled during the preparatory work for the POS annotation of the Reference Corpus of Contemporary Portuguese (Généreux et al., 2012).

We also automatically identify the DMs that are labelled as connectives in the Portuguese part of the TED-MDB corpus (Zeyrek et al., 2018). The TED-Multilingual Discourse Bank, or TED-MDB, is a parallel corpus of English TED talks transcripts and their translations in 5 languages (German, Russian, Polish, Portuguese and Turkish). The transcripts are manually annotated at the discourse level following the goals and principles of PDTB (Prasad et al., 2014). For each language, trained or experienced annotators go through each transcribed talk and proceed sentence by sentence, by identifying the type of relation (e.g. explicit, implicit, AltLex), the sense (using PDTB 3.0 sense hierarchy) and the arguments. The annotations are then discussed in multilingual group meetings where all TED-MDB members are physically present, to check annotation consistency. We refer to Zeyrek et al. (2018) for a detailed account of the annotation process.

To populate the lexicon, we retrieve the list of explicit and implicit connectives and the alternative lexicalizations that were marked in the corpus. This data inform the type of DMs that we include. Indeed, deriving the lexicon entries from the corpus annotation work leads us to include categories that are less typical of DMs, as we discuss in section 4.

Furthermore, we conducted a manual contrastive approach between English and Portuguese, based on the parallel Europarl corpus and on the list of English connectives of the PDTB (Mendes and Lejeune, 2016). We located DMs in the English corpus and inspected the Portuguese sentences to identify the corresponding DM. We applied a manual approach with several goals in mind: to procure fully accurate data, to identify potential new senses of the Portuguese connectives, to spot semantic and pragmatic differences between DMs denoting the same sense. The approach is close to the Translation Spotting Technique (Cartoni et al., 2013), although our motivation is not to capture the different meanings of a given connective in the source language but to acquire a diversified set of connectives in Portuguese. The manual identification of connectives based on a contrastive language analysis brings our attention to other lexical strategies that express coherence relations between text spans.

4. Contents of the Lexicon

The lexicon is structured as pairs of DMs/rhetorical senses, so as to cover polysemous markers. The lexicon includes at the moment 252 pairs of DMs/rhetorical senses. A unique numerical identifier is attributed to each DM/rhetorical sense pair. Additionally, there is a Comment feature available to add any observation or open discussion regarding the DM.

Rhetorical sense

We adopt the PDTB 3.0 sense hierarchy with 4 top-level senses (Comparison, Contingency, Expansion, Temporal) and second- and in some cases third-level senses (Webber et al., 2016). For instance, the DM *de modo que* 'so' is labelled as Contingency:Cause:Result, while the DM *da mesma forma que* 'in the same way as' is labelled Comparison:Similarity. DMs have frequently more than one possible rhetorical sense. Working on a lexicon of DMs involves a tension between multiplying the rhetorical senses of a DM or keeping a limited set of what may be considered as the prototypical or core values of the connective. Again, the acquisition method informs the results: the annotators of a discourse treebank will frequently choose different rhetorical senses for a single connective according to the context and this will be reflected in a treebank-driven lexicon. In our case, many of the DMs that are included in the lexicon are acquired from our work on TED-MDB. Here, the method followed the proposal of the PDTB: when the contexts lead to infer an additional sense, the explicit DM is labelled with its prototypical sense and an implicit connective is proposed and annotated with the inferred sense (Rohde et al., 2015). One example of such annotation in the Portuguese section of the TED-MDB Treebank is provided in (1): the explicit coordinate conjunction (underlined) is labelled with the sense Expansion:Conjunction (cf. 1a) and an additional implicit DM (underlined and in parentheses) accounts for the inferred sense Contingency:Cause:Result (cf. 1b).

(1) a. Estas iniciativas criam um ambiente de trabalho mais móvel e reduzem a nossa pegada imobiliária. (TED talk 1927) 'These initiatives create a more mobile work environment and reduce our housing footprint.'

b. Estas iniciativas criam um ambiente de trabalho mais móvel e (portanto) reduzem a nossa pegada imobiliária. 'These initiatives create a more mobile work environment and consequently reduce our housing footprint.'

As a result, the lexicon reflects the decisions taken in the treebank: we describe the intrinsic values of the DM independently of values that may be triggered by adjacency between sentences and by the lexical content of the clauses. For future automatic applications, we aim to combine the information in the lexicon with the data in the treebank related to explicit DMs that have been complemented by an implicit connective to account for inferred senses.

Internal structure of the DM

Two complementary features, inspired by the information in the DiMLex lexicon, describe the internal structure of the DMs. On the one hand, each DM is defined as continuous or discontinuous. Examples of discontinuous DMs are *por um lado… por outro lado* 'on the one hand… on the other hand', *tal como… também* 'just as… so too'. Discontinuous DMs are described as having two orthographic segments.

On the other hand, DMs are described as composed of a single token or as a multiword unit (phrasal). In the case of discontinuous DMs, each orthographic segment is also described in terms of single or phrasal. For instance, the conjunction *logo* 'thus' is a DM with a single token, the conjunction *logo que* 'as soon as' is a phrasal continuous marker, and *tal como… também* 'just as…so too' is a discontinuous DM, where orthographic part 1 is phrasal (*tal como*) and orthographic part 2 is single (*também*).

Type

We adopt a three-category typology: primary connectives, secondary connectives and Alternative Lexicalizations. The distinction between primary and secondary connectives follows the proposal of Rysová and Rysová (2015). Primary connectives are prototypical discourse connectives such as conjunctions, prepositions, adverbs and adverbial phrases. Secondary connectives are other devices that assure cohesion but show a lesser degree of lexicalization than the prototypical discourse connectives. Instances of secondary connectives in the lexicon frequently involve one element that may be replaced, such as deitics:

- *antes disso* 'before that', *da mesma maneira* 'in the same way', *nessa altura* 'at that time', *nesse caso* 'in that case', *nesta perspetiva* 'in this perspective / accordingly', *nessa perspetiva* 'in that perspective / accordingly', *neste sentido* 'in this sense / accordingly', *razão pela qual* 'reason for which', *motivo por que* 'motive for which' .

We also include in the lexicon Alternative Lexicalizations (AltLex), i.e, alternative expressions that denote a cohesive relation, following the PDTB typology (Prasad et al., 2010). What we mark as Alternative Lexicalizations are cases more or less equivalent to a third type in Rysová and Rysová's proposal, labelled 'free connective phrases', that differ from secondary connectives because they carry specific lexical content that restricts their use to a limited set of contexts. Examples of alternative lexicalizations:

- *não deixa de ser verdade que* 'it is nevertheless true that', *isto não significa que* 'but that doesn't mean that', *um dia depois* 'one day later'

We have also encountered borderline cases of intra-sentential discourse relations marked by a main causative verb (Danlos, 2006), such as *provocar* 'to provoke', obrigar 'to force', *reduzir* 'to reduce', which typically establish a causal coherence relation between two nominalizations (Lejeune et al., 2016).

While these alternative lexicalizations were with no doubt required to capture coherence relations in the annotations of texts, it was debated whether or not to include them in the lexicon, since they fall outside the obvious POS categories. We decided to include these expressions because they might prove useful for applications in automatic discourse relation identification and labelling. Their categorization in a specific category (AltLex) makes it possible to isolate and exclude them if required.

Category of the DM

Additional information on the category of the connective is provided in a required field *Category*. For the primary and secondary types, there are 4 categories:

- subordinate conjunction (csu),
- coordinate conjunction (cco),
- preposition (prep),
- and adverb and adverbial phrases (adv).

For the AltLex type, we use the categories above if applicable. In other cases, we give here information about the category of the semantic nucleus of the expression. For instance, *isto não significa que* 'but that doesn't mean that' (arg2-as-denier) is labelled as Category = verb. This allows us to quickly retrieve all verb based alternative expressions that assure coherence relations in texts.

Restrictions on the context

The lexicon provides information on restrictions on the mood of the clause introduced by the DM: we consider indicative as the default value and label as subjunctive otherwise. There is also information on the tense of the clause introduced by the DM: the default is a finite tense and we provide information if otherwise, such as infinitive, inflected infinitive and participle. The last two are illustrated in (2) and (3), respectively (we underline the DM and show the inflected infinitive form in italic).

(2) Apesar de não *terem* sido colegas, a amizade delas durava desde o tempo da Faculdade. (CRPC) 'Although they had not been colleagues at school, their friendship lasted since college time.'

(3) Uma vez *ultrapassada* a "fase de admissibilidade", o SEF emite uma autorização de residência válida por 60 dias e renovável por 30 até ser tomada uma decisão final. (CRPC) 'Once the 'phase of admissibility' is overcome, the SEF issues a green card valid for 60 days and renewable for 30 more days until a final decision is taken.'

Frequent modifiers of the DM, if any, are also indicated in the lexicon, although not consistently. One such case is the frequent presence of the adverb *muito* 'very' before the conjunction *embora* 'although': *muito embora*. These features might be especially important to deal with connectives that share a common rhetorical sense although they do not occur in the same contexts since "connectives are not always interchangeable and therefore cannot be treated as equivalents" (Cartoni et al., 2013).

English near-synonym

We provide one or more English near-synonyms for each DM/sense pair. We choose, when applicable, one of the entries of the DiMLex-en, compiled from data from the PDTB, and provide the unique identifier of the DM in the English lexicon (Stede et al., 2017).

Corpus Example

Finally, we provide for each entry of the lexicon a corpus example and information on the source of the example. Examples originate mostly from: (i) the Reference Corpus of Contemporary Portuguese, available through CQPweb[1] (Généreux et al., 2012); (ii) from the Portuguese subpart of the TED-MDB discourse treebank (in the case of alternative lexicalizations, because we tend to provide examples from a native corpus of Portuguese, namely CRPC, in what concerns primary and secondary connectives); (iii) and from Europarl texts, when they are identified through a contrastive approach with English.

5. Format of the lexicon

The integration of the lexicon into different types of applications requires structured information in a machine-readable format such as XML. But while machine-friendly and extremely rich and hierarchical, XML is certainly less human-friendly than a simple spread sheet that allows the immediate comparison and filtering of the entries. We have adopted a mixed approach for the lexicon of Portuguese: data is entered in a single spread sheet and later converted to an XML scheme compatible

[1] http://alfclul.clul.ul.pt/CQPweb/crpcfg16/

with the DiMLex format. The first row of the excel data sheet makes explicit how the field is later on converted to a structured xml file through a perl script. We follow the main components of DiMLex and consider four top-level main components: *Orthographical*, *Syntactic*, *Semantic*, *Synonym* and *Examples*. There are some differences in the contents of each component due to specificities of each project. In LDM-PT, each row corresponds to an association of DM/category/meaning. So, the same word form will occur in two different rows if it has two different categories or two different meanings. This is handled differently in DiMLex (a single entry aggregates different categories and meanings).

The syntactic component <syn> includes information on type, category, context restrictions (mood and tense) and modifiers of the DM. The semantic component <sem> states the 3-level sense. Finally, there are three additional components: <synonym>, <example> and <comments>.

We illustrate an XML entry of the lexicon in Figure 1.

```
<dmarkers>
 <dmarker word="a fim de que" id="dm1">
      <orth1 type="cont">
          <part1 type="phrasal">a fim de que</part1>
          <part2 type=""></part2>
      </orth1>
      <syn>
          <type>primary connective</type>
          <cat>csu</cat>
          <context>
                <mood>subjunctive</mood>
                <tense></tense>
          </context>
          <modifier1></modifier1>
          <modifier2></modifier2>
        </syn>
      <sem>
          <relationl1>contingency</relationl1>
          <relationl2>purpose</relationl2>
          <relationl3>arg2-as-goal</relationl3>
      </sem>
      <synonym lexicon="dimlex-en" entry-id="22">so
      that</synonym>
      <examples>
          <example1 source="CRPC">Por fim , a
Comissão sugere um sistema de etiquetagem das viaturas
a fim de que o cliente possa fazer uma escolha com
melhor conhecimento de causa. </example1>
          <example2 source=""></example2>
          <example3 source=""></example3>
        </examples>
        <comment></comment>
 </dmarker>
```

Figure 1: Full XML entry of the continuous and phrasal DM *a fim de que* 'so that'

The top-level <dmarker> component includes attributes regarding the word form of the DM and its numerical id. The Orthographical <orth> component (more than one <orth> component can be included to deal with variants such as initial capital letter and contractions) has an attribute type to describe the continuous our discontinuous nature of the DM. Continuous DMs are described in the subcomponent part1 as belonging to the type single or phrasal. We illustrate in Figure 2 the <orth> component of a discontinuous DM: the type of the <orth> component is "discont" and each part (part1 and part2) are labelled as phrasal or single.

```
<dmarkers>
 <dmarker word="tal como…também" id="dm235">
      <orth1 type="discont">
          <part1 type="phrasal">tal como</part1>
          <part2 type="single">também</part2>
      </orth1>
```

Figure 2: <orth> component of the XML entry of the discontinuous DM *tal como ... também* 'just as…so too'

The lexicon was later converted to the DIMLex format to be integrated in the multilingual resource Connective-Lex.info (Stede et al., 2017)[2] through a web app (Dombek, 2017). Due to the different entry structure of LDM-PT and DIMLex, the split-up entries for ambiguous connectives in LDM-PT had to be merged by grouping them, first by word, then by word class. Of the described fields unique to this lexicon, only type was taken over into the DiMLex representation, as a new type attribute for the entry tag, so that it can be displayed by the app. Neither the sense tagset nor the POS tagset had to be converted using mappings. Some used POS tags, e.g. verb, are not specifically represented in the app, but no mapping is necessary for this, as the app automatically represents all unknown tags as 'other'.

6. Applications

Resources with encoded discourse information like LDM-PT have different applications. First of all, they provide data for the annotation of discourse relations in discourse treebanks.

This information can be used directly for manual annotation, in the development of semi-automatic tools (Aleixo and Pardo, 2008), or in fully automatic systems that perform discourse parsing (Pardo and Nunes, 2008; Ziheng et al, 2014; Maziero et al, 2015).

Secondly, they can be integrated in NLP applications dealing with tasks like automatic summarization, information extraction, text generation, machine translation and sentiment analysis (Taboada and Mann, 2006), as well as in the new field of argumentation mining (Peldszus and Stede, 2013).

Finally, linking monolingual lexicons through a pivot English lexicon leads to a multilingual resource and provides data for multilingual applications.

7. Conclusion and future work

We have presented LDM-PT, a new lexicon of DMs for Portuguese. The set of DMs included in the lexicon is based on several sources, ranging from frequency lists extracted from a corpus of contemporary Portuguese, to a multilingual discourse treebank (TED-MDB) and contrastive analysis with English DMs. This accounts for the wide range of syntactic categories that are included in the resource: conjunctions, prepositions, adverbs and adverbial phrases, but also alternative lexicalizations that carry a cohesive function in texts.

[2] http://connective-lex.info

The rich set of features is inspired by both the DiMLex and the LEXCONN lexicons, and covers orthographical information, syntactic category, rhetorical relations, restrictions on the context, examples and an English near-synonym. The latter feature has enabled the linking of LDM-PT in connective-lex.info, a multilingual platform of lexicons of DMs.

The lexicon includes for now 252 pairs of discourse connectives/rhetorical senses. The coverage and sense inventory of the lexicon will be validated in the near future by comparing the set of rhetorical labels for each DM in the lexicon with the TED-MDB corpus, and also with a random selection of contexts from different genres taken from the CRPC corpus.

We plan to enlarge this resource by including pragmatic markers with interactional and modal meaning found in our spoken corpora of Portuguese. Also, our objective is to use the lexicon to automatically pre-annotate DMs in a discourse treebank of Portuguese and to develop automatic tools for discourse parsing.

8. Acknowledgements

This work was partially supported by national funds through FCT - Fundação para a Ciência e a Tecnologia (under the project PEst-OE/LIN/UI0214/2013), and some of its developments were implemented in the scope of the COST Action TextLink – Structuring Discourse in Multilingual Europe[3].

9. Bibliographical References

Cartoni, B., Zufferey, S. and Meyer, T. (2013). Annotating the Meaning of Discourse Connectives by Looking at their Translation: The Translation Spotting Technique, *Dialogue and Discourse* (2013), 68-86.

Cuenca, M. and Marín, M. J. (2009) : Co-occurrence of discourse markers in Catalan and Spanish oral narrative, *Journal of Pragmatics 41* (2009), 899–914.

Danlos, L., (2006) "Discourse Verbs" and Discourse Periphrastic Links. In Sidner, C., Harpur, J., Benz, A., Kühnlein, P. (eds), *Proceedings of the Second Workshop on Constraints in Discourse*, Maynooth, Ireland, 2006. 59–65.

Dombek, F. (2017) *Connective-lex.Info – A Web App for a Multilingual Connective Database*. Bachelor Thesis, Department of Linguistics, University of Potsdam.

Généreux, M., I. Hendrickx, A. Mendes (2012) A Large Portuguese Corpus On-Line : Cleaning and Preprocessing. In: Caseli, H. et al. (eds.) *Computational Processing of the Portuguese Language. Proceedings of the 10th International Conference PROPOR1012*. Berlin, Heidelberg: Springer-Verlag, pp. 113-120.

Halliday, M.A.K. and Hasan, R. (1976). *Cohesion in English*. London, Longman.

Lejeune, P., Mendes, A. and Martins, N. (2016) Some considerations on the use of main verbs to express rhetorical relations. In Degand, L., C. Dér, P. Furkó and B. Webber (eds.) *Conference Handbook of TextLink – Structuring Discourse in Multilingual Europe Second Action Conference*, Budapest, 11-14 April 2016, 89-92.

Lopes, A., Matos, D., Cabarrão, V., Ribeiro, R., Moniz, H., Trancoso, I., Mata, A. I. (2016). Towards Using Machine Translation Techniques to Induce Multilingual Lexica of Discourse Markers, March 2015, http://arxiv.org/abs/1503.0914, accessed 15 January 2016.

Mírovský, J., Synková, P., Rysová, M., Poláková, L. (2016). Designing CzeDLex – A Lexicon of Czech Discourse Connectives, *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation*, Seoul, South Korea, 449-457.

Peldszus, A. and Stede, M. (2013): From argument diagrams to argumentation mining in texts: A survey. *International Journal of Cognitive Informatics and Natural Intelligence* (IJCINI), 7(1):1–31, 2013.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. (2008): The Penn Discourse TreeBank 2.0. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC'08),* 2961-2968.

Prasad, R., Joshi, A., & Webber, B. (2010). Realization of discourse relations by other means: alternative lexicalizations. *Proceedings of the 23rd International Conference on Computational Linguistics: Posters,* pp. 1023-1031. Association for Computational Linguistics.

Prasad, R., Webber, B., & Joshi, A. (2014). Reflections on the Penn Discourse Treebank, comparable corpora, and complementary annotation. *Computational Linguistics*.

Rohde, H. Dickinson, A., Clark, C., Louis, A. Webber, B. (2015). Recovering discourse relations: Varying influence of discourse adverbials *Proceedings of the EMNLP 2015 Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pp. 22–31, Lisboa, Portugal.

Rysová, M. and Rysová, K. (2015). Secondary connectives in the Prague Dependency Treebank, *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, Uppsala, Sweden, August 24–26 2015, 291–299.

Scheffler, T. and Stede, M. (2016). Adding Semantic Relations to a Large-Coverage Connective Lexicon of German. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC'2016),* 1008-1013.

Taboada, M. and Mann, W. C. (2006). Applications of rhetorical structure theory. Discourse Studies 8 (4): 567–588.

Webber, B., Prasad, R., Lee, A., & Joshi, A. (2016). A Discourse-Annotated Corpus of Conjoined VPs. LAW X, p. 22.

Zeyrek, D., A. Mendes, M. Kurfali (2018) Multilingual Extension of PDTB-Style Annotation: The Case of TED Multilingual Discourse Bank. In *Proceedings of the 7th Language Resources and Evaluation Conference* (*LREC'2018*).

10. Language Resource References

Aleixo, P. & Pardo, T. (2008). CSTTool: Uma Ferramenta Semi-Automática para Anotação de Córpus pela Teoria Discursiva Multidocumento CST. NILC-TR-08-03. NILC. Maio de 2008.

Briz, A., Pons Bordería, S. and Portolés, J. (dirs.) (2003). *Diccionario de partículas discursivas del español*, online since 2003, www.dpde.es.

Feltracco, A., Jezek, E., Magnini, B. and Stede, M. (2016) LICO : A Lexicon of Italian Connectives, *Proceedings of the Third Italian Conference on Computational*

[3] www.textlink.metu.edu.tr

Linguistics (CLiC-it 2016), 2016, Napoli, December 5-7, 2016.

Maziero, E., Hirst, G. and Pardo, T. (2015). Adaptation of Discourse Parsing Models for the Portuguese Language. 140-145.

Mendes, A. and Lejeune, P. (2016). LDM-PT. A Portuguese Lexicon of Discourse Markers. In Degand, L., C. Dér, P. Furkó and B. Webber (eds.) *Conference Handbook of TextLink – Structuring Discourse in Multilingual Europe Second Action Conference*, Budapest, 11-14 April 2016, 89-92.

Pardo, T. and Nunes, M. (2008). On the development and evaluation of a Brazilian Portuguese discourse parser. Journal of Theoretical and Applied Computing, vol. 15, pp. 43–64, 2008.

Roze, C., Danlos, L. and Muller, P. (2012): Lexconn: a French lexicon of discourse connectives, *Revue Discours* (2012), http://discours.revues.org/8645.

Stede, M. (2002) DiMLex: A Lexical Approach to Discourse Markers, in A. Lenci – V. Di Tomaso (ed.), *Exploring the Lexicon - Theory and Computation*, Alessandria (Italy), Edizioni dell'Orso, 2002.

Stede, M., Scheffler, T. and Dombek, Felix (2017). Connective-lex.info, Potsdam University. http://connective-lex.info.

Ziheng, L., Hwee, T. N. and Min-Yen, K. (2014). A PDTB-Styled End-to-End Discourse Parser. Natural Language Engineering, 20, pp 151-184. Cambridge University Press.

One Language to rule them all: Modelling Morphological Patterns in a Large Scale Italian Lexicon with SWRL

Fahad Khan[1], Andrea Bellandi[1], Francesca Frontini[2], Monica Monachini[1]
[1]Istituto di Linguistica Computazionale "A. Zampolli" (ILC-CNR) Pisa, Italy
[2]PRAXILING UMR 5267 Univ Paul Valéry Montpellier 3 & CNRS - Montpellier, France
name.surname@ilc.cnr.it, francesca.frontini@univ.montp3.fr

Abstract

We present an application of Semantic Web Technologies to computational lexicography. More precisely we describe the publication of the *morphological layer* of the Italian Parole Simple Clips lexicon (PSC-M) as linked open data. The novelty of our work is in the use of the Semantic Web Rule Language (SWRL) to encode morphological patterns, thereby allowing the automatic derivation of the inflectional variants of the entries in the lexicon. By doing so we make these patterns available in a form that is human readable and that therefore gives a comprehensive morphological description of a large number of Italian words.

Keywords: Morphology, Linked Open Data, Italian Lexicon, SWRL, SQWRL

1. Introduction

The publication of lexical resources as Linked Data (LD) has recently become a important issue in computational lexicography. However in the rush to convert all kinds of lexicons and dictionaries into RDF it is perhaps the case that the technical limitations of this mode of publication, as well as the new potentialities which it offers, are not as thoroughly understood as they ought to be. What, then, are some advantages of using Semantic Web technologies to publish lexicons that might compensate for the loss in performance which is associated with LD with respect to other representation formats? Obviously the fact that LD makes it easier to link together datasets (interoperability), and to make them'open' and publicly available (as with Linked Open Data) is a crucial factor in its favour, but then there are also important technological reasons for publishing datasets as LD. For instance the Semantic Web gives us access to a whole ecosystem of standards, languages, and technologies which we can use to work with and to explore linked data. In this paper we look at one such language, the Semantic Web Rule Language (SWRL), and explore the extent to which it might potentially be able to play a useful role in the publication of lexicographic resources. We do this by detailing the conversion into RDF and publication of the **morphological layer (PSC-M)** of the Parole Simple Clips (PSC) Italian language lexicon. While the publication of this resource will make an important source of Italian morphological data freely and openly available to researchers and the wider public, the novelty of this work is in our use of SWRL to encode morphological patterns, something that allows the automatic derivation of the inflectional variants of the entries in the lexicon. By doing so we also make these patterns available in a form that is human readable and that therefore gives a comprehensive morphological description of a large lexicon's worth of Italian words. Note that we have already presented the first stage of the conversion of PSC-M, that of the nouns, in previous work (Khan et al., 2017). In the current article we will describe the complete conversion of the PSC-M into linked open data and focus on the challenges which arose in converting the other parts of speech in the lexicon.

2. Background

2.1. Why SWRL?

SWRL is, as its name suggests, a rule language[1]. It is based on a subset of Datalog with both unary and binary predicates and is probably the best known attempt at an implementation the 'Rules' layer of the Semantic Web stack. By providing an extension of the Web Ontology Language (OWL) with Horn-like clauses SWRL permits modelers to overcome some of OWL's expressive limitations as a formalism. Although there is a long tradition of using rule languages such as Prolog in computational linguistics, previous work on use of SWRL in this domain seems to be thin on the ground (see (Wilcock, 2007)) – we speculate that this is due in large part to SWRL's own limited expressivity, at least in comparison to most of the other rule languages used in the past, and which makes it inadequate to the task of representing more complex kinds of syntactic and semantic phenomena. And so one of the core aims behind this work was to understand the viability of using SWRL rules in the modeling of at least part of a language, and more precisely to see if SWRL could help us encode part of a medium-to-large scale lexicon. What we needed in order to do this was a resource that provided us with a large number of rules which we could encode using the restricted syntax offered by SWRL.

2.2. Why Parole Simple Clips?

Luckily the authors of the paper had access to just such a resource, namely Parole Simple Clips (PSC), a wide-coverage, multi-layered computational lexicon for Italian that was built up within the framework of three different national and international projects[2]. After studying the composition of the lexicon it became clear to us that the conversion of PSC's morphological (PSC-M) layer into LOD

[1] https://www.w3.org/Submission/SWRL/

[2] For more information about Parole Simple Clips see http://www.ilc.cnr.it/it/content/risorse.

would provide an excellent test case for the use of SWRL[3]. What made PSC-M so attractive in this regard was the fact it featured both extensional and intensional morphological data for each of its lexical entries, and in the latter case this was in the form of representations of morphological patterns.

2.2.1. The Make-Up of PSC-M

In terms of size, PSC-M contains over 53,000 lexical entries and over 380,000 inflected forms. As mentioned above it also contains **inflectional schemes** representing the derivation of inflected forms from the lemma of each lexical entry. Take for instance the lexical entry for the adjective "bello" (Figure 1), PSC-M lists its inflected form "bella" as well as registering the fact that it is a feminine singular adjective. But it also links the entry with a morphological rule that can be used to generate the feminine form (*remove 1 letter from the end of the lemma and add "a"*). While the explicitly stated inflected forms are unique to each lemma (taking homonyms into consideration of course), inflectional rules can apply to more than one lexical entry. This allows us to group lexical entries together in classes based on their morphological behaviour as encoded in their respective morphological patterns. We call such classes **inflectional classes**. Each morphological pattern in PSC-M belongs to a single one of these inflectional classes and each inflectional class is associated with two or more patterns describing the derivation of the morphological variants of each of the lexical entries belonging to the class.

PSC was originally stored as an relational database which made the morphological patterns difficult for human beings to read and which also meant that the patterns weren't immediately machine actionable either. Interestingly the Lexical Markup Framework standard (Francopoulo, 2013) has a morphology module that also allows for the representation of such rules; see for instance the LMF representation of the rule to generate the present first person plural of regular "**are**" verbs below.

```
<TransformSet>
   <Process>
      <feat att="operator" val="remove"/>
      <feat att="string" val="4"/>
   </Process>
   <Process>
      <feat att="operator" val="add"/>
      <feat att="string" val="IAMO"/>
   </Process>
   <GrammaticalFeatures>
      <feat att="morphofeat" val="P1IP"/>
   </GrammaticalFeatures>
</TransformSet>
```

Unfortunately no standard XML-technology exists to derive an inflected form from its lemma using these LMF encoded rules. In contrast to this however, the authors felt that if PSC-M's morphological patterns could be successfully encoded in SWRL then it would offer us the possibility of not only publishing the morphological information in PSC in a human readable format, but of doing so in a way that made these patterns immediately machine actionable using openly available and common standards and technologies. And so it was that after a first successful experiment in encoding the morphological patterns pertaining to the nouns into SWRL, as described in (Khan et al., 2017), we decided to go ahead and encode the rest of the morphological patterns (pertaining to the parts of speech verb and adjective) into SWRL. We describe this in the next section.

[3]The full conversion of PSC into LD is still ongoing. So far, aside from our work on the morphological layer, only part of the semantic layer of PSC has been converted into LOD (Del Gratta et al., 2015; Khan and Frontini, 2014).

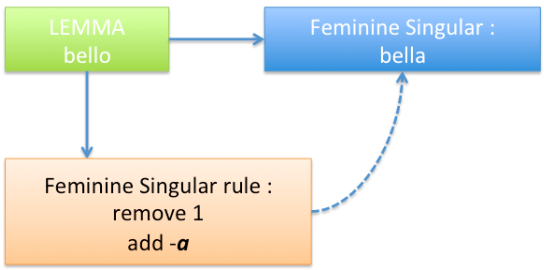


Figure 1: The content of the PSC-M in short

3. Modelling the PSC-M using SWRL

As in our previous experiment on converting nouns we decided that in the interests of efficiency it was better not to convert morphological patterns associated with inflectional classes containing a very small number of lexical entries, i.e., the most irregular entries, into SWRL rules. In such cases, we decided just to enumerate all the variants of an lexical entry without using rules. This meant that with respect to the nouns only the first 30 inflectional classes in the lexicon were converted into SWRL rules, offering a coverage of 96.6%, i.e., almost 97% of the nouns in the lexicon have their morphology captured by SWRL rules. The same strategy was applied to verbs and adjectives and the resulting coverage per part of speech can be seen in Table 1 where as expected the majority of the number of inflectional forms per each verb is very much larger than that for lexical entries for other parts of speech. In summary then, the vast majority of the inflected forms in the lexicon can be generated by SWRL rules and only a small number of irregular lexical entries need to have their morphological variants listed explicitly. Note also that due to the complexity of the verbal inflectional paradigms in Italian, the first 11 verbal inflectional classes have 588 rules that generate 311,543 inflected forms. In the following two subsections we look at how lexical entries and rules were encoded into linked data.

	Nouns	**Adjectives**	**Verbs**
lexical forms in PSC	76416	45723	345320
lexical forms with SWRL	73829	45503	311543
nr. of SWRL rules	79	64	588
nr. of classes	30	16	11
lexical coverage (%)	**96.6**	**99.5**	**90.2**

Table 1: Lexical coverage of part of speech categories.

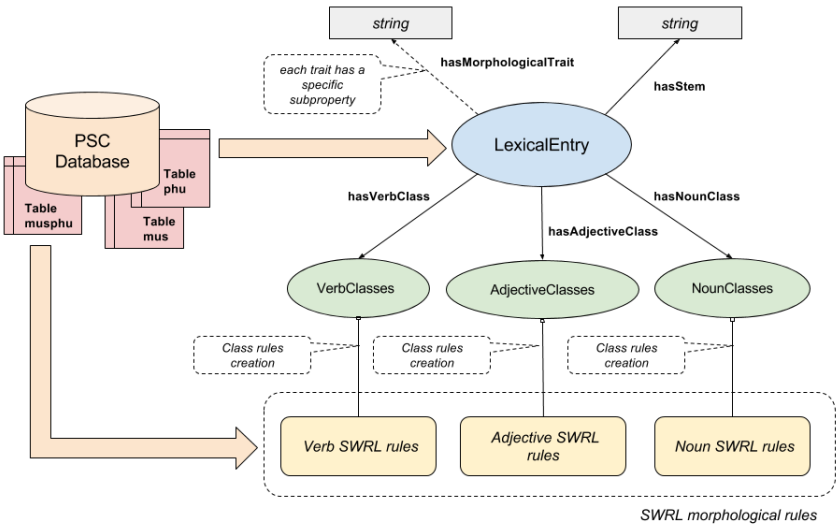


Figure 2: PSC-M to linked data conversion.

3.1. Lexical Entries

In this section we describe the extraction and encoding of the entries in the dataset; in the next section we look at the design and encoding of the rules. The overall lexicon schema is depicted in Figure 2(b).

For each lexical entry in the PSC-M database, amongst those covered by the classes[4], we extract its lemma form, its part of speech and the inflectional class to which it belongs. Each lexical entry is encoded as a member of the lemon[5] class *LexicalEntry*. For the three parts of speech covered by PSC-M we define individuals corresponding to each inflectional class; these are subclasses of the respective classes *NounClass*, *VerbClass*, and *AdjectiveClass*. Each lexical entry is linked to the corresponding inflection class by means of the respective properties *hasNounClass*, *hasVerbClass*, *hasAdjectiveClass*.

```
:VerbClass a owl:Class .
:VClass399 a :VerbClass,
                owl:NamedIndividual .
:hasVerbClass a owl:ObjectProperty ;
   rdfs:domain lemon:LexicalEntry ;
   rdfs:range :VerbClass
```

Although the morphological patterns defined in the original DB version of PSC-M were based on adding and removing suffixes from the lemma form, we decided that encoding this using SWRL rules would make the resulting rules too unwieldy and that the ruleset would end up being inefficient and unusable. Instead we preprocessed the lemmas by removing the suffixes given in the *remove* part of each pattern in the database in order to define a *stem* for each entry. In essence this means that our SWRL rules work by adding string suffixes to stem versions of each lemma. Accordingly we defined the datatype property *hasStem* to associate each lexical entry with its stem. In certain cases it was necessary to define more than one stem and here we use the properties *hasStem1* and *hasStem2*. So for example for the Italian verb *accendere* we have the following.

```
:accendere a lemon:LexicalEntry ;
   hasVerbClass :VClass399 ;
   hasStem1 "accend" ;
   hasStem2 "acce" .
```

PCS-M uses a specific code in order to refer to the morphological variants of a word. For instance the morphological code "S3IP" refers to the singular, third person, indicative, present form of a verb, whereas 'G' refers to the gerund. For each morphological code in PSC-M we created a corresponding datatype property, subproperty of *hasMorphologicalTrait*, in order to relate a lexical entry with its morphological variants represented as strings. The use of strings here was, once again, to simplify the SWRL rules for each inflectional class; instead of directly linking a lexical entry to its variants using a string data property, however, we could have gone via the lemon class *Form* and its property *writtenRep* but this would have made the resulting rules too complex. In the case of *accendere* the rules give us the following:

```
:abbaiare a lemon:LexicalEntry ;
   hasG "accendendo" ;
   hasS3IP "accende" .
```

3.2. The Rules

Once we had identified which inflectional classes we wanted to encode, we created SWRL rules for the corresponding morphological patterns. These rules work by generating new strings from an initial stem and are of the general form:

$$hasStemI(?x, ?y) \wedge hasXClass(?x, \alpha) \wedge stringConcat(?z, ?y, s)$$
$$\rightarrow hasMorphVar(?x, ?z)$$

where $hasStemI$ can either be $hasStem$, $hasStem1$ or $hasStem2$; $hasXC$ represents the appropriate data property for the part of speech to which the rule applies, e.g., $hasVerbClass$; α is the name of a inflectional class;

[4]For those not covered by the classes we extract all the different variant forms from the PSC database.

[5]http://lemon-model.net/

s is a string; $stringConcat$ is a built in property; and $hasMorphVar$ in the head of the rule represents a data property that associates lexical entries with specific morphological variants such as e.g., $hasS3IP$. As the general form shows, the premise of each rule is composed of 3 atoms: the first identifies the stem of an entry, the second its inflectional class and the last concatenates the right suffix for the inflected form to the right stem.In the following we give one of the rules for Class 399, for generating the singular, third person, indicative, present form:

```
hasVerbClass(?x, Class399)
^ hasStem1(?x, ?y)
^ swrlb:stringConcat(?z, ?y, "E")
-> hasS3IP(?x, ?z)
```

Class 399, like all the verbal inflectional classes is associated with around 50 SWRL rules.By running all the rules for class 399 on "accendere" our lexicon is populated as follows:

```
:accendere a lemon:LexicalEntry ;
 hasVerbClass :VClass399 ;
 hasStem1 "accend" ; hasStem2 "acce" ;
 hasF "accendERE" ;
 hasFP_PP "accendENTI" ;hasFP_PR "acceSE" ;
 hasFS_PP "accendENTE" ;hasFS_PR "acceSA" ;
 hasG "accendENDO" ;
 hasMP_PP "accendENTI" ;hasMP_PR "acceSI" ;
 hasMS_PP "accendENTE" ;hasMS_PR "acceSO" ;
 hasP1CI "accendESSIMO" ;hasP1CP "accendIAMO" ;
 hasP1DP "accendEREMMO" ;hasP1IF "accendEREMO" ;
 hasP1II "accendEVAMO" ;hasP1IP "accendIAMO" ;
 hasP1IR "accendEMMO" ;
 hasP2CI "accendESTE" ;hasP2CP "accendIATE" ;
 hasP2DP "accendERESTE" ;hasP2IF "accendERETE" ;
 hasP2II "accendEVATE" ; hasP2IP "accendETE" ;
 hasP2IR "accendESTE" ;hasP2MP "accendETE" ;
 hasP3CI "accendESSERO" ; hasP3CP "accendANO" ;
 hasP3DP "accendEREBBERO" ; hasP3IF "accendERANNO" ;
 hasP3II "accendEVANO" ; hasP3IP "accendONO" ;
 hasP3IR "acceSERO" ;   hasS1CI "accendESSI" ;
 hasS1CP "accendA" ; hasS1DP "accendEREI" ;
 hasS1IF "accendERO'" ; hasS1II "accendEVO" ;
 hasS1IP "accendO" ;hasS1IR "acceSI" ;
 hasS2CI "accendESSI" ; hasS2CP "accendA" ;
 hasS2DP "accendERESTI" ;hasS2IF "accendERAI" ;
 hasS2II "accendEVI" ; hasS2IP "accendI" ;
 hasS2IR "accendESTI" ; hasS2MP "accendI" ;
 hasS3CI "accendESSE" ; hasS3CP "accendA" ;
 hasS3DP "accendEREBBE" ;hasS3IF "accendERA'" ;
 hasS3II "accendEVA" ; hasS3IP "accendE" ;
 hasS3IR "acceSE .
```

4. Evaluation

In order to evaluate our approach and the resulting resource we decided to consider two different aspects of the dataset. On the one hand, it was important, in order to test the effectiveness of using SWRL rules as an integral part of a medium to large sized computational lexicon, to look at time and resource consumption issues. On the other hand, we wanted to check if the rules were able to generate all and only the correct variants for each lexical entry. With regard to the first point, our lexicon currently comes in two varieties: an *empty* variety that, in the case of regular lexical entries (i,e., those entries belonging to one of the inflection classes associated with SWRL rules), does not contain any of the morphological variants associated with the entry, but only indicates the inflection class to which the entry belongs; and a *post-generation* variety that for each entry includes all its morphological variants. We decided to run the rules on the empty version of the lexicon to see how it takes to generate the lexicon. We used a PC with an Intel®Core™i7 @3.4 GHZ with 16GB of RAM. Table 2 shows the generation time for each grammatical category. As the table shows the maximum generation time is 25 seconds.

	Nouns	**Adjectives**	**Verbs**
SWRL generation time (sec)	15.8	6.4	25

Table 2: Time for generating all the inflected forms.

Finally, and as a sort of informal evaluation of the lexicon we ran a series of test SPARQL queries on it, of varying degrees of complexity. For instance the following query returns all the inflected forms of the lemmas (adjectives) that start with "ESP"[6].

```
SELECT ?wr ?p ?infl
WHERE {
  ?le lemon:writtenRep ?wr .
  ?le lexinfo:PartOfSpeech lexinfo:adjective .
  ?le ?p ?infl .
  ?p rdfs:subPropertyOf psc:hasAMorphologicalTrait .
  FILTER (regex(str(?wr),"^esp"))
}
```

We believe these sorts of queries reveal the usefulness of our resource for answering reasonably complicated questions about Italian morphology.

5. Access

We have made a post generation version of the lexicon available, containing both the rules used to generate the lexicon and the axioms that result, as an RDF dump at `http://lari-datasets.ilc.cnr.it/pscMorph#`; a SPARQL endpoint is available at `http://lari-datasets.ilc.cnr.it/pscMorph/queryForm.html`. By the time of the conference itself, Spring 2018, we plan to have released a number of versions of the lexicon and to have developed an interface that includes a description of the different classes, the SPARQL endpoint and a number of example queries. One of the versions of the lexicon which we plan to publish will be a post-generation version in which the morphological data is structured using both the data properties mentioned above along with the linked data morphological vocabulary MMooNN (Klimek, 2017). Another version will contain the rules and the lexicon without the generated variants allowing users to generate them for themselves as and if they require.

6. Conclusion

One of the main aims of the present work has been to study the viability of encoding linguistic information using SWRL in a lexical linked data resource. In this case it seems that the answer is a positive one; SWRL rules allow us to present morphological patterns in both a human readable and machine actionable form — although we will have to wait for user feedback on our resource for a more authoritative confirmation of the former. Our particular case

[6] A number of the queries can be found on the web page of the endpoint.

study was Italian inflectional morphology, but the implications of our work go beyond this limited domain. A similar approach could be applied to similar phenomena in Italian and other languages such as derivational morphology and syntactic pattern transformations. Indeed numerous derivation rules can be extracted from the information contained in PSC. One further advantage of using rules is that we are able to quickly derive inflectional paradigms for new forms by associating them to an existing rule. In the future we plan to provide online facilities to allow users to enter morphological information for new words by associating them with pre-existing inflectional classes.

7. Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731015.

8. Bibliographical References

Del Gratta, R., Frontini, F., Khan, F., and Monachini, M. (2015). Converting the PAROLE SIMPLE CLIPS Lexicon into RDF with lemon. *Semantic web (Print)*, 6:387–392.

Francopoulo, G. (2013). *LMF Lexical Markup Framework*. John Wiley & Sons.

Khan, F. and Frontini, F. (2014). Publishing PAROLE SIMPLE CLIPS as Linguistic Linked Open Data. In *CLIC-IT, la Prima Conferenza di Linguistica Computazionale Italiana*, Pisa (Italy).

Khan, F., Bellandi, A., Frontini, F., and Monachini, M., (2017). *Using SWRL Rules to Model Noun Behaviour in Italian*, pages 134–142. Springer International Publishing, Cham.

Klimek, B. (2017). Proposing an ontolex-mmoon alignment: Towards an interconnection of two linguistic domain models. In *Proceedings of the LDK 2017 Workshops: 1st Workshop on the OntoLex Model (OntoLex-2017), Shared Task on Translation Inference Across Dictionaries & Challenges for Wordnets co-located with 1st Conference on Language, Data and Knowledge (LDK 2017), Galway, Ireland, June 18, 2017.*, pages 68–73.

Wilcock, G. (2007). An owl ontology for hpsg. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 169–172, Stroudsburg, PA, USA. Association for Computational Linguistics.

Metaphor Suggestions based on a Semantic Metaphor Repository

Gerard de Melo
Rutgers University – New Brunswick
Department of Computer Science
Piscataway, NJ, USA
gdm@demelo.org

Abstract

Metaphors are not only remarkably pervasive in both informal and formal text, but reflect fundamental properties of human cognition. This paper presents an algorithmic model that suggests metaphoric means of referring to concepts. For example, given a word such as *government*, the method may propose expressions such as *father*, *nanny*, corresponding to common ways of thinking about the government. To achieve this, the model draws on MetaNet, a manually created repository of conceptual metaphor, in conjunction with lexical resources like FrameNet and WordNet and automated interlinking techniques. These resources are connected and their overall graph structure allows us to propose potential metaphoric means of referring to the given input words, possibly constrained with additional metaphoric seed words, which may be provided as supplementary inputs. The experiments show that this algorithm greatly expands the potential of the original repository for this task by enabling new connections to be drawn.

Keywords: Metaphor, lexical resources, graph structure

1. Introduction

Whenever one says that issues become *clear*, stock markets *go up*, or time is *spent*, language is arguably being used in a non-literal, metaphorical manner, at least with respect to the original senses of the words. Corpus studies have found that metaphorical phenomena are very pervasive even in formal language (Steen et al., 2010; Shutova and Teufel, 2010). Not only is such metaphorical use of language one of the primary means for creative linguistic expression. It has been widely stipulated that our reliance on metaphor is a natural consequence of the way our brains reflect on and reason about the world.

This paper presents a model that can be used to suggest both well-entrenched and novel metaphoric means of referring to a given input word or set of related input words. For example, given a word such as *government*, the method may propose expressions such as *father*, *nanny*, corresponding to ways of thinking about the government.

While metaphor has been studied extensively in linguistics, cognitive science, as well as NLP, the task of automatically suggesting metaphors has received only little attention. Young (1987) relied on simple relational database queries to find related words. Abe et al. (2006) proposed a method that takes a noun and a set of adjectives as input (e.g., *person* and *young*, *innocent*, *fine*) and use corpus topic models to find other nouns with these properties. Veale and Hao (2007) extend this idea to Web-scale knowledge by using the Google search engine to find relevant adjectives describing a noun. Terai and Nakagawa (2010) present an alternative model for this, based on semantic similarity, which considers both adjectives and verbs as relevant noun properties. Approaches of this sort excel at finding novel poetic metaphors, e.g. *hope is like a lightbulb*, as discussed by Terai and Nakagawa (2010). The system discussed in this paper, in contrast, is biased towards finding novel variations of more fundamental conceptual metaphors that shape human thinking.

The model achieves this by constructing a graph to capture relationships between metaphors, words, as well as mental schemas. For this, it draws on the MetaNet respository (Dodge et al., 2015), a manually created database of metaphor. The approach additionally relies on lexical resources such as FrameNet (Baker et al., 1998; Ruppenhofer et al., 2006) and WordNet (Fellbaum, 1998) and automated interlinking techniques. These resources are connected and their overall graph structure allows us to propose potential metaphoric means of referring to the given input words, possibly constrained with additional metaphoric seed words, which may be provided as supplementary inputs.

2. Metaphor and Cognition

Metaphor is often regarded as a process that allows us to think of one thing in terms of another (Lakoff and Johnson, 1980). The following sentences provide examples of **linguistic metaphors**.

(1) Their spirits were *high*.
(2) You lifted me *up* when I was *down*.
(3) That really *raised* their morale.

Although these example sentences involve different metaphorical expressions, it is evident that they share in common the notion that words relating to elevated positions can be invoked in describing an emotional state. It turns out that these three individual instances of linguistic metaphors can be viewed as instantiations of a more general **conceptual metaphor** HAPPY IS UP. Sentence (2) simultaneously also exemplifies the related metaphor SAD IS DOWN, highlighting that even these more general conceptual metaphors can be generalized and related even further to each other.

While metaphor is often used as a creative, if not poetic linguistic device, Lakoff and Johnson (1980) have convincingly argued that metaphor is a more fundamental cognitive process, the primary function of which is in fact understanding. For instance, when speaking of time, most humans conceive of time in terms of a motion along a path.

Linguistic instantiations of this metaphor include sentences like *We have exciting times ahead of us.* or *That happened way back in the 1980s.* Most modern speakers of English also rely on the TIME IS MONEY metaphor, invoking expressions such as *spending time* without any particular conscious realization of this fact. This metaphor is so pervasive that the mere thought about time, for present-day speakers of English, is likely to invoke the metaphor and its entailments (e.g., that time is a limited and valuable resource). Psychological experiments have confirmed that metaphors in natural language covertly influence the way humans reason about things, even when they are not aware of the metaphor (Thibodeau and Boroditsky, 2013).

A metaphor such as TIME IS MONEY thus allows us to make sense of a **target domain** such as TIME in terms of a **source domain** such as MONEY. In conceptual metaphor theory, this is regarded as a directional mapping. Fauconnier and Turner (2008) have argued that the on-line interaction between source and target domain is best thought of as involving a form of conceptual blending of the two domains. Grady et al. (1999) explain that we can think of conventional metaphors, described in terms of source–target mappings, as launching blends, in the sense of providing inputs to and constraints on them. Typically, the source domain is more concrete than target domain. The notion of time is clearly quite abstract, while money has traditionally been more physical, i.e., something you might carry around in your pocket. It is thus argued that metaphor allows the more basic concrete world, e.g., things that can be physically experienced, grasped, or manipulated, to facilitate our understanding of more abstract concepts.

In conceptual metaphor theory, the target and source domains are often thought of as so-called **schemas**. A schema is an established cognitive structure reflecting a particular aspect of the brain?s interactions with the world. One can distinguish the following two types of schemas:

1. Cogs: Lakoff has proposed the notion of cogs to refer to concepts directly grounded in bodily experience. Following Gallese and Lakoff (2005), cogs can be neurally simulated in a secondary area (e.g., the premotor cortex) without active connections to a primary area (e.g., the motor cortex). It is claimed that such simulation can be used for reasoning and that cogs often correspond to the meaning of grammatical constructions, e.g., verb aspect as in *she is about to run* (Narayanan, 1997). Primitive image schemas (e.g., containment, source-path-goal, force dynamics, and orientation schemas) are assumed to be prime examples of cogs ?- see also Dodge and Lakoff (2005).

2. Frames: Frames are taken to include all other concepts, i.e., in particular those that stem from one?s cultural interactions. This is compatible with the notion of frames used in the theory of frame semantics (Fillmore, 1985), which encompasses traditional event representations (Rouces et al., 2015b) but also regards other sorts of entities as being manifested as frames.

As the source domain of a metaphor tends to be more concrete than the target domain, cogs frequently serve as source schemas. For example, the notion of object manipulation (e.g., *grasping*, *holding*) can be applied to target schemas such as the THINKING domain (e.g., *grasping an idea*, *holding views*). However, not all source schemas are cogs.

3. Exploiting the MetaNet Repository

The MetaNet project[1] (Dodge et al., 2015) has been developing a repository of conceptual metaphors that is both human-readable and machine-readable (Hong and Dodge, 2013). The information captured in such a resource is more systematic and formal and thus better-suited for computational processing than previous work on documenting conventionalized metaphors such as the Master Metaphor List (Lakoff and Schwartz, 1991).

The repository[2] directly represents schemas and metaphors. Metaphors are assigned a human-readable label and described in terms of their target and source schemas, as well as entailments, among other things. Schemas can be described in terms of the involved semantic roles, and a description of a conceptual metaphor can explicitly capture the relevant bindings between the roles of the target and source schemas. Additionally, relationships between different schemas and between different conceptual metaphors can be captured. For instance, the ARRIVING schema is connected to more general schemas, all the way to the very abstract MOTION schema, and even further. The TRUST-RELATIONSHIPS ARE BUILDINGS metaphor is a subcase of the more general metaphor RELATIONSHIPS ARE PHYSICAL STRUCTURES.

The repository currently covers four languages (English, Spanish, Russian, and Farsi). Some schemas are linked to related frames defined in the FrameNet project (Baker et al., 1998; Ruppenhofer et al., 2006). However, in practice, these links are often still missing.

4. Metaphor Suggestions

We shall now see how such information about conceptual metaphors can be used in a graph-based framework to suggest linguistic metaphors.

4.1. Overview

The input to the algorithm will typically consist words from the target domain, i.e. the domain that we want to talk about. Our goal is essentially to go from these original words to potential metaphorical expressions from suitable source domains. For instance, we may provide the word *election* as input, and the system will suggest metaphorical words that may be suitable when talking about an election. The outputs could include words such as *headstart*, *race*, *front-runner* from the source domain RACE, as well as words such as *battleground*, *victory*, *allies* from the source domain WAR. Of course, not all proposed words will always be suitable. Given one or more terms from the target domain, the system will produce a ranked list of candidate

[1] https://metanet.icsi.berkeley.edu/metanet/

[2] Available online at https://metaphor.icsi.berkeley.edu/pub/en/.

terms that may serve as metaphorical expressions for the input terms, chosen from suitable source domains automatically. Optionally, one may also provide relevant seed words from the desired source domain to bias the answers towards source domains of interest.

In its standard form, the algorithm focuses on variations and entailments motivated by existing conceptual metaphors rather than entirely poetic uses that do not bear any relation to common metaphoric cognition processes. The MetaNet repository readily provides a substantial number of such metaphors for a non-trivial number of schemas. However, its lexical coverage is limited. Our algorithm thus adopts a graph-based approach that considers not just the MetaNet repository but also additional lexical resources for much greater lexical coverage.

In conceptual metaphor theory, it is assumed that metaphoric understanding may involve a cascade of activation through a network (Hong and Dodge, 2013). The approach presented in this paper involves setting up a large representational graph structure and then using linear constraints to reflect such activation mechanisms. However, it must be noted that while this method does draw on cognitively inspired resources and mechanisms, no assertion is made that this algorithm comes with any degree of plausibility for human cognition. The goal here is merely to produce useful outputs given the inputs.

4.2. Algorithm

The algorithm operates on a directed graph $G = (V, A)$ with a heterogeneous set of nodes V representing words, schemas, metaphors, and other entities. A directed arc in $(u, v) \in A$ reflects a dependency between node relevance scores, with an arc weight w_{uv} determining to what degree the relevance of u entails the relevance of v. In the following, $V_{\mathrm{m}} \subset V$ shall denote the subset of nodes that represent conceptual metaphors, $A_{\mathrm{t}\to\mathrm{m}} \subseteq A$ to denote the subset of arcs that represent arcs from target schemas to conceptual metaphors, and $A_{\mathrm{m}\to\mathrm{s}} \subseteq A$ to denote the subset of arcs that represent arcs from conceptual metaphors to source schemas. More specific details about the graph are provided later in Section 4.3..

For each node $v \in V$ in the graph, there are two variables to capture their relevance scores: t_v reflects the degree of relevance in the target domain, while s_v reflects the degree of relevance in the source domain.

Formally, the input consists of a set of target node constraints C_{T} of the form $(v, t_{\min}, t_{\max})$ and source node constraints C_{S} of the form $(v, s_{\min}, s_{\max})$. Each target node constraint specifies a desired interval $[t_{\min}, t_{\max}]$ for the target domain relevance t_v of node v. Similarly, each source node constraint specifies a desired interval $[s_{\min}, s_{\max}]$ for the source domain relevance s_v of node v.

In the simplest case, one could simply have a single input word w, and constrain the target relevance t_{v_w} for the corresponding node v_w for w to be 1. To do this, one would provide $C_{\mathrm{T}} = \{(v_w, 0, 1)\}$, $C_{\mathrm{S}} = \emptyset$ as inputs to the algorithm.

Given the graph and the inputs, we seek to find a set of values for t_v, s_v based on the following constrained objective.

minimize

$$\sum_{v \in V} (t_v + s_v) + c \sum_{(u,v) \in A} (\sigma_{uv} + \tau_{uv})$$

subject to

$$t_v + \tau_{uv} + \epsilon \geq w_{uv} t_u \qquad \forall\, (u, v) \in A \setminus A_{\mathrm{m}\to\mathrm{s}} \quad (1)$$

$$s_v + \sigma_{uv} + \epsilon \geq w_{uv} s_u \qquad \forall\, (u, v) \in A \setminus A_{\mathrm{t}\to\mathrm{m}} \quad (2)$$

$$s_v \geq t_v \qquad \forall\, v \in V_{\mathrm{m}} \quad (3)$$

$$t_v + s_v \leq 1 \qquad \forall\, v \in V \setminus V_{\mathrm{m}} \quad (4)$$

$$t_v \in [t_{\min}, t_{\max}] \qquad \forall\, (v, t_{\min}, t_{\max}) \in C_{\mathrm{T}} \quad (5)$$

$$s_v \in [s_{\min}, s_{\max}] \qquad \forall\, (v, s_{\min}, s_{\max}) \in C_{\mathrm{S}} \quad (6)$$

$$t_v \in [0, 1] \qquad \forall\, v \in V \quad (7)$$

$$s_v \in [0, 1] \qquad \forall\, v \in V \quad (8)$$

$$\tau_{uv} \geq 0 \qquad \forall\, (u, v) \in A \quad (9)$$

$$\sigma_{uv} \geq 0 \qquad \forall\, (u, v) \in A \quad (10)$$

These inequalities have a natural interpretation. Constraints (1) and (2) consider arcs (u, v) from nodes u to nodes v and their corresponding arc weights w_{uv}. Any arc (u, v) that is not a metaphor-to-source one indicates to what degree v should acquire target domain relevance from u. Additionally, any arc (u, v) that is not a target-to-metaphor one indicates to what degree v should acquire source domain relevance from u. A small constant ϵ, which is set to 0.1, determines an extra loss of relevance that occurs at every hop along an arc in the graph. Slack variables τ_{uv} and σ_{uv} ensure that there is a feasible solution, but are highly discouraged from becoming non-zero, as the constant c is fixed to a very high value $> 2|V|$ in the objective function.

Conceptual metaphor nodes have a special function in this graph. At any conceptual metaphor node $v \in V_{\mathrm{m}}$, source domain relevance is positively tied to target domain relevance. Thus, it is only here that target domain relevance may be converted into source domain relevance.

At all other nodes, t_v and s_v constrain each other such that $t_v + s_v \leq 1$. The intuition here is that words from the target domain, which are used literally, are undesirable as outputs of the algorithm. The algorithm's output should instead consist of metaphorically relevant words from the source domain. In light of this, the algorithm constrains the two variables t_v, s_v for any v with respect to each other. If a word is fully in the target domain, it is not considered as being in the source domain, and vice versa.

The minimization objective ensures that the relevance and slack variables do not grow arbitrarily for no reason. A number of techniques may be used for this optimization process. Fortunately, in our setting, the number of variables is just $O(|A|)$, as nodes without arcs are irrelevant. This is an important difference from linear programming algorithms that need to keep track of pairwise connections between all nodes (de Melo, 2013). Hence, the current implementation relies on graph pruning and barrier optimization using CPLEX.

Upon obtaining an optimal solution, the set of all words with non-zero s_v form the overall output. If V_{t} is the set of all word nodes in V, a categorical distribution with

$p_v = \frac{s_v}{\sum_{v \in V_t} s_v}$ can be used to draw words from this output at random.

4.3. Graph Construction

The graph is constructed using a number of lexical resources. While the MetaNet repository already covers many of the most fundamental metaphors that shape our thinking, it is only extremely sparsely populated in the number of words attached to schemas and lacks rich knowledge about semantic relationships or associations between words. We shall thus draw on several additional sources to construct the graph.

Many of the nodes represent words, which are here taken to include multi-word expressions. Word nodes are identified by their string form and their language. The latter is necessary to distinguish words from different languages with the same string form.

Note that many of the links between nodes will be symmetric bidirectional ones, reflecting, for instance, synonymy or translation relationships. Such links result in two arcs (u, v), (v, u) that are inverses of each other and share the same arc weight $w_{uv} = w_{vu}$. In this case, the algorithm will aim at obtaining $t_v + \epsilon \geq w_{uv} t_u \geq w_{uv}^2 t_v - w_{uv}\epsilon$ to the extent possible. In other words, it will try to keep the two nodes close to each other, aiming at placing t_u in the interval $\left[w_{uv} t_v - \epsilon, \frac{1}{w_{uv}}(t_v + \epsilon)\right]$ and similarly for s_u, s_v.

From the MetaNet repository itself, we can adopt conceptual metaphors along with their links to source and target schemas and their links to related and subcase metaphors. This includes schemas as well as any links to words (lexical units). We can also incorporate links to FrameNet frames, which some schemas include as part of their metadata. All of this is imported in four different languages (English, Spanish, Russian, Farsi).

FrameNet (Baker et al., 1998; Ruppenhofer et al., 2006) is a lexical resource based on the theory of frame semantics (Fillmore, 1985), and thus highly compatible with conceptual structure of the MetaNet repository. From FrameNet, we can incorporate nodes for frames and lexical units and the corresponding links between them. Additionally, we can include the frame hierarchy, i.e., inheritance relationships between frames in both directions. FrameNet frames are useful as conceptual structures even beyond linguistic annotation. The FrameBase project, for instance, relies on them for knowledge representation (Rouces et al., 2015a; Rouces et al., 2016; Rouces et al., 2017).

From WordNet (Fellbaum, 1998), we can obtain nodes for words, word senses, and sense relationships. Links between words and word senses are incorporated, and hyponym/hypernym, similarity, and derivation links between word senses are taken as well, with manually specified edge type-specific weights. In order to increase the coverage of the graph, schemas without lexical units are automatically linked to WordNet synsets, using the first sense heuristic for disambiguation. Previous work has found that WordNet-like resources include certain types of commonsense knowledge that is highly relevant for capturing entailments in the target and source domains, although the coverage still tends to be limited (Lönneker, 2003).

From VerbNet (Schuler, 2005), we can include verb entries and their links to WordNet senses. The associated SemLink resource (Palmer et al., 2014) provides mappings between verb entries and FrameNet frames and lexical units, which are included as well.

Overall, this process yields a rich graph with numerous connections between words, schemas, and other entities. For example, the graph connects schemas from the MetaNet repository with corresponding FrameNet frames in the following ways: 1) by means of explicitly provided links from the repository, 2) by means of indirect connections through other resources (WordNet, VerbNet), 3) by means of indirect connections via shared terms, and 4) various hybrid forms of indirect connections, often based on semantic relations between words.

4.4. Randomization

While the aforementioned process has used static arc weights in the graph, in some settings, it may be advantageous to integrate a measure of chance into the processing. For this, we can treat each original arc weight $\hat{w}_{uv}$ as a mere hyperparameter and draw the actual arc weight wuv using one of the following two schemes.

Option 1 Draw w_{uv} from $\mathcal{U}(0, \hat{w}_{uv})$ for $(u, v) \in A$: Drawing arc weights from a uniform distribution ensures that the ranking of source relevance scores is perturbed. Hence, one can repeatedly obtain different highest-rated source domain words among those that the original unperturbed graph would provide.

Option 2 Draw $w_{uv} from \mathcal{N}(\hat{w}_{uv}, \sigma^2)$ for arbitrary $(u, v) \in A = V \times V$ and accept if larger than some threshold $w_{\min} \geq 0$: In this alternative scheme, one instead draws arcs using a normal distribution, allowing even previously non-existent arcs to be created with a non-zero probability. Note that the threshold $w_{\min}$ is needed to ensure the non-negativity of arc weights and in practice also to avoid a large quadratic number of arcs.

5. Results

This section describes initial experiments and statistics about the system.

5.1. Graph Creation

The input graph construction was based on June 30, 2013 dumps of the MetaNet repository and of FrameNet, additionally relying on WordNet 3.0, VerbNet 3.2, and SemLink 1.2.2c.

Metaphors	1,125
— English	613
— Spanish	373
— Farsi	82
— Russian	57
Schemas	1,372
— Source	656
— Target	571

Table 1: Input MetaNet Repository

Table 1 provides statistics about the input repository. Note that being a source or target schema is not an inherent property of a schema, but just refers to its involvement in metaphors described in the repository.

Table 2 shows the number of words immediately connected to schemas. It first lists the overall number, and then provides the breakdown by schemas that serve as target schema for some metaphor and schemas that serve as source schema for some metaphor. Clearly, only few terms are activated if one relies on the MetaNet repository in its original form. Once links to FrameNet frames are included, the coverage increases greatly. Adding VerbNet and SemLink does not increase the coverage in a meaningful way, but the main benefit of including these resources is that they serve as segue to WordNet synsets due to the incorporated links. The additional automatically predicted WordNet mappings lead to significant further increases in coverage. Table 2 provides the counts for the overall graph.

These numbers, however, only reflect those words that are somewhat unambiguously connected to a schema via mappings. The power of the approach of this paper lies in the fact that a dense network of semantic or commonsense links in the input graph allows the algorithm to cast a much wider net of possible words. For this, we can rely on the links between conceptual metaphors, links between schemas, links between frames, and WordNet's semantic relations, which overall result in a graph with over 700,000 directed arcs.

Table 3 shows the number of unique words that are connected to a schema at different maximum depth levels (max. number of hops in graph). Thus, with the extended graph, the system is able to select from a much larger pool of candidate words when making suggestions.

5.2. Algorithm Outputs

For instance, running the system using just the original repository for the target word *anger* (with target relevance score 1.0), the algorithm does not find any relevant source domain words. However, using the final graph, it can find several relevant metaphors, including ANGER IS FIRE, ANGER IS HEAT, and ANGER IS INSANITY. The top-ranked output words are *mad*, *steam*, *crazy*, *simmer*, *blow off*, *boil*, *warm*, *hot*, *stew*, which for the most part can quite well be used to describe anger metaphorically. In addition, many hundreds of other candidate words are returned. In lower ranks, one finds words such as *bake*, *kick*, *hammer*, *poison*, *smash*, *microwave*.

At the input side, instead of *anger*, we can also enter alternative target domain words such as *enrage*, *fury*, and so on, and the algorithm still finds relevant source domain words.

From an accuracy perspective, since the output results from graph links that overwhelmingly have been manually created, the presented words are clearly connected to the relevant domains. Still, the method simply provides relevant source domain words in a rather open-ended manner, and some will obviously be suitable, while in other cases it may still be challenging for writers to come up with a suitable way of employing the suggested terms in a sentence such that the proposed metaphorical interpretation comes to bear. Many suggestions may thus prove unsuitable. To address this, among the output words, one could apply additional filtering to select words that highlight one particular property of the target domain using the technique suggested by Veale and Hao (2007). This involves retaining from the set of proposed source domain words only those that satisfy the constraint of having significant corpus or Web frequency occurrence counts for patterns such as for instance *as* ⟨*property*⟩ *a* ⟨*word*⟩ (e.g., *as innocent as a child*) or ⟨*property*⟩ ⟨*word*⟩ (e.g., *innocent child*).

5.3. Cross-Lingual Applicability

In the above example, despite the use of the English input word *anger*, the algorithm also finds pertinent words in other languages such as the Spanish *calentar* and *arrojar*. This is possible because the MetaNet data includes cross-lingual connections of frames across languages. It is trivial to extend this even further by incorporating further multilingual terms for FrameNet frames (Čulo and de Melo, 2012) and WordNet synsets (de Melo and Weikum, 2009; de Melo and Weikum, 2014), or on multilingual word vectors (de Melo, 2015).

Obviously, there are also important caveats here. The MetaNet database currently described conventional metaphor in four languages, and any cross-lingual connection that the algorithm emits will need to be made via conventional metaphor links in one of those languages. In the literature, there have been studies about how metaphors compare across language boundaries (Gutiérrez et al., 2016). Clearly, many metaphors are highly language-specific, and thus when the algorithm is applied cross-lingually, some of the emitted output candidates are likely to be unsuitable.

However, in many cases, they turn out to be appropriate. For one, this may stem from similarities in metaphorical language use across related languages. For instance, in many Western languages, the word *transparency*, which in its original sense refers to the property of allowing the transmission of light through an object, is also used to refer to public evaluability and accountability. Additionally, this may also stem from a broadly shared experiential basis. For example, the connection between anger and heat derives from human biology. Empirically, Tsvetkov et al. (2014) found that they were able to apply models trained to detect English linguistic metaphors also to the task of detecting linguistic metaphors in other languages, some of which are not phylogenetically close (specifically, they considered Spanish, Farsi, and Russian).

6. Related Work

6.1. Metaphor Detection

Numerous papers have studied the task of automatically identifying metaphoric expressions in text. Many systems aim at achieving this by detecting violations of selectional preference restrictions (Fass, 1991; Shutova et al., 2010). For instance, the verb *to kill* usually applies to living beings, so when it is found in contexts such as *my process got killed*, it is quite likely that the word is being used metaphorically. Some approaches have additionally relied on lexical resources such as FrameNet (Gedigian et al., 2006) and HowNet (Tang et al., 2010) to increase the qual-

	Schemas	Target Schemas	Source Schemas
Original Repository	803	258	513
+ Direct FrameNet Links	5,349	2,283	2,776
+ VerbNet/SemLink	5,368	2,290	2,783
+ WordNet	6,302	2,626	2,849
+ WordNet mappings	6,789	3,304	3,497

Table 2: Word–Schema Connections, where subsequent rows show the counts as additional resources are added (including all previously mentioned ones).

Depth	1	2	3	4	5	6	7	8
Original Repository	742	742	742	742	742	742	742	742
Final Graph	742	1,202	9,024	29,246	62,411	94,065	117,372	125,038

Table 3: Words with Indirect Schema Connections.

ity. However, many such systems are brittle as they rely on rather small amounts of manually provided data.

The approach by Shutova et al. (2015) draws on a massive collection of images and videos with associated tags to draw inferences about which predicate-argument pairs are more likely to be concrete. The underlying assumption is that such multimodal data is more closely grounded in the real world and hence linguistic descriptions are more likely to be concrete and literal (e.g., *cutting hair*) as opposed to more metaphorical (e.g., *cutting costs*), which is often predominant in newswire text.

Only few systems have attempted to go beyond identifying individual linguistic metaphors towards recognizing more general conventionalized conceptual metaphors. The CorMet system (Mason, 2004) attempts to automatically infer metaphor mappings from a corpus by studying systematic differences in verb selectional preferences between domains. Such techniques could be used to extend the number of metaphors in our graph.

6.2. Metaphor Analysis

The Metaphor Magnet system (Veale and Li, 2012) addresses the complementary task of metaphor interpretation, providing a list of attributes that explain what qualities a given source domain shares with a target domain that lend the metaphor its strength. The system takes a target and source domain as input (e.g., LOVE IS A DRUG), and then uses Web corpus frequencies to highlight salient features of the source domain that are shared with the target domain. For instance, for the given example, it highlights *healing*, *satisfying*, and *intoxicating*, providing possible interpretations of how the target concept LOVE is modified when metaphors from the DRUG source domain are invoked.

Shutova et al. (2012) present an unsupervised approach for finding literal paraphrases for a given metaphorical expression. Paraphrases are identified using distributional semantics captured in a vector space model. Selectional preference statistics are then used to select literal expressions among the retrieved paraphrases.

6.3. Metaphor Suggestions

In contrast, the task of automatically suggesting metaphors has received only little attention. Young (1987) uses simple relational database queries to find related words. Abe et al. (2006) take a noun and a set of adjectives as input (e.g., *person* and *young*, *innocent*, *fine*). They then use corpus topic models to find other nouns with these properties. In their experiments, their system mainly emits related words such as *grandchild*, but their intuition is that such a framework could also emit more metaphorical ones such as *puppy*. Veale and Hao (2007) extend this idea to Web-scale knowledge by relying on the Google search engine to find relevant adjectives describing a noun. Terai and Nakagawa (2010) present an alternative model for this same task, based on semantic similarity, which considers both adjectives and verbs as relevant noun properties.

Approaches of this sort excel at finding novel poetic metaphors, e.g., *hope is like a lightbulb* as discussed by Terai and Nakagawa (2010). The system proposed in this paper, in contrast, both in its algorithm and in the kinds of resources we draw on, is biased towards finding variations of fundamental conceptual metaphors that shape human thinking.

7. Conclusion

This paper presents a framework for metaphor suggestion. The approach is centered on the MetaNet semantic repository of conceptual metaphors, which captures various cognitive phenomena, including the connections between target and source domains in common conceptual metaphors.

The approach in this paper involves connecting this information with several other lexical resources in a graph and then optimizing a constrained objective to determine relevance scores for potential source domain words. The algorithm can flexibly integrate additional information sources into its graph and incorporate pre-existing information about relevant source domain words as additional constraints.

The results show that this graph-based approach has a significantly higher coverage than the original repository, to the extent that even words in completely unrelated languages can be processed cross-lingually. Overall, this paves the way for novel applications that use language in a more creative, flexible way.

Acknowledgements

This research was supported in part by the DAAD and by the DARPA SocialSim program.

8. Bibliographical References

Abe, K., Sakamoto, K., and Nakagawa, M. (2006). A computational model of metaphor generation process. In *Proceedings of the 28th Annual Meeting of the Cognitive Science Society*, page 937?942.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet Project. ICCL '98, pages 86–90.

de Melo, G. and Weikum, G. (2009). Towards a universal wordnet by learning from combined evidence. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM 2009)*, pages 513–522, New York, NY, USA. ACM.

de Melo, G. and Weikum, G. (2014). Taxonomic data integration from multilingual Wikipedia editions. *Knowledge and Information Systems*, 39(1):1–39, April.

de Melo, G. (2013). Not quite the same: Identity constraints for the Web of Linked Data. In Marie desJardins et al., editors, *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI 2013)*, pages 1092–1098, Menlo Park, CA, USA. AAAI Press.

de Melo, G. (2015). Wiktionary-based word embeddings. In *Proceedings of MT Summit XV*.

Dodge, E. and Lakoff, G. (2005). Image schemas: From linguistic analysis to neural grounding. In Beate Hampe et al., editors, *From Perception to Meaning: Image Schemas in Cognitive Linguistics*, pages 57–91.

Dodge, E., Hong, J., and Stickles, E. (2015). Metanet: Deep semantic automatic metaphor analysis. In *Proceedings of the Third Workshop on Metaphor in NLP*, pages 40–49, Denver, Colorado, June. Association for Computational Linguistics.

Fass, D. (1991). Met*: A method for discriminating metonymy and metaphor by computer. *Comput. Linguist.*, 17(1):49–90, March.

Fauconnier, G. and Turner, M., (2008). *Rethinking Metaphor*, pages 53–66. Cambridge University Press, September.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Fillmore, C. J. (1985). Frames and the semantics of understanding. *Quaderni di Semantica*, 6(2):222–254.

Gallese, V. and Lakoff, G. (2005). The brain's concepts: The role of the sensory-motor system in conceptual knowledge. *Cognitive Neuropsychology*, 22(3-4):455–479, May.

Gedigian, M., Bryant, J., Narayanan, S., and Ciric, B. (2006). Catching metaphors. In *Proceedings of the Third Workshop on Scalable Natural Language Understanding*, ScaNaLU '06, pages 41–48, Stroudsburg, PA, USA. Association for Computational Linguistics.

Grady, J., Oakley, T., and Coulson, S. (1999). Blending and metaphor. In Raymond W. Gibbs Jr. et al., editors, *Metaphor in Cognitive Linguistics: Selected papers from the 5th International Cognitive Linguistics Conference*, Current Issues in Linguistic Theory 175, page 101ff.

Gutiérrez, E. D., Shutova, E., Lichtenstein, P., de Melo, G., and Gilardi, L. (2016). Detecting cross-cultural differences using a multilingual topic model. *Transactions of the Association for Computational Linguistics (TACL)*, 4:47–60.

Hong, Jisup, E. S. and Dodge, E. (2013). The MetaNet metaphor repository: Formalized representation and analysis of conceptual metaphor networks. In *Proceedings of the 12th International Cognitive Linguistics Conference*.

Lakoff, G. and Johnson, M. (1980). *Metaphors we Live by*. University of Chicago Press, Chicago.

Lakoff, George, J. E. and Schwartz, A. (1991). The master metaphor list (draft 2nd edition).

Lönneker, B. (2003). Is there a way to represent metaphors in wordnets? Insights from the Hamburg metaphor database. In *Proceedings of the ACL 2003 Workshop on the Lexicon and Figurative Language*, pages 18?–27. Association for Computational Linguistics.

Mason, Z. J. (2004). Cormet: A computational, corpus-based conventional metaphor extraction system. *Computational Linguistics*, 30:23–44.

Narayanan, S. S. (1997). *KARMA: Knowledge-based Active Representations for Metaphor and Aspect*. University of California, Berkeley.

Palmer, M., Bonial, C., and McCarthy, D. (2014). Semlink+: Framenet, verbnet and event ontologies. In *Proceedings of Frame Semantics in NLP: A Workshop in Honor of Chuck Fillmore (1929-2014)*, pages 13–17, Baltimore, MD, USA, June. Association for Computational Linguistics.

Rouces, J., de Melo, G., and Hose, K. (2015a). FrameBase: Representing n-ary relations using semantic frames. In *Proceedings of ESWC 2015*, pages 505–521.

Rouces, J., de Melo, G., and Hose, K. (2015b). Representing specialized events with FrameBase. In *Proceedings of the 4th International Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2015) at ESWC 2015*.

Rouces, J., de Melo, G., and Hose, K. (2016). Heuristics for connecting heterogeneous knowledge via FrameBase. In *Proceedings of ESWC 2016*, Lecture Notes in Computer Science. Springer.

Rouces, J., de Melo, G., and Hose, K. (2017). FrameBase: Enabling integration of heterogeneous knowledge. *Semantic Web*, 8(6):817–850, August.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R., and Scheffczyk, J. (2006). *FrameNet II: Extended Theory and Practice*. International Computer Science Institute, Berkeley, California. Distributed with the FrameNet data.

Schuler, K. K. (2005). *Verbnet: A Broad-coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, Philadelphia, PA, USA. AAI3179808.

Shutova, E. and Teufel, S. (2010). Metaphor corpus annotated for source - target domain mappings. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta,

Malta, may. European Language Resources Association (ELRA).

Shutova, E., Sun, L., and Korhonen, A. (2010). Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1002–1010, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shutova, E., van de Cruys, T., and Korhonen, A. (2012). Unsupervised metaphor paraphrasing using a vector space model. In *Proceedings of COLING 2012: Posters*, pages 1121–1130, Mumbai, India, December. The COLING 2012 Organizing Committee.

Shutova, E., Tandon, N., and de Melo, G. (2015). Perceptually grounded selectional preferences. In *Proceedings of ACL 2015*, pages 950–960.

Steen, G. J., Dorst, A. G., Herrmann, J. B., Kaal, A., Krennmayr, T., and Pasma, T. (2010). *A Method for Linguistic Metaphor Identification*. John Benjamins Publishing.

Tang, X., Qu, W., Chen, X., and Yu, S. (2010). Beyond selectional preference: Metaphor recognition with semantic relation patterns. *Int. J. of Asian Lang. Proc.*, 20(4):141–156.

Terai, A. and Nakagawa, M. (2010). A computational system of metaphor generation with evaluation mechanism. In *Proceedings of the 20th International Conference on Artificial Neural Networks: Part II*, ICANN'10, pages 142–147, Berlin, Heidelberg. Springer-Verlag.

Thibodeau, P. H. and Boroditsky, L. (2013). Natural language metaphors covertly influence reasoning. *PLOS ONE*, 8(1):1–7, 01.

Tsvetkov, Y., Boytsov, L., Gershman, A., Nyberg, E., and Dyer, C. (2014). Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 248–258, Baltimore, Maryland, June. Association for Computational Linguistics.

Čulo, O. and de Melo, G. (2012). Source-Path-Goal: Investigating the cross-linguistic potential of frame-semantic text analysis. *it - Information Technology*, 54.

Veale, T. and Hao, Y. (2007). Comprehending and generating apt metaphors: a web-driven, case- based approach to figurative language. In *Proceedings of the 22nd national conference on Artificial intelligence (AAAI 2007) – Volume 2*, page 1471?14762. AAAI Press.

Veale, T. and Li, G. (2012). Specifying viewpoint and information need with affective metaphors: A system demonstration of the metaphor magnet web app/service. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 7–12, Stroudsburg, PA, USA. Association for Computational Linguistics.

Young, L. F. (1987). The metaphor machine: A database method for creativity support. *Decision Support Systems*, 3(4):309–317.

The Linguistic Category Model in Polish (LCM-PL)

Aleksander Wawer and Justyna Sarzyńska
Institute of Computer Science, Polish Academy of Sciences, Jana Kazimierza 5, 01-248 Warszawa
Institute of Psychology, Polish Academy of Sciences, Jaracza 1, 00-378 Warszawa
Poland
axw@ipipan.waw.pl, jsarzynska@psych.pan.pl

Abstract

This article describes the first public release of Linguistic Category Model (LCM) dictionary for the Polish language (LCM-PL). It is used for verb categorization in terms of their abstractness and applied in many research scenarios, mostly in psychology. The dictionary consists of three distinctive parts: (1) sense-level manual annotation, (2) lexeme-level manual annotation, (3) lexeme-level automated annotation. The part (1) is of high quality yet the most expensive to obtain, therefore we complement it with options (2) and (3) to generate LCM labels for all verbs in Polish. Our dictionary is freely available for use and integrated with Słowosieć 3.0 (the Polish WordNet). Its quality will improve: we'll add more manually annotated senses and increase the quality of automated annotations.

Keywords: Linguistic Category Model, LCM, LCM-PL, Polish

1. Linguistic Category Model (LCM)

The LCM typology is a well-established tool to measure language abstraction, applicable for multiple problems as those listed in Section 1.1.. Its core idea is the categorization of verbs into classes reflecting their abstraction.

The most general, top level distinction of the Linguistic Category Model is the one between state verbs (SV) and action verbs. As LCM authors put it, *state verbs (SV) refer to mental and emotional states or changes therein. SVs refer to either a cognitive (to think, to understand, etc.) or an affective state (to hate, to admire, etc.)*. This verb category is the most abstract one and also present in Levin's typology.

The other more concrete type of verbs in the LCM are action verbs. This type is always instantiated as one of its two sub-types, descriptive and interpretative action verbs (DAV and IAV) that all refer to specific actions (e.g., to hit, to help, to gossip, etc.) with a clearly defined beginning and end. SVs,in contrast, represent enduring states that don't have a clearly defined beginning and end.

The distinction between DAVs and IAVs is based on double criteria. The first states that DAVs have at least one physically invariant feature (eg. to kick - leg, to kiss - mouth), whereas IAVs do not (therefore, are more abstract than DAVs). The second criterion, sentiment, states that IAVs have a pronounced evaluative component (e.g., positive IAVs such as to help, to encourage vs. negative IAVs such as to cheat, to bully), whereas DAVs do not (e.g., to phone, to talk). Descriptive action verbs(DAVs) are neutral in themselves (e.g. to push) but can gain an evaluative aspect dependent on the context (to push someone in front of a bus vs. to push someone away from an approaching bus). In practice, the criteria sometimes overlap. Some verbs have physical invariants but also have clear evaluative orientation. For instance, "to cry" always involves tears (an invariant physical feature), but carries negative sentiment.

1.1. Applications

Years of research has yielded considerable evidence that language abstraction is related to many psychological phenomena such as intergroup bias, stereotypes, expectancy bias or even human personality. LCM has proven to be an adequate tool for evaluating such phenomena. Wigboldus Semin and Spears (2000) (Wigboldus et al., 2000) used the LCM in their study which proved that describers use more abstract language for expectancy-consistent behaviors. This effect of expectancies is manifested in linguistic intergroup bias (LIB) wherein people encode and communicate desirable in-group and undesirable out-group behaviors more abstractly than undesirable in-group and desirable out-group behaviors (Maass et al., 1989). The same study found that abstract versus concrete communication play an important role in the perpetuation of stereotypes. In addition to expectancies, describers' goals such as self-presentational goals (Rubini and Sigall, 2002), the desire to compete or co-operate (de Montes et al., 2003) or willingness to protect one's group from threat (Maass et al., 1996) may affect one's level of language abstraction. The LCM was also used in studies demonstrating that language abstraction conveys information both about the person whose behavior is being described and also the describers themselves. Douglas and Sutton (2006) (Douglas and Sutton, 2006) asked participants to view a series of cartoons, each depicting a person performing a behavior deemed positive or negative as well as to reading a description of the behavior . They were then tested whether participants' judgments of describers' communication goals were affected by language abstraction. Participants were asked to rate the likelihood that the describer wanted to create a positive, negative and unbiased impression of the actor. The results show that describers who use relatively abstract language to describe others' behaviors are perceived to have biased attitudes and motives compared with describers who use more concrete language. Even personality differences may manifest themselves in language use. In a study conducted by Beukeboom, Tanis and Vermeulen (2012) (Beukeboom et al., 2013) participants' spontaneous verbal utterances in face-to face interactions were analyzed for language abstraction by applying the Linguistic Category Model. Results showed significant positive correlations between extraversion and language abstraction. The findings suggest that

the verbal style of extraverts is characterized by a higher level of abstract interpretation, whereas introverts tend to stick to concrete facts.

1.2. Previous Work in Polish

The first experiments with automated classification of Polish verbs according to the Linguistic Category Model were reported in (Rogozinska and Wawer, 2013). The research focused on a small set of 1170 verbs, translated from the English General Inquirer (Stone et al., 1966) `http://www.wjh.harvard.edu/~inquirer/` dictionary into Polish. The paper reported high level of agreement of LCM tags between English verbs and their Polish translations, as Kappa scores ranged from 0.83 to 0.87 depending on person (experiment involved two linguists).

However, translating verbs from the General Inquirer dictionary into Polish and copying their LCM labels, was not a satisfactory method to obtain a complete LCM dictionary for Polish due to poor coverage. As mentioned earlier, the 1170 verbs yielded by this method still missed about 90 percent of verbs present in the Polish WordNet (Słowosieć). Therefore, the goal of the experiments in (Rogozinska and Wawer, 2013) was to design and test automated methods of recognizing LCM classes from a sample of verb occurrences from the National Corpus of Polish. Authors applied machine learning to predict LCM class of a verb, based on various features designed using WordNet that explored mostly hyperonymy of nouns immediately following each verb, assuming such nouns as verb's arguments.

The results turned out promising in the sense of exceeding baselines, but due to low precision in recognizing DAV verbs (depending on the setup, precision was at 0.38 and 0.5) the method could not be assumed as satisfactory for use as the main source for LCM labels for the final dictionary.

2. The Polish LCM (LCM-PL): Methodology

2.1. Introduction

The efforts described in this paper are aimed to create the LCM dictionary with following goals: high quality, wide coverage and integration with the Polish WordNet. In this section we discuss these points in more detail.

2.2. Ensuring Quality

. We put significant efforts to achieve high quality LCM labeling. In particular, this rules out purely automatic verb labeling, at least when applied as reported in (Rogozinska and Wawer, 2013) due to low precision.

Ideally, we would like all our LCM labeling to be human-made, possibly by two independent annotators and a third (gold) one for resolving conflicts. However, due to budgetary constraints, this is currently not feasible and we manually annotated only a subset of Polish verbs, selecting the most frequently used verb senses. Also, at the moment our human annotations are single, not double.

2.3. Wide Coverage

. Contrary to previously described experimental works in Polish, we aimed for an LCM dictionary including as many word forms as possible, providing high quality for those most heavily used.

We used the frequency list of word lemmas (base forms) obtained from `http://nlp.pwr.wroc.pl/en/tools-and-resources/resources/frequency-list`. Using that list, we assigned frequencies to verb list found in Słowosieć 3.0.

2.4. Integration with Słowosieć (the Polish WordNet)

. For multiple reasons we decided to integrate our dictionary with the popular existing electronic dictionary (Piasecki et al., 2009). The most recent version 3.0 contains 32448 lexical entries for the verb part-of-speech ("czasownik").

Allowing LCM labels to be attached to WordNet's senses opens up interesting possibilities for high precision LCM labeling. From our study in Section 3. it follows that multiple senses of the same verb might, and indeed often have, different LCM tags. This is also the case for English language LCM annotation in the General Inquirer Dictionary (Stone et al., 1966). Therefore, aiming for precision requires providing sense-level rather than lexeme-level LCM tags.

Attaching LCM tags to WordNet senses is a difficult task. Słowosieć (The Polish WordNet) suffers from notoriously high number of senses, difficult to distinguish even for a professional lexicographer. It's often not easy to image the actual uses of some of the more rare senses.

Fortunately, the creators of Słowosieć provide a tool for automated word sense disambiguation. It takes Polish language texts as input and returns their senses as they are recognized in actual sentences (Kędzia et al., 2015).

Currently, our LCM dictionary contains manual annotation for 10000 verb senses, taken from the latest Słowosieć 3.0. We plan to gradually increase this number, prioritizing verb annotation by their frequency.

Unfortunately, domain of verbs (a property of verbs annotated in Słowosieć 3.0) is not usable for LCM. It seems that „cst" category consist only of verbs connected with enduring state or change of state and execlude verbs which refer to emotional or mental states of a subjective nature – the core of SV verbs according to the LCM Manual.

2.5. Annotation Guidelines

In the spirit of the English language annotation in the General Inquirer, where LCM tags were attached to word senses rather than words (lexical entries), we followed the same approach. Our sense inventory for verbs is based on Słowosieć 3.0 (Piasecki et al., 2009).

We provided annotators with the most recent, unpublished version of the LCM annotator guidelines (Schmid et al., 2017), obtained directly from prof. Klaus Fiedler, the author of LCM (Semin and Fiedler, 1988).

To annotate verb senses with LCM tags, we presented annotators all WordNet synsets of each verb. For each synset, we listed glosses and all synonyms (lexical entries) of that synset so that annotators could distinguish their meaning. We asked to annotate each sense with an appropriate LCM tag.

LCM	lexical id	synset id	domain	synsets / gloss
SV	81612	56818	state	sharing
DAV	89828	63657	ownership	share sth.
IAV	89829	63654	social life	divide, separate
DAV	89826	63655	change	divide
DAV	89827	56841	ownership	separate
DAV	81339	56584	thinking	Determine the quotient of two numbers

Table 1: LCM tags for the senses of Polish verb *dzielić* (eng. to divide)

LCM	lexical id	synset id	domain	synsets / gloss
IAV	85002	69644	social life	get lost
IAV	85000	55371	thinking	disappear, blur
IAV	85001	59688	ownership	fade away
SV	84630	69641	state	fade away
IAV	11550	4361	competition	die, lose life
DAV	22108	64281	change	break-up, perish, lose

Table 2: LCM tags for the senses of Polish verb *zginąć* (eng. to die)

2.6. Annotator Agreement in LCM

The lexeme-level agreement between annotators was reported in (Rogozinska and Wawer, 2013). The authors measured Cohen Kappa agreement between two Polish annotators, and also between LCM labels by each of the annotators and LCM labels of English equivalents of Polish verbs. The experiment has been performed on 1170 verbs taken from the General Inquirer `http://www.wjh.harvard.edu/~inquirer/` dictionary and translated into Polish. It adresses not only the issue of differences between two persons annotating the same verbs in one language, but also the difference between LCM labels of the same verb, translated into another language.

The results indicated that the Kappa between two Polish annotators was at 0.78, while the Kappa between their Polish LCM verb annotations and English LCM equivalents was between 0.83 and 0.87. As (Rogozinska and Wawer, 2013) concluded, the agreement is reasonably high, but it is also clear that the task is far from entirely easy and free from ambiguities.

In our paper we did not compute sense-level agreement, but we assume that it is likely that the level of agreement may be a bit lower than the one for lexeme-level annotations due to difficulties in understanding fine-grained senses and shades of meaning.

3. LCM and word senses

This section contains the description of selected verb senses, annotated with LCM tags. To illustrate sense-level annotations for LCM, let us focus on two verbs, picked from our dictionary: *dzielić* (eng. divide or share) in Table 1 and *zginąć* (eng. die) in Table 2. In each of the tables, column called 'domain' contains Polish WordNet verb domain of a specific sense `https://en.wikipedia.org/wiki/PlWordNet`.

Let our example discussion be based on the verb *dzielić* ('to divide'or 'to share') which has multiple senses that illustrate its various meanings spanning across all possible LCM labels.

The most abstract sense has the LCM label SV and its corresponding WordNet domain is state. It's English equivalent is 'to share'. In Polish, it refers to an abstract property of something being shared between multiple objects or people. For instance, in programming, an object reference may be shared between multiple class instances. A point of view may be shared between multiple people. No physical correlates are involved and the meaning is clearly an abstract one too, therefore SV label is the most appropriate.

In its sense related to social life domain, the verb becomes interpretative (IAV). It's English equivalent in this case means 'to divide'. An example of meaning reflected here may refer to groups of people divided by their opposite opinions, often linked to strong sentiments. There are no physical correlates and no objects are involved, therefore IAV tag is appropriate.

Finally, the verb may give a description of an observable event in a situational context. For example, a separation of ownership (eg. ownership of something is divided between multiple owners). This situation usually refers to some owned entity, therefore in this meaning the verb becomes a DAV.

Generally, the principles behind LCM labels make the distinction between IAV and DAV sometimes vague. If a verb refers to observable events in a situational context, but requires additional interpretation and evaluation, it is an IAV. Otherwise, we assumed it's a DAV, especially if some physical correlates may be found. As for the verb 'to share', some of its meanings rely on context, whereas other meanings possess an autonomous, context-independent meaning.

4. The Polish LCM (LCM-PL): Current State

Currently, the dictionary is available both as one downloadable file as well as in three separate pieces that reflect its structure (manual sense-level, lexeme-level annotations and automated annotations). The most recent version of the dictionary and its components are maintained at `http://clip.ipipan.waw.pl/LCM-PL`.

LCM	part (1)	part (2)	part (3)
SV	12%	4%	0%
IAV	35%	20%	2%
DAV	52%	75%	98%
count	10000 verb senses	1200 verbs	9200 verbs

Table 3: State of Polish LCM dictionary (LCM-PL)

- Part (1) Contains sense-level manual LCM annotations.
- Part (2) Contains lexeme-level manual LCM annotations.
- Part (3) Contains lexeme-level automatic LCM annotations.

The most recent state of the dictionary is reflected in Table 3 that contains percentages of LCM tags and verb (also verb sense) frequencies for manually annotated LCM tags.
Part (2) originates from The General Inquirer verbs with LCM tags translated into Polish. The translations have been manually corrected to ensure that the Polish verbs match English versions and their LCM tags were adjusted for Polish.
Part (3) reflecting automated LCM labels are generated using word embeddings and a neural network. We intend to provide the most recent evaluation (in terms of accuracy) of this part of the dictionary at the URL of the resource at `http://clip.ipipan.waw.pl/LCM-PL`.
The current state of automated predictions is as follows. For predicting LCM tags in (3) we used word2vec word embeddings of size 300 as verb representations. Embeddings were trained on the National Corpus of Polish (`http://www.nkjp.pl`) and Polish Wikipedia. To automatically predict LCM tags, we applied a two-layer (each size 300) perceptron-style neural network with selu activations that takes embedding of a verb as its input and predicts its output LCM tag. We measured the accuracy of this solution using 10-fold cross-validation on the data set from (2). The network, after hyper-parameter space optimizations, reached the average accuracy of 91.07 with standard deviation between folds at (+/- 2.73).
Our neural network solution turns out to be significantly better than results obtained using machine learning and hand-crafted features reported in (Rogozinska and Wawer, 2013). The evaluation data set was the same in both cases: the list of 1170 manually-labelled, lexeme-level, verbs taken from the English from the General Inquirer dictionary and translated into Polish.
Our results obtained using this enlarged data set and neural network approach are on similar or even slightly better level than the agreement between human annotators reported in (Rogozinska and Wawer, 2013). Despite this, we plan to increase the quality of predictions even further by experimenting with other neural architectures such as convolutional networks or the newest regularization algorithms.
One interesting conclusion from Table 3 is decreasing percentage of SV verbs between parts 1, 2 and 3, and increasing percentage of DAV verbs. This reflects the intuition that there are relatively few state verbs but they are among the most frequently used words. And on the contrary, there are many descriptive verbs, infrequently used, closely related to nominal meaning (sharing material correlates). The amount of IAV verbs is also decreasing with the frequency of usage.

5. Conclusions

In this paper we described the initial release of Linguistic Category Model (LCM) dictionary for the Polish language (LCM-PL), we believe the first widely usable version of the resource intended to measure the level of language abstractness in Polish.
The previous research on this topic (Rogozinska and Wawer, 2013) (especially in automated LCM dictionary generation) demonstrated limited usability of machine learning to automatically obtain LCM labels. In our paper we introduced a resource that is annotated manually in its most important parts: verbs whose senses that are used most frequently. The remaining part of Polish verbs, those less frequently used, was a subject of annotation on the level of lemmas, manual and automated – using word embeddings and a neural network. The quality of automated annotations has been evaluated and is comparable to human-level annotations.
In the future, we plan to further extend the manually annotated part of the dictionary. We also plan to increase the size of the manually labelled sense-level dictionary and manually verify more lemma-level annotations. We estimate that the size of the manually annotated part will not have to be larger than twice the size of the current state of the dictionary (as of early 2018).
The reason behind this is that the coverage of all Polish verbs using manual sense-level sense annotation is not only not feasible, but also quite possibly is not needed for effective LCM labeling of texts. Hopefully, word usage frequencies follow Zipf distributions and high quality sense-level coverage is important only for frequently used verbs. Many Polish verbs are used only occasionally.

6. Acknowledgements

Work financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

7. Bibliographical References

Beukeboom, C., Tanis, M., and Vermeulen, I. (2013). The language of extraversion: Extraverted people talk more abstractly, introverts are more concrete. *Journal of Language and Social Psychology*, 32(2):191–201.

de Montes, L. G., Semin, G. R., and Valencia, J. F. (2003). Communication patterns in interdependent relationships. *Journal of Language and Social Psychology*, 22(3):259–281.

Douglas, K. M. and Sutton, R. M. (2006). When what you say about others says something about you: Language abstraction and inferences about describers' attitudes and goals. *Journal of Experimental Social Psychology*, 42(4):500–508.

Kędzia, P., Piasecki, M., and Orlińska, M. (2015). Word sense disambiguation based on large scale polish clarin heterogeneous lexical resources. *Cognitive Studies / Études cognitives*, 15:269–292.

Maass, A., Salvi, D., Arcuri, L., and Semin, G. R. (1989). Language use in intergroup contexts: the linguistic intergroup bias. *Journal of personality and social psychology*, 57(6):981.

Maass, A., Ceccarelli, R., and Rudin, S. (1996). Linguistic intergroup bias: Evidence for in-group-protective motivation. *Journal of Personality and Social Psychology*, 71(3):512.

Piasecki, M., Szpakowicz, S., and Broda, B. (2009). *A Wordnet from the Ground Up*. Oficyna Wydawnicza Politechniki Wroclawskiej, Wrocław.

Rogozinska, D. and Wawer, A. (2013). Interpreting or describing? measuring verb abstraction. In *Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops*, ICDMW '13, pages 963–966, Washington, DC, USA. IEEE Computer Society.

Rubini, M. and Sigall, H. (2002). Taking the edge off of disagreement: linguistic abstractness and self-presentation to a heterogeneous audience. *European Journal of Social Psychology*, 32(3):343–351.

Schmid, J., Fiedler, K., Semin, G., and Englich, B. (2017). Measuring implicit causality: The linguistic category model. unpublished manuscript.

Semin, G. R. and Fiedler, K. (1988). The cognitive functions of linguistic categories in describing persons: Social cognition and language. *Journal of personality and Social Psychology*, 54(4):558.

Stone, P. J., Dunphy, D. C., Smith, M. S., and Ogilvie, D. M. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, MA.

Wigboldus, D. H., Semin, G. R., and Spears, R. (2000). How do we communicate stereotypes? linguistic bases and inferential consequences. *Journal of personality and social psychology*, 78(1):5.

WordNet-Shp: Towards the Building of a Lexical Database for a Peruvian Minority Language

Diego Maguiño-Valencia, Arturo Oncevay-Marcos and Marco A. Sobrevilla Cabezudo
Research Group on Artificial Intelligence (IA-PUCP)
Departamento de Ingeniería, Pontificia Universidad Católica del Perú, Lima, Perú
{dmaguino,arturo.oncevay,msobrevilla}@pucp.edu.pe

Abstract

WordNet-like resources are lexical databases with highly relevance information and data which could be exploited in more complex computational linguistics research and applications. The building process requires manual and automatic tasks, that could be more arduous if the language is a minority one with fewer digital resources. This study focuses in the construction of an initial WordNet database for a low-resourced and indigenous language in Peru: Shipibo-Konibo (shp). First, the stages of development from a scarce scenario (a bilingual dictionary shp-es) are described. Then, it is proposed a synset alignment method by comparing the definition glosses in the dictionary (written in Spanish) with the content of a Spanish WordNet. In this sense, word2vec similarity was the chosen metric for the proximity measure. Finally, an evaluation process is performed for the synsets, using a manually annotated Gold Standard in Shipibo-Konibo. The obtained results are promising, and this resource is expected to serve well in further applications, such as word sense disambiguation and even machine translation in the shp-es language pair.

Keywords: WordNet, lexical database, minority language

1. Introduction

The building of digital linguistic resources is a great support for endangered languages, as they help to preserve relevant information and knowledge related, not only for the language itself, but also for the community whose speak it. Nevertheless, if those resources are not developed to be able for further analysis and research, they may be insufficient to assist in the preservation efforts (Berment, 2002).

In that context, computational linguistics is a research area that aims to understand linguistic phenomena, in an automatic way, through the processing and exploiting either linguistic corpora or language patterns from large amounts of data. In order to achieve that goal, the corpus must be in a machine-readable format, and might include structured information and linguistic meta-data that helps to automatically understand patterns from the language.

Among the most important lexical and structured resources, the WordNet is included (Fellbaum, 1998). This resource could be used as a thesaurus for different languages, and its exploitation might ease more complex tasks such as word sense disambiguation or even machine translation.

For that reason, this article describes the building of an initial version of a WordNet for an endangered language, which faces additional problems caused by the low-density of digital resources. The language case study is Shipibo-Konibo (SHP), one of the 47 indigenous languages spoken in Peru, specifically in the Amazonian region and has over 23,000 speakers. (Ministerio de Educación, Perú, 2013). Like most of it peers, SHP is classified as a minority language from both a social and a computational perspective (Forcada, 2006).

The paper is organized as follow. Section 2 defines shortly what a WordNet is. Next, Section 3 presents works regarding the main aspects in the building of WordNet-like resources for other languages. After that, the construction of the WordNet for Shipibo-Konibo is detailed in Section 4. Additionally, there is an evaluation process in Section 5. Finally, conclusions and future work are discussed.

2. WordNet

A WordNet is a lexical database in an specific language. WordNet contains words grouped into synonyms sets (synsets) where each synset represent a different sense and is identified by a interlingual index (ILI) that allows to work in multilingual contexts. In addition to having a synonyms set, each synset may show a definition (gloss) and also some use examples. The synsets are connected between them through semantic and lexical relationships like hypernym, hyponym, and others (Fellbaum, 1998).

There are shallow similarities between a WordNet and a thesaurus, which different sets of terms are grouped based on a meaning-similarity criteria. On the other hand, the labels of the Wordnet are defined by the semantic relationship between the words or entries, while the clusters of words in a synonym dictionary may not follow any distinctive pattern of explicit meaning similarity (Miller, 1995).

For the Spanish language, there are two main multilingual options that are vastly used: MultiWordNet (Emanuele et al., 2002) and Multilingual Central Repository (MCR) (Gonzalez-Agirre et al., 2012). Despite the fewer amount of synsets contained in the latter repository, MCR will be used in the study because its more recent updates.

Finally, a WordNet is established ideally as a free and open-source resource, so the goal is similar for the Shipibo-Konibo WordNet.

3. Related Works

The section describes the different aspects related to the development or improvement of WordNet-like resources for different languages, whether they are minority ones or not.

Farreres et al. (1998) presented a semi-automatic approach to address development of new WordNets, using the English version as a model. There is a manual alignment for

verbs and nouns, while the validation step was automated. The latter process chose the most relevant terms and develop a complete taxonomy of semantic primitives. The proposed method was applied to build both Spanish and Catalan WordNets, using bilingual dictionaries and lexicons as the main sources.

Semi-automatic procedures were also proposed for building WordNets in other languages. For Persian, there were special considerations regarding the verb composition, as they are formed by more than 2 verbs usually (Rouhizadeh et al., 2008). Besides, an Ancient Greek WordNet was developed from a Greek-English dictionary, by supposing a semantic closeness regarding the terms translated (Bizzoni et al., 2014).

Thai WordNet was built using an own system called WordNet Builder (Sathapornrungkij and Pluempitiwiriyawej, 2005). The WordNet in English and machine readable dictionaries (MRD) were the main input sources, and both were connected trough a Link Analyzer for synset match. The validation process was performed by a statistical classifier, reducing human intervention.

In the Polish version, the validation process was performed with a similarity measure with the English WordNet, enhanced by human evaluation later (Broda et al., 2008). For the study, the synsets were built using a metric for semantic relation between different terms and POS-tags.

There were other studies focused in the improvements of previous WordNets. Mititelu (2012) proposed an addition of morpho-semantic relations for the Romanian WordNet. Heuristics were applied to group the terms and affixes in pairs. Then, these pairs were semantically tagged in three levels: monolingual, multilingual and for different NLP applications. Likewise, Bond et al. (2009) addressed improvements in the Japanese WordNet by increasing the vocabulary coverage, annotating more bilingual English-Japanese texts, and connecting the WordNet with different resources such as lexicons or image repositories.

Finally, Taghizadeh and Faili (2016) focused their efforts in building WordNet for low-resource scenarios. Using an Expectation-Maximization (EM) algorithm plus a cross-lingual word sense disambiguation method, a high quality WordNet could be built for Persian. Other positive aspect was the use of minimal resources, such as a bilingual dictionary and a monolingual textual corpus.

4. WordNet-Shp

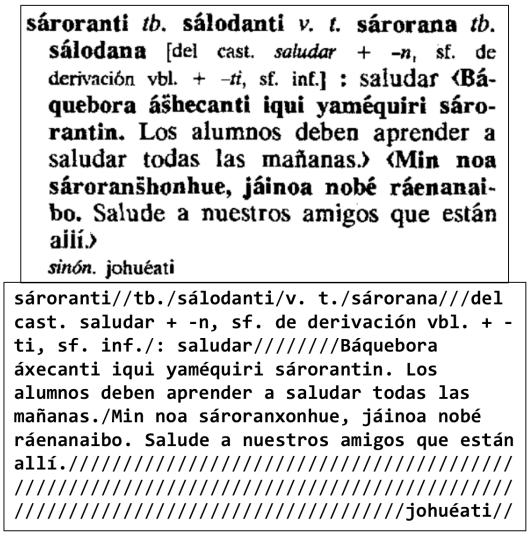
This section includes the steps followed for the development of the initial WordNet-Shp, a WordNet-like resource for Shipibo-Konibo. There are two main phases. The first phase consisted in the digitalization and pre-processing of a bilingual dictionary shp-es (Spanish). The second one consisted on the synset alignment task by using a similarity metric (provided by word2vec) with the definition glosses in the dictionary and the Spanish WordNet of Multilingual Central Repository (MCR)(Gonzalez-Agirre et al., 2012). The words of different glosses are compared against each other to obtain an average for each word obtained from the gloss of the dictionary and then averaged to obtain a final result the synset. The highest result is chosen.

The processed dictionary, the aligned synsets, and the Gold Standard for evaluation which consisted of a hundred synsets (see Subsection 5.1) are available in a project site[1].

4.1. Pre-Processing a Dictionary

An algorithm was built to automatically extract the words from an old-fashioned Spanish-Shipibo bilingual dictionary (Lauriout et al., 1993). For each word entry, an structured output is obtained, which includes several fields.First, there are 9 unique fields: term, reference, type of variant of the term, variant, grammatical category (POS-tag), main part, type of variant of the main part, variant of the main part, etymology. Then, given that a term might have more than one sense, each sense must be stored separately. So it was confirmed that the maximum number of different senses per word in the dictionary was 12. Each sense includes 6 glosses, 2 usage notes and 3 examples. Thus, it makes a total of eleven fields for every sense. Finally, there are 3 unique fields at the end of the entry: Synonymous paragraph, parent term, sub-entry type. The latter two only applies when the term is a sub-entry and is related to a parent main entry.

For instance, the parsed output for the term *sároranti* would be structured as in Figure 1.



sároranti *tb.* **sálodanti** *v. t.* **sárorana** *tb.* **sálodana** [del cast. *saludar* + *-n*, sf. de derivación vbl. + *-ti*, sf. inf.] : saludar ‹**Báquebora áshecanti iqui yaméquiri sárorantin.** Los alumnos deben aprender a saludar todas las mañanas.› ‹**Min noa sároranshonhue, jáinoa nobé ráenanaibo.** Salude a nuestros amigos que están allí.›

sinón. johuéati

```
sároranti//tb./sálodanti/v. t./sárorana///del
cast. saludar + -n, sf. de derivación vbl. + -
ti, sf. inf./: saludar////////Báquebora
áxecanti iqui yaméquiri sárorantin. Los
alumnos deben aprender a saludar todas las
mañanas./Min noa sároranxonhue, jáinoa nobé
ráenanaibo. Salude a nuestros amigos que están
allí.////////////////////////////////////////
////////////////////////////////////////////
/////////////////////////////////johuéati//
```

Figure 1: (Top) Original dictionary entry for *sároranti*. (Bottom) Parsed output for the entry. There are many separators ('/') together due to the possibility of extracting other elements between them. In this case, there is only one word sense, so there are a lot of separators together at the end, since up to 12 senses are expected as maximum.

The total amount of entries stored is about 5800, including 4815 terms from the main grammatical word classes (nouns, verbs, adjectives and adverbs). The remaining 985 are suffixes, prefixes, conjunctions, prepositions, among other categories. A distribution of the main word entries and their respective amount of senses in the Shipibo-Konibo Wordnet is presented in Table 1.

[1]WordNet-Shp data available in: chana.inf.pucp.edu.pe/resources/wordnet-shp

POS / #s.	Nouns	Verbs	Adj.	Adv.
1	2 231	1 453	357	96
2	174	266	62	11
3	59	48	13	2
4	14	16	1	0
5	6	3	0	0
6	2	0	0	0
7	0	0	0	1
Total	2 486	1 786	433	110

Table 1: Distribution of words with and without ambiguous senses found in the Shipibo-Konibo dictionary: Number of senses (#s.) per Part-of-Speech (POS) tag

4.2. Synsets Alignment

The alignment algorithm focuses in comparing the glosses of the word entries in Shipibo with the data of a Spanish WordNet, in order to obtain the closeness between the meanings among them. To define which synset each word sense belongs to, all the synsets in Spanish (terms, glosses and examples) were taken into account.
For this research, a word2vec model (Mikolov et al., 2013) was trained based on a general corpus (without annotation) of the Spanish language composed by approximately 1.4 billion words (Cardellino, 2016). The corpus was created by compiling various resources of the Spanish language that can be found on the Internet. Some of them are listed below:

- Collection of legal texts in Spanish from the European Union
- United Nations documents
- Parts of corpus in Spanish in other languages
- Protected articles from the Wikipedia in Spanish

With the trained word2Vec model, the classification algorithm for the synset alignment was the next step. The pseudo-code is presented below:

Algorithm 1 Synset alignment

for $i = 1$ to $|V|$ **do**
  for $j = 1$ to $|s_{v_i}|$ **do**
    $max_{ij} = 0$;
    for $k = 1$ to $|W_{es}|$ **do**
      $sim_{ijk} = word2vec_similarity(s_{jv_i}, W_{es_k})$
      if $sim_{ijk} > max_{ij}$ **then**
        $max_{ij} = sim_{ijk}$
      end if
    end for
    $insert_to_BD(s_{jv_i}, W_{es_{argmax(k)}})$
  end for
end for

Where $|V|$ is the size of vocabulary V, $|s_{v_i}|$ is the number of different senses for the word s_{v_i}, and $|W_{es}|$ is the size of entries in the Spanish WordNet (W_{es}). word2vec_similarity is obtained by calculating the cosine between the two vectors.
Regarding the pseudocode, this algorithm works as follows: Each sense of each word found in the Shipibo dictionary was compared to each synset of the WordNet in Spanish using the word2vec model previously trained to obtain a similarity metric. This measurement was expressed as a decimal number.
For the calculation, it was taken the same word, glosses and corresponding examples, if any, of each Spanish word found in the Shipibo gloss. The glosses were filtered to consider only the words of the categories that arise more frequently and that allow to understand the context. These categories are nouns and verbs; and to a lesser extent, adjectives and adverbs.
For example, in the case of a noun, the nouns and verbs of its gloss were considered (due to the strong connection between the two categories). Each word was taken only once, eliminating repetitions because it could be the case that the terms or words contained in the glosses would be repeated.
Additionally, the Tree-tagger (Schmid, 1995) was used to extract grammatical categories and lemmas from words in Spanish. For instance, for the verb *manéxti* (”clean” in English), which was taken in the sense whose gloss is ”to tie the hairs of the head or crown with other hairs of the same head”, the process would be as follows:

- Tie the hairs of the head or the crown with other hairs his own head to
- Tying, hair, head, crown, hair, head - Only verbs and nouns
- Tie, hair, head, crown - Lemmas were extracted and word repetitions were erased

The similarity was expressed as a decimal number between -1 and 1 (being the result of a cosine). The greater the result, the relationship between words would be closer. Therefore, to decide which synset corresponds to the word in Shipibo, the synset with the greater similarity is considered. Each time the synset corresponding to a word in Shipibo was found, everything related to the term was inserted in the database following the standard used by MCR (Gonzalez-Agirre et al., 2012).
Once the algorithm with two words was executed in Shipibo as a sample, the nearest Spanish synset was obtained, and that was found by the classification algorithm described. For example, *mocoxoti* (which means ”clean”) was correctly classified into one of the synsets where the word ”clean” (in Spanish) is found in the same sense as to sort or arrange. It also shows all the words associated with the synset, as well as its ili code (used as the international standard).

5. Evaluation

To carry out the evaluation a gold standard was needed. It was prepared manually by a group of linguists and native speakers of Shipibo. The evaluation metric in this study is the accuracy, which is calculated after testing the classification algorithm with the gold standard.

Category	# Synsets
Nouns	105
Verbs	43
Adjetives	8
Adverbs	2

Table 2: Current state of the WordNet-Shp

5.1. Gold Standard

The gold standard was made by linguists and a native Shipibo-Konibo speaker. They selected a group of words extracted from the WordNet in Spanish and put all the possible synonyms for each one. In this way, a hundred synsets were formed manually, separated by grammatical categories as thus: 76 formed by nouns, 15 by verbs, 7 by adjectives and 2 by adverbs.

5.2. Results

All the words of the gold standard were evaluated to analyze if the retrieved synset is really the corresponding one. The total accuracy obtained with the gold standard was 32.8%. Some samples of classified synsets are presented below:

- Synset: spa-30-03571439-n Word: *chachí* (injector). Gloss: injector. Synset Result: spa-30-03571439-n
- Synset: spa-30-05599617-n Word: *cói* (chin). Gloss: protruding part of the jaw. Synset Result: spa-30-05598147-n
- Synset: spa-30-03343853-n Word: *tóoati,tsakati* (gun) Gloss: portable weapon. Synset Result: spa-30-00001740-n

In the first sample, there is a match because the original and the retrieved synset are the same. In the second example, the synsets do not match so the classification was labeled as incorrect. However, in the synset result (spa-30-05598147-n) the word "nose" shows up, which is a part of one's face so it's related to what we were originally looking for (chin). In the third and final example, the classification was not correct either but there is a lexical gap because the word *tsakati* wasn't found in the dictionary as a entry.
The precision obtained might be due to the following points:

- The obtained synset makes sense but it's not the same ILI. This could happen because the wordnet is very refined
- Some or several words used in the gloss of a particular word do not bear much relation with it
- Minor errors in the dictionary processing

Finally, after processing words from the dictionary that were included in the gold standard and an extra hundred (200 in total), the current state of the Wordnet in Shipibo by number of synsets is presented in Table 2.

6. Conclusions and Future Work

This study aimed the development of a new lexical and synonym-based resource for the Shipibo-Konibo language. Likewise, this repository uses the international standard for locating translations in other languages based on the synsets codification. For this purpose, a bilingual dictionary was pre-processed for extracting information of word entries and their senses in Shipibo-Konibo. After that, using the Spanish WordNet, an algorithm aligned each word sense in Shipibo-Konibo with its Spanish peer. The existing relationships in the Spanish WordNet were considered in order to be inherited in the Shipibo-Konibo repository.
Regarding the evaluation, there was a manual analysis of the synset quality. This was supported by professional linguists, and the output was the development of the Gold Standard. The results of the alignment showed a close similarity in the word sense distribution between Shipibo-Konibo and Spanish. This is caused mainly by the high presence of unique-sense words. Besides, as the number of senses is higher, the amount of words decreases considerably.
Finally, the developed resource stores all the words including their different senses in Shipibo-Konibo. Also, there is a web interface under development for querying the entries. All of these resources will be available in the following link: https://github.com/iapucp/wordnet-shp-lrec2018
As future work, we want to improve our algorithm of synset alignment and to include other relations between synsets, like hypernyms or hyponyms. We believe that it will not be difficult because it is possible to bring these relations from WordNets in other languages (like Spanish).

Acknowledgments

We highly appreciate the linguistic team effort that made possible the creation of this resource: Dr. Roberto Zariquiey, Alonso Vásquez, Gabriela Tello, Renzo Ego-Aguirre, Lea Reinhardt and Marcela Castro. We are also thankful to our native speakers (Shipibo-Konibo) collaborators: Juan Agustín, Carlos Guimaraes, Ronald Suárez and Miguel Gomez. Finally, we gratefully acknowledge the support of the "Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica" (CONCYTEC, Peru) under the contract 225-2015-FONDECYT.

7. Bibliographical References

Berment, V. (2002). Several directions for minority languages computerization. In *Proceedings of the 19th international conference on Computational linguistics-Volume 2*, pages 1–5. Association for Computational Linguistics.

Bizzoni, Y., Boschetti, F., Diakoff, H., Del Gratta, R., Monachini, M., and Crane, G. R. (2014). The making of Ancient Greek WordNet. In *LREC*, volume 2014, pages 1140–1147.

Bond, F., Isahara, H., Fujita, S., Uchimoto, K., Kuribayashi, T., and Kanzaki, K. (2009). Enhancing the japanese wordnet. In *Proceedings of the 7th workshop on Asian language resources*, pages 1–8. Association for Computational Linguistics.

Broda, B., Derwojedowa, M., Piasecki, M., and Szpakowicz, S. (2008). Corpus-based semantic relatedness for the construction of Polish WordNet. In *LREC*.

Cardellino, C. (2016). Spanish Billion Words Corpus and Embeddings, March.

Emanuele, P., Luisa, B., and Christian, G. (2002). MultiWordNet: developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet, Mysore, India*.

Farreres, X., Rigau, G., and Rodriguez, H. (1998). Using wordnet for building wordnets. *arXiv preprint cmp-lg/9806016*.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Forcada, M. (2006). Open source machine translation: an opportunity for minor languages. In *Proceedings of the Workshop "Strategies for developing machine translation for minority languages", LREC*, volume 6, pages 1–6.

Gonzalez-Agirre, A., Laparra, E., and Rigau, G. (2012). Multilingual central repository version 3.0. In *LREC*, pages 2525–2529.

Lauriout, E., Day, D., and Loriot, J. (1993). Diccionario Shipibo-Castellano.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Ministerio de Educación, Perú. (2013). *Documento nacional de lenguas originarias del Perú*. MINEDU. URI: http://repositorio.minedu.gob.pe/handle/123456789/3549.

Mititelu, V. B. (2012). Adding morpho-semantic relations to the Romanian Wordnet. In *LREC*, pages 2596–2601.

Rouhizadeh, M., Shamsfard, M., and Yarmohammadi, M. A. (2008). Building a WordNet for Persian verbs. In *in the Proceedings of the Fourth Global WordNet Conference (GWC'08). The Fourth Global WordNet Conference, 2008*. Citeseer.

Sathapornrungkij, P. and Pluempitiwiriyawej, C. (2005). Construction of Thai WordNet lexical database from machine readable dictionaries. *Proc. 10th Machine Translation Summit, Phuket, Thailand*.

Schmid, H. (1995). Treetagger— a language independent part-of-speech tagger. *Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart*, 43:28.

Taghizadeh, N. and Faili, H. (2016). Automatic Wordnet development for low-resource languages using cross-lingual WSD. *J. Artif. Intell. Res.(JAIR)*, 56:61–87.

Retrieving Information from the French Lexical Network in RDF/OWL Format

Alexsandro Fonseca[1], Fatiha Sadat[1] and François Lareau[2]
[1]University of Quebec in Montreal, [2]University of Montreal
[1]201 President Kennedy, Montreal, Canada, [2]C.P. 6128 succ. Centre-Ville, Montreal, Canada
affonseca@gmail.com, sadat.fatiha@uqam.ca, francois.lareau@umontreal.ca

Abstract

In this paper, we present a Java API to retrieve the lexical information from the French Lexical Network, a lexical resource based on the Meaning-Text Theory's lexical functions, which was previously transformed to an RDF/OWL format. We present four API functions: one that returns all the lexical relations between two given vocables; one that returns all the lexical relations and the lexical functions modeling those relations for two given vocables; one that returns all the lexical relations encoded in the lexical network modeled by a specific lexical function; and one that returns the semantic perspectives for a specific lexical function. This API was used in the identification of collocations in a French corpus of 1.8 million sentences and in the semantic classification of these collocations.

Keywords: lexical network, lexical functions, Meaning-Text Theory, ontology, RDF, OWL

1. Introduction

The languages RDF/OWL have been an important tool for building interconnected resources in the web, due to their simplicity. RDF allows the construction of knowledge graphs. OWL allows the inference of logical relations among the objects represented in those graphs and the creation of classes of objects. RDF and OWL are, to date, the most successful knowledge representation languages (Hendler and van Harmelen, 2008). The set of resources in RDF/OWL format that are connected to each other through the internet is known as the Semantic Web.

Linguistic resources have been developed on top of RDF/OWL or transformed into an RDF/OWL format. As examples, we cite: WordNet (Fellbaum, 1998), DBPedia Wiktionary,[1] FrameNet (Fillmore, 1977), etc.

For a more detailed representation of linguistic information, however, the RDF/OWL languages are not sufficient. For this reason, metalinguistic ontologies were developed to represent information such as *part of speech*, *direct object*, *noun phrase*, etc. Those metalinguistic ontologies evolved into the *lexicon model for ontologies* (lemon) (McCrae et al., 2011), the most recent ISO standard for the representation of lexical information in the Semantic Web.

We have developed a metalinguistic ontology (*lexfom*) to represent Meaning Text Theory's (MTT) lexical functions. *Lexfom* uses the *lemon* model to represent information about lexical entries and lexical senses. This ontology was applied in the transformation of the French Lexical Network into an RDF/OWL format.

In this paper, we present a Java API that was developed to retrieve the lexical and combinatorial information from the French Lexical Network, which is based on lexical functions, in an RDF/OWL format.

This paper is divided as follows. In §2., we present the notions behind our API: the lexical functions and the French Lexical Network, a semantic classification of lexical functions and the metalinguistic ontologies, including the ontology that we have developed to represent MTT's lexical functions.

In §3., we present the functions that we have developed in our API to retrieve information from the French Lexical Network in RDF/OWL format. In §4., we conclude and discuss future work.

2. Related Work

2.1. The French Lexical Network

To our knowledge, the *French Lexical Network* (FLN) (Lux-Pogodalla and Polguère, 2011) is the only lexical network based on lexical functions. It has been developed as part of the *RELIEF* project[2] at *ATILF*.[3]

Unlike other lexical networks, such as WordNet (Fellbaum, 1998), the FLN does not make a taxonomic classification of words (Polguère, 2014). Moreover, the FLN contains syntagmatic relations between lexemes, usually absent from other lexical networks.

In this paper, we adopt the nomenclature used by the MTT: the term *vocable* refers to a canonical form of a word, independent of its meaning. The term *lexeme* refers to a specific acceptation of a vocable. For example, the vocable *mouse* has two different lexemes, $mouse_I$ (an animal) and $mouse_{II}$ (a computer device).

A lexical function (LF) (Mel'čuk, 1998) is a linguistic tool to represent different types of relations between lexemes.

Those relations can be paradigmatic, such as synonymy, antonymy and hyperonymy, or syntagmatic (horizontal relations in a sentence or collocation), such as intensification (e.g. *strongly condemn*) and subjective qualification (e.g. *fruitful analysis*).

LFs have the following general format: LF (*base*) = *value*. The *value* is a set of one or more lexemes. For example: Anti (*small*) = {*big*}; Hyper (*cat*) = {*feline*, *mammal*, *animal*}; Magn (*applause*) = {*thunderous*}. Simple LFs can be combined to form complex LFs: AntiMagn (*applause*) = {*scattered*}.

[1] www.dbpedia.org/page/Wiktionary

[2] www.atilf.fr/spip.php?article908

[3] Analyse et Traitement de la Langue Française: www.atilf.fr

The FLN is available for download in an XML format on ORTOLANG[4] (ATILF, 2017).
Since the information in the FLN is encoded in an XML format, and not in RDF/OWL, they cannot be immediately connected to the Semantic Web. Moreover, the information about LFs is only textual. This means that we do not have, for example, the following information:

- How complex LFs are formed from simple LFs. For example, that the LF $AntiMagn$ is composed from the LFs $Anti$ and $Magn$;
- How an LF like $Oper_1$ is related to the LFs $Real_1$ or $Func_1$ through the first actant (represented by the index *1*);
- Information about the semantic perspective of a lexical function (presented in the next section);

For this reason, we have developed a metalinguistic model called *lexfom*[5] (Fonseca et al., 2016a), which is presented in §2.4., to represent the characteristics of LFs and we have applied this model in the transformation of the FLN into an RDF/OWL format[6].

2.2. Semantic perspective for lexical functions

Jousse (2010) presents four different classifications for LFs: a semantic, a pragmatic, a combinatorial and a syntactic classification. These classifications are called "perspectives". In this paper, we are interested in the semantic perspective (SP).
The SP is comprised of ten classes: *action/event*, *causativity*, *element/set*, *equivalence*, *location*, *opposition*, *participants*, *phase/aspect*, *qualification* and *utilization form*. We added two classes to this classification, *semantically empty verb* and *support verb*.
Some of those classes have sub-classes. For example, the class *qualification* is sub-divided into *intensity* (e.g. Magn (*shave*) = {*close*}), *positive evaluation* (e.g. Bon (*contribution*) = {*valuable*}), and *negative evaluation* (e.g. AntiBon_1Involv (*car*) = {*smash into* N}, where N represents a noun).
Finally, the lexical relation between lexemes modeled by a specific LF can be classified in the same way.

2.3. Metalinguistic ontologies

The languages RDF/OWL only allow the representation of simple statements, encoded as triplets. For the representation of more complex linguistic information, metalinguistic ontologies based on RDF/OWL had to be developed.
The first metalinguistic ontology based on RDF/OWL was *ISOCat* (*ISO TC37 Data Category Registry*).[7] It was proposed and developed by the Psycholinguistic Department of the Max Planck Institute.[8] Its aim is to define grammatical categories, such as transitive and intransitive verbs, part of speech, predicate, etc.
Another important metalinguistic ontology is the *Lexical Markup Framework* (LMF) (Francopoulo et al., 2006). LMF is an ISO project that started in 2005 and was first published in 2007. Its aim is to be a common standard in the development of dictionaries for the Semantic Web. It is designed to represent morphological, syntactic and semantic information.
Some other metalinguistic ontologies were developed after LMF, leading to the publishing of a new W3C standard in 2016, called *lexicon model for ontologies* (lemon).[9] Lemon is based in previous models, such as LMF, ISOCat, LexInfo,[10] etc.
Lemon's main modules are the following: *Ontology-lexicon interface* (ontolex), *Syntax and Semantics* (synsem), *Decomposition* (decomp), *Variation and Translation* (vartrans) and *Linguistic Metadata* (lime).
The *ontolex* module implements a *LexicalEntry* object, which is used to represent a canonical form of a word, and a *LexicalSense* object, which is used to represent each specific sense of a word.
In our model, which is presented in the next section, each vocable and lexeme are represented by a *ontolex LexicalEntry* and a *LexicalSense* object, respectively.

2.4. Lexical functions ontology model

The *Lexical functions ontolgoy model* (lexfom) (Fonseca et al., 2016a; Fonseca et al., 2016b) is a metalinguistic ontology of lexical functions and lexical relations.
It comprises four modules:

- *Lexical functions representation* (lfrep) represents an LF's characteristics, such as its semantic actants;
- *Lexical functions relation* (lfrel) represents a relation between lexemes, which can be paradigmatic or syntagmatic;
- *Lexical functions family* (lffam) represents a syntactic classification for LFs. For example, the LF $Oper_1$ and the complex LFs composed by $Oper_1$ belong to the same family;
- *Lexical functions semantic perspective* (lfsem) is a semantic classification of LFs, based on the work of (Jousse, 2010).

We apply our model to create an RDF/OWL version of the FLN. About 46,000 paradigmatic relations and 8,000 syntagmatic relations extracted from the FLN are represented in an RDF/OWL format using the *lexfom* model.
Figure 1 shows the RDF code, in Turtle dialect,[11] representing the French collocation *porter un vêtement* (to wear a piece of clothing) using *lexfom*'s four modules and the

[4] www.ortolang.fr/market/item/lexical-system-fr/v1
[5] https://github.com/alex-fonseca/lexfom
[6] https://github.com/alex-fonseca/rlfowl
[7] www.isocat.org
[8] www.mpi.nl
[9] www.w3.org/community/ontolex
[10] lexinfo.net
[11] www.w3.org/TR/turtle

lemon's *ontolex* module. Not only the collocation is represented, but also each vocable with all their meanings found in the FLN and the LF modeling the syntagmatic relation in the collocation ($Real_1$ (*vêtement* = {*porter*}).

```
:lex_vetement a ontolex:LexicalEntry,
           ontolex:Word;
     ontolex:canonicalForm :form_vetement;
 ontolex:sense :vetement_sense_I.2;
 ontolex:sense :vetement_sense_I.1;
 ontolex:sense :vetement_sense_II;
 ontolex:sense :vetement_sense_III.1;
 ontolex:sense :vetement_sense_III.2;
     rdfs:label "vetement"@fr .

:form_vetement a ontolex:Form;
     ontolex:writtenRep "vetement"@fr .

vetement_sense_I.2 a ontolex:LexicalSense .
vetement_sense_I.1 a ontolex:LexicalSense .
vetement_sense_II a ontolex:LexicalSense .
vetement_sense_III.1 a ontolex:LexicalSense .
vetement_sense_III.2 a ontolex:LexicalSense .


:lex_porter a ontolex:LexicalEntry, ontolex:Word;
    ontolex:canonicalForm :form_porter;
        ontolex:sense :porter_sense_I.1;
        ontolex:sense :porter_sense_IV;
    rdfs:label "porter"@fr .

:form_porter a ontolex:Form;
     ontolex:writtenRep "porter"@fr .

porter_sense_I.1 a ontolex:LexicalSense .
porter_sense_IV a ontolex:LexicalSense .


LF-Real1 rdf:type lfrep:simpleLF,
           owl:NamedIndividual ;
   lfrep:belongsToLFFamily
           lffam:LFF-synt-realV-Real1;
   lfrep:hasSyntActant
           lfrep:lfrep-const-sa-ASynt_1;
   lfrep:dimension
           lfrep:lfrep-type-syntagmaticLF;
   lfrep:semanticPerspective
    lfsem:pSem-ae-utilizationTypicalOperation.


:lfsr_11420 a lfrel:SyntagmaticLFSenseRelation;
     lfrel:hasLexicalFunction lfrep:LF-Real1;
     lfrel:hasLFKeyword
                 ontolex:vetement_sense_I.2;
     lfrel:hasLFValue ontolex:porter_sense_IV;
     lfrel:hasGovPattern
                 lfgpat:"DET ~s"^^xsd:string;
     lfrel:relationDirection lfrel:valueKeyword;
     lfrel:hasFusedElement "false"^^xsd:boolean.
```

Figure 1: RDF code representing the vocables *vêtement* and *porter*, each lexical sense of both vocables, the LF $Real_1$, and finally the syntagmatic relation between a specific lexeme of each vocable.

3. Java API

In this section, we present the Java API to retrieve information from the FLN in RDF/OWL format.

3.1. API's general vision

We implemented different functions to retrieve information from the RLF in RDF/OWL format[12], using the SPARQL query language. Our implementation uses the Apache Jena ARQ,[13] a query engine implementing SPARQL.
The main functions in our API are:

- *getLexicalRelationForVocables (String vocable1, String vocable2, int typeRelation)*: given two vocables v_1 and v_2, this function returns the lexical relations (paradigmatic or syntagmatic) present in the RLF between any sense of v_1 and v_2. It is possible to search only for syntagmatic or paradigmatic relations between v_1 and v_2, by setting the variable *typeRelation*. This function is useful, for example, for applications searching for collocations, as shown in (Fonseca et al., 2017);

- *getLexicalRelationLFForVocables (String vocable1, String vocable2, int typeRelation)*: the difference between this function and the last one is the possibility of also searching for the LF modeling the relation between any senses of the vocables;

- *getLexicalRelationsForLF (String lf, typeRelation)*: given a LF *lf*, it is possible to find all the lexical relations modeled by *lf* in the FLN. It is also possible to specify only syntagmatic or paradigmatic relations;

- *getSemanticPerspectives (String lf)*: it returns all the semantic perspectives for a specific LF. Since some LFs are complex, they can have more than one semantic perspective. This function is useful, for example, in the identification of the semantic relation connecting the lexemes in a collocation. For example, the collocation *good review* is modeled by the LF *Bon*: Bon (*review*) = {*good*}. By identifying such a collocation in a text, we can find in the FLN that it is modeled by the LF *Bon* and that this LF has a semantic perspective of "positive evaluation". This information can be useful for applications in sentiment analysis, for example.

In the next subsection we show how the second function presented above is used in the identification of collocations from a corpus.

3.2. Identification of collocations

As an example of the application of this API, we applied it in the identification of collocations, as presented in (Fonseca et al., 2017). About 1.8 million phrases from the French part of the Eurosense corpus (Delli Bovi et al., 2017) were extracted and a dependency parser was applied to them. The dependency relations found in the corpus are searched in the FLN's syntagmatic relations, using the function *getLexicalRelationLFForVocables(String vocable1, String vocable2, int typeRelation)*. The positive matches are kept as possible collocations and later manually analyzed to decide if they are true collocations.
Fourteen different dependency relations are tested. We show here examples of five of these relations:

- a_obj: argument introduced by the preposition *à* - *à fond* (*thoroughly*);

- mod: modifiers (adjectival, nominal and adverbial) - *politique véritable* (*true policy*);

- obj: object of a verb - *traiter les maladies* (*to treat diseases*);

[12] https://github.com/alex-fonseca/lexfom-api

[13] jena.apache.org/documentation/javadoc/arq/org/apa-che/jena/query/package-summary.html

- p_obj: argument introduced by a preposition (other than *à* and *de* - *sur la table* (*on the table*);

- dep_coord: links a conjunct to the previous coordinator - *dans le car* (*in the car*);

The advantage of using dependency parsing combined with the FLN is shown by the following sentence: "*Quel pouvoir sur les âmes Hussey exerce-t-elle encore?*" (What power over souls is Hussey still exerting?). In this example, there is a dependency relation (*obj*) between *pouvoir* and *exerce*. The pair (*pouvoir*, *exerce*) can be searched in the FLN and the collocation *exercer le pouvoir* will be retrieved, together with the LF modeling this collocation: Oper_1 ($pouvoir_{II}$) = $exercer_{II.1}$. By this method, such a collocation can be identified, even though the vocables *pouvoir* and *exerce* are distant in the sentence.
Table 1 shows the precision for some dependencies and the total precision for the 14 dependencies. The complete table for all 14 dependency types is presented in (Fonseca et al., 2017).

Table 1: Precision in the identification of collocations by syntactic dependency.

dependency	# candidates	# true coll.	precision
mod	20 625	14 240	0.690
obj	4 869	4 720	0.969
a_obj	300	295	0.983
dep_coord	246	13	0.053
p_obj	90	86	0.956
Total	**43 629**	**33 273**	**0.763**

The most similar work to ours in the identification of collocations is the one presented by (Garcia et al., 2017). They identify collocations from three pairs of parallel corpora: English-Spanish, English-Portuguese and Spanish-Portuguese. The main difference between their work and ours is that they only use three dependencies: adjectival modifiers (*amod*), nominal modifiers (*nmod*) and verb-object (*vobj*), which are less likely to produce errors, since the governor and the dependent are adjacent to each other. Their average precision for the three language pairs are: 91.8% for *amod*, 90.6% for *nmod* and 86.2% for *vobj*.
In general, we expected to have good precision for all types of dependencies, since each candidate is matched against the collocations represented in the ontology and the ontology is based on the FLN, which is manually constructed.
However, we had false positives due to parsing errors. The most common are the errors connected to false positive collocations formed by the verbs: *pouvoir* (can) (35.1%), *avoir* (have) (31.1%) and *être* (be) (29.5%), as explained in (Fonseca et al., 2017).
Another frequent error is connected to the conjunction *car* (because), which is homonymous with the noun *car* (*bus*). In collocations like *dans le car* (*inside the bus*), it was often mistakenly tagged as a conjunction, with the dependency *dep_coord*. This explains why candidates in this group had low precision.

3.3. Classification of collocations in semantic categories

The fourth function presented in §3.1. is used in the semantic classification of collocations. The function *getSemanticPerspectives (String lf)* is used in the identification of the SP of each LF modeling each identified collocation.
For example, the API's function used to retrieve the collocation *exercer le pouvoir* (to exert power) from the FLN, presented in the previous subsection, also retrieves the LF Oper_1, which models the syntagmatic relation between the lexemes $pouvoir_{II}$ and $exercer_{II.1}$. We then use the function *getSemanticPerspectives* ($Oper_1$), which returns the SP *supportVerb*. By this method, we can identify the semantic category of each collocation.
As presented in (Fonseca et al., 2017), the main SPs for collocations identified from the EuroSense corpus are:

- *qualification* (33.9%). Example: *très grave* (very serious) - Magn (*grave*) = {*très*}.

- *supportVerb* (24.4%). Example: *exercer le pouvoir* (to exert power) - Oper_1 (*pouvoir*) = {*exercer*}.

- *location* (17.9%). Example: *dans le pays* (in the country) - Loc_{in} (*pays*) = {*dans*}.

- *actionEvent* (9.7%). Example: *l'avion atterrit* (the plane lands) - FinFact_0 (*avion*) = {*atterrir*}.

4. Conclusion and Future Work

The FLN is unique in the sense that it is the only lexical network based on lexical functions and the only one to represent syntagmatic relations between lexemes in a graph-based architecture.
The FLN is available for download in XML. Using a metalinguistic ontology created to represent lexical functions (Fonseca et al., 2016a; Fonseca et al., 2016b), we have created an RDF/OWL version of the relations inside the FLN.
In this paper, we presented a Java API developed to retrieve information from the RDF/OWL version of the RLF. We showed two applications for this API: the identification of collocations from a textual corpus and the semantic classification of the identified collocations.
As future work, we intend to connect each sense in the FLN to the senses in DBPedia, creating a stronger connection between the FLN and the Semantic Web.

5. Acknowledgements

We thank Professor Alain Polguère, director of the project RELIEF, for kindly providing us with access to the French Lexical Network before its public release.

6. Bibliographical References

ATILF. (2017). Réseau lexical du français (rl-fr). ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

Fonseca, A., Sadat, F., and Lareau, F. (2016a). Lexfom: a lexical functions ontology model. In *Proceedings of the Fifth Workshop on Cognitive Aspects of the Lexicon (CogALex), COLING*, pages 145–155, Osaka, Japan.

Fonseca, A., Sadat, F., and Lareau, F. (2016b). A lexical ontology to represent lexical functions. In *Proceedings of the 2nd Workshop on Language and Ontologies (OntoLex), LREC*, pages 69–73, Portorož, Slovenia.

Fonseca, A., Sadat, F., and Lareau, F. (2017). Combining dependency parsing and a lexical network based on lexical functions for the identification of collocations. In Ruslan Mitkov, editor, *Computational and Corpus-Based Phraseology*, pages 447–461, Cham. Springer International Publishing.

Garcia, M., García-Salido, M., and Alonso-Ramos, M. (2017). Using bilingual word-embeddings for multilingual collocation extraction. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 21–30, Valencia, Spain, April. Association for Computational Linguistics.

Hendler, J. and van Harmelen, F. (2008). The semantic web: Webizing knowledge representation. In Frank van Harmelen, et al., editors, *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 821–839. Elsevier.

Jousse, A.-L. (2010). *Modèle de structuration des relations lexicales fondé sur le formalisme des fonctions lexicales*. Ph.D. thesis. Thèse de doctorat dirigée par Sylvain Kahane et Alain Polguère, Université de Montréal et Université Paris Diderot (Paris 7).

Lux-Pogodalla, V. and Polguère, A. (2011). Construction of a French Lexical Network: Methodological Issues. In *First InternationalWorkshop on Lexical Resources, WoLeR 2011*, pages 54–61, Ljubljana, Slovenia, August.

McCrae, J., Spohr, D., and Cimiano, P. (2011). Linking lexical resources and ontologies on the semantic web with lemon. In *Proceedings of the 8th Extended Semantic Web Conference on The Semantic Web: Research and Applications - Volume Part I*, ESWC'11, pages 245–259, Berlin, Heidelberg. Springer-Verlag.

Mel'čuk, I. (1998). Collocations and lexical functions. *Phraseology. Theory, Analysis and Applications*, pages 23–53.

Polguère, A. (2014). From writing dictionaries to weaving lexical networks. *International Journal of Lexicography*, 27(4):396–418.

7. Language Resource References

Delli Bovi, C., Collados, J. C., Raganato, A., and Navigli, R. (2017). Eurosense: Automatic harvesting of multilingual sense annotations from parallel text. In *Proceedings of 55th annual meeting of the Association for Computational Linguistics (ACL 2017)*, Vancouver, Canada.

Christiane Fellbaum, editor. (1998). *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May.

Fillmore, C. J., (1977). *Scenes-and-frames semantics*. Number 59 in Fundamental Studies in Computer Science. North Holland Publishing.

Francopoulo, G., George, M., Calzolari, N., Monachini, M., Bel, N., Pet, Y., and Soria, C. (2006). Lexical Markup Framework (LMF. In *Proceedings of the International Conference on Language Resources and Evaluation - LREC-2006*.

Transforming Wikipedia into a Large-Scale Fine-Grained Entity Type Corpus

Abbas Ghaddar, Philippe Langlais
RALI-DIRO
Montreal, Canada
abbas.ghaddar@umontreal.ca, felipe@iro.umontreal.ca

Abstract

This paper presents WiFiNE, an English corpus annotated with fine-grained entity types. We propose simple but effective heuristics we applied to English Wikipedia to build a large, high quality, annotated corpus. We evaluate the impact of our corpus on the fine-grained entity typing system of Shimaoka et al. (2017), with 2 manually annotated benchmarks, FIGER (GOLD) and ONTONOTES. We report state-of-the-art performances, with a gain of 0.8 micro F1 score on the former dataset and a gain of 2.7 macro F1 score on the latter one, despite the fact that we employ the same quantity of training data used in previous works. We make our corpus available as a resource for future works.

Keywords: Annotated Corpus, Fine-Grained Entity Type, Wikipedia

1. Introduction

Entity typing is the task of classifying textual mentions into their respective types. While the standard Named-Entity Recognition (NER) task focuses on a small set of types (e.g. 4 classes defined by the CONLL shared task-2003 (Tjong Kim Sang and De Meulder, 2003)), fine-grained tagging deals with much larger type sets (e.g. 112 types used in (Ling and Weld, 2012)). Entity typing has received an increasing interest lately from the NLP community, due its role in Relation Extraction, Entity Linking, Question Answering, etc.

One issue in fine-grained typing is the absence of a well-established training corpus. The large number of types makes it difficult to manually annotate the amount of data needed for training. This bottleneck was addressed by using an automatic annotation procedure (Section 2), which follows two steps:

1. Identifying and linking entity mentions to a Knowledge Base (typically Freebase).
2. Assigning to each mention the set of types that apply in the context of the sentence.

Step 1 suffers a number of issues: lack of coverage when Wikipedia is used as a source (Ghaddar and Langlais, 2016b), and entity linking which is error prone (Ren et al., 2016). Step 2 also has limitations: the type of a mention is often resolved with strict pruning heuristics (regardless of the context) as in (Gillick et al., 2014); or the type of a mention is kept ambiguous following (Shimaoka et al., 2017). For instance, in the sentence: "*Gonzales* embarked on a pop career as the leader of the alternative rock band Son." The entity *Chilly Gonzales* has 3 labels in Freebase: *musician*, *writer* , *actor* but only *musician* applies here.

In this paper, we revisit the idea of automatically extracting fine-grained entity annotations from Wikipedia. Similarly to previous works, we gather annotations from anchored texts in an article, as well as their associated types in Freebase (Bollacker et al., 2008). In addition, we also generate annotations for texts not anchored in Wikipedia following (Ghaddar and Langlais, 2017). We do this by considering coreference mentions of anchored texts as candidate annotations, and by exploiting the out-link structure of Wikipedia. We propose an easy-first annotation pipeline described in Section 3 which happens to reduce noise. Second, we define simple yet efficient heuristics in order to prune the set of candidate types of each entity mention found in the article. These heuristics are based on: Freebase tuples, the high density of entity mentions, and the paragraph and section structure of the article.

We applied our methodology on a 2013 English Wikipedia dump, leading to a large annotated corpus called WiFiNE, which contains more annotations than similar corpora. We evaluate annotation quality intrinsically on a set of manually annotated mentions. We perform an extrinsic evaluation by training the entity typing model of (Shimaoka et al., 2017) on randomly generated subsets of WiFiNE. We compare the performances obtained by the resulting models on two well-established test sets: FIGER (GOLD) (Ling and Weld, 2012) and ONTONOTES (Gillick et al., 2014). The newly trained models clearly outperform previous ones on both benchmarks, demonstrating the superiority of our approach.

In summary, our contributions are the following:

- We provide over 110M proper name, nominal, and pronominal mentions annotated with fine-grained entity types in two taxonomies.
- We measure the efficiency of WiFiNE for training fine-grained entity typing. We outperform state-of the art results by 0.3 strict, and 0.8 macro F1 scores on the FIGER benchmark and by 0.9 strict, and 2.3 macro F1 scores on the OntoNotes dataset.

The remainder of paper is organized as follows. Section 2, discusses recent related works. We describe the annotation process along with the main statistics of our corpus in Section 3. An evaluation of WiFiNE on entity typing is described in Section 4, before concluding and discussing future works in Section 5.

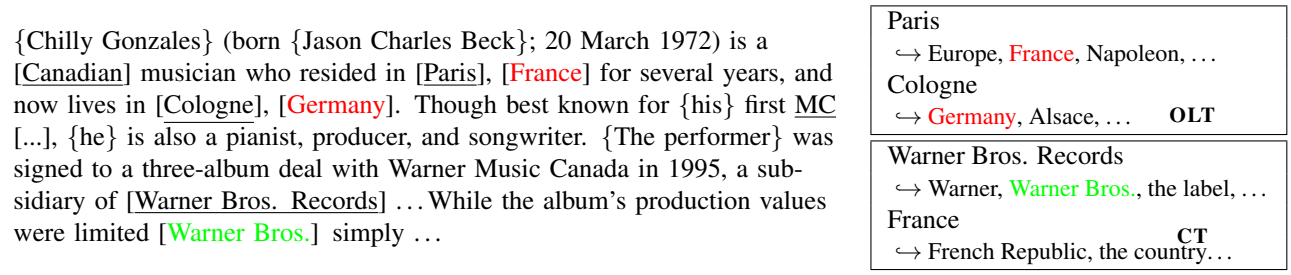


Figure 1: Illustration of the process with which we gather annotations into WiFiNE for the target page `https://en.wikipedia.org/wiki/Chilly_Gonzales`. Square Bracketed segments are the annotations; curly brackets indicate mentions from the resource of (Ghaddar and Langlais, 2016a); while underlined text are anchored texts in the corresponding Wikipedia page. OLT represents the out-link table (which is compiled from the Wikipedia out-link graph structure), and CT represents the coreference table we gathered from the resource.

2. Related Works

In previous works, the entity mention detection process is performed using one of two methods. The first one consists in using the internal links in Wikipedia as training data, where anchored strings (that have an equivalent page in Freebase) are treated as entity mentions (Ling and Weld, 2012; Ren et al., 2016). Another method is to directly use a Freebase entity resolver such as `DB-pedia Spotligh` (Daiber et al., 2013) to link textual mentions to their Freebase page (Gillick et al., 2014; Yogatama et al., 2015; Ren et al., 2016).

In both cases, the Freebase `object_type` attributes of the entity are mapped to a predefined set of types. In the last few years, two popular mapping schemes emerged: FIGER (Ling and Weld, 2012) (112 label) and GILLICK (Gillick et al., 2014) (89 label). They are both organized in a hierarchical structure, where children labels also inherit the parent label. FIGER defines a 2-level hierarchy (e.g. `/person` and `/person/musician`); while GILLICK uses 3 levels of types (e.g. `/person` and `/person/artist`, `/person/artist/musician`).

Most resolved entities have multiple type labels, but not all of them typically apply in a given context. One solution consists in ignoring the issue, and instead relying on the robustness of the model to deal with heterogeneous labels; this approach is adopted by (Yogatama et al., 2015; Shimaoka et al., 2017). Another solution involves filtering. In (Ling and Weld, 2012; Gillick et al., 2014), the authors apply hard pruning heuristics:

- **Sibling pruning** Removes sibling types if they came from a single parent type. For instance, a mention labelled as `/person/artist/musician` and `/person/artist/actor` would be tagged by `/person/artist` and `/person`.
- **Minimum count pruning** All labels that appear once in the document are removed. For example, if multiple entities in a document are labelled as `/person/artist/musician` and only one of them have `/person/artist/actor` as an extra label, the latter is considered noisy.

Such heuristics decrease the number of training data by 40-45% according to (Gillick et al., 2014; Ren et al., 2016).

Ren et al. (2016) propose a distant supervision approach to deal with noisy labelled data. Their method consists in using unambiguous mentions to de-noise mentions with heterogeneous labels that appear in a similar context.

Because only a tiny portion of texts in Wikipedia are anchored, some strategies are needed to infer more annotations. In this study, we revisited the approach of (Ghaddar and Langlais, 2017) which consist in annotating Wikipedia with coarse-grained entity type (PER, LOC, ORG and MISC), resulting in a corpus called WiNER. In this paper, we propose to extend this approach with more types and mentions, leading to WiFiNE. First, we enrich the corpus with nominal and pronominal coreference mentions, then we extend the set of types (4 previously) to either 112 (FIGER) or 89 (GILLICK). In the next Section, we summarize the original process proposed by (Ghaddar and Langlais, 2017) and then we describe our extensions.

3. WiFiNE

3.1 Mention Recognition

The pipeline used to extract annotations from Wikipedia is illustrated in Figure 1, for an excerpt of the Wikipedia article `Chilly_Gonzales`, hereafter named the target article. The anchored texts of out-links in the target article are elected entity mentions. For instance, we identify *Warner Bros. Records* and *Paris* as mentions. In general, a Wikipedia article has an equivalent page in Freebase. We remove mentions that do not have such a page. This way, we filter out anchored texts that are not named-entities (such as *List of Presidents of the United States*).

Because the number of anchored strings in Wikipedia is rather small — less than 3% of the text tokens — we propose to leverage: (1) the out-link structure of Wikipedia, (2) the information of all the surface strings used to describe the main concept of a Wikipedia article. For the latter, we rely on the resource[1] described in (Ghaddar and Langlais, 2016a) that lists, for all the articles in Wikipedia (those that have a Freebase counterpart), all the text mentions that are coreferring to the main concept of an article (CT of Figure 1). For instance, for the article *Chilly_Gonzales*, the

[1] `http://rali.iro.umontreal.ca/rali/en/wikipedia-main-concept`

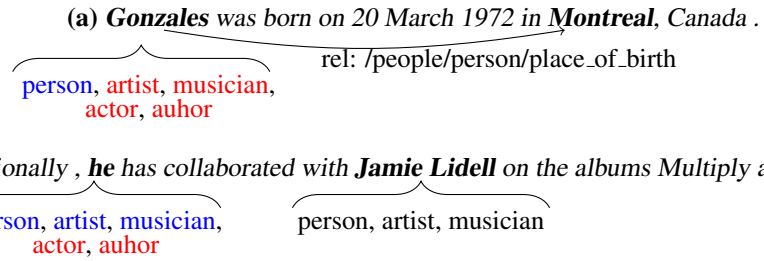


Figure 2: Illustration of our de-noise heuristic rules. Spans in bold are entity mentions. Blue labels are relevant ones, while red ones are irrelevant.

resource lists proper names (e.g. Gonzales, Beck), nominal (e.g. the performer) and pronominal (e.g. he) mentions that refer to Chilly Gonzales. Our strategy for collecting extra annotations is a 4-step process, where:

1. We consider direct out-links of the target article. We search the titles of the articles we reach that way. We also search for their coreferences as listed in the resource of (Ghaddar and Langlais, 2016a). For instance, we search (exact match) *Warner Bros. Records* and its coreferences (e.g. *Warner*, *Warner Bros.*) in the target article.

2. We follow out-links of out-links, and search in the target article (by an exact string match) the titles of the articles reached. For instance, we search for the strings *Europe*, *France*, *Napoleon*, as well as other article titles from the out-link list of the article *Paris*.

3. For the titles matched during step 2, we also match their coreferent mentions. For instance, because *France* was matched in the previous step, we also search its coreferences as listed in the coreference table (CT of Figure 1).

4. Last, we adapt the multi-sieve rule-based coreference resolver of (Raghunathan et al., 2010) to the specificity of Wikipedia in order to find the antecedent referents of a pronominal mention. The rules link a pronoun to its best antecedent mention based on attributes agreement (gender, number, entity type,...). We apply the pronoun coreference rules on each article, then discard all pronouns that do not refer to a Wikipedia entity mention.

During this process, some collisions may occur. We solve the issue of overlapping annotations by applying the steps exactly in the order presented above. Our steps have been ordered in such a way that the earlier the step, the more confidence we have in the strings matched at that step. It may also happen that two out-link articles contain the same mention (for instance *Washington_State* and *George_Washington* both contain the mention *Washington*), in which case we annotate this ambiguous mention with the type of the closest[2] unambiguous mention.

3.2 Manual Evaluation

Step 1 raises the coverage from less than 3% to 9.5%, step 2 further raises it to 11.5%, while step 3 and 4 increase it to 23% and 30% respectively. We assessed the annotation quality of a random subset of 1000 mentions. While we measure an accuracy of 92% and 88% for mentions detected during step 1 and 2 respectively, the accuracy decreases to 81% and 77% during step 3 and 4 respectively. We identified two main sources for errors in the coreferent mentions detection procedure.

a) *[Eldridge Pope]* was a traditional brewery.....Sixteen years later the *[Pope]*$_\star$ brothers floated the business...

b) Montreal Impact's biggest rival is *[Toronto FC]* because Canada's two largest cities have rivalries in and out of sport. Montreal and *[Toronto]*$_\star$ professional soccer teams have competed against each other for over 40 years.

Figure 3: Examples of errors in our annotation pipeline. Faulty annotations are marked with a star.

One source of error comes from the resource used to identify the mentions of the main concept. We measured in a previous work (Ghaddar and Langlais, 2016a), that the process we rely on for this (a binary classifier) has an accuracy of 89%. Example (a) of Figure 3 illustrates such a mistake where the family name *Pope* is wrongly assumed coreferent to the brewery *Eldridge Pope*. We also found that our 4-step process and the disambiguation rules fail in 15% of the cases. Figure 3 b) illustrates an example where we erroneously recognize the mention *Toronto* (referring to the town) as a coreferent of the (non ambiguous mention) *Toronto FC*, simply because the latter is close to the former.

3.3 Type Mapping

Following previous works, we map Freebase `object_type` attributes of each entity mention detected to a set of fine-grained types. An entity mention is said to be clean if its labels belong to only a single path (not necessarily a leaf); otherwise, it is noisy. For example, the mentions *France* or *Germany* with labels `/location` and `/location/country` are considered clean. On the other hand, the entity mention *Chilly Gonzales* annotated with 5 labels (`/person`, `/person/artist`, `/person/artist/musician`, `/person/artist/actor`, and `/person/artist/author`) is considered noisy because only one of the

[2]Before or after the named-entity.

last three types is qualified in a given context (see Fig. 2). We measured that 23% of mentions in WiFiNE that have two labels or more don't belong to a single path (noisy), and 47% of those have more than 2 noisy labels (e.g. *Gonzales* in Fig. 2). We propose to eliminate noisy labels in WiFiNE using rules based on the high coverage of entity mentions, coupled with Freebase triples and the paragraph and section structure of Wikipedia:

1. **Freebase Relation Type:** We label the mention by the type indicated by the relation. A Freebase relation is a concatenation of a series of fragments. The first two fragments of the relation indicate the Freebase type of the subject, and the third fragment indicates the relation type. In example (a) of Fig. 2, the triple (arg1: *Chilly Gonzales*; rel: `/people/person/place_of_birth`; arg2: *Montreal*) found in Freebase indicates that only `/person` should apply to the *Gonzales* mention in this context.

2. **Common Attribute Sharing:** If a non-ambiguous mention (*Jamie Lidell* in example (b)) has a type set which is a subset of another mention with noisy labels (*he*, referent of *Chilly Gonzales*) occurs in the same sentence, we assign to the noisy mention the common labels between both mentions.

We first apply our rules at the sentence level, then at the paragraph and section level. Whenever we de-noise an entity mention in such a way, all its coreferent mentions (in the scope) receive the same type.

Heuristic	**Pre**	**Rec**	**F1**
w/o Rules	31.8	**100.0**	48.3
Rule-1 only	48.8	87.2	62.3
Rule-2 only	56.4	85.6	68.0
Both Rules	**79.2**	81.8	**80.5**
Level of Application			
Sentence	66.5	85.5	73.7
+ Paragraph	72.7	82.6	78.6
+ Section	**79.2**	81.8	**80.5**

Table 1: De-noising rules evaluation on 1000 hand-labelled mentions following GILLICK type hierarchy.

We assessed the quality of our de-noising rules on 1000 randomly selected noisy mentions. Table 1 reports precision, recall and F1 scores on the ablation study of the proposed heuristics. We start with an accuracy of 48% when either rule is applied. We measure performance after removing labels identified as noisy by rule one, two and both. Also, we measure the accuracy when the rules are applied at sentence, paragraph and section levels. Results show that our rules greatly improve the annotation quality by roughly 32%. Also, we observe that the first rule is more important than the second, but both rules complement each other. As expected, applying the rules at paragraph and section levels further improve the performance. We identify two sources of errors: (1) pruning heuristics don't apply to 11% of mentions; (2) our rules failed to pick up the correct label in 9% of the cases. Example (a) of Figure 4 illustrates such a mistake where *Gonzales* is labelled as *musician* rather than *author* because *Feist* is considered as *musician* in this context. In example (b), *Gonzales* is wrongly labelled as *person* thought that the relation `/people/person/nationality` exist between both entity but the sentence don't state it.

a) *[Gonzales]*$_{musician\star}$ returned as a contributor on *[Feist]*'s 2007 album...

b) *[Gonzales]*$_{person\star}$ said in an interview: My experiences in *[Canada]* had been disappointing

Figure 4: Examples of errors in our de-noising rules. Faulty annotations are marked with a star.

Table 2 illustrates a randomly-picked selection of mentions annotated in WiFiNE, along with their type according to the GILLICK scheme. The last two examples illustrate noisy annotations. In the first one our process failed to distinguish between the company and its product. The second example is a mention detection error, we couldn't recognize *Viitorul Homocea* as an entity, because this soccer team does not have a page in Wikipedia or Freebase.

3.4 Corpus Statistics

WiFiNE is built from 3.2M Wikipedia articles, comprising more than 1.3G tokens accounting for 54M sentences, 41M of which contain at least one entity mention. Overall, it gathers 182.7M mentions: 95.1M proper, 62.4M nominal and 24.2M pronominal ones. Table 3 summarizes the mention statistics and label distribution over the number of levels of FIGER and GILLICK type hierarchies.

	FIGER	**GILLICK**
Total mentions	159.4	111.1
Proper mentions	82.5 (52%)	64.8 (58%)
Nominal mentions	55.9 (35%)	29.8 (27%)
Pronominal mentions	21.0 (13%)	16.5 (15%)
Total Labels	243.2	230.9
Level 1	153.8 (63%)	111.1 (48%)
Level 2	89.5 (37%)	90.0 (39%)
Level 3	-	29.8 (13%)

Table 3: Mention statistics and label distribution (in millions and percentages) over the number of levels of FIGER and GILLICK type hierarchy.

First, we note that the total number of mentions in FIGER and GILLICK is less than the total number of entity mentions. This is because: (a) we remove noisy mentions that our rules failed to disambiguate (11%), (b) some mentions cannot be mapped to either schemes (e.g. fictional characters). Second, we note that FIGER mentions out number those of GILLICK, simply because their scheme covers more types (112 vs 89).

Sentence	Labels
In **Kent v. Dulles** , 357 U.S. 116 (1958) , the Court held that the federal government . . .	/other /other/event
The **Cangrejal River** or **Río Cangrejal** is a river that drains several mountain tributaries . . .	/location /location/geography /location/geography/body_of_water
. . . editions of **Millionaire** to be aired between 7:00 and 7:30 pm	/other /other/art /other/art/broadcast
Mies Bouwman stopped **her** regular work after falling sick but has occasionally	/person /person/artist
. . . to imprisoned Christians and niece of the **Emperor Gallienus** , found Anthimus in prison .	/person /person/political_figure
. . . of **vinyl siding** which does not weather as wood does .	/other /other/product
The firm was the first state-owned rail vehicle in Argentina. . .	/organization /organization/company
The **1 – 2 ton** was a sailing event on the **Sailing at the 1900 Summer Olympics** program in Meulan	/other /other/event /other/event/sports_event
He took part in the White Council after **Sauron** 's return.. . .	/person /person/artist /person/artist/actor
Clove is **Syzygium aromaticum** and belongs to division of **Magnoliophyta** in the kingdom Plantae .	/other /other/living_thing
Pepsi⋆ also created a fellowship at Harvard University which enable students from. . .	/other /other/food
. . . Viitorul **Homocea**⋆ , Siretul Suraia and Trotusul Ruginesti deducted 3 points .	/location

Table 2: Random selection of annotations from WiFiNE following GILLICK type hierarchy. Faulty annotations are marked with a star.

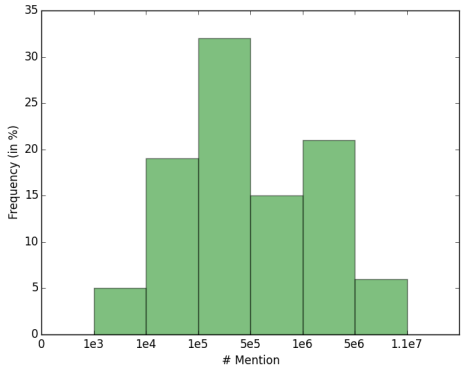


Figure 5: Distribution of entity type labels according to the FIGER type hierarchy.

Following the GILLICK scheme, each mention has 2 types on average, where 39% of them are of level 2, and 13% are of level 3. The distribution of level 2 and 3 labels in WiFiNE exceed its equivalent in the ONTONOTES (Gillick et al., 2014) dataset (29% and 3% respectively). Figure 5 illustrates the percentage of types that recieve a given number of mentions in WiFiNE. It shows that the majority of types have more than 100k mentions and roughly 25% (like `city`, `company`, `date`) exceeds 1M mentions. Also, we observe that 5% of the types have less than 10k mentions (e.g. `/event/terrorist_attack`), and none of them has less than 1k mentions[3].

[3]A similar distribution is obtained with GILLICK type hierarchy.

4. Evaluation on Entity Typing

4.1 Reference System

In all experiments, we deploy the off the shelf neural network model of (Shimaoka et al., 2017). Given a mention in its context, the model uses three representations in order to associate the mention with the correct types.

- **Mention representation:** the average of the mention words embedding.
- **Context representation:** First, a Bi-LSTM model is applied on the left and right context of the mention, then an attention layer is placed on top of the model.
- **Feature Representation:** They learn the representations of hand-crafted features.

We trained the tagger on various subsets of WiFiNE as described in the next section. We use the default configuration of the tagger, except the batch size which we set to 100 rather than 1000 and the learning rate that we changed from 0.001 to 0.0005[4]

4.2 Datasets and Evaluation Metrics

We evaluate the model on two manually annotated benchmark: FIGER (GOLD) (Ling and Weld, 2012) and ONTONOTES (Gillick et al., 2014). The first consist of 18 news reports annotated following FIGER scheme, while the second are 77 documents from the OntoNotes 5.0 (Weischedel et al., 2013) test set annotated according to the GILLICK scheme. Following previous works, we used Strict, loose Macro-averaged, and loose Micro-averaged

[4]We observed better results on the held-out development set.

F1 scores as metrics for evaluation. Strict measures exact match, while losses metrics measure macro/micro partial matches between gold and system labels. Macro is the average of F1 scores on all types, while Micro is the harmonic mean. Table 4 and 6 compared the performance obtained by the resulting models with those of previous works on FIGER (GOLD) and ONTONOTES test set respectively. We perform an ablation test on our 4-step process of Section 3.1 by training the model on 7 variants of WiFiNE:

- **Line 1-3:** hyperlinks + proper name coreference mentions (step 1 and 2 of Section 3.1)
- **Line 4:** hyperlinks + proper name + nominal coreference mentions (step 1-3 of Section 3.1).
- **Line 5:** hyperlinks + proper name + pronominal coreference mentions (step 1, 2 and 4 of Section 3.1).
- **Line 6-7:** hyperlinks + proper name + nominal + pronominal coreference mentions (all steps).

The goal is to validate if proper name, nominal and pronominal coreference mentions are necessary to fine-grained entity tying performance. For each variant, we report the average score on 5 randomly generated subsets. To be comparable with previous works, we used training materiel up to 4 million mentions, and leave experiments on the usefulness of the full WiFiNE for future work.

4.3 Results on FIGER (GOLD)

Previous works trained their models on 2.6 million mentions obtained by mapping hyperlinks in Wikipedia articles to Freebase[5].

Models				Strict	Macro	Micro
FIGER (Ling and Weld, 2012)				52.30	69.90	69.30
FIGER+PLE (Ren et al., 2016)				59.90	76.30	74.90
Attentive (Shimaoka et al., 2017)				59.68	78.97	75.36
(Abhishek et al., 2017)				65.80	**81.20**	77.40
	Proper	**Nominal**	**Pronominal**	**This work**		
(1)	1	0	0	61.99	76.20	75.12
(2)	2	0	0	63.77	77.56	76.25
(3)	3	0	0	63.41	78.03	76.32
(4)	1	1	0	64.83	79.26	77.36
(5)	1	0	1	63.06	79.00	76.77
(6)	1	1	1	65.19	79.59	77.55
(7)	2	1	1	**66.07**	79.94	**78.21**

Table 4: Results of the reference system trained on various subsets of WiFiNE, compared to other published results on the FIGER (GOLD) test set. Training data (in millions) include: proper name; nominal and pronominal mentions.

Our model trained on 4M mention (line 7) outperforms the initial model of (Shimaoka et al., 2017) by 6.2, 1.0 and 2.9 on strict, micro, macro F1 scores, and the state-of-the-art of (Abhishek et al., 2017) by 0.3 and 0.9 strict and macro F1 scores. First, we observe that using hyperlinks and proper name mentions (line 3) for training improves the performance of the original model of (Shimaoka et al., 2017) that uses data driven from hyperlinks only. Second, we notice that models trained on a mix of proper name and nominal (line 4) or pronominal (line 5) coreference mentions outperform the model trained on proper name mentions (line 2) solely. Third, we observe that the combination of 3 mention types (line 6-7) is required in order to outperform the state-of-the-art, which validate our 4-step method of Section 3.1.

[5]The dataset is distributed by (Ren et al., 2016)

Label type	FIGER (GOLD)	WiFiNE
/person	31.5%	16.6%
/organization	16.9%	7.7%
/location	13.2%	13.6%
/location/city	5.0%	4.3%
/organization/sports_team	4%	1.0%

Table 5: Comparison of the distribution of the top 5 types present in FIGER (GOLD) test set to that of WiFiNE.

Table 5 shows the 5 most frequent types the FIGER (GOLD) test set compared to those in WiFiNE. FIGER (GOLD) is a small dataset, it contains only 523 mentions annotated with 41 different labels. We observe that the type distribution in this dataset follows a zipfian curve, while the distribution of types in WiFiNE is similar to a normal distribution (Figure 5). Figure 6 illustrates some errors committed on FIGER (GOLD) dataset. Error mostly occur on mentions with labels that don't belong to a single path (example a), and on ambiguous mentions (example b).

(a) ... bring food for the employees at [Safeway] ...
Gold: /location /location/city
/organization /organization/company
Pred: /organization /organization/company

(b) *With the huge popularity of [EyeFi] cards ...*
Gold: /product
Pred: /organization

Figure 6: Examples of mentions erroneously classified in FIGER (GOLD) dataset.

4.4 Results on OntoNotes

Ren et al. (2016), Shimaoka et al. (2017) and Abhishek et al. (2017) trained their models on newswire documents present in OntoNotes (Weischedel et al., 2013), where entity mentions were automatically identified and linked to Freebase using DB-pedia Spotligh (Daiber et al., 2013). On the other hand, Gillick et al. (2014) and Yogatama et al. (2015) used an entity linker to automatically annotated 113k news documents. Results on the ONTONOTES dataset validate the observation we obtained on FIGER (GOLD). Models trained on proper names in addition to nominal (line 4 in Table 6) or pronominal (line 5) coreference mentions is better than only training on proper names (line 2). In addition, training on the combination of all coreference mentions (line 6-7) systematically improves performances.

Models				Strict	Macro	Micro
(Gillick et al., 2014)				N/A	N/A	70.0
K-WASABIE (Yogatama et al., 2015)				N/A	N/A	**72.98**
FIGER+PLE (Ren et al., 2016)				57.20	71.50	66.10
Attentive (Shimaoka et al., 2017)				51.74	70.98	64.91
(Abhishek et al., 2017)				52.20	68.50	63.30
	Proper	**Nominal**	**Pronominal**	**This work**		
(1)	1	0	0	55.25	68.21	61.49
(2)	2	0	0	57.05	71.96	66.03
(3)	3	0	0	57.47	72.87	66.97
(4)	1	1	0	57.17	73.07	67.30
(5)	1	0	1	57.50	73.08	67.35
(6)	1	1	1	57.80	73.60	67.82
(7)	2	1	1	**58.05**	**73.72**	67.97

Table 6: Results of the reference system trained on various subsets of WiFiNE, compared to other published results on the ONTONOTES test set. Training data (in millions) includes: proper name; nominal and pronominal mentions.

We outperform best results reported by previous works on strict, macro F1 scores by 0.9 and 2.3 receptively. On the other hand, we underperform (Gillick et al., 2014) and (Yogatama et al., 2015) and by 3 and 5 point on the micro metric respectively. In (Gillick et al., 2014; Yogatama et al., 2015), the authors do not report results on strict and macro metrics and neither their models nor their training data are available. Consequently, we couldn't specify the cause of the gap on the micro metric, but we report some improvement over (Shimaoka et al., 2017) model on the loose metrics. A potential reason behind this gap is that the text genre of their training data and that of ONTONOTES is the same (newswire). Our models were trained on randomly picked Wikipedia sentences (out of domain). Also, we note that in order to generate their corpus, (Gillick et al., 2014; Yogatama et al., 2015) applied filtering rules that are responsible for the loss of 45% of the mentions. We have no such heuristic here, but we still observe competitive performances.

Label type	Onto Test	WiFiNE
`/other`	44.0%	20.0 %
`/organization`	10.5%	6.3 %
`/person`	8.4%	17.6%
`/organization/company`	7.7%	2.3%
`/location`	7.6%	11.8%

Table 7: Comparison of the distribution of the top 5 types present in ONTONOTES test set to that of WiFiNE.

Table 7 shows the 5 most frequent types in the ONTONOTES dataset and in WiFiNE. Although ONTONOTES is much larger the FIGER (GOLD)[6], we still observe that the distribution of types in this dataset is zipfian. We also note that the type `/other` is over-represented (44%) in this dataset, because Gillick et al. (2014) annotated all non-entity mentions (examples in table 8) as `/other`. We observe that 73% of the wrong decisions that our model made on ONTONOTES are committed on this type. In WiFiNE, `/other` always refers to an entity mention, and in most cases the mention has an additional level two and three labels.

trouble
addition
personal reasons
some complications
additional evidence
diplomatic relations
a modest pretax gain
the active role taken
in the affairs of United
quotas on various economic indicators
the invitation of the Foreign Affairs Institute
amounts related to areas where deposits are received

Table 8: Examples of non-entity mentions annotated as `/other` in the of OntoNotes test set.

5. Conclusion

We built on the work of (Ghaddar and Langlais, 2017) which developed WiNER, a coarse-grained entity type corpus made merly from English Wikipedia articles, and propose WiFiNE, a fine-grained entity type corpus annotated with nominal and pronominal coreference mentions. We evaluated the impact of our corpus on a neural network tagging system with 2 human made benchmarks. Experiments shows state-of-the-art performances on both benchmarks, when WiFiNE is used as training materiel. Our analysis on both datasets indicates the following observations. First, enriching Wikipedia articles with proper names, nominal and pronominal mentions systematically leads to better performances, which validate our 4-step approach. Second, the correlation between the train and test type distribution is an important factor to entity typing performance. Third, models could benefit from an example selection strategy based on the genre of the test set. As future work, we want to study the usefulness of WiFiNE on a NER in Tweets, and if models can benefits from the full corpus. WiFiNE is publicly available at `http://rali.iro.umontreal.ca/rali/en/wifiner-wikipedia-for-et`. We hope this resource will foster further research on fine-grained entity type tagging.

6. Acknowledgements

This work has been founded by TRIBE, a Natural Sciences and Engineering Research Council of Canada (NSERC) Collaborative Research and Training Experience (CREATE) Program. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research. We thank the anonymous reviewers for their insightful comments.

[6] It contains roughly 9000 mentions annotated with 88 different types

7. Bibliographical References

Abhishek, A., Anand, A., and Awekar, A. (2017). Fine-Grained Entity Type Classification by Jointly Learning Representations and Label Embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 797–807, Valencia, Spain, April. Association for Computational Linguistics.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250.

Daiber, J., Jakob, M., Hokamp, C., and Mendes, P. N. (2013). Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

Ghaddar, A. and Langlais, P. (2016a). Coreference in Wikipedia: Main Concept Resolution. In *CoNLL*, pages 229–238.

Ghaddar, A. and Langlais, P. (2016b). WikiCoref: An English Coreference-annotated Corpus of Wikipedia Articles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, 05/2016.

Ghaddar, A. and Langlais, P. (2017). WiNER: A Wikipedia Annotated Corpus for Named Entity Recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 413–422.

Gillick, D., Lazic, N., Ganchev, K., Kirchner, J., and Huynh, D. (2014). Context-dependent fine-grained entity type tagging. *arXiv preprint arXiv:1412.1820*.

Ling, X. and Weld, D. S. (2012). Fine-Grained Entity Recognition. In *AAAI*.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501.

Ren, X., He, W., Qu, M., Voss, C. R., Ji, H., and Han, J. (2016). Label noise reduction in entity typing by heterogeneous partial-label embedding. *arXiv preprint arXiv:1602.05307*.

Shimaoka, S., Stenetorp, P., Inui, K., and Riedel, S. (2017). Neural Architectures for Fine-grained Entity Type Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1271–1280, Valencia, Spain, April. Association for Computational Linguistics.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., et al. (2013). Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*.

Yogatama, D., Gillick, D., and Lazic, N. (2015). Embedding Methods for Fine Grained Entity Type Classification. In *ACL (2)*, pages 291–296.

Error Analysis of Uyghur Name Tagging: Language-specific Techniques and Remaining Challenges

Abudukelimu Halidanmu[1], Abulizi Abudoukelimu[1], Boliang Zhang[2], Xiaoman Pan[2], Di Lu[2], Heng Ji[2], Yang Liu[1]

[1] Tsinghua University, [2] Rensselaer Polytechnic Institute

[1] {abdklmhldm}@gmail.com, [1] {keram1106}@163.com, [1] {liuyang2011}@tsinghua.edu.cn,

[2] {zhangb8,panx2,lud2,jih}@rpi.edu

Abstract

Regardless of numerous efforts at name tagging for Uyghur, there is limited understanding on the performance ceiling. In this paper, we take a close look at the successful cases and perform careful analysis on the remaining errors of a state-of-the-art Uyghur name tagger, systematically categorize challenges, and propose possible solutions. We conclude that simply adopting a machine learning model which is proven successful for high-resource languages along with language-independent superficial features is unlikely to be effective for Uyghur, or low-resource languages in general. Further advancement requires exploiting rich language-specific knowledge and non-traditional linguistic resources, and novel methods to encode them into machine learning frameworks.

Keywords: Low-resource Languages, Name Tagging, Error Analysis

1. Introduction

Uyghur is a language spoken by 8.2 million people, primarily by the Uyghur people in the Xinjiang Uyghur Autonomous Region of Western China. In terms of the number of native speakers, it's ranked at the 94th among all the languages in the world [1], but it has extremely low linguistic resources. There are very few Natural Language Processing (NLP) tools, standard annotated corpora, or language universal resources (e.g., World Atlas of Linguistic Structure (WALS) database (Haspelmath et al., 2005; Dryer and Haspelmath, 2013)) available. Even for naturally existing noisy annotations such as Wikipedia markups, Uyghur is ranked very low (the 195th [2]). There are only 2,566 Uyghur pages in Wikipedia, much fewer than its related languages such as Turkish (277,547 pages) and Uzbek (128,664 pages). Most Uyghur Wikipedia pages contain much less content than their counterparts in Turkish, Uzbek and English. The cross-lingual links are not carefully validated and thus contain many errors.

It's certainly important to develop automatic NLP tools for Uyghur, so as to distill information from textual documents written in Uyghur, as well as preserve their unique culture, music, art and the long history of which the Uyghurs are deeply prideful of. Unfortunately, the striking fact is that very little Uyghur NLP work has been published to catch the attention of the wider international NLP research community.

Using Uyghur name tagging as a case study, some previous studies (Li et al., 2011; Arkin et al., 2013b; Rozi et al., 2013; Turhun et al., 2012; Li, 2014; Arkin et al., 2013c; Arkin et al., 2013a; Maihefureti et al., 2014; Zhang, 2014; Zhang et al., 2015; Yu et al., 2015; Nizamidin et al., 2016) have adopted popular machine learning methods which were effective for other high-resource languages. The features suggested by these previous papers include: numbers, shape, stem, suffix, the number of suffixes, first syllable, last syllable, the number of syllables, Part-of-speech tags, the closest verb, word length, position in the sentence and special rules to identify Chinese person names. Further advances in this field require us to look into language-specific problems and recommended solutions to those challenges.

In this paper, we will look at the remaining errors of a high-performing Uyghur name tagger, and decompose the remaining errors into detailed categories in order to understand how varied components may contribute to improvement. We believe such comprehensive, quantitative and qualitative error analysis may help draw a roadmap for future research and resource development on this important and yet challenging task.

2. Approach Overview

We use a deep neural networks based Uyghur name tagger as our target system for analysis because of two reasons: (1) it achieves top performance at NIST LoreHLT2016 Evaluation [3] so it represents state-of-the-art; (2) unlike most previous work, this system has already exploited extensive language-specific features. We briefly describe the system as follows.

2..1 Learning Model

This system considers name tagging as a sequence labeling problem, to tag each token in a sentence as the Beginning (B), Inside (I) or Outside (O) of a name mention with a certain type (Person (PER), Organization (ORG), Geo-political Entity (GPE), and Location (LOC)). Following a framework similar to (Lample et al., 2016). The architecture consists of Bi-directional Long Short-Term Memory and Conditional Random Fields (CRFs) network. After processing through the Bi-LSTM networks, each token in a sentence sequence obtains a feature embedding that captures left and

[1] https://en.m.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers

[2] https://meta.wikimedia.org/wiki/List_of_Wikipedias

[3] https://www.nist.gov/itl/iad/mig/lorehlt16-evaluations

right context information, which is then fed into the CRF networks.

2..2 Pre-processing

The system starts with segmenting a document into tokens based on 50 punctuations pulled from Uyghur grammar books ((translated by Anne Lee), 2003; Zakir, 2007; Engesæth et al., 2009). Since Uyghur is a morphologically rich language, a set of name related suffixes is also extracted from grammar books, Wikitionary [4] and WALS [5], for stemming and feature encoding.

2..3 Features

Typical **implicit** linguistic features including character embeddings and word embeddings are learned from a large monolingual unlabeled corpus from LoreHLT2016 and then fed into the Bi-LSTM networks. Moreover, the following Uyghur-specific **explicit** context-dependent linguistic features are directly fed into the final CRFs model.

- The first and the last syllables of each token, based on the intuition that Uyghur names often include suffixes, and the first syllables of person names often follow some specific patterns.
- 319 common syllable patterns from person names. The most frequent patterns include (Latin: gue), (Latin: sha), مىر (Latin: mir), غا (Latin: gha), (Latin: uel), (Latin: ash), گۇل (Latin: guel), (Latin: buew), ئابدۇ (Latin: abdu) and مۇھەممە (Latin: muhemme)
- Suffixes are categorized into three types of features: (1) indicating animacy so that the word is likely to be part of a person or an organization name, including نىڭ (Latin: ning), نى (Latin: ni), لۇق (Latin: luq), and لىك (Latin: lik). (2) locative suffixes indicating GPE/LOC names, including كە (Latin: ke), گە (Latin: ge), قا (Latin: qa), غا (Latin: gha), تە (Latin: te), دە (Latin: de), تا (Latin: ta), دا (Latin: da), تىن (Latin: tin), دىن (Latin: din), تىكى (Latin: tiki), دىكى (Latin: diki), كىچە (Latin: kiche), گىچە (Latin: giche), قىچە (Latin: qiche), and غىچە (Latin: ghiche). (3) suffixes indicating a word is unlikely to be a name or part of a name, including لار (Latin: lar) and لەر (Latin: ler).
- Two words before and two words after the current token.
- Conjunction feature of stem and suffix.
- Name designators: English name designators are translated into Uyghur through a bi-lingual lexicon from LoreHLT2016.
- 446 Chinese last names are translated into Uyghur.

2..4 Post-processing

In the low-resource setting, the available resources are not sufficient to generalize some Uighur-specific linguistic phenomena as in high-resource language setting. We designed the following heuristic rules as post-processing to fix some obvious errors in informal genres like discussion forum and tweets.

- Remove a name if it includes digital numbers but it's not a poster or Twitter ID.
- If a name includes a URL link, remove the URL.
- Label places that don't have governing organizations as Location(LOC), including all continents, ئوتتۇرا شەرق (Latin: ottura sheriq, English: Middle East), etc.
- Label places with location modifiers as LOC, e.g., جەنۇبىي ئامرىكا (Latin: jenubiy amirika, English: North America).
- Label countries of countries as GPE, e.g., ياۋروپا ئىتتىپاقى (Latin: yawrupa ittipaqi, English: the European Union).
- Remove generic name mentions of people of the certain ethnicity (e.g., 'Uighur People', 'Americans', 'Arabs') by checking the combinations of country names and suffixes indicating 'people'.
- There are many very long nested organizations whose boundaries are difficult to determine. The basic principle is to tag every different, distinct entity by checking if it should be created as a unique entry when we construct a knowledge base. For example, بىرلەشكەن دۆلەتلەر تەشكىلاتى مائارىپ، پەن-مەدەنىيەت تەشكىلاتىنىڭ جۇڭگودا تۇرۇشلۇق ۋەكىللەر ئۆمىكى ئۇچۇر تارقىتىش باشقارمىسىنىڭ ("*Media Communication office from the United Nations Educational, Scientific, and Cultural Organization in China*") should be tagged as one single organization mention.
- Boundary extension: if a name doesn't include any suffix and its right contextual word is a name designator, then extend the name boundary to include the designator.
- Cross-genre propagation: when the types of the same name mention are conflicting between formal genres and informal genres, propagate the types from formal genres to informal genres.
- Poster names: Extract all poster names from the original thread structures, and identify all mentions in the posts, posters, and Twitter user names. Apply English entity linking (Pan et al., 2015) to each string after '@' or '#', and if it's linkable and its type can be inferred based on KB properties, then assign the type; otherwise tag it as PER.

2..5 Performance

For the experiment in this paper, we used the unsequestered Uyghur documents from the NIST LoReHLT16 evaluation. We used 99 documents for training and 30 documents for test and achieved 65.23% F-score. This performance is encouraging given the limited resources. The above explicit linguistic features achieved 2.4% F-score improvement. However, the overall performance is still much lower than other languages such as English, Spanish and Dutch (Lample et al., 2016), and also much lower than Uzbek (close to 80% F-score trained from a similar size of data) which is a similar language as Uyghur but has much more linguistic resources. In next section we will take a close look at the remaining errors.

3. Error Analysis

One major challenge to develop NLP techniques for low-resource languages is that system developers usually have little knowledge about the languages so it's very difficult to

[4] https://en.wiktionary.org/wiki/Wiktionary:Main_Page
[5] http://wals.info/

perform effective error analysis in order to do hill-climbing. Thus in this paper, we ask two Uyghur native speakers to focus on detailed error analysis. In this section, we aim to explain why Uyghur name tagging is so challenging, and discuss various methods we attempted to address these challenges, and potential language-specific solutions.

3..1 Error distribution

In Figure 1 we present the distribution of errors which need different techniques, according to their difficulty levels. The percentage numbers are approximate because some errors may rely on the combination of multiple types of features.

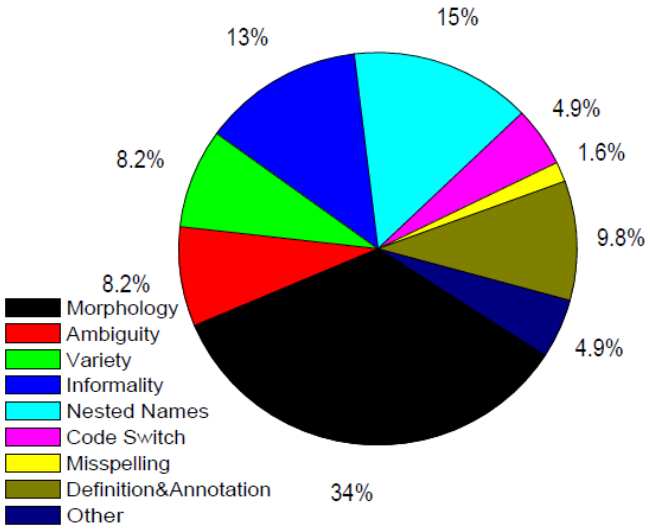


Figure 1: Error Type Distribution

3..2 Rich Morphology

From Figure 1 we can clearly see that morphology analysis dominates the causes for errors. Uyghur is a very 'sticky' language that contains very rich morphology. 90% words have arbitrary combinations of multiple suffixes denoting person, number, case, mood, etc. Sometimes a compound Uyghur word may even indicate an entire sentence, as shown in Table 1. Due to a large number of suffix combinations, among all of the unique tokens in our corpus, only 68.1% of them exist in the LDC provided lexicon, while 31.9% are Out-of-Vocabulary (OOV) words.

More than 90% GPE names include suffixes. State-of-the-art language universal morphology analyzers such as Morfessor (Creutz, 2003) don't perform well on identifying Uyghur suffixes for name tagging purpose. The suffixes we mined from grammar books, Wikitionary and WALS can perform some reasonable amount of stemming. However, it still remains a big challenge for machine learning models to generalize these rich suffix combinations.

3..3 Ambiguity

Names and their contexts are highly ambiguous in Uyghur. We further categorize the ambiguous cases as follows.

Ambiguity between name and non-name. A Uyghur person or GPE name usually has a positive meaning, and thus it can also be a common word (usually noun or adjective) in different contexts. Table 2 shows some examples.

Table 1: Uyghur Suffix Derivation Examples

Uyghur	English
زامان zaman	era
زامانۋى zamanivi	modern
زامانۋىلاش zamanivilax	modernization
زامانۋىلاشتۇر zamanivilaxtur	modernize
زامانۋىلاشتۇرۇل zamanivilaxturur	be modernized
زامانۋىلاشتۇرۇلما zamanivilaxtururma	can not be modernized
زامانۋىلاشتۇرۇلمايمىز zamanivilaxtururmaimiz	We can not be modernized

Table 2: Ambiguous Uyghur Name Examples

Type	Name	Meaning as Non-Name
PER	ئارسلان	lion
	ئالىم	scientist
	روزا	fasting
	قەھرىمان	hero
	بەختيار	happy
	ھېيت قۇربان	Corban Festival
	ئىسلام	Islam
	گۈزەل	beatiful
	يولۋاس	tiger
	تۇردى	stand
	رەجەپ	July
	ئاينۇر	moonlight
	جىنەستە	cherry
	دىلدار	sweetheart
	رەيھان	violet
	نىگار	lover
	ئايتۇرۇك	Moon of the Turkish
GPE	ئادىل	fair
	ئارال	island (transliterated as 'Alar')
	دۆلەت	country (transliterated as person name 'Dolet')

Ambiguity across name types. There also exists a lot of ambiguity across name types. Table 3 presents some examples of name mentions indicating various types in different contexts.

To solve these problems, we will need to develop and exploit more language-specific resources such as title lists and tools such as dependency parser to capture wider contexts. This ambiguity problem also makes it difficult to transfer indicative contextual words in English name tagger to Uyghur. On average any English action verb or title has more than 10 possible translations in Uyghur. For example, the word "*watch*" has 41 possible translations in Uyghur.

Table 3: Ambiguity across name types Ambiguous

Name	Translation	Type1	Type2
سايرام	Sayram	LOC	PER
ئالىم	scientist	PER	Nominal
نەبىجان دۆلەت	Nebijan country	GPE	PER
ئارمان	ARMAN	PER	ORG
ئابىدە	ABIDE	PER	ORG
ئىخلاس	IHLAS	PER	ORG
جۇڭغار	Dzungaria	LOC	ORG

Table 4: Informal Names

Uyghur Name	Literal Translation	Referent Entity
تىرامپ ھۆكۈمىتى	Trump government	USA
شام دۆلىتى	candle country	Islamic State
قاراقاش	thick eyebrows	Karakax County
تىبەت	Tibetan	Tibet
شىنزۇ ئابى ھۆكۈمىتى	Abe Shinzo government	Japan government
ئاستانە	Capital	Astane County

3..4 Variety

Name variety. Names with different origins (Uyghur, Han Chinese, foreign names) have different characteristics. For example, a place name in Xinjiang can be transliterated either by Chinese pinyin or by its original Uyghur pronunciation. For example, both "*Hetian*" and "*Hotan*" refer to the same city that appear frequently in English news articles. There is no established standard yet for transliterating Uyghur names, which makes it difficult to project name gazetteers in high-resource languages such as English or Chinese onto Uyghur documents for name identification. After romanizing Uyghur, many foreign names look very similar to their English forms. For example, "*donald trumpni*" refers to "*Donald Trump*", "*nato ken*" refers to "*Naoto Kan*", and "*amerika*" refers to "*America*". Therefore we tried to use a Soundex based matching method to perform cross-lingual entity linking on each romanized Uyghur ngram to English Knowledge Base and English gazetteers in order to determine whether it's a name. However, this simple approach produced many spurious errors. In the future, it might be more effective to add it as an implicit feature in the model.

Unlike English, there is no capitalization for names in Uyghur. For Han Chinese person names, there is a fixed list of last names which are usually one single character, and each first name is usually a limited 1-2 character. However, neither of these two characteristics exists in Uyghur person names. Similar to Turkish, a Uyghur person's last name is his/her father's first name. Moreover, each first name or last name is usually a common word that carries some positive meaning, which yields an almost infinite number of combinations. Therefore it's more challenging to determine Uyghur person name boundaries than English, Chinese and Turkish.

Context Variety. We also attempted to project English word embeddings to Uyghur using a bilingual lexicon. However, the available lexicon has too low coverage to provide any gains. The same approach provided significant gains (up to 5%) for both Turkish and Uzbek name tagging.

3..5 Informal Names

Due to historical and cultural reasons, a substantial amount of informal Uyghur names is being created, especially at social media platforms. Table 4 presents some examples. Our model also missed some informal poster names and twitter users which don't appear in indicate contexts. Many of these names are common words such as سەپەر ("*travel*"). In addition, many Uighur people like to create pen names for themselves as their middle names, which are also difficult to identify.

3..6 Long Nested Organizations

Organization names in Uyghur texts are often very long containing nested names. Some challenging examples are presented in Table 5. It's difficult to determine their boundaries, especially when they are not linkable to external knowledge bases, or contain names which are also common words (e.g., the nested person name "*Arman*" means "*dream*"; and "*Abida*" means "*milestone*"). Addressing this challenge would require us to develop more advanced name internal structure parsing techniques.

3..7 Code Switch

Another unique challenge of Uyghur name tagging is the frequent code-switch phenomena in Uyghur texts. A large variety of names are borrowed from other languages, including Mandarin which is taught in most Uyghur schools (e.g., جۇڭگو ("*China*"), شىخەنزە ("*Shihezi*"), تاۋباۋ ("*Taobao*")), Arabic which is due to religious reasons (e.g., مۇھەممەت ("*Mohammed*")), English (e.g., ئامېرىكا ("*United States*"), تىرامپ ("*Trump*")) and Russian which are due to commercial trades. Extracting these names correctly requires us to identify their origins and capture the detailed characteristics on how they were transliterated.

3..8 Misspellings

Many names and contextual words in Uyghur texts in both formal and informal genres are misspelled. For example, the common person name دېڭ شياۋپىڭ (*Deng Xiaoping*) is often misspelled as دېڭ شياۋپېڭ even including its Wikipedia title; the Wikipedia title of جۇڭگو ("*China*") is also misspelled as جوڭگو; ئۈرۈمچى (*Urumqi*) is often misspelled as ئۇرۇمچى. Up to date there are no effective Uyghur spelling correction techniques available yet.

3..9 Name Definition and Annotation Challenges

In the past two decades, many efforts have been made at defining the name tagging task, including Message Understanding Conference (MUC) (Grishman and Sundheim, 1996), Automatic Content Extraction (ACE) [6], and Entity, Relation and Event (ERE) (Song et al., 2015). However, there are many open issues which may cause confusions for both human annotators and systems. In particular, for low-resource languages like Uyghur, it's also challenging to train native speakers to follow a long annotation guideline

[6] http://www.itl.nist.gov/iad/mig/tests/ace/

Table 5: Long Nested Organizations

Nested Organization	English Translation
[ORG مۇسۇلمانچە يىمەكلىك سانائىتى گۇرۇھى چەكلىك شىركىتى [PER ئارمان] [GPE شىنجاڭ]]	[ORG [GPE Xinjiang] [PER Arman] Halal Food Group Co., Ltd.]]
[ORG [ئابدەPER] بىئو پەن - تېخنىكا تەرەققىيات چەكلىك شىركىتى [GPE شىنجاڭ]	[ORG [GPE Xinjiang] [PER Abida] Biotechnology Development Co., Ltd.]
[ORG [ORG [GPE ئېكولوگىيەسى ۋە جۇغراپىيە تەتقىقات مەركىزىنىڭ [GPE شىنجاڭ]] [ORG جۇڭگو پەنلەر ئاكادېمىيەسى]]	[ORG [ORG [GPE Xinjiang] Institute of Ecology and Geography] [ORG Chinese Academy of Sciences]]
[ORG [ORG مۇخبىرلار پونكىتى [GPE شىنجاڭ]] [ORG شىنخۇا ئاگېنتلىقى]]	[ORG [ORG [GPE Xinjiang] Editorial Office] of [ORG Xinhua News Agency]]

(usually more than 20 pages) which may still leave many language-specific issues underspecified or unresolved. Two annotators at Linguistic Data Consortium (LDC) performed independent annotations on a subset of the LoreHLT2016 Uyghur name tagging data. Compared to the ground truth their F-scores were only 60.5% and 78.8% respectively. In the following, we will discuss some remaining gray areas which may still lead to confusions and different interpretations, and thus often it's difficult to draw a line. These problems are often amplified due to the communication barrier among the guideline developers, system developers and annotators.

- Adjective form and roles: Names in adjectival form, or modifiers, such as "*[GPE American] army*", are taggable. But this definition causes confusions because the modifiers do not always play geo-political entity roles (e.g., "*Chinese*" in "*[GPE Chinese] food*"). On the other hand, when news organization names refer to publications instead of organization roles, they should not be tagged. For example, in "*Bob enjoys reading the New York Times*", "*New York Times*" should not be tagged as an organization. Similarly, when a facility (e.g., "*White House*") plays an ORG role (e.g., make a statement), it should be tagged as ORG. Accurately determining these semantic roles requires further deep understanding of implicit contexts.
- Designator: it's often debatable whether GPE/LOC/ORG designators like "*city*" and "*company*" should be included in name mentions.
- Specific entity: Most guidelines indicate that names of deceased people, fictional characters, religious entities should all be tagged. In contrast generic persons are not taggable, such as "*Americans*", "*Christians*", "*Arabs*", "*Democrats*" and "*Republicans*".
- Group entity: When a GPE name is used to refer to the people of a GPE, it should not be tagged as a PER or GPE name. For example, in "*The Swiss have joined us on the bus tour*", "*Swiss*" should not be tagged. In contrast, a group of countries such as "*the European Union*" should be tagged as GPE.
- Entity subpart: a subpart of GPE that doesn't have a government (e.g., "*South America*", "*North America*", "*Middle East*", "*South Asia*") should be tagged as a LOC.
- Nominal mentions: when a nominal mention refers to a specific entity with rich context, both human annotators and systems tend to mistakenly label it as a name. For example, in the following sentence "*According to the report from China News Web, the telegraph from Xinhua Network's Bureau in Xinjiang stated that after the earthquake in Kiriye, four teams, consisting of the members from the local army, the civil official, the health department, the police, the fire department, the power supply and so on, have left for the affected Atchan township.* ", it's difficult to decide whether "*the health department*" and "*the fire department*" are names or nominals.

4. Conclusions and Future Work

We conducted a thorough study on both quantitative and qualitative analysis on a wide variety of errors from a state-of-the-art Uyghur name tagger. We also discussed possible solutions for the remaining challenges. Recently there is a trend in the community to push the rapid development of language universal techniques for name tagging (Zhang et al., 2016; Littell et al., 2016; Tsai et al., 2016; Pan et al., 2017). These methods have achieved some success at setting up baseline name taggers with reasonable performance. However, based on the Uyghur case study in this paper we can clearly see that most of the remaining challenges are specific to the target language, and thus we will need to embrace language-specific resources and knowledge in order to break the performance ceiling. We hope that the detailed analysis we did in this paper can shed a light on future efforts to focus on Uyghur resource development instead of simply borrowing language-independent features and machine learning methods which were used by other languages.

References

Arkin, M., Hamdulla, A., and Tursun, D. (2013a). Recognition of uyghur place names based on rules. *Communications Technology*, 7.

Arkin, M., Mahmut, A., and Hamdulla, A. (2013b). Person name recognition for uyghur using conditional random fields. *IJCSI International Journal of Computer Science Issues*, 10.

Arkin, M., Mahmut, A., and Hamdulla, A. (2013c). Person name recognition for uyghur using conditional random fields. *International Journal of Computer Science Issues*, 10.

Creutz, M. (2003). Unsupervised segmentation of words using prior distributions of morph length and frequency. In *Proc. ACL2003*.

Matthew S. Dryer et al., editors. (2013). *WALS Online*.

Engesæth, T., Yakup, M., and Dwyer, A. (2009). *Greetings from the Teklimakan: a Handbook of Modern Uyghur*.

Grishman, R. and Sundheim, B. (1996). Message understanding conference-6: A brief history. In *Proceedings of COLING1996*.

Martin Haspelmath, et al., editors. (2005). *World Atlas of Language Structures*.

Lample, G., Ballesteros, M., Kawakami, K., Subramanian, S., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proc. the 2016 Conference of the North American Chapter of the Association for Computational Linguistics —Human Language Technologies (NAACL-HLT 2016)*.

Li, J., Liu, K., Aili, M., Lv, Y., Liu, Q., and Yibulayin, T. (2011). Recognition and translation for chinese names in uyghur language. *Journal OF Chinese Information Processing*, 25.

Li, J. (2014). Research on uyghur named entity recognition and translation.

Littell, P., Goyal, K., Mortensen, R. D., Little, A., Dyer, C., and Levin, L. (2016). Named entity recognition for linguistic rapid response in low-resource languages: Sorani kurdish and tajik. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*.

Maihefureti, Rouzi, M., Aili, M., and Yibulayin, T. (2014). Uyghur organization name recognition based on syntactic and semantic knowledge. *Computer Engineering and Design*, 35.

Nizamidin, T., Tuerxun, P., Hamdulla, A., and Arkin, M. (2016). A survey of uyghur person name recognition. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 9.

Pan, X., Cassidy, T., Hermjakob, U., Ji, H., and Knight, K. (2015). Unsupervised entity linking with abstract meaning representation. In *Proc. the 2015 Conference of the North American Chapter of the Association for Computational Linguistics —Human Language Technologies (NAACL-HLT 2015)*.

Pan, X., Zhang, B., May, J., Nothman, J., Knight, K., and Ji, H. (2017). Cross-lingual name tagging and linking for 282 languages. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics (ACL2017)*.

Rozi, A., ZONG, C., Mamateli, G., Mahmut, R., and Hamdulla, A. (2013). Approach to recognizing uyghur names based on conditional random fields. *Journal of Tsinghua University Science and Technology*, 53.

Song, Z., Bies, A., Strassel, S., Riese, T., Mott, J., Ellis, J., Wright, J., Kulick, S., Ryant, N., and Ma, X. (2015). From light to rich ere: Annotation of entities, relations, and events. In *Proceedings of the 3rd Workshop on EVENTS at the NAACL-HLT 2015*.

(translated by Anne Lee), H. T. (2003). *Modern Uyghur Grammar (Morphology)*.

Tsai, C.-T., Mayhew, S., and Roth, D. (2016). Cross-lingual named entity recognition via wikification. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*.

Turhun, N., Yu, H., and Li, Y. (2012). Research on crf based uyghur named entity recognition. *Consumer Electronics Magazine*, 7.

Yu, G., Xu, Y., Ma, M., Yu, Z., and Arkin, M. (2015). Research on rule based uyghur named entity recognition. *China Science and Technology Panorama Magazine*, 15.

Zakir, H. A. (2007). *Introduction to Modern Uighur*.

Zhang, L., Yang, Y., Mi, C., and Li, X. (2015). Recognition and translation of uyghur named entities in numerals class. *Computer Applications and Software*, 32.

Zhang, B., Pan, X., Wang, T., Vaswani, A., Ji, H., Knight, K., and Marcu, D. (2016). Name tagging for low-resource incident languages based on expectation-driven learning. In *Proc. the 2016 Conference of the North American Chapter of the Association for Computational Linguistics —Human Language Technologies (NAACL-HLT 2016)*.

Zhang, L. (2014). Recognition and translation of uyghur name entities for chinese-uyghur machine translation.

BiLSTM-CRF for Persian Named-Entity Recognition

ArmanPersoNERCorpus: the First Entity-Annotated Persian Dataset

Hanieh Poostchi[a,b], Ehsan Zare Borzeshi[b], Massimo Piccardi[a]
[a] University of Technology Sydney, [b] Capital Markets CRC
PO Box 123 Broadway NSW 2007, Australia; 55 Harrington Street The Rocks NSW 2000, Australia
{hpoostchi,ezborzeshi}@cmcrc.com, massimo.piccardi@uts.edu.au

Abstract

Named-entity recognition (NER) can still be regarded as work in progress for a number of Asian languages due to the scarcity of annotated corpora. For this reason, with this paper we publicly release an entity-annotated Persian dataset and we present a performing approach for Persian NER based on a deep learning architecture. In addition to the entity-annotated dataset, we release a number of word embeddings (including GloVe, skip-gram, CBOW and Hellinger PCA) trained on a sizable collation of Persian text. The combination of the deep learning architecture (a BiLSTM-CRF) and the pre-trained word embeddings has allowed us to achieve a 77.45% CoNLL $F1$ score, a result that is more than 12 percentage points higher than the best previous result and interesting in absolute terms.
Keywords: Named-entity recognition, recurrent neural networks, BiLSTM-CRF, Persian language, low-resource languages.

1. Introduction

Named-entity recognition (NER) is a natural language processing component that aims to identify all the "named entities" (NEs) such as names of people, locations, organisations and numerical expressions in an unstructured text. This information can be useful in its own right or facilitate higher-level NLP tasks such as text summarization and machine translation. To date, NER research has mostly focussed on languages with a high number of digitally annotated resources such as English and German (Tjong Kim Sang and De Meulder, 2003) and Spanish and Dutch (Tjong Kim Sang, 2002). The main reason why many other languages, including many from the Asian region, have received less attention is the significant scarcity of public, annotated digital resources. Amongst those, the Persian language is spoken by more than 110 million speakers world-wide and has more than 570K articles on Wikipedia. However, it has been rarely studied for NER (Khormuji and Bazrafkan, 2014) or even just text processing (Shamsfard, 2011).

Although language-agnostic NER systems such as Polyglot-NER (Al-Rfou et al., 2015) exist, their performance is generally not competitive in comparison to language-specific NER. For this reason, in our previous work (Poostchi et al., 2016) we developed a dedicated NER system for Persian[1]. Its development was supported by two datasets: a) a sizable unannotated dataset of Persian sentences for training word embeddings, and b) an entity-annotated dataset for training named-entity classifiers.

This paper makes three distinct contributions: 1) it officially releases the entity-annotated dataset with an ISLRN[2] that should make its utilisation easier; 2) it releases four different word embeddings trained on the unannotated resources for a comprehensive Persian dictionary of nearly 50K unique words, also available via an ISLRN[3]; 3) it proposes a deep learning Persian NER based on a state-of-the-art architecture, the BiLSTM-CRF (Huang et al., 2015; Lample et al., 2016). Thanks to this architecture and the trained word embeddings, we have been able to achieve an improvement of over 12 percentage points of CoNLL $F1$ score over our previous approach based on structural SVM (Poostchi et al., 2016).

2. Supervised and Unsupervised Datasets for Persian NER

Supervised NER usually involves two main steps: the unsupervised training of a word embedding from a large corpus, and the classification of named entities using an annotated dataset. This section describes the two datasets that we provide for NER in the Persian language.

2.1. The Unannotated Persian Corpus

An effective co-occurrence matrix can be calculated from an adequately large corpus of documents covering a range of contexts. To this end, we have collated three resources of Persian text: i) the *Leipzig corpora* (Goldhahn et al., 2012) with approximately 1M and 300K sentences from news websites and Wikipedia, respectively, ii) a subset of *VOA news* with 227K sentences[4], and iii) the *Persian Dependency Treebank* (Rasooli et al., 2013) with nearly 30K sentences.

The aggregated corpus, with a total number of sentences in excess of 1.6M, has gone through a pipeline of text normalisation and tokenisation (Feely et al., 2014) tools including PrePer (Seraji, 2013), the Farsi verb tokenizer

[1] Particularly, Western/Iranian Persian which is also known as Farsi.

[2] ISLRN: 399-379-640-828-6

[3] ISLRN: 921-509-141-609-6

[4] http://www.ling.ohio-state.edu/~jonsafari/corpora/index.html#persian

(Manshadi, 2013), SetPer (Seraji et al., 2012) and tok-tok (Dehdari, 2015). We refer the reader to (Poostchi et al., 2016) for more details.

After normalisation, we have trained four different word embeddings using the provided corpus. The methods are GloVe (Pennington et al., 2014), word2vec (both skip-grams and continuous bag of words) (Mikolov et al., 2013), and Hellinger PCA (HPCA) (Lebret and Collobert, 2014); they will be briefly explained in Section 3.1.. For every method, a window of size 5 in both directions was used to calculate the co-occurrence matrix. Then, only words with a minimum frequency of 15 were selected, resulting in a dictionary of $49,902$ distinct words. The length of the embedding vectors was set to 300. All these hyper-parameters were chosen empirically during an initial evaluation.

The collated corpus cannot be publicly released due to licensing restrictions on some of its parts. However, we have released all four word embeddings on GitHub[5], which will allow easy replication of our experiments.

2.2. The Entity-Annotated Persian Dataset

To create a Persian named-entity dataset, we have selected a subset of 7,682 news sentences from the **BijanKhan** (Bijankhan et al., 2011) corpus, which is the most-established POS tagged Persian corpus. The histogram of the sentences' length is shown in figure 1. The mode is around 24 words per sentence, but with a significant tail of longer sentences. An experienced annotator has led the annotation task and prepared a comprehensive instruction manual based on the definition of Sekine's extended named entities (Sekine, 2007). The annotation of the dataset was split over two native-speaking post-graduate students and disambiguated according to the context. For instance, word *Ferdowsi* has different labels in *"Ferdowsi$_{B\text{-}ORG}$ University$_{I\text{-}ORG}$"* and *"Ferdowsi$_{B\text{-}PER}$, the great epic poet"*. To evaluate the accuracy of the annotation, three other independent native-speaking reviewers have verified i) a random sample of 500 annotated NEs, and ii) a sample of 500 annotated NEs from the two most semantically-close classes (i.e., location and organisation). The percentages of corrections have only been 1.8% and 1.9%, respectively.

The annotated dataset, called **ArmanPersoNERCorpus**, contains a total of 250,015 tokens with a hit rate of 87.68% in the trained dictionary. Only 11.08% of the tokens are marked as part of entities (Poostchi et al., 2016). The NEs are annotated in IOB format and categorised into six pre-defined classes: *person*, *organisation*, *location*, *facility*, *product*, and *event*. More than 60% of the sentences have at least one entity of any type. Figure 2 shows the percentage of sentences containing at least one entity and a maximum between 1 and 7 entities of each particular class. The most frequent NE class is *organisation* with an appearance rate of more than 33% of the sentences. This

[5] https://github.com/HaniehP/PersianNER

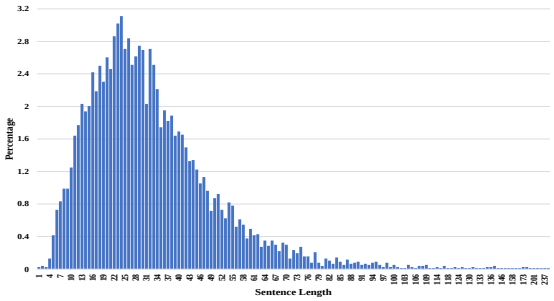


Figure 1: Histogram of the sentences' length in ArmanPersoNERCorpus.

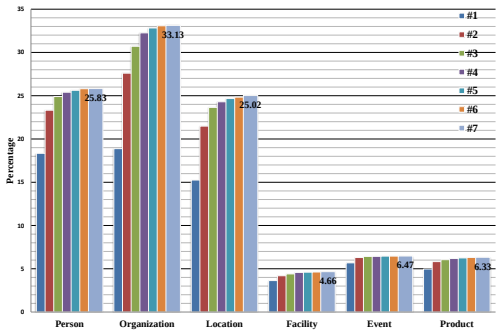


Figure 2: Percentage of sentences containing at least one entity and a maximum between 1 and 7 entities (in left-to-right order in the plot) of each particular named-entity class.

is followed by *person* and *location* with more than 25%. *Event* and *product* are far less frequent with just over 6% and *facility* has the lowest frequency with about 4%.

The dataset has been submitted to LR-MAP for global unique identification by an ISLRN. It is stored on GitHub and organised in the same 3 folds that we have used for the experiments. In addition to NER training, it could find use as an evaluation dataset for NER systems trained on silver standards.

3. Methods

Supervised NER is split into an initial step of word embedding followed by a step of token-level classification of the named entities. In this section we briefly describe the methods employed.

3.1. Word Embedding

A word embedding maps distinct words to high-dimensional feature vectors. GloVe (Pennington et al., 2014), word2vec (Mikolov et al., 2013), and Hellinger

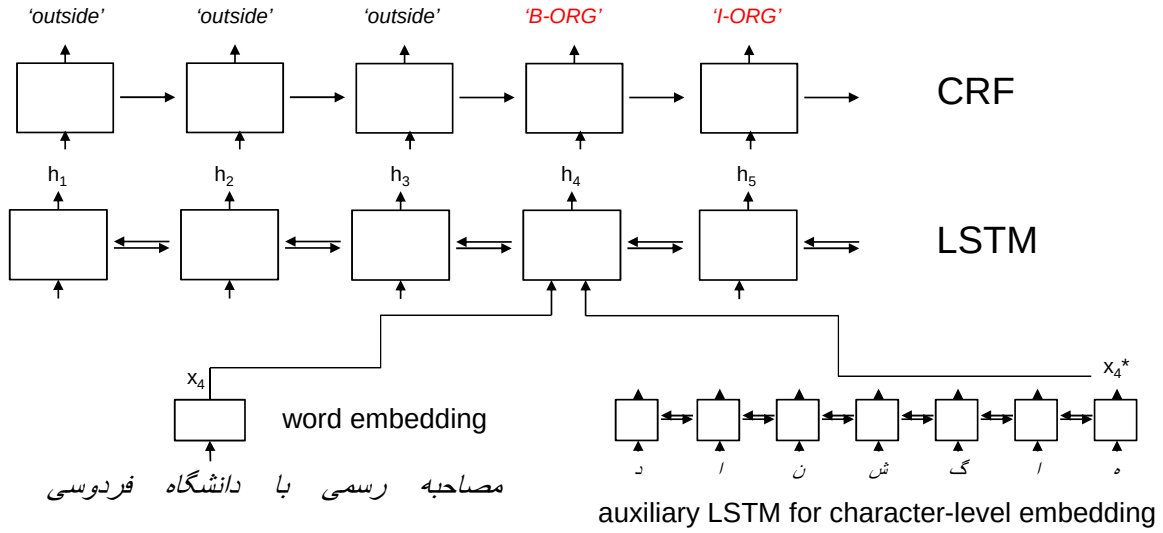


Figure 3: A diagram of the BiLSTM-CRF with an example of prediction. The input is a Persian sentence that consists of 5 tokens and translates into English as "an official interview with Ferdowsi University". The sentence is displayed in right-to-left order since this is how it would appear in Persian writing. However, this is not important for processing since both the tokens and their characters are processed in both directions. Token "University" is the 4-th token and its word embedding is noted as x_4 in the diagram. Its character-level embedding is the last output of the auxiliary LSTM and is noted as x_4^*. These embeddings are concatenated and used as the input of the corresponding slot in the main LSTM. In turn, the output of the LSTM slot is noted as h_4 and used as input for the CRF. Eventually, the CRF slot emits prediction "B-ORG". Token "Ferdowsi" is the 5-th token in the sentence and is predicted as "I-ORG".

PCA (HPCA) (Lebret and Collobert, 2014) are well-known examples of unsupervised word embeddings used successfully for NER.

GloVe is a global log-bilinear regression model with a weighted least-square objective that combines advantages of global matrix factorization and local context windows. The training objective of GloVe is to learn word vectors such that their dot product equals the logarithm of the words' probability of co-occurrence.

Word2vec is a generative model for continuous representations of words that preserves the linear regularities amongst words. This model has two variants described hereafter: 1) the *skip-gram* model aims to learn word vector representations that are useful for predicting the nearby words in a sentence. A shallow neural network consisting of an input projection layer, an output layer and a softmax activation is trained to maximize the average of the log probability of a context word surrounding a given word; 2) the *continuous bag of words (CBOW)* model is similar to the skip-gram except that the roles of the input and output are reversed: in this model, the probability of the current word given the context is explicitly estimated.

HPCA is a simple spectral method analogous to PCA. First, the co-occurrence matrix is normalised row-by-row to represent the words by proper discrete probability distributions. Then, the resulting matrix is transformed into Hellinger space before applying PCA to reduce its dimensionality.

3.2. The BiLSTM-CRF for Sequential Labelling

The BiLSTM-CRF is a recurrent neural network obtained from the combination of a long short-term memory (LSTM) and a conditional random field (CRF) (Huang et al., 2015; Lample et al., 2016). The LSTM is used first to process each sentence token-by-token and produce an intermediate representation. Then, this intermediate representation is used as input for the CRF to provide the prediction of all the tokens' labels. The two models enjoy complementary features: as a complex, nonlinear model, the LSTM is able to effectively capture the sequential relationships amongst the input tokens; at its turn, the CRF permits optimal, joint prediction of all the labels in the sentence, capturing the relationships at label level. The "bi" in the name stands for "bidirectional" and alludes to the fact that the LSTM processes each sentence in both left-to-right and right-to-left order to embed the sequential dependencies in both directions. Before being processed, each token needs to be converted to a high-dimensional numerical vector, and this embedding is learned automatically alongside all the other parameters as part of the training stage. Eventually, the network also includes a second, auxiliary LSTM that further encodes each token character-by-character to capture the regularities at character level. Prior to being processed, also the individual characters need to be mapped to numerical embeddings. Figure 3 shows a complete diagram of the BiLSTM-CRF with an ample caption describing all the

Methods	Entities						
	Person	**Organization**	**Location**	**Facility**	**Event**	**Product**	**Overall**
CRF (HPCA)	64.10	42.25	57.97	41.09	22.48	20.00	49.92
Jordan-RNN (HPCA)	72.13	57.28	62.70	51.92	39.79	42.08	60.52
SVM-HMM (HPCA)	75.65	61.59	66.67	61.20	52.58	41.37	65.13
BiLSTM-CRF (HPCA)	77.69	69.70	69.67	57.33	52.69	49.24	69.43
BiLSTM-CRF (CBOW)	87.32	74.84	76.06	66.38	56.93	55.06	76.19
BiLSTM-CRF (GloVe)	86.97	75.73	76.62	67.41	55.58	55.08	76.58
BiLSTM-CRF (Skip-Gram)	**88.18**	**76.03**	**76.94**	**70.47**	**60.12**	**55.69**	**77.45**

Table 1: Comparison of Persian NER results with different classifiers and word embeddings (by class and as overall micro-average). The results above the double horizontal line are from (Poostchi et al., 2016) and are based on the same data and splits.

main variables and components (the character embeddings have been omitted to avoid cluttering).
Given a training set of labelled sequences, $\{x_i, y_i\}, i = 1 \dots N$, where x denotes a sequence of tokens and y the sequence of their labels, the BiLSTM-CRF is trained by maximizing the conditional log-likelihood:

$$\bar{w} = \underset{w}{\operatorname{argmax}} \sum_{i=1}^{N} \ln p(y_i|x_i, w) \quad (1)$$

where w denotes all the model's parameters including the transition weights of the CRF, the weights of the main and auxiliary LSTMs, and the token and character embeddings. Once the model is trained, inference for a new sentence x is obtained as:

$$\bar{y} = \underset{y}{\operatorname{argmax}}\, p(y|x, \bar{w}) \quad (2)$$

by propagating x through the network and applying the Viterbi algorithm at the CRF output layer.

4. Experimental Results

In this section, we present the NER results obtained with the BiLSTM-CRF and the different word embedding and we compare them with those reported in (Poostchi et al., 2016). For the experiments, we have used a TensorFlow implementation of the BiLSTM-CRF [6] (Dernoncourt et al., 2017), running each training session for 80 epochs (a value where the validation accuracy always seemed to have stabilised). For processing, all digits have been replaced with 0s. All hyper-parameters have been left to their default values.

Table 1 shows a comparison of the CoNLL $F1$ scores (by class and as overall micro-average) over the NER task for the various classifiers. The CoNLL $F1$ score is a strict version of the standard $F1$ score where a true positive is scored only if all the tokens of a given named entity are classified correctly (including their B- and I- tags). Conversely, every incorrect B- prediction is counted as a false positive. All the experiments have been performed with three-fold cross validation, using each of the three folds in turn as the test set and the other two for training. Moreover, each experiment has been repeated three times to mollify the effects of the random initialisation of the network's weights. This means that the values reported in Table 1 for our system are the average of $3 \times 3 = 9$ runs.

As shown in Table 1, the scores achieved by the BiLSTM-CRF have been higher than any of the results previously presented in (Poostchi et al., 2016). The results with the different word embeddings have ranged from a minimum average of 69.43% $F1$ score with HPCA to a maximum of 77.45% with the skip-gram. This relative ranking seems in good accordance with other NER results from the literature (Huang et al., 2015; Ma and Hovy, 2016). Amongst the classes, *person* is clearly the easiest and *product* the most challenging. This could be explained by the fact that the latter has much fewer samples (6% vs 25% of sentences) or that its patterns are possibly more diverse and harder to learn. In all cases, the proposed system has managed to outperform the best previous results for all classes and by a remarkable 12.32 $F1$ score percentage points on average.

5. Conclusion

In this paper, we have presented an approach for Persian NER based on a deep learning architecture and released a Persian annotated corpus alongside four different Persian word embeddings based on GloVe, CBOW, skip-gram and HPCA. The proposed approach has achieved an average $F1$ score of 77.45% which, to the best of our knowledge, is the highest Persian NER $F1$ score reported in the literature by 12.32 percentage points over the previous best result. Moreover, in addition to NER, the released word embeddings could find future use in other Persian NLP tasks including translation, question answering and summarisation.

6. Acknowledgements

This research has been funded by the Capital Markets Cooperative Research Centre (CMCRC).

[6] https://github.com/Franck-Dernoncourt/NeuroNER

7. Bibliographical References

Al-Rfou, R., Kulkarni, V., Perozzi, B., and Skiena, S. (2015). Polyglot-ner: Massive multilingual named entity recognition. In *Proceedings of 2015 SIAM International Conference on Data Mining*.

Dehdari, J. (2015). A fast, simple, multilingual tokenizer. `https://github.com/jonsafari/tok-tok/`.

Dernoncourt, F., Lee, J. Y., and Szolovits, P. (2017). NeuroNER: an easy-to-use program for named-entity recognition based on neural networks. *Conference on Empirical Methods on Natural Language Processing (EMNLP)*.

Feely, W., Manshadi, M., Frederking, R. E., and Levin, L. S. (2014). The cmu metal farsi nlp approach. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 4052–4055, Reykjavik, Iceland.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Khormuji, M. K. and Bazrafkan, M. (2014). Persian named entity recognition based with local filters. *International Journal of Computer Applications*, 100(4):1–6.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Lebret, C. D., and Collobert, R. (2016). Neural architectures for named entity recognition. In *The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HCT-NAACL)*, pages 260–270, San Diego, California, USA.

Lebret, R. and Collobert, R. (2014). Word embedding through hellinger pca. In *The 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 482–490, Gothenburg, Sweden.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany, August. Association for Computational Linguistics.

Manshadi, M. (2013). Farsi verb tokenizer. `https://github.com/mehdi-manshadi/Farsi-Verb-Tokenizer/`.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, Lake Tahoe, Nevada, USA.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar.

Poostchi, H., Borzeshi, E. Z., Abdous, M., and Piccardi, M. (2016). Personer: Persian named-entity recognition. In *The 26'th International Conference on Computational Linguistics (COLING)*, pages 3381–3389, Osaka, Japan.

Sekine, S. (2007). The definition of sekine's extended named entities. `http://nlp.cs.nyu.edu/ene/version7_1_0Beng.html`. New York University.

Seraji, M., BeÃ¡ta, M., and Joakim, N. (2012). A basic language resource kit for persian. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 2245–2252, Istanbul, Turkey.

Seraji, M. (2013). Preper: A pre-processor for persian. In *Proceedings of Fifth International Conference on Iranian Linguistics*, Bamberg, Germany.

Shamsfard, M. (2011). Challenges and open problems in persian text processing. *Proceedings of LTC*, 11.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 142–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tjong Kim Sang, E. F. (2002). Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

8. Language Resource References

Bijankhan, M., Sheykhzadegan, J., Bahrani, M., and Ghayoomi, M. (2011). Lessons from building a persian written corpus: Peykare. *Language resources and evaluation*, 45(2):143–164.

Goldhahn, D., Eckart, T., and Quasthoff, U. (2012). Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *The 8th International Language Ressources and Evaluation (LREC'12)*, pages 759–765, Istanbul, Turkey.

Rasooli, M. S., Kouhestani, M., and Moloodi, A. (2013). Development of a persian syntactic dependency treebank. In *The 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HCT-NAACL)*, pages 306–314, Atlanta, USA.

Data Anonymization for Requirements Quality Analysis: a Reproducible Automatic Error Detection Task

Juyeon Kang[†*] **Jungyeul Park**[‡*]

[†] PROMETIL, 52 rue Jacques Babinet. 31100 Toulouse, France.
j.kang@prometil.com
[‡] CONJECTO, 74 rue de Paris. 35000 Rennes, France.
jungyeul@conjecto.com

Abstract

In this work, we aim at identifying potential problems of ambiguity, completeness, conformity, singularity and readability in system and software requirements specifications. Those problems arise particularly when they are written in a natural language. While we describe them from a linguistic point of view, the business impacts of each potential error are also considered in system engineering context. We investigate and explore error patterns for requirements quality analysis by manually analyzing the corpus. This analysis is based on the requirements grammar that we developed in our previous work. In addition, this paper extends our previous work in a two-fold way: (1) we increase more than twice the number of evaluation data (1K sentences) through a manual verification process, and (2) we anonymize all sensible and confidential entities in evaluation data to make our data publicly available. We also provide the baseline system using conditional random fields for requirements quality analysis, and we obtain 79.47% for the F_1 score on proposed evaluation data.
Keywords: Requirements quality analysis (ReQA), error detection in ReQA, data anonymization, reproducible task

1. Introduction

Among technical documents, requirements are a central issue since they must comply with a high number of constraints of e.g. readability, lack of ambiguity and implicit data, feasibility, relevance, traceability, conformity and overall cohesion and coherence (Firesmith, 2003; Alred et al., 2011). For example, *Ambient pressure shall be permanently maintained* relies too much on the operator's knowledge and practice: what pressure should be maintained to be ambient and what to do in case of interruption? A wrong interpretation may lead to accidents and damages.

There exists different types of references for guiding the writing of high quality requirements. Among them, we can cite:

- the standards of IEEE (ISO/IEC/IEEE29148:2011),
- ARP4754A (Aerospace Recommended Practice),
- the recommendations of INCOSE (Guide for Writing Requirements), and
- IREB (International Requirements Engineering Board)

We can also refer to the principles of controlled natural languages, mainly, those defined in ASD-STE100 (Simplified Technical English by Aerospace and Defense Industries Association of Europe). Those documents show that requirements must be non-ambiguous consistent, correct, complete, verifiable, singular, readable, feasible and traceable. However, when writing or revising a set of requirements, it is particularly challenging to make sure that texts read easily and are unambiguous for any domain actor (Weiss, 1990; Grady, 2013). There are several factors like overload, time missing, novices and specific domain knowledge needed. Tools controlling the authoring quality of requirements can be useful for automatically proofreading large quantities of requirements. Since we presented the very first linguistic model for requirements quality analysis (ReQA) (Kang and Park, 2016), there are many requests for releasing data. While we build training data from the public domain corpus (Baroni et al., 2009) using an automatic method to boost the overall learning ability, we extract sentences from the actual documents for requirements quality and annotate manually error labels for evaluation data. Most documents of requirements often contain restricted information, and documents that we used for evaluation data are also confidential in which we cannot directly release them. Therefore, in this paper, we perform the data anonymization process for ReQA evaluation data, and we release them for a reproducible automatic error detection task. Training and anonymized evaluation data are available at `https://github.com/jungyeul/rqa`.

2. Errors in ReQA

The requirements grammar and error patterns have been developed in Kang and Park (2016) to generate different models of requirements quality checking system. It was the first work to show the quantitative approach for ReQA including building training and evaluation data. The objective of our system is to verify the conformity of natural language requirements regarding to the criteria for high quality requirements as defined in the IEEE standards and the recommendations of INCOSE. Among the criteria for writing good requirements, we focus on the following five constraints: non-ambiguity, completeness, readability, conformity and singularity. These constraints can be checked by lexical and syntactic rules. For example, the Completeness concerns the use of passive voice and some incomplete expressions like *TBC* (To Be Confirmed) and *etc*. And the Singularity is not respected when a requirement contains too many combinators *and*. Each constraint is so checked

[*] Both authors contributed equally to this work.

when a requirement contains one of the lexical or syntactic errors as listed in Figure 1.

3. Anonymization

Among technical documents, Software and system Requirements Specifications (SRS) contain highly confidential data, so that the information security must be guaranteed. For example, a SRS of a vehicle includes software, hardware and networking subsystems that make up the total system. These documents specify key performance parameters (such as operations speed, response time, availability, portability and maintainability), functional capabilities, data structure and elements, safety, constraints, etc. The loss of this information can lead to significant financial losses and reputational damage. Thus, the accurate data anonymization allows to preserve the confidentiality of requirements documents and makes them useful as a linguistic corpus for a Natural Language Processing (NLP) task like automatic error detection.

Different approaches for the data anonymization were studied and applied mainly in preserving privacy from social media and health information from medical data. In social network sites (SNS), web pages are designed for human interaction like sharing opinions, experiences and feeling between unknown users of any domain. Massive data flow in SNS contains personal information (birthday, email address, social relationship, friendship, etc.) which should be protected. For the purpose of preserving such a sensitive data, some studies focus on anonymization techniques of those data. Among anonymization techniques and algorithms developed for the privacy protection on relational data, Fredj et al. (2015) describes an approach based on the generalization in order that the illustration of different generalization algorithms helps data publishers selecting an adequate technique for each data. Zhou et al. (2008) presents privacy information modeling methods along with the state-of-art anonymization approaches: clustering-based approach and graph modification approach. In clinical research, the health data anonymization is required for the protection of patient records. Dernoncourt et al. (2017) proposes a de-identification system of health information, referred to as protected health information (PHI), namely age, contact, date, ID, location, name, profession. Based on Artificial Neural Networks (ANN) and Conditional Random Fields (CRF) models, they obtained F_1-scores of 99.229% and 99.023%, respectively, on MIMIC de-identification datasets, showing that the ANN model outperforms the CRF model on all types of datasets on which the system is tested. In this work, we propose to anonymize sensitive elements that may result in the problem of confidentiality in using requirements documents. It is first necessary to identify confidential information. For that purpose, we manually analyzed 15 SRSs with more than 4000 requirements of different types and domains coming from several companies and organizations.

Figure 2 shows the most commonly appearing sensitive data in requirements documents: the named entities, quantitative values, parametric values and references. Named entities expose domain specific terms and acronyms proper to a specific project. Domain specific sensitive terms, due to their specificity, provide more information than common terms. These terms have significant meaning in a specific domain. For example, *ice condition* can be referred to as a meteorological term in a common sense, but in terms of the aviation safety, *ICE CONDITION* indicates a specific status waiting an action. An acronym ACU can be referred to as *Australian Catholic University* or *A confirmer ultérieusement* in a French SRS, but in a more specific context, ACU (Air Control Unit) is related to air flow controller which gives information on its used domain. Quantitative values disclose direct information about key performance parameters while parametric values do not directly expose parameters. However, it is possible that the latter may be indicative of different parameters. Concerning the references, some of standards and norms might be public but many of requirements also refer to internal guidelines. Thus, it will be reassuring to take them into account. Once identified the most sensitive elements to anonymize, we manually made up a dictionary of entities including domain specific terms, acronyms and parametric values. Based on this predefined dictionary, we replace original data by randomizing the characters of entities and numeric values. The dictionary of entities and anonymized data that is used in this work as evaluation data set was validated by an expert in Requirements Engineering in terms of data security and utility. During our survey, we could not find contributions applied to the data anonymization of requirements, mostly because the requirements are confidential texts which are not intended to be publicly released. Therefore, this paper focuses on the identification of sensitive and confidential information from the requirements documents.

4. ReQA Data

4.1. Training data

We already detailed how to build training data by using part-of-speech (POS) tagging and syntactic parsing in Kang and Park (2016). Since we use the automatic method to build the training data, we minimize the POS label errors and parsing results by introducing the filtering method. We use the consensus results $\hat{\mathcal{D}}$ by intersection between two results using $\mathcal{D}(\mathcal{M}_1) \cap \mathcal{D}(\mathcal{M}_2)$ where $\mathcal{D}$ is raw text data, $\mathcal{M}_i$ is a learning algorithm to annotate raw text data, and $\hat{\mathcal{D}}$ is filtered annotated data. For POS tagging, we use Hidden Markov Model (HMM) and CRF learned labeling models as described in Brants (2000) and Lavergne et al. (2010). For dependency syntactic parsing, we use two pre-trained dependency parsing models of MaltParser (Nivre et al., 2006), which uses Support vector machines (SVMs) with a polynomial kernel and linear SVMs. For raw text data, we use the first part of ukWaC, one of the WaCky corpora (Baroni et al., 2009). After consensus filtering for POS tagging and syntactic parsing, we assign five types of errors: non-ambiguity, conformity, completeness, singularity and readability. Figure 3, presented in Kang and Park (2016), shows an example sentence from our final training data. In this figure, first, *analytes or investigations* represents the ambiguity error because of *or* as explained in the Figure 1. Second, *shall be selected* is annotated as completeness error because the information about who realize the required action is not specified. Third, *their clinical relevance* also has

Singularity	Combinators: (i) `X and X'` where POS (part-of-speech) labels (or phrase type) of X and X' are same. X= verb (infinitive form), verb phrase, noun, noun phrase, adjective, value followed by a unit of measurement
Completeness	Passive construction: (i) `modal(shall) be {AdvP} Verb (Action, pp)`: *shall be used*, *shall be properly used* Pronouns: (i) `Pron (possessive) Noun`: *their application*; (ii) `Pron (possessive) NP`: *their proper development*; (iii) `Pron (demonstrative) modal(shall)`: *this shall*, *these shall*
Conformity	Negations: (i) `modal Neg`: *shall not* Modals: (i) `modal {AdvP} Verb(Action, inf)`: *would implement*, *should correctly implement*
Readability	Lexical items: quantifiers (e.g. *each*, *some*, *all*) and acronyms (e.g. *ACR*, *CPU*)
Ambiguity	Combinators: (i) `X or X'` where POS labels (or phrase types) of X and X' are same. X= verb (infinitive form), verb phrase, noun, noun phrase, adjective, value followed by a unit of measurement. Lexical items: confusing terms, vague adjectives, adverbs

Figure 1: Five constraints and their errors patterns

Named entities	Domain specific terms: *ICE CONDITION, Torque motor, Pre-cooler* Acronyms: *EMCU, MEK, AMMC, JSC, ECU*
Parametric values	*ASTM-G-3769, 5V_UPS, @TM_REACH_DEFAULT*
Quantitative values	any numeric symbols followed or preceded by units of measurement
References	Standards/Norms: *MIL-STD-753C, MT1, DO-160F* Requirements ID: *MCS-QQ-P-47*

Figure 2: Sensitive elements in requirements documents

The	DT	O
analytes	NNS	B-AMBI
or	CC	I-AMBI
investigations	NNS	I-AMBI
covered	VBN	O
by	IN	O
the	DT	O
Scheme	NNP	O
shall	MD	B-COMP
be	VB	I-COMP
selected	VBN	I-COMP
on	IN	O
the	DT	O
basis	NN	O
of	IN	O
their	PRP$	B-COMP
clinical	JJ	I-COMP
relevance	NN	I-COMP
.	.	O

Figure 3: An example sentence from training data.

	sentences	tokens
train	35,826	590,886
dev	3,980	64,769
eval	988	23,076

Table 1: Number of tokens and sentences

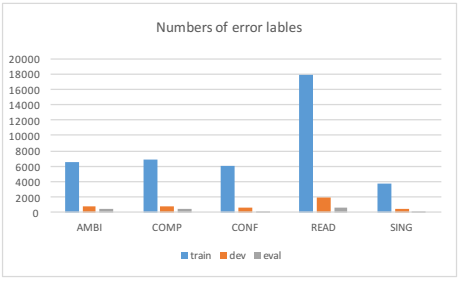


Figure 4: Number of annotated error labels

the completeness error because of the possessive pronoun *their*. It probably refers to one of the following antecedents: *analystes*, *investigations*, *the Scheme* but we need extra information to correctly identify the reference of *their*.

According to the standards IEEE (29148:2011), a requirement should be concise and a single sentence. In this paper, we add a *new* constraint $10 \leq n \leq 80$ where n is the length of sentences. Note that Kang and Park (2016) doesn't have the length constraint, and it yields the smaller data size. Then, we split 90-10 ratio for training and development data sets. Table 1 shows the number of tokens and sentences, and Figure 4 shows the number of annotated error labels in the current data sets.

4.2. Evaluation data

Among the SRSs documents that we have mentioned in §3., we extracted requirements which well represent four types of sensitive elements (Figure 2). Previously, evaluation data was composed of 319 technical requirements (481 sentences with 10,324 tokens) (Kang and Park, 2016). We extend the previous evaluation data to almost 1,000 sentences for a new evaluation data set, and they are all anonymized for data distribution as described in §3.. Based on what we define anonymization entities, we manually make up a dictionary of anonymization entities (over 270 entities) extracted from the original requirements of the extended evaluation data, and we randomize their alphanumeric characters. Figure 5 shows an example sentence of anonymized evaluation data. We add `#annonymized` for anonymized entities. `JKS`, `T2` and `E99/J04DR` are origi-

The	DT	O	
JKS	NN	O	#annonymized
internal	JJ	O	
transducer	NN	O	
T2	NN	O	#annonymized
shall	MD	O	
correspond	VB	O	
to	TO	O	
a	DT	B-READ	
E99/J04DR	NN	O	#annonymized
in	IN	O	
terms	NNS	O	
of	IN	O	
material	NN	O	
and	CC	O	
pin	NN	O	
configuration	NN	O	
.	.	O	

Figure 5: An example sentence from anonymized evaluation data.

nally for the name of the transducer and the connector, and they are completely anonymized.

5. Experiments and Results

We use CRFs (Lafferty et al., 2001) for training and testing with proposed data sets.[1] ± 2 word/POS window context and a bi-gram word/POS model are used as a feature set.[2] We use 0.1 for L2-regularization. We obtain 79.49% F_1-score on the new evaluation data set. Table 2 shows the detailed results for each error label. READ error labels are entirely based on lexical information and we correctly annotate over 95% of them because we have enough lexical information in training data. CONF error labels show only about 30% of precision because even though the expected modals as erroneous should have been detected exclusively in the main clauses, many of them were identified in the subordinated clauses where their use is allowed. AMBI and SING error labels for the combinator error pattern are usually required parsing results. While we used parsing results for building training data, we didn't consider these results for training the models. This is one of the main reasons that AMBI error labels have a relatively low recall. However, dependency information is difficult to be integrated in the sequence labeling model with dependency distance. Moreover, dependency parsing results are not often correct for conjunction marks such as *or* and *and*, which we use for the combinator error pattern.

6. Conclusion and Future Perspectives

We have presented linguistic models and a new anonymized evaluation data set for ReQA. A tool helping to improve the requirements authoring quality allows to reduce multiple proofreading steps which are time-consuming and costly but crucial in the whole life cycle of the Requirements Engineering. The accuracy of this kind of tool is obviously very important as technical authors (users) can reject to use them once they generate false positives of more than 20%. To reduce the rate of false positives, the model that we developed is based on the error patterns manually identified in the linguistic framework of the requirements grammar. In this paper, we can enrich the error patterns depending on the lexical information by adding more lexical items into our model. Additionally, the five constraints and the corresponding errors patterns do not cover all of the potential errors. It is necessary to revise and complete them in order to detect other error types: (1) detection of overspecified elements (design/solution parts – how the system realizes the required action – included in the requirements) for the Singularity, (2) detection of grammatical errors (e.g. ditransitive verbs missing one of arguments like *the system shall send the received configuration*) for the Completeness. There are also another types of constraints more ambitious such as the problem of consistency and of redundancy between requirements or sets of requirements. For those errors, we need to consider contextual information over a requirement sentence and to understand semantic meaning of the requirements and the relation between them. These are a great challenge for ReQA and we leave them as a future work of this paper.

	prec.(%)	recall(%)	F_1(%)	num.
AMBI	40.31	70.85	51.38	392
COMP	75.68	95.61	84.49	403
CONF	32.56	97.67	48.84	129
READ	91.57	98.80	95.05	629
SING	71.20	90.67	79.77	191
(all)	69.78	92.34	**79.49**	

Table 2: Experiment results: precision, recall, F_1-scores, and the number of entities.

[1] We use Wapiti (Lavergne et al., 2010) for training and evaluation, available at `https://wapiti.limsi.fr`.

[2] A trained model and training parameters are also available at `https://github.com/jungyeul/rqa`.

7. Bibliographical References

Alred, G. J., Brusaw, C. T., and Oliu, W. E. (2011). *Handbook of Technical Writing, Tenth Edition*. St. Martin's Press; 10 edition, New York.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Brants, T. (2000). TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, pages 224–231, Seattle, Washington, USA. Association for Computational Linguistics.

Dernoncourt, F., Lee, J. Y., Uzuner, O., and Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606.

Firesmith, D. (2003). Specifying Good Requirements. *Journal of Object Technology*, 2(4):77–87.

Fredj, F. B., Lammari, N., and Comyn-Wattiau, I. (2015). Abstracting Anonymization Techniques: A Prerequisite

for Selecting a Generalization Algorithm. *Procedia Computer Science*, 60:206–215.

Grady, J. O. (2013). *System Requirements Analysis*. Elsevier; 2 edition.

Kang, J. and Park, J. (2016). Generating a Linguistic Model for Requirement Quality Analysis. In *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Posters (PACLIC 30)*, pages 439–447, Seoul, Korea.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lavergne, T., Cappé, O., and Yvon, F. (2010). Practical Very Large Scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 504–513, Uppsala, Sweden. Association for Computational Linguistics.

Nivre, J., Hall, J., and Nilsson, J. (2006). MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*.

Weiss, E. H. (1990). *100 Writing Remedies: Practical Exercises for Technical Writing*. Greenwood.

Zhou, B., Pei, J., and Luk, W. (2008). A Brief Survey on Anonymization Techniques for Privacy Preserving Publishing of Social Network Data. *ACM SIGKDD Explorations Newsletter*, 10(2):12–22, 12.

A German Corpus for Fine-Grained Named Entity Recognition and Relation Extraction of Traffic and Industry Events

Martin Schiersch, Veselina Mironova, Maximilian Schmitt, Philippe Thomas, Aleksandra Gabryszak, Leonhard Hennig
DFKI GmbH
Berlin, Germany
{firstname.lastname}@dfki.de

Abstract

Monitoring mobility- and industry-relevant events is important in areas such as personal travel planning and supply chain management, but extracting events pertaining to specific companies, transit routes and locations from heterogeneous, high-volume text streams remains a significant challenge. This work describes a corpus of German-language documents which has been annotated with fine-grained geo-entities, such as streets, stops and routes, as well as standard named entity types. It has also been annotated with a set of 15 traffic- and industry-related n-ary relations and events, such as accidents, traffic jams, acquisitions, and strikes. The corpus consists of newswire texts, Twitter messages, and traffic reports from radio stations, police and railway companies. It allows for training and evaluating both named entity recognition algorithms that aim for fine-grained typing of geo-entities, as well as n-ary relation extraction systems.

Keywords: Named Entity Recognition, Relation Extraction

1. Introduction

Monitoring relevant news and events is of central importance in many economic and personal decision processes, such as supply chain management (Chae, 2015), market research (Mostafa, 2013), and personal travel planning (Schulz et al., 2013). Social media, news sites, and also more specialized information systems, such as online traffic and public transport information sources, provide valuable streams of text messages that can be used to improve decision making processes (Hennig et al., 2016). For example, a company's sourcing department may wish to monitor world-wide news for disruptive or risk-related events pertaining to their suppliers (e.g. natural disasters, strikes, liquidity risks), while a traveler wants to be informed about traffic events related to her itinerary (e.g. delays, cancellations). To fulfill such information needs, we need to extract events and relations from message streams that mention fine-grained entity types, such as companies, streets, or routes (Yaghoobzadeh and Schütze, 2017; Shimaoka et al., 2017). For example, from the sentence *"Berlin: Rail replacement service between Schichauweg and Priesterweg on route S2"*, we would like to extract a *Rail Replacement Service* event with the arguments *location="S2"* of type *location-route*, and *start-loc="Schichauweg"* respectively *end-loc="Priesterweg"* with types *location-stop*.

Detecting such relations in textual message streams raises a number of challenges. Social media streams, such as Twitter, are written in a very informal, not always grammatically well-formed style (Osborne et al., 2014), which cannot easily be processed with standard linguistic tools. News sites provide well-formed texts, but their content is very heterogeneous and often hard to separate from non-relevant web page elements. Domain-specific information sources, like traffic reports, on the other hand, are topic-focused, but employ a wide variety of formats, from telegraph style texts to table entries. In addition, existing corpora for German-language Named Entity Recognition (Tjong Kim Sang and De Meulder, 2003; Benikova et al., 2014) are mostly limited to standard entity types, and consist mainly of newswire and Wikipedia texts. These corpora also do not include annotations of events and relations. In this work, we present a large German-language corpus consisting of documents from three different genres, namely newswire texts, Twitter, and traffic reports from radio stations, police and railway companies (Section 2.). The documents have been annotated with fine-grained geo-entities, as well as standard entity types such as organizations and persons. In addition, the corpus has been annotated with a set of 15 mobility- and industry-related n-ary relation types (Section 3.). Many of these relation and event types, such as accidents, traffic jams, and strike events, are not available in standard knowledge bases and hence cannot be learned in a distantly supervised fashion. The final corpus consists of $2,598$ documents with $22,075$ entity and $1,507$ relation annotations (Section 4.). It allows for training and evaluating both named entity recognition algorithms that aim for fine-grained typing of geolocation entities, as well as for training of n-ary relation extraction systems.

2. Dataset Collection

To create the corpus, we collected a dataset of 3,789,803 tweets, 412,652 RSS feeds, and 860,307 news documents in the time period of Jan 1st, 2016 to March 31st, 2016. We aimed to collect only German-language texts by applying appropriate filter settings when crawling APIs, and by post-processing documents with langid.py (Lui and Baldwin, 2012). Figure 1 gives an example for each type of data source. All web documents, tweets, and RSS documents were transformed into a common Avro-encoded schema,[1] with fields for title, text, URI, and other attributes, as well

[1] avro.apache.org



(a) Example Twitter message (b) Excerpt of a RSS message (c) Excerpt of a news document

Figure 1: Examples for the three different data sources.

Entity / Concept	Description	Examples
Location (LOC)	General locations	Bayern, Zugspitze, Norden
Location-City (LOC-CIT)	Municipalities, e.g. cities, towns, villages	Berlin, Berlin-Buch, Hof
Location-Street (LOC-STR)	Named streets, highways, roads	Hauptstrasse, A1
Location-Route (LOC-ROU)	Named (public) transit routes	U1, ICE 557, Nürnberg – Hof
Location-Stop (LOC-STO)	Public transit stops, e.g. train stations, bus stops	S+U Pankow, Berlin-Buch
Organization (ORG)	General organizations	Greenpeace, Borussia Dortmund
Organization-Company (ORG-COM)	The subset of organizations that are businesses	Siemens AG, BMW
Person (PER)	Persons	Angela Merkel
OrgPosition (POS)	A person's position within an organization	CEO, Vizepräsident
Date (DAT)	Point in time, date	1. September 2017, gestern
Time (TIM)	Time of day	8:30, 5 Uhr früh
Duration (DUR)	Time periods	mehr als eine halbe Stunde
Distance (DIS)	Distances with unit	5 Kilometer
Number (NUM)	Other numeric entities, e.g. money, percentages	3%, 4 Millionen Euro
Disaster-Type (DIS-TYP)	Man-made and natural disaster types	Erdbeben, Überschwemmung
Trigger (TRI)	Trigger terms or phrases for events	Stau, Streik, Entlassungen

Table 1: Definition of entity types annotated in the corpus.

as fields for the annotations. From this dataset, we randomly sampled documents from each data source for annotation.

Twitter We use the Twitter Search API[2] to obtain a topically focused streaming sample of tweets. We define the search filter using a list of approximately 150 mobility- and industry-relevant channels and 300 search terms. Channels include e.g. airline companies, traffic information sources, and railway companies. Search terms comprise event-related keywords such as "traffic jam" or "roadworks", but also major highway names, railway route identifiers, and airport codes.

News We retrieve news pages and topically focused web sites using the uberMetrics Search API,[3] which provides an interface to more than 400 million web sources that are crawled on a regular basis. The API allows us to define complex boolean search queries to filter the set of web pages. We employ the same search terms as for Twitter, and limit the language to German. Boilerplate detection is used to remove extraneous contents from the HTML document (Kohlschütter et al., 2010). To speed up the annotation process, we limit each news document to the first 1000 characters, including the title, and discard the remainder of the text. Although this approach may result in the loss of some information, it is well known that in news writing, important information is presented first. The trimming may lead to incomplete final sentences, which annotators were advised to ignore.

RSS Feeds We implemented crawlers for a representative set of approximately 100 German-language RSS feeds that provide traffic and transportation information. Feed sources include federal and state police, radio stations, and air travel sources. The feeds were fetched at regular intervals during the 3-month period.

3. Annotation Guidelines

For the targeted applications of supply chain monitoring and personal travel planning, we are interested in the annotation of fine-grained geo-locations, such as street names and public transport stops, as well as relation (or event) mentions with typically multiple arguments. That is, we are not only interested to know that a given event type occurred, such as a traffic jam, but also, on which road, between which exits, and what the resulting time delay is for drivers. We hence aim to recognize and extract n-ary ACE/ERE-style relations (Doddington et al., 2004; Linguistic Data Consortium, 2015). The annotation guidelines and schema of the ACE Entities V6.6[4] and TimeML[5] served as a basis for annotating standard entity types, such as organizations, persons and dates. The main difference to ACE guidelines is the treatment of geo-political entities (GPE) – we chose to annotate them mainly as locations (LOC), and sometimes as organizations (ORG), in particular for cities, regions, or counties, as the relation types we are interested in typically refer to the location or organization aspects of a potential GPE entity.

[2] dev.twitter.com/rest/public/search

[3] doc.ubermetrics-technologies.com/api-reference/

[4] www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-entities-guidelines-v6.6.pdf

[5] www.timeml.org/publications/timeMLdocs/annguide_1.2.1.pdf

For the corpus annotation we use the markup tool Recon (Li et al., 2012), which allows annotating n-ary relations among text elements. Recon provides a graphical user interface that enables users to mark arbitrary text spans as entities, to connect entities to create relations, and to assign semantic roles to argument entities. Each document was annotated by two trained annotators. In cases of disagreement, a third annotator was consulted to reach a final decision. We measured inter-annotator agreement for entity and relation annotations. For entity annotations, we evaluated agreement at the entity level by comparing labels and offsets. A high inter-annotator agreement thus implies that annotators agreed both on the extent of entities and their type. For relation mentions, we measured role and relation type agreement at the level of relation arguments for each annotated relation mention. Similar to entity inter-annotator agreement, arguments were identified based on the underlying concepts/entities and their character offsets. A high inter-annotator agreement hence means that annotators agreed on entity, entity extent, role, and relation type labels. Table 2 lists the inter-annotator agreement values of our corpus. The pairwise kappa agreement is moderate at around 0.58 for entity annotations, which is somewhat lower than the 0.74 reported by Benikova et al. (2014). For relations, pairwise kappa agreement is 0.51.

Type	**Cohen's κ**	**Krippendorf's α**
Entities	0.58	0.57
Relations	0.51	0.45

Table 2: Inter-annotator agreement for entities and relations.

3.1. Entities

Table 1 lists the entity types we currently annotate, and provides a brief explanation of each type. In general, annotators were advised to choose the more specific entity type for a given entity mention (e.g. organization-company instead of organization), unless it was unclear from the context whether the entity mention referred to the specific or the more general type. This is for example the case for traffic jam and accident reports on main highways, where the exits often use the name of the closest city, e.g. *"accident on the A1 between [Bremen] and [Oldenburg]"*. Here, *"Bremen"* and *"Oldenburg"* are ambiguous between the types *location-city* and *location-street*.[6]

Organization-Company covers all commercial organizations, including media and entertainment businesses. It does not cover governmental or religious organizations, but may include sports teams. In general, though, we are interested in companies that provide services to other companies, e.g. in the form of products, parts, components, technologies, or non-physical services.

Location subtypes are of interest to pin-point exact locations for any event of interest, e.g. by a lookup in OpenStreetMap,[7] and to distinguish between location types where necessary. We do not tag locations if they are used as metonyms for organizations or GPEs, as in the case of capital cities denoting the government of a country.

In the case of traffic reports, we also consider directions, including cardinal points, as locations, for example:

(1) *Stau auf der B2 [stadteinwärts]*
(Traffic jam on the B2 [into town])

(2) *Auf der A1 Nähe Münster Stau in [beiden Richtungen]*
(On the A1, near Münster, traffic jam in [both directions])

For traffic-related locations, specifiers are included in the mention extent when required, e.g. *"[Kreuz München-Nord]"* (*"[Cross Munich-North]"*) as well as *"[Dreieck Havelland]"* (*"[Junction Havelland]"*), *"[Anschlussstelle Adlershof]"* (*"[Exit Adlershof]"*), *"[Abzweig nach Basel]"* (*"[Branch to Basel]"*), etc. Similarly, terms like *"Kreis"* (*"county"*) in *"[Kreis Tuttlingen]"* are included to distinguish the county from the city. However, terms like *"Ecke"* (*"corner"*) or *"Kreuzung"* (*"intersection"*) are not included in the extent of *Location-Street* entities, because they are not an integral part of the location's name.

City names that occur in transit routes are labeled as *Location-City* when they are used to indicate the direction of the route, and as *Location-Stop*s in every other case. In the case of highway exits, city names are labeled as *Location*s, since they actually denote the exit (and its geographic position), and not the city. For flight routes, we chose to label city names as *Location-City* unless the reference includes the specific airport used, e.g. *"Heathrow"* or *"MUC"*.

*Location-Route*s are either generic transit lines (e.g. *"S2"*), or a specific instance of this line (*"the next S2 which was supposed to arrive at 19:40"*). In general, they are referred to by letter-number combinations, but sometimes consist of concatenated stop or city names:

(3) *[Günzburg – Mindelheim]: Störung an einem Bahnübergang*
([Günzburg - Mindelheim]: Disruption at a crossing)

(4) *Ersatzverkehr auf der Linie [RE 3] [Stralsund/Schwedt (Oder) - Berlin – Elsterwerda]*
(Rail replacement service on the route [RE 3] [Stralsund/Schwedt (Oder) - Berlin - Elsterwerda])

Common nouns and noun phrases are annotated like proper names as entities of the corresponding type. Most often, they are used to denote a group of entities, e.g.:

(5) *Die [EC-Züge] zwischen Dresden Hbf und Praha hl.n. fallen aus*
(The [EC trains] between Dresden main station and Prague main station are cancelled)

(6) *Fraport übernimmt [14 griechische Flughäfen]*
(Fraport acquires [14 Greek airports])

Trigger concepts are a generic class of annotations that cover terms or phrases that indicate a specific event type, and that sometimes are required to create at least a binary relation mention within a sentence. For example, given the

[6] In the remainder of this document, '[' and ']' are used to denote the extent of an entity or relation mention

[7] openstreetmap.org

Relation / Event	Definition & Arguments
Accident	Collision of a vehicle with another vehicle, person, or obstruction ⊛location, ⊛trigger, delay, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Canceled Route	Cancellation of public transport routes ⊛location (LOC-ROU), ⊛trigger, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Canceled Stop	Cancellation of public transport stops ⊛location (LOC-STO), ⊛trigger, route, direction, start-date, end-date, cause (TRI)
Delay	Delay resulting from remaining traffic disturbances ⊛location, ⊛trigger, delay, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Disaster	Sudden catastrophe causing great damage to structures or loss of life ⊛type, ⊛location, date, victims (NUM), damage-costs (NUM), trigger
Obstruction	Temporary installation to control traffic ⊛location, ⊛trigger, delay, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Rail Replacement Service	Replacement of a passenger train by buses or other substitute public transport services ⊛location (LOC-ROU), ⊛trigger, delay, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Traffic Jam	Line of stationary or very slow-moving traffic ⊛location (LOC-STR), ⊛trigger, delay, jam-length, direction, start-loc, end-loc, start-date, end-date, cause (TRI)
Acquisition	Purchase of one company by another ⊛buyer, ⊛acquired, seller, date, price, trigger
Insolvency	Insolvency of a company ⊛company, ⊛trigger, date, location
Layoffs	Layoffs from companies, including number of people fired. ⊛company, ⊛trigger, date, location, num-laid-off
Merger	Merger of companies that is not a clear buy-up ⊛old (ORG-COM A), old (ORG-COM B), new (ORG-COM), date, trigger
Organization Leadership	Relationship between an organization and its leaders, board members, directors, etc. ⊛organization, ⊛person, position, from, to, trigger
SpinOff	Parent company "splits off" a section as a separate new company ⊛parent (ORG-COM), ⊛child (ORG-COM), location, trigger
Strike	Strike action affecting a company or organization ⊛company, ⊛trigger, date, location, num-striking, striker, union (ORG)

Table 3: Definition of the 15 target relations of the domains *Mobility* and *Industry*. ⊛ denotes the essential arguments of the relation that define the identity of a relation instance. Entity types are abbreviated or omitted in unambiguous cases.

message *"Stau auf der Warschauer Strasse"* (*"Traffic jam on Warschauer street"*), the location *"Warschauer Strasse"* alone is not sufficient to annotate a *Traffic Jam* event, which requires the additional annotation of the trigger *"Stau"* to distinguish it from other traffic-related event types. This reasoning applies for the relations *Insolvency*, *Layoffs* and *Strike* of the *Industry* domain, and for all relations of the *Mobility* domain except for the relation *Disaster*. However, the argument *type* of the relation *Disaster* can be filled only with concepts which can be considered as triggers for this event (earthquake, flood, nuclear accidents, etc.). The majority of mobility-related events we are interested in follow this pattern of *Location* + *Trigger* (or in the case of industry-related events, *Company* + *Trigger*) to distinguish between different event types that are expressed using the same syntactic patterns (see Section 3.2.).

Punctuation characters, such as "-", "/", "#" and "@" are not included in the mention extent unless they occurred inside a multi-token entity, e.g. *"#[Flughafen #Tempelhof]"*. Annotators were advised to make the mention extent as long as required to accurately denote a specific entity. The extent could include adjectives, numerals (*"more than"*, *"a few"*, *"several"*), or numbers, if these were used to denote a specific subset of a set-based named entity mention.

If an entity was referred to by two or more token sequences, e.g. *"Volkswagen (VW)"*, *"A1 Bremen - Hamburg"*, the annotators were advised to annotate two separate entities as in *"[Volkswagen] ([VW])"*.

As a rule, unless required for annotating a relation mention, nested entity mentions were not annotated, e.g. in *"PD Zwickau"* (*"police department Zwickau"*), *"[PD Zwickau]"* was labeled as an *Organization*, but the nested *"Zwickau"* was not labeled as a *Location-City*.

3.2. Relations and Events

We annotated two different sets of relations and events in the corpus, based on the requirements of the project this corpus was developed in. The first group of relations are mobility-related, and include for example *Traffic Jam*s, *Accident*s and *Disaster*s. The second group of relations concerns companies, and includes e.g. *Acquisition*, *Strike* and *Insolvency* events. Table 3 lists all relation types, together with their definitions and arguments. All relations have a set of required (typically two) and a set of optional arguments. For example, the relation *Acquisition* has required arguments *buyer* and *acquired*, and optional arguments *date*, *price*, and *seller*. The following examples illustrate our n-ary relation annotations:

Relation examples of the *Mobility* domain

(7) Accident*: [A8]$_{loc}$ Augsburg Richtung [München]$_{dir}$ - Schwerer [Unfall]$_{tri}$ - kurz vor [Ausfahrt Dasing]$_{sta}$.*
(A8 from Augsburg to Munich - a severe accident - just before the exit Dasing)

(8) Canceled Route*: Wegen des Warnstreiks hat die Lufthansa [mehrere Flüge]$_{loc}$ in [Hamburg]$_{sta}$, [Hannover]$_{sta}$ und weiteren Flughäfen [gestrichen]$_{tri}$.*
(Because of the warning strike, Lufthansa has canceled several flights in Hamburg, Hanover and other airports)

(9) Canceled Stop*: Rinjani macht Ärger: [Flughafen auf Bali]$_{loc}$ wegen [Vulkanausbruch]$_{cau}$ [gesperrt]$_{tri}$.*
(Rinjani causes trouble: Bali airport is closed due to volcanic eruption.)

(10) Delay*: [S-Bahn-Verkehr Stuttgart]$_{loc}$: [Notarzteinsatz]$_{cau}$ in [Feuerbach]$_{sta}$ sorgt für [Verspätungen]$_{tri}$*
(S-Bahn traffic Stuttgart: Emergency medical service in Feuerbach causes delays)

(11) Disaster*: [Mehrere Tote]$_{vic}$ bei erneutem [Erdbeben]$_{typ}$ in [Japan]$_{loc}$*
(Several dead in another earthquake in Japan)

(12) Obstruction*: Wegen [Notarzteinsatzes]$_{cau}$ ist derzeit die [Strecke]$_{loc}$ zwischen [Gerlenhofen]$_{sta}$ und [Senden]$_{end}$ [gesperrt]$_{tri}$.*
(Due to an emergency medical service, the route between Gerlenhofen and Senden is currently closed)

(13) Rail Replacement Service: *[RB59]$_{loc}$: Vom [11.6.]$_{sdat}$ - [3.7.]$_{edat}$ [Schienenersatzverkehr]$_{tri}$ zwischen [Soest]$_{sta}$ und [Holzwickede]$_{end}$ im Spätverkehr.*
(RB59: Rail replacement service from 6/11 until 7/3 between Soest and Holzwickede during evening hours.)

(14) Traffic Jam*: [A40]$_{loc}$ Duisburg Richtung [Venlo]$_{dir}$ zwischen [Neukirchen- Vluyn]$_{sta}$ und [Kempen]$_{end}$ [10 km]$_{len}$ [Stau]$_{tri}$*
(A40 Duisburg - Dortmund between Neukirchen-Vluyn and Kempen 10 km traffic jam)

Relation examples of the *Industry* domain

(15) Acquisition*: Wirecard AG und ihre Tochtergesellschaft [Wirecard Acquiring & Issuing]$_{buy}$ haben den Zahlungsdienstleister [Moip Pagamentos]$_{acq}$ [übernommen]$_{tri}$.*
(Wirecard AG and its subsidiary Wirecard Acquiring & Issuing have acquired the payment service provider Moip Pagamentos.)

(16) Insolvency*: [Imtech]$_{com}$ [Insolvenz]$_{tri}$ gefährdet BER-Eröffnung*
(Imtech insolvency endangers BER opening)

(17) Layoffs*: [Entlassungen]$_{tri}$ bei [Credit Agricole Indosuez]$_{com}$ in [Genf]$_{loca}$*
(Layoffs at Credit Agricole Indosuez in Genf)

(18) Merger*: Der Panzerhersteller [Krauss-Maffei Wegmann]$_{old}$ besiegelt den [Zusammenschluss]$_{tri}$ mit dem französischen Rüstungskonzern [Nexter]$_{old}$.*
(Tank manufacturer Krauss-Maffei Wegmann seals merger with French arms company Nexter.)

(19) Organization Leadership*: [Bernd Hansen]$_{per}$, CEO$_{pos}$ [Hansen Gruppe]$_{com}$*
(Bernd Hansen, CEO Hansen Group)

(20) SpinOff*: [Kölnische Unfall-Versicherungs-Aktiengesellschaft zu #Köln a.Rhein]$_{chi}$, gegr. [1919]$_{dat}$ als [Ableger]$_{tri}$ der [Colonia]$_{par}$*
(Kölnische Unfall-Versicherungs-Aktiengesellschaft zu #Köln a.Rhein, est. in 1919 as a spin-off of the Colonia)

(21) Strike*: Am [Freitag]$_{dat}$ haben die [Amazon]$_{com}$-Mitarbeiter im [Leipziger]$_{loc}$ Versandzentrum des Unternehmens erneut [gestreikt]$_{tri}$.*
(On Friday, Amazon employees in the company's shipping center in Leipzig once more went on strike.)

Some of the relations are semantically related to each other and can occur together even in very short texts such as tweets or RSS feeds. For example, the relation *Traffic Jam* often correlates with *Accident* and *Obstruction* relation mentions. This also applies to the relation *Obstruction* and the event *Disaster*, and *Delay* relations and the events *Canceled Route*, *Canceled Stop* and *Rail Replacement Service*. In the *Industry* domain, we observe that reports of corporate events often include information about leaders of an organization, i.e. a *Organization Leadership* relation is mentioned together with another relation.

The annotators annotated only explicitly expressed relation mentions where all arguments – required and optional – occurred within a single sentence. In cases of multiple occurrences of an argument, they chose the arguments occurring within the shortest overall text span. Future or planned relations, such as potential acquisitions or announced strikes, were also marked up, and labeled with an additional attribute to indicate this status. Negated relation mentions (e.g. a canceled acquisition), or events marking the end of a relation (e.g. *"Traffic jam has dissolved"*) were not annotated.[8] The following examples illustrate the three types of relation mentions:

(22) Factual*: Der Panzerhersteller Krauss-Maffei Wegmann besiegelt den Zusammenschluss mit dem französischen Rüstungskonzern Nexter.*
(Tank manufacturer Krauss-Maffei Wegman seals merger with French armaments group Nexter.)

(23) Potential*: Größer als BASF: US-amerikanische Chemie-Unternehmen DuPont und Dow Chemical planen Mega-Fusion.*
(Larger than BASF: US-American chemical companies DuPont and Dow Chemical are planning mega-merger)

[8] However, the files containing such mentions were marked by renaming them. Fully annotating the negated relation mentions and including them in the corpus remains future work.

	News	Twitter	RSS	Total
Documents	835	1,138	625	2,598
Sentences	5,951	1,842	1,031	8,824
Sentences (avg.)	7.13	1.62	1.65	3.40
Words	113,089	19,558	19,595	152,242
Words (avg.)	135.44	17.19	31.35	58.60

Table 4: Corpus Statistics

	News	Twitter	RSS	Total
Entities	13,500	3,478	5,097	22,075
Entities (avg.)	16.17	3.06	8.16	8.50
Relations	597	454	456	1,507
Relations (avg.)	0.71	0.40	0.73	0.58

Table 5: Annotation Statistics

(24) Negation*: BHF-Bank: Fosun beteuert, keine Fusion mit Hauck & Aufhäuser anzustreben*
(BHF-Bank: Fosun re-affirms not seeking a merger with Hauck & Aufhäuser)

4. Corpus Statistics

This section summarizes the key characteristics of the final corpus. It contains a total of $2,598$ documents with more than $150,000$ words. Table 4 shows a brief summary of all document types, while Table 5 summarizes the annotation statistics per document type. In total, the annotators labeled $22,075$ entities and $1,507$ relation occurrences. Due to their greater length, news documents contain the largest number of entity mentions, significantly more than the other two document types. Twitter documents on average contain fewer relation mentions than RSS and news documents. The overall fraction of documents containing at least a single relation mention is 58%, a rather high figure that can be attributed to the focused retrieval process which was used to create the initial dataset.

Figure 2 shows the distribution of annotated entities and relations across document types. Companies, general locations, and cities are the most frequent entity types in our dataset. Public transit stops, streets, and routes are less frequently mentioned, and occur predominantly in tweets and RSS traffic reports. This is an expected distribution, since major news outlets generally do not report on day-to-day, local traffic events. With regards to relation types, traffic events like *Traffic Jam*s and *Obstructions* occur very frequently. Other event types occur with lower frequency in our annotated data, in particular, the annotators identified only very few instances of *Canceled Stop* and *SpinOff* events.

4.1. Baseline NER and RE experiments

We conducted a series of experiments to report initial performance figures on the presented corpus for the tasks of named entity recognition and relation extraction. We use the Stanford CoreNLP tools (Manning et al., 2014) for training a NER classifier, and a dependency pattern based model for relation extraction. The relation extraction algorithm, DARE, learns minimal dependency subgraphs that connect all relation arguments, and is described in (Xu et al., 2007; Krause et al., 2012). We did not perform any filtering of the extracted dependency patterns, i.e. we include all learned patterns, even ambiguous or low-frequency ones, in the model.[9]

We randomly split the full dataset into 50% training and 50% test data. The NER model was trained using the standard feature configuration employed by Stanford CoreNLP for NER.[10] The RE models were trained with and without using gold standard NE annotations.

For NER, we evaluated the model's performance both at the token level and at the concept level. The results are shown in Table 6. We see that the overall token-level F1 score is close to 0.85, a respectable figure given the confusability of location subtypes such as streets, stops, and routes in our dataset. For *organization* and *organization-company* entities, the average token-level F1 score is lower at approximately 0.78, but for *location* and its subtypes, it lies between $0.85 - 0.92$ (not shown).

Evaluation Type	Precision	Recall	F1
CRF (token)	0.8966	0.8024	0.8469
CRF (concept)	0.7984	0.6797	0.7343

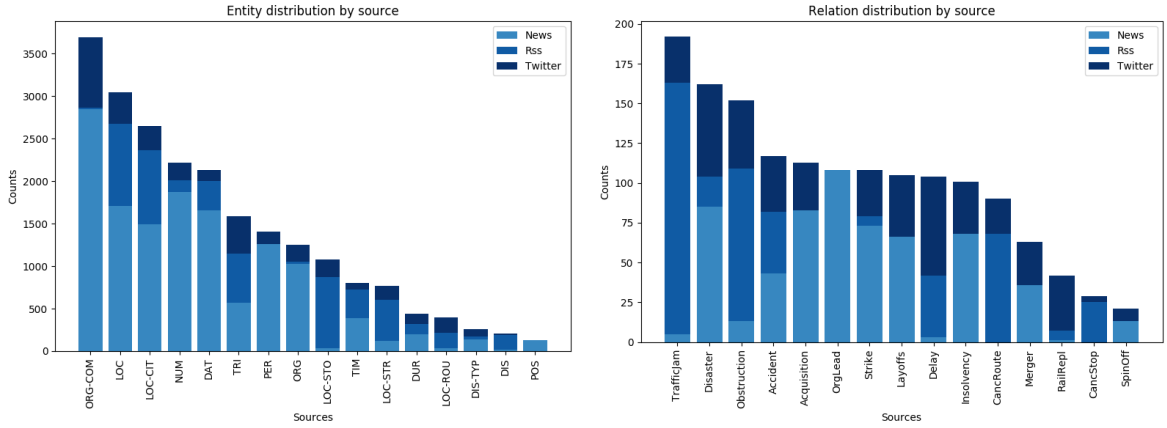
Table 6: Performance of a standard CRF-based NER classifier on the presented dataset.

For relation extraction, the models were evaluated at the mention level, by comparing predicted relation mentions with gold relation mentions. Since our dataset contains n-ary relations with optional and required arguments, we chose a soft matching strategy that counts a predicted relation mention as correct if all predicted arguments also occur in the corresponding gold relation mention, and if all required arguments have been correctly predicted, based on their role, underlying entity, and character offsets / extent. Optional arguments from the gold relation mention that are not contained in the predicted relation mention do not count as errors. In other words, we count a predicted relation mention as correct if it contains all required arguments and is subsumed by or equal to the gold relation mention.

Table 7 shows the results of two RE evaluation runs, once with gold-standard NE annotations, and once without any gold annotations. As can be expected, the performance of the RE models using gold-standard NE annotations is significantly higher than that of the models using the trained NER classifier. The dependency-based DARE model achieves an F1 score of 0.28 using gold-standard NEs, and is biased toward high-precision patterns, at the expense of recall.

[9] Obviously, properly filtering patterns may significantly improve performance, but state-of-the-art RE performance is not the goal of this study.

[10] See `nlp.stanford.edu/software/crf-faq.html`



(a) Distribution of annotated entities

(b) Distribution of annotated relations

Figure 2: Entity and relation type distribution across document source types

Model	Precision	Recall	F1
DARE (CRF NE)	0.4670	0.1308	0.2043
DARE (Gold NE)	0.5274	0.1923	0.2818

Table 7: Performance of a dependency pattern based RE model on the presented dataset.

5. Related Work

There are very few available NER and RE datasets for German. Most noteworthy are the NER dataset presented by Benikova et al. (2014), and the CoNLL-2003 dataset (Tjong Kim Sang and De Meulder, 2003). Both datasets contain annotations only for the three standard entity types, PER, ORG and LOC, as well as MISC/OTHER. The dataset by Benikova et al. includes nested annotations, whereas in our corpus, nested annotations are only annotated if they are required for a relation mention. Both datasets use news articles as their data source, with Benikova et al.'s dataset including Wikipedia texts. In contrast, our dataset also contains Twitter texts, as well as telegraphese-style reports from official traffic channels, which allows for text genre-specific evaluation of NER approaches.

The ACE datasets (Doddington et al., 2004; Linguistic Data Consortium, 2015) are similar to the dataset presented in this paper in that they include both NE and RE annotations. The various ACE datasets developed over the years consider a wide range of entity types, such as PER, ORG, LOC, GPE and FAC. Similarly, a range of different relation types are annotated in these datasets, including geographical, social and business relationships. However, all relations definitions are limited to binary relations, whereas our corpus contains n-ary relation mentions. None of the ACE datasets cover German-language documents.

Other well-known English relation extraction datasets include the corpora prepared for the TAC-KBP challenges (Ji et al., 2011; Surdeanu, 2013), the SemEval-2010 Task 8 dataset (Hendrickx et al., 2010), and the TACRED dataset by (Zhang et al., 2017).

6. Conclusion

We presented a corpus of German Twitter, news and traffic report texts that has been annotated with fine-grained geo-entities as well as a set of mobility- and industry-related events. Many of the event types annotated in the corpus are not available in standard knowledge bases, such as accidents, traffic jams, and strike events. We make the corpus and the guidelines available to the community at `https://dfki-lt-re-group.bitbucket.io/smartdata-corpus`. The dataset is distributed in an AVRO-based compact binary format, along with the corresponding schema and reader tools.

Acknowledgments

This research was partially supported by the German Federal Ministry of Economics and Energy (BMWi) through the projects SD4M (01MD15007B) and SDW (01MD15010A), by the German Federal Ministry of Transport and Digital Infrastructure through the project DAYSTREAM (19F2031A), and by the German Federal Ministry of Education and Research (BMBF) through the project BBDC (01IS14013E).

7. Bibliographical References

Benikova, D., Biemann, C., and Reznicek, M. (2014). NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2524–2531, Reykjavik, Iceland, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1251.

Chae, B. K. (2015). Insights from hashtag #supplychain and Twitter Analytics: Considering Twitter and Twitter data for supply chain practice and research. *International Journal of Production Economics*, 165(Supplement C):247 – 259.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. (2004). The Automatic

Content Extraction (ACE) Program - Tasks, Data, and Evaluation. In *Proc. of LREC*.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., Pennacchiotti, M., Romano, L., and Szpakowicz, S. (2010). SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July. Association for Computational Linguistics.

Hennig, L., Thomas, P., Ai, R., Kirschnick, J., Wang, H., Pannier, J., Zimmermann, N., Schmeier, S., Xu, F., Ostwald, J., and Uszkoreit, H. (2016). Real-time discovery and geospatial visualization of mobility and industry events from large-scale, heterogeneous data streams. In *Proceedings of ACL-2016 System Demonstrations*, pages 37–42, Berlin, Germany, August. Association for Computational Linguistics.

Ji, H., Grishman, R., and Dang, H. T. (2011). Overview of the TAC 2011 Knowledge Base Population Track. In *Proc. of the 4th Text Analysis Conference*.

Kohlschütter, C., Fankhauser, P., Nejdl, W., Kohlschütter, C., Fankhauser, P., and Nejdl, W. (2010). Boilerplate Detection Using Shallow Text Features. In *Proc. of WSDM*, pages 441–450.

Krause, S., Li, H., Uszkoreit, H., and Xu, F. (2012). Large-Scale Learning of Relation-Extraction Rules with Distant Supervision from the Web. In *Proc. of ISWC*, pages 263–278.

Li, H., Cheng, X., Adson, K., Kirshboim, T., and Xu, F. (2012). Annotating opinions in german political news. In *8th ELRA Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA), 5.

Linguistic Data Consortium. (2015). Rich ERE annotation guidelines overview. `http://cairo.lti.cs.cmu.edu/kbp/2015/event/summary_rich_ere_v4.1.pdf`.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proc. of ACL: System Demonstrations*, pages 25–30.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June. Association for Computational Linguistics.

Mostafa, M. M. (2013). More than words: Social networks' text mining for consumer brand sentiments. *Expert Systems with Applications*, 40(10):4241 – 4251.

Osborne, M., Moran, S., McCreadie, R., Von Lunen, A., Sykora, M., Cano, E., Ireson, N., Macdonald, C., Ounis, I., He, Y., Jackson, T., Ciravegna, F., and O'Brien, A. (2014). Real-time detection, tracking, and monitoring of automatically discovered events in social media. In *Proc. of ACL: System Demonstrations*, pages 37–42.

Schulz, A., Ristoski, P., and Paulheim, H. (2013). I see a car crash: Real-time detection of small scale incidents in microblogs. In *Extended Semantic Web Conference*, pages 22–33. Springer, Berlin, Heidelberg.

Shimaoka, S., Stenetorp, P., Inui, K., and Riedel, S. (2017). Neural Architectures for Fine-grained Entity Type Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1271–1280, Valencia, Spain, April. Association for Computational Linguistics.

Surdeanu, M. (2013). Overview of the TAC 2013 Knowledge Base Population Evaluation: English Slot Filling and Temporal Slot Filling. In *Proceedings of the Text Analysis Conference*.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans et al., editors, *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Xu, F., Uszkoreit, H., and Li, H. (2007). A Seed-driven Bottom-up Machine Learning Framework for Extracting Relations of Various Complexity. In *Proc. of ACL*, pages 584–591.

Yaghoobzadeh, Y. and Schütze, H. (2017). Multi-level Representations for Fine-Grained Typing of Knowledge Base Entities. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 578–589, Valencia, Spain, April. Association for Computational Linguistics.

Zhang, Y., Zhong, V., Chen, D., Angeli, G., and Manning, C. D. (2017). Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark, September. Association for Computational Linguistics.

A Corpus Study and Annotation Schema for Named Entity Recognition and Relation Extraction of Business Products

Saskia Schön, Veselina Mironova, Aleksandra Gabryszak, Leonhard Hennig
DFKI GmbH
Berlin, Germany
{firstname.lastname}@dfki.de

Abstract

Recognizing non-standard entity types and relations, such as B2B products, product classes and their producers, in news and forum texts is important in application areas such as supply chain monitoring and market research. However, there is a decided lack of annotated corpora and annotation guidelines in this domain. In this work, we present a corpus study, an annotation schema and associated guidelines, for the annotation of product entity and company-product relation mentions. We find that although product mentions are often realized as noun phrases, defining their exact extent is difficult due to high boundary ambiguity and the broad syntactic and semantic variety of their surface realizations. We also describe our ongoing annotation effort, and present a preliminary corpus of English web and social media documents annotated according to the proposed guidelines.

Keywords: Named Entity Recognition, Relation Extraction, Information Extraction

1. Introduction

Recognizing non-standard entity and relation types is an important task in many real-world information extraction applications like relation extraction, knowledge base construction and question answering. In areas such as market research and supply chain management, many companies would benefit from systems that automatically and continuously acquire up-to-date information about producers, vendors and other suppliers of specific parts, products, new technologies and components. Similarly, the construction of knowledge graphs that store supplier and vendor relationships would clearly benefit from information extraction approaches by reducing the manual effort required to create and maintain such databases. For example, in both scenarios it would be useful to extract information about e.g. a *CompanyProvidesProduct* relation from a news text like *"Sensata Technologies' products include speed sensors, motor protectors, and magnetic-hydraulic circuit breakers"*, where the *product* argument refers to a non-consumer product or product class entity such as *"speed sensors"* or *"magnetic-hydraulic circuit breakers"*.

However, when it comes to such specific domains, developing named entity recognition algorithms is severely hampered by the lack of publicly available training data and the difficulty of accessing existing dictionary-type resources, such as product catalogs. Many available named entity recognition corpora consist of general news articles (Tjong Kim Sang and De Meulder, 2003; Doddington et al., 2004; Weischedel et al., 2013), while information about B2B products is typically available on non-journalistic, specialized web portals and forums. Product mentions, as in the example above, are often general noun phrases, instead of proper names, which increases the difficulty of detecting them using gazetteer-based approaches. In addition, relational information about companies and their products is very limited in freely available knowledge bases (KB), such as Freebase (Bollacker et al., 2008), Wikidata (Vrandečić and Krötzsch, 2014), or DBpedia (Auer et al., 2007), since these KBs are in large parts based on Wikipedia, which aims to exclude commercial, non-encyclopedic information. For example, DBpedia contains only approximately $60,000$ triples for the *CompanyProvidesProduct* relation.

To address these problems, and to gain a better understanding of product mentions and their linguistic properties, in this study we first collect a large number of noisy product mentions. This is achieved with a bootstrapping approach that uses a set of manually defined lexical patterns for the relation *CompanyProvidesProduct* (Section 3.). We analyze the resulting set of mentions, and find that they often include extraneous linguistic material that should not be considered a part of the product extent, such as prepositional phrases and appositions. Consequently, we develop an annotation schema for *product* mentions and the *CompanyProvidesProduct* relation, in order to guide the manual annotation of texts (Section 4.). We are currently building a corpus of English web and social media documents with annotations for *product* entity and *CompanyProvidesProduct* relation mentions based on these annotation guidelines. We will make a first version of the corpus available to the community (Section 6.). The overall goal of our work is to make extraction of non-standard, B2B products and relations from unstructured text easier and more reliable.

2. Related Work

Most research in Named Entity Recognition (NER) has focused on common entity types, such as persons, organizations, and locations (Tjong Kim Sang and De Meulder, 2003; Finkel et al., 2005; Derczynski et al., 2016), and numeric types like date and time expressions (Strötgen and Gertz, 2013). Only a few corpora cover other entity types, such as geopolitical entities and facilities (Doddington et al., 2004; Weischedel et al., 2013). Corpora that include product annotations are rare: the BBN corpus covers (consumer) products mentioned in Wall Street Journal news articles (Weischedel and Brunstein, 2005). Liu et al. (2011) describe a corpus of tweets that has been annotated with products, but the dataset is not publicly avail-

able. Recent research in fine-grained NER has produced distantly (Ling and Weld, 2012; Weischedel et al., 2013) or weakly supervised datasets (Ni et al., 2017) using Freebase and Wikipedia, which therefore inherit the coverage and specificity limitations of these resources. The datasets include products and their subtypes, but the entities are generally consumer products, such as cars, mobile phones, and games. In the case of Ni et al. (2017), the dataset is not publicly available.

For relation extraction, there exist only very few datasets that have been manually annotated with company-product-related information. FrameNet (Baker et al., 1998) contains example sentences marked up with frames that provide information that is similar to the *CompanyProvidesProduct* relation, such as the *Business*, *Commerce_sell* and *Manufacturing* frames. The SemEval 2010 Task 8 contains 968 sentences annotated with pairs of nominals for mentions of the *Product-Producer* relation (Hendrickx et al., 2010). While the relation's name suggests similarity to our dataset, the scope in the SemEval dataset is much broader and includes any kind of production, e.g. blisters caused by a herpes virus, children "produced" by their parents, or questions asked by journalists.

The ACE guidelines for English relations (Linguistic Data Consortium, 2005) describe the relation type *Agent-Artifact*, which "applies when an agent owns an artifact, has possession of an artifact, uses an artifact, or caused an artifact to come into being." However, the non-organization argument of the corresponding relation definition only allows *facility*-type entities, and does not mention products.

3. Bootstrapping Product Annotation

This section presents the data sources used in this work, and the pattern-based bootstrapping approach used for pre-annotating products.

3.1. Source Datasets

We collected a large dataset of web pages from business news portals, company home pages, and special interest forums, as well as posts from social media channels such as Twitter and Facebook. Web pages and forums were crawled based on an iteratively refined set of source URLs and keywords, such as company name lists. Similar keyword sets were also used for retrieving public Twitter and Facebook posts using the respective APIs of these services. The dataset was crawled over a period of 1.5 years, between July 2016 and December 2017, and consists of approximately 5.8 million documents. We focused on B2B information related sites, but did not explicitly exclude documents reporting on consumer products. Figure 1 shows an example document from the corpus.

As can be expected, documents exhibit a large degree of linguistic variance, ranging from journalistic writing to colloquially formulated tweets. In addition, HTML-to-text conversion and boilerplate removal are far from perfect, resulting in extraneous and not-well-formed content. We noticed that B2B news sites often appended "canned" company summaries at the end of news articles, which typically contain a lot of useful information about the company's products. We included an example of this feature as the second paragraph of Figure 1.

Sensata Technologies Introduces Its Smallest Micro-fused Strain Gage Technology For Next Generation Brake Systems Especially Designed For Hybrid And Electric Vehicles

PR Newswire | Monday, 01 February 2016 03:37 (EST) ATTLEBORO, Mass., Feb. 1, 2016

/PRNewswire/ -- Sensata Technologies, Inc. (NYSE: ST), a leading manufacturer of sensing, electrical protection, control and power management solutions, has developed a line of smaller, lighter Micro-fused Strain Gage (MSG) pressure sensors for use in next-generation brake systems for hybrid, electric, and conventional vehicles. The eXtra-small Form Factor (XFF) sensor is available for design-in beginning January 2016. Sensata's automotive MSG pressure technology will now be offered at less than 5 grams, with a body diameter less than 7.8mm, and a height less than 30mm, including its revolutionary spring contact system. This provides system manufacturers with a new degree of design flexibility and including industry leading performance …

About Sensata Technologies: Sensata Technologies is one of the world's leading suppliers of sensing, electrical protection, control and power management solutions with operations and business centers in 16 countries. Sensata's products improve safety, efficiency and comfort for millions of people every day in automotive, appliance, aircraft, industrial, military, heavy vehicle, heating, air-conditioning and ventilation, data, telecommunications, recreational vehicle and marine applications. For more information please visit Sensata's website at www.sensata.com.

Figure 1: Example document relating B2B product and supplier information.

3.2. Pattern-based Product Pre-Annotation

To gain a better understanding of the linguistic properties of the different types of product mentions in our dataset, we included an automatic pre-annotation step in our corpus analysis. Pre-annotation can help to decrease the duration of manual annotation and generally ease the annotation process (Kwon et al., 2014). We first developed a set of lexical patterns for the relation *CompanyProvidesProduct*. We chose this approach since we are also interested in identifying instances of this relation, and since it significantly increases the precision of product mention identification (at the cost of recall).

The *CompanyProvidesProduct* relation maps a company (organization) to products created, manufactured, provided, distributed or vendored by this company (Section 5.). Table 1 lists some example patterns. Many of these patterns can be varied by exchanging the verb or verb nominalization used, e.g. *produce, create, develop, make, manufacture, offer*. In total, we defined 13 base patterns, yielding a total of 173 surface patterns. We used a chunk parser to label potential product mentions. It matches noun phrases, optionally preceded or followed by adjectives or cardinal numbers, e.g. *"high-resolution waveform analysis"*, *"High-Frequency 600mA DC-DC Buck Converter"*, and *"1500 ECL-PTU-208"*. We included the *VBG* tag as in rare cases, gerund verb forms may be part of the product mention, e.g. *"communicating sensors"*.

For pre-annotation, we randomly selected a set of $1,200$ documents from the source dataset. Each document's text was tokenized and part-of-speech tagged. Organizations were detected using Stanford NER (Manning et al., 2014).

Pattern	Example
ORG *'s* PRO:{<VBG|NN.*|JJ|CD>*<NN.*>+<NN.*|JJ|CD>*}	BMW's [1-Series Convertible] is a stylish convertible.
PRO *by* ORG	[Intuition Executive] by Honeywell collects and analyzes large amounts of data.
ORG [*to produce*|*to manufacture*|*to develop*|...] PRO	Sensata Technologies develops [sensors] and [controls].
ORG [*to be*] [*producer of*|*maker of*|...] PRO	Amazon is a vendor of [books] and [technology products].
ORG [*to be*] [*a*|*the*|*an*] PRO [*producer*|*provider*|*supplier*|...]	Apple and Samsung are [smartphone] providers.

Table 1: Example bootstrap patterns for the relation *CompanyProvidesProduct* used for pre-annotating product mentions. Company arguments are underlined, product arguments enclosed in brackets. For brevity, the chunking pattern applied to extract potential product mentions is only shown in the first row of the table.

We then applied the patterns, which resulted in a total of $1,308$ potential product mention matches.
While many of these matches did at least cover a product mention, many results were unsatisfactory because they included excess lexical material, as shown in the following examples (square brackets denote the correct mention extent):

(1) a. **highly accurate [3D magnetic sensor]*
b. **advanced [magnetic-hydraulic circuit breakers]*
c. **Rambus' [R+] industry-standard [interface solutions]*

We also observed that even when syntactical extraction worked correctly in a way that only phrases containing a product mention were retrieved, there were differences in semantic quality that should be taken into account. An adjective in the same position, for example, can either be of no value to the categorial specification of the product, in which case it can be neglected, or it can be a crucial part of the category description:

(2) a. *advanced [sensors]*
b. *[magnetic sensors]*

The extent to which these issues occurred seems to be closely linked with the specific product domain. These observations show that due to high boundary ambiguity as well as a broad syntactic and semantic variety of the surface variants of product mentions, it is necessary to define in advance which elements should be considered part of the extent of a product mention, and which should be excluded.

4. Annotation Guidelines

To formalize the annotation of *product* mentions and *CompanyProvidesProduct* relation mentions, we developed a set of annotations guidelines. For entity annotation, we base our guidelines on existing work, such as the ACE annotation guidelines (Doddington et al., 2004) for labeling *organizations* and *companies*. We try to follow similar guidelines for the annotation of products, but transform and expand these as detailed below.
Since many phrases in a typical document can be viewed as products or product classes (e.g. *"mobile services"*, *"hotel chains"*, *"personal devices"*), annotators limited their effort by adopting the following overall strategy for labeling a document: First, they annotated all name mentions of *organizations* and *products*, as well as any coreferential nominal and pronominal mentions of these. Coreference information was added as an extra relation type *Identity*. A single *Identity* relation was created for each coreference chain, with a *source* argument for the most precise name mention of an entity in a document, and *target* arguments for all other mentions of this entity. In a second step, annotators searched for occurrences of *CompanyProvidesProduct* relation mentions, and labeled (pro-) nominal product or product class references if they served as the argument of the identified relation mention. For example, annotators would label *"sensors"* and *"controls"* as *product*s in the sentence *"Sensata Technologies develops sensors and controls."* during this second step.[1] In a last step, coreferential mentions of these additional *product* mentions were also labeled in the remainder of the document, even if they did not occur as an argument of a relation mention. The reasoning here is that an NER algorithm should encounter consistent labels for the same token sequence, e.g. if *"smartphones"* is labeled as a *product* once, it should be labeled as a *product* everywhere in the document. All other product or product class references, i.e. those that were not part of a *CompanyProvidesProduct* relation mention, were not annotated as *product* entity mentions. We chose this strategy to limit the annotation effort for the initial corpus. In addition, disagreement by the annotators was very high when annotating all noun phrases that could potentially be viewed as products.

4.1. Products and Product Mentions

We define as a product any commercially available good, be it a finished product, a pre-product, or a part or component of a larger product. While the focus of this work is on non-consumer products, this definition also includes consumer products. A product does not have to be a tangible object, but can be a service or virtual object. Although they are semantically closely related, product-like entity mentions that refer to an industry sector or branch of business are not treated as products. The industry term can, however, be part of the product mention. Categories such as brand names and trademarks also often appear as part of the product mention.
For the reliable extraction of a product mention its maximum extent must be pre-defined. This means that one has to identify both the elements included in the extent and those outside of the extent that often appear alongside the

[1] Only if these nouns were not labeled as coreferential to some product name mention in the first step, of course.

Category	Example	Description
company name	[Dunlop] Sport M3 winters	This is the name of the company that provides the product when it is mentioned as part of the product name. It is usually found in the beginning of the product mention. It tends to be a proper noun that does not always follow orthographic rules and sometimes appears as an abbreviation (e.g. "Tumblr", "BMW"). The complexity of a company name tends to be reduced to the main word(s) when it is part of the product mention, i.e., "Toyota" instead of "Toyota Motor Corporation".
brand name	Apple [iPhone] 6S	The brand name is the name under which a certain product is marketed. Usually this is a proper name, however it is not always capitalized or otherwise orthographically correct. Like all categories that can be realized as a proper name, a brand name can include any word class, special characters and punctuation (e.g. "FILL OR BUST!").
series	VW Golf [VII]	This is the part of the product mention that denotes the series, generation, edition or model range. It is often realized as a number, sometimes as a name. Sometimes it includes the word "series" or "generation" or an equivalent abbreviation. Whether or not this can be part of a product mention is dependent on the domain.
model	BMW [i8]	This part of the product mention denotes the specific model of a product in a product series. It often consists of letters and numbers or a combination of the two.
trademark symbol	McRib[®]	The trademark symbol usually appears right after the brand name.
type	Nike Air Max 2016 [running [shoes]]	The type of the product is the broader category or subcategory a product falls into. The type is usually a common noun and can often be found at the end of the product mention. The category can include an attribute that serves as a specifier to the noun and it is the category in which nonspecific terms such as "product" or "solutions" can be included, serving as the head of the product mention, but only if specified further (e.g. "cosmetic product").
feature	[2006] Ford Mustang [GT] Convertible [2-Door] Samsung Galaxy S7 [32 GB] [black]	Most other relevant aspects of products we will categorize as a feature. Since this is the vaguest category and the one that is most highly domain-dependent, it can be represented by a broad variety of linguistic manifestations. It includes elements such as the year of fabrication, colors, sizes, variants, and special features. Features can appear in almost any position in the product mention.

Table 2: Categories of product mention elements. Square brackets denote the extent of elements.

product mention. We will discuss the different elements a product mention can consist of, considering both semantic categories and their word class counterparts, as well as elements that are excluded from the product extent according to our annotation schema.

The ways in which a product can be mentioned in a text are manifold:

(3) a. *vehicle*
b. *SUV*
c. *Land Cruiser*
d. *Toyota Land Cruiser*
e. *Toyota Land Cruiser 100 Series VX*
f. *Toyota Land Cruiser 100 Series VX SUV*

All of these examples are possible ways to refer to the same real-world product and could appear as the product argument in a relation expressing a product the company "Toyota" sells. 3[a.] and 3[b.] are rather vague, describing a product category, 3[c.] and 3[d.] are more specific, distinguishing the car from all other brands by all other companies, and 3[e.] and 3[f.] are so specific that the product cannot be confused with another model.

Product mentions are generally realized as noun phrases, containing at least one proper noun or one common noun. As a proper noun, the head of the noun phrase can consist of individual letters or numbers or a series of numbers and/or letters:

(4) a. *AP3405*
b. *1500 ECL-PTU-208*
c. *Samsung 14nm LPP Process*

Often, the noun is accompanied by further distinctive attributes that can appear in different word classes as illustrated in the following examples:

(5) a. *smart sensors* (adjective)
b. *communicating sensors* (verb, gerund)
c. *vision sensors* (common noun)
d. *Hall sensors* (proper noun)

4.2. Elements of Product Mentions

We found a limited set of elements that products usually consist of. This set can be subdivided into seven categories: *company name, brand name, series, model, trademark symbol, type* and *feature*. Not all of these elements appear in every product mention. Product mentions can vary strongly in length and complexity, from a single element (6[a.-c.]), to a combination of any of the categories (6[d.-f.]) to a coverage of all of the categories (6[g.]).

(6) a. *sensors* (type)
b. *Kleenex* (brand name)
c. *Q7* (model)
d. *Audi Q7* (company name and model)
e. *Innocent Drinks smoothies* (company name and type)
f. *white iPhone 6* (feature, brand name and model)
g. *Toyota Land Cruiser 100 Series VX SUV diesel turbo* (all of the above)

As all of the examples used thus far have shown, some categories are more essential to a product mention than others. A product mention contains at least a common noun, representing the product *type*, or a proper noun that can either refer to a *brand name* or a specific *model*. Like brand names and models, the company name that often appears as part of the product mention (but is not essential to it) can also consist of a proper name and therefore include any kind of word class and even punctuation. While most of the seven categories can include or can be realized as nouns, *series* and *generations* as well as *models* tend to consist of letters or numbers or combinations of the two. Table 2 lists the further specification of the individual categories as well as examples for each category. In the table, the different categories of elements that we consider part of product mentions are listed in the order in which they usually appear when a product mention contains more than one category, with the exception of the category *feature* that can be found in any position.
Since it is not always apparent which category a part of a product mention falls into – sometimes the same part of a product mention could be assigned to two or more categories – the nested elements that constitute a product mention are currently not annotated, but only used by the annotator to determine the product mention's extent.

4.3. Elements Excluded from Product Mentions

One of the major results of our analysis is that there are a number of elements that often appear alongside a product mention and may be mistaken as part of it. Unless they are included in the proper name of the product (usually the company name or the brand name part), articles, prepositional phrases and prepositions, relative clauses and appositions are never considered part of the product mention extent. A more detailed discussion of these elements can be found in our annotation guidelines. We will only go into detail here regarding the more interesting, less clear-cut cases, namely company names, adjectives and other attributive elements as well as conjunctions and punctuation elements.
A company name that is used as the first argument in a *CompanyProvidesProduct* relation mention is considered part of the product extent if it does not come with a possessive marker. Since the line between a company name and a brand name can be blurred, we follow this rule to differentiate between cases of a nested relation mention (a relation mention within the product mention) and separate mentions of company and product. Usually, punctuation between words marks a product mention's boundary. This is not the case for hyphens if they connect different elements of a product mention. Commas and linking conjunctions can also serve as connectors when they list different elements of the same product, such as features or attributes. They can, however, also list different products. This merges into the aspect of attributes that often precede the head of the noun phrase that is the product mention. As discussed before, adjectives and other attributive elements are not considered part of the product extent unless they serve to define the product further. If they do, but there is more than one attribute fulfilling that function, we have to differentiate between products that are described by several attributes on the one hand and different product mentions that share a head but are distinguished by the attributes on the other. In the former case, the commas or linking conjunctions are included in the extent of the product, in the latter case they are not, but two – or more – product mentions are annotated. The following examples illustrate this issue:

(7) a. *[semiconductor] and [IP products]*
b. *[analog], [digital] and [mixed-signal integrated circuits]*
c. *[wireless and self-powered LED controls]*

Examples 7[a.] and [b.] contain attributes that are assigned to different products, whereas 7[c.] illustrates the case of two different attributes that specify the same product.

5. The CompanyProvidesProduct Relation

The *CompanyProvidesProduct* relation consists of two mandatory arguments, a *company* (organization) and a *product*, as well as of one optional argument, a *trigger*.
A company can serve as the first argument if it is stated as the creator, manufacturer, provider, distributor or vendor of the product argument. The slot for the second mandatory argument can be filled by one or more product mentions (e.g. in the case of conjunctive enumerations). Trigger concepts are a generic class of annotations that cover lexical expressions (terms or phrases) or syntactical elements (e.g. possessive marker *-s* or prepositional constructs) that indicate a specific event type.
The annotators were instructed to annotate only relation instances mentioned within a sentence. The following examples illustrate the relation annotation:

(8) a. *[Parkifi]$_{company}$ is a fast-growing technology company focused on [providing]$_{trigger}$ their customers with [real-time parking data]$_{product}$*
b. *[Sensata Technologies Holding]$_{company}$ [produces]$_{trigger}$ [sensors]$_{product}$*
c. *[BMW]$_{company}$['s]$_{trigger}$ [Z3]$_{product}$*
d. *[Intuition Executive]$_{product}$ [by]$_{trigger}$ [Honeywell]$_{company}$ collects and analyzes large amounts of data*
e. *[[Apple]$_{company}$[Watch Series 2]]$_{product}$*

Our annotation guidelines also consider some specific cases. For example, if a sentence contains a full-length company name followed and coreferenced by the company

abbreviation, then we label both mentions as individual company mentions, but only a single relation mention, between the full-length company mention and the product, is annotated. The relation between the company acronym and a product is implicitly given by the coreference information. Example 9 illustrates this issue: the company name *IS International Services LLC*, the trigger *providing* and the product *engineering services* are annotated as relation arguments, while the company's abbreviation *IS* are connected by the *Identity* relation.

(9) a. *[IS International Services LLC]$_{company}$ ([IS]$_{company}$) is a uniquely qualified business [providing]$_{trigger}$ [engineering services]$_{product}$*

Furthermore, if a sentence contains more than one trigger for the same relation instance, then as many relation mentions are annotated as there are triggers. Example 10 contains one company, one product, and the three triggers *developer*, *manufacturer* and *vendor*, each of them referring to a different way of how the product is related to the company – therefore three relation mentions are created.

(10) a. *FUJIFILM invested in [Japan Biomedical Co.]$_{company}$, a [developer]$_{trigger}$, [manufacturer]$_{trigger}$ and [vendor]$_{trigger}$ of [additives for cell culture media]$_{product}$.*

6. Corpus Statistics

	Total	Mean
# Documents	152	-
# Sentences	4001	26.3
# Words	131929	868.0
# Companies	2191	14.4
# Products	1717	11.3
# CompanyProvidesProduct	379	2.5

Table 3: Corpus Statistics

This section describes the corpus of documents annotated with product mentions, including product parts, technologies, and product classes, using the guidelines described in the previous section. Documents included in the corpus are sampled from the dataset that we described in Section 3. Table 3 lists some statistics of the current state of the corpus. The annotation is being carried out by two trained linguistics students. In cases of disagreement, a third expert annotator is consulted to reach a final decision. The current datasets consists of 152 documents with more than $131,000$ words. Thus far, $2,908$ entity mentions ($2,191$ organizations, $1,717$ products) have been annotated, and a total of 379 *CompanyProvidesProduct* relation mentions.
For the corpus annotation we use the markup tool Recon (Li et al., 2012), which allows annotating n-ary relations among text elements. Recon provides a graphical user interface that enables users to mark arbitrary text spans as entities, to connect entities to create relations, and to assign semantic roles to argument entities. Since the corpus is still in the process of being created, we cannot report any reliable inter-annotator agreement scores at the moment. We will include information about inter-annotator agreement at the entity and relation mention level in the final release. The corpus and the guidelines will be made available at `https://dfki-lt-re-group.bitbucket.io/product-corpus`. We distribute the dataset in an AVRO-based compact binary format, along with the corresponding schema and reader tools.

7. Conclusion

In this work we presented a fine-grained analysis and annotation schema for mentions of *product* entities and *CompanyProvidesProduct* relations in English web and social media texts. The schema is motivated by linguistic aspects and addresses the needs of recognizing industry- and product-related facts and relations. We presented a semi-automatic annotation process in order to ease the annotation procedure. While we have only annotated a small set of documents so far, the annotation effort to increase the size of the corpus is ongoing.

Acknowledgments

This research was partially supported by the German Federal Ministry of Economics and Energy (BMWi) through the projects SD4M (01MD15007B) and SDW (01MD15010A) and by the German Federal Ministry of Education and Research (BMBF) through the project BBDC (01IS14013E).

8. Bibliographical References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg. Springer-Verlag.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet Project. In *Proc. of ACL-COLING*, pages 86–90.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proc. of SIGMOD*, pages 1247–1250.

Derczynski, L., Bontcheva, K., and Roberts, I. (2016). Broad Twitter Corpus: A Diverse Named Entity Recognition Resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. (2004). The Automatic Content Extraction (ACE) Program - Tasks, Data, and Evaluation. In *Proc. of LREC*.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., Pennacchiotti, M., Romano, L., and Szpakowicz, S. (2010). SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July. Association for Computational Linguistics.

Kwon, D., Kim, S., Shin, S.-Y., Chatr-aryamontri, A., and Wilbur, W. J. (2014). Assisting manual literature curation for protein–protein interactions using bioqrator. *Database: The Journal of Biological Databases and Curation*, 2014.

Li, H., Cheng, X., Adson, K., Kirshboim, T., and Xu, F. (2012). Annotating opinions in german political news. In *8th ELRA Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA), 5.

Ling, X. and Weld, D. (2012). Fine-Grained Entity Recognition. In *Proc. of AAAI '12*.

Linguistic Data Consortium. (2005). ACE (Automatic Content Extraction) English Annotation Guidelines for Relations. `https://www.ldc.upenn.edu/collaborations/past-projects/ace`.

Liu, X., Zhang, S., Wei, F., and Zhou, M. (2011). Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 359–367, Portland, Oregon, USA, June. Association for Computational Linguistics.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June. Association for Computational Linguistics.

Ni, J., Dinu, G., and Florian, R. (2017). Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1470–1480, Vancouver, Canada, July. Association for Computational Linguistics.

Strötgen, J. and Gertz, M. (2013). Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Walter Daelemans et al., editors, *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September.

Weischedel, R. and Brunstein, A. (2005). Bbn pronoun coreference and entity type corpus ldc2005t33. web download.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., and Franchini, M. (2013). Ontonotes release 5.0 ldc2013t19. web download.

Portuguese Named Entity Recognition using Conditional Random Fields and Local Grammars

Juliana P. C. Pirovani, Elias de Oliveira
Universidade Federal do Espírito Santo - UFES
Av. Fernando Ferrari, 514, 29075-910 Vitória, ES, Brazil
juliana.campos@ufes.br,elias@lcad.inf.ufes.br

Abstract

Named Entity Recognition involves automatically identifying and classifying entities such as persons, places, and organizations, and it is a very important task in Information Extraction. Conditional Random Fields is a probabilistic method for structured prediction, which can be used in this task. This paper presents the use of Conditional Random Fields for Named Entity Recognition in Portuguese texts considering the term classification obtained by a Local Grammar as an additional informed feature. Local grammars are handmade rules to identify named entities within the text. The Golden Collection of the First and Second HAREM considered as a reference for Named Entity Recognition systems in Portuguese were used as training and test sets respectively. The results obtained outperform the results of competitive systems reported in the literature.

1. Introduction

Named Entity Recognition (NER) aims at automatically identifying and classifying entities such as persons, places, organizations and values. This is a fundamental task in Information Extraction since, besides having several applications, other tasks such as relations and events extraction, question answering systems and entity-oriented search depend on it as a preprocessing step (Jiang, 2012).

NER is not a simple task. Several categories of named entities (NEs) are written similarly and they appear in similar contexts. In addition, the same NE can be classified into different categories depending on the surrounding context and some entities do not appear even in large training sets. Thus, dictionaries are not always useful.

In 1995, the Message Understanding Conference (MUC-6) (Grishman and Sundheim, 1996) included the NER task for the first time for the English. Several similar events have emerged later such as the CoNLL (Tjong Kim Sang and De Meulder, 2003) and the HAREM (Santos and Cardoso, 2007; Mota and Santos, 2008). HAREM was an initiative for the Portuguese and the annotated corpora used in the First and Second HAREM, known as the Golden Collections (GC), are used as a golden standard reference for NER systems in Portuguese.

HAREM differs from other similar events in two aspects (Mota and Santos, 2008): the classification of an NE depends only on its use in context and more than one classification can be assigned to an NE. Moreover, the HAREM classifies 10 categories of NEs (Person, Place, Organization, Value, Time, Event, Abstraction, Work, Thing and Other). Thus, the HAREM presents a more demanding task and, therefore, the performance values obtained using its reference data sets are still lower compared to the others (Santos and Cardoso, 2007).

NER systems can be developed using the following approaches: linguistics, machine learning or hybrid. This work seeks to explore the potential of the linguistics and machine learning approaches by constructing a hybrid system for NER in Portuguese. The presented strategy, CRF+LG, combines a labeling obtained by a Conditional Random Fields (CRF) with a term classification obtained from Local Grammars (LGs). Sutton and McCallum in (Sutton and McCallum, 2012) say that an interesting type of feature for the CRF can be the result of simpler methods for the same task. Thus, in this work we apply LGs to perform the pre-labeling by capturing general evidence of NEs in texts and the CRF performs sequential labeling using this pre-labeling. The pre-labeling is sent to the CRF together with other features and can be seen as a suggestion for the CRF.

Conditional Random Fields (Lafferty et al., 2001) is a machine learning method, which has been successfully used in several Natural Language Processing (NLP) tasks, including NER. NER is treated as a sequence labeling problem and a conditional model is constructed from a training data set to predict which is the best labeling sequence given an input sentence.

Local Grammars are one means of representing the contextual rules of the linguistics approach. "Local grammars are finite-state grammars or finite-state automata that represent sets of utterances of a natural language" (Gross, 1999).

This paper is organized in 5 sections. Section 2 presents the state of the art and the Section 3 presents the methodology used in this work. The results of the study are presented in Section 4 and Section 5 presents conclusions and future works.

2. State of the Art

The systems presented in (Amaral, 2013) and (Santos and Guimaraes, 2015) achieved the best results for the 10 categories of the HAREM to date.

The NERP-CRF system, based on Conditional Random Fields (CRF), achieved the best Precision and F-Measure results compared to systems of the Second HAREM for the 10 categories (Amaral, 2013). NERP-CRF was also one of the four tools used to recognize NEs in Portuguese texts compared in (Amaral et al., 2014). The system obtained the best Precision results and the best performance for the Organization class considering only Person, Place and Organization categories.

In (Santos and Guimaraes, 2015), the authors proposed a language-independent system based on the CharWNN Deep Neural Network (DNN), which uses word-level and character-level representations to perform sequential classification. The approach presented better results compared to the ETL_{CMT} system, an ensemble method that uses Entropy Guided Transformation Learning (ETL).

A combination of K-Nearest Neighbors (KNN) and CRF for English NER in tweets was proposed in (Liu et al., 2011). Due to insufficient information in a tweet and the unavailability of training data, a semi-supervised learning and 30 gazetteers were used. The KNN classifier conducts a word-level classification and the labeled results are fed into a CRF together with other conventional features. The KNN and CRF models are repeatedly retrained with an incrementally augmented training set. The method showed advantages over the baselines.

In (Constant and Tellier, 2012), the authors propose to evaluate the impact of external lexical resources into a CRF in order to perform the joint task of multiword segmentation and part-of-speech tagging in French. The information coming from dictionaries and local grammars recognizing numerical determiners and some NEs like organization or place was coupled as features into a CRF in two different ways: concatenating each possible POS category (Learn-concat) and considering each possible category in the resources as a new boolean property (Learn-bool). They obtained a gain of 0.5% in terms of F-measure and showed that the integration of lexicon-based features significantly compensates the use of a small training corpus.

This paper aims to perform the NER for the 10 categories of the HAREM using CRF as it was carried out in (Amaral, 2013); however, the preprocessing of the texts was performed differently and an initial information about the label of each word was obtained by LGs and added to the feature set sent to the CRF training phase.

This work also differs from those presented in (Liu et al., 2011) and (Constant and Tellier, 2012) by combining a rules-based approach with CRF for Portuguese NER and by not using gazetteers or dictionaries. To the best of our knowledege, there is not yet a work that combines LG (or other lexical resources) and CRF for NER in Portuguese.

3. The Methodology

In this work, the GC of the First and Second HAREM were used as training and test sets respectively. Both GC have 129 texts written in Portuguese and are available in (Linguateca, 2017).

During the training phase, initially each input file is splitted into sentences by the tool Unitex[1]. Unitex uses LGs to describe the different ways that indicate the end of a sentence. For this work, the LG that performs sentence segmentation in Unitex was changed to not split the sentences in a colon (:) and a semicolon (;).

A copy of the segmented files has their tags removed since the GC used has the NEs tags. An LG is applied to these files without any markup and the NEs identified by it are annotated.

On the other hand, the segmented files are tokenized using the OpenNLP[2] library. In order to represent the NER as a sequence labeling problem, a label must be assigned to each text token. Several notations can be used to delimit NEs and identify tokens in text (Konkol and Konopík, 2015), but the IO notation was chosen because it presented better results in previous tests performed during this work.

The IO notation is used as follows: all tokens which are part of the NE are then labeled with I (Inside) and all other tokens with O (Outside or Other). In this case, the class of the NE is also mentioned in label I as shown in Table 1.

Table 1: IO notation for the sentence *Meu pai é Gabriel Raimundo da Silva* (My father is Gabriel Raimundo da Silva)

(Token IO-Notation)
(Meu O) (pai O) (é O) (Gabriel I-PERSON) (Raimundo I-PERSON) (da I-PERSON) (Silva I-PERSON) (. O)

Next, several features are added for each token of the files. These features are used during supervised learning of the CRF prediction model. The features used were the same proposed by (Amaral, 2013), in addition to that feature corresponding to the label assigned by LG. The feature set is presented in Table 2.

The POS-Tagging of a word corresponds to its grammatical class and it was also assigned by the OpenNLP library. When a word does not have one of the previous words (p-1 or p-2) or posterior (p+1 or p+2), the corresponding feature values are "null". Table 3 presents an example of a vector of features.

The methodology used for testing is similar. The difference is that the input files do not have the NEs tags. In addition to the files containing the tokens and features, the CRF receives the previously trained model to predict a label for each token.

3.1. Local Grammars (LG)

An LG created in Unitex is represented as a set of one or more graphs. The LG built in this work consists of 10 graphs, one for each of the NEs categories considered by HAREM.

We observed in the training file in which context each type of NE appeared, what words could somehow indicate the existence of NE to construct each graph. We observed that, for example, words with the first letter capitalized preceded by the preposition *em* (in) were labeled as Place. We also observed that some NEs of the Person category are preceded by words such as *diz* (say), *explicou* (explained), *afirmou* (said), etc.

Thus, the graphs created capture some simple heuristics to the recognition of NEs in the training set. An example of rule in the graph created for the Person category is presented in Figure 1.

[1] http://unitexgramlab.org/

[2] http://opennlp.apache.org/

Table 2: Feature set assigned to each token

Features	Description
word	current word (position p)
tag	POS-Tagging of the word corresponding to its grammatical class
cap	if the word is composed of only capital letters, only lowercase or mixed
ini	if the word starts with uppercase, lowercase or symbols
simb	if the word is composed of symbols, digits or letters
prevW, prevT, prevCap	word, tag and cap for the word in position p-1
prev2W, prev2T, prev2Cap	word, tag and cap for the word in position p-2
nextW, nextT, nextCap	word, tag and cap for the word in position p+1
next2W, next2T, next2Cap	word, tag and cap for the word in position p+2
tip	label assigned by LG to the word

Table 3: Example of a vector of features to the Gabriel token in sentence *Meu pai é Gabriel Raimundo da Silva*

Token	Vector of features	IO Notation
Gabriel	word=Gabriel tag=prop cap=maxmin ini=cap simb=alpha prevW=é prevT=v-fin prevCap=min nextW=Raimundo nextT=n nextCap=maxmin prev2W=pai prev2T=n prev2Cap=min next2W=da next2T=v-pcp next2Cap=min tip=I-PERSON	I-PERSON

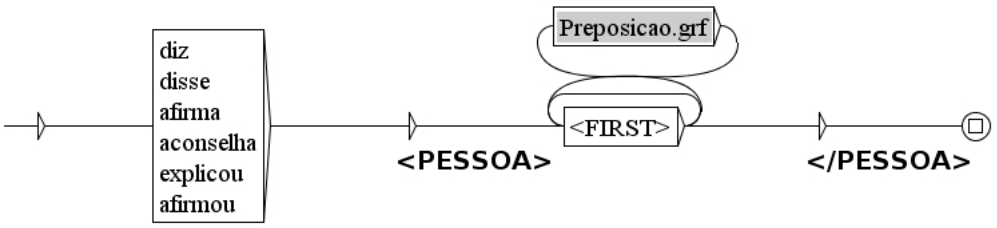


Figure 1: Example of rule in the graph that recognizes the Person category

This graph recognizes words such as *diz* (say) or *afirmou* (said) followed by words with the first letter capitalized, as identified by the code <FIRST> in Unitex dictionaries. Among words with the first letter capitalized, prepositions may appear whose recognition has been previously detailed in graph Preposicao.grf included as subgraph. Examples of occurrences identified by this graph were:
diz <PESSOA> Moncef Kaabi </PESSOA>
afirmou <PESSOA> José SÓCRATES</PESSOA>
afirma <PESSOA> Jason Knight </PESSOA>.
Note that identified person will appear between the tags <PESSOA> (<PERSON>) and </PESSOA> in the concordance file containing the list of occurrences identified.

3.2. Conditional Random Fields (CRF)

Conditional Random Fields (CRF) is a machine learning method for structured prediction proposed by (Lafferty et al., 2001). It is used for labeling of sequential data based on a conditional approach.
Let X = (x_1, x_2, ..., x_n) be a sequence of words in a text, we want to determine the best sequence of labels Y = (y_1, y_2, ..., y_n) for these words, corresponding to the categories of NEs (10 categories of the HAREM or the label O in this work). The CRF models a conditional distribution $p(Y|X)$ that represents the probability of obtaining the output Y given the input X.
In this work, we used a linear-chain CRF that predict the output variables Y as a sequence for sequences of input variables X. According to (Sutton and McCallum, 2012), a linear-chain CRF is a conditional distribution that takes the form shown in Equation 1:

$$p(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\} \quad (1)$$

where Z(x) is a normalization function given by Equation 2:

$$Z(x) = \sum_{y} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\} \quad (2)$$

$F = \{f_k(y_t, y_{t-1}, \mathbf{x}_t)\}_{k=1}^{K}$ is a set of feature functions that must be fixed according to the problem. An example is

a function which takes the value 1 when the word begins with a capitalized letter (component of the input vector $\mathbf{x}_t$), its label is Person (y_t) and the previous label (y_{t-1}) is Other and 0 otherwise. The vector $\mathbf{x}_t$ contains all the components of the global observations x that are needed for computing features at time t. $\theta = \{\theta_k\}$ is a vector of weights that must be estimated from the training set. This is usually done by maximum likelihood learning. The weights depend on each feature function and the more discriminating the function, the higher its computed weight will be.

The MALLET[3] toolkit was used in this work to estimate the vector of weights and then apply the CRF model obtained to label the test set. This CRF model combines the weights of each feature function to determine the probability of a certain value (y_t).

4. Results and Discussion

The annotated files by CRF+LG and NERP-CRF (Amaral, 2013) were submitted to SAHARA[4] for performance evaluation. SAHARA is an online system for automatic evaluations of the HAREM. This system computes metrics of Precision (P), Recall (R) and F-Measure (F) of an NER system after submitting annotated XML files and configures the evaluation desired by the user. In this setting, the evaluation mode is chosen, denoting which task should be evaluated: identification that evaluates only if the recognized string is actually an NE; or classification that in addition to checking the boundaries of the NE also checks if the category is correct. The only modification made to the default configuration was the assignment of the value zero for the β and γ parameters because they correspond to the types and subtypes of the categories that were not classified in this work.

The file annotated by NERP-CRF was obtained as indicated in (Amaral et al., 2014)[5]. We modified the identifiers (ID) of each NE by adding a unique number at the end for the evaluation in the SAHARA due to the NERP-CRF uses the same ID for all NEs in a document and this changes the actual system performance computed by SAHARA. Note that, when a unique ID is not assigned to every NE, the computed metrics do not consider all false positives, only one per document that has false positive. We realized this by studying the evaluation architecture of the Second HAREM and analyzing the files generated by each module.

The results obtained overcome the NERP-CRF results in more than 10% for the Recall metric in the identification task (Table 4) and more than 8% in the classification task (Table 5). For the F-Measure metric, CRF+LG overcome the NERP-CRF results in more than 8% and 7% in the identification and classification tasks respectively, representing considerable gain.

The authors in Santos and Guimaraes (Santos and Guimaraes, 2015) used the GC of the First HAREM as training set and the MiniHAREM as the test set. As (Santos and Guimaraes, 2015) did not present the results for the GC of the Second HAREM, CRF+LG was rerun using the GC that they used for training and testing. The CoNLL-2002[6] script that evaluates the classification task was also used as done by those authors to compute the metrics in our experiments. The selective scenario (categories Person, Place, Organization, Time and Value) was considered in this case because the results presented for the 10 categories of the HAREM were obtained using word-level embeddings previously trained by (Santos and Zadrozny, 2014) who used three other corpus (Portuguese Wikipedia, CETENFolha and CETEMPublico) to perform this unsupervised pre-training. Therefore the comparison with this result would be unfair since the CRF+LG uses only the GC of the First HAREM for the CRF training phase and LG construction. Hence, just for the sake of comparison, the GC of the First HAREM has approximately 78667 words while only the CETEMPublico, one of the three corpus used by (Santos and Guimaraes, 2015), has about 180 million words.

The results are presented in Table 6. Note that CRF+LG achieved a gain of approximately 2% in each metric evaluated.

Table 4: Comparison with NERP-CRF - Identification

Systems	P (%)	R (%)	F(%)
NERP-CRF	73.68	53.79	62.19
CRF+LG	**78.58**	**64.12**	**70.62**

Table 5: Comparison with NERP-CRF - Classification

Systems	P (%)	R (%)	F(%)
NERP-CRF	61.04	43.18	50.57
CRF+LG	**65.46**	**51.75**	**57.8**

Table 6: Comparison with CharWNN

Systems	P (%)	R (%)	F(%)
CharWNN	65.21	52.27	58.03
CRF+LG	**67.09**	**54.85**	**60.36**

We observed some errors when analyzing false positives and false negatives obtained by CRF+LG: prepositions like *de* (of) and conjunctions like *e* (and) that are not considered part of NEs since they are also common outside NEs (e.g., *Joaninha Sampaio* labeled as Person when the name was *Joaninha Sampaio e Melo*); names labeled as Person names when they are part of a larger NE (e.g., *José Mourinho* in *Liderança - As Lições de José Mourinho* that should be labeled as Work); capitalized words labeled as Organization (e.g., *FESTA* which is not NE).

[3] http://mallet.cs.umass.edu/

[4] http://www.linguateca.pt/harem/

[5] http://www.inf.pucrs.br/linatural/recursos_para_reconhecimento_de_entidades_nomeadas/NERP_CRF.xml

[6] http://www.cnts.ua.ac.be/conll2002/ner/bin/conlleval.txt

Several errors occurred due to some inconsistencies in the GC of the First HAREM and Second HAREM. For example, in the GC of the First HAREM, strings as "2004" preceded by the preposition *em* (in) are considered NEs of the Time category and the CRF+LG learned this and labeled all similar strings preceded by *em* as Time. However, in the GC of the Second HAREM, the preposition *em* is part of the NE. So all these NEs were wrongly labeled. The same happened in other situations of the categories Time, Value and Person.

5. Conclusions

This paper presented a hybrid approach for the Named Entity Recognition in Portuguese texts using Conditional Random Fields and Local Grammars. The term classification obtained initially from LG was sent as a feature for the learning process of the CRF prediction model together with other features. The CRF model performs the final labeling of the NEs. Our approach is a good way to consider the human expertise for capturing the rules that do not appear in examples of the annotated corpus used for training by the CRF.

The results obtained outperform the results of competitive systems reported in the literature when performing under equivalent conditions. It is important to mention that these are the results for a small corpus and the gains can become more expressive when using a larger corpus for training.

In future work, we will investigate the impact of some preprocessing decisions on the performance of the CRF and the impact of using the result of other classifiers to inform new features for the CRF learning process rather than an LG.

6. Bibliographical References

Amaral, D. O., Fonseca, E. B., Lopes, L., and Vieira, R. (2014). Comparative analysis of portuguese named entities recognition tools. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2554–2558.

Amaral, D. O. F. d. (2013). O reconhecimento de entidades nomeadas por meio de conditional random fields para a língua portuguesa. Master's thesis, Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brazil.

Constant, M. and Tellier, I. (2012). Evaluating the impact of external lexical resources into a crf-based multiword segmenter and part-of-speech tagger. In *8th International Conference on Language Resources and Evaluation (LREC'12)*, pages 646–650.

Grishman, R. and Sundheim, B. (1996). Message understanding conference-6: A brief history. In *COLING*, volume 96, pages 466–471.

Gross, M. (1999). A Bootstrap Method for Constructing Local Grammars. In Neda Bokan, editor, *Proceedings of the Symposium on Contemporary Mathematics*, pages 229–250. University of Belgrad.

Jiang, J. (2012). Information extraction from text. In *Mining text data*, pages 11–41. Springer.

Konkol, M. and Konopík, M. (2015). Segment representations in named entity recognition. In *International Conference on Text, Speech, and Dialogue*, pages 61–70. Springer.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML 2001*, volume 1, pages 282–289.

Linguateca. (2017). `http://www.linguateca.pt/HAREM/`. Accessed 27/01/2017.

Liu, X., Zhang, S., Wei, F., and Zhou, M. (2011). Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 359–367. Association for Computational Linguistics.

Mota, C. and Santos, D. (2008). Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM.

Santos, D. and Cardoso, N. (2007). Reconhecimento de entidades mencionadas em português: Documentação e actas do harem, a primeira avaliação conjunta na área.

Santos, C. N. d. and Guimaraes, V. (2015). Boosting named entity recognition with neural character embeddings. In *Proceedings of the Fifth Named Entities Workshop, ACL 2015*, pages 25–33.

Santos, C. N. d. and Zadrozny, B. (2014). Learning character-level representations for part-of-speech tagging. In *ICML*, pages 1818–1826.

Sutton, C. and McCallum, A. (2012). An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans et al., editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

M-CNER: A Corpus for Chinese Named Entity Recognition in Multi-Domains

Qi Lu, YaoSheng Yang, Zhenghua Li, Wenliang Chen, Min Zhang
School of Computer Science and Technology
Collaborative Innovation Center of Novel Software Technology and Industrialization
Soochow University, Suzhou 215006, China
luqibhf@qq.com, ysyang@stu.suda.edu.cn,
{ zhli13, wlchen, mingzhang}@suda.edu.cn

Abstract

In this paper, we present a new corpus for Chinese Named Entity Recognition (NER) from three domains : human-computer interaction, social media, and e-commerce. The annotation procedure is conducted in two rounds. In the first round, one sentence is annotated by more than one persons independently. In the second round, the experts discuss the sentences for which the annotators do not make agreements. Finally, we obtain a corpus which have five data sets in three domains. We further evaluate three popular models on the newly created data sets. The experimental results show that the system based on Bi-LSTM-CRF performs the best among the comparison systems on all the data sets. The corpus can be used for further studies in research community.

Keywords: Named Entity Recognition; Chinese Data Set; Information Extraction

1. Introduction

In recent years, there has been significant progress on the task of Named-Entity Recognition (NER) by using sequence labeling models in the settings of supervised learning, such as CRF and LSTM-CRF (Lafferty et al., 2001; Huang, Xu, and Kai 2015). NER is one of the most important natural language processing (NLP) tasks. Its performance highly affects further applications, such as relation extraction (Bunescu and Mooney, 2005), and question answering (Jurafsky and Martin, 2009).

As with the setting of supervised learning, building NER systems needs a massive amount of labeled training data which are often annotated by humans. However, for most languages, large-scale labeled datasets are only readily available in some domains, for example the news domain. For other domains like social media and dialog texts, there is a lack of such data sets. The NER systems trained on the news domain often perform worse in other domains. It is a reasonable solution to create human-annotated data in new domains to improve the performance of NER system.

In this paper, we present a new corpus for Chinese Named Entity Recognition in multi-domains, named M-CNER. We create several data sets in three domains: human-computer interaction, social media, and e-commerce, which are often used in real applications. We require the annotators to label some predefined entities. In the annotation procedure, the annotators label the sentences independently in the first round. One sentence is labeled by more than one persons. In the second round, experts check the entities which have disagreement among the annotators. The detailed settings are described in Section 2.

Most traditional high performance sequence labeling models for NER are linear statistical models, including Hidden Markov Models (HMM) and Conditional Random Fields (CRF) (Ratinov and Roth, 2009; Passos et al., 2014; Luo et al., 2015). CRF has been widely used for this task for the last decades, but in the most recent years, non-linear neural networks have become popular for NER. For example, Collobert et al. (2011) propose a simple but effective feed-forward neutral network that independently assigns the NE labels for each word by using contexts within a window with fixed size. Recurrent neural networks (RNN) (Goller and Kuchler, 1996), together with its variants such as long-short term memory (LSTM) (Hochreiter and Schmidhuber, 1997; Gers et al., 2000) and gated recurrent unit (GRU) (Cho et al., 2014), have shown great success in the task of NER. Among the above models, we choose three models including CRF, LSTM, and LSTM-CRF for comparisons.

In the experiments, we evaluate the three systems on the newly created data sets which are from different domains. The experimental results show that LSTM-CRF performs the best among the systems. The new corpus can be used for further studies in research community.

2. M-CNER Data

In this section, we describe how we create a corpus for Chinese Named Entity Recognition in Multi-domains, named M-CNER. We collect the sentences from three domains in Chinese: human-computer interaction (HCI), social media (SM), and e-commerce (ECO).

2.1 Annotation Procedure

In our annotation procedure, there are two rounds. In the first round, we hire several undergraduate students to perform annotation. They read guideline documents which describe the definitions of the predefined entity types. For each type, we additionally provide about 20 exemplifying sentences to help the annotators understand the definitions. Then they identify the named entities in the sentences and classify them as one of the predefined types. After the first round, there are some sentences for which the annotators give out different annotations. For those sentences, we let experts check the disagreed annotations carefully. Finally, the experts reach the agreements for all the cases with discussion.

2.2 Domains

2.2.1 Human-Computer Interaction (HCI)

In Human-Computer Interaction (HCI) domain, we collect raw sentences from an intelligent robotic company. And then we randomly select some sentences as our annotation

pool. We ask the annotators to label two types of entities: Person-name (PER) and Music-song (MUS). We create two data sets, HCI-PER for PER and HCI-MUS for MUS respectively. The annotators label the sentences independently and each sentence is assigned to three annotators. After annotation, we remove some illegal sentences reported by the annotators. Finally, we have 12,204 sentences for HCI-PER and 10,510 sentences for HCI-MUS.

Data set	#Sent	#Entity-type	#Entity-annotated
HCI-PER	12204	1	14359
HCI-MUS	10510	1	3646
SM-Weibo	3890	3	9534
ECO-Title	2323	5	10158
ECO-Query	2297	5	2665

Table 1: The information of M-CNER

Table 1 shows the information of annotated data, where #Sent refers to the number of sentences in the data set, #Entity-type refers to the number of prefined entity types, and #Entity-annotated refers to the number of entyties annoated in the sentences.

In the HCI dataset, HCI-PER includes: 1) full name, for example {习近平@PER}主席访美(President {Xi Jinping@PER} visited USA); 2) Surname+Title, for example {习主席@PER}访美({President Xi@PER} visited USA); 3) Musical ensemble, for example {羽泉@PER}({Yu-Quan@PER); 4) Nickname, for example 真甜啊，我的{甜心@PER }(You are so sweet, my {sweet-heart@PER}). HCI-MUS includes full song names.

2.2.2 Social Media (SM)

As for Social Media (SM) domain, the raw sentences are from the messages on Sina-Weibo (weibo.com). We apply the similar strategy as HCI to annotate the sentences. Three types of entities are defined: Person-name (PER), Orgnization-name (ORG), and Location (LOC). Peng and Dredze (2015) created a data set on Sina-Weibo, but the size is small. We additionally add 2,000 Weibo messages with the same entity definition. As for annotation guideline, we follow the definition of Peng and Dredze (2015). Totally, we get 3,890 messages for this domain as shown in Table 1. We treat one message as one sentence in the experiments.

2.2.3 E-Commerce (ECO)

As for E-Commerce (ECO) domains, we collect the sentences from an e-commerce plaform. The sentences are from two parts : one is titles of products (ECO-Title) and another is user queries (ECO-Query). We also use the similar annotation strategy for this domain. We seperate the sentences into two data sets : ECO-Title and ECO-Query, because the styles of sentences from two parts are quite different. Five types of entities are defined : brand, product, model, specifications, and material. Finally, we have 2,323 sentences for ECO-Title and 2,297 for ECO-Query as shown in Table 1.

We list some examples of ECO-Title and ECO-Query in Table 2, where we give one Title example and one Query example for each type.

2.3 Data Splits

For our experiments, we split the data into three parts : training, development, and test sets. Table 3 shows the detailed information of data splits for M-CNER. For HCI and ECO domains, we use the percentage 8 :1 :1 for three parts. As the SM data, we use newly annotated 2000 messages as training data, the data created by Peng and Dredze (2015) as development and test data.

Type	Examples
Brand	Title : {品胜@Brand}移动电源适用于{苹果@Brand}{华为@Brand }{OPPO @Brand} {Pingshen@Brand} mobile power for {Apple@Brand}, {Huawei@Brand}, and {OPPO@Brand} Query: {华为@Brand}和{荣耀@Brand}，哪个系列漂亮? {Huawei@Brand} and {Honer@Brand}, which series are beautiful?
Product	Title : 苹果 iphone8 全网通 4G{手机@product} Apple iphone8 Full Netcom 4G {mobile phone@Product} Query: 我想买牛肉味的{兰花豆@product} I want to buy beef flavor {orchid beans@Product}
Model	Title: 苹果{iphone8@Model}全网通 4G手机 Apple {iphone8@Model} Full Netcom 4G mobile phone Query: {iphone8@Model}比{iphone7 @Model}有哪些提升 Which features are {iphone8@Model} better than {iphone7@Model}
Material	Title: 英伦日常百搭【牛皮@Material】鞋子 British daily {cowskin@Material} shoes Query: 有亚麻裤子吗？ Do you have {flax@Material} pants?
Specif	Title: 镜片护理液{3 瓶@ Specif }{120ml @ Specif }装 lens care solution {3 bottles@Specif} {120ml@ Specif} Query: 我要买{三箱@ Specif }牛奶 I want to buy {three boxes@ Specif} of milk

Table 2 : Examples of ECO data

3. Comparison Approaches

In this paper, we compare the performance of three systems which are frequently used in the task of NER on the newly created data sets. The first one is a traditional system based on the CRF model (Lafferty et al., 2001), the other two are based on neural networks: Bi-LSTM without/with the CRF layer. We describe the models briefly since full details are presented in the related papers.

Domain	Train	Dev	Test
HCI-PER	10023	1114	997
HCI-MUS	8510	1000	1000
SM-Weibo	2000	890	1000
ECO-Title	1863	230	230
ECO-Query	1837	230	230

Table 3: The data splits of M-CNER

3.1 CRF

For sequence labeling (or general structured prediction) tasks, the performance can be improved by considering the correlations between labels in neighborhoods and the system jointly generates the best chain of labels for a given input sentence. For example, in the sequences with standard BIO2 schema (Tjong Kim Sang and Veenstra, 1999), I-ORG cannot follow I-PER. We build a NER system by using a conditional random field (CRF) model (Lafferty et al., 2001) which performs very well in the task of NER.

Formally, we use $\mathbf{x}=\{x_1, \cdots, x_n\}$ to represent a generic input sequence where x_i refers to the ith word. $\boldsymbol{y} = \{y_1, \cdots, y_n\}$ represents a generic sequence of labels for **x**. $\mathcal{Y}(\boldsymbol{x})$ denotes the set of possible label sequences for **x**. The probabilistic model calculates the conditional probability $p(\boldsymbol{y}|\boldsymbol{x})$ over all possible label sequences **y** given **x** with the following form:

$$\mathrm{p(y|x)} = \frac{1}{Z(x)} \exp\left(\sum_{i,k} w_k f_k(y_{i-1}, y_i, x) + \sum_{i,k} w_k{}' f_k{}'(y_i, x)\right)$$

where Z(x) is the normalization constant, f_k is a binary feature function, and w_k is the weight of f_k. Given the training data, the parameters of the model are trained to maximize the conditional log-likelihood. In the testing stage, given a sentence x in the test data, the tagging sequence y* is given by,

$$y^* = \arg max_{y'} p(y'|x)$$

CRF allows us to utilize a large number of observation features as well as different state sequence based features and other features we want to add. For a sequence CRF model (only interactions between two successive labels are considered), training and decoding can be solved efficiently by adopting the Viterbi algorithm. As for the feature templates, we use the supervised version of Zhao et.al (2008).

3.2 Bi-LSTM

In this section, we introduce the system based on bi-directional without the CRF layer.

3.2.1 LSTM Unit

Recurrent neural networks (RNNs) are a powerful family of connectionist models that capture time dynamics via cycles in the graph. Though, in theory, RNNs are capable to capturing long-distance dependencies, in practice, they fail due to the gradient vanishing/exploding problems (Bengio et al., 1994; Pascanu et al., 2012).

LSTMs (Hochreiter and Schmidhuber, 1997) are variants of RNNs designed to cope with these gradient vanishing problems. Basically, a LSTM unit is composed of three multiplicative gates which control the proportions of information to forget and to pass on to the next time step.

Formally, the formulas to update an LSTM unit at time t are:

$$i_t = \sigma(W_i h_{t-1} + U_i x_t + b_i)$$
$$f_t = \sigma\left(W_f h_{t-1} + U_f x_t + b_f\right)$$
$$\tilde{c}_t = \tanh(W_c h_{t-1} + U_c x_t + b_c)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$
$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o)$$
$$h_t = o_t \odot \tanh(c_t)$$

where σ is the element-wise sigmoid function and $\odot$ is the element-wise product. x_t is the input vector (e.g. char embedding) at time t, and h_t is the hidden state (also called output) vector storing all the useful information at (and before) time t. U_i, U_f, U_c, U_o denote the weight matrices of different gates for input x_t, and W_i, W_f, W_c, W_o are the weight matrices for hidden state h_t. b_i, b_f, b_c, b_o denote the bias vectors. It should be noted that we do not include peephole connections (Gers et al., 2003) in the our LSTM formulation.

3.2.2 Bi-LSTM

For many sequence labeling tasks it is beneficial to have access to both past (left) and future (right) contexts. However, the LSTM's hidden state h_t takes information only from past, knowing nothing about the future. An elegant solution whose effectiveness has been proven by previous work (Dyer et al., 2015) is bi-directional LSTM (Bi-LSTM). The basic idea is to present each sequence forwards and backwards to two separate hidden states to capture past and future information, respectively. Then the two hidden states are concatenated to form the final output. We treat NER as a classification problem in the final stage.

3.3 Bi-LSTM-CRF

Then, we can add a CRF layer to the Bi-LSTM model as shown in Figure 1. That is Bi-LSTM-CRFs (Huang, Xu, and Kai 2015) which are well-suited for sequence labeling. Bi-LSTM-CRF can be regarded as a combination of bidirectional LSTM and CRF.

By contrast to the local classification, CRFs (Lafferty, McCallum, and Pereira 2001) have the advantage of modeling at the sentence level instead of individual

positions. Finally, we feed the output of Bi-LSTM into the CRF layer directly for NER decoding.

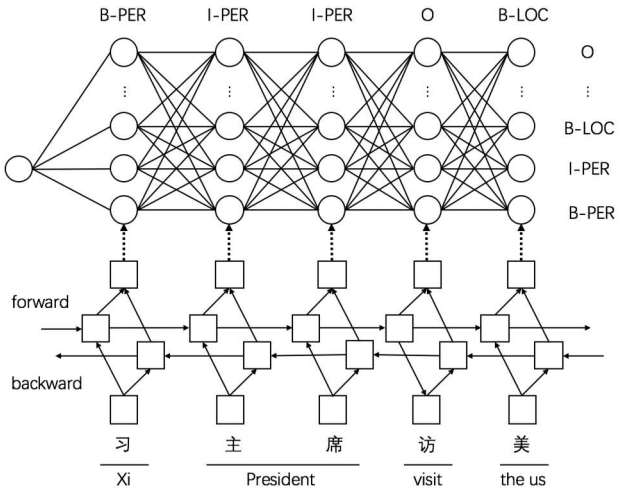


Figure 1: The framework of Bi-LSTM-CRF

3.4 Settings for Neural Networks

Lexical embeddings represent words in a continuous low dimensional space, which can capture semantic or syntactic properties of the lexicon. Similar words would have similar low dimensional vector representations. Embeddings have been used to gain improvements in a variety of NLP tasks. In NER specifically, several studies have shown improvements by using pre-trained neural embeddings as features in standard NER systems (Collobert and Weston, 2008; Turian et al., 2010; Passos et al., 2014). More recently, these improvements have been demonstrated on Twitter data (Cherry and Guo, 2015). Embeddings are especially helpful when there is little training data, since they can be trained on a large amount of unlabeled data.

However, training embeddings for Chinese is not straightforward: Chinese is not word segmented, so embeddings for each word cannot be trained on a raw corpus. Additionally, the state-of-the-art systems for downstream Chinese tasks, such as NER, may not use words. Thus, we use character embeddings for our systems instead.

3.4.1 Character Embeddings

We learn an embeddings for Chinese characters in the training corpus (Sun et al., 2014; Liu et al., 2014). This setting does not require pre-processing the text, and better fits our task studied in this paper: Chinese NER tagging over characters. Since there are many fewer characters than words in Chinese, we can reduce the size of embeddings. On the one hand, this means fewer parameters and less over-fitting. However, the reduction in parameters comes with a loss of specificity, where we may be unable to learn different behaviors of a character in different settings. We explore a compromise approach in the next section. The embeddings are directly incorporated into the NER system by adding embedding features for each character.

For each of the embeddings, we fine-tune pretrained embeddings in the context of the NER task. This corresponds to initializing the embeddings parameters using a pre-trained model, and then modifying the parameters during gradient updates of the NER model by back-propagation gradients. This is a standard method that has been previously explored in sequential and structured prediction problem (Collobert et al., 2011; Zheng et al., 2013; Yao et al., 2014; Pei et al., 2014).

3.4.2 Training Settings

To train model parameters, we exploit a negative log likelihood objective as the loss function. We apply softmax over all candidate output label sequences and use standard back-propagation method to minimize the loss function of the CRF model.

4. Experiments

In this section, we evaluate the three systems on M-CNER, our newly created data sets.

4.1 Evaluation Setup

The vector representations of characters are basic inputs of our systems based on neural networks, which are listed by the looking-up table. We use pretrained embeddings trained on large-scale raw corpus. For the systems on SM domain, we use a data downloaded from the site of Sina-Weibo, having 5M messages. And for the systems on HCI and ECO domains, we use a data from the user-generated content from Web, having 5M sentences. For training the embeddings, we use the word2vec in the experiments.

As for the hyper-parameters, we tune them on the development set. After tuning, we set the dimension size of character embeddings as 100, the dimension sizes of all the other hidden layers also as 100, the mini-batch size as 128, and the learning rate is 0.01. We adopt the dropout technique to avoid overfitting by a drop value of 0.2.

We report the scores by precision, recall, and F1 as the previous studies did.

4.2 Main Results

In this section, we show the model performances of the comparison systems. Table 4 shows the experimental results on all the datasets, where BM and BM-CRF refer to Bi-LSTM without/with the CRF layer, respectively.

From the table, we find that the CRF model provides better scores on precision than Bi-LSTM and Bi-LSTM-CRF while Bi-LSTM-CRF performs the best on F1 in the most cases except for SM-Weibo. In average, Bi-LSTM-CRF achieves better performance with absolute score +4.31% than CRF. This indicates that the recent neural networks models are more powerful for NER. The results of Bi-LSTM and Bi-LSTM-CRF also show that adding the CRF layer to Bi-LSTM is very important for improving the performance.

The information of Table 1 shows that the numbers of sentences in HCI-PER and HCI-MUS are much larger than the ones in SM-Weibo, ECO-Title and ECO-Query. The results from Table 4 show that the F1 scores are only on the level of 40-60% on SM-Weibo, ECO-Title and ECO-Query, while the scores are around 90% for HCI domains. We will label more sentences in SM and ECO domains to improve the performance further in future work.

Data	Model	Precision	Recall	F1
HCI-PER	CRF	**95.37**	79.14	86.5
	BM	83.18	83.57	83.37
	BM-CRF	90.8	**89.74**	**90.27**
HCI-MUS	CRF	**88.51**	77.74	83.89
	BM	73.90	80.80	77.20
	BM-CRF	87.77	**84.64**	**86.17**
SM-Weibo	CRF	**69.53**	40.19	**50.94**
	BM	32.38	31.99	32.18
	BM-CRF	50.54	**43.82**	46.94
ECO-Title	CRF	**71.81**	31.47	43.76
	BM	48.97	52.28	50.57
	BM-CRF	63.23	**54.99**	**58.82**
ECO-Query	CRF	**65.84**	47.32	55.06
	BM	45.99	**58.93**	51.66
	BM-CRF	61.03	58.04	**59.50**
Average	CRF	**78.21**	55.17	64.03
	BM	56.88	61.51	58.99
	BM-CRF	70.67	**66.25**	**68.34**

Table 4: Main results on M-CNER

5. Conclusion

In this paper, we have presented a new corpus for Chinese Named Entity Recognition in three domains, named M-CNER. The data sets are first labeled by the annotators and then reach the agreements via the discussion by the experts. We evaluate three popular systems on our newly created corpus. The experimental results show that Bi-LSTM-CRF performs better than the other two systems. This new corpus can be used as evaluation benchmark for research community and we can build better Chinese NER systems for the three domains.

In future work, we plan to add more sentences to SM and ECO domains, create more data set for other domains and define more types of entities. We will also build some state-of-the-art systems for comparisons on our data sets.

Acknowledgments

The research work is supported by the National Natural Science Foundation of China (61572338) and the Natural Science Foundation of the Jiangsu Higher Education Institutions(Grant No. 16KJA520001). The corresponding author is Wenliang Chen.

References

Lafferty J D, Mccallum A, Pereira F C N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data[C]. In ECML. Morgan Kaufmann Publishers Inc. 2001:282-289.

Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging[J]. Computer Science, 2015.

Bunescu R C, Mooney R J. A shortest path dependency kernel for relation extraction[C]. In EMNLP. Association for Computational Linguistics, 2005:724-731.

Jurafsky D, Martin J H. Speech and Language Processing (2nd Edition)[M]. Prentice-Hall, Inc. 2009.

Ratinov L, Roth D. CoNLL '09 Design Challenges and Misconceptions in Named Entity Recognition[C]. In CoNLL. 2009:147--155.

Passos A, Kumar V, Mccallum A. Lexicon Infused Phrase Embeddings for Named Entity Resolution[J]. Computer Science, 2014.

Luo G, Huang X, Lin C Y, et al. Joint Entity Recognition and Disambiguation[C]. In EMNLP. 2015:879-888.

Collobert R, Weston J, Karlen M, et al. Natural Language Processing (Almost) from Scratch[J]. Journal of Machine Learning Research, 2011, 12(1):2493-2537.

Goller C, Kuchler A. Learning task-dependent distributed representations by backpropagation through structure[C]. IEEE International Conference on Neural Networks. IEEE, 2002:347-352 vol.1.

Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735.

Gers F A, Schmidhuber J, Cummins F. Learning to forget: continual prediction with LSTM[J]. Neural Computation, 2000, 12(10):2451.

Cho K, Merrienboer B V, Bahdanau D, et al. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches[J]. Computer Science, 2014.

Peng N, Dredze M. Named Entity Recognition for Chinese Social Media with Jointly Trained Embeddings[C]. In EMNLP. 2015:548-554.

Sang E F T K, Veenstra J. Representing Text Chunks[J]. Computer Science, 1999:173--179.

Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult[M]. IEEE Press, 1994.

Pascanu R, Mikolov T, Bengio Y. On the difficulty of training Recurrent Neural Networks[J]. Computer Science, 2012, 52(3):III-1310.

Gers F A, Schraudolph N N. Learning precise timing with lstm recurrent networks[M]. JMLR.org, 2003.

Dyer C, Ballesteros M, Ling W, et al. Transition-Based Dependency Parsing with Stack Long Short-Term Memory[J]. Computer Science, 2015, 37(2):321–332.

Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging[J]. Computer Science, 2015.

Zhao, H., and Kit, C. 2008. Unsupervised segmentation helps supervised learning of character tagging for word segmentation and named entity recognition[C]. In IJCNLP, 106–111.

SlugNERDS: A Named Entity Recognition Tool for Open Domain dialogue Systems

Kevin K. Bowden, Jiaqi Wu, Shereen Oraby, Amita Misra, and Marilyn Walker
Natural Language and dialogue Systems Lab
University of California, Santa Cruz
{kkbowden, jwu64, soraby, amisra2, mawalker}@ucsc.edu

Abstract

In dialogue systems, the tasks of named entity recognition (NER) and named entity linking (NEL) are vital preprocessing steps for understanding user intent, especially in open domain interaction where we cannot rely on domain-specific inference. UCSC's effort as one of the funded teams in the 2017 Amazon Alexa Prize Contest has yielded Slugbot, an open domain social bot, aimed at casual conversation. We discovered several challenges specifically associated with both NER and NEL when building Slugbot, such as that the NE labels are too coarse-grained or the entity types are not linked to a useful ontology. Moreover, we have discovered that traditional approaches do not perform well in our context: even systems designed to operate on tweets or other social media data do not work well in dialogue systems. In this paper, we introduce Slugbot's Named Entity Recognition for dialogue Systems (SlugNERDS), a NER and NEL tool which is optimized to address these issues. We describe two new resources that we are building as part of this work: SlugEntityDB and SchemaActuator. We believe these resources will be useful for the research community.

Keywords: dialogue systems, entity recognition, knowledge graphs

1. Introduction

When building dialogue systems, reliable named entity recognition (NER) and named entity linking (NEL) are vital to understanding user intent, especially if these dialogue systems are open domain and intended to support conversations on any topic. In designing our open domain social bot, Slugbot(Bowden et al., 2017b), for the 2017 Amazon Alexa Prize Contest(Ram et al., 2017), we discovered several challenges specifically associated with both NER and NEL. This paper discusses these challenges, and shows how we address them with Slugbot's Named Entity Recognition for Dialogue Systems (SlugNERDS), a tool designed for NER and NEL in open domain dialogue. Additionally we present two corpora, SlugEntityDB and SchemaActuator, which are based on over 10,000 real user conversations with the system. We perform an extensive analysis of our system and the corpora to identify important areas of future work.

NER and NEL have been actively researched topics for decades (Finkel and Manning, 2009; Ratinov and Roth, 2009; Ritter et al., 2011; Derczynski et al., 2015; Nitish Gupta and Roth, 2017). However, the resulting entity classification is often coarse and does not encode an ontology. For example, Stanford NER features only a small number of abstract entity types such as PERSON, LOCATION, ORGANIZATION, and MISC (Manning et al., 2014; Finkel and Manning, 2009); these categories don't provide enough information for dialogue interpretation and generalization. Although other resources such as that from Ratinov and Roth (2009) utilize additional external knowledge by extracting 30 gazetteers from both the web and Wikipedia, the entity types are still not as varied as we need, and the framework lacks a clear ontology. Furthermore, the alignment of classes between systems can be inconsistent as there is no universally shared taxonomy between them and the various data streams necessary to support open domain conversation(Bowden et al., 2017a). While Ling and Weld (2012) attempt to address this by using 112 fine-grained entity types consistent with Freebase (Bollacker et al., 2008), Freebase is no longer maintained and recent inspection has shown it to be significantly incomplete (Dong et al., 2014). While the accuracy of these state of the art NER systems can be quite high, ranging between 80-90% on long text, on short informal text, such as tweets, accuracies drop to between 30-50% (Derczynski et al., 2015). Tweets are much more representative of the data we see from users in an interaction with a social bot than newswire data. Specifically, utterances tend to be short, and due to the open domain setting, relevant context is not guaranteed.

However, unlike tweets, a dialogue system must maintain a discourse model which can sustain multi-turn dialogue for the duration of the user interaction. In fact, it is clear that a dialogue system, which necessarily must operate in real time, has inherent challenges not present in other settings. Real time systems must be optimized such that they function without significant response delays between turns; such delays can be introduced by approaches that rely on running a machine learning classifier in real time. Additionally, new named entities are very commonly discussed in a social setting and need to be recognized as trends and current popular topics change: this requires systems to be constantly (nightly) retrained on newly annotated data. Moreover, there must be significant noise tolerance - in a social setting it is often the case that users will speak more informally; the spoken domain can be even more challenging as an entity resolution tool must also account for automatic speech recognition (ASR) misinterpretations.

2. Overview of Tools and Corpora

To address these challenges we present Slugbot's Named Entity Recognition for dialogue Systems (SlugNERDS), an NER and NEL tool which leverages the Google Knowledge Graph API in conjunction with the Schema.org taxonomy to identify known entities. The tool is optimized with respect to noisy open domain conversation and is able to per-

1	**User:** Let's talk about comic books. Who is your favorite character?	In open domain dialogue, the user can engage us on any topic.
2	**System:** I like [Moon Knight]$_{Thing}$ because [he]$_{Thing}$ can see the future sometimes.	Fictional entities, such as obscure comic book characters, are classified as *Thing*. Using our web based NEL, we could also summarized he is a *Fictional Super Hero*.
3	**User:** I guess that is a cool power, when did [he]$_{Thing}$ first appear?	SlugNERDS performs anaphoric resolution to perform discourse level entity linking, in this case associating *he* with *Moon Knight*.
4	**System:** I think [Marvel Comics]$_{Corporation}$ debuted [Moon Knight]$_{Thing}$ in [werewolf by night]$_{Thing}$ number 32	In SlugNERDS, correctly identifying *Moon Knight* allows us to perform web level entity linking - we have a link to a relevant Wikipedia article where we find more information about the character, specifically the series he first appeared in, *Werewolf by Night*.
5	**User:** you know I kind of like scary space **movies**. I saw [alien covenant]$_{Movie}$ the other day	The *movies* keyword matches directly to an entity type, thus increasing the systems probability of picking the correct movie entity.
6	**System:** I thought [Alien Covenant]$_{Movie}$ was scary and action packed.	Often times users are most interested in discussing trending entities, *Alien Covenant* had been released in the theaters recently.
7	**User:** That's cool, have you **seen** the new [Star Wars]$_{MovieSeries}$ **movie**?	Both the *movie* keyword and the *seen* action are useful to detect all *Movie* type entities, such as *MovieSeries*, *MovieClip*, and *Movie*.
8	**System:** I haven't had a chance yet. Which is your favorite in the series?	Differentiating between *Movie* and *MovieSeries* allows us to curate better follow up question.
9	**User:** I think my favorite [Star Wars]$_{MovieSeries}$ **movie** is [revenge of the sith]$_{Movie}$	We just need the partial title, *revenge of the sith*, to correctly identify the entity *Star Wars: Episode III - Revenge of the Sith*.
10	**System:** Who is your favorite actor in [it]$_{Movie}$?	We know that *Movies* have *actors*.
11	**User:** I'm **watching** the [Lakers]$_{SportsTeam}$ play right now, which is funny because I hate [Los Angeles]$_{City}$, even though I **live** in [California]$_{State}$ - best **state** in the [United States]$_{Country}$.	We note the various levels of granularity, instead of just a location tag we can detect each destination with more precision.
12	**System:** Whose your favorite player on the [Lakers]$_{SportsTeam}$?	Using the Schema.org attributes, we know that *SportsTeams* have *athletes*.
13	**User:** [albert einstein]$_{Person}$ is very smart [he]$_{Person}$ is my favorite scientist. I wonder if [he]$_{Person}$ would have liked **reading** [the hunger games]$_{BookSeries}$ [mockingjay]$_{Book}$.	Note that users can abruptly shift to a different context at any time. In this example, the *mockingjay* movies scored higher than the books initially, however once we take into account the expectation of Book type entities via the *reading* action, the correct entities are selected.

Table 1: Sample dialogue with entities highlighted in addition to SchemaActuators annotations.

form both discourse and web based entity linking. Table 1 represents an annotated conversation based on real interactions with Slugbot. While one could enhance the result further by utilizing the state of the dialogue system to set system expectations, we are interested in evaluating our tool without making any assumptions of the system using it.

To supplement SlugNERDS we also present SlugEntityDB, an annotated corpus which can be used to evaluate our system. This represents to our knowledge the first Schema.org entity type annotated corpus for this task. The SlugEntityDB contains 2100 samples, 500 taken directly from real user data collected by SlugBot(Bowden et al., 2017b) during the inaugural Amazon Alexa Prize contest, 1600 synthesized such that we can easily verify the richness of the corpus. Since we are operating in the open domain, it is also very likely that adding synthesized data will result in entities which have never been seen by the system previously. In this dataset the utterances are formatted similar to the input which a spoken dialogue system would receive from a state of the art ASR system. Each utterance is annotated in tuples which includes the direct strong overlap, entity types, and full entity name as per the Google Knowledge Graph. Table 2 includes a sample of this dataset.

Additionally, we provide the SchemaActuators corpus, a partially hand annotated probabilistic mapping between actions/specific keywords/phrases and entity classifications (such as indicating *watch* or *seen* are related to *Movie* entities). An example of these mappings can be seen in Table 1,

Utterance	Annotation
the lord of the rings was my childhood	(lord of the rings, MovieSeries Thing, The Lord of the Rings)
my favorite star wars movie is probably revenge of the sith	(revenge of the sith, Movie Thing, Star Wars: Episode III Revenge of the Sith); (star wars, MovieSeries Thing, Star Wars)
i want to visit black mountain	(black mountain, TouristAttraction Mountain Place Thing, Black Mountain)

Table 2: Samples from the SlugEntityDB.

where appropriate entries in the corpus are colored to match the associated entity. Currently the verbs used in this corpus have been hand annotated. These seed verbs are then expanded using synonym relations from Wordnet(Miller, 1995) and other lexical resources. Using these verbs in addition to prepositional phrases we automatically generate a list of short phrases associated with specific entities, such as *arrive at* for the *City* entity. We use a similar process of automatic expansion on the entity type to generate a list of candidate keywords which can potentially indicate an entity, such as associating the *flick* and *film* keyword with the *Movie* entity. It is our belief that this corpus will lead to improved results as it allows us to better adjust our system's expectations.

3. Tools and Methodology

In this section, we describe the tools and methodology we use to build SlugNERDS.

3.1. Google Knowledge Graph

The SlugNERDS tool primarily utilizes the Google Knowledge Graph API [1] to identify known entities. A query to the Google Knowledge Graph API returns a list of the N most probable entities which are sorted based on an associated base score. This score is provided by the API and is assumed to be a combination of contextual overlap and entity popularity. Each entity has substantial meta data including the full name of the entity, Schema.org type classification, and both a brief and long description with an associated Wikipedia article. Having automatic access to an associated Wikipedia article allows us to reliably solve the task of web based NEL contingent on successfully performing NER. An example query result can be seen in Figure 1.
Utilizing this API to identify known entities is ideal as it isn't domain specific, and requires no training from users of the tool. While directly querying the API in real time may sound expensive, this tool has been successfully deployed in our real time conversational agent, Slugbot, without significantly inhibiting the user experience.

[1] https://www.google.com/intl/bn/insidesearch/features/search/knowledge.html


```
{
    '@context': {
    ...
    },
    '@type': 'ItemList',
    'itemListElement': [
    {
    '@type': 'EntitySearchResult',
    'resultScore': 801.711243,
    'result': {
        'description': '2005 film',
        'url': ...,
        '@id': 'kg:/m/0fdv3',
        '@type': [
            'Movie',
            'Thing'
        ],
        'detailedDescription': {
            'articleBody': ...,
            'license': ...,
            'url': ...
        },
        'name': 'Star Wars: Episode III – Revenge of the Sith'
    }
    },
    ...
    ]
}
```


Figure 1: Google Knowledge Graph Search API Result For the Query *revenge of the sith*

3.2. Schema.org

The Google Knowledge Graph API classifies entities using the Schema.org [2] entity ontology. Schema.org is an effort to create a richer web infrastructure by proposing common MicroData for entities within a website. Millions of websites which contain rich structured data across an array of subjects - such as IMDb [3], BestBuy [4], BarnesNNobles [5], and Yelp [6] - have already adopted this MicroData in some form.
This not only enforces consistency across a multitude of data streams, but allows us to connect common entities to their related attribute types (such as *SportsTeam* $\rightarrow$ *athlete* $\rightarrow$ *Person* $\rightarrow$ *birthDate*), allowing the system to retrieve a large set of possible next topics, related facts, and associated entities. We can further expand on potential topics by utilizing the schema ontology to access properties of the entities higher up in the hierarchy.

3.3. SlugNERDS Pipeline

Figure 2 represents the general SlugNERDS pipeline. Our *Name Entity Recognition* consists of two standard modules, *Entity Segmentation* and *Entity Classification* (Ritter et al., 2011; Collins and Singer, 1999; Downey et al., 2007). Subsequently we perform *Entity Linking* on the recognized entity. We will examine this process with the following example utterance: *"I think my favorite star wars is revenge of the sith"*. Please note that punctuation is not included in the user utterance.

3.3.1. Entity Segmentation

In order to refine our list of candidate strings to query we must break our text into reasonable chunks. We utilize a

[2] https://www.schema.org
[3] https://www.imdb.com
[4] https://www.bestbuy.com
[5] https://www.barnesandnoble.com
[6] https://www.yelp.com

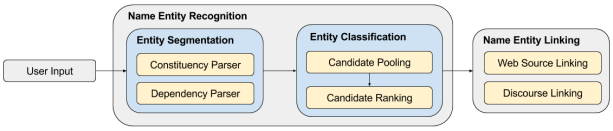


Figure 2: The SlugNERDS pipeline.

two pass approach to maximize our recall with a decent number of candidates. We maximize recall because a properly structured dialogue system will be able to pick the contextually relevant entity for follow-up questions and ignore extraneous entities which may have been misclassified. Furthermore, through empirical evaluation we have concluded that Slugbot missing an entity can be more detrimental to a conversation than over-classifying entities.

First, we construct a constituency tree using Stanford CoreNLP (Manning et al., 2014) and build our candidate pool by collapsing each of the noun phrases, verb phrases, and sentences in the tree. A sample constituency tree can be seen in Figure 3. Additionally, we collapse sequential noun clusters from the dependency parse which have not yet been associated with an entity to create a secondary pool of candidates, so as to include more candidate strings that are ignored by shallow parsing (Ritter et al., 2011). We exclude single pronouns such as *I* and *me* unless they seem extremely contextually relevant, such as in the case of *It*[7], when discussing movies with the user.

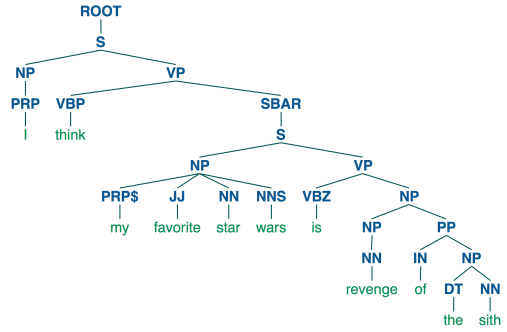


Figure 3: Constituency Tree

3.3.2. Entity Classification

Once we collect the candidate phrases in the Segmentation phase, we then query each of these phrases using the Google Knowledge Graph API and collect the top 6 relevant entities in *Candidate Pooling*. Here 6 is an empirically derived value which represents a good range of possible candidates without including excessive candidates. Sometimes it is possible for the entity candidate returned by the query to have the same exact title with different entity types. For example, the phrase *revenge of the sith* returns 5 candidates with the title *Star Wars: Episode III - Revenge Of The Sith* each with a different entity type (*Movie*, *Video Game*, *Book*, *MusicAlbum*, and *BookSeries*). The base score can vary significantly between these versions, for example the base score for the *Movie* entity is 795.59, while it is 138.05 for the *BookSeries* entity - this is assumed to be due to the popularity of the movie vs. the book series.

[7] *It* is currently a very popular horror movie.

User Utterance: I think my favorite *Star Wars* movie is *revenge of the sith*.
Candidate: Star Wars: Episode III - Revenge of the Sith $['Movie','Thing']$
Candidate Initial Vector: $[1,1,1,1,1,1,1,1]$
Candidate Overlap: $[1,1,0,0,0.11,.75,.75,1]$

Table 3: Example word vector translation.

Since the entities returned by the Google Knowledge Graph may not be an exact match to our query, it gives us more flexibility, while introducing some noise. Furthermore, if we are expecting a user to talk about certain entity types according to the context, as discussed when introducing the SchemaActuators corpus in Section 2., we place increased value on certain entities while penalizing others. Thus, we perform our scoring algorithm to maximize our performance in the *Candidate Ranking* phase.

First, we create a word overlap vector indicating the total overlap. Because we expect noise in the user's utterance, we allow for unexpected words to be inserted in the middle of the phrase realizing the entity with a distance based penalty. Table 3 gives an example of the word vector. The word vector is initialized as 1 for each candidate token, for instance, *Star Wars: Episode III - Revenge of the Sith* has 8 tokens, thus the initial vector is [1,1,1,1,1,1,1,1]. Then we compare the candidate tokens with the user utterance. The element of the vector will be penalized if tokens are missing or extra tokens are detected. For example, the element 3 and element 4 are set to zero because there is two missing words (*Episode III*), resulting in the 5th element of the vector receiving a penalty of $\frac{1}{3}$, which is the distance between the matching tokens. There are also two extra words (*movie is*) in the user utterance, thus an additional penalty of $\frac{1}{3}$ is applied to the 5th element, which yields a 0.11 score. We also note the imposed penalty on stop words by multiplying the element with 0.75 - this helps to increase our precision as we try to prevent correctly positioned stop words from accidentally forming an entity. The 6th and 7th elements (*of the*) are the stop word examples. Equation 1 demonstrates how we use this word vector to alter the score of an entity.

$$overlap_score = sum(word_vector) * \left(\frac{1}{word_vector.count(0)+1}\right) * basescore \quad (1)$$

Secondly, we account for any entities we are expecting based on early inspection of the utterance's context:

$$can_score = overlap_score * (num_matches + 1) \quad (2)$$

Once the entities are all scored, we rerank our list and consider only the top ranked entity for each node, while also pruning away nodes whose top scoring entity was less than a certain threshold (empirically driven). Finally, we merge overlapping nodes who have candidates. For example, if both *revenge of the sith* and a child node *revenge* have the

same entity as their top scoring entity, we will merge these two nodes or remove lower ranking conflicts.

In our last stage, we sync the query/candidate to our internal discourse state representation. In our example, two entities are extracted, mapping *star wars* to entity type *MovieSeries*, and *revenge of the sith* to *Star Wars: Episode III - Revenge Of The Sith* with entity type *Movie*.

3.3.3. Entity Linking

Named Entity Linking is primarily encapsulated in two phases, *Web Source Linking* and *Discourse Linking*. With Web Source Linking we are interested in linking a known entity to existing resources on the web while discourse linking is focused on linking each mention of the entity within the input to the same discourse entity in our internal representation (Brennan et al., 1987; Walker et al., 1997).

As mentioned in Section 3.1., the Google Knowledge Graph query returns a Wikipedia article associated with the entity. We can further increase our web based linking by utilizing the fact that a large number of popular websites use the Schema.org MicroData, allowing us to easily target relevant sources for information extraction. Finally, through empirical examination, we note that pairing the entity type with the precise entity name as provided in the query will allow for easy subsequent queries to large databases such as YAGO (Rebele et al., 2016) or DBpedia (Auer et al., 2007).

Finally, our tool uses an augmented version of the Stanford Coreference Annotator (Manning et al., 2014) to perform Discourse Linking.

4. Evaluation

4.1. NER Results

Our SlugNERDS tool was originally developed and utilized in the 2017 Alexa Prize Competition for SlugBot, which scored in the top 25% of competing social bots. For detailed system evaluation, we present a set of experiments to evaluate our SlugNERDS tool and SchemaActuator corpus, independently from SlugBot. We evaluate SlugNERDS using (1) the base scores using only text segmentation with the Google Knowledge Graph, (2) augmenting the scores using our scoring algorithm, (3) using the SchemaActuator actions to increase context, (4) using the SchemaActuator keywords to increase context, (5) an ensemble approach. Moreover we test these configurations with two different scoring thresholds (the minimum score required for an entity be accepted as the correct class). The two thresholds we will test are (1) 150, an empirically driven value deemed to be a reasonable threshold during the development of SlugBot and (2) 0, no threshold at all.

Table 4 presents the results of our NER experiments using the SchemaEntityDB. Since Stanford NER is still commonly used in many state of the art open domain conversational systems, we use it as our baseline system. Our other experiments include using just our text segment method, then adding SlugNERDS ranking, iteratively adding in verbs and nouns from our SchemaActuators corpus, and finally showing our ensemble method results (which merges all resources). Since Stanford provides coarse grain entity type such as PERSON, LOCATION, ORGANIZATION, and MISC, we map the specific gold standard label to these four types for the evaluation.

We use accuracy, macro-f1, micro-f1, and weighted-f1, to have a better understanding of our system's performance. High accuracy entity detection will ensure a more satisfiable conversation, and prevent us from missing the topics of the conversation. The macro-f1 treats all the classes evenly, the micro-f1 accounts for label imbalance, and the weighted-f1 is a weighted macro-f1. Since detecting infrequent entities is also important for us, we evaluate the macro-f1, the micro-f1 and the weighted-f1 for different interests. Table 4 shows that Stanford yields a macro-F1 score as 0.079. As predicted, we see from these results that Stanford NER is not a suitable system to use when detecting entities in open domain discourse. Our text segmentation model has an accuracy of 0.751, macro-F1 of 0.785, and weighted-F1 of 0.831. Our SlugNERDS models has a better accuracy 0.777 and a slightly worse macro-F1 of 0.746, but a better weighted-F1 0.85. After utilizing the SchemaActuator corpus, the macro-F1 are improved slightly, though the weighted-F1 is almost the same, which might due to the reason that the extra Act_Verbs and Act_Nouns are able to detect the edge cases which are infrequent entities. We believe that when integrated with a real dialogue system, the increase in contextual knowledge from our dialogue manager will yield further increase performance.

In our original experiment we see that a low threshold doesn't need contextual information to classify entities. However after examining the results it's clear that while more correct entities are being classified, there is also increased levels of over-classification which can be detrimental to the system. Therefore we aim to increase our accuracy with an with an empirically derived threshold of 150. In this experiment our model continued to outperformed the stanford baseline and the text segmentation results were the worst configuration for SlugNERDS. We can also see by comparing Table 4 and Table 5 that as we increase this threshold, we increase the impact of adding contextual information to SlugNERDS. This implies the importance of encoding contextually relevant data in our model while also increasing our tolerance of noise. In future work we aim at further analyzing our distribution of annotated entities to pick an optimized threshold value.

4.2. Distribution of Detected Entities

Figure 4 shows the distribution of Stanford Entities that are detected. The NUMBER, DATE, TIME, SET, ORDINAL, DURATION, MONEY, and PERCENT are not very useful to our system as we are interested primarily in known entities. Therefore we are most interested in the PERSON, ORGANIZATION, LOCATION, and MISC types - however, all of these entity types detected much less frequently than they appear in our annotated data, as can be seen in Figure 6. Figure 5 shows the top 15 most frequent entities that caught our SlugNERDS Models. We can see that our SlugNERDS model is able to successfully detect large amounts of various conversational entities such as Movie, Book, and MusicRecording, while also maintaining a very similar distribution as seen in our annotated data.

	Accuracy	macro-F1	micro-F1	weighted-F1
Stanford Baseline	.029	.079	.029	.052
Text Segmentation Only	.751	.785	.722	.831
SlugNERDS Scoring	.777	.746	.749	.850
SlugNERDS Scoring + Act_Verbs	.770	.752	.747	.849
SlugNERDS Scoring + Act_Nouns	.773	.762	.745	.848
Ensemble	.763	.747	.742	.845

Table 4: NER Results for the Threshold 0 experiment.

	Accuracy	macro-F1	micro-F1	weighted-F1
Stanford Baseline	.029	.079	.029	.052
Text Segmentation Only	.521	.656	.511	.663
SlugNERDS Scoring	.553	.624	.546	.692
SlugNERDS Scoring + Act_Verbs	.604	.655	.598	.735
SlugNERDS Scoring + Act_Nouns	.579	.619	.571	.714
Ensemble	.592	.634	.586	.726

Table 5: NER Results for the Threshold 150 experiment.

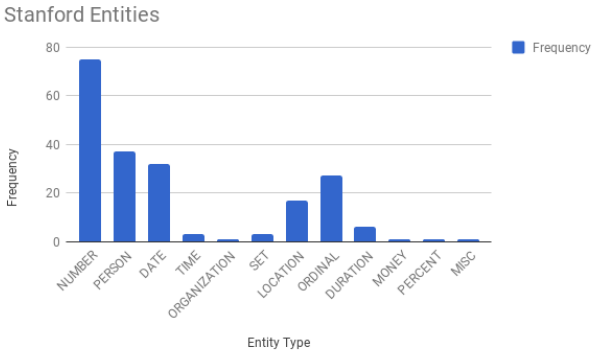


Figure 4: Distribution entity types detected by the Stanford NER.

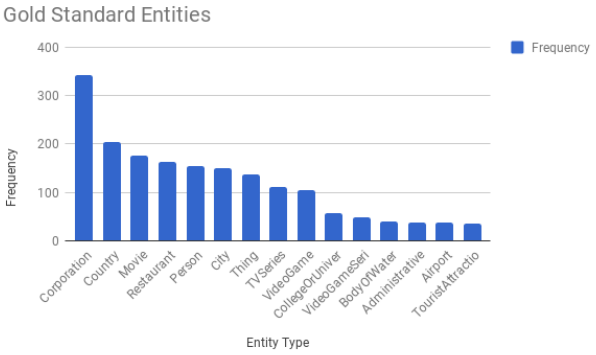


Figure 6: Distribution of top 15 entity types in the annotated data.

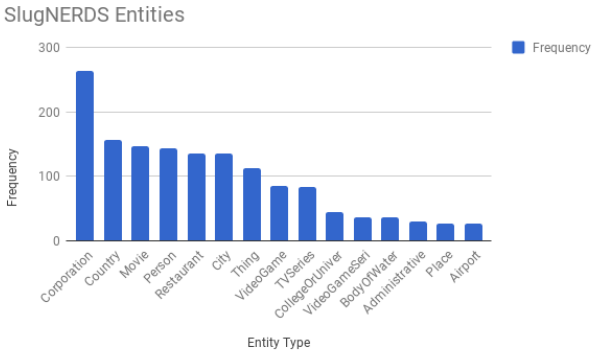


Figure 5: Distribution of top 15 entity types that our model detected.

4.3. Error Analysis

Here we will discuss different errors we noticed while analyzing the results of the experiments described above. Primarily, we see three different classes of errors; insufficient contextual information, natural language understanding deficiencies, and the difficulty of encapsulating every possible entity within the Google Knowledge Graph.

Table 6 demonstrates several errors which are the result of insufficient contextual information. While this evaluation was meant to analyze SlugNERDS as a stand-alone tool, it is clear that these errors can be resolved by enhancing the contextual knowledge we pass to our tool. For example, while there are no lexemes which indicate *lord of the rings* in Sample 6a is referring to the MovieSeries, the state of our dialogue system could inform us that we are discussing movies, rather than books. As in Sample 6b, we see that it common in colloquial speech to refer to an element of a series by it's common root word. For example *halo* in the expression *let's play halo* is valid for *halo combat evolved*, *halo 2*, *halo wars*, and various other titles in the series. Disambiguating this is not a trivial problem and requires a significantly more rich state than the previous example. We will note here that it is possible to also leverage the granularity of our Schema.org entity types to recognize we are talking about a VideoGameSeries and clarify the specific VideoGame with the user. Finally, Sample 6c

demonstrates how a lack of user meta-data, in this case location and frequency, results in incorrect classifications. A dialogue system can represent this contextual data by referencing a user model which may indicate that since the user lives in California, they are likely referring to San Jose, California, rather than San Jose, Costa Rica.

Table 7 demonstrates two areas in which adding an additional layer of NLU would yield increased performance. Specifically we notice in Sample 7b that abbreviations cause difficulty in classifying entities. While ucsd will infact return the correct entity as a potentially candidate from the Google Knowledge Graph, there is no lexical overlap, resulting in a false classification. By adding an additional layer of NLU which is able to expand abbreviated entities we would see an increase in performance. As described previously, the SlugEntityDB annotated corpora was designed assuming the input provided is from a spoken dialogue system - meaning it will suffer from the limitations of state of the art automatic speech recognition. More specifically, our utterances have no punctuation, capitalization, or non-alpha-numeric symbols. While we have already resolved this partially such that utterances like *x man* will correctly map to *X-man*, the lack of punctuation and capitalization can lead to inaccurate results from our parser - this can directly alter the queries which are sent to the Google Knowledge Graph. This can be seen in Sample 7a, where the best possible query for the original utterance was *love gordon ramsey* which does not yield the correct entity. After manually capitalizing the entity name, our best possibly query became *Gordon Ramsey*, which resulted in the correct entity.

Finally, as there exists an infinite amount of entities, it is reasonable to surmise that the Google Knowledge Graph, while quite robust, is incomplete. For example, querying either *xbox 360* or *windows* both return the *Microsoft* entity has the best possible candidate. While not necessarily common, this can be a difficult problem to deal with, especially when an entire category of entities, such as operating systems or video game consoles, are missing from the Knowledge Graph.

5. Conclusion and Future Work

In this paper we have presented SlugNERDS, an NER and NEL tool which is optimized with the respect to the challenges that are specific to open domain conversation. We have also presented two relevant corpora, one being the first dataset of its kind to be annotated with Schema.org named entity types in addition to the SchemaActuator corpus - a mapping of actions and keywords to their respective Schema.org entity types.

To our knowledge our system is the only one to utilize the Schema.org entity types for entity classification, but we plan to compare our system more extensively to other existing NER systems in future work, which may allow us to improve our system. One is T-NER System (Ritter et al., 2011) [8], which is optimized for NER in Tweets, as discussed in Section 1.. We are also interested in comparing against a state of the art neural NEL model (Nitish Gupta and Roth, 2017) [9]. This model more proactively tackles the open domain problem, and can include updated knowledge of existing entities without retraining. There are clearly many areas in which we can further evaluate and improve the performance of SlugNERDS. Specifically, we plan to evaluate the performance of SlugNERDS with an emphasis on the impact of contextual information provided as a result of our dialogue system's state tracking capabilities. We also plan to improve our natural language understanding pipeline by investigating methodologies for improving in-

Utterance	Correct Entity	Predicted Entity
the lord of the rings was my childhood	MovieSeries	BookSeries

(a) Without sufficient contextual knowledge, it is difficult to differentiate between entities with the exact same title.

Utterance	Correct Entity	Predicted Entity
halo has been dead for a while now	VideoGame, Halo: Combat Evolved	VideoGameSeries, Halo)

(b) Entities which are nested within a series can also be hard to detect, such as here when the user is talking about the first Halo game vs. the Halo series.

Utterance	Correct Entity	Predicted Entity
sacramento airport was pretty busy	Sacramento International Airport	Sacramento Airport
i did not know san jose is a capital	San Jose, California	San Jose

(c) Without contextual information about the user, it's difficult to differentiate between San Jose refers to San Jose California vs. San Jose Costa Rica. Similarly, "common sense" indicates that "Sacramento Airport" refers to Sacramento International Airport, rather than the much smaller Sacramento Airport.

Table 6: Common errors stemming from insufficient contextual knowledge.

Utterance	Correct Entity	Predicted Entity
i love gordon ramsay	Person	None
i love Gordon Ramsay	Person	Person

(a) A lack of capitalization and punctuation leads to parser errors, resulting in missed queries.

Utterance	Correct Entity	Predicted Entity
ucsd is number 23	University of California, San Diego	None

(b) Mismatch between the string in the utterance vs the actual entity name, as in the case with an abbreviation, causes classification to fail.

Table 7: Common errors stemming from gaps in our natural language understanding pipeline.

[8] https://github.com/aritter/twitter_nlp

[9] https://nitishgupta.github.io/neural-el/

put text quality, such as automatic capitalization and punctuation insertion. Finally, we are currently investigating how reinforcement learning can be applied to our existed conversational data to increase our models contextual insight.

6. References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg. Springer-Verlag.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250, New York, NY, USA. ACM.

Bowden, K. K., Oraby, S., Wu, J., Misra, A., and Walker, M. (2017a). Combining search with structured data to create a more engaging user experience in open domain dialogue. *SCAI 2017*.

Bowden, K. K., Wu, J., Oraby, S., Misra, A., and Walker, M. (2017b). Slugbot: An application of a novel and scalable open domain socialbot framework. In *Alexa Prize Proceedings*.

Brennan, S. E., Friedman, M. W., and Pollard, C. J. (1987). A centering approach to pronouns. In *Proc. 25th Annual Meeting of the ACL, Stanford*, pages 155–162.

Collins, M. and Singer, Y. (1999). Unsupervised models for named entity classification. In *In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 100–110.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., and Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing and Management*, 51(2):32 – 49.

Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 601–610, New York, NY, USA. ACM.

Downey, D., Broadhead, M., and Etzioni, O. (2007). Locating complex named entities in web text. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 2733–2739.

Finkel, J. R. and Manning, C. D. (2009). Joint parsing and named entity recognition. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 326–334, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ling, X. and Weld, D. S. (2012). Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 94–100. AAAI Press.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Nitish Gupta, S. S. and Roth, D. (2017). Entity linking via joint encoding of types, descriptions, and context. In *EMNLP*.

Ram, A., Prasad, R., Khatri, C., Venkatesh, A., Gabriel, R., Liu, Q., Nunn, J., Hedayatnia, B., Cheng, M., Nagar, A., King, E., Bland, K., Wartick, A., Pan, Y., Song, H., Jayadevan, S., Hwang, G., and Pettigrue, A. (2017). Conversational ai: The science behind the alexa prize. In *Alexa Prize Proceedings*.

Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, pages 147–155, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rebele, T., Suchanek, F., Hoffart, J., Biega, J., Kuzey, E., and Weikum, G., (2016). *YAGO: A Multilingual Knowledge Base from Wikipedia, Wordnet, and Geonames*, pages 177–185. Springer International Publishing, Cham.

Ritter, A., Clark, S., Mausam, and Etzioni, O. (2011). Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marilyn A. Walker, et al., editors. (1997). *Centering in Discourse*. Oxford University Press.

Transfer Learning for Named-Entity Recognition with Neural Networks

Ji Young Lee[*1], **Franck Dernoncourt**[*1,2], **Peter Szolovits**[1]
[1]MIT, [2]Adobe Research
jjylee@mit.edu, dernonco@adobe.com, psz@mit.edu

Abstract

Recent approaches based on artificial neural networks (ANNs) have shown promising results for named-entity recognition (NER). In order to achieve high performances, ANNs need to be trained on a large labeled dataset. However, labels might be difficult to obtain for the dataset on which the user wants to perform NER: label scarcity is particularly pronounced for patient note de-identification, which is an instance of NER. In this work, we analyze to what extent transfer learning may address this issue. In particular, we demonstrate that transferring an ANN model trained on a large labeled dataset to another dataset with a limited number of labels improves upon the state-of-the-art results on two different datasets for patient note de-identification.

Keywords: named-entity recognition, neural networks, transfer learning

1. Introduction

Electronic health records (EHRs) have been widely adopted in some countries such as the United States and represent gold mines of information for medical research. The majority of EHR data exist in unstructured form such as patient notes (Murdoch and Detsky, 2013). Applying natural language processing on the unstructured data in conjunction with analyzing the other EHR data can lead to a better understanding of health and diseases (Liao et al., 2015), and a more accurate phenotyping of patients to compare tests and treatments (Ananthakrishnan et al., 2013; Pivovarov and Elhadad, 2015; Halpern et al., 2016).

However, before patient notes can be shared with medical investigators, some types of information, referred to as protected health information (PHI), must be removed in order to preserve patient confidentiality. In the United States, the Health Insurance Portability and Accountability Act (HIPAA) (Office for Civil Rights, 2002) defines 18 different types of PHI, ranging from patient names and ID numbers to addresses and phone numbers. The task of removing PHI from a patient note is referred to as *de-identification*. The essence of de-identification is recognizing PHI in patient notes, which is a form of named-entity recognition (NER).

Existing de-identification systems are often rule-based approaches or feature-based machine learning approaches. However, these techniques require additional lead time for developing and fine-tuning the rules or features specific to each new dataset. Meanwhile, recent work using ANNs have yielded state-of-the-art performances without using any manual features (Dernoncourt et al., 2016). Compared to the previous systems, ANNs have a competitive advantage that the model can be fine-tuned on a new dataset without the overhead of manual feature development, as long as some labels for the dataset are available.

However, it may still be inefficient to mass deploy ANN-based de-identification system in practical settings, since creating annotations for patient notes is especially difficult. This is due to the fact that only a restricted set of individuals is authorized to access original patient notes; the annotation task cannot be crowd-sourced, making it slow and expensive to obtain a large annotated corpus. Medical professionals are therefore wary to explore patient notes because of this de-identification barrier, which considerably hampers medical research.

In this paper, we analyze to what extent transfer learning may improve de-identification performances on datasets with a limited number of labels. By training an ANN model on a large dataset (MIMIC) and transferring it to smaller datasets (i2b2 2014 and i2b2 2016), we demonstrate that transfer learning allows to outperform the state-of-the-art results.

2. Related Work

Transfer learning has been studied for a long time. There is no standard definition of transfer learning in the literature (Li, 2012). We follow the definition from (Pan and Yang, 2010): transfer learning aims at performing a task on a target dataset using some knowledge learned from a source dataset. The idea has been applied to many fields such as speech recognition (Wang and Zheng, 2015) and finance (Stamate et al., 2015).

The successes of ANNs for many applications over the last few years have escalated the interest in studying transfer learning for ANNs. In particular, much work has been done for computer vision (Yosinski et al., 2014; Oquab et al., 2014; Zeiler and Fergus, 2014). In these studies, some of the parameters learned on the source dataset are used to initialize the corresponding parameters of the ANNs for the target dataset.

Fewer studies have been performed on transfer learning for ANN-based models in the field of natural language processing. For example, Mou et al. (2016) focused on transfer learning with convolutional neural networks for sentence classification. To the best of our knowledge, no study has analyzed transfer learning for ANN-based models in the context of NER.

[*] These authors contributed equally to this work.

3. Model

The model we use for transfer learning experiments is based on a type of recurrent neural networks called long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997), and utilizes both token embeddings and character embeddings. It comprises six major components:

1. **Token embedding layer** maps each token to a token embedding.
2. **Character embedding layer** maps each character to a character embedding.
3. **Character LSTM layer** takes as input character embeddings and outputs a single vector that summarizes the information from the sequence of characters in the corresponding token.
4. **Token LSTM layer** takes as input a sequence of token vectors, which are formed by concatenating the outputs of the token embedding layer and the character LSTM layer, and outputs a sequence of vectors.
5. **Fully connected layer** takes the output of the token LSTM layer as input, and outputs vectors containing the scores of each label for the corresponding tokens.
6. **Sequence optimization layer** takes the sequence of vectors from the output of the fully connected layer and outputs the most likely sequence of predicted labels, by optimizing the sum of unigram label scores as well as bigram label transition scores.

Figure 1 shows how these six components are interconnected to form the model. All layers are learned jointly using stochastic gradient descent. For regularization, dropout is applied before the token LSTM layer, and early stopping is used on the development set with a patience of 10 epochs.

4. Experiments

4.1. Datasets

We use three de-identification datasets for the transfer learning experiments: MIMIC, i2b2 2014, and i2b2 2016. The MIMIC de-identification dataset was introduced in (Dernoncourt et al., 2016), and is a subset of the MIMIC-III dataset (Johnson et al., 2016; Goldberger et al., 2000; Saeed et al., 2011).The i2b2 2014 and 2016 datasets were released as part of the 2014 i2b2/UTHealth shared task Track 1 (Stubbs et al., 2015) and the 2016 i2b2 CEGS N-GRID shared task, respectively. Table 1 presents the datasets' sizes.

	MIMIC	i2b2 2014	i2b2 2016
Vocabulary size	69,525	46,803	61,503
Number of notes	1,635	1,304	1,000
Number of tokens	2,945,228	984,723	2,689,196
Number of PHI instances	60,725	28,867	41,142
Number of PHI tokens	78,633	41,355	54,420

Table 1: Overview of the MIMIC and i2b2 datasets. PHI stands for protected health information.

4.2. Transfer learning

The goal of transfer learning is to leverage the information present in a source dataset to improve the performance of an algorithm on a target dataset. In our setting, we apply transfer learning by training the parameters of the ANN model on the source dataset (MIMIC), and using the same ANN to retrain on the target dataset (i2b2 2014 or 2016) for fine-tuning. We use MIMIC as the source dataset since it is the dataset with the most labels. We perform two sets of experiments to gain insights on how effective transfer learning is and which parameters of the ANN are the most important to transfer.[1]

Experiment 1 Quantifying the impact of transfer learning for various train set sizes of the target dataset. The primary purpose of this experiment is to assess to what extent transfer learning improves the performances on the target dataset. We experiment with different train set sizes to understand how many labels are needed for the target dataset to achieve reasonable performances with and without transfer learning.

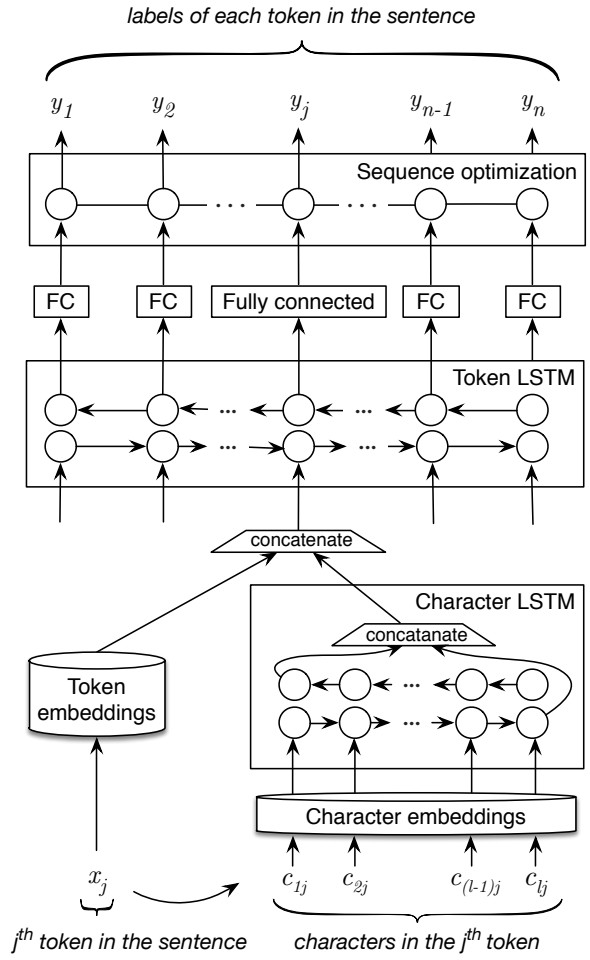


Figure 1: ANN model for NER. For transfer learning experiments, we train the parameters of the model on a source dataset, and transfer all or some of the parameters to initialize the model for training on a target dataset.

[1] Code: https://github.com/Franck-Dernoncourt/NeuroNER

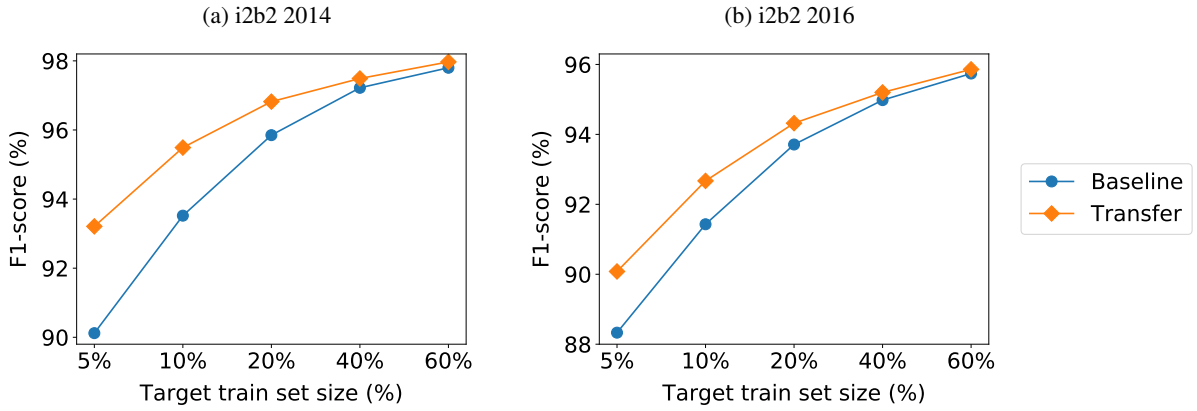


Figure 2: Impact of transfer learning on the F1-scores. Baseline corresponds to training the ANN model only with the target dataset, and transfer learning corresponds to training on the source dataset followed by training on the target dataset. The target train set size is the percentage of train set in the whole dataset, and 60% corresponds to the full official train set.

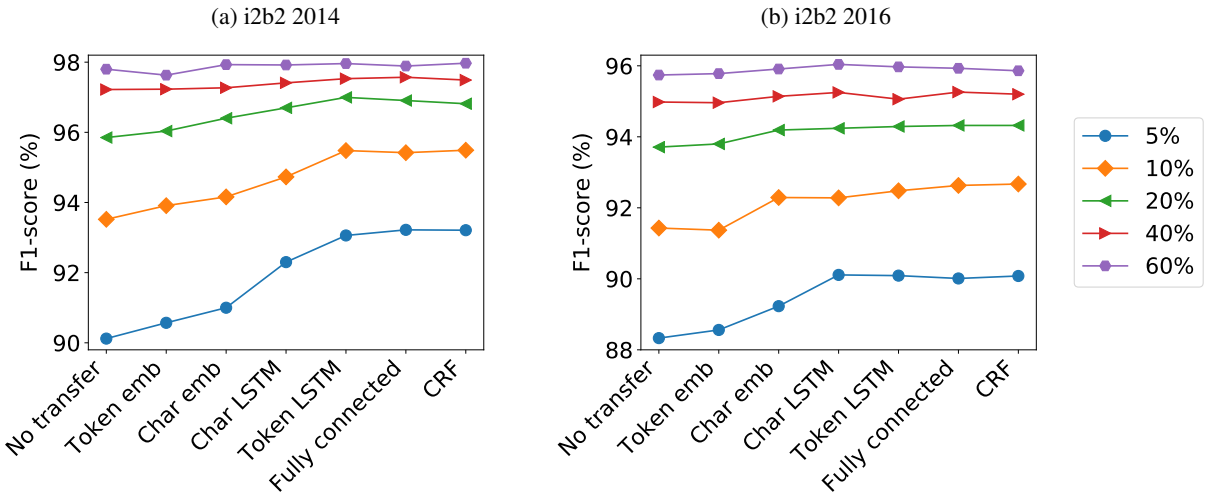


Figure 3: Impact of transferring the parameters up to each layer of the ANN model using various train set sizes on the target dataset: 5%, 10%, 20%, 40%, and 60% (official train set). "CRF" refers to the sequence optimization layer in Figure 1.

Experiment 2 Analyzing the importance of each parameter of the ANN in the transfer learning. Instead of transferring all the parameters, we experiment with transferring different combinations of parameters. The goal is to understand which components of the ANN are the most important to transfer. The lowest layers of the ANN tend to represent task-independent features, whereas the topmost layers are more task-specific. As a result, we try transferring the parameters starting from the bottommost layer up to the topmost layer, adding one layer at a time.

5. Results

Experiment 1 Figure 2 compares the F1-scores of the ANN trained only on the target dataset against the ANN trained on the source dataset followed by the target dataset. Transfer learning improves the F1-scores over training only with the target dataset, though the improvement diminishes as the number of training samples used for the target dataset increases. This implies that the representations learned from the source dataset are efficiently transferred and exploited for the target dataset.

Therefore, when transfer learning is adopted, fewer annotations are needed to achieve the same level of performance as when the source dataset is unused. For example, on the i2b2 2014 dataset, performing transfer learning and using 16% of the i2b2 train set leads to similar performance as not using transfer learning and using 34% of the i2b2 train set. Transfer learning thus allows to cut by half the number of labels needed on the target dataset in this case.

For both the i2b2 2014 and 2016 datasets, the performance gains from transfer learning are greater when the train set size of the target dataset is small. The largest improvement can be observed for i2b2 2014 when using 5% of the dataset as the train set (consisting of around 2k PHI tokens out of 50k tokens), where transfer learning increases the F1-score by around 3.1 percent point, from 90.12 to 93.21. Even when all of the train set is used, the F1-score improves when using transfer learning, albeit by just 0.17 percent point, from 97.80 to 97.97.

Experiment 2 Figure 3 shows the importance of each layer of the ANN in transfer learning. We observe that transferring a few lower layers is almost as efficient as transferring all layers. For i2b2 2014, transferring up to the token LSTM shows great improvements for each layer,

but there is less improvement for each added layer beyond that. For i2b2 2016, larger improvements can be observed up to the character LSTM and less so beyond that layer.

The parameters in the lower layers therefore seems to contain most information that are relevant to the de-identification task in general, which supports the common hypothesis that higher layers of ANN architectures contain the parameters that are more specific to the task as well as the dataset used for training.

Despite the observation that transferring a few lower layers may be sufficient for efficient transfer learning, it is interesting to see that adding the topmost layers to the transfer learning does not hurt the performance. When retraining the model on the target dataset, the ANN is able to adapt to the target dataset quite well despite some the higher layers being initialized to parameters that are likely to be more specific to the source dataset.

6. Conclusion

In this work, we have studied transfer learning with ANNs for NER, specifically patient note de-identification, by transferring ANN parameters trained on a large labeled dataset to another dataset with limited human annotations. We demonstrated that transfer learning improves the performance over the state-of-the-art results on two datasets. Transfer learning may be especially beneficial for a target dataset with small number of labels.

7. Bibliographical References

Ananthakrishnan, A. N., Cai, T., Savova, G., Cheng, S.-C., Chen, P., Perez, R. G., Gainer, V. S., Murphy, S. N., Szolovits, P., Xia, Z., et al. (2013). Improving case definition of Crohn's disease and ulcerative colitis in electronic medical records using natural language processing: a novel informatics approach. *Inflammatory bowel diseases*, 19(7):1411.

Dernoncourt, F., Lee, J. Y., Uzuner, O., and Szolovits, P. (2016). De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, page ocw156.

Halpern, Y., Horng, S., Choi, Y., and Sontag, D. (2016). Electronic medical record phenotyping using the anchor and learn framework. *Journal of the American Medical Informatics Association*, page ocw011.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Li, Q. (2012). Literature survey: domain adaptation algorithms for natural language processing. *Department of Computer Science The Graduate Center, The City University of New York*, pages 8–10.

Liao, K. P., Cai, T., Savova, G. K., Murphy, S. N., Karlson, E. W., Ananthakrishnan, A. N., Gainer, V. S., Shaw, S. Y., Xia, Z., Szolovits, P., et al. (2015). Development of phenotype algorithms using electronic medical records and incorporating natural language processing. *bmj*, 350:h1885.

Mou, L., Meng, Z., Yan, R., Li, G., Xu, Y., Zhang, L., and Jin, Z. (2016). How transferable are neural networks in NLP applications? *arXiv preprint arXiv:1603.06111*.

Murdoch, T. B. and Detsky, A. S. (2013). The inevitable application of big data to health care. *Jama*, 309(13):1351–1352.

Office for Civil Rights, H. (2002). Standards for privacy of individually identifiable health information. final rule. *Federal Register*, 67(157):53181.

Oquab, M., Bottou, L., Laptev, I., and Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Pivovarov, R. and Elhadad, N. (2015). Automated methods for the summarization of electronic health records. *Journal of the American Medical Informatics Association*, 22(5):938–947.

Stamate, C., Magoulas, G. D., and Thomas, M. S. (2015). Transfer learning approach for financial applications. *arXiv preprint arXiv:1509.02807*.

Wang, D. and Zheng, T. F. (2015). Transfer learning for speech and language processing. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2015 Asia-Pacific*, pages 1225–1237. IEEE.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.

8. Language Resource References

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. (2000). Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220.

Johnson, A. E. W., Pollard, T. J., Shen, L., wei Lehman, L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Scientific Data*, (in press).

Saeed, M., Villarroel, M., Reisner, A. T., Clifford, G., Lehman, L.-W., Moody, G., Heldt, T., Kyaw, T. H., Moody, B., and Mark, R. G. (2011). Multiparameter intelligent monitoring in intensive care II (MIMIC-II): a public-access intensive care unit database. *Critical care medicine*, 39(5):952.

Stubbs, A., Kotfila, C., and Uzuner, Ö. (2015). Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task track 1. *Journal of biomedical informatics*, 58:S11–S19.

ForFun 1.0: Prague Database of Forms and Functions An Invaluable Resource for Linguistic Research

Marie Mikulová, Eduard Bejček
Charles University, Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics
Malostranské náměstí 25, 118 00 Prague 1, Czech Republic
{mikulova,bejcek}@ufal.mff.cuni.cz

Abstract

In this paper, we introduce the first version of ForFun, Prague Database of Forms and Functions, as an invaluable resource for profound linguistic research, particularly in describing syntactic functions and their formal realizations. ForFun is built with the use of already existing richly syntactically annotated corpora, collectively called Prague Dependency Treebanks. ForFun brings this complex annotation of Czech sentences closer to researchers. We demonstrate that ForFun 1.0 provides valuable and rich material allowing to elaborate various syntactic issues in depth. We believe that nowadays when corpus linguistics differs from traditional linguistics in its insistence on a systematic study of authentic examples of language in use, our database will contribute to the comprehensive syntactic description.
Keywords: language resource, dependency syntax, semantic labeling, surface form, digital humanities

1. Motivation

What is the difference between location expressions "walk in King Street", "walk on King Street", and "walk along King Street"? Should we use a different preposition when talking about destination rather than about direction and location? Or more precisely, what function does the preposition "on" perform in contrast to the preposition "along" and which forms can express destination? Is the same form used in both spoken and written text? Is there any bias towards one form in translated text? For Czech, the answers can be found in a new database for inspecting thousands of real examples categorized by their *form* (e.g. by a prepositional case) as well as by their deep syntactic *function*.

2. Introduction

In this paper, we present the first version of ForFun, Prague Database of Forms and Functions, as an invaluable resource for different linguistic issues, particularly for the description of syntactic functions and their formal realizations. It takes advantage of several richly syntactically annotated corpora, collectively called Prague Dependency Treebanks (PDTs in the sequel) that have already been developed in Prague. Altogether, the treebanks contain around 180,000 sentences with their morphological, syntactic and semantic annotation. The ForFun database draws on the complex linguistic annotation of these corpora, arranges selected morphological and syntactical annotation into new shape, and offers a user-friendly access to a large resource of real examples.

3. Related Work

There is a wide range of corpora with rich linguistic annotation, e.g., Penn Treebank (Marcus et al., 1999), its successors PropBank (Kingsbury and Palmer, 2002) and NomBank (Meyers et al., 2004); for German, there is Tiger (Brants et al., 2002) and Salsa (Burchardt et al., 2006), and many others. The ForFun database is unique in that it is compiled from four different treebanks of Czech, uniformly annotated using the same scenario, with data coming from text, speech and Internet sources. It offers a really large material with the deep syntactic manual annotation which is well and comprehensibly sorted and easily accessible.

4. Data Resources

The database ForFun is extracted from PDTs. PDTs are the complex linguistically motivated treebanks based on the dependency syntactic theory, which provide interlinked hierarchical layers of standoff annotation. Their annotation scenario is described in detail e.g. in Hajič et al. (2017) and Mikulová et al. (2006).

The **Prague Dependency Treebank** version 3.5[1] (Hajič et al., 2018) is the newest edition of the core Prague Dependency Treebank published in 2006 (Hajič et al., 2006). The data consist of articles from Czech daily newspapers.

A slightly modified scenario was then used for the annotation of the Prague Czech-English Dependency Treebank, the Prague Dependency Treebank of Spoken Czech, and the PDT-Faust corpus. In contrast to the original project of PDT, in these treebanks, the morphological and surface syntactic annotations were done automatically, and the manually annotated deep syntactic layer does not contain annotation of information structure and some other special annotations. However, the annotation of functors (see sect. 5), which we use for building the ForFun database, has been done manually in all four treebanks.

The **Prague Czech-English Dependency Treebank** version 2.0[2] (Hajič et al., 2012), (Hajič et al., 2012) is a manually parsed Czech-English parallel corpus. The English part consists of the Wall Street Journal sections of the Penn Treebank (Marcus et al., 1999). The Czech part, which is used in the database, was manually translated from the English original.

The **Prague Dependency Treebank of Spoken Czech** version 2.0[3] (Mikulová et al., 2017b), (Mikulová et al., 2017)

[1] http://ufal.mff.cuni.cz/pdt3.5
[2] https://ufal.mff.cuni.cz/pcedt2.0/
[3] https://ufal.mff.cuni.cz/pdtsc2.0

contains slightly moderated testimonies of Holocaust survivors from the Malach project corpus[4] and dialogues (two participants chat over a collection of photographs) recorded for the EC-funded Companions project.[5]
The **PDT-Faust** corpus is a small treebank containing short segments (very often with vulgar content) typed in by various users on the `reverso.net` webpage for translation.

5. Functions and Forms

An exploration of what formal means (forms) are used for expressing various syntactic functions is one of the main tasks in syntax. The approach "from function to form" (corresponding to generation in computational linguistics) is the basic one. The reversed process – "from form to function" (corresponding to analysis) – describing conditions in which a partial form has the given function and not another one is also not omitted in syntactic research.
The basic semiotic relation between the function and form (known from the Saussure´s structural linguistics (Saussure, 1916) as the relation between "signifié" and "signifiant") is in the PDTs framework (called the Functional Generative Description, see Sgall et al. (1986)) perceived as a relation between two language layers. Concerning the relation between syntactic functions and forms, we deal with the surface layer (for forms) and deep syntactic layer (for functions). The deep syntactic layer of PDTs represents the most complex linguistic annotation that combines syntax and semantics in the form of semantic labeling, co-reference annotation, and argument structure description based on a valency lexicon. The types of the (semantic) dependency relations are represented by the *functor* attribute attached to all nodes. Functors are classified according to different criteria. The basic subdivision is based on valency. The valency criterion divides functors into argument functors and adjunct functors. There are five argument functors: Actor/Bearer (`ACT`), Patient (`PAT`), Addressee (`ADDR`), Origin (`ORIG`), and Effect (`EFF`). The repertory of adjuncts is much larger than that of arguments: their set might be divided into several subclasses, such as temporal, spatial, causal, etc. Other relations such as e.g. relations between the members of coordination or between parts of multi-word expressions, are also labeled by functors. A shortened list of functors is presented in Table 1. For a full list of all dependency functions and their descriptions and labels see (Mikulová et al., 2006). The theoretical description of the valency theory and deep syntactic functions (as developed originally in the theoretical framework of Functional Generative Description and then applied in PDTs) is summarized mainly by Panevová (1974; 1998; 1999).
The lower layers of PDTs contain surface syntax and morphological annotation. Among others they contain information about the formal realization of sentence units (e.g., POS, grammatical cases) in the form of morphological tags assigned to all tokens.
There is a one-to-one correspondence between the tokens at the morphological layer and the nodes at the surface syntactic layer. But there is no such clear correspondence between the nodes at the surface syntactic layer and the deep syntactic layer. The nodes of the deep syntactic layer represent semantic units, i.e. one node for each content word together with its auxiliary words such as prepositions, conjunctions or auxiliary verbs. For example, the prepositional phrase "on street" is represented by one node with the lemma "street". To preserve the original information, nodes on the surface layer are explicitly referred to from this node. Thus there are two links from the node "street" to the surface layer: to the noun "street" and to the preposition "on". These links allow to combine information from different layers of the corpus. We take a big advantage of this linking in building the ForFun database.

Spatial functors		**Causal functors**	
`LOC`	where?	`CAUS`	cause
`DIR1`	where from	`AIM`	aim
`DIR2`	which way?	`CNSC`	concession
`DIR3`	where to?	`COND`	condition
		`INTT`	intention
Temporal functors			
`TWHEN`	when?	**Coordination relations**	
`TSIN`	since when?	`CONJ`	conjunction
`TTILL`	till when?	`ADVS`	adversative
`THL`	how long?	`CSQ`	consecution
`TFHL`	for how long?	`CONFR`	confrontation
`THO`	how often?	`DISJ`	disjunction
`TPAR`	during what time?	`GRAD`	gradation
`TFRWH`	from when?	`REAS`	reason
`TOWH`	to when?	`APPS`	apposition
Functors for manner		**Other functors**	
`MANN`	manner	`ACMP`	accompaniment
`CPR`	comparison	`INTF`	intensifier
`CRIT`	criterion	`BEN`	benefactor
`DIFF`	difference	`RHEM`	rhematizer
`EXT`	extent	`RSTR`	attribute

Table 1: Shortened list of functors (the total number is 66).

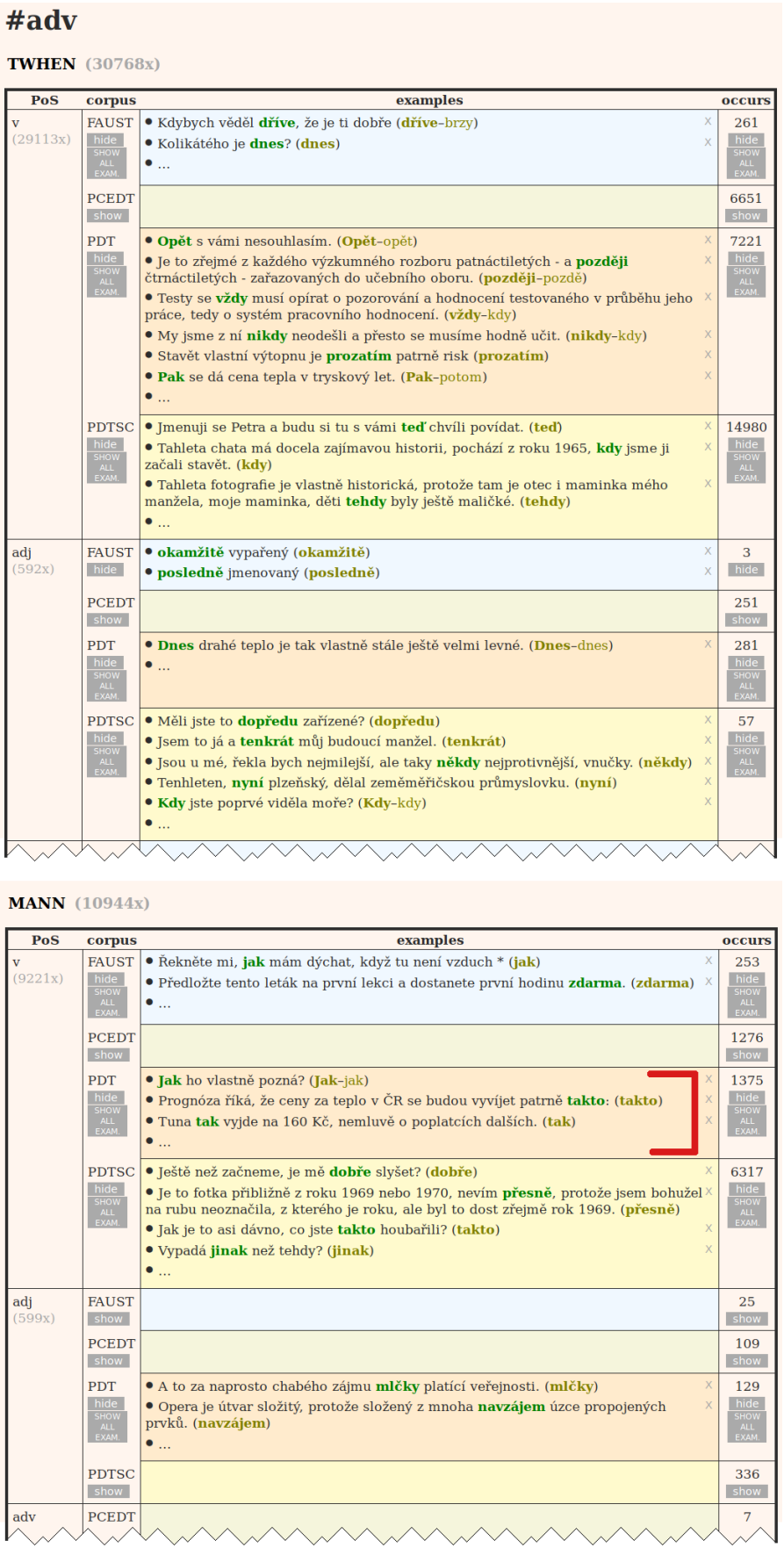
6. ForFun 1.0

Prague Database of Forms and Functions 1.0 (ForFun 1.0) is a rich database of syntactic functions and their formal realizations with a large amount of examples coming from both written and spoken Czech texts. The database is extracted from PDTs (see Sect. 4) and it is provided as a digital open source accessible to all scholars via the LINDAT/CLARIN language resource open repository.[6]

6.1. Design

In language, one form can usually represent various functions, and one function can have several forms. Thus the distinction between form and function is a useful way to tackle two main syntactic approaches: "from function to form" and "from form to function".
The ForFun database is split in the same manner into two interconnected but reversed sets (cf. Figure 1 and Figure 2).

[4] `https://ufal.mff.cuni.cz/cvhm/vha-info.html`
[5] `http://cordis.europa.eu/project/rcn/96289_en.html`
[6] `http://hdl.handle.net/11234/1-2542`



#adv

TWHEN (30768x)

PoS	corpus	examples	occurs
v (29113x)	FAUST hide SHOW ALL EXAM.	• Kdybych věděl **dříve**, že je ti dobře (**dříve**–brzy) x • Kolikátého je **dnes**? (**dnes**) x • ...	261 hide SHOW ALL EXAM.
	PCEDT show		6651 show
	PDT hide SHOW ALL EXAM.	• **Opět** s vámi nesouhlasím. (**Opět**–opět) x • Je to zřejmé z každého výzkumného rozboru patnáctiletých - a **později** čtrnáctiletých - zařazovaných do učebního oboru. (**později**–pozdě) x • Testy se **vždy** musí opírat o pozorování a hodnocení testovaného v průběhu jeho práce, tedy o systém pracovního hodnocení. (**vždy**–kdy) x • My jsme z ní **nikdy** neodešli a přesto se musíme hodně učit. (**nikdy**–kdy) x • Stavět vlastní výtopnu je **prozatím** patrně risk (**prozatím**) x • **Pak** se dá cena tepla v tryskový let. (**Pak**–potom) x • ...	7221 hide SHOW ALL EXAM.
	PDTSC hide SHOW ALL EXAM.	• Jmenuji se Petra a budu si tu s vámi **teď** chvíli povídat. (**teď**) x • Tahleta chata má docela zajímavou historii, pochází z roku 1965, **kdy** jsme ji začali stavět. (**kdy**) x • Tahleta fotografie je vlastně historická, protože tam je otec i maminka mého manžela, moje maminka, děti **tehdy** byly ještě maličké. (**tehdy**) x • ...	14980 hide SHOW ALL EXAM.
adj (592x)	FAUST hide	• **okamžitě** vypařený (**okamžitě**) x • **posledně** jmenovaný (**posledně**) x	3 hide
	PCEDT show		251 show
	PDT hide SHOW ALL EXAM.	• **Dnes** drahé teplo je tak vlastně stále ještě velmi levné. (**Dnes**–dnes) x • ...	281 hide SHOW ALL EXAM.
	PDTSC hide SHOW ALL EXAM.	• Měli jste to **dopředu** zařízené? (**dopředu**) x • Jsem to já a **tenkrát** můj budoucí manžel. (**tenkrát**) x • Jsou u mé, řekla bych nejmilejší, ale taky **někdy** nejprotivnější, vnučky. (**někdy**) x • Tenhleten, **nyní** plzeňský, dělal zeměměřičskou průmyslovku. (**nyní**) x • **Kdy** jste poprvé viděla moře? (**Kdy**–kdy) x • ...	57 hide SHOW ALL EXAM.

MANN (10944x)

PoS	corpus	examples	occurs
v (9221x)	FAUST hide SHOW ALL EXAM.	• Řekněte mi, **jak** mám dýchat, když tu není vzduch * (**jak**) x • Předložte tento leták na první lekci a dostanete první hodinu **zdarma**. (**zdarma**) x • ...	253 hide SHOW ALL EXAM.
	PCEDT show		1276 show
	PDT hide SHOW ALL EXAM.	• **Jak** ho vlastně pozná? (**Jak**–jak) x • Prognóza říká, že ceny za teplo v ČR se budou vyvíjet patrně **takto**: (**takto**) x • Tuna **tak** vyjde na 160 Kč, nemluvě o poplatcích dalších. (**tak**) x • ...	1375 hide SHOW ALL EXAM.
	PDTSC hide SHOW ALL EXAM.	• Ještě než začneme, je mě **dobře** slyšet? (**dobře**) x • Je to fotka přibližně z roku 1969 nebo 1970, nevím **přesně**, protože jsem bohužel na rubu neoznačila, z kterého je roku, ale byl to dost zřejmě rok 1969. (**přesně**) x • Jak je to asi dávno, co jste **takto** houbařili? (**takto**) x • Vypadá **jinak** než tehdy? (**jinak**) x • ...	6317 hide SHOW ALL EXAM.
adj (599x)	FAUST show		25 show
	PCEDT show		109 show
	PDT hide SHOW ALL EXAM.	• A to za naprosto chabého zájmu **mlčky** platící veřejnosti. (**mlčky**) x • Opera je útvar složitý, protože složený z mnoha **navzájem** úzce propojených prvků. (**navzájem**) x • ...	129 hide SHOW ALL EXAM.
	PDTSC show		336 show
adv	PCEDT		7

Figure 1: A screenshot of the form adverb in ForFun. The figure presents only a part of the full response obtained from the ForFun database. Adverb can serve for as many as 55 functions (see also Table 2), two of them (`TWHEN` and `MANN`) are shown here.

Function		Examples	Raw Frequency
TWHEN	When?	*dnes* ‘today’; *hned* ‘immediately’; *pozdě* ‘late’; *nikdy* ‘never’	29113
LOC	Where?	*venku* ‘outside’; *doma* ‘at home’; *všude* ‘everywhere’; *dole* ‘down’	16251
MANN	How?	*krásně* ‘beautifully’; *dobře* ‘well’; *detailně* ‘in detail’; *trpělivě* ‘patiently’	9221
DIR3	Where to?	domů ‘home’; zpět ‘back’; jinam ‘elsewhere’; dovnitř ‘indoors’	4357
EXT	How much?	*příliš* ‘too much’; *vůbec* ‘not at all’; *úplně* ‘entirely’; *trošku* ‘a little’	3815
THO	How often?	*často* ‘often’; *občas* ‘sometimes’; *zřídka* ‘rarely’; *pravidelně* ‘regularly’	3211
THL	How long?	*ještě* ‘still’; *pořád* ‘all the time’	2721
TTILL	Till when?	*doposud* ‘heretofore’; *dodnes* ‘up to now’	762
CAUS	Why?	*náhodou* ‘accidentally’; *právem* ‘by right’	648
DIR1	Where from?	*odtud* ‘thereof’; *zdola* ‘from below’; *zprava* ‘from the right’	503
		⋮	
		another 43 functions	
		⋮	
MEANS	means	*koňmo* ‘on horseback’; *ručně* ‘manually’	52
TOWH	To when?	*nakonec* ‘finally’	4

Table 2: Functions of adverbs. Shortened list (total number of functions is 55) gained from ForFun. For each function some examples with translation and the number of examples in the database is given.

The **“from function to form”** set contains a list of all deep syntactic functions (66 functors altogether). When choosing one function type, the user can search for all forms that may represent that function. (See Figure 2 that shows a result of a search for a function “manner”.) For each function-form relation there are plenty of examples in the form of sentence with the highlighted expression representing the relation. All examples are sorted by various criteria:

- the word class of a parent node,
- the particular forms for the function and
- the source of text data (written, spoken, translated texts and texts from Internet users).

The number of examples available in the database is always shown for each specified 4-combination (given form, functor, parent word class and source). Either the first ten or all examples are displayed on demand.

The **“from form to function”** set contains a long list (almost 1 500 items) of all formal realizations of particular sentence units that occur in PDTs: prepositionless cases, prepositional cases, subordinated and coordinate conjunctions, adverbs, infinitive and finite verb forms, etc. For any form (see Figure 1 for adverbs), there are again plenty of examples sorted by function, word class of the parent node, and the source of text data, always with the frequency in the data. In both sets, examples can be also filtered by their source, which allows the user to hide e.g. all forms used only in spoken language or use only sentences from written corpora.

An illustration of how the result of user search for all functions of an adverb phrase looks like is given in Figure 1. In the upper part, there are examples of the form “#adv” (meaning either an adverb phrase or an adverb as a word) representing the time expression “when” (i.e. the functor TWHEN); there are 30 768 occurrences available. The occurrences of adverb form are divided according to their syntactic parents (be it a v(erb), adj(ective), adv(erb) or a n(oun), see the first column); their distribution within particular treebank is given in the second column followed by real examples from the corresponding treebank. A sample of them is displayed on demand right in the table whereas many others (see the last column for their numbers) stay hidden and can be displayed in a full list.

In the lower part of Figure 1, the same form “#adv” is exemplified in the same style as an expression of manner (i.e. the functor MANN, third most frequent). See Table 2 for functions represented by adverbs other than TWHEN and MANN.

For the opposite direction (“from function to form”), see Figure 2, where (among others) the same sentences (for adverb form) can be found when searching for all representations of the functor MANN (see the sentence *Jak ho vlastně pozná?* and others in both Figures 1 and 2). Other forms are less frequent and include a preposition *na* together with either a genitive, accusative or locative case[7] or a preposition *mimo* with an accusative case etc.; see also Table 4.

6.2. Volume

The database contains 2.2 million examples altogether for all forms (and the same number from the function point of view), split approx. 3:1 between written and spoken text (see Table 3). Each example is one sentence long.[8] They can be examined from the function side (66 functors) or the form side (1 469 forms). All examples are split into 13.5 thousand of 4-combinations, each with 163 examples on average. There is also a 4-combination with almost 100 000 examples. Maximum number of examples for a function is 490 000 across all forms and corpus sources (function RSTR). Maximum number of examples for a form is 370 000 (nominative case).

While the average number is high, the median is only two examples. The reason is that there is a long tail of 4-combi-

[7] Morphological cases in ForFun are indicated by numbers, thus forms mentioned above are shortened as *na#2*, *na#4* and *na#6*.

[8] One sentence typically contains many different functions and can be used once for each of its parts.

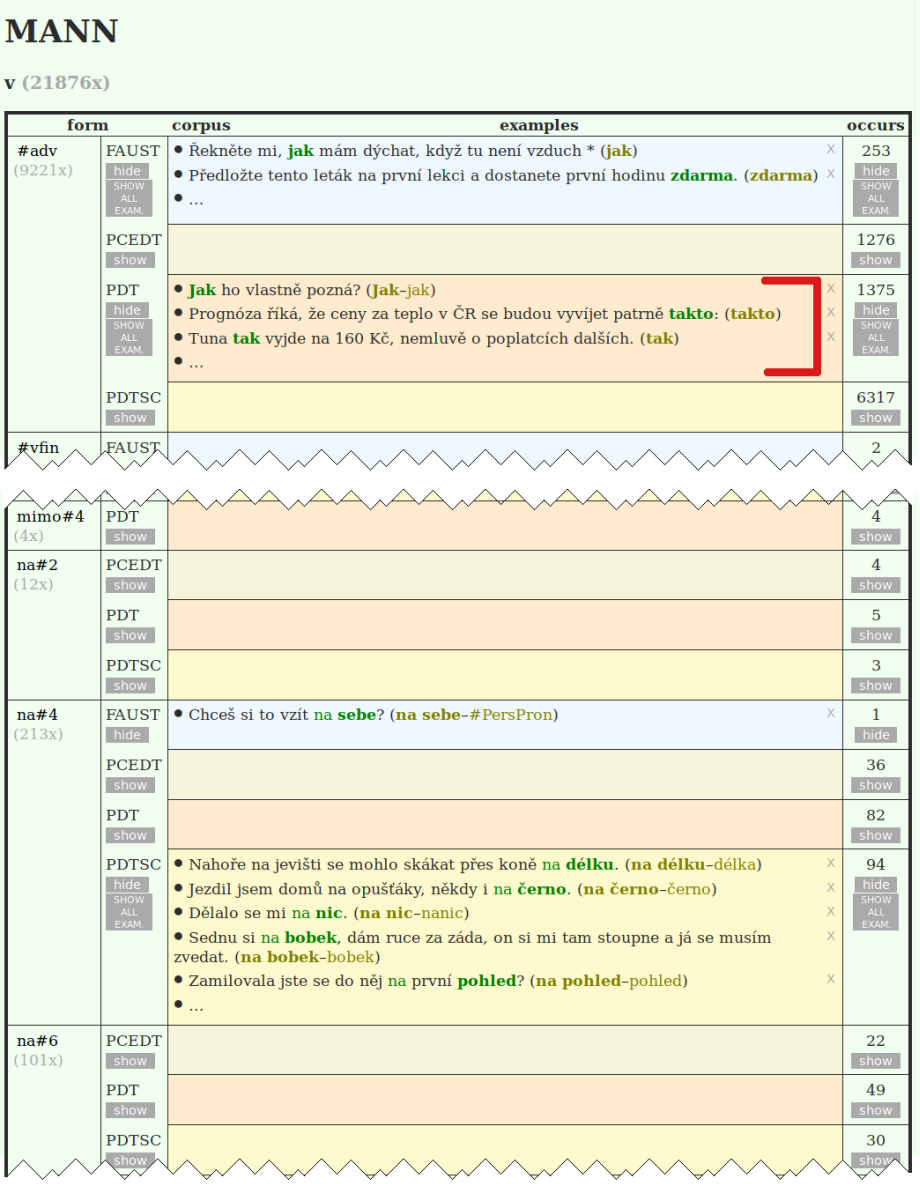


Figure 2: A screenshot of function "manner" in ForFun. The figure presents only a part of the full response obtained from the ForFun database, because the functor MANN is represented by as many as 122 forms (see also Table 4).

nations used very rarely. These occurrences with very low frequencies in the data are one of the main benefits of the large volume of database, but they have to be used carefully. Every result has to be always understood solely as an input for a subsequent research, as ForFun may contain errors (caused by annotators as well as speakers/writers), especially considering its volume.

7. What is ForFun good for?

Linguistic research is predominantly text-based. Before the corpus era, researchers had to rely on their own excerpts; nowadays, however, a vast amount of supporting material is available in digital form. Such resources are truly valuable only if they are enriched with different layers of linguistic annotation ranging from morphology and syntax to semantics. However, there are many researchers who (want to) use corpora in their everyday work and look for various occurrences of specific words, forms or patterns, syntactic functions, etc. but they are not interested or just do not need to deal with various technical, formal and annotation issues (because they are just researchers in humanities and not so fluent also in technology). Moreover, often if an annotation scenario is based on a sound linguistic theory, it is quite complex and perhaps too complicated for everyday users. Thus, there is a requirement for voluminous and richly linguistically annotated resource which is easy to use. And that is ForFun!

The ForFun database brings the rich and complex annotation in PDTs closer to such everyday, simple use. A rather straightforward use of ForFun is to retrieve which functions can be expressed by the particular form and which forms can express the particular function. To display the richness of the material in the ForFun database we present here two simple examples. Table 2 demonstrates multifunctionality of form (we choose the adverb phrase as an example) and Table 4 demonstrates formal diversity of

examples from written text	1 608 061
examples from spoken text	593 400
examples altogether	2 201 461
number of functions	66
number of forms	1 469
number of 4-combinations	13 514
avg. examples for function	33 355
avg. examples for form	1 500
avg. examples for a 4-combination	163
max. number of examples for a function	490 121
max. number of examples for a form	370 586
max. number of examples for a 4-combination	97 469

Table 3: Volume of the ForFun database.

function (we choose "manner" as example, i.e., the functor`MANN`). We can see that the relation between forms and their functions is many-to-many, one form is used for expressing many functions and one function can be expressed using various forms (see also Bejček et al. (2017)).
Besides analysis of the form-function relation, ForFun is user-friendly source of examples for other various explorations in syntax, e.g., valency behavior, coordination/discourse relations, idioms and complex predicates, comparison of written and spoken texts, etc. The first linguistic studies based on the ForFun database analyze subtle meanings of spatial and temporal adverbials (2017a; 2018).

8. Conclusion

We have introduced a unique resource for linguistic studies in syntax: the ForFun 1.0 database. We have demonstrated that ForFun is:

- a simplified interface to PDTs,
- a tool primarily for linguists,
- a database of 180 000 Czech sentences,
- a source of information about syntax,
- a place where 2.2 million examples can be studied,
- a gateway to forms (for a given function),
- a gateway to functions (of a given form).

We believe that nowadays when corpus linguistics differs from traditional linguistics in its insistence on a systematic study of authentic examples of language in use, our database will contribute to a comprehensive syntactic studies.

Acknowledgements

The research reported in the paper was supported by the Czech Science Foundation under the project GA17-12624S. This work has also been supported by the LINDAT/CLARIN project of Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071).

Bibliographical References

Bejček, E., Hajičová, E., Mikulová, M., and Panevová, J. (2017). The relation of form and function in linguistic theory and in a multi-layer treebank. In Jan Hajič, editor, *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 56–63, Praha, Czechia. Univerzita Karlova, Univerzita Karlova.

Brants, S., Dipper, S., Hansen, S., Lezius, W., and Smith, G. (2002). The tiger treebank. In *Proceedings of the workshop on treebanks and linguistic theories*, volume 168.

Burchardt, A., Erk, K., Frank, A., Kowalski, A., Padó, S., and Pinkal, M. (2006). The salsa corpus: a german corpus resource for lexical semantics. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006)*, pages 969–974. European Language Resources Association.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Bojar, O., Cinková, S., Fučíková, E., Mikulová, M., Pajas, P., Popelka, J., Semecký, J., Šindlerová, J., Štěpánek, J., Toman, J., Urešová, Z., and Žabokrtský, Z. (2012). Announcing prague czech-english dependency treebank 2.0. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 3153–3160. European Language Resources Association, Istanbul, Turkey.

Hajič, J., Hajičová, E., Mikulová, M., and Mírovský, J., (2017). *Prague Dependency Treebank,* Handbook on Linguistic Annotation, pages 555–594. Springer Verlag, Dordrecht, Netherlands.

Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *Proceedings of the 2th International Conference on Language Resources and Evaluation (LREC-2002)*, pages 1989–1993. European Language Resources Association.

Marcus, M., Santorini, B., Marcinkiewicz, M. A., and Taylor, A. (1999). *Penn Treebank-3*. Linguistic Data Consortium, LDC99T42, University of Pennsylvania.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., and Grishman, R. (2004). Annotating noun argument structure for nombank. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004)*, pages 803–806. European Language Resources Association.

Mikulová, M., Bémová, A., Hajič, J., Hajičová, E., Havelka, J., Kolářová, V., Kučová, L., Lopatková, M., Pajas, P., Panevová, J., Razímová, M., Sgall, P., Štěpánek, J., Urešová, Z., Veselá, K., and Žabokrtský, Z. (2006). Annotation on the tectogrammatical level in the Prague Dependency Treebank. Annotation manual. Technical Report 30, Prague, Czech Rep.

Mikulová, M., Bejček, E., Kolářová, V., and Panevová, J. (2017a). Subcategorization of adverbial meanings based on corpus data. *Journal of Linguistics / Jazykovedný časopis*, 68(2):268–277.

Mikulová, M., Mírovský, J., Nedoluzhko, A., Pajas, P., Štěpánek, J., and Hajič, J. (2017b). PDTSC 2.0 - spoken corpus with rich multi-layer structural annotation. In *Text, Speech, and Dialogue 20th International Conference, TSD 2017*, Lecture Notes in Computer Science, pages 129–137, Cham / Heidelberg / New York / Dordrecht / London. Charles University, Springer International Publishing.

Mikulová, M., Bejček, E., and Panevová, J. (2018). What can we find out about time and space in the forfun database? In *Proceedings of the Second Workshop on Corpus-Based Research in the Humanities*

Form	Examples	Raw Frequency
adverb	***Hlasitě*** *se hádali.* 'They was arguing **loudly**.'	9221
v+locative	*Chodil tam* ***ve starých teplákách****.* 'He has been walking there **in old tracksuit**.'	1000
instrumental	*Lupiči mu* ***násilím*** *ukradli hodinky.* 'The robbers has stolen his watches **by force**.'	747
(*jak*)+verb	*Všechno je to tak,* ***jak to má být****.* 'Everything is **as it should be**.'	275
(*tak*) *že*+verb	*V soukromí se tváří,* ***že nevěří ničemu****.* 'In private she pretends **she does not believe anything**'	251
na+accusative	*Zamilovali jste se do sebe* ***na první pohled****?* 'Have you fallen in love to each other **at first sight**?'	213
pod+instrumental	*Hráli jsme* ***pod velkým psychickým tlakem****.* 'We've been playing **under great psychic pressure**.'	150
s+instrumental	*Domácí hráli* ***s nadšením****.* 'Locals have played **with a passion**.'	150
na+locative	***Na jedné noze*** *se tenis hrát nedá.* 'A tennis cannot be played **on one leg**.'	101
po+locative	***Po tmě*** *jsem ráno šel na vlak.* 'I've gone for a train **in the dark**.'	81
	⋮ another 110 forms ⋮	
ve formě+genitive	*publikuje* ***ve formě přehledů*** *a tabulek* 'he publishes **in the form of surveys**'	14
přes+accusative	*obchoduje* ***přes přepážky*** 'she trades **over the counters**'	14

Table 4: Formal realizations of the MANN functor. Shortened list (from more than 100 rows) of the forms that can express manner (functor MANN) gained from ForFun. For each form the number of examples in the database is given.

CRH-2, Gerastree proceedings, pages 133–142. Austrian Academy of Science, Dept. of Geoinformation, Wien, Austria.

Panevová, J. (1974). On verbal frames in functional generative description. *Prague Bulletin of Mathematical Linguistics*, 22(3-40):6–3.

Panevová, J. (1998). Ještě k teorii valence. In *Slovo a slovesnost 59, č.1*.

Panevová, J. (1999). Valence a její univerzální a specifické projevy. In Petr Karlík Zdeňka Hladká, editor, *Čeština - univerzália a specifika. Sborník konference ve Šlapanicích u Brna 17.-18. 11. 1998*. Masarykova univerzita, Brno.

Saussure, F. d. (1916). Cours de linguistique générale, ed. *C. Bally and A. Sechehaye, with the collaboration of A. Riedlinger, Lausanne and Paris: Payot*.

Sgall, P., Hajičová, E., and Panevová, J. (1986). *The Meaning of the Sentence and Its Semantic and Pragmatic Aspects*. Academia/Reidel Publishing Company, Prague/Dordrecht.

Language Resource References

Hajič, J., Panevová, J., Hajičová, E., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M., Žabokrtský, Z., Ševčíková-Razímová, M., and Urešová, Z. (2006). *Prague Dependency Treebank 2.0 (LDC2006T01)*. Linguistic Data Consortium.

Hajič, J., Bejček, E., Bémová, A., Buráňová, E., Hajičová, E., Havelka, J., Homola, P., Kárník, J., Kettnerová, V., Klyueva, N., Kolářová, V., Kučová, L., Lopatková, M., Mikulová, M., Mírovský, J., Nedoluzhko, A., Pajas, P., Panevová, J., Poláková, L., Rysová, M., Sgall, P., Spoustová, J., Straňák, P., Synková, P., Ševčíková, M., Štěpánek, J., Urešová, Z., Vidová Hladká, B., Zeman, D., Zikánová, Š., and Žabokrtský, Z. (2018). *Prague Dependency Treebank 3.5*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University, LINDAT/CLARIN PID: http://hdl.handle.net/11234/1-2621.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Cinková, S., Fučíková, E., Mikulová, M., Pajas, P., Popelka, J., Semecký, J., Šindlerová, J., Štěpánek, J., Toman, J., Urešová, Z., and Žabokrtský, Z. (2012). *Prague Czech-English Dependency Treebank 2.0*. Charles

University in Prague, UFAL, LINDAT/CLARIN PID: http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4.

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., and Taylor, A. (1999). *Penn Treebank*. LDC, ISLRN 141-282-691-413-2.

Mikulová, M., Bémová, A., Hajič, J., Hajičová, E., Ircing, P., Kolářová, V., Lopatková, M., Mareček, D., Mírovský, J., Nedoluzhko, A., Pajas, P., Panevová, J., Peterek, N., Romportl, J., Sgall, P., Ševčíková, M., Štěpánek, J., Urešová, Z., and Žabokrtský, Z. (2017). *Prague Dependency Treebank of Spoken Czech 2.0*. Charles University in Prague, UFAL.

The LIA Treebank of Spoken Norwegian Dialects

Lilja Øvrelid[†], Andre Kåsen [†‡], Kristin Hagen[‡], Anders Nøklestad[‡], Per Erik Solberg[*] and Janne Bondi Johannessen[‡]

[†] Department of Informatics, University of Oslo

[‡] Department of Linguistics and Scandinavian Studies, University of Oslo

[*] Department of Philosophy, Classics, History of Arts and Ideas, University of Oslo

{liljao,andrekaa}@ifi.uio.no, {kristin.hagen,j.b.johannessen, anders.noklestad}@iln.uio.no, p.e.solberg@ifikk.uio.no

Abstract

This article presents the LIA treebank of transcribed spoken Norwegian dialects. It consists of dialect recordings made in the period between 1950–1990, which have been digitised, transcribed, and subsequently annotated with morphological and dependency-style syntactic analysis as part of the LIA (Language Infrastructure made Accessible) project at the University of Oslo. In this article, we describe the LIA material of dialect recordings and its transcription, transliteration and further morphosyntactic annotation. We focus in particular on the extension of the native NDT annotation scheme to spoken language phenomena, such as pauses and various types of disfluencies, and present the subsequent conversion of the treebank to the Universal Dependencies scheme. The treebank currently consists of 13,608 tokens, distributed over 1396 segments taken from three different dialects of spoken Norwegian. The LIA treebank annotation is an on-going effort and future releases will extend on the current data set.

Keywords: treebanks, spoken language, dialects, Norwegian, Universal Dependencies

1. Introduction

Large-scale initiatives like the CoNLL shared tasks on dependency parsing (Surdeanu et al., 2008), the Universal Dependencies (UD) initiative (Nivre et al., 2016) and the recent shared task on multilingual parsing from raw text (Zeman et al., 2017) have made available syntactic treebanks for a large number of languages, thus enabling parsing research for a wide variety of languages. Available treebanks are still, however, largely based on written textual resources, with a few exceptions (Dobrovoljc and Nivre, 2016; Östling et al., 2017).

The LIA project[1] has as its main objective to create a corpus consisting of old dialect recordings and make these accessible for research in linguistics and digital humanities. By digitization, transcription and further linguistic processing, this corpus can play an important role in the diachronic study of Norwegian dialects and more generally the linguistic variation in Norway. This article describes the LIA treebank of spoken Norwegian dialects. A longterm goal of this work is to develop a parser for spoken Norwegian, with the immediate goal of parsing the whole LIA material. This will enable more fine-grained linguistic analyses to be carried out over the material.

In this paper we present the LIA data set, its transcription and subsequent morphological and syntactic annotation, with a focus on the extended annotation guidelines of the Norwegian Dependency Treebank (NDT) for spoken language phenomena and the conversion of the treebanked data to the Universal Dependencies (UD) scheme (de Marneffe et al., 2014). The UD version of the treebank was made available with the v2.1 release of the UD treebanks (Nivre et al., 2017).

[1] http://www.hf.uio.no/iln/english/research/projects/language-infrastructure-made-accessible

2. The LIA material

The LIA project (Language Infrastructure made Accessible) is a five-year national collaboration project between four Norwegian universities (University of Oslo, University of Bergen, University of Tromsø and The Norwegian University of Science and Technology), Norsk ordbok 2014 and Språkbanken at the National Library, in addition to international partners.

The main aim of the LIA project is to collect dialect recordings from the four participating universities, digitise them, inventorise, catalogue and safely store them and make them accessible for further research. The most interesting recordings are transcribed and text-sound synchronised with the transcription tool ELAN[2]. Finally they are morphologically tagged and parsed. This process is described below. The final outcome of the LIA project is a user-friendly searchable dialect corpus.

The audio files that constitute the data set are recorded between 1950 and 1990 in order to explore and survey the many different dialects in Norway. Sometimes the research questions also concern person or place names. Most of the informants are older people who are native speakers of their dialect. Typically, the recordings are interviews about old trades such as agriculture, fisheries, logging and life at the summer farm. Other topics are weaving, knitting, baking or dialects. The recordings are semi-formal or informal and often take place in an informant's home.

2.1. Transcription

The LIA project makes use of a semi-phonetic transcription standard, similar to that of Papazian and Helleland (2005) and described in Hagen et al. (2017). This standard is chosen mainly to conserve particularities in the different dialects.

The speech flow is separated into what we call *segments*. A segment is our spoken language approximation of a sen-

[2] https://tla.mpi.nl/tools/tla-tools/elan

tence. Few special characters are in use. The exceptions are the Norwegian letters æ, ø and å, quotation marks for indicating indirect speech, '#' signifying a pause in the speech flow and variations of '+' and '%' combined with a letter indicating unclear speech or laughter etc. The '%' character followed by a letter represents an independent incident in the speech flow like laughter, coughing etc. The '+' characterizes the following word or word group. '+u' means for instance that the following word(s) are unclear. '+x' means that the word(s) are not listed in the dictionary. The variants of '+' and '%' are stripped from the transcripts prior to further morphosyntactic processing. This is done under the assumption that these phenomena do not have any syntactic significance. They are inserted back into the transcripts before the transcripts are made available for search online.

2.2. Transliteration

Before tokenization and lemmatization the semi-phonetic transcriptions are semi-automatically transliterated to standard Norwegian Nynorsk[3] orthography by the Oslo Transliterator[4]. The transliterator can be trained to transliterate any dialect or language variety into any other orthographical representation, and it is so far trained on more than 100 Norwegian dialects in the LIA project. The outcome from the transliterator is manually corrected and the resulting pair of transcriptions are used for training the transliterator for this particular dialect, improving performance on subsequent transliterations of that dialect.

The Oslo Transliterator has a web interface where the transcriptions can be uploaded and associated with the appropriate dialect. The transcriptions are divided into smaller parts, which are transliterated one by one. Each part is manually corrected and added to the training material before the transliterator is trained once more and performs better on the next transcription part. The results of each iteration of the training process are stored in a MySQL database. When all parts are completed, the transcriptions can be downloaded as ELAN files with the semi-phonetic transcription and the orthograpic transcription as separate layers.

New dialects can also be registered in the web interface. Instead of starting from scratch on the new dialect, the transliterator employs a technique in which suggestions for transliterated word forms for the new dialect are based on combinations of stored word form correspondences from a set of dialects selected among those that are already transliterated. Each of those dialects is given a weight based on how similar it is perceived to be to the new dialect by the human transliterator. This technique enables us to take advantage of various degrees of dialect similarity without requiring large amounts of training data or labour-intensive work on creating string similarity mappings.

[3]Norwegian has two official orthographic standards: Bokmål and Nynorsk. For the LIA transcriptions we have chosen to use Nynorsk, the standard closest to most Norwegian dialects.

[4] http://www.hf.uio.no/iln/english/about/organization/text-laboratory/services/oslo-transliterator/

Head	Dependent
Preposition	Prepositional complement
Finite verb	Complementizer
First conjunct	Subsequent conjuncts
Finite auxiliary	Lexical/main verb
Noun	Determiner

Table 1: Annotation choices in the NDT

3. Morphosyntactic annotation

For grammatical phenomena which are not specific to spoken language, we have followed the annotation scheme of the Norwegian Dependency Treebank (NDT) (Solberg et al., 2014). An important reason for this choice was the detailed language-specific annotation guidelines which were developed for the NDT project (Kinn et al., 2014). These guidelines are custom-made for Norwegian, following the Norwegian Reference Grammar (Faarlund et al., 1997) closely. Furthermore, the NDT scheme has performed well in previous measures of inter-annotator agreement. Solberg et al. (2014) report agreement scores of 96.8% unlabeled and 95.3% labeled accuracy and Skjærholt (2014) quantified inter-annotator agreement using a chance-corrected metric derived from Krippendorff's α and showed that agreement on the NDT data is high: scoring an α of about 98%, among the highest of all the data sets studied. This annotation scheme was therefore a natural choice for the current project. An automatic conversion procedure to Universal Dependencies has furthermore been developed for the written NDT data set (Øvrelid and Hohle, 2016; Velldal et al., 2017). It is in other words possible to convert the LIA treebank to the UD annotation scheme, with a few modifications, see section 5.

3.1. The NDT scheme

The Norwegian Dependency Treebank contains manually annotated syntactic and morphological information for both varieties of Norwegian. The part-of-speech annotation follows the Oslo-Bergen Tagger scheme (Hagen et al., 2000). This scheme also marks inflectional features such as tense, number, gender and categories such as demonstrative and quantifier. As mentioned above, the syntactic annotation scheme is, to a large extent, based on the Norwegian Reference Grammar (Faarlund et al., 1997) and the dependency representations are inspired by choices made in comparable treebanks, in particular the Swedish treebank Talbanken (Nivre et al., 2006). Table 1 summarizes the main annotation choices concerning head status and dependency graphs in NDT.

3.2. Preprocessing

The transliterated transcripts are tokenized with whitespace as token delimiter and time code as segment delimiter. Quotation marks and '#'s are considered to be tokens. Lemmatization is completed with lemmas from Norsk ordbank, a lexicographic database for Norwegian.

	Segments	Tokens
Eidsberg	679	5880
Vardø	450	5361
Austevoll	267	2367
total	1396	13608

Table 2: Raw counts for the different informants in the data set.

3.3. Annotation process

Prior to the manual morphosyntactic annotation, the LIA data set was automatically tagged with OBT+stat, a rule-based Constraint Grammar tagger with a HMM-based overlay (Johannessen et al., 2012) and parsed with the MATE parser (Bohnet, 2010) trained on the Nynorsk part of NDT, which consists largely of newspaper text. This parser has been reported to achieve a labeled accuracy score (LAS) of 89.54 on the Nynorsk test set of NDT (Solberg et al., 2014). The automatic tag assignments are then corrected by trained linguists using a browser-based application[5]. The dependency analyses are also manually corrected, following the extended guidelines described in section 4. below. Dependency annotation was performed using the TrEd application, which is the annotation tool developed for the annotation of the Prague Dependency Treebank (Böhmová et al., 2003).

3.4. Treebank data

Our data set, at present, consists of elderly speakers (80+) of the Eidsberg, Austevoll and Vardø dialects. This represents a diverse set of dialects from different regions of the country. Table 2 presents the number of segments and tokens, and their distributions across the three different dialects.

4. The LIA annotation guidelines

Spoken language contains several phenomena that distinguish it from written language, such as various types of disfluencies, repetitions and deletions (Shriberg, 1996; Johannessen and Jørgensen, 2006). Spoken language furthermore contains a larger number of fragmentary segments than written text. In the LIA guidelines we extend the annotation scheme of NDT with dependency analyses of syntactic phenomena that are specific to spoken language. In the following we will describe the main additions to the NDT scheme. These are further summarized in table 3, which shows the added part-of-speech tags and dependency relations. Figure 1 shows the dependency graph for an example sentence from the treebank.

4.1. Spoken language PoS

In order to account for spoken language, some additional PoS tags are added to the tagset. Incomplete or interrupted words are tagged with the tag `ufullst`, pauses ('#') with the tag `pause` and filled pauses or hesitations with `nol`. The category of interjections, we found, is quite frequent in our material. Therefore, a list of standardized interjections has been compiled and these receive the existing NDT tag for interjections (`interj`), see Figure 1 which includes the interjection *å*. Another issue is the preproprial article (investigated in Håberg (2010)) which is wide-spread in spoken Norwegian, but not common in written text. These are assigned the pronominal part-of-speech `pron` but function syntactically as a determiner (`DET`).

[5] `http://github.com/andrely/tag-annotator`

4.2. Spoken language syntax

The extended annotation guidelines for the syntactic annotation of the LIA material is built on the work of Dobrovoljc and Nivre (2016), who describe the annotation of the Slovenian spoken language treebank with Universal Dependencies. Below we describe our treatment of extra-linguistic tokens, various types of disfluencies, ellipsis and discourse elements.

4.2.1. Extra-linguistic tokens

During transcription, some extra-linguistic tokens are introduced in order to mark phenomena such as pauses or unfinished/incomplete words. The examples in (1)-(3) illustrate phenomena that introduce extra-linguistic tokens and will be discussed further below.

(1) *ja # og køyrde mjølka ut i byen igjen*
yes # and drove milk out in town again
'Yes # and drove the milk to town again'

(2) *å det var e det var noko*
oh it was mm it was something
forferdeleg trafikk
terrible traffic
'Oh there was mm there was terrible traffic'

(3) *så det var mykje g- e mykje greier*
so it was much t- mm much things
'So there were many t- mm many things'

Pauses (#), as in (1), and filled pauses (*e* 'mm'), as in (2), are treated similarly in the dependency structure, and are attached to the following dependent. In cases where there is more than one possible attachment site, we attach as high in the tree as possible, while keeping with a projective analysis. These extra-linguistic tokens are assigned the filler dependency relation (`FYLL`), see figure 1. Incomplete or interrupted words, as in example (3), are marked during transcription with a hyphen word-finally. If the incomplete word bears no relation to the surrounding context, it is given the syntactic function `FYLL`, otherwise it is treated as part of a repair relation, see section 4.2.2.

4.2.2. Disfluencies

We distinguish two types of disfluencies in our annotation of the LIA material: repairs and deletions, and introduce two new dependency relations (`REP` and `SLETT`) to account for these.

A repair consists of two parts: the reparandum and the repair. The reparandum is attached to its repair, which is always to the right of it. The repair relation `REP` is used both for repetitions, substitutions and reformulations. The reparandum will have the `REP`-relation to its repair as in

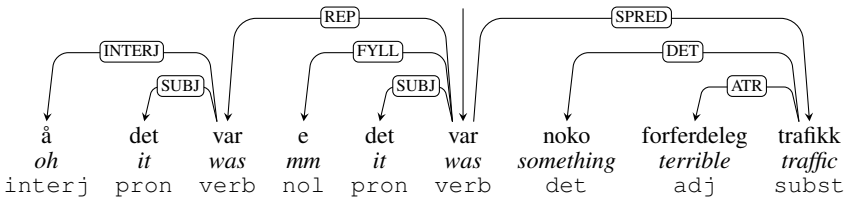


Figure 1: Example sentence from the LIA treebank with corresponding English gloss, PoS and dependency analysis.

	NDT Addition	Description	UD conversion
PoS	nol	filled pauses	X
	ufullst	incomplete words	X
	pause	pauses	PUNCT
Deprel	FYLL	fillers	discourse:filler
	REP	repairs	reparandum
	SLETT	deletions	parataxis:deletion

Table 3: Overview of additions to the NDT schemes in terms of part-of-speech tags and dependency relations in the LIA treebank, along with their converted UD relation.

figure 1, which shows the dependency graph for the example sentence in (2), where *det var* 'it was' is repeated. Note that a repair relation will only be used if there is some shared content between the reparandum and the repair. In the example in figure 1, we see an example of a repetition, where the repair repeats part of the reparandum. Otherwise, the deletion (SLETT) relation should be employed.

A deletion is distinguished from a repair by being semantically unrelated to the subsequent material. Example (4) illustrates a deletion, where the initial part of the sentence *måtte du* 'did you have to' is followed directly by the unrelated sentence *det var rasjonert* 'it was rationed'.

(4) *måtte du det var rasjonert?*
must you it was rationed?
'Did you have to it was rationed?'

Both deletions and repairs are attached as high as possible in the ensuing structure with which it is related, preserving projectivity. Our treatment of deletions departs from that of Dobrovoljc and Nivre (2016), who denote these as "restarts" and choose the incomplete element as head of the ensuing structure (the restart). We follow Shriberg (1996) in naming these deletions and we attach the deleted segment to the restart (the ensuing complete part of the utterance), which is situated to the right. So for example (4) above, the verb *måtte* 'must' is attached to the following finite verb *var* 'was'. This is thus similar to the analysis of discourse fillers (filled pauses), see below, and we preserve the overall structure of the segment despite the initial or internal incomplete structure.

4.2.3. Ellipsis

Ellipsis is a quite common phenomenon in spoken material (Johannessen and Jørgensen, 2006). The LIA treebank follows the treatment of ellipsis adopted in the NDT treebank. It does not introduce empty nodes. So, if the subject of a clause is ellided, as in example (5), there is simply no subject dependent.

(5) *her i bygda var # tretten gardsbruk ##*
here in town was # thirteen farms ##
og to #plassmenn
and two #place-men
'Here in town were thirteen farms and two smallholders'

Fragmentary segments are also common in spoken language (Shriberg, 1996). Segments that lack a finite verb are analyzed as fragments, using the FRAG-relation from the NDT scheme. We follow the prominence hierarchy provided in Kinn et al. (2014) in order to determine the head of the segment. It states that in the absence of a finite verb, head status should be given to non-finite verbs. If there is no non-finite verb, the most prominent element is the subject, followed by indirect objects or subject predicatives, etc. The same hierarchy is employed for cases of verbal ellipsis in coordination, where we follow the NDT guidelines in assigning a dedicated dependency relation KOORD-ELL to the remaining argument.

4.2.4. Discourse elements

For the treatment of interjections, we follow the NDT guidelines, which assign the dependency relation INTERJ to these elements, see figure 1. Interjections may often also constitute the root of a segment, a phenomenon which is common in spoken language.

Discourse fillers (or filled pauses in the terminology of Shriberg (1996)) are assigned a separate part-of-speech tag nol and given the dependency relation FYLL. These elements are attached to the right, as illustrated by the dependency graph in Figure 1, where the discourse filler *e* 'mm' is attached to the following finite verb *var* 'was' with the FYLL relation.

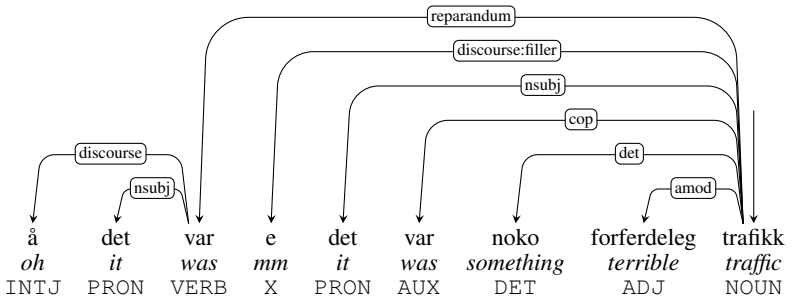


Figure 2: Example sentence from the UD conversion of the LIA treebank with corresponding English gloss, PoS and dependency analysis.

5. Conversion to Universal Dependencies

Universal Dependencies builds on several previous initiatives for universally common morphological (Zeman, 2008; Petrov et al., 2012) and syntactic dependency (McDonald et al., 2013; Rosa et al., 2014) annotation. Among its main tenets is the primacy of content words, i.e., content words, as opposed to function words, are syntactic heads wherever possible. It is intended to be a universal annotation scheme, i.e., applicable to any language, however it also offers some possibilities for language-specific information. With reference to the NDT annotation choices in Table 1, the UD scheme adopts the reverse attachment for auxiliaries, infinitival markers and prepositions.

The NDT and UD schemes differ in terms of both PoS tagset and morphological features, as well as structural analyses. The conversion therefore requires non-trivial transformations of the dependency trees, in addition to mappings of tags and labels that make reference to a combination of various kinds of linguistic information. For instance, in terms of PoS tags, the UD scheme offers a dedicated tag for proper nouns (PROPN), whereas NDT expresses information about noun type among its morphological features. UD further distinguishes auxiliary verbs (AUX) from main verbs (VERB). This distinction is not explicitly made in NDT, hence the conversion procedure makes use of the syntactic context of a verb; verbs that have a non-finite dependent are marked as auxiliaries. Further details about the conversion is given in Øvrelid and Hohle (2016), as well as in Velldal et al. (2017), which describes the extension of the conversion to cover the Nynorsk variant of Norwegian.

When it comes to part-of-speech tags, the universal tagset must be employed and there are few possibilities for language-specific adaptation. For dependency relations, there is the possibility to add treebank-specific subtypes of the universal dependency relations (on the form `udep:subtype`). Table 3 shows the treatment of the spoken language specific PoS tags and dependency relations during conversion to UD. Hesitations, as in example (2), and incomplete words, as in (3), are assigned the PoS tag `X` which is used for unknown words in UD and is the tag chosen by Dobrovoljc and Nivre (2016) for these phenomena. Pauses, marked by #, are assigned the PoS tag PUNCT.

For the conversion of the `FYLL` relation, we follow Dobrovoljc and Nivre (2016) in mapping directly to the universal relation `discourse`, with the subtype `filler`. Repairs are also straight-forwardly converted to the UD relation `reparandum`. For the analysis of restarts, or deletions as we have called them, we introduce a subtype of the universal `parataxis` relation called `deletion`. Figure 2 shows the converted UD version of the sentence from Figure 1. We observe that the structure differs markedly from the structure in the NDT format. The NDT version in Figure 1 annotates the finite verb *var* 'was' as the root of the segment, whereas the UD version appoints the predicative argument *trafikk* 'traffic' as root with the verb as a dependent with the `cop` (copula) relation type.

6. Availability of the treebank

The treebank will be made available for searching in Glossa (Nøklestad et al., 2017), which is a web-based corpus search interface being developed at the Text Laboratory, University of Oslo. This interface, which currently only supports searching in morphosyntactic information, will be extended with capabilities for searching in syntactic dependency structures as well.

For syntactic search we aim to implement an example-based approach along the lines of the GrETEL system[6], where the user can input an example of the kind of construction they are interested in, have the system analyse the example, select the relevant parts of the analysis (e.g. particular syntactic or morphosyntactic categories, lemmas and/or concrete word forms), and receive a list of all constructions in the treebank that match the given search criteria.

The Universal Dependencies version of the data set has been made available with the v2.1 release of the UD treebanks (Nivre et al., 2017). The treebank annotation continues and future releases will extend the treebank presented in this article with more data from more dialects.

7. Conclusion

In this article we have introduced the LIA treebank of spoken Norwegian dialects. The treebank currently consists of 13,608 tokens taken from three different dialects. We have presented our extended guidelines for morphological and syntactic annotation, as well as the conversion of the treebank to the Universal Dependencies scheme.

[6] `http://gretel.ccl.kuleuven.be/gretel3/`

8. Bibliographical References

Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2003). The Prague Dependency Treebank. In *Treebanks: Building and Using Parsed Corpora*, pages 103–127. Kluwer, Dordrecht, the Netherlands.

Bohnet, B. (2010). Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the International Conference on Computational Linguistics COLING*, pages 89–97, Beijing, China.

de Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal Stanford dependencies. A cross-linguistic typology. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 4585–4592, Reykjavik, Iceland.

Dobrovoljc, K. and Nivre, J. (2016). The Universal Dependencies Treebank of Spoken Slovenian. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 1566–1573, Portorož, Slovenia.

Faarlund, J. T., Lie, S., and Vannebo, K. I. (1997). *Norsk referansegrammatikk*. Universitetsforlaget, Oslo.

Håberg, L. (2010). Den preproprielle artikkelen i norsk.

Hagen, K., Johannessen, J. B., and Nøklestad, A. (2000). A constraint-based tagger for Norwegian. In *17th Scandinavian Conference in Linguistics*, pages 31–48.

Hagen, K., Håberg, L., Olsen, E., and Søfteland, Å. (2017). Transkripsjonsrettleiing for LIA. Technical report, The Text Laboratory, University of Oslo.

Johannessen, J. B. and Jørgensen, F. (2006). Annotating and parsing spoken language. In Peter Juel Henrichsen et al., editor, *Treebanking for Discourse and Speech*, volume 32 of *Copenhagen Studies in Language*, pages 83–104.

Johannessen, J. B., Hagen, K., Lynum, A., and Nøklestad, A., (2012). *OBT+stat: A combined rule-based and statistical tagger*, volume 49, page 51. John Benjamins, Amsterdam, the Netherlands.

Kinn, K., Solberg, P. E., and Eriksen, P. K. (2014). NDT guidelines for morphological and syntactic annotation. Technical report, National Library of Norway, Oslo.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., and Täckström, O. (2013). Universal dependency annotation for multilingual parsing. In *Proceedings of Association for Computational Linguistics (ACL)*, pages 92–97.

Nivre, J., Nilsson, J., and Hall, J. (2006). Talbanken05: A Swedish treebank with phrase structure and dependency annotation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Nivre, J., Agić, Ž., Ahrenberg, L., Antonsen, L., Aranzabe, M. J., Asahara, M., Ateyah, L., Attia, M., Atutxa, A., Augustinus, L., Badmaeva, E., Ballesteros, M., Banerjee, E., Bank, S., Barbu Mititelu, V., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bobicev, V., Börstell, C., Bosco, C., Bouma, G., Bowman, S., Burchardt, A., Candito, M., Caron, G., Cebiroğlu Eryiğit, G., Celano, G. G. A., Cetin, S., Chalub, F., Choi, J., Cinková, S., Çöltekin, Ç., Connor, M., Davidson, E., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dirix, P., Dobrovoljc, K., Dozat, T., Droganova, K., Dwivedi, P., Eli, M., Elkahky, A., Erjavec, T., Farkas, R., Fernandez Alcalde, H., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D., Garcia, M., Gärdenfors, M., Gerdes, K., Ginter, F., Goenaga, I., Gojenola, K., Gökırmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Grūzītis, N., Guillaume, B., Habash, N., Hajič, J., Hajič jr., J., Hà Mỹ, L., Harris, K., Haug, D., Hladká, B., Hlaváčová, J., Hociung, F., Hohle, P., Ion, R., Irimia, E., Jelínek, T., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kayadelen, T., Kettnerová, V., Kirchner, J., Kotsyba, N., Krek, S., Laippala, V., Lambertino, L., Lando, T., Lee, J., Lê H`ông, P., Lenci, A., Lertpradit, S., Leung, H., Li, C. Y., Li, J., Li, K., Ljubešić, N., Loginova, O., Lyashevskaya, O., Lynn, T., Macketanz, V., Makazhanov, A., Mandl, M., Manning, C., Mărănduc, C., Mareček, D., Marheinecke, K., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Mendonça, G., Miekka, N., Missilä, A., Mititelu, C., Miyao, Y., Montemagni, S., More, A., Moreno Romero, L., Mori, S., Moskalevskyi, B., Muischnek, K., Müürisep, K., Nainwani, P., Nedoluzhko, A., Nešpore-Bērzkalne, G., Nguy˜ên Thị, L., Nguy˜ên Thị Minh, H., Nikolaev, V., Nurmi, H., Ojala, S., Osenova, P., Östling, R., Øvrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Pitler, E., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Ramasamy, L., Rama, T., Ravishankar, V., Real, L., Reddy, S., Rehm, G., Rinaldi, L., Rituma, L., Romanenko, M., Rosa, R., Rovati, D., Sagot, B., Saleh, S., Samardžić, T., Sanguinetti, M., Saulīte, B., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shen, M., Shimada, A., Sichinava, D., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Stella, A., Straka, M., Strnadová, J., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Trosterud, T., Trukhina, A., Tsarfaty, R., Tyers, F., Uematsu, S., Urešová, Z., Uria, L., Uszkoreit, H., Vajjala, S., van Niekerk, D., van Noord, G., Varga, V., Villemonte de la Clergerie, E., Vincze, V., Wallin, L., Washington, J. N., Wirén, M., Wong, T.-s., Yu, Z., Žabokrtský, Z., Zeldes, A., Zeman, D., and Zhu, H. (2017). Universal dependencies 2.1. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Nøklestad, A., Hagen, K., Johannessen, J. B., Kosek, M., and Priestley, J. (2017). A modernised version of the glossa corpus search system. In Jörg Tiedemann, editor, *Proceedings of the 21st Nordic Conference on Computa-*

tional Linguistics, pages 251–254.

Östling, R., Börstell, C., Gärdenfors, M., and Wirén, M. (2017). Universal dependencies for swedish sign language. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 303–308.

Øvrelid, L. and Hohle, P. (2016). Universal Dependencies for Norwegian. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Papazian, E. and Helleland, B. (2005). *Norsk talemål. Lokal og sosial variasjon.* Høyskoleforlaget, Oslo.

Petrov, S., Das, D., and McDonald, R. (2012). A universal part-of-speech tagset. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 2089–2096.

Rosa, R., Mašek, J., Mareček, D., Popel, M., Zeman, D., and Žabokrtskỳ, Z. (2014). Hamledt 2.0: Thirty dependency treebanks stanfordized. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2334–2341.

Shriberg, E. (1996). Disfluencies in switchboard. In *Proceedings of the International Conference on Spoken Language Processing*, volume 96, pages 11–14.

Skjærholt, A. (2014). A chance-corrected measure of inter-annotator agreement for syntax. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Solberg, P. E., Skjærholt, A., Øvrelid, L., Hagen, K., and Johannesen, J. B. (2014). The Norwegian Dependency Treebank. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

Surdeanu, M., Johansson, R., Meyers, A., Màrquez, L., and Nivre, J. (2008). The CoNLL 2008 Shared Task on Joint Parsing of Syntactic and Semantic Dependencies. In *Proceedings of the Conference on Natural Language Learning*.

Velldal, E., Øvrelid, L., and Hohle, P. (2017). Joint ud parsing of norwegian bokmål and nynorsk. In Jörg Tiedemann, editor, *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 1–10.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., Kanerva, J., Ojala, S., Missilä, A., Manning, C. D., Schuster, S., Reddy, S., Taji, D., Habash, N., Leung, H., de Marneffe, M.-C., Sanguinetti, M., Simi, M., Kanayama, H., dePaiva, V., Droganova, K., Martínez Alonso, H., Çöltekin, c., Sulubacak, U., Uszkoreit, H., Macketanz, V., Burchardt, A., Harris, K., Marheinecke, K., Rehm, G., Kayadelen, T., Attia, M., Elkahky, A., Yu, Z., Pitler, E., Lertpradit, S., Mandl, M., Kirchner, J., Alcalde, H. F., Strnadová, J., Banerjee, E., Manurung, R., Stella, A., Shimada, A., Kwak, S., Mendonca, G., Lando, T., Nitisaroj, R., and Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, Vancouver, Canada.

Zeman, D. (2008). Reusable tagset conversion using tagset drivers. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*.

ERRATOR: a Tool to Help Detect Annotation Errors in the Universal Dependencies Project

Guillaume Wisniewski
LIMSI, CNRS, Univ. Paris-Sud, Université Paris Saclay, 91 403 Orsay, France
guillaume.wisniewski@limsi.fr

Abstract

Enforcing guidelines compliance is today one of the main challenge faced by the Universal Dependencies project. This work introduces ERRATOR, a set of tools implementing the *annotation variation principle* that can be used to help annotators find and correct errors in the different layers of annotations of UD treebanks. The results of a first annotation campaign that used ERRATOR to correct and harmonize the annotations of the different French corpora are also described.

Keywords: Universal Dependencies, Annotation Coherence, Evaluation

1. Introduction

The Universal Dependencies project (Nivre et al., 2017) aims at developing cross-linguistically consistent treebank annotations for a wide array of languages. Each treebank contains raw text, sentence and word segmentation, PoS tags, dependency relations and in many cases lemmas and morphological features.

In its latest release, the UD project gathers 70 treebanks covering 50 languages. Many of these corpora result from a manual or semi-automatic transformation from existing dependency or constituent treebanks into the UD formalism (see, for instance, (Bosco et al., 2013) or (Lipenkova and Souček, 2014) for a description of such transformations). Because many treebanks have been annotated and/or converted independently by different groups, the risk of incoherence and errors in the application of annotation guidelines is increased. There may indeed be several sources of errors in the produced annotations: in addition to the divergences in the theoretical linguistic principles that governed the design of the original annotation guidelines, errors may also result from automatic (pre-)processes, human post-editing, or human annotation.

As a matter of fact, several works have recently pointed out that different treebanks for the same language are not consistently annotated even though they should follow the same guidelines (Aufrant et al., 2017; Vilares and Gómez-Rodríguez, 2017). More generally, (Wisniewski et al., 2014) has shown that, in spite of common annotation guidelines, the main bottleneck in cross-lingual transfer is the difference in the annotation conventions across corpora and languages.

Enforcing guidelines compliance to improve annotation quality is therefore one of the main challenge faced by the UD project today: as it is, performance achieved on UD corpora, especially in a cross-lingual or cross-corpus setting may be underestimated and this drop may results mainly from divergences in annotations. This work describes ERRATOR, a set of tools implementing the *annotation variation principle* that has been proposed to detect errors in PoS annotations (van Halteren, 2000; Dickinson and Meurers, 2003) and syntactic annotations (Dickinson and Meurers, 2005; Boyd et al., 2008). We propose an extension of this principle to word segmentation making ERRATOR able to help annotators find and correct errors in the different layers of annotations of the UD corpora.

The rest of this paper is organized as follows: we will first explain how the annotation variation principle can be used to identify potential annotation errors (§ 2.). We will then describe our implementation (§ 3.) and the results of a first annotation campaign that used ERRATOR to correct and harmonize the annotations of the different French corpora. ERRATOR is open-source and can be freely downloaded from `https://perso.limsi.fr/wisniews/errator/`.

2. Identifying Potential Annotation Errors

Principle ERRATOR implements the *annotation variation principle* (Boyd et al., 2008) to detect potential annotation errors. This principle states that if two identical sequences of words are annotated in different ways, there is a high chance that one of the annotation is erroneous.

More precisely, we consider that a corpus annotated with PoS, tokenization or dependencies information defines an *alignment* between source sentences and their annotations (in the usual Machine Translation meaning). Following these alignment links, formally defined in the next paragraph, it is possible, given a substring of a source sentence to extract the 'corresponding' part of the annotation. The annotation variation principle can then be implemented as follows:

1. Given one or several corpora of annotated sentences, find all *maximal repeats*, that is to say a substring that occurs at least in two different sentences and cannot be extended to the left or to right to a longer substring. The *maximal repeat problem* can be solved efficiently[1] using Generalized Suffix Tree (Gusfield, 1997).
2. For all repeats, extract their corresponding annotation following alignment links;
3. if not all the annotations corresponding to a single repeat are identical, flag them as a potential annotation error.

[1] If the corpus contain n words, extracting all the maximal repeats takes $\mathcal{O}(n)$ to build the GST and $\mathcal{O}(n)$ to list all the repeats.

These potential errors have then to be reviewed by a human annotator that can eventually correct them.

Aligning source text with their annotation As explained in the previous paragraph, alignment aims at allowing us to 'extract' the part of the annotation that corresponds to a sub-part of a sentence. Aligning tokenized words with PoS or morphological information is straightforward as each word is associated to exactly one label.
For dependencies we represent the dependency tree with a list of pairs (head index, label),[2] that, again, can be mapped directly with the words of the sentence. To test if the dependency annotation of two identical (sub-)sequences of words are the same, we consider that heads outside the sequence are replaced by a special symbol and head indices are replaced by their relative position. Figure 1 shows an example of this representation.[3]
For word segmentation, things are more complicated as, as illustrated in Table 1, the tokenization is not necessarily concatenative: for instance, according to UD guidelines, concatenation and clitics are expanded. It is therefore necessary to define an alignment between the characters of the raw and segmented texts in which some of characters of the raw string can be aligned to more than one characters of the tokenized string.
The alignment is built as follows: we start by looking for the longest common substring (at the character level) between the raw and the tokenized strings and align all their characters. We then remove this substring from both strings and, recursively, re-apply this procedure to the resulting strings as long as there is a common substring. Parts of the annotation that have not been matched are then arbitrarily aligned with the first character of the following match.

①	Souvent␣,␣celui-ci␣l'assume␣seul␣. Souvent␣,␣celui␣-␣ci␣l'␣assume␣seul␣.
②	je␣vais␣au␣jardin je␣vais␣à␣le␣jardin
③	la␣galerie␣des␣batailles␣,␣dans la␣galerie␣de␣les␣batailles␣,␣dans

Table 1: Example of the alignments for word segmentation: groups represented in the same color define 1:1 or 1:n alignments between characters. In examples ② and ③, the segmentation is not concatenative.

3. ERRATOR

In this Section, we describe ERRATOR a set of tools we have developed to automatically detect potential annotation errors and help users to filter out false positive and correct true annotation errors. ERRATOR is freely available from `https://perso.limsi.fr/wisniews/errator/`.
ERRATOR is made of two parts:

- a Python script that implements the method described in Section 2. to extract potential errors from one or several corpora in the CoNLL-U format. This implementation relies on our in-house Generalized Suffix Tree library that can represent an annotated corpus of sentences and efficiently handle all the required operations on strings.
- a web server that can be used to visualize and interacts with these potential errors. This server is developed using the micro web framework `Flask`[4] and relies on both Python and Javascript to generate the web pages and manage interactions with the user.

Figure 2 shows an example of the web interface. This interface allows the user to browse, sort, query potential annotation errors, highlighting the differences in annotations and link them to corresponding sentences in the UD corpora. It is also possible to filter out annotation variations that correspond to truly ambiguous sequence of words either by selecting a particular entry or by writing rules (regular expression that can match either the raw text, the annotations or the two). Annotations can not be corrected directly within ERRATOR so that users can use their favorite 'external' tools for annotating treebanks.

repeat length (words)	# repeats
2	71,349
3	65,764
4	31,274
5	12,053
≤ 5	206,296
≤ 10	+7,058
≤ 20	+359
≤ 84	+83

Table 2: Distribution of the match length in the French treebanks of the UD project.

4. Analyzing Annotation Errors in the French Corpora

In this section, we will present the results of our first experiments with ERRATOR to correct the errors on the different French treebanks of the Universal Dependencies project.
There are 5 treebanks for French in the UD project, representing a total of 40,101 sentences and 1,099,571 tokens. Considering all corpora, 213,796 sequence of two words or more that appear in more than one sentence, the longest repeat containing 84 words, and 7,000 repeats being made of more than 10 words. Table 2 gives some statistics on match length.
Table 3 shows the number of matches for which the annotation is not the same and are therefore considered as a potential error. Examples of the output produced by ERRATOR for the PoS annotation of the French `Sequoia` treebank are given in Figure 4.
To evaluate the impact of annotation incoherences on prediction performance, we have manually checked all the

[2] Formally, for a sentence of n words, the parse tree can be represented by a list `l` of n pairs so that `l[i]` is the index of the head of the i-th word of the sentence and the label of the dependence

[3] Considering alternative representations (e.g. by dropping labels or considering PoS rather than words) is left for future work.

[4] `flask.pocoo.org`

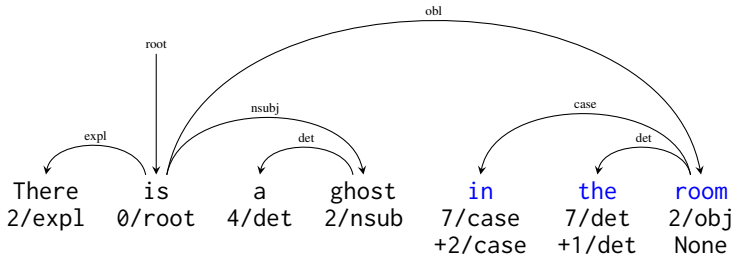


Figure 1: Representation of a dependency tree as a list of heads (first line) and representation of its substring 'on the issue' with local indices used when comparing to tree fragments (second line).



Figure 2: ERRATOR User Interface that allows users to browse, query and filter potential annotation errors.

annotation	number of potential errors
word segmentation	2,617
PoS	25,527
Dependencies	48,986

Table 3: Number of potential errors in the French treebanks of the UD project.

matches between the French UD and the French FTB correcting incoherences and errors.

The impact of these modifications on the performance of a PoS tagger are reported in Table 5. These result show, in particular, that, for PoS tagging, part of the drop in performance observed when testing on out-domain data is due to divergences in annotations, as shown by the comparison on the results in bold.

5. Conclusion

We have introduced ERRATOR, a set of tools implementing the *annotation variation principle* that can be used to help annotators find and correct errors in the different layers of annotations of UD treebanks. Our experience with ERRATOR to correct and harmonize the annotations of the different French corpora show that has proved the usefulness of this tool. ERRATOR could, however, be more efficient if it is tightly integrated with annotation tools to detect potential errors as soon as possible during the annotation campagn. This will be the goal of our future developments.

Acknowledgments

This work has been partly funded by the *Agence Nationale de la Recherche* (ParSiTi project, ANR-16-CE33-0021).

6. Bibliographical References

Aufrant, L., Wisniewski, G., and Yvon, F. (2017). Limsi@conll'17: Ud shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 163–173, Vancouver, Canada, August. Association for Computational Linguistics.

emea-fr-dev_00104	Pour les patients atteints d' un SCA , la dose initiale recommandée est un bolus intraveineux de 0,1 mg / kg suivi d' une perfusion de 0,25 mg / kg / h . ADP DET NOUN VERB ADP DET NOUN PUNCT DET NOUN ADJ VERB AUX DET NOUN ADJ ADP NUM NOUN ADP NOUN VERB ADP DET NOUN ADP NUM NOUN ADP NOUN ADP NOUN PUNCT
emea-fr-dev_00510	Pour les patients atteints d' un SCA , la dose initiale recommandée est : ADP DET NOUN VERB ADP DET NOUN PUNCT DET NOUN ADJ VERB VERB PUNCT
frwiki_50.1000_00767	- Michel Roussin , reconnu coupable de complicité et recel de corruption , a été condamné à 4 ans de prison avec sursis et une amende de 50 000 euros . PUNCT PROPN PROPN PUNCT AUX NOUN ADP NOUN CCONJ NOUN ADP NOUN PUNCT AUX AUX VERB ADP NUM NOUN ADP NOUN ADP NOUN CCONJ DET NOUN ADP DET NOUN PUNCT
frwiki_50.1000_00773	- Louise-Yvonne Casetta , trésorière occulte de le RPR , a été reconnue coupable de complicité et recel de corruption , et a été condamnée à 20 mois de prison avec sursis et 10 000 euros d' amende . PUNCT PROPN PROPN PUNCT NOUN ADJ ADP DET PROPN PUNCT AUX AUX AUX ADJ ADP NOUN CCONJ NOUN ADP NOUN PUNCT CCONJ AUX AUX VERB ADP NUM NOUN ADP NOUN ADP NOUN CCONJ DET NOUN ADP NOUN PUNCT
emea-fr-dev_00228	Les saignements majeurs se sont produits le plus fréquemment à le site de ponction (voir Tableau 3) . DET NOUN ADJ PRON AUX VERB DET ADV ADV ADP DET NOUN ADP NOUN PUNCT VERB PROPN NUM PUNCT PUNCT
emea-fr-dev_00258	Les saignements majeurs se sont produits le plus fréquemment à le site de ponction (voir Tableau 5) . DET NOUN ADJ PRON AUX VERB DET ADV ADV ADP DET NOUN ADP NOUN PUNCT VERB NOUN NUM PUNCT PUNCT

Table 4: Example of errors in PoS annotation found by ERRATOR on the French Sequoia corpus of the UD project. Words in blue and red appears in two different sentence, but the red part has not the same annotation.

Bosco, C., Montemagni, S., and Simi, M. (2013). Converting italian treebanks: Towards an italian stanford dependency treebank. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 61–69, Sofia, Bulgaria, August. Association for Computational Linguistics.

Boyd, A., Dickinson, M., and Meurers, W. D. (2008). On detecting errors in dependency treebanks. *Research on Language and Computation*, 6(2):113–137, Oct.

Dickinson, M. and Meurers, W. D. (2003). Detecting errors in part-of-speech annotation. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, pages 107–114, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dickinson, M. and Meurers, W. D. (2005). Detecting errors in discontinuous structural annotation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 322–329, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Gusfield, D. (1997). *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, New York, NY, USA.

Lipenkova, J. and Souček, M. (2014). Converting russian dependency treebank to stanford typed dependencies representation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 143–147, Gothenburg, Sweden, April. Association for Computational Linguistics.

van Halteren, H. (2000). The detection of inconsistency in manually tagged text. In *Proceedings of the COLING-2000 Workshop on Linguistically Interpreted Corpora*, pages 48–55, Centre Universitaire, Luxembourg, August. International Committee on Computational Linguistics.

Vilares, D. and Gómez-Rodríguez, C. (2017). A non-projective greedy dependency parser with bidirectional lstms. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 152–162, Vancouver, Canada, August. Association for Computational Linguistics.

Wisniewski, G., Pécheux, N., Gahbiche-Braham, S., and Yvon, F. (2014). Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1779–1785, Doha, Qatar, October. Association for Computational Linguistics.

7. Language Resource References

Nivre, J., Agić, Ž., Ahrenberg, L., et al. (2017). Universal Dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, Prague.

train	test	% error
FTB corrected	FTB corrected	2.96%
	FTB original	4.06%
	UD corrected	6.49%
	UD original	6.70%
UD original	UD original	4.51%
	UD corrected	4.53%
	FTB corrected	5.93%
	FTB original	**6.78%**
FTB original	FTB original	3.17%
	FTB corrected	3.88%
	UD corrected	6.91%
	UD original	7.01%
UD corrected	UD corrected	4.42%
	UD original	4.60%
	FTB corrected	**5.71%**
	FTB original	6.75%

Table 5: Comparison of the performance of a state-of-the-art PoS tagger on two different French dataset before and after the errors detected by ERRATOR have been corrected.

SandhiKosh: A Benchmark Corpus for Evaluating Sanskrit Sandhi Tools

Shubham Bhardwaj, Neelamadhav Gantayat, Nikhil Chaturvedi, Rahul Garg, Sumeet Agarwal
Indian Institute of Technology, IBM Research
New Delhi, INDIA
shubhamiitd.007@gmail.com, neelamadhavg@in.ibm.com, 709nikhil@gmail.com,
rahulgarg@cse.iitd.ac.in, sumeet@ee.iitd.ac.in

Abstract

Sanskrit is an ancient Indian language. Several important texts which are of interest to people all over the world today were written in Sanskrit. The Sanskrit grammar has a precise and complete specification given in the text Aṣṭādhyāyī by Pāṇini. This has led to the development of a number of *Sanskrit Computational Linguistics* tools for processing and analyzing Sanskrit texts. Unfortunately, there has been no effort to standardize and critically validate these tools. In this paper, we develop a Sanskrit benchmark called SandhiKosh to evaluate the completeness and accuracy of Sanskrit Sandhi tools. We present the results of this benchmark on three most prominent Sanskrit tools and demonstrate that these tools have substantial scope for improvement. This benchmark will be freely available to researchers worldwide and we hope it will help everyone working in this area evaluate and validate their tools.

Keywords: Sanskrit, Sandhi, Morphophonology

1. Introduction

On the 11th of December, 2014, the United Nations General Assembly unanimously passed a resolution declaring the summer solstice, June 21st, as the International Day of Yoga (UN, 2014). The resolution recognizes that "yoga provides a holistic approach to health and well-being" and that "a wider dissemination of information about the benefits of practicing yoga would be beneficial for the health of the world population". The popularity of Yoga has been growing steadily all over the world primarily due to its potential health and healing benefits in ailments such as depression (Uebelacker et al., 2010), cardiovascular disorders (Raub, 2002) and other chronic diseases (Yang, 2007).

With the growing worldwide popularity of this ancient discipline, there is also a growing interest to understand and practice this discipline in its pure and unadulterated form. All of the classic texts of this discipline such as Gherand Samhita (Saraswati, 2013), Hath-Yoga-Pradiptika (Singh, 1915), Yoga-Vashistha (Mitra, 1891), Patanjali-Yoga-Sutra (Prasada, 1998) and Bhagavad-gītā (Swarupananda, 2016) were composed in the Sanskrit language. While translations of some of these texts are available in other languages, for a serious Yoga practitioner, reading and understanding these texts directly in their native language has a lot of value. Unfortunately, Sanskrit is now spoken by only a small number of people. These classical texts, although available, remain inaccessible to most of the world.

Due to this and some more reasons, there is growing interest in learning the Sanskrit language (HT, 2007; Ghosh, 2015). For example, the St. James Junior school in London has introduced Sanskrit language in the junior school because the "knowledge of grammar ultimately gives the pupils a greater clarity and accuracy in thinking, reading and speaking" (SJJS, 2017). Fortunately, the Sanskrit language has undergone very little modification and by learning this language, it is possible to read and understand most of the classical texts in Sanskrit, including those which date back to centuries BC.

One of the major distinguishing features of the Sanskrit language is an accurate specification of its grammar rules. The authoritative work on the Sanskrit grammar is by Pāṇini in the form of Aṣṭādhyāyī, meaning a collection of eight books (Pāṇini and Katre, 1987), which comprises a total of 3,959 *sūtras* (concise rules). Aṣṭādhyāyī gives an almost complete specification of the Sanskrit grammar. Due to the precise specification of these rules, the language has not undergone much modification. The rules of the language, though small in number, are precise and rich in their structure which allow the users of the language tremendous flexibility in their usage of the language. The verb-roots, which are less than 2000 in number, can be combined and modified according to the well defined rules to form new words, making a rich lexicon of size limited only by the creativity of the writer.

The precise and complete specification of the Sanskrit grammar also opens up the possibility of the development of computational tools to assist the students of the language or general readers to help translate and interpret Sanskrit texts. This has spawned a new and active interdisciplinary area of research called *Sanskrit Computational Linguistics*, which aims to use computational tools for automating the analysis of Sanskrit texts (Huet, 2003; Huet, 2009; Huet, 2005; Kumar et al., 2010; Bharati and Kulkarni, 2007; Goyal et al., 2009; Kapp and Malten, 1997; Goyal and Huet, 2013; Kulkarni, 2017; Jha, 2017b; ILTP, 2012; GM, 2017; Omkarananda, 2003; UBC, 2017). Due to the availability of a complete formal specification of the Sanskrit grammar, the development of a perfect Sanskrit parser seems to be the guiding factor behind this research. This necessitates the development of sub-tools like dictionaries, morphological analyzers, Sandhi splitters, and de-compounders without which successful Sanskrit parsing cannot be done.

The Cologne Sanskrit Dictionary Project (Kapp and Malten, 1997) aims to digitize the major bilingual Sanskrit dictionaries and provide easy access to the meanings of all the Sanskrit words which may be used by computer programs that help analyze Sanskrit texts. The most prominent among the remaining tools are (a) the Sanskrit Reader Companion (Goyal and Huet, 2013) by Inria which has

tools for declension, conjugation, Sandhi splitting and merging along with word stemming (b) the Saṃsādhanī - A Sanskrit Computational Toolkit by University of Hyderabad (Kulkarni, 2017), which has tools for morphological analysis and generation, Sandhi splitting and merging, declension, conjugation and other form of word modifications and (c) the Sanskrit language processing tools developed at the Jawaharlal Nehru University (JNU) (Jha, 2017b), which comprise tools for Sandhi splitting and merging, declension, conjugation, POS tagger and other forms of word modifications.

Although researchers have been working in the field of Sanskrit Computational Linguistics for many years, there has been no effort to standardize, validate or critically evaluate the outcome of the work done so far. In this paper, we take a small step in the direction of standardizing and validating the research in this field. We examine the process by which two or more words combine to form a new word, a process known as *Sandhi*. The process of Sandhi is fundamental to the Sanskrit language as it enables the formation of new and more complex words using simpler words. Any computational tool for processing Sanskrit needs to be able to merge and split the words according to the rules of Sandhi. The correctness of any such tool critically depends on the correctness of its Sandhi processing.

We create a benchmark corpus called *SandhiKosh* that may be used by researchers to evaluate and validate the correctness and accuracy of their Sanskrit Sandhi and Sandhi-splitting tools. This corpus consists of examples of words along with their correct Sandhi-splitted root words. These examples have been categorized into the following five subcorpora: (a) a list of 282 examples based on the Aṣṭādhyāyī rules; (b) a list of 150 examples hand picked from eleven well-known Sanskrit texts; (c) a list of 1432 examples taken from the most famous Sanskrit text Bhagavad-gītā; (d) a list of 10107 examples taken from digitized Sanskrit text at University of Hyderabad (Kulkarni, 2017); and (e) a list of 2700 examples taken directly from the Aṣṭādhyāyī text, which itself has been written in Sanskrit. Some of the examples were hand picked and manually verified for correctness while the other examples were created using the existing computational tools and validated computationally using a variety of methods.

We evaluated the three major Sanskrit Sandhi tools ((Jha, 2017a), (Kulkarni, 2017), and (Goyal and Huet, 2013)) using our SandhiKosh benchmark. Our results indicate the all these tools can benefit substantially from SandhiKosh. The SandhiKosh benchmark corpus will be freely available to researchers working in this area and we hope that it will lead to significant improvement in the state-of-the-art in the field of Sanskrit Computational Linguistics.

In Section 2. we describe the process of Sandhi in a little more detail and in Section 3. we describe the three major Sanskrit Computational Linguistics tools. The methodology followed during the creation of SandhiKosh has been described in Section 4. and the evaluation results are presented in Section 5.. We discuss possible future improvements in the Sanskrit Sandhi tools and in the SandhiKosh benchmark in Section 6. and conclude in Section 7..

2. Introduction to Sandhi in Sanskrit

Word formation in Sanskrit is centered around a root verb, modified by a suffix (and additionally a prefix in certain cases). Each of these three (roots, prefixes, and suffixes) represents a morpheme category, as these are the meaningful morphological units of the language and none of them can be further divided. Further, Sanskrit texts contain numerous words which are formed by the combination of two or more words. This process, generally known as *Sandhi*, takes place according to certain rules codified by the grammarian Pāṇini in his Aṣṭādhyāyī. The reverse process of getting back the component morphemes/words from the Sandhied words is known as *Sandhi Viccheda* or Sandhi splitting.

Interestingly, each of the two words *Sandhi* and *Viccheda* is itself made up of two components – *Sam* + *dhi* and *vi* + *cheda* respectively. *Sam* (meaning together) and *dhi* (meaning placement/location) combine to give *Sandhi* which means 'placed together/joined/merged'. *Vi* (meaning special) and *cheda* (meaning split/ breaking down) combine to give *Viccheda* which means special splitting (as opposed to simply splitting a word into each of its component letters).

The Sandhi process is akin to that in some other languages, such as in English, *"come"* + *"-ing"* → *"coming"*, where we lose the additional "*e*" in the word *"come"* while merging. Another category of examples includes words such as "indirect", "impossible", and "irrelevant", where all these words have the same prefix as "in-", however, that got modified when merging with the root word. However, there is a very important difference between Sandhi in Sanskrit and such a combination process in English, as explained later in this section.

2.1. Conditions for Sandhi

Interestingly, the word *Sandhi* does not appear in any of the Aṣṭādhyāyī *sūtras* (concise rules). There are certain *sūtras* that are governed by a condition known as *saṃhitā* which as defined in *sūtra* 109 of Chapter 4 of Book 1, means "closest proximity of letters". These *sūtras* talk about the changes that take place when two letters are in "closest proximity". *Sūtras* 73 to 157 of Chapter 1 of Book 6 and all *sūtras* of Chapters 3 and 4 of Book 8 of Aṣṭādhyāyī are governed by the condition of *saṃhitā*. These rules are hereafter referred to as Sandhi rules. Thus, Sandhi is an umbrella term that is used to refer to sound changes that take place when two sounds are close enough.

The sound changes can take place in a variety of ways, depending not always only the two sounds (represented by the last character of the first word and the first character of the second word) combining but also sometimes on other factors, as described by Pāṇini in Aṣṭādhyāyī. The two sounds may merge to give a single sound, one of the two sounds (the former or the latter) may get changed/reduplicated before combining with the other, or even get elided. A new sound may also come in between.

2.2. Types of Sandhi

Sandhi can take place either within a word (internal Sandhi) or between two or more words (external Sandhi). Also, depending on whether the two letters that are being combined

Criteria	Type	Explanation	Example	Analogy with English
Position	Internal	Root + Pref/Suffix	*bho + anam → bhavanam*	*come + ing → coming*
	External	Words combine	*tau + ekadā → tāvekadā*	*modify + ability → modifiability*
Type of character	Vowel	Vowel + Vowel	*hima + ālayaḥ → himālayaḥ*	*forgive + able → forgivable*
	Visarga	Visarga first	*punaḥ + janma → punarjanma*	No visarga in English
	Consonant	Other cases	*vṛkṣa + chāyā → vṛkṣacchāyā*	*forget + able → forgettable*

Table 1: Different types of Sandhi classification

are vowels, consonants or the first of them is a visarga[1], the Sandhi is classified as vowel, consonant or visarga Sandhi. The classification and the examples thereof have been summarized in Table 1. Similar examples from English language, wherever applicable, have also been provided.

2.3. Importance of Sandhi Splitting

Sandhi is very frequently encountered in classical texts of Sanskrit and these texts cannot be understood as long as the complex words (particularly the ones involving external Sandhi) are not broken down. There is an important difference between combination of words in English and that in Sanskrit. In English, combination of words is restricted by meaning and parts of speech involved. Thus, for example, in the sentence 'The regrettable decision of the chairman is now causing great harm to him', each of the words 'regrettable' , 'chairman' and 'causing' represents a combination, but combinations like 'Theregrettable' or 'regrettabledecision' or 'isnow' or worse 'Theregrettabledecisionofthechairmanisnowcausinggreatharmtohim' are simply not allowed. On the other hand, in Sanskrit, these are not only allowed but encountered very frequently. So while no combination is possible between the words of the sentence 'Ravi arrived in forest', all the words in the Sanskrit equivalent '*Raviḥ vane āgataḥ* ' can combine to form '*Ravirvanayāgataḥ*' (please note the changes at the boundaries of merging). Thus, Sandhi splitting is not only helpful but indispensable in the analysis of classical Sanskrit texts.

3. Existing Sandhi Tools

Over recent years a considerable amount of research has been carried out in the field of Sanskrit Computational Linguistics. A number of tools have been developed in this domain. As mentioned earlier, the development of a perfect Sanskrit parser seems to be the guiding factor behind this research, but this by itself necessitates the development of sub-tools like morphological analyzers, Sandhi splitters, de-compounders among others, without which successful parsing cannot be done.

In this section, we present the three most popular publicly available set of Sandhi and Sandhi Splitting tools. Given two words, the Sandhi process occurs as per the rules mentioned in the relevant sections of Aṣṭādhyāyī but there are multiple approaches to Sandhi splitting to get back the original words and we discuss the techniques used by these tools for the same. Comparison of these tools using the benchmark data set is provided in Section 5..

3.1. JNU Tools

The JNU Sandhi tool, known as the *Sanskrit Sandhi Generator*, was developed under the supervision of Prof. Girish Nath Jha. The corresponding Sandhi splitting tool, known as *Sanskrit Sandhi Recognizer and Analyzer* (Jha, 2017a) is specifically designed for vowel based Sandhi (Sachin, 2007). Using a dictionary of possible morphemes, this tool, at every location recursively checks for binary splits. To be a valid split, both the left and right split segments must be available in the dictionary. If the second segment has more than one sound marked for Sandhi, then only the first segment is matched against the dictionary.

3.2. UoH Sandhi Tools

These tools were developed at the Department of Sanskrit Studies, University of Hyderabad (UoH) under the guidance of Prof. Amba Kulkarni (Kulkarni, 2017). The Sandhi Splitting tool in this case also recursively breaks a word at every possible position and applies appropriate Sandhi rules to generate possible morpheme candidates and passes them through a morphological analyzer. The split words are considered as valid only if all its constituents are recognized by the morphological analyzer. Weights are assigned to the accepted candidates and then ranked based on the descending order of weights.

3.3. INRIA Tools

The Sandhi tool, known as *The Sandhi Engine*, was developed under the guidance of Prof. Gerard Huet at INRIA, France (Goyal and Huet, 2013). Of the three sandhi tools discussed here, this is the only tool which makes an explicit distinction between internal and external sandhi, giving the user both options to choose from. The external one corresponds to doing external sandhi in a deterministic fashion, with the most frequent rule, not taking into account optional rules. This is different from the UoH sandhi tool, that returns all possible solutions.The internal one is a rather ad-hoc processing, also deterministic, but corresponding more or less to INRIA's case generation with retroflexion. The other tool, called *The Sanskrit Reader Companion*, is actually more than a Sandhi splitting tool. It is designed to help a novice Sanskrit reader parse complex Sanskrit sentences. Sandhi splitting is only one part of the analysis. Initially, the word is analyzed to gather stems and their morphological parameters, such as permitted genders of nomi-

[1]According to the Merriam-Webster dictionary Visarga refers to a "Sanskrit postvocalic sound or group of sounds produced by keeping the vocal organs above the glottis in the same position as for the preceding vowel and continuing to expel air from the lungs but not vibrating the vocal cords."

nal stems, allowed classes, and attested pre-verbs for roots. In the next stage, another round of stem generation is performed considering the various tenses, moods, absolutives, and participles in 10 varieties. Finally, inflexional morphology paradigms derive the inflected forms according to the morphological parameters, some of which are read from the word itself, while the others are defined in specific tables.

4. Creation of SandhiKosh

The SandhiKosh comprises of five sub-corpora that provide for a complete coverage of all the Sandhi rules of Aṣṭādhyāyī while at the same time are designed to give a good estimate of the accuracy of the Sandhi tools when applied to real Sanskrit texts.

4.1. Rule-based Corpus

The rule-based corpus is designed for checking for accuracy and completeness of the existing Sanskrit Sandhi tools. For this, all the rules of Aṣṭādhyāyī related to Sandhi were identified and unique examples corresponding to each of the rules were added to this corpus. Since some of the tools implement only a particular type of Sandhi, two separate datasets were created - one for internal Sandhi rules and the other for external Sandhi rules. At least one example for each rule in each of the two datasets was provided. If a rule applies to both internal and external Sandhi, an example was included in both corpora. This resulted in 150 examples for internal Sandhi and 132 examples for external Sandhi, with a total of 282 examples.

4.2. Literature Corpus

Some of the Sandhi rules are very common in Sanskrit texts while some other rules are rare. Therefore, although the rule-based corpus can give a good estimate of the completeness and validity of the Sandhi tools, it cannot estimate the accuracy of the tools when applied to real-world Sanskrit texts. In order to estimate the accuracy of Sandhi tools on classical and contemporary Sanskrit texts, 150 examples from a total of 11 different literary texts were handpicked to constitute the literature sub-corpus.

4.3. *Bhagavad-gītā* Corpus

In order to estimate the performance of Sandhi tools on some of the old Sanskrit classical texts, a corpus based on the *Bhagavad-gītā* was created. The *Bhagavad-gītā* is the best known and the most widely read and translated book from Sanskrit literature (Davis, 2014). It is organized into eighteen chapters comprising 700 verses. All the verses of its first nine chapters were critically analyzed and all the cases involving external Sandhi were split manually into their constituents leading to creation of a sub-corpus with 1430 examples.

4.4. UoH Corpus

The corpora described so far were manually created and therefore are comparatively small in size. These corpora may not estimate the performance of Sandhi tools accurately due to their small sizes and the particular examples and texts they are based on. A Sanskrit text with a different literary style may have very different statistical properties and give different performance when these Sandhi tools are applied. It was therefore very important to include a large corpus in the benchmark.

The University of Hyderabad (UoH) has digitized a large number of Sanskrit texts and made it freely available for researchers and general users (Kulkarni, 2017). For 39 of these texts, all the complex words are split according to the Aṣṭādhyāyī rules into smaller constituents and made available. This gives $113,913$ Sandhi splitting examples. While trying to make use of this corpus, we discovered several errors. These errors were filtered using a combination of the Cologne dictionaries (Kapp and Malten, 1997) (details omitted) leading to a list containing $9,368$ examples.

4.5. *Aṣṭādhyāyī* Corpus

Aṣṭādhyāyī is also written in Sanskrit and its split of words is also available at (SD, 2017). This was found to be another good source of Sandhi examples. However, even this source suffered from the limitation of insufficient splits. Moreover, a very significant number of splits were not located in any dictionary because of the way this text has been composed. Since the fundamental challenge is the insufficiency of splits, the splits which can undergo further splitting themselves are likely to be of greater length than fundamental morphemes. Thus using the length of the split words as a heuristic, a total of $3,959$ examples are reduced to $2,700$ where further splitting is applicable. Also, the results were noted for different values of the word lengths – 10, 20, 30, 40 and 50 (results omitted).

5. Evaluation Results

The results of evaluation of the Sandhi and the Sandhi Splitting tools on the corpora described above are presented in this section. These results hold true as of February 22, 2018 and may change in the future as and when the tools get updated.

5.1. Evaluation Methodology

We used a python-based automated evaluation method to automatically send web-requests to each of the tools, parse the HTML output and extract the relevant information automatically. The *requests* module of python was used to fetch the web page from source tool. The *BeautifulSoup* package was used to parse each of the web pages. Some of the tools generate multiple outputs for a single input along with a *filtered* output that is most likely to be correct. For all the three tools, we considered all the results instead of just the filtered output and marked the output to be correct if any of the results matched with the expected output of SandhiKosh. In case of Sandhi, several examples in the SandhiKosh corpus contain more than two words to be joined. However, all the three tools have provision for joining only two words at a time. For the examples where there were more than two words to be joined, we iteratively obtained the results from these tools to form the final Sandhi word.

5.2. Accuracy of Automated Evaluation

The automated evaluation methodology evaluated the accuracy of the tools by automatically extracting the results

Corpus	Words	JNU	UoH	INRIA
Rule based - Internal	150	21 (14.0%)	36 (24.0%)	79 (52.7%)
Rule based - External	132	38 (28.8%)	57 (29.5%)	67 (50.8%)
Literature	150	53 (35.3%)	130 (86.7%)	128 (85.3%)
Bhagavad-gītā	1430	338 (23.64%)	1045 (73.1%)	1184 (82.1%)
UoH	9368	3506 (37.4 %)	7480 (79.8%)	7655 (81.7%)
Āstaadhyaayi	2700	455 (16.9%)	1752 (64.9%)	1762 (65.2%)

Table 2: Sandhi accuracy obtained by the three different Sandhi Tools available in the literature.

Corpus	Words	JNU	UoH	INRIA
Rule based - Internal	150	10 (6.8%)	27 (18.0%)	3 (2.0%)
Rule based - External	132	22 (16.9%)	48 (36.4%)	38 (29.2%)
Literature	150	13 (8.7%)	98 (65.3%)	101 (67.3%)
Bhagavad-gītā	1430	67 (4.9%)	650 (45.5%)	962 (70.8%)
UoH	9368	934 (10.0%)	6393 (68.2%)	6490 (69.3%)
Āstaadhyaayi	2700	18 (0.7%)	263 (9.7%)	510 (18.9%)

Table 3: Sandhi splitting accuracy obtained by the three different splitting tools available in the literature.

from the HTML output of the web-pages and then applying an exact string match to assessing the correctness of the results. This process of automatic evaluation may lead to reporting slightly higher error rate as compared to a manual evaulation process, due to differences in punctuation, spacing or other minor errors in the Sandhi process such as omission of *visarga* etc. To assess this more systematically, we took a small corpus and evaluated it manually on all of the three tools and then compared the results to automated evaluation. It was found that for the UoH and JNI tools, the difference between automated and manual evaluation was small with manual evaluation reporting only marginally higher accuracy than the automated evaluation. However, in case of the INRIA tool, the difference was found to be slightly more but always less than 23%.

5.3. Sandhi Tools

The performance of Sandhi tools on SandhiKosh is presented in Table 2. The INRIA tool provides an option to select internal or external Sandhi that needs to be applied while creating the joined word. Since the SandhiKosh does not have information about internal or external Sandhi (except for the rule based corpus), we evaluated the entire corpus with both the options on the INRIA tool. If any of the option gives the correct word merging, the Sandhi is marked as correct. Thus, the results are presented by combining the results of both internal and external Sandhi options in the most optimistic manner.

For the rule-based (internal & external) corpus INRIA is the best performing tool at 51.8% accuracy followed by UoH and JNU Sandhi tools. On the Literature corpus, the accuracy of INRIA and UoH tool is comparable whereas that of JNU is substantially lower at 53%. These trends are consistent for rest of the corpora as well with INRIA and UoH tool performing at similar levels of accuracy (ranging from 23 to 65%).

5.4. Sandhi Splitting Tools

For Sandhi splitting, the INRIA tool as well as the UoH tool gives multiple possible splits of a given word. For the evaluation purposes, we examine all the possibilities given by these tools and mark the Sandhi splitting as correct if the correct split has been given as one of the options. In this way, we combine the multiple options given by these tools in the most optimistic manner.

The accuracy of Sandhi splitting tools on SandhiKosh is presented in Table 3. On the rule-based (internal & external) corpus, the UoH tool performs at 26.6% accuracy whereas the INRIA and JNU tools perform respectively at 14.5% and 11.4%. The JNU tool performs the worst for all the corpora, whereas INRIA tool performs best on Bhagavad-gītā and Aṣṭādhyāyī corpora and UoH tool performs best on the Literature and UoH corpora.

5.5. Sandhi type based performance

The Sandhi splitting and merging performance is evaluated on three different types of Sandhi – vowels, consonants, and visarga Sandhi types and the results are shown in Figure 1 (for all the corpora put together).

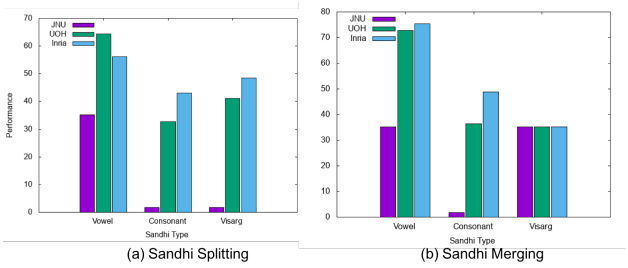


Figure 1: Performance of different splitting tools on the combined corpora for three Sandhi types.

The JNU tools which presently support Vowel Sandhi only are barely able to give any correct results on Consonant and Visarga Sandhi types as expected. However, on the Vowel Sandhi type, the accuracy of the JNU tools is less than that of the UoH and INRIA tools for Sandhi splitting but comparable in the case of Sandhi merging.

6. Discussion

We draw attention to some of the reasons as to why the Sandhi splitting tools are not able to get the correct splits.

- **Rules not implemented:** Some of the rules have not been implemented by one or more of the three splitters. For example, none of the three splitters is able to do the following split: *sa yogī* → *saḥ* + *yogī*.

  The *sūtra* which applies in this case is 6.1.132 (*sūtra* 132 in first chapter of book 6 of Aṣṭādhyāyī) as per which the visarga *ḥ* of the words *saḥ* and *eṣaḥ* is elided when they combine with a word beginning with a consonant. Thus, sandhi-splitting in this case involves restoring the visarga in both the cases when the forms *sa* and *eṣa* are encountered before a word beginning with a consonant. As mentioned earlier, none of the splitters has implemented this string-match condition.

  There are other *sūtras* similar to this, in which the sandhi phenomenon is not restricted to a focus only on two letters but also extends to the entire word/string under consideration.

- **Optional rules:** There are some rules which are optional in nature and less frequently used. The following case is an example where none of the three splitters is able to detect the correct split: *kumāri atra* → *kumārī* + *atra*. This is because *kumārī* + *atra* generally leads to *kumāryatra* through the *sūtra* 6.1.77 but the application of another *sūtra* 6.1.127 (which is less frequently applied) leads to the optional result *kumāri atra*. Getting back the correct split would require the reverse application of this rule.

- **Cascading effect:** There are some rules in which the result of combination of two letters may create the possibility of another sandhi, when a suitable context exists. A change may also occur far beyond the merging boundaries of the two words. For example, in *ṣaṭ* + *navatiḥ* → *ṣaṇṇavatiḥ*, the *n* of *navatiḥ* changes to *ṇ*, which then causes the change of *ṭ* of *ṣaṭ* to *ṇ* , thus resulting into *ṣaṇṇavatiḥ*. Thus, to get back the original words, both of these changes will have to be undone.

- **Validation problem:** The process of Sandhi splitting involves splitting at different potential locations, and validating the splits to check which one of them is correct. If the set used for validation is not complete, even correct splits may sometimes not be validated. For example, in *a* + *chedyaḥ* → *acchedyaḥ*, none of the three splitters performed correctly in the beginning because *a* may not have been validated as a proper split. However, the INRIA Sanskrit Reader Companion was updated recently to take care of this condition, when this was brought to the notice of the tool developers.

- **Compounding effect:** The process of compounding, due to which words come together without there necessarily being a change when they merge, also creates problems. While the UoH and the INRIA tools do have the provision of decompounding along with Sandhi splitting, the JNU splitter does not have a way to do both together. For example, in *lakṣyasyārthatvavyavahārānurodhena* → *lakṣyasya* + *arthatvavyavahāra* + *anurodhena* the second split is not validated without decompounding, and thus even though, only vowel Sandhis are involved, the JNU splitter is not able to correctly split the word.

It is to be noted that our benchmarking corpus SandhiKosh is expected to undergo refinements in the future as there is scope for its improvement. In subsequent releases of the corpus, we aim to deal with the optional rule scenarios by presenting all valid splits and/or merged words for a given corpus entry. We also plan to provide a trace of the Aṣṭādhyāyī rules applied in the process of Sandhi for each example. Along with these, we also plan to propose standardizations for word spacing, the usage of avagraha and the presence of anuswara to make the evaluation process more rigorous and extensive.

7. Conclusion

Standardization of benchmarks has a profound impact on the corresponding field. The benchmarks shape a field by giving an objective yardstick against which different researchers strive to improve their performance, thereby leading to a faster overall development. Benchmarks such as top 500 supercomputers (Dongarra et al., 1997) and SPEC (Henning, 2006) in the area of computing have shaped how computers were designed and built.

In this paper we have attempted to design the first benchmark called SandhiKosh in the area of Sanskrit Computational Linguistics. SandhiKosh provides a way to measure the accuracy and completeness of Sanskrit Sandhi tools. The examples in SandhiKosh were selected usign five different methods in order to provide a corpus that is complete as well as able to reflect the performance of Sandhi tools on actual Sanskrit literature.

We evaluated the performance on three available Sanskrit Sandhi tools on SandhiKosh. Our results indicate that the best performing Sandhi merging tools give an accuracy of in the range of 50-60% where as the worst performing tools result in 20-30% accuracy. For Sandhi splitting, which is a harder problem, the best tools give an accuracy of 50-60% where as the worst performing tools give an accuracy between 5-15%.

SandhiKosh will be freely available to researchers and we hope that it will lead to faster overall progress in the area of Sanskrit Computational Linguistics.

8. Acknowledgements

We thank Prof. Amba Kulkarni (University of Hyderabad), Prof. Gerard Huet (INRIA, France) and Prof. Girish Nath Jha (Jawaharlal Nehru University, Delhi), for providing insightful comments and useful improvements to our benchmark corpus. We would also like to thank Samskrita Bharati and its teachers such as Pt. Narayan Dutt Suvedi and Dr. Sachin Kathale who, without any financial or any other expectations, taught Sanskrit to some of the authors. They patiently kindled the interest in the language by demonstrating its Maths-like precision along with its poetic beauty which inspired us to explore and work on this further. This work would not have been possible without the seeds of love and selfless service sown by these teachers.

9. Bibliographical References

Bharati, A. and Kulkarni, A. (2007). Sanskrit and computational linguistics. In *First International Sanskrit Computational Symposium. Department of Sanskrit Studies, University of Hyderabad*.

Davis, R. H. (2014). *The Bhagavad Gita: A Biography*. Princeton University Press.

Dongarra, J. J., Meuer, H. W., Strohmaier, E., et al. (1997). TOP500 supercomputer sites. *Supercomputer*, 13:89–111.

Ghosh, A. (2015). Sanskrit fever grips Germany. Daily Mail Online http://www.dailymail.co.uk/, Published 14^{th} April, 2015.

GM. (2017). Sanskrit online tools. Green Message, http://greenmesg.org/sanskrit_online_tools/.

Goyal, P. and Huet, G. (2013). Completeness analysis of a sanskrit reader. In *Proceedings, 5th International Symposium on Sanskrit Computational Linguistics. DK Printworld (P) Ltd*, pages 130–171.

Goyal, P., Arora, V., and Behera, L. (2009). Analysis of Sanskrit text: Parsing and semantic relations. In *Sanskrit Computational Linguistics*, pages 200–218. Springer.

Henning, J. L. (2006). SPEC CPU2006 benchmark descriptions. *ACM SIGARCH Computer Architecture News*, 34(4):1–17.

HT. (2007). Popularity of Sanskrit on rise in US, Europe. Hindustan Times News, Published 10^{th} July, 2007, https://www.hindustantimes.com/.

Huet, G. (2003). Towards computational processing of Sanskrit. In *International Conference on Natural Language Processing (ICON)*.

Huet, G. (2005). A functional toolkit for morphological and phonological processing, application to a Sanskrit tagger. *Journal of Functional Programming*, 15(4):573–614.

Huet, G. (2009). Formal structure of Sanskrit text: Requirements analysis for a mechanical Sanskrit processor. In *Sanskrit Computational Linguistics*, pages 162–199. Springer.

ILTP. (2012). Indian language technology proliferation and deployment center, sandhi-splitter. http://tdil-dc.in/san/sandhi_splitter/index_dit.html.

Jha, G. N. (2017a). Sanskrit sandhi recognizer and analyzer. http://sanskrit.jnu.ac.in/sandhi/viccheda.jsp.

Jha, G. N. (2017b). Special centre for Sanskrit studies. http://sanskrit.jnu.ac.in/index.jsp.

Kapp, D. B. and Malten, T. (1997). Report on the Cologne Sanskrit Dictionary Project. In *10th International Sanskrit Conference, Bangalore*.

Kulkarni, A. (2017). Saṃsādhanī - A Sanskrit computational toolkit. http://sanskrit.uohyd.ac.in/scl/.

Kumar, A., Mittal, V., and Kulkarni, A., (2010). *Sanskrit Compound Processor*, pages 57–69. Springer Berlin Heidelberg, Berlin, Heidelberg.

Mitra, V. L. (1891). Yoga Vashistha Maharamayana of Valmiki (4 vols). *Calcutta: Bannerji*.

Omkarananda, A. H. (2003). Itranslator. https://www.oah.in/Sanskrit/itranslator2003.htm.

Pāṇini and Katre, S. M. (1987). *Aṣṭādhyāyī of Pāṇini*. University of Texas Press.

Prasada, R. (1998). *Patanjali's yoga sutras*. South Asia Books.

Raub, J. A. (2002). Psychophysiologic effects of hatha yoga on musculoskeletal and cardiopulmonary function: a literature review. *The Journal of Alternative & Complementary Medicine*, 8(6):797–812.

Sachin, K. (2007). Sandhi splitter and analyzer for Sanskrit (with reference to ac Sandhi). *M. Phil. degree at SCSS, JNU (submitted, 2007)*.

Saraswati, N. (2013). *Gheranda Samhita*. Yoga Publications Trust, Munger, Bihar, India.

SD. (2017). Panini Ashtadhyayi sutras with commentaries: Sortable index. http://sanskritdocuments.org/learning_tools/ashtadhyayi/.

Singh, P. (1915). *Hatha yoga pradipika*. Dev Publishers.

SJJS. (2017). St James Junior School, Languages Curriculum. https://www.stjamesschools.co.uk/juniorschools/curriculum/academic/languages/.

Swarupananda, S. (2016). *Srimad Bhagavad Gita*. Advaita Ashrama (A publication branch of Ramakrishna Math, Belur Math).

(2017). UBC Sanskrit learning tools. https://ubcsanskrit.ca/.

Uebelacker, L. A., Epstein-Lubow, G., Gaudiano, B. A., Tremont, G., Battle, C. L., and Miller, I. W. (2010). Hatha yoga for depression: Critical review of the evidence for efficacy, plausible mechanisms of action, and directions for future research. *Journal of Psychiatric Practice®*, 16(1):22–33.

UN. (2014). United Nations Resolution 69/131. *General Assembly of the United Nations*. Available from http://www.un.org/en/events/yogaday/.

Yang, K. (2007). A review of yoga programs for four leading risk factors of chronic diseases. *Evidence-Based Complementary and Alternative Medicine*, 4(4):487–491.

Czech Legal Text Treebank 2.0

Vincent Kríž and Barbora Hladká
Charles University
Faculty of Mathematics and Physics
Prague, Czech Republic
{kriz, hladka}@ufal.mff.cuni.cz

Abstract

The Czech Legal Text Treebank 2.0 (CLTT 2.0) contains texts that come from the legal domain and are manually syntactically annotated. The syntactic annotation in CLTT 2.0 is more elaborate than in CLTT 1.0. In addition, CLTT 2.0 contains two new annotation layers, namely the layer of entities and the layer of semantic entity relations. In total, CLTT 2.0 consists of two legal documents, 1,121 sentences and 40,950 tokens.

Keywords: treebank, legal, long sentences, syntactic annotations, named entities, semantic relations

1. Introduction

We have been developing approaches and systems for detecting and extracting semantic relations from unstructured texts. We have developed the RExtractor system (Kríž et al., 2014; Kríž and Hladká, 2015). This system implements an extraction pipeline which processes input texts by linguistically-aware tools and extracts entities and relations using queries over dependency trees. The language used for testing RExtractor is Czech and the legal domain was chosen to be explored in detail.

We surveyed existing syntactically annotated corpora and only a few of them contain texts from the legal domain, e.g., the Universal Dependencies v2.[1] To have a gold-standard data for the RExtractor evaluation, we created the Czech Legal Text Treebank 1.0 (Kríž et al., 2016). In total, 1,121 sentences from the Collection of Laws of the Czech Republic were annotated morphologically and syntactically in accordance with the Prague Dependency Treebank annotation framework.

In this paper, we introduce the next version of CLTT with more elaborate syntactic annotations and enriched with two annotation layers. The remainder of this paper is organized as follows: Section 2. presents a brief description of CLTT 2.0. Modifications in the syntactic annotation are described in Section 3. Section 4. describes the layer of entities and Section 5. presents the layer of semantic relations. Finally, Section 6. provides more details about getting CLTT 2.0.

2. Czech Legal Text Treebank 2.0

We provide basic characteristics of CLTT 2.0 with a special attention paid to the differences between CLTT 1.0 and CLTT 2.0.

2.1. Annotation Layers

Both CLTT 1.0 and CLTT 2.0 annotation principles fit the framework originally formulated in the Prague Dependency Treebank project (PDT, (Hajič et al., 2018)).[2] According to this annotation framework, dependency trees are annotated on the three layers:

- **Word Layer** (`w-layer`)
  A text is segmented into documents and paragraphs and individual tokens are recognized and associated with unique identifiers.
- **Morphological layer** (`m-layer`)
  A sequence of tokens of the word layer is divided into sentences. Annotation of a sentence consists of attaching several attributes to the tokens of the `w-layer`, the most important ones are morphological lemma and tag.
- **Analytical layer** (`a-layer`)
  A sentence is represented as a rooted ordered tree with labeled nodes and edges. One token from the morphological layer is represented by exactly one node in the tree and the dependency relation between two nodes is captured by an edge between the two nodes. The actual type of the relation is given as an analytical function label of the edge.

There are two new layers in CLTT 2.0:

- **Entities Layer** (`e-layer`)
  We focus on entities from the accounting domain. Each entity detected in a text is represented by (i) unique entity identifier, (ii) reference to the dictionary of accounting entities (see below), (iii) identification of the document, the sentence and the tokens where the entity was detected, and (iv) text chunk with the given accounting entity form.
- **Semantic Relations Layer** (`r-layer`)
  A relation is defined as a triple of *subject*, *predicate* and *object*, where both subject and object are accounting entities and predicate is a token (typically a verb) which represents a semantic relation. Analogously to the annotation of entities, each relation has a unique identifier and we distinguish relations of three types, definitions, obligations, and rights.

[1] `http://universaldependencies.org/`
[2] `http://hdl.handle.net/11234/1-2621`

2.2. Data format

Both CLTT 1.0 and CLTT 2.0 use the Prague Markup Language (PML) defined as a main data format by Pajas and Štěpánek (2006). The PML is a generic XML-based data format designed for representation of a rich linguistic text annotation. Both CLTT versions come with a slight modification of the PDT PML Schema.
In CLTT 2.0, the PML files contain new node attributes for entity identification (if an associated token is a part of some entity). In addition, `e-layer` and `r-layer` are stored in separate JSON files which are easily readable by both human and machines.

3. Syntactic Annotation

The syntactic annotations in CLTT 2.0 differs from the ones in CLTT 1.0 in two main aspects: (i) we fixed several errors in the dependency trees, and (ii) we modified the existing naming convention of the node identifiers so it is more readable and easy to understand.

3.1. Fixed Dependency Trees

To make manual syntactic annotation comfortable, we split long and complex sentences into *segments*. A *complex sentence* is a sentence containing at least two segments. A *segment* is a part of a sentence between two numbering markers. It might not be a complete sentence nor even a complete clause. However, its manual annotation becomes more annotator friendly.

The syntactic annotation itself was provided as manual checking and correcting the output of an automatic parser by human annotators. They checked each segment individually – both the tree structure and the analytic function assignment. After that, annotators used inter-segment links to capture dependencies between the nodes from different segments. In fact, using inter-segment links presents a way of building a dependency tree from partial dependency trees. Finally, an automatic procedure joined segment annotations into the final dependency trees for complete complex sentences.

In CLTT 2.0 we checked the dependency trees manually. We fixed several errors that came from both manual inter-segment linking and automatic processing. Unfortunately, several sentences annotated with too erroneous dependency trees had to be removed from the treebank. Thus CLTT 2.0 contains valid dependency trees.

Each dependency tree has been checked three times. The human annotator checked (i) each segment individually, (ii) each final dependency tree (before publishing CLTT 1.0) and (iii) each final dependency tree once more (before publishing CLTT 2.0). All three annotation campaigns have been done by the experienced PDT annotator. Therefore we are not able to provide inter annotator agreement.

3.2. Naming Convention of Node Identifiers

As we mentioned above, the *complex* sentences in CLTT were split into *segments* to make the treebank easier for manual annotation and manipulation. To make searching the complex sentences even more comfortable, we modified the node identifiers in CLTT 2.0 so that the identifiers contain a hierarchical structure that helps to determine the segment position in the complex sentence.

`document`Doc_{id}–`sentence`$Sent_{id}$–[`section`Sec_{id}–[`subsection`Sub_{id}]]

Figure 1: Sentence identifier schema used in the CLTT 2.0

Typically, complex sentence segments depends on each other and so we can describe their hierarchical structure. Table 1 shows a an example of typical complex sentence. In our naming convention, we define *sections* to be complex sentence segments on the first level of numbering, i.e. segments that depend on the introductory segment (line 1 in Table 1). In our example, segments on lines 2, 3 and 6 in Table 1 are *sections*. Analogously, we define *subsections* as segments that depend on a *section* as segments on lines 4 and 5 do.

A sentence identifier schema is presented in Figure 1 and it consists of the following elements:

- **Document identification** – `document`Doc_{id}
  CLTT is distributed in several files. Each sentence identifier starts with Doc_{id} to determine the PML file where the sentence is stored.
- **Sentence identification** – `sentence`$Sent_{id}$
  This identifier provides a unique sentence identification in the PML file.
- **Section identifier** – `section`Sec_{id}
  If a given sentence is complex, then the Sentence identifier determines the first level of numbering used in the complex sentence. We assign the `section0` identifier to the segment where the numbering starts.
- **Subsection identifier** – `subsection`Sub_{id}
  If a given sentence is complex, then the Subsection identifier determines the second level of numbering used in the complex sentence. We assign the `subsection0` identifier to the segment where the numbering starts.

Table 1 presents an example of the naming convention in practice. In fact, two levels of numbering (i.e., section and subsection identifiers) cover all complex sentences in CLTT. However, this strategy could be easily extended to other numbering levels.

Out of 1,121 sentences, 92 sentences were identified as complex sentences and we segmented them into 507 segments. Using the complex sentence segmentation, the average sentence length decreased from 35.9 to 26.2 tokens per sentence.

4. Entity Annotations

In CLTT 2.0, we introduced a new annotation layer of entities. We exploited the dictionary of accounting terms

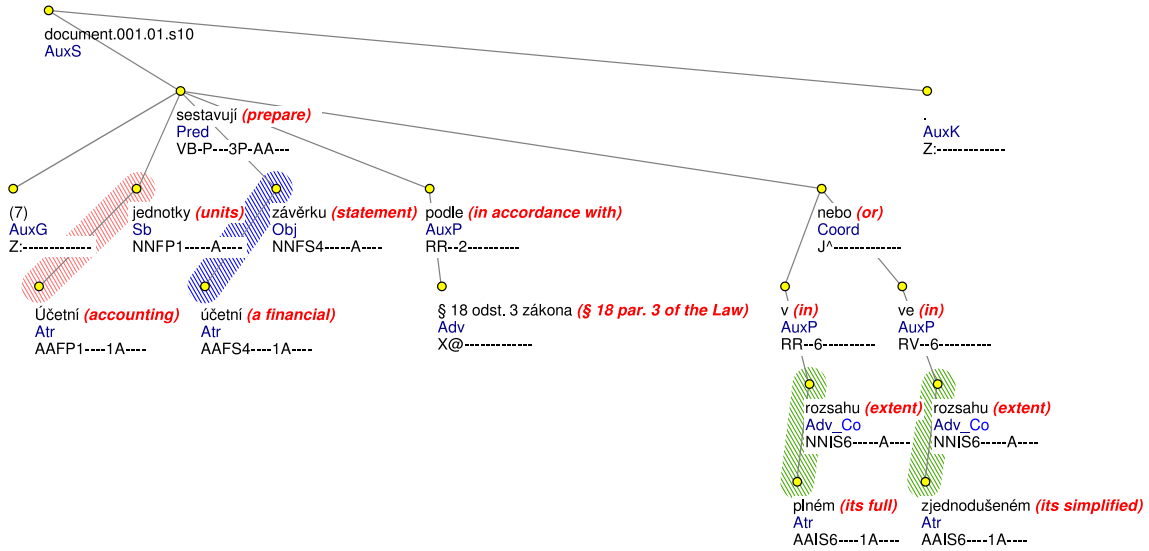


Figure 2: A sample sentence from CLTT 2.0 with highlighted accounting entities.

	Sentence sample	Node identifier prefix
1	(1) Complex sentence:	doc1-sent1-sect0
2	a) first section,	doc1-sent1-sec1
3	b) second section,	doc1-sent1-sec2-sub0
4	1. subsection,	doc1-sent1-sec2-sub1
5	2. subsection,	doc1-sent1-sec2-sub2
6	c) third subsection.	doc1-sent1-sec3
7	(2) Simple sentence.	doc1-sent2

Table 1: An example of the naming convention for the node identifiers in CLTT. The complete identifiers are abbreviated due to the lack of space, i.e., *doc* stands for *document* in the data.

that was created for the RExtractor system. Subsequently, we used the RExtractor system for automatic identification of entities in the CLTT dependency trees.

The dictionary of accounting terms consists of 1,733 different terms classified into 25 categories (see Table 2). The RExtractor system identified 7,332 occurrences in CLTT 2.0. Each detected entity is linked with the particular dictionary entry and its category.

account	general subject	obligation
accounting concept	general term	period
accounting report	incomes	regulation
activity	institution	revenues
agreement	legal person	right
assets	liabilities	state
costs	method	taxes
document	moment	
expenses	natural person	

Table 2: A list of categories in the Accounting Dictionary.

Technically, the detected entities are available in the PML files, namely see the `cltt_entity_id` attribute in the `e-layer`. It allows making tree queries with an entity specification as well as using their visual presentation in the TrEd editor (see Section 6. for more details). All detected entities are also listed in a standalone JSON file.

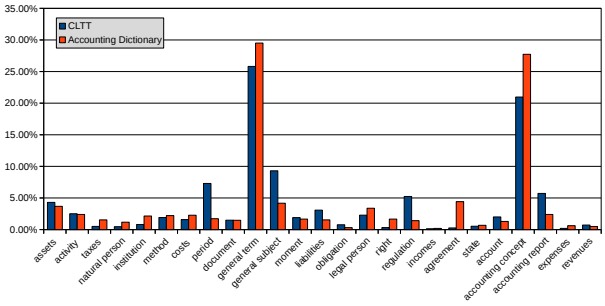


Figure 3: Accounting categories distribution in CLTT 2.0 data and in the Accounting Dictionary.

Figure 3 presents a distribution of different Accounting entities categories over the Accounting Dictionary entries as well as over the entities detected in CLTT 2.0 sentences.

5. Relations

The layer of semantic relations `r-layer` is newly introduced in CLTT 2.0. Relations are represented as (*subject*, *predicate*, *object*) triples, where *subject* and *object* have to be entities and *predicate* represents a relation. Three types of semantic relations were manually annotated in the CLTT texts:

- **Definitions**
  Relations link an entity (*subject*) and its definition (*object*).

- **Rights**
  Relations link an entity (*subject*) which have a given right (*object*) to do something.
- **Obligations**
  Relations link an entity (*subject*) which have a given obligation (*object*) to do something.

Technically, the annotated relations are available in a standalone JSON file with a simple, both human and machine readable structure. Each relation – *definition*, *right*, *obligation* – has a unique identifier. Subject and objects in the relation are represented using references to the entities in the `e-layer`. Predicates are represented by the node reference.

Relations in CLTT 2.0 have been manually annotated by one experienced annotator. As a result, CLTT 2.0 contains 483 manually annotated relations classified into 3 categories. Table 3 presents a relation types distribution and Table 4 lists the most frequent pairs of entity types that appear as relations subjects and objects.

Relation type	Frequency	
Definitions	79	16.36%
Obligations	347	71.84%
Rights	57	11.80%

Table 3: A distribution of different relation types in CLTT 2.0.

Relation	Subject type	Object type	Frequency
Oblig.	general subj.	general term	16.19%
Oblig.	general subj.	acc. concept	9.84%
Oblig.	general subj.	acc. report	8.40%
Oblig.	general subj.	acc. concept	7.17%
Oblig.	general subj.	liabilities	3.69%
Oblig.	general subj.	assets	3.48%
Oblig.	general subj.	account	3.07%

Table 4: The most frequent entity type pairs between subjects and objects.

6. Distributional Notes

CLTT 2.0 is distributed under the Creative Commons, Attribution-NonCommercial-ShareAlike 4.0 International Licence (CC BY-NC-SA 4.0).

6.1. Download

CLTT 2.0 can be downloaded from the LINDAT/CLARIN repository:

`http://ufal.mff.cuni.cz/czech-legal-text-treebank`

In addition, there are various tools for browsing and querying the treebank either locally or on-line, e.g., the TrEd graphical editor, the KonText KWIC search tool and PML TreeQuery:

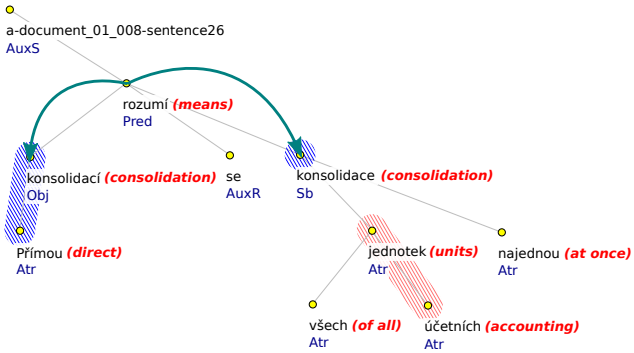


Figure 4: A sample sentence from CLTT 2.0 with the entities and relations highlighted: the *definition* relation between the *direct consolidation* and *consolidation of units*

6.2. TrEd editor

The users can view the treebank off-line using the TrEd editor[3] that we used for the manual annotation of the CLTT. We implemented a new TrEd extension *CLTT* that can be installed directly in TrEd using *Setup* → *Manage Extensions* → *Get New Extensions*.

6.3. KonText

KonText[4] is a web application for querying corpora on-line within the LINDAT/CLARIN project. Users can evaluate simple and complex queries, display their results as concordance lines, compute frequency distribution, calculate association measures for collocations and do further work with the data.

6.4. Tree Query

Tree Query[5] is a powerful open-source search tool for all kinds of linguistically annotated treebanks available on-line within the LINDAT/CLARIN project. Users can evaluate complex tree queries and display their results graphically highlighted in the dependency trees. Tree Query can be run in the TrEd editor.

7. Conclusions

The Czech Legal Text Treebank contains texts from the legal domain. Sentences in legal texts are typically long and very complex. This fact makes the treebank unique and interesting language resource.

We introduced the new version 2.0 of the treebank. It contains 1,121 sentences annotated syntactically using the Prague Dependency Treebank annotation guidelines. In addition, two annotation layers were added, namely the layer of accounting entities and the layer of semantic relations of three types – *definitions*, *rights*, and *obligations*.

CLTT 2.0 is available for free for non-commercial and academic purposes.

[3] `http://ufal.mff.cuni.cz/tred/`

[4] `https://lindat.mff.cuni.cz/services/kontext/first_form?corpname=legaltext_cs_a`

[5] `https://lindat.mff.cuni.cz/services/pmltq`

Acknowledgments

We gratefully acknowledge support from the Charles University project No. SVV 260 453. This work has been using language resources and tools developed and/or stored and/or distributed by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2015071).

Bibliographical References

Kríž, V. and Hladká, B. (2015). RExtractor: a robust information extractor. In Matt Gerber, et al., editors, *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 21–25, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kríž, V., Hladká, B., Nečaský, M., and Knap, T. (2014). Data extraction using NLP techniques and its transformation to linked data. In *13th Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Mexico, November 16-22, 2014. Proceedings, Part I*, volume 8856 of *Lecture Notes in Computer Science*, pages 113–124, Switzerland. Instituto Tecnológico de Tuxtla Gutiérrez, Springer International Publishing.

Pajas, P. and Štěpánek, J. (2006). XML-based representation of multi-layered annotation in the PDT 2.0. In *Proceedings of the LREC Workshop on Merging and Layering Linguistic Information (LREC 2006)*, pages 40–47.

Language Resource References

Hajič, Jan and Bejček, Eduard and Bémová, Alevtina and Buráňová, Eva and Hajičová, Eva and Havelka, Jiří and Homola, Petr and Kárník, Jiří and Kettnerová, Václava and Klyueva, Natalia and Kolářová, Veronika and Kučová, Lucie and Lopatková, Markéta and Mikulová, Marie and Mírovský, Jiří and Nedoluzhko, Anna and Pajas, Petr and Panevová, Jarmila and Poláková, Lucie and Rysová, Magdaléna and Sgall, Petr and Spoustová, Johanka and Straňák, Pavel and Synková, Pavlína and Ševčíková, Magda and Štěpánek, Jan and Urešová, Zdeňka and Vidová Hladká, Barbora and Zeman, Daniel and Zikánová, Šárka and Žabokrtský, Zdeněk. (2018). *Prague Dependency Treebank 3.5*. Institute of Formal and Applied Linguistics, LINDAT/CLARIN, Charles University.

Kríž, V., Hladká, B., and Urešová, Z. (2016). Czech legal text treebank 1.0. In Nicoletta Calzolari, et al., editors, *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2387–2392, Paris, France. European Language Resources Association.

Creation of a Balanced State-of-the-Art Multilayer Corpus for NLU

Normunds Gruzitis, Lauma Pretkalnina, Baiba Saulite,
Laura Rituma, Gunta Nespore-Berzkalne, Arturs Znotins and Peteris Paikens
University of Latvia, Institute of Mathematics and Computer Science, Raina blvd. 29, Riga, Latvia
Latvian Information Agency LETA, Marijas street 2, Riga, Latvia
{normunds.gruzitis,lauma.pretkalnina,baiba.valkovska}@lumii.lv, {peteris.paikens}@leta.lv

Abstract

This paper presents a work in progress to create a multilayered syntactically and semantically annotated text corpus for Latvian. The broad application area we address is natural language understanding (NLU), while more specific applications are abstractive text summarization and knowledge base population, which are required by the project industrial partner, Latvian information agency LETA, for the automation of various media monitoring processes. Both the multilayered corpus and the downstream applications are anchored in cross-lingual state-of-the-art representations: Universal Dependencies (UD), FrameNet, PropBank and Abstract Meaning Representation (AMR). In this paper, we particularly focus on the consecutive annotation of the treebank and framebank layers. We also draw links to the ultimate AMR layer and the auxiliary named entity and coreference annotation layers. Since we are aiming at a medium-sized still general-purpose corpus for a less-resourced language, an important aspect we consider is the variety and balance of the corpus in terms of genres, authors and lexical units.

Keywords: UD, FrameNet, PropBank, AMR, Latvian

1. Introduction

Natural language understanding (NLU) systems for abstractive text summarization, knowledge base population, and many other tasks rely, explicitly or implicitly, on full-stack syntactic and semantic parsing, including semantic role labeling, named entity recognition and linking, and coreference resolution. State-of-the-art parsers, in turn, rely on supervised machine learning which requires substantial language resources – syntactically and semantically annotated text corpora and extensive linked lexicons.

In the industry-oriented research project "Full Stack of Language Resources for Natural Language Understanding and Generation in Latvian", we are creating a balanced text corpus with multilayer annotations, adopting widely acknowledged and cross-lingually applicable representations: Universal Dependencies (UD) (Nivre et al., 2016), FrameNet (Fillmore et al., 2003), PropBank (Palmer et al., 2005) and Abstract Meaning Representation (AMR) (Banarescu et al., 2013).

Figure 1 outlines the inter-layer relationships. First, the UD representation is automatically derived from a more elaborated manually annotated hybrid dependency-constituency representation (see Section 3.). This also ensures that paragraphs, sentences and tokens are correctly and uniformly split, and represented in the standard CoNLL-U data format.[1] Thus, all the annotation layers can be afterwards merged based on the document, paragraph, sentence and token identifiers. Second, the FrameNet annotations are manually added, guided by the underlying UD annotations (see Section 5.1.). Third, the PropBank layer is automatically derived from the FrameNet and UD layers (see Section 5.2.). Fourth, the semi-automatic annotation of named entities, as well as named entity linking, is done in parallel to and independently from the annotation of semantic frames (see Section 4.). Coreference annotations are added afterwards, on top of the named entity annotations, consulting the underlying UD tree if necessary. These two auxiliary layers are required by the ultimate AMR layer. Fifth, draft AMR graphs are to be derived from the UD, PropBank, named entity and coreference annotation layers, with the potential to integrate the FrameNet frames and frame elements into the AMR graphs. As our preliminary experiments show, the semantically richer FrameNet annotations are also helpful in acquiring more accurate AMR graphs (see Section 5.3.). Despite some bootstrapping, all sentences at all layers are eventually checked and post-edited by experienced linguists.

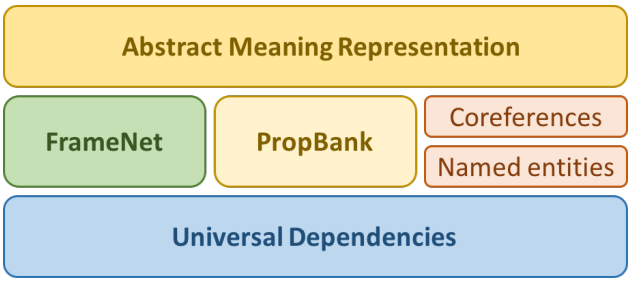


Figure 1: Annotation layers of the NLU corpus.

In this paper, we primarily focus on the consecutive and closely related creation of the treebank and framebank layers, as well as the auxiliary named entity layer. Also note that the above mentioned project addresses only verb frames. A spin-off project has been just launched to work on nominalizations.

The inspiration to create an integrated multilayer corpus comes from the OntoNotes corpus (Hovy et al., 2006) and the Groningen Meaning Bank (GMB) (Bos et al., 2017). The general difference from the OntoNotes approach is that we use the UD model at the treebank layer, and we annotate FrameNet frames in addition to the PropBank frames.

[1] http://universaldependencies.org/format.html

In fact, FrameNet is the primary frame-semantic representation in our approach. Another difference is that we aim at whole-sentence semantic annotation at the ultimate AMR layer. This in some sense is similar to the goal of the GMB project, but the meaning representation used in GMB, Discourse Representation Theory, is a deeper and more complex formalism that can be translated into first-order logic. For pragmatic reasons, we are following the more shallow and lossy AMR formalism. Our experience developing state-of-the-art systems for text-to-AMR parsing (Barzdins and Gosko, 2016) and AMR-to-text generation (Gruzitis et al., 2017), by combining machine learning and grammar engineering approaches, has convinced us that both FrameNet and AMR have a great potential to establish as powerful and complementary semantic interlinguas which can be furthermore strengthened and complemented by other multilingual representations.

Although this work focuses on Latvian, we believe that our experience and findings will be useful for the systematic creation of similar multilayered corpora for other less-resourced languages.

2. Balanced Data Set

Since we are aiming at a medium-sized corpus – around 10,000 sentences – it is crucially important to ensure that it is balanced in terms of text genres and writing styles, as well as lexical units.

A fundamental design decision is that the text unit in our multilayered corpus is an isolated paragraph. The corpus therefore consists of manually selected paragraphs from many different texts of various types.

Regarding genres, representative paragraphs are selected in different proportions from a balanced 10-million-word text corpus: around 60% come from various news sources, around 20% is fiction, around 10% are legal texts, around 5% is spoken language (transcripts), and the rest is miscellaneous. As for the lexical units, our goal is to cover around 1,000 most frequently occurring verbs, calculated from the 10-million-word corpus. Since the most frequent verbs tend to be also the most polysemous, we expect that the number of lexical units (verb senses w.r.t. FrameNet and PropBank frames) will be larger – at least 1,500 units. Nevertheless, the frequently used verbs should have proportionally as many example sentences as possible, while a single sentence often exemplifies the usage of more than one target verb.

Paragraphs to be annotated are therefore selected based on verbs they contain, not randomly (see Figure 2), and curators are constantly updated on the current balance or imbalance of the corpus w.r.t. genres and verb frequencies.

This approach has several benefits:

- Text units are not isolated sentences, allowing for coreference annotation and discourse analysis within the paragraph boundaries. Although coreference resolution is rather irrelevant for the FrameNet and PropBank annotation itself, and sentence-boundary coreferences are sufficient for single-sentence AMR annotation, paragraph-boundary coreference resolution is required by the downstream NLU applications, in combination with named entity linking, to resolve semantic roles expressed by anaphoric references, as well as to connect semantic frames and AMR graphs within the paragraph scope.

- Text units are small enough, allowing for flexible adjustments and paragraph-scrambled distribution of the data set. The adjustments can be made regarding genres, target words and word senses, types and density of named entities, etc.

- The diversity of text authors and writing styles is achieved implicitly – due to the large number of text units selected randomly w.r.t. the authors.

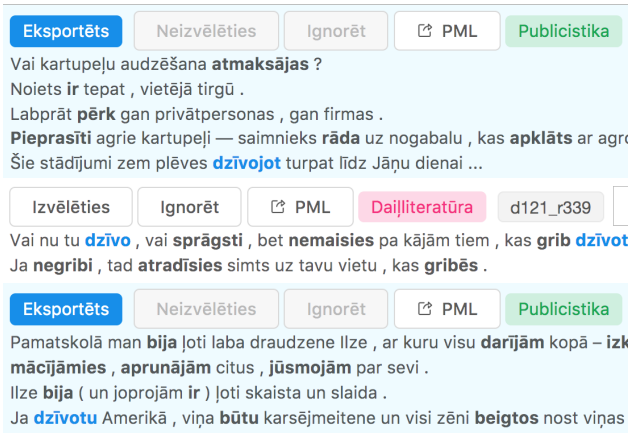


Figure 2: A screenshot of the paragraph selection tool. Candidate paragraphs are filtered by the given target verb 'dzīvot' ('to live', 'to reside' or 'to exist'); other verbs are also highlighted. Genre tags: journalism ('publicistika'), fiction ('daiļliteratūra'), etc.

Our decision about the data selection is justified also by the lessons learned in other treebanking and framebanking projects. For instance, Bick (2017) concludes that a sentence-randomized framebank (based on a sentence-randomized treebank) not only has a limited usage w.r.t. anaphoric relations and discourse analysis but also provides a limited coverage of lexical units.

3. Treebank

Latvian is an Indo-European language with rich morphology and relatively free word order, still many analytic forms are used as well. To capture the language-specific details and to accommodate the linguistic tradition on the one hand, and to meet the goal of the cross-lingual application on the other hand, the treebank annotation is provided in two complementary formalisms. First, the selected paragraphs are manually annotated according to our own hybrid grammar model developed with the linguistic tradition in mind. Second, the hybrid annotation is automatically converted to Universal Dependencies (UD) to achieve the cross-lingual compatibility, as well as to provide training data for efficient and robust parsers.

3.1. Annotation Using a Hybrid Dependency-Constituency Model

For the manual annotation, we use a hybrid dependency-constituency grammar formalism developed in the previous Latvian Treebank (LVTB) pilot project (Pretkalnina et al., 2011). LVTB uses a dependency based hybrid grammar model inspired by Tesnière's concept of syntactic nucleus (Barzdins et al., 2007; Nespore et al., 2010). The syntactic structure of the sentence is modeled by a dependency tree, however, dependencies can be linked not only between single words but also between words and more complex phrasal constructions. This allows for structural distinction between the dependant of a head word and the dependant of a whole phrase.

In LVTB, phrasal constructions are grouped into three classes: (i) coordination, (ii) punctuation mark attachment, and (iii) analytic constructions and other phrases with fixed of partially fixed word order, e.g. prepositional phrases, compound predicates, multi-word numerals, etc.

A sample sentence is shown in Figure 3, where the basic dependency links are brown, the constituency links of analytic forms are green, and the constituency links of the punctuation mark attachment are purple.

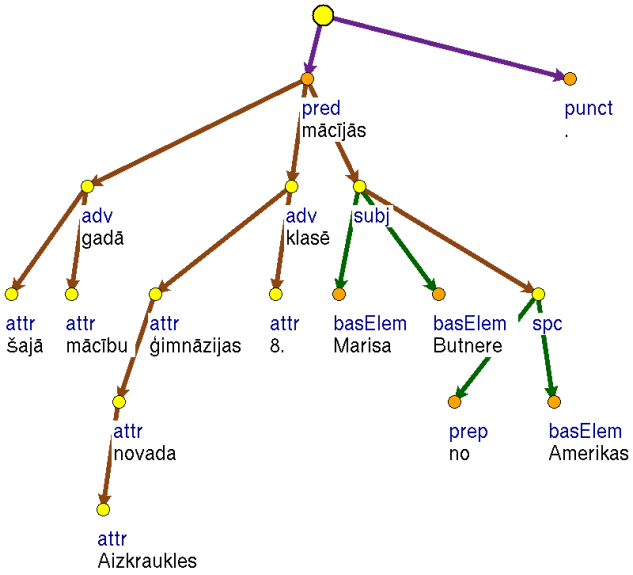


Figure 3: A sample sentence annotated according to our hybrid dependency-constituency grammar model. See Figure 6 and Table 2 for its linearization and enhanced UD representation.

The manual treebank annotation process is as follows:

1. The selected paragraph is automatically tokenized, lemmatized and morphologically tagged. A draft parse tree is automatically acquired using a basic dependency parser trained on LVTB, with limited rule-based post-conversion to the hybrid representation. While the morphological tagging is highly accurate (above 90%), the parser makes a lot of errors (well below 80%). Still, the post-editing of the draft trees is time saving.

2. The hybrid parse trees are post-edited using the TrEd tool (Pajas and Štěpánek, 2008) adapted for the hybrid model.

3. While most of the data is annotated by a single annotator, the perceived problem cases are regularly discussed among the annotators.

3.2. Conversion to Universal Dependencies

To obtain the UD representation from the hybrid representation, we have developed an automatic transformation procedure (Pretkalnina et al., 2016). The latest Latvian UD Treebank (LVUDTB) is periodically released though the UD repository.[2] It was also included in the CoNLL 2017 shared task on multilingual UD parsing (Zeman et al., 2017), being already classified as a relatively big treebank. Till UD v2.0, LVUDTB included only basic dependencies as specified by the UD guidelines. From UD v2.1 we are adding the enhanced dependencies. It is done automatically, extracting as much enhanced UD information as possible from the hybrid annotations and leaving some inaccuracies when there is not enough information available.

The transformation procedure is based on heuristics and on analytic comparison of the two representations. While most of the information necessary for the UD model can be derived from LVTB straightforwardly, it lacks some of the necessary distinctions. Most notably, LVTB does not indicate if a complement takes its own subject (*xcomp* and *ccomp* in UD). Also, since there are no articles or definite suffixes used in Latvian, the distinction between *DET* and *PRON*, and *det* and *nmod* is made heuristically by analyzing the tree structure and the pronoun agreement.

Since the enhanced dependencies were introduced quite after we started to contribute the UD treebank, our transformation was first build to construct the basic UD trees. It was later modified to produce also the enhanced UD graphs from the basic trees, consulting the original LVTB data as well. However, this leads to some inaccuracies, and we plan to rewrite the transformation so that the enhanced graph is built first as it closer follows the original hybrid representation, and then it is reduced to the basic dependencies.

The hybrid to UD transformation currently consists of the following steps:

1. **Tokenization**. The LVTB tokenization guidelines are tweaked to match the UD guidelines: abbreviations and numbers with spaces as group separators are treated as single tokens (e.g. 'u.c.', '1 000 000').

2. **Lemmatization**. Since UD has no strict guidelines on lemmatization, lemmas are taken as is from LVTB.

3. **POS tags and morphological features**. Most of the POS tags can be obtained directly from the LVTB tagset, but some tags (e.g. *DET*) are decided by analyzing the syntactic structure. The LVTB tagset ensures a good coverage of the morphological features as well.

[2] http://universaldependencies.org/#download

4. **Basic dependencies and ellipsis**. Basic dependencies are obtained via a bottom-up traversal through the original tree. Conveniently, phrasal constructions in the hybrid trees correspond to isolated subtrees in the UD trees. Thus, every phrase can be transformed by considering only its constituents and its parent (or the direct ancestor if the parent is a coordinated constituent). Basic dependencies between single words are used as is. Any dependency to/from a phrase can be transformed to a dependency to/from the root of the subtree representing the corresponding phrase.

   The transformation of roles is more complicated, as the relationship between the LVTB roles and the UD roles is mostly many-to-many. This is because LVTB uses a more semantically oriented role set than UD. For instance, while LVTB distinguishes between attributes (adjectives or nouns) and adverbial modifiers (adverbs or nouns), UD distinguishes between nominal modifiers, numeric modifiers and adverbs.

   Due to the fact that ellipsis are represented by empty nodes in LVTB, it is convenient to build an enhanced dependency "backbone" already in the same step. The backbone tree contains the relevant arcs from the basic dependency tree, the ellipsis nodes, and the links connecting the ellipsis nodes to the tree.

5. **Other enhanced dependencies**. Both coordination and compound predicates are annotated as phrasal constructions in LVTB. Since dependants of a phrasal construction (as a whole) are annotated structurally different from the dependants of head constituents of the phrase, obtaining links related to the coordination propagation and raised (controlled) subjects is relatively straightforward. Similarly, it is possible to obtain the case information from either the morphological tag of the preposition, or from the non-terminal phrase representing the prepositional construction.

For the enhanced dependencies, the UD guidelines suggest adding the following types of annotations:

- Null nodes for elided predicates – added.
- Additional subject relations for control and raising constructions – added.
- Propagation of conjuncts – added with rare inaccuracies. The current implementation can fail to obtain a correct enhanced dependency role for the second and further conjuncts, if the conjuncts belong to different parts-of-speech. This problem will be solved when the transformation is rewritten as planned.
- Modifier labels that represent the preposition, or other case-marking information – partially added. Case information is available for prepositional constructions and some types of subordinated clauses.
- Coreferences in relative clause constructions – not available. LVTB contains no information on coreferences. Coreference annotations will be eventually available from the coreference layer (see Figure 1 and Section 4.2.).

The necessity to transform the LVTB data to the UD representation also helps to review the hybrid annotation scheme from a different perspective. To acquire more accurate UD transformation results, we have improved the LVTB annotation guidelines, e.g. for more detailed annotation of compound predicates, and for revised annotation of comparison constructions. Overall, the hybrid annotation model contains more information compared not only to the basic but also to the enhanced UD dependencies, even though it lacks some of the distinctions which UD makes.

4. Named Entities

Named entities are essential for most NLU tasks, since they link the textual content to the real world, making the extracted facts (frames) meaningful for a knowledge base population task, for instance. From the multilayer corpus perspective, the AMR annotation heavily relies on named entity recognition and linking (as illustrated in Figure 7), and on within-sentence coreference resolution, using the reentrancy representation. This allows for connecting individual AMR graphs and subgraphs into a wider context.

4.1. Named Entity Annotation

In our corpus, we are using the following set of named entity categories: *person*, *organization*, *geopolitical entity* (GPE), *location*, *product*, *time* (relative or absolute date, time, or duration), *event*, and *entity* (entities of other categories that occur rarely but could be considered in future). We mostly follow the MUC-7 annotation guidelines (Chinchor, 1998) which we have extended for compatibility with the top-level named entity categories specified by the AMR guidelines.[3] For serialization of the named entity layer, we use a version of the CoNLL-2003 data format[4] on top of CoNLL-U (see Table 1).

In practice, many named entities contain references to other named entities. We do annotate nested (hierarchical) entities (see Figure 4), which enables us to exploit the annotations of either the outer or inner entities, or both, when annotating coreferences or deriving AMR graphs. This would not be possible with a flat annotation scheme. Furthermore, hierarchical annotations not only allow for the development of an automatic hierarchical named entity recognizer (NER) but also provide more training data for the development of a flat NER.

4.2. Named Entity Linking and Coreference Resolution

We approach coreference annotation in a pragmatic way, focusing on precision and annotating coreferences only within the paragraph boundaries. This allows to annotate a lot of various text units, and it makes the annotation process easier and less error prone. We are mostly following the English coreference guidelines used in the OntoNotes project.[5] We annotate pronominal and nominal noun phrases referring to real-word entities, and non-specific mentions if they are antecedents of pronouns. Bridging

[3] https://github.com/amrisi/amr-guidelines

[4] https://www.clips.uantwerpen.be/conll2003/ner/

[5] https://catalog.ldc.upenn.edu/docs/LDC2013T19/

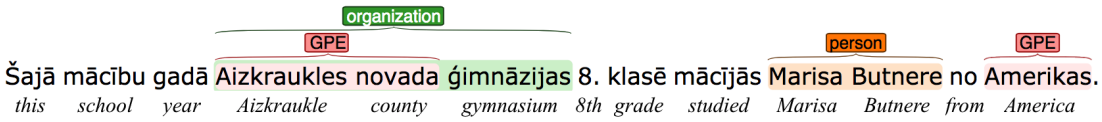


Figure 4: Hierarchical named entity annotation and wikification (behind the scenes) using WebAnno.

Table 1: A data format used to serialize the named entity layer of the corpus: a version of CoNLL-2003 based on CoNLL-U. Because of space restrictions, several less relevant CoNLL-U fields are excluded from the table. In addition to the CoNLL-2003 fields, we include extra fields for the inner entities and for the wikification of both outer and inner entities.

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	**BIOTAG$_1$**	**BIOTAG$_2$**	**WIKI$_1$**	**WIKI$_2$**
1	Šajā	šis	DET	pd0fsln	O	_	_	_
2	mācību	mācība	NOUN	ncfpg4	O	_	_	_
3	gadā	gads	NOUN	ncmsl1	O	_	_	_
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	B-organization	B-GPE	_	lv:Aizkraukles_novads
5	novada	novads	NOUN	ncmsg1	I-organization	I-GPE	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	I-organization	_	_	_
7	8.	8.	ADJ	xo	O	_	_	_
8	klasē	klase	NOUN	ncfsl5	O	_	_	_
9	mācījās	mācīties	VERB	vmyisi330an	O	_	_	_
10	Marisa	Marisa	PROPN	npfsn4	B-person	_	_	_
11	Butnere	Butnere	PROPN	npfsn5	I-person	_	_	_
12	no	no	ADP	spsg	O	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	B-GPE	_	en:United_States	_
14	.	.	PUNCT	zs	O	_	_	_

relations, discontinuous expressions, split antecedents and zero anaphora are ignored.

In addition to annotating the named entity spans and categories, we also specify a corresponding Wikipedia identifier (URI) if one exists. First, we look for a corresponding article in the English Wikipedia. Second, if no article is found, we look for a corresponding article in the Latvian Wikipedia (see the different namespace prefixes in Table 1). For training a named entity linker (NEL), such corpus would be considered a very small one, but it will be helpful for evaluating a NEL. For this reason, we have specially included text units mentioning different persons with the same name, for instance. The manually verified Wikipedia identifiers are also useful when generating draft AMR graphs (see Section 5.3.).

5. Semantic Frames

The annotation of PropBank frames is relatively more simple if compared to FrameNet, since PropBank frames are less abstract, and their semantic roles directly follow from the syntactic verb argument structure. Creating the Latvian framebank, however, we start with annotating FrameNet frames, and we are deriving the PropBank annotations automatically from the FrameNet and UD annotations.

5.1. UD-Based Annotation of FrameNet

The creation of the FrameNet annotation layer is as follows. Paragraphs for which the manual treebank annotation is finalized and which have been successfully converted from the hybrid grammar to the UD representation are stored in a separate repository. While treebank, named entity and coreference annotation is done paragraph by paragraph, this is not a productive workflow for annotating semantic frames, especially in case of the highly abstract FrameNet frames. Instead, a concordance view is required, so that the linguist can focus on a target verb and its different senses (frames), without constantly switching among different sets of frames. This also improves the annotation consistency.

To provide such environment, we automatically extract all UD-annotated sentences from the finalized paragraphs containing the requested target verb, and we store the result in a separate temporary CoNLL-U file. When more paragraphs are finalized at the UD layer, they are considered in the next concordance queries. The acquired concordance files are imported in the WebAnno platform (Eckart de Castilho et al., 2016) which we have specifically configured for the FrameNet annotation. When the annotation is done, the finalized concordances are exported from WebAnno and are eventually reorganized back into paragraphs.

Figure 5 illustrates a sample concordance with the resulting FrameNet frame and frame element (FE) annotations (the UD annotations are hidden for the sake of simplicity). The actual annotation, however, is done on top of the UD layer, as illustrated in Figure 6 and Table 2. Such approach has a significant consequence: FEs are not annotated as spans of text – only the head word of a UD subtree is annotated; the whole span can be expanded automatically by traversing the respective subtree. This not only makes the annotation process more simple and the annotations more consistent, but it also facilitates the learning of automatic semantic role labeling, since it is easier to identify the syntactic head of a FE than a span of a string. Still, most FrameNet corpora are annotated in terms of spans, relying on syntactic parsing as a post-processing step.

The creation of the FrameNet layer is described in more detail by Gruzitis et al. (2018).

Table 2: A data format used to serialize the FrameNet layer of the corpus: a version of CoNLL-2009 based on CoNLL-U. Because of space restrictions, several CoNLL-U fields are excluded from the table.

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	DEPS	**FILLPRED**	**PRED**	**APRED$_1$**
1	Šajā	šis	DET	pd0fsln	3:det	_	_	_
2	mācību	mācība	NOUN	ncfpg4	3:nmod:gen	_	_	_
3	gadā	gads	NOUN	ncmsl1	9:obl:loc	_	_	Time
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	5:nmod:gen	_	_	_
5	novada	novads	NOUN	ncmsg1	6:nmod:gen	_	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	8:nmod:gen	_	_	Institution
7	8.	8.	ADJ	xo	8:amod	_	_	_
8	klasē	klase	NOUN	ncfsl5	9:obl:loc	_	_	Level
9	mācījās	**mācīties**	VERB	vmyisi330an	0:root	Y	**Education_teaching**	_
10	Marisa	Marisa	PROPN	npfsn4	9:nsubj	_	_	Student
11	Butnere	Butnere	PROPN	npfsn5	10:flat:name	_	_	_
12	no	no	ADP	spsg	13:case	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	10:nmod:no	_	_	_
14	.	.	PUNCT	zs	9:punct	_	_	_

Table 3: A data format used to serialize the PropBank layer of the corpus: a version of CoNLL-2009 based on CoNLL-U. The PropBank annotations are semi-automatically derived from the FrameNet layer (see Table 2).

ID	FORM	LEMMA	UPOSTAG	XPOSTAG	DEPS	**FILLPRED**	**PRED**	**APRED$_1$**
1	Šajā	šis	DET	pd0fsln	3:det	_	_	_
2	mācību	mācība	NOUN	ncfpg4	3:nmod:gen	_	_	_
3	gadā	gads	NOUN	ncmsl1	9:obl:loc	_	_	AM-TMP
4	Aizkraukles	Aizkraukle	PROPN	npfsg5	5:nmod:gen	_	_	_
5	novada	novads	NOUN	ncmsg1	6:nmod:gen	_	_	_
6	ģimnāzijas	ģimnāzija	NOUN	ncfsg4	8:nmod:gen	_	_	_
7	8.	8.	ADJ	xo	8:amod	_	_	_
8	klasē	klase	NOUN	ncfsl5	9:obl:loc	_	_	AM-LOC
9	mācījās	**mācīties**	VERB	vmyisi330an	0:root	Y	**study.01**	_
10	Marisa	Marisa	PROPN	npfsn4	9:nsubj	_	_	A0
11	Butnere	Butnere	PROPN	npfsn5	10:flat:name	_	_	_
12	no	no	ADP	spsg	13:case	_	_	_
13	Amerikas	Amerika	PROPN	npfsg4	10:nmod:no	_	_	_
14	.	.	PUNCT	zs	9:punct	_	_	_



Figure 5: WebAnno screenshot: FrameNet-annotated occurrences of the target verb 'dzīvot' ('to live/reside/exist').

5.2. Conversion to PropBank

The semantic roles in PropBank are much more robust compared to FrameNet, and the overall PropBank annotation systematically follows the verb argument structure. Therefore the PropBank layer can be semi-automatically derived from the FrameNet and UD layers, by providing a mapping configuration from lexical units in FrameNet to PropBank frames (see Table 4), and a mapping configuration from FrameNet frame elements to PropBank semantic roles for the given pair of FrameNet and PropBank frames (see Table 5). We are building on the previous work on SemLink (Palmer, 2009) and Predicate Matrix (Lopez de Lacalle et al., 2016). We use the provided mapping between English FrameNet and English PropBank as a draft configuration. The linguistically intensive manual task is to map the lexical units from Latvian FrameNet to the semantic frames of English PropBank. The rest is a straightforward automation. Table 3 illustrates a PropBank-annotated sentence, where the annotation has been derived from the FrameNet and UD layers (Table 2). Note that the FrameNet annotation is semantically richer, as well as it might be non-projective w.r.t. the underlying UD tree. In the given example, the frame element *Institution* is not transfered to the PropBank layer because it is not a syntactic argument of the target verb.

5.3. Generation of AMR

So far we have conducted only limited preliminary experiments on the AMR annotation based on the underlying UD,

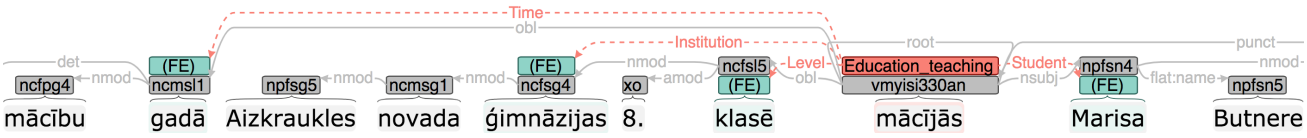


Figure 6: FrameNet annotation on top of a UD tree. Only head nodes are selected while annotating frame elements (FE). The FE spans can be acquired automatically by traversing the respective subtrees: [.. *school year*]$_{\text{Time}}$ [*Aizkraukle county gymnasium*]$_{\text{Institution}}$ [*8th grade*]$_{\text{Level}}$ ***studied***$_{\text{EDUCATION_TEACHING}}$ [*Marisa Butnere* ..]$_{\text{Student}}$.

Table 4: Mapping from FrameNet frames to PropBank frames, taking the lexical units into account.

LEMMA	UPOSTAG	PRED$_{\text{FrameNet}}$	**PRED$_{\text{PropBank}}$**
mācīties	VERB	Education_teaching	study.01
mācīt	VERB	Education_teaching	teach.01
mācība	NOUN	Education_teaching	training.01
dzīvot	VERB	Residence	reside.01

Table 5: Mapping from FrameNet frame elements to PropBank semantic roles, taking the UD dependency relations into account.

PRED$_{\text{FN}}$	APRED$_{\text{FN}}$	DEPREL	PRED$_{\text{PB}}$	**APRED$_{\text{PB}}$**
Education_teaching	Student	nsubj	study.01	A0
Education_teaching	Student	obj	teach.01	A2
Education_teaching	Student	iobj	teach.01	A2
Education_teaching	Subject	obj	study.01	A1
Education_teaching	Subject	obj	teach.01	A1
Education_teaching	Teacher	obl	study.01	A2
Education_teaching	Teacher	nsubj	teach.01	A0
Education_teaching	Institution	obl	study.01	AM-LOC
Education_teaching	Institution	obl	teach.01	AM-LOC
Education_teaching	Level	obl	study.01	AM-LOC
Education_teaching	Time	obl	study.01	AM-TMP
Education_teaching	Time	obl	teach.01	AM-TMP

PropBank and FrameNet layers, as well as the auxiliary named entity and coreference layers. However, it seems feasible to systematically generate draft AMR annotations for manual post-editing, thus, boosting the productivity and acquiring more consistent AMRs. An illustrative example is given in Figure 7.

The auxiliary layers were not part of the initial work plan. The information they convey was planned to be added during the AMR annotation – only as part of the AMR graphs. However, we have introduced them as separate layers to facilitate the semi-automatic generation of AMR representations in addition to the utility of these auxiliary layers per se in practical applications.

Note that although FrameNet frames and frame elements are not explicitly integrated in AMR, FrameNet annotations still support the systematic construction of AMR graphs. In the above example, the *x8* instance is linked by the *location* relation instead of a more general relation thanks to the FrameNet frame element *Institution* which is not covered at the PropBank layer.

6. Conclusion

The consecutive treebank and framebank annotation workflow has turned out to be very productive and mutually beneficial. The dependency tree facilitates the annotation of semantic frames and roles, while the frame semantic analysis of verb valency often unveils some inconsistencies or bugs in the dependency annotation or in the morphological tagging. These issues are immediately fixed in the treebank. Similarly, the annotation of other layers helps to notice other kinds of bugs and inconsistencies in the underlying layers.

```
(x1 / study-01
  :ARG0 (x2 / person
    :name (x3 / name
      :op1 "Marisa" :op2 "Butnere")
    :source (x4 / GPE
      :name (x5 / name :op1 "Amerika")
      :wiki "United_States"))
  :location (x6 / klase
    :ord (x7 / ordinal-entity :value 8)
    :location (x8 / organization
      :name (x9 / name
        :op1 "Aizkraukles"
        :op2 "novada"
        :op3 "ģimnāzija")))
  :time (x10 / gads
    :mod (x11 / šis)
    :mod (x12 / mācība)))
```

Figure 7: A draft AMR graph to be generated, consulting all the underlying annotation layers (see Table 3, 2 and 1).

While the FrameNet annotation helps to eliminate certain types of syntactic annotation errors, the automatic conversion from the hybrid dependency-constituency grammar to the UD representation helps to unveil and fix other types of inconsistencies and bugs. The decision to use the more complex and rich hybrid representation for the manual annotation of the treebank has paid off even more: the enhanced UD dependencies can be derived without any additional annotation efforts, and there is more information to derive if necessary.

Although we are primarily focusing on verbs and verb frames that are the primary units making a sentence (or clause), the syntactic and semantic arguments in the sentence are not controlled only by verbal predicates. Deverbal derivatives, i.e. frame-evoking nouns and adjectives are often used as well, and they usually preserve the syntactic and semantic features of a verb, such as transitivity or the capability of taking nominal or verbal complements. Deverbal derivatives are inevitable in real-world full-text se-

mantic parsing, therefore we address these phenomena in a spin-off project which has been just launched and is building on the same data set.

The multilayer corpus is being gradually released on GitHub under the CC BY-NC-SA 4.0 licence.[6]

7. Acknowledgements

This work has received financial support from the European Regional Development Fund under the grant agreements No. 1.1.1.1/16/A/219 and No. 1.1.1.2/VIAA/1/16/188.

8. Bibliographical References

Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract Meaning Representation for Sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria.

Barzdins, G. and Gosko, D. (2016). RIGA at SemEval-2016 Task 8: Impact of Smatch extensions and character-level neural translation on AMR parsing accuracy. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, pages 1143–1147, San Diego, California.

Barzdins, G., Gruzitis, N., Nespore, G., and Saulite, B. (2007). Dependency-based hybrid model of syntactic analysis for the languages with a rather free word order. In *Proceedings of the 16th Nordic Conference of Computational Linguistics*, pages 13–20, Tartu, Estonia.

Bick, E. (2017). From Treebank to Propbank: A Semantic-Role and VerbNet Corpus for Danish. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 202–210, Gothenburg, Sweden.

Bos, J., Basile, V., Evang, K., Venhuizen, N., and Bjerva, J. (2017). The Groningen Meaning Bank. In Nancy Ide et al., editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer.

Chinchor, N. (1998). Appendix E: MUC-7 Named Entity Task Definition. In *Proceedings of th 7th Message Understanding Conference (MUC)*, Fairfax, Virginia.

Eckart de Castilho, R., Mújdricza-Maydt, E., Yimam, S. M., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C. (2016). A Web-based Tool for the Integrated Annotation of Semantic and Syntactic Structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities*, pages 76–84, Osaka, Japan.

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to FrameNet. *International Journal of Lexicography*, 16(3):235–250.

Gruzitis, N., Gosko, D., and Barzdins, G. (2017). RIGOTRIO at SemEval-2017 Task 9: Combining machine learning and grammar engineering for AMR parsing and generation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 924–928, Vancouver, Canada.

Gruzitis, N., Nespore-Berzkalne, G., and Saulite, B. (2018). Creation of Latvian FrameNet based on Universal Dependencies. In *Proceedings of the 11th International Conference on Language Resources and Evaluation*, Miyazaki, Japan.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% Solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA.

Lopez de Lacalle, M., Laparra, E., Aldabe, I., and Rigau, G. (2016). A Multilingual Predicate Matrix. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 2662–2668, Portoroz, Slovenia.

Nespore, G., Saulite, B., Barzdins, G., and Gruzitis, N. (2010). Comparison of the SemTi-Kamols and Tesniere's dependency grammars. In *Human Language Technologies – The Baltic Perspective*, volume 219 of *Frontiers in Artificial Intelligence and Applications*, pages 233–240. IOS Press.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1659–1666.

Pajas, P. and Štěpánek, J. (2008). Recent advances in a feature-rich framework for treebank annotation. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 673–680, Manchester, UK.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Palmer, M. (2009). SemLink: Linking PropBank, VerbNet and FrameNet. In *Proceedings of the Generative Lexicon Conference*, pages 9–15, Pisa, Italy.

Pretkalnina, L., Nespore, G., Levane-Petrova, K., and Saulite, B. (2011). A Prague Markup Language profile for the SemTi-Kamols grammar model. In *Proceedings of the 18th Nordic Conference of Computational Linguistics*, pages 303–306, Riga, Latvia.

Pretkalnina, L., Rituma, L., and Saulite, B. (2016). Universal Dependency treebank for Latvian: A pilot. In *Human Language Technologies – The Baltic Perspective*, volume 289 of *Frontiers in Artificial Intelligence and Applications*, pages 136–143. IOS Press.

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., et al. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada.

[6] https://github.com/LUMII-AILab/FullStack

Test Sets for Chinese Nonlocal Dependency Parsing

Manjuan Duan, William Schuler
Department of Linguistics
The Ohio State University
1712 Neil Ave. Columbus OH, 43210
{duan.59, schuler.77}@osu.edu

Abstract

Chinese is a language rich in nonlocal dependencies. Correctly resolving these dependencies is crucial in understanding the predicate-argument structure of a sentence. Making full use of the trace annotations in the Penn Chinese Treebank (Xue et al., 2005), this research contributes several test sets of Chinese nonlocal dependencies which occur in different grammatical constructions. These datasets can be used by an automatic dependency parser to evaluate its performance on nonlocal dependency resolution in various syntactic constructions in Chinese.

Keywords: nonlocal dependency, parsing, Mandarin Chinese

1. Introduction

Recovering unbounded dependencies is challenging in English, as reported by Rimell et al. (2009). However, it serves as an important test of an automatic parser which cannot be easily accomplished by shallow language techniques. In spite of the low frequency of some nonlocal dependency constructions, correctly resolving these dependencies is crucial in understanding the underlying predicate-argument structure of a sentence.

Although trace categories are annotated in the Penn Treebank (Xue et al., 2005), few state-of-the-art constituent parsers make use of these annotations to make predictions of the nonlocal dependencies. Categorial grammar annotations of the Treebank are advocated partly for their well-defined representations of filler-gap phenomena in natural languages. Parsers trained on categorial grammar annotations are found to produce superior performance in recovering nonlocal dependencies in English (Rimell et al., 2009; Nguyen et al., 2012).

In this research we focus on various nonlocal dependency types in Mandarin Chinese, a language that makes heavy use of nonlocal dependencies (Kummerfeld et al., 2013). We make full use of the trace categories annotated in the Penn Chinese Treebank (Xue et al., 2005) to generate test sets for eight nonlocal dependency constructions. We evaluate the nonlocal dependency recovery performance of parsers trained on generalized categorial grammar annotations (Bach, 1981; Nguyen et al., 2012; Duan and Schuler, 2015) with the annotated test sets. We hope these test sets can make it easier to automatically evaluate nonlocal dependency recovery in Mandarin Chinese.

2. Nonlocal Dependency Data Sets

2.1. The Constructions

In a nonlocal dependency, a constituent seems to be moved from its canonical position in the predicate-argument pattern, while it still needs to be interpreted in the position from which it is moved. For this research we examined the trace categories annotated in Penn Chinese Treebank and included those constructions where either both the trace and the filler were clearly annotated or they can reliably be recovered from the syntactic trees. For example, the Chinese Penn Treebank has clear annotations about the locations of traces in relative clauses but the fillers are 'WHNP' empty categories. However, the head noun of a relative clause can always be reliably and accurately located given the tree structure. We therefore include relative clauses as a nonlocal construction in this study.

The only exception is a type of relative clause where the head noun is an adverbial modifier of the relative clause, as shown in (1).

(1) Zhe jiu shi ta jingchang dubo de yuanyin
this exactly is he often gamble de reason
'That's why he often gambles.'

In (1), the head noun *yuanyin* 'reason' is annotated as a moved adverbial modifier of the verb *dubo* 'gamble'. We did not include this type of relative clause as a nonlocal construction because unlike other nonlocal constructions we annotated, the filler in this type of construction cannot be put back into the trace location to form a grammatical sentence in Chinese, which makes the nonlocal dependency controversial. Also this type relative clause has identical syntactic structure to an appositive clause in Chinese. Sometimes, they can be semantically indistinguishable.

We annotated eight types of nonlocal dependencies in which both the trace and filler are reliably available in the Treebank annotations. Here is a brief introduction to each of them.

Subject relative clause

Subject relative clauses are constructions in which a subject is extracted from a relative clause and a trace category is annotated in the subject position in the relative clause. In Mandarin Chinese, a relative clause is followed by a particle *de* and the head noun occurs after the *de* particle.

(2) meiyou yingxiang de xinwen yao diudiao
do-not-have video de news needs dump
'dump the news which do not have videos'
1(*meiyou* 'do-not-have', *xinwen* 'news')

For example, in (2), the head noun *xinwen* 'news' occurs after *de* and serves as the subject of the verb *meiyou* 'do-not-have' in the relative clause in (2). We label the relation '1' between the extracted subject, *xinwen* 'news', and the verb, *meiyou* 'do-not-have' , to indicate the extracted subject is the first argument of the verb. We exclude those cases from our test sets where the head noun of the relative clause is not specified linguistically, so that a concrete dependency can be established between a pair of words.

Object relative clause
Object relative clauses are constructions where the object is extracted from the relative clause as shown in (3).

(3) zhexie dou buneng dailai ta xuyao de xingfu
these all cannot bring he needs de happiness
'All these cannot bring the happiness that he needs'
2(*xuyao* 'need', *xingfu* 'happiness')

In (3), a label '2' is given to the dependency between the extracted object, *xingfu* 'happiness', and the verb, *xuyao* 'need'. It means the extracted object is the direct object of the verb. We have not found any extracted indirect objects in relative clauses in the Treebank data.

Topic relative clause
Topic relative clauses are constructions where the topic is extracted from the relative clause. In all the topic relative clauses we examined, we find there is no predicate-argument relation between the extracted topic and the verb in the relative clause. Instead, the extracted topic has a semantic relation with the subject of the relative clause which is similar to the relation in English expressed by 'the subject *of* the topic'.

(4) huzi changguo toufa de welia
beard longer-than hair de Welia
'Welia, whose beard is longer than his hair'
of-asso(*huzi* 'beard', *welia* 'Welia')

For example, in (4), the extracted topic *Welia* is not an argument of the verb *changguo* 'longer-than'. It is related to the subject *huzi* 'beard' instead. Therefore, we labeled an 'of-asso' dependency between the extracted topic and the subject of the relative clause.

Focus construction
Focus constructions, or sometimes called *even*-constructions in Mandarin Chinese, have the constituent in focus occur pre-verbally.

(5) ta shuijiao de difang dou meiyou
he sleep de place even do-not-have
'He does not even have a place to sleep.'
2(*meiyou* 'do-not-have', *difang* 'place')

In (5), *shuijiao de difang* 'place to sleep' is the constituent in focus. It occurs before the verb *meiyou* 'do-not-have', rather than after the verb, the canonical position of a direct object. Since *shuijiao de difang* 'place to sleep' is the direct object of *meiyou* 'do-not-have', there is a labeled '2' dependency between them.

Passivization of direct objects
Passivizations of direct objects are passivized sentences where the direct object is fronted to become the subject of the sentence.

(6) gonggong changhe chouyan yinggai bei jinzhi
public place smoking should bei forbidden
'Smoking in public should be forbidden'
2(*jinzhi* 'forbid', *chouyan* 'smoking')

Most passivized sentences in the Treebank data belong to this category. We give the label '2' to the dependency between the verb and the fronted subject since the subject is the second argument of the transitive verb.

Passivization of indirect objects
Passivizations of indirect objects are passivized sentences where the indirect object is fronted to be the subject of the sentence.

(7) liangren bei geiyu nanmin shenfen
two-people bei given refugee status
'Two people were given the status of refugee'
3(*geiyu* 'give', *liangren* 'two-people')

We give the label '3' to the dependency between the ditransitive verb and the fronted object since it is the third argument of the verb.

Topicalization
Topicalizations are constructions where a word or phrase is moved to the sentence initial position to serve as a topic of the sentence. The dependency label could be '1' or '2', depending on whether the fronted word or phrase is the first argument or the second argument of the verb.

(8) zhezhong shiqing ni bushi diyici
this-sort-of things you not first-time
pengdao
came-across
'It is not the first time that you came across this sort of things.'
2(*pengdao* 'came-across', *shiqing* 'things')

In (8), the direct object *shiqing* 'things' is moved to the front of the sentence. Therefore a label '2' is given to the dependency between *pengdao* 'came-across' and *shiqing* 'things'.

Extraction from an embedded clause
Extractions from an embedded clauses are constructions where a subject or an object is moved across at least two clause boundaries.

(9) shouji keyi shuo shi zhege shidai de yi bufen
cell-phone can say is this age de a part
'Cell phone, we can say, is a part of this age'
1(*shi* 'is', *shouji* 'cell-phone')

In (9), the noun phrase *shouji* 'cell-phone' is extracted to the sentence initial position from its subject position within the complement clause of *shuo* 'say'. A '1' dependency is annotated between *shi* 'is' and *shouji* 'cell phone' to indicate that 'cell-phone' is the first argument of 'is'.

Construction	# sents	# deps	Freq %
Sbj rel	111	147	23.0
Obj rel	112	127	9.0
Tpc rel	97	104	1.4
Foc	87	96	0.6
Pass direct	110	178	2.0
Pass indirect	61	65	0.2
Topicalization	116	141	0.6
Embedded	42	48	0.1

Table 1: Test data size and frequency of the constructions.

2.2. The Data

We divided the Chinese Penn Treebank 6 into training, development and test sets as indicated by Tse and Curran (2010). From the test set and all the other sentences in the Penn Chinese Treebank 8 that are not included in the training set, for each nonlocal construction, we randomly choose 120 sentences to annotate with dependencies. If we could not find 120 cases for some rare constructions, we include all the occurrences from the test set in order to have a test set of reasonably large size. We annotate all the dependencies according to their trace annotations in the Penn Chinese Treebank. We removed the sentences where either the head or the dependent in the nonlocal dependency is not present linguistically. Sometimes, there is more than one occurrence of a particular construction in one sentence. In these cases, more than one nonlocal dependency is annotated. The size of the test set for each construction and the distribution of the construction in the Penn Chinese Treebank are shown in Table 1.

We can see subject relative clauses and object relative clauses are very common in the corpus. Around a third of sentences in the Penn Chinese Treebank contain at least a subject relative clause or an object relative clause. Focus constructions, topicalizations, passivization with fronted indirect objects and extractions from embedded clauses occur relatively rarely in the corpus. It is possible that some constructions, such as the focus construction, can be more frequent in more colloquial text.

3. Experiments

We first examined the proportion of each test set that can be recovered from Stanford dependencies converted from gold Treebank trees to explore the possibility to evaluate nonlocal dependency recovery on an ideal parser producing Stanford dependencies. Then we evaluated the performance of several automatic parsers on the task of recovering nonlocal dependencies to have a preliminary understanding of how difficult it is to recover each of these nonlocal dependencies.

3.1. Stanford dependencies

Stanford dependencies (Marneffe et al., 2006; Chang et al., 2009) are widely used in many dependency parsing evaluations and can be easily obtained from Treebank annotations. Constituent parsers, such as the Stanford parser (Klein and Manning, 2003), the Berkeley parser (Petrov and Klein, 2007) and the Brown parser (Charniak, 2000), can all be trained on Treebank annotations to yield Stanford dependencies. Dependency parsers, such as MaltParser (Nivre et al., 2006), Mate (Bohnet, 2010) and MSTParser (MacDonal and Pereira, 2006), can be trained directly on Stanford dependencies to predict Stanford dependencies. Therefore we implemented a heuristic extraction of nonlocal dependencies from gold Stanford dependencies and evaluated the results against our annotations. Heuristically, we mapped 'nsubj' in Stanford dependencies into '1' dependencies in our annotations, 'dobj' into '2' dependencies and 'iobj' into a '3' dependencies.

For relative clauses, Chinese Stanford dependencies have a dependency labeled 'rcmod' between the head noun and the main verb of a relative clause, regardless of the type of the relative clause. For example, the Stanford dependencies of (2) will contain 'rcmod(news, do-not-have)' and the Stanford dependencies of (3) will contain 'rcmod(happiness, need)'. However, this dependency label does not provide any information about whether the head noun is the first or second argument of the verb. After examining the statistics of relative clauses in Stanford dependencies of the training set, we found that when the subject is present in the relative clause while the object is missing, the relative clause is most likely to be an object relative clause; when the subject is missing but the object is present, it is most likely to be a subject relative clause; and if both subject and object are absent, it is most likely to be a subject relative clause. By these principles, we mapped the rcmod dependencies into subject or object relative clause dependencies in our annotations. However, it is more difficult to recover nonlocal dependencies for topic relative clauses. For example, Stanford dependencies have 'rcmod(Welia, longer-than)' for (4), while 'Welia' is not a participant of 'longer-than'. Although we can map all the relative clauses where both the subject and object are present into topic relative clauses, our statistics show that relative clauses with both subject and object present are most likely to be a relative clause relativizing an adjective or adverbial modifier as shown in (1). Mapping the relative clause with both subject and object present into topic relative clauses is not supported by our data. Therefore, we cannot have labeled dependency results of topic relative clauses from gold Stanford dependencies (sd-orcl) in Table (2).

For passive voice, Stanford dependencies have a 'nsubjpass' dependency between the subject and the verb in a passivized sentence. In most cases, the subject is the second argument of the verb. Therefore, 'nsubjpass' dependencies are converted into '2' dependencies in our annotations.

Considering the possibility of mismatching the dependency labels between Stanford dependencies and our annotations, we also set up the evaluation of unlabeled dependencies to check whether pairs of words which have nonlocal dependencies in our annotations are related in any type of dependency in Stanford dependencies. The results of recovering nonlocal dependencies, either labeled (L) or unlabeled (U), from gold Stanford dependencies (sd-orcl) are shown in Table 2.

Table 2 shows that some nonlocal dependencies, even when unlabeled, are absent in the Stanford dependencies converted from gold Treebank trees. This could indicate

Construction	L/U	sd-orcl	gcg-s	gcg-l
Sbj rel	L	.81	.51	.50
	U	.81	.57	.61
Obj rel	L	.63	.63	.77
	U	.87	.70	.81
Tpc rel	L	-	.49	.48
	U	-	.53	.50
Focus	L	.28	.06	.33
	U	.97	.82	.80
Pass_do	L	.42	.83	.89
	U	.71	.89	.92
Pass_io	L	0	.60	.62
	U	.69	.74	.88
Topicalization	L	0	.08	.11
	U	.86	.59	.62
Embedded	L	.02	.38	.31
	U	.04	.42	.33

Table 2: Nonlocal dependency recovery results from gold Chinese Stanford dependencies (sd-orcl) and from two automatic parsers trained on generalized categorial grammar annotations of Mandarin Chinese (gcg-s and gcg-l).

that syntactic dependencies extracted from the Penn Treebank are not a close approximation of semantic predicate-argument structures extracted from categorial grammar annotations where filler-gap constructions are well defined in the syntactic derivations.

3.2. A GCG parser

To examine the performance of recovering nonlocal dependencies from an automatic parser, we experimented on parsers trained by the Berkeley latent variable PCFG trainer (Petrov and Klein, 2007) on generalized categorial grammar annotations (Bach, 1981; Nguyen et al., 2012; Duan and Schuler, 2015) of the Penn Chinese Treebank. We experimented with two different training sets, a small training set (gcg-s) and a large training set (gcg-l). The small training set is the same training set used in Tse and Curran (2010) which contains 15,957 sentences. The large training set includes all the sentences from the Penn Chinese Treebank 8 except those sentences used in nonlocal dependency test sets. The large training set contains 50,635 sentences. We trained two parsers with these two training sets and parsed the sentences in the nonlocal dependency test sets with these two parsers. Then we extracted the dependencies out of the parse outputs and evaluated the results against the dependencies annotated in the eight test sets.

4. Discussion

In general, a larger training set is beneficial for recovering nonlocal dependencies, as shown in Table 2. The improvements for object relative clauses and focus constructions are large. Small drops in accuracy are observed for subject relative clauses, topic relative clauses and extractions from an embedded clause. These usually only involve one or two more wrongly predicted dependencies on the side of the large training set.

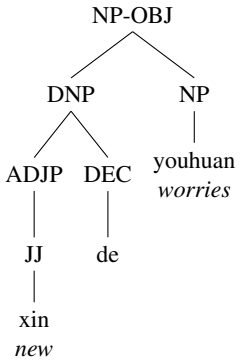


Figure 1: Treebank annotations for *xin de youhuan* 'new worries'

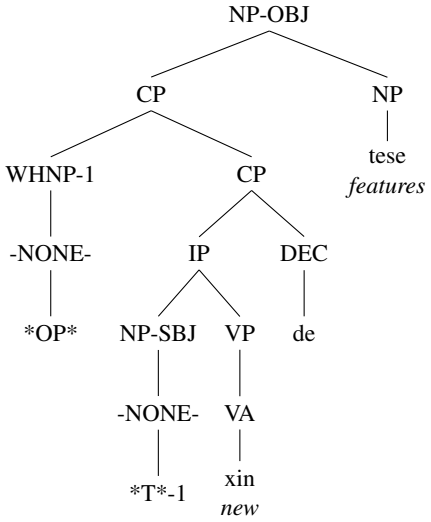


Figure 2: Treebank annotations for *xin de tese* 'new features'

In order to better inspect the parsing errors, we annotated a small development set for seven nonlocal constructions. We did not have a development set for extractions from embedded clauses because they rarely occur in the data. The development set for each construction consists of seven to ten sentences. The error analysis below is conducted based on the parsing outputs of the development sets.

In spite of their frequent occurrence in the corpus, subject relative clauses are not easy to parse correctly, as suggested by the results in Table 2. Examining the failed cases in the development set suggests two potential difficulties for parsing subject relative clauses correctly. The first difficulty is that there are inconsistent Treebank annotations for the same noun phrase construction, as shown in Figure 1 and 2. There is no apparent semantic or syntactic reason to analyze the noun phrases in Figure 1 and 2 differently. This confusion caused the parser to be unable to reliably predict the internal structure of this type of noun phrases.

Another mistake observed in parsing subject relative clauses is caused by noun-verb confusion in Chinese.

(10) shamao de xiansheng
silly de man
'a silly man'

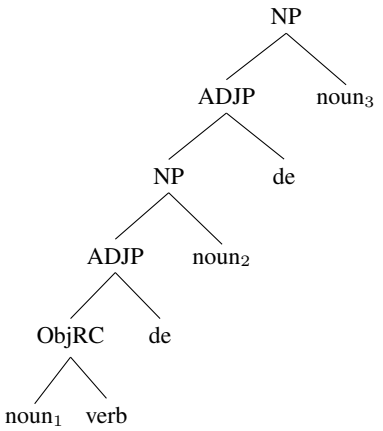


Figure 3: A possible parse for *noun + verb + de + noun + de + noun*

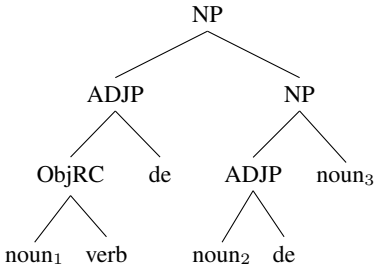


Figure 4: Another possible parse for *noun + verb + de + noun + de + noun*

Shamao 'silly' in Chinese can either be a verb *be silly* or a noun *a silly*. *Shamao* is annotated as a verb in (10) in Treebank annotations and *shamao de* forms a relative clause modifying *xiangsheng* 'man'. However, the gcg-l parser parses the word as a noun and the structure of the noun phrase is 'noun + de + noun', which is also a very common structure for noun phrases in Mandarin Chinese.

The errors in parsing object relative clauses are often caused by the difficulty of predicting correct internal structure for noun phrases. For example, a word sequence of 'noun$_1$ + verb + de + noun$_2$ + de + noun$_3$' is structurally ambiguous between two possible parses as shown in Figures 3 and 4. The parse in Figure 3 yields the dependency 2(verb, noun$_2$) from the object relative clause. The parse in Figure 4 yields the dependency 2(verb, noun$_3$).

The difficulty of parsing topic relative clauses correctly is that there are multiple ways to parse the word sequence of a topic relative clause. A topic relative clause usually has a word sequence of 'noun$_1$ + verb + (noun$_2$) + de + noun$_3$' where noun$_3$ is the topic of the relative clause and we expect to have the dependency of-asso(noun$_1$, noun$_3$). The parser can parse noun$_1$ as adverbial modifier and the rest as a subject relative clause and yield the dependency 1(verb, noun$_3$). If the noun$_2$ is not present, the parser is also likely to parse this word sequence as an object relative clause and yield the dependency 2(verb, noun$_3$). Since subject relative clauses and object relative clauses are both very frequent in the data, these two parsing possibilities are very competitive.

Table2 shows that the parsing of a passivization with a fronted direct object is comparatively easy. This is because the passivization particle *bei* gives an unequivocal indication of the predicate-argument structure of the sentence. Since all ten sentences in the development set are parsed correctly, we speculate that the errors in the test set come from mistaken parsing of some noun phrase involved in the construction.

The parsing of passivizations with fronted indirect objects tends to be more challenging. Error analysis of the development set seems to suggest the parsing confusion between the second and third argument of a ditransitive verb accounts for most of the errors. A sentence 'John gave me a book' can be expressed in Chinese with a passive voice either as 'I + bei + give + a book' or 'A book + bei + give + me'. When presented with the word order 'noun$_1$ + bei + verb + noun$_2$', the parser can have a hard time deciding whether noun$_1$ is the second argument (direct object) or the third argument (indirect object) of the verb.

Focus constructions usually come with the word order 'noun$_1$ + noun$_2$ + verb' where noun$_1$ is the subject and noun$_2$ is the fronted object. However noun$_1$ is often not present linguistically because Chinese is a pro-drop language. In that case, noun$_2$ is often parsed mistakenly as the subject. Even if both noun$_1$ and noun$_2$ are both present, there is also chance to parse either noun$_1$ or noun$_2$ as an adverbial modifier because of the relatively low frequency of focus constructions in the corpus.

Table 2 suggests that it is hard to correctly resolve the nonlocal dependencies in topicalizations and extractions from embedded clauses. The fact that these two constructions occur very rarely in the data definitely contributes to the difficulty. Also, unlike relative clauses and passivized sentences, there is no syntactic marker to indicate the possible predicate-argument structure. The distance between the head and the dependent in these two constructions can be very long. An extra noun phrase at the beginning of a sentence can be parsed as a modifier or an adverbial rather than an extracted argument. Correctly resolving the nonlocal dependencies in these two constructions is very challenging.

5. Conclusions

In this study we contribute several test sets for nonlocal dependencies occurring in various constructions in Mandarin Chinese. The poor match between gold Chinese Stanford dependencies and the nonlocal dependencies we annotated indicates that Stanford dependencies cannot provide sufficiently specific information for nonlocal dependencies. The preliminary parsing results suggest that resolving nonlocal dependencies can be a challenging task for some nonlocal constructions. In the future, we are going to experiment with more dependency parsers against the test sets. We hope the availability of these test sets can help to promote better performance of nonlocal dependency recovery in Mandarin Chinese.

6. Bibliographical References

Bach, E. (1981). Discontinuous constituents in generalized categorial grammars. *Proceedings of the Annual Meeting of the Northeast Linguistic Society (NELS)*, 11:1–12.

Bohnet, B. (2010). Top accurancy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, pages 89–97.

Chang, P.-C., Tseng, H., Jurafsky, D., and Manning, C. D. (2009). Discriminative Reordering with Chinese Grammatical Relations Features. In *Proceedings of the third Workshop on Syntax and Structure in Statistical Translation*, pages 51–59.

Charniak, E. (2000). A maximum-entropy inspired parser. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL'00)*, pages 132–139, Seattle, Washington.

Duan, M. and Schuler, W. (2015). Parsing Chinese with a Generative Categorial Grammar. In *Proceedings of Grammar Engineering Across Frameworks*, pages 25–32.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan.

Kummerfeld, J. K., Tse, D., Curran, J. R., and Klein, D. (2013). An empirical examination of challenges in chinese parsing. In *Proceedings of ACL'13*, pages 98–103, Sofia, Bulgaria.

MacDonal, R. and Pereira, F. (2006). Online learning of approximate dependency parsering algorithms. In *Proceedings of the 11th Conference of the European Chapter of the ACL (EACL 2006)*, pages 81–88.

Marneffe, M.-C. d., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC 2006*.

Nguyen, L., van Schijndel, M., and Schuler, W. (2012). Accurate unbounded dependency recovery using generalized categorial grammars. In *Proceedings of COLING '12*, pages 2125–2140, Mumbai, India.

Nivre, J., Hall, J., and Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing. In *In Proc. of LREC*, pages 2216–2219.

Petrov, S. and Klein, D. (2007). Improved inference for unlexicalized parsing. In *Proceedings of NAACL HLT 2007*, pages 404–411, Rochester, New York, April. Association for Computational Linguistics.

Rimell, L., Clark, S., and Steedman, M. (2009). Unbounded dependency recovery for parser evaluation. In *Proceedings of EMNLP 2009*, volume 2, pages 813–821.

Tse, D. and Curran, J. R. (2010). Chinese CCGbank: extracting CCG derivations from the penn chinese treebank. In *Proceedings of COLING '10*, pages 1083–1091.

Xue, N., Xian, F., Chiou, F.-D., and Palmer, M. (2005). The Penn Chinese Treebank: Phrase Structure annotation of a large corpus. *Natural Language Engineering*, 11:207–238.

Adding Syntactic Annotations to Flickr30k Entities Corpus for Multimodal Ambiguous Prepositional-Phrase Attachment Resolution

Sebastien Delecraz, Alexis Nasr, Frederic Bechet, Benoit Favre
Aix-Marseille Univ, Université de Toulon, CNRS, LIS
163 avenue de Luminy - Case 901
F-13288 Marseille - Cedex 9 - FRANCE
{firstname.lastname}@univ-amu.fr

Abstract

We propose in this paper to add to the captions of the *Flickr30k Entities* corpus some syntactic annotations in order to study the joint processing of image and language features for the Preposition-Phrase attachment disambiguation task. The annotation has been performed on the English version of the captions and automatically projected on their French and German translations.

Keywords: PP-attachments, Multimodal Corpus, Multilingual Corpus

1. Introduction

Joint processing of image and text is a very active area of research. It is studied mostly in the context of natural language generation, for example for generating a textual description of a video or an image. Recently open-domain language generation from images or videos received a lot of attention through the use of multimodal deep neural networks (Vinyals et al., 2015). Theses models build a unified representation for both image and language features and generate in an end-to-end process a text directly from an image, without an explicit representation (syntactic or semantic) of the text generated.

In this paper we propose to study joint image and language processing for language parsing rather than generation. The main idea of this study is to check if the use of visual features extracted from an image can be useful in order to disambiguate the linguistic analysis of a caption that describes the same image. To do so we propose a new framework for testing multimodal approaches on the specific task of ambiguous Prepositional-Phrase attachment (*PP-attachment*) resolution.

PP-attachments are known to be an important source of errors in parsing natural language. The main reason being that, in many cases, correct attachments cannot be predicted accurately based on pure syntactic considerations: their prediction ask for precise lexical co-occurrences. Such information is usually not found in treebanks that are limited in their size and therefore do not model many bi-lexical phenomena. Besides, disambiguation may ask for non linguistic knowledge which is not present in treebanks.

In this paper, we propose to create a corpus for supporting PP-attachment disambiguation research by combining textual and visual information. The contribution of this study is the selection and the manual annotation of a corpus of ambiguous PP-attachments from the multimodal corpus *Flickr30k Entities* (Plummer et al., 2017). A full parse of the sentences containing a hand-corrected PP-attachment, which is compatible with the manual attachment is also produced. In addition, we use MT alignments to transfer the annotations to French and German translations from the *Multi30k* corpus (Elliott et al., 2016; Elliott et al., 2017).



1. someone is holding out a punctured ball in front of a brown dog with a red collar .
2. A man holding out a deflated soccer ball to a gray dog .
3. The owner tries to hand a deflated ball to his dog .
4. Large gray dog being handed a white soccer ball .
5. A brown dog starring at a soccer ball .

Figure 1: Example of the *F30kE* annotations. The image is described with five captions, each annotated with entities. Entities that corefer with a visual element in the image are linked to the corresponding bounding box.

Finally, for every preposition manually attached, a set of possible attachment alternatives for use in a reranking system is produced.

2. Enriching the Flickr30k Entities Corpus with PP-Attachment Annotations

Corpora with joint annotation of image and text has recently become widely available. The corpus used in this work is the *Flickr30k Entities* (*F30kE*) (Plummer et al., 2017), an extension of the original *Flickr30k* dataset (Young et al., 2014). This corpus is composed of almost $32K$ images and, for each image, five captions describing the image have been produced. Besides, every object in the image that corresponds to a mention in the captions has been manually identified with a bounding box.

Preposition	Occ.	% Noun	% Verb	Dist.
in	4191	0.59	0.41	2.21
with	3018	0.59	0.41	2.47
for	1777	0.36	0.64	1.57
near	1452	0.65	0.35	1.69
through	1420	0.05	0.95	2.01
on	1359	0.26	0.74	2.03
next to	1342	0.08	0.92	3.23
from	1172	0.30	0.70	2.43
into	1123	0.08	0.92	2.32
over	941	0.39	0.61	2.38
by	890	0.10	0.90	2.60
at	720	0.20	0.80	2.75
of	700	0.97	0.03	1.03
around	589	0.18	0.82	2.26
in front of	570	0.11	0.89	3.72
under	544	0.18	0.82	3.29
behind	544	0.35	0.65	1.78
along	500	0.37	0.63	1.79
during	423	0.14	0.86	5.08
across	415	0.11	0.89	2.21
down	393	0.66	0.34	2.53
against	365	0.39	0.61	1.56
outside	356	0.38	0.62	2.28
towards	276	0.08	0.92	2.41
out of	252	0.13	0.87	2.08
beside	245	0.03	0.97	3.51
above	241	0.43	0.57	2.90
in the middle of	240	0.12	0.88	3.63
onto	210	0.08	0.92	2.34
outside of	206	0.12	0.88	3.30
inside	197	0.13	0.87	3.72
between	189	0.72	0.28	1.29
past	170	0.19	0.81	2.95
toward	167	0.34	0.66	1.53
on top of	166	0.14	0.86	2.93
like	159	0.34	0.66	1.32
among	142	0.37	0.63	2.13
after	126	0.10	0.90	3.12
away from	109	0.04	0.96	1.75
off	104	0.31	0.69	3.72
up	96	0.82	0.18	1.93
up to	86	0.12	0.88	2.35
before	71	0.10	0.90	4.14
atop	60	0.23	0.77	2.97
about	54	0.59	0.41	1.52
along with	54	0.15	0.85	4.98
underneath	49	0.18	0.82	3.16
without	46	0.54	0.46	2.98
out	42	0.29	0.71	4.43
at the top of	40	0.10	0.90	2.23
inside of	39	0.23	0.77	3.72
amongst	33	0.09	0.91	3.18
close to	33	0.15	0.85	4.42
upon	31	0.13	0.87	1.71
amidst	28	0.29	0.71	3.18
beneath	26	0.12	0.88	3.54
within	24	0.33	0.67	3.62
below	23	0.52	0.48	1.83
at the bottom of	22	0.18	0.82	2.86
amid	22	0.18	0.82	5.00
in between	18	0.22	0.78	2.89
up against	16	0.06	0.94	3.69
ahead of	14	0.00	1.00	1.71
together with	13	0.08	0.92	2.54
such as	13	0.38	0.62	5.31
besides	12	0.42	0.58	2.33
beyond	10	0.70	0.30	3.30
on the top of	10	0.10	0.90	2.70
while	10	0.20	0.80	6.40
near to	10	0.10	0.90	4.10
Total	29068	0.36	0.64	2.37

Table 1: Prepositions annotated with their occurrence number in the corpus and statistics about their attachment.

Bounding boxes and the mentions in the captions have been paired together via coreference links. A total of $244K$ such links have been annotated.

Furthermore, each mention in the captions has been categorized into eight coarse-grained conceptual types. These types are: people, body parts, animals, clothing, instruments, vehicles, scene, and other. One example of the corpus has been reproduced in Figure 1.

Captions in the *F30kE* corpus are annotated at the conceptual level, but no syntactic annotation is provided. Since our goal in this study is to evaluate several sets of multimodal features for the PP-attachment task, we needed to add such a level of annotation to the corpus.

We did not have the resources for manually annotating the whole *F30kE* caption corpus with syntactic annotations. Therefore we limited our effort to the manual annotation of PP-attachments in ambiguous contexts.

In order to select ambiguous PP-attachments we applied the following process: first the captions of *F30kE* were processed by a Part-Of-Speech tagger (Nasr et al., 2011); then a set of regular expressions on the POS labels were defined in order to select sentences that contain a preposition that can be attachment to more than one word; finally, the ambiguous prepositions have been manually attached to their correct syntactic governor.

Captions containing ambiguous PP-attachment have been identified using two simple rules: a preposition is considered ambiguous if it is preceded by at least two nouns or a verb and a noun, in other word, the captions must match one of the following regular expressions:

- `X* N X* N X* p X*`
- `X* V X* N X* p X*`

where `N` and `V` stand for the POS tags noun and verb, `X` stand for any POS tag and `p` is the target preposition. $22,800$ captions were selected this way, that correspond to $15,700$ different images. They constitute our *PP-Flickr* corpus. This corpus contains $29,068$ preposition occurrences that have been manually attached to their syntactic governor. In the manual annotation process information given to the annotator is limited to a caption, the target preposition which needs to be attach to its governor, and the corresponding image.

Table 1 presents an overview of some statistics on the corpus. For each preposition, ordered by occurrence, we give the rate of attachment to a noun and a verb and the average distance between the preposition and its governor. For readability, we did not display the 25 prepositions with less than 10 occurrences but the total counts are computed on the whole corpus. 64% of the prepositions occurrences are attach to a verb and the average distance between the preposition and its governor is 2.37 words. The ten more frequent prepositions represent 61.22% of the annotated corpus. For preposition with a highest average distance ($>$ 3 words) that represent 25 prepositions (*i.e.* 4497 attachments), the governor is a verb in 74% of cases.

3. A Multimodal Corpus for Syntactic Parse Reranking

The annotation process described in the previous section only provides links between a preposition and its governor. We added syntactic annotations to the corpus in order to propose a task of multimodal reranking of syntactic parsing hypotheses on the *F30kE* corpus.

The first step in this process was to parse all the captions manually annotated with PP-attachment with a transition-based dependency parser. The parser allows to set some dependencies, prior to parsing, and generate the most likely parse including these dependencies. In our case all the PP-attachments manually annotated were set, then the parser provided the best dependency parse compatible with these attachments. Although these automatic annotations cannot be considered as *gold* labels, they can be considered as fairly robust as the main source of ambiguities (and therefore errors) is neutralized.

Once the best syntactic parse for each caption of the corpus is produced, we apply a method for generating alternative PP-attachment sites: given a sentence S, a parse T for S and a target preposition p, we define a set G_p of candidate governors for p. The set G_p is initialized with g, the actual governor of p in the parse T. The following rules are then applied to T and new potential governors are added to G_p:

1. $N \leftarrow V \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N\}$
2. $N \leftarrow P \leftarrow V \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N\}$
3. $N' \leftarrow N \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N'\}$
4. $N' \leftarrow P \leftarrow N \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N'\}$
5. $N' \rightarrow X \rightarrow N \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N'\}$
6. $N \rightarrow N \rightarrow p \quad \Rightarrow G_p = G_p \cup \{N\}$
7. $V \rightarrow N \rightarrow p \quad \Rightarrow G_p = G_p \cup \{V\}$

These rules are inspired by the ideas of (Anguiano and Candito, 2011; Attardi and Ciaramita, 2007; Hall and Novák, 2005). For example, the rule 1 is interpreted as follows: if target preposition p has a verbal governor which has a noun N as a direct dependent, N is added as a candidate governor. In rule 2, if the target preposition p is dependent of a verb V, it can also be attached to any noun that is itself governed by another preposition attached to V. The application of all rules must meet a general condition which is that the tree produced must be projective. These rules have been designed in such a way that most possible governors are included in the set G_p. The application of these rules on the test set showed that in 92.28% on the cases, the correct governor is in G_p.

Given the sentence *a man throws a child into the air at a beach*, and target preposition *at* that the parser has attached to *child*, the two rules 4 and 7 apply, yielding $G_p = \{child,\ air,\ throws\}$

	man	*throws*	*child*	*into*	*air*	*at*
4.			N	P	$N*$	p
7.		$V*$	N			p

Thanks to this process we have now for each caption with an ambiguous PP-attachment a set of syntactic parses that differ only by the governor chosen for the target preposition. We can now introduce the task of *multimodal syntactic parsing reranking* that can take advantage of all the visual features available in the *F30kE* corpus for finding the correct PP-attachment among all the possible parses.

4. Baseline PP-attachment reranking

In order to calibrate future research, we provide baseline results for PP-attachment reranking. The task consists in selecting the correct attachment from a list of potential attachments. A classifier is first trained to detect incorrect PP-attachments, and then the highest scoring PP-attachment alternative is output.

The classifier used as baseline is the *Icsiboost* classifier (Favre et al., 2007). This *Adaboost* classifier is a combination of weak learners that learn a threshold for continuous features, and a binary indicator for discrete ones. Training minimizes the exponential loss function by greedily selecting the best classifier and re-weighing the training set to focus on misclassified examples. This classifier is a strong baseline as it performs feature selection and has been shown to perform well on a range of tasks.

The features used to train the classifiers are defined for a governor-preposition-dependent triplet.

- (P)reposition: lemma of preposition
- (T)ext: part-of-speech and lemma of governor, dependent, both, and syntactic function of preposition, distance between governor and preposition.
- Visual (C)oncepts: concept of governor, concept of dependent, concepts of both
- Visual (S)patial: the normalized distance between the top-left and bottom-right corners of the governor and dependent bounding boxes, the areas of those boxes, and the ratio between the areas of the governor and dependent boxes.

It is important to notice that the visual features in our study are limited to spatial information about bounding boxes and visual concepts. No image analysis of the content of the boxes is done since this level of information is covered by the *visual concept features* which attach to each box a concept tag related to its content.

Table 2 presents the accuracy of PP-attachment after correction according to the candidates provided in the corpus, with different feature set combinations. Adding conceptual features to textual features improves accuracy, however spatial features have no impact when used in conjunction with other feature sets.

Features	Test
Baseline	0.75
P+T	0.85
P+C	0.82
P+S	0.77
P+T+C	0.86
P+T+S	0.86
P+C+S	0.82
P+T+C+S	0.86

Table 2: Baseline PP-attachment reranking accuracy on the test set.

5. Multilingual Extension

We extend the corpus by taking advantage of the translations produced for the WMT shared task on multimodal machine translation (Elliott et al., 2017). That corpus contains $31,014$ German (Elliott et al., 2016) and French translations from the *Flickr30k* corpus created by professional translators (German) and crowd sourcing (French). One arbitrary caption was translated per image to both languages. Out of those $31K$ translations, $5,225$ captions overlap with the gold standard PP attachment annotations.

In order to transfer the annotations across languages, we align them with the `fast_align` program (Dyer et al., 2013) and merge forward and reverse alignments with the `grow-diag-final-and` heuristic. In order to improve the accuracy of the alignment, we concatenate the translated sentences from the *F30kE* corpus with the news commentary *Bitexts* and the *Freedict* bilingual dictionary. This process helps aligning common expressions by providing more evidence to the unsupervised algorithm. From the alignments, we propagate two types of informations. First, the *Flickr30k Entities* segments which include unique entity references and types. This propagation is available for the whole set of translated captions, and could be used for instance to train a phrase retrieval system in the images, a coreference tracking system or other type of systems exploiting the *Flickr30k Entities* data, but in French and German. The second type of information propagated is the PP-attachment gold standard. The mapping is performed by transferring the annotation of preposition, and the head word of the governing entity and the hypothesis generated by the baseline English parser, to the words they are aligned to in the target language. If the source language words are aligned to multiple words, we use the first word by word order in the sentence.

The resulting multilingual annotations are made available along with the rest of the corpus in order to foster parsing research in all of the three languages, as well as the interesting link between PP-attachment resolution and machine translation. The transfer results in $30K$ sentences with entity boundaries, types and identity in French and German, as well as $5,225$ sentences with gold PP attachment hypotheses and gold standard in those languages (Figure 2).

The quality of transfer is highly dependent on the quality of the automatic alignment, a known difficult problem, for which the error rate is typically around 30%. We analyzed a random sample of 100 alignments for the French subset,



EN Large furry dog [G walking] in the [H sand] [P near] large rocks .

FR Un gros chien poilu [G marchant] dans le [H sable] [P près de] gros rochers .

DE Großer Hund mit langem Fell [G läuft] in der [P Nähe] großer Felsen durch den [H Sand] .

Figure 2: Example of annotations transferred from English to French and German. [P] represents the target preposition, [H] is the baseline governor predicted by the parser for English, [G] represents the gold standard manually annotated. Note that entities and coreference links are also transferred while not depicted in this figure.

and manually annotated incorrect propagations. The errors can come from the preposition, its gold governor, or predicted governor not being aligned, or being aligned to a word with the wrong part-of-speech. Often multiword idiomatic expressions in either language, that are characterized by arbitrary head-words, usually result in wrong alignment (*to bike* $\rightarrow$ *faire du vélo*, where *bike* is aligned with *vélo*, which might also be due to the fact that *bike* can both be a verb and a noun). Another frequent problem is the preposition being aligned to the wrong preposition in the target language when the target sentence contains several prepositions.

The result of this hand analysis is that 21% of transfers in the sample are erroneous. The transfer quality could be improved by accounting for part of speech tags, for example by using a joint grammar of both languages to enforce constituent-level alignments. A better heuristic could also be devised for processing multiply aligned source words for which the choice of alignment is not always successful. Finally, enforcing that aligned entities should have the same semantic category could improve the confidence of the transfer. We have released all the tools used to generate the transferred annotations in hope that they can be extended to improve the final result.

6. Distributed Data

The annotated corpus is available at `https://gitlab.lis-lab.fr/sebastien.delecraz/pp-flickr.git`. The annotation are given in JSON format for the three languages (English, French and German). In addition, we provide the English corpus in

#	Word	Lemma		POS		Governor	Label	Entity ID	Gold Gov.
1	someone	someone	_	NN	_	2	SBJ	227018	_
2	is	be	_	VBZ	_	0	ROOT	_	_
3	holding	hold	_	VBG	_	2	VC	_	_
4	out	out	_	RP	_	3	PRT	_	_
5	a	a	_	DT	_	7	NMOD	227019	_
6	punctured	puncture	_	VBN	_	7	NMOD	227019	_
7	ball	ball	_	NN	_	3	OBJ	227019	_
8	in front of	in front of	_	IN	_	3	ADV	_	3
9	a	a	_	DT	_	11	NMOD	227017	_
10	brown	brown	_	JJ	_	11	NMOD	227017	_
11	dog	dog	_	NN	_	8	PMOD	227017	_
12	with	with	_	IN	_	11	NMOD	_	11
13	a	a	_	DT	_	15	NMOD	227021	_
14	red	red	_	JJ	_	15	NMOD	227021	_
15	collar	collar	_	NN	_	12	PMOD	227021	_
16	.	.	_	.	_	2	P	_	_

Table 3: Example an annotated sentence in CoNLL format

CoNLL format (Table 3). Columns one to eight correspond to the standard CONLL columns, column nine corresponds to entity ID in the *F30kE* and column ten indicates whether the dependency is hand-corrected, such as words 8 and 12, or not, which is the case for all other words.

7. Conclusion

We have proposed in this paper a corpus for supporting PP-attachment reranking research when attachments can be disambiguated with an image. The corpus was created by enriching the *Flickr30k Entities* corpus with $29,068$ PP-attachments, from $22,800$ captions describing $15,700$ images, manually resolved by looking at the images.
We provide a testbed for reranking attachments generated from a forced parse with the correct attachment and a set of rules. A baseline classifier using reference visual features (concepts and spatial relations), and textual features yields PP-attachment accuracy of 86% from an original accuracy of 75% form a standard transition-based parser. The corpus is enriched with multilingual annotations transfered to French and German through automatic alignment.

8. Acknowledgements

This work has been carried out thanks to the support of French DGA in partnership with Aix-Marseille University as part of the "Club des partenaires Défense".

9. Bibliographical References

Anguiano, E. H. and Candito, M. (2011). Parse correction with specialized models for difficult attachment types. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1222–1233. Association for Computational Linguistics.

Attardi, G. and Ciaramita, M. (2007). Tree revision learning for dependency parsing. In *HLT-NAACL*, pages 388–395.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics.

Elliott, D., Frank, S., Sima'an, K., and Specia, L. (2016). Multi30k: Multilingual english-german image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74.

Elliott, D., Frank, S., Barrault, L., Bougares, F., and Specia, L. (2017). Findings of the Second Shared Task on Multimodal Machine Translation and Multilingual Image Description. In *Proceedings of the Second Conference on Machine Translation*, Copenhagen, Denmark, September.

Favre, B., Hakkani-Tür, D., and Cuendet, S. (2007). Icsiboost. `http://code.google.come/p/icsiboost`.

Hall, K. and Novák, V. (2005). Corrective modeling for non-projective dependency parsing. In *Proceedings of the Ninth International Workshop on Parsing Technology*, pages 42–52. Association for Computational Linguistics.

Nasr, A., Béchet, F., Rey, J., Favre, B., and Le Roux, J. (2011). Macaon: An nlp tool suite for processing word lattices. *Proceedings of the ACL 2011 System Demonstration*, pages 86–91.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. (2017). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *International Journal of Computer Vision*, 123(1):74–93.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Analyzing Middle High German Syntax with RDF and SPARQL

Christian Chiarcos, Benjamin Kosmehl, Christian Fäth, Maria Sukhareva
Goethe-Universität Frankfurt
chiarcos|kosmehl|faeth|sukhareva@informatik.uni-frankfurt.de

Abstract

The paper presents technological foundations for an empirical study of Middle High German (MHG) syntax. We aim to analyze the diachronic changes of MHG syntax on the example of direct and indirect object alterations in the middle field. In the absence of syntactically annotated corpora, we provide a rule-based shallow parser and an enrichment pipeline with the purpose of quantitative evaluation of a qualitative hypothesis. We provide a publicaly available enrichment and annotation pipeline grounded. A technologically innovative aspect is the application of CoNLL-RDF and SPARQL Update for parsing.

Keywords: Middle High German, syntactic parsing, semantic enrichment, computational philology

1. Background

Middle High German (MHG, ISO 639-3 gmh) refers to a historical stage in the development of (High) German during the middle ages (1050-1350), a period with an extensive literature and a long-time subject of philological interest. Also in terms of linguistics, the MHG period is particularly interesting, as it allows to explore the development of German syntax (whereas most Old High German prose is translated from Latin, an extant body of original MHG literature survives).

The paper presents technological foundations for the empirical study of Middle High German (MHG) syntax, with a focus on the order of postverbal arguments. German is well-known for its relatively free word order (scrambling) in the middle field, but so far, the historical dimension of word order in German has not been studied on a broad-scale quantitative basis, but only on grounds of qualitative analysis of *sample* data. As an example, (Petrova, 2009) limited their analysis of Old High German syntax to sentences where word order deviates from the Latin original, and similarly, (Speyer, 2011) manually analyzed Middle High German sentences that comprised specific verb forms that had been identified beforehand.

At the time of writing, the lack of syntactically annotated corpora prohibits quantitative studies of MHG syntax. We thus complement an existing corpus that already contains morphosyntactic annotations with a deterministic rule-based parser. It should be noted that this parser has been designed for a specific research problem, and this is reflected by its disambiguation strategies.[1]We thus refer to our implementation as a 'chunker' rather than a 'parser': Its analyses are shallow in the sense that no sophisticated disambiguation strategies are applied, but default rules, only. The chunker is implemented in SPARQL Update (Buil Aranda et al., 2013). Using the CoNLL-RDF architecture (Chiarcos and Fäth, 2017), we read a CoNLL TSV file (or, data stream) with user-defined column labels, split it into sentences, and apply graph transformation rules to each sentence. The result can be rendered in CoNLL, again, or, alternatively, evaluated directly with SPARQL.

In addition, we want to explore selected determinants of word order variation in Middle High German. We thus provide a CoNLL-TSV-based enrichment pipeline that includes the chunker in addition to other annotators in order to assess the impact of, for example, semantic factors such as animacy on word order alternation. In particular, this allows to verify existing hypotheses established by traditional qualitative research in the philologies by means of quantitative methods. In the same vein, any insight or analysis obtained by automated annotations needs to be confirmed by qualitative methods: While quantitative analyses allow to process large amounts of data and thus have a greater potential to identify statistically significant patterns than qualitative studies, their methods are incapable of reaching a comparable level of accuracy.

German word order has been a widely discussed topic in linguistics and philology. Generally, German word order is described as verb-second with a relatively free order of verb arguments. However, this order is not arbitrary and despite of the fact that no unified theory of word order in German emerged, the scholars still agree on multi-factorial nature of order variation. With respect to variation of direct and indirect objects in the middle field[2], four main groups of factors have been identified: *type of referential expressions*, *syntactic*, *semantic* and *discourse* factors, which interact flexibly in the determination of word order preferences, thereby leading to the perception of relatively free word order. An important aspect in this regard is the study of diachronic developments of word order flexibility and its possible triggers. However, syntactically annotated corpora for early stages of German are currently not available,[3] and in particular, the great wealth of Middle High German literature and functional writing has only been tackled on the morphosyntactic level, most prominently in the Referenzkorpus Mittelhochdeutsch (Klein et al., 2016, ReM, https://www.linguistics.rub.de/rem/), on which we

[1] For example, we implement low prepositional phrase (PP) attachment because we study the word order of dative and accusative objects. Knowing that no PPs may occur between dative and accusative arguments simplifies subsequent queries.

[2] According to the topological model of German syntax, the finite verb and (optional) verbal particles constitute a 'bracket' around the syntactic core arguments of a main (and similarly, relative) clause, referred to as the 'middle field'.

[3] Notable exceptions such as the Early New High German corpus (https://enhgcorpus.wikispaces.com/) only include texts from the 16th c. onwards.

also rely.

2. Addressing MHG word order

This paper aims to facilitate the development of quantitative methods for the study of diachronic order variation in the Middle High German (MHG) middle field in order to complement established qualitative methods. Empirical qualitative methods have long been one of the core strategies of humanities research. Such methods involve an in-depth thorough study of small amount of data associated with a certain phenomenon and the scholar induces conclusions about the phenomenon based on the made observations. Qualitative methods are time-consuming and require manual work, thus, it is impossible to apply them to large amount of data. Therefore, it can be argued that the made observations lack statistical significance.

As opposed to qualitative methods, quantitative methods can be applied to analyze large amount of data. While quantitative methods lack scrupulousness that is inherent to qualitative methods, they can provide statistically significant evidence to support or reject the qualitative hypothesis. Applying qualitative methods to historical languages has only recently become possible after multiple digitized historical corpora were released. However, the study of scrambling in the MHG middle field is impossible using morphosyntactically annotated corpora alone, therefore we describe the enrichment of the existing ReM annotations with a rule-based chunker.

2.1. Topological fields and word order flexibility

The theory of topological fields models a German sentence by splitting it into three major coherent chunks: prefield, middle field and postfield. The field delimiters ('brackets'), are defined by the position of (different parts of) the predicate: With a compound predicate in a main clause, the finite verb (or auxiliary) represents the left bracket (LB), and the non-finite part (verbal particle, infinite verbs) represents the right bracket (RB) of the middle field. Example (1.a,b) illustrates middle field construction in modern German.

(1)

a. *Peter* *hat* ***das Buch*** ***dem Freund*** *gegeben*
Peter has.LB **the book.ACC** **the friend.DAT** gave.RB

b. *Peter* *hat* ***dem Freund*** ***das Buch*** *gegeben*
Peter has.LB **the friend.DAT** **the book.ACC** gave.RB

"Peter has given the book to the friend"

c. *si* *zeigete* *dem küninge* *den mandel*
she.NOM showed.LB the king.DAT the coat.ACC

"she showed the coat to the king"

Figure 1: Example Middle Field

In (1.a), the middle field brackets are formed by the auxiliary verb and the present participle which enclose the direct object (DO) and indirect object (IO) of the verb. Similarly, other types of compound predicates such as phrasal verbs, compound nominal and adjective predicates etc. can serve as middle field brackets.

The middle field in German would include all the non-clausal verbal arguments and adjuncts as well as all the non-clausal arguments and adjuncts of the elements in the middle field. The clausal arguments and adjuncts are placed in the postfield. While the German language has constraints on the placement of the topological field, the word order within the fields is relatively free. As shown in (1.a,b), direct object can precede or succeed the indirect object without affecting the meaning.

According to (Paul, 1918) the theory of a topological fields model can be applied to the analysis of Middle High German sentences as well, within certain limitations. An example taken directly form the ReM corpus is shown in (1.c) which will be used further on. It should be noted that in this example there is no right bracket as it is regarded as optional both in modern German as well as in Middle High German.

Traditionally, it is assumed that this level of word order flexibility (scrambling in the middle field) is a relatively archaic feature of German, and that only the relatively recent loss of morphological case contributed to its disappearance in English. By analogy, one would assume that word order flexbility in German either remained stable since its separation from Old English, or that it decreased since then (as case morphology was simplified since Old High German). However, (Speyer, 2011) reported an unexpected *increase* of word order flexibility for direct and indirect NP arguments since the middle ages. While the numbers he reported were not statistically significant, this is an observation that calls for verification. Speyer employed a data-driven, but qualitative methodology: Using a list of ditransitive verbs, he extracted attestations of these verbs from prose texts and manually analyzed the structure of the sentences that these verbs occurred in. In order to achieve statistically significant numbers, in order to identify factors that contribute to reordering preferences, but also in order to identify factors that may possibly have been confounded with Speyer's classification, we develop a rule-based shallow parser for Middle High German topological fields, building on a morphosyntactically annotated corpus.

2.2. Reference corpus Middle High German (ReM)

The ReM corpus used in the context of this paper was first published in December 2016 and consists of roughly 2.5 million tokens. It includes about 397 texts and text fragments of various genres (e.g. administrative texts, prose, poems, letters etc.) and covers several dialects of MHG. The corpus is annotated with MHG lemmata. For lemmata annotations the authors relied on the Lexer dictionary (Lexer, 1872 1878). The corpus also has morphological (gender, case etc.) and morphosyntactic annotations, i.e. Parts-Of-Speech (POS) tags. The morphosyntactic annotations are done in accordance with HiTS tagset (Dipper et

al., 2013) which was developed on the basis STTS tagset (Schiller et al., 1999) and adjusted towards the diachronic periods of German.
The ReM corpus does not provide annotations for topological fields nor syntactic arguments, but their respective building stones in parts-of-speech, morphosyntactic features and clause separators. We thus take these as a basis for our chunker, guided by grammars of Middle High German, native competence on (Modern) German and the expertise of language experts.

2.3. RDF-based parsing of topological fields

On a technical level, chunking is implemented as a rule-based graph transformation. Building on recent developments at the intersection of NLP and the Semantic Web community, we employ the recently proposed CoNLL-RDF framework (Chiarcos and Fäth, 2017).[4] CoNLL-RDF provides an isomorphic reconstruction of CoNLL data structures in RDF, and enables the application of SPARQL Update for the flexible querying and transformation of this data. The updated CoNLL-RDF data can then be accessed via a SPARQL end point or transformed back into a CoNLL format, fed into an RDF triple store or visualized in a compact, human-readable fashion. In a mode operating sentence-by-sentence, this also allows manipulations on data streams. Supported are all tab-separated-value (TSV) corpus formats following the one-word-per-line (OWPL) principle, with configurable labels and order of columns. This includes CoNLL-U, CoNLL-X, all other CoNLL TSV dialects, the CWB format, the Sketch Engine format, etc. Using Semantic Web formalisms allows us to develop a modular and slim architecture with highly reusable and portable components (SPARQL) embedded in a thriving technological ecosystem that provides a rich off-the-shelf technology stack that can subsequently be applied to the data.
CoNLL-RDF employs RDF (multi-)graphs, for which various serializations exist. In the following, we use Turtle (Beckett et al., 2014) which fundamentally builds on the notion of triples, i.e., the segmentation of a graph into pairs of nodes and their connecting edge: a subject URI ('node'), a property (relation, 'edge') URI, and an object URI (or value). Triples are separated by (`.`). Abbreviations include the prefix notation for URIs as well as the use of triple separators that allow to skip (keep) the subject (`;`) or subject and property (`,`). RDF data can be queried with SPARQL (Buil Aranda et al., 2013) which extends Turtle with query and update operators (`SELECT`, `INSERT`, `DELETE`), the introduction of variables (marked by `?`), filters, etc. As data representation and query language are closely tied to each other, it is relatively easy to develop parsing rules for existing data samples.
The chunking rules are applied deterministically and create structures in a bottom-up fashion, identifying NPs, PPs, verbal chunks, topological fields, clauses, and clausal juncture, respectively. As the resulting parse contains nonterminal nodes, we also provide a transformation of those nodes into newly established CoNLL "words". This functionality is useful for *visualizing parses* on grounds of an existing visualization for dependency syntax. For operating and querying parse trees, however, we recommend to operate diretly on the RDF, as this allows to address elements and paths directly with SPARQL without creating artificial 'word's and without compressing its information into a less interpretable string representation (such as the original rendering of PTB syntax since CoNLL 2005).

[4] `https://github.com/acoli-repo/conll-rdf`

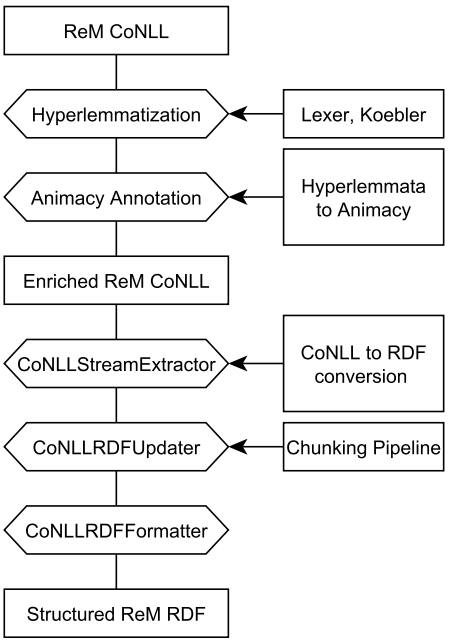


Figure 2: Enrichment pipeline

3. Annotation pipeline

This pipeline processes the files of the ReM corpus step-by-step from the raw data to a linguistically structured and annotated RDF format that can be easily queried. We ground our pipeline in the wide-used CoNLL format(s), i.e., as tab-separated values with one word per line, annotations separated by tabulators, and sentences separated by empty lines. We consider this a minimal and portable setting, as CoNLL is a relatively minimalistic and well-understood annotation format, but also, it can be easily customized for novel applications (by adding or dropping columns) and remains processable by both low-level shell commands and specialized tools. We provide an implementation of the pipeline as a simple Bash script which components are sketched out in Fig. 2. The code of the pipeline and its modules are available from `https://github.com/acoli-repo/germhist` under the Apache 2.0 license. An experimental UiMA implementation has been developed in addition but is currently not included in the release.

3.1. Corpus preprocessing

The ReM corpus is available in several formats, including CorA-XML (Bollmann et al., 2014).[5] We convert CorA-

[5] `https://cora.readthedocs.io/en/latest/coraxml/`

XML files into CoNLL with a focus on subsequent syntactic analysis: As such, we use the canonical ('modernized') version of a word as the value for the token (Klein and Dipper, 2016) instead of the original orthographical value. Additionally a number of other token-level features are pruned, as well. This includes, for example, morphological information such as the inflectional class. The resulting CoNLL files contain one token per line with the following columns: ID (token id), WORD (the canonical form of the written word), LEMMA, POS (parts-of-speech annotation), INFL (morphosyntactic features) and BOUNDARY (the clausal or sentence boundary marking). In accordance with the original annotation, sentences are separated by an empty line.

3.2. Enriching with hyperlemmas

Using CoNLL formats with user-defined column labels, it is possible to add novel components into the pipeline. At the moment, we support two types of enrichment modules, hyperlemmatization and animacy annotation.

By hyperlemmatization (or, more precisely, hyperlemmatization/translation), we mean to assign a historical word its modern counterpart. Ideally, this is a word that corresponds to the historical original both in meaning and etymology. However, depending on the strategy employed by the respective module, this may also be a translation, if a corresponding word cannot be found or if its meaning has changed.

We support two types of hyperlemmatization strategies: A wordlist-based approach and a transliteration-based approach. The wordlist-based approach is extensible to dictionaries, and thus capable to produce translations rather than hyperlemmas, it is, however, limited in coverage. The hyperlemmatization module thus uses its word list to train an internal transliterator, which is applied only if the lookup failed. Within the pipeline, the hyperlemmatizer is called several times for several word lists. We compiled the initial word list from Lexer (1872 1878), a 19th-century dictionary, by returning the head word together with the Levenshtein-closest word per gloss. However, this approach gave imprecise results: Frequently, glosses circumscribe a word rather than to provide its German equivalent. For multiword glosses, we thus added a similarity threshold for extraction. Moreover, we found that many glosses are in Latin rather than Modern German, and often not in line with modern orthography (for example, *wonung* for Modern *Wohnung*). Therefore, we consulted Köbler (2014) as a source of secondary evidence.

Hyperlemmatization modules can be run multiple times, with different word lists and in different configurations (e.g., for fuzzy search), each adding a column with hyperlemma candidates.

3.3. Enriching with animacy

Animacy is considered to be a major factor of word order, with the assumption that animate referents tend to precede inanimate referents (Jacobs, 1988) , and it can be relatively easily derived from lexical resources such as the Princeton WordNet (Fellbaum, 1998), resp., GermaNet (Hamp et al., 1997; Henrich and Hinrichs, 2010).

As WordNet is hierarchically organized, we retrieved the top-level synsets from WordNet 3.1 and (where appropriate) classified them for their animacy. We employ three primary classes: Human, Animate (non-human), Inanimate. Human includes persons, but also groups and organizations; Animate includes animals, plants and bacteria, but no plant or animal products; Inanimate includes substances and objects, but also abstractions.

For non-classified synsets, we increased the search depth and iterated the procedure. For verification, we ran animacy annotation against the entire WordNet and inspected 100 random samples for possible errors. After 5 iterations of the procedure, no more errors could be found. Animacy extraction from WordNet was implemented with a SPARQL SELECT statement against a local instance of WordNet 3.1. Animacy extraction for German was derived from WordNet using the existing GermaNet-WordNet linking.

In preparation for annotating textual data, we compiled a word list, where every German *lexeme* is described with its animacy classification. Note that we do not perform word sense disambiguation, so that a lexeme may have more than one animacy feature. As such, *Schwein* is animate ('pig'), inanimate ('pork') and human ('a person regarded as greedy and pig-like').

The actual animacy annotation reads hyperlemmatized ReM data from stdin, it takes a word list (TSV file) and a column number (the column of the hyperlemma) as parameters, and adds this information as another column. Again, animacy annotation can be run multiple times for different hyperlemma columns. If a TSV file for another feature is provided, this annotator can also be used for other kinds of lexical annotation.

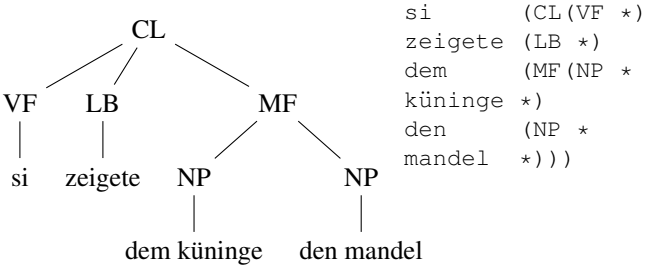


Figure 3: MHG sample parse (ReM, M403-G1, simplified) in tree view and conventional CoNLL

3.4. Annotation with CoNLL-RDF

CoNLL is an established exchange format in NLP, and enjoys high popularity as a representation formalism for dependency syntax, e.g., in the context of the Universal Dependencies (Marneffe et al., 2014) . For representing topological fields as part of a syntactic analysis, however, it is necessary to establish nonterminal nodes that span multiple words, and that are combined to form clauses and sentences. The conventional representation of CFG parses introduced with CoNLL 2005, however, requires to represent nonterminal nodes implicitly by pairs of matching brackets in different words: The word *si* in Fig. 3 thus carries the annotations of the prefield (VF) node as well as those of

its dominating clause (CL) node, whose right bracket only closes with the word *mandel*.

Processing phrasal annotations in CoNLL thus requires an internal mapping into a structured representation, e.g., a graph. For CoNLL data in general, such a mapping is provided by **CoNLL-RDF**, a data format and a library that provides the isomorphic and lossless reconstruction of CoNLL data as an RDF graph (Chiarcos and Fäth, 2017).[6]

This graph can then be extended with novel elements, but also, it is possible to perform rule-based graph transformation. Using the RDF query language SPARQL 1.1 (Buil Aranda et al., 2013), and in particular its update functions, CoNLL data can be enriched with additional structures and these can subsequently be transformed into CoNLL annotations.

Relevant features of CoNLL-RDF include:

- Assign every non-empty row a unique URI ('primary key') based on a user-provided base URI for the document, the sentence number and the word ID (or position): In the resource `file:M403-G1.conll`, the 26th word in the 59th first sentence will receive the URI `file:M403-G1.conll#s59.26`, resp., `:s59.26` in short.
- Define every row as a word, and connect it to its successor using the NIF vocabulary (Hellmann et al., 2013):[7] `:s59.26 a nif:Word; nif:nextWord :s59.27 .`
- Given a **user-provided list of column labels** (as an example: `LEMMA`), we create datatype properties in the `conll:` namespace, and assign the word its corresponding annotation as a literal value, e.g., `:s59.26 conll:LEMMA ``ër''.`

In consequence, we obtain an isomorphic representation of the original CoNLL data structure in RDF which is semantically shallow,[8] but can be effectively queried, manipulated and serialized back into CoNLL using off-the-shelf RDF technology. In particular, this includes a rich infrastructure of databases, webservices, APIs, models for resource publication and linking (Chiarcos et al., 2013). Even though it lacks formal semantics (by design), the CoNLL-RDF model can also serve as a basis to transform CoNLL data into semantically well-defined formalisms such as POWLA (Chiarcos, 2012) or NIF (Hellmann et al., 2013).

CoNLL-RDF comes with a Java API that allows to parse CoNLL data into CoNLL-RDF, to apply and to iterate SPARQL update transformations on this data, and to serialize `conll:` graphs in a lossless fashion as TSV (e.g., CoNLL-U or CoNLL-X), a human-readable dependency view or as a compact RDF/TTL representation that uses one word per line, ;-separated annotations and attribute-value pairs for different annotations. The latter serialization is also referred to as canonical CoNLL-RDF.

[6] `https://github.com/acoli-repo/conll-rdf`

[7] `http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core`

[8] The `conll:` namespace used here is not connected with any ontology, but populated by properties as defined by the user (column labels).

3.5. Querying with POWLA

POWLA (Chiarcos, 2012) is a small vocabulary that provides an OWL2/DL rendering of the Linguistic Annotation Framework (Ide and Suderman, 2014) . Here, we use the following vocabulary elements:

- `powla:Node` represents anything that can carry a linguistic annotation
- `powla:hasParent` points from a node to its parent node, thereby establishing a hierarchical structure
- children of the same `powla:Node` that are not interrupted by other siblings should be connected by a `powla:nextNode` property.

These data structures complement the original CoNLL-RDF rendering of (enriched) ReM CoNLL: The data structures that result from the chunking process (`conll:SHIFT` and `conll:REDUCE`, see below) are transformed with SPARQL Update into POWLA representations. Relations between siblings, ancestors, and descendants can thus be effectively queried and represent a basis for the quantitative evaluation with SPARQL: Using SPARQL SELECT, we can easily retrieve or count attestations of arbitrary graph patterns, including both CoNLL and POWLA data structures.

4. Shallow parsing

In this part we describe principles of the parsing process, we illustrate sample rules for a given sentence, and we provide a code example for the verbal chunking.

4.1. Parsing principles

Our chunker is designed to be a shallow parser in the sense that it does not attempt to disambiguate critical attachment decisions. Instead, deterministic default rules are applied that yield an analysis that is particularly convenient (but not limited to) the study of the order of arguments in the MHG middle field. One major limitation includes the treatment of PP attachment: As we are interested in the order of argument NPs, only, we perform low attachment. In this way, a PP positioned between a dative and an accusative NP will be attached to the preceding NP and both arguments will be adjacent siblings in the parse tree. Likewise, we regard genitive NPs (which can – rarely – have argument status in MHG) as nominal modifiers and treat them accordingly.

SPARQL by itself does support unrestricted graph transformation, albeit grounded in URI-defined properties and RDF resources. For implementing syntactic parsers, it is thus advisable to establish a designated vocabulary to represent data structures required during the parsing process. As a rule of thumb, however, 'data structures' refers to relations (object properties) between annotation elements, not to collections of partial parses. While this approach is qualitatively different from conventional parsing, we adopt the terminology of classical Shift-Reduce parsing (Nivre et al., 2007, 100-104): We introduce the properties `conll:SHIFT` to connect (the root nodes of) adjacent partial parses, and `conll:REDUCE` to represent attachment within a (partial) parse.[9]

[9] Originally, 'shift' and 'reduce' refer to parsing operations.

```
#ID WORD    LEMMA    POS   INFL
26  si      ër       PPER  Fem.Nom.Sg.3
27  zeigete zèigen   VVFIN *.Past.Sg.3
28  dem     dër      DDART Masc.Dat.Sg
29  küninge küni(n)g NA    Dat.Sg
30  den     dër      DDART Masc.Akk.Sg
31  mandel  mantel   NA    Akk.Sg
```

Figure 4: ReM CoNLL sample (ReM, M403-G1, slightly simplified)

Parsing is initialized by adding a `conll:SHIFT` relation for every `nif:nextWord` property in the graph, i.e., the 'queue' of unparsed words corresponds to the sequence of words. Parsing rules modify the existing graph, and if an attachment rule applies for a word/partial parse X, it is removed from the 'queue' of words (which is no longer distinguished from the 'stack' of partial parses) by dropping its `conll:SHIFT` relations. Instead, a `conll:REDUCE` relation with its head is established, and the sequence of `conll:SHIFT`s is restored by connecting the root of the partial parse with the root of the preceding partial parse. After initialization, `SHIFT`s over clause boundaries are removed (and restored later on from `nif:nextWord` for clausal juncture).

In the implementation here, parsing is deterministic and greedy. As RDF graphs are unordered, parsing is not conducted in a sequential fashion, but bottom-up and simultaneously for all matching graph patterns. All manipulations are expressed as SPARQL Update statements, which are applied and iterated in a predefined order until no more transformations occur, i.e., because a single root for the sentence has been established.

In the following chapters we describe selected parsing steps and rules using the example sentence (1.c) given above. Fig. 4 shows the resp. CoNLL representation. We represent parsing rules in SPARQL, but omit `SHIFT` updates. In the tree visualizations, vertical edges represent `REDUCE`, neighboring root nodes of partial parses (or words) are connected by `SHIFT`.

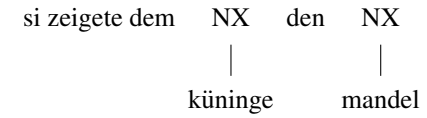
Furthermore, annotations are restructured: `conll:UPOS` provides the UD part-of-speech for the ReM part-of-speech, and separate properties `conll:CASE`, `conll:NUMBER` and `conll:FIN`(iteness) are extrapolated from `conll:INFL`.

4.2. Nominal, prepositional and verbal chunks

In a first step, noun chunks (NX) are formed for every token marked as nominal (noun or nominalization) if

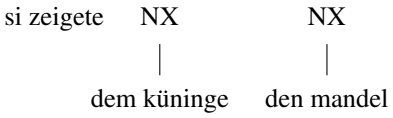
- it is the last word in a sentence,
- the next word is not a nominal, or
- the next word differs in case or number

The second case is illustrated in Fig. 5. In the example, the noun *küninge* projects an NX because the next word is a demonstrative with a different case marking, *mandel*




```
INSERT { _:nx conll:CAT    "NX";
              conll:CASE   ?case;
              conll:NUM    ?num.
          ?n  conll:REDUCE _:nx. }
WHERE {   ?n  conll:UPOS   "NOUN";
              conll:SHIFT/conll:UPOS ?p
              FILTER(?p!="NOUN").
          ?n  conll:CASE   ?case;
              conll:NUMBER ?num.}
```

Figure 5: Chunking: NX creation




```
INSERT { ?x conll:REDUCE ?nx. }
WHERE {  ?x conll:SHIFT  ?nx;
            conll:CASE   ?case;
            conll:NUMBER ?num.
         ?nx conll:CASE   ?case;
            conll:NUMBER ?num. }
```

Figure 6: Chunking: NX expansion

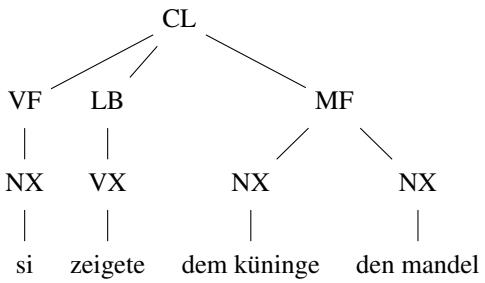


Figure 7: Topological fields and CL node

projects an NX because it precedes a punctuation sign. After the sequence of `SHIFT`s is restored, the next chunker rule does apply.

Subsequently, NX nodes are extended to the left by preceding words that match in case and number:[10] In Fig. 6, both determiners have the same case (`Dat`, `Akk`) and number (`Sg`) as the chunk they precede (resp. its head, cf. Fig. 4), and the NX is thus extended. This rule is iterated.

Furthermore, another rule creates NX elements for (unattached) pronouns.

Building on noun chunks, prepositional chunks are created when a preposition stands directly in front of an NX. The preposition and the noun chunk are joined into the new prepositional chunk (PX). We implement low attachment: The resulting PX nodes are attached to any immediately

Here, both terms refer to the data structures consulted/generated during these operations.

[10] Additional rules do exist for other (less frequent) constructions, e.g., attributive adjectives that follow their head noun.

preceding NX node.

Similar to nominal chunks, verb chunks (VX) are created from consecutive sequences of verbs and selected particles (e.g., ReM POS PTKVZ). If a verb carries the finiteness feature (conll:FIN ``true'', inferred from the ReM POS tag, e.g., from VV**FIN**), this is propagated to the VX. In Fig. 7, the verb *zeigete* is identified as a verbal chunk.

4.3. Topological fields and clausal junction

Middle field detection is relatively complicated and requires specialized rules for main and relative clauses as well as a special handling of discontinuous clauses. The most essential rules are the following: As a first step, left and right bracket (LB and RB) are detected. For main clauses, LB is a finite VX, and RB is a verbal particle or the next non-finite VX. We define the middle field (MF) as the sequence of chunks between a LB and an RB, resp., the end of the clause. The prefield (VF, for German *Vorfeld*) is a NX, PX or adverb immediately preceding the LB, the pre-prefield (VVF) is created for a conjunction preceding VF or LB, the postfield (NF, for German *Nachfeld*) is the span of chunks between RB and the end of the clause. Figure 7 illustrates the application of these rules to the example clause. For every LB, the preceding (optional) VVF and VF, and the following (optional) MF, RB and NF are then conjoined in a clause (CL) node, yielding a tree structure akin to Fig. 3. These rules (and a similar rule set for relative clauses) have been developed and tested on sample sentences from the ReM corpus.

It should be noted that these rules, as formulated so far, succeed only for continuous clauses. In order to handle clause fragments separated by a dependent relative clause, conll:SHIFT transitions over clause boundaries are restored from nif:nextWord, and a rule is applied that attaches the relative clause to the last NX. It should be noted that this rule exceeds the shift-reduce approach described so far in that we dive into the structure of the preceding clause: It does not attach to the root of the last parse fragment, but to its last NX descendant (possibly across a RB node, thereby producing a non-projective tree). In SPARQL, this is possible because the data structure can be traversed in the same way as SHIFT and REDUCE relations. All NX (?nx) chunks from the parse fragment that precedes a relative clause ?relCL can be retrieved by the first 3 lines in Fig. 8, and ?nx is the last NX chunk if (any nif:Word in) ?nx is not (MINUS) followed by any (nif:Word in another) NX chunk ?nx2 in ?last.

After attachment, clause fragments formerly separated by ?last can be conjoined and processed as above. Again, this rule does not perform disambiguation but it implements a low attachment strategy. The example illustrates how SPARQL can elegantly exceed beyond the local context, as well as some more advanced SPARQL expressions, like iterated (⋆, +) and concantenated (/) properties, as well as an example for FILTERs and set operators (MINUS).

Given this degree of expressivity, it is not surprising to find that SPARQL can be successfully employed to perform parsing using off-the-shelf Semantic Web technologies.

tokens (nif:Word)	2,514,585
sentences (conll:CAT "S")	147,398
middle fields (clauses) (conll:CAT "MF")	224,820
Acc before Dat	3,498
Dat before Acc	8,197

Table 1: Parsing statistics for the ReM corpus

5. Application and evaluation

After parsing, the result is transformed via SPARQL Update into POWLA data structures as illustrated in Fig. 10. Instances of dative and accusative arguments in the middle field can now easily be retrieved (and likewise, counted) by a query such as

```
SELECT ?acc ?dat
WHERE {
 ?acc conll:CASE "Akk".  # (an NX with) Akk
 ?acc powla:hasParent/conll:CAT "MF".
                       # in the middle-field
 ?acc powla:nextNode+ ?dat. # that precedes
 ?dat conll:CASE "Dat".  # (an NX with) Dat
}
```

Similarly, animacy features of nouns covered by the NX nodes can be counted, as well.

Without gold annotated data, we evaluate the parser only with respect to coverage as summarized in Tab. 1. A qualitative evaluation assessing the precision of middle field argument extraction is currently being conducted. In total, we obtained frequencies as shown in Fig. 9.

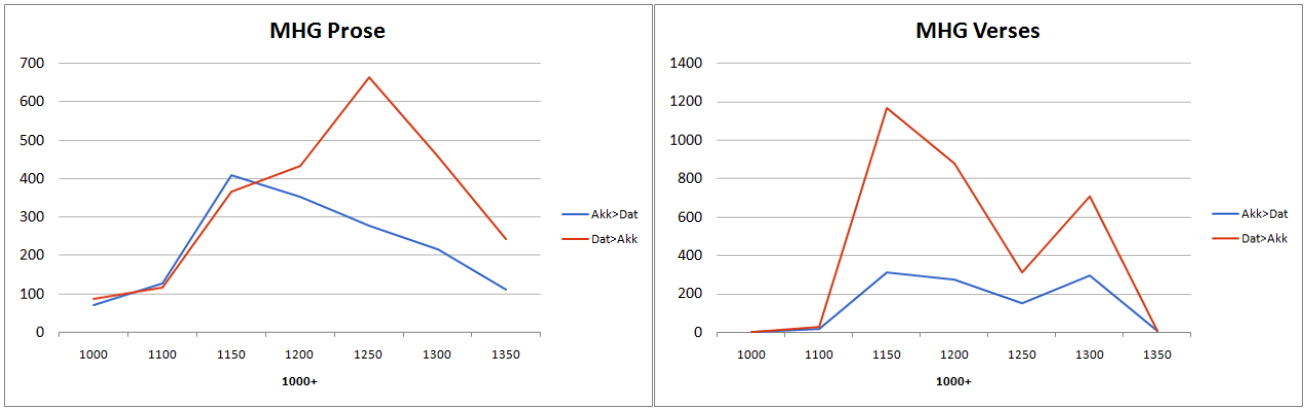
Above, we mentioned Speyer's qualitative experiments (Speyer, 2011) which indicated an increase of word order flexibility since the middle ages therefore contradicting the general scientific consensus. As a quantitative control experiment we thus developed a MHG shallow parser, and calculated the relative frequency of accusative and dative arguments in their relative order for all 50-year periods in the Middle High German era (1050 - 1350). A detailed linguistic analysis of these results, as well as a qualitative evaluation of the accuracy of argument and middle field detection is currently being conducted. In this paper, we focus on providing the technical pre-requisits for such a study, i.e., syntactic annotations, a convenient query language, as well as a workflow for enrichment with lexical (hyperlemmas) and semantic (animacy) features whose impact on diachronic word order variation is to be studied along with other shallow semantic annotations.

At a first glance, the results on prose text as shown in Fig. 9 do indeed seem to conform with the scientific consensus, i.e., that we see a decrease of word order flexibility during the middle ages: Until 1150, we find DO (direct object, accusative) > IO (indirect object, dative) about as often as IO > DO. After 1200, IO > DO is relatively more frequent, with a peak around 1250. The apparent decrease afterwards is due to the number of total attestations (i.e., texts from these periods). In verse, we always see a dominance of IO > DO, the reasons are not well understood, but we can expect interference with rhyme and meter.

This analysis is yet to be extended to Early Modern High German in order to verify (or refute) Speyer's thesis.

```
?last conll:SHIFT ?relCL.      # ?last directly precedes ?relCL
?nx conll:REDUCE* ?last.       # ?nx is a descendant of ?last
?nx conll:CAT "NX".            # and it is a NX
MINUS {                              # exclude all matches with
  ?nxWord conll:REDUCE+ ?nx.                 #  any word in ?nx
  ?nxWord nif:nextWord+/conll:REDUCE+ ?nx2. # followed by ?nx2
  ?nx2 conll:CAT "NX". FILTER(?nx2 != ?nx)  # i.e., another NX
  ?nx2 conll:REDUCE* ?last. }           # from the same clause
```

Figure 8: Advanced SPARQL: Identifying the last NX in the preceding partial parse



(a) Distribution of direct and indirect objects in MHG prose

(b) Distribution of direct and indirect objects in MHG verses

Figure 9: Diachronic quantitative analysis of the word order of direct and indirect objects in MHG

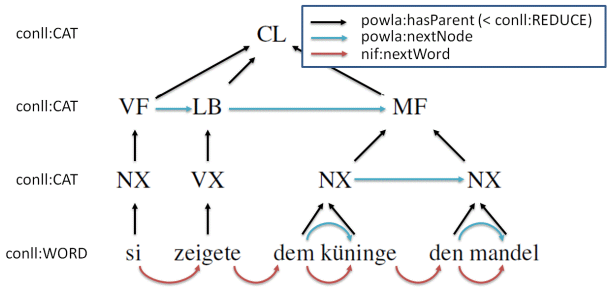


Figure 10: Resulting POWLA RDF graph

6. Summary and conclusion

We describe a pipeline for the syntactic annotation and the semantic enrichment of Middle High German. To our best knowledge, NLP for Middle High German consists of early prototypes towards morphosyntactic annotation (Hinrichs and Zastrow, 2012; Schulz and Kuhn, 2016). For more abstract levels, however, we are not aware of any attempts to conduct automated syntactic or semantic annotation on Middle High German.

Our approach builds on two core formalisms, the CoNLL format (resp., a specific dialect), and RDF. In general, pipeline modules communicate via CoNLL, resp. a TSV format, however, this seamlessly integrates with the SPARQL-based extraction of semantic features from WordNet 3.1 (i.e., a SPARQL SELECT query which produces TSV data) and with the SPARQL-based syntactic annotation (building on CoNLL-RDF). The resulting POWLA RDF data structure can be conveniently queried using SPARQL SELECT.

With respect to syntactic parsing we provide – to our best knowledge – the first publicly available implementation of a parser which solely relies on off-the-shelf Semantic Web technology. Related research includes the application of RDF and OWL for corpus representation and querying (Burchardt et al., 2008; Chiarcos, 2012) as well as a backend formalism for manual dependency annotation (Mazziotta, 2010). The only experiment on automated natural language parsing we are aware of (Wilcock, 2007), differs greatly by design from our implementation. Unfortunately, this implementation never left an experimental stage (p.c. G. Wilcock, Sep 2015). This experiment heavily relied on OWL/DL reasoning, resp., the use of rule languages building on top of OWL (Wilcock, 2006), and was thus relatively resource-intense. In comparison, our approach is designed to perform shallow, fast and transparent graph transformations using a formalism (CoNLL-RDF) that allows its integration in existing NLP pipelines. Its modular structure allows simple and comfortable integration of additional rules implemented as SPARQL updates.

In summary, we report the development of a shell-based enrichment pipeline for Middle High German including a CoNLL-RDF-based chunker for the analysis of Middle High German syntax and its semantic determinants. Both efforts improve the state of the art in natural language processing on Middle High German, and in terms of the technology applied, also for the processing of historical and low resource languages in general.

Acknowledgments

The research described in this paper was conducted at the Goethe Universität Frankfurt, Germany, within a project on Quantitative and Qualitative Methods in German Historical Philology (QuantQual@CEDIFOR), at the Centre for the Digital Foundation of Research in the Humanities, Social, and Educational Sciences (CEDIFOR) [11], funded by the German Ministry of Research and Education (BMBF, first phase 2014-2017). We would like to thank Ralf Plate, Arbeitsstelle Mittelhochdeutsches Wörterbuch, Trier / Institut für Empirische Sprachwissenschaft, Goethe-Universität Frankfurt, for the fruitful collaboration within this project. Furthermore, we would like to thank Thomas Klein and Claudia Wich-Reif for providing us with an access to the ReM corpus even before its ultimate publication, as well as Thomas Burch and the Trier Center for Digital Humanities [12], for the access to the digital Lexer[13] dictionary data. We would like to thank Margarete Springeth for access to the Middle High German Conceptual Database (MHDBDB)[14] at the Universität Salzburg. While not reported here, our hyperlemmatization routine was also applied to produce an annotation with MHDBDB concepts. Finally, we thank the anonymous reviewers for helpful comments and feedback.

7. References

Beckett, D., Berners-Lee, T., Prud'hommeaux, E., and Carothers, G. (2014). RDF 1.1 turtle. Technical report, W3C Recommendation.

Bollmann, M., Petran, F., Dipper, S., and Krasselt, J. (2014). Cora: A web-based annotation tool for historical and other non-standard language data. In *Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 86–90, Gothenburg, Sweden.

Buil Aranda, C., Corby, O., Das, S., Feigenbaum, L., Gearon, P., Glimm, B., Harris, S., Hawke, S., Herman, I., Humfrey, N., Michaelis, N., Ogbuji, C., Perry, M., Passant, A., Polleres, A., Prud'hommeaux, E., Seaborne, A., and Williams, G. (2013). SPARQL 1.1 overview. Technical report, W3C Recommendation.

Burchardt, A., Padó, S., Spohr, D., Frank, A., and Heid, U. (2008). Formalising multi-layer corpora in OWL/DL - lexicon modelling, querying and consistency control. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP-2008)*, pages 389–396, Hyderabad, India, January.

Chiarcos, C. and Fäth, C. (2017). Conll-rdf: Linked corpora done in an nlp-friendly way. In Jorge Gracia, et al., editors, *Language, Data, and Knowledge: First International Conference, LDK 2017, Galway, Ireland, June 19-20, 2017, Proceedings*, pages 74–88. Springer International Publishing, Cham.

Chiarcos, C., McCrae, J., Cimiano, P., and Fellbaum, C. (2013). Towards open data for linguistics: Linguistic linked data. In Alessandro Oltramari, et al., editors, *New Trends of Research in Ontologies and Lexical Resources: Ideas, Projects, Systems*, pages 7–25. Springer Berlin Heidelberg, Berlin, Heidelberg.

Chiarcos, C. (2012). A generic formalism to represent linguistic corpora in rdf and owl/dl. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Dipper, S., Donhauser, K., Klein, T., Linde, S., Müller, S., and Wegera, K.-P. (2013). Hits: ein tagset für historische sprachstufen des deutschen. *Journal for Language Technology and Computational Linguistics*, 28(1):85–137. Special Issue.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

Hamp, B., Feldweg, H., et al. (1997). Germanet-a lexical-semantic net for german. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Hellmann, S., Lehmann, J., Auer, S., and Brümmer, M. (2013). Integrating NLP using linked data. In *Proc. of the 12th International Semantic Web Conference (ISWC-2013)*, pages 98–113, Sydney, Australia, Oct. Springer.

Henrich, V. and Hinrichs, E. (2010). GernEdiT - The GermaNet Editing Tool. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC-2010)*, pages 2228–2235, Valletta, Malta, May.

Hinrichs, E. and Zastrow, T. (2012). Automatic annotation and manual evaluation of the diachronic german corpus tüba-d/dc. In *Proceedings of the Eigth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 1622–1627, Istanbul, Turkey, May.

Ide, N. and Suderman, K. (2014). The linguistic annotation framework: a standard for annotation interchange and merging. *Language Resources and Evaluation*, 48(3):395–418, Sep.

Jacobs, J. (1988). Probleme der freien wortstellung im deutschen. *Arbeitsbericht Nr. 5 des Forschungsprogramms Sprache und Pragmatik*, pages 8–37.

Klein, T. and Dipper, S. (2016). *Handbuch zum Referenzkorpus Mittelhochdeutsch*. Ruhr-Universität Bochum. Bochumer Linguistische Arbeitsberichte; 19.

Klein, T., Wegera, K.-P., Dipper, S., and Wich-Reif, C. (2016). Referenzkorpus Mittelhochdeutsch (1050–1350), Version 1.0. `https://www.linguistics.ruhr-uni-bochum.de/rem/`. ISLRN 332-536-136-099-5.

Köbler, G. (2014). Mittelhochdeutsches Wörterbuch. `http://www.koeblergerhard.de/mhd/2A/mhd.html`, last visited on 07/02/2018. 3rd ed.

Lexer, M. (1872-1878). *Mittelhochdeutsches Handwörterbuch*. S. Hirzel, Leipzig.

Marneffe, M.-C. D., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal Stanford dependencies: a cross-linguistic typology. In *Proceedings of the Ninth International Conference*

[11] `https://www.cedifor.de/en/`
[12] `http://kompetenzzentrum.uni-trier.de/de/`
[13] `http://woerterbuchnetz.de/Lexer/`
[14] `http://mhdbdb.sbg.ac.at/`

on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Mazziotta, N. (2010). Building the syntactic reference corpus of medieval french using notabene rdf annotation tool. In *Proceedings of the Fourth Linguistic Annotation Workshop (LAW-2010)*, pages 142–146. Association for Computational Linguistics.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryiğit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95–135, 06.

Paul, H. (1918). *Mittelhochdeutsche Grammatik*, volume 2 of *Sammlung kurzer Grammatiken germanischer Dialekte. A, Hauptreihe*. Niemeyer Tübingen. reprint 2007, 25. edition.

Petrova, S. (2009). Information structure and word order variation in the Old High German Tatian. *Information Structure and Language Change : New Approaches to Word Order Variation in Germanic*, pages 251–280.

Schiller, A., Teufel, S., Stöckert, C., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (kleines und großes Tagset). Technical report, Universität Stuttgart, Universität Tübingen, Stuttgart, Germany.

Schulz, S. and Kuhn, J. (2016). Learning from within? comparing pos tagging approaches for historical text. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Slovenia, May. European Language Resources Association (ELRA).

Speyer, A. (2011). Die Freiheit der Mittelfeldabfolge im Deutschen. Ein modernes Phänomen. *Beiträge zur Geschichte der deutschen Sprache und Literatur (PBB)*, 133(1):14–31.

Wilcock, G. (2006). Natural language parsing with GOLD and SWRL. In *Proceedings of the Second International Conference on Rules and Rule Markup Languages for the Semantic Web (RuleML 2007)*, Athens, Georgia, November. `http://2006.ruleml.org/group3.html#3`.

Wilcock, G. (2007). An OWL ontology for HPSG. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL-2007)*, pages 169–172, Prague, Czech Republic. Association for Computational Linguistics.

Cheating a Parser to Death: Data-driven Cross-Treebank Annotation Transfer

Djamé Seddah[1,2] Eric de la Clergerie[2] Benoît Sagot[2]
Héctor Martínez Alonso [4] Marie Candito[3]

[1]Université Paris-Sorbonne [2]Inria [3]Université Denis Diderot & CNRS [4] Thomson Reuters Lab
djame.seddah@paris-sorbonne.fr, {eric.de_la_clergerie, benoît.sagot}@inria.fr,
hector.martinezalonso@thomsonreuters.com, marie.candito@linguist.univ-paris-diderot.fr

Abstract

We present an efficient and accurate method for transferring annotations between two different treebanks of the same language. This method led to the creation of a new instance of the French Treebank (Abeillé et al., 2003), which follows the Universal Dependency annotation scheme and which was proposed to the participants of the CoNLL 2017 Universal Dependency parsing shared task (Zeman et al., 2017). Strong results from an evaluation on our gold standard (94.75% of LAS, 99.40% UAS on the test set) demonstrate the quality of this new annotated data set and validate our approach.

Keywords: Treebanking, Universal Dependencies, Syntax, Automatic Correction, Cross-annotation Transfer

1. Introduction

After many decades of treebanking initiatives (Einarsson, 1976; Marcus et al., 1993), the interest in developing annotated corpora no longer needs to be justified. Although a distinction can be noted between treebanks created for linguistic purposes and those only conceived in a natural language processing perspective, it tends to fade away in the face of the ever growing machine learning addiction to new sources of labeled data. In fact, not only can any annotated corpus be used as a primary or secondary source of training data within more or less complex systems, but hand-crafted syntactic resources such as grammars and lexicons can be used as sources of features to guide data driven systems (Øvrelid et al., 2009; Villemonte De La Clergerie, 2014a). The crucial point here lies in the interopability of such heterogenous sources of information. Before the rise of the Universal Dependency initiative (Nivre et al., 2017) and its eponymous scheme, henceforth UD, which resulted in the release of 81 treebanks on more than 50 language, the situation was at best complicated. Nevertheless, the pre-UD multitude of annotation schemes allowed many to use stacking methodologies for predicting syntactic annotations of a certain type and following specific guidelines (e.g. UD dependencies) with the help of other types of annotations that follow different schemes, sometimes even of a different topological nature (Farkas and Bohnet, 2012; Björkelund et al., 2013; Ambati et al., 2013; Ribeyre et al., 2015). In most cases, taking into account such heterogenous syntactic information in the form of additional features does improve parsing accuracy.

Unsurprisingly, the performance gain is generally outstanding whenever such features are extracted from gold annotations. When the goal is to produce new reference annotated data, such an performance gain results in fewer post-annotation corrections. In case of converting one treebank to another annotation scheme, such gold information is of course readily available and has the potential to considerably ease this process.

In this paper, we describe such a conversion effort, for which we had to meet with another drastic constraint; in the context of the preparation of the CoNLL 2017 shared task on "Multilingual Parsing from Raw Text to Universal Dependencies" (Zeman et al., 2017), we had less than two weeks for converting the French Treebank (Abeillé et al., 2003, hereafter FTB) in its SPMRL[1] dependency version (Seddah et al., 2013) into a new one that complies with the UD guidelines.

Such an objective forced us to think of all possible techniques that could help producing a treebank that would follow the UD scheme with the best possible accuracy. Since we were to produce a new data set, the use of a data-driven process fed with gold features whenever possible was the only way out. The result of our conversion process, as measured on a silver standard in terms of labeled attachment accuracy (LAS), reaches around 98.50% on the Sequoia UD Treebank (Candito and Seddah, 2012; Nivre et al., 2017). Against a smaller and manually validated subset, we reach 94.75% of LAS and 99.42 for unlabeled attachment score. These scores are likely to reflect the high quality of our resulting data set.

In the remaining of this paper, we describe the methodology we used to build the UD version of the FTB, hereafter FTB-UD, and present our evaluation process and results. The FTB-UD is available under the same licence conditions as the original FTB.[2]

2. Method Overview

The basic idea is the following: we had access to a rule-based system for automatically converting another treebank, namely the French Sequoia Treebank (Candito and Seddah, 2012, hereafter SEQUOIA), into UD. After adapting the FTB's native tokenization scheme to UD, this conversion system was directly applied to the FTB. This resulted in many errors: 16% of the sentences contained one or more errors at one or more levels (POS, dependency, head), between 6 and 7% of tokens were flagged as `Fail-`

[1]Statistical Parsing of Morphologically-Rich Languages.

[2]`https://github.com/UniversalDependencies/UD_French-FTB`

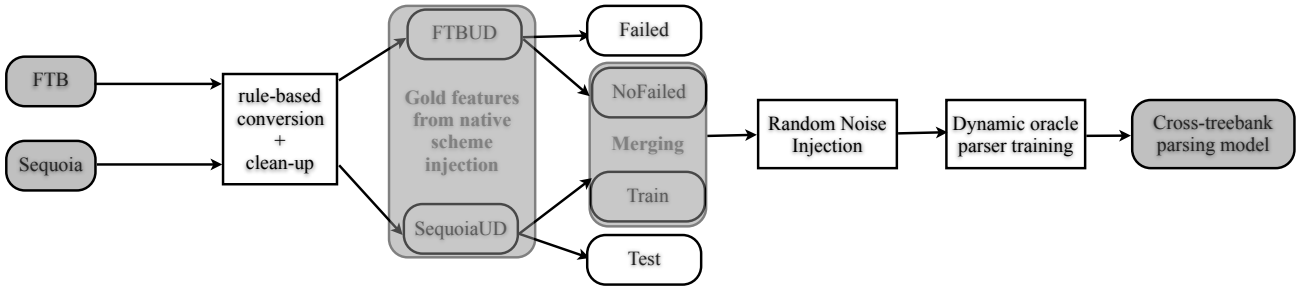


Figure 1: Overview of our cross-treebank parser training process

ure conversion, leading of course to many more incorrect tree structures. The FTB being six times larger than SEQUOIA, adapting and extending the initial set of rules was not feasible in such a short time. We automatically corrected incorrect coordination tree structures and manually corrected missing POS resulting from conversion failures. We then decided to reparse all error-flagged dependencies using our robust shift-reduce parser with dynamic oracle (Villemonte De La Clergerie, 2013).

The idea was to build a pseudo gold training set (made of 90% of the SEQUOIA treebank and of the FTB training sentences that contained no conversion errors, leaving aside 20% of those for pseudo-gold evaluation) to which we injected both (i) external gold morpho-syntactic features coming from the FTB SPMRL version and (ii) random noise, such as empty dependencies, in the same proportions as the initial conversion errors (see Figure 1 for an overview of the training process). We then parsed all erroneous sentences (all incorrect edges were deleted) with this model with the hypothesis that the parser would be able to predict correct dependencies assuming the proper external gold features were to be provided.

3. Building the FTB-UD

Besides providing another source of annotated French data to the CoNLL 2017 shared task participants, our primary goal was to enable cross-parsing comparisons between different annotation schemes, namely the native FTB dependency scheme (Candito et al., 2010) as instantiated in the SPMRL shared tasks (Seddah et al., 2013; Seddah et al., 2014) and the then upcoming UD 2.0 scheme (Nivre et al., 2017) that was to be used for this shared task (Zeman et al., 2017). For these reasons, our starting point is the FTB SPMRL instance and not its latest incarnation.[3]

3.1. Multi-word Expression Treatment

We started by adapting the annotation scheme for multi-word expressions (MWEs). The treebank with less types of MWEs annotated is the Sequoia treebank, containing fixed functional MWEs. We thus used the existing rule-based software of Candito and Crabbé (2009) to "undo" non functional MWEs, namely to recover a regular syntactic structure for regular nominal, adjectival, verbal and adverbial MWEs. The patterns for spotting and undoing MWEs are a subset of those of Candito and Crabbé (2009). All the remaining MWEs were then represented using the `fixed` dependency label, used for functional MWEs. This choice can be discussed in the light of the current debate within the UD community regarding the status to give to named entities. For example, the FTB contains many nominal named entities (tagged `N N`, e.g for persons), assuming a proper disambiguation step, those could have received a `flat:name` label instead. [4]

However, we then adapted the word segmentation to that of French UD 2.0, the main difference concerning amalgamated prepositions: e.g. the amalgamated preposition+determiner *au (litt. "to the")* is systematically treated as one token but two words (*à (to)* and *le (the)*).

3.2. Application of Sequoia to UD rule-based converter

Before we started this work, another research team was working on the conversion of the SEQUOIA treebank to the UD annotation scheme (Guillaume et al., to appear) using their graph rewriting engine (Guillaume et al., 2012). Because the SEQUOIA treebank native annotation scheme uses the same guidelines as the FTB, the use of the rule-set they developed was favored in order to bootstrap the conversion process. However, both corpora differ considerably in size (resp. 3k vs 18k sentences) and domains (wikipedia, europarl, biomedical for SEQUOIA, international and national news-wire for the FTB), leading the application of the SEQUOIA to UD conversion process to a new domain to be non-trivial. As we mentioned in the previous section, the resulting treebank contained 16% of sentences with one or more errors and 6% after correction of some coordinate structures. The next two sections describe how we corrected those errors.

3.3. POS Correction and Injection of Gold Features

POS-correction The application of the conversion rules resulted in a failure to produce a POS tag for 89 word-forms (61 in the training set, 3 in the development set, 25 in the test set). We manually reviewed and POS-annotated all these cases.

Injection of Morpho-syntactic Gold Features We first developed an algorithm for automatically post-align

[3] `http://ftb.linguist.univ-paris-diderot.fr`, released in December 2016.

[4] Please note that the version distributed for the UD Shared Task did not contain this regularization, which will be included in the next major release.

the output of the conversion with the original FTB SPMRL files, which differ in how they are segmented into tokens and wordforms. This algorithm reads both versions of the same sentences, stores wordforms from each file and multi-wordform tokens form the converted version. It then aligns tokens using a robust synchronization algorithm that traverses both token sequences for a given sentence in a left-to-right manner. Whenever tokens do not match, the algorithm performs a lookahead on both token sequences until it is able to find a new anchor point, the "forward anchor". The search for a forward anchor is itself robust to tokenization mismatches, making use of the notion of "weak match" only used for comparing right contexts. The notion of weak match is defined as a disjunction of patterns; the main pattern looks for two consecutive matches or "pseudo-matches" between tokens in the original token sequence and tokens in the converted token sequence.[5] Once a forward anchor is found, tokens between the current position and the forward anchor are aligned according to a finite number of patterns, some of which are aware of the discrepancy in how some prepositions and determiners are agglutinated in the original tokenization scheme (e.g. *des* < *de les*).

Next, for each converted token which is aligned with an original token, its gold syntactic information is extracted from the original SPMRL annotations `(gold_SPMRL_head, gold_SPMRL_fpos, gold_SPMRL_delta, gold_SPMRL_label)` and associated with the converted token in the form of additional features, appended for convenience to the relevant field. These features respectively provide information about the head, fine-grained POS, distance from the governor and label of the current word's governor.

3.4. Parsing-based Treebank Correction

Inspired what had been tried when stacking a symbolic parser with DYALOG-SR (Villemonte de la Clergerie, 2014b), guiding gold features `pseudogold_UD_label` and `pseudogold_UD_delta` were added based on the result of the preliminary automatic conversion. They respectively refer in this preliminary UD version to the label and the (ordered) distance to the governor (if any). Obviously, with such features, which are not gold because of conversion errors but nevertheless quite accurate, training looks like a rather trivial task! However, based on a random process, about 6% of these guiding features were deleted, in order for the parser to learn how to correct a certain amount of errors, based on information about nearby dependencies, words, POS, and obviously SPMRL-based gold features added as per the previous section. It should be also noted that because all these feature are only indicative, the parser may even learn not to follow them under some conditions, in other words, decide that some gold annotations are actually maybe not so correct.

Clearly, that kind of scheme (introducing a small amount of error) can not only be used to correct errors when converting to a new annotation schema (as tried here) but also to track and correct errors in gold annotations.

Initially developed for participating to the SPMRL shared task, the parser we used, DYALOG-SR, is a shift-reduce dependency parser, using Dynamic Programming and beams to explore its search space and a feature-rich perceptron to weight the parser actions (Villemonte De La Clergerie, 2013). Early and aggressive updates of the perceptron are performed at training time. In particular, following ideas from dynamic oracles (Goldberg and Nivre, 2012), updates may occur for actions that clearly results in violations of the gold tree, for instance when adding a bad dependency.

Using such a setting, our model was able to provide a high level of performance on the SEQUOIA gold data (10% not used in the training data and parsed with the same configuration as the data we aimed to correct) with 98.50% of LAS. The same range of accuracy was achieved on the dev and test section of the FTB that contained no conversion errors (resp. 98.48 and 98.64% of LAS).

4. Evaluation

Treebank conversion is a laborious task full of minutiae, and many conversion efforts improve their conversion in an iterative fashion, or as new relevant conversion needs are identified. A full manual evaluation of a converted treebank could represent an effort comparable to full re-annotation of a large part of the data. Indeed, few of the UD-conversion papers provide accuracy scores of the conversion on a manually annotated testbench.

For instance, The Danish conversion of Johannsen et al. (2015), uses a small set of hand-annotated sentences that reflect specific phenomena and hard cases that is used as held-out section during the iterative development of conversion rules. The Hungarian conversion of Vincze et al. (2017) uses a hand-corrected gold standard of 1,800 sentences. When comparing the quality of the conversion with the gold standard, they consider the accuracy (87.81 UAS and 75.99 LAS) not sufficient to release the resulting treebank.

We draw inspiration on their method to develop a hand-corrected sample to evaluate the quality of our conversion.One of the authors of the article, an expert in dependency annotation very familiar with the UD formalism, reviewed 100 sentences from the test section and 100 sentences from the dev section manually, correcting edges and labels that were either not properly attached, or not compliant with UD2.

Table 1 shows the scores for the manual validation. The Unlabeled Attachment Score (UAS) is very high, as the annotator did not disagree with most of the edges resulting from the conversion. However, the results are more drastic when analyzing the quality of the labels.

If we examine the label corrections by the expert annotation, we find that most of them reside on the label *fixed*, which has been used conservatively for all associated multiword expressions. Out of 360 relabelings overall, 274 are

[5] For instance a token *face* 'in front' in the converted token sequence will be considered as a pseudo-match with a token *face_à* 'in front of' in the original token sequence. This pseudo-match will result in an offset of 1 on the converted side, in order to skip the probable token *à* that follows the converted token *face*. A weak match will therefore be found if the converted token following this *à* is a match or pseudo-match with the token following *face_à* in the original token sequence.

Section	UAS	LAS
Dev	99.44	93.27
Test	99.40	94.75

Table 1: Manual evaluation scores for 100-sentences excerpts from the dev and test section.

relabelings for edges converted into *fixed* that should otherwise be *compound* or *flat:name*.

While some of the corrections for the *fixed* relation can be automated depending on the syntactic role of overall multiword subtree—e.g. a subtree that works as *case* is a multiword adposition and should be labeled *fixed*, while a *nsubj* label would per a proper name or a compound—the distinction between these tree types of relations, that are not exactly dependency relations in nature but must be described as such by virtue of the UD formalism, requires per-item linguistic analysis.

We have not observed any cases of mis-conversion of the core nominal arguments of verbs, which means that subjects and objects are always properly annotated, as well as the root note. In general, missattachments happen at lower points of the dependency tree that are closer to the leaves and are thus less relevant for overall dependency quality (Plank et al., 2015).

After multiword expressions, there are roughly thirty cases where the expert determined that the preferred relation should have been either *appos* (apposition) or *parataxis*. These are already controversial labels and are not easy to annotate. However, this indicates that the quality of the treebank is high enough for the most frequent expert relabelings to be within the domain of the fine distinctions of syntactic-semantic relations. Indeed, there was only one sentence out of the pooled 200 where there were present errors caused by coordination embedding, where the tree had to be corrected for the inner coordinates not to attach outside of the scope of their closest subsuming coordination.

5. Conclusion

We have described our effort to provide a highly reliable conversion of FTB into UD2.0 based on a convert-then-reparse principle. This method provides very high unlabeled accuracy (99.42 on average between 200 sentences). However, the quality of the resulting treebanks will need to be kept up to date with the advancements in the UD formalism, including a more homogeneous treatment of parataxis and appositions, as well as a detailed per-item analysis of multiword expressions and their potential relabeling. This method will be applied to the French Question Bank (Seddah and Candito, 2016) and to other data sets for English.

6. Acknowledgment

We warmly thank Bruno Guillaume and Guy Perrier for kindly running their set of rules on our data set, as well as Teresa Lynn for guiding the first author throughout the UD intricacies. This work was partly funded by the French ANR projects ParSiTi (ANR-16-CE33-0021 and SoSweet (ANR-15-CE38-0011-01), as well as by the Program "Investissements d'avenir" ANR-10-LABX-0083 (Labex EFL).

7. Bibliographical References

Anne Abeillé, Lionel Clément, and François Toussenel, 2003. *Building a Treebank for French*. Kluwer, Dordrecht.

Bharat Ram Ambati, Tejaswini Deoskar, and Mark Steedman. 2013. Using ccg categories to improve hindi dependency parsing. In *ACL (2)*, pages 604–609.

Anders Björkelund, Ozlem Cetinoglu, Richárd Farkas, Thomas Mueller, and Wolfgang Seeker. 2013. (re) ranking meets morphosyntax: State-of-the-art results from the spmrl 2013 shared task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145.

Marie Candito and Benoît Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technologies - IWPT'09*, pages 169–172, Paris, France, October.

Marie Candito and Djamé Seddah. 2012. Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Proc. of TALN*.

Marie Candito, Benoit Crabbé, and Pascal Denis. 2010. Statistical french dependency parsing: Treebank conversion and first results. In *Proc. of LREC*.

Jan Einarsson. 1976. Talbankens talspråkskonkordans.

Richárd Farkas and Bernd Bohnet. 2012. Stacking of dependency and phrase structure parsers. In *COLING*, pages 849–866.

Yoav Goldberg and Joakim Nivre. 2012. A dynamic oracle for arc-eager dependency parsing. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 959–976.

Bruno Guillaume, Guillaume Bonfante, Paul Masson, Mathieu Morey, and Guy Perrier. 2012. Grew : un outil de réécriture de graphes pour le TAL. In *Proc. of TALN*, Grenoble, France.

Bruno Guillaume, Guy Perrier, and Guillaume Bonfante. (to appear). *Application de la Réecriture de Graphes au Traitement Automatique des Langues*.

Anders Johannsen, Héctor Martínez Alonso, and Barbara Plank. 2015. Universal Dependencies for Danish. In *International Workshop on Treebanks and Linguistic Theories (TLT14)*, page 157.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Eckhard Bick, Cristina Bosco, Gosse Bouma, Sam Bowman, Marie Candito, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Fabricio Chalub, Jinho Choi, Çağrı Çöltekin, Miriam Connor, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza

Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Nizar Habash, Jan Hajič, Linh Hà Mỹ, Dag Haug, Barbora Hladká, Petter Hohle, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Natalia Kotsyba, Simon Krek, Veronika Laippala, Phương Lê Hồng, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Vitaly Nikolaev, Hanna Nurmi, Stina Ojala, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Loganathan Ramasamy, Livy Real, Laura Rituma, Rudolf Rosa, Shadi Saleh, Manuela Sanguinetti, Baiba Saulīte, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jonathan North Washington, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, and Hanzhi Zhu. 2017. Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Lilja Øvrelid, Jonas Kuhn, and Kathrin Spreyer. 2009. Improving data-driven dependency parsing using large-scale lfg grammars. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 37–40. Association for Computational Linguistics.

Barbara Plank, Héctor Martínez Alonso, Željko Agić, Danijela Merkler, and Anders Søgaard. 2015. Do dependency parsing metrics correlate with human judgments? In *Eighteenth Conference on Computational Natural Language Learning (CoNLL 2015)*.

Corentin Ribeyre, Éric Villemonte De La Clergerie, and Djamé Seddah. 2015. Because Syntax does Matter: Improving Predicate-Argument Structures Parsing Using Syntactic Features. In *Proc. of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Denver, USA, June.

Djamé Seddah and Marie Candito. 2016. Hard time parsing questions: Building a questionbank for french. In *Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Djamé Seddah, Reut Tsarfaty, Sandra Kübler, Marie Candito, Jinho D. Choi, Richárd Farkas, Jennifer Foster, Iakes Goenaga, Koldo Gojenola Galletebeitia, Yoav Goldberg, Spence Green, Nizar Habash, Marco Kuhlmann, Wolfgang Maier, Joakim Nivre, Adam Przepiórkowski, Ryan Roth, Wolfgang Seeker, Yannick Versley, Veronika Vincze, Marcin Woliński, Alina Wróblewska, and Eric Villemonte de la Clergerie. 2013. Overview of the SPMRL 2013 shared task: A cross-framework evaluation of parsing morphologically rich languages. In *Proc. of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 146–182, Seattle, Washington, USA, October. Association for Computational Linguistics.

Djamé Seddah, Sandra Kübler, and Reut Tsarfaty. 2014. Introducing the spmrl 2014 shared task on parsing morphologically-rich languages. In *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, pages 103–109.

Éric Villemonte De La Clergerie. 2013. Exploring beam-based shift-reduce dependency parsing with DyALog: Results from the SPMRL 2013 shared task. In *4th Workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL'2013)*.

Éric Villemonte De La Clergerie. 2014a. Jouer avec des analyseurs syntaxiques. In *TALN 2014*, Marseilles, France, July. ATALA.

Éric Villemonte de la Clergerie. 2014b. Jouer avec des analyseurs syntaxiques. In *Proc. of TALN 2014 (Volume 1: Long Papers)*, pages 67–78, Marseille, France, July. Association pour le Traitement Automatique des Langues.

Veronika Vincze, Katalin Simkó, Zsolt Szántó, and Richárd Farkas. 2017. Universal dependencies and morphology for hungarian-and on the price of universality. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 356–365.

Daniel Zeman, Filip Ginter, Jan Hajič, Joakim Nivre, Martin Popel, Milan Straka, and et al. 2017. CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–20, Vancouver, Canada.

Universal Dependencies and Quantitative Typological Trends. A Case Study on Word Order

Chiara Alzetta[+], Felice Dell'Orletta[*], Simonetta Montemagni[*], Giulia Venturi[*]
[+] Università degli Studi di Genova, Via Balbi 5, Genova, Italy
[*]Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR) – ItaliaNLP Lab, *www.italianlp.it*
via G. Moruzzi, 1- Pisa, Italy
chiara.alzetta@edu.unige.it, {felice.dellorletta,simonetta.montemagni,giulia.venturi}@ilc.cnr.it

Abstract

The paper presents a new methodology aimed at acquiring typological evidence from "gold" treebanks for different languages. In particular, it investigates whether and to what extent algorithms developed for assessing the plausibility of automatically produced syntactic annotations could contribute to shed light on key issues of the linguistic typological literature. It reports the first and promising results of a case study focusing on word order patterns carried out on three different languages (English, Italian and Spanish).

Keywords: Linguistic Knowledge Extraction, Dependency Treebanks, Linguistic Typology

1. Introduction

The interaction between linguistics and computational linguistics has a long history dating back to the 60's. In Kučera (1982), it is explicitly stated that "computational linguistics provides important potential tools for the testing of theoretical linguistic constructs and of their power to predict actual language use". This still appears to represent a key objective, as claimed e.g. by Martin Kay in his ACL Lifetime Award speech in 2005 (Kay, 2005), or by the more recent papers gathered in the Special Issue of the journal "Linguistic Issues in Language Technology" (LiLT) focusing on the relationship between language technology and linguistic insights (Baldwin and Kordoni, 2011). After more than 40 years from the first pioneering studies, the growing availability of linguistic resources such as annotated corpora for many languages combined with the increasing reliability of Natural Language Processing (NLP) methods and tools enables the acquisition of quantitative evidence ranging across different levels of linguistic description which can significantly contribute to the study of open issues of the theoretical linguistic literature.

This holds particularly true for the area of typological studies which can benefit a lot from this synergy, making it possible to acquire quantitative evidence shedding light on how, why and to what extent languages vary with respect to key features covering major areas of language structure. By exploring collections of linguistically annotated corpora for different languages, complex and articulated frequency distributions of language constructions can be extracted. Information acquired from available corpora can significantly enrich typological descriptions of languages such as the *World Atlas of Language Structures* (WALS) (M. S. Dryer and M. Haspelmath, 2013) [1], the most commonly-used and broadest database of structural (phonological, grammatical, lexical) properties of languages whose data are based on primary sources such as grammars, dictionaries and scientific papers. But impact and role of this information type cannot be limited to the descriptive level. Typological evidence inferred from linguistically annotated corpora for different languages can significantly contribute to model linguistic variation within and across languages. Word order variation represents a widely investigated topic of the typological literature whose recent developments include fine-grained studies based on a wide range of features and their frequency distributions typically acquired from annotated corpora (O'Horan et al., 2016). To mention only a few, see e.g. Gulordava and Merlo (2015a), Gulordava and Merlo (2015b) Futrell et al. (2015), Merlo (2016).

More recently, such an approach to typological studies has also been prompted by the availability of multi–lingual treebanks such as those designed and constructed within the Universal Dependencies project[2] for over 50 languages. Universal Dependencies (UD) is a framework for cross–linguistically consistent treebank annotation aiming to capture similarities as well as idiosyncrasies among typologically different languages (e.g., morphologically rich languages, pro–drop languages, and languages featuring clitic doubling). The goal in developing UD was not only to support comparative evaluation and cross–lingual parsing but also to enable comparative linguistic studies (Nivre, 2015). Within this area of research, the paper reports the results of preliminary experiments aimed at acquiring quantitative typological evidence from a selection of the UD treebanks for different languages. In particular, it focuses on the widely investigated topic of word order, with the specific aim of reconstructing word order patterns within and across languages, for what concerns both the linear order of words and its degree of flexibility (giving rise to a wide typology of languages going from fixed-order to free word order languages). For the acquired word order patterns, the study also aims at investigating the factors underlying the preference for one or the other order, both intra- and cross-linguistically.

To pursue this goal, we decided to test whether existing algorithms for assessing the plausibility of automatically produced syntactic annotations could be used to acquire useful quantitative typological evidence. In fact, the result of these algorithms is typically driven by linguistic properties

[1] Available online at http://wals.info

[2] http://universaldependencies.org/

characterizing the language being processed: by comparing the results achieved against different languages, it is possible to acquire information concerning typological similarities and differences. This kind of algorithms operate at the level of either the whole syntactic tree (cfr. for example Dell'Orletta et al. (2011) and Reichart and Rappoport (2009)), or individual dependencies (see, among others, Dell'Orletta et al. (2013) and Che et al. (2014)). Given the focus of this study on specific constructions, we selected the class of algorithms operating at the level of individual dependencies, and in particular on those ranking dependencies by decreasing plausibility of annotation. These algorithms, originally meant to discern reliable from unreliable annotations within the automatic output of parsers, have also been applied to manually revised (i.e. "gold") linguistic annotations with the final aim of identifying annotation inconsistencies, and thus for also detecting annotation errors (Alzetta et al., 2018). Tusa et al. (2016) represent the first attempt to exploit the plausibility score returned by this class of algorithms to acquire linguistic evidence, i.e. to infer the prototypicality degree of specific linguistic constructions. The experiment was carried out against the *Italian Universal Dependency Treebank* (IUDT) (Bosco et al., 2013) with promising results: the plausibility-based ranking of dependencies corresponding to specific syntactic constructions turned out to closely reflect their linguistic "markedness" degree.

In what follows, we focus on word pattern variation across three different languages, English, Italian and Spanish. This goal is pursued by applying a plausibility assessment algorithm against the UD treebanks available for these languages. Achieved results have been compared with the threefold aim of: i) reconstructing the frequency distributions of different word order patterns, with particular attention to specific constructions (Subject-Verb and Adjective-Noun); ii) assessing similarities and differences across languages; and iii) identifying and weighting the factors underlying the different word order patterns identified.

2. Background and Motivation

Starting from Greenberg (1963), word order has been used to set up a typology of languages based on the notion that "certain languages tend consistently to put modifying or limiting elements before those modified or limited, while others just as consistently do the opposite". Within this area of research, the relative ordering of constituents at the clausal level (e.g. verb and subject) as well as at the phrasal level (e.g. noun and modifying adjective) has been widely investigated in the typological literature. Nowadays, the outcome of these studies has been collected in publicly-accessible typological databases. Table 1 reports - for the three languages considered in our study - the different orderings of lexical (i.e. non pronominal) Subject and Verb, and of Adjective and Noun as resulting from the *World Atlas of Language Structures* (or WALS) and the *Syntactic Structures of the World's Languages* (or SSWL) (SSWL, 2009) databases. It can be noted that the two provide a slightly different picture for the three languages, maybe following from the fact that whereas the former records the "dominant" order the latter testifies "productive" word order patterns, which may not always coincide. According to WALS, no dominant Subject-Verb order exists for both Italian and Spanish (M. S. Dryer, 2013b), whereas Subject-Verb is considered a productive word order for both languages in SSWL. If Adjective-Noun is a productive order for all the three languages according to SSWL, in WALS the Noun-Adjective order is considered dominant in Italian and Spanish while the reverse order is reported for English (M. S. Dryer, 2013a).

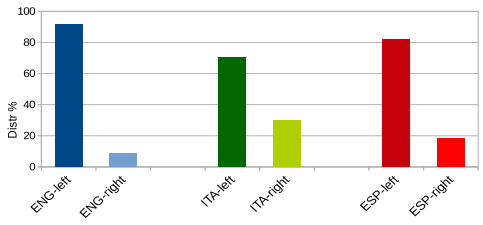


Figure 1: Distribution of right– vs left–headed non–pronominal *nsubj* relations in the three UD treebanks.

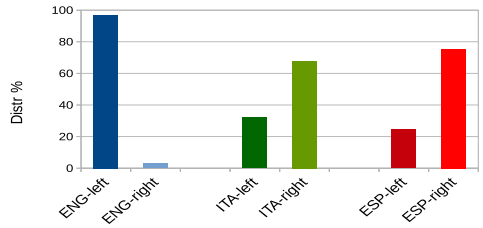


Figure 2: Distribution of right– vs left–headed *amod* relations in the three UD treebanks.

Consider now the picture emerging from actual usage as attested in the English, Italian and Spanish UD treebanks. Figures 1 and 2 report, for the three treebanks, the percentage distribution of different word orders involving non pronominal Subjects and Verbs (corresponding to the *nsubj* dependency relation), and Adjectives and Nouns (*amod*), respectively. Note that *Left* and *Right* in the figures refer to the position of the dependent (subject or adjectival modifier) with respect to its syntactic head. As it can be noted, all languages turned out to prefer the Subject-Verb order, but with significant differences: namely, the Italian and Spanish treebanks are characterized by a much higher percentage of left–headed subjects than English (i.e. 30% in Italian, 18% in Spanish and 8% in English). For what concerns adjectival modifiers, Figure 2 shows that whereas for English the Adjective-Noun order is highly preferred (though not the only possible one) in Italian and Spanish the reverse order is rather preferred, i.e. with the adjective occurring on the right side of the head.

Corpus-based evidence helps quantifying attested word order patterns across languages, thus leading to a more articulated picture of word order variation. Registered order variants, however, do not appear to be evenly distributed, both intra- and cross-linguistically. Some are used more often and show less grammatical or stylistic restrictions than others. Let us consider, for example, the relative ordering of

WALS				
	Word Order	**English**	**Italian**	**Spanish**
Subject-Verb	Subject-Verb	+	No dominant order	No dominant order
	Verb-Subject	-	No dominant order	No dominant order
Adjective-Noun	Adjective-Noun	+	-	-
	Noun-Adjective	-	+	+
SSWL				
Subject-Verb	Subject-Verb	+	+	+
	Verb-Subject	-	+	+
Adjective-Noun	Adjective-Noun	+	+	+
	Noun-Adjective	+	+	+

Table 1: Subject-Verb and Adjective-Noun order in WALS and SSWL.

Subject and Verb. Figure 3 reports different *nsubj* instances occurring in the Italian UD treebank[3]. The subject in a) (lit. 'In this case, *he* answers within limits ...') and c) (lit. '... *the rights not overtly granted by the licensor* remain confidential') occurs in the same position, i.e. pre-verbally. There are however important differences worth noting here: in a) a pronominal subject immediately precedes the verb, while in c) a long–distance dependency relation links the nominal subject to its head. The question which naturally arises here is how a) and c) relate to each other, and what are the underlying properties explaining this difference, if any. On the other hand, the sentence reported in b) (lit. 'Here, from each corner of the world, arrive *300 thousand patients*') exemplifies a different, less common, Verb-Subject order involving a nominal subject. The question at this point is how a) and c), both with pre-verbal subjects, relate to b), with a post-verbal subject. Both questions cannot be answered by simply considering finer-grained frequency distributions of different types of ordering. On the basis of this, we can claim that a) and c) represent more likely verb-subject instances than b). But this may not be the case. To answer questions like these, the properties underlying the different word order patterns in a given language and cross-linguistically need to be investigated.

Thanks to the availability of corpora for different languages with manually revised linguistic annotation, the focus of studies on word order variation across languages moved from discerning possible vs impossible word orders as in the pioneering studies, to defining dominant vs rare word order patterns based on actual frequencies attested in corpora. More recently, thanks to the wide variety of features which can be tracked down and quantified in linguistically annotated corpora, current explanations of word order variation can also aim at capturing finer-grained distinctions able to predict the frequency distribution of attested word orders in different languages. In what follows, we will try to exploit the wide range of features which can be extracted from treebanks not only to characterize word order patterns across languages, but also to identify and weight the factors underlying them.

3. Method and Data

The methodology we devised to acquire typological evidence from gold treebanks is based on the parse plausibility assessment algorithm named LISCA (LInguiStically–driven Selection of Correct Arcs) (Dell'Orletta et al., 2013). As illustrated in details in Section 3.1., the algorithm exploits statistics about a wide range of linguistic features (covering different description levels, going from raw text to morpho-syntax and dependency syntax) extracted from a large reference corpus of automatically parsed sentences and uses them to assign a *plausibility* score to each dependency arc contained in a target corpus belonging to the same variety of use (i.e. textual genre) of the automatically parsed corpus. Accordingly, all the arcs contained in the target corpus are ranked from those characterized by a high LISCA score to arcs with lower scores: the higher the score, the more similar the linguistic context of an arc with respect to the statistics acquired from the large reference corpus. The underlying assumption is that syntactic structures that are more frequently generated by a parser are more likely to be plausible than less frequently generated ones.

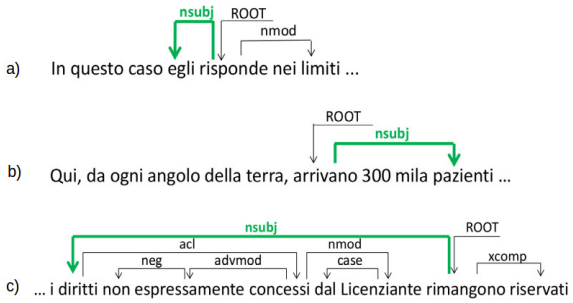


Figure 3: Different instances of the *nsubj* relation in the Italian UD treebank.

3.1. The LISCA Algorithm

LISCA takes as input a set of parsed sentences and it assigns a plausibility score to each dependency, which is defined as a triple (d, h, t) where d is the dependent, h is the head, and t is the type of dependency connecting d to h. The algorithm operates in two steps: *1)* it collects statistics about a set of linguistically motivated features extracted from a dependency annotated corpus obtained through automatic dependency parsing, and *2)* it combines the feature statistics extracted from the corpus used during the previous step. The final plausibility score associated with a given dependency arc results from the combination of the weights associated with these features: the score is computed as a

[3] In the examples, the dependent is italicized and the head underlined.

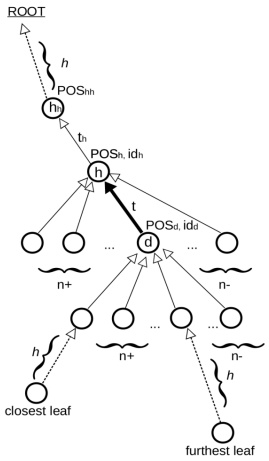


Figure 4: Features used by LISCA to measure arc(d, h, t) plausibility.

simple product of the individual feature weights.

Figure 4 illustrates the features taken into account by LISCA for measuring the plausibility of a given syntactic dependency (d, h, t). For the purposes of the present study, LISCA has been used in its de–lexicalized version in order to abstract away from variation resulting from lexical effects. In particular, two different types of features are considered:

- *local* features, corresponding to the characteristics of the syntactic arc considered, such as the distance in terms of tokens between d and h, or the associative strength linking the grammatical categories (i.e. POS$_d$ and POS$_h$) involved in the relation, or the POS of the head governor and the type of syntactic dependency connecting it to h;
- *global* features, aimed at locating the arc considered within the overall syntactic structure of the sentence: for example, the distance of d from the root of the tree, or from the closest or most distant leaf node, or the number of "brothers" and "children" nodes of d, occurring respectively to its right or left in the linear order of the sentence.

LISCA was successfully used against both the output of dependency parsers and gold treebanks. While in the first case the plausibility score was meant to identify unreliable automatically produced dependency relations, in the second case it was used to detect shades of syntactic markedness of syntactic constructions in manually annotated corpora. The latter is the case of Tusa et al. (2016), where the LISCA ranking was used to investigate the linguistic notion of "markedness" (Haspelmath, 2016): a given linguistic construction is considered "marked" when it deviates from the "linguistic norm", i.e. it is "abnormal". Accordingly, unmarked constructions are expected to be characterized by higher LISCA scores and – conversely – constructions characterized by increasing degrees of markedness are associated with lower scores.

Let us go back to the different instances of the *nsubj* relation in Figure 3. The plausibility score assigned to them by LISCA results in the following ranking: a) is assigned a higher score with respect to b), whose score in turn is higher than that assigned to c). This ranking follows from the combination of both local and global features taking into account the overall tree structure. From the resulting LISCA score, it turned out e.g. that in Italian longer distance subjects are less prototypical than post–verbal and shorter ones.

4. Languages and Corpora

For the specific concerns of this study we focused on two typologically close languages, namely Italian and Spanish, and a more distant one, English: for what concerns morphology, all three languages are fusional, although English has very few inflectional morphemes, which makes it rather similar to isolating languages. These properties imply that the prototypical sequence of the main constituents in English strictly follows the linear order Subject-Verb-Object (SVO); Italian and Spanish are SVO languages too, but show more syntactic freedom in the linear ordering of constituents. Because of these properties, highly related to the linguistic type each language belongs to, we expect to observe a similar behaviour for typologically close languages and, on the other hand, significant differences in case of typologically distant languages.

The corpora used to collect the statistics to build the LISCA models (step 1 in Section 3.1. above) are represented by the English, Italian and Spanish Wikipedia, for a total of around 40 million tokens for each language. The Spanish and English corpora were morpho–syntactically annotated and parsed by the UDPipe pipeline (Straka et al., 2016) trained on the Universal Dependency treebanks, version 2.0 (J. Nivre and A. Željko and A.Lars and et alii, 2017). The Italian corpus was morpho–syntactically tagged using the ILC–POS–Tagger (Dell'Orletta, 2009), and then parsed with UDPipe.

LISCA, trained on the models we created earlier, was then applied to the Italian, English and Spanish UD Treebanks in order to assign a plausibility score to each dependency relation. The English Web Treebank (Silveira et al., 2014) contains 16,624 sentences and 254,830 tokens, while the Italian Universal Dependency Treebank (Bosco et al., 2013) contains 13,815 sentences corresponding to 325,816 tokens. The Spanish UD treebank (McDonald et al., 2013) is the smallest one, with 4,000 sentences and 112,718 tokens. For all resources, most part of the sentences comes from blogs and/or newspapers.

5. Data Analysis

For each treebank, the dependencies were first ordered by decreasing LISCA scores. The list of ordered dependencies was then subdivided into 10 groups, henceforth "bins", each corresponding to 10% of the total. The distribution of syntactic dependencies in the LISCA bins was analyzed in order to investigate whether and to what extent it could be used to acquire typological trends, i.e. similarities and differences across languages. This analysis has been carried out by comparing the dependency rankings by LISCA (each subdivided into 10 bins) for the three languages taken into account. As reported by O'Horan et al. (2016), the

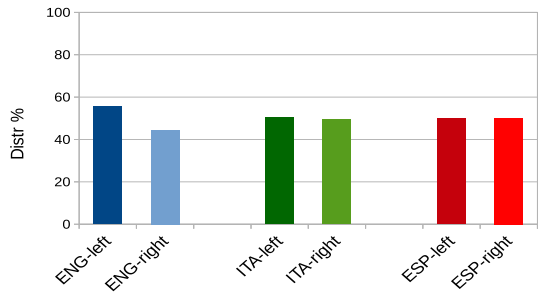


Figure 5: Frequency distribution of right– vs left–headed dependencies in the three UD treebanks.

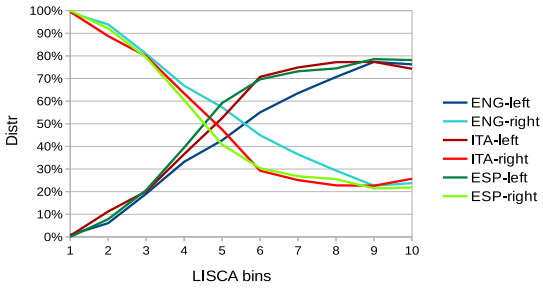


Figure 6: Distribution of right– vs left–headed dependencies across the bins.

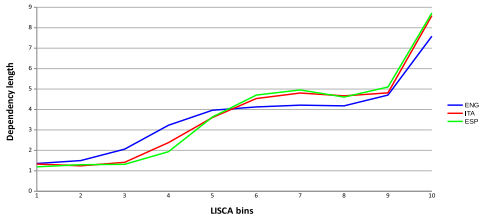


Figure 7: Average length of dependencies across the bins.

automatic learning of typological information is typically carried out from parallel texts. In the case of our study, parallelism is not concerned with texts, but rather with the ranking of instances of dependendency relations by LISCA: besides the fact that the LISCA score is based on the same set of properties for the three languages, comparability is guaranteed here by the same inventory of dependency relations and annotation guidelines shared by the UD treebanks taken into account.

In what follows, we will focus on word order patterns: Section 5.1. focuses on a cross-lingual analysis of general trends of word order formalized in terms of the direction of the dependency link, whereas Section 5.2. reports the results of an in–depth analysis of the frequency distribution of two dependency relations, i.e. nominal subject (*nsubj*) and adjectival modifier (*amod*), corresponding to widely investigated constructions in the typological literature.

5.1. Word Order Patterns: General Trends

As a first step, for each treebank we analyzed how the dependencies are distributed with respect to their direction. For the specific purposes of this study, the order is defined by the right or left direction of the dependency that connects d to h with respect to the linear order of words in the sentence. Figure 5 shows the frequency distribution of all dependencies in the three considered UD treebanks, distinguishing right–headed dependencies (i.e. $d > h$) vs left–headed ones (i.e. $h < d$).[4] Interestingly enough, in the three treebanks the frequency of the the two word orders is similar: whereas for Italian and Spanish they are equally partitioned (50% both $d > h$ and $h < d$), for English the ordering $d > h$ covers 55% of the cases. Yet, if we focus on the distribution of dependencies across the LISCA bins interesting differences across languages can be observed (see Figure 6). Despite their similar frequency in the three languages, right– vs left–headed dependencies are described by opposite trends: whereas in the first bins $d > h$ dependencies are more frequent, in the latter $h < d$ dependencies are mainly observed.

This result confirms the intuition we started from, i.e. that statistics about the frequency distribution of right- vs left-headed dependencies in treebanks do not say much about the underlying structural properties of languages. On the contrary, the direction-based distribution of dependencies resulting from the LISCA ranking shows interesting similarities and differences across languages. All languages share the same trend, i.e. the top LISCA (namely 1-3) bins mainly contain right-headed dependencies, whose occurrence progressively decreases across the bins, until the last two bins (9-10) where they represent around the 20-25% of the dependencies. Similar observations hold in the case of left-headed dependencies, which are characterized by the reverse trend: their occurrence starts in the second bin and progressively increases to cover about 80% of the relations in the last two bins (9-10). Although this trend is shared by the three languages taken into account, there are also significant differences worth being pointed out. In fact, the trend reported in Figure 6 for English is slightly different if compared with what observed for the other two Romance languages. For English, the lines representing the distribution of the left– and right–headed dependencies are less steep and cross at the level of the 6th bin, while the Italian and Spanish lines are steeper and cross in the 5th one. This is to say that the distribution of dependencies in the interval between the 4th and 9th bins is language-specific. A possible explanation for this is that English word order is more rigid and shows a higher number of right–headed dependencies (including explicit subjects). On the contrary, Italian and Spanish, characterized by a more flexible word order, show a higher variability at the level of dependency direction.

We can now try to understand what distinguishes right– or left–headed dependencies occurring in the top vs bottom bins. Consider the distribution of dependencies with respect to their length, as shown in Figure 7: all languages share the fact that longer dependencies are ranked

[4] In all figures *left* and *right* refer to the position of d relative to its governor h, respectively corresponding to right–headed (i.e. $d > h$) vs left–headed (i.e. $h < d$) dependencies.

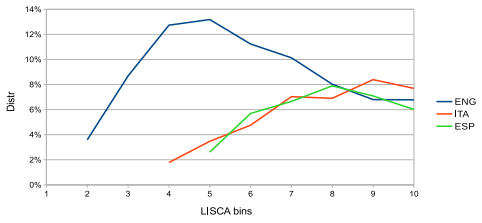


Figure 8: Distribution of the *nsubj* relation across the bins.

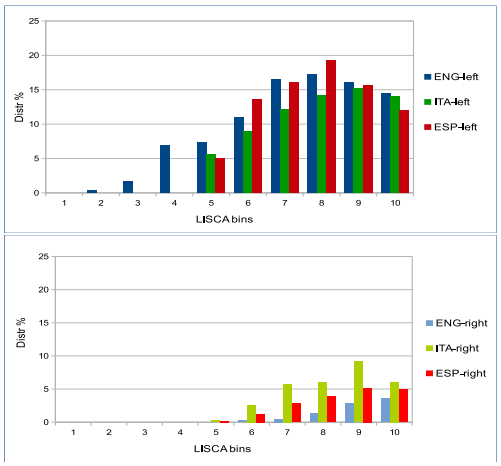


Figure 9: Direction of the *nsubj* relation across the bins.

by LISCA in the last bins. This in line with the Dependency Length Minimization principle (Temperly, 2007), i.e. the idea that languages tend to place closely related words close together since shortest dependency links reduce the human processing load and make the sentence comprehension process easier (Gibson, 1998). If we consider the average length of dependencies for each language, we note that they are quite similar, i.e. they are about three–token long on average. However, the LISCA ranking highlighted interesting differences between typologically different languages. The average length of Italian and Spanish dependencies share a more similar trend with respect to the English ones: the confidence interval of the difference between means is lower for Italian and Spanish (0.077 points) than both English/Italian (0.47 points), and English/Spanish (0.54 points). On the basis of this we can hypothesize an interesting interplay between dependency direction and dependency length: among the underlying properties of relations occurring in the last bins there are structural factors at work such as dependency length.

5.2. The case of *nsubj* and *amod* relations

Let us focus now on specific dependency relations, namely nominal subjects (*nsubj*) and adjectival modifiers (*amod*), and their distribution across the LISCA bins. These relations correspond to constructions widely investigated in the typological literature, and for this reason they represent two challenging testbeds for our methodology of analysis. Given the typology of features underlying LISCA, different factors contribute to the distribution of the same relation across the bins, concerning local features such as the linear ordering of words involved in the relation or their distance, to more global characteristics reflecting the position of the dependency arc within the overall dependency tree. In what follows, we try to reconstruct what are properties playing a role in determining the distribution of different word order patterns across the LISCA bins.

Nominal subjects (*nsubj*). Figure 8 reports the distribution of nominal subjects (both lexical and pronominal) across the LISCA bins. As it can be noted, the three languages are characterized by different distributions. First, whereas for English *nsubj* relations already appear in the top positions of the LISCA ranking, i.e. in the 2nd bin, the first occurrences of *nsubj* relations for Italian and Spanish are in the 4th and 5th bins respectively. Second, main differences can be observed at the level of frequency distributions. Both differences can be explained by considering that whereas Italian and Spanish are pro–drop languages English obligatorily requires an explicit subject. This is also reflected by the frequency distribution of pronominal subjects, which correspond to 59% of the total amount of nominal subjects in the English treebank, while they cover only about 3% of the cases in both Italian and Spanish treebanks.

Consider now the distribution of *nsubj* relations by dependency direction: in line with what reported in Section 2., we focus now on lexical (i.e. non-pronominal) subjects only. In Figure 1, it is shown that in all considered languages subjects are mostly right-headed, whereas significant differences are recorded for left–headed subjects whose percentage is much higher for Italian and Spanish (i.e. 30% and 18% respectively) than for English (9%). Figure 9 reports, for the three languages, the distribution of right–headed subjects (i.e. $d > h$) vs left–headed ones (i.e. $h < d$) across the LISCA bins. Left–headed subjects turned out to be mainly concentrated in the second half of the LISCA bins, starting from the 5th one. Consider as an example the following *nsubj* relation ranked in the 10th bin for Italian: 'Dalla rabbia dei valonesi non si salva *niente e nessuno*' (lit. 'From the rage of Valaisers not be saved *nothing and nobody*', 'From the rage of Valaisers nothing and nobody can be saved'). The frequency of left–headed subjects for the English language is much lower than for Romance languages, being they mainly restricted to parenthetical clauses, such as for example *[...], said Bush*, and existential clauses.

We have seen that the LISCA bins progressively gather occurrences of rarer and less prototypical relation instances, such as left-headed subjects, whose occurrence mainly concentrates in the last four bins. Yet, as already pointed out in Section 2., the position with respect to the governing head may be influenced by different linguistic properties. Consider the following examples for the three languages:

- Italian: '*La proposta presentata dalla commissione conformente al suo mandato* costituisce un punto di partenza' (lit. '*The proposal presented by the commission in accordance with its mandate* represents a starting point'), where the subject *proposta* ('proposal') is modified by a participial phrase which creates a long-distance *nsubj* dependency;

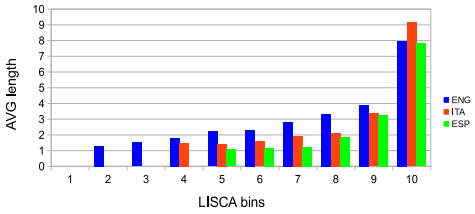


Figure 10: Average length of *nsubj* relations across the bins.

- Spanish: '*El descenso de la población indígena y la falta de mano de obra para los obrajes españoles* originó el comercio de pobladores secuestrados ...' (lit. '*The decline of the indigenous population and the lack of labor for the Spanish obrajes* led to the trade of kidnapped settlers ...'), where a coordinated subject determines a long-distance dependency link;

- English: 'Sergey Brin has actually a mathematical proof that the company's self–driven research *strategy, which gives employees one day a week to do research projects on their own, is a good, respectable* idea', where 22 tokens occur between the lexical subject *strategy* and its syntactic head.

All *nsubj* relations exemplified above have been ranked in the last (i.e. 10th) LISCA bin of each ordered list of dependencies: they all represent long distance dependencies involving a right-headed subject. These examples suggest an interesting and complex interplay between dependency length and the position of the subject with respect to the governing head in influencing word order patterns, which is worth being investigated.

In Section 5.1. we hypothesized a correlation between word order and dependency length. Let us now explore how the two interact for a specific dependency relation, *nsubj*. In Figure 10, it is reported that for all languages "heavier" (i.e. long distance) subjects (both right- and left-headed) are ranked in the last bins. For most part of the bins, Romance languages show stronger similarities with respect to English. Despite these differences, each language appears to follow the same trend, characterized by the fact that the ordered bins contain increasingly longer dependencies. Two questions arise at this juncture: i) whether dependency length is also influenced by the syntactic realization of the subject, i.e. lexical or pronominal; ii) for lexical subjects, whether and to what extent dependency length influences subject order patterns.

Concerning the first question, our intuition is that longer dependencies are typically represented by lexical subjects, i.e. by *nsubj* relations with a noun as dependent. This is confirmed by the fact that for all languages *nsubj* relations ranked in the first LISCA bins mostly have a pronoun as a dependent. On the other hand, relations ranked in the last bins are represented by long-distance dependencies with lexical subjects, as exemplified by the sample sentences reported above for the three languages.

Consider now the second open issue above, in particular the distribution of left– and right–headed subjects across the

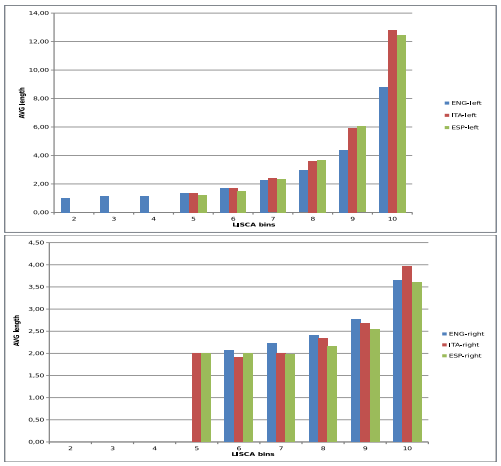


Figure 11: Average length of right– and left–headed lexical *nsubj* across the LISCA bins.

LISCA bins with respect to their length, to assess the correlation, if any, between dependency length and word order. As shown in Figure 11, for all languages the maximum average length of right–headed *nsubj* relations is higher (i.e. about 12 tokens for the Romance languages and about 8 for English) than the maximum average length of less frequent left–headed subjects (which is about 4 tokens for all languages). Besides reported differences among languages, a similar trend can be reported here: namely, all languages tend to minimize the dependency length when an alternative (with respect to the dominant, i.e. most frequent one) word order is used. Dependency length minimization varies across languages: whereas in English the average dependency length of right–headed subjects is only slighly lower with respect to left–headed ones (i.e. 2.62 vs 2.76), for Italian and Spanish length minimization is more evident (i.e. 2.48 vs 4.64 and 2.38 vs 4.54 respectively). From what seen so far, we can put forward the hypothesis that length minimization plays a stronger role with less frequent, and therefore more marked, word orders: in other words, if on the one hand marked order is associated with minimized dependency length, on the other hand the dominant unmarked order allows significantly longer dependencies.

Adjectival modifiers *amod*. Figure 12 reports the distribution of adjectival modifiers across the bins: it can be noticed that while for English they already appear in the 1st bin, for both Romance languages the first occurrences of *amod* relations start in the 2nd bin. This difference can be explained by considering that even in this area English is characterized by a generally fixed order or at most slightly variable structures, which are more easily predictable. Their distribution across the bins also varies significantly, with Italian and Spanish sharing a similar trend as opposed to English.

Consider now the relative order of Adjective and Noun: in Figure 2 it was shown that the pre–nominal adjectives are more common in the English UD treebank than in the treebanks of the Romance languages. The distribution of the *amod* relation across the LISCA bins shows that right–headed adjectives are more prototypical in English than in Italian and Spanish (see Figure 2). This is due to the fact

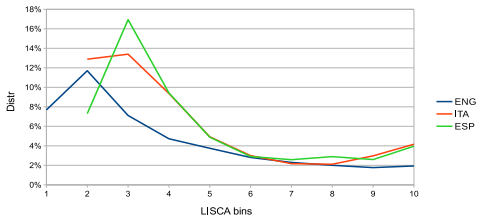


Figure 12: Distribution of the *amod* relation across the bins.

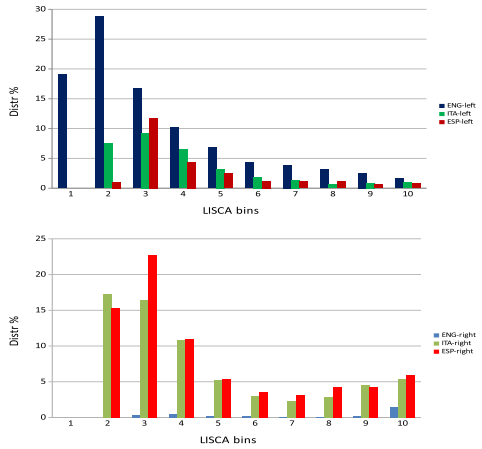


Figure 13: Direction of the *amod* relation.

that, differently from English, a marked position of adjectives is allowed in Italian and Spanish. The distribution of relations across the LISCA bins allows detecting adjectival modifiers occurring in prototypical constructions.

Differently from what observed for subjects, the average distance between d and h in *amod* relations remains tendentially constant through the bins (see Figure 14), ranging between 1 and about 3. Some differences can be observed if we compare English and the two Romance languages: the latter tend to be characterized by shorter relations. This may be explained in terms of adjacency to the nominal head: i.e. in the left–headed position adjectival modifiers are typically adjacent to the head, which is not necessarily the case in the case of prenominal position where the adjectival modifier can be separated from the head by elements that belong to the same subtree.

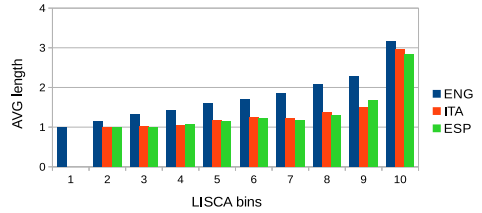


Figure 14: Average length of *amod* relations.

6. Conclusion

In this paper we presented a new methodology aimed at acquiring typological evidence from "gold" treebanks for different languages. In particular, we investigated whether algorithms for measuring the plausibility of dependency relations within the output of dependency parsers could be exploited to acquire quantitative evidence from gold treebanks to shed new light in linguistic typological studies. The methodology was tested in a case study carried out on UD treebanks for different languages: English, Italian and Spanish. By relying on a wide range of linguistic properties aimed at weighting the plausibility of a given dependency arc, it has been possible to reconstruct an articulated profile of word order patterns attested in the languages considered, in line with the literature and which has been enriched with new interesting insights. Starting from the study of general word order trends and their relationship with dependency length, we focused on two dependency relations widely investigated in the typological literature, nominal subjects (*nsubj*) and adjectival modifiers (*amod*). For both of them, word order similarities and differences were reported for all languages, significantly enriching the picture emerging from typological databases and showing the added value of the LISCA-based methodology with respect to simple frequency distributions. We also investigated the underlying properties influencing the preference towards one word order or the other. Among them, dependency length turned out to play a significant role: its impact, however, appears to vary according to whether a dominant or marked order is used. Dependency length minimization seems to be at work with less frequent order patterns, thus suggesting an interesting interaction between word order and dependency length.

However, the potentialities of the method are not restricted to the area of typological studies. Nowadays, linguistic typology is starting to play a role in multilingual Natural Language Processing (O'Horan et al., 2016). While the growing importance of typological information in supporting multilingual tasks has been recognized, existing typological databases such as WALS have still a partial coverage, and most importantly here, do not always reflect real language use. Methods for automatic induction of typological information are still at the beginning: this paper represents a promising attempt in this area. It is a widely acknowledged fact that word-order affects the automatic analysis of sentences: free–order languages are harder to parse (Gulordava and Merlo, 2015c). Acquired information from real language usage can be used among the "selective sharing parameters" in a cross–lingual transfer parsing scenario (Naseem et al., 2012).

Further directions of research include: the application of the methodology to other languages, including typologically distant ones, to reconstruct shades of typological proximity starting from real language usage; the analysis of other dependency relations as well as of more complex structures such as dependency subtrees; the extension of the range of properties expected to influence the preference towards a given word order pattern. Last but not least, we are planning to test the effectiveness of acquired typological information in a cross–lingual parsing scenario.

Acknowledgements

The work reported in the paper was partially supported by the 2–year project (2017-2019) *UBIMOL, UBIquitous Massive Open Learning*, funded by Regione Toscana (BANDO POR FESR 2014-2020).

7. Bibliographical References

Alzetta, C., Dell'Orletta, F., Montemagni, S., and Venturi, G. (2018). Dangerous relations in dependency treebanks. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories (TLT16)*, pages 201–210, Prague, Czech Republic, January.

Baldwin, T. and Kordoni, V., (2011). *Interaction between Linguistics and Computational Linguistics*. CSLI Publications.

Bosco, C., Montemagni, S., and Simi, M. (2013). Converting italian treebanks: Towards an italian stanford dependency treebank. In *Proceedings of the ACL Linguistic Annotation Workshop & Interoperability with Discourse*, Sofia, Bulgaria, August.

Che, W., Guo, J., and Liu, T. (2014). Reliable dependency arc recognition. *Expert Systems with Applications*, 41(4):1716–1722.

Dell'Orletta, F., Venturi, G., and Montemagni, S. (2011). ULISSE: an unsupervised algorithm for detecting reliable dependency parses. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning (CoNLL 2011)*, pages 115–124, Portland, Oregon, USA, June. Association for Computational Linguistics Stroudsburg, PA, USA.

Dell'Orletta, F., Venturi, G., and Montemagni, S. (2013). Linguistically-driven selection of correct arcs for dependency parsing. *Computaciòn y Sistemas*, 2:125–136.

Dell'Orletta, F. (2009). Ensemble system for part-of-speech tagging. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December.

Futrell, R., Mahowald, K., and Gibson, E. (2015). Quantifying word order freedom in dependency corpora. In *Proceedings of the Third International Conference on Dependency Linguistics, DepLing 2015, August 24-26 2015, Uppsala University, Uppsala, Sweden*, pages 91–100.

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68:1–76.

Greenberg, J. (1963). Some universals of grammar with particular reference to the order of meaningful elements. In Joseph H. Greenberg, editor, *Universals of Language*, pages 73–113.

Gulordava, K. and Merlo, P. (2015a). Diachronic trends in word order freedom and dependency length in dependency-annotated corpora of latin and ancient greek. In *Proceedings of the Third International Conference on Dependency Linguistics, DepLing 2015, August 24-26 2015, Uppsala University, Uppsala, Sweden*, pages 121–130.

Gulordava, K. and Merlo, P. (2015b). Structural and lexical factors in adjective placement in complex noun phrases across romance languages. In *Proceedings of the 19th Conference on Computational Natural Language Learning, CoNLL 2015, Beijing, China, July 30-31, 2015*, pages 247–257.

Gulordava, K. and Merlo, P. (2015c). Structural and lexical factors in adjective placement in complex noun phrases across romance languages. In *Proceedings of Conference on Computational Natural Language Learning (CoNLL)*, pages 247–257, Beijing, China, July.

Haspelmath, M. (2016). Against Markedness (and what to Replace it with). *Journal of Linguistics*, 42:25–70.

Kay, M. (2005). Acl lifetime achievement award: A life of language. *Computational Linguistics*, 31(4).

Kučera, H. (1982). Markedness and frequency: a computational analysis. In *Proceedings of COLING 82*.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Castelló, N. B., and Lee, J. (2013). Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 92–97.

Merlo, P. (2016). Quantitative computational syntax: some initial result. *Italian Journal of Computational Linguistics*, 2.

Naseem, T., Barzilay, R., and Globerson, A. (2012). Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 629–637, Jeju, Republic of Korea, July.

Nivre, J. (2015). Towards a universal grammar for natural language processing. In *Computational Linguistics and Intelligent Text Processing - Proceedings of the 16th International Conference, CICLing 2015, Part I*, pages 3–16, Cairo, Egypt, April.

O'Horan, H., Berzak, Y., Vulic, I., Reichart, R., and Korhonen, A. (2016). Survey on the use of typological information in natural language processing. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 1297–1308.

Reichart, R. and Rappoport, A. (2009). Sample selection for statistical parsers: Cognitively driven algorithms and evaluation measures. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL 2009)*, pages 3–11. Association for Computational Linguistics Stroudsburg, PA, USA.

Silveira, N., Dozat, T., de Marneffe, M., Bowman, S., Connor, M., Bauer, J., and Manning, C. (2014). A gold standard dependency corpus for english. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*.

Straka, M., Hajic, J., and Strakova, J. (2016). Udpipe: Trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*.

Temperly, D. (2007). Minimization of dependency length

in written english. *Cognition*, 105(2):300–333.

Tusa, E., Dell'Orletta, F., Montemagni, S., and Venturi, G. (2016). Dieci sfumature di marcatezza sintattica: verso una nozione computazionale di complessitá. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it)*, pages 3–16, Napoli, Italy, December.

8. Language Resource References

J. Nivre and A. Željko and A.Lars and et alii. (2017). *Universal Dependencies 2.0*. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, 2.0, ISLRN http://hdl.handle.net/11234/1-1983.

M. S. Dryer and M. Haspelmath. (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, ISLRN http://wals.info/.

M. S. Dryer. (2013a). *Order of Adjective and Noun*. Max Planck Institute for Evolutionary Anthropology, ISLRN http://wals.info/chapter/87.

M. S. Dryer. (2013b). *Order of Subject and Verb*. Max Planck Institute for Evolutionary Anthropology, ISLRN http://wals.info/chapter/82.

SSWL. (2009). *Syntactic Structures of the World's Languages*. ISLRN http://sswl.railsplayground.net/.

Undersampling Improves Hypernymy Prototypicality Learning

Koki Washio, Tsuneaki Kato
Department of Language and Information Sciences
The University of Tokyo
3-8-1, Komaba, Meguroku, Tokyo 153-8902 Japan
{kokiwashio@g.ecc, kato@boz.c}.u-tokyo.ac.jp

Abstract

This paper focuses on supervised hypernymy detection using distributional representations for unknown word pairs. Levy et al. (2015) demonstrated that supervised hypernymy detection suffers from overfitting hypernyms in training data. We show that the problem of overfitting on this task is caused by a characteristic of datasets, which stems from the inherent structure of the language resources used, hierarchical thesauri. The simple data preprocessing method proposed in this paper alleviates this problem. To be more precise, we demonstrate through experiments that the problem that hypernymy classifiers overfit hypernyms in training data comes from a skewed word frequency distribution brought by the quasi-tree structure of a thesaurus, which is a major resource of lexical semantic relation data, and propose a simple undersampling method based on word frequencies that can effectively alleviate overfitting and improve distributional prototypicality learning for unknown word pairs.

Keywords: lexical semantic relations, hypernymy detection, taxonomy induction

1. Introduction

Detecting hypernymy relations between unknown words contributes to Taxonomy Induction, which induces a taxonomy in a new domain (Panchenko et al., 2016).

Supervised distributional hypernymy detection represents each word pair (x, y) as combined distributional representations, and trains a classifier that discriminates based on whether the word pair has a relation. Frequently used methods for combining word representations include vector concatenation $\vec{x} \oplus \vec{y}$ (CONCAT) and difference $\vec{y} - \vec{x}$ (DIFF) (Baroni et al., 2012; Fu et al., 2014; Roller et al., 2014; Weeds et al., 2014; Vylomova et al., 2016). The supervised methods have been reported to be better than the distributional unsupervised measures (Roller et al., 2014; Weeds et al., 2014).

However, Levy et al. (2015) demonstrated that supervised classifiers have some problems. Two major problems are as follows:

Problem 1 Classifiers do not learn relations in word pairs but only learn distributional prototypicality at best.

Problem 2 Classifiers overfit hypernyms, especially those in training data (*lexical memorization*).

Distributional prototypicality, if learned correctly, is still useful. Shwartz et al. (2016) integrated this information into their neural path-based model, which captures relations between two words, and improved the performance significantly. Roller and Erk (2016) demonstrated that the prototypicality learned by CONCAT captures Hearst patterns such as "fruits such as apples." Their method using the CONCAT model as a feature detector has high generalization performance. Thus, resolving Problem 2 is expected to improve the performance of the previous models.

In this paper, we investigate why classifiers overfit hypernyms in training data. We analyze this problem from the point of view of the skewed distribution of frequencies of hypernyms in training data, which stems from the inherent structure of language resources such as hierarchical thesauri. We show that an imbalance of word frequencies adversely affects classifiers, and we verify our analysis by experiments.

Moreover, we show that a simple undersampling method to balance frequencies of words in training data effectively alleviates the overfitting of hypernyms. Our experiment demonstrates that the undersampling method successfully improves the generalization performance for unknown word pairs.

2. Problems of Supervised Methods

Problem 1 is demonstrated by the tendency that supervised classifiers incorrectly assign hypernymy labels to switched pairs, which are mismatched instance-category pairs, *e.g.*, (apple, vehicle). Levy et al. (2015) provided a mathematical analysis on why linear classifiers cannot learn word relations. The forms of DIFF and CONCAT can be described as follows:

$$\begin{aligned} DIFF(x, y; \vec{\theta}) &= \vec{\theta} \cdot (\vec{y} - \vec{x}) \\ &= \vec{\theta} \cdot \vec{y} - \vec{\theta} \cdot \vec{x} \end{aligned} \quad (1)$$

$$\begin{aligned} CONCAT(x, y; \vec{\theta_1}, \vec{\theta_2}) &= (\vec{\theta_1} \oplus \vec{\theta_2}) \cdot (\vec{x} \oplus \vec{y}) \\ &= \vec{\theta_1} \cdot \vec{x} + \vec{\theta_2} \cdot \vec{y} \end{aligned} \quad (2)$$

where $\vec{\theta}$ and $(\vec{\theta_1}, \vec{\theta_2})$ are parameter vectors of DIFF and CONCAT, respectively. While the parameter vectors of DIFF and CONCAT can be interpreted as distributional prototypicality, these methods do not consider interactions between x and y. Thus, they cannot capture the relation between the word pair. Levy et al. (2015) also tried to use nonlinear kernel SVM, which can capture interactions between word vectors. However, the improvements were marginal.

Problem 2 is demonstrated by the fact that when the training data and test data have no lexical overlap (lexical split setting), classifiers perform extremely poorly. Levy et al. (2015) also showed that even if classifiers learn with only

Dataset		Mean	Median	Mean-Median	Max	Min
HyperLEX	hyper	2.37	1	1.37	62	1
	hypo	1.18	1	0.18	4	1
EVALution	hyper	2.73	1	1.73	71	1
	hypo	1.54	1	0.54	7	1
LEDS	hyper	3.41	1	2.41	60	1
	hypo	1.21	1	0.21	4	1

Table 1: Statistics of word frequencies in each position of hypernymy pairs of each dataset.

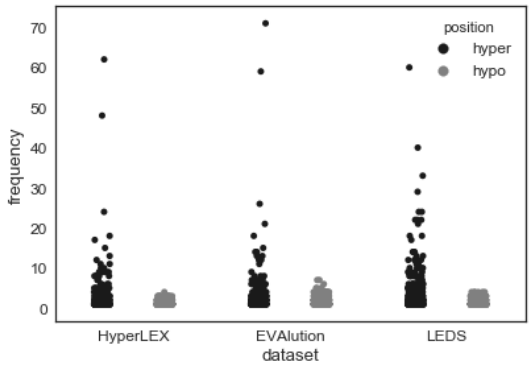


Figure 1: Strip plot of frequencies of hypernyms and hyponyms on each dataset.

$\vec{y}$ of word pair (x, y), their performance does not decrease so much in the lexical split setting. This indicates that classifiers ignore x's information. Problem 2 makes classifiers incapable of appropriately classifying words not included in the training data. This is a critical issue for a downstream task such as Taxonomy Induction.

While Problem 1 was provided with sufficient analysis by Levy et al. (2015), why do the classifiers overfit hypernyms in training data and ignore the information of hyponyms? This is the problem we address in this paper.

3. A Reason for Overfitting Hypernyms

We build a hypothesis that focuses on the distribution of hypernym frequencies[1] of training data to investigate what causes the overfitting of hypernyms.

Thesauri, major resources of word relation datasets, typically have a quasi-tree structure. One word/concept can have many hyponyms, but only one or a few hypernyms. If word pairs are extracted from thesauri, in training data, the same words that have general meanings appear naturally many times at the hypernym position of word tuples. As a result, the distribution of frequencies of a particular word being a hypernym of other words in training data becomes skewed in that the distribution is long-tailed or has many outliers. Figure 1 displays strip plots of the frequencies of hypernyms and hyponyms on three datasets: HyperLEX (Vulić, Ivan and Gerz, Daniela and Kiela, Douwe and Korhonen, Anna, 2016), EVAlution (Santus, Enrico and Yung, Frances and Lenci, Alessandro and Huang, Chu-Ren, 2015), and LEDS (Baroni, Marco and Bernardi, Raffaella and Do, Ngoc-Quynh and Shan, Chung-chieh, 2012). Table 1 displays the statistics of the word frequencies in each position of the hypernymy pairs of each dataset, where the difference of the mean and median of the hypernym position are larger than those of the hyponym position. These shows that the hypernym frequencies are largely skewed, while the hyponym frequencies are slightly skewed on all datasets. In these datasets, general and common hypernyms, such as *food*, *animal*, and *vehicle*, have significantly high frequencies.

How does this property affect DIFF and CONCAT? In training data, the number of types of hypernyms is small, and some types appear many times, while the number of types of hyponyms is large, and each type appears only a few times. Thus, words with a hypernym position have tendencies such as domain similarity in addition to the expected features of the prototypical hypernymy, while words with a hyponym position have few tendencies. This makes DIFF and CONCAT concentrate on repeated hypernym vectors $\vec{y}$ and ignore $-\vec{\theta} \cdot \vec{x}$ in equation (1) and $\vec{\theta}_1 \cdot \vec{x}$ in equation (2), as hyponyms share fewer features than do duplicating hypernyms. The biased supervised training shifts the parameter vectors to the features of hypernyms rather than to the true prototypicality, and results in overfitting hypernyms in the training data and ignoring hyponym information.

3.1. Experiments

To confirm our hypothesis, we conduct two experiments. First, we investigate how the skewed distribution of words affects the performance of the classifiers by adding extra pairs to the training data. Second, we examine the correlation between the hypernym frequencies and the mean inner products of the trained parameter vector (distributional prototypicality) and the feature vectors.

3.1.1. Setup

For distributional representation, we exploit the pretrained dependency word embeddings of Levy and Goldberg (2014). For datasets, we use HyperLEX, EVAlution, and LEDS. Only noun pairs of each dataset are used in our experiments. We remove samples containing words out of the vocabulary of the representations. We split each dataset into train/test subsets while keeping a roughly 75/25 ratio in random/lexical splitting.[2]

[1] In this paper, we use *hypernym/hyponym frequency* as the frequency of a particular word being a hypernym/hyponym of other words in word pair data.

[2] For HyperLEX, we use the standard train/test/dev splits that were provided in the dataset. In our experiment, the development

Split	HyperLEX	EVAlution	LEDS
random	607 (+1616)	880 (+2456)	328 (+2074)
lexical	586 (+1051)	547 (+799)	303 (+719)

Table 2: Numbers of added samples with frequent hypernyms. Values in parentheses show numbers of original training samples.

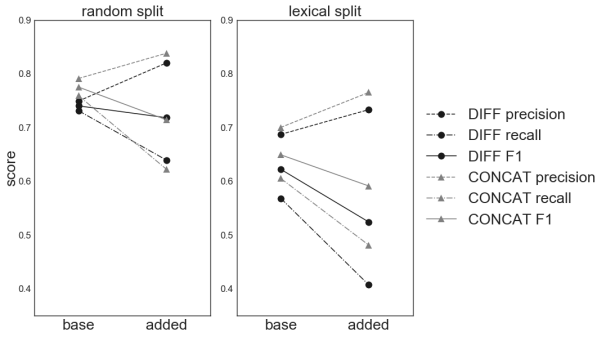


Figure 2: Plot of performance scores of DIFF and CONCAT classifiers in random and lexical splitting on HyperLEX.

We use logistic regression with L2 regularization for classifiers, exploiting scikit-learn[3] (Pedregosa et al., 2011) with the default hyperparameters, with the exception of the use of balanced class weights.

3.1.2. Skewed Distribution Influence

To investigate how skewed distributions of words affect the performance, we conduct the following experiment:
We extract the 10 most frequent hypernyms from the training data. Then, we extract direct hypernymy pairs with these frequent hypernyms from WordNet (Fellbaum, Christiane, 1998) and add the pairs that are not included in either the training or the test data to the training data.[4] This process makes the distribution of hypernym frequencies in training data more skewed. The numbers of the added samples are listed in Table 2. We evaluate the performance of classifiers in random and lexical splitting when adding new pairs for each splitting.
Figure 2 shows the results for HyperLEX. We obtained similar results for the other datasets. We can see that the more skewed the hypernym frequencies in the training data, the higher the precision and the lower the recall, dropping the F1 score as a result.[5] This tendency can be interpreted in that the classifiers focus only on frequent hypernyms in the training data and fail to correctly classify hypernymy pairs with infrequent hypernyms. This experiment demonstrates that skewed distributions of words give rise to overfitting.

3.1.3. Correlation Experiment

We train DIFF and CONCAT classifiers on each entire dataset without splitting, and examine the correlation between the hypernym frequencies and the mean of the inner products of the trained parameter vector and combined word representations on each hypernym frequency.
If the classifier goes through the ideal supervised learning, the obtained prototypical hypernymy, namely the parameter vector, should be irrelevant to frequencies of hypernyms in the training data. However, the correlations are significantly high at all settings ($\rho > 0.7$)[6]. This indicates that hypernym frequencies in the training data have a relationship to overfitting.
These two experiments demonstrate that a skewed distribution of hypernyms is a major factor in overfitting hypernyms. In addition, it leads to the ignoring of information about hyponyms whose distribution is not skewed.

4. Undersampling Method

Based on the analyses of Section 3, we propose an undersampling method to alleviate the overfitting of hypernyms and improve distributional prototypicality learning.
This method first calculates the third quartile of the hypernym frequencies in training data and removes from the training data hypernymy pairs including hypernyms that are more frequent than the third quartile. For each hypernym of those pairs, randomly chosen portions are brought back until the frequency of the hypernym matches the third quartile[7].
This is a simple method to correct the skew of the distribution of the frequencies of hypernyms in training data. We call this method *lexical undersampling* (LU), which is expected to alleviate overfitting hypernyms and improve the classifiers' performance on unknown word pairs.

4.1. Experiments and Results

We use the same datasets, word representations, and logistic regression model described in Section 3.1.1. The baselines are CONCAT and DIFF models with no data augmentation method.
Table 3 displays the results for each dataset. In almost all settings, LU lowers the precision but improves the recall, with the exception of CONCAT on the lexical split setting of EVAlution. This is the opposite trend as what was seen in Section 3.1.2. Thus, it seems that the overfitting of hypernyms is alleviated.
In almost all of the random split settings where classifiers benefit from lexical memorization, the baseline model outperforms +LU on F1 with the exception of DIFF on HyperLEX. This might be because LU makes it difficult for the models to take advantage of lexical memorization because of undersampling. These results indicate that LU is not beneficial to the random split because LU disturbs lexical memorization.
In all of the lexical split settings where the performance for unknown word pairs is evaluated, +LU outperforms the baselines on F1 for both CONCAT and DIFF. These results demonstrate that LU improves the learned distributional prototypicality and the generalization performance

set and the training set are merged, producing the new training set.

[3] http://scikit-learn.org/stable/

[4] In lexical split settings, we add only the pairs that do not contain the vocabulary of the test data.

[5] Only in the random splitting on LEDS does adding samples slightly lower the precision of DIFF ($0.780 \rightarrow 0.777$).

[6] The actual values in each dataset are listed in Table 4 along with the results of later proposals.

[7] If the third quartile has a decimal point, it is rounded off.

Dataset	Method	DIFF			CONCAT		
		precision	recall	F1	precision	recall	F1
Random split							
HyperLEX	baseline	**0.749**	0.731	0.740	**0.791**	0.759	**0.775**
	+LU	0.728	**0.795**	**0.760**	0.753	**0.795**	0.773
EVAlution	baseline	**0.507**	0.611	**0.554**	**0.528**	0.655	**0.585**
	+LU	0.441	**0.624**	0.517	0.477	**0.668**	0.556
LEDS	baseline	**0.780**	0.836	**0.807**	**0.766**	0.824	**0.794**
	+LU	0.765	**0.841**	0.802	0.756	**0.827**	0.790
Lexical split							
HyperLEX	baseline	**0.687**	0.568	0.622	**0.700**	0.605	0.649
	+LU	0.654	**0.630**	**0.642**	0.667	**0.741**	**0.702**
EVAlution	baseline	**0.424**	0.574	0.488	0.466	0.603	0.526
	+LU	0.410	**0.632**	**0.497**	**0.479**	**0.662**	**0.556**
LEDS	baseline	**0.782**	0.601	0.680	**0.821**	0.608	0.699
	+LU	0.763	**0.629**	**0.690**	0.769	**0.699**	**0.733**

Table 3: Performance for each model and splitting.

Method	baseline		+LU	
	DIFF	CONCAT	DIFF	CONCAT
HyperLEX	0.719	0.767	0.573	0.655
EVAlution	0.833	0.720	0.110	0.111
LEDS	0.744	0.710	0.347	-0.551

Table 4: Correlation between frequency of hypernyms in data and mean inner products.

	base	+LU
HyperLEX	0.488	**0.668**
EVAlution	0.471	**0.750**
LEDS	0.877	1.143

Table 5: Ratio of mean of squared parameters of hyponym to that of hypernym on CONCAT models.

for unknown words. These results indicate that LU is effective for unknown word pairs. Handling unknown words well is important to applications such as taxonomy induction. It is also possible to change the model depending on whether a pair in question includes a known word.

4.1.1. Diminished Correlation

In addition, we apply LU to the correlation experiments of Section 3.1.3. We use LU when learning the distributional prototypicality, and calculate the correlation with the original dataset. Table 4 shows that LU successfully diminishes the correlations and reduces the bias to frequent hypernyms. The negative correlation for CONCAT on LEDS might be because two-thirds of the negative samples of this dataset are switched pairs derived from the positive samples, which makes the frequent hypernyms negative signals when LU is applied.

4.1.2. Well-Balanced Weighted Features

In order to explore how the models weight words at the hypernym and hyponym positions, we investigate the ratio of the mean of the squared parameters of the hyponym position to that of the hypernym position of the CONCAT classifiers on each dataset. If the ratio is close to 1, the model equally weights the word at each position. On the other hand, if the ratio is close to 0, the model weights only the hypernym position word, which indicates that the model ignores the hyponym position word. Table 5 displays the ratio for each dataset. We can see that LU makes the classifiers focus more on the words of hyponym positions in all datasets. Applying LU successfully obtains a ratio close to 1 for HyperLEX and EVAlution, although it reverses the ratio for LEDS for the same reason as the negative correlation in Table 3. This means that the classifiers trained with LU on these datasets assign weights more equally to the hypernym vector and the hyponym vector. These results indicate that LU alleviates overfitting hypernyms and ignoring hyponyms.

	Roller and Erk (2016)	+LU
HyperLEX	0.667	**0.712**
EVAlution	0.538	**0.573**
LEDS	0.772	**0.801**

Table 6: F1 score of Roller and Erk (2016) in lexical split setting.

4.1.3. Contributing to Sophisticated Methods

Finally, LU contributes to the generalization performance of sophisticated methods using a distributional prototypicality such as that seen in Roller and Erk (2016) by providing valid components.

The model of Roller and Erk (2016) works through an iterative procedure similar to Principal Component Analysis, in which CONCAT is trained as a feature detector capturing a distributional prototypicality, and then this information is removed from the CONCAT vectors, resulting in a vector rejection. Training is repeated using the obtained vector rejection. With the CONCAT's parameters as a feature detector of distributional prototypicality, each process produces meta-features including the similarity of two words, hypernymy prototypicality, and distributional inclusion. After n

times of feature extracting, the final classifier is trained with these meta-features.

We apply LU to the feature detection step of their model and examine the F1 in the lexical split setting[8]. Table 6 shows that LU significantly improves the performance in the lexical split setting. This result demonstrates that the methods exploiting distributional prototypicality benefit from our undersampling method.

5. Conclusion

We investigated why classifiers overfit hypernyms in supervised distributional hypernymy detection. We showed that the skewed distribution of hypernym frequencies of training data makes classifiers overfit hypernyms and ignore hyponym information. This problem exemplifies the complex relationship between a task and its datasets.

Moreover, we proposed a simple undersampling method, lexical undersampling, to balance the hypernym frequencies in the training data. We demonstrated that this method successfully alleviates the overfit, and improves the distributional prototypicality learned by the classifiers and their generalization performance for unknown word pairs.

6. Acknowledgment

This work was supported by JSPS KAKENHI Grant numbers JP17H01831, JP15K12873.

7. Bibliographical References

Baroni, M., Bernardi, R., Do, N.-Q., and Shan, C.-c. (2012). Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209, Baltimore, Maryland.

Levy, O. and Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 302–308, Baltimore, Maryland.

Levy, O., Remus, S., Biemann, C., and Dagan, I. (2015). Do supervised distributional methods really learn lexical inference relations? In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 970–976, Denver, Colorado.

Panchenko, A., Faralli, S., Ruppert, E., Remus, S., Naets, H., Fairon, C., Ponzetto, S. P., and Biemann, C. (2016). Taxi at semeval-2016 task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1320–1327, San Diego, California, June. Association for Computational Linguistics.

[8]We set the number of iterations of the feature detection at $n = 4$, following Roller and Erk (2016).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Roller, S. and Erk, K. (2016). Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2163–2172, Austin, Texas.

Roller, S., Erk, K., and Boleda, G. (2014). Inclusive yet selective: Supervised distributional hypernymy detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1025–1036, Dublin, Ireland.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany.

Vylomova, E., Rimell, L., Cohn, T., and Baldwin, T. (2016). Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1671–1682, Berlin, Germany.

Weeds, J., Clarke, D., Reffin, J., Weir, D., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, Dublin, Ireland.

8. Language Resource References

Baroni, Marco and Bernardi, Raffaella and Do, Ngoc-Quynh and Shan, Chung-chieh. (2012). *Entailment above the word level in distributional semantics*. Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics.

Fellbaum, Christiane. (1998). *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, ISLRN 379-473-059-273-1.

Santus, Enrico and Yung, Frances and Lenci, Alessandro and Huang, Chu-Ren. (2015). *EVALution 1.0: an Evolving Semantic Dataset for Training and Evaluation of Distributional Semantic Models*. Proceedings of the 4th Workshop on Linked Data in Linguistics, Beijing.

Vulić, Ivan and Gerz, Daniela and Kiela, Douwe and Korhonen, Anna. (2016). *HyperLex: A Large-Scale Evaluation of Graded Lexical Entailment*. Language Technology Lab, University of Cambridge.

Interoperability of Language-related Information: Mapping the BLL Thesaurus to Lexvo and Glottolog

V. Dimitrova, C. Fäth, C. Chiarcos, H. Renner-Westermann, F. Abromeit
Goethe University Frankfurt, Germany
{v.dimitrova@ub, faeth@em, chiarcos@em, h.renner-westermann@ub, abromeit@em}.uni-frankfurt.de

Abstract

Since 2013, the thesaurus of the Bibliography of Linguistic Literature (BLL Thesaurus) has been applied in the context of the *Lin|gu|is|tik* portal, a hub for linguistically relevant information. Several consecutive projects focus on the modeling of the BLL Thesaurus as ontology and its linking to terminological repositories in the Linguistic Linked Open Data (LLOD) cloud. Those mappings facilitate the connection between the *Lin|gu|is|tik* portal and the cloud. In the paper, we describe the current efforts to establish interoperability between the language-related index terms and repositories providing language identifiers for the web of Linked Data.

After an introduction of Lexvo and Glottolog, we outline the scope, the structure, and the peculiarities of the BLL Thesaurus. We discuss the challenges for the design of scientifically plausible language classification and the linking between divergent classifications. We describe the prototype of the linking model and propose pragmatic solutions for structural or conceptual conflicts. Additionally, we depict the benefits from the envisaged interoperability - for the *Lin|gu|is|tik* portal, and the Linked Open Data Community in general.

Keywords: *Lin|gu|is|tik* portal, Bibliography of Linguistic Literature (BLL), thesaurus, language identifiers, Linguistic Linked Open Data (LLOD), Glottolog, Lexvo

1. Introduction

The *Lin|gu|is|tik* portal (`www.linguistik.de`) is a hub for linguistically relevant information developed by the University Library Frankfurt and the Applied Computational Linguistics lab at the Goethe University Frankfurt. As a main service it provides a research tool that comprises selected online sources, databases, open access documents, bibliographies and catalogs for linguistic literature. Recently, the portal has been connected with the Linguistic Linked Open Data (LLOD) cloud[1]. A novel, LOD-based search function has been developed, and numerous language resources have been integrated.

The thesaurus of the Bibliography of Linguistic Literature[2] (BLL Thesaurus) serves as a connecting point. It has been modeled according to LOD principles and linked to a linguistic ontology covering the domains of morphology, syntax and morphosyntax.

In this paper, we describe current efforts to enhance the functionality by establishing interoperability between the BLL Thesaurus and two LLOD terminological repositories that provide language identifiers: Lexvo[3] and Glottolog[4] .

2. Motivation and previous work

The LLOD cloud comprises lexical-conceptual resources (dictionaries, knowledge bases), corpora, terminological repositories (thesauri, ontologies), and metadata collections. Published under an open license, these resources can be of great benefit to the users of the *Lin|gu|is|tik* portal. Thus, we decided to establish a connection between both platforms and make as much language resources as possible visible and searchable via the *Lin|gu|is|tik* portal.

The concept we developed builds on the interoperability of linguistic terminology and the interconnected nature of the resources in the cloud (Chiarcos et al. (2016)). Since the BLL Thesaurus provides the majority of the subject headings used for indexation within the *Lin|gu|is|tik* portal, it plays a key role for the implementation.

The BLL Thesaurus is a hierarchically categorized bilingual thesaurus of domain-specific index terms in German and English. Since 1971, the thesaurus has been used in the context of the Bibliography of Linguistic Literature. The subject terms are, therefore, interlinked with a significant amount of bibliographical references.

As of February 2018, the BLL Thesaurus comprises 7,965 subject terms organized in five top-level branches. The main branch *Domains*[5] covers the subdisciplines of linguistics (e.g., *Psycholinguistics*, *Sociolinguistics*) and lists 3,350 subject terms. The branch *Levels* includes the levels of language description (e.g., *Syntax*, *Phonology*) and consists of 2,037 subject terms. 312 subject terms are subsumed under the branch *General topics*. Additionally, the BLL Thesaurus provides 2,242 subject terms for the encoding of language-related information.

In a previous project (finalized in December 2016), interoperability between the BLL Thesaurus and the Ontologies of Linguistic Annotations (OLiA) (Chiarcos and Sukhareva (2015)) was established. The implementation involved the conversion of the BLL Thesaurus in Simple Knowledge Organization System (SKOS)[6] format as well as the modeling of the subject terms as ontological classes using the Web Ontology Language (OWL)[7]. The building of the ontological model (i.e., the BLL Ontology) focused initially on the thesaurus branch *Levels* of language description (Dim-

[1] `http://linguistic-lod.org/llod-cloud`
[2] `http://www.blldb-online.de`
[3] `http://www.lexvo.org/`
[4] `http://glottolog.org/`

[5] Thesaurus subject terms are represented in *italics*, and ontological classes or properties in `typewriter font`.
[6] `https://www.w3.org/2004/02/skos/`
[7] `https://www.w3.org/TR/owl-ref/`

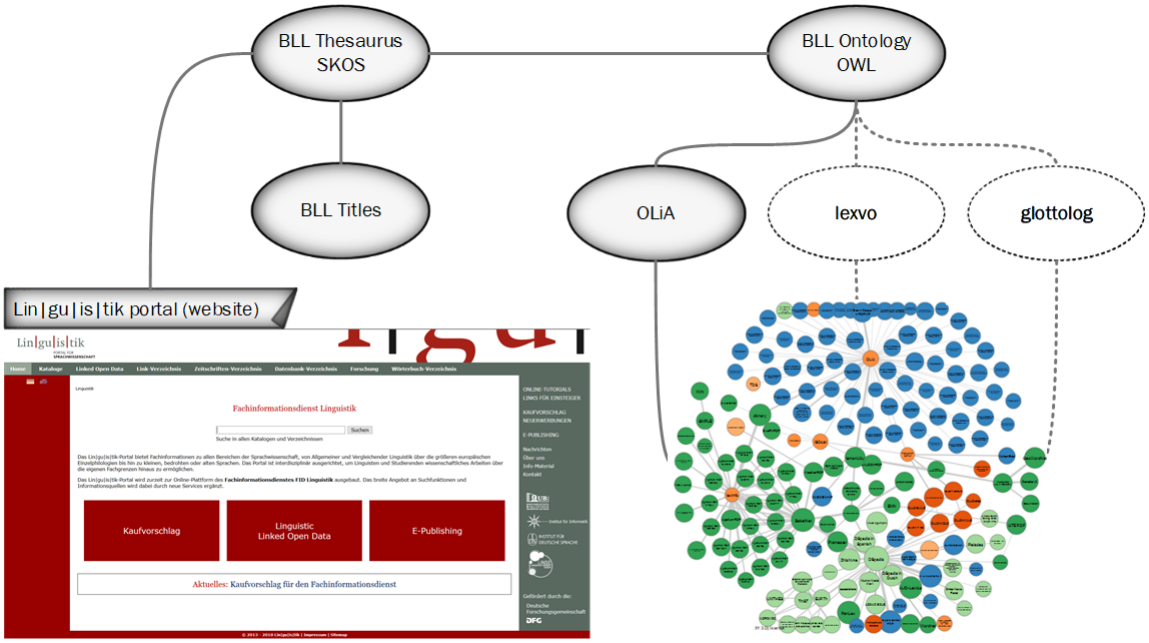


Figure 1: Project architecture

itrova et al. (2016)). By the end of the project, circa 1,100 subject terms covering mainly the domains of syntax and morphology were integrated in the BLL Ontology[8]. Subsequently, the BLL Ontology was linked to the OLiA Reference Model, and on this basis a search algorithm was developed.

Figure 1 features a schematic representation of the connection between the *Lin|gu|is|tik* portal and the LLOD cloud together with the existing (OLiA) and the prospective (Lexvo, Glottolog) links between the platforms. Via the links between the BLL Ontology and OLiA, resources using annotation models interoperable with OLiA have already been integrated into the *Lin|gu|is|tik* portal. Currently, we focus on the branches of the BLL Thesaurus containing language-related information and are working on the integration of the relevant subject terms into the BLL Ontology and their mapping to the LLOD vocabularies Lexvo and Glottolog. Thus, we will achieve broader coverage, enable finer-grained queries, and include more LLOD resources into the portal.

3. Language Identifiers for the Semantic Web

The problems of defining a language make an exact enumeration of the world's languages extremely difficult. Depending on the data source and the classification criteria, the number of the world's languages varies significantly.

The 20th edition of Ethnologue[9], for example, lists 7,099 living languages grouped in 145 language families. As of January 2018, Glottolog defines 7,389 spoken L1 languages classified into 241 families and 188 isolates. The international standard ISO 639-3[10] provides currently 7,858 three-letter codes denoting individual languages or macrolanguages.

3.1. Lexvo

Lexvo is an online service that publishes lexical and language information in both human-readable and machine-readable form. It provides language identification as well as language descriptions. Since 2008, Lexvo has defined URIs of the form `http://www.lexvo.org/id/iso639-3/eng` for the languages covered by the ISO 639-3 standard[11] (de Melo, 2015). The ISO 639-3 standard does not provide codes for dialectal or other substandard forms: each identifier is supposed to denote all spoken or written varieties of the respective language. For language families and language collections, Lexvo defines URIs based on ISO 639-5. The codes provided by this standard are, however, also not exhaustive[12].

For each language, Lexvo delivers extensive descriptions (e.g., multilingual language names, scripts, geographic regions) that are expressed using properties and classes from the Lexvo Ontology[13]. Additionally, the Lexvo Ontology provides properties for notions of identity and near-identity aimed at mitigating with long-standing problems in the Linked Data world (de Melo, 2013). Lexvo not only defines global IDs (URIs) for language-related objects, but

[8]The SKOS version of the BLL Thesaurus and the BLL Ontology can be retrieved from `http://data.linguistik.de/`

[9]`https://www.ethnologue.com/`

[10]`http://www-01.sil.org/iso639-3/`

[11]Unfortunately, the provided identifiers are not completely up-to-date. Since the last update took place in January 2014, changes to the ISO 639-3 codes that took place afterwards are not included in the database.

[12]As of August 2008, ISO 639-5 defined 114 collective codes, covering thus just a portion of the established language families and subfamilies.

[13]`http://lexvo.org/ontology`

also ensures that these identifiers are dereferenceable and highly interconnected as well as externally linked to a variety of resources on the Web, e.g., DBpedia, YAGO (de Melo, 2015).

3.2. Glottolog

The ISO 639-3 standard offers practical solutions for many language identification needs. For the research of worldwide linguistic diversity, however, it is not granular enough. The goal of Glottolog is to collect and formalize information about languages and language resources (Nordhoff and Hammarström, 2011). It provides an exhaustive bibliographical coverage of the world's lesser-known languages that serves as an empirical ground for extensional definitions of languages and language classification.

Applying a novel approach, Glottolog defines a language by the set of documents which describe it. Relations between languages can thus be modeled in a set-theoretic manner. Glottolog introduces the term *languoid* as a cover term for dialect, language and language family. Every languoid is seen as a set that has its own URI and is annotated for ancestors, siblings, children, names, codes, geographic location and references. Subset and superset relations represent genealogical relationships.

Currently, Glottolog lists 320,559 references and defines 23,495 languoids. Languoids are modeled using SKOS and RDFS, and linked to ontologies like GOLD, Lexvo and geo. Furthermore, links to MultiTree, LL-Map, Ethnologue, ODIN, WALS, OLAC, and Wikipedia are provided (Nordhoff, 2012).

3.3. BLL Thesaurus

The language-related information in the BLL Thesaurus is organized in two top-level branches: *Indo-European languages* (825 items) and *Non-Indo-European languages* (1,417 items).

The structure and the granularity of the thesaurus are an outcome of the bibliography's specialization. English, German, and Romance linguistics belong to the focal areas of the BLL. The broad bibliographical coverage of these areas and the required detailed indexing explain the disproportional numbers: *German*, for example, has 115 subterms on five different hierarchical levels, while the subbranch *Afro-Asiatic languages* encompasses only 86 subterms[14].

As Table 1 shows, the subject terms representing language-related information are heterogeneous in nature. Generally, they can be grouped in three main types: individual language/variety, collection, and descriptive type. The first type refers to subject terms denoting a single language variety[15]. The language may be living, ancient, reconstructed or extinct as well as artificial or a sign language. A subject term may also denote a historical stage of a language, or refer to a code-switching phenomenon or register.

[14]In this family, Ethnologue lists 379 living languages. According to Glottolog, the family consists of 374 languages, not including the Omotic subfamily.

[15]Within the BLL Thesaurus, there is no consistent formal representation of the status of a language variety, i.e., whether it is considered a language or a dialect.

The second type denotes collections such as language families, subfamilies, dialectal groups or groupings by region. The remaining subject terms are of a mixed, descriptive type: they not only identify the language, but also describe the context of use (e.g., spatial or temporal aspects).

Type	**Description**	**BLL language identifier**
individual language / variety	living	Romanian
	ancient	Gothic
	extinct	Dalmatian
	historical	Old High German
	constructed	Klingon
	dialectal	Pantiscu
	code-switching	Trasianka
	register	Tok Master
	sign language	New Zealand Sign Language
	linguistic reconstruction of a common ancestor	Proto-Slavic
collection	language family	Afro-Asiatic languages
	language subfamily	Celtic languages
	geographical designation	Caucasian languages
	dialectal group	Scanian dialects
descriptive	individual language in a spatial context	German in Romania
	individual language in a temporal context	15th-18th century Italian

Table 1: Languages in the thesaurus

Compared to the other two models, the BLL Thesaurus seems to have more in common with Glottolog than with Lexvo. While Lexvo supplies a list of global IDs based on the codes defined by the ISO 639 registration authorities, the thesaurus subject terms are defined by an editorial team and encoded in a hierarchically structured way. The method applied for the addition of new BLL language identifiers resembles to some extent the resource-based approach of Glottolog. As a general rule, a new subject term can be included in the thesaurus only if the phenomenon or language in question has already been encountered in scientific publications indexed in the bibliography.

While the taxonomies within the relevant thesaurus branches follow mainly bibliographical principles and are only loosely based on the genetic relatedness between the language varieties, the Glottolog family trees are defined solely on genealogical principles.

The three repositories differ not only in structure, but also in scope and granularity, and none of them covers the other completely. Initial sampling showed that some BLL index terms might well find equivalents in Lexvo, but not in Glottolog (e.g., *Norn*, *Vandalic*). And the other way around, for some BLL language identifiers we can only find matches in Glottolog, but not in Lexvo (e.g., *Elfdalian*, *Hottentot Pidgin Dutch*). In the coverage of dialects, language families and subfamilies, Lexvo and Glottolog differ fundamentally. Determined to make as many BLL subject terms as possible interoperable, we conceptualized a mixed linking model between BLL, Lexvo and Glottolog.

4. Linking the BLL Ontology to Lexvo and Glottolog

When working on the interoperability of the subject terms from the thesaurus branch *Levels*, we applied the methodology introduced by Chiarcos et al. (2016) and briefly outlined in Section 2. This method involved two main steps: the remodeling of the thesaurus subject terms as ontological classes and their linking to the corresponding OLiA classes.
We decided to represent the subject terms as OWL classes for several reasons. First of all, applying OWL constraints facilitates the development of a consistent representation of the domain terminology and helps to uncover problematical modeling. OWL provides description logical operators to represent and to (partially) resolve conceptual overload and ambiguity as observed in the BLL Thesaurus[16]. Furthermore, the establishment of valid links to terminological repositories that adopt OWL as their primary formal framework requires an OWL modeling. And moreover, a fully-fledged ontology is suitable for reasoning and can be used to develop an ontology-based search function.
Since the OLiA Reference Model applied similar modeling principles, the linking between the BLL Ontology and OLiA was implemented by assigning BLL ontological concepts corresponding OLiA superconcepts by means of `rdfs:subClassOf` properties.
The BLL bibliographical entries were modeled in the BLL Ontology as instances of OWL classes (representing the corresponding BLL subject terms). With the thesaurus subject terms being empirically grounded in the bibliography, they could be interpreted – on an abstract conceptual level - as collections of references to linguistically relevant publications. The subject term *Auxiliary verb*, for example, could be seen as an abstraction of all the bibliographical references that concern this morphosyntactical category.
As we started focusing on the language-related information within the BLL Thesaurus, however, we had to reconsider some of the previous design decisions.

4.1. Instances

Terminological repositories that provide language identifiers often model those as instances of ontological classes. Lexvo, for example, uses the class `lvont:language` as defined in the Lexvo ontology, and Glottolog applies `dcterms:LinguisticSystem`, `gold:Language`, `gold:LanguageSubfamily`, `gold:LanguageFamily` and `gold:Dialect`.
We, by contrast, model the BLL language identifiers as ontological classes (e.g., a class `German` with subclasses `HighGerman` and `LowGerman`) following the previously described methodology. Thus, we provide a consistent ontological representation for all subject terms, avoid splitting the thesaurus into heterogeneous fractions, and are able to use the same standard reasoning principles across all branches.
However, the establishment of links between BLL classes and Lexvo or Glottolog instances may lead to formal inconsistencies. Therefore, we made several adjustments to our model.
First of all, we assume a new additional layer of individuals as specific realizations of each concept (in the SKOS version of the thesaurus) or class (in the BLL Ontology).
Within the framework of OLiA, a similar approach is employed: in the annotation models, there is a layer of instances referencing the actual tags of the annotated tag set[17]:

```
:VAINF a :AuxiliaryInfinitive;
  olia_sys:hasTag "VAINF"^^xsd:string .
```

Applying this approach to BLL not only improves its compatibility with OLiA on a conceptual level, but also facilitates a formally consistent representation of the links to Lexvo and Glottolog.
Because of these changes, we had to reconsider the modeling of the bibliographical entries. They are no longer represented as instances of the BLL classes, corresponding to the subject terms used for indexation. Instead, they are now modeled as individuals of the newly created class `bll:Title`. The relationship to the respective subject terms is expressed on the instance level by means of the `foaf:topic` property.

4.2. Class hierarchy

In the BLL Ontology, the classes are represented in a hierarchically structured way. When modeling the BLL language identifiers we face challenges specific to the domain of language classification.
The classification of the world's languages is a notoriously controversial field where political and social aspects often play a more important role than scientific facts. That is why, within the field of linguistics, the question whether a specific variety must be considered a language or a dialect is no longer of primary importance. With regard to the relationships between the languages, many linguists consider the family tree model the only approach of scientific relevance. However, due to lack of sufficient data, it is hardly applicable to all known human languages.
When describing or classifying languages, different traditions or naming conventions can pose further difficulties. The language presently known as Occitan, for example, has been described throughout the centuries as Provençal, Langue d'oc, Limousin or Southern Gallo-Romance (Blanchet and Schiffman, 2004).
The ontological modeling of the BLL language identifiers will take the expectations of the linguistic community into account, but it will not be based exclusively on genealogical principles. We aim at a classification that reflects established conventions and traditions, and, simultaneously, complies with the class structure underlying OWL. Furthermore, the nature and specificity of the BLL Thesaurus should be preserved where appropriate.
The BLL Ontology has been designed in a way that allows enhancement and seamless integration of additional thesaurus branches. The inclusion of the branches

[16]The subject term *Accusative*, for example, captures not only the case and its morphological marking, but also different syntactic aspects of the phenomenon.

[17]The example is taken from the annotation model of the STTS-Tagset, retrieved from `http://purl.org/olia/stts.owl`

Indo-European languages and *Non-Indo-European languages* is facilitated by the definition of a new top-level class `LanguageRelatedTerm` that serves as a structural anchor. The second hierarchical level comprises classes representing well established language families (e.g., `Indo-European languages`) as well as classes designating groupings based on typological criteria (e.g., `CreoleOrPidgin`), geographic location (e.g., `Australian languages`), or modality (e.g., `SignLanguage`).

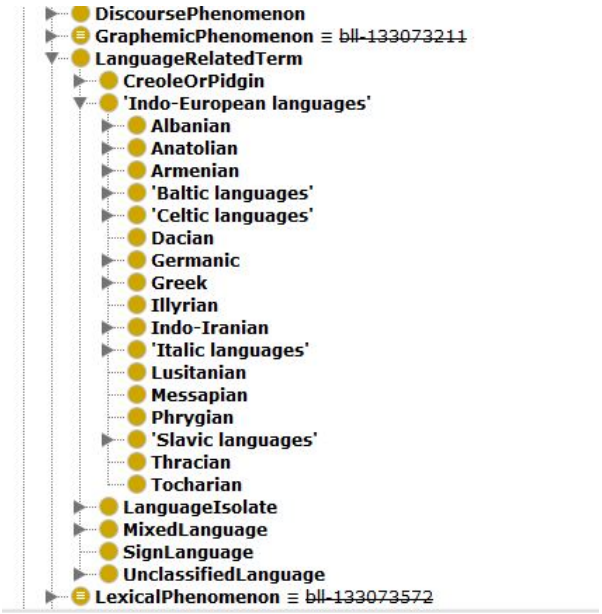


Figure 2: Ontological modeling of the thesaurus branch *Indo-European languages*: hierarchical structure.

Figure 2 illustrates the position `Indo-European languages` takes in the class hierarchy, its superclass and subclasses. Numerous structural reorganizations have to be undertaken in order to create a model that adheres to scientific insights and well established conventions. *Romance*, for example, represents a subconcept of *Indo-European languages* in the BLL Thesaurus. In the BLL Ontology, however, it is modeled as a subclass of `Italic languages` depicting the generally accepted classification for this language group.

Where appropriate, further elaboration of the hierarchical structure takes place. For example, in order to reflect the conventional subgrouping of the Slavic languages in East, South and West Slavic, we defined an intermediate level between the node `Slavic languages` and the individual representatives of the group (e.g., `Russian`, `Bulgarian`, and `Slovak`), and subdivided the languages accordingly.

The BLL Ontology adopts some of the general principles applied in the thesaurus. For example, there is no explicit differentiation between languages, dialects or historical varieties. Historical forms are usually coded as subterms of the respective language and normally "occupy" the same hierarchical level as the dialects of the given language.

4.3. Properties (for interlinking)

When establishing links between BLL concepts and language identifiers provided by Lexvo and Glottolog, we have to determine the nature of the relationship first and then find a fitting relational property for its formal representation. According to preliminary analyses, the entities in the different repositories demonstrate not only genuine identity and near-identity, but also more specific forms of similarity. In order to avoid constraint violations, we apply the property `owl:sameAs` only if the strict form of identity is guaranteed. Although the Lexvo Ontology already provides properties for notions of near-identity, those properties do not seem to suffice our modeling purposes. The simple hierarchy of `lvont:somewhatSameAs` and `lvont:nearlySameAs` is insufficiently distinguishable regarding the properties' strength and use cases. Hence we decided to extend the Lexvo Ontology with a mereologically defined set of properties.

Closely following the W3C best practices[18], we propose a more specific hierarchy for the relations between the language identifiers (Figure 3) which we define as subproperties of the Lexvo property `lvont:somewhatSameAs`. The `bll:overlaps` property is the most general of these and only states that there is at least a subset of entities (e.g., dialectal varieties, individual languages) which is contained in the definitions of both terms interlinked by it. It can therefore be asserted as a symmetric property. The `bll:hasPart` property and its inverse `bll:partOf` describe a transitive, full containment relation where one term contains all elements of the other term but not vice-versa .

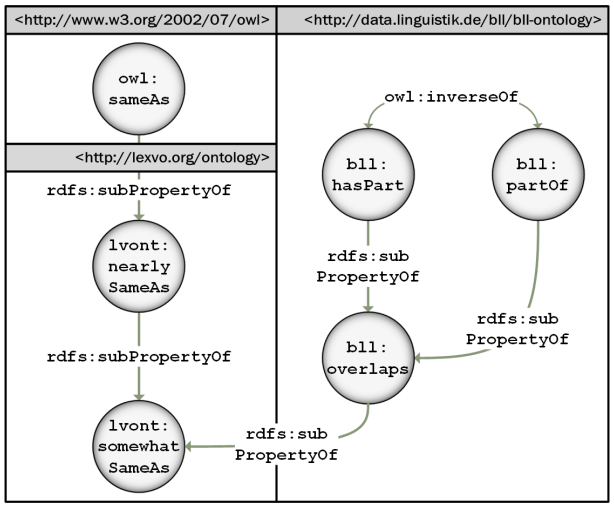


Figure 3: Properties for interlinking language terms

Here are some examples for the prototypical use of the newly defined properties.

The BLL concept Luwian, for instance, is a "container" subject term: it refers to the two varieties of Luwian known after the scripts in which they are written, namely

[18] `https://www.w3.org/2001/sw/BestPractices/OEP/SimplePartWhole/simple-part-whole-relations-v1.5.html`

Cuneiform Luwian and Hieroglyphic Luwian. Since ISO 639-3 defines distinct codes for both varieties, the following links between BLL and Lexvo are proposed[19]:

bll:Luwian `bll:hasPart` **lexvo:Cuneiform Luwian**
bll:Luwian `bll:hasPart` **lexvo:Hieroglyphic Luwian**

The property `bll:overlaps` will be applied mainly when linking BLL concepts with Glottolog language families or subfamilies. For example, both BLL and Glottolog define Anatolian as a subconcept of Indo-European. The BLL concept Anatolian, however, has five subconcepts (Hittie, Luwian, Lycian, Lydian, and Palaic), while the corresponding Glottolog subfamily Anatolian consists of ten individual languages (including the aforementioned five). Despite identical labels and hierarchical position, `owl:sameAs` is not an option here, since both concepts are not isomorphic on a graph-theoretic level. Therefore, the link will be established by means of `bll:overlaps`. But what if a mereological statement is not possible? The BLL language identifier Italo-Albanian, for example, is conceptually related to the Glottolog's languoid Arbëreshë Albanian. As divergent labels often indicate difference in the semantics, both terms cannot be equated. In similar cases Nordhoff and Hammarström (2011) use `skos:closeMatch`. Instead, we employ `lvont:nearlySameAs` as recommended by the Lexvo Ontology.

At the current stage, it is not possible to predict how often the mereological approach will be actually applicable. Taking into account the complexity of the field, using highly underspecified properties such as `lvont:somewhatSameAs` would be, of course, easier to implement. We prefer, however, more specific properties that semantically enrich the data and allow for more elaborate applications. After the linking to Lexvo and Glottolog has been completed, we will evaluate the newly defined properties with regard to frequency of usage and confidence of the relations.

5. Summary and Outlook

In this paper, we described the current efforts to expand the LLOD interface of the *Lin|gu|is|tik* portal, and the required extension of the existing architecture comprising the BLL Thesaurus and the BLL Ontology. In addition to the already implemented ontological representation of the branch *Levels* of language description and its linking to OLiA, we established a framework for the inclusion of the thesaurus branches containing language-related information. By modeling the BLL language identifiers as a hierarchy of ontological classes, we maintain a high level of consistency within the BLL Ontology. By defining individuals as specific realizations of the underlying concepts, we further improve the interoperability. Now, the LOD representation of the BLL can easily be connected to other terminological repositories or ontologies - on the instance level as well as on the class level.

Furthermore, we define a set of additional properties for interlinking language identifiers. We use the gradations of `owl:sameAs` specified in the Lexvo Ontology as a basis, extend them with mereologically defined subproperties and thus enable flexible and precise linking.

The implementation of the OWL modeling of the BLL language identifiers and their linking to Lexvo and Glottolog is work in progress[20]. As of February 2018, circa 85% (700 items) of the subject terms from the thesaurus branch *Indo-European languages* could be hierarchically organized and integrated in the BLL Ontology. For the necessary restructuring, 52 additional classes with no equivalents in the BLL Thesaurus were defined. Approximately 60% of the concepts subsumed under the ontological class `Indo-European languages` could be linked to at least one LLOD repository.

The targeted interoperability between the language-related information in the BLL Thesaurus and the LLOD repositories Lexvo and Glottolog will result in mutual benefits for both the *Lin|gu|is|tik* portal and the LLOD cloud.

Through the established links, the LOD-based search in the *Lin|gu|is|tik* portal will be enhanced: LOD resources that use Lexvo or Glottolog identifiers will become visible and searchable via the portal. Furthermore, the LOD-based search functionality will allow a fine-grained selection from the level of language families down to dialects using either the original hierarchy of the BLL Thesaurus or the manually annotated BLL Ontology. Additionally, the users of the *Lin|gu|is|tik* portal will profit from the integration of the bibliographical data listed by Glottolog.

Building on the LOD principles, we will gain access to information that can facilitate further functions. Glottolog, for instance, provides information about the geographical distribution of languages, and the spatial data can be used as basis for the development of a geographical search. Also, encyclopedic information about language varieties can be integrated through the links between Lexvo and DBpedia.

As of the LLOD community, it will benefit from the inclusion of a significant source of bibliographic material: the BLL lists currently more than 460,000 entries, and the records published before 2001 (circa 250,000 titles) are freely available as RDF. This can be very useful for a platform like Glottolog that employs a resource-based definition of language.

The interoperability of language-related information can also facilitate new applications. Presently, we are working on the implementation of an extended extraction algorithm for LLOD entries which automatically indexes existing resources with information about the languages they cover and the annotation models they use.

Acknowledgments

We would like to thank three anonymous reviewers for insightful feedback and helpful comments.

The research described in this paper is supported by the German Research Foundation (DFG) in the context of the

[19] For better readability in this quasi-Turtle example some local names have been replaced by their corresponding labels.

[20] The updated BLL Ontology and the mapping will be published under `http://data.linguistik.de/` by the end of the project.

project Specialised Information Service Linguistics (*Fachinformationsdienst Linguistik*, funding period 2017-2019).

6. References

Blanchet, P. and Schiffman, H. (2004). Revisiting the sociolinguistics of Occitan: a presentation. *International Journal of the Sociology of Language*, 169:3–24.

Chiarcos, C. and Sukhareva, M. (2015). OLiA - Ontologies of Linguistic Annotation. *Semantic Web Journal*, 6:379–386.

Chiarcos, C., Fäth, C., Renner-Westermann, H., Abromeit, F., and Dimitrova, V. (2016). Lin|gu|is|tik: Building the Linguist's Pathway to Bibliographies, Libraries, Language Resources and Linked Open Data. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4463–4471, Portorož, Slovenia, May. European Language Resources Association (ELRA).

de Melo, G. (2013). Not Quite the Same: Identity Constraints for the Web of Linked Data. In Subbarao Kambhampati (Conference Chair), et al., editors, *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, pages 1092–1098, Bellevue, Washington, USA, July. Association for the Advancement of Artificial Intelligence.

de Melo, G. (2015). Lexvo.org: Language-Related Information for the Linguistic Linked Data Cloud. *Semantic Web Journal*, 6:393–400.

Dimitrova, V., Fäth, C., Chiarcos, C., Renner-Westermann, H., and Abromeit, F. (2016). Building an Ontological Model of the BLL Thesaurus: First Steps Towards and Interface with the LLOD Cloud. In John P. McCrae, et al., editors, *Proceedings of the 5th Workshop on Linked Data in Linguistics: Managing, Building and Using Linked Language Resources (LDL-2016)*, pages 50–58, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Nordhoff, S. and Hammarström, H. (2011). Glottolog/Langdoc: Defining dialects, languages, and language families as collections of resources. In Tomi Kauppinen, et al., editors, *Proceedings of the First International Workshop on Linked Science 2011*, pages 53–58, Bonn, Germany.

Nordhoff, S. (2012). Linked Data for Linguistic Diversity Research: Glottolog/Langdoc and ASJP Online. In Christian Chiarcos, et al., editors, *Linked Data in Linguistics: Representing and Connecting Language Data and Language Metadata*, pages 191–200. Springer, Berlin, Heidelberg.

Browsing and Supporting Pluricentric Global Wordnet, or just your Wordnet of Interest

António Branco, Ruben Branco, Chakaveh Saedi and João Silva
University of Lisbon
NLX-Natural Language and Speech Group, Department of Informatics
Faculdade de Ciências
Campo Grande, 1749-016 Lisboa, Portugal
{antonio.branco, ruben.branco, chakaveh.saedi, jsilva}@di.fc.ul.pt

Abstract

In this paper we proceed with a systematic gathering of design requirements for wordnet browsers that permit to consult the content of wordnets. This is undertaken together with a review of the functionalities of existing browsers. On the basis of this analysis, we present a new wordnet browser we developed that meets these requirements and thus complies with the most ample range of design features. This is an open source browser that is freely distributed and can be reused by anyone interested in doing research on or just using wordnets. We also introduce the notion of a pluricentric global wordnet, for whose undertaking this new advanced browser appears as an important instrument and motivation. This is a promising operative conception for a bootstrapped yet effective process towards the ultimate global wordnet, where all individual wordnets from all languages are meant to eventually converge together, in spite of the plurality of their formats, licenses, depth, etc. that is intrinsic to an inherently plural endeavor undertaken by multiple actors under multiple constraints across the world.

Keywords: wordnet, wordnet browser, pluricentric global wordnet

1. Introduction

This paper reports on the results of the work initially motivated by the aim of making available an online service that would permit to consult and browse a wordnet we have been developing.

Our work started by looking for and trying to reuse some pre-existing browser that could be adopted with as little adaptation effort as possible. As we proceeded, we found ourselves pondering on the needs and design requirements that such an application should satisfy and the work eventually took the unanticipated route that would lead to the building of a new browser from scratch.

This had yet further unanticipated consequences, beyond the instrumental goal that triggered this implementation of a new wordnet browser. The analysis of the requirements on how wordnets should better be presented set us on the path to ponder on how this could serve a wider purpose of an as much inclusive support as possible for wordnets and for their flexible integration with each other. That is what we are trying to capture with the notion of a pluricentric global wordnet.

In the next Section 2., we present the list of design requirements that guided the development of the browser eventually implemented, which we collected after the analysis of existing browsers, briefly described in Section 3..

In Sections 4. and 5., the first version of the browser is presented in its functional and technical aspects, respectively. A second version is described in Section 6., which motivates the notion of a pluricentric global wordnet, whose browsing it can support.

Sections 7. and 8. close the paper, respectively, with indication of the resources being distributed, and with some concluding remarks and notes on future work.

2. Browser requirements

In our initial quest for finding a reusable wordnet browser, we end up coming across a number of browsers (addressed in the next section) that had some features that others were lacking and vice-versa. As a result of this exercise, we eventually drew a list of all those should-have requirements, which was further extended with possible other features that a general purpose browser should have. It is that list of requirements that we present now:

+ *Detached from any editor*

  In order to maximize its re-usability, the browser should be detached from any particular wordnet editor so that it allows to peruse wordnets built with the help any editing application or approach.

+ *Detached from any online service*

  Likewise, the browser should be detachable from any particular online service or website, so that it can be reused locally or to support other online sites or services.

+ *Detached from any wordnet*

  The browser should allow to be associated with any wordnet, modulo possible format adaptation or conversion needed, to allow for its perusal.

+ *Compatible with any license*

  The licensing terms of the browser should make its usage compatible with wordnets distributed under any license, and permit it to be distributed together with the wordnets if convenient.

+ *Peruses one or several wordnets*

The browser can be used to permit the consultation of one wordnet in isolation, or the perusal of several wordnets accessible to the user in the same working space and connected to each other.

+ *Displays translations into selected languages*

  In case several wordnets are accessible for browsing, it should be possible to select the subset of the ones from which results should be displayed that are the translations of the lemma searched.

+ *Performs multilingual search*

  Also when several wordnets are available for browsing in the same working space, it should be possible to perform multilingual searches, that is search by entering a lemma in any of those languages without the need of any previous selection of the intended language.

+ *Peruses wordnets aligned with any pivot language*

  It should allow to peruse any set of wordnets that happen to be aligned pairwise or with the help of any pivot languages, and not only those aligned with English.

+ *Adheres to a de facto standard format*

  To maximize its re-usability, the browser is compliant with a de facto standard wordnet format, and should be equipped with as many converters to that format as possible.

+ *Searches by input lemma*

  The browser should allow to have access to a given concept/synset by retrieving it on the basis of a search triggered by one of its lemmas.

+ *Displays lemmas targeted by selected relations*

  It should be possible to have lemmas displayed that are the target of relations outgoing from a source lemma, and have the possibility to select the relations of interest for that to happen.

+ *Displays lemmas transitively targeted*

  The lemmas transitively targeted by outgoing relations should also be possible to be displayed.

+ *Displays definitions, examples, etc*

  Besides lemmas, displaying definitions and examples should be possible.

+ *Multilingual user interface*

  The browser should ensure a multilingual interface where the menu and instruction can be displayed in different languages, with the inclusion of further languages being possible.

+ *Web-based platform independent*

  In order to be accessible by the largest possible type or number of potential users (irrespective of the operative systems or platforms they may be using), in the largest number of possible locations, the wordnet browser should be web-based and independent of the web-browsing application at use.

+ *Code open to reuse and extension*

  The browser should be cost-free and openly reusable, distributed with a permissive license.

3. Reusable browsers

There is a variety of applications and browsers for wordnets, most of which work as a dictionary providing the user with the "synonyms" (e.g WordVis (Vercruysse, 2012)) or in some cases "synonyms" as well as "antonyms" (e.g. Synonym-net (Network, 2017)) of the searched word.
Bearing in mind that a wordnet includes many informative relations, and not only synonymy and antonymy, we focused on applications and browsers with a more extensive data coverage. Among the wordnet browsers, most are to be used online while the rest can be downloaded to be installed and run locally. Some are originally designed for the special wordnet under development in the respective research team while the others are more general and compatible with other wordnets. In what follows, a brief overview of major wordnet browsers is presented.

- DEBVisDic (Horák et al., 2006) is one of the most common tools for developing and browsing wordnets. It is widely used since it is not developed for a particular wordnet. The user interface is in English and there is a converter available to change the data in Princeton format into the XML format accepted by DEBVisDic. It allows to search in multiple wordnets. A major limitation of this system is that it does not support further installations given all the data must be uploaded and saved in its supporting server.

- WordNetLoom (Piasecki et al., 2013), designed and developed as part of the Polish WordNet project (Derwojedowa et al., 2008), is a tool for editing and browsing wordnets. It offers a variety of useful features specially for editing a wordnet. As a browser, it has a dynamic graphical view of the search results which supports connections among various wordnets. It provides the user with several search options from the lemma to POS and relation types. It is accessible both online and offline. However, this browser is compatible with the database created by the polish editor, which makes it not independent from this editor. It must be mentioned the current online version only includes English and Polish WordNet.

- SloWTool (Fišer and Novak, 2011) is also a well known tool for browsing, editing and visualizing wordnets. It is independent of any particular web browser and can be used online or in a local installation. Although it provides the user with several interesting search options, it does not support the mapping among wordnets of different languages.

- English Princeton Browser, as a part of the English Princeton wordnet (Fellbaum, 1998), is publicly available to be reused. It is designed to browse through

synsets and senses in any wordnet compatible with the Princeton format. However, like the previous example, it does not support connections among wordnets of different languages.

- Visuwords (Critchfield, 2017) is an online graphical wordnet browser that employs colors and shapes to distinguish between synsets in various parts of speeches and types of semantic relations. Although very user friendly, this browser does not support multilingual wordnet browsing either.

- Open Multilingual Wordnet (Bond and Foster, 2013) connects a large number of wordnets from different languages while using the English Princeton wordnet as the pivot one. The wordnets it resorts to have permissive licences for derivatives and redistribution and searching through the browser shows results in all their languages. However the source code of the browser itself is not available to be reused, and it is a browser that in any case offers no options to peruse wordnets on the basis of, direct or transitive, semantic relations.

- Multi-WordNet (Pianta et al., 2002) is a well-known project aiming at aligning wordnets of different languages. Currently, it covers seven different languages — English, Hebrew, Italian, Latin, Portuguese, Romanian and Spanish. Its browser is accessible online and provides the users with the possibility of navigating through the wordnet graph in addition to showing the aligned data in all included wordnets. As in the previous case, this browser is also among the non-reusable ones in different installation given its source code is not distributed.

- Wnbroswer is the browser for the Romanian wordnet (Tufiş et al., 2008). This system receives data in the Princeton format and is freely available to be reused. It contains several useful options for search and presents a graphical representation of the search result. Even though it has many good features, the static search result makes it impossible to navigate through the wordnet graph. Two other limitations, considering the items listed in 2., are the need to run it in a local installation and the lack of connections among wordnets of different languages.

- GermaNet-Explorer (Finthammer et al., 2008) is a software to visualize the German wordnet. It displays word senses and synsets and can present all semantic relations among them. This browser is freely available for academic use. However, it needs to be installed and run only locally and does not provide the user with the option to search across wordnets of different languages.

- University of Rochester Computer Science wordnet browser (Ferguson, 2010) can be installed on different operating systems (Windows, Mac or Ubuntu). It uses wordnet files compatible with the Princeton Format and contains a simple interface to search for a word in addition to options for the user to choose what to receive as search results (e.g. definitions, examples, sense numbers, links to related synsets, etc.). A limitation of this browser is the need to run on a local installation. Furthermore, there is no options for connections among different wordnets.

4. Browser: Functional matters

While aiming at responding to concrete needs and constraints from their specific contexts of development, the different publicly available browsers briefly reviewed in Section 3. contributed to the advancement of the state of the art in this respect by introducing novel functionalities. To a very large extent, the set of requirements in Section 2. results from gathering all such nice design features from previous browsers.

As it turns out from the analysis in Section 3., every existing browser happens not to meet one or other of the requirements lined up in Section 2.. This provided the motivation for our development of a new browser that seeks to meet all these requirements, to whose presentation we turn now.

The user interface is rendered in a web page that is depicted in Figure 1. There is a second page listing the wordnets that may be at use in a given installation of the browser and respective references, which can be accessed by pressing the button on the right of the search button. An installation of this browser [1]can be experimented with here: http://194.117.45.198:9001.

At the start, the user has two fields he can fill in. In one of these fields, he can type the lemma to search for, and press the button to enter that query.

Right below that field, in the second field the user can select, or deselect, the languages to which the retrieved lemma should be translated and whose translations should be displayed with the outcome of the search. Multiple languages are supported at the same time.

When entering a lemma to be searched for, if that lemma is included in the wordnet being browsed, all concepts/synsets that include that word are displayed, one per line.

The search can be continued from these results that are shown. Each of the synsets is prefixed by the string 'rels', standing for 'related concepts', which if pressed, permits to toggle a menu containing links to all types of the semantic relations available for that synset. That menu also includes a link for a translation of that synset to be displayed, if such translation exists.

[1]While many more wordnets are being prepared to be included, at the moment of submitting this paper, the current installation of this browser makes use of the following ones: Finn Wordnet (Lindén and Carlson, 2010), Arabic Wordnet (Elkateb et al., 2006), Portuguese Wordnet (Branco et al., 2009), Catalan, Basque, Galician and Spanish Wordnet (Gonzalez-Agirre et al., 2012), Chinese Wordnet (Wang and Bond, 2013), Danish Wordnet (Pedersen et al., 2009), Farsi Wordnet (Montazery and Faili, 2010), Hebrew Wordnet (Ordan and Wintner, 2007), Indonesian and Malaysian Wordnet (Noor et al., 2011), Japanese Wordnet (Isahara and Kanzaki, 2008), Nynorsk and Bokmål Wordnet (Fjeld and Nygaard, 2009), Polish Wordnet (Piasecki et al., 2009), Slovene Wordnet (Fišer et al., 2012), Swedish Wordnet (Borin et al., 2013), Thai Wordnet (Thoongsup et al., 2009).

By clicking on a given type of semantic relation, the synsets that are the target of that relation are displayed (or hidden) below the respective source synset. Indentation is used to represent the successive directed arcs on the semantic graph.

By clicking on the translation link, in turn, translations are showed (or hidden) on the right side of the page, with each translation in a separate box, and with boxes stacking as a column if multiple languages are selected. The language name is included in the header of each box, with a Wikipedia link to know more about it.

The current pilot installation of the wordnet browser used for testing and debugging (depicted in Figure 1) resorts to the Princeton wordnet for the semantic network that is to be searched for. And for the translation functionality, it resorts to several wordnets of other languages, whose list can be found by opening a menu through the button to the left of the search button.

This pilot installation is online only for demo purposes. The head picture with "MY WORDNET" is just a placeholder where the logo of the wordnet being offered to be browsed can be inserted, together with the link to the web page of that wordnet.

The code of the browser can be downloaded from here: https://github.com/nlx-group/LX-WordNetBrowser. It is being distributed under as open source as an MIT License.

The development of an installation of the browser based on a particular wordnet is quite straightforward. With respect to the wordnet to be browsed, it only needs that the files of the Princeton Wordnet used in the pilot installation above be replaced by the respective files of that wordnet of interest. Similar procedure is needed with respect to the files for the translations desired.

Concerning the language of the user interface, all that is required is that the few English words of the menu be linked (by straightforwardly editing the code) to their translation equivalents in the language chosen for the user interface, if such translation is not already existent in the current set of user interface supported languages.

We will return to the issues of the support for multilinguality and of the multilingual interface below in Section 6..

5. Browser: Technical matters

The browser runs on a client-server architecture based on Apache and Django framework, and is implemented by resorting to Python, JavaScript and some complementary libraries, viz. jQuery, Select2 and Bootstrap.

Looking from the side of the web page that acts as the user interface, the interactions with the server are handled using JavaScript and its libraries. As the information is received from the user session, the DOM (Document Object Model) is processed, which in turns changes the HTML displayed on the screen.

Together with Django, Apache HTTP web server is used to exchange the files (javascript, html, css and assets) with the client as it takes care of the in- and out-going traffic.

While Apache is acting as the web server, the information retrieval is done with the help of Python. The request is analyzed through Django's request API and the appropriate Python script is run, whose outcome is sent over to the client for the JavaScript code to handle and display.

Concerning wordnet data, two formats are used. One of them is the Princeton wordnet format.[2] This is the format needed to support the search of the information regarding the semantic relations between the words that are relevant to the lemma entered in the query.

The other format concerns the mapping of a lemma in some language to their counterpart synset in English, and is used for finding the translations of the input lemma. In this format, there is one lemma per line, which is tab-separated from its translational equivalent.[3]

In our current implementation, the tab files resort to English as the pivot language. This does not hamper to opt for other languages to function as a pivot language by simply replacing the tab-separated files with the ones suited for that purpose.

In order to minimize the time spent accessing the data, the result of each previous search has its offsets cached. If the user wishes to explore a given synset that was retrieved, its cached offset helps to perform a direct lookup. These caches are kept in the lines of the list with the search results: Each line of the list contains a lemma, and has its 'Class' set as the offset of the respective wordnet.

The wordnet data is kept in memory in native Python data structures, such as dictionaries, to speed up retrieval of information in each request. The data is kept separate from the browser service by running in parallel through a local and private XML-RPC server. The queries are submitted to this server, and their outcome are used to proceed with the handling of client requests.

To fully support multilinguality, the possibility of integrating as many languages as needed is vital. The queries, the data processed and the information displayed to the user must be able to encode any character featured in the alphabets of the languages currently supported by the pilot installation of the browser and any future ones. This is achieved by having all the processes encoded in UTF-8, allowing to encode Latin alphabets and other more logo-graphic alphabets and scripts.

Due to a simpler code base that the first installation features, it is a good choice for development and usage with a single language system, compared to the extended code base in the second installation, that supports multilinguality.

6. Towards a pluricentric global wordnet

By allowing to look also for the translations of the lemma searched for in the wordnet of interest, the browser presented in Section 4. permits the perusing of any wordnet, on which it is based, in a multilingual setting. However, this still offers a quite limited compliance with a truly multilingual browser.

[2] This format is described in http://wordnet.princeton.edu/man/wndb.5WN.html

[3] This format is described in http://compling.hss.ntu.edu.sg/omw/. The files in this format used in the pilot installation of the browser were obtained from http://compling.hss.ntu.edu.sg/omw/summx.html.

This motivated another browser that, in addition to all design features of the one previously presented, has an enhanced multilingual design. Its interface page is depicted in Figure 2. An installation of this browser can be experimented with here: http://194.117.45.198:9002.

First, it accepts that the lemma entered to be searched is from any language, and ensures that the language of that query is automatically identified and the respective synset(s) is retrieved from the respective wordnet.

Second, this other version of the browser also allows for the customization on the fly of the user interface. Above the logo at the top of the page, there is a menu that permits to chose the language in which the interface should be displayed.[4]

Third, it should also be noted that by its very own design, this browser (as well as the previous one) further supports multilinguality by permitting to peruse sets of wordnets from different languages that happen to be assembled and linked with each other around a pivot language other than English.

Importantly, it permits the perusal of wordnets that are not linked to the Princeton wordnet [5] or to any other wordnet in other language. And it permits to search and display synsets that are not linked to synsets in the Princeton wordnet (even if other synsets of the same wordnet are).

This is the kind of browser that is closer than any other before to be fit and useful to peruse the ultimate Global Wordnet — eventually encompassing all wordnets of all languages — one day when it comes into existence. While that day has not arrived, this browser is a most useful instrument to peruse ensembles of wordnets whose different composition respond to diverse interests, needs, constraints or limitations.

For instance, there may be the interest of gathering just the set of wordnets of the languages belonging to the same language family; or the wordnets built around a given pivot language, etc.

And there certainly exists the need to allow the browsing, in a multilingual setting, of a given wordnet with some licensing restrictions concerning its distribution. In a counterfactual setting that imposes that all wordnets must be redistributed under the same license to be displayed online all together, that would not be feasible.

While the conditions are being prepared but not yet fulfilled to eventually there being a Global Wordnet with all world wordnets connected among themselves, the present browser is an important instrument for there being multiple installations and sites of multilingual wordnets of multiple sizes and compositions, in what we would designate as an interim pluricentric global wordnet.

Importantly, this browser is also a novel instrument to help to browse and support any new multilingual wordnet that will be set up as an alternative to an eventual multilingual wordnet whose authors, based on their authorship of that derivative work, may have it accessible for perusal behind a non detachable browser that cannot receive further independent installations by other research teams — or may decide to lock that derivative work in some restrictive license; or even not to distribute it at all (even though the individual wordnets they are based on have permissive licenses, and may be not even conspicuously cited).[6]

Pluricentric global wordnet appears thus as a promising operative conception for a bootstrapped yet effective process towards an ultimate global wordnet, where all wordnets from all languages are meant to eventually converge together. It allows to progress in this direction in spite of the plurality of formats, licenses, maturity, volume, coverage, depth, etc. of individual wordnets, thus supporting an effective way to cope with the lack of their harmonization — in the present days and in the foreseeable future — that it is intrinsic to an inherently plural endeavor undertaken by multiple actors in multiple circumstances under multiple motivations and constraints across the world.

Pluricentric global wordnet allows also to circumvent the illusory appeal of a monolithic derivative endeavor that gathers as much wordnets as possible under some harmonized ensemble that ends up however trapped behind the wall of some non permissive license or even of non allowed redistribution.

7. Resources Distributed

The software whose development is reported in the present paper is distributed at https://github.com/nlx-group/LX-WordNetBrowser under an MIT license.

This application is termed as **LX-WordNetBrowser** and users of this resource should refer it by citing the present paper.

8. Conclusions and Future Work

This paper presented a new wordnet browser that satisfies the ample design requirements that were systematically gathered after an analysis of the publicly available wordnet browsers. This is an open source browser that is freely distributed and can be reused by anyone interested in doing research on or using wordnets.

In one of its versions, this software is distributed under an open source license to support the installations of independent web based browsers for any wordnet.

In another version, distributed under identical license, this software supports fully-fledged multilingual installations of wordnet browsers, and it is thus contributing to the advancement of a pluricentric global wordnet.

The distribution under an open source license meets the requirement *Code open to reuse and extension.*

Both versions, successfully follow the requirements set forth in Section 2., as described in Section 4. and 5..

As of now, it can be applied to the multiple existing wordnets in the Princeton wordnet format, the de facto standard format it is based on. By using this format, and given the many tools that exist to convert from other formats into it, it is possible to detach the browser from the wordnet editors. Its full potential will be deployed when there will be more converters from any wordnet format to that format, than

[4] At the moment of the submission of this paper, this design feature is being expanded through crowd sourcing, where volunteers can offer the translation of menu items in their languages.

[5] Naturally, provided that they are in the de facto standard format the browser is based upon, that is the Princeton format.

[6] The BabelNet approach in this respect comes to mind here.

those that already exist today. Important future work will thus consist in developing more converters of this sort, further detaching the browser from any wordnet.

In a more ambitious and long term perspective, it would be interesting to connect different installations of this browser through web services such that the content of every wordnet available to be searched in some of its installations would be available to be searched from any other installation thus connected. This will help to move from the browsing of an interim pluricentric global wordnet to the browsing of a fully global wordnet.

A pluricentric global wordnet appears as an operative conception for a bootstrapped yet effective process towards the ultimate global wordnet, where all individual wordnets from all languages are meant to eventually converge together, despite their plurality that is inherent to a multifarious endeavor undertaken by multiple actors under multiple constraints across the world.

Acknowledgment

The present research was partly supported by the Infrastructure for the Science and Technology of the Portuguese Language (CLARIN Língua Portuguesa), and by the ANI/3279/2016 grant.

Bond, F. and Foster, R. (2013). Linking and Extending an Open Multilingual Wordnet. In Proc. 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1352–1362.*

Borin, L., Forsberg, M., and Lönngren, L. (2013). Saldo: a touch of yin to wordnet's yang. Language resources and evaluation*, 47(4):1191–1211.*

Branco, A., Costa, F., Ferreira, E., Martins, P., Nunes, F., Silva, J. a., and Silveira, S. (2009). Lx-center: a center of online linguistic services. In Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, pages 5–8. Association for Computational Linguistics.*

Critchfield, T. S. (2017). Visuwords®: a handy online tool for estimating what nonexperts may think when hearing behavior analysis jargon. Behavior Analysis in Practice*, pages 1–5.*

Derwojedowa, M., Piasecki, M., Szpakowicz, S., Zawisławska, M., and Broda, B. (2008). Words, concepts and relations in the construction of polish wordnet. In Proceedings of the Global WordNet Conference (GWC 2008), Seged, Hungary*, pages 162–177.*

Elkateb, S., Black, W., Rodríguez, H., Alkhalifa, M., Vossen, P., Pease, A., and Fellbaum, C. (2006). Building a wordnet for arabic. In Proceedings of The fifth international conference on Language Resources and Evaluation (LREC 2006)*, pages 22–28.*

Fellbaum, C. (1998). WordNet*. Wiley Online Library.*

Ferguson, G. (2010). University of rochester computer science (urcs) wordnet browser. http://www.cs.rochester.edu/research/cisd/wordnet.

Finthammer, M., Cramer, I. M., et al. (2008). Exploring and navigating: Tools for germanet. In Proceeding of the 6th International Conference on Language Resources and Evaluation (LREC-2008).

Fišer, D. and Novak, J. (2011). Visualizing slownet. Proceedings of the Electronic Lexicography in the 21st Century (eLex 2011)*, pages 76–82.*

Fišer, D., Novak, J., and Erjavec, T. (2012). slownet 3.0: development, extension and cleaning. In Proceedings of 6th International Global Wordnet Conference (GWC 2012)*, pages 113–117.*

Fjeld, R. V. and Nygaard, L. (2009). Nornet—a monolingual wordnet of modern norwegian.

Gonzalez-Agirre, A., Laparra, E., and Rigau, G. (2012). Multilingual central repository version 3.0: upgrading a very large lexical knowledge base. In Proceedings of 6th International Global Wordnet Conference (GWC 2012)*, page 118.*

Horák, A., Pala, K., Rambousek, A., and Povolný, M. (2006). Debvisdic–first version of new client-server wordnet browsing and editing tool. In Proceedings of the Third International WordNet Conference (GWC 2006)*, pages 325–328.*

Isahara, Hitoshi, B. F. U. K. U. M. and Kanzaki, K. (2008). Development of the japanese wordnet. In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. European Language Resources Association (ELRA).*

Lindén, K. and Carlson, L. (2010). Finn wordnet-wordnet på finska via översättning. LexicoNordica*, (17).*

Montazery, M. and Faili, H. (2010). Automatic persian wordnet construction. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 846–850. Association for Computational Linguistics.*

Network, S. (2017). Welcome to synonyms.net. http://www.synonyms.net/.

Noor, N. H. B. M., Sapuan, S., Bond, F., et al. (2011). Creating the open wordnet bahasa. In Pacific Asia Conference on Language, Information and Computation (PACLIC)*, pages 255–264.*

Ordan, N. and Wintner, S. (2007). Hebrew wordnet: a test case of aligning lexical databases across languages. International Journal of Translation*, 19(1):39–58.*

Pedersen, B. S., Nimb, S., Asmussen, J., Sørensen, N. H., Trap-Jensen, L., and Lorentzen, H. (2009). Dannet: the challenge of compiling a wordnet for danish by reusing a monolingual dictionary. Language resources and evaluation*, 43(3):269–299.*

Pianta, E., Bentivogli, L., and Girardi, C. (2002). Multiwordnet: developing an aligned multilingual database. first global wordnet conference (gwc 2002). India, January.

Piasecki, M., Broda, B., and Szpakowicz, S. (2009). A wordnet from the ground up*. Oficyna Wydawnicza Politechniki Wrocławskiej Wrocław.*

Piasecki, M., Marcińczuk, M., Ramocki, R., and Maziarz, M. (2013). Wordnetloom: a wordnet development system integrating form-based and graph-based perspectives. International Journal of Data Mining, Modelling and Management*, 5(3):210–232.*

Thoongsup, S., Robkop, K., Mokarat, C., Sinthurahat, T., Charoenporn, T., Sornlertlamvanich, V., and Isahara, H.

(2009). Thai wordnet construction. In Proceedings of the 7th workshop on Asian language resources*, pages 139–144. Association for Computational Linguistics.*

Tufiş, D., Ion, R., Ceauşu, A., and Ştefănescu, D. (2008). Racai's linguistic web services. In Proceedings of the 6th international conference on Language Resources and Evaluation (LREC 2008)*, pages 28–30.*

Vercruysse, S. (2012). About wordvis. http://wordvis.com/about.html.

Wang, S. and Bond, F. (2013). Building the chinese open wordnet (cow): Starting from core synsets. In Proceedings of the 11th Workshop on Asian Language Resources, a Workshop at IJCNLP*, pages 10–18.*

Annex

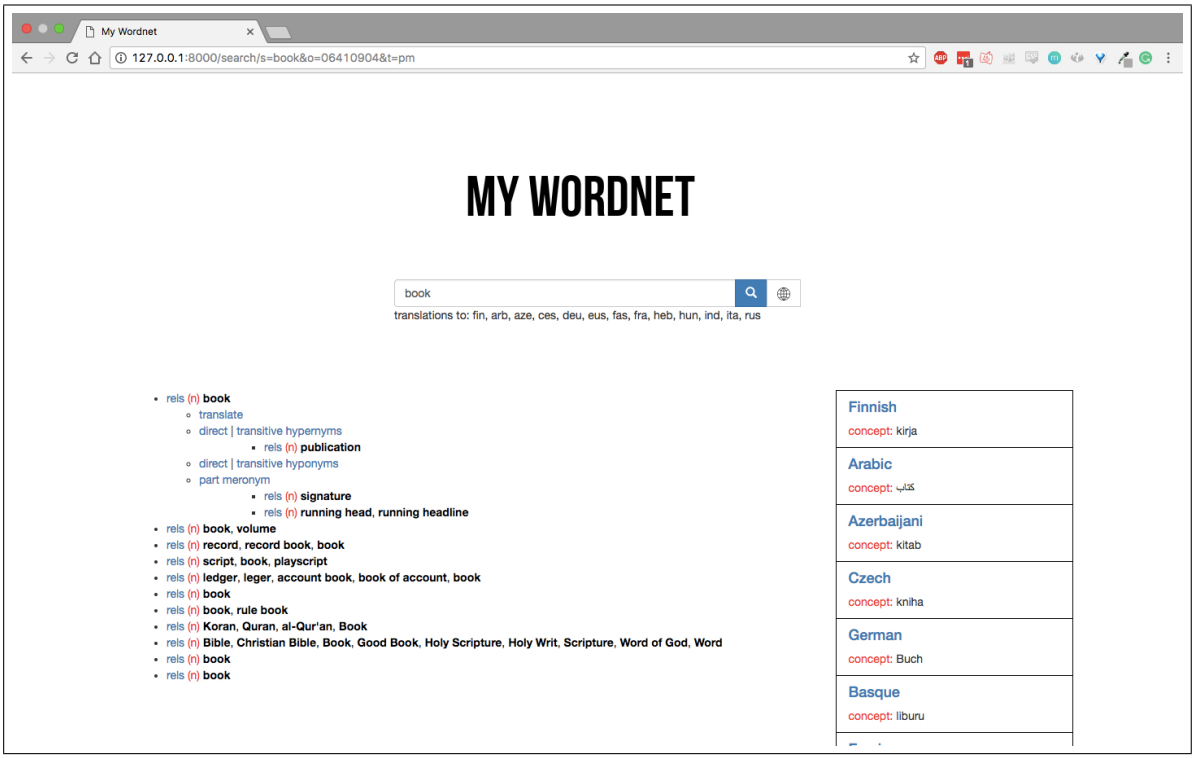


Figure 1: ”My Wordnet” installation of the wordnet browser: example with Princeton Wordnet as the wordnet to be browsed.

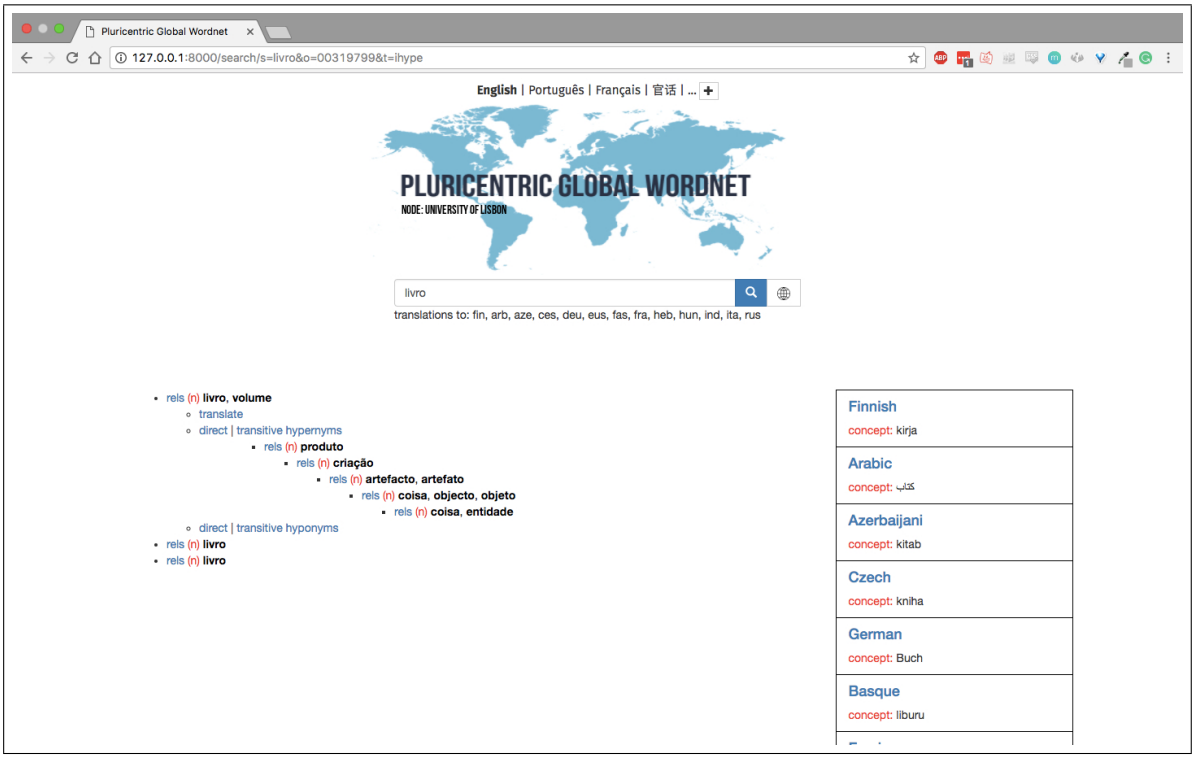


Figure 2: ”Pluricentric” installation of the wordnet browser: example with Portuguese selected to be browsed, and with English for the user interface.

Cross-checking WordNet and SUMO Using Meronymy

Javier Álvez, Itziar Gonzalez-Dios, German Rigau
LoRea Group, IXA Group, IXA Group
University of the Basque Country (UPV/EHU)
javier.alvez@ehu.eus, itziar.gonzalezd@ehu.eus, german.rigau@ehu.eus

Abstract

We report on the practical application of a black-box testing methodology for the validation of the knowledge encoded in WordNet, SUMO and their mapping by using automated theorem provers. Our proposal is based on the part-whole information provided by WordNet, out of which we automatically create a large set of tests. Our experimental results confirm that the proposed system enables the validation of some pieces of information and also the detection of missing information or inconsistencies among these resources.

Keywords: Meronymy, Knowledge validation, Automated Theorem Proving

1. Introduction

Despite being created manually, knowledge resources such as WordNet (Fellbaum, 1998) and SUMO (Niles and Pease, 2001) are not free of errors and inconsistencies. Unfortunately, improving, revising and correcting such large knowledge bases is a never-ending task that has been mainly carried out manually. A few automatic approaches have also been applied focusing on checking certain structural properties on WordNet e.g. Daudé et al. (2003) and Richens (2008) or using automated theorem provers on SUMO e.g. Horrocks and Voronkov (2006) and Álvez et al. (2012). Just a few more have studied automatic ways to validate the knowledge content encoded in these resources by cross-checking them. For instance, Álvez et al. (2008) exploit the EuroWordNet Top Ontology (Rodríguez et al., 1998) and its mapping to WordNet for detecting many ontological conflicts and inconsistencies in the WordNet nominal hierarchy.

In Álvez et al. (2017), we proposed a method for the automatic creation of *competency questions* (CQs) (Grüninger and Fox, 1995), which enabled to evaluate the competency of SUMO-based ontologies. Our proposal was based on several predefined question patterns (QPs) that were instantiated using information from WordNet and its mapping into SUMO (Niles and Pease, 2003). In addition, we described an application of automated theorem provers (ATPs) for the automatic evaluation of the proposed CQs. This proposal was used in Álvez and Rigau (2018) for a preliminary validation of WordNet, SUMO and their mapping.

The main contribution of this paper is a proposal for the automatic validation of WordNet, SUMO and their mapping. For this purpose, we create a new set of CQs that is obtained on the basis of the part-whole data of WordNet and which is an improved version of the set proposed in Álvez and Rigau (2018). By means of our proposal, we demonstrate the practical capabilities of the method introduced in Álvez et al. (2017) for the automatic detection of semantic agreements and inconsistencies in large-scale knowledge resources. For example, the knowledge encoded in the WordNet relations *part*(*elementary_particle*^1_n,*atom*^1_n), *member*(*national*^1_n,*country*^3_n) and *substance*(*cartilage*^1_n,*cartilaginous_structure*^3_n) can also be inferred from SUMO. On the contrary, according to our interpretation of the meronymy relations of WordNet, the knowledge in the relations *part*(*cell*^2_n,*cell_nucleus*^1_n) and *substance*(*grape*^1_n,*wine*^1_n) is incompatible with SUMO. In addition, our proposal enables the detection of missing knowledge: for instance, WordNet relates *waist*^1_n and *torso*^1_n by *part*, but the resulting conjectures are not proved to be entailed by SUMO in our experiments using ATPs. By a manual inspection of the ontology, we discover that this issue is due to the fact that the SUMO concepts connected to *waist*^1_n and *torso*^1_n according to the mapping between WordNet and SUMO —*Waist* and *Torso* respectively— are not related in SUMO.

Outline of the paper. In the following three sections, we introduce WordNet (Section 2), SUMO (Section 3), and their mapping (Section 4). Then, we describe the process for the creation of CQs in Section 5. Next, we report on and discuss our evaluation results in Sections 6 and 7 respectively. Finally, we conclude in Section 8.

2. Meronymy Relations in WordNet

In WordNet, meronymy —the part-whole relation— holds between synsets like *backrest*^1_n and *seat*^1_n (i.e. parts) and *chair*^1_n (i.e. whole). Parts are inherited from their superordinates: if a chair has a seat, then an armchair has a seat as well. But parts are not inherited "upward" as they may be characteristic only of specific kinds of things rather than the class as a whole: chairs and kinds of chairs have a seat, but not all kinds of furniture have a seat.

There are 3 main meronymy relations in WordNet v3.0 that relate noun synsets: i) *part*, the general meronymy relation; ii) *member*, which relates particulars and groups; and iii) *substance*, which relates physical matters and things. In total, there are 22,187 (ordered) meronymy synset pairs (around a %10 of the synset pairs in WordNet): 9,097 pairs using *part*, 12,293 pairs using *member* and 797 pairs using *substance*. For example, the synsets *police_officer*^1_n and *police_force*^1_n are related by *member*, while *grape*^1_n and *wine*^1_n are related by *substance*.

3. SUMO

SUMO[1] has its origins in the nineties, when a group of engineers from the IEEE Standard Upper Ontology Working Group pushed for a formal ontology standard. Their goal was to develop a standard upper ontology to promote data interoperability, information search and retrieval, automated inference and natural language processing.

Currently, SUMO consists of about 20,000 terms and about 70,000 axioms organized in several levels. In the upper two levels —*Top* and *Middle* levels— the concepts, relations and axioms that are meta, generic or abstract can be found. From now on, we refer to the upper two levels of SUMO as its *core*. On the basis of these two levels, concepts that are specific to particular domains are in the so-called *domain* ontologies. Adimen-SUMO (Álvez et al., 2012) is obtained by means of a suitable transformation of the knowledge in the core of SUMO into First-Order Logic (FOL), which enables its use by FOL ATPs such as Vampire (Kovács and Voronkov, 2013) and E (Schulz, 2002).

The knowledge in SUMO is organized around the notions of *individuals* and *classes* —the main SUMO concepts. These concepts are respectively defined in Adimen-SUMO by means of the *meta*-predicates *$instance* and *$subclass*. SUMO objects and classes are not disjoint, since every SUMO class is defined to be instance of *Class*, and thus every SUMO class is also a SUMO object. Additionally, SUMO differentiates *relations* and *attributes*. In particular, SUMO distinguishes between *individual* relation and attributes —that is, instances of the SUMO classes *Relation* and *Attribute* respectively— and *classes* of relations and attributes —that is, subclasses of the SUMO classes *Relation* and *Attribute* respectively. SUMO provides specific predicates for dealing with relations and attributes. Some of the most important ones are *subrelation*, *subAttribute*, *holds*^{k} and *attribute*. For example, in the next SUMO axiom the predicate *attribute* is used for the characterization of *subAttribute*:

```
(forall (?ATTR1 ?ATTR2)                                   (1)
  (=>
    (subAttribute ?ATTR1 ?ATTR2)
    (forall (?OBJ)
      (=>
        (attribute ?OBJ ?ATTR1)
        (attribute ?OBJ ?ATTR2)))))
```

From now, on we denote the nature of SUMO concepts by adding as subscript the following symbols:

- o SUMO individuals that are neither classes nor relations nor attributes
- c SUMO classes that are neither classes of relations nor classes of attributes
- r individual SUMO relations
- a individual SUMO attributes
- R classes of SUMO relations
- A classes of SUMO attributes

For example, *Cell*$_c$ represents the *Cell* class, *member*$_r$ the *member* individual relation and *Larval*$_a$ the *Larval* attribute.

4. The Mapping Between WordNet and SUMO

WordNet is linked to SUMO by means of the mapping described in Niles and Pease (2003). This mapping connects synsets of WordNet to terms of SUMO using three relations: *equivalence*, *subsumption* and *instance*. The relation *equivalence* denotes that the related WordNet synset and the SUMO concept are equivalent in meaning, whereas the relations *subsumption* and *instance* indicate that the WordNet synset is subsumed by the SUMO concept or is an instance of the SUMO concept respectively. Furthermore, the mapping uses the complementaries of *equivalence* and *instance*. We denote mapping relations by concatenating the symbols '$=$' (*equivalence*), '$+$' (*subsumption*), '@' (*instance*), '$\widehat{=}$' (complementary of *equivalence*) and '$\widehat{+}$' (complementary of *subsumption*) to the corresponding SUMO concept. For example, the synsets *waist*^1_n and *torso*^1_n are connected to *Waist*$_o=$ and *Torso*$_c+$ respectively.

From the 82,115 noun synsets defined in WordNet v3.0, 73,472 noun synsets are directly connected to concepts that are defined in the core of SUMO —and, thus, in Adimen-SUMO—, while only 7,578 synsets are linked to SUMO concepts defined in domain ontologies. As described in Álvez et al. (2017), those synsets linked to concepts defined in domain ontologies can be connected to concepts from the core of SUMO by means of the SUMO structural relations *$subclass*, *subrelation*$_r$ and *subAttribute*$_r$. Finally, it is worth to remark that some noun synsets are connected to several SUMO concepts (concretely, 1,043 synsets).

The knowledge in the mapping between WordNet and SUMO can be formalized by means of Adimen-SUMO statements as described in Álvez et al. (2017). For this purpose, we take advantage from the fact that most of the SUMO knowledge is based on the notion of *objects* and that only a few of SUMO predicates provide information at the level of *classes*. Thus, the proposed Adimen-SUMO statements relate synsets to sets of SUMO objects in most cases. For the construction of those Adimen-SUMO statements, we choose the most suitable SUMO predicate according to nature of the SUMO concepts to which the synset is connected: *equal*$_r$ for SUMO objects, *$instance* for SUMO classes and *attribute*$_r$ for SUMO attributes.[2] In addition, we use *$subclass* for the construction of Adimen-SUMO statements in the few cases where synsets have to relate the knowledge of SUMO at the level of classes. Independently from the SUMO predicate that is selected, we introduce a new variable that is associated to the given synset in the proposed Adimen-SUMO statement. For example, the next Adimen-SUMO statements relate the synsets

[1] http://www.ontologyportal.org

[2] The mappings to SUMO relations are discarded for the moment.

Meronymy relations	Pairs		1st QP		2nd QP		3rd QP		4th QP		Total	
	Total	Error	Pairs	CQs	Pairs	CQs	Pairs	CQs	Pairs	CQs	**Pairs**	**CQs**
part	9,097	1,367	6,797	1,346	122	110	750	445	61	59	**7,730**	**1,960**
member	12,293	11,939	318	88	19	19	15	12	2	2	**354**	**121**
substance	797	632	147	47	6	6	10	9	2	2	**165**	**64**
Total	**22,187**	**13,938**	**7,262**	**1,481**	**147**	**135**	**775**	**466**	**65**	**63**	**8,249**	**2,145**

Table 1: CQs obtained from WordNet meronymy

$waist^1_n$, $atom^1_n$ and $police_officer^1_n$—respectively connected to $Waist_o$, $Atom_c$ and $PoliceOfficer_a$— to sets of SUMO objects by introducing the new variables *?W*, *?A* and *?O*:

(equal ?W Waist) (2)

($instance ?A Atom) (3)

(attribute ?O PoliceOfficer) (4)

Similarly, the next Adimen-SUMO statement relates the synsets $cartilage^1_n$, which is connected to $Tissue_c$, to a set of SUMO classes introducing the new variable *?T*:

($subclass ?T Tissue) (5)

Finally, the quantification of the introduced variables is decided according to the mapping relation that connects synsets and the SUMO concepts:

- If the given synset is connected using *equivalence* (resp. the complement of *equivalence*), then we can assume that the synset is related to all (resp. is not related to any of) the potential SUMO objects that are characterized by the Adimen-SUMO statement proposed above. For this purpose, the variable associated to the given synset is considered to be universally quantified.
- Otherwise —the synset is connected using *subsumption* (resp. the complement of *subsumption*) or *instance*—, we can only assume that the synset is related to (resp. is not related to) some of the potential SUMO objects that are characterized by the Adimen-SUMO statement proposed above. This means that the variable associated to the given synset is considered to be existentially quantified.

In the next section, we describe the use of question patterns for the combination of the Adimen-SUMO statements that are obtained for the synsets in a given WordNet pair.

5. Competency Questions Based on Meronymy

The automatic validation of WordNet and SUMO on the basis of CQs and ATPs requires to translate all the information into a formal language. By means of Adimen-SUMO, the core information of SUMO is already written in FOL. In addition, the mapping information between WordNet and SUMO can be translated into Adimen-SUMO statements as we describe in the above section. Thus, it suffices to translate the semantics of the WordNet meronymy relations in terms of Adimen-SUMO. For this purpose, we have inspected SUMO in order to find the relations that are synonym or semantically similar to them. In SUMO, the main meronymy relation is $part_r$ and we can find 30 different subrelations of $part_r$ in its core. Among them, we have selected the SUMO predicates $part_r$ and $member_r$ as counterpart of the WordNet relations *part* and *member* respectively. In addition, we have selected $material_r$, which is not subrelation of $part_r$, as counterpart of *substance*. As for every SUMO relation, SUMO provides *domain* axioms that restrict the set of SUMO objects that can be related by the above predicates as follows:

- $part_r$ relates pairs of $Object_c$ individuals.
- $member_r$ relates $SelfConnectedObject_c$ objects (first argument) to $Collection_c$ objects (second argument).
- $material_r$ relates subclasses of $Substance_c$ (first argument) to $CorpuscularObject_c$ objects (second argument).

Additionally, SUMO provides incompatibilities between SUMO objects. Among others, objects of $CorpuscularObject_c$ are incompatible with both $Collection_c$ and $Substance_c$.

On the basis of the above formalization of the meronymy information of WordNet in terms of Adimen-SUMO, we proceed to the creation of CQs. For this purpose, we propose 4 QPs depending on the mapping relation by which the synsets in the given WordNet pair are connected to SUMO. Those QPs are instantiated by using (a) the Adimen-SUMO statements that formalize the mapping information of synsets, and (b) the SUMO predicate that is selected depending on the given WordNet meronymy relation. During the process of instantiation, we can already detect some incompatibilities on the basis of individual SUMO incompatibilities.

We report on the number of WordNet pairs tested and the number of CQs resulting from each QP in Table 1 and in the next subsections we describe the proposed QPs.

```
(exists (?X, ?Y)
  (and
    < s_part, ?X >
    < s_whole, ?Y >
    (< SUMO predicate > ?X ?Y)))
```

Figure 1: First question pattern for $\langle s_part, s_whole \rangle$ meronymy pairs

5.1. First Question Pattern

The first question pattern is designed to be applied to meronymy pairs where both synsets are connected using (the negation of) *subsumption* and *instance*.
In Figure 1, we describe the combination of the selected SUMO predicate and the Adimen-SUMO statements that result from the mapping information of synsets by considering the introduced variables to be existentially quantified.

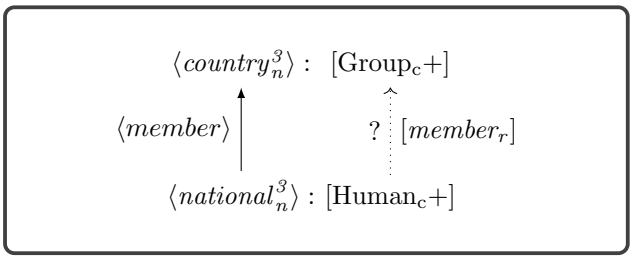


Figure 2: *national*$_n^1$ and *country*$_n^3$.

Next, we illustrate the instantiation of the resulting question pattern by considering the WordNet pair *member*(*national*$_n^1$,*country*$_n^3$). As described in Figure 2, the synsets in that pair are respectively connected to *Human*$_c$+ and *Group*$_c$+. Thus, the combination of the SUMO statements that result from their mapping information with the SUMO predicate *member*$_r$ yields the following CQ:

```
(exists (?X, ?Y)                                    (6)
   (and
      ($instance ?X Human)
      ($instance ?Y Group)
      (member ?X ?Y)))
```

In the same way, the synsets in the WordNet pair *substance*(*cartilage*$_n^1$,*cartilaginous_structure*$_n^1$) are respectively connected to *Tissue*$_c$ and *BodyPart*$_c$. In this case, we have to relate *cartilaginous_structure*$_n^1$ with a set of SUMO classes, since the selected SUMO predicate is *material*$_r$. Thus, the resulting CQ is:

```
(exists (?X, ?Y)                                    (7)
   (and
      ($subclass ?X Tissue)
      ($instance ?Y BodyPart)
      (material ?X ?Y)))
```

Using this first QP, we obtain 1,481 CQs from 7,262 WordNet pairs (see Table 1).

5.2. Second Question Pattern

The second question pattern is designed for the meronymy synset pairs $\langle s_part, s_whole \rangle$ where s_part is connected by (the negation of) *equivalence* and s_whole is connected by (the negation of) *subsumption* or *instance*.
In this case, the variable associated to s_whole is considered to be universally quantified, while the variable associated to s_part is considered to be existentially quantified. The resulting question pattern is described in Figure 3.

```
(forall (?X)
   (=>
      < s_part, ?X >
      (exists (?Y)
         (and
            < s_whole, ?Y >
            (< SUMO predicate > ?X ?Y)))))
```

Figure 3: Second question pattern for $\langle s_part, s_whole \rangle$ meronymy pairs

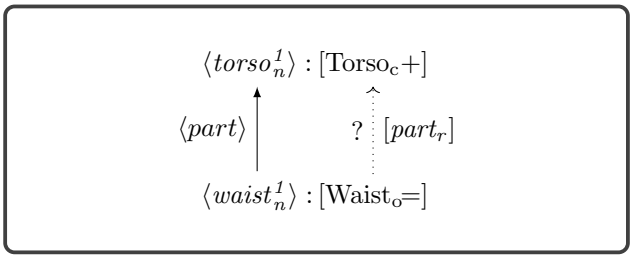


Figure 4: *waist*$_n^1$ and *torso*$_n^1$.

In order to illustrate the instantiation of this second question pattern, we consider the synset pair *part*(*waist*$_n^1$,*torso*$_n^1$), where the involved synsets are respectively connected to *Waist*$_o$= and *Torso*$_c$+ as described in Figure 4. On the basis of the above mapping information, the selected SUMO predicate is *part*$_r$ and we obtain the following CQ:

```
(forall (?X)                                        (8)
   (=>
      ($instance ?X Waist)
      (exists (?Y)
         (and
            ($instance ?Y Torso)
            (part ?X ?Y)))))
```

As reported in Table 1, from this QP we obtain 135 CQ on the basis of 147 WordNet pairs.

```
(forall (?Y)
   (=>
      < s_whole, ?Y >
      (exists (?X)
         (and
            < s_part, ?X >
            (< SUMO predicate > ?X ?Y)))))
```

Figure 5: Third question pattern for $\langle s_part, s_whole \rangle$ meronymy pairs

5.3. Third Question Pattern

The third question pattern is the dual of the second one because it is designed for meronymy synset pairs $\langle s_part, s_whole \rangle$ where s_part is connected by (the negation of) *subsumption* or *instance*, and s_whole is connected by (the negation of) *equivalence*. Consequently, the variables associated to s_whole and s_part are considered to be universally and existentially quantified respectively, as described in Figure 5.

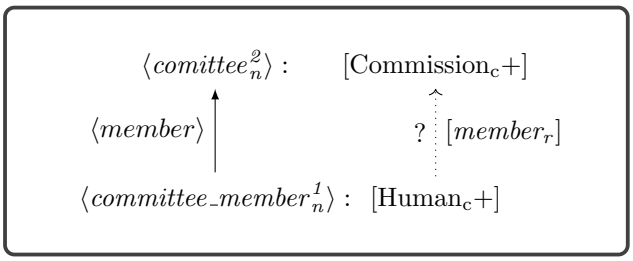


Figure 6: *committee_member*^1_n and *committee*^1_n.

This third question pattern is applied to synset pairs like *member*(*committee*^1_n,*committee_member*^1_n), where synsets are respectively connected to *Human*$_c$+ and *Commission*$_c$=. By using the SUMO predicate *member*$_r$, the resulting CQ is:

```
(forall (?Y)                                        (9)
  (=>
    ($instance ?Y Commission)
    (exists (?X)
      (and
        ($instance ?X Human)
        (member ?X ?Y)))))
```

Using this third QP, we obtain 466 CQs from 775 WordNet pairs (see Table 1).

5.4. Fourth Question Pattern

The last question pattern is designed for its application to meronymy pairs where both synsets are connected using (the negation of) *equivalence*.

In this case, the question pattern is obtained by the conjunction of the second and the third question patterns (see Figure 7). In order to illustrate its application, we consider the synset pair *part*(*elementary_particle*^1_n,*atom*^1_n), where synsets are respectively connected to *SubatomicParticle*$_c$=

```
(and
  (forall (?X)
    (=>
      < s_part, ?X >
      (exists (?Y)
        (and
          < s_whole, ?Y >
          (< SUMO predicate > ?X ?Y)))))
  (forall (?Y)
    (=>
      < s_whole, ?Y >
      (exists (?X)
        (and
          < s_part, ?X >
          (< SUMO predicate > ?X ?Y))))))
```

Figure 7: Fourth question pattern for $\langle s_part, s_whole \rangle$ meronymy pairs

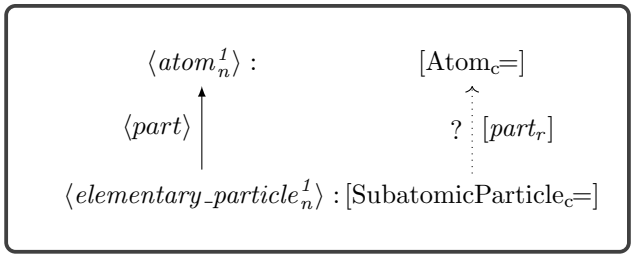


Figure 8: *elementary_particle*^1_n and *atom*^1_n.

and *Atom*$_c$= as described in Figure 8. The resulting CQ is:

```
(and                                                (10)
  (forall (?X)
    (=>
      ($instance ?X SubatomicParticle)
      (exists (?Y)
        (and
          ($instance ?Y Atom)
          (part ?X ?Y)))))
  (forall (?Y)
    (=>
      ($instance ?Y Atom)
      (exists (?X)
        (and
          ($instance ?X SubatomicParticle)
          (part ?X ?Y))))))
```

From this last QP, we obtain 63 CQs on the basis of 65 WordNet pairs (see Table 1).

Meronymy relations	QPs								Total			
	QP #1		QP #2		QP #3		QP #4					
	Pairs	CQs	Pairs	CQs	Pairs	CQs	Pairs	CQs	**Pairs**	**CQs**	**Pairs**	**CQs**
part	+3,272	599	+122	56	+288	162	+8	8	**+3,690**	**825**	**47.74%**	**42.09%**
	-21	6	-0	0	-1	1	-5	5	**-27**	**12**	**0.35%**	**0.61%**
member	+20	10	+1	1	+1	1	+0	0	**+22**	**12**	**6.21%**	**9.92%**
	-24	9	-0	0	-0	0	-0	0	**-24**	**9**	**6.78%**	**7.44%**
substance	+94	17	+1	1	+3	2	+0	0	**+98**	**20**	**59.39%**	**31.25%**
	-0	0	-2	2	-0	0	-0	0	**-2**	**2**	**1.21%**	**3.13%**
Total	**+3,386**	**626**	**+124**	**58**	**+292**	**165**	**+8**	**8**	+3,810	857	**46.19%**	**39.95%**
	-45	**15**	**-2**	**2**	**-1**	**1**	**-5**	**5**	-53	23	**0.64%**	**1.07%**

Table 2: Evaluation results

6. Evaluation

In this section, we discuss the results obtained from the experimental evaluation of the set of CQs described in the above section. This experimentation has been performed in an Intel® Xeon® CPU E5-2640v3@2.60GHz with 2GB of RAM memory per processor and different. All the required knowledge resources—the ontology Adimen-SUMO v2.6,(Álvez et al., 2017) the set of CQs and conjectures, the mapping between Adimen-SUMO v2.6 and WordNetv3.0, WordNet v3.0 meronymy pairs— and the resulting execution reports are publicly available.[3]

For the evaluation of CQs, we have applied the framework proposed in Álvez et al. (2017). By following this proposal, we get two conjectures for each CQ: i) the conjecture that describes the CQ itself, which is expected to be entailed by the ontology (called *truth-test*), and ii) its negation, which is expected not to be entailed (called *falsity-test*). Each of the resulting 4,830 conjectures has been tested using several versions of Vampire[4] (Kovács and Voronkov, 2013) and E[5] (Schulz, 2002). In each test, we provide the conjecture together with Adimen-SUMO as input to the ATP system.

If a truth-test is proved to be entailed by the ontology, then we decide that the knowledge in the WordNet pairs that yield the corresponding CQ is *compatible* with SUMO and the mapping information. Thus, we conclude that those WordNet pairs are *well-aligned* —or simply *aligned*— with SUMO and the mapping information. On the contrary, if a falsity-test is proved to be entailed by the ontology, then the knowledge in the WordNet pairs that yield the corresponding CQ is *incompatible* with SUMO and the mapping information. Consequently, we say that those WordNet pairs are *misaligned* with SUMO and the mapping information. If either the truth-test or the falsity-test of a given CQ is proved to be entailed by the ontology, then we say than the corresponding CQ is *resolved*.

In Table 2 we summarize our experimental results as follows. For each meronymy relation (1st column) and each QP, we provide the following information from the 2nd to the 9th column:

- In Pairs columns, prefixed by '+' (resp. by '-') we give the number of WordNet pairs that are well-aligned (resp. misaligned) with SUMO and the mapping information.
- In CQs columns, we give the number of truth-tests/falsity-tests that have been proved for the classification of WordNet pairs as aligned/misaligned with SUMO and the mapping information.

In addition, in the last 4 columns we summarize the number/percentage of WordNet pairs that are well-aligned/misaligned with SUMO and the mapping information, and the number and percentage of truth-tests/falsity-tests that have been proved.

To sum up, the knowledge in 3,810 WordNet pairs (46.19 % of the tested WordNet pairs) is decided to be compatible with SUMO and the mapping information, while the knowledge in only 53 WordNet pairs (0.64 % of the tested pairs) is decided to be incompatible. Among them, it is easy to see that *part* and *substance* pairs are better aligned with the knowledge in SUMO than *member* pairs: on one hand, 3,690 *part* pairs (47.74 % of the tested *part* pairs) and 98 *substance* pairs (59.39 % of the tested *substance* pairs) are decided to be well-aligned with SUMO and the mapping information against 22 *member* pairs (6.21 % of the tested *member* pairs); on the other hand, 24 *member* pairs (6.78 % of the tested *member* pairs) are decided to be misaligned with SUMO and the mapping information, against 27 *part* pairs (0.35 % of the tested *part* pairs) and 2 *substance* pairs (0.64 % of the tested *substance* pairs). Further, if we consider the total number of meronymy pairs, the percentage of *part* pairs that are well-aligned with SUMO and the mapping information —40.56 % (3,690 from 9,097 *part* pairs)— is clearly larger than the percentage of *member* and *substance* pairs that are well-aligned —0.18 % (22 from 12,293 *member* pairs) and 12.3 % (98 from 797 *substance* pairs respectively). Finally, the percentage of *member* pairs that are tested —only 2.88 % (354 from 12,293 *member* pairs)— is the lowest one and, in addition, the number of *member* pairs that are decided to be misaligned with SUMO and the mapping information is larger than the number of *member* pairs that are decided to be well-aligned —24 against 22. These two facts, but especially the first

[3] `http://adimen.si.ehu.es/web/AdimenSUMO`

[4] Using the following parameters: `--proof tptp --output_axiom_names on --mode casc -t 600 -m 2048.`

[5] Using the following parameters: `--auto --proof-object -s --cpu-limit=600 --memory-limit=2048.`

one, lead us to conclude that the mapping information of the involved synsets can be substantially improved.

With respect to CQs, ATPs are able to successfully resolve 880 CQs (41.03 % of 2,145 CQs): 857 truth-tests (39.95 % of 2,145 truth-tests) plus 23 falsity-tests (1.07 % of 2,145 falsity-tests). It is worth noting that the percentage of meronymy-based CQs that are resolved by ATPs is quite similar to the percentage of CQs for the validation of the knowledge in the ontology that were resolved by ATPs in the experimentation reported in Álvez et al. (2017). Regarding QPs, the percentage of CQs obtained from the the first 3 QPs that are resolved —43.28 % (641 from 1,481 CQs), 44.44 % (60 from 158 CQs) and 35.62 % (166 from 466 CQs) respectively— is clearly larger than the percentage of CQs obtained from the last QP that are resolved — 20.63 % (13 from 63 CQs). This fact is not surprising since the truth-tests that result from the first 3 QPs are much weaker than the truth-tests that are obtained from the last QP.

7. Discussion

In this section, we proceed to discuss some of the particular results that we have extracted from our experiments. First of all, our proposal enables the detection of semantics agreements between WordNet, SUMO and their mapping. For example, the synset pairs *member*(*national*^1_n,*country*^3_n), *substance*(*cartilage*^1_n,*cartilaginous_structure*^3_n) and *part*(*elementary_particle*^1_n,*atom*^1_n) are decided to be well-aligned with SUMO and the mapping information since the truth-tests obtained from CQs (6,7,10) are proved to be entailed by Adimen-SUMO v2.6. Similarly, the synsets in the pair *member*(*police_officer*^1_n,*police_force*^1_n) are respectively connected to *PoliceOfficer*$_a=$ and *PoliceOrganization*$_c+$. Thus, by applying the second QP we obtain the following CQ:

```
(forall (?Y)                                              (11)
   (=>
      (attribute ?Y PoliceOfficer)
      (exists (?X)
         (and
            ($instance ?X PoliceOrganization)
            (member ?X ?Y)))))
```

The above CQ is resolved by ATPS since its truth-test is proved to be entailed by Adimen-SUMO v2.6. Consequently, the pair *member*(*police_officer*^1_n,*police_force*^1_n) is decided to be well-aligned with SUMO and the mapping information.

Second, our proposal enables the detection of inconsistencies among WordNet, SUMO and their mapping. Indeed, we do not always require the help of ATPs, since some inconsistencies can be discovered during the process of instantiation of QPs on the basis of incompatibilities among SUMO objects. For instance, the synsets *grape*^1_n and *wine*^1_n are related by *substance* and respectively connected *FruitOrVegetable*$_c+$ and *Wine*$_c=$. According to our interpretation of the semantics of *substance* and the mapping information, we have to use the third QP and the SUMO predicate *material*$_r$ in order to translate the knowledge in *substance*(*grape*^1_n,*wine*^1_n) in terms of Adimen-SUMO. However, *FruitOrVegetable*$_c$ is defined to be subclass of *CorpuscularObject*$_c$ in SUMO. Therefore, *FruitOrVegetable*$_c$ is incompatible with *Substance*$_c$, which prevents the use of *material*$_r$ for the instantiation of the third QP. This fact leads us to discover that the pair *substance*(*grape*^1_n,*wine*^1_n) should be better represented by *substance*(*grape_juice*^1_n,*wine*^1_n), where *grape_juice*^1_n is connected to *Substance*$_c+$. This also would require a new relation between *grape*^1_n and *grape_juice*^1_n. Additional examples can be taken from many synset pairs related by *member*, where both synsets are connected to the same SUMO concept although the first one denotes an individual organism and the second one the species, genus or family to which the organism belongs. For instance, *bear*^1_n and *ursidae*^1_n are both connected to *Mammal*$_c+$, which reveals the existence of an incompatibility according to our interpretation of the meronymy relations in WordNet and the mapping information. Anyway, inconsistencies are also detected by means of the use of ATPs. For example, the synsets in *part*(*cell*^2_n,*cell_nucleus*^1_n) are connected to *Cell*$_c=$ and *CellNucleus*$_c=$ respectively. Hence, we obtain the following QP by the instantiation of the fourth QP:

```
(and                                                      (12)
   (forall (?X)
      (=>
         ($instance ?X CellNucleus)
         (exists (?Y)
            (and
               ($instance ?Y Cell)
               (part ?X ?Y)))))
   (forall (?Y)
      (=>
         ($instance ?Y Cell)
         (exists (?X)
            (and
               ($instance ?X CellNucleus)
               (part ?X ?Y))))))
```

ATPs are able to prove the falsity-test that results from the above CQ, which enables the detection of an inconsistency according to our interpretation of the knowledge in WordNet and the mapping information. Concretely, the falsity-test that results from our interpretation is incompatible with the fact that some cells lack a nucleus, as stated by the following SUMO axiom:

```
(=>                                                       (13)
   ($instance ?C RedBloodCell)
   (not
      (exists (?N)
         (and
            ($instance ?N CellNucleus)
            (part ?N ?C)))))
```

Similarly, the knowledge in the pair *member*(*malacosoma_americana*^1_n,*genus_malacosoma*^1_n) is detected to be incompatible with SUMO and the mapping information. As the involved synsets are respectively connected to *Insect*$_c$+ and *Larval*$_a$+, we obtain the following CQ by the instantiation of the first QP:

```
(exists (?X, ?Y)                                    (14)
   (and
      ($instance ?X Insect)
      (attribute ?Y Larval)
      (member ?X ?Y)))
```

By inspecting the proof of the falsity-tests that results from the above CQ, we discover that the problem is due to the mapping of *genus_malacosoma*^1_n to *Larval*$_a$+, since the second argument of *member*$_r$ is restricted to be instance of *Collection*$_c$ and *Larval*$_a$+ cannot be applied to instances of *Collection*$_c$.

Finally, our proposal also enables the detection of missing knowledge. For example, ATPs cannot prove the truth- and falsity-tests inherited from (8) because *Waist*$_o$ and *Torso*$_c$ are not properly related in SUMO, as discussed in Section 1. In the same fashion, we discover that SUMO lacks the appropriate knowledge relating the concepts of *Human*$_c$ and *Commission*$_c$ by *member*$_r$ since the truth- and falsity-tests inherited from (9) cannot be proved by ATPs.

8. Conclusions and Future Work

By analysing our experimentation results, we can conclude that our proposal enables a sophisticated cross-checking of the information in WordNet, SUMO and their mapping. In particular, by means of practical examples, we have illustrated that the proposed system enables (a) the validation of some pieces of knowledge and (b) the detection of inconsistencies and missing knowledge. Further, our results also demonstrate the suitability of the involved resources for their application to practical tasks related to natural language processing. In future work, we plan to correct some of the issues detected with the mapping information and to augment the knowledge in SUMO in order to increase the level of alignment among WordNet, SUMO and their mapping.

9. Acknowledgements

This work has been partially funded by the Spanish Projects TUNER (TIN2015-65308-C5-1-R) and GRAMM (TIN2017-86727-C2-2-R), the Basque Project LoRea (GIU15/30) and the UPV/EHU project OEBU (EHUA16/33).

10. Bibliographical References

Álvez, J. and Rigau, G. (2018). Towards cross-checking WordNet and SUMO using meronymy. In P. Vossen, et al., editors, *Proceedings of the 9th Global WordNet Conference (GWC 2018)*.

Álvez, J., Atserias, J., Carrera, J., Climent, S., Laparra, E., Oliver, A., and Rigau, G. (2008). Complete and consistent annotation of WordNet using the Top Concept Ontology. In Nicoletta Calzolari, et al., editors, *Proc. of the 6th Int. Conf. on Language Resources and Evaluation (LREC 2008)*, pages 1529–1534. European Language Resources Association (ELRA), may.

Álvez, J., Lucio, P., and Rigau, G. (2012). Adimen-SUMO: Reengineering an ontology for first-order reasoning. *Int. J. Semantic Web Inf. Syst.*, 8(4):80–116.

Álvez, J., Lucio, P., and Rigau, G. (2017). Black-box testing of first-order logic ontologies using WordNet. *CoRR*, abs/1705.10217.

Daudé, J., Padró, L., and Rigau, G. (2003). Making WordNet mapping robust. *Procesamiento del lenguaje natural*, 31.

C. Fellbaum, editor. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Grüninger, M. and Fox, M. S. (1995). Methodology for the design and evaluation of ontologies. In *Proc. of the Workshop on Basic Ontological Issues in Knowledge Sharing (IJCAI 1995)*.

Horrocks, I. and Voronkov, A. (2006). Reasoning support for expressive ontology languages using a theorem prover. In J. Dix et al., editor, *Foundations of Information and Knowledge Systems*, LNCS 3861, pages 201–218. Springer.

Kovács, L. and Voronkov, A. (2013). First-order theorem proving and Vampire. In N. Sharygina et al., editors, *Computer Aided Verification*, LNCS 8044, pages 1–35. Springer.

Niles, I. and Pease, A. (2001). Towards a standard upper ontology. In Guarino N. et al., editor, *Proc. of the 2nd Int. Conf. on Formal Ontology in Information Systems (FOIS 2001)*, pages 2–9. ACM.

Niles, I. and Pease, A. (2003). Linking lexicons and ontologies: Mapping WordNet to the Suggested Upper Merged Ontology. In H. R. Arabnia, editor, *Proc. of the IEEE Int. Conf. on Inf. and Knowledge Engin. (IKE 2003)*, volume 2, pages 412–416. CSREA Press.

Richens, T. (2008). Anomalies in the WordNet verb hierarchy. In *Proc. of the 22nd Int. Conf. on Computational Linguistics-Volume 1*, pages 729–736. Association for Computational Linguistics.

Rodríguez, H., Climent, S., Vossen, P., Bloksma, L., Peters, W., Alonge, A., Bertagna, F., and Roventini, A. (1998). The top-down strategy for building EuroWordNet: Vocabulary coverage, base concepts and top ontology. In *EuroWordNet: A multilingual database with lexical semantic networks*, pages 45–80. Springer.

Schulz, S. (2002). E - A brainiac theorem prover. *AI Communications*, 15(2-3):111–126.

11. Language Resource References

J. Álvez and P. Lucio and G. Rigau. (2017). *Adimen-SUMO*. 2.6.

Extended HowNet 2.0 – An Entity-Relation Common-Sense Representation Model

Wei-Yun Ma, Yueh-Yin Shih
Institute of Information Science, Academia Sinica
128 Academia Road, Section 2, Nankang, Taipei, Taiwan 115
{ma, yuehyin}@iis.sinica.edu.tw

Abstract

In this paper, we propose Extended HowNet 2.0 – an entity-relation common-sense representation model. Comparing to HowNet and Extended HowNet, E-HowNet 2.0 has the following improvements: (a) Reorganizing the hierarchical structure of primitives and basic concepts; (b) Rich lexical information: In addition to sense definition, each entry of lexical sense may also include operational expressions as well as semantic functions which facilitate future semantic composition processes. (c) Improvement of sense definitions and sense definitions for basic concepts. (d) Developing a new automatic ontology reconstruction system. (e) Developing a query system called E-HowNet Relation Database for flexibly clustering concepts.We hope Extended HowNet 2.0 can bring significant benefits to the community of lexical semantics and natural language understanding.

Keywords: E-HowNet, entity-relation, lexical semantics, ontology

1. Introduction

The purpose of designing the lexical semantic representation model E-HowNet is for natural language understanding. E-HowNet, an evolution and extension of HowNet (Dong and Dong, 2006), is a frame-based entity-relation representation model to define lexical senses and to achieve compositional semantics.

The current E-HowNet 2.0 [1] shows the following improvements: (a) Reorganizing the hierarchical structure of primitives and basic concepts; (b) Rich lexical information: In addition to sense definition, each entry of lexical sense may also include operational expressions as well as semantic functions which facilitate future semantic composition processes. Event frames are also provided. (c) Improvement of sense definitions and sense definitions for basic concepts. (d) Developing a new automatic ontology reconstruction system: In case of revisions of lexical sense expressions or nodes of conceptual hierarchy, the ontology reconstruction system may re-attach each lexical entry to appropriated ontological nodes and results a new ontology. (e) Developing a query system called E-HowNet Relation Database for flexibly clustering concepts.

The remainder of this paper is organized as follows. We describe the background of developing E-HowNet in Section 2 and elaborate the feature improvements of the current E-HowNet version 2.0 in Section 3. The online systems based on E-HowNet are introduced in Section 4. Section 5 is conclusions and future work.

2. Background

HowNet is an on-line common-sense knowledge base unveiling the inter-conceptual relations and inter-attribute relations of concepts conveyed by Chinese words and their English equivalents (Dong & Dong, 2006). Compared with WordNet, HowNet's architecture provides richer information apart from hyponymy relations. It also enriches relational links between words via encoded feature relations. The advantages of HowNet are (a) inherent properties of concepts are derived from encoded feature relations in addition to hypernymous concepts, and (b) information regarding conceptual differences between different concepts and information regarding morpho-semantic structure are encoded. HowNet's advantages make it an effective electronic dictionary for NLP applications. In recent years, HowNet has been applied to the researches of word similarity calculation (Liu & Li, 2002), machine translation (Dong 1999), and Information Retrieval (Dorr, Levow and Lin, 2000) etc.

When we say that a sentence is 'understood', we mean that the concepts and the conceptual relationships expressed by the sentence are unambiguously identified, and we can make correct inferences and/or responses. Therefore, to achieve natural language understanding, computer systems should know the sense similarity and dissimilarity of words and sentences. A representational framework which represents knowledge about lexical concepts and performs the following functions is needed. (a) Identifies synonymous concepts and measures similarity distance between two concepts (Liu and Li, 2002). (b) Knows the shared semantic features and feature differences between two concepts. (c) Provides unique indices to each concept, such that associated knowledge can be coded and accessed. (d) Language independent sense encoding. (e) Logical inferences through conceptual property inheritance system. (f) Dynamic concept decomposition and composition mechanisms. None of the currently available ontology provides all of the above functions and so far there has been little research on applying HowNet to semantic composition. We therefore extend HowNet to deal with this problem. The resulting system is called E-HowNet.

The development of E-Hownet started in 2003 (Chen et al., 2005). We adopt the set of primitives and taxonomy of HowNet and adjusted to suit the goal of semantic composition. The major extension features are: (a) Word senses are defined by not only primitives but also any well-defined basic concepts and conceptual relations; (b) Semantic relations are explicitly expressed; (c) A Uniform representation for content words, function words, as well

[1] http://ehownet.iis.sinica.edu.tw/index.php

as phrases; (d) The capacities of semantic composition and decomposition; (e) Near canonical representations for lexical senses and phrasal senses. The above characteristics of the E-HowNet which make the E-HowNet different from other ontologies.

Rather than creating a completely new ontology, E-HowNet links different ontologies. For instance, we established the links between HowNet sememes and WordNet synsets. Thus WordNet synsets (version 1.6) are used as an alternative intermediate representational language. In order to achieve unambiguous and language-independent definitions, E-HowNet adopts WordNet synsets as an alternative vocabulary for conceptual indexing and representation. As a conceptual representation that may use WordNet synsets as its description language, E-HowNet is universal and language-independent.



Figure 1: Top-Level of E-HowNet ontology

3. Improvements of Extended-HowNet

As mentioned above, the E-HowNet ontology is a reconstruction of the HowNet ontology. The major revision was to include the hierarchy for relations to enable semantic composition and decomposition (Chen et al., 2004). Therefore, the E-HowNet ontology is formed by entity taxonomy and relation taxonomy. Each word sense is a node of the taxonomy and expressed by an E-HowNet expression. Synonyms or near synonyms should be expressed by the same expression. The top levels of E-HowNet ontology is shown in Figure 1 and a complete taxonomy can be found in http://ckip.iis.sinica.edu.tw/taxonomy/.

3.1 Reorganizing the Hierarchical Structure of Primitives and Basic Concepts

In E-HowNet 2.0 all concepts are either primitive concepts or defined by simpler concepts (either primitive concepts or basic concepts) in terms of an entity-relation model (Chen et al., 2004; Chen K.J. et al., 2005; Chen Y.J.et al., 2005; Huang, Chung and Chen, 2008). A primitive concept will have an English equivalent beside it, e.g. {read|讀}, whereas a basic concept will be expressed by a Chinese word and its English translation pair which is further defined by primitive concepts, e.g. {狗|dog} defined as {livestock|牲畜:telic={TakeCare|照料:patient={family|家庭},agent={~}}}.

E-HowNet ontology is formed by all lexical senses as well as primitive and basic concepts in a hierarchical order. Any concept inherits all the fundamental features of its hypernym and must have at least one feature that its hypernym does not own. The improvement of E-HowNet 2.0 will be elaborated in the following subsections.

3.1.1 Multi-level Sense Representation by Primitives and Basic Concepts

Conventional sense representation have used semantic primitives to define and achive canonical representation for concepts (Wierzbicka, 1972), such as Conceptual Dependency representation (Schank, 1975) and HowNet. However, using primitives only to define concepts causes information degrading as it is almost impossible to understand a definition of a complex concept merely with primitives. Furthermore, it is debatable whether there exists a limited and fixed set of so-called primitives. Therefore, we adopt 2,233 primitives from HowNet and extend 2698 basic concepts which make a deeper hierarchical structure and more precise semantic branching. It also results that lexical senses expressed based on basic concepts become more precise and readable. For example, both *dog* 狗 and *Beijing dog* 北京狗 are defined as def: {livestock|牲畜} in HowNet and the hypernym-hyponym relation of these two concepts is missing.



Figure 2: Example of a hieratical structure including primitives and basic concepts

In E-HowNet 2.0, {狗|dog} is a basic concept under {livestock| 牲畜} and defined as {livestock| 牲畜 :telic={TakeCare| 照料 :patient={family| 家庭},agent={~}}}. Thus, the top-level definition of *Beijing dog* 北京狗 is as (1a) and can be further extended into the ground level definition as (1b). Such a multi-level representational framework makes sense definitions more precise. It also retains the advantage of using semantic primitives to achieve canonical sense representation. The hierarchical structure of {狗|dog}, {livestock|牲畜} and other related concepts is shown in Figure 2.

(1a) def :{狗|dog:source={北京|Beijing}}
(1b)def :{livestock|牲畜:
telic={TakeCare|照料:
patient={family|家庭},
agent={~}}},
source={北京|Beijing}}}

3.1.2 Hierarchy Structures for both Entities and Relations

We also adjust the ontology structure into two parts. The first part is hierarchy for entities and the second part is hierarchy for relations, i.e. semantic roles. The entity subtree is formed by event subtree and object subtree. The relations include attribute and function.

Entities indicate concepts that have substantial content. By contrast, relations play the role of linking semantic relations between entities. (Chen et al., 2004; Chen K.J. et al., 2005; Chen Y.J. et al., 2005; Huang, Chung and Chen, 2008). Semantic roles also form a hierarchical structure from coarse-grained semantic roles to fine-grained semantic roles. There are 409 relations in E-HowNet 2.0.

Function is a special kind of relation, i.e. a one-to-one relation, in which a concept is mapped onto another concept of the same domain. Rather than establishing the thematic relation or property attribute between two parameters, functions transform a concept to a new concept. Function has compositional property. New functions can be constructed by combining functions of the same domain. For instances, the kinship function of {YoungerBrother({father({x})} denotes 'younger brother of x's father (叔父)' and the direction function of {north({east({place|地方})})} denotes 'the direction of north-east (東北方)' Both are compositions of basic functions.

3.1.3 Uniform Representation for content words and function words

The sense of a natural-language sentence is the result of the composition of the senses of constituents/words and their relations. Conventional linguistic theories classify words into content words and function words. Content words denote entities and function words mainly mark grammatical functions. Actually, there is no clear-cut distinction between the two classes. Therefore, by adding the hierarchy for relations, E-HowNet provides a uniform representation for both function words and content words and enable the capabilities of semantic composition and decomposition. An example is given below to demonstrate the semantic composition process under the framework of E-HowNet.

(2) Because of raining, clothes are all wet. 因為下雨，衣服都濕了

In the above sentence, '*wet* 濕', '*clothes* 衣服' and '*rain* 下雨' are content words while '*all* 都', '*le* 了' and '*because* 因為' are function words. Their E-HowNet sense representations are shown in Table 1. The difference of their representation is that function words start with a relation but content words have under-specified relations.

Word	POS	Definition
because 因為	Cb(conjunction)	cause ={ }
rain 下雨	VA(intransitive verb)	{rain|下雨}
clothes 衣服	Na(common noun)	{clothing|衣物}
all 都	Da(adverb)	quantity={complete|整}
wet 濕	VH(state verb))	{wet|濕}
le 了	Ta(particle)	aspect={Vachieve|達成}

Table1: Sense definitions for each constitute in sentence(2)

If a content word plays a dependency daughter of a head concept, the relation between the head concept and this content word will be established after parsing process. Suppose that the following dependency structure are derived after parsing the sentence (2).

(3) S(reason:VP(Head:Cb: 因為 |dummy:VA: 下雨)|theme:NP(Head:Na:衣服) | quantity: Da:都 | Head:Vh: 濕|particle:Ta:了)。

After unification process, the following semantic composition result (4) is derived. The representations of dependency daughters became the feature attributes of the sentential head 'wet|濕'.

(4) def:{wet|濕: theme={clothing|衣物},
aspect={Vachieve|達成},
quantity={complete|整},
cause={rain|下雨}}.

In (4), function word '*because* 因為' links the relation of 'cause' between head concept '*wet* 濕' and '*rain* 下雨'. The result of composition is expressed as cause(wet|濕)={rain|下雨}.For the sake of notational convenience, the head argument of a relation is omitted. Therefore cause(wet|濕)={rain|下雨} is expressed as cause={rain|下雨}; theme(wet|濕)={clothing|衣物} is expressed as theme={clothing|衣物} and so on.

3.1.4 Correspondence between Attribute Types and Value Types

Some attributes may have specific range of values. For instance, values of color are red, blue, and yellow etc. In E-HowNet, attributes and their respective values are constructed in parallel. Such information is very useful in identifying semantic relations between two constituents while doing semantic composition.

3.2 Rich Lexical Information for Automatic Semantic Composition

A lexical word may play different syntactic and semantic functions and ambiguously denote many lexical concepts. Therefore, in E-HowNet, each lexical concept of a word is identified and provided with its sense definition, English translation, part-of-speech. To facilitate automatic semantic composition and language understanding, E-HowNet 2.0 ontology provides additional lexical information other than conceptual definitions and part-of-speeches. Operational expression, event frames, semantic functions etc. are provided for lexical entries to facilitate semantic composition processing.

3.2.1 Operational Expression

A lexical word may play different syntactic and semantic functions and ambiguously denote many lexical concepts. Therefore, in E-HowNet, each lexical concept of a word is identified and provided with its sense definition, English translation, part-of-speech, and major semantic functions. To facilitate automatic semantic composition and language understanding, E-HowNet ontology provides additional lexical information other than conceptual definitions and part-of-speeches.

For instance, *orange* 橙色 is a "ColorValue|顏色值" but may play different grammatical functions such as subject/object, predicate, modifier. If *orange* 橙色 plays the role of object such as in (5), the sense definition should be applied in the composition process. However, in (6), *orange* 橙色 plays the role of modifier so operational expression should be applied. Possible lexical features for *orange* 橙色 are shown in Table 2.

(5) I like the color orange. 我喜歡橙色

def:{FondOf|喜歡:experiencer={speaker|說話者}

content={color({柳橙|orange})}}

(6) orange flowers 橙色的花

def :{flower|花:color={color({柳橙|orange})}}

詞彙訊息

詞彙:	橙色
詞性:	Nad
英文意涵:	orange color
Event Frame:	
Primitive relations:	implication={and({yellow|黃},{red|紅})}
Conflation of events:	
定義式: [編輯]	{color({柳橙|orange})}
操作式:	color={color({柳橙|orange})}
語義功能:	ColorValue|顏色值
語義特徵:	+color
展開式:	
WordNet 自動連結:	{orange.n.02, color.n.01, color.n.03, color.n.08, color.n.02}

Table 2: The encoded information for *orange* 橙色 in E-HowNet

3.2.2 Event Frame

Other than basic semantic expression, we like to know its event frame (i.e. arguments) while it plays the predicate role. Arguments of each event type are provided. Take {buy|買} as an example shown in Table 3. Whenever the event "buy" occurs, agent (buyer), theme (commodity), and source (seller) indicated in the event frame of {buy|買} must participated in it. They are crucial to establish relations between constituents of a phrase/sentence and are necessary elements for doing semantic composition.

義原訊息

Primitive:	buy|買
定義式:	{take|取:means={pay|付}}
Event frame	ACTOR:*buyer*{agent},THEME:*commodity*{theme},SOURCE:*seller*{source}
Primitive relations:	implication={pay|付}; same event={sell|賣}
Conflation of events:	agent({buy|買})=target({sell|賣}); theme({buy|買})=theme({sell|賣}); source({buy|買})=agent({sell|賣})
語義功能:	
語義特徵:	
WordNet 連結:	buy.v.1,(同義)
WordNet 自動連結:	{get.v.01, buy.v.01, entity.n.01, abstraction.n.06, communication.n.02}
[編輯]	
Event Frame	
EHowNet Frame:	agent,possession
TreeBank Frame:	{agent,theme}{agent,goal}
[編輯]	

Table 3: Information table of { buy|買}

3.2.3 Semantic Function

A lexical sense may have different meaning facets. For example, {老師|teacher} in E-HowNet is a subcategory of {專業人士|professional} therefore a hyponym of {human|人}. However, 'teacher', also denotes a kind of occupation and should be regarded as an 'occupation value' as well. Thus, we mark the semantic function of ' teacher' as {OccupationValue|職業值}to include both meaning facets. The same phonamenon occurs for most subnodes of {專業人士|professional}, so we simply mark the semantic function of {專業人士|professional} as {OccupationValue|職業值} and subnodes of {專業人士|professional} will inherit the feature automatically.

3.2.4 Other Semantic Links

E-HowNet ontology is constructed by is-a relation which has the inherent property. Hyponym concepts inherit the properties of hypernym concepts. There are also many other important relations other than is-a relation among concepts. We can look back to Table 3 to see what the relations might be. The primitive relations of {buy|買} are "implication={pay|付}" and "same event={sell|賣} ". That means whenever a event "buy" occurs, that imply the event "pay (money)" happened as well. The event "buy" and event "sell" are actually the same event but just mentioned from different participators' points of view. Since they are the same event, we can also derive the conflation of events of {buy|買} are: agent({buy|買})=target({sell|賣}); theme({buy|買})=theme({sell|賣});

source({buy|買})=agent({sell|賣}). Those conflations are indicated with lexical entries in E-HowNet 2.0.

4. On-line Systems of E-HowNet 2.0

The current E-HowNet ontology shown on the web is the result of automatic constructed by a computer program according to the pre-defined hierarchical structure of primitive and basic concepts as well as E-HowNet expressions, which contain more than 88,000 lexical senses. Based on this system, the E-HowNet Relation Database is also constructed to provide a new direction of clustering concepts.

4.1 Automatic Ontology Reconstruction

To construct a complete lexical taxonomy, we use a strategy that categorizes concepts automatically (Chen et al, 2010).

Step 1. Attach lexical senses. Words and associated sense expressions are first attached to the top level ontology nodes according to their head concepts. For instance, the head concept of the expression '{choose|選擇:manner={cautious|慎}}' is 'choose|選擇'.

Step 2. Sub-categorization by attribute-values. Lexical concepts with the same semantic head are further sub-categorized according to their attribute values. Lexicons that have the same attribute values share specific characteristics; therefore further sub-categorization is performed based on the distinct attribute-values of the lexicons.

Step 3. Repeat step 2 if there are too many lexical concepts in one category. Although the lexicons are classified after step (2), some sub-categories might still contain too many lexicons. In this situation, we further classify the lexicons in the sub-category with other attribute-values until all sub-categories contain fewer members than a predefined threshold, or all members of a category are synonyms.

In case of revisions of lexical sense expressions or nodes of conceptual hierarchy, the ontology reconstruction system may re-attach each lexical entry to appropriated ontological nodes and results a new ontology. For instance, 貓頭鷹 'owl' is defined as {bird|禽 :predication={SelfMove|自移 :duration={night|夜},theme={~}}} and we can find several similar words defineds as the same way. Therefore, {貓頭鷹|owl} is chosen as basic concepts under {bird|鳥} and lexical entries with the above definition are all redefined as {貓頭鷹|owl} and placed in the same subcategory of {bird|鳥}.

However, some concepts do not have natural head (hypernymous) concepts and it is problematic for classification. For example, relations such as kinship relations (e.g. father's younger brother 叔父) and directions (e.g. northeast 東北方) are not suitable to be defined by their hypernyms but the compositions of basic functions. In E-HowNet 2.0, we set rules to classify words of this type according to their first function. Therefore, "father's younger brother 叔父" and "northeast 東北方" are attached to {YoungerBrother|弟} and {north|北} respectively.

Some attribute-type and value-type words are not distinguishable due to having the same sematic head and need to be differentiated by marking the semantic function. For example, "price 價位" is defined as def:{price|價格} and "mid-priced 中價位" is defined as def:{price|價格 :value={intermediate|中等}}. We need to mark semantic function of "price 價位" as ATTRIBUTE and "mid-priced 中價位" as PriceValue|價格值 then they can be attached to appropriate position in the ontology.

4.2 The E-HowNet Relation Database

With the rapid development of semantic networks, related search tools have progressively emerged. Users can set query criteria to find words that match the condition. In Chinese WordNet[2], the interface allows users to enter a keyword and the result shows both lexical meanings and semantic relations of that word. In addition to word senses and relations, Extended-HowNet also clearly presents the position of the word in the ontology. Take *bird* 鳥 for an example, the search results of Chinese WordNet and Extended-HowNet are shown in figure 4 and figure 5 respectively.



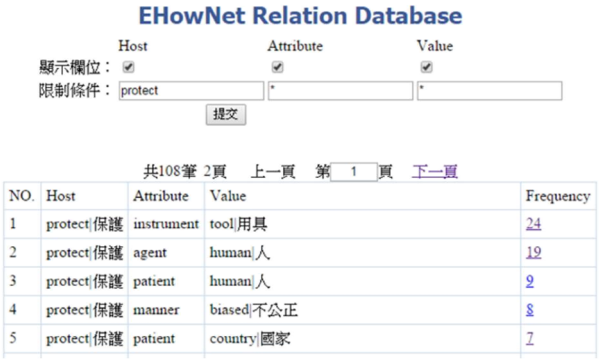
Figure 4: 鳥(bird) in Chinese WordNet



Figure 5: 鳥(bird) in E-HowNet

[2] http://lope.linguistics.ntu.edu.tw/cwn/query/

However, this keyword-search method cannot succeed in finding semantic relation among entities. We advocate that a system should be more flexible in searching specific semantic relations and words should be able to further classify into categories according to their semantic relations. For example, if a user want to find "all entities that contain the function of protection", or "all entities that denote some kind of protectors", 墨鏡"sunglasses" (tool to protect eyes) and 專利"patent" (rights to protect intellectual properties) are the possible answers for the former, and 護花使者"lady's escort"(human to protect the female) and 保鏢 "guard for goods/persons in transit"(human to protect goods/persons) are for the latter.



NO.	Host	Attribute	Value	Frequency
1	protect|保護	instrument	tool|用具	24
2	protect|保護	agent	human|人	19
3	protect|保護	patient	human|人	9
4	protect|保護	manner	biased|不公正	8
5	protect|保護	patient	country|國家	7

Figure 6: Lexicon categories with host of 'protect

From the word similarity point of view, the degree of similarity for 墨鏡"sunglasses" and 專利"patent" should not be high no matter which ontology is applied, for the former is a concrete object but the latter is an abstract one. Therefore, their distance in an ontology is also far from each other. However, they could be dynamically clustered to a category while certain semantic constrain is applied.



NO.	WORD	POS	EHOWNET
1	介胄	Nab	{tool|用具:telic={protect|保護:instrument={~},while={fight|爭鬥}}}
2	甲胄	Nab	{tool|用具:telic={protect|保護:instrument={~},TimePoint={fight|爭鬥}}}
3	耳栓	Nab	{tool|用具:telic={protect|保護:instrument={~},patient={耳朵|ear}}}
4	耳塞	Nab	{tool|用具:telic={protect|保護:patient={耳朵|ear},instrument={~}}}
5	帙	Nab	{tool|用具:telic={protect|保護:instrument={~},patient={publications|書刊}}}
6	面罩	Nab	{tool|用具:telic={protect|保護:instrument={~},patient={臉|face}}}
7	書套	Nab	{tool|用具:telic={protect|保護:instrument={~},patient={publications|書刊}}}
8	針箍	Nab	{tool|用具:telic={protect|保護:instrument={~}}}
9	蛇籠	Nab	{tool|用具:telic={protect|保護:instrument={~},patient={地基|ground}}}
10	頂針兒	Nab	{tool|用具:telic={protect|保護:instrument={~}}}

Figure 7: Lexicons with host of 'protect', attribute of 'instrument' and value of 'tool'

Such dynamic semantic clustering search can achieve a comprehensive hierarchical overview for words with the same sematic relation and provide a practically useful new query tool for lexical semantic studies. Therefore, we also developed the E-HowNet Relation Database[3] to achieve this pupose. Taxonomically unrelated but conceptually related concepts can also be computably associated through their E-HowNet definitions. Words with the same semantic relation should be able to group together no matter how far the distance is from the ontology point of view. An example of the E-HowNet Relation Database is given as figure (6) and (7) below.

Figure (6) shows lexicon categories with host of 'protect'. Once clicking the first category, the system will list all lexicons with host of 'protect', attribute of 'instrument' and value of 'tool', as shown in Figure (7).

5. Conclusions and Future Work

HowNet proposed a new model to represent lexical knowledge, inspiring us to expand this framework to achieve the task of mechanical natural language understanding. E-HowNet confines each concept to a semantic type and defines the relation between these types. E-HowNet has a uniform representation system for both function words and content words to achieve semantic composition, such that meaning representations for morphemes, words, phrases, and sentences can be uniformly represented under the same framework. New concepts can be defined by previously known concepts and definitions can be dynamically decomposed into lower level representations until the ground-level definition is reached. In E-HowNet 2.0 we reorganized the hierarchical structure of primitives and basic Concepts. Near-canonical representation thus can be achieved at a suitable level of representation for synonyms or paraphrases. We also suggested compositional functions to extend the expression of new concepts and make word and phrase definitions more detailed and accurate.

To facilitate automatic semantic composition and language understanding, E-HowNet 2.0 provides additional lexical information other than conceptual definitions and part-of-speeches. Operational expression, event frames, semantic functions etc. are provided for lexical entries to facilitate semantic composition processing.

The E-HowNet 2.0 ontology online is able to demonstrate the taxonomy, sub-categories, and lexicons in a hierarchical tree structure. In addition, we provide a new direction for clustering concepts. Taxonomically unrelated but conceptually related concepts can also be computably associated through their lexical definitions.

There are still many obstacles to achieving the goal of automatically extracting knowledge from language. Apart from sense disambiguation, discord between syntactic structures and their associated semantic representations is another critical problem. To reveal all fine-grained semantic relations for constituents at different levels of syntactic structure, we had just start the project of E-HowNet SemBank annotation. Gap filling processes, as discussed, need to be an integral part of the mechanism. Normalization of sense representation to achieve real canonical sense representation and fine-grained semantic representations are also indispensable. Our future research will continue to address these issues.

[3] http://ckip.iis.sinica.edu.tw/~mhbai/relation/

6. Bibliographical References

Dorr, Bonnie J., Gina-Anne Levow, and Dekang Lin, 2000, Construction of Chinese-English Semantic Hierarchy for Information Retrieval, Workshop on English-Chinese Cross Language Information Retrieval, International Conference on Chinese Language Computing, Chicago, IL, 187-194.

Chen, Keh-Jiann, Shu-Ling Huang, Yueh-Yin Shih, and Yi-Jun Chen, 2005, Extended-HowNet- A Representational Framework for Concepts, OntoLex 2005- Ontologies and Lexical Resources IJCNLP-05 Workshop, Jeju Island, South Korea

Chen, Keh-Jiann, Shu-Ling Huang, Yueh-Yin Shih and Yi-Jun Chen 陳克健, 黃淑齡, 施悅音, 陳怡君. 2004. Multi-level Definitions and Complex Relations in Extended-HowNet 多層次概念定義與複雜關係表達—繁體字知網的新增架構 Workshop on Chinese Lexical Semantics, Beijing University.

Chen, Wei-Te, Su-Chu Lin, Shu-Ling Huang, You-Shan Chung, and Keh-Jiann Chen, 2010, E-HowNet and Automatic Construction of a Lexical Ontology, the 23rd International Conference on Computational Linguistics, Beijng, China.

Chen, Yi-Jun, Shu-Ling Huang, Yueh-Yin Shih and Keh-Jiann Chen 陳怡君, 黃淑齡, 施悅音, 陳克健. 2005. Semantic Representation and Definitions for Function Words in Extended-HowNet, 繁體字知網架構下之功能詞表達初探 Workshop on Chinese Lexical Semantics, Xiamen University.

Chung, You-Shan, Shu-Ling Huang and Keh-Jiann Chen. 2007. Modality and Modal Sense Representation in E-HowNet. Proceedings of Paclic 21:136-145.

CKIP. Semantic Roles and Semantic Role Labeling. Technical Report no.15-01

Dong, Zendong. 1999. Bigger Context and Better Understanding -- Expectation on Future MT Technology, Proc. of the International Conference on Machine Translation & Computer Language Information Processing 17-25.

Dong Zendong, and Qiang Dong. 2006, HowNet and the Computation of Meaning. World Scientific Publishing Co. Pte. Ltd.

Dowty, David R. 1989. On the semantic content of the notion 'thematic role'. Properties, types and meanings, vol. II, ed. by Barbara Partee, Gennaro Chierchia, and Ray Turner, 69-130. Dordrecht: Kluwer.

Huang, Shu-Ling, You-Shan Chung and Keh-Jiann Chen. 2008. E-HowNet: the Expansion of HowNet, the First National HowNet Workshop. Beijing, China.

Liu, Qun, and Sujian Li 劉群, 李素建. 2002. Word Similarity Computing Based on How-net 基於《知網》的辭彙語義相似度計算. In *International Journal of Computational Linguistics and Chinese Language Processing* 7(2):59-76.

Pustejovsky, James. 1995. The Generative Lexicon. Cambridge: MIT Press.

Schank, R., 1975, *Conceptual Information Processing*, Elsevier Science, New York.

Wierzbicka, A., 1972, *Semantic Primitives*, Athenaum, Frankfurt.

7. Language Resource References

Fellbaum, Christiane. WordNet. http://wordnet.princeton.edu/

Fillmore, Charles. FrameNet. http://www.icsi.berkeley.edu/~framenet/

The Circumstantial Event Ontology (CEO) and ECB+/CEO: an Ontology and Corpus for Implicit Causal Relations between Events

Roxane Segers[*], **Tommaso Caselli** [∓], **Piek Vossen** [*]
[*]Vrije Universiteit Amsterdam, The Netherlands
{r.h.segers, p.t.j.m.vossen}@vu.nl
[∓]Rijksuniversiteit Groningen, The Netherlands
t.caselli@rug.nl

Abstract

In this paper, we describe the Circumstantial Event Ontology (CEO), a newly developed ontology for calamity events that models semantic circumstantial relations between event classes, where we define circumstantial as inferred implicit causal relations. The circumstantial relations are inferred from the assertions of the event classes that involve a change to the same property of a participant. Our model captures that the change yielded by one event, explains to people the happening of the next event when observed. We describe the meta model and the contents of the ontology, the creation of a manually annotated corpus for circumstantial relations based on ECB+ and the first results on the evaluation of the ontology.

Keywords: Ontology, Event Modeling, Event Chaining, Causality, Annotated Corpora, Text Mining, Semantic Role Labeling

1. Introduction

Suppose we read a sequence such as: *"Today was the burial of Mary Johnson, that was broadcasted live on TV. The pop star died last week when her yacht capsized and sunk after hitting a tanker. Johnson was not wearing a life jacket and drowned."* As it is clear to most readers, but implicit in this sentence, there must be some relation between "hit", "capsize", "sinking", "drown", "die" and "burial". The interpretation of this sentence as a text, i.e. a unitary message (De Beaugrande and Dressler, 1981), requires some coherence relations between the events, that are not explicitly expressed. In the context of this occurrence, it is normal for a human reader to interpret the events as a chain of *consequences*. This coherence is the result of the fact that the events imply changes on a set of shared properties.

We consider this type of relations between event pairs as a case of *circumstantial relations*, i.e. relations between events which allows interpreting their occurrence in the world, and in a text, as coherent. A circumstantial relation makes clear "why" something happened, without necessarily predicting it. Circumstantial relations are a set of relations which include temporal, causal, entailment, prevention and contingency relations, among others.

We distinguish two types of circumstantial relations: episodic and semantic. An *episodic* circumstantial relation is a relation that holds between a pair of specific actual event instances in a specific context, where their connection is necessary to understand what is described in a meaningful and coherent way. For instance, the relation between the events "[a yacht] sunk" and "hitting [a tanker]" is a case of an episodic circumstantial relation: both events may happen independently without implying the other necessarily, but when described in the same context, or circumstance, a connection is created that explains their occurrence as a dependent relation.

On the other hand, we define *semantic* circumstantial relations as a relation that holds between event classes (abstracting from actual event instances), where an event of class A gives rise to another event of class B or vice versa, based on shared properties in the formalization of the classes.

For instance: the class "ceo:Shooting" has a semantic circumstantial relation with the class "ceo:Impacting", because they both share the property of translocation of an object from location X to Y. The latter as the outcome of the event, and the former as a condition to take place. Likewise, an "Impacting" event may, but not necessarily, lead to "ceo:Injuring" or "ceo:Damaging", which is based on the shared property of some object being damaged.

Modeling these relations provides a means to track chains of logically related events and their shared participants within and across documents. Semantic circumstantial relations define possible explanatory sequences of events, but not the actual explanatory sequences. Episodic relations, on the other hand, define circumstantial relations that are dependent on the actual occurrences of events in the world.

The Circumstantial Event Ontology (CEO) (Segers et al., 2017) [1], described in this paper, models such semantic relations, based on *shared properties* of the event classes with the aim to support the detection of episodic circumstantial relations in texts.

Modeling these semantic relations in an ontology will allow us to 1.) abstract over the different lexical realizations of the same concept (i.e. at an event mention level); and 2.) facilitate reasoning between event classes and enrich the extraction of information for event knowledge and event sequences.

The remainder of this paper is organized as follows: in section 2., we describe related work; in section 3. we explain the meta model and the development of CEO. Section 4. describes an annotated corpus of episodic circumstantial relations, that has been used to run preliminary experiments for the evaluation of the CEO. Experiments and results are

[1]CEO is publicly available with a CC-BY-SA license at `https://github.com/newsreader/eso-and-ceo`.

described in 5.. Finally, conclusions and future work are reported in section 6.

2. Related Work

Existing ontologies and models such as SUMO (Niles and Pease, 2001) and FrameNet (Ruppenhofer et al., 2006) provide explicit causal relations between event classes (SUMO), or preceding and causal relations (FrameNet). These causal relations are strict, meaning that if A happens, then B must happen as well. However, our relations are circumstantial, meaning that some instance of event class C and D can happen independently, but given the circumstance that they coincide, C likely implies D or D is likely implied by C because they share a property or a set of properties. The implication is however not necessary.

Previous work on the encoding of semantic relations between event pairs has focused on specific subsets of circumstantial relations. For instance, one example is the encoding of the entailment relations in WordNet (Fellbaum, 1998). With respect to the WordNet approach, we abstract from various event types (i.e. lexical items) and do not depend on relations defined at a synset level, by formalizing event knowledge and relations in an ontology. We also provide more details on the property involved.

Another related approach are narrative chains (Chambers and Jurafsky, 2010), that provide chains of various event mentions. However, the relation between these mentions is not specified explicitly but based on co-occurrence of participants and a basic precedence relation. Manual inspection of these chains revealed that dissimilar relations are implied within these chains, varying from temporal ordering, to episodic, up to causal.

The Penn Discourse TreeBank (PDTB) (Prasad et al., 2007) annotates contingency relations, of which causal relations are a subclass. In PDTB, the focus of the annotation is between two Abstract Objects (called Arg1 and Arg2), corresponding to discourse units, rather than event mentions. The contingency relation is annotated either in presence of an explicit connective, i.e. a lexical item, between the two abstract objects, or implicitly, by adjacency in discourse. In our approach, contingency relations are one of the possible values which express circumstantial relations, and, most importantly, they are independent of the presence of connectives or adjacency in discourse, but grounded on (shared) properties of events.

A related resource is the Rich Event Ontology (REO) (Brown et al., 2017), that provides an independent semantic backbone to different lexical resources such as FrameNet and VerbNet. REO will have explicit causal relations between event classes as well as predefined pre- and post conditions. However, these relations are more strictly defined and on class level. On the other hand, CEO maintains a looser definition in terms of causality, and takes into account the roles affected by the event and the circumstantial relation.

A resource such as CEO is envisioned to be of added value for several NLP tasks such as script mining, question answering, information extraction, and textual entailment, among others. Furthermore, the explicitly defined relations between events can be of help in reconstructing storylines (Van den Akker et al., 2010; Vossen et al., 2015) and improve the coherence of existing narrative chain models (Chambers and Jurafsky, 2010).

pre situation:	damaging-undergoer	inState	damaging-state-1
	damaging-state-1	hasRelativeValue	"+"
post situation:	damaging-undergoer	inState	damaging-state-2
	damaging-state-2	hasRelativeValue	"-"
	damaging-undergoer	isDamaged	true
	damaging-undergoer	hasDamage	damaging-damage
	damaging-damage	hasNegativeEffectOn	activity

Figure 1: The ESO assertions for the class eso:Damaging

3. The Circumstantial Event Ontology

CEO builds upon an existing event ontology called the Event and Implied Situation Ontology (ESO) (Segers et al., 2016). ESO is designed to run over the output of Semantic Role Labeling systems by making explicit both the ontological type of the predicative element and the situation that holds before, during and after the predicate. Each so called pre-, post- and during situation consists of a set of properties and roles that define what holds true. For instance, as can be seen in Figure 1, the pre- and post-situations of the event class "eso:Damaging" define:

- that something is in a "relatively plus (+)" state (pre-situation);
- that this something is in a "relatively less (-)" state, i.e. it underwent a loss or a negative change, relatively to the state before the damaging (post-situation);
- that some object is in a state "damaged" after the event (post-situation);
- that something has some damage which has some negative effect on some activity (post-situation).

ESO allows to track chains of states and changes over time, whether explicitly reported or inferred. However, ESO does not provide any explicit definition on what event class logically precedes or follows some other event class, i.e. the pre-, post- and during situations provide only descriptions of properties of the participants of the event in analysis. In CEO, we further developed the event hierarchy of ESO, and the expressiveness of the pre-, post-, and during situations in order to infer the circumstantial semantic relations between the classes.

3.1. The CEO Meta Model

CEO is an OWL2 ontology and its meta model fully adopts and extends the ESO model (Segers et al., 2016). The reasons to reuse and extend it are twofold: 1.) The ESO classes and roles are mapped to FrameNet, therefore we can rely on existing SRL techniques and models to instantiate CEO (Björkelund et al., 2009; de Lacalle et al., 2016); 2.) ESO provides a model that defines what situation, or state, is true before and after an event, thereby already providing the

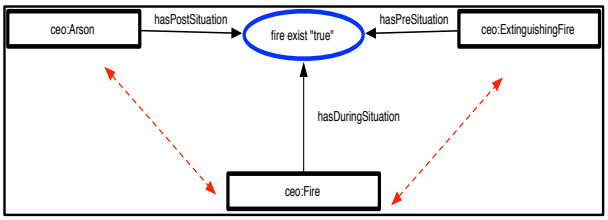


Figure 2: The meta model of CEO and the chaining of classes by shared properties in the pre-, during-, and post situations.

initial hooks to infer the circumstantial semantic relations. This principle is illustrated in Figure 2. The black boxes represent event classes in CEO; each class has at least one assertion (ceo:fire exist ”true”) that is shared with two other classes. In the case of “ceo:Arson” it is part of its post situation; it is the during situation of “ceo:Fire” and the pre situation of “ceo:ExtinguishingFire”. Based on these shared properties we can infer a semantic circumstantial relation that is in this model represented by the red arrows. Whether the shared property is in a pre-, during-, or post situation implicitly defines the logical order of the events.

The full expressiveness of a class in CEO.owl is illustrated in Figure 3, where we transcribed the class “ceo:Arson” and its assertions in a human readable format. Each class has a subclass relation (subclassOf) and a definition (Definition). Furthermore, the class “ceo:Arson” is mapped to FrameNet (fn:Action) and SUMO (sumo:Arson). All mappings were created manually. Next, we show the assertions in the pre- (pre situation), during- (during situation), and post situation (post situation). Each assertion consists of a property and one or more roles that are mapped to FrameNet (role mappings are not shown).

CEO properties consist of 1.) binary properties where two roles are connected, e.g. (hasPurpose, deteriorates), 2.) unary properties that connect a role with a boolean expression ”true” or ”false” (e.g. inDanger), or a relative value ”+” or ”-” (e.g. hasRelativeValue). For some roles, we defined an OWL existential restriction if no instance can be found in a text. In this case, the role will be instantiated with a blank node and some URI. In Figure 3, this occurs for the roles “damaging-state-1” and “damaging-state-2”.

Figure 4 illustrates the inference capabilities of CEO using FrameNet-based role labeling. Only those assertions can be fired and instantiated if an instance of the CEO role is found via the FrameNet mappings. In this case, there is no Frame element and instance found for the CEO role ”damage”, hence the assertion can not be instantiated. In line 2, we see how a blank node is created for the role ”damaging-state-1”, encoded here as ”abc123”.

In short, the assertions in Figure 4 define that 1.) the fire does not exist before the Arson (line 5), but it does during (line 10) and after (line 21); 2.) Mary is in offense during (line 14) and after (line 22) the arson of the stables, 3.) the stables and the village are in danger during (lines 12 and 13) and after (lines 19 and 20) the arson; and 4.) the stables are damaged after the arson (line 18).[2]

0	**-Arson**		
1	subclassOf: IntentionalDamaging		
2	Definition: “The subclass of IntentionalDamaging where someone		
3	deliberately sets some object on fire.”		
4	CLASS MAPPINGS:		
5	closeMatch: fn:Arson		
6	closeMatch: sumo:Arson		
7	ASSERTIONS:		
8	*pre situation*		
9	agent	hasPurpose	purpose
10	undergoer	inState	damaging-state-1
11	damaging-state-1	hasRelativeValue	"+"
12	undergoer	inDanger	false
13	fire	exist	false
14	*during situation*		
15	agent	deteriorates	undergoer
16	undergoer	inDeterioration	true
17	agent	hasIntention	“deterioration-of-undergoer”
18	fire	exist	true
19	agent	endangers	undergoer
20	undergoer	inDanger	true
21	place	inDanger	true
22	agent	inOffense	true
23	*post situation*		
24	undergoer	inState	damaging-state-2
25	damaging-state-2	hasRelativeValue	"-"
26	undergoer	isDamaged	true
27	undergoer	inDanger	true
28	place	inDanger	true
29	fire	exist	true
30	undergoer	hasDamage	damage
31	damage	hasNegativeEffectOn	activity
32	agent	inOffense	true

Figure 3: The expressiveness of an event class in CEO, including subclass relation, mappings and assertions and roles in the pre-, post, and during situation.

3.2. Semantic Circumstantial Relations between Event Classes

CEO is modeled in such way that it allows for inferencing, chaining classes, and reasoning over the assertions, roles, and role instances.

For chaining the event classes, the most basic way is to track paths trough the ontology, based on shared properties in the class assertions. This is illustrated in Figure 5. Here, in each box we show eight different sentences related to the same Arson incident. The property in red (inOffense ”true”) is in the post situation of “ceo:Arson” and in the pre- situation of the event class “ceo:Arresting”. Likewise, the property ”fire exist true”, which is marked here in orange, ties a circumstantial relation from “ceo:Arson” to the class “ceo:Fire”, and from this latter class to “ceo:ExtinguishingFire”. As such, we can chain the event mentions based on shared semantic properties. To exploit the model at its maximum, a reasoner will have to take into account the properties and their values, the roles, as well as the role instances.

[2] A full transcription of the CEO classes including all assertions, the inherited assertions and example sentences that show the instantiation can be found at `https://github.com/newsreader/eso-and-ceo`.

Arson
"[Mary] deliberately **set** [three stables] **on fire** in [John's village]."

1	*pre situation*		
2	three stables	inState	"abc123"
3	"abc123"	hasRelativeValue	"+"
4	three stables	inDanger	false
5	fire	exist	false
6	*during situation*		
7	Mary	deteriorates	three stables
8	three stables	inDeterioration	true
9	Mary	hasIntention	"deterioration-of-undergoer"
10	fire	exist	true
11	Mary	endangers	three stables
12	three stables	inDanger	true
13	John's village	inDanger	true
14	Mary	inOffense	true
15	*post situation*		
16	three stables	inState	"abc456"
17	"abc456"	hasRelativeValue	"-"
18	three stables	isDamaged	true
19	three stables	inDanger	true
20	John's village	inDanger	true
21	fire	exist	true
22	Mary	inOffense	true

Figure 4: Example of what the CEO assertions infer from a SRL labeled sentence for the pre-, during and post situation of the event.

3.3. Building the CEO

CEO is designed to capture chains of events in newswire, more specifically calamity events. We define a calamity event as any event where some situation turns from relatively positive to some relatively negative state due to changes in the world, either intentional or not. Event classes that define processes where some agent tries to improve some situation in reaction to some calamity are also modeled in CEO, e.g. going from a relatively negative situation back to a relatively positive situation. Examples of calamity event classes are "CyberAttack" and "Earthquake". Examples of event classes where an attempt to some improvement of a situation is made are "Repairing" and "Evacuation".

ESO already provides event classes for calamities, though the coverage is rather limited, because it was designed for the economic-financial domain. As such, we massively extended the hierarchy from the initial 63 event classes in ESO to the 223 event classes in CEO 1.0. To the best of our knowledge, no formal ontology specific for calamities and the inter-event relations exist. Some thesauri such as the IPTC [3] contain terms for calamities but these are not formalized and provide few relations. Therefore, we decided to define a new model, reusing existing resources as much as possible.

As a starting point for the identification of instances of the calamity classes in CEO, we used Chamber's narrative chains (Chambers and Jurafsky, 2010). This selection was made manually, based on at least three calamity events per event chain. We also manually selected FrameNet frames that capture calamity events and we used the SUMO ontology as a backbone for modeling our initial list of verbs and frames. Finally, we defined SKOS mappings from each CEO event class to FrameNet and SUMO[4], thus providing the opportunity to use CEO on SRL labeled text as well as to find the vocabulary expressing calamities by means of the lexical units mapped to frames in FrameNet and the mappings to Princeton WordNet that are defined in SUMO. An overview and specification of all modeling decisions regarding class selection, class hierarchy and defining the assertions, properties, roles and role mappings to FrameNet can be found in the CEO documentation. [5]

3.4. Contents of CEO

In January 2018, we released CEO 1.0. The ontology consists of 223 event classes of which 189 are fully modeled with pre-, during and post situations. For 34 classes, we have a minimal set of assertions. These classes pertain to natural disasters and will be modeled for CEO version 1.1. Further, we defined 92 binary properties and 29 unary properties. In total, 189 unique situation rules were defined that consist of 192 binary situation rule assertions and 264 unary rule assertions.

Further, all classes are mapped to FrameNet frames (265 mappings) and SUMO classes (195 mappings), and the CEO roles to FrameNet elements (265 mappings).

4. The ECB+/CEO Corpus

In addition to the CEO, we developed a corpus of annotated circumstantial event relations. For this, we build upon an existing corpus, specifically annotated for event coreference: the ECB+ Corpus (Cybulska and Vossen, 2014). ECB+ consists of 984 news articles divided over 42 topics. From these topics, we manually selected 22 topics (508 articles) that cover calamities such as earthquakes, murders, hijacks and arson. In ECB+, only the most relevant event mentions are manually annotated. For ECB+/CEO, we automatically extended the set of annotated event mentions by applying a state-of-the art machine learning based system [6]. Two linguistically trained annotators were hired for the selection of relevant calamity events and the annotation of circumstantial relations.

More specifically, the annotation procedure consisted of the following steps:

1. Select event mentions denoting calamity events and generate corresponding event instances;
2. Extending existing ECB+ coreference sets with new mentions;
3. Creating new coreference sets for new calamity mentions;
4. Creating circumstantial relations (CEO links) between the event instances where each instance refers to a set of coreferential mentions.

[3] https://iptc.org/

[4] https://www.w3.org/2004/02/skos/

[5] https://github.com/newsreader/eso-and-ceo

[6] (Caselli and Morante, to appear) https://github.com/cltl/TimeMLEventTrigger

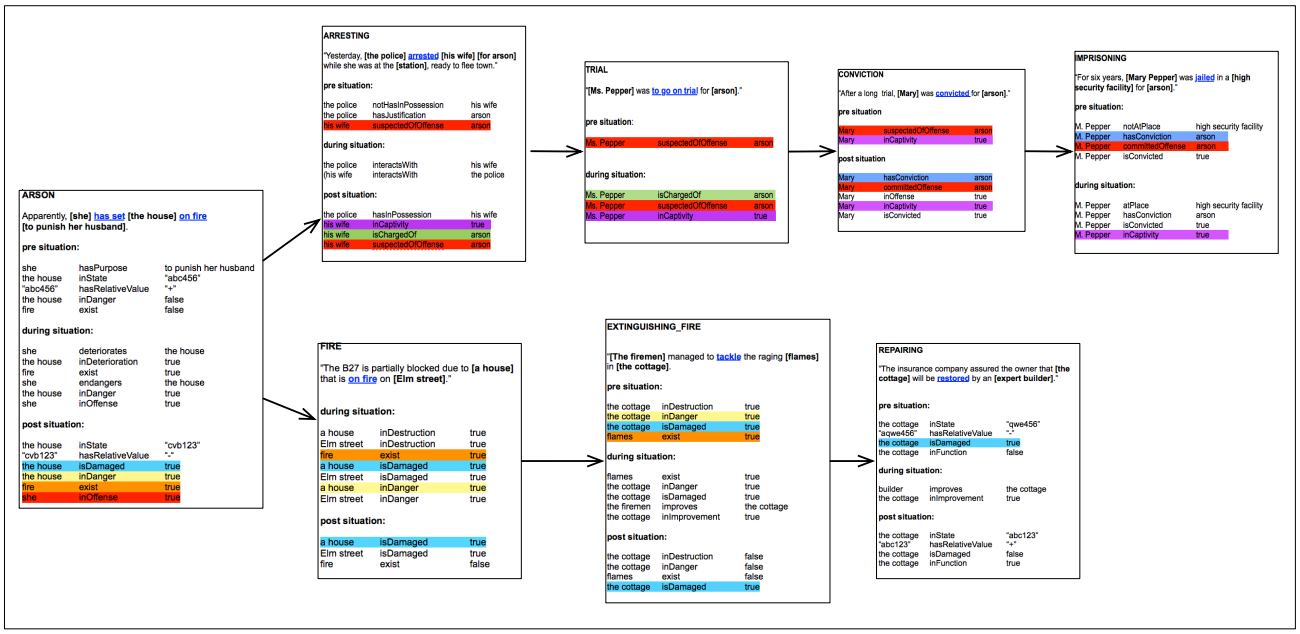


Figure 5: Inferring circumstantial relations from shared properties in the pre-, post-, and during assertions between event expressions in eight sentences.

Annotators were asked to connect pairs of calamity event instances with a CEO link if one event instance could be used to explain the occurrence of the other.

For the value of a CEO relation, the annotators could opt for the default value (has circumstantial post event - `HCPE`) or the subset relation (`hasSubevent`).[7] The `HCPE` relation is directional and is defined from a source, or trigger, event to a target, or consequence, event.

We followed the original ECB+ annotation guidelines where applicable and we deviated on certain points. For instance, we only annotated calamity event mentions; the participants, locations and time expressions were not annotated. Furthermore, speech acts and events expressing cognition, perception and emotions were excluded for the annotation.

Negated events are annotated and added to the CEO links, as a statement that something did *not* happen points at the fact that it usually does happen (e.g. he was *shot* but not *injured* severely).

For the definition of coreference, we specified that two event mentions are coreferential if they (more or less) denote the same concept, and they share the same participants, time, and location. Event coreference was only annotated within document, and not across documents, like in ECB+.

In table 1 we show the results of the annotation. In total, 508 articles were annotated for ECB+/CEO which resulted in 3038 new event instances expressing calamities. Further, 3448 new event coreference sets were created. Not every instance and coreference set ends up in a CEO link as for many events no circumstantial event or subevent is present in the text. As such, 2437 CEO links were created of which 2244 circumstantial ones and 193 subevents. On average, every ECB+/CEO article contains about 7 new coreference sets and about 5 different circumstantial relations.

	ECB+	ECB+/CEO
Instances	3323	3038
Coreference sets	3323	3448
CEO relations	-	2437
- of which Circumstantial	-	2244
- of which subEvent	-	193

Table 1: Overview of the annotations made for ECB+/CEO in contrast with ECB+ for the topics annotated

For the annotation, we used the CAT annotation tool (Bartalesi Lenzi et al., 2012) which outputs the annotations in XML. In terms of annotation effort, a single article took about 30 minutes to annotate on average. The corpus and the annotation guidelines are publicly available at `https://github.com/newsreader/eso-and-ceo`.

Inter Annotator Agreement For the calculation of the Inter Annotator Agreement (IAA), we selected 25 articles from five different topics in ECB+/CEO covering variation in article length and complexity. The evaluation was carried out on the CEO links. Agreement was calculated on the existence, or identification, of CEO links.

CEO links are created between event instances, where each instance points to a set of event mentions in the document. These sets are defined as coreference relations. To evaluate the quality and reliability of the CEO links, we calculated the inter-annotator agreement (IAA) by means of Cohen's Kappa score (Cohen, 1960). We obtained a value of 0.54. To better understand the reasons behind such a score, we randomly inspected some annotated articles. As an outcome of this inspection, it appeared that the major differences beween the annotators were due to mismatches

[7] Subevents are currently not modeled in CEO, but they were annotated for future experiments and evaluations.

in the coreference sets, rather than in actual disagreements on the presence/absence of a CEO link. As such, we manually added a post processing step to align those coreference sets where either one or both annotators missed one or more mentions. To avoid introducing bias, we harmonized the coreference sets only if there were no conceptual differences between them. To clarify, if annotator A created one coreference set with three different mentions, and annotator B created two sets with the same mentions, we did not merge the sets of annotator B. With this post processing step, we solved 107 cases of partial disagreements on event coreference.
After this, we calculated again the IAA Cohen's kappa and reached a score of 0.76. Following Landis and Koch (1977), a score between 0.61 and 0.80 is considered substantial.
Both reported kappa scores are based on 21 out of the 25 initial articles. For four articles, the annotators agreed that there were no CEO links at all, and thus we excluded them.

Analysis of the disagreements We inspected some cases of clear disagreements in the annotated CEO links. These disagreements relate to differences in interpretation and to some unavoidable errors. For differences in interpretation, we see that the annotators disagreed whether some mention denoted the same concept or not. For instance, A1 created a CEO link between "suicide, hang" and "dead", while A2 interpreted all three mentions as denoting the same concept and did not create a CEO link. Further, there are disagreements on whether or not some mention still expresses a calamity and aftermath. As such, most agreements where e.g. A1 added an additional CEO relation and A2 did not, the relation leans towards a episodical one and not a semantic one. For those CEO relations for which there is agreement, these episodical relations are sparse. Further, we did not see any cases where the annotators disagreed on the type of the relation (HCPE or subEvent), or disagree on the directionality of the relation.

Creation of an initial CEO vocabulary For the annotation of ECB+/CEO, the annotators have focused on the creation of circumstantial relations between event instances. The instances themselves were not typed with a CEO class as it was thought to be too difficult for the annotators to do this. In order to know what class an event mention refers to, we extracted all mentions from the event coreference sets in the corpus. All mentions have been mapped manually to a CEO class. In total, 650 unique mentions were annotated with a total frequency of 3982. 14 unique mentions could not be mapped as they were too polysemous, 25 unique mentions were not mapped as they were out of domain. In terms of coverage, the vocabulary extracted from the corpus covers about 50% of the classes in CEO, meaning that 111 classes modeled in the ontology are not represented in the corpus. Likewise, the vocabulary points at 78 mentions that potentially can be added to the ontology, e.g. 'peace' and 'bankruptcy'. Most of these mentions however, point at very fine grained sub events related to trials and are basically out of domain.

5. Experiment and Evaluation on the ECB+/CEO corpus

We ran a first experiment to analyse to what extent CEO is able to connect events by means of semantic circumstantial relations, based on shared situation properties only. That implies that for this experiment, we deliberately did not take into account the CEO roles, the property values or the role instances to further fine tune the event chaining. The reason for this was twofold: 1.) we wanted to be able to analyse what CEO can achieve without any advanced reasoner and with just simple heuristics and 2.) we did not want to be affected by error propagation coming in from a NLP pipeline.
For this experiment, we developed the CEO-Pathfinder[8] (version 0.1) that checks for possible relations between events based on shared event properties in the pre-, post-, and during situations. CEO-Pathfinder compares all the mentions of events within a specified context window and checks the pre-, post- and during properties for matches. It uses a lexicon of 650 mentions that have been mapped to one or more CEO classes. The properties of associated classes (C1) and (C2) are compared as follows:

1. from a post situation in C1 to a pre situation in C2;
2. from a during situation in C1 to a pre situation in C2;
3. from a post situation in C1 to a during situation in C2;

We count the number of matching properties across classes of two mentions in both directions, assuming that the order of mention is not necessarily the order of the events in time. The software uses a threshold for the minimal matching properties. If below the threshold, no circumstantial relation is extracted. For both directions: C1 is circumstantial to C2 or C2 is circumstantial to C1, we then take the highest number of shared properties. If the shared properties are equal, the order of the mentions determines the direction of the circumstantial relation. The software can use the directly expressed properties or the inherited properties as well. We experimented with both options but got the best results with the directly expressed properties.
Finally, we implemented different context strategies for comparing mentions of events: 1) mentions within the same sentence (most strict), 2) one preceding and following sentence, 3) two preceding and following sentences, 4) all mentions in the full document.

Baseline system As a baseline, we compared all the mentions within the previous context windows 1, 2, 3 and 4 sentences, by assuming a CEO relation between all of them following the mention order. Table 2 shows the precision, recall and F1 results considering the order of the relation and ignoring the order (loose). B-1s is the baseline where we compare only mentions within the same sentence. B-3s is the baseline considering also one preceding and one following sentence, B-5 two preceding and following, and B-all the full document.
Not surprisingly, the precision results are all very low, both for order sensitive and loose matching. Highest recall is

[8] https://github.com/cltl/ceopathfinder

Baseline	B-1s	B-3s	B-5s	B-all
Precision order	0.236	0.202	0.188	0.144
Recall order	0.072	0.140	0.200	0.511
F1 order	0.110	0.166	0.194	0.225
Precision loose	0.556	0.432	0.386	0.282
Recall loose	0.169	0.300	0.409	0.999
F1 loose	0.259	0.354	0.397	0.439

Table 2: Result of the baseline system with different context windows

obtained for comparing all mentions ignoring the order: 0.99. When we take the order into account, we see that recall drops to 0.502. This means that about 50% of the event pairs with a CEO relation also are mentioned in their causal order. This pattern also holds for the other baselines where we compare mentions within limited contexts: recall drops by more or less 50% in all cases. Obviously, recall drops when we restrict the context, while precision increases. This means that there is a substantial amount of circumstantial relations expressed beyond the sentence boundary and event a context of five sentences that appears to be relevant.

Evaluation results In Table 3, we show the results for the CEO-Pathfinder exploiting the shared assertions from the ontology. The upper part represents the results when setting the threshold to one matching assertion and the lower part setting the threshold to two matching assertions. The different columns show the different context windows for comparing mentions similar to the previous baseline results. Overall, the precision and F1 results of the CEO-based approach outperform the baseline. We can see that the recall is much lower as can be expected.

1 assertion	CEO-1s	CEO-3s	CEO-5s	CEO-all
Precision order	0.455	0.400	0.379	0.311
Recall order	0.011	0.023	0.043	0.086
F1 order	0.021	0.044	0.077	0.135
Precision loose	0.650	0.563	0.512	0.420
Recall loose	0.015	0.033	0.058	0.117
F1 loose	0.029	0.062	0.104	0.183
2 assertions	CEO-1s	CEO-3s	CEO-5s	CEO-all
Precision order	0.645	0.498	0.464	0.405
Recall order	0.006	0.011	0.020	0.040
F1 order	0.011	0.021	0.038	0.073
Precision loose	0.710	0.556	0.509	0.439
Recall loose	0.006	0.012	0.021	0.044
F1 loose	0.012	0.023	0.041	0.079

Table 3: Results of Pathfinder using different settings (1 or 2 shared assertions) and varying context windows

The highest precision (0.710P) is achieved using the same sentence as a context window and, remarkably, ignoring the order. We also see that 2 shared assertions instead of 1, increases precision. Increasing the context window lowers precision and increases recall, where we have the highest recall (0.117R) and F1 (0.183F1) using the complete document and 1 shared assertion but ignoring the order.

To analyze the low recall, we collected all mentions for which the lexicon did not provide a CEO class to see if this could explain the difference in recall between the baseline and the CEO-version. The baseline does not use any external resource and is not dependent on the lexicon to map mentions to CEO classes. We found 3246 out-of-vocabulary cases that represent 12,999 mentions. Note that the event mentions are generated using ECB+ gold data and silver-data generated from the full text documents. We analyzed the most frequent of these mentions and did not observe any major gaps in the lexicon (an exception being *drunken driving* and *drunk driving* occur 8 and 9 times) that could explain the drop in recall.

We also abstracted from the assertions by only considering the property predicate. When ignoring the order (loose), we get 0.114P, 0.165R and 0.135F1. We thus see a slight drop in precision but higher recall and F1. Nevertheless, the difference is small and does not outweigh the value of using full assertions to connect events using circumstantial causal relations with specific implications for the involved participants.

To conclude: there is still substantial ground to cover in the CEO to increase the recall but the results for precision of the relations without using any further information on time and participants are promising. Especially, as the CEO appears to capture relations far beyond the context of sentences and even paragraphs.

6. Conclusion and Future Work

We have described our work on an event ontology that captures calamity events in newswire and the semantic circumstantial relations that hold between event classes, based on shared properties in the pre-, post- or during situations defined for each class.

First experiments and evaluations show that applying very basic heuristics to retrieve circumstantial relations based on assertions properties gives promising results with respect to precision. For increasing both recall and precision, adjustment and extension of the defined situation assertions will be needed as well as developing reasoner that can take into account the roles, property values and role instances to further scope the chaining of event instances.

Future work includes developing a reasoner and additional experiments on finding more sophisticated heuristics for salient circumstantial paths in the ontology. Further, we will evaluate the added value of our model extrinsically, by means of a QA task. For this, we are designing a Question-Answering task, where systems will have to provide answers to questions "why" a certain event has taken place rather than factoid questions by providing the most relevant and direct preceding event that can be seen as an explanation.

7. Acknowledgements

We thank Nynke Visscher and Joy van Wooning for their work and input on the annotation task.

The work presented in this paper was funded by the Netherlands Organization for Scientific Research (NWO) via the Spinoza grant, awarded to Piek Vossen in the project "Understanding Language by Machines".

8. Bibliographical References

Bartalesi Lenzi, V., Moretti, G., and Sprugnoli, R. (2012). Cat: the celct annotation tool. In *LREC*, pages 333–338.

Björkelund, A., Hafdell, L., and Nugues, P. (2009). Multilingual semantic role labeling. In *Proceedings of CoNLL-2009*, Boulder, CO, USA.

Brown, S., Bonial, C., Obrst, L., and Palmer, M. (2017). The rich event ontology. In *Proceedings of the Events and Stories in the News Workshop*, pages 87–97, Vancouver, Canada, August. Association for Computational Linguistics.

Caselli, T. and Morante, R. (to appear). Agreements and Disagreements in Temporal Processing: An Extensive Error Analysis of the TempEval-3 Systems. In *Proceedings of Language Resources and Evaluation Conference (LREC 2018)*.

Chambers, N. and Jurafsky, D. (2010). A database of narrative schemas. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2010)*.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Cybulska, A. and Vossen, P. (2014). Using a sledgehammer to crack a nut? lexical diversity and event coreference resolution. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2014)*, Reykjavik, Iceland, May 26-31.

De Beaugrande, R. and Dressler, W. (1981). *Textlinguistics*. New York: Longman.

de Lacalle, M. L., Laparra, E., Aldabe, I., and Rigau, G. (2016). A multilingual predicate matrix. In *Proceedings of Language Resources and Evaluation Conference (LREC 2016)*.

Fellbaum, C. (1998). *WordNet: an electronic lexical database*. MIT Press.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Niles, I. and Pease, A. (2001). Towards a standard upper ontology. In *Proceedings of FOIS-Volume 2001*. ACM.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). The penn discourse treebank 2.0 annotation manual.

Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C., and Scheffczyk, J. (2006). *FrameNet II: Extended Theory and Practice*. International Computer Science Institute, Berkeley, California.

Segers, R., Rospocher, M., Vossen, P., Laparra, E., Rigau, G., and Minard, A. (2016). The event and implied situation ontology: Application and evaluation. In *Proceedings of Language Resources and Evaluation Conference (LREC 2016)*.

Segers, R., Caselli, T., and Vossen, P. (2017). The circumstantial event ontology (ceo). In *Proceedings of the Events and Stories in the News Workshop*, pages 37–41, Vancouver, Canada, August. Association for Computational Linguistics.

Van den Akker, C., Aroyo, L., Cybulska, A., van Erp, M., Gorgels, P., Hollink, L., Jager, C., Legene, S., van der Meij, L., Oomen, J., van Ossenbruggen, J., Schreiber, G., Segers, R., Vossen, P., and Wielinga, B., (2010). *Historical Event-based Access to Museum Collections*, pages 1–9. CEUR-WS (online).

Vossen, P., Caselli, T., and Kontzopoulou, Y. (2015). Storylines for structuring massive streams of news. In *Proceedings of the 1st Workshop on Computing News StoryLines (CNewS 2015) at the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, Bejing, China.

9. Language Resource References

VV.AA. (2017). *WordNet*. unspecified, ISLRN 379-473-059-273-1.

Profiling Medical Journal Articles Using a Gene Ontology Semantic Tagger

Mahmoud El-Haj[1], **Paul Rayson**[1], **Scott Piao**[1] **and Jo Knight**[2]
[1]School of Computing and Communications and [2]Medical School, Lancaster University, UK
{m.el-haj, p.rayson, s.piao, jo.knight}@lancaster.ac.uk

Abstract

In many areas of academic publishing, there is an explosion of literature, and sub-division of fields into subfields, leading to stove-piping where sub-communities of expertise become disconnected from each other. This is especially true in the genetics literature over the last 10 years where researchers are no longer able to maintain knowledge of previously related areas. This paper extends several approaches based on natural language processing and corpus linguistics which allow us to examine corpora derived from bodies of genetics literature and will help to make comparisons and improve retrieval methods using domain knowledge via an existing gene ontology. We derived two open access medical journal corpora from PubMed related to psychiatric genetics and immune disorder genetics. We created a novel Gene Ontology Semantic Tagger (GOST) and lexicon to annotate the corpora and are then able to compare subsets of literature to understand the relative distributions of genetic terminology, thereby enabling researchers to make improved connections between them.

Keywords: semantic tagger, ontology, genetics, medical

1. Introduction

The explosion of scientific literature in all fields makes it hard to keep apace of new knowledge. This is particularly true in the relatively new field of genomics. For example, a search in the main citation database for biomedical literature (PubMed) for the term 'genome wide association study' results in just 5 papers from 1995, 141 from 2005 and 3,633 from 2015. We contend that the myriad of techniques developed in Information Retrieval coupled with Natural Language Processing can help address these scaling and searching issues. Such a set of techniques could help in a myriad of ways, for example, summarisation of papers or a set of papers, collocation methods to investigate drug-disease-gene interactions, and query expansion where terminology varies from one subfield to another. Previously such techniques have been used to perform tasks such as identifying gene-gene or gene-phenotype interactions (Bundschus et al., 2008; Kann, 2007). In addition, by using corpus comparison methods originating in Corpus Linguistics, we aim to identify key words and concepts emerging from a body of literature that will provide new clues to disease aetiology.

In the remainder of this paper, we describe related work on biomedical text mining and corpus comparison. Then we explain how we created an open access corpus derived from medical journal abstracts, and a novel semantic tagger to apply a lexicon derived from a standard Gene Ontology. Finally, we illustrate how these new resources allow us to profile medical journal articles using domain-specific ontologies.

2. Related Work

Over a number of years, Natural Language Processing (NLP) techniques have been widely applied to biomedical text mining to facilitate large-scale information extraction and knowledge discovery from the rapidly increasing body of biomedical literature. Substantial efforts have been dedicated to this research area. Among the early researchers in this area are Ananiadou et al. (2006), who identified the challenging issue of finding useful information from the plethora of biomedical scientific literature which are manually unmanageable. Kann (2007) also suggested that Text Mining approaches are essential for discovering information about disease and protein interactions buried within millions of biomedical records. Since the recognition of the importance of the Biomedical text mining, a variety of NLP tools have been developed and modified to support it. Among the main tools and corpora developed for such purposes include Genia tagger/corpus (Tsuruoka et al., 2005; Thompson et al., 2017), Termine[1], and LAPPS GRID (Ide et al., 2016). These tools have typically focused only on lexical, syntactic and shallow semantic (named-entity) approaches. Another related biomedical annotation tool is the Penn BioTagger[2] (Jin et al., 2006), which is capable of tagging gene entities, genomic variations entities and malignancy type entities. Despite the progress over the past years, there are still various issues which remain unsolved, including the lack of NLP tools tailored for specific subfields of biomedical research, and the need to link entities at the conceptual level. In this work, we report on our experiment in which we modify a semantic tagger and create a corpus semantically tagged with both related sub-sets of the Gene Ontology categories and generic semantic field categories for an aetiology study.

Comparing corpora is a key method in corpus linguistics, and is a vital step towards measuring the differences between collections of textual documents. Previous approaches have been focused on word level comparisons only, finding terms or keywords that can differentiate one corpus from the other (Kilgarriff, 2001; Rayson and Garside, 2000). When the method is applied at the semantic level (for example with the general purpose USAS taxonomy[3]), this enables confirmation of the word-level findings but also the ability to uncover key semantic categories, which are more dispersed across a wider group of words and would not otherwise be highlighted as key (Rayson, 2008). In a medical context, we hypothesise that it is impor-

[1]http://www.nactem.ac.uk/software/termine/
[2]http://seas.upenn.edu/∼strctlrn/BioTagger/BioTagger.html
[3]http://ucrel.lancs.ac.uk/usas/

tant to use a more fine-grained semantic taxonomy which embodies greater medical domain knowledge, hence our undertaking the research presented here which derives and applies a gene ontology semantic lexicon to this problem.

3. Dataset

We collected medical journal abstracts from PubMed[4] by restricting the search to retrieve only English medical articles discussing human genetics studies in psychiatry and immune related disorders. Table 1 shows the dataset statistics in terms of article and word counts. The searches have been adapted to ensure appropriate literature coverage. For example, whilst including `immun*` in the abstract search picks up papers on many diseases such as psoriasis, the same approach using the term `psych*` is not as effective. In our results, we directly compare the Immune and Psychiatric subcorpora only, but the Reference dataset statistics are included here to show the relative size of the two subcorpora. We will also be employing the Reference corpus in other experiments and to check vocabulary coverage of the existing semantic lexicon. We chose immune and psychiatric genetics corpora as examples that would be very different from each other allowing us to test the utility of the tools. The selected domains fall within the fourth author's research expertise and this has helped in appropriately interpreting the findings (Pouget et al., 2016).

The dataset was downloaded from PubMed in large XML file format[5]. We built a Java suite for parsing PubMed XML file format and extract abstracts along with other information such as journal titles, author names, publication date, DOI and so on. Our code is publicly available for research purposes.[6]

Table 1: Corpus Statistics

Corpus	#Articles	#Words	Keywords
Immune	21.5K	4.8M	(geneti* OR gene OR genot*) AND (immunol* OR immunog* OR immune)
Psychiatric	15.2K	2.8M	(geneti* OR gene OR genot*) AND (psychi)
Reference	296.5K	79.0M	(geneti* OR gene OR genot*)
Total	333.2K	86.7M	

4. Gene Ontology Semantic Tagger

For our initial experiments, the corpora were uploaded to Wmatrix[7] where we ran automatic part-of-speech tagging using CLAWS, semantic field tagging using USAS, and counted word frequencies and compared sub-corpora using the keywords method from corpus linguistics. We quickly realised that we needed to provide better coverage of the more fine-grained medical terminology in the PubMed corpora, and therefore included an extra level of annotation by tagging the corpora using The Gene Ontology Consortium's[8] OBO Basic Gene Ontology (go-basic.obo) categories[9].

The Gene Ontology (GO) project is a collaborative effort to address the need for consistent descriptions of gene products across databases. The *go-basic.obo* is the basic version of the GO ontology, filtered such that the graph is guaranteed to be acyclic paths, and annotations can be propagated up the graph. We focused on the *is_a* relation in order to trace ancestors and children for each entry in the ontology. We chose the *is_a* relationship in the first instance because it has a more intuitive meaning. Something is only considered *is_a* if an instance of the child process is an instance of the entire parent process.

To parse the OBO file we created Java code that combines the use of publicly available OBO library[10] with Java Directed Graph (Digraphs) to trace the paths from a node child to the root. The code used Breadth First and Depth First algorithms to quickly and accurately extract the paths. Figure 1 shows an example of a directed graph for the *basophil homeostasis* GO entry. The figure shows two paths starting from the child entry up to the biological process root.

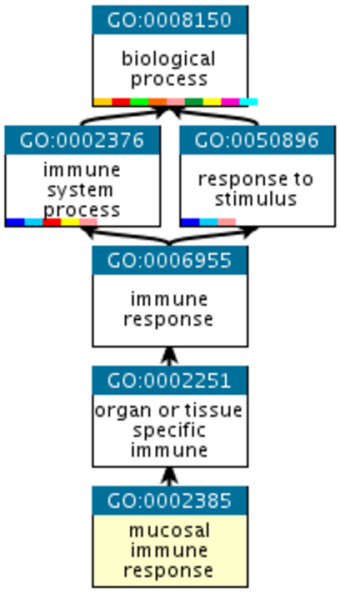


Figure 1: GO Directed Graph Sample

Our code allowed us to generate a USAS tagger dictionary file where each entry in the OBO ontology is tagged with the GO IDs shown in its path. Taking the "mucosal immune response" OBO entry shown in Figure 1 we can see there are two paths starting from the child node towards the "biological process" root. The dictionary creation process works as follows:

[4] https://www.ncbi.nlm.nih.gov/pubmed

[5] Instead of using PubMed API we searched PubMed website directly and exported the results to XML using PubMed "Send To File" service.

[6] https://github.com/drelhaj/BioTextMining

[7] http://ucrel.lancs.ac.uk/wmatrix/

[8] http://geneontology.org/

[9] http://purl.obolibrary.org/obo/go/go-basic.obo

[10] https://github.com/sugang/bioparser

1. determine whether the child node is single word or multi-word expression. The example shows the latter.
2. determine the number of paths towards the root.
3. get each path's GoID entries (child node's ancestors)
4. include the level of each ancestor by adding that to the end of each entry (e.g. .1 to refer to the first parent (GOO:0002251).
5. determine whether the path passes through an "immune system process", which is the one with GoID: 0002376. If so we add .I to the end of the GoID tag to refer to immune entry, otherwise we add .N referring to a non-immune entry.

Following the steps above, the child node GO:0002385 will be considered a multi-word expression entry and will have the following semantic dictionary tags:
`GO:0008150.4.I, GO:0002376.3.I, GO:0050896.3.N, GO:0006955.2.I, GO:0002385.0.I, GO:0002251.1.N, GO:0006955.2.N, GO:0002385.0.N, GO:0002251.1.I, GO:0008150.4.N.`
In the above dictionary, tags such as GO:0006955 will be extended with a `.2` suffix referring to level two (counting from level zero) and will appear twice; once as an immune entry with a `.I` suffix (GO:0006955.2.I) and another as a non-immune entry with a `.N` suffix (GO:0006955.2.N). While the GO directed graph snippet shown in figure 1 is relatively simple, figure 2 shows a much more complex example illustrating that the dictionary creation process can become more troublesome with overlapping hierarchies and levels that can be skipped for some graph traversals.
The resultant GO term and ID map collection from the process described above, which contains 433 single word bioterms and 44,180 multiword bioterms, has been merged into the Lancaster UCREL Semantic lexicons to create a new version of the Lancaster USAS semantic annotation system (Rayson et al., 2004; Piao et al., 2017), named GOST (Gene Ontology Semantic Tagger), in order to automatically annotate the bioterms with GO IDs in the journal articles, along with generic USAS semantic tags. Currently, using the GOST, we have tagged 237,615 PubMed abstracts in our corpus. This corpus provides a valuable new resource for mining Biomedical and health information from the Biomedical literature.
Table 2 shows a sample from a tagged abstract, where the part-of-speech tags are from CLAWS C7 tagset[11], the generic semantic tags are from the USAS tagset[12], the tags with leading code *GO* are from the Gene Ontology, and the MWE tags encode multiword term information including sequential number, term length and location of each word in the given term. As shown in the table, such a tagging can facilitate analysis of Gene information at any hierarchical levels of the Gene Ontology as researchers need. For example, researchers can

[11]http://ucrel.lancs.ac.uk/claws7tags.html
[12]http://ucrel.lancs.ac.uk/usas/

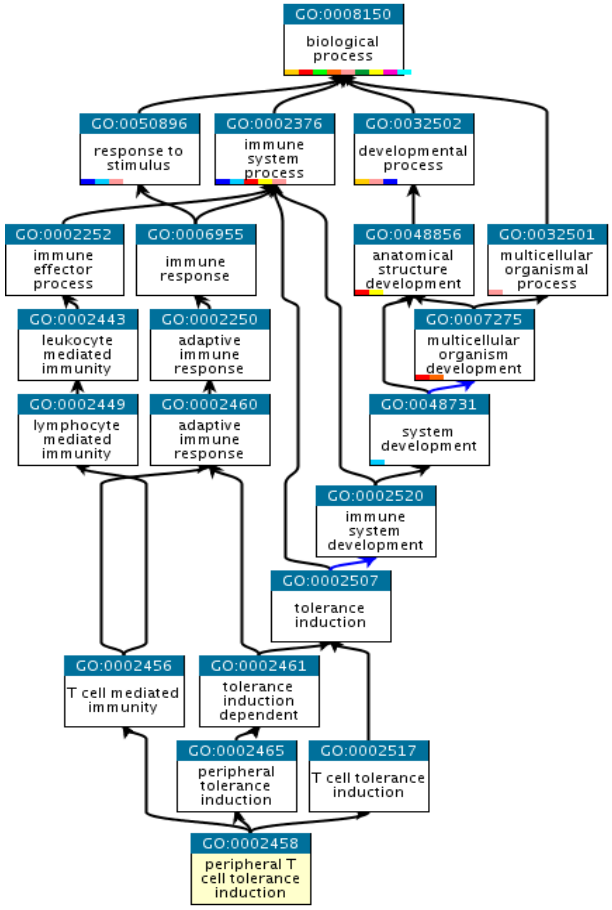


Figure 2: GO Directed Graph More Complex Sample

filter their analysis results by setting a range of hierarchical levels of [3-4], in which case only GO categories {GO:0008152.3.N, GO:0071704.3.N, GO:0008150.4.N, GO:0009987.3.N, GO:0006807.3.N, GO:0008152.4.N} would be considered for the term "cellular protein metabolic process" in Table 2.

5. Results

In our preliminary work using only a word level comparison (El-Haj et al., 2017), we uncovered many subject specific words have a much higher proportional representation in one corpus (e.g. schizophrenia). Other less predictable words such as "risk" are also found to be more frequent in psychiatric literature. The increased proportional representation suggests that language is used different despite both corpora describing genetic studies of a complex trait.
With the new GOST annotated corpora, we are able to compare the two corpora at the semantic level using the Gene Ontology concepts, see Table 3 for keyness sorted results. The final six columns show the actual and relative frequencies for the immune and psych sub-corpora, an indication of over- and under-use (by a direct comparison of the relative frequencies) and the log-likelihood keyness value. Many of the GO terms with the most significantly different frequencies between the two corpora are those strongly related to the suspected biological underpinning of the traits. For example "immune response", "immune system process" and "response to stimulus" were all more frequent in the im-

Table 2: Sample tagged text

WORD	LEMMA	POS	SEM	MWE
several	several	DA2	N5	0
processes	process	NN2	A1.1.1 X4.2	0
potentially	potentially	RR	A7+	0
involved	involved	JJ	A1.8+ A12-	0
in	in	II	Z5	0
MN	mn	FO	Z99	0
,	PUNC	YCOM	PUNC	0
including	including	II	A1.8+	0
extracellular	extracellular	JJ	GO:0022617.0.N GO:0016043.3.N GO:0044763.2.N GO:0043062.2.N GO:0030198.1.N GO:0016043.2.N GO:0008150.4.N GO:0044699.3.N GO:0022411.1.N GO:0044763.3.N GO:0044699.4.N GO:0071840.4.N GO:0071840.3.N GO:0022617.0.N GO:0009987.3.N GO:0008150.5.N GO:0009987.4.N	1:3:1
matrix	matrix	NN1	GO:0022617.0.N GO:0016043.3.N GO:0044763.2.N GO:0043062.2.N GO:0030198.1.N GO:0016043.2.N GO:0008150.4.N GO:0044699.3.N GO:0022411.1.N GO:0044763.3.N GO:0044699.4.N GO:0071840.4.N GO:0071840.3.N GO:0022617.0.N GO:0009987.3.N GO:0008150.5.N GO:0009987.4.N	1:3:2
disassembly	disassembly	RR	GO:0022617.0.N GO:0016043.3.N GO:0044763.2.N GO:0043062.2.N GO:0030198.1.N GO:0016043.2.N GO:0008150.4.N GO:0044699.3.N GO:0022411.1.N GO:0044763.3.N GO:0044699.4.N GO:0071840.4.N GO:0071840.3.N GO:0022617.0.N GO:0009987.3.N GO:0008150.5.N GO:0009987.4.N	1:3:3
and	and	CC	Z5	0
organization	organization	NN1	S5+c S7.1+	0
,	PUNC	YCOM	PUNC	0
cell	cell	NN1	GO:0007155.0.N GO:0022610.1.N GO:0008150.2.N	2:2:1
adhesion	adhesion	NN1	GO:0007155.0.N GO:0022610.1.N GO:0008150.2.N	2:2:2
,	PUNC	YCOM	PUNC	0
cell-cell	cell-cell	JJ	Z99	0
signaling	signaling	NN1	GO:0023052.0.N GO:0008150.1.N	0
,	PUNC	YCOM	PUNC	0
cellular	cellular	JJ	GO:0008152.3.N GO:0019538.1.N GO:1901564.2.N GO:0071704.3.N GO:0044267.0.N GO:0008150.4.N GO:0044260.1.N GO:0044237.2.N GO:0043170.2.N GO:0044238.2.N GO:0009987.3.N GO:0006807.3.N GO:0008150.5.N GO:0008152.4.N	3:4:1
protein	protein	NN1	GO:0008152.3.N GO:0019538.1.N GO:1901564.2.N GO:0071704.3.N GO:0044267.0.N GO:0008150.4.N GO:0044260.1.N GO:0044237.2.N GO:0043170.2.N GO:0044238.2.N GO:0009987.3.N GO:0006807.3.N GO:0008150.5.N GO:0008152.4.N	3:4:2
metabolic	metabolic	JJ	GO:0008152.3.N GO:0019538.1.N GO:1901564.2.N GO:0071704.3.N GO:0044267.0.N GO:0008150.4.N GO:0044260.1.N GO:0044237.2.N GO:0043170.2.N GO:0044238.2.N GO:0009987.3.N GO:0006807.3.N GO:0008150.5.N GO:0008152.4.N	3:4:3
process	process	NN1	GO:0008152.3.N GO:0019538.1.N GO:1901564.2.N GO:0071704.3.N GO:0044267.0.N GO:0008150.4.N GO:0044260.1.N GO:0044237.2.N GO:0043170.2.N GO:0044238.2.N GO:0009987.3.N GO:0006807.3.N GO:0008150.5.N GO:0008152.4.N	3:4:4
,	PUNC	YCOM	PUNC	0

Table 3: Gene Ontology Semantic Keyness Results

GOID	Name	Immune	%	Psych	%	O/U	Keyness
GO:0005623	cell	33346	7.31	1524	1.02	+	10696.95
GO:0005575	Cellular Component	34577	7.58	1808	1.20	+	10332.02
GO:0007610	behavior	199	0.04	2095	1.40	-	4611.01
GO:0032501	multicellular organismal process	616	0.13	2364	1.57	-	3915.62
GO:0002376	immune system process	7253	1.59	88	0.06	+	3416.63
GO:0008150	Biological Process	7253	1.59	88	0.06	+	3416.63
GO:0006955	immune response	6992	1.53	84	0.06	+	3298.74
GO:0006955	immune response	6992	1.53	84	0.06	+	3298.74
GO:0050877	neurological system process	426	0.09	1756	1.17	-	2991.92
GO:0050896	response to stimulus	7034	1.54	192	0.13	+	2764.12
GO:0002376	immune system process	2958	0.65	28	0.02	+	1443.03
GO:0008150	Biological Process	2933	0.64	28	0.02	+	1429.29
GO:0050890	cognition	10	0.00	536	0.36	-	1402.85
GO:0050877	neurological system process	16	0.00	548	0.37	-	1394.05
GO:0005575	Cellular Component	5013	1.10	308	0.21	+	1357.84

mune disorder related corpus. The following terms were more frequent in the psychiatric corpus ("neurological system process" and "cognition"). Some of these terms are expected, and help to confirm that our methodology is successful and some categories offer routes for further investigation. We have therefore proved that in principal the method is working and we will continue to mine the results for biological insight.

6. Conclusion and Future Work

In this paper, we have illustrated our early explorations into extending corpus and computational linguistics methods to permit genomics researchers to explore their rapidly growing literature in new ways. Our main contributions are the corpus-based explorations of the research literature on human genetics studies, a method for the creation of a semantic lexicon from an existing Gene Ontology, a Gene Ontology Semantic Tagger (GOST) to apply this to corpora of scientific papers, and freely available annotated corpora. In terms of future work, we will further investigate how our new fine-grained taxonomy performs in terms of contextual accuracy, and whether the level of detail introduced is too much for our planned application. We have already investigated this type of fine-grained task in research related to historical contexts with the Historical Thesaurus Semantic Tagger (Piao et al., 2017). Here, there may need to be a compromise between levels of accuracy and domain-specific explainability. We also intend to carry out a corpus-based investigation into variability of GO terms which may not be replicated exactly in the corpora, for example inflectional or derivational suffixes such as "processes" instead of "process", and the potential for intervening items within multiword expressions. This will allow us to increase the

tagger's accuracy as well as potentially offering a semi-automatic route for updating GO itself. The corpora and Java code to parse and annotate the dataset in addition to the ontology lexicon are made publicly available for research purposes.[13] The Gene Ontology Semantic Tagger has also been released via the downloadable graphical interface[14].

7. Acknowledgements

We are grateful for support for this research from a Wellcome Trust Seed funding grant (reference 204475/Z/16/Z). For more details about our research, see the project website: http://wp.lancs.ac.uk/btm/

8. Bibliographical References

Ananiadou, S., Kell, D. B., and Tsujii, J. (2006). Text mining and its potential applications in systems biology. *Trends in Biotechnology*, 24(12):571–579.

Bundschus, M., Dejori, M., Stetter, M., Tresp, V., and Kriegel, H.-P. (2008). Extraction of semantic biomedical relations from text using conditional random fields. *BMC Bioinformatics*, 9(1):207, Apr.

El-Haj, M., Piao, S., Rayson, P., and Knight, J. (2017). A comparison between genetics papers relating to immune disorders and psychiatric disorders. In *Proceedings of The 2017 Annual Meeting of the International Genetic Epidemiology Society*.

Ide, N., Suderman, K., Pustejovsky, J., Verhagen, M., and Cieri, C. (2016). The language application grid and galaxy. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Jin, Y., McDonald, R. T., Lerman, K., Mandel, M. A., Carroll, S., Liberman, M. Y., Pereira, F. C., Winters, R. S., and White, P. S. (2006). Automated recognition of malignancy mentions in biomedical literature. *BMC Bioinformatics*, 7(492).

Kann, M. G. (2007). Protein interactions and disease: computational approaches to uncover the etiology of diseases. *Briefings in Bioinformatics*, 8(5):333–346.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics*, 6(1):97–133.

Piao, S., Dallachy, F., Baron, A., Demmen, J., Wattam, S., Durkin, P., McCracken, J., Rayson, P., and Alexander, M. (2017). Time-sensitive historical thesaurus-based semantic tagger for deep semantic annotation. *Computer Speech and Language*, 46:113–135.

Pouget, J. G., GonÃ$\mathbf{S}_a$lves, V. F., , Spain, S. L., Finucane, H. K., Raychaudhuri, S., Kennedy, J. L., and Knight, J. (2016). Genome-wide association studies suggest limited immune gene enrichment in schizophrenia compared to 5 autoimmune diseases. *Schizophrenia Bulletin*, 42(5):1176–1184.

Rayson, P. and Garside, R. (2000). Comparing corpora using frequency profiling. In *Proceedings of the Workshop on Comparing Corpora - Volume 9*, WCC'00, pages 1–6, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rayson, P., Archer, D., Piao, S., and McEnery, A. (2004). The ucrel semantic analysis system. In *Proceedings of the beyond named entity recognition semantic labelling for NLP tasks workshop*, pages 7–12.

Rayson, P. (2008). From key words to key semantic domains. *International Journal of Corpus Linguistics*, 13(4):519–549.

Thompson, P., Ananiadou, S., and Tsujii, J., (2017). *Handbook of Linguistic Annotation*, chapter The GENIA Corpus: Annotation Levels and Applications. Springer, Dordrecht, Netherlands.

Tsuruoka, Y., Tateishi, Y., Kim, J.-D., Ohta, T., McNaught, J., Ananiadou, S., and Tsujii, J. (2005). Developing a robust part-of-speech tagger for biomedical text. In *Advances in Informatics - 10th Panhellenic Conference on Informatics (PCI 2005), LNCS 3746*, pages 382–392.

[13] https://github.com/drelhaj/BioTextMining

[14] http://ucrel.lancs.ac.uk/usas/gui/

Towards A Conversation-Analytic Taxonomy of Speech Overlap

Felix Gervits and Matthias Scheutz
Human-Robot Interaction Laboratory, Tufts University
200 Boston Ave. Medford, MA 02155
{felix.gervits, matthias.scheutz}@tufts.edu

Abstract

We present a taxonomy for classifying speech overlap in natural language dialogue. The scheme classifies overlap on the basis of several features, including onset point, local dialogue history, and management behavior. We describe the various dimensions of this scheme and show how it was applied to a corpus of remote, collaborative dialogue. Moving forward, this will serve as the basis for a computational model of speech overlap, and for use in artificial agents that interact with humans in social settings.

Keywords: overlap, turn-taking, conversation analysis

1. Introduction

1.1. Background

Speech overlap is a common phenomenon found in human dialogue across the world (Schegloff, 2000). Overlap is not the same as interruption, as the former is considered to be a product of turn-taking organization while the latter a violation of conversational norms (Drew, 2009; Drummond, 1989). Much of the past work on speech overlap comes from the field of Conversation Analysis (CA). CA emphasizes talk-in-interaction and aims to study the ways in which social interaction is managed by the participants through dialogue. At the heart of CA is the model proposed by Sacks et al. (1974) (hereafter referred to as SSJ) which elegantly describes the turn-taking organization at the core of human social interaction. The SSJ model makes some important predictions about the structure of turn-taking and how it relates to speech overlap. The most important of these predictions is that while speakers exchange turns in the course of a typical dialogue, they tend to follow the "one-speaker-at-a-time" rule. Another prediction is that speaker changes occur with minimal gap or overlap, and when overlap does occur, it is usually resolved very quickly.

1.2. Speech Overlap

Much of our understanding of the structure of overlap comes from the work of Gail Jefferson (Jefferson, 1982; Jefferson, 1986; Jefferson, 2004). Jefferson showed that overlap is an orderly process characterized by precise onset times with regards to turn-taking structure. She identified several types of overlap based on these onset points (described in Section 3.2.1 below), and found that they generally coincide with the point in a turn at which a speaker change can occur. This suggests that overlap is a consequence of peoples' adherence to the one-speaker rule and the goal to minimize gaps in between turns. On this account, overlap in collaborative dialogue is not seen as rude, but rather supportive. It indicates that conversational partners are receptive to one another and attempt to make smooth and efficient turn transitions - all predictions of the SSJ model.

In addition to characterizing overlap onset, there has also been some work on understanding how people manage and recover from overlap. Jefferson (2004) addresses some aspects of overlap management by describing ways in which people can drop out or hold the turn during overlap, as well as how people deal with segments of speech in overlap that were not heard. In terms of overlap recovery, Schegloff (2000) describes an "overlap resolution device" used by participants in an interaction to recover from overlapping speech. Because of the focus on recovery from overlap, Schegloff limits his analysis to *competitive overlap*, in which there is an explicit claim for the floor that needs to be resolved. As a result, he excludes certain types of *non-competitive overlap* from his analysis, including: terminal overlap, continuers, conditional access to turn (e.g., word search), and "chordal" cases (e.g., laughter). However, these cases are very common dialogue phenomena, and should be included in any thorough account of overlap.

1.3. Present Work

The present work seeks to develop a comprehensive taxonomy of speech overlap that incorporates past work from CA as well as our own contributions. While several aspects of overlap have been studied independently, there does not exist an overall scheme that captures all the critical dimensions for classifying overlap. We seek to develop such a scheme, utilizing methods from CA as well as discourse analysis in order to balance ecological validity with experimental control (De Ruiter and Albert, 2017).

Importantly, the features in our scheme are quantifiable and computationally tractable so as to be useful not only as an explanation of empirical data but also in computational models. This necessitates an important trade off between empirical validity and computational tractability. Computational models and dialogue systems that utilize this framework must operate at the millisecond time scale, so the number of features to consider must be kept to a minimum. As a result, cues from the visual modality (e.g., gaze) are absent from our scheme due to computational complexity, even though they have been identified as important for managing turn-taking in face-to-face interactions.

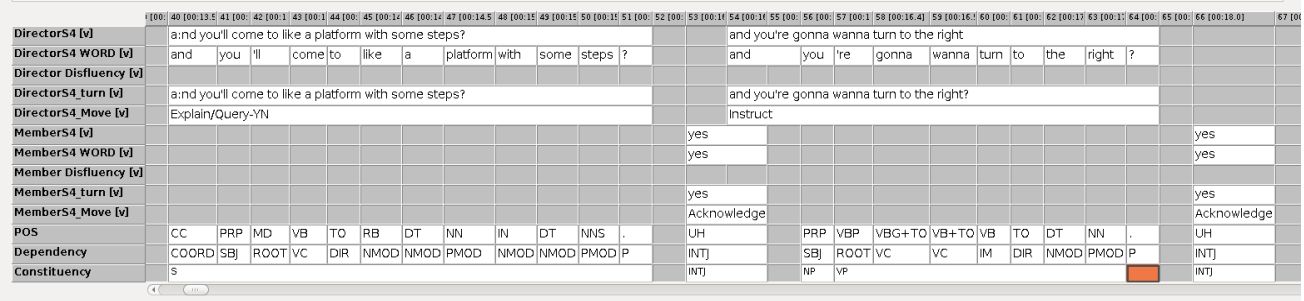


Figure 1: Corpus annotation in the EXMARaLDA Partitur Editor

2. Corpus

We used annotations of the Cooperative Remote Search Task (CReST) corpus (Eberhard et al., 2010) to develop our taxonomy. The corpus contains about 8 minutes of unscripted task-oriented dialogue from each of 10 dyads that performed the task (2712 utterances and 15194 words). The corpus was annotated using the EXMARaLDA Partitur Editor (Schmidt, 2001), and includes the following features: utterances, words, syntactic structure, part of speech, disfluencies, conversational moves, and turns.

The collaborative task at the heart of the corpus (described in more detail in Gervits et al. (2016b) involves two human teammates, a director and searcher, performing a joint search task. The two teammates communicate through remote headset and must achieve a variety of goals within a limited time of 8 minutes.

Speech overlap was relatively frequent in the corpus due to the remote communication, time pressure, and interaction demands imposed by the task. We extracted all instances of overlapping speech in the corpus and classified them according to our scheme.

3. Taxonomy of Speech Overlap

3.1. Definitions

It is important to define a few terms that readers may be unfamiliar with before moving forward. According to the SSJ model the main unit of dialogue is the turn-construction unit, or TCU. A TCU can be a word, clause, phrase, or sentence, and it represents a turn-at-talk. In between (and within) TCUs are points at which speaker change may occur - these are know as transition-relevance places, or TRPs. The TRP signifies a point of completion (grammatical, prosodic, or pragmatic) of the TCU, and is the point at which the next speaker *may* take the turn. When discussing overlap, we use the terms *first starter* and *second starter* to denote the order in which speakers initiated speech.

Another important term to define is the conversational beat. Schegloff (2000) defines a conversational beat as roughly equivalent to the average length of a syllable in spontaneous speech. This corresponds to the average gap time (silence) between speakers' turns, and has been estimated to be between 80-180 ms (Wilson and Wilson, 2005; Wilson and Zimmerman, 1986). Since this varies depending on rate of speech, we use the upper bound of 180 ms as one conversational beat.

3.2. Categories

In order to classify overlap according to our scheme, we define the following categories. These will be discussed below with example dialogues from the corpus.

Onset Point
- Transition-Space, Post-Transition, Interjacent, Last-Item

Local Dialogue History
- *Turn-Holder:*
  - Previous, Current, Next
- *Dialogue Move:*
  - Initiation, Response, Ready

Overlap Management

Non-Competitive
- Drop Turn, Single Item, Wrap Up, Finish Turn, Laughter

Competitive
- *Continue*
- *Disfluency:*
  - Prolongation, Silent Pause, Filled Pause, Combination
- *Self-Repair:*
  - Repetition, Substitution Insertion, Deletion

3.2.1. Onset Point

Perhaps the most important feature to classify overlap is the onset point at which it occurs. Our scheme includes the following types based on Jefferson (1986):

A *last-item* overlap occurs at the point immediately before a TRP (see D1[1]). They typically involve an overlap on the last word, but could also occur at the last lexical item ("cardboard box", "phone number", etc.). Sometimes a person will attempt to come in at the last-item position, but the first starter will continue their turn after the TRP. These are still treated as last-item overlaps since the second starter's entrance was at the perceived last-item position (see D2).

D1) *D: There is . one yellow block . per blue b[ox*
S: [ok]ay

[1]In the dialogue examples, brackets indicate overlap, colons indicate prolonged syllables, hyphens indicate repaired segments, periods indicate brief silent pauses of one beat, and longer pauses are indicated in parentheses.

D2) *S: There is an open do[or to my rig]ht*
D: [per:::fect p]erfect

A *transition-space* overlap (i.e., simultaneous startup) occurs in the transition space between TCUs when the previous speaker continues their prior turn at the same time that the other speaker started their new turn (see D3). The startup can be simultaneous or offset within up to one conversational beat (up to 180 ms). One special case here is when a speaker prolongs the last item of a TCU and the second starter comes in at this point. Instead of being marked as an last-item, this would actually be a transition-space since the speaker was aiming for the TRP (see D4).

D3) *S: Yes*
(0.5)
D: [So is]-
S: A[n d I] just leave that there correct?

D4) *D: Ye : :[: :s*
S: [o k]a y

A *post-transition* overlap occurs when a speaker starts their turn slightly after the current speaker started a new TCU (i.e., after the transition space). "Slightly after" is defined as within 1-2 conversational beats (180-360 ms) of the first starter. It is distinct from the transition-space overlap in that one speaker has already laid claim to the turn. This type of overlap usually occurs when the second starter refers to something that the first starter said in their previous TCU (see D5 - the TRP is between "sure" and "where").

D5) *S: Is there a time limit?*
D: I'm- I'm not sure whe[re are you?]
S: [o k a y]

An *interjacent* overlap occurs in the middle of a turn, not directly near a TRP (see D6). Thus, any overlap that does not fall within the 2-beat window of a transition-space/post-transition or on the last-item of a speaker's turn can be classified as interjacent. While these are closest to "interruption", in practice these types of interjections are usually what are known as *recognitional* overlaps. They occur when speakers seek to correct, clarify, or otherwise respond to something that the first starter said. Continuers or other acknowledgments can also occur at the interjacent point, but it is more common that they occur near TRPs (Duncan, 1972).

D6) *D: Okay maybe that was a-*
(0.5)
D: like they said th[e r e w a s]- [okay
S: [it was a pin]k b[ox

3.2.2. Local Dialogue History

Local dialogue history is a crucial element of our scheme, as people appear to be sensitive to this information when resolving overlap of different types(Schegloff, 2000). For example, if a speaker asks a question and then overlaps the recipient as s/he is responding, then the recipient is likely to drop out. This is because the first speaker violated the adjacency pair, thus creating an implicature that an expansion or clarification of the initial question will occur. Knowledge of the previous turn-holder and dialogue move is necessary to identify this type of behavior. Another common pattern is when a person drops out of competitive overlap only to restart exactly what they were attempting to say at the next available opportunity. Knowledge of the local dialogue history is necessary to classify these examples.

Overlap onset point alone is not sufficient to account for such cases, so our scheme includes information about the previous, current, and next turn-holder, as well as the corresponding dialogue moves with respect to the overall sequence organization. Dialogue move classification is based on the annotation scheme from Carletta et al. (1997), which codes dialogue moves as types of *Initiation, Response*, and *Ready* moves. Expanded Acknowledgment categories are from Eberhard et al. (2010). While the other features are straightforward to code, *current turn holder* requires slightly more consideration in cases of overlap. For interjacent and last-item overlaps, we mark the first starter as the current turn-holder since they already had a turn in progress. For post-transition overlaps, we similarly mark the first starter as current turn-holder because they have made a perceivable sound ($>$ 1 beat) to claim the turn. In transition-space overlaps, however, current turn-holder is set to "both", as both speakers have laid claim to the turn simultaneously.

3.2.3. Overlap Management

Jefferson (2004) describes the following types of general behaviors that can occur to manage overlap, and return the dialogue back to a single speaker: First starter drops - second starter begins; Second starter drops out after false start; Both parties continue simultaneously; both parties drop out simultaneously.

We expand on this preliminary scheme to capture both competitive and non-competitive overlap management behaviors. Though we use the term *competitive* to describe a fight for the turn, it is important to note that such "fights" are very brief and are typically not contentious (barring political debates). They are used as a means to quickly establish who will take the next turn. In our scheme, these behaviors are only considered as overlap management mechanisms if they occurred within two beats of the end of the overlap (following Schegloff (2000)).

The non-competitive categories denote ways in which people come in during overlap with no attempt to take (if second starter) or hold (if first starter) the turn. One such type is *Single Item*, in which the speaker utters a single word (or lexical item) TCU in overlap. Oftentimes these are continuers such as "okay", "right", etc. Another type is *Wrap Up*, which we define as finishing up a turn when overlap is detected. The first starter continues their turn just enough to get to the next TRP, and then allows the second starter to take the floor. This is in contrast to *Finish Turn* which involves completing the remaining item and relinquishing the turn, as in last-item overlap. *Drop Turn* is when a speaker drops out before a TRP, abandoning their utterance. *Laughter* is the final category here. It is a non-competitive activity that typically elicits a similar response from the recipient.

The competitive overlap management categories include several behaviors which participants use to take or maintain the turn during overlap. One such category includes disfluencies such as *prolongations* ($>$ 180 ms/syllable), *silent pauses*, and *non-lexical filled pauses* (uh/um). *Combina-*

Table 1: Frequency of overlap onset points.

Overlap onset	Frequency
Transition-Space	35%
Post-Transition	15%
Interjacent	15%
Last-Item	35%

Table 2: Frequency of overlap management behaviors for asynchronous onset cases (post-transition, interjacent, last-item).

Overlap Management (asynchronous)	First Starter	Second starter
Non-Competitive	**38%**	**26%**
Drop Turn	2.6%	1.5%
Single Item	8.8%	21%
Wrap Up	3.4%	1.6%
Finish Turn	20%	<1%
Laughter	3.4%	1.5%
Competitive	**12%**	**24%**
Continue	6.5%	16%
-Disfluency-	4.3%	4.7%
Prolongation	1.9%	2.6%
Silent Pause	2%	<1%
Filled Pause	<1%	<1%
Combination	<1%	<1%
-Self-Repair-	1.1%	3%
Repetition	<1%	1.8%
Substitution	<1%	<1%
Insertion	0%	<1%
Deletion	0%	<1%

tions of the above behaviors can occur, such as a filled pause followed by a silent pause. The other category includes self-repairs from the HCRC map task coding scheme - repetitions, substitutions, insertions, and deletions (Lickley, 1998). *Repetitions* can be a restart from the beginning of the turn, or can involve repeated syllables or fragments, as in "recycled turn beginnings" (Schegloff, 1987). *Substitutions* occur when a word/item is replaced in the TCU, and *Insertions* occur when a new word/item is added. *Deletions* are abandoned utterances followed by a restart. If a disfluency (pause or prolongation) occurs within a self-repair then the self-repair is given priority for purposes of annotation. This is done to resolve ambiguity in coding overly complex repair combinations which sometimes arise. Finally, the scheme includes a category called *Continue* which indicates that the speaker continued to talk through overlap with no disfluent behavior. They also did not stop at the next TRP, as in the *Wrap Up* case.

4. Corpus Annotation Results

4.1. Summary and Interpretation of Results

As a demonstration of the present scheme, we extracted the above categories from the annotated CReST corpus. While a complete analysis of the corpus is a work in progress, here we report on some observed frequencies in the data.

Table 3: Frequency of overlap management behaviors for synchronous onset cases (transition-space).

Overlap Management (synchronous)	Frequency
Non-Competitive	**55%**
Drop Turn	7%
Single Item	39%
Wrap Up	5.8%
Finish Turn	<1%
Laughter	2.1%
Competitive	**45%**
Continue	24%
-Disfluency-	15.6%
Prolongation	9.6%
Silent Pause	3%
Filled Pause	1.2%
Combination	1.8%
-Self-Repair-	5.8%
Repetition	3%
Substitution	1.2%
Insertion	<1%
Deletion	1.5%

There were a total of 541 overlaps in the 10 teams we analyzed. Table 1 shows the frequency of each type of overlap based on the onset point. Transition-space and last-item overlaps accounted for 70% of all overlap in the corpus. While this may seem surprising given that these overlaps have the smallest window of classification (less than a beat in most cases), this finding highlights the orderly nature of overlap.

We also looked at overlap management behaviors from our scheme. There were 1082 cases here (twice the number of overlaps) because we tracked both speakers' responses. Table 2 shows the distribution of behaviors for post-transition, interjacent and last-item overlaps (total: 741), while Table 3 shows the distribution of transition-space overlaps (total: 341). The data were divided in this way because transition-space overlaps involve a synchronous startup of both speakers, and do not have a first or second starter.

Overall, there was a numerically higher rate of non-competitive overlap in all cases. This supports the SSJ model in that most overlap is not competitive and is resolved quickly. For asynchronous cases, the most frequent behavior for first starters was *Finish Turn* (20%), with most of these coming from last-item onsets. For second starters, the most frequent behavior was *Single Item* (21%). Interjacent overlaps had a high proportion of these, which suggests that they served as verbal acknowledgments, i.e., "continuers". A total of 33% of turns in which a speaker was overlapped at the interjacent point contained a *Single Item* utterance by the second starter. *Continues* were also frequent for second starters (16%), and often occurred at last-item positions. Despite being classified as competitive, when *Continues* occur at the last-item point there is often no competition for the turn; the first starter typically finishes the turn immediately. For the synchronous onset cases, *Single Items* were by far the most common (39%), and often involved both speakers producing them simulta-

neously (e.g., "OK", "OK") in the transition space between turns. *Continues* (24%) and *Disfluencies* (15.6%) were also relatively common, and here they were often used to hold the floor. This suggests that transition-space overlaps may have led to more competition for the floor than the other types. This is not surprising given the fast paced nature of the task and the fact that teammates could not rely on visual cues to predict turn completion.

4.2. Future Work

As we move forward, we will evaluate our taxonomy on additional corpora. The challenge with using traditional data sets is that they often lack the kind of fine-grained turn annotations (e.g., TCU/TRP) necessary to apply our scheme. Corpora that do have turn annotations are typically from natural open-ended interactions (with no particular task), and so may not inform behavior in the kinds of task-oriented settings that are of interest to us. To address these concerns, we will construct a new corpus of task-oriented dialogue with annotations of various turn-taking features.

The long-term goal of this work is to extract the taxonomic features automatically from a dataset. This automated extraction will be a necessary step towards the development of a computational model of speech overlap, as these features will need to be identified in real time. Such a model will be useful both as a theoretical testbed and also in dialogue systems in order to improve communication and make interaction more natural. This latter goal will allow artificial agents to interpret and utilize disfluent segments of dialogue (such as those used to manage overlap) to improve coordination and to serve as better teammates (Gervits et al., 2016a; Gervits, 2017).

5. Conclusion

We have introduced a novel taxonomy of speech overlap that extends prior work in CA and discourse analysis. The scheme classifies overlap on the basis of onset point, local dialogue history, and management behavior. We applied our scheme to a corpus of collaborative, task-oriented dialogue and reported the distribution of the various features of interest. Moving forward, we plan to implement a computational model based on our scheme, with the goal of identifying a minimum set of computationally tractable features that can be used for real-time overlap classification in dialogue systems and artificial agents.

6. Acknowledgements

This work was in part funded by a NASA Space Technology Research Fellowship under award 80NSSC17K0184.

7. Bibliographical References

Carletta, J., Isard, S., Doherty-Sneddon, G., Isard, A., Kowtko, J. C., and Anderson, A. H. (1997). The reliability of a dialogue structure coding scheme. *Computational linguistics*, 23(1):13–31.

De Ruiter, J. and Albert, S. (2017). An appeal for a methodological fusion of conversation analysis and experimental psychology. *Research on Language and Social Interaction*, 50(1):90–107.

Drew, P. (2009). Quit talking while I'm interrupting: a comparison between positions of overlap onset in conversation. In *Talk in Interaction: Comparative Dimensions*, pages 70–93, 01.

Drummond, K. (1989). A backward glance at interruptions. *Western Journal of Communication (includes Communication Reports)*, 53(2):150–166.

Duncan, S. (1972). Some signals and rules for taking speaking turns in conversations. *Journal of personality and social psychology*, 23(2):283.

Eberhard, K. M., Nicholson, H., Kübler, S., Gundersen, S., and Scheutz, M. (2010). The indiana "cooperative remote search task"(crest) corpus. In *Proceedings of LREC 2010*.

Gervits, F., Eberhard, K., and Scheutz, M. (2016a). Disfluent but effective? a quantitative study of disfluencies and conversational moves in team discourse. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3359–3369.

Gervits, F., Eberhard, K., and Scheutz, M. (2016b). Team communication as a collaborative process. *Frontiers in Robotics and AI*, 3:62.

Gervits, F. (2017). Disfluency handling for robot teammates. In *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 341–342. ACM.

Jefferson, G. (1982). Two explorations of the organization of overlapping talk in conversation. In *Tilburg Papers in Language and Literature 28*. University of Tilburg.

Jefferson, G. (1986). Notes on 'latency' in overlap onset. *Human Studies*, 9(2-3):153–183.

Jefferson, G. (2004). A sketch of some orderly aspects of overlap in natural conversation. *Pragmatics and Beyond New Series*, 125:43–62.

Lickley, R. J. (1998). HCRC disfluency coding manual. In *Technical Report HCRC/TR-100*. Human Communication Research Centre, University of Edinburgh.

Sacks, H., Schegloff, E. A., and Jefferson, G. (1974). A simplest systematics for the organization of turn-taking for conversation. *language*, pages 696–735.

Schegloff, E. A. (1987). Recycled turn beginnings: A precise repair mechanism in conversation's turn-taking organization. *Talk and social organisation*, 1:70–85.

Schegloff, E. A. (2000). Overlapping talk and the organization of turn-taking for conversation. *Language in society*, 29(1):1–63.

Schmidt, T. (2001). The transcription system exmaralda: An application of the annotation graph formalism as the basis of a database of multilingual spoken discourse. In *Proceedings of the IRCS Workshop On Linguistic Databases*, pages 219–227.

Wilson, M. and Wilson, T. P. (2005). An oscillator model of the timing of turn-taking. *Psychonomic bulletin & review*, 12(6):957–968.

Wilson, T. P. and Zimmerman, D. H. (1986). The structure of silence between turns in two-party conversation. *Discourse Processes*, 9(4):375–390.

Indian Language Wordnets and their Linkages with Princeton WordNet

Diptesh Kanojia[1,2,3], **Kevin Patel**[1], **Pushpak Bhattacharyya**[1]
[1]IIT Bombay, [2]Monash University,
[3]IITB-Monash Research Academy,
{diptesh, kevin.patel, pb}@cse.iitb.ac.in

Abstract

Wordnets are rich lexico-semantic resources. Linked wordnets are extensions of wordnets, which link similar concepts in wordnets of different languages. Such resources are extremely useful in many Natural Language Processing (NLP) applications, primarily those based on knowledge-based approaches. In such approaches, these resources are considered as gold standard/oracle. Thus, it is crucial that these resources hold correct information. Thereby, they are created by human experts. However, human experts in multiple languages are hard to come by. Thus, the community would benefit from sharing of such manually created resources. In this paper, we release mappings of 18 Indian language wordnets linked with Princeton WordNet. We believe that availability of such resources will have a direct impact on the progress in NLP for these languages.

1. Introduction

Wordnets (Fellbaum, 1998) have been useful in different Natural Language Processing applications such as Word Sense Disambiguation (TufiŞ et al., 2004; Sinha et al., 2006), Machine Translation (Knight and Luk, 1994) *etc.*

Linked Wordnets are extensions of wordnets. In addition to language-specific information captured in constituent wordnets, linked wordnets have a notion of an interlingual index, which connects similar concepts in different languages. Such linked wordnets have found their application in machine translation (Hovy, 1998), cross-lingual information retrieval (Gonzalo et al., 1998), *etc.*

Given the extensive application of wordnets in different NLP applications, creation and maintenance of wordnets involve expert involvement. Such involvement is costly both in terms of time and resources. This is further amplified in case of linked wordnets, where experts need to have knowledge of multiple languages.

India is a vast country with massive language diversity. According to a census in 2001, there are 122 major languages [1], out of which, 29 have more than a million native speakers. The IndoWordNet project contains wordnets of 18 of these languages. These wordnets were created using expansion approach with Hindi Wordnet as the pivot.

This paper makes the following contributions:

- We release the latest version of 18 wordnets under the IndoWordNet project as a single bundle[2].
- Using mappings between Princeton WordNet and Hindi wordnet, we create and release mappings between Princeton WordNet and these 18 languages wordnet.

The rest of the paper is organized as follows: Section 2. covers some background and related work needed for further discussions. Section 3. describes the released resources. Section 4. discusses different issues encountered in the creation of these datasets, followed by the conclusion and future work.

[1]http://en.wikipedia.org/wiki/Languages_of_India
[2]http://www.cfilt.iitb.ac.in/ilw

2. Background and Related Work

Princeton WordNet or the English WordNet was the first wordnet and inspired the development of many other wordnets. EuroWordNet (Vossen and others, 1997) is a linked wordnet comprising of wordnets for European languages, *viz*, Dutch, Italian, Spanish, German, French, Czech and Estonian. Each of these wordnets is structured in the same way as the Princeton WordNet for English (Miller et al., 1990) - synsets (sets of synonymous words) and semantic relations between them. Each wordnet separately captures a language-specific information. In addition, the wordnets are linked to an Inter-Lingual-Index, which uses Princeton WordNet as a base. This index enables one to go from concepts in one language to similar concepts in any other language. Such features make this resource helpful in cross-lingual NLP applications.

IndoWordNet (Bhattacharyya, 2010) is a linked wordnet comprising of wordnets for major Indian languages, *viz*, Assamese, Bengali, Bodo, Gujarati, Hindi, Kannada, Kashmiri, Konkani, Malayalam, Manipuri, Marathi, Nepali, Oriya, Punjabi, Sanskrit, Tamil, Telugu, and Urdu. These wordnets have been created using the expansion approach with Hindi WordNet as a pivot, which is partially linked to English WordNet. We exploit these links to create mappings from English WordNet to wordnets of other languages.

3. Resources

In this section, we describe the resources released with our work. We release two primary resources with our dataset which are described in subsections 3.1. and 3.2. below.

3.1. Indian Language WordNets

The creation of IndoWordNet began in 2000 with Hindi WordNet. Due to the complex nature of Indian language families, and many other reasons such as morphological richness, gender information etc. it was decided that Hindi be used as a pivot for linking all the Indian Languages. Hindi shares many common features and borrowed concepts from ancient Indian languages like Sanskrit and is the

	Noun	Verb	Adjectives	Adverbs	Total
Assamese	9065	1676	3805	412	14958
Bengali	27281	2804	5815	445	36346
Bodo	8788	2296	4287	414	15785
Gujarati	26503	2805	5828	445	35599
Hindi	29807	3687	6336	541	40371
Kannada	12765	3119	5988	170	22042
Kashmiri	21041	2660	5365	400	29469
Konkani	23144	3000	5744	482	32370
Malayalam	20071	3311	6257	501	30140
Manipuri	10156	2021	3806	332	16351
Marathi	23271	3146	5269	539	32226
Nepali	6748	1477	3227	261	11713
Odiya	27216	2418	5273	377	35284
Punjabi	23255	2836	5830	443	32364
Sanskrit	32385	1246	4006	265	37907
Tamil	16312	2803	5827	477	25419
Telugu	12078	2795	5776	442	21091
Urdu	22990	2801	5786	443	34280

Table 1: Number of synsets in different wordnets

	Nouns		Verbs		Adjectives		Adverbs		Total
	D	H	D	H	D	H	D	H	
Assamese	7019	679	1300	36	2744	0	294	0	12072
Bengali	11049	7680	1824	99	3356	3	312	0	24323
Bodo	6940	603	1594	64	2854	1	293	0	12349
Gujarati	10910	7533	1825	99	3356	3	312	0	24038
Hindi	11584	8221	1988	212	3542	4	344	0	25895
Kannada	7806	1973	1921	154	3453	3	133	0	15443
Kashmiri	9363	6261	1767	100	3240	2	294	0	21027
Konkani	10545	6952	1888	128	3391	2	328	0	23234
Malayalam	9146	4754	1970	206	3525	4	340	0	19945
Manipuri	7192	823	1324	43	2712	0	244	0	12338
Marathi	9874	6556	1839	144	3092	0	333	0	21838
Nepali	5217	496	1114	42	2202	1	200	0	9272
Odiya	11039	7680	1679	66	3187	2	271	0	23924
Punjabi	10215	6382	1822	99	3355	3	312	0	22188
Sanskrit	8396	6470	1048	28	2873	2	241	0	19058
Tamil	8130	3066	1821	98	3353	3	312	0	16783
Telugu	6944	1843	1819	98	3350	0	312	0	14366
Urdu	10424	6816	1822	98	3356	3	313	0	22832

Table 2: Linkage Statistics for English to Indian Language WordNets. D stands for Direct links, and H stands for Hypernymy links

most commonly spoken language in India. The expansion approach adopted for IndoWordNet creation is:

1. Creation of a Hindi synset with synonymous words.
2. Mapping of the synset with relations such as hypernymy and hyponymy *etc.*
3. Tagging of the synset with an ontological category.
4. Allotment of a unique synset ID to the concept de-

scribed in the synset.

5. Creation of the same synset in the other Indian languages leading to an implicit linkage of relations, ontological categories.

We release the latest data in IndoWordNet with statistics described in subsection 3.1.2. below.

3.1.1. Construction Principles

- **Minimality:** We try to capture the minimal set of words in the synset which uniquely define the concept and ensure that it is identifiable via the use of these words.
- **Coverage:** We also try to stress on the completion of the synset and try to capture all the words which represent the concept.
- **Replaceability:** This principle states that all the words in the synset should be able to replace one another in an example sentence quoted along with the synset. These words must be able to replace each other in the same sense.

3.1.2. Current Statistics: IndoWordnet

Table 1 shows the statistics of the released wordnets. These wordnets have, on an average, approximately 28,000 synsets, with Nepali and Hindi having the minimum and the maximum number of synsets respectively. The number of synsets in Hindi is maximum due to the fact that work on IndoWordNet started with the Hindi language. It should also be noted that the ratio of nouns, verbs, adjectives, and adverbs is also on an average 48:6:13:1; the trend being similar to Princeton WordNet.

3.2. Linkage between English and Indian Language WordNets

For linking Indian language wordnets with the Princeton WordNet, we link the Hindi Wordnet data with Princeton WordNet data manually with the help of lexicographers. This has been an ongoing work since many years, and a resource release was long standing. We delve deep into the language related issues in linking both the languages and ensure that only a valid relation is established between both the lexicons. The principles used and the current linkage statistics are described in the subsections below.

3.2.1. Principles

We use the simple principles of concept representation to ensure a valid linkage between the two languages. While linking two concepts, we refer to all words present in both the synsets for creating the linkage. First, we start with linking the known common concepts between both the WordNets of Hindi and English (Direct Linkages). We, then, start to link Hypernymy linkages from Hindi to English. For *e.g., younger paternal uncle* and *elder paternal uncle* are two different specific concepts, and thus have two different synsets in Hindi language. English language, on the other hand, has only the concept of *uncle*, and hence we link both the Hindi language concepts to uncle as Hypernymy linkages.

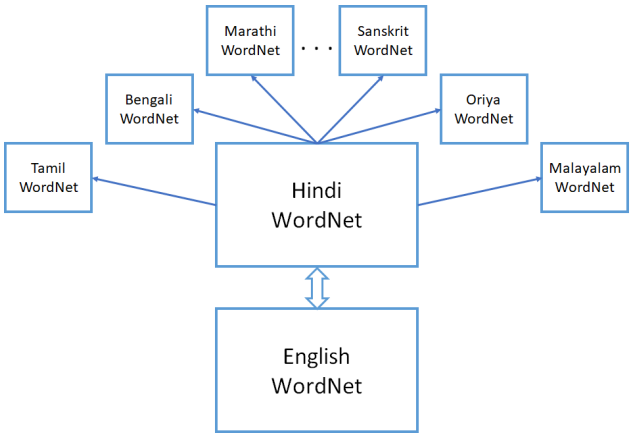


Figure 1: Indian Language WordNet linkages with Princeton WordNet. D stands for links of the type Direct, whereas H stands for the links of the type HYPERNYM.

3.2.2. Princeton Statistics

At present, the Princeton Wordnet has a total of 117659 synsets, with 82115 nouns, 13767 verbs, 18156 adjectives, and 3621 adverbs[3]. They further categorize some of their adjectives into satellite adjectives but the statistics shown include both adjectives and satellite adjectives. We use Princeton WordNet version 3.0 for the purpose of linkage. We began linking Hindi WordNet with version 2.1 and shifted to WordNet version 3.0 using the mappings provided[4] by Princeton WordNet.

3.2.3. Current Statistics: Linkages for Language pairs

Table 2 shows the statistics of the released linkages. There are approximately 20,000 links for an English-Indian language pair on average, with Nepali and Hindi having the minimum and the maximum number of links. Again, the number of links in Hindi is maximum due to the fact that work on IndoWordnet started with the Hindi language, and we link Hindi directly with English. At times, the concept present in Hindi is not present in the other Indian languages thus leading to the less number of linkages for the other languages, in some cases. Table 2 show part-of-speech category-wise distribution of the linked synsets, and also indicated the number of directly linked synsets (D) along with the synset linkages which have been marked as hypernymy linkages (H).

The statistics show our progress in updating IndoWordnet as a resource. The relatively large number of linkages also show that the Indian wordnets have matured considerably.

4. Discussion

Many concepts in the Indian languages are specific to the Indian culture. Thus, their corresponding variant is not available in the Princeton WordNet (and is not likely to be included anytime). Thus, one needs to maintain the translation/transliteration of such notions from Indian languages

[3] https://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

[4] https://wordnet.princeton.edu/man/sensemap.5WN.html

to the English language as a separate bilingual mapping [5]. A similar issue arises in case of proper nouns, which should be present in an Indian lexicon but they are not present in Princeton WordNet. They are also handled using bilingual mappings (Singh et al., 2016). Some of the synsets in Indian languages are too fine-grained and have a common representation in the English language. This is why we use the principle of Hypernymy linkages for linking such concepts. We reserve a set of synset id numbers later for language specific concepts and create them to include in these wordnets, individually. These are not linked to the Princeton WordNet and hence are not included in our resource.

5. Conclusion and Future Work

In this paper, we describe two resources released along with this paper. We discussed the Indian language wordnets that are part of the IndoWordNet project. We enlisted the statistics of the latest version, which we provide as a single bundle along with this paper. Next, we described the linkage process for creating English-Indian language links using English-Hindi language links. We then enlisted the statistics of the latest version of this linked data, which is also provided along with this paper.

In future, we plan to continue building the wordnets and increase linkage. We will also investigate semi-automatic linkage tools such as the ones created by Joshi et al. (2012b), *etc.* so that the workload on our lexicographers and researchers can be reduced to a certain extent

6. Acknowledgement

We would like to acknowledge the work done by lexicographers at Center For Indian Language Technology (CFILT), IIT Bombay without which we would not have been able to link Indian Language Wordnets.

7. Bibliographical References

Balamurali, A. (2012). Cross-lingual sentiment analysis for indian languages using linked wordnets.

Bhattacharyya, P. (2010). Indowordnet. In *In Proc. of LREC-10*. Citeseer.

Bojar, O., Diatka, V., Rychlý, P., Straňák, P., Suchomel, V., Tamchyna, A., and Zeman, D. (2014). HindMonoCorp 0.5.

Bouma, G. (2009). Cross-lingual dutch to english alignment using eurowordnet and dutch wikipedia. In *Proceedings of the 4th International Workshop on Ontology Matching, CEUR-WS*, volume 551, pages 224–229. Citeseer.

Clough, P. and Stevenson, M. (2004). Cross-language information retrieval using eurowordnet and word sense disambiguation. In Sharon McDonald et al., editors, *Advances in Information Retrieval*, volume 2997 of *Lecture Notes in Computer Science*, pages 327–337. Springer Berlin Heidelberg.

Fellbaum, C. (1998). *WordNet*. Wiley Online Library.

Gonzalo, J., Verdejo, F., Chugur, I., and Cigarran, J. (1998). Indexing with wordnet synsets can improve text retrieval. *arXiv preprint cmp-lg/9808002*.

Hovy, E. (1998). Combining and standardizing large-scale, practical ontologies for machine translation and other uses. In *Proceedings of the 1st International Conference on Language Resources and Evaluation (LREC)*, pages 535–542.

Joshi, S., Chatterjee, A., Karra, A. K., and Bhattacharyya, P. U. (2012a). Eating your own cooking: automatically linking wordnet synsets of two languages.

Joshi, S., Chatterjee, A., Karra, K. A., and Bhattacharyya, P. (2012b). Eating your own cooking: Automatically linking wordnet synsets of two languages. In *Proceedings of COLING 2012: Demonstration Papers*, pages 239–246. The COLING 2012 Organizing Committee.

Knight, K. and Luk, S. K. (1994). Building a large-scale knowledge base for machine translation. In *AAAI*, volume 94, pages 773–778.

Lev Finkelstein, Evgeniy Gabrilovich, Y. M. E. R. Z. S. G. W. and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Trans. Inf. Syst.*, 20(1):116–131, January.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. (2002). Multiword expressions: A pain in the neck for nlp. In *Computational Linguistics and Intelligent Text Processing*, pages 1–15. Springer.

Singh, M., Shukla, R., Jha, J., Kashyap, L., Kanojia, D., and Bhattacharyya, P. (2016). Mapping it differently: A solution to the linking challenges. In *Eighth Global Wordnet Conference*. GWC 2016.

Sinha, M., Reddy, M., and Bhattacharyya, P. (2006). An approach towards construction and application of multilingual indo-wordnet. In *3rd Global Wordnet Conference (GWC 06), Jeju Island, Korea*.

TufiŞ, D., Ion, R., and Ide, N. (2004). Fine-grained word sense disambiguation based on parallel corpora, word alignment, word clustering and aligned wordnets. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1312. Association for Computational Linguistics.

Vossen, P. et al. (1997). Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the DELOS workshop on Cross-language Information Retrieval*, pages 5–7.

Yang, D. and Powers, D. M. W. (2006). Verb similarity on the taxonomy of wordnet. In *In the 3rd International WordNet Conference (GWC-06), Jeju Island, Korea*.

[5] Since bilingual mappings are not standardized, we do not release them along with our resources